ECE 5745 Complex Digital ASIC Design
Course Overview

Christopher Batten

School of Electrical and Computer Engineering
Cornell University

http://www.csl.cornell.edu/courses/ece5745
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - Why should students want to take this course?
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
The Computer Systems Stack

Gap too large to bridge in one step
(but there are exceptions e.g., magnetic compass)
In its broadest definition, computer architecture is the design of the abstraction/implementation layers that allow us to execute information processing applications efficiently using available manufacturing technologies.
In its broadest definition, computer architecture is the design of the abstraction/implementation layers that allow us to execute information processing applications efficiently using available manufacturing technologies.

**Application Requirements**
- Provide motivation for building system
- SW/HW interface expressive yet productive

**Technology Constraints**
- Restrict what can be done efficiently
- New technologies make new arch possible

Computer architects provides feedback to guide application and technology research directions.
Key Metrics in Computer Architecture

Primary Metrics
- Execution time (cycles/task)
- Energy (Joules/task)
- Cycle time (ns/cycle)
- Area (µm²)

Secondary Metrics
- Performance (ns/task)
- Average power (Watts)
- Peak power (Watts)
- Cost ($)
- Design complexity
- Reliability
- Flexibility

Discuss qualitative first-order analysis from ECE 4750 on board
How can we quantitatively evaluate area, cycle time, and energy?

How do we actually implement processors, memories, and networks in a real chip?

How should we implement/analyze application-specific accelerators?

- Very loosely coupled memory-mapped accelerators
- More tightly coupled co-processor accelerators
- Specialized instructions and functional units
ASIC: Application-Specific Integrated Circuit

- Course Goal, Structure, Motivation
- Activity
- ASIC Design Case Studies

Network

Accelerated Instructions

Simple Proc

Processor Power Constraint

Out-of-Order Superscalar Superpipelined

Superscalar w/ Deeper Pipelines

Multicore

Specialized Accelerators

Energy (Joules per Task)

Performance (Tasks per Second)
ASIC: Application-Specific Integrated Circuit
Goal for ECE 5745 is to answer these questions!

- How can we quantitatively evaluate area, cycle time, and energy?
- How do we actually implement processors, memories, and networks in a real chip?
- How should we implement/analyze application-specific accelerators?
  - Very loosely coupled memory-mapped accelerators
  - More tightly coupled co-processor accelerators
  - Specialized instructions and functional units
Full Custom Design vs. Standard-Cell Design

- **Full-Custom Design (ECE 4740)**
  - Designer is free to do anything, anywhere; though team usually imposes some design discipline
  - Most time consuming design style; reserved for very high performance or very high volume chips (Intel microprocessors, RF power amps for cellphones)

- **Standard-Cell Design (ECE 5745)**
  - Fixed library of “standard cells” and SRAM memory generators
  - Register-transfer-level description is automatically mapped to this library of standard cells, then these cells are placed and routed automatically
  - Enables agile hardware design methodology
Standard-Cell Design Methodology

- Also called Cell-Based ICs (CBICs)
- Fixed library of cells plus memory generators
- Cells can be synthesized from HDL, or entered in schematics
- Cells placed and routed automatically
- Requires complete set of custom masks for each design
- Currently most popular hard-wired ASIC type (6.884 will use this)

Cells arranged in rows
Generated memory arrays

Clock Rail (not typical)
Power Rails in M1
NAND2
GND Rail
Flip-flop

Well Contact under Power Rail
Cell I/O on M2

Ripple carry adder with carry chain highlighted
Standard-Cell Design Methodology

- Design in HDL
- Standard Cells
- Switching Activity
- Gate-Level Model
- Place&Route
- Layout
- Power Analysis

<table>
<thead>
<tr>
<th>HDL Simulator</th>
<th>Synthesis</th>
<th>Place&amp;Route</th>
<th>Power Analysis</th>
</tr>
</thead>
</table>

- Area ($\mu$m$^2$)
- Cycle Time (ns)
- Energy (J/task)

Execution Time (cycles/task)
Example Standard-Cell Chip Plot

Control Processor 8.1%
Vector Register File 56.9%
Vector Integer ALUs 9.7%
Vector FPUs 9.4%
Vector Memory Units 7.6%
Other 8.3%

810 μm
What is Complex Digital ASIC Design?

Complex digital ASIC design is the process of quantitatively exploring the area, cycle time, execution time, and energy trade-offs of various application-specific accelerators (and general-purpose proc+mem+net) using automated standard-cell CAD tools and then to transform the most promising design to layout ready for fabrication.
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - **Why should students want to take this course?**
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
Technology Scaling is Slowing

- Course Goal, Structure, Motivation
- Activity
- ASIC Design Case Studies

System Performance


- Vacuum Tube
- Discrete Transistor
- Integrated Bipolar
- Integrated CMOS
- 7nm, ~50B Transistors

New Technology?
Technology Fallow Period?
Golden Age of Chip Design?
OR

Adapted from D. Brooks Keynote at NSF XPS Workshop, May 2015.
Example Application Domain: Image Recognition
Machine Learning: Training vs. Inference

Training
- Many images
- Model
- Forward "starfish"
- Backward error
- Labels

Inference
- Few images
- Model
- Forward "dog"
ImageNet Large-Scale Visual Recognition Challenge

Top 5 Error Rate

- '10: 28%
- '11: 26%
- '12: 16%
- '13: 12%
- '14: 7%
- '15: 3.6%
- '16: 3%
- '17: 2.3%

Entries Using GPUs

- 2010: 0%
- 2011: 0%
- 2012: 14%
- 2013: 74%
- 2014: 89%
- 2015: ~100%
- 2016: 89%
- 2017: ~100%

Software: Deep Neural Network

Hardware: Graphics Processing Units
Accelerators for Machine Learning in the Cloud

NVIDIA DGX Hopper
- Graphics processor specialized just for accelerating machine learning
- Available as part of a complete system with both the software and hardware designed by NVIDIA

Google TPU v4
- Custom chip specifically designed to accelerate Google’s TensorFlow C++ library
- Tightly integrated into Google’s data centers

Microsoft Catapult
- Custom FPGA board for accelerating Bing search and machine learning
- Accelerators developed with/by app developers
- Tightly integrated into Microsoft data center’s and cloud computing platforms
Accelerators for Machine Learning at the Edge

Amazon Echo
- Developing AI chips so Echo line can do more on-board processing
- Reduces need for round-trip to cloud
- Co-design the algorithms and the underlying hardware

Facebook Oculus
- Starting to design custom chips for Oculus VR headsets
- Significant performance demands under strict power requirements

Movidius Myriad 2

ECE 5745 Course Overview
Top-five software companies are all building custom accelerators

- **Facebook**: w/ Intel, in-house AI chips
- **Amazon**: Echo, Oculus, networking chips
- **Microsoft**: Hiring for AI chips
- **Google**: TPU, Pixel, convergence
- **Apple**: SoCs for phones and laptops

Chip startup ecosystem for machine learning accelerators is thriving!

- Graphcore
- Nervana
- Cerebras
- Wave Computing
- Horizon Robotics
- Cambricon
- DeePhi
- Esperanto
- SambaNova
- Eyeriss
- Tenstorrent
- Mythic
- ThinkForce
- Groq
- Lightmatter
The field of complex digital ASIC design is experiencing a disruptive sea change and has a critical choice:

1. A technological fallow period
2. A golden age of ASIC design

This course will help you appreciate and possibly contribute to this golden age!
Course Motivation: Comp Arch Research Perspective

Cross-Layer Interaction is Critical

Architecture-level researchers need to quantitatively understand area, cycle time, and energy trade-offs to create new architectures for the accelerator era.

Cross-layer interaction can generate some of the most exciting research ideas!
Course Motivation: Circuits Research Perspective

Your Digital Circuit Here

Your Analog Circuit Here

Apple M2 System-on-Chip (2022)
20 Billion transistors
Cross-Layer Interaction is Critical

Circuit-level researchers need to appreciate the system-level context for their circuits.

Cross-layer interaction can generate some of the most exciting research ideas!
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - Why should students want to take this course?
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
Course Structure

Part 1
ASIC Design Overview

Part 2
Digital CMOS Circuits

Part 3
CAD Algorithms

Prereq
Computer Architecture

Part 2
Digital CMOS Circuits

Part 3
CAD Algorithms

Course Goal, Structure, Motivation
Part 1: ASIC Design Overview

- **Topic 1**: Hardware Description Languages
- **Topic 2**: CMOS Devices
- **Topic 3**: CMOS Circuits
- **Topic 4**: Full-Custom Design Methodology
- **Topic 5**: Automated Design Methodologies
- **Topic 6**: Closing the Gap
- **Topic 7**: Clocking, Power Distribution, Packaging, and I/O
- **Topic 8**: Testing and Verification
Part 2: Digital CMOS Circuits

Topic 9: Combinational Logic

Topic 10: Sequential State

Topic 11: Interconnect
Part 3: CAD Algorithms

**Topic 12**
Synthesis Algorithms

- **RTL to Logic Synthesis**
  - \[ x = a'bc + a'bc' \]
  - \[ y = b'c' + ab' + ac \]

- **Technology Independent Synthesis**
  - \[ x = a'b \]
  - \[ y = b'c' + ac \]

- **Technology Dependent Synthesis**

**Topic 13**
Physical Design Automation

- **Placement**
- **Global Routing**
- **Detailed Routing**
Five-Week Design Project

- Course Goal, Structure, Motivation
- Activity
- ASIC Design Case Studies

Performance (Tasks per Second)
Energy Efficiency (Tasks per Joule)
Simple Processor
High-Performance Architectures
Flexibility vs. Specialization

- Design Power Constraint
- Design Performance Constraint
- Embedded Architectures
- Custom ASIC
- Less Flexible Accelerator
- More Flexible Accelerator

- Network
- Accelerated Instructions
- P
- Xcel
- D$
- P

- Xcel
- D$
- Xcel
- D$
- D$
- D$
- D$
- D$
- D$
- D$
- D$

ECE 5745 Course Overview 34 / 58
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - Why should students want to take this course?
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
Fixed-Latency Iterative Multiplier Datapath

- Activity

![Fixed-Latency Iterative Multiplier Datapath Diagram]
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - Why should students want to take this course?
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
Scalar Processors with Multithreading

Programmer's Logical View:
- 
- 
- 
- 
- 

Typical Core Micro-Architecture:
- Instr Memory
- Data Memory
- Multi-threaded Cores

Energy vs. Performance:
- MIMD
Vector-SIMD Processors

Programmer's Logical View

Typical Core Micro-Architecture

Instruction Memory

Vector Lanes

Data Memory

Energy

Performance

MIMD

Vector-SIMD
Quantitative Area Evaluation

Control Processor 8.1%
Vector Register File 56.9%
Vector Integer ALUs 9.7%
Vector FPUs 9.4%
Vector Memory Units 7.6%
Other 8.3%

810 µm
Quantitative Area Evaluation

- Quad-Core w/ Vertical Multithreading
- Vector-SIMD (8 elm/lane)

Single-core multi-lane design reduces area by 15%
Multi-core single-lane design increases area by 20%
Quantitative Performance and Energy Evaluation

Multithreaded Multicore (increasing number of threads per core)

Performance reduction with increasing threads due to increased cycle time and thread management overhead on fine-grain loops

Single-Lane Vector (increasing vlen)

Normalized Tasks / Second

Normalized Energy / Task

Normalized Energy / Task

Normalized Tasks / Second

Energy / Task (μJ)

Quad-Core w/ Vertical Multithreading

Vector-SIMD (4 core w/ 1 lane)
Complex Digital ASIC Design

- Course goal, structure, motivation
  - What is the goal of the course?
  - Why should students want to take this course?
  - How is the course structured?

- Activity: Evaluation of Integer Multiplier

- ASIC Design Case Studies
  - Example design-space exploration
  - Example real ASIC chips
Simple RISC Processor ASIC

- ASIC Design Case Studies
- Simple RISC Processor ASIC
  - SP Datapath
  - SP Regfile
  - RAM Subbank (2KB)
  - RAM Subbank (2KB)
  - RAM Subbank (2KB)
  - RAM Subbank (2KB)
  - AHIP Controller
  - SP Control
  - SP Datapath
  - RAM Interface
  - VCO

- Host PC
- PLX 9050 ATB0 Controller
- Power Supplies
- Probe Points
- PCI SDRAM
- STC1 Host ATB0 Daughter Card
- Test System Block Diagram

- The test system includes the ATB0 test baseboard, a daughter card with the STC1 chip, and a host computer to control the test system.
Simple RISC Processor ASIC

- RISC processor w/ 8 KB SRAM
- TSMC 0.18 µm process
- 1.7 × 2.1 mm
- 610K Transistors
- 450 MHz at 1.8 V
Scale Vector-Thread Processor ASIC
Scale Vector-Thread Processor ASIC

TSMC 0.18µm • 7.14 Million Transistors • 16.6 mm² Core Area
Scale Energy vs. Performance Results

Power consumption ranges from 0.4 to 1.1 Watts at 260 MHz

ECE 5745 Course Overview
Batten Research Group Test Chips

TSMC 180nm, 28nm, 16nm; SkyWater 130nm
GF 130nm, 12nm; Intel 22FFL

Simple RISC-V cores
Coarse-grain reconfigurable arrays
Clustered manycore architectures

Mesh on-chip networks
Crossbar interconnects
BRG Test Chip 1 (2016)

Post-Silicon Evaluation Strategy
The testing platform enables running small test programs on BRGTC1 to compare the performance and energy of pure-software kernels versus the HLS-generated sorting accelerator.

Taped-out Layout for BRGTC1
2x2mm 1.3M transistors in IBM 130nm RISC processor, 16KB SRAM HLS-generated accelerators
Static Timing Analysis Freq. @ 246 MHz
Celerity System-on-Chip Overview (2017)

Target Workload: High-Performance Embedded Computing

- 5 × 5mm in TSMC 16 nm FFC
- 385 million transistors
- 511 RISC-V cores
  - 5 Linux-capable Rocket cores
  - 496-core tiled manycore
  - 10-core low-voltage array
- 1 BNN accelerator
- 1 synthesizable PLL
- 1 synthesizable LDO Vreg
- 3 clock domains
- 672-pin flip chip BGA package
- 9-months from PDK access to tape-out
BRG Test Chip 2 (2018)

Block Diagram
4xRV32IMAF cores with “smart” sharing L1$/LLFU, synthesizable PLL

Taped-out Layout for BRGTC2
2x2mm, 1.2M-trans, IBM 130nm Static Timing Analysis Freq. @ 500MHz
BRG Test Chip 3/4 (2020/2021)

- ASIC Design Case Studies

ECE 5745 Alumni Tape-Out!

- 2x2.5mm, TSMC 180nm
- SPI minion interface
- Open-source FPU
- Synthesizable digital clock generator
- BRGTC3 had hold time issue in the SPI minion
- BRGTC4 fully functional
ECE 5745 Teaching Tapeout (2022)

- First teaching tapeout in 10 years
  - SkyWater 130nm through efabless
  - Taped out using completely open-source EDA tools!

- Four student projects
  - CRC32 checksum unit implemented using C++ HLS
  - Latency insensitive synthesizable memory implemented in PyMTL3
  - 2x2 systolic array multiplier implemented in SystemVerilog
  - Greatest common divisor unit implemented in SystemVerilog
  - Each unit included dedicated SPI interface
BRG Test Chip #5 (2022)

RISC-V RV32IM core with 32-KB of SRAM
SPI minion for config; SPI master and GP I/O for peripherals
2x2.5mm, TSMC 180nm
100% done using PyMTL3 by ECE 5745 Alumni
BRG Test Chip #5 (2022)
BRG Test Chip #5 (2022)

Simulated and Measured Energy per Instruction at 66 MHz and 3.3 V Core Voltage

- **Measured**
- **ctrl**
- **ldiv**
- **imul**
- **ALU**
- **RF**
- **dmem**
- **imem**

---

ECE 5745 Course Overview
Take-Away Points

- Complex digital ASIC design is the process of quantitatively exploring the area, cycle time, execution time, and energy trade-offs of general-purpose and application-specific designs using automated standard-cell CAD tools and then to transform the most promising design to layout ready for fabrication.

- Course provides an excellent foundation for students interested in pursuing a career in industry development of ASICs or can provide useful experience with cross-layer interaction for students interested in pursuing research in computer architecture or circuits.