A Framework for High-Level Synthesis of System-on-Chip Designs

James E. Stine, Johannes Grad, Ivan Castellanos, Jeff Blank, Vibhuti Dave, Mallika Prakash, Nick Iliev, and Nathan Jachimiec

Illinois Institute of Technology, Chicago IL USA

http://vlsi.ece.iit.edu

VLSI Computer Architecture, Arithmetic, and CAD Research Group (VCAAC)

IIT CELL LIBRARY

The IIT cell library has been developed to provide a free, non-proprietary library for MOSIS SCMOS rules.

It supports both commercial and public domain EDA tools.

The library is available for free upon request from http://vlsi.ece.iit.edu/scells

Supported MOSIS Technologies

- AMI 0.5µm (includes pads)
- AMI 0.35µm (includes pads)
- TSMC 0.25µm
- TSMC 0.18µm

List of Cells

AND2X1	INVX4
AND2X2	INVX8
AOI21X1	LATCH
AOI22X1	MUX2X1
BUFX2	NAND2X1
BUFX4	NAND3X1
CLKBUF1	NOR2X1
CLKBUF2	NOR3X1
CLKBUF3	OAI21X1
DFFNEGX1	OAI22X1
DFFPOSX1	OR2X1
DFFSR	OR2X2
FAX1	TBUFX1
FILL	TBUFX2
HAX1	XNOR2X1
INVX1	XOR2X1
INVX2	

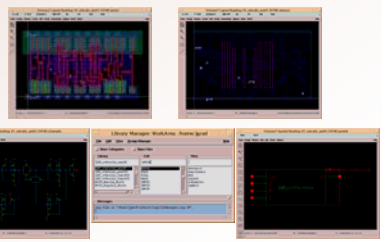
Highlights

Characterization

- Performed using Cadence Signalstorm
- Spice accurate delay and power umbers
- 5x5 lookup tables

New for Version 2

- Set/Reset Flops
- Support for AMI 0.35µm
- Techfiles for Fire&Ice, Voltagestorm
- DF-II Schematics



SOCKS: SOC SIMULATION AND VERIFICATION

The SOCKS platform is a Leon based system-on-chip with an AMBA bus.

It provides an empty AMBA block for student experiments and a testbench with debug support.

SOCKS includes several C/C++ code examples, make-files and Cadence NC-Sim simulation scripts.

The SOCKS platform is used in a SOC class at IIT for the implementation and verification of a floating point unit designed by students accessed by a LEON processor (a compliant SPARC v8 processor)

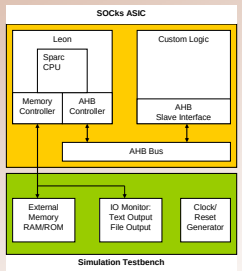


Figure: SOCKS Platform Block Diagram

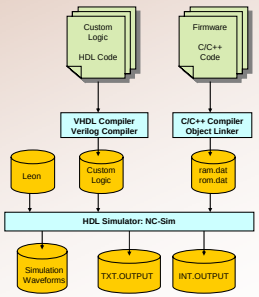
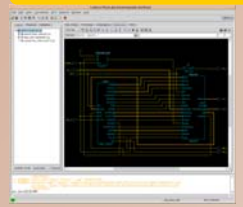


Figure: SOCKS Simulation Flow

IIT PHYSICAL DESIGN FLOW

- Flow templates for many University-supported Cadence and Synopsys tools
- Fully automated interface between all tools
- Used at IIT in research as well as several undergraduate and graduate classes

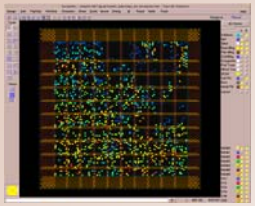
1) Logic Synthesis



- Cadence PKS
- Synopsys Design Compiler
- Cadence RTL Compiler

Figure: MIPS Core HDL Toplevel in PKS

2) Floorplanning

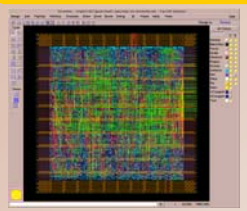


Cadence SOC Encounter

- Power Planning
- Placement
- Clock Tree Synthesis

Figure: Clock Tree Display in Cadence Encounter

3) Place & Route

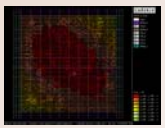


Place & Route Flow Steps

- In-Place Optimization
- Routing (Cadence Nanoroute)
- Extraction (Cadence Fire&Ice)
- Static Timing Analysis
- GDS Export

Figure: Routed MIPS Core in Encounter

4) Power Analysis



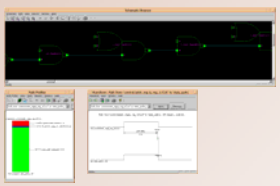
- Cadence Voltagestorm
- IR-Drop Analysis

Figure: MIPS Core IR-Drop Plot

- Dynamic power analysis in Encounter
- Switching Activity from VCD File

Figure: MIPS Core Dynamic Power Consumption

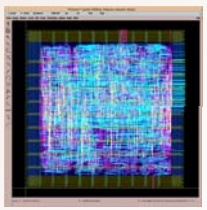
5) Signoff Timing



- Post-Route Timing Analysis
- Uses extracted parasitics
- Supported
  - Synopsys Primetime
  - Cadence CTE

Figure: MIPS Core Static Timing Analysis in Synopsys Primetime

6) Chip Finishing



- Fully automated import
- LVS (Diva)
- Mask level extraction (DIVA)
- Uses NCSU techkit

Figure: MIPS Core in Cadence Virtuoso