

RTX RTOS Platform

IntervalZero's RTX real-time software transforms Microsoft Windows into a real-time operating system (RTOS). IntervalZero's RTX64 supports 64-bit operating systems and RTX supports 32-bit operating systems.

Overview

RTX64 and RTX are key components of the IntervalZero RTX RTOS Platform that also includes x64 and x86 multi-core multiprocessors and the Windows operating system to outperform real-time hardware such as DSPs and MCUs and reduce the development costs for systems requiring determinism or hard real-time.

Both RTX64 and RTX provide hard real-time determinism and symmetric multiprocessing (SMP) capabilities.

RTX64 takes full advantage of 64-bit memory and performance capabilities. Uniquely, the RTX64 RTOS scheduler enables real-time applications to directly access the 512GB of addressable physical memory available on 64-bit Windows. This is critical to modern-day real-time systems and represents a gigantic leap from the 4GB physical memory limit of traditional 32-bit Windows systems.

Determinism

- Guaranteed precision – set timer periods down to 1 microsecond, and Interrupt Service Thread (IST) latencies of less than 10 microseconds
- Separation from Windows – Windows processes cannot interfere with real-time applications
- Scalability – one scheduler is used across all real-time processors. SMP aware scheduler that utilizes both priority-driven and pre-emptive algorithms to ensure critical thread context switches. Yields to threads of higher priority occurring in the sub-microsecond range.

Control

- Flexibility to configure the amount of processing capability used for real-time processes (up to 63 dedicated processors for RTX64 and up to 31 dedicated processors for RTX)
- Full control of real-time process threads, with the ability to load balance as needed. Developers can set thread and interrupt affinities
- If Windows issues a STOP message or shutdown, real-time applications can continue running to safety shutdown



Simplify

- Use a single operating system for applications
 - RTX64 is supported on Windows 7 SP1, Windows Embedded Standard 7 SP1, Windows 8.1 with Update, Windows Embedded Standard 8.1 with Update, and Windows 10
 - RTX is supported on Windows 7 SP1; Windows Embedded Standard 7 SP1;
- Use commercial off-the shelf (COTS) target system; no special hardware required
- Use one development environment – Microsoft Visual Studio
- Use common languages (C/C++) for Windows and real-time applications
- Use common Win32 API; same code can be run as a Windows or real-time process
- Use managed code for Windows application and still communicate with your real-time applications
- No driver model to follow; real-time process can talk directly to hardware
- Use standard inter-process communication between Windows applications and real-time processes (events, mutexes, and semaphores)
- Use shared memory between Windows and real-time process for sharing of data

Reduce Costs

- Eliminate additional system to perform the HMI
- Eliminate proprietary controller and communications cards
- Improved asset utilization: take advantage of underused multi-core capacity
- Reduced manufacturing costs and fewer physical parts

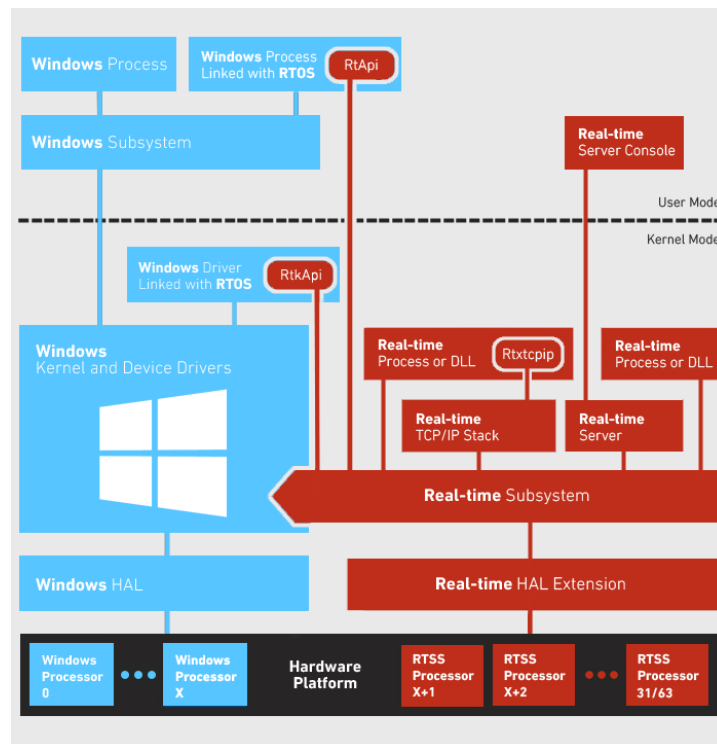
Improve Efficiency

- Eliminate some inventory costs and reduce maintenance costs
- Accomplish field upgrades through software download rather than board replacement

Editions

The edition ...	Includes support for real-time operations on...
Solo	1 dedicated RTSS processor
Entry	Up to 2 dedicated RTSS processors
Basic	Up to 3 dedicated RTSS processors
Professional	Up to 7 dedicated RTSS processors
Premium	Up to 15 dedicated RTSS processors
Ultimate	Up to 63 dedicated RTSS processors (RTX64); Up to 31 dedicated processors (RTX)

Architecture



Key Features

Real-time Runtime

- Native 64-bit (RTX64) and 32-bit (RTX) real-time subsystems
- Scalable from 1 to 63 real-time processors (RTX64), 1 to 31 real-time processors (RTX)
- SMP aware scheduler utilizes both priority-driven and pre-emptive algorithms
- Configurable thread and interrupt affinity
- Configurable timer period
- Ability to attach to line-based and message-based (MSI/MSI-X) interrupts
- Shutdown handling on Windows STOP or shutdown
- Deterministic memory
- Access to Windows file system and registry (custom search path configuration in RTX64)
- Dynamic link library support through real-time DLLs
- Real-time inter-process communication between Windows user processes or Windows kernel drivers and real-time processes
 - Native and managed interface for Windows processes
 - Objects available: events, mutexes, and semaphores
 - Data sharing through shared memory

- Windows user groups for limiting access to real-time subsystem features.
- Tools and Utilities
 - Control Panel – configuration
 - Server Console – display print messages
 - SRTM – view system timer to timer handler response on a given core
 - KSRTM – view system timer to interrupt service routine (ISR) response
 - Latency View – view and compare system timer response latencies across multiple cores
 - Task Manager – displays a list of running RTSS processes
 - Monitor (RTX64) and Time View (RTX) – profiles RTSS applications
 - RTX64Objects (RTX64) and Object Viewer (RTX) – view internal RTSS object states
 - RTX64CPUUsage (RTX64) and Performance View (RTX) – view CPU usage of RTSS processors

Real-time TCP/IP Stack Runtime

- Real-time stack and network drivers that provide TCP/UDP/IP networking support for IPv4 and IPv6 (separate purchasable feature in RTX64).

Software Development Kit

- Headers and libraries
 - Real-time API (RTAPI)
 - Real-time kernel API (RTKAPI)
 - Real-time Network API (RTNAPI)
 - Real-time Network Driver API (RTNDAPI)
 - Managed Code Interface
- Microsoft Visual Studio 2012 (RTX64), 2013 (RTX64 and RTX), 2015, 2017(RTX64 and RTX) and 2019 (RTX64)
 - Application and DLL templates
 - API code snippets
 - Microsoft C Runtime support
 - Debug support
- Microsoft WinDBG extension and RTSS symbols
- Perceptio Tracealyzer for RTX64 – graphical tool to analyze monitoring data
- Sample source to show basic concepts