CoDEx: CoDesign for Exascale

John Shalf, David Donofrio

X-Stack PI Meeting

May 28, 2014

CoDEx: CoDesign for Exascale

V1: 2010-2013

U.S. DEPARTMENT OF ENERGY | Office of Science

Objective

Create a comprehensive architectural simulation platform to accelerate hardware/software co-design for exascale computing.

Our Tools

- **ROSE Compiler**: Automated Application Code Analysis
- **RAMP/Green Flash:** FPGA Accelerated model of Advanced Compute Node Architectures
- **SST / Macro:** Simulation of Future System Scale Interconnects

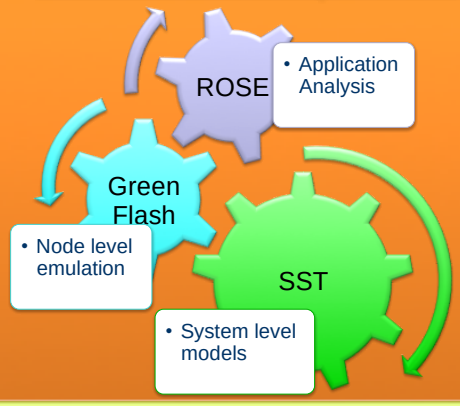
Our Team

Lawrence Berkeley National Laboratory
John Shalf,
David Donofrio

Livermore
Dan Quinlan

Sandia
Gilbert Hendry

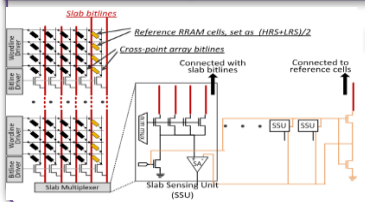
Interoperating Simulation Components



Our Vision: Create a comprehensive simulation environment for design space exploration

- **Simulate advanced computing systems BEFORE we build them!**
- Enable DOE to use **predictive performance and energy models** of future HPC systems to make design decisions and **guide Exascale Software Design.**

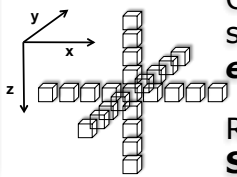
Impact: Simulating New Technologies



Developed **NVRAMsim** component to study emerging nonvolatile memory technology and arch.

Nominated for Best Student Paper at SC13

Impact: Green Wave, Inc.



CoDEx Tools used to Co-design seismic imaging supercomputer ***3x-11x more energy efficient*** **than latest GPU or CPU!**

Resulted in ***3 International Patents*** and **Spin-Off Company (Green Wave Inc.)**

Impact: Enabling Computer Architecture Research

CoDEx delivered **multiple multi-core simulation** environments that enable research into sustainable, scalable computing platforms for exascale.

Broadly used by DOE Co-design centers

Impact: Unprecedented Industry Collaboration

CoDEx introduced DOE architectural simulators to the **System-C** to **share simulator components** and collaborate with **industrial partners**

SYSTEMC | tensilica | intel | AMD

Objective

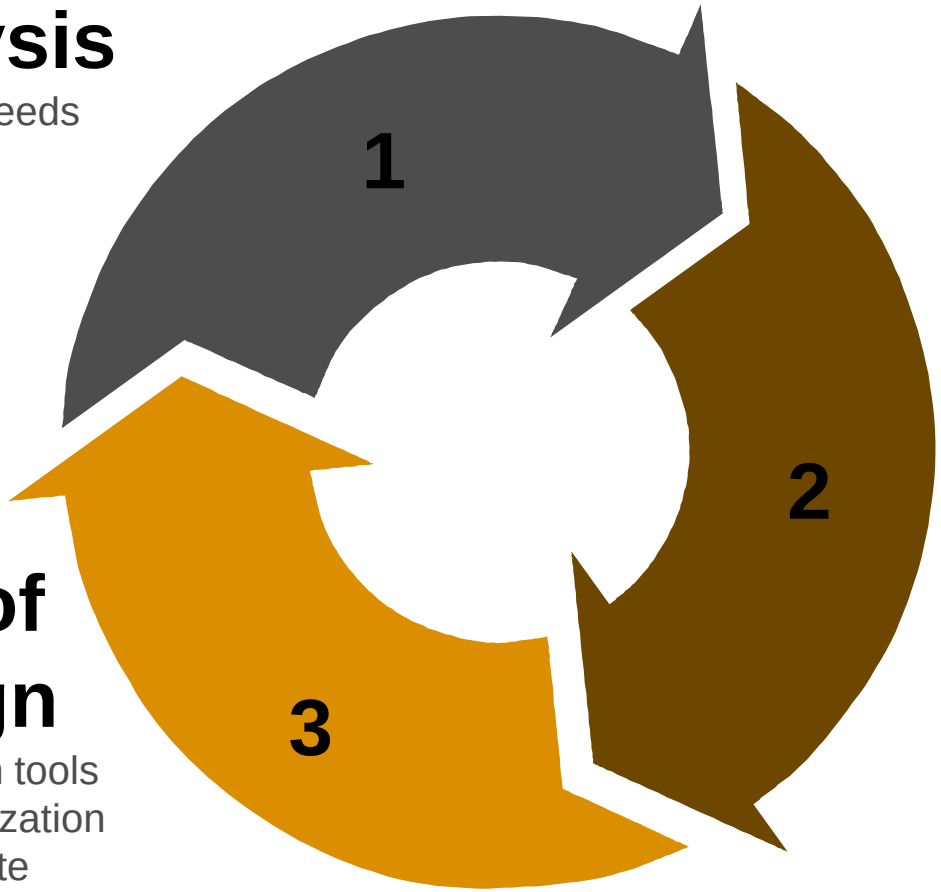
Create a comprehensive architectural simulation platform to accelerate hardware/software co-design for exascale computing

CoDEx Objective

Create a comprehensive architectural simulation platform to accelerate hardware/software co-design for exascale computing

Code Analysis

Static analysis, basic speeds and feeds, unbounded hardware models



Rapid Exploration

Use software based simulators and software kernels to explore hardware parameter space

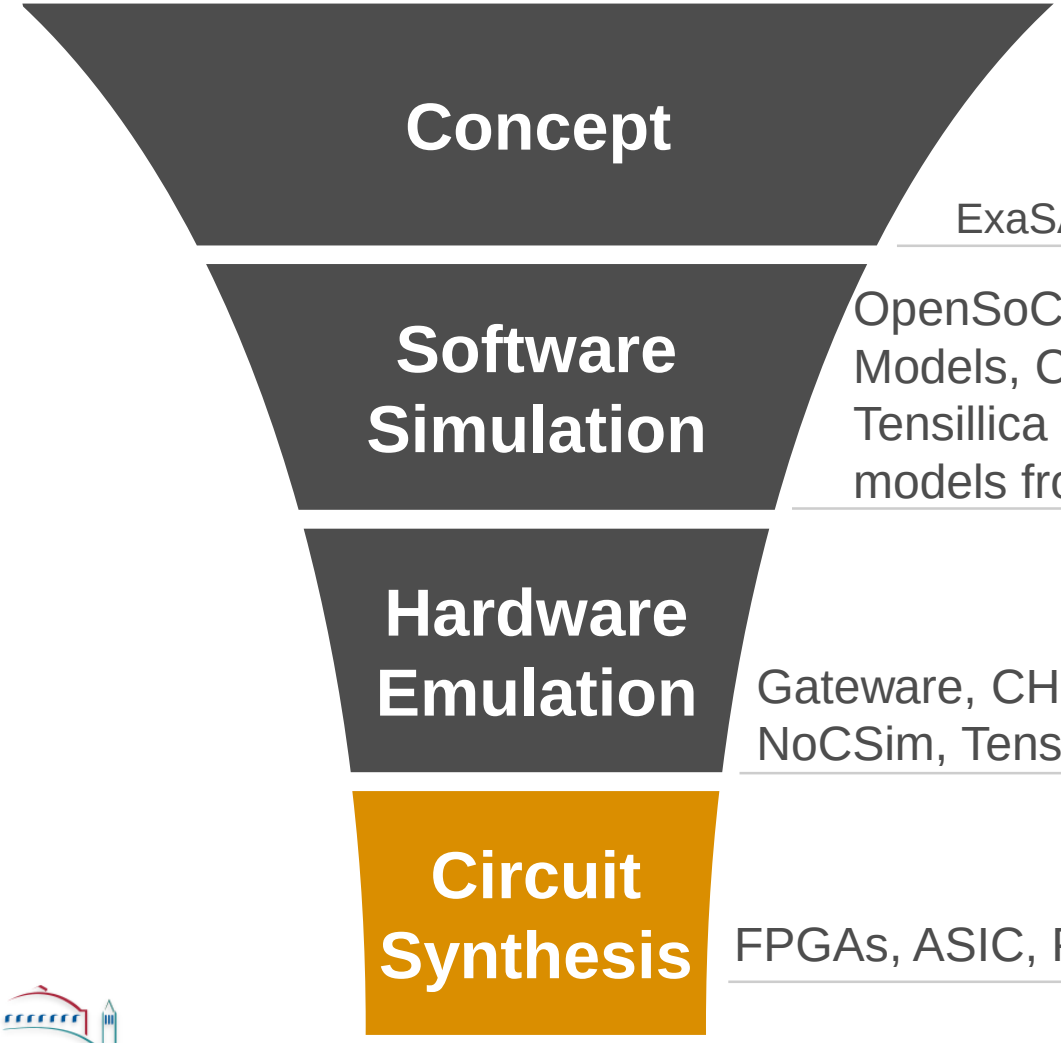
Synthesis of Point Design

Use hardware emulation tools for full application optimization and extraction of accurate power and area estimates

CoDEx has tools for each phase of the CoDesign process

CoDEx Tool Summary

Tools span from high-level, concept exploring tools to physical circuit synthesis



ExaSAT and spreadsheet models

OpenSoC, DRAMSim, NVRAMSim, HMC Models, CHISEL generated components, Tensillica Xtensa models, SystemC based models from industry

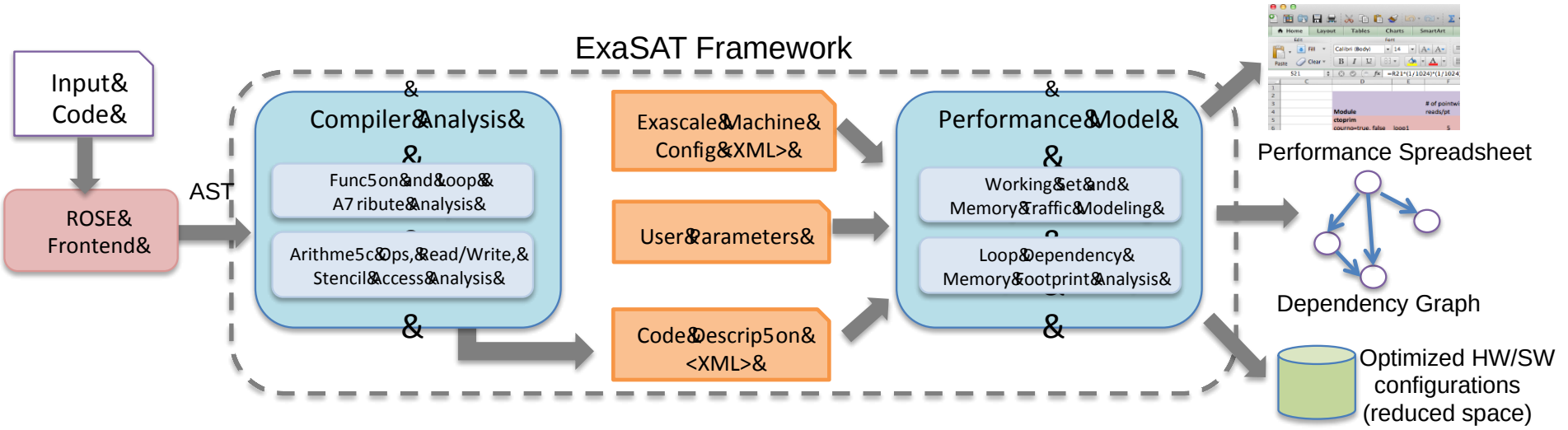
Gateware, CHISEL generated components, NoCSim, Tensilica Xtensa processors

FPGAs, ASIC, Power model parameter extraction

CoDEx Tools: ExaSAT

Compiler driven performance analysis framework

- **Extracts key application statistics in HW independent fashion**
  - HW configs parameterized for code performance estimation
- **Estimate performance benefits of code transforms *without* changing the code**



The Old Way of Analytic Modeling

Module		Read-only	Write-only	first read, then written RW	first written then read WR	Stencil?	Vars that can be read from registers after written
ctoprim							
ctoprim(t)	loop1	U1-U5	Q1,Q5,Q6		Q2,Q3,Q4	No	Q2,Q3,Q4
	loop2	Q1-Q5					
ctoprim(f)	loop1	U1-U5	Q1,Q5,Q6		Q2,Q3,Q4	No	Q2,Q3,Q4
diffterm			D1			No	
ux,vx,wx	loop1	Q2, Q3, Q4	ux, vx, wx			Yes	D2-D4 across loops
uy,vy,wy	loop2	Q2, Q3, Q4	uy, vy, wy			Yes	
uz,vz,wz	loop3	Q2, Q3, Q4	uz, vz, wz			Yes	
imx	loop4	Q2, vy, wz	D2			Yes	
imy	loop5	Q3, ux, wz	D3			Yes	
imz	loop6	Q4, ux, vy	D4			Yes	
iene	loop7	Q2-4,Q6, D2-4	D5			Yes	
		ux-z, vx-z, wx-z					
hypterm							
	loop1	U2-U5, Q2, Q5	F1-F5			Yes	F1-F5 acroos loops
	loop2	U2-U5, Q3, Q5		F1-F5		Yes	
	loop3	U2-U5, Q4, Q5		F1-F5		Yes	
CalcU							
N+1/3	loop1	U, D, F	Unew			No	
N+2/3	loop2	U, D, F		Unew		No	
N+1	loop3	Unew, D, F		U		No	

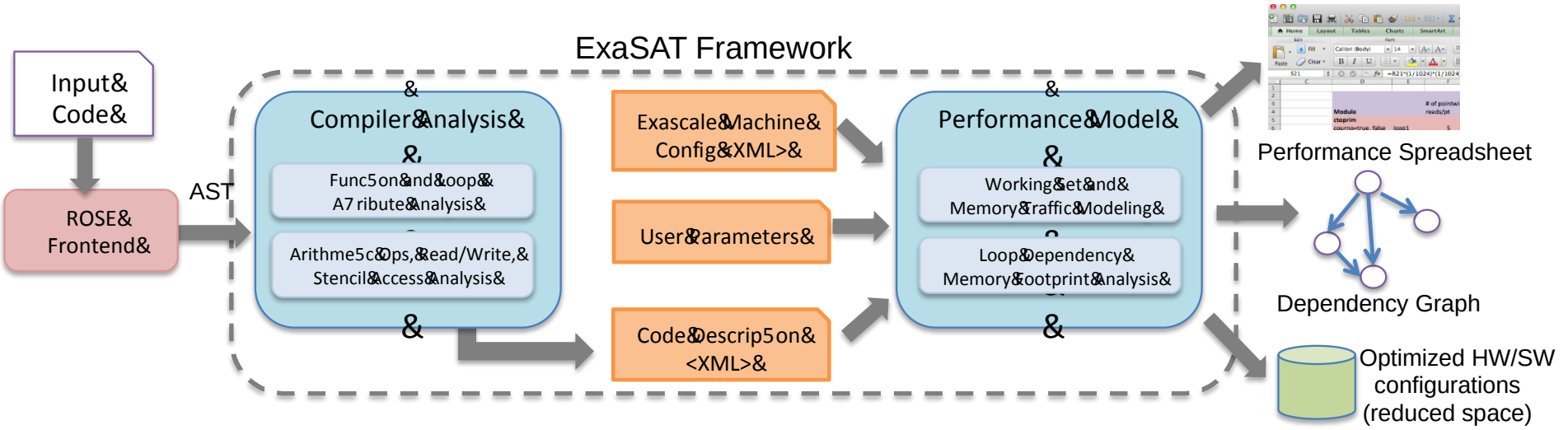
The Old Way of Analytic Modeling

Module		# of Reads pt	# of Reads w halo/pt	# of Writes /pt	in bytes Reads w/o halo /box	in bytes Reads w halo /box	in bytes Writes /box	in bytes Total Memory Access /box	in MB /box
ctoprim									
courno=true	loop1	5	0	6	1310720	0	1572864	2883584	2.75
	loop2	5	0	0	1310720	0	0	1310720	1.25
courno=false	loop1	5	0	6	1310720	0	1572864	2883584	2.75
diffterm		0	0	1	0	0	262144	262144	0.25
ux,vx,wx	loop1	0	3	3	0	1376256	786432	2162688	2.0625
uy,vy,wy	loop2	0	3	3	0	1376256	786432	2162688	2.0625
uz,vz,wz	loop3	0	3	3	0	1376256	786432	2162688	2.0625
imx	loop4	0	3	1	0	1376256	262144	1638400	1.5625
imy	loop5	0	3	1	0	1376256	262144	1638400	1.5625
imz	loop6	0	3	1	0	1376256	262144	1638400	1.5625
iene	loop7	15	1	1	3932160	458752	262144	4653056	4.4375
hypterm									
	loop1	0	6	5	0	2752512	1310720	4063232	3.875
	loop2	5	6	5	1310720	2752512	1310720	5373952	5.125
	loop3	5	6	5	1310720	2752512	1310720	5373952	5.125
CalcU									
N+1/3	loop1	15	0	5	3932160	0	1310720	5242880	5
N+2/3	loop2	20	0	5	5242880	0	1310720	6553600	6.25
N+1	loop3	20	0	25	5242880	0	6553600	11796480	11.25

CoDEx Tools: ExaSAT

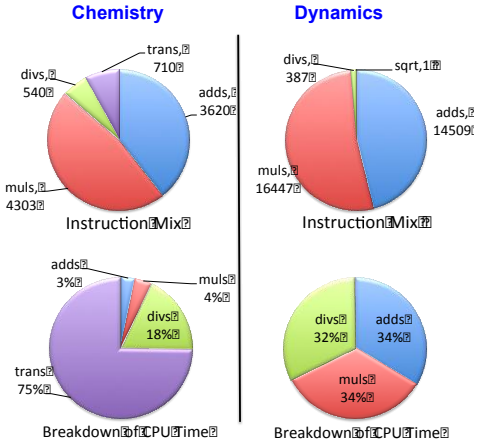
Compiler driven performance analysis framework

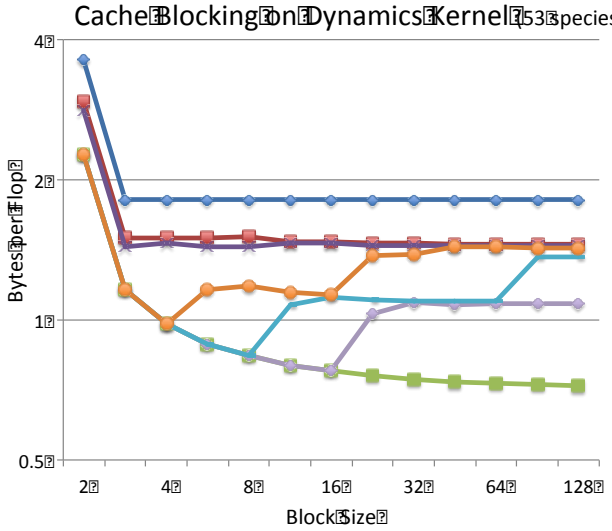
- **Extracts key application statistics in HW independent fashion**
  - HW configs parameterized for code performance estimation
- **Estimate performance benefits of code transforms *without* changing the code**

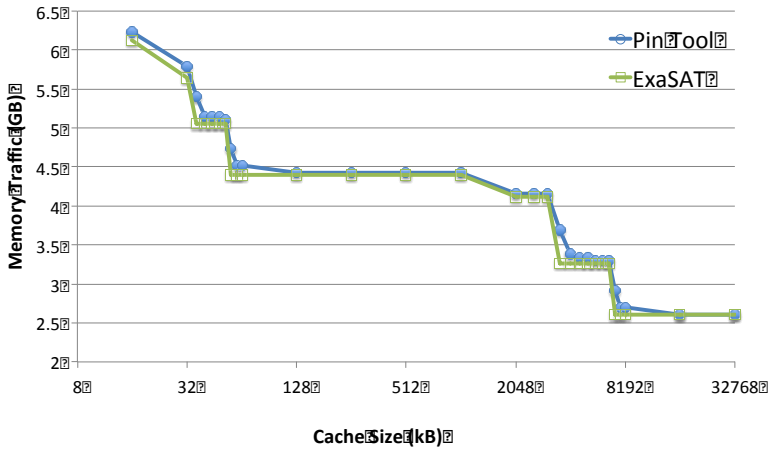


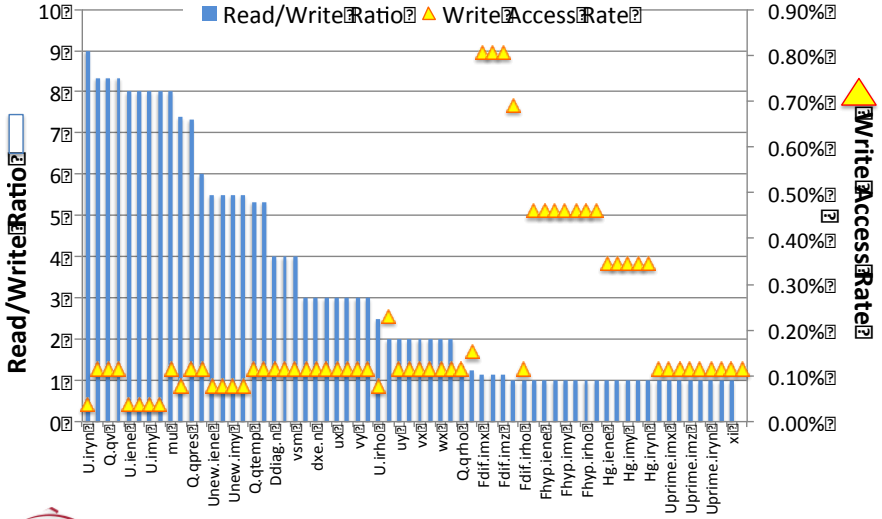
ExaSAT Analysis

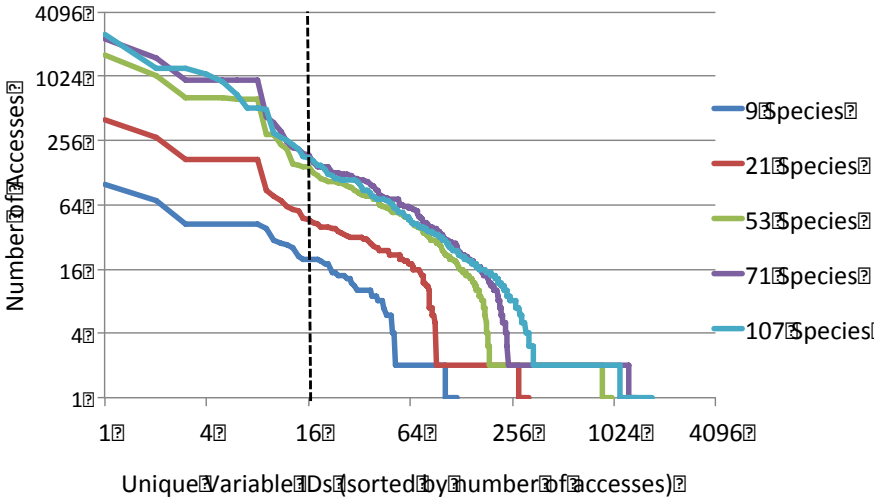
Automate a lot of tedious analysis



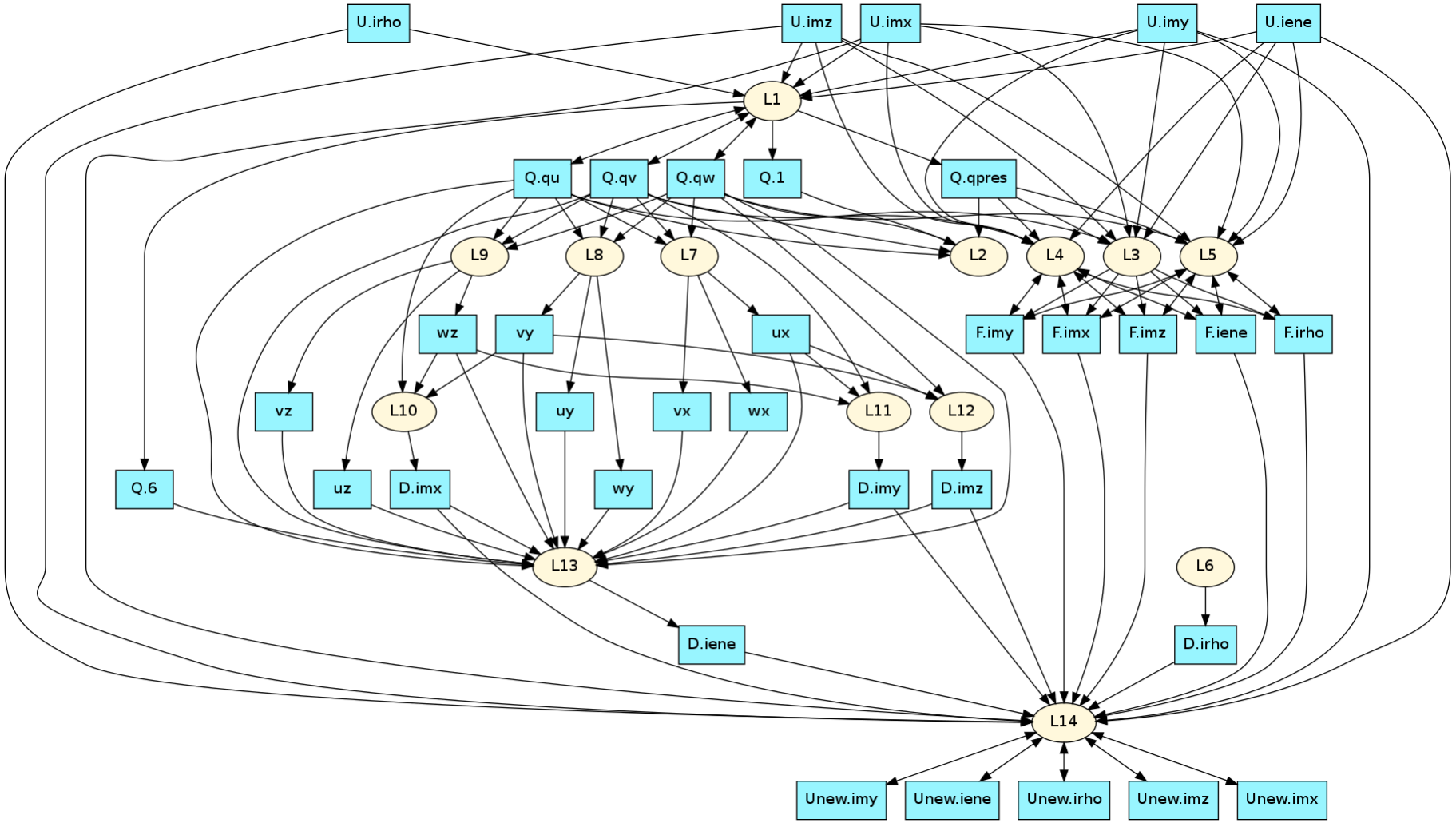






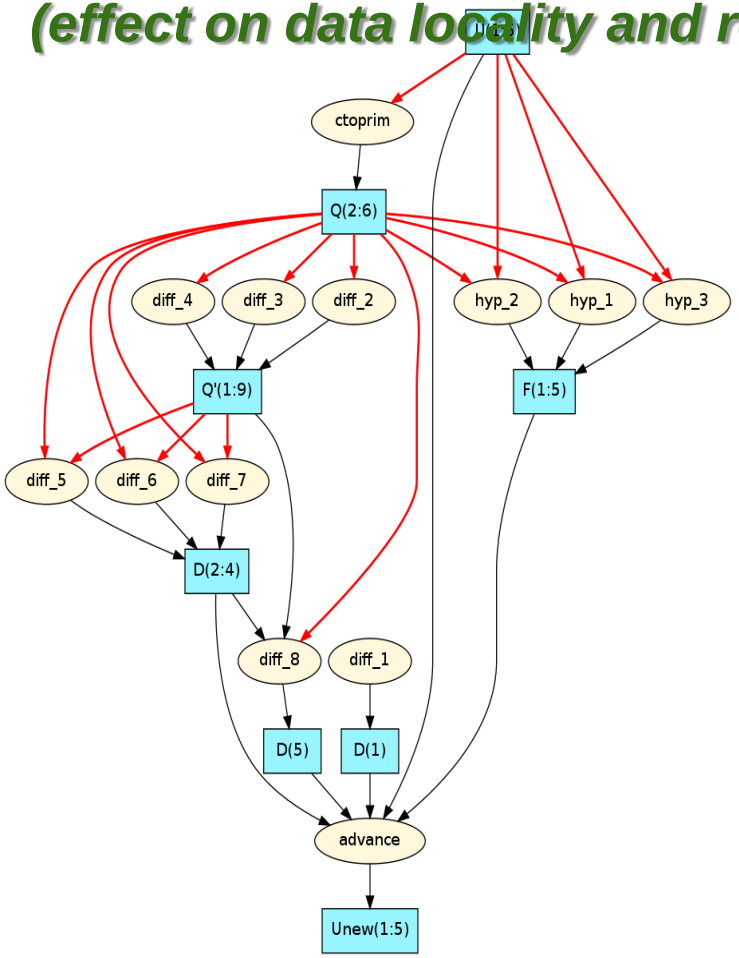


Deep Dependency Graph Analysis



Using Dependency Analysis to do Aggressive Fusion

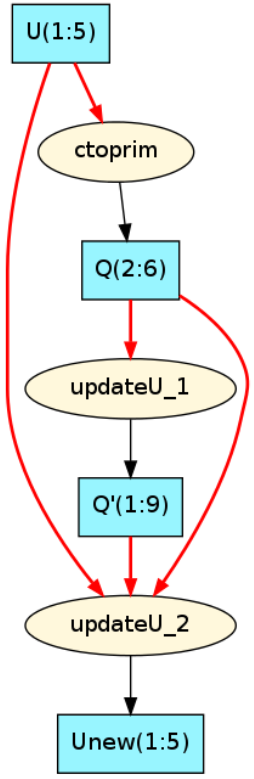
(effect on data locality and reuse distance)



Baseline

2.9 GB/sweep

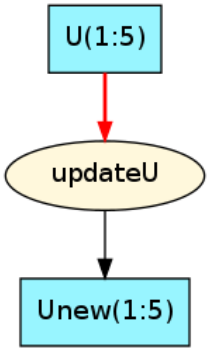
1.78 Bytes/Flop



Simple Fusion

1.6 GB/sweep (–46%)

0.96 Bytes/Flop

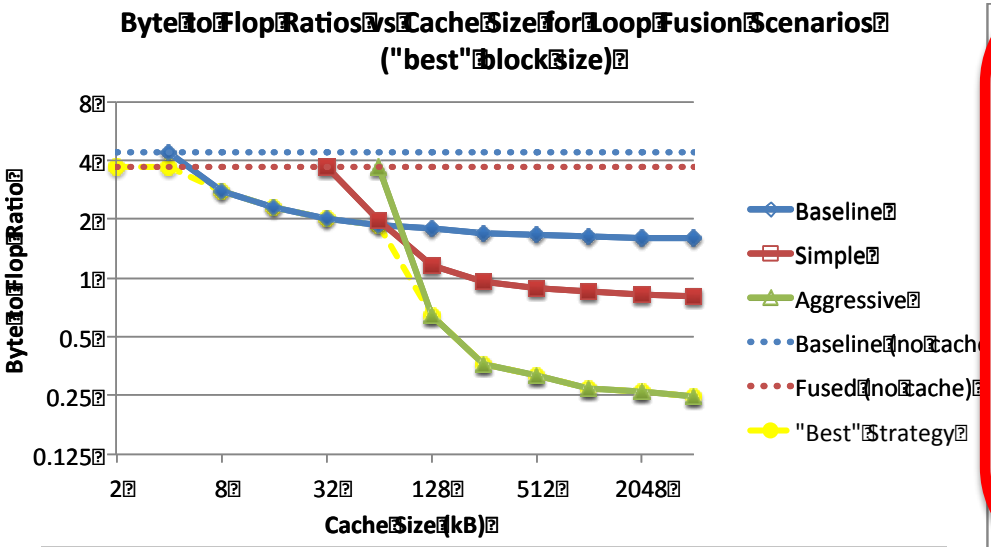


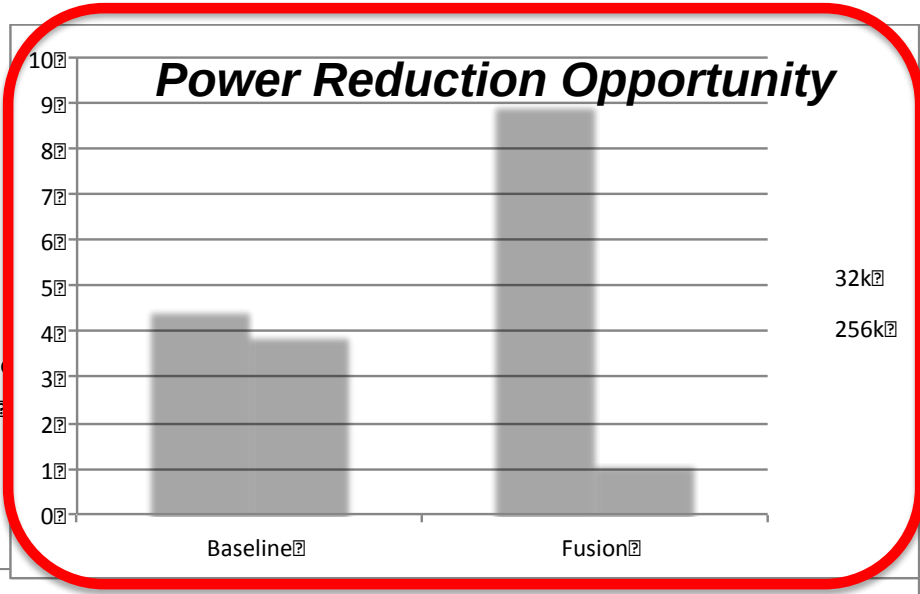
Aggressive Fusion

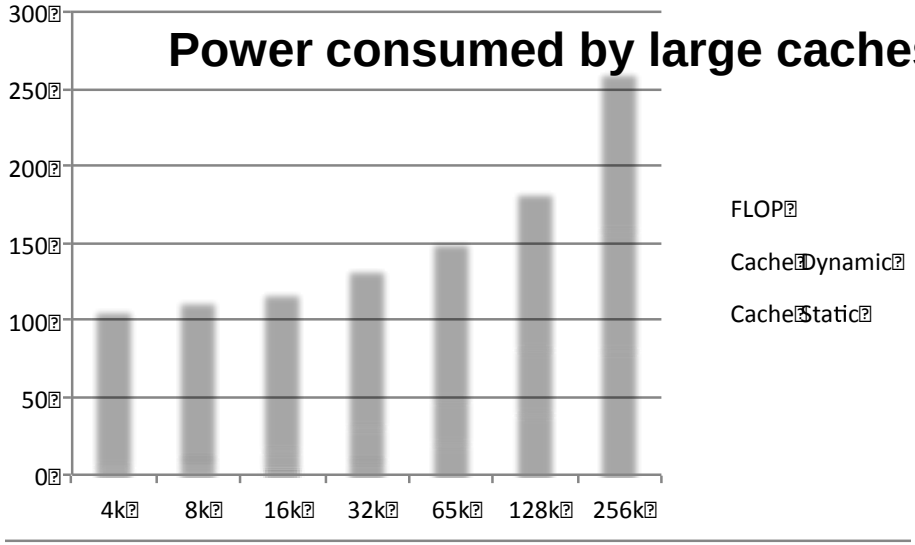
0.48 GB/sweep (–84%)

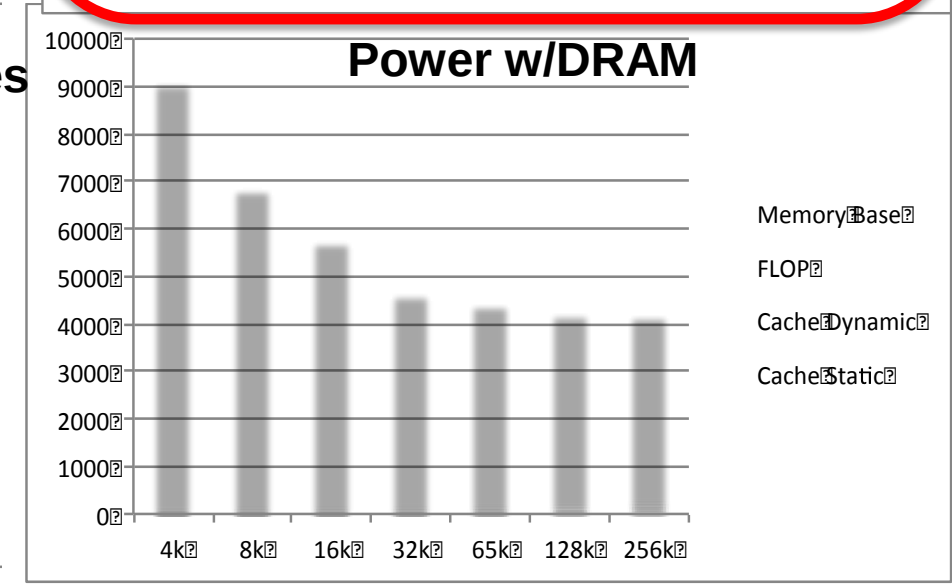
0.29 Bytes/Flop

Energy Efficiency Opportunity for Large L1 Scratchpads

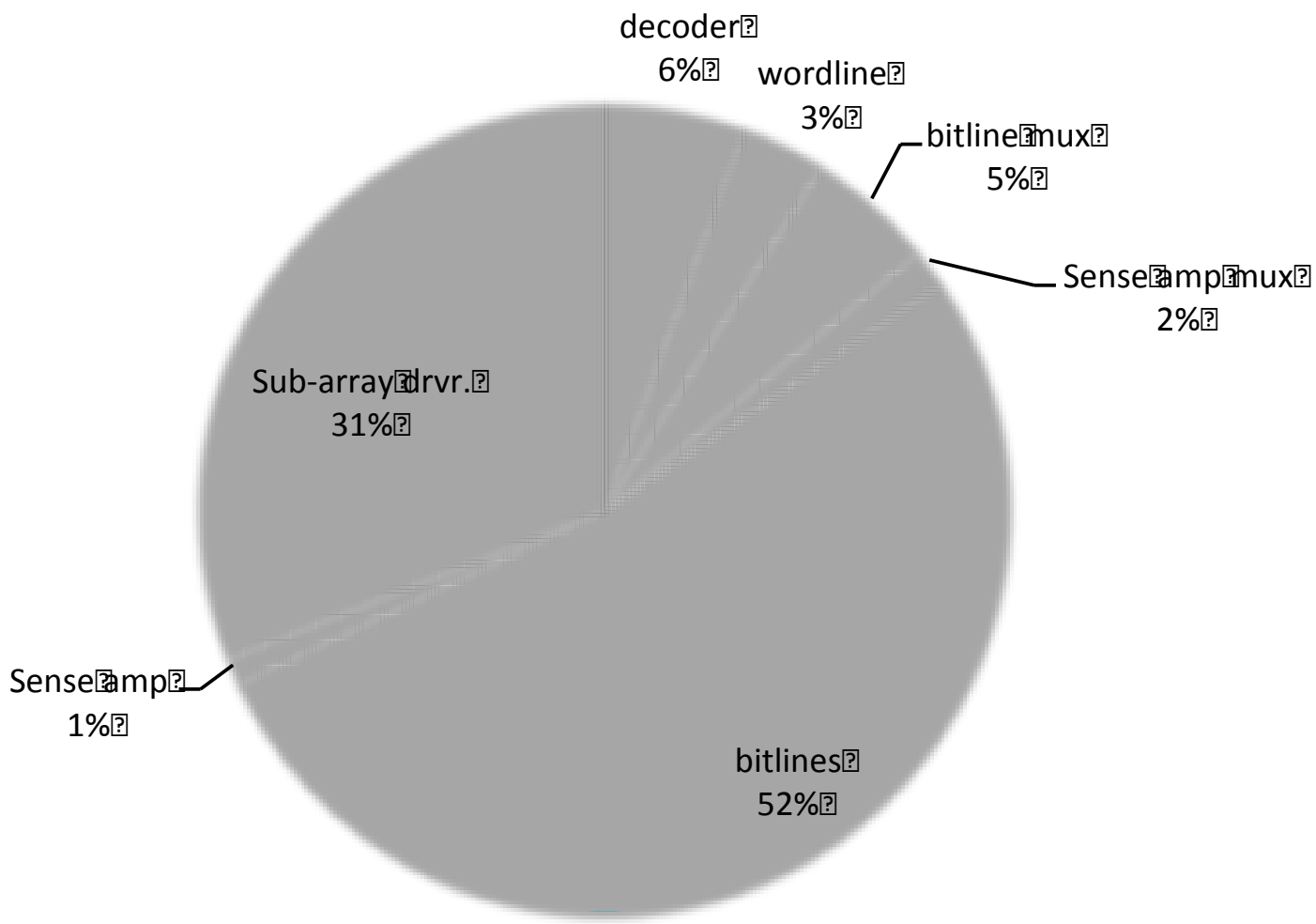




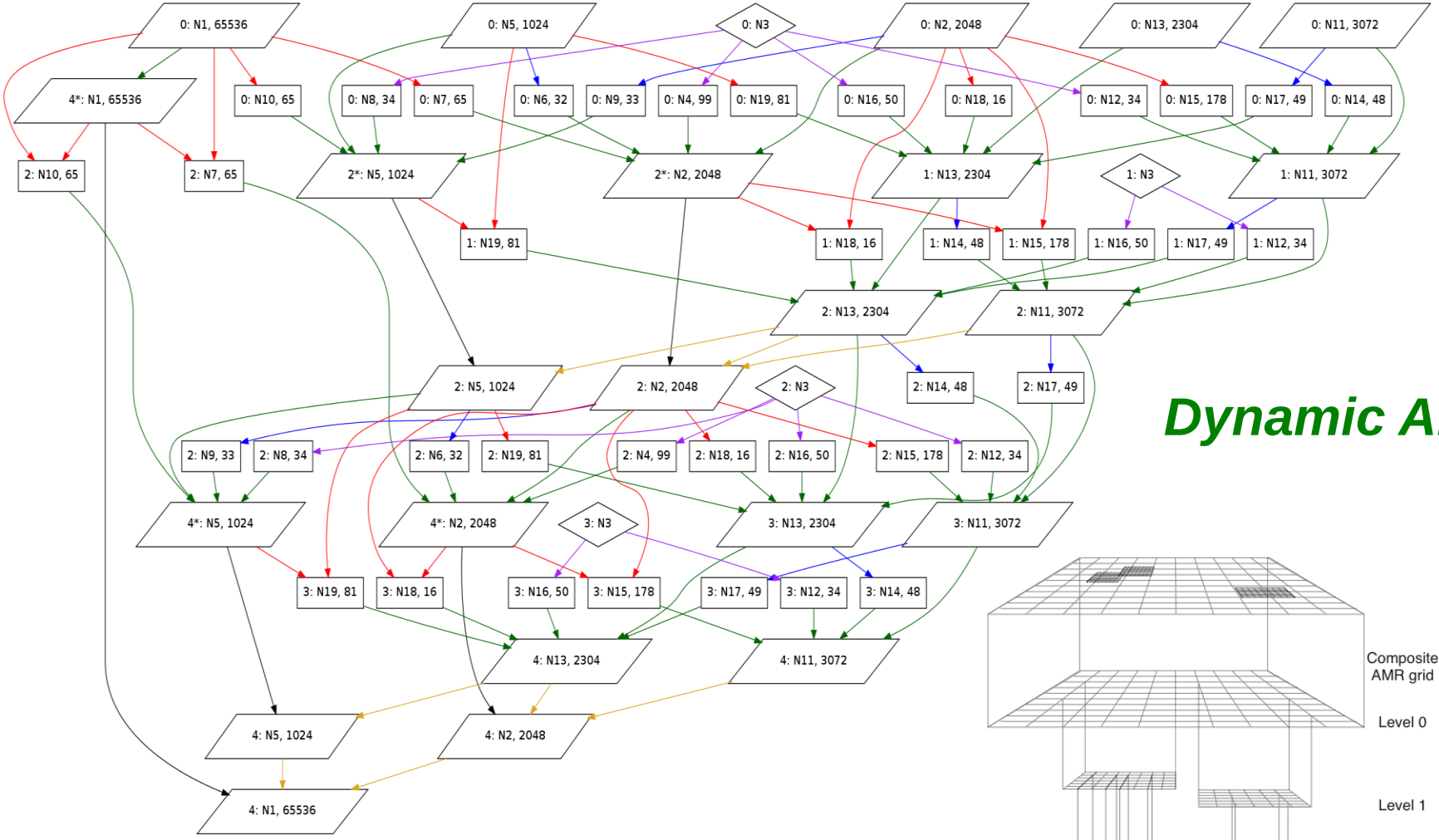




Power Breakdown for SRAM *(its mostly data movement)*



LMC (AMR) Code Control Dependencies

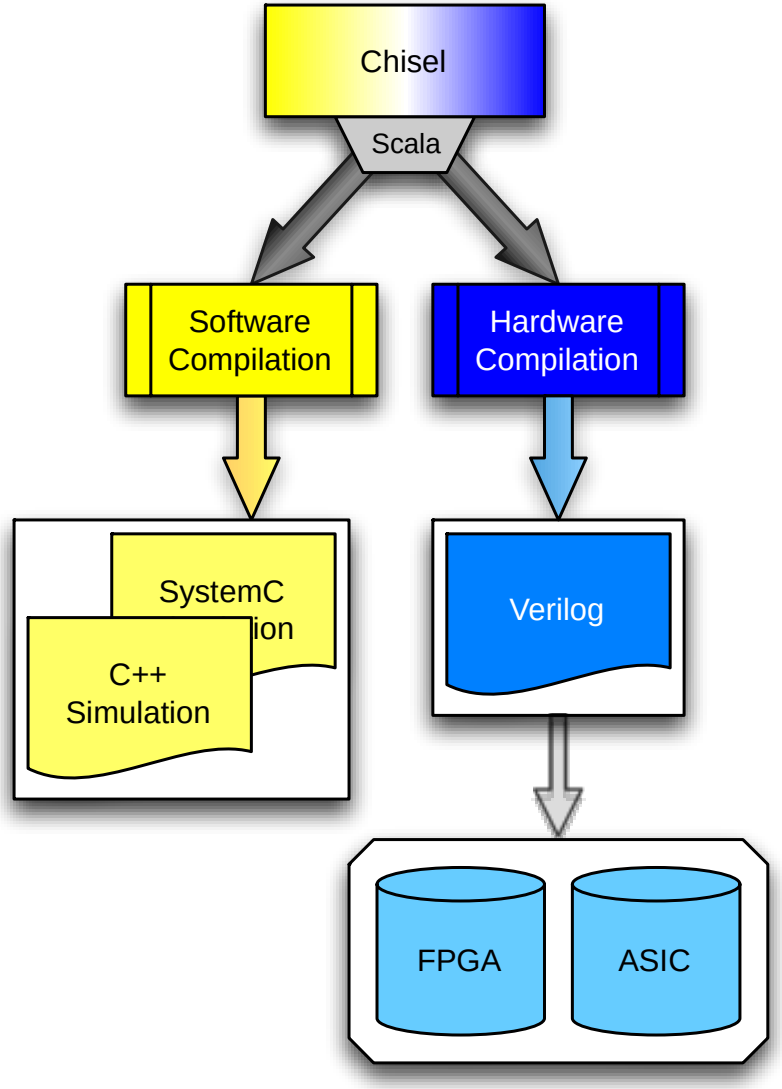


Dynamic Analysis

CoDEx Tools: *Putting Chisel to work for DOE*

A path to hardware and software models

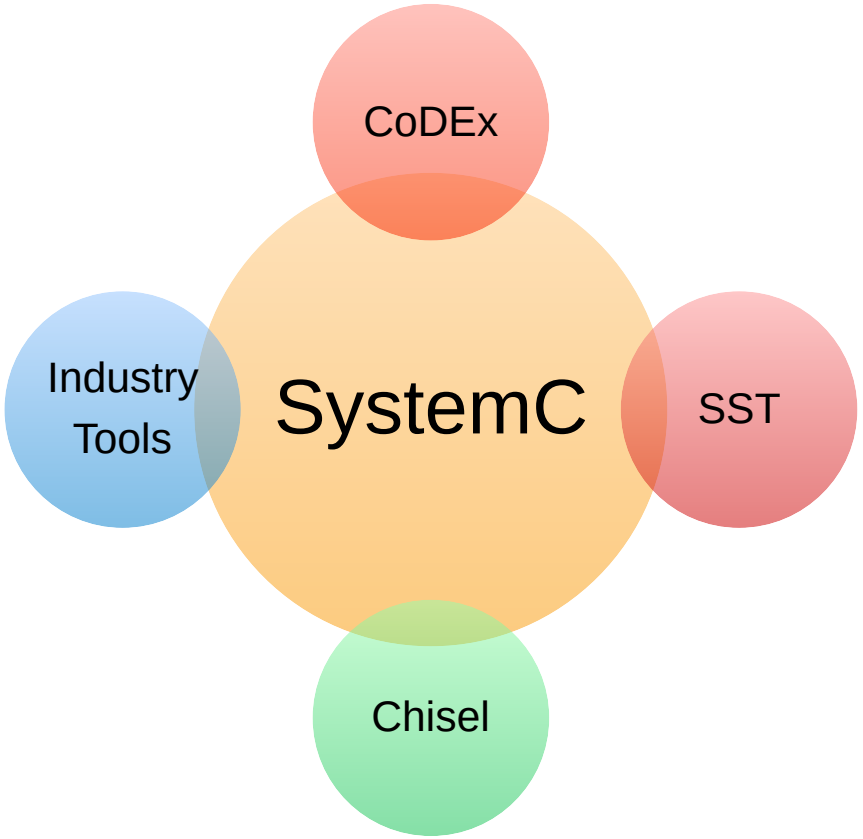
- **New hardware DSL**
- **Scala based**
  - Powerful generators
  - Obj Oriented constructs
- **Generate C++ and Verilog models from *single description***
- **Glue to existing infrastructure with SystemC**



CoDEx Interoperability

SystemC as a platform for re-use

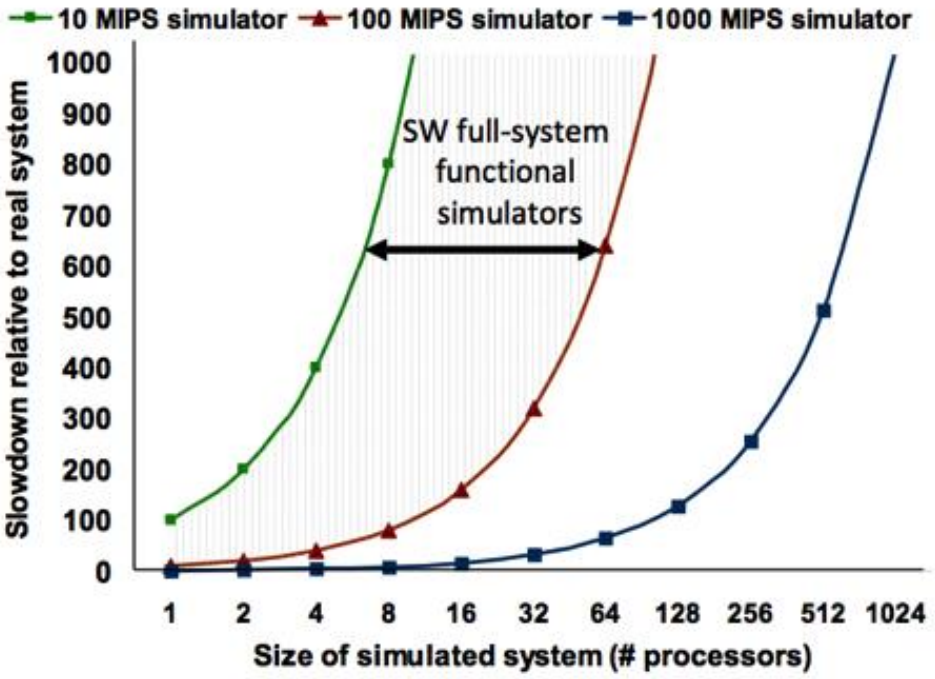
- **SystemC embraced by**
  - Industrial partners
    - Intel, Micron, ARM, Cadence, Synopsis
  - Other research efforts
    - CAL, FastForward
- **All CoDEx software simulation tools based on SystemC**



CoDEx Tools: FPGA Emulation

Enabling full application optimization

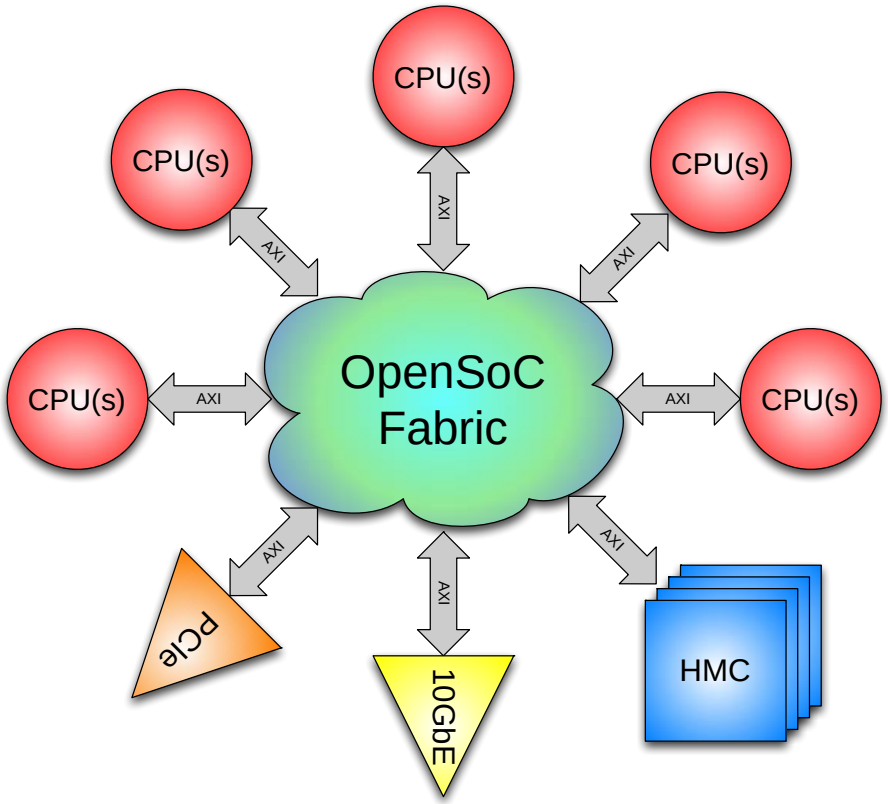
- **1000x Faster than SW models**
  - Scales independent of processor count
- **CoDEx Software models have parallel HW emulation path**
- **Performance and Energy counters provide SW model level of detail**



CoDEx Tools: OpenSoC Fabric

Parameterized NoC generation tool

- **Leverage Chisel to create highly parameterized, flexible model for NoC generation**
  - Dimensions, topology, VCs, etc. all configurable
  - Fast, functional SW model with SystemC integration
  - Verilog model for FPGA and ASIC flows
- **AXI Based Interface**
  - Integrate with Tensilica as well as ARM based cores
- **Builds on previous PhoenixSim network model**

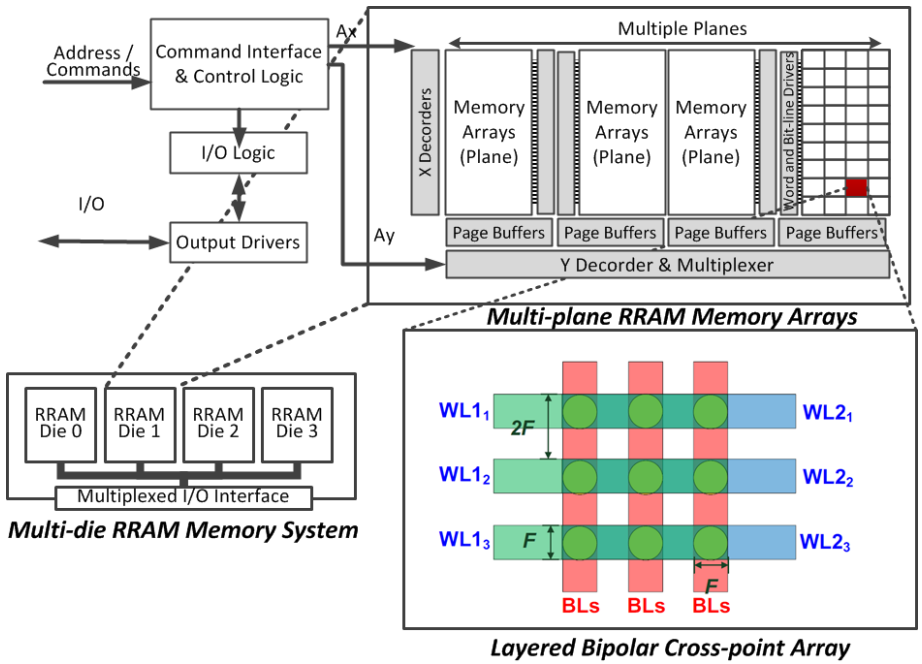


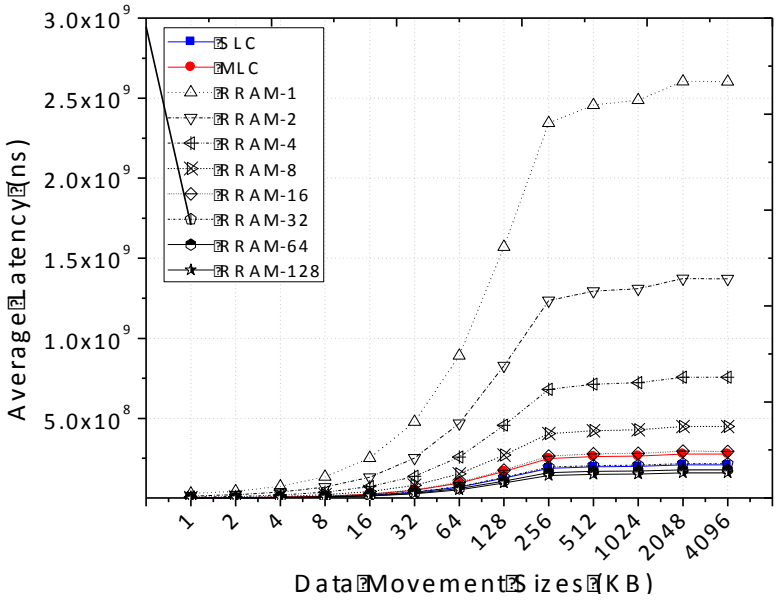
CoDEx Tools: NVRAMSim

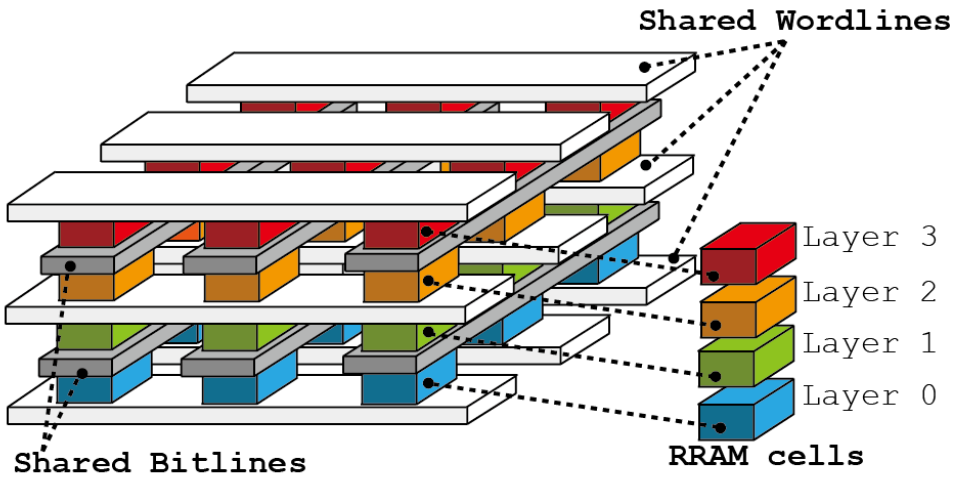
Non-Volatile Memory Modeling

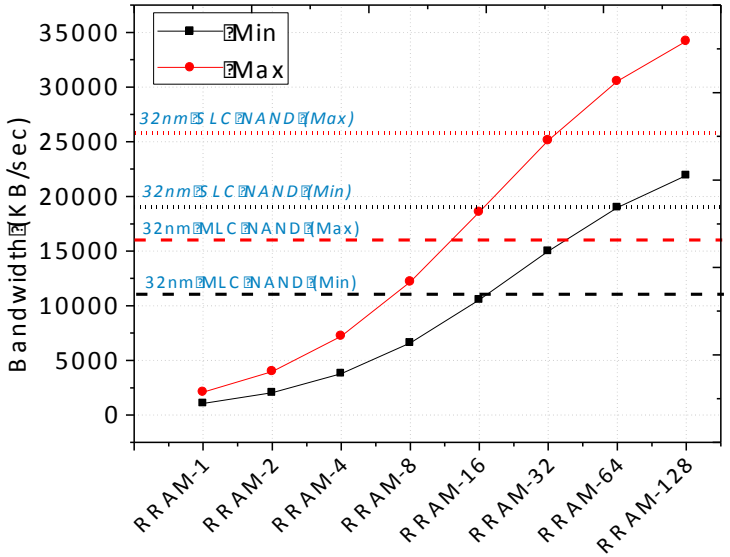
- **Collaboration with Myoungsoo Jung UT Austin**
- **Build on prior work of page/block addressable NVRAM simulator**
  - Extend to byte / word addressability
  - Alternative memory cell architecture
- **Dynamic energy model**
  - Aware of internal NVRAM components with variable clock frequencies
- **Validated against hardware evaluation platform**
- **Integrates with CoDEx processor and NoC models**
  - Supplements existing CoDEx DRAM model (DRAMSim2)

NRAMsim: *Alternative Burst Buffer RRAM Organization*









CoDEx Summary

- **CoDEx tools span range of abstraction levels**
- **Embrace SystemC as common glue**
  - Enables interoperation with DOE and industrial models
- **Validated processor model**
- **DRAM and NVRAM models**
- **Parallel modeling path includes**
  - Flexible software models
  - High-speed FPGA based emulation

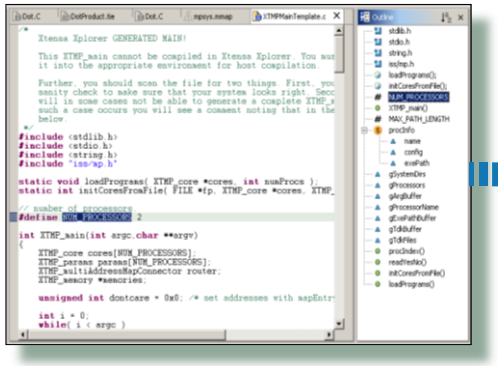
CoDEx Tools: Processor Models

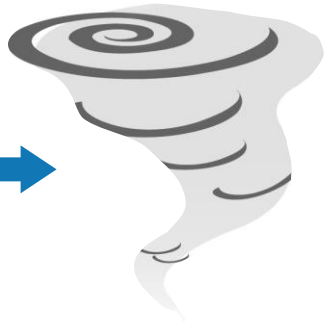
Highly configurable XTensa embedded core

- **Tensilica XTensa processor generator**
  - Fast configuration of cache, local store and easily extensible ISA
  - Highly flexible FIFO based interfaces (TIEQueues)
- **All custom cores generated with**
  - Fast functional model
  - SystemC based performance model
  - Verilog implementation ready for FPGA or ASIC flow
- **Verified power model**
- **Integrates with all CoDEx hardware and software models**

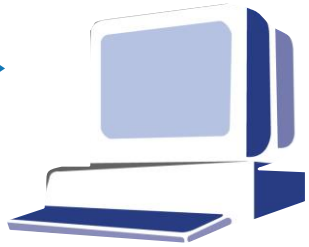
Embedded Processor Design

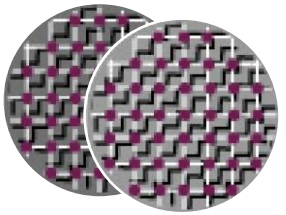
Example of existing Tensilica Design Flow

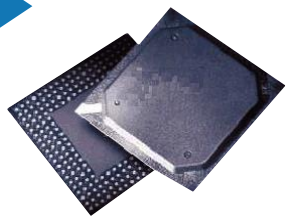




Application-optimized processor implementation (RTL/Verilog)







<u>Processor configuration</u>

1. Select from menu
2. Automatic instruction discovery (XPRES Compiler)
3. Explicit instruction description (TIE)

<u>Processor Generator</u>
(*Tensilica*)

Tailored SW Tools: Compiler, debugger, simulators, Linux, other OS Ports (Automatically generated together with the Core

Build your design with any process in any fab