Digital Integrated Circuits

(83-313)

Lecture 4:

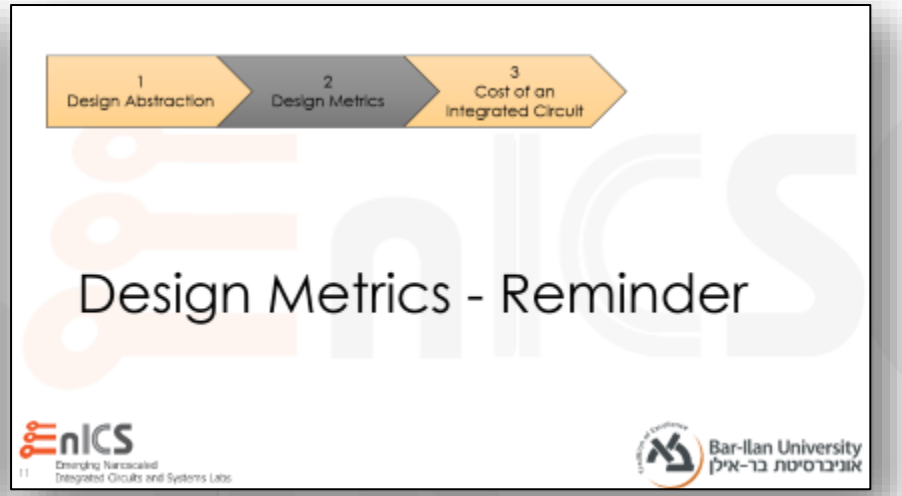
Design Metrics

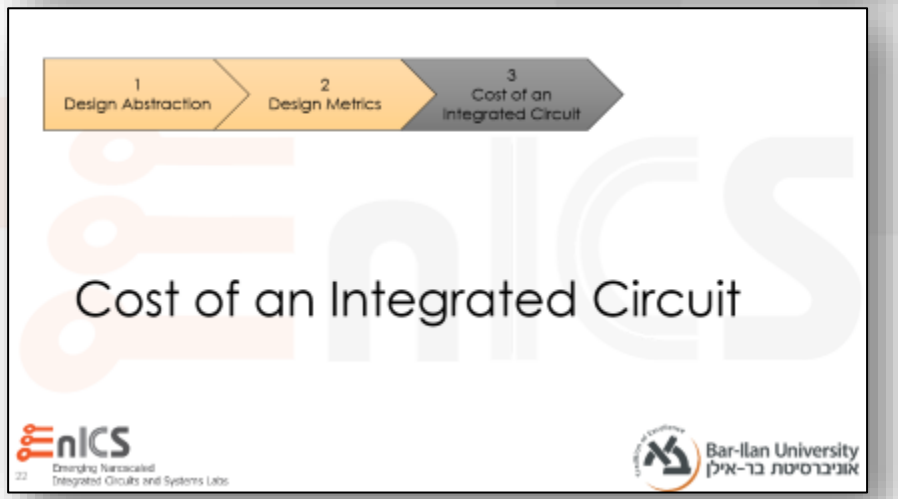
28 April 2020

Emerging Nanoscaled
Integrated Circuits and Systems Labs

Bar-Ilan University
אוניברסיטת בר-אילן

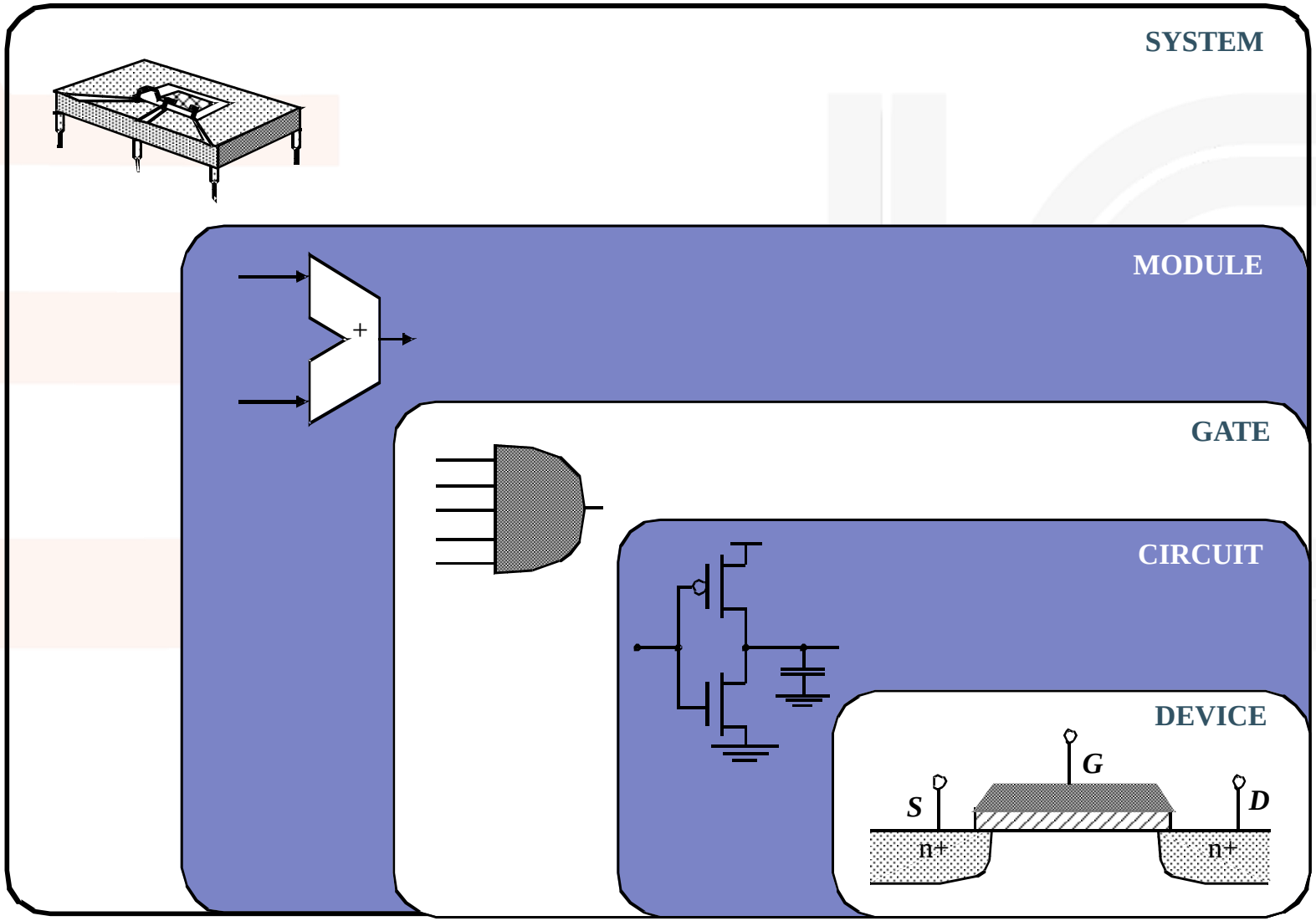
Lecture Outline





Design Abstraction

Design Abstraction Levels



Device Level Abstraction

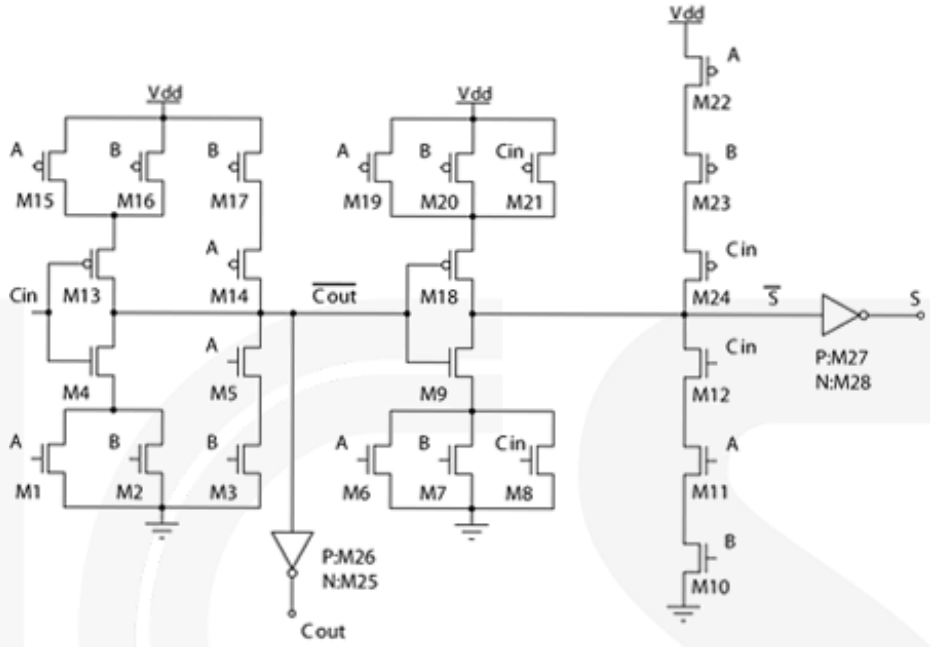
- **Fabrication Plants or *Foundries* supply a *Process Design Kit* (PDK).**
- **The PDK includes:**
  - Devices
    (Transistors, Resistors, Capacitors, Diodes)
  - Layers
  - Rules
- **Various "Flavors" of PDKs are available, e.g.:**
  - General Purpose/High Speed/Low Power
  - RF/Image Sensor
  - Flash/DRAM
  - Various number of Metal Interconnect Layers

www.extremetech.com

Device Level

Circuit Level Abstraction

- **Using the devices supplied in the PDK, *Schematics* are drawn.**
- **Simulators, such as *SPICE*, are used to test and optimize the circuits.**
- **Various parameters (called *CDF parameters*) can be modified to optimize the schematic, e.g.:**
  - Length and Width of transistors.
  - Number of Fingers.
  - Capacitances and Resistances.
- **Circuits are drawn with a *Layout Editor* and *parasitics* are extracted for accurate simulation.**



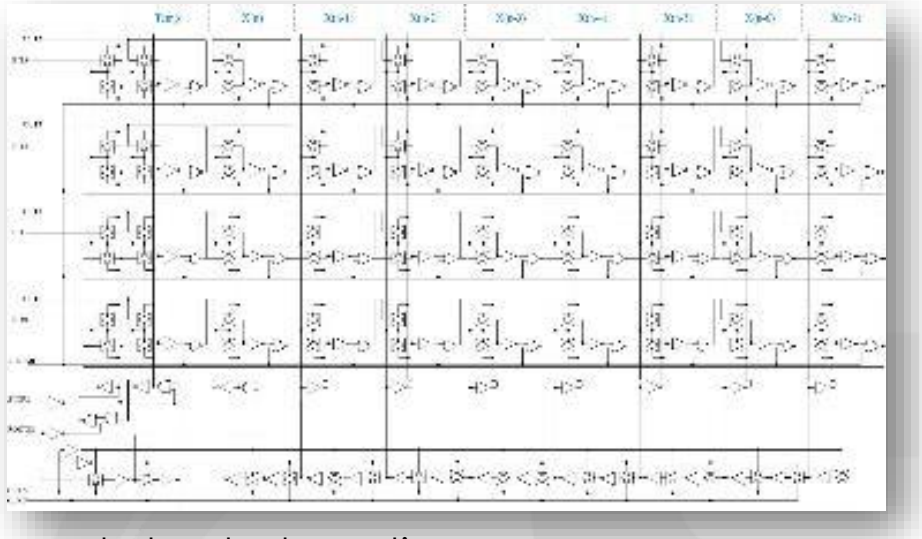
Circuit Level

↑

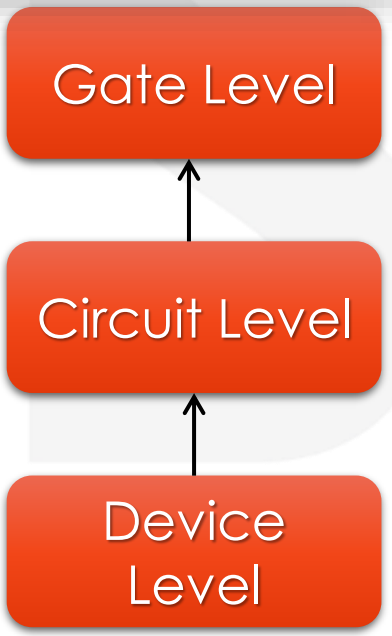
Device Level

Gate Level Abstraction

- **Drawn circuits are abstracted into a black box for use as gates.**
- **These gates are defined by easy to use characteristics, such as:**
  - Boolean Functionality.
  - Interface (i.e. *pins* or *ports*).
  - Delay and power consumption.
  - Input and output capacitance.
  - Size and geometry.
- **Once a gate is abstracted, it can be used by higher level tools, such as *HDLs*.**

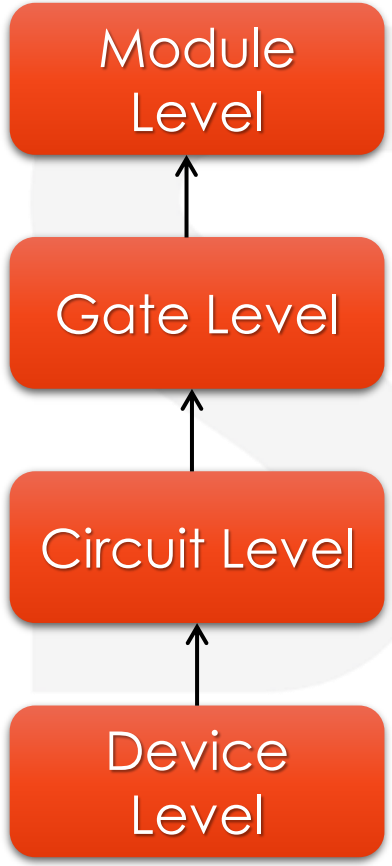


8-bit adder gate-level schematic.
http://pneuhiver.info



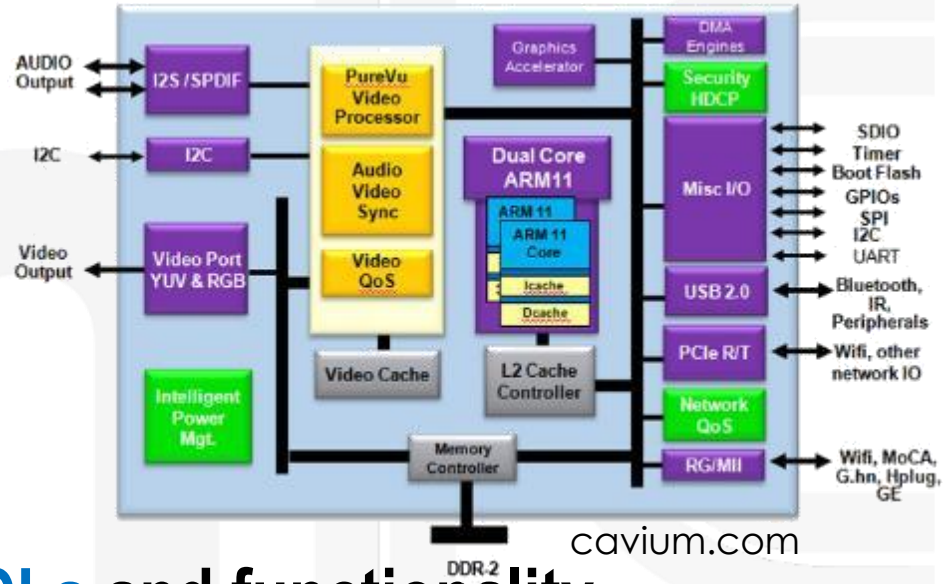
Module Level Abstraction

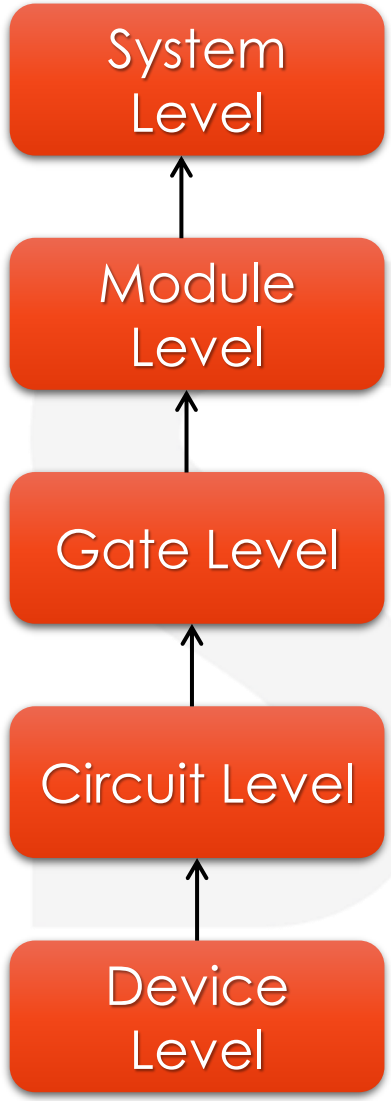
- Gates and other low level circuits are connected together into modules (adders, memories, etc.).
- These are tested for functionality and are abstracted for system integration.
- Analog modules are abstracted from circuit level.
- Digital modules and full systems are defined with HDLs, *instantiating* gates and modules.
- EDA tools are used to verify functionality at all depths of hierarchy.



System Level Abstraction

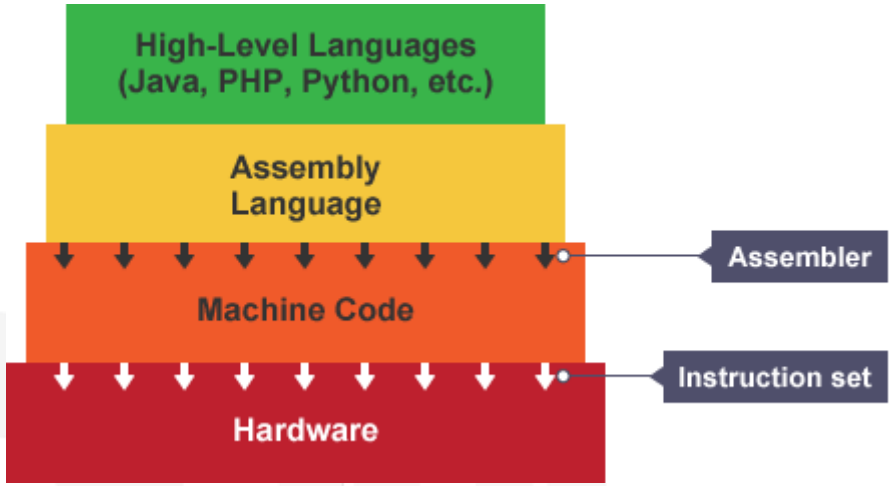
- **Architectural design** defines high level abstractions to build a system.
- This abstraction level defines:
  - Registers
  - Instruction Sets
  - Control Blocks
  - Buses
  - etc.
- **Systems** are implemented with **HDLs** and functionality is verified with **logical verification**.
- **System** are defined to comply with ***standards*** and implement ***protocols***.



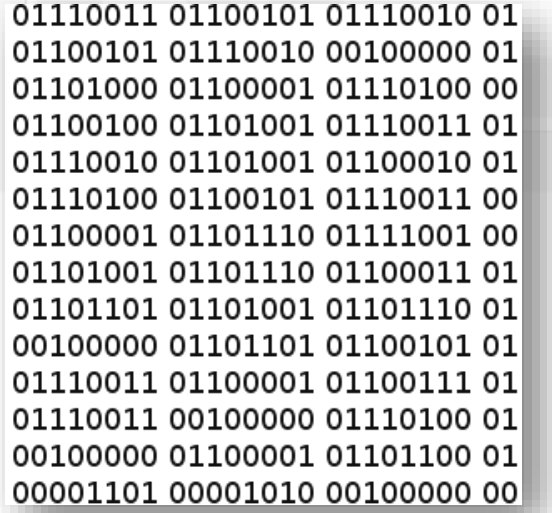


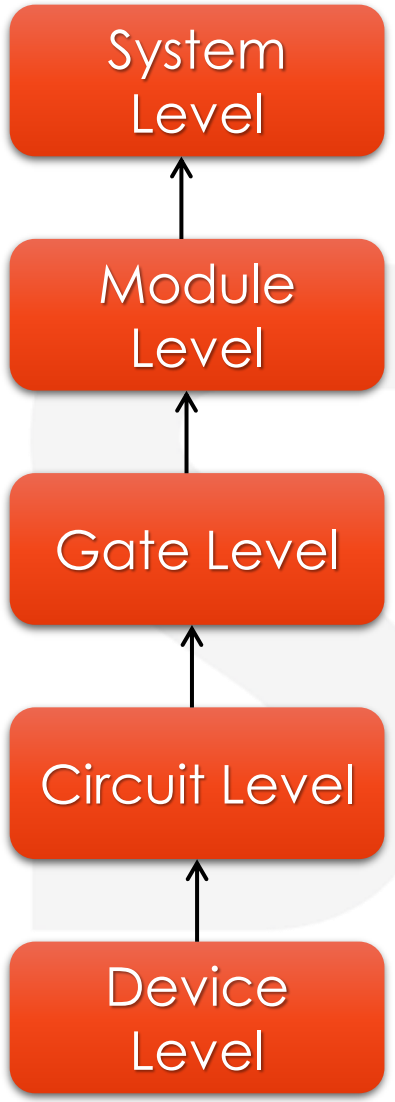
Higher Level Abstraction

- **Programmers write code in a *high level programming language* (C, Java, Perl, etc.)**
- **A *Compiler* translates the code into an *Assembly* language.**
  - Compilers try to optimize the instructions according to the actual architecture they are compiling to.
- **An *Assembler* translates the Assembly into *Machine Language* (i.e. 0's and 1's).**



bbc.co.uk





Design Metrics - Reminder

There's No Free Lunch!

- Remember, the job of a VLSI designer is to understand, design and optimize the *trade-offs* between:

Performance | **Cost**

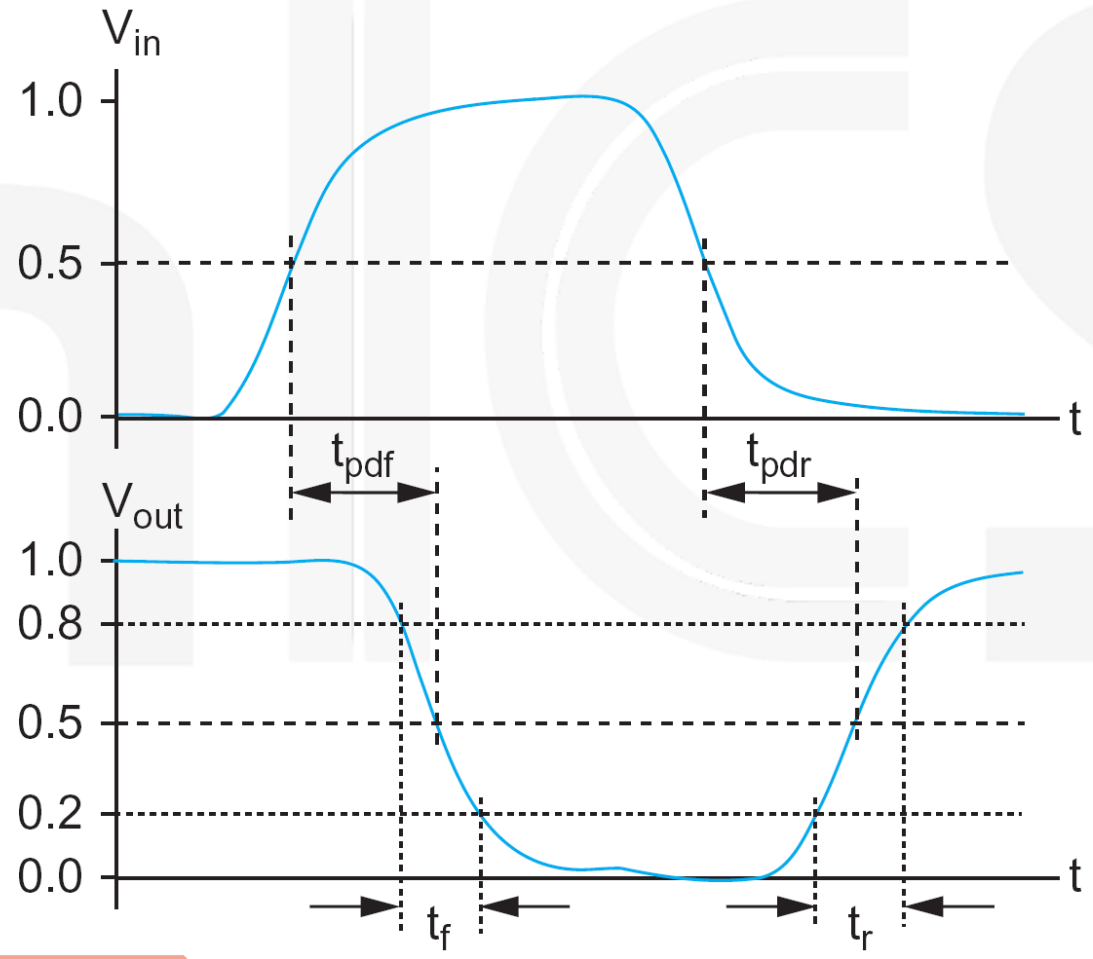
Reliability | **Power**

- In this section, we will go over the main design metrics for evaluating these design aspects.
- You can rarely improve all these factors simultaneously.

Performance Metrics

Performance

- **When measuring performance, we usually use:**
  - Propagation delay t_{pd}
  - Rise/Fall time t_r/t_f

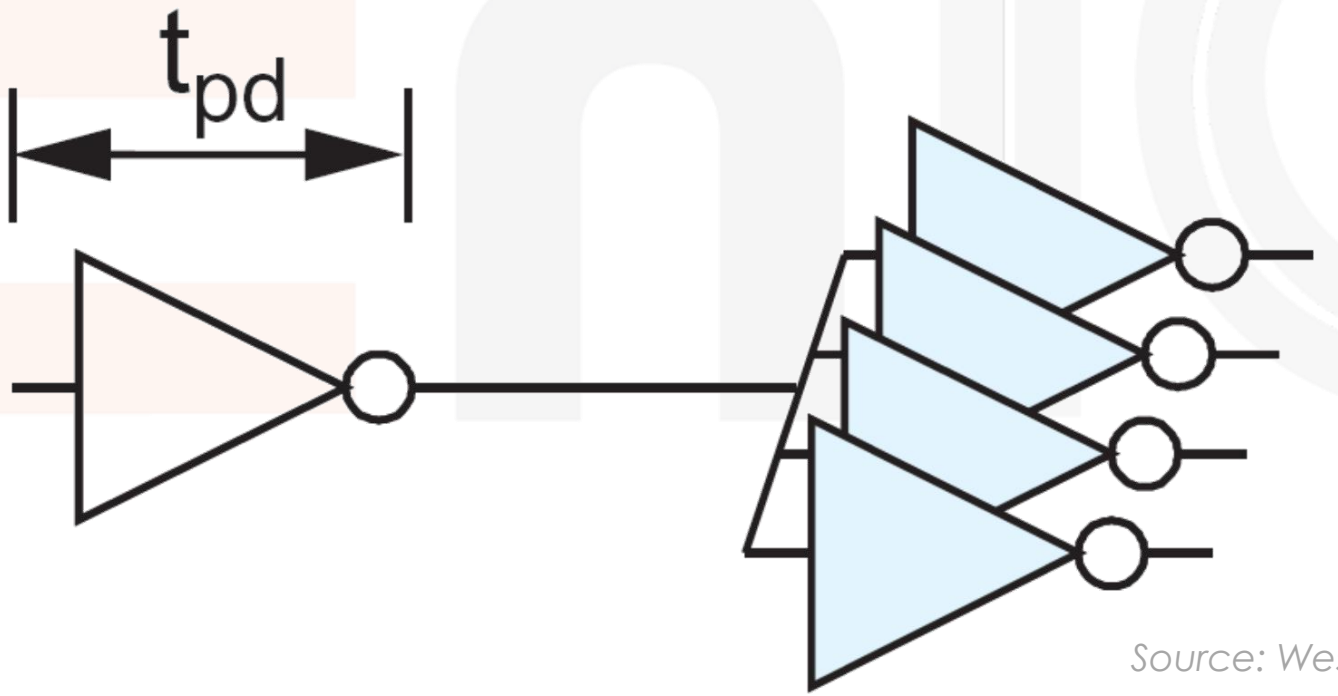


Source: Weste, Harris 4th ed.

Performance Metrics

Performance

- Based on our derivation of optimal load driving and *Logical Effort*, the performance of a gate is often given in terms of "Fan out of 4" (FO4).
- How would we measure FO4 in a simulation?



Source: Weste, Harris 4th ed.

How to measure FO4 delay

Performance

Power Metrics

Power

- **When discussing *power*, we generally mean the *average power* consumption of a component.**
  - Other times we may discuss *peak power*, but this is less common.
- **The power consumption is usually decomposed into three factors:**
  - **Dynamic Power**, which is essentially the *energy* consumed during a *switching* operation.
  - **Static Power**, which is usually caused by *leakage* currents in steady states.
  - **Short Circuit Power,** which is the component of dynamic power that is due to direct currents between the supply rails.

$$P_{av} = \frac{1}{T}\int_0^T p(t)\,dt = \frac{V_{supply}}{T}\int_0^T i_{supply}(t)\,dt$$

$$P_{peak} = i_{peak} V_{supply} = \max\left[p(t)\right]$$

Dynamic Power

Power

- The energy required to charge a capacitive node to V_{DD} is approximately:

$$E_{LH} = C_{switch} V_{DD}^2$$

- Therefore, if a circuit is operated at frequency f, the dynamic power can be written as:

$$P_{dynamic} = f \cdot C_{switch} V_{DD}^2$$

- However, not all of the circuit capacitance switches during every cycle. The percentage of capacitance that switches will be called the activity factor α, resulting in:

$$P_{dynamic} = \alpha \cdot f \cdot C_{total} V_{DD}^2$$

Static and Short Circuit Power

Power

- **Both static and short circuit power are caused by current flowing between the supply rails.**
  - ***Static power*** is due to *leakage* or *static currents* in the steady state of a circuit.
  - ***Short circuit power*** is due to low resistive paths between the supplies during a switching event.

$$P_{static} = I_{static} V_{supply}$$

- **Both types of consumption are unwanted and should be minimized.**

Reliability Metrics

Reliability

- **Reliability** is the ability of the circuit to provide correct results during all operations throughout the lifetime of the system.
- The only measure of reliability we have learned so far is the digital noise margin metric:

$$NM_H = V_{OH\min} - V_{IH}$$

$$NM_L = V_{IL} - V_{OL\max}$$

$$NM = \min\left(NM_L, NM_H\right)$$

- We will discuss further reliability aspects later on in the course.

Cost Metrics

Cost

- The final, and often, most important, design metric is **cost**.
- The only **cost** feature we have previously considered is **area/size** of a transistor/gate. But we never actually quantified this in terms of **prices**.
- This is an important enough issue to deserve an entire section on its own.

Source: www.betterthanpants.com

The Computer Hall of Fame

Source: wikipedia

- **The computer that brought computers home**

The IBM PC

- *"IBM bringing out a personal computer would be like teaching an elephant to tap dance."*

  Introduced in 1981, sold for $1,565
- Ran on an Intel 8088 processor – developed in Haifa!

  4.77 MHz, 16-256 kB
- Along with its release, a new operating system, called MS-DOS, introduced Microsoft to the world.

Source: computerhistory.org

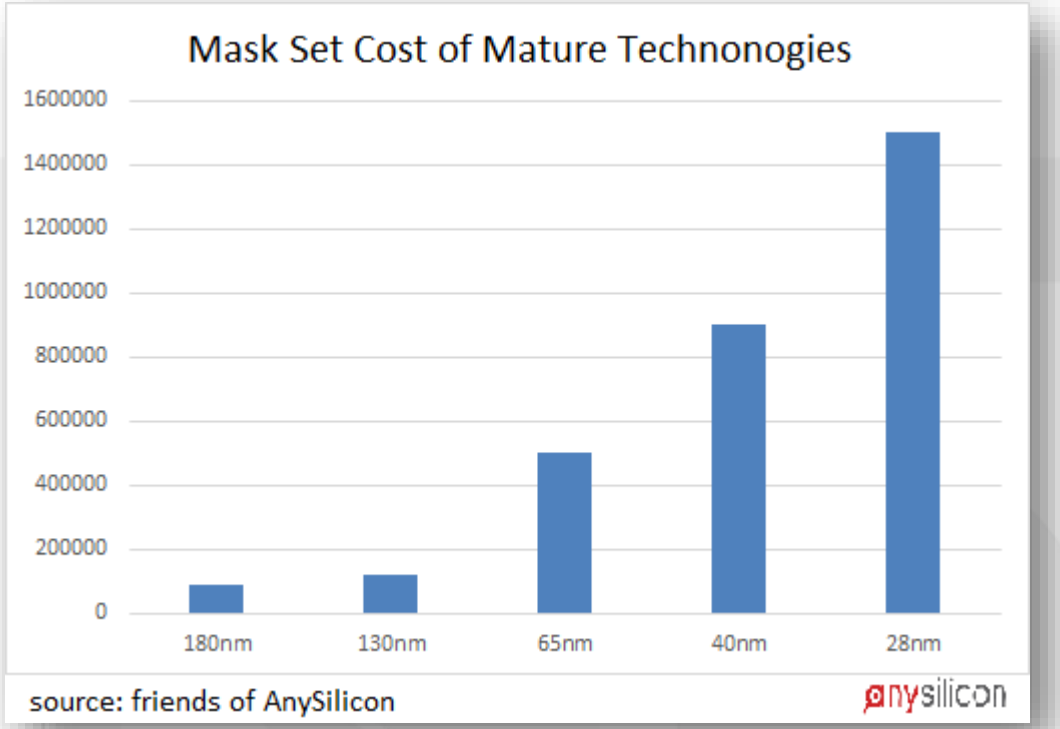
Cost of an Integrated Circuit

How much does it cost?

- In the end, it all comes down to **cost**.
- **Cost** can come in many ways, but to get a basic understanding of this concept, let us develop a simple **cost model** for an IC chip.

Cost of Integrated Circuits

- **NRE (non-recurrent engineering) costs**
  - One-time cost factor
  - Design time and effort
  - Mask generation
- **Recurrent costs**
  - Silicon processing, packaging, test
  - Proportional to volume
  - Proportional to chip area



Source: anysilicon

Total Cost

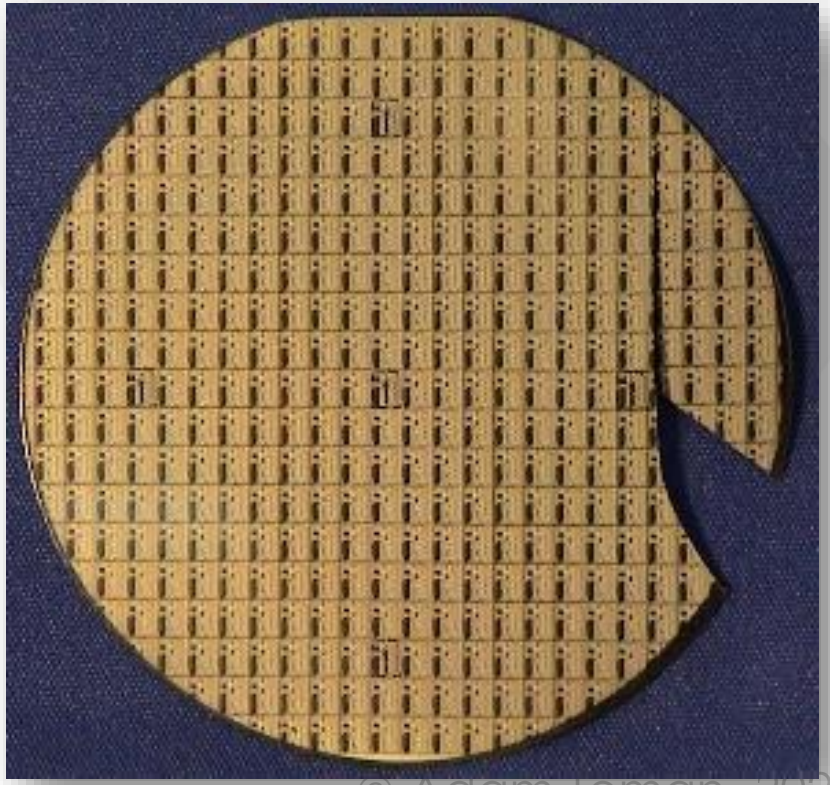
- **Cost per IC**

$$\text{cost per IC} = \text{variable cost per IC} + \left(\frac{\text{fixed cost}}{\text{volume}}\right)$$

- **Variable cost**

$$\text{variable cost} = \frac{\text{die cost} + \text{package cost} + \text{test cost}}{\text{final test yield}}$$

$$Yield = \frac{\text{No. of good chips}}{\text{Total number of chips}} \times 100\%$$



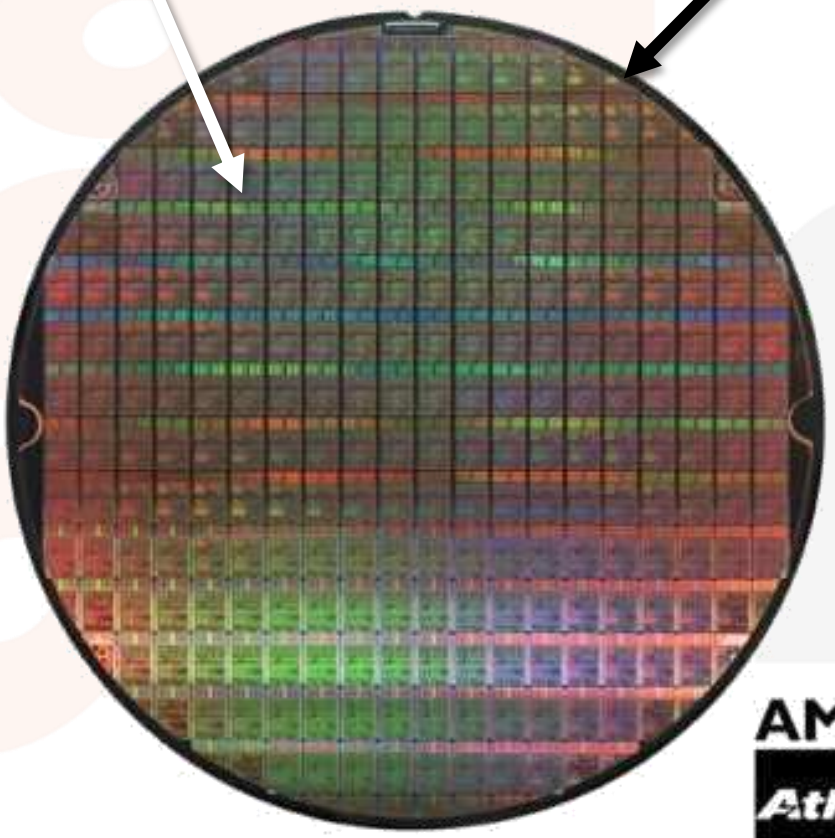
Die Cost

$$\text{cost per IC} = \text{variable cost per IC} + \left(\frac{\text{fixed cost}}{\text{volume}}\right)$$

$$\text{variable cost} = \frac{\text{die cost} + \text{package cost} + \text{test cost}}{\text{final test yield}}$$

Single Die

Wafer

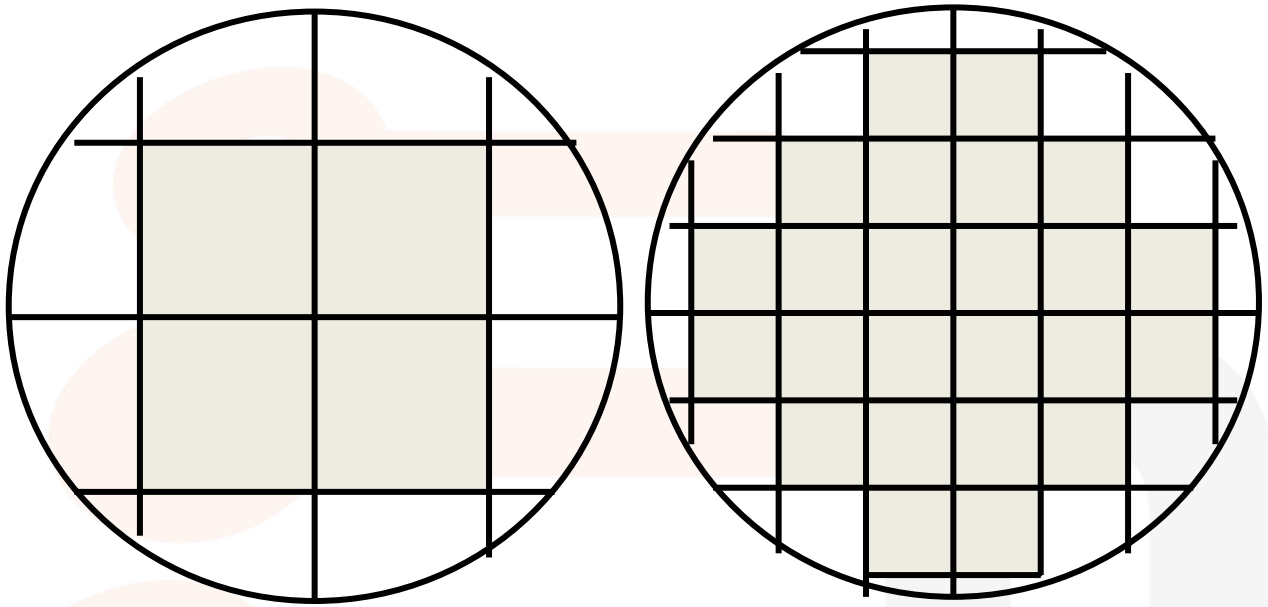


$$\text{cost of die} = \frac{\text{cost of wafer}}{\text{dies per wafer} \times \text{die yield}}$$

Testing on the wafer.
Before packaging!

From http://www.amd.com

Edge Losses



$$\text{cost per IC} = \text{variable cost per IC} + \left(\frac{\text{fixed cost}}{\text{volume}}\right)$$

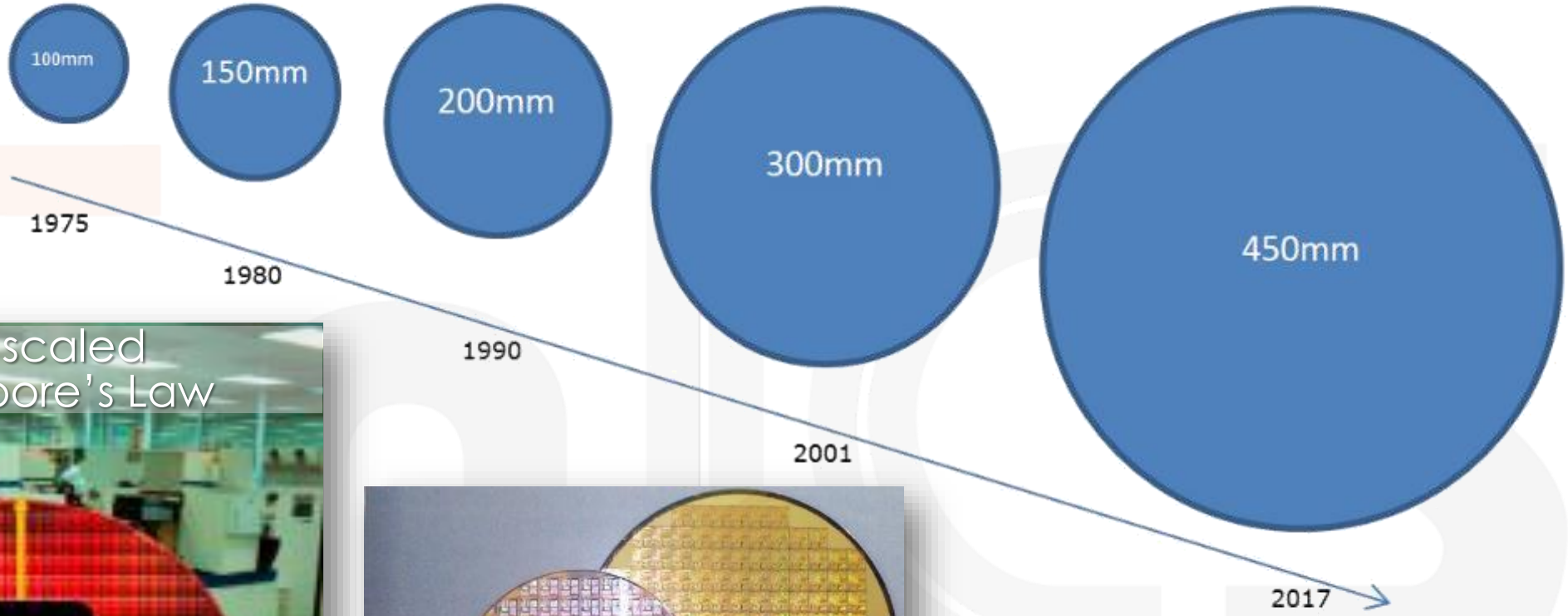
$$\text{variable cost} = \frac{\text{die cost} + \text{package cost} + \text{test cost}}{\text{final test yield}}$$

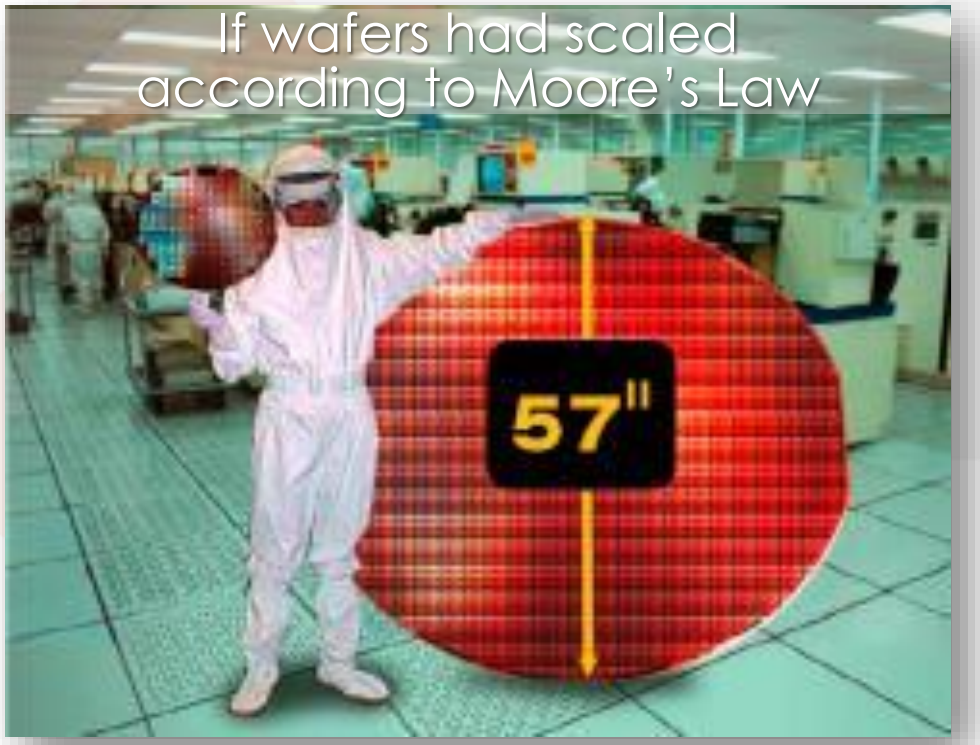
$$\text{cost of die} = \frac{\text{cost of wafer}}{\text{dies per wafer} \times \text{die yield}}$$

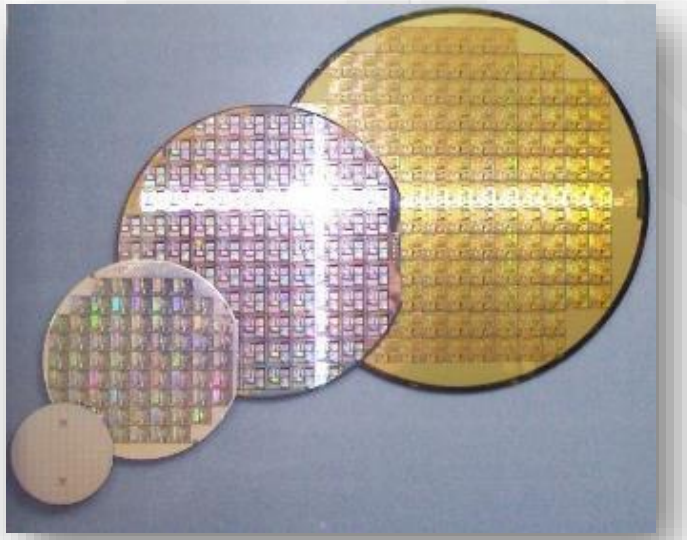
$$\text{Dies per wafer} = \frac{\pi \times (\text{wafer diameter}/2)^2}{\text{die area}} - \frac{\pi \times \text{wafer diameter}}{\sqrt{2 \times \text{die area}}}$$

Wafer Size

Source: anysilicon.com







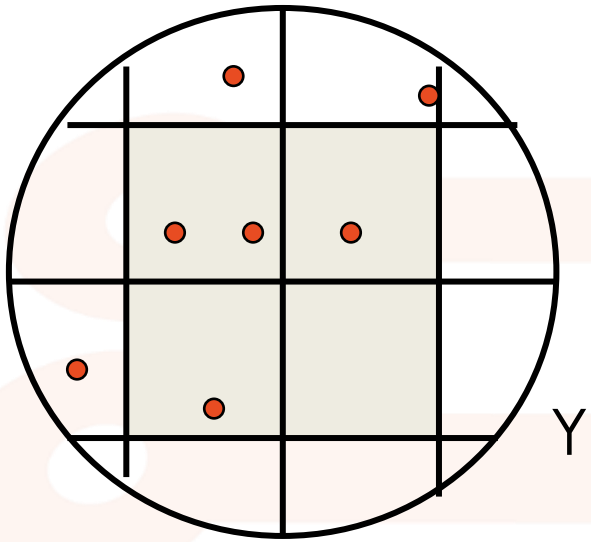
Source: wikipedia

Defects

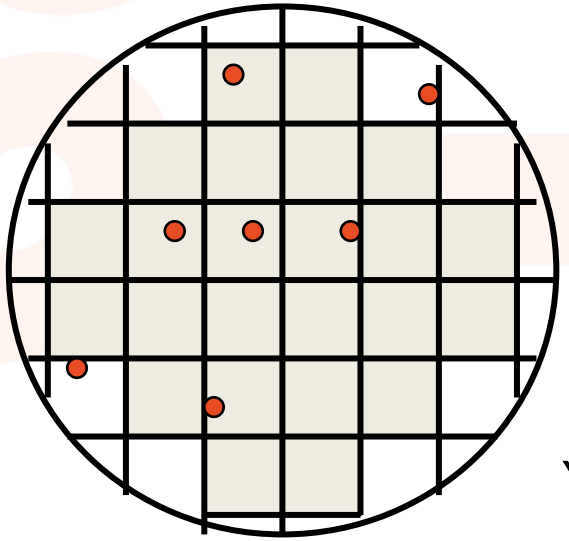
$$\text{cost per IC} = \text{variable cost per IC} + \left(\frac{\text{fixed cost}}{\text{volume}}\right)$$

$$\text{variable cost} = \frac{\text{die cost} + \text{package cost} + \text{test cost}}{\text{final test yield}}$$

$$\text{cost of die} = \frac{\text{cost of wafer}}{\text{dies per wafer} \times \text{die yield}}$$



Yield = ¼ = 0.25



Yield = 19/24 = 0.79

$$\text{die cost} = f(\text{die area}) \propto \text{die area}^4$$

Total Cost - summary

$$\text{cost per IC} = \text{variable cost per IC} + \left(\frac{\text{fixed cost}}{\text{volume}}\right)$$

$$\text{variable cost} = \frac{\text{die cost} + \text{test cost} + \text{package cost}}{\text{final test yield}}$$

After packaging

$$Yield = \%good\ chips$$

$$\text{die cost} \propto \text{die area}^4$$

$$\text{die cost} = \frac{\text{cost of wafer}}{\text{dies per wafer} \times \text{die yield}}$$

Before packaging (on wafer)

$$\text{Dies per wafer} = \frac{\pi \times (\text{wafer diameter}/2)^2}{\text{die area}} - \frac{\pi \times \text{wafer diameter}}{\sqrt{2 \times \text{die area}}}$$

Some Examples (1994)

Chip	Metal layers	Line width	Wafer cost	Def./ cm^2	Area mm^2	Dies/ wafer	Yield	Die cost
386DX	2	0.90	$900	1.0	43	360	71%	$4
486 DX2	3	0.80	$1200	1.0	81	181	54%	$12
Power PC 601	4	0.80	$1700	1.3	121	115	28%	$53
HP PA 7100	3	0.80	$1300	1.0	196	66	27%	$73
DEC Alpha	3	0.70	$1500	1.2	234	53	19%	$149
Super Sparc	3	0.70	$1700	1.6	256	48	13%	$272
Pentium	3	0.80	$1500	1.5	296	40	9%	$417

From "Estimating IC Manufacturing Costs," by Linley Gwennap, Microprocessor Report, August 2, 1993, p. 15

Add the packaging and test costs…

Chip	Die cost	Pins	Package Type	Cost	Test & Assembly	Total	Sale Price	Comment
386DX	$4	132	QFP	$1	$4	$9	$31	Intense Competition
486 DX2	$12	168	PGA	$11	$12	$35	$245	No Competition
Power PC 601	$53	304	QFP	$3	$21	$77	$280	
HP PA 7100	$73	504	PGA	$35	$16	$124		
DEC Alpha	$149	431	PGA	$30	$23	$202	$1231	Recoup R&D?
Super Sparc	$272	293	PGA	$20	$34	$326		
Pentium	$417	273	PGA	$19	$37	$473	$965	Early in Shipments

Some actual numbers

- **According to Adapteva (2014), the cost of chip development (NRE):**
  - Hardware development (Engineers): $300K - $200M
  - Software development (Engineers): $0 - $800M
  - IP Licensing: $0 - $10M
  - EDA Tools: $0 - $10M
  - Chip Tapeout (mostly masks): $100K - $3M
  - Test Development: $5K - $1M
  - **TOTAL NRE**: $1M - $1B

http://www.adapteva.com/andreas-blog/semiconductor-economics-101/

Some actual numbers

- **According to Adapteva (2014), the cost per chip:**
  - Die Cost: $0.1 - $1000
  - Package: $0.1 - $30
  - Assembly: $0.1 - $50
  - Testing: $0 - $10
  - IP Royalty: $0 - $2
  - **TOTAL Recurring Cost**: $0.3 - $1K

http://www.adapteva.com/andreas-blog/semiconductor-economics-101/

Further Reading

- J. Rabaey, "*Digital Integrated Circuits*" 2003, Chapter 1.3
- E. Alon, Berkeley *EE-141*, Lecture 2 (Fall 2009)
  http://bwrc.eecs.berkeley.edu/classes/icdesign/ee141_f09/
- http://bnrg.eecs.berkeley.edu/~randy/Courses/CS252.S96/Lecture05.pdf
- ADAPTEVA http://www.adapteva.com