INtime® 5 Software Development Kit

**INtime Features**

- Field-proven, x86-based, hard real-time OS for cycle times under 50 µs to run time-critical I/Os
- Runs stand-alone on x86 PC platform or alongside an off-the-shelf 32-bit or 64-bit version of Windows, up to Windows 8
- Scalable multi-core platform support from Intel Atom to Xeon processors for up to 31 INtime RTOS kernels
- Development on a Windows PC with fully integrated Visual Studio environment (VS2012/2010/2008), in real-time at "user-mode" level
- Deterministic inter-process communication (IPC) within and across platforms (GOBSnet)
- Dedicated access to I/O interfaces like PCI, Ethernet or USB

INtime® 5 SDK is a comprehensive development environment for the INtime RTOS family, fully preemptive 32-bit OSEs capable of running process loop times of 50µs and less.

The INtime 5 SDK supports two usage configurations:

- **INtime for Windows**, where the INtime RTOS runs simultaneously alongside the Microsoft® Windows® operating system.
- **INtime Distributed RTOS**, where INtime runs as a stand-alone RTOS

Configurations run the same binary application and support multicore implementations with one or more instances of the INtime RTOS running on the same platform.

**INtime for Windows**

Windows alone cannot provide deterministic support for real-time application needs. In applications where Windows is ideal for implementing human-machine interfaces (HMIs) or executing Windows-based applications, INtime for Windows provides real-time services to a standard Windows 32bit or 64bit platform. It enables embedded systems to separate deterministic processes from the Windows task management to maximize system performance and ensure reliable operation. On a multicore processor, Windows can be set up to run on one or more cores, leaving the remaining cores to run one or several independent instances of INtime, one per core. Communication between the Windows and the INtime instances is made possible with Global Objects networking (GOBSnet), a built-in, inter-processor communication network.

**INtime Distributed RTOS**

Designed to be easily installed on any x86 PC platform, INtime Distributed RTOS comes as a complete solution with tools and utilities to facilitate the development, deployment, and field maintenance of embedded real-time applications. All development is performed with the INtime 5 SDK running on a standard Windows-based PC platform. Deployment of application code to the target systems is made simple with a ready-to-use boot-loader. Application-code maintenance and configuration changes in the field are made easy with a built-in http and ftp server. INtime Distributed RTOS allows applications to be scaled over several CPU cores of a multicore processor or spread between processors in a multi-platform system, with applications running on individual instances of INtime RTOS that are loaded on each processor and communicating with GOBSnet.

**INtime 5 Family of RTOS Products**

INtime for Windows platform communicates with the INtime Distributed RTOS platform through GOBSnet. Windows runs an advanced HMI on a dedicated processor core while the other cores run vision and motion control SW.
Improve productivity and reduce time to market

- **INtime uses the same Visual Studio tools as Windows for application development and debug**

**INTime Overview**

- Use Microsoft® Visual Studio® integrated development environment for all edit, compile, link, and debug tasks
- Development environment runs on an off-the-shelf PC with Windows® 8, Windows 7, Vista, XP, and Windows Server 2008 or 2003—32bit or 64bit
- Binaries generated by INtime 5 SDK will run in both usage configurations, INtime for Windows or INtime Distributed RTOS.

**Familiar, easy-to-use development environment**

INtime applications are edited, compiled and debugged using the same Microsoft Visual Studio tools that are used to build standard Windows applications — eliminating the need to learn or purchase other development tools.

The INtime development environment includes Visual Studio project wizards with which basic applications can be built with the simple selection of fundamental code building blocks. These may then be expanded upon as necessary. Integration with the online help system ensures quick access to the details of the INtime real-time application programming interface (API).

The INtime real-time programming environment includes a comprehensive INtime API and a subset of Win32 API which facilitates the process of porting existing Windows-based C-code applications to the INtime environment.

The Windows part of an INtime for Windows application may be developed and debugged at the same time as the INtime real-time application part. Any .NET language may also be used to create the Windows part of an INtime for Windows application.
Library Support

**Full C and C++ support:** The INtime C99-compatible C library and ANSI C++ library include full support for STL. An INtime RtClass library provides a C++ interface to standard real-time objects.

**Shared real-time libraries (RSLs):** INtime Real-time Shared Libraries are similar to Windows DLLs. RSLs provide a means for multiple real-time processes to share a single real-time code library.

**Comprehensive SSE library:** Utilizing the Intel Integrated Performance Primitive static libraries (IPP), developers can quickly implement a variety of signal processing functions on the INtime real-time platform.

I/O Support

Real-time applications always have direct high-speed access to I/O. There is no requirement to learn difficult kernel-mode debugging techniques or create unique device drivers in order to access specialized I/O hardware.

**PCI & PCI Express access:** INtime APIs include a complete set of functions to facilitate the task of locating and initializing a PCI or PCI Express device. Sample code and templates are provided in addition in order to simplify the task of accessing hardware from INtime applications.

**Real-time access to TCP/IP networks:** INtime features a BSD TCP/IP network stack and programming API. Layer 2 protocol access is available through Berkeley Packet Filter (BPF) or TenAys High Performance Ethernet (HPE) stack for high throughput applications.

*USB client development support:* Includes real-time stack support for USB 3.0 host controller interfaces. Using the stack, INtime applications have direct access to USB I/O devices for real-time data acquisition and control.

**Industrial I/O drivers:** Custom or third-party control bus and industrial communication protocols are available and are easily incorporated into INtime applications.

Debug and Run-time Support

**Exception handling and fault management:** INtime includes several levels of exception handling and fault management to improve debugging efficiency and to provide run-time reliability. If an exception is not handled in the application by using structured exception handling or C++ exception handling, the default system handler allows manual selection of a number of methods to debug the fault. In the lowest-level case, a faulting thread is simply suspended, ensuring that the rest of the system will continue to run.

**Distributed System Manager (DSM):** An INtime system consists of multiple processes running in cooperation with each other on a single processor host, a multicore host, or several hosts. The Distributed System Manager is an integrated software service which enables the monitoring of processes and the links between them in order to provide reliability for the whole system. Using simple DSM services, a process may choose to be notified of system-wide events such as network link failure or failure of a host, or of process events such as a client process terminating. Windows-hosted applications which are part of the INtime system may also use the DSM services.

Dynamic object browser: INtex

INtime Explorer (INtex) gives the developer a convenient way to see and interact with the state of processes, threads, and objects while real-time continues running.

INtime Explorer allows the user to browse a system at the object level, to inspect objects in detail and to interact with those objects at a high level. Control functions such as deleting objects, signalling them, suspending and resuming threads can be done interactively to assist in system debugging. Unhandled exceptions can be analyzed with the monitoring and reporting function, generating a post-crash analysis report. A system statistics monitoring function is also available to display CPU usage per-thread and per-process.

Real-time system analysis: INscope

The INtime real-time performance analyzer, INscope, facilitates acquisition of precise time and sequence data of real-time threads. INscope is a Windows application that is used to trace the execution of a real-time application.

The precise timing of real-time events including thread switches, system library calls, and interrupts is displayed on a logic-analyzer type graphical trace.

INscope includes an API to add custom events to the trace log and precisely control trace trigger points. The INscope API can be used to verify operation of time-critical code, as well as proper sequencing of real-time events.
Ordering information

**INTIME5-DK-HWKEY (INtime Development Kit)**
INtime 5 software development kit featuring USB hardware license key.
SDK includes the development sample code, libraries, header files, TCP/IP and USB stacks, and selected device drivers; Visual Studio 2012, 2010 and 2008 plugins; INtime Explorer, INscope and Spider utilities; one each INTIME-MCRT and RTOS-MCRT run-time license; and one year of technical support, updates, and upgrades.

**INTIME5-DK-NETSRV (INtime Network Development Kit)**
Network starter package of six (6) INtime 5 software development seats. Includes network server software. DK-NETSRV content is the same as one HWKEY version.

**INTIME5-DK-ADDNET (Network Licenses Increment)**
Incremental addition of a single INtime 5 software development seat for an existing INTIME5-DK-NETSRV installation.

**INTIME5-DK-MAINT**
Extension of maintenance and support for existing INtime 5 SDK for one (1) additional year. Requires one maintenance extension per each INtime 5 SDK license. Includes all updates and upgrades issued during the period agreement in effect.

System requirements
Minimum requirements for a Windows® PC-compatible host to run INtime 5 SDK:

- Any Intel or AMD processor that runs Windows— including single-core, multicore and hyper-threaded processors
- At least 16MB of DRAM available for dedicated INtime + real-time application usage

Visit the TenAsys website or contact us directly for information regarding device driver availability.