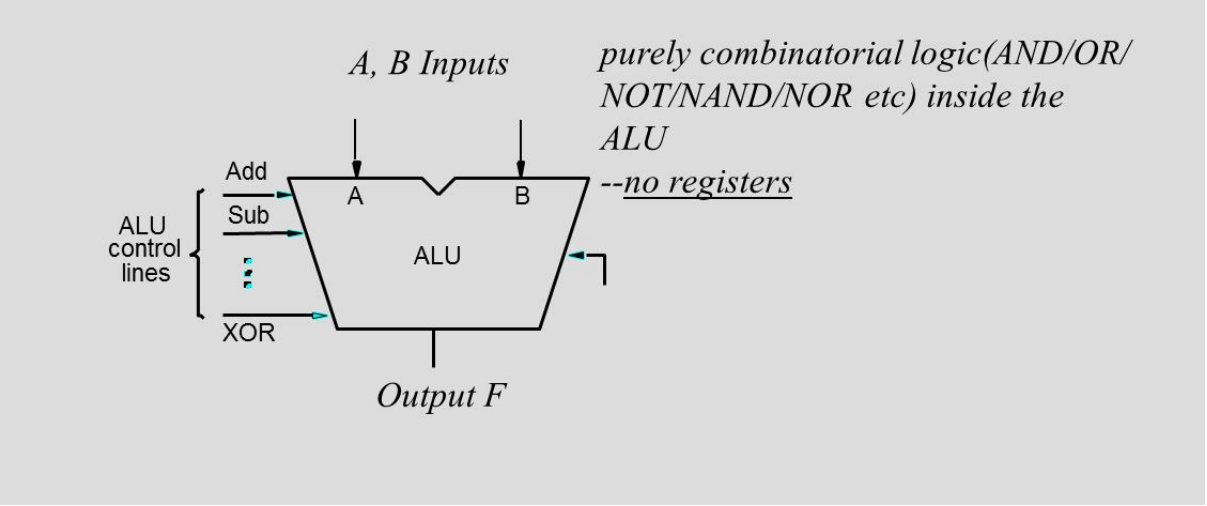
CPU architecture

Arthimetic Logic UNIT(ALU)

- ALu is a purely combinatorial Device:
  - It has no memory or internal storage.
- It has 2 input vectors:
  - These may be called the **A - and B- vector** or the **R- and S- vector**.
  - The inputs are as wide as the registers/system bus(16 32 bits….)
- the ALU has **1 output vector**
  - Usually denoted as **F**

ALU STRUCTURE



- **ALU** must have only one **input** connection from the **bus**
- The other **input** must be stored in a holding register called **Y register**
- A **multiplexer** selects among **register Y** and 4 depending upon select line
- One **operand of a two-operand** instruction must be placed into the **Y register** before the other **operand** must be placed onto the **bus.**
- There must be an ouput **register Z** which collects the **output** of the **ALU** at the end of each cycle.
- This way there can be
  - One operand in the **Y register**
  - One operand on the **bus**
  - The result stored in the **Z register.**
- **Example: R3 = R1 + R2**
  - 1. Place the contents of register **R1 into the Y** register in the **first clock cycle.**
  - 2. Place the contents of register **R2 onto the bus** in the **second clock cycle.** Both inputs to the ALU are now **valid. Select register Y**, and assert the ALU command **F = A + B**

    - 3. **In the third clock cycle**, **Z register** has latched the output of the ALU. Thus the contents of the **Z register** can be copied into register **R3.**
- Clock cycle 1. Y = R1
- Clock cycle 2. Z = R1 + R2
- Clock cycle 3. R3 = Z

- Inputs of the **ALU**
    - Input **B** is tied to the bus.
    - Input **A** is tied to the ouput of the **multiplexer.**
- Output of the **ALU**
    - Tied to the input of the **Z** register.
- **Z** register
    - Input tied to the ouput of the **ALU**
    - Output tied to the **bus.**
    - **Rin loads from bus Zin loads from ALU OUTPUT**