

University of Engineering and Technology (UET), Peshawar, Pakistan

Lecture 11

CSE-304: Computer Organization and Architecture

BY:

Dr. Muhammad Athar Javed Sethi

Data Dependences

- An instruction j is data dependent on instruction i if either of the following hold;
 - Instruction i produces a result used by instruction j, or
 - Instruction j is data dependent on instruction k, and instruction k is data dependent on instruction i.
- If dependent, can't execute in parallel
- Easy to determine for registers (fixed names)
- Hard for memory:
 - Does 100(R4) = 20(R6)?

Name Dependences

- Name dependence occurs when two instructions use the same register or memory location called a name.
- Types:
 - Anti dependence
 - Occurs between instruction i and instruction j, when instruction j writes a register or memory location that instruction i reads.
 - Output Dependence
 - Occurs between instruction i and instruction j, when instruction i and j write the same register or memory location.
- Solution:
- Register renaming

Data Hazards

- RAW (Read after Write) Hazard.
 - j tries to read a source before i writes it, so j incorrectly gets the old value.

- WAR (Write after Read) Hazard.
 - j tries to write a destination before it is read by i, so i incorrectly gets the new value.
 - Antidependence.

Data Hazards

- WAW (Write after Write) Hazard.
 - j tries to write an operand before it is written by i. The write end up being performed in wrong order, leaving the value written by i rather than the value written by j in the destination.
 - Output dependence.

Control Dependences

- Another kind of dependence called control dependence
- Example

```
if p1 {S1;};
```

- if p2 {S2;};
- S1 is control dependent on p1 and S2 is control dependent on p2 but not on p1.

Control Dependences

- Two (obvious) constraints on control dependences:
 - An instruction that is control dependent on a branch cannot be moved before the branch so that its execution is no longer controlled by the branch.
 - An instruction that is not control dependent on a branch cannot be moved to after the branch so that its execution is controlled by the branch.

Examples: Control Dependences

• Example1:

- · DADDU R2,R3,R4
- BEQZ R2, L1
- LW R1, 0(R2)
- L1: <statements>

• Example2:

- DADDU R1,R2, R3
- BEQZ R4,L
- DSUBU R1,R5,R6
- L: <statements>
- OR R7,R1,R8

Dynamic Scheduling HW Schemes: Instruction Parallelism

- In dynamic scheduling, the hardware rearranges the instruction execution to reduce the stalls while maintaining data flow.
- In order instruction issue
- Out of order execution
- Scoreboards allow instruction to execute whenever 1 & 2 hold, not waiting for prior instructions
 - 1. Issue—decode instructions, check for structural hazards
 - 2.Read operands—wait until no data hazards, then read operands
- CDC 6600: In order issue, out of order execution, out of order commit (also called completion)

Four Stages of Scoreboard Control

- Issue—decode instructions & check for structural hazards (ID1)
 - If a functional unit for the instruction is free and no other active instruction has the same destination register (WAW), the scoreboard issues the instruction to the functional unit and updates its internal data structure. If a structural or WAW hazard exists, then the instruction issue stalls, and no further instructions will issue until these hazards are cleared.
- 2. Read operands—wait until no data hazards, then read operands (ID2)
 - A source operand is available if no earlier issued active instruction is going to write it, or if the register containing the operand is being written by a currently active functional unit. When the source operands are available, the scoreboard tells the functional unit to proceed to read the operands from the registers and begin execution. The scoreboard resolves RAW hazards dynamically in this step, and instructions may be sent into execution out of order.

Four Stages of Scoreboard Control

3. Execution—operate on operands (EX)

 The functional unit begins execution upon receiving operands. When the result is ready, it notifies the scoreboard that it has completed execution.

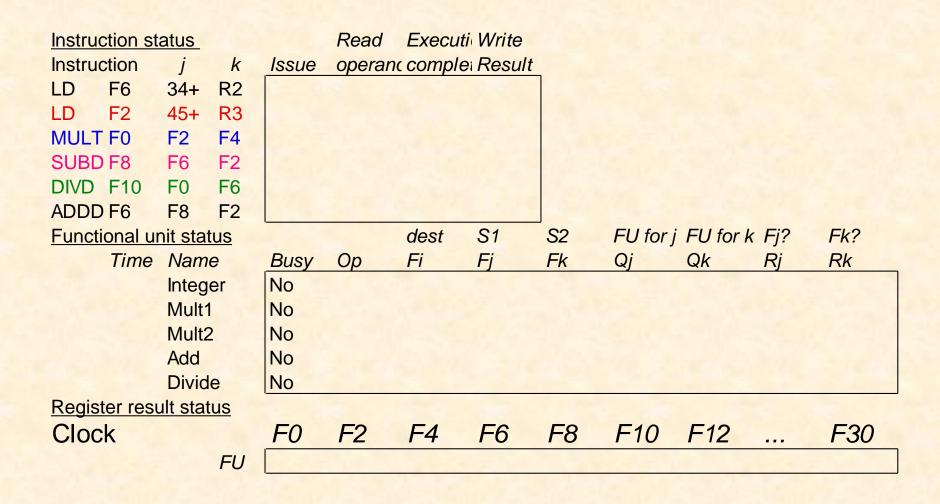
4. Write result—finish execution (WB)

 Once the scoreboard is aware that the functional unit has completed execution, the scoreboard checks for WAR hazards. If none, it writes results. If WAR, then it stalls the instruction.

Three Parts of the Scoreboard

- 1. Instruction status—which of 4 steps the instruction is in
- Functional unit status—Indicates the state of the functional unit (FU). 9 fields for each functional unit
 - Busy—Indicates whether the unit is busy or not
 - Op—Operation to perform in the unit (e.g., + or –)
 - Fi—Destination register
 - Fj, Fk—Source-register numbers
 - Qj, Qk—Functional units producing source registers Fj, Fk
 - Rj, Rk—Flags indicating when Fj, Fk are ready
- 3. Register result status—Indicates which functional unit will write each register, if one exists. Blank when no pending instructions will write that register

Scoreboard Example



Instruction s	status			Read	Execut	ic Write)				
Instruction	j	k	Issue	operan	d comple	et Resu	ılt				
LD F6	34+	R2	1			3 3 1					
LD F2	45+	R3									
MULT F0	F2	F4	1000								
SUBD F8	F6	F2									
DIVD F10	F0	F6									
ADDD F6	F8	F2									
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	e	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	Yes	Load	F6		R2				Yes
	Mult'	1	No								
	Mult2	2	No								
	Add		No								
	Divid	le	No								
Register res	sult sta	atus									
Clask			F0	F2	F4	F6	F8	F10	F12		F30
Clock											
CIOCK 1		FU	15.3 3			Integ	er	1000			

tus		Read	Execu	ıtic Write					
j k	Issu	e operal	nd compl	et Resu	lt				
4+ R2	2 1	2							
5+ R3	3								
2 F4									
6 F2	2								
0 F6	5								
8 F2									
t status			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
lame	Bus	у Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
nteger	Yes	Load	F6		R2				Yes
/lult1	No								
/lult2	No								
dd	No								
ivide	No								
t status									
	F0	F2	F4	F6	F8	F10	F12		F30
FU	J		THE RE	Intege	er				
	54+ R2 55+ R3 66 F2 60 F6 78 F2 10 Status 10 Status 11 Status 12 Status 12 Status 12 Status 13 Status 14 Status 15 Status 16 Status 17 Status 17 Status 17 Status 18 S	j k Issult 64+ R2 1 5+ R3 6 F2 7 F3 8 F2 1 Status 1 No 2 No 2 No 3 No 4 No 5 No 6	j k Issue operation 44+ R2 1 2 55+ R3 52 F4 56 F2 50 F6 58 F2 51 status Name Busy Op Treger Yes Load No No No No No No No Treger No N	issue operand completed in the second	Jame Susue Operand complet Results Susue Operand complet Results Susue S	J		Sue Operand complet Result 1 2 2 5 6 F2 5 6 F2 5 6 F2 F2	Same Same

Issue 2nd LD?

Instruction status		Read	Execut	ic Write)				
Instruction j k	Issue	operan	d comple	et Resu	lt				
LD F6 34+ R2	1	2	3		177.6				
LD F2 45+ R3	4.77				100				
MULT F0 F2 F4									
SUBD F8 F6 F2					- 17				
DIVD F10 F0 F6	(d) 10				119				
ADDD F6 F8 F2									
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	Yes	Load	F6		R2				Yes
Mult1	No								
Mult2	No								
Add	No								
Divide	No	- 7576					100		
Register result status					Шя,				
Clock	F0	F2	F4	F6	F8	F10	F12		F30
3 FU				Integ	er				

Issue MULT?

Instruction s	status			Read	Executi	Write					
Instruction	j	k	Issue	operand	d complet	t Resu	lt				
LD F6	34+	R2	1	2	3	4	-177.4				
LD F2	45+	R3									
MULT F0	F2	F4									
SUBD F8	F6	F2									
DIVD F10	F0	F6	Garage 1								
ADDD F6	F8	F2									
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	Yes	Load	F6		R2				Yes
	Mult'	1	No								
	Mult2	2	No								
	Add		No								
	Divid	le	No	14.76					100		
Register res	sult sta	atus									
Clock			FO	F2	F4	F6	F8	F10	F12		F30
4		FU				Intege	er				

Instruction status		Read	Execut	ic Write)				
Instruction j l	Issue	operand	d comple	et Resu	ılt				
LD F6 34+ R2	2 1	2	3	4					
LD F2 45+ R3	5								
MULT F0 F2 F4									
SUBD F8 F6 F2									
DIVD F10 F0 F6									
ADDD F6 F8 F2					7.64				
Functional unit status			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time Name	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	Yes	Load	F2		R3				Yes
Mult1	No								
Mult2	No								
Add	No								
Divide	No								
Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12		F30
5 FL	J	Integer							

Instruction s	status			Read	Execution	Write					
Instruction	j	k	Issue	operand	complet	t Resu	lt				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6							
MULT FO	F2	F4	6								
SUBD F8	F6	F2					- 1				
DIVD F10	F0	F6	SILITY OF				000				
ADDD F6	F8	F2									
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	Yes	Load	F2		R3				Yes
	Mult1	1	Yes	Mult	F0	F2	F4	Integer		No	Yes
	Mult2	2	No								
	Add		No								
	Divid	le	No	7876		100	760	77	1476		
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
6		FU	Mult1	Integer							

Insti	uction s	status			Read	Executi	(Write					
Insti	uction	j	k	Issue	operand	comple	t Resu	lt				
LD	F6	34+	R2	1	2	3	4					
LD	F2	45+	R3	5	6	7						
MUI	_T F0	F2	F4	6								
SUE	BD F8	F6	F2	7				441				
DIVI) F10	F0	F6	1100								
ADE	DF6	F8	F2									
Fun	ctional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
	Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
		Integ	er	Yes	Load	F2		R3				Yes
		Mult	1	Yes	Mult	F0	F2	F4	Integer		No	Yes
		Mult2	2	No								
		Add		Yes	Sub	F8	F6	F2		Integer	Yes	No
		Divid	le	No								
Reg	ister res	sult sta	atus									
Clo	ock			F0	F2	F4	F6	F8	F10	F12		F30
	7		FU	Mult1	Integer			Add				

Read multiply operands?

Instruction j k Issue operand complet Result	Instruction s	tatus			Read	Executi	(Write					
LD F2 45+ R3 5 6 7 MULT F0 F2 F4 6 SUBD F8 F6 F2 7 DIVD F10 F0 F6 8 ADDD F6 F8 F2 Functional unit status Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Integer Yes Load F2 R3 Yes Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Add Yes Sub F8 F6 F2 Integer Yes No Divide Yes Div F10 F0 F6 Mult1 No Yes Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30	Instruction	j	k	Issue	operan	d comple	t Resu	lt				
MULT F0 F2 F4 6 SUBD F8 F6 F2 7 DIVD F10 F0 F6 8 ADDD F6 F8 F2 Functional unit status dest S1 S2 FU for j FU for k Fj? Fk? Find Fine Name Busy Op Fi Fj Fk Qj Qk Rj Rk Integer Yes Load F2 F4 Integer No Yes Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Yes Div F10 F0 F6 Mult1 No Yes Register result status F0 F2 F4 F6 F8 F10 F12 F30	LD F6	34+	R2	1	2	3	4					
SUBD F8 F6 F2 7 DIVD F10 F0 F6 8 ADDD F6 F8 F2 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 Yes Load F2 R3 Yes Yes Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Yes Sub F8 F6 F2 Integer Yes No Add Yes Div F10 F0 F6 Mult1 No Yes Register result status F0 F2 F4 F6 F8 F10 F12 F30	LD F2	45+	R3	5	6	7						
DIVD F10 F0 F6 ADDD F6 F8 F2 8 Functional unit status dest S1 S2 FU for j FU for k Fj? Fk? Time Name Integer Mult1 Busy Op Fi Fj Fk Qj Qk Rj Rk Mult1 Yes Load F2 R3 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Add Yes Sub F8 F6 F2 Integer Yes No Divide Yes Div F10 F0 F6 Mult1 No Yes Register result status F0 F2 F4 F6 F8 F10 F12 F30	MULT FO	F2	F4	6								
ADDD F6 F8 F2 Functional unit status Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Integer Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 Add Yes Sub F8 F6 F2 Integer Yes No Divide Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30	SUBD F8	F6	F2	7				1923				
Functional unit status Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Integer Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 Add Yes Sub F8 F6 F2 Integer Yes No Divide Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30	DIVD F10	F0	F6	8								
Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Integer Yes Load F2 R3 Yes Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Add Yes Sub F8 F6 F2 Integer Yes No Divide Yes Div F10 F0 F6 Mult1 No Yes Register result status F0 F2 F4 F6 F8 F10 F12 F30 Register result status F0 F2 F4 F6 F8 F10 F12 F30 Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Yes Yes Yes Yes No Yes No No Yes Yes No No Yes No Register result status F0 F2 F4 F6 F8 F10 F12 F30 Time Name Busy Op Fi Fj Fk Qj Qk Rj Rk Yes Load F2 R3 Yes Yes No Yes No No Yes No Yes Register result status F0 F8 F10 F12 F30 Time Name F2 F4 F6 F8 F10 F12 F30 Time Name F2 F4 F6 F8 F10 F12 F30 Time Name F2 F4 F6 F8 F10 F12 F30 Time Name F2 F4 F6 F8 F10 F12 F30 Time Name F1 F1 F1 F1 F1 F1 Time Name F1 F1 F1 F1 F1 Time Name F2 F4 F6 F8 F10 F12 F30 Time Name F1 F1 F1 F1 F1 F1 Time Name F1 F1 F1 F1 F1 F1 Time Name F1 F1 F1 F1 Time Name F1 F1 F1 F1 Time Name F1 F1 F1 F1 F1 Time Name F1 F1 F1 F1 F1 Time Name F1 F1 F1 F1 Time Name F1 F1 F1 F1 Time	ADDD F6	F8	F2	//	1970							
Integer Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 Add Divide Register result status Clock Yes Load F2 R3 Yes No Yes No Yes No F2 F4 Integer No Yes No Yes No F6 F2 Integer Yes No Yes No F6 F6 F7 F	Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Mult1 Yes Mult F0 F2 F4 Integer No Yes Mult2 No Add Yes Sub F8 F6 F2 Integer Yes No Divide Yes Div F10 F0 F6 Mult1 No Yes Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30	Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Mult2 Add Add Yes Sub F8 F6 F2 Integer Yes No Divide Pegister result status Clock No Yes Div F10 F0 F6 Mult1 No Yes F0 F2 F4 F6 F8 F10 F12 F30		Integ	er	Yes	Load	F2		R3				Yes
Add Yes Sub F8 F6 F2 Integer Yes No Yes Div F10 F0 F6 Mult1 No Yes Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30		Mult1	1	Yes	Mult	F0	F2	F4	Integer		No	Yes
Divide Yes Div F10 F0 F6 Mult1 No Yes Register result status Clock F0 F2 F4 F6 F8 F10 F12 F30		Mult2	2	No								
Register result status Clock FO F2 F4 F6 F8 F10 F12 F30		Add		Yes	Sub	F8	F6	F2		Integer	Yes	No
Clock F0 F2 F4 F6 F8 F10 F12 F30		Divid	le	Yes	Div	F10	F0	F6	Mult1		No	Yes
	Register res	sult sta	atus									
8 FU Mult1 Integer Add Divide	Clock			F0	F2	F4	F6	F8	F10	F12		F30
× × × × × × × × × × × × × × × × × × ×	8		FU	Mult1	Integer			Add	Divide			

Instruction s	status			Read	Executi	ic Write					
Instruction	j	k	Issue	operan	d comple	t Resu	lt				
LD F6	34+	R2	1	2	3	4	-				
LD F2	45+	R3	5	6	7	8					
MULT F0	F2	F4	6								
SUBD F8	F6	F2	7								
DIVD F10	F0	F6	8								
ADDD F6	F8	F2					76				
Functional u	unit sta	atus	7		dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
	Mult1	1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	2	No								
	Add		Yes	Sub	F8	F6	F2			Yes	Yes
	Divid	le	Yes	Div	F10	F0	F6	Mult1		No	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
8		FU	Mult1				Add	Divide			

Instruction status		Read	Execu	Writ	te				
Instructio j k	Issue	opera	comp	Res	ult				
LD F6 34+ R2	1	2	3	4					
LD F2 45+ R3	5	6	7	8	1/4/7				
MULF0 F2 F4	6	9							
SUB F8 F6 F2	7	9			- 1				
DIVEF10 F0 F6	8								
ADD F6 F8 F2									
Functional unit st	atus		dest	S1	S2	FU for	FU for	Fj?	Fk?
Tim Name	Busy	On	Fi	⊏i.		O:	01.	D :	DI
	Dasy	Op		Fj	Fk	Qj	Qk	Rj	Rk
Integer	No	/ Ορ	Г		rk	QJ	QK	RJ	RK
	No	Mult	F0	F2		Qj	QK	Yes	Yes
Integer	No					Qj	QK		
Integer 10 Mult1	No Yes	Mult		F2		Qj	QK	Yes	
Integer 10 Mult1 Mult2	No Yes No	Mult Sub	FO	F2	F4	Mult1	QK	Yes	Yes
Integer 10 Mult1 Mult2 2 Add	No Yes No Yes Yes	Mult Sub	F0 F8	F2 F6	F4 F2		QK	Yes Yes	Yes Yes
Integer 10 Mult1 Mult2 2 Add Divide	No Yes No Yes Yes	Mult Sub Div	F0 F8	F2 F6 F0	F4 F2 F6	Mult1	F12	Yes Yes No	Yes Yes

Read operands for MULT & SUBD? Issue ADDD?

		Read		_		Jycic			
Instruction status									
Instructio j k	Issue	operai	comp	Res	ult				
LD F6 34+ R2	1	2	3	4					
LD F2 45+ R3	5	6	7	8					
MULF0 F2 F4	6	9							
SUB F8 F6 F2	7	9	11						
DIVEF10 F0 F6	8								
ADD F6 F8 F2									
Functional unit sta	atus		dest	S1	S2	FU for	FU for	Fj?	Fk?
Tim Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
<i>Tim Name</i> Integer	Busy No	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	No		Fi F0			Qj	Qk	<i>Rj</i> Yes	<i>Rk</i> Yes
Integer 8 Mult1	No					Qj	Qk		
Integer 8 Mult1 Mult2	No Yes	Mult		F2	F4	Qj	Qk	Yes	
Integer 8 Mult1 Mult2 0 Add	No Yes No Yes	Mult	F0 F8	F2	F4	Qj Mult1	Qk	Yes	Yes
Integer 8 Mult1 Mult2 0 Add	No Yes No Yes Yes	Mult Sub	F0 F8	F2 F6	F4 F2		Qk	Yes Yes	Yes Yes
Integer 8 Mult1 Mult2 0 Add Divide	No Yes No Yes Yes	Mult Sub	F0 F8 F10	F2 F6 F0	F4 F2 F6	Mult1		Yes Yes No	Yes Yes

Instruction s	status			Read	Execut	ic Write					
Instruction	j	k	Issue	operand	d <mark>comple</mark>	t Resu	lt				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8					
MULT FO	F2	F4	6	9							
SUBD F8	F6	F2	7	9	11	12	TALE				
DIVD F10	F0	F6	8				7 64				
ADDD F6	F8	F2	9-11								
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
7	Mult1	1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	2	No								
	Add		No								
	Divid	le	Yes	Div	F10	FO	F6	Mult1		No	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
12		FU	Mult1					Divide			

Read operands for DIVD?

Instruction status		Read	Execut	tic <mark>Write</mark>					
Instruction j	k Issue	operan	d <mark>co</mark> mple	et Resu	lt				
LD F6 34+ R	2 1	2	3	4					
LD F2 45+ R	3 5	6	7	8					
MULT F0 F2 F4	4 6	9							
SUBD F8 F6 F2	2 7	9	11	12					
DIVD F10 F0 F	6 8								
ADDD F6 F8 F2	2 13	1970							
Functional unit status	<u>s</u>		dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time Name	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	No								
6 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	<u>s</u>								
Clock	F0	F2	F4	F6	F8	F10	F12		F30
13 F	U Mult1			Add		Divide			

Instruction status		Read	Execut	ic Write	,				
Instruction j k	Issue	operan	d comple	et Resu	lt				
LD F6 34+ R2	1	2	3	4					
LD F2 45+ R3	5	6	7	8	-				
MULT F0 F2 F4	6	9							
SUBD F8 F6 F2	7	9	11	12	17.				
DIVD F10 F0 F6	8								
ADDD F6 F8 F2	13	14							
Functional unit status			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time Name	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	No								
5 Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
Mult2	No								
2 Add	Yes	Add	F6	F8	F2			Yes	Yes
Divide	Yes	Div	F10	F0	F6	Mult1	1476	No	Yes
Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12		F30
14 FU	Mult1			Add		Divide			

Instruction status			Read	Execution	Write					
Instruction j	k	Issue	operand	complet	Resu	lt				
LD F6 34+ I	R2	1	2	3	4					
LD F2 45+ I	R3	5	6	7	8					
MULT F0 F2 I	F4	6	9							
SUBD F8 F6 I	F2	7	9	11	12					
DIVD F10 F0 I	F6	8								
ADDD F6 F8 I	F2	13	14							
Functional unit stat	<u>us</u>			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time Name		Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Intege	r	No								
4 Mult1		Yes	Mult	F0	F2	F4			Yes	Yes
Mult2		No								
1 Add		Yes	Add	F6	F8	F2			Yes	Yes
Divide		Yes	Div	F10	F0	F6	Mult1		No	Yes
Register result state	<u>us</u>	THE REAL PROPERTY.				-				
Clock		F0	F2	F4	F6	F8	F10	F12		F30
15	FU	Mult1			Add		Divide			

Instruction s	status			Read	Execut	ic Write	,				
Instruction	j	k	Issue	operan	d comple	et Resu	lt				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8					
MULT FO	F2	F4	6	9							
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8				7				
ADDD F6	F8	F2	13	14	16						
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
3	Mult1	1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	2	No								
0	Add		Yes	Add	F6	F8	F2			Yes	Yes
	Divid	le	Yes	Div	F10	F0	F6	Mult1		No	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
16		FU	Mult1			Add		Divide			

Instruction s	tatus			Read	Execut	ic <mark>Write</mark>	,				
Instruction	j	k	Issue	operan	d <mark>comple</mark>	et Resu	<u>l</u> t				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8					
MULT F0	F2	F4	6	9							
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8								
ADDD F6	F8	F2	13	14	16						
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
1	Mult1	1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	2	No								
	Add		Yes	Add	F6	F8	F2			Yes	Yes
	Divid	le	Yes	Div	F10	F0	F6	Mult1		No	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
18		FU	Mult1			Add		Divide			

Instruction state	us_		Read	Execut	ic <mark>Write</mark>					
Instruction j	k	Issue	operand	d comple	et Resu	lt				
LD F6 34	l+ R2	1	2	3	4					
LD F2 45	5+ R3	5	6	7	8					
MULT FO F2	2 F4	6	9	19						
SUBD F8 F6	6 F2	7	9	11	12					
DIVD F10 F0) F6	8				7				
ADDD F6 F8	3 F2	13	14	16						
Functional unit	status			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time Na	ame	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Int	teger	No								
0 M	ult1	Yes	Mult	F0	F2	F4			Yes	Yes
M	ult2	No								
Ac	dd	Yes	Add	F6	F8	F2			Yes	Yes
Di	vide	Yes	Div	F10	F0	F6	Mult1		No	Yes
Register result	status									
Clock		F0	F2	F4	F6	F8	F10	F12		F30
19	FU	Mult1			Add		Divide			

Instruction	status			Read	Execut	tic Write					
Instruction	j	k	Issue	operan	d comple	et Resu	lt				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8	- 1				
MULT F0	F2	F4	6	9	19	20					
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8								
ADDD F6	F8	F2	13	14	16						
<u>Functional</u>	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Tim	e Nam	e	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
	Mult	1	No								
	Mult2	2	No								
	Add		Yes	Add	F6	F8	F2			Yes	Yes
	Divid	le	Yes	Div	F10	F0	F6		78.20	Yes	Yes
Register re	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
20		FU				Add		Divide			

Instruction s	status			Read	Execut	ic Write	,				
Instruction	j	k	Issue	operan	d comple	et Resu	<u>l</u> t				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8	-				
MULT F0	F2	F4	6	9	19	20					
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8	21							
ADDD F6	F8	F2	13	14	16						
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	е	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	er	No								
	Mult'	1	No								
	Mult2	2	No								
	Add		Yes	Add	F6	F8	F2			Yes	Yes
	Divid	le	Yes	Div	F10	F0	F6		14.16	Yes	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
21		FU				Add		Divide			

Instruction s	status			Read	Execut	ic Write					
Instruction	j	k	Issue	operand	d <mark>comple</mark>	et Resu	<u>l</u> t				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8					
MULT FO	F2	F4	6	9	19	20					
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8	21			7				
ADDD F6	F8	F2	13	14	16	22					
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	ie	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	jer	No								
	Mult	1	No								
	Multa	2	No								
	Add		No								
40	Divid	de	Yes	Div	F10	F0	F6	250		Yes	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
22		FU						Divide			

Instruction s	status			Read	Execut	ic Write					
Instruction	j	k	Issue	operand	d comple	et Resu	lt				
LD F6	34+	R2	1	2	3	4					
LD F2	45+	R3	5	6	7	8					
MULT FO	F2	F4	6	9	19	20					
SUBD F8	F6	F2	7	9	11	12					
DIVD F10	F0	F6	8	21	61						
ADDD F6	F8	F2	13	14	16	22	7.00				
Functional u	unit sta	atus			dest	S1	S2	FU for j	FU for k	Fj?	Fk?
Time	Nam	ie	Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	Integ	jer	No								
	Mult	1	No								
	Mult	2	No								
	Add		No								
0	Divid	de	Yes	Div	F10	F0	F6			Yes	Yes
Register res	sult sta	atus									
Clock			F0	F2	F4	F6	F8	F10	F12		F30
61		FU						Divide			

	Read	Execut	ic Write)				
Issue	operan	d comple	et Resu	ılt				
1	2	3	4					
5	6	7	8					
6	9	19	20					
7	9	11	12					
8	21	61	62					
13	14	16	22					
		dest	S1	S2	FU for j	FU for F	k Fj?	Fk?
Busy	Ор	Fi	Fj	Fk	Qj	Qk	Rj	Rk
No								
No								
No								
No								
No								
F0	F2	F4	F6	F8	F10	F12		F30
	1 5 6 7 8 13 Mo No	Issue operan 1	Issue operand completed 1	Issue operand complet Results 1	Issue operand complet Result			

CDC 6600 Scoreboard

- Limitations of 6600 scoreboard:
 - No forwarding hardware
 - Small number of functional units (structural hazards), especially integer/load store units
 - Do not issue on structural hazards (Prevent WAW hazards)
 - · Wait for RAW and WAR hazards