



V3 - Design with SystemC

System Design and Simulation with SystemC

Objectives

- Understand the benefits of system simulation with SystemC
- Mastering the different levels of modeling
 - Transactional Models
 - Software models
 - Hardware models
 - Synthesizable models
- System modeling:
 - UTF (UnTimed Functional model)
 - TF (Timed Functional model)
- Hardware modeling:
 - BCA (Bus Cycle Accurate model)
 - PCA (Pin Cycle Accurate model)

This course is complemented by numerous exercises and describes SystemC version 2.2

Course material

- A PC workstation per trainee group with:
 - Visual Studio 2005
 - SystemC-2.2

Prerequisites

- Basic knowledge of C++ (see for example our [L3 - C++ course](#))

Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

Course Outline

First day

Reminders on C++

- Object Oriented Programming
 - Classes and objects
 - Attributes
- Methods and operators
 - Overloading
 - Constructors and Destructors
 - Virtual methods
 - References
 - Default parameters
- Memory management
 - The new and delete operators

- Name spaces
- Standard input/output (Streams)

Advanced features of C++ needed by SystemC

- Class and function templates
 - Template definition
 - Constraints
 - Automatic instantiation
 - Manual instantiation
- Type conversions
 - Implicit conversions
 - User-defined conversions
 - Copy and initialisation operators
 - Up casts and down casts
- Exceptions

Second day

Introduction to SystemC

- The basics of SystemC
 - Language objectives
 - History
 - Advantages and disadvantages of SystemC
- Transaction Level Modeling (TLM)
- The SystemC design flow
 - Algorithmic model
 - TLM model
 - Hardware/software partitioning
 - Direct synthesis or HDL translation
 - Model simulation
- The SystemC architecture
 - Communication channels
 - Structural elements
 - Data types
 - The simulation engine

Eléments de base du langage SystemC

- Structural elements
 - Modules, Ports and Signals
 - Primitive Channels
- Creating model structure
 - Instantiating Modules
 - Connecting ports
- Processes and Time Management
 - Methods and Threads
 - Events
 - Static or dynamic sensitivity
 - Time and clocks
 - Dynamic processes

Third day

Simulation of a SystemC model

- Starting and stopping the simulation
- Model elaboration
 - Static elaboration phases
 - Dynamic elaboration phases
 - The event finder concept
 - Elaboration callbacks
- The simulation phases
 - Event notifications
 - Waiting on events and triggers
 - Event queues
- Debug techniques
 - Reporting and tracing
 - Error handling
 - Tracing hidden signals and local variables

Fourth day

Bus and Pin Accurate Models

- Modeling busses
 - Interfaces and communication channels
 - Master and slave interfaces
 - Interface methods (blocking and non-blocking)
 - Using events with channels
 - Channels with dynamic sensitivity
- Modeling multi-port busses
 - Port binding policies
- Pin Accurate Models
 - Fully specified data types
 - Assignment and truncation
- Logical types and vectors
 - Selecting bits and slices
 - Concatenating values
 - Resolving types
- Integer and fixed point types

Modeling by refining models

- Refining algorithms
 - Creating UnTimed Functional (UTF) models
 - Refining to Timed Functional (TF) models
 - Partitioning hardware and software
 - Adding timing annotations
- Refinement policies
 - Refining structure
 - Refining data
 - Refining communications
- Channel refinement
 - The adaptator concept
 - Building an adaptator
 - Creating a specialized event finder