THE FUTURE OF THE HP1000 ON A WINDOWS 2000 PLATFORM

presented by
Alvin Q Morgan and Alan Tibbetts

STROBE DATA, INC
8405 165th NE
Redmond, WA 98052

voice: (425) 861-4940
fax: (425) 861-4295
strobedata.com

THE FUTURE OF THE HP1000
ON A WINDOWS 2000 PLATFORM

the strengths of the HP1000 minicomputer design

In its day, it may well have been the ideal real-time and data acquisition computer, its architecture honed to high capacity throughput and the reliability demands of the space age, particularly space telemetry and testing of rocket engines.

The computer systems of the 1970s were modest by today's standards. Basic instruction execution in the microsecond range and limited memory capacity were still powerful enough to crunch more data than the systems could gobble up or spit out. Applications involving massive amounts of data forced designers into optimizing I/O at all costs. This resulted in what we call "I/O-centric designs". When in doubt, get the bottleneck out! And the bottleneck was almost always I/O.

In contrast, today's desktop computers are "CPU-centric". Effort is made to shrink clock cycles and speed up execution of basic machine instructions, counting on increasing the speed of the peripherals themselves to take care of the I/O bottleneck.

As Hewlett Packard engineers evolved the HP1000 from the HP 2116 through the M/E/F series models through the A-400 through the A-900 and A990 -- over a 28-year time span -- they took the basic technology of 1970s generation of minicomputers and did what they could to increase internal data processing within a conservative design framework. They expanded available memory from 64 kbytes to 32 Mbytes.
They tacked on code and data separation to facilitate high-speed execution of larger programs. Starting with the RISC-like instruction sets common to the minicomputers, they expanded the instruction set into CISC-like realm turning scientific functions from macros to microcoded instructions. They cached memory to shorten execution times of memory addressing instructions.

However, it was in I/O optimization where they excelled, and for that reason we claim that the HP1000 is essentially an I/O-centric design. The engineers architected their computer with parallel I/O processors, assumed that all I/O was direct memory access (DMA) and then buffered and cached DMA processes everywhere they could.

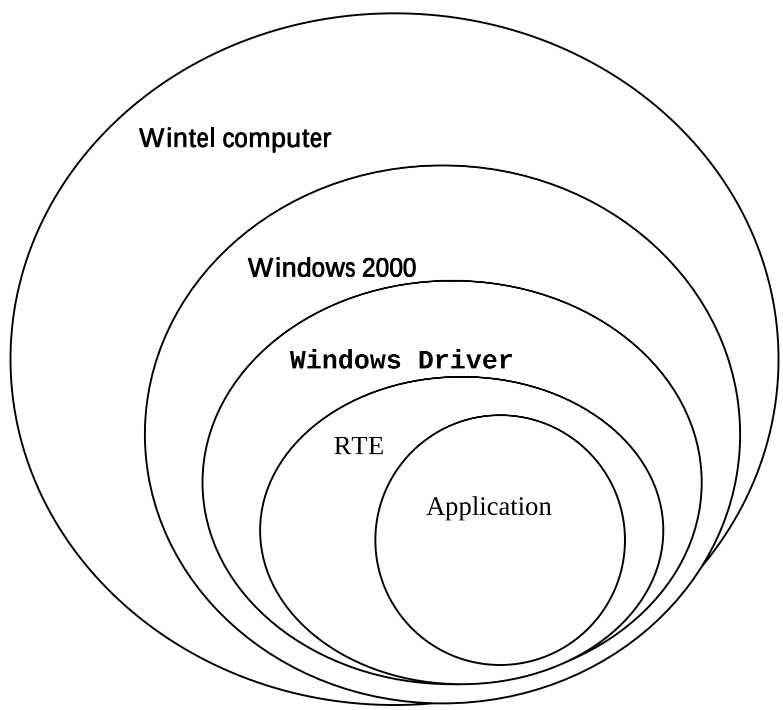
The I/O-centric design became the ultimate lab computer. Closely tieing its operating system environment to the actual physical hardware made the computer easy to install and easy to use. Except for the earliest models, a single general interface, the HP-IB bus, became the lingua franca of laboratory devices and, indeed, of most of the devices on the computer. This simplifying step further added to the reliability and maintainability of the HP1000. When high speed data communication requirements threatened to overwhelm the data acquisition/processing capacities of other minicomputers, the HP engineers took the cautious approach of buffering DMA transfers, not by a measly 2 or 4 or 8 bytes, but in 1024 byte buffers. Each precious bit from a dying satellite or a jet engine test rig was going to be captured and saved not matter what the transmission circumstances.

All that tinkering produced a machine uniquely honed to a high data throughput scientific computing niche. Even today it is difficult to beat the HP1000 at the things it does best, which is why the A990, introduced in 1992 and the last living descendent of the 1970's vintage M-Series processor, can outperform today's powerful CPU-centric desktop technology in a number of high throughput applications.

migration to a Wintel platform

Last Spring we gave a paper at InterWorks 2000 enumerating some of the daunting difficulties of recoding large systems as a way to move applications to modern hardware. One aspect of this migration that we did not cover was the problem of making even the fastest CPU-centric computers do the work of I/O-centric computers, particularly in cases where the CPU demands are considerable to start with. When the performance bottleneck is the I/O, faster, more efficient I/O management helps, but may not contribute enough to maintain throughput.

In that paper we described an "ideal" migration method for the family of minicomputers that includes the HP1000. We proposed "embedding" the applications

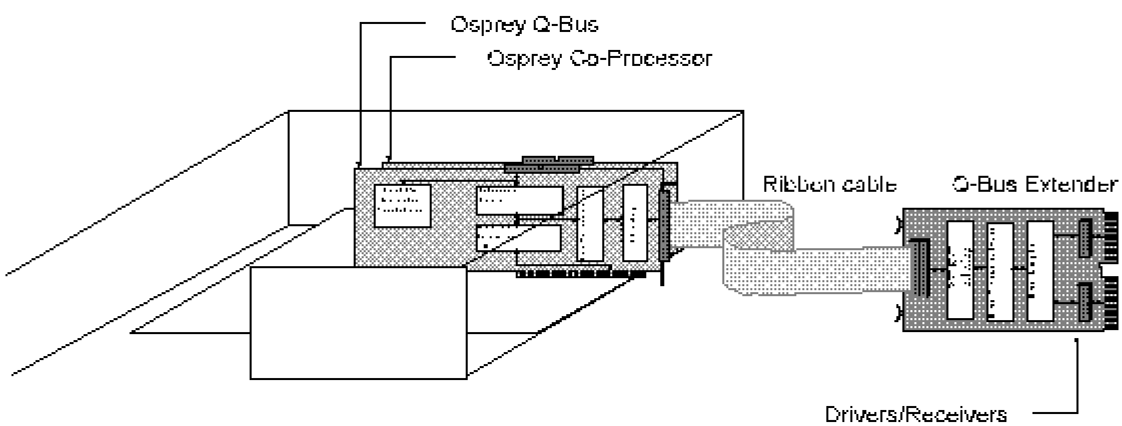


Typical application in an RTE environment controlled by a Windows driver in a Windows 2000 environment running on a Wintel computer.

software plus the operating system (in this case either RTE-A or RTE-6) in something which might be considered a Windows 2000 "driver". In the embedding process, the operating system and application would run in an "emulation" mode oblivious to the fact that they were executing on hardware associated with a Wintel platform.

This is an approach Strobe Data has made successful with DOS and Windows replacement products for the Data General Nova/Eclipse and Digital Equipment Corp's PDP-11 series. The DG replacement product is the "Hawk Co-Processor". The "Osprey Co-Processor" is the PDP-11 version. What is important to keep in mind is that these are not "software" emulations. They are hardware devices -- PC add-in boards with driver software running under Windows -- literally "co-processors".

Strobe achieved a high degree of replication by, in effect, re-architecting the Nova/Eclipse and various flavors of PDP-11 as brand new computers on PC-format ISA and PCI cards. The replication is so good that all the various bus characteristics were retained. The high degree of accuracy in emulating the bus characteristics enable the Hawk and Osprey card sets to faithfully generate Nova/Eclipse bus signals and the Unibus and Qbus signals of the PDP-11 series. This makes it possible to replace these minicomputers in real time and test environments where the computer is interfaced with special sensor and mechanical equipment.



Osprey Co-Processor, an ISA or PCI card set for standard PC-type computers communicates directly with Q-bus device controllers

This approach worked extremely well when the computers being replaced were the Data General Nova/Eclipse and Digital Equipment Corp's PDP-11. I/O on these machines was essentially managed by the central processors, and in this way they resemble your garden variety PC. However, the HP1000 is optimized for reliable high data throughput by means of hardware parallel I/O processing structures. And although the whole idea of embedding an application and its operating system wholescale in an entirely new computer with an entirely different operating system simplifies the migration project considerably, it is not obvious how it overcomes the effects of an I/O bottleneck. The Wintel machine is, after all, a notoriously CPU-centric design. One would think that the result would throttle the data throughput aspects of the system and result in what, to the user, appears to be a slower machine!

retaining I/O throughput characteristics

Not so slow after all, as it turns out! The Hawk, the Osprey -- and now, the HP1000 version -- the Kestrel -- benefit from the fact that they are full-blown computers in and of themselves, separate hardware from the Wintel computer backplane they are parasitically attached to. All computation, all actual computer instructions, are executed in CPU logic on the Hawk board, the Osprey board -- and now on the Kestrel board. The Wintel computer, running under Windows, serves as a glorified device controller for disks, serial lines, console, tapes, parallel ports, ethernet cards, etc. As a dumb-bunny device controller for the Osprey or Hawk, very little of its capacity is used -- maybe 1% in most cases.

For the Kestrel, however, the Wintel computer's I/O task is more complex. The Kestrel system is designed to retain the essential logical elements of the HP1000 environment, and that includes its highly parallel form of I/O processing. The Wintel computer now has the job of replicating the parallel I/O processing of the HP1000 using the not inconsiderable CPU, memory and the parallel processing capability of a commodity priced PC running Windows 2000.

In our opinion, a 500 MHz Wintel computer, running Windows 2000 is up to the task, especially since the data transfer to/from the Kestrel board to/from the Wintel computer is basically instantaneous with data transfer overhead measured in nano-seconds.

EOL announcement does not mean the HP1000 architecture is dead

Despite the end-of-life status Hewlett Packard has announced for the HP1000, the HP1000 far from dead! The Kestrel promises new-life for this doughty survivor of the minicomputer era. The major design elements of the HP1000, including the entirety of its instruction set, its I/O structure, even its backplane signals are given an on-going virtual existence. Of course, the implementation of those elements is radically different. The Kestrel takes advantage of advances in technology which have revolutionized computing in the last 15 years. The whole of the computer fits on one PCI board, plus a little emulating software on the host Wintel platform. Although at present not quite as fast as the A990, it is much faster than the A900 and it is expected that A990 speeds should shortly be achievable.

Today's Kestrel has been optimized to the A900 design. With minor modifications it will also emulate the A400. And with other planned small additions, it will emulate the M/E/F series as well. Of course, its basic speed will be much greater than the A400 and the M/E/F family of computers. This may present a problem to some applications.

real time and embedded systems present special challenges

Retaining the timing characteristics of a real-time application continues to be a challenge for any migration project. Embedded systems driving mechanical devices are constrained by the response pattern of those devices. Moving the software to a faster platform may be tempting, but it also may break the application. Rewriting the application, taking into account the performance characteristics of a new computer might seem to be the only way to protect the timing requirements of devices and human responses.

Most RTE programmers make use of interval timers to match signal processing with the frequency rates of the instruments their software drives and/or reads. Maintaining the integrity of interval timers, be they hardware or software, is a basic responsibility of the Kestrel, given that the Kestrel's design goal is to provide a computer replacement for which no recoding is necessary.

The HP1000 real time clock signal is, of course, faithfully provided by the Kestrel. However, programmers tend to make assumptions about what processing can or cannot be accomplished within a given time interval. Sloppy programming based on the perception that a certain section of code will not terminate before a clock signal interrupts it, can result in very strange results when a countdown is reached and no clock signal has occurred. Since both the Hawk and Osprey are much faster than the minicomputers they replace, Strobe trouble-shooters have run into the consequences of such misperceptions many times. They expect to come across similar problems with the Kestral.

In some cases, the broken application can be "fixed" by doing no more than changing a time constant or two in the application. But there are applications out there that haven't been touched for 10 years, for which no reliable sources can be found, and no programmer experienced with the application is available. Migrating a real time system in that environment may preclude a simple software fix. So the solution Strobe engineers turn to is a variable clock rate for the co-processor board. On the Osprey, for example, the user can set the clock rate from its normal 33 or 40 MHz down as low as 7 MHz. This is accomplished by changing a software configuration parameter on the host Wintel computer.

Indeed, the simplicity by which the internal timing of the system can be controlled is characteristic of this whole approach of replacing the now defunct HP1000 with its new-technology, Windows 2000 brother. In most instances, the "migration" activity can be accomplished in a few hours or less.

migration? a drop-in replacement with no sysgen

The Osprey graphic above included a Q-bus "adapter" board. This board is connected by a cable to the Osprey card or card set. It plugs into the Qbus card cage on a PDP-11 chassis. When "migrating" a PDP-11 application to the Osprey, the user copies the contents of the PDP-11 bootable hard disk onto a special file on the Osprey's host PC using a garden variety PDP-11 OS copy command. The real PDP-11 disk is read over the backplane extender board and its cable onto a container file on the PC. If needed, the user sets some configuration parameters by editing a file on the PC (the Configuration File), then invokes the Osprey host control software by clicking its Windows icon to boot the user's operating system (RSX, for example) and the normal OS console sequence will appear from which the application can be executed. The virtual control panel of a PDP-11, displaying on the PC monitor, acts just as it does on the original machine. For models with front panel switches, the user can set those switches as Configuration File parameters.

Moving operating system (RTE) and application onto the Kestrel operates in exactly the same way.

Most configuration issues are resolved through the co-processor's Configuration File assigning Wintel PC devices to the HP1000 peripherals. A default is provided. Deviations from the default are edited under the standard Windows word processors. The user has the power to assign any PC/Windows device to any HP1000 device, within reason. (Obviously assigning a hard disk to a PC monitor/keyboard would have limited utility, particularly if you intended to boot from it.)

the future of the HP1000 architecture

Since Hewlett Packard has issued its end-of-life schedule for the computer, the future of the HP1000 architecture is now the Kestrel. True, RTE applications will continue to be rewritten for totally different architectures. But as far as we know, no other attempts have been made to preserve the characteristics of the HP1000's I/O structure and backplane signals.

For applications still hosted on the HP1000, the future is bright. The Kestrel/PC approach allows those applications to communicate more directly with the modern world networks and the Internet. Making use of a utility which extracts data from the HP1000 container file on the Wintel machine and converts them to Windows-compatible format, files generated on the HP1000 environment can now be used directly by Windows utilities and transmitted to other processing entities. The HP1000 application can now use files generated in other environments (including files transmitted over the Internet) by using the same utility in reverse. In fact, this data transfer between HP1000 environment and Wintel environment can occur dynamically using other features of the Kestrel.

Obsolete peripherals hardware is now replaced by modern disk, tape, CD-ROM, communications devices, etc. The Kestrel environment even allows for remote bootstrap. By moving the application onto this peculiar "bridge" from the "old" minicomputer technology to today's high speed, high access, wide communications "new" technology, the HP1000, itself, moves into the new millennium with the advantages inherent in any new design, any new computer adapted for the milieu it was designed in. The Kestrel interpretation of the HP1000 IS a new, modern-age computer running the newest version of the world's most popular and accessible operating system. But it remembers its "roots". It retains the essentials of the HP1000 design.