Simulink® for System and Algorithm Modeling

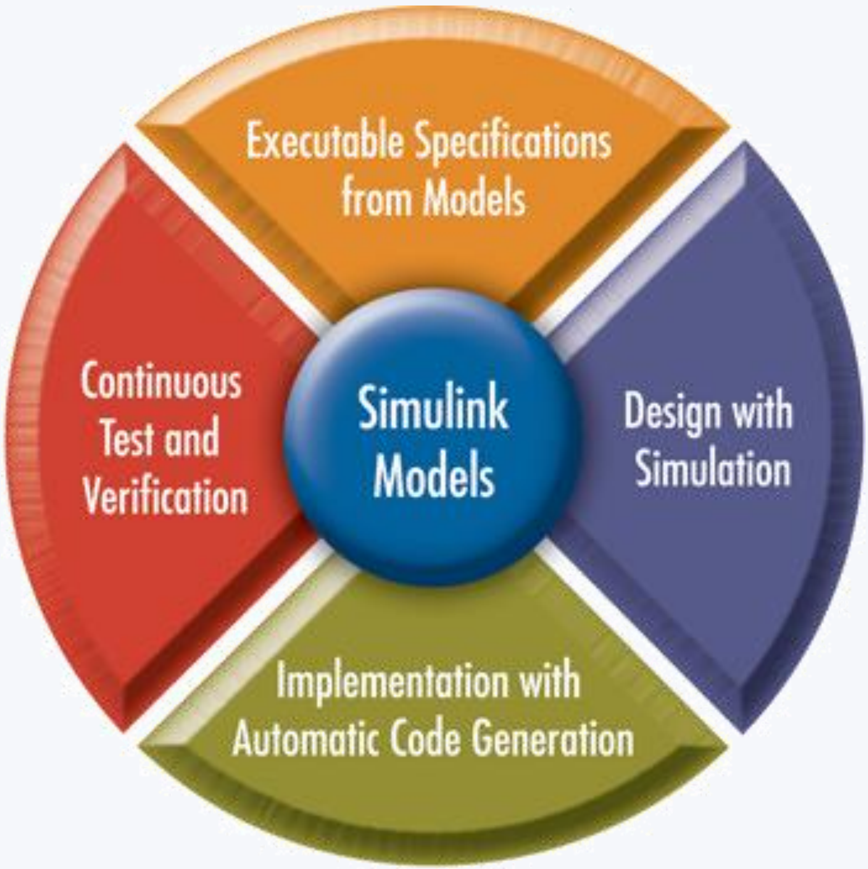
Introduction to System Modeling

The MathWorks
Training Services

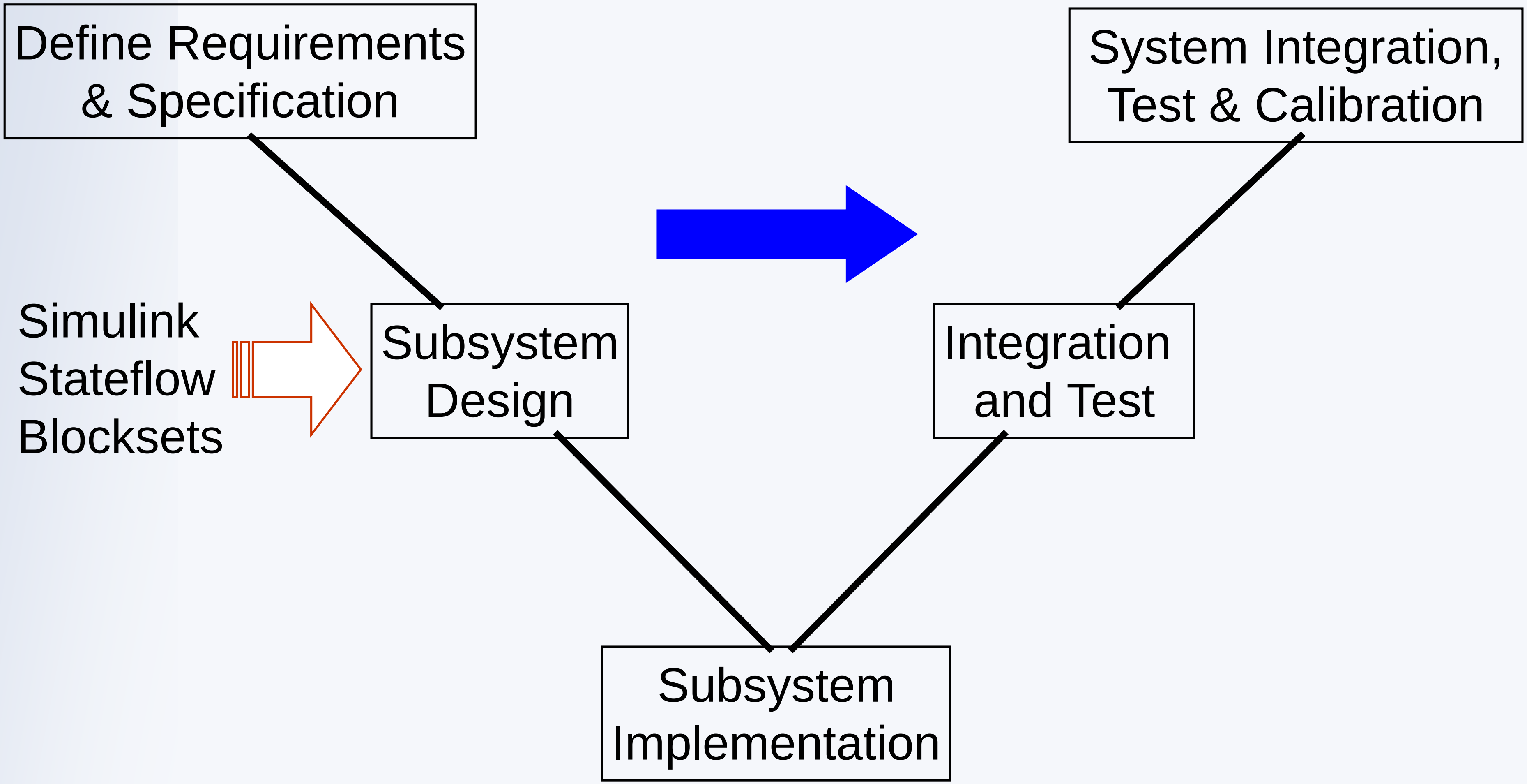
Outline

- Model-Based Design
- Types of modeling
- System modeling with Simulink
- Modeling steps

Model-Based Design with Simulink



System Design Process



Types of Modeling

- First-principles modeling
  - Simulink
- Finite-state machine
  - Stateflow
- Physical modeling
  - SimMechanics, SimPowerSystems, SimDriveline
- System identification
  - Neural networks, system identification

System Modeling with Simulink

- A graphical environment for hierarchical block diagram development
- Supplied with libraries of basic and compound blocks for general system buildup
- Supplied with functions that automate model constructions and simulations
- Extensible for custom feature development and distribution
- Supports modeling of continuous-time, discrete-time (including single-rate, multirate, and asynchronous) and hybrid systems
- Supports integration of custom and legacy code

Simulink Environment

- Simulink Library Browser contains basic and add-on blocks.
- Simulink block diagram editor facilitates the buildup of a model or subsystem block diagram.
- The Simulink solver engine steps the model in time.
- The solver engine propagates signals block by block after each update.
- Simulink manages the interaction between the model and the solver engine during a simulation.

Simulink Dependency on MATLAB

- Simulink depends on the MATLAB workspace to define and evaluate model and block parameters.
- Simulink depends on the MATLAB workspace to define model inputs.
- Simulink can use the MATLAB workspace to store model outputs for analysis.
- Simulink can integrate calls to MATLAB operators and functions in models.

Simulink Add-Ons

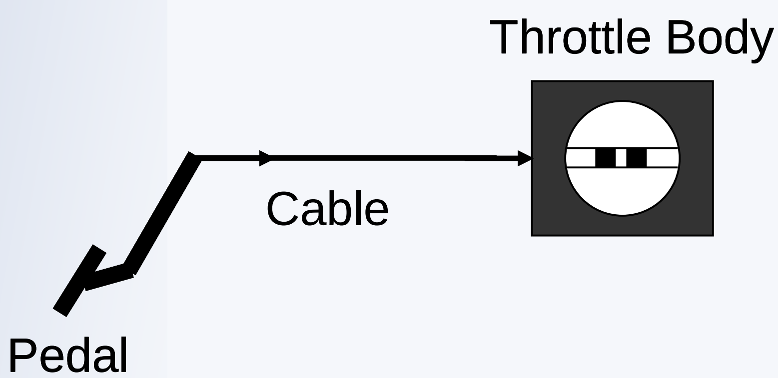
- Application-specific features via add-on blocksets
- Complex flow charts and state machine design environment via Stateflow
- Automatic code generation from models/subsystems via Real-Time Workshop and its add-on targets to support
  - Rapid simulation
  - Rapid prototyping
  - Embedded design testing
- Automatic generation of model documentation or specifications via Simulink Report Generator

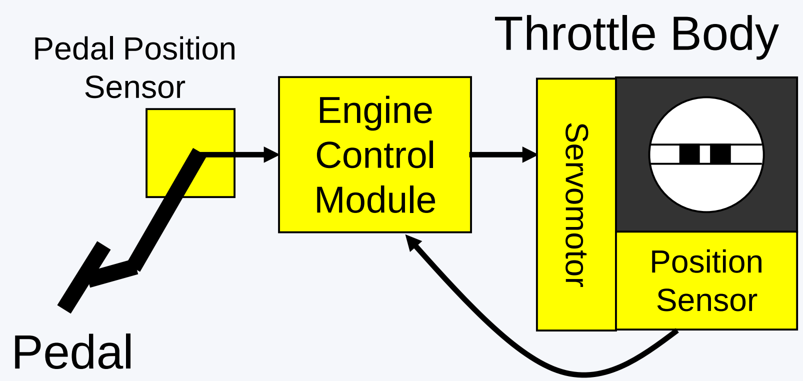
Modeling Steps

- Defining the system.
- Identifying the system components.
- Modeling the system with equations.
- Building a block diagram for the model using Simulink.
- Simulating the model.
- Validating the simulation results.

Defining the System

- Electronic throttle control

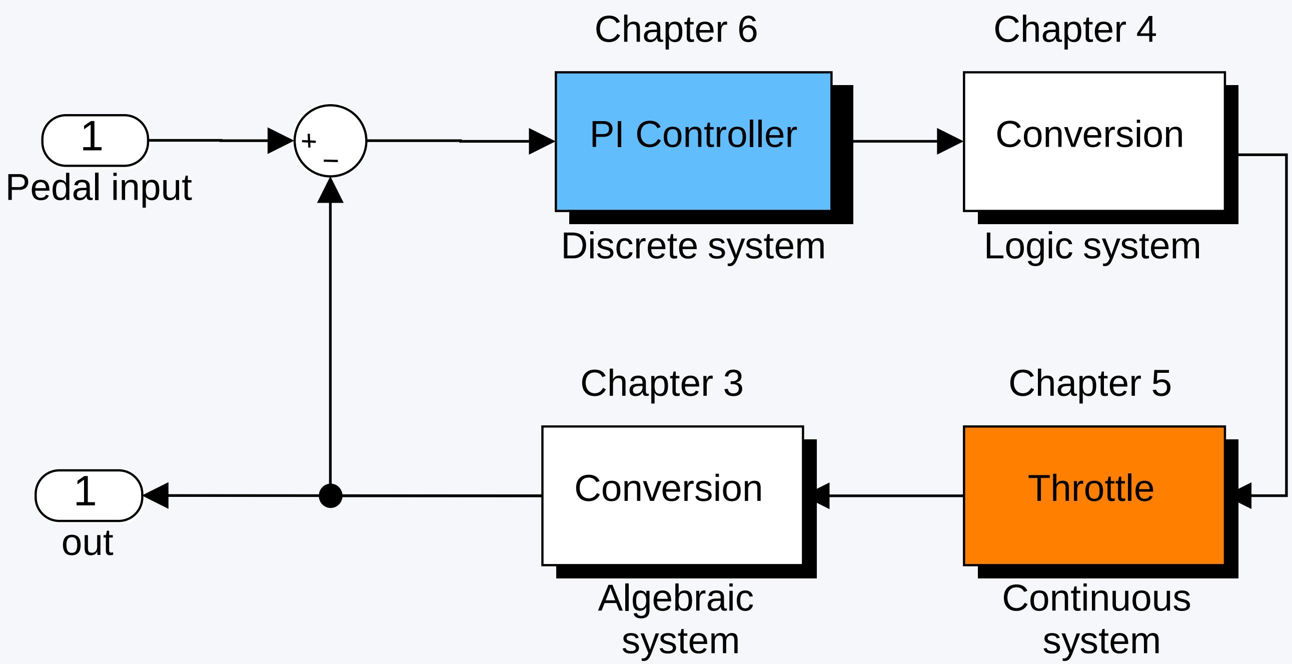




Traditional mechanical linkage throttle control

Electronic throttle control (ETC) – Throttle by wire

Course Layout

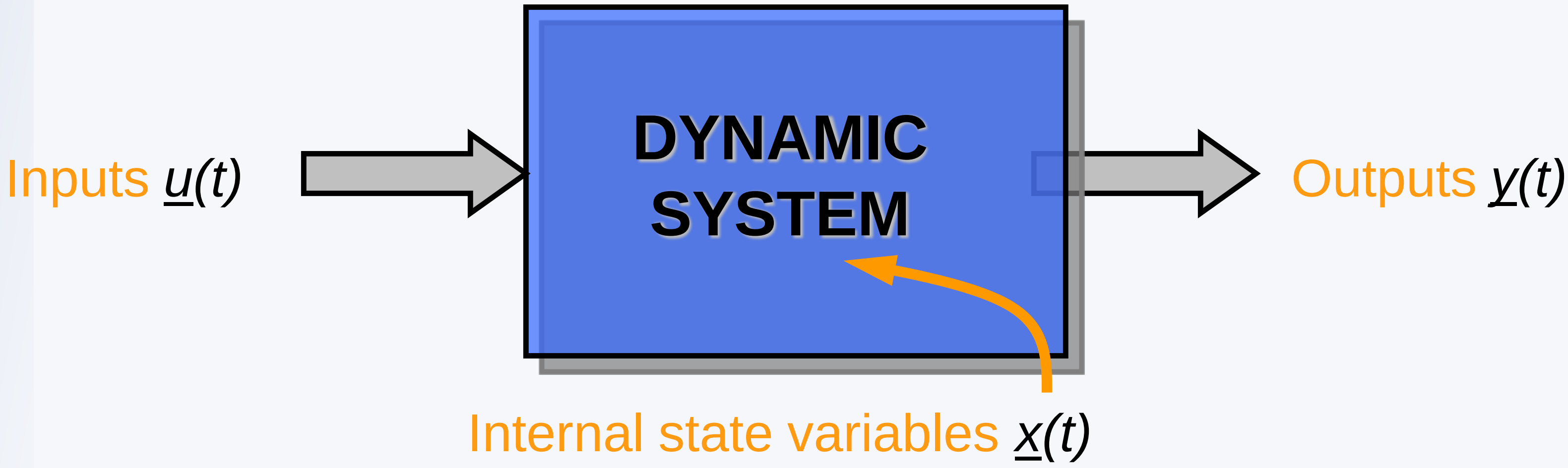


Identifying the System Components

- What are the input signals?
- What are the output signals?
- What are the intermediate signals?
- What are the parameters?
- What are the discrete states?
- What are the continuous states?

- For the electronic throttle controller you have
  - One input — the pedal position
  - One output — the throttle angle
  - Two continuous states — throttle angular position and velocity
  - One discrete state coming from the discrete integrator in the PI controller
  - A number of intermediate signals and parameters

Overview of a General Dynamic System



$$\dot{x}_c(t) = f_c(x,u,t)$$ ← Continuous Dynamics

$$x_{k+1}(t) = f_d(x,u,t)$$ ← Discrete Dynamics

$$y(t) = q(x,u,t)$$ ← Output Equation

Summary

- Model-Based Design
- Types of modeling
- System modeling with Simulink
- Modeling steps