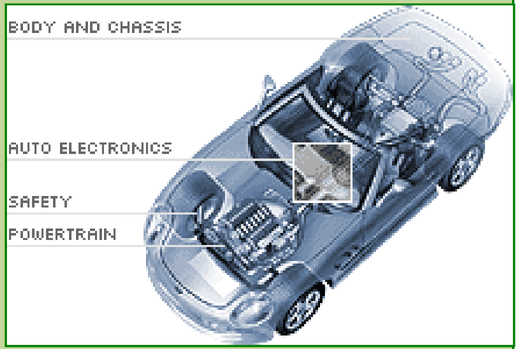

EE 249 Embedded System Design: Models, Validation, and Synthesis

Location: 521 Cory Hall Time: TuTh 11-12:30PM (Lecture) Tu 5-6 Discussion Th 4-6PM (Lab)
Instructor: Prof. Alberto Sangiovanni-Vincentelli (alberto@eecs.berkeley.edu)
GSI: Liangpeng Guo (glp@eecs.berkeley.edu)
CCN: 25830, 25833 Units: 4

DESCRIPTION: Embedded systems are electronics systems that sense physical quantities, elaborate the data and respond to the environment by sending commands to actuators. These computing systems are increasingly becoming a necessity in our everyday lives, from complex automobile electrical systems to high-performance building temperature and power control systems. New design methods are needed to efficiently deal with the growing design complexity and heterogeneity of these systems. This class presents approaches to the new system science based on theories, methods and tools that are in use in industry and were in part developed at the Berkeley Center for Hybrid and Embedded Software Systems (CHESS), the Giga-scale System Research (GSRC) and MuSyc Centers where heterogeneity, concurrency, multiple levels of abstraction play an important role and where a set of correct-by-construction refinement techniques are introduced as a way of reducing substantially design time and errors. Real-life applications including car electronics and building automation are used to illustrate system-level design methodologies and tools.

Course Topics	
Part 1: Introduction	Design complexity, Example of embedded systems, traditional design flow, Platform-Based Design
Part 2: Design Capture and Entry	Formalisms and tools, DOORS
Part 3: Functional modeling, analysis and simulation	Introduction to models of computation. Finite State Machines and Co-Design Finite State Machines, Kahn Process Networks, Data Flow, Petri Nets, Hybrid Systems. Unified frameworks: the Tagged Signal Model, Agent Algebra
Part 4: Architecture and performance abstraction	Definition of architecture, examples. Distributed architecture, coordination, communication. Real time operating systems, scheduling of computation and communication.
Part 5: Mapping	Definition of mapping and synthesis. Software synthesis, quasi static scheduling. Behavioral synthesis. Communication Synthesis and communication-based design
Part 6: Verification	Validation vs Simulation. Formal Verification of hybrid system. Interface automata and assume guarantee reasoning.
Part 7: Applications	*Automotive: car architecture, communication standards (CAN, FlexRay, AUTOSAR), scheduling and timing analysis Building automation: Communication (BanNet, LonWorks, ZigBee). Applications to monitoring and security*

GRADING: Grading for this class will be based on a final project, lab/HW assignments, and literature discussions.