UCONN
SCHOOL OF ENGINEERING

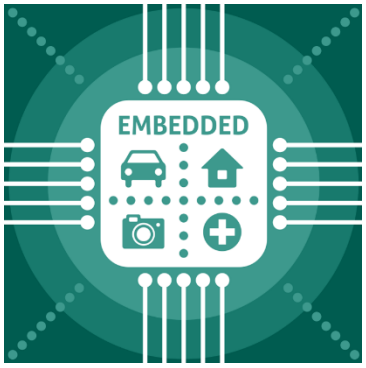
United Technologies Corporation
Institute for Advanced Systems Engineering
UNIVERSITY OF CONNECTICUT

Graduate Courses in Systems Engineering

SE 5201 Embedded/Networked Systems Modeling Abstractions

What's Exciting About this Course? Familiarize with design flows used in industry for designing, implementing and verifying embedded systems, and learn skills necessary to specify requirements and perform platform-based design, analysis and modeling of embedded and networked systems.

Course Description. Students will become cognizant of the role of embedded controllers and devices in the system design process, as they relate to event-driven and data-driven systems, and supervisory control of hybrid (continuous and discrete-time) systems. This will include exposure to platform-based design principles with an emphasis on requirements capture and refinement to platform architecture mapping, analysis and verification. Students will learn the technical aspects of modeling principles relevant to embedded systems – specifically modeling system architecture, system functions, computation, software, real-time systems, and distributed systems. Use of software engineering tools (Rhapsody, Simulink, Stateflow and Simulink/MATLAB coder) in the embedded system design flows is emphasized.



Course Outcomes

- Learn what embedded systems are, what is desired and what can typically go wrong in embedded system design and implementation.
- Understand how to formulate and model embedded system requirements.
- Learn how to analyze and map requirements into embedded system architectures.
- Learn how to model system architectures, including heterogeneous systems, using a system modeling language, such as SysML for architecture analysis and design.
- Understand fundamental principles of finite state machines and their use in modeling embedded systems for time-critical, event-driven and data-centric systems.
- Learn the principles of modeling computation and functional units.
- Learn the principles of object and software modeling (using UML) and automatic code generation.
- Learn the basic concepts of real-time operating systems and real-time task models.
- Learn basic concepts of distributed systems modeling.

Topics: CTL and LTL Model Checking, Abstract Interpretation, Black-box testing, Switched Systems, Symbolic and Numerical Model Checkers for Timed and Hybrid Systems, Design Flows for Embedded System Design, Implementation & Verification, Embedded Systems Requirements Capture and Architecture Selection, Functional unit modeling methods and tools, software modeling and code generation, real-time architectures and operating systems, distributed system modeling.

Course Objectives and Links to Overall Program Goals

Students can model and integrate system elements into more reliable networked cyber-physical systems. With the demand for increasing levels of complexity in systems, this course prepares engineers to design embedded systems that fulfill stakeholder needs by conforming more closely to system specifications.