

## Scoreboard Method

Instruction Status Table [Description 1]

- ✦ The Instruction Status Table indicates if an instruction is issued for execution
- ✦ For issued instruction the table gives the execution stage
- ✦ The decoder stage makes the decision if an instruction may be issued.
- ✦ If possible the decoder issues the instruction to the most suitable functional unit.

## Scoreboard Method

Instruction Status Table [Description 2]

- ✦ An instruction is issued if there is no
  - ✦ WAW hazards
  - ✦ Structural hazards
- ✦ Instructions with dependencies are executed in order
- ✦ While independent instructions may be executed out of order.

## Scoreboard Method

Instruction Status Table

Instructions	Issued	Operand Fetch Complete	Execution Complete	Write Back Complete

## Scoreboard Method

Functional Unit Status Table [Description ]

- ✦ Functional Unit Status Table:
  - ✦ Indicates busy if an instruction is issued
  - ✦ Identifies destination register
  - ✦ Identifies availability of source register
  - ✦ Source register is available if not destination for a different instruction

## Scoreboard Method

Functional Unit Status Table

Unit ID	Unit Name	Busy	Destination Register $R_d$	Source Registers			
				$R_{s1}$	Ready	$R_{s2}$	Ready

## Scoreboard Method

Destination Register Status Table [Description]

- ✦ Destination Register Status Table:
  - ✦ I identifies unwritten destination register
  - ✦ I identifies the functional unit that writes the destination register

## Scoreboard Method

Destination Register Status Table

	R1	R2	R3	R4	R5	R6
Unit Id						

## Scoreboard Method

Execution procedure

- ✦ During Operand Fetch Stage
  - ✦ Scoreboard checks if source register are available
  - ✦ Operands are read if not used as destination for other functional units
- ✦ After execution is complete scoreboard checks for WAR hazards before the result is written to the destination register

## Scoreboard Method

Example Program

```

LD      R1, A
LD      R2, B
LD      R3, C
LD      R4, D
MUL     R5, R1, R2  -- A*B
ADD     R2, R3, R4  -- C+D
ADD     R2, R2, R5  -- (A*B)+(C+D)

```

## Scoreboard Method

Instruction Status Table [Load R<sub>4</sub>, D]

Instructions	Issued	Operand Fetch Complete	Execution Complete	Write Back Complete
Load R <sub>1</sub> , A	YES	YES	YES	YES
Load R <sub>2</sub> , B	YES	YES	YES	YES
Load R <sub>3</sub> , C	YES	YES	YES	YES
Load R <sub>4</sub> , D	YES	YES		
Mul R <sub>5</sub> , R <sub>1</sub> , R <sub>2</sub>	YES	YES		
Add R <sub>2</sub> , R <sub>3</sub> , R <sub>4</sub>	YES			
Add R <sub>2</sub> , R <sub>2</sub> , R <sub>5</sub>				

Waiting for R<sub>4</sub>(WAR)  
R<sub>2</sub> is Destination

## Scoreboard Method

Functional Unit Status Table [Load R<sub>4</sub>, D]

Unit ID	Unit Name	Busy	Destination Register R <sub>d</sub>	Source Registers			
				R <sub>s1</sub>	Ready	R <sub>s2</sub>	Ready
1	Load/Store	YES	R4				
2	Multiplier	YES	R5	R1	YES	R2	YES
3	Adder_1	YES	R2	R3	YES	R4	NO
4	Adder_2	NO					

## Scoreboard Method

Destination Register Status Table [Load R<sub>4</sub>, D]

	R1	R2	R3	R4	R5	R6
Unit Id		3		1	2	

## Scoreboard Method

Instruction Status Table [Add  $R_2, R_3, R_4$ ]

Instructions	Issued	Operand Fetch Complete	Execution Complete	Write Back Complete
Load $R_1, A$	YES	YES	YES	YES
Load $R_2, B$	YES	YES	YES	YES
Load $R_3, C$	YES	YES	YES	YES
Load $R_4, D$	YES	YES	YES	YES
Mul $R_1, R_1, R_2$	YES	YES	YES	
Add $R_2, R_3, R_4$	YES	YES	YES	YES
Add $R_2, R_2, R_5$	YES			

Now Complete

Issued