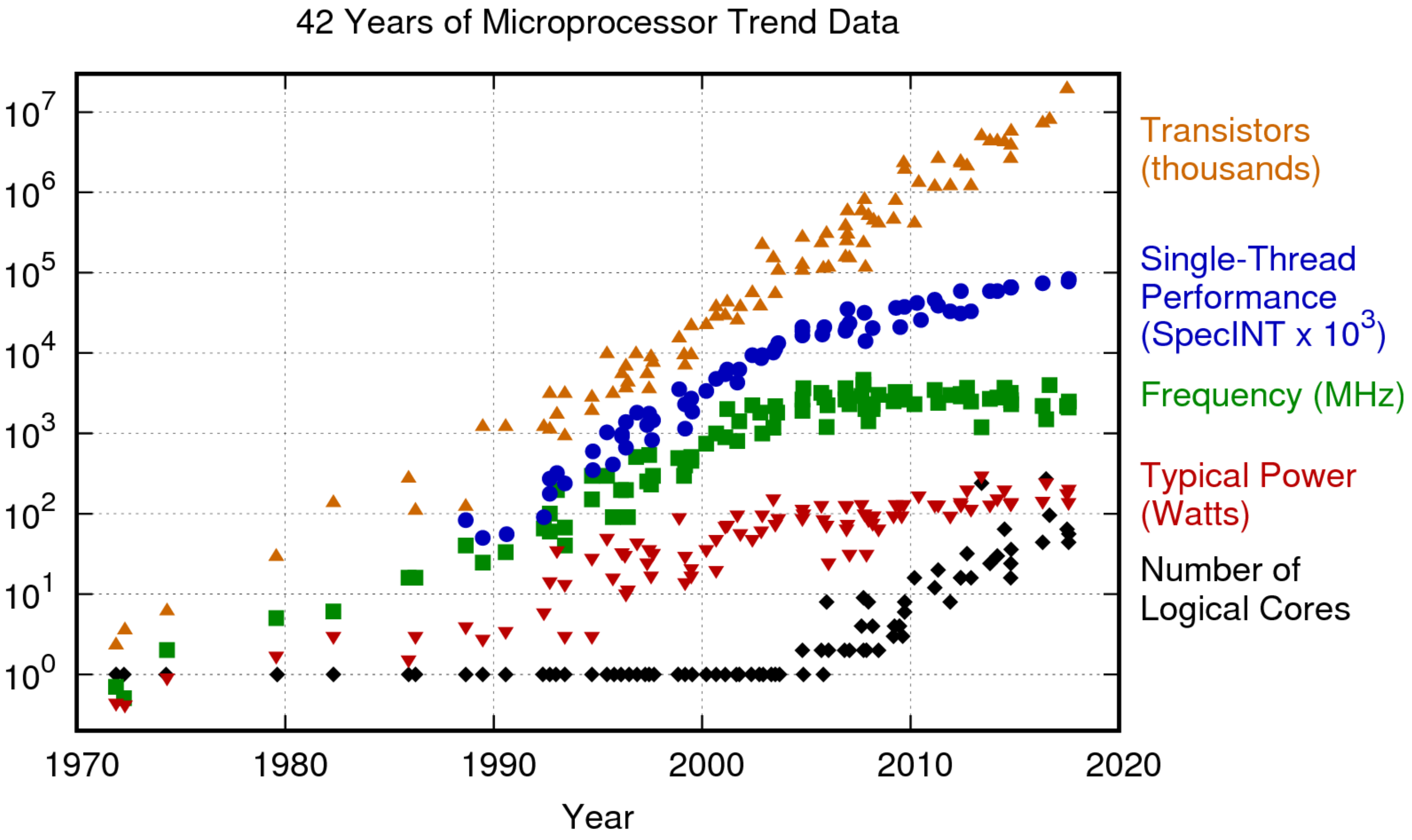
EE3301/EE5184
Introduction to VLSI Design

Instructor: Naresh Emani

Credits: 2 - 1st Aug to 19th Oct

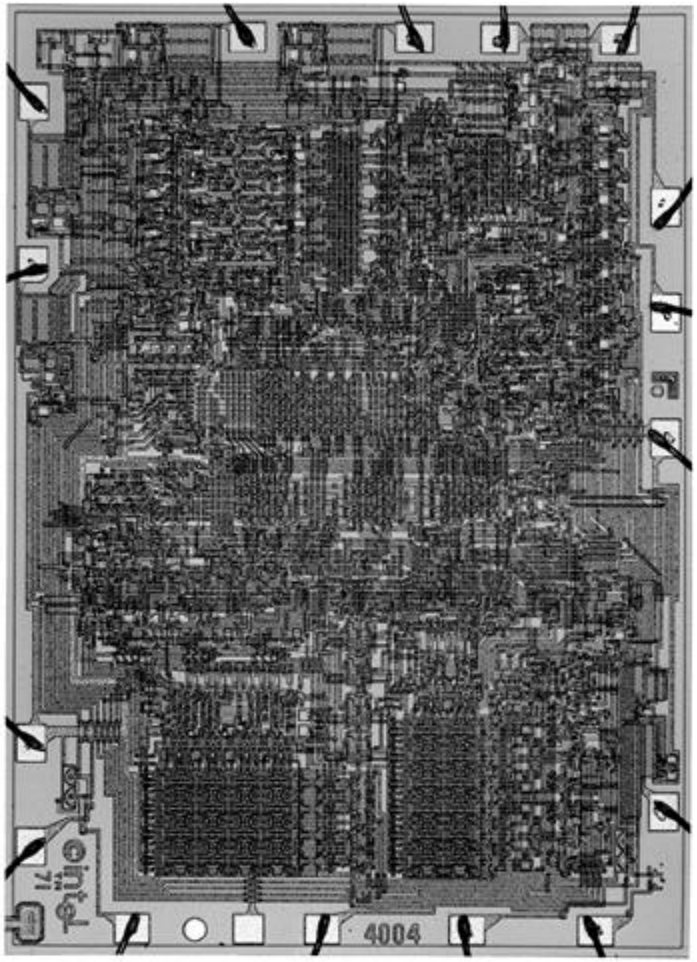
Class: Mon 12-1 PM, Tue 9-10 AM and Fri 10 – 11 AM

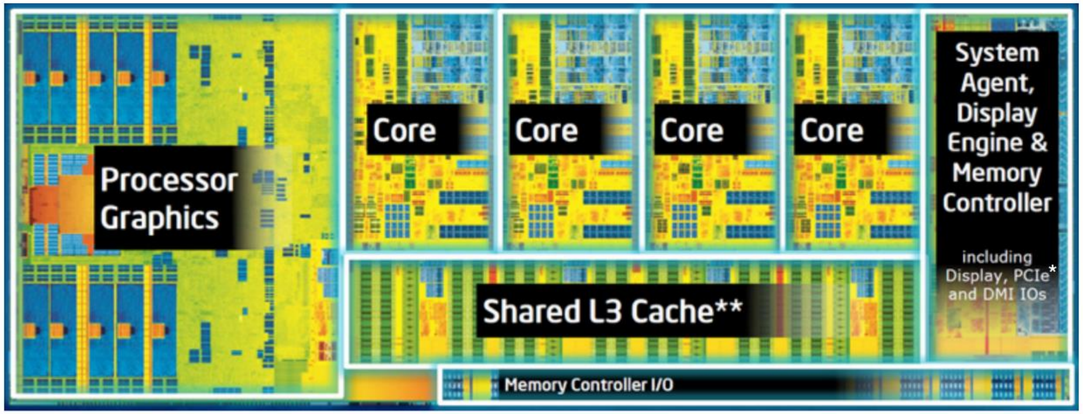
Microprocessor Evolution



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

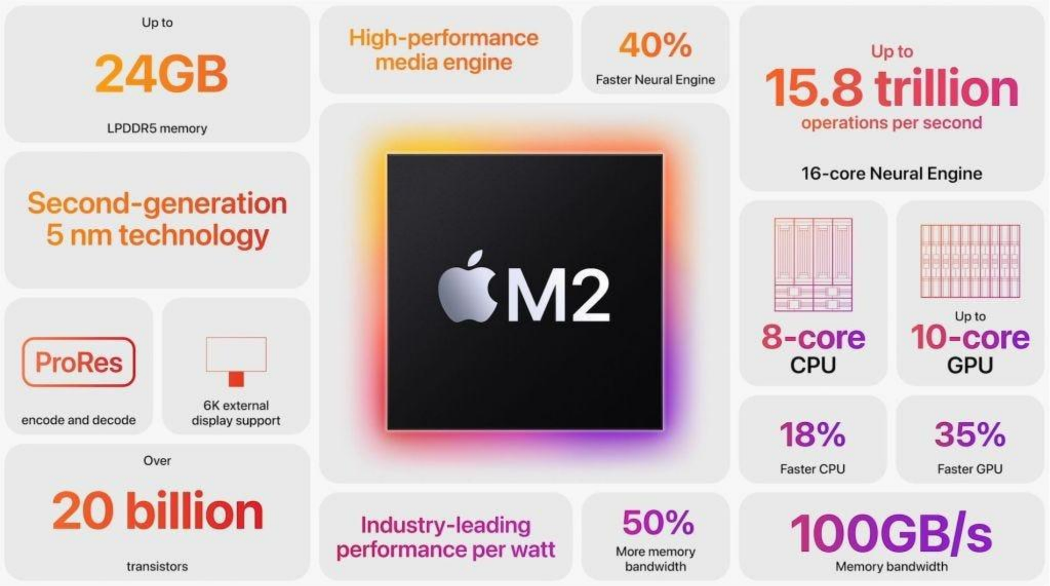
Evolution of microprocessor architecture




Processor Graphics
Core
Core
Core
Core
System Agent, Display Engine & Memory Controller
including Display, PCIe* and DMI IOs
Shared L3 Cache**
Memory Controller I/O

- ❑ Intel 4004 Processor (1971) - 2300 Transistors, 0.6 MHz operation
- ❑ Intel Core i7 Quad Core (4th Gen) - 22 nm (2013) - 1.4 Billion transistors, 177 mm2
- ❑ Cerebras WSE - 2.6 trillion transistors, 46225 mm^2

Apple M2



Up to **24GB** LPDDR5 memory

High-performance media engine

40% Faster Neural Engine

Up to **15.8 trillion** operations per second

16-core Neural Engine

Second-generation 5 nm technology

ProRes encode and decode

6K external display support

M2

8-core CPU

Up to **10-core** GPU

18% Faster CPU

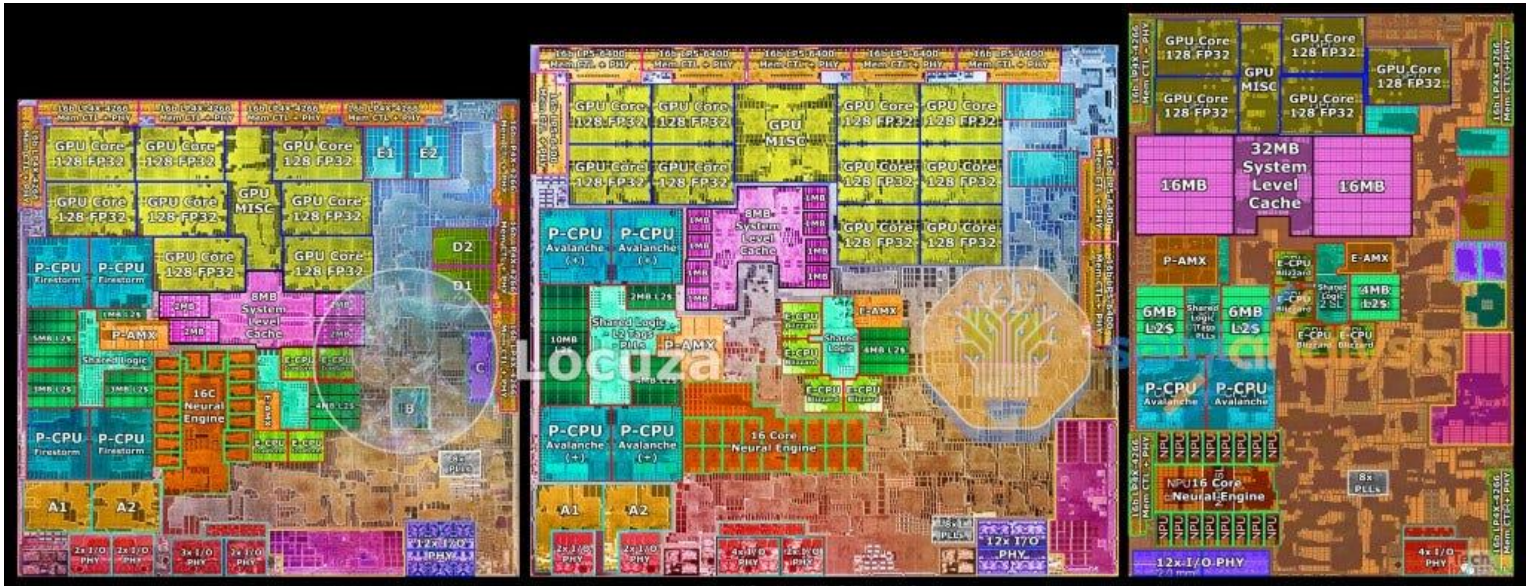
35% Faster GPU

Over **20 billion** transistors

Industry-leading performance per watt

50% More memory bandwidth

100GB/s Memory bandwidth

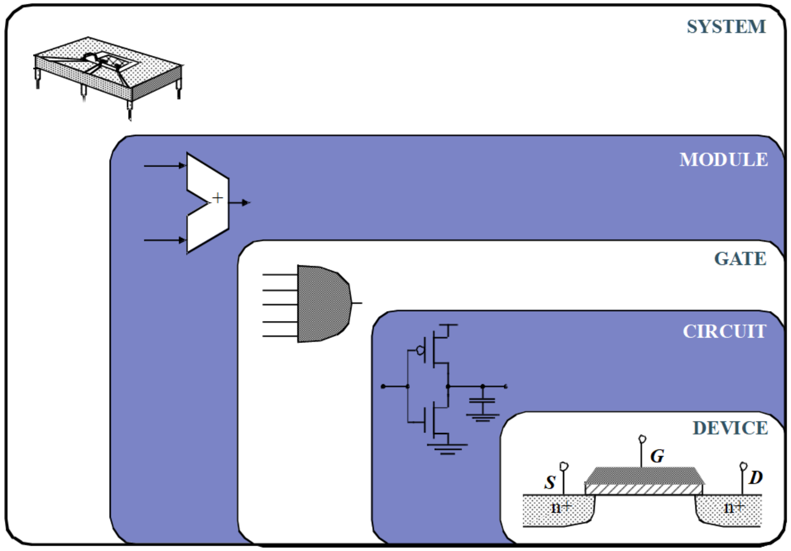


M1 = 118.91 mm²

M2 = 155.25 mm²
(Apple presented scaling ~ 141.70 mm²)

A15 = 107.68 mm²

Design abstraction



- ❑ Architecture – Instruction set, registers, size of memory, etc

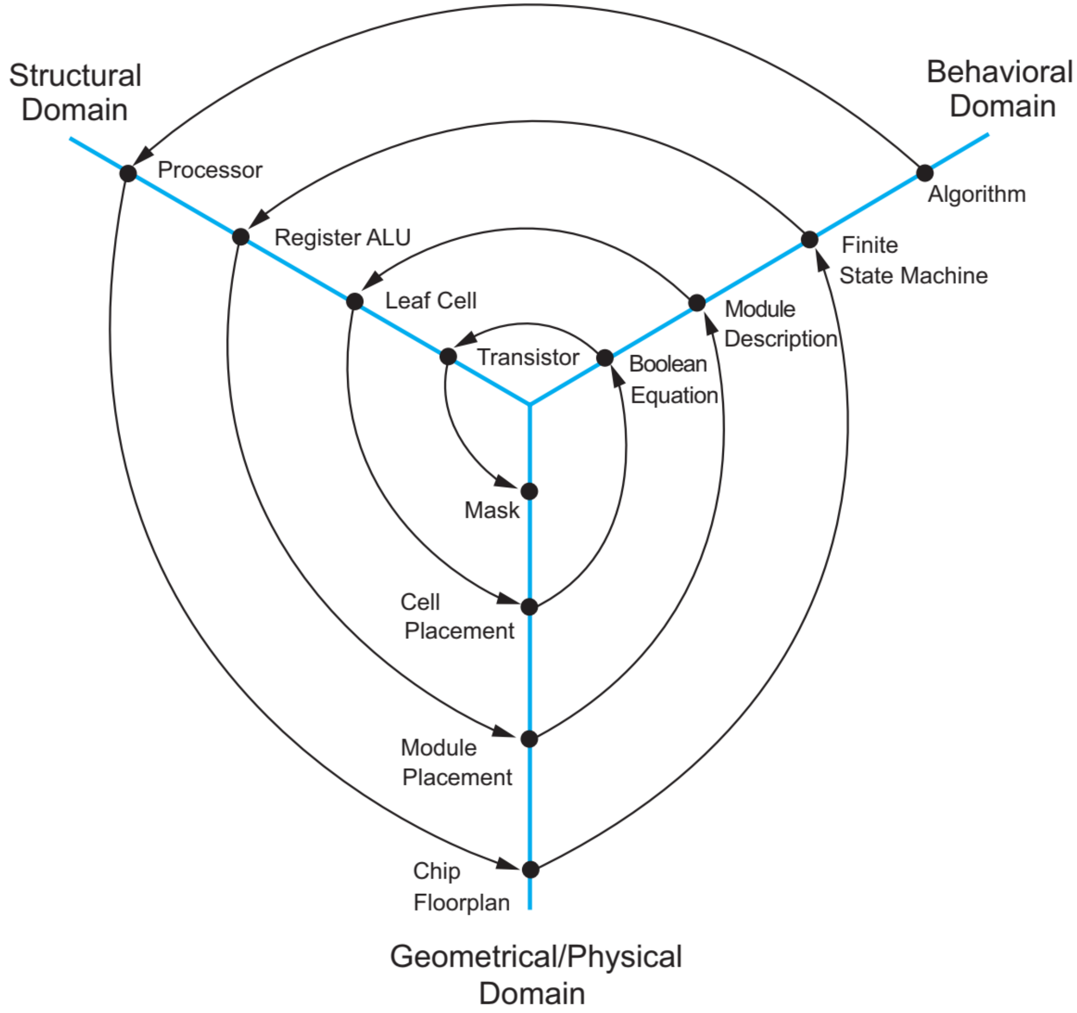
- ❑ Microarchitecture – How is memory partitioned, size of registers, et

- ❑ Logic design – How are logical expressions implemented, *e.g*. ripple carry or carry look ahead

- ❑ Circuit design – Circuit Static or dynamic circuits?

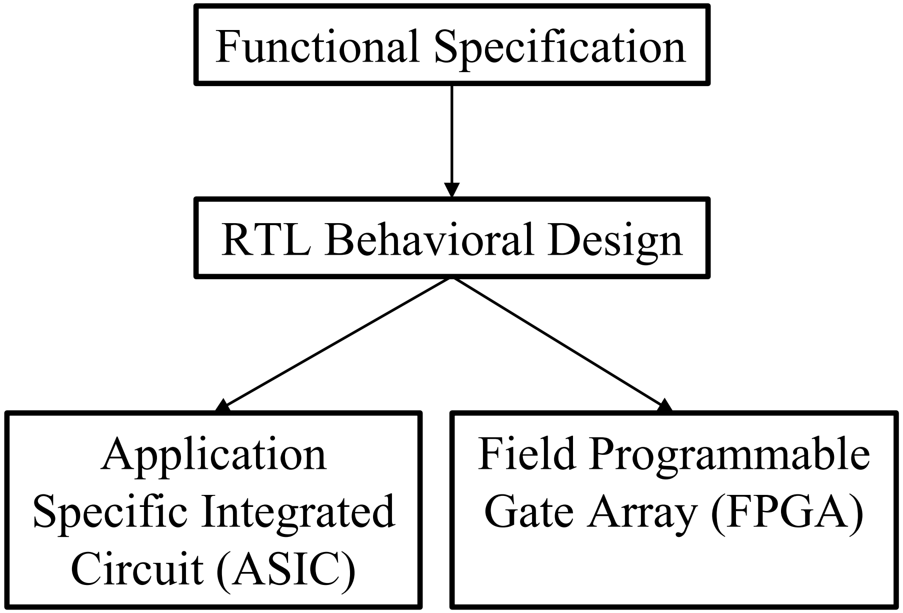
- ❑ Physical design –Size of transistors, parasitic RCs per unit area?

Another design abstraction approach



Implementation strategies

Full custom	All circuits/transistor layouts optimized for a specific application
Standard cell (ASIC)	Arrays of small function blocks (gates, FFs) are automatically placed and routed
Gate array (structured ASIC)	Partially fabricated wafers customized with metal layers or vias
FPGAs	Prefabricated chips customized with loadable latches
Microprocessor	Instruction set interpreter customized through software
Domain specific processor	Special instruction set interpreters (e.g. DSP, GPU, etc)


Functional Specification
RTL Behavioral Design
Application Specific Integrated Circuit (ASIC)
Field Programmable Gate Array (FPGA)

VLSI design flow – FPGA vs ASIC

FPGA’s are useful when

- Design constraints are not stringent
- Speed and power requirements are within the reach
- Quick Prototyping

ASIC’s are essential when

- Design of high-speed devices e.g. microprocessors
- Low power ICs
- Design of FPGA’s themselves

How do you implement an adder?

```
module full_adder (inputs)
        input …
        assign ….
end module
```

What is missing?

- □ How is the adder connected to external environment
- □ How many transistors is it driving
- □ What are the parasitic resistance, capacitance and inductance
- □ Global signals – clock synchronization, power distribution

Background Material

Topics	Online Links
Introduction	https://youtu.be/iEJ8aMpClxE
Implementation strategies, SPICE	https://youtu.be/MQYDx9H5uLw
Design metrics	https://youtu.be/iPXURPw3KoU
Design metrics (contd..) and device models	https://youtu.be/IRumibg5cmk
Review of device physics	https://youtu.be/_ta-d2dPSNE
Short channel effects	https://youtu.be/1VEP9XHZ1VM
Process corners, propagation delay	https://youtu.be/4dA5xYZeX5Q
MOSFET Capacitances	https://youtu.be/-f1L5sldzZ8
MOSFET Capacitances (contd…)	https://youtu.be/3fje8CpRh-0
Introduction to Semiconductor Devices NPTEL Course (Weeks 7-10 deal with MOS devices)	https://nptel.ac.in/courses/108106181

EE3301/EE5184: Introduction to VLSI Design

Overview

This course will provide an introduction to CMOS VLSI circuits that lie at the heart of modern microprocessors. By the end of this course you should have acquired the necessary skills to design a simple arithmetic and logic unit by performing circuit simulation, physical layout and post layout verification.

Outline of topics

- Overview; Design abstraction levels; MOSFET device SPICE model; transistor scaling, design rules; timing and power considerations in design
- CMOS Inverter - Static and dynamic response, threshold, noise margin; propagation delay, parasitic capacitance estimation; static and dynamic power consumption; SPICE analysis; inverter layout
- Combinatorial logic design in CMOS; Ratioed logic; Pass transistor logic; Introduction to Dynamic CMOS design
- Introduction to sequential logic circuits; latches and registers - static and dynamic response; Pipelining;
- Designing arithmetic building blocks; Datapaths in digital processor architecture; Adder; Multiplier; Shifter; Other arithmetic operations; Power-speed trade-off's in design

Evaluation

The final grade will be calculated **tentatively** with the following weights:

- Surprise quizzes during class - 20%
- Mid-term exam - 30%
- Laboratory assignments - 20%
- Final exam - 30%

Primary References

- Rabaey, Chandrakasan and Nikolic, *Digital Integrated Circuits - A Design Perspective*, 2nd Edition (2002)
- Weste and Harris, *CMOS VLSI Design - A Circuits and System Perspective*, 4th Edition (2011)

Course Policies

- You may discuss assignments/homework/projects with your classmates. However, final submission must be your own. **It is your responsibility to understand what constitutes plagiarism/academic dishonesty**. Please refer to https://cse.iith.ac.in/?q=Academics/Plagiarism%20Policy for an introduction. **Academic dishonesty will be reported to Dean Academics, and a FR grade will be given for the course.**

- I expect you to attend classes regularly and participate in class discussions. If you miss a class it is your responsibility to catch up with the material covered. If you miss an exam, the grade for that will be zero, unless a valid reason is given ahead of time. If you must miss a class for an unavoidable reason, I will appreciate if you send me an email ahead of time.

- Assignments and homeworks deadlines will be strictly enforced. If there is a delay a penalty of 20% per day upto a maximum of 5 days will be enforced. If you have a valid excuse you must email me well ahead of deadline.

Implementation of a simple ALU
Last Updated: 30th July 2022

Overview

An Arithmetic Logic Unit(ALU) is the heart of any digital processor. The first stage of this course involves RTL description of an ALU using HDL. In parallel, in EE5184 you will learn about design, layout and LVS of simple logic gates. In the second stage of this course you will be required to use the library generated by you and verify the operation of a 4-bit ALU. You will also optimize your design for delay and area. You are encouraged to form teams of 2-3 students to design, implement and verify the operation of a four bit ALU.

Specifications

The ALU that you will design has to perform the following operations:

OpCode	Operation	Description
000	ADD A,B	Add 4 bit inputs A and B
001	SUB A,B	Subtract 4 bit inputs A and B
010	CMP A,B	Check if 4 bit inputs are equal
011	SHL A	Shift 4 bit input A left by 1 bit

Design Constraints

(a) Use TSMC 180 nm library

(b) You are free to choose any supply voltage and logic swing of upto 1.8 V

(c) All input signals have a rise and fall time of 50ps

(d) The propagation delay for static CMOS design is defined as the time interval between the 50% transition point of the inputs and the 50% point of the worst case output signal. Make sure you pick the worst-case condition.

(e) The area is defined as the smallest rectangular box that can be drawn around the design.

Evaluation Process (Tentative)

(a) Correctness of functionality - 20 points

(b) Implementation of design - 20 points

(c) Testing and verification of design - 20 points

(d) A project report summarizing your project. Your report must be concise and present your major results in a professional manner - 20 points

(e) 20 bonus points will be awarded for innovative ideas which significantly go beyond the basic requirements listed above. This will be awarded only upon satisfactory completion of (a) - (d) above

(f) There will be a team grade which is uniform for all the team members and has 80% weightage. There will be 20 % weightage to individual viva which can be on any topic related to the project.

(g) You are required to finalize your teams by 15th Aug 2022, and inform the instructor of the team composition.

(h) Deadline for the project: Monday 28th Nov 2022, 9 AM. Evaluation and viva will be held tentatively on Monday 28th Nov 2022.