Wissenswertes rund um den Z280

Title: Z280, truly a microsystem in silicon
Author(s): Magill, J.
Author Affiliation: Zilog, Campbell, CA, USA
Conference Title: Midcon/87 Conference Record p.388-93
Publisher: Electron. Conventions Manage, Los Angeles, CA, USA
Publication Date: 1987 Country of Publication: USA vi+516 pp.
Conference Sponsor: IEEE; ERA
Conference Date: 15-17 Sept. 1987 Conference Location: Chicago, IL, USA
Availability: Western Periodicals Co., North Hollywood, CA, USA
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P); Product Review (R)
Abstract: Zilog's 'system on silicon' called the Z280 provides a 16-bit processor which runs Z80 code together with many other closely coupled advanced system components. Other major blocks include memory management, peripherals, memory refresh logic, cache memory, wait state generators, and a clock oscillator. The peripherals include just about everything else needed, apart from some RAM and ROM to put together a complete system; included are 4 DMA channels, 3 counter/timers, and a UART. Specialised glue logic ties all this together, making a closely coupled, high performance, low cost microsystem. To minimise pin count the Z280 communicates to the external world via a multiplexed address and data bus. (0 Refs)
Descriptors: CMOS integrated circuits; computer architecture; microprocessor chips; VLSI
Identifiers: timers; Zilog; counters; pin count minimisation; multiplexed bus; closely coupled system; microsystem in silicon; system on silicon; Z280; 16-bit processor; Z80 code; memory management; peripherals; memory refresh logic; cache memory; wait state generators; clock oscillator; DMA channels; UART; glue logic; low cost microsystem; multiplexed address and data bus; 16 bit
Class Codes: B1265F (Microprocessors and microcomputers); B2570D (CMOS integrated circuits); C5130 (Microprocessor chips)
Numerical Indexing: word length 1.6E+01 bit

TIS iServer. (Hardware Review) (one of three evaluations of fault-tolerant microcomputers in 'Servers that survive') (evaluation)
Lavin, Paul
PC User ISSUE: n164 PAGINATION: p39(1)
PUBLICATION DATE: July 31, 1991
ARTICLE TYPE: evaluation
AVAILABILITY: FULL TEXT ONLINE LINE COUNT: 00088
SPECIAL FEATURES: illustration; photograph; table
CAPTIONS: PC User verdict. (table)
SOURCE FILE: CD File 275
OPERATING PLATFORM(S): Intel 80486
PRODUCT NAME(S): TIS iServer (Microcomputer) - evaluation
COMPANY NAME(S): TIS - Products
DESCRIPTORS: Evaluation; Fault Tolerance; Microcomputers
SIC CODE: 3571
ISSN: 0263-5720

PC USER VERDICT

Products: TIS iServer
Description: Faul-tolerant Unix server.

Supplies by: TIS

Tel: (0628) 810909

Price: 13,995 pounds sterling

	Poor	Moderate	Good	Excellent
Characteristics		[unkeyable]		
Ease of use		[unkeyable]		
Documentation		[unkeyable]		
Engineering			[unkeyable]	
Performance			[unkeyable]	

* Expensive, but it comes from a pedigree open systems stable with vast-experience of multiuser systems. With TIS you get what you pay for.

TIS is a highly recommended Unix system Value Added Reseller (VAR), which found it couldn't get what it wanted for its Intel-based open systems customers off the shelf. The company commissioned Intel to produce a machine which it christened the iServer and is designed to provide the resources required by small Unix systems users.

The iServer is a short, broad box measuring 14x18x20 inches, reminding you of one of those valuables safes you sometimes find in hotel rooms. It arrives in your office strapped to a wooden pallet and it's better to get two people if you want to move it. The weight comes from the heavy-gauge steel construction and the UPS batteries.

The floppy disk and tape drives are hidden behind a door which isn't connected to the key lock on the top of the unit. The lock serves only to secure the system electrically. There aren't any alarms to tip the wink on an unauthorised intruder. The top fascia panel has LEDs for AC and DC power status, and LAN and serial I/O activity.

There are several strengths of iServers and I examined the top-of-the-line Model 425. It has a 25MHz 486DX chip which makes it considerably more powerful for a multiuser Unix installation than either the 325 or 333 models, which both use 386 CPUs. While the 386 machines get a 64Kb static RAM cache, the 425 has only the 8Kb internal cache in the 486.

There's no socket for an auxiliary floating-point co-processor, nor would you expect one on an Intel board. The motherboard is the same one that Intel puts into its SYP machines. I've used one of those Unisys/Intel boxes for a year with hardly a hiccup.

The standard memory configuration is 8Mb, but expansion to 32Mb is possible, using 8Mb expansion boards that slot into Intel's own 32-bit bus. This bus is compatible with 16 and 8-bit cards, but when you put a memory board in it, it speeds up to provide fast memory access.

There are four 32-bit slots, and if you fill them all with memory, you'll only have one 16-bit slot have one 16-bit slot and one 8-bit slot left after the inclusion of the SCSi and serial I/O controllers. That's not as dire as it sounds, given that the only other likely inhabitant of the iServer would be a 16-bit Ethernet card.

The 16-bit busmastering Adaptec 1540 SCSI controller, whichs sits in one of the 16-bit slots, can deliver 10Mbps DMA transfer. The maximum synchronous SCSI transfer rate is 5Mbps. A second disk controller can be

added for the protection and performance bonuses that duplexing can provide.

The serial I/O controller uses a 20MHz Z280 chip to drive up to four sub-controllers, each of which carries eight RS232 connectors. Eight ports are standard. The system also carries two additional serial ports labelled console, and a parallel printer port.

Standard mass storage ia a 5 1/4-inch floppy disk drive and a 150Mb QIC tape cartridge drive – something of a standard in the Unix world.

The chassis will accommodate up to two full-height SCSI hard disks. TIS offers options of 120, 325 and 650Mb, although drives that hold 2Gb are available in that form factor. A 1.3Gb DAT drive is an option for unattended large volume backups. If you go for the maximum security configuration (two drives mirrored with duplex controllers), the maximum capacity is 650Mb from four 325Mb half-height drives.

TIS provided a series of measurements using the standard AIM III suite, which establish the 486 iServer in the same league as the Olivetti LSX5010, and the Apricot FTs. The iServer beat the Altos 100/486 on every measure. The tests used were the Multiuser, Random Disk Performance, Sequential Disk Performance and CPU Performance.

Providing a measure of protection from the slings and arrows of the power company is a 300W power supply with a UPS battery backup, which can keep a fully loaded system running for 10 vital minutes so that the file system can be secured. The battery takes 24 hours to recharge.

The TIS iServer runs an enhanced version of AT&T's System V Release 4 Unix. The disk and I/O drivers have been massaged for reliability and performance and the UPS integrated, so that the system will perform an orderly shutdown if the mains power goes away. Disk mirroring software is standard, as is security that complies with the NSCS Orange Book C2 standard.

TIS' software bundle includes TCP/IP, NFS, Xenix and BSD compatibility, X.Window (X11R4) and Open Look. The iServer supports Portable NetWare as well as the ISO/OSI protocol stack.

The TIS iServer is a husky little creation that's been tailored specifically for small Unix systems users. I can't fault it technically in either quality or specification, but in relation to the other grown-up PCs reviewed here, it's pretty expensive.

Operating-system software. (includes a related article on accelerating system performance through GUI) (Software Review) (All-Star PC Part 4.) (evaluation)
Leibson, Steven H.
EDN VOL.: v35 ISSUE: n9 PAGINATION: p171(8)
PUBLICATION DATE: April 26, 1990
ARTICLE TYPE: evaluation
AVAILABILITY: FULL TEXT ONLINE LINE COUNT: 00278
SOURCE FILE: CD File 275
PRODUCT NAME(S): Open Desktop (Operating system); MS-DOS 4.01 (Operating system); OS/2 2.0 (Operating system); DESQview 386 (Utility program); Microsoft Windows-386 2.11 (Graphical user interface)

COMPANY NAME(S): Microsoft Corp. - products; Quarterdeck Office Systems - products; Perceptive Solutions Inc. - products; Santa Cruz Operation Inc. - products; Future Domain Corp. - products
DESCRIPTORS: Operating Systems; Software Selection; Evaluation; Comparison ; Graphical User Interface
SIC CODE: 3672; 7371; 7372; 3577
ISSN: 0012-7515
TICKER SYMBOL: MSFT
OPERATING-SYSTEM SOFTWARE

Operating systems get very little respect from users, probably because they get in your way more often than not. Cryptic commands and mysterious command-line switches make text-oriented operating systems difficult to usE. Yet these essential programs define how you interact with your computer. Microsoft's DOS is currently the most popular PC operating system, but it isn't the only player available for high-end Pcs like EDN's All-Star PC. Multitasking operating systems, including Unix and OS/2, are waiting in the on-deck circle.

Many people think DOS is getting long in the tooth. DOS and the PC were introduced together in 1981 and both have since evolved. DOS 1.0 mimicked Digital Research's CP/M operating system, which ran on the 8080 [microprocessor]. But DOS 2.0 gained a character all its own by incorporating hierarchical directories. The most recent release, DOS 4.01, overcomes many of the operating system's lingering limitations. It supports expanded memory, large hard disks, and a limited form of multitasking. The All-Star PC includes DOS 4.01 as one of its operating systems because of the many applications programs available for DOS-based systems.

DOS 4.01 allows the All-Star PC to use the entire 315M bytes of the Seagate Wren Runner hard-disk drive as one physical disk drive. Earlier versions of DOS could handle logical drives no larger than 32M bytes. DOS 3.3 split each of the All-Star PC's 315M-byte Wren Runners into 10 logical drives, a shortcoming that caused problems with some programs. DOS 4.01's solution is much cleaner.

Since DOS's introduction, the industry's definition of just what an OS should provide has evolved. In the heyday of CP/M and DOS 1.0, most PCs had either limited or no graphics hardware. Consequently, the operating systems for these machines provided very little support. Now however, graphics displays are integral to most PCs. Yet DOS still shuns graphics support. It clings to the past because its existing base of applications programs can't take advantage of new graphics services. DOS 4.01 provides a graphics-shell program that creates the illusion of a graphical user interface (GUI), but the applications programs that you activate from that shell must supply their own graphics drivers, like any DOS program. As a result, every DOS applications program looks and acts differently. There's nothing inherently wrong with differentiation, but a wide-open operating environment like DOS extracts a penalty in user inefficiency and in learning time.

Microsoft recognized years ago that DOS was falling short of contemporary operating-system expectancies, so it developed the Windows operating-system shell to add missing features. The earliest versions of Windows provided a standard set of graphics services that allowed applications programmers to produce software with a common look and feel. Unfortunately, Windows can only provide graphics services to programs that are specifically written to make use of them. No software shell can retrofit graphics on a program created ignorant of GUI concepts. But DOS programs still outnumber Windows applications programs by a substantial margin, so most existing PCs currently operate without the Windows GUI.

This situation may change however, as more 80386-and 80486-based PCs enter the workplace. Windows/386 version 2.11, the latest and most powerful version of the package, runs DOS applications. With the added hardware boost from the 80386-microprocessor] family, it also provides memory management and multitasking services. Thus Windows/386 provides a bridge between existing DOS applications and a GUI-based OS: DOS plus Windows.

Quarterdeck Office Systems recognized the limitations inherent in DOS and took a different approach to enhancing DOS's capabilities. The company's DESQview 386 operating-system shell for DOS provides multitasking and windowing services like Microsoft's Windows/386. QEMM (the Quarterdeck expanded memory manager for the 80386) is part of the DESQview 386 package; it provides memory-management services for DESQview. With QEMM, you can convert all memory above the first megabyte into so-called expanded memory (called block-switched memory in the days of CP/M). Many DOS programs can take advantage of expanded memory for large data structures, and the DESQview shell uses expanded memory to provide multitasking under DOS. Windows/386 performs similar feats with expanded memory. However, unlike Windows/386, DESQview doesn't bestow a graphics-interface standard upon application programs. Instead, its windows differ greatly in appearance from one to the next, depending on the user interface built into the underlying program.

EDN's All-Star PC runs both Windows/386 and DESQview 386, but it can only run one at a time because both programs want to manage the system's memory, an even that would lead to conflicts if both operated concurrently. To use both, you must keep two different CONFIG.SYS files on your hard disk and place the one you want in the root directory, then reboot your PC to activate the desired shell program. This procedure lacks any semblance of elegance, but it works.

Concurrency streamlines work

Windows/386 and DESQview 386 make PC users more efficienty by allowing many programs to run concurrently. In such an environment, PC users can jump from program to program instantly as warranted by the day's tasks and by those pesky interruptions everyone deals with. Both programs also allow processing to background tasks, so you can initiate a long processing job and then switch to another task while the background processing occurs. Far too many people state that most PC users do not need and cannot take advantage of concurrent-program operation. That's bunk! Concurrent-program operation allows you to leave one task to handle an interruption and later return to the same spot. Everyone deals with interruptions on a daily basis, so it's foolish to say that few can benefit from the advantages provided by programs like DESQview 386 or Windows/386.

Software compilation, for example, greatly benefits from background processing. Software developers have used background compilation on minicomputers and mainframe computers for decades. A multitasking Os allows you to jump from a debugger to an editor to a compiler and back to the debugger. Similarly, a hardware designer can jump from a schematic-capture package to a simulator to a pc-board-layout package to a thermal-analysis package and back again. Engineers can bery definitely benefit from concurrent-program operation and multitasking.

But DOs applications aren't really designed for multitasking environments and therefore lack features that could make them even more useful. For example, programs designed for a single-tasking environment aren't likely to contain facilities to communicate with other running programs. Only one program at a time can run under a single-tasking OS, so

it would be silly to include such capabilities. Yet real-time communications among programs can be quite useful, particularly for engineering applications.

For example, a schematic-capture program can transmit design changes to a bill-of-materials program to a program that calculates power dissipation to still another program that calculates the project's component cost. The electronic-product-development tools in the latest software release from Mentor Graphics Corp (Beaverton, OR) do exactly that. However these tools, collectively called Software Release 8.0, currently run under Unix and only on workstation-class computers. Without multitasking, you would need to run each program in turn to compute the various effects caused by the design changes. Because a collection of integrated application programs like Software Release 8.0 promises to provide a large jump in individual productivity, the PC market is girding for the jump to multitasking operating systems over the next two years.

Dueling operating systems

Unix and OS/2 are fighting for the lead as the PC multitasking operating system of choice. Most engineers have at least heard of Unix. Workstation and software vendors like Mentor made Unix the primary operating system for their technical applications. GUIs such as the Open Software Foundation's Motif metamorphose Unix from a hacker's delight into an OS that can be used by people with a wide range of computer skills. Intel claims. based on industry surveys, that almost 60% of Unix-based cokmputer systems shipped annually now incorporate 80286 and 80386 [microprocessor]s, and that many of those systems are Pcs. Minicomputers,

RISC-based products, and Motorola 68000-based computers constitute the remainder of the market. PCs are clearly a big part of the Unix world, although Unix currently represents only a small part of the PC's market.

For high-end PC's, OS/2 will be Unix's biggest competitor. OS/2 looks and operates a lot like DOS but provides additional capabilities such as multitasking. With the addition of the Presentation Manager GUI, OS/2 becomes a graphical OS that resembles the DOS/Windows combo. But OS/2 is strictly an OEM product for Microsoft; the company doesn't sell the operating system to end users, at least not yet. Currently, computer OEMs must use Microsoft's OS/2 development kit to buil a custom-tailored version of OS/2 for their specific hardware configurations. Although earlier versions of OS/2 (through version 1.2) run on both 80286-and 80386-based computers, Microsoft's OS/2 version 2.0, currently available only in development-kit form, runs only on computers than employment [microprocessor]s from the 80386 family.

In addition to multitasking, Unix and OS/2 provide another sorely needed feature: the ability to use more than 640K bytes of RAM for a program. This long-standing DOS-imposed limitation seemed inconsequential when DOS was first introduced. Compared to CP/M's 64k-byte limit, DOS's 640k-byte ceiling seemed vast. But software developers have made grand extensions to their products since 1981. They have attempted (and often succeeded) to take on tasks never thought possible on a PC. As a result, word processors have evolved into desktop-publishing packages, schematic-capture programs grapple with pc boards of ever increasing complexity, and some PLD compilers now attempt to fit designs into multiple devices. All of these added capabilities demand more memory, so the 640k-byte barrier, once a liberating factor, has now become a constant stumbling block.

Both Unix and OS/2 circumvent DOS's 640k-byte limitation. By their inherent designs, they make use of the memory above the first megabyte as conventional (not extended or expanded) memory. Because Unix and OS/2 version 2.0 don't run on 8088- and 80286-based machines, they don't limit programs to the first megabyte of memory. In addition, both operating systems support virtual-memory operation, which permits applications programs to run as though there were more RAM in the system than is actually present. Virtual-operating systems load only the parts of a program that are required for immediate operations and leave unneeded sections of code on the system's disk until they are needed. Disk capacity becomes the new limiting factor for program size, and hard-disk drives are getting very big indeed.

Unix for the PC

EDN's All-Star PC runs two flavors of Unix--Xenix and Unix/386. Both are offered by the Santa Cruz Operation (SCO). Xenix is a compact version of Unix designed to operate on 80286- and 80386-based PCs. The stock version supports only two types of hard-disk controllers--the de facto standard Western Digital WD1003 controller for ST-506 hard-disk drives and Adaptec's AHA1540 SCSI-host-adapter family. The WD1003 doesn't work with SCSI-based disk drives, and, as discussed in Part 2, the Adaptec boards conflict with Quarterdeck's DOS-extension products. The solution, for Xenix, is to add a special Xenix driver for the IN-2000. Always Technology, the vendor that supplied the IN-2000 SCSI host adapter for the All-Star PC, finished a disk driver for SCO's Xenix just in time for this article.

SCO's Unix/386 is part of the company's Open Desktop package, which combines a version of Unix System V version 3.2, a PC version of Motif, and various standard networking and communications programs into one comprehensive product. Open Desktop is a very big operating system by PC standards. To run it, a PC needs at least 8M bytes of RAM and should have at least 100M bytes of hard-disk space available. Unlike Xenix, SCO's Open Desktop package runs strictly on PCs that employ [microprocessors] from the 80386 family.

The Open Desktop can share the hard disk with other operating systems, such as DOS and OS/2, through DOS's disk-partitioning scheme. However, you must use the respective operating systems' FDISK programs to activate a partition. The active partition determines which operating system will take over the next time the computer boots. SCO Xenix has a "DOS" command that is supposed to switch a system over to DOS, but this command didn't work on the All-Star PC. Steven Katz, the vice president of software engineering at Always Technology, surmised that Unix's DOS command failed because it didn't recognize the large DOS 4.01 disk partition as a valid DOS partition. DOS 4.01's disk partition differs from the partitions created by older DOS vintages.

A tenuous, 3-way OS marriage

Trying to get DOS, OS/2, and Unix/386 to coexist on a hard disk on the All-Star PC presented several problems. The first was getting the right disk drivers for all three operating systems. OS vendors write their disk drivers for the PC's lowest common denominator: the WD1003. Any deviation requires a hard-disk driver to link the operating system to the "foreign" controller. The All-Star PC's IN-2000 incorporates a BIOS ROM that provides its own DOS driver, so no extra software is needed. Always also supplied a loadable driver for Xenix because Xenix does not use the BIOS ROM's disk-access routines. Because OS/2 and Unix also won't make use of the IN-2000's BIOS ROM, they too require disk drivers. As the deadline for this

article approached, it appeared that Always Technology would not be able to supply OS/2 and Unix drivers for its board in time for this project.

Fortunately, Perceptive Solutions Inc (PSI) had just completed its first production run of Hyperstore 1600 disk controllers. The Hyperstore 1600 is a caching disk controller with interchangeable media adapters. It can operate ST-506-, ESDI-, and SCSI-based hard disks. The controller board accepts 4M bytes of dynamic RAM (DRAM) for its disk cache and employs a Zilog Z280 [microprocessor] for SCSI and cache control. An auxiliary memory board can boost the controller's cache RAM to 20M bytes. The Hyperstore 1600 operates in either a native SSP (standard-storage-protocol) mode or a WD1003 emulation mode. PSI says that its SSP mode is slightly faster than the WD1003 mode. But, any PC operating system can use the emulation mode without additional disk drivers. The major drawback to emulation is that the WD1003 register-set definition limits the hard-disk capacity to two 450M-byte drives. Fortunately, that limitation presented no problems for the All-Star PC's two 315M-byte drives.

Though installing the Hyperstore 1600 presented few problems, there was one tough nut that had to be cracked--a conflict between the Hyperstore 1600's integral floppy-disk controller and the Compaticard IV card from Microsolutions, which was already in the All-Star PC. You can't disable the Hyperstore 1600's floppy-disk controller (although you can move it to a secondary address, which also conflicts with the Compaticard IV). As a result, the final configuration gives control of the All-Star PC's two 5-1/4-in. floppy-disk drives to the Hyperstore 1600 and retains the Compaticard IV for the two 3-1/2-in. floppy-disk drives. In this configuration, the Hyperstore 1600 uses the primary floppy-disk-controller address, and the Compaticard IV occupies the secondary address.

Tape-drive control became a problem when the Hyperstore 1600 entered the mix. Because I would be switching between the IN-2000 and the Hyperstore 1600, the All-Star PC needed a new tape-backup scheme. PSI has not yet developed tape drivers for its SCSI controller, and the Novastor tape-utility program supplied with the IN-2000 only works with the IN-2000. The addition of a third SCSI host adapter, Future Domain's TMC-860, solved the problem. Future Domain introduced drivers for the Sytos tape-backup program late last year and supplied that software for the All-Star PC project.

Sytos from Sytron Corp (Westboro, MA) is a general-purpose tape-backup utility. The Future Domain drivers link Sytos and the TMC-860 to the All-Star PC's Exabyte EXB-8200 cartridge-tape subsystem. Packing three SCSI host adapters into one PC is hardly an efficient configuration. One SCSI controller should be able to control all of the All-Star PC's SCSI-based mass-storage devices, but the TMC-860 is a bit slow for the Wren Runner disk drives, the IN-2000 has a limited repertoire of OS drivers, and the Hyperstore 1600 lacks tape drivers. The vendors of these SCSI host adapters will undoubtedly solve most of these problems in the near future, but for now, the All-Star PC must rely on the services of several SCSI host adapters to accomplish all of the project's goals.

Using either the IN-2000 or the Hyperstore 1600, DOS 4.01 and Windows/386 loaded easily. However, while using the Hyperstore 1600 to load either SCO's Open Desktop or Microsoft's OS/2 version 2.0, the All-Star PC locked up shortly after beginning the installation. As it turned out, neither problem was related to the Hyperstore 1600. Unable to solve the Unix and OS/2 installation problems alone, I took the All-Star PC and Steven Katz to Colorado Springs to visit Ron Sartore at Cheetah's R&D labs.

Sartore solved the Unix/386 problem in less than a day. Early in the installation process, Unix/386 tries to gauge the size of a PC's memory.

When it tried to read the first byte of the 17th megabyte, the All-star PC's memory subsystem refused to complete the memory-bus handshake and the system froze. A new address-decoder PLD cured that problem, and we loaded SCO's Open Desktop without further trouble.

OS/2 version 2.0 also didn't like the All-Star PC's configuration and refused to load. However, in this case the PC's hardware wasn't freezing; the processor continued to execute code, but externally, the PC was dead and wouldn't load OS/2 from the floppy. The same problem appeared when we tried to load OS/2 version 2.0 on a Cheetah system using a genuine WD1003 disk controller, again exonerating the Hyperstore 1600. It was probably a bit of a reach to expect that this pre-release version of OS/2 would run on the All-Star PC's complicated configuration, but we made the attempt anyway. By then time had run out, so we took a fall-back position. OS/2 version 1.1 loaded without any problems.

High-end PCs based on Intel's 80386 [microprocessor] family give you many operating-system alternatives. For today's PC software, such as the engineering packages to be discussed in Part 5, DOS is the clear operating system of choice. However, the advent of multitasking, virtual-operating systems such as Unix/386 and OS/2 version 2.0 will allow software developers to create more powerful, interrelated sets of application programs for PCs in the years ahead.

Adding DMA to other functions boosts speed. (direct memory access) (includes related article on DMA basics)

Leibson, Steven H.
EDN VOL.: v34 ISSUE: n22 PAGINATION: p95(9)
PUBLICATION DATE: Oct 26, 1989
AVAILABILITY: FULL TEXT ONLINE LINE COUNT: 00214
SPECIAL FEATURES: illustration; photograph; chart
CAPTIONS: (Two types of DMA transfers.); The combination of a DMAC and several system peripheral devices.
SOURCE FILE: CD File 275
DESCRIPTORS: Controllers; Technology; Data Transfer Rate; Direct Memory Access; Integrated circuits; Performance Improvement

Adding DMA to other functions boosts speed

You can't beat DMA for high-speed data transfers in a [microprocessor]- or [mu]C-based system. DMA controllers (DMACs) waste no bus cycles to fetch instructions; these hard-wired machines already know how to do the job. DMACs are really zero-instruction-set computers and offer superior performance to even the much-publicized reduced-instruction-set computers (RISCs), at least for transferring data.

Because they provide the highest possible data-transfer rates, DMACs quickly became an essential component in the [microprocessor]'s support-chip retinue. Today, however, DMACs appear to have been almost totally eclipsed by the 32-bit processor architectures introduced by semiconductor manufacturers during the last two or three years. In fact, many IC vendors currently offer no DMACs or 16-bit controllers to support their highly touted, 32-bit RISC and complex-instruction-set computer power-houses.

But the DMAC isn't dead yet. It has simply moved into other types of ICs and fused with other IC functions. These hybrid controllers can deliver performance superior to a multichip solution in a form that's easier for

you to use in a system.

One company, Siemens Components Inc, continues to advance the idea of discrete DMAC ICs. Siemens developed its 4-channel 82257 DMAC a few years ago, and it's still available for $30 (1000). However, Siemens has upgraded the 82257 design twice, and the latest product of that evolution is the 82258A advanced DMA (ADMA) IC.

The 82258A supports only 8- and 16-bit flow-through transfers, (see box, "DMA basics: flow-through and fly-by transfers"). Rather than build a 32-bit device, the 82258A's designers chose to use additional transistors so they could include other capabilities. For instance, one of the 82258A's four DMA channels supports a multiplexed mode that services as many as 32 devices without [microprocessor] intervention. The 82258A also sports features such as data comparison and matching (which terminates a transfer or triggers a change in transfer parameters when a match occurs), verification (which compares two data blocks without moving data), and translation (which uses a translation table to perform data conversion on the fly during transfers). A 10-MHz 82258A costs $80 (1000).

32-bit DMACs suit 32-bit [mu]Cs

The 82258A does not support 32-bit flow-through data transfers but can manage 32-bit fly-by transfers because its address incrementer can count by four. However, many processor-based systems that employ the latest crop of 32-bit [microprocessor]s require full 32-bit performance from a DMAC. Recognizing this need, Intel Corp currently offers two ICs that incorporate 8-channel, 32-bit DMACs: the 82370 and the 82380. These two devices are companions to the company's 80376 and 80386 [microprocessor]s. Both chips are 32-bit devices with equivalent features. The address and data buses of the 82370 match those of the 80376 [microprocessor] and the 82380's buses mate to the 80386 [microprocessor]. The 82370 (16 MHz) costs $45 (1000) and the 82380 (16 MHz) costs $70 (1000).

The 82370 and the 82380 are not simply DMACs. Their designers also gave them several system-level functions, such as a 20-level interrupt controller, reset logic, a programmable wait-state generator, four interval timers, and a dynamic RAM (DRAM) refresh controller (Fig 1). The combination of these components results in a device that offers more performance than you might extract from the individual ICs and saves you precious board space.

Integration breeds performance

Both the [microprocessor] and the DMAC share the 82370's and the 82380's wait-state generator. The generator lets you create six wait-state settings for your system: three for the memory space and three for I/O space. During an access cycle, the device being addressed drives two input pins that select one of the preset wait-state counts. The generator then postpones the cycle's completion by the requested number of counts. The combination of the wait-state generator and the DMAC on one chip eliminates the need to add any additional wait-state circuitry to your system design.

Intel recently introduced a similar device—the 82357 integrated system peripheral—as part of its Extended Industry Standard Architecture (EISA) chip set. (EISA is a specification for an enhanced version of IBM's PC/AT bus. It accommodates 32-bit expansion cards and multiprocessing.) Intel's 82357 incorporates a timer/counter, an interrupt controller, a bus arbiter, and a DRAM refresh controller, in addition to a DMAC, which supports EISA's three special burst-DMA modes. Intel only sells the 82357

with a companion 82358 EISA bus controller. The 2-chip set (25 MHz) costs $99 (1000).

Zilog offers a similar device for its 16-bit [microprocessor]s—the Z16C20 General Logic Unit. It can control peripheral devices and several types of system memory in addition to performing DMA transfers. The Z16C20 incorporates a 1-channel DMAC, DRAM and static-RAM controllers, a wait-state generator, an EPROM controller that speeds up EPROM accesses by as much as 25%, and two 16-bit timers. One of the timers can also serve as a watchdog timer. The Z16C20's DMAC operates only in the fly-by mode. The device costs $13.33 (100).

Merging [microprocessor]s and DMACs

Intel's 82370, 82380, and 82357 and Zilog's Z16C20 are part of a definite trend in DMAC IC design. Many chip designers are using the bounty of extra transistors, made available by the increasing circuit densities of advanced IC fabrication processes, for additional system functions rather than devoting all their IC's transistors to the DMA controller. To put it another way, many types of ICs now incorporate DMACs. In fact, as Table 1 illustrates, an increasing number of [microprocessor]s and [mu]Cs include DMACs as part of the processors' basic feature set.

Intel's recently introduced 80960CA [microprocessor] is an excellent example of a processor with an integrated DMAC. The 80960CA incorporates a 4-channel DMAC in addition to an interrupt controller, a bus controller, and an 80960 processor core. The [microprocessor]'s DMAC is so tightly woven into the fabric of the chip that it shares many of the IC's hardware resources with the processor core. These resources include the integer unit (for incrementing and decrementing registers) and the microcode ROM.

Through a variety of clever design features, the combined abilities of the 80960CA's processor core, DMAC, bus controller, and interrupt controller let the 80960CA achieve better system performance than if these functions were implemented with individual ICs. The interaction between the 80960CA's DMAC and bus controller demonstrates the real power of the combined on-chip functions.

As Fig 2 illustrates, two 128-bit data buses link the [microprocessor]'s DMAC, bus controller, data registers, and 1k-byte internal data RAM. Because the DMAC and the bus controller use these very wide buses, they can minimize the number of data transfers made inside the IC by packing the data into 128-bit words.

Move 16 bytes per bus cycle

Using only one internal-bus cycle, the 80960's DMAC can perform a 16-byte transfer between the [microprocessor]'s registers or on-chip RAM and the bus controller. In addition, because there are two of these wide data buses on the 80960CA, the [microprocessor]'s scheduling circuitry never has to turn one of these wide buses around between transfers. Packaging and pin limitations make 128-bit buses impractical as external [microprocessor] buses, and two such external buses are simply out of the question. But if critical functions are placed on the chip, as they are in the 80960CA, you can employ such a performance-boosting architecture for a 1-chip [microprosessor] design.

The 80960CA's bus controller uses a memory-region configuration table that lets you partition the chip's addresses space into 16 regions. For each region, you can specify

* the bus width (8-, 16-, or 32-bit)

* the byte ordering (big-or-little-endian)

* whether or not the chip's burst-mode feature should be enabled

* the number of wait states required

* whether or not to use address pipelining. The configuration table specifies how the external bus should operate for transactions initiated by both the processor core and the DMAC.

I/O requests from the 80960CA's processor core and from the DAMAC appear nearly identical to those from the bus controller. The [microprocessor]'s scheduling circuitry treats the DMAC as a true coprocessor and intermixes its requests with requests from the CPU in the chip's instruction pipeline. In addition, the processor's instruction set includes two DMa instructions ("set up DMA" and "update DMA") that facilitate construction of fast interrupt service routines. DMA instructions represent just one more example of the enhanced performance that results from building a DMAC into a [microprocessor].

A DMAC can transfer data faster than a CPU operating at the same clock rate because the DMAC is essentially a zero-instruction-set computer. While transferring data, a DMAC uses no bus bandwidth for fetching instructions. In addition, the transfer of bus control between a CPU and a DMAC requires less time than an interrupt context switch does. This situation gives DMA-based I/O transfers even more of a performance edge over interrupt-driven transfers. For the 80960CA, intermixing CPU and DMAC requests in the chip's instruction pipeline completely eliminates context-switching overhead.

However, the performance advantages of integrated, on-chip DMA don't just benefit new 32-bit [microprocessor]s. You can find integrated DMACs in newer versions of well-established 8- and 16-bit [microprocessor]s as well. For example, Zilog's Z80 processor architecture serves as the core processor for its Z280 [microprocessor] and for Hitachi's HD64180 (also available from Zilog as the Z180). Motorola's 68000 [microprocessor], a processor with a 16-bit data bus but 32-bit registers, is the core processor for the Signetics 68070 integrated [microprocessor]. The 64180, Z280, and 68070 [microprocessor]s all incorporate DMACs, as shown in Table 1.

[mu]Cs benefit from DMACs, too

Several 8-bit muCs also include integrated DMA facilities. Hitachi's HD647180X, a [mu]C verison of the company's HD64180 [microprocessor], replicates the [microprocessor] version's DMAC capabilities. Other 8-bit [mu]Cs such as Toshiba's TMP90C84X series and NEC's 78k2XX and 78K3XX [mu]Cs provide a form of DMA controlled by the CPU's hardware.

Block diagrams of the NEC and Toshiba [mu]Cs do not show functional blocks marked "DMA controller," but they have DMA capabilities nevertheless. Toshiba calls its DMA operations "micro DMA processing." NEC calls its DMA operations "macro service functions."

In both NEC's and Toshiba's [mu]C families, the DMA controller circuitry is merged into the processor core. Blocks of [muC registers hold source and destination addresses and a terminal count. After a program loads these registers with appropriate values and activates the DMA transfer, the special DMA circuitry in the processor core executes the transfer operation without further program intervention.

NEC has enhanced the macro service functions of its 78K2XX [mu]Cs, and its DMA operations can often replace quite a bit of software. the [mu]C's type C macro service function accepts interruprts from one of the chip's timers. For each interrupt, the function transfers one byte of data to an I/O port and transfers a new value to the interrupting timer. This enables you to create two tables, one containing data to be output and one containing timer values; the type C macro service function transfers the data using the intervals specified in the timer table.

NEC specifically developed this particular DMA mode to allow the [mu]C to control the ramp-up and rampd-down operations of stepper motors. However, this facility can come in handy when you need to transfer data at irregular intervals. Most processors can only do this through interrupt-service routines, which have relatively long and unpredictable latency times. The 78K2XX-[mu]C family's macro service function treats the job as a background task. In addition, the macro service function is invisible to the [mu]C's program when operating and has a short, fixed, latency time.

Intel's 8XC152 [mu]C family incorporates a 2-channel DMAC to support its high-speed, on-chip, multiprotocol communications controller. This controller can handle several network protocols including synchronous data link control and carrier-sense, multiple access with camer detection, and it is targeted at high-speed LAN applications.

LAN communications generally involve high-speed transfer rates, and a DMAC helps ensure that the [mu]CC can move data from the LAN to memory quickly and efficiently. For example, Ethernet LAN-controller ICs have incorporated DMA capabilities for years. No other data-transfer technique can support the data rates required by high-speed LANs.

Peripheral chips get DMACs

AT&T recently introduced its T7115, a 32-channel HDLC controller designed for digital T1 and primary-rate Integrated Services Digital Network communications systems. It incorporates a DMAC to handle the demanding data rates required by these telecommunications applications. The T7115 costs $55 (1000).

DMACs make sense for other serial-I/O protocols. too. Most data-communications protocols employ inherently asynchronous processes that need DMA's low-latency characteristics. For example, Zilog offers two serial devices that incorporate DMACs: the $75 (100) Z16C30 universal serial controller (USC) and the $ 25 (1000) Z16C35 integrated serial communications controller (ISCC). The Z16C30 supports serial data rates up to 10M bps. making it a prime candidate for DMA service. It also incorporates a 32-byte FIFO buffer so that its integrated DMAC can move data from the USC to memory in 32-byte bursts. Burst operation eliminates much of the bus-exchange overhead that occurs between a CPU and DMAC. That overhead can bog down individual DMA transfers.

The Z16C35's two full-duplex serial channels operate at rates up to 4M bps, and its 4-channel DMAC can move data at 3.1M bytes/sec. With two serial channels, each running at 4M bps, the Z16C35 has data-transfer requirements almost as stringent as the Z16C30's.

One of the key benefits of the integrated-DMAC approach is its support for fly-by DMA transfers, which are the most efficient type. Because DMACs are integrated into serial devices, a fly-by DMA transfer involves only two system components (memory and the serial chip) rather than three (memory,

the serial chip. and the DMAC)). Elimination of the external DMAC also reduces or eliminates the need for additional glue logic.

Integrated DMA controllers require less board space, improve performance, and lower costs. As higher integration levels allow IC designers to redefine existing ICs and invent new devices, you will see an increasing use of DMA to move data in [microprocessor]- and [mu]C-based systems. For data movement, not even RISC [microprocessor]s can outperform DMACs.

Nixdorf opts for 16-bit Z280 version of Z80 at low, Unix at high end for retail line. (Nixdorf Computer AG)

Computergram International ISSUE: n1237 PAGINATION: CGI08080007
PUBLICATION DATE: August 8, 1989 LINE COUNT: 00052
SOURCE FILE: CD File 275
PRODUCT NAME(S): Nixdorf POS 2000 (Computer system) - product specifications; Nixdorf POS 2000 10 (Computer system) - marketing; Nixdorf POS 2000 20 (Computer system) - marketing; Nixdorf POS 2000 30 (Computer system) - marketing; Nixdorf POS 2000 40 (Computer system) - marketing
COMPANY NAME(S): Nixdorf Computer AG - marketing
DESCRIPTORS: Marketing Strategy; POS Terminals; Retailing; UNIX
SIC CODE: 3571

NIXDORF OPTS FOR 16-BIT Z280 VERSION OF Z80 AT LOW,

UNIX AT HIGH END FOR RETAIL LINE

Nixdorf Computer Ltd's Retail Division, which launched its new range of point of sale terminals last week (CI No 1,230), looks to the new line to buy it a stronger market penetration of the department store, garage forecourt and hotel and leisure sectors, while retaining its hold on the food multiples, non-food multiples and cash receipting sectors. The technological key to further expansion lies with its new POS 2000 family. Nixdorf's reputation in retail has hitherto been built upon its modular range of terminals, the 8812 family which was launched in 1977 and which can be configured to suit different retail environments. To date, the division has contracts with such illustrious UK high street names as Bally Shoes, Currys, Dixons, Toys 'R' Us, Peacock's Stores and SavaCentre as well as Laura Ashley. In fact this last name was taken by Nixdorf as of great significance in terms of hype for the single European Market, since it has operated on a Euro-pean scale for many years. In an attempt to steal a march on companies such as ICL Data-checker (CI No 1,227) and British Coal's Compower (CI No 1,223) Nixdorf has moved towards Unix with its new POS 200 range.

Uniform designs

Each of the four products in the range is based on a uniform design which comprises a printer, keyboard and customer display, with the central processor housed in an electronic box under the printer. The low-end model, the 2000/10, which is available now, uses a Zilog Z280 16-bit processor with a main memory of 1Mb and is compatible with Nixdorf's 8812 range, offering an upgrade gateway to the POS 2000 family via the exchange of a central processor card. Such terminals are normally backed by a personal computer but the 2000/20 and 2000/30 terminals integrate front and back office applications through their AT-compatible architectures which drive POS 2000 peripherals. The /20 version is an 80286 single user MS-DOS systemc

enabling the user to hot key between the POS application and any back office personal computer application. The /30 80386 multiuser version runs Unix and supports further back office terminals and a second POS terminal. Both these products will ship by the end of the year. Finally, Urange's big boy is the 2000/40 which is designed to be both a POS terminal and controller. The /40 is built around Intel's 80386X chip, has up to 4Mb of memory and interfaces with RS232, RS485, as well as OCIA for scanners, SCSI for mass storage peripherals, an>andard Nixdorf interfaces. It can be equiped with a 3.5" floppy and 3.5" hard disk, s}porting operating systems such as Flexos and Unix. This model will be ready to ship some time in 1990. So how do these terminals fit in with Nixdorf's marketing drive? Well, the Retail Division is particularly targeting the /10 and /40 models at department stores which it believes will find the full screen capability this configuration affords useful for train-ing and for sales pushes; the /10 and /30 terminals offering open systems compatibility and connectivity to a variety of peripherals will, it is hoped wow, garage forecourt managers; while the /10 and /20 versions offered as stand-alones with the option of a special catering keyboard are intended to hook the hotel, and leisure sectors.