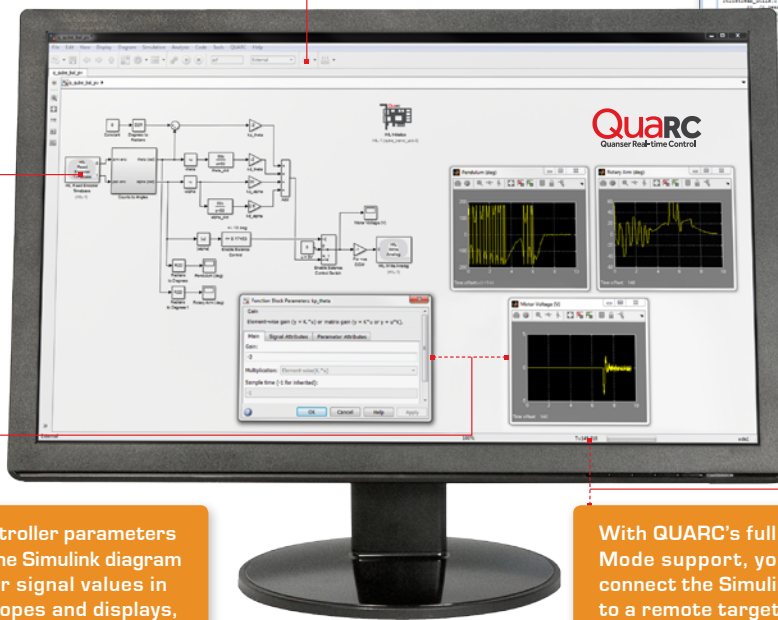# QUARC REAL-TIME CONTROL & RAPID PROTOTYPING SOFTWARE FOR MATLAB/SIMULINK

Quanser's QUARC™ software adds powerful tools and capabilities to MATLAB® and Simulink® to make the development and deployment of sophisticated real-time mechatronics and control applications easier. QUARC generates real-time code directly from Simulink-designed controllers and runs it in real-time on the Windows® target - all without digital signal processing or without writing a single line of code.

Create a new Simulink model using Simulink and QUARC libraries.

Click on the "Build" in the QUARC menu to generate real-time code.

More than 60 demonstration models and fully explained examples available to speed up development of your application.

Change controller parameters directly in the Simulink diagram and monitor signal values in Simulink scopes and displays, while the model runs on the target.

With QUARC's full External Mode support, you can connect the Simulink model to a remote target, and run the controller on it in real time.

> Using Quanser's QUARC software, we can easily design control systems for many plants. We can apply complex control strategies quickly and effectively – and it is very easy to verify theory on the real plant.
>
> - **Kenichi Yano,** Associate Professor, Gifu University, Japan

Built on the legacy of Quanser's WinCon, the first real-time software to run Simulink-generated code on a PC, QUARC redefines the traditional design-to-implementation interface toolset. The Canadian Space Agency played an intricate role in defining and confirming many of the features of QUARC. This was done in the context of their micro-satellite development program on an early stage prototype. QUARC has since been adopted by thousands of educational and research institutions worldwide, as well as industries requiring high performance and development flexibility, such as the aerospace, defense and medical device industries.

SIMPLE AND INTUITIVE TOOL FOR TEACHING LABS

QUARC™ is an integral part of Quanser's teaching lab workstations. With QUARC, students' efforts focus on key control concepts rather than tedious code writing. QUARC user interfaces are easy to understand, which means students do not require any extensive training. They work with controllers that are clear and match standard control system block diagrams used in textbooks. Students can tune parameters of the running model by changing block parameters in the Simulink diagram, view the status of a signal in the model and stream data to MATLAB workspace or to a file for off-line analysis.

Quanser teaching lab workstations are accompanied by comprehensive courseware with Simulink-based lab exercises. Developed by Quanser engineers, it effectively demonstrates the mechatronic design approach practised in industry, taking students through modeling, controller design and simulation to implementation.
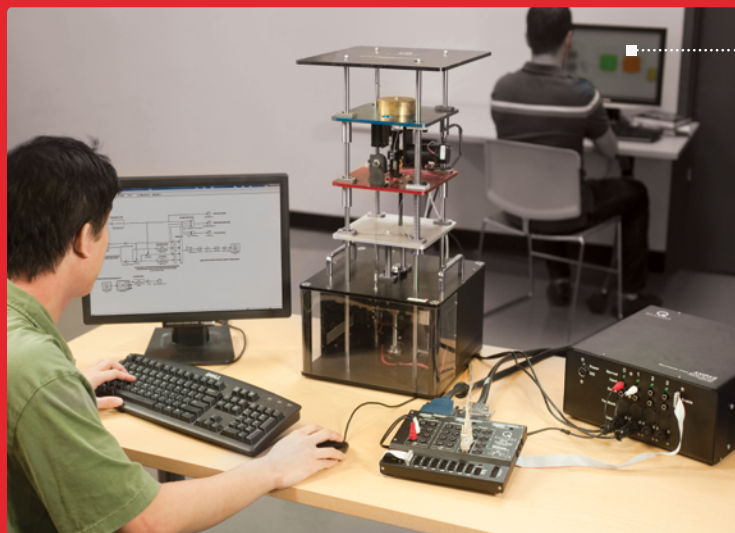
QUARC is designed so that the fundamental work is done, allowing students to focus more on the control design theory and less on workings of MATLAB and Simulink, thus improving the learning experience.

Students can actually design something that will work within the time frame of an undergraduate degree. They can really focus on the important control aspects. Without QUARC, they wouldn't have a hope of completing a project, in my opinion.

-David Wang,
Professor, University of Waterloo, Canada

QUARC Architecture

MATLAB SIMULINK®

Graphical Control Design

QUARC integrates seamlessly with Simulink's high-level graphical environment, and allows Simulink diagrams to interface hardware and be run in real-time on a variety of local and/or remote targets. With its extensive set of features and capabilities, QUARC makes it easier to design, implement and manage complex real-time applications.

+

**QUARC HOST
Simulink Development Environment (SDE)**

- Code Generation
- DAQ Support
- Real-time Communication
- Third-party Device Support
- 3D Visualization
- User Interface Support

QUARC TARGET

- Multiple Concurrent Targets
- Real-time Execution

EFFICIENT DEVELOPMENT ENVIRONMENT FOR COMPLEX APPLICATIONS

QUARC seamlessly integrates with Quanser's research platforms to implement virtually any control algorithm. Combined with Quanser power amplifiers and data acquisition cards, QUARC provides an ideal rapid control prototyping and hardware-in-the-loop development environment, ideal for design, simulation, implementation, and testing of time-varying systems used in communications, controls, signal processing, video processing, image processing, and more.

With its extensive collection of features and capabilities, QUARC presents a rapid control prototyping environment adaptable to virtually any mechatronic interface and scalable for complex multi-input and multi-output systems. This means you can reduce week-long development process to a few hours, turning overwhelmingly complex applications into feasible, even easy ones. The main features of QUARC include:

• **Simple and flexible hardware interfacing**
The QUARC HIL API provides a whole new way of accessing hardware, creating a flexible and extensible framework supporting DAQ and external devices. By changing a single parameter, you can change DAQ used by an entire diagram.

• **Support of multi-threaded, multi-rate and asynchronous models**
QUARC can be configured to automatically create multiple threads for multi-rate models, where Simulink diagram contains blocks with differing sample times.

QUARC also handles threads that run completely asynchronously and are aperiodic. Such asynchronous threads are useful for communications, for worker threads, or for idle-time processing.

• **Flexible and protocol-independent communication framework**
The QUARC Stream API allows communication between QUARC models; between a QUARC model and external third-party applications (e.g., GUI); or even between two external third-party applications.

The Stream API is independent of the development environment and can be used in C/C++, .NET, MATLAB®, or LabVIEW™. The Stream API also enables the communication between multiple real-time models over the internet. This could be used for distributed control, teleoperation, device interfacing, etc. The Stream API natively supports TCP/IP, UDP, serial, shared memory, named pipes, ARCNET, and other protocols.

• **Extensive third-party devices support**
Researchers incorporating off-the-shelf devices in their projects further benefit from QUARC's extensive suite of third-party device blocks. These blocks allow a Simulink model to communicate with external devices, such as Denso and Kinova robots, PGR cameras and GeoMagic haptic devices, and to implement the mathematical framework for controlling them. All this is possible without the need to learn new tools or hand code, since the controller design and integration are performed in an environment most researchers are familiar with, such as Windows®, MATLAB® and Simulink®.

For more features, check the specifications on page 4, or visit www.quanser.com/products/quarc

Image courtesy of Toronto Rehab, S.K. Advani

The Toronto Rehab's Challenging Environment Assessment Lab (CEAL) is the only facility of its kind in the world. This massive underground lab features the world's first hydraulic motion simulator that can mimic everyday environmental challenges faced by older people and those with disabling injury or illness. The CEAL's computer system was developed using QUARC software.
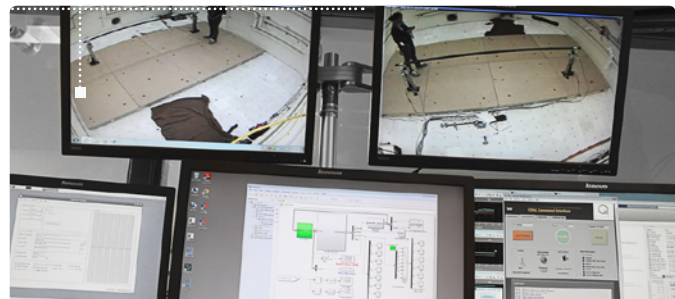
> " The Host Computer System for the Challenging Environment Assessment Laboratory (CEAL) at the Toronto Rehabilitation Institute (TRI) was developed using Quanser's QUARC real-time software. The power of QUARC, with Quanser's engineering support, enabled TRI to create a flexible development environment for researchers to implement sophisticated real-time experiments, using a large-scale 11 ton, 6 DOF motion platform and high performance audio-visual rendering systems. "
>
> **-Geoff Fernie,** Director, Toronto Rehab Institute, Canada

User interface
- Full support for Simulink® external mode, including scopes, floating scopes, displays, to workspace, online parameter tuning, etc.
- Support for model referencing, allowing modular object-oriented model design and incremental compilation and linking for accelerating the development of large models
- Support for scope triggering and data archiving for sophisticated, large volume, data collection
- Log data to MAT-file and M-files

- Simulink Coder code optimization support
- Performance diagnostics, e.g., measure of sample time, computation time, elapsed time, etc.
- Console for monitoring standard I/O (stdio) of a model even on a remote target
- Ability to interface controllers with MATLAB GUIs, LabVIEW panels, Java GUIs, Altia, etc.
- Visualization blockset for 3D visualization of simulation or actual hardware in real-time

Simple and flexible hardware interface
- Unified, expandable data acquisition architecture supporting cards from Quanser, National Instruments and other manufacturers
- Single or multiple PC/ board configurations supported

- Support for multiple targets (OS's and chipsets) from a single Simulink diagram
- External Hardware-In-the-Loop card interfacing provided in C/C++, MATLAB, and .NET languages

Flexible and protocol-independent communication framework
- Full set of data streaming blocks to enable any synchronous or asynchronous communication with processes outside of the running controller
- Native support of TCP/IP, UDP, serial, shared memory, named pipes, SPI, I2C, Peak CAN, ARCNET and other protocols

- Extensible framework enabling distributed control, device interfacing, teleoperation and general interprocess communication between controllers and local or remote applications
- External communication interfacing provided in C/C++, MATLAB, and .NET languages

Support of multi-threaded, multi-rate and asynchronous models
- Multi-processor support for improved sampling rates and performance
- Control of thread priorities and CPU affinity
- Run multi-threaded and multi-rate models
- Support for asynchronous threads in Simulink® models, ideal for asynchronous communications, etc.

- Run more than one model on a single target or multiple targets at the same time, locally and/or remotely
- Standalone controller execution
- Self-booting target for embedded modes

Extensive third-party devices support
- Support for third-party devices ranging from infrared cameras to Denso, Kinova and KUKA robots

- Support for peripherals such as mouse, keyboard and force feedback joysticks
- Host speech recognition and synthesis

Licensing options
- Network-administered license option

- Single- or multiple-user installation
- Multiple blocksets available

QUARC 2018 SYSTEM REQUIREMENTS

Target Types	Host Operating System	Compiler Tools	MATLAB, Simulink, MATLAB Coder, Simulink Coder, Control System Toolbox (required for labs)		
			R2017a	R2017b	R2018a
32 or 64-Bit Windows	Windows 7 SP1, Windows 8.1, or Windows 10	Microsoft Visual Studio 2017 Professional Edition or Community Edition with Visual C++	N/A*	✅	✅
		Microsoft Visual Studio 2015 or 2013 Professional Edition with Visual C++	✅	✅	✅
		Microsoft Visual Studio 2012 Professional Edition with Visual C++ or Microsoft Windows SDK 7.1	✅	N/A	N/A
Linux Embedded	Windows 7 SP1, Windows 8.1, or Windows 10	Cross-compiler Supplied	✅	✅	✅

* MATLAB R2017a requires patches supplied by Mathworks in order to use Microsoft Visual Studio 2017 compilers.

V5