QXi-6000 Computer System Board Manual

Version 1.6

Website: www.quixant.com
Email: support@quixant.com
Telephone: +39 0765 030 023
Fax: +39 0765 400 408

Document Revision History

Version	Date	Comments	Author
1.0	2016/12/02	First revision of the document	Paolo Ruscitti
1.1	2016/12/05	Paragraph numbering correction	Paolo Ruscitti
1.2	2017/02/13	Some improvements: COM Port table Intrusions table Description of drivers and libraries	Paolo Ruscitti
1.3	2017/03/02	Updated content in section 21.5.1	Paolo Ruscitti
1.4	2017/05/08	Removed paragraphs 17.2 and 18.18	Paolo Ruscitti
1.5	2017/07/24	Minor changes	Paolo Ruscitti
1.6	2018/11/16	Corrected number of monitors supported in sections 5.1 & 5.3 and updated default setting for JP14 in section 14.2	Mike Riley

For further information, please visit www.quixant.com

Contents

1 Introduction

The QXi-6000 from Quixant is an advanced computer system, designed to provide slot machine manufacturers with a high performance, compact, low cost controller for slot machines. The QXi-6000 platform consists of the QXi-6000 motherboard enclosed in a secure aluminum case that also acts to efficiently dissipate heat from the internal electronic components. The QXi-6000 hardware is complimented by Quixant extensive range of device drivers, libraries and gaming protocols, both on Windows and Linux platforms.

The QXi-6000 architecture is based around the Personal Computer X86 system architecture, which provides numerous benefits over older, proprietary architectures, including high performance, simple application development and a wide range of compatible software & hardware. However, standard PC implementations also have numerous inherent limitations when it comes to gaming. The QXi-6000 has been specifically designed from the ground up to deliver the benefits of PC architecture but at the same time eliminating the characteristics of the standard PC architecture that are undesirable for the role of a gaming controller.

Some of the advantages the QXi-6000 provides over a standard PC solution include:

- No hot, unreliable components
- Entirely fanless operation for most applications eliminates unreliable fans
- Sealed case with no ventilation holes ensures both security and restricts the entry of foreign matter
- Patented passive thermal design effectively cools entire system in a silent and reliable manner
- Long component product lifetime with strict configuration control
- Advanced security features
- Single +12V power input
- Fast and enhanced Non-Volatile memory (NV-RAM)
- Fast boot times
- 6 serial ports
- 2xCFast (SATA 3.0) sockets, up to 6Gb/s
- Battery powered logging processor
- 32 digital inputs – possible to expand up to 64
- 32 digital outputs– possible to expand up to 64
- Gaming BIOS supporting Secure Boot
- Electrically compatible with the QXi-300, and QXi-4000.
- Mechanically compatible with the QXi-300, and QXi-4000 (same mounting holes, same position for the connectors)
- Software API compatible with most of the Quixant platforms.

This user manual describes the functionality of the hardware platform, and provides all the information necessary for integration into a complete gaming machine.

2 QXi-6000 Motherboard Specification

2.1 APU/CPU

AMD Embedded R-Series “Merlin Falcon” SOC (System on a Chip) APU (Accelerated Processing Unit) Integrates the CPU, GPU, ECC memory support, and I/O controller on the same chip.

- Both dual and quad-core embedded 3rd generation “Excavator” cores
- High performance 64-bit architecture (AMD64)
- Speed grades from 1.6GHz to 2.1GHz
- 64KB L1 cache per core
- Up to 2MB L2 cache (shared per core pair)
- MMX, SSE, SSE2, SSE3, SSE3+, SSE4a, SSE4.1/4.2, AVX1.0/1.1, XOP & FMA4, AES and F16C instruction support
- Full Out of Order (OoO) execution
- Virtualization technology
- Insyde BIOS, dual SPI ROM with write protection function

2.2 Main Memory

- Up to 32GBytes DDR4-2133 (PC4-17000) memory
  - Memory can be ECC
- Dual SODIMM sockets
- Dual DRAM banks, giving 128-bit bus width

2.3 APU Integrated Graphics Controller

- AMD Radeon HD10000 Series high-performance graphics controller
- Third-generation Graphics Core Next (GCN) architecture
- 80 shader cores
- Discrete-level graphics performance
- Support for up to three independent displays from the APU
- Three DisplayPort interfaces
  - DP++ interfaces support DisplayPort 1.2, DVI, VGA, and HDMI
  - MST support for daisy-chaining of monitors (Up to 3 monitors)
  - Q-Port interface support
- DirectX® 11, Shader Model 5, OpenGL® 4.2, OpenCL™ 1.2, DirectCompute compatible
- 512MB to 2GB of shared system memory (UMA)
- Highly optimized 128-bit 2D graphics engine:
  - Microsoft DirectDraw® support
- Advanced 3D graphics acceleration:
  - Full DirectX®12, OpenGL® 4.5 support
- AMD Accelerated Parallel Processing (APP) Technology
  - OpenCL™ 2.0, Mantle, Vulkan™, and, DirectCompute 11 support, and Microsoft C++ AMP
  - Accelerated video encoding, transcoding and upscaling
- Advanced UVD 6.0 dedicated playback accelerator for: 4K60 H.265/HEVC, 4K H.264 AVC, MPEG-2, MPEG-4, H.264, DivX, AND VC-1
- VCE 3.1 hardware encode of 4K H.264 video

2.4 Communication Interfaces

- 6 x serial ports
  - 16550 compatible
  - RS232, RS485,cctalk, iButton and JCM ID0003 support
  - 2 x Micro-fit connector (J1) on motherboard, 4 x DB9 COM ports on front panel
  - Quixant QxCOM PCIe 16559 serial option for each port
- Dual PCI Express® 10/100/1000Mbit/s Ethernet LAN RJ-45 ports (RTL8111G)

- USB: External 2 x 3.0 & 2 x 2.0 Ports, Internal 4 x 2.0 header
- 1 x SPI interface for clock serial peripherals (e.g. SEC meter)
- 1 x I^2C (Inter-Integrated Circuit) bus for connecting peripheral devices

2.5 Power Requirement

- Single +12V nominal input designed to work from main cabinet supply
- Maximum power draw around 39 to 71W, depending on APU type
- No need for dedicated ATX-type PSU, increasing reliability and saving cost
- Power failure monitoring with early warning system

2.6 Storage

- 2 x CFAST (CFast 1.0 / SATA 3.0) sockets on board, up to 6Gb/s
- 2 x SATA 3.0 sockets, up to 6 Gb/s (for SATA DOM, SSD, HDD) with power on pin 7 optional
- Capability to mount 2 x internal 2.5inch HDD/SSD SATA drive

2.7 Gaming Hardware

- 3 x Lithium-Ion CR2450, high power density, 3V battery backup with a 5-year battery life
    - Optional Lithium Pentoxide rechargeable cells
- Up to 16MBytes of NVRAM
    - Non-volatile MRAM or battery backed SRAM
    - Ultra-fast PCI Express® connection
    - Hardware accelerated mirroring and CRC support
    - Dual independent physical banks
    - Hardware automatic bank mirroring and error detection
    - Hardware CRC generation and checking of NVRAM areas
- Logging Processor with battery backup
    - 13 intrusion inputs
        - Configurable as standard and A/C driven opto switches
    - Date/time stamped recording of 64 events
    - Auto system switch on (time)
    - Programmable “always on” watchdog timer
    - Battery voltage monitoring & warning
    - System reset logging
    - Interrupt-driven architecture
    - Automatic meter handling
- Advanced digital I/O connected via high speed PCI Express® - no protocol overhead
    - Single PCI Express® read or write can access all inputs or outputs at once
    - 32 digital inputs
        - Individually configurable for interrupt generation on rising, falling, or both edges
        - Input pulse width measurement and screening
        - Configurable hardware debounce filtering
    - 32 digital outputs
        - 24 x 50V 350mA open drain with overload protection
        - 4 x 50V 3A open drain
        - 4 x TTL level
        - Feedback available – for open circuit and short circuit detection.

2.8 Security

- 128/256-bits AES Hardware Decryption & Encryption Engine on PCI Express® bus
- Guaranteed unique electronic serial number

- Optional Trusted Computing Group TPM 1.2 support
- Secure I^2C hardware engine for SHA-1 hash algorithm
- 1Kb EEPROM with SHA-1 MAC engine
- AMD Platform Security Processor (PSP)
- AMD Secure Virtual Machine Architecture
- SEC Meter
- 1 x Key lock

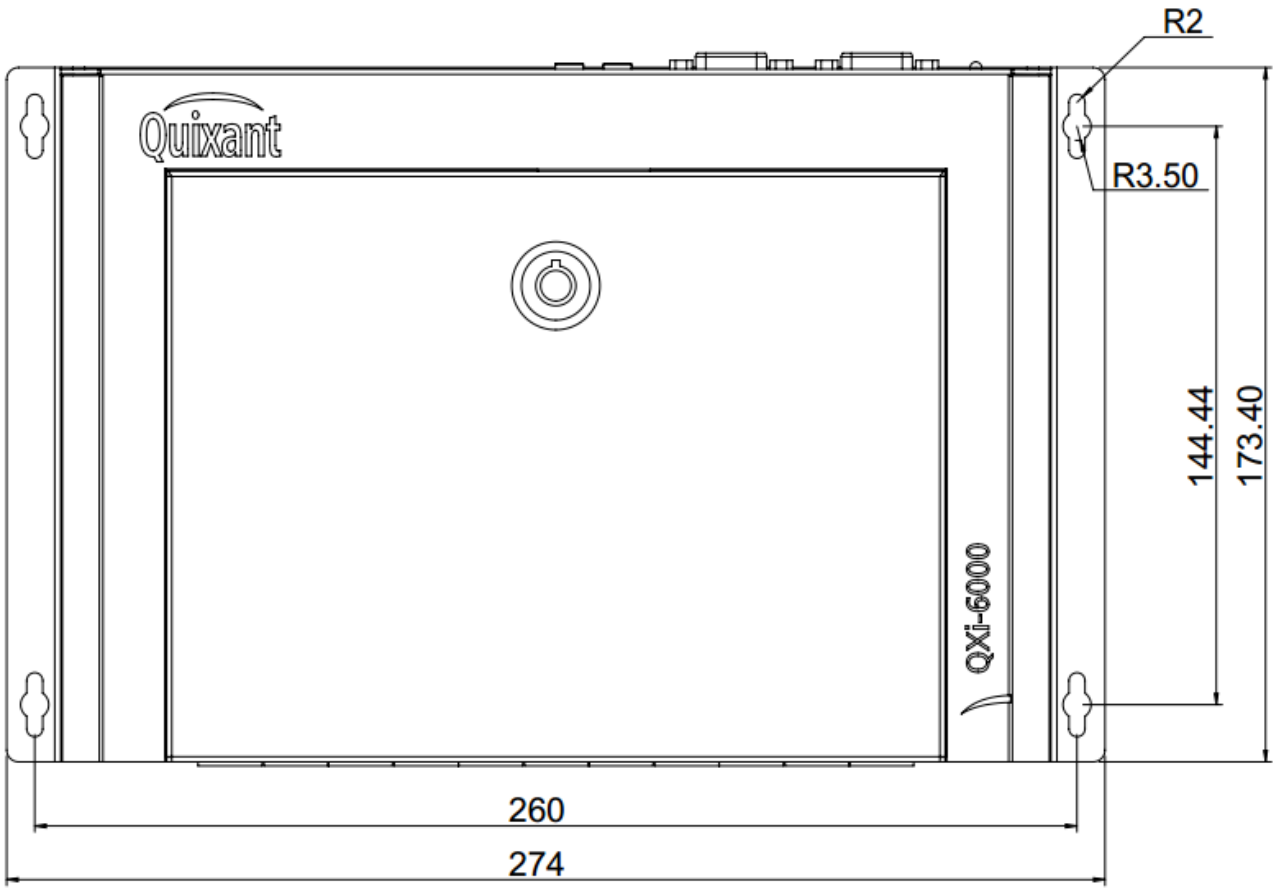
- 1 x Intrusion switch for monitoring door access

2.9 Audio

- 7.1. channel HD audio codec (Line In/Out, Microphone In, C/Subwoofer, Side, Rear)
- 18W RMS (4-ohm default) per Channel Stereo Class-D amplifier for direct speaker drive
  - Option to order configured for 8 ohm speakers
- Stereo line input
- Microphone input

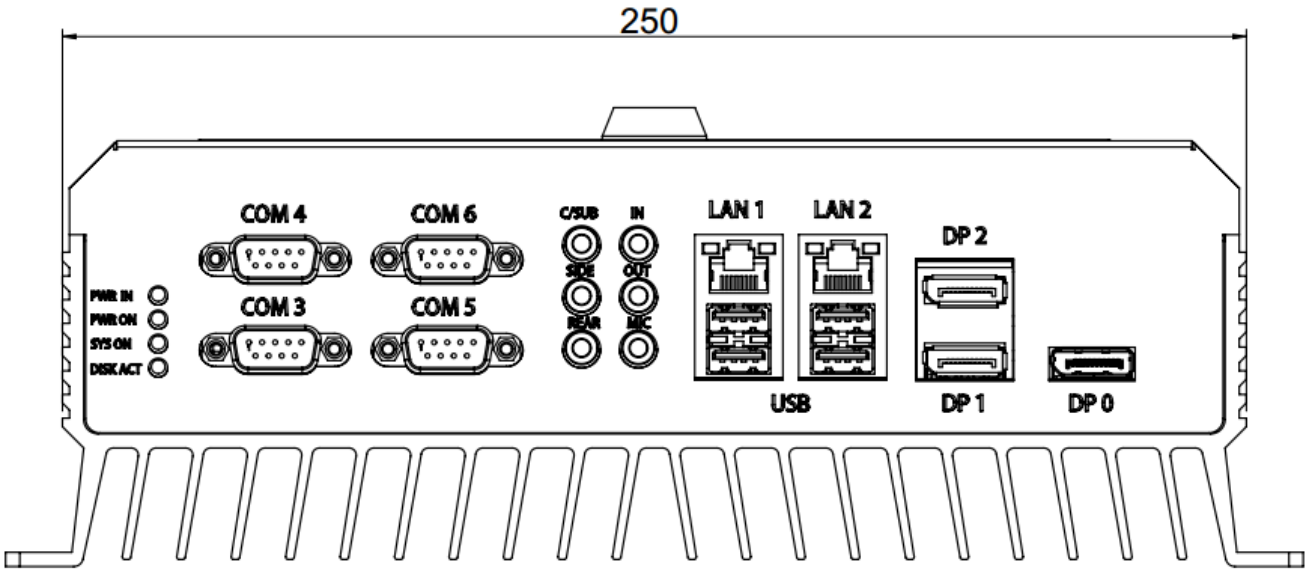
3 Mechanical Drawings

3.1 Plan View Showing Mounting Hole Locations

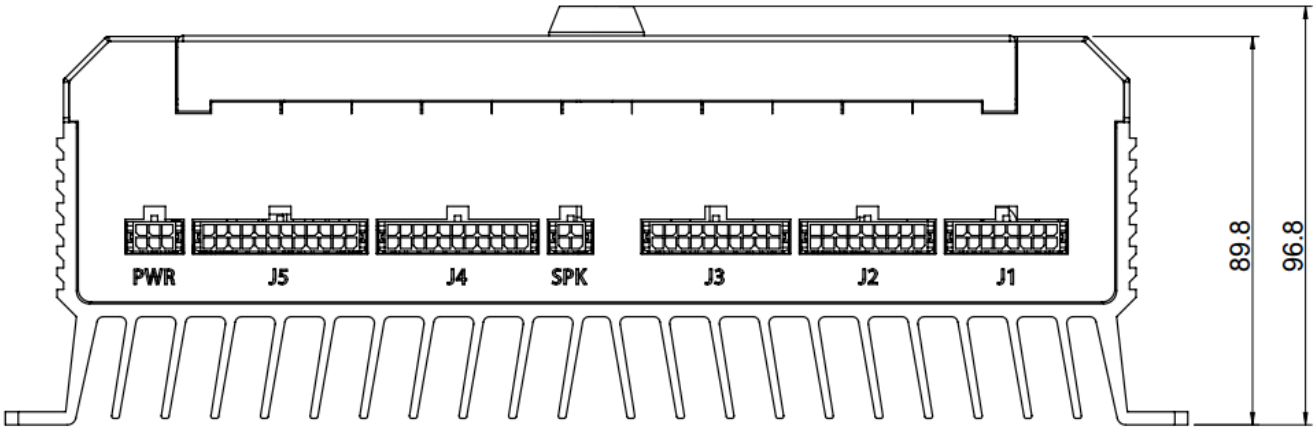


3.2 Front View

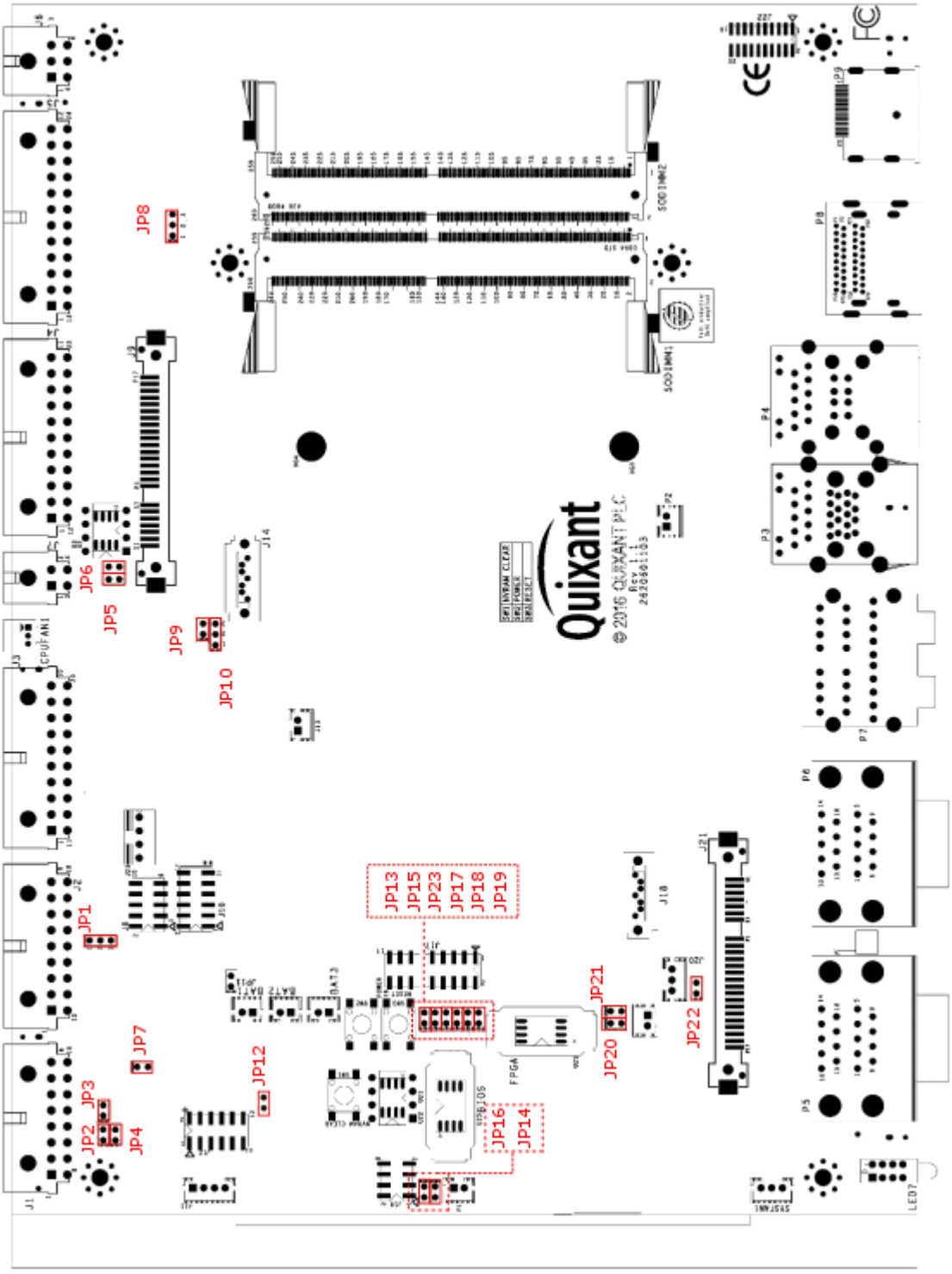
Note that either the front or the back of the QXi-6000 must be oriented so that one of them is facing up. This places the vanes for the heat sink in a vertical position, which promotes air convection for cooling. If the vanes are in a horizontal orientation, or are lying flat, airflow is greatly reduced and limits how much heat can be dissipated to cool the system.



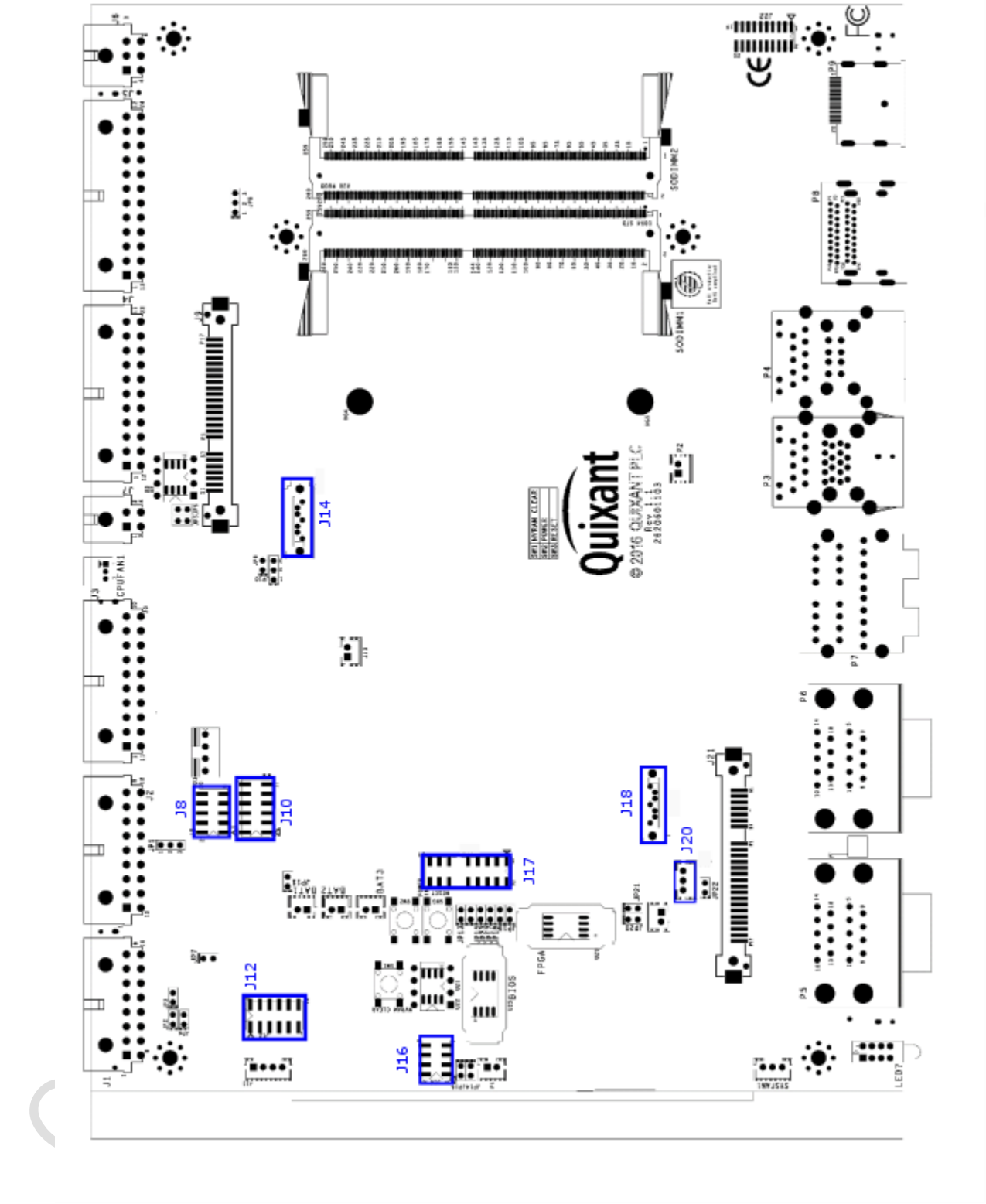
3.3 Rear View



3.4 Jumper Positions



3.5 Connectors Positions



4 QXi-6000 PC Chipset Features

4.1 AMD R-Series SOC

The QXi-6000 utilizes processors from AMD's R-Series SOC, formerly codenamed "Merlin Falcon", which are based on a FP4 package standard and on a sophisticated SOC architecture that integrates up to four of AMD's latest high-performance "Excavator" x86 CPU cores with the latest AMD Radeon™ graphics and an I/O controller. All that guarantees considerable graphics performances, with true 4K decode and encode. These processors are designed for use in embedded systems and are optimized for high performance with low heat generation.

At the time of launch there are the variants of R-Series SOCs used as follows:

Model	Cores	CPU Clock	L2 Cache (shared)	GPU (CUs)	GPU Clock	TDP
RX-421BD	4	2.1/3.4GHz	2MB	R7 (8)	600/800MHz	12-35W
RX-216GD	2	1.6/3.0GHz	1MB	R5 (4)	464/800MHz	12-15W

The AMD Embedded R-Series SOC is compliant with Heterogeneous System Architecture (HSA) 1.0 technology, which boosts parallel processing performance, balancing workloads between CPU and GPU. HSA is also said to reduce latency while maximizing access to 2MB of shared L2 cache using Heterogeneous Uniform Memory Access (hUMA) technology.

The AMD Embedded R-Series SOC provides astounding graphics and high multimedia performance, including capability of true 4K encode and decode and integrated support for DirectX® 12, Unified Video Decode (UVD) 6 (4K H.265 and H.264 decode) and Video Coding Engine (VCE) 3.1 (4K H.264 encode). Up to three displays are supported with options to use the Embedded DisplayPort (eDP) 1.4, DisplayPort (DP) 1.2, Digital Video Interface (DVI) or HDMI™ 1.4/2.0 interfaces.

The AMD Embedded R-Series SOC features a configurable thermal design power (cTDP) capability that allows designers to adjust the TDPs from 12W to 35W in 1W increments to tune power consumption and providing a greater flexibility. All the chips support a 0°C to 90°C range, and this is the first time the R-Series is offered in an industrial -40 to 105°C version. Moreover, the AMD Embedded R-Series SOC has a 35 percent reduced footprint when compared to the 2nd Generation AMD Embedded R-Series APU, making it an excellent choice for small form factor applications.

The R-Series SOC is also the first R-Series to support faster DDR4 RAM. More specifically, it's claimed to be the first embedded processor with dual-channel 64-bit DDR4 or DDR3 with Error-Correction Code (ECC). RAM speed ranges up to DDR4-2400 and DDR3-2133, and there is support for 1.2V DDR4 and 1.5V/1.35V DDR3 operation.

The R-Series SOC also supports interfaces including PCIe Gen 3, USB 3.0, SATA 3, SD, GPIO, SPI, I2S, I2C, and UART, among others. Other embedded-friendly features include a dedicated AMD Secure Processor that supports secure boot with AMD Hardware Validated Boot (HVB) and initiates trusted boot environment before starting x86 cores.

4.2 AMD High Performance integrated FCH

The Merlyn Falcon integrated FCH provides:

- USB 2.0 & 3.0 ports
- SATA 6Gbps ports
- GPIO, SPI,I2S,I2C, UART
- BIOS Interface
- Legacy functionality
- PCIe Gen3

4.3 Main System Memory

The main memory specified on the QXi-6000 is DDR4 type in the 64-bit, 260-pin SODIMM module format. DDR4 memory is the fourth generation of the Double Data Rate synchronous dynamic random access memory technology and supersedes DDR3. DDR4 provides significant speed advantages over older DDR memory.

The QXi-6000 DDR4 memory connects directly to the CPU itself, so the speed specification for the memory is defined by the CPU used. Quixant installs the correct speed and specification of DDR4 SODIMM to match the CPU and ensure optimum performance.

Only SODIMMs supplied and validated by Quixant should be used with the QXi-6000. The SODIMM speed grade needs to be DDR4-2133 (PC4-17000) or better. The actual operating speed depends on the model of APU installed, but all APUs listed in section 4.1 operate at 2133MHz.

The two SODIMM sockets on the QXi-6000 connect to two separate memory channels on the APU, providing a 128-bit memory interface for Merlin Falcon models of the system. In order to ensure optimum performance, it is **essential** that two identical SODIMM modules are installed to enable the full potential memory bandwidth to be achieved.

Each SODIMM socket can contain a 4GB, 8GB, or 16GB DDR4-2133 RAM and dual RAM sticks must be the same size. So the maximum memory is 32GB.

The RAM modules can be either standard or ECC type memory modules.

5 QXi-6000 Video Architecture

One of the key features of the QXi-6000 is its advanced graphics architecture. Utilizing leading edge AMD Radeon™ graphics technology it enables the creation of games with exceptional multi-screen graphical content, resulting in more attractive games for players.

The QXi-6000 provides a wide range of options and price/performance points for graphics that enables customers to tune the purchased system to their exact needs. The GPU integrated inside the APU has characteristics that vary dependent on the model of APU used.

5.1 APU Graphics Controller

The QXi-6000's APU (Accelerated Processing Unit) includes an advanced HD10000 series GPU on-chip, providing excellent 2D and 3D performance along with advanced standard and high definition video decoding and encoding capability. It features 128-bit 2D and 256-bit 3D pipelines. The features of the integrated GPU vary according to the APU model as follows:

Model	GPU Model	Compute Units	Stream Processors	Standard Clock	Boost Clock
RX-421BD	R7	8	512	600MHz	800MHz
RX-216GD	R5	4	192	464MHz	800MHz

General features provided by the GPU integrated inside the APU are as follows:

Feature/Function	QXi-6000 APU Graphics
GPU Supported Standards	DirectX 12, SM5.0, OpenGL 4.5, OpenCL 2.0, DirectCompute 11, AMD Mantle, Vulkan
Number of Compute Units	4 or 8
Number of Stream Processors	192 or 512 (64 per CU)
Texture units	12 or 32 (4 per CU)
Compute Performance (peak)	178 or 614GFLOPs (204 or 702GFLOPs Boost)
Memory	32M/64M/128M/256M/512M/1G/2G (shared memory)
Memory Bus	128-bit DDR4-2133
Physical Monitor interfaces	3 DisplayPort connectors, supporting DP++60 DP0 supports QPORT
AMD EyeFinity Technology	Yes, up to 3 displays
Universal Video Decoder (UVD)	UVD 6.0 (4K60 H.265/HEVC, 4K H.264 AVC, H264, VC-1, DivX, MPEG-2, MPEG-4 part 2)
Dual HD Decode	Yes
Hardware Video Encode	Video Codec Engine (VCE) 3.1

The APU is capable of driving three independent monitors with display driver version 13.12 and later. You can utilize the MST feature of the DisplayPort standard to realize the maximum number of monitors. See section 5.5 for more information on MST. The APU is also only able to support a maximum of two legacy (DVI or HDMI) monitors directly. If a VGA device is used, the maximum number of monitors is limited to two. See section 5.4 for more details.

5.2 APU Graphics Memory

The GPU inside the APU utilizes a portion of the main DDR3 for the purposes of video memory. This is termed UMA (Unified Memory Architecture). It is therefore important to understand that a portion of the main memory installed will be reserved for use by the GPU and will not be available for the rest of the system. The amount that is allocated to the GPU is selected in the BIOS Setup Menu. For example, if a total of 4GB of DRAM is installed and 1GB is allocated for video memory only 3GB will appear to a booted OS such as Windows.

Allocating the correct amount of memory is critical to ensuring adequate performance. Allocating too little memory to graphics could limit graphics performance. Utilizing multiple monitors will also increase the requirement for video memory. The techniques used in software for generating and managing graphical data can also affect the amount of required video memory.

So, allocating too little memory for use by the APU GPU can restrict the graphics function. However, care should also be taken not to allocate too much memory to graphics and reduce the memory available to the OS and applications. System performance can be severely adversely affected if too little system memory is available. Some experimentation will therefore be required to decide the total amount of memory that is required and how this is best divided between graphics memory and main memory.

5.3 DisplayPort Interfaces

All the display interfaces on the QXi-6000 are compliant with the DisplayPort 1.2 standard, a royalty-free digital display interface developed by the Video Electronics Standard Association (VESA). DisplayPort (DP) is the latest display interface standard and will gradually replace legacy VGA, DVI and LVDS interfaces on computer systems. Some of the key benefits of DisplayPort are:

- Compact, locking connector well suited to gaming applications
- Four data lanes operating at a fixed link rate of 5.4Gbits per second (Gb), 2.7Gb or 1.62Gbps per lane, giving a total bandwidth of up to 2.16GBytes (GB) per second. Able to operate with 1 to 4 lanes active. See: http://en.wikipedia.org/wiki/DisplayPort
    - On the QXi-6000 only 3 monitors are supported, but for 4K devices it can use all 4 lanes
- Half-duplex bidirectional AUX channel supporting up to 720Mbps communication
- Micro-packet based protocol as opposed to the digitized and serialized raster protocol of other standards such as DVI and HDMI
- Supports the connection of multiple monitors to a single DisplayPort port using multi-stream (MST) technology
- Comprehensive backwards compatibility with VGA, DVI and HDMI monitors using dongle cables (passive adapters in the case of dual mode DP++ sources such as the QXi-6000 connecting to DVI/HDMI). See chapter 5.4 “Limitations with “Legacy” (DVI, HDMI) displays”.

DisplayPort connectors are similar in size and style to HDMI connectors, permitting more connectors in a smaller space than with DVI, crucial on multi-monitor systems. HDMI utilizes signaling similar to DVI (DC-coupled), whereas DP signals are AC-coupled for compatibility with future lower voltage process technologies. HDMI is generally used on domestic video equipment, whereas DP is intended as the replacement for VGA & DVI on computer equipment.

In addition, the signals output from the APU are compatible with what is known as “dual mode” DisplayPort, or DP++. This is a method by which the APU Graphic Controller is able to identify legacy DVI and HDMI monitors and output the signals required by these monitors. This enables a simple “passive” dongle to be used to drive HDMI and DVI monitors. See chapter 5.4 “Limitations with “Legacy” (DVI, HDMI) displays”

The raw DisplayPort signals are brought directly from the APU to the integrated DisplayPort connectors. The APU is capable of driving up to three independent monitors.

5.4 Limitations with "Legacy" (DVI, HDMI) displays

The APU is capable of driving a maximum of two legacy (DVI and/or HDMI) monitors directly through a DP++ connector and a passive adapter implementing DVI or HDMI connectors. If it is required to drive more than two legacy monitors from the APU it is necessary to implement an active converter. This can be done by utilizing an active "dongle" that converts a DP signal into either DVI or HDMI outputs. These active converters make legacy displays appear as DP displays to the QXi-6000. Contact Quixant for more information if you need to drive legacy displays. Quixant is also able to supply a wide range of DP converter dongles, both active and passive.

Note: The DP0 QPORT only support active dongle.

5.5 DisplayPort Multi-Stream Technology (MST)

One of the key features of the DisplayPort 1.2 standard is the ability to drive multiple independent displays from a single DisplayPort output. This is achieved using a feature introduced in the 1.2 standard called Multi-Stream Technology (MST). MST is possible due to the packet-based nature of the communication employed by DisplayPort. Previous display interface technologies have been based on the data from the host device being directly linked to the display data rate. For example, there is usually a clock embedded in the display signals and this is directly linked to the fundamental pixel clock of the display device. In DisplayPort the data is packet based and is transmitted at a fixed rate, as in network interfaces, which allows data to be sent from one output that is intended for several different target destinations. Recipient MST devices are able to extract the packets they need and pass on the other packets to other devices. This is a key capability of DisplayPort that distinguishes it from older display interface standards.

There are clear cabling benefits for gaming applications provided by DP MST. For example, it is typical to have two main displays on the cabinet front door. Using DP SMT this now only requires a single cable from the logic box, which could also drive smaller ancillary displays as well. The number of displays that can be driven is now no longer limited by the number of physical monitor outputs on the logic box itself, so long as they are DisplayPort and MST capable.

It is important to note that MST was only introduced in version 1.2 of the standard. Also, to enable multiple displays to be connected to a single MST capable DP output it is necessary to employ either:

- DisplayPort 1.2 compliant displays that provide an MST daisy-chain output to connect to the next monitor in the chain, or
- An MST hub device that takes in the output from a DP MST source and splits the data out to multiple physical display outputs.

Further information on DisplayPort and MST can be found at http://www.displayport.org

5.6 AMD Infinity and Windows "Single Large Surface"

Microsoft introduced a feature called "Single Large Surface" (SLS) on Windows Vista and later versions of Windows. This is the ability to create "Single Large Surface" of all the displays on a controller arranged for example as 1x2, 1x3, 2x2, etc. The displays defined as forming an SLS can be viewed by 3D or 2D or video applications as a single virtual display and will be treated as one display by that application. AMD has implemented this under their Infinity™ screen configuration.

One advantage of the EyeFinity mode is that the refresh rates of the monitors are kept in sync, which prevents tearing of images that span multiple monitors.

Note: *AMD driver for APU Series-G SOC doesn't support Infinity at this time and could only support 1x2 or 2x1 if it did.*

6 Mass Storage

6.1 CFast™

The QXi-6000 provides two CFast sockets inside the case. CFast is a recent standard for flash media introduced by the Compact Flash Association, the creators of the original Compact Flash (CF) standard. CFast is designed to have the same physical characteristics of CF cards but utilizes the modern SATA interface rather than the legacy PATA parallel interface that is now becoming obsolete. CFast cards are robust, rugged cards that can survive rough handling, which is a key requirement in the gaming industry. CFast sockets are actually more rugged than the old CF sockets, as they do not have the large number of delicate pins that can become damaged.

CFast cards therefore combine the mechanical benefits of CF together with the speed improvements provided by the SATA interface. On the QXi-6000 CFast cards with a transfer rate up to 6Gbit/s are supported.

CFast cards are available that include hardware write protection functionality. This is activated either by a write protect switch on the media itself or on some cards is enabled by a pin on the connector. The QXi-6000 supports both types. For cards that employ a signal on the connector the QXi-6000 supports this on pin PC10. When a card that supports a WP signal on this pin is inserted it is necessary for the associated jumper to be installed to enable writes. Hence if this jumper is not present the card is automatically protected when the card is installed. See the chapter on jumpers for more information.

Just as for Compact Flash, CFast cards are available in a range of capacities and performance grades. It is important to validate that the chosen card has the necessary characteristics for the required application. If a given CFast card appears to be slow try an alternative – the interface on the QXi-6000 is simply 6Gbps SATA, so performance is down to the CFast card being used.

6.2 Serial ATA (SATA)

The QXi-6000 provides two connectors for Serial ATA (SATA) mass storage devices. SATA replaces PATA as the main interface to mass storage peripherals in modern PC systems. SATA devices are software compatible with PATA, but use a different physical interface. SATA is a point-to-point interface and hence each connector supports a single device. The QXi-6000 complies with the SATA 3.0 specification, supporting SATA transfer rates of 1.5Gbps, 3Gbps and 6Gbps.

Most modern mass storage devices are available in the SATA interface format. SATA is the interface of choice for any new application, with faster performance and smaller cables.

In addition to allowing standard SATA devices to be connected using SATA cables, the QXi-6000 enables SATA flash modules to be plugged directly into the SATA sockets. The QXi-6000 provides a jumper for each SATA socket that enables power to be provided on pin 7 of the SATA connector. This enables SATA flash modules that accept power on pin 7 of the SATA connector to be used without the need for a separate power connection.

Note: These power jumpers should **NOT** be installed if the associated SATA socket utilizes a standard SATA device or a standard SATA cable is inserted, as this will short pin 7 to ground.

SATA flash modules are available that incorporate write protect switches, which provides physical protection of the flash media from any attempted write operations. Consult Quixant for more details of the compact, high capacity, high performance SATA flash modules.

The QXi-6000 can also accommodate a standard 2.5inch size SATA HDD or SSD. A tray can be fitted to take the drive and power connections made using J23 (see section 18.3) and data using one of the two available SATA connectors J14 or J18 (see section 18.2).

6.3 Important Considerations When Utilizing Flash-Based Media

It is critically important to be fully aware of the lifetime issues associated with flash-based media and to verify the method of use will not result in any reliability issues. Write endurance is always limited, and this must be investigated thoroughly. Write endurance varies depending on whether the flash chips used are SLC (Single Level Cell) or MLC (Multi Level Cell) type, the flash chip brand, the flash controller used and the percentage utilization of the media capacity. The highest write endurance will normally be achieved with media based on MLC flash devices with a high percentage of unused space (hence available for wear leveling across the device) and a controller that has an efficient wear leveling algorithm.

In applications where the flash media is write protected and writes are not performed then write endurance is not an issue. However, it is still necessary to be aware of the guaranteed lifetime of the data stored on the media being used.

Finally, it is also necessary to consider unexpected power cycle events and the effect this might have on the integrity of data held on the device should power be lost during a write transaction. Resilience to this kind of event is often dependent on the flash controller utilized in the media and the way in which it writes to the flash chips. Be aware that it is possible the data and/or the structure of the media could be irretrievably damaged by loss of power coincident with a media write operation.

7 Sound

7.1 Audio Controller and Codec

The QXi-6000 features a 7.1-channel High Definition Audio controller integrated within the APU that is Unified Audio Architecture (UAA) compatible. Bus Master DMA transfers of up to four multi-channel audio output and input streams are supported, with 32-bit resolution at rates up to 192 kHz.

The on-board HD Audio CODEC (Realtek ALC892) provides 7.1+2 channels of sound output at line level as well as stereo output to the on-board audio amplifier. There is also a stereo audio line level input. All these signals are available through the connectors on either the front or back (see sections 16.2 and 17.8). The CODEC complies with WLP 3.x (Windows Logo Program) and is compatible with WaveRT, DirectSound 3D™ and I3DL2. 16/20/24-bit PCM formats at 44.1/48/96/192kHz are supported.

7.2 Audio Amplifier

The QXi-6000 features a sophisticated Class D stereo amplifier that is capable of directly driving a pair of stereo loudspeakers with good fidelity. The amplifier is connected to dedicated outputs on the CODEC, allowing flexibility of which stereo channel is amplified. As standard the amplifier is optimized to drive 4ohm speakers, with power levels up to 18W RMS per channel. If necessary the QXi-6000 can be supplied configured to drive 8ohm speakers, but at a reduced power output.

CAUTION: The QXi-6000 audio amplifier is of the Bridge Tied Load (BTL) type, which drives the speakers with a differential signal to the speaker + and – terminals. It is important that the speaker – (negative) terminal **IS NOT** connected to ground. Both speaker terminals must be connected individually to the amplifier + and – outputs. In addition, to reduce the effects of EMI it is recommended that the speaker wire used is twisted together. The cable should also be adequate to handle the high output currents – at least 5 Amps.

8 Communications

8.1 COM Ports

The QXi-6000 provides a total of six COM ports (UARTs). The UARTS are compatible with the standard 16550 type, but all have enhancement to provide hardware support for 9-bit data protocols. By default, the UARTs are configured to be standard 16550 compatibles, set to 8-bit mode.

A variety of electrical interface standards are supported across the six available COM ports. COM1& COM2 support ccTalk and iButton, while COM2 also supports JCM (see Table 1 for references). For iButton mode signals JCM_TX & JCM_RX should be joined together externally. It is possible to use the ccTalk1 and ccTalk2 as iButton interface (but need to install JP2 or JP3 in the motherboard – see chapter 14.2 –"Jumper Settings").

COM6 can be configured for either RS232 or RS485. See Table 1 for details of the supported interface standards for each COM port and the signals provided. See Section 14 for details of the jumper settings required to configure each COM port. Note that Linux port names are an example and may be different based on the version of Linux being used, so check the names used using "dmesg | grep tty" and compare the I/O address and IRQ against the table below:

Windows Port Name (1)	Linux Port Name	Serial Interface Standard	I/O	IRQ
COM1	ttyS4	ccTalk, iButton	0x5300	30
COM2	ttyS5	ccTalk, JCM ID003 TTL, iButton	0x5308	30
COM3	ttyS0	RS232	0x5310	30
COM4	ttyS1	RS232	0x5318	30
COM5	ttyS2	RS232	0x5320	30
COM6	ttyS3	RS232 or RS485	0x5328	30

Table 1

Note (1): serial ports in Windows are enumerated through serenum.sys when the multiport driver finds them, so the actual numbering may differ from the one listed in the above table. Serial port names can be changed through advanced tab of device properties in Device Manager.

8.1.1 Linux Serial Ports

To enable the use of serial ports under Linux you may have to set a kernel parameter called acpi_osi to Linux (the capital 'L' is required). This is described on this web page:

https://www.kernel.org/doc/Documentation/kernel-parameters.txt

This can be done changing the /etc/default /grub file. Add the following to the line defining GRUB_CMDLINE_LINUX_DEFAULT:

acpi_osi=Linux nr_uarts=6 8250.nr_uarts=6

Do an update-grub to update the lines in /boot/grub/grub.cfg and reboot the system to have the change take effect.

Note that nr_uarts and 8250.nr_uarts sets the number of serial ports for the system. Some Linux systems don't look beyond four serial ports unless you tell them how many there are.

Moreover, take into consideration that the ttyS number depends also on the total number of allowed serial port (given through the nr_uarts and 8250.nr_uarts grub parameters): the relation between ttyS number and COM number, for a given nr_uarts variable setting, is established through the following formula:

ttyS Number = (COM number + 3) mod nr_uart value.

Example 1: the COM3 serial port, given nr_uarts and 8250.nr_uarts=6, corresponds to the

ttyS number = (3+3) mod 6 = 0, that is ttyS0.

8.2 10/100/1000Mbit Ethernet LAN Interfaces

The QXi-6000 includes two independent Realtek RTL8111G 10/100/1000Mbit network interfaces, each of which supports half-duplex and full duplex communication on standard Cat 5/6 UTP network cabling. The LAN controllers automatically switch between duplex modes and speed modes dependent on the capabilities of the switch/hub to which they are attached. The controllers are IEEE 802.3 and 802.3x standards compatible. For higher network transfer performance and reduced system overhead, it supports Multithread I/O Link Mastering with Read/Write Concurrent transaction.

The devices are programmed with unique MAC addresses at the time of manufacture.

LAN activity and link LEDs are provided on the LAN connector body. See section 16.3.

8.3 USB

The QXi-6000 provides two USB 3.0 and two USB 2.0 ports on the front panel and a second set of four is available via a header on the motherboard internally (see Section 18.11). These ports all support standard USB 2.0 and 1.0 devices.

Note: The QXi-6000 is not designed to provide power for peripheral devices. As the 5V line on the QXi-6000 is generated on-board from the 12V input there is limited amount of current available. High-power USB devices should therefore have their own independent source of power. While +5V power is provided on all the USB sockets (sufficient to power devices such as keyboard & mouse) the total current drawn from the QXi-6000 USB ports must not exceed 400mA.

8.4 I^2C

An I^2C interface is provided that is available to connect to I^2C devices on the J2 connector inside the case. Further details on how to communicate using the I^2C interface is available in the Quixant API documentation. It is recommended that customers consult with Quixant when planning to connect I^2C devices to this interface to confirm correct design implementation. In addition, this I^2C bus is used to interface to several on-board devices available for customer use – see sections 10.4.1 for further information. When adding I^2C devices care must be taken to prevent any clashes in device I^2C addresses. The I^2C bus presented to the connector utilizes 5V signaling levels.

8.5 SPI Interface

An SPI interface is provided on J1 (see Section 17.2) that is available to connect to SPI devices.

Further details on how to communicate using the SPI interface is available in the Quixant SEC Meter API documentation. It is recommended that customers consult with Quixant when planning to connect SPI devices to this interface to confirm correct design implementation. The SPI bus presented to J1 utilizes 5V signaling levels.

Note that the *SPI_SS#* signal on J1 is the “Slave Select” signal and can be left disconnected if there is only one SPI device connected.

9 NVRAM

The QXi-6000 integrates a complete high-performance NVRAM function designed to meet the needs of gaming both in terms of regulatory requirements and performance. The NVRAM also provides a high level of functionality to ease game development and reduce the overhead usually associated with the NVRAM management function.

The QXi-6000 NVRAM is connected to the system PCI Express bus by high performance bridge logic. This logic provides both exceptional raw data performance and also a number of high-level logic functions aimed at reducing the need to transfer data across the bus and hence further increasing performance. The NVRAM function is supported by device drivers and libraries to enable easy integration of the NVRAM function into games under both Windows and Linux operating systems. The NVRAM API is common across OSs and also across other Quixant hardware platforms, enabling easy code portability and re-use across different applications.

9.1 Battery Backed SRAM

The QXi-6000 integrates up to 16MB of 32-bit battery backed SRAM (Static Random Access Memory) located on the PCIe bus and running at PCIe speeds. This SRAM is typically used for storage of game data that needs to be saved in a fast, non-volatile manner. Table **2** shows the SRAM configurations supported on the QXi-6000.

Table 2 – SRAM Configurations

Bank0	Bank1	Configuration	SRAM Size
512k x 16	512k x16	512k x 32bit	2MBytes
1024k x16	**1024k x16**	**1024k x 32bit**	**4Mbytes (default)**
2048k x 16	2048k x 16	2096k x 32bit	8Mbytes
4096k x 16	4096k x 16	4096k x 32bit	16MBytes

Byte, Word and DWord accesses to the SRAM are supported, with the restriction that Word accesses must be “Word aligned” and DWord accesses must be “DWord aligned”. Refer to the “QX Series Software Guide” for more details.

The QXi-6000 architecture supports automatic mirroring of data written to the SRAM. The device driver supports mirroring of data into the two independent SRAM banks as well as the creation of additional logical copies and automatic comparison of data read between the banks, hence fulfilling the needs of various gaming regulatory bodies. Consult QX Series software documentation for details of these features. SRAM requires batteries while MRAM does not.

Note: SRAM bank0 is powered by BAT2 and SRAM bank1 is powered by BAT3.

9.2 NVRAM Configurations

The QXi-6000 integrates up to 16MB of NVRAM using either MRAM (Magneto-Restrictive Random Access Memory) or battery backed SRAM (Static Random Access Memory) devices. NVRAM is typically used for storage of critical game related data that needs to be saved in a fast, non-volatile manner. The QXi-6000 NVRAM comprises up to two physical devices, each with its own independent 16-bit bus to the Quixant PCI Express bridge ensuring high performance, high bandwidth operation of the NVRAM. Writes can be made on a per-byte granularity but operations can also simultaneously access the full 64-bits. NVRAM requires a battery while MRAM does not.

Byte, Word, DWord and block accesses to NVRAM are supported”. Refer to the “QX Series Software Guide” for more details.

The default capacity for each MRAM/SRAM device is 16Mb (2Mbyte), although 32Mb (4MByte) and 64Mb (8MByte) SRAMs are also available. Between 1 and 2 physical devices can be installed, giving a maximum capacity of 16MBytes. SRAM requires batteries while MRAM does not.

9.3 NVRAM Batteries

When the QXi-6000 is supplied installed with SRAM devices as NVRAM the data is maintained when system power is removed by two independent batteries (BAT2 & BAT3). This meets the common jurisdictional requirement for a minimum of two completely independent physical banks. The batteries can be specified to be either non-rechargeable CR2450 primary lithium cells (capacity of 620mAh) or rechargeable (secondary) Lithium Pentoxide cells. SRAM standby current varies dependent on the device being used, but for a 16Mb device is typically 1.5uA. When system power is on the batteries are not drained, increasing lifetime. However, once CR2450 cells become discharged they need to be replaced. This is relatively simple as the batteries are socketed. The QXi-6000 logging processor provides a mechanism to monitor battery voltages and provide advance warning if batteries are in need of replacement.

The QXi-6000 can optionally be supplied installed with Panasonic VL2330 Lithium Pentoxide cells. These secondary cells are designed specifically for long life and do not degrade like NiCd or NiMh cells when continuously connected to charging power. This makes them the only viable option for rechargeable cells for SRAM backup in gaming applications. The disadvantage of these cells is that they have a much lower capacity (50mAh). Given the same scenario as above these cells would last less than 2 years. However, these batteries are guaranteed to regain 100% capacity after the system is switched on for just 24 hours, so SRAM data will always be preserved for this period of time after a system is switched off, with the benefit that the batteries will never require replacement.

If the NVRAM is configured to operate as two physical banks BANK0 is supported by BAT2 and BANK1 by BAT3. When MRAM is installed for the NVRAM there are no batteries for the NVRAM as MRAMs are inherently non-volatile devices. MRAMs provide the same read/write performance as SRAMs.

9.4 NVRAM High Level Data Management Functions

As well as providing high performance direct access to the raw NVRAM data across the PCI Express bus and 64-bit interface to the NRAM devices, the NVRAM logic also provides powerful features that can provide even greater performance and ease the management of NVRAM data in the game. The NVRAM logic can carry out complex operations locally to the NVRAM in hardware, dramatically reducing the requirement for transferring data to and from the game code. This functionality can be broken down into two distinct sections – operating modes and commands that can be executed on the NVRAM data by the NVRAM hardware.

9.5 NVRAM Modes of Operation

9.5.1 Mirror Mode

When this mode is enabled data written to one bank is automatically and simultaneously written to other banks. During read operations the data from each bank is automatically compared and an error flagged if any discrepancy is found.

9.5.2 Auto Write Verify Mode

When this is mode is enabled the NVRAM logic automatically performs a read back of the actual data held in the physical NVRAM devices after every write operation. Any mismatch causes an error to be flagged

9.5.3 Read Cache Mode

When this mode is enabled a local cache of the entire NVRAM data is held at the device driver level. Reads of NVRAM data are satisfied from the local cache and hence read access speeds are significantly improved, as fetches of data across the PCI Express® bus are not required. All writes to NVRAM continue to be executed immediately to the physical NVRAM devices and the cache copy is updated afterwards.

9.6 NVRAM Commands

9.6.1 Bank Copy

This API call (QXT_NVRAM_BLOCK_COPY) will copy from a start address to and ending address in the currently selected bank, placing the copy in the specified bank(s). On systems with more than 2 banks the bank identifiers can be ORed together to copy to multiple banks in the single operation.

9.6.2 Verify Banks

This API call (QXT_NVRAM_BLOCK_COMPARE) will compare data from a starting address to and ending address in the currently selected bank with the same addresses in the specified bank(s). On systems with more than 2 banks the bank identifiers can be ORed together to copy to multiple banks in the single operation.

9.6.3 Generate CRC

This API call (QXT_NVRAM_CALCULATE_CRC) will generate a CRC (CRC-32) of NVRAM data within a specified address range. The polynomial used is:

(X^32 + X^26 + X^23 + X^22 + X^16 + X^12 + X^11 + X^10 + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1)

10 Gaming Logic

The QXi-6000 integrates powerful hardware that enables complex gaming functionality to be achieved easily, at very high speed and with low system overhead. The combination of all these features are necessary to ensure that typical gaming functions are easily achieved and integrated into the customer's game software, and that this low-level functionality provides excellent responsiveness without stealing valuable performance from the overall game. The QXi-6000 architecture achieves all of these goals. The QXi-6000 gaming hardware is supported by comprehensive software drivers with well-defined APIs that ease development of games to leverage the power of the hardware. Quixant Gaming Logic also provides a Quixant platform identification number. In the case of the QXi-6000, the platform ID code is always 0x13. This value can be accessed through the Quixant Core API

The QXi-6000 gaming logic is connected directly to the PCI-Express bus, allowing 32-bit transfers at the full 2.5GHz PCIe speed. Access to all registers, including digital inputs and outputs, is achieved using a direct single PCIe bus transaction, without any communication protocol overhead. This results in extremely fast and efficient operation.

10.1 Digital Inputs

The Quixant gaming architecture is common across multiple physical platforms. This makes it easy to generate games that can operate on multiple physical platforms that satisfy different performance and/or jurisdictional requirements. The software architecture allows for up to 64 digital inputs, but the hardware architecture provides flexibility for what physical interfaces are available on each system. This fact, combined with the powerful per-input range of interrupt generation options, allows digital input functionality to be implemented with virtually zero system overhead.

The QXi-6000 provides 32 general purpose digital inputs that are available for use by the customer. Features of these inputs are:

The input signals are 5V TTL level with 4.7K pull-ups to 5V on each input
Each input can be individually enabled/disabled for generation of interrupts
Each input can generate interrupts on rising, falling or both edges
Each input can be set to screen pulses widths that fall outside pre-determined limits

The digital input registers and associated interrupt control/status registers are accessible through the Quixant Core API. Quixant Core API function calls make access to the digital input ports very easy from user space in the operating system, allowing powerful event driven games to be easily created.

The QXi-6000 also defines several digital inputs for specific purposes. The overall allocation of digital inputs is described in the table below:

DIN0~DIN31	General purpose 5V TTL level digital inputs available on rear panel. 4.7K pull-up to 5V on each input
DIN56~DIN59	Internal customer software option switches. Can be used by customer to configure game options etc. Turning a switch on generates a "0" input level
DIN60	Meter output current sense. This input goes low if the system detects a current outflow from the METER_PWR output exceeding approximately 40mA. This can be used to detect correct meter operation
DIN61	12V_FAIL#. Low when internal detector senses 12V input falls below approximately 10V
DIN63	PSU_ACFAIL# input from external PSU. Can be used to generate early indication to game software of imminent system failure

10.2 Digital Input Bounce Filter

The QXi-6000 digital inputs are designed to be able to work with high speed input signals. However, it is often also necessary to connect at least some of the digital inputs to simple switches or push-buttons. These are electromechanical devices that generate imperfect output signals - not simple single transitions but rather multiple "noisy" edges before finally settling. This can be an issue particularly when utilizing interrupts to notify changes in input state, as such inputs can generate unwanted multiple interrupts.

The problem of switch bounce can often be resolved by the simple use of appropriate value capacitors on the relevant inputs. However, the QXi-6000 provides the facility to activate an input filter on any of the available inputs.

Each input has its own filter block which can be enabled if required, the default state being disabled. All the filter blocks are driven from a single programmable clock source (de-bounce clock). Each filter has eight states. The input is sampled on each clock and if the state of the input is unchanged compared to the last clock cycle the particular filter will advance to the next state. However, if the input state has changed the filter state will be reset back to zero. Once the filter state reaches 7 (meaning that the input has remained unchanged for seven successive clock cycles) the output will then be free to change state at the next clock. If the input state changes the filter state will be reset back to zero again. The effect of this is that a change of state of an input is registered immediately, but it must remain stable for a pre-determined period before a further change in input is recognized.

The application software is able to set the frequency of the common de-bounce clock used by all the individual input filters and hence is able to fine tune the filters to match the characteristics of the input device.

10.3 Digital Outputs

As for the digital inputs, the Quixant gaming software architecture provides for up to 64 digital outputs. Each system in the Quixant range can have different types and numbers of digital outputs available, but from a software perspective all Quixant systems are compatible with each other.

The QXi-6000 provides a total of 32 digital outputs. These are as follows:

DOUT0~DOUT23	Open drain outputs. Max 350mA, 50V each. Overvoltage and overcurrent protected. When on, these outputs are connected to ground by the QXi-6000, allowing power to flow from an external power supply through an external load into the digital output and then to ground. Feedback is available for those Digital outputs, allowing the user to detect open circuits and shortcuts.
DOUT24~DOUT27	Open drain outputs. Max 3Amps, 50V each. No overcurrent or over voltage protection. When on, these outputs are connected to ground by the QXi-6000, allowing power to flow from an external power supply through an external load into the digital output and then to ground.
DOUT28~DOUT31	5V TTL level push-pull outputs

Digital outputs may be set/reset by performing I/O writes to the digital output registers by using the appropriate Quixant Core API function calls.

10.4 On-Board I²C Devices

The QXi-6000 provides several on-board I²C devices that can be utilized by game code. Information on how to communicate using the I²C interface is available in the Quixant API documentation. This should be consulted in combination with the datasheets for the individual devices. Each device has its own address. It is important to make sure no clashes in I²C addresses can occur.

10.4.1 I²C EEPROM (if installed)

The QXi-6000 includes an 8-pin DIP socket (See section 19.5) that can be used to install an I²C serial EEPROM of the 24C0x variety. This EEPROM can then be used by customer's game software to store information in a non-volatile form.

Supported devices are 24C02 (2Kb) devices that are designed to operate from a 3.3V supply.

The address range occupied by the EEPROM will vary depending on the type installed as follows:

24C02 1010010b

Caution – devices manufactured by Microchip should NOT be used, as devices from this manufacturer do not support address selects on pins 1, 2 & 3.

The EEPROM can only be written when jumper JP13 is installed – when this jumper is removed the device is write protected.

Consult Quixant software documentation for information on using the I²C bus on the QXi-6000 together with EEPROM chip documentation.

10.4.2 I²C Surface Mounted EEPROM U21 (if installed)

The QXi-6000 can include a permanently mounted I2C EEPROM device of the 24C0x variety. This EEPROM U21) can then be used by customer's game software to store information in a non-volatile form. The address pins of the device are connected A2=0, A1=1, A0=0. If U21 is installed (SMD type) devices cannot be installed in socket U22, as they share the same address configuration. JP13 is used to control the WP# pin of U22 (see section 14.2).

10.4.3 I²C Unique Serial Number

The QXi-6000 includes two silicon devices that provides a guaranteed unique 64-bit serial number.

The first device is a DS28CM00 from Maxim and resides on the same I²C bus as the serial EEPROM described above. This device is located at I²C address 1010000b.

The second device is a DS28CN01 from Maxim and is also on the I²C bus. It includes a SHA-1 engine for generating SHA-1 MACs.

Consult Quixant software documentation for information on using the I²C bus on the QXi-6000 together with the Maxim documentation for the DS28CM00 and DS28CN01 chips.

10.5 Meter Power

The QXi-6000 provides a special +12V power output (METER_PWR) designed to supply power to electromechanical meters. The current drain from this output pin is sensed and if it exceeds a threshold of approximately 40mA the state of DIN60 will become logic 0. METER_PWR can be used to provide power to the positive side of a bank of electromechanical meters. The negative side of the meters should be connected to individual digital outputs. By strobing the outputs to the meters individually and sensing the state of DIN60 (after an appropriate delay or just before the DOUT is turned back off) the flow of current through the meters can be verified. Hence a removed or faulty meter can be detected. It should be remembered that electromechanical meters are inductive devices, and so it takes time for the current through them to rise to the final value. This is important to remember when writing software to detect correct operation.

Please note this signal is present just for backward compatibility with other Quixant platforms (like QXi-300 and QXi-4000). For new design, the user can make usage of the faster direct feedback from most of the Digital Output (see section 10.3).

11 Security

11.1 Hardware Decryption/Encryption Engine

The Advanced Encryption Standard (AES) is a security encryption algorithm that has been approved by the National Institute of Standards and Technology (NIST) as the latest symmetric encryption algorithm for the protection of sensitive information by the U.S. Government.

The QXi-6000 features hardware accelerated AES decryption & encryption engine with storage for up to sixteen separate 128-bit/256-bit decryption keys. Customers can pass encrypted data through this engine and receive a decrypted output or vice-versa.

Please refer to the Quixant SECS API specification for more details on how to use the AES decryption engine.

11.2 Logging Processor

The QXi-6000 provides a battery-backed processor that takes care of various security roles, even if system power is not present. The functions performed by the logging processor are:

Monitoring and logging of 13 intrusion switch inputs
Monitoring of the voltage levels of the three batteries at a configurable time interval
Provide a second Real Time Clock (RTC) function that is used to time stamp events and can also be used to check against the standard PC RTC
Provide a watchdog timer (WDT) function that will generate a hardware reset to the system if a “refresh” operation of the WDT is not performed within a pre-set time limit. The user software can also decide to receive a software interrupt instead of a hardware reset.
Provide a “start-up” watchdog: if the system is not running properly within a predefined time after power on the machine is re-started
Provide a mechanism to enable the system to be started up automatically at a prescribed time each day if required
Logs any event relevant to the physical security of the platform, such as power on, power off, watchdog, etc. Event log is 64 entries deep and is protected against attempts to flood the event queue to mask events.

Intrusion switch inputs and battery level are reported to the system in one of the following ways:

Notified by an event/callback (recommended)As an entry in the event log
By polling the status (not recommended)

Note: The Security Processor is powered by battery BAT1, which also powers the RTC inside the PC subsystem.

11.3 Conventional Intrusion Switches

The QXi-6000 is able to continuously monitor the condition of sixteen intrusion inputs, even when system power is removed. Any change of state of an intrusion input is automatically logged with a date/time stamp. Four intrusion inputs are used internally inside the QXi-6000, while four are available externally to allow monitoring of the gaming cabinet. The sixteen intrusion inputs are as follows:

INTR0-1	Case intrusion switch
INTR-7	Lid intrusion detector (see section 18.3)
INTR2-INTR5	External intrusion inputs available for customer use (see section 17.1)

Some intrusions are used for detection of hardware configuration changes:

INTR10	User switch (see section 18.16)
INTR11	BIOS protection change detection (see section 14.2)
INTR12	FPGA protection change detection (see section 14.2)
INTR13	CFAST1 protection change detection (see section 14.2)
INTR14	CFAST2 protection change detection (see section 14.2)

Implementing an intrusion input externally is simply a matter of connecting a switch between an intrusion input and ground. It should be noted that the intrusion inputs are very high impedance, and so great care should be taken to avoid sources of electrical noise when routing external cabling. If noise is encountered, it could cause false intrusion events to be recorded. If noise is encountered either re-route the cabling to avoid the source of noise or use a screened cable – center core for the intrusion signal and ground as the screen/shield. **Do not** connect any other components (such as capacitors, pull-up resistors etc.) to the intrusion input signals as this will have a severe effect on the battery life of the logging processor.

It is acceptable to wire several switches in series between an intrusion input and ground (any switch opened causes the input to be high) or in parallel between a single intrusion input and ground (any switch closed causes input to be low). In this way the system is able to monitor more than four physical external switches.

Each intrusion switch can be individually configured to be normally open (N.O.), normally closed (N.C.), optical or not used. The logging processor will report the status of the switch according to this definition.

11.4 Opto-Coupled Intrusion Switches

The QXi-6000 provides the capability to drive and monitor opto-coupled intrusion switches. Opto coupled switches comprise an LED emitter and a photo detector directly aligned with each other. The beam is interrupted by the presence of an optically opaque movable object that is usually mounted on a door assembly. When the object is removed light from the LED is received by the detector. To prevent tampering it is usual to apply an AC drive waveform to the LED emitter and compare the received signal with it. In this way a forced open circuit or short circuit of the detector will still be detected as a tamper.

Implementation of an optical intrusion switch involves connecting the LED to a drive signal and the receiver to an intrusion input. The emitter LED should be connected as follows:

1) Connect anode of LED via a series resistor “RLED” to a suitable power supply, voltage “VS“
2) Connect cathode of LED to QXi-6000 output “OPTO_DRV”

The value of R should be selected based on the following equation:

$$RLED = \frac{(VS - VF)}{IF} - 33$$

Where IF is the recommended operating current of the LED emitter and VF is the forward voltage of the LED emitter at current IF, as defined in the device datasheet.

Where more than one opto-switch is being used each emitter should have its own individual resistor in series between the anode and the supply voltage. The purpose of RLED is to limit the current flow through the LED to match the specification. Make sure the power rating of resistor RLED is adequate to dissipate a power of P= $(IF)^2$ * RLED, including a suitable safety margin.

The detector, which will usually be a photo transistor or photo diode, should be connected between one of the intrusion inputs and GND. When light shines on the detector it will cause current to flow, generating a low level voltage on the intrusion input line.

When using opto-coupled intrusion switches it is important to understand that they will not function when system power is off. They will therefore usually be used in addition to conventional switches which will provide protection even when power is off.

It is necessary for the game software to define what type of switch is connected to the intrusion inputs for correct operation. Refer to Quixant software documentation for more details.

11.5 Real-time Clocks

The QXi-6000 provides two real-time clocks (RTCs), both of which are battery backed. These are used for keeping a reference of both time elapsed and logging of events by the security processor. These features may be implemented to allow for time-disabled game creation. Events are stored in the security processor with a time stamp, even when the main system power is off so, for example, an intrusion event is detected.

It is recommended that game software maintain and compare both the system (PC) RTC and also the RTC inside the logging processor. If the time/date of the two RTCs is significantly different this should flag an action of some kind.

12 System Environment Control & Monitoring

12.1 Temperature Monitoring

The QXi-6000 continuously monitors the temperature of both APU (through the APU SB-TSI interface) and discrete GPU (using the GPU's thermal diode). This temperature can be monitored by the game software using a Quixant Core API, with the ability to set up alarms etc. if certain thermal limits are exceeded. Consult Quixant Core API information for more details.

13 Power

13.1 Power Supply

The QXi-6000 is designed to run under +12V nominal power input, and uses high-quality switching supplies internally to generate all other necessary internal voltages for the logic. Typically, the QXi-6000 will be powered from the same power supply used to power the rest of the game cabinet peripherals, saving the cost of a dedicated PSU for the logic box. Quixant advises against the use of ATX-style PSUs as they are generally of inferior quality and sometimes will not work properly without a load on their 3.3V and 5V outputs. Quixant recommends the use of a single, high reliability, high efficiency 12V PSU for most system designs.

While steady-state power draw of the system is very low compared to standard computers, certain conditions can cause temporary spikes in demand, such as during the power-on process. Quixant therefore recommends use of a power supply that is capable of delivering a minimum of 4A (50W) to the QXi-6000 on the 12V rail. Any peripheral power requirements need to be added to this.

Note: The QXi-6000 generates sufficient 5V power internally for a notebook style 2.5" HDD and a total of 1A for other devices, such as USB, CFast cards, DOMs etc. Where peripherals require 5V power this should be provided by the external main PSU.

13.2 AC Power Failure Warning

The QXi-6000 provides an input (PSU_ACFAIL#) that can be connected to an "ACFAIL" output on the main PSU supplying the QXi-6000 or cabinet. This can provide advance warning of an imminent power failure, allowing vital game data to be saved to battery backed memory before the main system power fails. The state of this input signal is represented by DIN63. PSU_ACFAIL# is a 5V TTL level input signal with a 10K pull-up to 5V. The time available between this input becoming active and the system failing will depend on the hold-up specification of the PSU being used. The longer the hold-up time the more time will be available to perform critical functions. The response time is also dependent on the OS being used.

13.3 12V Input Level Warning

The QXi-6000 provides a warning should the level of the DC input voltage drop below approximately 10V. This can also be used as a means of detecting imminent power failure, though use of an AC power failure is the preferable solution for the earliest possible warning. DIN61 will go low when a low voltage is detected.

13.4 Batteries

The on-board flash and EEPROM memory is non-volatile and does not require supply of power for it to maintain the integrity of the data stored within it. The on-board SRAM, however, requires appropriate battery backup power at all times to maintain its contents when main power is not present. Additionally, certain logging functions such as intrusion events must be monitored when the main power is off.

The QXi-6000 has 3 high-power density Lithium CR2450 3V batteries. These are non-rechargeable; however, they are only utilized when the main system power is off. All the time power is provided on the +12V input the batteries will not be used. When new, each battery has a capacity of 620mAh.

BAT1 powers the system RTC and also the security/logging processor functions. With continuous use, this battery has a typical lifespan of about 5 years.

BAT2 and BAT3 power the two on-board SRAM devices. SRAM standby current varies depending on the model of SRAM device being used, but for a 16Mb device is typically 1.5uA. When using CR2450 cells this provides a calculated battery lifetime in excess of 16 years, assuming the system is never switched on. These are not present if MRAM is being used for NVRAM.

The QXi-6000 can optionally be supplied installed with Panasonic VL2330 Lithium Pentoxide cells. These secondary cells are designed specifically for long life and do not degrade like NiCad or NiMh cells when continuously connected to charging power. This makes them the only viable option for rechargeable cells for SRAM backup in gaming applications. The disadvantage of these cells is that they have a much lower capacity (50mAh). Given the same scenario as above these cells would last less than 4 years. However, these batteries are guaranteed to regain 100% capacity after the system is switched on for just 24 hours, so SRAM data will always be preserved for this period of time after a system is switched of, with the benefit that the batteries will never require replacement.

It should be noted that the action of reading the battery voltages places a higher than usual load on the batteries, so this function should not be performed more often than necessary.

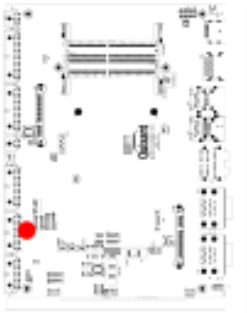
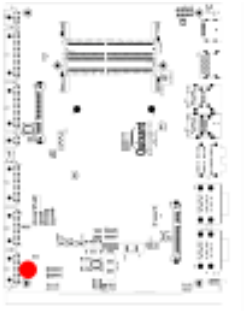
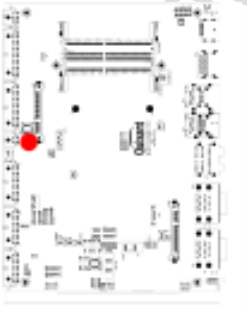
Note: It is the responsibility of the end-user to ensure a regular battery maintenance schedule. The batteries should be replaced well in advance of failure to ensure continuous reliable system operation. If the batteries are replaced when the system is in soft-off power mode, or when it is in main power on mode, the contents of the SRAM will remain intact.

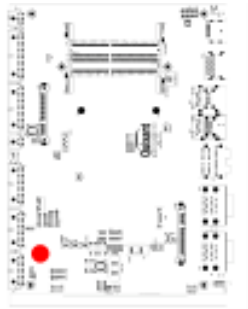
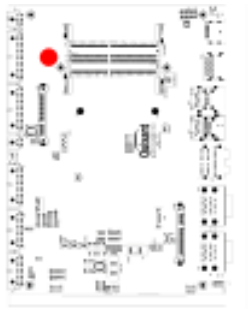
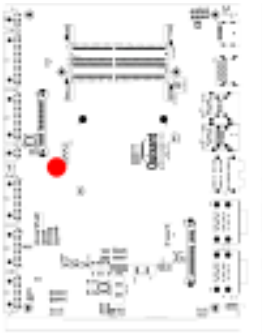
14 Jumper Settings

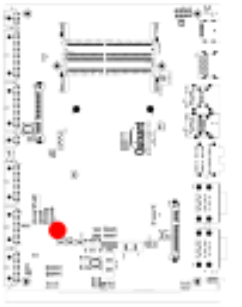
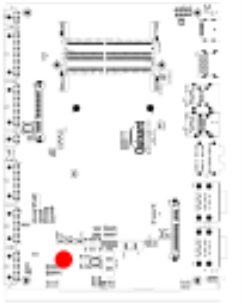
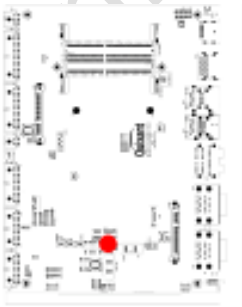
14.1 Jumper/Connector Positions

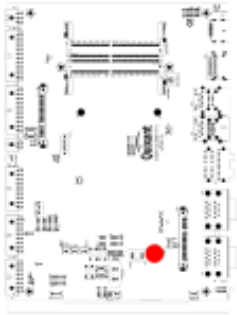
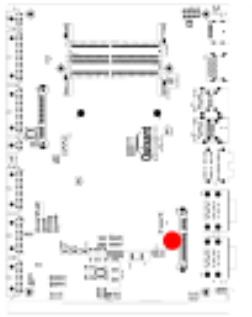
Refer to the PCB layout drawing in section 3.4 for position information on QXi-6000 jumpers. All jumpers use 2.0mm jumper blocks.

14.2 Jumper Settings

Position	Jumper	Function
	JP1	DIO Expansion • Pin 1-2 Disable (default) • Pin 2-3 Enable J8 connector is used for DIO expansion Inserting the jumper on JP1(position 2-3), we can extend the number of Digital I/O and J8 to become 64bit DIO.
	JP2	Increase CCTALK1 pull-high capability • Installed – CCTALK1 pull-high capability enable (default) • Not Installed – CCTALK1 pull-high capability disable
	JP3	Increase CCTALK2 pull-high capability • Installed – CCTALK2 pull-high capability enable (default) • Not Installed – CCTALK2 pull-high capability disable
	JP4	COM6 RS485 mode select: • Installed 1-2 – COM6 functions in RS485 • Not installed – COM6 functions in R232 (default)
	JP5	Write protection of CFast1 (J9): • Installed – write protection of CFast1 disabled • Not Installed – write protection of CFast1 enabled (default)
	JP6 of U2 JP13 of U22	Enable the FPGA to generate Write protection of EEPROM: • Installed – write protection of EEPROM disable • Not Installed – write protection of EEPROM enable (default)

	JP7	CCTALK2 mode select. • Installed – ccTalk mode enabled (default) • Not Installed – JCM mode Physical connections should only be made to EITHER the ccTalk or JCM signals, not both at the same time
	JP8	Digital Output short fault protection current limited mode • Installed 1-2 –Short fault protection enable (default) • Installed 2-3 – Short fault protection disable
	JP9	J14 Pin7 SATA Power: • Installed – 5V power provided on J14 SATA connector pin 7 • Not Installed – Standard SATA device (default) **Warning** – Only install jumper when using a SATA DOM that takes power from pin 7 of the SATA connector. This will supply power to the DOM without the need for a separate power cable. DO NOT install jumper when connecting either a SATA cable or a standard SATA device to J14 as this will cause the power on pin 7 to be shorted to ground.
	JP10	JP10 Clear CMOS Memory • Enables the power for the system Pin 1-2 Normal operation (default) • Pin 2-3 RTC power shorted to ground It should be noted that the QXi-6000 requires RTC power to be present in order to enable the system to boot. Temporarily moving the jumper to 2-3 while all external power is removed will clear the CMOS memory

	JP11	Auto power-on select (AT/ATX mode): • Installed – AT Mode: System boots as soon as +12V input applied (default) • Not Installed – ATX Mode: System only boots when signal PWR_BTN# is grounded
	JP12	Enables the Logging Processor to generate Watch dog timer for the system: • Installed – Watch dog disable • Not Installed – Watch dog enable (default)
	JP14	Enable the CPLD to generate Write protection of BIOS: • Installed – write protection of BIOS disable • Not Installed – write protection of BIOS enable (default) **Note:** BIOS settings cannot be changed while write protected
	JP15 & JP23	Enable the Logging processor to generate Write protection of logging processor: • Installed – write protection disable • Not Installed – write protection enable (default)

	JP20	Write protection of FPGA (U22) • Installed – disables write protection of FPGA (default) • Not Installed – enables write protection of FPGA
	JP21	J18 Pin7 SATA Power: • Installed – 5V power provided on J18 SATA connector pin 7 • Not Installed – Standard SATA device (default) **Warning** – Only install jumper when using a SATA DOM that takes power from pin 7 of the SATA connector. This will supply power to the DOM without the need for a separate power cable. DO NOT install jumper when connecting either a SATA cable or a standard SATA device to J18 as this will cause the power on pin 7 to be shorted to ground.
	JP22	Write protection of CFast2 (J21): • Installed – write protection of CFast2 disabled • Not Installed – write protection of CFast2 enabled (default)

15 Status LEDs & User Switches

15.1 Front Panel LEDs

The QXi-6000 has four stacked LEDs on the front panel. The function of these LEDs is as follows:

	Label	Description
Top	+12V/ IN	Lit when +12V input power is present
	PWR ON	Lit when system power is switched on and +3.3V is stable
	SYS ON	Lights when the system is on or in “soft power off” mode
Bottom	DISK ACT	Lights when there is activity on the SATA interface

15.2 User Switches

SW4 is a bank of four switches that is available to the user, the values of which are readable in software through DIN 56 to 59.

16 Front Panel Connectors

16.1 P5, P6 – COM3, COM4, COM5, COM6 RS232

The COM3, COM4, COM5 & COM6 RS232 interfaces are available on two standard dual DB9P connectors, P5 & P6. The connector assignments are as follows:

- COM3 = P5 Bottom connector (nearest PCB)
- COM4 = P5 Top connector
- COM5 = P6 Bottom connector (nearest PCB)
- COM6 = P6 Top connector

The pin-out of the DB9P connectors follows the standard adopted for the IBM PC, with only 4 data signals supported. The DSR, CD, DTR, and RI signals are not present on the connector.

Connector P4, P5 – DB9P – RS232

GND	5		
NC	4	9	**NC**
TXD	3	8	**CTS**
RXD	2	7	**RTS**
NC	1	6	**NC**

16.2 P7 – Audio Line Level Interface

P7 contains six 3.5mm stereo jack sockets arranged vertically as two sets of three jacks. Internal to the QXi-6000 the line level output signals are also routed to the on-board 7.1 HD audio codec.

Color	Label	Description	Description	Label	Color
Orange	C/SUB	Center speaker & Sub-woofer	Stereo line input	IN	Blue
Black	SIDE	Side left & right speakers	Stereo line output	OUT	Green
Grey	REAR	Rear left & right speakers	Stereo microphone input	MIC	Pink

16.3 P3, P4 – LAN + USB Ports

Connector P3 & P4 each contain an RJ45 LAN jack interface and 2 USB Type A connectors. Connector P4 is LAN controller #1, P3 is LAN controller #2. The Ethernet interfaces support speeds of 10/100/1000Mb/s using CAT5/CAT6 cabling. P4 is with USB 3.0 and P3 is with USB2.0

The LAN connectors each have two status LEDs. A yellow LED indicates a LAN link established at 10Mbps, orange indicates 100Mbps and green indicates a connection at 1Gbps. Flashing LEDs indicate LAN traffic/activity.

Note: The USB connectors provide +5V power in accordance with the USB specification. However, the QXi-6000 is not intended to supply significant power to peripherals. Total power drawn through all the USB connectors should be limited to less than 400mA. This is normally sufficient to power simple devices, such as keyboard, mouse, and flash memory stick but for other devices separate power arrangements should be made.

16.4 DP0 to DP2 – Integrated Graphics DisplayPort Connectors

DP0, DP1, and DP2 are the three display outputs of Graphics Processing Unit (GPU) of the QXi-6000 Motherboard.

All DisplayPort connections are DisplayPort 1.2 compliant. Quixant recommends using DisplayPort cables with locking latches to prevent unintentional disconnection.

The display interface signals coming from the QXi-6000 MB are the raw DisplayPort signals. The APU is capable of interfacing to DisplayPort, DVI, HDMI and VGA monitors using differing combinations of interface circuitry in the cables or via a convertor device / dongle. To retain maximum flexibility all DP connections using the dual mode interface, normally referred to as “DP++”. DP++ connectors allow DVI and HDMI displays to be connected using simple passive adapter cables, with certain limitations. Refer to the section 5.4 in this manual for more information about the limitations for those displays.

DP0 is a QPort™ as defined by Quixant. QPort-capable monitors can provide USB connection capability through the QPort cable. **Note:** To use this port with a legacy interface (DVI or HDMI) you ***must*** use an Active adapter. VGA adapters are always Active adapters.

17 Rear Panel Connectors

The connectors arranged along the rear panel of the QXi-6000 provide access to the non-PC standard signals specific to gaming.

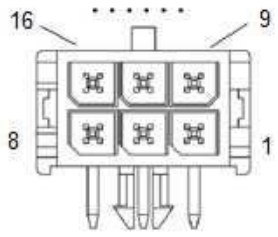
17.1 Rear Panel Connector Signal Descriptions

The table below gives the names and descriptions of the signals accessed through connectors on the rear panel of the QXi-6000.

Signal Name	Type	Description
COM6_485+, COM6_485-	RS485 I/O	RS485 data pair when COM6 configured for RS485 operation
JCM_TX	TTL Output	ID003 compatible signals when COM2 configured for JCM mode. Join together for iButton interface
JCM_RX	TTL Input	
ccTalk1	ccTalk	ccTalk compatible signals for COM1
ccTalk2	ccTalk	ccTalk compatible signals for COM2
SPI_SS#	TTL Output	SPI Slave Select (5V signal)
SPI_CLK	TTL Output	SPI Clock signal (5V signal)
SPI_MISO	TTL Input	SPI Data Input (5V signal)
SPI_MOSI	TTL Output	SPI Data Output (5V signal)
METER_PWR	Power Out	+12V supply output to electromechanical meters. See section 10.4 for more details
I2C_DAT	TTL I/O	Interface to external I2C devices. 5V signal
I2C_CLK	TTL Output	
PSU_ACFAIL#	TTL Input	Input from mains PSU to indicate power failure
OPTO_DRV	Output O/C	Output to drive cabinet LED emitter
INTR_2 ~ INTR_5	Input with P/U	Intrusion switch inputs. Connect via switches to GND
DIN0 ~ DIN31	Input with P/U	Digital inputs. 5V signal
DOUT0 ~ DOUT23	O/C Output	40V, 350mA Open Collector outputs
DOUT24 ~ DOUT27	O/C Output	40V, 3Amp Open Collector outputs
DOUT28 ~ DOUT31	TTL Output	5V TTL digital outputs

17.2 Rear Panel Connector J1

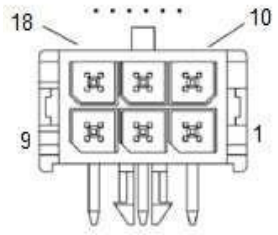
The tables below list the pinouts of the connectors on the QXi-6000 rear panel.



Molex Microfit type 43045-1600

Signal	Pin	Pin	Signal
COM6_485-	**9**	**1**	COM6_485+
GND	**10**	**2**	GND
JCM_RX	**11**	**3**	JCM_TX
GND	**12**	**4**	CCTALK1
GND	**13**	**5**	CCTALK2
SPI_SS#	**14**	**6**	SPI_CLK
SPI_MISO	**15**	**7**	SPI_MOSI
GND	**16**	**8**	METER_PWR

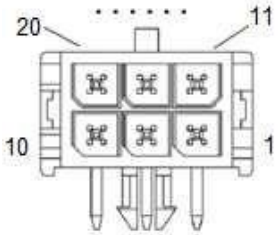
17.3 Rear Panel Connector J2



Molex Microfit type 43045-1800

Signal	Pin	Pin	Signal
I2C_DAT	**10**	**1**	I2C_CLK
GND	**11**	**2**	PSU_ACFAIL#
OPTO_DRV	**12**	**3**	INTR_2
GND	**13**	**4**	INTR_3
GND	**14**	**5**	INTR_4
GND	**15**	**6**	INTR_5
DIN1	**16**	**7**	DIN0
DIN3	**17**	**8**	DIN2
DIN5	**18**	**9**	DIN4

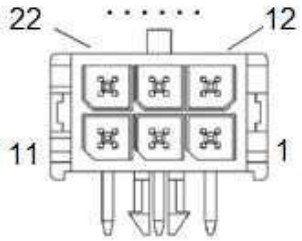
17.4 Rear Panel Connector J3



Molex Microfit type 43045-2000

Signal	Pin	Pin	Signal
GND	**11**	**1**	GND
DIN7	**12**	**2**	DIN6
DIN9	**13**	**3**	DIN8
DIN11	**14**	**4**	DIN10
DIN13	**15**	**5**	DIN12
DIN15	**16**	**6**	DIN14
DIN17	**17**	**7**	DIN16
DIN19	**18**	**8**	DIN18
DIN21	**19**	**9**	DIN20
GND	**20**	**10**	GND

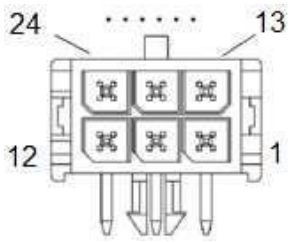
17.5 Rear Panel Connector J4



Molex Microfit type 43045-2200

Signal	Pin	Pin	Signal
GND	**12**	**1**	GND
DIN23	**13**	**2**	DIN22
DIN25	**14**	**3**	DIN24
DIN27	**15**	**4**	DIN26
DIN29	**16**	**5**	DIN28
DIN31	**17**	**6**	DIN30
GND	**18**	**7**	GND
DOUT1	**19**	**8**	DOUT0
DOUT3	**20**	**9**	DOUT2
DOUT5	**21**	**10**	DOUT4
DOUT7	**22**	**11**	DOUT6

17.6 Rear Panel Connector J5



Molex Microfit type 43045-2400

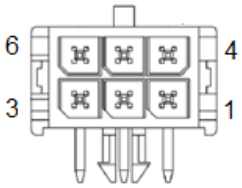
Signal	Pin	Pin	Signal
DOUT9	**13**	**1**	DOUT8
DOUT11	**14**	**2**	DOUT10
DOUT13	**15**	**3**	DOUT12
DOUT15	**16**	**4**	DOUT14
DOUT17	**17**	**5**	DOUT16
DOUT19	**18**	**6**	DOUT18
DOUT21	**19**	**7**	DOUT20
DOUT23	**20**	**8**	DOUT22
DOUT25	**21**	**9**	DOUT24
DOUT27	**22**	**10**	DOUT26
DOUT29	**23**	**11**	DOUT28
DOUT31	**24**	**12**	DOUT30

17.7 J6 – Power Input (PWR)

The QXi-6000 operates from a nominal 12/24 V input supply. All other voltages required internally by the QXi-6000 are generated derived from this input. Total current into this connector depends on the components installed and system activity, but it is suggested that heavy duty cables, each core capable of carrying at least 5Amps, be used to wire to the power supply.

The QXi-6000 is able to operate over quite a wide range of input voltages, making it resistant to power glitches and eliminating the need for a power supply dedicated to the processor unit. Usually the QXi-6000 can utilize the main cabinet PSU, eliminating the need for a separate “ATX” style PSU.

Warning: It is strongly recommended that all power and GND pins are wired using heavy gauge cable, each core capable of carrying at least 5Amps, directly back to the system PSU. It should be remembered that the GND connections carry the current conducted through all the digital outputs in addition to the power used by the QXi-6000 itself.



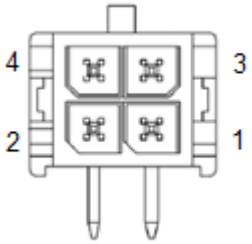
Molex Microfit type 43045-0600

Signal	Pin	Pin	Signal
+12V Input	**4**	**1**	GND
+12V Input	**5**	**2**	GND
GND	**6**	**3**	GND

17.8 J23 – Stereo Amplified Speaker Output

The QXi-6000 Motherboard integrates a high quality Class D stereo audio amplifier capable of driving 18W RMS/channel (36W total) into 4ohm speakers. The connector used for the speaker outputs is a Molex Microfit type 43045-0400.

Warning: The speaker outputs are differential signals (BTL type). Under no circumstances should any output be connected to ground. Use cable rated at ≥ 4 Amps and twist together as a pair to reduce EMI.



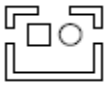
Molex Microfit type 43045-0400

Signal	Pin	Pin	Signal
RIGHT-	**3**	1	RIGHT+
LEFT-	**4**	**2**	LEFT+

18 Internal Connectors & Headers

18.1 BAT1, BAT2, and BAT3 – RTC & SRAM Battery Power

These are the connectors for the battery pack used by the Logging Processor & RTC, as well as the SRAM when power is not coming from an external power supply.

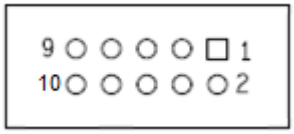


JST B2B-PH-K-S

Pin	Function
1	Vcc (+3.3V nominal)
2	GND

18.2 J8 – Digital I/O Expansion

Contact Quixant for more details. Reserved - do not use this header.



Connector is 2x5 2.0mm" pin box header

Signal	Pin	Pin	Signal
Vcc (+5V)	**1**	**2**	SERO_LD#
GND	**3**	**4**	SEROUT
SERIN	**5**	**6**	SERFB
SERIN_CLK	**7**	**8**	SEROUT_CLK
SERI_LD	**9**	**10**	SER_CLR#

18.3 J9 CFast1, J21 CFast2 – CFast™ Sockets

The QXi-6000 provides two vertical CFast sockets that support devices compatible with the Compact Flash Association standard (www.compactflash.org), including support for Ultra DMA transfers introduced from revision 3.0. See section 6.1 for more detailed information on CFast storage devices.

18.4 J11 – I²C Header

Header J11 provides an internal interface for the I²C bus of the QXi-6000.

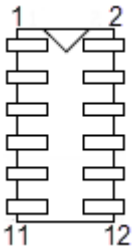
Note: The I²C connector provides +5V power. However, the QXi-6000 is not intended to supply significant power to peripherals. Total power drawn through this and all the USB connectors should be limited to less than 500mA. This is normally sufficient to power simple devices, such as keyboard, mouse, and flash memory stick but for other devices separate power arrangements should be made.

JST B4B-PH-K-S

Pin	Function
1	+5V
2	I2CDATA
3	I2CCLK
4	GND

18.5 J12 – Internal 4-port USB Header

Header J12 provides access to four internal USB2 ports. The connector type is a 2mm pitch dual-row 12-pin header and the pin-out is shown below:

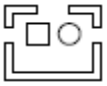


Pin	Function	Pin	Function
1	+5V	2	USBN3
3	USBN1	4	USBP3
5	USBP1	6	+5V
7	GND	8	USBN4
9	USBN2	10	USBP4
11	USBP2	12	GND

Note: The USB connectors provide +5V power in accordance with the USB specification, as does the I²C connector. However, the QXi-6000 is not intended to supply significant power to peripherals. Total power drawn through all the USB connectors should be limited to less than 500mA. This is normally sufficient to power simple devices, such as keyboard, mouse, and flash memory stick but for other devices separate power arrangements should be made.

18.6 J13 & J19 – External SATA DOM Power

This connector can provide power for SATA DOM modules that need power to be provided externally.



JST B2B-PH-K-S

Pin	Function
1	+5V
2	GND

18.7 J14, J18 – Serial ATA (SATA) sockets

The QXi-6000 provides two vertical connectors for Serial ATA (SATA) mass storage devices. SATA replaces PATA as the main interface to mass storage peripherals in modern PC systems. SATA devices are software compatible with PATA, but use a different physical interface. SATA is a point-to-point interface and hence each connector supports a single device. The QXi-6000 complies with the SATA 3.0 specification, supporting SATA transfer rates of 1.5Gbps, 3Gbps and 6Gbps.

Most modern mass storage devices are available in the SATA interface format. SATA is the interface of choice for any new application, with faster performance and smaller cables.

In addition to allowing standard SATA devices to be connected using SATA cables, the QXi-6000 enables SATA flash modules to be plugged directly into the SATA sockets. The QXi-6000 provides a jumper for each SATA socket that enables power to be provided on pin 7 of the SATA connector. This enables SATA flash modules that accept power on pin 7 of the SATA connector to be used without the need for a separate power connection.

Note: These power jumpers should **NOT** be installed if the associated SATA socket utilizes a standard SATA device or a standard SATA cable is inserted, as this will short pin 7 to ground.

SATA flash modules are available that incorporate write protect switches, which provides physical protection of the flash media from any attempted write operations. Consult Quixant for more details of the compact, high capacity, high performance SATA flash modules.

The QXi-6000 can accommodate a single standard 2.5-inch size SATA HDD or SSD. A tray can be fitted to take the drive and power connections made using J23 (see section 18.6) and data using one of these two SATA connectors.

18.8 J16 – SPI BIOS ROM Programming Header

Reserved - do not use.

18.9 J17A – FPGA Programming Header (Pins 1-10)

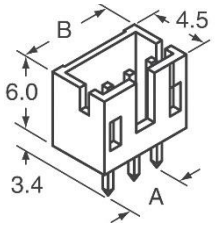
Reserved – do not use.

18.10 J17B – AVR (Logging Processor) Programming Header (Pins 13-18)

Reserved – do not use.

18.11 J20 – Reserved for future

Reserved for future use.

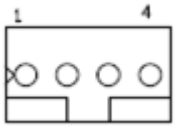


JST B2B-PH-K-S

Pin	Function
1	TBD
2	TBD
3	GND

18.12 J23 – SATA HDD Power

This connector can be used to provide power for an external disk drive. Care must be taken not to draw more than 0.5A from this connector.

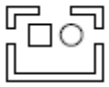


Molex 22-27-2041

Pin	Function
1	+12V
2	GND
3	GND
4	+5V

18.13 P1 – Intrusion Input Wafer

Spare internal intrusion input.

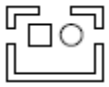


JST B2B-PH-K-S

Pin	Function
1	INTR_6
2	GND

18.14 P2 – Intrusion Input Wafer

Logic door intrusion input hooked up to micro switch on the logic door by default.



JST B2B-PH-K-S

Pin	Function
1	INTR_7
2	GND

18.15 SODIMM1, SODIMM2

The QXi-6000 provides sockets for two DDR4 SODIMM modules. These support JEDEC standard DDR3 SODIMM modules. Quixant strongly recommends only using modules supplied by Quixant to ensure stable operation and optimum performance. Details of the SODIMM module interface are outside the scope of this manual.

18.16 SW1- User Button

This button is connected to INTR10 of the Logging Processor. User can use this button to generate the intrusion event when the board is turned off. If enabled, this event will be reported to the application when restarted.

18.17 SW4 – User switches

This bank of four switches is available to the user, the values of which are readable in software through digital inputs DIN56~59.

19 Device Sockets

19.1 SODIMM1, SODIMM2 – DDR4

The QXi-6000 provides sockets for two DDR4 SODIMM modules. These support JEDEC standard DDR4 SODIMM modules. Quixant strongly recommends only using modules supplied by Quixant to ensure stable operation and optimum performance. Details of the SODIMM module interface are outside the scope of this manual.

19.2 U2 - EEPROM

24C02 pin compatible EEPROM c. Consult Quixant for further details on usage. Not available to game application software.

19.3 U25 – BIOS ROM

SPI BIOS ROM. This device holds the BIOS code for the QXi-6000 system.

19.4 U29 – FPGA ROM

SPI ROM. This device holds the configuration data for the QXi-6000 gaming logic.

19.5 U21 – User I^2C EEPROM

User 24C02 pin compatible EEPROM connected to the gaming logic. This device can be accessed using the Quixant I2C API. See section 9.4 or further details. Do not use if U22 is populated.

20 BIOS Settings and Installation

20.1 Operating System Minimum Requirements

Quixant recommends using a 64-bit OS to get the best performance with the QXi-6000. This is required to access memory past 4GB and it means that the memory used by the APU for video will not restrict how much memory your application can use as much, if you have 8GB or more.

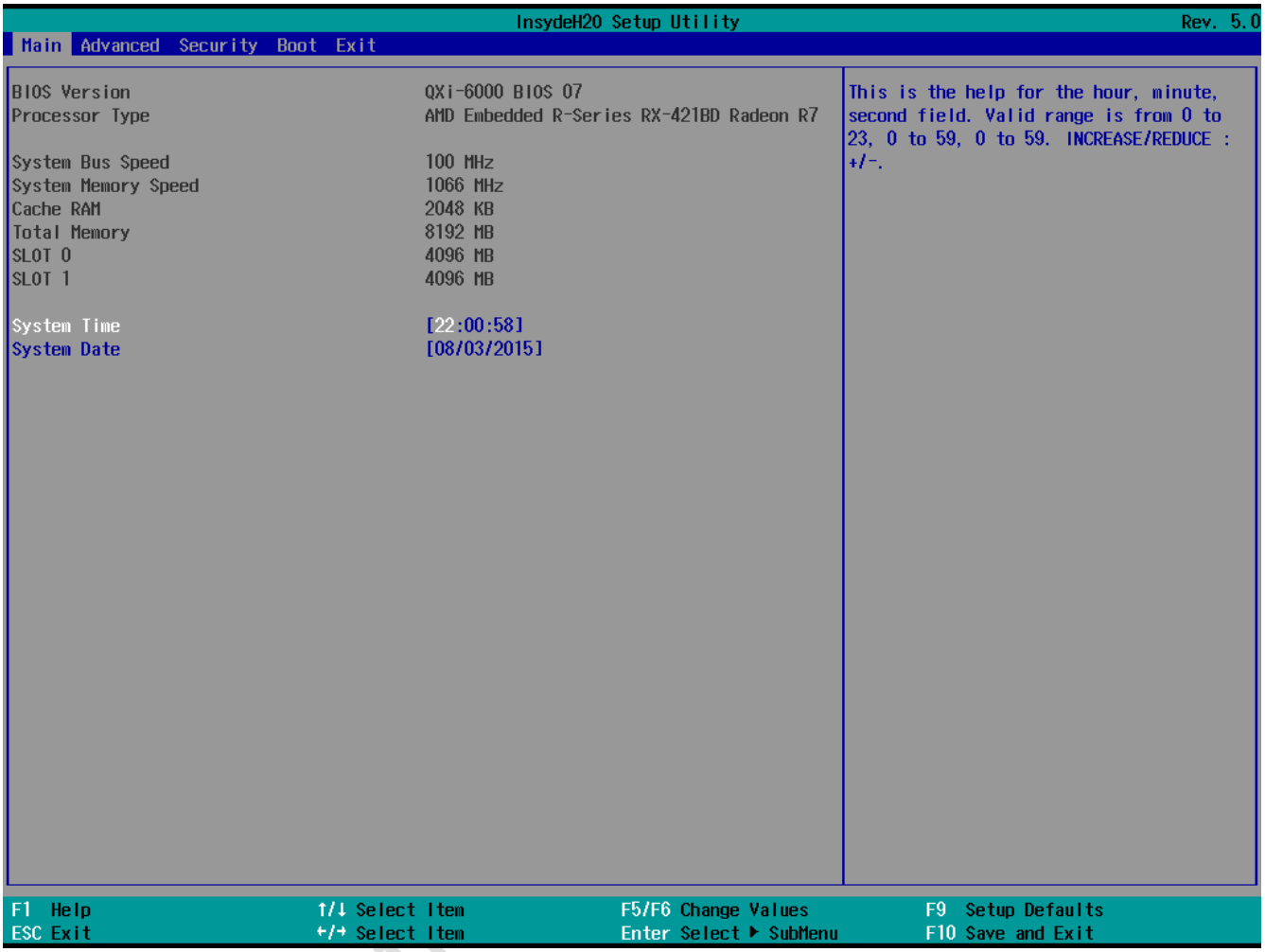
Any operating system should support SATA devices during installation, so if you use Windows XP (not recommended) you should use the Service Pack 3 (SP3) version. This is not a hard requirement, since SATA devices can run in the older PATA mode for backwards compatibility, but they will not give as good a performance when used that way.

20.2 Entering BIOS Setup

When the system boots you can use the **Esc** key to enter into the BIOS configuration screen. The Quixant systems boot very quickly, so be ready to hit the key as soon as the system boot. You will only have 1-2 seconds to do so before it starts to load from any bootable media, so be ready.

20.3 Main Screen – Setting Date and Time

This is the initial screen you see when you enter the BIOS Setup screens:



[System Time (hh/mm/ss)]:

[System Date (week mm/dd/yyyy)]:

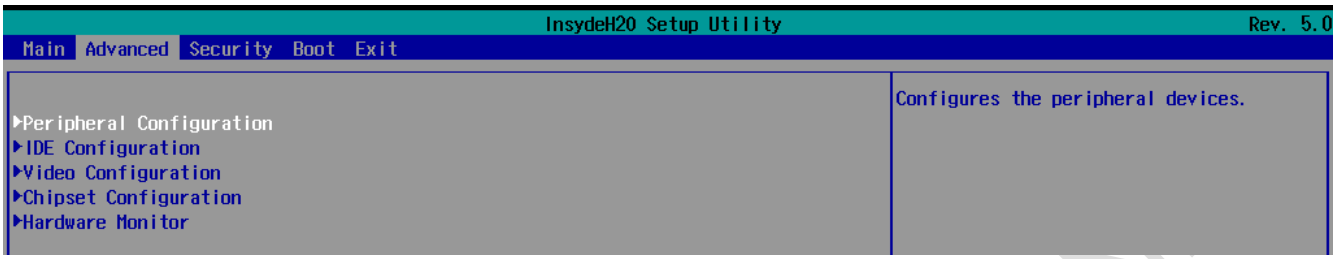
Allows the user to set the time and date for the system. The Time is displayed in 24-hour format. The Date can be set from January 1st, 1980 to December 31, 2099.

The values set in these two fields take effect immediately.

Note: This is the only screen image that will be shown full size, so you can see the legends at the bottom. All others will be cropped to below the last line of the screen.

20.4 Advanced Screen

Using the right (or left) cursor key, move to the **Advanced** screen from your current screen.



[Peripheral Configuration]: Configures the peripheral devices.

[IDE Configuration]: Select the host controller and hard disk drive type installed in your system.

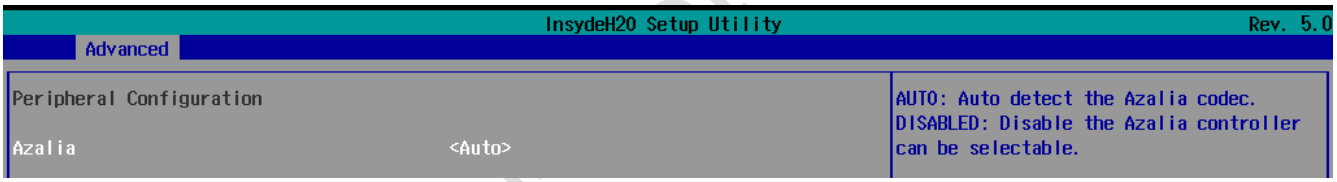
[Video Configuration]: Configures video settings.

[Chipset Configuration]: Advanced Chipset Configuration Options.

[Hardware Monitor]: Monitors CPU temperature and Fan speed.

20.4.1 Peripheral Configuration

Using the down (or up) cursor keys move until the **Peripheral Configuration** selection in the menu is highlighted and hit **<Enter>** to select it:



[Azalia]: Allow user to auto-enable or disable detecting the Azalia HD audio controller feature.

[Auto]: Auto detects the Azalia controller.

[Disabled]: Disables the Azalia controller.

[SD controller configure]: Allow user to configure the SD controller.

[Disabled]: Disable the SD controller.

[SD 3.0]: Configure SD controller to support SD 3.0.

[SD 2.0]: Configure SD controller to support SD 2.0.

[SD 3.0 SDR50]: Configure SD controller to support SD 3.0 with SDR50 transfer mode.

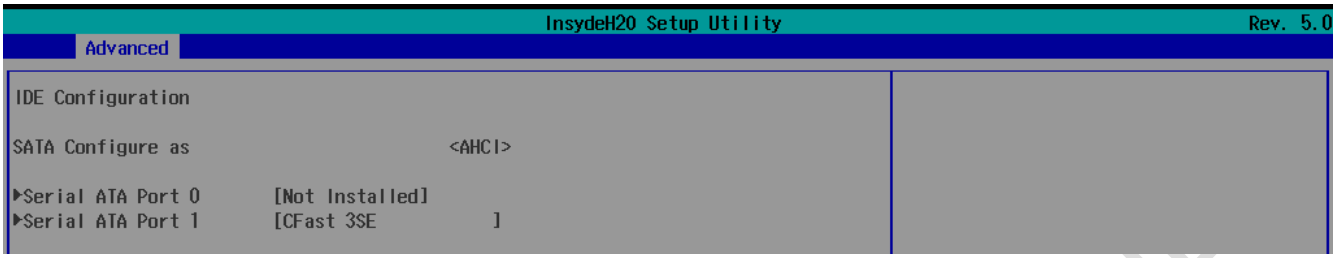
[SD 3.0 DDR50]: Configure SD controller to support SD 3.0 with DDR50 transfer mode.

[SD 3.0 SDR50]: Configure SD controller to support SD 3.0 with SDR104 transfer mode.

When complete, use the **<ESC>** key to get back to the **Advanced** screen.

20.4.2 IDE Configuration

Using the down (or up) cursor keys move until the **IDE Configuration** selection in the menu is highlighted and hit **<Enter>** to select it:



[SATA Configure as]: Allow user to set SATA Mode

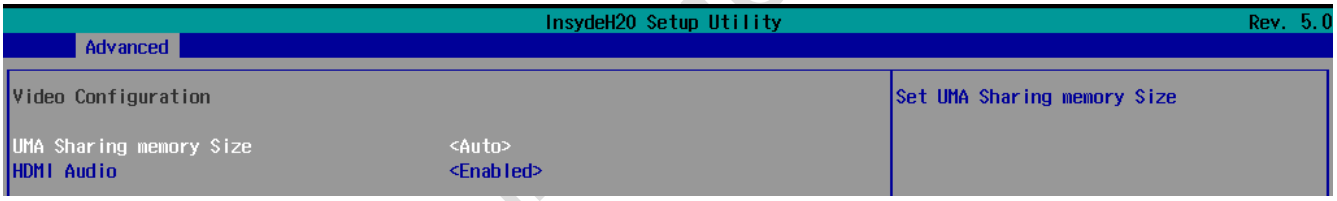
[IDE]: Set host controller to IDE mode.

[AHCI]: Set host controller to AHCI mode.

When complete, use the **<ESC>** key to get back to the **Advanced** screen.

20.4.3 Video Configuration

Using the down (or up) cursor keys move until the **Video Configuration** selection in the menu is highlighted and hit **<Enter>** to select it:



[UMA Sharing memory Size]: Allow user to set the amount of system memory used by the internal GPU.

[Auto]: Auto mode follow Memory size.

[1GB]: Set UMA Sharing Memory Size is 1GB.

[2GB]: Set UMA Sharing Memory Size is 2GB.

[HDMI Audio]: Allow user to enable/disable HDMI Audio.

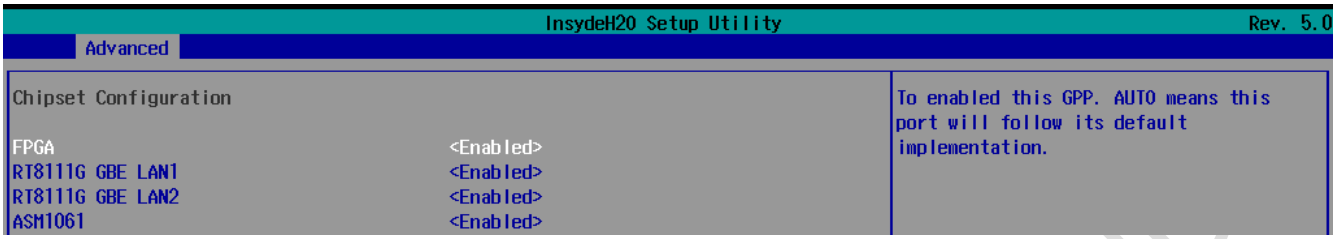
[Enabled]: Enable HDMI Audio.

[Disabled]: Disable HDMI Audio.

When complete, use the **<ESC>** key to get back to the **Advanced** screen.

20.4.4 Chipset Configuration

Using the down (or up) cursor keys move until the **Chipset Configuration** selection in the menu is highlighted and hit **<Enter>** to select it:



Currently this screen if for information only and contains no settings that are changeable by the user.

[FPGA]: Allow user to enable/disable FPGA function.

[Enabled]: Enabled FPGA function.

[Disabled]: Disabled FPGA function.

[RT8111G GBE LAN1]: Allow user to enable/disable RT8111G GBE LAN1 function.

[Enabled]: Enabled function.

[Disabled]: Disabled function.

[RT8111G GBE LAN2]: Allow user to enable/disable RT8111G GBE LAN2 function.

[Enabled]: Enabled function.

[Disabled]: Disabled function.

[ASM1061]: Allow user to enable/disable ASM1061 function.

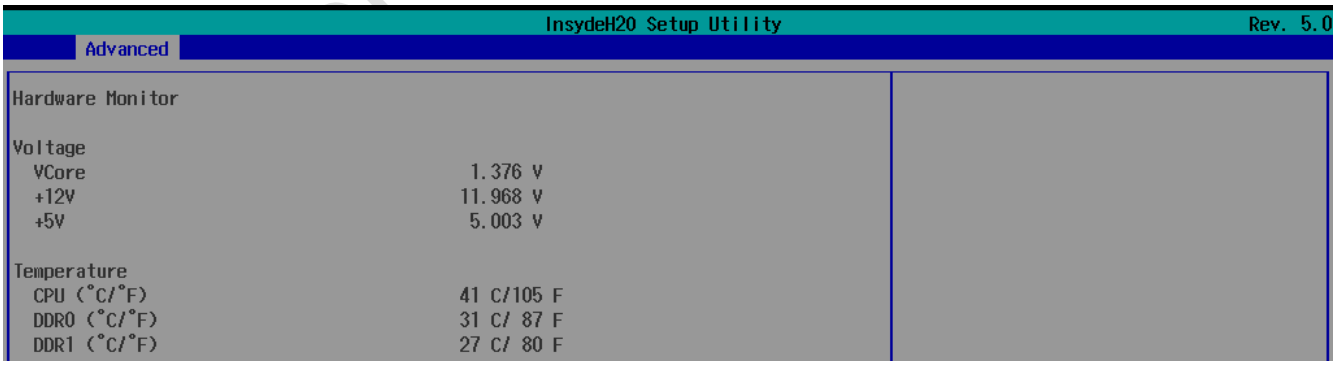
[Enabled]: Enabled ASM1061 function.

[Disabled]: Disabled ASM1061 function.

When ready, use the **<ESC>** key to get back to the **Advanced** screen.

20.4.5 Hardware Monitor

Using the down (or up) cursor keys move until the **Hardware Monitor** selection in the menu is highlighted and hit **<Enter>** to select it:

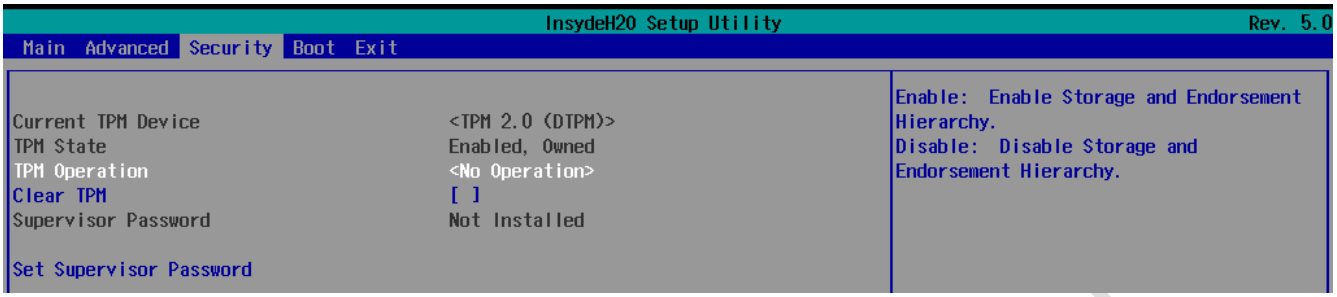


Nothing here can be changed, but the temperature and fan speed values will be continuously updated, showing overheating or fan problems. On a fanless system there will be no fan speed reported.

When ready, use the **<ESC>** key to get back to the **Advanced** screen.

20.5 Security Screen – TPM Operation, Supervisor Password, and Power on Password

Using the right or left cursor key, move to the **Security** screen from your current screen.



[TPM Operation]: Allow user to enable/disable TPM function.

[No Operation]: TPM chip will stay activated or deactivated based on the last selected and saved operation.

[Disable and Deactivate]: Disable and deactivate TPM.

[Enable and Activate]: Enable and activate TPM.

[Clear TPM]: Allow user to clear TPM function (optional).

[Set Supervisor Password]: Allow user to set the supervisor password that will be required to enter Setup. The password can be removed by entering a blank new password. Hit **<Enter>** when done to return to the **Security** screen

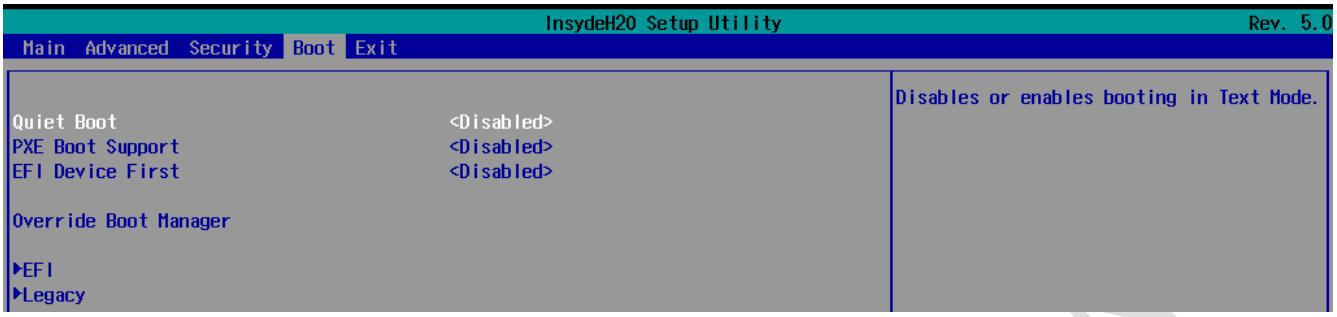
[Power on Password]: Allow user to ask input password at post time. **Note:** This option only appears if a Supervisor password has been set.

[Enabled]: This causes the system to ask the user to input the Supervisor Password, at the end of POST completing, to allow the system to boot.

[Disabled]: Anyone is allowed to boot the system without a password.

20.6 Boot Screen

Using the right or left cursor key, move to the **Boot** screen from your current screen.



[Quiet Boot]: Allow user to enable/disable booting in Text mode.

[Enabled]: Set the system to display a logo when booting.

[Disabled]: Set booting in Text mode.

[PXE Boot Support]: Allow user to enable/disable network boot support (optional).

[Enabled]: Enable PXE network boot support.

[Disabled]: Disable PXE network boot support.

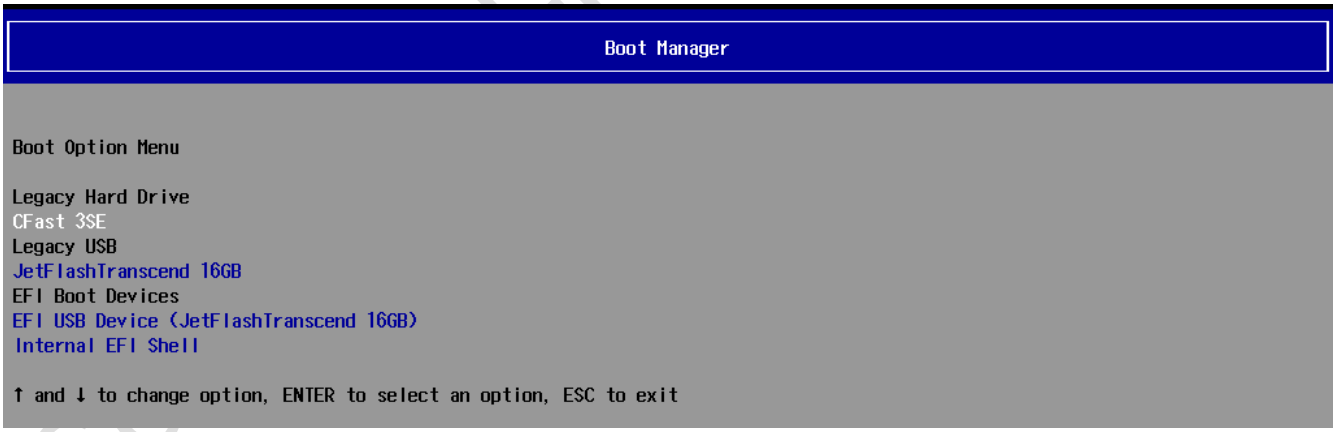
[EFI Device First]: Allow user to determine if an EFI or legacy device should be checked for boot first.

[Enabled]: Any EFI device that is present is checked first to see if it is bootable.

[Disabled]: Any Legacy device is checked first for being able to boot.

20.6.1 Override Boot Manager

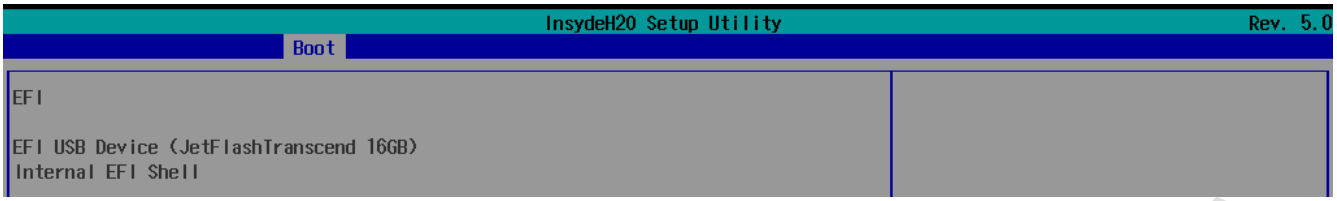
Select this item to allow you to manually choose which device to boot from. This will override the automatic search order in the case where you have multiple bootable devices.



Simply use the up and down cursor keys to select what you wish to boot and hit **<Enter>** to boot from it.

You can hit **<Esc>** to exit this screen if you decide not to choose a bootable device.

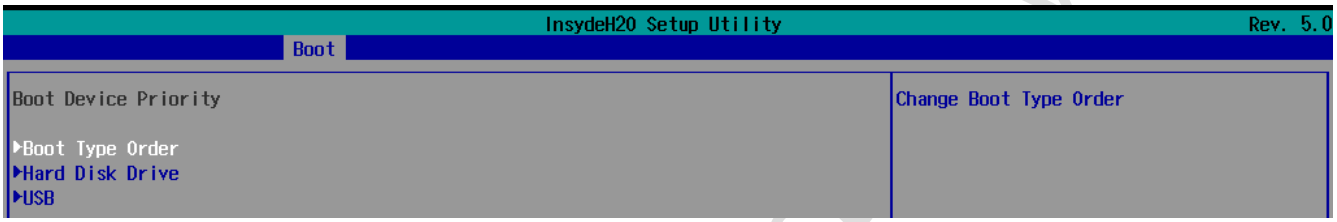
20.6.2 EFI Screen boot device configuration



This allows you to choose the order of EFI devices to boot. On a system with no EFI devices the list is simply **Internal EFI Shell**. This is the shell you end up in if no bootable devices are present.

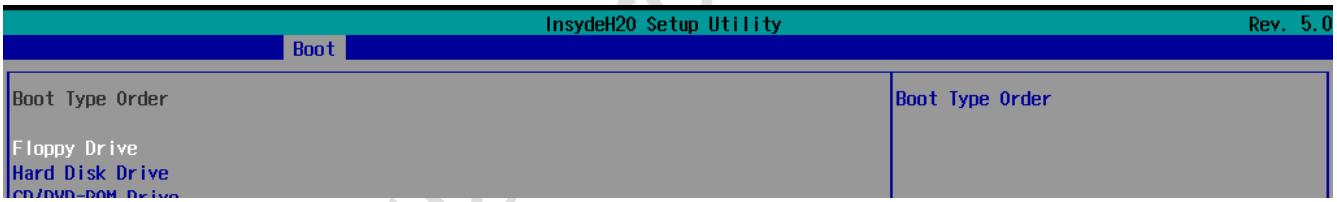
20.6.3 Legacy Screen boot device configuration

Bring up the legacy boot device by selecting **Legacy** from the **Boot** screen and pressing **<Enter>**. This screen is used to configure SATA and CFast boot devices, as well as floppy, CD/DVD, PXE, and USB boot devices. It looks something like this when you enter it:



This screen has nothing in it to set, but you can choose from three sub-screens where you can change settings that affect boot device priority

20.6.4 Legacy – Boot Type Order



To change the boot order of legacy devices you choose the **Boot Type Order** from the **Legacy** screen and press the **<Enter>** key. The default configuration looks like the example below, but your list may be slightly different depending on your BIOS version. If you will be using a CD/DVD or USB device to boot from on occasion you probably should move Hard Disk Drive down below those entries.

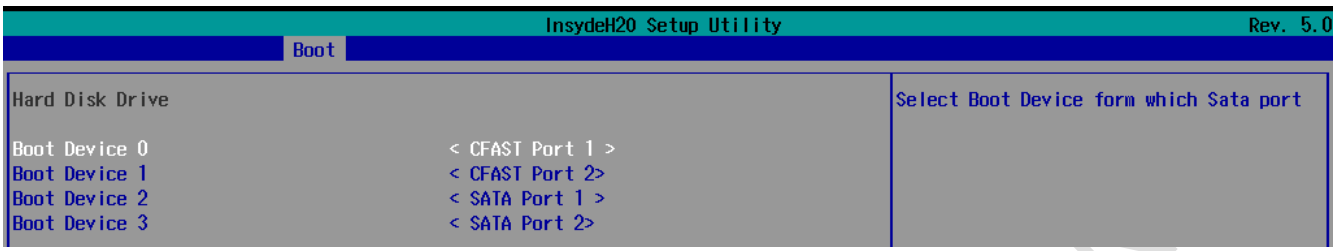
To change the order use **<F5>** to move the selected item lower in the list and **<F6>** to move it higher.

As in most cases for screens like this you use **<ESC>** to return back to the previous screen.

20.6.5 Legacy – Hard Disk Drive order configuration

To change the order of Hard Disk devices, select **Hard Disk Drive** in **Legacy**: The default order is shown in the example below.

As in most cases for screens like this you use **<ESC>** to return back to the previous screen.



[Boot Device 0]: Select Frist Boot device.

[CFAST Port 1]: Set Boot Device 0 as CFAST Port1.

[CFAST Port 2]: Set Boot Device 0 as CFAST Port2.

[SATA Port 1]: Set Boot Device 0 as SATA Port 1.

[SATA Port 2]: Set Boot Device 0 as SATA Port 2.

[Boot Device 1]: Select Second Boot device. Options are the same with Boot Device 0.

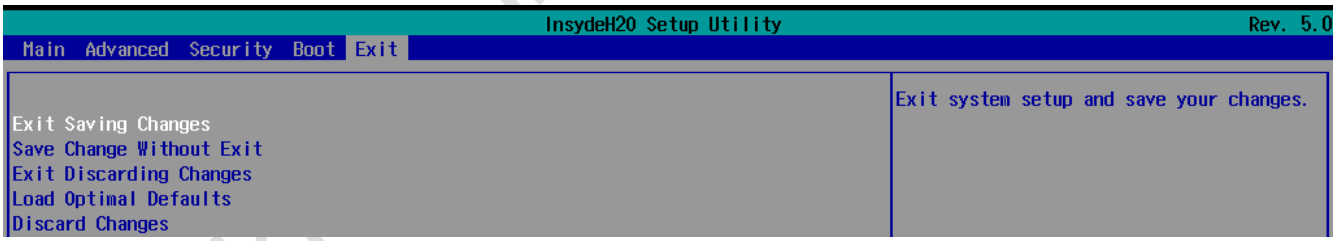
[Boot Device 2]: Select Third Boot device. Options are the same with Boot Device 0 (optional).

[Boot Device 3]: Select Fourth Boot device. Options are the same with Boot Device 0 (optional).

Note: The BIOS will let you set duplicate boot devices. This allows you to exclude devices from the being considered as bootable hard disks. There is no provision to set a Boot Device to None, so setting it to be the same as an earlier device is how you can do this.

20.7 Exit Screen

Using the right or left cursor key, move to the **Exit** screen from your current screen.



[Exit Saving Changes]: Exit system setup and save your changes.

[Save Change Without Exit]: Save your changes without exiting the Setup screens.

[Exit Discarding Changes]: Exit system setup without saving your changes.

[Load Optimal Defaults]: Load optimal standard defaults for the system.

[Discard Changes]: Discard any changes made to setting since the last Save.

Note: You must be sure to save any changes that have been made prior to booting the system. You can also do the **Exit Saving Changes** at any time by entering **<F10>**. Similarly, you can **Load Optimal Defaults** by entering **<F9>**. These two keys can be used from any screen, saving the need to navigate to the **Exit** screen.

21 Software Drivers and Libraries

21.1 Quixant Support Website

All drivers needed to work with the hardware of your Quixant QXi-6000 system are available from the Quixant Support website: http://support.quixant.com

You can request a login account from your Quixant Sales representative. A Non-Disclosure Agreement (NDA) and a Software License Agreement (SLA) need to be in place with your company to create the account. If you need accounts for outside contractors, they must also have an NDA and SLA in place.

The website also contains libraries that work with the drivers and provide a consistent API between Quixant platforms. The APIs are OS agnostic and can be used with either 32-bit or 64-bit programs..

Drivers and libraries are provided for Windows and Linux in both 32-bit and 64-bit versions. If a version you need is not available, please request it from your Sales or technical support representative. Every attempt is made to keep the Windows and Linux versions in sync, but sometimes a platform version or revision is made available before the others due to customer needs.

21.2 How to Install Quixant Drivers – Windows Systems

All Quixant-specific devices are by default listed in Windows systems **Device Manager** (*Control Panel → System and Security → Device Manager* for Windows 7). Where devices show up depends on if a driver is installed already or not for the device.

If no driver is installed they will be under the **Other devices** node, as shown in Figure 21-1, plus your Display adapters will appear as VGA devices, your LAN ports as Ethernet Controller, and you will see an SM BUS and USB Controller as well.

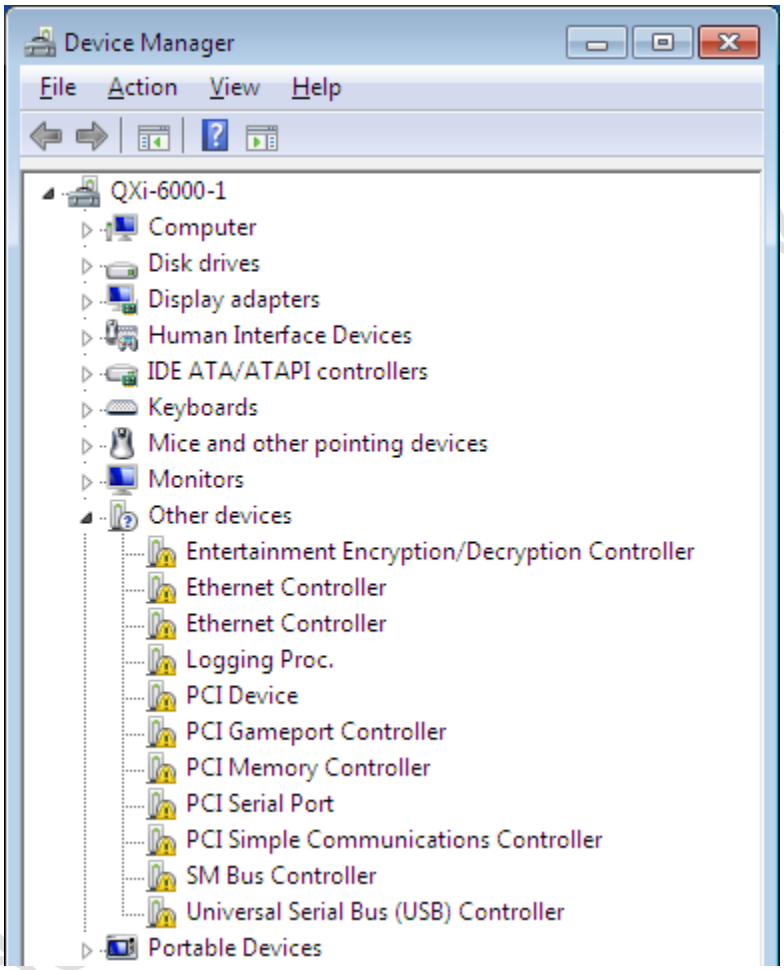


Figure 21-1: Device Manager showing Quixant devices

Note: For Windows 7 users, please be sure to have update KB3033929 installed on your system before installing Quixant Drivers. For more info, see your driver package README file. Since all driver signatures released after the 1st May 2015 have been updated to SHA-2, Windows 7 must have either KB3033929 (specific) or KB2718704 (cumulative security update) updates installed. All of these updates are freely available through Windows Download Center on the Microsoft Website.

Windows XP drivers won't be updated anymore, and last stable build will not be signed, since this requirement is not enforced in Windows XP. To know if a specific driver can be ported to Windows Vista, contact support at support@quixant.com. For Windows 8.1 no update is required.

All drivers for these devices can be downloaded from Quixant Support Site.

Listed below are the drivers supplied by Quixant corresponding to the various devices:

Device name	Corresponding Driver	Product ID

Entertainment Encryption/Decryption Controller	Quixant Security SS	R_DRV_QXTSECS
PCI Gameport Controller	Quixant Core	R_DRV_CORE
PCI Memory Controller	Quixant NVRAM	R_DRV_NVRAM
Logging Processor	Quixant Logging Processor	R_DRV_LP_WIN
PCI Serial Port	Quixant Serial Multiport	R_DRV_SERUNI
Communications Port	Quixant SERU	R_DRV_COM
PCI Simple Communication Device	Quixant QXCOM Multiport	R_DRV_QXCUNI
Multifunction Device	Quixant QXCOM	R_DRV_QXCOM

It is recommended that you install drivers in this order (no reboot required until are have been installed):

1. AMD Video (which includes several chipset drivers including those for SM Bus and USB 3.0)
2. Ethernet
3. Audio
4. Quixant Core, SECS, NVRAM, and Logging Processor (in no particular order)
5. Quixant SERUNI
6. Quixant COM (for port naming, see section 21.3.1)
7. Quixant QXCUNI
8. Quixant QXCOM
9. Logging Processor

The first three items are third party drivers and use a Setup application to install. The AMD Video driver is recommended to be the first driver, because in addition to video drivers it also installs several chipset drivers for the system, like USB 3.0 drivers.

QXi-6000 equips a serial multiport interface for both COM and QXCOM ports. When Quixant Multiport Drivers are installed, six new port devices are created for each one: in the case of standard COM ports, system drivers are installed automatically (check port enumeration), while for QXCOM ports, devices are listed as “Multifunction device” and must be manually installed using Quixant QXCOM drivers.

To install a Quixant device driver, download the Windows Multi Platforms driver package with the corresponding product ID from Quixant Website, and unzip the folder corresponding to the operating system installed on the QXi-6000 on your hard drive.

From Device Manager, right-click the device for which you downloaded the drivers and choose ‘Update Driver Software…’ from the context menu, as shown in Figure 21-2.

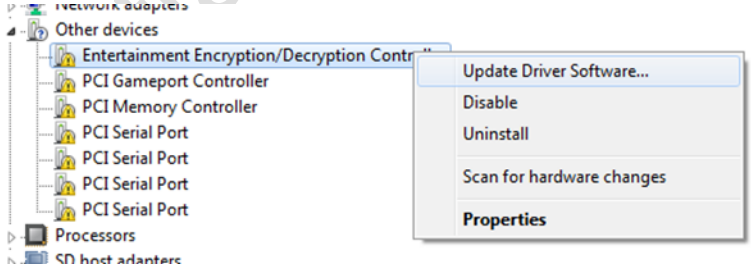


Figure 21-2: Driver installation from Device Manager

From the Update Driver Software menu, choose Browse my computer for driver software (see Figure 21-3), and in the next screen, hit 'Browse…' button and select the folder where you previously unzipped the drivers you downloaded, as shown in Figure 21-4. After all is selected, click "*Next*" and the installation should proceed automatically. If, during installation, a message box appears asking if you want to trust Quixant as a driver manufacturer, choose "*Yes*" to complete the installation. After a short delay, the driver should be installed, and the device will be available right away. You only need to install it with his option for the first device. After the driver has been installed for the first device you can choose "Search automatically for updated driver software" for all the remaining devices.

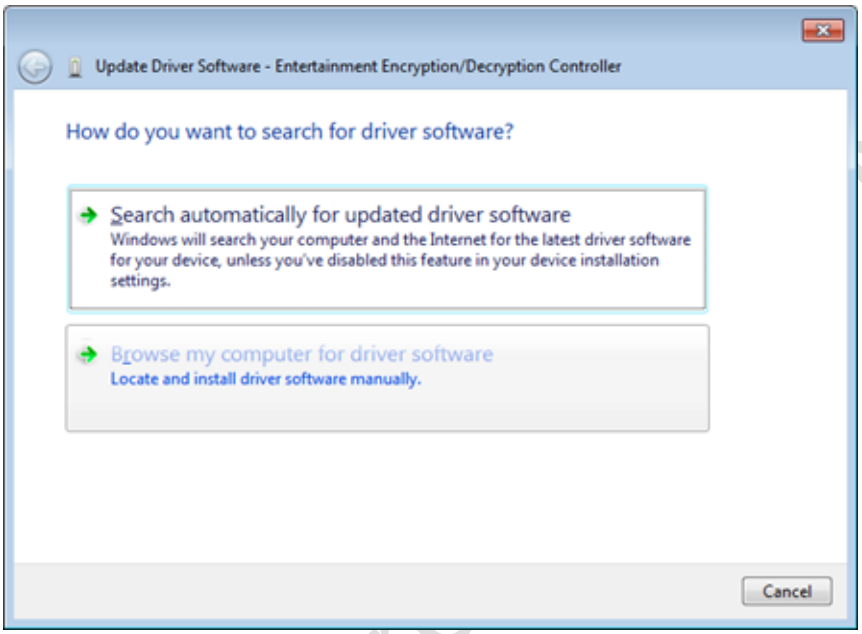


Figure 21-3: Update Driver Software dialog

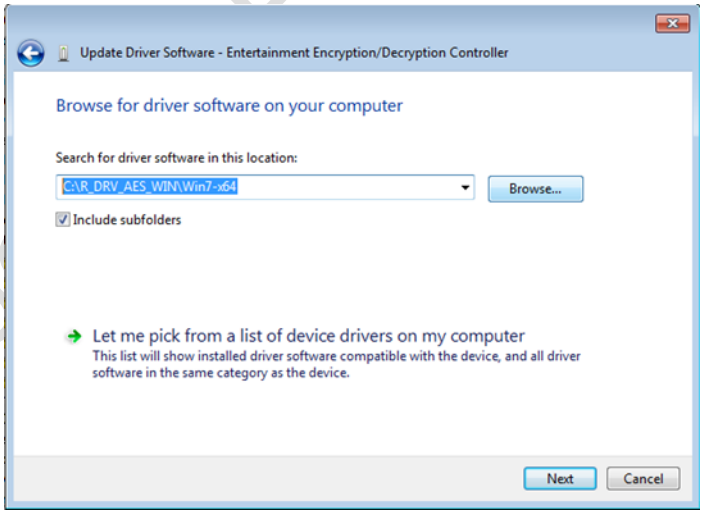


Figure 21-4: Browse for Update Driver Software dialog

To install Microsoft standard serial drivers, instead of Quixant Comm drivers for a PCI Serial Port, follow the same steps described earlier for a Quixant driver until you reach the dialog shown in Figure 21-3 and there choose "Let me pick from a list of device drivers on my computer".

21.3 Installing Serial Port Drivers

Once all the drivers except the serial drivers have been installed most of the *Other Devices* entries will have disappeared except for two. These two devices represent the multiport serial devices that the QXi-6000 supports and will look like this:

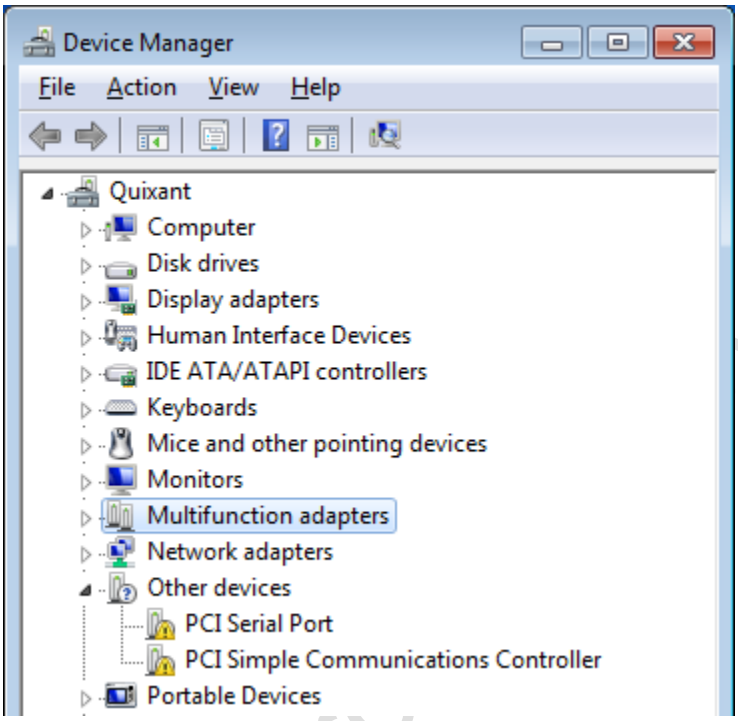


Figure 21-5: After installing all drivers except serial multiport drivers

As mentioned before the QXi-6000 has two different types of serial drivers. For standard COM serial devices, you must install the Quixant Serial Multiport driver first. Once this is installed the serial ports will appear in Device Manager and are automatically assigned to the Microsoft COM driver. Note that they will **not** be numbered correctly, because these are PCI devices at non-standard addresses. Until this driver is installed you will not see *Ports (COM and LPT)* showing in the Device Manager list. See Figure 21-6 to see how the COM ports can appear after this driver is installed.

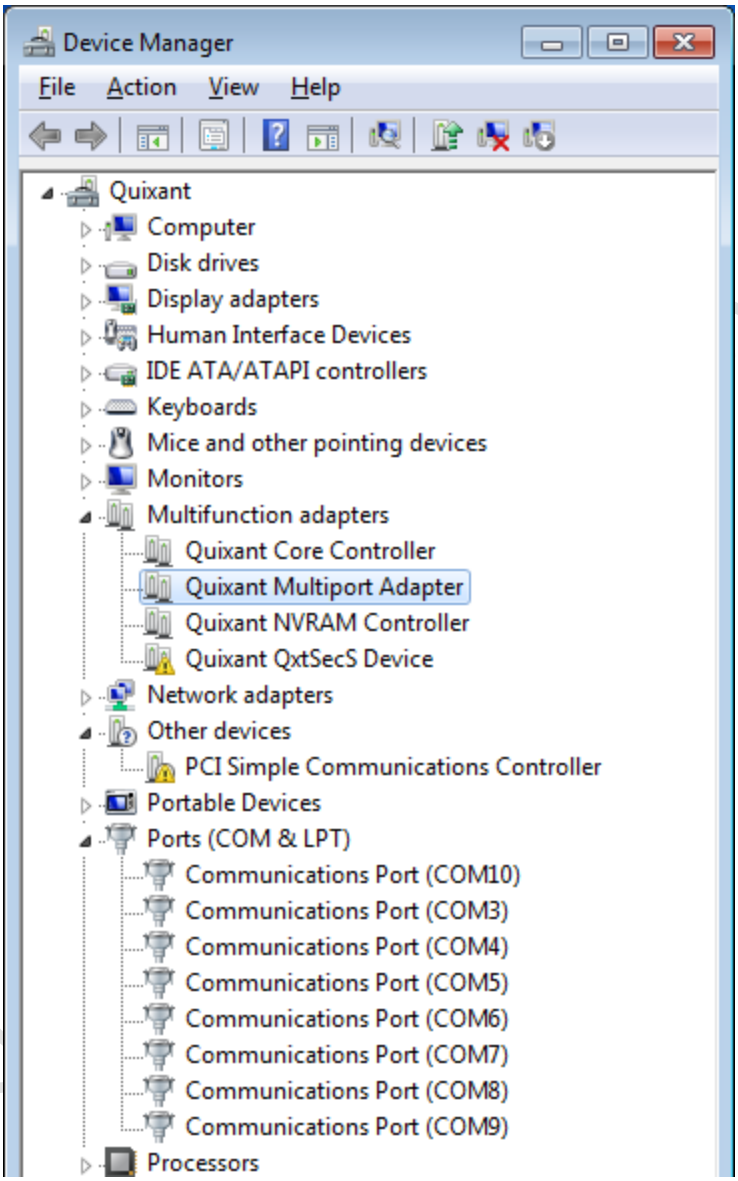


Figure 21-6: After installing Quixant Serial Multiport driver

To use the Quixant SAS software you need to install the Quixant SERU driver by following the same steps described earlier for a Quixant driver. Browse to the directory that contains the SERU driver and it will replace the Microsoft COM driver. In Device Manager this will show as a Quixant Comm Device, but with the same name as the Microsoft COM driver (COM1 for example). This driver is a modified version of the Microsoft COM driver designed to support 9-bit serial as used by the Quixant SAS library. You only need to install this for a port that will be using the Quixant SAS library. If using a third-party SAS solution you can leave the standard Microsoft COM driver in place if you wish.

21.3.1 How to Assign COM Device Names to Match Physical Ports

After installing Quixant Serial Multiport, Windows does not list the COM ports in the actual order the physical devices name been named, as you can see in Figure 21-6. If you are using Linux a similar issue may happen with the ttys device names. In Windows this can be resolved by manually assigning the COM device name using Device Manager and the Properties for the specific device. By default, Windows reserves COM1 and COM2 for devices at the standard PC addresses. Since none of the QXi-6000 ports have that address they will start at COM3 or higher.

While QXCOM (more on these later) devices will correctly name themselves, the COM devices will not. Here is the process to follow to properly assign the COM device names so they match the physical ports on the system

You can determine the physical device by the *I/O Range* numbers in the *Resources* tab of the device Properties dialog in Device manager. The easiest way to find which COM port is COM1 to COM6 is to open the Properties dialog for each on, click on the Resources tab, and arrange them on the screen in numeric order from low to high. The port addresses are in hexadecimal. Here are two examples showing some port ranges. The first I/O port number range is COM1 and the next one is 8 higher for COM2, etc. **Note: Your port addresses may show other numbers, depending on the system configuration.**

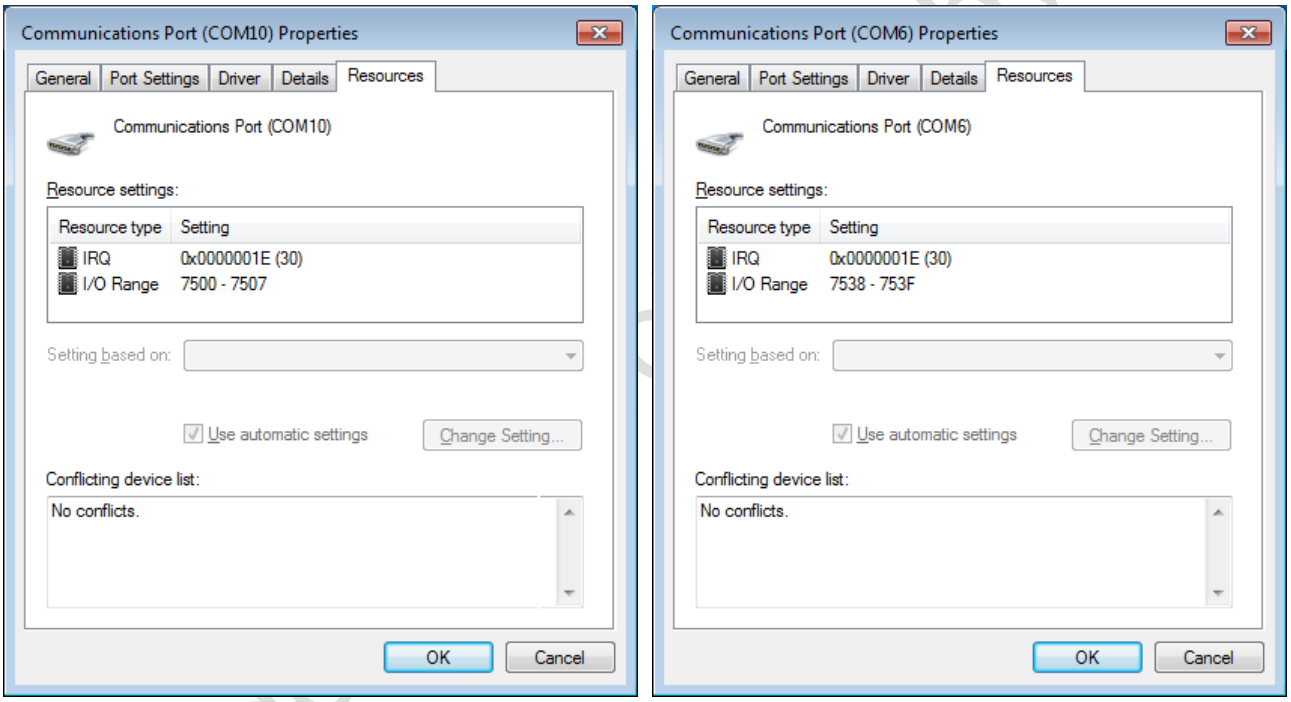


Figure 21-7 COM Port Resources

Once you have identified a COM device and know which port it belongs to you need to assign it to that logical name for Windows. This is done using the Port Settings tab on the Properties dialog you have open. Select that tab and you will see a button labeled Advanced, which you need to click on to change the assigned name, as show in Figure 21-8.

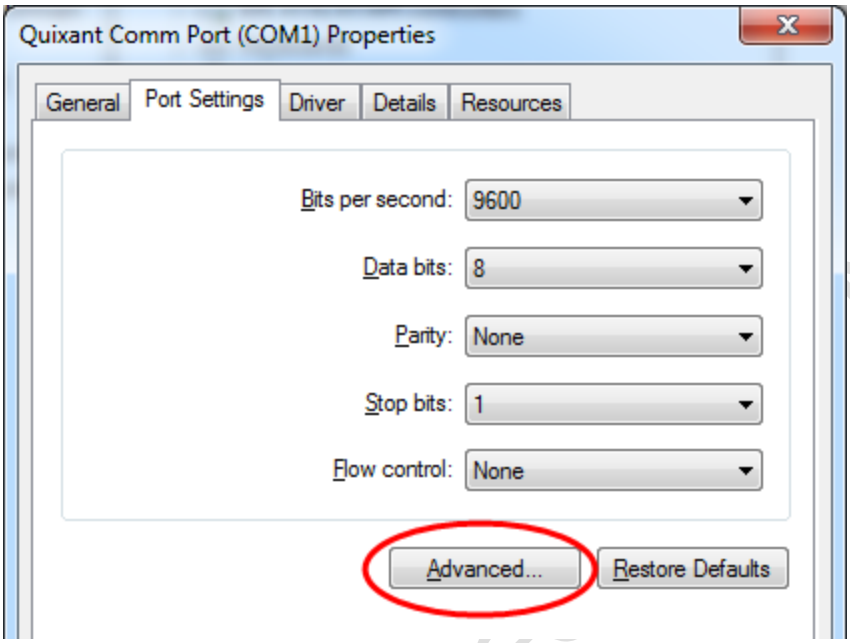


Figure 21-8 Port Settings

Once in the Advanced dialog you will see COM Port Number at the bottom left. This shows the name it is currently assigned to, but by clicking on it you get a scrollable list of all possible names. Other names that have had a device assigned to them will show “in use” after the name. Some of those devices may not appear in Device Manager, but this is because they previously had a device assigned and it was changed during a device configuration (driver installations and reboots can do this). Choose the device name you wish to use from this list.

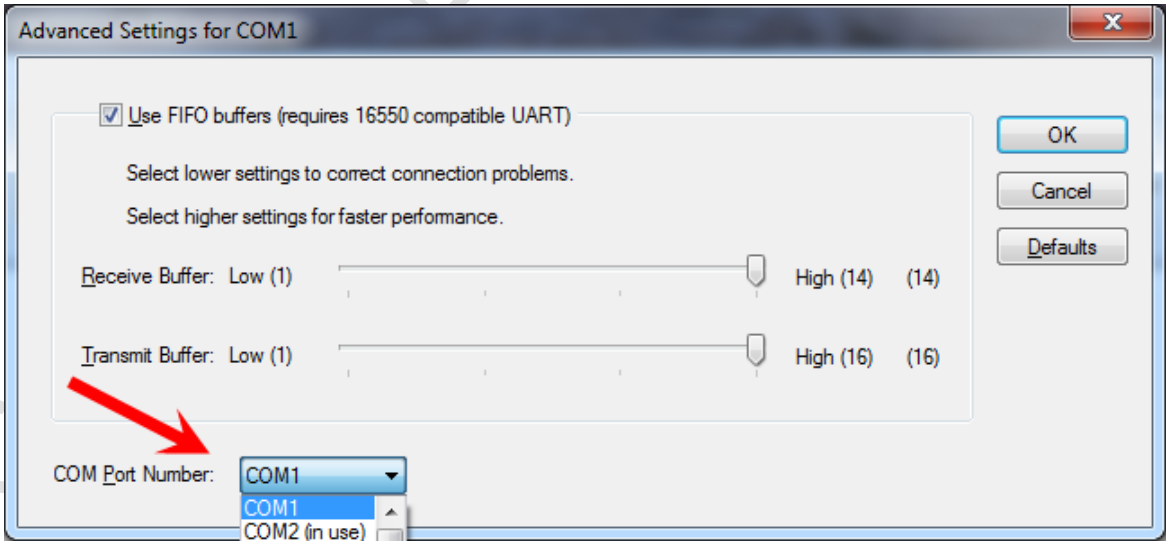


Figure 21-9 Advanced Settings for COM Ports

In the example dialog in Figure 21-9 you can see the list going down showing that COM2 is “in use”. Provided that you are using the disk only for this board type, you can safely rename ports, even if they are shown as “in use”. Just be sure not to configure them while you are naming them

These steps are not needed for the QXCOM devices, because they are not what Microsoft considers a standard COM device. For those devices the QXCOM driver will have been correctly assigned to ports 1 to 8 so that it matches the physical COM ports 1 to 8.

If you choose to use the standard Windows COM driver, you can still use the method outlined here in 21.3.1 to assign the driver to the correct physical COM port. You simply can't use the Quixant SAS library with it, but other SAS libraries that work with the standard driver will work just fine.

Once you have assigned all the COM devices to their correct name your Device Manager should now look like Figure 21-10 below.

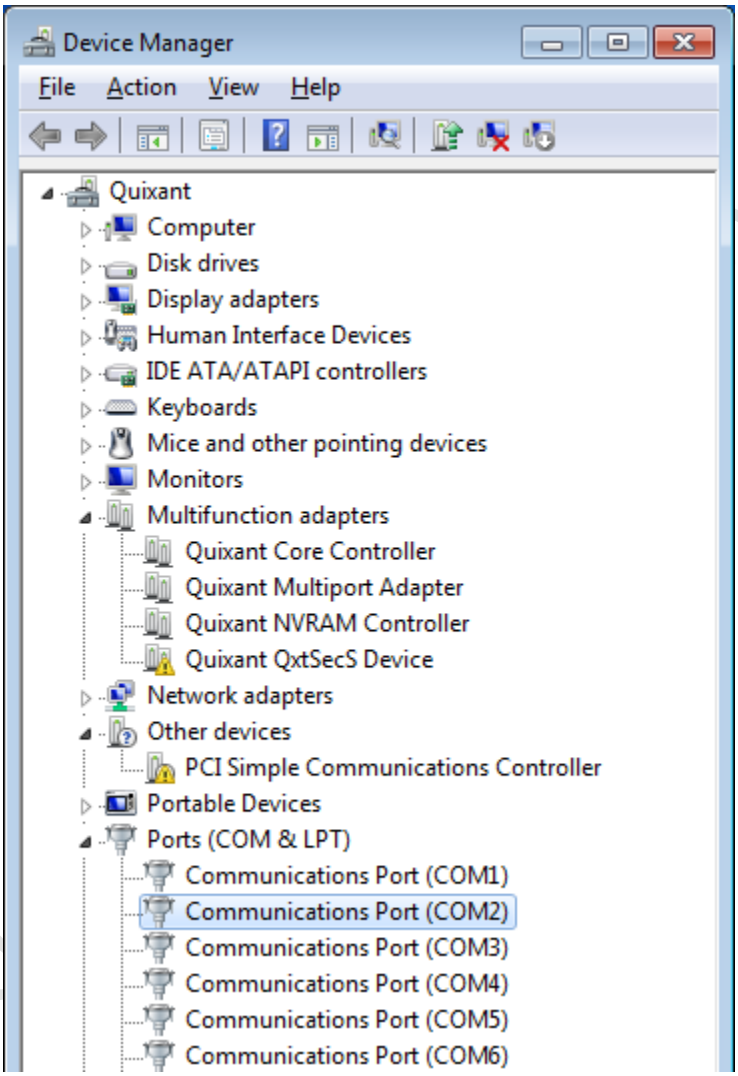


Figure 21-10: Device Manager with all COM port names correctly assigned

Note: After device names have been changed the system must be rebooted for the device names to be used for those devices. Windows uses the settings that were present at boot time until it has been rebooted with the new device name assignments. It is not recommended to change only a part of the serial drivers, while the others stay the same, because this may lead to unexpected behavior in the least recent drivers installed and to system instability. If Quixant COM drivers are needed, Quixant recommends to install them on all serial ports (standard serial functionality is still guaranteed).

You can tell which driver is assigned to a particular COM port by looking in Device manager. If it is described as a *Communications Port* the Microsoft driver is assigned to it. If it is described as a *Quixant Comm Port* the Quixant SERU driver is assigned to it. Both drivers will use the name COM#, where '#' is the number for the COM port.

21.3.2 Quixant Multiport Devices

QXi-6000 equips two multiport devices that share the same hardware connectors:

- A serial multiport that exposes six 165550-compatible UARTs
- A QXCOM multiport that exposes six QXCOM ports

In Windows, it is required to install multiport drivers in order to be able to use these devices. For serial multiport, install R_DRV_SERUNI driver. For QXCOM multiport, install R_DRV_QXCUNI.

See sections 21.2 and 21.3 for more information about how to install drivers for these devices.

NOTE: In Windows Embedded, driver components Multifunction and Multiport serial must be installed for this driver to work.

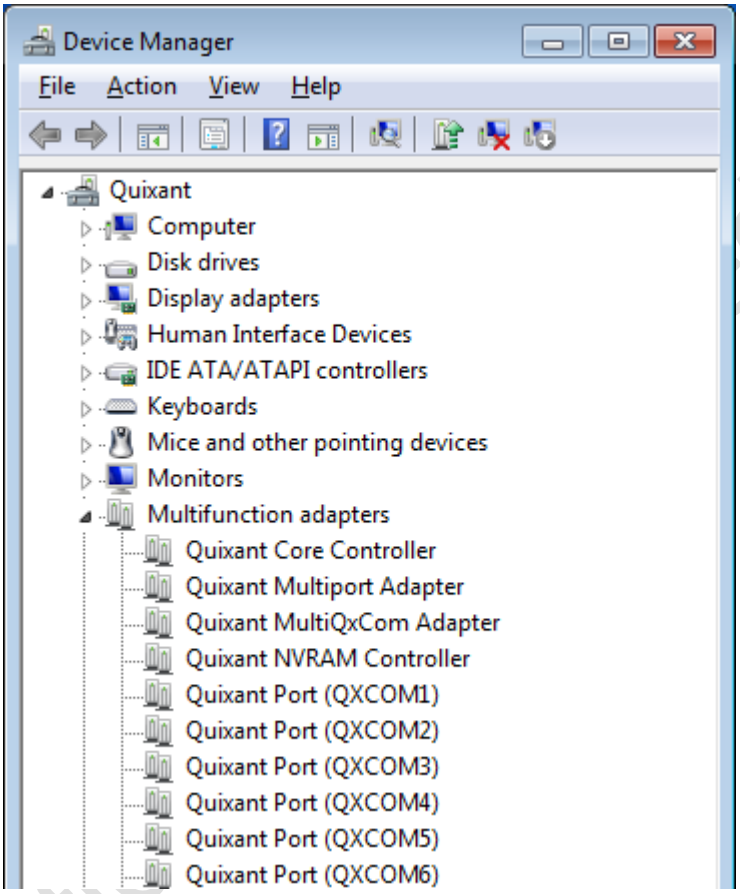


Figure 21-10 Device Manager showing Quixant Drivers

In Figure 21-10 above you can see how a QXi-6000 will look in Device Manager when all the Quixant drivers are installed and the COM devices are properly configured.

21.4 How to Install Quixant Drivers – Linux Systems

All the Quixant Device Drivers are provided as Unix TGZ compressed archives.

To obtain a fully working Qxi-6000 board, you need to install the following packages:

SECS drivers: R_DRV_SECS_LIN_x64 for 64-bit or R_DRV_SECS_LIN_x86 for 32-bit.

NVRAM Drivers: R_DRV_NVRAM_LIN_X64 for 64-bit or R_DRV_NVRAM_LIN_X86 for 32-bit.

Quixant Core Drivers: R_DRV_QXT_LIN_X64 for 64-bit or R_DRV_QXT_LIN_X86 for 32-bit.

Each archive filename has the format R_DRV_*xxx*_LIN_X*yyy-a.b.c.d*.tgz, where *xxx* is the package identifier; *yyy* is the architecture identifier (x86 or x64); *a, b, c, d* is a four digits signature, which identifies the version release.

For kernel neutrality, each archive contains the device driver and a simple tool to rebuild the driver against the current kernel.

The user must initially decompress each package, then run the driver build tool. In order to obtain an error free procedure, please follow the steps depicted in the next paragraphs.

1. Verify if the system includes the complete kernel headers package. Specific distribution tools (e.g. apt-get, emerge, yum, etc.) could be needed to provide this essential feature to install the Quixant drivers.
2. Identify the system architecture: 64- or 32-bit Linux. The output of the “uname” utility could be useful.
3. Decompress the archives. From the command line, supposing all the TGZ archives was present in the same directory, where a terminal is open, type this command (the following instructions suppose the 64-bit system; if you have a 32-bit Linux, substitute any x64 occurrence with x86):

```
$ tar zxvf ./R_DRV_SECS_LIN_x64-*.tgz
$ tar zxvf ./R_DRV_NVRAM_LIN_X64-*.tgz
$ tar zxvf ./R_DRV_QXT_LIN_X64-*.tgz
```

4. After the tar commands, four new subdirectories will be created in the current directory, namely, for 64-bit system and the current driver releases:

```
./R_DRV_SECS_LIN_x64-4.0.0.0
./R_DRV_NVRAM_LIN_x64-2.2.2.0
./R_DRV_QXT_LIN_x64-0.1.5.0
```

The 32-bit version would show the x86 identifier, instead of the x64 identifier.

5. Go inside the `R_DRV_SECS_LIN_x64-4.0.0.0` subdirectory and type the command:

```
./driverscomp
```

This tool rebuilds the driver against the current kernel release.

The first lines of the command output will appear similar to the next ones:

```
filename:       R_DRV_SECS_LIN-x86-4.0.0.0/secs.ko
version:        4.0.0.0
description:    Quixant SECS Driver
```

If no error message is displayed, the new secs.ko driver file will be present in the current directory and the `./qxt shell` script, which permits to install the driver.

6. Type the command:

```
./qxt
```

7. This command loads the kernel module and, if not otherwise signaled, the driver will be now up and running.
8. Repeat the steps 5 – 7 for the `R_DRV_NVRAM_LIN_x64-2.2.2.0` directory

The `./driverscomp` command this time will show the message:

```
filename:    R_DRV_NVRAM_LIN-x64-2.2.2.0/qxtnvram.ko
version:     2.2.2.0
description: Quixant NVRAM Driver
```

9. Repeat the steps 5 – 7 for the `R_DRV_QXT_LIN_x64-0.1.5.0` directory

The `./driverscomp` command this time will show the message:

```
filename:       R_DRV_QXT_LIN-x64-0.1.5.0/qxtio.ko
version:        0.1.5.0
description:    Quixant Gaming IO Driver
```

21.5 Driver Packages

You need to download drivers from the Quixant web site Client Area. While some of these drivers can be found on the Internet with some searching, it is strongly recommended that you only use those from the Quixant site. In some cases, the versions for Quixant hardware have changes that are not available for download elsewhere, plus they have been tested by Quixant.

Note that Windows 7 and most Linux distributions have a Realtek network and HD audio driver that will recognize the network and HD audio hardware at installation, but it is still recommended that you install the one from the Quixant Web Site. The drivers on the web site have been tested with the Quixant hardware platform.

You should also disable automatic updates to prevent driver updates from being installed.

In the table below third-party drivers for Quixant hardware platforms are listed. Many drivers may work for multiple Quixant hardware platforms where the hardware is compatible.

Driver Type	Driver Name
AMD Video & Audio	AMD Embedded APU Drivers
HD Audio	Audio and Codec Drivers
Network	Ethernet Drivers

For boards that had drivers that predate Windows XP, Windows XP drivers are bundled in the driver package. For Windows Vi / 7 / 8 / 8.1, x86 and x64 contain the compatible driver packages. For Windows 10, new releases of drivers contain a “Win10” folder with all the required drivers.

The 3rd party drivers from AMD and Realtek may come in versions specific to an operating system. These are packaged by those companies and Quixant has no control over that.

21.5.1 AMD Embedded APU Drivers

These drivers are available for download on Quixant web site Client Area. For Linux, a guide on how to install them is also available.

21.5.2 Audio and Codec Drivers

This is a driver for the Realtek HD audio hardware. For Windows XP there is a separate download from the other Windows versions (at this time).

Note: Some Linux distros have been known to have a driver that will attach to this device, but then later cause problems. Be sure to use the version from the Quixant website or a newer version from Realtek.

21.5.3 Multiport Drivers

These are available only in Windows (Linux serial drivers are automatically installed and QXCOM drivers take care of multiport aspect without needing an additional driver). These drivers expose six COM and six QXCOM ports, while occupying hardware resources as a single device (each).

21.5.4 Logging Processor Driver

In Windows, it is represented by an .inf file in a driver package. For Linux, follow instructions in driver installation section. This driver allows communication with on-board logging processor.

21.5.5 QXCOM Drivers

(In Windows, QXCOM Multiport driver is required to be installed to be able to use these drivers). These drivers allow the use of QXCOM ports, that are serial ports optimized for 9-databit communication, master -slave message exchange and incoming data packet timestamping.

21.5.6 COM Driver

This driver is only available for Windows at this time and is required is you are using the Quixant SAS library to support 9-bit serial protocol and high resolution time-stamping.

If you are not using SAS and do not need 9-bit serial support you can use the standard Windows and Linux drivers. The Quixant serial hardware is fully compatible with the existing drivers for those operating systems.

In Windows, Serial Multiport driver is required to be installed to be able to use these drivers.

21.5.7 Ethernet Driver

This is a driver for the Realtek network hardware. Because it is from Realtek it has different versions you need to download based on the OS you are using.

21.5.8 Quixant SECS, Core, and NVRAM Drivers

These drivers are supplied by Quixant. They provide many functions, including access to the digital I/O, logging processor, and I^2C devices. More drivers may be added to this list after this manual is published.

21.6 Software Libraries

This manual offers a brief, overview of all software libraries, which might be not up-to-date. To get updated information, **always** check the specific library documentation, available on Quixant Support Site. The libraries provided by Quixant are designed to work with the drivers provided by Quixant and also to reduce the need to have to write your own routines for some common gaming functions. The libraries that deal directly with Quixant drivers are designed so they can be used for 32-bit applications running on a 32-bit or 64-bit OS as well as 64-bit applications on a 64-bit OS.

The API for the libraries also stays consistent between Windows and Linux, allowing for migration of software from one OS to the other with little change.

Library Name	Windows Package	Linux Package	.NET Package
Core Library	R_LIB_QXT_WIN	R_LIB_QXT_LIN	R_NET_QXT_WIN
NVRAM Library	R_LIB_NVRAM_WIN	R_LIB_NVRAM_LIN	*
Security SS Library	R_LIB_SECS_WIN	R_LIB_SECS_LIN	R_NET_SECS_WIN**
Quixant Serial Library	R_LIB_QXCOM_WIN	R_LIB_QXCOM_LIN	*
I²C Library	R_LIB_I2C_WIN	R_LIB_I2C_LIN**	*
Logging Processor	R_LIB_LP_WIN	R_LIB_LP_LIN	R_NET_LP_WIN

(*): Included in all-in-one Windows library package

(**): Will be released soon

Windows libraries are all released as DLL, MT-Build, both for x64 and x86 architectures. All libraries are compatible with Windows 7 and later versions of Windows.

For static, MD-Build and XP versions of the libraries, contact Quixant Support at support@quixant.com

.NET libraries must match the corresponding architecture of the underlying DLL that is wrapped.

Please refer to the MSDN article for more info on how to use these libraries when building your code:

http://msdn.microsoft.com/en-us/library/2kzt1wy3.aspx

Here is a simple table to help set your build environment to avoid these errors based on which Static library you are linking with and how you are building your project. Note that other settings for your project might also affect this and this example if basically using default settings for a project in VS 2010:

Debug Win32 w/WinXP-x86 libraries		
Setting	Static MT-build libraries	Static MD-build libraries
/MT	Link Works	Link Works
/MTd	Link Fails	Link Fails
/MD	Link Works	Link Works
/MDd	Link Works	Link Works
Release Win32 w/WinXP-x86 libraries		
/MT	Link Works	Link Fails
/MTd	Link Fails	Link Fails
/MD	Link Works	Link Works
/MDd	Link Works	Link Works

Note: All libraries that are designed to work with a driver have a minimum driver version that must be installed for them to work as designed. Any driver version requirement will be listed on the download page when you download the library.

Two common errors can happen when linking with the Static libraries using Microsoft Visual Studio:

1. error LNK2038: mismatch detected for '_ITERATOR_DEBUG_LEVEL': value '0' doesn't match value '2' in <filename>.obj

2. error LNK2038: mismatch detected for '_MSC_VER': value '1700' doesn't match value '1600' in <filename>.obj
    a. The numbers for _MSC_VER will vary based on the compiler used to build the library and the code being linked with it.

Both of these errors are designed to prevent linking C++ object that are built be different compilers or with different debug configurations. This is because Microsoft changes how objects are passed depending on the compiler version and type of debug method it is built for. Since Quixant libraries do not use C++ object this is not a concern, except it can prevent a link from generating an executable.

Error #1 is caused by linking an object that has iterator debugging disabled ('0') or enabled ('1' or '2'). A setting of 2 is the default. For more information, check this article and video blog from Microsoft:

- https://msdn.microsoft.com/en-us/library/hh697468.aspx
- http://blogs.msdn.com/b/vcblog/archive/2011/04/05/10150198.aspx

Error #2 is caused by the version of the compiler being used. This is also explained in the video blog mentioned above, as well as in this article:

https://msdn.microsoft.com/en-us/library/b0084kay.aspx

You can modify the framework being used to avoid error #2 by following this article:

https://msdn.microsoft.com/en-us/library/ff770576.aspx

A common warning you may also see says it can't find a vc110.pdb or <Project>/Debug for a library and it says it links as if there is no debug information. This warning can be ignored, because there is no debug information in the Quixant libraries.

Because of these errors we recommend that you use DLL instead of Static libraries. Quixant is planning to stop supporting Static libraries in the near future.

21.6.1 SECS

This library provides an API that allows your code to do encryption and decryption using the hardware acceleration built into the Quixant system. It requires that the SECS driver be installed (section 21.5.8).

21.6.2 ccTalk

This library provides an API to perform ccTalk communication with external devices.

21.6.3 Quixant QXCOM Library

This library provides a master-slave serial interface that works with 16550 UARTs (COMx and ttySx ports) and QXCOM ports as well. It provides a unified interface as well as support for 9-databit communication, timestamping handling, and send/receive of messages with semantics support.

NOTE for Windows users: to use QXCOM library with 16550 UARTs, Quixant Serial Port driver is required in the target port.

NOTYE for Linux users: in 16550 UARTs, 9-databit support is present, but is emulated through a software interpolation. Check QXCOM Programmers Reference for details.

21.6.4 Error Handling

This library only offers legacy support for older libraries. Error handling is now embedded in every library release.

21.6.5 NVRAM

This is the library which supplies the API used to manage the NVRAM in the system. It allows for multiple banks and offloads the burden of reading and writing by using the hardware controlled by the NVRAM driver. The NVRAM driver is required for this library (Section 21.5.8).

21.6.6 Quixant I2C Library

This is a new library that has routines allowing for the management of a wider range of I2C device than the older Quixant library. It also has some significant performance enhancements in it compared to the Quixant library. This library requires the Quixant Core Drivers to be installed (Section 21.5.8).

The I2C functions in the Quixant library should be considered deprecated and code migrated to use this library instead.

21.6.7 Core Library

This library provides the API that is used to access basic I/O, timer and SPI access, and also implements legacy I^2C communication functions. It should always have the initialization routine called first before any other Quixant library is used. This library requires the Quixant Core Drivers be installed (Section 21.5.8).

21.6.8 QxtRav

This library provides a common API to handle various printers, barcode readers, and bill acceptors & validators common in the industry. This library requires that the Quixant Core Drivers be installed (Section 21.5.8).

The purpose of this library is to provide a common API that you can write your code for to handle whatever devices are required for the machine your software gets installed into. Currently it supports the JCM and MEI protocols.

21.6.9 SAS 6.0.2

The one library that is not provided on the website is SAS. To obtain the Quixant SAS library you must provide a full copy of your SAS license, so we can verify that you are licensed to have access to it. The SAS libraries and some example code will be emailed to you directly.

21.6.10 I²C Library

This library provides a modular API that is used to access I²C communication. It provides both generic functions for the most common access cases and specialized, atomic functions that can be used to script your own data access strategies and to interact with devices that enforce special rules to access data.

21.6.11 Logging Processor Library

This library is used to access logging processor features. It substitutes libqxt access functions with a new API and a new transport layer protocol.

21.7 Software Documentation

There are several documents, which are primarily designed to describe the API for each library, available in the client area.

21.8 Example Software

These are example programs you can use to see how the system works. In most cases it includes source code. Some of them have both in C and C# examples. In the case of C examples many can be run on Linux or Windows at the command line.

The list changes all the time, so they are not listed here.

21.9 White Papers

These are collections of best practices papers written by Quixant developers. Like the Example Software, the list changes over time as things are added, so they are not listed here.

21.10 Frequently Asked Questions

You have the ability to ask questions or make requests here that can be seen by Quixant engineers. For questions that may be of use to other customer's, answers will be made publicly available. These questions are answered as time permits.

Because a question may have been asked for a specific system it typically shows up as an answer under that system first. If it is determined that other systems may also benefit from the answer they are added.

So it can be worthwhile looking at FAQs under other systems if you do not see it as having been answered for your platform. Please let us know if you find an answer under a different platform that also applies to yours so we can review it and add it to the list for your platform.

22 Precautions and Regulations

22.1 Static/ESD

The QXi-6000 is an electro-static sensitive device, and can be damaged by high voltages caused by static electricity. Proper anti-static procedures should be followed when handling the QXi-6000 at all times, including field maintenance. This particularly applies when the case is opened for whatever reason.

22.2 Safety

Although the QXi-6000 motherboard features well-protected power supplies, and only uses low voltages, care should still be taken to avoid short circuits caused by conductive objects. The QXi-6000 should never have power applied if it has become damp, either through spillage or condensation, or damage may occur.

22.3 Ventilation

The QXi-6000 generates heat internally and the walls of the case are used to dissipate this heat into the surrounding air. Care should be taken to locate the QXi-6000 inside a cabinet in such a way as to ensure sufficient flow of cooling air over the finned surfaces of the QXi-6000 case. It is recommended that the QXi-6000 is mounted on a vertical surface so that the cooling fins are vertical within the cabinet. Thorough analysis of the actual thermal performance of the cabinet with all installed peripherals operational is recommended, taking into account the required external operating environment. The QXi-6000 is not designed to be housed inside a sealed enclosure.

22.4 RoHS

The QXi-6000 is designed using RoHS-compliant components, and is manufactured on a lead-free production line.

22.5 EMC

The QXi-6000 is designed to meet the requirements of CE and FCC when located inside a suitably designed metal enclosure and sensible EMC precautions are taken with all connected cabling and other peripherals. Care should be taken with all interconnecting cables to minimize EMC emission/susceptibility.

22.6 Batteries

The three system batteries on the QXi-6000 are 3V Lithium CR2450 devices, and should only be replaced with devices of the same type that are UL approved. Always check the orientation before inserting, and make sure that they are aligned correctly and are not damaged or leaking.

Never allow the battery to become short-circuited during handling. Never attempt to recharge a flat battery.

Follow all precautions provided by battery manufacturers on storage, temperature and humidity.

22.7 Environmental Information

Operation Temperature	**0°C ~ 45°C**
Storage Temperature	-20°C ~ 80°C
Relative Humidity	95% @ 40°C, non-condensing
Operation Vibration	1Grms/5~500Hz, IEC 60068-2-64
Non-Operation Shock	30Grms, 11ms, IEC 60068-2-27