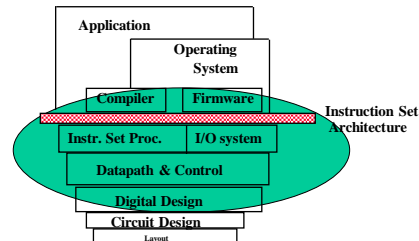
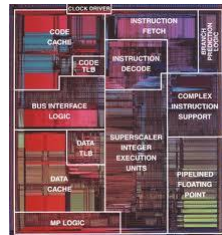


# CS/SE 3340

## Computer Architecture



### Pipelined Datapath and Control

*Adapted from slides by Profs. D. Patterson, J. Hennessey and M. Irwin*

## Questions

1. *How* does a MIPS pipelined datapath look like, and *why* hazards happen in pipelined datapath?
2. *What/Why* of pipeline registers?
3. *How* does a pipelined datapath operate?
4. *What* are the two types of pipelined datapath diagrams?
5. *How* control signals are generated for a pipelined datapath?



Arlington GM plant:

*“Approximately 1,200 vehicles are produced daily –3/2017”*

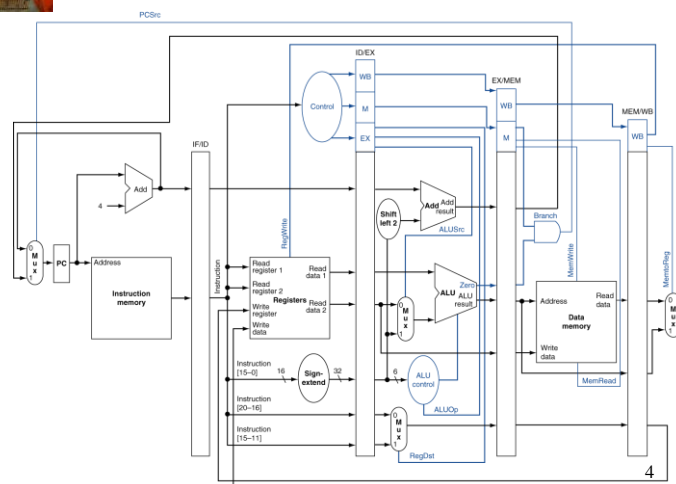
**one SUV every ~72 seconds!!!**

[http://media.gm.com/media/us/en/gm/company\\_info/facilities/assembly/arlington.html](http://media.gm.com/media/us/en/gm/company_info/facilities/assembly/arlington.html)

*“In October, the plant set a production record, building 31,982 SUVs. It expected total output to exceed 300,000 vehicles in 2015”*

**one SUV every ~85 seconds!!!**

<http://www.gosanangelo.com/business/expanded-arlington-gm-plant-rolls-with-suv-sales-28d893d4-dd1a-1-ddc-e053-0100007ffc40-364707301.html>

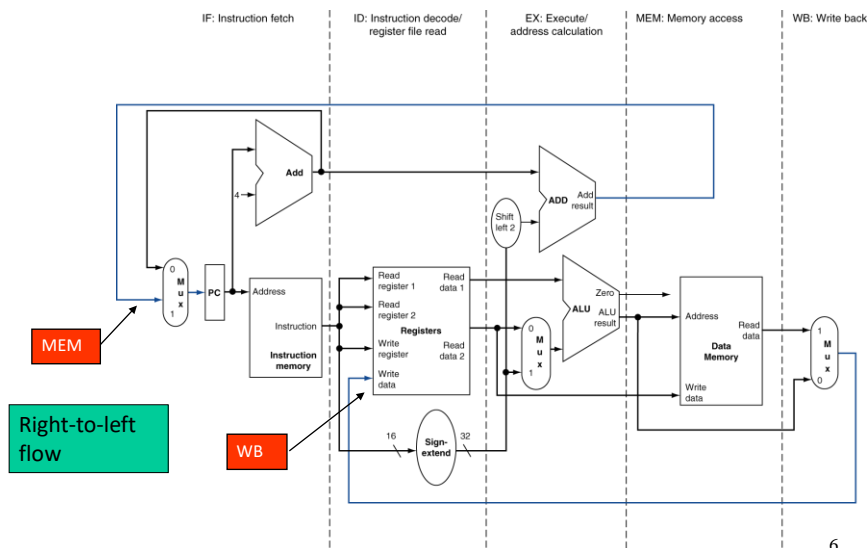


## Pipeline Recap

- Pipelining improves performance by increasing instruction throughput
  - Executes multiple instructions in parallel
  - Each instruction has the same latency
- Subject to **hazards**
  - *Structure, data, control*
- Instruction set design affects complexity of pipeline implementation

5

## MIPS Pipelined Datapath



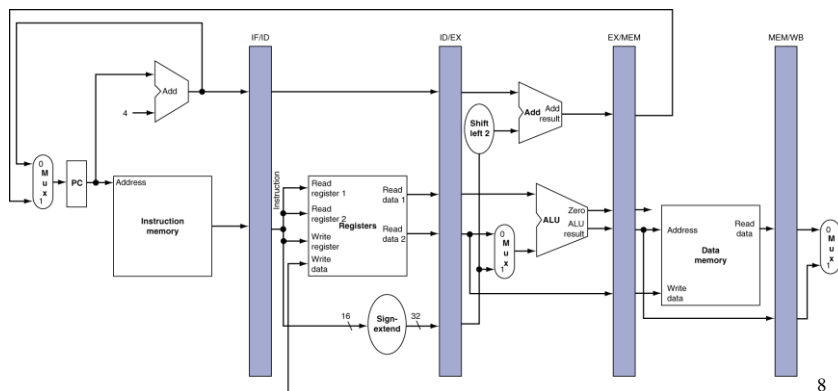
## Pipelined Data Path

- Data flows from left to right on the pipeline
- There are *two exceptions*:
  1. WB that writes the result back into the register file
  2. Selection of the next value of the PC, one input comes from the calculated branch address from the MEM stage
- Later instructions in the pipeline can be influenced by these two right-to-left data movements
  - The first one (WB to ID) leads to **data hazards**
  - The second one (MEM to IF) leads to **control hazards**

7

## Pipeline Registers

- Need registers between stages
  - To hold information produced in previous cycle, e.g. the destination register address, control signals

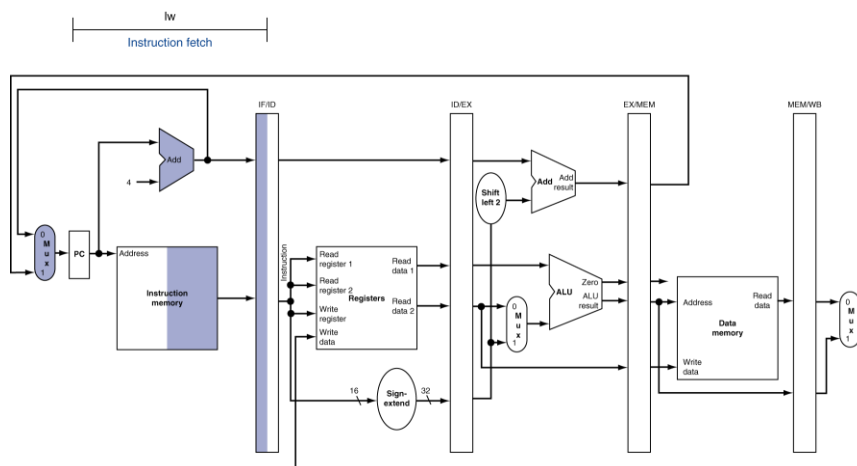


# Pipeline Operation

- Cycle-by-cycle flow of instructions through the pipelined datapath
  - “Single-clock-cycle” pipeline diagram
    - Shows pipeline usage in *a single cycle*
    - Highlight resources used
  - c.f. “multi-clock-cycle” diagram
    - Graph of operation over time
- We’ll look at “single-clock-cycle” diagrams for *load & store* instructions

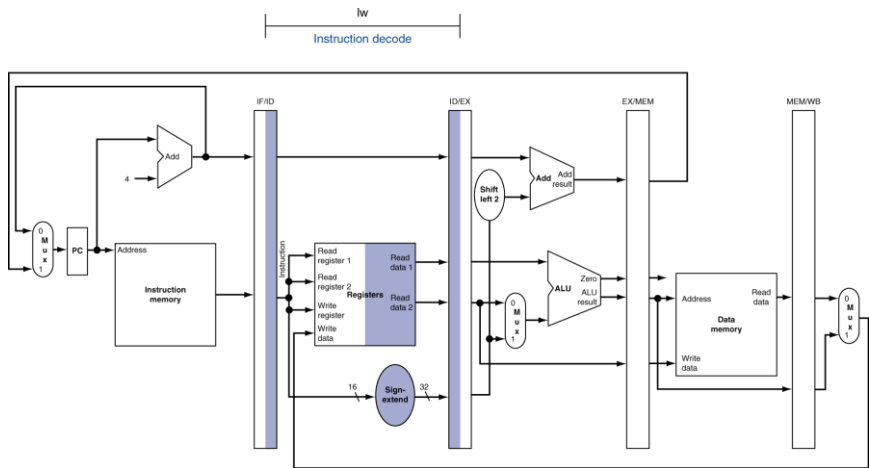
9

## IF for Load, Store, ...



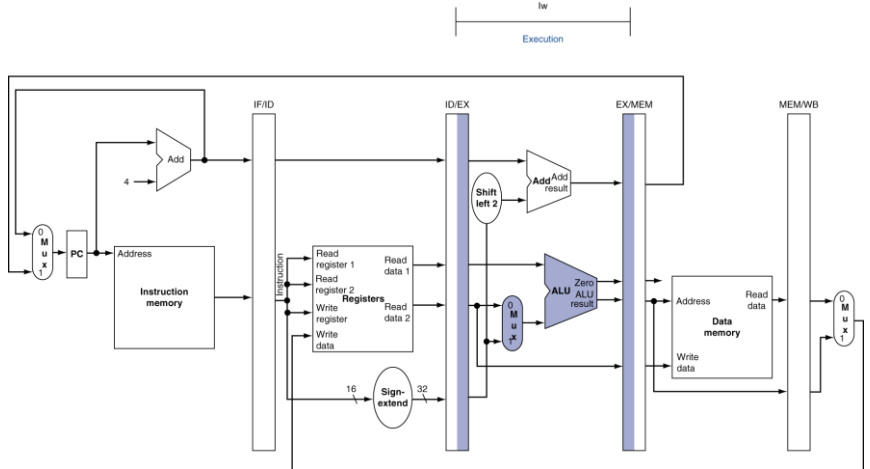
10

# ID for Load, Store, ...



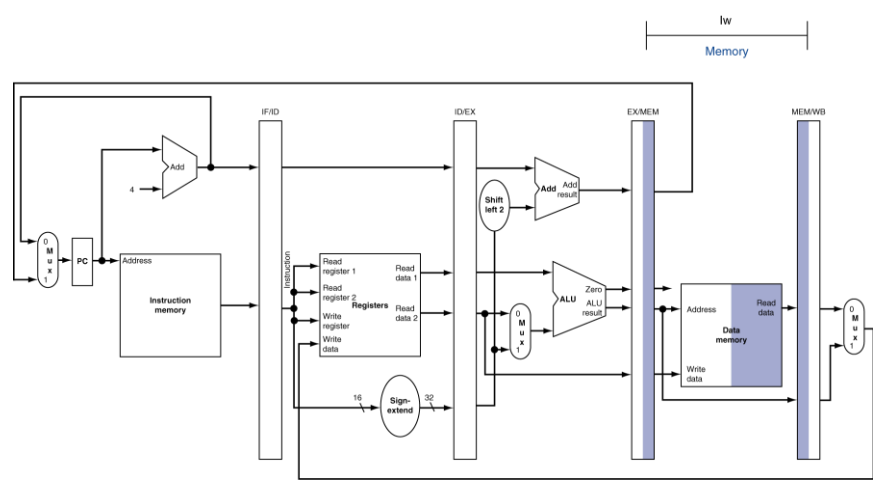
11

# EX for Load



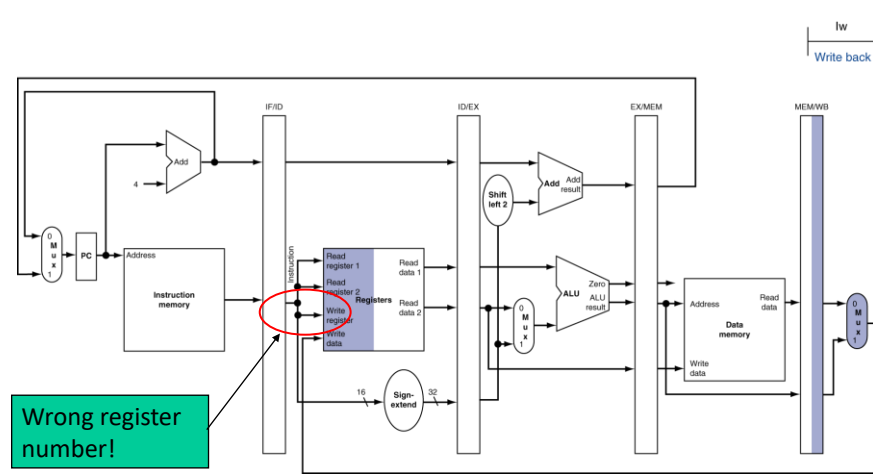
12

# MEM for Load



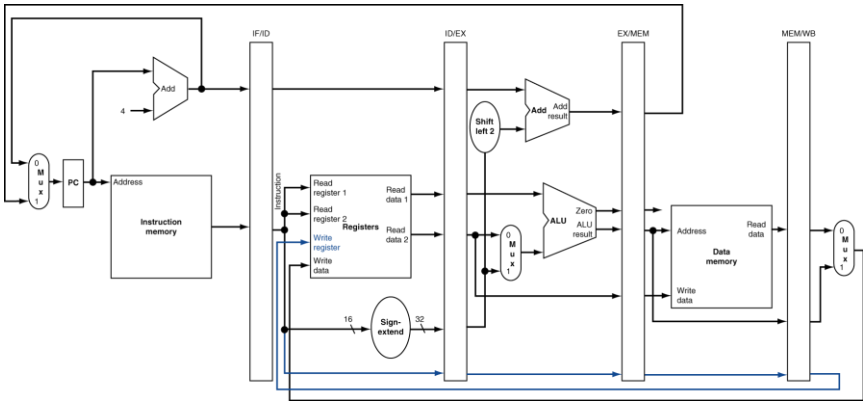
13

# WB for Load



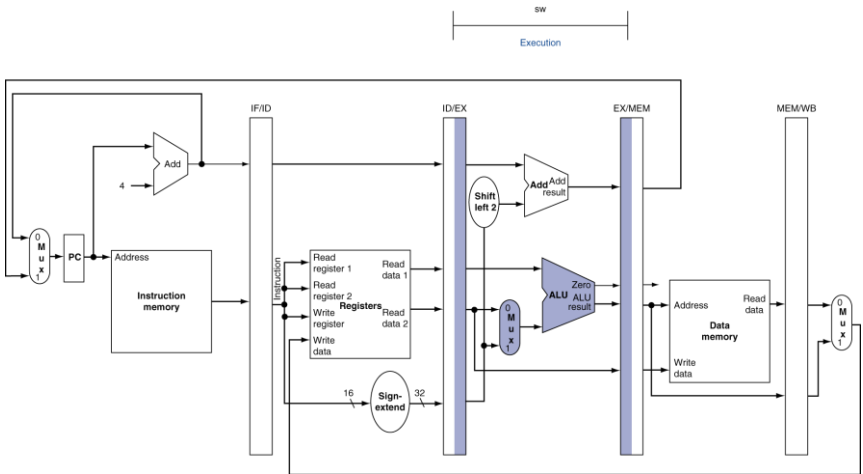
14

# Corrected Datapath for Load



15

# EX for Store



16



# MEM for Store

The diagram illustrates the MEM stage of a 5-stage processor for a Store instruction. The stages are labeled IF/ID, ID/EX, EX/MEM, and MEM/WB. The MEM stage is highlighted in blue.

**IF/ID Stage:** The instruction is fetched from Instruction memory. The PC (Program Counter) is updated with the next sequential instruction address. The instruction is then passed to the ID/EX stage.

**ID/EX Stage:** The instruction is decoded. The register file is accessed to read the two source registers (Read register 1 and Read register 2). The register file also provides the Write register and Write data. The instruction is then passed to the EX/MEM stage.

**EX/MEM Stage:** The ALU performs a store operation. The ALU inputs are the two source registers (Read data 1 and Read data 2) and the ALU operation code (Store). The ALU result is the address of the memory location to be stored. The ALU result is then passed to the EX/MEM stage.

**MEM/WB Stage:** The data is written to Data memory. The Data memory is accessed with the address from the ALU and the write data from the register file. The data is then written to the Data memory. The data is then passed to the MEM/WB stage.

**Sign-extend:** A 16-bit sign-extend block is shown, which takes a 16-bit value and extends it to 32 bits. This block is used to sign-extend the immediate value from the instruction.

**SW Memory:** A label "SW Memory" is placed above the EX/MEM stage, indicating that this stage is part of the store instruction's memory access phase.

# WB for Store

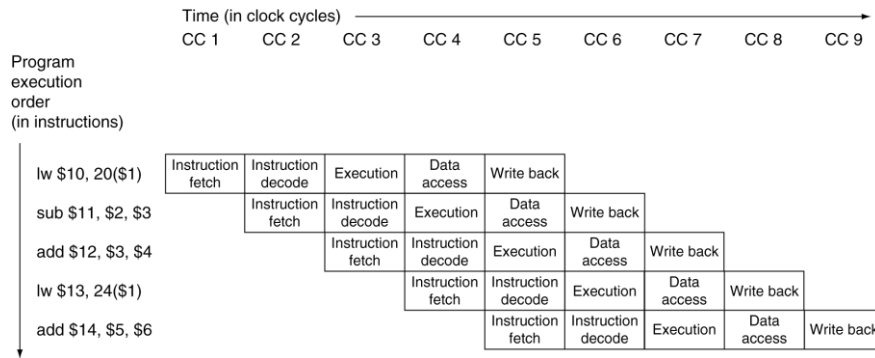
The diagram illustrates the write-back (WB) stage for a store instruction in a 5-stage MIPS processor. The stages are labeled IF/ID, ID/EX, EX/MEM, and MEM/WB. The final stage, MEM/WB, shows the write-back of data to memory. The diagram includes components such as the Program Counter (PC), Instruction Memory, Registers, ALU, Shift Register, and Data Memory. A legend indicates 'SW' for store write-back and 'Write-back' for the final output.

**Legend:**

- SW: Store Write-back
- Write-back: Final output

# Multi-Cycle Pipeline Diagram

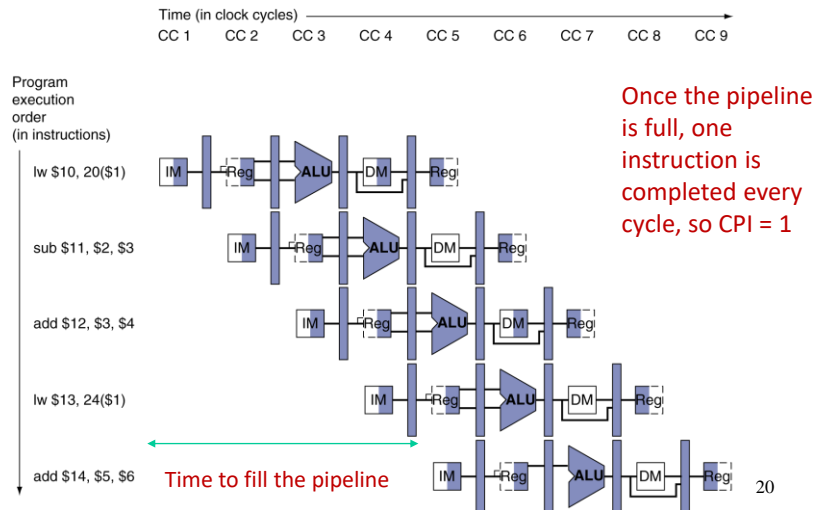
- Traditional form



19

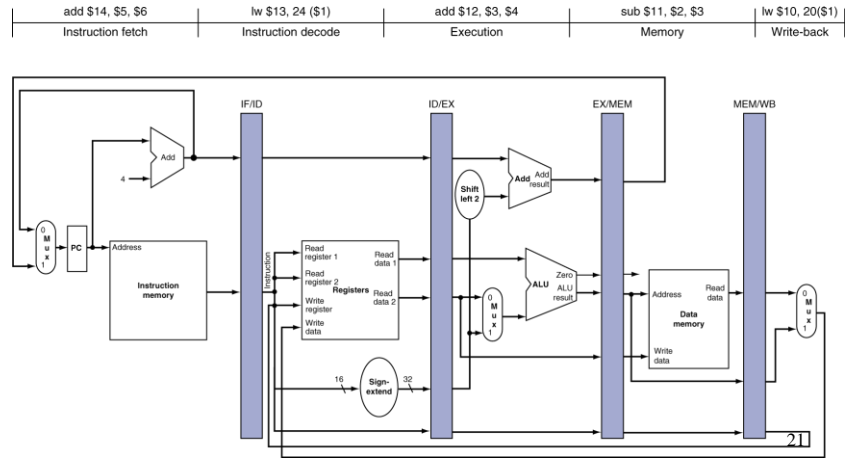
# Multi-Cycle Pipeline Diagram

- Form showing resource usage

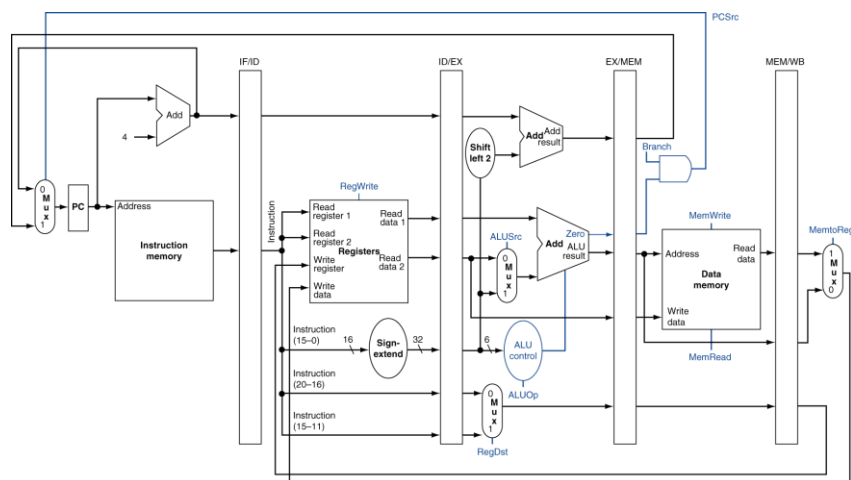


# Single-Cycle Pipeline Diagram

- State of pipeline in cycle CC5

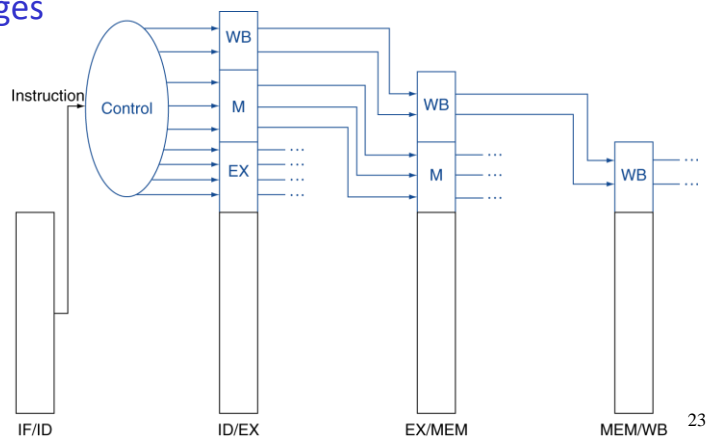


# Pipelined Control (Simplified)



# Pipelined Control

- Control signals derived from instruction in ID stage and held in pipeline registers between stages



# Pipelined Control

- IF stage: read instruction memory (always asserted) and write PC (on system clock edge)
- ID stage: no optional control signals to set

	EX Stage				MEM Stage			WB Stage	
	Reg Dst	ALU Op1	ALU Op0	ALU Src	Brch	Mem Read	Mem Write	Reg Write	Mem toReg
R									
lw									
sw									
beq									

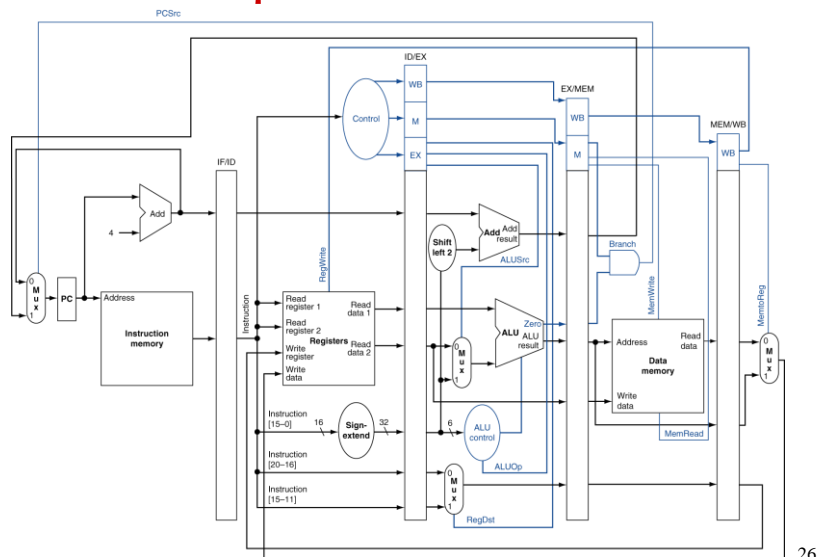
## Pipelined Control – con't

- IF stage: read instruction memory (always asserted) and write PC (on system clock edge)
- ID stage: no optional control signals to set

	EX Stage				MEM Stage			WB Stage	
	Reg Dst	ALU Op1	ALU Op0	ALU Src	Brch	Mem Read	Mem Write	Reg Write	Mem toReg
R	1	1	0	0	0	0	0	1	0
lw	0	0	0	1	0	1	0	1	1
sw	X	0	0	1	0	0	1	0	X
beq	X	0	1	0	1	0	0	0	X

25

## Pipelined Control



26

## Summary

- MIPS pipeline is consisted of five stages
  - Registers between stages to hold information produced in previous cycle
  - Data flows from left to right with exception of WB and MEM
- Two types of pipeline diagrams
  - Multi-cycle pipeline diagram shows resource usage
  - Single-cycle diagram shows state of pipeline in a given cycle
- Control signals derived from instruction in ID stage and held in pipeline registers between stages

27