JAMES B. JOHNSON
STEVE KASSEL

The Multibus Design Guidebook

Structures, Architectures, and Applications

JAMES B. JOHNSON
STEVE KASSEL

The Multibus Design Guidebook

Structures, Architectures, and Applications

McGraw-Hill Book Company

New York St. Louis San Francisco Auckland
Bogotá Hamburg Johannesburg London Madrid
Mexico Montreal New Delhi Panama Paris
São Paulo Singapore Sydney Tokyo Toronto

Library of Congress Cataloging in Publication Data

Johnson, James B.
The multibus design guidebook.

Includes bibliographical references and index.
1. Microcomputers—Buses. 2. Computer architecture.
I. Kassel, Steve. II. Title.
TK7895.B87J63 1984 621.3819'535 83-16258
ISBN 0-07-032599-5

Davis, the designer was Riverside Graphics, and the production supervisor was Thomas G Kowalczyk It was set in Caledonia by University Graphics, Inc

Printed and bound by R. R. Donnelley & Sons Company.

Photographs, tables, and figures are used courtesy of Intel Corporation, Santa Clara, California.

Preface

The Multibus/IEEE-796 is a commercial-quality industry-standard bus structure for use in microprocessor-based systems. Additionally, three separate buses have been developed to complement and extend the capabilities of the Multibus structure. Together these four structures form the Multibus family of structures. The Multibus family consists of the Multibus system bus, which is the center of all Multibus-based systems; the iSBX bus, a low-cost local (on-board) input-output expansion bus; the Multichannel bus, a very high speed cable bus designed to move blocks of data between peripherals and intelligent subsystems and Multibus-based systems; and the iLBX bus, a high-speed memory execution bus that allows a microprocessor on a single-board computer to expand its local memory using multiple boards.

This book provides the reader with a basic understanding of the structures, architectures, and detailed hardware designs of the various modules that can be used in association with the Multibus system bus, the iSBX local input-output bus, the Multichannel high-speed cable bus, and the iLBX local execution bus. It describes these various bus structures using simple concepts, and then builds on them until the reader understands the different architectures that can be constructed. The book, which is intended for board- and system-level hardware design and evaluation engineers and their managers, is essential for anyone involved with Multibus-based products. It provides detailed bus interface information and also serves as a quick reference for those designing Multibus-based systems. The text is supported by a wealth of examples and illustrations.

The book is divided into three parts: (1) structures, in which the electrical and mechanical specifications of the Multibus family members are described; (2) architectures, in which the different architectures are described that can be built on and around the Multibus family members; and (3) applications, in which hardware design examples are given for interfacing modules to the various Multibus family members.

The structures section reviews each of the different Multibus family structures. First the structures are described conceptually; then the functions and the electrical and mechanical specifications of the bus are described in detail.

The Multibus family of structures supports a wide spectrum of system architectures, from simple, low-cost uniprocessing systems to sophisticated, distributed multiple-processor systems yielding high throughput. The architectures section considers the benefits and trade-offs of each of these different architectures in detail. Examples of several types of systems, including uniprocessing, multicomputing, and multiprocessing systems, are used to explain the major architectural approaches, interconnection schemes, and related hardware and software trade-offs. Other topics covered include an overview of system design issues and some discussion of highly reliable computers.

The applications section gives the reader examples of various interface circuits for the Multibus family structures. Each example provides enough detail to make it possible to actually implement the module or interface. This section also provides evaluation criteria for purchasing Multibus-compatible products.

James B. Johnson
Steve Kassel

Acknowledgments

We wish to acknowledge the help of many colleagues at Intel's OEM Microcomputer Operations group (Hillsboro, Oregon) over the years during which most of these ideas, specifications, and applications were formulated and refined. The special efforts of Clark Allsworth, who provided constructive criticism of the early manuscript, have been most helpful. The final manuscript also has benefited greatly from the careful reading and comments of Craig Kinnie and Mary Slamp. The assistance of the engineering staff of Intel Corporation—especially Phil Drain, Pete MacWilliams, John Deignan, Scott Tetrick, and Ron Dilbeck—is also acknowledged.

Contents

PART 1

The Multibus Family of Bus Structures

1 Introduction

This chapter provides a basic framework for evaluating microprocessor system buses and offers a little history of some of the most popular of such buses: the Multibus/IEEE-796 system bus and its extensions: the iSBX bus, the iLBX bus, and the Multichannel bus.[1] Basic system architectures of the Multibus/IEEE-796 family will also be defined.

1.1 OBJECTIVES AND GOALS OF MICROPROCESSOR-BASED SYSTEM BUSES

The system bus is the foundation of any computer system; it will influence the flexibility, cost, performance, and reliability of the system for its entire operating life. Advances in very large scale integration (VLSI) technology result in increased system complexity. The system bus, as a result, is recognized as the primary architectural resource, and it can frequently be the limiting factor in performance, reliability, and modularity. The most basic portion of a system bus is the bus structure, which defines all the signals and how the various system components interact with each other. These signals run along the backplane, where they can be supplied to the interface modules. A typical bus structure defines the word length, data types, and address length, as well as data transfer protocols such as memory reads, input-output (I/O) writes, and direct memory accesses (DMA). It will also specify some type of intermodule signaling such as interrupts, as well as a protocol to exchange control of the bus to various bus modules.

[1]Multibus, iSBX, iLBX, and Multichannel are trademarks of Intel Corporation, Santa Clara, California.

1.1.1 Do You Need a System Bus?

Not all users need a bus-oriented system. Such a system is generally more flexible, easier to upgrade, and easier to implement, but it is more expensive, module for module, than a specialized system. This expense is due to the greater component count required to meet the bus interface specification. Typically, a bus specification requires that each signal line be buffered. That can result in excess drive capacity, since most system designs use only a small fraction of the allowable receivers permitted on a signal. The buffers also require additional area on the board and increase power consumption. The additional parts increase the component cost, assembly time, and test time, which results in increased manufacturing cost.

In applications with lesser volume it will generally be found that the added cost of using standard bus design methodologies will be favorably offset by lower development costs during the shorter development time. In many instances, complete systems can be configured with off-the-shelf board-level products. In applications in which some custom design is required, standard bus design methodology is still applicable. The system design can be divided into two parts: the custom boards and the standard boards. The customized portion of the system can be completed with less expenditure of time and money because the system bus interface is already designed. The entire system development cost is lower because part of the system uses standard products. Bus-oriented systems also have a greater degree of configuration flexibility because different modules can be mixed and matched to produce a particular product or version. Products can easily be configured to meet the exact need of the end user.

Another important aspect of using standard bus-oriented systems is the ability to buffer a system design from the rapid technological changes in VLSI components. If a design needs more speed, it can be upgraded by plugging in a new bus-compatible module that uses a faster microprocessor or faster memory. Even using new technology such as converting a current design which used an 8-bit microprocessor to a 16-bit microprocesor would be permitted if the module met the bus interface requirements.

In summary, in applications that have very high volume, such as terminals or low-cost test equipment, it will be found that the use of a standard bus system adds undesirable cost to the end product. Systems that are dominated by manufacturing costs and do not need a great deal of configuration flexibility should use specialized configurations to avoid the costs of unnecessary parts and interconnections. On the other hand, systems that are dominated by development costs or need configuration flexibility should use a system bus scheme.

1.2 PICKING YOUR MULTIBUS FAMILY STRUCTURES

The Multibus system bus is a commercial quality bus for use in microprocessor-based systems. Some Multibus boards are shown in Fig. 1-1. The Multibus struc-