High-Level System-on-Chip Simulator

Orlov Aleksandr
SUAI
67, Bolshaya Morskaya, 190000,
Saint-Petersburg, RUSSIA
alexanderorlov@live.ru
Microelectronic market

Consumer’s requirements are to get next production:
  – high-speed
  – reliable
  – small-sized
  – low-power

Manufacturer’s task is to market their products as quickly as possible.
New product development

Software development

Hardware development

March
April
May
June
July
August

T1
T2
Software development

- Software architecture
  - High-level implementation
  - Low-level implementation
    - X

- High-level simulator
- Hardware
  - X
High-level simulator

1. allows rapidly and with minimal cost to
   - inspect
   - set
   - test real system program-logic model without its building

2. provides to a user an opportunity to
   invoke and debug software for this model on it.
Requirements to simulator

• Provide an ability to model invoking high-level software presentation on the systems-on-chip model with an arbitrary configuration;
• Model the next hardware platform’s characteristics:
  ▪ delays of data packets sending and reception for each SoC device;
  ▪ delays during data packets transmission through channels;
  ▪ delays during program processes invoking on processing elements;
  ▪ delays for accessing common memory etc.
• Collect statistical information about a simulation process;
• Show textual and graphical information after a simulation process is finished.
Simulator’s structure

HIGH-LEVEL SoC
SIMULATOR

Core
DCNSimulator
Program on visual programming language VPL
VPL Language objects types

- Operators
- Data
- Links
- Structures (subprograms)
- References
Simulator’s structure

- File with hardware platform description
- File with program description

- Core DCNSimulator
- HIGH-LEVEL SoC SIMULATOR
Hardware configuration

- Channels
- Processing elements
- Routers
Hardware configuration

Configuration file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<platform>
  <devices>
    <pe name="PE1"></pe>
    <pe name="PE2"></pe>
    <pe name="PE3"></pe>
    <router name="Router1"></router>
  </devices>
  <channels>
    <channel src="Router1" dst="PE1"></channel>
    <channel src="Router1" dst="PE2"></channel>
    <channel src="Router1" dst="PE3"></channel>
    <channel src="PE1" dst="Router1"></channel>
    <channel src="PE2" dst="Router1"></channel>
    <channel src="PE3" dst="Router1"></channel>
  </channels>
</platform>
```
Program description file parsing
Hardware platform description file parsing
Hardware platform preparation for simulation

Routing table
1-Ch1->Ch2-2
2-Ch2->Ch3-3
....

DATA

PE1

Ch1

PE2

PE3

Ch2

Ch3
Simulator’s window
Statistic

Text file with common statistic about processes:

1500 ns: FuncProc 10005 is invoking on PE1 on iteration 1
1500 ns: FuncProc 10094 is invoking on PE3 on iteration 1
1500 ns: FuncProc 10020 is invoking on PE2 on iteration 1
1521 ns: IfProc 10157 is invoking on PE4 on iteration 1
2033 ns: FuncProc 10142 is invoking on PE5 on iteration 1
3000 ns: FuncProc 10020 is invoking on PE2 on iteration 2
3000 ns: FuncProc 10094 is invoking on PE3 on iteration 2
3000 ns: FuncProc 10005 is invoking on PE1 on iteration 2
3017 ns: IfProc 10157 is invoking on PE4 on iteration 2

Processing element occupation graphic.
CONCLUSIONS

• High-level simulator allows to model and debug high-level software representation for SoC model

It helps to solve the task of time reduction to market
THANK YOU FOR ATTENTION!