INTRODUCTION TO THE

INTEL386™ SL MICROPROCESSOR SUPERSET

TECHNICAL OVERVIEW

LITERATURE

To order Intel Literature or obtain literature pricing information in the U.S. and Canada call or write Intel Literature Sales. In Europe and other international locations, please contact your ***local*** sales office or distributor.

INTEL LITERATURE SALES
P.O. BOX 7641
Mt. Prospect, IL 60056-7641

In the U.S. and Canada
call toll free
(800) 548-4725
This 800 number is for external customers only.

CURRENT HANDBOOKS

Product line handbooks contain data sheets, application notes, article reprints and other design information. All handbooks can be ordered individually, and most are available in a pre-packaged set in the U.S. and Canada.

TITLE	INTEL ORDER NUMBER	ISBN
SET OF THIRTEEN HANDBOOKS (Available in U S and Canada)	**231003**	**N/A**
CONTENTS LISTED BELOW FOR INDIVIDUAL ORDERING:		
COMPONENTS QUALITY/RELIABILITY	210997	1-55512-132-2
EMBEDDED APPLICATIONS	270648	1-55512-123-3
8-BIT EMBEDDED CONTROLLERS	270645	1-55512-121-7
16-BIT EMBEDDED CONTROLLERS	270646	1-55512-120-9
16/32-BIT EMBEDDED PROCESSORS	270647	1-55512-122-5
MEMORY PRODUCTS	210830	1-55512-117-9
MICROCOMMUNICATIONS	231658	1-55512-119-5
MICROCOMPUTER PRODUCTS	280407	1-55512-118-7
MICROPROCESSORS	230843	1-55512-115-2
PACKAGING	240800	1-55512-128-4
PERIPHERAL COMPONENTS	296467	1-55512-127-6
PRODUCT GUIDE (Overview of Intel's complete product lines)	210846	1-55512-116-0
PROGRAMMABLE LOGIC	296083	1-55512-124-1
ADDITIONAL LITERATURE: (Not included in handbook set)		
AUTOMOTIVE HANDBOOK	231792	1-55512-125-x
INTERNATIONAL LITERATURE GUIDE (Available in Europe only)	E00029	N/A
CUSTOMER LITERATURE GUIDE	210620	N/A
MILITARY HANDBOOK (2 volume set)	210461	1-55512-126-8
SYSTEMS QUALITY/RELIABILITY	231762	1-55512-046-6

intel®

U.S. and CANADA LITERATURE ORDER FORM

NAME: ______________________________

COMPANY: ______________________________

ADDRESS: ______________________________

CITY: ____________________ STATE: ________ ZIP: ________

COUNTRY: ______________________________

PHONE NO.: (   ) ______________________________

ORDER NO.	TITLE	QTY.		PRICE		TOTAL
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	
□□□□□□			×		=	

Subtotal ________

Must Add Your
Local Sales Tax ________

Include postage:
Must add 15% of Subtotal to cover U.S. and Canada postage. (20% all other.)

→ Postage ________

Total ________

Pay by check, money order, or include company purchase order with this form ($100 minimum). We also accept VISA, MasterCard or American Express. Make payment to Intel Literature Sales. Allow 2-4 weeks for delivery.

□ VISA □ MasterCard □ American Express Expiration Date ______________________

Account No. ______________________________

Signature ______________________________

Mail To: Intel Literature Sales
P.O. Box 7641
Mt. Prospect, IL 60056-7641

International Customers outside the U.S. and Canada should use the International order form on the next page or contact their local Sales Office or Distributor.

For phone orders in the U.S. and Canada
Call Toll Free: (800) 548-4725

Prices good until 12/31/91
Source HB

INTERNATIONAL LITERATURE ORDER FORM

NAME: ______________________________

COMPANY: ______________________________

ADDRESS: ______________________________

CITY: ____________________ STATE: ________ ZIP: ________

COUNTRY: ______________________________

PHONE NO.: (   ) ______________________________

ORDER NO.	TITLE	QTY.		PRICE		TOTAL
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____
□□□□□□	____________	____	×	_____	=	_____

Subtotal ________

Must Add Your
Local Sales Tax ________

Total ________

PAYMENT

Cheques should be made payable to your ***local*** Intel Sales Office (see inside back cover).

Other forms of payment may be available in your country. Please contact the Literature Coordinator at your ***local*** Intel Sales Office for details.

The completed form should be marked to the attention of the LITERATURE COORDINATOR and returned to your ***local*** Intel Sales Office.

intel®

Intel 386™ SL
MICROPROCESSOR SUPERSET

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice

Contact your local sales office to obtain the latest specifications before placing your order.

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

> 376, Above, ActionMedia, BITBUS, Code Builder, DeskWare, Digital Studio, DVI, EtherExpress, ETOX, FaxBACK, Grand Challenge, i, i287, i386, i387, i486, i487, i750, i860, i960, ICE, iLBX, inboard, Intel, Intel287, Intel386, Intel387, Intel486, Intel487, Intel inside., Intellec, iPSC, iRMX, iSBC, iSBX, iWARP, LANPrint, LANSelect, LANShell, LANSight, LANSpace, LANSpool, MAPNET, Matched, MCS, Media Mail, NetPort, NetSentry, OpenNet, PRO750, ProSolver, READY-LAN, Reference Point, RMX/80, SatisFAXtion, Snapin 386, Storage Broker, SugarCube, The Computer Inside., TokenExpress, Visual Edge, and WYPIWYF

MDS is an ordering code only and is not used as a product name or trademark, MDS is a registered trademark of Mohawk Data Sciences Corporation.

CHMOS and HMOS are patented processes of Intel Corp.

Intel Corporation and Intel's FASTPATH are not affiliated with Kinetics, a division of Excelan, Inc. or its FASTPATH trademark or products.

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

Intel Corporation
Literature Sales
P.O. Box 7641
Mt. Prospect, IL 60056-7641

OG-072391

INTEL SERVICE

INTEL'S COMPLETE SUPPORT SOLUTION WORLDWIDE

Intel Service is a complete support program that provides Intel customers with hardware support, software support, customer training, and consulting services. For detailed information contact your local sales offices.

Service and support are major factors in determining the success of a product or program. For Intel this support includes an international service organization and a breadth of service programs to meet a variety of customer needs. As you might expect, Intel service is extensive. It can start with On-Site Installation and Maintenance for Intel and non-Intel systems and peripherals, Repair Services for Intel OEM Modules and Platforms, Network Operating System support for Novell NetWare and Banyan VINES software, Custom Integration Services for Intel Platforms, Customer Training, and System Engineering Consulting Services. Intel maintains service locations worldwide. So wherever you're using Intel technology, our professional staff is within close reach.

ON-SITE INSTALLATION AND MAINTENANCE

Intel's installation and maintenance services are designed to get Intel and Intel-based systems and the networks they use up and running—fast. Intel's service centers are staffed by trained and certified Customer Engineers throughout the world. Once installed, Intel is dedicated to keeping them running at maximum efficiency, while controlling costs.

REPAIR SERVICES FOR INTEL OEM MODULES AND PLATFORMS

Intel offers customers of its OEM Modules and Platforms a comprehensive set of repair services that reduce the costs of system warranty, maintenance, and ownership. Repair services include module or system testing and repair, module exchange, and spare part sales.

NETWORK OPERATING SYSTEM SUPPORT

An Intel software support contract for Novell NetWare or Banyan VINES software means unlimited access to troubleshooting expertise any time during contract hours—up to seven days per week, twenty-four hours per day. To keep networks current and compatible with the latest software versions, support services include access to minor releases and "patches" as made available by Novell and Banyan.

CUSTOM SYSTEM INTEGRATION SERVICES

Intel Custom System Integration Services enable resellers to order completely integrated systems assembled from a list of Intel386™ and Intel486™ microcomputers and validated hardware and software options. These services are designed to complement the reseller's own integration capabilities. Resellers can increase business opportunities, while controlling overhead and support costs.

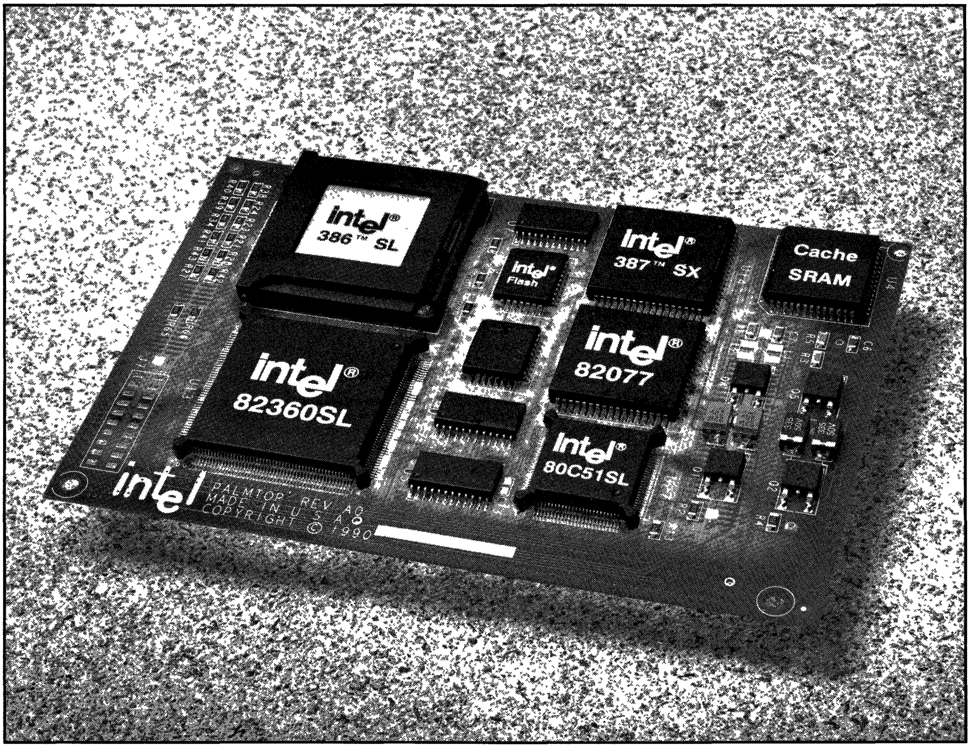
CUSTOMER TRAINING

Intel offers a wide range of instructional programs covering various aspects of system design and implementation. In just three to five days a limited number of individuals learn more in a single workshop than in weeks of self-study. Covering a wide variety of topics, Intel's major course categories include: architecture and assembly language, programming and operating systems, BITBUS™, and LAN applications.

SYSTEM ENGINEERING CONSULTING

Intel provides field system engineering consulting services for any phase of your development or application effort. You can use our system engineers in a variety of ways ranging from assistance in using a new product, developing an application, personalizing training and customizing an Intel product to providing technical and management consulting. Working together, we can help you get a successful product to market in the least possible time.

TABLE OF CONTENTS



Intel386™ SL Microprocessor SuperSet Delivers Full ISA Compatibility in the Smallest Possible Footprint

PREFACE

The Intel386™ SL Microprocessor SuperSet is the newest addition to the Intel386 32-bit microprocessor product line. Contained within just two VLSI components are virtually all of the critical elements needed for a range of microcomputer-based products, from checkbook-sized "palm-top" personal computers and notebook computers to full-featured, fully-expandable lap-top portables.

The Intel386 SL Microprocessor SuperSet includes two components which provide: the Intel386 SL central processing unit that is fully compatible with the Intel386 microprocessor architecture, main memory and bus interface control system; and the 82360SL peripheral I/O subsystem.

The components bring a number of new design techniques to the personal computer world for the first time. For instance:

- Extensions to the processor architecture support a new system-management mode with which OEM firmware can monitor and control hardware operation and conserve battery power without sacrificing Intel386 software compatibility
- Processor and peripheral clock frequencies can be adjusted dynamically to conserve power without sacrificing overall system performance
- Each component's internal design is fully static so the clock can be stopped without losing critical data
- SmartHit cache control logic boosts performance and saves power by eliminating unneeded memory transfers
- FlexLogic circuitry lets software control all operating characteristics and decouple system behavior from hardware design
- An uncommitted ideaPort interface lets original equipment manufacturers develop unique new value-added functions transparent to existing software

The components contain more than 1.5 million transistors in all and are fabricated from Intel's one-micron CHMOS-IV technology. While SL SuperSet-based systems benefit from the highest possible integration level, they also deliver significantly better performance than is possible with conventional technology.

The Intel386 SL SuperSet components comply with all Industry Standard Architecture (ISA) design conventions and are thus fully compatible with the IBM PC/AT personal computer and its derivatives. Systems built with the Intel SL SuperSet can therefore run all existing business and scientific applications developed for the 8086, 80286, 80386, and 80486 microprocessors. The SL SuperSet likewise supports all popular microprocessor operating systems developed for the original 8086 and Intel386 Family architecture, including MS-DOS†, OS/2**, Windows**, Windows-386**, and UNIX†† System V.

Moreover, each component reduces power requirements to an absolute minimum, not just for the chips themselves but for the main memory, display interface, and other system components. Reductions in run-time power in turn extend the useful operating life of battery-operated portable computers, and automatic built-in stand-by modes stretch the battery life of idle systems still further.

Manual Objectives and Organization

This manual serves as a basic introduction and overview for the full Intel386 SL Microprocessor SuperSet family, and is intended to help hardware and software engineers and managers evaluate the suitability of the products for personal computers and other application areas.

Since the Intel386 SL SuperSet combines the functions of an entire computer system into just a few components, different readers will view the Intel SuperSet from different perspectives. Hardware engineers will want to understand how the products connect to other system components. Programmers developing basic I/O system (BIOS) software must understand the capabilities of the on-chip peripheral registers in order to initialize the PC with the desired characteristics.

Marketing managers may wish to understand how price reductions allowed by slower memories and support components affect overall system performance, and how Intel SL SuperSet hardware lets them augment built-in functions with external circuitry for greater flexibility and product differentiation. Those concerned with the problems of allocating a limited battery-power budget among system resources and peripherals must understand the power-management options and capabilities supported by each chip.

In order to address different reader perspectives, this manual is divided into five chapters. **Chapter 1** is an overview of product-line highlights and surveys a range of personal computer configurations that will benefit from various SL SuperSet features. **Chapter 2** discusses system hardware design, including the functions performed by each SL SuperSet component and how the components interact with other system elements.

Intel386™ SL, Intel386™, and Intel486™ are trademarks of Intel Corporation
CHMOS-IV is a patented process technology of Intel Corporation
† MS-DOS® is a registered trademark of Microsoft Corporation.
** OS/2™, Windows, and Windows-386 are trademarks of Microsoft Corporation.
†† UNIX® is a registered trademark of AT&T.

Chapter 3 covers the processor architecture and memory-address mapping facilities, and explores software design issues related to the on-chip peripheral facilities and configuration options.

Chapter 4 goes into greater depth explaining the new system management mode, especially as it relates to power management, and discusses its special software requirements. The block diagrams in **Chapter 5** show example system configurations supported by the Intel SL SuperSet, and shows how individual OEMs can add unique functions to the Intel SL SuperSet to expand its end-user capabilities.

Related Publications

This manual gives only a glimpse of the architecture and peripheral capabilities of the Intel386 SL Microprocessor SuperSet, and makes no attempt to describe its underlying instruction set, electrical specifications, or timing parameters. For design information of this type consult the following related publications:

- *Intel386™ SL Microprocessor SuperSet System Design Guide*, Intel Order No. 240816
- *Intel386™ SL Microprocessor SuperSet Programmer's Reference Manual*, Intel Order No. 240815
- *Intel386™ SL Microprocessor SuperSet Data Sheet*, Intel Order No. 240814
- *Intel386™ Microprocessor Software Writer's Guide*, Intel Order No. 231499

Chapter 1
Product Highlights

CHAPTER 1
PRODUCT HIGHLIGHTS

The personal computer industry has expanded in ways its creators never expected. On the one hand, designers have converged on a well-defined set of processor, memory, peripheral, and system backplane functions that are now part of all industry-standard personal computers (see Figure 1-1). On the other hand, the simple, conventional desktop computer systems of the mid '80s have evolved into a bewildering array of physical configurations. New peripheral devices and new system capabilities continue to augment the basic functions of early PCs.

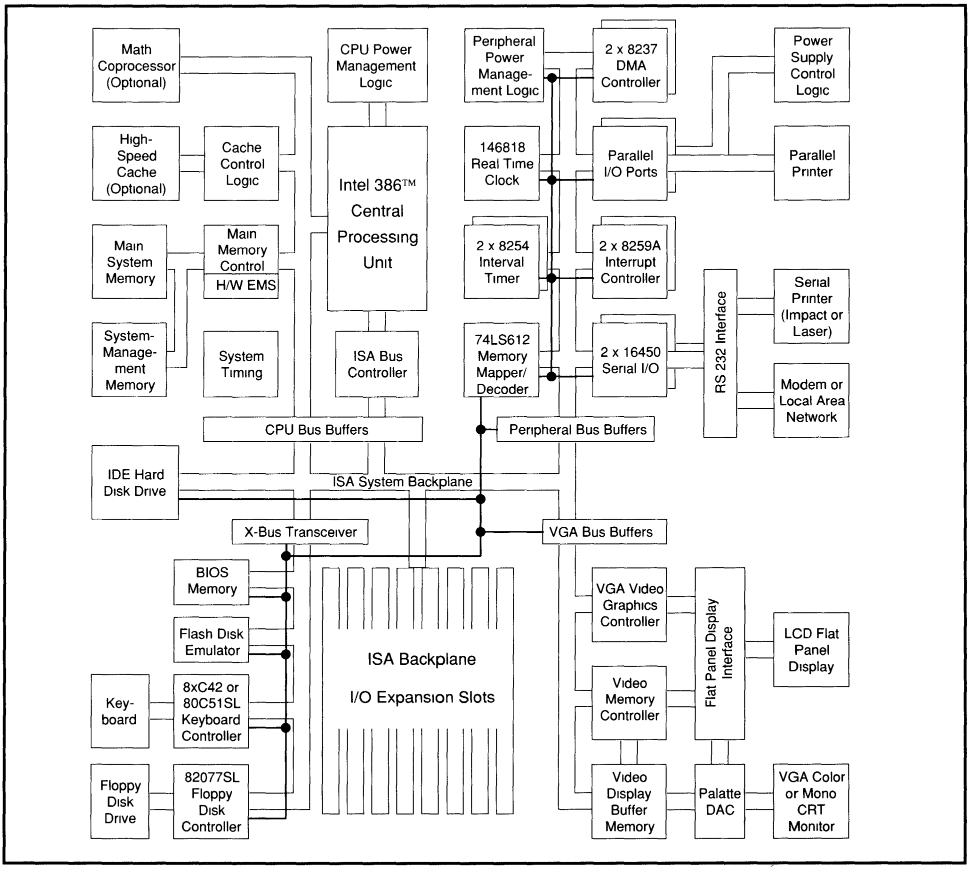


Figure 1-1. Industry-Standard PCs Contain a Common Set of System Functions and Peripherals

While the performance delivered by high-end PCs continues to soar, the market now demands a range of alternative designs, at different price points, in different form-factors. Satisfying this demand poses a whole new set of challenges to today's microcomputer hardware and software engineers.

Technological Challenges of Modern PCs

Standard microcomputers no longer seem as "micro" as they once did. When PCs must compete for desk space with telephones, modems, facsimile machines, and other desk-top clutter, it's important to keep the PC footprint small.

This is especially true for portable PCs. If it's bigger than a bread box it's no longer in vogue: today's users prefer systems that fit a briefcase or coat pocket. Shrinking enclosures demand ever-smaller circuit boards, so the components used must be the most highly integrated devices achievable within a given state of the art.

The challenge posed by battery-powered portable systems is doubly severe. Not only are size constraints tight, but power conservation is critical. Battery size and capacity are limited, so the less power the PC consumes, the longer it can go between charges. Unfortunately, past efforts to include power-management capabilities have had mixed success: ports, programs, and new resources needed for power-management functions unavoidably introduced a hazard that existing software might malfunction on the revised hardware.

Efforts to reconcile the competing factors of performance, size, power consumption, and software compatibility have created new opportunities and market niches for a remarkable variety of PC configurations. In order to differentiate their products from the competition, PC designers seek to provide unique features such as innovative hardware design, effective packaging technology, fast and convenient mass storage, or more natural user interfaces (NUIs).

A mouse, track-ball, or note-pad stylus for handwritten data, or other novel input devices often augment the standard keyboard, while circuitry for speech recognition and gesture detection may allow more intuitive control paradigms. Thus the components from which modern PCs are built must leave the hardware designer complete flexibility in adapting and extending his product with unique functionality.

At the same time, however, modern PCs must guarantee their users absolute compatibility with the wealth of applications software developed since the PC business began. This compatibility must be assured at all levels of hardware and software: within the processor itself, its memory control systems, the peripherals it supports, and the bus interface through which third-party add-in boards expand system functions.

The Intel386 SL Microprocessor SuperSet

The Intel386 SL Microprocessor SuperSet (Figure 1-2) addresses each of the above concerns. At its heart is a full-featured 32-bit central processor unit that is fully compatible with the industry-standard Intel386 microprocessor architecture. Built into the CPU are a main-memory subsystem controller with a large address space and extensive address translation and remapping logic, tag registers and control logic needed for a cache memory system, and support for an optional Intel387™ SX Math CoProcessor.

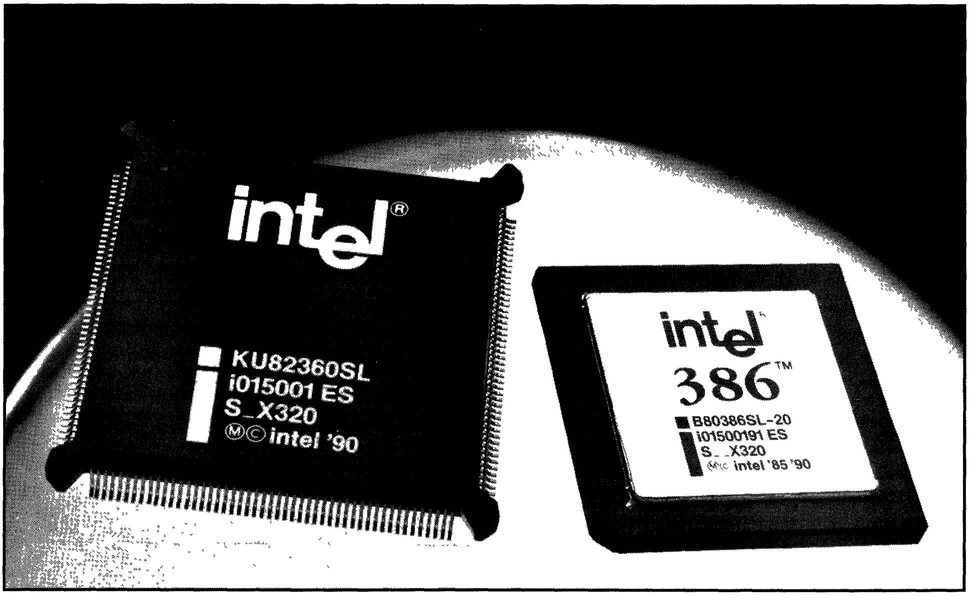


Figure 1-2. The Intel386™ SL Microprocessor SuperSet Includes Two Highly-Integrated Components

Supporting the processor are a host of general-purpose peripheral interface devices, including serial ports, parallel ports, timers, interrupt controllers, keyboard and disk interfaces. Special power-management circuitry within each chip makes the most efficient use possible of limited power supplies.

System Design Options

The Intel386 SL Microprocessor SuperSet gives hardware designers a tremendous degree of flexibility, as shown in Table 1-1. The main system memory and display buffers can be built with static (SRAM) or dynamic (DRAM) memory devices of various sizes and speeds, with optional cache subsystems and non-volatile Flash EPROM arrays. End users can add memory in various combinations and configurations, typically by inserting single in-line memory modules (SIMMs) or credit-card-style expansion modules. The SL SuperSet can detect the size and configuration of installed memory during system initialization, and software can then adjust address multiplexing, control, and interleave schemes accordingly.

Table 1-1. The Intel386™ SL Microprocessor SuperSet Leaves OEMs a Choice of System Design Options

Feature	Design Options
Central Processing Unit	Real Mode, 286 Protected Mode, 32-Bit Intel386 Protected Mode, Operating Modes
Virtual 8086 Mode	
CPU Frequency Control	25 MHz (Maximum); Reduced Frequency (Maximum ÷2, ÷4, or ÷8); or Fully-Static (Stopped Clock)
Math Coprocessor Support	Interface for Optional Intel387 SX Math CoProcessor
Coprocessor Frequency Control	25 MHz Maximum Frequency ÷1, ÷2, ÷4, ÷8, or ÷16
Main Memory Type	Static (SRAM) or Dynamic (DRAM)
Main Memory Capacity	512 Kilobytes to 20 Megabytes
Cache Memory Support	None, 16K, 32K, or 64K bytes; Direct Mapped, Two-Way, or Four-Way Set Associative
Memory Expansion	Standard 386 CPU Paged Memory Management, plus Hardware Support for LIM 4.0 Memory Expansion Standard
Main Memory Parity Support	Optional Parity Generation and Verification Logic On-Chip
Main Memory Enhancements	Optional Automatic EPROM Shadowing, Backfill, and Roll-Over
BIOS Memory Support	ROMs, Conventional EPROMs, or Flash EPROMs; 64K–128K bytes; 8- or 16-bit Bus Width; 0–15 wait states;
Mass Storage Support	Floppy Disk Controller Interface; IDE Hard Disk Interface; and Flash EPROM Disk Emulation
Video Graphics Device Types	Monochrome STN LCD panels, TFT LCD Displays
Video Graphics Standards	IBM VGA Standard
DMA Controller Frequency	Bus Clock Frequency ÷1, ÷2, or Stopped Clock
Keyboard Controller Frequency	System Clock Frequency ÷2, ÷4, or Stopped Clock
Power Conservation Modes	Fully-Programmable: Processors and peripherals can be individually configured for an arbitrary range of power-conservation modes, from Full-speed/Full-power execution to Low-frequency or Stopped-clock/Ultra-low-power mode. OEM-developed firmware can institute individual CPU , memory, and peripheral power-management policies based on system-wide power-usage patterns, computational load, and user preferences

The SL SuperSet includes all of the peripheral devices now standard in ISA PCs, including serial and parallel ports, interrupt and DMA controllers, timers, and so forth. And, the forthcoming display subsystem is compatible with the IBM VGA standard using standard monochrome STN LCD panels and color TFT LCD displays. The SL SuperSet supports solid-state Flash EPROM disk emulators in addition to rotating media for conventional diskettes and hard drives. Using Intel's FlexLogic configuration scheme, each such peripheral can be enabled, disabled, or reconfigured in various operating modes as needed, all under software control.

Clock Frequency Options

The Intel386 SL Microprocessor SuperSet components allow clock frequencies up to 25 MHz. When battery life is more critical than performance, the clock can be divided by a software-selectable factor to produce a slower, more power-efficient effective CPU frequency. Memory, I/O system, and coprocessor clock frequencies are all likewise software controlled, independent of each other and of the main CPU.

Portable systems can therefore conserve power by running at slower speeds during long periods of text entry, database downloading, or activities that take little computational power. During periods of intense computation, such as searching, sorting, or cataloguing a database, or recomputing a large spreadsheet, the system can switch back to its fastest CPU frequency, for what is sometimes called "turbo" operation.

The number of CPU cycles taken for each memory or expansion-bus access is software adjustable to compensate for different CPU operating frequencies. An SRAM that needs several wait states at the highest CPU frequency, for example, may run without wait states when the clock divides by four. Thus the CPU clock can be slowed without degrading system performance proportionately.

Performance Factors

While hardware designers worry about physical system size, complexity, and power consumption, end users are more concerned with system throughput. The SL SuperSet's central processor and all of its supporting circuitry provide designers with the benefits of high integration and low power consumption without sacrificing the end-user performance expected of 386 microprocessor-based PCs.

Specially-tuned control logic and the cache memory subsystem let the SL SuperSet use relatively slow, low-cost memory components without sacrificing performance. Systems that manipulate extensive scientific or engineering data can add an optional floating-point math coprocessor. The coprocessor operating frequency can be adjusted under software control according to its computational load. To save further power, the coprocessor can be configured separately from the main CPU, or disabled entirely when not in use.

Guaranteed Compatibility

Each Intel386 SL Microprocessor SuperSet component complies with all industry-standard PC design conventions. Extensions to the basic CPU architecture provide facilities for configuring system hardware, switching modes, and managing system power consumption, totally transparent to existing software. The new System Management Mode provides a new, dedicated interrupt vector and a new control instruction for supervisory control and power management functions. Software and data structures needed to support these functions all reside in a memory partition that is inaccessible to conventional software. Unlike conventional design approaches, SL SuperSet-based systems do not depend on "terminate and stay resident" (TSR) routines that can interfere with existing application programs.

While the SL SuperSet includes a number of new capabilities for power-reduction, each circumvents the hazards found in conventional design techniques. Each new capability is supported by an extension to the basic Intel386 architecture, rather than by appropriating some aspect of the original architecture that may or may not be needed for a particular piece of software. All of the resources of the original 8086, 80286, and Intel386 Family processors are thus preserved intact, ensuring full compatibility with all existing application programs and operating systems.

Systems built with these products can therefore be made fully compatible at both the hardware and software levels with all existing business and scientific applications programs developed for the 8086, 80286, and Intel386 Family microprocessors and all of the 8086 and Intel386 microprocessor operating systems now in use, including MS-DOS, OS/2, Windows 3.0, Windows-386, and UNIX System V.

The Expanding Horizons of PC Design

The SL SuperSet is versatile enough to be used in personal computers spanning a spectrum of hardware configurations. At one extreme might be the tiny, palm-top PC shown in Figure 1-3. Computers with this form-factor have compact keyboards, small LCD matrix displays, ROM or Flash EPROM-based operating-system and applications software, and a moderate amount of data memory, typically implemented using low-power static (SRAM) devices. Operating power is supplied by replaceable penlight cells or a small rechargeable battery.

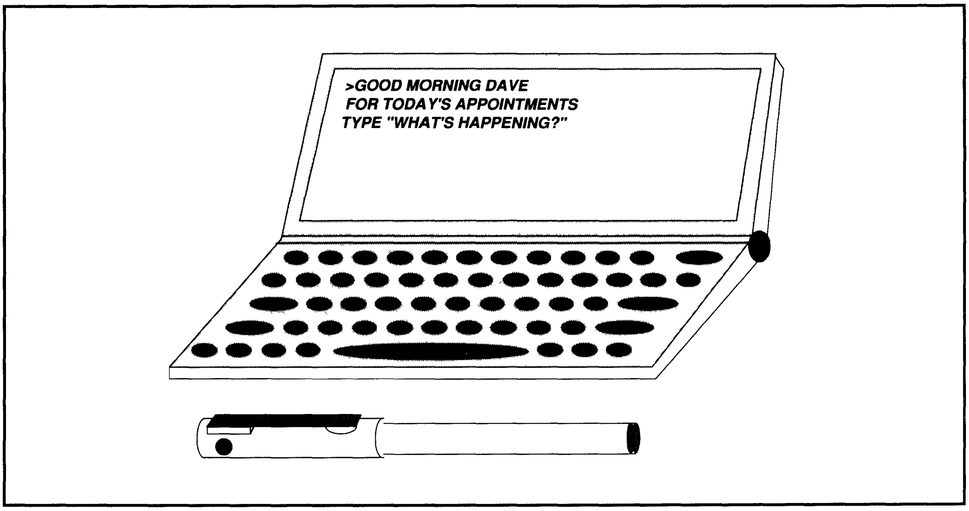


Figure 1-3. "Palm-Top" Personal Computer Folds to Fit Within a Purse or Shirt Pocket

Such a product may connect to larger host computers through a serial cable or modem for transferring program and data files, and may augment its memory capacity via credit-card-sized expansion boards. A step above palm-top PCs are small note-pad-sized computers with larger displays and easier-to-use keyboards. In both markets, however, physical size, weight, cost, and power preservation factors outweigh raw performance and expandability concerns.

At the other end of the spectrum are conventional desk-top computers with high-resolution color CRT displays, full-function keyboards, and mouse input devices (see Figure 1-4). With circuit-board area and power consumption at less of a premium, such systems may include internal high-capacity hard disks, built-in modems, a local-area network interface, and expansion connectors for a range of serial and parallel printers, optical scanners, and external CD-ROM disk drives.

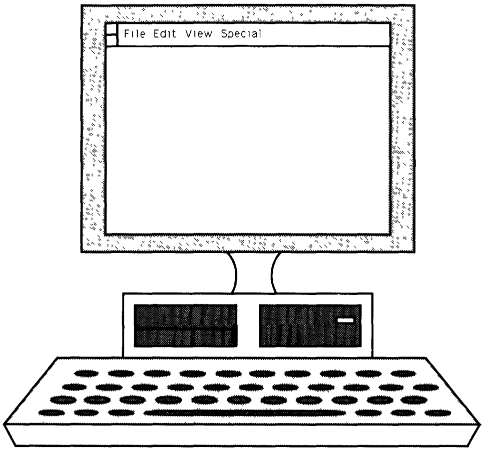


Figure 1-4. Typical Desk-Top PC Configuration

While desktop configurations also benefit from a small design footprint and low entry-level system cost, it's critical that their design not be limited with respect to performance, upgradeable memory capacity, and compatibility with off-the-shelf applications software and standard I/O expansion boards. The main memory in such configurations is typically DRAM based, with from 1 to 32 megabytes. High-speed SRAM caches may also be included to improve the performance of DRAM main memory.

Intermediate Design Options

Between these extremes one can envision a tremendous variety of design options (see Figure 1-5). Today's portable computers range in size from "lap-top" to "luggable". Power supplies support both plug-in and battery-powered operating modes. Display technologies range from simple LCD displays to large monochrome and color CRTs. Mass storage options include some combination of internal floppy-disk drives, hard disks, or solid-state disk emulators.

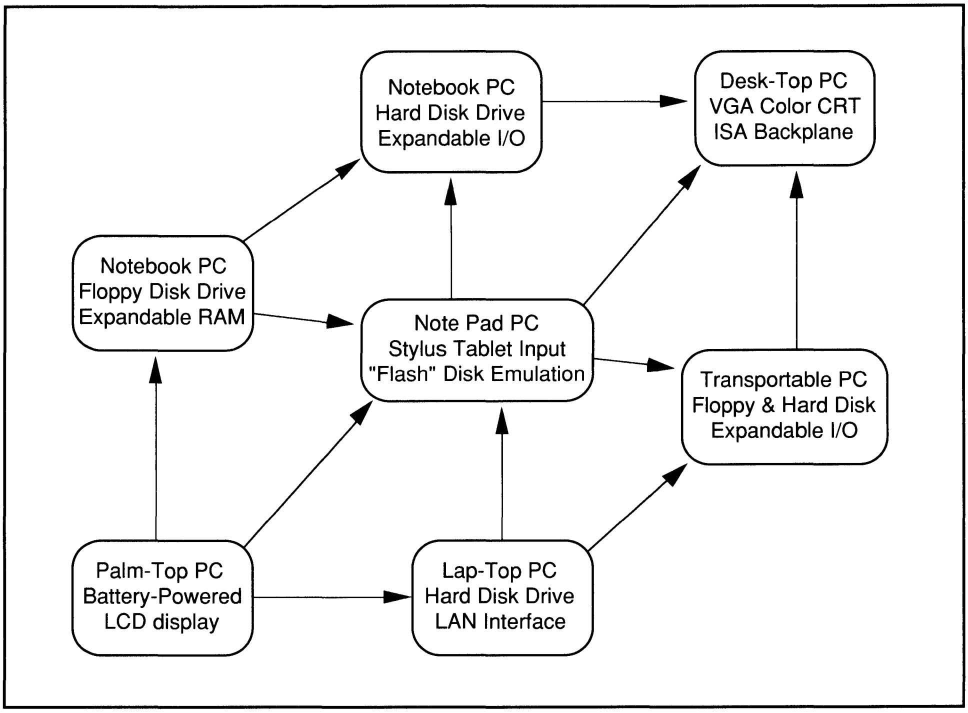


Figure 1-5. The Intel386™ SL Microprocessor SuperSet is Designed for Applications from Battery-Powered Palm-Top PCs to Fully-Expandable Desk-Top Systems

All PCs, however, share the a need for speed and software flexibility. Today's graphical user interfaces (GUIs) and newly-emerging "natural user interfaces" (NUIs) demand the 32-bit precision, higher performance, expanded address range, and software protection mechanisms provided by the Intel386 architecture.

Other features of the SL SuperSet benefit PC configurations throughout the size, price, and complexity continuum. High integration is vital to palm-top designs, but can also shrink desk-top systems. Reduced power demands extend battery life, but can also reduce the size and cost of A.C. power supplies and eliminate cooling fans. Extensions to the CPU architecture can provide increased data security and improve system integrity. Systems of all sizes built with the SL SuperSet can deliver significantly better performance than those that use less sophisticated technology.

A New Class of "Hybrid" PCs

Each system shown in Figure 1-5 can be built with conventional technology, of course, if one is willing to make compromises in price, performance, and power consumption. Designers can optimize whichever factors they choose for a particular configuration at the hardware level by making corresponding trade-offs in control logic and component selection.

The Intel386 SL Microprocessor SuperSet family, however, uses the same components for all PC form factors and levels of complexity, with power and performance trade-offs accomplished by software. FlexLogic control circuitry enables functions appropriate to each configuration. CPU operating frequencies can be selected, memory control algorithms can be refined, and peripheral modes can be enabled dynamically, according to computational load, in order to optimize both performance and battery life.

The ability to adjust the behavior of fixed hardware through software makes it possible to build a new class of "hybrid" PCs (see Figure 1-6) that function in multiple modes. For home or office use such a system would behave like a conventional office PC; i.e., it would have a full-size color CRT, keyboard, and mouse, include diskette and high-capacity hard-disk drives, attach to a telephone jack or local-area network, have slots for standard I/O expansion boards, and draw A.C. power from the wall. In desk-top mode this system would continuously deliver the uncompromised performance levels users expect from 386-microprocessor-class PCs.

If the computer was needed elsewhere, however, a small "compute engine" module consisting of the central processor and main memory and a solid-state disk emulator could detach from its desk-top docking module and become a fully-functional stand-alone unit, with its own integrated keyboard, flat-panel LCD display, and internal battery power. Space and power considerations might change somewhat the way the portable configuration operates—its system clock rate, memory size, or mass storage capacity may be reduced, or it may run a special ROM-based version of the host operating system—but each configuration would use the same processor and basic system memory, run the same utilities and applications programs, and retrieve and update the same text and data files.

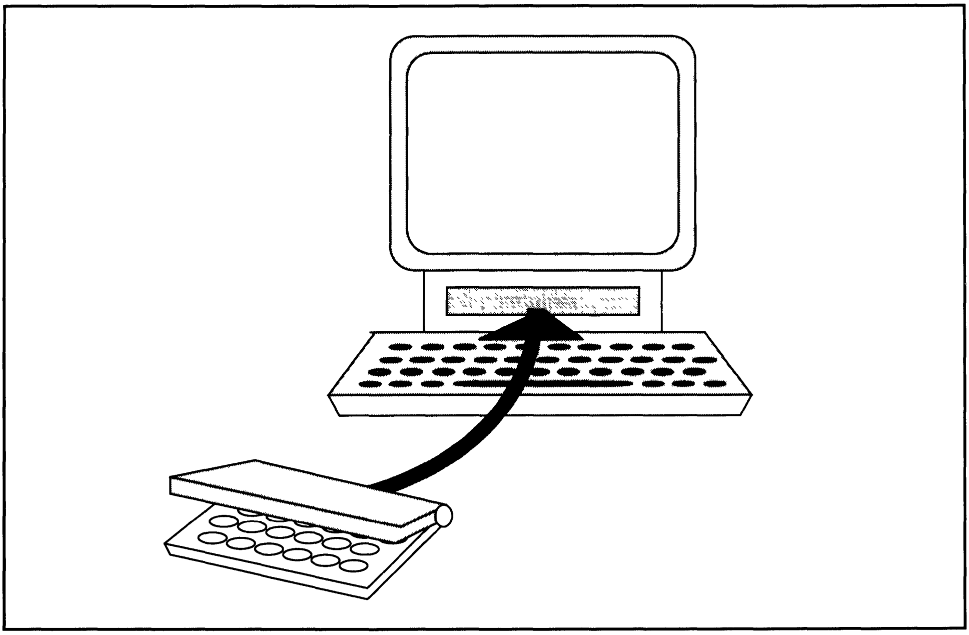


Figure 1-6. Stand-Alone Portable Module Acts as the Compute Engine of a Hybrid PC Design

Product Highlights Summary

As portable, battery-powered personal computers come into widespread use, microprocessors must provide new features and facilities tailored to such systems. Such systems require the highest level of system integration and the minimum possible power consumption, while sustaining the performance and flexibility users have come to expect from the Intel386 microprocessor family. The Intel386 SL Microprocessor SuperSet is the first pair of fully-integrated 32-bit personal computer chips designed expressly for portable computers, and provides system hardware and software designers with a wide range of options with which they can develop PC products that would be difficult or impossible to build with conventional technology.

Chapter 2
Hardware System Architecture

CHAPTER 2
HARDWARE SYSTEM ARCHITECTURE

The Intel386 SL Microprocessor SuperSet lets personal computers take advantage of the highest possible level of system integration, while at the same time preserving complete freedom in selecting a combination of system features, power/performance trade-offs, and value-added enhancements.

The SL SuperSet combines essentially all of the components needed to build an industry-standard personal computer within just two components: the Intel386 SL central processor, main memory, and bus interface control system; and the 82360SL peripheral component subsystem. The only other components needed for a complete, working PC are various memory arrays, a keyboard interface, graphics controller, and electrical buffers for certain optional peripherals.

This chapter first reviews the general functions performed by each of the SL SuperSet components, and then discusses some of the options and degrees of flexibility allowed in the design of the various memory arrays and display subsystem.

Functional Partitioning

Intel SL SuperSet-based products will generally contain each of the functional blocks shown in Figure 2-1. The functions performed by each block are described briefly below.

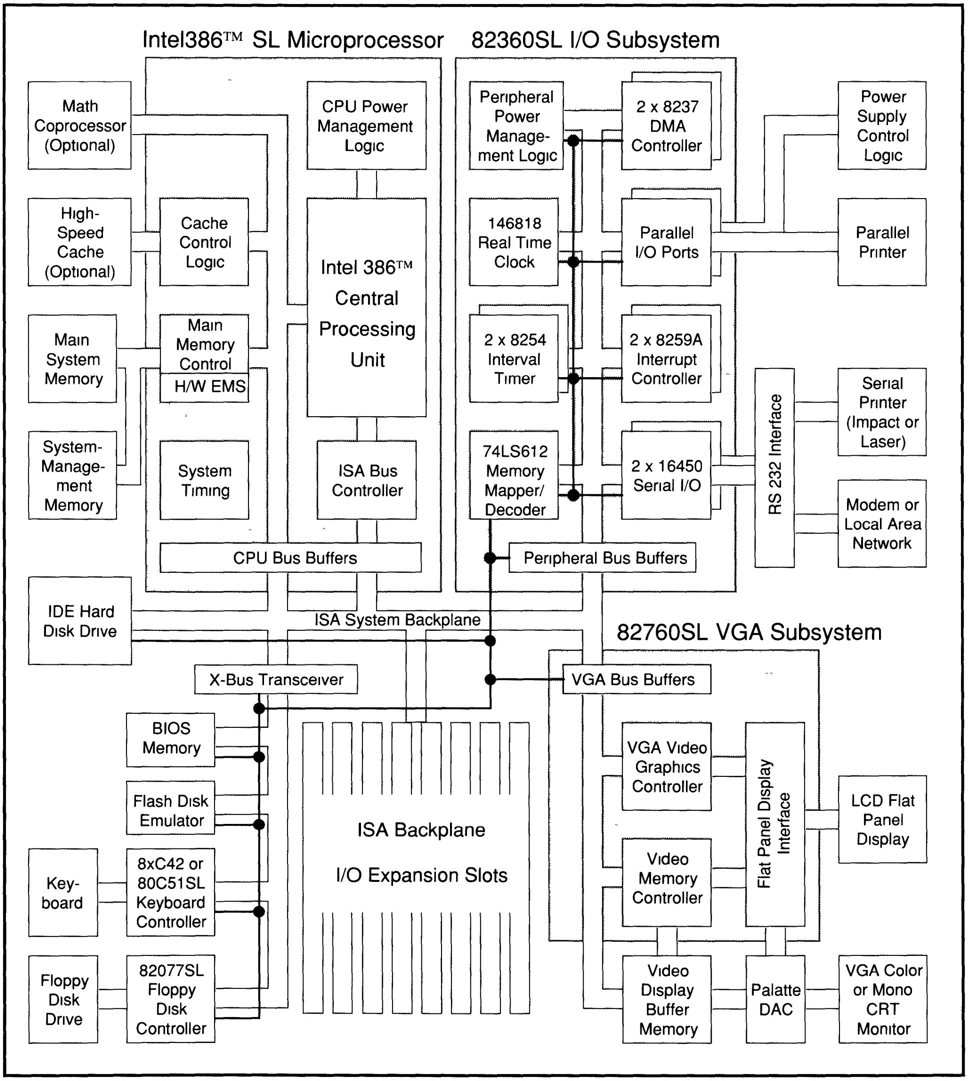


Figure 2-1. Intel 386™ SL Microprocessor SuperSet Family Components Bring Together All Critical PC Functions

Intel386 SL Microprocessor Functions

The Intel386 SL Microprocessor is highly integrated, both in its transistor count and the functions it performs. At the heart of the device is a central processing unit (CPU) compatible with the Intel386microprocessor. The CPU has been fully re-engineered for low-power static (stopped-clock) operation, and its architecture has been augmented with extensions needed for general-purpose system-management functions, as described in **Chapter 4**.

The CPU contains the same paged-memory-management system that is standard on all Intel386microprocessors. In addition, on-chip hardware implements an expanded memory system (EMS) compatible with the Lotus/Intel/Microsoft (LIM 4.0) standard. Address-mapping and control logic within the Intel386 SL processor can "shadow" BIOS ROMs with faster memory devices, and supports all common memory roll-over and back-filling schemes. The Intel386 SL processor contains all the control and interface logic needed to drive directly a 20-megabyte main memory system and a smaller, faster, optional cache.

The Intel386 SL processor also contains bus drivers and control circuitry for two expansion interfaces. A high-performance peripheral-interface bus communicates with devices on the same circuit board as the SL SuperSet, while an ISA-compatible "system backplane" bus communicates with industry-standard peripheral add-in boards. On-chip control logic automatically routes each memory or I/O operation to the appropriate memory array, cache, peripheral bus, or backplane, without involving the operating system or applications software in any way.

All system configuration logic in the Intel386 SL processor is initialized under software control. This FlexLogic control system may be reconfigured at run-time to match the system hardware design, the size and type of main memory and peripherals currently installed, and the end-user's power consumption and performance goals.

82360SL Peripheral I/O Subsystem Functions

The 82360SL Peripheral I/O Subsystem contains dedicated logic to perform a number of CPU and memory support functions, a complete set of input/output (I/O) peripherals including serial ports, parallel ports, timers, and interrupt and DMA controllers, and interfaces for separate keyboard, diskette, and IDE hard-disk-drive controllers. The peripheral registers can be configured under software control to operate exactly like the discrete components commonly found in industry-standard PCs. See **Chapter 3** for information on the standard peripherals and their configuration options.

Processor and memory support functions contained in the 82360SL eliminate most of the external random-logic "glue" that might otherwise be required. The 82360SL provides internal variable-frequency clock generators for the CPU, backplane, video subsystem, and external math coprocessor. These features are discussed in **Chapter 4.**

Finally, the 82360SL contains an extensive set of power-management control facilities which further reduce energy requirements of battery-powered portable computers. A special low-power DRAM refresh timer keeps the main system memory alive and refreshed when the CPU is otherwise inactive. These, too, are discussed in **Chapter 4**. All of the standard peripheral registers, clock-generation logic, and power-management facilities have been designed to ensure compatibility with all existing operating systems and applications software.

82760SL VGA Graphics Control Subsystem Functions

The 82760SL VGA Graphics Control Subsystem will implement all of the control logic needed for a complete display subsystem fully compatible at the register level with the IBM VGA standard. It will be configured to connect to either monochrome or full-color CRTs or flat-panel liquid-crystal display (LCD) technologies with resolution up to 640 x 480 pixels, or 800 x 600 pixels, and will automatically translate each of 256 colors into up to 64 gray levels for monochrome displays.

The 82760SL will include power-management circuitry that keeps track of display system usage and determines when it's safe to power-down flat-panel backlights or the display interface circuitry. These features are also discussed in **Chapter 4.**

External Memory Arrays

The block labeled "Main System Memory" in Figure 2-1 generally holds the computer's operating system software, application programs loaded from disk, and temporary data or files. The Intel386 SL processor hooks directly to either SRAM or DRAM memory devices with total capacity from 512 Kilobytes to 20 megabytes, with optional parity. Depending on the type and size of the memory devices involved, control logic generates the appropriate chip-select, bank-select, byte-enable, and read and write control signals.

The Intel386 SL processor also contains control logic for a smaller, optional memory array labeled "High-Speed Cache" in Figure 2-1. The internal control unit includes the tag bits and comparators needed for a variety of cache sizes and configurations. The cache interface requires no "glue": separate, dedicated pins drive all cache address and data buses and generate all chip-select and byte-enable control signals, so external cache systems consist of just one, two, or four memory components.

The block labeled "BIOS Memory" indicates ROM, EPROM, or Flash EPROM memory devices that hold the basic I/O system software for ISA computers. The same array may also contain system-specific initialization and configuration software, and may contain interrupt and trap-handler routines used for power-management software.

The SL SuperSet supports two additional optional memory arrays for special system functions. The "Flash Disk Emulator" block contains non-volatile memory devices used to replace or augment conventional disks and diskette drives with a solid-state file-storage system. This memory can be arbitrarily large, and can hold OS code, application programs, and important data files indefinitely, even with all power removed from the system.

The optional memory block labeled "System Management Memory" in Figure 2-1 holds code and data needed by supervisory system functions, information that would normally be concealed from OS and application programs. Portions of this memory may be implemented with Flash EPROMs or micro-power static RAMs, in which case critical system status information can be retained when power to the rest of the computer is disabled. With the proper software, this facility lets the computer power itself down when idle, and later resume program execution automatically at the exact point it ceased. System management functions are explained in **Chapter 4**.

Finally, a separate, dedicated DRAM memory array labeled "Video Memory" holds the display buffer used by the video graphics subsystem. The 82760SL directly supplies all address signals, data buses, and control strobes used by the display buffer, providing a truly "glue-less" memory interface

System Interconnect Buses

The various pins on the SL SuperSet components generally connect to corresponding pins on other components, external memory arrays, peripherals, or the expansion bus. Certain pins supply the clock signal inputs to each device and support the various power management functions. Others pass control and status information between chips and supply pre-decoded chip-select signals, eliminating external random-logic "glue".

All signal pins that connect to the bus labeled "ISA System Backplane" in Figure 2-1 can drive directly up to eight standard expansion slots. These include 24-bit system and local address (SA and LA) buses and a 16-bit system data (SD) bus. A control bus supervises memory and I/O read and write operations and services requests for interrupts and direct-memory-access (DMA) transfers. Table 2-1 summarizes the SL SuperSet pins that attach directly to the ISA backplane connectors.

Table 2-1. SL SuperSet Expansion Bus Pins Directly Drive the ISA Backplane

i386™SL	82360SL	Signal Mnemonic	Signal Function
X	X	SD15:0	System Data Bus
X		SA19:17	System Address Bus
X	X	SA16:0	System Address Bus
X	X	LA23:17	Local Address Bus
	X	SMEMW#	System Memory Write
	X	SMEMR#	System Memory Read
X	X	MEMW#	Memory Write
X	X	MEMR#	Memory Read
X	X	IOW#	I/O Port Write
X	X	IOR#	I/O Port Read
	X	AEN	System Address Enable
	X	IRQ15:10, 8:3, 1	System Backplane Interrupt Requests
X	X	IRQ9	VGA Interrupt Request
	X	DRQ7:5,3:0	Direct Memory Access Requests
	X	DACK7:5,3:0	Direct Memory Access Acknowledge
X	X	SBHE#	System Bus High Enable
	X	TC	Terminal DMA Transfer Cycle Count
X	X	IOCHRDY	I/O Channel Ready
X	X	0WS#	Zero Wait-State Transfer
	X	REFRESH#	System Memory Refresh Cycle
X	X	MASTER#	AT Bus Master
X	X	BALE	Buffered Address Latch Enable
	X	RESETDRV	Cold System Reset
	X	OSC	System Bus Oscillator
X		MEMCS16#	16-Bit Mem Transfer Mode Chip Select
X	X	IOCS16#	16-Bit I/O Transfer Mode Chip Select
	X	IOCHCHK#	I/O Channel Check
X	X	SYSCLK	System Clock

Main Memory Options

The Intel386 SL processor contains control and interface logic for main memory arrays built with either static (SRAM) or dynamic (DRAM) memory devices, with or without parity. The address and control functions performed by Intel386 SL processor pins vary, depending on the memory type selected. Address signals can be either latched or multiplexed, and control outputs can provide bank-select, chip-select, and byte-enable signals as appropriate for the sizes of memory components currently installed. DRAM refresh sequencing and parity generation and verification (if enabled) are automatic.

SRAM Main Memory

Power consumption is critical in palm-top and note-pad sized computers. For maximum battery life, such systems generally use low-power SRAMs for their main memory systems. An SRAM device draws essentially no power except when it is accessed. Since only one location can be addressed at a time, the total power consumption of an SRAM-based main memory system is slight, no matter what its size.

Figure 2-2 shows a simple 128K-byte main-memory system built with four SRAM chips, each driven directly by Intel386 SL CPU output pins. The Intel386 SL CPU can supply an arbitrary 22-bit address value in two stages, high-order half first, through pins MA10:0. To save time, ensuing accesses within the same 2K-byte page update only the low-order address bits.

The top address bits are also decoded internally to produce chip-select outputs (CS3:0#) for up to four SRAM memory banks. System configuration software determines the size of each bank. By decoding different sets of high-order address bits, the bank-select outputs can support memory systems with programmable bank size.

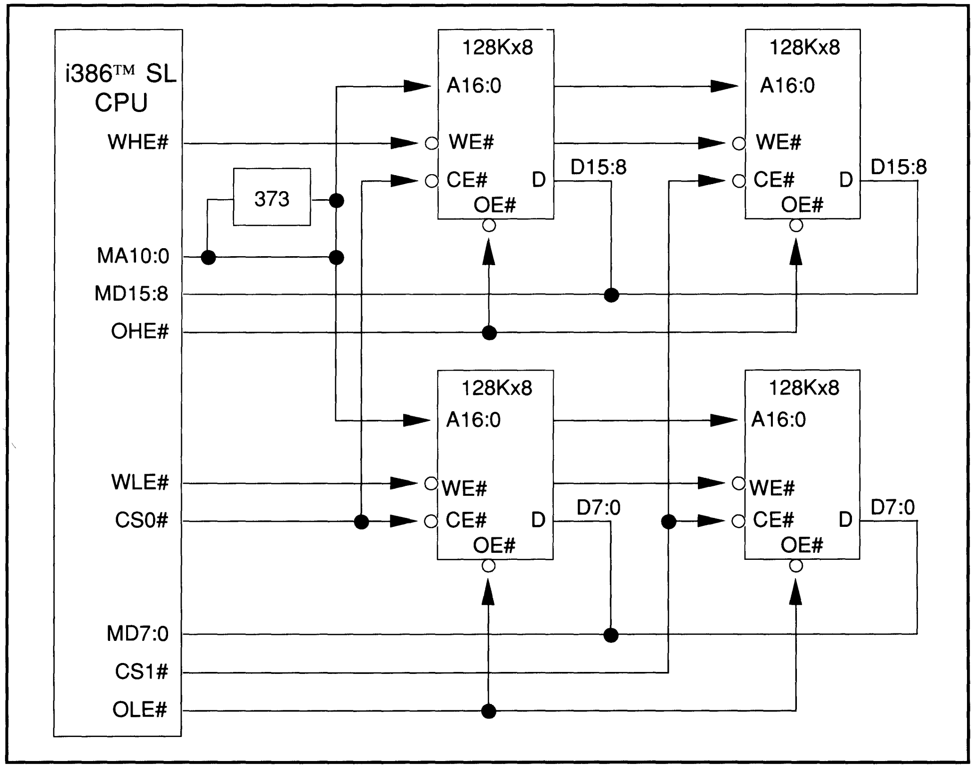


Figure 2-2. 512K-Byte Static RAM Memory Array Connects Directly to the Intel 386™SL CPU

Data bus MD15:0 transfers one or two bytes of data at a time. Transfer-type information is internally combined with memory address bit 0 to produce individual output-enable and write-enable signals for the high- and low-order bytes of the array (OHE#, OLE#, WHE#, and WLE#). SRAM configurations are expandable up to 20 megabytes with external buffers and decoders. Figure 2-3 shows a 4 megabyte SRAM main-memory configuration.

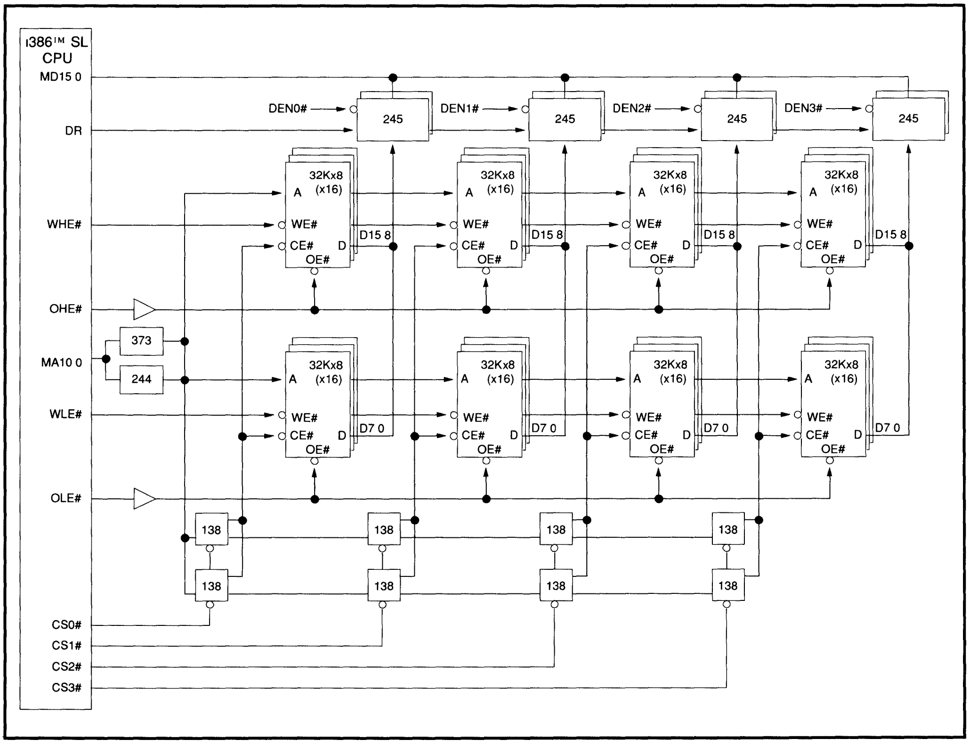


Figure 2-3. Typical Four-Bank, 4-Megabyte Buffered SRAM Memory Configuration

DRAM Main Memory

High-capacity main-memory arrays generally require fewer chips when built using DRAM devices. The DRAM control logic built into the Intel386 SL processor is extremely flexible. DRAM arrays can be 16 or 18 bits wide depending on whether automatic parity verification is enabled. Dedicated RAS#, CAS#, and WE# strobes are provided separately for the high- and low-order bytes of each DRAM bank. Figure 2-4 shows a four-bank, 20-megabyte DRAM memory array.

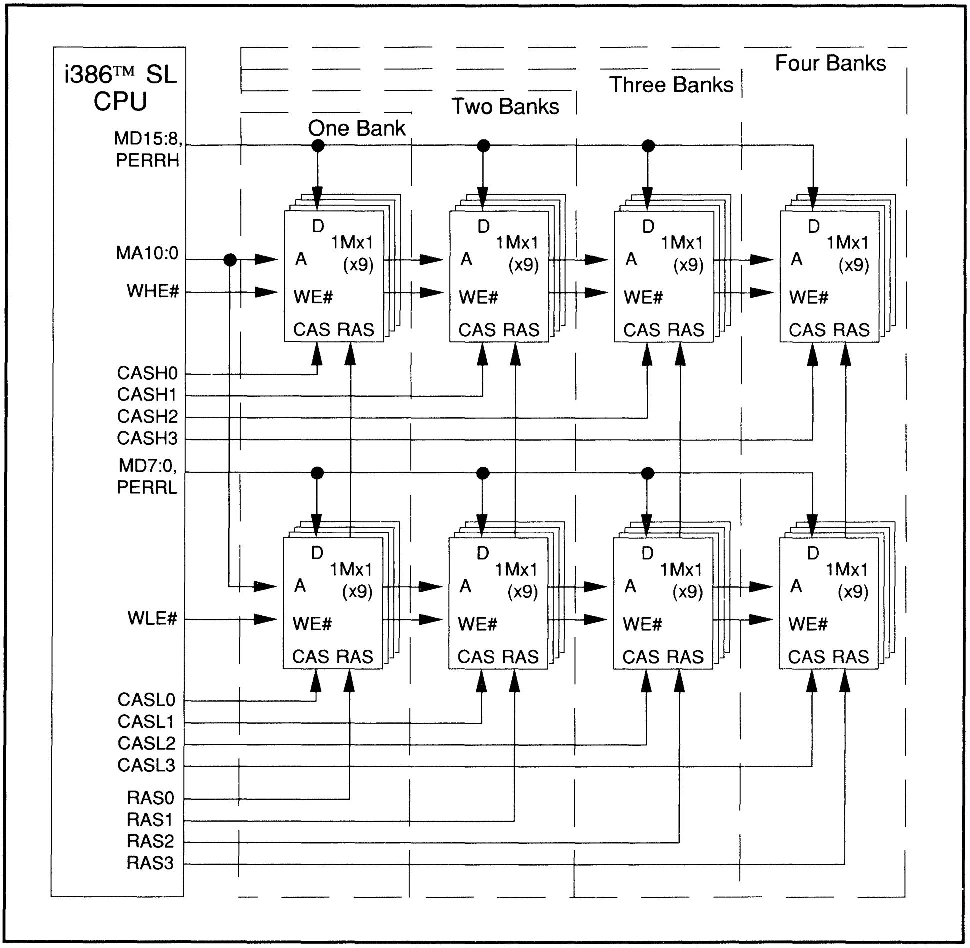


Figure 2-4. Four-Bank, 32-Megabyte Dynamic RAM Memory Configuration

The DRAM controller supports three different memory speeds. The number of CPU cycles allotted to each transfer varies to compensate for different CPU frequencies and memory speeds. One, two, three or four banks may be installed at a time. With two or four banks installed, accesses are interleaved between banks one and two, three and four for higher transfer rates.

With four banks installed, each pair may be a different size, allowing a range of memory configurations with total capacity from 512K to 20M bytes (see Table 2-2 for all possible DRAM configurations). Pins MA10:0 multiplex different sets of address bits according to the number of DRAM banks installed and the size and interleave mode of the components in each bank.

Table 2-2. Intel 386™ SL Processor Control Logic Supports DRAM Configurations Up to 20 Mbytes

	Individual Bank Size (x 16 bits)			Total System
Bank 0	**Bank 1**	**Bank 2**	**Bank 3**	**Capacity (Bytes)**
512K	512K	—	—	2 Meg
512K	512K	1 Meg	—	4 Meg
512K	512K	—	1 Meg	4 Meg
512K	512K	1 Meg	1 Meg	6 Meg
512K	512K	4 Meg	—	10 Meg
512K	512K	—	4 Meg	10 Meg
512K	512K	4 Meg	1 Meg	12 Meg
512K	512K	1 Meg	4 Meg	12 Meg
512K	512K	4 Meg	4 Meg	18 Meg
1 Meg	—	—	—	2 Meg
1 Meg	—	1 Meg	—	4 Meg
1 Meg	—	—	1 Meg	4 Meg
1 Meg	—	1 Meg	1 Meg	6 Meg
1 Meg	—	4 Meg	—	10 Meg
1 Meg	—	—	4 Meg	10 Meg
1 Meg	—	4 Meg	1 Meg	12 Meg
1 Meg	—	1 Meg	4 Meg	12 Meg
1 Meg	—	4 Meg	4 Meg	18 Meg
1 Meg	1 Meg	—	—	4 Meg
1 Meg	1 Meg	1 Meg	—	6 Meg
1 Meg	1 Meg	—	1 Meg	6 Meg
1 Meg	1 Meg	1 Meg	1 Meg	8 Meg
1 Meg	1 Meg	4 Meg	—	12 Meg
1 Meg	1 Meg	—	4 Meg	12 Meg
1 Meg	1 Meg	4 Meg	1 Meg	14 Meg
1 Meg	1 Meg	1 Meg	4 Meg	14 Meg
1 Meg	1 Meg	4 Meg	4 Meg	20 Meg
1 Meg	4 Meg	—	—	10 Meg
1 Meg	4 Meg	1 Meg	—	12 Meg
1 Meg	4 Meg	—	1 Meg	12 Meg
256K	256K	—	—	1 Meg
256K	256K	1 Meg	—	3 Meg
256K	256K	—	1 Meg	3 Meg
256K	256K	1 Meg	1 Meg	5 Meg
256K	256K	4 Meg	—	9 Meg
256K	256K	—	4 Meg	9 Meg

A number of special Intel386 SL processor facilities reduce DRAM power consumption. Only the memory devices involved in each transfer are enabled. With page-mode DRAMs, successive transfers within the same page produce CAS#-only transfer cycles for greater speed and reduced lower power. The refresh rate is programmable, and the controller can perform CAS# before RAS# refresh sequencing to reduce power-supply transients ("spiking"), improve performance, and increase power efficiency.

Memory System Sizing and Control Mechanisms

The characteristics of the main memory controller, including the type and operating mode of the devices used, the number of the installed banks, and the size of each bank are configured through software at initialization time. System initialization software can test the size and characteristics of memory components currently installed and dynamically adjust memory-control algorithms according to the results of those tests. A number of control pins alter their function depending on the basic memory technology selected; the alternate functions performed by each of these pins is shown in Table 2-3. For further details, consult the **Intel 386™ SL Microprocessor SuperSet Programmer's Reference Manual**.

Table 2-3. Intel SL SuperSet Multifunction Main Memory Control Signals

Signal Mnemonic	DRAM-Mode Pin Function	SRAM-Mode Pin Function
CMUX0	CAS, Low Byte, Bank 3	Transceiver Direction
CMUX1	CAS, High Byte, Bank 3	Address Latch Enable
CMUX2	CAS, Low Byte, Bank 2	Transceiver Enable, Bank 3
CMUX3	CAS, High Byte, Bank 2	Transceiver Enable, Bank 2
CMUX4	CAS, Low Byte, Bank 1	Transceiver Enable, Bank 1
CMUX5	CAS, High Byte, Bank 1	Transceiver Enable, Bank 1
CMUX6	CAS, Low Byte, Bank 0	Transceiver Enable, Bank 0
CMUX7	CAS, High Byte, Bank 0	Transceiver Enable, Bank 0
CMUX8	RAS, Bank 3	Chip Enable, Bank 3
CMUX9	RAS, Bank2	Chip Enable, Bank 2
CMUX10	RAS, Bank1	Chip Enable, Bank 1
CMUX11	RAS, Bank0	Chip Enable, Bank 0
CMUX12	Low-Order Byte Parity Error	Low-Byte Output Enable
CMUX13	High-Order Byte Parity Error	High-Byte Output Enable

Cache Memory System

High-performance computers traditionally include cache memory to reduce main memory latency and boost system throughput, albeit at the expense of increased system complexity and power requirements. The optional high-speed cache system supported by the SL SuperSet allows the simplest possible implementation, and can both improve CPU performance and reduce system power consumption.

The Intel386 SL processor drives the cache memory components directly through separate, dedicated address, data, and control buses, eliminating all external glue. The only components needed to add cache to an SL SuperSet-based system are the memory devices themselves, potentially just a single external SRAM. Control logic built into the Intel386 SL processor remaps the cache memory to support different cache sizes and organizations, and includes address, status, and tag bits and comparators for each.

Configuration Options

The Intel386 SL processor supports cache configurations with capacities of 16K, 32K, or 64K bytes, using just one, two, or four external SRAMs. As shown in Figure 2-5, address bus CA15:0 directly drives the address inputs of each SRAM, and data bus CD15:0 connects directly to the SRAM data pins. Dedicated write-enable and output-enable signals for the cache (CWE# and COE#) drive the SRAM control inputs directly. Separate chip-select outputs (CCHE# and CCLE#) enable the high- and low-order cache bytes.

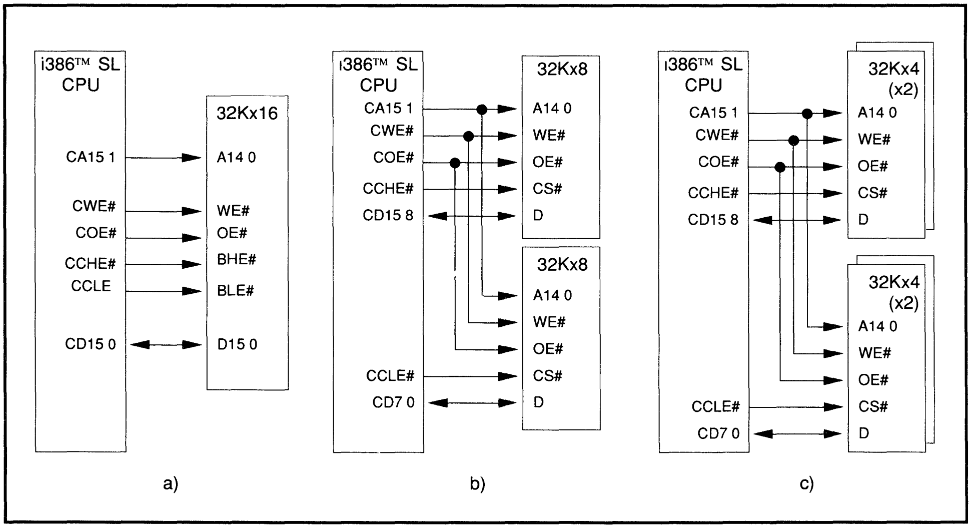


Figure 2-5. Optional High-Speed Caches Requires (a) One, (b) Two, or (c) Four External Chips

Each cache line is two bytes wide, and each set of tag bits controls a block of 16 cache lines. The cache controller contains 2048 sets of tag bits, which may be programmed to form any of the three organizations shown in Figure 2-6. The simplest (direct-mapped) organization arranges the tags as a single linear array, in which case each address in the main memory space corresponds directly to a single cache location.

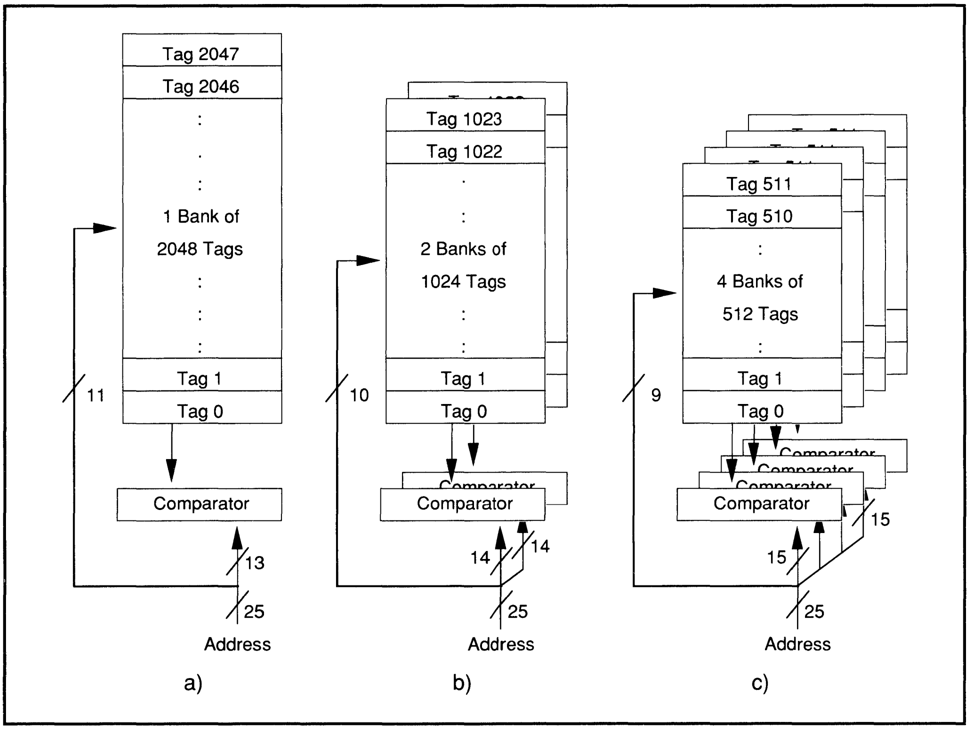


Figure 2-6. 64K Cache Set-Associativity Options: (a) Direct-Mapped, (b) Two-Way, or (c) Four-Way

Tags may also be grouped into two banks of 1024 tags (two-way set-associative) or four banks of 512 tags (four-way set associative), in which case each main memory address can map onto any of several locations within the cache. Direct, two-way or four-way associativity can be achieved with any number of cache memory chips.

When a cache miss occurs, control logic uses a least-recently used (LRU) algorithm to determine which cache block to replace with more recent data. The cache always operates in write-through mode, i.e., main memory is also updated whenever the CPU writes new data to the cache. This assures the main memory will always match updated data in the cache, and assures full compatibility with existing system software and hardware.

Cache Performance Factors

Figure 2-7 shows how cache memory affects the performance of several typical PC configurations executing the 32-bit Dhrystone benchmark program. The "Theoretical Maximum Performance" is defined as the level that would be achieved by an "ideal" Intel 386™SX CPU-based computer, i.e., one whose entire main memory is built with non-pipelined zero-wait-state memory devices for CPU speeds up to 20 MHz. Conventional (cacheless) DRAM systems typically incur an average of 0.8 wait states per memory access, which lowers performance to about 80% of the theoretical maximum.

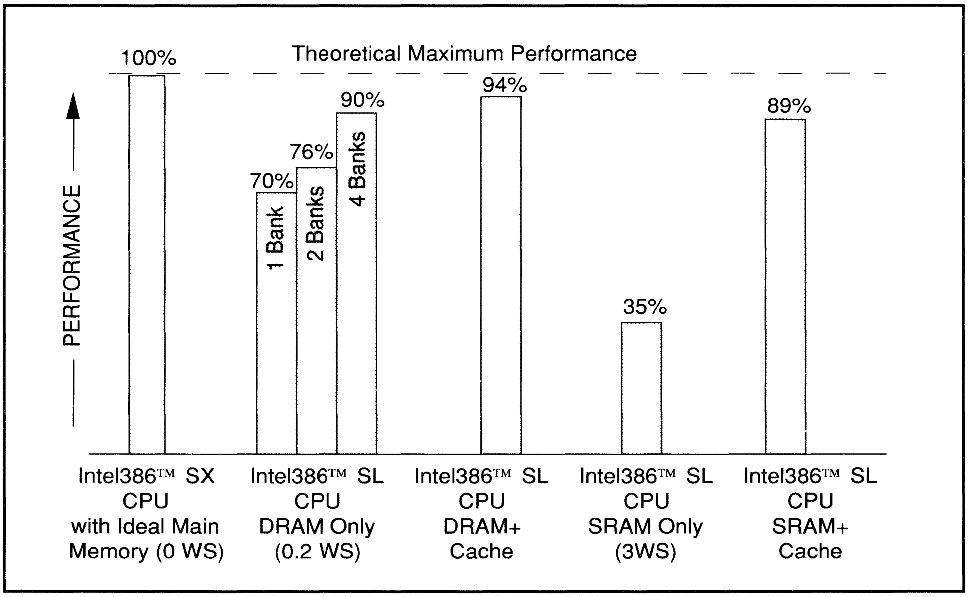


Figure 2-7. Adding Cache Memory Delivers Near-Ideal CPU Performance Levels

Memory control logic in conventional PCs cannot begin a new transfer until its intended destination is known, which is not until the target address appears on the CPU address pins. With the Intel386 SL processor , however, the DRAM control logic is part of the CPU. Information about successive transfers is known before their addresses leave the part, which lets the SL SuperSet implement control algorithms that are considerably more sophisticated than those possible with discrete DRAM controllers. Page-mode data transfers and memory-bank interleaving eliminate unnecessary precharge cycles, so the performance of a cacheless SL SuperSet-based PC with 80-ns DRAMs is close to 90% of the theoretical maximum. Adding cache boosts this figure to about 94%.

While the SRAMs used as cache memories are relatively small but quite fast, those used for personal computer main memories are typically larger, slower, and use less power than their DRAM counterparts. An SRAM-based Intel386 SL processor system with no cache might typically insert three wait states into each transfer, which reduces performs to about one-third the potential maximum rate. A cache eliminates most of these wait states. Adding a cache therefore improves the performance of SRAM-based PCs dramatically, and effectively boosts throughput to the same level as considerably faster DRAMs.

Cache Power-Consumption Factors

Figure 2-8 compares the power requirements of PC memory configurations with and without cache. Conventional PC caches generally increase system power requirements by the amount consumed by the control logic and cache components themselves. Adding cache to an Intel386 SL microprocessor system, on the other hand, can actually reduce system power.

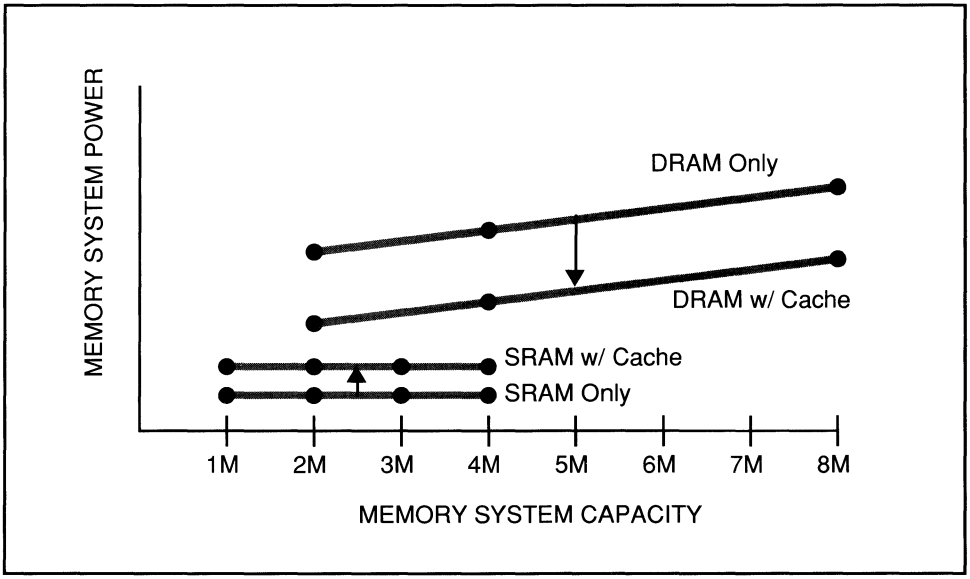


Figure 2-8. Adding Cache Memory Can Reduce Total System Power Consumption

The Intel386 SL processor's internal cache controller implements a SmartHit control algorithm that pipelines tag look-ups with data retrieval. Tag bits determine whether a memory request will hit within the cache array in time to prevent unnecessary transfers to main memory. Only the necessary cache operation completes. Conversely, the SmartHit control logic knows when a cache miss occurs early enough to initiate a main-memory or expansion-bus transfer, as needed, without further delay.

The power consumed by DRAM memory systems is proportional to the number of transfer cycles they must perform. Since a cache subsystem satisfies the majority of all memory requests, adding cache eliminates most main memory access cycles. SRAM uses significantly less power per transfer cycle than DRAM, so the system power saved by eliminating DRAM transfers more than offsets the added load of the cache components.

The power consumed by SRAM-based main memories, on the other hand, is already quite low. In such designs the power savings attributable to cache is slight, but does help offset the power consumed by the cache memories. When conventional memory accesses must be performed, however, control logic disables the cache components to further reduce power.

System initialization software determines whether or not the cache system is enabled. The size of the external cache array and the set-associativity options are also set under software control.

BIOS Memory Array

The Intel386 SL processor directly generates control signals to enable a ROM, EPROM, or Flash EPROM array to hold basic I/O system (BIOS) and/or ROM-based system software. Depending on configuration options selected at initialization time, the BIOS memory array can be either 8 or 16 bits wide, and up to 128 Kilobytes long; arbitrarily large EPROM arrays can supported with external decode logic. The processor generates special control signals to enable Flash memory reprogramming, so customized or updated versions of the BIOS can be downloaded after a system has been manufactured or sold.

Since ROMs and EPROMs are available at a variety of price/performance points, the BIOS control logic automatically inserts 0 to 15 wait states within each access cycle. A number of features increase the density, flexibility, or effective performance of the BIOS memory array. BIOS memory can be shadowed in much faster main memory, and VGA subsystem can be combined with the main system BIOS, lowering chip count and complexity.

Flash Memory Support

Figure 2-9 shows an optional Flash EPROM array used as a solid-state disk emulator. The Flash EPROM array is accessed through the same address and data bus as the ISA expansion interface, but with separate, dedicated control lines. For expanded memory access, remapping hardware divides the entire array into a collection of 64K-byte "windows" accessed through a reconfigurable block of memory addresses in the low-order megabyte of the system address space.

Flash EPROM devices can occupy any portion of the Intel386 SL microprocessor's 32-megabyte physical memory space not filled with conventional memory. With external bank-switching logic, the Flash array can be arbitrarily large.

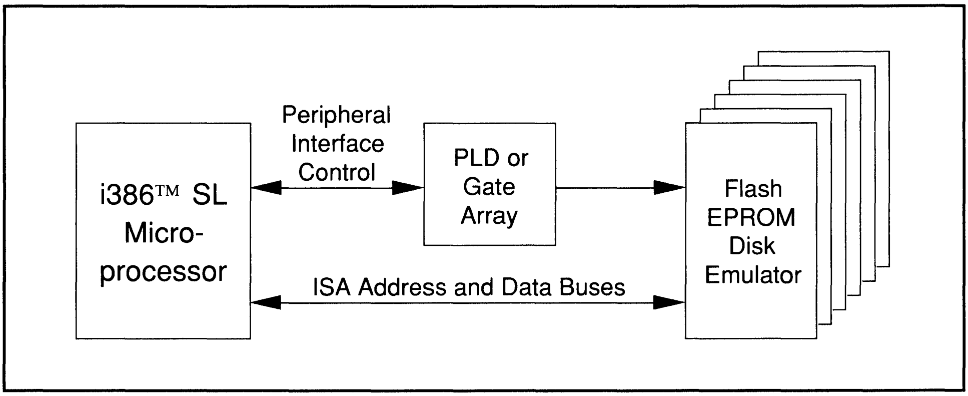


Figure 2-9. Dedicated Control Signals Simplify Flash EPROM Disk Emulator Systems

Graphics Display Subsystem

The 82760SL implements all of the control logic needed for a complete display subsystem fully compatible with the popular graphic standard. It can be configured to connect to either a monochrome or full-color CRT or a variety of monochrome flat-panel LCD displays. Each display mode is discussed below.

CRT Display Modes

The 82760SL control logic is fully-compatible at the register level with the IBM VGA graphic standard. CRT display resolution can range from 320 x 200 to 800 x 600 pixels, with up to eight bits of color information per pixel. An external color palette and digital-to-analog converter (palette DAC) controls the CRT color mapping and electron-beam intensity.

LCD Display Modes

The 82760SL supports LCD and flat-panel displays with resolutions ranging from 320 x 200 to 640 x 480 pixels. Depending on the resolution of the graphics mode selected, the display region controlled by the 82760SL is automatically shifted to fill the center of the active display area.

In color-display mode, with monochrome LCDs, the 82760SL internally translates color indexes into up to 64 levels of gray. If the display is unable to support this many levels, the 82760SL automatically performs spatial and temporal "dithering" to increase gray-scale resolution.

Display Buffer Memory

The display system image buffer occupies a separate, local memory array, either eight or 16 bits wide, containing up to two megabytes of DRAM or pseudo-static memory devices. The 82760SL multiplexes address values onto the display buffer address bus and provides a separate video data bus and dedicated control signals.

The 82760SL automatically refreshes the video buffer DRAM array separately from main memory, at a rate defined under software control. Dedicating a separate block of memory to the display buffer increases system performance and reduce power consumption by eliminating unnecessary main-memory cycles.

Display Control Mechanisms

Basic control functions for the 82760SL (VGA port addresses, CRT vs. LCD display types, buffer memory size and characteristics, DRAM refresh rate, and so forth) are all determined by register values defined by system initialization software. Thereafter, the 82760SL is fully compatible with existing VGA software standards. Registers that control the various operating modes, pixel resolution, and color palette assignments, all have the same operating characteristics as the IBM VGA standard.

System Timing Requirements

The SL SuperSet has a separate oscillator for each system timing function. For example, a 25-MHz Intel386 SL timing is derived from an external 50-MHz oscillator, while a separate oscillator (typically 16 MHz) controls expansion-bus transfer timing. The 82360SL has separate oscillators for serial communication data rates, the ISA-backplane OSC signal, and the time-of-day clock. The 82760SL has two additional oscillators to control graphics dot timing. Table 2-4 lists oscillator frequencies for ISA-compatible backplanes and VGA graphics modes. For complete information on electrical characteristics and timing specifications, consult the **Intel 386™ SL Microprocessor SuperSet Data Sheet.**

Table 2-4. The Intel 386™ SL SuperSet Provides a Dedicated Oscillator For Each System Timing Function

Oscillator Function	Related Device	Nominal Input Frequency
Main CPU Frequency	Intel386 SL	50 MHz
Expansion Bus Transfer Timing	82360SL	16 MHz
Expansion Bus OSC Signal	82360SL	14.31818 MHz
Serial I/O Data Rates	82360SL	1.8432 MHz
Time-of-Day Clock	82360SL	32.768 kHz
Video Graphics Adapter Timing	82760SL	25.175 MHz and 28.322 MHz

Hardware Architecture Summary

As the personal computer industry has matured, hardware developers have adopted a so-called Industry-Standard Architecture, including a standard set of peripheral controllers, bus specifications, and other design conventions. The Intel386 SL Microprocessor SuperSet has been designed to comply fully with all ISA hardware and peripheral conventions. At the same time, the SL SuperSet leaves hardware designers the flexibility to revise, reconfigure, or augment chip set facilities with whatever custom features may be needed to distinguish their products from those of the competition.

Chapter 3
Software System Architecture

CHAPTER 3
SOFTWARE SYSTEM ARCHITECTURE

Programmers developing software for today's personal computers must be familiar with several aspects of the target system's architecture: the central processing unit itself, the organization of main memory, the peripheral registers visible to application programs, and the special configuration and status registers through which system functions are initialized and maintained.

Each of these topics is discussed below as it applies to the Intel386 SL Microprocessor SuperSet. In each case, the SL SuperSet hardware has been designed to assure full compatibility with personal computers built using previous-generation processors. Extensions and improvements to the basic architecture have all been made in ways that guarantee existing software will run unchanged on SL SuperSet-based PCs.

Intel386™ Microprocessor Family Architecture Overview

The most basic issue of software compatibility relates to the system-level and applications-level programming architecture of the central processor itself. The CPU at the heart of the Intel386 SL Microprocessor SuperSet is a member of the Intel386 microprocessor family, joining the original Intel386™ DX Microprocessor, the Intel386™ SX Microprocessor, the ultra-high performance Intel486™ Microprocessor, and the newest member, the Intel486 SX.

Like each of its predecessors, the Intel386 SL processor embodies a full 32-bit internal architecture, with an efficient, full-featured instruction set, complex memory addressing modes, interprocess protection mechanisms for enhanced data security, and an essentially unlimited 32-bit address space for programs and data. Like all Intel386 family members, the Intel386 SL microprocessor has special hardware to directly execute Intel's earlier, 16-bit microprocessors, including the 8088, 8086, and 80286.

This means that in addition to newly-developed 32-bit software, Intel386 SL processor based computers can run, without modification, all application programs and operating system software originally developed for the IBM PC, PC/AT, and other compatible computers. Table 3-1 summarizes the hardware and software features and capabilities of each microprocessor in the x86 product line.

Table 3-1. Intelx86 Microprocessor Family Feature Comparison

Attribute	8088	8086	80286	i386™ SX CPU	i386™ DX CPU	i486™ SX CPU	i486™ DX CPU	i386™ SL CPU
Processor Execution Modes	8088/86	8088/86	8088/86, 80286	8088/86, 80286, 80386	8088/86, 80286, 80386	8088/86, 80286, 80386	8088/86, 80286, 80386	8088/86, 80286, 80386
Processor Data Precision	8/16 bits	8/16 bits	8/16 bits	8/16/32 bits	8/16/32 bits	8/16/32 bite	8/16/32 bits	8/16/32 bits
Physical Address Space	1 Mbyte	1 Mbyte	16 Mbytes	16 Mbytes	4 Gbytes	4 Gbytes	4 Gbytes	32 Mbytes
Virtual Address Space	(None)	(None)	16 Mbytes	4 Gbytes	4 Gbytes	4 Gbytes	4 Gbytes	4 Gbytes
External Bus Width	8 bits	16 bits	16 bits	16 bits	32 bits	32 bits	32 bits	16 bits
Coprocessor Support	8087	8087	80287	i387™ SX	i387™ DX	i487™ SX	On-Chip	i387 SX
Cache Controller	—	—	—	80385SX or 80395SX	80385DX or 80395DX	On-Chip	On-Chip	On-Chip

Architectural Extensions for System Management

Portable PCs built with conventional technology encounter several areas of difficulty controlling power consumption. Power management functions require dedicated ports for monitoring and controlling power usage, dedicated interrupt requests, vectors, and service routines to process asynchronous power-related events, and a dedicated sections of main system memory for power-management software, stacks, and data structures. Previously these resources had to be taken from among those supported by the original microprocessor architecture. If other software or hardware functions made use of the same resources, conflicts could develop, introducing the hazard that standard 32-bit operating systems and applications programs could malfunction on early portable PCs.

The new Intel386 SL Microprocessor SuperSet power-management capabilities are supported by extensions to the basic Intel386 architecture. Instead of locating instruction and data memory needed by system-management functions within the original Intel386 architecture address space, for example, these functions occupy an entirely separate region of memory, with separate control signals asserted by processor hardware. Instead of consuming one of the original Intel386 architecture's fixed number of interrupt levels for system-management functions, the Intel386 SL microprocessor provides a separate interrupt input (SMI) complete with its own interrupt vector.

The memory space and interrupt systems provided by earlier processors are thus preserved intact, ensuring full compatibility with all existing application programs and operating systems. Thus there is no danger that power-management interrupt logic or trap handler routines might conflict with end-user system or applications software. **Chapter 4** discusses in detail how the new Intel386 SL microprocessor system management extensions are used to solve the problems of power management.

Memory System Management

Early 16-bit personal computers imposed a 640-Kilobyte limit on system memory. This puts a severe constraint on modern software developers, so chip and system designers have developed a variety of hardware techniques to expand the memory available to newer programs. In order to boost performance and utilize memory as efficiently as possible, the Intel386 SL processor includes control logic for each of the standard memory expansion techniques.

Virtual Memory Paging

Like all members of the Intel386 architecture family, the Intel386 SL processor supports a complete virtual-memory management system. Execution programs manipulate code and data addresses that are 32 bits wide, which yields a 4 Gigabyte memory space. In simple programming environments these 32-bit values are treated as physical addresses and sent unchanged to main memory.

In more complex environments, program-generated addresses can act as virtual addresses. With the virtual-memory option enabled, the Intel386 SL processor transforms each computed address to a corresponding physical address before it leaves the chip. The address translation logic also includes a demand paging system. With the proper operating system support, application programs function as though 4 gigabytes of main memory were available to each, even with relatively little physical memory installed.

LIM 4.0 Memory Expansion

The Intel386 architecture's 4-Gbyte address space and virtual memory translation systems aren't enabled by conventional 16-bit application programs and operating systems such as DOS. Such software must rely on techniques for expanding system memory that involve special external hardware.

One popular technique is based on an expanded memory specification developed jointly by Lotus, Intel, and Microsoft, now called the LIM EMS standard, version 4.0. The Intel386 SL processor contains on-chip all of the hardware registers, translation tables, and address transformation logic needed to support the LIM standard. Four sets of translation registers are provided, each of which can remap up to 768K bytes of extended system memory into the low-order address space accessible to 16-bit programs. Initialization software adjusts the starting address and size each LIM window, and the number and size of the LIM memory pages mapped by each.

ROM Shadowing

In most personal computers, EPROMs hold BIOS software and peripheral interface drivers. High-end systems commonly use a hardware technique called "ROM shadowing" to improve the effective speed of these relatively slow devices. During system initialization, software copies the contents of each EPROM into a reserved section of DRAM, and the memory control logic is reconfigured so later EPROM references read much faster DRAMs instead. The Intel386 SL processor contains internal configuration registers and control logic to perform the address transformations needed for ROM shadowing.

Memory Backfilling and Roll-Over

Today's most sophisticated PCs also employ two other types of address transformations to regroup or reposition blocks of memory. "Backfilling" is a technique that treats fragments of memory physically located at high-order addresses as though they occupied lower-order addresses. As a result, these memory regions can then be referenced more conveniently by 16-bit DOS software.

A technique called "memory space rollover" collects together individual memory fragments that might otherwise be unusable—perhaps because BIOS EPROMs occupy the same physical address space—and then consolidates and relocates them to a region of high-order addresses. Software can then access memory as a contiguous block rather than isolated fragments. Again, the Intel386 SL processor contains the translation and control logic needed for both memory rollover and backfilling.

Memory Region Retargeting Logic

Figure 3-1 illustrates how the Intel386 SL processor performs the various address transformation steps outlined above. Once the transformations are complete, mapping logic determines whether the final address corresponds to a section of main memory accessed through the local processor memory bus, or whether it corresponds to a memory expansion card or add-in board accessed through the ISA backplane. Control circuitry then drives the appropriate bus and asserts the appropriate control signals accordingly.

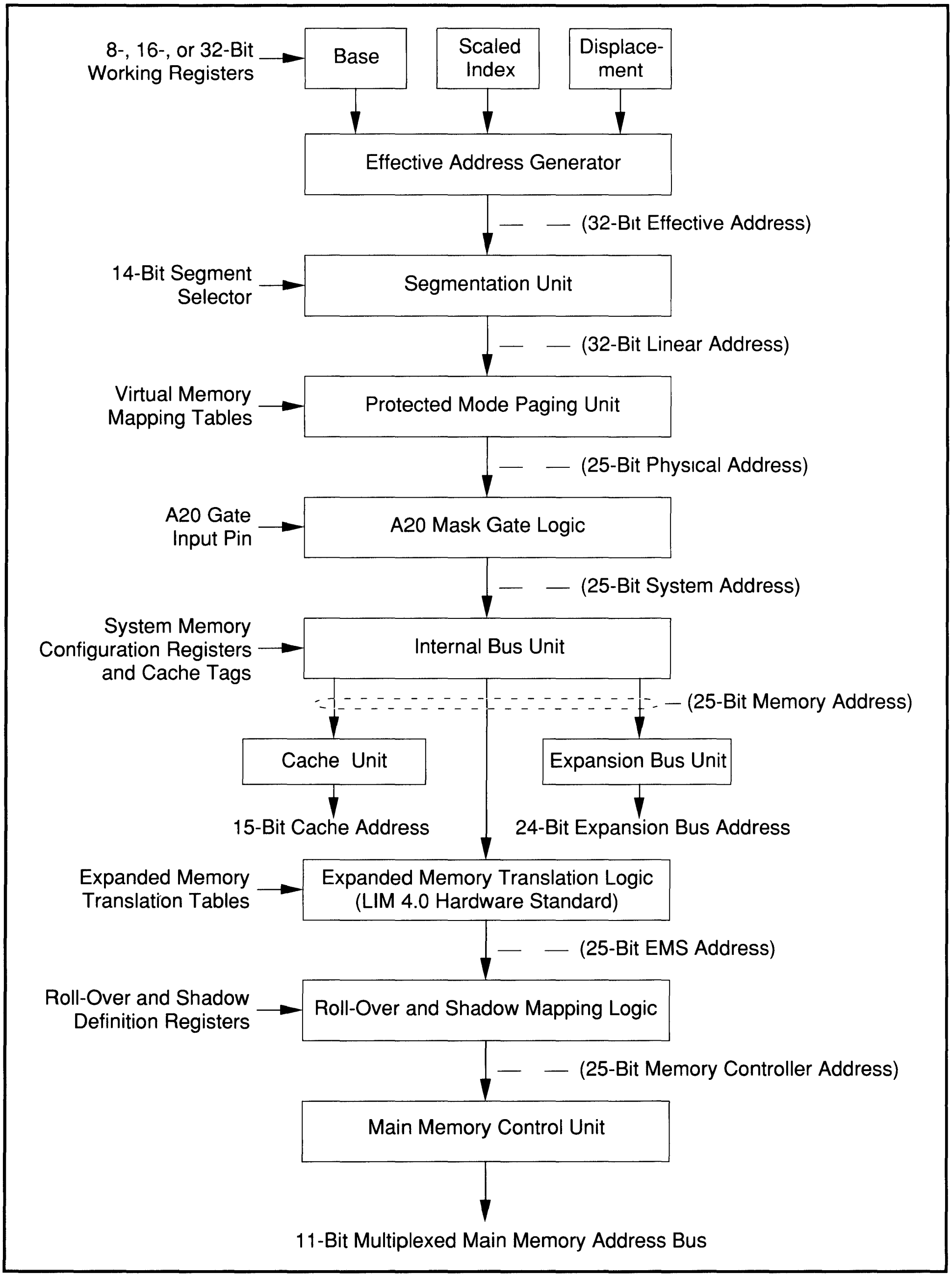

8-, 16-, or 32-Bit Working Registers
Base
Scaled Index
Displacement
Effective Address Generator
(32-Bit Effective Address)
14-Bit Segment Selector
Segmentation Unit
(32-Bit Linear Address)
Virtual Memory Mapping Tables
Protected Mode Paging Unit
(25-Bit Physical Address)
A20 Gate Input Pin
A20 Mask Gate Logic
(25-Bit System Address)
System Memory Configuration Registers and Cache Tags
Internal Bus Unit
(25-Bit Memory Address)
Cache Unit
Expansion Bus Unit
15-Bit Cache Address
24-Bit Expansion Bus Address
Expanded Memory Translation Tables
Expanded Memory Translation Logic (LIM 4.0 Hardware Standard)
(25-Bit EMS Address)
Roll-Over and Shadow Definition Registers
Roll-Over and Shadow Mapping Logic
(25-Bit Memory Controller Address)
Main Memory Control Unit
11-Bit Multiplexed Main Memory Address Bus

Figure 3-1. Intel386™ SL Processor Memory Address Translation Logic Eliminates External Propagation Delays

CPU and Memory Configuration and Control Registers

Registers within the Intel386 SL processor control CPU operating modes, configure the memory control logic, and transform memory references to match system hardware. For further information concerning these registers, the functions they perform, and how to write system software to initialize and maintain them, consult the **Intel386™ SL Microprocessor SuperSet Programmer's Reference Manual**.

Standard Peripheral Interface Registers

Most of the I/O facilities and control logic provided by the SL SuperSet are contained within the 82360SL. Among these facilities are standard peripheral controllers of the sort commonly used in personal computers, including two serial I/O channels compatible with the 16450 UART (designated COMA and COMB, but reconfigurable in a DOS environment as COM1, COM2, or any other COM ports), a parallel port (commonly called LPT1), two 8254 programmable interval timer/counters, two 8259A programmable interrupt controllers, and two 8237 direct-memory-access (DMA) controllers.

Logic within the 82360SL compatible with the 146818 real-time clock keeps track of the time and date when power to the rest of the system is removed. Environment and other configuration parameters can occupy a 256-byte block of low-power RAM, twice the size of the backup memory provided by standard PCs. Table 3-2 lists the peripheral ports and real-time clock registers generally available to end-user application programs.

For simple and low-cost secondary storage, the 82360SL provides a decoded CS# signal and control logic for an external 82077SL floppy-disk controller, and an expansion port that connects directly to a standard IDE hard-disk interface. A special control port compatible with the PS/2 product line implements standard ISA system functions including a fast reset option, address mask gate, and bidirectional data interface to the line printer.

Other control ports implement a fast reset option, a control pin for the A20 mask gate, a parallel bidirectional line printer interface, and other extended system functions required by the PS/2 product line.

The 82360SL also provides an uncommitted general-purpose output port called the ideaPort. Depending on OEM-supplied system management software, this port can be configured to control power supplies, security features, or other system functions at the designer's discretion.

Table 3-2. Program-Visible I/O Ports Include All Standard ISA Peripheral Registers

Functional Description	Default Address
DMA Controller 1	00H—0FH
Interrupt Controller 1	20H—21H
Programmable Interval Timer 1	40H—43H
Keyboard Controller and Port B	60H—64H
NMI and RTC	70H—71H
Memory Address Mapper	80H—8FH
Port 92	92H
Interrupt Controller 2	0A0H—0A1H
DMA Controller 2	0C0H—0DEH
Fast A20 Gate	0EEH
Fast CPU Reset	0EFH
Port 102	102H
Hard Disk Controller Interface	1F0H—1F7H
Parallel Port 2 (LPT2)	278H—27AH
Serial Controller 2 (COM2)	2F8H—2FFH
Parallel Port 1	378H—37AH
Parallel Port 3	3BCH—3BEH
Floppy Disk Controller Interface	3F0H—3F7H
Hard Disk Controller Interface	3F6H—3F7H
Serial Controller 1	3F8H—3FFH

Each of the 82360SL peripheral functions is completely configurable, under software control, through a separate set of configuration registers. OEM software can initialize each peripheral register to behave exactly like the discrete components found in industry-standard PCs. Initialization code can also relocate each peripheral functions to appear at an arbitrary I/O port address, or disabled it entirely, to expand the designer's options and avoid conflicts with external components at the same port addresses.

VGA Graphics System Registers

The control registers and special control logic for the graphics display subsystem are all contained in the 82760SL. These registers can also be configured by software at initialization time to match the standard IBM VGA (video graphics adapter) specification.

New Peripheral I/O Facilities

In addition to the standard peripheral and graphic interface functions visible to most application programs, the SL SuperSet contains a number of "hidden" registers, accessed only by system initialization and maintenance software. Some are used to regulate the operating characteristics of various CPU, memory, and peripheral functions, and have been described earlier. Others perform power-management functions to minimize the energy requirements of battery-powered portable computers. Tables 3-3a through 3-3n contains a complete list of configuration and control registers newly-added to the SL SuperSet .

Table 3-3a. Configuration Space Control Registers

Register Mnemonic	Register Function
SIGNATURE	Signature Register
CPUPWRMODE	CPU Power Mode Register
CFGSTAT	Configuration Index Status Register
CFGINDEX	Configuration Index Index Register
CFGDATA	Configuration Index Data Register
IDXLCK	Configuration Index Lock Register

Table 3-3b. On-Board Memory Control Registers

Register Mnemonic	Register Function
CCR	Cache Configuration Register
OMDCR	Lower Memory Disable Register
OMSR	On-board Memory Suspend Refresh Register
OMRBCR	On-board Roll-over Base Register
OMLCR	On-board Memory Limit Register
OMS[A:F]CR	On-Board Memory Shadow Registers
OMBRCR	On-board Memory Block Roll-over Register
NC[A:C]CR	Lower Memory Non-Cachable Configuration Registers
NC[D:G]CR	Memory-Mapped I/O Non-Cachable Configuration Registers
MCMODE	Memory Controller Mode Register
MCRF	Memory Controller Refresh Register
MCAS	Memory Controller Memory Auto Scan Register
MCSRAMWS	Memory Controller SRAM Wait State Register
MCDRAMMD	Memory Controller DRAM Mode Register
MCBS	Memory Controller Bank Size/Enable Register
MCPEL[A:B]	Memory Controller Parity Error Latch Register
MCRASTO	Memory Controller RAS Time Out Counter Register
MCBSEXT	Memory Controller Bank Size/Enable Extension Register

Table 3-3c. LIM 4.0 EMS Support Registers

Register Mnemonic	Register Function
EMSBASE	EMS Base Register
EMSCNTLREG	EMS Control Register
EMSINDEXREG	EMS Index Register
EMSDPREG	EMS Data Port Register

Table 3-3d. PI-Bus Configuration Registers

Register Mnemonic	Register Function
ISAWINDOW	ISA Sliding Window Register
EBC1CR	External Bus Unit Configuration Register 1
EBC2CR	External Bus Unit Configuration Register 2
GAACR	Graphics Configuration Register A
GABCR	Graphics Configuration Register B

Table 3-3e. Power Management Control Registers

Register Mnemonic	Register Function
SM_REQ_CNTRL	System Management Control Register
SMOUT_CNTRL	System Managment Output Control Register
MCSMRAM	Memory Controller On-board System Management RAM Area Select Register
RESUME_MASK	Resume Mask Register
STDBY_TMR_CNTRL	Local Standby Device Idle Control Timer Register
SUS_REF	Suspend Refresh Enable Register

Table 3-3f. Power Management Support Registers

Register Mnemonic	Register Function
SM_FILO	System Management FILO and Instruction Length Register
SMLIL	System Management Last Instruction Length Register
SMI_MARK	SMI CPU Reset Block Register
SMI_CLR	SMI Arbitrate Register

Table 3-3g. Power Management Status Registers

Register Mnemonic	Register Function
EXT_STS	External Pin Status Register
SM_REQ_STS	General System Management Request Register
SPND_STS	Suspend Request Register
LSTDBY_STS	Local Standby Request Register
LTRP_STS	Local Trap Request Register

Table 3-3h. Clock Control Registers

Register Mnemonic	Register Function
DMA_STP_CLK	DMA Stop Clock Register
KC_CLK_CNTRL	Keyboard Clock Control Register
STP_CLK	Stop Clock Register

Table 3-3i. System Activity Registers

Register Mnemonic	Register Function
TRP_ADR[L:H]_DEV[0:5]	Device Base Address Registers
TRP_ADR_MSK_DEV[0:5]	Device I/O Trap Enable and Mask Registers
SYS_EVNT_CFG[0:2]	System Event Registers
STP_BRK_CFG[0:2]	Stop Break Registers

Table 3-3j. Power Management Timer Registers

Register Mnemonic	Register Function
APWR_TMR[L:H]	Auto Power Off Timer Count Registers
GSTDBY_TMR[L:H]	Global Standby Timer Count Registers
LSTDBY_TMR_DEV[0:5]	Local Device Idle Timer Count Registers

Table 3-3k. Suspend Warning Timer Count Registers

Register Mnemonic	Register Function
SUS_WRN_TMR_APWR	Auto Power Off Suspend Warning Timer Count Register
SUS_WRN_TMR_ESMI	Suspend Warning Timer Count for External SMI
SUS_WRN_TMR_SSMI	Suspend Warning Timer Count for Software SMI
SUS_WRN_TMR_BAT	Suspend Warning Timer Count for Battery Low
SUS_WRN_TMR_SRBTN	Suspend Warning Timer Count for Suspend/Resume Button Press

Table 3-3l. Enhancement Features Registers

Register Mnemonic	Register Function
SFS_ENABLE	Special Feature Set Enable
SFS_DISABLE	Special Feature Set Disable
SLOW_CPU	Slow CPU Register
FAST_CPU	Fast CPU Register
PRT461	Fail-Safe NMI Control Register
CRST_TMR	CPU Reset Delay Timer Count Register
CRST_PULSE	CPU Rest Pulse Width Timer Count
FAST_A20_GATE	Fast A2O GATE Register
FAST_CPU_RESET	Fast CPU Reset Register
PORT92	PS/2 Port 92
PORT102	Parallel Port Control Register

Table 3-3m. Peripheral Configuration Registers

Register Mnemonic	Register Function
ROMCS_DEC	ROM Chip Select Decode
CFGR[1:4]	System Configuration Register [1:4]
COM_[A:B]BA_[L:H]	Serial Port Base Address Registers
DMA_WS[1:2]	DMA Wait State Registers
ASMI_ADDRH	High Byte ASMI Base Address
ASMI_ADDRL	Low Byte ASMI Base Address

Table 3-3n. Timer 2 Registers

Register Mnemonic	Register Function
T2_CH0_CNT	Timer 2 Counter 0 Count Register
T2_CH1_CNT	Timer 2 Counter 1 Count Register
T2_CH2_CNT	Timer 2 Counter 2 Count Register
T2_COMMAND	Timer 2 Command Register

Still other processor and memory support functions contained in the 82360SL eliminate much of the discrete external logic and "glue" that might otherwise be required by a PC. The 82360SL provides internal variable-frequency clock generators for the CPU, backplane, and video subsystem, and a special low-power DRAM refresh timer used to keep main system memory alive and refreshed during low-power stand-by operation.

Peripheral Control Register "Shadow" Latches

In conventional microcomputer applications, initialization software generally sets each peripheral control register to a default operating mode. Thereafter, application programs can change these registers from time to time as the need arises.

Unfortunately, the control registers for most of the discrete peripherals used in conventional PC designs are write-only. The internal operating mode bits cannot be read, so system software generally cannot determine what control parameters were last sent to each peripheral. If power to the peripheral is removed while a program is executing, its internal state will be lost, so control registers cannot reliably be reloaded when power returns. (Software could in principle keep track of command values sent to each register, but existing programs do not do so.) This makes it impossible for conventional portable PCs to restore peripherals to their previous state once they have been disabled, or following system power failures.

To solve this problem, each peripheral control register within the SL SuperSet includes a corresponding status register that saves each command value. By reading these status registers, each peripheral's complete current operating mode may be monitored. Power-conservation software can therefore read and preserve the state of any peripheral before disabling power to the related subsystem. When power is restored, software can rewrite correct configuration values to the port, so program operation can continue. Table 3-4 lists the control state shadow registers implemented within the SL SuperSet.

Table 3-4. 82360SL Peripheral and I/O Shadow Registers

Register Mnemonic	Shadowed Control Register
SHDMA0BA	DMA Channel 0 Base Address
SHDMA0WC	DMA Channel 0 Count
SHDMA0MOD	DMA Channel 0 Mode
SHDMA1BA	DMA Channel 1 Base Address
SHDMA1WC	DMA Channel 1 Count
SHDMA1MOD	DMA Channel 1 Mode
SHDMA2BA	DMA Channel 2 Base Address
SHDMA2WC	DMA Channel 2 Count
SHDMA2MOD	DMA Channel 2 Mode
SHDMA3BA	DMA Channel 3 Base Address
SHDMA3WC	DMA Channel 3 Count
SHDMA3MOD	DMA Channel 3 Mode
SHDMAMSK1	DMA Controller 1 Mask Register
SHDMA4BA	DMA Channel 4 Base Address
SHDMA4WC	DMA Channel 4 Count
SHDMA4MOD	DMA Channel 4 Mode

Table 3-4 (cont.) 82360SL Peripheral and I/O Shadow Registers

Register Mnemonic	Shadowed Control Register
SHDMA5BA	DMA Channel 5 Base Address
SHDMA5WC	DMA Channel 5 Count
SHDMA5MOD	DMA Channel 5 Mode
SHDMA6BA	DMA Channel 6 Base Address
SHDMA6WC	DMA Channel 6 Count
SHDMA6MOD	DMA Channel 6 Mode
SHDMA7BA	DMA Channel 7 Base Address
SHDMA7WC	DMA Channel 7 Count
SHDMA7MOD	DMA Channel 7 Mode
SHDMAMSK2	DMA Controller 2 Mask Register
SHT1XH0CL	Timer 1 Counter 0 Count Low
SHT1XH0CH	Timer 1 Counter 0 Count High
SHT1XH1CL	Timer 1 Counter 1 Count Low
SHT1XH1CH	Timer 1 Counter 1 Count High
SHT1XH2CL	Timer 1 Counter 2 Count Low
SHT1XH2CH	Timer 1 Counter 2 Count High
SHT2XH0CL	Timer 2 Counter 0 Count Low
SHT2XH0CH	Timer 2 Counter 0 Count High
SHT2XH1CL	Timer 2 Counter 1 Count Low
SHT2XH1CH	Timer 2 Counter 1 Count High
SHT2XH2CL	Timer 2 Counter 2 Count Low
SHT2XH2CH	Timer 2 Counter 2 Count High
SHINT1ICW1	PIC 1 ICW 1
SHINT1ICW2	PIC 1 ICW 2
SHINT1ICW3	PIC 1 ICW 3
SHINT1ICW4	PIC 1 ICW 4
SHINT1OCW2	PIC 1 OCW 2
SHINT1OCW3	PIC 1 OCW 3
SHINT2ICW1	PIC 2 ICW 1
SHINT2ICW2	PIC 2 ICW 2
SHINT2ICW3	PIC 2 ICW 3
SHINT2ICW4	PIC 2 ICW 4
SHINT2OCW2	PIC 2 OCW 2
SHINT2OCW3	PIC 2 OCW 3
SHNMIMASK	NMI Mask and RTC Index

Compatibility Safeguards

Engineers often face the challenge of new capabilities to an existing product line without losing compatibility with the original design. Complete hardware and software compatibility is especially vital in the personal computer marketplace to keep software from malfunctioning or running erratically. The techniques by which new capabilities are supported in the SL SuperSet are designed to ensure no incompatibilities would arise with respect to earlier hardware and software designs.

The underlying CPU architecture, standard peripheral registers, and power-management facilities naturally ensure complete compatibility with existing operating systems and applications software. Each of the other hardware facilities has likewise been designed such that the presence of the new functionality can be totally hidden from program execution.

A20 Pin Mask

For example, there's a discrepancy in the way 8088 and 80286 microprocessors handle addresses above one megabyte. Most AT-class PCs have discrete external logic on address pin A20 to overcome this discrepancy. The Intel386 SL processor moves this logic on-chip to simplify system hardware and ensure the addresses tracked by the internal cache controller always match those sent to main memory.

Non-Cachability of Memory-Mapped I/O

Another potential hazard involves the caching of memory regions shared between processing elements. While regions of physical memory that are read or written only by the Intel386 SL processor can always be cached safely, those that can be modified by a memory-mapped I/O device, network interface board, or hard-disk controllers, on the other hand, present a hazard and should not be held in the cache.

If they were, the Intel386 SL processor might read "stale" data from the cache instead of consulting the communications device itself. To prevent this hazard from arising, initialization software can designate any block of memory addresses accessed through the backplane expansion bus as non-cachable.

Cache Invalidation on LIM Page Changes

The LIM memory expansion logic presents a similar cachability hazard, but contains similar safeguards. Changing the contents of LIM mapping registers effectively replaces an entire block of data within a given memory region. If data from a previous block were present in the cache, attempts to reference new data might fail. Conventional PC designs solve this problem by designating as non-cachable the entire range of addresses through which LIM memory can be viewed.

This technique does work safely, but disabling the cache can significantly degrade system throughput, especially when LIM memory pages contain code overlays. The Intel386 SL processor mapping logic supports this scheme, but also supports a second, better scheme that uses dedicated logic within the SL SuperSet. Code and data values accessed through the LIM expansion window are all copied into the cache, and can be retrieved repeatedly from the cache. Whenever LIM mapping registers are changed, however, the cache controller automatically invalidates any values initially read from the now invalid LIM windows.

Concealing Peripherals From Applications Programs

Adding new ports to a conventional PC can adversely affect existing software. If the address of the new port conflicts with an earlier port, contention will arise when either is accessed. Sometimes existing software verifies that ports that are thought not to be in use are indeed vacant. If so, problems will arise if a newly added register occupies previously vacant addresses.

The SL SuperSet accesses its control and status registers in a way that removes the risk of address conflicts. Most registers are accessed in two steps. Software first writes the index of the register it wishes to interrogate or modify to a resource selection port, and then reads or writes register data through a second common port. This two-step process consumes only a few conventional port addresses, regardless of the number of registers involved.

Secondly, even the common interface registers are hidden from existing software. System maintenance software must execute a designated sequence of special instructions before the new configuration, mode, and status registers will become visible; any conventional I/O ports at the same addresses are temporarily disabled. When maintenance functions are done, another special sequence disables the new registers and reenables the conventional ports.

Software Architecture Summary

The existing software base of personal computer operating systems and applications programs creates a *de facto* software standard with which all personal computers must comply. Unfortunately, conventional technology makes it impossible to build portable computers that support certain operating modes and desirable power-conservation features without losing some degree of software compatibility.

The Intel386 SL Microprocessor SuperSet contains an assortment of special hardware functions and dedicated registers that eliminate the hazards of earlier systems. Moreover, new System Management extensions to the original Intel386 architecture support these facilities in a way that guarantees full compatibility with all existing software.

Chapter 4
Power Management Strategy

CHAPTER 4
POWER MANAGEMENT STRATEGY

Early microprocessor designs were naturally more concerned with performance, cost, design complexity, and system manufacturability than with power consumption. It was not until battery-operated portable PCs were developed that power-conservation factors became critical. Unfortunately, conventional PC components limit the extent to which power consumption can be reduced, and conventional design techniques to reduce power unavoidably introduce software compatibility hazards.

The Intel386 SL Microprocessor SuperSet is the first general-purpose microprocessor for which power-conservation concerns affect all aspects of device architecture and implementation. This chapter describes these facilities, and shows how they can be used to improve the capabilities and extend the battery life of small portable computers.

Power Management Overview

The problems encountered in optimizing power usage are three-fold: processor and peripheral components must be designed with modes that reduce power demands or let power be removed entirely; dedicated control logic must monitor peripheral usage to determine when to disable or re-enable power to peripheral subsystems; and all power-management resources and software must be implemented in ways that ensure full compatibility with existing operating systems and applications.

Figure 4-1 shows where the power goes in a typical portable PC. Mechanical peripheral devices consume most of the power, especially the display subsystem, hard disk, and floppy-disk drives. Main memory is the next largest power sink, and other electronic components—the CPU, peripherals, and clock circuitry—account for the rest. At times all system elements may be in use at once. In “Full-On” mode, while a backup program is copying files from hard-disk through memory to floppy disks, for example, power consumption is at its maximum.

Most of the power consumed by a fully energized PC is wasted, unfortunately, most of the time. Though hard-disks may spin continuously, data transfers may only be sporadic. PCs may sit idle or unattended for extended periods of time, for example when the operator receives a phone call. During such periods a PC may do no useful work, yet power continues to be drained.

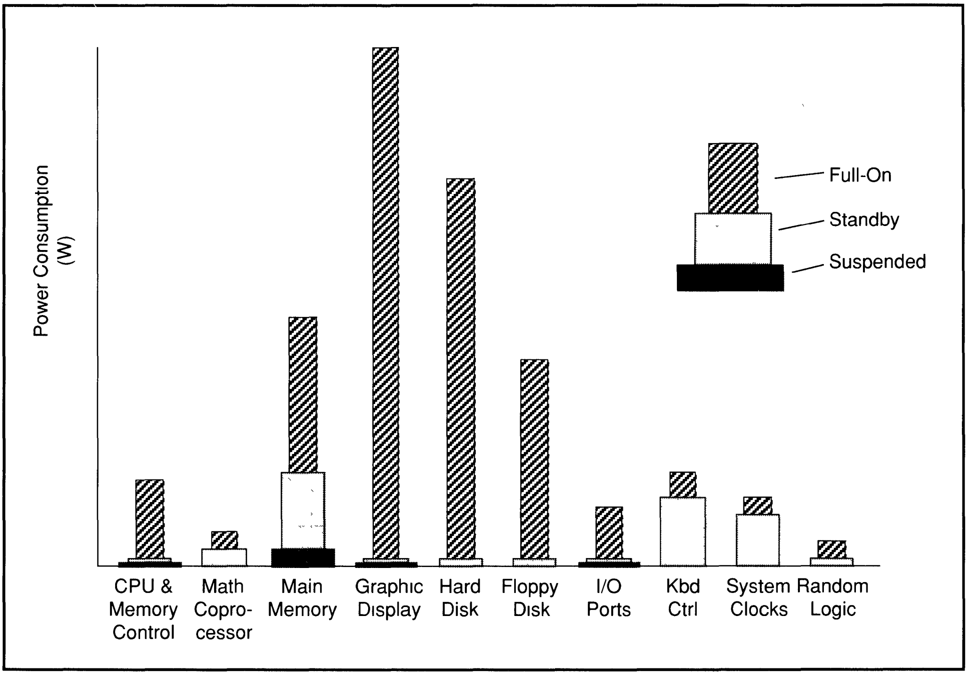


Figure 4-1. Lap-Top System Power Consumption Varies Greatly Depending on Operating Mode

Basic Power Reduction Techniques

Battery-operated PCs that seek to maximize the useful time between charges cannot tolerate unnecessary power wastage. Many portable computers therefore provide a low-power "Standby" mode that disables certain power-hungry peripherals and other devices for which power reduction is most straight forward. Simply disabling a light-emitting display or extinguishing an LCD backlight, for example, reduces its power usage to near zero, and saves perhaps one-third of the full-on system power.

Switching off power entirely to hard disk and diskette drives when they're not being used may reduce system power by another third. Low-power operating modes built into other peripherals may be invoked, and the central processing unit and other circuitry may run at a reduced frequency to save additional power. Programs continue executing, however, and normal operation can resume simply by re-energizing the peripherals.

To extend battery life still further, some portable systems also have a "Suspend" or "Sleep" mode. Critical system status is copied to low-power CMOS RAM, the CPU is disabled, and all clocks are stopped. System power remains enabled but the hardware is effectively dormant. Typically the PC keyboard includes switches or pushbuttons with which the user can select the desired power conservation mode.

Note that operation can switch between full-power, standby, or suspended mode without losing information. With the proper hardware and software support, a PC can be shut down when idle, and later resume operation from the exact point at which it left off.

Power Management Hardware Requirements

Most power reduction techniques require some level of hardware support. Word processors, spread sheets, and other standard application programs give no indication when a peripheral is no longer being actively used. Decisions concerning when and how to disable peripherals, alter system operating modes, or adjust CPU operating frequency must therefore be made using dedicated hardware to monitor software behavior.

Applications software also does not offer advance warning when a long-idle drive must be revived. Attempts to read or write a disk would malfunction if the drive had previously been disabled or turned off. The access must instead be intercepted, the drive must be re-energized and brought up to speed, and the original access must be reinitiated if the transfer is to succeed. This means the I/O system must provide some mechanism for interrupting the CPU if I/O operations to disabled peripherals are detected.

Restarting a halted drive itself takes time and wastes a certain amount of power, so power should not be removed too soon from peripherals in relatively frequent use. It's best not to disable such devices until they have been idle for some designated period of time. This requires additional hardware support, such as activity timers that keep track of how long it's been since each peripheral was last used.

Even with special external hardware, however, it's difficult to build a fully power-efficient PC using conventional components. Most microprocessors specify a minimum frequency at which they must be run. Unless the CPU clock frequency is reduced drastically or stopped entirely, the CPU will continue to draw some considerable fraction of its full-speed, worst-case power.

Power Management Hazards to Software Compatibility

Power-management hardware must necessarily make use of certain system resources, including input ports to detect system status, output ports to control hardware operating modes, interrupt request inputs to detect asynchronous events, and program and data memory for power management service routines and control algorithms. Portable PCs based on conventional microprocessor components must "steal" these resources from among those available to all software. Power-related interrupt requests, for example, may be routed through an interrupt request input pin also used by other system functions and applications programs.

It's a prerequisite of any portable computer, of course, that it remain fully compatible with existing software. Forcing a single interrupt pin or other resource to do double-duty for power-management creates a serious potential hazard. It's possible for an interrupt service routine to handle both the original interrupt function and new power-management requirements, but it may not always do so reliably.

The problems that sometimes arise resemble those encountered by the DOS operating system when TSR (terminate and stay resident) I/O drivers and utility programs written by different vendors are intermixed: if multiple TSRs manipulate the same interrupt vector, or contend for the same memory region, each can interfere with and corrupt the functions performed by the others. As a result, software developed for conventional desk-top computers may not run properly on portable PCs built using conventional technology.

These hazards are most serious with 32-bit operating systems such as OS/2 and Windows-386, so-called "DOS-extender" utilities, and high-end application programs that enable the Intel386 architecture's protected operating mode. Such programs generally make more frequent changes to the processor interrupt vectors and install their own service routines for each request type. Efforts to mix power-management software with other service routines can malfunction if different interrupt handlers are alternately activated and disabled by software. Power-management-related firmware cannot seize control of the interrupt system or remap service-routine vectors without running the risk of breaking the rest of the system.

Hardware Support for Power Management

The Intel386 SL Microprocessor SuperSet combines a number of features to make system power usage as simple, efficient, and seamless as possible. These features include basic capabilities to reduce component power usage, to determine when power should be disabled to certain subsystems, and to ensure full compatibility with all existing software. Moreover, the SL SuperSet extends the Intel386 microprocessor architecture in ways that provide OEMs with increased system flexibility while avoiding the software problems encountered with conventional processors. The following sections describe each of these features.

Intel386 SL Microprocessor SuperSet Power Management Strategy

Individual elements of the SL SuperSet support each of the power-reduction methods mentioned above. Under a temporary worst-case full-power condition, a typical SL SuperSet-based PC may draw on the order of 12 Watts. If operating in a mode comparable to the standby mode of conventional PCs, power consumption falls by about 95%, improving battery life twenty-fold. In the equivalent of suspended mode, consumption falls by another factor of 20. While a hand-held PC's battery might provide only 4 hours of continuous full-power operation, the same charge may sustain a suspended PC's memory for up to two months.

But while conventional portable systems define a limited number of system-wide operating modes, and may require the operator to select explicitly the desired mode, OEM-developed firmware running on the SL SuperSet makes possible a virtually unlimited number of intermediate reduced power modes. Each system element can be switched between its various modes as appropriate, so power-hungry elements can spend as much time as possible in the lowest-power mode allowed. Slowing or stopping the clock to the CPU and peripherals when they are only moderately busy can cut their power consumption to near zero without otherwise affecting system behavior.

Ideally, system elements can switch seamlessly between their various frequencies or modes, without the operator having to intervene or even be aware as the transitions occur. With sufficient hardware flexibility and the proper control software, an SL SuperSet based PC can provide an arbitrary number of intermediate standby and suspended modes. When a system has been idle for a given period, for example, the PC could automatically power itself down. When the operator returns, reopens the computer lid, strikes a key, or moves the mouse, for example, the dormant computer can automatically reawaken itself and continue executing from the exact point at which it had left off. Each element of the SL SuperSet has been designed to help the OEM developer achieve these goals.

Central Processor Power Reductions

The microprocessor at the heart of the SL SuperSet has been fully redesigned for low-frequency and fully-static operation. In effect, each CPU clock edge charges or discharges a particular set of circuit nodes with some fixed number of electrons. The average power consumed by the processor depends on total electron flow, and is therefore directly proportional to CPU clock rate.

One technique for adjusting CPU power is simply to adjust its clock frequency to match computational load. The CPU may run at its fastest rate when the computer is in active use, but switch itself to a lower frequency after extended periods with no operator input. Or, if conserving battery life is most critical, the CPU may run at a slower frequency during most processing sessions, but switch to the maximum frequency (sometimes called “Turbo Mode”) for computation-intensive spread-sheet updates or database searches at the operator’s request.

The CPU can also disable its clock entirely when no computations are needed. When the Intel386 SL processor is fully suspended, a special low-frequency, low-power timer in the 82360SL component controls DRAM refresh rates to keep DRAM memory contents intact.

Math Coprocessor Power Reductions

The frequency options available to the main CPU also apply to the optional Intel387™ SX Math CoProcessor. Depending on the programs being run, the Intel387 SX Math CoProcessor can run flat out, at a reduced frequency, or be disabled entirely. CPU and coprocessor operating frequencies can be adjusted independently.

In conventional system designs, the coprocessor frequency is constant, though the device is referenced only occasionally. Even floating-point-intensive applications use the coprocessor only part of the time. Spreadsheet updates may use floating-point extensively, for example, yet the coprocessor sits idle for long periods during data entry and formula redefinition.

The SL SuperSet has provisions to greatly reduce the power consumed by the Intel387 SX Math CoProcessor under these conditions. The Intel386 SL microprocessor automatically adjusts coprocessor frequency dynamically, according to the operations encountered in the instruction stream. Control logic in the SL SuperSet lowers the coprocessor clock frequency automatically to its slowest rate except while floating-point computations are actually in progress.

Peripheral Power Reductions

Like the Intel386 SL processor, the Intel360 SL Peripheral I/O Subsystem is designed to be fully static, so its clocks can be slowed or stopped without losing data. The SL SuperSet also contains considerable additional circuitry to monitor the behavior of executing programs, determine when various peripherals have been idle for a given period, and control the power to those peripherals as appropriate.

A special block of dedicated circuitry detects software events that involve any of six ports, registers, or I/O devices. Events are defined as a read or write operation to any arbitrary range of I/O port addresses. The event detection logic may typically be configured at initialization time to sense accesses to port addresses corresponding to the floppy disk, hard disk, keyboard, printer, modem, and network controller, for example, but the hardware itself is generic and can sense accesses to any range of addresses. For example, software operations that involve the hard disk may be detected by setting the event logic to the range of addresses occupied by the hard-disk interface. Events involving a stylus and graphics tablet might be detected by initializing the same logic to detect references to tablet-interface port addresses.

When an event is detected, hardware generates an internal interrupt, after which OEM-developed service-routine firmware can decide whether special processing is required. In addition, six separate timers in the 82360SL can keep track of how long it has been since each peripheral—a hard-disk or diskette drive, for example—was last accessed. Each timer can count down programmable periods in four-second increments, defined independently by OEM-developed firmware. When an application program accesses the peripheral, the corresponding activity timer is automatically reset. When the activity timer expires, indicating the peripheral is idle, the timer invokes a service routine to retrieve and save the peripheral's status (if necessary) and remove power to the device.

When power-management firmware determines that a peripheral has not been accessed for a sufficiently long time, power to the devices of the related subsystem can be disabled. The simplest way to control peripheral device power is via the output pins of the general-purpose ideaPort interface. This port provides six uncommitted output pins designated SMOUT0 through SMOUT5. These pins are typically used to switch power directly to the peripherals tracked by the event detection logic, as shown in Figure 4-2.

The ideaPort interface pins are manipulated only under firmware control, however, so the correspondence between the pins and peripherals can vary as needed. For example, unused control pins can serve as general-purpose outputs, driving indicator lights, alarms, and battery switching circuitry. Elaborate power-control systems may program the interface to multiplex power-control command codes with other system support functions on the same pins.

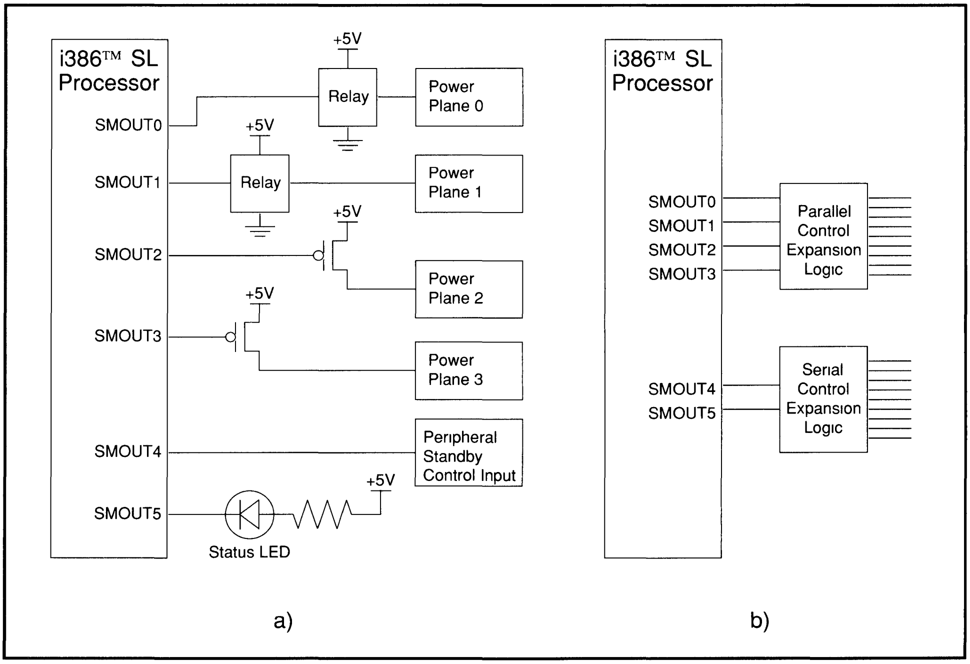


Figure 4-2. The ideaPort Interface Supports Multiple Power Control Options: a) Direct Peripheral Device Control; b) Parallel and Serial Control Expansion

Memory System Power Reductions

Other than the display and mechanical peripherals, the largest single power drain in a portable PC is the main memory system. The SL SuperSet can reduce memory-system power consumption both during normal operation and in a number of special power-down modes. SRAM-based main memories consume essentially no power except when a particular device is accessed. DRAM power consumption is approximately proportional to the number of access cycles performed. In each case, eliminating unnecessary accesses to main memory will significantly reduce the power it consumes.

In non-cached systems, the number of main-memory cycles performed is related to the CPU clock frequency. Slowing the CPU clock during periods of relative inactivity as described above will therefore also reduce the number of memory accesses performed.

At times it's preferable for the CPU to run at top speed until a program task is completed, then disable itself entirely to save power. At other times the CPU may need to run continuously, in which case it is most power-efficient to slow the clock. With the proper power-management firmware, SL SuperSet-based systems can support both alternatives.

All memory accesses are optimized for low-power operation by eliminating unnecessary bus cycles, precharge cycles, or control line toggling. The DRAM memory controller always transfers data with the most power-efficient mode allowed. There are separate RAS# and CAS# signals for each of four banks of DRAMs, and for the high- and low-order bytes within each bank. Only the devices required for each specific transfer are activated, and the others remain idle. When successive transfers involve data from the same page of the same device, page-mode transfers eliminate unnecessary address clocking cycles.

The DRAM control logic performs CAS# before RAS# refresh cycles, which disables DRAM output drivers in order to consume less power. The refresh rate is programmable, so DRAMs are refreshed at the slowest and most power-conscious rate allowed, independent of the CPU mode and frequency.

In SRAM-only systems, memory refresh can be disabled entirely to save power. Control signals normally produced to refresh memory accessed through the ISA expansion bus can also be optionally disabled.

Graphics Display System Power Reductions

The 82760SL control circuitry applies many of the same techniques of power conservation as the main memory. The DRAM display buffer operates in the most power-efficient mode and is refreshed at the slowest rate its device specifications permit.

The 82760SL also contains idle time-out counters that operate much like those for disk drives and other peripherals. Unlike the peripherals, however, the VGA activity timer detects references both to I/O port addresses and memory regions allocated to display buffers. After a predetermined period of inactivity, power to LCD backlights or the light-emitting elements of an active display can be switched off, while keeping the display control circuitry alive and while continuing to refresh the display buffer memory.

Software Support for Power Management

Each of the new SL SuperSet hardware facilities for power-reduction has been designed to circumvent the hazards found in conventional design techniques. Instead of appropriating some aspect of the original architecture that may need to be shared with existing software, these new capabilities are supported by an extension to the basic Intel386 architecture, a new operating mode called System Management Mode. All of the resources of the original 8086, 80286, and Intel386 Family processors are thus preserved intact, ensuring full compatibility with all existing application programs and operating systems.

System Management Interrupt

System Management Mode can be entered only through a System Management Interrupt request. Each of the power-related events discussed above—references to designated port-address ranges, underflows of device activity timers, or external interrupts on the ISA system bus—can optionally invoke an SMI. Whether or not a power-management event triggers an SMI is determined by the state of the SMI Trap Control Register. Bits may be set or cleared in this register to disable or enable interrupts from each source, as shown in Figure 4-3.

In addition to the I/O event-detection latches and idle-device counter underflows, external logic can assert an external pin to force a system-management interrupt. This pin lets system designers use the Intel386 SL processor's System Management extensions for purposes other than just power management, such as data security or supervision or network communications.

SMI requests are higher priority than all other interrupts in the SL SuperSet system, including Non-Maskable Interrupts. When an enabled SMI event occurs, its service routine is therefore guaranteed to execute immediately.

Private System Management Memory

The Intel386 SL SuperSet provides several new resources exclusively for power-management functions. A separate memory region between addresses 030000H and 03FFFFH is enabled only while system management software executes. Physically, this region can be part of the main memory array, physically separate devices, or accessed through the system backplane.

Execution of special SMI trap-handler software begins within the SMI memory region, and a special stack is automatically enabled to preserve CPU state and hold subroutine nesting information without affecting the regular system stack. A new RSM (resume) instruction has been added to the Intel386 SL microprocessor instruction set to designate completion of the SMI software routine.

By keeping the resources used for SMI trap handling separate from those used by conventional software, power-management interrupts can be handled totally transparently to existing programs. SMI functions are therefore guaranteed not to interfere with normal software operation.

SMI Mode Independence

Each microprocessor in the Intel386 Family support a variety of basic operating mode: "real mode", to which the processor defaults following reset, "native" or "protected" mode, in which memory management and certain other 32-bit features are enabled, and two "virtual" modes that simulate the operation of the 8086 and 80286 microprocessors. Each mode provides a different combination of system address limits, protection mechanisms, and other capabilities.

Whichever mode is enabled when a system management interrupt request is detected, the Intel386 SL processor switches back into real mode operation when the service routine begins executing. Since system management software always runs in the same mode, OEM firmware only needs to provide a single set of SMI service routines. Since real mode is essentially a subset of each of the other modes, it is generally the one for which software development is most straight-forward. SMI firmware developers therefore need not be concerned with the virtual memory system, page translation tables initialized by other tasks, interprocess protection mechanisms, and so forth.

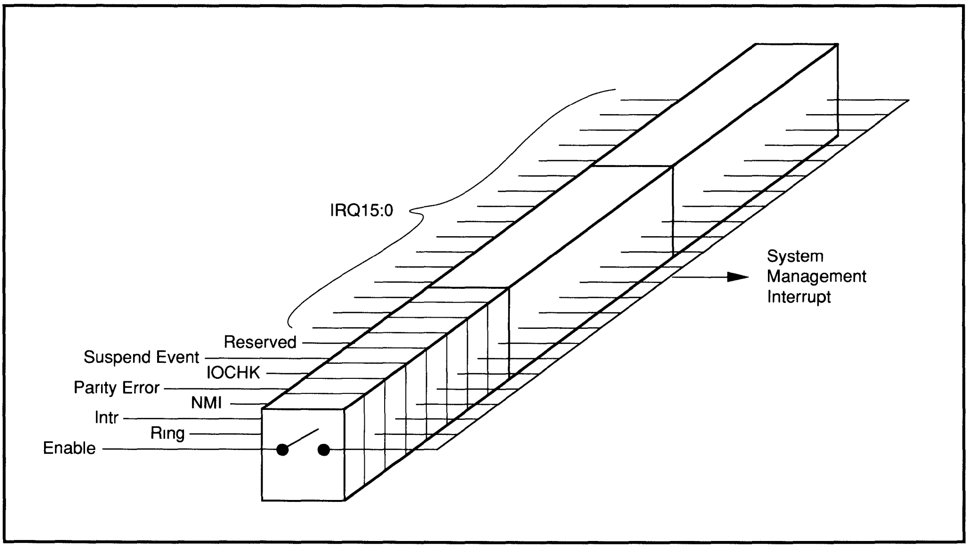


Figure 4-3. System Management Interrupt Requests and Mask Registers

Power Management Software Design

Software to configure and maintain the SL SuperSet components and to initialize and maintain the system-management facilities performs three distinct functions. These are described briefly below.

System Management Mode Initialization

When power is first applied to an Intel386 SL microprocessor based system, peripheral control registers must be initialized to match the desired characteristics of the system hardware. The ideaPort interface may need to be be initialized to enable the devices present, event detection logic should be set to the range of port addresses that correspond to each device, and idle-device timers should be initialized with appropriate time-out intervals. The SMI handler software, which initially resides in non-volatile EPROM memory, may also need to be copied to a protected region of system-management RAM during initialization. Alternatively, Flash EPROM devices may be used to hold system-management software, allowing simple field updates and software upgrades. Figure 4-4 shows the flow of a simple system initialization routine.

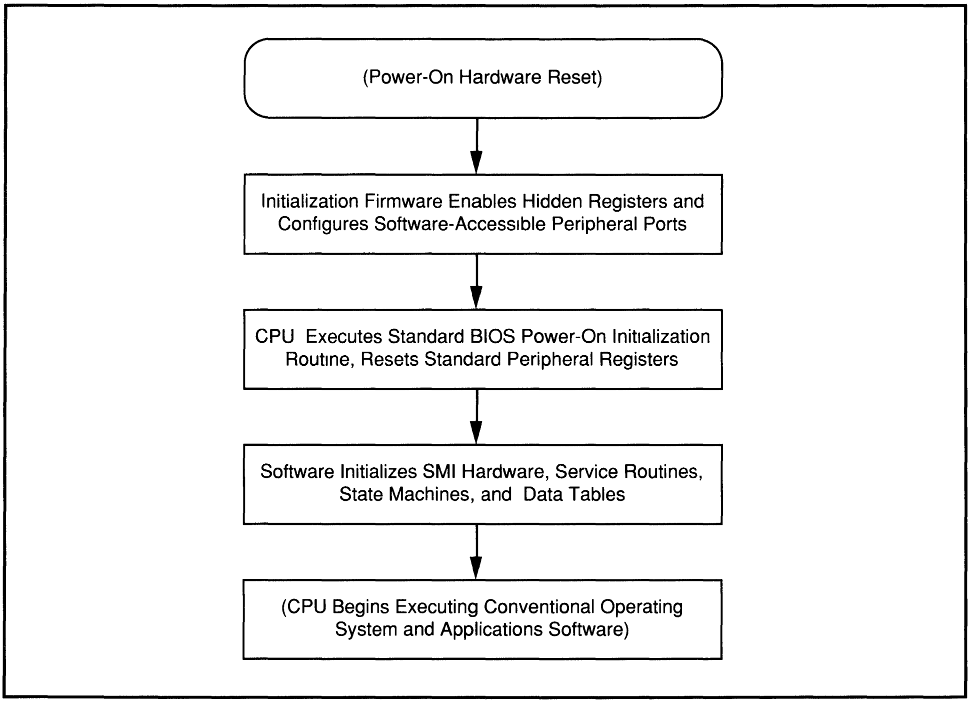

(Power-On Hardware Reset)
Initialization Firmware Enables Hidden Registers and Configures Software-Accessible Peripheral Ports
CPU Executes Standard BIOS Power-On Initialization Routine, Resets Standard Peripheral Registers
Software Initializes SMI Hardware, Service Routines, State Machines, and Data Tables
(CPU Begins Executing Conventional Operating System and Applications Software)

Figure 4-4. System Management Mode Initialization Sequence

Entry and Exit to SMI Service Routines

When an SMI request occurs, the Intel386 SL processor automatically enables the System Management RAM memory region, stacks the complete state of the CPU, and begins executing the SMI service routine. Within the service routine, software can determine what source initiated the SMI request and handle the request accordingly, enabling or disabling device power and saving or restoring peripheral and machine state as needed. If the SMI service routine determines the interrupt was initiated by an activity timer underflow or other event that requires disabling power to a peripheral, the handler retrieves and saves the peripherals current state before disabling power. This process is shown in Figure 4-5.

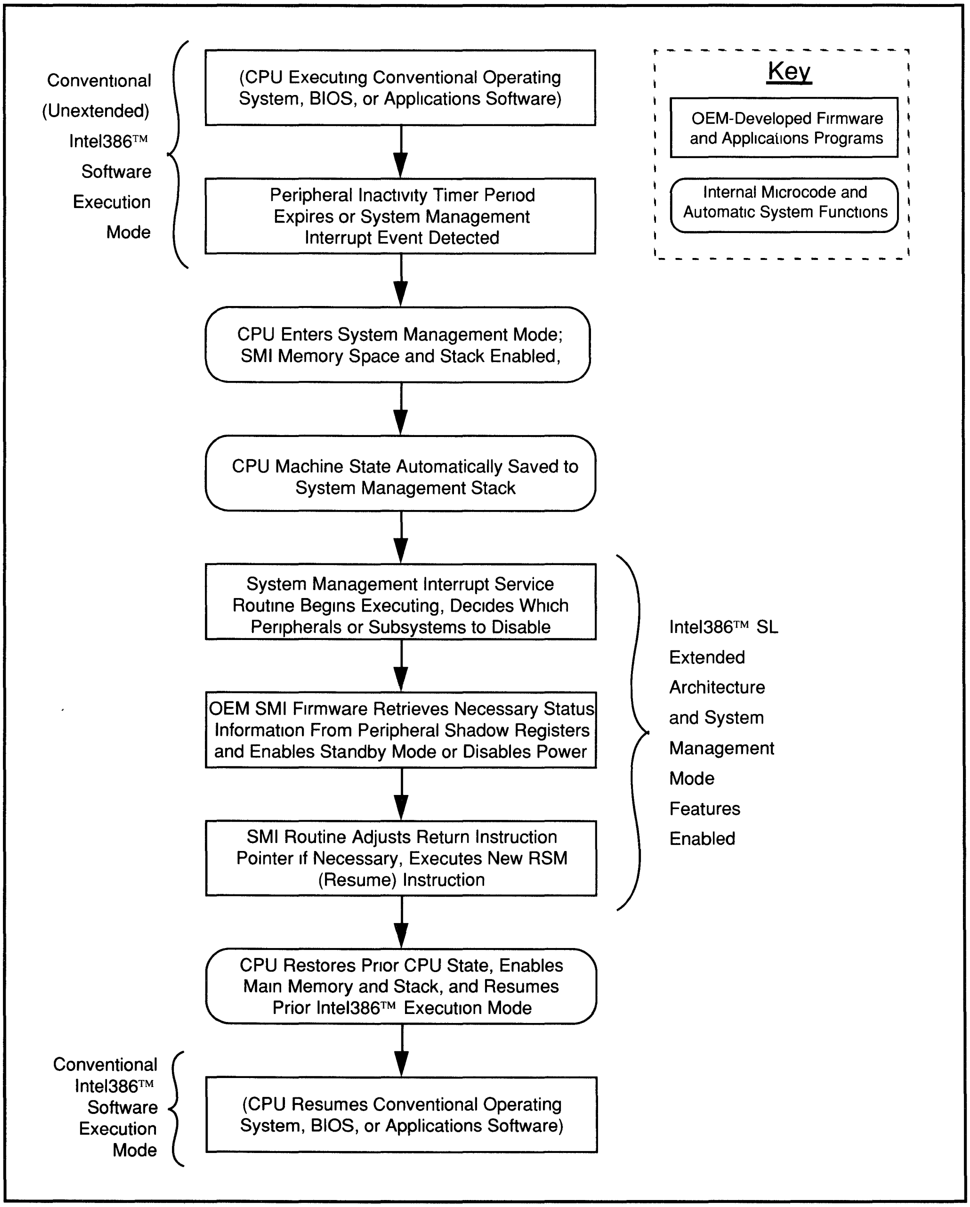

Conventional (Unextended) Intel386™ Software Execution Mode
(CPU Executing Conventional Operating System, BIOS, or Applications Software)
Peripheral Inactivity Timer Period Expires or System Management Interrupt Event Detected
CPU Enters System Management Mode; SMI Memory Space and Stack Enabled,
CPU Machine State Automatically Saved to System Management Stack
System Management Interrupt Service Routine Begins Executing, Decides Which Peripherals or Subsystems to Disable
OEM SMI Firmware Retrieves Necessary Status Information From Peripheral Shadow Registers and Enables Standby Mode or Disables Power
SMI Routine Adjusts Return Instruction Pointer if Necessary, Executes New RSM (Resume) Instruction
CPU Restores Prior CPU State, Enables Main Memory and Stack, and Resumes Prior Intel386™ Execution Mode
(CPU Resumes Conventional Operating System, BIOS, or Applications Software)
Conventional Intel386™ Software Execution Mode
Intel386™ SL Extended Architecture and System Management Mode Features Enabled
Key
OEM-Developed Firmware and Applications Programs
Internal Microcode and Automatic System Functions

Figure 4-5. System Management Interrupt Service Routine to Disable Peripheral Power

While a peripheral device is powered down, read and write operations to the device are ineffective. The device cannot safely be accessed again until power has been restored and normal operation resumed. When software attempts to reference a port that has been disabled, hardware within the SL SuperSet automatically detects the attempt, saves the offending operation code and port address in dedicated registers, and generates a new SMI to the CPU. Power to the device can then be re-enabled under software control, the peripheral can be given time to resume proper operation, and the offending operation can be repeated before returning control to the point of interruption. The entire software sequence needed to resume peripheral operation is thus totally transparent to the PC operating system and application software. This process is shown in Figure 4-6.

In either case, the service routine completes SMI processing by executing the special RSM (resume) instruction. The Intel386 SL processor automatically restores full processor status from the System Management RAM, and then disables the SM-RAM memory space. Execution then returns to the point in the original program at which the original SMI request had occurred.

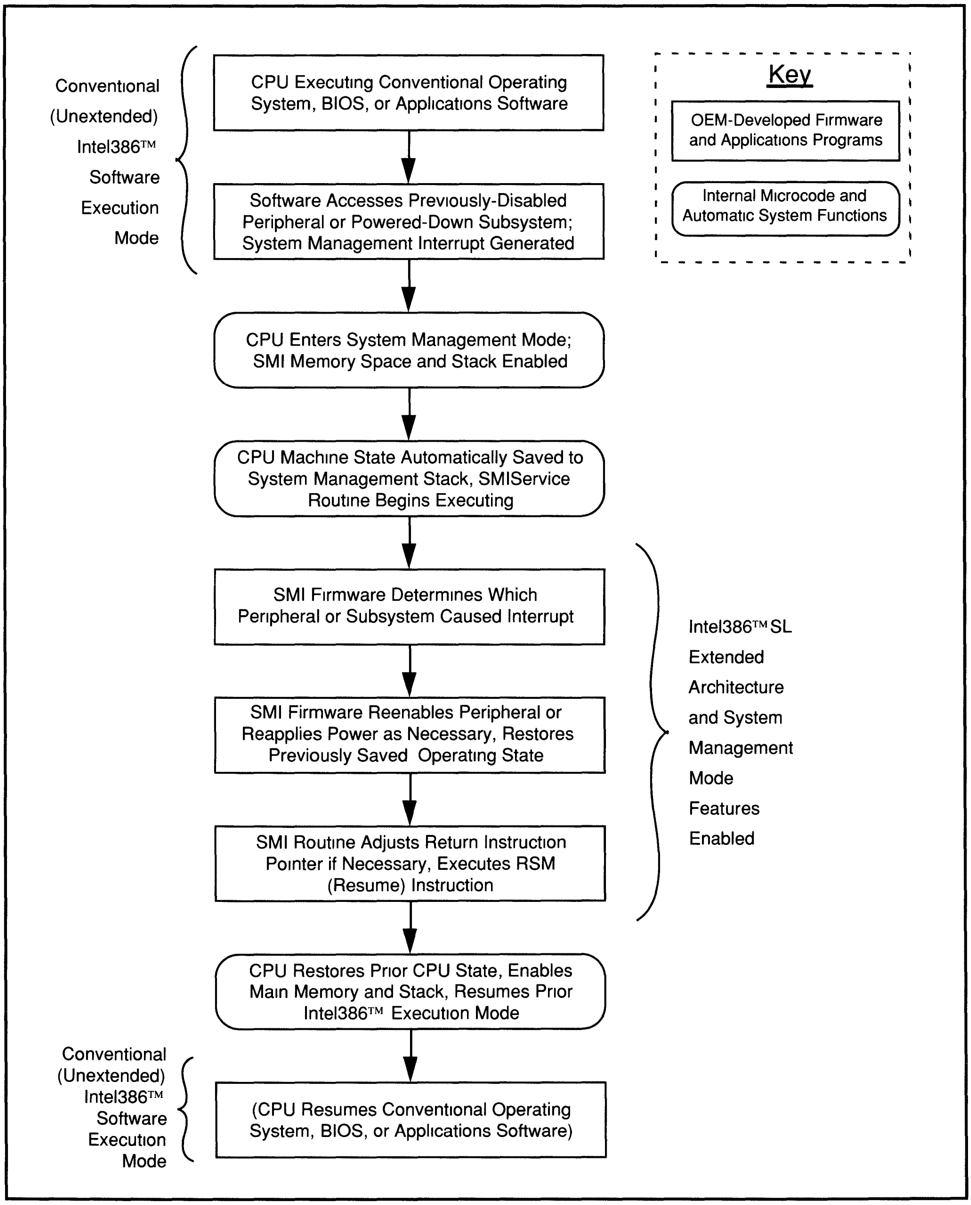

Key
OEM-Developed Firmware and Applications Programs
Internal Microcode and Automatic System Functions
Conventional (Unextended) Intel386™ Software Execution Mode
CPU Executing Conventional Operating System, BIOS, or Applications Software
Software Accesses Previously-Disabled Peripheral or Powered-Down Subsystem; System Management Interrupt Generated
CPU Enters System Management Mode; SMI Memory Space and Stack Enabled
CPU Machine State Automatically Saved to System Management Stack, SMIService Routine Begins Executing
SMI Firmware Determines Which Peripheral or Subsystem Caused Interrupt
SMI Firmware Reenables Peripheral or Reapplies Power as Necessary, Restores Previously Saved Operating State
SMI Routine Adjusts Return Instruction Pointer if Necessary, Executes RSM (Resume) Instruction
Intel386™ SL Extended Architecture and System Management Mode Features Enabled
CPU Restores Prior CPU State, Enables Main Memory and Stack, Resumes Prior Intel386™ Execution Mode
Conventional (Unextended) Intel386™ Software Execution Mode
(CPU Resumes Conventional Operating System, BIOS, or Applications Software)

Figure 4-6. SMI Service Routine to Re-Enable Peripheral Power

Power Management Hardware Design Issues

Most digital systems turn power on or off as a unit, so all components are energized or disabled together. In contrast, portable computers may energize the main CPU, display unit, and disk drives separately to save power. Most logic families are not designed to intermix powered and unpowered devices; since the I/O characteristics of unpowered chips are undefined, unpowered devices may draw current from other device outputs. High-voltage levels on unpowered CMOS inputs can cause latch-up and thermal runaway which can damage the device. In the past, these hazards have made it difficult to switch power safely to individual subsystems.

The Intel386 SL Microprocessor SuperSet has been designed to take into account power-switching factors between SuperSet components and external circuitry. Intel SL SuperSet outputs include push-pull and three-state drivers, open source and open-drain outputs, and built-in pull-up and pull-down resistors on critical pins as needed. With the proper design, SL SuperSet outputs never attempt to drive unpowered external devices, nor do SL SuperSet components attempt to read input pins floating at an indeterminate voltage. Collectively, these features let systems with complicated combinations of subsystem power control be built with complete reliability and integrity, with a bare minimum of external glue logic.

Power Management Summary

While the Intel386 SL Microprocessor SuperSet provides an assortment of dedicated hardware facilities for power management, these facilities are controlled and coordinated via firmware developed by each system OEM. Normally power-management software would share interrupt vectors with other system functions, and consume some portion of the instruction and data memory normally available to other programs. Appropriating processor resources that were originally used by or available to conventional software creates compatibility hazards with some operating systems and applications programs.

The architectural extensions implemented by the SL SuperSet processor eliminate these hazards. Power-management events invoke a new interrupt type, not present in the original Intel386 architecture. As the service routine code for this request begins executing, the CPU automatically enables a region of system memory not visible to conventional software. Since all power-management related software and data is concealed from existing programs, there is no danger that existing software will malfunction when executing on an SL SuperSet-based portable computer.

Chapter 5
System Design Examples

CHAPTER 5
SYSTEM DESIGN EXAMPLES

As stated in the Overview chapter, the Intel386 SL Microprocessor SuperSet is well suited for a wide spectrum of PC system configurations. This chapter shows example block diagrams of four possible system designs, with the capabilities, system complexity, and approximate chip count shown for each.

Ultra-Compact "Palm-Top" PC

Figure 5-1 shows the block diagram for the simplest possible computer built with the SL SuperSet. This design consists of just ten devices: the basic SL SuperSet, SRAMs for main memory, a Flash EPROM for the BIOS and operating system firmware, interface buffers for modem and printer connections, and keyboard and graphics controllers.

Such systems are sometimes called "palm-top" computers. With surface-mount assembly techniques, the assembled system board can be as small as 2-1/2" x 6". System memory includes 128K bytes of EPROM for the system and applications software, plus 512K bytes of data storage. Additional data storage and applications software can be added by inserting credit-card-sized expansion boards.

The control software for this product may be a ROMable version of standard DOS or fully customized operating system package. Software can be downloaded from standard DOS diskettes with a conventional PC serving as the palm-top system's "host." Text and data files can be transferred to or from the host using a direct serial cable connection. This system could operate from standard penlight batteries.

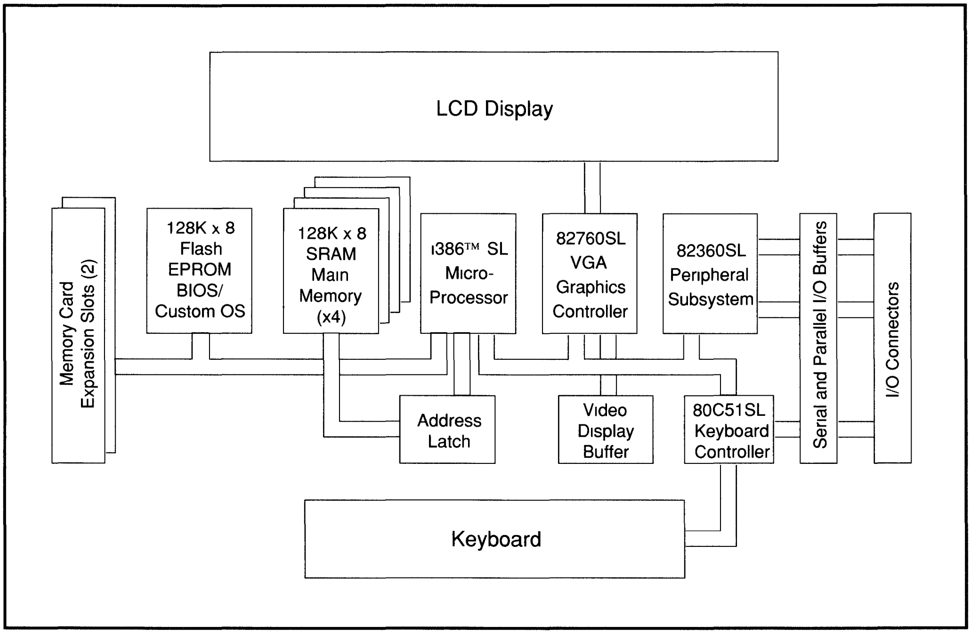

LCD Display
Memory Card Expansion Slots (2)
128K x 8 Flash EPROM BIOS/ Custom OS
128K x 8 SRAM Main Memory (x4)
i386™ SL Micro-Processor
82760SL VGA Graphics Controller
82360SL Peripheral Subsystem
Serial and Parallel I/O Buffers
I/O Connectors
Address Latch
Video Display Buffer
80C51SL Keyboard Controller
Keyboard

Figure 5-1. Ten-Chip Palm-Top PC Block Diagram

Note-Pad PC

The note-pad PC shown in Figure 5-2 is a larger and more powerful version of the palm-top system shown above, about the size of a thick pad of writing paper. The full-size LCD display emulates VGA graphics modes. The system does not have a conventional PC keyboard; instead, data is entered using a touch screen or stylus for hand-written input.

Main memory is built with DRAMs to grow capacity to 2 Megabyte using 1M x 4 devices. But a cache has been added to maintain low system power. A Flash EPROM contains the BIOS, power-management software, and standard ROM-DOS.

An optional 3-1/2” external diskette drive assists in exchanging programs and data with other computers. Since the system is fully DOS-compatible, standard business software can be loaded from diskettes. The system also includes connectors for standard PC peripherals, including serial and parallel ports, and a separate, full-sized keyboard. This system is powered by custom rechargeable batteries built into the case.

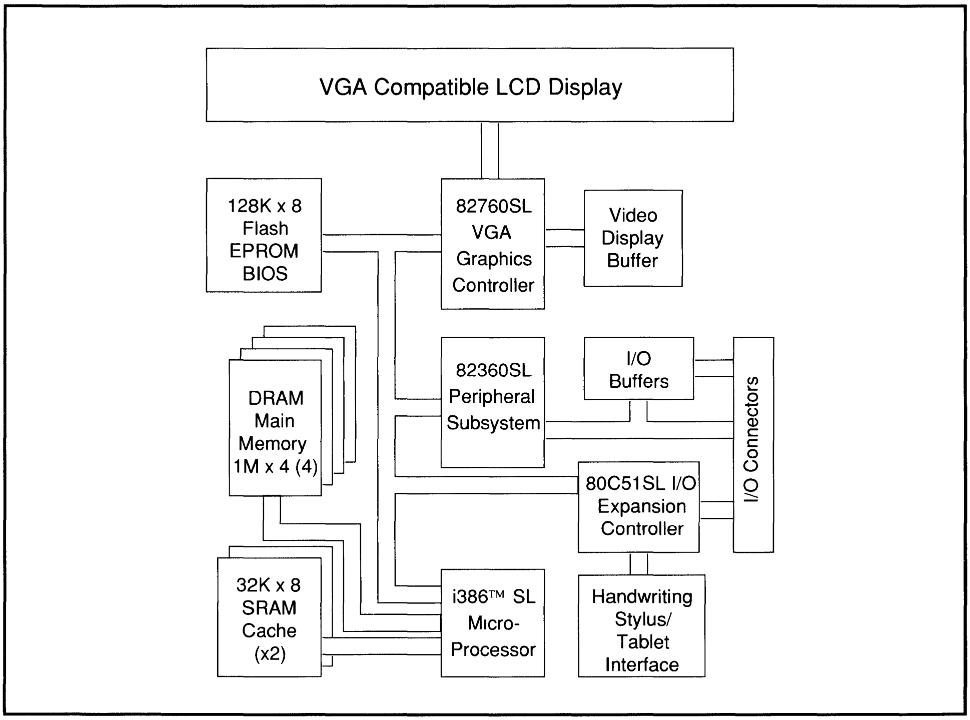

VGA Compatible LCD Display
128K x 8 Flash EPROM BIOS
82760SL VGA Graphics Controller
Video Display Buffer
DRAM Main Memory 1M x 4 (4)
82360SL Peripheral Subsystem
I/O Buffers
I/O Connectors
80C51SL I/O Expansion Controller
32K x 8 SRAM Cache (x2)
i386™ SL Micro-Processor
Handwriting Stylus/ Tablet Interface

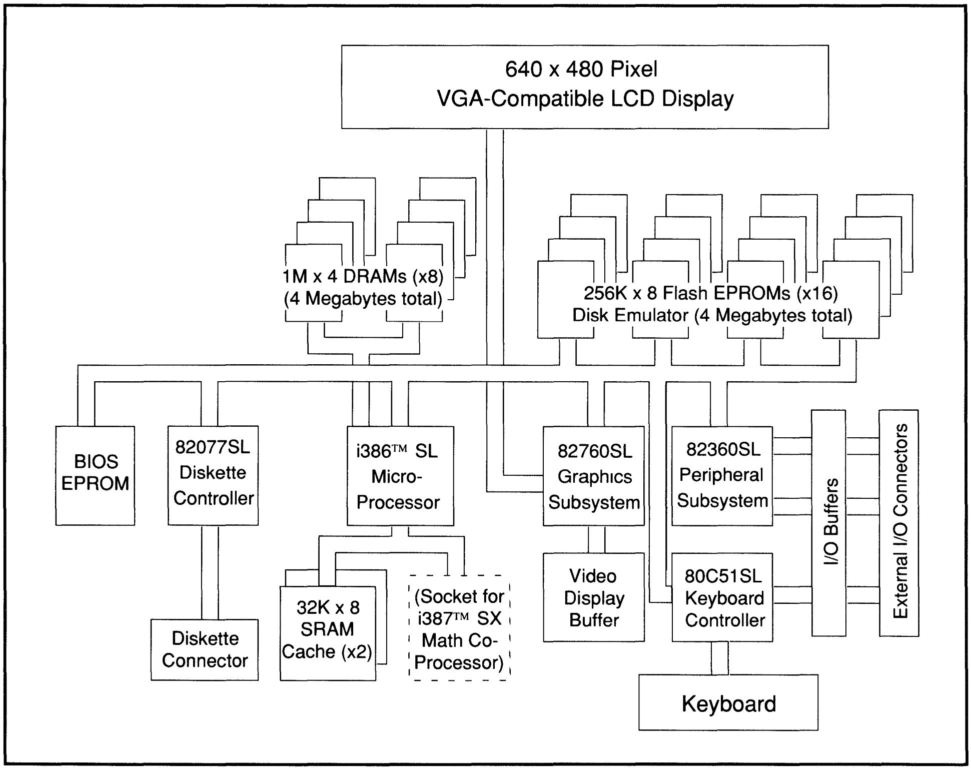
Figure 5-2. Note-Pad PC Block Diagram

Notebook PC

Figure 5-3 shows a block diagram for a "notebook" computer that is somewhat more flexible than the "note-pad" system. Main memory capacity has grown to 4 Megabytes using 1M x 4 surface-mount DRAMs, with an SRAM cache included to reduce DRAM power and extend battery life.

A 4-Megabyte Flash EPROM disk emulator holds the operating system and the most frequently-used applications programs, while a standard 3-1/2" diskette drive provides a more effective method for interchanging software and data with larger, desk-top computers. A socket for an optional Intel387 SX CoProcessor improves performance of spreadsheets and other math-intensive applications.

The overall dimensions of this configuration are approximately 8" x 11" x 1-1/2". Its keyboard is considerably larger that the palm-top's, and its monochrome LCD displays VGA graphics. The stystem could run off replaceable, rechargeable batteries.



640 x 480 Pixel
VGA-Compatible LCD Display
1M x 4 DRAMs (x8)
(4 Megabytes total)
256K x 8 Flash EPROMs (x16)
Disk Emulator (4 Megabytes total)
BIOS EPROM
82077SL Diskette Controller
i386™ SL Micro-Processor
82760SL Graphics Subsystem
82360SL Peripheral Subsystem
I/O Buffers
External I/O Connectors
32K x 8 SRAM Cache (x2)
(Socket for i387™ SX Math Co-Processor)
Video Display Buffer
80C51SL Keyboard Controller
Diskette Connector
Keyboard

Figure 5-3. Notebook PC Block Diagram

Lap-Top PC

Figure 5-4 shows a larger and considerably more powerful "lap-top" computer. The basic main memory in this case consists of four megabytes of surface-mounted DRAMs attached to the main system board. Sockets for four additional DRAM SIMMs (single in-line memory modules) let the end user expand system memory to 5, 8, or 20 megabytes total. An Intel387 SX Math CoProcessor is standard.

Secondary storage in this system includes both a 3-1/2" diskette drive and a 40 megabyte hard disk. The display emulates VGA graphics on a monochrome screen with a resolution of 640 x 400 pixels. On the road, power is supplied by a built-in lead-acid "gel-cell" battery; at home or in the office the PC connects to an A.C. adapter that provides power and recharges the battery.

Four expansion slots support standard ISA memory and I/O expansion boards. The system can attach to a full-sized keyboard and/or mouse through connectors on the back, and the graphics control logic automatically reconfigures itself to drive an external color CRT if present.

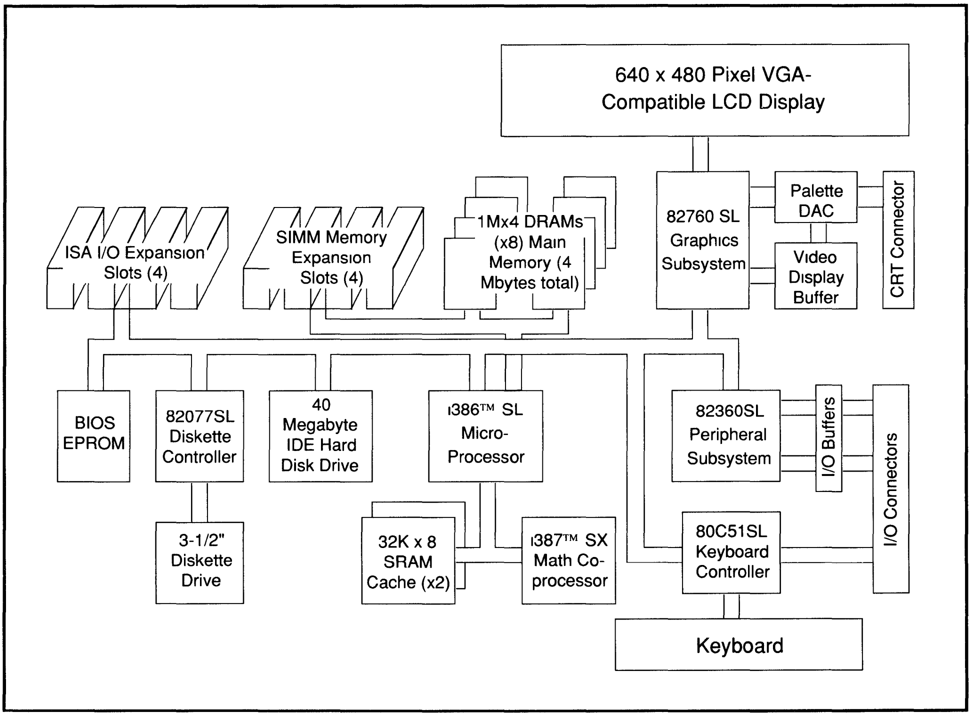

640 x 480 Pixel VGA-Compatible LCD Display
ISA I/O Expansion Slots (4)
SIMM Memory Expansion Slots (4)
1Mx4 DRAMs (x8) Main Memory (4 Mbytes total)
82760 SL Graphics Subsystem
Palette DAC
Video Display Buffer
CRT Connector
BIOS EPROM
82077SL Diskette Controller
40 Megabyte IDE Hard Disk Drive
i386™ SL Micro-Processor
82360SL Peripheral Subsystem
I/O Buffers
I/O Connectors
3-1/2" Diskette Drive
32K x 8 SRAM Cache (x2)
i387™ SX Math Co-processor
80C51SL Keyboard Controller
Keyboard

Figure 5-4. Lap-Top PC Block Diagram

UNITED STATES
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

JAPAN
Intel Japan K.K.
5-6 Tokodai, Tsukuba-shi
Ibaraki, 300-26

FRANCE
Intel Corporation S.A.R.L.
1, Rue Edison, BP 303
78054 Saint-Quentin-en-Yvelines Cedex

UNITED KINGDOM
Intel Corporation (U.K.) Ltd.
Pipers Way
Swindon
Wiltshire, England SN3 1RJ

WEST GERMANY
Intel Semiconductor GmbH
Dornacher Strasse 1
8016 Feldkirchen bei Muenchen

HONG KONG
Intel Semiconductor Ltd.
10/F East Tower
Bond Center
Queensway, Central

CANADA
Intel Semiconductor of Canada, Ltd.
190 Attwell Drive, Suite 500
Rexdale, Ontario M9W 6H8

Printed in U.S.A./0991/15K/CAP MW
MICROPROCESSORS

Order No. 240852-002