

Introduction To Processor Design

5/1/26

- Moore's law allowed us to pack more transistors onto the same chip, increasing computation speed.
However, non-ideal effects start to become more prominent in the lower dimensions, leading to the death of Moore's law.
- Now, developments in processor architecture are focusing more on domain-specific accelerators like AI accelerators, photonic computing, IMC, etc.

Textbook: Computer Architecture: A Quantitative Approach, Patterson, Hennessy and Kazanakii (7th | 6th Edn.)

- RISC - Reduced Instruction Set Computer (Open Source)
ARM - Advanced RISC Machine (Proprietary)

* Grading:

Quiz - 10%

Endsem - 30%

Project - 60% → Assignments included (each 5%)

→ Probably 2

- Modern processors have heterogeneous tiling, ie, subprocessors that are optimized for certain applications like GPUs, NPUs, TPUs.

TPU → Systolic Architecture (Google's TPU)

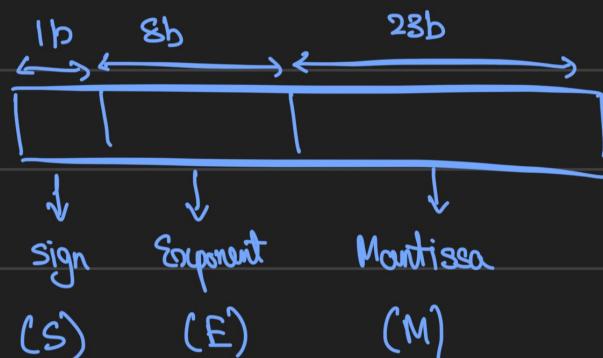
8/1/26

* Computer Abstractions and Technology :-

- Edge Computing - Local Computation. Best for speed and privacy.
- Cloud Computing - Centralized Computation. Best for accessibility and performance.

→ Floating Point Number System :-

- IEEE 754 is a standard used to represent 32 bit floating point numbers.



32 bit - Single prec.
64 bit - Double prec.

- The number is of the form $s(1.M \times 2^E)$
 $\hookrightarrow E$ - biased, ie,
 - > 127 , +ve exponent
 - < 127 , -ve exponent
 - $= 127$, 0 exponent
- To do floating point multiplication,

$$S_1(1.M_1 \times 2^{E_1}) \times S_2(1.M_2 \times 2^{E_2}) = (S_1 S_2) \left(1.M_1 M_2 \times 2^{E_1 + E_2}\right)$$

$S_1 \quad S_2 \quad S_1 S_2$

$S_1 S_2 \rightarrow$	sign	\Rightarrow	+	+	+	}	$\underline{\underline{XOR}}$	$\begin{cases} + = 0 \\ - = 1 \end{math}$
			+	-	-			
			-	+	-			
			-	-	+			

$M_1 M_2 \rightarrow$ Ill to regular multiplication

$E_1 + E_2 \rightarrow$ Done using normal addn. instruction

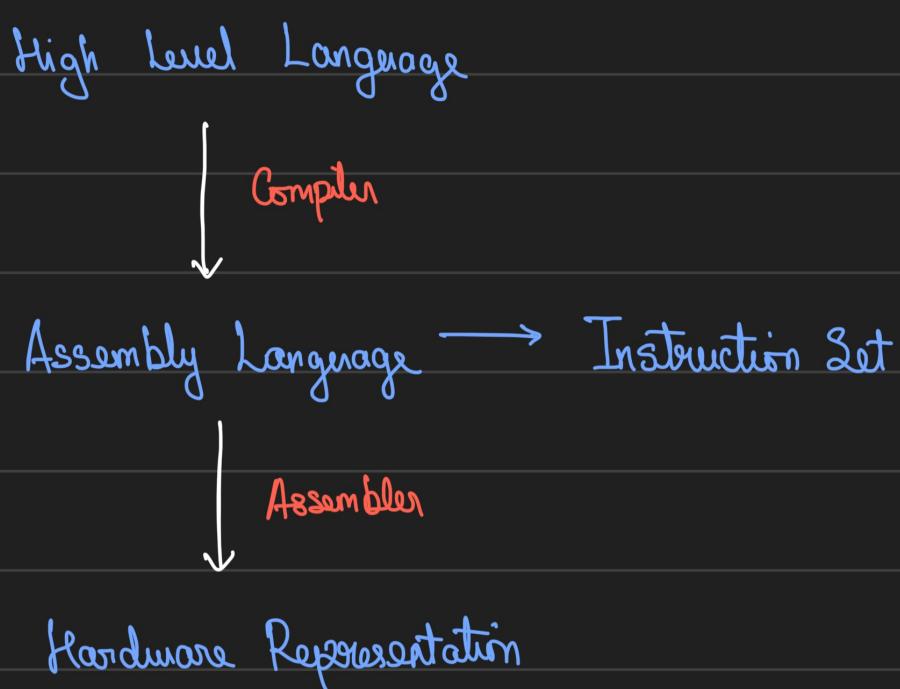
- This type of computation is used since it gives us greater accuracy using lesser amount of bits.

Mini - Assignment : Design a floating-point ALU.

→ Understanding Performance :-

- Algorithms - Determine the number of operations to be performed

- Language, Compiler, Architecture - Determines no. of machine instructions for operation. (Compiler Architecture determines Compiler efficiency)
- Processor & Memory - How fast instructions are executed
- I/O System - Determines how fast I/O operations are executed.
(Interface, Bandwidth, Protocol)



→ Inside The Processor :-

- Datapath - A sequence of operations on data
- Control - Sequencing of datapaths
- Cache - Collection of SRAMs for fast memory access

- Response Time - Time taken to finish an operation
- Throughput - No. of tasks performed by the processor in unit time.
- Performance = $1 / \text{Execution Time}$
- The operation of the CPU is clocked by a digital clock of a certain frequency. Higher frequency \rightarrow More operations.

$$\text{CPU Time} = \text{CPU Clock Cycles} \times \text{Clock Time Period}$$

$$\text{Clock Cycles} = \text{Instruction Count} \times \text{Cycles per Instruction}$$

\hookrightarrow Depends on Compiler, ISA, Program

$$\Rightarrow \text{CPU Time} = \text{Ins. Count} \times \text{CPI} \times T$$

\rightarrow Power Metrics :-

$$\text{Power} = f C V^2$$

f - Clock frequency

C - Capacitive Load

V - Operating Voltage.

12/11/26

Eg: Represent - 5.5 in the floating point number system.

$$-5.5 = -101.1_2$$

$$= \left(-1.011 \times 10^2 \right)_2$$

$$\begin{array}{r} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \left| \begin{array}{r} 129 \\ 64 \\ 22 \\ 16 \\ 8 \\ 4 \\ 2 \end{array} \right.$$

$$\text{Exponent} = 2 + 127 = \underline{\underline{129}} = 10000001$$

$$\text{Mantissa} = 011000\ldots$$

Ques: Express 0.10000011011010100... in Decimal

$$\Rightarrow (1.0110101 \times 10^{10000001})_2$$

$$1.0110101 = 1 + 0.25 + 0.125 + 0.03125 + 0.0078125$$

≈ 1.414

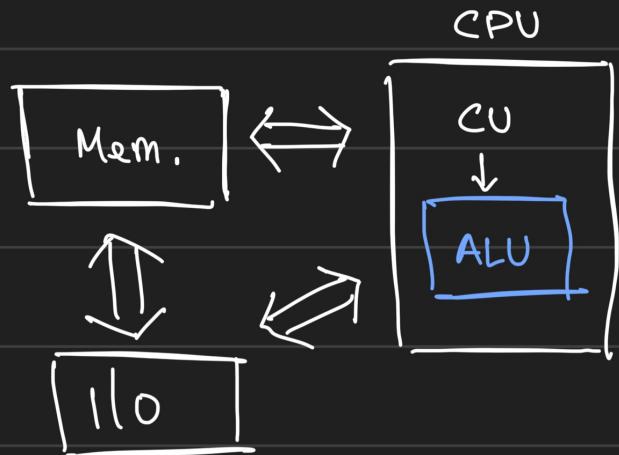
$$10000011_2 = 2^7 + 2^2 + 1 = 128 + 2 + 1 = \underline{\underline{131}}$$

$$\Rightarrow \text{Exp} : |3| - 12\pi = \underline{\underline{4}}$$

$$\Rightarrow Z = 1.414 \times 2^4 = 1.414 \times 16$$

$$= \underline{22.624}$$

→ Von Neumann Architecture :-



- ° Traditional computational model.
- ° Registers: Memory devices placed close to the computational units, to speed up operations. Made using flipflops so expensive to fabricate.