



FP7 – 216693 - MULTICUBE Project

MULTI-OBJECTIVE DESIGN SPACE EXPLORATION OF MULTI-PROCESSOR SOC
ARCHITECTURES FOR EMBEDDED MULTIMEDIA APPLICATIONS

Deliverable D2.1.2: Refined Performance and Power Estimation Prototype Tool Revision [1]

Delivery due date: M24 (December 2009)

Actual submission date: February 2nd, 2010

Lead beneficiary: UC

Dissemination Level of Deliverable		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	
Nature of Deliverable		
R	Report	
P	Prototype	X
D	Demonstrator	
O	Other	



Author(s):	<i>Héctor Posadas, Sara Real, Eugenio Villar (UC)</i>		
Reviewer(s):	<i>Gianluca Palermo (POLIMI), Marcos Martínez (DS2)</i>		
WP/Task No:	2.1	Number of pages:	18
Identifier:	D2.1.2_UC_12.2009	Dissemination level:	Public
Issue Date:	February 2 nd , 2010		

Keywords: Prototype, Open-Source, Performance Estimation, Power Estimation

Abstract: This deliverable (issued at M24) presents the results of the activity carried on from M13 to M24 in Task 2.1 (Open-source, performance and power consumption estimation framework) under the leadership of UC, with contributions from ESTECO and reviewed by POLIMI and DS2.

The main goal of this deliverable is the release of the refined version of the prototype tool for performance and power estimation developed in the project following the Multicube specifications. The prototype tool is called Multicube-SCoPE. The current, initial version of Multicube-SCoPE is composed of SCoPE v1.1.4 and the plug-in M3P v1.0.4. M3P is a SCoPE plug-in providing the Multicube required new features.

Additional work has been performed in SCoPE in order to support these new features, resulting in a new Multicube- SCoPE. Multicube- SCoPE refined prototype provides new functionalities to make more flexible the use of the tool. New component models as long as ways to introduce timing information of specific components obtained at other abstraction levels have been added to the tool. It allows reducing the simulation times and accurately modeling the Multicube use-cases.

This report contains:

- A summary of the structure and functionality of the refined version of Multicube-SCoPE.
- The current internet address of the web page from which the Multicube-SCoPE can be downloaded.
- A snapshot of the current Multicube-SCoPE web site content.
- Summary of improvements performed inside the tool.

Approved by the Project Coordinator:



Date:

February 2nd, 2010



Table of contents

I.	Executive summary (Upgraded)	4
II.	Link to the webpage of the prototype	7
III.	Website content.....	8
III. 1.	Home page	9
III. 2.	Download page	10
III. 3.	Installation page.....	11
III. 4.	Team page.....	12
IV.	Improvements to SCoPE internals (Updated).....	13
V.	M3P characteristics (Updated).....	15
V. 1.	Multicube-SCoPE as a Modeling Tool	15
V. 2.	Automatic System Model Building.....	15
V. 3.	Automatic System Configuration.....	16
V. 4.	Automatic Metric reporting.....	17
VI.	Conclusions.....	18
VII.	Glossary.....	18



I. Executive summary (Upgraded)

The main goal of this deliverable is the creation of an refined prototype tool for performance and power estimation following the Multicube specifications. The prototype tool is called Multicube-SCoPE. The current, refined version of Multicube-SCoPE is composed of SCoPE v1.1.4 and the SCoPE plug-in called M3P v1.0.4.

The prototype tool is based on a previous tool called SCoPE. SCoPE is a tool capable of simulating SystemC systems and providing some performance and power information. However, automatic system model generation and configuration, or automatic metric reporting are completely not supported by previous SCoPE versions. Furthermore, additional features must be added to model the use-cases proposed in Multicube.

The improvements performed internally in SCoPE to support the Multicube DSE flow are mainly related to automatically obtaining system metrics and simplifying system model construction. Additional support for debugging purposes and more flexible ways to integrated user-defined component models have been also included in the new version of the tool. These improvements have been developed in order to support the Multicube use cases. Memory space separation modeling for SW tasks and more powerful HW interfaces are some of these improvements. The additional work performed in SCoPE in order to support these new features has been included in a new SCoPE version: v1.1.4.

M3P is the SCoPE plug-in providing the additional facilities developed to support the Multicube required features. This plug-in provides the new services to support the Multicube requirements specified in D1.1 and D1.4.1. Furthermore, the plug-in allows using Multicube-SCoPE as a tool instead of as a C++ library.

The new Multicube-SCoPE framework presents the following features:

- **Dual use of Multicube-SCoPE as a tool or a library (New)**

The initial prototype version was developed as a library, not as a complete tool. As a consequence, no executable program was created when installing the tool. Each time an example was compiled, the library had to be linked together with the example-specific code in order to obtain an executable. The new refined version provides an example-independent Multicube-SCoPE tool executable. This reduces compilation times and providing a more flexible and user-friendly solution. Furthermore, the possibility of using the code as a library is still maintained.

- **Automatic system model generation**

The open source tool provides a XML input capable of creating the system models from XML descriptions. This capability allows dynamically creating the system models, and thus, dynamically supporting any system modification that want to be explored by the Design Space Exploration (DSE) tools. As a consequence, DSE process can be completely performed without compiling the simulation model, and thus reducing the exploration time.



- **Dual input for system descriptions (New)**

The refined tool has been prepared to accept system descriptions in two formats: as a SystemC code or in XML files. This extends the capabilities of the initial prototype, where only XML description was valid. To allow exploration of SystemC code based descriptions, the new feature of system configuration through global variables has to be used. In the Multicube project the SystemC code is used to model the power-line use case, while the XML files are used in the multimedia use case.

- **Automatic system configuration**

The tool provides a second XML input to easily modify the system configuration in order to perform the DSE process.

- **Exploration through global variables (New)**

The initial prototype tool allowed exploring systems by defining unfixed parameters in the XML system description file. The refined tool allows also exploring internal details of the system component models. These internal details are characteristics that are described in the model code, but without visibility in the XML system description file. To allow these exploration, the parameters must be declared as global variables in the code files and in the XML files. Furthermore, this technique allows exploring system described completely in SystemC code, not in XML files.

- **Memory space separation for system component models (New)**

The refined prototype tool is prepared to isolate the memory space of each component model. As the tool is developed as a host process running the SystemC kernel, only one memory space is provided for the entire simulation. Thus, duplicating names of global elements result in simulation errors. The isolation provided avoids conflicts when using the same names for functions or global variables in several component models. Furthermore, it allows instantiating the same component several times.

- **SW/HW communication through direct register memory addresses (New)**

In embedded systems HW/SW communication is usually performed letting the SW to access the memory address where the peripheral registers are mapped. These addresses are handled as pointers where data are read or stored. Previous versions of the tool, and in general SystemC-based solutions does not support this solutions, as it implies trying to access the peripherals of the host machine. This refined version provides the features required to support direct pointer access to HW peripherals from SW component models.

- **Time and power parameters in generic HW integration wrappers (New)**

Integration of HW components, both generic and user defined ones, is done in SCoPE by using generic Integration Wrappers. The use of these generic wrappers allows the automatic system model creation. Now, the wrappers also provide facilities to include time and power information for each component in an standard way. Thus, components



performance effects can be easily integrated in the XML system description file. This increases the accuracy of time and power metrics obtained with Multicube-SCoPE.

- **Integration of IMEC ARP library for reuse of timing information (New)**

The Atomium Record/Playback library of IMEC has been integrated in Multicube- SCoPE. This new feature allows storing data from a reference simulation in order to be re-used in other simulations. During the exploration process multiple simulations of the same code are done, considering different HW configurations. As the code executed is the same, performance information from one simulation can be reused in other simulations with compatible configurations. Furthermore, performance information obtained with other simulators at different abstraction levels can be used during the exploration process, increasing the accuracy of the process.

- **Automatic metric reporting**

The tool is capable of automatically reporting a list of system metrics, following D1.1. The metrics the user wants to be reported must be indicated through a XML file at the beginning of the simulation. When a simulation finishes, a new XML file with the results is automatically created.

- **Portability**

The tool is portable across a wide range of systems. This goal is achieved by not sacrificing the efficiency of the overall exploration engine. The standard ANSI C++ programming language is used for developing the open source framework. The Standard Template Library as well as other open source libraries are used during the development process.

- **Multiple executions**

The tool has been adapted to allow multiple concurrent executions in different directories with complete independence among them. This feature allows the exploration tool to run several instances of the simulation to speed up the exploration process.

- **Additional and improved HW models (New)**

Additional models, as a new, more accurate model of an AMBA bus has been developed. Furthermore, component models as DMAs or memories have been improved.

Multicube-SCoPE is distributed under the GPL license,.

This report is structured as follows. Section 2 presents the link to the website from which Multicube-SCoPE can be downloaded. Section 3 contains a summary of content of the website where the source code, license and documentation can be found. Section 4 presents a brief description of the main modification performed in the SCoPE tool. Finally, section 5 describes the features of the M3P plug-in.



II. Link to the webpage of the prototype

The public website of the Initial Prototype of the Open-Source MULTICUBE Exploration Framework can be reached by following the appropriate link on the MULTICUBE official website under the open-source tools page:

http://www.multicube.eu/open_source.html

Alternatively, the initial prototype of the Open-Source *MULTICUBE-SCoPE* can be reached directly with the following link:

<http://www.teisa.unican.es/gim/en/scope/multicube.html>

The following figure is a snapshot of the home page:

Multicube-SCoPE

Welcome to the **Multicube-SCoPE** project website! This project has been developed by GIM-TEISA department at the University of Cantabria (GIM-TEISA-UC) and funded by the FP7 EU Project MULTICUBE. This FP7 EU project aims at the development of a design exploration flow covering several abstraction levels based on the application of several *multiobjective design space exploration* techniques by means of a multi-objective exploration engine and multilevel simulation tools to achieve automatic architectural (hardware) and run-time (software) optimizations of MPSoCs.

Multicube-SCoPE is an evolution of SCoPE oriented to **Design Space Exploration (DSE)**. Multicube-SCoPE is a fast system performance analyzer capable of supporting dynamically configurable XML system descriptions. Multicube-SCoPE automatically provides the system metrics required to drive the DSE process. Multicube-SCoPE is the result of combining two elements: the new **SCoPE** version (v1.1.4) and the **M3P** plugin for SCoPE.

The diagram illustrates the system architecture. It shows an **Exploration Architect** interacting with a **Design Space Exploration Tool** via a **Human Machine Interface**. The **Design Space Exploration Tool** is connected to **XML Design Space**, **XML System Config**, and **XML System Metrics**. The **XML Design Space** is provided by a **Use Case and Simulator Provider**. The **XML System Config** and **XML System Metrics** are processed by the **M3P** plugin, which then interacts with the **SCoPE** (XML Use Case) component.

M3P, is an *open source* plug-in developed by the GIM-TEISA-UC to integrate SCoPE within the MULTICUBE project design flow. It is the extension of the SCoPE System-level Simulator which allows the simulation tool to communicate with a *multicube Explorer* performing the DSE using the XML interface defined within the MULTICUBE project to exchange system configuration and system metrics data.


Funded by FP7 Project:

 Developed by: GIM-TEISA UNIVERSITY OF CANTABRIA

Figure 1: Screenshot of the home page of the public website of Multicube-SCoPE

III. Website content

The website is composed of the following web pages:

- Home
- Source
- Installation
- Developers

The following sections describe in detail the content of each page with a small summary highlighted in *italic*.



III. 1. Home page

This page is the welcome page for the Multicube-SCoPE Open Source Tool. It gives a brief overview of the tool goals and features and provides information on the people involved in the development.

The *home page* introduces Multicube-SCoPE tool. This tool represents the extension of the previous tool SCoPE for Design Space Exploration. Multicube-SCoPE is a combination of the new SCoPE version (v1.1.4) and the M3P plug-in.

Then, the web introduces the MP3 plug-in and how it fits within the MULTICUBE exploration framework as the piece of software that enables the SCoPE system level simulation communicate with the design space simulation tool following the XML interface definition established by the MULTICUBE project to exchange system configuration and metrics results. The following figure shows the MULTICUBE design space exploration and simulation framework highlighting the M3P and SCoPE pair as the core of this deliverable.

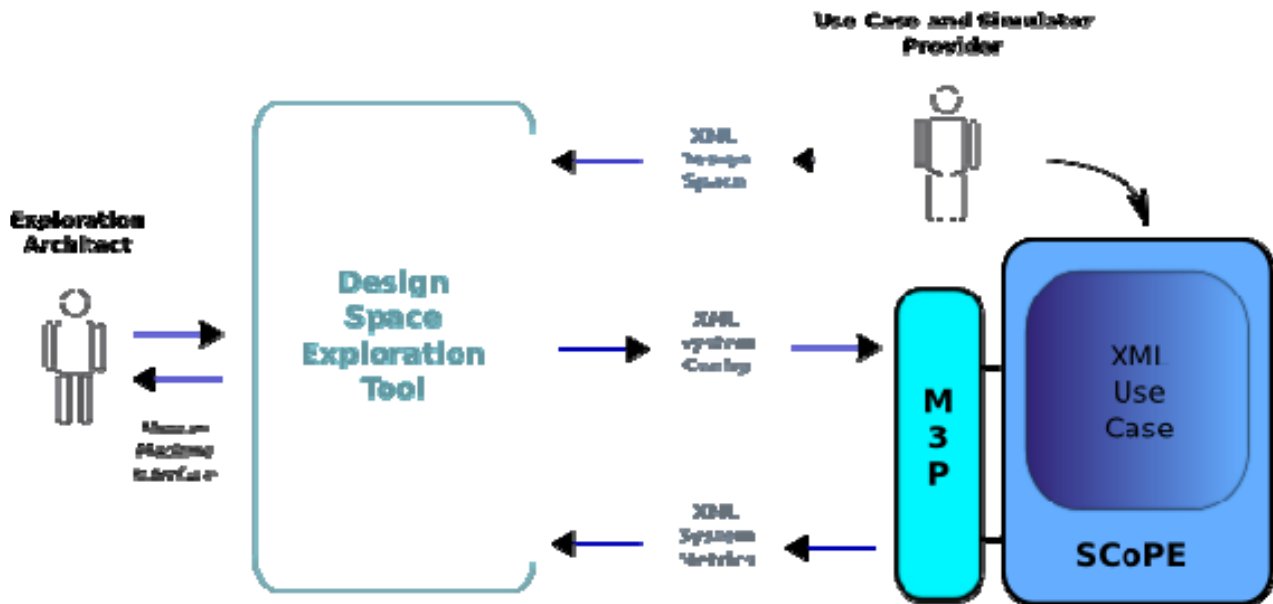


Figure 2: Screenshot of the home page of the public website of Multicube-SCoPE Simulator.

In addition to the introductory information stated above, the home page also emphasizes the funding and objectives of the FP7 EU MULTICUBE project to give a proper context and credits to the Multicube-SCoPE tool introduction.



III. 2. Download page

This page provides the link to download the elements that compose the initial prototype of the Multicube-SCoPE open-source performance and power estimation tool for DSE.

The source page briefly describes all the source code developed in this deliverable and gives the instructions to download the Multicube-SCoPE components: SCoPE and M3P. A pack with all the Multicube-SCoPE components can be downloaded from the web page.

Furthermore, links for downloading the Multicube-SCoPE components separately are also reported. The M3P plug-in sources and documentation can be downloaded directly from this web page.

A link to the SCoPE website is provided together.

At the official SCoPE web page can also be found the documentation required to understand how to use the tool. This documentation includes the user manual and the developer API reference manual.



Figure 3: Screenshot of the download page of the public website of Multicube-SCoPE Simulator.

The download page includes a link to an auto-installation script which allows the user to automatically install and configure the M3P plug-in and example together with the SCoPE tool thus avoiding the somewhat painful installation and configuration steps that must be followed if performing manually the whole installation process with each piece of software.

Latest stable release of M3P:

December 22th, 2009 - Release 1.0.4



III. 3. Installation page

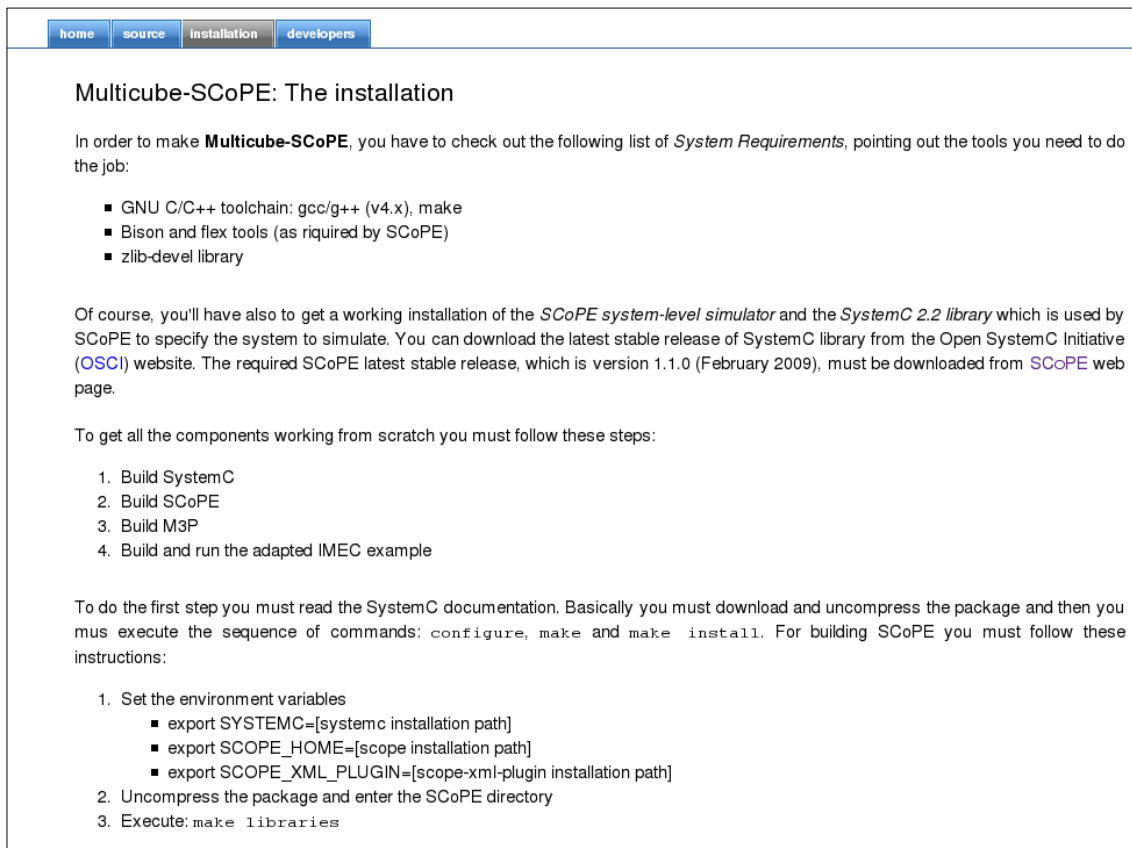
This page shows how to install the Multicube-SCoPE tool.

This page provides the instructions to configure and install all the elements needed to successfully use the Multicube-SCoPE tool. The page starts by indicating all the system requirements that must be checked out before trying the main configuration and installation steps. The GNU C/C++ tool chain, bison and flex analyzers and SystemC library are the system prerequisites.

After checking out that all the building tools and helpers are there, the first step of the configuration and installation process is to build SCoPE. It is a more or less trivial task after getting the proper sources that are available within the restricted area of the MULTICUBE project web site, setting some shell system variables and executing the command `make library`. More information can be found at the SCoPE web site:

<http://www.teisa.unican.es/gim/en/scope/docu.html>

Finally for the configuration and installation of the M3P plug-in and MPEG4 codec example the auto installation script should be used. Alternatively the installation page gives the steps to manually build the M3P library and MPEG4 example from the source code files that can be downloaded from the links provided in the source page.



home source installation developers

Multicube-SCoPE: The installation

In order to make **Multicube-SCoPE**, you have to check out the following list of *System Requirements*, pointing out the tools you need to do the job:

- GNU C/C++ toolchain: gcc/g++ (v4.x), make
- Bison and flex tools (as required by SCoPE)
- zlib-devel library

Of course, you'll have also to get a working installation of the *SCoPE system-level simulator* and the *SystemC 2.2 library* which is used by SCoPE to specify the system to simulate. You can download the latest stable release of SystemC library from the Open SystemC Initiative (OSCI) website. The required SCoPE latest stable release, which is version 1.1.0 (February 2009), must be downloaded from [SCoPE web page](#).

To get all the components working from scratch you must follow these steps:

1. Build SystemC
2. Build SCoPE
3. Build M3P
4. Build and run the adapted IMEC example

To do the first step you must read the SystemC documentation. Basically you must download and uncompress the package and then you must execute the sequence of commands: `configure`, `make` and `make install`. For building SCoPE you must follow these instructions:

1. Set the environment variables
 - `export SYSTEMC=[systemc installation path]`
 - `export SCOPE_HOME=[scope installation path]`
 - `export SCOPE_XML_PLUGIN=[scope-xml-plugin installation path]`
2. Uncompress the package and enter the SCoPE directory
3. Execute: `make libraries`

Figure 4: Screenshot of the installation page of the public website of Multicube-SCoPE Simulator.



III. 4. Team page

This page shows some information about the development team.

This page shows some information about the development team and their relationship with the GIM-TEISA department at the University of Cantabria and a mention to the funding from FP7 EU project.

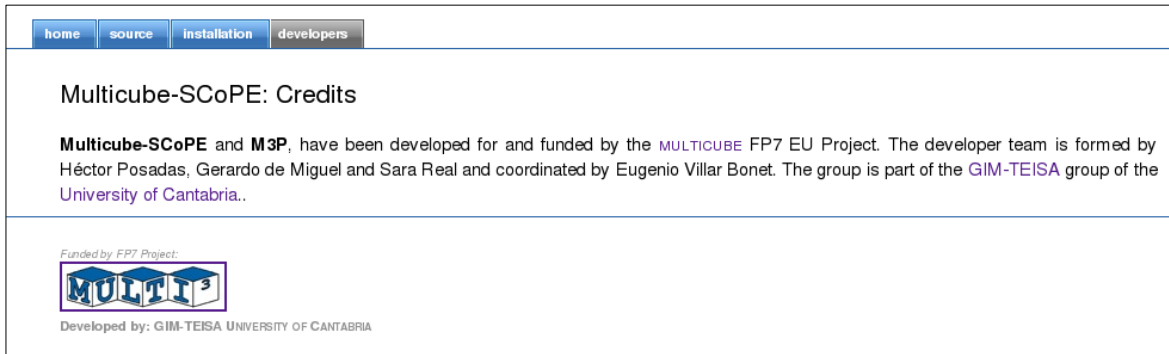


Figure 5: Screenshot of the credits page of the public website of Multicube-SCoPE Simulator.



IV. Improvements to SCoPE internals (Updated)

SCoPE is a SystemC library capable of performing timed simulations of embedded systems composed of SW components running over TLM HW platform models. In addition SCoPE obtains some default metrics about system performance and power consumption of these SW components. The goal of SCoPE is to provide the designer with a fast and sufficiently accurate emulation of the system behavior for functionality verification and constraints checking.

However, Multicube project presents a different approach. In the project the goal is to check the system performance when modifying the platform. Multiple platform configurations for the same system are proposed and the performance estimations obtained are used to select the best possibilities. This new approach implies two new challenges to be resolved:

- Platform to be simulated cannot be a SystemC description. A SystemC description is a fixed C code describing the system components, and its interconnections. That code is compiled together with the entire simulation executable, and thus cannot be dynamically modified. As a consequence, modifying the system components, as the type of bus, or the number of processors, require a complete recoding and compiling process.
- When simulating several times the same system, parts of the simulation can remain unchanged. Reusing information, the modeling of these parts can be improved, optimizing simulation speed.

Furthermore, the Multicube Use Cases require modeling elements not supported by the original SCoPE infrastructure.

To solve these limitations, the original SCoPE had to be improved. This extension has been included in the new SCoPE version, called v1.1.4

The main modifications performed in the SCoPE internal code are:

- Simplification of the platform model building process. Functions for connecting the system components, create the memory maps and loading SW tasks has been simplified in order to allow dynamic, automatic system model construction.
- Extensions of the system components default management. The HW system components that can be used to build the system inherit from a basic set of components. To support all the metrics required these basic components has been extended. The extended components support power features and are able of saving information about system communications and timing information. This information can be used to obtain the system metrics.
- Support for multi-thread parallelism. Previous versions of SCoPE allow modeling parallel execution of processes in multiprocessor systems. However, threads of the same process cannot be executed in parallel even when there are empty processors in the system. The multimedia example requires this level of concurrency, so the SCoPE internal mechanisms have been modified to support it.
- SCoPE operation has been fixed to guarantee concurrent execution of different simulations in order to optimize the use of the host infrastructure.
- Time and Power Parameters in Generic HW Integration Wrappers (New): In order to obtain accurate metrics of the entire system performance, the integration wrappers have been



extended. Now, the wrappers also include facilities to consider time and power information for each component from the XML system descriptions.



V. M3P characteristics (Updated)

M3P is a plug-in for SCoPE which adapts SCoPE to be used for a Design Space Exploration process. Its initial version is centered on modifying the original SCoPE interfaces in order to comply with the Multicube requirements. The initial M3P features are focused on automatic system model building, automatic system configuration and automatic metric reporting.

V. 1. Multicube-SCoPE as a Modeling Tool

The new extensions in M3P allow using Multicube SCoPE as a really powerful tool. This extends the previous prototype capabilities. The new prototype can be used both as a library or a tool, including more powerful modeling features.

- **Dual use of Multicube-SCoPE as a Tool or a Library (New)**

When installing the refined version of M3P the program `scope_tool.x` is generated. This program integrates SCoPE and M3P. To use it, the XML files for system description and system configuration must be provided, together with dynamic libraries containing the use-case specific SW and HW components. Furthermore, the prototype can be used as a library, like previous versions.

- **SW/HW Communication Through Direct Register Memory Addresses (New)**

The refined version provides the features required to support HW/SW communications through direct pointer access to HW peripheral registers. A novel technology using exception management, memory paging and code injection has been developed to support that feature.

- **Memory Space Separation for System Component Models (New)**

The refined prototype tool is prepared to isolate the memory space of each component model. The isolation provided avoids compiling and simulation errors due to name duplications when using the same names for functions or global variables in several component models. Furthermore, it allows instantiating several times the same components. This is really useful when evaluating networked systems, as the power-line use case. Using that feature, nodes can be replicated without limitations.

- **Integration of IMEC ARP Library for Reuse of Timing Information (New)**

The Atomium Record/Playback library of IMEC has been integrated in Multicube- SCoPE. This new feature allows storing data from a reference simulation in order to be re-used in other simulations. During the exploration process multiple simulations of the same code are done, considering different HW configurations. As the code executed is the same, performance information from one simulation can be reused in other simulations with compatible configurations. Furthermore, performance information obtained with other simulators at different abstraction levels can be used during the exploration process, increasing the accuracy of the process.

V. 2. Automatic System Model Building

In order to allow a complete exploration of the design space, the M3P allows the complete building of the system model dynamically. The system architecture and characteristics are



described in an XML file. From this XML file, M3P manages the SCoPE resources to build the system model. To allow the exploration, M3P support incomplete system descriptions. Thus the system is built depending on the configuration options. As a consequence the modeling technique is really flexible, allowing not only to explore the optimal characteristics of the system components, but also to analyze the whole system architecture.

- **Dual input for System Descriptions (New)**

Electronic systems can be described using the XML file formats proposed. The tool is then capable of automatically generating the system model required to perform the simulation. Additionally, system models can also be provided as SystemC code to be explored. Mixed solutions based on XML system descriptions including subsystems provided as SystemC code are also possible under certain restrictions.

V. 3. Automatic System Configuration

M3P is prepared to accept the XML System Configuration files specified in the Multicube D1.1 and D1.4.1 documents. This capability allows M3P to consider automatically the different system configurations selected from the DSE tools to be analyzed.

- **Exploration of System Components Configuration**

The tool allows defining unfixed parameters for the system components in the XML system description file. This unfixed parameters can be set though the XML System Configuration file. Then, the best configuration for each component can be easily explored.

- **Exploration of System Architectures**

Selecting the best system architecture is basic when developing an optimal design. Selecting the type of components, their number and their interconnections must be explored. Using the capabilities for automatic platform generation of Multicube-SCoPE it is possible to provide XML system descriptions with variable architectures. Depending on the configuration files, the number of components and their interconnections can be fixed for each simulation and then automatically explored.

- **Exploration Through Global Variables (New)**

The refined tool allows also exploring internal details of the system component models, not only those details described in the XML description files. To allow these exploration, the parameters must be declared as global variables in the code files and in the XML files. Furthermore, this technique allows exploring systems completely described in SystemC code. Using SystemC codes based on “if/else” structures, several configuration options can be described in the code in order to be explored. This feature allows exploring these option in the Multicube exploration flow.

- **AMBA Bus Modeling (New)**

A new, more accurate model of an AMBA bus has been developed. The model use probabilistic estimations to evaluate blocks and collisions in the bus. As a consequence, the effects of modifying the access priorities for the bus masters can be explored.



V. 4. Automatic Metric reporting

M3P is prepared to get the output metrics required for the system exploration and automatically provide them to the DSE tools. M3P is capable of reading the XML Design Space Definition files specified within the Multicube project to know the metrics that must be reported. Furthermore, it automatically generates an XML output file with the metric results conforming to the Multicube specification.

These three capabilities guarantee the correct integration of Multicube-SCoPE into a complete DSE framework if the DSE tools follow the Multicube specifications for tool integration. It has been checked with modeFRONTIER and Multicube Explorer.



VI. **Conclusions**

This document is the final report of the open source, performance and power estimation framework developed as a simulator within the MULTICUBE Design Space Exploration framework. The simulation framework generated, called Multicube-SCoPE, has been implemented on top of SystemC, as an extension of a previous tool called SCoPE.

The new features included in the tool have been developed to accomplish the requirements of the design space exploration framework and the modeling requirements of the Multicube use-cases supported.

As a result of this work, a freely downloadable packet has been deployed in a web page developed specifically for the purpose, and hosted in the web server of the University of Cantabria. Additionally, manuals and installation scripts has been at disposal in the web.

The tool has been checked with the Multimedia and the Power Line use cases of Multicube project. Furthermore, the tool has been integrated with both exploration tools integrated within the project: M3Explorer and modeFRONTIER.

Additionally, the tool has been prepared to accept performance information obtained from simulator at other levels abstraction in order to obtain more accurate simulation results. This feature has been checked using the information obtained from the simulator developed in task 2.2 of the Multicube project.

VII. **Glossary**

Please refer to the MULTICUBE web page (<http://www.multicube.eu>) and the user guide of Multicube-SCoPE.