Building Blocks for Embedded Control Systems

About...

Developing embedded control systems using a building-block approach at all the parts enables an efficient and fast design process.

Requirements
- Reusability for competitive fast development
- Well-defined interfaces, separation of interface and body
- Efficient maintenance
- Simulate-ability
  - Checking alternative solutions during design
- Flexibility in design
  - Hardware-/Software Co-design

Embedded System
- Three parts
  - Embedded Computer & Software
  - Interfacing towards appliance
  - Appliance itself
- Embedded Control Systems
  - Dynamic behavior of appliance is essential: closed loop system
- Embedded Data Systems
  - Behavior of appliance described as waiting times between subsequent commands from the software

Software Building Blocks

Data Flow Diagrams
- Vertices: Processes
- Edges: Interprocess communication

Formalism
- Communicating Sequential Processes
- Formal process algebra CSP
- Language support, Runtime support, Tool support

Channel communication
- Synchronization
- Rendezvous, synchronous, unbuffered
- Asynchronous via buffers
- Scheduling

Encapsulates thread programming

CTJ Library
- Communication Framework
- Synchronization, Scheduling, Data transfer
- Mapping SW on HW topology
- Encapsulation of hardware dependency
- Implementation
  - Java, C++, C

Appliance Building Blocks

Bond graphs
- Vertices: submodels, (dynamic) behavior
- Edges: ideal exchange of energy

Formalism
- Physical-domain independent
due to analogies on physical level
- Limited amount of basic elements (9)
- Combines with block diagrams

Vertices (submodels)
- Encapsulation granted:
  - Ports as interfaces
  - Sets of two variables, product is power
- Equations specified as real equations (not as assignments)

Edges (bonds)
- Energy flow (modeling)
- Bilateral signal flow (analysis, simulation)

Compilation to algorithm for simulation

20-sim
- Modeling and simulation package

Case

Industrial Robot with digital controller
- Robot
- Two revolute joints, one translational
- I/O
- Three servo motors, one per axis
- Software

Basic single-axis digital controller

Software Building Blocks

HWSW Appl

Software

Computer & Software

Appliance

Appliance

I/O

Computer & Software

Appliance

I/O

SW

Appliance

I/O

Appliance

Software Building Blocks

Appliance Building Blocks

Simulation and ECS Implementation

ECS Design Trajectory

Physical System Modeling
- Dynamic behavior object-orientedly modeled
- Bond graphs
- Control Law Design
- Model based
- Physical system model or simplified version
- Embedded Control System Implementation
- Control laws into computer code
- Stepwise refinement
- Realization
- Hardware in the loop simulation / testing
- Concurrent engineering

Stepwise refinement
- Control laws only
  - Implementation assumed ideal
- Non-ideal Components
  - Adding relevant dynamic effects
  - Safety & Command Interfacing
    - External commands
    - Maintenance
    - Non-ideal Computer hardware
      - Computational latency
      - Accuracy
      - Scheduling / optimization techniques

Embedded Control System Implementation

Validation and Testing

Realization

Verification by Simulation

Control Law Design

Control System Design

Embedded Control System Implementation

Verification by Simulation

Physical System Modeling

Computer & Software

Appliance

I/O

SW

Appliance

I/O

Appliance

I/O

Appliance

SW

Appliance

I/O

Appliance

SW

Appliance

I/O

Appliance

I/O

Appliance

I/O

Appliance

I/O

Appliance

Verification by Simulation

Validation and Testing

Realization

Hardware in the loop simulation / testing

Concurrent engineering

Jan F. Broenink
Gerald H. Hilderink
André W.P. Bakkers

Drebbel Institute
Control Laboratory at Electrical Engineering
University of Twente
P.O. Box 217, 7500 AE Enschede
E-mail: J.F.Broenink@el.utwente.nl
http://www.rt.el.utwente.nl