Experience with Image Compression Chip Design using **U**nified **S**ystem **C**onstruction Tools*

Pravil Gupta, Chih-Tung Chen, J.C. DeSouza-Batista† and Alice C. Parker
Department of Electrical Engineering - Systems
University of Southern California
EEB 300, MC 2562
Los Angeles, CA 90089-2562, USA

Abstract – This paper describes the use of Unified System Construction tools under development at the University of Southern California. The goal of the project is to automate the construction of heterogeneous, application-specific systems. Key elements of the USC system include multiprocessor synthesis, multi-chip datapath synthesis, memory-intensive synthesis, and multi-chip partitioning. The tools were applied to design of an image compression chip set, and results of using these tools are reported on here. Our results are comparable to manual designs reported in the literature.

1 Introduction

Communications, entertainment, and other electronic systems are in widespread use. These systems are generally multi-chip, heterogeneous, and application-specific. Chip-level synthesis tools are invaluable for the rapid production of such systems, and such tools are becoming available for general use. System-level tools can also be used to significantly increase a designer's ability to meet a schedule along with a set of performance and cost constraints, but only a few of these tools have been available in the past.

The Unified System Construction (USC) project at the University of Southern California involves the production of an integrated set of system-level tools for synthesizing multi-chip, heterogeneous application-specific systems which meet cost, performance and power constraints. *This paper presents the use of these system-level tools to perform a multi-chip design exercise, a JPEG image compression chip set.* The focus of the USC project is on real-time systems, such as entertainment and communication technologies, but does not exclude other applications requiring specialized system design. A block diagram of the system is shown in Figure 1.

*This work was supported by the Advanced Research Projects Agency and monitored by the Federal Bureau of Investigation under Contract No. JFBI90092.

†Supported by Conselho Nacional de Desenvolvimento Cientifico e Tecnologico - CNPQ/Brazil.

The user of the USC software first selects a style for the system. Styles currently supported include

- heterogeneous multiprocessors consisting of
  - processors interconnected with point-to-point connections, operating in a non-pipelined fashion.
  - non-pipelined processors in a ring,
  - non-pipelined processors connected by a bus,
  - pipelined processors with point-to-point connections,
- multiple custom VLSI chips, communicating asynchronously,
- multiple custom VLSI chips, communicating synchronously, with common clock, and
- memory-intensive modules consisting of a custom VLSI chip and a separate memory chip.

Many other styles of systems are currently under development. Once a style is selected, specialized tools are invoked to complete the design process. Ultimately, any custom VLSI chips which must be synthesized are then processed by the ADAM high-level synthesis system, which produces a cell netlist. This netlist is input to the Cascade Design Automation Chipcrafter Silicon Compiler, and a chip layout is produced.

The following sections give an overview of each major style of design, in the order each was applied to the compression example. The remaining sections describe the image compression system to be designed and detail various design activities conducted using the USC tools.

2 Synthesis of Memory-Intensive Systems

A subset of the USC tools performs automatic synthesis of memory-intensive application-specific systems, with emphasis on hierarchical storage architecture design. The storage architecture is closely connected to the datapath of the system, and isolating its synthesis from datapath synthesis may not result in an efficient solution. Therefore, the design of the datapath and storage architecture

Figure 2: SMASH: Synthesis of Memory-intensive Application Specific Hardware

Figure 3: A flow chart of the synthesis system for asynchronous multi-chip designs

3 Synthesis of Asynchronous Multi-chip Systems

In practice, we find that many DSP and other ASIC designs consist of multiple concurrent and interacting processes. Though high-level synthesis has received enormous attention over the years, most approaches were concentrated on synthesizing single process (one thread of control) designs. Synthesizing a design with multiple concurrent processes poses many new challenges. For example, since the processes interact with each other, the synthe-

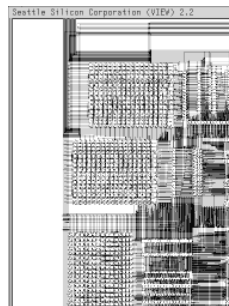
fication is translated from VHDL to a synthe
resentation called the Design Data Structure
The next step is to perform a number of pro
formations in order to trade off among hardwa
control complexity, communication overhead a
process-level chip partitioner, ProPart [4], is t
find new cost-effective chip boundaries accor
up-to-date packaging library. In addition, the
will distribute chip resources to the processes
to their performance-versus-area characterist
termine the interconnection structure as wel
concurrent approach for multiple-process syntl
to synthesize each process into its own datapa
trol path. The objective is to meet the timin
performance constraints as well as to synchroni
munication among the processes. Finally, w
brid symbolic/numeric simulation to verify the
and timing correctness of the RTL implementa
RTL implementation is submitted to the AD
to obtain the final chip layout.

ProPart was used in the experiment of a J
compression system to be described later in
Unlike most of the previous behavioral parti
proaches which focus on partitioning design b
the operation level into a number of synchron
ProPart tries to partition a set of sequential a
current behaviors into custom chips. There
advantages to process-level partitioning. Fo
there are far fewer objects at the process level
at the operation level, which allows us to u
more comprehensive techniques like mixed in
programming and at the same time to take i
more partitioning issues, like chip package se
chip resource distribution.

SpecPart [19] is the first system-level behav
tioning work which elevates the objects to be
to a higher level of abstraction (such as process
cedures), and uses a group migration techniqu
the Kernighan-Lin algorithm for partitioning.
hensive survey of other behavioral partitioning
at the operation level has been done by Vahid

4 Multiprocessor Synthesis

the second 1D-DC
that ProPart placed
and lumped the re
Finally, we gener
and 2D-DCT chip
ChipCrafter (Figur
tribution (Table 5)



Fig

Table 7: 2D-DCT implementations from SOS

required in order to obtain the final layouts from the RTL datapaths. It was clear from the exercise that many more dimensions of the design space could have been searched by our tools, given more time. For example, SOS produced a variety of architectures for 2D-DCT that can be used to meet different design requirements. Our estimation tools, which were not used, will provide valuable information early for our tools when used in actual design situations.

References

[1] S. H. Bokhari. *Assignment Problems in Parallel and Distributed Computing*. Kluwer Academic Publishers, 1987.

[2] C.F. Chang and B.J. Sheu. A Multi-Chip Module Design for Portable Video Compression Systems. In *IEEE Multi-Chip Module Conf.*, pages 39–44, 1993.

[3] C.T. Chen and A.C. Parker. VHDL2DDS: A VHDL Language to DDS Data Structure Translator. Tech. Rep. CEng 91–21, Dept. of EE-Systems, Univ. Southern California, July 1991.

[4] C.T. Chen and A. C. Parker. ProPart: A Process-Level Behavioral Partitioner. Tech. Rep. CEng 93–38, Dept of EE-Systems, Univ. of Southern California, Sept. 1993.

[8] H. Fujiwara, M.L. Liou
M.M. Maruyama, K. Sho
All-ASIC Implementatio
Codec. *IEEE Trans. on*
Video Technology, 2(2):1

[9] P. Gupta and A. C. Par
for Scheduling Memory-
cific Hardware. In *7th Int*
thesis, May 1994.

[10] E. K. Haddad. Optimal
lel and Distributed Proce
Department of Computer
nic Institute and State U

[11] L. J. Hafer and E. Huto
Tech. Rep. CMPT TR
Science, Simon Fraser U
1S6, Mar. 1990.

[12] K. Kucukcakar and A. C
offs using MABAL. *27th*
June 1990.

[13] P.E.R. Lippens, J.L. va
Werf, W.F.J. Verhaegh, a
ory Synthesis for High S
Proc. of the IEEE Custo
May 1991.

[14] P. Marwedel. The MIM
tailed Description of the
Design Automation Conf

[15] H. Park and V.K. Prasan
chitectures for Huffman C
tics, Speech and Signal P

[16] A. C. Parker, P. Gupta,
fects of Physical Design C
Performance Tradeoff C
tomation Conf., June 199

[17] S. Prakash and A. Pa
Application-Specific Het
Systems. *Journal of Par*
puting, 16:338–351, Dec.