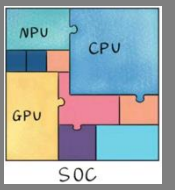


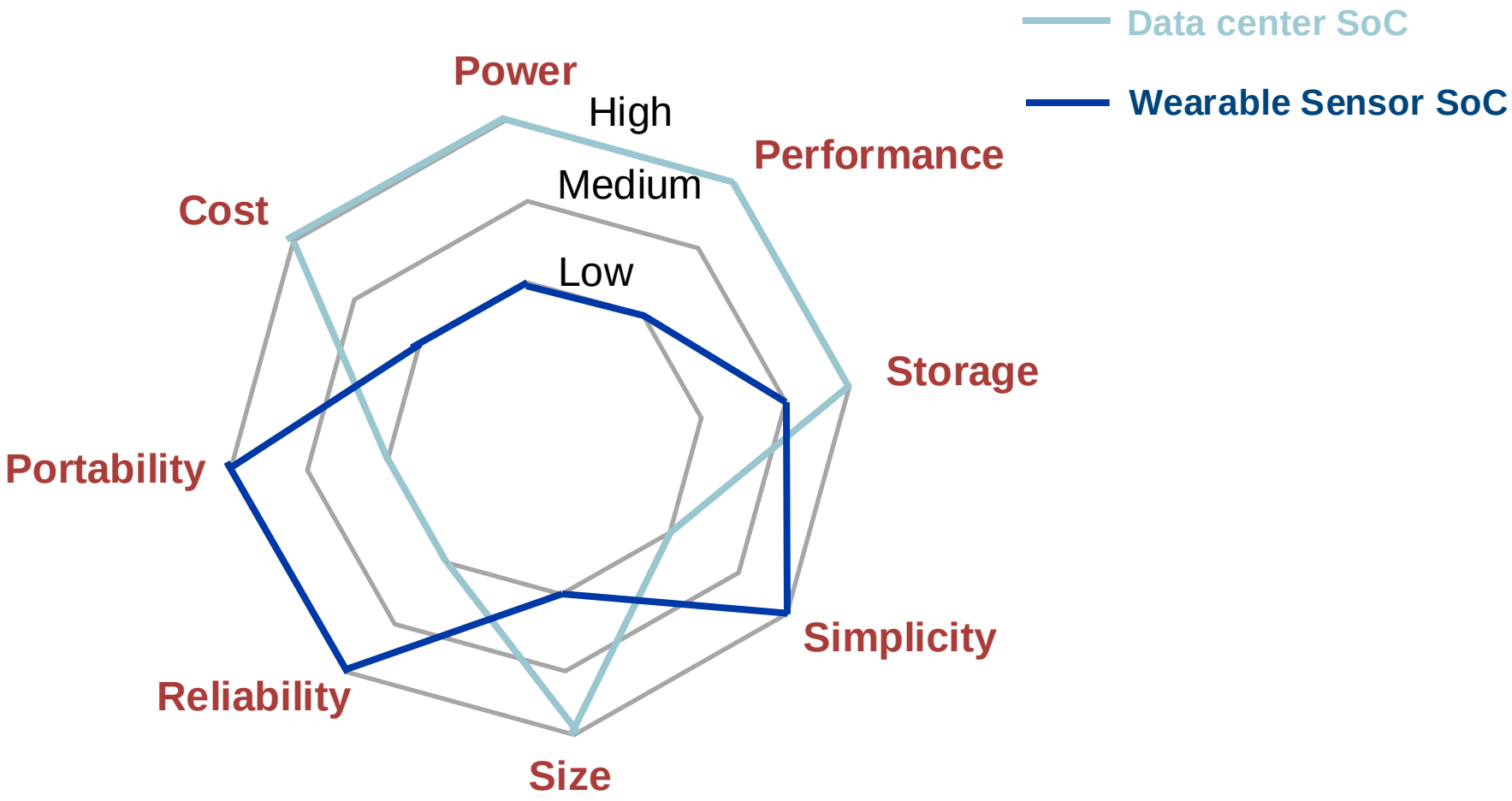
Topic 5

SoC Design Space I

Xinfei Guo
xinfei.guo@sjtu.edu.cn

November 6th, 2024

What is design space?



Adapted from [R. Sanchez-Iborra, etc. IEEE Circuits and Systems Magazine, 2020]

T5 learning goals

- Chip design space
  - Key metrics
  - Timing and Area
  - Power
  - Reliability

Challenges in SoC Design

"Microscopic Problems"

- Ultra-high speed design
- Interconnect
- Noise, Crosstalk
- Reliability, Manufacturability
- Power Dissipation
- Clock distribution.

"Macroscopic Issues"

- Time-to-Market
- Millions of Gates
- High-Level Abstractions
- Reuse & IP: Portability
- Predictability
- etc.

Everything Looks a Little Different

…and There's a Lot of Them!

?

Design Metrics

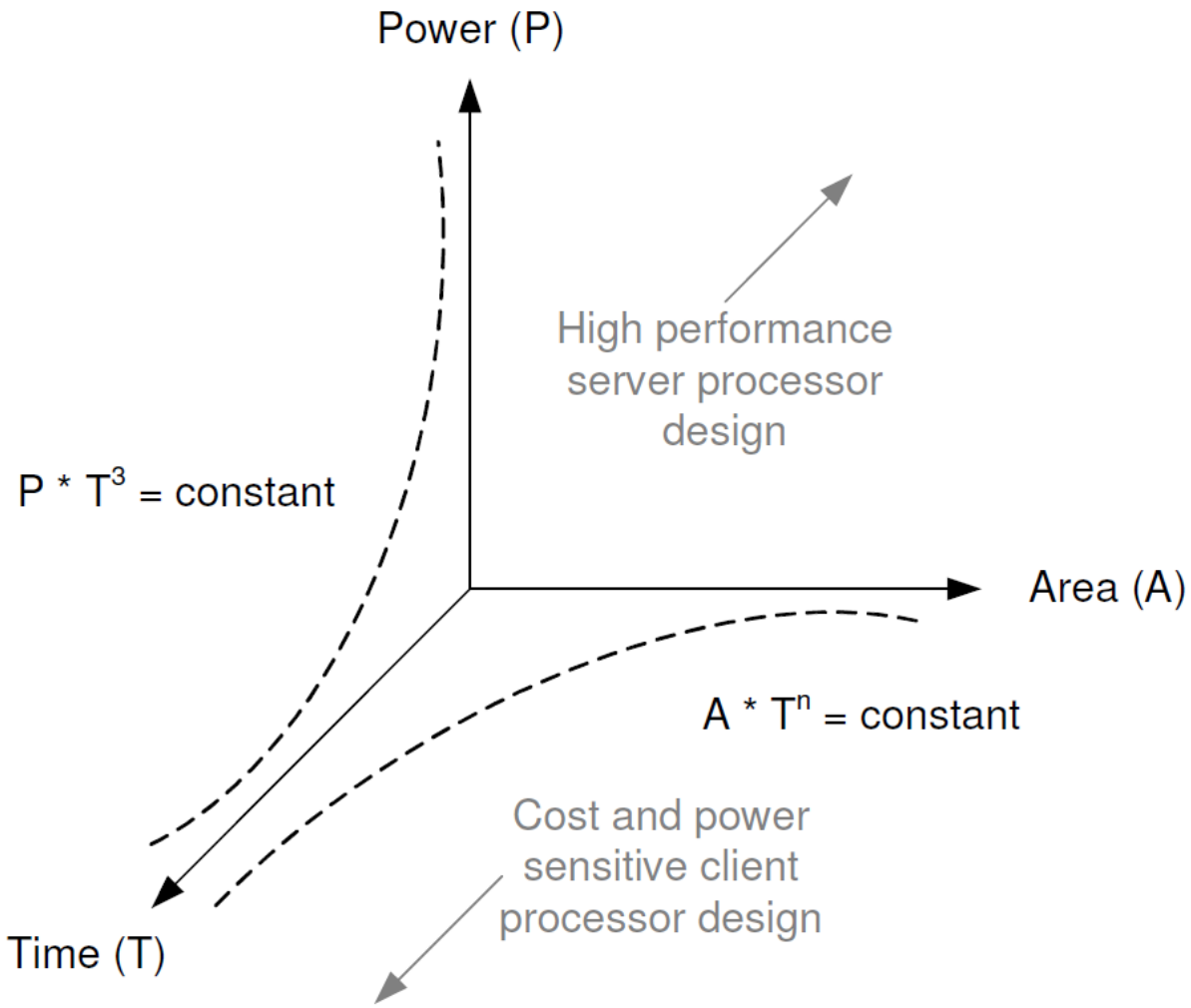
- How to evaluate performance of a digital circuit (gate, block, SoC…)?
  - Cost
  - Reliability
  - Scalability
  - Speed (delay, operating frequency)
  - Power dissipation
  - Energy to perform a function

Five Big Issues for SoC Design

- Time: Cycle time relates to Performance
- Chip Area: It also determines the IC cost, Some Instruction Sets need more chip area are less valuable than those requiring less area.
- Power Consumption: Performance as well as Implementation.
- Reliability: It relates to deep submicron effects.
- Configurability: Standardization in manufacturing and customization for application.

Tradeoffs in IP selection and design: PPA (performance, area, power)

- Increase time it takes to complete a task, decrease power
- Decrease area, decrease power consumption.
- Decrease SoC area, possible increase time. Why?



SoC Requirements & Specifications

- High-performance systems will optimize time at the expense of cost and power (probably reconfigurability).
- Low-cost systems will optimize die cost, design reuse and may be low power.
- Gaming systems have low cost - especially the production cost. However, performance with reliability is a lesser consideration.
- Wearable systems stress on low power leading to lower weight of power supply. These systems, such as cell phones, have realtime constraints and their performance cannot be ignored.
- Embedded systems used in planes (aerospace) and other safety critical applications require reliability, along with performance and design for lifetime (configurability).

PERFORMANCE

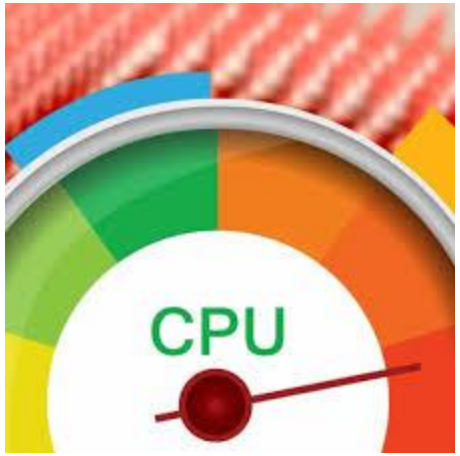
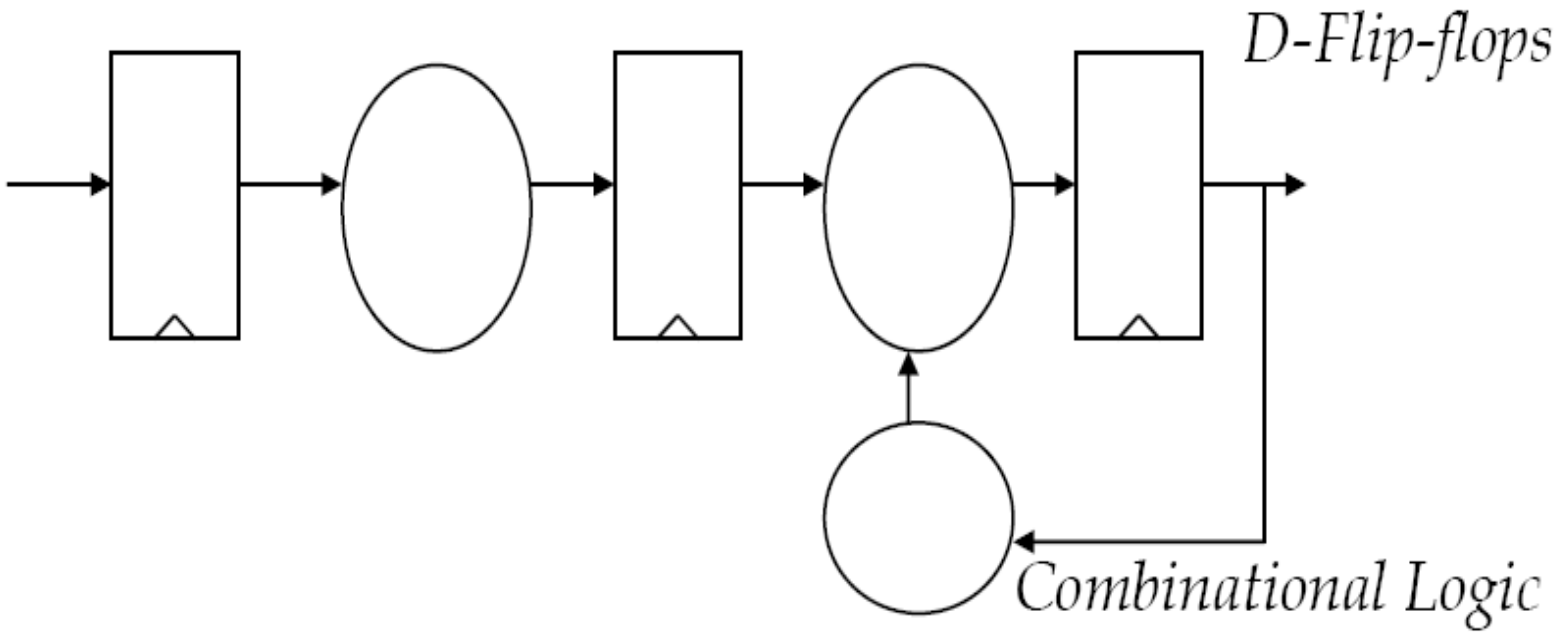


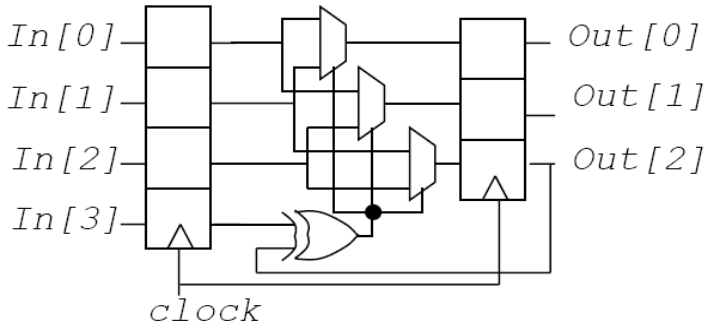
Figure: https://windowsreport.com/empty-task-manager-high-cpu/

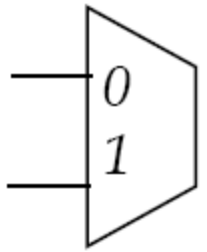
General Approach to Timing Design

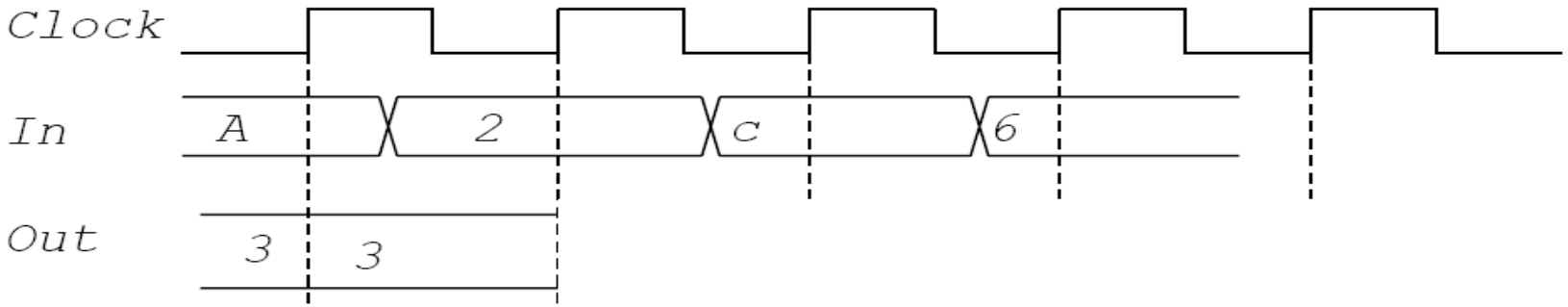
- In general, all signals start and end in registers every clock period



Clock Level Timing

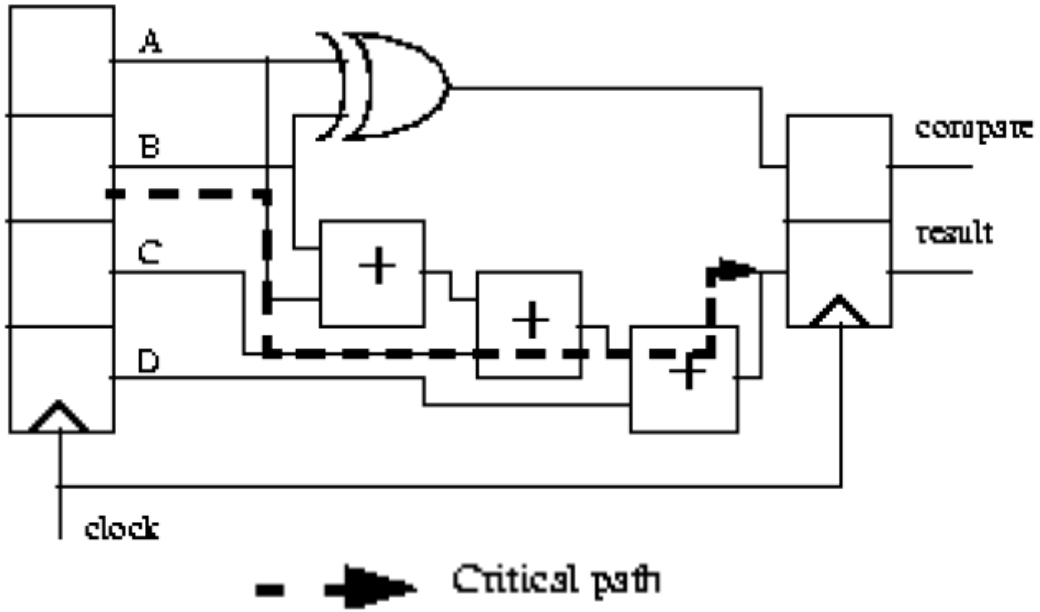






Critical Path

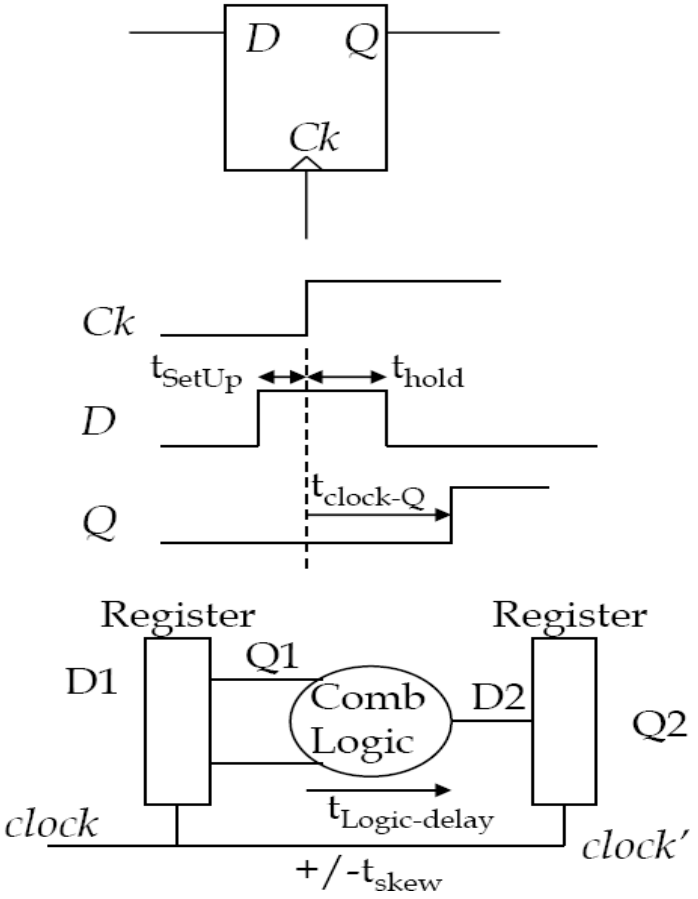
- Thus, the clock speed is determined by the slowest feasible path between registers in the design
  - *Often referred to as "the critical path"*



Critical path is longer with increased *logic depth* (# gates in series)

Flip-Flop based design

- *Edge triggered D-flip-flop*
  - *Q becomes D after clock edge*
- *Set-up time:*
  - *Data can not change no later than this point before the clock edge.*
- *Hold time:*
  - *Data can not change during this time after the clock edge.*
- *t_clock-Q*
  - *Delay on output (Q) changing from positive clock edge*



Preventing setup violations

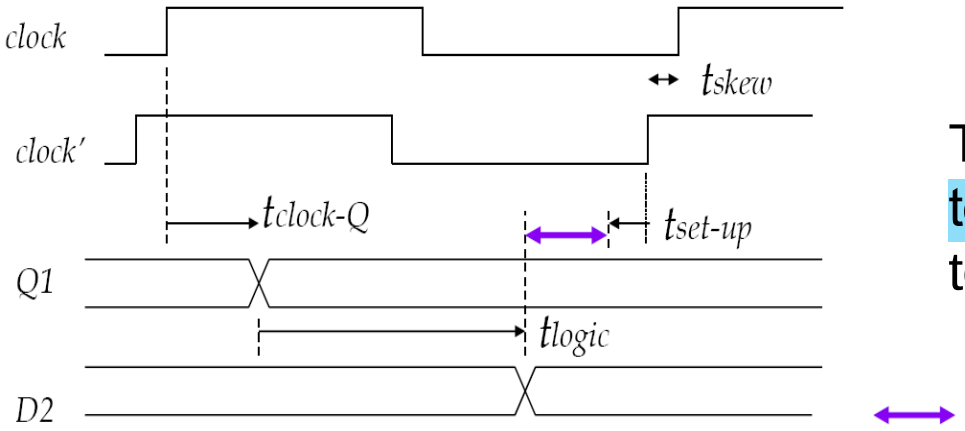
Set-up violation:

Logic is too slow for the correct logic value to arrive at the inputs to the register on the right before one set-up time before the clock edge

Constraint to prevent this:

$$t_{clock} \geq t_{clock-Q-\max} + t_{logic-\max} + t_{set-up} + t_{skew}$$

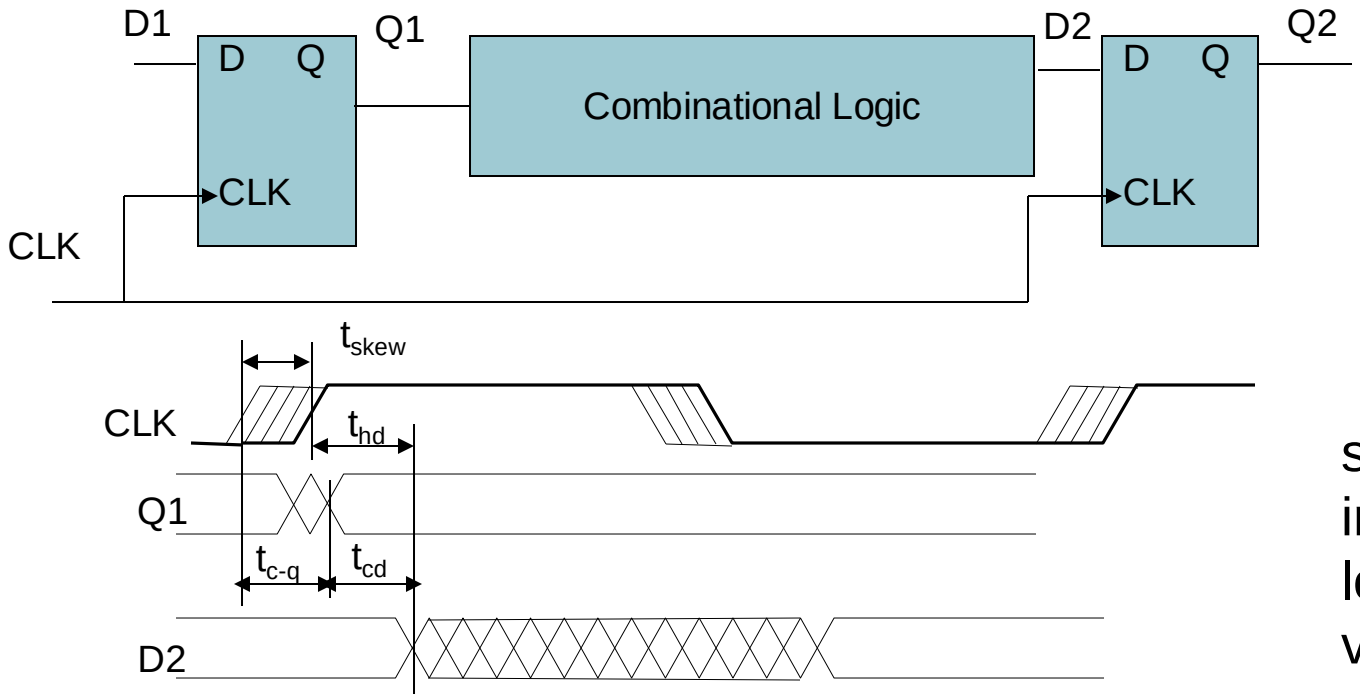
clock period



The amount of time required to turn '>' into '=' is referred to as **timing slack**

Preventing hold Violations

- Hold violations occur when race-through is possible
  Constraint to prevent hold violations:

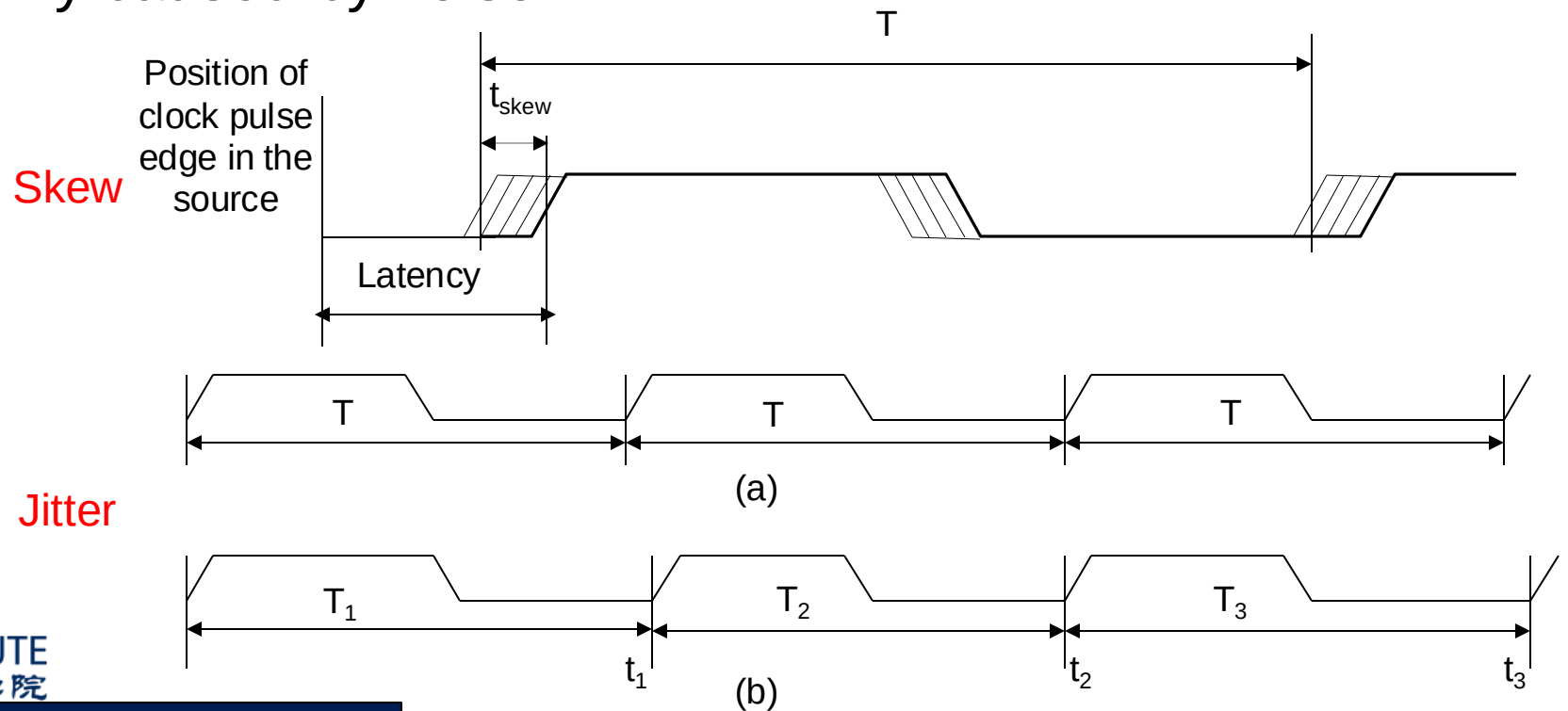


sometimes have to insert additional logic to prevent hold violations

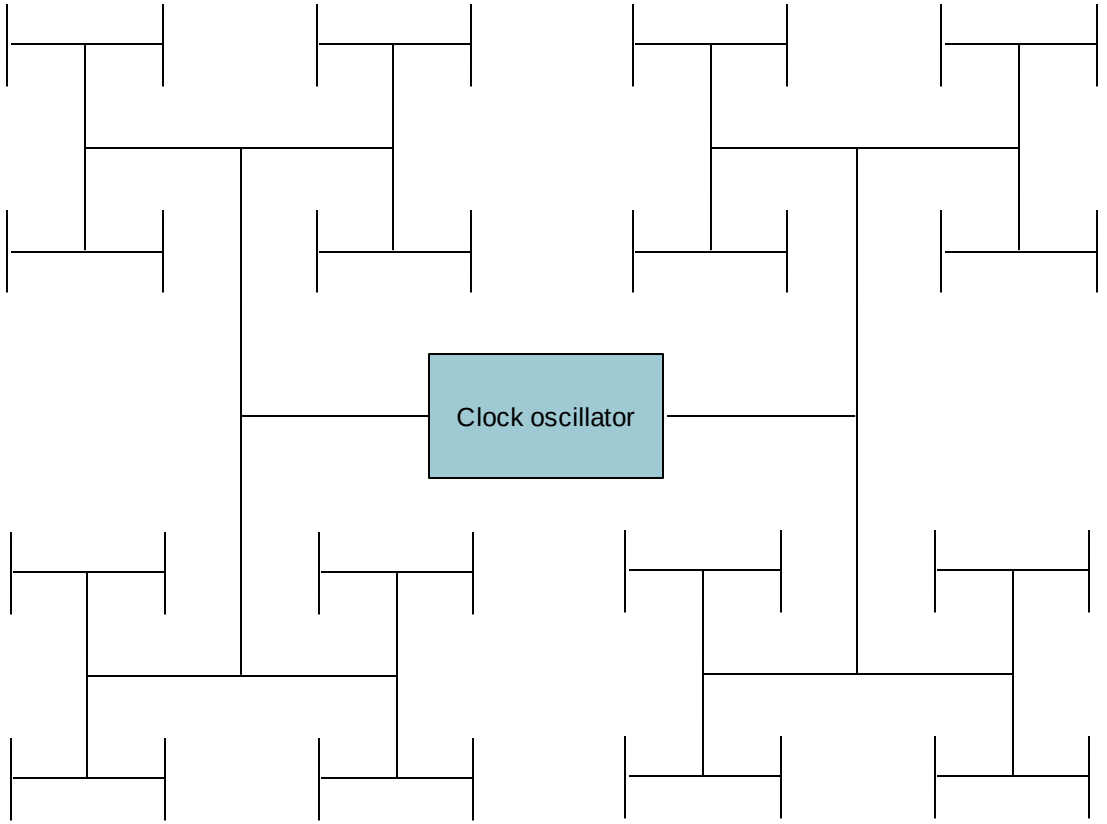
$$t_{skew} + t_{hold} < t_{c-q} + t_{cd}$$

Clock skew and jitter

- Clock skew = systematic clock edge variation between sites
  - Mainly caused by delay variations introduced by manufacturing variations
- Random variation
  - Clock jitter = variation in clock edge timing between clock cycles
  - Mainly caused by noise



Clock Signal Distribution and Propagation



- H-tree of clock signal propagation – provide minimum skew

AREA

Figure: https://www.clearchoicelaser.com/common-questions/what-does-lasik-eye-surgery-cost/

Die Area and Cost

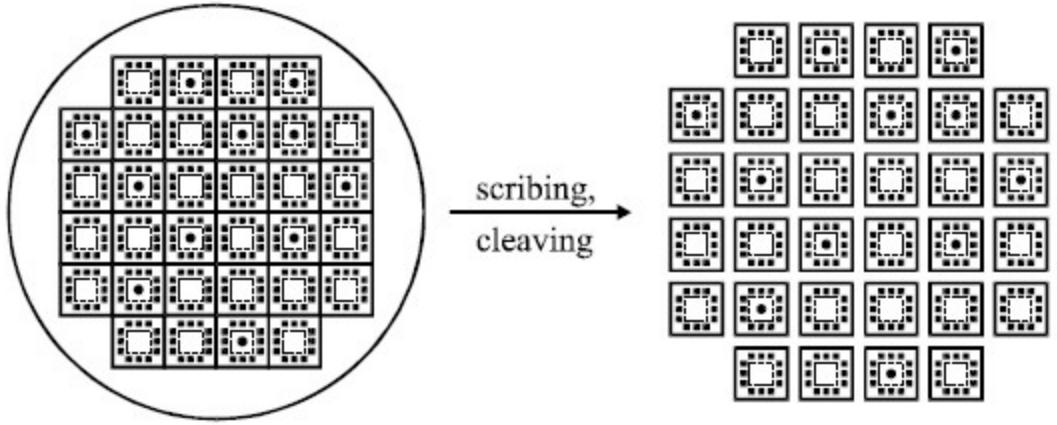
- There are significant side effects that die area has on the fixed and other variable costs
- SoCs usually have die sizes of about 10-15 mm on a side.
- The die is produced in bulk from a larger wafer, perhaps 30 cm in diameter.
- Silicon wafers and processing technologies are not perfect. Defects randomly occur over wafer surface

Die, Wafer size and other Technology Parameters

Year	2010	2013	2016
Technology generation (nm)	45	32	22
Wafer size, diameter (cm)	30	45	45
Defect density (per cm^2)	0.14	0.14	0.14
μP die size (cm^2)	1.9	2.6	2.6
Chip frequency (GHz)	5.9	7.3	9.2
Million transistors per square centimeter	1203	3403	6806
Max power (W) high performance	146	149	130

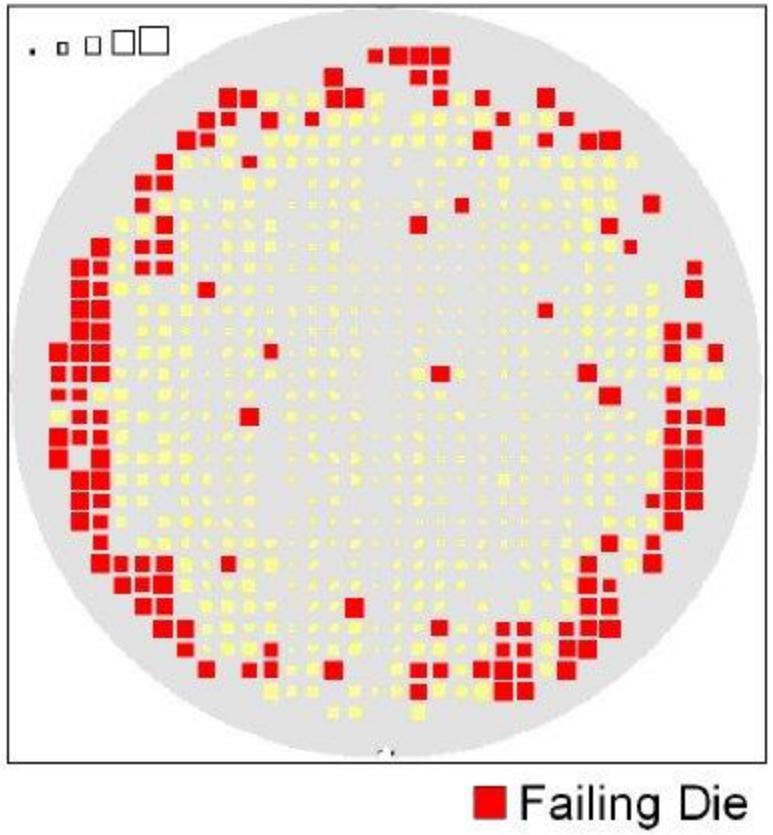
Scribing and Cleaving

- Fabricated wafers are separated into individual dice by scribing and cleaving.
- Scribing is to create a groove along scribe channels which have been left between the rows and columns of individual chips.
- Cleaving is the process of breaking the wafer apart into individual dice between the adjacent dies on a wafer



Defect

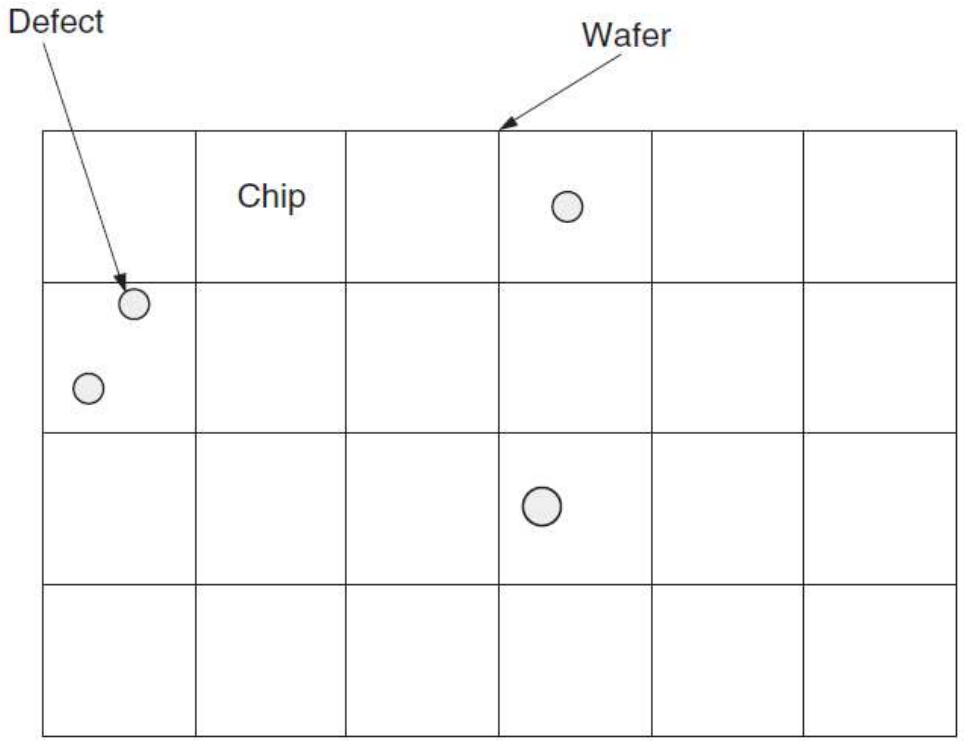
- silicon and technology processes are imperfect.
- defect can lead to failing dies

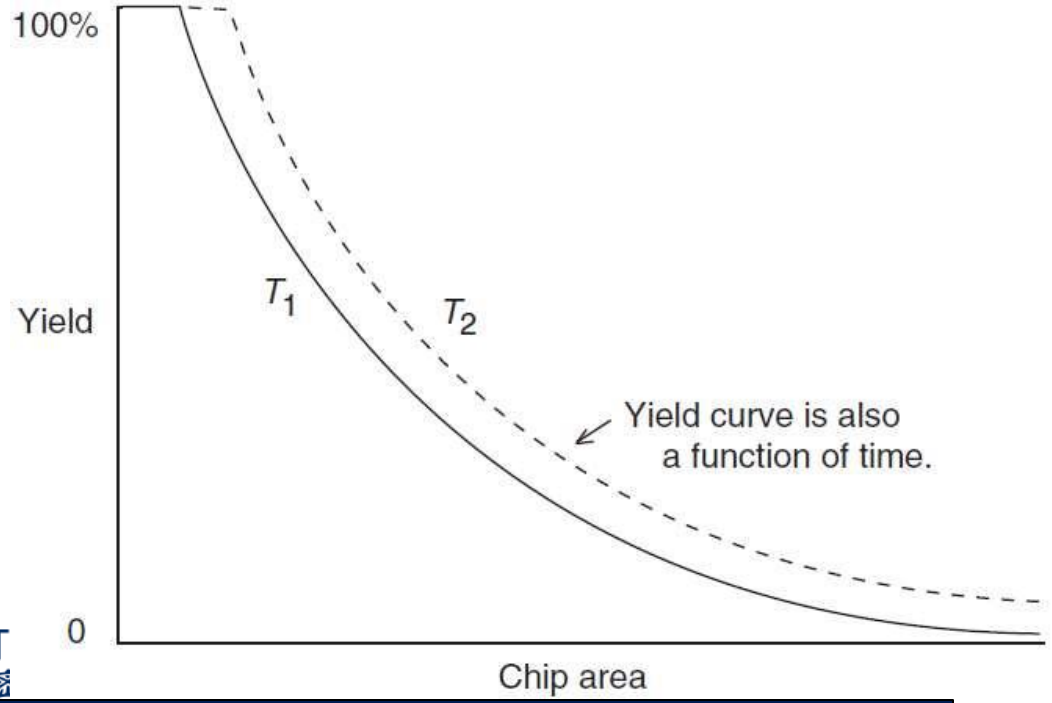


source: https://www.researchgate.net/publication/3953891_Statistical_post-processing_at_wafersort-an_alternative_to_burn-in_and_a_manufacturable_solution_to_test_limit_setting_for_sub-micron_technologies/figures?lo=1

Wafer Defects

- Defects randomly occur over the wafer surface.
- Large SoC chip area requires an absence of defects over that area
-

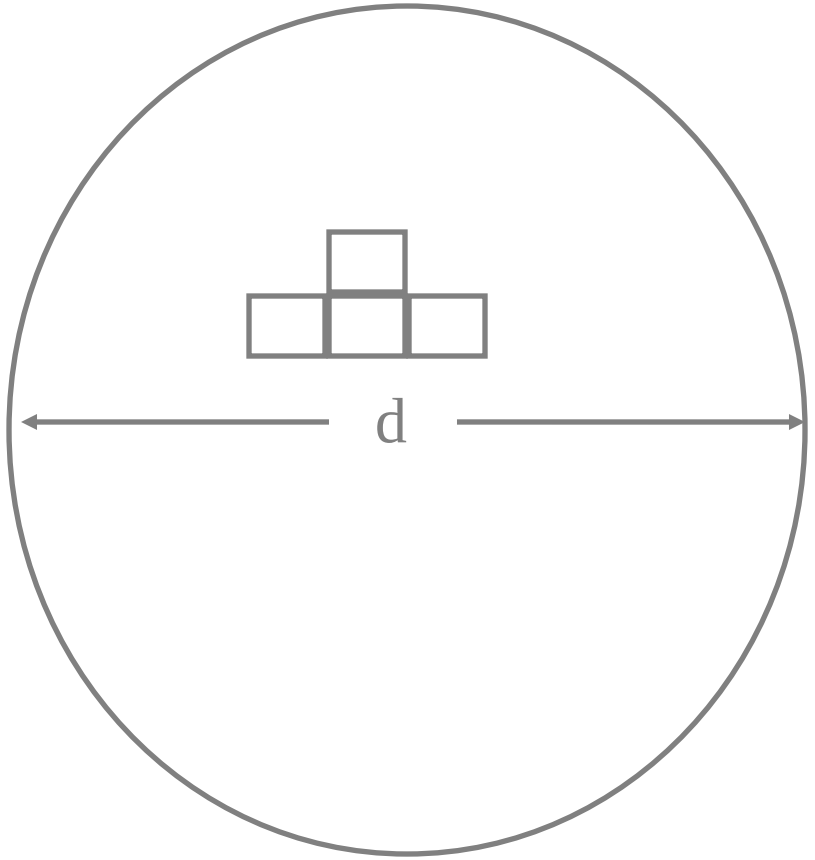




Die Area and Yield

- A good SoC design is not necessarily the one that has the maximum yield.
- Reducing the area of a design below a certain amount has only a marginal effect on yield.
  - Small designs waste chip area.
  - There is an overhead area for pins and separation between the adjacent dies on a wafer.
- Area available to a designer is a function of the manufacturing processing technology.
  - Purity of the silicon crystals,
  - Absence of dust and other impurities,
  - Overall control of the process technology.
- Improved manufacturing technology allows larger dice to be realized with higher yields.

Wafers and chips



suppose the wafer has diameter d and each die is square with area A

Wafers and chips: example

If N is the number of dice on the wafer,

$$N = \pi (d)^2 / (4A) \text{ [Gross Yield]}$$

Let N_G be number of good dice
and N_D be the number of defects on a wafer.

Given N dice of which N_G are good.....suppose we randomly add 1 new defect to the wafer. What's the probability that it strikes a good die....and changes N_G ?

Wafers and chips: example

Probability of the defect hitting a good die $= N_G / N$

The change in N_G is $d N_G / d N_D = - N_G / N$

Rewriting this we get $d N_G / N_G = - (1/N) d N_D$

Integrating and solving: $\ln(N_G) = -N_D/N + C$

Since $N_G = N \Rightarrow N_D = 0$, C must be $\ln(N)$

$N_G / N = \text{Yield} = e^{- N_D/N}$

let defect density (defects / cm^2) $= \rho_D$

$N_d = \rho_D$ x wafer area $= \rho_D$ x A x N

Yield $= N_g / N = e^{- \rho_D A}$

typically $\rho_D = 0.3 – 1.0$ defect / cm^2

Using yield to size a die

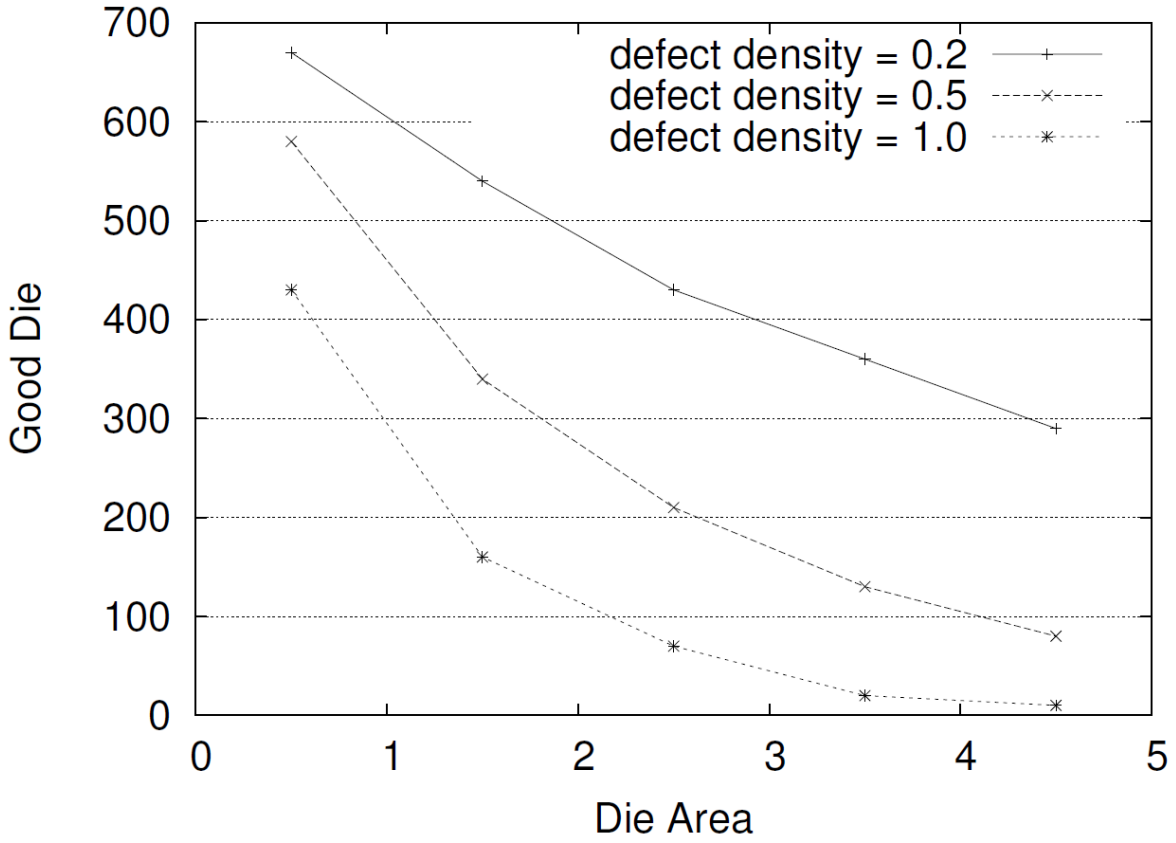
to find the cost per die:

1. find N , the number of die on a wafer
2. find Yield
3. find N_g = Yield x N
4. cost/die = wafer cost/ Ng

Wafer Diameter (cm)	Defect Density (per cm2)	Wafer Cost ($)	Die Size (cm)	Gross Yield	Yield	Good dice	Cost per good die ($)	
21	1	5000	1	314	0.37	116	$ 43	
21	1	5000	1.5	133	0.11	14	$ 357	

Wafer Defects

- Large die sizes are very costly. Doubling the die area has a significant effect on the yield for a large $\rho_D \times A$ *(≈ 5 – 10 or more).*
- A modern fab. facility would have a ρ_D *of* (0.15 → 0.5) It depends on the maturity of the process and the expense charges by the fab. facility



What can be put on the die?

- depends on the lithography and die area
- lithography determined by f, minimum feature size
- feature size is related to λ, the mask registration variation
  - f = 2 λ

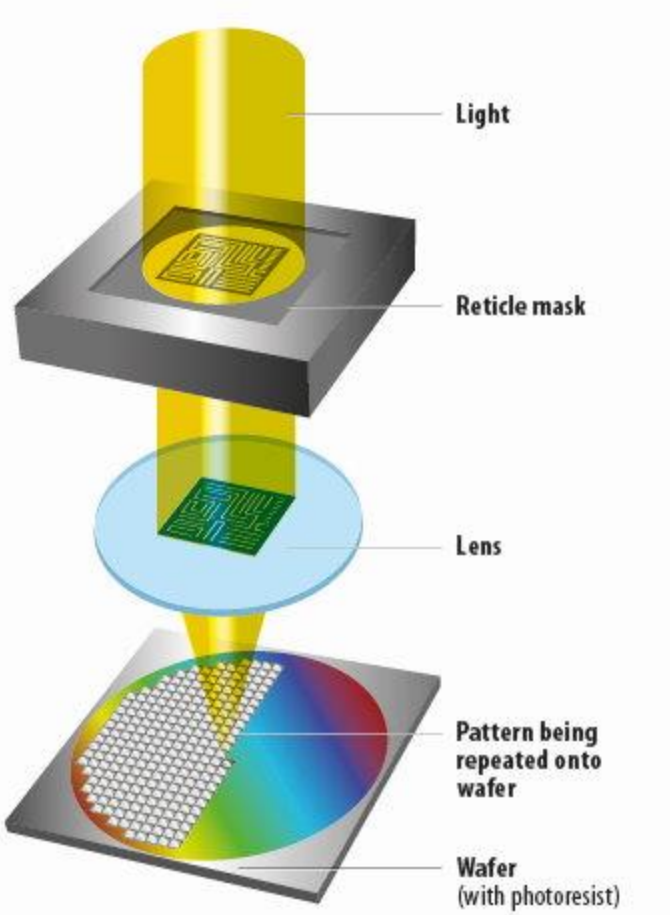
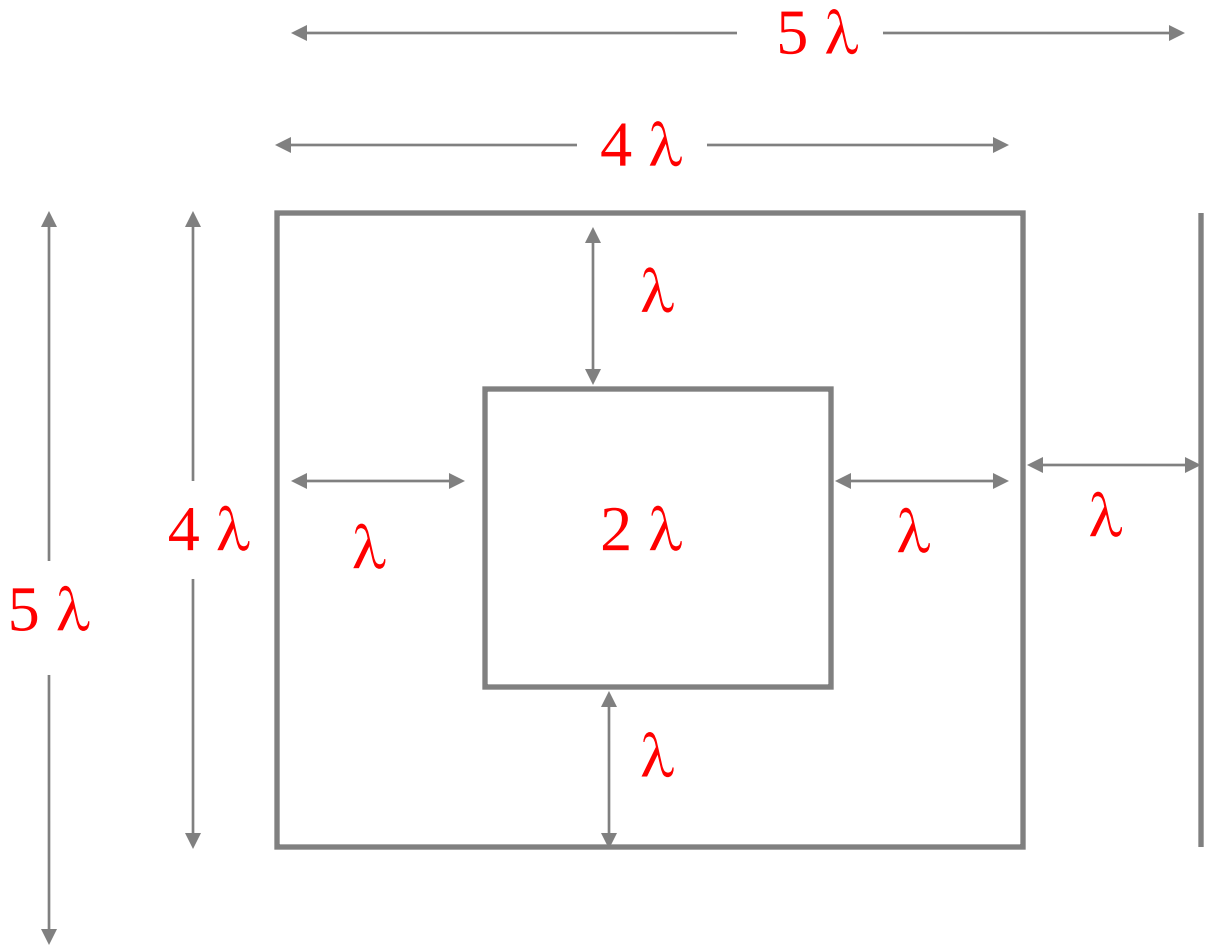


Figure: https://bits-chips.nl/artikel/euv-for-dummies/

Feature and Area Unit

- A mm^2 area unit is good, but photolithography and geometries' resulting minimum feature sizes are constantly shifting, a dimensionless unit is preferred.
- A unit λ is the distance from which a geometric feature on any one layer of mask may be positioned from another.
- A transistor is $4\lambda^2$, positioned in a minimum region of $25\lambda^2$ (Next slide).
- The minimum feature size, ***f*** is the length of one Poly-silicon gate, or the length of one transistor, **f** = 2λ.

Smallest device: 5 λ x 5 λ



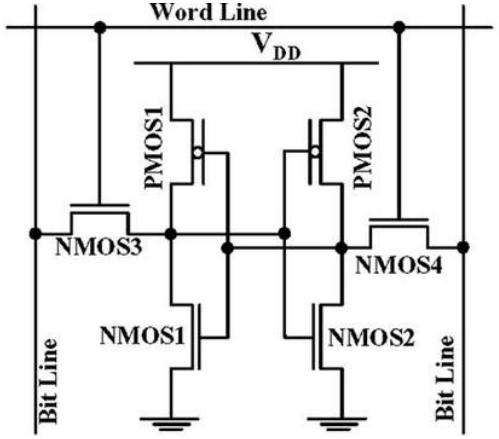
Area Units: rbe and A

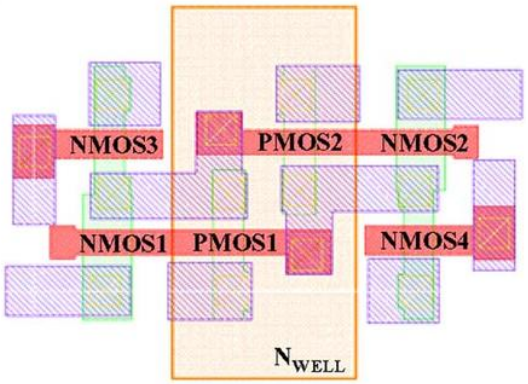
- Register bit equivalent (rbe) : small area unit for sizing functional units of the processor, a useful unit defined to be a 6-transistor register **(memory)** cell
- Suppose we define another larger unit, A, as $1A = f^2 \times 10^6$, then $1A = 10^6 / 675 = 1481$ rbe
- since 1481 is close to 1444 we can also refer to the simple register file as occupying 1 A (This is also the area occupied by a 32×32 bit three-ported register file)

Unit	Relative Size
λ **mask registration**	
f **minimum feature size**	**$f = 2\lambda$**
rbe **register bit equivalent**	**$rbe = 2700\lambda^2 = 675 f^2$**
A **functional unit area**	**$A = 10^6 f^2 = 1481$ rbe**

The area units

Unit	Relative Size
λ mask registration	
f minimum feature size	f = 2 λ
rbe register bit equivalent	rbe = 2700 λ^2 = 675 f^2
A functional unit area	A = 10^6 f^2 = 1481 rbe





Q: Why $2700\lambda^2$

This is defined to be a six-transistor register cell. It is significantly more than 6x the area of a single transistor, since it includes larger transistor, their interconnections and necessary inter-bit isolating spaces.

source: "Altitude SEE test European platform (ASTEP) and first results in CMOS 130 nm SRAM"

Area of other cells

1 register bit (rbe)	1.0 rbe
1 static RAM bit in an on-chip cache	0.6 rbe
1 DRAM bit	0.1 rbe
rbe corresponds to (in feature size: f)	1 rbe = $675f^2$
Item: Size in A Units	
A corresponds to 1 mm^2 with $f = 1\,\mu m$.	
$1\,A$	$=f^2 \times 10^6$ (f in μm)
or about	$\approx$1481 rbe
A simple integer file (1 read + 1 read/write) with 32 words of 32 bits per word	=1444 rbe
or about	$\approx 1\,A$ (=0.975 A)
A 4-KB direct mapped cache	=23,542 rbe
or about	$\approx 16\,A$
Generally a simple cache (whose tag and control bits are less than one-fifth the data bits) uses	$=4\,A/KB$
Simple Processors (Approximation)	
A 32-bit processor (no cache and no floating point)	$=50\,A$
A 32-bit processor (no cache but includes 64-bit floating point)	$=100\,A$
A 32-bit (signal) processor, as above, with vector facilities but no cache or vector memory	$=200\,A$
Area for interunit latches, buses, control, and clocking	Allow an additional 50% of the processor area.

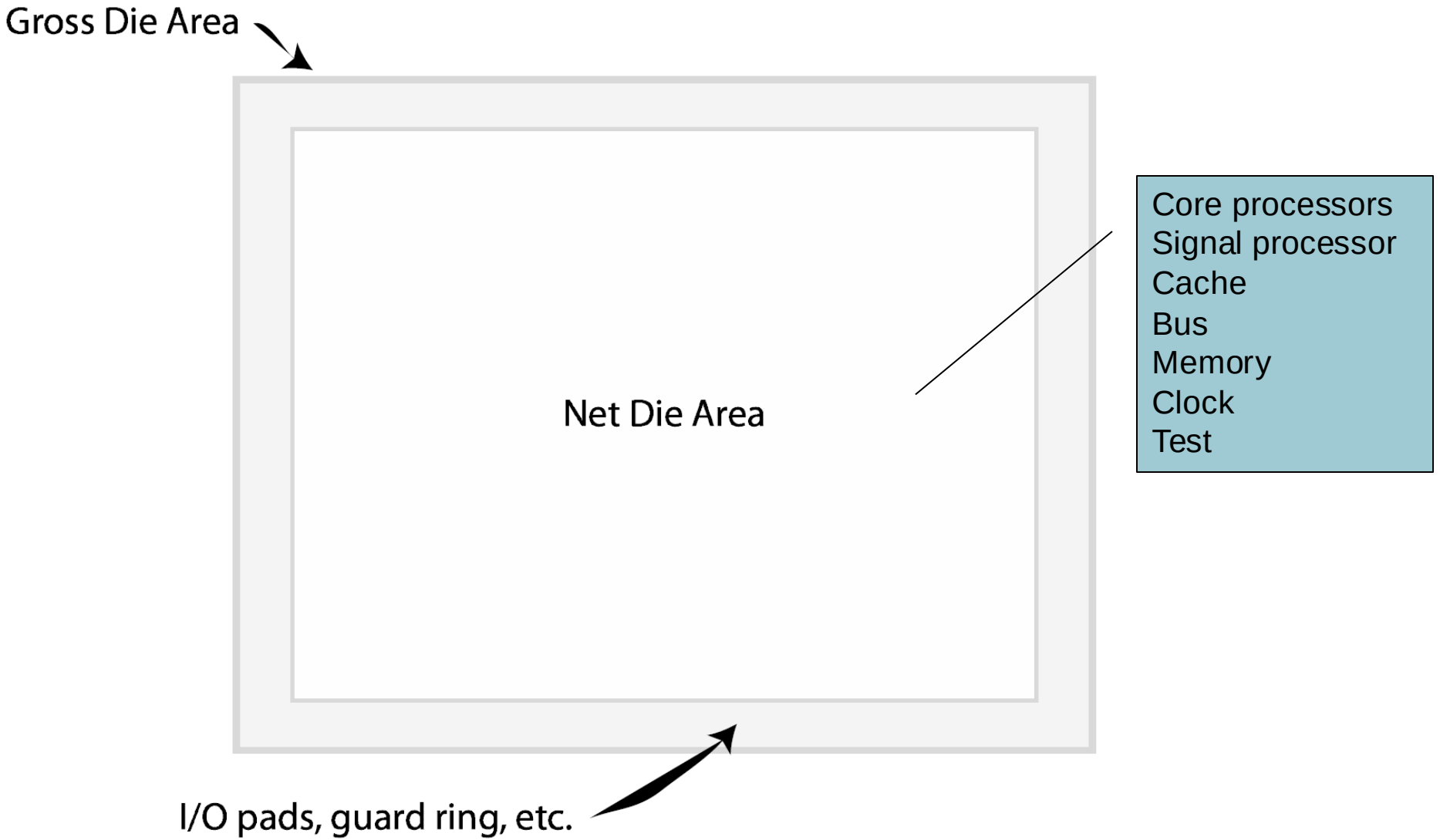
These are the parameters for basic cells in most design tradeoffs

source: Dr. Gul N. Khan

SoC Die floorplanning methodology

- pick target cost based on market requirements
- determine total area available within cost budget
  - defect and yield model
- compute net available area for processors, caches and memory
  - account for I/O, buses, test hooks, I/O pads etc.
- select core processors and assess area and performance
- re-allocate area to optimize performance
  - cache, signal processors, multimedia processors, etc.

Floorplan and area allocation



The baseline: I

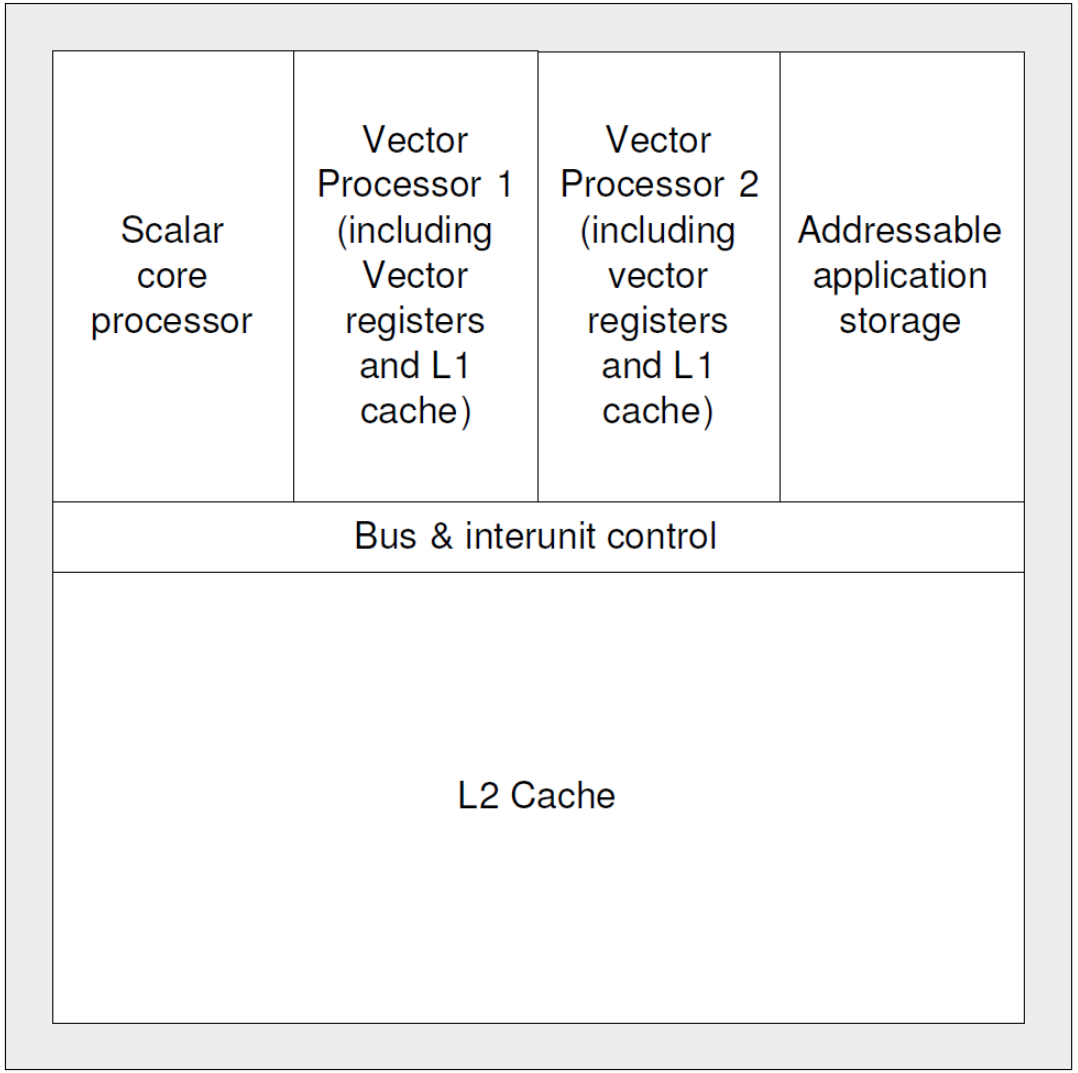
- suppose ρ_d is 0.2 defects /cm^2 and we target 80% yield
- then A = 110 mm^2 gross or (allowing 20% for periphery) guard 88 mm^2 net
- if f = 0.13 μ we have 5200A area units for our design
- we want to realize
  - a 32b core processor (w 8kB I & 16kB D cache)
  - 2 32b Vector proc. W 16 x 1k x 32 vector memory + I and D cache
  - 128kB ROM
  - anything else is SRAM (then decide how many bits we can store on chip)

The baseline: II

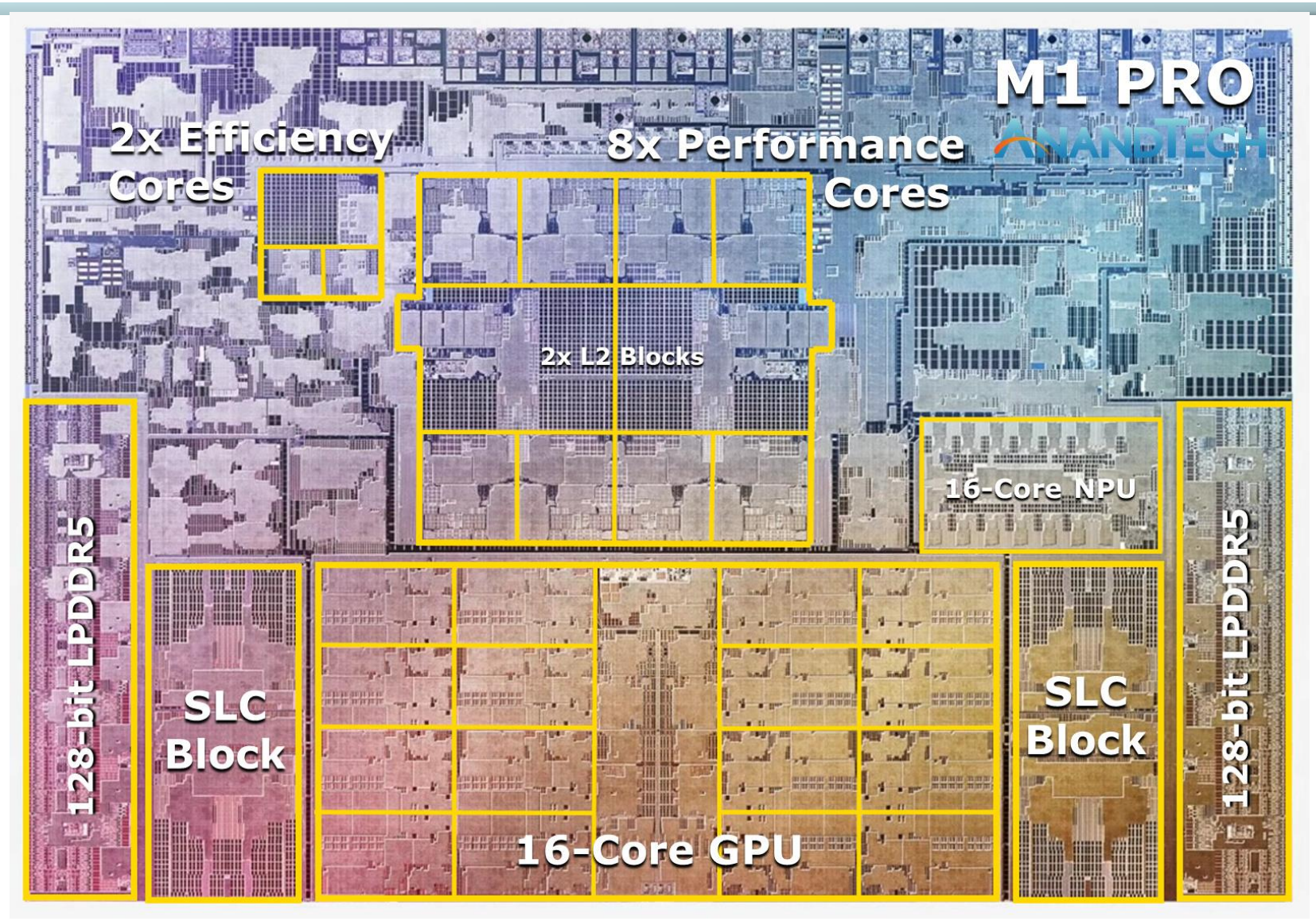
Unit	*Area*
Core Processor (32^b)	$100A$
Core cache ($24KB$)	$96A$
Vector Processor #1	$200A$
Vector Registers & cache #1	$256 + 96A$
Vector Processor #2	$200A$
Vector Registers & cache #2	$352A$
Bus and bus control (50%)	$650A$
Application memory ($128KB$)	$512A$
Subtotal	$2{,}462A$

This leaves 5200 - 2462 = 2538*A* available for data SRAM
This implies about 512kB of SRAM

Example SOC floorplan



Example: Appl M1 Pro



source: anandtech

SoC Area Design Rules

Feature Size (μm)	Number of A per mm^2
1.000	1.00
0.350	8.16
0.130	59.17
0.090	123.46
0.065	236.69
0.045	493.93

1. Compute the target chip size using the yield and defect density.
2. Compute the die cost and determine whether it is satisfactory.
3. Compute the net available area. Allow 10 – 20% (or other appropriate factor) for pins, guard ring, power supplies, etc.
4. Determine the **rbe size from the minimum feature size.**
5. Allocate the area based on a trial system architecture until the basic system size is determined.
6. Subtract the basic system size (5) from the net available area (3). This is the die area available for cache and storage optimization.

source: Dr. Gul N. Khan

Area and Costs

- When we increase area, we will more than likely be:
  - Increasing complexity of the design
  - Increasing the HW design effort
  - Increasing power
  - Increasing time-to-market
  - *Increasing documentation for the product*
  - *Increasing the effort to service the system*

source: Dr. Gul N. Khan

SoC Area summary

- cost: an exponential function of area
- successful business model
  - targets initial production at relatively low yield (~0.3)
  - ride learning curve and leverage technology to reduce cost and improve performance
- technical innovation and analysis
  - intersect with business decisions to make a product
  - use design feasibility studies and empirical targets
  - methodology for cost and performance evaluation
  - marketing targets: determine weighting of performance metrics

Where are we Heading?

- SoC Design Spaces II

Action Items

- SoC Review Assignment is due!
- Reading Materials
  - Ch. 2

Acknowledgement

Slides in this topic are inspired in part by material developed and copyright by:

- Dr. Wayne Luk (Imperial College)
- Dr. Gul N. Khan (Ryerson University)
- Dr. Andreas Gerstlauer (UT Austin)
- Dr. Anand Raghunathan (Purdue)
- Dr. Konstantinos Tatas
- Dr. Paul Franzon (NCSU)