RTS Hypervisor
Version R5.6.00

Release Notes

Real-Time Systems GmbH
Gartenstrasse 33
88212 Ravensburg
Germany
—
www.real-time-systems.com
—
info@real-time-systems.com

Document: 106.189.221213

Table of Contents

1 About this document

This document describes release-specific information. Please take time now to become thoroughly familiar with this information.

2 Requirements

For additional information, please refer to our document – **Real-Time Hypervisor – Product Description and User's Guide** – in your Real-Time Hypervisor Software Release Package:

\Documentation\RTS-Hypervisor.pdf

2.1 Hardware

- Personal Computer with x86 multi-core processor(s). Supported processors:
  - Intel: Processors featuring virtualization technology (*Intel VT-x*) including support of *Extended Page Tables (EPT)* and *Unrestricted Guest*. This includes Intel x86 processors since 2013. For processors without EPT or Unrestricted Guest use R5.1.02.
  - AMD: Processor families 0x10 (*K10*), 0x15 (*Excavator, Steamroller, Piledriver, Bulldozer*), 0x16 (*Puma, Jaguar*), 0x17 (*Zen, Zen+, Zen 2*).
- For OS memory relocation and / or DMA protection, an IOMMU (Intel VT-d or AMD IOMMU) must be present. This is typically the case for current platforms.

Note that features like Virtualization Technology or VT-d / IOMMU might need to be enabled in the firmware setup.

2.2 Software

- Operating Systems, compatible with x86 platforms
- Real-Time Systems or OS vendor supplied RTS Hypervisor Board Support Packages (BSP) for all Operating Systems requiring real-time and deterministic behavior.

2.3 Preinstalled Software (if EFI Windows is used)

- ESP (EFI System Partition) and Windows installation must be located on the same drive
- Windows installed on a drive connected to an AHCI controller or Non-Volatile Memory (NVMe)
- Standard access to RTC (Real-Time Clock) available

3 New Features and Updates

Please Note: Every new version of the RTS Hypervisor typically contains improvements, performance updates and bug fixes.

Notable changes contained in release version R5.6.00:

- Software license information for the RTS Hypervisor is now included as an SPDX file found in the Documentation folder of the Release Package.
- Support for kernel 5.15 (Real-Time Linux and Xenomai) in the Privileged Mode.
- Support for Linux ramdisks in the Privileged Mode based on Debian 11.
- The RTS Hypervisor configuration file can now reference other configuration files using an include mechanism. This mechanism allows for the separation of platform-specific and platform-nonspecific configuration options.
- The "shared_serial" key allows for both the configuration of serial ports for Hypervisor log output and the assignment of an OS simultaneously. When using the "shared_serial" key, the Hypervisor log output over the serial port is active whenever the OS to which the serial port is assigned is not running.
- Memory maps from the operating systems' point of view are now printed to the Hypervisor log. If the development mode is on, the memory map is printed on every boot of an OS.
- Windows with BitLocker-encrypted drive can now be booted on top of the RTS Hypervisor. An Application Note document is available from RTS on request.
- ACPI tables are now presented to Linux and VxWorks when running in the Privileged Mode.
- Enhanced PCI Configuration Access Mechanism is now enabled for Linux and VxWorks when running in the Privileged Mode with Virtual MMU and Restricted I/O enabled.
- For processors supporting "Intel HwP" (e.g., Tiger Lake, Alder Lake), fixed or dynamic processor core frequencies can be configured on a per-OS basis. For instance, a fixed processor frequency can be configured for real-time operating systems, while core frequencies for a general-purpose operating system (e.g., Windows or Linux) are dynamically adjusted based on the workload.

Previous additions and updates of the RTS Hypervisor since R4.1.00:

Version R5.5.02:

- Support for VxWorks 7 version 21.11.

Version R5.5.01:

- Virtualized Mode performance improvements on processors supporting Posted Interrupts (indicated by log output).
- Support for Intel processors code-named Alder Lake and Ice Lake-D.
- Support for hybrid architecture with "performance" and "efficiency" cores. Refer to "Core assignment in hybrid architectures" in RTS-Hypervisor.pdf.
- Support for "INtime Distributed RTOS 7.0."

Version R5.5.00:

- Support for Windows 11.
- Support for Debian 11 (Virtualized Mode).
- MSI with multiple messages are now supported in all execution modes.

Version R5.4.06:

- Maintenance release only.

Version R5.4.05:

- New drive sharing diagnostics feature. Refer to "Drive Access Timing" in RTS-Hypervisor.pdf.

Version R5.4.04:

- Cache Locking / Software SRAM: Configure Shared Memory areas to have their content always cached, resulting in deterministic and low-latency data access.
- Support for CPU frequency configuration on AMD processors based on the Zen architecture.
- Grub can now be loaded started by the RTS Hypervisor, which allows for bootloader configuration changes to take effect while other operating systems keep running.
- On Intel Core processors the Memory Access Throttle is now effective for uncached accesses as well.
- Added assignment of ACPI devices, see RTS-Hypervisor.pdf.
- Support for Linux kernel 5.10 rt-preempt / Xenomai running in the Privileged Mode.

- New DKMS based components provided for Debian and related Linux distributions running in the Virtualized Mode allow for better maintainability. For details refer to the RTH installation guide for Linux running in the Virtualized Mode.

Version R5.4.01:

- A new Interrupt Mode (3) enables MSI-X emulation for devices only supporting MSI. This is required for certain Intel TSN device drivers. Please refer to RTS-Hypervisor.pdf for details.
- Ethernet logging support for Elkhart Lake integrated network adapters.

Version R5.4.00:

- The ATA TRIM command for shared mass storage devices is now available in all OS execution modes.
- Support for Intel processors codenamed Elkhart Lake and Tiger Lake.
- Internal memory management improvements eliminate previous limitations.
- New configuration key "CPUs_remainder" for logical CPU assignment regardless of which cores the logical CPUs belong to. Please use only if the corresponding "core" keys are not suitable. Refer to section 5.2.8.1 "Cores versus Logical CPUs" in RTS-Hypervisor.pdf.
- Support for QNX 7.1.
- Support for VxWorks 7 SR0650.
- #AC on Split Lock, a Time Coordinated Computing (TCC) feature found in recent Intel processors, can be selectively turned on or off.

Version R5.3.01:

- Automatic installation of bootloader and Hypervisor on preinstalled computers with one of the following Linux Distributions:
  - CentOS 7
  - CentOS 8
  - Debian 10
  - Ubuntu 20.04
  - Ubuntu 18.04

Version R5.3.00:

- Shared NVMe Controller. Assign Drives or Partitions to operating systems
- The Windows service now positions and keeps the Grub2Win boot loader entry at first place in the EFI boot order list

- Support for QNX 7 32-bit and 64-bit
- Support for Xenomai
- Configuration of Cache Allocation Technology (CAT) by percentage of available shared cache
- Support of CAT on AMD Zen processors
- Support for Memory Access Throttle (MAT) on AMD Zen processors
- Auto configuration of CAT and assignment of Cache Segments when using the "CPU_resource_partitioning" key for more than two operating systems

Version R5.2.02:

- Configurable default actions on OS shutdown / reboot.
- Device capability UniqueID for PCI devices supported in Windows.
- Fix: Windows service rthService.exe was generating excessive event log messages when Windows is timesync slave.
- By default, PCI base system peripheral devices behind explicitly assigned PCI-to-PCI bridges, are now assigned to the same OS to which the PCI-to-PCI bridge is assigned.
- Fix for sporadic hypervisor internal error, potentially causing memory corruption (RTV-320).

Version R5.2.01:

- Fix vulnerability that allowed direct access to protected device registers from operating systems running in Virtualized Mode (Ref. RTHSUP-105)
- One-click automatic installation of Boot Loader and Hypervisor on computers with pre-installed MS Windows
- Configuration of Power Button use
- Allow raw_access for guest OS UEFI boot with Drive Sharing enabled
- Various improvements of deterministic system behavior and temporal isolation during reboot of guest OS, restart of the GPU or in case of a graphic mode switch

Version R5.2.00:

- Support of Privileged Mode Linux 4.19
- Bootloader Grub2win update with support for wide-character language packs, like Japanese and overall improved user experience
- Fixed MAC addresses can be assigned for each OS on virtual network
- Events can be received by VxWorks and QNX even if the respective OS is not registered within the event system; providing independence of application start time

- EFI framebuffer can be passed to an OS even if it is located above 4 GB
- Support for multiple root buses in Virtualized Mode
- Restrict API access for timeSync (master, interval)
- API to signal a successful boot of an OS to the hypervisor
- MSI-X support for operating systems in Virtualized Mode
- Booting from NVMe (Non-Volatile Memory)
- Pre-initialized shared memory by a given byte, e.g. 0 or 0xFF
- Improvements in physical memory assignment for operating systems executing in Privileged Mode with Virtual MMU and IOMMU enabled. Typically increasing the amount of memory available, to be allocated to an OS.
- Assign different ports of a single USB Hub, used by multiple Operating Systems to individual operating systems.
- Configure use of Message Signaled Interrupt (MSI) mode as a default
- Instead of a "system reset" (default), a "full system reset" can be configured for rebooting a system
- Support for systems booting in x2APIC mode
- Added Drive Sharing support for disks without a partition table (raw, e.g. fully encrypted disks)
- Allow quotes (prefixed by escape character) in bootline, Example: "bootline" = "text \"text\" text"
- In Development Mode, a count of SMIs that may have occurred can be displayed to the user.
- Hypervisor sample code provided under open source license

Version R5.1.02:

- Virtualized Mode Linux BSP:
  - (rthinitVirt) suppress kernel link updates if modules cannot be compiled
  - (rthinitVirt) Ubuntu 18.04 vnet ip address loss fixed
  - support compiling drivers for kernel 4.20
- Option to accept multiple hardware IDs for the same runtime license
- QNX BSP: bug fix in libRth memory management QNX BSP: bug fix in devb-eide (error due to a timeout)

Version R5.1.01:

- Option Support for AVX and AVX2 in Virtualized Mode
- Support for TSC Deadline mode in Privileged Mode Linux

Version R5.1.00:

- Option to reset the PC after early boot error (before guest OSs have started)
- New default CPU assignment
- New CPU assignment options (percent, remainder)
- Microcode update performed by Hypervisor (Intel only)
- Support for CPUID information and MSRs introduced by AMD firmware update to address the Branch Target Injection vulnerability
- Disable C1E by default. Implicitly disables MWAIT and HLT exits.
- Support of VxWorks 7 32-bit
- xHCI port assignment support for Privileged Mode with memory relocation (Intel and AMD xHCI controllers only)
- Improved configuration parser
- Temperature Sensor support in Virtualized Mode (Intel)
- Intel Cache Allocation Technology (CAT) detection for processors known as Apollo Lake, Skylake, Kaby Lake, Coffee Lake
- Configuration key for automatic CPU cache optimization
- MXCSR re-initialization on guest reboot
- Windows Tools: Starting with Version V4.8.06, the time-zone name is displayed as "UTC+/-hh:mm" instead of "RTH" if a guest OS is in time sync slave mode and the time-zone name provided by Windows is not recognized
- Minor improvements and bug fixes

Version R5.0.01:

- Support for Intel 7th Generation "Kaby Lake" Processors
- Performance improvement for VNET and Event System. Note: Requires Hypervisor as well as Driver Updates
- Other Performance Optimizations

Version R5.0.00:

- Support for native EFI installations without legacy BIOS or CSM
- Intel Cache Allocation Technology (CAT) support for Level 2 Caches, found in Apollo Lake Processors
- Support for Privileged Mode Linux kernel 4.9

Version R4.7.02

- Support for CPUID information and MSRs introduced by Intel firmware update to address vulnerability CVE-2017-5715, also known as Branch Target Injection

Version R4.7.01:

- Support for GZIP compressed OS images
- New API functionalities (refer to libRth.h for details)
- Support for configuring multiple OS Runtimes
- Support for Multiboot 2 framebuffer (print RTH messages to screen when using Multiboot 2)

Version R4.7.00:

- xHCI Controller Sharing / Port Assignment for Intel and AMD controllers
- Intel Cache Allocation Technology (CAT) support for selected Intel processors, e.g. Skylake.
- SMP in Virtual MMU mode on AMD for Linux
- Logging to FAT partition on system fault
- Assignment of PCI devices by Vendor ID only (without Device ID)
- RTOS COM port output redirection to RTH log in Privileged Mode with Restricted I/O
- Assign memory for VxWorks 6.9 32-bit in multiple blocks to avoid conflicts with reserved memory
- i219 Ethernet logging support
- Ethernet Logging support for multifunction devices
- Support for Multiboot 2
- Default assignment of PCI devices behind explicitly assigned PCI-to-PCI bridges
- Print messages about assigned but missing PCI devices only to log and not to screen

Version R4.6.01:

- Improved AHCI error handling for Drive Sharing
- Allow non-existent drives to be assigned to Operating Systems running in Virtualized Mode

Version R4.6.00:

- Linux Kernel 4.1 support in Privileged Mode
- Selectively disable SMI on Skylake Platforms
- API to read Hypervisor Version information in Windows
- RTH performs Function-Level Reset (FLR) by default for all Conventional PCI and PCI Express devices that support it.
- Allow more than eight logical CPUs for QNX 6.6

- Allow more than eight logical CPUs for Operating Systems running in Virtualized Mode
- RTH API function rthSystemPowerButtonStatus() for VxWorks
- Configuration Key to hide Virtual PCI Network adapter from OS
- I/O port sharing between operating systems.

Version R4.5.02:

- RTH enables NMI on the Chipset
- SMP in Virtual MMU mode on AMD for VxWorks, QNX Neutrino, Windows Embedded Compact, RTOS-32
- mem_access_throttle for Intel Broadwell / Skylake and AMD Jaguar / Bulldozer added
- GPT support for shared drives (MBR still required for Windows)
- RTH runs without PC BIOS (CSM) if configured (still required for Windows)
- Support for TBoot on Intel and AMD
- Support for VxWorks 7 Generic x86-64 (Replaces previous BSPs)
- Support for Linux UEFI Installations in Virtualized Mode
- Early serial logging can be enabled in boot line
- Configuration key to disable SMIs on AMD Fusion Controller Hub.

Version R4.5.01:

- In Microsoft Windows, multiple processes may now use shmOpen() if all processes use the same parameters for "size" and "offset"
- Devices may be assigned by the Subsystem Vendor ID and Subsystem Device ID

Version R4.5.00:

- Configuration for implicit Carriage Return and Line Feed for serial logging
- Show real Processor Brand String in Virtualized Mode if configured
- Restricted I/O mode if supported by hardware
- Shared ATA Controller. Assign Drives or Partitions to operating systems

Version R4.4.00:

- Support for AMD processor families 15h and 16h
- Support for VxWorks 7 64-bit

Version R4.3.01:

- Long Bootline Support for Operating Systems in Privileged Mode

Version R4.3.00:

- Memory and image relocation for RTOSs (Virtual MMU with IOMMU)
- NUMA support
- VxWorks 6.9 64-bit support
- QNX Neutrino 6.5 pci-bios-v2 support
- One Windows setup.exe for 32-bit and 64-bit instead of two MSI filesSet

Version R4.2.00:

- Set Hypervisor API Access Rights
- Set Hypervisor SHM Access Rights
- Virtual MMU for RTOSs (Requires processors featuring EPT)
- Provide configuration of GPU frequency limiter
- Windows 'rth' tool provided with source code
- Use of VT-d (I/O-MMU) Configurable
- Minor improvements and bug fixes

Version R4.1.06:

- Support of Physical Address Extension (PAE) with QNX Neutrino 6.5
- Minor improvements and bug fixes

Version R4.1.05:

- HPET assignment for Virtualized Mode
- PCI device visibility for Virtualized Mode
- BayTrail CPU Frequency and Memory Access Throttle support
- Support for Windows Embedded Compact 2013

Version R4.1.03:

- Support of IRQ-Balancing for Guest Operating Systems
- Turbo Mode can be configured to be turned on
- OS control API calls for "system shut down" and "system reboot"
- Allow HPET counter read access for Privileged Mode Operating Systems
- System reset on INIT IPI (watchdog) if configured
- Support for Display Brightness Control in Virtualized Mode

Version R4.1.02:

- Virtualized Mode Memory Access Throttle to reduce impact on RTOS
- Enhanced Intel Speed Step Technology support
- Support for VIA Eden X2 processor

- Log Level configurable in bootline

Version R4.1.00:

- Instead of two different binaries for 32-Bit and for 64-Bit processors, only one RTH binary (rthx86) is used and replaces the former rthi386 and rthx86-64 binaries
- The Control Module "rthCtrl.out" is no longer used or required
- Privileged Mode Linux PAE support has been added
- Privileged Mode Linux 64 bit is supported

4 Installation

Please refer to your RTS Hypervisor Installation Guide.

4.1 Upgrading / Compatibility

Upgrading existing platforms does not require new operating systems images to be built. Previous installations are compatible with the latest version of the RTS Hypervisor.

For upgrading a System refer to the new installation guide for your RTS Hypervisor provided with the product.

5 Notes

5.1 OS Compatibility

The following table show the compatibility status of the current RTS Hypervisor version with various operating system versions.

OS compatibility status information:

- **Featured**: Compatibility for the OS version is actively developed and updated in every RTS Hypervisor release.
- **Maintained**: This OS version is no longer actively developed with, but the RTS Hypervisor is still compatible with it. No need to re-install or rebuild OS images when updating the RTS Hypervisor.
- **Third Party**: The BSP for the OS is developed and maintained by a third party, typically the OS vendor.

General-Purpose Operating Systems (GPOS):

OS	Version	Status
Ubuntu	22 LTS	Featured
Ubuntu	20 LTS	Maintained
Ubuntu	18 LTS	Maintained
Debian	11	Featured
Debian	10	Maintained
Debian	9	Maintained
CentOS	8	Maintained
CentOS	7	Maintained

OS	Version	Status
Windows	11	Featured
Windows	10	Featured
Windows	7	Maintained

Real-Time Operating Systems (RTOS):

OS	Version	Status
INtime Distributed RTOS	7.0	Third Party
Real-Time Linux	kernel 5.15 / Debian 11	Featured
Real-Time Linux	kernel 5.10 / Debian 11	Featured
Real-Time Linux	kernel 5.10 / Debian 10	Maintained
Real-Time Linux	kernel 5.4 / CentOS 8	Maintained
Real-Time Linux	kernel 5.4 / Debian 10	Maintained
Real-Time Linux	kernel 4.19 / CentOS 8	Maintained
Real-Time Linux	kernel 4.19 / Debian 10	Maintained
Real-Time Linux	kernel 3.12 / CentOS 6	Maintained
Real-Time Linux	kernel 4.1 / CentOS 6/7	Maintained
Real-Time Linux	kernel 4.9 / CentOS 6/7	Maintained
Linux RedHawk	kernel 3.16 RedHawk 7 / CentOS 6/7	Maintained

OS	Version	Status
Linux Xenomai	kernel 5.15 / Debian 11	Featured
Linux Xenomai	kernel 5.4 / Debian 10	Maintained
Linux Xenomai	kernel 4.19 / CentOS 8	Maintained
Linux Xenomai	kernel 4.19 / Debian 10	Maintained
OS-9	5.1 through 6.1	Third Party
QNX Neutrino	7.1.0	Featured
QNX Neutrino	7.0.3	Featured
QNX Neutrino	6.6.0	Featured
QNX Neutrino	6.5.0 SP1	Featured
RTOS-32	5.16 through 6.25	Third Party
T-Kernel	2.0	Featured
VxWorks	7 21.11	Featured
VxWorks	7 SR0650	Maintained
VxWorks	7 SR0620	Maintained
VxWorks	7 SR0610	Maintained
VxWorks	6.9.4.12 64-bit	Maintained
VxWorks	6.9.4.12 32-bit	Maintained
VxWorks	6.9.4.8 64-bit	Maintained

OS	Version	Status
VxWorks	6.9.4.8 32-bit	Maintained
Windows Embedded Compact	2013	Featured
Windows Embedded Compact	7	Featured
Windows Embedded CE	6.0	Maintained

5.2 Operating System Support Status

While this version of the RTS Hypervisor may still be compatible with the operating systems listed below, Board Support Packages, drivers, setup and related tools are no longer subject to active development.

No longer under development is support for:

- Windows XP
- Windows Vista
- Windows 7
- Windows 8.1
- Windows Server versions
- Windows Embedded CE 6.0
- VxWorks 7 SR650 and older
- QNX 6.5.0 SP1 and older
- Linux kernel 5.4 and older (Privileged Mode only)
- CentOS versions
- Ubuntu 20 LTS and older
- Debian 10 and older

5.3 Operating System Support by Third Parties

RTS Hypervisor Support for the operating systems listed below are provided by third parties, i.e., the operating system vendors directly. For availability of supported features and compatibility please consult with the respective operating system vendor.

Under Development by Third Parties:

RTOS-32	On Time Informatik GmbH
T-Kernel	Personal Media Ltd.
OS-9	MicroSys Electronics GmbH / RTSI Inc.
INtime	TenAsys Corporation

5.4 New notes for this version

- The RTS Hypervisor has supported legacy BIOS, EFI firmware, and EFI firmware with active Compatibility Support Module (CSM) which provides legacy BIOS compatibility. Now with the EFI firmware becoming the standard for x86 platforms we have decided to discontinue the RTS Hypervisor compatibility with legacy BIOS and CSM. The current RTS Hypervisor version (R5.6) will be the last to include compatibility for legacy BIOS and CSM, which also includes Windows 7 installations. The next major version (R5.7) will then only be compatible with EFI firmware.

5.5 Notes for previous versions

These notes of previous versions still apply to the current version.

- OS sections typically contain Runtime sections (see "Runtime Path Declaration" in RTS-Hypervisor.pdf) referencing OS image files. The RTS Hypervisor stores all configured image files in memory for later boot, reboot, or reload through our API (refer to "Functions for Loading a new image" in RTS-Hypervisor.pdf). For compatibility with operating systems incapable of accessing data beyond 32-bit addressable memory image files are stored in the first 4 gigabytes of system memory. On systems with little memory (e.g., only 1 GB) mapped to the first 4 GB configuring large image files (e.g., hundreds of MB) presents a limitation in the number of loadable operating systems. If this limit is exceeded loading or booting operating systems will fail.
  In this case users have the following options:
    - If max_image_size_*n* is set to a high value, reduce it.
    - If large Shared Memory Partitions are configured, reduce their sizes.
    - Load smaller image files. Consider installations on mass storage partitions rather than large ramdisks.
    - Configure fewer operating systems.
    - Check if firmware settings increase the amount of memory mapped below 4 GB. Refer to the board vendor.
    - Select a different system with more memory below 4 GB.
- The amount of usable memory below 4 GB is printed to the log by the RTS Hypervisor.
- Naturally 32-bit operating systems without support for PAE are incapable of accessing device memory above 4 GB. This can typically be worked around by disabling the corresponding option in the firmware setup, e.g., Advanced -> PCI configuration -> Memory Mapped I/O above 4GB

- Version R5.2.00 and later requires the use of CPUs with Second Level Address Translation (SLAT, aka "Nested Paging") support. (see Section 2.1). The configuration key "VTLB_mode" is no longer available.
- Intel 82576 (0x8086/0x10C9) is no longer supported for hypervisor Ethernet logging
- A device configured for device sharing cannot be assigned to an OS explicitly
- Hypervisor sample code now provided under open source license
- The way how privileged mode Linux signals a successful boot to the hypervisor changed. Previously, a successful boot was signaled unconditionally from within the Linux kernel's main.c (before console_init()). Now the successful boot of a privileged mode Linux is to be signaled in the rthOSCtrlTask (See Privileged_Mode_Linux.pdf).
- On Laptop computers certain ACPI functionality, such as special keys, may not be available to guest operating systems.
- All Linux kernels provided and supported by the previous RTH releases (R4.7.01 and older) must be compiled with the tools (gcc and binutils) found in CentOS 6 or CentOS 7 operating systems. Compiling any of these kernels (3.12, 3.16 or 4.1) with newer gcc versions (5.x or 6.x) is not supported and will result in an unbootable system image.
- If Time Sync is used and Linux is configured to be Master, the Time Zones will be considered as a default. Versions before R4.7.00 used UTC without Time Zones as a default.
- Features like e.g. CPU and GPU frequency settings are available on certain hardware platforms only, indicated by Hypervisor log messages.
- In order make use of the disk sharing in Microware OS-9 and T-Kernel please contact your OS vendor.
- High Precision Event Timer (HPET) may not work as expected for operating systems executing in Virtualized Mode. Please contact RTS Support if you urgently require this functionality.
- If logging via serial port is enabled (see configuration key "serial" in Section [/LOG]), the COM-Port used for logging will no longer be available to guest operating systems as the RTS Hypervisor since Version R4.3 may provide some log messages during runtime.
- The use of Intel onboard Graphics Adapters may result in increased latencies. Please contact Support at Real-Time Systems for additional information.
- Reboot of operating systems do not go through a hardware reset. Certain device drivers that require a hardware reset to work properly may not function properly after a selective reboot.
- Reboot of a virtualized operating system may lead to increased latencies depending on the hardware platform used (always use latest version for best results).
- Logical CPUs found in Hyper-Threading Architecture of a given Core should always be assigned to the same operating system if real-time performance is critical.

6 Known Issues

- Drive sharing based on NVMe results in a BSOD. Affected OS is Windows 11 version 10.0.22621.
- Due to a new dhcp client introduced by Debian 11 the behavior of network init for Privileged Mode Linux changed with the new Linux_BSP_R2.4:
  - With Privileged Mode Linux images prior to Linux_BSP_R2.4 all NIC assigned to Privileged Mode Linux were initialized automatically on boot.
  - In Linux_BSP_R2.4, the dhcp client hast to be called manually after boot.
- Considerations for using BitLocker with the RTS Hypervisor:
  - Only systems which provide the CRB and FIFO interfaces for the TPM are supported.
  - TPM and BitLocker tools may not work when Windows runs on top of the RTS Hypervisor.
  - BitLocker must be enabled in Windows running natively, i.e., without RTS Hypervisor.
  - Rebooting Windows on top of the RTS Hypervisor with BitLocker enabled results in a black screen.
- Configuring an OS in the Privileged Mode with `"default_shutdown_action"` set to a value of 5 (system turn-off) may not work. To work around this the values 3 (system shutdown) or 6 (system reset) can be used. Please refer to the section "Shutdown and Reboot" in RTS-Hypervisor.pdf. The general ability to turn off the system after all operating systems have halted or shut down is not affected.
- Even if more than one CPU is assigned to the INtime demo version included in the RTS Hypervisor Release Package, only the first "INtime Node" is usable.
- Linux running in the Privileged Mode may fail to format a partition on an SSD / NVMe drive if Drive Sharing is configured.
- In the Virtualized Mode USB input devices don't work through EFI services. This is an issue if input is needed in the early boot phase before the OS has loaded its USB drivers.
- Configuring more than 20 operating systems may result in an OS load or boot failure, regardless of memory and CPU assignment.
- Using NVMe drives or drive sharing on platforms with CSM (legacy BIOS) is not recommended and may fail. We recommended using EFI based installations.
- Drive sharing with Virtualized Mode Linux: If Linux tries to write to a part of the disk that is not assigned to it, the Hypervisor will return an error for this disk command. Linux will then retry endlessly to write the data and "ATA read DMA error" messages are seen on system console.

- During reboot of a guest OS the RTS Hypervisor may issue a Function-Level Reset (FLR) to PCI devices which are assigned to the rebooting OS. If after the FLR a device doesn't work as expected or the driver fails to initialize the device, we recommend disabling the FLR in the RTS Hypervisor Configuration File. See "Function-Level Reset" in RTS-Hypervisor.pdf for available options.
- If an xHCI driver is used, a minimum of 128MB memory below 4GB must be assigned to QNX
- Windows 10 2015 LTSB is not supported (Use Windows 10 2016 LTSB instead)
- Due to a known software bug in the Linux driver for Intel C600 SAS Controller (isci), do not load the isci module along with its dependent modules.
  Depending on your distribution a possible way to do this could be by adding the following line to /etc/modprobe.d/blacklist.conf:

  ```
  blacklist isci libsas scsi_transport_sas
  ```

  NOTE: The initramfs may have to be rebuilt in this case.

  Alternatively you may add "modprobe.blacklist=isci" to the kernel command line in which case the initramfs does not need to be modified.
- AMD CPUs:
  - Turning on IOMMU when using the on-chip GPU may not lead to the desired results.
  - T-Kernel cannot be used in Virtual MMU Mode.
- VIA CPUs:
  - No support for "Virtual MMU Mode"
  - "Fully Virtualized Mode" is not supported for operating systems requiring PAE or running in 64-Bit mode.
  - INTx interrupts dispatched by the same I/O APIC cannot be assigned to different operating systems.

7 Technical Support

Software updates or patches, answers to questions, or advice and guidance are provided to registered users by knowledgeable software engineers via telephone or e-mail. Real-Time Systems GmbH also provides registered users with E-mail announcements regarding this and other related products.

Please send inquiries to:

AMERICAS:	**support-US@real-time-systems.com**
Japan:	**support-JP@real-time-systems.com**
EMEA and ROW:	**support@real-time-systems.com**