EMBEDDED DEVICES ARE BEGINNING to look a lot like desktop PCs. Many embedded-system customers now demand graphical interfaces, instantaneous communications, multimedia,
and full integration with Internet-delivered data and services. With these features already built into computers, it is only natural that developers are considering variations of the Windows operating system for new embedded products. Windows is a tempting option for an embedded operating system, because it has a familiar API, and you can take advantage of many powerful desktop features. Microsoft’s most recent embedded products, Windows XP embedded and Windows CE.Net, bring new features and capabilities to developers and deserve renewed consideration for the next generation of complex devices.

Although Microsoft is relatively new to the embedded scene, Windows-based products have instant credibility with many developers because they are rich in functions and desktop-PC market share. High-end embedded systems—such as medical and test systems in which compatible PCs are part of the hardware suites—have been using the desktop version of the Windows operating system for more than a decade. Microsoft began pursuing smaller embedded systems in 1996 with its release of Windows CE 1.0, a from-scratch code base targeting handheld-device applications with a graphical interface that simulated the look and feel of Windows 95. Likewise, the componentized version of desktop Windows was born in 1999 with a tool set and about 250 components from Windows NT.

As developers pack more features into the smallest devices, demands on the application software and operating system escalate. In fact, there has been a major change in many embedded devices as designers move from the simplest 8-bit processors to faster, more powerful, 16- and 32-bit units. Lower hardware prices are one reason for the shift, but customer...
requirements for a higher level of software functions force most projects into more complex hardware. Users have grown accustomed to the never-ending list of features and capabilities available on the desktop and expect a similar experience when interacting with an embedded device. Faster processors have also given embedded-system developers another reason to consider a Windows operating system. Some applications that once required a real-time operating system to meet thread-scheduling demands may now perform satisfactorily under Windows because of faster program-execution rates.

Software complexity is just one of several reasons to consider a Windows variation for embedded-device applications. For example, newer embedded devices require networking and communications software to connect with local or Internet databases and services. Many commercial operating systems offer communications features, such as TCP/IP stacks and 802.11 drivers, as extra-cost options that may also include recurring royalties. Even if you are developing a self-contained or homergrown operating system, you will probably use third-party communications software to avoid the reinventing and testing efforts. Both Windows CE.NET and Windows XP embedded include an extensive library with hardware drivers from which you can build your exact communications suite.

EMBEDDED SECURITY

Network security is another recently added requirement for many wired and wireless embedded systems. Applications such as residential gateways may call for VPN (virtual-private-network) and firewall routines to protect Internet data. For example, IPSec (Internet Protocol Security) authenticates the host before and during communications, encrypts IP packets, and identifies modified or spoofed traffic. In addition to IPSec, embedded Windows products include the Kerberos security protocol and Web Server SSL (Secure Socket Layer) support. Microsoft also spends considerable time and resources fixing and testing security flaws for desktop and embedded products. Because security problems appear almost weekly, development projects need the support of a large operating-system vendor with programmers devoted to upgrades and bug fixes.

Although many have ridiculed the “blue screen of death” from previous versions and pointed to it as an example of unpredictable software, Windows software is reliable in the fixed-software scenario of an embedded environment. Most fatal software exceptions or crashes occur when incompatible software applications attempt to use the same memory or hardware resources. Developers usually make sure that all software combinations are well-tested before the release of any fixed-function embedded devices.

One of the main attractions of a Windows-based operating system is the availability of skilled programmers and low-cost parts. Of all desktop PCs, 95% are Windows-based, and an ample supply of programmers knows how to write compatible applications. Of course, desktop and embedded-system applications require different skills, but many of them are interchangeable. Embedded-system developers also have an advantage when selecting Windows-compatible hardware. Along with other processors in the case of Windows CE.NET, both embedded Windows products take advantage of the popular x86 architecture. Vendors produce these processors and support chips in such high volume that costs are low and parts are readily available. A Windows-based system also gives designers plenty of hardware-design flexibility with thousands of peripheral-device drivers to choose from.

Windows also gives embedded-device designers access to a variety of third-party applications to simplify software development. For example, you can purchase off-the-shelf auxiliary functions, such as global positioning, database, or speech recognition, and integrate them alongside the main application software. For example, Sensory offers a fluent-speech software-development kit, allowing developers to easily integrate speech recognition into consumer-electronics products. The speaker-independent fluent-speech engine can recognize several thousand words or phrases and requires no user training. The kit is available for devices us-

![Figure 1](image-url)
ing StrongARM processors running Microsoft’s Windows CE and is available directly from Sensory’s Web site for $1995.

**CODE BLOAT**

In contrast to its positive features, using an embedded Windows operating system has several disadvantages. For example, the code size for any Windows variant is much larger than competing commercial operating systems. This large code means that you need extra hardware, which translates into a higher recurring cost for the embedded device. Microsoft claims that the minimum Windows CE.NET footprint is about 200 kbytes without graphics, but memory requirements typically run about 2 Mbytes when you include graphics and networking routines. A Windows XP embedded runtime image consumes a minimum of 5 Mbytes of memory.

Although it is one of the most capable operating systems for embedded developers, Windows is also one of the most expensive. Users must purchase the proprietary tools for software development and then add royalties to the recurring costs of their products. Microsoft occasionally offers promotional prices on tools when it introduces versions. For example, you can purchase the latest versions of both Windows CE.NET and Windows XP Embedded development tools for $995, less that half of its normal price, through the first quarter of 2003. With free and zero-royalty alternatives available, price is an important deciding factor for embedding Windows.

One of the most often-cited reasons for not developing a Windows-based embedded product is the product’s poor response to real-time inputs. Embedded Windows XP has no inherent real-time capabilities, and thread-switch times extend to the milliseconds range depending on software activity. Although faster processors reduce average response times, they cannot make a non-real-time system deterministic. Venturcom offers RTX (Real Time eXtension), a third-party add-on to Windows XP Embedded that enables both deterministic and non-real-time processing within the same computer (Figure 1). With RTX, maximum thread-switching times drop to 10 to 25 msec. Other embedded-system applications that require deterministic responses may still take advantage of the benefits of Windows XP embedded by using external I/O processors. Although Microsoft designed Windows CE.NET from the ground up as a hard real-time system, much faster commercial RTOSs are available. Microsoft quotes an average response time of only 50 μsec on a 200-MHz x86 system. Windows CE.NET includes 256 levels of thread priority, offers nested interrupts, and deals with priority inversion.

After considering the trade-offs and deciding on a Windows-based embedded development, you will need to examine the product features and hardware requirements to determine whether Windows CE.NET or Windows XP embedded best fits your product. Windows CE.NET best suits smaller devices, such as PDAs, digital cameras, and basic set-top boxes; Windows XP embedded-device applications tend to be more advanced and include point-of-sale controllers, printers, and advanced residential gateways. If Windows CE.NET and Windows XP embedded both fit an application, the final choice depends on design requirements.

**10,000 PARTS**

Microsoft based Windows XP Embedded on the same binary code as the professional version of Microsoft’s XP desktop operating system, and XP embedded is the successor to Windows NT Embedded 4.0. The embedded version includes all of the security, multimedia, Web browsing, power management, and device support of the desktop version but breaks it into more than 10,000 components, so developers can choose elements to build a customized, small-footprint operating system. Windows XP Embedded and the most recent service pack includes a number of new communications, file-system, and development-tool options that may convince a design team to select it for its next project.

One useful new feature for embedded devices is the remote-boot service that enables a Windows XP Embedded-based client device to boot using an image downloaded from a server. This feature

---

**FOR MORE INFORMATION...**

For more information on products such as those discussed in this article, go to www.edn.com/info and enter the reader service number. When you contact any of the following manufacturers directly, please let them know you read about their products in *EDN*.

<table>
<thead>
<tr>
<th>Company</th>
<th>Phone</th>
<th>Email</th>
<th>Website</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applied Data Systems</td>
<td>1-301-490-4007</td>
<td><a href="http://www.applieddata.net">www.applieddata.net</a></td>
<td>Enter No. 311</td>
</tr>
<tr>
<td>Arcom</td>
<td>1-913-549-1000</td>
<td><a href="http://www.arcomcontrols.com">www.arcomcontrols.com</a></td>
<td>Enter No. 312</td>
</tr>
<tr>
<td>Arrow Electronics</td>
<td>1-888-427-2250</td>
<td><a href="http://www.arrow.com">www.arrow.com</a></td>
<td>Enter No. 313</td>
</tr>
<tr>
<td>Avnet</td>
<td>1-800-794-8507</td>
<td><a href="http://www.avnet.com">www.avnet.com</a></td>
<td>Enter No. 314</td>
</tr>
<tr>
<td>Bsquare</td>
<td>1-425-519-5900</td>
<td><a href="http://www.bsquare.com">www.bsquare.com</a></td>
<td>Enter No. 315</td>
</tr>
<tr>
<td>Hitachi</td>
<td>1-770-409-3000</td>
<td><a href="http://www.hitachi.com">www.hitachi.com</a></td>
<td>Enter No. 316</td>
</tr>
<tr>
<td>Microsoft</td>
<td>1-425-882-8080</td>
<td><a href="http://www.microsoft.com">www.microsoft.com</a></td>
<td>Enter No. 317</td>
</tr>
<tr>
<td>Sensory</td>
<td>1-408-537-9000</td>
<td><a href="http://www.sensoryinc.com">www.sensoryinc.com</a></td>
<td>Enter No. 319</td>
</tr>
<tr>
<td>Super Info Number</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>For more information on the products available from all of the vendors listed in this box, enter no. 321 at <a href="http://www.edn.com/info">www.edn.com/info</a>.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
means that embedded devices may operate without nonvolatile storage, such as a hard drive or flash memory. Like its desktop equivalent, Windows XP Embedded includes device-driver rollback capability so that you can recall a previously installed driver if driver issues occur. To enhance products destined for the international market, Windows XP Embedded has support for more than 20 languages. Multilingual user-interface-language packs allow developers to serve multiple markets by adding the desired language pack to a basic English configuration.

In the communications area, Windows XP Embedded includes support for USB versions 1.0 and 2.0. Operating-system components also include 802.1X for secure access to wireless local networks and Ethernet with interoperable user identification, centralized authentication, and dynamic key management. In addition, the IrDA (Infrared Data Association) protocol allows low-cost, line-of-sight communications between any two equipped devices. For local security, Windows XP Embedded includes a multiuser encrypting file system that transparently encrypts each file with a randomly generated key.

For multimedia and television applications, Windows XP Embedded includes support for digital and analog television, as well as a stream buffer engine that enables time-shifting for video recording. It also includes support for the .NET Framework, enabling developers to build and deploy Web-based applications, XML Web services, and standalone applications built upon the Common Language Runtime and .NET Framework class libraries. Windows XP Embedded also supports Internet Protocol Version 6, which provides longer IP addresses, better routing capabilities, and more integrated security.

Microsoft Windows Embedded Studio is a set of development tools for creating a custom Windows XP Embedded image, based on the requirements of the embedded system hardware and software. Design templates in the target designer tool allow developers to easily manipulate and select from the more than 10,000 Windows XP components to create a custom image. A target analyzer probes the target hardware to detect the necessary information to generate a basic operating system. The embedded studio also includes a component designer to assist in converting unique drivers and application software into XP-compatible components.

**TINY FOOTPRINTS**

Windows CE.NET is the latest version of Microsoft’s operating system for small-footprint embedded devices. Microsoft built Windows CE.NET on a different architecture from that of the desktop operating systems. Its design objectives include extensive wireless connectivity, real-time features, and device-independent Web services. Unlike Windows XP Embedded, Windows CE.NET supports the ARM, MIPS, SHx, and x86 families of processors, extending its applicability to a range of embedded products. It supports the ARM720T, ARM920T, ARM1020T, StrongARM, and Xscale ARM processors. MIPS silicon includes MIPS II/32, MIPS 16, and MIPSIV/64 processors with and without integral floating-point units. The Hitachi SH family includes the SH-3, SH-3 DSP, and SH-4 processors, and the x86 units include 486, 586, Geode, and Pentium processors.

In addition to most of the wireless 802.11 communications features of Windows XP Embedded, Windows CE.NET includes provisions for short-range Bluetooth networking. This cable-replacement technology allows enabled devices within a 10m range to communicate over a 2.4-GHz RF link. To enhance device roaming, Windows CE.NET includes MediaSense, a feature that notifies applications of network status—indicating when a system is out of range, for example. More improvements in Windows CE are in the works at Microsoft. The company promises voice-over-IP by 2003 and expanded storage capability in 2005.

Windows CE.NET includes the .NET Compact Framework, Microsoft’s smart-device development platform to provide customers with Web services at any time, any place, and on any device. The .NET Compact Framework allows developers to build and deploy XML Web services to smart devices and allows the execution of downloadable applications on small, portable devices. To support the .NET Compact Framework, Windows CE.NET includes an integrated end-to-end tool set and several customized development languages, such as Visual Basic.NET; Visual C++.NET; and Visual C#.NET, a new, object-oriented, type-safe language. Windows CE.NET also includes a host emulator that allows developers to build and test their designs on a Windows XP desktop workstation before delivering target hardware (Figure 2).

Windows XP Embedded and Windows CE-.NET are available from Microsoft distributors, such as Arrow Electronics, Avnet, Bsquare, and Venturcom in the United States, plus many others worldwide. Distributors and third-party vendors may also provide integration and training services. Several third-party vendors also provide board-support packages and development kits to ease Windows-based integration.

Figure 2

Figure 3

Arcom’s SBC-GX1 development kit includes a single-board computer with 128 Mbytes of DRAM, cables, connectors, a power supply, and Windows XP Embedded.