Futurebus+ Interface Family
Data Manual

Protocol, Arbitration and
Backplane Transceiver Logic

TEXAS
INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments Incorporated (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of reievant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to current specifications in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Please be aware that TI products are not intended for use in life-support appliances, devices, or systems. Use of TI product in such applications requires the written approval of the appropriate TI officer. Certain applications using semiconductor devices may involve potential risks of personal injury, property damage, or loss of life. In order to minimize these risks, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards. Inclusion of TI products in such applications is understood to be fully at the risk of the customer using TI devices or systems.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Printed in UK

PREFACE

This preface presents the benefits of the TI Futurebus+ interface chip set and develops the context for this data manual.

Futurebus+ has rapidly gained wide acceptance throughout the world in multiple-compute paradigms. Historically, the success of backplane buses has been based on multiple sources of standard silicon.

Standard Silicon for a Standardized Bus

Benefits of Standard Silicon

- System design time reduction
- Program risk reduction
- Greater degree of integration
- Accommodates specification complexity
- Joint development opportunities
- Ease of product performance improvement

The Futurebus+ family of specifications is very complex. Therefore, it is not a trivial task to implement these specifications in bus-interface silicon. Texas Instruments, with its long-term involvement in the development of Futurebus+, is ideally suited to meet customer requirements in this area. Through understanding of the specifications, customer feedback, and joint development activities, a highly integrated chip set has been designed. Use of this chip set will shorten system development time significantly. Implementing the interface in a small number of controllers reduces overall risk and provides tremendous real estate savings, allowing more features to be added to each module. Additionally, a standard solution eases the development of new products; product performance may be improved or features added without changing the software interface to the Futurebus+. Second sourcing by Philips Components-Signetics ensures the customer of reliable dual sources for these devices.

TFB2000 Family: High-Performance Silicon Support for Futurebus+

Features of the TI Futurebus+ Chip Set

- Fully compliant to Futurebus+ and related specifications
- Incorporates control and status registers on chip
- JTAG support provided
- Silicon support for loosely coupled and tightly coupled systems
- Support for compelled and packet modes
- Interface to both central and distributed arbitration schemes

JTAG Support Required

Texas Instruments is an acknowledged leader in the development of testability standards. When employing complex integrated circuits in even more complex systems, assistance in testing is a must. Each of the controller devices in the TI Futurebus+ chip set provides JTAG test access ports and boundary scan capability.

Generic Host Interface to Controller Devices

Features

- Designed to support current and future microprocessors from several manufacturers
- Minimal control-signal translation required to interface to processor(s) or secondary cache
- Supports 32- or 36-bit addressing and 32- or 64-bit (multiplexed) data capability
- Provides single, burst, and multiple-burst transfer capability

The host interface, portions of which are provided on each of the Texas Instruments Futurebus+ controller devices, provides a generic host interface, enabling several lines of popular processors to be used with the chip set with minimal glue logic. This interface is used to interconnect processors, local memory, and I/O devices with the Futurebus+ interface logic. Facilities are provided for burst-mode transfers and synchronous handshaking. A detailed explanation of the host interface is included in this manual.

The TI TFB2xxx Futurebus+ Family

TI develops and produces transceivers, arbiters, protocol controllers, and data path devices to meet a wide range of system uses.

Current TI chip set devices are the TFB2002B I/O controller (IOC), the TFB2022A data-path unit (DPU), and the TFB2010 arbitration bus controller (ABC). The TFB2002B/2022A pair fully supports profile B I/O bus usage and significant portions of other profiles. These devices may be used for memory boards or I/O boards in Profile F (cache-coherent) systems.

Associated Documentation

This data manual occasionally refers to IEEE 896.x Futurebus+ standard documentation. Although the data manual is a stand-alone document, the reader may want to order the IEEE 896.x documents by calling the IEEE in New York City (1-800-678-IEEE).

Texas Instruments Futurebus+ Bulletin Board System (BBS)

A computer bulletin board system (BBS) with modem input is available for customer support purposes. The BBS's characteristics and performance are provided in the following table.

TEXAS INSTRUMENTS FUTUREBUS+ BULLETIN BOARD SYSTEM	
CUSTOMER SUPPORT (VIA MODEM) NUMBER: 214-997-3195	
Data Rate	300–14,400 baud
Recommended Protocol	Zmodem
Data, Stop, and Parity	8, none, and 1 or 7, even, and 1
Current Capability	Messages, file read/download/upload
Future Capability	Questionnaire, Internet

If a satisfactory connection to the BBS is unavailable, call the product information center (PIC) at 214-644-5580 for assistance.

Signal Name Notation

To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896. 1–1991), an active-low signal is denoted herein by use of the trailing asterisk (*) on the signal name.

Document Organization

This section is intended to give the reader a feel for the information to be presented in Chapters 1–6 of the data manual and the rationale for the order of presentation. The chip set is described in general first and then in increasing detail. These chapters describe the actual use of the chip set in a chronological order that relates to the order of events that the devices would see in practice, from reset and control-register programming to transactions moving data across the bus.

Chapter 1 Introduction

This chapter presents some history of the Futurebus+ standard and some of the trade-offs considered when architecting the chip set.

Chapter 2 Orientation

This chapter presents an overview of the environment in which the chip set are used and defines some terms that are used throughout the data manual.

Chapter 3 Configuration

This chapter identifies the procedures necessary to bring the chip set online in an application.

Chapter 4 Host Interface

This chapter describes the host interface that provides communications between the chip set and the board controller.

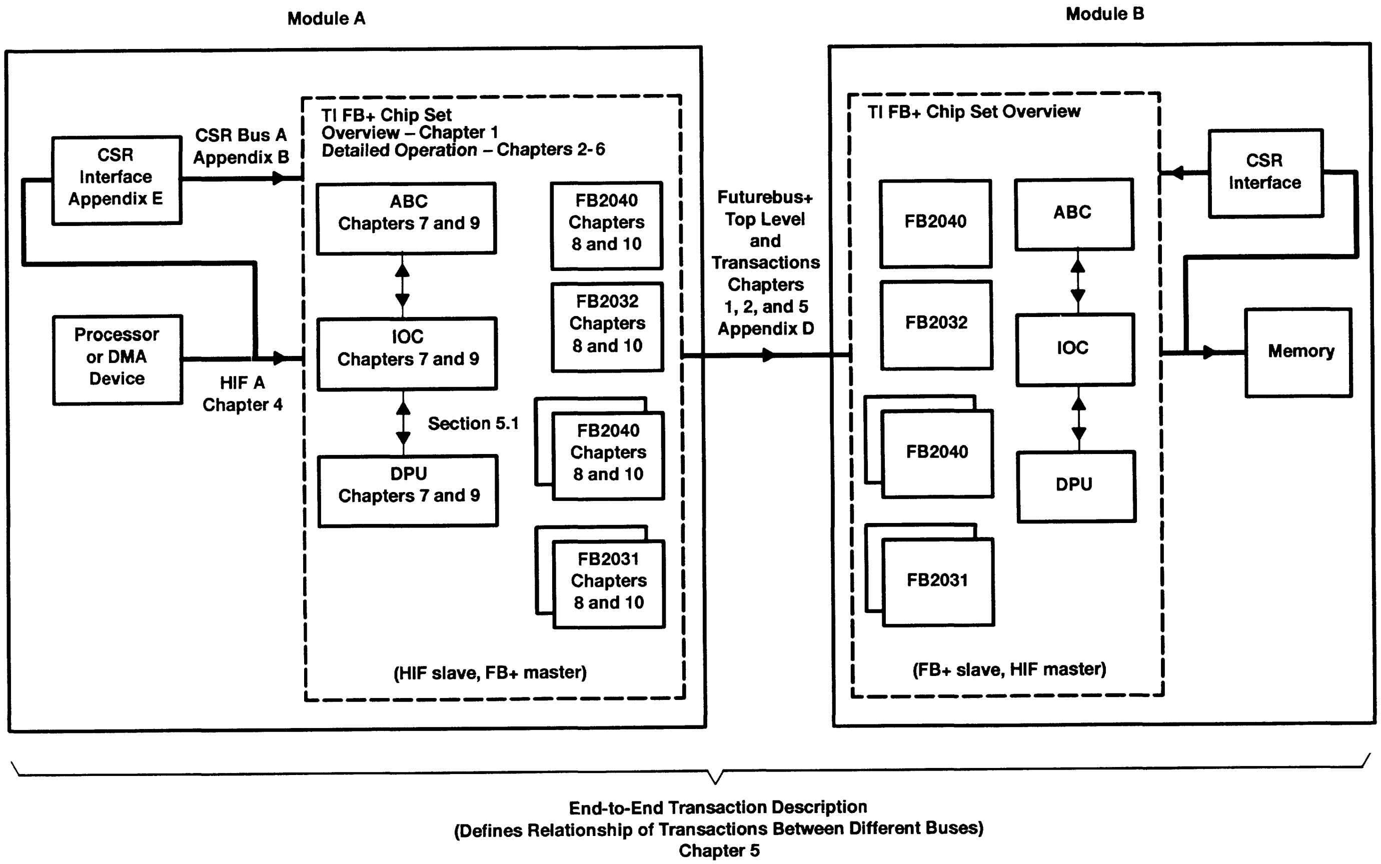
Chapter 5 Transactions

This chapter describes the transactions across Futurebus+ generated from the host.

Chapter 6 Arbitration

This chapter describes the Futurebus+ arbitration protocols.

Guide to the TI Futurebus+ Chip Set Data Manual



Contents

Section *Title* *Page*

Contents (Continued)

Contents (Continued)

List of Illustrations

List of Tables

Chapter 1

Overview

1 OVERVIEW

1.1 Introduction

Futurebus+ (FB+) is an IEEE specification for high-performance backplane-based computing that permits architectural consistency across a broad range of computer products. It is the first comprehensive architectural specification defined as an open standard — that is, an interface standard for which there are no preconceived restrictions in terms of architecture, microprocessor, and software implementations. It is also the first standard explicitly designed to support multiple generations of computer technology, leading to system speeds significantly greater than current systems. FB+ has officially been approved as a standard by the IEEE in September 1991. The foundation of FB+ has clearly been established and FB+ products are proliferating.

FB+ has been in development for over a decade. It is envisioned as the foundation of next-generation computing-system architectures by the U.S. Navy and the VMEbus and MultibusII communities. Due to its advanced architectural features, numerous companies (each a part of the 120-member working group) from the United States, Europe, and Japan were actively involved in the final specification development. These companies, a microcosm of the computer industry, include both system designers and end users of products ranging from desktops to mainframe systems. Texas Instruments has actively supported Futurebus+ development and fills several committee leadership roles.

1.1.1 WHAT IS FB+?

Figure 1–1 illustrates how FB+ provides a backplane interconnection between high-performance computing blocks. This figure demonstrates the considerable breadth of possible FB+ interface applications. To efficiently manage these applications, the specification development is divided into smaller pieces with a team of experts designated to work on each piece. The idea of a collection of FB+ specifications to create a single standard is a departure from other bus standards. The result is a collection of profiles, which allows FB+ to serve wide-ranging applications with each profile fulfilling the needs of specialized segments.

FB+ architecture can be flexibly implemented; however, this flexibility represents a dilemma as to how to minimize unnecessary variations in products developed by different vendors. The profiles are used to address this problem. Profiles tie together the various specifications and support standards to create interoperability. Restrictions in profile specifications are designed to ensure compatibility amongst products. Therefore, the word conformance has no meaning within the IEEE Futurebus+ family of specifications except when used in conjunction with an approved profile. The content of a profile depends on the needs of a particular market and its applications. Table 1–1 lists the current profiles along with their general application and sponsoring group. Profiles A, B, F and M have become IEEE standards, while Profiles C, T, and S remain in Working Group.

Table 1–1. Futurebus+ Profiles

PROFILE	APPLICATION	INTERESTED GROUP
A	General	VITA/VME Community
B	I/O	Mini and Mainframe Computers
F	Workstation	Workstations
C	Cable	Inter-Rack Cabling
M	Military	U.S. DOD and Other Military
T	Telecommunication	U.S., Europe, and Japan
S	Space	Space Systems

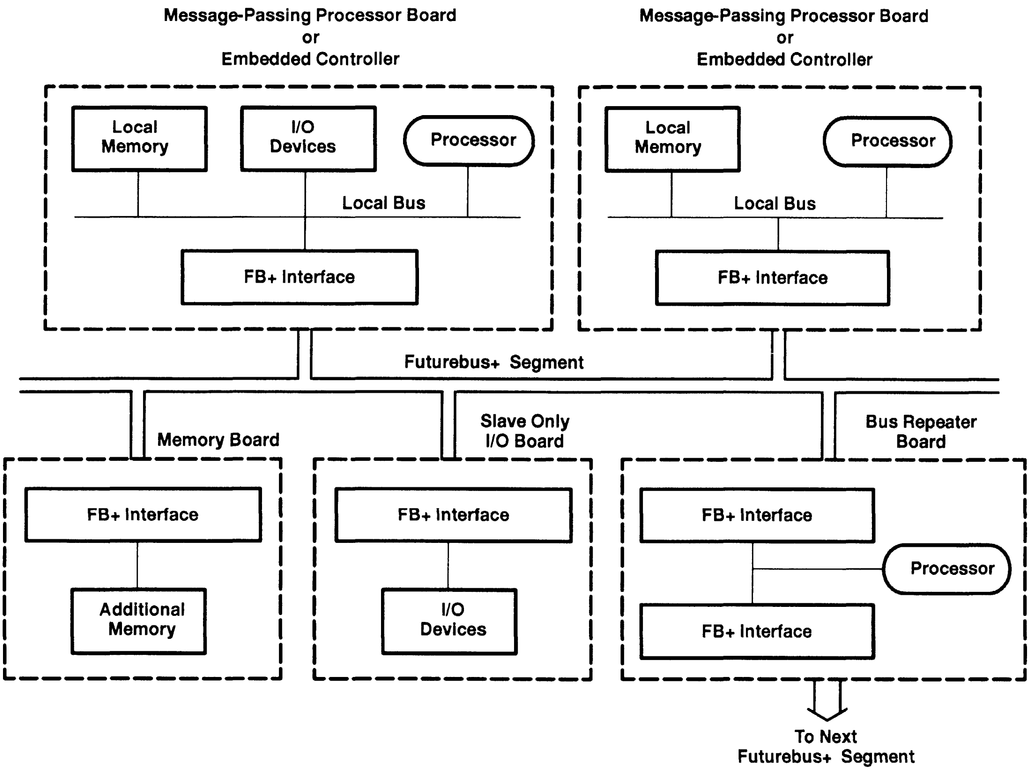


Figure 1–1. FB+ Interface Applications

1.1.2 FB+ IN PERSPECTIVE

FB+ represents a significant step forward in the facilities and performance available to designers of multiprocessor systems. Specifically:

- FB+ has an architecture designed for maximum throughput instead of being optimized for a particular processor.
- FB+ technology independence drives the FB+ principle of no technology-based performance limitation. Configuration and transaction modes produce interoperability when two devices of different speeds or of different generations communicate, providing FB+ with the ability to support multiple generations of computers, well into the 21st century.
- FB+ has timing and bus handshake protocols that are limited by "law of nature" types of constraints rather than requirements of current and projected technology.
- FB+ has provisions for the bus standard to be easily extended in the future.
- FB+ has flexibility to allow the system architecture to be defined by the tasks and resultant software that must be executed.
- FB+ is the first open bus to develop a cache-coherence protocol.
- FB+ is suitable for use in applications requiring high reliability and/or fault tolerance.
- FB+ protocols provide headroom for system growth and explicitly supports real-time scheduling, fault tolerance, and high-availability and high-reliability systems.
- FB+ has logically layered specifications that offer a wealth of architectural features with which designers may implement a wide variety of systems.

- FB+ supports both loosely and tightly coupled compute paradigms via the parallel protocols, cache coherence, and message-passing protocols.
- FB+ control and status registers provide a standard software interface to the FB+, easing development and transportability of I/O drivers and other system software.
- FB+ is optimized for the backplane environment via backplane transceiver logic (BTL) circuits. Note: TI offers a wide variety of BTL products for numerous bus-driving applications.

1.1.3 CURRENT AND FUTURE APPLICATIONS, TRENDS

As previously indicated, the master foundation of FB+ is a reality and products are beginning to proliferate. Profiles A, B, F and M have become IEEE standards, while others are in Working Group. Table 1–2 provides an update on FB+ specification development.

Table 1–2. FB+ Specification Development Update As of December 1993

SPECIFICATION		STATUS/DETAILS
896.1–1991	Futurebus+ Logical Layer	IEEE Standard
896.2–1991	Futurebus+ Physical Layer and Profiles	IEEE Standard Node Management, Profiles A, B, and F
896.3–1993	Futurebus+ Recommended Practices	IEEE Standard
896.4–1993	Futurebus+ Conformance Testing	IEEE Standard
896.5–1993	Futurebus+ Profile M	IEEE Standard
P896.6	Futurebus+ Profile T	Working Group. Specification for telecommunications systems
P896.7	Futurebus+ Profile C	Working Group. Cache-coherent cable bus interconnect
P896.10	Space Applications of Futurebus+	Working Group
P1014.1	Futurebus+ to VME64 Bridge	Review Ballot
1101.3	Conduction Cooled Eurocard	IEEE Standard
1101.4	Specification for Military Module Form Factor	IEEE Standard
1149.1–1990	IEEE Standard Test Access Port and Boundary Scan Architecture	IEEE Standard
P1156	Environmental Specifications for Microcomputers	IEEE Standard
P1156.1	Environmental Specifications for Computer Modules	IEEE Standard
P1156.2	Environmental Specifications for Computer Systems	
P1156.3	Specifications for Computer Power Supplies	
1194.1–1991	Electrical Characteristics of Backplane Transceiver Logic Interface Circuits	IEEE Standard
1212–1992	Control and Status Register Architecture	IEEE Standard
1212.1–1993	Control and Status Register DMA Framework Architecture	IEEE Standard
P1275	Open Boot	Sponsor Ballot
1301–1991	Standard for a Metric Equipment Practice for Microcomputers, Coordination Document	IEEE Standard
1301.1–1991	Detailed Standard for a Metric Equipment Practice for Microcomputers Using 2-mm Connectors and Convection Cooling	IEEE Standard
P1301.2–1992	Recommended Practices for the Implementation of P1301	
P1301.4	SCEM, Mechanical	Working Group
P1341	Multimedia Extensions for Futurebus+	Working Group
P1342	Bus Modeling	Working Group
P1394	High-Speed Serial Bus	Working Group

1.2 Futurebus+ Framework

1.2.1 FUTUREBUS+ PHILOSOPHY AND SYSTEM-LEVEL BACKPLANE

FB+, like other buses, facilitates communications and networking between dispersed units of intelligence. In simple terms, the fundamental purpose of FB+, or any bus standard, is simply to move data around.

FB+ consists of address/data, status, command, handshake, and arbitration lines. FB+ provides a 64-bit architecture with a compatible 32-bit subset and data path extensions to 128 or 256 bits. To keep the amount of signal lines reasonable while supporting a large address space and wide data-path, the address lines are multiplexed with the lowest 64 data lines (AD63*–AD0*). Parallel transfers of 32, 64, 128, and 256 bits at a time are supported to address spaces containing 2^{32} or 2^{64} bytes. Synchronization signals coordinate the broadcast of addresses to modules, transfer of data across the bus, and transfer of bus mastership among modules.

Unlike older standard buses, FB+ is optimized for the backplane environment. To support the highest possible speed, modules and backplanes use backplane transceiver logic (BTL) circuits. The advantage of BTL logic is to provide incident-wave switching capability (thus no settling time) with low capacitance, high current drive capability, and controlled 1-V voltage swings for better noise margins. When a bus signal driver transmits a logic 1 or a logic 0 down the bus, the receiver logic must respond only to the resulting voltage wave as it arrives and not be affected by signal reflections caused by impedance mismatches.

In addition to the bus protocols, the standard defines an organized set of control and status registers (CSRs). The CSR area consists of several memory locations that are used by modules to control and determine various bus and system parameters. This results in a standard way to dynamically control system configuration. A unit architecture (or unit) is a group of CSRs related to a particular function (often an I/O device, such as an UART, SCSI controller, Ethernet port or controller, or a DMA device).

1.2.2 INDIVIDUAL MODULE

A module is a collection of circuitry that performs specific functions that includes an interface to FB+ and connects to one or more slots on the backplane. It is removable from and replaceable in a backplane assembly via connectors. The module is the entity that is removed from or inserted into a FB+ subrack as a unit. FB+ signaling is defined between modules; however, address and data lines (AD<63:0>* and D<255:64>)* actually transmit addresses and data between nodes.

A node includes a set of control and status-register addresses (including identification – ROM and reset command registers), which are initially defined in a 4K-byte initial node address space. A node typically includes a processor with memory as shown in Figure 1–2, which functions independently of actions on other sets of CSRs.

For the TI chip set, the node-side bit in the node IDs CSR is set to 0. Therefore the TI solution allows thirty addressable nodes in a FB+ subrack. FB+ actually defines an address bit called node side that allows two distinct nodes to reside on one module, but the chip set always programs the node-side bit to a 0, so only one node per module is allowed. Note that this manual uses module and node interchangeably.

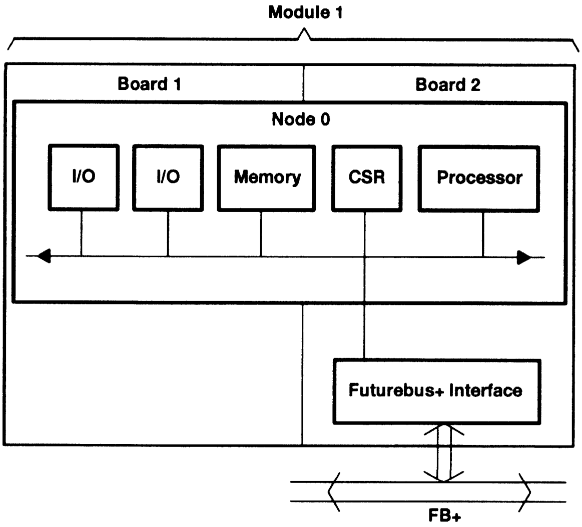


Figure 1–2. FB+ Boards, Modules, and Nodes

A Futurebus+ chip-set layout for a module with one node is shown in Figure 1–3. The CSR bus is used to interface support devices to reduce the overhead requirements of the host interface. The host interface (HIF) interconnects the CSR bus, memory, processor, or cache functions. The TI FB+ chip set then interfaces these interconnected functions to FB+.

1.3 TI Interface Chip Set

1.3.1 CONTROLLER ICs

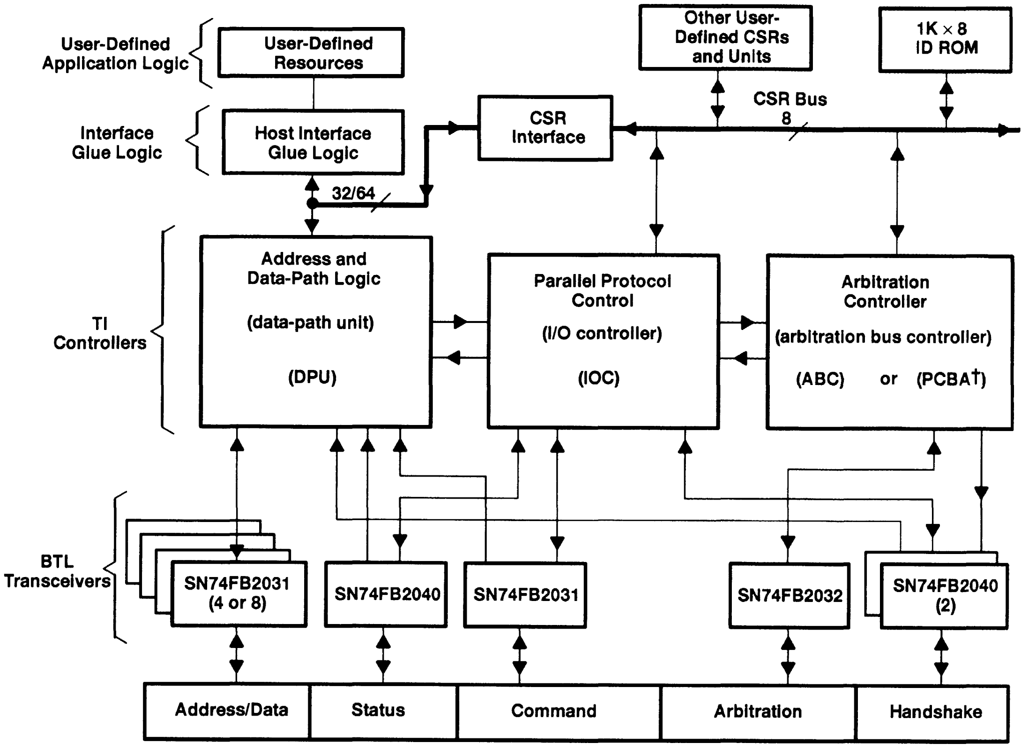
The fundamental purposes of the data-path unit (DPU), I/O controller (IOC), and arbitration bus controller (ABC) are:

DPU:
- data-path buffering
- address decoding
- packet-mode encode/decode

IOC:
- generates timing for the DPU's data-path storage and address-decoding functions
- handles the Futurebus+ and host interface protocols

ABC:
- performs distributed arbitration to gain FB+ tenure
- transfers arbitrated messages and interrupts over the arbitration bus

Figure 1–3 presents an overview perspective on how the DPU, IOC, and ABC fit into the module architecture and provides insight into the HIF and the CSR bus. Note that the CSR bus provides access by the processor (via the HIF) to the control and status registers (CSRs) within the IOC and ABC. Also, the CSR interface translates the 32-bit address/data path on the HIF to a 12-bit address/8-bit data path for the IOC and ABC to minimize their pin count.

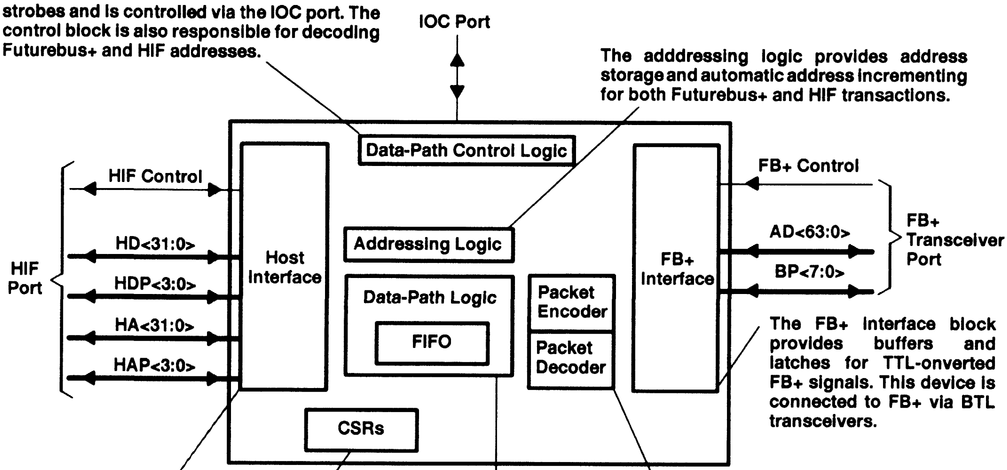


† The PCBA is a central arbiter IC which may be developed in the future.

Figure 1–3. Futurebus+ Chip-Set Layout

1.3.1.1 Data-Path Unit (DPU)

A simple block diagram of the DPU is provided in Figure 1–4.



The internal control block contains a synchronous state machine that controls the flow of data between the HIF, the CSR, and the FIFO. It monitors both the HIF and Futurebus+ strobes and is controlled via the IOC port. The control block is also responsible for decoding Futurebus+ and HIF addresses.

The adddressing logic provides address storage and automatic address incrementing for both Futurebus+ and HIF transactions.

The FB+ interface block provides buffers and latches for TTL-onverted FB+ signals. This device is connected to FB+ via BTL transceivers.

The host interface block provides buffers and latches for HIF signals.

The packet encoder takes data from the data-path logic and supplies an encoded packet to the FB+ interface. The packet decoder receives packets from the FB+ interface and supplies data blocks to the data-path logic.

The CSR block provides control and status registers.

The data-path logic provides storage and an interconnect path for both Futurebus+ and HIF data. It also provides byte-lane encoders and decoders, disconnect data handling, and parity checkers and generators. The FIFO is bidirectional and contains 256 bytes of data. Parity is passed through the FIFO. The data path is designed to accommodate all combinations of 32- and 64-bit transfers on the Futurebus+ and host interface.

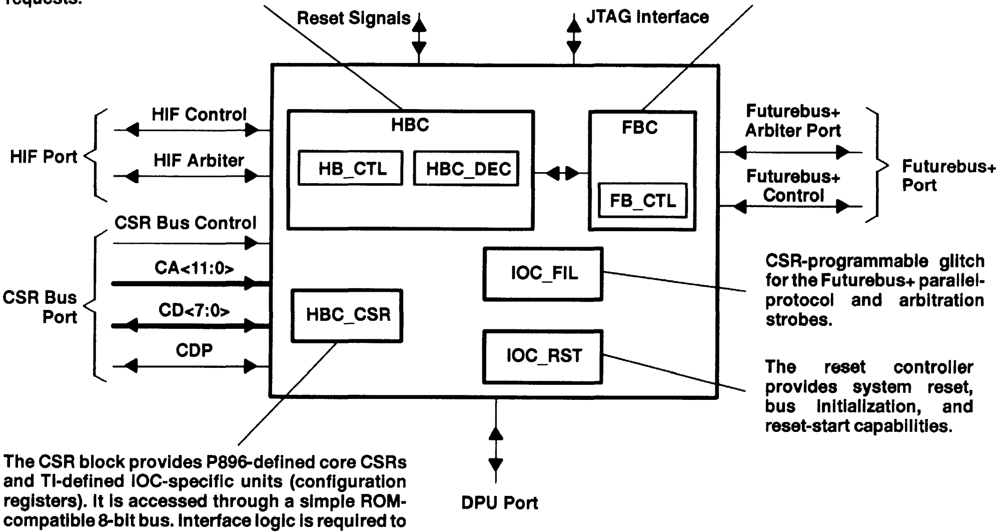
Figure 1–4. DPU Block Diagram

1.3.1.2 I/O Controller (IOC)

A simple block diagram of the IOC is provided in Figure 1–5. The IOC design has an underlying symmetrical aspect. The IOC-to-CSR bus and IOC-to-FB+ each have separate control timing and an associated communication path. Each control timing and communication path also communicates with the DPU. Specifically, the HBC communicates with the CSR bus and DPU, respectively, through the HIF while the FBC communicates with FB+ and DPU through the FB+ port.

The HBC requests bus mastership from both HIF and Futurebus+ arbiters. It handshakes with an external DPU chip to control transfers between the HIF, the CSR, and the FIFO. The HBC uses the HIF control and signaling information, the FB+ command, and the module configuration in the HBC CSR to decide how to best service HIF and FB+ requests.

The FBC provides the Futurebus+ decode and control. An asynchronous state machine handshakes with the external DPU chip to control Futurebus+ to FIFO transfers. It also provides a Futurebus+ phase indication from the HBC. External module configuration and snoop filters also affect the FBC.



CSR-programmable glitch for the Futurebus+ parallel-protocol and arbitration strobes.

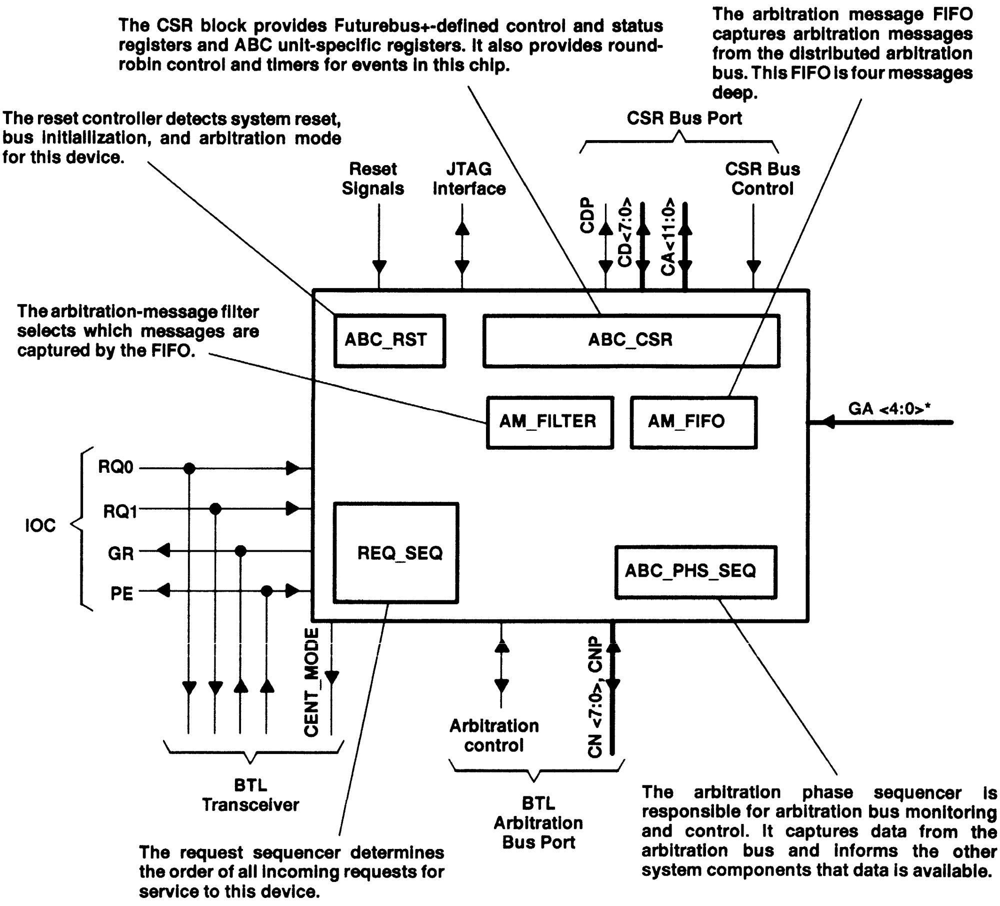
The reset controller provides system reset, bus initialization, and reset-start capabilities.

The CSR block provides P896-defined core CSRs and TI-defined IOC-specific units (configuration registers). It is accessed through a simple ROM-compatible 8-bit bus. Interface logic is required to connect the CSR bus to the HIF.

Figure 1–5. IOC Block Diagram

1.3.1.3 Arbitration Bus Controller (ABC)

A simple block diagram of the ABC is provided in Figure 1–6.



The CSR block provides Futurebus+-defined control and status registers and ABC unit-specific registers. It also provides round-robin control and timers for events in this chip.

The arbitration message FIFO captures arbitration messages from the distributed arbitration bus. This FIFO is four messages deep.

The reset controller detects system reset, bus initialization, and arbitration mode for this device.

The arbitration-message filter selects which messages are captured by the FIFO.

The request sequencer determines the order of all incoming requests for service to this device.

The arbitration phase sequencer is responsible for arbitration bus monitoring and control. It captures data from the arbitration bus and informs the other system components that data is available.

Figure 1–6. ABC Block Diagram

1.3.2 CHIP-SET INTERCONNECT

A complete interconnect diagram for the TI interface chip set is shown in Figure 1–7 on the following two pages . The interface to FB+ is shown on the right side of the diagram, while the interface to the HIF and the CSR bus is shown on the left.

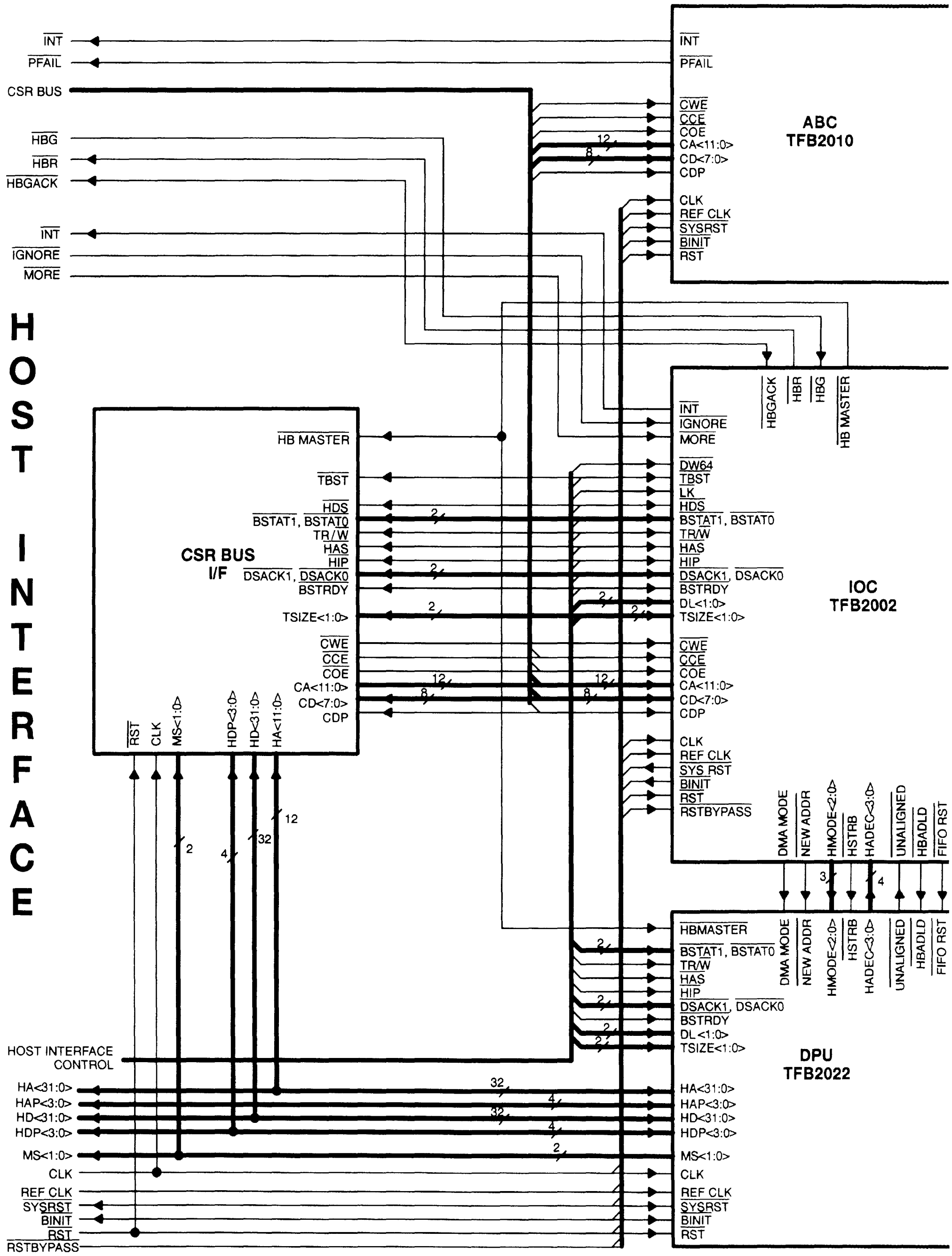
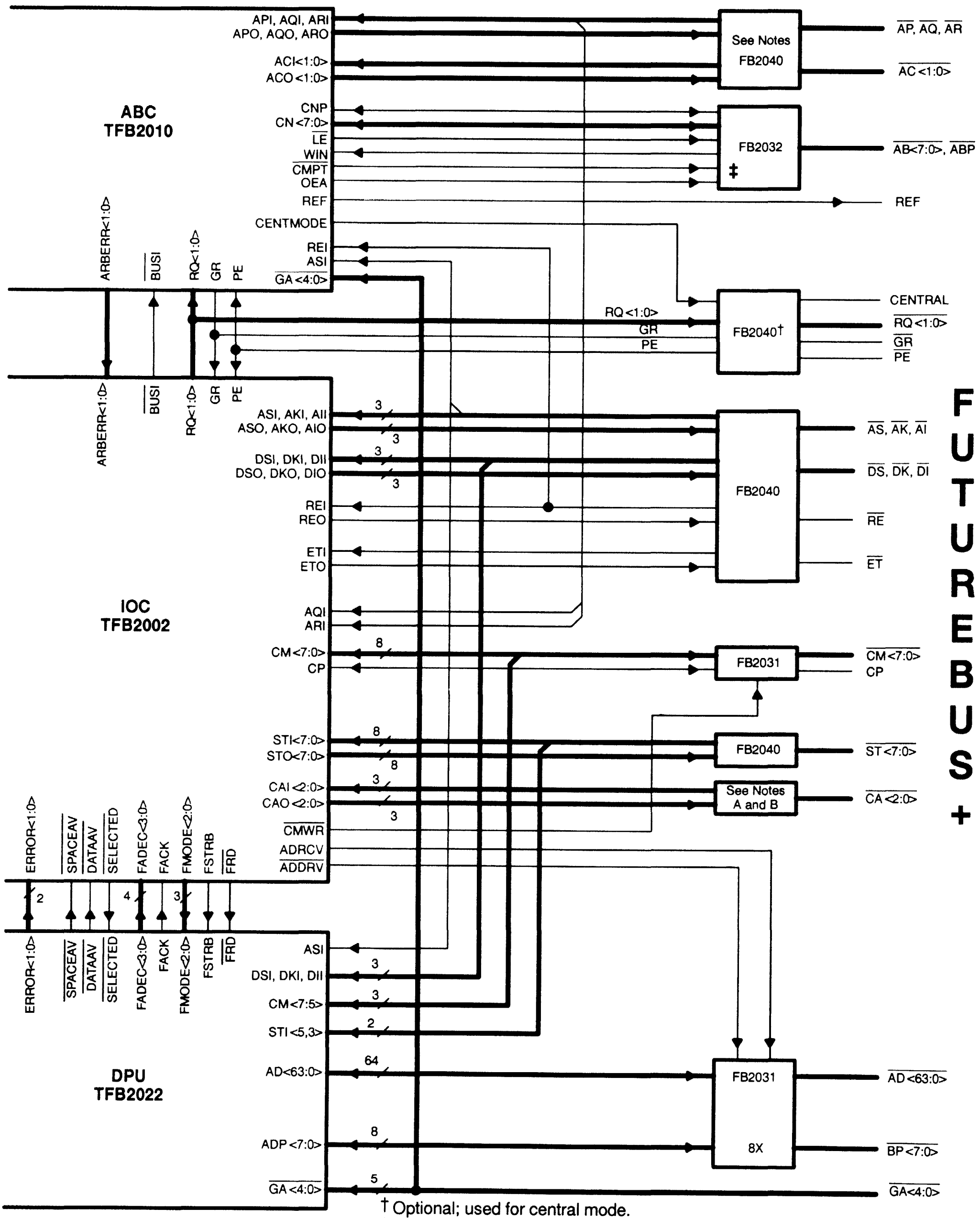


Figure 1–7. TI Interface Chip Set Interconnect Diagram



† Optional; used for central mode.

‡ CMPT from the TFB2010 connects to both COMPETE and OEB on the FB2032.

NOTES: A. The CA<2:0> lines share a transceiver with the AP, AQ, AR lines.
B. The overbar replaces the asterisk indicator for this diagram only; i.e., TR/W = TR/W *.

Figure 1–7. TI Interface Chip Set Interconnect Diagram (Continued)

Chapter 2

Orientation

2 ORIENTATION

2.1 Introduction

This chapter presents a general orientation of the TI Futurebus+ (FB+) TFB20xx controller chip set and some of its capabilities and limitations, as well as a general guideline on how to use the chip set. Information is presented at a relatively high level. Details on any aspect of the chip set are included in subsequent chapters. Since the purpose of any bus controller is the efficient transfer of data between two points, the orientation focuses on how the chip set can be used in this manner. The necessary setup is covered, and some terms that are used throughout this manual are defined.

The chip set services all requests for off-board access from the host interface by generating the appropriate FB+ transaction and conversely handles all incoming requests from FB+ by generating the necessary host interface transaction. The specifics of the host interface are described in Chapter 4. All transactions using the chip set begin with the chip set acting as a slave, either a host slave or a FB+ slave, respectively. In response, the chip set becomes a master of the other side; i.e., a FB+ master or host master, respectively. Thus, the chip set may perform as a host slave and FB+ master for an outgoing host transaction and as a FB+ slave and host master for an incoming FB+ transaction.

2.2 Transfer Example

Figure 2–1 shows two boards connected via FB+ for the purpose of defining the information flow across both the host interface and FB+. Note that the term host is used in this context as a generic term, and it does not imply that any predefined element resides on a board. The two boards (or modules) are arbitrarily labeled A and B. Suppose host A wants to write a block of data to host B. Host A arbitrates for the host interface bus and when it becomes the master it accesses chip set A as a host slave with an address that points to module B. Chip set A recognizes that the address is off the module and arbitrates for the FB+ using the ABC. After winning arbitration, chip set A becomes a FB+ master using the IOC/DPU and places the original address on FB+. Chip set B recognizes the address as within its memory space, and it becomes a FB+ slave. Once chip set A recognizes that a valid FB+ slave has responded, it loads data from the host interface into its internal FIFO and transmits it onto FB+. Chip set B pulls the data off FB+ and stores it in its FIFO and then becomes a host interface master to transmit the data to host B. Finally, status information is passed back from chip set B to chip set A, and the transaction is complete.

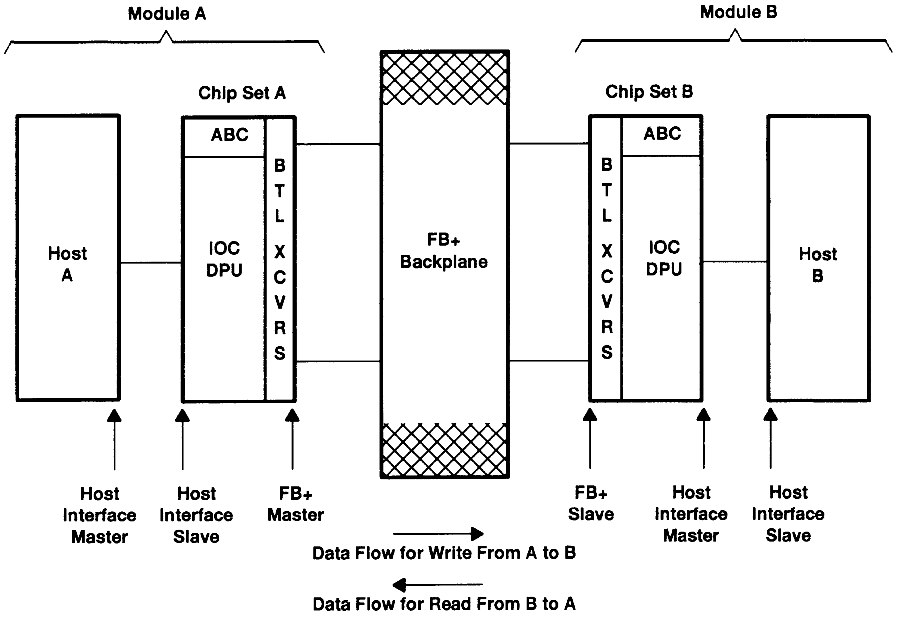


Figure 2–1. Transaction Nomenclature Example

2.3 Transfer Overhead

The address transfer on FB+ is referred to as the connection phase, the data transfer as one or more data beats, and the completion of the transaction transfer as the disconnection phase. (Status information is actually transferred in all three FB+ phases.) The data phase is where the actual work is done. The arbitration, connection, and disconnection phases are all required overhead that is incurred for every transaction. This overhead is referred to as transaction overhead, which is incurred at run time; i.e., it is directly proportional to the number of transactions performed. The other type of overhead encountered using the chip set is programming overhead in which specific parameters that govern the actual transactions are loaded into device registers. In FB+, these registers occupy a portion of the control and status-register (CSR) address space. This overhead is usually encountered only once at system start up.

The arbitration phase incurs a variable amount of overhead depending on the system traffic. Typically, the arbitration for the next FB+ master is performed in parallel with ongoing transactions so most of this time is not seen directly as overhead. Also, the master can execute many transactions under one mastership, or tenure, which requires only a single successful arbitration phase. Exception conditions such as errors and busy conditions can add an arbitrary amount of overhead to the system.

2.4 Command and Status Registers

The CSR space defined by FB+ is a control and status area at a fixed location that provides common configuration, identification, interrupt management, I/O access, and test methods for every module. It contains programmable values that control the actions of nodes on FB+. Some values in the CSRs must be programmed before any data transactions can take place; others power up with default values and may be used with no update necessary. The CSRs can be programmed over FB+ by setting the address in the connection phase to point at a CSR location, in much the same way as data is passed. All CSR data passed over FB+ is 32 bits wide.

The CSR space is partitioned into bus address spaces, and each bus address space is partitioned into node address spaces as shown in Figure 2–2. The TI chip set fully supports the extended addressing model of FB+. The CSR contains 64K nodes that are addressable, each with an initial node space of 4K bytes; thus, the CSR space consumes 256M bytes.

The base address of the initial node space is determined by the node's 16-bit node_id field, which contains a 10-bit bus_id field and a 6-bit offset_id field. The bus_id is expected to be assigned by software when the system is configured. The offset_id consists of the 5-bit geographical ID signals provided by the backplane and the 1-bit node side configuration bit. The TI chip set always programs the node side bit to zero. The addressing is summarized in Figure 2–3.

One bus_id value, 1023, has a special meaning; transactions to the corresponding addresses are accepted by all nodes on the local bus, regardless of their configured bus_id value. This fixed bus address can be used to initially configure the nodes before their actual bus_id has been assigned. For complete systems that use only a single backplane, it is not necessary to program a bus_id value since bus 1023 can be used to access all nodes.

Two offset_id values also have special meaning. Offset_id 63 is defined as the broadcast address for all nodes on the same bus. It can be used to quickly program common values shared by all nodes. Offset_id 62 is defined as the local self_id node; i.e., it is used to read and write CSRs on the local module itself.

This chip set supports only node-side 0, so the node count is limited to 31 per physical bus. The FB+ standard defines control spaces for a total of 1023 buses, each with a possible 62 nodes for a total of 63426 unique nodes. This chip set supports one half or 31 713, which is sufficient for most systems. Bus bridges are needed to link together two or more physical buses.

Each node's space is 4K bytes long, and the registers are addressed with 32-bit words. This space is further divided into CSR core, FB+ dependent (Dep) area, ROM, and initial units space. The core, FB+ dependent, and ROM areas correspond to FB+ CSR definitions. The initial units space is vendor unique. The chip set contains a portion of each of these areas except ROM, as shown in Figure 2–2. The address offset within each 4K block is defined in the individual data sheets for the DPU, IOC, and ABC, and conforms to the mapping shown in Figure 2–2. The first 512 bytes are for the CSR core, which are used to initialize, test, and configure the node. The next 512 bytes are for FB+-dependent registers that are used to tailor a particular bus implementation. The next 1K bytes are used as a window to onboard ROM, which is accessed by the system monarch at start up to determine the memory size and extended units size of the module. The last 2K bytes are for unit-specific registers. Any node can contain multiple logical units and the addressing to each unit's registers is vendor specific. If this 2K-byte region is not sufficient to hold all units needed on a node, an extended units space is defined by FB+ to allow additional storage in the memory space.

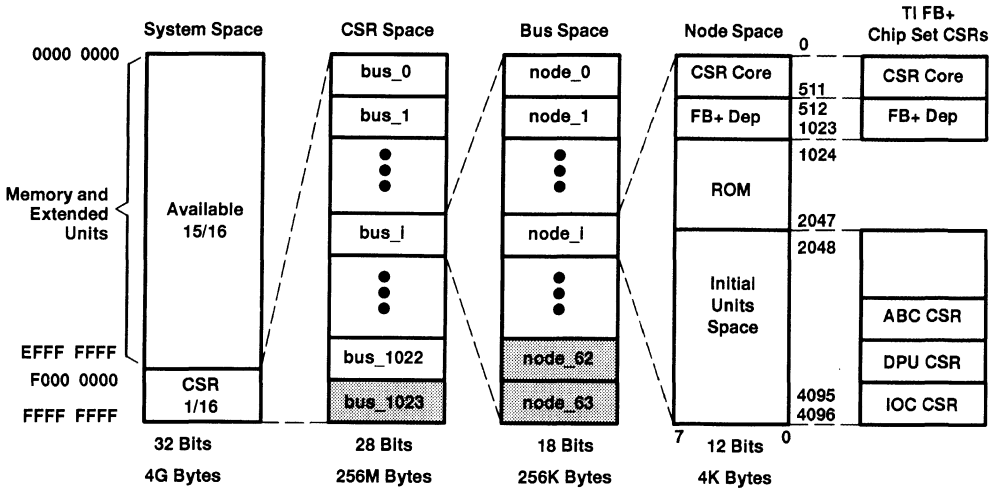


Figure 2–2. 32-Bit Address-Mode CSR Mapping

1111	bbbb bbbb bb	gg gggn	cccc cccc cccc

1111 = base of CSR space in system space
bbbb bbbb bb = represents the bus_id (out of potential 1024)
gg ggg = represents the geographical address (out of potential 32)
n = represents the node side (always 0 in TI chip set)
cccc cccc cccc = represents the offset of the CSR (1024 4 byte registers)

Note: Offset_id field = geographical address (ggggg) and node side (n).

Figure 2–3. Control and Status Register (CSR) Addressing

2.5 System Memory Partitioning

The CSR core contains the BUSID register, the memory base and bounds registers, and the extended units space base and bounds registers. The memory and extended unit space regions can occupy any area in system memory except the 256M bytes set aside for the CSR. These core registers enable the chip set to determine if the module is being accessed or if it needs to gain FB+ mastership to pass a transaction to another module on the FB+. The three areas that can be accessed in FB+ transactions are memory spaces, extended units spaces, and CSR space. The memory and extended units spaces for each node are programmed into the respective registers in the core CSR. CSR access is determined by the bus_id and node_id as described above. From the host interface point of view, an access to any memory address is defined as local if the address falls within the memory space, unit space, or CSR space of that node. Any address outside those areas is defined as remote. A transaction on the host interface is only conveyed to FB+ if the address is to a remote node.

In the transfer example listed earlier, chip set A determined that it needed access to FB+ by comparing the address on the host interface to the programmed address spaces of the chip set. When it determined that the address is not one contained in any of its ranges, it requested FB+ mastership to pass the transaction to the desired module. On the other side of FB+, chip set B sees the address on FB+ as one programmed within its memory space and thus becomes a FB+ slave to accept the transaction. Conversely, when a host slave chip set sees an address within its space, it does not create any FB+ activity. It may become involved in a local CSR transfer if the particular register is contained in the chip set. Otherwise, it does not participate. Similarly, a chip set accessed as a potential FB+ slave that sees an address outside its range does not become involved in the transaction.

2.6 Operating Environment

The chip set serves as a link between two buses, the FB+ and the host interface, which are operating asynchronously with respect to each other. Activity can occur on the host interface independently of activity on FB+ if a given node is not currently serving as the link. This event independence means that the host side can request the chip set to become a FB+ master and before this occurs, the chip set can be selected from the FB+ side as a FB+ slave. The chip set has been designed such that incoming FB+ activity takes precedence over pending host activity to eliminate livelock conditions. Since the incoming FB+ activity has already gone through the FB+ arbitration process, it is also more efficient to allow it to continue over other activity that still needs to win FB+.

Another reason for allowing the FB+ activity to dominate is to allow closure for outstanding split transactions. Profile B allows a FB+ slave to split a requested transaction such that the response occurs during a subsequent transaction, which conserves bus bandwidth when the FB+ slave has a relatively slow response. The splitting FB+ slave for the request transaction becomes the FB+ master for the response transaction, and conversely the request FB+ master becomes the response FB+ slave. An incoming FB+ transaction to a node that is waiting for a split response is likely to be that split response. The TI chip set as a FB+ master supports split requests from a FB+ slave, but the chip set never generates a split request.

2.7 Transfer Capabilities

A brief description has been given of the flow from the host interface to FB+. Some specific capabilities on each side of the chip set are presented to further illustrate the features of the chip set.

The host interface is designed to be compatible with many synchronous interface microprocessors. The chip set can handle both 32-bit-wide data with a nonmultiplexed 32- or 36-bit address or 64-bit-wide data with the high-order 32 bits of data multiplexed with the address lines. Data can be moved in single transfers or burst-mode transfers of 8, 16, 32, or 64 bytes. Single mode supports partial transfers of 1, 2, or 3 bytes. Burst mode supports two transfer speeds, a fast mode that performs a transfer on every rising edge of the clock, and a slow mode that inserts at least one clock-cycle wait state in every transfer. Parity for each data byte is checked or generated.

The FB+ interface supports both the compelled mode and high-performance packet mode of operation. The chip set contains a 64-bit-wide data port (plus parity) FB+ interface and can support partial transfers as well. Support is provided for 32-bit and 36-bit-wide addresses. The 36-bit addressing on FB+ is actually 64-bit mode with the high-order 28 bits set to zero.

Transfers on one side of the chip set are mapped to the most efficient transfer on the other side in an attempt to maximize performance. In general, the chip set attempts to use burst mode for host interface transfers and packet mode for FB+ transfers because these are the most efficient. Some limitations apply. For instance, the host single-mode transfers are always mapped to FB+ compelled transfers, and host partials are always mapped to FB+ partials. CSR programming through the chip set is a case of host interface single mode to FB+ compelled mode; these transfers are always performed using a 32-bit-wide word in a single transfer.

Chapter 3

Initialization and Configuration

3 INITIALIZATION AND CONFIGURATION

3.1 Introduction

This section covers the various system initialization and configuration options that are available to the module designer using the TI chip set. The user is guided through the operation of a system from the time it is powered up until it is powered down. This includes system power up, live insertion, initial conditions, monarch support, node configuration, the reset protocol, and safe power-down support. The sequence of events and status of a node is summarized in Figure 3–1.

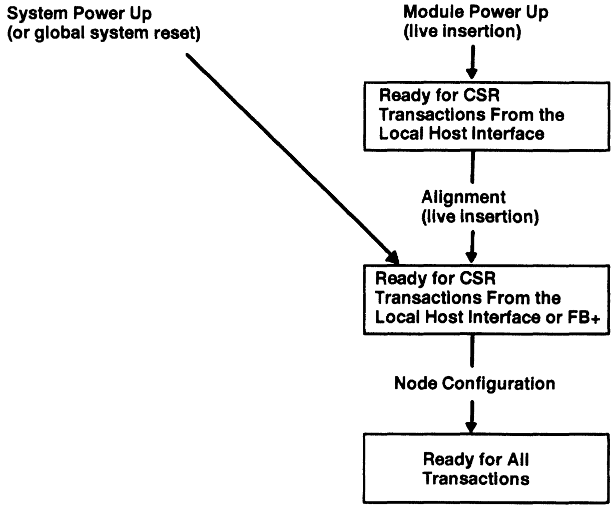

System Power Up
(or global system reset)
Module Power Up
(live insertion)
Ready for CSR Transactions From the Local Host Interface
Alignment
(live insertion)
Ready for CSR Transactions From the Local Host Interface or FB+
Node Configuration
Ready for All Transactions

Figure 3–1. Various System Initializing and Configuration Options Using the TI Chip Set

3.2 Power Up

Normally, power is applied to a module when the user flips the power switch to the system. Each module is reset and configured at this point. In addition to the normal power-up reset and configuration, the TI chip set supports a module that is inserted into a system after the system is powered up. This is called live insertion. Supporting live insertion requires a robust power-up reset mechanism in addition to several special CSR functions. The TI chip set utilizes the FB+ reset signal RE* in addition to four local signals (RST*, SYSRESET*, BINIT*, and BUSI*) and a reset control CSR to perform the reset operations required by the module. The detailed operation of the reset mechanism is described in Section 3.6.

3.2.1 SYSTEM POWER-UP RESET

When power is first applied to the system, all the modules connected to the system must be reset and brought to a known state before transactions may begin. The initial reset that is generated following power application to the system is referred to as system power-up reset. There are two recommended mechanisms for resetting all modules during system power-up reset with the TI chip set. This provides the designer using these parts with maximum application diversity.

- An agent that detects the condition of the power supply (also called the power supply module) may assert RE* (low) long enough to cause power-up reset.
- One module resets itself and then performs system reset by writing to the system reset bit in the reset control CSR of the IOC.

3.2.2 MODULE POWER UP DURING LIVE INSERTION

Each live insertable module must provide RST* asserted (low) to the TI chip set during module power up to reset all state machines, hold all bidirectional signals in the high-impedance state, and prepare for alignment. Following the release of RST* (high), the TI chip set starts counting to automatically align itself to the bus. After 130 ms, the chip set aligns itself to the bus. Automatic alignment is especially useful for slave-only modules that are live inserted. If the module requires longer than 130 ms to initialize, it may set the alignment-wait bit in the reset control CSR until the module is initialized.

3.3 Initial Conditions

Following the assertion of RST* and the power up of the module, the TI chip set is in a known state. All state machines are cleared and all FB+ and host handshake signals are released. Incoming Futurebus+ signals other than RE* are ignored. Incoming host transactions can be serviced to internal CSRs only. Once system power-up reset or alignment is complete, the chip set enters a state where it can participate in FB+ transactions as a slave in a minimal way; it has not yet been configured to understand the memory partitioning, for example. At this point, the FB+ handshake signals are operational and the device can service CSR transactions. The initial conditions of the CSRs are shown in Tables 3.1 and 3.2. All the required functionality to program the node configuration is available.

Two methods for initially accessing FB+, described below, become available after an alignment or global reset. Method 1 is simply to attempt FB+ access and rely on the chip set's automatic insertion of wait states into the host transaction until FB+ becomes available. By attempting to access FB+ with a remote CSR access rather than with a memory access, the need to check the status of FB+ is avoided. Method 2 is to have the host poll the module's master enable bit in the logical module control register. If set, FB+ is available. Justification for this method is that the monarch programs all modules' CSRs first, including memory base and memory bound CSRs, and then sets the modules' master enable bits. Private memory is immediately available to the host by asserting the IGNORE signal, assuming that a module's related memory decode logic has been implemented.

Table 3–1. CORE CSRs and Futurebus+ Dependent CSRs Following Assertion of RST*†

CSR REGISTER	CSR DEFAULT CONDITIONS
State Clear	State bits reset to running
Node IDs	Bus address = 1023; geographical address = no change; node side = 0; priority = 0
Reset Start	Not affected by RST*
Split Timeout Lo	Infinite time-out value
Test Start	No test implemented
Test Status	Maximum run time of default tests shall not exceed 10 seconds; manual, system, extended, and initialize tests may not be performed
Units Base Units Bound	No node extended unit addresses available
Memory Base Memory Bound	No node memory available (availability of memory address space is indicated by the ext adr cap bit in the node capabilities ext rom CSR).
Target Interrupt Value Set	No interrupts are queued.
Target Interrupt Mask	No writes to the target interrupt value set are accepted as an interrupt.
Error Hi Error Lo	No bus specific errors are present.
Logical Common Control	Central mode arbitration is supported if a central arbiter is present. Multiple packet transfer mode is disabled and all packet lengths are disabled. Packet transfer speed is not supported, and the module cannot send distributed arbitration messages. The module may not split bus transactions.
Logical Module Control	A non-cached data width of 32 bits is supported, and cache transfer data widths are not supported. Message transfer data widths are not supported, and a CSR access data width of 32 bits is supported. The node has no local clock, and the node does not contain the system reference clock. The maximum compelled data length is unrestricted, and packet mode is disabled. The node may not use 64-bit addresses.

† All time delay and time-out values are based on a 40 MHz reference clock, unless noted.

Table 3–1. CORE CSRs and Futurebus+ Dependent CSRs Following Assertion of RST* (Continued)†

CSR REGISTER	CSR DEFAULT CONDITIONS
Bus Propagation Delay	Set to 14.7 ns, which is used for (wire-ORed) integrator and arbitration settling time delays
Competition Settling Time	Set to 358 ns
Transaction Timeout	Set to 122 µs, which is used by the master to determine when to set the transaction time-out bit in the error hi CSR
Message-Passing Select Mask	All CSR locations between 128 and 191 cannot receive a broadcast message.
Busy Retry Counter	A single busy results in the busy retry threshold exceeded bit being set in the error lo CSR.
Busy Retry Delay	The busy retry delay time is 0.

† All time delay and time-out values are based on a 40-MHz reference clock.

Table 3–2. Unit CSRs (Memory Addresses 2048–4092)

CSR REGISTER	CSR DEFAULT CONDITIONS
RQ0 Priority	If RQ0 is asserted, 00H is used as the arbitration number in competition for the bus.
RQ1 Priority	If RQ1 is asserted, 80H is used as the arbitration number in competition for the bus.
Send Arbitration Message/Message FIFO	No arbitration message is sent since this register is not written to upon reset.
ABC Interrupt Status Clear	All interrupts are cleared.
ABC Interrupt Status Set	All interrupts are clear.
ABC Interrupt Enable	All interrupts are disabled.
ABC Interrupt ID	No interrupts are pending.
Target Interrupt Clear	All interrupts are cleared.
Target Interrupt ID	No interrupts are pending.
ABC Configuration	Byte 1, bit <0> of this register is only applicable to the programmable central bus arbiter (PCBA). The glitch filter is not bypassed and it assumes a 40-MHz–50-MHz reference clock is being used. Standard CSR reads of the logical module control, logical common control, and bus propagation delay CSRs are ignored. Arbitration competition decisions are not examined for errors, and the power-fail arbitration message is FF. A FIFO overflow error is reported through the error lo CSR, and the message that caused the overflow is lost. The arbitration controller does not stall the protocol if the FIFO is full and wait for the FIFO to be read before allowing the arbitration bus protocol to continue. PE is set when a module of equal or higher priority requests the bus, and round-robin fairness is enabled. The deadman timer is disabled, and duplicate received messages that are already in the message FIFO are stored in the FIFO if possible.
Arbitrated Message Control	All masks are disabled, so all messages are accepted.
Optional Arbitration Settling Time	The arbitration settiing time CSR is used to determine the competition settling time.
Master ID	Not meaningful, since there is no master or master elect upon reset
Arbitration Bus Monitor	Not meaningful, since used for test purposes
Module Configuration	In the central arbiter, these registers correspond to what the individual arbiter registers are initialized to (all RQ1s are initialized to 80 and all RQ0s are initialized to 00).
DPU Configuration	Internal DPU CSR decoding is enabled, and the HIF is in the 32-bit address mode. The DPU checks incoming HIF data and FB+ parity. The bypass glitch filter is disabled.
DPU Status	None of the events, which this register's bits signify, has occurred.
IOC Reset Control	Bits for triggering alignment, bus initialization, and system reset are not activated, and the alignment wait bit for delaying alignment is not activated.
IOC Configuration	A parity check on CSR bus data is not performed. The module test capability is not enabled, so a write to the test start CSR does not cause the test status CSR to go to the checking state. A 32-bit host data-width bus is enabled, and the glitch filter is not bypassed. The data length of the host bus is 8 bytes. The slow burst mode with one wait state between each data is disabled. Bit <0> is reserved.

Table 3–2. Unit CSRs (Memory Addresses 2048–4092) (Continued)

CSR REGISTER	CSR DEFAULT CONDITIONS
IOC Status Clear	None of the events, which this register's bits signify, have occurred. (The register byte for receiving a locked command is cleared. Byte 1 is reserved. Alignment has not occurred, and the state CSR has not been written to. A locked command has not been received, the message mailbox has not been accessed, a host bus-parity error has not occurred, the test start CSR has not been written to, and a split response has not been received).
IOC Status Set	No attempt is made to set the TFB2002 status interrupt CSR bits.
Error Set	No action is taken to set any of the error bits in the error hi or error lo CSRs.
IOC Interrupt Enable	All interrupts are disabled.
IOC Interrupt ID	Since there are no interrupts enabled in the default condition, there are no interrupts pending.
Lock Command Extension	When the LK* signal is asserted, the 000 locked command is ORed with the LKFLD 2, 1, 0 inputs to determine the locked field in the FB+ command.

3.4 Monarch Support

The monarch is the processor that is selected to manage the initialization and configuration of all modules on a bus. After power up, built-in-self-test (BIST), and the release of REI, a single processor, is selected to perform the monarch duties. The selection of a monarch may utilize hardware or software mechanisms. Two mechanisms are recommended for monarch selection.

- Hardware: The module located in slot 1 (for example) is the monarch. This simple mechanism causes a slot dependence that may be unacceptable for some applications. For those using a central arbiter, slot 1 is already slot dependent and may contain the monarch processor along with the central arbiter.
- Software: Following the power-up reset, all potential monarch modules send an arbitration message containing their own geographical addresses. The first message received identifies the winning monarch. This mechanism allows the monarch module to be placed in any slot.

The TI chip set supports both mechanisms by the inclusion of several features: self-ID node address, node ID CSR, send arbitration message/message FIFO CSR, and the receiver-FIFO-not-empty bit. The self-ID node address is an address used by the TI chip set to identify local CSR resources prior to the time when software knows which slot it is in. The node ID CSR is a register that contains the node's slot ID in the form of a geographical address (GA) and bus ID. The send arbitration message and message FIFO CSRs are registers located at the same address where writes are serviced by the send register and reads are serviced by the FIFO. The receiver-FIFO-not-empty bit is a bit in the ABC interrupt status CSR, indicating that all arbitrated messages have been received.

The self-ID node address (62) can be used by the monarch to access its own CSRs prior to determining its own geographical address from the node ID CSR. The self-ID address is also useful during run time to prevent the need to know what slot the software is in when programming local CSRs.

The two ABC CSRs mentioned are useful to monarch selection method 2. In this case, all monarch modules configure their own modules and send an arbitration message containing their own geographical address and a code indicating that the message is a monarch competition message. This is done by writing the configuration monarch competition message (2 code bits (10) followed by the geographical address) to their ABC send arbitration message CSRs (3720). When the receiver-FIFO-not-empty bit is set, the first message in the ABC message FIFO contains the identity of the module containing the monarch process.

3.5 Node Configuration

The monarch must consider several levels of node configuration. The monarch must evaluate node and system capabilities such as arbitration type, address regions, timer values, and bus delay values for optimal system performance. The configuration values of a monarch, processor module, and slave-only module all differ somewhat. Likewise, each system has different configuration values depending on the system design. The goal of configuration by the monarch is to establish a Futurebus+ parallel protocol configuration that 1) supports basic communication among all nodes in the system, and 2) supports optimized communication among modules compliant to a particular profile.

3.5.1 MINIMUM CONFIGURATION

With the TI chip set on a module, no local module programming is required. The monarch is capable of setting up all modules in the system through Futurebus+ transactions. Following reset, the TI chip set is capable of receiving Futurebus+ transactions to local CSRs without programming any CSR registers. This allows the monarch to read ROM IDs and configure the system. Slave-only modules are not required to be configured beyond the initial reset value. In many cases, the default value of a CSR is sufficient without additional programming.

In order to master Futurebus+ transactions, two registers must be programmed: arbitration type (logical common control) and master enable (logical module control). These registers can be programmed with host or Futurebus+ transactions. The monarch programs itself over the host interface, while other master capable modules are programmed by the monarch using Futurebus+ transactions.

If the master enable bit in the chip sets' logical module control register is cleared, the chip set does not request the FB+ to perform a remote transaction but inserts wait states in the host transaction indefinitely. If a monarch attempts to enable this bit from FB+ while the chip set is inserting wait states on HIF, the HIF transaction is backed off/retried (BSTAT1*, 0* = LL) so that the monarch may proceed. The chip set then honors the host's FB+ request on the next HIF transaction.

3.5.2 MEMORY CONFIGURATION

Modules that contain shared memory resources and extended units require additional programming beyond the minimum configuration. The memory base and memory bounds registers define the addresses that must be serviced by the local module. The extended units base and extended units bounds registers define where extended local units are accessed. The monarch programs these registers based upon ROM entries located in the ROM ID region of the CSR space for the local module. The TI chip set CSR bus accommodates an EEPROM implementation that can be used to implement this ROM ID. Once the base and bounds registers have been programmed by the monarch for memory and extended units, this chip set responds as a selected slave to all Futurebus+ transactions addressing a location within these spaces.

3.5.3 TRANSFER ATTRIBUTES

Transfer attributes can be programmed by the system monarch to fine tune the modules to their collective optimal performance. CSRs such as the bus propagation delay, logical module control, logical common control, RQ0 priority, RQ1 priority, competition settling time, and configuration registers in each of the chip's unit architectures provide the monarch great flexibility in controlling the TI chip set transfer capabilities.

3.5.4 TIMERS (busy, error, split, transaction)

The TI chip set provides timers with CSR interfaces for controlling busy retry, split time out, and transaction time outs. The module designer or software must provide the resources required to perform error recovery. Error recovery is application dependent and is therefore beyond the scope of this chip set.

3.5.5 INTERRUPTS

The TI chip set provides interrupts for status and error conditions. All profile-B-required module interrupts are provided through the IOC. All system interrupts use the targeted interrupt mechanism. The targeted interrupt mechanism is provided on the ABC. Other interrupt capabilities can be added using the interrupts provided by the ABC. The DPU contains status and address information that may be useful in error recovery and debugging.

All interrupt conditions are maskable. If an interrupt-enable bit is set (1) in the interrupt enable register, then an interrupt is generated. The corresponding status bit is set regardless of the state of the interrupt-enable bit.

All interrupt-status bits can be individually set or cleared by software using special-purpose write-one-to-clear and write-one-to-set registers. If a bit being written is a one for a write-one-to-clear register, then the corresponding bit in the register is cleared; if the bit being written is zero, the corresponding bit remains at its prior value. Conversely, if a bit being written is a one for a write-one-to-set register, the corresponding bit is set; if the bit being written is zero, the corresponding bit remains at it prior value.

A priority encoding of all the presently set interrupts are provided on both the IOC and ABC. This provides the software with an ID vector that may be used to service interrupts, greatly simplifying error recovery mechanisms.

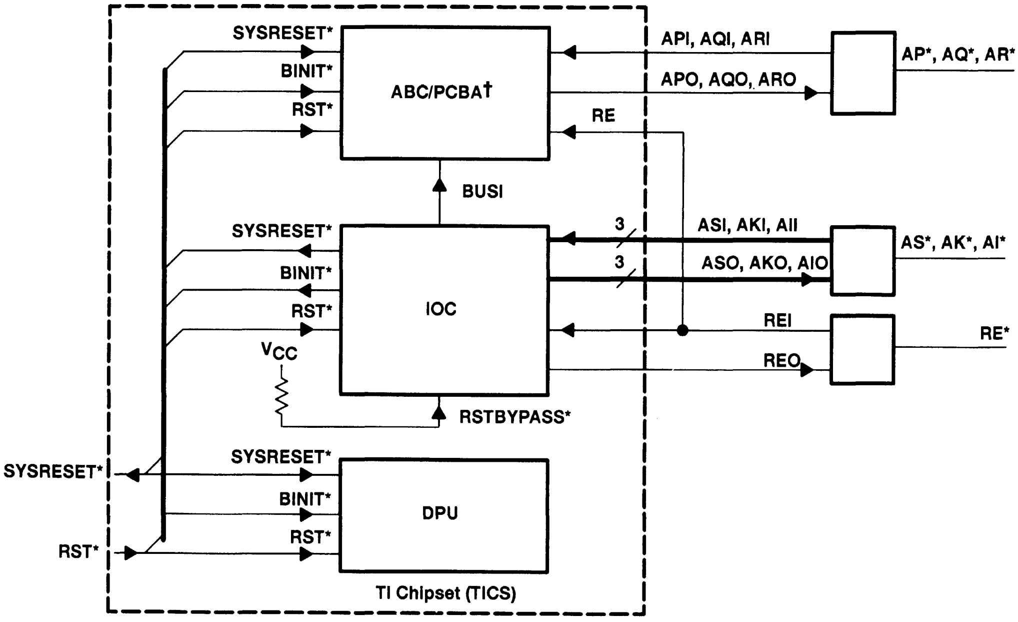
3.5.6 TESTING

State, test start, and test status registers are provided on this chip set to allow any processor in the system to initiate local diagnostics. These registers conform to profile B during the various reset conditions. After reset, they are readable and writable by the system. Writing to the test start register triggers an interrupt if the corresponding interrupt-enable bit is set and cause the test status register to go to the checking (10000) state or remain in the zero state (0000) indicating that no test is implemented. The module software is required to write the appropriate test status value to the test status register when the test is complete if test capabilities are implemented. The IOC knows whether test capabilities are implemented via a bit in the IOC configuration register (4036).

Silicon testing is provided through the test port registers on the various chips within the chip set. These are not intended for use by the module designer. Do not write to these bits; unpredictable results may occur if these bits are written during operation of the system.

3.6 TI Chip Set Reset Protocol

The TI Futurebus+ (FB+) chip set provides a flexible and wide range of reset protocols to support various kinds of system management schemes. In addition to the power-up and live-insertion protocols mentioned in Section 3.2, the TI chip set also supports alignment, bus initialization, global system reset, and local system reset. The pertinent pins for the reset protocols are shown in Figure 3–2.

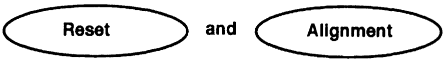


† The PCBA is a central arbiter IC which may be developed in the future.

Figure 3–2. Block Diagram of Pertinent Lines for Reset Protocols of TI Chip Set

3.6.1 RESET AND ALIGNMENT PROTOCOL SUBFLOWS

Two subflows are commonly used throughout the reset protocol; these are shown in Figure 3–3. Rather than duplicating the detail of Figure 3–3, the following symbols are used when presenting the reset protocol for simplification purposes:



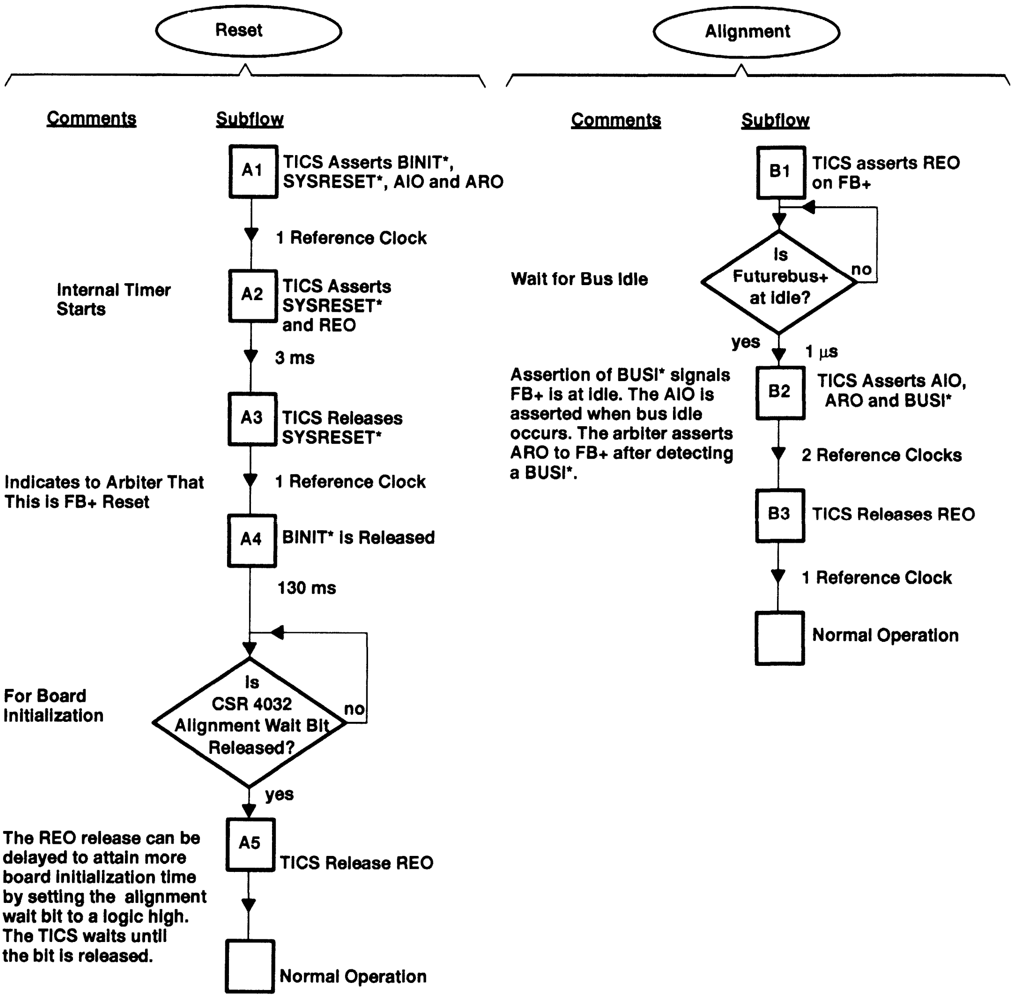

Reset
Comments
Subflow
A1
TICS Asserts BINIT*, SYSRESET*, AIO and ARO
1 Reference Clock
Internal Timer Starts
A2
TICS Asserts SYSRESET* and REO
3 ms
A3
TICS Releases SYSRESET*
Indicates to Arbiter That This Is FB+ Reset
1 Reference Clock
A4
BINIT* Is Released
130 ms
For Board Initialization
Is CSR 4032 Alignment Wait Bit Released?
no
yes
The REO release can be delayed to attain more board initialization time by setting the alignment wait bit to a logic high. The TICS waits until the bit is released.
A5
TICS Release REO
Normal Operation
Alignment
Comments
Subflow
B1
TICS asserts REO on FB+
Wait for Bus Idle
Is Futurebus+ at Idle?
no
yes
1 µs
Assertion of BUSI* signals FB+ is at idle. The AIO is asserted when bus idle occurs. The arbiter asserts ARO to FB+ after detecting a BUSI*.
B2
TICS Asserts AIO, ARO and BUSI*
2 Reference Clocks
B3
TICS Releases REO
1 Reference Clock
Normal Operation

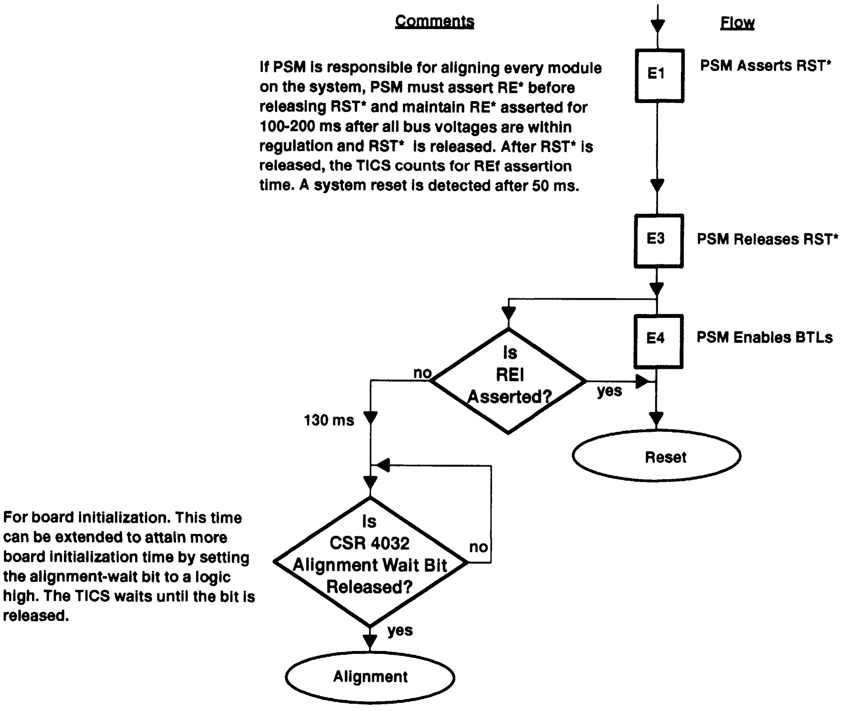
Figure 3–3. Subflows That are Common to Several Protocol Flows

3.6.2 SYSTEM POWER-UP PROTOCOL

The system power-up protocol flow is shown in Figure 3–4. As described in Section 3.2, a power-supply module (PSM) may be assigned the task of initializing every module in the system by asserting RST* and RE* on the FB+. Once RST* is released, the TI chip set starts counting the time that REf is asserted. A FB+ system reset is detected after REf is asserted for 50 ms. The TI chip set responds by asserting AIO, ARO, and REO to FB+ and BINIT* and SYSRESET* to the module. In the event that a PSM is not capable of asserting RE* on FB+ to induce a global system reset, the TICS can perform the equivalent task by setting the system reset configuration bit in CSR address 4032 to a logic high. A global system reset is performed, and every module is aligned upon completion. Refer to Figure 3–5 under global-system reset for the protocol flow.

3.6.3 LIVE-INSERTION/AUTO-ALIGNMENT PROTOCOL

The live-insertion protocol flow is shown in Figure 3–4. If the RE* line on FB+ is not asserted following the release of RST*, the TI chip set assumes that it has been live inserted into a system and must perform an alignment to get on the bus. It waits 130 ms for board initialization before initiating alignment by asserting REO on FB+. This automatic alignment can be delayed by setting the alignment wait reset-control configuration bit, CSR address 4032, to a logic high. This is tailored for those modules that require more than 130 ms before initiating an alignment to FB+.


Comments

If PSM is responsible for aligning every module on the system, PSM must assert RE* before releasing RST* and maintain RE* asserted for 100-200 ms after all bus voltages are within regulation and RST* is released. After RST* is released, the TICS counts for REI assertion time. A system reset is detected after 50 ms.

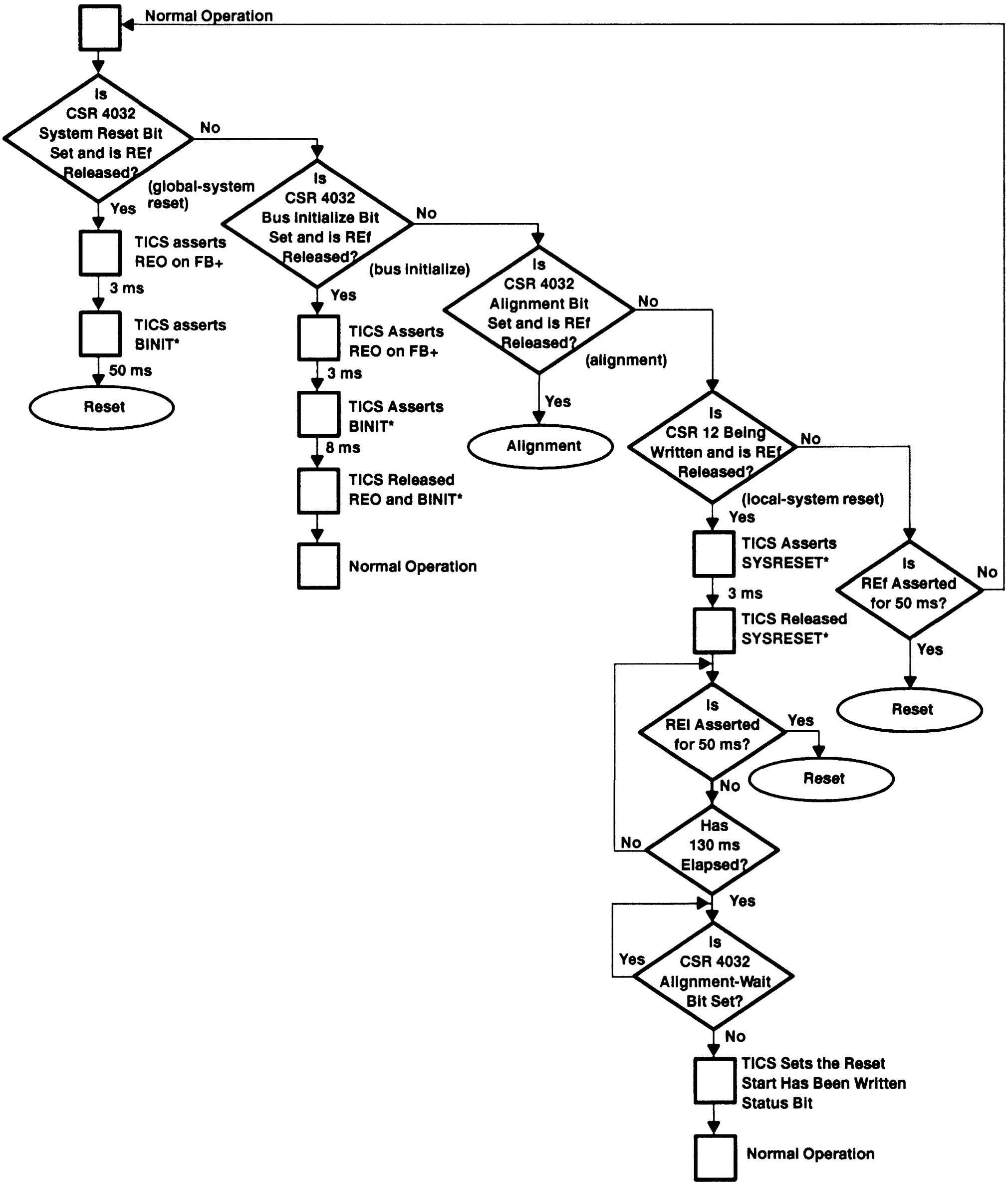
For board initialization. This time can be extended to attain more board initialization time by setting the alignment-wait bit to a logic high. The TICS waits until the bit is released.

Figure 3–4. Live Insertion and Power Up Protocol Flows

3.6.4 ALIGNMENT PROTOCOL (0 <RE* LOW < 3 ms)

When REI is asserted, the TI chip set prepares for alignment by completing any FB+ transactions and waiting for any further host transactions requiring the TI chip set until REI is released. Host activity can always continue as long as the Futurebus+ interface is not required. At the release of REI, the module's alignment has occurred CSR status bit, CSR address 4040, is set.

Note from Figure 3–5 that alignment can be TI chip-set originated by setting the alignment-reset control-configuration bit, CSR address 4032. This bit can be written from the host or FB+ side of the interface. When written, the TI chip set monitors the REI line on FB+. If REI is asserted for greater than the global system reset time (Section 3.6.7), this request is ignored and global system reset is performed. If REI is not asserted, the TI chip set asserts REO on FB+ and waits for FB+ to be idle for at least 1 μs (1.5 μs) and then asserts BUSI* to the board to signal FB+ is at idle. The IOC asserts AIO to FB+ when bus idle occurs. The TI chip set releases REO after 2 reference clocks. The arbiter asserts ARO to FB+ after detecting a BUSI*. If REO is not asserted by the local IOC, the arbiter simply disables its idle arbitration counting mechanism. BINIT* cannot be generated following RST* until either global-system reset or alignment is performed. Host activity may continue as long as the FB+ interface is not required to complete the operation.



Normal Operation
Is CSR 4032 System Reset Bit Set and Is REf Released?
No
Yes
(global-system reset)
TICS asserts REO on FB+
3 ms
TICS asserts BINIT*
50 ms
Reset
Is CSR 4032 Bus Initialize Bit Set and Is REf Released?
No
Yes
(bus initialize)
TICS Asserts REO on FB+
3 ms
TICS Asserts BINIT*
8 ms
TICS Released REO and BINIT*
Normal Operation
Is CSR 4032 Alignment Bit Set and Is REf Released?
No
Yes
(alignment)
Alignment
Is CSR 12 Being Written and Is REf Released?
No
Yes
(local-system reset)
TICS Asserts SYSRESET*
3 ms
TICS Released SYSRESET*
Is REf Asserted for 50 ms?
No
Yes
Reset
Is REI Asserted for 50 ms?
Yes
Reset
No
Has 130 ms Elapsed?
No
Yes
Is CSR 4032 Alignment-Wait Bit Set?
Yes
No
TICS Sets the Reset Start Has Been Written Status Bit
Normal Operation

Figure 3–5. TICS-Originated Reset Protocols (Global-System Reset, Local-System Reset, Bus Initialize and Alignment)

3.6.5 AUTO-ALIGNMENT BYPASS

Auto-alignment bypass is available with the TI chip set, but this feature violates the FB+ reset protocol in IEEE 896.1–1991. This function is not intended for normal operation, but for optional test purposes. This pin should be tied high during normal operation. A module can assert AI* to FB+ and start participating in FB+ transactions after being live inserted into a system by applying a low-high-low pulse to the RSTBYPASS* pin on the IOC. This action does not stop the built-in reset protocol controller from monitoring the REI line on FB+ and reacting accordingly.

3.6.6 BUS-INITIALIZATION PROTOCOL (3 ms < RE* LOW < 50 ms)

After REI is asserted for 3 ms, BINIT* is asserted and remains asserted until REI is released (Figure 3–5). This sequence resets all FB+ controllers to the idle state. Host activity may continue as long as the FB+ interface is not accessed.

The TI chip set and FB+-originated bus-initialization protocols are shown in Figure 3–5 and 3–6 respectively. Note from Figure 3–5 that bus initialization can be TI chip-set originated by setting the bus-initialize reset-control configuration bit, CSR address 4032. This bit can be written from the host or FB+ side of the interface. When written, the TI chip set monitors the REI line on FB+. If REI is asserted for greater than the global-system time(Section 3.6.7), this request is ignored and global-system reset is performed. If REI is not asserted, the TI chip set performs the bus initialization flow as shown in Figure 3–5.

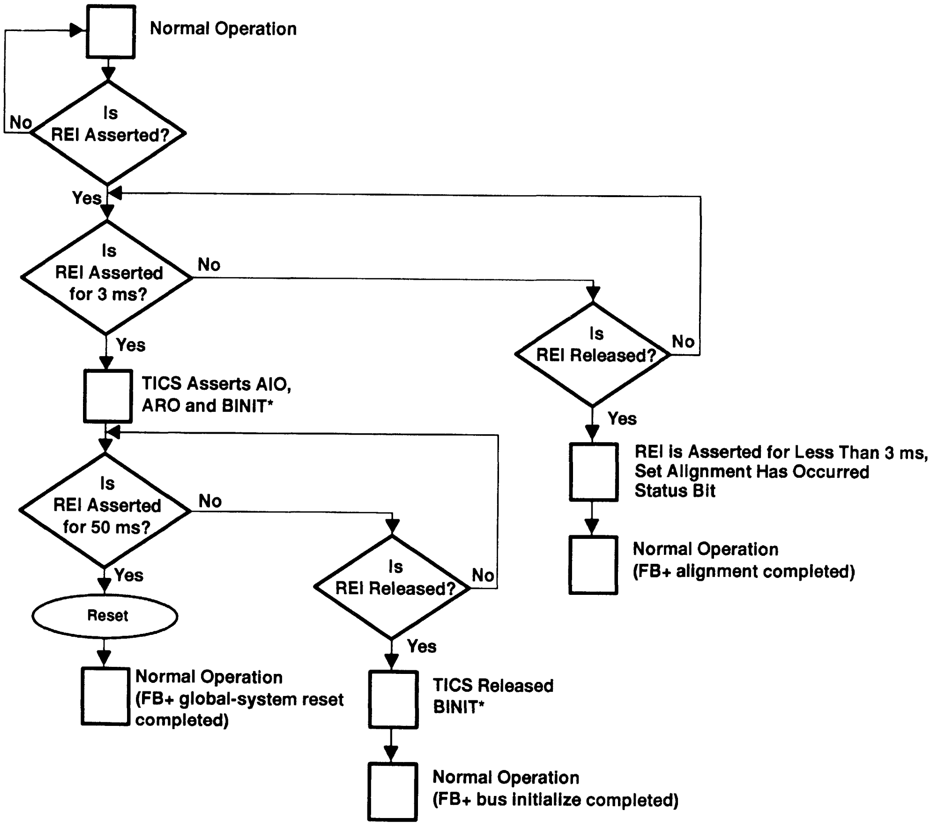

Normal Operation
No
Is REI Asserted?
Yes
Is REI Asserted for 3 ms?
No
Yes
Is REI Released?
No
Yes
TICS Asserts AIO, ARO and BINIT*
REI is Asserted for Less Than 3 ms, Set Alignment Has Occurred Status Bit
Is REI Asserted for 50 ms?
No
Yes
Normal Operation (FB+ alignment completed)
Is REI Released?
No
Reset
Yes
Normal Operation (FB+ global-system reset completed)
TICS Released BINIT*
Normal Operation (FB+ bus initialize completed)

Figure 3–6. FB+ Originated Reset Protocols
(FB+ Global-System Reset, Alignment, and Bus Initialize)

3.6.7 GLOBAL-SYSTEM RESET PROTOCOL (RE* LOW > 50 ms)

After RE* is asserted for 50 ms, BINIT* continues to be asserted while SYSRESET* is asserted. SYSRESET* remains asserted for 3 ms. BINIT* is released 1 reference-clock period after SYSRESET* is released. All controllers are reset and all host operations are reset until SYSRESET* is released.

A global-system reset can be TI chip-set originated by setting the system-reset control-configuration bit, CSR address 4032. Similar to other configuration bits, this bit can be written from the host or FB+ side of the interface. When written, the TI chip set monitors the REI on FB+. If REI is asserted for more than 50 ms, a global-system reset initiated by FB+ is performed. The request to the TI chip set to initiate a global-system reset by setting the system-reset configuration bit is ignored. If REI is not asserted, the TI chip set asserts REO on FB+ and performs the global system reset flow as shown in Figure 3–5.

3.6.8 LOCAL SYSTEM RESET PROTOCOL

The local-system reset-protocol flow is shown in Figure 3–5. A local-system reset can be initiated by writing the reset start CSR, CSR address 12. Once this address is written, the TI chip set monitors the REI line on FB+. If REI is asserted for longer than 50 ms, this request is ignored and global-system reset is performed. Even when this request is in progress and the REI is asserted for greater than the global-system reset time (50 ms), this request is interrupted and a global-system reset is performed. The TI chip set remains aligned to the FB+ during this operation. At the completion of a local-system reset, the status bit in CSR address 4040 is set with the interrupt (if enabled) to indicate to the module that a local-system reset has occurred.

3.7 Power Down

The FB+ standard has defined a power-fail mechanism that can be used by the system to detect and report power fail to all modules. This allows the modules to potentially store critical data before actually losing power. In order for any action to be taken, there must be sufficient reserve power (an uninterruptible power supply)to complete backup operations.

The TI chip set supports this safe-shutdown feature by providing the ability to detect or send the power-fail message. Power fail is detected by the ABC and is indicated on the PFAIL* signal. Power fail is determined to be any arbitrated message 0x60 through 0x7F if the profile B configuration bit is set. Otherwise, only 0x7F is interpreted as power fail. If the module is acting as the reporting agent for the power supply, power fail may be generated by writing 0x7F to the send arbitration message CSR in the ABC.

Chapter 4

HIF and CSR Bus Interface Description

4 HIF AND CSR BUS INTERFACE DESCRIPTION

4.1 Introduction

The host interface (HIF) is used to interface between the TI Futurebus+ chip set and the host module. The host module may contain processor(s), memory, control and status registers (CSRs), and any user-defined I/O units desired by the module designer. Information received from Futurebus+ intended for the host module is transferred to the host interface via the TI Futurebus+ chip set. The host interface is also used by the TI chip set to pass CSR information arriving from Futurebus+ via the data-path unit to the other devices in the chip set and through the CSR bus. (Refer to Figure 1–3)

The host interface incorporates many of the protocols and control signals found in a variety of processors. The host interface operates with a synchronous handshake using either burst or single transfers. It can be designed to support the needs of real-time and fault-tolerant applications. These features allow the interface to be used with current and next-generation processors and components from a variety of manufacturers.

4.2 Host Interface Design Goals

The basic objective of the host interface is to provide scalability and versatility to meet the needs of a variety of systems. In order to satisfy this requirement, the following objectives must be met:

- Provide a generic device interface that communicates easily to many processors, I/O devices, and buses.
- Provide scalable data and addressing. All combinations of 32- or 36-bit addressing and 32- or 64-bit data are supported.
- Provide synchronous burst transfers for both fast and slow devices. Fast devices operate with no wait states. Slow devices are those that require a single wait state between each datum (half speed).
- Provide a synchronous single-transfer mechanism.
- Support both single and multimaster host modules.

The host interface was jointly developed with Force Computers, Inc.

4.3 Host Interface Capabilities

4.3.1 TRANSACTIONS

The following transactions are supported by the TI FB+ chip set:

- Single-transfer read
- Single-transfer write
- Single-transfer read partial
- Single-transfer write partial
- Burst-transfer read
- Burst-transfer write

Other transactions may be used on the host interface as long as the TI chip set is informed not to participate (via IGNORE* signal) when a cache or local agent is responsible for the transaction.

4.3.2 BURST-TRANSFER LENGTH

Burst transfers allow the host to transfer 8, 16, 32, or 64 bytes within a single host transaction. The burst-transfer transaction consists of a host-interface address phase followed by a burst-data phase. The burst data phase can be high speed or low speed. High-speed burst transfers transmit data on each rising clock edge. Low-speed burst transfers transmit data on every other rising clock edge when BSTRDY* is asserted. The width and speed of the burst are respectively selected by the slave using DSACK1* and DSACK0*.

4.3.3 LOGICAL TRANSFER WIDTHS

Logical transfer widths of 8, 16, 24, 32 or 64 bits are supported on the host interface. Single transfers can be 8, 16, 24, or 32 bits wide. TSIZE<1:0> is used by the master to indicate a desired transfer width. DW64* must be released (high). Burst transfers can be 32 or 64 bits wide. The 64-bit transfer width is selected by asserting DW64*.

4.3.4 WAIT STATES

There are three types of wait states: master wait states, slave-address wait states, and slave-burst wait states.

A master requests the host interface, supplies the address information, and then may maintain HDS* released (high) until data is ready to be transmitted. This is used by the TI Futurebus+ chip set when the Futurebus+ is writing to the host module.

A slave can insert wait states in the address phase prior to the beginning of the data phase by not responding with DSACK1* and DSACK0* and BSTRDY* until it is ready to begin the data phase. A slave can also insert wait states during slow-speed burst transfers by releasing BSTRDY* (high) during each data cycle until data is ready. These wait states can be cyclical following a particular pattern to indicate a lower-speed resource or waits can be inserted intermittently to indicate availability of the slave's resources. Slave-burst wait states are intended to be used for devices requiring burst transfers of half speed or less (one transfer every other cycle).

4.3.5 BACKOFFS AND ERRORS

The host interface master can be backed off by the slave prior to the beginning of the host data phase. Backoff is used by a slave to indicate that the master must stop the present transaction and release host mastership in order to ensure forward progress of the system. The master is expected to complete the transaction later.

The slave can indicate an error condition at any time prior to completion of the transaction. These conditions are indicated by the slave using BSTAT1*, 0*.

4.3.6 ADDRESSING

The host interface fully supports 32- and 36-bit addresses. Upon power up, 32-bit addressing is the default. The TFB2022 configuration register can enable the 36-bit addressing. The most significant 4 bits of address occupy HAP<3:0>.

4.3.7 HIF ARBITRATION

The host interface uses a three-wire handshake to arbitrate for mastership. Several master-capable devices may coexist on the host interface. Each master is required to handshake with the host arbiter to determine its tenure. Host arbiters can be designed to allow locking, priority, and fairness capabilities.

4.3.8 COMMUNICATION HIERARCHY OF THE CHIP SET INTERFACE

Figure 4–1 shows the overall communication hierarchy of the interface chip set.

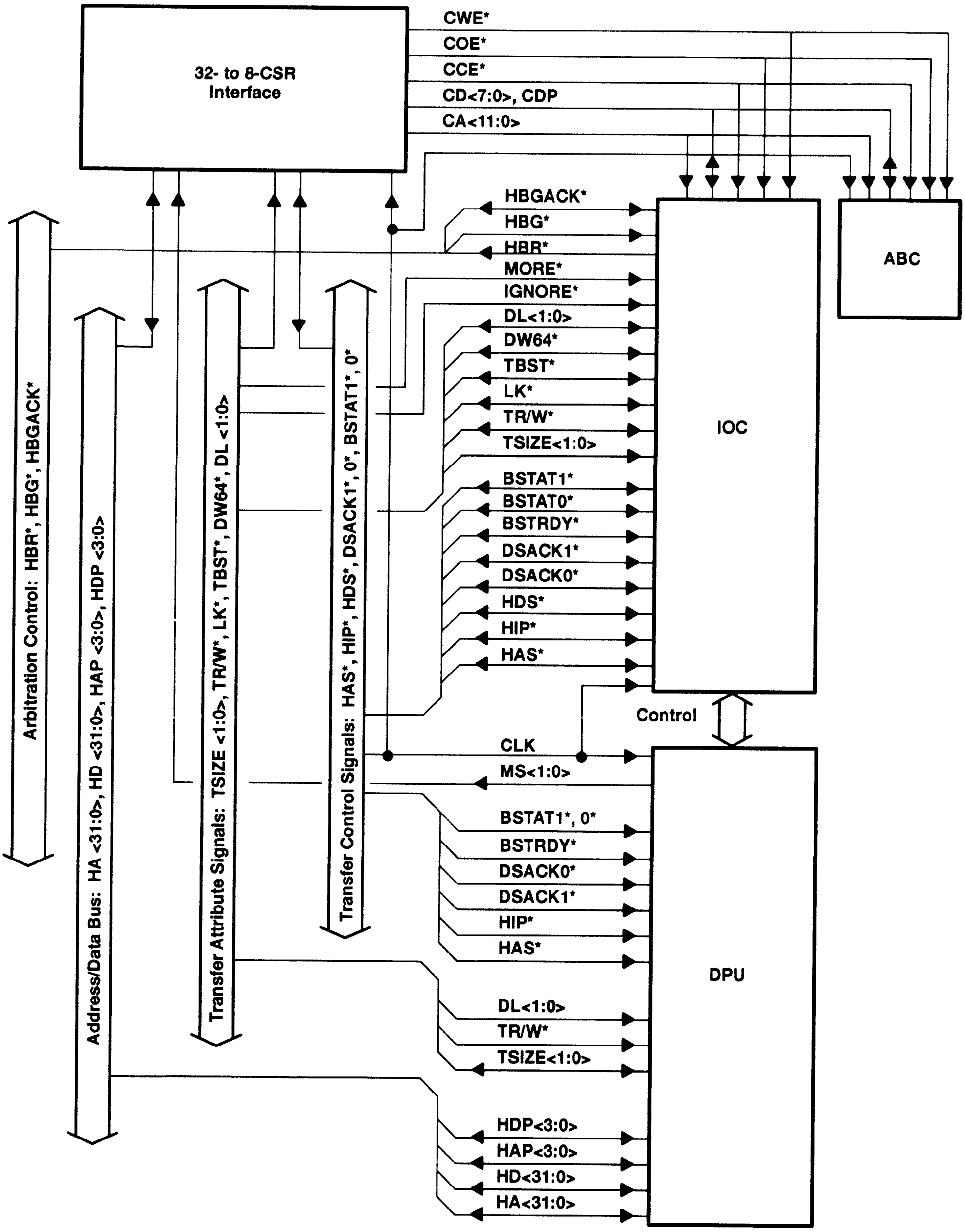


Figure 4–1. Overall Communication Hierarchy of the Chip-Set Interface

4.4 Host Interface Signals

Host Interface Clock

NAME	TYPE	DESCRIPTION
CLK	I	Host interface clock. CLK is used to reference external timing. All timing, unless otherwise noted, is based with respect to the rising edge of CLK.

Address/Data Bus (direction is with respect to bus master, where the master can be either the host or the chip set)

HA<31:0>	I/O	Address and extended data bus. This bus is driven by the master during address phase to indicate the location of the data. For 32-bit-data-width transfers, this bus always indicates the address. For 64-bit-data-width transfers, these lines are used as HD<63:32> during data phase and are driven by the source of the data.
HD<31:0>	I/O	Data bus. This is always used as the data bus during data phase and driven by the source of the data.
HAP<3:0>	I/O	Extended-address and odd-data parity lines. These signals are used as the four most significant address lines for 36-bit-address width. In this mode, HAP<3> is equal to HA<35>, HAP<2> equals HA<34>, etc. Also in 64-bit data width, this bus is used as odd-data parity corresponding to the data on HA<31:0> during data phase.
HDP<3:0>	I/O	Data-odd parity lines. These lines are used as odd data parity corresponding to data on the HD bus during data phase. HDP<0> is used as an odd-parity bit for HD<7:0>, and so on.

Transfer Control Signals(direction is with respect to bus master, where the master can be either the host or the chip set)

HAS*	O	Address strobe. HAS* is asserted by the bus master for the first clock cycle to indicate that the address on the bus is valid and should be clocked in by the slaves on the rising edge of the current cycle. The address should be valid for the entire time HAS* is asserted. Whenever HAS* is asserted, HIP* should also be asserted; otherwise, a slave might interpret the beginning of a bus transfer.
HIP*	O	Transaction in progress. The bus master drives this signal to indicate a bus cycle has begun and all the transfer signals are valid. HIP* shall remain active until the end of the bus transaction. NOTE: When HIP* is high, all control signals should be disregarded except for LK* and arbitration signals.
HDS*	O	Data strobe. The master of the bus uses this signal to indicate that it is ready for a data transfer. When HDS* is asserted by the master in the case of a read, the master is ready to receive data. Likewise, in the case of a write, HDS* is asserted by the master when it is ready to write the data. A slave cannot acknowledge any data transfer until the master has asserted this signal.
DSACK1*, DSACK0*	I	Data strobe acknowledge. The slave on the bus uses these signals with BSTRDY* as data-cycle acknowledge and/or wait-state indicators during the data phase of a transaction. When DSACK1* and DSACK0* are asserted by the slave during a read, the slave is ready to source data. Likewise, during a write when DSACK1* and DSACK0* are asserted, the slave is ready to receive the data. Single Mode (TBST* = high) DSACK1* DSACK0* Result L L Complete cycle – data bus port 32 L H Reserved H L Reserved H H Insert wait state

Transfer Control Signals(direction is with respect to bus master, where the master can be either the host or the chip set) (continued)

NAME	TYPE	DESCRIPTION
(Continued) DSACK1*, DSACK0*		Burst Mode (TBST* = low) DSACK1* DSACK0* Result L L Low-speed, 32-burst capable L H High-speed, 32-bit burst capable H L Low-speed, 64-burst capable H H High-speed, 64-bit burst capable To convert a burst transfer into a single transfer, a slave asserts DSACK1* and DSACK0* as it would for a non-burst transaction but not BSTRDY*. This causes a master to break up a burst transfer into single 32-bit transfers, each with separate address cycles. A slave may not respond to a 32-bit burst request (DW64* high) from a master with a 64-bit capable response. However, a slave may respond to a 64-bit data-width request (DW64* low) from a master with 32-bit burst-capable response In high-speed burst transfers, no wait states can be added in the transfer once the data phase has started. In low-speed burst transfers, one wait state is automatically added in each data cycle once the data phase has started and additional wait states may be added, by the slave by releasing BSTRDY* for the number of wait states that need to be added. NOTE: DSACK1* and DSACK0* and BSTRDY* shall not be asserted by the slave before HDS* is asserted by the master. Also, DSACK1* and DSACK0* should not be driven without BSTRDY* unless it is to insert wait states in slow-mode burst transfers or to convert burst transfers into single transfers.
BSTRDY*	I	Burst ready. The slave asserts this signal with DSACK1* and DSACK0* to indicate to the bus master it is ready to transfer data. In the case of a single transfer (TBST* high), the slave asserts BSTRDY* and DSACK1* and DSACK0* to complete the transaction. In the case of burst transfer, (TBST* low), the slave asserts BSTRDY* and DSACK1* and DSACK0* to start the data phase. When doing slow-mode transfers, BSTRDY* may be released after the first data cycle to insert wait additional states. In fast-mode burst transfers, wait states are prohibited. To convert a burst transfer into a single transfer, a slave asserts DSACK1* and DSACK0* as it would for a non-burst transaction but not BSTRDY*. This causes a master to break up a burst transfer into single 32-bit transfers, each with separate address cycles. To insert wait states in the first data cycle (fast or slow mode), DSACK1* and DSACK0* and BSTRDY* should be held high until the slave is ready to respond to the first cycle. When inserting wait states in slow-mode burst transfers, DSACK1* and DSACK0* should not change during the wait state.
BSTAT1*, 0*	I	Bus error status. This field is driven by the bus slave to indicate a bus error has occurred or a back-off condition exists. BSTAT1* BSTAT0* Result H H Default, Normal operation H L Reserved L H Bus Error, Parity or transaction error L L Back off, Retry transaction

Transfer Attribute Signals(direction is with respect to bus master, where the master can be either the host or the chip set)

TSIZE<1:0>	O	Transaction size. TSIZE is driven by the master to indicate the width of data to be transferred during non-burst transfers (TBST* high). During burst transfers (TBST* low), TSIZE<1:0> should be driven low by the master and ignored by slaves. The encoding is as follows: TSIZE<1> TSIZE<0> Transfer size L L Word (32 bits or greater) L H Byte (8 bits) H L Half-word (16 bits) H H Three bytes (24 bits) HA<1:0> is used as an offset indicator to determine which byte lanes the data appears on during a non-burst transaction.
TR/W*	O	Read/write* indication. This is driven by the master to indicate direction of the data. When TR/W* is high, a data read is in progress. When low, a data write is in progress.
LK*	O	Lock transaction. LK* is driven by the master to indicate that the current bus cycle is part of an indivisible series of bus transactions (e.g., read-modify-write).

Transfer Attribute Signals(direction is with respect to bus master, where the master can be either the host or the chip set) (continued)

NAME	TYPE	DESCRIPTION
TBST*	O	Transaction burst request. TBST* is asserted by the master to request a burst transfer. When TBST* is asserted, a slave must evaluate DL<1:0> as well as DW64*. NOTE: A slave that does not support burst-data transfers should respond to a transaction as 32-bit compelled (i.e., BSTRDY* high, DSACK1* low, DSACK0* low).
DW64*	O	Data-width 64-bit request. DW64* is asserted by the master during a burst request to indicate a 64-bit-wide burst-data transfer request. In this type of transfer during the data phase, the most significant 32 bits of data appear during data phase on HA<31:0> and the least significant 32 bits on HD<31:0>. NOTE: A slave that does not support 64-bit-burst data transfers may respond to a transaction as 32-bit burst or compelled (e.g., BSTRDY* low, DSACK1* low, and DSACK0* low is 32-bit slow burst).
DL<1:0>	O	Data length. This field is asserted by the master during a burst request to indicate the amount of data to be burst in a burst transfer. The encoding of DL<1:0> is as follows: DL<1> DL<0> Data length L L 64 bytes L H 32 bytes H L 16 bytes H H 8 bytes NOTE: A slave that does not support burst data transfers should respond to a transaction as 32-bit compelled (i.e., BSTRDY* high, DSACK1* low, and DSACK0* low).

Arbitration Control Signals(direction is with respect to chip set)

HBR*†	O	Host bus request. A potential bus master asserts a separate HBR* to request mastership of the bus. HBR* shall remain asserted until a separate bus grant (HBG*) is received. Once the requester has been acknowledged bus mastership, it may release HBR* or continue to assert HBR* pending additional bus cycles.
HBG*	I	Host bus grant. When asserted, the requester may become master of the bus once HBGACK* is released. The requester receives a bus grant in response to its bus request, and HBG* must remain asserted until HBGACK* is asserted after the bus is idle. This enables an arbiter to preselect a module even before the bus has become idle.
HBGACK*†	I/O	Bus grant acknowledge. A module asserts this signal to notify the bus arbiter as well as other modules that the bus is busy. Once HBG* is asserted to a requesting module, that module must monitor HBGACK* until it is released. Once HBGACK* is released, the requesting module that has received HBG* may assert HBGACK* and start its transaction(s). HBGACK* shall remain asserted until the module has released all master driven signals.

Special Signals

INT*†	O	Interrupt (direction is with respect to chip set). When asserted, this signal indicates an unserviced interrupt is pending.
MORE*	I	More data. When asserted, the part links multiple host transactions into a single FB+ transaction using packet-mode data transfers. Assertion of this bit causes the chip set to assert PR* (CM0*) during packet data phase.
IGNORE*	I	Ignore current transaction. This signal is used to inform the chip set that the current transaction is directed at another device, even though the address may overlap with one that the chip set could act on. If this capability is not needed, this pin should be tied high.

† HBR*, HBGACK*, and INT* are open-drain outputs, which require pullups.

4.5 Host-Interface Transaction Operation

4.5.1 MASTER/SLAVE OPERATION

The transfer properties (such as data width, length, and speed) of host-interface transactions are the negotiated result of master and slave interaction. A bus master is responsible for controlling the type of transfer as well as the amount of data that is transferred during the transaction. The bus master has some control over the speed of the transaction, but the speed of the transaction is ultimately controlled by the bus slave. A master is able to request the data width, amount of the data, and whether the data is transferred in a single- or burst-mode of operation. Slaves are able to override the requested transfer width and mode of operation (single or burst). Additionally, slaves determine whether burst transfers are high speed (no wait states) or slow speed (at least one wait state).

4.5.2 SINGLE WRITE TRANSACTION

A bus master initiates a transaction by driving HA<31:0>, HAS*, TR/W*, HDS*, HIP*, DW64*, and TBST* to their proper levels. HAS* is asserted low for one clock cycle during which the slave may latch the address on HA<31:0>. The master also drives TSIZE<1:0> to indicate the transfer width of 32 bits. TBST* is high for the entire transaction, indicating a single transaction. HDS* remains high until the bus master drives HD<31:0> and HDP<3:0> with the proper data and parity, then HDS* is asserted low to indicate to the slave that the data is valid. After a master has asserted HDS*, it must wait for a slave to acknowledge the data cycle. Once the slave has acknowledged the data cycle, as described in the next paragraph, the master releases its control signals and the transaction is considered complete.

Once a slave has evaluated the address and determined that it is the selected unit for the transaction, it must wait for HDS* to be asserted before it is required to take any action. After HDS* is asserted, a slave acknowledges receipt of the data by asserting DSACK1*, DSACK0*, and BSTRDY* to indicate to the bus master that it latches the data on the upcoming rising edge of the clock. If a slave is not ready to accept data, it may hold off the master by not asserting DSACK1*, DSACK0*, and BSTRDY*; this is considered a slave wait state. After DSACK1*, DSACK0*, and BSTRDY* complete the transaction, the slave must release its control signals (DSACK1*, DSACK0*, and BSTRDY*).

Single-Transfer Acknowledgement (TBST* = high)

DSACK1*	DSACK0*	RESULT
L	L	Complete cycle – data bus 32 bits
L	H	Reserved
H	L	Reserved
H	H	Insert wait state

4.5.3 SINGLE READ TRANSACTIOn

A bus master initiates a read transaction by driving HA<31:0>, HAS*, TR/W*, HDS*, HIP*, TSIZE<1:0>, DW64*, and TBST* to their proper levels. HAS* is asserted low for one clock cycle. HDS* remains high until the bus master is ready to accept data on HD<31:0> and HDP<3:0>. Once the master is ready, it drives HDS* and waits for the slave to put data on the bus and acknowledge the transfer via DSACK1*, DSACK0*, and BSTRDY*.

Once a slave has evaluated the address and determined that it is the selected unit for the transaction, it may put the data on the HD<31:0> at any time but waits for HDS* to be asserted before it may acknowledge the transaction. After HDS* is asserted, a slave qualifies the data with DSACK1*, DSACK0*, and BSTRDY* to indicate to the bus master that it must clock in the data at the upcoming rising edge of the clock. After this event, both the master and slave release their respected signals and the transaction is finished.

4.5.4 PARTIAL TRANSACTIONS

A partial transaction is one in which less than 32 bits are transferred. The only difference between a partial transaction and a single 32-bit transaction is the use of TSIZE<1:0> and HA<1:0>. In a single 32-bit transfer, TSIZE<1:0> = LL indicates 32 bits and HA<1:0> = LL (aligned to a 32-bit boundary). A partial transaction requires a master and slave to evaluate TSIZE<1:0> and HA<1:0> to determine which and how many data lines are valid. Refer to the following table.

HOST INTERFACE PARTIAL DESCRIPTIONS

TSIZE	HA<1:0>	HD<31:24>	HD<23:16>	HD<15:8>	HD<7:0>	DESCRIPTION
LH	LL				Valid	8 bit
LH	LH			Valid		8 bit
LH	HL		Valid			8 bit
LH	HH	Valid				8 bit
HL	LL			Valid	Valid	16 bit
HL	LH		Valid	Valid		16 bit
HL	HL	Valid	Valid			16 bit
HH	LL		Valid	Valid	Valid	24 bit
HH	LH	Valid	Valid	Valid		24 bit

NOTE:All unlisted combinations of TSIZE and HA<1:0> are considered illegal transactions.
Blank spaces in this table indicate data that is not meaningful.

4.5.5 BURST TRANSACTIONS

Burst transfers may be 8, 16, 32, or 64 bytes long within a single bus transaction, and may use a data width of 32 bits or 64 bits. Burst transfers are always address aligned on even multiples of their transfer length. TBST* is asserted by the master to indicate the transfer is a burst. TSIZE<1:0> is ignored, DL<1:0> is used to determine the amount of data in the burst transfer. The starting address is applied at the beginning of the transaction; all the other addresses are implied, so both master and slave must have an auto increment function for the address. Even though a master initiates a burst transfer, the slave must choose the speed of the transfer (high or low). Therefore, if a bus master initiates a burst transfer, it must be able to transfer the data in fast mode or have prior knowledge the slave will respond with a slow-mode acknowledge.

During burst transfers, the definition of DSACK1* and DSACK0* is changed from that used during non-burst transfers:

Burst Acknowledgement (TBST* = low)

DSACK1*	DSACK0*	RESULT
L	L	Low-speed, 32-bit burst capable
L	H	High-speed, 32-bit burst capable
H	L	Low-speed, 64-bit burst capable
H	H	High-speed, 64-bit burst capable

4.5.5.1 High- versus Low-Speed Operation

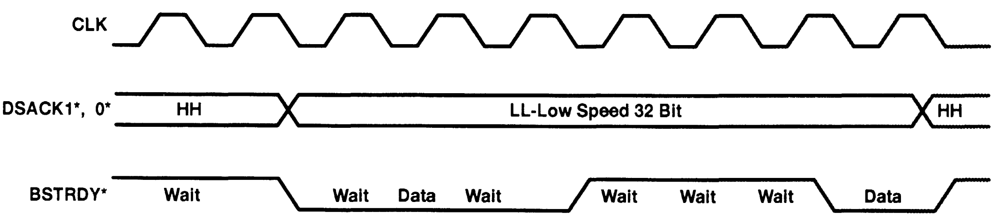
In the fast mode with 32-bit-wide burst transfers, data is transferred at the rate of 4 bytes per clock cycle and extra wait states between the data are not permitted. The master is able to hold off the start of the data by keeping HDS* held high. Also, the slave is able to hold off the start of the data by holding off the acknowledge of data via DSACK1*, DSACK0*, and BSTRDY*. Once the first acknowledge is asserted, the rest of the burst transfer must follow with no extra wait states.

In the slow mode with 32-bit-wide burst transfers, the maximum data rate is 4 bytes every other clock cycle, with an option by the slave to insert extra wait states. The data in slow-mode burst transfers is always valid during the second clock cycle in which BSTRDY* is asserted.

4.5.5.2 Inserting Extra Wait States (in slow-burst mode)

A slave is able to insert extra wait cycles after the first data cycle of a burst transfer. No extra wait states are allowed in the first data cycle because a slave is able to perform the same function by delaying the assertion of BSTRDY*, DSACK1*, and DSACK0*. In order to insert extra wait states, a slave must keep BSTRDY* asserted one clock after the previous data transfer then release BSTRDY* for the desired number of wait cycles. DSACK1* and DSACK0* must be asserted to slow mode. When the slave is ready to transfer data, it may assert BSTRDY* to continue data transfer.

In this limited example, three extra wait cycles are added to a slow 32-bit wide, 8-byte burst transfer. The extra wait cycles simply extend the slow mode sequence [wait, data valid, . . . , wait, data valid] out in time while BSTRDY is released.

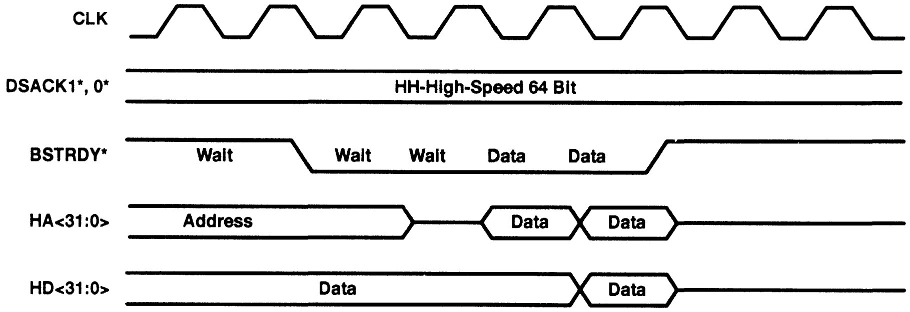


4.5.5.3 32-/64-Bit Data Width

All single transfers use a 32-bit data width, and partial transfers use either 8-bit, 16-bit, or 24-bit data width. Burst transfers have an option, by master and slave agreement, to use either 32-bit or 64-bit data width. A master asserts DW64* and TBST* at the beginning of a bus transaction to request a 64-bit-wide burst transfer. If a master does not assert DW64*, a slave may only respond to the transfer as 32-bit capable (DSACK1*, DSACK0* = LL or LH). If a master asserts DW64*, a slave may respond with either a 32- or 64-bit DSACK1* DSACK0* acknowledge. Therefore, a 64-bit-wide data transfer requires that both the master and slave be capable of this transfer width.

In 32-bit-wide data transfers, all the data is transferred over HD<31:0>. When using 64-bit data transfers, the high-order 32 bits of data are multiplexed onto the HA<31:0> bus. Because of this multiplexing of address/data, two clock cycles are added to the transaction between the address and data phases of a 64-bit transfer. These cycles are used by the master and slave to turn around the drivers on the HA<31:0> bus in the event the transaction is a read. After DSACK1*, 0* and BSTRDY are asserted; the first cycle is used by the master to evaluate DSACK1*and DSACK0* because the slave could be converting the transaction into a 32-bit transfer. The second cycle is used by the master to tri-state the HA<31:0> bus. On the third cycle of a 64-bit wide transfer, the data is valid.

The first three cycles of a 64-bit fast mode and 64-bit slow mode look exactly alike. After the data transfer starts in fast-mode burst transactions, data is transferred at the rate of 8 bytes per clock cycle and extra wait states between the data are not permitted. In slow-mode burst transfers, data is transferred at the rate of at most 8 bytes every other clock cycle, with an option by the slave to insert extra wait states. The following figure shows an example of a 16-byte transfer using a 64-bit-wide host interface.



When a slave responds to a 64-bit data transfer request by the master with a 32-bit DSACK1* and DSACK0* response, the transaction continues as a 32-bit transaction. All the data is transferred on HD<31:0> and HA<31:0> is used for the address.

4.5.5.4 Converting Burst into Single Transfers

One possible dilemma that could arise is that a master requests a burst transfer, but a slave is not able to perform such a transfer. In this case, a slave drives DSACK1* and DSACK0* to LL (32-bit capable) without asserting BSTRDY*. This indicates to the master that the slave is not able to accept burst transfers. The timing for this converted transaction is very close to that of single transfers with the exception that data is considered valid on the second clock cycle rather than the first, after DSACK1* and DSACK0* are asserted.

As in a single transfer, the master and slave must disconnect after the data is transferred by releasing their respective control signals. In order for the bus master to hold the bus, it must keep HBGACK* asserted, then the master must reconnect as many times as needed until the transfer is complete. When a master reconnects, the address is required to be incremented by 4 bytes; however, master may still assert TBST* and DL<1:0> as it would if it was to request a complete burst transfer.

A slave is required to respond to the subsequent burst requests the same way it did to the first, without asserting BSTRDY*. This sequence of events continues until the original burst-size request is complete. If a master requests a 64-byte burst transfer and the slave cannot perform such a transfer, the master and slave would convert the burst into 16 single-mode 4-byte transfers.

4.5.5.5 Critical-Word First

Critical-word first (CWF) mode is the default mode of operation for the chip set when doing burst-read transfers as the host interface slave. CWF is disabled when the chip set is a host-interface master, doing MORE* transactions as an HIF slave, or when the DMA MODE pin is high on the DPU. The words that follow the critical word are incremented in order until the upper data-length boundary. Then the remaining data is transferred starting from the lower data-length boundary, and the address is incremented until completion. Burst transfers are always address aligned on even multiples of their transfer length. CWF provides a method of getting the needed word first, but ultimately the entire aligned block is transferred. For example, a 16-byte transfer with the 16-byte block between 00001000 to 0000100F with critical word address 00001008 starts at 00001008, continues to the end of the block boundary, and follows with the remaining data. The actual address sequence of the 4-byte words would be -1008, -100C, -1000, and -1004.

For 8-byte burst transfers, HA<2> must be evaluated to determine the start location of the data. Likewise for 16-byte burst transfers, HA<3:2> must be evaluated. For 32-byte burst transfers, HA<4:2> must be evaluated, and for 64-byte burst transfers, HA<5:2> must be evaluated. A similar situation exists for 64-bit-wide data; however, the address is incremented by 8 each time.

If a module never needs CWF and wants to disable this feature, the board designer can disconnect the TFB2002 DMAMODE output from the TFB2022 input and tie the TFB2022 DMAMODE input high.

4.5.6 EXTENDED-BLOCK TRANSFER OPERATION

The MORE* input pin on the TFB2002 (IOC) is used to link multiple host transactions into a single Futurebus+ transaction when a transfer size of greater than 64 bytes is desired on FB+. Similar to a LOCK* operation, the MORE* operation eliminates the need for multiple FB+ arbitration requests. Unlike the LOCK* operation, however, this mode also eliminates the need for multiple connection and disconnection phases on Futurebus+. The data phase in this FB+ operation is extended by transferring multiple data packets in one FB+ tenure by the chip set automatically asserting the PR* bit (CM0) during packet data phase. When this pin is asserted low during the host address phase, the chip set continues FB+ data handshakes instead of proceeding to the disconnection phase. When utilizing this feature, the user must ensure that the block transfer does not cross a memory boundary and that packet-mode data transfers are being used on Futurebus+. The IOC disables critical-word first operation during a FB+ read when MORE* is active.

To enable extended block transfers, the MORE* pin must be asserted when HAS* is asserted. The MORE* pin stays low until the last transaction. Before starting the final host transaction, MORE* is driven high before HIP* is driven low. This allows the TFB2002/22 to end the FB+ tenure after the extended block transfer is complete. Even though this causes only one address phase on FB+, an address phase for each transaction is needed on the host interface. The TFB2022 (DPU) assumes all addresses after the first on the host interface are contiguous and subsequent addresses will be ignored unless the part is pre-empted on Futurebus+.

In the case of a pre-emption on Futurebus+ during an extended-block transfer operation, the TFB2002 disconnects from FB+. The software is responsible for resuming the transaction from the appropriate point.

4.5.7 IGNORE OPERATION

The IGNORE* input pin on the TFB2002 (IOC) is used to inform the TFB2002/22 that the the present transaction on the host interface should be ignored. This function is useful when two or more addresses are to the same location (i.e., private space and CSR-defined space). When the IGNORE* pin is asserted low during the entire host transaction, the TFB2002 does not respond. When this function is not desired on a board, the IGNORE* pin should be tied high. The module is responsible for ignoring the MS<1:0> field.

4.5.8 RETRY AND ERROR CONDITIONS

There are two conditions that can cause a transaction to be suspended; these are a retry condition and an error condition. The retry requires a bus master to disconnect from the bus and rearbitrate at a later time (back-off retry). The error requires the bus master to stop the transaction. Additional hardware or software must be used to recover from an error condition.

4.5.8.1 Back-Off Retry

Back-off retry occurs when a conflict for the bus arises. Examples of such conflicts are: a HIF transaction is starting and the slave is still performing another transaction, a slave does not yet have the data and it takes several clock cycles to retrieve the data, or the chip set is being accessed as a FB+ slave and host interface slave at the same time. In all cases, the master of the current transaction must disconnect after detecting BSTAT1* and BSTAT0* = LL. It must then rearbitrate for the bus and retry the transaction at a later time. Errors could arise if a back off is not handled properly.

4.5.8.2 Bus Error

This condition is a major bus error. Events that cause this condition are protocol and parity errors. When this condition arises, the master must disconnect from the bus but does not reconnect. Some type of error handler is required to correct the bus error and reset or replace the unit that caused the error. This condition is indicated by BSTAT1* and BSTAT0* = LH.

4.5.9 HOST-INTERFACE ARBITER

The host interface uses three signals (HBR*, HBG*, and HBGACK*) to arbitrate for mastership of the host interface. The HIF arbiter is external to the chip set. HBR* is used to request host mastership from the host arbiter. HBG* is used to indicate to the requesting device that it will be the next host master. Each master capable device on the host interface has unique HBR* and HBG* signals connected between it and the host arbiter. HBGACK* is a wire-ORed signal used to indicate actual host bus mastership. HBGACK* provides a handshake to allow the host arbiter to pre-elect the next master on the host interface before the present master has completed its host tenure. The host arbiter does not necessarily need to receive HBGACK* unless some masters do not implement this feature. The sequence of events for host arbitration is as follows:

1. Any host master-capable device requests host-interface tenure by asserting HBR* (low).
2. The host arbiter asserts HBG* (low) to a single requesting device (referred to as the selected master).
3. The selected master sees HBGACK* released (high) from the previous selected master who has completed its tenure.
4. The selected master begins its tenure by asserting HBGACK* (low).
5. The selected master may release HBR* (high) at any time following the assertion of HBGACK*, allowing the host arbiter to select the next master.

NOTE: If a host device has only the HBR* and HBG* signals, then the host arbiter must observe HBGACK* from the chip set and assert HBG* to the new host master only after HBGACK* is released. If the TI chip set is the only master on the host interface, HBR* and HBG* can be tied together with a pullup.

4.6 Typical Hookups

4.6.1 CASE 1

If the module has 32- or 64-bit data (no parity), 36-bit addressing, fixed high-speed burst data length of 64 bytes, partial transaction support, burst and single transfer, and is a FB+ slave or master, the following connections can be made:

HA<31:0>	Connected to module address bus
HD<31:0>	Connected to module data bus
HAP<3:0>	Connected to 4 most significant bits of address
HDP<3:0>	Not used (pulled high)†
TSIZE<1:0>	Masters wishing to perform partial transfers must drive to the appropriate value.
TR/W*	This 3-state signal is connected to the read/write control pin on each device.
LK*	This 3-state signal is connected to each master capable device that sources locked operations and pulled high.
TBST*	This 3-state signal is connected to each master capable device.
DW64*	This signal may be pulled high to cause a default transfer width of 32 bits. This allows masters performing 64-bit wide transfers to assert it (drive low). If the default case is 64 bits, this signal can be pulled low.
DL<1:0>	These signals are pulled low to indicate fixed 64-byte transfers.
HAS*	Connected to address strobe and pulled high
HDS*	Connected to data strobe and pulled high
HIP*	Connected to transaction in progress indicator and pulled high
DSACK1*	DSACK1* is sourced by the module's control logic indicating transfer capability and pulled high.
DSACK0*	DSACK0 is sourced by the module's control logic indicating transfer capability and pulled high.
BSTRDY*	Connected to module BSTRDY* control signal.
BSTAT1*	Connected to pullup resistor
BSTAT0*	Connected to pullup resistor
HBR*	Connected to host-interface arbiter and pulled high
HBG*	Connected to host-interface arbiter
HBGACK*	Connected to host-interface arbiter and pulled high
RST*	Generated by the module on power up of the module
INT*	Connected to module-interrupt-handling logic and pulled high
CLK	Connected to the desired host clock (this clock determines host speed).
REFCLK	Connected to the desired packet-mode clock (1/2 FB+ packet-mode speed: 25–40 MHz).

† These signals are sourced by the TI Futurebus+ chip set; therefore, they are not used by the slave-only devices on this module. It is recommended that the module pull these signals high to eliminate noise sources.

4.6.2 CASE 2

If the module has 32-bit data (no parity), 32-bit address, fixed high-speed burst data length of 64 bytes, no partials, burst only with no transaction error reporting or master backoff, and has both master and slave capabilities, the following connections can be made:

HA<31:0>	Connected to module address bus
HD<31:0>	Connected to module data bus
HAP<3:0>	Not used (pulled high)†
HDP<3:0>	Not used (pulled high)†
TSIZE<1:0>	Pulled low
TR/W*	Connected to module read/write-control signal
LK*	Connected to module lock-control signal and pulled high
TBST*	Pulled low
DW64*	Pulled high
DL<1:0>	DL1 and DL0 pulled low
HAS*	Connected to HAS* control signal and pulled high
HDS*	Connected to HIP* and data-strobe control signal and pulled high
HIP*	Connected to transaction in progress indicator and pulled high
DSACK1*	DSACK1* can be sourced by the module's BSTRDY* control signal (connected through a 3-state buffer to prevent contention) and pulled high.
DSACK0*	Pulled high
BSTRDY*	Connected to module BSTRDY* control signal and pulled high.
BSTAT1*, 0*	Pulled high
HBR*	Connected to host-interface arbiter and pulled high
HBG*	Connected to host-interface arbiter
HBGACK*	Connected to host-interface arbiter and pulled high
RST*	Generated by the module on power up of the module
INT*	Connected to module-interrupt-handling logic and pulled high
CLK	Connected to the desired host clock (this clock determines host burst speed).
REFCLK	Connected to the desired packet-mode clock (1/2 FB+ packet-mode speed: 25–40 MHz).

† These signals are sourced by the TI Futurebus+ chip set; therefore, they are not used by the slave-only devices on this module. It is recommended that the module pull these signals high to eliminate noise sources.

4.6.3 CASE 3

If the module has 32-bit data (no parity), 32-bit address, fixed high-speed burst data length of 64 bytes, no partials, burst only with no dynamic error reporting or master backoff, and is a FB+ slave only module, then the following connections can be made:

HA<31:0>	Connected to module address bus
HD<31:0>	Connected to module data bus
HAP<3:0>	Not used (pulled high)†
HDP<3:0>	Not used (pulled high)†
TSIZE<1:0>	Pulled down
TR/W*	Connected to module read/write-control signal and pulled high
LK*	Connected to module-lock-control signal
TBST*	Pulled low†
DW64*	Pulled high†
DL<1:0>	Pulled low†
HAS*	Pulled high
HDS*	Pulled high
HIP*	Pulled high
DSACK1*	Can be connected to module's BSTRDY*-control signal and pulled high
DSACK0*	Pulled high
BSTRDY*	Connected to module BSTRDY*-control signal and pulled high
BSTAT1*, 0*	Pulled high
HBR*	Connected to BG* and pulled high
HBG*	Connected to to BR*
HBGACK*	Pulled high
RST*	Generated by the module on power up of the module
INT*	Unconnected since there is no local interrupt-handling logic
CLK	Connected to the desired host clock (this clock determines host speed).
REFCLK	Connected to the desired packet-mode clock (1/2 FB+ packet-mode speed: 25–40 MHz).

† These signals are sourced by the TI Futurebus+ chip set; therefore, they are unused by the slave-only devices on this module. It is recommended that the module pull these signals high to eliminate noise sources.

4.7 CSR Bus Interface Specification

4.7.1 INTRODUCTION

The CSR bus (see Figure 1–3) is used to interface to host ID and capability ROM (see IEEE Std 896.2 for additional information), user-defined units, and the unit architecture for the ABC and PCBA. This bus is separated from the host in order to reduce the loading of the host by noncritical devices. It is an 8-bit-wide bus with control signals similar to many SRAMs and I/O devices (i.e., CE*, OE*, WE*, CA<11:0>, CD<7:0>). This simple 8-bit bus allows the Futurebus+-required CSR ROM to be implemented as a single PROM or SRAM. The CSR bus interface can be a simple byte unstacker and address decoder (see Appendix A). The capability to lock transactions is available; however, this is optional and is left to the system designer. Multiple master capability may be implemented.

The CSR bus interface is meant to satisfy the following primary objectives:

- Provide a single load to the host for all performance-noncritical host devices
- Provide a simple interface for PROM, SRAM, configuration of the controller, and user-defined I/O units
- Provide for read, write, and locked transactions on the CSR bus
- Provide multiple master capability in order to support I/O units with built-in DMA

4.7.2 CSR BUS ARBITRATION

CSR bus arbitration is identical to host-interface arbitration; see sections 4.3 to 4.5 for details.

4.7.3 CSR BUS INTERFACE SIGNAL LIST

CSR BUS-INTERFACE SIGNAL DEFINITION		
NAME	**TYPE**	**DESCRIPTION**
Address and Data Signals (direction specified with respect to master)		
CA<11:0>	Bidirectional	CSR bus address
CD<7:0>	Bidirectional	CSR bus data
CDP	Bidirectional	CSR bus data parity
Transfer Attribute Signals (direction specified with respect to master)		
LK* (optional)	3-state output	Indivisible cycle indication (LOCK)
Transfer Control Signals (direction specified with respect to slave)		
CCE*	Input	Chip enable
COE*	Input	Output enable
CWE*	Input	Write enable
Arbitration Signals (only used on a master)		
HBR* (optional)	3-state output	Bus request
HBG* (optional)	Input	Bus grant
HBGACK* (optional)	Open-collector bidirectional	Bus grant acknowledge
Other Signals (direction specified with respect to slave)		
RESET*	Input	Host module reset signal
INT	Output	Interrupt
CLK	Input	CSR bus reference clock

CA<11:0> – Bidirectional address lines driven by the CSR master

CD<7:0> – Bidirectional data lines driven by the master on writes or driven by the slave on reads

CDP – Bidirectional data parity bit. Used to indicate odd parity.

LK* – Locked transaction. This signal is asserted by the master to indicate that the current bus cycle is part of an indivisible series of bus transactions. It remains asserted during all bus cycles that need to be locked.

CCE* – Chip enable indicates that a particular slave chip has been selected. This signal is the result of decoding a 32-bit host address. Other masters on the CSR bus provide a CCE* to the CSR- to host-interface device and provide the desired 32-bit address for the host transaction. The transaction is then repeated on the host, sending it to the desired location.

COE* – Output enable indicates that a read transaction is in progress, and the addressed slave location should now provide the data. The slave drives the bus when COE* is low.

CWE* – Write enable indicates that a write transaction is in progress. At the rising edge of CWE*, the addressed slave location should receive data.

HBR* – Bus request (same as on host)

HBG* – Bus grant (same as on host)

HBGACK* – Bus grant acknowledge (same as on host)

RESET– System reset. This signal is an asynchronous reset used to initialize the host module. This signal must be asserted for an adequate number of clock cycles after V_{CC} has reached its specified level to ensure that all devices are reset and CLK is stable.

INT – Interrupt. This signal indicates that an unserviced interrupt is pending.

CLK – System clock. All CSR transactions are referenced to this clock.

4.7.4 CSR BUS TRANSACTION LIST

The CSR bus supports read and write transactions, which are based on SRAM reads and writes. For SRAMs without output enables, the CCE* bit may be used alone. During a write, masters should keep the data bus in the high-impedance state until CWE* is low to avoid bus contention. A series of reads and writes may be locked by using the LK* signal.

4.7.5 CSR BUS INTERFACE TIMING DIAGRAMS

Read and write timing diagrams, which describe CSR bus access, are shown in Figures 4–2 and 4–3.

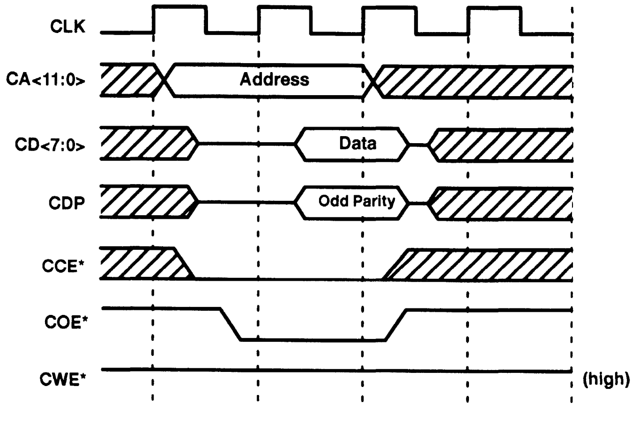


Figure 4–2. CSR Read

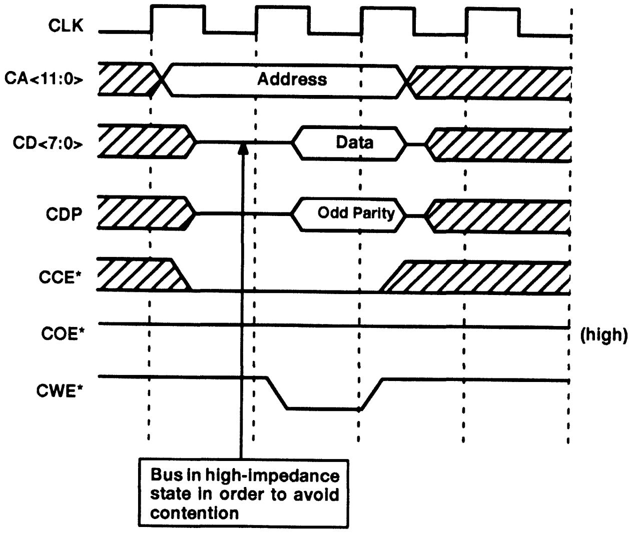


Figure 4–3. CSR Write

Chapter 5

Transactions

5 TRANSACTIONS

This chapter describes how the IOC and DPU interact to service host and FB+ transactions. The interconnecting signals between the IOC and DPU used to support these transactions are described. Local and remote transactions are used to clarify the operation of the TI chip set.

5.1 Intra-Chip-Set Signal Definitions

The IOC controls the transaction handshake and control operations on both Futurebus+ and the host interface. In order to make decisions about the actions it is required to take, it relies on the address decoding, FIFO status, error information, and synchronization handshake provided by the DPU. The DPU handles all data transfer, packet decode, and buffering between Futurebus+ and the host interface. It also captures, decodes, and controls addressing between the two buses. The interface between the two chips can be separated into the five related segments shown in Table 5–1: host interface control and synchronization, FIFO control, Futurebus+ control and synchronization, address control, and intra-chip-set error reporting.

Table 5–1. Intra-Chip-Set Signal Definitions

<table>
<tr><th>SIGNAL NAME</th><th>SOURCE</th><th>HOST INTERFACE CONTROL AND SYNCHRONIZATION</th></tr>
<tr><td>HMODE&lt;2:0&gt;</td><td>IOC</td><td>Host mode. Indicates to the DPU what action is to be taken in the host interface.
<table>
<tr><th>HMODE</th><th>DATA FLOW</th><th>DESCRIPTION</th></tr>
<tr><td>LLL</td><td></td><td>Reset condition</td></tr>
<tr><td>LLH</td><td>FIFO &lt;–&gt; HIF (compelled)</td><td>FB+ compelled read or write to local non-DPU space
HIF compelled read or write from remote space</td></tr>
<tr><td></td><td>HIF &lt;–&gt; DPU CSR</td><td>HIF read or write to DPU internal register</td></tr>
<tr><td></td><td>HIF –&gt; DPU CSR and FIFO</td><td>HIF write to broadcast address</td></tr>
<tr><td></td><td>DPU CSR –&gt; FIFO (internal transaction)</td><td>FB+ read from DPU CSR</td></tr>
<tr><td>LHL</td><td>FIFO –&gt; DPU CSR and HIF</td><td>FB+ write to DPU CSR</td></tr>
<tr><td>LHH</td><td></td><td>Reserved</td></tr>
<tr><td>HLL</td><td></td><td>Reserved</td></tr>
<tr><td>HLH</td><td>FIFO &lt;–&gt; HIF (burst mode)</td><td></td></tr>
<tr><td>HHL</td><td></td><td>Reserved</td></tr>
<tr><td>HHH</td><td></td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>HSTRB*</td><td>IOC</td><td>HMODE&lt;2:0&gt; is valid when this signal is low.</td></tr>
<tr><td>HADEC&lt;3:0&gt;</td><td>DPU</td><td>Host address decode. Address decoding for the host interface address. (see Section 5.2.1)
<table>
<tr><th>HADEC</th><th>HIF SLAVE ENCODING</th><th>HIF MASTER ENCODING</th></tr>
<tr><td>LLLL</td><td>Not in HIF address domain</td><td>Reset condition</td></tr>
<tr><td>LLLH</td><td>Host memory</td><td>Compelled-only memory address</td></tr>
<tr><td>LLHL</td><td>Host extended unit space</td><td>Maximum burst-size capable extended unit or memory</td></tr>
<tr><td>LLHH</td><td>Host CSR</td><td>64-byte burst-capable memory</td></tr>
<tr><td>LHLL</td><td>Broadcast mailbox</td><td>32-byte burst-capable memory</td></tr>
<tr><td>LHLH</td><td>Reserved</td><td>16-byte burst-capable memory</td></tr>
<tr><td>LHHL</td><td>Matches address of a transaction that was split by Futurebus+</td><td>8-byte burst-capable memory</td></tr>
<tr><td>LHHH</td><td>Futurebus+ CSR address</td><td>Reserved</td></tr>
<tr><td>HLLL</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>HLLH</td><td>Broadcast CSR</td><td>Reserved</td></tr>
<tr><td>HLHL</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>HLHH</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>HHLL</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>HHLH</td><td>DPU CSR</td><td>Reserved</td></tr>
<tr><td>HHHL</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>HHHH</td><td>Reserved</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>DMAMODE</td><td>IOC</td><td>When DMAMODE is asserted (high), critical word first data ordering is disabled. In this mode, data is read from Futurebus+ from the address indicated and continued until the data length is reached in a linear address fashion (i.e., no wraparound). When DMAMODE is released (low) and an unaligned address with respect to the data length is supplied from the HIF, FIFO data is wrapped around for critical-word first delivery.</td></tr>
<tr><td>HBMASTER*</td><td>IOC</td><td>This signal is asserted (low) when the IOC has gained mastership of the HIF.</td></tr>
<tr><th>SIGNAL NAME</th><th>SOURCE</th><th>FIFO CONTROL</th></tr>
<tr><td>FIFORST*</td><td>IOC</td><td>FIFORST* causes the FIFO to be reset when it is asserted (low).</td></tr>
<tr><td>SPACEAV*</td><td>DPU</td><td>Space available in FIFO. In the compelled mode, this indicates that space is available in the FIFO for another transfer. In packet or burst mode, this indicates that space is available in the FIFO for another packet or burst.</td></tr>
</table>

Table 5–1. Intra-Chip-Set Signal Definitions (Continued)

SIGNAL NAME	SOURCE	FIFO CONTROL (continued)
DATAAV*	DPU	Data available in FIFO. In the compelled mode, this indicates if any data is in the FIFO. In packet or burst mode, this indicates if a packet or burst data of length DL is available.
SIGNAL NAME	**SOURCE**	**FUTUREBUS+ CONTROL AND SYNCHRONIZATION**
FMODE<2:0>	IOC	Futurebus+ mode. Indicates to the DPU what action is to be taken in the Futurebus+ interface: **FMODE** **DESCRIPTION** LLL Compelled transfer At each transition of FSTRB, perform next beat of transfer. LLH Packet transfer At each transition of FSTRB, perform next packet transfer. LHL Partial transfer for master write LHH Send disconnect data HLL Reserved HLH Reserved HHL Reserved HHH Disconnect data for split requestor Reserved
FADEC<3:0>	DPU	Futurebus+ address decode **FADEC** **FB+ SLAVE ENCODING** LLLL Initial condition LLLH Host memory LLHL Host extended unit space LLHH Host CSR LHLL Broadcast mailbox LHLH Reserved LHHL Reserved LHHH Reserved HLLL Mailbox address HLLH Packet-mode capable memory HLHL Reserved HLHH Reserved HHLL Reserved HHLH DPU CSR HHHL Broadcast CSR (non-DPU) HHHH Broadcast CSR (DPU)
FSTRB	IOC	Futurebus+ strobe. Perform next Futurebus+ event.
FACK	DPU	Futurebus+ acknowledge. Futurebus+ event complete.
FRD*	IOC	Futurebus+ read. L = data moves from FB+ into the FIFO; H = data moves from the FIFO out to FB+.
UNALIGNED*	DPU	UNALIGNED* is asserted (low) when this chip set is the selected slave for a partial transaction on Futurebus+ that requires two partial transactions on the HIF.
SIGNAL NAME	**SOURCE**	**ADDRESS CONTROL**
NEWADDR*	IOC	NEWADDR* increments address in the address register of the DPU.
HBADLD*	IOC	Futurebus+ has been granted for the requested transaction; load the host address for use on FB+.
SELECTED*	IOC	A Futurebus+ transaction uses this module; load the FB+ address for use on the HIF.
SIGNAL NAME	**SOURCE**	**INTRA-CHIP-SET ERROR REPORTING**
ERROR<1:0>	DPU	Futurebus+ data-path error indicators: **ERROR** **DATA-PATH ERROR** LL Initial condition – no error LH Futurebus+ parity error Sets error high CSR (384) byte 1 bit 7 HL Packet longitudinal parity error Sets error lo CSR (388) byte 3 bit 5 HH Host-interface parity error Sets status CSRs (4040 and 4044) byte 3 bit 2
ARBERR <1:0>	ABC	Futurebus+ arbitration bus-error indicators: ARBERR **ARBITRATION BUS ERROR** LL Initial condition – no error LH AC0 and AC1 asserted during phase 3 Sets error lo CSR (388) byte 3 bit 4 HL Arbitration comparison error Sets error lo CSR (388) byte 3 bit 3 HH Arbitration time-out error (phase 2 or 4) Sets error lo CSR (388) byte 3 bit 2

5.1.1 HOST-INTERFACE CONTROL AND SYNCHRONIZATION

Host synchronization is based on two synchronous state machines: one is located in the IOC and one is located in the DPU. The state machine in the IOC controls the host-interface handshake signals as both a master and a slave. When this state machine decides upon the next operation that needs to be performed on the host interface, it generates the corresponding HMODE and then asserts HSTRB* to inform the DPU of the action it is required to take. HADEC and FADEC are used by the IOC to determine the type of action that needs to be taken when the chip set masters an HIF transaction (caused by a FB+ transaction). The state machine in the DPU observes both the HIF handshake signals and the HMODE/HSTRB* signals.

DMAMODE controls the order of burst transfers on the host interface. If DMAMODE is asserted (high), the sequence of the burst transfer is purely sequential. If DMAMODE is released (low), the burst begins at the addressed location and continues until the end of the aligned block of the data length indicated during the burst. The transaction then wraps to the beginning of the aligned block and continues until the original address is reached. This operation is known as critical-word first. The protocol is used by many modern processors and is included in this chip set for simplifying the interface to these processors. If it is not needed, simply disconnect the DMAMODE output from the IOC and tie the input to the DPU high.

The IOC asserts DMAMODE when MORE* has been asserted. It is released for transactions where MORE* is released. Critical-word-first ordering only applies when the TI chip set is the host-interface slave during a read transaction.

HBMASTER* is generated by the IOC to tell the DPU to enable its output transceivers for TSIZE<1:0>, HA<31:0>, and HAP<3:0>. The transceivers for HD<31:0> and HDP<3:0> are enabled as well if the transaction is a write.

5.1.2 FIFO CONTROL

SPACEAV* indicates that the DPU's FIFO has enough space to begin a data transfer. In the packet mode on Futurebus+, it means that a packet can be received without overflowing the FIFO. In the burst mode on the host interface, it means that there is adequate space to receive a burst transfer from the host interface. When the TI chip set is performing a read from the host interface, the host-interface controller on the IOC waits for SPACEAV* to be asserted (low) prior to transferring data from the host interface to the FIFO by asserting HDS* (low). When the TI chip set is performing a read from Futurebus+, the Futurebus+ controller in the IOC waits for SPACEAV* (low) prior to performing the next data handshake (DSO). During a write from the host interface, the host interface-controller waits for SPACEAV* (low) before performing the data-acknowledge handshake. During a write from Futurebus+, the Futurebus+ controller in the IOC waits for SPACEAV* (low) before performing the DKO and DIO handshake for the next data transfer.

DATAAV* indicates that the FIFO contains enough data to perform the requested transfer. This means that a packet is available for transfer in the packet mode on Futurebus+ or a burst of the length indicated on the host interface is available for transfer. During a read from the host interface, the host-interface controller on the IOC waits for DATAAV* to be asserted (low) prior to transferring data from the FIFO to the host interface and performing the host interface data handshake. During a read from Futurebus+, the Futurebus+ controller in the IOC waits for DATAAV* (low) prior to performing the next data handshake (DKO, DIO). When writing to the host interface, the host-interface controller waits for DATAAV* (low) before asserting HDS* for the data transfer. When the TI chip set writes to Futurebus+, the Futurebus+ controller in the IOC waits for DATAAV* (low) before performing the data handshake for the next data transfer.

FIFORST* causes all pointers in the FIFO to be cleared.

5.1.3 FUTUREBUS+ CONTROL AND SYNCHRONIZATION

FSTRB is generated by the IOC to validate the FMODE signals and inform the DPU that the next Futurebus+ operation needs to be performed. Based on the sequence of the ASI and FSTRB, the DPU configures itself to be a master or slave on the Futurebus+. If ASI is released (high) before FSTRB is asserted (low), the DPU is configured to be a Futurebus+ slave and it latches the address.

If FSTRB is asserted (low) before ASI is released (high), the DPU is configured to be a Futurebus+ master and it sources the address. The first FSTRB assertion (low) forces the DPU FB+ state machine to be in the connection phase. After that, the DPU asserts FACK (low) to inform the IOC the acknowledge of the end of the connection phase. During the connection phase, the DPU latches the address and data-width attributes. The first FSTRB* release (high) causes the DPU state machine to be in the data-phase state. At the beginning of the data phase, the DPU latches the data-length attributes. For every FSTRB* action, there is a FACK reaction and FACK follows the logic level of FSTRB except for the following cases:

- If the DPU is a FB+ master and is required to send disconnect data (FMODE = LHH), the DPU toggles FACK at the end of the data phase.
- If the DPU is mastering a FB+ partial transaction, it drives FACK high after it receives a request from the IOC to send byte-lane enables (FMODE = LHL, partial) during connection. Then the IOC asserts DS0 to validate the byte-lane enables, and the DPU drives FACK low.

When the DPU is a FB+ slave, the FB+ address is latched. The DPU generates the correct FADEC based on the address provided. The DPU asserts SELECTED* when FADEC equals one of the following values:

FADEC = LLLH, host memory
FADEC = LLHL, host unit space
FADEC = LLHH, host CSR
FADEC = LHLL, broadcast mailbox
FADEC = LHHL, split-response hit
FADEC = HLLL, mailbox address
FADEC = HLLH, packet-mode capable
FADEC = HHLH, DPU CSR
FADEC = HHHL, HHHH, broadcast CSR

SELECTED* latches the FADEC value for MS lines decoding.

The IOC uses FADEC information to signal an end of transaction (ED), a compelled (FMODE = LLL) transaction, or packet (FMODE = LLH) transaction.

FRD* is generated by the IOC to tell the DPU whether to source data to Futurebus+ (FRD* = H) or to receive data from Futurebus+ (FRD* = L).

UNALIGNED* is asserted by the DPU when it detects a 64-bit partial transaction on Futurebus+ that requires two host transactions to complete. In this case, the DPU masters the HIF and it generates the correct HIF address and TSIZE value.

5.1.4 ADDRESS CONTROL

Located within the DPU is a circuit that generates addresses for HIF and FB+ transactions (see Figure 5–1). This circuit can latch Futurebus+ or host-interface addresses, increment host-interface addresses, and source addresses to FB+ and HIF. The multiplexer selects between the FB+ address, the HIF address, and the stored address plus the data length (in bytes) of the previous burst transfer. The present address latch is used to hold the FB+ or HIF address stable during the address portion of the transfer. The store-address register is used to hold the previous address stable while the present address is being incremented.

When the TI chip set becomes the selected slave on Futurebus+, the IOC asserts SELECTED* (low) and the DPU control logic loads the present address with the FB+ address after it sees the assertion of SELECTED* (low). When the the TI chip set becomes the master on Futurebus+, the IOC asserts HBADLD* (low) and the DPU control logic loads the present address with the HIF address after it sees the assertion of HBADLD* (low).

The DPU loads the present address with the incremented address following the assertion of NEWADDR* (low). The IOC asserts NEWADDR* after it has completed sending a block of data.

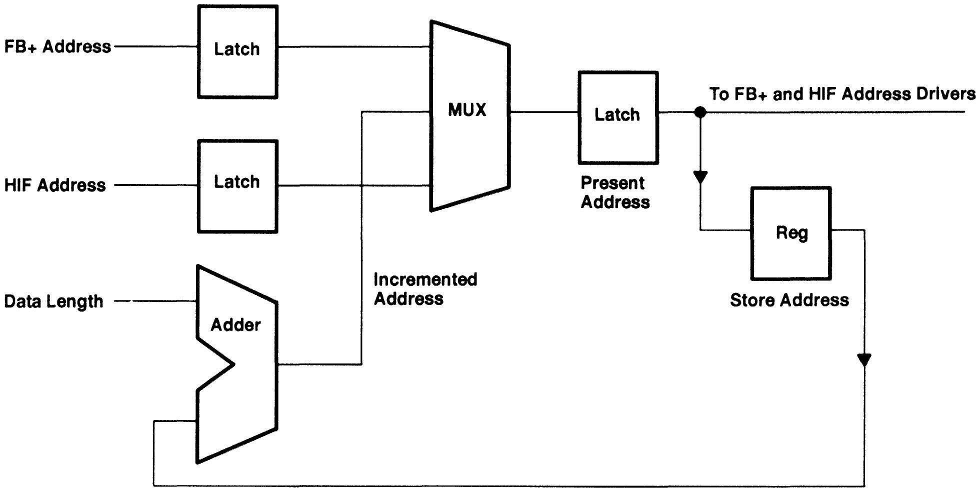


Figure 5–1. DPU Address Generation for HIF and FB+ Transactions

5.1.5 INTRA-CHIP-SET ERROR REPORTING

Errors are combined by the IOC as they are reported by the DPU and the ABC and its own internal logic to form the errorhi and errorlo error-reporting registers. The error conditions and the operation of these registers are described in the Table 5–1.

5.2 Local Versus Remote Transactions

The TI Futurebus+ chip set contains registers that define the relationship between the local memory mapping and the system memory map. Host-interface transactions are considered local to a module if the address points to a location mapped to the local host memory map. They are considered remote if the transaction address points to a location mapped to another module; this mapping is illustrated in Figure 5–2. In this example, transactions initiated by a processor on module 1 are local if the address falls within the memory base and bounds range, extended units base and bounds, or the CSR space located on module 1. A remote transaction would be required to access system memory with an address that falls within the memory base and bounds located on module 2. Local transactions are serviced locally without the need for Futurebus+ transactions. Remote transactions require Futurebus+ transactions in order to complete. The host transaction is ignored by the chip set if the IGNORE* signal is asserted (low) regardless of the programmed memory mapping.

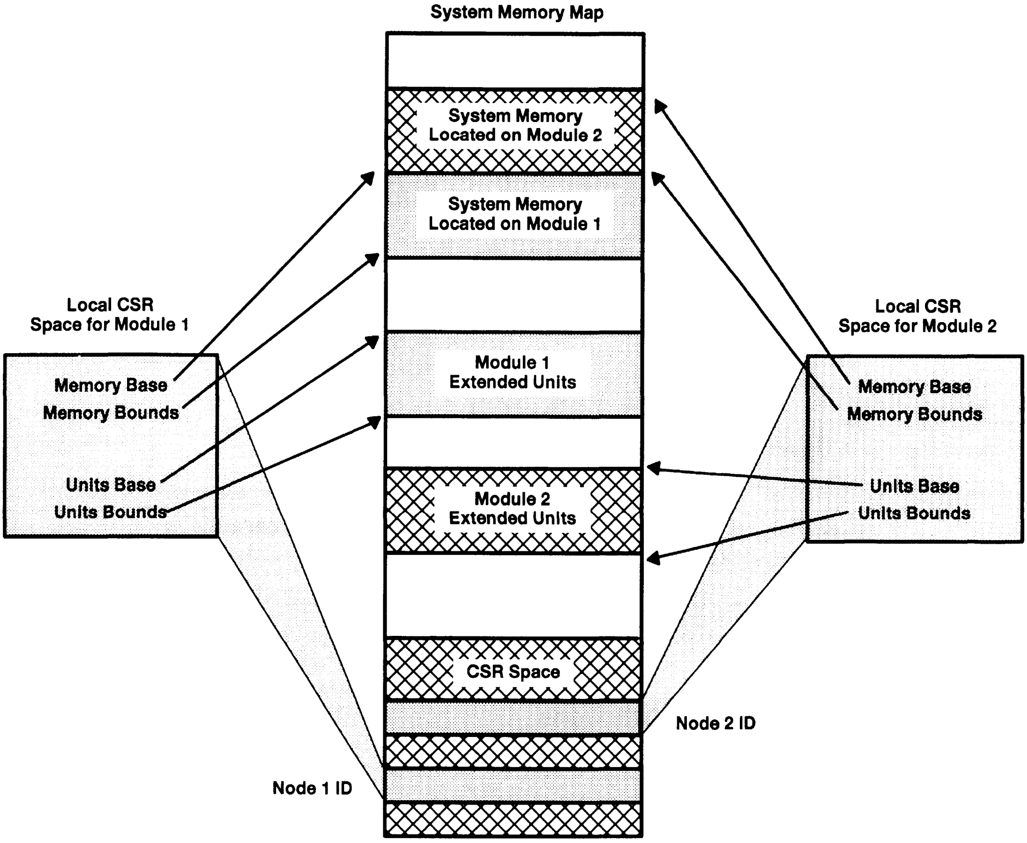


Figure 5–2. System Memory Map

By programming the memory mapping and extended-units registers, the user defines the responses taken by the chip set to host interface and Futurebus+ transactions. The relationship between the memory-mapping registers and the local space is illustrated in Figure 5–2. Table 5–2 lists the CSRs, which define the relationship between local and remote locations for each module:

Table 5–2. Relationship Between Local and Remote Locations

ADDRESS	REGISTER NAME	DESCRIPTION
8	Node IDs	Bus ID and node ID for CSR accesses
48, 52	Units Base	Extended-units starting address
56, 60	Units Bounds	Last byte address of extended units plus 1
64, 68	Memory Base	Starting address of local memory
72, 76	Memory Bounds	Last byte address of local memory plus 1

The memory and extended-unit-space enable bits in CSRs 68 and 52, respectively, affect FB+ accesses only.

5.2.1 ADDRESS DECODING

There are two types of host address decoding provided by the DPU: host-address decoding (HADEC<3:0>) and memory-space-selection decoding (MS<1:0>).

5.2.1.1 Host-Address Decoding

HADEC<3:0> are used to indicate where the host address is pointing in the system memory map. From these HADEC signals along with host control information, the TI Futurebus+ chip set can determine the type of service required to support the host transaction. Table 5–3 summarizes the HADEC encoding when the TI chip set is a slave on the host interface.

Table 5–3. HADEC Encoding

HADEC	DECODING	LOCATION	DESCRIPTION
LLLL	Unselected	Remote	Host transaction is to Futurebus+ domain
LLLH	Host Memory	Local	Host-to-host transaction, DPU does not participate
LLHL	Host Unit	Local	Host-to-host transaction, DPU does not participate
LLHH	Host CSR	Local	Host-to-host transaction, DPU does not participate
LHHL	Split Response	Remote	Host transaction is reconnecting to transaction previously split by Futurebus+.
LHHH	Futurebus+ CSR	Remote	Host transaction is to remote CSR; performs a compelled transaction on Futurebus+ to remote CSR location
HLLH	Broadcast CSR	Remote	Host transaction is to broadcast CSR; performs broadcast transaction on Futurebus+ to remote location
HHLH	DPU CSR	Local	Host transaction to DPU internal CSR; Futurebus+ is not required but IOC still performs host interface handshake loading the information into the DPU

5.2.1.2 Local-Module-Address Decoding

MS<1:0> are used to indicate to the local module which memory space is selected within the local memory map. These signals can be used by the local module's address-decode logic. They allow the module designer to avoid implementing external base and bounds registers and decoders (see Figure 5–3). MS<1:0> determine the device on the host interface that is being selected by the address. MS<1:0> are the combinational result of the address and are valid regardless of the role this chip set is playing in the ongoing host transaction (master, slave, or non-participant) except for modules using mailboxes. Table 5–4 illustrates the memory-space-selection decoding.

Table 5–4. Local-Space-Selection Encoding

MS<1:0>	HBMASTER*	SELECTED REGION
LL	L	This HIF transaction is the result of a FB+ transaction to a mailbox. (The DPU/IOC is mastering this HIF transaction). If mailboxes are not implemented, then this encoding will not occur.
LL	H	Unselected. (This HIF transaction is to either the DPU CSRs or to some remote location; the local module's memory decoder is deselected.)
LH	X	Local Memory
HL	X	Local Extended Units
HH	X	Local CSR

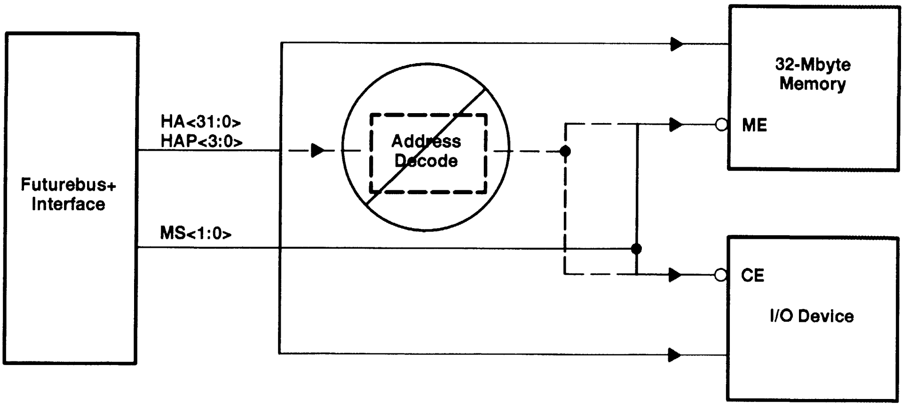


Figure 5–3. MS<1:0> Replaces Global-Address Decoding on Host Module

The decoding for MS<1:0> is internal to the DPU. When the Texas Instruments Futurebus+ chip set is a host slave (see Figure 5–4), the timing is not valid until two clocks after HIP* is asserted low. However, when the chip set is a host master (see Figure 5–5), the MS<1:0> field is valid when HIP* is asserted low.

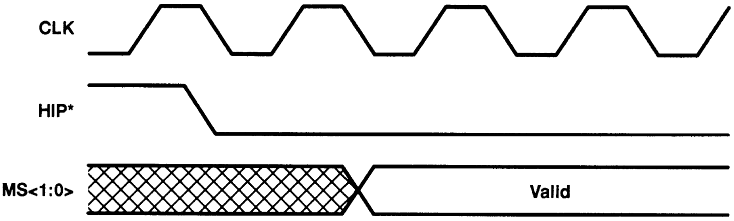


Figure 5–4. MS<1:0> Timing When Texas Instruments Chip Set is Host Slave

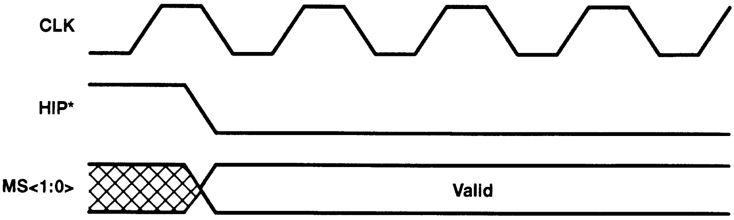


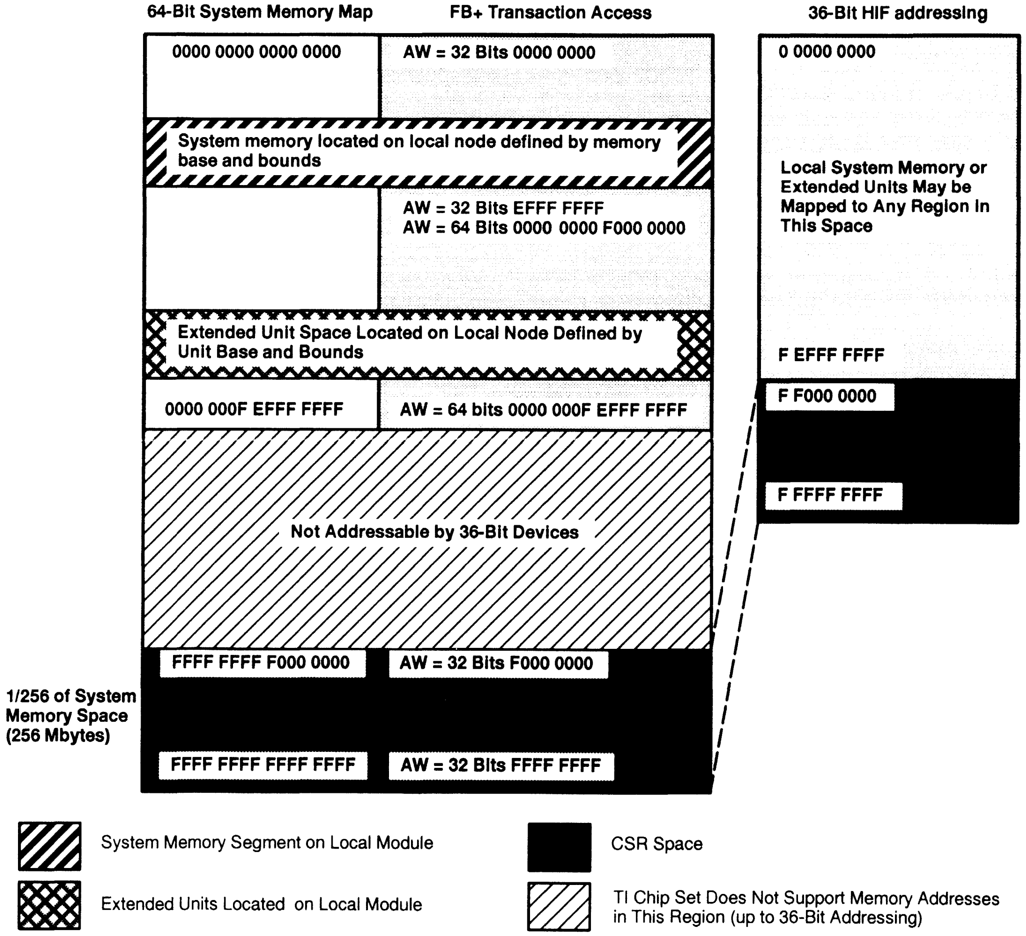
Figure 5–5. MS<1:0> Timing When Texas Instruments Chip Set is Host Master

5.2.1.3 36-Bit Host Addressing and 64-Bit Futurebus+ Addressing

The host side of the TI chip set can be accessed using 32- or 36-bit addressing. The FB+ side can be accessed using 32- or 64-bit addressing. When using 36-bit addressing, the only 36-bit host transactions converted into 32-bit Futurebus+ transactions are to CSR space. Special care should be taken in this space because 64-bit Futurebus+ addresses between 0000000000000000 to 00000000EFFFFFFF are prohibited. A 32-bit equivalent address must be used. The upper 28 bits of the address are zero extended to convert the 36-bit host address into a 64-bit FB+ address. 64-bit address FB+ transactions must contain zeroes in the upper 28 bits in order for this device to respond as a selected slave. This mechanism allows 32-bit-capable modules to coexist with 64-bit-capable address-width modules in the same backplane. The address mapping of the TI chip set is shown in Figure 5–6. The diagram shows the address ranges that a 36-bit host master can produce (in the right-most column) and demonstrates how this is converted to a FB+ transaction address (in the FB+ transaction-access column) and where this would map into a 64-bit system memory map.

The logical-module control (CSR 516, byte 3, bit 7) register controls the AW64 enable (FBAW64_en) used to master a FB+ transaction as either 32 or 64 bits. The DPU configuration (CSR 3908, byte 3, bit 3) register sets the HIF address width to either 32 or 36 bits (HBAW36_en). When FBAW64_en is set, the chip set always performs a FB+ transaction as a 64-bit address unless it is to CSR space. Likewise, when HBAW36_en is set, the DPU decodes and generates 36-bit HIF addresses.

When both FBAW64_en and HBAW36_en are set and a transaction on the HIF occurs in the range between 36'h0_0000_0000 and 36'h0_f000_0000, the chip set does not release AW* during the connection phase on FB+. In order to avoid this, the user must perform one of the following options: use 32-bit host addressing, avoid access to this space, dynamically change FBAW64_en, or use only TI parts in the system (as a selected slave, TI parts properly decode the address).



64-Bit System Memory Map
FB+ Transaction Access
36-Bit HIF addressing
0000 0000 0000 0000
AW = 32 Bits 0000 0000
System memory located on local node defined by memory base and bounds
AW = 32 Bits EFFF FFFF
AW = 64 Bits 0000 0000 F000 0000
Extended Unit Space Located on Local Node Defined by Unit Base and Bounds
0000 000F EFFF FFFF
AW = 64 bits 0000 000F EFFF FFFF
Not Addressable by 36-Bit Devices
FFFF FFFF F000 0000
AW = 32 Bits F000 0000
1/256 of System Memory Space (256 Mbytes)
FFFF FFFF FFFF FFFF
AW = 32 Bits FFFF FFFF
0 0000 0000
Local System Memory or Extended Units May be Mapped to Any Region In This Space
F EFFF FFFF
F F000 0000
F FFFF FFFF
System Memory Segment on Local Module
Extended Units Located on Local Module
CSR Space
TI Chip Set Does Not Support Memory Addresses in This Region (up to 36-Bit Addressing)

Figure 5–6. TI Chip Set 36-Bit Address Mapping

5.3 Local Transactions

Local transactions are serviced without Futurebus+ access. A local host-interface address can point to local memory space, local extended-units space, or local CSR. Local CSR transactions may point to registers located in any of the interface devices or to any user-defined CSRs located on the local host module.

5.3.1 HOST INTERFACE TO LOCAL MEMORY

When MS<1:0> equal LH, the host address is pointing to local memory. The MS<1:0> signals can be used as a chip select to the memory controller or directly to the memory chips themselves. For example, consider a module that has 1-Mbyte of SRAM memory organized as 256K × 32 (4 chips each with 256K × 8). The ID ROM contains an entry indicating to the monarch that this module has 1 Mbyte of memory (MEMORY EXTENT ROM entry); the monarch programs the memory base and bounds registers to unique values in the system memory map; the TI chip set responds to host addresses in this range by driving MS<1:0> to LH; the local module generates a chip select to the four SRAM chips when MS<1:0> equal LH. This scheme eliminates the need for address decoding for small memory segments and reduces the number of bits that address decoders must handle for larger memory segments.

5.3.2 HOST INTERFACE TO LOCAL EXTENDED UNITS

When MS<1:0> equal HL, the host address is pointing to local extended units. The term units refers to a variety of devices: I/O components such as LAN and WAN controllers, memory, maintenance and test facilities, or any other user-defined configuration or control ports. Extended units are used when the initial units space (2 Kbytes located in the local CSR space) cannot contain all the units. The ID ROM contains an entry indicating to the monarch the size of the extended-units space (extended units extent ROM entry); the monarch programs the extended units base and bounds registers to a unique location in the system memory map; the TI chip set responds to host addresses in this range by driving MS<1:0> to HL; the local module generates chip selects to each unit based upon MS<1:0> = HL and whatever additional address bits are required. This reduces the addressing requirements of the local module and eliminates the need to store the extended-units base and bounds information externally to this chip set.

5.3.3 PRIVATE MEMORY

The signal IGNORE* can be used to access private local memory. IGNORE* is asserted (low) to inform the chip set to ignore the host-interface transaction. Private memory space may be aliased to system memory locations. IGNORE* is generated by the module and supplied to the TI chip set. Address decoding for ignored addresses are the responsibility of the module designer. If there is no private memory space, this signal should be pulled high. It is important to note that the MS lines are not valid during IGNORE* transactions.

5.4 Remote Transactions

Remote transactions are those that require Futurebus+ service in order to complete. Futurebus+ service is required if the addressed location does not reside on the local module. The host-interface address along with the CSR memory-mapping registers are used to determine that the addressed location is not in the local space.

5.4.1 TYPICAL TRANSACTION SEQUENCE

All transactions are initiated by a host interface master. The master can either write to or read from a location. When the location addressed is on a different module, Futurebus+ is required to transfer the information.

As a simple model of a system, suppose there are only two modules, A and B, connected to Futurebus+ via the TI chip set. On module A there is a processor and on module B there is addressed memory. In order to service the remote host transaction from the processor, the TI chip set on module A acts as a host-interface slave and performs a Futurebus+ transaction as a master of Futurebus+. At the other end of this transaction, the TI chip set on module B identifies itself as the selected slave on Futurebus+ and masters a host interface transaction. The sequence of events for this example are described in detail in Sections 5.4.1.1 and 5.4.1.2 for both write and read transactions.

5.4.1.1 Write Example

In this example (see Figure 5–5), assume that a host-bus burst write occurs with MORE* asserted (low) to an address that maps to a remote memory location. Single compelled transfers and remote CSRs operate in similar fashion. The HIF transactions used here have been described previously. This example is intended to describe the operation of the interface logic between the IOC and DPU.

1. The processor device on module A performs a write transaction on HIF A to an address that is located on module B.
2. The DPU decodes the HIF A address and informs the IOC that the transaction is remote via HADEC = deselected (LLLL). HADEC is valid 2 clocks following the assertion of HAS* (low).
3. The IOC decides to participate in the HIF A transaction and requests Futurebus+ by asserting RQ0. Waits are inserted on the HIF by the IOC until it determines that the FIFO on the DPU can be filled from the host interface without having its transaction busied on Futurebus+ (at the end of the connection phase). Waits are inserted by keeping BSTRDY* (high), DSACK1* (high), and DSACK0* (high) released.
4. When GR is asserted by the arbiter and ET* is released from the previous master, the IOC on module A begins its tenure on Futurebus+. Capability (CA2O, CA1O, CA0O=LLL), command (CM<7:0>= 0x31), and address (AD<31:0>) signals are provided to the transceivers along with the assertion (low) of the BTL enables for the command and address transceivers (CMWR* and ADDRV*). Following this, ASO is asserted (high) to begin the Futurebus+ transaction.
5. AKO is immediately asserted (high) after detecting ASI by the IOC on module B. The DPU on module B captures the address when ASI is asserted and indicates FADEC = host memory (HLLH) to the IOC. The IOC uses FADEC to determine that it is the selected slave. It responds by asserting ST20 (high) and then releasing AIO (low). When the IOC on host B determines that it is the selected slave, it arbitrates for HIF B mastership by asserting HBR* (low). When it becomes master of HIF B (HBG* asserted and HBGACK* released), it asserts HBGACK* (low), releases HBR*, and performs the handshake with the DPU and the host B memory controller to provide the address by asserting the address, HAS* and HIP*.
6. When AII (filtered) is detected and released at the IOC on module A, the IOC completes the host transfer by performing the HIF handshakes with the HIF A master while indicating HMODE = burst between HIF and FIFO (HLH) to the DPU. When data is available for transfer to Futurebus+, the DPU asserts DATAAV* (low). DATAAV* remains asserted as long as there is data in the FIFO. When the FIFO does not have space for another host burst, SPACEAV* is released (high) indicating that no further HIF A transactions can occur until data is transmitted on FB+. When the HIF A transaction is complete, HIF A tenure is released and the next master may begin its tenure. The IOC/DPU on module A continue transferring data on Futurebus+ until the data transfer is complete. DSO is used by the IOC on module A to inform the slave that the next data packet is ready to be transferred on Futurebus+. DKI and DII are used to determine that the slave is ready for the transfer. The IOC on module B uses DKO and DIO to indicate that it is ready to receive the next data packet.
7. On module B, HDS* remains high until data is transferred into the FIFO. When the data has been transmitted to module B, the FIFO indicates that data is available to transfer to HIF B by asserting DATAAV* (low) and HDS* is asserted low. If the FIFO fills, SPACEAV* is released (high) indicating that no further data may be received from FB+ until data is transmitted to HIF B.
8. After the data transfer on Futurebus+ is complete, the IOC on module A disconnects (releases ASO) and releases its tenure (releases RQ0). The IOC/DPU on module B performs the disconnect handshake on Futurebus+ and then completes transmitting the data to the HIF B. After completing the data transfer on HIF B, HIF B is released [complete HIF B handshake and release HBGACK* (high)].

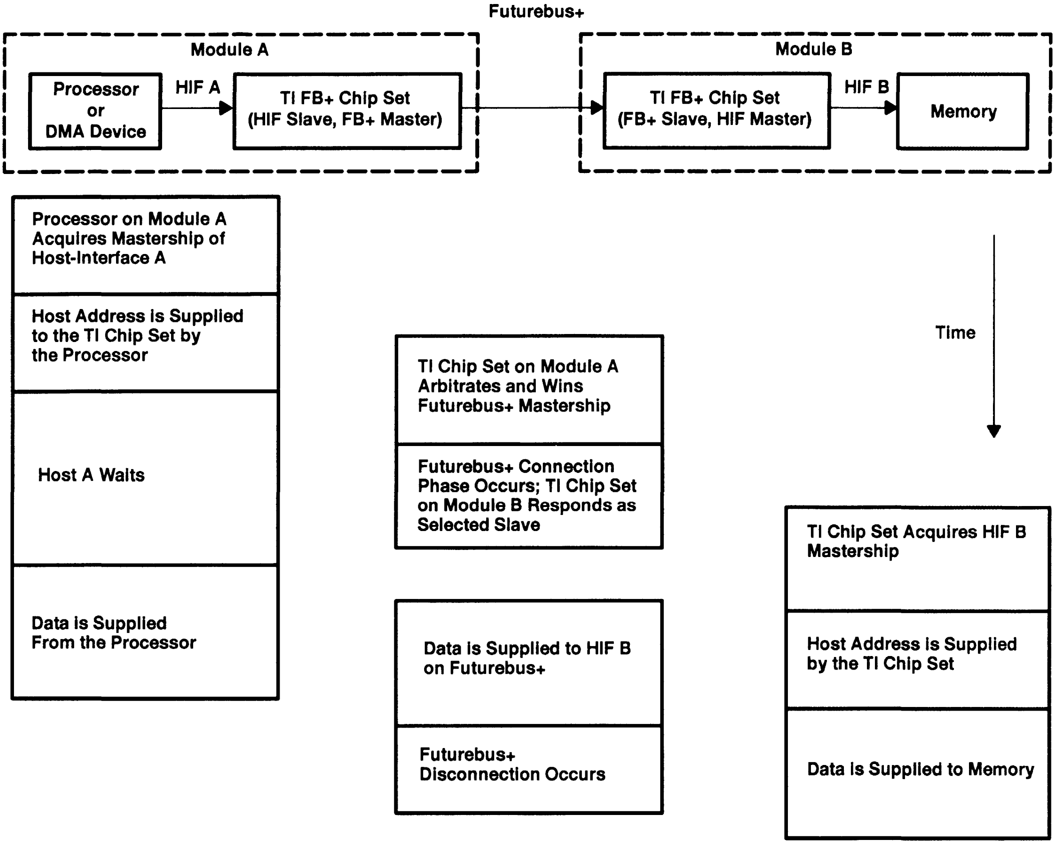

Futurebus+
Module A
Processor or DMA Device
HIF A
TI FB+ Chip Set (HIF Slave, FB+ Master)
Module B
TI FB+ Chip Set (FB+ Slave, HIF Master)
HIF B
Memory
Processor on Module A Acquires Mastership of Host-Interface A
Host Address is Supplied to the TI Chip Set by the Processor
Host A Waits
Data is Supplied From the Processor
TI Chip Set on Module A Arbitrates and Wins Futurebus+ Mastership
Futurebus+ Connection Phase Occurs; TI Chip Set on Module B Responds as Selected Slave
Data is Supplied to HIF B on Futurebus+
Futurebus+ Disconnection Occurs
Time
TI Chip Set Acquires HIF B Mastership
Host Address is Supplied by the TI Chip Set
Data is Supplied to Memory

Figure 5–7. Write Operation (End-to-End)

5.4.1.2 Read Example

In this example, a host-bus burst read occurs to an address that maps to a remote memory location. Multiple reads are linked together into a single multiple-packet Futurebus+ transfer. Single-compelled transfers and remote CSRs operate in a similar fashion. The HIF transactions used here have been described previously. This example is intended to describe the operation of the interface logic between the IOC and DPU.

1. The processor device on module A performs a read transaction on HIF A to an address that is located on module B.
2. The DPU decodes the HIF A address and informs the IOC that the transaction is remote via HADEC = deselected (LLLL). HADEC is valid 2 clocks following the assertion of HAS* (low).
3. The IOC decides to participate in the HIF A transaction and requests Futurebus+ by asserting RQ0. Waits are inserted on the HIF by the IOC until data has been transferred over Futurebus+ and is available from the FIFO on the DPU. Waits are inserted by keeping BSTRDY* (high), DSACK1* (high), and DSACK0* (high) released.
4. When GR is asserted by the arbiter and ET* is released from the previous master, then the IOC on module A begins its tenure on Futurebus+. Capability (CA2O, CA1O, CA0O = LLL), command (CM<7:0> = 0x31), and address (AD<31:0>) signals are provided to the transceivers along with the assertion (low) of the BTL enables for the command (CMWR*) and address transceivers (ADDRV*). Following this, ASO is asserted (high) to begin the Futurebus+ transaction (see Chapter 6).

5. AKO is immediately asserted (high) after the IOC on module B detects ASI. The DPU on module B captures the address when ASI is asserted and indicates FADEC = host memory (LLLH) to the IOC. The IOC uses FADEC to determine that it is the selected slave. It responds by asserting ST20 (high) and then releasing AIO (low). When the IOC on host B determines that it is the selected slave, it arbitrates for HIF B mastership by asserting HBR* (low). When it becomes master of HIF B (HBG* asserted and HBGACK* released), it asserts HBGACK* (low), releases HBR*, and performs the handshake with the host-B memory controller and the DPU to provide the address.
6. Data is taken from the HIF B as it becomes available and is placed in the FIFO in the DPU. As soon as DSI is detected asserted (high) and data is available (DATAAV* (low)), the IOC on module B asserts DKO (high) and then begins the packet transfer. When the IOC is ready for the next DSI, it releases DIO (low). The next (even) data beat begins when the DSI signal is released (low) and the next data packet is ready. In this data beat, DIO is asserted (high) first and DKO is released (low) at the end of the second packet transfer. All subsequent odd data beats operate like the first data beat and all subsequent even data beats operate like the second data beat. When the FIFO does not have space for another host burst, SPACEAV* is released (high) indicating that no further HIF B transactions can occur until data is transmitted on FB+.
7. When the HIF B transfer is complete, HIF B tenure is released and the next master may begin its tenure. The IOC/DPU on module B continues transferring data on the Futurebus+ until the data transfer is complete.
8. When the data has been transmitted to module A, the FIFO here indicates that data is available to transfer to HIF A by asserting DATAAV* (low). If the FIFO fills, SPACEAV* is released (high) indicating that no further data may be received from FB+ until data is transmitted to the HIF A.
9. After the data transfer on Futurebus+ is complete, the IOC on module A disconnects (releases ASO) and releases its tenure (releases RQ0). The IOC/DPU on module B performs the disconnect handshake on Futurebus+. The IOC/DPU on module A then completes transmitting the data to the HIF A. After completing the data transfer on the HIF A, HIF A is released [complete HIF A handshake and release HBGACK* (high)].

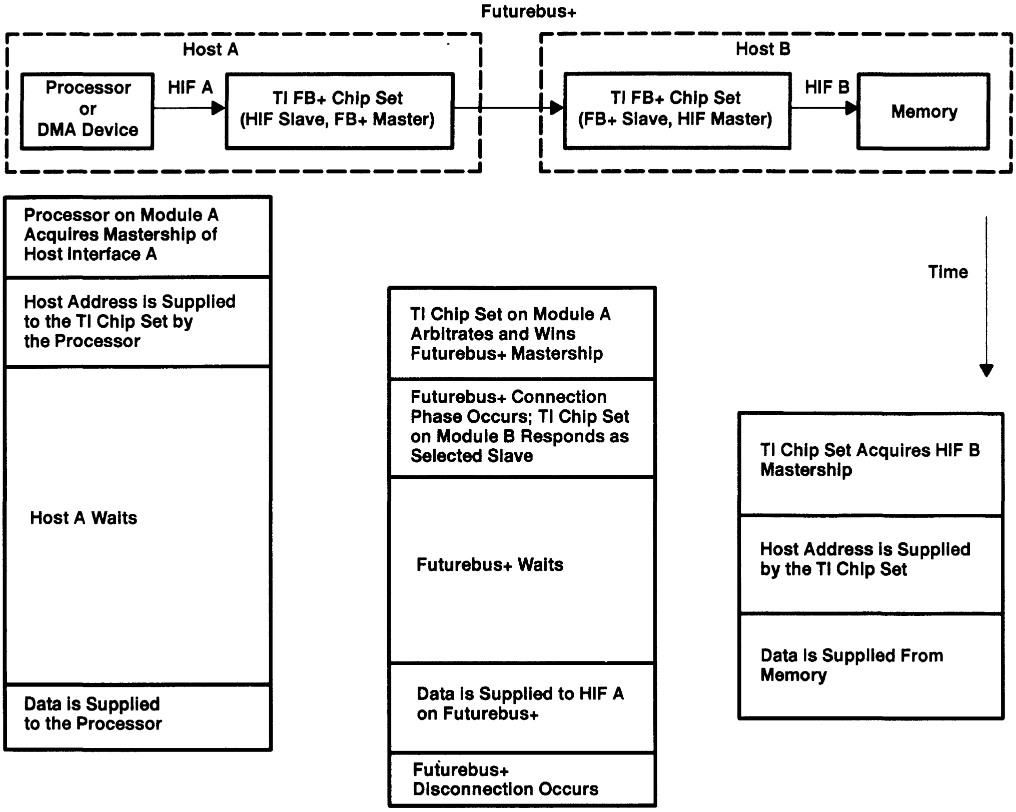

Futurebus+
Host A
Processor or DMA Device
HIF A
TI FB+ Chip Set (HIF Slave, FB+ Master)
Host B
TI FB+ Chip Set (FB+ Slave, HIF Master)
HIF B
Memory
Processor on Module A Acquires Mastership of Host Interface A
Host Address Is Supplied to the TI Chip Set by the Processor
Host A Waits
Data is Supplied to the Processor
TI Chip Set on Module A Arbitrates and Wins Futurebus+ Mastership
Futurebus+ Connection Phase Occurs; TI Chip Set on Module B Responds as Selected Slave
Futurebus+ Waits
Data is Supplied to HIF A on Futurebus+
Futurebus+ Disconnection Occurs
Time
TI Chip Set Acquires HIF B Mastership
Host Address is Supplied by the TI Chip Set
Data Is Supplied From Memory

Figure 5–8. Read Operation (End-to-End)

5.4.1.3 Read Prefetch

Read prefetching is used by the chip set in order to shorten read transactions. This feature is only used when the amount of data for the transaction is unknown. The anticipation of data beyond what is currently requested causes extra data to be loaded into the chip set. Characteristic of this read prefetch is that the amount of extra data is unpredictable because the speed of the requesting HIF, FB+, and responding HIF all affect the amount of data that is transferred. There are three events that cause the chip set to prefetch data during a read transaction:

- A MORE* read transaction initiated by HIF
- A FB+ unrestricted (FB+ DL = 3'b000) read transaction
- A FB+ multiple-packet read

During a MORE* read transaction on HIF, a prefetch occurs over FB+ and the responding HIF. More data is transferred across FB+ and the responding HIF slave than the HIF master actually requested. This situation is the only one in which more data passes over FB+ than transferred over the requesting HIF.

In the event of an unrestricted FB+ read transaction, the chip set requests data from the responding HIF either 4 or 8 bytes at a time for as long as the chip set remains in the FB+ data phase.

When FB+ attempts to read a packet from the chip set, read prefetching occurs on the responding HIF as long as the PR* signal is asserted.

In all of the above transactions, once finished, the extra data that was loaded into either DPU is purged. When a FB+ unrestricted or packet read occurs to extended-units space, the chip set only fetches (not prefetches) data from HIF as each data beat or packet occurs on FB+.

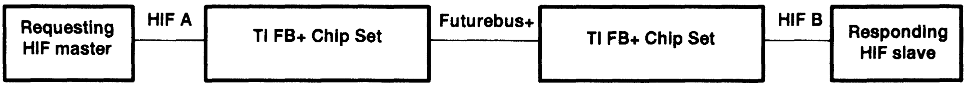


Figure 5–9. Prefetch

5.4.2 SINGLE HOST TRANSFERS

Single host transfers to remote locations on the host interface result in single compelled transactions on Futurebus+. Compelled transfers on Futurebus+ result in single host transfers on the host interface.

5.4.3 BURST HOST TRANSFERS

Burst host transfers to remote locations result in packet transfers on Futurebus+ of the same length indicated for the host transaction. Packet enables must be on; otherwise, multiple compelled transfers are generated. The host-interface signal, MORE*, can be used to generate multiple packet transfers on Futurebus+.

Packet transfers on Futurebus+ result in burst transfers on the host interface. If the host slave does not support burst transfers, the slave on the host interface is capable of downgrading the burst transfer into multiple single transfers.

5.4.4 PARTIAL TRANSFERS

Partial transfers on the host interface to remote locations result in partial transfers on Futurebus+. A read partial on the host interface is a single compelled 32-bit read operation where TSIZE is not equal to LL.

Partial transfers from Futurebus+ result in one or more partial transfers on the host interface. Discontiguous byte partials are not supported. UNALIGNED* is used to inform the IOC that two partials are required on the HIF to support the single partial that is occurring on Futurebus+. Tables 5–5 and 5–6 summarize the translation of partials to the host bus.

Table 5–5. Host Transactions Resulting From a 32-Bit Futurebus+ Partial Transaction

DESCRIPTION	Futurebus+ BYTE LANE DISABLES AD<3:0>	UNALIGNED*	HA<1:0>	HOST TRANSACTION(s)
Error – 0 byte partial	1111	NA	NA	None (error)
1 byte partial	1110 1101 1011 0111	H H H H	00 01 10 11	1 single transfer, 8 bit (TSIZE = LH) 1 single transfer, 8 bit 1 single transfer, 8 bit 1 single transfer, 8 bit
2 byte partial	1100 1001 0011	H H H	00 01 10	1 single transfer, 16 bit (TSIZE = HL) 1 single transfer, 16 bit 1 single transfer, 16 bit
3 byte partial	1000 0001	H H	00 01	1 single transfer, 24 bit (TSIZE = HH) 1 single transfer, 24 bit
4 byte partial	0000	H	00	1 single transfer, 32 bit (TSIZE = LL)

Table 5–6. Host Transactions Resulting From a 64-Bit Futurebus+ Partial Transaction

DESCRIPTION	Futurebus+ BYTE LANE DISABLES AD<7:0>	UNALIGNED*	HA	HOST TRANSACTION(s)
Error – 0 byte partial	1111 1111	NA	NA	None (error)
1 byte partial	1111 1110	H	000	1 single transfer, 8 bit (TSIZE = LH)
	1111 1101	H	001	1 single transfer, 8 bit
	1111 1011	H	010	1 single transfer, 8 bit
	1111 0111	H	011	1 single transfer, 8 bit
	1110 1111	H	100	1 single transfer, 8 bit
	1101 1111	H	101	1 single transfer, 8 bit
	1011 1111	H	110	1 single transfer, 8 bit
	0111 1111	H	111	1 single transfer, 8 bit
2 byte partial	1111 1100	H	000	1 single transfer, 16 bit (TSIZE = HL)
	1111 1001	H	001	1 single transfer, 16 bit
	1111 0011	H	010	1 single transfer, 16 bit
	1110 0111	L	011/100	2 single transfers, 8 bit
	1100 1111	H	100	1 single transfer, 16 bit
	1001 1111	H	101	1 single transfer, 16 bit
	0011 1111	H	110	1 single transfer, 16 bit
3 byte partial	1111 1000	H	000	1 single transfer, 24 bit (TSIZE = HH)
	1111 0001	H	001	1 single transfer, 24 bit
	1110 0011	L	010/100	2 single transfers, 16 bit/8bit
	1100 0111	L	011/100	2 single transfers, 8 bit/16bit
	1000 1111	H	100	1 single transfer, 24 bit
	0001 1111	H	101	1 single transfer, 24 bit
4 byte partial	1111 0000	H	000	1 single transfer, 32 bit (TSIZE = LL)
	1110 0001	L	001/100	2 single transfers, 24 bit/8 bit
	1100 0011	L	010/100	2 single transfers, 16 bit/16 bit
	1000 0111	L	011/100	2 single transfers, 8 bit/24 bit
	0000 1111	H	100	1 single transfer, 32 bit
5 byte partial	1110 0000	L	000/100	2 single transfers, 32 bit/8 bit
	1100 0001	L	001/100	2 single transfers, 24 bit/16 bit
	1000 0011	L	010/100	2 single transfers, 16 bit/24 bit
	0000 0111	L	011/100	2 single transfers, 8 bit/32 bit
6 byte partial	1100 0000	L	000/100	2 single transfers, 32 bit/16 bit
	1000 0001	L	001/100	2 single transfers, 24 bit/24 bit
	0000 0011	L	010/100	2 single transfers, 16 bit/32 bit
7 byte partial	1000 0000	L	000/100	2 single transfers, 32 bit/24 bit
	0000 0001	L	001/100	2 single transfers, 24 bit/32 bit
8 byte partial. This is equivalent to a 64-bit transaction, but it should never occur.	0000 0000	L	000/100	2 single transfers, 32 bit

5.4.5 LINKING MULTIPLE TRANSACTIONS

Multiple host-burst transactions can be linked into one multiple-packet transaction on Futurebus+ by asserting MORE* (low) with HIP* (low) during the first host transaction. MORE* must be released (high) at least one clock cycle prior to the assertion of HIP* (low) during the last linked host transaction. The TI chip set does not release ETO (low) until MORE* has been released (high) and the last transfer has begun on Futurebus+ [indicated by the assertion of ASO (high)]. Figure 5–10 illustrates multiple host-burst write transactions being linked into a single multiple packet write on Futurebus+.

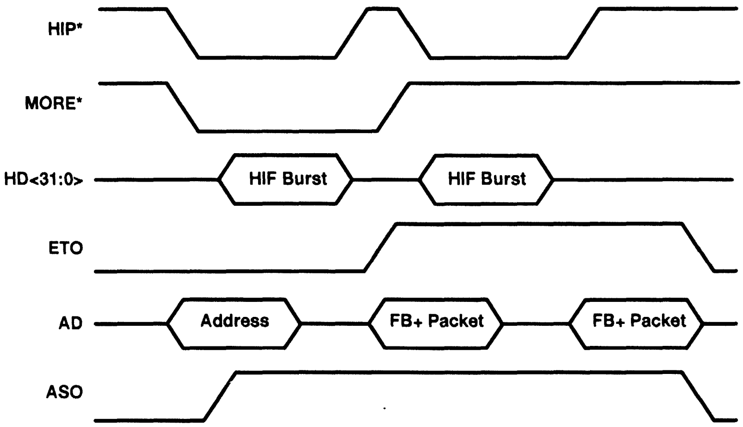


Figure 5–10. Linking Multiple HIF Transactions

5.4.6 LOCKING MULTIPLE TRANSACTIONS

Locked transactions ensure proper ordering of events in shared resources such as dual-ported RAM, shared queues, and shared uncached memory. Multiple transactions can be locked into indivisible transactions on Futurebus+ by asserting LK* (low) before the first host-interface clock that HIP* is asserted (low) during the first transaction to be locked. LK* must be released (high) at least one clock cycle prior to the release of HIP* (high) during the last locked transaction. The TI chip set does not release ETO (low) until LK* has been released (high) and the last transfer has begun on Futurebus+ [indicated by the assertion of ASO (high)]. The TI chip set performs locked transactions on Futurebus+ by asserting TC0 (CM0 = high) during the connection phase and setting LC2(CM2), LC1(CM1), LC0(CM0) equal to the value located in the locked-command extension register during the data phase.

In Figure 5–11, write transactions are locked together in the same Futurebus+ tenure; separate writes are used for each transaction. While this is not a likely locked sequence, it is used here to compare and contrast with Figure 5–10 illustrating linked transactions. Notice that LK* has replaced MORE*. Also notice the multiple transitions of ASO.

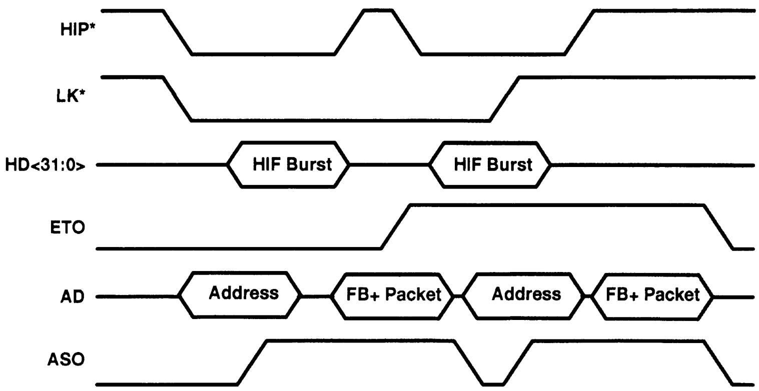


Figure 5–11. Locking Multiple HIF Transactions

5.4.7 EXCEPTION HANDLING

Four responses from a Futurebus+ slave can happen to a transaction that change the normal flow of a transaction. The responses are busy, error, intervention, or split.

5.4.7.1 Futurebus+ Busy Support

The Futurebus+ protocol allows a slave module to assert ST1* (BS*) during the connection phase of a Futurebus+ transaction. When this event occurs, the Texas Instruments chip set performs a few duties that are unique to this type of operation.

When the chip set is a Futurebus+ master and receives a busy response on a Futurebus+ transaction, four actions are initiated. The Futurebus+ transaction proceeds to the disconnection phase without transferring data, the IOC internal busy timer starts, the retry counter increments by one, and the HIF transaction is backed off. As long as the busy timer is less than the retry delay (CSR 544) after a Futurebus+ busy response, the chip set continues to back off HIF transactions and no Futurebus+ request is posted.

The busy timer stops when it reaches the retry delay. Then an HIF transaction to Futurebus+ generates a Futurebus+ request, and the chip set masters a Futurebus+ transaction when it receives a Futurebus+ grant. The busy timer resets when the chip set connects on Futurebus+ as a master. If another busy response is detected during the connection phase of the Futurebus+ transaction, the previous four steps occur again. An error is logged when the retry counter reaches the retry threshold specified in CSR 540. However, if the chip set masters a nonbusy Futurebus+ transaction, the retry counter and timer are cleared and the Futurebus+ transaction completes normally with no errors being logged.

The default value for the retry delay (CSR 544) is 0. This means that the chip set does not need to wait for the busy timer to expire before passing an HIF transaction onto Futurebus+ following a Futurebus+ busy transaction.

The default value for the retry threshold (CSR 540) is also 0. This is a special case for the retry counter and threshold. The Futurebus+ specification states that when the retry threshold is 0, a single Futurebus+ busy response generates an error. Therefore, the operation is the same whether the retry threshold is programmed as a zero or one.

5.4.7.2 Error

The assertion of the ST6* (BE*) or ST0* (TE*) signals by a Futurebus+ slave causes the chip set to:

- Set the appropriate bit in the error hi or error lo CSRs
- Set interrupt if enabled
- Disregard incoming information but continue any remaining handshakes
- Store command and status information in the error hi CSR. The address is available from the DPU address CSR.

5.4.7.3 Intervention

If the TI chip set is the selected slave and ST5* (IV*) is asserted on a Futurebus+ cache coherent Futurebus+ read transaction, it receives instead of sourcing the data from Futurebus+ and perform a write on the host interface. The TI chip set never asserts ST5* (IV*) nor does it master a cache coherent transaction since it never represents a cache.

5.4.7.4 Futurebus+ Split Support

Splits are supported by the chip set as a Futurebus+ master. When the chip set is a Futurebus+ master, the outgoing Futurebus+ transactions are allowed to be split by a slave. This is done by asserting CA<2>* during the connection phase. The chip set never splits any transactions and does not support Futurebus+ master induced splits. Futurebus+ read and write split transactions are handled differently.

At the time of the split on a Futurebus+ read transaction, the HIF master is backed off and continues to be backed off until the split response occurs on Futurebus+. While the split response is still pending, no outgoing Futurebus+ transactions are generated by the chip set. After the split response occurs (via a read response) and the originating device retries the HIF transaction, the data is finally sourced to the HIF read. This is the only time the chip set is a bus slave simultaneously to Futurebus+ and HIF. If a split time-out occurs while the chip set is waiting for a shared response, the split is cleared and the chip set is allowed to master a Futurebus+ transaction once again.

When a Futurebus+ write transaction is split, the data is transferred during the same tenure. The split write operation is completed with a Futurebus+ address-only transaction (write response) to update the error/status registers in the CSR space. The HIF is not involved in the write response.

5.4.7.5 Futurebus+ Transaction Time-Out Timer

The IOC contains a transaction timer that is controlled by the reference clock (REFCLK). This timer is optimized for a 40-MHz REFCLK. If the REFCLK for the system is different from 40 MHz, this timer scales proportionately. For example, the transaction timer default value is 122 µs. For a 33 MHz REFCLK, the actual transaction time-out is 122* 40/33 or 148 µs.

The transaction timer is started when tenure begins on Futurebus+ and is reset at the end of a transaction. When the transaction timer expires, a bus initialization is performed by the chip set.

5.5 CSR Transactions

For a CSR access to be considered local to the module, the address must correspond to the correct ID. The ID matches if all of the following conditions are true:

- The address bits HA<31:28> = h'F (for 32-bit addressing) or HAP<3:0>, HA<31:28> = h'FF (for 36-bit addressing)
- The address bits HA<27:18> = 1023 or is equal to the bus ID value in the node ID's CSR
- The address bits HA<17:12> = 63 (broadcast node ID) or 62 (self ID) or is equal to the node ID value located in the node ID's CSR

These conditions are included in the DPU address decoding logic and are indicated via HADEC or MS as indicated earlier.

CSR registers may be contained uniquely within any of the individual chips in the chip set, may be shared between two or more chips, or they may reside in a user-defined device.

5.5.1 DPU UNIQUE REGISTERS

CSR registers located on the DPU can be written directly to or read directly from the host interface. The DPU indicates to the IOC that the register is internal via the HADEC signals. The IOC then performs the correct handshake with the host interface master, allowing the DPU to receive or transmit the requested CSR information.

5.5.2 IOC AND ABC UNIQUE REGISTERS

Registers located on the IOC or ABC are programmed and read via the CSR bus. The CSR bus interface translates each 32-bit transfer on the host interface into four 8-bit transfers on the CSR bus.

The CSRs located within these two devices have been designed to completely decode the 12-bit CSR address. The IOC and ABC enable the existence of other devices in this 4-Kbyte space by forcing their outputs to the high-impedance state when the addressed register is not theirs. This simplifies the address decoding responsibilities of the CSR interface logic.

5.5.3 SHARED REGISTERS

Some registers are shared among the different devices such as the logical module-control CSR, logical common-control CSR, and bus-propagation delay CSR. These CSRs are handled easily when programmed or read from Futurebus+ but require special consideration when programmed from the local host module. Local programming of these CSRs is likely to be constrained to monarch self-programming.

When accessed from Futurebus+, shared CSRs are easy to program and read as long as all CSRs reside on the CSR bus or in the DPU. The Futurebus+ CSR access is transferred to the host interface as a 32-bit transaction; the DPU is programmed if necessary; the CSR bus interface logic responds to the transaction and performs the necessary service on the CSR bus. The CSR locations are programmed or read.

The host interface lacks the capability to perform broadcast transactions, which complicates the programming of CSRs shared between the DPU (or other devices that handshake directly on the host interface) and devices that are located on the CSR bus. For example, the system monarch's local writes to shared CSRs between its own DPU and IOC must be performed twice: once to its DPU and then to its IOC via the CSR bus. Following the write to the DPU CSR location, the DPU CSR's input acceptance must be turned off in order to do the second write to the IOC. The DPU CSR responses are shut off via the DPU configuration register bit 3 of byte 2.

Alternately, the CSR interface logic can be designed to shadow the write to the DPU (receiving the data from the host interface at the same time that it is written to the DPU). Since the handshake control of the CSR interface has been eliminated for this case, the module designer must be certain the CSR interface logic is ready to receive data at the same rate as the DPU. This operation would mimic a broadcast operation, and care should be taken to ensure timing compatibility.

Shared registers located only on the CSR bus are easily accessed by software since this bus allows broadcast operations. Special considerations must be made to ensure that one unique device responds to each read operation. The IOC and ABC comprehend shared CSRs in their address-decoding schemes. Writes to CSRs contained in both the IOC and ABC are stored in both devices; reads are serviced by the IOC sourcing the most recently written data while the ABC keeps its data bus in the high-impedance state.

Some CSRs are not only shared among CSRs on the CSR module but are shared with other modules. These CSRs include the logical common-control CSR and the bus-propagation delay CSR. The sequence of events to broadcast information to these CSRs is shown in Figure 5–12.

Figure 5–12 is an example of the operations a monarch performs to broadcast information to its own CSRs and then to remote locations. In step 1 the monarch processor programs its own DPU to disable shared CSRs. In step 2 it programs its own IOC shared CSRs. In step 3 it programs the DPU to respond to shared CSRs and programs its DPU. In step 4 the remote DPU receives the FB+ CSR transaction and updates the appropriate CSR, and it passes this information on to the CSR bus with a HIF transaction. In step 5 the remote IOC receives the CSR bus transaction and updates the appropriate CSR. Steps 4 and 5 are performed with a single HIF transaction on module A resulting in a single FB+ transaction that results in a single HIF transaction on module B.

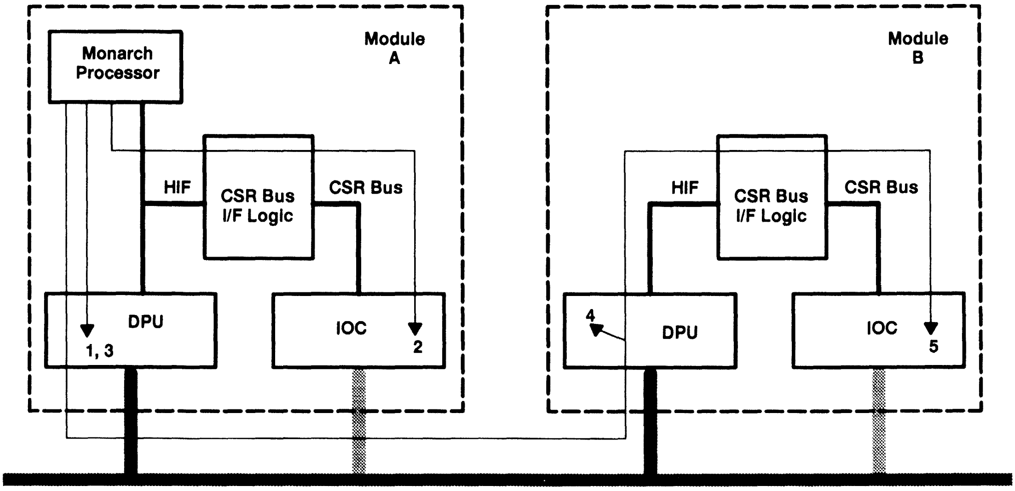


Figure 5–12. Monarch Broadcasting Information to its Own CSRs and Remote Location CSRs

5.5.4 USER-DEFINED CSRs

User-defined CSRs and units may coexist with the TI chip set. They may be shared or unique as long as the conditions mentioned previously have been met. Addresses between 2048 and 3712 in the CSR space have been left open for use by the module designers. If additional unit space is required, the module may use the units base and units-bounds CSRs to create a local extended-units address space.

5.6 Futurebus+ Transaction Commands

5.6.1 UNCACHED COMMANDS

5.6.1.1 Read Unlocked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF block read or an HIF single 32-bit read to Futurebus+. As a Futurebus+ slave, this command is converted into an HIF single or block read.

5.6.1.2 Write Unlocked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF block write or an HIF single 32-bit write to Futurebus+. As a Futurebus+ slave, this command is converted into an HIF single or block write.

5.6.1.3 Address-Only Unlocked

This command is never generated by the Texas Instruments chip set as a Futurebus+ master. As a Futurebus+ slave, this command is converted into an HIF address-only transaction.

5.6.1.4 Read Locked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF block read or an HIF single 32-bit locked read from Futurebus+. As a Futurebus+ slave, this command is converted into an HIF single or block read locked. The lock flag is held asserted on all buses as long as the originator of the lock asserts LK. The locked commands are generated from LKFLD [2:0] on HIF and are passed into the HIF during a lock operation. The IOC and/or DPU do not perform the locked commands; the chip set just passed along the information.

5.6.1.5 Write Locked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF block write or an HIF single 32-bit locked write from Futurebus+. As a Futurebus+ slave, this command is converted into an HIF single or block write locked. The lock flag is held asserted on all buses as long as the originator of the lock asserts LK. The locked commands are generated from LKFLD [2:0] on HIF and are passed into the HIF during a lock operation. The IOC and/or DPU do not perform the locked commands; the chip set just passed along the information.

5.6.1.6 Address-Only Locked

This command is never generated by the Texas Instruments chip set as a Futurebus+ master. As a Futurebus+ slave, this command is converted into an HIF address-only transaction. The lock flag is held asserted on HIF as long as the originator of the lock asserts LK.

5.6.1.7 Read Partial

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF 8-, 16-, or 24-bit read from Futurebus+. The Texas Instruments chip set only masters 32-bit partial transactions. As a Futurebus+ slave, this command is converted into a HIF single and/or one or two HIF partial reads. As a Futurebus+ slave, 64-bit partials are supported. Only contiguous byte lines are supported.

5.6.1.8 Write Partial

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF 8-, 16-, or 24-bit write to Futurebus+. The Texas Instruments chip set only masters 32-bit partial transactions. As a Futurebus+ slave, this command is converted into an HIF single and/or one or two HIF partial writes. As a Futurebus+ slave, 64-bit partials are supported. Only contiguous byte lines are supported.

5.6.1.9 Read Partial Locked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF 8-, 16-, or 24-bit read from Futurebus+. The Texas Instruments chip set only masters 32-bit partial transactions. As a Futurebus+ slave, this command is converted into a HIF single and/or one or two HIF partial reads. As a Futurebus+ slave, 64-bit partials are supported. Only contiguous byte lanes are supported. The lock flag is held asserted on all buses as long as the originator of the lock asserts LK.

5.6.1.10 Write Partial Locked

This command is supported as a Futurebus+ master and as a Futurebus+ slave. This command is generated as a Futurebus+ master on an HIF 8-, 16-, or 24-bit write to Futurebus+. The Texas Instruments chip set only masters 32-bit partial transactions. As a Futurebus+ slave, this command is converted into an HIF single and/or one or two HIF partial writes. As a Futurebus+ slave, 64-bit partials are supported. Only contiguous byte lanes are supported. The lock flag is held asserted on all buses as long as the originator of the lock asserts LK.

5.6.1.11 Write Response

This command is used as a write split response. The Texas Instruments chip set never generates this command as a Futurebus+ master since it never splits a transaction. As a slave, it is recognized once a split has occurred.

5.6.1.12 Read Response

This command is used as a read split response. The Texas Instruments chip set never generates this command as a Futurebus+ master since it never splits a transaction. As a slave, it is recognized once a split has occurred.

5.6.1.13 Write No Acknowledge

The Texas Instruments chip set generates this command as a Futurebus+ master. As a Futurebus+ slave, this command is treated the same as a write unlocked.

5.6.2 CACHE COMMANDS (Intervention is Supported as a Slave)

5.6.2.1 Read Invalid

Supported as a Futurebus+ slave, the module sources data the same as read unlocked if there is no intervention.

5.6.2.2 Write Invalid

Supported as a Futurebus+ slave, the module is the destination for data the same as write unlocked.

5.6.2.3 Read Shared

Supported as a Futurebus+ slave, the module sources data the same as read unlocked if there is no intervention.

5.6.2.4 Copyback

Supported as a Futurebus+ slave, the module receives data the same as write unlocked.

5.6.2.5 Read Modified

Supported as a Futurebus+ slave, the module sources data the same as read unlocked if there is no intervention.

5.6.2.6 Invalidate

Invalidate is not supported. The chip set is not selected.

5.6.2.7 Shared Response

Supported as a Futurebus+ slave, the module receives data the same as write unlocked in order to update memory.

5.6.2.8 Modified Response

Modified response is not supported. The chip set is not selected.

5.6.3 RESERVED COMMANDS

Reserved commands are not supported. The Texas Instruments Futurebus+ chip set does not generate any of the reserved commands; the chip set does not signal selected when a reserved command is received.

5.7 Futurebus+ Data-Width Support

There are four standard data widths for Futurebus+ transactions (32, 64, 128 and 256 bits). The Texas Instruments Futurebus+ chip set correctly encodes, decodes, and performs all 32- and 64-bit data-width transactions; however, the chip set never generates a 128- or 256-bit data-width transaction. When the chip set is a Futurebus+ slave, a 256-bit data width causes the chip set to not operate. When the chip set is a selected slave to a 256-bit data width, the Futurebus+ and host handshakes do not complete properly requiring a system reset to get the chip set into a known state. When the chip set is a selected slave to a 128-bit data-width transaction, the handshake is correct but the data is not because only one path unit supplies and receives data for each bus.

5.8 Futurebus+ Data-Length Support

There are seven Futurebus+ defined data lengths (1, 2, 4, 8, 16, 32 and 64) and one unrestricted data length. Compelled-mode transactions allow the use of all possible data lengths, whereas packet-mode only utilizes 2-, 4-, 8-, 16-, 32- and 64-bit data lengths. The term data length refers to the number of beats in compelled mode or the number of bits in the packet mode of operation. By evaluating the data width and length, the total number of bytes being transferred in a Futurebus+ transaction is determined. For example, a Futurebus+ transaction that utilized a 64-bit data width and a data length of 8 compelled beats or a single 8-bit packet (DL equals 8 in both cases) is a 64-byte transaction.

The chip set does not utilize all of the data lengths available. As a Futurebus+ master, data-length support of up to 16 is autonomous where the combination of data length and data width does not exceed 64 bytes except for unrestricted compelled transfers. As a Futurebus+ slave, the chip set provides full support for all data lengths in compelled mode, but the 64-bit data length is not supported in packet mode. Even though the chip set supports limited data lengths, unrestricted compelled mode and multiple packet transactions enable the chip set to transfer an unlimited amount of data in a single tenure.

5.9 Futurebus+ Message Mailbox Support

The chip set supports the mailbox message-passing scheme. The message-passing scheme either sends the message to the specific node's mailbox or sends the message to the broadcast message mailbox of all the nodes. Since the Futurebus+ protocol stipulates that mailboxes may only be written, write unlocked and write no acknowledge are the only transaction types used for sending a message across Futurebus+. There are four different message-passing transactions:

- Futurebus+ sends a message to the specific node's mailbox
- Futurebus+ sends a message to a mailbox of all the nodes
- HIF sends a message to a Futurebus+ specific node's mailbox
- HIF broadcasts a message to the Futurebus+ for all the nodes' mailbox

5.9.1 SYSTEM MESSAGE SLAVE MAILBOX

Futurebus+ sends a message to the chip sets mailbox if the transaction is write unlocked or write no acknowledge and the address matches all of the following conditions:

- Futurebus+ address is 32 bits wide
- The address bits AD<31:28> = h'F
- The address bits AD<27:12> = <bus id, node id> in the node IDs CSR or AD<27:12> = <h'3FF, geographical address, b'0>
- The address bits AD<11:0> are greater than or equal to h'080 and are less than or equal h'0FC

When Futurebus+ sends a message to the chip sets mailbox, the chip set is a Futurebus+ slave, FADEC is HLLL (mailbox address), and MS is LL (deselected). The address and data are sent out to the HIF as a write transaction on the HIF (chip set is HIF master), and the address may not align to the word boundary.

5.9.2 SYSTEM BROADCAST MESSAGE

Futurebus+ broadcasts a message to the chip sets mailbox if the transaction is write unlocked or write no acknowledge and the address matches all of the following conditions:

- Futurebus+ address is 32 bits wide
- The address bits AD<31:28> = h'F
- The address bits AD<27:12> = h'FFFF or AD<27:12> = <bus id, h'3F>
- The address bits AD<11:0> are greater than or equal to h'080 and are less than h'0C0
- The address bits AD<5:0> = 0 or the bit pointed by AD<5:0> in message-passing-select mask CSR 532, 536 is set to 1

When Futurebus+ sends a message to the chip sets mailbox, the chip set is a Futurebus+ slave, FADEC is LHLL (broadcast address), MS is LL (deselected), and the Futurebus+ status indicates a broadcast transaction. The address and data (which contains the message) send to the HIF a regular write transaction on the HIF side (chip set is HIF master). The address may not align on the word boundary.

5.9.3 LOCAL MESSAGE SLAVE MAILBOX

The HIF sends a message to the Futurebus+ specific nodes mailbox with a write transaction where the address matches all of the following conditions:

- The address bits HA<31:28> = h'F (for 32-bit addressing) or HAP<3:0>, HA<31:28> = h'FF (for 36-bit addressing)
- The address bits HA<27:12> are not equal to the local <bus id, node id> in the node IDs CSR and not equal to the <h'3FF, geographical address, b'0> and HA<17:12> are not equal to h'3F
- The address bits HA<11:0> are greater than or equal to h'080 and are less than or equal h'0FC

When the HIF sends a message to a Futurebus+ nodes mailbox with a write transaction, the chip set is an HIF slave and a Futurebus+ master, HADEC is LLLL (deselected), and MS is LL (deselected). The address and data (which contains the message) are sent out to the Futurebus+ as a write-unlocked transaction. The address may not align on the word boundary.

5.9.4 LOCAL BROADCAST MESSAGE

The HIF broadcasts a message to any Futurebus+ nodes mailbox with a write transaction where the address matches all of the following conditions:

- The address bits HA<31:28> = h'F (for 32-bit addressing) or HAP<3:0>, HA<31:28> = h'FF (for 36-bit addressing)
- The address bits HA<17:12> are equal to h'3F
- The address bits HA<11:0> are greater than or equal to h'080 and are less than h'0C0

When the HIF sends a message to a Futurebus+ nodes mailbox with a write transaction, the chip set is a HIF slave and a Futurebus+ master, HADEC is LHLL (broadcast mailbox), and MS is LL (deselected). The address and data (which contains the message) are sent to Futurebus+ as a write-unlocked transaction. The address may not align on the word boundary.

Chapter 6

Arbitration

6 ARBITRATION

6.1 Arbitration Overview

The arbitration process provides a method for modules to equitably share the bus resource for intermodule communication. Before sending data to or obtaining data from another module, a module must first gain mastership of the bus using the arbitration process. The time that any master has control of the bus is referred to as its bus tenure. Since two or more modules may seek to gain tenure of the bus at the same time, the arbitration process is used to restrict tenure of the bus to only one winning module at a time. Modules are awarded use of the bus based on a priority number. A fairness algorithm can be used to ensure equitable use of the bus by modules competing at the same priority level. When priority and fairness do not provide a single arbitration winner, a geographical address assigned by the card slot within the box provides a final method of distinction.

6.1.1 CENTRAL VERSUS DISTRIBUTED ARBITRATION

Two arbitration methods are provided by the Futurebus+ specification and supported by the Texas Instruments Futurebus+ chip set: central and distributed-mode arbitration. The central arbitration scheme uses separate request and grant signals between each module and a central arbitration facility called a central arbiter. The central arbiter resolves simultaneous requests and sequences grants to the requesting modules. Distributed arbitration resolves simultaneous requests and sequences grants through a competitive process amongst the modules using a dedicated distributed arbitration bus. The arbitration bus can be viewed as an entirely separate bus from the parallel protocol bus. It uses its own synchronization and data lines and its own unique protocol. The distributed arbitration bus is also used in both central and distributed-mode systems to send messages separately from the primary data transfer bus. These messages may be used to convey interrupts or system events between modules.

6.1.1.1 Fully Distributed-Mode System

In the distributed mode, the ABC distributed arbiter uses the arbitration bus to perform bus acquisitions for the host module and to send/receive distributed arbitration messages. Figure 6–1 demonstrates a two-module distributed-mode system.

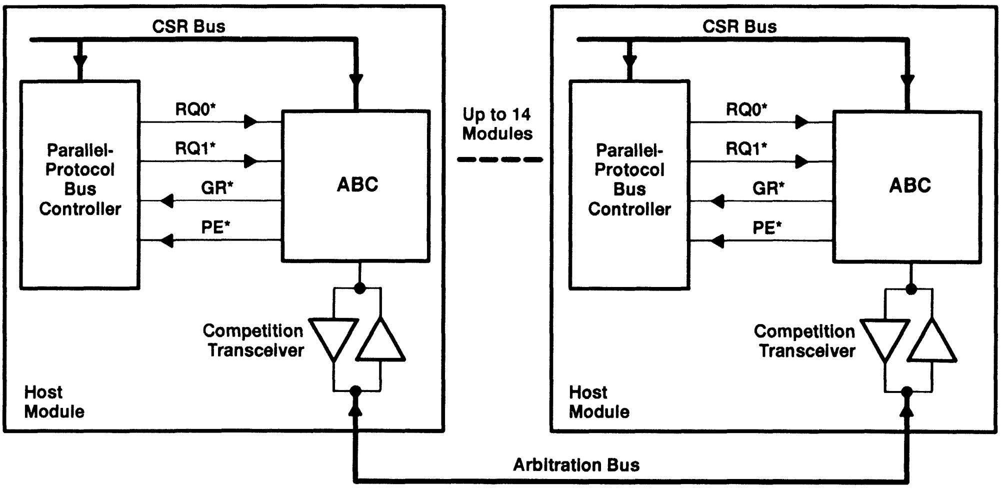


Figure 6–1. Distributed Arbitration Configuration

6.1.1.2 Programmable Central-Mode Bus Arbitration With Arbitration Messages

In the central mode, a distributed arbiter such as the ABC uses the arbitration bus to program its module's priorities into the central bus arbiter and to send/receive arbitration messages. A central arbiter, such as the future PCBA, handles the bus-acquisition process for all modules. A full ABC distributed arbiter is contained within a PCBA providing both functions in one package. This allows the central arbitration module to send/receive arbitration messages as well as receive the central arbitrated messages used by the system's distributed arbiters to program the module arbitration priorities into the PCBA. Central arbitration minimizes the time to grant the bus to a requester and provides the highest performance. Figure 6–2 displays a typical central-mode system configuration.

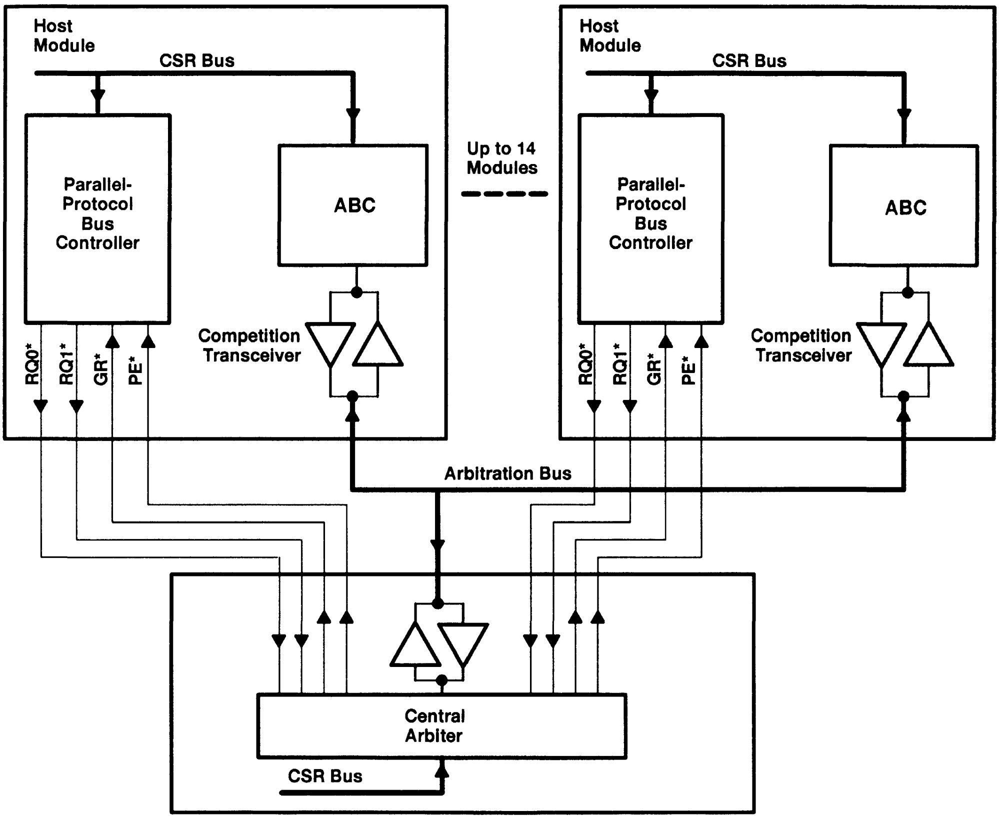


Figure 6–2. Central Arbitration Configuration With Distributed Arbitration for Messages

6.1.2 DISTRIBUTED ARBITRATION PROTOCOL

In response to a module's need for arbitration, the distributed-arbitration protocol is used over the dedicated arbitration bus. The distributed-arbitration protocol uses a three-wire asynchronous handshake. AP*, AQ*, and AR* are wire-ORed signals simultaneously monitored and controlled by all distributed-arbitration capable modules to sequence through a series of distributed arbitration phases. An arbitration phase change is instigated when a module asserts one of the handshake signals. Other modules respond by assertion of the same signal. Because the bus is wired ORed, handshake signal releases are not recognized until all modules have released the signal. Prior to releasing a handshake signal, each module will have completed the required action for the phase. Phase changes on handshake signal releases are governed by the slowest module. This technique compels the arbitration function keeping modules of varying speeds synchronized with each other.

6.1.2.1 Arbitration Phase Description

Phase 0 (idle phase) occurs when the arbitration bus is idle waiting for a module to request arbitration. Phase 0 may also be the midpoint between a two-pass arbitration; no new competitors may enter competition if this is the second pass. A module asserts AP* to move to phase 1.

Phase 1 (decision phase) is the last chance for modules to decide whether to compete for the bus during the current arbitration cycle. Competitors enable competition priorities to the bus. Noncompeting modules continue to follow the arbitration handshake. Phase 1 is complete when all modules have released the AR* handshake signal.

Phase 2 (competition phase) occurs when all competing modules have their competition numbers driven to the arbitration bus. The competition transceivers resolve the winner, and the winning transceiver's WIN signal is asserted. ABC waits for the competition process to settle using the arbitration settling time value prior to observing the WIN signal. AQ* is asserted initially by the winning module to instigate transition to phase 3.

Phase 3 (error-check phase) occurs when all modules evaluate the winning competition number. Each module determines the arbitration type (i.e., message or bus acquisition), whether a second pass is required, and if any errors have occurred. For the final pass of a distributed-mode bus acquisition, the winning module becomes the master elect and all modules wait in this phase for the master to complete the current transaction (indicated by the release of AS*). When the AS* release is detected for arbitrated messages,or when a second pass is required, the arbitration process moves to phase 4 when all modules have released AQ*.

Phase 4 (master release) occurs when, for distributed-mode bus acquisition, the current bus master has the option to set a flag cancelling transfer of tenure in phase 5 so it can perform additional transactions. All modules wait for the bus master to assert AR* and initiate the move to phase 5. In the central mode, for arbitrated messages or when a second arbitration pass is required, all modules move to phase 5 by asserting AR*.

Phase 5 (tenure/message transfer) occurs when, if no errors have occurred and another arbitration pass is not required, the master elect becomes the master or the message is accepted by all modules. The arbitration process moves to phase 0 when all modules have completed this phase and released AQ*. If another pass is required, the second pass begins in phase 0 and the process is repeated.

6.1.2.2 Arbitration Conditions

During phase 2 of the arbitration process, all competing modules' competition numbers are asserted on the arbitration bus and the competition logic on each module (see arbitration transceiver) resolves the highest priority present on the bus. The arbitration mode (central or distributed) and the highest priority competition number allow the arbitration devices to determine the type of arbitration (i.e., message, bus acquisition, priority update). Depending on the type of arbitration, one or two passes through the six arbitration phases and one or two competitions may be required to resolve a single winner. Since the arbitration bus is only eight bits wide, competitions requiring resolution of more than eight bits cannot be resolved after only one pass. Modules winning during the first pass compete again in the second pass leaving only one winner of a complete arbitration cycle. Competition numbers for each type of arbitration are detailed in later sections.

Two arbitration condition signals AC1* and AC0* are provided in Futurebus+ to indicate errors and provide additional control of the arbitration handshake process. Table 6–1 summarizes the conditions causing the assertion of the arbitration condition signals and in which phase the assertion occurs. The signals are cleared in phase 0.

6.1.2.3 Arbitration Messages (Central and Distributed Mode)

Arbitration messages are used to convey system events, interrupts, or general information from one module to one or more modules over the arbitration bus. Futurebus+ defines certain competition numbers to be messages. For messages, the most significant bit of the 8-bit competition number is always 1 creating 128 message values. This makes messages the highest priority competition numbers. The message values sent are the same for distributed arbitrated messages (distributed mode) and central arbitrated messages (central mode). In the central mode, the message can be sent with one arbitration pass. For the distributed mode, a two-pass arbitration is required to distinguish a message from a single-pass bus-acquisition cycle. The first pass uses a competition number of 0xFF that gives messages the highest priority.

Table 6–1. Arbitration Phases and Conditions

AC0*	AC1*	PHASE	CONDITION
0	0	3	Error condition, i.e., winner did not have highest priority or parity error was detected.
1	0	3	A module desires to depose the master elect (distributed mode).
		3	A second arbitration pass is required.
		3	Final pass of distributed-mode arbitrated message. AC1 is used to distinguish this from the second pass of a bus acquisition operation and prevent transfer of tenure.
		4	Distributed-mode master has decided to retain the bus for additional transactions. AC1 prevents transfer of tenure.
1	1		No errors or special conditions
0	1		Illegal

6.1.2.4 Central Arbitrated Messages (Central-Mode Only)

Central arbitrated messages (central-mode only) are used to program or update module arbitration priorities into the central bus arbiter. 256 priorities may be programmed for each of two request levels. Central arbitrated messages require two passes of the distributed arbitration process. Only the central arbiter need receive these messages.

6.1.2.5 Distributed-Mode Bus Acquisition (Distributed-Mode Only)

In the distributed mode, the arbitration process is used to assign a requesting module as master of the data transfer bus. The arbitration bus may be used to send messages or elect the next bus master independent of the current master's activity on the data transfer bus. The module elected to be the next bus master is called the master elect. This module waits at an interim point in the arbitration process for the master to finish using the data bus. When the master is through, the arbitration process is completed, the master elect becomes the master, and a new arbitration may begin. Performing a majority of the arbitration process in parallel with the master's bus tenure decreases the time between a request for the bus and a grant. While the master elect is waiting to become master, however, a higher priority arbitration may arise. For example, a higher priority module may request the bus or a distributed arbitration message may need to be sent by another or even the same module. In this case, the master elect may be deposed or preempted while the arbitration bus is used for another higher priority purpose. The deposed master elect has to arbitrate for the bus again when the higher priority processes are complete.

6.1.3 CENTRAL ARBITRATION PROTOCOL

For systems containing a central bus arbiter, the central arbitration protocol may be used for bus acquisition. A distributed arbiter in this type of system would only be used for messages and priority programming. The central-mode protocol uses a two-signal asynchronous handshake. Dedicated Futurebus+ signals, two request signals (RQ0 and RQ1), and a grant signal (GR) are used to carry out the handshake between a module and the central bus arbiter. The two request signals correspond to different priority levels contained in the central bus arbiter. These priority levels may also be programmable. A module asserts the request (RQ0 or RQ1) corresponding to the urgency or priority of the task to be performed over the Futurebus+. The central bus arbiter reviews requests from all modules and grants the bus to the module with the highest priority. When a module is finishing its last transaction, it releases the request signal, and the grant for this module is released by the central bus arbiter. A grant is then asserted by the central bus arbiter to the next arbitration winner. The newly granted module may begin using the bus when the Futurebus+ ET* signal is released by the current bus master indicating the last transaction is finished.

Prior to receiving a grant, a module may elect to increase its request priority by asserting the second request signal. This may be done when a task of higher priority arises after the request for a lower priority task has already been asserted. Both requests are released when the module is preparing to end its bus tenure. In other words, even though there are two request signals, the handshake still operates like a simple two-wire asynchronous handshake.

The preemption signal (PE) is broadcast to all modules by the central bus arbiter to inform the current master that a module of higher priority is waiting for the bus. It is then up to the master to release its request and finish with the bus so the central arbiter can grant the bus to the higher priority module.

6.2 ABC Distributed-Arbitration Bus Controller

The ABC distributed-arbitration bus controller operates in either central or distributed modes. In the central mode, the ABC performs the distributed-arbitration protocol to send and receive general arbitrated messages and to send central arbitrated messages. In the distributed mode, the ABC performs the distributed-arbitration protocol to send/receive distributed messages and to gain data transfer bus tenure for the module.

6.2.1 ABC INTERFACES

6.2.1.1 Futurebus+ Interface

The ABC interfaces to Futurebus+ through a set of transceivers to buffer local module-signal levels to Futurebus+ BTL signal levels. Distributed arbitration handshake signals (AP*, AQ*, and AR*) and arbitration condition signals (AC0 and AC1) are bidirectional signals that are divided into separate input and output signals by the transceivers to interface with the ABC. Address strobe (AS*), geographical address (GA<4:0>*), and reset (RE*) are all buffered inputs from Futurebus+.

The ABC works in conjunction with a competition transceiver such as the SN74FB2032. A competition number with parity is loaded into the transceiver by the ABC over CN<7:0> and CDP and then latched by the LE signal. When the ABC asserts CMPT* to compete, the transceiver applies the module's competition number to the arbitration bus. Contention logic in the competition transceiver works in conjunction with the other modules to determine which competition number on the bus has the greatest priority. The module's transceiver with the greatest priority asserts WIN to the corresponding ABC to convey the competition victory. Each module's transceiver makes the winning competition number available over CN<7:0> and CDP when OEA is asserted by the ABC.

If the ABC is used in a system that utilizes a central bus arbiter, RQ1, RQ0, GR, and PE can be conveniently connected for central- or distributed-mode operation using the central-mode signal to control transceivers as shown in Figure 6–3. Systems using only distributed mode do not require transceivers for RQ1, RQ0, or GR. The ABC observes PE during system reset to determine the mode of operation. Connecting PE allows a central bus arbiter to assert the PE signal during system reset and put the ABC in the central mode. If no central arbiter is present, PE is pulled up on Futurebus+ causing a low PE input to the ABC and the distributed mode is enabled. Alternatively, the PE input may be resistively tied low to permanently indicate distributed-mode operation for modules that never operate in the central mode.

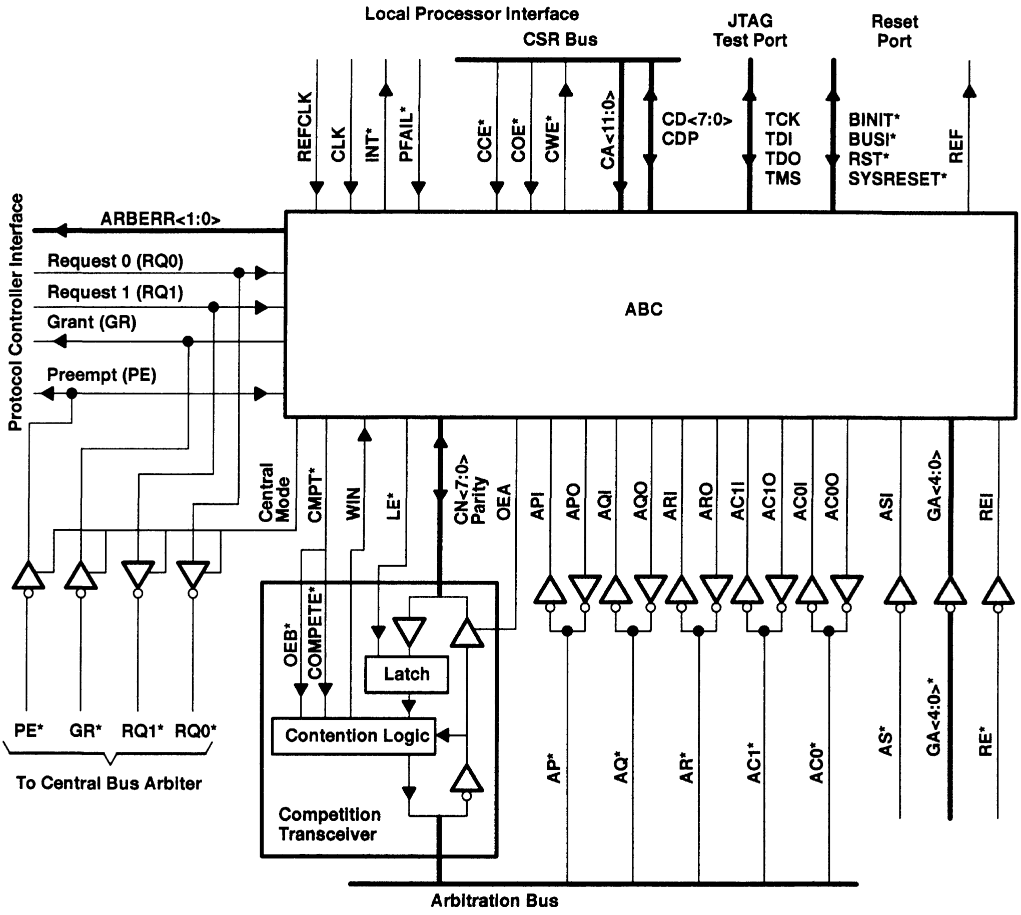


Figure 6–3. Futurebus+ Interface

6.2.1.2 Protocol Controller Interface

The ABC signal interface for distributed-mode bus acquisition has been designed to mimic the request/grant and preemption (RQ0, RQ1, GR, and PE) handshake of a central bus arbiter. Thus, an ABC can easily interface to module protocol control circuits that were designed to work with a central arbiter. This also enables a module to be designed to operate in either central or distributed mode as described in Sections 6.1.2.4 and 6.1.2.5.

Arbitration error signals are provided to allow error bits to be set in CSR registers that are not located in the ABC. These signals produce encoded values synchronous with the CLK signal and are active for one CLK cycle. ARBERR<1:0> are encoded as shown in Table 6–2. See IOC error high and low registers for more detail on how these signals may be used to set bits in standard 896.2-defined CSR registers.

Table 6–2. 1-Pass Bus-Acquisition Competition Number

ARBERR<1:0>	DEFINITION
00	No Error
01	AC0 and AC1 asserted during phase 3. This error is set regardless of which module detected the arbitration error.
10	Arbitration compare or parity error. During arbitration phase 3, the winning competition number is checked for correct parity and compared to the competition number of this device. If a parity error is detected or if the winning competition number is less than this device's competition number, then this error is flagged and AC0 is asserted.
11	Arbitration tlme-out error. During arbitration phases 2 and 4, a 1-µs timer is used to prevent bus lock up. If this time out is exceeded, then this error condition is set.

6.2.1.3 Local Processor Interface

ABC programming and control is accomplished through the CSR bus interface. The ABC functions as a CSR bus slave. Registers may be written to or read from to configure the device, set arbitration priorities, send messages, observe device status, and receive messages and interrupts. The CSR interface is synchronous with respect to CLK. For writes, data is captured on the rising edge of the first CLK when the address lines (CA<11:0>) are valid and chip enable (CCE*) and write enable (CWE*) are asserted (low). The register is actually loaded on the next rising edge of CLK. Register reads occur when chip enable (CCE*) and output enable are asserted (low) while the address lines are valid for the desired register byte. Data in dynamic registers, such as the interrupt registers, is latched to provide stable data to the CSR interface throughout a read cycle.

6.2.1.3.1 ADDRESS MAPPING

Registers are accessed through the 4K byte CSR register space associated with a Futurebus+ node. Even though this device does not require a full 4K byte register space, the entire 12-bit CSR offset for the node is decoded by the ABC and all registers are hard mapped into the 4K space. CSR core and Futurebus+-dependent CSR registers implemented in this device are mapped directly according to IEEE Standard 896.2. Registers specific to this device are directly mapped into the initial units-space area defined in IEEE Standard 896.2. Initial units-space register mapping has been coordinated with other TI Futurebus+ products and standard CSR devices so addresses do not conflict. Data output drivers are only enabled for reads of implemented registers. This allows chip enables of CSR devices to be tied low and minimizes address decoding requirements if the predefined register mapping is acceptable. The module designer may use chip enables to remap the registers into a different address space if needed.

6.2.1.3.2 SHARED REGISTERS

Only one device on the node is expected to respond for reads to any one CSR register address. Three Futurebus+-dependent CSR registers (logical module control, logical common control, and bus propagation delay) in this device have bits that are shared with other devices or are only partially implemented in this device. Typically, the device that uses the majority of the bits in the respective CSR register maintains all the bits for that register and responds to a read of that address. The ABC powers up with read capability disabled for these registers assuming that some other device will respond to the read. This eliminates read conflicts when the chip enables have been tied low as described above. Read capability can be enabled using bit three in byte two of the ABC configuration register if needed; however, the read capability of other devices containing the shared registers must be disabled to avoid contention. Writes to these shared register addresses may simply be broadcast to all devices.

6.2.1.3.3 BYTE ACCESS

Registers in the ABC are organized as 32-bit registers even though the data port (CD<7:0>) is 8 bits. Individual bytes are accessed using the least significant bits of the CSR address (CA<1:0>) according to the table below. See the ABC data sheet for detailed register descriptions.

CA<1:0>	BYTE NUMBER
00	Byte 0
01	Byte 1
10	Byte 2
11	Byte 3

6.2.1.3.4 SOFTWARE INTERFACE

Even though the Futurebus+ distributed-arbitration protocol differs slightly between operating in the central mode or distributed mode, the ABC register interface is designed for module software compatibility between the two modes. The mode of operation is transparent to software for updating arbitration priorities and sending/receiving messages. This allows the ABC to easily operate in a variety of system configurations and reduces software development time for systems that may operate in either the central or distributed mode.

6.2.1.3.5 INTERRUPTS

INT* provides a hardware indicator that an interrupt is active in the ABC. Operation of this pin is detailed in the interrupt operation section (Section 6.2.2.4) of this document. PFAIL indicates that a power-fail message has been received. This pin is a special hardware indicator provided for modules that need to react quickly to a power-fail situation.

6.2.1.3.6 CLOCKS

The CLK input provides synchronization for the CSR bus interface and arbitration error signals. REFCLK is used as a time base for the arbitration settling timer, internal delay buffers, and glitch filter logic. For more information, see the signal descriptions in the ABC data sheet chapter.

6.2.1.4 Reset Port

The ABC uses several signals for resetting the device and aligning the asynchronous handshake with other modules on the arbitration bus. These signals provide for normal power up of a module within a system as well as for live insertion of a module into an operational system. See ABC reset operations (Section 6.2.2.5) for more information.

6.2.1.5 JTAG Test Port

Boundary SCAN is provided through a JTAG test port.

6.2.2 ABC OPERATION

The ABC performs distributed arbitration in both central- and distributed-mode systems to provide the operational capability listed in Table 6–3.

Table 6–3. Distributed Arbitration in Two Modes

MODE	OPERATIONS
Distributed	Bus acquisition to determine a bus master
Distributed and central	Send and receive arbitrated messages amongst the various FB+ modules
Central	Send messages to the central bus arbiter to program module arbitration priorities

6.2.2.1 Arbitration Messages

The ABC provides the capability to send and receive messages autonomously from the data transfer portion of Futurebus+ using the distributed-arbitration protocol in both central and distributed modes. General-arbitrated messages (central mode) and distributed-arbitrated messages (distributed mode) provide a common method of system-event generation (intermodule interrupts) and the capability for sending and receiving messages defined by vendors or specific Futurebus+ profiles. Also in the central mode, central-arbitrated messages allow the ABC to program priority and geographical address information for its respective module into the system central-arbitration device. Messages sent using the arbitration bus should not be confused with the Futurebus+ message-passing protocol that requires Futurebus+ tenure and uses the primary data transfer bus.

6.2.2.1.1 GENERAL ARBITRATED MESSAGES (CENTRAL MODE) AND DISTRIBUTED ARBITRATED MESSAGES (DISTRIBUTED MODE)

Although general arbitrated messages and distributed arbitrated messages use a different protocol, they convey the same information in a central-mode or distributed-mode system, respectively. They are both sent by writing a 7-bit value (AM<6:0>) to the send-message register through the CSR interface. The ABC automatically performs the appropriate Futurebus+ distributed-arbitration protocol to broadcast the message to all other distributed-arbitration-capable modules. In the distributed mode, a two-pass arbitration is performed using the competition numbers as shown in Table 6–4.

Table 6–4. Distributed-Arbitration-Message Competition Numbers (Distributed Mode)

Bit	CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0
Pass 1	1	1	1	1	1	1	1	1
Pass 2	1	AM6	AM5	AM4	AM3	AM2	AM1	AM0

In the central mode, only one arbitration pass is required using the competition number as shown in Table 6–5.

Table 6–5. General-Arbitration-Message Competition Number (Central Mode)

Bit	CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0
Pass 1	1	AM6	AM5	AM4	AM3	AM2	AM1	AM0

The Futurebus+ protocol is transparent to software because the register interface is the same regardless of the arbitration mode. A message-sent interrupt is generated to inform software when the message has been sent. There are 128 different message values that can be sent or received; the most significant bit is always a 1. Messages are received in one of several ways depending on the value of the message as detailed in the following paragraphs. Note that all messages sent are also received by the module that sent them.

6.2.2.1.1.1 Targeted Interrupts

Futurebus+ uses message values 0x80–0x9F for its event mechanism. These 32 messages are called targeted interrupts and cause bits in the target interrupt register to be set if the corresponding targeted interrupt mask bit permits. This generates an interrupt to inform software that a targeted interrupt has been received. See targeted interrupt operation (Section 6.2.2.3).

6.2.2.1.1.2 Messages Received in FIFO

Messages that are not targeted interrupts may be stored into a four-message-deep FIFO. A local module interrupt (receive FIFO not-empty interrupt) is generated whenever the FIFO contains messages. Messages can then be read from the FIFO in the order they were received.

6.2.2.1.1.2.1 FIFO Message Masking

The message-control register provides additional control over which FIFO messages are received. The ABC screens the FIFO messages into two accepted classifications. Messages that meet the criteria for either classification are accepted; other messages are ignored. The message-control register provides a message mask-value byte and a message mask-enable byte for each of the two message classes. Message mask-enable bits enable message mask-value bits to be compared to the incoming message value. For each bit in the mask-enable register that is set, the corresponding bit in the incoming message must match the bit in the mask-value register for the message to be received. For each bit in the mask-enable register that is cleared, corresponding bits in the incoming message are not compared to the mask value and do not effect the decision to receive or reject the message. For example, if the class 1 message-mask bytes are loaded as shown below, all messages with 1110 in the upper nibble are received.

Example:
Class 1 message-mask operation
11101001 message-mask value byte
11110000 message-mask enable byte
1110xxxx messages that are stored in the FIFO   x = don't care

The second set of mask registers (for class 2 messages) operate in a similar fashion. One application of these registers might be as follows: each module in the Futurebus+ system might have class 1 programmed the same so all of the messages meeting this criteria would be globally received. Class 2 message masking could then be programmed differently on each module allowing the system designer to define certain messages specific to a particular module.

Two messages are unmaskable: 0xFE and 0xFF. The message 0xFF indicates a power fail has occurred. This message is stored in the message FIFO, the PFAIL pin is asserted, and the PFAIL interrupt is activated in the interrupt register to alert the module that a power fail is imminent. A bit in the configuration register is provided to configure all message values in the range 0x60 – 0xFF to be power-fail messages as required by profile B.

6.2.2.1.1.2.2 FIFO Overflows and Other FIFO Message Control

If the FIFO is full and a new message is received, the new message is ignored and the FIFO overflow interrupt is set. Additional message receipt control is provided by three bits in the ABC configuration register to assist with minimizing the occurrence of FIFO overflows; the ignore duplicate message bit prevents the same message from being stored in the FIFO more than once. This feature can be used when the quantity of a given message is unimportant. The stall bit causes the ABC to remain or stall the arbitration process in phase 3 if the FIFO is full and a message transfer is taking place. This prevents FIFO overflows without eliminating any messages; however, it may hang the arbitration bus until a message is read from the FIFO. The enable FIFO overflow reporting enables signaling of a full FIFO over the arbitration bus using AC0 to indicate an error. This capability cancels message receipt in phase 5 and forces the message sender to resend the message until it is successfully received by all modules.

6.2.2.1.2 CENTRAL ARBITRATED MESSAGES (CENTRAL MODE)

Central arbitrated messages are used by module distributed arbiters to program the module priority and geographical address for the two Futurebus+-defined priority levels (RQ0 and RQ1) into the system central-arbitration device. These messages require two passes of the arbitration protocol using the competition number detailed in Table 6–6. They are automatically sent by writing to a priority register in an ABC distributed arbiter operating in central mode. A message corresponding to the priority for request 1 (RQ=1) is sent if priority register 1 is written. Similarly, writing priority register 0 creates a message to modify the priority corresponding to request 0 (RQ=0). This type of message is used only by the system's central arbiter and ignored otherwise. Note that writing a priority register in central mode has the same system level effect as in distributed mode; it changes the priority which will be used for bus acquisition arbitration. In a central mode system however, the central bus arbiter needs to be programmed with the appropriate priorities. In distributed mode, the priorities in the ABC priority registers are used directly.

When a central arbitrated message has been sent, the priority-updated interrupt is generated instead of the message-sent interrupt to inform the local module software that the central arbiter has been loaded with the new priority. This also mimics the distributed-mode-priority register modification. In the distributed mode, the interrupt is generated immediately following the priority register write.

Table 6–6. Central-Mode Central-Arbitrated-Message Competition Number

Bit	CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0
Pass 1	0	PR7	PR6	PR5	PR4	PR3	PR2	PR1
Pass 2	1	PR0	RQ	GA4	GA3	GA2	GA1	GA0

6.2.2.2 Distributed-Mode Bus Acquisition

The ABC provides a module with the capability to gain mastership of Futurebus+ using the Futurebus+ distributed-arbitration protocol. Asserting ABC bus request inputs RQ0 or RQ1 initiates the arbitration process if the module has been aligned and the master enable bit has been set in the logical module-control CSR. If RQ0 is asserted, the ABC competes for use of Futurebus+ with the priority value stored in priority register 0. If RQ1 is asserted, priority register 1 is used to compete for use of Futurebus+. When the arbitration process is complete and the competition has been won by a particular module, the ABC for that module asserts its bus grant signal (GR). RQ0 and RQ1 may be cleared after GR is received when the module no longer requires Futurebus+. GR is cleared in response to RQ0 and RQ1 release, and the Futurebus+

arbitration process may be used to grant the bus to another module. Distributed-mode operation must be selected at system reset time (see Section 6.2.2.5 for reset operations) or through the logical common-control CSR central-mode bit. Master enable in the logical module-control CSR must be enabled for bus requests to be acted upon.

6.2.2.2.1 COMPETITION PRIORITY

The values stored in priority registers 0 and 1 are used to build competition numbers used in the distributed-mode bus-acquisition arbitration process. The highest priority number wins the competition and becomes the master elect. The arbitration process may require one or two passes of the arbitration protocol to resolve the identity of the highest priority competitor. If a module is competing with a request priority (PR<7:0>) of 0xFE or 0xFF, CN7 is asserted during the first pass and only one pass is required as shown in Table 6–7.

Table 6–7. 1-Pass Bus-Acquisition Competition Number

Bit	CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0
Pass 1	1	PR0	RR	GA4	GA3	GA2	GA1	GA0

If all modules are competing with priorities less than 0xFE, then CN7 is released during pass 1 and two passes are required to determine a winner as shown in Table 6–8.

Table 6–8. Distributed-Arbitration-Message Competition Numbers (Distributed Mode)

Bit	CN7	CN6	CN5	CN4	CN3	CN2	CN1	CN0
Pass 1	0	PR7	PR6	PR5	PR4	PR3	PR2	PR1
Pass 2	1	PR0	RR	GA4	GA3	GA2	GA1	GA0

For single or dual priority systems, 0xFE and 0xFF increase performance by eliminating the second arbitration pass. When modules of the same priority compete for the bus, the round-robin bit (RR) and then the module's geographical address (GA<4:0>) are used to determine the competition winner.

6.2.2.2.2 ARBITRATION PRIORITY UPDATES

The arbitration priority can be modified after RQ0 or RQ1 has been asserted using one of two methods detailed below. This capability might be used when a higher priority task arises while waiting to obtain the bus for a lower priority task, or the urgency of the task for which the bus is being requested is increasing with time. This capability allows the system designer to eliminate priority inversion (lower priority tasks taking precedence over higher priority tasks) from occurring within a given module. Increasing the priority of a task may simply cause a higher priority to be used during the next competition, or it may actually preempt the current master elect to begin a new arbitration cycle (see Section 6.2.2.2.3.1 for master-elect preemption).

6.2.2.2.2.1 Additional RTQ0/1 Assertion

One method of changing the priority is to assert the RQ0/1 signal that is not currently asserted while arbitrating for the bus. If the priority register for the new RQ0/1 assertion is higher than the priority currently being used for arbitration, then the new higher priority is used when the arbitration sequence goes through a phase where the competition number can be updated. If the priority register value for the new RQ0/1 assertion is lower, then arbitration continues at the current priority level and the new RQ0/1 has no effect. After using the bus, RQ0 and RQ1 are both cleared.

6.2.2.2.2.2 Priority Register Modification

A second method of changing the priority of an ongoing bus arbitration is by writing the priority register corresponding to the RQx line currently asserted. If the new priority is higher, it is used for competition when the arbitration sequence goes through a phase where the competition number may be updated. If the priority is lower , the new priority value is ignored until the next bus tenure is requested. If both RQ1 and RQ0 are asserted, then the new priority must be higher than both of the original priorities to have an effect.

6.2.2.2.3 PREEMPTION

Preemption is provided by Futurebus+ to allow the system designer more control over the order in which system tasks have use of Futurebus+. As with priority updates for a module, this type of control prevents priority inversion problems from occurring. The ABC implements two types of preemption: master elect and master.

6.2.2.2.3.1 Master-Elect Preemption (Deposition)

Master-elect preemption or deposition occurs when the master elect has a lower priority than another module desiring the Futurebus+. The higher priority module may depose the master elect and initialize an arbitration process to determine a new master elect. The module with the highest priority during the new arbitration process becomes the new master elect regardless of which module instigated the preemption. This feature is most useful when a master uses the bus for a long period of time and new higher priority tasks arise following selection of the master elect. The module with a higher priority can preempt a master elect via a new bus request, a priority register update, or an arbitration message. Similarly, arbitration messages may also depose a master elect. Arbitration messages have a higher priority than a master elect. Sending an arbitrated message deposes the master elect and then determination of a master elect reoccurs due to the still-pending bus requests. See arbitration messages (Section 6.2.2.1) for details on sending messages.

Using bit three in byte three of the configuration register to enable or disable master-elect preemption sourcing for a module allows system designers to tailor preemption for their system needs. When disabled, the module does not depose a master elect for bus acquisition; however, deposition induced by other modules remains unaffected. This bit does not affect messages; arbitration messages are never prevented from deposing a master elect.

6.2.2.2.3.2 Master Preemption

Master preemption informs the current bus master that another module desires to use Futurebus+. When the arbitration process appoints a module as master elect, the current master ABC compares the priority of the master elect to its own priority. The result of the compare determines if the preemption signal (PE) is asserted. Bit four of byte three in the configuration register selects between two compare functions: master-elect priority greater than or equal to or simply greater than the master's priority. PE is asserted if the compare result meets the selected compare function requirement. Assertion of the PE signal does not necessarily cause a preemption; it only informs the current master that a master elect with a priority meeting the selected compare requirement desires to use the bus. This feature allows the selective partitioning of a long bus tenure based on the priority of the waiting master elect. The compare-select control allows this to occur even within a given priority level. The PE signal for distributed-mode arbitration is analogous to the PE signal asserted by a central arbiter in central-mode systems. PE is released when grant (GR) is released.

6.2.2.2.4 FAIRNESS

The ABC implements the Futurebus+ round-robin fairness scheme to allow modules of a given priority to share bus time without starving any module. A round-robin bit (RR) is included as part of the competition number for bus acquisition. A competition number's most significant bits are comprised of the priority bits PR<7:0>. The round-robin bit is next, followed by the geographical address that is least significant. For a given priority, the round-robin bit is set when a competition is lost to a module with the same priority but with a higher geographical address. The competition winner clears its round-robin bit allowing another module to win arbitration at that priority level. The ABC maintains a separate round-robin bit for each of the two priority registers. When a priority register is written, the round robin-bit corresponding to that register is cleared. Configuration-register byte three, bit two enables or disables the round-robin capability. If it is disabled (priority only), both round-robin bits are set to zero and within a given priority, the geographical address is used to distinguish a winner. If round robin is enabled, the round-robin bit in the competition number takes priority over the geographical address.

6.2.2.2.5 BUS PARKING

Bus parking eliminates the need for bus arbitration when the previous bus master desires to use the bus again and no other modules have requested use of the bus in the mean time. RQ0 or RQ1 is asserted as for a normal bus request and the bus is granted (GR) immediately.

6.2.2.2.6 GRANT INTERRUPT

When the bus is granted to a module, the grant interrupt for that module is made available through the ABC interrupt register. The grant interrupt allows software to update priority register values for subsequent bus acquisitions as soon as the current bus acquisition is complete. This eliminates unwanted preemptions that priority register changes might cause if done prior to the bus being granted. For example, a module may have a queue of tasks, each with a different system priority. Using the grant interrupt, software can prepare to perform its next bus-acquisition arbitration during its current bus tenure. The ABC is ready to perform the next arbitration as soon as the current bus tenure is complete.

6.2.2.3 Targeted Interrupt Operation

Targeted interrupts provide an intermodule event capability for Futurebus+ systems. The ABC provides logic to manage the 32 Futurebus+ targeted interrupts defined by the Futurebus+ specification 896.2. Targeted interrupts may be generated by the current bus master over the data transfer bus or by any distributed-arbitration-capable module over the arbitration bus. 896.2 defines two CSR addresses for manipulating targeted interrupts: the interrupt target register (address 80) and the interrupt target mask register (address 84). Writing a 1 to a bit in the interrupt-target CSR sets the corresponding bit in the target-interrupt register if the corresponding target-interrupt mask bit is enabled. Writing a 0 to a bit in this register has no effect. The interrupt mask CSR (address 84) provides a bit-by-bit masking capability for the interrupts. For a target interrupt to be set, the corresponding mask bit must be set. The ABC defines an additional target-interrupt manipulation register whose address is mapped into the module's initial unit space. Writing a 1 to a bit in the target-interrupt clear register (address 3740) resets a target interrupt if it has been set. Writing a 0 has no effect on the corresponding interrupt. Note that writing 1 to set and writing 1 to clear to separate addresses allows manipulation of the interrupts without requiring a read-modify-write operation that simplifies software. The target-interrupt functional operation is shown in Figure 6–4.

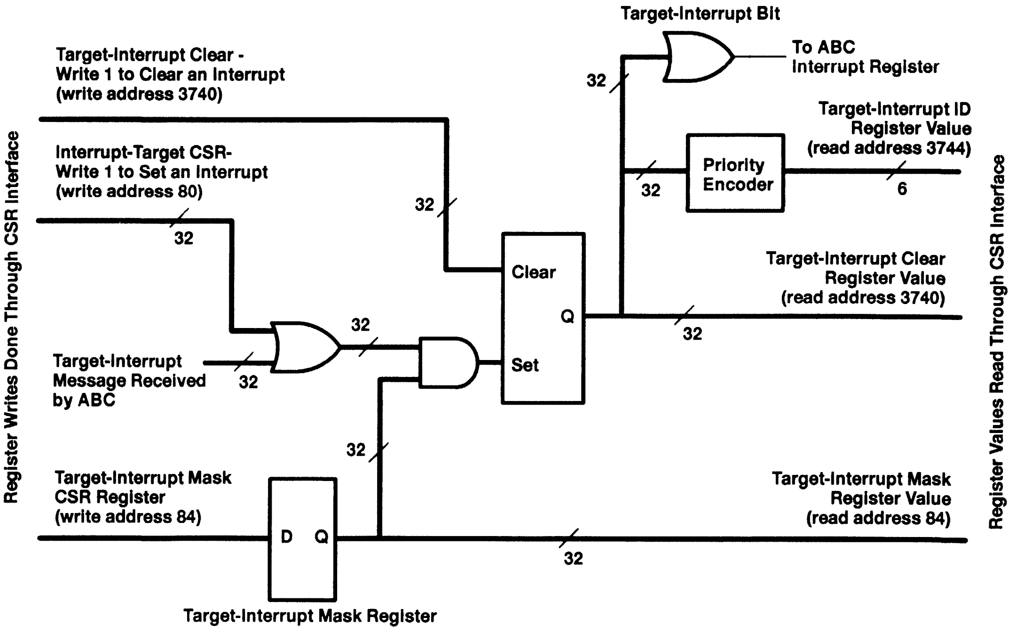

Target-Interrupt Bit
To ABC Interrupt Register
Target-Interrupt Clear - Write 1 to Clear an Interrupt (write address 3740)
Interrupt-Target CSR- Write 1 to Set an Interrupt (write address 80)
Target-Interrupt ID Register Value (read address 3744)
Priority Encoder
Clear
Q
Set
Target-Interrupt Clear Register Value (read address 3740)
Target-Interrupt Message Received by ABC
Target-Interrupt Mask CSR Register (write address 84)
Target-Interrupt Mask Register Value (read address 84)
D Q
Target-Interrupt Mask Register
32
6
Register Writes Done Through CSR Interface
Register Values Read Through CSR Interface

Figure 6–4. Target-Interrupt Functional Operation

6.2.2.3.1 DISTRIBUTED-ARBITRATION MESSAGE TARGET-INTERRUPT GENERATION

Arbitration messages may also be used to assert targeted interrupts. Futurebus+ specification 896.2 defines 32 arbitration messages (0x80–0x9F) as targeted interrupt messages. Receipt of one of these messages sets the corresponding target interrupt bit in the target-interrupt register if the target-interrupt mask bit is enabled.

6.2.2.3.2 MODULE-INTERRUPT GENERATION

All 32 target interrupts in the target-interrupt register are ORed to produce a single interrupt . This interrupt is made available to the module through the ABC interrupt pin and interrupt status register. Upon detecting that the target interrupt is active through the interrupt status register, software can interrogate the target-interrupt clear or ID registers to identify which target interrupts are active.

6.2.2.3.3 RECEIVING-TARGET INTERRUPTS

Two addresses allow access to target-interrupt information: the target-interrupt clear register (address 3740) and the target-interrupt ID register (address 3744). Reading the target-interrupt clear register provides the values of each of the target interrupts currently set. Reading the target-interrupt ID register provides a 6-bit encoded value of the most significant target interrupt currently set. This may be used for vectoring software to routines based on the targeted interrupt being serviced. Reading the interrupt-target CSR register always returns zero as specified in 896.2.

6.2.2.4 Interrupt Operation

The ABC provides an active-low interrupt pin (INT*) to alert local module processing elements that some action is required. INT* reflects the logical OR of all the interrupt conditions stored in the interrupt status register that are enabled by the interrupt-enable register. Upon receiving the interrupt signal, software may obtain the value of the interrupts currently active by reading either the interrupt status set or clear register. An encoded value of the highest priority interrupt can be determined by reading the interrupt-status ID register. Interrupts are cleared by writing a 1 to the interrupt clear register for the bit to be cleared. Similarly, interrupts may be manually set by writing a 1 to the interrupt set register for the bit to be set. See Figure 6–5.

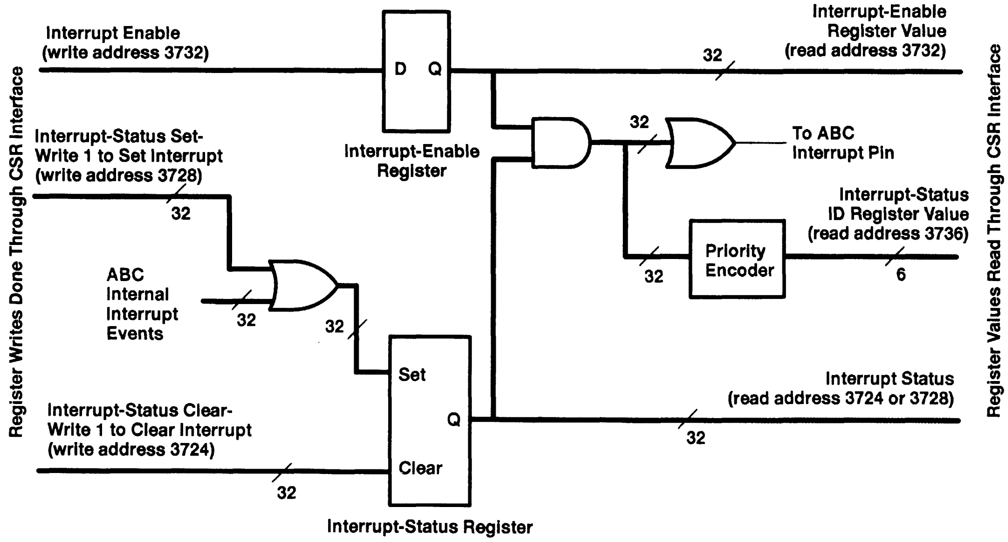


Figure 6–5. ABC Interrupt Functional Operation

6.2.2.5 Reset Operations

The ABC depends on a module power-up signal (RST*) for initially clearing the device following a power cycle. To perform Futurebus+ reset operations based on the RE* signal, the ABC depends on another device on the module (such as the IOC) to decode the Futurebus+ RE* signal. As detailed in the Futurebus+ specification P896.1, RE* takes on different meanings depending on the duration of its assertion. See also Chapter 3.

6.2.2.5.1 MODULE POWER UP – RST* ASSERTED

Upon assertion of RST*, this device resets all internal controllers, puts I/O pins in the high-impedance state, and releases all outputs except central mode. The ABC defaults to the central mode, and central mode is asserted. This makes PE an input that can be observed during system reset to determine the mode (central or distributed). Arbitration bus handshake signals all remain released until system reset, bus initialization, or alignment. This is referred to as the disabled mode because the device cannot participate in the arbitration handshake.

6.2.2.5.2 BUS ALIGNMENT

Initial assertion of RE* indicates that some module in the system wishes to align its asynchronous handshake signals with the other modules in the system. The ABC monitors RE* through the REI signal to determine when an alignment is occurring. When REI is asserted, any ongoing arbitration competition or bus tenure is allowed to complete; all other arbitration requests are left pending, and the arbitration bus remains in phase 0 until REI is released. After the bus is inactive or idle for 128 µs, the module performing the alignment asserts AR* to enter the idle arbitration phase 0 along with all other aligned modules in the system. When RE* is released, all aligned modules may participate in arbitration. If RE remains asserted, the alignment is promoted to a bus initialization. REI assertion is also reported to software via the REI interrupt bit in the interrupt status register.

Only the module asserting RE* to perform an alignment may align itself. The ABC uses the bus idle signal BUSI* to determine when it is being aligned. In response to BUSI*, the ABC asserts ARO and becomes aligned with the arbitration bus handshake.

Alignment is used following a module live insertion. The live-inserted module performs its own power-up reset and proceeds to align itself with other modules on Futurebus+ by asserting RE*. Following alignment, the ABC participates in the arbitration handshake when other modules initiate an arbitration process. Because the live-inserted module has no knowledge about the mode (central or distributed) in which the system is operating, the ABC follows the handshake to remain aligned but does not receive messages or initiate any arbitration activity. This is called the follow mode. To put the ABC into the run mode, the central-mode bit in the logical common control register must be written by the monarch or other system-knowledgeable module. Writing this register and setting the bit for distributed or central mode enables the ABC to fully participate in all arbitration activities.

6.2.2.5.3 BUS INITIALIZATION

Any modules detecting RE* asserted for 2 µs –30 µs should all perform a bus initialization. The module RE* decoder should assert BINIT* to inform the ABC that a bus initialization is occurring. The ABC resets all Futurebus+-related controllers to the idle arbitration phase 0 (APO and AQO released and ARO asserted). The CSR interface and registers remain unaffected by BINIT*. A bus initialization received after the ABC has been in the run mode resets the controllers, and the run mode is resumed when RE* and BINIT* are released. A bus initialization received after a power up puts the ABC into the follow mode similar to an alignment with BUSI* asserted as described above for the live-inserted module. A system reset normally follows a bus initialization if the system has just been powered up. If RE* remains asserted, the bus initialization is promoted to a system reset.

6.2.2.5.4 SYSTEM RESET

If RE* remains asserted beyond 30 µs, SYSRESET* is asserted by the module along with the previously asserted BINIT*; this is called a system reset. In addition to the actions taken during the bus initialization, the

CSR interface, all registers and interrupts, and arbitration requests are reset. PE is observed to automatically determine the arbitration mode. If PE = 1, then the central mode is used. If PE = 0, the distributed mode is used. A system reset leaves the ABC in the run mode meaning it is aligned, reset, and it understands which mode the system is using.

6.2.2.5.5 LOCAL MODULE RESET

For local module resets, the SYSRESET* signal may be asserted without the BINIT* signal. This resets the CSR interface including registers and interrupts. Arbitration requests not yet active are also cleared. It does not affect the arbitration bus activity. This type of reset should be used very carefully to avoid hanging partially completed arbitration requests.

6.2.2.6 Glitch Filters

The ABC contains programmable glitch filters for all signals on the arbitration bus for which a wire-ORed glitch might be observed that includes API, AQI, ARI, and REI. Only the REF signal (REI filtered) is output from the ABC for use by the module designer. All other filtered signals are used internally only. Glitch filters add delay proportional to the size of the glitch the filter is programmed to eliminate. The maximum duration assumed for a wire-ORed glitch is equal to two times the electrical length of the backplane (two times backplane end-to-end propagation delay). Glitch filter programming is done through the CSR interface using the 6-bit bus propagation delay register. This register is calibrated in terms of the backplane electrical length in the Futurebus+ specification. The ABC groups the bits in this register to provide four different glitch-filter values as shown in Table 6–9.

The maximum glitch filtered is also equal to the amount of time the incoming signal is delayed. For maximum performance, this register should be programmed to the shortest delay for which the system can operate reliably in light of the occurrence of wire-ORed glitches. This register defaults to 3F, which is the maximum delay. Glitch filters may be disabled all together using the bypass glitch filter bit in the configuration register.

Table 6–9. Glitch Filter Times for Register Values 00–3F

REGISTER VALUE RANGE (hex)	MAXIMUM BACKPLANE ELECTRICAL LENGTH (ns)	MAXIMUM GLITCH FILTERED (ns)
00 - 0F	3.5	7
10 - 1F	7.5	15
20 - 2F	11	22
30 - 3F	15	30

6.2.2.7 Error Handling

The following paragraphs describe the way the ABC checks for and responds to error situations arising from arbitration operations.

6.2.2.7.1 CSR INTERFACE PARITY CHECK/GENERATE

Data integrity can be verified for the CSR interface using parity. The enable-CSR bus-parity error-reporting bit in the configuration register enables the ABC to check parity during register writes. If a parity error is detected, the errored data is not written. CSR parity errors are reported through the CSR par err interrupt bit. If parity checking is disabled, the parity bit CDP is ignored during writes. Parity is always generated for CSR interface read operations.

6.2.2.7.2 ARBITRATION BUS ERRORS

Errors detected on the arbitration bus are reported to the local through interrupts and the ARBERR signals described previously. Arbitration condition signals AC1O and AC0O are used to signal other FB+ modules about error conditions.

6.2.2.7.2.1 Competition Number Errors

During arbitration phase 3, the winning competition number is checked for odd parity and losing competitors compare their competition number to the winning number to verify it has a greater priority. If an error is detected, the arbitration process is cancelled by assertion of AC1* and AC0* and the errors are reported through the interrupt register. The cmparerr interrupt bit reports that either a parity or compare error has been detected by this device during arbitration phase 3. The arberr interrupt indicates that some module has asserted AC0*; this is detected during arbitration phase 5. Competition compare-error checking can be enabled or disabled using the disable competition-compare error bit in the configuration register. Parity reporting can be enabled by the parity report-enable bit in the logical common control register. If this bit is set, AC0O and AC1O signals are asserted in phase 3 when a parity error is detected and the error is reported locally through the cmparerr interrupt and ARBERR signals. If this bit is cleared, AC0O and AC1O are not asserted, however, the error is still reported locally.

6.2.2.7.2.2 1-µs Timer

A 1-µs timer is required by Futurebus+ in arbitration phases 2 and 4 to prevent the bus from getting stuck and locking the system up. If the 1-µs timer expires in phase 2, this indicates that there is no module that has determined it is the winner as a result of the competition phase. A module detecting a 1-µs time out in phase 2 moves the bus on to phase 3 and asserts AC1O and AC0O to indicate to other modules that an error has occurred. The PHS24TO interrupt is generated, and the error is also reported by setting the ARBERR<1:0> signals to 11.

If the 1-µs time out occurs in phase 4, the error is only reported locally via the PHS24TO interrupt and the ARBERR signals. The module detecting the error then moves the arbitration to phase 5.

6.2.2.7.2.3 Dead-Man Timer

The ABC provides a timer for Futurebus+ phase 0 and phase 1 that are not covered by the 1-µs timer. The dead-man timer runs during phase 0 if it is the second pass of an arbitration cycle. It also runs during phase 1. The dead-man timer generates an interrupt upon expiration. This is a signal to module software that something is wrong with the arbitration system. The dead-man timer can be disabled using a bit in the configuration register.

6.2.2.8 Arbitration Settling Times

During distributed-arbitration phase 2, contention logic in a competition transceiver works to resolve the highest priority competitor. The WIN signal during this time period is invalid until this process is complete. The amount of time required for the competition to settle and a winner to be resolved is a function of the electrical length of the backplane and the competition numbers involved in any particular competition. Any given competition number contains a certain number of 1-to-0 transitions such as the one shown below which contains 2:

```
CN7      CN0
 00101001
    V V
    1 2
```

The number of 1-to-0 transitions determines the number of times a module's number must propagate down the backplane and back before that module can determine whether it is a winner. A competition number with no 1-to-0 transitions such as 00001111 knows whether it is the winner as soon as its number has propagated down the bus and back once. Thus, the total number of bus iterations equals the number of 1-to-0 transitions plus 1 initial iteration.

The ABC provides a programmable timer to wait for the competition to settle before observing the WIN signal to determine whether the competition has been won. The timer uses the combination of a synchronous counter and a variable series of delay elements to produce a wide variety of settling times. Two registers are provided to control this timer: the competition settling time CSR (defined in 896.2) and the optional competition settling time register.

6.2.2.8.1 COMPETITION SETTLING-TIME REGISTER

The competition settling-time register is a standard CSR defined in the Futurebus+ 896.2 specification that provides twelve bits to store an arbitration settling time value in increments of seconds. The value in this register is used whenever the value for the corresponding competition number in the optional competition settling-timer register is zero. This register defaults to 0x600. Only the most significant eight bits are decoded and used by the ABC timer circuit.

6.2.2.8.2 OPTIONAL COMPETITION SETTLING-TIME REGISTER

To improve performance, the ABC divides competition numbers into four groups (group 0–group 3) based on the number of 1-to-0 transitions contained in the number. Once a group is decoded, the corresponding byte in the optional competition settling-time register is used to determine the competition settling time. If the byte value for a group is zero, then the standard competition settling-time register value is used for competition numbers decoded to that group. The competition timer is loaded at the beginning of a competition with the 8-bit value corresponding to the group decoded from the competition number. This dynamic determination of the arbitration settling time allows the arbitration bus to use the shortest arbitration settling times for the competition numbers being used during any given competition. This register should not be changed while a request for arbitration is pending for that device. The table below shows the competition number group decoding and an equation to determine what the maximum settling time should be for that competition group.

Table 6–10. Competition Number Group Decoding

GROUP	COMPETITION NUMBER 1-to-0 TRANSITIONS	OPTIONAL COMPETITION SETTLING-TIME REGISTER	WORST-CASE SETTLING TIME†	EXAMPLE TIME	EXAMPLE REGISTER VALUE‡
0	0 or 1	Byte 0	$T_A = 4\ t_{pd} + 2\ t_{ext} + 1\ t_{int} + t_{win}$	52 ns	14 (0x0E)
1	2	Byte 1	$T_A = 6\ t_{pd} + 3\ t_{ext} + 2\ t_{int} + t_{win}$	78 ns	21 (0x15)
2	3	Byte 2	$T_A = 8\ t_{pd} + 4\ t_{ext} + 3\ t_{int} + t_{win}$	104 ns	28 (0x1C)
3	4	Byte 3	$T_A = 10\ t_{pd} + 5\ t_{ext} + 4\ t_{int} + t_{win}$	130 ns	35 (0x23)

† This value must be divided by 3.73 ns and rounded off to the nearest bit before programming the register.

‡ The optional arbitration settling-time register is programmed to 0x0E15–0x1C23 for this example case. For group 0 arbitration numbers, over 80 ns of competition delay is saved in this example. With careful use of competition numbers, significant performance enhancements are possible.

Where:

t_{pd} = end-to-end signal propagation delay along the bus (each of the coefficients associated with t_{pd} assumes that each competition iteration takes 2 × t_{pd} to complete; this is unrealistically pessimistic. Roughly 70% of the coefficient value should be sufficient.)

t_{ext} = maximum delay of all possible external competitors from observing a higher arbitration number to the release of its own arbitration number. (If only Texas Instruments contest transceivers are used throughout the system, this corresponds to the timing parameter for the FB2032 from B to Bn–1, which is 9 ns.)

t_{int} = delay within the local module from detecting a 0 (high) on bit N to reasserting bit N–1. (This corresponds to the timing parameter for the FB2032 from B to Bn–1, which is 9 ns).

t_{win} = delay from observing a match on the arbitration bus to asserting the win signal. (This corresponds to the timing parameter for the FB2032 from B to WIN, which is 8.5 ns).

The above approach works if the REFCLK is 30 MHz or 40 MHz and the 30M/40M* bit is respectively set to 1 or 0. However, if a 30- or 40-MHz clock is not used, the mechanics of the arbitration settling time design must be taken into consideration. As shown in Table 6–11, this design counts a certain number of REFCLKs for a particular combination of register value and 30M/40M*-bit value. Similarly for Table 6–12, this design allows a certain number of delay units to transpire for a particular combination of register value and 30M/40M*-bit value.

Table 6–11. Number of Clocks (REFCLKs) for the Arbitration Settling Time

REGISTER BYTE VALUE	NUMBER OF REFCLKS	
(HEX)	30 M/40 M* Bit = 0	30 M/40 M* Bit = 1
00 – 10	0	0
11 – 17	2	1
18 – 1F	3	2
20 – 2F	5	3
30 – 3F	7	5
40 – 4F	9	7
50 – 5F	12	9
60 – 6F	14	11
70 – 7F	17	13
80 – 8F	19	15
90 – 9F	21	16
A0 – AF	24	18
B0 – BF	26	20
C0 – CF	28	22
D0 – DF	31	23
E0 – EF	33	25
F0 – FF	63	63

The relationship between a byte in the optional arbitration settling-times register (see Table 6–10) and the resulting arbitration settling time is given by the following equation:

$$\text{arbitration settling time} = \text{REFCLK period}\left(\text{number of REFCLKs} + \frac{\text{number of delay units}}{10}\right)$$

The number of REFCLKs and the number of delay units for a particular byte in the optional arbitration settling-times register is given by Tables 6–11 and 6–12, respectively. For example, with a 35-MHz REFCLK, the 30M/40M*-bit set to 1, and a register byte value of 12; the number of REFCLKs = 1 (from Table 6–11), the REFCLK period = 1/35 MHz, and the number of delay units = 8 (from Table 6–12). Therefore, the arbitration settling time is given by the following equation:

$$\text{arbitration settling time} = \frac{1}{35\ \text{MHz}}\left(1 + \frac{8}{10}\right) = 51.4\ \text{ns}$$

Table 6–12. Number of Delay Units for the Arbitration Settling Time

REGISTER BYTE VALUE	# OF DELAY UNITS	
(HEX)	30 M/40 M* Bit = 0	30 M/40 M* Bit = 1
00	2	2
01	2	2
02	5	5
03	5	5
04	8	8
05	8	8
06	12	12
07	12	12
08	15	15
09	15	15
0A	18	18
0B	18	18
0C	22	22
0D	22	22
0E	26	26
0F	26	26
10	26	26
11	2	5
12	5	8
13	5	8
14	8	12
15	8	12
16	12	15
17	12	15
18	2	5
19	2	5
1A	5	8
1B	5	8
1C	8	12
1D	8	12
1E	12	15
1F	12	15

REGISTER BYTE VALUE†	# OF DELAY UNITS	
(HEX)	30 M/40 M* Bit = 0	30 M/40 M* Bit = 1
y0	2	2
y1	2	2
y2	5	5
y3	5	5
y4	8	8
y5	8	8
y6	12	12
y7	12	12
y8	15	15
y9	15	15
yA	18	18
yB	18	18
yC	22	22
yD	22	22
yE	26	26
yF	26	26

† y = 2 through F

Chapter 7

Commercial Controller Data Sheets

SLLS168A – JANUARY 1992 – REVISED MARCH 1994

- **Provides Control Logic Necessary to Operate a Data Path Unit (TFB2022A) on Futurebus+**
- **Parallel-Protocol Support Is Fully Compliant to Futurebus+ Standard (IEEE Std 896.1–1991)**
- **Interfaces Easily to a Variety of Popular Microprocessors Such as SPARC™, R4000, 680x0, 88xxx, 80x86, and Alpha AXP™**
- **Provides Full Support for Futurebus+ Cache Commands (for Memory or I/O Modules in Shared-Memory Systems)**
- **Capable of Handling a Single Outstanding Split Transaction**
- **Parallel-Protocol-Related CSR Locations Are Provided on Chip**
- **Offers Autonomous Control for Futurebus+ and Host-Module Reads and Writes**

description

The TFB2002B I/O controller (IOC) is a member of the Texas Instruments Futurebus+ (FB+) chip set. This chip set provides a highly integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities. The Futurebus+ chip set is capable of supporting 32- or 64-bit data widths in any combination on both the host-bus interface (HIF) and Futurebus+. The address width is programmable to be 32 bits or 36 bits (with either data width).

The TFB2002B contains the control logic necessary to translate Futurebus+ transactions into host bus transactions and vice versa. It contains a high-speed Futurebus+ handshake controller, a synchronous host bus controller, and reset-type-determination logic.

When combined with a TFB2022A Futurebus+ data path unit (DPU), the TFB2002B provides a complete 64-bit-wide interface to the Futurebus+. The TFB2002B provides the necessary control logic for the data path unit to provide a complete interface to the Futurebus+ for a Profile-B-compliant module. It may also be used on I/O or memory modules in a cache-coherent system.

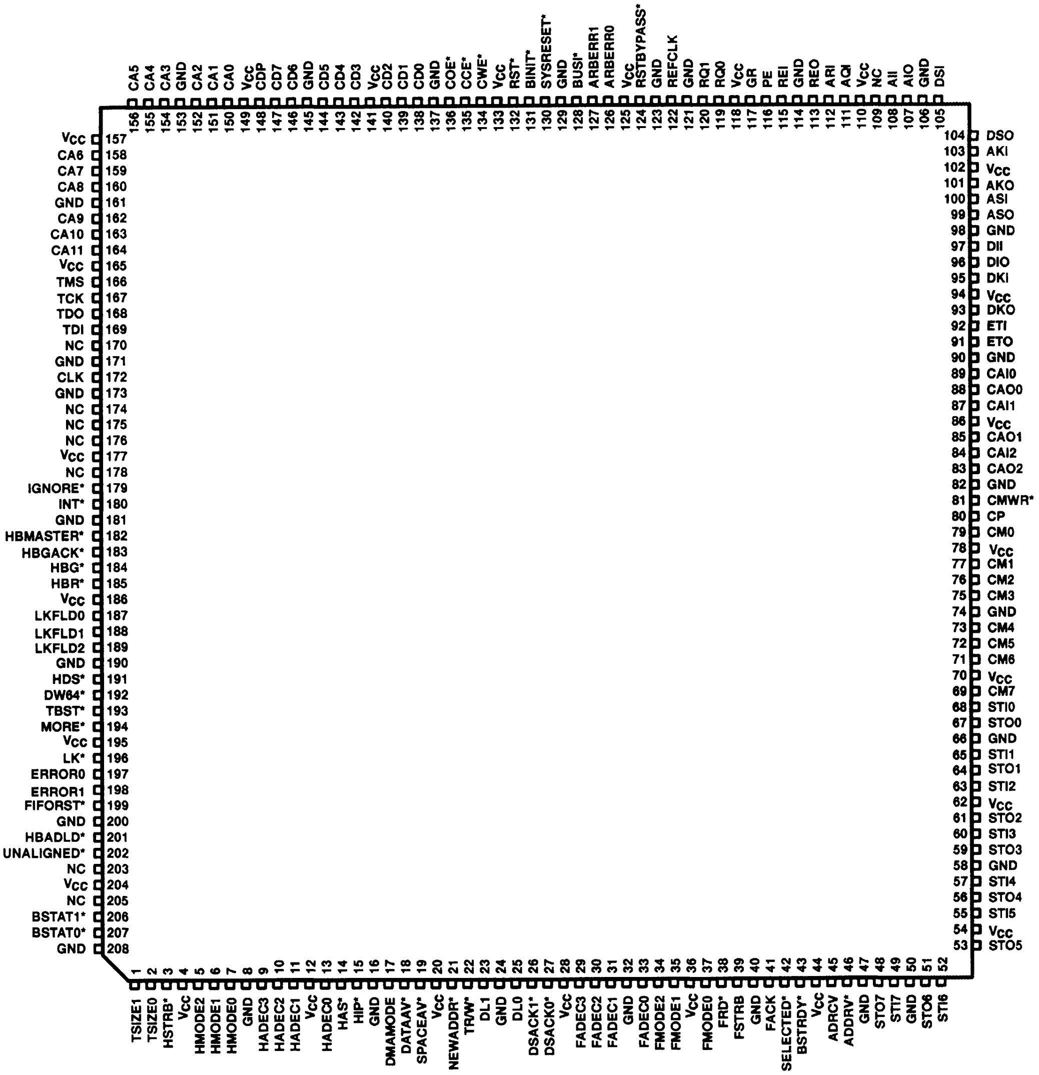
The TFB2002B is offered in a 208-pin plastic quad flat package (PPM). The TFB2002B is characterized for operation over the commercial temperature range of 0°C to 70°C.

NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896. 1–1991), an active-low signal is denoted herein by use of the trailing asterisk (*) on the signal name.

SPARC is a trademark of Sun Microsystems, Inc.
Alpha AXP is a trademark of Digital Equipment Corporation.

terminal assignments

PPM PACKAGE
(TOP VIEW)



NC – No internal connection

Terminal Functions

host interface

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
BSTAT<1:0>*	206 207	I/O	Host interface	Host interface status: HH Normal HL Reserved LH Bus error LL Backoff/retry
BSTRDY*	43	I/O	Host interface	Burst ready
CLK	172	I	Host interface	Clock input. This is the processor clock for synchronous transactions on the host side. Up to 25 MHz is recommended.
DL<1:0>	23, 25	I/O	Host interface	Host interface data length: LL 64 bytes LH 32 bytes HL 16 bytes HH 8 bytes
DSACK<1:0>*	26 27	I/O	Host interface	Data acknowledge: Single mode (TBST* = high): LL Complete cycle, data bus port 32 LH Reserved HL Reserved HH Insert wait state Burst mode (TBST* = low): LL Low-speed, 32-bit burst capable LH High-speed, 32-bit burst capable HL Low-speed, 64-bit burst capable HH High-speed, 64-bit burst capable
DW64*	192	I/O	Host interface	Host interface data width of 64 (burst mode only)
HAS*	14	I/O	Host interface	Host interface address strobe
HBG*	184	I	Host interface	Host interface grant input
HBGACK*	183	I/O (open collector)	Host interface	Host interface grant acknowledge
HBR*	185	O (open collector)	Host interface	Host interface request output
HDS*	191	I/O	Host interface	Host interface data strobe
HIP*	15	I/O	Host interface	Host interface transaction in progress
IGNORE*	179	I	Host interface	Ignore the current host transaction input. This signal is supplied by the host memory decoder when an access to private memory occurs. This signal is optional and should be tied high if it is not used.
INT*	180	O (open collector)	Host interface	Host interrupt output. When an enabled interrupt condition occurs, this signal is driven low. Interrupts are cleared by writing a one to the appropriate bit in the interrupt register. The interrupt goes high during the write cycle to the interrupt register even if another interrupt is pending. Also used from FB+ CM<2:0> lines to a mastered HIF locked operation. These pins are used as inputs when the IOC is a host interface slave/FB+ master.
LK*	196	I/O	Host interface	Host cycle is locked (indivisible)
LKFLD0, LKFLD1, LKFLD2	187 188 189	I/O	Host interface	Locked-command bits passed from the host interface to FB+ or from FB+ to the host interface via the CM<2:0> lines during a mastered FB+ data phase in a locked operation. Also used from FB+ CM<2:0> lines to a mastered HIF locked operation. These pins are used as inputs when the IOC is a host interface slave/FB+ master.

Terminal Functions

host interface (continued)

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
MORE*	194	I	Host interface	Host cycle is part of a longer transaction input. This signal is used in DMA writes to indicate that this transaction should be included in the same tenure with the next host interface transaction. It is used in reads to indicate that MR* should be asserted during the Futurebus+ transaction.
TBST*	193	I/O	Host interface	Host transaction burst request
TR/W*	22	I/O	Host interface	Host interface read or write
TSIZE<1:0>	1, 2	I	Host interface	Host interface transaction size input: LL Word (32 bits or greater) LH Byte (8 bits) HL Half word (16 bits) HH Three bytes (24 bits)

other module interface signals

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
ARBERR<1:0>	127, 126	I	Arbiter	Arbitration error input: LL No error LH ACO and AC1 asserted during phase 3 HL Arbitration comparison error HH Arbitration time-out error (phase 2 or 4)
REFCLK	122	I	Module	Clock input. A 25-MHz, 50% ± 5% duty-cycle signal is recommended; any frequency between 20 MHz and 40 MHz and duty cycle of 50% ± 5% can be tolerated.

CSR bus

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
CA<11:0>	164, 163, 162, 160, 159, 158, 156, 155, 154, 152, 151, 150	I	CSR bus	CSR bus address input
CD<7:0>	147, 146, 144, 143, 142, 140, 139, 138	I/O	CSR bus	CSR bus data input
CDP	148	I/O	CSR bus	CSR bus data odd parity
CCE*	135	I	CSR bus	CSR bus chip enable input
COE*	136	I	CSR bus	CSR bus output enable input
CWE*	134	I	CSR bus	CSR bus write enable input

Terminal Functions

interface to TFB2022A

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
DATAAV*	18	I	TFB2022A DPU	Data available in FIFO input. In compelled mode, this signal indicates if any data is in the FIFO. In packet or burst mode, this signal indicates if a packet or burst data of length encoded on the Futurebus packet size or the DL<1:0> lines is available.
DMAMODE	17	O	TFB2022A DPU	DMA operation is occurring output. Turns off critical word first on the TFB2022A.
ERROR<1:0>	198, 197	I	TFB2022A DPU	Futurebus+ error indicator input: LL No error LH Futurebus+ parity error HL Packet longitudinal parity error HH Host bus parity error
FADEC<3:0>	29, 30 31, 33	I	TFB2022A DPU	Futurebus+ address decode input: LLLL Unselected LLLH Host memory LLHL Host extended unit space LLHH Host CSR LHLL Broadcast mailbox LHLH Reserved LHHL Reserved LHHH Reserved HLLL Mailbox address HLLH Packet-mode-capable memory address HLHL Reserved HLHH Reserved HHLL Reserved HHLH DPU CSR HHHL Broadcast CSR (non-DPU) HHHH Broadcast CSR (DPU)
FIFORST*	199	O	TFB2022A DPU	FIFORST* output. FIFORST* resets the FIFO pointers.
FACK	41	I	TFB2022A DPU	Futurebus+ acknowledge input. FACK indicates Futurebus+ event is complete
FMODE<2:0>	34, 35, 37	O	TFB2022A DPU	Futurebus+ mode output. Indicates to the TFB2022A what action is to be taken in the Futurebus+ interface: LLL Compelled-mode Futurebus+ LLH Packet-mode Futurebus+ LHL Partial transfer LHH Disconnect data for master write HLL Reserved HLH Reserved HHL Disconnect data for split requestor HHH Reserved
FRD*	38	O	TFB2022A DPU	Futurebus+ read/write indicator output: L = read from Futurebus+ to FIFO; H = write from FIFO to Futurebus+
FSTRB	39	O	TFB2022A DPU	Futurebus+ strobe. FSTRB performs next Futurebus+ event

Terminal Functions

interface to TFB2022A (continued)

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
HADEC<3:0>	9, 10, 11, 13	I	TFB2022A DPU	Host address decode input. Address decoding for the host interface address: Slave encoding: / Master encoding: LLLL Unselected / LLLL Unselected LLLH Host memory / LLLH Memory address compelled LLHL Host extended unit space / LLHL Maximum capable burst or extended unit space LLHH Host CSR / LLHH Memory address 64-byte burst LHLL Broadcast mailbox / LHLL 32-byte-memory-address capable LHLH Reserved / LHLH 16-byte-memory-address capable LHHL Split response hit / LHHL 8-byte-memory-address capable LHHH Futurebus+ CSR address / LHHH Reserved HLLL Reserved / HLLL Reserved HLLH Broadcast CSR address / HLLH Reserved HLHL Reserved / HLHL Reserved HLHH Reserved / HLHH Reserved HHLL Reserved / HHLL Reserved HHLH DPU CSR / HHLH Reserved HHHL Reserved / HHHL Reserved HHHH Reserved / HHHH Reserved
HBADLD*	201	O	TFB2022A DPU	Host address load output. Futurebus+ has been granted for the requested transaction output.
HBMASTER*	182	O	TFB2022A DPU	Host master output. This device is mastering the host bus transaction.
HMODE<2:0>	5, 6, 7	O	TFB2022A DPU	Host mode output. Indicates to the TFB2022A what action is to be taken in the host interface: LLL Reserved LLH Between FIFO and host interface (single transfer), between TFB2022A resident CSR and host interface, or between TFB2022A resident CSR and FIFO LHL From FIFO to TFB2022A resident CSR. LHH Reserved HLL Reserved HLH Between FIFO and host interface (burst mode) HHL Reserved HHH Reserved
HSTRB*	3	O	TFB2022A DPU	Host strobe output. HSTRB* performs next host interface request as indicated in HMODE<2:0>.
NEWADDR*	21	O	TFB2022A DPU	New address output. NEWADDR* increments address in the TFB2022A address register.
SELECTED*	42	O	TFB2022A DPU	Module selected output. Futurebus+ transaction uses this module. The DPU is used as a slave of the Futurebus+ transaction
SPACEAV*	19	I	TFB2022A DPU	Space available in FIFO input. In compelled mode, this signal indicates that space is available in the FIFO for another transfer. In packet or burst mode, this signal indicates that space is available in the FIFO for another packet or burst
UNALIGNED*	202	I	TFB2022A DPU	FB+ slave partial unaligned operation input

Terminal Functions

JTAG test port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
TCK	167	I	Module	JTAG test clock input
TDI	169	I	Module	JTAG test data input
TDO	168	O	Module	JTAG test data output
TMS	166	I	Module	JTAG test-mode select input

reset port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
AQI	111	I		AQI is used to determine if the arbitration bus has been idle for 1 µs. If the arbitration bus is not implemented, this signal should be tied low.
ARI	112	I		ARI is used to determine if the arbitration bus has been idle for 1 µs. If the arbitration bus is not implemented, this signal should be tied high.
BINIT*	131	O	Module	BINIT* is an open-collector signal indicating that a bus interface reset is required.
BUSI*	128	O		Bus has been idle for longer than 1 µs, and REO is asserted.
REI	115	I		Futurebus+ reset input
REO	113	O		Futurebus+ reset output
RST*	132	I	Module	Module power-up reset input. RST* resets all logic; output signals go to their inactive states, and 3-state outputs and bidirectional signals take on the high-impedance state.
SYSRESET*	130	O	Module	System reset required output. SYSRESET* is an open-collector signal indicating that a system reset is required.
RSTBYPASS*	124	I	Module	Bypass auto alignment after power up input

Terminal Functions

Futurebus+ interface

TERMINAL NAME	TERMINAL NO.	I/O	FROM/TO	DESCRIPTION
ADRCV	45	O	Futurebus+	Transceiver receiver enable output
ADDRV*	46	O	Futurebus+	Transceiver driver enable output
ASI, AKI, AII	100, 103, 108	I	Futurebus+	Futurebus+ address synchronization input signals: address strobe (ASI), address acknowledge (AKI), address acknowledge inverse (AII)
ASO, AKO, AIO	99, 101, 107	O	Futurebus+	Futurebus+ address synchronization output signals: address strobe (ASO), address acknowledge (AKO), address acknowledge inverse (AIO)
CAI<2:0>	84, 87, 89	I	Futurebus+	Futurebus+ capability input bits
CAO<2:0>	83, 85, 88	O	Futurebus+	Futurebus+ capability output bits
CM<7:0>, CP	69, 71, 72, 73, 75, 76, 77, 79, 80	I/O	Futurebus+	Futurebus+ command bits and parity
CMWR*	81	O	Futurebus+	Transceiver control for command: H = read, L = write
DSI, DKI, DII	105, 95, 97	I	Futurebus+	Futurebus+ data-path-synchronization input signals: data strobe (DSI), data acknowledge (DKI), data acknowledge inverse (DII)
DSO, DKO, DIO	104, 93, 96	O	Futurebus+	Futurebus+ data-path-synchronization output signals: data strobe (DSO), data acknowledge (DKO), data acknowledge inverse (DIO)
ETI	92	I	Futurebus+	Futurebus+ end-of-tenure input
ETO	91	O	Futurebus+	Futurebus+ end-of-tenure output
GR	117	I	Futurebus+	Futurebus+ mastership has been granted (bus tenure may begin when ETI is released).
PE	116	I	Futurebus+	Futurebus+ preemption has occurred.
RQ<1:0>	120, 119	O	Futurebus+	Futurebus+ is requested at level 1 or level 0. RQO is used for DMA operations; RQ1 is used for all other operations.
STI<7:0>	49, 52, 55, 57, 60, 63, 65, 68	I	Futurebus+	Futurebus+ status inputs
STO<7:0>	48, 51, 53, 56, 59, 61, 64, 67	O	Futurebus+	Futurebus+ status outputs

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage range, V_{CC} (see Note 1)	–0.5 V to 7 V
Input voltage range, V_I (at any input)	–0.5 V to 7 V
Output voltage range, V_O	–0.5 V to 7 V
Continuous total power dissipation	See Dissipation Rating Table
Power dissipation	500 mW
Operating free-air temperature range, T_A	0°C to 70°C
Storage temperature range	–65°C to 150°C
Case temperature for 10 seconds	260°C

NOTE 1: All voltage values are with respect to GND.

DISSIPATION RATING TABLE

PACKAGE	$T_A \le 25°C$ POWER RATING	DERATING FACTOR ABOVE $T_A = 25°C$	$T_A = 70°C$ POWER RATING
PPM	3175 mW	25.4 mW/°C	2032 mW

recommended operating conditions

	MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.75	5	5.25	V
High-level input voltage, V_{IH}	2		V_{CC}	V
Low-level input voltage, V_{IL}	–0.5		0.8	V
Operating free-air temperature range, T_A	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1\ \mu A$, $C_L = 7.4$ pF		1.3		V
V_{IT+}	Positive-going input threshold voltage	IPI09LK			1.6		V
V_{IT-}	Negative-going input threshold voltage				1.2		V
V_{OL}	Low-level output voltage (open drain)	OPI82LK	$I_{OL} = 8$ mA			0.5	V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 4$ mA			0.5	V
V_{OH}	High-level output voltage	OPI83LK	$I_{OH} = -8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 8$ mA			0.5	V
V_{OH}	High-level output voltage	OPIH3LK	$I_{OH} = -12$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 12$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ83LK	$I_{OH} = -8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 8$ mA			0.5	V

macros

Table 1 lists the internal and external buffer macros used in the TFB2002B design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

Table 1. TFB2002B (IOC) Pin Names and Macro Numbers

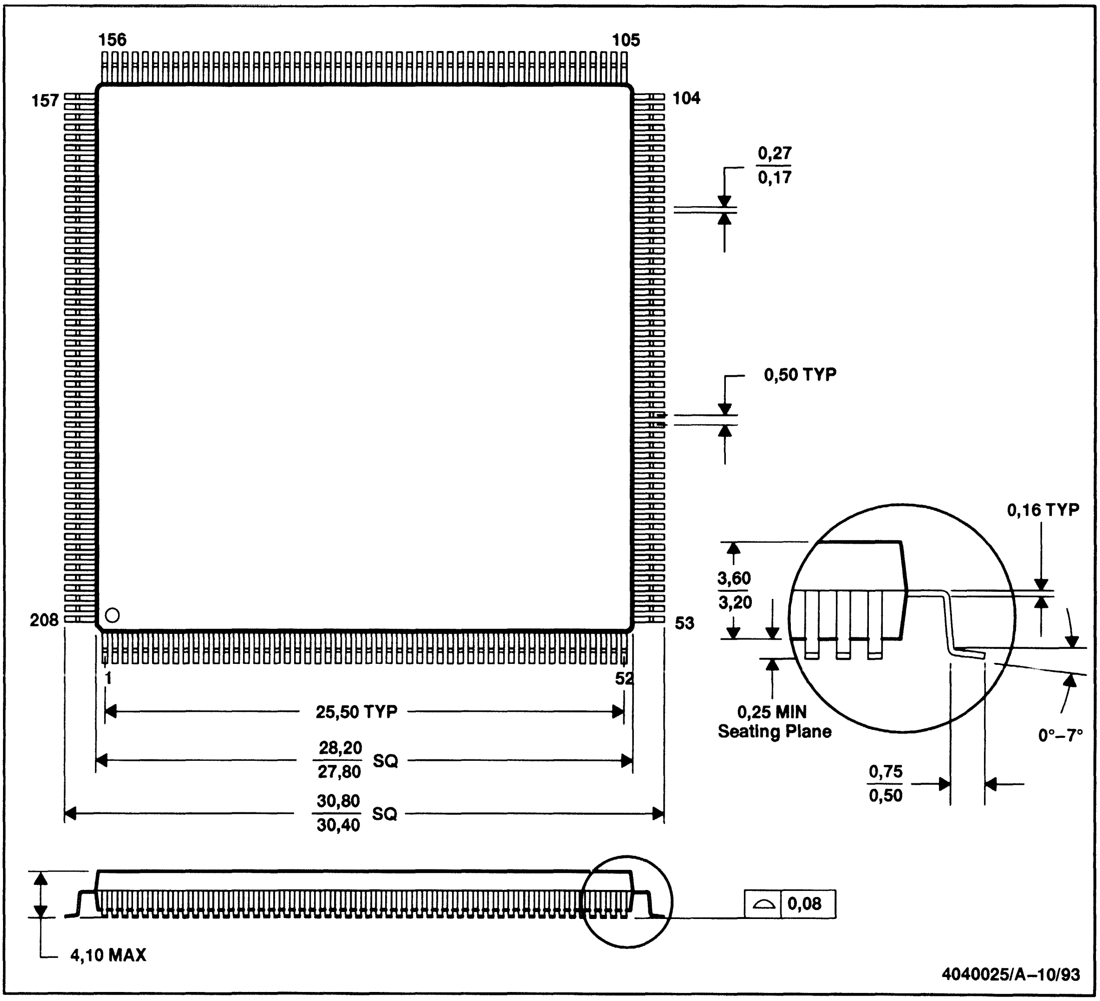
PIN NAME	INPUT MACRO	OUTPUT MACRO
ADDRV*		OPIH3LK
ADRCV		OPIH3LK
AII	IPI04LK	
AIO		OPI43LK
AKI	IPI04LK	
AKO		OPI43LK
AQI	IPI04LK	
ARBERR<1:0>	IPI04LK	
ARI	IPI04LK	
ASI	IPI04LK	
ASO		OPI43LK
BINIT*		OPI43LK
BSTAT<1:0>*	IPI04LK	OPIH3LK
BSTRDY*	IPI04LK	OPIH3LK
BUSI*		OPI43LK
CA<11:0>	IPI04LK	
CAI<2:0>	IPI04LK	
CAO<2:0>		OPI43LK
CCE*	IPI04LK	
CD<7:0>	IPI04LK	OPJ83LK
CDP	IPI04LK	OPJ83LK
CLK	IPI04LK	
CM<7:0>	IPI04LK	OPI43LK
CMWR*		OPI43LK
COE*	IPI04LK	
CP	IPI04LK	OPI43LK
CWE*	IPI04LK	
DATAAV*	IPI04LK	

Table 1. TFB2002B (IOC) Pin Names and Macro Numbers (continued)

PIN NAME	INPUT MACRO	OUTPUT MACRO
DII	IPI04LK	
DIO		OPI43LK
DKI	IPI04LK	
DKO		OPI43LK
DL<1:0>	IPI04LK	OPIH3LK
DMAMODE		OPI43LK
DSACK<1:0>*	IPI04LK	OPIH3LK
DSI	IPI04LK	
DSO		OPI43LK
DW64*	IPI04LK	OPIH3LK
ERROR<1:0>	IPI04LK	
ETI	IPI04LK	
ETO		OPI43LK
FACK	IPI04LK	
FADEC<3:0>	IPI04LK	
FIFORST*		OPI43LK
FMODE<2:0>		OPI43LK
FRD*		OPI43LK
FSTRB		OPI43LK
GR	IPI04LK	
HADEC<3:0>	IPI04LK	
HAS*	IPI04LK	OPIH3LK
HBADLD*		OPI43LK
HBG*	IPI04LK	
HBGACK*	IPI04LK	OPIH3LK
HBMASTER*		OPI43LK
HBR*		OPI82LK
HDS*	IPI04LK	OPIH3LK
HIP*	IPI04LK	OPIH3LK
HMODE<2:0>		OPI43LK
HSTRB*		OPI43LK
IGNORE*	IPI04LK	
INT*		OPI82LK
LK*	IPI04LK	OPIH3LK
LKFLD0, 1, 2	IPI04LK	OPI43LK
MORE*	IPI04LK	
NEWADDR*		OPI43LK
PE	IPI04LK	
REFCLK	IPI04LK	
REI	IPI04LK	
REO		OPI43LK
RQ<1:0>		OPI43LK
RST*	IPI09LK	
RSTBYPASS*	IPI04LK	
SELECTED*		OPI43LK
SPACEAV*	IPI04LK	
STI<7:0>	IPI04LK	
STO<7:0>		OPI43LK
SYSRESET*		OPI83LK
TBST*	IPI04LK	OPIH3LK
TCK	IPI04LK	
TDI	IPI04LK	
TDO		OPI43LK
TMS	IPI04LK	
TR/W*	IPI04LK	OPIH3LK
TSIZE<1:0>	IPI04LK	
UNALIGNED*	IPI04LK	

MECHANICAL DATA

PPM/S-PQFP-G208 **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-143.

TFB2010
FUTUREBUS+ ARBITRATION BUS CONTROLLER

SLLS125A – OCTOBER 1990 – REVISED NOVEMBER 1993

- **Supports Distributed Arbitration for Futurebus+ Master Selection**
- **Supports Arbitrated Messages in Distributed and Central Modes**
- **Enables Use of a Common Hardware and Software Interface for Both Distributed and Central Modes**
- **Requires No Hardware Modifications for Changing Between Distributed and Central Modes**
- **Provides a CSR Bus Interface for Easy Integration into the Futurebus+ CSR Address Space**
- **Has Two Bus Request Lines That Each May Be Assigned Any One of 256 Priority Levels**
- **Supports Round-Robin Fairness Arbitration Within Two Separate Priority Levels to Avoid Starvation of Any Single Module**
- **Supports Distributed-Mode Bus Parking to Improve Performance of Successive Bus Acquisitions By a Single Module During Idle Bus Conditions**
- **Offers Accurate Arbitration Settling Time and Glitch Filter Programmability to Allow Optimal Arbitration Bus Performance**
- **Provides a FIFO for Capturing up to Four Incoming Arbitrated Messages**
- **Provides Hardware Support of Targeted Interrupts**
- **Supports Power-Fail Message Indication With a Separate Terminal and Interrupt**
- **Provides On-Chip Error Time-Out Detection**
- **Has a JTAG Test Port**

description

The TFB2010 arbitration bus controller (ABC) is a member of the Texas Instruments Futurebus+ chip set. This chip set provides an integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities.

The TFB2010 performs the Futurebus+ distributed arbitration protocol to gain tenure of the bus (distributed mode only), to send and receive arbitrated messages (central or distributed mode), and to update central-mode arbiter priorities (central mode only).

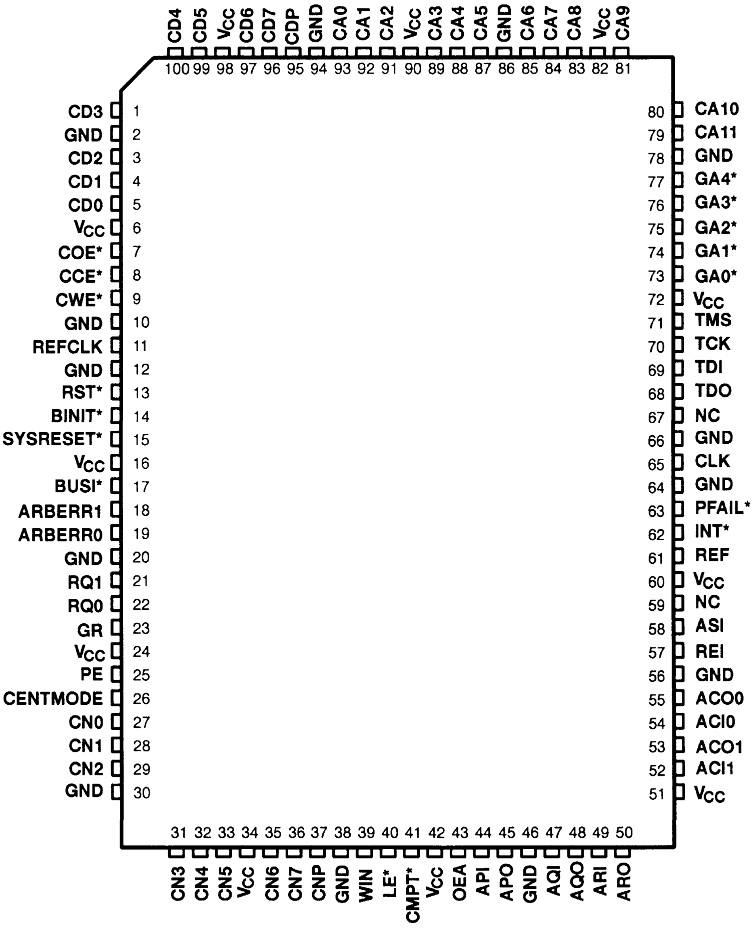
The TFB2010 can be used in conjunction with a central-bus arbiter as an arbitrated-message controller to program the central bus arbiter, send asynchronous interrupts, or send event messages or interrupts to other modules. In the case of a failure in the central-bus arbiter or if distributed arbitration is desired, it can be used as a distributed-arbitration controller without a change in the host software. Priority changes are sent to the central arbiter as arbitrated messages. This device monitors the bus for arbitration messages, storing these in a FIFO or in the targeted interrupt register for reference by the processor. It also provides the necessary control functions to gain control of the Futurebus+ for a module attempting to perform a bus transaction when operating in the distributed-arbitration mode.

The TFB2010 is offered in a 100-pin plastic quad flat package (PJM) to enhance interface capability. The TFB2010 is characterized for operation over the commercial temperature range of 0°C to 70°C.

NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896.1–1991), an active low-signal is denoted herein by use of the trailing asterisk (*) on the signal name.

terminal assignments

PJM . . . PACKAGE
(TOP VIEW)



NC – No internal connection

Terminal Functions

CSR bus

TERMINAL NAME	TERMINAL NO.	I/O	FROM/TO	DESCRIPTION
CA<11:0>	79,80,81,83, 84,85,87,88, 89,91,92,93	I	CSR bus	CSR bus address inputs
CCE*	8	I	CSR bus	CSR bus chip enable input
CD<7:0>	96,97,99, 100,1,3,4,5	I/O	CSR bus	CSR bus data
CDP	95	I/O	CSR bus	CSR bus data odd parity
COE*	7	I	CSR bus	CSR bus output enable input
CWE*	9	I	CSR bus	CSR bus write enable input

protocol controller interface

TERMINAL NAME	TERMINAL NO.	I/O	FROM/TO	DESCRIPTION
ARBERR<1:0>	18,19	O		Arbitration error outputs: LL No error LH AC0 and AC1 asserted during phase 3 HL Arbitration comparison error or parity error HH Arbitration time-out error (phase 2 or 4)
GR	23	O		Futurebus+ mastership has been granted output (bus tenure may begin). This signal remains in the high-impedance state while in the central-bus arbitration mode.
PE	25	I/O		In distributed mode when this device is the bus master, the TFB2010 asserts PE to indicate that a module with a higher priority has become the master elect. PE is released along with GR when RQ1 and RQ0 are released. In central mode, the TFB2010 puts this output in a high-impedance state to allow the central arbitration controller to control preemption. PE is monitored by the TFB2010 during a Futurebus+ system reset to determine the system operational mode (central or distributed) following the reset.
RQ<1:0>	21,22	I		Futurebus+ mastership is requested in centralized mode input: RQ0 asserted: use arbitration number in the RQ0 priority register RQ1 asserted: use arbitration number in the RQ1 priority register Once a request is asserted, it is not released until GR* has been asserted (the TI protocol controllers perform this handshake internally). Once GR* is asserted, RQn* may be released at any time after AS has been asserted by the module in the last bus transaction (AS may already be released if no further transactions are to take place). Both request lines must be released prior to release of GR*. Another RQn* can be asserted after GR* and PE have been released.

Terminal Functions

other module interfaces

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
CLK	65	I		Clock input. CLK is used by the CSR bus master(s).
INT*	62	O (open-collector)		Host interrupt output. When an enabled interrupt condition occurs, INT is driven low. Interrupts are cleared by writing a zero to the appropriate bit in the interrupt register. The interrupt goes high during the write cycle to the interrupt register even if another interrupt is pending.
PFAIL*	63	O		Power-fail message received output
REFCLK	11	I	Module	Clock input. The recommended frequency and duty cycle are 33 MHz, 50%±5%; 25 MHz to 33 MHz and 50% ±5% can be tolerated.

JTAG test port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
TCK	70	I	Module	JTAG test clock input
TDI	69	I	Module	JTAG test data input
TDO	68	O	Module	JTAG test data output
TMS	71	I	Module	JTAG test mode select input

reset port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
BINIT*	14	I	Module	Bus interface reset input. BINIT is an open-collector signal indicating that a bus interface reset is required
BUSI*	17	I		Bus has been idle for longer than 1 μs, and reset is asserted by this module
REF	61	O		Futurebus+ reset filtered output
REI	57	I		Futurebus+ reset input
RST*	13	I	Module	Module power-up reset input. RST resets all logic; output signals go to their inactive states; 3-state outputs and bidirectionals go to the high-impedance state (for live-insertion considerations).
SYSRESET*	15	I	Module	System reset input. SYSRESET* signal indicates that a system reset is required.

Terminal Functions

Futurebus+ interface

TERMINAL NAME	 NO.	I/O	DESCRIPTION
ACI<1:0>	52,54	I	Futurebus+ arbitration condition input
ACO<1:0>	53,55	O	Futurebus+ arbitration condition output
API, AQI, ARI	44,47,49	I	Futurebus+ arbitration handshake input
APO, AQO, ARO	45,48,50	O	Futurebus+ arbitration handshake output
ASI	58	I	Futurebus+ address handshake input
CENTMODE	26	O	Central-mode operation is in effect output
CMPT*	41	O	Arbitration contest logic compete indication output. Connects to $\overline{\text{COMPETE}}$ and $\overline{\text{OEB}}$ on the competition transceiver.
CN<7:0>, CNP	36,35,33,32,31, 29,28,27,37	I/O	Futurebus+ contest number and parity
GA<4:0>*	77,76,75,74,73	I	Futurebus+ geographical address input
LE*	40	O	Enable latch on competition transceiver output (1 = competition number latched)
OEA	43	O	Enable TTL drivers on competition transceiver output
WIN	39	I	Arbitration contest logic win indication input

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage range, V_{CC} (see Note 1)	−0.5 V to 7 V
Input voltage range, V_I (at any input)	−0.5 V to 7 V
Output voltage range, V_O	−0.5 V to 7 V
Continuous total power dissipation	See Dissipation Rating Table
Power dissipation	500 mW
Operating free-air temperature range, T_A	0°C to 70°C
Storage temperature range	−65°C to 150°C
Case temperature for 10 seconds	260°C

NOTE 1: All voltage values are with respect to GND.

DISSIPATION RATING TABLE

PACKAGE	$T_A \le 25°C$ POWER RATING	DERATING FACTOR ABOVE $T_A = 25°C$	$T_A = 70°C$ POWER RATING
PJM	1500 mW	12 mW/°C	960 mW

recommended operating conditions

	MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.75	5	5.25	V
High-level input voltage, V_{IH}	2		V_{CC}	V
Low-level input voltage, V_{IL}	−0.5		0.8	V
Operating free-air temperature range, T_A	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1$ μA, $C_L = 7.4$ pF		1.3		V
V_{IT+}	Positive-going input threshold voltage	IPI09LK			1.6		V
V_{IT-}	Negative-going input threshold voltage	IPI09LK			1.2		V
V_{OH}	High-level output voltage	OPJ43LK	$I_{OH} = -4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 4$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ83LK	$I_{OH} = -8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 8$ mA			0.5	V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 4$ mA			0.5	V
V_{OL}	Low-level output voltage	OPI42LK	$I_{OL} = 4$ mA			0.5	V

macros

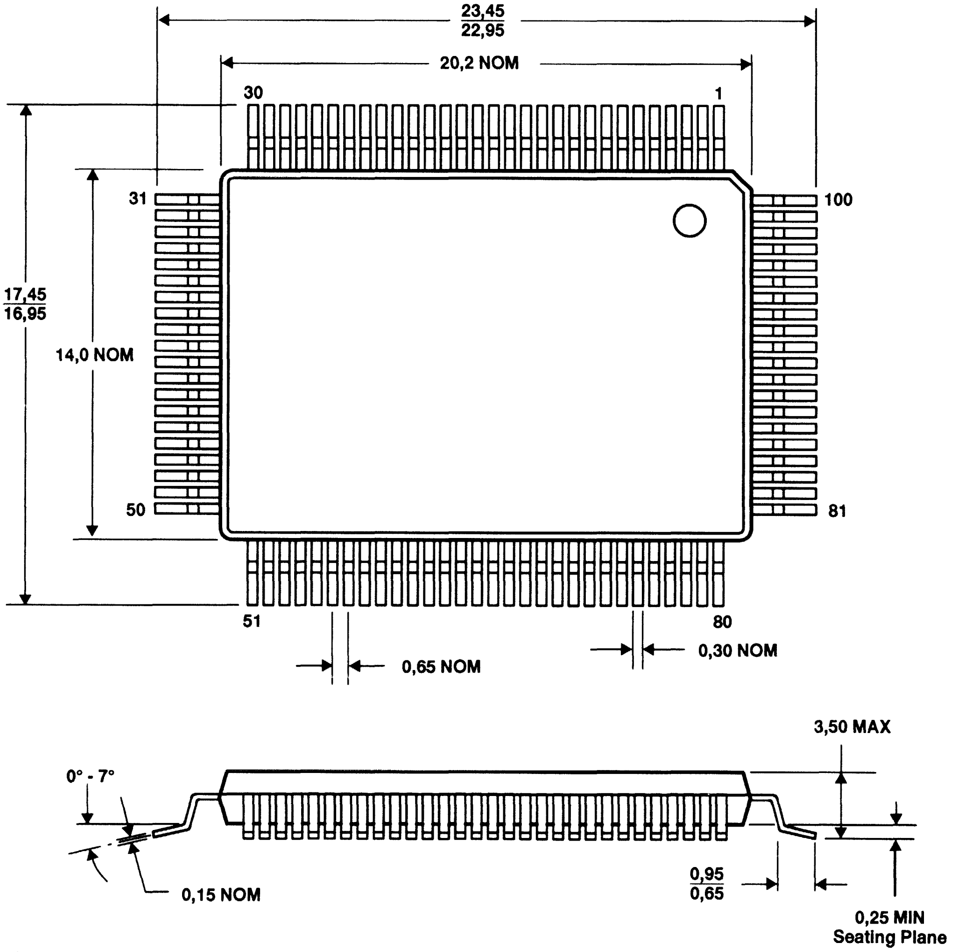
Table 1 lists the internal and external buffer macros used in the TFB2010 design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

Table 1. TFB2010 (ABC) Pin Names and Macro Numbers

PIN NAME	INPUT MACRO	OUTPUT MACRO
ACI<1:0>	IPI04LK	
ACO<1:0>		OPI43LK
API	IPI04LK	
APO		OPI43LK
AQI	IPI04LK	
AQO		OPI43LK
ARBERR<1:0>		OPI43LK
ARI	IPI04LK	
ARO		OPI43LK
ASI	IPI04LK	
BINIT*	IPI09LK	
BUSI*	IPI09LK	
CA<11:0>	IPI04LK	
CCE*	IPI04LK	
CD<7:0>	IPI04LK	OPJ83LK
CDP	IPI04LK	OPJ83LK
CENTMODE		OPI43LK
CLK	IPI04LK	
CMPT*		OPI43LK
CN<7:0>	IPI04LK	OPI43LK
CNP	IPI04LK	OPI43LK
COE*	IPI04LK	
CWE*	IPI04LK	
GA<4:0>*	IPI04LK	
GR		OPI43LK
INT*		OPI42LK
LE*		OPI43LK
OEA		OPI43LK
PE	IPI04LK	OPI43LK
PFAIL*		OPI43LK
REF		OPI43LK
REFCLK	IPI04LK	
REI	IPI04LK	
RQ<1:0>	IPI04LK	
RST*	IPI09LK	
SYSRESET*	IPI09LK	
TCK	IPI04LK	
TDI	IPI04LK	
TDO		OPI43LK
TMS	IPI04LK	
WIN	IPI04LK	

MECHANICAL DATA

100-Pin Plastic Quad Flat Package (PJM Package)



Leads are coplanar to within 0,1 mm.

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS.

TFB2022A
FUTUREBUS+ DATA PATH UNIT

SLLS169A – OCTOBER 1990 – REVISED MARCH 1994

- **Parallel-Protocol Support Is Fully Compliant to Futurebus+ Standard (IEEE Std 896.1–1991)**
- **Interfaces Easily to a Variety of Popular Microprocessors Such as SPARC™, 680x0, 88xxx, 80x86, and Alpha AXP™**
- **Can Be Used in Conjunction With the TFB2002B Futurebus+ I/O Controller or Standalone With a User-Defined Controller**
- **64 Data Channels and 8 Parity Channels on Board**
- **Supports 32 or 36 Bits of Addressing**
- **On-Board Address Decoding Determines Whether Transaction Is to Host Memory, Extended Unit Space, Message-Passing Mailbox, or Other CSR Location**
- **Parallel-Protocol-Related CSR Locations Are Provided on Chip**
- **Provides Support for Module Live Insertion**
- **Handles Both Packet and Compelled Transfers**
- **Capable of Buffering up to 256 Bytes Per Transaction**

description

The TFB2022A data path unit (DPU) is a member of the Texas Instruments Futurebus+ (FB+) chip set. This chip set provides an integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities. The Futurebus+ chip set is capable of supporting 32- or 64-bit data widths in any combination on both the host-bus interface (HIF) and Futurebus+. The address width is programmable to be 32 bits or 36 bits (with either data width).

The TFB2022A may be used with a TFB2002B Futurebus+ I/O controller to provide a complete 64-bit Profile-B interface. It allows great flexibility in the design of the system and in the host features that may be supported. It may also be used with a user-defined controller to provide a variety of performance features. When used together, the TFB2022A and TFB2002B provide the Futurebus+ and host-bus protocol control for the first 64 bits of data and 36 bits of address. The TFB2022A contains a bidirectional FIFO for high-speed transmission of data in either compelled or packet mode, address control for 36 bits of address, and related CSR locations. All Profile-A- and Profile-B-required CSRs are implemented either on this device or the TFB2002B.

The TFB2022A is optimized for Profile-B modules. Several processors may reside on a single module with the DPU as long as they do not require the DPU/IOC to understand cache-coherent operation. The module may contain memory or I/O units in addition to processors. The TFB2022A is best suited for I/O or memory modules.

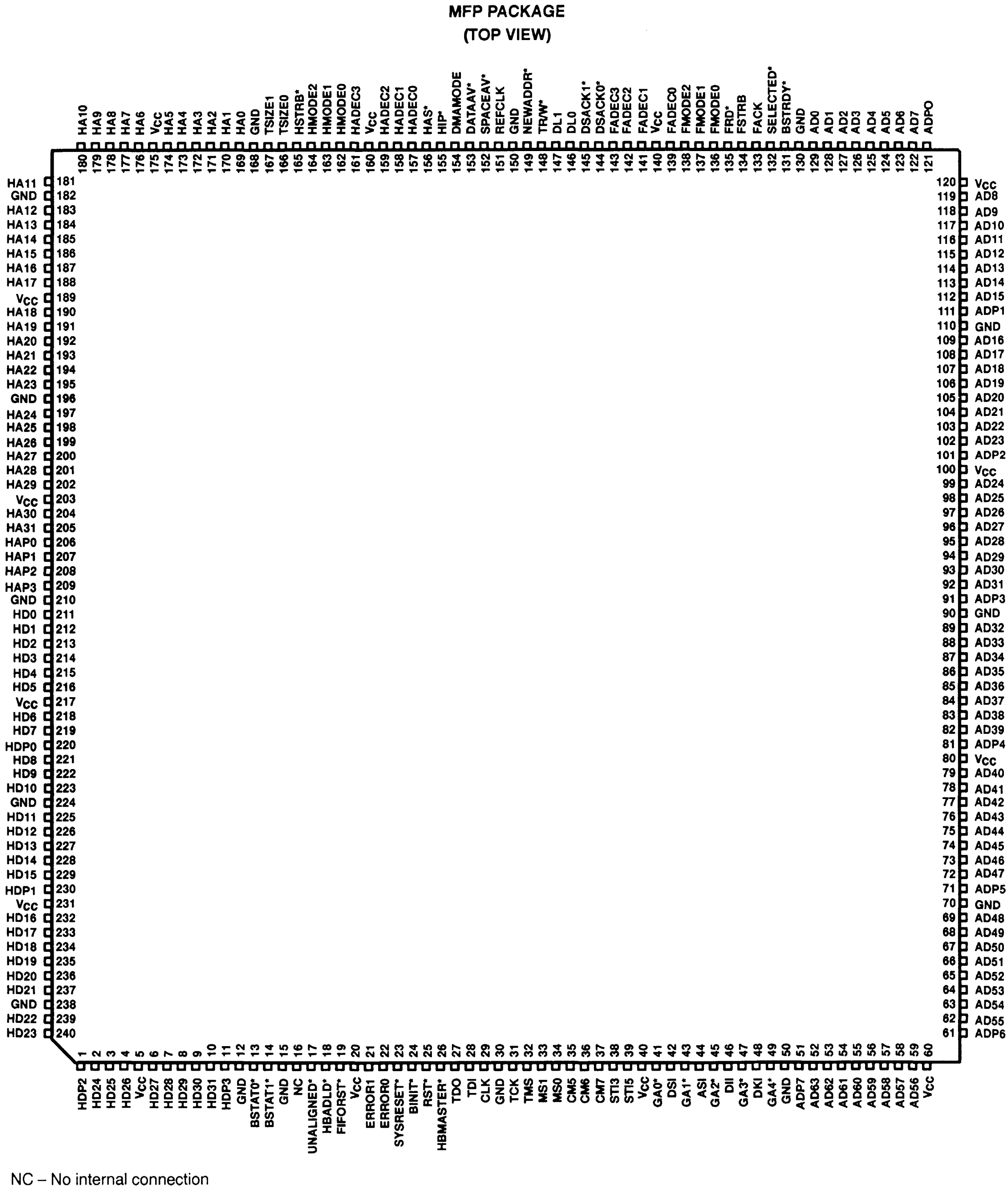
The MS<1:0> signals provide a preaddress decode mechanism, enabling the user to implement simplified decode logic in the logic interface. These signals indicate whether an access is being made to host memory, extended units space, host CSR space, or to a message mailbox.

The TFB2022A is offered in a 240-pin metal quad flat package (MFP). The TFB2022A is characterized for operation over the commercial temperature range of 0°C to 70°C.

NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896.1–1991), an active-low signal is denoted herein by use of the trailing asterisk (*) on the signal name.

SPARC is a trademark of Sun Microsystems, Inc.
Alpha AXP is a trademark of Digital Equipment Corporation.

terminal assignments



NC – No internal connection

Terminal Functions

host interface

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
BSTAT<1:0>*	14, 13	I	Host interface	Host interface status: HH Normal HL Reserved LH Bus error LL Backoff/retry
BSTRDY*	131	I	Host interface	Burst ready input
CLK	29	I	Host interface	Clock input. CLK is the processor clock for synchronous transactions on the host side. Up to 25 MHz is recommended.
DL<1:0>	147, 146	I	Host interface	Host interface data length: LL 64 bytes LH 32 bytes HL 16 bytes HH 8 bytes
DSACK<1:0>*	145, 144	I	Host interface	Data acknowledge input: Single mode (TBST* = high): Burst mode (TBST* = low): LL Complete cycle, data bus port 32 LL Low-speed, 32-bit burst capable LH Reserved LH High-speed, 32-bit burst capable HL Reserved HL Low-speed, 64-bit burst capable HH Insert wait state HH High-speed, 64-bit burst capable
HA<31:0>	205–204, 202–197, 195–190, 188–183, 181–176, 174–169	I/O	Host interface	Host interface address or upper quadlet of data
HAP<3:0>	209–206	I/O	Host interface	Extended host interface address or parity for upper quadlet of host interface data
HAS*	156	I	Host interface	Host interface address strobe input
HD<31:0>	10–6, 4–2, 240–239, 237–232, 229–225, 223–221, 219–218, 216–211	I/O	Host interface	Lower quadlet of host interface data
HDP<3:0>	11, 1, 230, 220	I/O	Host interface	Parity for lower quadlet of host interface data
HIP*	155	I	Host interface	Host interface transaction in progress input
TR/W*	148	I	Host interface	Host interface read or write input
TSIZE<1:0>	167, 166	I/O	Host interface	Host interface transaction size: LL Word (32 bits or greater) LH Byte (8 bits) HL Half word (16 bits) HH Three bytes (24 bits)

Terminal Functions

other module interface signals

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
MS<1:0>	33, 34	O	Module	Memory decode of the host address: LL Unselected LH Host memory HL Host unit space HH CSR space
REFCLK	151	I	Module	Clock input. A 25-MHz, 50%±5% duty-cycle signal is recommended; any frequency between 25 MHz and 33 MHz and duty cycle of 50%±5% can be tolerated. This DPU signal determines packet-mode transfer speed.

interface to TFB2002B

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
DATAAV*	153	O	TFB2002B IOC	Data available in FIFO output. In compelled mode, DATAAV* indicates if any data is in the FIFO. In packet or burst mode, this signal indicates if a packet or burst data of length encoded on Futurebus+ packet size or the DL<1:0> lines is available.
DMAMODE	154	I	TFB2002B IOC	DMA operation is occurring input. FMODE and HMODE are modified for this function.
ERROR<1:0>	21, 22	O	TFB2002B IOC	Futurebus+ error indicators: LL No error LH Futurebus+ parity error HL Packet longitudinal error HH Host interface data parity error
FADEC<3:0>	143, 142, 141, 139	O	TFB2002B IOC	Futurebus+ address decode: LLLL Unselected LLLH Host memory LLHL Host extended unit space LLHH Host CSR LHLL Broadcast mailbox LHLH Reserved LHHL Reserved LHHH Reserved HLLL Mailbox address HLLH Packet-mode-capable memory address HLHL Reserved HLHH Reserved HHLL Reserved HHLH DPU CSR HHHL Broadcast CSR (non-DPU) HHHH Broadcast CSR (DPU)
FIFORST*	19	I	TFB2002B IOC	FIFORST* resets the FIFO pointers input
FACK	133	O	TFB2002B IOC	Futurebus+ acknowledge outputs: Futurebus+ event complete
FMODE<2:0>	138, 137, 136	I	TFB2002B IOC	Futurebus+ mode input. Indicates to the TFB2022A what action is to be taken in the Futurebus+ interface: LLL Compelled-mode Futurebus+ LLH Packet-mode Futurebus+ LHL Partial transfer LHH Disconnect data for master write HLL Reserved HLH Reserved HHL Disconnect data for split requestor HHH Reserved

Terminal Functions

interface to TFB2002B (continued)

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
FRD*	135	I	TFB2002B IOC	Futurebus+ read/write indicator input: L = read (move data from Futurebus+ to FIFO); H = write (move data from FIFO to Futurebus+)
FSTRB	134	I	TFB2002B IOC	Futurebus+ strobe input: perform next Futurebus+ event
HADEC<3:0>	161, 159, 158, 157	O	TFB2002B IOC	Host address decode output. Address decoding for the host interface address: Slave encoding: Master encoding: LLLL Unselected LLLL Unselected LLLH Host memory LLLH Memory address compelled LLHL Host unit space LLHL Maximum burst capable or extended unit space LLHH Host CSR LLHH Memory address 64-byte burst LHLL Broadcast mailbox LHLL 32-byte-memory-address capable LHLH Reserved LHLH 16-byte-memory-address capable LHHL Split response hit LHHL 8-byte-memory-address capable LHHH Futurebus+ CSR address LHHH Reserved HLLL Reserved HLLL Reserved HLLH Broadcast CSR address HLLH Reserved HLHL Reserved HLHL Reserved HLHH Reserved HLHH Reserved HHLL Reserved HHLL Reserved HHLH DPU CSR HHLH Reserved HHHL Reserved HHHL Reserved HHHH Reserved HHHH Reserved
HBADLD*	18	I	TFB2002B IOC	Host address load input. Futurebus+ has been granted for the requested transaction
HBMASTER*	26	I	TFB2002B IOC	Host master input. Indicates host interface mastership
HMODE<2:0>	164, 163, 162	I	TFB2002B IOC	Host mode. Indicates to the TFB2022A what action is to be taken in the host interface: LLL Reserved LLH Between FIFO and host interface (single transfer), between TFB2022A resident CSR and host interface, or between TFB2022A resident CSR and FIFO LHL From FIFO to TFB2022A resident CSR LHH Reserved HLL Reserved HLH Between FIFO and host bus (burst mode) HHL Reserved HHH Reserved
HSTRB*	165	I	TFB2002B IOC	Host strobe input. Perform next host interface request as indicated in HMODE<2:0>
NEWADDR*	149	I	TFB2002B IOC	New address input. Increment address in the TFB2022A address register
SELECTED*	132	I	TFB2002B IOC	Module selected input. Futurebus+ transaction uses this module. The DPU is used as a slave of the Futurebus+ transaction.
SPACEAV*	152	O	TFB2002B IOC	Space available in FIFO. In compelled mode, this signal indicates that space is available in the FIFO for another transfer. In packet or burst mode, this signal indicates that space is available in the FIFO for another packet or burst.
UNALIGNED*	17	O	TFB2002B IOC	FB+ slaved partial unaligned operation output

Terminal Functions

JTAG test port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
TCK	31	I	Module	JTAG test clock input
TDI	28	I	Module	JTAG test data input
TDO	27	O	Module	JTAG test data output
TMS	32	I	Module	JTAG test-mode select input

reset port

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
BINIT*	24	I	Module	Signal indicating that a bus interface reset is required
RST*	25	I	Module	Module power-up reset input. RST* resets all logic; output signals go to their inactive states, and 3-state outputs and bidirectional signals take on the high-impedance state.
SYSRESET*	23	I	Module	System reset input. Signal indicating that a system reset is required

Futurebus+ Interface

TERMINAL		I/O	FROM/TO	DESCRIPTION
NAME	NO.			
AD<63:0>	52–59, 62–69, 72–79, 82–89, 92–99, 102–109, 112–119, 122–129	I/O	Futurebus+	Multiplexed Futurebus+ address and data
ADP<7:0>	51, 61, 71, 81, 91, 101, 111, 121	I/O		Futurebus+ parity
ASI	44	I	Futurebus+	Futurebus+ address synchronization strobe
CM<7:5>	37, 36, 35	I	Futurebus+	Futurebus+ command bits
DSI, DKI, DII	42, 48, 46	I	Futurebus+	Futurebus+ data path synchronization signals in: data strobe (DSI), data acknowledge (DKI), data acknowledge inverse (DII)
GA<4:0>	49, 47, 45, 43, 41	I	Futurebus+	Geographical address
STI5, STI3	39, 38	I	Futurebus+	Futurebus+ status inputs

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage range, V_{CC} (see Note 1)	−0.5 V to 7 V
Input voltage range, V_I (at any input)	−0.5 V to 7 V
Output voltage range, V_O	−0.5 V to 7 V
Continuous total power dissipation	See Dissipation Rating Table
Power dissipation	2 W
Operating free-air temperature range, T_A	0°C to 70°C
Storage temperature range	−65°C to 150°C
Case temperature for 10 seconds	260°C

NOTE 1: All voltage values are with respect to GND.

DISSIPATION RATING TABLE

PACKAGE	$T_A \le 25°C$ POWER RATING	DERATING FACTOR ABOVE $T_A = 25°C$	$T_A = 70°C$ POWER RATING
MFP	6250 mW	50 mW/°C	4000 mW

recommended operating conditions

	MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.75	5	5.25	V
High-level input voltage, V_{IH}	2		V_{CC}	V
Low-level input voltage, V_{IL}	−0.5		0.8	V
Operating free-air temperature range, T_A	0		70	°C

electrical characteristics over recommend operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1$ µA, $C_L = 7.4$ pF		1.3		V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 4$ mA			0.5	V
V_{OH}	High-level output voltage	OPI83LK	$I_{OH} = -8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 8$ mA			0.5	V
V_{OH}	High-level output voltage	OPIH3LK	$I_{OH} = -12$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 12$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ43LK	$I_{OH} = -4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 4$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ83LK	$I_{OH} = -8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 8$ mA			0.5	V

macros

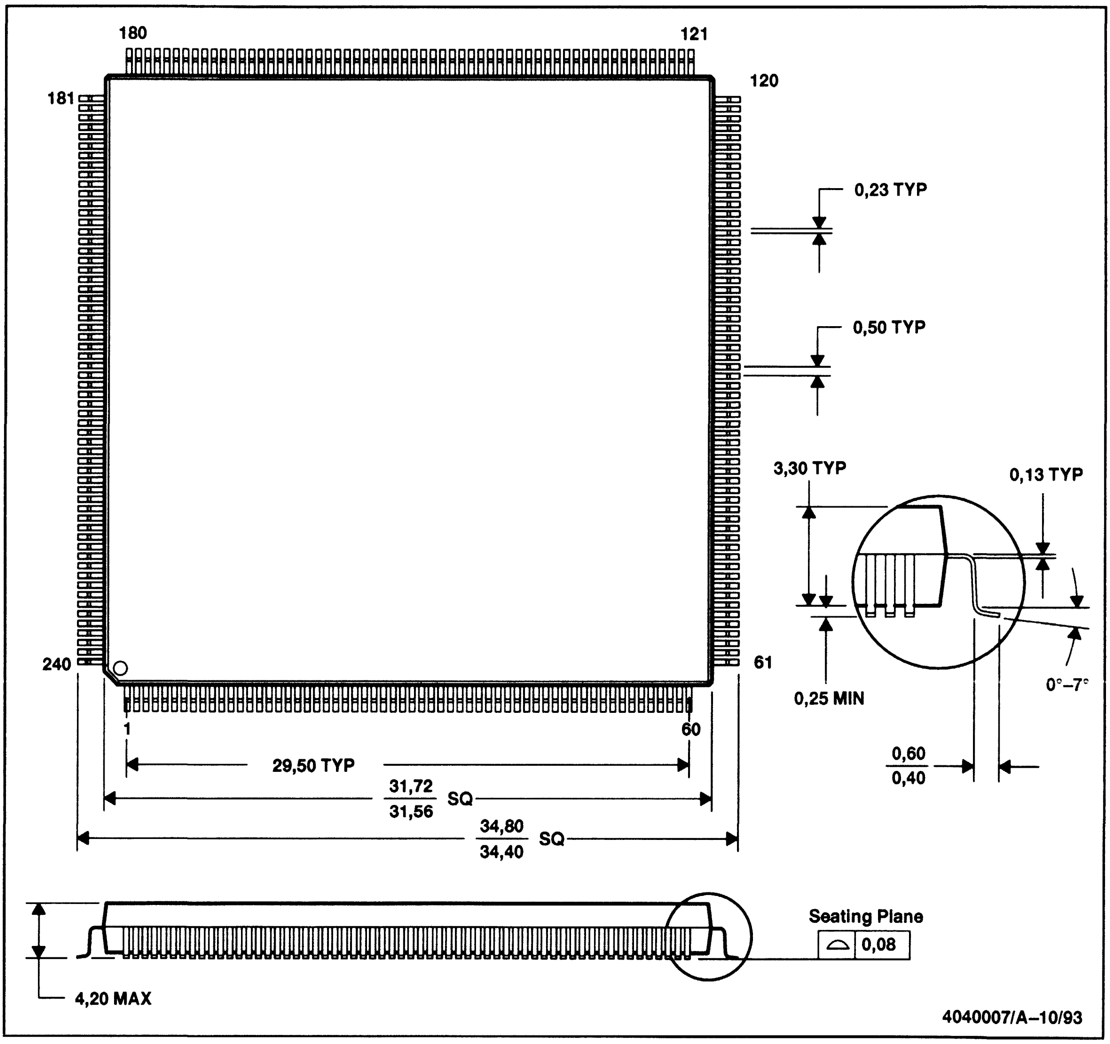
Table 1 lists the internal and external buffer macros used in the TFB2022A design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

Table 1. TFB2022A (DPU) Pin Names and Macro Numbers

PIN NAME	INPUT MACRO	OUTPUT MACRO
AD<63:0>	IPI04LK	OPJ43LK
ADP<7:0>	IPI04LK	OPJ43LK
ASI	IPI04LK	
BINIT*	IPI04LK	
BSTAT<1:0>*	IPI04LK	
BSTRDY*	IPI04LK	
CLK	IPI04LK	
CM<7:5>	IPI04LK	
DATAAV*		OPI43LK
DII	IPI04LK	
DKI	IPI04LK	
DL<1:0>	IPI04LK	
DMAMODE	IPI04LK	
DSACK<1:0>*	IPI04LK	
DSI	IPI04LK	
ERROR<1:0>		OPI43LK
FACK		OPI43LK
FADEC<3:0>		OPI43LK
FIFORST*	IPI04LK	
FMODE<2:0>	IPI04LK	
FRD*	IPI04LK	
FSTRB	IPI04LK	
GA<4:0>*	IPI04LK	
HA<31:0>	IPI04LK	OPJ83LK
HADEC<3:0>		OPI43LK
HAP<3:0>	IPI04LK	OPJ83LK
HAS*	IPI04LK	
HBADLD*	IPI04LK	
HBMASTER*	IPI04LK	
HD<31:0>	IPI04LK	OPJ83LK
HDP<3:0>	IPI04LK	OPJ83LK
HIP*	IPI04LK	
HMODE<2:0>	IPI04LK	
HSTRB*	IPI04LK	
MS<1:0>		OPI83LK
NEWADDR*	IPI04LK	
REFCLK	IPI04LK	
RST*	IPI04LK	
SELECTED*	IPI04LK	
SPACEAV*		OPI43LK
STI<5,3>	IPI04LK	
SYSRESET*	IPIO4LK	
TCK	IPI04LK	
TDI	IPI04LK	
TDO		OPI43LK
TMS	IPI04LK	
TR/W*	IPI04LK	
TSIZE<1:0>	IPI04LK	OPIH3LK
UNALIGNED*		OPI43LK

MECHANICAL DATA

MFP/S-MQFP-G240 **METAL QUAD (MQUAD®) CAVITY-UP FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. MQUAD is a registered trademark of Olin Corporation.
D. This quad flat package consists of a circuit mounted on a leadframe and encased within an anodized aluminum shell. The package is intended for parts requiring either a lower stress environment or higher thermal dissipation capabilities than can be supplied by plastic. Ultrasonic cleaning of this package or boards with this package is not permitted.

Chapter 8

Military Controller Data Sheets

TFB2002BM
FUTUREBUS+ I/O CONTROLLER

SGLS073A – JANUARY 1992 – REVISED MARCH 1994

- **Provides Control Logic Necessary to Operate a Data Path Unit (TFB2022A) on Futurebus+**
- **Parallel-Protocol Support is Fully Compliant to Futurebus+ Standard (IEEE Std 896.1–1991)**
- **Interfaces Easily to a Variety of Popular Microprocessors Such as SPARC™, R4000, 680X0, 88XXX, 80X86, and Alpha AXP™**
- **Package is a Ceramic Quad Flat-Pack With a 3 Inch Non Conductive Tie Bar. Lead Pitch is 0.5 mm (19.7 mil).**
- **Provides Full Support for Futurebus+ Cache Commands (for Memory or I/O Modules in Shared-Memory Systems)**
- **Capable of Handling a Single Outstanding Split Transaction**
- **Parallel-Protocol-Related CSR Locations Are Provided on Chip**
- **Offers Autonomous Control for Futurebus+ and Host-Module Reads and Writes**

description

The TFB2002BM I/O controller (IOC) is a member of the Texas Instruments Futurebus+ chip set. This chip set provides a highly integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities. The Futurebus+ chip set is capable of supporting 32- or 64-bit data widths in any combination on both the host interface and Futurebus+. The address width is programmable to be 32 bits or 36 bits (with either data width).

The TFB2002BM contains the control logic necessary to translate Futurebus+ transactions into host bus transactions and vice versa. It contains a high-speed Futurebus+ handshake controller, a synchronous host bus controller, and a reset type determination logic.

When controlled with a TFB2022AM Futurebus+ data path unit (DPU), the TFB2002BM provides a complete 64-bit-wide interface to the Futurebus+. The TFB2002BM provides the necessary control logic for the data path unit to provide a complete interface to the Futurebus+ for a Profile-B-compliant module. It may also be used on I/O or memory modules in a cache-coherent system.

The TFB2002BM is characterized over the full military temperature range of –55°C to 125°C.

PRODUCT PREVIEW

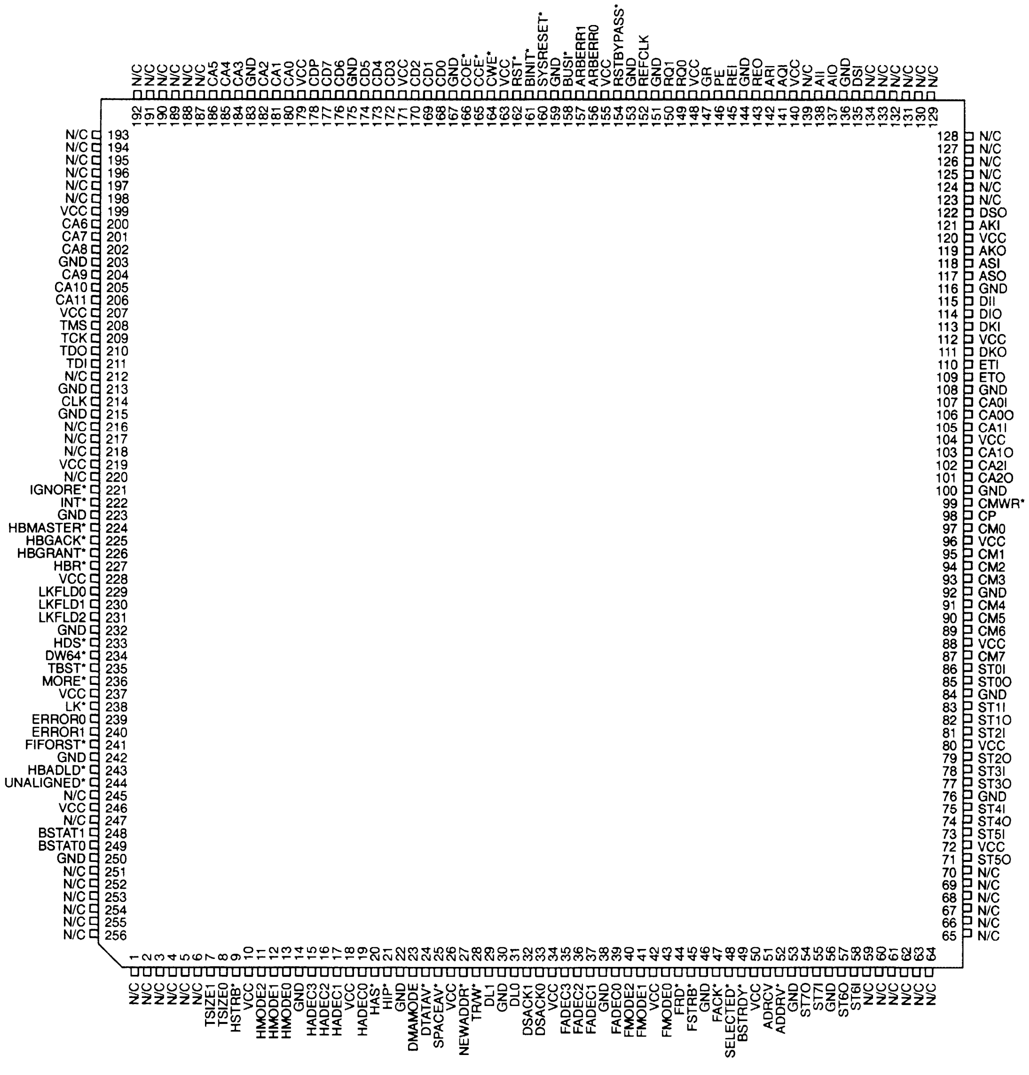
NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896.1–1991), an active-low signal is denoted herein by use of the trailing asterisk(*) on the signal name.

SPARC is a trademark of Sun Microsystems, Inc.
Alpha AXP is a trademark of Digital Equipment Corporation.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

terminal assignments

CQFP PACKAGE
(TOP VIEW)



PRODUCT PREVIEW

terminal functions and detailed description

Refer to the commercial section under TFB2002B for terminal functions and a detailed description. However, please use this section for electrical and switching characteristics for military applications.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1$ µA, $C_L = 7.4$ pF		1.3		V
V_{IT+}	Positive-going input threshold voltage	IPI09LK			1.6		V
V_{IT-}	Negative-going input threshold voltage				1.2		V
V_{OL}	Low-level output voltage	OPI82LK	$I_{OL} = 6.8$ mA			0.5	V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -3.4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 6.8$ mA			0.5	V
V_{OH}	High-level output voltage	OPI83LK	$I_{OH} = -6.8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 6.8$ mA			0.5	V
V_{OH}	High-level output voltage	OPIH3LK	$I_{OH} = -10.2$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 10.2$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ83LK	$I_{OH} = -6.8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 6.8$ mA			0.5	V

† Typical values are at $V_{CC} = 5$ V and $T_A = 25°C$.

macros

Table 1 lists the internal and external buffer macros used in the TFB2002BM design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

Table 1. TFB2002BM (IOC) Pin Names and Macro Numbers

PIN NAME	INPUT MACRO	OUTPUT MACRO
ADDRV*		OPIH3LK
ADRCV		OPIH3LK
AII	IPI04LK	
AIO		OPI43LK
AKI	IPI04LK	
AKO		OPI43LK
AQI	IPI04LK	
ARBERR<1:0>	IPI04LK	
ARI	IPI04LK	
ASI	IPI04LK	
ASO		OPI43LK
BINIT*		OPI43LK
BSTAT1*, BSTAT0*	IPI04LK	OPIH3LK
BSTRDY*	IPI04LK	OPIH3LK
BUSI*		OPI43LK
CA<11:0>	IPI04LK	
CAI<2:0>	IPI04LK	
CAO<2:0>		OPI43LK
CCE*	IPI04LK	
CD<7:0>	IPI04LK	OPJ83LK
CDP	IPI04LK	OPJ83LK
CLK	IPI04LK	
CM<7:0>	IPI04LK	OPI43LK
CMWR*		OPI43LK
COE*	IPI04LK	
CP	IPI04LK	OPI43LK
CWE*	IPI04LK	
DATAAV*	IPI04LK	
DII	IPI04LK	
DIO		OPI43LK
DKI	IPI04LK	
DKO		OPI43LK
DL<1:0>	IPI04LK	OPIH3LK
DMAMODE		OPI43LK
DSACK1*, DSACK0*	IPI04LK	OPIH3LK
DSI	IPI04LK	
DSO		OPI43LK
DW64*	IPI04LK	OPIH3LK

PRODUCT PREVIEW

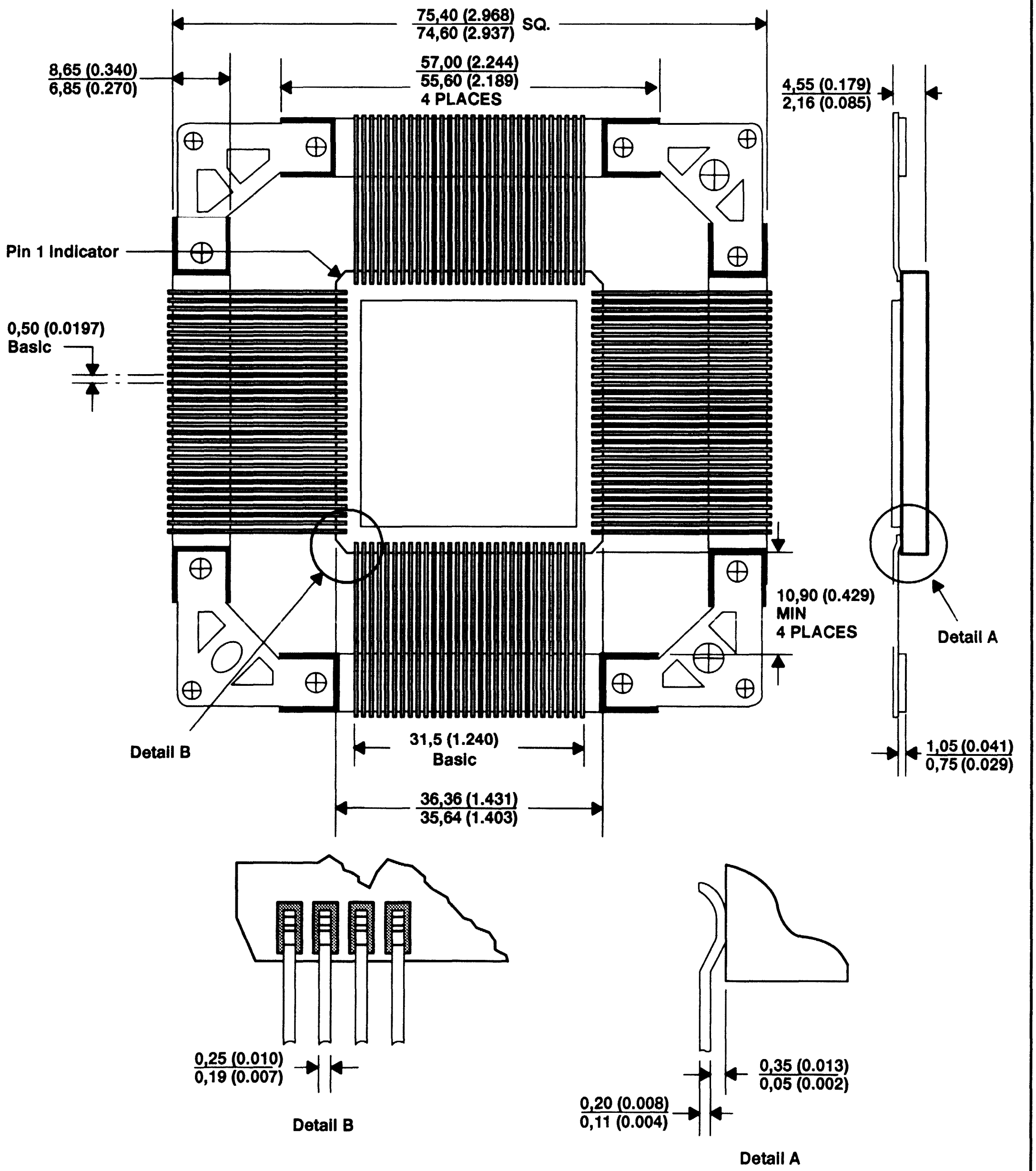
Table 1. TFB2002BM (IOC) Pin Names and Macro Numbers (continued)

PIN NAME	INPUT MACRO	OUTPUT MACRO
ERROR<1:0>	IPI04LK	
ETI	IPI04LK	
ETO		OPI43LK
FACK	IPI04LK	
FADEC<3:0>	IPI04LK	
FIFORST*		OPI43LK
FMODE<2:0>		OPI43LK
FRD*		OPI43LK
FSTRB		OPI43LK
GR	IPI04LK	
HADEC<3:0>	IPI04LK	
HAS*	IPI04LK	OPIH3LK
HBADLD*		OPI43LK
HBG*	IPI04LK	
HBGACK*	IPI04LK	OPIH3LK
HBMASTER*		OPI43LK
HBR*		OPI82LK
HDS*	IPI04LK	OPIH3LK
HIP*	IPI04LK	OPIH3LK
HMODE<2:0>		OPI43LK
HSTRB*		OPI43LK
IGNORE*	IPI04LK	
INT*		OPI82LK
LK*	IPI04LK	OPIH3LK
LKFLD0, 1, 2	IPI04LK	OPI43LK
MORE*	IPI04LK	
NEWADDR*		OPI43LK
PE	IPI04LK	
REFCLK	IPI04LK	
REI	IPI04LK	
REO		OPI43LK
RQ<1:0>		OPI43LK
RST*	IPI09LK	
RSTBYPASS*	IPI04LK	
SELECTED*		OPI43LK
SPACEAV*	IPI04LK	
STI<7:0>	IPI04LK	
STO<7:0>		OPI43LK
SYSRESET*		OPI83LK
TBST*	IPI04LK	OPIH3LK
TCK	IPI04LK	
TDI	IPI04LK	OPI43LK
TDO		OPI43LK
TMS	IPI04LK	
TRW*	IPI04LK	OPIH3LK
TSIZE<1:0>	IPI04LK	
UNALIGNED*	IPI04LK	

MECHANICAL DATA

256-PIN CERAMIC QUAD FLAT PACKAGE

NOTE: ALL LEADS NOT SHOWN FOR DRAWING CLARITY PURPOSES



NOTE: ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

PRODUCT PREVIEW

TFB2010M
FUTUREBUS+ ARBITRATION BUS CONTROLLER

SGLS074 – JANUARY 1992 – REVISED NOVEMBER 1993

- **Supports Distributed Arbitration for Futurebus+ Master Selection**
- **Supports Arbitrated Messages In Distributed and Central Modes**
- **Enables Use of a Common Hardware and Software Interface for Both Distributed and Central Modes**
- **Requires No Hardware Modifications for Changing Between Distributed and Central Modes**
- **Provides a CSR Bus Interface for Easy Integration into the Futurebus+ CSR Address Space**
- **Has Two Bus Request Lines That Each May Be Assigned Any One of 256 Priority Levels**
- **Supports Round-Robin Fairness Arbitration Within Two Separate Priority Levels to Avoid Starvation of Any Single Module**
- **Supports Distributed-Mode Bus Parking to Improve Performance of Successive Bus Acquisitions By a Single Module During Idle Bus Conditions**
- **Offers Accurate Arbitration Settling Time and Glitch Filter Programmability to Allow Optimal Arbitration Bus Performance**
- **Provides FIFO for Capturing up to Four Incoming Arbitrated Messages**
- **Provides Hardware Support of Targeted Interrupts**
- **Supports Power-Fail Message Indication With a Separate Terminal and Interrupt**
- **Provides On-Chip Error Time-Out Detection**
- **Has a JTAG Test Port**
- **Package is a Ceramic Quad Flat Package Lead Pitch is 0.5 mm (19.7 mil)**

description

The TFB2010M arbitration bus controller (ABC) is a member of the Texas Instruments Futurebus+ chip set. This chip set provides an integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities. The Futurebus+ chip set is capable of supporting, in any combination, 32- or 64-bit data widths on both the host-bus interface (HIF) and Futurebus+. The address width is programmable to be 32-bits or 36-bits (with either data width).

The TFB2010M performs the Futurebus+ distributed arbitration protocol to gain tenure of the bus (distributed mode only), to send and receive arbitrated messages (central or distributed mode), and to update central-mode arbiter priorities (central mode only).

The TFB2010M can be used in conjunction with a central-bus arbiter as an arbitrated-message controller to program the central bus arbiter, send asynchronous interrupts, or send event messages or interrupts to other modules. In the case of a failure in the central-bus arbiter or if distributed arbitration is desired, it can be used as a distributed-arbitration controller without a change in the host software. Priority changes are sent to the central arbiter as arbitrated messages. This device monitors the bus for arbitration messages, storing these in a FIFO or in the targeted interrupt register for reference by the processor. It also provides the necessary control functions to gain control of the Futurebus+ for a module attempting to perform a bus transaction when operating in the distributed-arbitration mode.

Initialization, interrupt handling, and control of this device are handled through the CSR registers. All registers are memory mapped into the Futurebus+ CSR space and made accessible to a module-processing element through the CSR interface.

Priority numbers ranging from 0 to 255 are independently maintained for RQ<0> and RQ<1> in two CSR registers. A third register is used to send any desired arbitrated message. Priority and send arbitration-message (SAM) registers may be modified at any time via the CSR bus. A newly modified arbitration number, request level (RQ1), or mode of operation is not used by a module until the idle state is observed.

Time-out error checkers are provided for phases 1, 2, 4 and for phase 0 between the first and second pass of a two-pass arbitration cycle. Phases 2 and 4 are hardwired to a time out of between 1 and 2 µs. Programmable wired-OR glitch filters are provided for the handshake lines and the RE input.

NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896.1–1991), an active-low signal is denoted herein by use of the trailing asterisk(*) on the signal name.

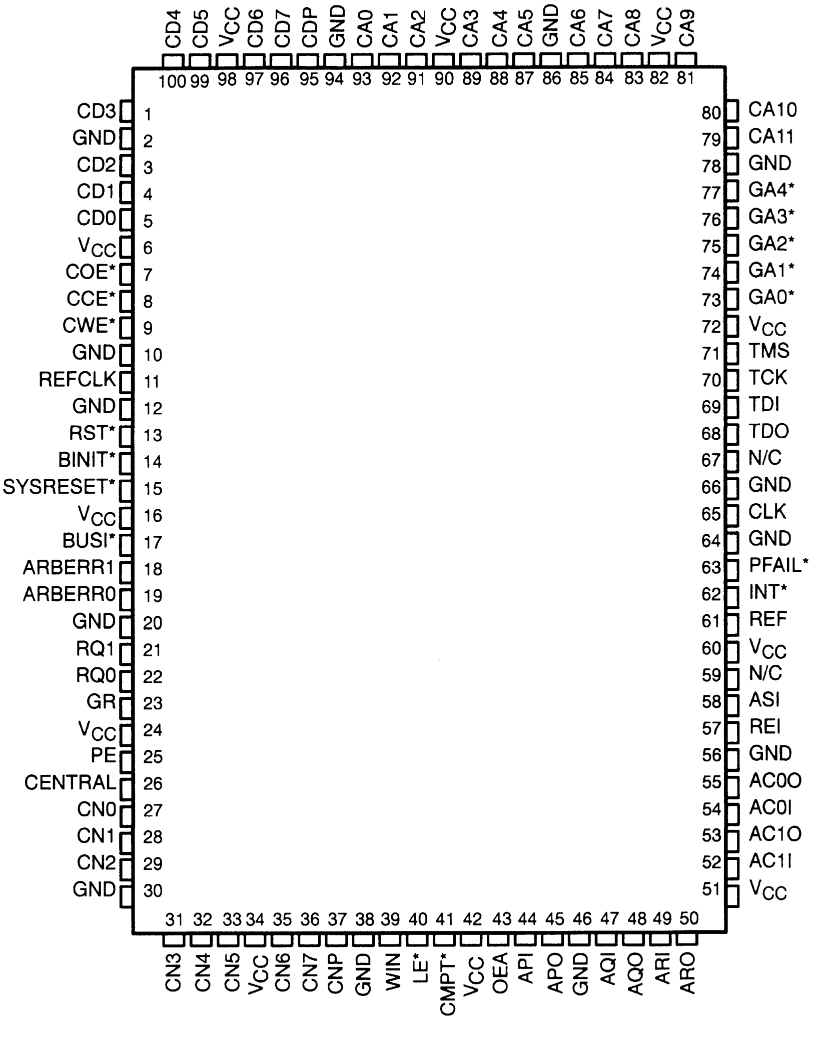
PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

PRODUCT PREVIEW

terminal assignments

CQFP PACKAGE
(TOP VIEW)



terminal functions and detailed description

Refer to the commercial section under TFB2010 for terminal functions and a detailed description. However, please use this section for electrical and switching characteristics for military applications.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1$ µA, $C_L = 7.4$ pF		1.3		V
V_{IT+}	Positive-going input threshold voltage	IPI09LK			1.6		V
V_{IT-}	Negative-going input threshold voltage				1.2		V
V_{OH}	High-level output voltage	OPJ43LK	$I_{OH} = -3.4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 3.4$ mA			0.5	V
V_{OH}	Low-level output voltage	OPJ83LK	$I_{OH} = -6.8$ mA	3.7			V
V_{OL}	High-level output voltage		$I_{OL} = 6.8$ mA			0.5	V
V_{OH}	Low-level output voltage	OPI43LK	$I_{OH} = -3.4$ mA	3.7			V
V_{OL}	High-level output voltage		$I_{OL} = 3.4$ mA			0.5	V
V_{OL}	Low-level output voltage	OPI42LK	$I_{OL} = 3.4$ mA			0.5	V

† Typical values are at $V_{CC} = 5$ V and $T_A = 25°C$.

cell numbers and characteristics

Table 1 lists the internal and external buffer macros used in the TFB2010M design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

Table 1. TFB2010M (ABC) Pin Names and Macro Numbers

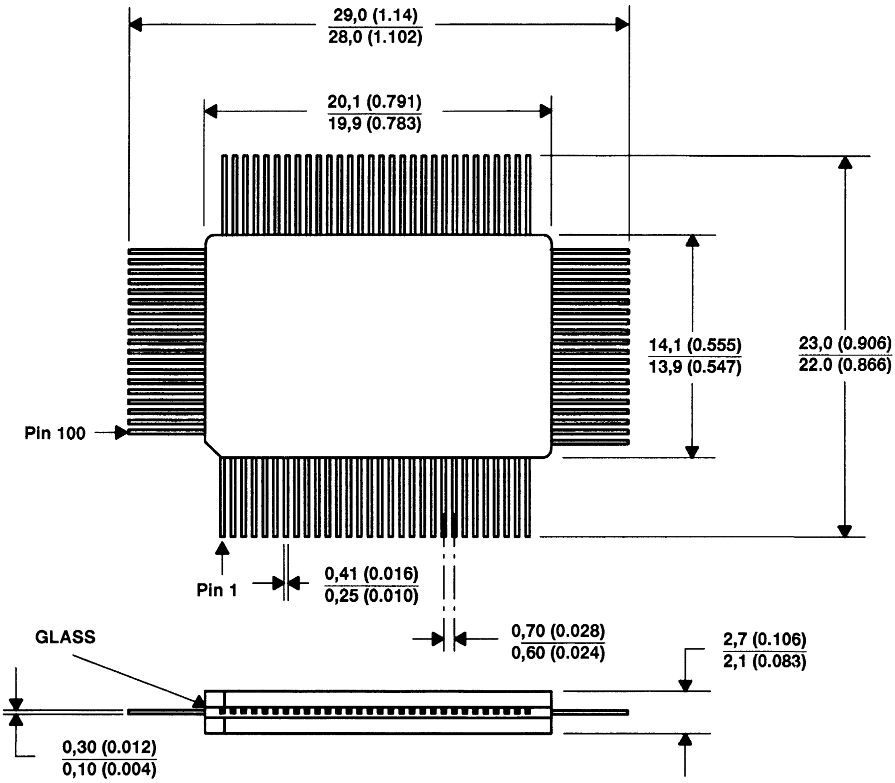
PIN NAME	INPUT MACRO	OUTPUT MACRO
ACI<1:0>	IPI04LK	
ACO<1:0>		OPI43LK
API	IPI04LK	
APO		OPI43LK
AQI	IPI04LK	
AQO		OPI43LK
ARBERR<1:0>		OPI43LK
ARI	IPI04LK	
ARO		OPI43LK
ASI	IPI04LK	
BINIT*	IPI09LK	
BUSI*	IPI09LK	
CA<11:0>	IPI04LK	
CCE*	IPI04LK	
CD<7:0>	IPI04LK	OPJ83LK
CDP	IPI04LK	OPJ83LK
CENTMODE		OPI43LK
CLK	IPI04LK	
CN<7:0>	IPI04LK	OPJ43LK
CNP	IPI04LK	OPJ43LK
COE*	IPI04LK	
COMPT*		OPI43LK
CWE*	IPI04LK	
GA<4:0>*	IPI04LK	
GR		OPI43LK
INT*		OPI42LK
LE*		OPI43LK
OEA		OPI43LK
PE	IPI04LK	OPI43LK
PFAIL		OPI43LK
REF		OPI43LK
REFCLK	IPI04LK	
REI	IPI04LK	
RQ<1:0>	IPI04LK	
RST*	IPI09LK	
SYSRESET*	IPI09LK	
TCK	IPI04LK	
TDI	IPI04LK	
TDO		OPI43LK
TMS	IPI04LK	
WIN	IPI04LK	

PRODUCT PREVIEW

PRODUCT PREVIEW

MECHANICAL DATA

100-PIN CERAMIC QUAD FLAT PACKAGE



NOTE: ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES.

TFB2022AM
FUTUREBUS+ DATA PATH UNIT

- Parallel-Protocol Support Is Fully Compliant to Futurebus+ Standard (IEEE Std 896.1–1991)
- Interfaces to a Variety of Popular Microprocessors Such as SPARC™, 680x0, 88xxx, 80x86, and Alpha AXP™
- Can be Used in Conjuction With the TFB2002BM or Standalone With a User-Defined Controller
- 64 Data Channels and 8 Parity Channels on Board
- Supports 32 or 36 Bits of Addressing
- Package is a Ceramic Quad Flat Package With a 3-Inch Nonconductive Tie Bar, Lead Pitch is 0.5 mm (19.7 mil)
- Can Handle a Single Outstanding Split Transaction
- On-Board Address Decoding Determines Whether Transaction is to Host Memory, Extended Unit Space, Message-Passing Mailbox, or Other CSR Location
- Parallel-Protocol-Related CSR Locations Are Provided on Chip
- Provides Support for Module Live Insertion
- Handles Both Packet and Compelled Transfers
- Capable of Buffering up to 256 Bytes Per Transaction

description

The TFB2022AM data path unit (DPU) is a member of the Texas Instruments Futurebus+ chip set. This chip set provides an integrated approach to the Futurebus+ interface that reduces new-product design time, allows more functionality per circuit board, improves overall interface reliability, and reduces end-user down time through built-in test capabilities.The Futurebus+ chip set is capable of supporting, in any combination, 32- or 64-bit data widths on both the host bus interface and Futurebus+. The address width is programmable to be 32 bits or 36 bits (with either data width).

The TFB2022AM may be used with a TFB2002BM Futurebus+ I/O controller to provided a complete 64-bit Profile B interface. It allows great flexibility in the design of the system and in the host features that may be supported. It may also be used with a user-defined controller to provide a variety of performance features. When used together, the TFB2022AM and TFB2002BM provide the Futurebus+ and host bus protocol control for the first 64 bits of data and 36 bits of address. The TFB2022AM contains a bidirectional FIFO for high-speed transmission of data in either compelled or packet mode, address control for 36 bits of address, and related CSR locations. All Profile-A-/ and Profile-B-required CSRs are implemented either on this device the or TFB2002BM.

The TFB2022AM is optimized for Profile-B modules. Several processors may reside on a single module with the DPU as long as they do not require the DPU/IOC to usderstand cache-coherent operation. The module may contain memory or I/O units in addtion to processors. The TFB2022AM is best suited for I/O or memory modules.

The MS<1:0> signals provide a pre-address decode mechanism, enabling the user to implement simplified decode logic in the logic interface. These signals indicate whether an access is being made to host memory, extended units space, host CSR space or to a message mailbox.

The TFB2022AM is characterized over the full military temperature range of –55°C to 125°C

PRODUCT PREVIEW

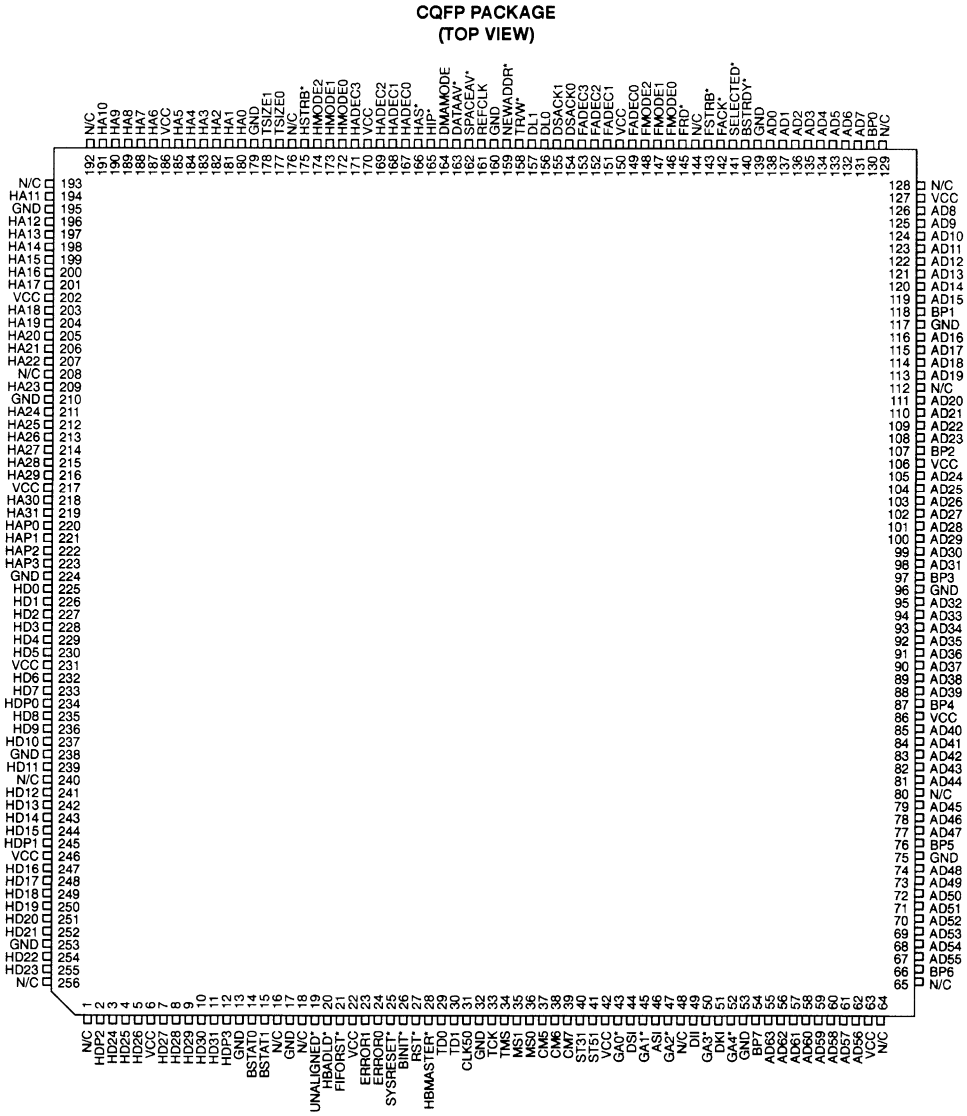
NOTE: To maintain consistency with the notation used in the Futurebus+ standard (IEEE Std 896.1–1991), an active-low signal is denoted herein by use of the trailing asterisk(*) on the signal name.

SPARC is a trademark of Sun Microsystems, Inc.
Alpha AXP is a trademark of Digital Equipment Corporation.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

PRODUCT PREVIEW

terminal assignments



NC – No internal connection

terminal functions and detailed description

Refer to the commercial section under TFB2022A for terminal functions and a detailed description. However, please use this section for electrical and switching characteristics for military applications.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		MACRO	TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{IT}	Input threshold voltage	IPI04LK	$V_I = V_{CC}$ or 0 V, $I_I = \pm 1$ µA, $C_L = 7.4$ pF		1.3		V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -3.4$ mA	3.7			V
V_{OL}	Low-level output voltage	OPI82LK	$I_{OL} = 3.4$ mA			0.5	V
V_{OH}	High-level output voltage	OPI43LK	$I_{OH} = -6.8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 6.8$ mA			0.5	V
V_{OH}	High-level output voltage	OPI83LK	$I_{OH} = -10.2$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 10.2$ mA			0.5	V
V_{OH}	High-level output voltage	OPIH3LK	$I_{OH} = -3.4$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 3.4$ mA			0.5	V
V_{OH}	High-level output voltage	OPJ83LK	$I_{OH} = -6.8$ mA	3.7			V
V_{OL}	Low-level output voltage		$I_{OL} = 6.8$ mA			0.5	V

† Typical values are at $V_{CC} = 5$ V and $T_A = 25°C$.

PRODUCT PREVIEW

macros

Table 1 lists the internal and external buffer macros used in the TFB2022AM design. To use this table, find the pin of interest and note the macro name(s). If there is an entry only in the input macro column, the pin is an input. If there is an entry only in the output macro column, the pin is an output. If there is an entry in both columns, this is a 3-state bidirectional pin. The macro(s) are also listed in the electrical characteristics table.

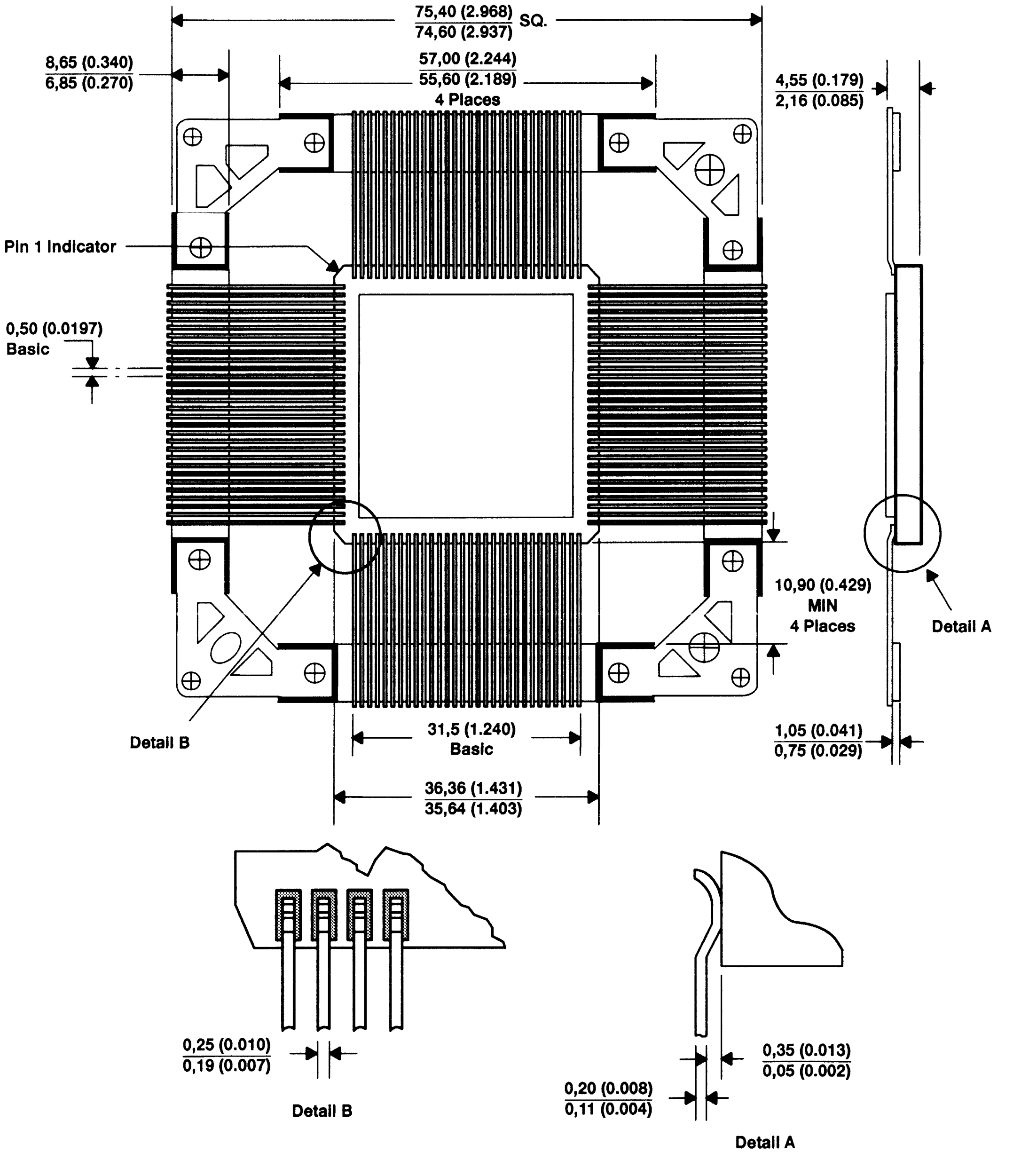
Table 1. TFB2022AM (DPU) Pin Names and Macro Numbers

PIN NAME	INPUT MACRO	OUTPUT MACRO
AD<63:0>	IPI04LK	OPJ43LK
ADP<7:0>	IPI04LK	OPJ43LK
ASI	IPI04LK	
BINIT*	IPI04LK	
BSTAT1*, BSTAT0*	IPI04LK	
BSTSRDY*	IPI04LK	
CLK	IPI04LK	
CM<7:5>	IPI04LK	
DATAAV*		OPI43LK
DII	IPI04LK	
DKI	IPI04LK	
DL<1:0>	IPI04LK	
DMAMODE	IPI04LK	
DSACK1*, DSACK0*	IPI04LK	
DSI	IPI04LK	
ERROR<1:0>		OPI43LK
FACK		OPI43LK
FADEC<3:0>	IPI04LK	OPI43LK
FIFORST*	IPI04LK	
FMODE<2:0>	IPI04LK	
FRD*	IPI04LK	
FSTRB	IPI04LK	
GA<4:0>*	IPI04LK	
HA<31:0>	IPI04LK	OPJ83LK
HADEC<3:0>		OPI43LK
HAP<3:0>	IPI04LK	OPJ83LK
HAS*	IPI04LK	
HBADLD*	IPI04LK	
HBMASTER*	IPI04LK	
HD<31:0>	IPI04LK	OPJ83LK
HDP<3:0>	IPI04LK	OPJ83LK
HIP*	IPI04LK	
HMODE<2:0>	IPI04LK	
HSTRB*	IPI04LK	
MS<1:0>		OPI83LK
NEWADDR*	IPI04LK	
REFCLK	IPI04LK	
RST*	IPI04LK	
SELECTED*	IPI04LK	
SPACEAV*		OPI43LK
STI<5,3>	IPI04LK	
SYSRESET*	IPI04LK	
TCK	IPI04LK	
TDI	IPI04LK	
TDO		OPI43LK
TMS	IPI04LK	
TRW*	IPI04LK	
TSIZE<1:0>	IPI04LK	OPIH3LK
UNALIGNED*		OPI43LK

MECHANICAL DATA

256-PIN CERAMIC QUAD FLAT PACKAGE

NOTE: ALL LEADS NOT SHOWN FOR DRAWING CLARITY PURPOSES



NOTE: ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES.

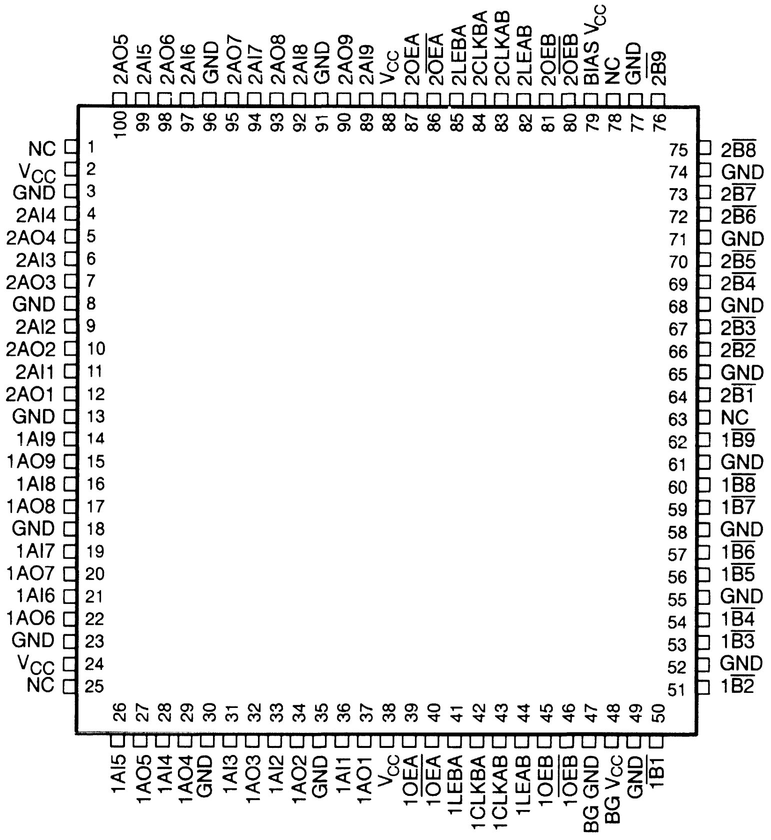
PRODUCT PREVIEW

Chapter 9

Transceiver Data Sheets

- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **Available in Ceramic Quad Flatpack (HQA) Package**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL Input Structures Incorporate Active Clamping and Bus Hold Networks**

**HQA PACKAGE
(TOP VIEW)**



NC – No internal connection

PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description

The SN54FB1650 contains two 9-bit transceivers designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{B}$ port operates at BTL signal levels. The open-collector $\overline{B}$ ports are specified to sink 100 mA. Two output enables, OEB and $\overline{OEB}$, are provided for the $\overline{B}$ outputs. When OEB is low, $\overline{OEB}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{B}$ port is turned off.

The A port operates at TTL signal levels. The A outputs reflect the inverse of the data at the $\overline{B}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Active bus-hold circuitry is provided to hold unused or floating TTL inputs at a valid logic state.

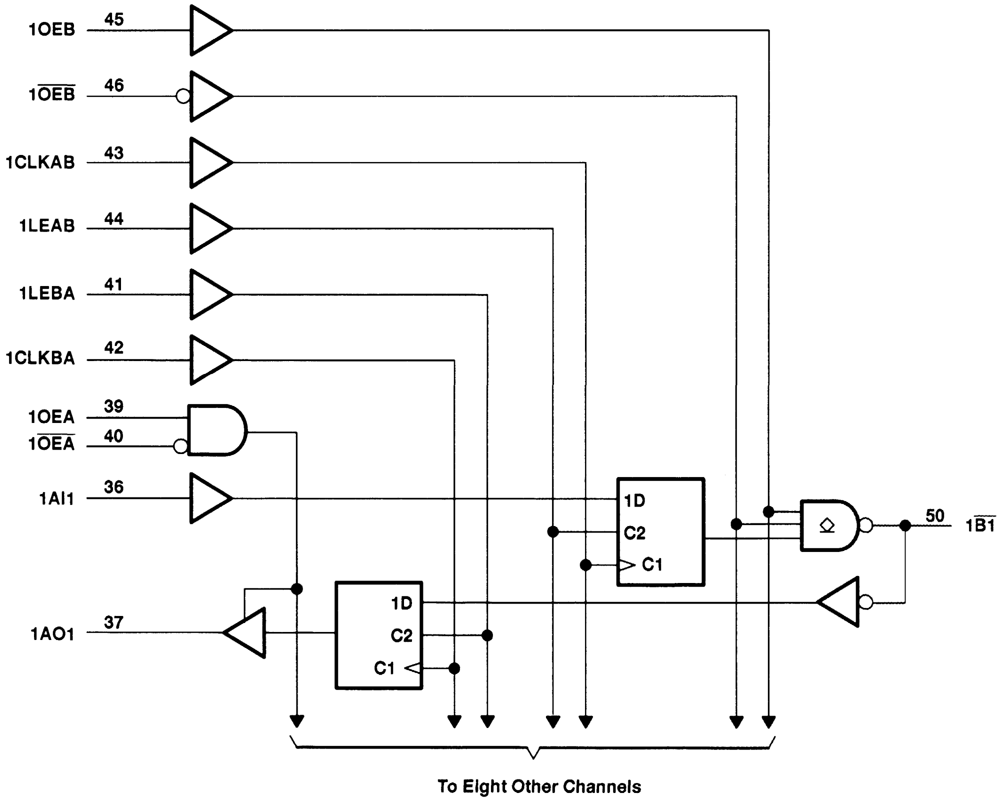
BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN54FB1650 is characterized for operation over the full military temperature range of –55°C to 125°C.

PRODUCT PREVIEW

functional block diagram



TRANSCEIVER FUNCTION TABLE

INPUTS				FUNCTION
$\overline{\text{OEA}}$	OEA	OEB	$\overline{\text{OEB}}$	
X	X	H	L	$\overline{\text{A}}$ data to B bus
L	H	X	X	$\overline{\text{B}}$ data to A bus
L	H	H	L	$\overline{\text{A}}$ data to B bus, $\overline{\text{B}}$ data to A bus
X	X	L	X	B-bus isolation
X	X	X	H	
H	X	X	X	A-bus isolation
X	L	X	X	

STORAGE-MODE TABLE

INPUTS		FUNCTION
LE	CLK	
H	X	Transparent
L	↑	Store data
L	L	Storage

PRODUCT PREVIEW

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias V_{CC}, BG V_{CC}	−0.5 V to 7 V
Input voltage range, V_I: (except $\overline{\text{B}}$ port)	−1.2 V to 7 V
($\overline{\text{B}}$ port)	−1.2 V to 3.5 V
Input current range (except $\overline{\text{B}}$ port)	−40 mA to 5 mA
Voltage range applied to any $\overline{\text{B}}$ output in the disabled or power-off state	−0.5 V to 5.5 V
Voltage range applied to any output in the high state	−0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{\text{B}}$ port	200 mA
Storage temperature range	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			MIN	NOM	MAX	UNIT
V_{CC}, BG V_{CC}	Supply voltage		4.75	5	5.25	V
BIAS V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{\text{B}}$ port	1.62		2.3	V
		Except $\overline{\text{B}}$ port	2			
V_{IL}	Low-level input voltage	$\overline{\text{B}}$ port	0.75		1.47	V
		Except $\overline{\text{B}}$ port			0.8	
I_{IK}	Input clamp current				−18	mA
I_{OH}	High-level output current	A port			−3	mA
I_{OL}	Low-level output current	A port			24	mA
		$\overline{\text{B}}$ port			100	
T_A	Operating free-air temperature		−55		125	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW

electrical characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	$\overline{\text{B}}$ port	V_{CC} = 4.75 V,	I_I = –18 mA			–1.2	V
	Except $\overline{\text{B}}$ port and AO port	V_{CC} = 4.75 V,	I_I = –40 mA			–0.5	V
V_{OH}	AO port	V_{CC} = 4.75 V	I_{OH} = –1 mA				V
			I_{OH} = –3 mA	2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.75 V,	I_{OL} = 24 mA		0.35	0.5	V
	B port	V_{CC} = 4.75 V	I_{OL} = 80 mA	0.75		1.1	
			I_{OL} = 100 mA			1.15	
I_I	Except $\overline{\text{B}}$ port	V_{CC} = 5.25 V,	V_I = 5.25 V			50	µA
I_{IH}‡	Except $\overline{\text{B}}$ port	V_{CC} = 5.25 V,	V_I = 2.7 V			50	µA
I_{IL}‡	Except $\overline{\text{B}}$ port	V_{CC} = 5.25 V,	V_I = 0.5 V			–50	µA
	$\overline{\text{B}}$ port†	V_{CC} = 5.25 V,	V_I = 0.75 V			–100	
$I_{I(hold)}$	AI port	V_{CC} = 5.25 V,	V_I = 2 V	–100			µA
	AI port	V_{CC} = 5.25 V,	V_I = 0.5 V	100			
I_{OZH}	AO port	V_{CC} = 5.25 V,	V_O = 2.7 V			50	µA
I_{OZL}	AO port	V_{CC} = 5.25 V,	V_O = 0.5 V			–50	µA
I_{OH}	$\overline{\text{B}}$ port	V_{CC} = 0 to 5.25 V,	V_O = 2.1 V			100	µA
I_{OS}§	A port	V_{CC} = 5.25 V,	V_O = 0	– 30		–150	mA
I_{CC}	A port to $\overline{\text{B}}$ port	V_{CC} = 5.25 V,	I_O = 0		25		mA
	$\overline{\text{B}}$ port to A port				60		
	Outputs disabled						
C_i		V_I = V_{CC} or GND				5	pF
C_o	A port	V_O = V_{CC} or GND					pF
C_{io}	B port per P1194.0	V_{CC} = 0 to 4.75 V				6	pF
		V_{CC} = 4.75 V to 5.25 V				5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

live insertion specifications over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS			MIN	MAX	UNIT
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.75 V,	V_B = 0 to 2 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V		450	µA
		V_{CC} = 4.25 V to 5.25 V				10	
V_O	$\overline{\text{B}}$ port	V_{CC} = 0,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V		1.62	2.1	V
I_O	$\overline{\text{B}}$ port	V_{CC} = 0,	V_B = 1 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	–1		µA
		V_{CC} = 0 to 5.25 V,	OEB = 0 to 0.8 V			100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V			100	

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

			MIN	MAX	UNIT
f_{clock}	Clock frequency				MHz
t_w	Pulse duration	LE high			ns
		CLK high or low			
t_{su}	Setup time	A or $\overline{B}$ before LE		2	ns
		A or $\overline{B}$ before CLK↑		2	
t_h	Hold time	A or $\overline{B}$ after LE		1	ns
		A or $\overline{B}$ after CLK↑		1	

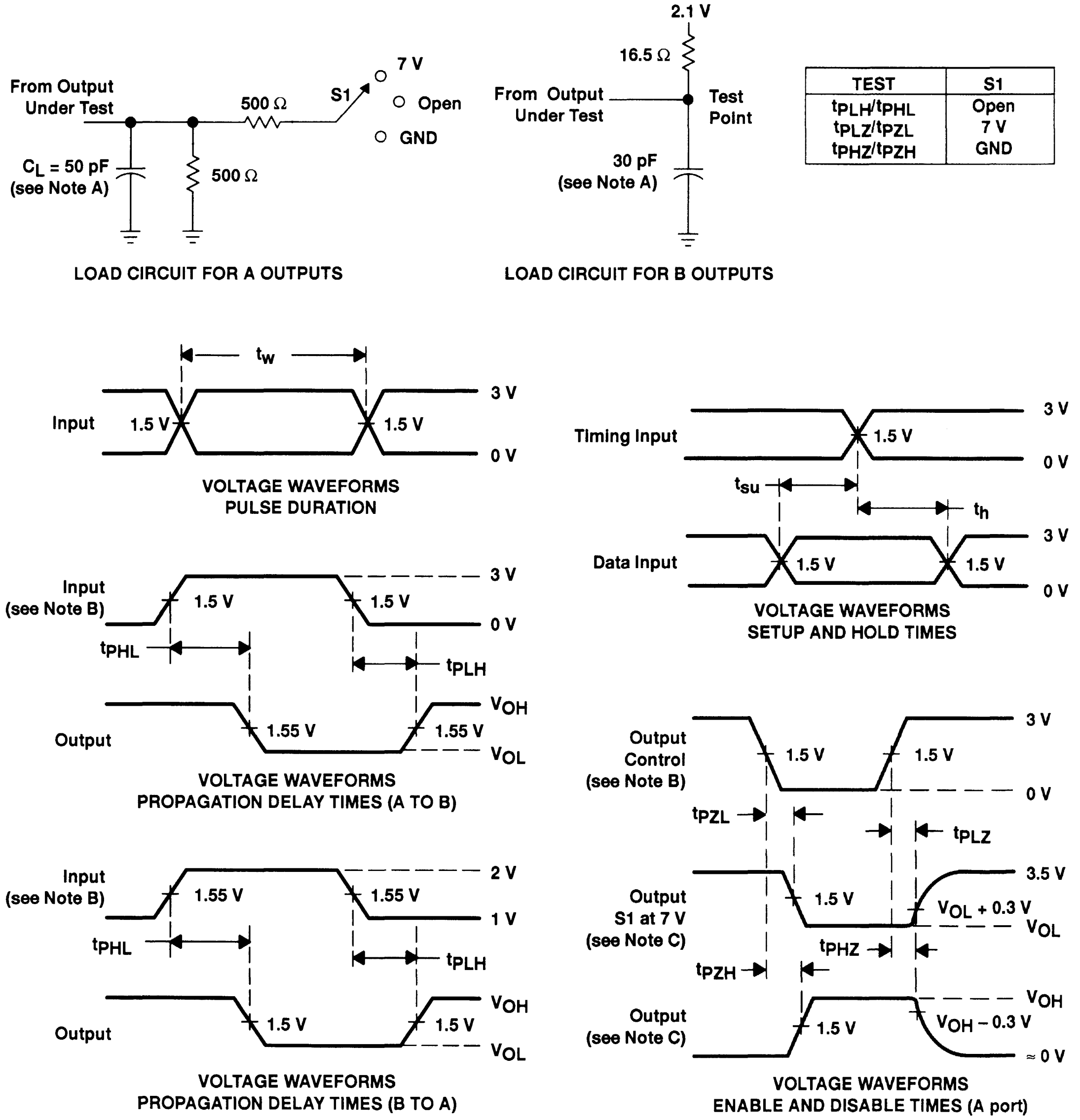
switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	MIN	TYP†	MAX	UNIT
t_{PLH}	A	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PLH}	LEAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	CLKAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	LEBA	A			6	ns
t_{PHL}					6	
t_{PLH}	CLKBA	A			6	ns
t_{PHL}					6	
t_{PLH}	$\overline{B}$	A			5	ns
t_{PHL}					5	
t_{PLH}	OEB or $\overline{OEB}$	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PZH}	OEA or $\overline{OEA}$	A			5	ns
t_{PZL}					5	
t_{PHZ}	OEA or $\overline{OEA}$	A			5	ns
t_{PLZ}					5	
$t_{sk(p)}$‡	Skew for any single channel $\lvert t_{PHL} - t_{PLH} \rvert$	A to $\overline{B}$ or $\overline{B}$ to A		0.5		ns
$t_{sk(o)}$‡	Skew between drivers in the same package	A to $\overline{B}$ or $\overline{B}$ to A		1		ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)		1	2	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection			1		ns

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ Skew values are applicable for through mode only.

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION



TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND

LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A port)

PRODUCT PREVIEW

NOTES: A. C_L includes probe and jig capacitance.

B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.

C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.

D. The outputs are measured one at a time with one transition per measurement.

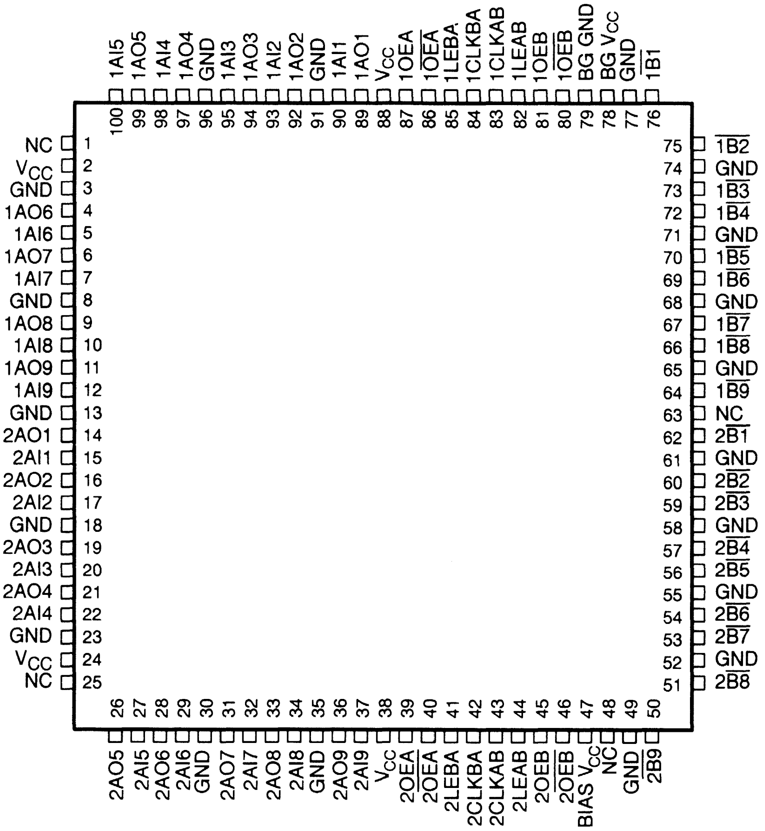
Figure 1. Load Circuit and Voltage Waveforms

SN74FB1650
18-BIT TTL/BTL UNIVERSAL STORAGE TRANSCEIVER

SCBS178B – AUGUST 1992 – REVISED FEBRUARY 1994

- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **Packaged in the High-Power Shrink Quad Flat Package (PCA) With 0.5-mm Pin Pitch**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL Input Structures Incorporate Active Clamping and Bus Hold Networks**

PCA PACKAGE
(TOP VIEW)



100 1AI5
99 1AO5
98 1AI4
97 1AO4
96 GND
95 1AI3
94 1AO3
93 1AI2
92 1AO2
91 GND
90 1AI1
89 1AO1
88 VCC
87 1OEA
86 1$\overline{\text{OEA}}$
85 1LEBA
84 1CLKBA
83 1CLKAB
82 1LEAB
81 1OEB
80 1$\overline{\text{OEB}}$
79 BG GND
78 BG VCC
77 GND
76 1$\overline{\text{B1}}$
NC 1
VCC 2
GND 3
1AO6 4
1AI6 5
1AO7 6
1AI7 7
GND 8
1AO8 9
1AI8 10
1AO9 11
1AI9 12
GND 13
2AO1 14
2AI1 15
2AO2 16
2AI2 17
GND 18
2AO3 19
2AI3 20
2AO4 21
2AI4 22
GND 23
VCC 24
NC 25
26 2AO5
27 2AI5
28 2AO6
29 2AI6
30 GND
31 2AO7
32 2AI7
33 2AO8
34 2AI8
35 GND
36 2AO9
37 2AI9
38 VCC
39 2OEA
40 2$\overline{\text{OEA}}$
41 2LEBA
42 2CLKBA
43 2CLKAB
44 2LEAB
45 2OEB
46 2$\overline{\text{OEB}}$
47 BIAS VCC
48 NC
49 GND
50 2$\overline{\text{B9}}$
75 1$\overline{\text{B2}}$
74 GND
73 1$\overline{\text{B3}}$
72 1$\overline{\text{B4}}$
71 GND
70 1$\overline{\text{B5}}$
69 1$\overline{\text{B6}}$
68 GND
67 1$\overline{\text{B7}}$
66 1$\overline{\text{B8}}$
65 GND
64 1$\overline{\text{B9}}$
63 NC
62 2$\overline{\text{B1}}$
61 GND
60 2$\overline{\text{B2}}$
59 2$\overline{\text{B3}}$
58 GND
57 2$\overline{\text{B4}}$
56 2$\overline{\text{B5}}$
55 GND
54 2$\overline{\text{B6}}$
53 2$\overline{\text{B7}}$
52 GND
51 2$\overline{\text{B8}}$

NC – No internal connection

PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description

The SN74FB1650 contains two 9-bit transceivers designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{B}$ port operates at BTL signal levels. The open-collector $\overline{B}$ ports are specified to sink 100 mA. Two output enables, OEB and $\overline{OEB}$, are provided for the $\overline{B}$ outputs. When OEB is low, $\overline{OEB}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{B}$ port is turned off.

The A port operates at TTL signal levels. The A outputs reflect the inverse of the data at the $\overline{B}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Active bus-hold circuitry is provided to hold unused or floating TTL inputs at a valid logic state.

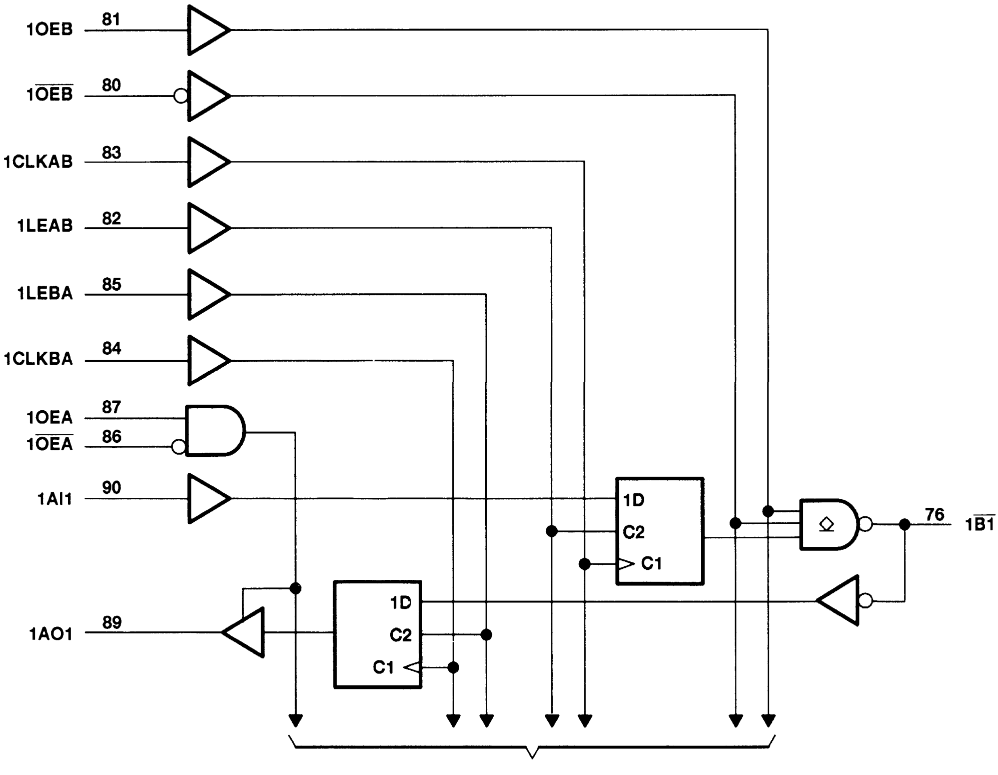
BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN74FB1650 is characterized for operation from 0°C to 70°C.

PRODUCT PREVIEW

functional block diagram



To Eight Other Channels

PRODUCT PREVIEW

TRANSCEIVER FUNCTION TABLE

INPUTS				FUNCTION
$\overline{OEA}$	OEA	OEB	$\overline{OEB}$	
X	X	H	L	$\overline{A}$ data to B bus
L	H	X	X	$\overline{B}$ data to A bus
L	H	H	L	$\overline{A}$ data to B bus, $\overline{B}$ data to A bus
X X	X X	L X	X H	B-bus isolation
H X	X L	X X	X X	A-bus isolation

STORAGE-MODE TABLE

INPUTS		FUNCTION
LE	CLK	
H	X	Transparent
L	↑	Store data
L	L	Storage

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias V_{CC}, BG V_{CC}	–0.5 V to 7 V
Input voltage range, V_I: (except $\overline{B}$ port)	–1.2 V to 7 V
($\overline{B}$ port)	–1.2 V to 3.5 V
Input current range (except $\overline{B}$ port)	–40 mA to 5 mA
Voltage range applied to any $\overline{B}$ output in the disabled or power-off state	–0.5 V to 5.5 V
Voltage range applied to any output in the high state	–0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{B}$ port	200 mA
Storage temperature range	–65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			MIN	NOM	MAX	UNIT
V_{CC}, BG V_{CC}	Supply voltage		4.75	5	5.25	V
BIAS V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	V
		Except $\overline{B}$ port	2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	V
		Except $\overline{B}$ port			0.8	
I_{IK}	Input clamp current				–18	mA
I_{OH}	High-level output current	A port			-3	mA
I_{OL}	Low-level output current	A port			24	mA
		$\overline{B}$ port			100	
T_A	Operating free-air temperature		0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW

electrical characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	$\overline{B}$ port	V_{CC} = 4.75 V,	I_I = –18 mA			–1.2	V
	Except $\overline{B}$ port and AO port	V_{CC} = 4.75 V,	I_I = –40 mA			–0.5	V
V_{OH}	AO port	V_{CC} = 4.75 V	I_{OH} = –1 mA				V
			I_{OH} = –3 mA	2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.75 V,	I_{OL} = 24 mA		0.35	0.5	V
	B port	V_{CC} = 4.75 V	I_{OL} = 80 mA	0.75		1.1	
			I_{OL} = 100 mA			1.15	
I_I	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 5.25 V			50	µA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 2.7 V			50	µA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 0.5 V			–50	µA
	$\overline{B}$ port†	V_{CC} = 5.25 V,	V_I = 0.75 V			–100	
$I_{I(hold)}$	AI port	V_{CC} = 5.25 V,	V_I = 2 V	–100			µA
	AI port	V_{CC} = 5.25 V,	V_I = 0.5 V	100			
I_{OZH}	AO port	V_{CC} = 5.25 V,	V_O = 2.7 V			50	µA
I_{OZL}	AO port	V_{CC} = 5.25 V,	V_O = 0.5 V			–50	µA
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.25 V,	V_O = 2.1 V			100	µA
I_{OS}§	A port	V_{CC} = 5.25 V,	V_O = 0	– 30		–150	mA
I_{CC}	A port to $\overline{B}$ port	V_{CC} = 5.25 V,	I_O = 0		25		mA
	$\overline{B}$ port to A port				60		
	Outputs disabled						
C_i		V_I = V_{CC} or GND				5	pF
C_o	A port	V_O = V_{CC} or GND					pF
C_{io}	B port per P1194.0	V_{CC} = 0 to 4.75 V				6	pF
		V_{CC} = 4.75 V to 5.25 V				5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

live insertion specifications over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		MIN	MAX	UNIT
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.75 V,	V_B = 0 to 2 V, V_I (BIAS V_{CC}) = 4.5 V to 5.5 V		450	µA
		V_{CC} = 4.25 V to 5.25 V			10	
V_O	$\overline{B}$ port	V_{CC} = 0,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	1.62	2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0, V_B = 1 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	–1		µA
		V_{CC} = 0 to 5.25 V,	OEB = 0 to 0.8 V		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100	

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

			MIN	MAX	UNIT
f_{clock}	Clock frequency				MHz
t_w	Pulse duration	LE high			ns
		CLK high or low			
t_{su}	Setup time	A or $\overline{B}$ before LE		2	ns
		A or $\overline{B}$ before CLK↑		2	
t_h	Hold time	A or $\overline{B}$ after LE		1	ns
		A or $\overline{B}$ after CLK↑		1	

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

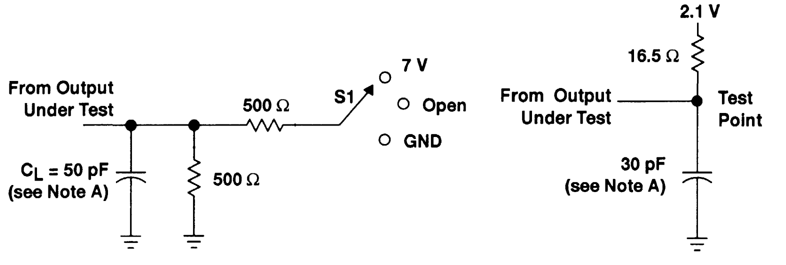
PARAMETER	FROM (INPUT)	TO (OUTPUT)	MIN	TYP†	MAX	UNIT
t_{PLH}	A	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PLH}	LEAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	CLKAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	LEBA	A			6	ns
t_{PHL}					6	
t_{PLH}	CLKBA	A			6	ns
t_{PHL}					6	
t_{PLH}	$\overline{B}$	A			5	ns
t_{PHL}					5	
t_{PLH}	OEB or $\overline{OEB}$	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PZH}	OEA or $\overline{OEA}$	A			5	ns
t_{PZL}					5	
t_{PHZ}	OEA or $\overline{OEA}$	A			5	ns
t_{PLZ}					5	
$t_{sk(p)}$‡	Skew for any single channel | $t_{PHL} - t_{PLH}$ |	A to $\overline{B}$ or $\overline{B}$ to A		0.5		ns
$t_{sk(o)}$‡	Skew between drivers in the same package	A to $\overline{B}$ or $\overline{B}$ to A		1		ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)		1	2	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection			1		ns

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ Skew values are applicable for through mode only.

PRODUCT PREVIEW

PRODUCT PREVIEW

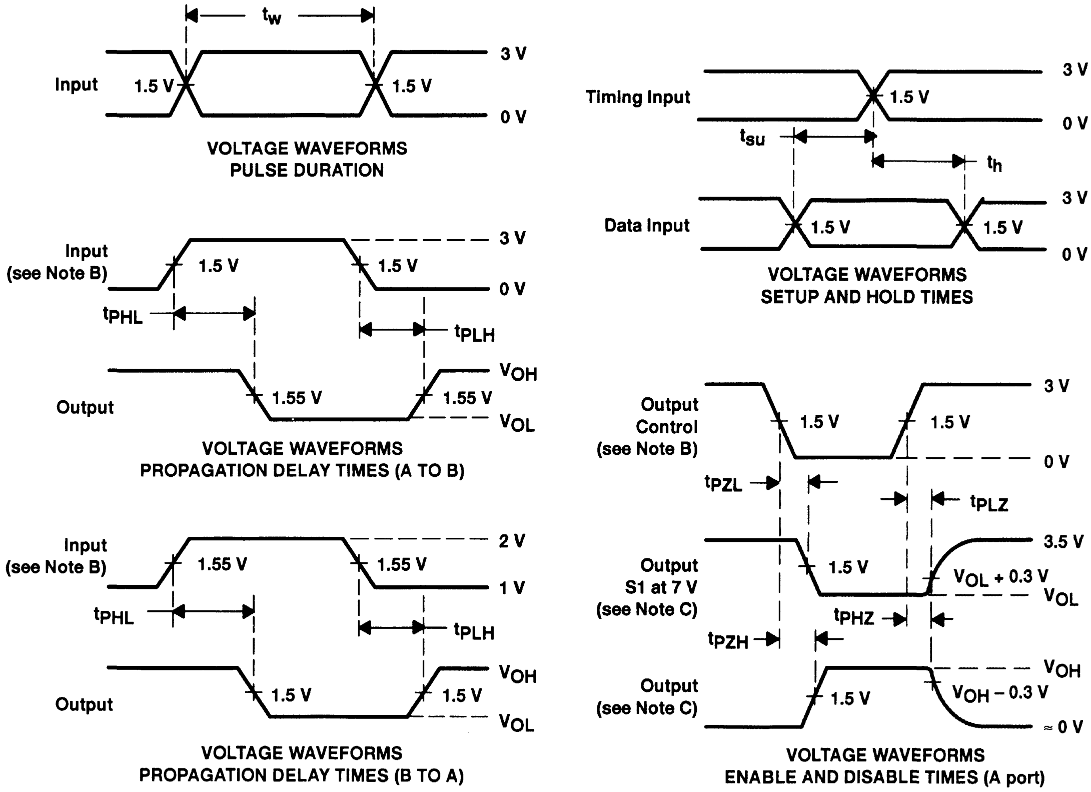
PARAMETER MEASUREMENT INFORMATION



LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND



VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A port)

NOTES: A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

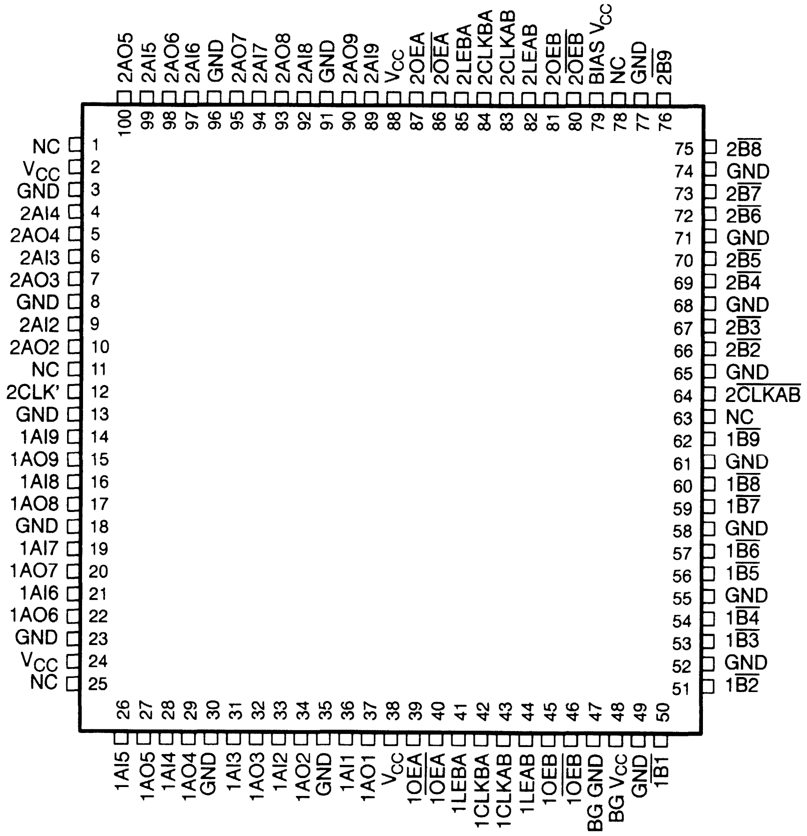
Figure 1. Load Circuit and Voltage Waveforms

SN54FB1651
17-BIT TTL/BTL UNIVERSAL STORAGE TRANSCEIVER WITH BUFFERED CLOCK LINE

SGBS003 – FEBRUARY 1994

- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **Available in Ceramic Quad Flatpack (HQA) Package**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL-Input Structures Incorporate Active Clamping and Bus-Hold Networks**

HQA PACKAGE (TOP VIEW)



NC – No internal connection

PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

TEXAS INSTRUMENTS

description

The SN54FB1651 contains an 8-bit and a 9-bit transceiver with a buffered clock. The clock and the transceivers are designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{B}$ port operates at BTL signal levels. The open-collector $\overline{B}$ ports are specified to sink 100 mA. Two output enables, OEB and $\overline{OEB}$, are provided for the $\overline{B}$ outputs. When OEB is low, $\overline{OEB}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{B}$ port is turned off.

The A port operates at TTL signal levels. The A outputs reflect the inverse of the data at the $\overline{B}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Active bus-hold circuitry is provided to hold unused or floating TTL inputs at a valid logic state.

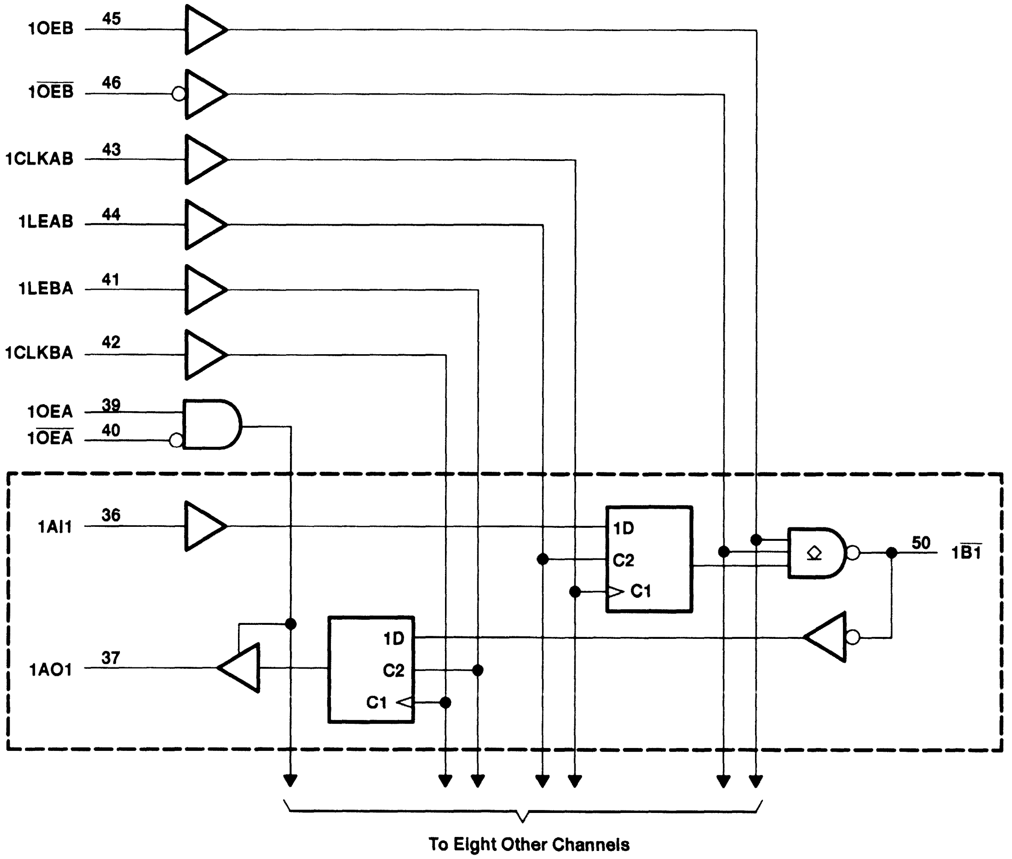
BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN54FB1651 is characterized for operation over the full military temperature range of –55°C to 125°C.

PRODUCT PREVIEW

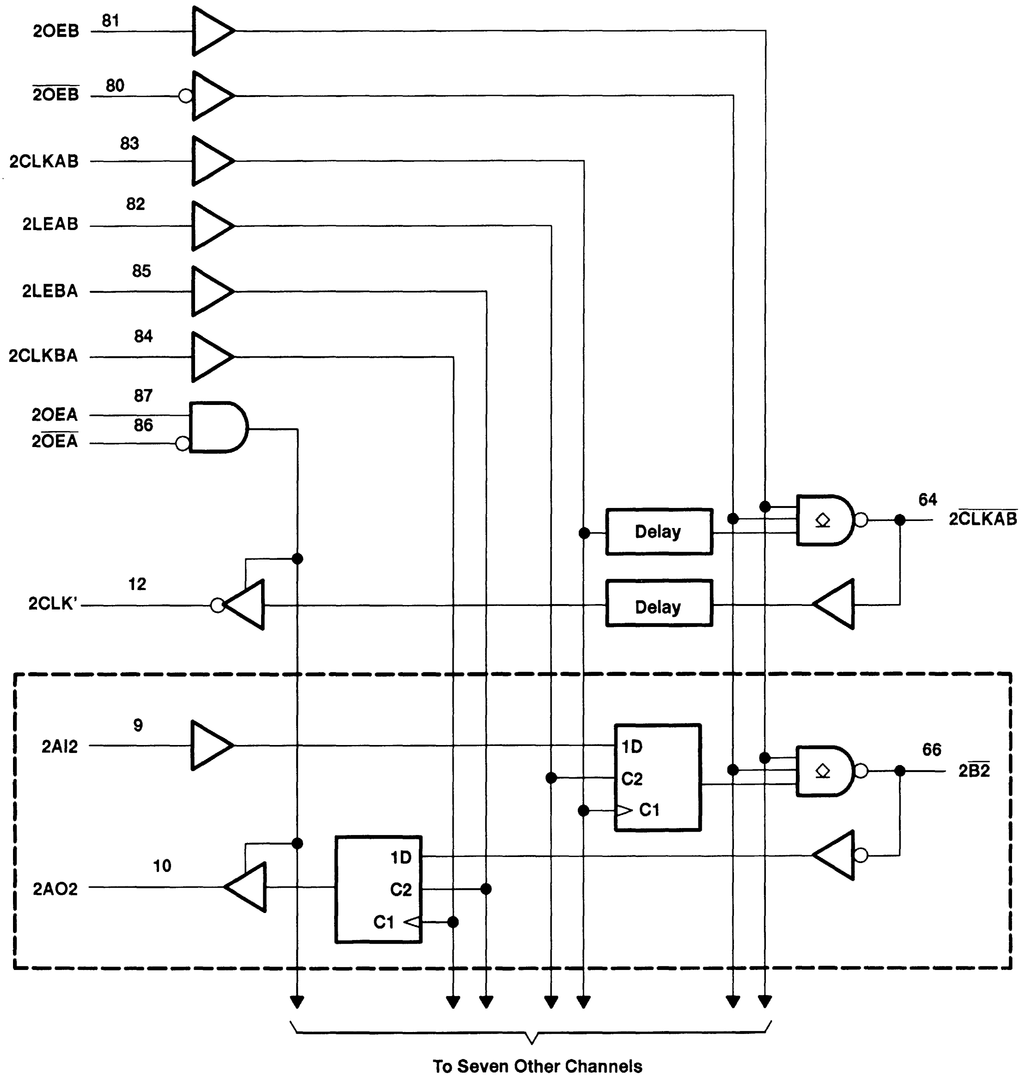
functional block diagram



PRODUCT PREVIEW

PRODUCT PREVIEW

functional block diagram (continued)



TRANSCEIVER FUNCTION TABLE

INPUTS				FUNCTION
$\overline{OEA}$	OEA	OEB	$\overline{OEB}$	
X	X	H	L	$\overline{A}$ data to B bus
L	H	X	X	$\overline{B}$ data to A bus
L	H	H	L	$\overline{A}$ data to B bus, $\overline{B}$ data to A bus
X	X	L	X	B-bus isolation
X	X	X	H	
H	X	X	X	A-bus isolation
X	L	X	X	

STORAGE-MODE TABLE

INPUTS		FUNCTION
LE	CLK	
H	X	Transparent
L	↑	Store data
L	L	Storage

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias V_{CC}, BG V_{CC}	−0.5 V to 7 V
Input voltage range, V_I: (except $\overline{B}$ port)	−1.2 V to 7 V
($\overline{B}$ port)	−1.2 V to 3.5 V
Input current range (except $\overline{B}$ port)	−40 mA to 5 mA
Voltage range applied to any $\overline{B}$ output in the disabled or power-off state	−0.5 V to 5.5 V
Voltage range applied to any output in the high state	−0.5 V to V_{CC}
Current applied to any single output in the low state: (A port)	48 mA
($\overline{B}$ port)	200 mA
Storage temperature range	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

PRODUCT PREVIEW

recommended operating conditions (see Note 1)

			MIN	NOM	MAX	UNIT
V_{CC}, BG V_{CC}	Supply voltage		4.75	5	5.25	V
BIAS V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	V
		Except $\overline{B}$ port	2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	V
		Except $\overline{B}$ port			0.8	
I_{IK}	Input clamp current				−18	mA
I_{OH}	High-level output current	A port			−3	mA
I_{OL}	Low-level output current	A port			24	mA
		$\overline{B}$ port			100	
T_A	Operating free-air temperature		−55		125	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

electrical characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	$\overline{B}$ port	V_{CC} = 4.75 V,	I_I = –18 mA			–1.2	V
	Except $\overline{B}$ port and AO port	V_{CC} = 4.75 V,	I_I = –40 mA			–0.5	V
V_{OH}	AO port	V_{CC} = 4.75 V	I_{OH} = –1 mA				V
			I_{OH} = –3 mA	2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.75 V,	I_{OL} = 24 mA		0.35	0.5	V
	B port	V_{CC} = 4.75 V	I_{OL} = 80 mA	0.75		1.1	
			I_{OL} = 100 mA			1.15	
I_I	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 5.25 V			50	µA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 2.7 V			50	µA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 0.5 V			–50	µA
	$\overline{B}$ port†	V_{CC} = 5.25 V,	V_I = 0.75 V			–100	
I_{hold}	AI port	V_{CC} = 5.25 V,	V_I = 2 V	–100			µA
	AI port	V_{CC} = 5.25 V,	V_I = 0.5 V	100			
I_{OZH}	AO port	V_{CC} = 5.25 V,	V_O = 2.7 V			50	µA
I_{OZL}	AO port	V_{CC} = 5.25 V,	V_O = 0.5 V			–50	µA
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.25 V,	V_O = 2.1 V			100	µA
I_{OS}§	A port	V_{CC} = 5.25 V,	V_O = 0	– 30		–150	mA
I_{CC}	A port to $\overline{B}$ port	V_{CC} = 5.25 V,	I_O = 0		25		mA
	$\overline{B}$ port to A port				60		
	Outputs disabled						
C_i		V_I = V_{CC} or GND				5	pF
C_o	A port	V_O = V_{CC} or GND					pF
C_{io}	B port per P1194.0	V_{CC} = 0 V to 4.75 V				6	pF
		V_{CC} = 4.75 V to 5.25 V				5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

live insertion specifications over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.75 V,	V_B = 0 to 2 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V			450	µA
		V_{CC} = 4.25 to 5.25 V					10	
V_O	$\overline{B}$ port	V_{CC} = 0,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V		1.62		2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0,	V_B = 1 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	–1			µA
		V_{CC} = 0 to 5.25 V,	OEB = 0 to 0.8 V				100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V				100	

PRODUCT PREVIEW

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

			MIN	TYP	MAX	UNIT
f_{clock}	Clock frequency					MHz
t_w	Pulse duration,	LE high				ns
		CLK high or low				
t_{su}	Setup time	A or $\overline{B}$ before LE			2	ns
		A or $\overline{B}$ before CLK↑			2	
t_h	Hold time	A or $\overline{B}$ after LE			1	ns
		A or $\overline{B}$ after CLK↑			1	

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

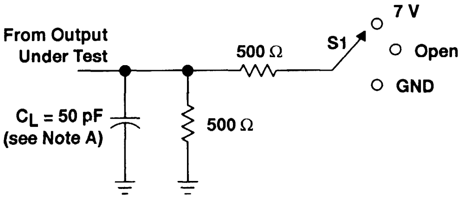
PARAMETER	FROM (INPUT)	TO (OUTPUT)	MIN	TYP†	MAX	UNIT
t_{PLH}	A	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PLH}	LEAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	CLKAB	$\overline{B}$	2.4		6.5	ns
t_{PHL}			2.2		6.5	
t_{PLH}	2CLKAB	2$\overline{CLKAB}$	3.9		10.2	ns
t_{PHL}			3.8		10.1	
t_{PLH}	LEBA	A			6	ns
t_{PHL}					6	
t_{PLH}	CLKBA	A			6	ns
t_{PHL}					6	
t_{PLH}	$\overline{B}$	A			5	ns
t_{PHL}					5	
t_{PLH}	2CLKAB	CLK'	4.3		12.7	ns
t_{PHL}			4.5		12.4	
t_{PLH}	OEB or $\overline{OEB}$	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PZH}	OEA or $\overline{OEA}$	A			5	ns
t_{PZL}					5	
t_{PHZ}	OEA or $\overline{OEA}$	A			5	ns
t_{PLZ}					5	
$t_{sk(p)}$‡	Skew for any single channel $\lvert t_{PHL} - t_{PLH} \rvert$	A to $\overline{B}$ or $\overline{B}$ to A		0.5		ns
$t_{sk(o)}$‡	Skew between drivers in the same package	A to $\overline{B}$ or $\overline{B}$ to A		1		ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)		1	2	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection			1		ns

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ Skew values are applicable for through mode only.

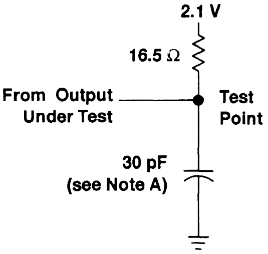
PRODUCT PREVIEW

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION

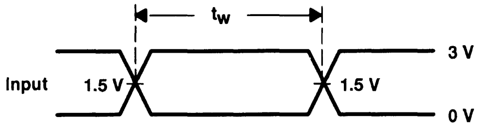


LOAD CIRCUIT FOR A OUTPUTS

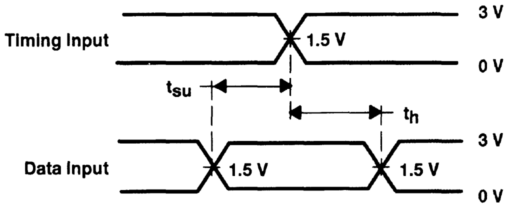


LOAD CIRCUIT FOR B OUTPUTS

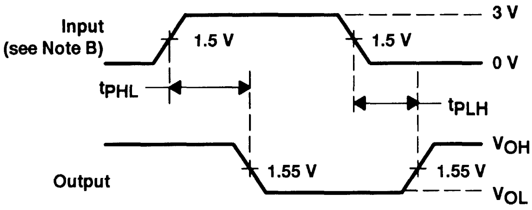
TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND



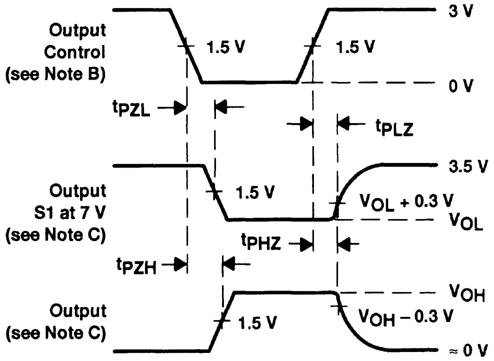
VOLTAGE WAVEFORMS
PULSE DURATION

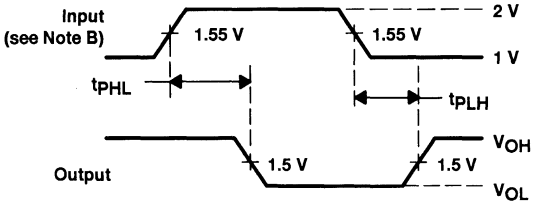


VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES



VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)





VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A port)

NOTES: A. C$_L$ includes probe and jig capacitance.

B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.

C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.

D. The outputs are measured one at a time with one transition per measurement.

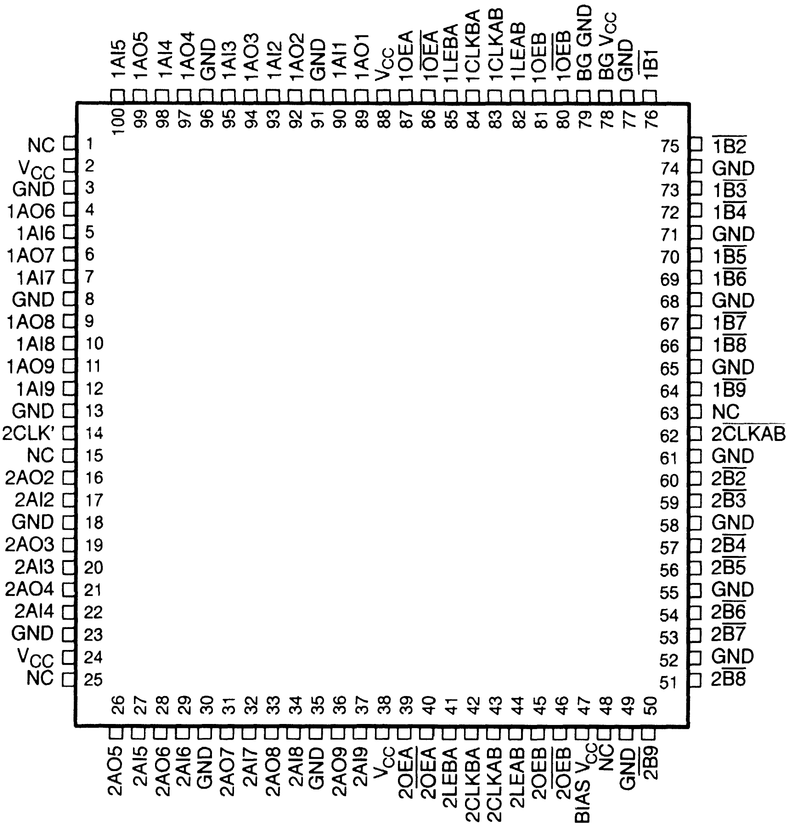
Figure 1. Load Circuit and Voltage Waveforms

SN74FB1651
17-BIT TTL/BTL UNIVERSAL STORAGE TRANSCEIVER WITH BUFFERED CLOCK LINE

SCBS177A – OCTOBER 1993 – REVISED JANUARY 1994

- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{B}$ Port**
- **Open-Collector $\overline{B}$-Port Outputs Sink 100 mA**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **Packaged in the High-Power Shrink Quad Flat Package (PCA) With 0.5-mm Pin Pitch**
- **$\overline{B}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL-Input Structures Incorporate Active Clamping and Bus-Hold Networks**

PCA PACKAGE
(TOP VIEW)



NC – No internal connection

PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description

The SN74FB1651 contains an 8-bit and a 9-bit transceiver with a buffered clock. The clock and the transceivers are designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{B}$ port operates at BTL signal levels. The open-collector $\overline{B}$ ports are specified to sink 100 mA. Two output enables, OEB and $\overline{\text{OEB}}$, are provided for the $\overline{B}$ outputs. When OEB is low, $\overline{\text{OEB}}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{B}$ port is turned off.

The A port operates at TTL signal levels. The A outputs reflect the inverse of the data at the $\overline{B}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Active bus-hold circuitry is provided to hold unused or floating TTL inputs at a valid logic state.

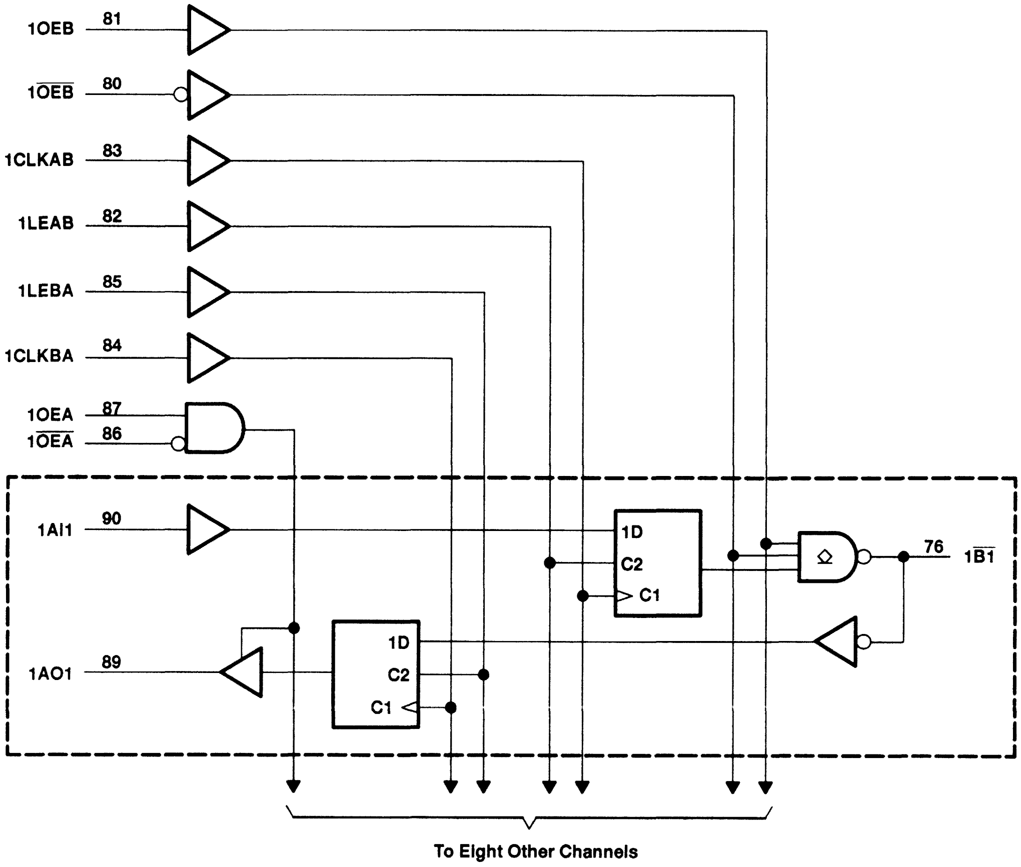
BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN74FB1651 is characterized for operation from 0°C to 70°C.

PRODUCT PREVIEW

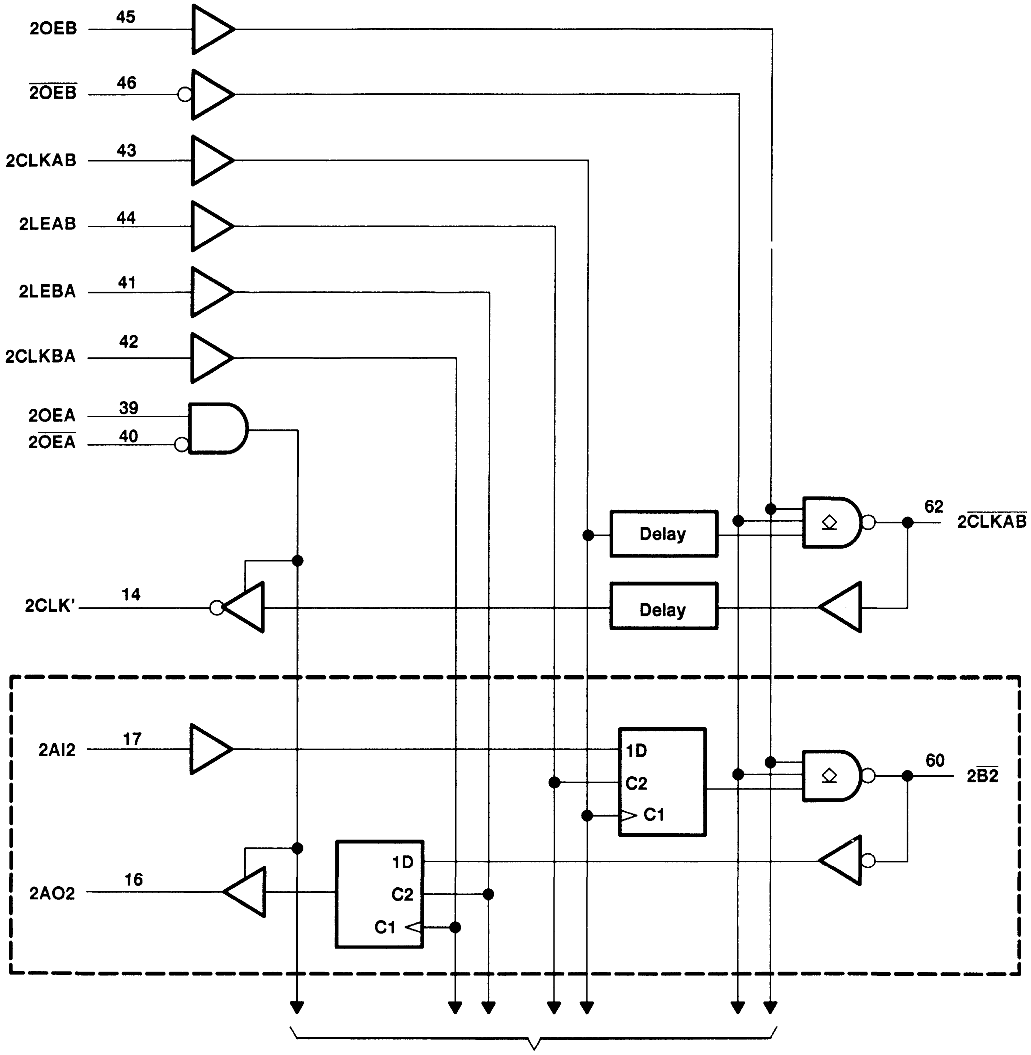
functional block diagram



PRODUCT PREVIEW

PRODUCT PREVIEW

functional block diagram (continued)



To Seven Other Channels

TRANSCEIVER FUNCTION TABLE

INPUTS				FUNCTION
$\overline{\text{OEA}}$	OEA	OEB	$\overline{\text{OEB}}$	
X	X	H	L	$\overline{\text{A}}$ data to B bus
L	H	X	X	$\overline{\text{B}}$ data to A bus
L	H	H	L	$\overline{\text{A}}$ data to B bus, $\overline{\text{B}}$ data to A bus
X	X	L	X	B-bus isolation
X	X	X	H	
H	X	X	X	A-bus isolation
X	L	X	X	

STORAGE-MODE TABLE

INPUTS		FUNCTION
LE	CLK	
H	X	Transparent
L	↑	Store data
L	L	Storage

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias VCC, BG V_{CC}	−0.5 V to 7 V
Input voltage range, V_I: (except $\overline{\text{B}}$ port)	−1.2 V to 7 V
($\overline{\text{B}}$ port)	−1.2 V to 3.5 V
Input current range (except $\overline{\text{B}}$ port)	−40 mA to 5 mA
Voltage range applied to any $\overline{\text{B}}$ output in the disabled or power-off state	−0.5 V to 5.5 V
Voltage range applied to any output in the high state	−0.5 V to V_{CC}
Current applied to any single output in the low state: (A port)	48 mA
($\overline{\text{B}}$ port)	200 mA
Storage temperature range	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			MIN	NOM	MAX	UNIT
V_{CC}, BG V_{CC}	Supply voltage		4.75	5	5.25	V
BIAS V_{CC}	Supply voltage		4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{\text{B}}$ port	1.62		2.3	V
		Except $\overline{\text{B}}$ port	2			
V_{IL}	Low-level input voltage	$\overline{\text{B}}$ port	0.75		1.47	V
		Except $\overline{\text{B}}$ port			0.8	
I_{IK}	Input clamp current				−18	mA
I_{OH}	High-level output current	A port			−3	mA
I_{OL}	Low-level output current	A port			24	mA
		$\overline{\text{B}}$ port			100	
T_A	Operating free-air temperature		0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW

PRODUCT PREVIEW

electrical characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	$\overline{B}$ port	V_{CC} = 4.75 V,	I_I = –18 mA			–1.2	V
	Except $\overline{B}$ port and AO port	V_{CC} = 4.75 V,	I_I = –40 mA			–0.5	V
V_{OH}	AO port	V_{CC} = 4.75 V	I_{OH} = –1 mA				V
			I_{OH} = –3 mA	2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.75 V,	I_{OL} = 24 mA		0.35	0.5	V
	B port	V_{CC} = 4.75 V	I_{OL} = 80 mA	0.75		1.1	
			I_{OL} = 100 mA			1.15	
I_I	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 5.25 V			50	µA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 2.7 V			50	µA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 0.5 V			–50	µA
	$\overline{B}$ port†	V_{CC} = 5.25 V,	V_I = 0.75 V			–100	
I_{hold}	AI port	V_{CC} = 5.25 V,	V_I = 2 V	–100			µA
	AI port	V_{CC} = 5.25 V,	V_I = 0.5 V	100			
I_{OZH}	AO port	V_{CC} = 5.25 V,	V_O = 2.7 V			50	µA
I_{OZL}	AO port	V_{CC} = 5.25 V,	V_O = 0.5 V			–50	µA
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.25 V,	V_O = 2.1 V			100	µA
I_{OS}§	A port	V_{CC} = 5.25 V,	V_O = 0	– 30		–150	mA
I_{CC}	A port to $\overline{B}$ port	V_{CC} = 5.25 V,	I_O = 0		25		mA
	$\overline{B}$ port to A port				60		
	Outputs disabled						
C_i		V_I = V_{CC} or GND				5	pF
C_o	A port	V_O = V_{CC} or GND					pF
C_{io}	B port per P1194.0	V_{CC} = 0 to 4.75 V				6	pF
		V_{CC} = 4.75 V to 5.25 V				5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

live insertion specifications over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS			MIN	TYP	MAX	UNIT
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.75 V,	V_B = 0 to 2 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V			450	µA
		V_{CC} = 4.25 to 5.25 V					10	
V_O	$\overline{B}$ port	V_{CC} = 0,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V		1.62		2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0,	V_B = 1 V,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	–1			µA
		V_{CC} = 0 to 5.25 V,	OEB = 0 to 0.8 V				100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V				100	

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

			MIN	TYP	MAX	UNIT
f_{clock}	Clock frequency					MHz
t_w	Pulse duration,	LE high				ns
		CLK high or low				
t_{su}	Setup time	A or $\overline{B}$ before LE			2	ns
		A or $\overline{B}$ before CLK↑			2	
t_h	Hold time	A or $\overline{B}$ after LE			1	ns
		A or $\overline{B}$ after CLK↑			1	

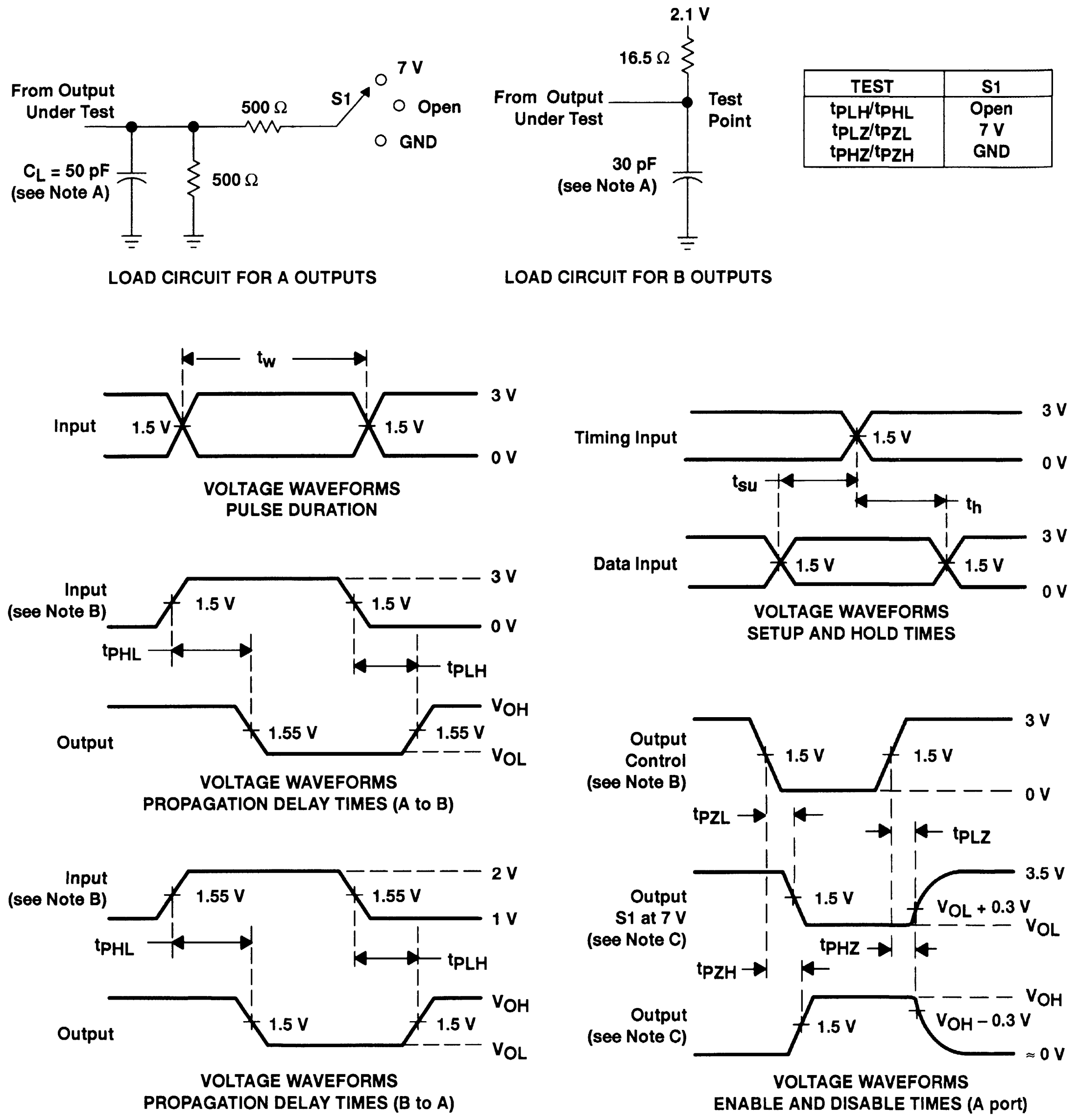
switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	MIN	TYP†	MAX	UNIT
t_{PLH}	A	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PLH}	LEAB	$\overline{B}$			6	ns
t_{PHL}					6	
t_{PLH}	CLKAB	$\overline{B}$	2.4		6.5	ns
t_{PHL}			2.2		6.5	
t_{PLH}	2CLKAB	2$\overline{CLKAB}$	3.9		10.2	ns
t_{PHL}			3.8		10.1	
t_{PLH}	LEBA	A			6	ns
t_{PHL}					6	
t_{PLH}	CLKBA	A			6	ns
t_{PHL}					6	
t_{PLH}	$\overline{B}$	A			5	ns
t_{PHL}					5	
t_{PLH}	2CLKAB	CLK'	4.3		12.7	ns
t_{PHL}			4.5		12.4	
t_{PLH}	OEB or $\overline{OEB}$	$\overline{B}$			5	ns
t_{PHL}					5	
t_{PZH}	OEA or $\overline{OEA}$	A			5	ns
t_{PZL}					5	
t_{PHZ}	OEA or $\overline{OEA}$	A			5	ns
t_{PLZ}					5	
$t_{sk(p)}$‡	Skew for any single channel | t_{PHL} – t_{PLH} |	A to $\overline{B}$ or $\overline{B}$ to A		0.5		ns
$t_{sk(o)}$‡	Skew between drivers in the same package	A to $\overline{B}$ or $\overline{B}$ to A		1		ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)		1	2	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection			1		ns

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ Skew values are applicable for through mode only.

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION



TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND

LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A to B)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A port)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B to A)

NOTES: A. C_L includes probe and jig capacitance.

B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.

C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.

D. The outputs are measured one at a time with one transition per measurement.

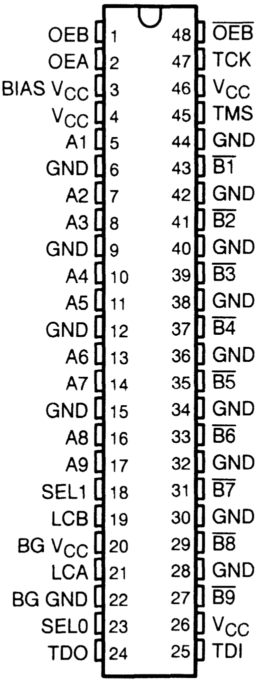
Figure 1. Load Circuit and Voltage Waveforms

SN54FB2031, SN74FB2031
9-BIT TTL/BTL ADDRESS/DATA TRANSCEIVERS

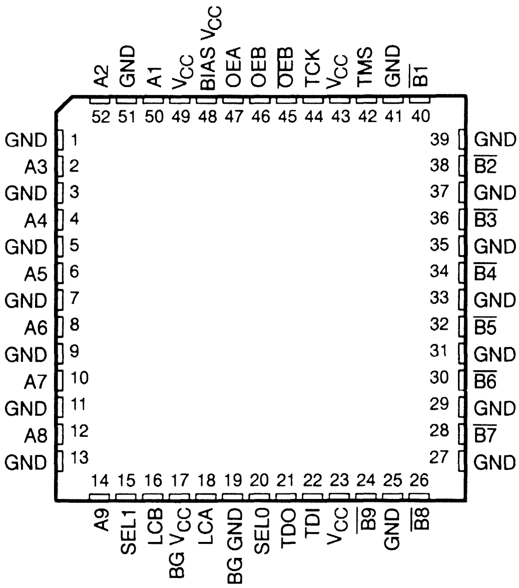
SCBS176A, NOVEMBER 1991 – REVISED JANUARY 1994

- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Minimum $\overline{\text{B}}$-Port Edge Rate = 2 ns**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **BIAS V_{CC} Pin Minimizes Signal Distortion During Live Insertion/Withdrawal**
- **Available in Plastic Quad Flatpack (RC) and Ceramic Flatpack (WD) Packages**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL-Input Structures Incorporate Active Clamping Networks to Aid in Line Termination**

SN54FB2031 . . . WD PACKAGE
(TOP VIEW)



SN74FB2031 . . . RC PACKAGE
(TOP VIEW)



PRODUCT PREVIEW

description

The ′FB2031 is a 9-bit transceiver designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{\text{B}}$ port operates at BTL-signal levels. The open-collector $\overline{\text{B}}$ ports are specified to sink 100 mA and have minimum output edge rates of 2 ns. Two output enables, OEB and $\overline{\text{OEB}}$, are provided for the $\overline{\text{B}}$ outputs. When OEB is low, $\overline{\text{OEB}}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{\text{B}}$ port is turned off.

The A port operates at TTL-signal levels. The A outputs reflect the inverse of the data at the $\overline{\text{B}}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description (continued)

Pins are allocated for the four-wire IEEE 1149.1 (JTAG) test bus, which will be implemented in a future version of the 'FB2031. Currently TMS and TCK are not connected and TDI is shorted to TDO.

BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN54FB2031 is characterized for operation over the full military temperature range of –55°C to 125°C. The SN74FB2031 is characterized for operation from 0°C to 70°C.

TRANSCEIVER FUNCTION TABLE

INPUTS			FUNCTION
OEA	OEB	$\overline{\text{OEB}}$	
L	H	L	$\overline{\text{A}}$ data to B bus
H H	L X	X H	$\overline{\text{B}}$ data to A bus
H	H	L	$\overline{\text{A}}$ data to B bus, $\overline{\text{B}}$ data to A bus
L L	L X	X H	Isolation

STORAGE MODE TABLE

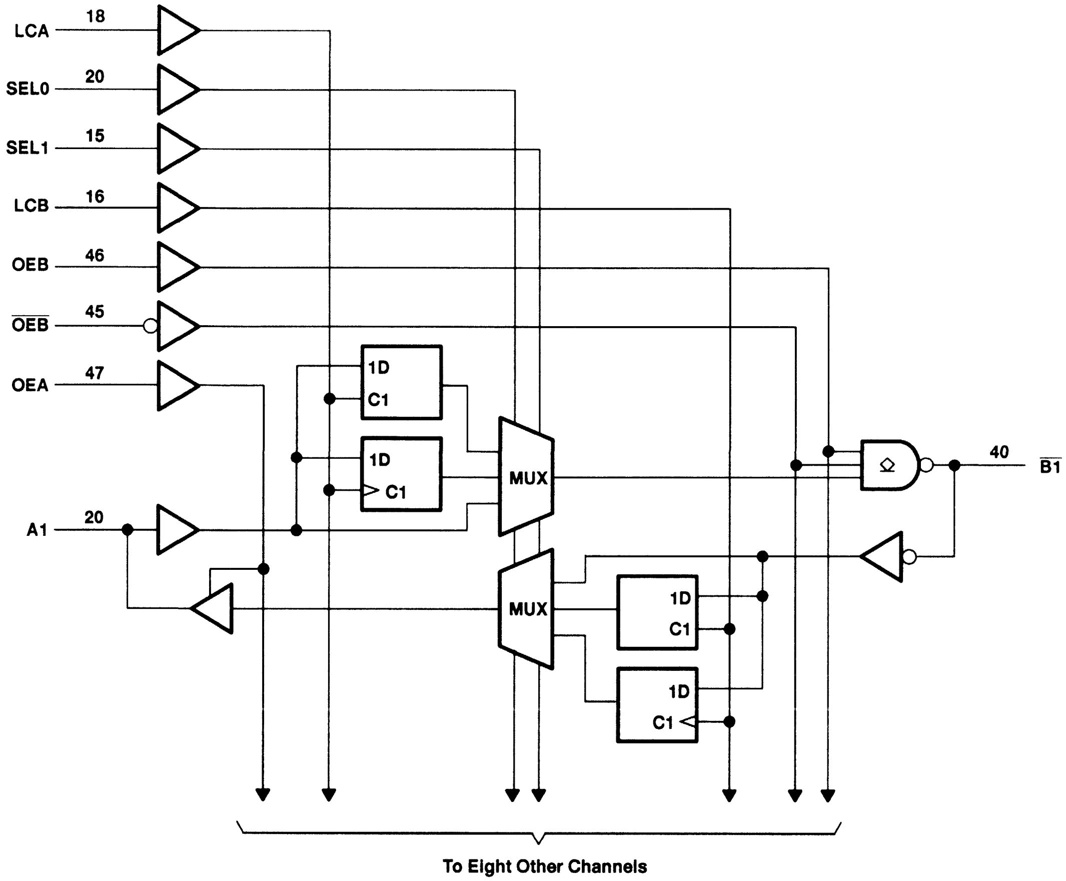
LCA, LCB	RESULT
0	Transparent
1	Latches latched
↑	Flip-flops triggered

SELECT FUNCTION TABLE

SEL1	SEL0	MUX A→B	MUX B→A
0	0	Latch	Latch
0	1	Thru	Thru
1	0	Flip-flop	Flip-flop
1	1	Flip-flop	Latch

PRODUCT PREVIEW

functional block diagram



Pin numbers shown are for the RC package.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, BG V_{CC}, Bias V_{CC}	–0.5 V to 7 V
Input voltage range, V_I (except $\overline{B}$ port)	–1.2 V to 7 V
V_I ($\overline{B}$ port)	–1.2 V to 3.5 V
Input current range (except $\overline{B}$ port)	–40 mA to 5 mA
Voltage range applied to any $\overline{B}$ output in the disabled or power-off state	–0.5 V to 5.5 V
Voltage range applied to any output in the high state	–0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{B}$ port	200 mA
Maximum power dissipation at T_A = 55°C (in still air): RC package	1.4 W
Storage temperature range	–65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

PRODUCT PREVIEW

PRODUCT PREVIEW

recommended operating conditions (see Note 1)

			SN54FB2031			SN74FB2031			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}, BIAS V_{CC}, BG V_{CC}	Supply voltage		4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	1.62		2.3	V
		Except $\overline{B}$ port	2			2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	0.75		1.47	V
		Except $\overline{B}$ port			0.8			0.8	
I_{IK}	Input clamp current				–18			–18	mA
I_{OH}	High-level output current	A port			–3			– 3	mA
I_{OL}	Low-level output current	A port			24			24	mA
		$\overline{B}$ port			100			100	
T_A	Operating free-air temperature		–55		125	0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS		SN54FB2031			SN74FB2031			UNIT
				MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}	$\overline{B}$ port	V_{CC} = 4.5 V,	I_I = –18 mA			–1.2			–1.2	V
	Except $\overline{B}$ port	V_{CC} = 4.5 V,	I_I = –40 mA			–0.5			–0.5	
V_{OH}	A port	V_{CC} = 4.5 V	I_{OH} = – 1 mA							V
			I_{OH} = –3 mA	2.5	3.3		2.5	3.3		
V_{OL}	A port	V_{CC} = 4.5 V	I_{OL} = 20 mA							V
			I_{OL} = 24 mA		0.35	0.5		0.35	0.5	
	$\overline{B}$ port	V_{CC} = 4.5 V	I_{OL} = 80 mA	0.75		1.1	0.75		1.1	
			I_{OL} = 100 mA							
I_I	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 5.5 V			50			50	μA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 2.7 V			50			50	μA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 0.5 V			–50			–50	μA
	$\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 0.75 V			–100			–100	
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.5 V,	V_O = 2.1 V			100			100	μA
I_{OS}§	A port	V_{CC} = 5.5 V,	V_O = 0	– 30		–150	– 30		–150	mA
I_{CC}	A port to $\overline{B}$ port	V_{CC} = 5.5 V,	I_O = 0		25			25		mA
	$\overline{B}$ port to A port				60			60		
	Outputs disabled									
C_i		V_I = V_{CC} or GND				5			5	pF
C_o	A port	V_O = V_{CC} or GND								pF
C_{io}	$\overline{B}$ port per P1194.0	V_{CC} = 0 to 4.5 V				6			6	pF
		V_{CC} = 4.5 V to 5.5 V				5			5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, T_A = 25°C			SN54FB2031		SN74FB2031		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{PLH}	A (thru mode)	$\overline{B}$			5					ns
t_{PHL}					5					
t_{PLH}	A (transparent)	$\overline{B}$			6					ns
t_{PHL}					6					
t_{PLH}	LCA	$\overline{B}$			7					ns
t_{PHL}					7					
t_{PLH}	LCB	A			9					ns
t_{PHL}					9					
t_{PLH}	SEL1 or SEL0	A			5.5					ns
t_{PHL}					5.5					
t_{PLH}	SEL1 or SEL0	$\overline{B}$			7					ns
t_{PHL}					7					
t_{PLH}	$\overline{B}$ (thru mode)	A			6					ns
t_{PHL}					6					
t_{PLH}	$\overline{B}$ (transparent)	A			7					ns
t_{PHL}					7					
t_{PLH}	OEB or $\overline{OEB}$	$\overline{B}$			5.5					ns
t_{PHL}					5.5					
t_{PZH}	OEA	A			4					ns
t_{PZL}					4					
t_{PHZ}	OEA	A			5					ns
t_{PLZ}					5					
$t_{sk(p)}$	Skew for any single channel $\lvert t_{PHL} - t_{PLH} \rvert$	A to $\overline{B}$ or $\overline{B}$ to A		0.5						ns
$t_{sk(o)}$	Skew between drivers in the same package	A to $\overline{B}$ or $\overline{B}$ to A		1						ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)			2		1	3	1	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection					1		1		ns

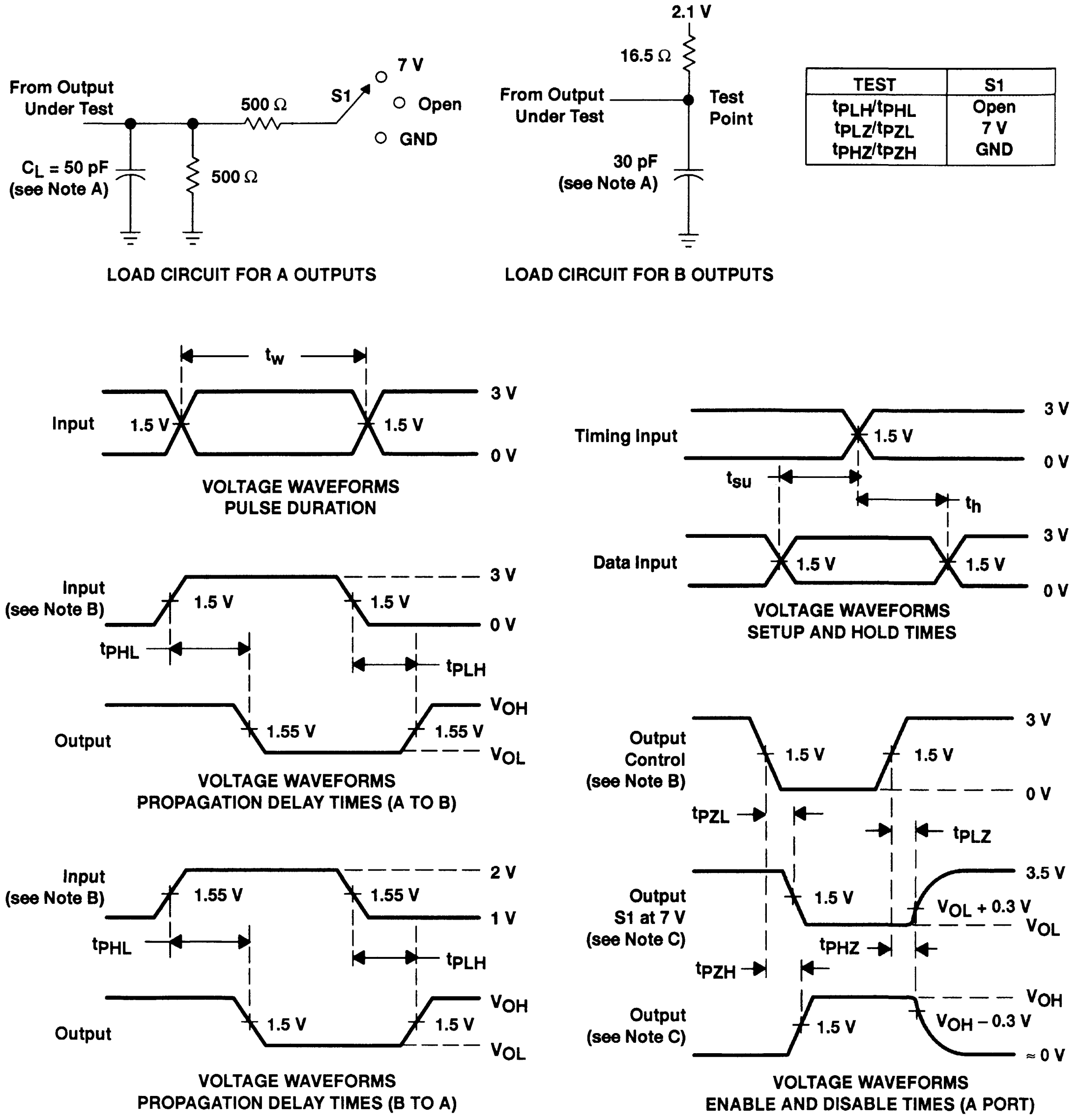
live-insertion characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		SN54FB2031		SN74FB2031		UNIT
				MIN	MAX	MIN	MAX	
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.5 V	V_B = 0 to 2 V, BIAS V_{CC} = 4.5 V to 5.5 V		450		450	µA
		V_{CC} = 4.5 V to 5.5 V			10		10	
V_O	$\overline{B}$ port	V_{CC} = 0,	V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	1.62	2.1	1.62	2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0 ,	V_B = 1 V, BIAS V_{CC} = 4.5 V to 5.5 V	−1		−1		µA
		V_{CC} = 0 to 5.5 V,	OEB = 0 to 0.8 V		100		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100		100	

PRODUCT PREVIEW

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION



TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND

LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A PORT)

NOTES:
A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

Figure 1. Load Circuit and Voltage Waveforms

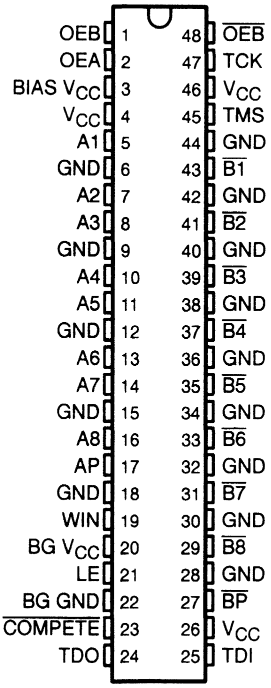
SN54FB2032, SN74FB2032
9-BIT TTL/BTL COMPETITION TRANSCEIVERS

SCBS175A, NOVEMBER 1991 – REVISED JANUARY 1994

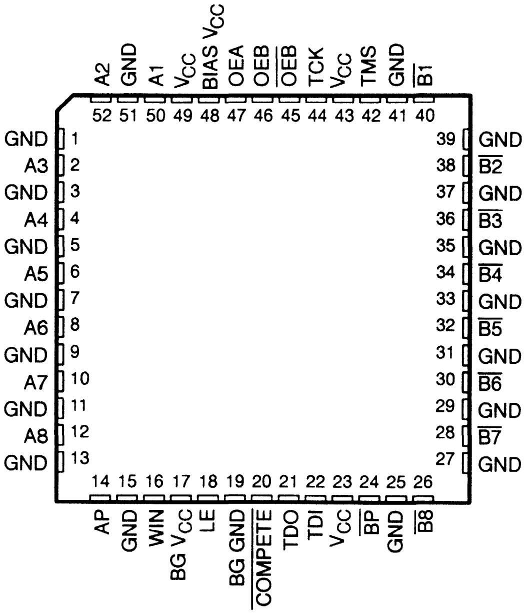
- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\mathrm{B}}$ Port**
- **Open-Collector $\overline{\mathrm{B}}$-Port Outputs Sink 100 mA**
- **Minimum $\overline{\mathrm{B}}$-Port Edge Rate = 2 ns**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **BIAS V_{CC} Pin Minimizes Signal Distortion During Live Insertion/Withdrawal**
- **Available in Plastic Quad Flatpack (RC) and Ceramic Flatpack (WD) Packages**
- **$\overline{\mathrm{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL-Input Structures Incorporate Active Clamping Networks to Aid in Line Termination**

PRODUCT PREVIEW

SN54FB2032 . . . WD PACKAGE
(TOP VIEW)



SN74FB2032 . . . RC PACKAGE
(TOP VIEW)



description

The ′FB2032 is a 9-bit transceiver designed to translate signals between TTL and backplane transceiver logic (BTL) environments and to perform bus arbitration. It is specifically designed to be compatible with IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{\mathrm{B}}$ port operates at BTL-signal levels. The open-collector $\overline{\mathrm{B}}$ ports are specified to sink 100 mA and have minimum output edge rates of 2 ns. Two output enables, OEB and $\overline{\mathrm{OEB}}$, are provided for the $\overline{\mathrm{B}}$ outputs. When OEB is low, $\overline{\mathrm{OEB}}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{\mathrm{B}}$ port is turned off.

The A port operates at TTL-signal levels. The A outputs reflect the inverse of the data at the $\overline{\mathrm{B}}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description (continued)

The A-port data can be latched by taking the latch enable (LE) high. When LE is low, the latches are transparent.

The Futurebus+ protocol logic can be activated by taking $\overline{\text{COMPETE}}$ low. The module (device) then compares its A data (arbitration number) against the A data of another identical module also connected to the $\overline{\text{B}}$ arbitration bus and sets WIN high if the A data is greater than the A data of the other module (i.e., has higher priority). A8 and $\overline{\text{B8}}$ are the most significant bits, and A1 and $\overline{\text{B1}}$ are the least significant bits. If OEB is high and $\overline{\text{OEB}}$ is low during this operation and the A bus of the first module wins priority, it will assert its arbitration number on the $\overline{\text{B}}$-arbitration bus.

AP and $\overline{\text{BP}}$ are the bus parity bits. The winning module may assert $\overline{\text{BP}}$ low if its parity bit (AP) is high.

In a typical operating sequence, a Futurebus+ arbitration controller will latch its arbitration number into the A port and wait for the results of a competition. When the competition is complete and if the controller's arbitration number did not win, the controller will read back the current value of the $\overline{\text{B}}$ bus (by taking OEA high) and determine the winning arbitration number. This allows the module to change its arbitration number for the next competition cycle, if desired.

Pins are allocated for the four-wire IEEE 1149.1 (JTAG) test bus, which will be implemented in a future version of the ′FB2032. Currently TMS and TCK are not connected and TDI is shorted to TDO.

BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

BG V_{CC} and BG GND are the supply inputs for the bias generator.

The SN54FB2032 is characterized for operation over the full military temperature range of −55°C to 125°C. The SN74FB2032 is characterized for operation from 0°C to 70°C.

TRANSCEIVER FUNCTION TABLE

INPUTS			FUNCTION
OEA	OEB	$\overline{\text{OEB}}$	
L	H	L	$\overline{\text{A}}$ data to B bus
H H	L X	X H	$\overline{\text{B}}$ data to A bus
H	H	L	$\overline{\text{A}}$ data to B bus, $\overline{\text{B}}$ data to A bus
L L	L X	X H	Isolation

STORAGE MODE TABLE

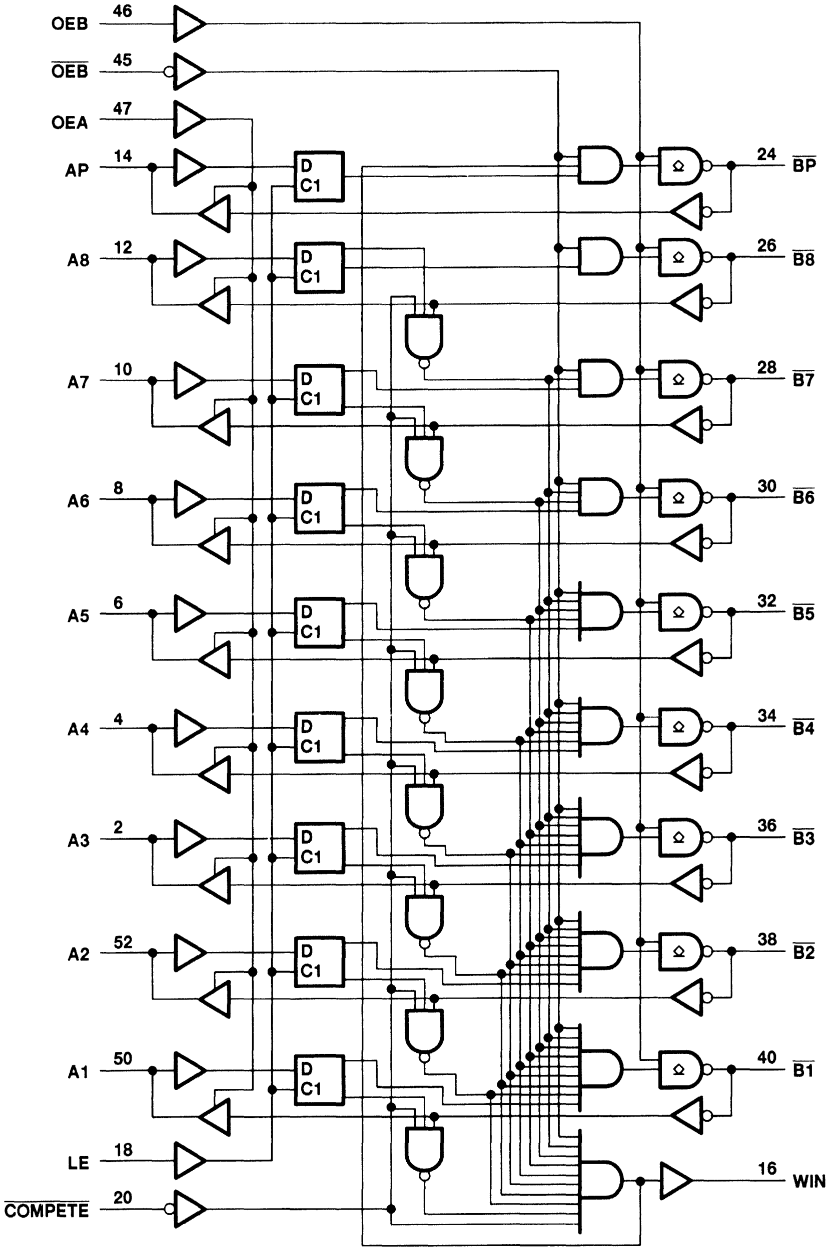
LCA, LCB	RESULT
0	Transparent
1	Latches latched
↑	Flip-flops triggered

SELECT FUNCTION TABLE

SEL1	SEL0	MUX A→B	MUX B→A
0	0	Latch	Latch
0	1	Thru	Thru
1	0	Flip-flop	Flip-flop
1	1	Flip-flop	Latch

PRODUCT PREVIEW

functional block diagram



Pin numbers shown are for the RC package.

PRODUCT PREVIEW

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}	–0.5 V to 7 V
Input voltage range, V_I (except $\overline{\text{BP}}$, $\overline{\text{B}}$ port)	–1.2 V to 7 V
V_I ($\overline{\text{BP}}$, $\overline{\text{B}}$ port)	–1.2 V to 3.5 V
Input current range (except $\overline{\text{B}}$ port)	–40 mA to 5 mA
Voltage range applied to any $\overline{\text{B}}$ output in the disabled or power-off state	–0.5 V to 5.5 V
Voltage range applied to any output in the high state	–0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{\text{B}}$ port	200 mA
Maximum power dissipation at T_A = 55°C (in still air): RC package	1.4 W
Storage temperature range	–65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			SN54FB2032			SN74FB2032			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}, BIAS V_{CC}, BG V_{CC}	Supply voltage		4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	1.62		2.3	1.62		2.3	V
		Except $\overline{\text{B}}$ port, $\overline{\text{BP}}$	2			2			
V_{IL}	Low-level input voltage	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	0.75		1.47	0.75		1.47	V
		Except $\overline{\text{B}}$ port, $\overline{\text{BP}}$			0.8			0.8	
I_{IK}	Input clamp current				–18			–18	mA
I_{OH}	High-level output current	AP, WIN, A port			-3			-3	mA
I_{OL}	Low-level output current	AP, WIN, A port			24			24	mA
		$\overline{\text{BP}}$, $\overline{\text{B}}$ port			100			100	
T_A	Operating free-air temperature		–55		125	0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS		SN54FB2032			SN74FB2032			UNIT
				MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 4.5 V,	I_I = –18 mA			–1.2			–1.2	V
	Except $\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 4.5 V,	I_I = –40 mA			–0.5			–0.5	
V_{OH}	AP, WIN, A port	V_{CC} = 4.5 V	I_{OH} = – 1 mA							V
			I_{OH} = –3 mA	2.5	3.3		2.5	3.3		
V_{OL}	AP, WIN, A port	V_{CC} = 4.5 V	I_{OL} = 20 mA							V
			I_{OL} = 24 mA		0.35	0.5		0.35	0.5	
	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 4.5 V	I_{OL} = 80 mA	0.75		1.1	0.75		1.1	
			I_{OL} = 100 mA							
I_I	Except $\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 5.5 V,	V_I = 5.5 V			50			50	µA
I_{IH}‡	Except $\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 5.5 V,	V_I = 2.7 V			50			50	µA
I_{IL}‡	Except $\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 5.5 V,	V_I = 0.5 V			–50			–50	µA
	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 5.5 V,	V_I = 0.75 V			–100			–100	
I_{OH}	$\overline{\text{BP}}$, $\overline{\text{B}}$ port	V_{CC} = 0 to 5.5 V,	V_O = 2.1 V			100			100	µA
I_{OS}§	AP, WIN, A port	V_{CC} = 5.5 V,	V_O = 0	– 30		–150	– 30		–150	mA
I_{CC}	A port to $\overline{\text{B}}$ port	V_{CC} = 5.5 V,	I_O = 0		25			25		mA
	$\overline{\text{B}}$ port to A port				60			60		
	Outputs disabled									
C_i		V_I = V_{CC} or GND				5			5	pF
C_o	A port	V_O = V_{CC} or GND								pF
C_{io}	$\overline{\text{B}}$ port per P1194.0	V_{CC} = 0 to 4.5 V				6			6	pF
		V_{CC} = 4.5 V to 5.5 V				5			5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

PRODUCT PREVIEW

PRODUCT PREVIEW

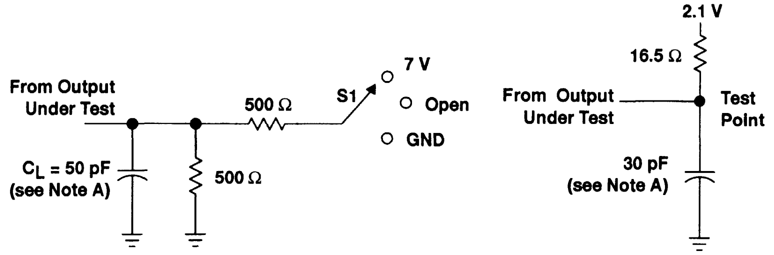
switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, T_A = 25°C			SN54FB2032		SN74FB2032		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{PLH}	A or AP	$\overline{B}$ or $\overline{BP}$					8		8	ns
t_{PHL}							8		8	
t_{PLH}	A	$\overline{B}_{n-1}$					9		9	ns
t_{PHL}							9		9	
t_{PLH}	A	$\overline{BP}$					10		10	ns
t_{PHL}							10		10	
t_{PLH}	$\overline{B}$	$\overline{B}_{n-1}$					9		9	ns
t_{PHL}							9		9	
t_{PLH}	LE	$\overline{B}$					7.5		7.5	ns
t_{PHL}							7.5		7.5	
t_{PLH}	LE	$\overline{BP}$					7.5		7.5	ns
t_{PHL}							7.5		7.5	
t_{PLH}	$\overline{B}$ or $\overline{BP}$	A or AP					7.5		7.5	ns
t_{PHL}							7.5		7.5	
t_{PLH}	$\overline{B}$	WIN					8.5		8.5	ns
t_{PHL}							8.5		8.5	
t_{PLH}	A	WIN					7.6		7.6	ns
t_{PHL}							7.6		7.6	
t_{PLH}	LE	WIN					7		7	ns
t_{PHL}							7		7	
t_{PLH}	$\overline{COMPETE}$	WIN					5.5		5.5	ns
t_{PHL}							5.5		5.5	
t_{PLH}	$\overline{OEB}$	WIN					6		6	ns
t_{PHL}							6		6	
t_{PLH}	$\overline{COMPETE}$	$\overline{B}$					7.5		7.5	ns
t_{PHL}							7.5		7.5	
t_{PLH}	$\overline{COMPETE}$	$\overline{BP}$					6.5		6.5	ns
t_{PHL}							6.5		6.5	
t_{PLH}	OEB	$\overline{B}$					6.5		6.5	ns
t_{PHL}							6.5		6.5	
t_{PLH}	$\overline{OEB}$	$\overline{B}$					6.5		6.5	ns
t_{PHL}							6.5		6.5	
t_{PZH}	OEA	A					5.5		5.5	ns
t_{PZL}							5.5		5.5	
t_{PHZ}	OEA	A					7		7	ns
t_{PLZ}							7		7	
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)		2			1	3	1	3	ns
t_{PR}	$\overline{B}$-port input pulse rejection						1		1	ns

live-insertion characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		SN54FB2032		SN74FB2032		UNIT
				MIN	MAX	MIN	MAX	
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.5 V	V_B = 0 to 2 V, BIAS V_{CC} = 4.5 V to 5.5 V		450		450	µA
		V_{CC} = 4.5 V to 5.5 V			10		10	
V_O	$\overline{B}$ port	V_{CC} = 0,	BIAS V_{CC} = 4.5 V to 5.5 V	1.62	2.1	1.62	2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0,	V_B = 1 V, BIAS V_{CC} = 4.5 V to 5.5 V	−1		−1		µA
		V_{CC} = 0 to 5.5 V,	OEB = 0 to 0.8 V		100		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100		100	

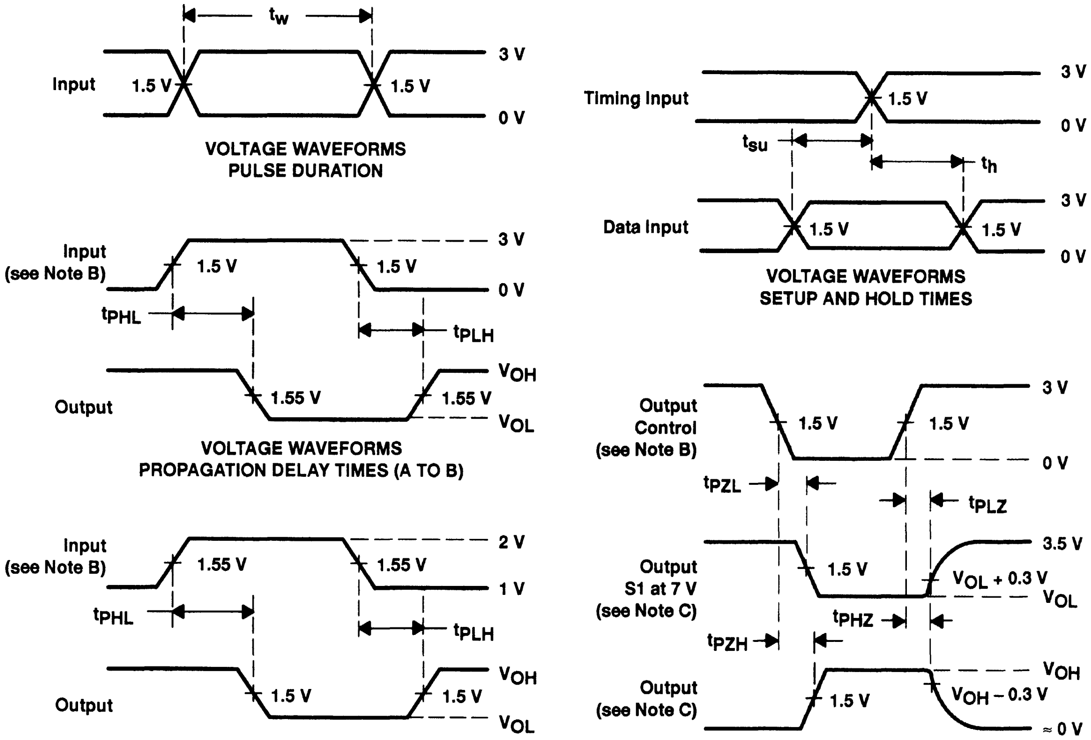
PARAMETER MEASUREMENT INFORMATION



LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	GND



VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A PORT)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

PRODUCT PREVIEW

NOTES:
A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

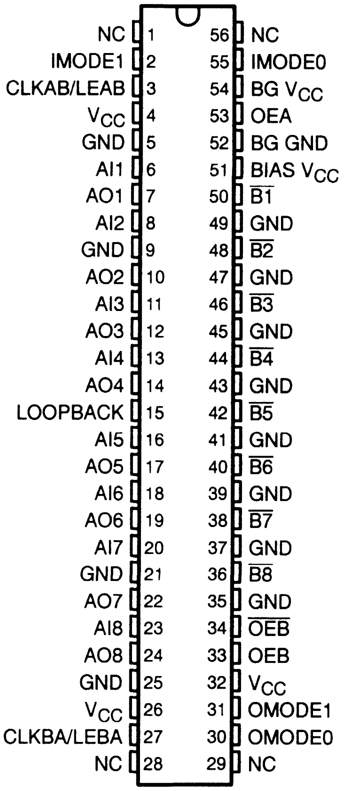
Figure 1. Load Circuit and Voltage Waveforms

SN54FB2033, SN74FB2033
8-BIT TTL/BTL REGISTERED TRANSCEIVERS

SCBS174A – NOVEMBER 1991 – REVISED JANUARY 1994

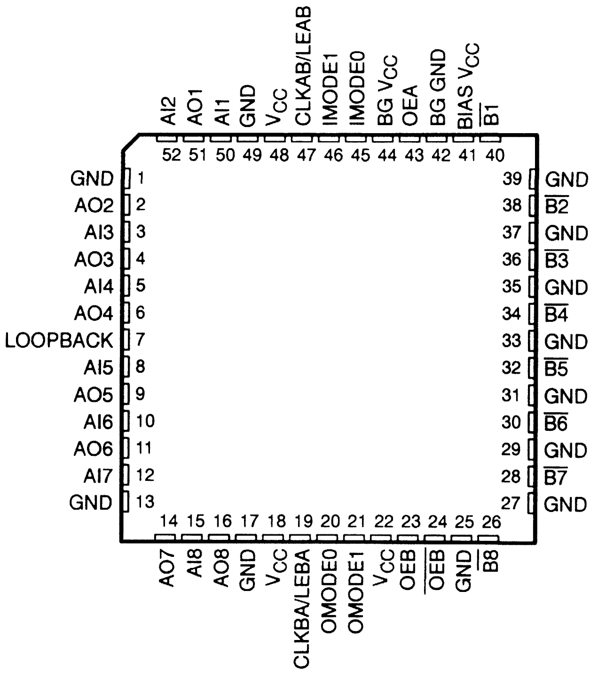
- **TTL A Port, Backplane Transceiver Logic (BTL) $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **BIAS V_{CC} Pin Minimizes Signal Distortion During Live Insertion/Withdrawal**
- **Available in Plastic Quad Flatpack (RC) and Ceramic Flatpack (WD) Packages**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**
- **TTL-Input Structures Incorporate Active Clamping Networks to Aid in Line Termination**

SN54FB2033 . . . WD PACKAGE
(TOP VIEW)



NC – No internal connection

SN74FB2033 . . . RC PACKAGE
(TOP VIEW)



TEXAS INSTRUMENTS

description

The ′FB2033 is an 8-bit transceiver featuring a split input (AI) and output (AO) bus on the TTL-level A port. The common-I/O, open-collector $\overline{\text{B}}$ port operates at backplane transceiver logic (BTL) signal levels.

The logic element for data flow in each direction is configured by two mode inputs (IMODE1 and IMODE0 for B-to-A, OMODE1 and OMODE0 for A-to-B) as a buffer, a D-type flip-flop, or a D-type latch. When configured in the buffer mode, the inverted input data appears at the output port. In the flip-flop mode, data is stored on the rising edge of the appropriate clock input (CLKAB/LEAB or CLKBA/LEBA). In the latch mode, the clock pins serve as active-high transparent latch enables.

Data flow in the B-to-A direction, regardless of the logic element selected, is further controlled by the LOOPBACK input. When LOOPBACK is low, $\overline{\text{B}}$-port data is the B-to-A input. When LOOPBACK is high, the output of the selected A-to-B logic element (prior to inversion) is the B-to-A input.

The AO port-enable/-disable control is provided by OEA. When OEA is low or when V_{CC} is less than 2.5 V, the AO port is in the high-impedance state. When OEA is high, the AO port is active (high or low logic levels).

The $\overline{\text{B}}$ port is controlled by OEB and $\overline{\text{OEB}}$. If OEB is low or $\overline{\text{OEB}}$ is high or when V_{CC} is typically less than 2.5 V, the $\overline{\text{B}}$ port is inactive. If OEB is high and $\overline{\text{OEB}}$ is low, the B port is active.

BG V_{CC} and BG GND are the bias generator reference inputs.

The A-to-B and B-to-A logic elements are active regardless of the state of their associated outputs. The logic elements can enter new data (in flip-flop and latch modes) or retain previously stored data while the associated outputs are in the high-impedance (AO port) or inactive ($\overline{\text{B}}$ port) states.

Output clamps are provided on the BTL outputs to reduce switching noise. One clamp reduces inductive ringing effects on V_{OH} during a low-to-high transition. The other clamps out ringing below the BTL V_{OL} voltage of 0.75 V. Both these clamps are active only during AC switching and do not affect the BTL outputs during steady-state conditions.

BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

The SN54FB2033 is characterized for operation over the full military temperature range of –55°C to 125°C. The SN74FB2033 is characterized for operation from 0°C to 70°C.

FUNCTION TABLE

INPUTS								FUNCTION/MODE
OEA	OEB	$\overline{\text{OEB}}$	OMODE1	OMODE0	IMODE1	IMODE0	LOOPBACK	
L	L	X	X	X	X	X	X	Isolation
L	X	H	X	X	X	X	X	
X	H	L	L	L	X	X	X	AI to $\overline{\text{B}}$, buffer mode
X	H	L	L	H	X	X	X	AI to $\overline{\text{B}}$, flip-flop mode
X	H	L	H	X	X	X	X	AI to $\overline{\text{B}}$, latch mode
H	L	X	X	X	L	L	L	$\overline{\text{B}}$ to AO, buffer mode
H	X	H	X	X	L	L	L	
H	L	X	X	X	L	H	L	$\overline{\text{B}}$ to AO, flip-flop mode
H	X	H	X	X	L	H	L	
H	L	X	X	X	H	X	L	$\overline{\text{B}}$ to AO, latch mode
H	X	H	X	X	H	X	L	
H	L	X	X	X	L	L	H	AI to AO, buffer mode
H	X	H	X	X	L	L	H	
H	L	X	X	X	L	H	H	AI to AO, flip-flop mode
H	X	H	X	X	L	H	H	
H	L	X	X	X	H	X	H	AI to AO, latch mode
H	X	H	X	X	H	X	H	
H	H	L	X	X	X	X	L	AI to $\overline{\text{B}}$, $\overline{\text{B}}$ to AO

Function Tables

ENABLE/DISABLE

INPUTS			OUTPUTS	
OEA	OEB	$\overline{\text{OEB}}$	AO	$\overline{\text{B}}$
L	X	X	Hi Z	
H	X	X	Active (H or L)	
X	L	L		Inactive (H)
X	L	H		Inactive (H)
X	H	L		Active (H or L)
X	H	H		Inactive (H)

BUFFER

INPUT	OUTPUT
L	H
H	L

LATCH

INPUTS		OUTPUT
CLK/LE	DATA	
H	L	H
H	H	L
L	X	Q_0

LOOPBACK

LOOPBACK	Q†
L	$\overline{\text{B}}$ port
H	Point P‡

† Q is the input to the B-to-A logic element.

‡ P is the output of the A-to-B logic element (see functional block diagram).

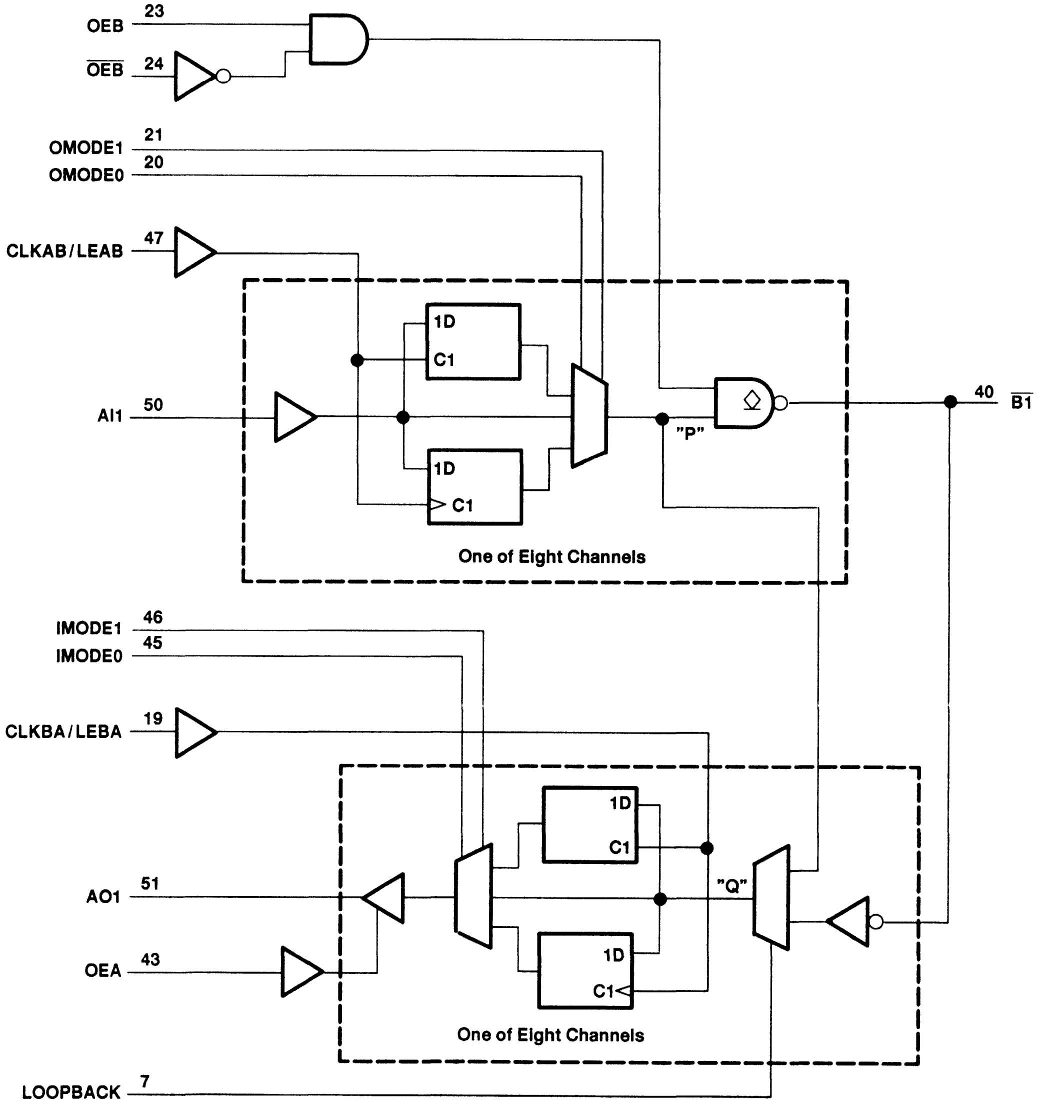
SELECT

INPUTS		SELECTED LOGIC ELEMENT
MODE1	MODE0	
L	L	Buffer
L	H	Flip-flop
H	X	Latch

FLIP-FLOP

INPUTS		OUTPUT
CLK/LE	DATA	
L	X	Q_0
↑	L	H
↑	H	L

functional block diagram



Pin numbers shown are for the RC package.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}	–0.5 V to 7 V
Input voltage range, V_I (except $\overline{B}$ port)	–1.2 V to 7 V
V_I ($\overline{B}$ port)	–1.2 V to 3.5 V
Input current range, (except $\overline{B}$ port)	–40 mA to 5 mA
Voltage range applied to any B output in the disabled or power-off state	–0.5 V to 5.5 V
Voltage range applied to any output in the high state	–0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{B}$ port	200 mA
Maximum power dissipation at T_A = 55°C (in still air): RC package	1.4 W
Storage temperature range	–65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			SN54FB2033			SN74FB2033			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}, BG V_{CC}	Supply voltage		4.75	5	5.25	4.75	5	5.25	V
BIAS V_{CC}	Supply voltage		4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	1.62		2.3	V
		Except $\overline{B}$ port	2			2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	0.75		1.47	V
		Except $\overline{B}$ port			0.8			0.8	
I_{OH}	High-level output current	AO port			-3			-3	mA
I_{OL}	Low-level output current	AO port			24			24	mA
		$\overline{B}$ port			100			100	
T_A	Operating free-air temperature		–55		125	0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS		SN54FB2033			SN74FB2033			UNIT
				MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}		V_{CC} = 4.75 V,	I_I = –18 mA			–1.2			–1.2	V
V_{OH}	AO port	V_{CC} = 4.75 V to 5.25 V,	I_{OH} = –10 µA			V_{CC}–1.1			V_{CC}–1.1	V
		V_{CC} = 4.75 V	I_{OH} = –3 mA	2.5	2.85	3.4	2.5	2.85	3.4	
			I_{OH} = –32 mA	2			2			
V_{OL}	AO port	V_{CC} = 4.75 V	I_{OL} = 20 mA		0.33	0.5		0.33	0.5	V
			I_{OL} = 55 mA			0.8			0.8	
	$\overline{B}$ port	V_{CC} = 4.75 V	I_{OL} = 100 mA	0.75		1.1	0.75		1.1	
			I_{OL} = 4 mA	0.5			0.5			
I_I	Except $\overline{B}$ port	V_{CC} = 0,	V_I = 5.25 V			100			100	µA
I_{IH}	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 2.7 V			50			50	µA
	$\overline{B}$ port‡	V_{CC} = 0 to 5.25 V,	V_I = 2.1 V			100			100	
I_{IL}	Except $\overline{B}$ port	V_{CC} = 5.25 V,	V_I = 0.5 V			–50			–50	µA
	$\overline{B}$ port‡	V_{CC} = 5.25 V,	V_I = 0.75 V			–100			–100	
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.25 V,	V_O = 2.1 V			100			100	µA
I_{OZH}	AO port	V_{CC} = 5.25 V,	V_O = 2.7 V			50			50	µA
I_{OZL}	AO port	V_{CC} = 5.25 V,	V_O = 0.5 V			–50			–50	µA
I_{OS}§	AO port	V_{CC} = 5.25 V,	V_O = 0	– 40	–80	–150	– 40	–80	–150	mA
I_{CC}	All outputs on	V_{CC} = 5.25 V,	I_O = 0		45	60		45	60	mA
C_i	AI port and control inputs	V_I = V_{CC} or GND			5			5		pF
C_o	AO port	V_O = V_{CC} or GND			5			5		pF
C_{io}¶	$\overline{B}$ port per P1194.0	V_{CC} = 0 to 4.75 V				6			6	pF
		V_{CC} = 4.75 V to 5.25 V				6			6	

† Parameter is based on characterization data but not tested.

‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.

§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

¶ Parameter is based on characterization data but is not tested

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 2)

		V_{CC} = 5 V, T_A = 25°C		SN54FB2033		SN74FB2033		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
f_{clock}	Clock frequency	0	150	0	150	0	150	MHz
t_w	Pulse duration, CLKAB/LEAB or CLKBA/LEBA	4		4		4		ns
t_{su}	Setup time, data before CLKAB/LEAB or CLKBA/LEBA↑	4		4		4		ns
t_h	Hold time, data after CLKAB/LEAB or CLKBA/LEBA↑	1		1		1		ns

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 2)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, T_A = 25°C			SN54FB2033		SN74FB2033		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
f_{max}			150			150		150		MHz
t_{PLH}	AI (thru mode)	$\overline{B}$	2.3	3.8	5.3	1.8	6.5	1.8	6.5	ns
t_{PHL}			1.2	2.6	4	1.2	4.2	1.2	4.2	
t_{PLH}	$\overline{B}$ (thru mode)	AO	2.2	3.9	5.7	1.9	6.6	1.9	6.6	ns
t_{PHL}			3.8	5.2	6.7	3.2	7.3	3.2	7.3	
t_{PLH}	AI (transparent)	$\overline{B}$	3.5	5	6.7	2.9	8.1	2.9	8.1	ns
t_{PHL}			2.1	3.6	5.3	2	5.7	2	5.7	
t_{PLH}	$\overline{B}$ (transparent)	AO	2.6	4.3	6.3	2.3	7.2	2.3	7.2	ns
t_{PHL}			4.3	5.6	7.1	3.7	7.6	3.7	7.6	
t_{PLH}	$\overline{OEB}$	$\overline{B}$	2.4	3.7	5.3	2	6.4	2	6.4	ns
t_{PHL}			1.2	2.6	4.1	1.2	4.4	1.2	4.4	
t_{PLH}	$\overline{OEB}$	$\overline{B}$	2.5	3.8	5.3	2.2	6.4	2.2	6.4	ns
t_{PHL}			1.4	2.9	4.5	1.3	4.9	1.3	4.9	
t_{PZH}	OEA	AO	1.8	3.5	5.1	1.5	5.6	1.5	5.6	ns
t_{PZL}			2.6	4.3	5.9	1.8	6.2	1.8	6.2	
t_{PHZ}	OEA	AO	1.7	3.5	5.3	1.4	5.7	1.4	5.7	ns
t_{PLZ}			1	2.7	4.5	1	4.9	1	4.9	
t_{PLH}	CLKAB/LEAB	$\overline{B}$	3.5	5	6.7	3	8.1	3	8.1	ns
t_{PHL}			2	3.6	5.2	1.9	5.5	1.9	5.5	
t_{PLH}	CLKBA/LEBA	AO	2.2	3.8	5.4	1.9	5.8	1.9	5.8	ns
t_{PHL}			2.7	4.1	5.6	2.4	5.7	2.4	5.7	
t_{PLH}	OMODE	$\overline{B}$	3.2	4.8	6.5	2.7	7.9	2.7	7.9	ns
t_{PHL}			1.9	3.5	5.2	1.7	5.7	1.7	5.7	
t_{PLH}	IMODE	AO	2	3.6	5.3	1.7	6	1.7	6	ns
t_{PHL}			2.5	4.1	5.6	1.8	5.8	1.8	5.8	
t_{PLH}	LOOPBACK	AO	2.3	4.6	6.8	2	7.5	2	7.5	ns
t_{PHL}			3.2	4.8	6.4	2.9	6.4	2.9	6.4	
t_{PLH}	AI	AO	2.1	3.7	5.4	1.9	5.8	1.9	5.8	ns
t_{PHL}			2.9	4.3	5.9	2.5	6.4	2.5	6.4	
t_t	Rise time, 1.3 V to 1.8 V	$\overline{B}$		1.5						ns
	Fall time, 1.8 V to 1.3 V			1.5						
	Rise or fall time, 10% to 90%	AO		3.5						
	$\overline{B}$-port input pulse rejection					1		1		ns

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

live-insertion characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		SN54FB2033		SN74FB2033		UNIT
				MIN	MAX	MIN	MAX	
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.5 V	V_B = 0 to 2 V, BIAS V_{CC} = 4.5 V to 5.5 V		400		400	µA
		V_{CC} = 4.5 V to 5.5 V			10		10	
V_O	$\overline{B}$ port	V_{CC} = 0,	BIAS V_{CC} = 4.5 V to 5.5 V	1.62	2.1	1.62	2.1	V
I_O	B port	V_{CC} = 0, V_I (BIAS V_{CC}) = 4.5 V to 5.5 V	V_B = 1 V,	–1		–1		µA
		V_{CC} = 0 to 5.5 V,	OEB = 0 to 0.8 V		100		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100		100	

miscellaneous characteristics

			TEST CONDITIONS	SN54FB2033		SN74FB2033		UNIT
				MIN	MAX	MIN	MAX	
V_{OHP}†	Peak output voltage during turnoff of 100 mA into 40 nH	$\overline{B}$ port	See Figure 1		4		4	V
V_{OHV}†	Minimum output voltage during turnoff of 100 mA into 40 nH			1.62		1.62		V
V_{OLV}	Minimum output voltage during high-to-low switch		I_{OL} = –50 mA	0.3		0.3		V

† Parameter is based on characterization data but not tested.

PARAMETER MEASUREMENT INFORMATION

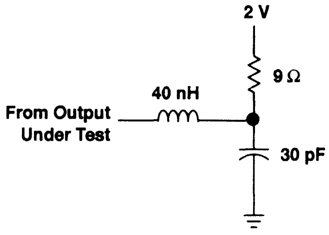
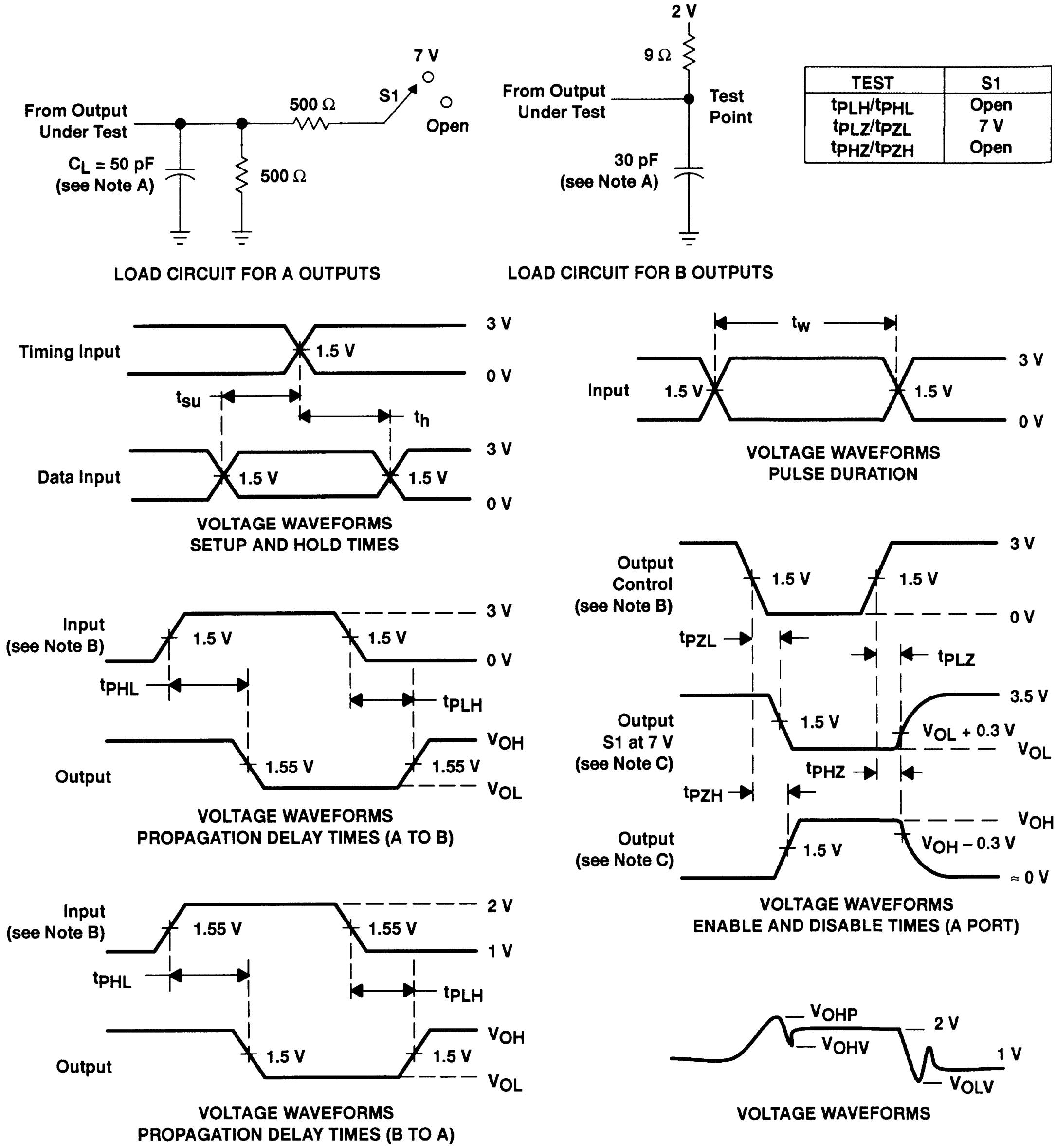


Figure 1. Load Circuit for V_{OHP}, V_{OHV}

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.



TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	Open

LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

VOLTAGE WAVEFORMS
PULSE DURATION

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A PORT)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS

NOTES: A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns, BTL inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

Figure 2. Load Circuit and Voltage Waveforms

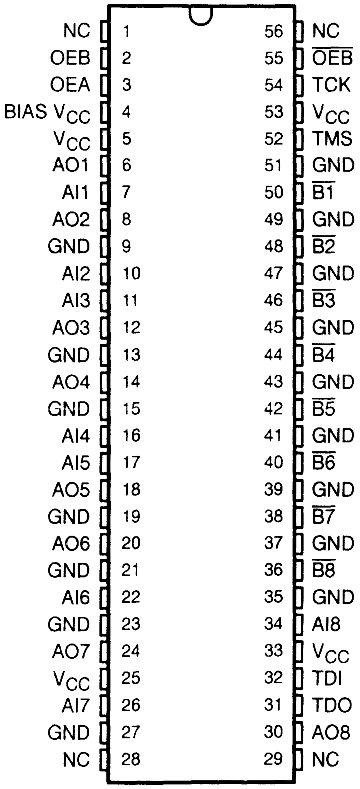
SN54FB2040, SN74FB2040
8-BIT TTL/BTL TRANSCEIVERS

SCBS173A – NOVEMBER 1991 – REVISED JANUARY 1994

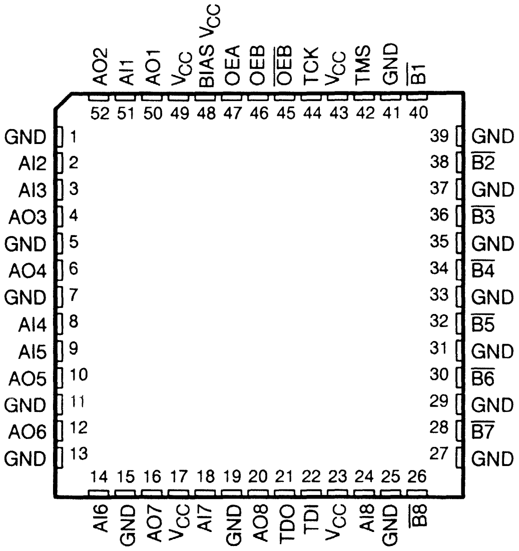
- **Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards**
- **TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port**
- **Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA**
- **Minimum $\overline{\text{B}}$-Port Edge Rate = 2 ns**
- **Isolated Logic-Ground and Bus-Ground Pins Reduce Noise**
- **BIAS V_{CC} Pin Minimizes Signal Distortion During Live Insertion/Withdrawal**
- **Available in Plastic Quad Flatpack (RC) and Ceramic Flatpack (WD) Packages**
- **$\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage**

PRODUCT PREVIEW

SN54FB2040 . . . WD PACKAGE
(TOP VIEW)



SN74FB2040 . . . RC PACKAGE
(TOP VIEW)



description

The ′FB2040 is an 8-bit transceiver designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{\text{B}}$ port operates at BTL-signal levels. The open-collector $\overline{\text{B}}$ ports are specified to sink 100 mA and have minimum output edge rates of 2 ns. Two output enables, OEB and $\overline{\text{OEB}}$, are provided for the $\overline{\text{B}}$ outputs. When OEB is high and $\overline{\text{OEB}}$ is low, the $\overline{\text{B}}$ port is active and reflects the inverse of the data present at the A-input pins. When OEB is low, $\overline{\text{OEB}}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{\text{B}}$ port is turned off.

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description (continued)

The A port operates at TTL-signal levels and has split input and output pins. The A outputs reflect the inverse of the data at the $\overline{\text{B}}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Pins are allocated for the four-wire IEEE 1149.1 (JTAG) test bus, which will be implemented in a future version of the ′FB2040. Currently TMS and TCK are not connected and TDI is shorted to TDO.

BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

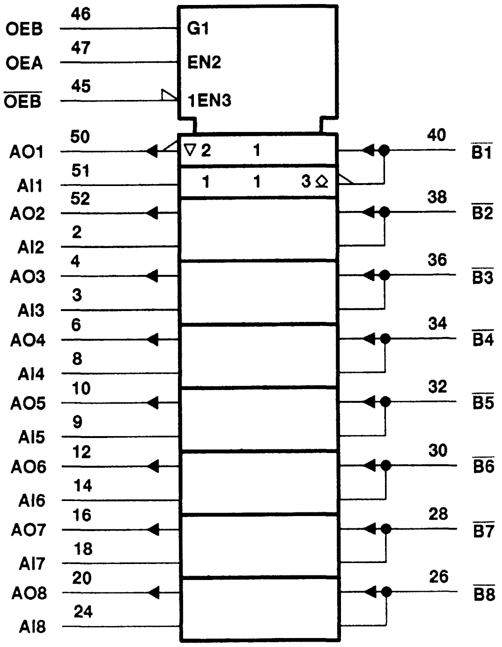
The SN54FB2040 is characterized for operation over the full military temperature range of −55°C to 125°C. The SN74FB2040 is characterized for operation from 0°C to 70°C.

PRODUCT PREVIEW

FUNCTION TABLE

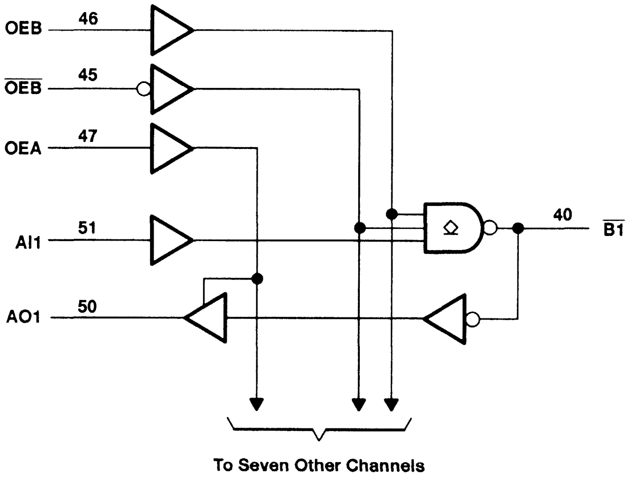
INPUTS			FUNCTION
OEB	$\overline{\text{OEB}}$	OEA	
L X	X H	L L	Isolation
L X	X H	H H	$\overline{\text{B}}$ data to AO bus
H	L	L	$\overline{\text{AI}}$ data to B bus
H	L	H	$\overline{\text{AI}}$ data to B bus, $\overline{\text{B}}$ data to AO bus

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for RC package.

functional block diagram



Pin numbers shown are for RC package.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias V_{CC}	−0.5 V to 7 V
Input voltage range, V_I: (except $\overline{B}$ port)	−1.2 V to 7 V
($\overline{B}$ port)	−1.2 V to 3.5 V
Input current range (except $\overline{B}$ port)	−40 mA to 5 mA
Voltage range applied to any $\overline{B}$ output in the disabled or power-off state	−0.5 V to 5.5 V
Voltage range applied to any output in the high state	−0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{B}$ port	200 mA
Operating free-air temperature range, T_A: SN54FB2041	−55°C to 125°C
SN74FB2041	0°C to 70°C
Maximum power dissipation at T_A = 55°C (in still air): RC package	1.4 W
Storage temperature range	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

PRODUCT PREVIEW

recommended operating conditions (see Note 1)

			SN54FB2040			SN74FB2040			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}, BIAS V_{CC}, BG V_{CC}	Supply voltage		4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	1.62		2.3	V
		Except $\overline{B}$ port	2			2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	0.75		1.47	V
		Except $\overline{B}$ port			0.8			0.8	
I_{IK}	Input clamp current				–18			–18	mA
I_{OH}	High-level output current	AO port						–3	mA
I_{OL}	Low-level output current	AO port						24	mA
		$\overline{B}$ port			100			100	
T_A	Operating free-air temperature		–55		125	0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS		SN54FB2040			SN74FB2040			UNIT
				MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}	$\overline{B}$ port	V_{CC} = 4.5 V	I_I = –18 mA			–1.2			–1.2	V
	Except $\overline{B}$ port		I_I = –40 mA			–0.5			–0.5	
V_{OH}	AO port	V_{CC} = 4.5 V	I_{OH} = –1 mA							V
			I_{OH} = –3 mA	2.5	3.3		2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.5 V	I_{OL} = 20 mA							V
			I_{OL} = 24 mA		0.35	0.5		0.35	0.5	
	B port	V_{CC} = 4.5 V	I_{OL} = 80 mA	0.75		1.1	0.75		1.1	
			I_{OL} = 100 mA			1.15			1.15	
I_I	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 5.5 V			50			50	µA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 2.7 V			50			50	µA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V	V_I = 0.5 V			–50			–50	µA
	$\overline{B}$ port		V_I = 0.75 V			–100			–100	
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.5 V,	V_O = 2.1 V			100			100	µA
I_{OZH}	AO port	V_{CC} = 5.5 V,	V_O = 2.7 V			50			50	µA
I_{OZL}	AO port	V_{CC} = 5.5 V,	V_O = 0.5 V			–50			–50	µA
I_{OS}§	AO port	V_{CC} = 5.5 V,	V_O = 0	– 30		–150	– 30		–150	mA
I_{CC}	AI port to $\overline{B}$ port	V_{CC} = 5.5 V,	I_O = 0		25			25		mA
	$\overline{B}$ port to AO port				60			60		
	Outputs disabled									
C_i	AI port and control inputs	V_I = V_{CC} or GND								pF
C_o	AO port	V_O = V_{CC} or GND								pF
C_{io}	B port per P1194.0	V_{CC} = 0 to 4.5 V				6			6	pF
		V_{CC} = 4.5 V to 5.5 V				5			5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

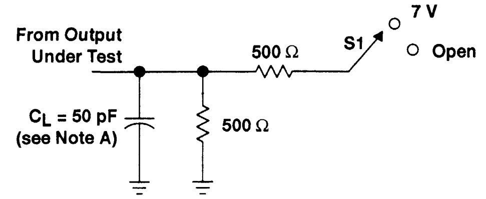
PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, T_A = 25°C			SN54FB2040		SN74FB2040		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{PLH}	AI	$\overline{B}$		3.9						ns
t_{PHL}				3.6						
t_{PLH}	$\overline{B}$	AO		3.9						ns
t_{PHL}				3.8						
t_{PLH}	OEB	$\overline{B}$		5.1						ns
t_{PHL}				4.3						
t_{PLH}	OEB	$\overline{B}$		4.4						ns
t_{PHL}				4.1						
t_{PZH}	OEA	AO		3.2						ns
t_{PZL}				3						
t_{PHZ}	OEA	AO		3.2						ns
t_{PLZ}				2.7						
$t_{sk(p)}$	Skew for any single channel $\lvert t_{PHL} - t_{PLH} \rvert$	AI to $\overline{B}$ or $\overline{B}$ to AO					0.75		0.75	ns
$t_{sk(o)}$	Skew between drivers in the same package	AI to $\overline{B}$ or $\overline{B}$ to AO		1	1.5		2		2	ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)			2		1	3	1	3	ns
	$\overline{B}$-port input pulse rejection					1		1		ns

live-insertion characteristics over recommended operating free-air temperature range

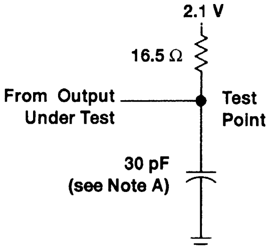
PARAMETER		TEST CONDITIONS		SN54FB2040		SN74FB2040		UNIT
				MIN	MAX	MIN	MAX	
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.5 V	V_B = 0 to 2 V, BIAS V_{CC} = 4.5 V to 5.5 V		450		450	µA
		V_{CC} = 4.5 to 5.5 V			10		10	
V_O	$\overline{B}$ port	V_{CC} = 0,	BIAS V_{CC} = 4.5 V to 5.5 V	1.62	2.1	1.62	2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0,	V_B = 1 V, BIAS V_{CC} = 4.5 V to 5.5 V	–1		–1		µA
		V_{CC} = 0 to 5.5 V,	OEB = 0 to 0.8 V		100		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100		100	

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION

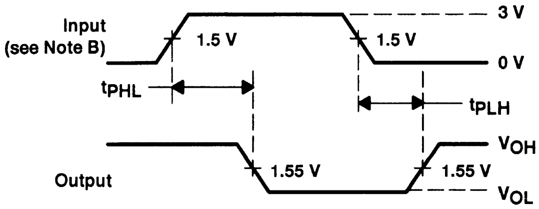


LOAD CIRCUIT FOR A OUTPUTS

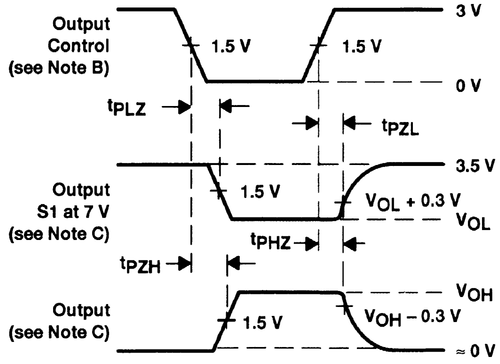


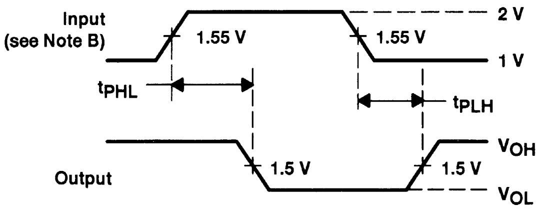
LOAD CIRCUIT FOR B OUTPUTS

TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	Open



VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)





VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

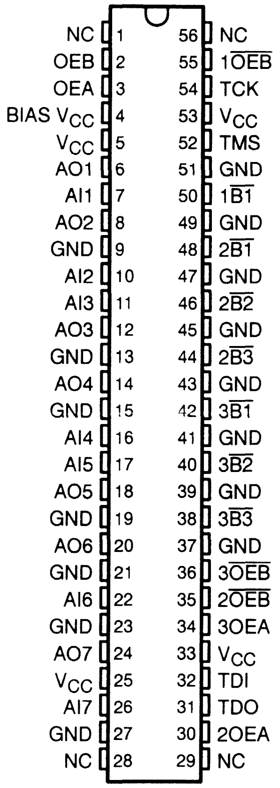
VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A PORT)

NOTES:
A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

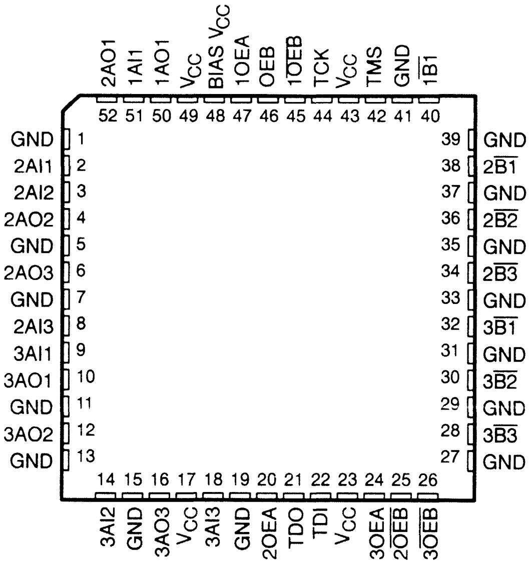
Figure 1. Load Circuit and Voltage Waveforms

- Compatible With IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) Standards
- TTL A Port, Backplane Transceiver Logic $\overline{\text{B}}$ Port
- Open-Collector $\overline{\text{B}}$-Port Outputs Sink 100 mA
- Minimum $\overline{\text{B}}$-Port Edge Rate = 2 ns
- Isolated Logic-Ground and Bus-Ground Pins Reduce Noise
- BIAS V_{CC} Pin Minimizes Signal Distortion During Live Insertion/Withdrawal
- Available in Plastic Quad Flatpack (RC) and Ceramic Flatpack (WD) Packages
- $\overline{\text{B}}$-Port Biasing Network Preconditions the Connector and PC Trace to the Backplane Transceiver Logic High-Level Voltage

SN54FB2041 . . . WD PACKAGE
(TOP VIEW)


NC 1 56 NC
OEB 2 55 1$\overline{\text{OEB}}$
OEA 3 54 TCK
BIAS V_{CC} 4 53 V_{CC}
V_{CC} 5 52 TMS
AO1 6 51 GND
AI1 7 50 1$\overline{\text{B1}}$
AO2 8 49 GND
GND 9 48 2$\overline{\text{B1}}$
AI2 10 47 GND
AI3 11 46 2$\overline{\text{B2}}$
AO3 12 45 GND
GND 13 44 2$\overline{\text{B3}}$
AO4 14 43 GND
GND 15 42 3$\overline{\text{B1}}$
AI4 16 41 GND
AI5 17 40 3$\overline{\text{B2}}$
AO5 18 39 GND
GND 19 38 3$\overline{\text{B3}}$
AO6 20 37 GND
GND 21 36 3$\overline{\text{OEB}}$
AI6 22 35 2$\overline{\text{OEB}}$
GND 23 34 3OEA
AO7 24 33 V_{CC}
V_{CC} 25 32 TDI
AI7 26 31 TDO
GND 27 30 2OEA
NC 28 29 NC

SN74FB2041 . . . RC PACKAGE
(TOP VIEW)



NC – no internal connection

PRODUCT PREVIEW

PRODUCT PREVIEW information concerns products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

description

The 'FB2041 is a 7-bit transceiver designed to translate signals between TTL and backplane transceiver logic (BTL) environments. It is specifically designed to be compatible with IEEE 1194.1-1991 (BTL) and IEEE 896.2-1991 (Futurebus+) standards.

The $\overline{B}$ port operates at BTL-signal levels. The open-collector $\overline{B}$ ports are specified to sink 100 mA and have minimum output edge rates of 2 ns. Two output enables, OEB and $\overline{OEB}$, are provided for the $\overline{B}$ outputs. When OEB is high and $\overline{OEB}$ is low, the $\overline{B}$ port is active and reflects the inverse of the data present at the A-input pins. When OEB is low, $\overline{OEB}$ is high, or V_{CC} is typically less than 2.5 V, the $\overline{B}$ port is turned off. The enable/disable logic partitions the device as two 3-bit sections and one 1-bit section.

The A port operates at TTL-signal levels and has split input and output pins. The A outputs reflect the inverse of the data at the $\overline{B}$ port when the A-port output enable, OEA, is high. When OEA is low or when V_{CC} is typically less than 2.5 V, the A outputs are in the high-impedance state.

Pins are allocated for the four-wire IEEE 1149.1 (JTAG) test bus, which will be implemented in a future version of the SN74FB2041. Currently TMS and TCK are not connected and TDI is shorted to TDO.

BIAS V_{CC} establishes a voltage between 1.62 V and 2.1 V on the BTL outputs when V_{CC} is not connected.

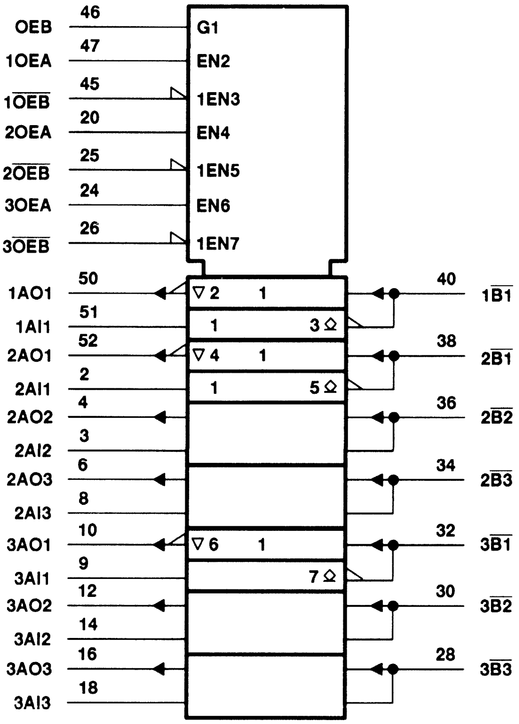
The SN54FB2041 is characterized for operation over the full military temperature range of –55°C to 125°C. The SN74FB2041 is characterized for operation from 0°C to 70°C.

PRODUCT PREVIEW

FUNCTION TABLE

INPUTS			FUNCTION
OEB	$\overline{OEB}$	OEA	
L X	X H	L L	Isolation
L X	X H	H H	$\overline{B}$ data to AO bus
H	L	L	$\overline{AI}$ data to B bus
H	L	H	$\overline{AI}$ data to B bus, $\overline{B}$ data to AO bus

logic symbol†

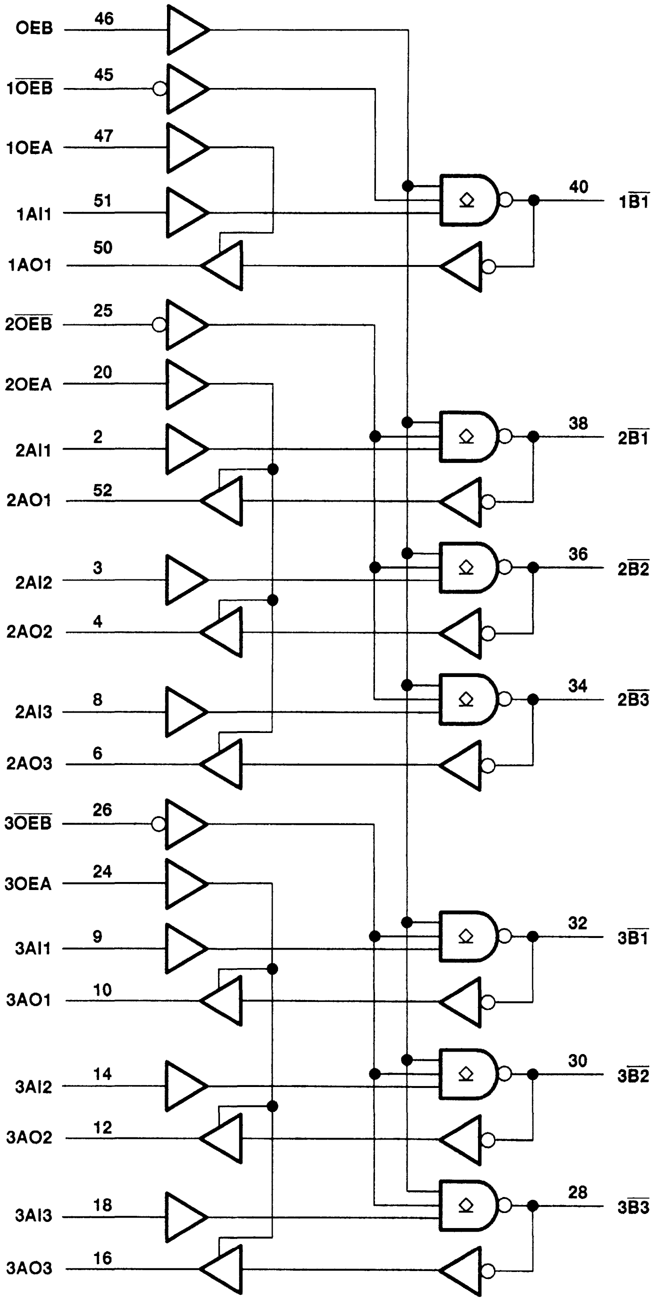


† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for RC package.

PRODUCT PREVIEW

PRODUCT PREVIEW

functional block diagram



Pin numbers shown are for RC package.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC}, Bias V_{CC}	−0.5 V to 7 V
Input voltage range, V_I: (except $\overline{B}$ port)	−1.2 V to 7 V
($\overline{B}$ port)	−1.2 V to 3.5 V
Input current range (except $\overline{B}$ port)	−40 mA to 5 mA
Voltage range applied to any $\overline{B}$ output in the disabled or power-off state	−0.5 V to 5.5 V
Voltage range applied to any output in the high state	−0.5 V to V_{CC}
Current applied to any single output in the low state: A port	48 mA
$\overline{B}$ port	200 mA
Operating free-air temperature range, T_A: SN54FB2041	−55°C to 125°C
SN74FB2041	0°C to 70°C
Maximum power dissipation at T_A = 55°C (in still air): RC package	1.4 W
Storage temperature range	−65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions (see Note 1)

			SN54FB2041			SN74FB2041			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}, BIAS V_{CC}, BG V_{CC}	Supply voltage		4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	$\overline{B}$ port	1.62		2.3	1.62		2.3	V
		Except $\overline{B}$ port	2			2			
V_{IL}	Low-level input voltage	$\overline{B}$ port	0.75		1.47	0.75		1.47	V
		Except $\overline{B}$ port			0.8			0.8	
I_{IK}	Input clamp current				−18			−18	mA
I_{OH}	High-level output current	AO port			-3			-3	mA
I_{OL}	Low-level output current	AO port			24			24	mA
		$\overline{B}$ port			100			100	
T_A	Operating free-air temperature		−55		125	0		70	°C

NOTE 1: Unused or floating pins (input or I/O) must be held high or low.

PRODUCT PREVIEW

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS		SN54FB2041			SN74FB2041			UNIT
				MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}	$\overline{B}$ port	V_{CC} = 4.5 V	I_I = –18 mA			–1.2			–1.2	V
	Except $\overline{B}$ port		I_I = –40 mA			–0.5			–0.5	
V_{OH}	AO port	V_{CC} = 4.5 V	I_{OH} = – 1 mA							V
			I_{OH} = –3 mA	2.5	3.3		2.5	3.3		
V_{OL}	AO port	V_{CC} = 4.5 V	I_{OL} = 20 mA							V
			I_{OL} = 24 mA		0.35	0.5		0.35	0.5	
	B port	V_{CC} = 4.5 V	I_{OL} = 80 mA	0.75		1.1	0.75		1.1	
			I_{OL} = 100 mA			1.15			1.15	
I_I	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 5.5 V			50			50	µA
I_{IH}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V,	V_I = 2.7 V			50			50	µA
I_{IL}‡	Except $\overline{B}$ port	V_{CC} = 5.5 V	V_I = 0.5 V			–50			–50	µA
	$\overline{B}$ port		V_I = 0.75 V			–100			–100	
I_{OH}	$\overline{B}$ port	V_{CC} = 0 to 5.5 V,	V_O = 2.1 V			100			100	µA
I_{OZH}	AO port	V_{CC} = 5.5 V,	V_O = 2.7 V			50			50	µA
I_{OZL}	AO port	V_{CC} = 5.5 V,	V_O = 0.5 V			–50			–50	µA
I_{OS}§	AO port	V_{CC} = 5.5 V,	V_O = 0	– 30		–150	– 30		–150	mA
I_{CC}	AI port to $\overline{B}$ port	V_{CC} = 5.5 V,	I_O = 0		25			25		mA
	$\overline{B}$ port to AO port				65			65		
	Outputs disabled									
C_i	AI port and control inputs	V_I = V_{CC} or GND								pF
C_o	AO port	V_O = V_{CC} or GND								pF
C_{io}	B port per P1194.0	V_{CC} = 0 to 4.5 V				6			6	pF
		V_{CC} = 4.5 V to 5.5 V				5			5	

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ For I/O ports, the parameters I_{IH} and I_{IL} include the off-state output current.
§ Not more than one output should be shorted at a time, and the duration of the short circuit should not exceed one second.

PRODUCT PREVIEW

switching characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 1)

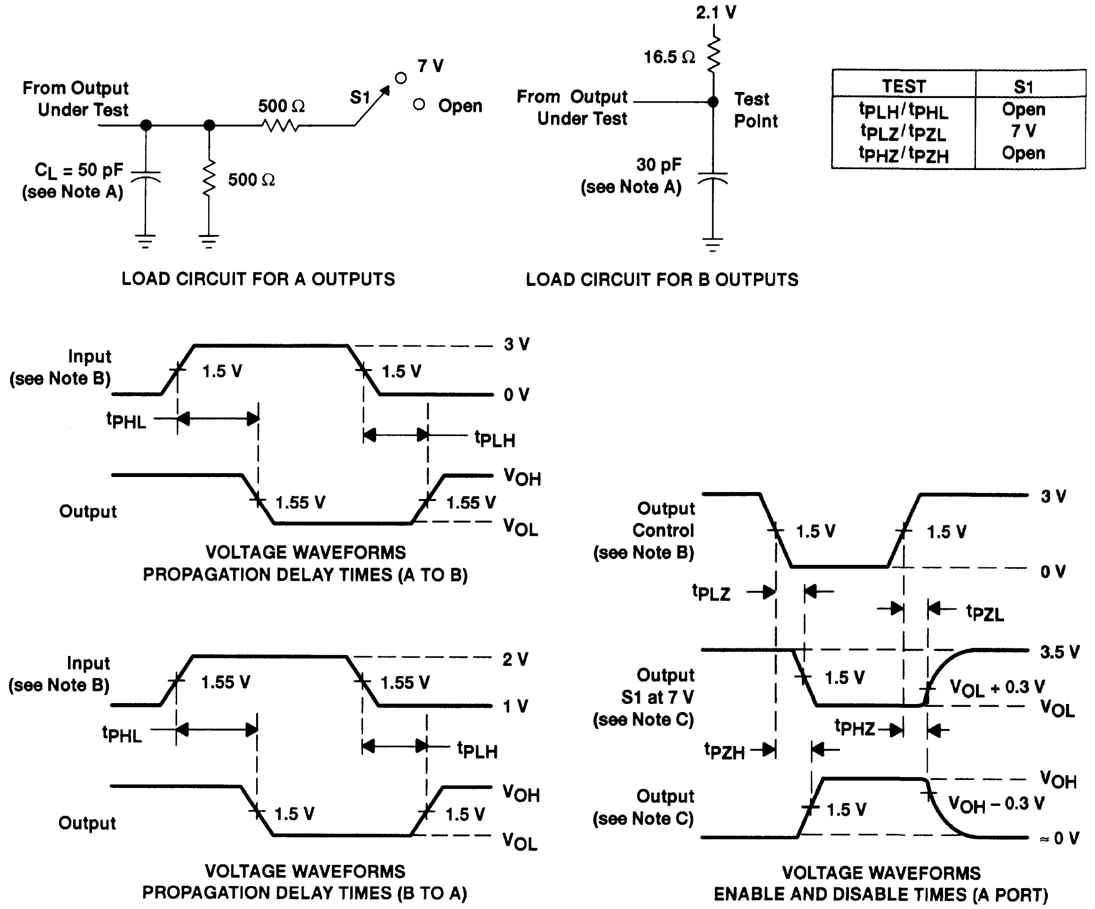
PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, T_A = 25°C			SN54FB2041		SN74FB2041		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{PLH}	AI	B		3.9						ns
t_{PHL}				3.6						
t_{PLH}	B	AO		3.8						ns
t_{PHL}				3.8						
t_{PLH}	OEB	B		4.8						ns
t_{PHL}				4.3						
t_{PLH}	OEB	B		4.2						ns
t_{PHL}				3.8						
t_{PZH}	OEA	AO		3						ns
t_{PZL}				3						
t_{PHZ}	OEA	AO		3.3						ns
t_{PLZ}				2.6						
$t_{sk(p)}$	Skew for any single channel $\lvert t_{PHL} - t_{PLH} \rvert$	AI to $\overline{B}$ or $\overline{B}$ to AO					0.75		0.75	ns
$t_{sk(o)}$	Skew between drivers in the same package	AI to $\overline{B}$ or $\overline{B}$ to AO		1	1.5		2		2	ns
t_t	Transition time, $\overline{B}$ outputs (1.3 V to 1.8 V)			2		1	3	1	3	ns
	$\overline{B}$-port input pulse rejection					1		1		ns

live-insertion characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS		SN54FB2041		SN74FB2041		UNIT
				MIN	MAX	MIN	MAX	
I_{CC} (BIAS V_{CC})		V_{CC} = 0 to 4.5 V	V_B = 0 to 2 V, BIAS V_{CC} = 4.5 V to 5.5 V		450		450	µA
		V_{CC} = 4.5 V to 5.5 V			10		10	
V_O	$\overline{B}$ port	V_{CC} = 0,	BIAS V_{CC} = 4.5 V to 5.5 V	1.62	2.1	1.62	2.1	V
I_O	$\overline{B}$ port	V_{CC} = 0,	BIAS V_{CC}= 4.5 V to 5.5 V	–1		–1		µA
		V_{CC} = 0 to 5.5 V,	OEB = 0 to 0.8 V		100		100	
		V_{CC} = 0 to 2.2 V,	OEB = 0 to 5 V		100		100	

PRODUCT PREVIEW

PARAMETER MEASUREMENT INFORMATION



LOAD CIRCUIT FOR A OUTPUTS

LOAD CIRCUIT FOR B OUTPUTS

TEST	S1
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	7 V
t_{PHZ}/t_{PZH}	Open

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (A TO B)

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES (B TO A)

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES (A PORT)

NOTES:
A. C_L includes probe and jig capacitance.
B. All input pulses are supplied by generators having the following characteristics: TTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns. BTL Inputs - PRR ≤ 10 MHz, Z_O = 50 Ω, t_r ≤ 2.5 ns, t_f ≤ 2.5 ns.
C. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
D. The outputs are measured one at a time with one transition per measurement.

Figure 1. Load Circuit and Voltage Waveforms

Appendix A

Transceiver Interconnect Application Note

Appendix A
Transceiver Interconnect Application Note

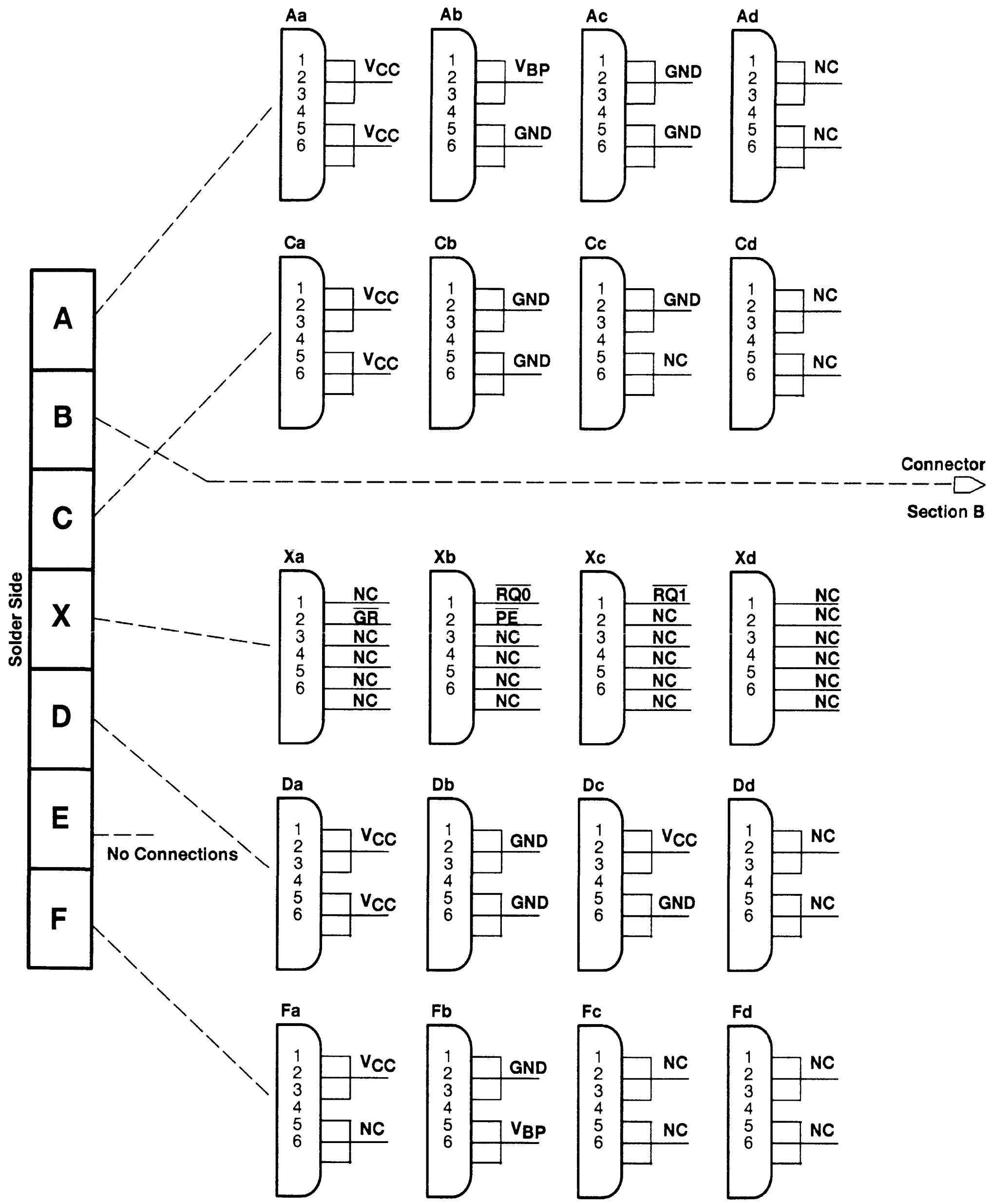
Notes

NOTES: 1. PU is a standard TTL pullup to V_{CC} through a 1 kΩ resistor. See Figures A–2, A–4, and A–6

2. PD is a tie point to GND.

3. Each individual line (HA<31:0>, HAP<3:0>, HD<31:0>, and HDP<3:0>) must have a series 25 Ω resistor at the DPU. See Figure A–4.

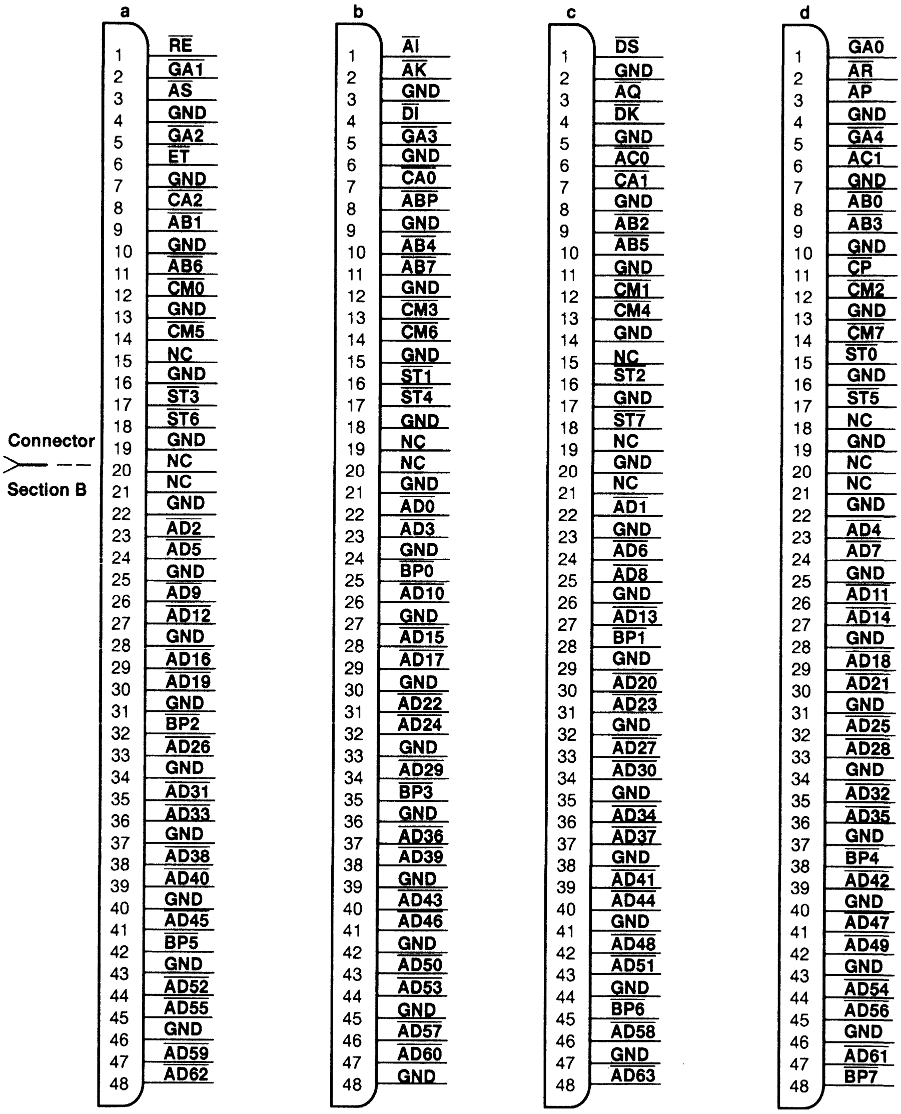
4. Bias V_{CC} connections to V_{BP} are necessary only if live insertion is supported. Otherwise, Bias V_{CC} can be connected to V_{CC}. (See transceivers in Figures A–6, A–7, and A–8)



NC – No internal connection

NOTE: Pinout shown is for the backplane connector, looking at backplane from front panel opening.

Figure A–1. Futurebus+ 12SU Backplane Connector



	a	b	c	d
1	RE	AI	DS	GA0
2	GA1	AK	GND	AR
3	AS	GND	AQ	AP
4	GND	DI	DK	GND
5	GA2	GA3	GND	GA4
6	ET	GND	AC0	AC1
7	GND	CA0	CA1	GND
8	CA2	ABP	GND	AB0
9	AB1	GND	AB2	AB3
10	GND	AB4	AB5	GND
11	AB6	AB7	GND	CP
12	CM0	GND	CM1	CM2
13	GND	CM3	CM4	GND
14	CM5	CM6	GND	CM7
15	NC	GND	NC	ST0
16	GND	ST1	ST2	GND
17	ST3	ST4	GND	ST5
18	ST6	GND	ST7	NC
19	GND	NC	NC	GND
20	NC	NC	GND	NC
21	NC	GND	NC	NC
22	GND	AD0	AD1	GND
23	AD2	AD3	GND	AD4
24	AD5	GND	AD6	AD7
25	GND	BP0	AD8	GND
26	AD9	AD10	GND	AD11
27	AD12	GND	AD13	AD14
28	GND	AD15	BP1	GND
29	AD16	AD17	GND	AD18
30	AD19	GND	AD20	AD21
31	GND	AD22	AD23	GND
32	BP2	AD24	GND	AD25
33	AD26	GND	AD27	AD28
34	GND	AD29	AD30	GND
35	AD31	BP3	GND	AD32
36	AD33	GND	AD34	AD35
37	GND	AD36	AD37	GND
38	AD38	AD39	GND	BP4
39	AD40	GND	AD41	AD42
40	GND	AD43	AD44	GND
41	AD45	AD46	GND	AD47
42	BP5	GND	AD48	AD49
43	GND	AD50	AD51	GND
44	AD52	AD53	GND	AD54
45	AD55	GND	BP6	AD56
46	GND	AD57	AD58	GND
47	AD59	AD60	GND	AD61
48	AD62	GND	AD63	BP7

NC – No internal connection

NOTE: Pinout shown is for the backplane connector, looking at backplane from front panel opening.

Figure A–2. Futurebus+, Section B of 12SU Backplane Connector

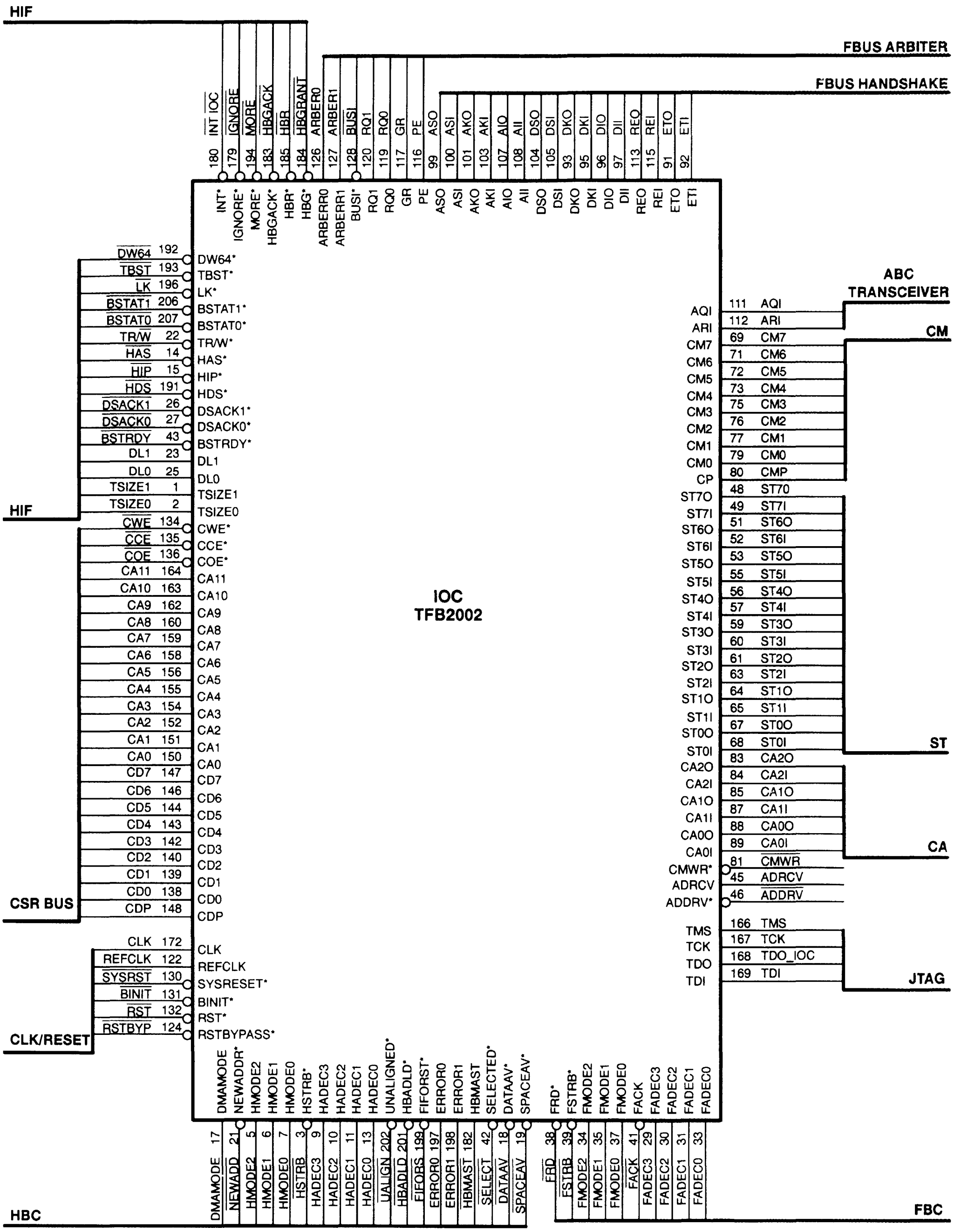


Figure A–3. Interconnect

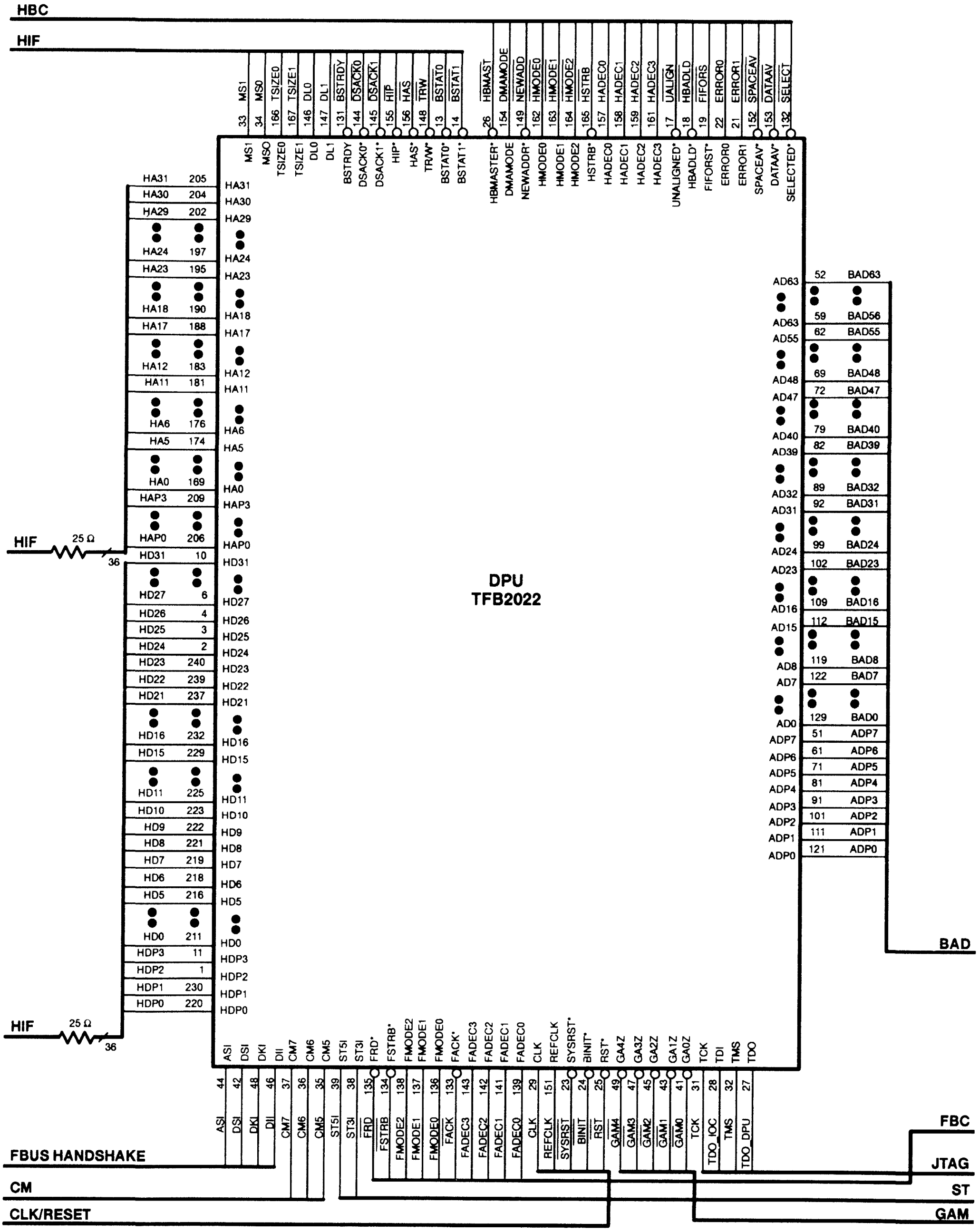


Figure A–4. Interconnect

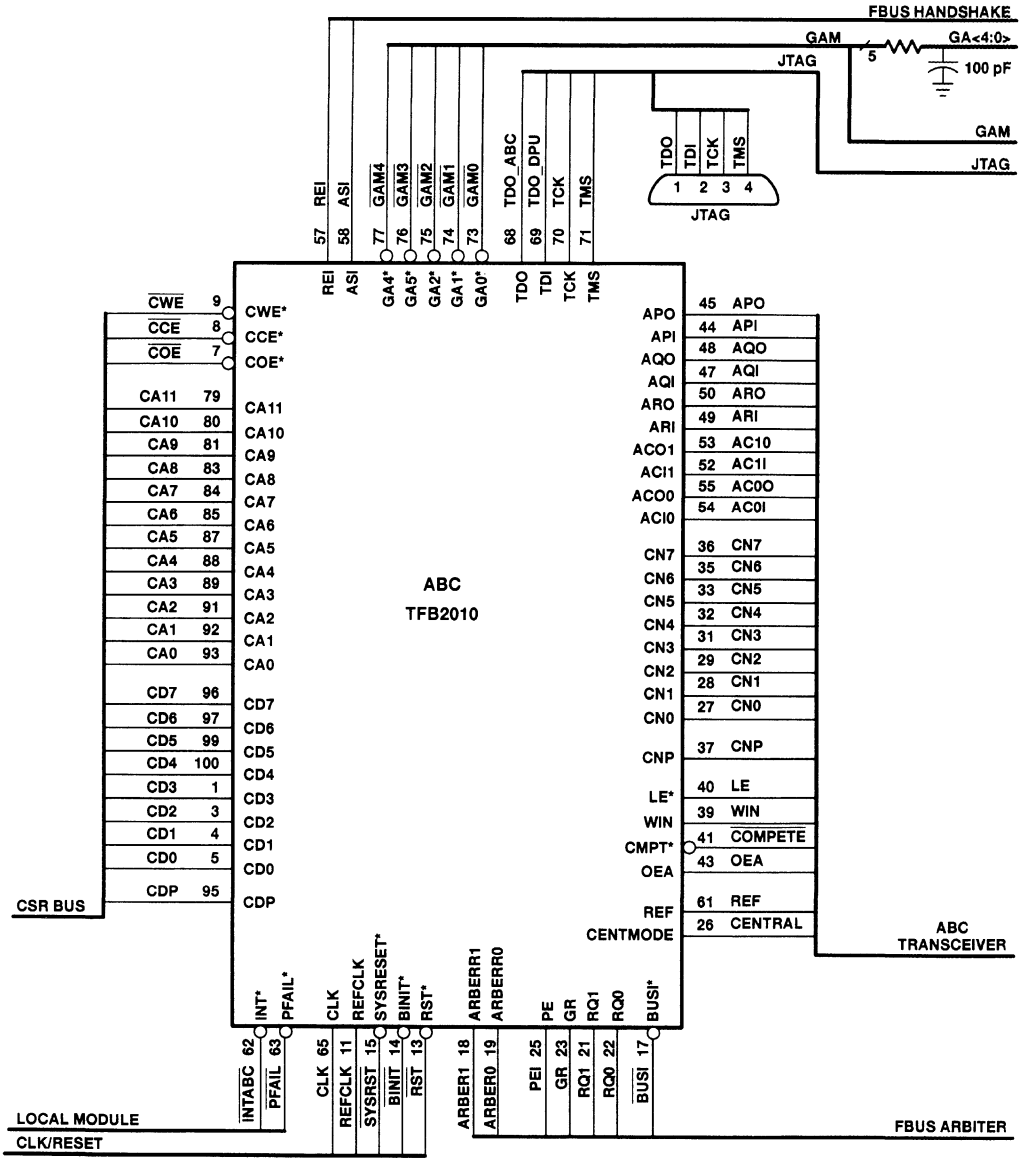


Figure A–5. Interconnect

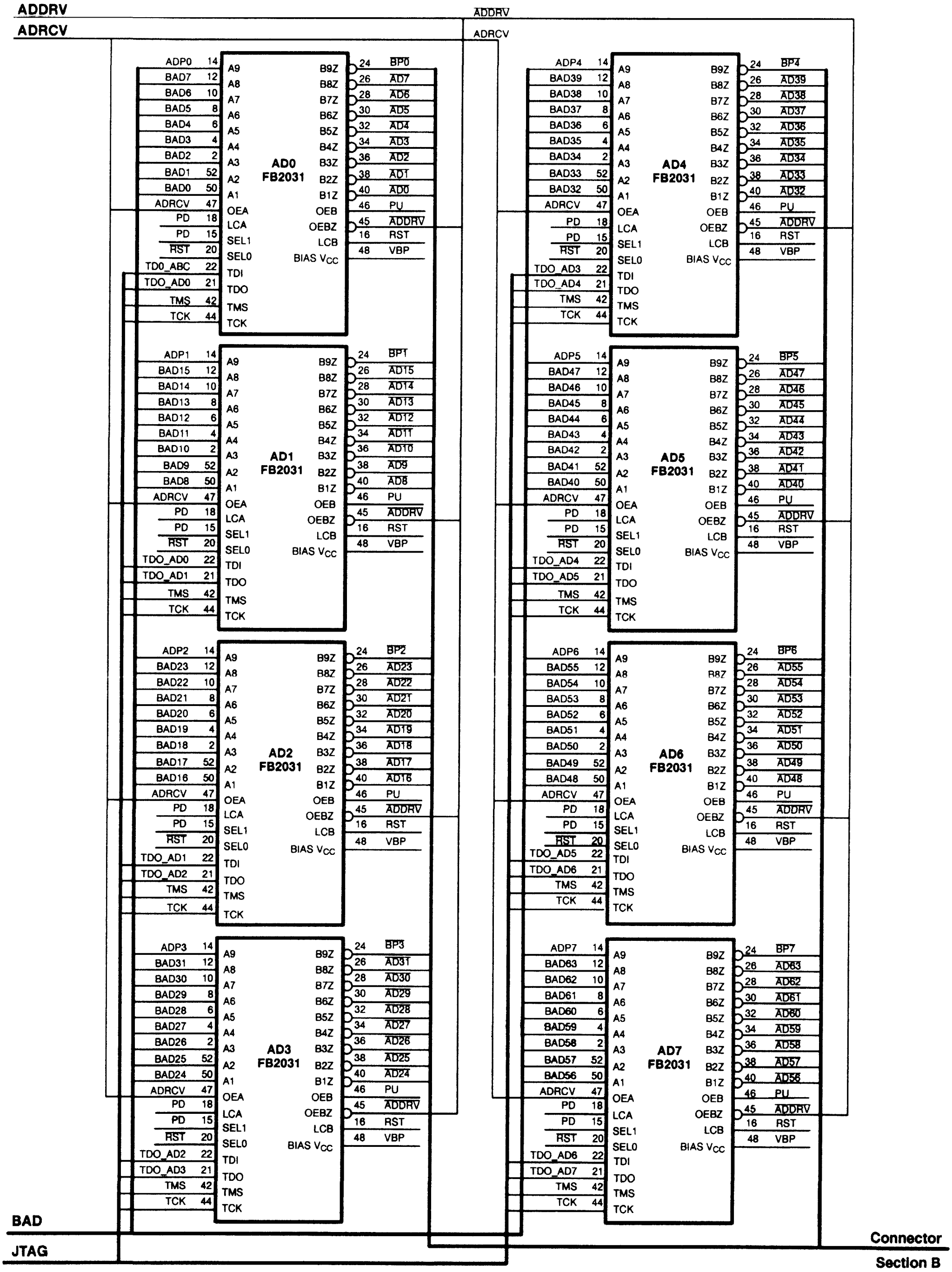


Figure A–6. Interconnect

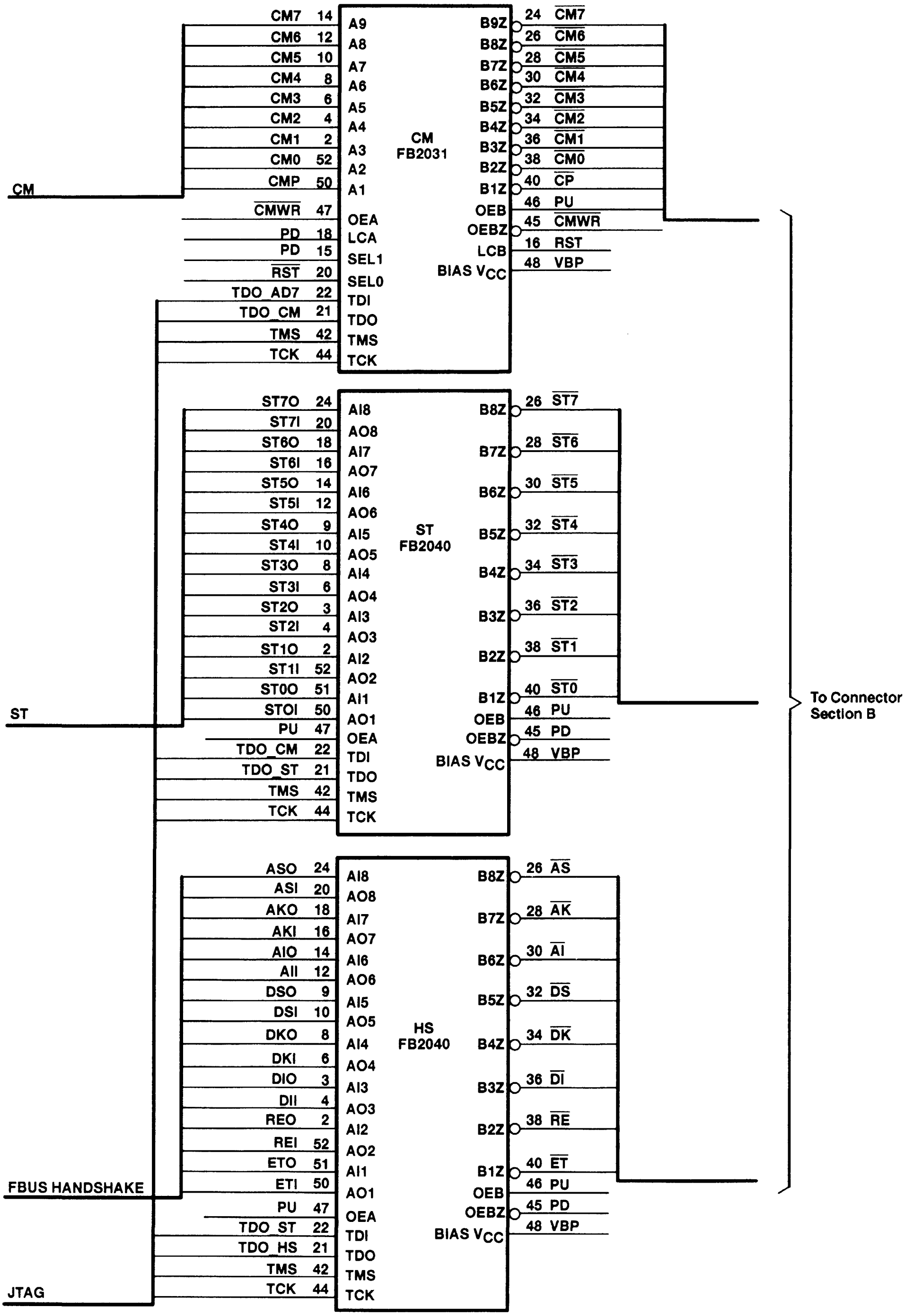


Figure A–7. Interconnect

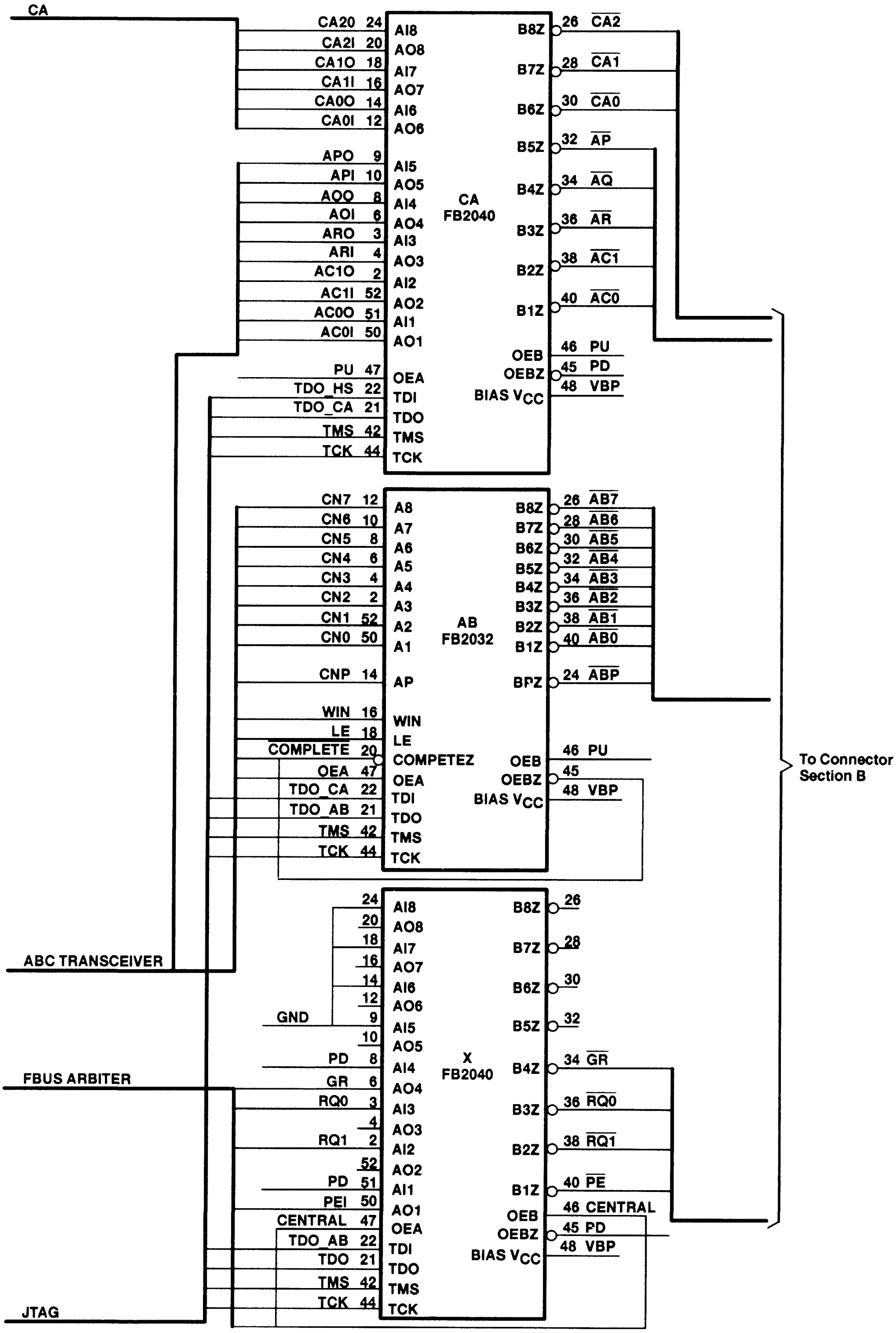


Figure A–8. Interconnect

Appendix B

CSR Bus Interface Application Note

Appendix B
CSR Bus Interface Application Note

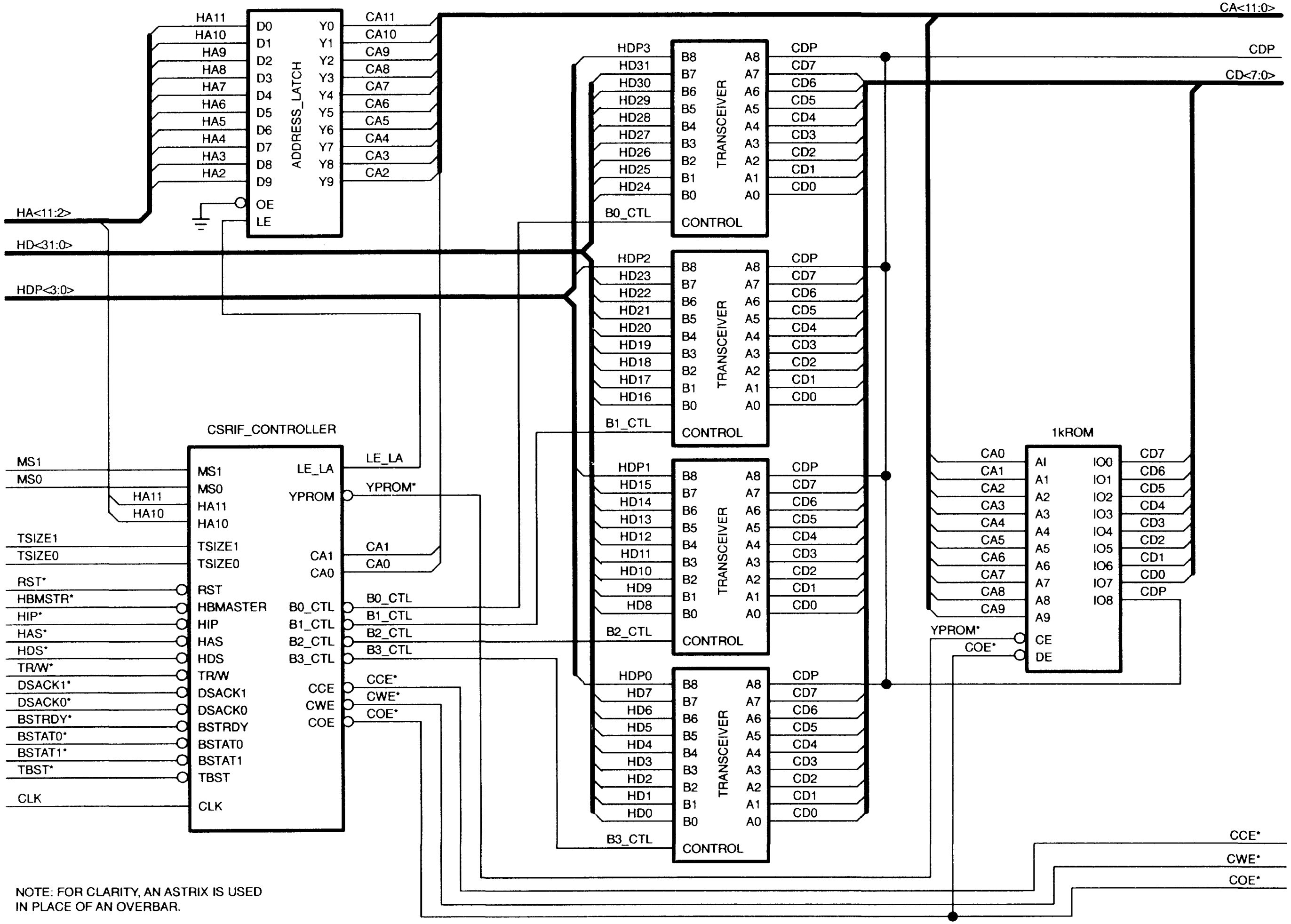
CSR Register Mapping

The control and status registers (CSRs) support the FB+ memory map, device configuration, message mailboxes, and many other functions. These register values are programmed and/or read through the CSR bus. The Futurebus+ controllers' (IOC, ABC, DPU, and PCBA) registers are accessed through the 4K CSR register space associated with a Futurebus+ node. Although the controllers do not require a full 4K register space, the entire 12-bit CSR offset (4K initial node space address) for the node is decoded by the controllers.

CSR core and Futurebus+ dependent CSRs implemented in the controllers are mapped directly according to IEEE standard 896.2. Initial unit space register mapping has been coordinated with other TI Futurebus+ products and standard CSR devices, so addresses do not conflict. Data output drivers are only enabled for reads of implemented registers. This allows chip enables of CSR devices to be tied low and minimizes address decoding requirements if the predefined register mapping is acceptable. The module designer may however use chip enables to re-map the registers into a different address space if needed.

Only one device on the node is expected to respond for reads to any one CSR address. Three Futurebus+ dependent CSRs (logical module control, logical common control, and bus propagation delay) have bits that are shared with other devices or are partially implemented in other devices. Typically, the device that uses the majority of the bits in the respective CSR register maintains all the bits for that register and will respond to a read of that address.

SCOPE, Widebus, UBT, and EPIC-IIB are trademarks of Texas Instruments Incorporated.


ADDRESS_LATCH
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 OE LE
Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9
HA11 HA10 HA9 HA8 HA7 HA6 HA5 HA4 HA3 HA2
CA11 CA10 CA9 CA8 CA7 CA6 CA5 CA4 CA3 CA2
HA<11:2>
HD<31:0>
HDP<3:0>
CA<11:0>
CDP
CD<7:0>
CSRIF_CONTROLLER
MS1 MS0 HA11 HA10 TSIZE1 TSIZE0 RST HBMASTER HIP HAS HDS TR/W DSACK1 DSACK0 BSTRDY BSTAT0 BSTAT1 TBST CLK
LE_LA YPROM CA1 CA0 B0_CTL B1_CTL B2_CTL B3_CTL CCE CWE COE
MS1
MS0
TSIZE1
TSIZE0
RST*
HBMSTR*
HIP*
HAS*
HDS*
TR/W*
DSACK1*
DSACK0*
BSTRDY*
BSTAT0*
BSTAT1*
TBST*
CLK
LE_LA
YPROM*
CA1
CA0
B0_CTL
B1_CTL
B2_CTL
B3_CTL
CCE*
CWE*
COE*
TRANSCEIVER
B8 B7 B6 B5 B4 B3 B2 B1 B0 CONTROL
A8 A7 A6 A5 A4 A3 A2 A1 A0
HDP3 HD31 HD30 HD29 HD28 HD27 HD26 HD25 HD24 B0_CTL
HDP2 HD23 HD22 HD21 HD20 HD19 HD18 HD17 HD16 B1_CTL
HDP1 HD15 HD14 HD13 HD12 HD11 HD10 HD9 HD8 B2_CTL
HDP0 HD7 HD6 HD5 HD4 HD3 HD2 HD1 HD0 B3_CTL
CDP CD7 CD6 CD5 CD4 CD3 CD2 CD1 CD0
1kROM
AI A1 A2 A3 A4 A5 A6 A7 A8 A9 CE DE
IO0 IO1 IO2 IO3 IO4 IO5 IO6 IO7 IO8
CA0 CA1 CA2 CA3 CA4 CA5 CA6 CA7 CA8 CA9
CD7 CD6 CD5 CD4 CD3 CD2 CD1 CD0 CDP
YPROM*
COE*
CCE*
CWE*
COE*

NOTE: FOR CLARITY, AN ASTRIX IS USED
IN PLACE OF AN OVERBAR.

Figure B–1. Block Diagram of a Simple CSR Bus Interface

CSR Bus Interface Discrete Implementation

The CSR bus interface converts host interface (HIF) transactions into CSR bus transactions. Figure B–1 shows a block diagram of a simple CSR bus interface. It consists of a controller block, 4 bi-directional latching transceivers, an address latch, and Futurebus+ required 1K × 8 ROM in the initial unit space.

The controller block contains an HIF slave state machine, a byte stacker/unstacker state machine, and a CSR ROM decoder. The CSR bus interface monitors HIF strobes and attributes to respond to an HIF read or write. In its simplest implementation, it decodes the memory select lines (MS<1:0>) to determine when the HIF transaction is in CSR space. MS<1:0> is produced by the DPU's internal address decoders. After the HIF transaction has started, the byte stacker/unstacker state machine performs the following actions:

1. Chooses the appropriate direction for the tranceivers to service a read or write.
2. Stacks or unstacks the bytes in the case of a read or write respectively while asserting the appropriate byte address (CA<1:0>) and CSR bus strobes.

After the byte stacker/unstacker state machine is finished it signals the HIF state machine to acknowledge the HIF transaction.

An address latch is provided to minimize capacitive loading of the HIF by CSR bus devices.

The Futurebus+ CSR ROM is shown attached to the CSR bus. Decode is provided by the CSR interface controller (CSRIF).

Since the MS<1:0> lines provide decode of the extended unit space, The controller block could be enhanced to provide service for devices mapped into extended unit space and attached to the CSR bus. Of course since the extended unit space must not conflict with the initial unit space, a chip enable independent of the ABC/IOC chip enable (CCE*) would have to be generated.

CSR Bus Interface FPGA Implementation

This application note describes a simple and inexpensive implementation of the CSR bus interface (CSRIF). The CSRIF is a host interface (HIF) to CSRIF bridge. It is required by TI's Futurebus+ chip set to translate HIF transactions to CSRIF transactions.

All transactions to CSR space are A32 – D32. The DPU's internal CSRs are accessed from Futurebus+ or the host interface. However, the IOC's and ABC's CSRs are only accessible from the A12 – D8 CSR bus interface. The CSR bus can also service the vendor ROM, vendor defined initial units, and extended units.

TI's family of field programmable gate arrays (FPGA) provide excellent products to implement a wide variety of CSR bus interfaces. This example design provides the following features:

- HIF slave state machine
- CSR bus master state machine
- Read and write capability

Since the design was developed using ViewLogic's Workview tools. Verilog-HDL, and Synopsys synthesis, a customer can easily create more capable derivatives using parts with a higher pin count if necessary.

The functional groups of signals the CSRIF needs to deal with are shown below:

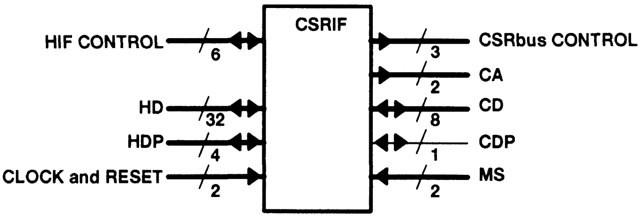


Figure B–2. CSR Bus Interface Functional Signal Groups

Figure B–3 is a schematic symbol that shows the pin detail including physical pin numbers for the CSR bus interface implemented in a TI FPGA. Part number TPC1020 AFN – 084C1

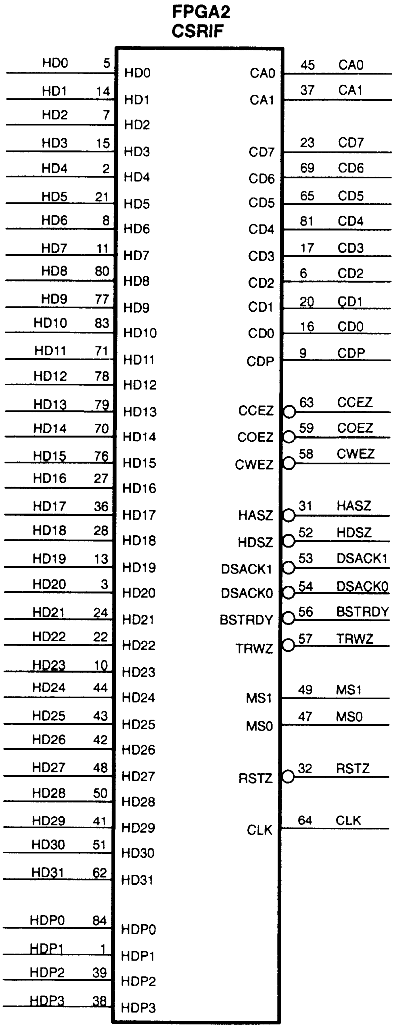


Figure B–3. CSR Bus Interface Schematic Symbol

Figure B–4 is a high level block diagram that shows the major portions of the CSR bus interface.

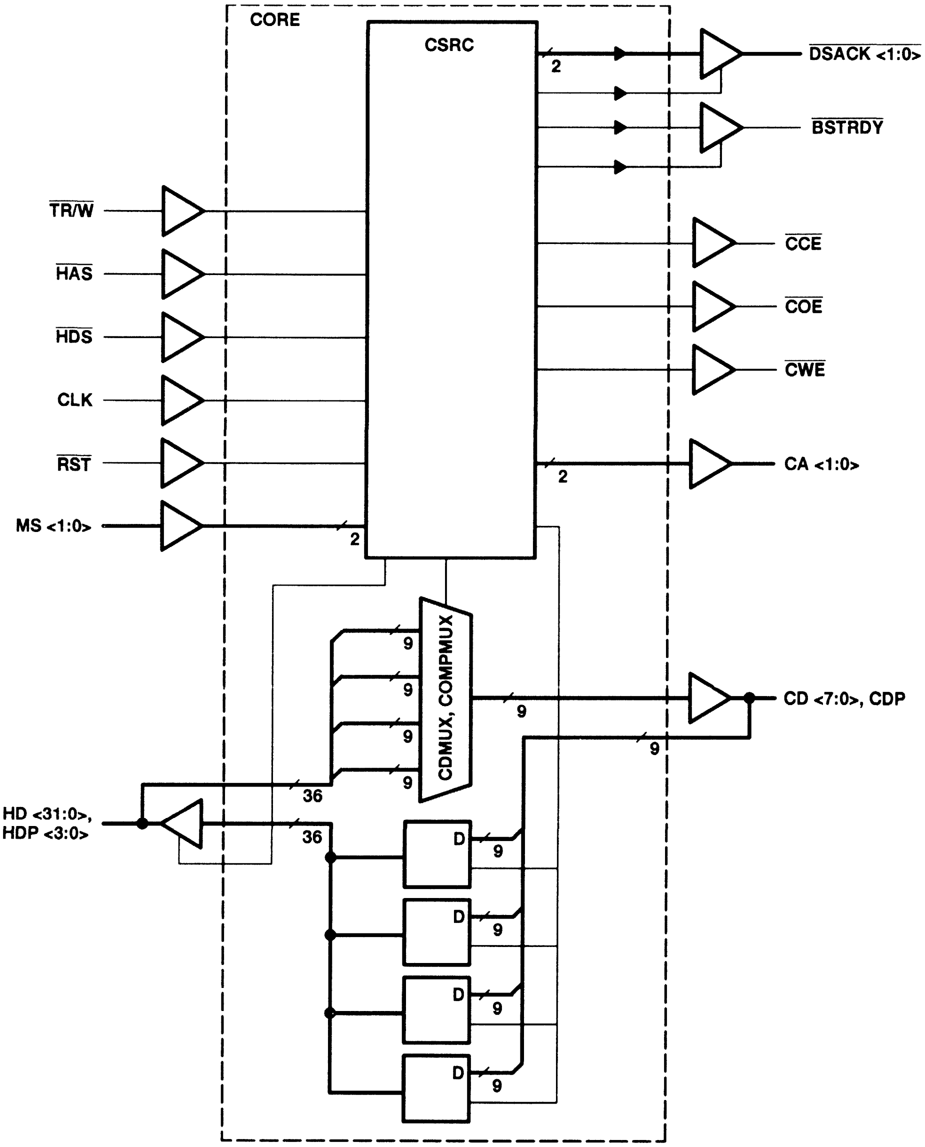


Figure B–4. CSR Bus Interface Functional Block Diagram

Pushing one level into the hierarchy, Figure B–5 shows the I/O ring and core. The core is connected to the input, ouput, clock, and bi-directional buffers in the I/O ring. The core also drives bi-directional buffer control signals.

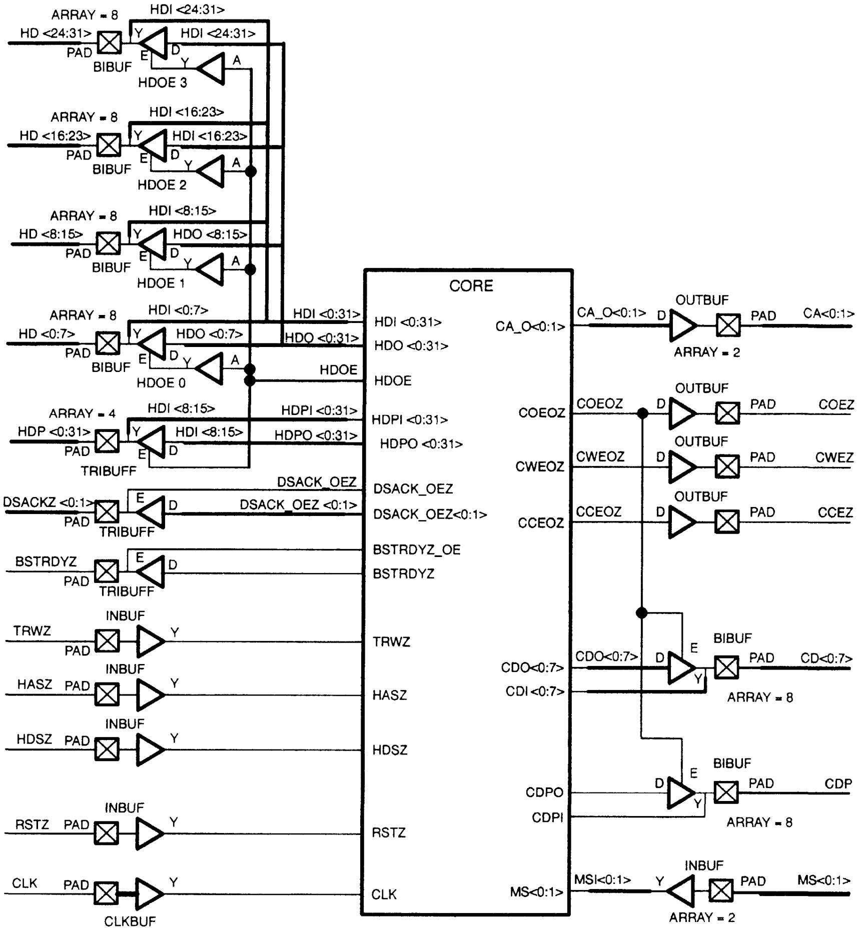


Figure B–5. Top Level Viewlogic Schematic

The second level of the hierarchy is comprised of two schematic sheets. Sheet one contains the controller block and multiplexers used during writes. The blocks CDMUX and CDPMUX route the appropriate bytes of the HIF data bus and parity to the CSR bus data bus parity. Sheet one is shown in Figure B–6 below:

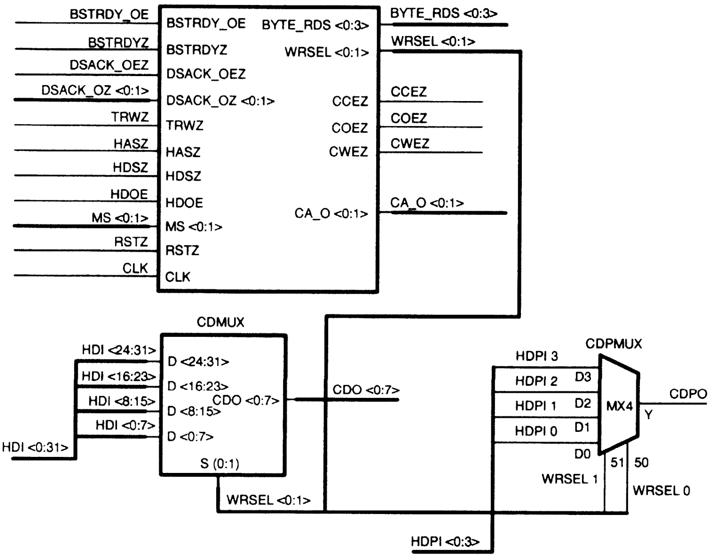


Figure B–6. Sheet One Of Two Of The Core Block

The schematic below shows the implementation for CDMUX which is a byte-wide four-to-one multiplexer.

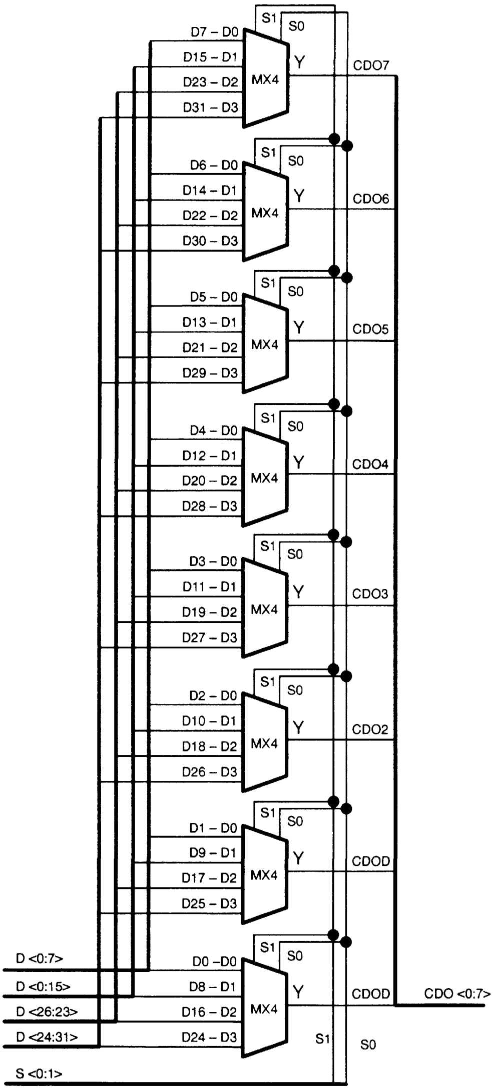


Figure B–7. Sheet One Of One Of The CDMUX Block

Sheet two contains the latches used during reads These latches capture four bytes with byte addresses zero, one, two, and three from the CSR bus data bus and sources them to the HIF data bus as the data for a single CSR read. Sheet two is shown in Figure B–8 below.

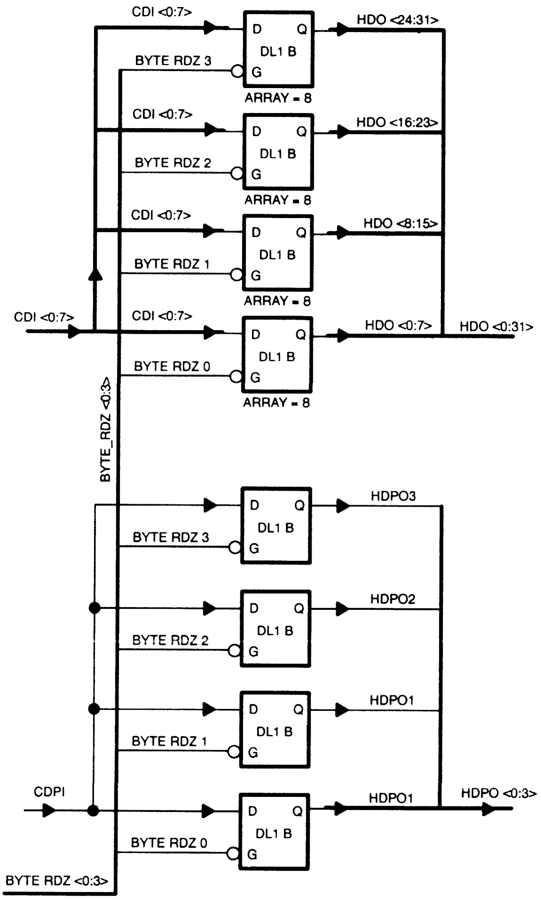


Figure B–8. Sheet Two Of Two Of The Core Block

The third level of the hierarchy contains the HIF transaction start detector, CSR bus byte address counter, write byte select counter, and the CSRIF state machine. The schematic is shown in Figure B–9.

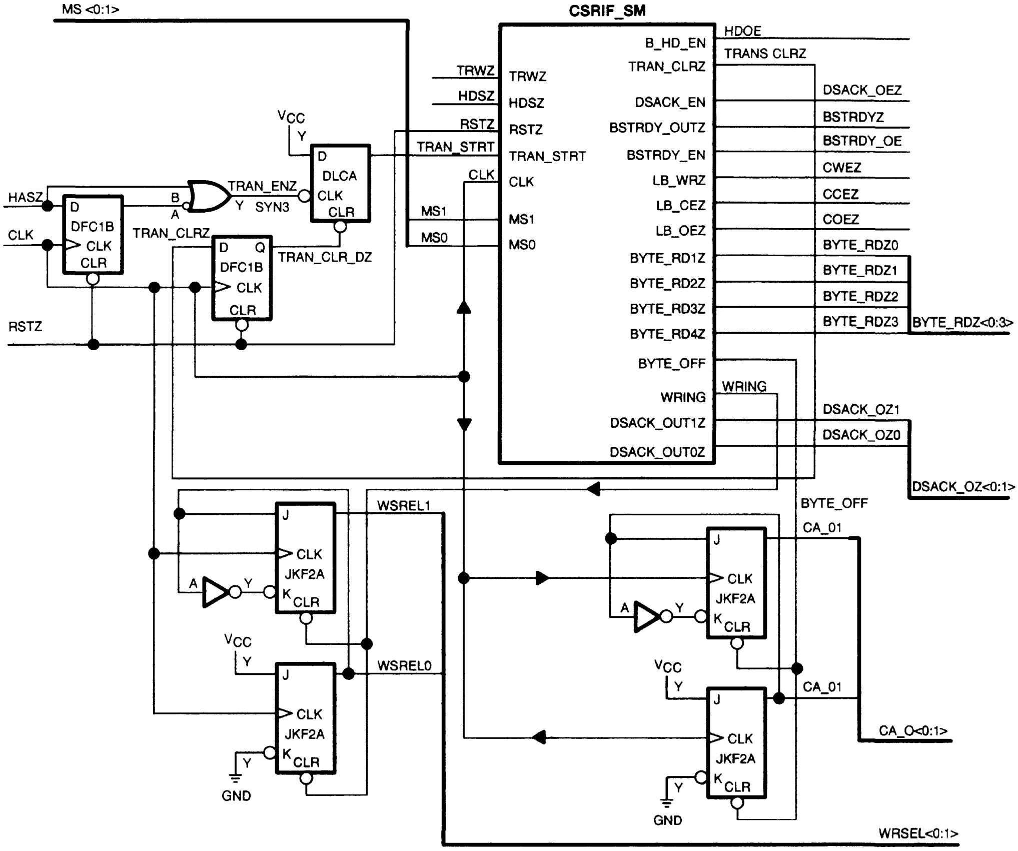


Figure B–9. Sheet One Of One Of The CSRC Block

The schematic for the CSRIF state machine is shown below. This state machine's instance name is LSRIF_SM in Figure B–9.

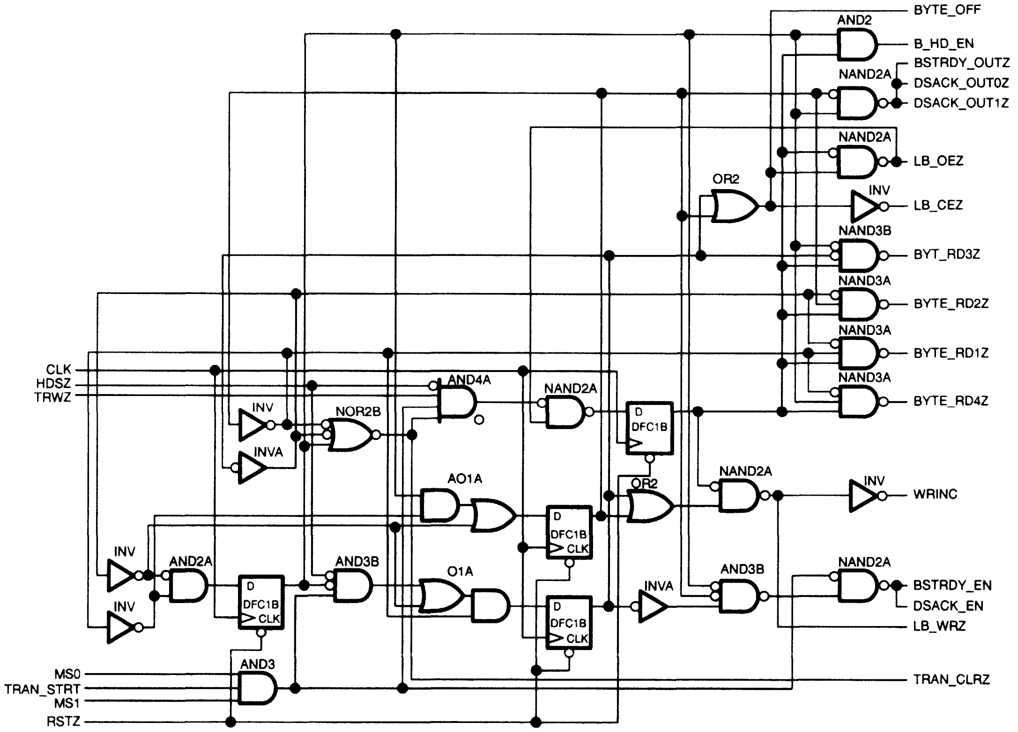


Figure B–10. Sheet One Of One Of The CSRIF_SM Block

This state machine was first described as a register transfer level (RTL) Verilog module. It was synthesized (mapped to gates) using the Synopsys design compiler. The schematic shown in Figure B–10 is machine generated but it can be captured by hand. The Verilog module is shown below.

```
//******************************************************************************
// Module Definition
//******************************************************************************
module CSRIF_SM(

          // Inputs
                               MS1     , // Memory Select
                               MS0     , // Memory Select
                               TRWZ   , // HBus Read Write
                               HDSZ   , // HBus Data Strobe
                               RSTZ   , // HBus Reset
                               CLK     , // HBus Clock
                               TRAN_STRT   ,

          // Outputs
                               B_HD_EN   ,
                               TRAN_CLRZ   ,
                               DSACK_OUT1Z   ,
                               DSACK_OUT0Z   ,
                               DSACK_EN   ,
                               BSTRDY_OUTZ   ,
                               BSTRDY_EN   ,
                               LB_WRZ      ,
                               LB_CEZ      ,
                               LB_OEZ      ,
                               BYTE_RD1Z   ,
                               BYTE_RD2Z   ,
                               BYTE_RD3Z   ,
                               BYTE_RD4Z   ,
                               BYTE_OFF    ,
                               WRINC       ,
                   );

input
  MS1 ,
  MS0 ,
  TRWZ ,
  HDSZ ,
  RSTZ ,
  CLK ,
  TRAN_STRT ,

output
  WRINC ,
  BYTE_OFF ,
  B_HD_EN ,
  TRAN_CLRZ ,
  DSACK_EN ,
  BSTRDY_EN ,
  BSTRDY_OUTZ ,
  LB_WRZ ,
  LB_CEZ ,
  LB_OEZ ,
  DSACK_OUT1Z ,
  DSACK_OUT0Z ,
  BYTE_RD1Z ,
  BYTE_RD2Z ,
  BYTE_RD3Z ,
  BYTE_RD4Z ;
reg
  DSACK_OUT1Z ,
  DSACK_OUT0Z ,
  WRINC ,
  BYTE_OFF ,
  TRAN_CLRZ ,
  LB_WRZ , LB_CEZ , LB_OEZ ,
  B_HD_EN ,
  BYTE_RD1Z , BYTE_RD2Z , BYTE_RD3Z, BYTE_RD4Z ,
  BSTRDY_OUTZ , DSACK_EN , BSRTDY_EN ;
reg [3:0]
  nx_st ;
reg [3:0]
  st_reg ;
DFC1B u4 (.D(nx_st[3]) , .Q(st_reg[3]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u3 (.D(nx_st[2]) , .Q(st_reg[2]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u2 (.D(nx_st[1]) , .Q(st_reg[1]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u1 (.D(nx_st[0]) , .Q(st_reg[0]) , .CLR(RSTZ) , .CLK(CLK)) ;
  always
     begin
        case (st_reg) // synopsys parallel_case full_case
          4'b0000:
             begin : idle
                BYTE_OFF = 1'b0 ;
                WRINC = 1'b0 ;
                LB_WRZ = 1'b1 ;
                LB_CEZ = 1'b1 ;
                LB_OEZ = 1'b1 ;

                DSACK_OUT1Z = 1'b1 ;
                DSACK_OUT0Z = 1'b1 ;
                BSTRDY_OUTZ = 1'b1 ;
                DSACK_EN = 1'b0 ;
                BSTRDY_EN = 1'b0 ;

                 BYTE_RD1Z = 1'b1 ;
                 BYTE_RD2Z = 1'b1 ;
                 BYTE_RD3Z = 1'b1 ;
                 BYTE_RD4Z = 1'b1 ;

                 TRAN_CLRZ = 1'b1 ;
        B_HD_EN = 1'b1 ;

        if ((TRAN_STRT == 1'b1) && (MS1 == 1'b1) && (MS0 == 1'b1))
           begin
              DSACK_EN = 1'b1 ;
              BSTRDY_EN = 1'b1 ;
              if ((TRWZ == 1'b0) && (HDSZ == 1'b0))
                nx_st = 4'b0001 ;  //csr write
              else if ((TRWZ == 1'b1) && (HDSZ == 1'b0))
                nx_st = 4'b1001 ;  //csr read
              else
                nx_st = 4'b0000 ;
           end
        else
           nx_st = 4'b0000 ;  //stay in idle state
    end

// ---------- csr_write ----------
4'b0001 :
  begin : csr_write1
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b1 ;
    LB_WRZ = 1'b0 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0011 ;  //csr_write2
  end

4'b0011 :
  begin : csr_write2
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b1 ;
    LB_WRZ = 1'b0 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b1 ;
```

```
        DSACK_OUT1Z = 1'b1 ;
        DSACK_OUT0Z = 1'b1 ;
        BSTRDY_OUTZ = 1'b1 ;
        DSACK_EN = 1'b1 ;
        BSTRDY_EN = 1'b1 ;

        BYTE_RD1Z = 1'b1 ;
        BYTE_RD2Z = 1'b1 ;
        BYTE_RD3Z = 1'b1 ;
        BYTE_RD4Z = 1'b1 ;

        TRAN_CLRZ = 1'b0 ;
        B_HD_EN = 1'b0 ;

        nx_st = 4'b0010 ;  //csr_write3
    end

  4'b0010 :
    begin : csr_write3
        BYTE_OFF = 1'b1 ;
        WRINC = 1'b1 ;
        LB_WRZ = 1'b0 ;
        LB_CEZ = 1'b0 ;
        LB_OEZ = 1'b1 ;

        DSACK_OUT1Z = 1'b1 ;
        DSACK_OUT0Z = 1'b1 ;
        BSTRDY_OUTZ = 1'b1 ;
        DSACK_EN = 1'b1 ;
        BSTRDY_EN = 1'b1 ;

        BYTE_RD1Z = 1'b1 ;
        BYTE_RD2Z = 1'b1 ;
        BYTE_RD3Z = 1'b1 ;
        BYTE_RD4Z = 1'b1 ;

        TRAN_CLRZ = 1'b0 ;
        B_HD_EN = 1'b0 ;

        nx_st = 4'b0110 ;  //csr_write4
    end


  4'b0110 :
    begin : csr_write4
        BYTE_OFF = 1'b1 ;
        WRINC = 1'b1 ;
        LB_WRZ = 1'b0 ;
        LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0100 ;
  end

4'b0100 :
  begin : csr_write_dsack
    BYTE_OFF = 2'b0 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b1 ;
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b0 ;
    DSACK_OUT0Z = 1'b0 ;
    BSTRDY_OUTZ = 1'b0 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0000 ;   // go to idle state
  end

// ---------- csr_read ----------
4'b1001 :
  begin : csr_read1
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
```

```
      LB_CEZ = 1'b0 ;
      LB_OEZ = 1'b0 ;

      DSACK_OUT1Z = 1'b1 ;
      DSACK_OUT0Z = 1'b1 ;
      BSTRDY_OUTZ = 1'b1 ;
      DSACK_EN = 1'b1 ;
      BSTRDY_EN = 1'b1 ;

      BYTE_RD1Z = 1'b0 ;
      BYTE_RD2Z = 1'b1 ;
      BYTE_RD3Z = 1'b1 ;
      BYTE_RD4Z = 1'b1 ;

      TRAN_CLRZ = 1'b0 ;
      B_HD_EN = 1'b0 ;

      nx_st = 4'b1011 ;  //csr_read2
    end

  4'b1011 :
    begin : csr_read2
      BYTE_OFF = 1'b1 ;
      WRINC = 1'b1 ;
      LB_WRZ = 1'b1 ;
      LB_CEZ = 1'b0 ;
      LB_OEZ = 1'b0 ;

      DSACK_OUT1Z = 1'b1 ;
      DSACK_OUT0Z = 1'b1 ;
      BSTRDY_OUTZ = 1'b1 ;
      DSACK_EN = 1'b1 ;
      BSTRDY_EN = 1'b1 ;

      BYTE_RD1Z = 1'b1 ;
      BYTE_RD2Z = 1'b0 ;
      BYTE_RD3Z = 1'b1 ;
      BYTE_RD4Z = 1'b1 ;

      TRAN_CLRZ = 1'b0 ;
      B_HD_EN = 1'b0 ;

      nx_st = 4'b1010 ;  //csr_read3
    end

  4'b1010 :
    begin : csr_read3
      BYTE_OFF = 1'b1 ;
      WRINC = 1'b0 ;
      LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b0 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b0 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b1110 ;  //csr_read4
  end

4'b1110 :
  begin : csr_read4
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b0 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b0 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b1 ;

    nx_st = 4'b1100 ;
  end

4'b1100 :
  begin : csr_read_dsack
    BYTE_OFF = 1'b0 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
```

```
              LB_CEZ = 1'b1 ;
              LB_OEZ = 1'b1 ;

              DSACK_OUT1Z = 1'b0 ;
              DSACK_OUT0Z = 1'b0 ;
              BSTRDY_OUTZ = 1'b0 ;
              DSACK_EN = 1'b1 ;
              BSTRDY_EN = 1'b1 ;

              BYTE_RD1Z = 1'b1 ;
              BYTE_RD2Z = 1'b1 ;
              BYTE_RD3Z = 1'b1 ;
              BYTE_RD4Z = 1'b1 ;

              TRAN_CLRZ = 1'b0 ;
              B_HD_EN = 1'b1 ;

              nx_st = 4'b0000 ;  //go to idle state
            end

         endcase
      #1;
      end  // end of sm

endmodule
```

Figure B–11. Verilog Module

This is a minimal implementation of the CSR bus interface. There are several enhancements that could be added by the board designer:

- This design expects the CSR bus word address (CA<11:2>) to be directly connected to the host interface CSR word address (HA<11:2>). The CSRIF generates only the CSR bus byte address (CA<1:0>). If the board design places enough loading on the CSR bus to degrade the performance of the HIF, CA<11:2> can be isolated from HA<11:2> with a latch that is transparent when HIP* is not asserted. This can be implemented discretely with a 10-bit latch or on chip by selecting an FPGA that would accommodate the extra 21 pins required.
- If the vendor ROM is a 1 k × 8 device connected to the CSR bus, a chip select for the ROM can be generated by adding HA<11:10> as inputs and using them along with MS<1:0> as inputs to a decoder. The output of the decoder would function as the chip enable for the vendor ROM.
- Vendor defined initial units (CSR registers) could be implemented in the CSRIF FPGA. These registers can be embedded in the chip by bringing the HIF CSR word address (<HA11:2>) on chip for use as a register address decode.
- The CSRIF could provide service to extended units on the CSR bus by decoding MS = HL (local extended units along with MS = HH (local CSR)). This would echo transactions with addresses that fall within unit base and units bound onto the CSR bus. The HIF address would have to be decoded to the proper granularity to support the memory map of the existing extended units.

Appendix C

TI Futurebus+ Chip Set IEEE 1149.1 Description

Appendix C
TFB20002B, TFB2010, TFB2022A, JTAG Description

INTRODUCTION

The TFB2002B, TFB2010, and TFB2022A Futurebus+ devices are members of the Texas Instruments SCOPE™ testability IC family. This family of devices supports IEEE Standard 1149.1-1990 boundary scan to facilitate testing of complex circuit-board assemblies. Scan access to the test circuitry is accomplished via the 4-wire test access-port (TAP) interface.

In the normal mode, these devices operate as Futurebus+ functions. The test circuitry can be activated by the TAP to take snapshot samples of the data appearing at the device pins or to perform a self test on the boundary test cells. Activating the TAP in the normal mode does not affect the functional operation of the Futurebus+ devices.

In the test mode, the normal operation of the Futurebus+ devices is inhibited and the test circuitry is enabled to observe and control the I/O boundary of the device. When enabled, the test circuitry can perform boundary scan test operations as described in IEEE Standard 1149.1-1990.

Four dedicated test pins are used to control the operation of the test circuitry. The are test data input (TDI), test data output (TDO), test-mode select (TMS), and test clock (TCK). All testing and scan operations are synchronized to the TAP interface.

Overview

A functional block diagram of the IEEE 1149.1 test architecture standard is shown in Figure C–1. The implementation includes an 8-bit instruction register, a one-bit bypass register, and a boundary-scan register.

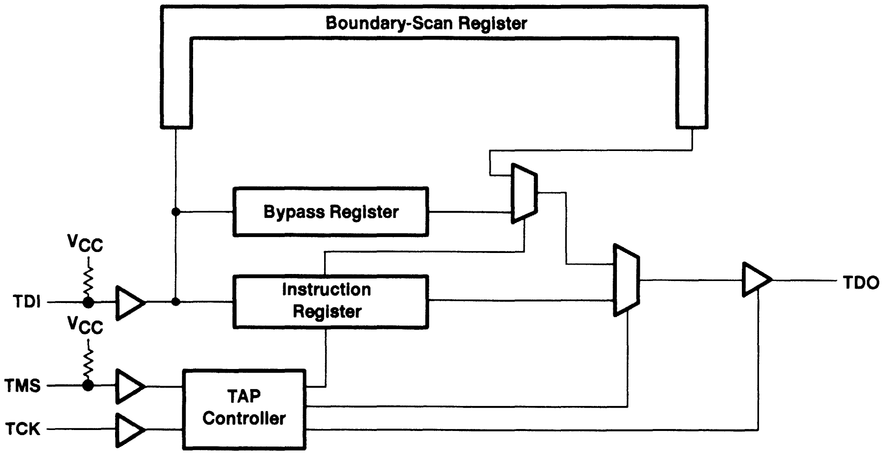


Figure C–1. Functional Block Diagram

SCOPE is a trademark of Texas Instruments Incorporated.

Terminal Functions

TERMINAL NAME	I/O	DESCRIPTION
GND		Ground
TCK	I	Test clock input. One of four terminals required by IEEE Standard 1149.1-1990. Test operations of the device are synchronous to the test clock. Data is captured on the rising edge of TCK, and outputs change on the falling edge of TCK.
TDI	I	Test data input. One of four terminals required by IEEE Standard 1149.1-1990. The test data input is the serial input for shifting data through the instruction register or selected data register. An internal pullup forces TDI to a high level if left unconnected.
TDO	O	Test data output. One of four terminals required by IEEE Standard 1149.1-1990. The test data output is the serial output for shifting data through the instruction register or selected data register.
TMS	I	Test-mode select input. One of four terminals required by IEEE Standard 1149.1-1990. The test-mode select input directs the device through its test access port (TAP) controller states. An internal pullup forces TMS to a high level if left unconnected.
V_{CC}		Supply voltage

Test Architecture

Serial test information is conveyed by means of a 4-wire test bus, or test-access port (TAP), that conforms to IEEE Standard 1149.1-1990. Test instructions, test data, and test control signals are all passed along this serial test bus. The TAP controller monitors two signals from the test bus, namely TCK and TMS. The function of the TAP controller is to extract the synchronization (TCK) and state control (TMS) signals from the test bus and generate the appropriate on-chip control signals for the test structures in the device. Figure C–2 shows the TAP controller state diagram.

The TAP controller is fully synchronous to the TCK signal. Input data is captured on the rising edge of TCK, and output data changes on the falling edge of TCK. This scheme ensures that data to be captured is valid for fully one-half of the TCK cycle.

The functional block diagram in Figure C–1 illustrates the IEEE Standard 1149.1-1990 4-wire test bus and boundary-scan architecture and the relationship between the test bus, the TAP controller, and the test registers. As illustrated, the device contains an 8-bit instruction register and two test-data registers: a boundary-scan register and a bypass register.

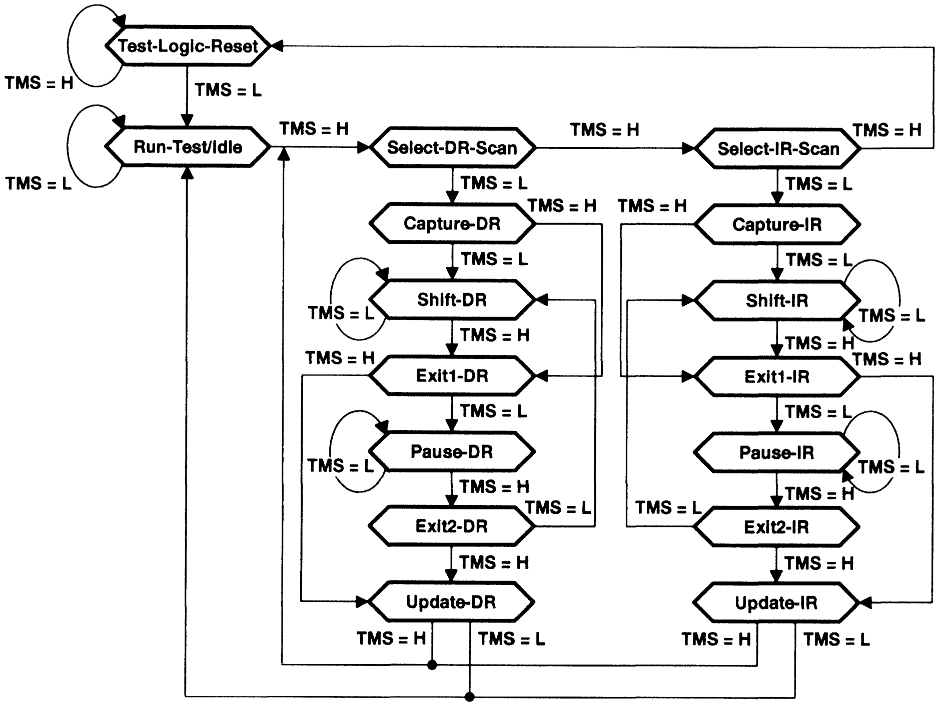


Figure C–2. TAP Controller State Diagram

STATE DIAGRAM DESCRIPTION

The test-access port (TAP) controller specified in IEEE Standard 1149.1-1990 is a synchronous finite state machine that provides test control signals throughout the device. The TAP controller proceeds through its states based on the level of TMS at the rising edge of TCK.

As illustrated, the TAP controller consists of sixteen states. There are six stable states (indicated by a looping arrow in the state diagram) and ten unstable states. A stable state is defined as a state the TAP controller can retain for consecutive TCK cycles. Any state that does not meet this criterion is an unstable state.

There are two main paths through the state diagram: one to access and control the selected data register, and one to access and control the instruction register. Only one register may be accessed at a time.

Test-Logic-Reset

The device powers up in the test-logic-reset state. In the stable test-logic-reset state, the test logic is reset and is disabled so that the normal logic function of the device is performed. The instruction register is reset to an opcode that selects the bypass instruction.

The state machine is constructed such that the TAP controller returns to the test-logic-reset state in no more than five TCK cycles if TMS is left high. The TMS pin has an internal pullup resistor that forces it high if left unconnected or if a board defect causes it to be open circuited.

Run-Test/Idle

The TAP controller must pass through the run-test/idle state (from test-logic-reset) before executing any test operations. The run-test/idle state may also be entered following data register or instruction register scans. Run-test/idle is provided as a stable state in which the test logic may be actively running a test or may be idle.

Select-DR-Scan, Select-IR-Scan

No specific function is performed in the select-DR-scan and select-IR-scan states, and the TAP controller exits either of these states on the next TCK cycle. These states are provided to allow the selection of either data register scan or instruction register scan.

Capture-DR

When a data register scan is selected, the TAP controller must pass through the capture-DR state. In the capture-DR state, the selected-data register may capture a data value as specified by the current instruction. Such capture operations occur on the rising edge of TCK upon which the TAP controller exits the capture-DR state.

Shift-DR

Upon entry to the shift-DR state, the selected data register is placed in the scan path between TDI and TDO. On the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO enables to the logic level present in the least significant bit of the selected data register.

While in the stable shift-DR state, data is serially shifted through the selected data register on each TCK cycle. The first shift occurs on the first rising edge of TCK after entry to the shift-DR state (i.e., no shifting occurs during the TCK cycle in which the TAP controller changes from capture-DR to shift-DR or from exit2-DR to shift-DR). The last shift occurs on the rising edge of TCK upon which the TAP controller exits the shift-DR state.

Exit1-DR, Exit2-DR

The exit1-DR and exit2-DR states are temporary states used to end a data register scan. It is possible to return to the shift-DR state from either exit1-DR or exit2-DR without recapturing the data register. On the first falling edge of TCK after entry to exit1-DR, TDO goes from the active state to the high-impedance state.

Pause-DR

No specific function is performed in the stable pause-DR state, in which the TAP controller can remain indefinitely. The pause-DR state provides the capability of suspending and resuming data-register scan operations without loss of data.

Update-DR

If the current instruction calls for the selected data register to be updated with current data, then updates occur on the falling edge of TCK following entry to the update-DR state.

Capture-IR

When an instruction register scan is selected, the TAP controller must pass through the capture-IR state. In the capture-IR state, the instruction register captures its current status value. This capture operation occurs on the rising edge of TCK, and the TAP controller exits the capture-IR state. For these devices, the status value loaded in the capture-IR state is the fixed binary value 10000001.

Shift-IR

Upon entry to the shift-IR state, the instruction register is placed in the scan path between TDI and TDO, and on the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO enables to the logic level present in the least significant bit of the instruction register.

While in the stable shift-IR state, instruction data is serially shifted through the instruction register on each TCK cycle. The first shift occurs on the first rising edge of TCK after entry to the shift-IR state (i.e., no shifting occurs during the TCK cycle in which the TAP controller changes from capture-IR to shift-IR or from exit2-IR to shift-IR). The last shift occurs on the rising edge of TCK upon which the TAP controller exits the shift-IR state.

Exit1-IR, Exit2-IR

The exit1-IR and exit2-IR states are temporary states used to end an instruction register scan. It is possible to return to the shift-IR state from either exit1-IR or exit2-IR without recapturing the instruction register. On the first falling edge of TCK after entry to exit1-IR, TDO goes from the active state to the high-impedance state.

Pause-IR

No specific function is performed in the stable pause-IR state, in which the TAP controller can remain indefinitely. The pause-IR state provides the capability of suspending and resuming instruction register scan operations without loss of data.

Update-IR

The current instruction is updated and takes effect on the falling edge of TCK following entry to the update-IR state.

REGISTER OVERVIEW

With the exception of the bypass register, any test register may be thought of as a serial shift register with a shadow latch on each bit. The bypass register differs in that it contains only a shift register. During the appropriate capture state (capture-IR for instruction register, capture-DR for data registers), the shift register may be parallel loaded from a source specified by the current instruction. During the appropriate shift state (shift-IR or shift-DR), the contents of the shift register are shifted out from TDO while new contents are shifted in at TDI. During the appropriate update state (update-IR or update-DR), the shadow latches are updated from the shift register.

INSTRUCTION-REGISTER DESCRIPTION

The instruction register (IR) is eight bits long and is used to tell the device what instruction is to be executed. Information contained in the instruction includes the mode of operation (either normal mode in which the device performs its normal logic function, or test mode in which the normal logic function is inhibited or altered), the test operation to be performed, which of the two data registers is to be selected for inclusion in the scan path during data-register scans, and the source of data to be captured into the selected data register during capture-DR.

Table C–4 lists the instructions supported by these devices. The even-parity feature specified for SCOPE devices are supported in these devices. Bit 7 of the instruction opcode is the parity bit. Any instructions that are defined for SCOPE devices but are not supported by these devices default to bypass.

During capture-IR, the IR captures the binary value 10000001. As an instruction is shifted in, this value is shifted out via TDO and can be inspected as verification that the IR is in the scan path. During update-IR, the value that has been shifted into the IR is loaded into shadow latches. At this time, the current instruction is updated, and any specified mode change takes effect. At power up or in the test-logic-reset state, the IR is reset to the binary value 10000001.

The instruction register order of scan is illustrated in Figure C–3.

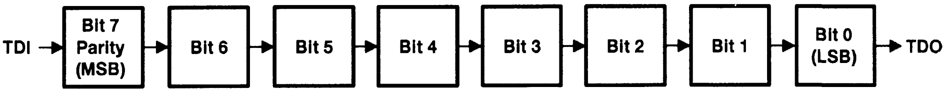


Figure C–3. Instruction Register Order of Scan

DATA-REGISTER DESCRIPTION

Boundary-Scan Register

The boundary-scan register (BSR) is used 1) to store test data that is to be applied internally to the inputs of the normal on-chip logic and/or externally to the device output pins, and/or 2) to capture data that appears internally at the outputs of the normal on-chip logic and/or externally at the device input pins.

The boundary-scan register configurations for the TFB2002B, TFB2022A, and TFB2010 are provided in Tables C–1, C–2, and C–3, respectively. The boundary-scan registers are not preset or cleared at power up or in test-logic-reset. The boundary-scan register order of scan is from TDI to TDO, where boundary-scan cell zero (0) is closest to TDO. The source of data to be captured into the BSR during capture-DR is determined by the current instruction.

Table C–1. TFB2002B Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
146	CLOCK	109	DATAAV*	072	ST0O	035	GR
145	IGNORE*	108	SPACEAV*	071	ST0I	034	RQ0
144	INT*	107	NEWADDR*	070	CM7	033	RQ1
143	HBMASTER*	106	TRW*	069	CM6	032	REFCL*
142	HBGACK*	105	DL1	068	CM5	031	RSTBYPASS*
141	HBG*	104	DL0	067	CM4	030	ARBERR0
140	HBR*	103	DSACK1*	066	CM3	029	ARBERR1
139	LKFLD0	102	§DSOE*	065	CM2	028	BUSI*
138	LKFLD1	101	DSACK0*	064	CM1	027	SYSRST*
137	LKFLD2	100	FADEC3	063	CM0	026	BINIT*
136	HDS*	099	FADEC2	062	CMP	025	RST*
135	DW64*	098	FADEC1	061	¶CMWR*	024	CWE*
134	†HBOE*	097	FADEC0	060	CA2O	023	CCE*
133	TBST*	096	FMODE2	059	CA2I	022	COE*
132	MORE*	095	FMODE1	058	CA1O	021	CD0
131	LK*	094	FMODE0	057	CA1I	020	CDI
130	ERROR0	093	FRD*	056	CA0O	019	CD2
129	ERROR1	092	FSTRB*	055	CA0I	018	CD3
128	FIFORST*	091	FACK*	054	ETO	017	#CDOE*
127	HBADLD*	090	SELECTED*	053	ETI	016	CD4
126	UNALIGNED*	089	BSTRDY*	052	DKO	015	CD5
125	BSTAT1*	088	ADRCV	051	DKI	014	CD6
124	‡BSOE*	087	ADDRV*	050	DIO	013	CD7
123	BSTAT0*	086	ST7O	049	DII	012	CDP
122	TSIZE1	085	ST7I	048	ASO	011	CA0
121	TSIZE0	084	ST6O	047	ASI	010	CA1
120	HSTRB*	083	ST6I	046	AKO	009	CA2
119	HMODE2	082	ST5O	045	AKI	008	CA3
118	HMODE1	081	ST5I	044	DSO	007	CA4
117	HMODE0	080	ST4O	043	DSI	006	CA5
116	HADEC3	079	ST4I	042	AIO	005	CA6
115	HADEC2	078	ST3O	041	AII	004	CA7
114	HADEC1	077	ST3I	040	AQI	003	CA8
113	HADEC0	076	ST2O	039	ARI	002	CA9
112	HAS*	075	ST2I	038	REO	001	CA10
111	HIP*	074	ST1O	037	REI	000	CA11
110	DMAMODE	073	ST1I	036	PEI	—	—

† Active-low output enable for HAS*, HIP*, HDS*, LKFLD<2:0>, LK*, DW64*, TBST*, DL<1:0>,TRW*, and HBGACK*

‡ Active-low output enable for BSTATI* and BSTAT0*

§ Active-low output enable for DSACK1*, DSACK0*, and BSTRDY*

¶ Active-low output enable for CM<7:0> and CMP

Active-low output enable for CD<7:0> and CDP

Table C–2. TFB2022A Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
211	CLOCK	174	AD47	137	AD15	100	DATAAV*
210	MS1	173	AD46	136	AD14	099	DMAMODE
209	MS0	172	AD45	135	AD13	098	HIP*
208	CM5	171	AD44	134	AD12	097	HAS*
207	CM6	170	AD43	133	AD11	096	HADEC0
206	CM7	169	AD42	132	AD10	095	HADEC1
205	ST3	168	AD41	131	AD9	094	HADEC2
204	ST5	167	AD40	130	AD8	093	HADEC3
203	GA0*	166	ADP4	129	ADP0	092	HMODE0
202	DS	165	AD39	128	AD7	091	HMODE1
201	GA1*	164	AD38	127	AD6	090	HMODE2
200	AS	163	AD37	126	AD5	089	HSTRB*
199	GA2*	162	AD36	125	AD4	088	TSIZE1
198	DI	161	AD35	124	AD3	087	TSIZE0
197	GA3*	160	AD34	123	AD2	086	HA0
196	DK	159	AD33	122	AD1	085	HA1
195	GA4*	158	AD32	121	AD0	084	HA2
194	ADP7	157	ADP3	120	BSTRDY*	083	HA3
193	AD63	156	AD31	119	SELECTED*	082	HA4
192	AD62	155	AD30	118	FACK*	081	HA5
191	AD61	154	AD29	117	FSTRB*	080	HA6
190	AD60	153	AD28	116	FRD*	079	HA7
189	AD59	152	AD27	115	FMODE0	078	HA8
188	AD58	151	AD26	114	FMODE1	077	HA9
187	AD57	150	AD25	113	FMODE2	076	HA10
186	AD56	149	AD24	112	FADEC0	075	HA11
185	ADP6	148	ADP2	111	FADEC1	074	HA12
184	AD55	147	AD23	110	FADEC2	073	HA13
183	AD54	146	AD22	109	FADEC3	072	HA14
182	AD53	145	AD21	108	DSACK0*	071	HA15
181	AD52	144	AD20	107	DSACK1*	070	HA16
180	AD51	143	AD19	106	DL0	069	HA17
179	AD50	142	AD18	105	DL1	068	§HAOE*
178	AD49	141	AD17	104	TRW*	067	HA18
177	AD48	140	AD16	103	NEWADDR*	066	HA19
176	†ADHOE*	139	‡ADLOE*	102	REFCLK	065	HA20
175	ADP5	138	ADP1	101	SPACEAV*	064	HA21

† Active-low output control for AD<63:32> and ADP <7:4>

‡ Active-low output control for AD<31:0> and ADP <3:0>

§ Active-low output control for HA<31:0> and HAP <3:0>

Table C–2. TFB2022A Boundary-Register Configuration (Continued)

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
063	HA22	047	HD2	031	†HDOE*	015	HD30
062	HA23	046	HD3	030	HD16	014	HD31
061	HA24	045	HD4	029	HD17	013	HDP3
060	HA25	044	HD5	028	HD18	012	BSTAT0*
059	HA26	043	HD6	027	HD19	011	BSTAT1*
058	HA27	042	HD7	026	HD20	010	TEST_PIN1
057	HA28	041	HDP0	025	HD21	009	TEST_PIN2
056	HA29	040	HD8	024	HD22	008	UNALIGNED*
055	HA30	039	HD9	023	HD23	007	HBADLD*
054	HA31	038	HD10	022	HDP2	006	FIFORST*
053	HAP0	037	HD11	021	HD24	005	ERROR1
052	HAP1	036	HD12	020	HD25	004	ERROR0
051	HAP2	035	HD13	019	HD26	003	SYSRST*
050	HAP3	034	HD14	018	HD27	002	BINIT*
049	HD0	033	HD15	017	HD28	001	RST*
048	HD1	032	HDP1	016	HD29	000	‡HBMASTER*

† Active-low output control for HD<31:0> and HDP <3:0>

‡ Active-low output control for TSIZE1 and TSIZE0

Table C–3. TFB2010 Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
71	GA0*	53	CD7	35	ARBERR0	17	CMPT*
70	GA1*	52	CD6	34	RQ1	16	OEA
69	GA2*	51	CD5	33	RQ0	15	API
68	GA3*	50	†CDOE*	32	GR	14	APO
67	GA4*	49	CD4	31	PE	13	AQI
66	CA11	48	CD3	30	CENTRAL	12	AQO
65	CA10	47	CD2	29	CN0	11	ARI
64	CA9	46	CD1	28	CN1	10	ARO
63	CA8	45	CD0	27	CN2	09	AC1I
62	CA7	44	COE*	26	CN3	08	AC1O
61	CA6	43	CCE*	25	‡CNOE*	07	AC0I
60	CA5	42	CWE*	24	CN4	06	AC0O
59	CA4	41	REFCLK	23	CN5	05	REI
58	CA3	40	RST*	22	CN6	04	ASI
57	CA2	39	BINIT*	21	CN7	03	REF
56	CA1	38	SYSRST*	20	CNP	02	INT*
55	CA0	37	BUSI*	19	WIN	01	PFAIL*
54	CDP	36	ARBERR1	18	LE*	00	CLOCK

† Active-low output control for CD<7:0>, CDP

‡ Active-low output control for CN<7:0>, CNP

Bypass Register

The bypass register is a one-bit scan path that can be selected to shorten the length of the system scan path, thereby reducing the number of bits per test pattern that must be applied to complete a test operation.

During capture-DR, the bypass register captures a logic 0. The bypass register order of scan is illustrated in Figure C–4.

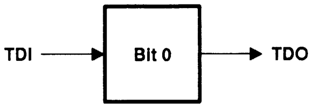


Figure C–4. Bypass Register Order of Scan

INSTRUCTION-REGISTER OPCODE DESCRIPTION

The instruction-register opcodes are shown in Table C–4. The following descriptions detail the operation of each instruction.

Table C–4 Instruction Register Opcodes

BINARY CODE† BIT 7 → BIT 0 MSB → LSB	OPCODE	DESCRIPTION	SELECTED-DATA REGISTER	MODE
00000000	Extest	External test	Boundary-scan	Test
10000010	Sample/Preload	Sample boundary	Boundary-scan	Normal
11111111	Bypass	Bypass scan	Bypass	Normal
00001010	Readbn	Boundary read	Boundary-scan	Normal
10001011	Readbt	Boundary read	Boundary-scan	Test
00001100	Celltst	Boundary self test	Boundary-scan	Normal
All others	Bypass	Bypass scan	Bypass	Normal

† Bit 7 is used to maintain even parity in the 8-bit instruction.

External Test

This instruction conforms to the IEEE Standard 1149.1-1990 extest instruction. The boundary-scan register is selected in the scan path. Data appearing at the device input pins is captured in the input boundary-scan cells (BSCs). Data that has been scanned into the output BSCs is applied to the device output pins. The device operates in the test mode.

Sample Boundary

This instruction conforms to the IEEE Standard 1149.1-1990 sample/preload instruction. The boundary-scan register is selected in the scan path. Data appearing at the device input pins is captured in the input BSCs, while data appearing at the outputs of the normal on-chip logic is captured in the output BSCs. The device operates in the normal mode.

Internal Test

This instruction conforms to the IEEE Standard 1149.1-1990 intest instruction. The boundary-scan register is selected in the scan path. Data appearing at the outputs of the normal on-chip logic is captured in the output BSCs. Data that has been scanned into the input BSCs is applied to the inputs of the normal on-chip logic. The device operates in a modified test mode in which all outputs are placed in the high-impedance state.

Boundary Read

The boundary-scan register is selected in the scan path. The value in the boundary-scan register remains unchanged during capture-DR.

Boundary Self Test

The boundary-scan register is selected in the scan path. All BSCs capture the inverse of their current values during capture-DR. In this way, the contents of the shadow latches may be read out to verify the integrity of both shift register and shadow latch elements of the boundary-scan register. The device operates in the normal mode.

Bypass Scan

This instruction conforms to the IEEE Standard 1149.1-1990 bypass instruction. The bypass register is selected in the scan path. A logic 0 value is captured in the bypass register during capture-DR. The device operates in the normal mode.

TIMING DESCRIPTION

All test operations of the devices are synchronous to the test clock (TCK). Data on the TDI, TMS, and normal-function inputs is captured on the rising edge of TCK. Data appears on the TDO and normal-function output pins on the falling edge of TCK. The TAP controller is advanced through its states (as illustrated in Figure C–1) by changing the value of TMS on the falling edge of TCK and then applying a rising edge to TCK.

A simple timing example is illustrated in Figure C–5. In this example, the TAP controller begins in the test-logic-reset state and is advanced through its states as necessary to perform one instruction register scan and one data register scan. While in the shift-IR and shift-DR states, TDI is used to input serial data, and TDO is used to output serial data. The TAP controller is then returned to the test-logic-reset state. Table C–5 explains the operation of the test circuitry during each TCK cycle.

Table C–5 Explanation of Timing Example

TCK CYCLE(S)	TAP STATE AFTER TCK	DESCRIPTION
1	Test-Logic-Reset	TMS is changed to a logic 0 value on the falling edge of TCK to begin advancing the TAP controller toward the desired state.
2	Run-Test/Idle	
3	Select-DR-Scan	
4	Select-IR-Scan	
5	Capture-IR	The IR captures the 8-bit binary value 10000001 on the rising edge of TCK as the TAP controller exits the capture-IR state.
6	Shift-IR	TDO becomes active, and TDI is made valid on the falling edge of TCK. The first bit is shifted into the TAP on the rising edge of TCK as the TAP controller advances to the next state.
7–13	Shift-IR	One bit is shifted into the IR on each TCK rising edge. With TDI held at a logic 1 value, the 8-bit binary value 11111111 is serially scanned into the IR. At the same time, the 8-bit binary value 10000001 is serially scanned out of the IR via TDO. In TCK cycle 13, TMS is changed to a logic 1 value to end the instruction register scan on the next TCK cycle. The last bit of the instruction is shifted as the TAP controller advances from shift-IR to exit1-IR.
14	Exit1-IR	TDO becomes inactive (goes to the high-impedance state) on the falling edge of TCK.
15	Update-IR	The IR is updated with the new instruction (bypass) on the falling edge of TCK.
16	Select-DR-Scan	
17	Capture-DR	The bypass register captures a logic 0 value on the rising edge of TCK as the TAP controller exits the capture-DR state.
18	Shift-DR	TDO becomes active, and TDI is made valid on the falling edge of TCK. The first bit is shifted into the TAP on the rising edge of TCK as the TAP controller advances to the next state.
19–20	Shift-DR	The binary value 101 is shifted in via TDI, while the binary value 010 is shifted out via TDO.
21	Exit1-DR	TDO becomes inactive (goes to the high-impedance state) on the falling edge of TCK.
22	Update-DR	In general, the selected-data register is updated with the new data on the falling edge of TCK.
23	Select-DR-Scan	
24	Select-IR-Scan	
25	Test-Logic-Reset	Test operation completed

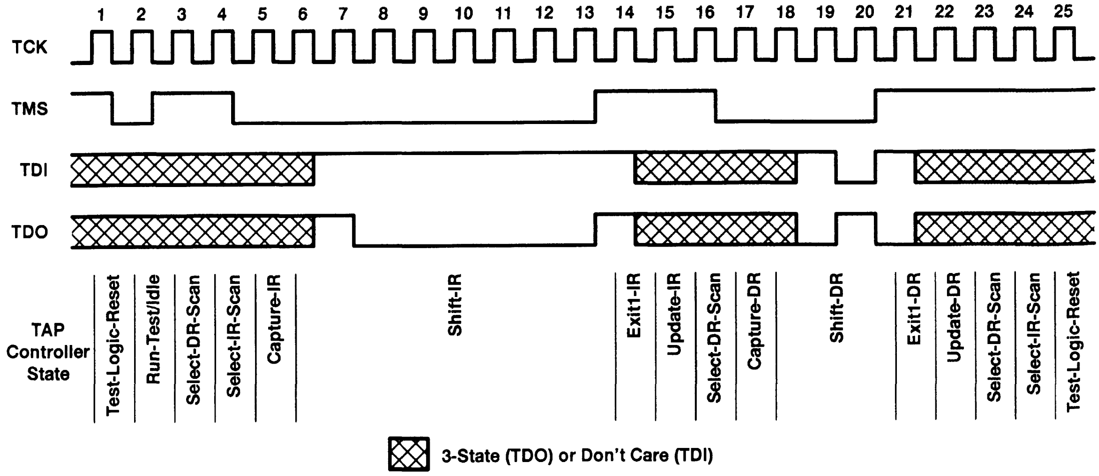


Figure C–5. Timing Example

Appendix D

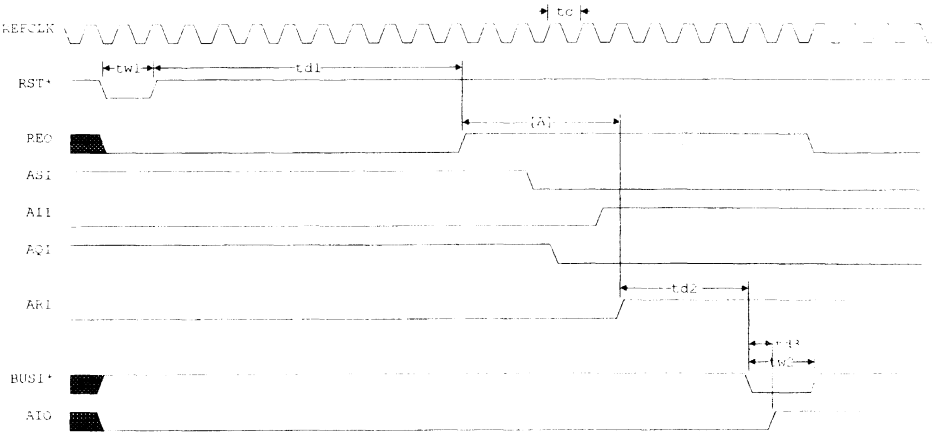
Transaction Diagrams

Appendix D
Transaction Diagrams

Reset, CSR, Host, and FB+ Transactions

- Numeric characters 0 and1 will be used to represent signal levels on but outputs and bidirects in output mode and alphabetic characters L and H will be used to represent signal levels on bus inputs and bidirects in input mode.
- SIGI<2> or SIGO<3> label is equivalent to SIG2I or SIG3O label, e.g., STO<1> is equivalent to ST1O.
- SIGI<2> indicates the input signal of SIG bit 2, and SIGO<2> indicates the output signal of SIG bit 2.
- The diagrams are not on scale.
- [A], [B] are note numbers.

Index to Reset Timing Diagrams



NOTE A: Time from REO to FB+ idle state depends on individual design. FB+ idle state is defined as ASI and AQI released and ARI and AII asserted.

Figure D–1. Live Insertion Without Alignment Wait Feature

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
$t_{c(REFCLK)}$	Clock cycle time	25	25	25	ns
t_{w1}	Pulse duration, RST* low	1			µs

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, RST*↑ to REO↑	C_L = 20 pF	150	150	150	ms
t_{d2}	Delay time, FB+ idle state to BUSI*↓		1.6	1.6	1.6	µs
t_{d3}	Delay time, BUSI*↓ to AIO↑		7	10	14	ns
t_{w2}	Pulse duration, BUSI* low		2 tc	2 tc	2 tc	ns

The second level of the hierarchy is comprised of two schematic sheets. Sheet one contains the controller block and multiplexers used during writes. The blocks CDMUX and CDPMUX route the appropriate bytes of the HIF data bus and parity to the CSR bus data bus parity. Sheet one is shown in Figure B–6 below:

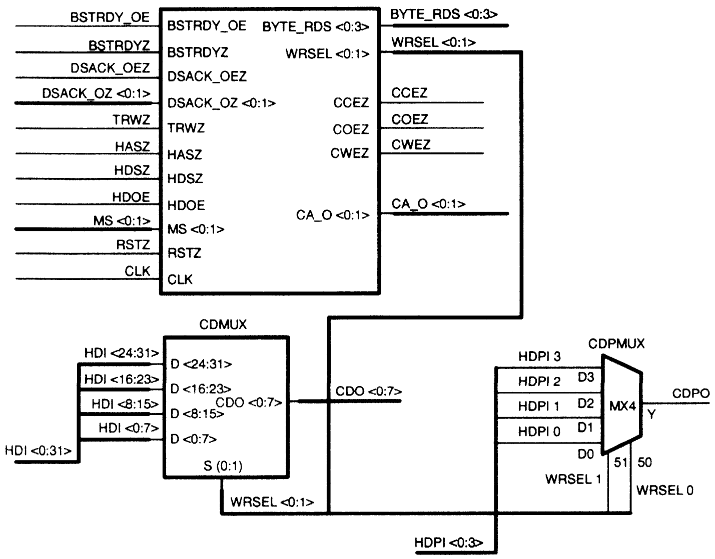


Figure B–6. Sheet One Of Two Of The Core Block

The schematic below shows the implementation for CDMUX which is a byte-wide four-to-one multiplexer.

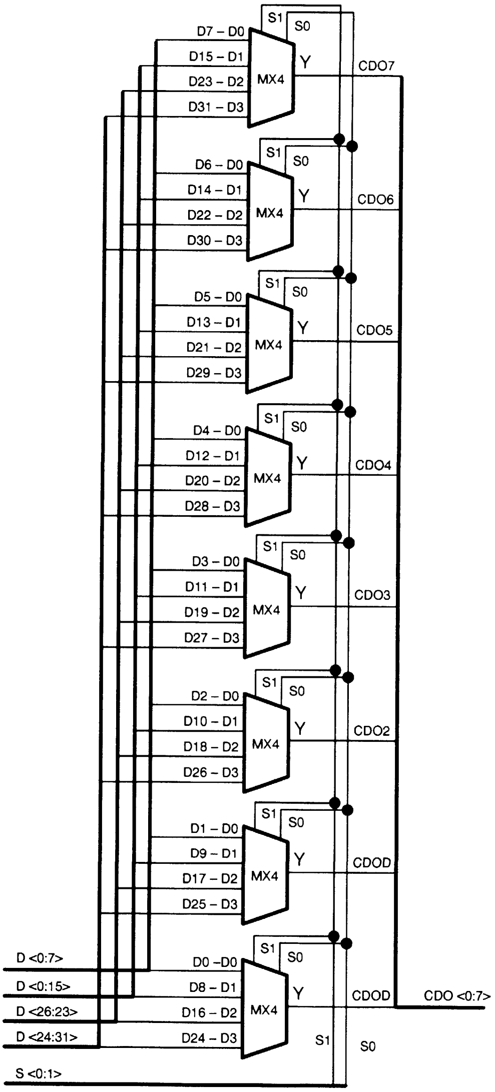


Figure B–7. Sheet One Of One Of The CDMUX Block

Sheet two contains the latches used during reads These latches capture four bytes with byte addresses zero, one, two, and three from the CSR bus data bus and sources them to the HIF data bus as the data for a single CSR read. Sheet two is shown in Figure B–8 below.

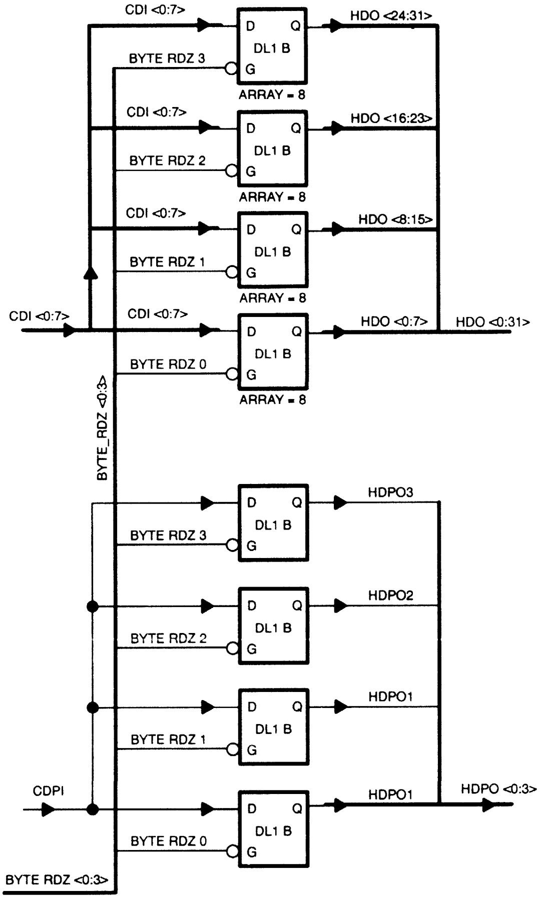


Figure B–8. Sheet Two Of Two Of The Core Block

The third level of the hierarchy contains the HIF transaction start detector, CSR bus byte address counter, write byte select counter, and the CSRIF state machine. The schematic is shown in Figure B–9.

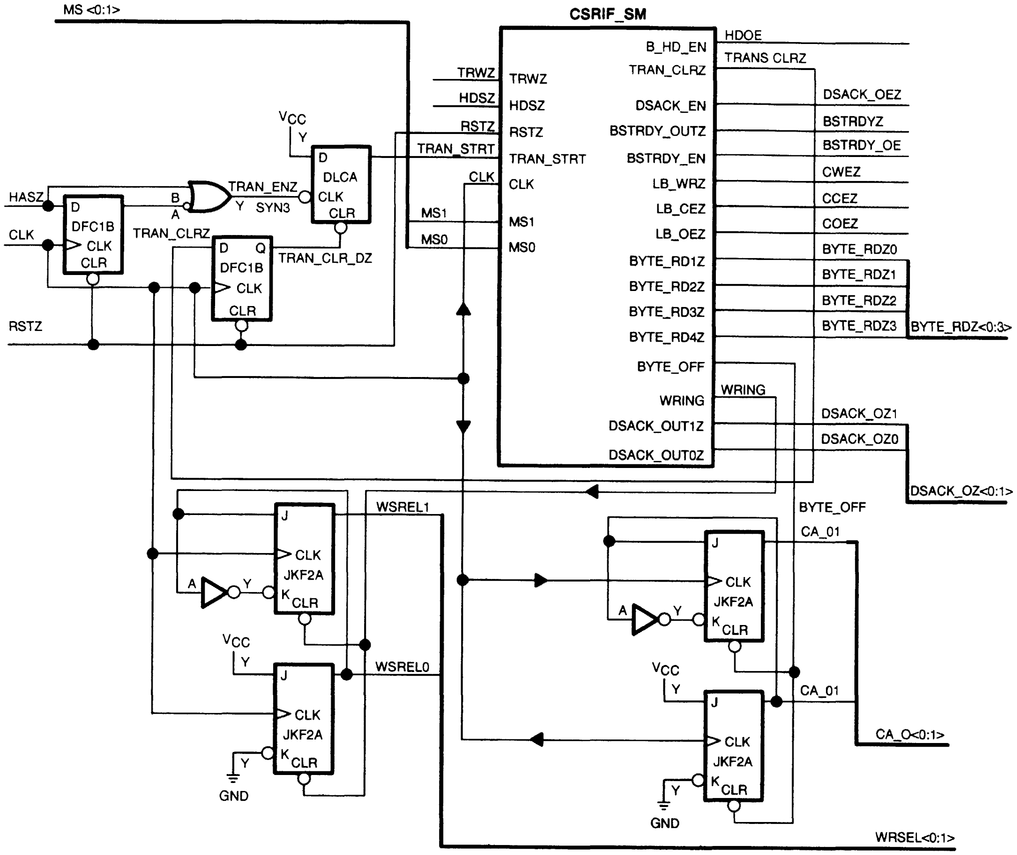


Figure B–9. Sheet One Of One Of The CSRC Block

The schematic for the CSRIF state machine is shown below. This state machine's instance name is LSRIF_SM in Figure B–9.

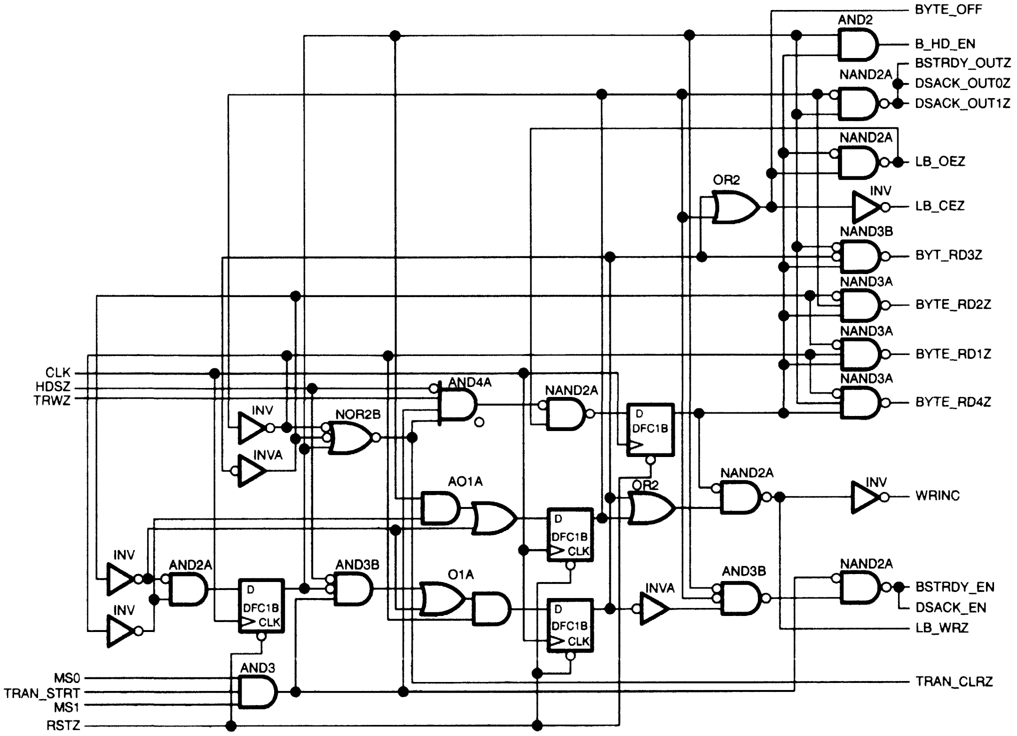


Figure B–10. Sheet One Of One Of The CSRIF_SM Block

This state machine was first described as a register transfer level (RTL) Verilog module. It was synthesized (mapped to gates) using the Synopsys design compiler. The schematic shown in Figure B–10 is machine generated but it can be captured by hand. The Verilog module is shown below.

```
//******************************************************************************
// Module Definition
//******************************************************************************
module CSRIF_SM(

          // Inputs
                               MS1      , // Memory Select
                               MS0      , // Memory Select
                               TRWZ    , // HBus Read Write
                               HDSZ    , // HBus Data Strobe
                               RSTZ    , // HBus Reset
                               CLK      , // HBus Clock
                               TRAN_STRT    ,

          // Outputs
                               B_HD_EN    ,
                               TRAN_CLRZ    ,
                               DSACK_OUT1Z    ,
                               DSACK_OUT0Z    ,
                               DSACK_EN    ,
                               BSTRDY_OUTZ    ,
                               BSTRDY_EN    ,
                               LB_WRZ       ,
                               LB_CEZ       ,
                               LB_OEZ       ,
                               BYTE_RD1Z    ,
                               BYTE_RD2Z    ,
                               BYTE_RD3Z    ,
                               BYTE_RD4Z    ,
                               BYTE_OFF     ,
                               WRINC        ,
                   );

input
  MS1 ,
  MS0 ,
  TRWZ ,
  HDSZ ,
  RSTZ ,
  CLK ,
  TRAN_STRT ,

output
  WRINC ,
  BYTE_OFF ,
  B_HD_EN ,
  TRAN_CLRZ ,
  DSACK_EN ,
  BSTRDY_EN ,
  BSTRDY_OUTZ ,
  LB_WRZ ,
  LB_CEZ ,
  LB_OEZ ,
  DSACK_OUT1Z ,
  DSACK_OUT0Z ,
  BYTE_RD1Z ,
  BYTE_RD2Z ,
  BYTE_RD3Z ,
  BYTE_RD4Z ;
reg
  DSACK_OUT1Z ,
  DSACK_OUT0Z ,
  WRINC ,
  BYTE_OFF ,
  TRAN_CLRZ ,
  LB_WRZ , LB_CEZ , LB_OEZ ,
  B_HD_EN ,
  BYTE_RD1Z , BYTE_RD2Z , BYTE_RD3Z, BYTE_RD4Z ,
  BSTRDY_OUTZ , DSACK_EN , BSRTDY_EN ;
reg [3:0]
  nx_st ;
reg [3:0]
  st_reg ;
DFC1B u4 (.D(nx_st[3]) , .Q(st_reg[3]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u3 (.D(nx_st[2]) , .Q(st_reg[2]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u2 (.D(nx_st[1]) , .Q(st_reg[1]) , .CLR(RSTZ) , .CLK(CLK)) ;
DFC1B u1 (.D(nx_st[0]) , .Q(st_reg[0]) , .CLR(RSTZ) , .CLK(CLK)) ;
  always
     begin
        case (st_reg) // synopsys parallel_case full_case
          4'b0000:
             begin : idle
                BYTE_OFF = 1'b0 ;
                WRINC = 1'b0 ;
                LB_WRZ = 1'b1 ;
                LB_CEZ = 1'b1 ;
                LB_OEZ = 1'b1 ;

                DSACK_OUT1Z = 1'b1 ;
                DSACK_OUT0Z = 1'b1 ;
                BSTRDY_OUTZ = 1'b1 ;
                DSACK_EN = 1'b0 ;
                BSTRDY_EN = 1'b0 ;

                 BYTE_RD1Z = 1'b1 ;
                 BYTE_RD2Z = 1'b1 ;
                 BYTE_RD3Z = 1'b1 ;
                 BYTE_RD4Z = 1'b1 ;

                 TRAN_CLRZ = 1'b1 ;
        B_HD_EN = 1'b1 ;

        if ((TRAN_STRT == 1'b1) && (MS1 == 1'b1) && (MS0 == 1'b1))
           begin
              DSACK_EN = 1'b1 ;
              BSTRDY_EN = 1'b1 ;
              if ((TRW̄Z == 1'b0) && (HDSZ == 1'b0))
                nx_st = 4'b0001 ;  //csr write
              else if ((TRWZ == 1'b1) && (HDSZ == 1'b0))
                nx_st = 4'b1001 ;  //csr read
              else
                nx_st = 4'b0000 ;
           end
        else
           nx_st = 4'b0000 ;  //stay in idle state
    end

// ---------- csr_write ----------
4'b0001 :
  begin : csr_write1
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b1 ;
    LB_WRZ = 1'b0 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0011 ;  //csr_write2
  end

4'b0011 :
  begin : csr_write2
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b1 ;
    LB_WRZ = 1'b0 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b1 ;
```

```
          DSACK_OUT1Z = 1'b1 ;
          DSACK_OUT0Z = 1'b1 ;
          BSTRDY_OUTZ = 1'b1 ;
          DSACK_EN = 1'b1 ;
          BSTRDY_EN = 1'b1 ;

          BYTE_RD1Z = 1'b1 ;
          BYTE_RD2Z = 1'b1 ;
          BYTE_RD3Z = 1'b1 ;
          BYTE_RD4Z = 1'b1 ;

          TRAN_CLRZ = 1'b0 ;
          B_HD_EN = 1'b0 ;

          nx_st = 4'b0010 ;  //csr_write3
        end

      4'b0010 :
        begin : csr_write3
          BYTE_OFF = 1'b1 ;
          WRINC = 1'b1 ;
          LB_WRZ = 1'b0 ;
          LB_CEZ = 1'b0 ;
          LB_OEZ = 1'b1 ;

          DSACK_OUT1Z = 1'b1 ;
          DSACK_OUT0Z = 1'b1 ;
          BSTRDY_OUTZ = 1'b1 ;
          DSACK_EN = 1'b1 ;
          BSTRDY_EN = 1'b1 ;

          BYTE_RD1Z = 1'b1 ;
          BYTE_RD2Z = 1'b1 ;
          BYTE_RD3Z = 1'b1 ;
          BYTE_RD4Z = 1'b1 ;

          TRAN_CLRZ = 1'b0 ;
          B_HD_EN = 1'b0 ;

          nx_st = 4'b0110 ;  //csr_write4
        end


      4'b0110 :
        begin : csr_write4
          BYTE_OFF = 1'b1 ;
          WRINC = 1'b1 ;
          LB_WRZ = 1'b0 ;
          LB_CEZ = 1'b0 ;
```

```
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0100 ;
  end

4'b0100 :
  begin : csr_write_dsack
    BYTE_OFF = 2'b0 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b1 ;
    LB_OEZ = 1'b1 ;

    DSACK_OUT1Z = 1'b0 ;
    DSACK_OUT0Z = 1'b0 ;
    BSTRDY_OUTZ = 1'b0 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b0000 ;   // go to idle state
  end

// ---------- csr_read ----------
4'b1001 :
  begin : csr_read1
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b0 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b0 ;
    BYTE_RD2Z = 1'b1 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b1011 ;  //csr_read2
  end

4'b1011 :
  begin : csr_read2
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b1 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1'b0 ;
    LB_OEZ = 1'b0 ;

    DSACK_OUT1Z = 1'b1 ;
    DSACK_OUT0Z = 1'b1 ;
    BSTRDY_OUTZ = 1'b1 ;
    DSACK_EN = 1'b1 ;
    BSTRDY_EN = 1'b1 ;

    BYTE_RD1Z = 1'b1 ;
    BYTE_RD2Z = 1'b0 ;
    BYTE_RD3Z = 1'b1 ;
    BYTE_RD4Z = 1'b1 ;

    TRAN_CLRZ = 1'b0 ;
    B_HD_EN = 1'b0 ;

    nx_st = 4'b1010 ;  //csr_read3
  end

4'b1010 :
  begin : csr_read3
    BYTE_OFF = 1'b1 ;
    WRINC = 1'b0 ;
    LB_WRZ = 1'b1 ;
    LB_CEZ = 1’b0 ;
    LB_OEZ = 1’b0 ;

    DSACK_OUT1Z = 1’b1 ;
    DSACK_OUT0Z = 1’b1 ;
    BSTRDY_OUTZ = 1’b1 ;
    DSACK_EN = 1’b1 ;
    BSTRDY_EN = 1’b1 ;

    BYTE_RD1Z = 1’b1 ;
    BYTE_RD2Z = 1’b1 ;
    BYTE_RD3Z = 1’b0 ;
    BYTE_RD4Z = 1’b1 ;

    TRAN_CLRZ = 1’b0 ;
    B_HD_EN = 1’b0 ;

    nx_st = 4’b1110 ;  //csr_read4
  end

4’b1110 :
  begin : csr_read4
    BYTE_OFF = 1’b1 ;
    WRINC = 1’b0 ;
    LB_WRZ = 1’b1 ;
    LB_CEZ = 1’b0 ;
    LB_OEZ = 1’b0 ;

    DSACK_OUT1Z = 1’b1 ;
    DSACK_OUT0Z = 1’b1 ;
    BSTRDY_OUTZ = 1’b1 ;
    DSACK_EN = 1’b1 ;
    BSTRDY_EN = 1’b1 ;

    BYTE_RD1Z = 1’b1 ;
    BYTE_RD2Z = 1’b1 ;
    BYTE_RD3Z = 1’b1 ;
    BYTE_RD4Z = 1’b0 ;

    TRAN_CLRZ = 1’b0 ;
    B_HD_EN = 1’b1 ;

    nx_st = 4’b1100 ;
  end

4’b1100 :
  begin : csr_read_dsack
    BYTE_OFF = 1’b0 ;
    WRINC = 1’b0 ;
    LB_WRZ = 1’b1 ;
```

```
              LB_CEZ = 1'b1 ;
              LB_OEZ = 1'b1 ;

              DSACK_OUT1Z = 1'b0 ;
              DSACK_OUT0Z = 1'b0 ;
              BSTRDY_OUTZ = 1'b0 ;
              DSACK_EN = 1'b1 ;
              BSTRDY_EN = 1'b1 ;

              BYTE_RD1Z = 1'b1 ;
              BYTE_RD2Z = 1'b1 ;
              BYTE_RD3Z = 1'b1 ;
              BYTE_RD4Z = 1'b1 ;

              TRAN_CLRZ = 1'b0 ;
              B_HD_EN = 1'b1 ;

              nx_st = 4'b0000 ;  //go to idle state
            end

         endcase
      #1;
      end  // end of sm

endmodule
```

Figure B–11. Verilog Module

This is a minimal implementation of the CSR bus interface. There are several enhancements that could be added by the board designer:

- This design expects the CSR bus word address (CA<11:2>) to be directly connected to the host interface CSR word address (HA<11:2>). The CSRIF generates only the CSR bus byte address (CA<1:0>). If the board design places enough loading on the CSR bus to degrade the performance of the HIF, CA<11:2> can be isolated from HA<11:2> with a latch that is transparent when HIP* is not asserted. This can be implemented discretely with a 10-bit latch or on chip by selecting an FPGA that would accommodate the extra 21 pins required.
- If the vendor ROM is a 1 k × 8 device connected to the CSR bus, a chip select for the ROM can be generated by adding HA<11:10> as inputs and using them along with MS<1:0> as inputs to a decoder. The output of the decoder would function as the chip enable for the vendor ROM.
- Vendor defined initial units (CSR registers) could be implemented in the CSRIF FPGA. These registers can be embedded in the chip by bringing the HIF CSR word address (<HA11:2>) on chip for use as a register address decode.
- The CSRIF could provide service to extended units on the CSR bus by decoding MS = HL (local extended units along with MS = HH (local CSR)). This would echo transactions with addresses that fall within unit base and units bound onto the CSR bus. The HIF address would have to be decoded to the proper granularity to support the memory map of the existing extended units.

Appendix C

TI Futurebus+ Chip Set IEEE 1149.1 Description

Appendix C
TFB20002B, TFB2010, TFB2022A, JTAG Description

INTRODUCTION

The TFB2002B, TFB2010, and TFB2022A Futurebus+ devices are members of the Texas Instruments SCOPE™ testability IC family. This family of devices supports IEEE Standard 1149.1-1990 boundary scan to facilitate testing of complex circuit-board assemblies. Scan access to the test circuitry is accomplished via the 4-wire test access-port (TAP) interface.

In the normal mode, these devices operate as Futurebus+ functions. The test circuitry can be activated by the TAP to take snapshot samples of the data appearing at the device pins or to perform a self test on the boundary test cells. Activating the TAP in the normal mode does not affect the functional operation of the Futurebus+ devices.

In the test mode, the normal operation of the Futurebus+ devices is inhibited and the test circuitry is enabled to observe and control the I/O boundary of the device. When enabled, the test circuitry can perform boundary scan test operations as described in IEEE Standard 1149.1-1990.

Four dedicated test pins are used to control the operation of the test circuitry. The are test data input (TDI), test data output (TDO), test-mode select (TMS), and test clock (TCK). All testing and scan operations are synchronized to the TAP interface.

Overview

A functional block diagram of the IEEE 1149.1 test architecture standard is shown in Figure C–1. The implementation includes an 8-bit instruction register, a one-bit bypass register, and a boundary-scan register.

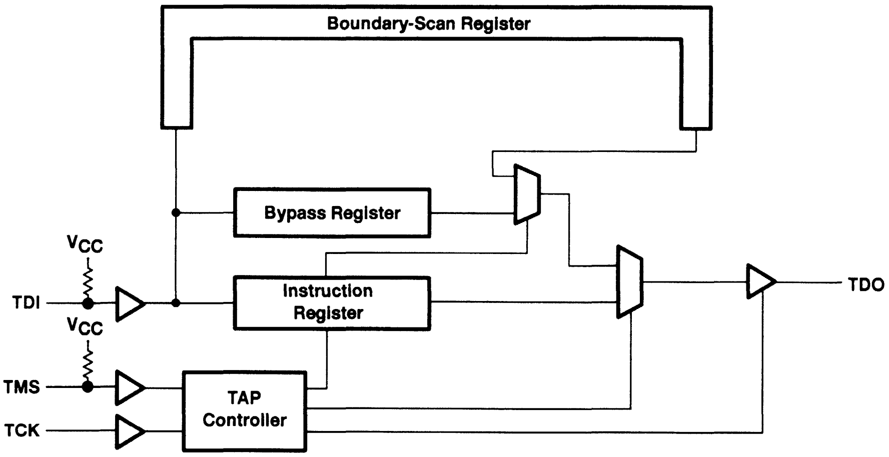


Figure C–1. Functional Block Diagram

SCOPE is a trademark of Texas Instruments Incorporated.

Terminal Functions

TERMINAL NAME	I/O	DESCRIPTION
GND		Ground
TCK	I	Test clock input. One of four terminals required by IEEE Standard 1149.1-1990. Test operations of the device are synchronous to the test clock. Data is captured on the rising edge of TCK, and outputs change on the falling edge of TCK.
TDI	I	Test data input. One of four terminals required by IEEE Standard 1149.1-1990. The test data input is the serial input for shifting data through the instruction register or selected data register. An internal pullup forces TDI to a high level if left unconnected.
TDO	O	Test data output. One of four terminals required by IEEE Standard 1149.1-1990. The test data output is the serial output for shifting data through the instruction register or selected data register.
TMS	I	Test-mode select input. One of four terminals required by IEEE Standard 1149.1-1990. The test-mode select input directs the device through its test access port (TAP) controller states. An internal pullup forces TMS to a high level if left unconnected.
V_{CC}		Supply voltage

Test Architecture

Serial test information is conveyed by means of a 4-wire test bus, or test-access port (TAP), that conforms to IEEE Standard 1149.1-1990. Test instructions, test data, and test control signals are all passed along this serial test bus. The TAP controller monitors two signals from the test bus, namely TCK and TMS. The function of the TAP controller is to extract the synchronization (TCK) and state control (TMS) signals from the test bus and generate the appropriate on-chip control signals for the test structures in the device. Figure C–2 shows the TAP controller state diagram.

The TAP controller is fully synchronous to the TCK signal. Input data is captured on the rising edge of TCK, and output data changes on the falling edge of TCK. This scheme ensures that data to be captured is valid for fully one-half of the TCK cycle.

The functional block diagram in Figure C–1 illustrates the IEEE Standard 1149.1-1990 4-wire test bus and boundary-scan architecture and the relationship between the test bus, the TAP controller, and the test registers. As illustrated, the device contains an 8-bit instruction register and two test-data registers: a boundary-scan register and a bypass register.

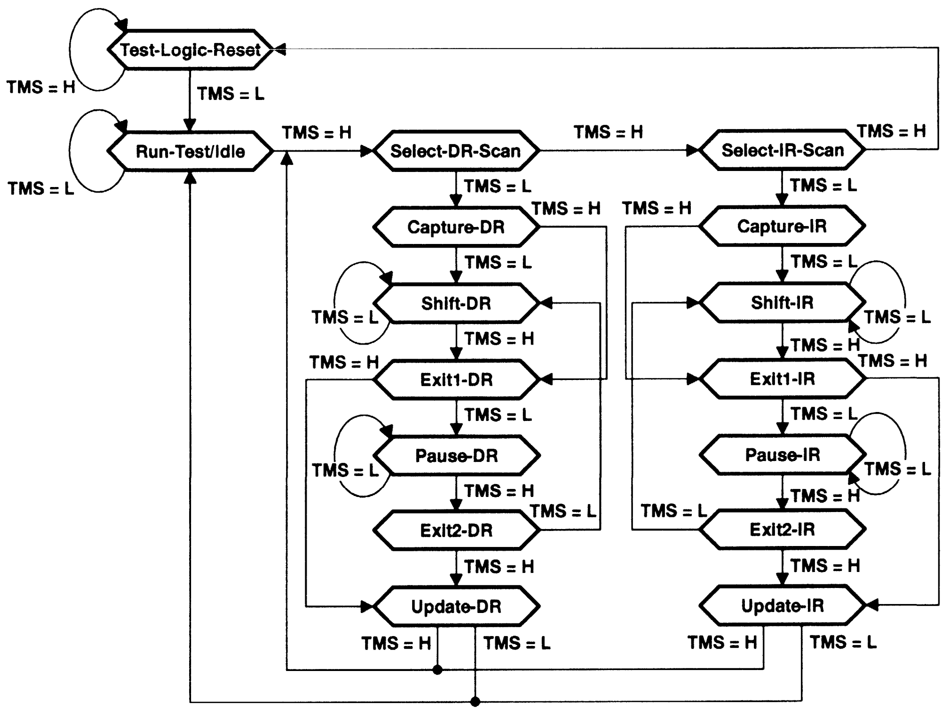


Figure C–2. TAP Controller State Diagram

STATE DIAGRAM DESCRIPTION

The test-access port (TAP) controller specified in IEEE Standard 1149.1-1990 is a synchronous finite state machine that provides test control signals throughout the device. The TAP controller proceeds through its states based on the level of TMS at the rising edge of TCK.

As illustrated, the TAP controller consists of sixteen states. There are six stable states (indicated by a looping arrow in the state diagram) and ten unstable states. A stable state is defined as a state the TAP controller can retain for consecutive TCK cycles. Any state that does not meet this criterion is an unstable state.

There are two main paths through the state diagram: one to access and control the selected data register, and one to access and control the instruction register. Only one register may be accessed at a time.

Test-Logic-Reset

The device powers up in the test-logic-reset state. In the stable test-logic-reset state, the test logic is reset and is disabled so that the normal logic function of the device is performed. The instruction register is reset to an opcode that selects the bypass instruction.

The state machine is constructed such that the TAP controller returns to the test-logic-reset state in no more than five TCK cycles if TMS is left high. The TMS pin has an internal pullup resistor that forces it high if left unconnected or if a board defect causes it to be open circuited.

Run-Test/Idle

The TAP controller must pass through the run-test/idle state (from test-logic-reset) before executing any test operations. The run-test/idle state may also be entered following data register or instruction register scans. Run-test/idle is provided as a stable state in which the test logic may be actively running a test or may be idle.

Select-DR-Scan, Select-IR-Scan

No specific function is performed in the select-DR-scan and select-IR-scan states, and the TAP controller exits either of these states on the next TCK cycle. These states are provided to allow the selection of either data register scan or instruction register scan.

Capture-DR

When a data register scan is selected, the TAP controller must pass through the capture-DR state. In the capture-DR state, the selected-data register may capture a data value as specified by the current instruction. Such capture operations occur on the rising edge of TCK upon which the TAP controller exits the capture-DR state.

Shift-DR

Upon entry to the shift-DR state, the selected data register is placed in the scan path between TDI and TDO. On the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO enables to the logic level present in the least significant bit of the selected data register.

While in the stable shift-DR state, data is serially shifted through the selected data register on each TCK cycle. The first shift occurs on the first rising edge of TCK after entry to the shift-DR state (i.e., no shifting occurs during the TCK cycle in which the TAP controller changes from capture-DR to shift-DR or from exit2-DR to shift-DR). The last shift occurs on the rising edge of TCK upon which the TAP controller exits the shift-DR state.

Exit1-DR, Exit2-DR

The exit1-DR and exit2-DR states are temporary states used to end a data register scan. It is possible to return to the shift-DR state from either exit1-DR or exit2-DR without recapturing the data register. On the first falling edge of TCK after entry to exit1-DR, TDO goes from the active state to the high-impedance state.

Pause-DR

No specific function is performed in the stable pause-DR state, in which the TAP controller can remain indefinitely. The pause-DR state provides the capability of suspending and resuming data-register scan operations without loss of data.

Update-DR

If the current instruction calls for the selected data register to be updated with current data, then updates occur on the falling edge of TCK following entry to the update-DR state.

Capture-IR

When an instruction register scan is selected, the TAP controller must pass through the capture-IR state. In the capture-IR state, the instruction register captures its current status value. This capture operation occurs on the rising edge of TCK, and the TAP controller exits the capture-IR state. For these devices, the status value loaded in the capture-IR state is the fixed binary value 10000001.

Shift-IR

Upon entry to the shift-IR state, the instruction register is placed in the scan path between TDI and TDO, and on the first falling edge of TCK, TDO goes from the high-impedance state to an active state. TDO enables to the logic level present in the least significant bit of the instruction register.

While in the stable shift-IR state, instruction data is serially shifted through the instruction register on each TCK cycle. The first shift occurs on the first rising edge of TCK after entry to the shift-IR state (i.e., no shifting occurs during the TCK cycle in which the TAP controller changes from capture-IR to shift-IR or from exit2-IR to shift-IR). The last shift occurs on the rising edge of TCK upon which the TAP controller exits the shift-IR state.

Exit1-IR, Exit2-IR

The exit1-IR and exit2-IR states are temporary states used to end an instruction register scan. It is possible to return to the shift-IR state from either exit1-IR or exit2-IR without recapturing the instruction register. On the first falling edge of TCK after entry to exit1-IR, TDO goes from the active state to the high-impedance state.

Pause-IR

No specific function is performed in the stable pause-IR state, in which the TAP controller can remain indefinitely. The pause-IR state provides the capability of suspending and resuming instruction register scan operations without loss of data.

Update-IR

The current instruction is updated and takes effect on the falling edge of TCK following entry to the update-IR state.

REGISTER OVERVIEW

With the exception of the bypass register, any test register may be thought of as a serial shift register with a shadow latch on each bit. The bypass register differs in that it contains only a shift register. During the appropriate capture state (capture-IR for instruction register, capture-DR for data registers), the shift register may be parallel loaded from a source specified by the current instruction. During the appropriate shift state (shift-IR or shift-DR), the contents of the shift register are shifted out from TDO while new contents are shifted in at TDI. During the appropriate update state (update-IR or update-DR), the shadow latches are updated from the shift register.

INSTRUCTION-REGISTER DESCRIPTION

The instruction register (IR) is eight bits long and is used to tell the device what instruction is to be executed. Information contained in the instruction includes the mode of operation (either normal mode in which the device performs its normal logic function, or test mode in which the normal logic function is inhibited or altered), the test operation to be performed, which of the two data registers is to be selected for inclusion in the scan path during data-register scans, and the source of data to be captured into the selected data register during capture-DR.

Table C–4 lists the instructions supported by these devices. The even-parity feature specified for SCOPE devices are supported in these devices. Bit 7 of the instruction opcode is the parity bit. Any instructions that are defined for SCOPE devices but are not supported by these devices default to bypass.

During capture-IR, the IR captures the binary value 10000001. As an instruction is shifted in, this value is shifted out via TDO and can be inspected as verification that the IR is in the scan path. During update-IR, the value that has been shifted into the IR is loaded into shadow latches. At this time, the current instruction is updated, and any specified mode change takes effect. At power up or in the test-logic-reset state, the IR is reset to the binary value 10000001.

The instruction register order of scan is illustrated in Figure C–3.

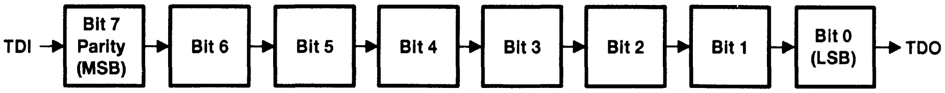


Figure C–3. Instruction Register Order of Scan

DATA-REGISTER DESCRIPTION

Boundary-Scan Register

The boundary-scan register (BSR) is used 1) to store test data that is to be applied internally to the inputs of the normal on-chip logic and/or externally to the device output pins, and/or 2) to capture data that appears internally at the outputs of the normal on-chip logic and/or externally at the device input pins.

The boundary-scan register configurations for the TFB2002B, TFB2022A, and TFB2010 are provided in Tables C–1, C–2, and C–3, respectively. The boundary-scan registers are not preset or cleared at power up or in test-logic-reset. The boundary-scan register order of scan is from TDI to TDO, where boundary-scan cell zero (0) is closest to TDO. The source of data to be captured into the BSR during capture-DR is determined by the current instruction.

Table C–1. TFB2002B Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
146	CLOCK	109	DATAAV*	072	ST0O	035	GR
145	IGNORE*	108	SPACEAV*	071	ST0I	034	RQ0
144	INT*	107	NEWADDR*	070	CM7	033	RQ1
143	HBMASTER*	106	TRW*	069	CM6	032	REFCL*
142	HBGACK*	105	DL1	068	CM5	031	RSTBYPASS*
141	HBG*	104	DL0	067	CM4	030	ARBERR0
140	HBR*	103	DSACK1*	066	CM3	029	ARBERR1
139	LKFLD0	102	§DSOE*	065	CM2	028	BUSI*
138	LKFLD1	101	DSACK0*	064	CM1	027	SYSRST*
137	LKFLD2	100	FADEC3	063	CM0	026	BINIT*
136	HDS*	099	FADEC2	062	CMP	025	RST*
135	DW64*	098	FADEC1	061	¶CMWR*	024	CWE*
134	†HBOE*	097	FADEC0	060	CA2O	023	CCE*
133	TBST*	096	FMODE2	059	CA2I	022	COE*
132	MORE*	095	FMODE1	058	CA1O	021	CD0
131	LK*	094	FMODE0	057	CA1I	020	CDI
130	ERROR0	093	FRD*	056	CA0O	019	CD2
129	ERROR1	092	FSTRB*	055	CA0I	018	CD3
128	FIFORST*	091	FACK*	054	ETO	017	#CDOE*
127	HBADLD*	090	SELECTED*	053	ETI	016	CD4
126	UNALIGNED*	089	BSTRDY*	052	DKO	015	CD5
125	BSTAT1*	088	ADRCV	051	DKI	014	CD6
124	‡BSOE*	087	ADDRV*	050	DIO	013	CD7
123	BSTAT0*	086	ST7O	049	DII	012	CDP
122	TSIZE1	085	ST7I	048	ASO	011	CA0
121	TSIZE0	084	ST6O	047	ASI	010	CA1
120	HSTRB*	083	ST6I	046	AKO	009	CA2
119	HMODE2	082	ST5O	045	AKI	008	CA3
118	HMODE1	081	ST5I	044	DSO	007	CA4
117	HMODE0	080	ST4O	043	DSI	006	CA5
116	HADEC3	079	ST4I	042	AIO	005	CA6
115	HADEC2	078	ST3O	041	AII	004	CA7
114	HADEC1	077	ST3I	040	AQI	003	CA8
113	HADEC0	076	ST20	039	ARI	002	CA9
112	HAS*	075	ST2I	038	REO	001	CA10
111	HIP*	074	ST1O	037	REI	000	CA11
110	DMAMODE	073	ST1I	036	PEI	—	—

† Active-low output enable for HAS*, HIP*, HDS*, LKFLD<2:0>, LK*, DW64*, TBST*, DL<1:0>,TRW*, and HBGACK*
‡ Active-low output enable for BSTATI* and BSTAT0*
§ Active-low output enable for DSACK1*, DSACK0*, and BSTRDY*
¶ Active-low output enable for CM<7:0> and CMP
Active-low output enable for CD<7:0> and CDP

Table C–2. TFB2022A Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
211	CLOCK	174	AD47	137	AD15	100	DATAAV*
210	MS1	173	AD46	136	AD14	099	DMAMODE
209	MS0	172	AD45	135	AD13	098	HIP*
208	CM5	171	AD44	134	AD12	097	HAS*
207	CM6	170	AD43	133	AD11	096	HADEC0
206	CM7	169	AD42	132	AD10	095	HADEC1
205	ST3	168	AD41	131	AD9	094	HADEC2
204	ST5	167	AD40	130	AD8	093	HADEC3
203	GA0*	166	ADP4	129	ADP0	092	HMODE0
202	DS	165	AD39	128	AD7	091	HMODE1
201	GA1*	164	AD38	127	AD6	090	HMODE2
200	AS	163	AD37	126	AD5	089	HSTRB*
199	GA2*	162	AD36	125	AD4	088	TSIZE1
198	DI	161	AD35	124	AD3	087	TSIZE0
197	GA3*	160	AD34	123	AD2	086	HA0
196	DK	159	AD33	122	AD1	085	HA1
195	GA4*	158	AD32	121	AD0	084	HA2
194	ADP7	157	ADP3	120	BSTRDY*	083	HA3
193	AD63	156	AD31	119	SELECTED*	082	HA4
192	AD62	155	AD30	118	FACK*	081	HA5
191	AD61	154	AD29	117	FSTRB*	080	HA6
190	AD60	153	AD28	116	FRD*	079	HA7
189	AD59	152	AD27	115	FMODE0	078	HA8
188	AD58	151	AD26	114	FMODE1	077	HA9
187	AD57	150	AD25	113	FMODE2	076	HA10
186	AD56	149	AD24	112	FADEC0	075	HA11
185	ADP6	148	ADP2	111	FADEC1	074	HA12
184	AD55	147	AD23	110	FADEC2	073	HA13
183	AD54	146	AD22	109	FADEC3	072	HA14
182	AD53	145	AD21	108	DSACK0*	071	HA15
181	AD52	144	AD20	107	DSACK1*	070	HA16
180	AD51	143	AD19	106	DL0	069	HA17
179	AD50	142	AD18	105	DL1	068	§HAOE*
178	AD49	141	AD17	104	TRW*	067	HA18
177	AD48	140	AD16	103	NEWADDR*	066	HA19
176	†ADHOE*	139	‡ADLOE*	102	REFCLK	065	HA20
175	ADP5	138	ADP1	101	SPACEAV*	064	HA21

† Active-low output control for AD<63:32> and ADP <7:4>

‡ Active-low output control for AD<31:0> and ADP <3:0>

§ Active-low output control for HA<31:0> and HAP <3:0>

Table C–2. TFB2022A Boundary-Register Configuration (Continued)

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
063	HA22	047	HD2	031	†HDOE*	015	HD30
062	HA23	046	HD3	030	HD16	014	HD31
061	HA24	045	HD4	029	HD17	013	HDP3
060	HA25	044	HD5	028	HD18	012	BSTAT0*
059	HA26	043	HD6	027	HD19	011	BSTAT1*
058	HA27	042	HD7	026	HD20	010	TEST_PIN1
057	HA28	041	HDP0	025	HD21	009	TEST_PIN2
056	HA29	040	HD8	024	HD22	008	UNALIGNED*
055	HA30	039	HD9	023	HD23	007	HBADLD*
054	HA31	038	HD10	022	HDP2	006	FIFORST*
053	HAP0	037	HD11	021	HD24	005	ERROR1
052	HAP1	036	HD12	020	HD25	004	ERROR0
051	HAP2	035	HD13	019	HD26	003	SYSRST*
050	HAP3	034	HD14	018	HD27	002	BINIT*
049	HD0	033	HD15	017	HD28	001	RST*
048	HD1	032	HDP1	016	HD29	000	‡HBMASTER*

† Active-low output control for HD<31:0> and HDP <3:0>

‡ Active-low output control for TSIZE1 and TSIZEO

Table C–3. TFB2010 Boundary-Register Configuration

BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL	BSR BIT NUMBER	DEVICE SIGNAL
71	GA0*	53	CD7	35	ARBERR0	17	CMPT*
70	GA1*	52	CD6	34	RQ1	16	OEA
69	GA2*	51	CD5	33	RQ0	15	API
68	GA3*	50	†CDOE*	32	GR	14	APO
67	GA4*	49	CD4	31	PE	13	AQI
66	CA11	48	CD3	30	CENTRAL	12	AQO
65	CA10	47	CD2	29	CN0	11	ARI
64	CA9	46	CD1	28	CN1	10	ARO
63	CA8	45	CD0	27	CN2	09	AC1I
62	CA7	44	COE*	26	CN3	08	AC1O
61	CA6	43	CCE*	25	‡CNOE*	07	AC0I
60	CA5	42	CWE*	24	CN4	06	AC0O
59	CA4	41	REFCLK	23	CN5	05	REI
58	CA3	40	RST*	22	CN6	04	ASI
57	CA2	39	BINIT*	21	CN7	03	REF
56	CA1	38	SYSRST*	20	CNP	02	INT*
55	CA0	37	BUSI*	19	WIN	01	PFAIL*
54	CDP	36	ARBERR1	18	LE*	00	CLOCK

† Active-low output control for CD<7:0>, CDP

‡ Active-low output control for CN<7:0>, CNP

Bypass Register

The bypass register is a one-bit scan path that can be selected to shorten the length of the system scan path, thereby reducing the number of bits per test pattern that must be applied to complete a test operation.

During capture-DR, the bypass register captures a logic 0. The bypass register order of scan is illustrated in Figure C–4.

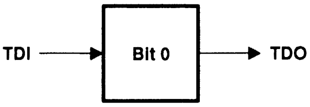


Figure C–4. Bypass Register Order of Scan

INSTRUCTION-REGISTER OPCODE DESCRIPTION

The instruction-register opcodes are shown in Table C–4. The following descriptions detail the operation of each instruction.

Table C–4 Instruction Register Opcodes

BINARY CODE† BIT 7 → BIT 0 MSB → LSB	OPCODE	DESCRIPTION	SELECTED-DATA REGISTER	MODE
00000000	Extest	External test	Boundary-scan	Test
10000010	Sample/Preload	Sample boundary	Boundary-scan	Normal
11111111	Bypass	Bypass scan	Bypass	Normal
00001010	Readbn	Boundary read	Boundary-scan	Normal
10001011	Readbt	Boundary read	Boundary-scan	Test
00001100	Celltst	Boundary self test	Boundary-scan	Normal
All others	Bypass	Bypass scan	Bypass	Normal

† Bit 7 is used to maintain even parity in the 8-bit instruction.

External Test

This instruction conforms to the IEEE Standard 1149.1-1990 extest instruction. The boundary-scan register is selected in the scan path. Data appearing at the device input pins is captured in the input boundary-scan cells (BSCs). Data that has been scanned into the output BSCs is applied to the device output pins. The device operates in the test mode.

Sample Boundary

This instruction conforms to the IEEE Standard 1149.1-1990 sample/preload instruction. The boundary-scan register is selected in the scan path. Data appearing at the device input pins is captured in the input BSCs, while data appearing at the outputs of the normal on-chip logic is captured in the output BSCs. The device operates in the normal mode.

Internal Test

This instruction conforms to the IEEE Standard 1149.1-1990 intest instruction. The boundary-scan register is selected in the scan path. Data appearing at the outputs of the normal on-chip logic is captured in the output BSCs. Data that has been scanned into the input BSCs is applied to the inputs of the normal on-chip logic. The device operates in a modified test mode in which all outputs are placed in the high-impedance state.

Boundary Read

The boundary-scan register is selected in the scan path. The value in the boundary-scan register remains unchanged during capture-DR.

Boundary Self Test

The boundary-scan register is selected in the scan path. All BSCs capture the inverse of their current values during capture-DR. In this way, the contents of the shadow latches may be read out to verify the integrity of both shift register and shadow latch elements of the boundary-scan register. The device operates in the normal mode.

Bypass Scan

This instruction conforms to the IEEE Standard 1149.1-1990 bypass instruction. The bypass register is selected in the scan path. A logic 0 value is captured in the bypass register during capture-DR. The device operates in the normal mode.

TIMING DESCRIPTION

All test operations of the devices are synchronous to the test clock (TCK). Data on the TDI, TMS, and normal-function inputs is captured on the rising edge of TCK. Data appears on the TDO and normal-function output pins on the falling edge of TCK. The TAP controller is advanced through its states (as illustrated in Figure C–1) by changing the value of TMS on the falling edge of TCK and then applying a rising edge to TCK.

A simple timing example is illustrated in Figure C–5. In this example, the TAP controller begins in the test-logic-reset state and is advanced through its states as necessary to perform one instruction register scan and one data register scan. While in the shift-IR and shift-DR states, TDI is used to input serial data, and TDO is used to output serial data. The TAP controller is then returned to the test-logic-reset state. Table C–5 explains the operation of the test circuitry during each TCK cycle.

Table C–5 Explanation of Timing Example

TCK CYCLE(S)	TAP STATE AFTER TCK	DESCRIPTION
1	Test-Logic-Reset	TMS is changed to a logic 0 value on the falling edge of TCK to begin advancing the TAP controller toward the desired state.
2	Run-Test/Idle	
3	Select-DR-Scan	
4	Select-IR-Scan	
5	Capture-IR	The IR captures the 8-bit binary value 10000001 on the rising edge of TCK as the TAP controller exits the capture-IR state.
6	Shift-IR	TDO becomes active, and TDI is made valid on the falling edge of TCK. The first bit is shifted into the TAP on the rising edge of TCK as the TAP controller advances to the next state.
7–13	Shift-IR	One bit is shifted into the IR on each TCK rising edge. With TDI held at a logic 1 value, the 8-bit binary value 11111111 is serially scanned into the IR. At the same time, the 8-bit binary value 10000001 is serially scanned out of the IR via TDO. In TCK cycle 13, TMS is changed to a logic 1 value to end the instruction register scan on the next TCK cycle. The last bit of the instruction is shifted as the TAP controller advances from shift-IR to exit1-IR.
14	Exit1-IR	TDO becomes inactive (goes to the high-impedance state) on the falling edge of TCK.
15	Update-IR	The IR is updated with the new instruction (bypass) on the falling edge of TCK.
16	Select-DR-Scan	
17	Capture-DR	The bypass register captures a logic 0 value on the rising edge of TCK as the TAP controller exits the capture-DR state.
18	Shift-DR	TDO becomes active, and TDI is made valid on the falling edge of TCK. The first bit is shifted into the TAP on the rising edge of TCK as the TAP controller advances to the next state.
19–20	Shift-DR	The binary value 101 is shifted in via TDI, while the binary value 010 is shifted out via TDO.
21	Exit1-DR	TDO becomes inactive (goes to the high-impedance state) on the falling edge of TCK.
22	Update-DR	In general, the selected-data register is updated with the new data on the falling edge of TCK.
23	Select-DR-Scan	
24	Select-IR-Scan	
25	Test-Logic-Reset	Test operation completed

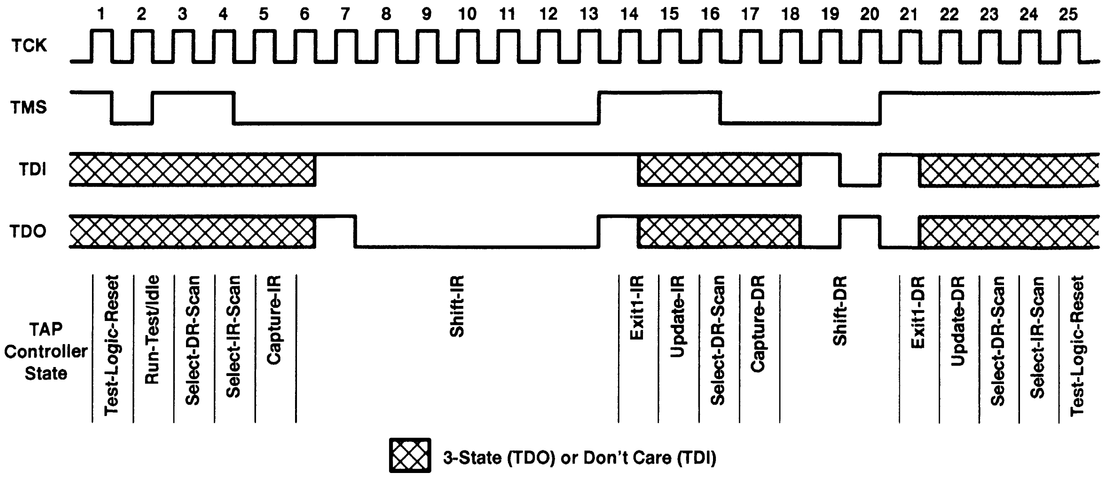


Figure C–5. Timing Example

Appendix D

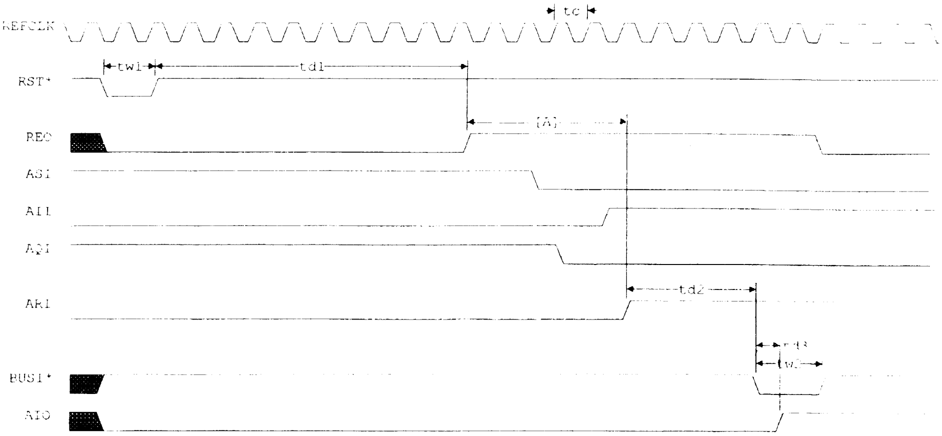
Transaction Diagrams

Appendix D
Transaction Diagrams

Reset, CSR, Host, and FB+ Transactions

- Numeric characters 0 and1 will be used to represent signal levels on but outputs and bidirects in output mode and alphabetic characters L and H will be used to represent signal levels on bus inputs and bidirects in input mode.
- SIGI<2> or SIGO<3> label is equivalent to SIG2I or SIG3O label, e.g., STO<1> is equivalent to ST1O.
- SIGI<2> indicates the input signal of SIG bit 2, and SIGO<2> indicates the output signal of SIG bit 2.
- The diagrams are not on scale.
- [A], [B] are note numbers.

Index to Reset Timing Diagrams



NOTE A: Time from REO to FB+ idle state depends on individual design. FB+ idle state is defined as ASI and AQI released and ARI and AII asserted.

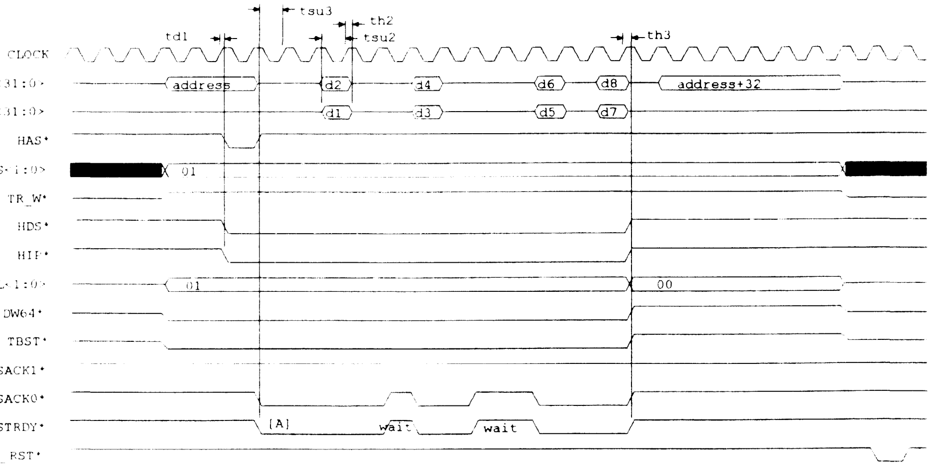
Figure D–1. Live Insertion Without Alignment Wait Feature

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
$t_{c(REFCLK)}$	Clock cycle time	25	25	25	ns
t_{w1}	Pulse duration, RST* low	1			µs

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, RST*↑ to REO↑	C_L = 20 pF	150	150	150	ms
t_{d2}	Delay time, FB+ idle state to BUSI*↓		1.6	1.6	1.6	µs
t_{d3}	Delay time, BUSI*↓ to AIO↑		7	10	14	ns
t_{w2}	Pulse duration, BUSI* low		2 tc	2 tc	2 tc	ns



NOTE A: The first data word may not have any wait states.

Figure D–34. Host Master, 8 Byte Wide, 32 Byte, Slow Burst With Wait States Read from Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su2}	Setup time, HD<31:0> on read before CLOCK↑	2	2	2	ns
t_{h2}	Hold time, HD<31:0> on read after CLOCK↑	1	1	1	ns
t_{su3}	Setup time, DSACK0*, DSACK1*, BSTRDY* before CLOCK↑	2	4	6	ns
t_{h3}	Hold time, DSACK0*, DSACK1*, BSTRDY* after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to HIP*↓	11	20	34	ns

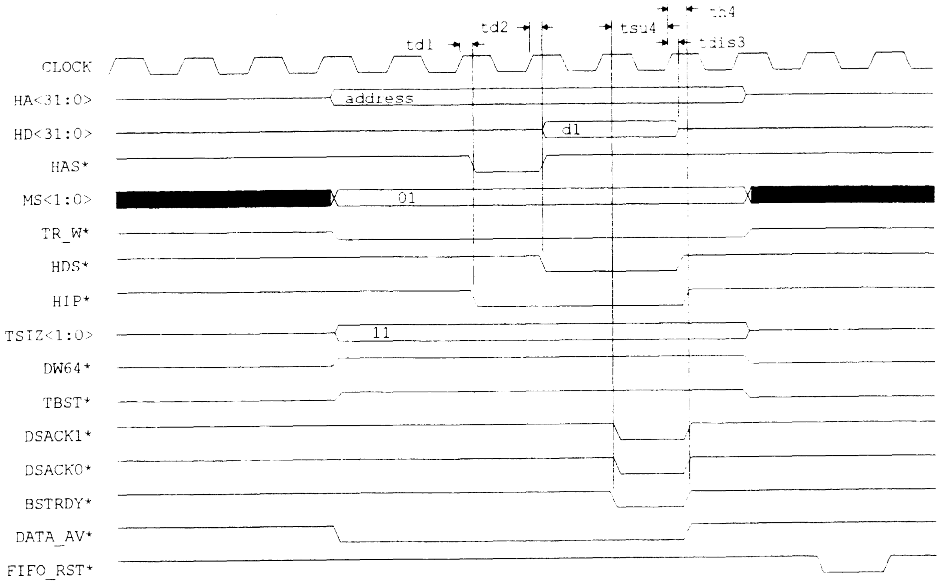


Figure D–35. Host Master, 3 Byte Write to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
tsu4	Setup time, DSACK0*, DSACK1*, BSTRDY* before CLOCK↑	2	4	6	ns
t_{h4}	Hold time, DSACK0*, DSACK1*, BSTRDY* after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to HIP*↓	11	20	34	ns
t_{d2}	Delay time, CLOCK↑ to HDS*↓	8	15	23	ns
t_{dis3}	Disable time, CLOCK↑ to HD<31:0> on write	7	13	22	ns

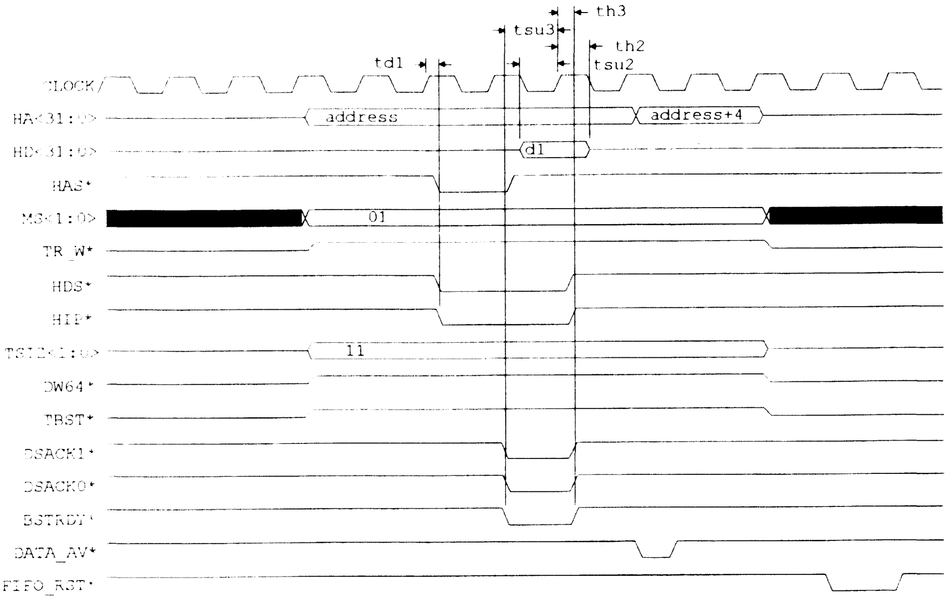


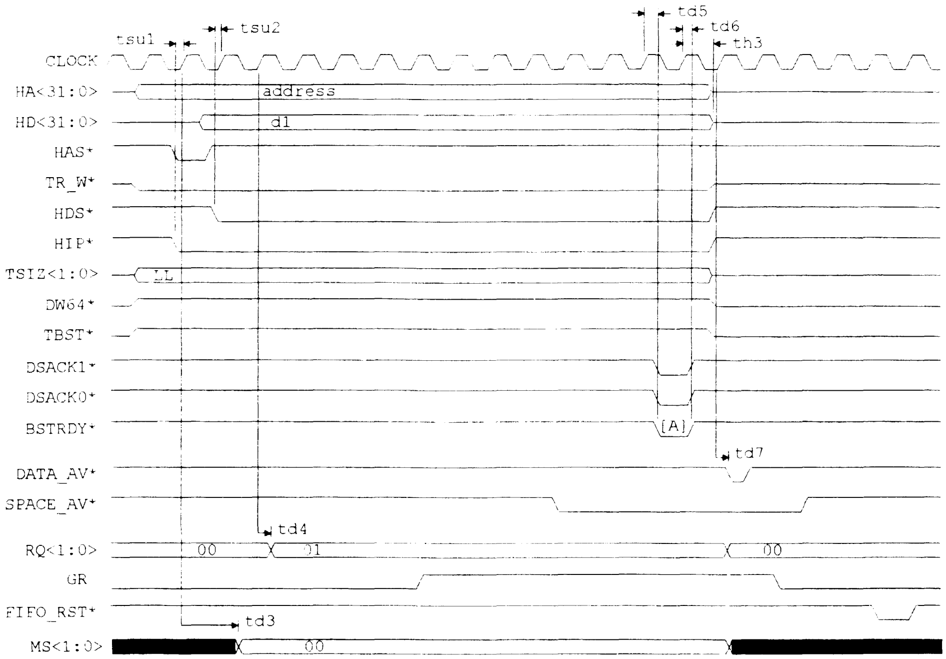
Figure D–36. Host Master, 3 Byte Read from Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su2}	Setup time, HD<31:0> on write before CLOCK↑	2	2	2	ns
t_{h2}	Hold time, HD<31:0> on write after CLOCK↑	1	1	1	ns
tsu3	Setup time, DSACK0*, DSACK1*, BSTRDY* before CLOCK↑	2	4	6	ns
t_{h3}	Hold time, DSACK0*, DSACK1*, BSTRDY* after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to HIP*↓	11	20	34	ns



NOTE A: The chip set will drive DSACK* and BSTRDY* once there is a selected slave on FB+.

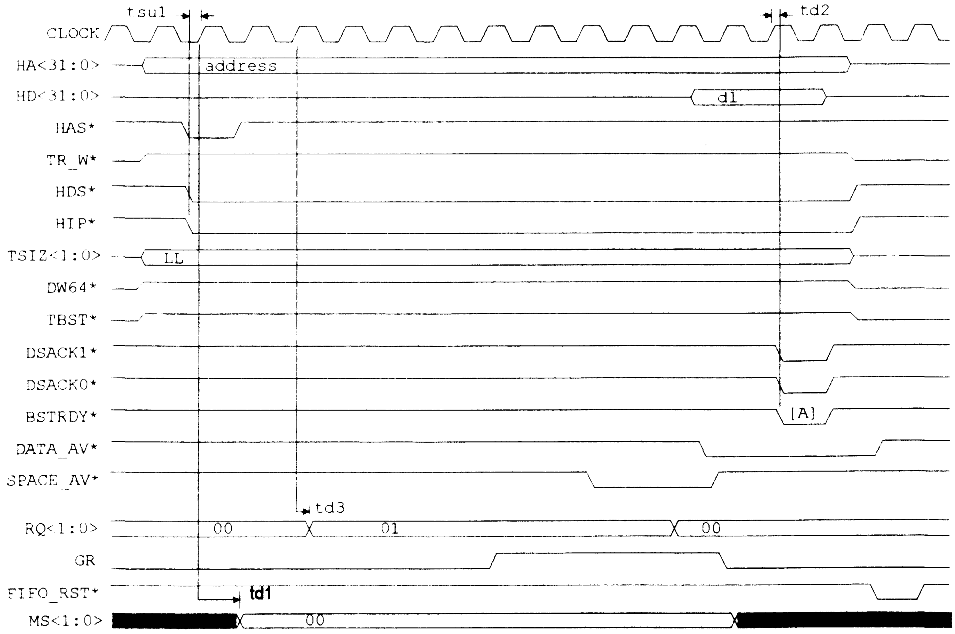
Figure D–37. Host Slave, Single Write from Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{h3}	Hold time, HD<31:0> after CLOCK↑	1	1	1	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{d3}	Delay time, MS<1:0> valid from HAS* CLK	12	25	40	ns
t_{d4}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
td5	Delay time, DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d6}	Delay time, DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns
t_{d7}	Delay time, CLOCK↑ to DATA_AV*↓	8	25	44	ns



NOTE A: The chip set will drive DSACK* and BSTRDY* when the data for the transfer in the FIFO.

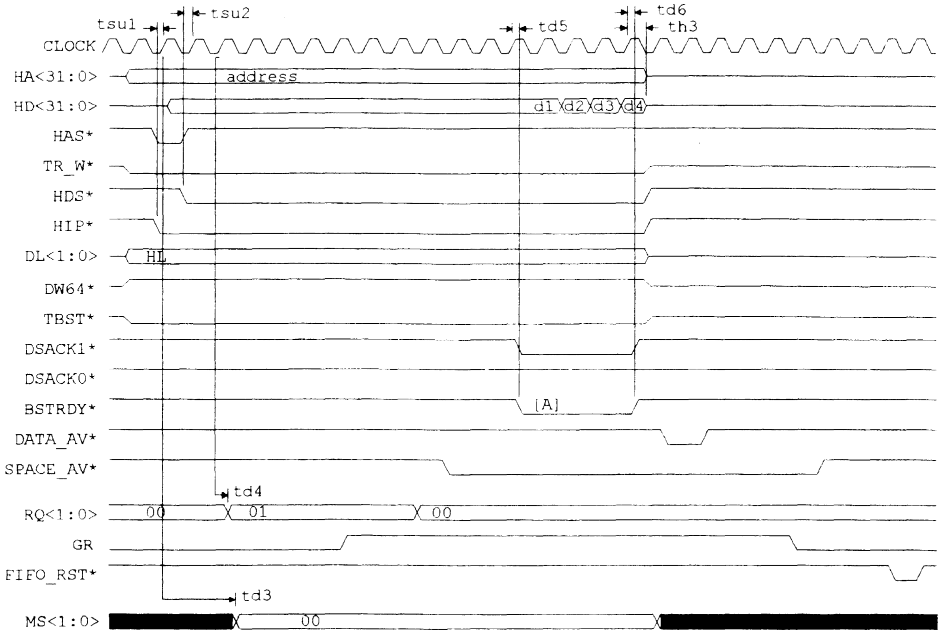
Figure D–38. Host Slave, Single Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, MS<1:0> valid from CLOCK	12	25	40	ns
t_{d2}	Delay time, DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d3}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns



NOTE A: The chip set will drive DSACK* and BSTRDY* once there is a selected slave on FB+.

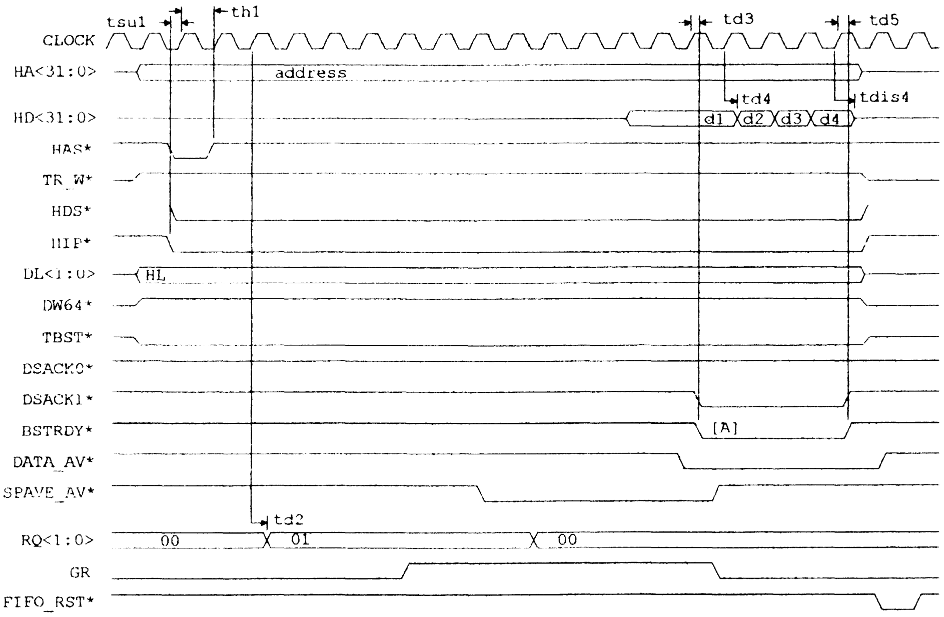
Figure D–39. Host Slave, 4 Byte Wide, 16 Byte, Fast Burst Write From Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{h3}	Hold time, HD<31:0> after CLOCK↑	1	1	1	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d3}	Delay time, CLOCK↑ to MS<1:0> valid	12	25	40	ns
t_{d4}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d5}	Delay time, DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d6}	Delay time, DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns



NOTE A: The chip set will drive DSACK* and BSTRDY* when data is available for the transaction.

Figure D–40. Host Slave, 4 Byte Wide, 16 Byte, Fast Burst Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{h1}	Hold time, HAS*↑ after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d4}	Delay time, CLOCK↑ to HD<31:0> to valid	9	20	34	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns
t_{dis4}	Disable time, CLOCK↑ to HD<31:0> to invalid	26	27	60	ns

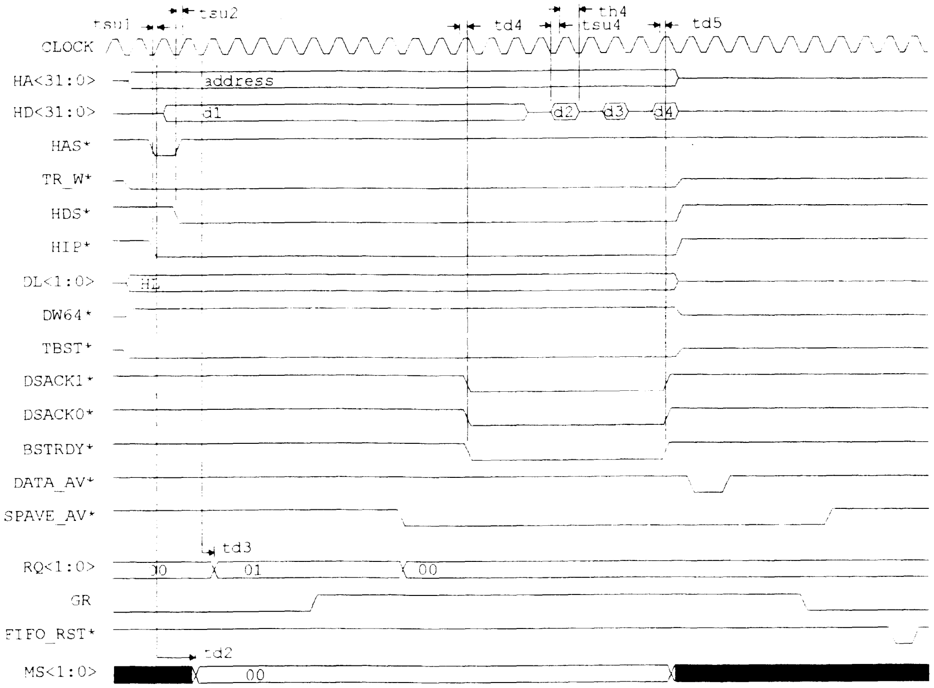


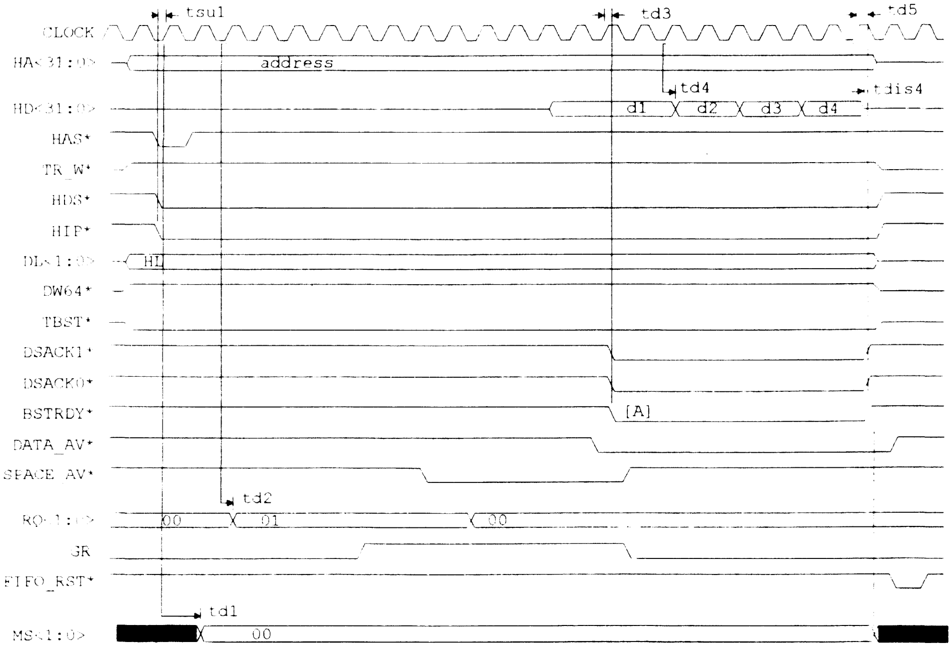
Figure D–41. Host Slave, 4 Byte Wide, 16 Byte, Slow Burst Write From Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{su4}	Setup time, HD<31:0> before CLOCK↑	2	2	2	ns
t_{h4}	Hold time, HD<31:0> after CLOCK↑	1	1	1	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to MS<1:0> valid	12	25	40	ns
t_{d3}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d4}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns



NOTE A: The chip set will drive DSACK* and BSTRDY* when data is available for the transaction.

Figure D–42. Host Slave, 4 Byte Wide, 16 Byte, Slow Burst Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to MS<1:0>	12	25	40	ns
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d4}	Delay time, CLOCK↑ to HD<31:0> to valid	9	20	34	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns
t_{dis4}	Disable time, CLOCK↑ to HD<31:0> to invalid	26	27	60	ns

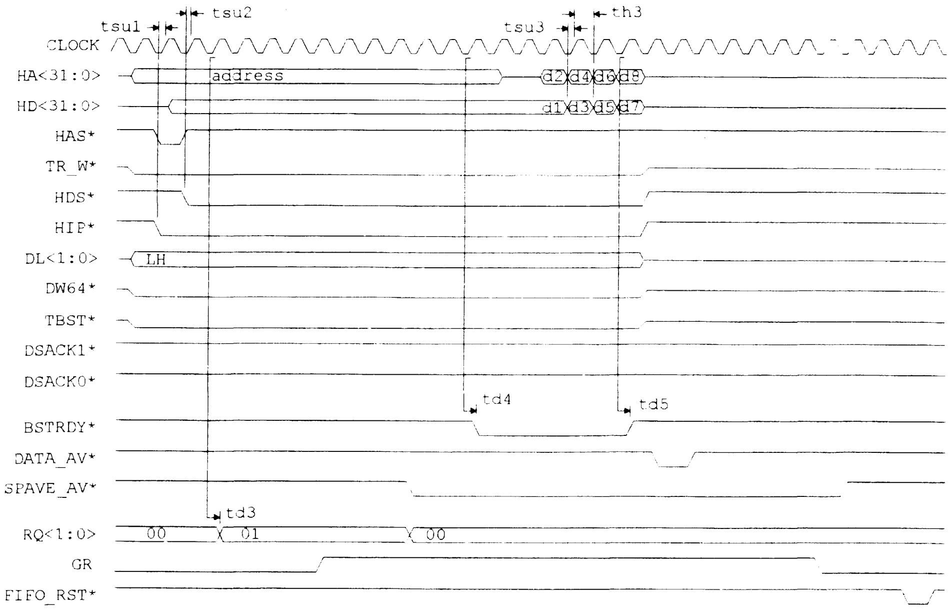


Figure D–43. Host Slave, 8 Byte Wide, 32 Byte, Fast Burst Write From Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{su3}	Setup time, HA<31:0>, HD<31:0> before CLOCK↑	2	2	2	ns
t_{h3}	Hold time, HA<31:0>, HD<31:0> after CLOCK↑	1	1	1	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d3}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d4}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns

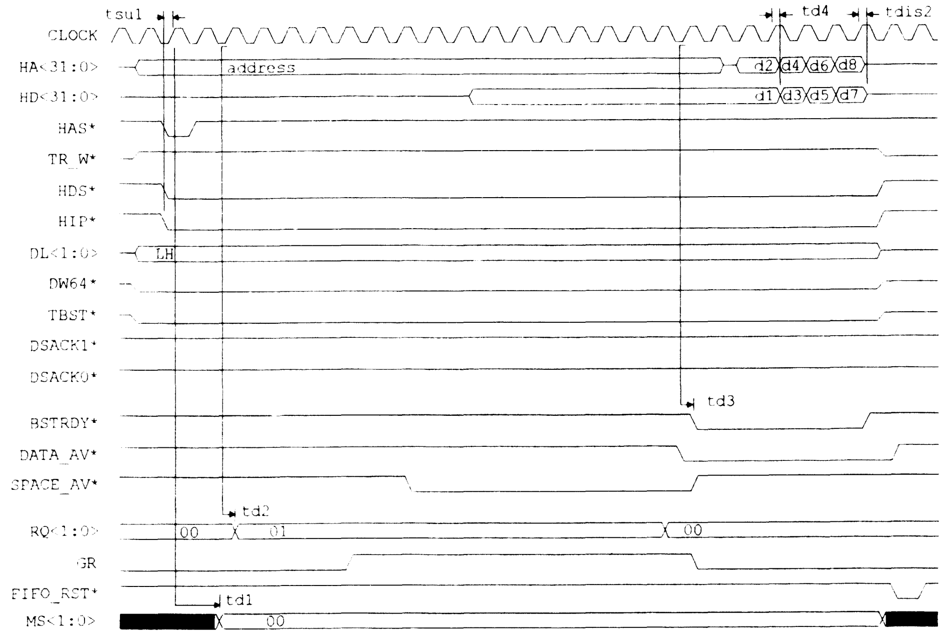


Figure D–44. Host Slave, 8 Byte Wide, 32 Byte, Fast Burst Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to MS<1:0>	12	25	40	ns
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d4}	Delay time, CLOCK↑ to HD<31:0> to valid	9	20	34	ns
t_{dis2}	Disable time, CLOCK↑ to HD<31:0> to invalid	26	27	60	ns

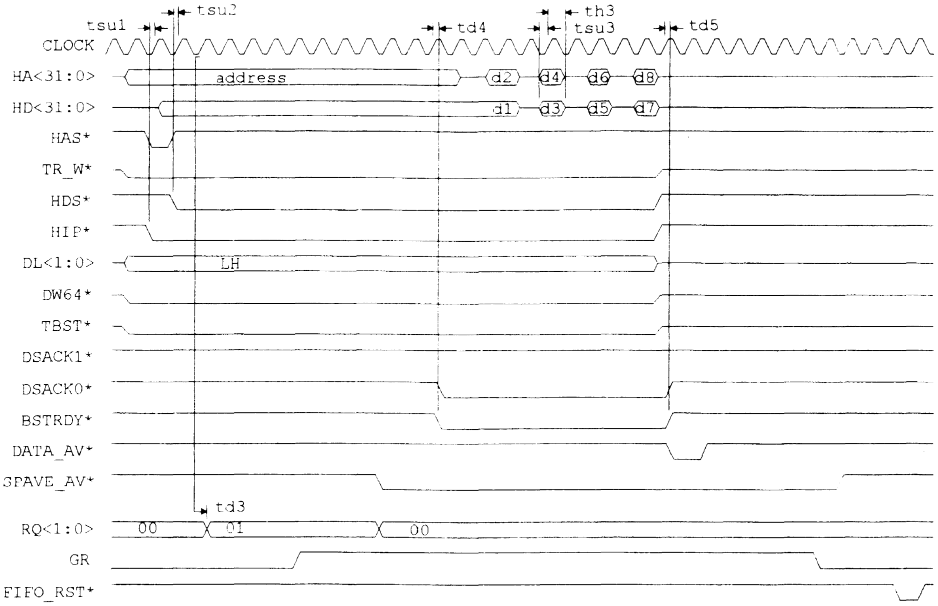


Figure D–45. Host Slave, 8 Byte Wide, 32 Byte, Slow Burst Write From Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{su3}	Setup time, HA<31:0>, HD<31:0> before CLOCK↑	2	2	2	ns
t_{h3}	Hold time, HA<31:0>, HD<31:0> after CLOCK↑	1	1	1	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d3}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d4}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns

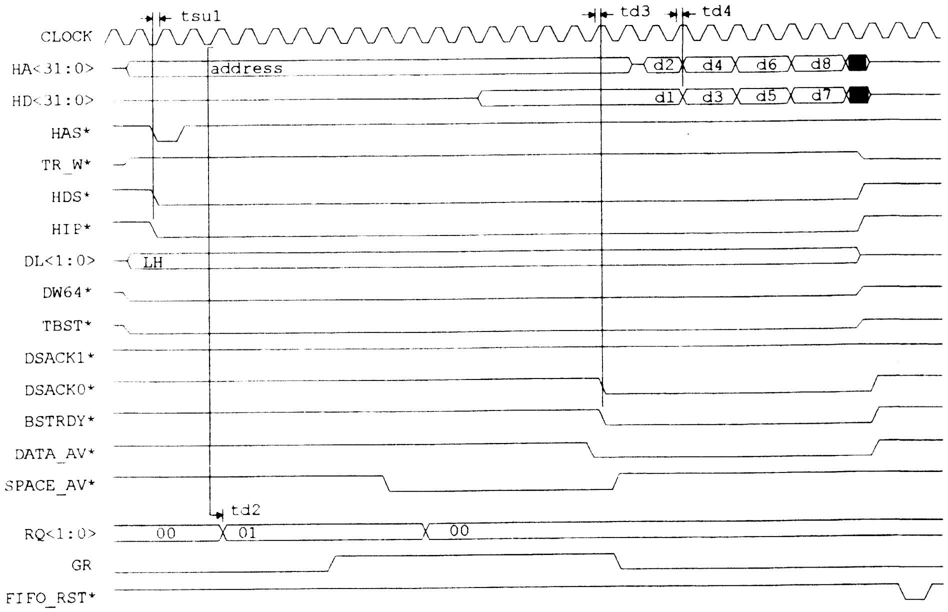


Figure D–46. Host Slave, 8 Byte Wide, 32 Byte, Slow Burst Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d4}	Delay time, CLOCK↑ to HD<31:0> to valid	9	20	34	ns

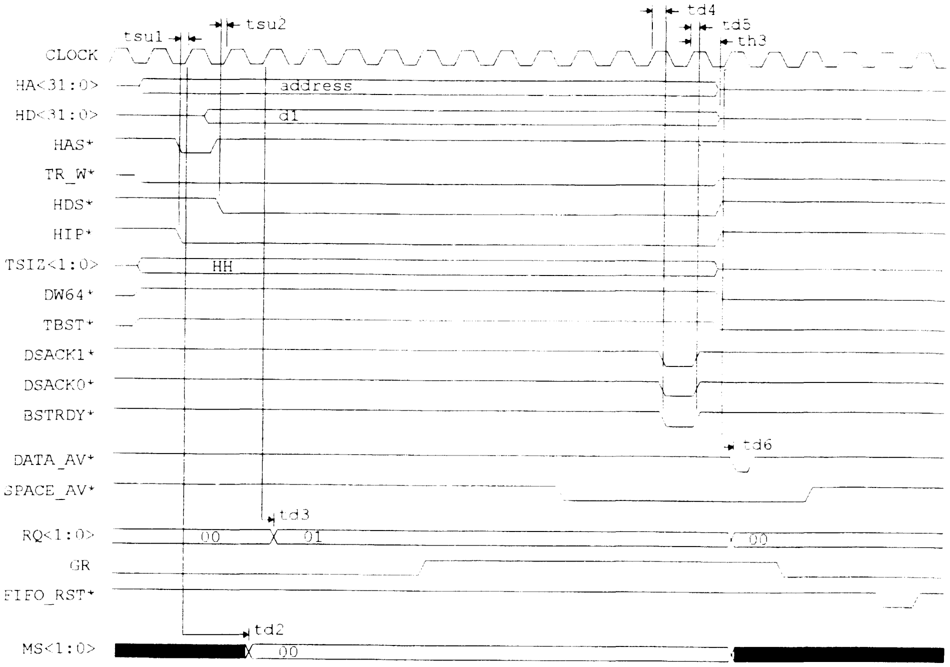


Figure D–47. Host Slave, 3 Byte Write From Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns
t_{su2}	Setup time, HDS*↓ before CLOCK↑	0	0	0	ns
t_{h3}	Hold time, HA<31:0>, HD<31:0> after CLOCK↑	2	2	2	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to MS<1:0>	12	25	40	ns
t_{d3}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d4}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d5}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to invalid	16	24	42	ns
t_{d6}	Delay time, CLOCK↑ to DATA_AV*↓	8	25	44	ns

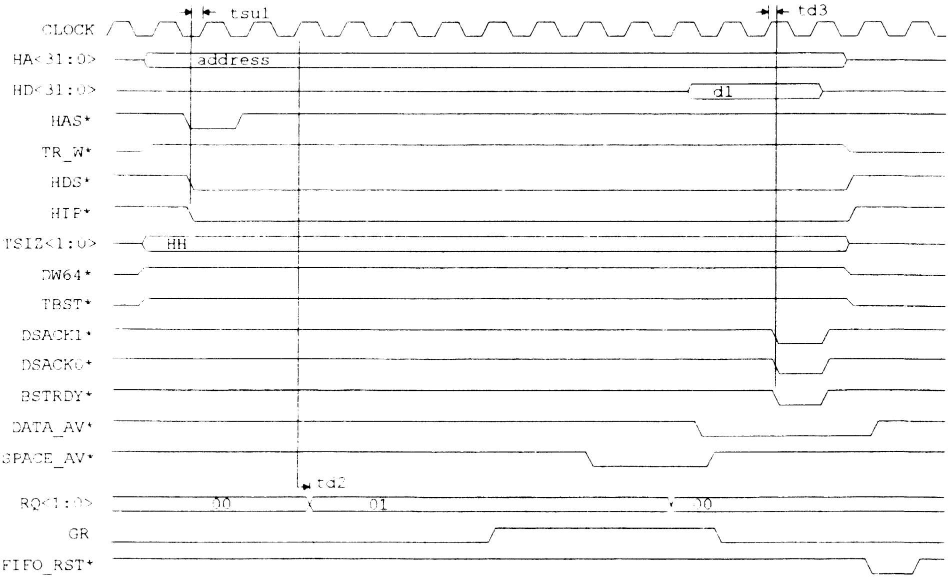


Figure D–48. Host Slave, 3 Byte Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓ before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns

36 Bit Addressing

Figure D–49. Host Master, 36 Bit Address Bus, 4 Byte Wide, Fast Burst Write to Host Interface

Figure D–50. Host Slave, 36 Bit Address Bus, 8 Byte Wide, Fast Burst Read to Host Interface

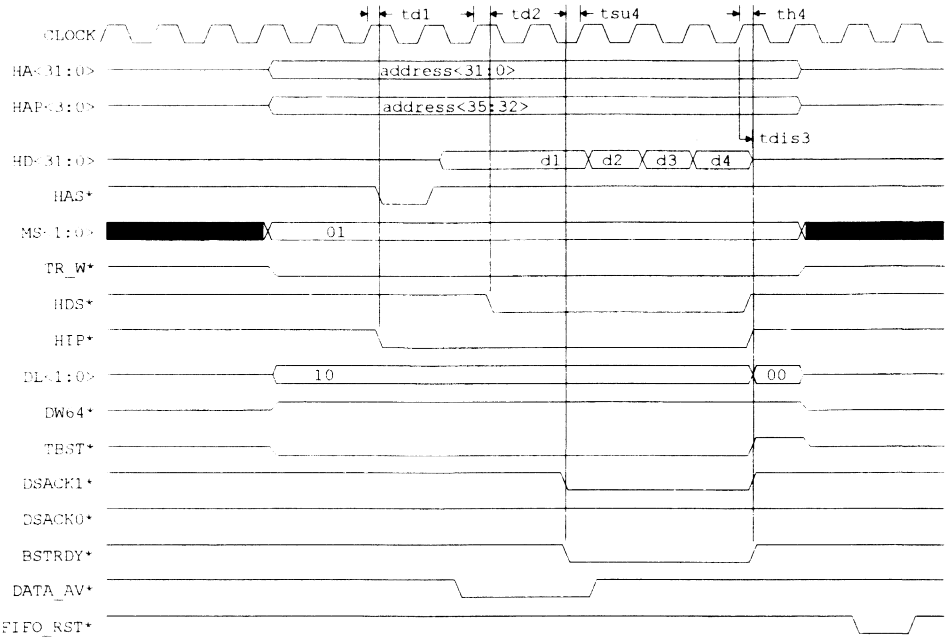


Figure D–49. Host Master, 36 Bit Address Bus, 4 Byte Wide, Fast Burst Write to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su4}	Setup time, DSACK0*, DSACK1*, BSTRDY* before CLOCK↑	2	4	6	ns
t_{h4}	Hold time, DSACK0*, DSACK1*, BSTRDY* after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to HIP*↓	11	20	34	ns
t_{d2}	Delay time, CLOCK↑ to HDS*↓ on write	8	15	23	ns
t_{dis3}	Disable time, CLOCK↑ to HD<31:0> on write	7	13	22	ns

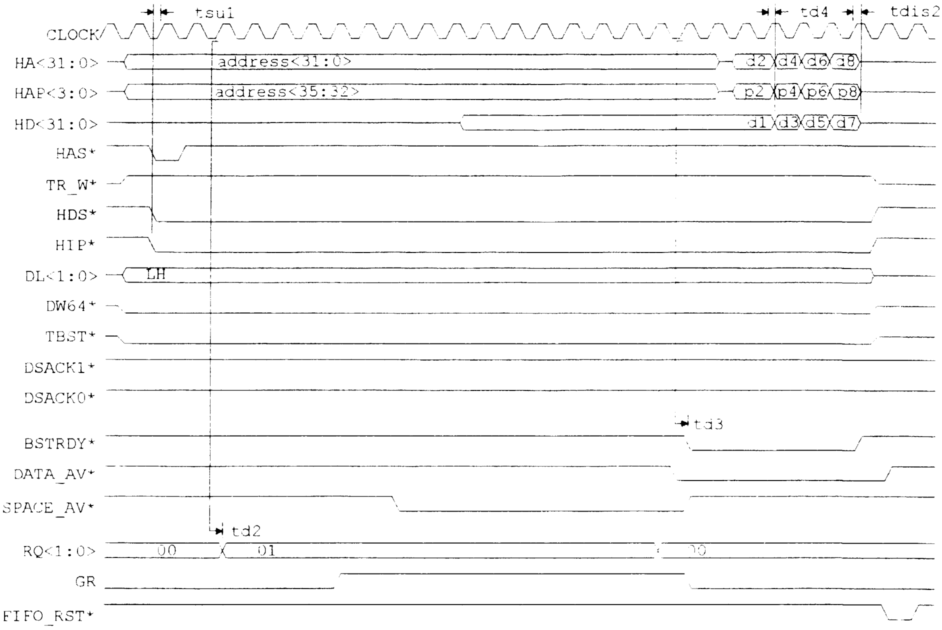


Figure D–50. Host Slave, 36 Bit Address Bus, 8 Byte Wide, Fast Burst Read to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, HIP*↓before CLOCK↑	2	4	6	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, CLOCK↑ to RQ<1:0>	7	12	20	ns
t_{d3}	Delay time, CLOCK↑ to DSACK0*, DSACK1*, BSTRDY* to valid	6	11	20	ns
t_{d4}	Delay time, CLOCK↑ to HD<31:0> to valid	9	20	34	ns
t_{dis2}	Disable time, CLOCK↑ to HD<31:0> to invalid	26	27	60	ns

Special

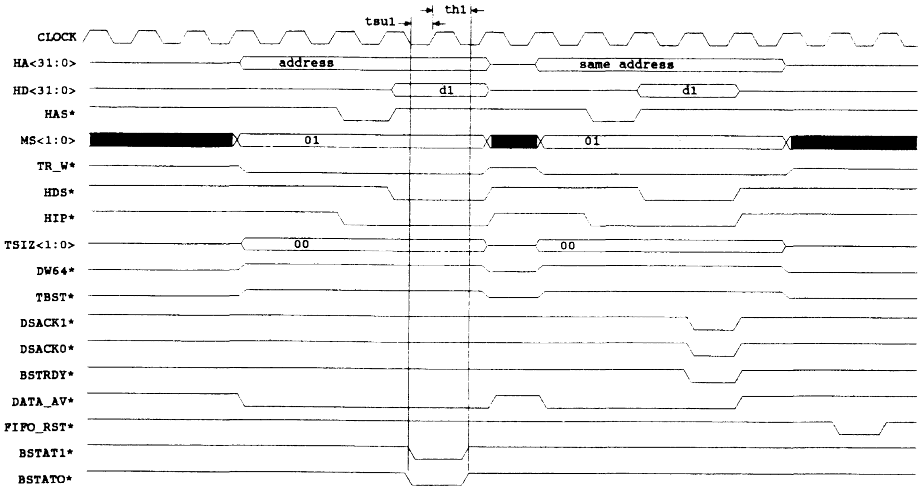
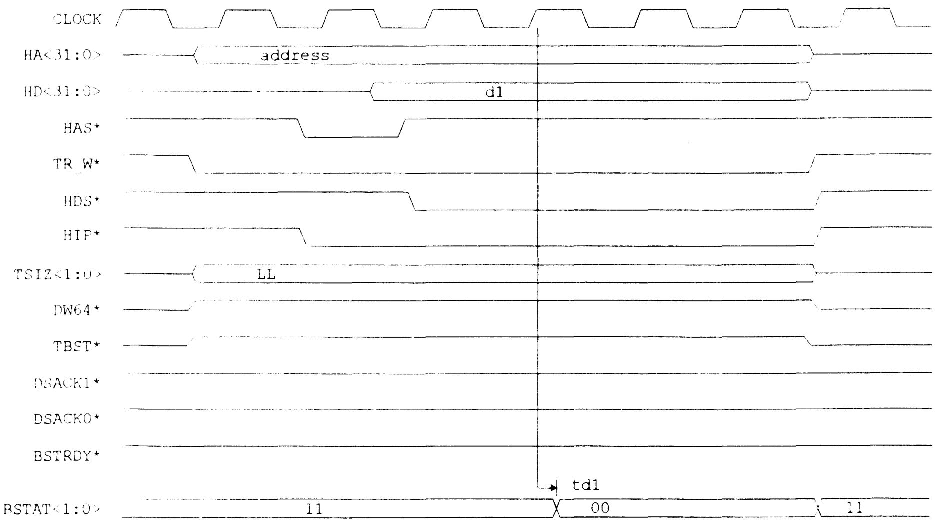


Figure D–51. Host Master With Back-Off/Retry Single Write to Host Interface

timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{su1}	Setup time, BSTAT1*, BSTAT0* before CLOCK↑	0	0	0	ns
t_{h1}	Hold time, BSTAT1*, BSTAT0* after CLOCK↑	0	0	0	ns



NOTE A: The host interface master is responsible for reconnecting to the same address on this transaction.

Figure D–52. Host Slave With Back-Off/Retry Single Write from Host Interface

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to BSTAT1*, BSTAT0*	8	15	27	ns

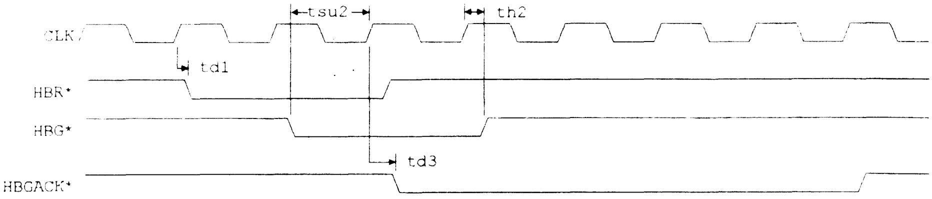


Figure D–53. Host Interface Arbitration

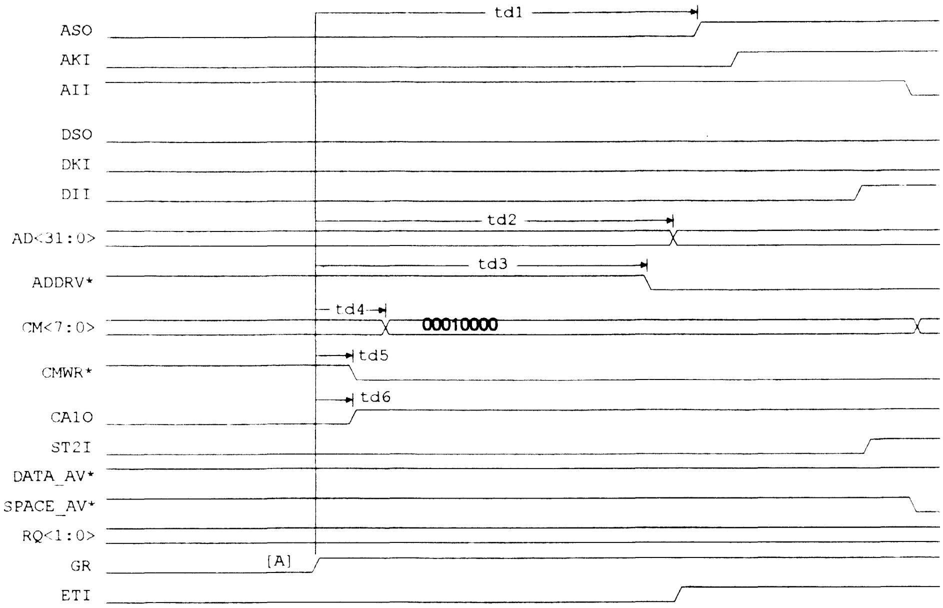
timing requirements over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{su2}	Setup time, HBG*↓ before CLOCK↑	0	0	0	ns
t_{h2}	Hold time, HBG*↑ after CLOCK↑	0	0	0	ns

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, CLOCK↑ to HBR*↓	6	11	19	ns
t_{d3}	Delay time, CLOCK↑ to HBGACK*↓	17	24	40	ns

FB+ Connection Phase

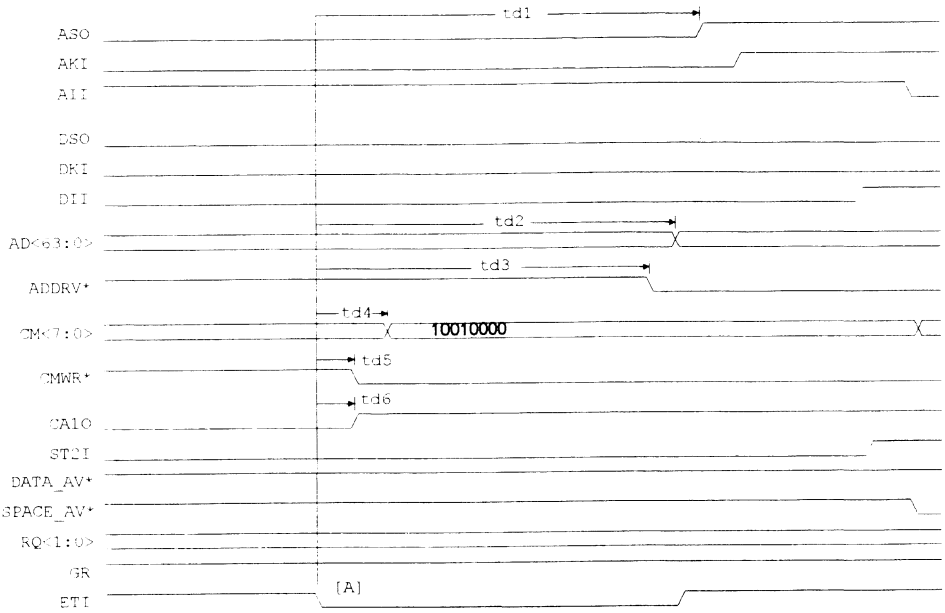


NOTE A: ETI indicates no other master is on Futurebus+ so the chip set can access Futurebus+ immediately after GR.

Figure D–54. FB+ Master, Connection Phase 32 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, GR↑ to ASO↑	2 c + 38	2 c + 73	2 c + 132	ns
t_{d2}	Delay time, GR↑ to AD<31:0>	2 c + 29	2 c + 56	2 c + 101	ns
t_{d3}	Delay time, GR↑ to ADDRV*↓	2 c + 18	2 c + 32	2 c + 57	ns
t_{d4}	Delay time, GR↑ to CM<7:0>	6	12	24	ns
t_{d5}	Delay time, GR↑ to CMWR*↓	7	12	21	ns
t_{d6}	Delay time, GR↑ to CA1O↑	2 c + 23	2 c + 37	2 c + 64	ns



NOTE A: The chip set must wait for ETI to be low before driving any Futurebus+ signals, even though GR is asserted high.

Figure D–55. FB+ Master, Connection Phase 64 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ETI↓ to ASO↑	2 c + 38	2 c + 73	2 c + 132	ns
t_{d2}	Delay time, ETI↓ to AD<63:0>	2 c + 29	2 c + 56	2 c + 101	ns
t_{d3}	Delay time, ETI↓ to ADDRV*↓	2 c + 18	2 c + 32	2 c + 57	ns
t_{d4}	Delay time, ETI↓ to CM<7:0>	6	12	24	ns
t_{d5}	Delay time, ETI↓ to CMWR*↓	7	12	21	ns
t_{d6}	Delay time, ETI↓ to CA1O↑	2 c + 23	2 c + 37	2 c + 64	ns

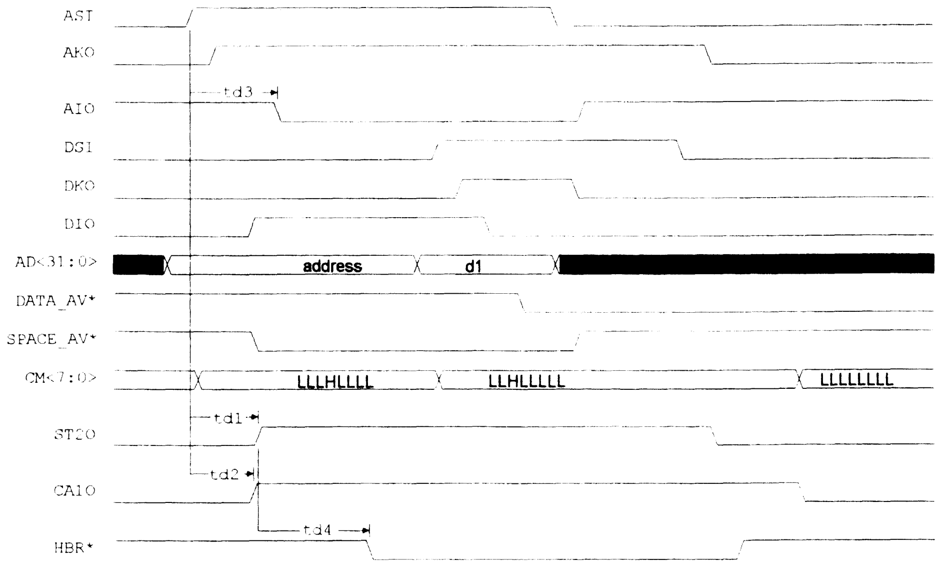


Figure D–56. FB+ Slave, Selected Connection Phase 32 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↑ to ST20↑	32	62	112	ns
t_{d2}	Delay time, ASI↑ to CA1O↑	32	62	112	ns
t_{d3}	Delay time, ASI↑ to AIO↓	40	775	137	ns
t_{d4}	Delay time, ST20↑ to HBR*↓	2 c + 6	2 c + 11	2 c + 19	ns

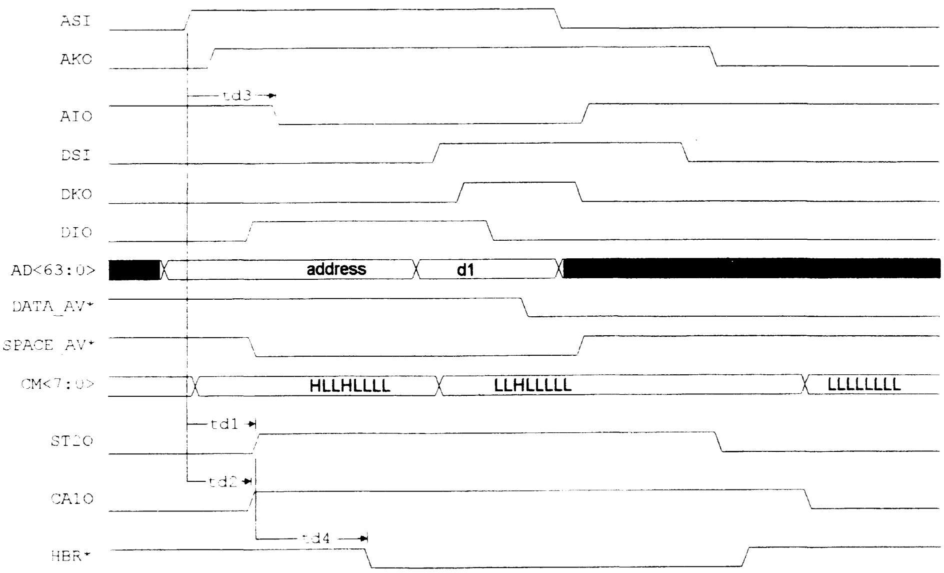


Figure D–57. FB+ Slave, Selected Connection Phase 64 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↑ to ST20↑	32	7462	112	ns
t_{d2}	Delay time, ASI↑ to CA1O↑	32	7462	112	ns
t_{d3}	Delay time, ASI↑ to AIO↓	40	75	137	ns
t_{d4}	Delay time, ST20↑ to HBR*↓	2 c + 6	2 c + 11	2 c + 19	ns

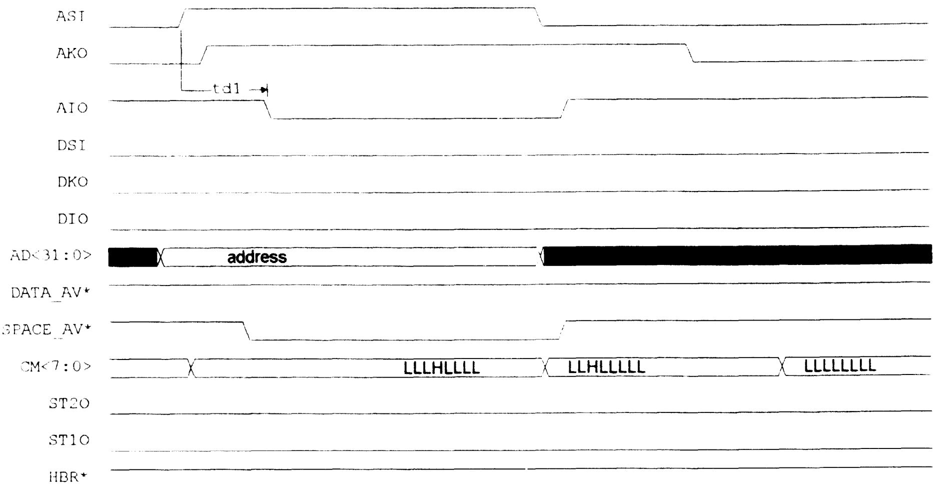


Figure D–58. FB+ Slave, Unselected Connection Phase 32 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↑ to AIO↓	40	75	137	ns

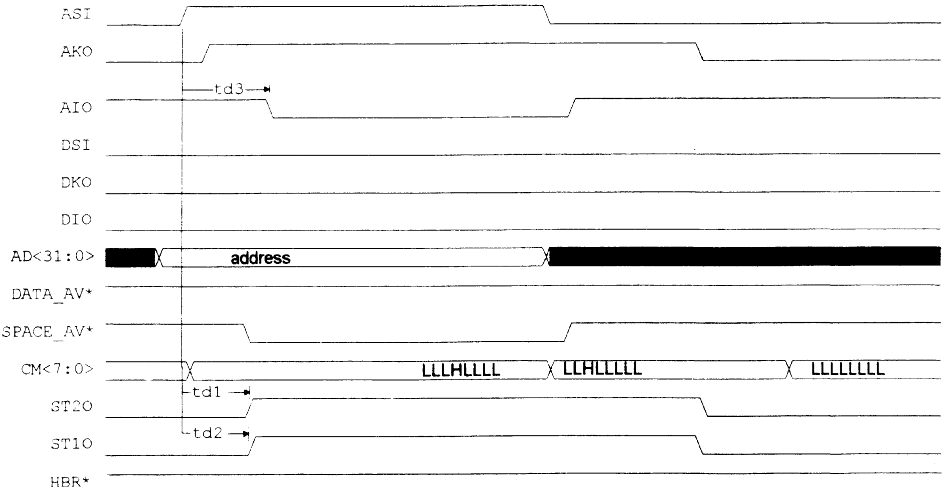


Figure D–59. FB+ Slave, Busy Connection Phase 32 Bit Address Bus

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↑ to ST2O↑	32	62	112	ns
t_{d2}	Delay time, ASI↑ to ST1O↑	32	62	112	ns
t_{d3}	Delay time, ASI↑ to AIO↓	40	75	137	ns

FB+ Data Phase Timing

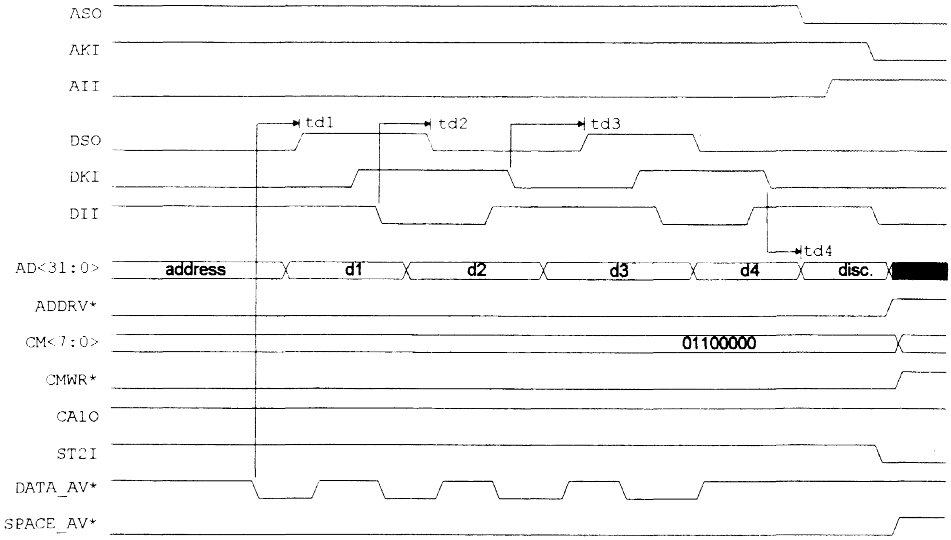


Figure D–62. FB+ Master, Compelled 4 Byte Wide, 16 Byte Write to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to DSO↑	33	61	108	ns
t_{d2}	Delay time, DII↓ to DSO↓	32	61	111	ns
t_{d3}	Delay time, DKI↓ to DSO↑	29	57	102	ns
t_{d4}	Delay time, DKI↓ to data phase release	26	51	91	ns

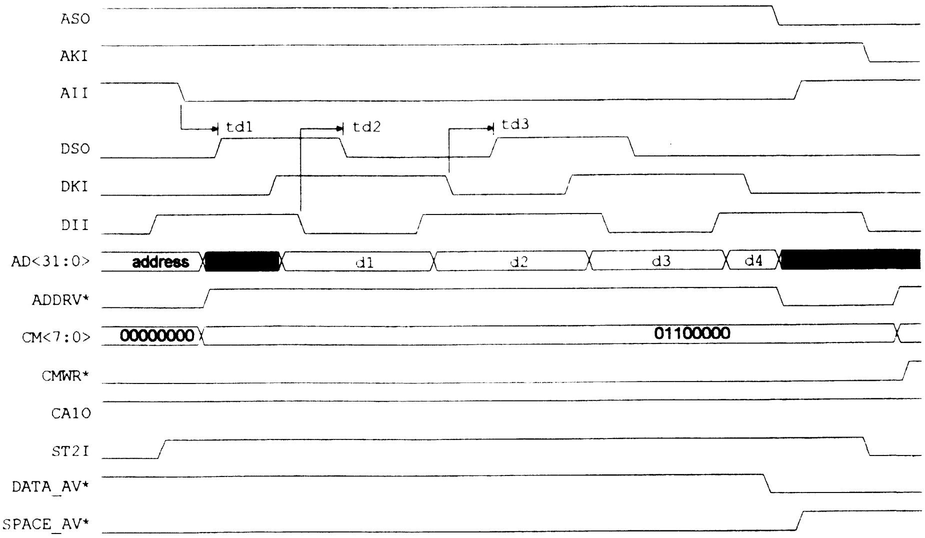


Figure D–63. FB+ Master, Compelled 4 Byte Wide, 16 Byte Read from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, AII↓ to DSO↑	21	37	66	ns
t_{d2}	Delay time, DII↓ to DSO↓	29	54	95	ns
t_{d3}	Delay time, DKI↓ to DSO↑	28	52	92	ns

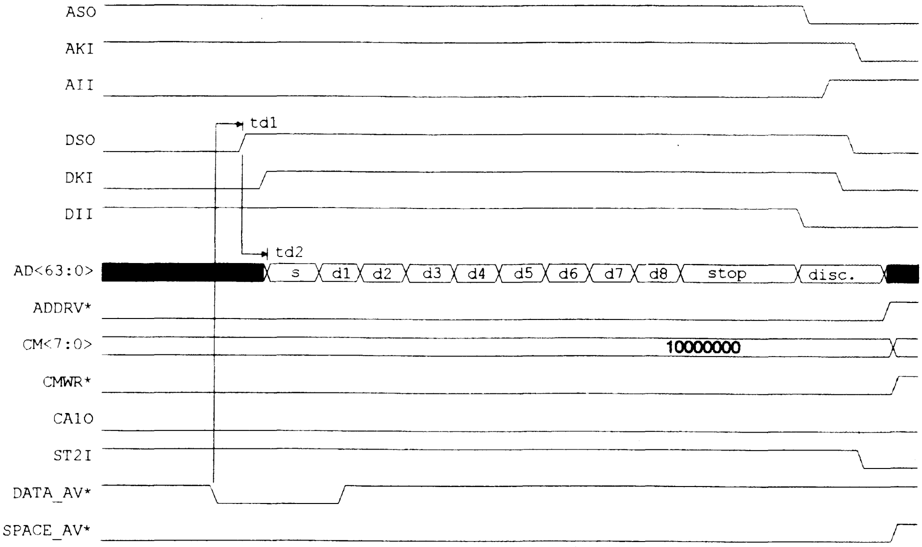


Figure D–64. FB+ Master, Packet 8 Byte Wide, 64 Byte, 8 Bit Packet Write to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to DSO↑	33	61	108	ns
t_{d2}	Delay time, DSO↑ to start bit of AD<63:0>	54	84	104	ns

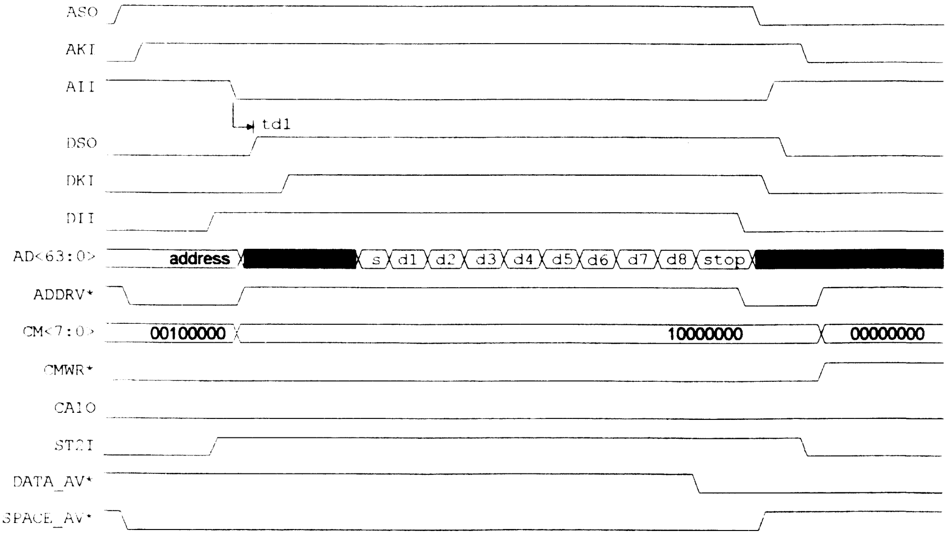


Figure D–65. FB+ Master, Packet 8 Byte Wide, 64 Byte, 8 Bit Packet Read from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d2}	Delay time, AII↓ to DSO↑	20	37	58	ns

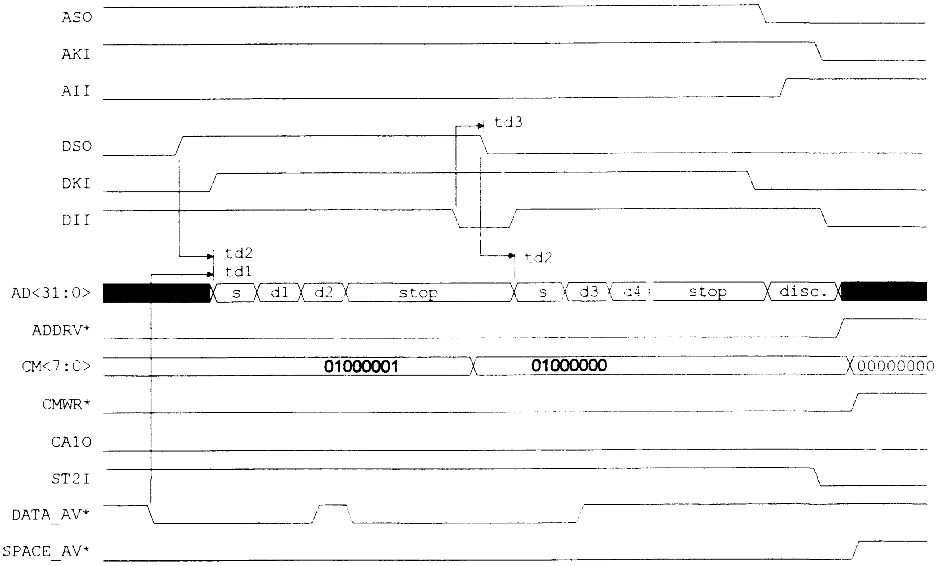


Figure D–66. FB+ Master, Packet 4 Byte Wide, 16 Byte, 2 Bit Packet Write to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to start bit of AD<31:0>	65	102	140	ns
t_{d2}	Delay time, DSO↑ to start bit of AD<31:0>	54	84	104	ns
t_{d3}	Delay time, DII↓ to DSO↓	14	27	45	ns

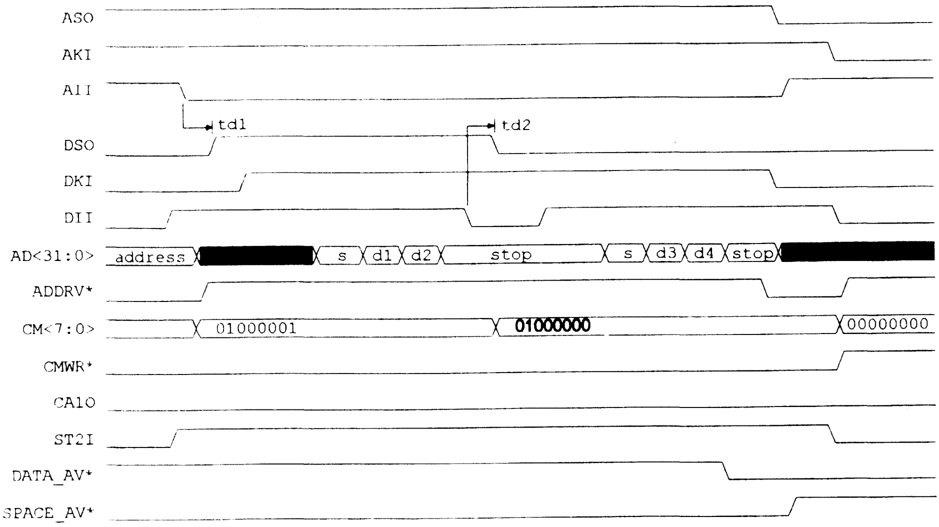


Figure D–67. FB+ Master, Packet 4 Byte Wide, 16 Byte, 2 Bit Packet Read from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, AII↓ to DSO↑	20	37	58	ns
t_{d2}	Delay time, DII↓ to DSO↓	24	58	102	ns

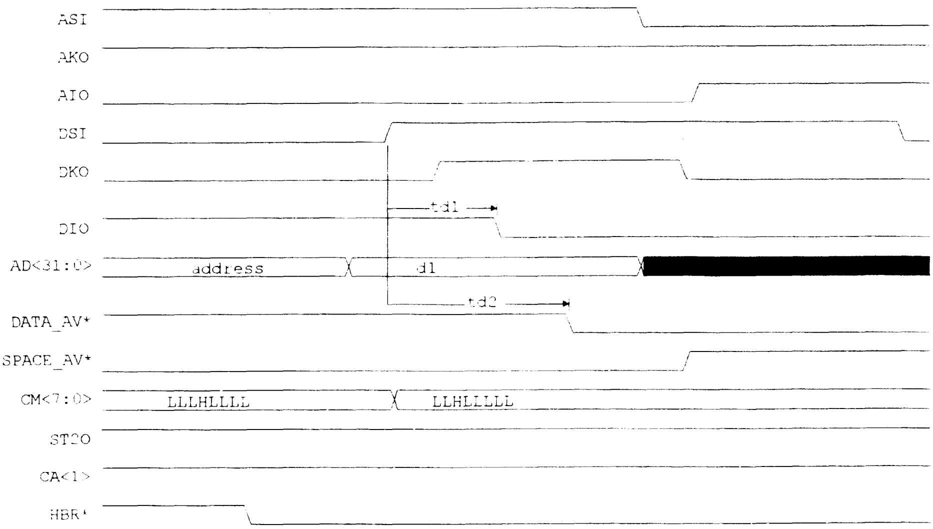


Figure D–68. FB+ Slave, Compelled 4 Byte Wide, 4 Byte, Write from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, First DSI↑ to DIO↓	23	60	124	ns
t_{d2}	Delay time, DSI↑ to DATA_AV*↓	23	40	72	ns

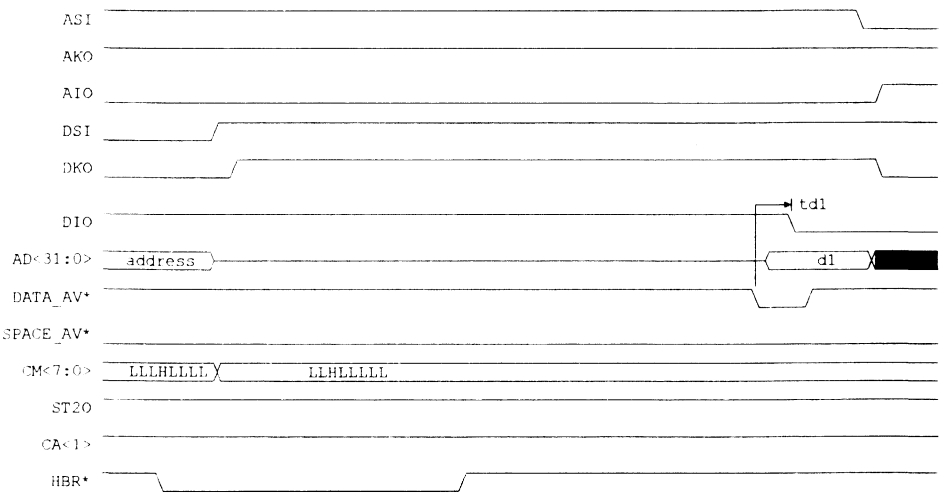


Figure D–69. FB+ Slave, Compelled 4 Byte Wide, 4 Byte, Read to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to DIO↓	33	62	111	ns

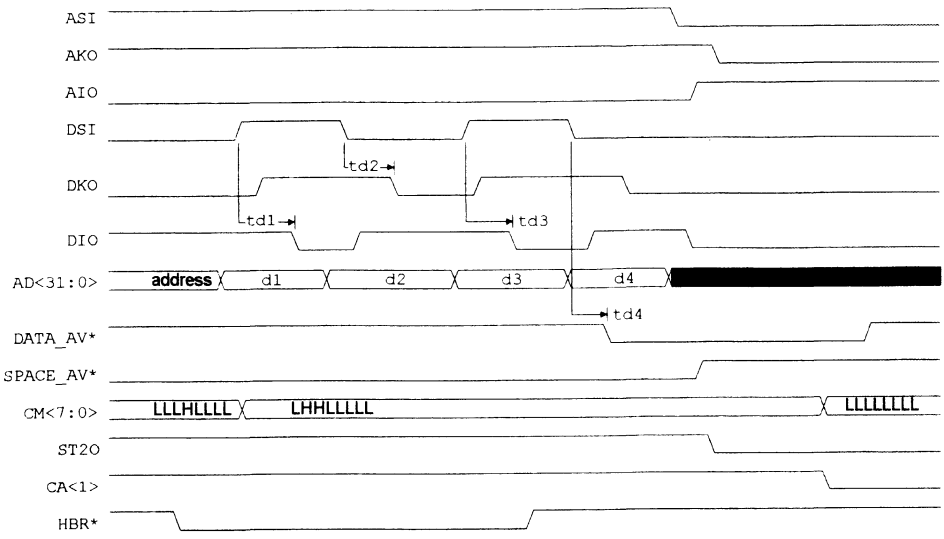


Figure D–70. FB+ Slave, Compelled 4 Byte Wide, 16 Byte, Write from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

	PARAMETER	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, First DSI↑ to DIO↓	23	60	124	ns
t_{d2}	Delay time, DSI↓ to DKO↓	21	40	72	ns
t_{d3}	Delay time, DSI↑ to DIO↓	23	43	77	ns
t_{d4}	Delay time, DSI↓ to DATA_AV*↓	23	40	72	ns

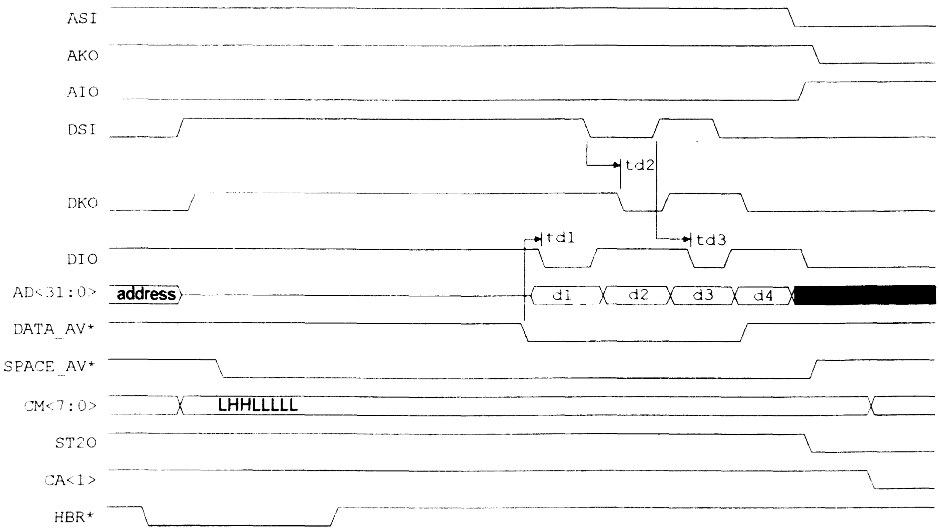


Figure D–71. FB+ Slave, Compelled 4 Byte Wide, 16 Byte, Read to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to DIO↓	33	62	111	ns
t_{d2}	Delay time, DSI↓ to DKO↓	32	58	104	ns
t_{d3}	Delay time, DSI↑ to DIO↓	32	56	99	ns

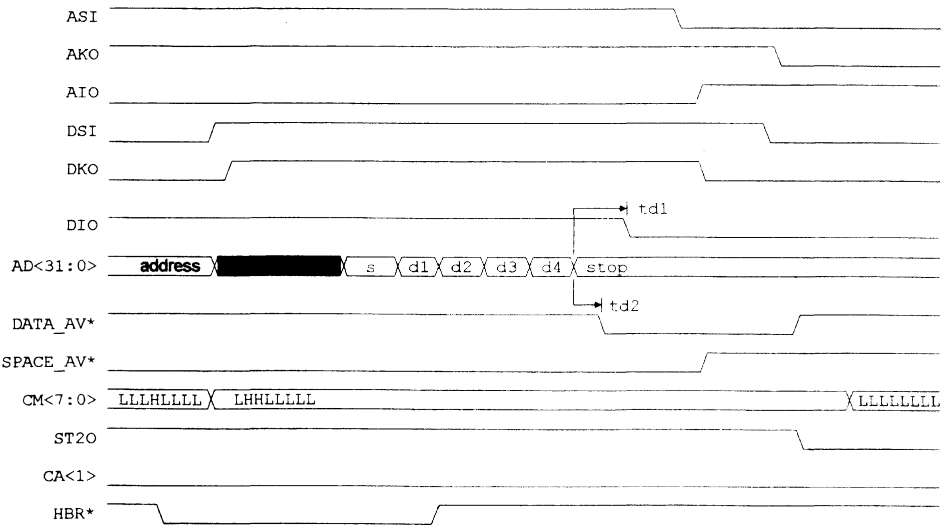


Figure D–72. FB+ Slave, Packet 4 Byte Wide, 16 Byte, 4 Bit Packet Write from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, stop bit of AD<31:0> to DIO↓	35	62	103	ns
t_{d2}	Delay time, stop bit of AD<31:0> to DATA_AV*↓	8	24	47	ns

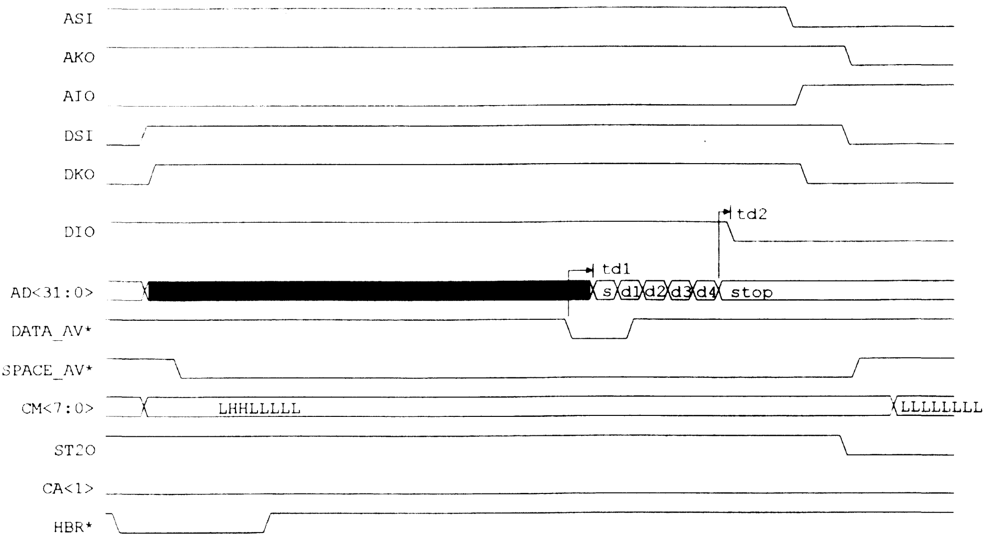


Figure D–73. FB+ Slave, Packet 4 Byte Wide, 16 Byte, 4 Bit Packet Read to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to start bit of AD<31:0>	61	97	140	ns
t_{d2}	Delay time, stop bit of AD<31:0> to DIO↓	9	16	25	ns

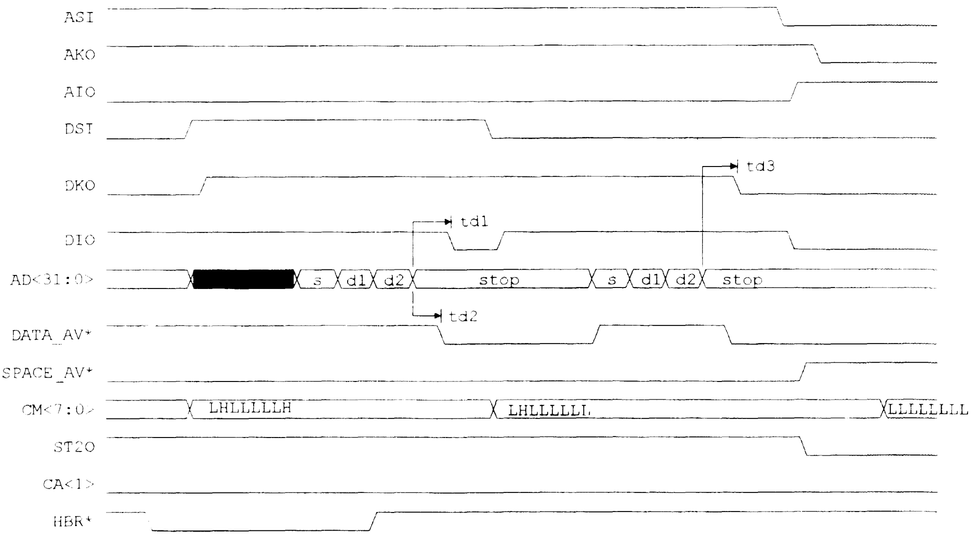
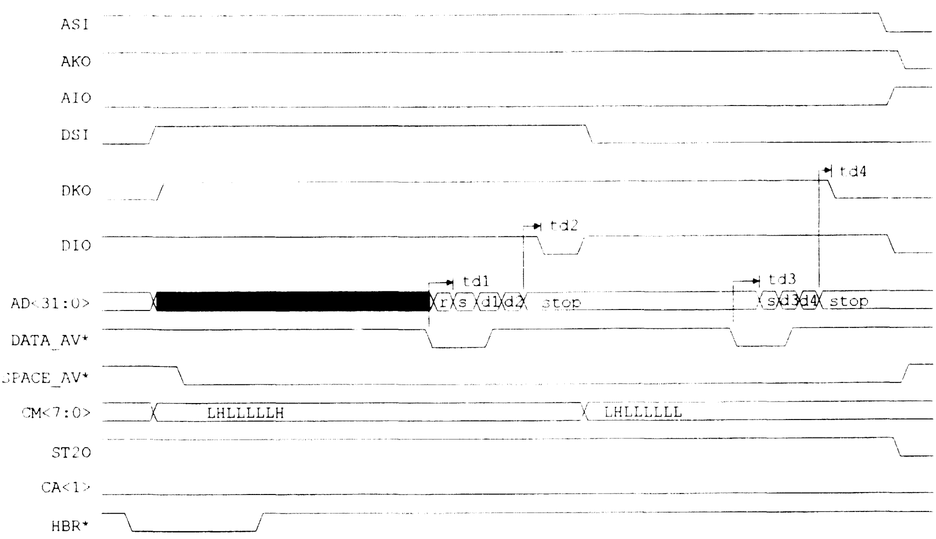


Figure D–74. FB+ Slave, Packet 4 Byte Wide, 16 Byte, 2 Bit Packet Write from FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, stop bit of AD<31:0> to DIO↓	35	62	103	ns
t_{d2}	Delay time, stop bit of AD<31:0> to DATA_AV*↓	8	24	47	ns
t_{d3}	Delay time, stop bit of AD<31:0> to DKO↓	34	59	94	ns



NOTE A: r represents a released bus

Figure D–75. FB+ Slave, Packet 4 Byte Wide, 16 Byte, 2 Bit Packet Read to FB+

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DATA_AV*↓ to start bit of AD<31:0>	61	97	140	ns
t_{d2}	Delay time, stop bit of AD<31:0> to DIO↓	9	16	25	ns
t_{d3}	Delay time, DATA_AV*↓ to start bit of AD<31:0>	61	97	140	ns
t_{d4}	Delay time, stop bit of AD<31:0> to DKO↓	9	13	23	ns

FB+ Disconnection Phase

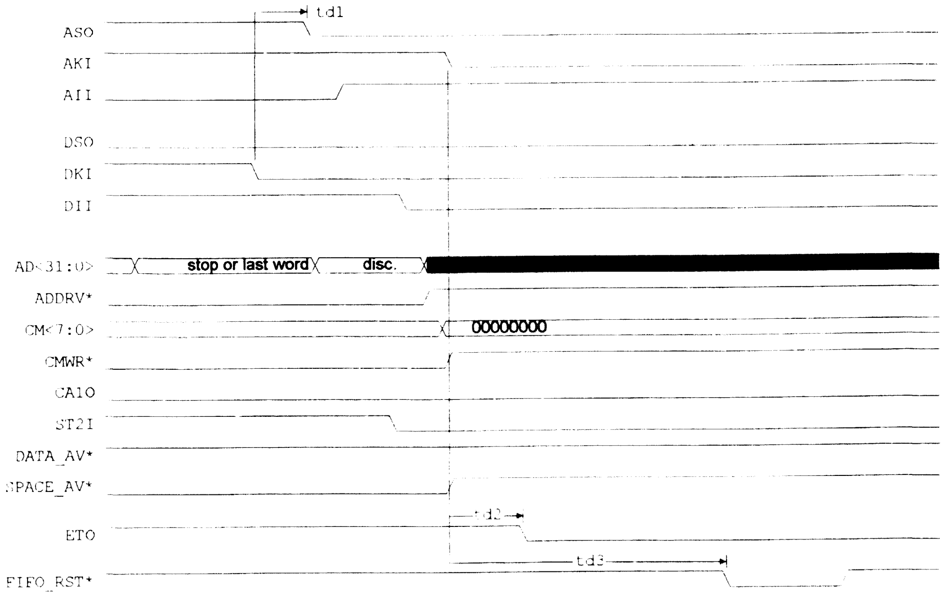


Figure D–76. FB+ Master, Write Disconnection Phase

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DKI↓ to ASO↓	37	68	124	ns
t_{d2}	Delay time, AKI↓ to ETO↓	20	35	63	ns
t_{d3}	Delay time, AKI↓ to FIFO_RST*↓	3 c + 7	3 c + 13	3 c + 22	ns

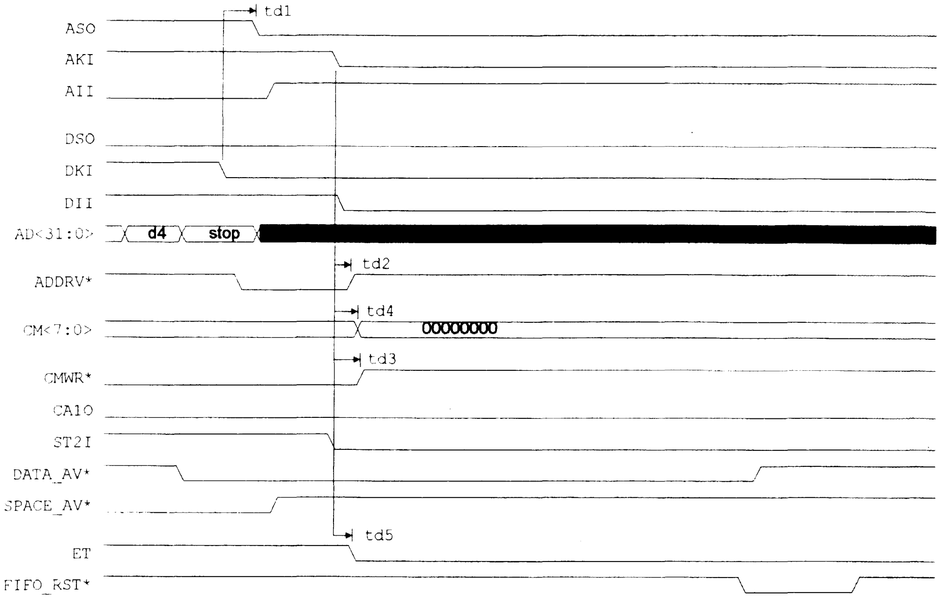


Figure D–77. FB+ Master, Read Disconnection Phase

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, DKI↓ to ASO↓	48	90	160	ns
t_{d2}	Delay time, AKI↓ to ADDRV*↑	17	29	53	ns
t_{d3}	Delay time, AKI↓ to CMWR*↑	21	37	69	ns
t_{d4}	Delay time, AKI↓ to CM<7:0>	21	40	65	ns
t_{d5}	Delay time, AKI↓ to ETO↓	20	35	65	ns

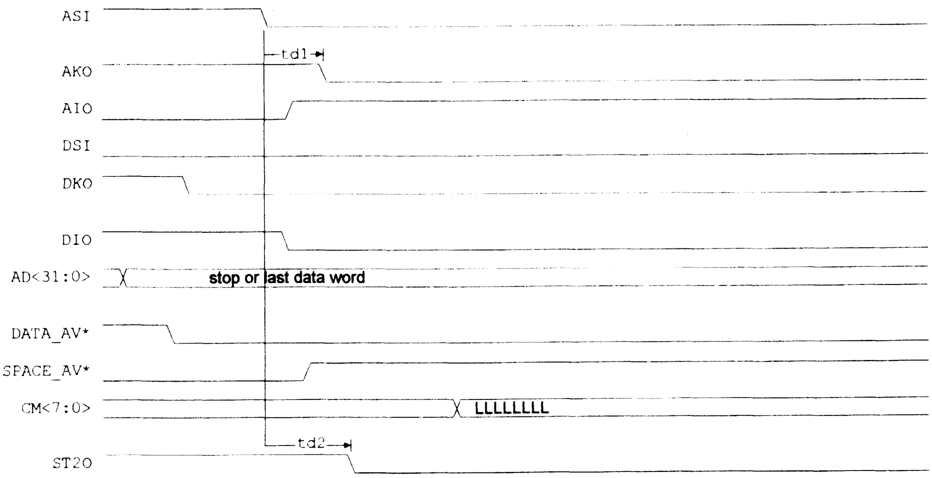


Figure D–78. FB+ Slave, Write Disconnection Phase

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↓ to AKO↓	11	21	39	ns
t_{d2}	Delay time, ASI↓ to ST20↓	18	37	57	ns

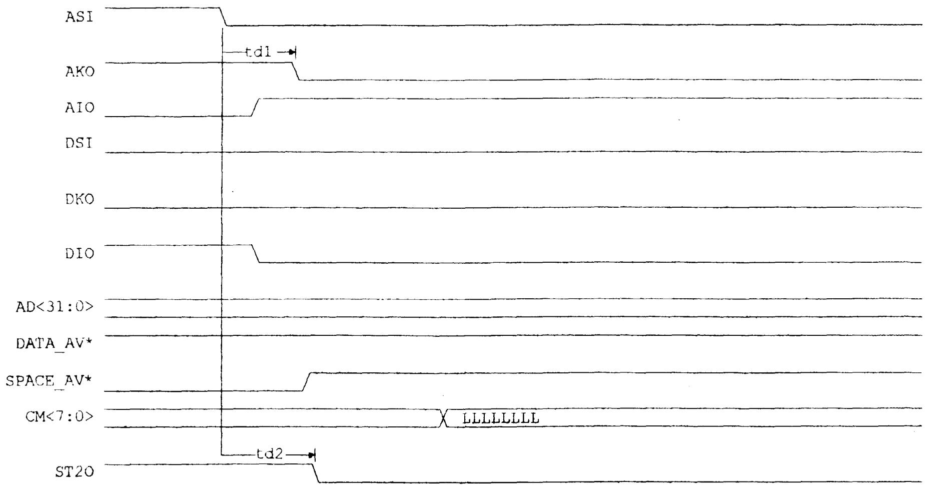


Figure D–79. FB+ Slave, Read Disconnection Phase

switching characteristics over recommended operating free-air temperature range and supply voltage ranges (unless otherwise noted)

PARAMETER		MIN	TYP	MAX	UNIT
t_{d1}	Delay time, ASI↓ to AKO↓	18	29	51	ns
t_{d2}	Delay time, ASI↓ to ST20↓	23	41	70	ns

Appendix E

TI Futurebus+ Chip Set Memory Map with Reset Values

Appendix E
TI Futurebus+ Chip Set Memory Map with Reset Values

General Information

Control and status registers (CSRs) are used to configure and report the status of the module to any software agent in the system. Any module may address any CSR, regardless of whether it is local or remote. The table in Section E.4 defines the software accessible registers located within the TI chip set and how to access and manipulate these registers.

Memory-Map Table Format

Each register entry in the table in Section E.4 contains eight fields: ADDRESS, NAME, DEVICE, TYPE, RST*, SYSRESET*, BINIT*, and description.

The ADDRESS field contains a decimal and a hexadecimal number. The decimal number has no prefix, and the prefix 0x is used to indicate a hexadecimal number. In the address field of sixty-four bit registers, two numbers are shown. The smaller address is used for bytes zero through three; the larger address is used for bytes four through seven in accordance with IEEE 896.2-1991. Only the 12 least significant bits of the address are shown in this field. The remaining 20 bits verify that the transaction addresses this node CSRs. Only when the upper 20 bits indicate that a register on this node is being accessed are any of these addresses used (i.e., 32'(address) = {4'b1111, 16'(match node ID), 12'(address)}). For details on the memory-mapping logic provided by this chip set, see Sections 2.4 and 5.2.

The NAME field contains the name of the register. The name of the each register implies the function of that register.

The DEVICE field indicates the name of the device that contains the register. The words IOC, DPU, ABC, or a combination of these names appear in this field for each register.

The TYPE field alludes to the place that the CSR is defined: core, FB+, unit-specific CSRs are defined by the 1212 and 896.2 IEEE standards. Unit-specific registers are defined for this chip set by Texas Instruments. They obey the general principles and philosophy of core and bus specific CSRs but are unique to the TI chip set.

The RST* field shows the value that would be contained in a register following the assertion (low) and release (high) of RST*. All values in this field are in hexadecimal format.

The SYSRESET* field shows the value that would be contained in a register following the assertion (low) and release (high) of SYSRESET*. All values in this field are in hexadecimal format.

The BINIT* field shows the value that would be contained in a register following the assertion (low) and release (high) of BINIT*. All values in this field are in hexadecimal format.

Each register entry contains a description of the register's operation. Core and FB+ registers are only briefly described; for complete information on these registers and how to use them, check the IEEE 896.2-1991 standard.

CSR Byte Ordering

CSR byte ordering is specified in IEEE 896.2 and presented in the first two rows of Figure E–1. For all CSRs in 896.2, when reading or writing the CSR from Futurebus+, the correspondence between Futurebus+ address/data lines and associated CSR bytes is shown in these first two rows. These two rows also show the relationship of a particular address/data line to a particular bit within a particular CSR byte. The third and fourth rows of this figure overlay the IEEE 896.2 byte ordering of the first two rows with the TI chip set host interface and CSR bus interface signal lines. This overlay approach shows the correspondence between a particular CSR bit and the associated host interface or CSR bus interface signal line.

	BYTE 0 (MSB)	BYTE 1	BYTE 2	BYTE 3 (LSB)	
CSR Bytes	BIT 7 ••• BIT 0	BIT 7 ••• BIT 0	BIT 7 ••• BIT 0	BIT 7 ••• BIT 0	In Accordance With IEEE 896.2
Futurebus+ Address/Data Lines	AD07 ••• AD00	AD15 ••• AD08	AD23 ••• AD16	AD31 ••• AD24	
	CA<1:0>=00	CA<1:0>=01	CA<1:0>=10	CA<1:0>=11	TI Chip-Set Signal Lines: CA<1:0>
CSR Bus Interface Lines	CD7 ••• CD0	CD7 ••• CD0	CD7 ••• CD0	CD7 ••• CD0	CD<7:0>
Host Interface Lines	HD07 ••• HD00	HD15 ••• HD08	HD23 ••• HD16	HD31 ••• HD24	HD<31:0>

Figure E–1. Byte Ordering and Correspondence Between CSR Bits and Associated Host Interface or CSR Bus Interface Lines

CSR Table

The following table provides an in-depth description of the TI chip set CSRs.

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
0 = clear 4 = set	STATE	IOC	CORE	00 00 00 00	00 00 00 00	SEE TEXT
0x00 0x04	If a change occurs in this register, the state register has been written bit in the status CSR is set to one. An interrupt is generated on HIF if the corresponding enable bit is set. The STATE field (bit<1:0>) resets to running (00). ELOG is reset to 0 by bus initialize (BINIT*) while the others are not changed. Bytes 0 and 1 Unit-specific. No effect on chip-set operation. Byte 2 Bus-dependent. No effect on chip-set operation. Byte 3 bit<7:5> Reserved. No effect on chip-set operation. bit<4> ELOG. The logical OR result of error-hi and error-lo CSR bits. bit<3> ATN. No effect on chip-set operation. bit<2> OFF. No effect on chip-set operation. bit<1:0> STATE. No effect on chip-set operation.					
8	NODE IDs	DPU	CORE	SEE TEXT	SEE TEXT	NO CHANGE
0x08	The bus address field is set to all ones and the priority field is reset to zeros after a power-up or system reset (RST* or SYSRESET*). There is no change to geographical address nor node side in both cases. Byte 0 and Byte 1 bit<7:6> Bus address. Byte 1 bit<5:1> Geographical address (read only). bit<0> Node side. A zero for this chip set. Byte 2 Reserved (not implemented). Byte 3 Priority.					
12	RESET START	IOC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x0C	Writing to this register causes a system reset to be performed on the HIF. The chip set remains aligned to the Futurebus+ during this operation. The corresponding bit in the status CSR is set when the task is completed, and an interrupt is generated on HIF if the interrupt bit is enabled. A global system reset overrides this local operation. The data written has no effect on the chip-set operation. Reading this register always returns zeros.					
28	SPLIT TIMEOUT LO	IOC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x1C	This is the number of fractional seconds to wait for a split response before setting the split time-out bit in the error-hi CSR. If this register contains the value zero, then the time out is disabled. Bits 5–0 of byte 0 are used to determine the split time-out count. Reading from the other bytes always returns zeros. All time delay and time-out values are based on a 40-MHz reference clock. Byte 0 bit<7> Reserved (not implemented). bit<6> Reserved (not implemented). bit<5> Msb of split time-out count [2^{-3} seconds] bit<4> Split time-out count [2^{-4} seconds] bit<3> Split time-out count [2^{-5} seconds] bit<2> Split time-out count [2^{-6} seconds] bit<1> Split time-out count [2^{-7} seconds] bit<0> Lsb of split time-out count [2^{-8} seconds (about 3.9 ms)] Bytes 1–3 Reserved (not implemented).					
40	TEST START	IOC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x28	This register is used by the module's test unit. When it is written, it sets the test start register has been written bit in the status CSR. An interrupt is generated on HIF if the corresponding enable bit is set. If the module test capability implemented bit is set in the configuration CSR, then the test status CSR's TEST_STATE field goes to the checking (10000) state; otherwise, it goes to the no test implemented (00000) state.					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
44	**TEST STATUS**	IOC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x2C	This 32-bit register contains the test status from the module's test unit. All bits in this register can be modified by the module's test unit via a CSR write. The TEST_STATE field (byte 3 bit<3:0>) can also be updated internally via a write to the test start CSR. Byte 0 bit<7:4> CAT. A 4-bit field that indicates the category of test that passed or failed when active is zero in the TEST_STATE field (see 896.2 for details). The module's test unit should write the correct value to this field when it has determined the category of the test performed. Byte 0 bit<3:0>, Byte 1, and Byte 2 bit<7:4> TEST_STEP. A 16-bit field represents either the test step running or the test step that failed as indicated in the TEST_STATE field (see 896.2 for details). The module's test unit should write the correct value to this field when it changes test step or detects a failure. Byte 2 bit<3:0>, Byte 3 bit <7:6> FRU. A 6-bit field specifies the failed component as specified by the module designer (see 896.2 for details). The module's test unit should write the correct value to this field when a failure occurs and it has determined the ID of a failed component. Byte 3 bit<5> Reserved. No effect on chip-set operation. bit<4> Active. Cleared on a write to the test start CSR if the module-test capability-implemented bit in the configuration CSR is 0 and is set on a write to the test start CSR if the module-capability implemented bit is 1. bit<3:0> TEST_STATE. The module's test unit should write the correct value to this field to provide an overview of how the test sequence is progressing or has completed.					
48, 52	**UNITS BASE**	DPU	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x30, 0x34	The base address of the local extended unit space (only the four least significant bits of register 48 are implemented). For register 52, only the 20 most significant bits and the lsb (enable bit) are implemented.					
56, 60	**UNITS BOUND**	DPU	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x38, 0x3C	The bound address of the local extended unit space (only the four least significant bits of register 56 are implemented). For register 60, only the 20 most significant bits are implemented.					
64, 68	**MEMORY BASE**	DPU	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x40, 0x44	The base address of the local memory space (only the four least significant bits of register 64 are implemented). For register 68, only the 20 most significant bits and the lsb (enable bit) are implemented.					
72, 76	**MEMORY BOUND**	DPU	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x48, 0x4C	The bound address of the local memory space (only the four least significant bits of register 72 are implemented). For register 76, only the 20 most significant bits are implemented.					
80	**INTERRUPT-TARGET CSR**	ABC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x50	When the interrupt-target CSR is written, the data is ANDed with the current value of the interrupt-mask CSR and the result is stored in the ABC-target interrupt register. Arbitration messages received with values 80–9F also set corresponding bits in the target interrupt register. Reading this register always returns zeroes. The value of the target interrupt bits can be obtained by reading the target-interrupt-clear CSR (address = 3740). The value of the highest priority target interrupt set can be obtained by reading the target-interrupt ID CSR (address = 3744).					
84	**INTERRUPT-TARGET MASK**	ABC	CORE	00 00 00 00	00 00 00 00	NO CHANGE
0x54	This CSR enables target interrupts as described above for the interrupt-target CSR. Reading this register returns the last value written.					
128 to 188	**MESSAGE-REQUEST DECODERS**	DPU	CORE			
0x80, to 0xBC	The message registers are not implemented in this device, but the message masks use these register accesses to determine if a message applies to this device. If the Futurebus+ writes to one of these registers and its mask is set in the message-passing select register, the message is transported to the host bus.					

<table>
<tr><th>ADDRESS</th><th>NAME</th><th>DEVICE</th><th>TYPE</th><th>RST*
BYTE
0 1 2 3</th><th>SYSRESET*
BYTE
0 1 2 3</th><th>BINIT*
BYTE</th></tr>
<tr><td>192 to 252</td><td>MESSAGE-RESPONSE DECODERS</td><td>DPU</td><td>CORE</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0xC0
to
0xFC</td><td colspan="6">The message registers are not implemented in this device, but the message masks use these register accesses to determine if a message applies to this device. If Futurebus+ writes to one of these registers and its mask is set in the message-passing select register, the message is transported to the host interface.</td></tr>
<tr><td>384</td><td>ERROR HI</td><td>IOC</td><td>CORE</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0x180</td><td colspan="6">The errors reported in this register generate an interrupt on the HIF if the enable bit is set in the interrupt-mask enable CSR. They can be set by writing a logic high to the corresponding bit in the error-set CSR. They can be cleared by writing a logic high to the error-hi CSR.

Byte 0
bit<7> Error summary. Set if one or more of these bits is set (error-hi CSR's byte 0 bit<2> and bit<0>, byte 1 bit<7:3>, byte 2 bit<6> and bit<0>, and error-lo CSR's byte 3 bit<7:0>).
bit<6> Master. Set if an error is detected when this chip set is a FB+ master.
bit<5> Connection phase. Set if an error is detected during connection phase.
bit<4> Data phase. Set if an error is detected during data phase.
bit<3> Disconnection phase. Set if an error is detected during disconnection phase.
bit<2> Nonexistent address. Set if SL is not asserted when AI* is released and the error-summary bit is not set.
bit<1> NU1. A user-defined error that can be set via the error-set CSR from HIF or FB+.
bit<0> Command parity error. Set if a parity error is detected on the FB+ command field during connection/data phase when the chip set is a FB+ master and the error-summary bit is not set.
Byte 1
bit<7> Parity error on FB+ address /data field. Set if a parity error is detected on FB+ address/data field and the error-summary bit is not set.
bit<6> Protocol error. Set if an unsupported command is decoded during connection phase when the chip set is a FB+-selected slave and the error-summary bit is not set.
bit<5> Transaction time out. Set if a transaction time out is detected and the error-summary bit is not set (a transaction time out also starts a bus initialize on FB+).
bit<4> Split time out. Set if a split time out is detected and the error-summary bit is not set.
bit<3> Reserved. An unit-specific error that can be set via error-set CSR from HIF or FB+.
bit<2:0> Capability. The capability field value on FB+ when an error is detected.
Byte 2 Status. The status field value on FB+ when an error is detected.
Byte 3 Command. The command field value on FB+ when an error is detected.</td></tr>
<tr><td>388</td><td>ERROR LO</td><td>IOC</td><td>CORE</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0x184</td><td colspan="6">The errors reported in this register generate an interrupt on the HIF if the enable bit is set in the interrupt-mask-enable CSR. They can be set by writing a logic high to the corresponding bit in the error-set CSR. They can be cleared by writing a logic high to the error-lo CSR.

Bytes 0 and 1 Reserved (not implemented).
Byte 2 Reserved. No effect on chip-set operation.
Byte 3
bit<7> Busy retry threshold exceeded. Set if busy retry threshold is exceeded and the error-summary bit is not set.
bit<6> Error retry threshold exceeded. No effect on chip-set operation.
bit<5> Length error. Set if the number of packet data bits received does not match the given length and the error-summary bit is not set.
bit<4> Arbitration error. Set if an arbitration error is detected and the error-summary bit is not set.
bit<3> Arbitration compare or parity error. Set if arbitration compare or parity error is detected and the error-summary bit is not set.
bit<2> Arbitration time-out error. Set if arbitration time-out error is detected during phase 2 or 4 and the error-summary bit is not set.
bit<1> Tag-parity error. No effect on chip-set operation.
bit<0> Nonexistent transaction ID. No effect on chip-set operation.</td></tr>
</table>

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
512	LOGICAL COMMON CONTROL	IOC, ABC	FB+	SEE TEXT	SEE TEXT	SEE TEXT
0x200	A 32-bit field representing the configuration of the system as defined in IEEE Standard 896.2. The bits in this CSR may be shared with more than one device. Writes may be broadcast to all devices for simultaneous configuration. The IOC maintains a copy of all the CSR bits and should respond alone to all reads of this CSR. The ABC zeros bytes 0 and 1 and maintains a copy of bytes 2 and 3 of this CSR. Only bit 5 in byte 2 (central arbiter) and bit 1 in byte 0 (distributed-message enable) are used by the ABC. Chip enables or special shared read configuration bits are provided so only one device responds during reads of this CSR. The IOC and ABC read configuration bits default to the IOC responding to reads and the ABC data lines remaining in the high-impedance state for reads of this CSR. All bits are reset to zeros after a power up, system reset, or bus initialize (RST*, SYSRESET*, or BINIT*) except for the central arbiter bit, which does not change on BINIT* and samples the PE* value when REf is released (low) in the case of power up and system reset. Bytes 0, 1 and Byte 2 bit<7:6> Reserved. No effect on chip-set operation. bit<5> Central arbiter. This bit is the locked value of PE* when REf transitions to 0 after a power up or system reset (e.g., 1 = PE* is asserted). bit<4:2> Reserved. No effect on chip-set operation. bit<1> Packet-length-16 enable. A logic high enables packet length of 16 operation. bit<0> Packet-length-8 enable. A logic high enables packet length of 8 operation. Byte 3 bit<7> Packet-length-4 enable. A logic high enables packet length of 4 operation. bit<6> Packet-length-2 enable. A logic high enables packet length of 2 operation. bit<5:2> Reserved. No effect on chip-set operation. bit<1> Distributed-message enable. A logic high enables distributed-message passing. bit<0> Split enable. Should be 0 since this chip set is not capable of initiating a split.					
516	LOGICAL MODULE CONTROL	IOC, ABC	FB+	00 00 00 00	00 00 00 00	SEE TEXT
0x204	A 32-bit field representing the configuration of the module as defined in IEEE Standard 896.2. The bits in this CSR may be shared with more than one device. Writes may be broadcast to all devices for simultaneous configuration. The IOC maintains a copy of all the CSR bits and should respond alone to all reads of this CSR. The ABC zeros bytes 0,1, and 2 and maintains a copy of byte 3 of this CSR. Only bit 0 (master enable) and bit 2 (parity-report enable) in byte 3 are used by the ABC. Chip enables or special shared read configuration bits are provided so only one device responds during reads of this CSR. The IOC and ABC read configuration bits default to the IOC responding to reads and the ABC data lines remaining in the high-impedance state for reads of this CSR. All bits are reset to zeros after a bus initialize (BINIT*) except for byte 0, 1, and bit<7:6> of byte 2, which does not change on BINIT* Byte 0 bit<7:4> Reserved. No effect on chip-set operation. bit<3> Noncache-data-width-64 enable. A logic high enables data width of 64 operation. bit<2:0> Reserved. No effect on chip-set operation. Byte 1 bit<7> Cache-data-width-64 enable. A logic high enables data width of 64 operation. bit<6:5> Reserved. No effect on chip-set operation. bit<4:0> Reserved. No effect on chip-set operation. Permitted message-transfer-data width is the same as typical data width enabled. CSR access data width is always 32 bits for this chip set. Byte 2 bit<7:4> Reserved. No effect on chip-set operation. bit<3:1> Compelled data length. No effect on chip-set operation. The information in the node capabilities ext CSR is used to determine which compelled data length to operate on. This chip set is capable of supporting all compelled data-length types. bit<0> Packet mode. A logic high enables the packet mode operation. Byte 3 bit<7> 64-bit-address enable. A logic high enables 64-bit addressing on FB+. bit<6:3> Reserved. No effect on chip-set operation. bit<2> Parity report enable. A logic high enables IOC to report parity error on FB+. bit<1> Reserved. No effect on chip-set operation. bit<0> Master enable. A logic high enables IOC to become the FB+ master.					

<table>
<tr><th>ADDRESS</th><th>NAME</th><th>DEVICE</th><th>TYPE</th><th>RST*
BYTE
0 1 2 3</th><th>SYSRESET*
BYTE
0 1 2 3</th><th>BINIT*
BYTE
0 1 2 3</th></tr>
<tr><td>520</td><td>BUS PROPAGATION DELAY</td><td>IOC</td><td>FB+</td><td>00 00 00 3F</td><td>00 00 00 3F</td><td>00 00 00 3F</td></tr>
<tr><td>0x208</td><td colspan="6">This CSR value selects a glitch-filter value according to the following table. Any glitches less than this value do not pass through the filter. This value adds to the overall delay in the system. The maximum duration of a wire-ORed-glitch is two times the electrical length of the backplane (two times backplane end-to-end propagation delay). For maximum performance, this CSR should be programmed to the shortest delay for which the system operates reliably in the presence of wire-ORed-glitches.
This CSR is calibrated in terms of the Futurebus+ backplane electrical length in the specification. This CSR defaults to 3F, which is the maximum delay.
<table>
<tr><th>REGISTER VALUE RANGE (hex)</th><th>MAXIMUM GLITCH FILTERED (ns) AT REFCLK = 33 MHz</th></tr>
<tr><td>00 – 0F</td><td>10</td></tr>
<tr><td>10 – 1F</td><td>15</td></tr>
<tr><td>20 – 2F</td><td>25</td></tr>
<tr><td>30 – 3F</td><td>30</td></tr>
</table></td></tr>
<tr><td>520</td><td>BUS PROPAGATION DELAY</td><td>DPU, ABC</td><td>FB+</td><td>00 00 00 3F</td><td>00 00 00 3F</td><td>00 00 00 3F</td></tr>
<tr><td>0x208</td><td colspan="6">This CSR value selects a glitch-filter value according to the following table. Any glitches less than this value do not pass through the filter. This value adds to the overall delay in the system. The maximum duration of a wire-ORed-glitch is two times the electrical length of the backplane (two times backplane end-to-end propagation delay). For maximum performance, this CSR should be programmed to the shortest delay for which the system operates reliably in the presence of wire-ORed-glitches.
This CSR is calibrated in terms of the Futurebus+ backplane electrical length in the specification. This CSR defaults to 3F, which is the maximum delay.
<table>
<tr><th>REGISTER VALUE RANGE (hex)</th><th>MAXIMUM GLITCH FILTERED (ns) AT ANY OTHER REFCLK FREQUENCY</th></tr>
<tr><td>00 – 0F</td><td>2.5 + 300/REFCLK frequency (MHz)</td></tr>
<tr><td>10 – 1F</td><td>2.5 + 500/REFCLK frequency (MHz)</td></tr>
<tr><td>20 – 2F</td><td>2.5 + 700/REFCLK frequency (MHz)</td></tr>
<tr><td>30 – 3F</td><td>2.5 + 900/REFCLK frequency (MHz)</td></tr>
</table></td></tr>
</table>

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE 0 1 2 3
524	**COMPETITION SETTLING TIME**	ABC	FB+	00 00 06 00	00 00 06 00	00 00 06 00
0x20C	The competition settling-time CSR is a read/write CSR that defines the settling time for the worst-case arbitration number for this module. This is a 12-bit value located in CSR bytes 2 and 3. The most significant 4 bits of byte two and bytes 1 and 0 are hard set to zero during CSR reads. This CSR should only be modified when its respective module is not competing. IEEE 896.2 defines the time stored in this CSR in increments of seconds. The ABC uses REFCLK as a time base for a counter to implement the time delays as programmed in this CSR. See the arbitration settling times section in Chapter 6 for more detailed programming information.					
528	**TRANSACTION TIME-OUT**	IOC	FB+	00 08 00 00	00 08 00 00	00 08 00 00
0x210	This register contains the time to wait before a transaction time out occurs. When a transaction lasts longer than this value, the transaction time-out bit in the error-hi CSR is set. It defaults to 122 μs and is reset to 122 μs when a bus initialize (BINIT*) occurs. A value of zero means time out is disabled. Bits <4:1> of byte 1 are implemented. All other bits return zeros when read. All time delay and time-out values are based on a 40-MHz reference clock. Byte 0 Reserved (not implemented). Byte 1 bit<7> Reserved (not implemented). bit<6> Reserved (not implemented). bit<5> Reserved (not implemented). bit<4> Msb of transaction time-out count [2^{-12} seconds] bit<3> Transaction time-out count [2^{-13} seconds] bit<2> Transaction time-out count [2^{-14} seconds] bit<1> Lsb of transaction time-out count [2^{-15} seconds (about 30.5 μs)] bit<0> Reserved (not implemented). Bytes 2 and 3 Reserved (not implemented).					
532, 536	**MESSAGE-PASSING SELECT MASK**	DPU	FB+	00 00 00 00	00 00 00 00	NO CHANGE
0x214, 0x218	This register determines which message-passing transactions are transported to the host bus.					
540	**BUSY-RETRY COUNTER**	IOC	FB+	00 00 00 00	00 00 00 00	NO CHANGE
0x21C	This register contains the busy-retry threshold and count values. Bytes 0 and 1 Busy-retry threshold. Bytes 2 and 3 Busy-retry counter.					
544	**BUSY-RETRY DELAY**	IOC	FB+	00 00 00 00	00 00 00 00	NO CHANGE
0x220	This register contains the time to wait before the IOC retries a busied transaction. It defaults to 0 when a reset occurs. The delay continues its count during bus initialize. It resets for a reset start. A value of 0 means disable time out; 13 bits are implemented. All other bits return zeros when read. All time delay and time-out values are based on a 40-MHz reference clock. Byte 0 bit<7> Reserved (not implemented). bit<6> Reserved (not implemented). bit<5> Reserved (not implemented). bit<4> Reserved (not implemented). bit<3> Msb of busy-retry delay [2^{-5} seconds] bit<2> Busy-retry delay [2^{-6} seconds] bit<1> Busy-retry delay [2^{-7} seconds] bit<0> Busy-retry delay [2^{-8} seconds] Byte 1 bit<7> Busy-retry delay [2^{-9} seconds] bit<6> Busy-retry delay [2^{-10} seconds] bit<5> Busy-retry delay [2^{-11} seconds] bit<4> Busy-retry delay [2^{-12} seconds] bit<3> Busy-retry delay [2^{-13} seconds] bit<2> Busy-retry delay [2^{-14} seconds] bit<1> Busy-retry delay [2^{-15} seconds] bit<0> Busy-retry delay [2^{-16} seconds] Byte 2 bit<7> Lsb of busy-retry delay [2^{-17} seconds (about 7.6 μs)] bit<6:0> Reserved (not implemented). Byte 3 Reserved (not implemented).					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT*
3712	**RQ0 PRIORITY**	**ABC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xE80	This read/write CSR is used to program the 8-bit priority used for bus arbitration when RQ0 is asserted. It is located in byte 3. Bytes 0, 1, and 2 are hard set to zero during reads. In the distributed-arbitration mode, the contents of this CSR are used directly to formulate the arbitration competition number for request level 0 (RQ0). In the central mode, writing this CSR causes the ABC to send a central arbitrated message to the central arbiter. This message programs the central bus arbiter with the new priority.					
3716	**RQ1 PRIORITY**	**ABC**	**Unit-specific [read or write]**	**00 00 00 80**	**00 00 00 80**	**NO CHANGE**
0xE84	This read/write CSR is used to program the 8-bit priority used for bus arbitration when RQ1 is asserted. It is located in byte 3. Bytes 0, 1, and 2 are hard set to zero during reads. In the distributed-arbitration mode, the contents of this CSR are used directly to formulate the arbitration competition number for request level 1 (RQ1). In the central mode, writing this CSR causes the ABC to send a central arbitrated message to the central arbiter. This message programs the central bus arbiter with the new priority.					
3720	**SEND-ARBITRATION MESSAGE/ MESSAGE FIFO**	**ABC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xE88	The send-arbitration message/message-FIFO address utilizes only byte 3. Bytes 0, 1, and 2 are hard set to zero during reads. Writing this CSR causes the ABC to send bits <6:0> of byte 3 as an arbitrated message (arbitrated messages hard set bit 7 to a 1). The message-sent interrupt is generated when the message has been sent. Reading byte 3 retrieves received messages from the ABC four message FIFO buffer. Consecutive reads present messages in the order they are received. If the FIFO is empty, the last value read is returned. The receive-FIFO-not-empty (RFNE) interrupt is generated any time there is a message available in the FIFO. See Chapter 6 for details on which arbitrated messages are stored in the FIFO.					
3724	**INTERRUPT STATUS CLEAR**	**ABC**	**Unit-specific [read or write 1 to clear]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xE8C	Writing a one to this address clears the corresponding interrupt bit. Writing a zero has no effect. Reading bytes 2 and 3 of this CSR returns the status of the interrupt bits as shown below. Bytes 0 and 1 are unused and are hard set to zero during reads. Following a clear, RFNE and target int interrupts generate an interrupt in this CSR any time the FIFO contains any data or a target interrupt is active respectively. All other interrupt bits require the condition causing the interrupt to be released and then occur again to cause another interrupt. Bytes 0 and 1 Not used Byte 2 NO. Status Interrupt bit<7> 16 PFAIL. Power-fail message received. bit<6> 15 ARBERR. An arbitration error occurred during Futurebus+ competition. bit<5> 14 CMPPARERR. A compare or parity error occurred during Futurebus+ competition. bit<4> 13 PHS24TO. Phase 2 or 4 1-µs time out occurred. bit<3> 12 DMTOUT. Dead-man time-out occurred. This indicates that phase 1 or phase 0 for a two-pass competition exceeded the time-out value. bit<2> 11 CSR_PAR_ERR. A CSR parity error has been detected. bit<1> 10 RE. Interrupt if RE* is asserted. bit<0> 9 AMOV. Arbitration message FIFO overflow has occurred. Byte 3 bit<7> 8 Reserved bit<6> 7 TARGET_INT. A target interrupt has been received. bit<5:4> 6,5 Reserved bit<3> 4 GRANT. Bus has been granted (GR asserted). Informs software when the priority registers may be updated without affecting an ongoing bus acquisition. bit<2> 3 RFNE. The arbitration-message-receive FIFO is not empty. bit<1> 2 MSG_SENT. The arbitration message has been sent. bit<0> 1 PRI_UPD. The arbitration priority has been updated. Writing a one to a bit in this register clears the bit until the interrupt condition is released and then recurs.					

<table>
<tr><th>ADDRESS</th><th>NAME</th><th>DEVICE</th><th>TYPE</th><th>RST*
BYTE
0 1 2 3</th><th>SYSRESET*
BYTE
0 1 2 3</th><th>BINIT*</th></tr>
<tr><td>3728</td><td>INTERRUPT STATUS SET</td><td>ABC</td><td>Unit-specific
[read or write
1 to set]</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0xE90</td><td colspan="6">Writing a one to this CSR sets the corresponding interrupt bit. Writing a zero has no effect. Reading bytes 2 and 3 of this CSR returns the status of the interrupt bits as shown for the interrupt-status-clear CSR. Bytes 0 and 1 are unused and are hard set to zero during reads. Following a clear, RFNE and target int interrupts generate an interrupt in this CSR any time the FIFO contains any data or a target interrupt is active respectively. All other interrupt bits require the condition causing the interrupt to be released and then occur again to cause another interrupt. Bits 7, 5, and 4 in byte 3 are never set by internal ABC conditions; however, they may be set and cleared by software using the interrupt-status-set and clear CSRs and their status is reflected when byte 3 is read.
The interrupt-status set bit descriptions are identical to the bit descriptions for the interrupt-status clear CSR.</td></tr>
<tr><td>3732</td><td>INTERRUPT ENABLE</td><td>ABC</td><td>Unit-specific
[read or
write]</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0xE94</td><td colspan="6">This register is used to enable the interrupt conditions in the interrupt status register. When an interrupt is enabled and the interrupt condition is set, INT is asserted to the host module, and the ID of the interrupt condition is reported in the interrupt-ID register.</td></tr>
<tr><td>3736</td><td>INTERRUPT ID</td><td>ABC</td><td>Unit-specific
[read only]</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0xE98</td><td colspan="6">This CSR contains a five-bit value representing the priority of the highest priority interrupt that is currently set and enabled in the ABC-interrupt status register. This value may be used for interrupt service routine vectoring to minimize software decoding of interrupt priorities. The next highest priority enabled interrupt is seen when the current interrupt is cleared. This CSR is read only and the bits are located in byte 3, bits <4:0>. Bits <7:5> are unused and are hard set to zero. Bytes 0, 1, and 2 are unused and are also hard set to zero.
Bytes 0, 1, and 2 Not used
Byte 3
bit<4:0> Indication
Value (see interrupt-status clear)
16 No. 16 Status interrupt is set
15 No. 15 Status interrupt is set
• •
• •
• •
1 No. 1 Status interrupt is set
0 No status interrupts are set</td></tr>
<tr><td>3740</td><td>TARGETED INTERRUPT CLEAR</td><td>ABC</td><td>Unit-specific
[read or write
1 to clear]</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO CHANGE</td></tr>
<tr><td>0xE9C</td><td colspan="6">This CSR is used to observe and clear the 32 targeted interrupts which may have been set by writing to the interrupt-target CSR address 80 or by arbitrated messages 80–9F. Writing a logic high to a bit in this CSR causes the corresponding targeted interrupt to be cleared. Reading this CSR returns the value of the ABC-targeted interrupt register. See the targeted-interrupt section in Chapter 6 for more details. The table below shows the targeted-interrupt priority in decimal and the arbitrated-message value received in hex for each targeted-interrupt bit formatted as priority/message value.
<table>
<tr><th>CSR Data Bit</th><th>Byte 0</th><th>Byte 1</th><th>Byte 2</th><th>Byte 3</th></tr>
<tr><td>bit<7></td><td>32/9F</td><td>24/97</td><td>16/8F</td><td>8/87</td></tr>
<tr><td>bit<6></td><td>31/9E</td><td>23/96</td><td>15/8E</td><td>7/86</td></tr>
<tr><td>bit<5></td><td>30/9D</td><td>22/95</td><td>14/8D</td><td>6/85</td></tr>
<tr><td>bit<4></td><td>29/9C</td><td>21/94</td><td>13/8C</td><td>5/84</td></tr>
<tr><td>bit<3></td><td>28/9B</td><td>20/93</td><td>12/8B</td><td>4/83</td></tr>
<tr><td>bit<2></td><td>27/9A</td><td>19/92</td><td>11/8A</td><td>3/82</td></tr>
<tr><td>bit<1></td><td>26/99</td><td>18/91</td><td>10/89</td><td>2/81</td></tr>
<tr><td>bit<0></td><td>258/98</td><td>17/90</td><td>9/88</td><td>1/80</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>ADDRESS</th><th>NAME</th><th>DEVICE</th><th>TYPE</th><th>RST*
BYTE
0 1 2 3</th><th>SYSRESET*
BYTE
0 1 2 3</th><th>BINIT*</th></tr>
<tr><td>3744</td><td>TARGETED-INTERRUPT ID</td><td>ABC</td><td>Unit-specific
[read only]</td><td>00 00 00 00</td><td>00 00 00 00</td><td>NO
CHANGE</td></tr>
<tr><td>0xEA0</td><td colspan="6">This CSR contains a 5-bit value representing the priority of the highest priority targeted interrupt that is currently set in the ABC-targeted interrupt register. This value may be used for targeted interrupt service routine vectoring to minimize software decoding of targeted interrupt priorities. The next highest priority targeted interrupt is seen when the current one is cleared. This CSR is read only and the bits are located in byte 3, bits <6:0>. Bit 7 is unused and is hard set to zero. Bytes 0, 1, and 2 are unused and are also hard set to zero.

Bytes 0, 1, and 2 Not used
Byte 3
bit<6:0> Indication
Value
32 Target interrupt 32 is set
31 Target interrupt 31 is set
• •
• •
• •
1 Target interrupt 1 is set
0 No target interrupts are set

An interrupt ID of 0 indicates that no targeted interrupts are currently set. The 32 targeted interrupts are assigned priorities of 1 to 32 where 1 is the lowest and 32 is the highest priority.</td></tr>
<tr><td>3748</td><td>ABC CONFIGURATION</td><td>ABC</td><td>Unit-specific
[read or write]</td><td>00 00 00 00</td><td></td><td>NO
CHANGE</td></tr>
<tr><td>0xEA4</td><td colspan="6">Byte 0 Not used
Byte 1
bit<7:1> Not used
bit<0> (PCBA only) Enable central arbiter. When set, this module performs central arbitration for the system. This overrides the CENTRAL_EN terminal that was used at power up to tell this device that it is the power-up central arbiter.
Byte 2
bit<6> Only master enable. Setting this bit informs the ABC that it is the only potential master in a distributed arbitration system. The arbitration process is bypassed and GR is asserted immediately in response to RQ1 or RQ0.
bit<5> Bypass glitch filter
bit<4> 30M/40M = 0: 40-MHz to 50-MHz reference clock used
30M/40M = 1: 30-MHz to 40-MHz reference clock used
bit<3> Enable standard CSR reads. When this bit is released, reads of the logical module control, logical common control, and bus propagation delay CSRs are ignored (the CSR data bus is 3-stated), while all other CSR reads operate normally. Reads of these CSR registers are typically handled by another device on the module since, so few bits are implemented in this device.
bit<2> Enable competition-compare errors
bit<1> CSR bus-parity mode
= 0: odd parity
= 1: even parity
bit<0> Enable CSR bus-parity error reporting
= 0: disable
= 1: check
Byte 3
bit<7> Profile-B encoding of power fail. If zero, power fail = arbitration message FF.
bit<6> Enable FIFO overflow reporting to Futurebus+ (via ac0 error). Set ac0 error when this bit is set and FIFO overflow would result if present arbitration cycle ended normally. If this bit is released, FIFO overflow occurs. Stall (bit<5> assertion) overrides this operation.
bit<5> Stall. Remain in phase 3 of the second pass of an arbitration message cycle if the receive FIFO is full.
bit<4> Preemption mode (distributed mode only). If this bit value is 0, PE is set when another module of equal or higher priority requests the bus. If this bit value is 1, PE is set only when a higher-priority module requests the bus.
bit<3> Preemption enable. Enable ABC to preempt a master elect during distributed arbitration. Note that arbitration messages preempt a master elect regardless of this bit.
bit<2> Priority only:
= 0: round robin
= 1: priority only
bit<1> Enable dead-man timer. If enabled, each phase of arbitration is checked to ensure that it does not exceed 1.31 ms (phase 0 is checked only between the first and second passes of two-pass arbitration).
bit<0> Ignore duplicate messages:
= 0: store all incoming messages in the FIFO
= 1: ignore received messages that are already in the message FIFO</td></tr>
</table>

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT*
3752	**MESSAGE CONTROL**	**ABC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xEA8	This register contains the message-mask values and enable parameters. Refer to the arbitration message section in Chapter 6. Byte 0 Message-mask enable 0 Byte 1 Message-mask value 0 Byte 2 Message-mask enable 1 Byte 3 Message-mask value 1					
3756	**OPTIONAL ARBITRATION SETTLING TIMES**	**ABC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xEAC	If any byte in this register is zero, the corresponding group uses the value in the arbitration settling-time register (524). Each byte represents a delay in 2^{-28} seconds. Ni refers to the maximum number of bus iterations required by the arbitration number. An Ni of zero means that the device knows whether it is the winner as soon as its number has propagated down the bus and back. Ni = 4 may require four iterations of the bus delay plus the initial bus delay. The Ni number is equal to the number of 1-to-0 transitions there are in the outgoing competition number. These values are loaded during initialization and must not be changed during an arbitration sequence. This register is cleared when either RST* or BINIT* is asserted. Refer to the section on arbitration settling times in Chapter 6. Byte 0 for Ni = 0 or 1 Byte 1 for Ni = 2 Byte 2 for Ni = 3 Byte 3 for Ni = 4					
3760	**MASTER ID**	**ABC**	**Unit-specific [read only]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xEB0	Bytes 0 and 1 Current master's ID and priority Bytes 2 and 3 Master elect's ID and priority					
3764	**ARBITRATION BUS MONITOR**	**ABC**	**Unit-specific [read only]**	**NO CHANGE**	**NO CHANGE**	**NO CHANGE**
0xEB4	Byte 0 Not used Byte 1 bit<7:5> Not used bit<4:0> GA<4:0> Byte 2 bit<7> AC0 bit<6> AC1 bit<5> AR bit<4> AQ bit<3> AP bit<2> WIN bit<1> Not used bit<0> ABP Byte 3 bit<7:0> AB<7:0>					
3768	**TEST PORT**	**ABC**	**Unit-specific [read only]**	**NOT SPECIFIED**	**NOT SPECIFIED**	**NOT SPECIFIED**
0xEB8	Silicon testing, not intended for customer use					
3772	**TEST PORT**	**ABC**	**Unit-specific [read only]**	**NOT SPECIFIED**	**NOT SPECIFIED**	**NOT SPECIFIED**
0xEBC	Silicon testing, not intended for customer use					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
3908	**DPU CONFIGURATION**	**DPU**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xF44	This register is used to configure the DPU. Byte 0, 1 and Byte 2 bit<7:4> Reserved. No effect on chip-set operation. bit<3> DPU CSR decode disable. A logic high disables the DPU to decode an internal CSR address. This allows the ABC/IOC to decode a shared CSR address (e.g., bus propagation delay CSR). bit<2:0> Not used. No effect on chip-set operation. Byte 3 bit<7:4> Reserved. No effect on chip-set operation. bit<3> 36-bit address mode. A logic high enables the 36-bit address mode. bit<2> HIF-parity generate enable. A logic high enables DPU to ignore incoming HIF parity and generate new parity to be sent to FB+. Default is to generate parity and compare to incoming bits, and an error is reported if there is a mismatch. Incoming HIF parity bits are passed through to FB+. bit<1> FB+ parity disable. A logic high disables the DPU to check incoming FB+ data parity. bit<0> Glitch filter bypass. A logic high causes the DPU to bypass the glitch filter.					
3928	**STATUS**	**DPU**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xF58	Writing a logic high clears the bit. Bytes 0, 1, 2 Reserved (not implemented). Byte 3 bit<7:5> Reserved (not implemented). bit<4> Broadcast mailbox accessed. Set if a broadcast message address is decoded on FB+. bit<3> Message mailbox accessed. Set if a message mailbox address is decoded on FB+. bit<2> HIF data parity error. Set if a parity error is detected on HIF data field and the HIF parity generate enable bit is not set. bit<1> FB+ packet longitudinal parity error. Set if a longitudinal parity error is detected on an incoming FB+ packet data field and the FB+ parity disable bit is not set. bit<0> FB+ address/data parity error. Set if a parity error is detected on an incoming FB+ address/data field and the FB+ parity disable bit is not set.					
3932	**TEST PORT**	**DPU**	**(Do not access)**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xF5C	Silicon testing only, not intended for customer use					
0xF5C	Byte 3 bit<3> If set, REFCLK is used for the packet transmitter. Otherwise, the internal locked loop clock is used.					
4032	**IOC RESET CONTROL**	**IOC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFC0	Byte 0 bit<7:4> Reserved (not implemented). bit<3> Alignment wait. Writing a logic high to this bit inhibits the IOC from automatic alignment to FB+ following the release of RST*. Since this bit is initially reset by RST*, a 1 must be written within 160 ms of the release of RST*; otherwise, the autoalignment is performed. The user is also responsible for resetting this bit to resume the normal flow. bit<2> System reset. Writing a logic high to this bit instructs the IOC to perform a system reset on FB+. This bit is self-cleared upon the completion of a system reset. bit<1> Bus initialize. Writing a logic high to this bit instructs the IOC to perform a bus initialize on FB+. This bit is self-cleared upon the completion of a bus initialize. bit<0> Alignment. Writing a logic high to this bit instructs the IOC to perform an alignment on FB+. The chip set participates in FB+ activities once it is aligned. Reset is not required. Bytes 1, 2, and 3 Reserved (not implemented).					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
4036	**IOC CONFIGURATION**	**IOC**	**Unit-specific [read or write]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFC4	This register is used to configure the IOC. Byte 0 bit<7> Critical-word-first disable. A logic high disables the critical word first feature. bit<6> Module test capability implemented. If this bit is set, the TEST_STATE field in the test-status CSR goes to checking when the test-start CSR is written. bit<5> 64-bit HIF enable. A logic high enables 64-bit-data-length mode on the HIF. Default is 32-bit mode. bit<4> Glitch-filter bypass. A logic high causes the IOC to bypass the glitch filter. bit<3:2> Reserved. No effect on chip-set operation. bit<1> Slow burst mode enable. A logic high enables slow burst mode on the HIF. Default is disable to optimize system performance. bit<0> Reserved. No effect on chip-set operation. Byte 1, 2, and 3 Reserved (not implemented).					
4040	**IOC STATUS CLEAR**	**IOC**	**Unit-specific [read or write 1 to clear]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFC8	Writing a logic high clears the status bit. The status reported in this register generates an interrupt on the HIF if the enable bit is set in the interrupt mask-enable CSR. Byte 0, 1 Reserved (not implemented). Byte 2 bit<7:2> Reserved (not implemented). bit<1> Alignment has occurred. bit<0> Transaction flag error. Byte 3 bit<7> State register has been written. bit<6> Reset start register has been written. bit<5> FB+ wait detected. bit<4> Locked command received. bit<3> Message mailbox accessed. bit<2> Host-bus parity error. bit<1> Test start register has been written. bit<0> Split response received.					
4044	**IOC STATUS SET**	**IOC**	**Unit-specific [read or write1 to set]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFCC	Same as the status-clear CSR. Writing a logic high sets the status bit.					
4048	**ERROR SET**	**IOC**	**Unit-specific [read or write1 to set]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFD0	Writing a logic high sets the corresponding bit in the error-hi and error-lo CSRs. Reading from this register returns the value in the corresponding location. Byte 0 Byte 0 of the error-hi CSR. Byte 1 Byte 1 of the error-hi CSR. Byte 2 Byte 2 of the error-lo CSR. Byte 3 Byte 3 of the error-lo CSR.					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT* BYTE
4052	**INTERRUPT MASK ENABLE**	**IOC**	**Unit-specific [read or write 1 to clear]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFD4	When a bit is set, the corresponding interrupt-status condition results in an interrupt being generated. When the bit is reset, an interrupt is not generated. Byte 0 bit<7> Nonexistent address bit<6> NU1. A user-defined error bit<5> Command parity error bit<4> Parity error on FB+ address/data field bit<3> Protocol error bit<2> Transaction time out bit<1> Split time out bit<0> Reserved. A unit-specific error Byte 1 bit<7> Busy-retry threshold exceeded bit<6> Reserved (error-retry threshold exceeded) bit<5> Packet-length error bit<4> Arbitration error bit<3> Arbitration compare or parity error bit<2> Arbitration time-out error bit<1> Transaction flag error bit<0> Reserved (nonexistent transaction ID) Byte 2 bit<7> State register has been written bit<6> Reset start register has been written bit<5> FB+ wait detected bit<4> Locked command received bit<3> Message mailbox accessed bit<2> Host-bus parity error bit<1> Test-start register has been written bit<0> Split response received Byte 3 Reserved (not implemented)					
4056	**INTERRUPT ID**	**IOC**	**Unit-specific [read only]**	**00 00 00 00**	**00 00 00 00**	**NO CHANGE**
0xFD8	This 6-bit field contains a number representing the highest priority interrupt that has been set and is enabled. This may correspond to a vector ID in the interrupt-service routine. This value is located in byte 3, bit<5:0>. The interrupt ID values are listed below. Byte 0, 1, and 2 Reserved (not implemented). Byte 3 bit<7:6> Reserved (not implemented). bit<5:0> Interrupt ID value Binary Decimal Description 11000 24 Nonexistent address 10111 23 NU1. A user-defined error 10110 22 Command parity error 10101 21 Parity error on FB+ address/data field 10100 20 Protocol error 10011 19 Transaction time out 10010 18 Split time out 10001 17 Reserved. A unit-specific error 10000 16 Busy-retry threshold exceeded 01111 15 Reserved (error-retry threshold exceeded) 01110 14 Packet-length error 01101 13 Arbitration error 01100 12 Arbitration compare or parity error 01011 11 Arbitration time-out error 01010 10 Transaction flag error 01001 9 Reserved (nonexistent transaction ID) 01000 8 State register has been written 00111 7 Reset start register has been written 00110 6 FB+ wait detected 00101 5 Locked command received 00100 4 Message mailbox accessed 00011 3 Host-bus parity error 00010 2 Test-start register has been written 00001 1 Split response received 00000 0 No interrupt pending					

ADDRESS	NAME	DEVICE	TYPE	RST* BYTE 0 1 2 3	SYSRESET* BYTE 0 1 2 3	BINIT BYTE
4060	**LOCKED COMMAND EXTENSION**	IOC	Unit-specific [read only]	00 00 00 00	00 00 00 00	NO CHANGE
0xFDC	The incoming locked command (LC<2:0>) is stored in byte 2, bits <2:0> of this CSR. Terminals 189, 188, and 187 also contain LC<2:0> during a locked operation. These terminals are bidirectional with internal pulldowns and are logically ORed with this CSR byte 3, bits<2:0> in the event of a mastered FB+ locked operation. Bytes 0 and 1 Reserved (test port). Byte 2 bit<7:3> Not used. No effect on chip-set operation. bit<2:0> Locked field. Asserted on the HIF when the chip set is a master of a locked bit operation. Byte 3 bit<7:3> Not used. No effect on chip-set operation. bit<2:0>> LC2–LC0. When LK* is asserted and while the chip set is an HIF slave, these bits are logically ORed with LKFLD2–0 to determine the locked command to use on FB+.					
4080	**BUSY RETRY DELAY TIMER**	IOC	Unit-specific [read or write]	00 00 00 00	00 00 00 00	NO CHANGE
0xFF0	Byte 0 and 1 Busy-retry delay timer (byte 1, 0 = timer<15:8, 7:0>). Byte 2 bit<7:5> Not used. No effect on chip-set operation. bit<4:0> Busy retry delay timer (timer<20:16>). Byte 3 Test port. Read only bit<7> AKI. bit<6> AKIf. bit<5> AII. bit<4> AIIf. bit<3> DKI. bit<2> DKIf. bit<1> DII. bit<0> DIIf.					
4084	**SPLIT RESPONSE TIMER**	IOC	Unit-specific [read or write]	00 00 00 00	00 00 00 00	NO CHANGE
0xFF4	Byte 0 and 1 Split-response timer (byte 1, 0 = timer<15:8, 7:0>). Byte 2 bit<7> Not used. No effect on chip-set operation. bit<6:0> Busy-retry delay timer (timer<22:16>). Byte 3 Test port. Read only bit<7> AQI. bit<6> AQIf. bit<5> REI. bit<4> REIf. bit<3:0> Not used. No effect on chip-set operation.					
4088	**RESET TIMER**	IOC	Unit-specific [read or write]	00 00 00 00	00 00 00 00	NO CHANGE
0xFF8	Bytes 0, 1, 2, and 3 Reset timer (byte 3, 2, 1, 0 = timer<31:24, 23:16, 15:8, 7:0>).					
4092	**TRANSACTION TIMER**	IOC	Unit-specific [read or write]	00 00 00 80	00 00 00 80	NO CHANGE
0xFFC	Byte 0 Transaction timer (timer<7:0>). Byte 1 bit<7:6> Not used. No effect on chip-set operation. bit<5:0> Transaction timer (timer<13:8>). Byte 2 Reserved (test port). Byte 3 bit<7> CSR output enable. A logic high enables the IOC to drive the CSR bus data lines when selected. A 0 disables the IOC to drive the CSR bus data lines even when it is selected. bit<6> Packet-data-width 32. When this bit is set and the packet mode is enabled, the chip set only masters 32-bit-wide data on FB+. bit<5> Write-compelled data length 64. When MORE* is asserted on HIF during a write, the chip set asserts DL64 during the data phase of a FB+ compelled transaction. bit<4:0> Reserved (test port)					

TI Worldwide Sales Offices

ALABAMA: Huntsville: 4960 Corporate Drive, Suite 150, Huntsville, AL 35805, (205) 837-7530.

ARIZONA: Phoenix: 8825 N. 23rd Avenue, Suite 100, Phoenix, AZ 85021, (602) 995-1007.

CALIFORNIA: Irvine: 1920 Main Street, Suite 900, Irvine, CA 92714, (714) 660-1200; **San Diego:** 5625 Ruffin Road, Suite 100, San Diego, CA 92123, (619) 278-9600; **Santa Clara:** 5353 Betsy Ross Drive, Santa Clara, CA 95054, (408) 980-9000; **Woodland Hills:** 21550 Oxnard Street, Suite 700, Woodland Hills, CA 91367, (818) 704-8100.

COLORADO: Aurora: 1400 S. Potomac Street, Suite 101, Aurora, CO 80012, (303) 368-8000.

CONNECTICUT: Wallingford: 9 Barnes Industrial Park South, Wallingford, CT 06492, (203) 269-0074.

FLORIDA: Altamonte Springs: 370 S. North Lake Boulevard, Suite 1008, Altamonte Springs, FL 32701, (407) 260-2116. **Fort Lauderdale:** 2950 N.W. 62nd Street, Suite 100, Fort Lauderdale, FL 33309, (305) 973-8502; **Tampa:** 4803 George Road, Suite 390, Tampa, FL 33634-6234, (813) 885-7588. ·

GEORGIA: Norcross: 5515 Spalding Drive, Norcross, GA 30092-2560, (404) 662-7967.

ILLINOIS: Arlington Heights: 515 West Algonquin, Arlington Heights, IL 60005, (708) 640-2925.

INDIANA: Carmel: 550 Congressional Drive, Suite 100, Carmel, IN 46032, (317) 573-6400; **Fort Wayne:** 103 Airport North Office Park, Fort Wayne, IN 46825, (219) 489-4697.

KANSAS: Overland Park: 7300 College Boulevard, Lighton Plaza, Suite 150, Overland Park, KS 66210, (913) 451-4511.

MARYLAND: Columbia: 8815 Centre Park Drive, Suite 100, Columbia, MD 21045, (410) 964-2003.

MASSACHUSETTS: Waltham: Bay Colony Corporate Center, 950 Winter Street, Suite 2800, Waltham, MA 02154, (617) 895-9100.

MICHIGAN: Farmington Hills: 33737 W. 12 Mile Road, Farmington Hills, MI 48018, (313) 553-1581;

MINNESOTA: Eden Prairie: 11000 W. 78th Street, Suite 100, Eden Prairie, MN 55344, (612) 828-9300.

MISSOURI: St Louis: 12412 Powerscourt Drive, Suite 125, St. Louis, MO 63131, (314) 821-8400.

NEW JERSEY: Iselin: Metropolitan Corporate Plaza, 485 Bldg. E. U.S. 1 South, Iselin, NJ 08830, (908) 750-1050.

NEW MEXICO: Albuquerque: 2709 J. Pan American Freeway, N.E., Albuquerque, NM 87101, (505) 345-2555.

NEW YORK: East Syracuse: 6365 Collamer Drive, East Syracuse, NY 13057, (315) 463-9291; **Fishkill:** 300 Westage Business Center, Suite 140, Fishkill, NY 12524, (914) 897-2900; **Melville:** 48 South Service Road, Suite 100, Melville, NY 11747, (516) 454-6601; **Pittsford:** 2851 Clover Street, Pittsford, NY 14534, (716) 385-6770.

NORTH CAROLINA: Charlotte: 8 Woodlawn Green, Charlotte, NC 28217, (704) 527-0930; **Raleigh:** 2809 Highwoods Boulevard, Suite 100, Raleigh, NC 27625, (919) 876-2725.

OHIO: Beachwood: 23775 Commerce Park Road, Beachwood, OH 44122-5875, (216) 765-7528; **Beavercreek:** 4200 Colonel Glenn Highway, Suite 600, Beavercreek, OH 45431, (513) 427-6200.

OREGON: Beaverton: 6700 S.W. 105th Street, Suite 110, Beaverton, OR97005, (503) 643-6758.

PENNSYLVANIA: Blue Bell: 670 Sentry Parkway, Suite 200, Blue Bell, PA 19422, (215) 825-9500.

PUERTO RICO: Hato Rey: 615 Mercantil Plaza Building, Suite 505, Hato Rey, PR 00919, (809) 753-8700.

TEXAS: Austin: 12501 Research Boulevard, Austin, TX 78759, (512) 250-6769; **Dallas:** 7839 Churchill Way, Dallas, TX 75251, (214) 917-1264; **Houston:** 9301 Southwest Freeway, Commerce Park, Suite 360, Houston, TX 77074, (713) 778-6592; **Midland:** FM1788 & I-20, Midland, TX 79711-0448, (915) 561-7137.

UTAH: Salt Lake City: 2180 South 1300 East, Suite 335, Salt Lake City, UT 54106, (801) 466-8972.

WISCONSIN: Waukesha: 20825 Swenson Drive, Suite 900, Waukesha WI 53186, (414) 798-1001.

CANADA: Nepean: 301 Moodie Drive, Suite 102, Mallom Center, Nepean, Ontario, Canada K2H 9C4, (613) 726-1970; **Richmond Hill:** 280 Centre Street East, Richmond Hill, Ontario, Canada, L4C 1B1, (416) 884-9181; **St. Laurent:** 9460 Trans Canada Highway, St. Laurent, Quebec, Canada, H4S 1R7, (514) 335-8392.

AUSTRALIA (& NEW ZEALAND): Texas Instruments Australia Ltd., 6-10 Talavera Road, North Ryde (Sydney), New South Wales, Australia 2113, 2-878-9000; 14th Floor, 380 Street, Kilda Road, Melbourne, Victoria, Australia 3004, 3-696-1211; 171 Philip Highway, Elizabeth, South Australia 5112, 8 255-2066.

BELGIUM: Texas Instruments Belgium S.A./N.V., Avenue Jules Bordetlaan 11, 1140 Bruxelles/Brussel, Belgium, (02) 242 30 80.

BRAZIL: Texas Instuments Electronicos do Brasil Ltda., Av. Eng. Luiz Carlos Berrini, 1461-11o, andar, 04571, Sao Paulo, SP, Brazil, 11-535-5133

DENMARK: Texas Instruments A/S, Borupvang 2D, DK-2750 Ballerup, Denmark, (44) 68 74 00.

FINLAND: Texas Instruments OY, Tekniikantie 12 02150 Espoo, Finland, (0) 43 54 20 33.

FRANCE: Texas Instruments France, 8-10 Avenue Morane-Saulnier, B.P. 67, 78141 Velizy-Villacoublay Cedex, France, (1) 30 70 10 01.

GERMANY: Texas Instruments Deutschland GmbH., Haggertystrasse 1, 8050 Freising, (08161) 80-0; Kurfurstendamm 195-196, 1000 Berlin 15, (030) 8 82 73 65; Dusseldorfer Strasse 40, 6236 Eschborn 1, (06196) 80 70; Kirchhorster Strasse 2, 3000 Hannover 51, (0511) 64 68-0; Maybachstrasse II, 7302 Ostfildern 2 (Nellingen), (0711) 3403257; Gildehofcenter, Hollestrasse 3, 4300 Essen 1, (0201) 24 25-0.

HOLLAND: Texas Instruments Holland B.V., B.P. 5320, 1182 HL Amstelveen, Holland, (020) 545 06 00.

HONG KONG: Texas Instruments Hong Kong Ltd., 8th Floor, World Shipping Center, 7 Canton Road, Kowloon, Hong Kong, 737-0338.

HUNGARY: Texas Instruments Representation, Budaörsi u. 50, 3rd floor H-1112 Budapest, Hungary, (1) 1. 66 66 17.

IRELAND: Texas Instruments Ireland Ltd., 7/8 Harcourt Street, Dublin 2, Ireland, (01) 475 52 33.

ITALY: Texas Instruments Italia S.p.A., Centro Direzionale Colleoni, Palazzo Perseo-Via Paracelso, 12 20041 Agrate Brianza (Mi), Italy, (039) 63 221; Via Castello della Magliana, 38, 00148 Roma, Italy (06) 657 26 51; Via Amendola, 17, 40100 Bologna, Italy (051) 554004.

JAPAN: Texas Instruments Japan Ltd., Aoyama Fuji Building 3-6-12 Kita-Aoyama Minato-ku, Tokyo, Japan 107, 03-498-2111; MS Shibaura Building 9F, 4-13-23 Shibaura, Minato-ku, Tokyo, Japan 108, 03-769-8700; Nissho-iwai Building 5F, 2-5-8 Imabashi, Chuou-ku, Osaka, Japan 541, 06-204-1881; Dai-ni Toyota Building Nishi-kan 7F, 4-10-27 Meieki, Nakamura-ku, Nagoya, Japan 450, 052-583-8691; Kanazawa Oyama-cho Daiichi Seimei Building 6F, 3-10 Oyama-cho, Kanazawa-shi, Ishikawa, Japan 920, 0762-23-5471; Matsumoto Showa Building 6F, 1-2-11 Fukashi, Matsumoto-shi, Nagano, Japan 390, 0263-33-1060; Daiichi Olympic Tachikawa Building 6F, 1-25-12, Akebono-cho, Tachikawa, Tokyo, Japan 190, 0425-27-6760; Yokohama Business Park East Tower 10F, 134 Goudo-cho, Hodogaya-ku, Yokohama-shi, Kanagawa, Japan 240, 045-338-1220; Nihon Seimei Kyoto Yasaka Building 5F, 843-2, Higashi Shiokohji-cho, Higashi-iru, Nishinotoh-in, Shiokohji-dori, Shirmogyo-ku, Kyoto, Japan 600, 075-341-7713; Sumitomo Seimei Kumagaya Building 8F, 2-44 Yayoi, Kumagaya-shi, Saitama, Japan 360, 0485-22-2240; 2597-1, Aza Harudai, Oaza Yasaka, Kitsuki-shi, Oita, Japan 873, 09786-3-3211.

KOREA: Texas Instruments Korea Ltd., 28th Floor, Trade Tower, 159, Samsung-Dong, Kangnam-ku Seoul, Korea, 2-551-2800.

MALAYSIA: Texas Instruments, Malaysia, Sdn. Bhd., Aisa Pacific, Lot 36.1 #Box 93, Menara Maybank, 100 Jalan Tun Perak, 50050 Kuala Lumpur, Malaysia, 2306001.

MEXICO: Texas Instruments de Mexico S.A. de C.V., Alfonso Reyes 115, Col. Hipodromo Condesa, Mexico, D.F., 06170, 5-515-6081.

PEOPLE'S REPUBLIC OF CHINA: Texas Instruments China Inc., Beijing Representative Office, 7-05 CITIC Building, 19 Jianguomenwai Dajie, Beijing, China, 500-2255, Ext. 3750.

PHILIPPINES: Texas Instruments Asia Ltd., Philippines Branch, 14th Floor, Ba-Lepanto Building, Paseo de Roxas, Makati, Metro Manila, Philippines, 2-8176031.

PORTUGAL: Texas Instruments Equipamento Electronico (Portugal) LDA., Eng, Frederico Ulricho, 2650 Moreira Da Maia, 4470 Maia, Portugal (2) 948 10-03.

SINGAPORE (& INDIA, INDONESIA, THAILAND): Texas Instruments Singapore (PTE) Ltd., Asia Pacific, 101 Thomson Road, #23-01, United Square, Singapore 1130, 3508100.

SPAIN: Texas Instruments Espana S.A., c/Gobelas 43, Urbanizasion La Florida, 28023, Madrid, Spain (1) 372 80 51; c/Diputacion, 279-3-5, 08007 Barcelona, Spain, (3) 317 91 80.

SWEDEN: Texas Instruments International Trade Corporation (Sverigefilialen), Isafjordsgatan Box 30, 164 93 Kista, Sweden, (08) 752 58 00.

TAIWAN: Texas Instruments Taiwan Limited, Taipei Branch, 10th Floor, Bank Tower, 205 Tung Hua N. Road, Taipei, Taiwan, 10592, Republic of China, 2-713 9311.

TURKEY: Texas Instruments, DSEG MidEast Regional Marketing Office, Karum Center, Suite 442, Iran Caddesi 21, 06680 Kavaklldere, Ankara, Turkey, 4-468-0155.

UNITED KINGDOM; Texas Instruments Ltd., Manton Lane, Bedford, England, MK41 7PA, (234) 270 111.

TEXAS
INSTRUMENTS

TEXAS INSTRUMENTS

Printed in UK