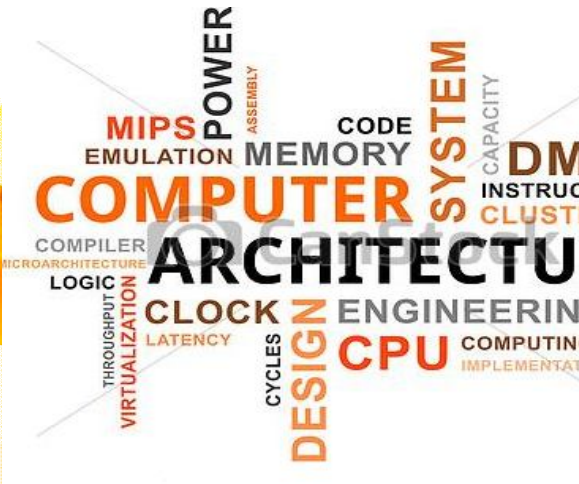
# UNIT III
# PROCESSOR AND PIPELINING

Fundamental concepts – **Execution of a complete instruction** – **Multiple bus organization** – Hardwired control – Micro programmed control – Pipelining: Basic concepts – Data hazards – Instruction hazards – Influence on Instruction sets – Data path and control consideration.



# Recap the previous Class

# Execution of a Complete Instruction

- Add (R3), R1
- Fetch the instruction
- Fetch the first operand (the contents of the memory location pointed to by R3)
- Perform the addition
- Load the result into R1

# Execution of a Complete Instruction

Add (R3), R1

| Step | Action |
|---|---|
| 1 | $PC_{out}$, $MAR_{in}$, Read, Select4,Add, $Z_{in}$ |
| 2 | $Z_{out}$, $PC_{in}$, $Y_{in}$, WMF C |
| 3 | $MDR_{out}$, $IR_{in}$ |
| 4 | $R3_{out}$, $MAR_{in}$, Read |
| 5 | $R1_{out}$, $Y_{in}$, WMF C |
| 6 | $MDR_{out}$, SelectY, Add, $Z_{in}$ |
| 7 | $Z_{out}$, $R1_{in}$, End |

Figure Control sequence for execution of the instruction Add (R3),R1.

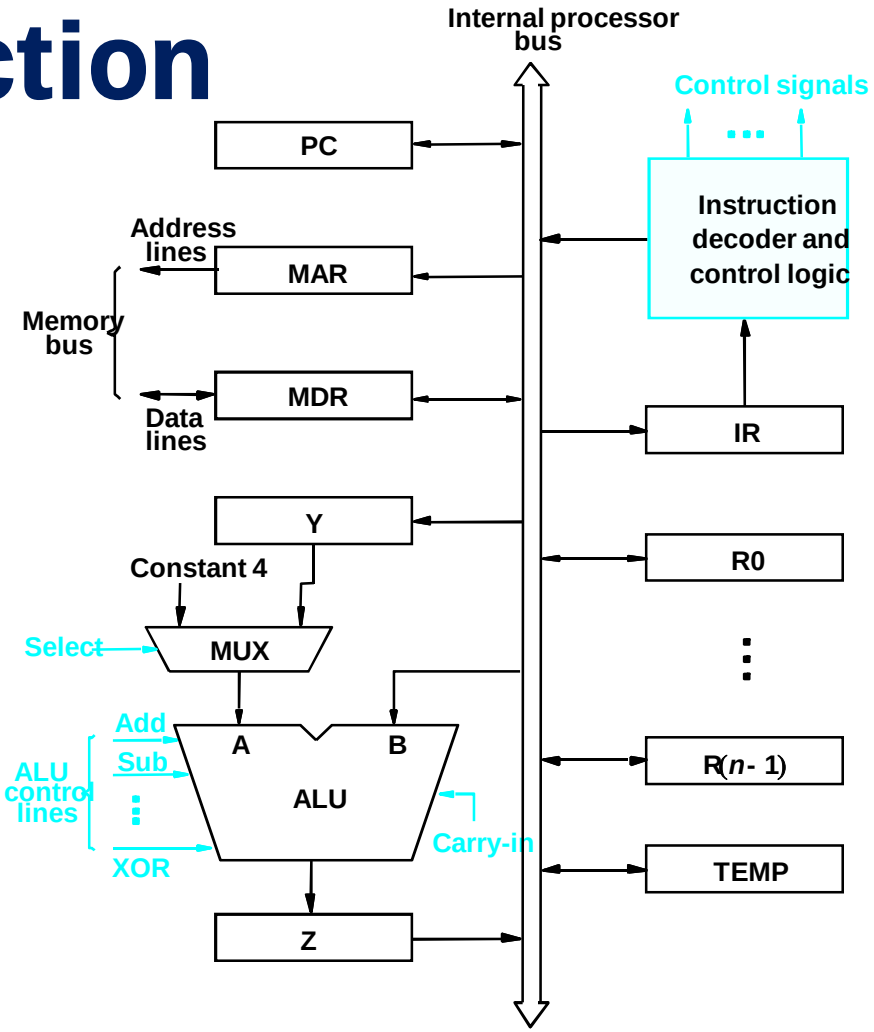



Figure 7.1. Single-bus organization of the datapath inside a

# Execution of Branch Instructions

- A branch instruction replaces the contents of PC with the **branch target address**, which is usually obtained by adding an offset X given in the branch instruction.
- The offset X is usually the difference between the branch target address and the address immediately following the branch instruction.
- UnConditional branch

# Execution of Branch Instructions

| Step | Action |
|---|---|
| 1 | $PC_{out}$ , $MAR_{in}$ , Read, Select4,Add, $Z_{in}$ |
| 2 | $Z_{out}$, $PC_{in}$ , $Y_{in}$, WMFC |
| 3 | $MDR_{out}$ , $IR_{in}$ |
| 4 | Offset-field-of-$IR_{out}$ , Add, $Z_{in}$ |
| 5 | $Z_{out}$, $PC_{in}$ , End |

Figure: Control sequence for an unconditional branch instruction.

# Multiple-Bus Organization

- Allow the contents of **two different registers to be accessed** simultaneously and have their contents placed on buses A and B.
- Allow the data on bus C to be loaded into a third register during the same clock cycle.
- Incrementer unit.
- ALU simply passes one of its two input operands unmodified to bus C

→ control signal: R=A or R=B

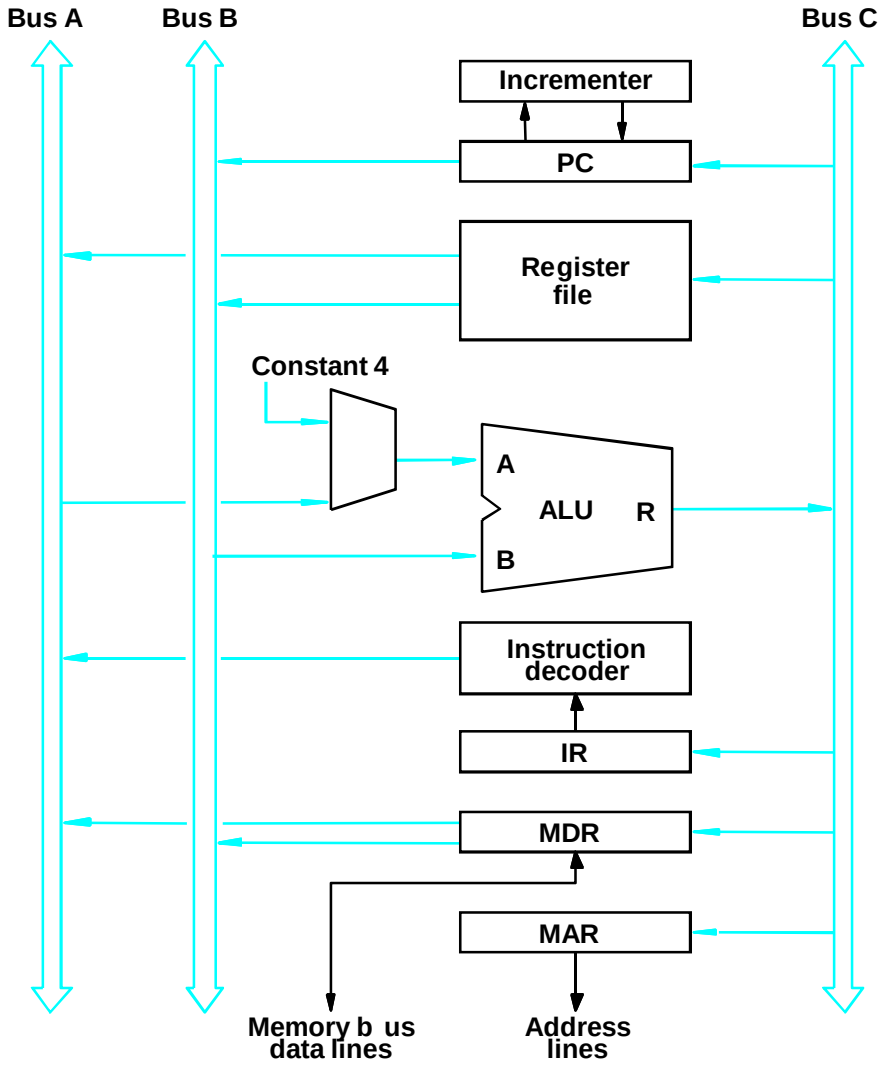



Figure : Three-bus organization of the datapath.

- General purpose registers are combined into a single block called registers.
- 3 ports,2 output ports –access two different registers and have their contents on buses A and B
- Third port allows data on bus c during same clock cycle.
- Bus A & B are used to transfer the source operands to A & B inputs of the ALU.
- ALU operation is performed.
- The result is transferred to the destination over the bus C.

- ALU may  simply pass one of its 2 input operands unmodified to bus C.
- The ALU control signals for such an operation R=A or R=B.
- Incrementer unit is used to increment the PC by 4.
- Using the incrementer eliminates the need to add the constant value 4 to the PC using the main ALU.
- The source for the constant 4 at the ALU input multiplexer can be used to increment other address such as load multiple & store multiple

# Multiple-Bus Organization

- Add R4, R5, R6

| Step | Action |
|---|---|
| 1 | $PC_{out}$, R=B, $MAR_{in}$, Read, IncPC |
| 2 | WMFC |
| 3 | $MDR_{outB}$, R=B, $IR_{in}$ |
| 4 | $R4_{outA}$, $R5_{outB}$, SelectA, Add, $R6_{in}$, End |

Figure :. Control sequence for the instruction. Add R4,R5,R6, for the three-bus organization.

- **Step 1**:The contents of PC are passed through the ALU using R=B control signal & loaded into MAR to start a memory read operation

  At the same time PC is incrementer by 4

- **Step 2**:The processor waits for MFC
- **Step 3**: Loads the data ,received into MDR ,then transfers them to IR.
- **Step 4:** The execution phase of the instruction requires only one control step to complete.

# TEXT BOOK

Carl Hamacher, Zvonko Vranesic and Safwat Zaky, “Computer Organization”, McGraw-Hill, 6th Edition 2012.

## REFERENCES


1. David A. Patterson and John L. Hennessey, “Computer organization and design”, MorganKauffman ,Elsevier, 5th edition, 2014.
2. William Stallings, “Computer Organization and Architecture designing for Performance”, Pearson Education 8th Edition, 2010
3. John P.Hayes, “Computer Architecture and Organization”, McGraw Hill, 3rd Edition, 2002
4. M. Morris R. Mano “Computer System Architecture” 3rd Edition 2007
5. David A. Patterson “Computer Architecture: A Quantitative Approach”, Morgan Kaufmann; 5th edition 2011


# THANK YOU