

**10220CS410001**  
**Computer Architecture**

# **Tutorial 2**

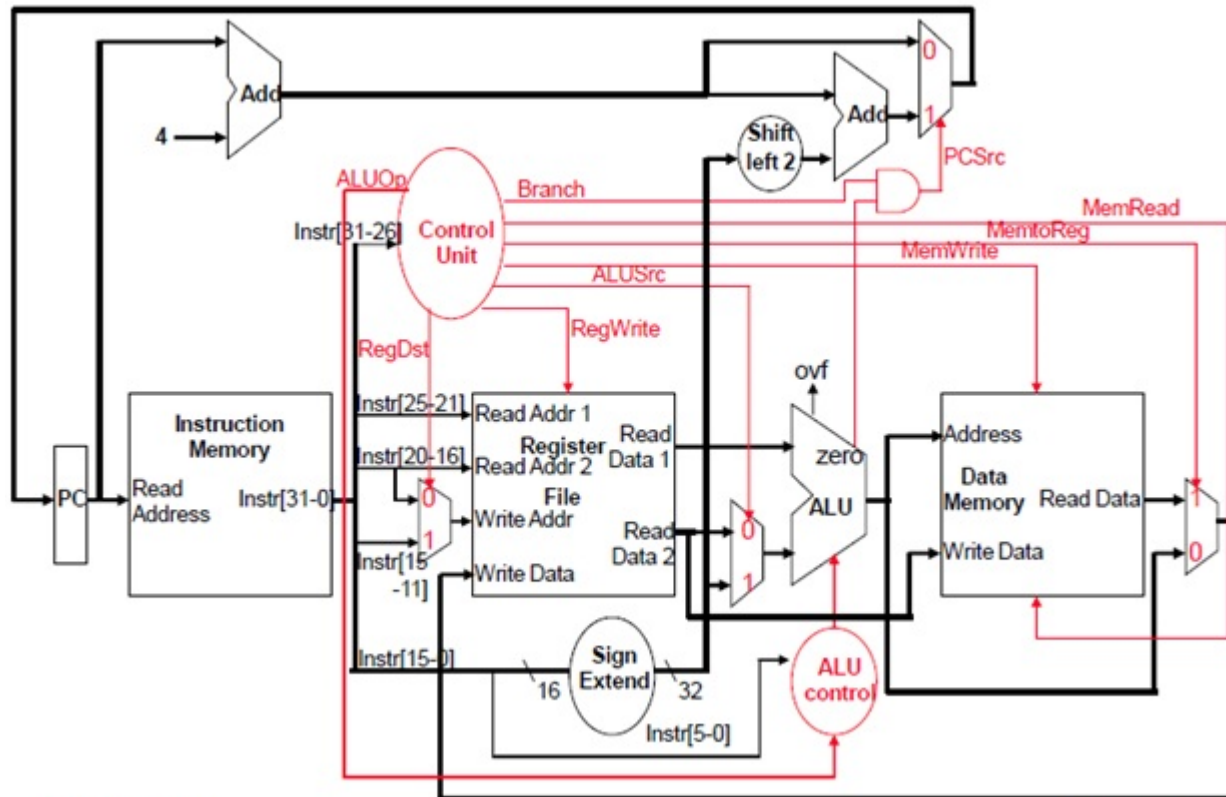
2014/04/08

1

# Outline

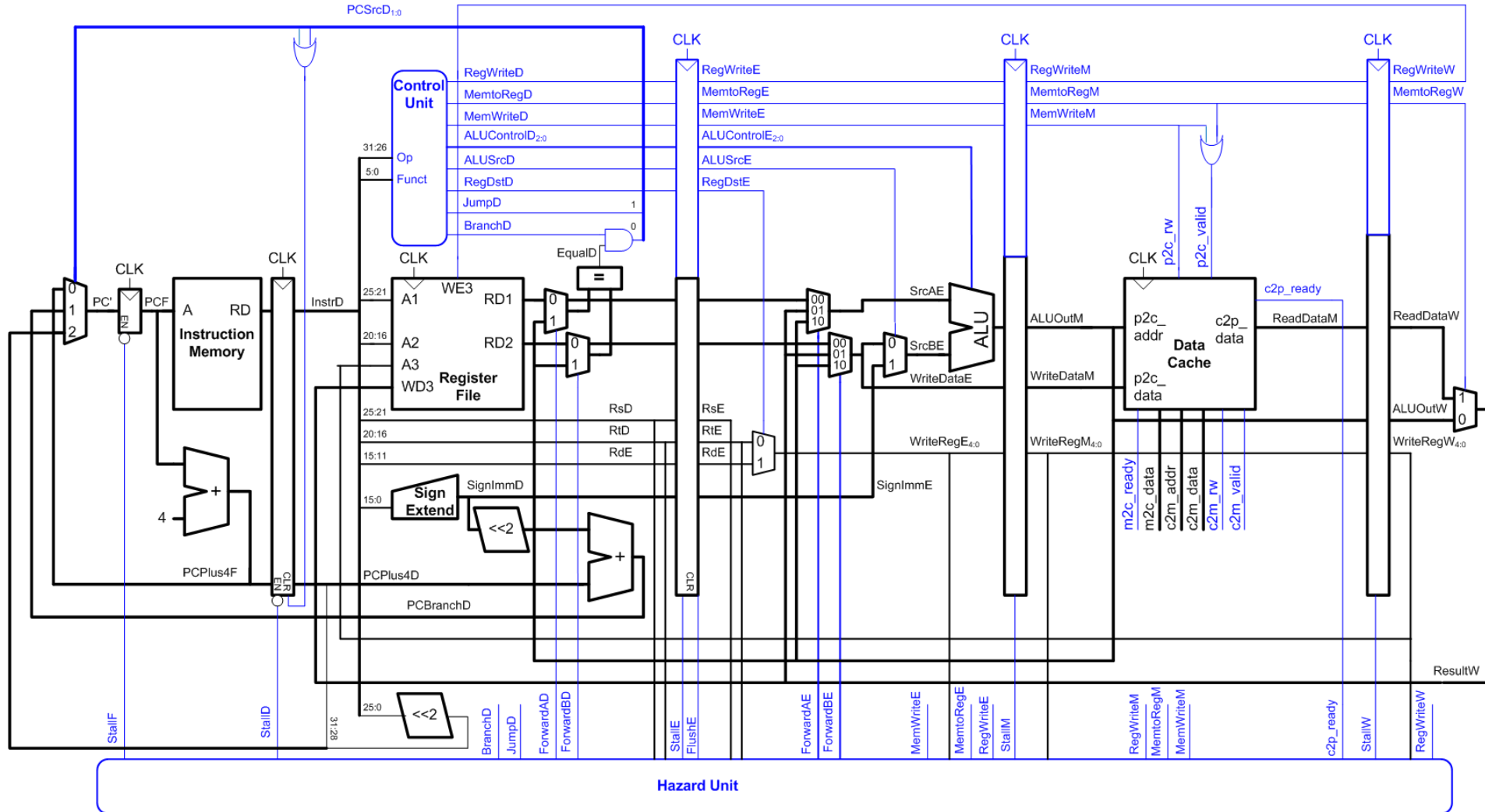
- **Project2-Pipeline**
- Linux utility-screen

# Single Cycle Data Path in project1



- Low design complexity
- Low ILP(instruction-level parallelism)

# Pipeline datapath in project2



- Higher design complexity
- High ILP(instruction-level parallelism)

# 5 Stage in Pipeline

- IF
    - instruction fetch
    - PC+4
  - ID
    - Read Register File
    - Sign extension
    - Control unit decode
    - Hazard detect, Branch/Jump detect
- 
- EX
    - ALU / Branch
  - DM
    - Access data memory
  - WB
    - Write value to Register File

Forwarding Unit
EX/DM to ID
EX/DM to EX
DM/WB to EX

↑ independent of instruction type

---

↓ dependent of instruction type

↓ R-type, I-type, J-type

# snapshot.rpt content

- When complete all instruction in each cycle, please output:
  1. value of all register
  2. MIPS instruction in each stage, and include
    - a. whether stalled in next cycle
    - b. whether forwarding in this cycle

# error\_dump.rpt content

- Note: proper order of simulating 5 stage  
WB → DM → EX → ID → IF  
➤ To avoid re-execution of some stages
- When multiple error occurs, detect the set of error in the following order.
  1. Write To Register \$0
  2. D-Memory Address Overflow
  3. D-Memory Miss Align Error
  4. Number Overflow
  5. I-Memory Address Overflow
  6. I-Memory Miss Align Error

```
lw $2, 0($3)
or $3, $1, $4
beq $2, $3, some_line
and $1, $1, $0
```

	IF	ID	EX	DM	WB
cycle 0	lw	nop	nop	nop	nop
cycle 1	or	lw	nop	nop	nop
cycle 2	beq	or	lw	nop	nop
cycle 3	and	beq	or	lw	nop → Stall detected
cycle 4	and	beq	nop	or	lw → Forwarding detected

cycle 3

...(content of registers)

IF: 0x00200824 to\_be\_stalled

ID: BEQ to\_be\_stalled

EX: OR

DM: LW

WB: NOP

cycle 4

...(content of registers)

IF: 0x00200824

ID: BEQ fwd\_EX-DM\_rt\_\$3

EX: NOP

DM: OR

WB: LW

# Outline

- Project2-Pipeline
- **Linux utility-screen**



# What is Screen

- A full-screen window manager
- One terminal/console → only one interactive shell  
One screen → many interactive shell
- Screen instruction:

screen	create a screen session
Command parameter	
-ls	List all executing session
-r [pid.tty.host]	Reconnect certain session
....	

## In Screen Session

Ctrl+a , c	Create window
Ctrl+a , [0-9]	To window [0-9]
Ctrl+a , n	To next window
Ctrl+a , p	To previous window
Ctrl+a , w	List all window
Ctrl+a , k	Close current window
Ctrl+a , d	Detach screen
Ctrl+a , Esc	Scrollbar, copy mode
exit	Close current session
Ctrl+a , ?	Instruction guideline
Ctrl+a , \	Close all window