AFRL-IF-WP-TM-2003-1534
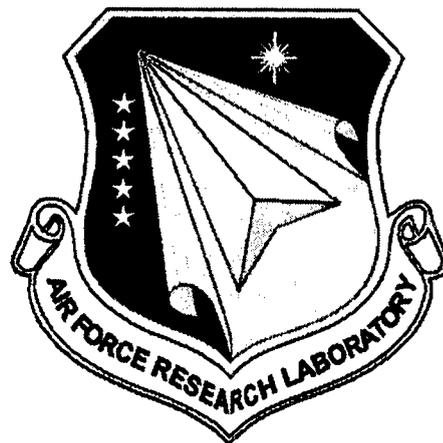
MULTICOMPONENT SYNTHESIS FOR MULTICHIP MODULES (MCMs)

Dr. Ranga Vemuri

University of Cincinnati
Department of Electrical and Computer Engineering
30 Rhodes Hall
Cincinnati, OH 45221-0030

DECEMBER 2001

Final Report for 21 September 1991 – 01 October 1997

20030805 051

INFORMATION DIRECTORATE
AIR FORCE RESEARCH LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7334

NOTICE

USING GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA INCLUDED IN THIS DOCUMENT FOR ANY PURPOSE OTHER THAN GOVERNMENT PROCUREMENT DOES NOT IN ANY WAY OBLIGATE THE US GOVERNMENT. THE FACT THAT THE GOVERNMENT FORMULATED OR SUPPLIED THE DRAWINGS, SPECIFICATIONS, OR OTHER DATA DOES NOT LICENSE THE HOLDER OR ANY OTHER PERSON OR CORPORATION; OR CONVEY ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY RELATE TO THEM.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

DARRELL BARKER
Project Engineer
Embedded Info Sys Engineering Branch
Information Technology Division

JAMES S. WILLIAMSON, Chief
Embedded Info Sys Engineering Branch
Information Technology Division
Information Directorate

Do not return copies of this report unless contractual obligations or notice on a specific document requires its return.

REPORT DOCUMENTATION PAGE

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
December 2001	Final	09/21/1991 – 10/01/1997

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
MULTICOMPONENT SYNTHESIS FOR MULTICHIP MODULES (MCMs)	F33615-91-C-1811
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
	62204F

6. AUTHOR(S)	5d. PROJECT NUMBER
Dr. Ranga Vemuri	6096
	5e. TASK NUMBER
	20
	5f. WORK UNIT NUMBER
	18

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
University of Cincinnati Department of Electrical and Computer Engineering 30 Rhodes Hall Cincinnati, OH 45221-0030	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY ACRONYM(S)
Information Directorate Air Force Research Laboratory Air Force Materiel Command Wright-Patterson AFB, OH 45433-7334	AFRL/IFTA
	11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-IF-WP-TM-2003-1534

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The Multicomponent Synthesis System (MSS) is a vertically integrated collection of design synthesis tools for top-down design of multichip modules (MCMs) from behavioral specifications. MSS consists of a high-level synthesis subsystem, a partitioning engine, a collection of structural silicon compilers, a test compiler, a test bench compiler, a package compiler, a performance specification and verification subsystem, and various component libraries and technology files.

MSS is a VHDL-centered design environment. MSS blends synthesis and simulation tools operating at various levels of abstraction to quickly design correct application-specific MCMs. Functional test bench compilation, in conjunction with boundary-scan architecture and WAVES test data exchange notation is used as the primary means of achieving this goal.

This document describes the objectives of the MSS project, methods employed, and final accomplishments.

15. SUBJECT TERMS

VHDL, behavioral synthesis, test bench compilation, multichip modules, electronic computer-aided design

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE	SAR	14	Darrell Barker
Unclassified	Unclassified	Unclassified			19b. TELEPHONE NUMBER *(Include Area Code)* (937) 255-6548 x3605

Table of Contents

1. Objectives of the MSS Project

The chief objective of the Multicomponent Synthesis Systems (MSS) project is to design and develop a vertically integrated ynthesis environment for multicomponent designs. While specifically targeted to multichip module (MCM) packaging technologies, most of the MSS tools are expected to be retargetable to generate other multicompnent carrier designs. MSS environment is expected to fully support the following:

- *Automated Synthesis:* Several levels of automated design synthesis tools form the cornerstone of the MSS environment. Full synthesis support includes behavioral, logic, and layout synthesis to generate functional Very Large Scale Integrated) VLSI chip designs from large, monolithic specifications.

- *Testability:* Muticomponent designs generated by MSS are expected to e fully testable by external testing alone. To facilitate this, boundary scan devices are to be integrated into the designs.

- *Automated Design Validation:* Due to the large size and complexity of the designs, hand generation of test vectors is ruled out. MSS should consist of automated test bench generation tools to produce test benches *stimuli and expected responses) for functional testing. Such test benches should automatically exercise the designs and compare the actual responses with the expected ones.

To further delineate the scope of the project, the following biases are defined:

- *HDL-Centeredness*: MSS should be a Hardware Description Language (HDL) centered design environment. The VHDL hardware description language is the pivotal language of MSS and is used for behavioral, register level, and gate level design descriptions. All levels of synthesis should cater to take advantage of the various features of VHDL as appropriate.

- *Conformance to Standards:* MSS interfaces should conform to appropriate formal standards provided such standards have the potential to be widely accepted by the industry. Following this, the multicomponent designs should contain Institute of Electrical and Electronic Engineers (IEEE) standard 1149.1 boundary-scan architectures and test-access ports. All test benches must be in "WAVES" format which is the IEEE standard for denoting test data. Conversely, where acceptable standard notations do not currently exist, notations developed for MSS are expected to spur some standardization activity as appropriate.

- *Performance-Intensive* Technologies: MSS algorithms should be as technology independent as possible. However, a successful performance-intensive implementation of the design environment must have a target technology. MSS is currently being targeted to scalable Complementary Metal-Oxide Semiconductor (CMOS) technology for chip and High Density Interconnect (HDI) technology for packaging.

- *Integration of Industrial Strength Tools:* Whenever industrial strength tools or usable university tools exist for various design tasks, MSS should make use of them, provided such tools are available to the research team at little or no cost. Two categories of tools that belong to this category are physical design automation tools and simulation tools. MSS should make use of existing physical design tools for both chip and package development at the back-end and VHDL simulators for design validation. Currently, the Lager IV silicon assembler from Mississippi State University, MCM station from Mentor Graphics Corp., and the VHDL simulators from MCC and from Intermetrics Inc. fall into this category.

- *Implementation Platform:* All the tools in MSS execute on a Sun Sparc Station. MSS software is written primarily in C++ with some components in C and Prolog. Execution of MSS in general requires X-Windows and OpenWindows. Some tools can however be executed in text mode. Compiler generators such as Bison++, Yacc and Lex are extensively used.

2. Development Teams

The MSS project is conducted by the Laboratory for Digital Design Environments, Electrical, and Computer Engineering Department of the University of Cincinnati (UC) in collaboration with the Texas Instruments (TI) Defense Systems and Electronics Group. Dr. Ranga Vemuri is the principal investigator of the project. TI is a subcontractor to the University of Cincinnati. Mr. Neal Stollon is the program manager at TI. All the algorithms and software are being developed at UC.

3. Methods and Tools in MSS

A typical design flow through MSS begins with a behavioral specification in VHDL and a performance specification of the digital system in Performance Description Language (PDL). The digital system is usually a large-scale application specific module that is to be implemented. The various tools in MSS as described below then process the specification.

High-Level Synthesis System: The VHDL specification is first translated into an intermediate format called VHDL Intermediate Format (VIF). An internal data flow graph is initialized based on VIF representation of behavior. The high-level synthesis system generates a register level design. The high-level synthesis system is a collection of efficient algorithms for data flow graph scheduling, peformance estimation, resource allocation and binding. It uses a library of parameterized register-level modules also written in VHDL.

Partitioning and Tradeoff Analysis Engine: The partitioning engine then partitions the design into multiple components based on the stated performance goals. Tradeoff analysis and partition evaluation uses the PDL subsystem, which in turn uses a library of predefined performance evaluations written in PDL

Test Structure Compiler: due to the limited access to physically probe a multichip module during rtest, integration of test structures in a MCM design is extremely important. The test structure compiler generates boundary-scan test structures and test-access ports around each of the register level designs generated by the high-level synthesis system. All the scan-structures are connected into a system-wide scan chain.

Structural Silicon Compilers: Each register level design is then processed by a structural silicon compiler to generate mask layouts of individual chips. Different technologies may be used based upon the constraint specification or user choice. Various commercial and in-house silicon compilers are provided for this purpose.

Test Bench Compiler: The test-bench compiler, operating in close association with the test-structure compiler, accepts a WAVES test-bench used to simulate the behavioral-level specification of the system written in VHDL and certain timing information generated by the high level synthesis system and generates a collection of WAVES test benches to test each chip in the MCM design as well as the system-side functionality and interconnect structure. These test-benches can be used to simulate the register level design or switch level design of the individual chips or the entire MCM or to drive the Automatic Test equipment for testing.

Performance Specification and Verification: Critical performance issues make the design of multichip modules a complex and time-consuming task. The MSS contains a performance specification and a performance description language called PDL. PDL provides rigorous and executable notations for specification of performance attributes at arbitrary levels of abstraction. The PDL interpreter can execute the PDL programs to generate performance data for a given design at any level of abstraction.

4. Accomplishments

The main accomplishments of the MSS project are as follows:

1. Development of a MSS with integrated partitioning and synthesis.

2. Development of a boundary-scan test structure compiler.

3. Development of a boundary-scan test bench compiler.

4. Development of a PDL and analysis environment.

5. Development of various partitioning, placement, and routing methods for MCMs.

6. Development of partitioning and synthesis techniques for multiple Field Programmable Gate Array Designs.