Computer Architecture - Lecture 2

- Decode unit (control unit): change the **instruction** into control **signal**.
- Decode unit takes **[IR]** as an input and change it into **signal**.
- [MAR] and [MDR] are interfaces between the **processor** and **memory**.
- [MAR] → connect to **address bus** that goes to memory.
- [MDR] → connect to **data bus** that goes to memory.
- **<u>Fetch stage example:</u>**

300	Add
	Sub
	..

  - [MAR] = 300
  - Read 300
  - [MDR] → Add
  - Add will be put in [IR]
  - [PC] = 301
- **<u>Control unit:</u>**
  - Hardwired control.
    - Disadvantage: It should be executed at one time.
  - Split the instruction into several small instructions.
  - Micro-programmed control.
    - Advantage: flexible.
  - Ex: Add C, A, B
    - Fetch A
    - Fetch B
    - Add
    - Store in C
- **<u>Basic components of a processor:</u>**
  - Control unit [CU].
  - [ALU]
  - Register file:
    - Special registers: eg: [MAR], [MDR], [IR], [PC]
    - General purpose register (for temporary storage of intermediate results) eg: Accumulator. (to store the result of the computation to use it again)
- **<u>Instruction types:</u>**

OP code	Operands

  - OP code: operation, eg: add (code of operation)
  - Operands: A, B, C
  - They are written in binary.

- **Instructions:**
    - Zero operand instruction (instruction without operand)
    - One operand instruction
    - Two operand instruction (add A and B, then answer will be in A)
    - Three operand instruction (add A and B, then answer will be in C)
- Semi operand → Ex: Add X
    - This will add the X to the accumulator.
- **Performance of computers:**
    - Speed
    - Storage (main memory)
- **Speed of computer system is measured by:**
    - Clock speed [HZ: cycle/sec]
    - Instruction per second [MIPS: million instruction per second]
    - Flops → floating operations/sec (أكتر طريقة بتأخذ وقت)
        - Ex: $3*10^5 + 5*10^3 = 3*10^5 + 0.05*10^5 = 3.05*10^5$
- To compare the performance of computer systems, we use "benchmark programs".
- Benchmark → statistical indication on speed of processor by using common operation.
- If the benchmark program contains 3 classes of instructions: class A, class B and class C where:
    - Class A takes one clock cycle.
    - Class B takes two clock cycle.
    - Class C takes three clock cycle.

  If the program has → 10,000 instructions of class A.
  → 20,000 instructions of class B.
  → 15,000 instructions of class C.

  And the clock speed of the computer is 400 MHZ.

  Find:

  1. CPU time of the program execution.

  Number of clock cycles = 10,000 + 2*20,000 + 3*15,000 =

  $T_{Clock} = \frac{1}{f_{clock}} = \frac{1}{400 \times 10^6} = 0.25 * 10^{-8} = 2.5 * 10^{-9} = 2.5$ nsec

  Execution time = number of clock cycles * T_{Clock}

  2. Compute the average CPI (clock per instruction).

  Average CPI = $\frac{10000 \times 1 + 2000 \times 2 + 15000 \times 3}{10000 + 20000 + 15000}$

  Average duration = Average CPI * T
- Speed up: A measurement to compare two systems A, B.
- Speed up of A relative to B = $\frac{Execution\ of\ B}{Execution\ of\ A} = \frac{Speed\ of\ A}{Speed\ of\ B}$
- In parallel systems, we compare the **parallel system** with a **serial system** having the same architecture.

- Speedup = $\frac{Serial\ exeution\ time}{Parallel\ execution\ time}$ = $\frac{Speed\ of\ parallel}{Speed\ of\ serial}$
    - If > 1 يبقي سرعة كويسة
    - If < 1
- الوقت عكس السرعة
- **<u>Amdahls's law:</u>**
- Given a benchmark program executed serially in time T_s, if p is a portion of the program that can run in parallel over n processors (and thus 1-p run serially) What is speedup?
    - Serial time = T_s
    - Parallel time = (1-p) T_s + $\frac{P\ \mathrm{Ts}}{n}$
    - Speedup = $\frac{\mathrm{Ts}}{(1-\mathrm{p})\mathrm{Ts} + \frac{P\ \mathrm{Ts}}{n}}$ = $\frac{1}{1-p+\frac{P}{n}}$ = $\frac{1}{1-\left(1-\frac{1}{n}\right)p}$ = $\frac{1}{1-\left(\frac{n-1}{n}\right)p}$
    - $\lim\limits_{n\to\infty}$ **speedup =** $\frac{\mathbf{1}}{\mathbf{1-p}}$
    - Ex: IF p = 0.9

      Max speedup = $\frac{1}{1-0.9}$ = 10 (as $n \to \infty$)
- To compute $n_{1/2}$(number of processors needed to get half the max speedup)

$$\frac{1}{1-p+\frac{P}{n_{1/2}}} = \frac{1}{2(1-p)}$$

$$1-p + \frac{P}{n_{1/2}} = 2-2p$$

$$\frac{P}{n_{1/2}} = 1-p$$

$$n_{1/2} = \frac{P}{1-p}\ (Rule)$$

In the previous example: $n_{1/2}$ = $\frac{0.9}{1-0.9}$ = 9 processors.