

Planificación dinámica - Concepto

Emisión y ejecución en orden: una limitación.

Stall también detiene a operaciones independientes.

DIVD F0,F2,F4 (1)

ADDD F10,F0,F8 (2)

SUBD F12,F8,F14 (3)

No siempre ejecutar en el orden del programa.

Dividir la etapa ID en dos:

Emisión: decodificación y hazards estructurales.

Leer operandos: si no hay hazard, lee los operandos.

7/11/18



Guillermo Aguirre

1

Hazards de datos

Se produce cuando por la segmentación, el orden de LECTURA de los operandos y la ESCRITURA de resultados se modifica respecto a lo especificado en el programa.

Se produce un riesgo si existe dependencia entre instrucciones que se ejecutan concurrentemente.

Dominio: operandos de la instrucción

Rango: resultado de la instrucción

Si $a = b \odot c$

.....

$S_j \quad c = a \otimes e$

.....

$S_k \quad a = d \div e$

Situaciones: i precede a j

$D(i) \cap D(j) \neq \emptyset$ Sin RIESGO

$D(i) \cap R(j) \neq \emptyset$ riesgo WAR

$R(i) \cap D(j) \neq \emptyset$ riesgo RAW

$R(i) \cap R(k) \neq \emptyset$ riesgo WAW

7/11/18



Guillermo Aguirre

2

Planificación dinámica - Riesgos WAR y WAW

DIVD F0,F2,F4
ADDD F10,F0,F8
SUBD F8,F8,F14

Antidependencia (WAR)

DIVD F0,F2,F4
ADDD F10,F0,F8
SUBD F10,F8,F14

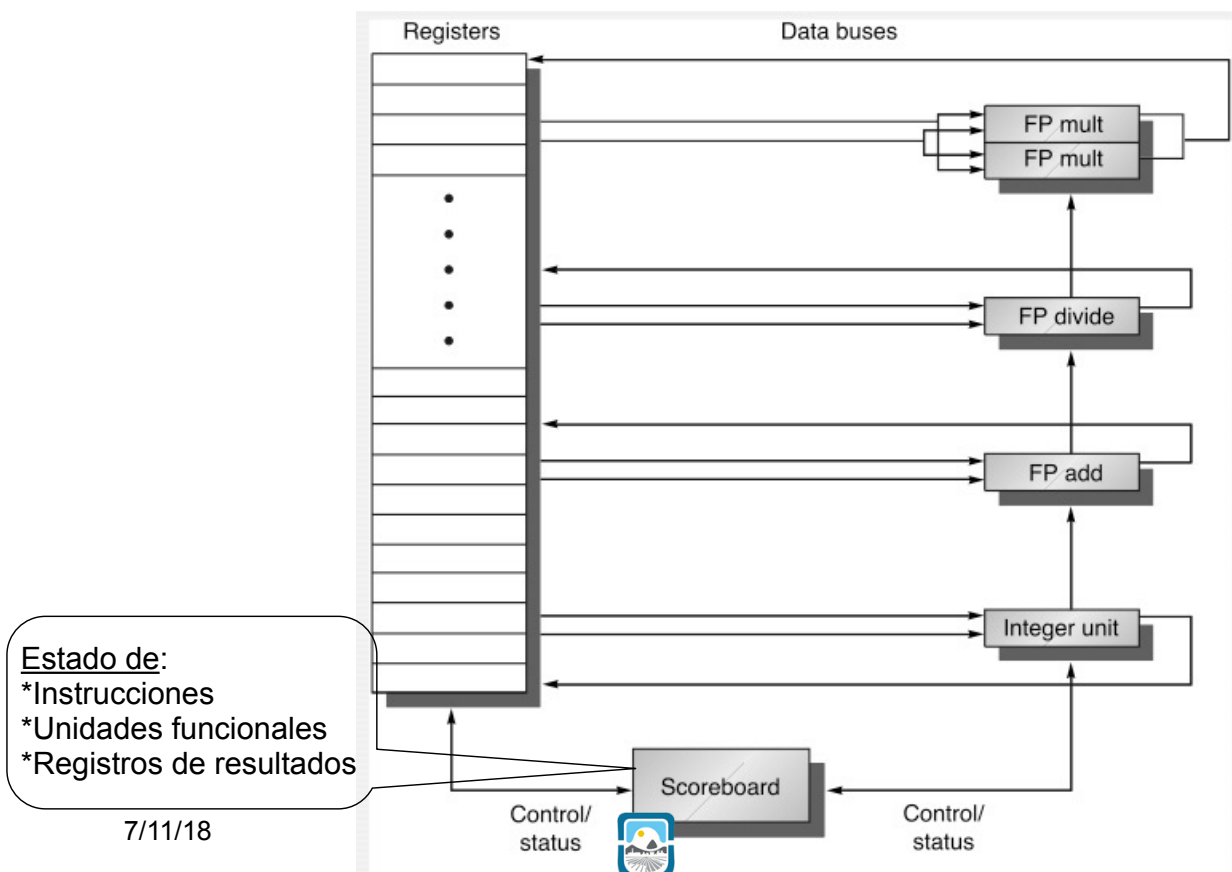
Riesgo WAW

7/11/18



Guillermo Aguirre

3



7/11/18

Etapas en Scoreboarding (1)

Etapas de despacho: controla que la unidad funcional esté disponible y que no haya otra instrucción que tenga el mismo registro destino. Con esto se evitan hazards estructurales y hazard WAW. Se despacha la instrucción y se actualizan las estructuras de datos. Si hay WAW o hazard estructural se genera un stall.



Etapas en Scoreboarding (2)

Etapas de emisión: Controla si los operandos están disponibles.

Resuelve RAW dinámicamente.

Si es posible se leen los operandos desde los registros para comenzar la ejecución.



Etapas en Scoreboarding (3)

Etapas de ejecución : Hace la ejecución y notifica al scoreboard que se ha completado la ejecución.

Reemplaza la etapa de EX en el pipeline clásico.



Etapas en Scoreboarding (4)

Etapas de escritura: Se chequea por hazards WAR y se hace un stall de la instrucción a punto de finalizar, si es necesario. Si el resultado se escribe en un operando fuente de alguna instrucción activa anterior, la instrucción no puede finalizar y debe esperar que la otra lea el operando.



Ejemplo ScoreBoard - Ciclo 3

<u>Instruction status</u>				Issue	Read operand	Execution complete	Write Result
Instruction	j	k		1	2	3	
LD F6 34+ R2							
LD F2 45+ R3							
MUL F0 F2 F4							
SUB F8 F6 F2							
DIV F10 F0 F6							
ADD F6 F8 F2							

Registros:
destino y
fuentes

Unidad que
produce el
valor

Flags:
operando
listo?

		dest	S1	S2	FU for j	FU for k	Fj?	Fk?
		Fi	Fj	Fk	Qj	Qk	Rj	Rk
Busy	Op							
Integer	Load	F6		R2				Yes
Mult1		No						
Mult2		No						
Add		No						
Divide		No						

<u>Register result status</u>		F0	F2	F4	F6	F8	F10	F12	...	F30
Clock	FU	Integer								
3										

7/11/18

Guillermo Aguirre

9

Ejemplo ScoreBoard - Ciclo 9

Instruction status				Read	Execution	Write						
Instruction	j	k		Issue	operands complete	Result						
LD F6 34+ R2				1	2	3	4					
LD F2 45+ R3				5	6	7	8					
MUL F0 F2 F4				6	9							
SUB F8 F6 F2				7	9							
DIV F10 F0 F6				8								
ADD F6 F8 F2												
Functional unit status												
Time	Name			Busy	Op	dest	SI	S2	FU for j	FU for k	Fj?	Fk?
						Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integer			No								
10	Mult1			Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2			No								
2	Add			Yes	Sub	F8	F6	F2			Yes	Yes
	Divide			Yes	Div	F10	F0	F6	Mult1		No	Yes
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
9	FU			Mult1				Add	Divide			

Ejemplo ScoreBoard - Ciclo 17

Instruction status				Read	Execution	Write	
Instruction	j	k		Issue	operand: complete	Result	
LD F6 34+ R2				1	2	3	4
LD F2 45+ R3				5	6	7	8
MULIF0 F2 F4				6	9		
SUBDF8 F6 F2				7	9	11	12
DIVDF10 F0 F6				8			
ADDDF6 F8 F2				13	14	16	

La suma toma 2 ciclos

Functional unit status				dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integer	No								
2	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	No								
	Add	Yes	Add	F6	F8	F2			Yes	Yes
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status				F0	F2	F4	F6	F8	F10	F12	...	F30
Clock												
17	FU	Mult1	Add	Divide								

7/11/18



Guillermo Aguirre

11

Ejemplo ScoreBoard - Ciclo 62

Instruction status				Issue	Read operands	Execution complete	Write Result
Instruction	j	k					
LD F6 34+ R2				1	2	3	4
LD F2 45+ R3				5	6	7	8
MULIF0 F2 F4				6	9	19	20
SUBDF8 F6 F2				7	9	11	12
DIVDF10 F0 F6				8	21	61	62
ADDDF6 F8 F2				13	14	16	22

La multiplicación toma 10 ciclos

La división toma 40 ciclos

Functional unit status				dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
0	Divide	No								

Register result status				F0	F2	F4	F6	F8	F10	F12	...	F30
Clock												
62	FU											

7/11/18



Guillermo Aguirre

12

Costo y beneficio

Mejora del desempeño entre 1.7 y 2.5

El scoreboard no requiere muchos más circuitos que las unidades comunes.

Requiere muchos buses, \cong 4 veces más.

De interés al considerar :

- Despacho de múltiples instrucciones por ciclo
- Especulación



Factores limitantes

Cantidad de paralelismo disponible entre las instrucciones.

Número de entradas en el ScoreBoard.

Número y tipo de unidades funcionales.

Presencia de antidependencias y dependencias de salida



Algoritmo de Scoreboard (1)

Emisión o Issue

Esperar que	Tareas
Unidad Funcional[FU] no esté ocupada y Result[D] esté en blanco	$\text{Busy[FU]} \leftarrow \text{yes}; \text{Op[FU]} \leftarrow \text{op};$ $\text{Fi[FU]} \leftarrow D \text{ } \text{Fj[FU]} \leftarrow S1;$ $\text{Fk[FU]} \leftarrow S2; \text{Qj} \leftarrow \text{Result}[S1];$ $\text{Qk} \leftarrow \text{Result}[S2]; \text{Rj} \leftarrow \text{not Qj};$ $\text{Rk} \leftarrow \text{not Qk}; \text{result}[D] \leftarrow \text{FU}$

7/11/18



Guillermo Aguirre

15

Algoritmo de Scoreboard (2)

Lectura de registros y ejecución

Esperar que	Tareas
Rj y Rk sean verdaderos	$\text{Rj} \leftarrow \text{no}; \text{Rk} \leftarrow \text{no};$ $\text{Qj} \leftarrow 0; \text{Qk} \leftarrow 0$
La FU finalice	

7/11/18



Guillermo Aguirre

16

Algoritmo de Scoreboard (3)

Escritura de resultados

Esperar que	Tareas
$\forall f(($ $Fj[f] \neq Fi[FU] \vee Rj[f] = \text{no}$ $) \wedge ($ $Fk[f] \neq Fi[FU] \vee Rk[f] = \text{no}$ $)$ $)$	$\forall f(\text{ Si } Qj[f] = FU \text{ then } Rj[f] \leftarrow \text{yes};$ $\forall f(\text{ Si } Qk[f] = FU \text{ then } Rk[f] \leftarrow \text{yes};$ $\text{Result}[Fi[FU]] \leftarrow 0; \text{Busy}[FU] \leftarrow \text{No}$