GAUT – A Free and Open Source High-Level Synthesis Tool for FPGA-Based Acceleration of Scientific Computing

Philippe COUSSY, Dominique HELLER

Lab-STICC/CNRS – Universite de Bretagne-Sud– France

http://lab-sticc.fr/www-gaut

Abstract *- GAUT is an open source High-Level Synthesis tool. From a bit-accurate C/C++ specification it automatically generates a RTL architecture described in VHDL that can be used by commercial logical synthesis tools like ISE (Xilinx), Quartus (Altera). GAUT also generates TLM and CABA SystemC simulation models for virtual prototyping.*

Introduction

The increasing complexity and the data rates of applications require efficient hardware implementations like dedicated accelerators [1][2]. ESL level tools are needed in order to raise the specification abstraction level up to the « algorithmic one » [3]. Algorithmic descriptions enable to focus on functionality and target performances rather than debugging RTL. CatapultC from Mentor Graphics, Cynthesizer from Forte or C-to-Silicon from Cadence are EDA software tools that automatically generates synthesizable RTL starting from untimed C/C++/SystemC. GAUT is an academic and open source HLS tool dedicated to highly computational applications.

GAUT, a HLS tool

GAUT [4][5] takes as input a C/C++ description of the algorithm that has to be synthesized where Algorithmic C™ class library from Mentor Graphics can be used. This allows the designer to specify signed and unsigned bit-accurate integer and fixed-point variables by using bit accurate integer and fixed-fixed data types [6]. The mandatory constraints are the throughput (specified through an initiation interval II which represents the constant interval between the start of successive iterations) and the clock period. Optional design constraints are the memory mapping [7] and I/O timing diagram [8]. The architecture of the hardware components that GAUT generates is composed of three main functional units: a processing unit PU, a memory unit MEMU and a Communication & Interface Unit COMU. The PU is a datapath composed of logic and arithmetic operators, storage elements, steering logic and a controller (FSM). The MEMU is composed of memory banks and their associated controllers. The COMU includes a synchronization processor and an operation memory which allow to have a GALS / LIS communication interface [9]. To validate the generated architecture, a test bench is automatically generated to apply stimulus to the design and to analyze the results. GAUT generates not only VHDL models but also scripts necessary to compile and simulate the design with the Modelsim simulator.

GAUT generates an IEEE P1076 VHDL file. The VHDL file is an input for commercial, off the shelf, logical synthesis tools like ISE/Foundation from Xilinx, Quartus from Altera or Design Compiler from Synopsys. GAUT generates a VHDL test-bench and is seamlessly interfaced with Modelsim from Mentor Graphics. SystemC simulation models can be automatically generated. These models are Cycle Accurate Bit Accurate (CABA) and TLM-DT (Transaction Level Modeling with Distributed Time) and can be used in the SocLib platform [15] for virtual prototyping.

GAUT generates protocol specific interfaces. This enables to execute the synthesized DSP applications in a mixed hardware/software system. This approach has been validated with a platform based on C6x from TI and Virtex from Xilinx.

GAUT ICs are tested with the PALMYRE platform in various FPGAs contexts. FPGAs are stimulated by test pattern generators and results collected with logic analyzers. Mixed software/hardware configurations also allow to connect DSPs to/from FPGAs and deliver/collect data on the field without the need for heavy instrumentation. Hardware and software interfaces/libraries have been developed to ease interconnect of processes at C code level.

GAUT also supports FSL (Fast Simplex Link) interfaces which allow to easily connect customized IPs to the MicroBlaze or the PowerPC processors in Xilinx FPGAs.

GAUT also addresses the design of multi-mode architectures [12]. Given a unified description of a set of time-wise mutually exclusive tasks and their associated throughput constraints, a single RTL hardware architecture optimized in area is generated [12][14]. GAUT supports hierarchical synthesis and will generate multiple clock domain architecture for low-power design on FPGA [13].

References

[1] IEEE Design and Test of Computer, Special Issue on High-Level Synthesis, July-August 2009

[2] "High-Level Synthesis: From Algorithm to Digital Circuit", P. Coussy and A. Morawiec, Springer, Berlin, Germany, 2008

[3] "An Introduction to High-Level Synthesis", P. Coussy, G. Gajski, A. Takach, M. Meredith, Special issue on High-Level Synthesis, IEEE Design and Test of Computers, Vol. 26, Issue 4, July/August, 2009

[4] GAUT web site: http://lab-sticc.fr/www-gaut

[5] "GAUT: A High-Level Synthesis Tool for DSP Applications", P. Coussy, C. Chavet, P. Bomel et al., in High-Level Synthesis: From Algorithm to Digital Circuits, Springer, 2008

[6] "Multiple Word-Length High-Level Synthesis", P. Coussy, G. Le Breton, D. Heller, EURASIP Journal on Embedded Systems, 2008

[7] "Memory Accesses Management During High Level Synthesis", G. Corre, E. Senn, P. Bomel, N. Julien, E. Martin, IEEE Int. Conference on Hardware-Software Codesign and System Synthesis, (CODES+ISSS) 2004

[8] "A Formal Method for Hardware IP Design and Integration under I/O and Timing Constraints", P. Coussy, E. Casseau, P. Bomel, A. Baganne, E. Martin, ACM Trans. on Embedded Computing Systems, Vol 5, No. 1, 2006

[9] "Synchronization Processor Synthesis for Latency Insensitive Systems", P. Bomel, E. Martin, E. Boutillon, IEEE International Conference on Design Automation and Test in Europe (DATE) 2005

[10] "Pipelined memory controllers for DSP applications handling unpredictable data accesses", B. Le Gal, E. Casseau, S. Huet, E. Martin, IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2005

[11] "A Design Methodology for Space-Time Adapter", C. Chavet, P. Coussy, P. Urard, E. Martin, IEEE/ACM Great Lakes Symposium on VLSI (GLSVLSI), 2007

[12] "A Design Flow Dedicated to Multi-mode Architectures for DSP Applications", Chavet C., Andriamisaina C., Coussy P., Casseau E., Juin E., Urard P., Martin E., Dans IEEE International Conference on Computer Aided Design, (ICCAD), 2007

[13] "Hierarchical and Multiple-Clock Domain High-Level Synthesis for Low-Power Design on FPGA", G. Lhairech-Lebreton, P. Coussy, E. Martin, IEEE Int. Conference on Field Programmable Logic and Applications, (FPL) 2010

[14] "High-Level Synthesis for Designing Multi-mode Architectures", C. Andriamisaina, P. Coussy, E. Casseau, C. Chavet, IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems TCAD, Vol. 29, Issue 11, Nov. 2010.

[15] https://www.soclib.fr/