

Possibility of SystemC code generation from SDL specification

Pavel Morozkin

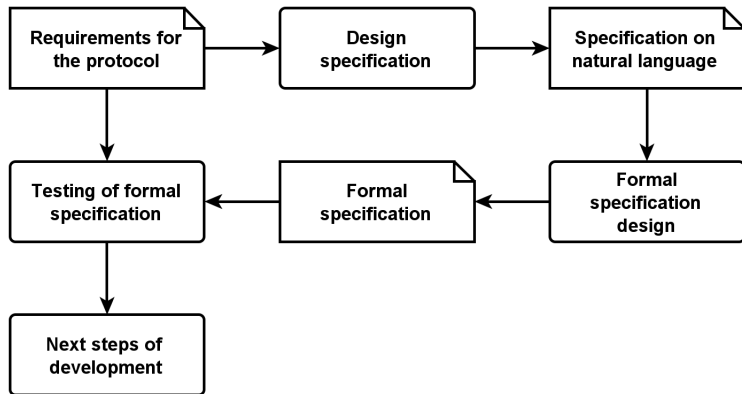
SUAI | 11th FRUCT conference

April 26, 2012

Agenda

- Network protocol design
- SDL language and its features
- Purpose of SDL specification
- Obtaining an executable SDL model
- SDL model testing
- Structure of SDL/SystemC model
- Advantages and disadvantages of SDL/SystemC co-simulation
- Proposed solution overview
- Obtaining an executable SDL model due to solution
- SDL parser, AST tree, CBCS structure
- Example of SDL model and its several representations
- Actual tasks
- Conclusion

Network protocol design



From requirements for the protocol to testing of formal specification.

SDL language and its features

- Specification and Description Language
- targeted at the unambiguous specification and description of the behaviour of distributed systems
- defined by the ITU-T (Recommendations Z.100 to Z.106)
- provides both a graphical Graphic Representation (SDL/GR) and textual Phrase Representation (SDL/PR)
- based on Extended Communicating Finite State Machines (ECFSM)
- main terms of language – system, block, process
- current version of language – SDL-2010

Purpose of SDL specification

The purpose of SDL specification is unambiguous description of the real system in form of formal model.

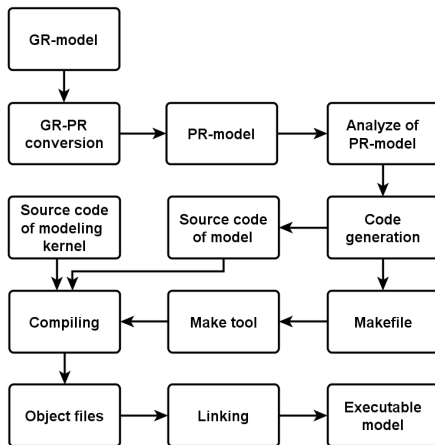
Design of SDL specification:

- description of structure of system
- description of functioning components
- description of communication components (channels, signal routes)

SUAI is using IBM Rational SDL and TTCN Suite 6.3, which contain the next main tools:

- graphical SDL editor
- GR-PR converter
- PR analyzer
- source code generator

Obtaining an executable SDL model



SDL model testing

Testing of SDL-model – the process of testing executable SDL-model generated from the original SDL-model. Executable SDL model is checked on the requirements of the protocol.

By specificity of SDL-models there is no single testing technology.

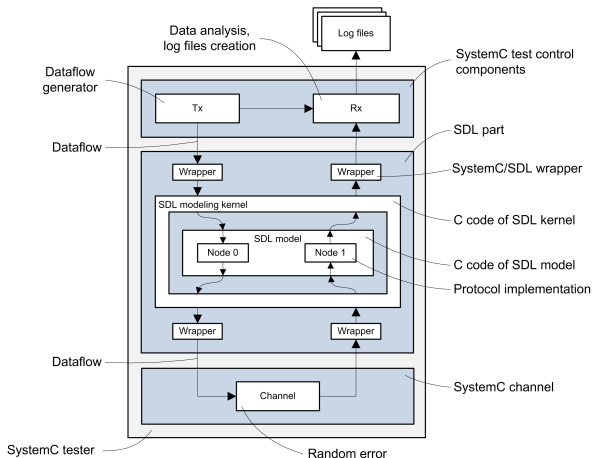
Technologies of testing of SDL-models normally developed at independent companies. Usually these technologies are private.

Recently a group of researchers from SUAI and NOKIA Research Center has proposed and successfully implemented testing method with SDL and SystemC languages co-modeling use.

This method involves:

- generation of source code of SDL model
- integration this source code with SystemC test environment

Structure of SDL/SystemC model



The SDL/SystemC-model is implemented as a single executable program.



Advantages and disadvantages of SDL/SystemC co-simulation

Advantages:

- flexible testing of SDL-model due to its integration with SystemC-test environment
- integration of SDL and SystemC models
- design of SystemC models containing a lot of components, which are described by the SDL language

Disadvantages:

- need for SystemC-SDL signal converters (wrappers)
- dependence on IBM Rational SDL and TTCN Suite SDL modeling kernel
- difficulty of reconfiguration SDL modeling kernel for the design of specific SDL models

Proposed solution overview

The proposed solution consists in development a technology of generation of executable SDL-models presented in the SystemC language.

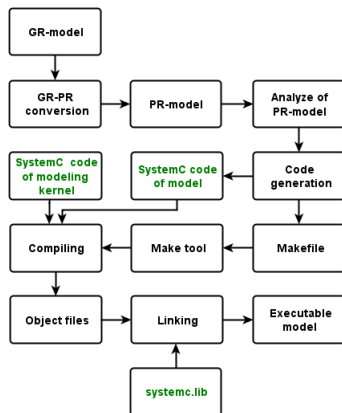
The solution allows:

- provide testing of SDL model by «soft»integration of SDL models with SystemC test environment
- develop specific mechanisms for SDL language required for design of SDL-models of network protocols
- develop new tools based on the new SDL reconfigurable modeling kernel

The need of new tools:

- SDL parser of PR models
- SystemC code generator
- SDL modeling kernel implemented on the SystemC language

Obtaining an executable SDL model due to solution



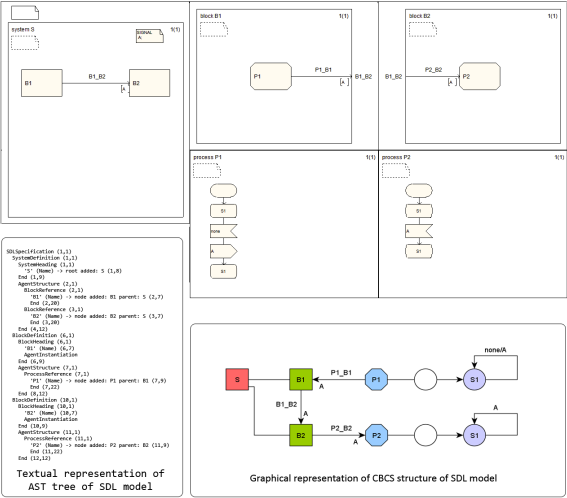
In this approach, there is a possibility of obtaining an executable SDL model (independent application) and the ability of integration the implementation (pure SystemC code) of SDL model with SystemC test environment.

SDL parser, AST tree, CBCS structure

Abstract:

- ANTLR parser generator (author – Terence Parr)
ANTLR takes as input a grammar that specifies a language and generates as output source code for a recognizer for that language.
- SDL parser based on the ANTLR (author – Michael Schmitt)
This tool analyses the PR-model and generates AST-tree.
- AST (Abstract Syntax Tree)
AST is a tree representation of the abstract syntactic structure of source code written in a some language.
- CBCS structure (Component, Behavior and Communication Structure)
The data structure that describes a hierarchy of models of entities and their functioning.

Example of SDL model and its several representations



Actual tasks and conclusion

The tasks, which are solving now:

- adaptation of Michael Schmitt SDL parser to the IBM Rational SDL and TTCN Suite by modifying the grammar of the SDL language
- the task of generation of CBCS structure from the AST-tree

Problem that requiring the solution:

- the task of developing SDL modeling kernel, presented in the SystemC language
- the task of generation of SystemC source code of SDL model from the CBCS structure

The proposed solution allows:

- test the SDL-model according to the existing technology of SystemC/SDL co-modeling (by integration of SDL models with SystemC test environment)
- use of reconfigurable SDL modeling kernel for further research
- develop a new SDL tools (for example, the graphical SDL editor), which allowing the use of advanced properties of SDL language (in future)

Thank you!

pavel.morozkin@guap.ru