op	R1 = R2 op R3										
op	R4 = R1 op R2										
		IF	ID	Reg	Ex	Mem	WB				Muxes F1 (
			IF	ID	Reg	Ex	Mem	WB			
op	R1 = R2 op R3										
op with CCR dependence											
		IF	ID	Reg	Ex	Mem	WB				forward x
			IF	ID	Reg	Ex	Mem	WB			
op	R1 = R2 op R3										
x											
op with CCR dependence											
		IF	ID	Reg	Ex	Mem	WB				forward y
			x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		
op	R1 = R2 op R3										
X	X										
op	R4 = R1 op R2										
		IF	ID	Reg	Ex	Mem	WB				Muxes F1 (
				IF	ID	Reg	Ex	Mem	WB		
R7 =R1 (unconditional operation) R2		IF	ID	Reg	Ex	Mem	WB				
			x	x	x	x	x	x			
				x	x	x	x	x	x		
					x	x	x	x	x	x	
						IF	ID	Reg	Ex	Mem	WB
R3 = R1 op R2		IF	ID	Reg	Ex	Mem	WB				
R7 = R1 op(conditional) R2			IF	ID	Reg	Ex	Mem	WB			
(next)				x	x	x	x	x	x		
					x	x	x	x	x	x	
				Case 1:		IF	ID	Reg	Ex	Mem	WB

or F2 use a

into ccr

into ccr

or F2 use b Unless X = another op conflict, in which case use a.
(closest conflict)

Forward a to PC

(if the first instruction did not set CCR)

				Case 2:		x	x	x	x	x	x
							IF	ID	Reg	Ex	Mem
op	R1 = R2 op R3										
load	R4 = mem(R1 +Im6)										
		IF	ID	Reg	Ex	Mem	WB				Muxes F1
			IF	ID	Reg	Ex	Mem	WB			
op	R1 = R2 op R3										
X	X										
load	R4 = mem(R1 +Im6)										
		IF	ID	Reg	Ex	Mem	WB				Muxes F1
				IF	ID	Reg	Ex	Mem	WB		
load	R4 = mem(R1 +Im6)										
op	R2 = R4 op R3										
		IF	ID	Reg	Ex	Mem	WB				Muxes F1
	stall		x	x	x	x	x				
				IF	ID	Reg	Ex	Mem	WB		
load	R4 = mem(R1 +Im6)										
load	R3 = mem(R4 +Im6)										
			IF	ID	Reg	Ex	Mem	WB			Mux F1 us
		stall		x	x	x	x	x			
					IF	ID	Reg	Ex	Mem	WB	
op	R1 = R2 op R3										
store	mem(R4+Im6) = R1										
		IF	ID	Reg	Ex	Mem	WB				
			IF	ID	Reg	Ex	Mem	WB			Mux F1 us

WB (if the first instruction did set CCR) (forward from a)

or F2 use a

or F2 use b

or F2 use c

load		load	
x	and	x	use just forward, stall is implemented by the intermediate instruction
op		load	

ses c

es b

op	R1 = R2 op R3										
store	mem(R1+Im6) = R2										
		IF	ID	Reg	Ex	Mem	WB				
			IF	ID	Reg	Ex	Mem	WB			Mux F1 us
op	R1 = R2 op R3										
store	mem(R1+Im6) = R1										
		IF	ID	Reg	Ex	Mem	WB				
			IF	ID	Reg	Ex	Mem	WB			Mux F1 us
											Mux F3 us
op	R1 = R2 op R3										
x	x										
store	mem(R1+Im6) = R2										
		IF	ID	Reg	Ex	Mem	WB				
			x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		Mux F1 us
op	R1 = R2 op R3										
x	x										
store	mem(R1+Im6) = R1										
		IF	ID	Reg	Ex	Mem	WB				
			x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		
load	R2 = mem(R1+Im6)										
store	mem(R3+Im6) = R2										
		IF	ID	Reg	Ex	Mem	WB				
			IF	ID	Reg	Ex	Mem	WB			Mux F3 us

ses a

es a

es c

ses b

Mux F1 uses b

es c

load	R2 = mem(R1+Im6)										
store	mem(R2+Im6) = R3										
		IF	ID	Reg	Ex	Mem	WB				
	stall		x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		Mux F1 us
load	R2 = mem(R1+Im6)										
store	mem(R2+Im6) = R2										
		IF	ID	Reg	Ex	Mem	WB				
	stall		x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		Mux F1 us
After JAL and JLR stalls are always required.followed by a forward											
JAL	R2 = PC+1	IF	ID	Reg	Ex	Mem	WB				
x	x		x	x	x	x	x	x			
				IF	ID	Reg	Ex	Mem	WB		Mux F4 us
JAL	R2 = PC+1	IF	ID	Reg	Ex	Mem	WB				
op	R3 = R2 op R1		x	x	x	x	x	x			Muxes F1
				IF	ID	Reg	Ex	Mem	WB		Mux F4 us
JAL	R2 = PC+1	IF	ID	Reg	Ex	Mem	WB				
load	R3 = mem(R2 +Im6)		x	x	x	x	x	x			Muxes F1
				IF	ID	Reg	Ex	Mem	WB		Mux F4 us
JAL	R2 = PC+1	IF	ID	Reg	Ex	Mem	WB				
store	mem(R2 +Im6) = R3		x	x	x	x	x	x			Muxes F1

es c

es c

es d

or F2 use b

es d

or F2 use b

es d

]

or F2 use b

				IF	ID	Reg	Ex	Mem	WB		Mux F4 us
JAL	R2 = PC+1	IF	ID	Reg	Ex	Mem	WB				
store	mem(R2 +Im6) = R2		x	x	x	x	x	x			Muxes F1
				IF	ID	Reg	Ex	Mem	WB		Mux F4 us
JAL		IF	ID	Reg	Ex	Mem	WB				
op R1 = R2+R7			x	x	x	x	x	x			use g to fo
				IF	ID	Reg	Ex	Mem	WB		
JLR		IF	ID	Reg	Ex	Mem	WB				
			x	x	x	x	x	x			
				x	x	x	x	x	x		
					IF	ID	Reg	Ex	Mem	WB	
Note JLR has no error, if followed by R7 operation											
Beq											
		IF	ID	Reg	Ex	Mem	WB				
			IF	ID	NOP	NOP	NOP	NOP			
				IF	NOP	NOP	NOP	NOP	NOP		
				Shown what happens if branch is taken, else branch not taken, normal with no NOP							
LDW R7		IF	ID	Reg	Ex	Mem	WB				

es d

or F2 use b

es d

orward appropriate value of R7

Mux F4 uses e

flush the first two pipline registers.

NOP = no operation, when IR = 16'b1111000000000000

or LDM 1xxxxxxx			x	x	x	x	x	x			
				x	x	x	x	x	x		
					x	x	x	x	x	x	
						x	x	x	x	x	x
							x	x	x	x	x
							IF	ID	Reg	Ex	Mem
LHI R7,Im6											
	only one stall										
	forward h to be loaded into PC										
LHI R5,Im6		IF	ID	Reg	Ex	Mem	WB				Forward i
R1 R2 op R5			IF	ID	Reg	Ex	Mem	WB			
LHI R5,Im6		IF	ID	Reg	Ex	Mem	WB				Forward j
x											
R1 R2 op R5				IF	ID	Reg	Ex	Mem	WB		

Forward c to be loaded into PC!

x

WB

to the inputs of ALU

to the inputs of ALU