MDD, Spring 2011

Bengt Jonsson

January 18, 2011
Syllabus:

- Modelling of Physical Environments (e.g., using Matlab/Simulink)
- Control Modelling (e.g., using StateFlow)
- Verification/Validation (e.g., by Simulation/Testing/Formal Verification)
- Architecture Modeling (e.g., using SysML):
  - Different Models of Computation (e.g., Data Flow Networks, Petri Nets, Hybrid automata)
- Case Studies and Tools:
Examination:

- 4 home assignments during period 3, solved individually
- Mid-term exam at end of period 3
- Project during period 4 (solved in pairs)

Current Information will be available at
http://www.it.uu.se/edu/course/homepage/modbasutv/vt11
Who are we:

- Bengt Jonsson (main instructor)
  - bengt@it.uu.se
  - http://user.it.uu.se/~bengt
  - Research interests: Modeling, testing, verification, etc. of parallel and distributed programs

- Assistant (not yet determined)
  (not yet, depends on administrator support)
Tools:

- For the Course, you must install and use MatLab/Simulink
- Available for students, See http://www.teknat.uu.se/student/matlab/
Literature and Resources:

- Control Tutorials for MATLAB and Simulink: http://www.engin.umich.edu/class/ctms/tem
Links to Related Courses on the Internet:

- More will appear on the course page
Models are used in all branches of Engineering

Why Model Based for Embedded Systems?
- Higher level of abstraction - more productive
- Code generation from model (sometimes)
- Platform independent - reuse
- Early Verification and Validation
- Documentation
What is Model-Based Design

1. Create a **mathematical model** of all the parts of the system
   - Physical world
   - Control system
   - Software environment
   - Hardware platform
   - Network
   - Sensors and actuators

2. Construct the implementation from the model
   - Automated code generation (sometimes)
   - Typically, many part of the code are created manually
Models for Software vs. Models for Control System

- Models for Software
  - UML, SysML, Timed Automata, AADL, StateCharts, ...
- Models for Control Systems,
  - Matlab, Simulink, Modelica, Targetlink, ASCET
Simulink

- Graphical “block-diagram environment for modeling and simulating dynamical systems on top of Matlab
- Introduced in early 90s
  - Continuous-time
  - Sampled data systems
  - Event-triggered systems: Stateflow
  - Composed behaviors
- Partly undevined behavior (hidden in Simulink engine)
- Many add-on tools
Modeling Software and Hardware

- **Structure**
  - Code, class diagrams, component diagrams (e.g., in UML)
  - System Architecture (e.g., in SysML)

- **Behavior**
  - Sequence diagrams, state machines, other structures (models of computation)