6.375 Spring 2013
Final Projects

Richard Uhler

March 15, 2013

Final Projects

Build a complex digital design using FPGA

- Groups of 2 to 3 students
- Groups meet individually with Arvind, TA, Mentor weekly during assigned slot sometime 2:30-4pm Monday, Wednesday, or Friday in Arvind's office
- Weekly reports due day before the meeting, emailed in PDF format to 6.375-staff@mit.edu and your mentor

Schedule

Week	Date	Deliverable
0	Monday, March 18	Preliminary Proposal
0	Wednesday, March 20	Project Idea Presentation
1	Week of April 1	Final Proposal, High-Level Design and Test Plan
2	Week of April 8	Microarchitectural Description
3	Week of April 15	Implementation Status and Planned Exploration
4	Week of April 22	First Synthesis Results
5	Week of April 29	Simulation Demonstration
6	Week of May 6	FPGA Demonstration
7	Wednesday, May 15	Final Report, Final Presentation

Project Considerations

- Should benefit from using an FPGA
  - High performance matters,
  - Can take advantage of high parallelism,
  - Or otherwise makes sense to do in hardware

Project Considerations

- Should benefit from using an FPGA
  - High performance matters,
  - Can take advantage of high parallelism,
  - Or otherwise makes sense to do in hardware
- Application should be well understood
  - Is there accessible reference C code?
  - For domain specific applications, you should be familiar with the domain.

Project Considerations

- Should benefit from using an FPGA
  - High performance matters,
  - Can take advantage of high parallelism,
  - Or otherwise makes sense to do in hardware
- Application should be well understood
  - Is there accessible reference C code?
  - For domain specific applications, you should be familiar with the domain.
- Reuse of infrastructure *extremely* valuable
  - For example, reuse SMIPS or audio pipeline, or past years projects infrastructure.

Past Projects

Posted on Website under Projects

2010

- Ray Tracing
- Genetic Algorithm to Discover Efficient Sorting Networks
- Advanced Processor Design
- SMIPS SIMD
- Homomorphic Encryption
- Multi-Voice Audio Playback
- Pedestrian Detection

Past Projects

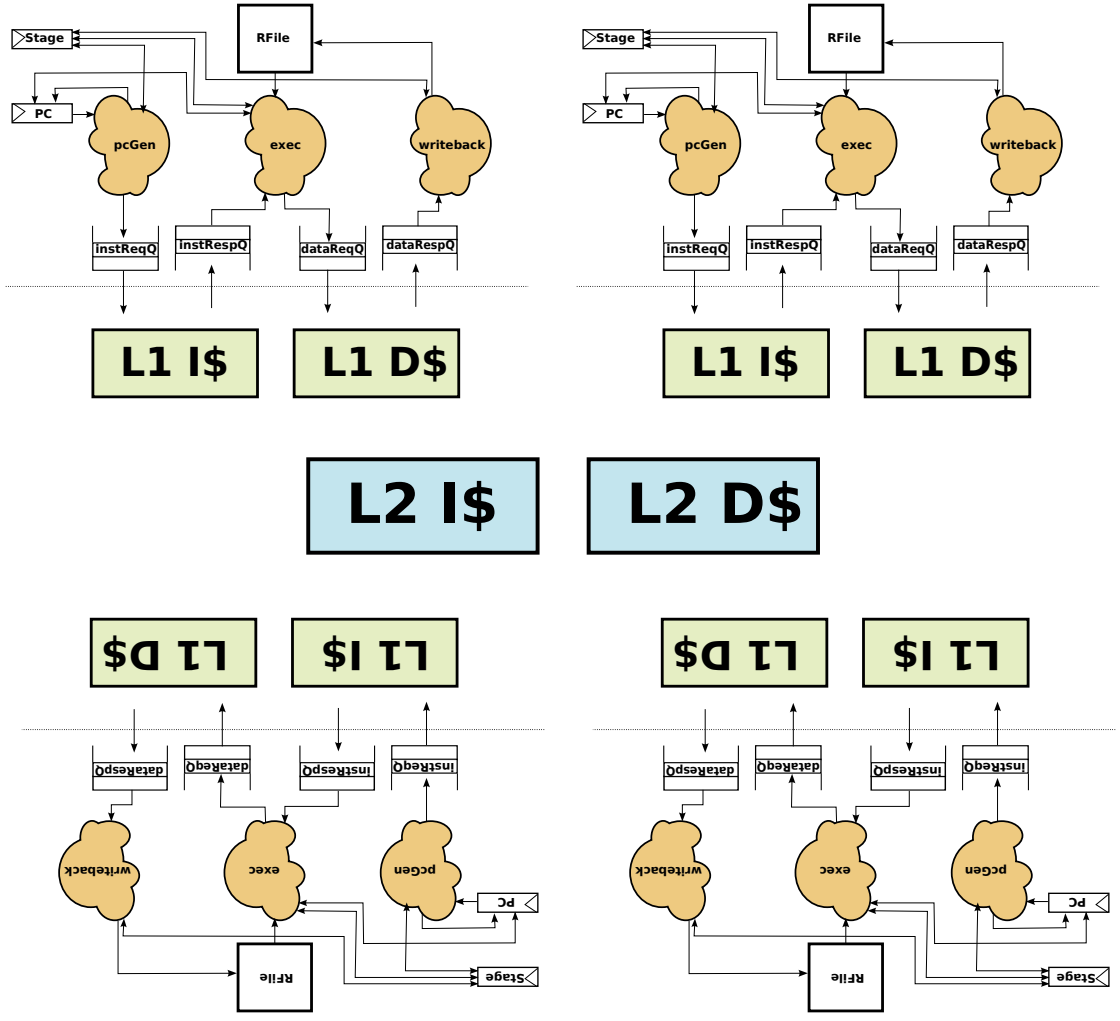
2011

- Rateless Wireless Networking with Spinal Codes
- Data Movement Control PowerPC
- Optical Flow Algorithm
- H.265 Motion Estimation
- Viterbi Decoder

Project Ideas

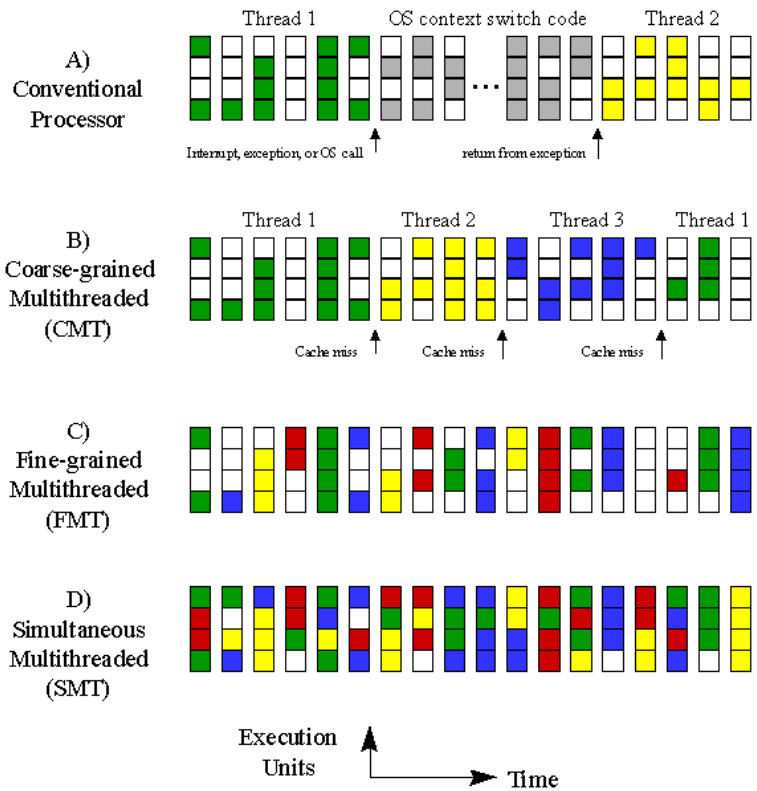
Multicore SMIPS

- Do something interesting with a Multicore SMIPS



Multithreaded SMIPS

- Implement an SMIPS processor that interleaves the execution of multiple threads in hardware
- You can experiment with cores support 2-8 threads
- Implement fine-grain, coarse-grain, or simultaneous multithreading.

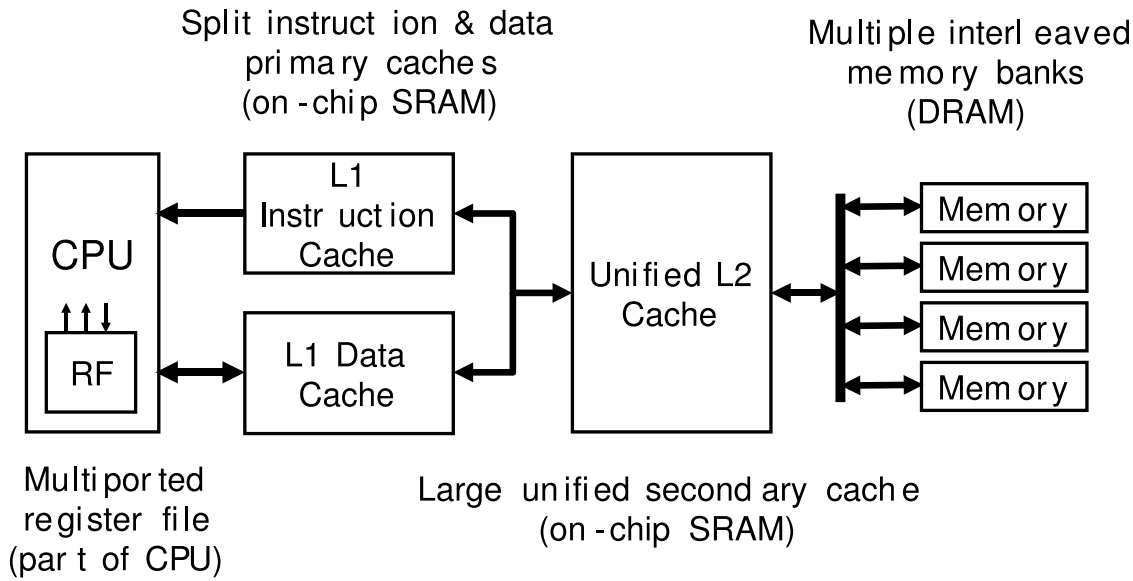


(http://www.realworldtech.com/page.cfm?articleid=RWT122600000000)

Cache Hierarchy Exploration with SMIPS

- Experiment with different types and levels of caching
- Try different: associativity, inclusivity, replacement policies

A Typical Memory Hierarc hy c.2006



Other SMIPS Project Ideas

Out-of-order superscalar SMIPS Processor

For example, using Tomasulo's algorithm for out-of-order execution with register renaming through reservation stations.

Other SMIPS Project Ideas

Out-of-order superscalar SMIPS Processor

For example, using Tomasulo's algorithm for out-of-order execution with register renaming through reservation stations.

SMIPS DSP Extensions

Use the SMIPS coprocessor interface to add a DSP accelerator to a basic SMIPS processor. You will need to extend the SMIPS ISA and write appropriate test/benchmark codes. Compare performance against baseline SMIPS.

Other SMIPS Project Ideas

Prefetching

Try implementing a hardware prefetcher to bring values into cache before the processor requests them. Stream buffers are one technique which predicts the stride of regular accesses.

Other SMIPS Project Ideas

Prefetching

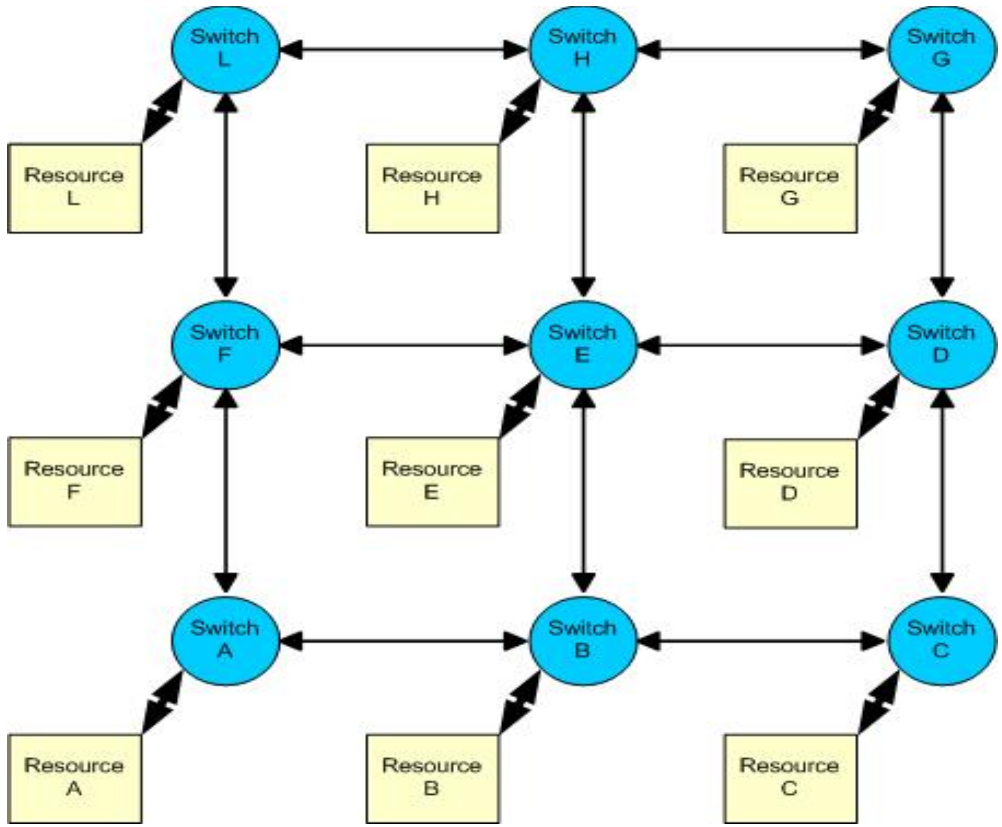
Try implementing a hardware prefetcher to bring values into cache before the processor requests them. Stream buffers are one technique which predicts the stride of regular accesses.

Compressed Memory Systems

Implement a compressed memory system, where cache lines are uncompressed when loaded into cache, and compressed again when evicted to main memory.

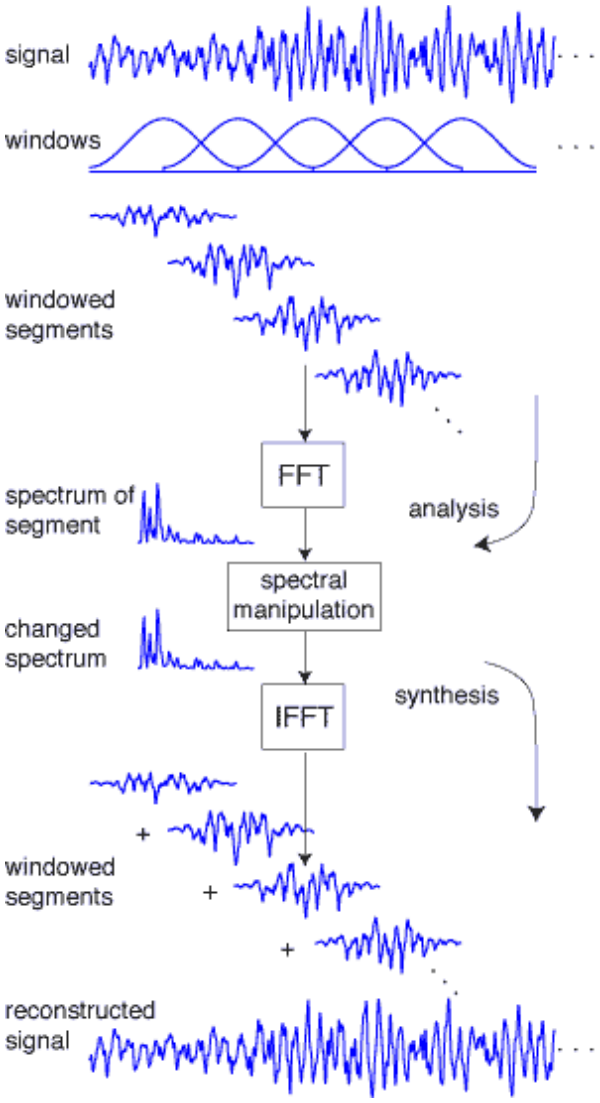
Modeling On Chip Networks

- Experiment with virtual channels, arbitration in 2D Mesh network.
- Processor elements could be: SMIPS, Special Processors, or just stubs



Resources =Cores =Processing Elements (P.E.)

High Quality Pitch Shifting Audio Pipeline



(http://sethares.engr.wisc.edu/vocoders/phasevocoder.html)

- Refactor Audio Pipeline from labs to work with 1024 point FFT and use other tricks to make it really sound good.

Generalized Sudoku Solver

Design Contest for 2009 International Conference on Field-Programmable Technology (http://fpt09.cse.unsw.edu.au/competition.html)

			7	16			11			9					10
	3								5	11		4			
1			15			9			6					5	
							14	7					3		
		6		2		5		12							7
		3		14										6	
4								16		2			5		
8	2				4										12
10				1				8	4			15			
		8												7	
			5			15							2		
				13						3					
						4					1				
	10				5									13	
		5							2				9		
			8			16						11			

SAT Solver

- Given Boolean formula in conjunctive normal form, figure out if any assignment of variables makes the formula true
- Satisfiability is NP-Complete

$$(A \vee B) \wedge (\neg B \vee C \vee \neg D) \wedge (D \vee \neg E)$$