intel

The 8086 Family User's Manual

October 1979

intel

The 8086 Family User's Manual

October 1979

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

The information in this document is subject to change without notice.

Intel Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this document. Intel Corporation makes no commitment to update nor to keep current the information contained in this document.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication or disclosure is subject to restrictions stated in Intel's software license, or as defined in ASPR 7-104.9(a)(9).

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of Intel Corporation.

The following are trademarks of Intel Corporation and may be used only to describe Intel products:

i	iSBC	Multimodule
iCE	Library Manager	PROMPT
iCS	MCS	Promware
Insite	Megachassis	RMX
Intel	Micromap	UPI
Intelevision	Multibus	μScope
Intellec		

and the combination of ICE, iCS, iSBC, MCS, or RMX and a numerical suffix.

Table of Contents

CHAPTER 1
INTRODUCTION

PAGE

Manual Organization 1-1
8086 Family Architecture 1-1
Functional Distribution 1-1
Microprocessors 1-2
Interrupt Controller 1-3
Bus Interface Components 1-3
Multiprocessing 1-3
Bus Organization 1-4
Local Bus 1-4
System Bus 1-5
Processing Modules 1-6
Bus Implementation Examples 1-6
Development Aids 1-12

CHAPTER 2
THE 8086 AND 8088 CENTRAL PROCESSING UNITS

Processor Overview 2-1
Processor Architecture 2-3
Execution Unit 2-5
Bus Interface Unit 2-5
General Registers 2-6
Segment Registers 2-7
Instruction Pointer 2-7
Flags 2-7
8080/8085 Register and Flag Correspondence 2-8
Mode Selection 2-8
Memory 2-8
Storage Organization 2-8
Segmentation 2-10
Physical Address Generation 2-11
Dynamically Relocatable Code 2-13
Stack Implementation 2-14
Dedicated and Reserved Memory Locations 2-14
8086/8088 Memory Access Differences 2-15
Input/Output 2-15
Input/Output Space 2-16
Restricted I/O Locations 2-16
8086/8088 Memory Access Differences 2-16
Memory-Mapped I/O 2-16
Direct Memory Access 2-17
8089 Input/Output Processor (IOP) 2-17
Multiprocessing Features 2-17
Bus Lock 2-17
WAIT and $\overline{\text{TEST}}$ 2-18
Escape 2-19
Request/Grant Lines 2-20
Multibus™ Architecture 2-21

PAGE

8289 Bus Arbiter 2-22
Processor Control and Monitoring 2-22
Interrupts 2-22
External Interrupts 2-22
Internal Interrupts 2-24
Interrupt Pointer Table 2-25
Interrupt Procedures 2-26
Single-Step (Trap) Interrupt 2-28
Breakpoint Interrupt 2-28
System Reset 2-29
Instruction Queue Status 2-29
Processor Halt 2-29
Status Lines 2-30
Instruction Set 2-30
Data Transfer Instructions 2-31
General Purpose Data Transfers 2-31
Address Object Transfers 2-32
Flag Transfers 2-32
Arithmetic Instructions 2-33
Arithmetic Data Formats 2-33
Arithmetic Instructions and Flags 2-34
Addition 2-35
Subtraction 2-36
Multiplication 2-36
Division 2-37
Bit Manipulation Instructions 2-38
Logical 2-38
Shifts 2-39
Rotates 2-39
String Instructions 2-40
Program Transfer Instructions 2-43
Unconditional Transfers 2-43
Conditional Transfers 2-45
Iteration Control 2-45
Interrupt Instructions 2-46
Processor Control Instructions 2-47
Flag Operations 2-47
External Synchronization 2-48
No Operation 2-48
Instruction Set Reference Information 2-48
Addressing Modes 2-68
Register and Immediate Operands 2-68
Memory Addressing Modes 2-68
The Effective Address 2-68
Direct Addressing 2-69
Register Indirect Addressing 2-69
Based Addressing 2-70
Indexed Addressing 2-70

PAGE

Based Indexed Addressing 2-71
String Addressing 2-72
I/O Port Addressing 2-72
Programming Facilities 2-72
Software Development Overview 2-73
PL/M-86 2-75
Statements and Comments 2-75
Data Definition 2-75
Assignment Statement 2-77
Program Flow Statements 2-79
Procedures 2-81
ASM-86 2-83
Statements 2-83
Constants 2-84
Defining Data 2-85
Records 2-85
Structures 2-87
Addressing Modes 2-87
Segment Control 2-88
Procedures 2-90
LINK-86 2-90
LOC-86 2-90
LIB-86 2-91
OH-86 2-91
CONV-86 2-92
Sample Programs 2-92
Programming Guidelines and Examples 2-96
Programming Guidelines 2-96
Segments and Segment Registers 2-96
Self-Modifying Code 2-96
Input/Output 2-97
Operating Systems 2-97
Interrupt Service Procedures 2-99
Stack-Based Parameters 2-100
Flag Images 2-100
Programming Examples 2-100
Procedures 2-100
Jumps and Calls 2-105
Records 2-110
Dynamic Code Relocation 2-113
Memory-Mapped I/O 2-115
Breakpoints 2-117
Interrupt Procedures 2-119
String Operations 2-125

CHAPTER 3
THE 8089 INPUT/OUTPUT PROCESSOR

Processor Overview 3-1
Evolution 3-1
Principles of Operation 3-2
CPU/IOP Communications 3-2
Channels 3-4
Channel Programs (Task Blocks) 3-4

PAGE

DMA Transfers 3-5
Bus Configurations 3-5
A Sample Transaction 3-10
Applications 3-12

Processor Architecture 3-13
Common Control Unit (CCU) 3-13
Arithmetic/Logic Unit (ALU) 3-13
Assembly/Disassembly Registers 3-14
Instruction Fetch Unit 3-14
Bus Interface Unit (BIU) 3-16
Channels 3-16
I/O Control 3-16
Registers 3-17
Program Status Word 3-18
Tag Bits 3-19
Concurrent Channel Operation 3-20
Memory 3-21
Storage Organization 3-22
Dedicated and Reserved Memory Locations 3-23
Dynamic Relocation 3-23
Memory Access 3-24
Input/Output 3-25
Programmed I/O 3-25
I/O Instructions 3-25
Device Addressing 3-26
I/O Bus Transfers 3-26
DMA Transfers 3-27
Preparing the Device Controller 3-27
Preparing the Channel 3-27
Beginning the Transfer 3-31
DMA Transfer Cycle 3-32
Following the Transfer 3-33
Multiprocessing Features 3-34
Bus Arbitration 3-34
Request/Grant Line 3-35
8289 Bus Arbiter 3-36
Bus Arbitration for IOP Configurations 3-36
Bus Load Limit 3-36
Bus Lock 3-37
Processor Control and Monitoring 3-37
Initialization 3-37
Channel Commands 3-40
DRQ (DMA Request) 3-43
EXT (External Terminate) 3-43
Interrupts 3-43
Status Lines 3-43
Instruction Set 3-44
Data Transfer Instructions 3-44
Arithmetic Instructions 3-45
Logical and Bit Manipulation Instructions 3-46
Program Transfer Instructions 3-48
Processor Control Instructions 3-49
Instruction Set Reference Information 3-51

此为试读,需要完整PDF请访问: www.ertongbook.com

PAGE

Addressing Modes 3-59
Register and Immediate Operands 3-59
Memory Addressing Modes 3-59
The Effective Address 3-60
Based Addressing 3-60
Offset Addressing 3-60
Indexed Addressing 3-60
Indexed Auto-Increment Addressing 3-61
Programming Facilities 3-63
ASM-89 3-63
Statements 3-63
Constants 3-66
Defining Data 3-66
Structures 3-67
Addressing Modes 3-68
Program Transfer Targets 3-68
Procedures 3-69
Segment Control 3-69
Intermodule Communication 3-70
Sample Program 3-73
Linking and Locating ASM-89 Modules 3-76
Programming Guidelines and Examples 3-79
Programming Guidelines 3-79
Segments 3-79
Self-Modifying Code 3-79
I/O System Design 3-79
Programming Examples 3-81
Initialization and Dispatch 3-81
Memory-to-Memory Transfer 3-85
Saving and Restoring Registers 3-85

CHAPTER 4
HARDWARE REFERENCE INFORMATION

Introduction 4-1
8086 and 8088 CPUs 4-1
CPU Architecture 4-1
Bus Operation 4-5
Clock Circuit 4-10
Minimum/Maximum Mode 4-10
Minimum Mode 4-11
Maximum Mode 4-11
External Memory Addressing 4-14
I/O Interfacing 4-15
Interrupts 4-16
Machine Instruction Encoding and Decoding 4-18
8086 Instruction Sequence 4-37
8089 I/O Processor 4-38
System Configuration 4-39
Local Mode 4-39
Remote Mode 4-40
Bus Operation 4-41
Initialization 4-44
I/O Dispatching 4-46

PAGE

DMA Transfers 4-47
DMA Termination 4-50
Peripheral Interfacing 4-50
Instruction Encoding 4-52

APPENDIX A
APPLICATION NOTES

AP-67 8086 System Design A-3
AP-61 Multitasking for the 8086 ..... A-67
AP-50 Debugging Strategies and Considerations for 8089 Systems A-85
AP-51 Designing 8086, 8088, 8089 Multiprocessing Systems with the 8289 Bus Arbiter A-111
AP-59 Using the 8259A Programmable Interrupt Controller A-135
AP-28A Intel® Multibus™ Interfacing A-175
AP-43 Using the iSBC-957™ Execution Vehicle for Executing 8086 Program Code A-209

APPENDIX B
DEVICE SPECIFICATIONS

8086 Family
8086/8086-2/8086-4 16-Bit HMOS Microprocessor B-1
M8086 16-Bit HMOS Microprocessor B-22
I8086 16-Bit HMOS Microprocessor B-23
8088 8-Bit HMOS Microprocessor B-24
8089 8/16-Bit HMOS I/O Processor B-46
8282/8283 Octal Latch B-59
8284 Clock Generator and Driver for 8086, 8088, 8089 Processors B-63
M8284 Clock Generator and Driver for 8086, 8088, 8089 Processors B-69
I8284 Clock Generator and Driver for 8086, 8088, 8089 Processors B-70
8286/8287 Octal Bus Transceiver B-71
8288 Bus Controller for 8086, 8088, 8089 Processors B-75
8289 Bus Arbiter B-81
8237/8237-2 High Performance Programmable DMA Controller B-92
8259A/8259A-2/8289A-8 Programmable Interrupt Controller B-106
8085 Peripherals
8155/8156/8155-2/8156-2 2048 Bit Static MOS RAM with I/O Ports and Timer ... B-124
8185/8185-2 1024 x 8-Bit Static RAM for MCS-85™ B-125
8355/8355-2 16,384-Bit ROM with I/O B-126
8755A/8755A-2 16,384-Bit EPROM with I/O B-127

PAGE

Standard Peripherals
8041A/8741A Universal Peripheral Interface 8-Bit Microcomputer B-128
8202 Dynamic RAM Controller..... B-129
8205 High Speed 1 Out of 8 Binary Decoder .. B-130
8251A Programmable Communication Interface..... B-131
8253/8253-5 Programmable Interval Timer... B-132
8255A/8255A-5 Programmable Peripheral Interface..... B-133
8271/8271-6/8271-8 Programmable Floppy Disk Controller B-134
8273 Programmable HDLC/SDLC Protocol Controller..... B-135
8275 Programmable CRT Controller B-136
8279/8279-5 Programmable Keyboard/Display Interface..... B-137
8291 GPIB Talker/Listener B-138
8292 GPIB Controller..... B-139
8293 GPIB Transceiver..... B-140
8294 Data Encryption Unit B-141
8295 Dot Matrix Printer Controller B-142

RAM Memories
2114A 1024 x 4 Bit Static RAM B-143
2142 1024 x 4 Bit Static RAM..... B-144

PAGE

2148 1024 x 4 Bit Static RAM..... B-145

EPROM Memories
2716 16K (2K x 8) UV Erasable PROM..... B-146
2732 32K (4K x 8) UV Erasable PROM..... B-147
2758 8K (1K x 8) UV Erasable Low Power PROM B-148

Development Tools
Model 230 Intellec® Series II Microcomputer Development System.... B-149
8086/8088 Software Development Package B-153
8089 Assembler Support Package B-163
ICE-86™ 8086 In-Circuit Emulator B-165
iSBC 86/12A™ Single Board Computer B-171
iSBC 957™ Intellec®-iSBC 86/12A™ Interface and Execution Package..... B-179
iSBC 300/340™ iSBC 300™ 32K-Byte RAM Expansion Module iSBC 340™ 16K-Byte EPROM/ROM Expansion Module B-184
SDK-86 MCS-86™ System Design Kit B-188
SDK-C86 MCS-86™ System Design Kit B-194

CHAPTER 1
INTRODUCTION

This publication describes the Intel® 8086 family of microcomputing components, concentrating on the 8086, 8088 and 8089 microprocessors. It is written for hardware and software engineers and technicians who understand microcomputer operating principles. The manual is intended to introduce the product line and to serve as a reference during system design and implementation.

Recognizing that successful microcomputer-based products are judicious blends of hardware and software, the *User's Manual* addresses both subjects, although at different levels of detail. This publication is the definitive source for information describing the 8086 family components. Software topics, such as programming languages, utilities and examples, are given moderately detailed, but by no means complete, coverage. Additional references, available from Intel's Literature Department, are cited in the programming sections.

1.1 Manual Organization

The manual contains four chapters and three appendices. The remainder of this chapter describes the architecture of the 8086 family, and subsequent chapters cover the individual components in detail.

Chapter 2 describes the 8086 and 8088 Central Processing Units, and Chapter 3 covers the 8089 Input/Output Processor. These two chapters are identically organized and focus on providing a *functional* description of the 8086, 8088 and 8089, plus related Intel hardware and software products. Hardware reference information—electrical characteristics, timing and physical interfacing considerations—for all three processors is concentrated in Chapter 4.

Appendix A is a collection of 8086 family application notes; these provide design and debugging examples. Appendix B contains complete data sheets for all the 8086 family components and system development aids; summary data sheets covering compatible components from other Intel product lines are also reproduced in Appendix B.

1.2 8086 Family Architecture

Considered individually, the 8086, 8088 and 8089 are advanced third-generation microprocessors. Moreover, these processors are elements of a larger design, that of the 8086 family. This systems architecture specifies how the processors and other components relate to each other, and is the key to the exceptional versatility of these products.

The components in the 8086 family have been designed to operate together in diverse combinations within the systematic framework of the overall family architecture. In this way a single family of components can be used to solve a wide array of microcomputing problems. A component mix can be tailored to fit the performance needs of an application precisely, without having to pay for unneeded capabilities that may be bundled into more monolithic, CPU-centered architectures. Using the same family of components across multiple systems limits the learning curve problem and builds on past experience. Finally, the modular structure of the family architecture provides an orderly way for systems to grow and change.

The 8086 family architecture is characterized by three major principles:

1. System functions are distributed among specialized components.
2. Multiprocessing capabilities are inherent in the hardware.
3. A hierarchical bus organization provides for the complex data flows required by high-performance systems without burdening simpler systems with unneeded capabilities.

Functional Distribution

Table 1-1 lists the components that constitute the 8086 microprocessor family. All components are contained in standard dual in-line packages and require single +5V power sources.

Table 1-1. 8086 Component Family

Microprocessor	Technology	Pins	Description
8086 Central Processing Unit (CPU)	HMOS	40	8/16 bit general-purpose microprocessor; 16-bit external data path.
8088 Central Processing Unit (CPU)	HMOS	40	8/16 bit general-purpose microprocessor; 8-bit external data path.
8089 Input/Output Processor (IOP)	HMOS	40	8/16 bit microprocessor optimized for high-speed I/O operations; 8-bit and 16-bit external data paths.

Support Component	Technology	Pins	Function
8259A Programmable Interrupt Controller (PIC)	NMOS	28	Identifies highest-priority interrupt request.
8282 Octal Latch 8283 Octal Latch (Inverting)	Bipolar	20	Demultiplexes and increases drive of address bus.
8284 Clock Generator and Driver	Bipolar	18	Provides time base.
8286 Octal Bus Transceiver 8287 Octal Bus Transceiver (Inverting)	Bipolar	20	Increases drive on data bus.
8288 Bus Controller	Bipolar	20	Generates bus command signals.
8289 Bus Arbiter	Bipolar	20	Controls access of microprocessors to multimaster system bus.

Microprocessors

At the core of the product line are three microprocessors that share these characteristics:

- Standard operating speed is 5 MHz (200 ns cycle time); a selected 8 MHz version of the 8086 CPU is also available.
- Chips are housed in reliable 40-pin packages.
- Processors operate on both 8- and 16-bit data types; internal data paths are at least 16 bits wide.
- Up to 1 megabyte of memory can be addressed, along with a separate 64k byte I/O space.
- The address/data and status interfaces of the processors are compatible (the address and data buses are time-multiplexed at the processor, i.e., an address transmission is followed by a data transmission over a subset of the same physical lines).

The 8086 and 8088 are third-generation central processing units (CPUs) that differ primarily in their external data paths. The 8088 transfers data between itself and other system components 8 bits at a time. The 8086 can transfer either 8 or 16 bits in one bus cycle and is therefore capable of greater throughput. Both processors have two operating modes, selectable by a strapping pin. In minimum mode, the CPUs emit the bus control signals needed by memory and I/O peripheral components. In maximum mode, an 8288 Bus Controller assumes responsibility for controlling devices attached to the system bus. CPU pins no longer needed for bus control are then redefined to provide signals that support multiprocessing systems.

The 8089 Input/Output Processor (IOP) is an independent microprocessor whose design has been optimized for transferring data. The 8089

typically runs under the direction of a CPU, but it executes a separate instruction stream and can operate in parallel with other system processors. The IOP contains two independent I/O channels that combine attributes of both CPUs and advanced DMA (direct memory access) controllers. The channels can execute programs and perform programmed I/O operations similar to CPUs. They may also transfer data by DMA, at rates up to 1.25 megabytes per second (5 MHz version). The channels can support mixes of 8- and 16-bit I/O devices and memory. Combining speed with programmable intelligence, the 8089 can assume the bulk of I/O processing overhead and thereby free a CPU to perform other tasks.

Interrupt Controller

The 8259A Programmable Interrupt Controller (PIC) is a new, 8086 family-compatible version of the familiar 8259 that has been enhanced to operate with the advanced interrupt facilities of the 8086 and 8088 CPUs. The 8259A accepts interrupt requests from up to eight sources; up to 64 sources may be accommodated by "cascading" additional 8259As. Each interrupt source is assigned a priority number that typically reflects its "criticality" in the system. The 8259A has several built-in, priority-resolving mechanisms that are selectable by software commands from the CPU. These modes operate somewhat differently, but in general the 8259A continuously identifies the highest-priority active interrupt request and generates an interrupt request to the CPU if this request has higher priority than the request currently being processed. When the CPU recognizes the interrupt request, the 8259A transfers a code to the CPU that identifies the interrupt source.

Bus Interface Components

Components may be selected from this modular group to implement different system bus configurations. Except for the 8284, all components are optional; their inclusion in a system is based on the needs of the application. All of the bus interface components are implemented using bipolar technology to provide high-quality, high-drive signals and very fast internal switching.

The 8284 Clock Generator and Driver provides the time base for the 8086 family microprocessors. It divides the frequency signal from an external crystal or TTL signal by three and outputs the 5 MHz or 8 MHz processor clock signal. It also provides the microprocessors with reset and ready signals.

8282 or 8283 Octal Latches may be added to a system to demultiplex the combined address/data bus generated by the 8086 family microprocessors. A demultiplexed bus provides separate stable address and data lines required by many peripheral components. Two latches demultiplex 16 bits of the bus to provide an address space of up to 64k bytes, while three latches generate the full 20-bit (megabyte) address space. The latches also provide the high drive on the address lines needed in larger systems.

8286 and 8287 Octal Bus Transceivers are used to provide more drive on data lines than the processors themselves are capable of providing. One or two transceivers may be used depending on the width of the data bus (8 or 16 bits).

The 8288 Bus Controller decodes status signals output by an 8089, or a maximum mode 8086 or 8088. When these signals indicate that the processor is to run a bus cycle, the 8288 issues a bus command that identifies the bus cycle as memory read, memory write, I/O read, I/O write, etc. It also provides a signal that strobes the address into 8282/83 latches. The 8288 provides the drive levels needed for the bus control lines in medium to large systems.

The 8289 Bus Arbiter controls the access of a processor to a multimaster system bus. A multimaster bus is a path to system resources (typically memory) that is shared by two or more microprocessors (masters). Arbiters for each master may use one of several priority-resolving techniques to ensure that only one master drives the shared bus.

Multiprocessing

Employing multiple processors in medium to large systems offers several significant advantages over the centralized approach that relies on a single CPU and extremely fast memory:

- system tasks may be allocated to special-purpose processors whose designs are optimized to perform certain types of tasks simply and efficiently;

- very high levels of performance can be attained when multiple processors can execute simultaneously (parallel processing);
- robustness can be improved by isolating system functions so that a failure or error in one part of the system has a limited effect on the rest of the system;
- the natural partitioning of the system promotes parallel development of subsystems, breaks the application into smaller, more manageable tasks, and helps isolate the effects of system modifications.

The 8086 family architecture is explicitly designed to simplify the development of multiple processor systems by providing facilities for coordinating the interaction of the processors.

The architecture supports two types of processors: independent processors and coprocessors. An independent processor is one that executes its own instruction stream. The 8086, 8088 and 8089 are examples of independent processors. An 8086 or 8088 typically executes a program in response to an interrupt. The 8089 starts its channels in response to an interrupt-like signal called a channel attention; this signal is typically issued by a CPU.

The 8086 architecture also supports a second type of processor, called a coprocessor. Coprocessor "hooks" have been designed into the 8086 and 8088 so that this type of processor can be accommodated in the future. A coprocessor differs from an independent processor in that it obtains its instructions from another processor, called a host. The coprocessor monitors instructions fetched by the host and recognizes certain of these as its own and executes them. A coprocessor, in effect, extends the instruction set of its host processor.

The 8086 family architecture provides built-in solutions to two classic multiprocessing coordination problems: bus arbitration and mutual exclusion. Bus arbitration may be performed by the bus request/grant logic contained in each of the processors, by 8289 Bus Arbiters, or by a combination of the two when processors have access to multiple shared buses. In all cases, the arbitration mechanism operates invisibly to software.

For mutual exclusion, each processor has a LOCK (bus lock) signal which a program may activate to prevent other processors from obtaining a shared system bus. The 8089 may lock the bus during a DMA transfer to ensure that both the transfer completes in the shortest possible time and that another processor does not access the target of the transfer (e.g., a buffer) while it is being updated. Each of the processors has an instruction that examines and updates a memory byte with the bus locked. This instruction can be used to implement a semaphore mechanism for controlling the access of multiple processors to shared resources. (A semaphore is a variable that indicates whether a resource, such as a buffer or a pointer, is "available" or "in use"; section 2.5 discusses semaphores in more detail).

Bus Organization

Figure 1-1 summarizes the 8086 family bus structure. There are two different types of buses: system and local. Both buses may be shared by multiple processors, i.e., both are multimaster buses. Microprocessors are always connected to a local bus, and memory and I/O components usually reside on a system bus. The 8086 family bus interface components link a local bus to a system bus.

Local Bus

The local bus is optimized for use by the 8086 family microprocessors. Since standard memory and I/O components are not attached to the local bus, information can be multiplexed and encoded to make very efficient use of processor pins (certain MCS-85™ peripheral components can be directly connected to the local bus). This allows several pins to be dedicated to coordinating the activity of multiple processors sharing the local bus. Multiple processors connected to the same local bus are said to be local to each other; processors on different local buses are said to be remote to each other, or configured remotely. Both independent processors and coprocessors may share a local bus; on-chip arbitration logic determines which processor drives the bus. Because the processors on the local bus share the same bus interface components, the local configuration of multiple processors provides a compact and inexpensive multiprocessing system.

此为试读，需要完整PDF请访问：www.ertongbook.com

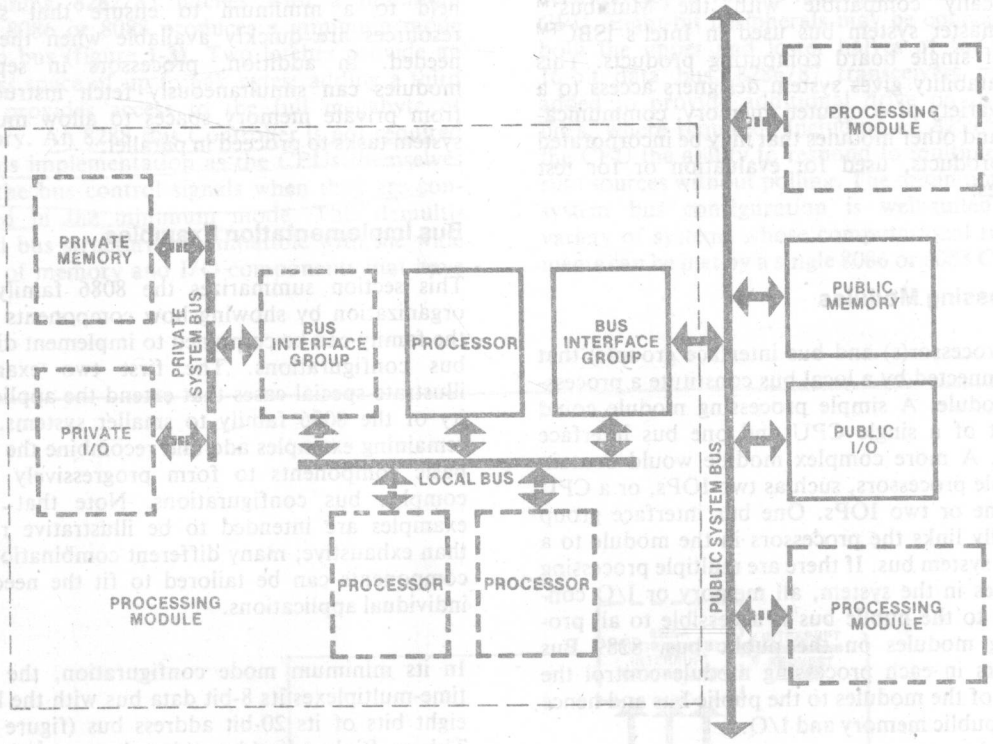

Figure 1-1. Generalized 8086 Family Bus Structure

System Bus

A full implementation of an 8086 system bus consists of the following five sets of signals:

1. address bus,
2. data bus,
3. control lines,
4. interrupt lines, and
5. arbitration lines.

These signals are designed to meet the needs of standard memory and I/O devices; the address and data buses are demultiplexed and traditional control signals (memory read/write, I/O read/write, etc.) are provided on the system bus.

The system bus design is modular and subsets may be implemented according to the needs of the application. For example, the arbitration lines are not needed in single-processor systems or in multiple-processor systems that perform arbitration at the local-bus level.

A group of bus interface components transforms the signals of a local bus into a system bus. The number of bus interface components required to generate a system bus depends on the size and complexity of the system; reduced application needs translate directly into reduced component counts. These main variables determine the configuration of a bus interface group: address space size (number of latches), data bus width (number of transceivers), and arbitration needs (presence of a bus arbiter).

The 8086 family system bus is functionally and electrically compatible with the Multibus™ multimaster system bus used in Intel's iSBC™ line of single board computing products. This compatability gives system designers access to a wide variety of computer, memory, communications and other modules that may be incorporated into products, used for evaluation or for test vehicles.

Processing Modules

The processor(s) and bus interface group(s) that are connected by a local bus constitute a processing module. A simple processing module could consist of a single CPU and one bus interface group. A more complex module would contain multiple processors, such as two IOPs, or a CPU and one or two IOPs. One bus interface group typically links the processors in the module to a public system bus. If there are multiple processing modules in the system, all memory or I/O connected to the public bus is accessible to all processing modules on the public bus. 8289 Bus Arbiters in each processing module control the access of the modules to the public bus and hence to the public memory and I/O.

A second bus interface group may be connected to a processing module's local bus, generating a second bus. This bus can provide the processing module with a private address space that is not accessible to other processing modules. Distributing memory and I/O resources in this manner can improve system robustness by isolating the effects of failures. It can also increase system throughput dramatically. If processor programs and local data are placed in private memory, contention for use of the public system bus can be held to a minimum to ensure that shared resources are quickly available when they are needed. In addition, processors in separate modules can simultaneously fetch instructions from private memory spaces to allow multiple system tasks to proceed in parallel.

Bus Implementation Examples

This section summarizes the 8086 family bus organization by showing how components from the family can be combined to implement diverse bus configurations. The first two examples illustrate special cases that extend the applicability of the 8086 family to smaller systems. The remaining examples add and recombine the same basic components to form progressively more complex bus configurations. Note that these examples are intended to be illustrative rather than exhaustive; many different combinations of components can be tailored to fit the needs of individual applications.

In its minimum mode configuration, the 8088 time-multiplexes its 8-bit data bus with the lower eight bits of its 20-bit address bus (figure 1-2). This multiplexed address/data bus, and the bus control signals emitted by the 8088, are directly compatible with the multiplexed bus components of Intel's 8085 family. These peripherals contain on-chip logic that demultiplexes a combined address/data bus. In addition, many of these devices are multifunctional, combining, for example, RAM, I/O ports and a timer on a single chip. By using these components, it is possible to build small (as few as four chips) economical systems that are nonetheless capable of performing significant computing tasks.

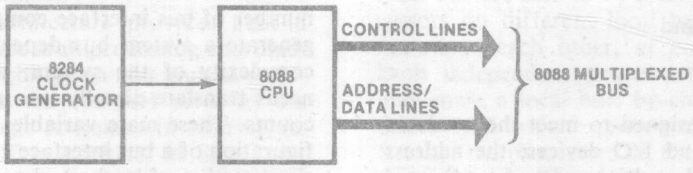

Figure 1-2. 8088 Multiplexed Bus

Combining 8282/83 latches with a minimum mode 8086 or 8088 produces a minimum mode system bus (figure 1-3). Two latches provide an address space of up to 64k bytes; adding a third latch provides access to the full megabyte of memory. An 8288 Bus Controller is not required for this implementation as the CPUs themselves emit the bus control signals when they are configured in the minimum mode. This demultiplexed bus structure is compatible with the wide array of memory and I/O components that have been developed for the industry-standard 8080A CPU. Eight-bit peripherals may be connected to both the upper and lower halves of the 8086's 16-bit data bus. 8286/87 transceivers may be added to provide additional drive on the data lines, where required. Including an 8259A gives the CPU the ability to respond to multiple interrupt sources without polling. The minimum mode system bus configuration is well-suited to a variety of systems whose computational requirements can be met by a single 8086 or 8088 CPU.

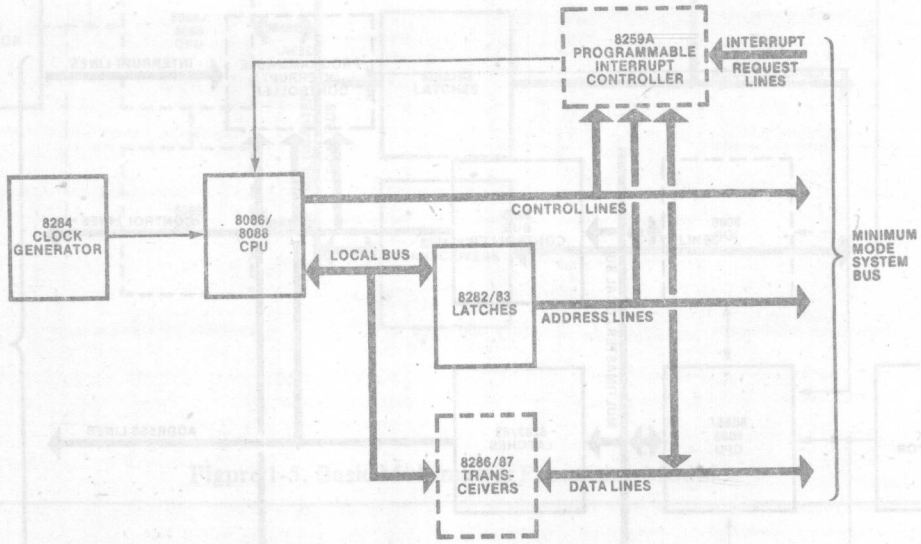

Figure 1-3. Minimum Mode System Bus

When an 8086 or 8088 is configured in maximum mode and an 8288 is added to control the system bus, one or two 8089s may be directly connected to the CPU (figure 1-4). The processors all share the same latches, transceivers, clock and bus controller, via the local bus. Arbitration logic built into the 8086, 8088 and 8089 coordinates use of the local bus, and thus of the system bus. This bus configuration enables the powerful I/O handling capabilities of the 8089 to be incorporated into systems of moderate size and cost.

The 8289 enables high-performance systems to be designed as a series of independent processing modules whose activities are coordinated via a shared system bus. Figure 1-5 shows the multimaster system bus interface; this bus structure is electrically compatible with the Multibus™ architecture used in Intel iSBC™ single-board computing systems.

Several different combinations of processors may be attached to the local bus of a multimaster computing module:

- a single 8086 or 8088
- a single 8089
- two 8089s
- an 8086 or 8088 and one 8089
- an 8086 or 8088 and two 8089s

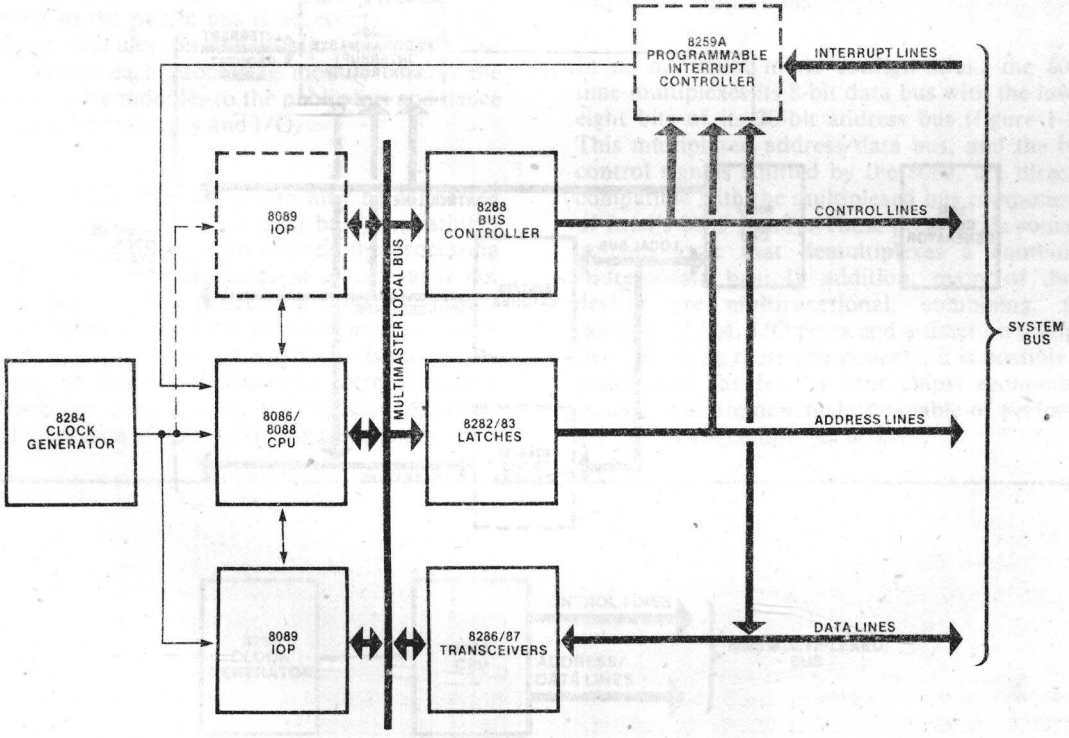

Figure 1-4. Multimaster Local Bus

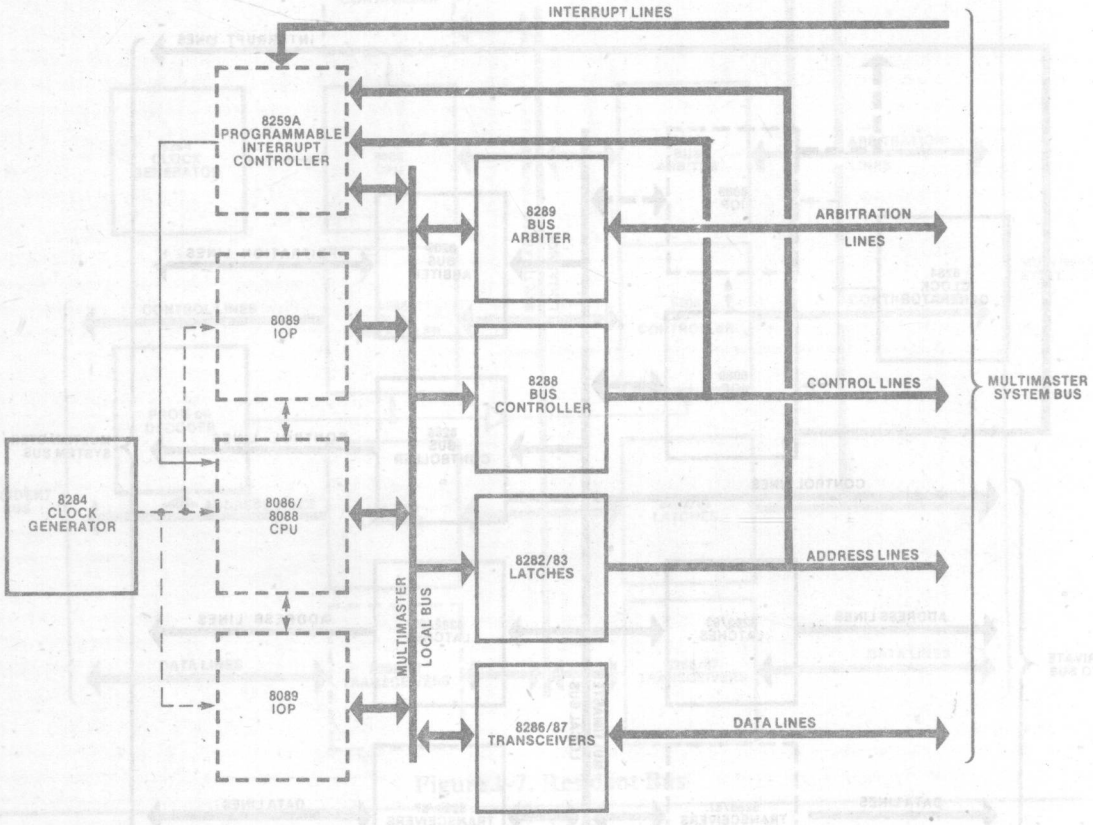

Figure 1-5. Basic Multimaster Processing Module

All of the processors on the local bus obtain access to the system bus through a single set of interface components.

One or two 8089s in a multimaster processing module may be configured with a private I/O bus as shown in figure 1-6. In this configuration, memory access commands are directed to the public multimaster system bus, while I/O commands use the private I/O bus. Memory, containing the 8089's programs, as well as I/O devices, may be connected to the private I/O bus. Taking this approach can greatly reduce the 8089's use of the system bus as most memory and I/O accesses can be made to the private address space. The system bus is thus made available for use by other processors, and the 8089 can execute in parallel with other processors for extended periods. A limited private I/O bus may be implemented using the 8-bit multiplexed peripherals of the 8085 family, eliminating the latches and transceivers shown in figure 1-6.

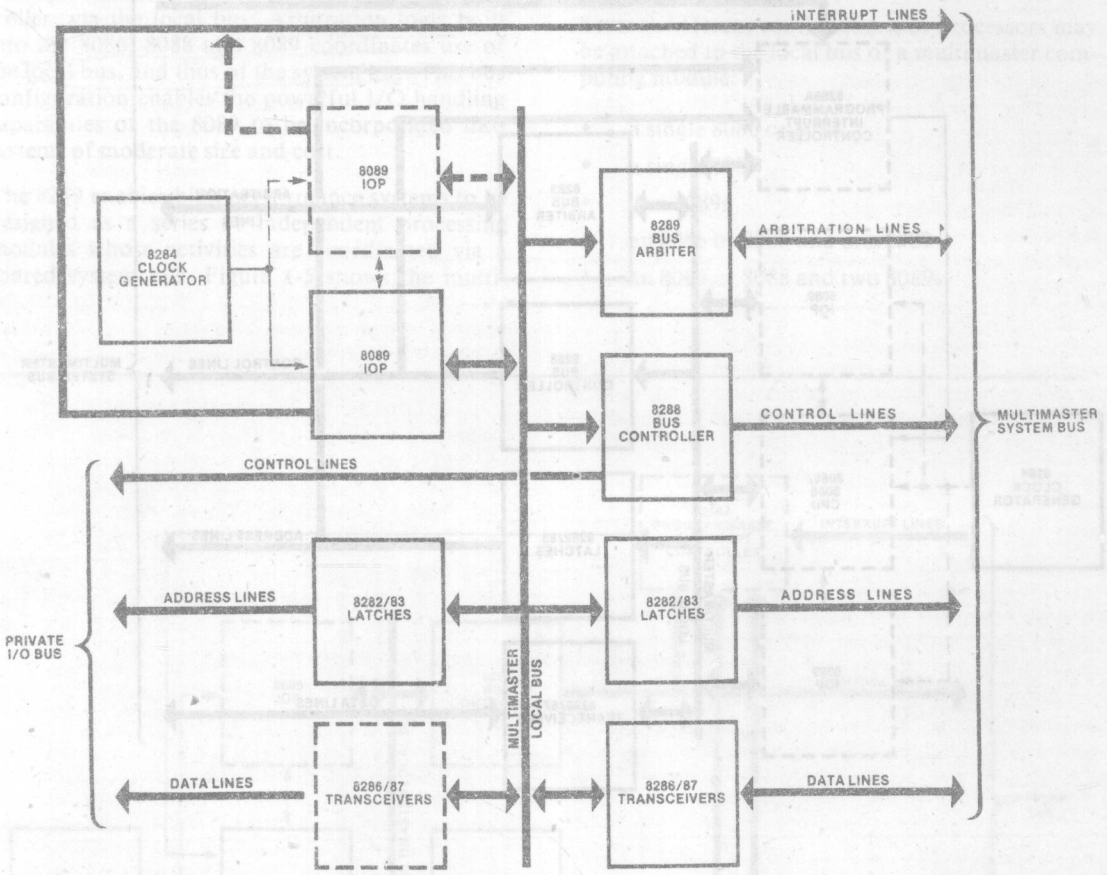

Figure 1-6. Private I/O Bus

Adding a second 8288 to the local bus allows an 8086 or 8088 in a processing module to divide its address space into system and resident sections (figure 1-7). A PROM or decoder is used to direct an address reference to the system bus or to the resident bus. The resident bus allows the CPU to run out of its own address space to minimize its use of the system bus. Since no other processors can access the private memory on the CPU's resident bus, operating system code and data in this space is protected from errors in other processor programs. If a second 8289 is added to a resident bus module, the resident bus becomes a second multimaster system bus.

此为试读，需要完整PDF请访问：www.ertongbook.com