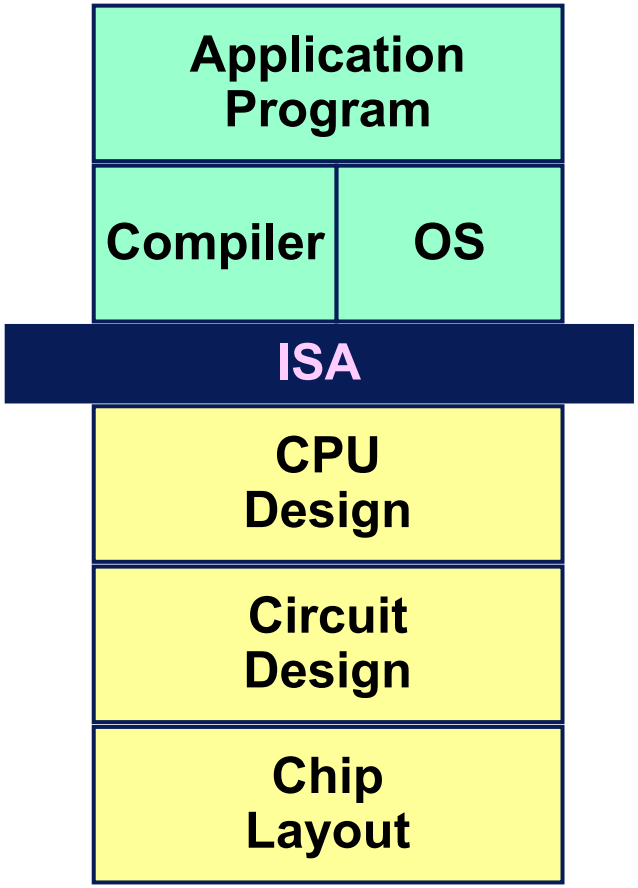
Computer Architecture

—Basic Concept

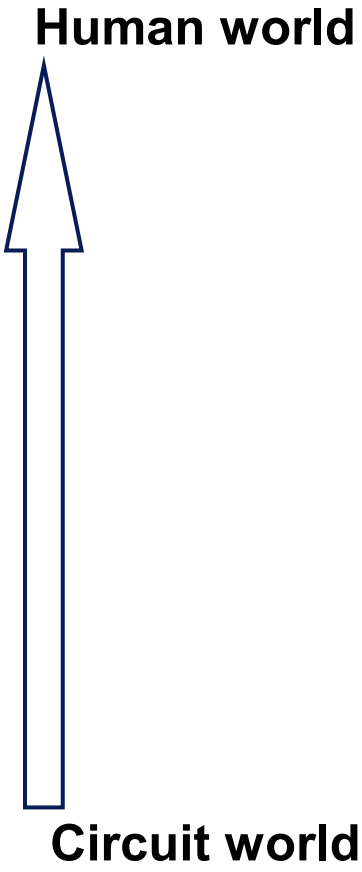
What is "Computer System"?

- Coordination of many *levels of abstraction*
- Under a rapidly changing set of forces
- Design, Measurement, *and* Evaluation



Machine Abstraction Layers

- Functional Language (e.g. Prolog) Level
- OOP Language & Framework & Generic Programming etc. Level
- Preemptive Language (e.g. C) Level
- Assembly Language Level
- Machine Language Level
- Micro-ops Level or called Micro-architecture Level



Machine Abstraction Layers (cont.)

- **Higher level abstraction is fitter to real world.**
  - **Every level abstraction is a way of thinking.**
  - **Abstraction layers is to fill the gap between human world and the circuit world.**
  - **Human invent higher level abstraction to escape from the circuit and to meet the need of real world.**
- **Higher level abstraction provide higher portability.**
- **Every level abstraction loss some performance.**
  - **e.g. saturation operation in C**
- **There will be more and more machine abstraction layers.**

Computer Architecture's Changing Definition

- **1950s to 1960s: Computer Architecture Course: Computer Arithmetic**
- **1970s to mid 1980s: Computer Architecture Course: Instruction Set Design, especially ISA appropriate for compilers**
- **1990s: Computer Architecture Course:**
  Design of CPU, memory system, I/O system, Multiprocessors, Networks
- **2010s: Computer Architecture Course: Multi-Core? Self adapting systems? Self organizing structures?**
  DNA Systems/Quantum Computing? Cloud Computing? AI Computing?

What is “Computer Architecture”

Computer Architecture =

Instruction Set Architecture (ISA) +

Machine Organization +

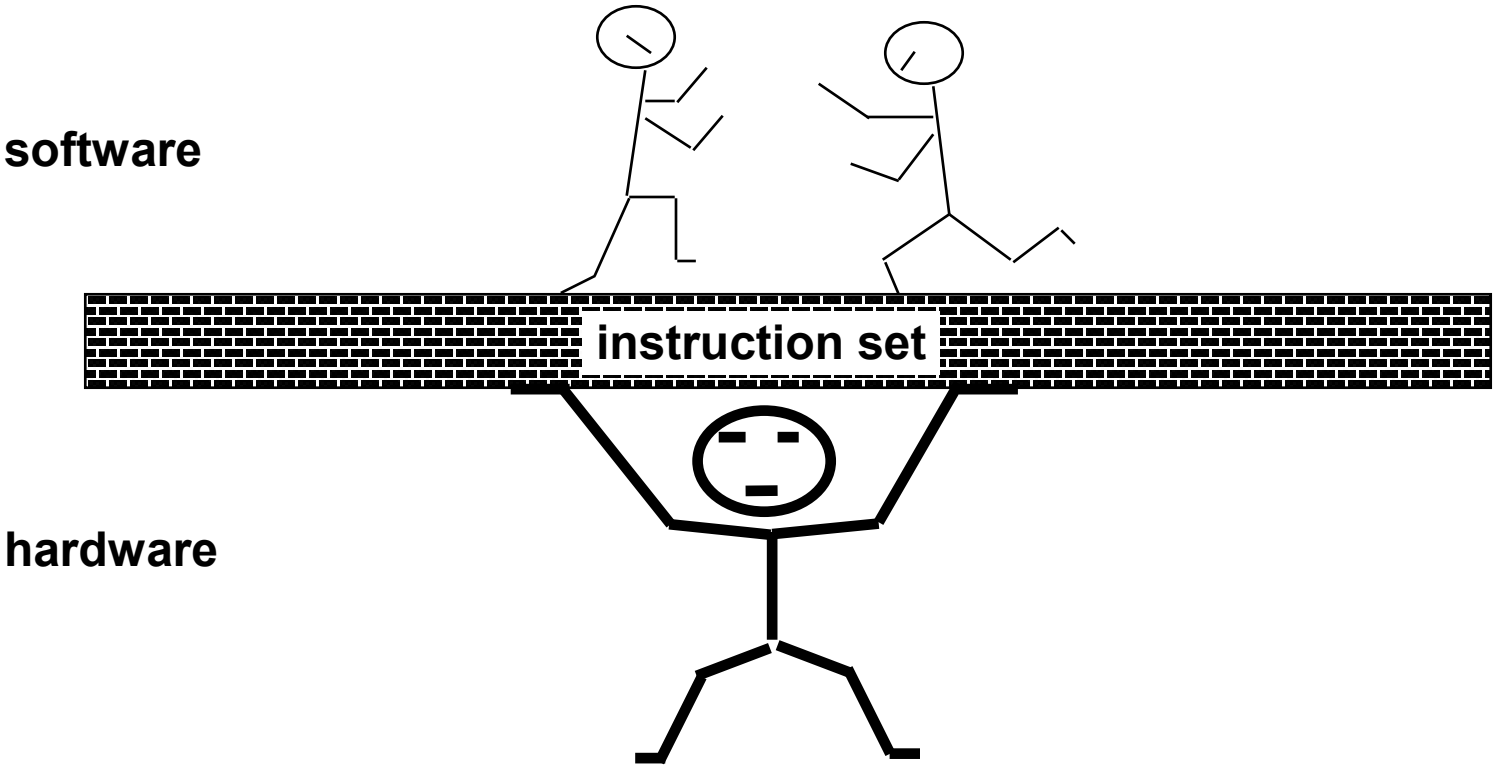
Physical Implementation…..

Instruction Set Architecture (subset of Computer Arch.)

... the attributes of a [computing] system as seen by the programmer, *i.e.* the conceptual structure and functional behavior, as distinct from the organization of the data flows and controls, the logic design, and the physical implementation.

– Amdahl, Blaaw, and Brooks, 1964

The Instruction Set: a Critical Interface



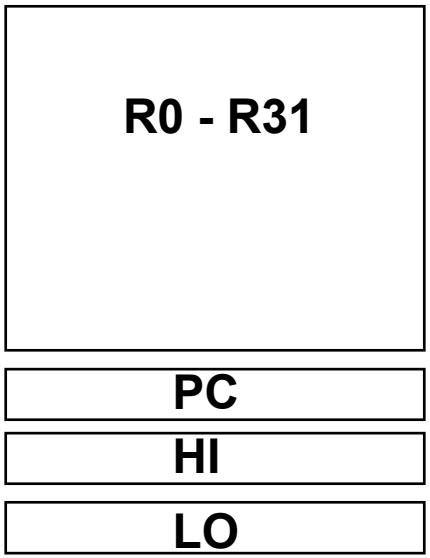
Example ISAs (Instruction Set Architectures)

Digital Alpha	(v1, v3)	1992-1997
HP PA-RISC	(v1.0, v1.1, v2.0)	1986-1996
Sun Sparc	(v7, v8, v9, UltraSPARC)	1986-2007
SGI MIPS	(MIPS I, II, III, IV, V)	1986-1996
Intel	(8086, 80286, 80386, 80486, Pentium, MMX, ...)	1978-1996
	IA-64	2001
	Intel 64	2004
ARM	(ARM v1, v2, v3, v4, v5, v6, v7, v8, v9)	1985-2021
RISC-V	RISC-I, RISC-II, RISC-III (SOAR), RISC-IV (SPUR)	1981-1988
	RISC-V	2011

MIPS R3000 Instruction Set Architecture (Summary)

- **Instruction Categories**
  - Load/Store
  - Computational
  - Jump and Branch
  - Floating Point
    - coprocessor
  - Memory Management
  - Special

Registers



3 Instruction Formats: all 32 bits wide

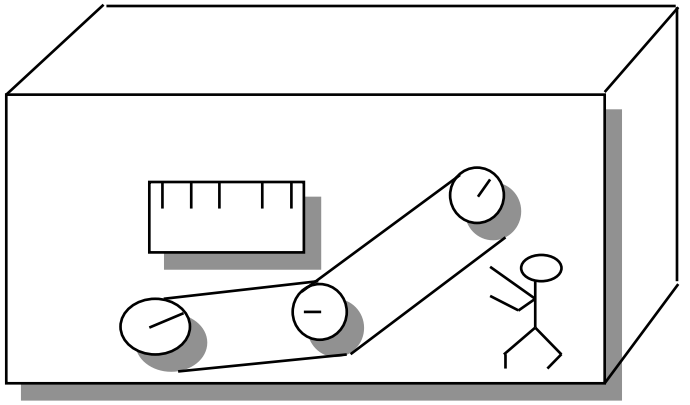
Machine Organization

- Capabilities & Performance Characteristics of Principal Functional Units
  - (e.g., Registers, ALU, Shifters, Logic Units, ...)
- Ways in which these components are interconnected
- Information flows between components
- Logic and means by which such information flow is controlled.
- Choreography of FUs to realize the ISA
- Register Transfer Level (RTL) Description

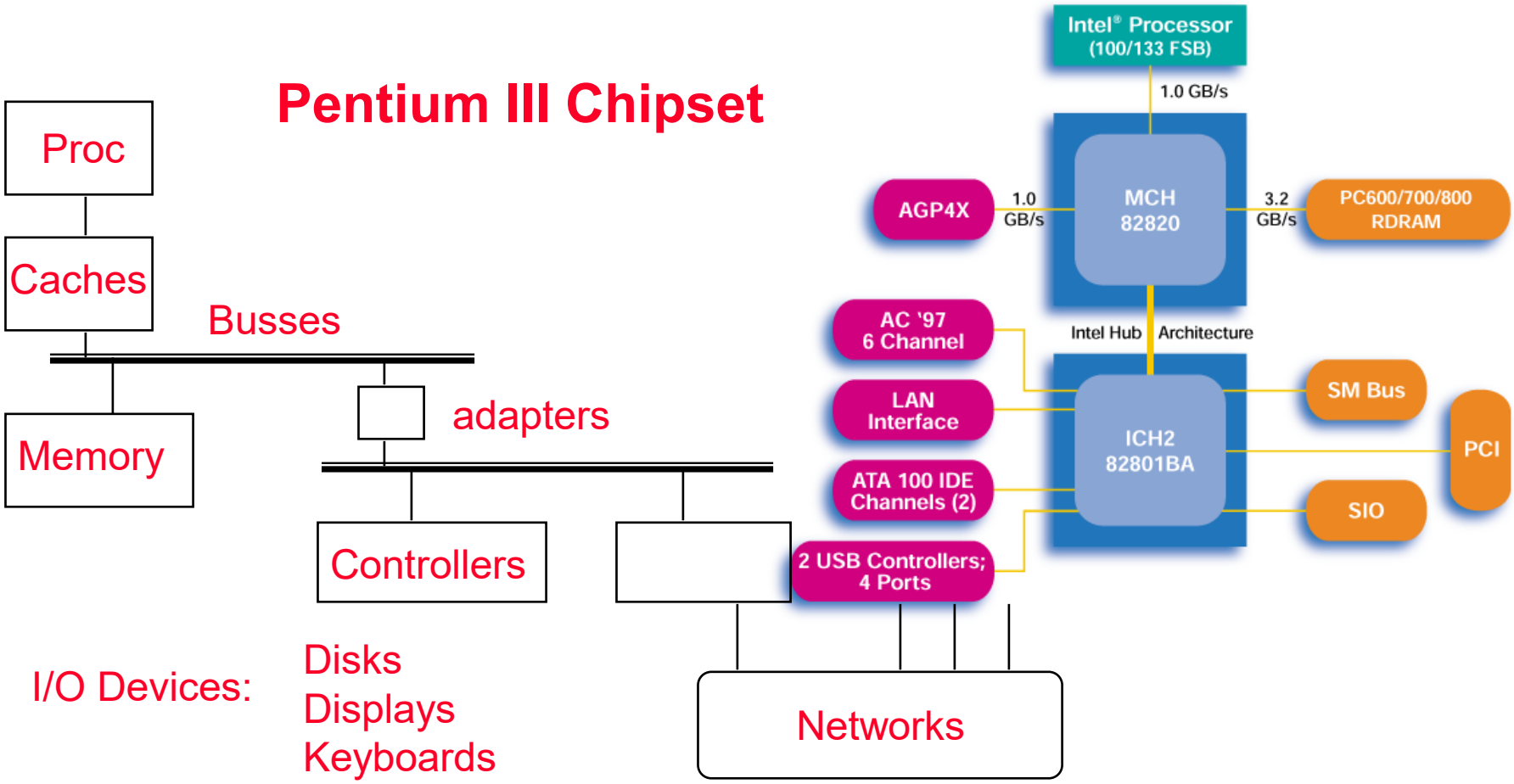
Logic Designer's View

ISA Level

FUs & Interconnect

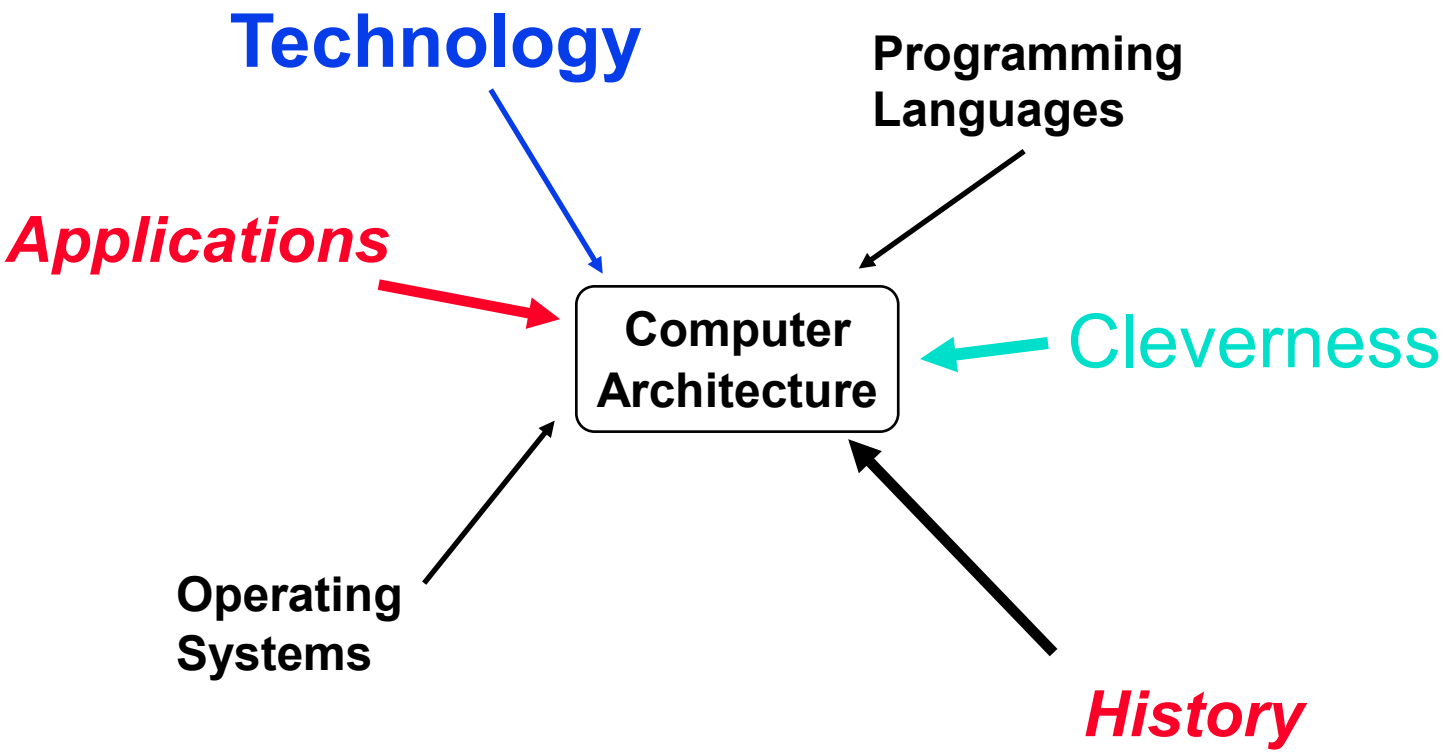


Sample Organization: It's all about communication



- **All have interfaces & organizations**

Forces on Computer Architecture



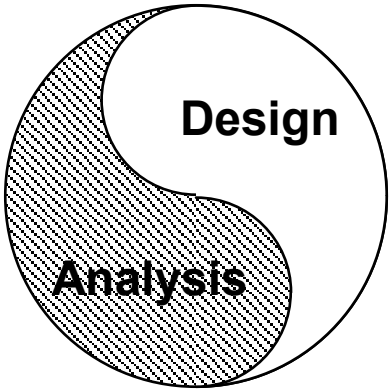
Applications and Languages

- **CAD, CAM, CAE …**
- **Lotus, DOS …**
- **Multimedia …**
- **The Web …**
- **Cloud**
- **Big-data**
- **Quantum**
- **AI**
- **……**

Three Design Level

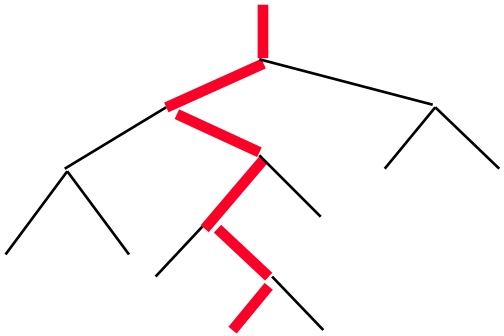
- **Bottom-up**
- **Middle-out**
- **Top-down**

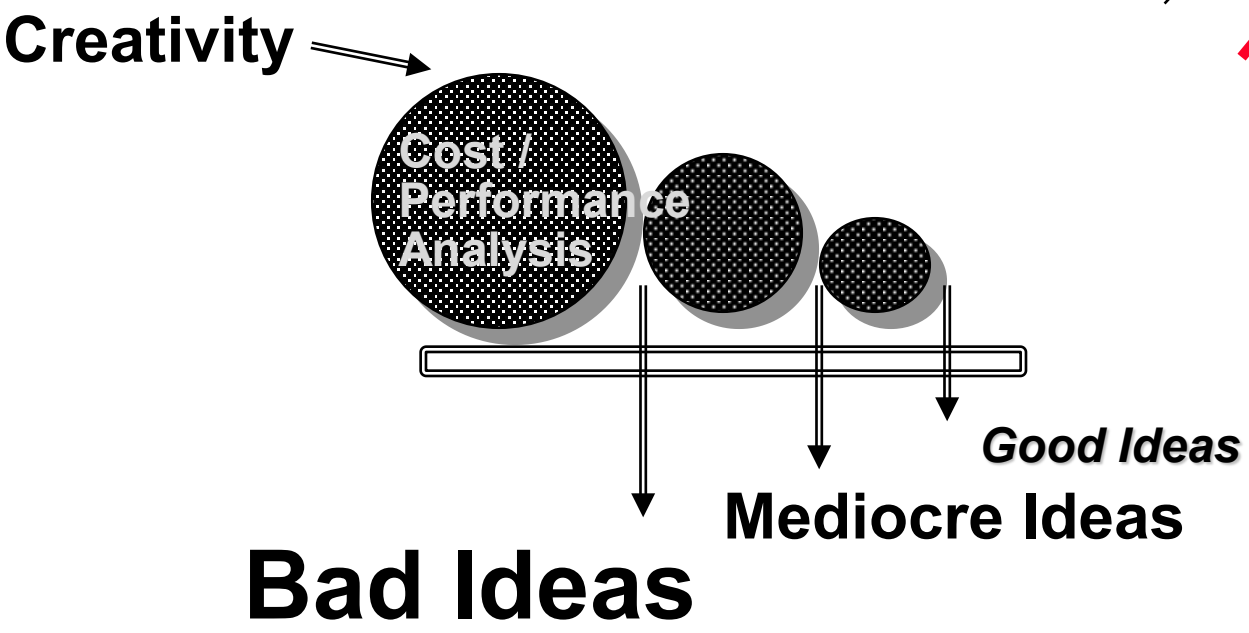
Measurement and Evaluation



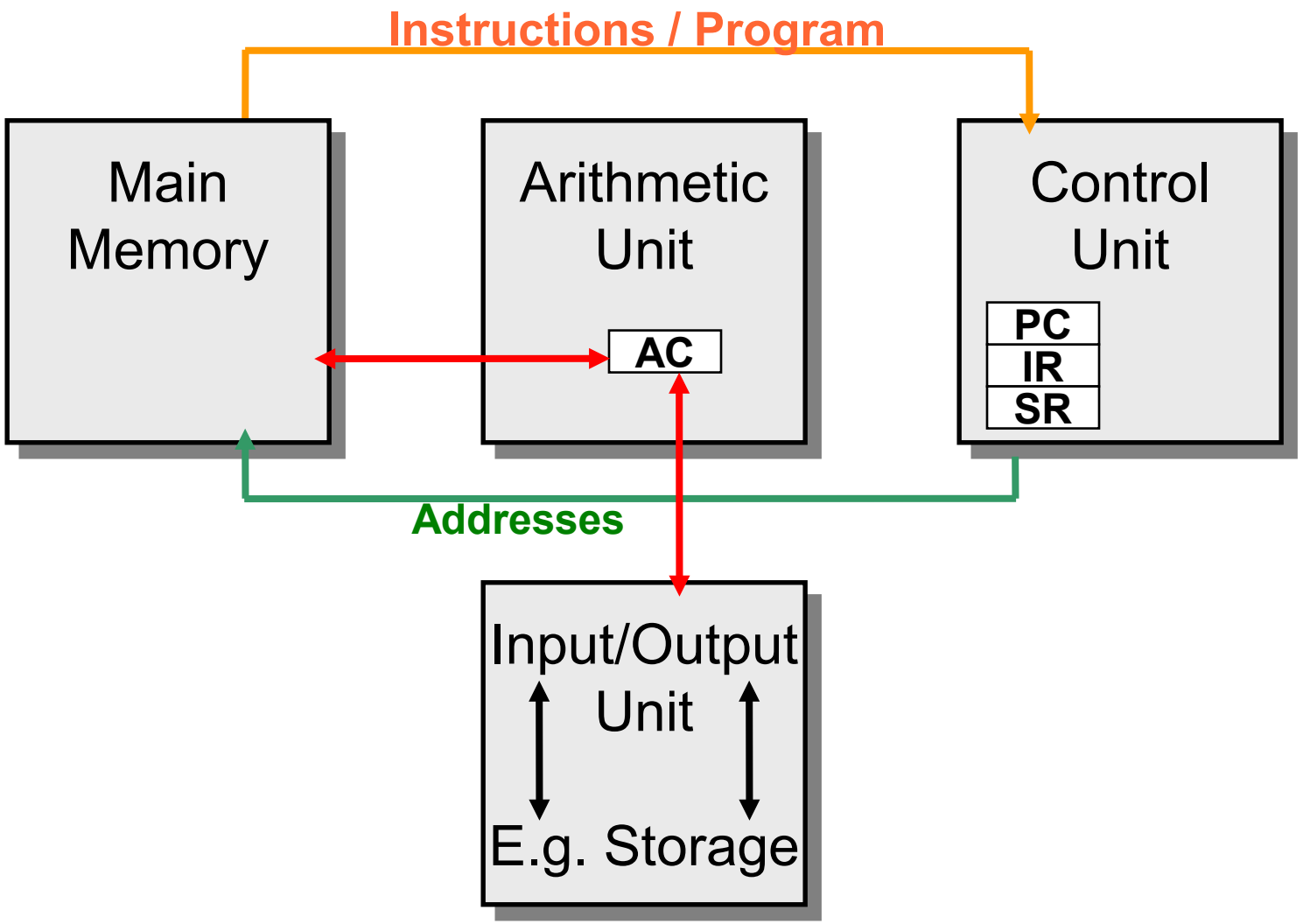
Architecture is an iterative process

- **-- searching the space of possible designs**
- **-- at all levels of computer systems**
- **-- tradeoff**



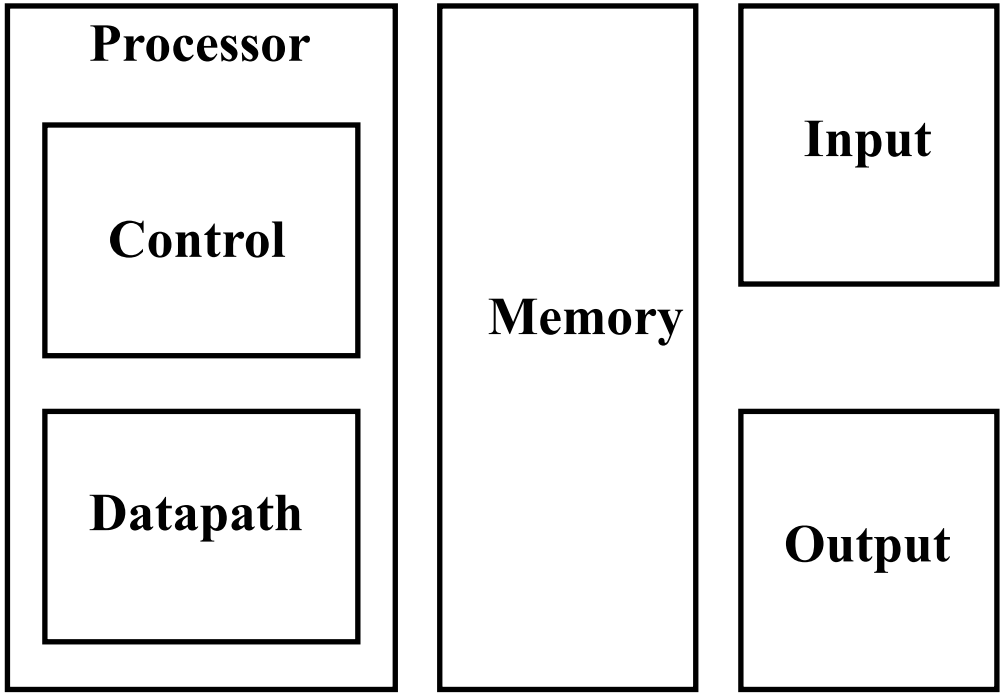


Von Neumann – Overview



The Big Picture

- Since 1946 all computers have had 5 components



Von Neumann – Today

- **Still the dominant architecture in current systems**
  - **Used in all popular systems / chips**
- **Only minor modifications**
  - **Control und Arithmetic unit combined**
    Result: CPU (Central Processing Unit)
  - **New memory paths between memory and I/O**
    Direct Memory Access (DMA)
- **Additions to the concept**
  - **Multiple arithmetic units / Multiple CPUs**
  - **Parallel processing**

Microprocessor Parallelism History

- **4-Bit, 8-Bit Processors**
  - **Intel 4004 (~1971)**
  - **Intel 8008**
- **16-Bit Processors**
  - **Texas Instruments TMS 9900 (~1977)**
  - **Intel 8086**
  - **Zilog Z8000**
  - **Motorola MC68000**
  - **National Semiconductor NS16016**

} (~1978-1980)

Microprocessor Parallelism History

- **16/32-bit Processors**

  (external 16-bit Bus, internal 32 Bit Structure)

  - **Motorola MC68010**
  - **National Semiconductor NS16032**

- **Additional Functionality on the Chip**

  - **Direct Memory Access (DMA) (Intel 80186)**
  - **Virtual memory management (MC68010, Intel 80286)**
  - **Optional Coprocessor (Intel 8086/80286, NS16032)**
  - **Extended Address Space**

Microprocessor Parallelism History

- **32-bit Processors**
  - **CISC Processors**
    - **Motorola MC680x0**
    - **Intel i386 / i486 / Pentium**
    - **National Semiconductor NS32x32**
    - **Concept of a Processor Family**
    - **Binary Compatibility**
    - **Compatible with 16 Bit Processors**
  - **RISC Processors**
    - **Advanced Micro Devices Am29000 (~1987)**
    - **Sun Microsystems SPARC**
    - **MIPS technologies MIPS R2000 / MIPS R3000**

Microprocessor Parallelism History

- **64/32-bit Processors**
  - **SUN Microsystems SuperSPARC**
  - **Motorola 88110**
  - **IBM, Motorola PowerPC 601 (MPC601)**
- **“Modern” Processors**
  - **64-bit Structure**
  - **Internal Parallelism**
    - **Instruction pipelining**
    - **Arithmetic Pipelining**
  - **Instruction and Data Caches**
  - **Advanced Memory and Peripheral Connections**

Bit Level Parallelism (upto mid 80's)

- 4 bit microprocessors replaced by 8 bit, 16 bit, 32 bit etc.
- doubling the width of the datapath reduces the number of cycles required to perform a full 32-bit operation
- mid 80's reap benefits of this kind of parallelism (full 32-bit word operations combined with the use of caches)

Instruction Level Parallelism (mid 80's to mid 90's)

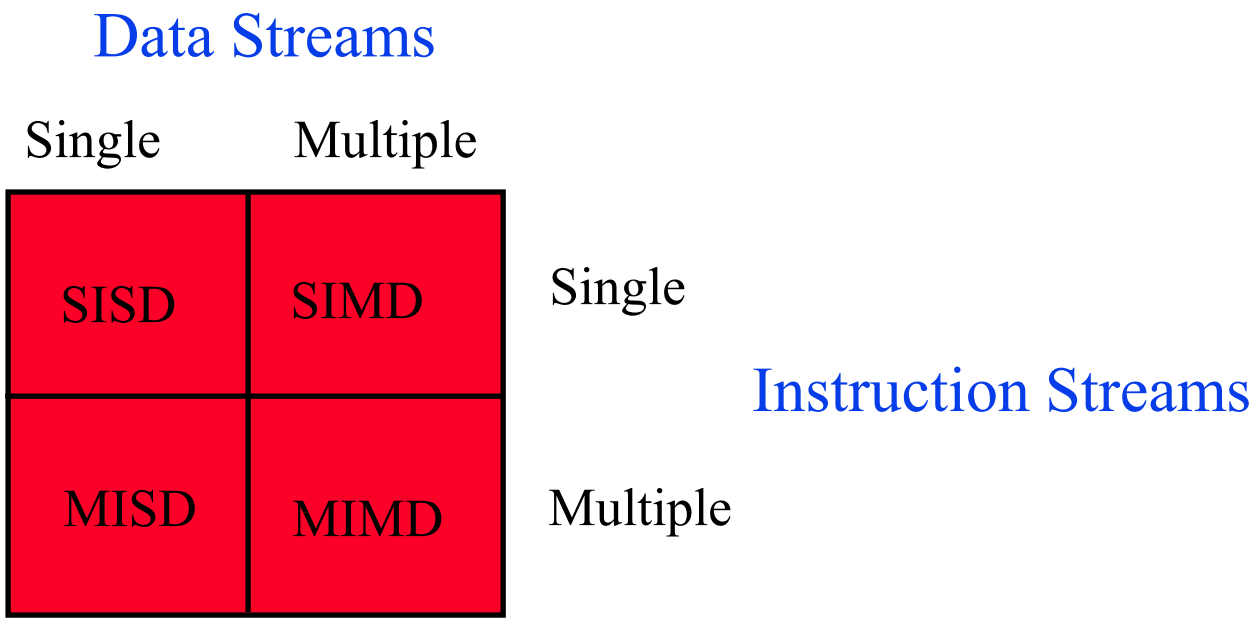
- Basic steps in instruction processing (instruction decode, integer arithmetic, address calculations, could be performed in a single cycle)
- Pipelined instruction processing
- Reduced instruction set (RISC)
- Superscalar execution
- Branch prediction

Thread/Process Level Parallelism (mid 90's to present)

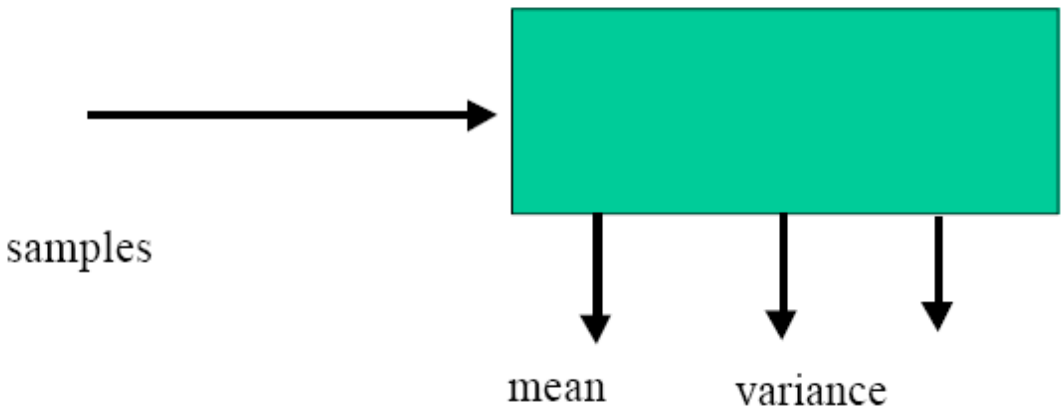
- On average control transfers occur roughly once in five instructions, so exploiting instruction level parallelism at a larger scale is not possible
- Use multiple independent "threads" or processes
- Concurrently running threads, processes

Flynn's Taxonomy

- Classifies computer architectures according to:
  1. Number of instruction streams it can process at a time
  2. Number of data elements on which it can operate simultaneously



MISD



SPMD Model (Single Program Multiple Data)

- Each processor executes the same program asynchronously
- Synchronization takes place only when processors need to exchange data
- SPMD is extension of SIMD (relax synchronized instruction execution)
- SPMD is restriction of MIMD (use only one source/object)

Parallel Processing Terminology

- *Data Parallelism*:

  -model of parallel computing in which a single operation can be applied to all data elements simultaneously

  -amenable to SIMD or SPMD style of computation

- *Control Parallelism*:

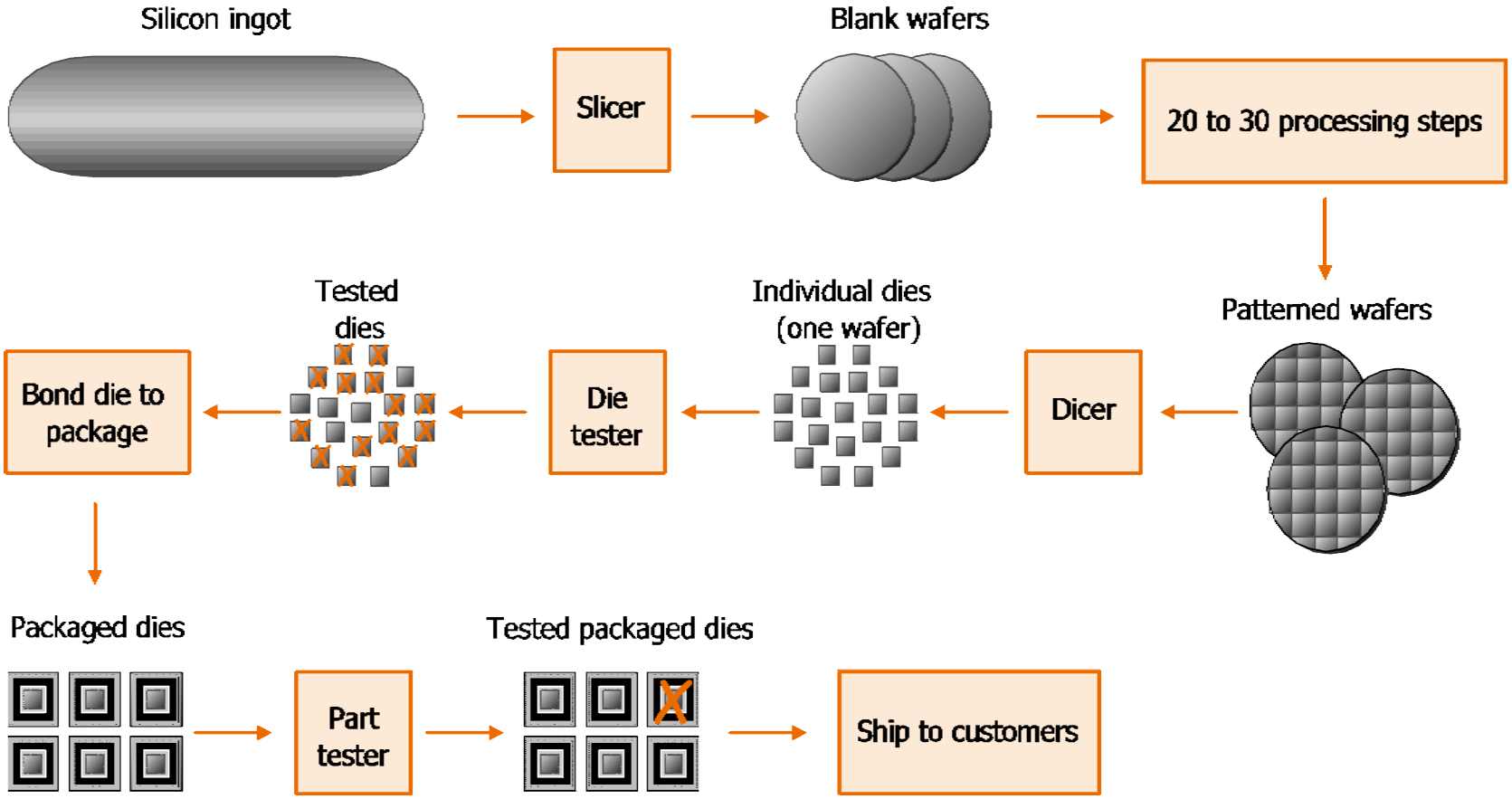
  -many different operations may be executed concurrently

  -require MIMD/SPMD style of computation

Parallel Processing Terminology

- *Scalability:*
  - If the size of problem is increased, number of processors that can be effectively used can be increased (i.e. there is no limit on parallelism).
  - Data parallel algorithms are more scalable than control parallel algorithms
- *Granularity:*
  - fine grain machines: employ massive number of weak processors each with small memory
  - coarse grain machines: smaller number of powerful processors each with large amounts of memory

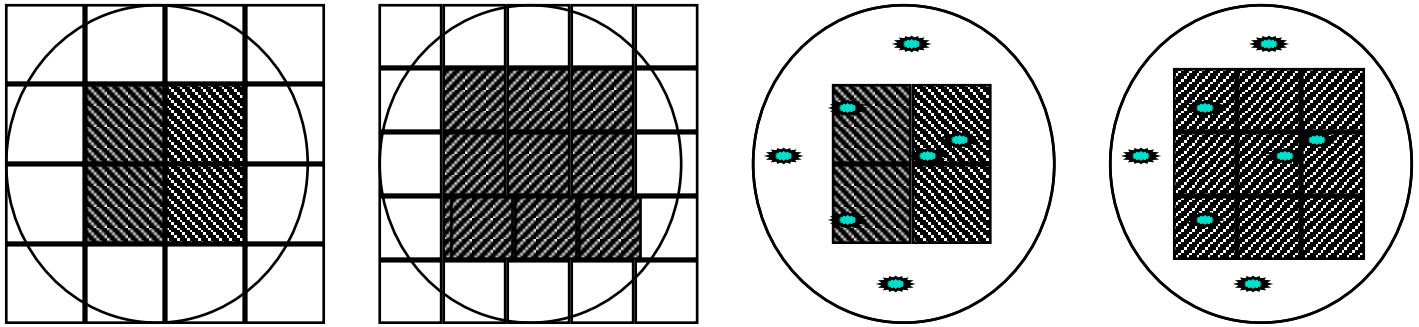
Integrated Circuit



Integrated Circuit Costs

$$\text{Die cost} = \frac{\text{Wafer cost}}{\text{Dies per Wafer} * \text{Die yield}}$$

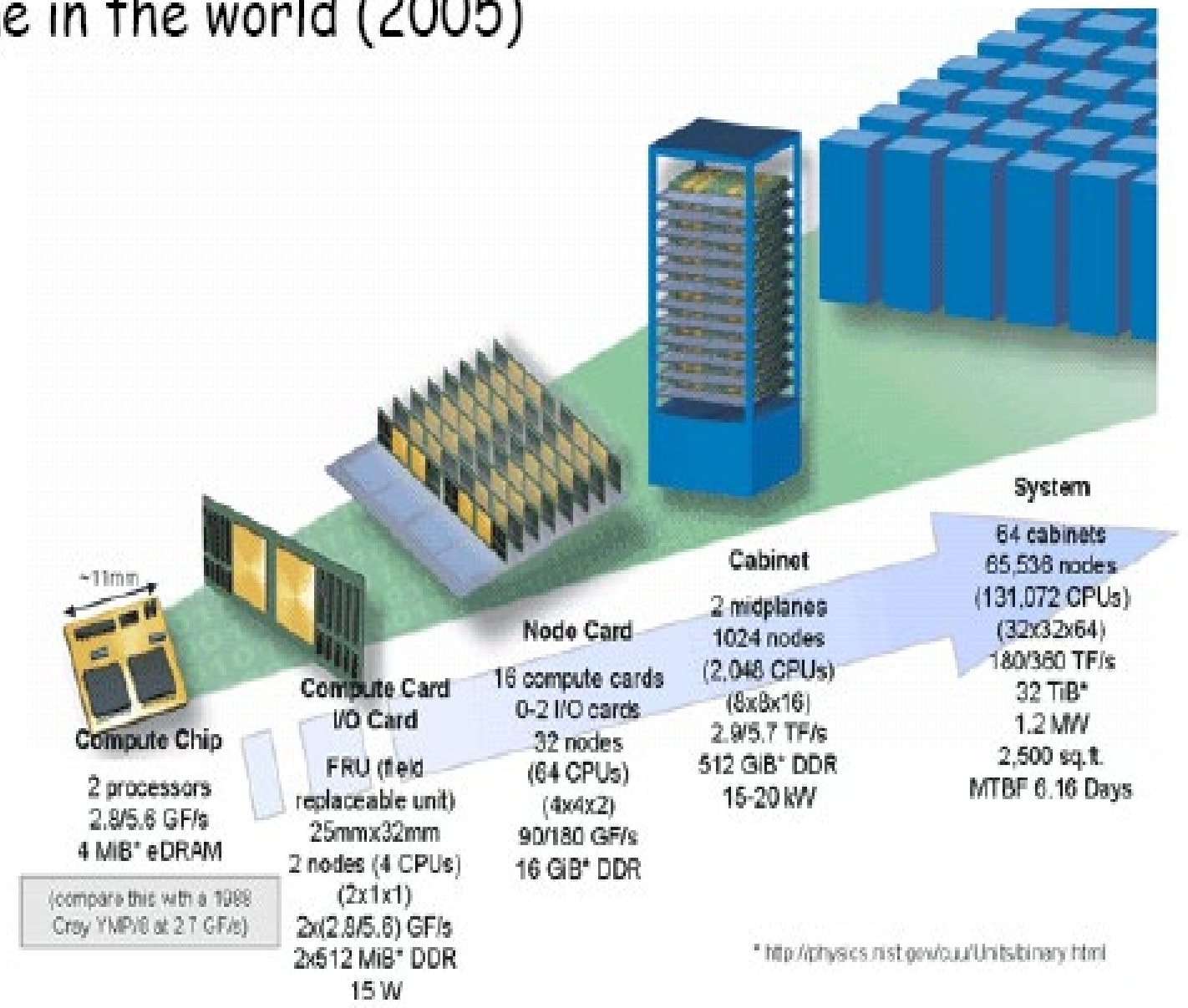
$$\text{Dies per Wafer} = \frac{\pi * (\text{Wafer\_diam} / 2)^2}{\text{Die Area}} - \frac{\pi * \text{Wafer\_diam}}{\sqrt{2 * \text{Die Area}}} - \text{Test dies} \approx \frac{\text{Wafer Area}}{\text{Die Area}}$$



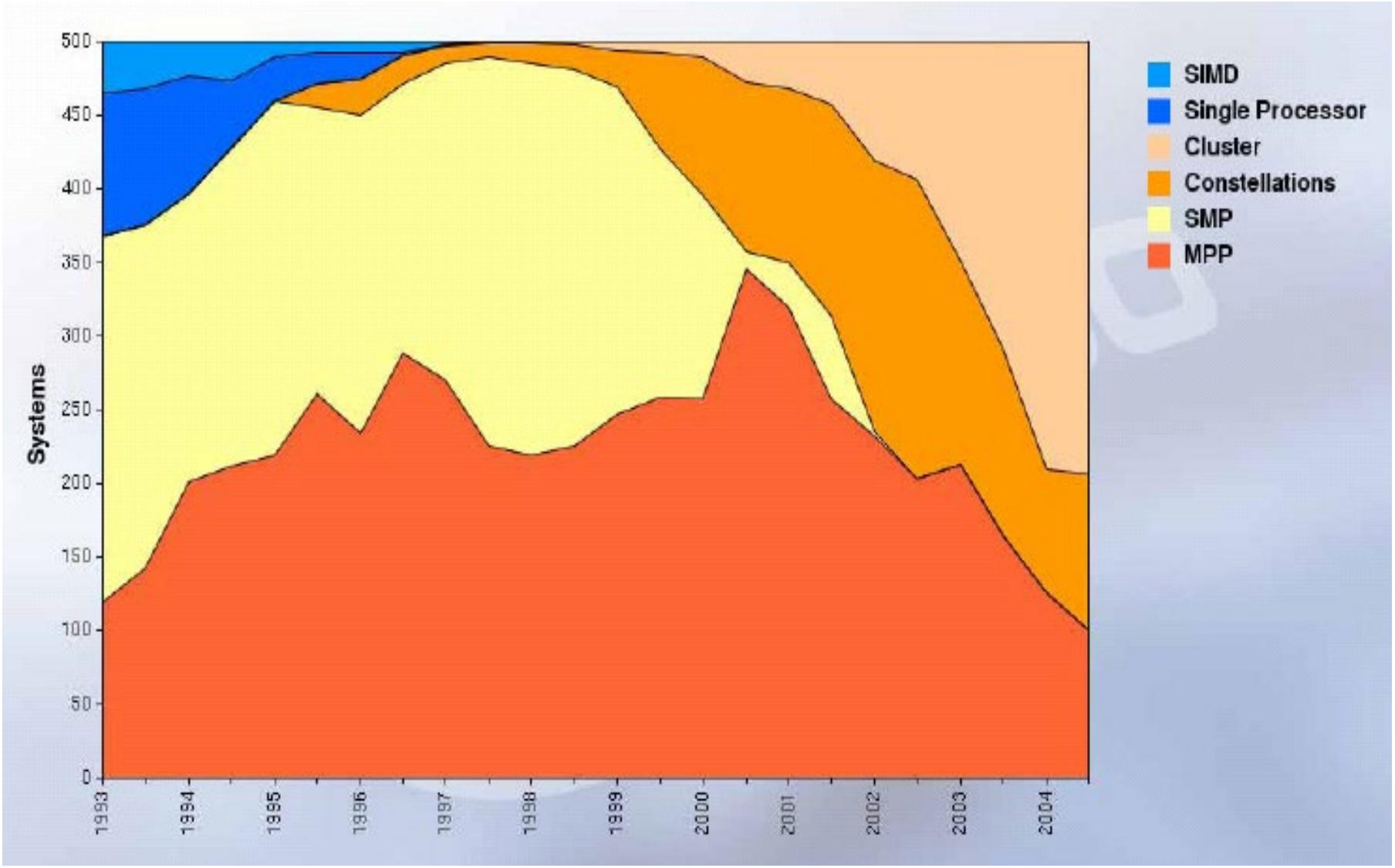
$$\text{Die Yield} = \frac{\text{Wafer yield}}{\left\{ 1 + \frac{\text{Defects\_per\_unit\_area} * \text{Die\_Area}}{\alpha} \right\}^{\alpha}}$$

Die Cost is goes roughly with (die area)^3 or (die area)^4

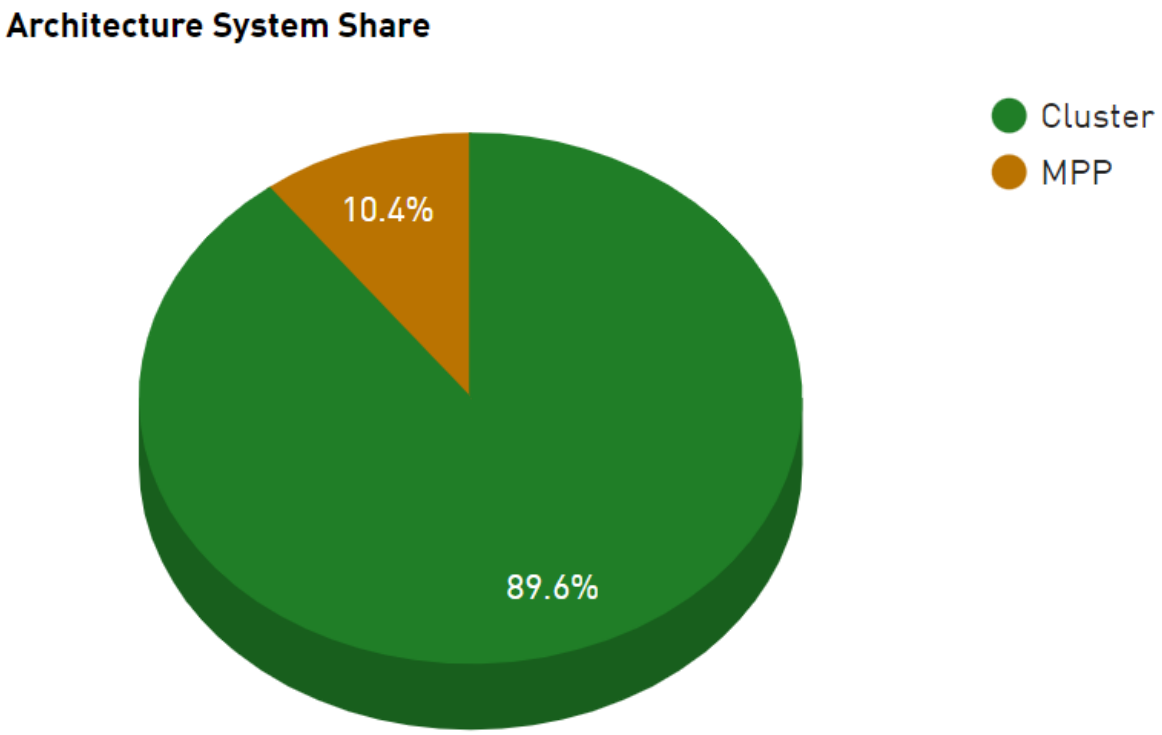
IBM Blue Gene/L - Fastest machine in the world (2005)



Top 500----Architecture/System



Top 500----Architecture/System (Nov. 2022)



Frontier: #1 of Top500 (Nov. 2022)

- “全世界的科学家和工程师将用这些非凡的计算速度解决这个时代的若干最具挑战性的问题，这将是许多人开始探索这个世界的**第一天**。”

Frontier: #1 of Top500 (Nov. 2022)

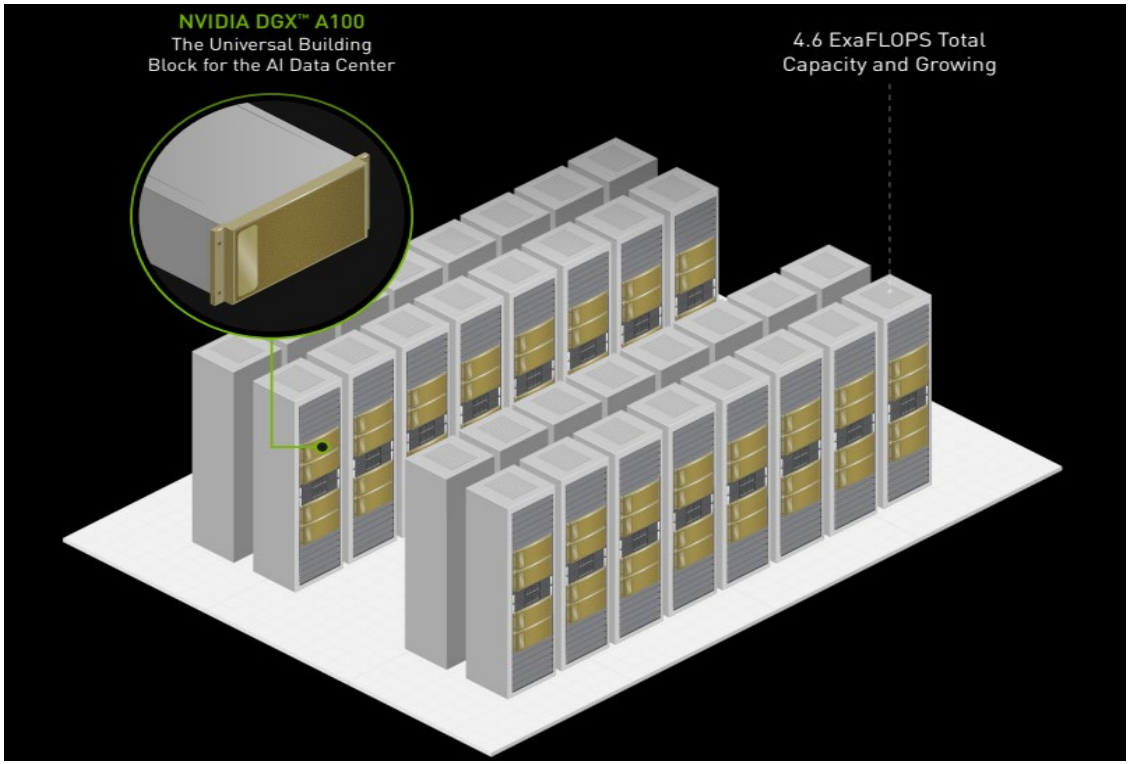
- “一台机器的性能占整个榜单总性能的25%，这是一个令人印象深刻的成就。”——Simon McIntosh-Smith，英国布里斯托大学
- “有了这种水平的计算能力，科学家就不必像以前那样近似计算。对于较旧的计算机，我不得不假设这个项是无关紧要的、那个项是无关紧要的、也许我不需要那个等式。 物理学术语称之为制造‘球形牛(spherical cow)’：将复杂的现象（如牛）转化为高度简化的事物（如球）。有了百亿亿亿次级计算机，科学家可以避免削减这些角落，并模拟一头真正的牛——更接近现实表现的东西。” ——Douglas Kothe，橡树岭计算和计算科学副实验室主任
  - 在气象学中，它可以使飓风预报不那么模糊；在化学中，它可以用不同的分子构型实验，看看哪些可以制造出出色的超导体或药物化合物；
  - 在医学方面，它已经分析了导致 COVID 的病毒 SARS-CoV-2 的所有基因突变——将计算时间从一周缩短到一天——以了解这些调整如何影响病毒传染性；
  - 在天文学中，它模拟了暗物质和暗能量的影响，研究大爆炸后宇宙如何膨胀的初始条件；
  - 在核物理中，它已经对反应堆从启动到整个燃料循环过程的演变进行建模，研究中子如何移动并影响核裂变的链式反应，以及裂变产生的热量如何在系统中移动。

Frontier: #1 of Top500 (Nov. 2022)

The first to achieve an unprecedented level of computing performance known as exascale

Performance	1.6 EF (peak) / 1.1 EF (overall)
Cabinets	74
Processors	9,472 CPUs & 37,888 GPUs (8,730,112 cores total)
Node	1 HPC and AI Optimized 3rd Gen AMD EPYC CPU 4 Purpose Built AMD Instinct 250X GPUs
CPU-GPU Interconnect	AMD Infinity Fabric
System Interconnect	Multiple Slingshot NICs providing 100 GB/s network bandwidth. Slingshot network which provides adaptive routing, congestion management and quality of service.
Storage	in-system storage layer: 75TB/s peak read & 35TB/s peak write
	Orion center-wide file system: 700PB capacity & 5TB/s peak write
Power	21MW (since liquid-cooled system)

DGX SuperPod: Nvidia's supercomputer for AI

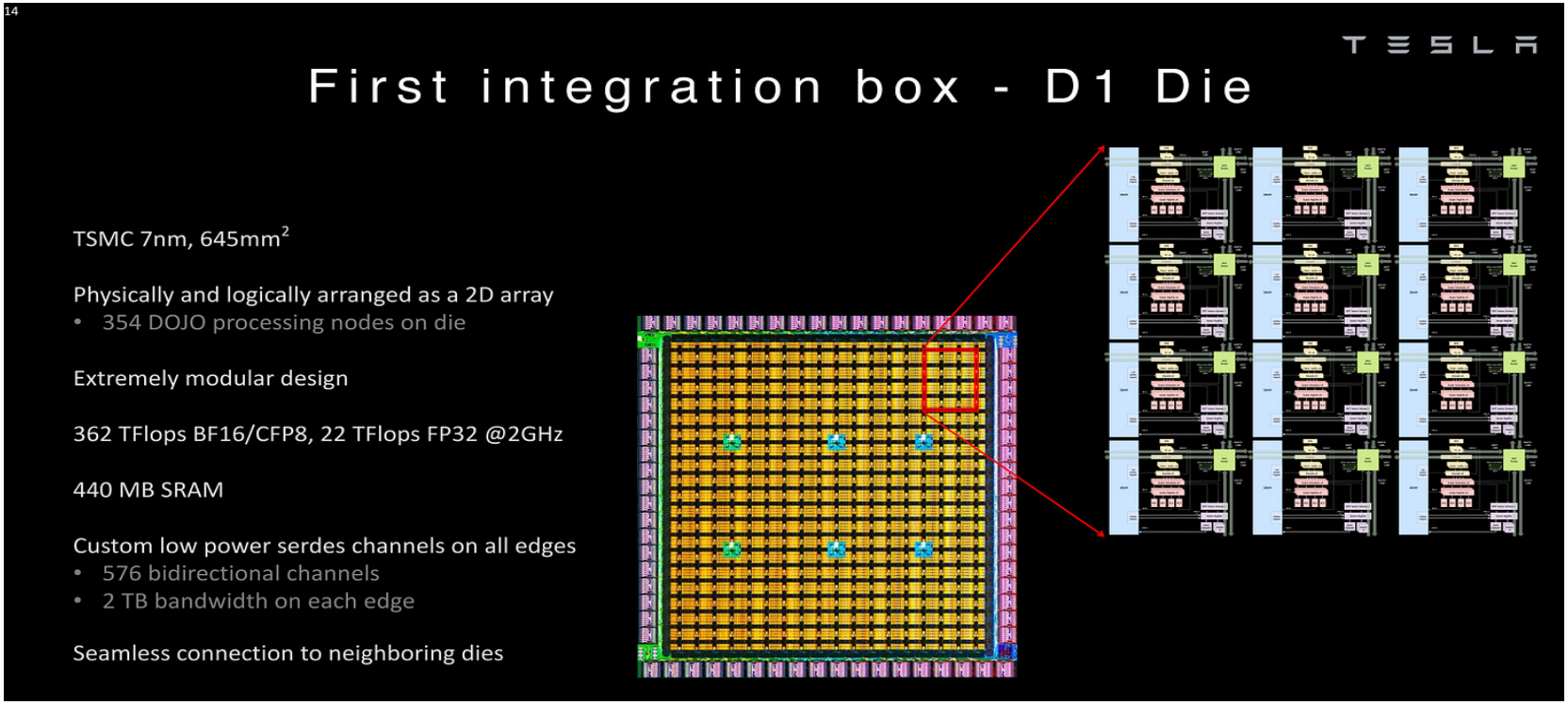


- Purpose-Built for the Unique Demands of AI
- The world's first turnkey AI data center solution
- 27.5 PFLOPs for FP64 (equivalent to #15 in Top500)
- Nvidia is packaging the H100 into its DGX computing modules to replace A100

Dojo: Supercomputer for AI

- Built by Tesla to train its AI deep neural networks and machine learning algorithms to build its self-driving system
- Dojo is designed to replace Nvidia chips for running those AI models in the cars themselves

Dojo D1 chip



- Dojo D1 is a 645 mm2 silicon die, built on TSMC's N7 process. 645mm^2 is very big, as big as massive server chips and almost as big as the biggest GPUs.
- Unlike many AI chips, the smallest irreducible Tesla Dojo D1 process unit is a core, and 354 cores in total arranged in a 2D array.
- 25 D1 chips are arranged in a 5 x 5 square, and are put on a massive silicon interposer to create a 'Dojo Training Tile'

Dojo exapod

Trying to build an exascale supersystem for AI, would be released at Q1, 2023

Performance	1.1 EF for AI specific format (≈ 50+ PF for FP64)
Cabinets	10 (108 PF each)
Processors	42,480 cores
Node	354 cores per D1 chip 25 D1 chips per training tile 6 tiles per tray 2 trays per cabinet (3000 D1 chips total)
System Interconnect	Dojo Interface Processor & Tesla Transport Protocol-over-Ethernet & Z-plane topology multiple routes for data to travel across network, balancing latency and bandwidth and minimizing congestion
Storage	1.3TB High-Speed SRAM & 13TB High Bandwidth DRAM
Power	15 kW per tile

Performance Trends