System Level Modeling of Dynamic Reconfigurable System-on-Chip

Valentin Rozanov, Elena Suvorova, Nadezhda Matveeva, Alexey Rabin

Saint-Petersburg State University of Aerospace Instrumentation
valentin.rozanov@guap.ru, suvorova@aanet.ru, {nadezhda.matveeva, alexey.rabin}@guap.ru
What is reconfigurable system?

- Reconfigurability denotes system behavior
- Reconfigurable devices are able to adapt their hardware to application demands
- System behavior can be changed by reconfiguration parameters
Benefits of reconfigurable system

- Extend application field
- Good balance between implementation efficiency and flexibility
- Using in several applications and products for achieving low cost per chip
- Post-manufacturing programmability allows correct problems after fabric
- They can be used for adaptation under current application
Static and dynamic level of reconfiguration

Reconfigurable System-on-Chip

**Static**

- Simplest and most common approach
- Hardware resources remain static for the life
- It is possible on the Register Transfer Level (RTL) development stage

**Dynamic**

- Reallocate hardware at a runtime
- Circuits are loaded and unloaded
- Reconfigured dynamically during the operation of the system

Configure system RTL level

Execute system

Configure 1

Configure 2

… Configure N

Execute system
Types of reconfigurable systems

Reconfigurable System-on-chip

Static development of register-transfer level (RTL) model

- FCP technology reload bitfile
- Switch on/Switch off individual components (components and interconnection redundancy)

Dinamic during operation device

- Change components behavior
- Using the look-up tables (LUT)
- Libraries with reconfigurable logical components (NAND, NOR...)

in the start of operation or after reset

runtime

Change interconnection structure between components
Reconfigurable system implementation

**ASIC**
- Dynamically change the behavior of components with the look-up tables (LUT)
- Changing of the interconnection structure

**FPGA**
- Reallocate hardware at a runtime
- Provide very high degree of flexibility
- Dynamically change the behavior of components with configurable logic block (CLB)
- CLB consists of LUTs and others
SystemC and TLM

SystemC is a set of C++ classes and macros which provide an event-driven simulation interface.

The TLM-2.0 transaction level modeling standard involves communication between SystemC processes using function calls.

SystemC and TLM-2.0 advantages and disadvantages for system level modeling

**Advantages:**

- dynamic processes support
- function `sc_spawn` is used to create a static or dynamic spawned process instance
- `sc_mutex` and `sc_semaphore` may be instantiated dynamically
- process control member functions: `suspend` and `resume`, `disable` and `enable`, `sync_reset_on` and `sync_reset_off`, `kill`, `reset`, `throw_it`
- function `bind` of the standard port and socket classes has been made virtual

**Disadvantages:**

- does not support dynamic generation and elimination objects of class `sc_module`, `sc_port`, `sc_export`, or `sc_prim_channel` during a simulation
## Approaches for dynamic reconfigurable system simulation

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Change SystemC kernel</td>
<td>no</td>
<td>no information</td>
<td>yes</td>
<td>no information</td>
</tr>
<tr>
<td>Use TLM</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>Open access</td>
<td>yes, free library</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>
Reconfiguration in ASIC systems

1. Interconnection between components

2. Reconfigure functional blocks
   - components based on LUT
   - technology libraries of reconfigurable components
     (NAND, NOR, INV)
Dynamically reconfigured interconnection between components

**ASIC**
- Insert of selectors based on multiplexers or otherwise
- Selectors connect different lines to the same port at different times

**System level model**
- Using `multi_passthrough_initiator_socket` and target from TLM-2.0
- Provides binding of one or multiple source and one or multiple receivers
- Binding depends on the system configuration

---

Typical fragment based on ASIC

System level model with TLM-2.0 sockets
Dynamically reconfigured functional blocks

**RTL model**
- Ratio of the number/capacity of subcomponents could be dynamically changed to provide dynamical reconfiguration of component’s behavior
- carry signals can be used or not in each ALU depending on the configuration

**System level model**
- data received by the handler is represented as transactions with fixed size
- class `simple_target_socket` is used to transmit data transactions and configuration transactions
- one process divides received via `simple_target_socket` data to slices for every ALU
- second process assembles data slices from ALU to the output transaction
- processes operation according to configuration data

**Register-transfer level (RTL) model**

**System level model with TLM-2.0 sockets**
Dynamically reconfigured functional blocks

Finite state automata

- Idle
  - finish_pack = 1
  - start_packet = 1
- Work
  - finish_arb = 1
  - finish_pack = 1
- Arbitration
  - finish_pack = 1
  - start_packet = 1

Register-transfer level (RTL) model

- Input port controller
- Look-up table (SRAM block)
- Content
  - Reload_content
- Start_pack
- Finish_pack
- Finish_arb
- Fl_OK
- Wr_prio
- Wr_id
- Wr_addr
- Wr_fl

System level model with TLM-2.0 sockets

- Input port controller
- Look-up table (array C++)
- Content
  - simple_target_socket
- Sc_signal
  - Start_pack
  - Finish_pack
  - Finish_arb
  - Fl_OK
  - Wr_prio
  - Wr_id
  - Wr_addr
  - Wr_fl
Conclusion

• We consider advantages and disadvantages SystemC and TLM for tasks of dynamic reconfigurable SoC modeling.

• Dynamic Reconfigurable SoC can be realized as system level model with SystemC and TLM-2.0.

• Standard SystemC and TLM-2.0 classes, functions are enough for an implementation of a system level model for considered reconfiguration type.

• Present approaches for the development of the system level models of dynamic reconfigurable SoC implemented on ASIC.