Concurrent DOS-286 Challenges Unix

BYTE West Coast
BYTE Magazine

May 1985

In January, Digital Research (of CP/M fame) unwrapped the latest product in its Concurrent DOS line of multitasking operating systems. Concurrent DOS-286 is a multitasking, real-time, network-compatible operating system for single- or multiuser, 80286-microprocessor-based systems and is written in the C language. With that mouthful of attributes, Concurrent DOS is plainly in line to compete with the new microcomputer versions of UNIX. While ROM implementations of UNIX (see "The HP Integral Personal Computer" by Phillip Robinson, February BYTE, page 98) and new programs that allow UNIX to run PC-DOS programs (see BYTE West Coast January, page 415) are moving that powerful minicomputer operating system down into the microcomputer world, Concurrent DOS-286 is moving microcomputers up into the world of multitasking and passwords.

Concurrent DOS-286 is not just a revision of the earlier Concurrent operating systems; it has an entirely new architecture. Specifically, it is built around advanced hardware features of the Intel 80286 (also known as the iAPX286) microprocessor. Those features—descriptor tables, call gates, and protection levels (see "The 80286 Microprocessor" by Paul Wells, November 1984 BYTE, page 231)—allow speedy context switching as well as protection of files and programs.

Compatibility

Concurrent DOS-286 provides a migration path for PC-DOS or CP/M-86 users who need more power; it can run both CP/M and PC-DOS applications. Gary Gysin of Digital Research explained to us that "To start ..., our strategy is one of following what IBM does. So, if they come out with PC-DOS 2, PC-DOS 3, PC-DOS 3.1, whatever it is, our operating-system strategy is to embody whatever they're doing and at minimum to be compatible with [it]." An example of that strategy is Digital Research's new hierarchical file system: CP/M files have been left behind. The file system, all the system calls, and all of the utilities are a superset of those in PC-DOS 3.0. Because of this change, if you want to run CP/M applications, you have to transfer them to PC-DOS disks (the native media of Concurrent DOS-286). Concurrent DOS is compatible with PC-DOS 3.0; with the PC-DOS front end it will run PC-DOS 1.1, 2.0, 2.1, and 3.0 applications.

Although more will be available in the future, the two most important front ends, one for CP/M-86 users and one for PC-DOS users, will be available immediately. This lets you run applications from those operating systems on Concurrent DOS-286. However, as Gysin further explained, "Theoretically, if we wanted to add some other front end to the operating system, be it UNIX, XENIX—whatever might become an industry-standard operating-system front end—then that possibility is there," because of the modular design of Concurrent DOS-286.

Cross-compilers and cross-assemblers for the VAX are not available, but the source code we're told can easily be moved to a VAX. All Digital Research languages will be available for Concurrent DOS-286. Also, if you know CP/M, you may be interested to know that ED is gone. Digital Research had a funeral for the text editor that is now replaced by DR EDIX—a function-key-driven editor.

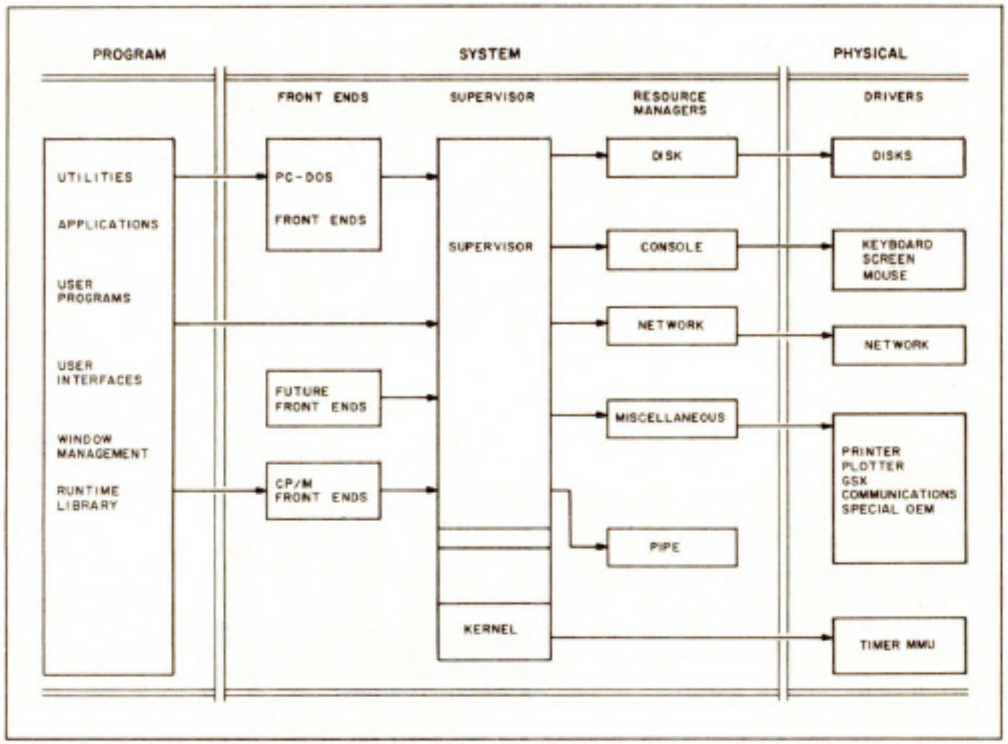

Figure 1: A diagram of the three functional units, program, system, and physical, and two primal interfaces of Concurrent DOS-286.

Architecture

As shown in figure 1, Concurrent DOS-286 has three functional units and two primal interfaces. The functional units are termed *program*, *system*, and *physical* hardware. The system receives requests from the programs and translates them into instructions for the physical hardware. Optional front ends let the operating system run PC-DOS or CP/M applications. The *supervisor* portion of the system controls the flow of services to the underlying *resource managers*, which in turn control the actual physical peripherals.

Concurrent DOS-286 can run multiple applications using virtual consoles on each physical console, handle asynchronous events and software-interrupt handling, and deal with interprocess communications and synchronization. Storage depends on a hierarchical, shared-disk-file system with record and file locking. File ownership is controlled through user and group IDs requiring log-on with passwords.

The console is a standard VT52 interface with character and bit-mapped screen interfaces. The keyboard interface uses both standard 16- and 8-bit codes including function keys, numeric keypads, and multikeyed characters. Also, Concurrent DOS-286 supports windows, mice, graphics (raster and vector), and networks as well as file sharing, file locking, record locking within a file, locking a specified number of bytes within a file, and even overlapping locks.

Concurrent DOS-286 with floppy- and hard-disk drivers, character-mapped and bit-mapped consoles, and a printer driver requires about 160K bytes of RAM. The minimum suggested system configuration has 512K bytes of RAM. The operating system is ROMable.

The kernel is based on an event-driven dispatcher. Time slicing by a timer event occurs once per tick (16 to 20 milliseconds), and scheduling of equal-priority processes is based on a round-robin scheme. Process communication and synchronization are done through named *pipes* that pass messages from one process to another or synchronize processes by acting as semaphores. The 80286 chips have a protection mechanism based on address manipulations. On-chip calculations handle the translation of virtual and physical addresses. Certain segments of memory can be marked as exception areas for particular users or programs. Whenever a program tries to address an exception area, the central processor is interrupted, a trap is generated, and processing is forced to another routine.

Stopping the Operating System Bypass

Trapping is a vital element in Concurrent DOS-286's ability to handle poorly behaved programs. Programs that employ operating-system routines to handle the computer hardware are termed *well behaved*. Programs that bypass operating-system routines and make direct calls to hardware—for instance, to the screen—are *poorly behaved*. While bypassing the operating system sometimes improves a program's execution speed, it hurts program portability. A program written around the operating system works on any computer running that operating system. A program tailored to a specific computer by means of direct hardware calls only runs on that computer and its nearly identical copies—poorly behaved programs are the waterloo of clone-makers. Concurrent DOS-286, however, can trap the calls made directly to the *physical* hardware and route them to *virtual* hardware.

For example, Lotus 1-2-3 is a poorly behaved program that assumes it owns the entire computer display screen. Gysin pointed out that if you tried to run both Lotus 1-2-3 and WordStar concurrently on a system that couldn't trap hardware calls, "you'd have trash all over the screen." Lotus 1-2-3 wouldn't let WordStar have a window. In Concurrent DOS-286, however, whenever a program tries to control the screen, the operating system traps the call and sends it to a window instead. The program thinks it is running on a full IBM Personal Computer, but instead it is running on a virtual console manager—a part of the Concurrent DOS-286 system. This is another advantage for computer companies that want to sell IBM-compatible personal computers. As Gysin tells competitors, "Go ahead and build whatever kind of screen you want, go ahead and build whatever kind of super-duper machine you can, we can still guarantee you we'll run PC-DOS applications, given that you've got a 286 chip."

The 80286 provides you with both compatibility and practical concurrency. All the fancy trapping could be done entirely by software, but it would take much longer than the hardware trapping. Trapping and re-routing signals could slow programs down to the point that they don't work properly. Because all of the trapping chores are handled within a single chip (the 80286), Concurrent DOS-286 doesn't have that problem. According to Gysin, Digital Research's quest to have applications run as fast under Concurrent DOS-286 as they do under PC-DOS has been successful. He claims, "We've gotten there, there's no difference."

Context switching—changing the active program by suspending one program, saving the registers, restoring the registers, and then restarting another program—is now done in software. Digital Research estimates that each software context switch takes between 200 and 400 microseconds. Digital Research plans to take advantage of the 80286's on-chip context-switching hardware that can handle the switch in only 20 microseconds.

Dynamic Drivers

Another big change in Concurrent DOS-286 is that the device drivers are dynamic. In previous incarnations of CP/M and Concurrent DOS, all the drivers were in a single section of code called BIOS (basic input/output system) or XIOS (extended input/output system). You had to load the drivers into the operating system before starting up. Dynamic loading means that you can add or modify device drivers while the operating system is running.

User Interface

Digital Research is also presenting a new user interface for Concurrent DOS-286 and Concurrent PC-DOS. It is essentially a menu system and a file manager that allows you to control all system utilities and all running applications with the function keys. "What we're trying to do is take away the A> prompt," says Gysin. But if you prefer the standard CP/M-style prompts, just press the Escape key to get back onto familiar ground

TopView, GEM, and GSX

What Digital Research calls "desktop primitives," which support such application environments as Microsoft Windows, Visi On, TopView, or "unannounced products from Digital Research" (Gysin's words), are built into Concurrent DOS-286. Gysin adds, "Whatever becomes standard, that is something that we'll map to our operating system. If some other bit-map graphics type of interface is the standard, then we'll also support that one." Digital Research's new GEM (Graphics Environment Manager) is a Macintosh-like interface that fits that bill.

Digital Research's GXS (Graphics System Extension) software is also supplied with Concurrent DOS-286. This graphics operating system lets you write to a standard set of graphics calls so you don't need to know at programming time which specific peripherals you will have.

International Flavors

Concurrent DOS-286 has three features that should delight software engineers outside the United States: All the system messages are kept in a separate pool, the user tables have a *country* code, and the keyboard interface supports 16-bit character I/O (which is required for kanji, for example). Those design tidbits are understandable when you realize that 40 percent of Digital Research's business is done overseas.

Summary

What's new with Concurrent DOS-286? The architecture has changed. Previous versions of Concurrent DOS provided support for PC-DOS 1.1 and CP/M-86 applications. Concurrent DOS-286 provides support for PC-DOS 1.1, 2.0, and 3.0 applications as well as memory protection, log-on and log-off, dynamic local device drivers, a hierarchical file system, completely modular design, pipes, I/O redirection, and an address-exception mechanism. In essence, Concurrent DOS-286 is intended to release you from needing IBM hardware or a clone to run IBM Personal Computer programs such as Lotus 1-2-3. Now, any system that can run Concurrent DOS-286 can run IBM Personal Computer programs.

How well does Concurrent DOS-286 work? We don't know. The descriptions in this article are based on discussion with Digital Research and Intel technical experts, not on a review of the operating system itself. When we do get into testing Concurrent DOS-286, our first consideration will probably be its use of memory. Although Concurrent DOS-286 is supposed to run as many programs concurrently as the hardware can handle, BYTE staff members have had problems with the voracious memory appetite of previous versions of Concurrent DOS.