## Solution to Homework 10, CpE 313 - Fall 2004

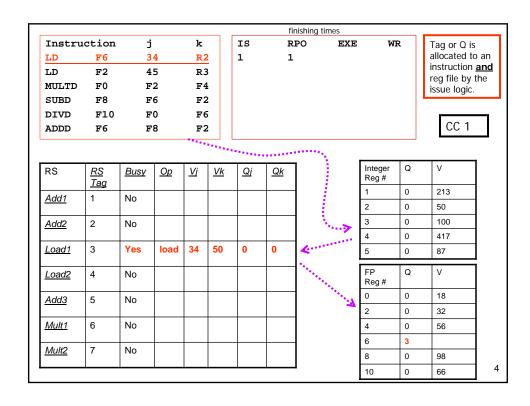
1

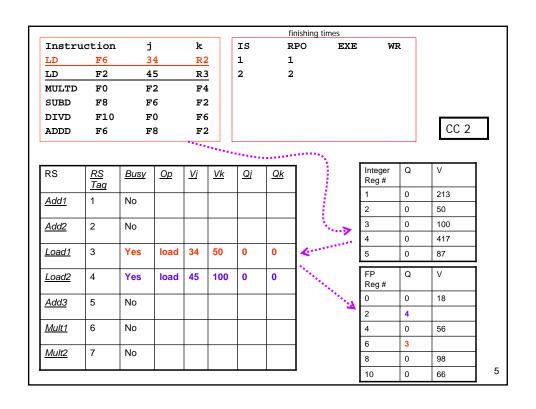
## Solution to Homework 10 - Question 1

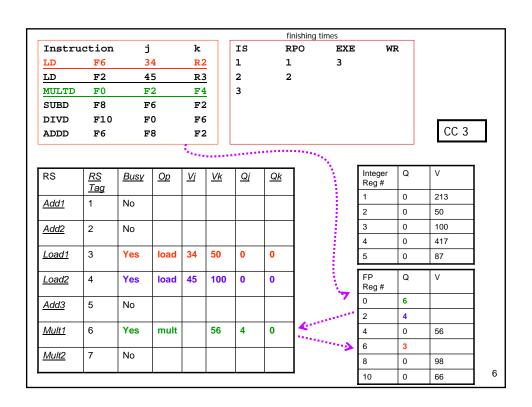
- Please complete the Tomasulo's algorithm work sheets in this handout for:
- 1) CC 1--CC 8,
  - 2) CC 10--CC11,
  - 3) CC15--CC16, and
  - 4) CC56--CC57.
- This HW is due in class on Tuesday Nov 2, 2004.

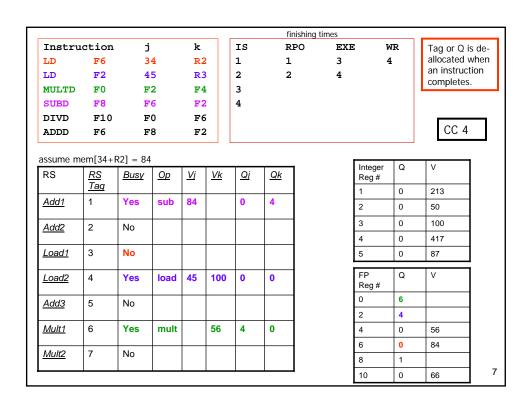
#### Homework 10 - Question 1

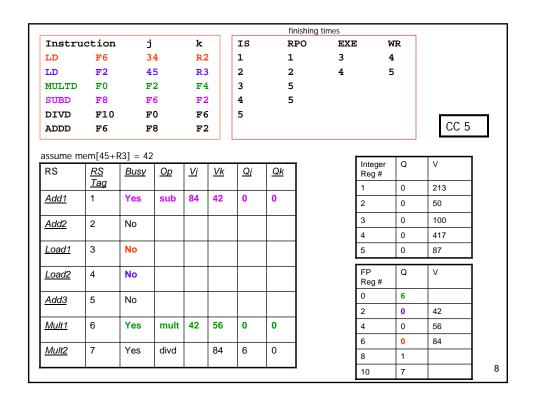
- assume
  - floating point add/sub take 2 cycles of execution
  - floating point mult takes 10 cycles of execution
  - load takes 2 cycles of **execution**
  - floating point divide takes 40 cycles of execution
  - three FP adder units, two mult units (mult also does divide)
  - two load units (store units not discussed in this example)
- in following slides,
  - Vj = value of source operand 1
  - Vk = value of source operand 2
  - Qj = for a pending op, this is # of the RS which will produce Vj
  - Qk = for a pending op, this is # of the RS which will produce Vk

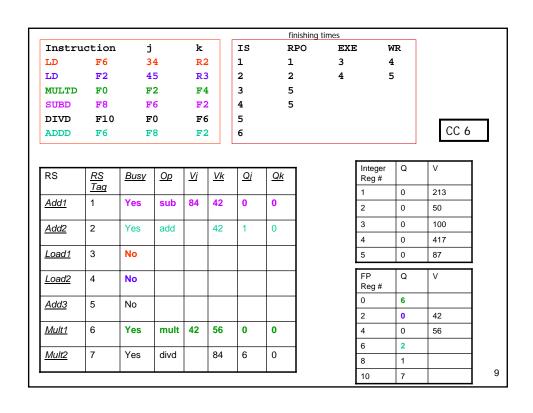


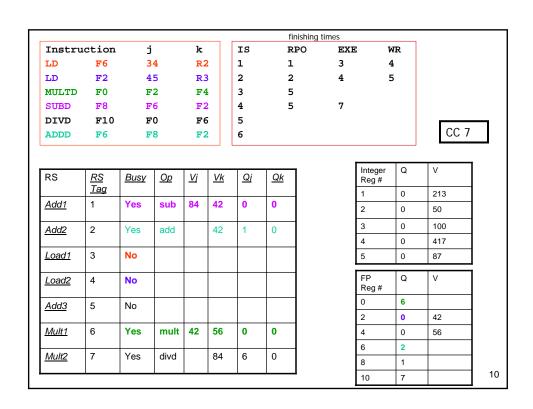


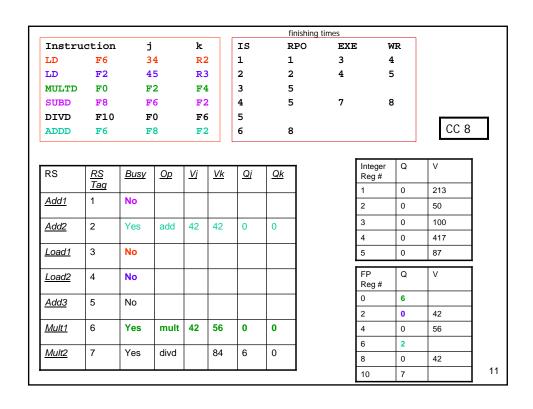




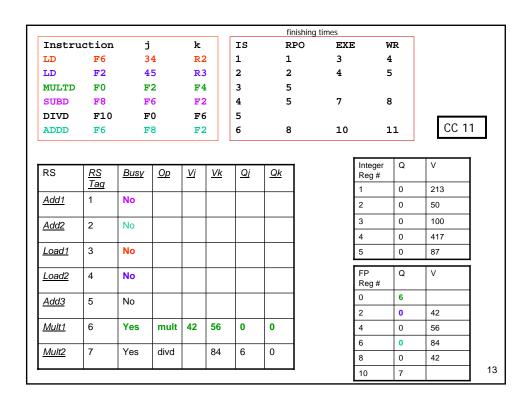




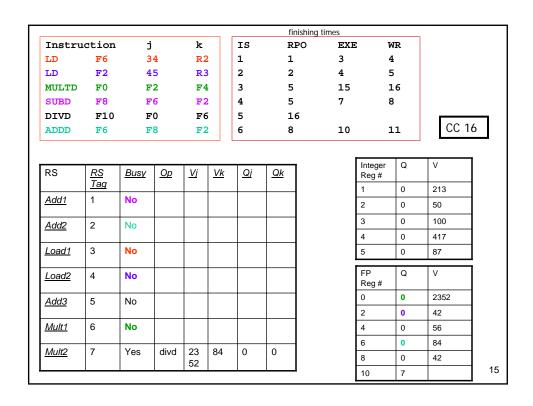




							finis	hing times				
Instru	ction	j		k		IS	RPO	) EXE	S V	VR.		
LD	F6	3	4	R	2	1	1	3	4	1		
LD	F2	4	5	R.	3	2	2	4	5	5		
MULTD	F0	F	2	F4	1	3	5					
SUBD	F8	F	6	F	2	4	5	7	8	3		
DIVD	F10	F	0	F	5	5						_
ADDD	F6	F	8	F	2	6	8	10			CC 1	10
RS	RS	Busy	Ор	<u>Vj</u>	Vk	Qį	Qk		Integer Reg #	Q	V	1
	Tag								1	0	213	1
Add1	1	No							2	0	50	1
Add2	2	Yes	add	42	42	0	0		3	0	100	1
									4	0	417	1
Load1	3	No							5	0	87	]
Load2	4	No							FP Reg#	Q	V	]
Add3	5	No							0	6		1
									2	0	42	1
Mult1	6	Yes	mult	42	56	0	0		4	0	56	]
Multo	7	Yes	divd		84	6	0		6	2		1
Mult2	'	res	aiva		04	٥			8	0	42	
									10	7		7



							finis	hing times				
Instru	ction	j		k		IS	RPO	) EXE		VR.		
LD	F6	3	4	R2	2	1	1	3	4	4		
LD	F2	4	5	R3	3	2	2	4		5		
MULTD	F0	F	2	F4	Ł	3	5	15				
SUBD	F8	F	6	F2	2	4	5	7	8	3		
DIVD	F10	F	0	F	5	5						_
ADDD	F6	F	8	F2	2	6	8	10		11	CC 1	5
RS	RS	Busy	Ор	<u>Vj</u>	Vk	Qį	Qk		Integer Reg #	Q	V	1
	Tag								1	0	213	1
<u>Add1</u>	1	No							2	0	50	1
Add2	2	No							3	0	100	1
7 IGGE	_	140							4	0	417	1
Load1	3	No							5	0	87	]
Load2	4	No							FP Reg#	Q	V	]
Add3	5	No							0	6		1
									2	0	42	1
<u>Mult1</u>	6	Yes	mult	42	56	0	0		4	0	56	
14.40	7	Vas	alis (al		0.4	-	+		6	0	84	
Mult2	7	Yes	divd		84	6	0		8	0	42	1
									10	7		1



							fir	nishing 1	times				
Instru	ction	j		k		IS	R	PO	EXE	V	VR.		
LD	F6	3	4	R2	2	1	1		3	4	Ł		
LD	F2	4	5	R3	3	2	2		4	5	5		
MULTD	F0	F	2	F4	Ł	3	5		15	1	L6		
SUBD	F8	F	6	F2	2	4	5		7	8	3		
DIVD	F10	F	0	F	5	5	1	6	56			22.5	_
ADDD	F6	F	8	F2	2	6	8		10	1	L <b>1</b>	CC 5	6
RS	<u>RS</u>	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	I		Integer Reg #	Q	V	]
	<u>Tag</u>									1	0	213	1
<u>Add1</u>	1	No								2	0	50	1
Add2	2	No						1		3	0	100	
										4	0	417	1
<u>Load1</u>	3	No								5	0	87	]
Load2	4	No								FP Reg#	Q	V	]
Add3	5	No						1		0	0	2352	
										2	0	42	
<u>Mult1</u>	6	No								4	0	56	
Mulio	7	Vas	مان رما	22	84	+-	10			6	0	84	
Mult2	'	Yes	divd	23 52	84	0	0			8	0	42	
				02				ı		10	7		1

							fir	nishing 1	times				
Instru	ction	j		k		IS	R	PO	EXE	: V	/R		
LD	F6	3	4	R2	2	1	1		3	4	Ł		
LD	F2	4	5	R3	3	2	2		4	5	5		
MULTD	F0	F	2	F4	l l	3	5		15	1	.6		
SUBD	F8	F	6	F2	2	4	5		7	ε	3		
DIVD	F10	F	0	Fθ	5	5	1	6	56	5	57		
ADDD	F6	F	8	F2	2	6	8		10	1	.1	CC 5	7
RS	<u>RS</u>	<u>Busy</u>	<u>Op</u>	<u>Vi</u>	<u>Vk</u>	Qį	<u>Qk</u>	1		Integer Reg #	Q	V	1
	<u>Tag</u>									1	0	213	1
Add1	1	No								2	0	50	1
Add2	2	No						i		3	0	100	1
										4	0	417	1
<u>Load1</u>	3	No								5	0	87	]
Load2	4	No								FP Reg#	Q	V	]
Add3	5	No					1	1		0	0	2352	
										2	0	42	
<u>Mult1</u>	6	No								4	0	56	
Multo	7	No		+				ł		6	0	84	
Mult2	'	INO								8	0	42	
										10	0	28	1

## Solution to Homework 10 - Question 2

- assume
  - add/sub take 2 cycles of **execution**
  - mult takes 10 cycles of **execution**
  - load takes 2 cycles of **execution**
  - store takes 2 cycles of **execution**
  - two adder units, two mult units
  - two load units, one store unit
- assume
  - mem[34+R2] = 4
  - mem[45+R2] = 5

# **Tomasulo Example – Notation**

- in following slides,
  - Vj = value of source operand 1
  - Vk = value of source operand 2
  - Qj = for a pending op, this is # of the RS which will produce Vj
  - Qk = for a pending op, this is # of the RS which will produce Vk

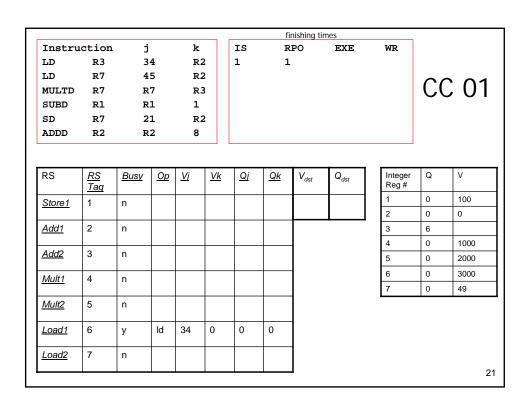
19

Instruc	tion	j	k
LD	R3	34	R2
LD	R7	45	R2
MULTD	R7	R7	R3
SUBD	R1	R1	1
SD	R7	21	R2
ADDD	R2	R2	8

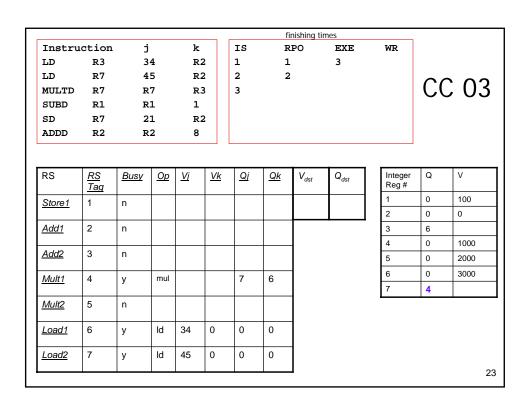


RS	<u>RS</u> <u>Tag</u>	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>
Store1	1	n							
Add1	2	n							
Add2	3	n							
<u>Mult1</u>	4	n							
Mult2	5	n							
Load1	6	n							

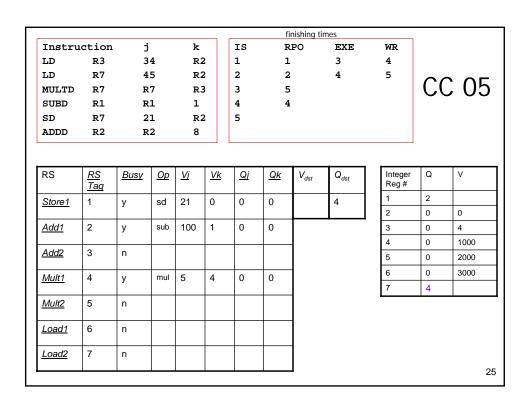
Integer Reg #	Q	V
1	0	100
2	0	0
3	0	31
4	0	1000
5	0	2000
6	0	3000
7	0	49



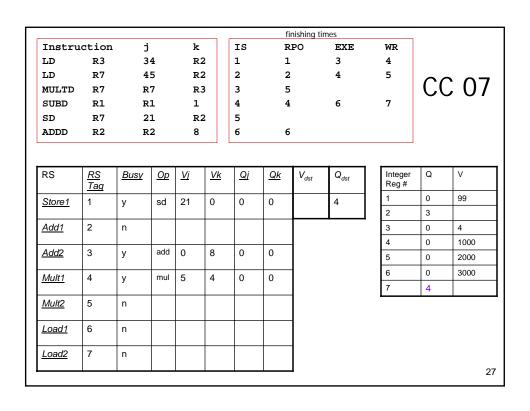
							fi	nishing ti	mes			
Instru	ction	j		k		IS	R	PO	EXE	WR		
LD	R3	3-	4	R	2	1	1					
LD	R7	4.	5	R	2	2	2				$\sim$	
MULTD	R7	R	7	R	3						C(	02
SUBD	R1	R	1	1								
SD	R7	2	1	R	2							
ADDD	R2	R	2	8								
RS	<u>RS</u> <u>Tag</u>	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V
Store1	1	n								1	0	100
										2	0	0
Add1	2	n								3	6	
	_							-		4	0	1000
<u>Add2</u>	3	n								5	0	2000
Mult1	4	n						1		6	0	3000
<u> </u>	-	''								7	7	
Mult2	5	n										
Load1	6	у	ld	34	0	0	0	1				
Load2	7	у	ld	45	0	0	0	1				
								1				



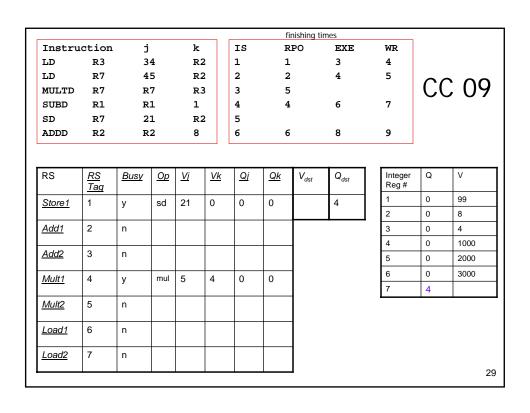
							fi	nishing ti	mes				
Instru	ction	j		k		IS	R	PO	EXE	WR			
LD	R3	3	4	R2	2	1	1		3	4			
LD	R7	4	5	R2	2	2	2		4		00		
MULTD	R7	R	7	R3	3	3						04	
SUBD	R1	R	1	1		4	4					_	
SD	R7	2	1	R2	2								
ADDD	R2	R	2	8									
20				\ <i>n</i>	T . #	101		1.,	1. 1		T <sub>0</sub>	T.,	
RS	<u>RS</u> <u>Tag</u>	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V	
Store1	1	n								1	2		
			. aub	eub							2	0	0
<u>Add1</u>	2	У	sub	100	1	0	0			3	0	4	
A -1-10	_	_						1		4	0	1000	
Add2	3	n								5	0	2000	
Mult1	4	У	mul		4	7	0	1		6	0	3000	
										7	4		
Mult2	5	n											
<u>Load1</u>	6	n						1					
Load2	7	у	ld	45	0	0	0	1					
								J					



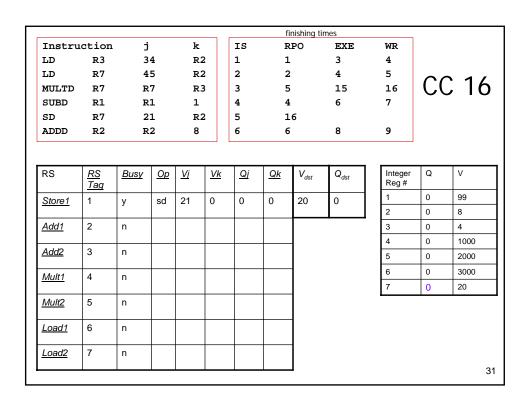
							fi	inishing ti	mes			
Instru	ction	j		k		IS	R	PO	EXE	WR		
LD	R3	3	4	R2	2	1	1	-	3	4		
LD	R7	4	5	R2	2	2	2	2	4	5	00	0/
MULTD	R7	R	7	R3	3	3	5	5				06
SUBD	R1	R	1	1		4	4	Ŀ	6			
SD	R7	2	1	R2	2	5						
ADDD	R2	R	2	8		6	6	i				
RS	<u>RS</u> <u>Tag</u>	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V
Store1	1	у	sd	21	0	0	0		4	1	2	
										2	3	
Add1	2	У	sub	100	1	0	0			3	0	4
4 4 10				_	-	+	+	1		4	0	1000
Add2	3	У	add	0	8	0	0			5	0	2000
Mult1	4	у	mul	5	4	0	0	1		6	0	3000
<u> </u>	-	'								7	4	
Mult2	5	n										
<u>Load1</u>	6	n										
Load2	7	n						1				
												2



							fi	nishing ti	mes			
Instru	ction	j		k		IS	R	PO	EXE	WR		
LD	R3	3	4	R2	2	1	1	•	3	4		
LD	R7	4	5	R2	2	2	2	1	4	5	00	
MULTD	R7	R	7	R3	3	3	5	i				30
SUBD	R1	R	1	1		4	4	:	6	7		
SD	R7	2	1	R2	2	5						
ADDD	R2	R	2	8		6	6	i	8			
RS	RS	Busy	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	Qį	Qk	$V_{dst}$	Q <sub>dst</sub>	Integer	Q	Ιν
	<u>Tag</u>	Basy		<u>"</u>	<u> </u>	<u> </u>	<u> </u>	dst	dst	Reg #		
Store1	1	У	sd	21	0	0	0		4	1	0	99
								<b>-</b>		2	3	
<u>Add1</u>	2	n								3	0	4
Add2	3	у	add	0	8	0	0	1		4	0	1000
Auuz	١	, ,	""	"	"	"	0			5	0	2000
Mult1	4	у	mul	5	4	0	0			6	0	3000
								1		7	4	
Mult2	5	n										
Load1	6	n						1				
Load2	7	n						1				
						1		1				2



							f	inishing ti	mes			
Instru	ction	j		k		IS	F	RPO	EXE	WR		
LD	R3	3	4	R	2	1	1	-	3	4		
LD	R7	4	5	R	2	2	2	2	4	5	00	
MULTD	R7	R	7	R	3	3	5	5	15			15
SUBD	R1	R	1	1		4	4	Ŀ	6	7		
SD	R7	2	1	R	2	5						
ADDD	R2	R	2	8		6	6	5	8	9		
20					T . "	101		1,,	1.		T <sub>0</sub>	1,,
RS	<u>RS</u> <u>Tag</u>	<u>Busy</u>	<u>Op</u>	<u>Vi</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V
Store1	1	у	sd	21	0	0	0		4	1	0	99
								<u> </u>		2	0	8
<u>Add1</u>	2	n						1		3	0	4
4 -1-10	0	_				+		┨		4	0	1000
Add2	3	n						1		5	0	2000
Mult1	4	у	mul	5	4	0	0	1		6	0	3000
								_		7	4	
Mult2	5	n										
Load1	6	n										
Load2	7	n						1				
								_				



		finishing times											
Instruction		j		k		IS	RPO		EXE	WR			
LD	R3	3	4	R2	2	1	1	L	3	4			
LD	R7	4	5	R2	2	2	2	2	4	5	00	40	
MULTD	R7	R	R7		3	3		5	15	16	CC 18		
SUBD	R1	R1 21 R2		1		4	4		6	7			
SD	R7			R2	2	5	16 6		18 8				
ADDD	R2			8		6				9			
RS	<u>RS</u> Tag	<u>Busy</u>	<u>Op</u>	<u>Vi</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V	
Store1	1	у	sd	21	0	0	0	20	0	1	0	99	
010707	ľ	,	Jou	- '					ľ	2	0	8	
Add1	2	n								3	0	4	
								4		4	0	1000	
Add2	3	n								5	0	2000	
Mult1	4	n						┪		6	0	3000	
<u>warr</u>		''								7	0	20	
Mult2	5	n											
<u>Load1</u>	6	n											
Load2	7	n						1					
								J				3	

					finishing times								
Instruction R3		j 34		k R2		IS 1	RPO 1		EXE 3	WR			
										4			
LD	R7	4	5	R2	2	2	2		4	5	00	. 10	
MULTD	ULTD R7		R7		3	3	5		15	16	CC 19		
SUBD	R1	1 R1		1		4	4 16		6 18	7			
SD	R7 21		1	R2	2	5				19			
ADDD	R2	R	2	8		6	6		8	9			
RS	<u>RS</u> Tag	<u>Busy</u>	<u>Op</u>	<u>Vj</u>	<u>Vk</u>	<u>Qj</u>	<u>Qk</u>	V <sub>dst</sub>	Q <sub>dst</sub>	Integer Reg #	Q	V	
Store1	1	n								1	0	99	
								<u> </u>		2	0	8	
Add1	2	n								3	0	4	
	_							-		4	0	1000	
Add2	3	n								5	0	2000	
Mult1	4	n						1		6	0	3000	
										7	0	20	
Mult2	5	n								-			
<u>Load1</u>	6	n											
Load2	7	n						1					
								J					