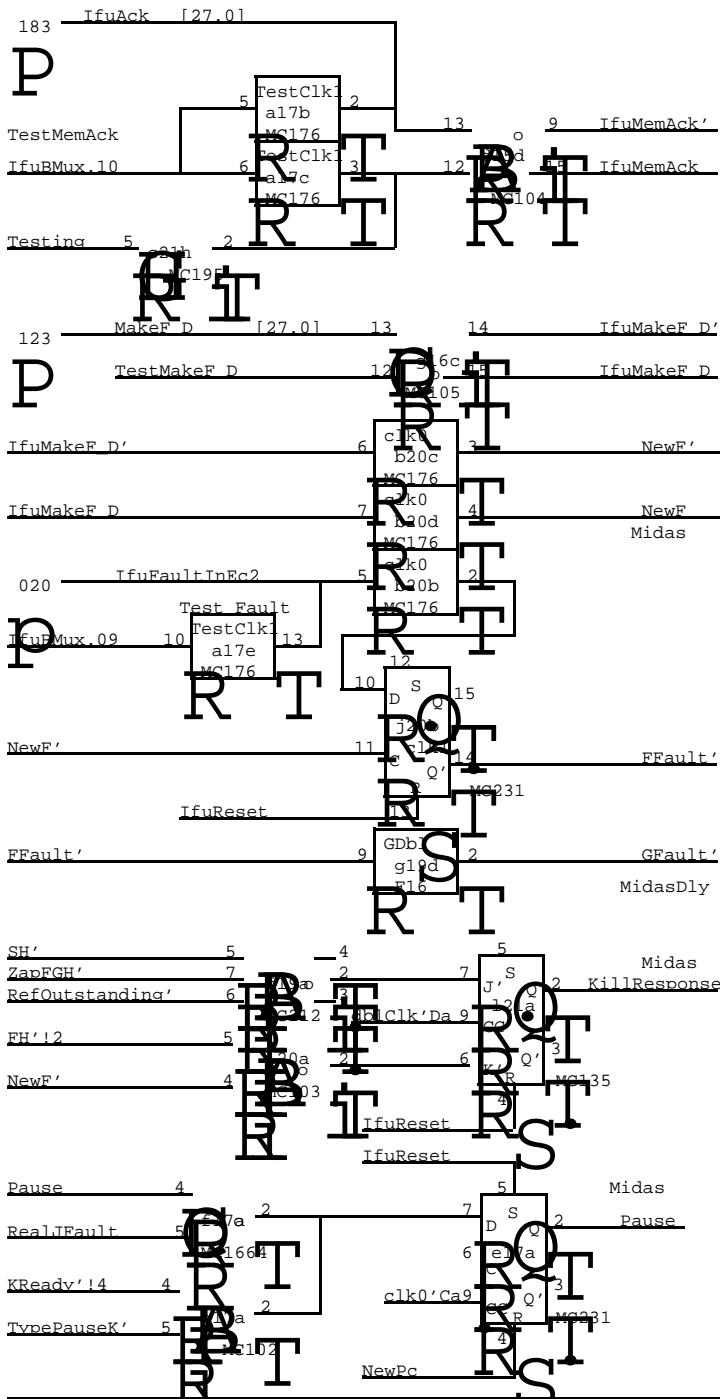


DORADO SCHEMATICS

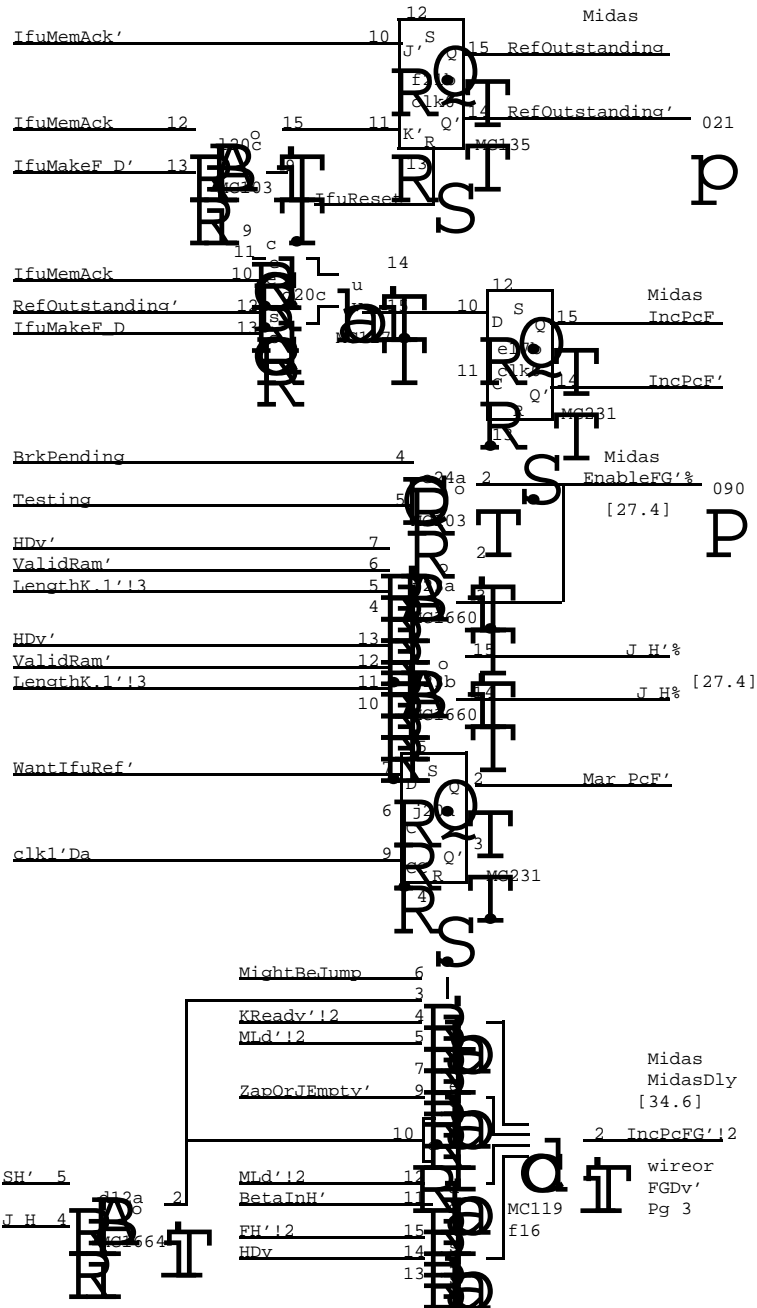
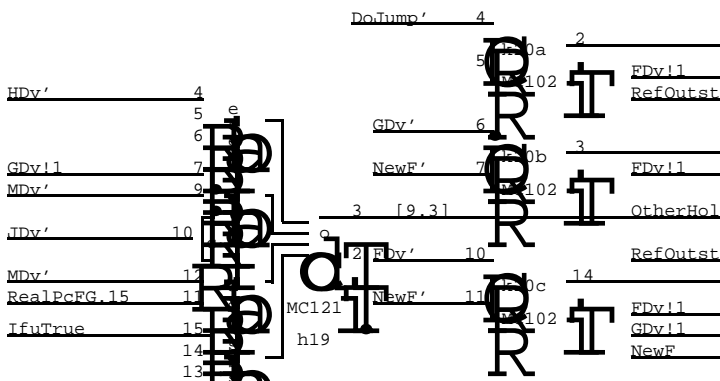
Instruction Fetch Unit

Table of contents

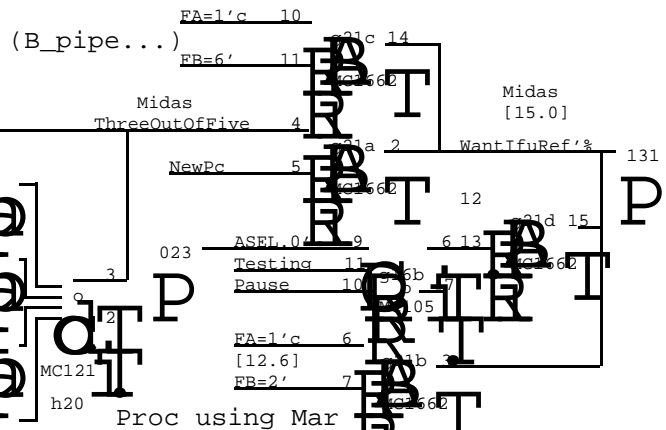
<u>TITLE</u>	<u>Page</u>
Mememory Request Logic	01
IFData, Jumps, and Exceptions	02
Loads and DVs	03
FF Decodes	04
Byte Bit Slice	05
Instruction Bit Slice	06
Instruction Address Bit Slice	07
PC Bit Slice (High byte)	08
PC Bit Slice (Low byte)	09
Pc Pipe Control	10
Chip Control	11
Ifu Ram Control	12
Event Counter A	13
Event Counter B	14
Wakeups/Temperature Sensor	15
Midas control and multiplexers	16
Midas control and multiplexers	17
Clock distribution	18
Layout Upper Left quadrant	19
Layout Upper Right quadrant	20
Layout Lower Left quadrant	21
Layout Lower Right quadrant	22
IF Data Selector Prom	23



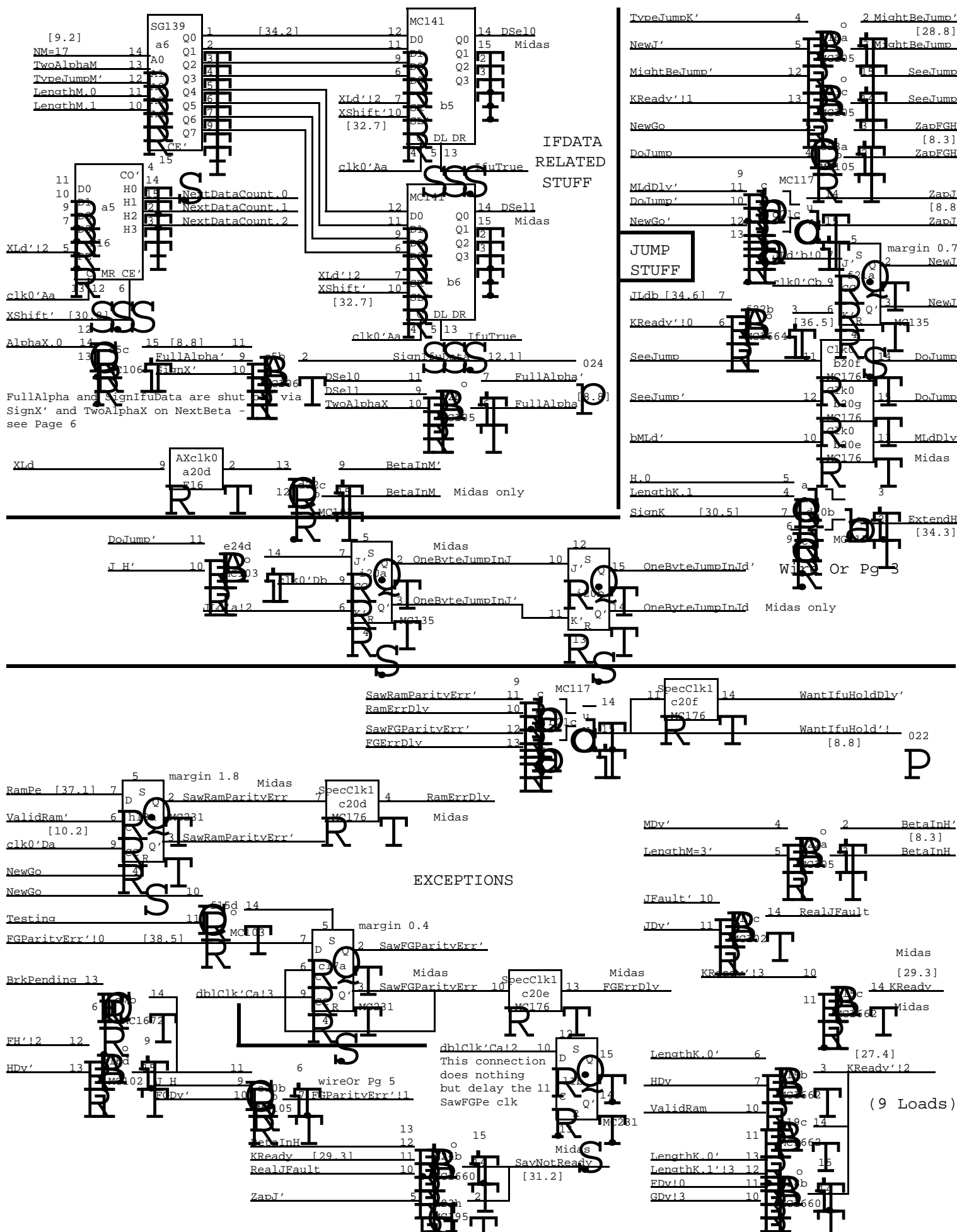
Can't detect the holes created by X_M or AlphaM_H because the Loads aren't ready until after T1, due to Hold

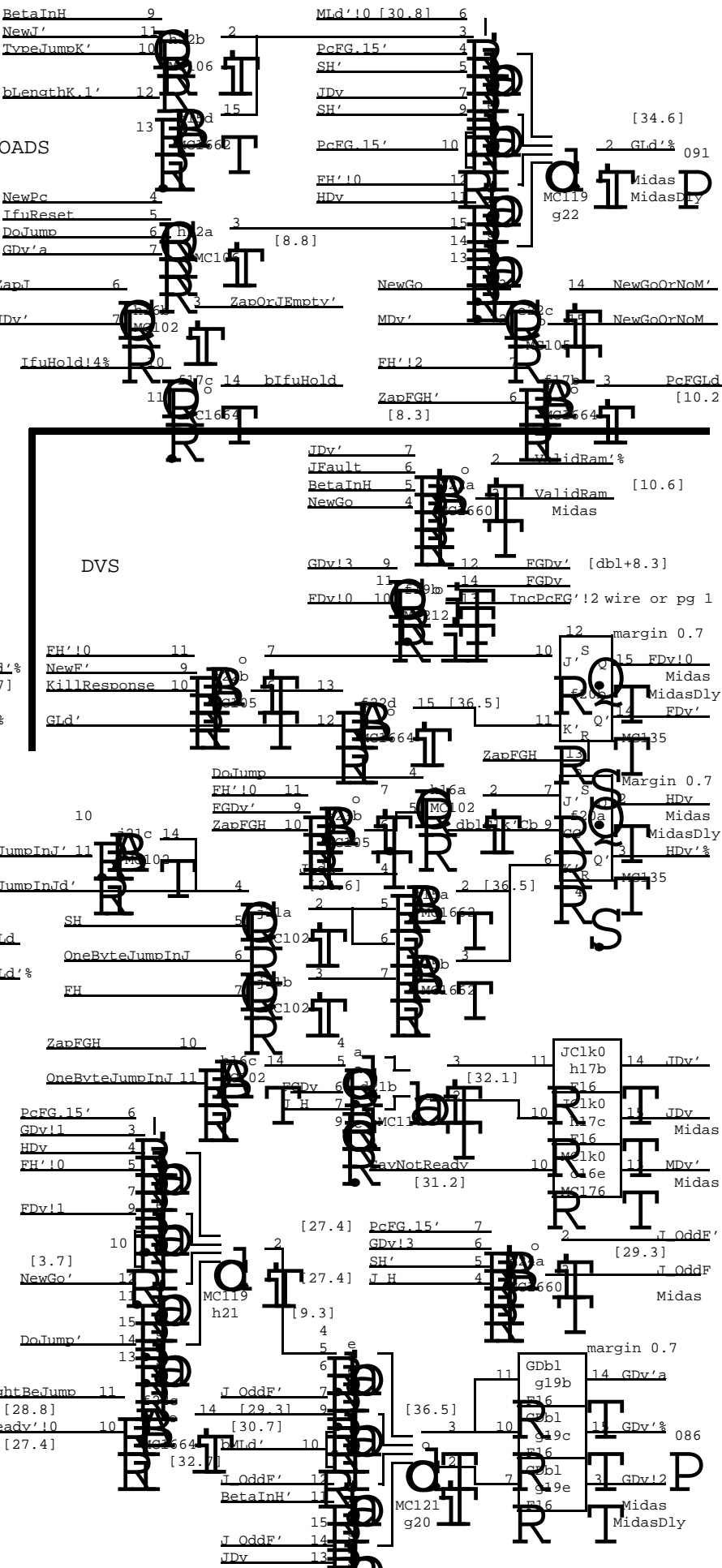
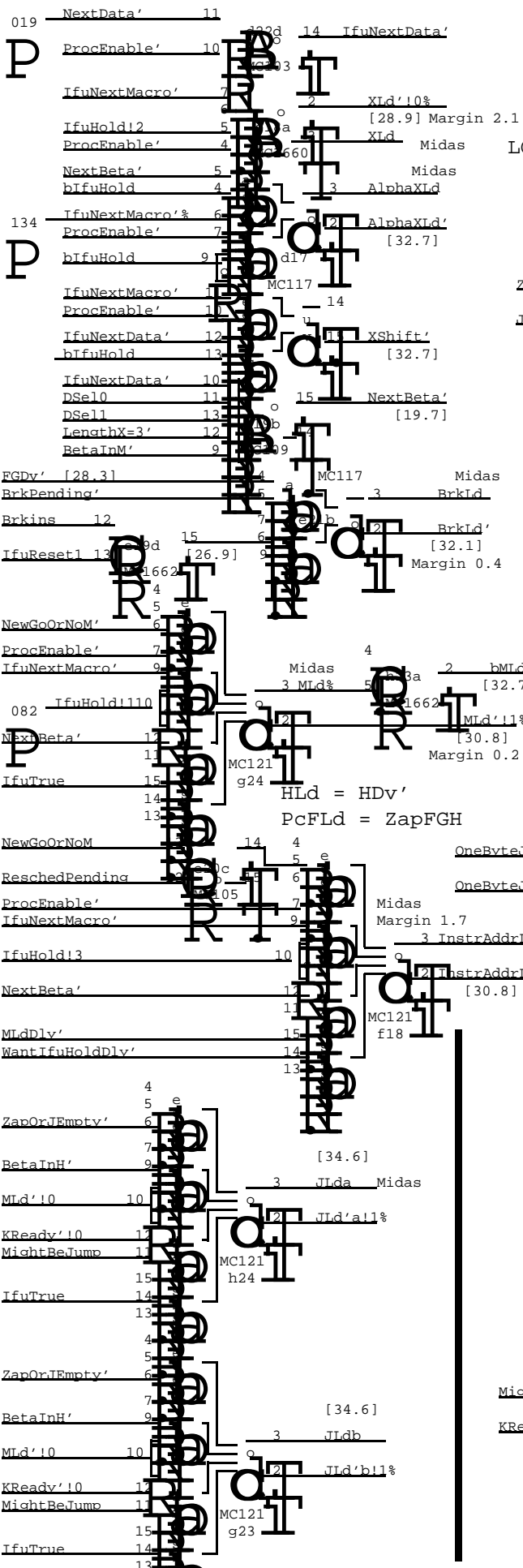


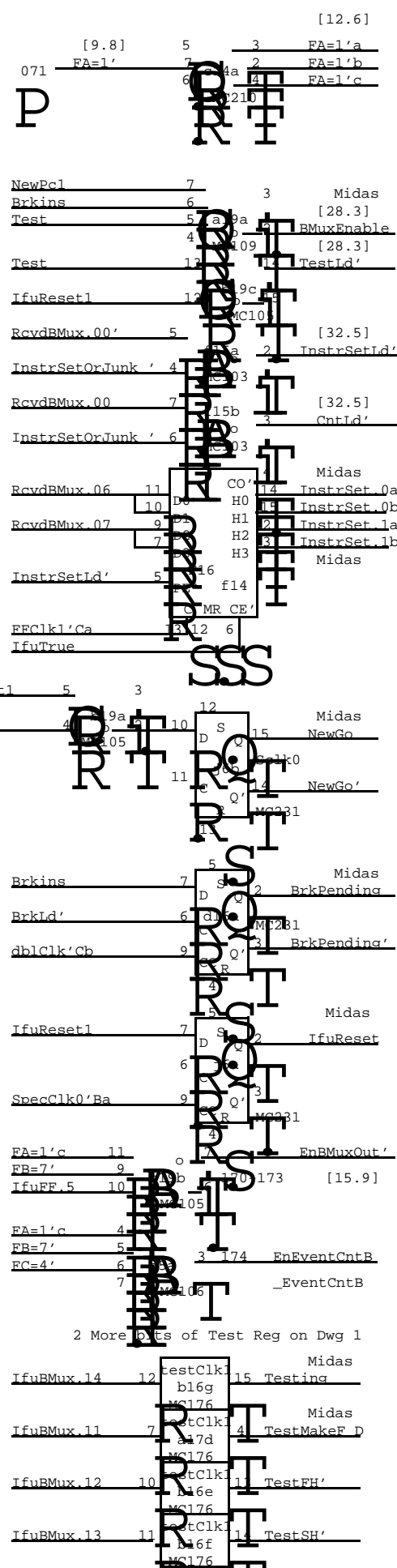
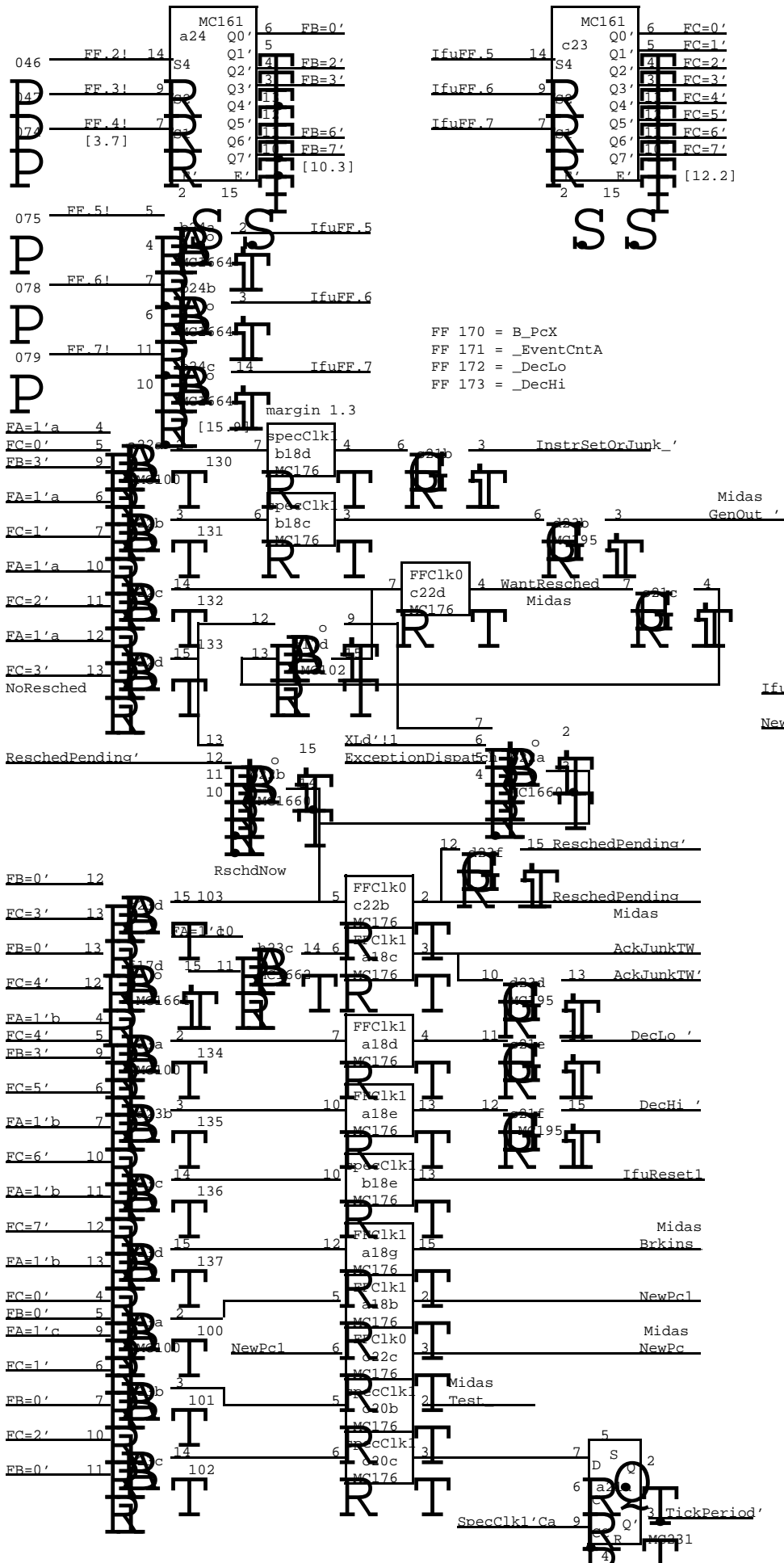
Combination of F valid and G invalid is impossible in First Half

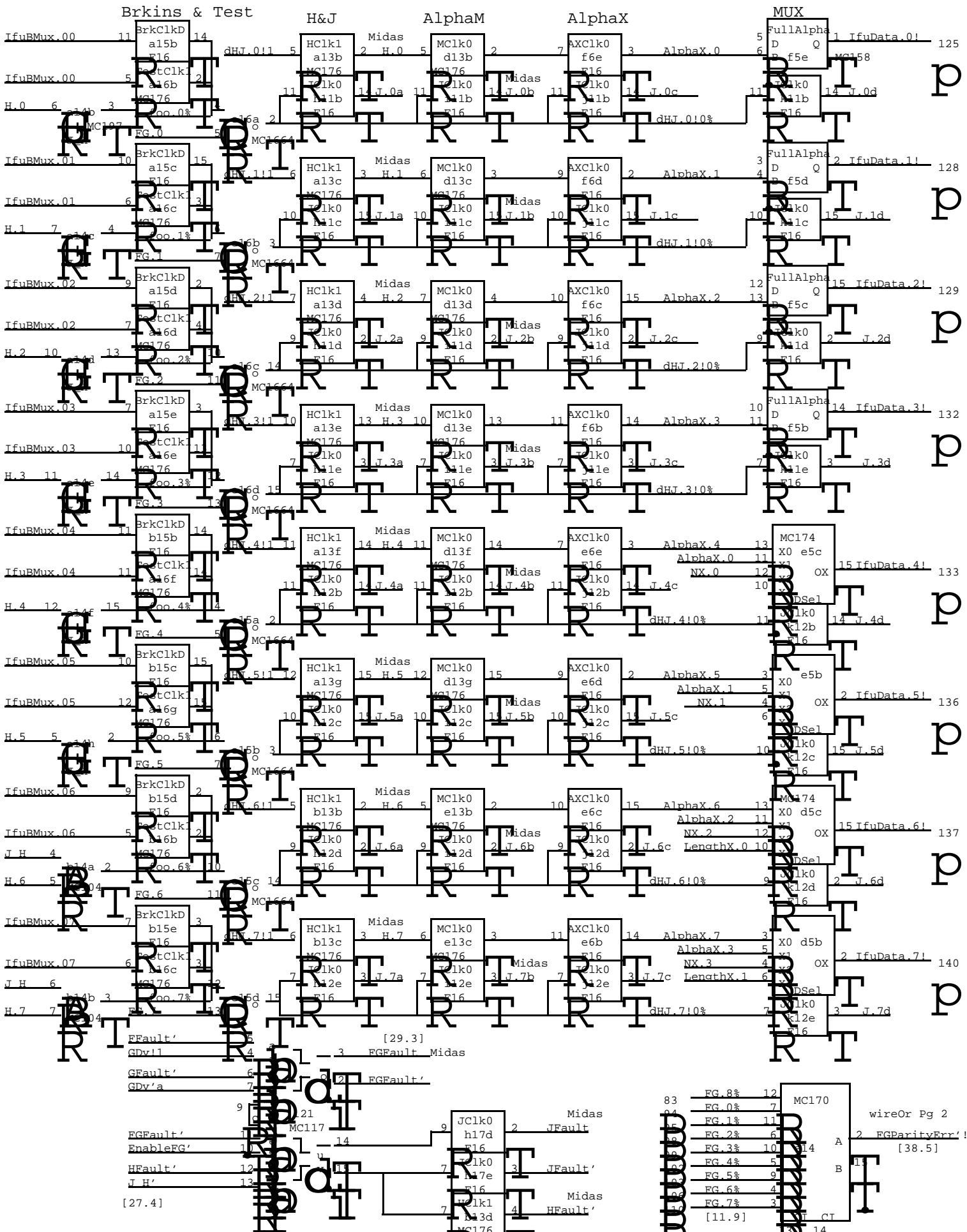


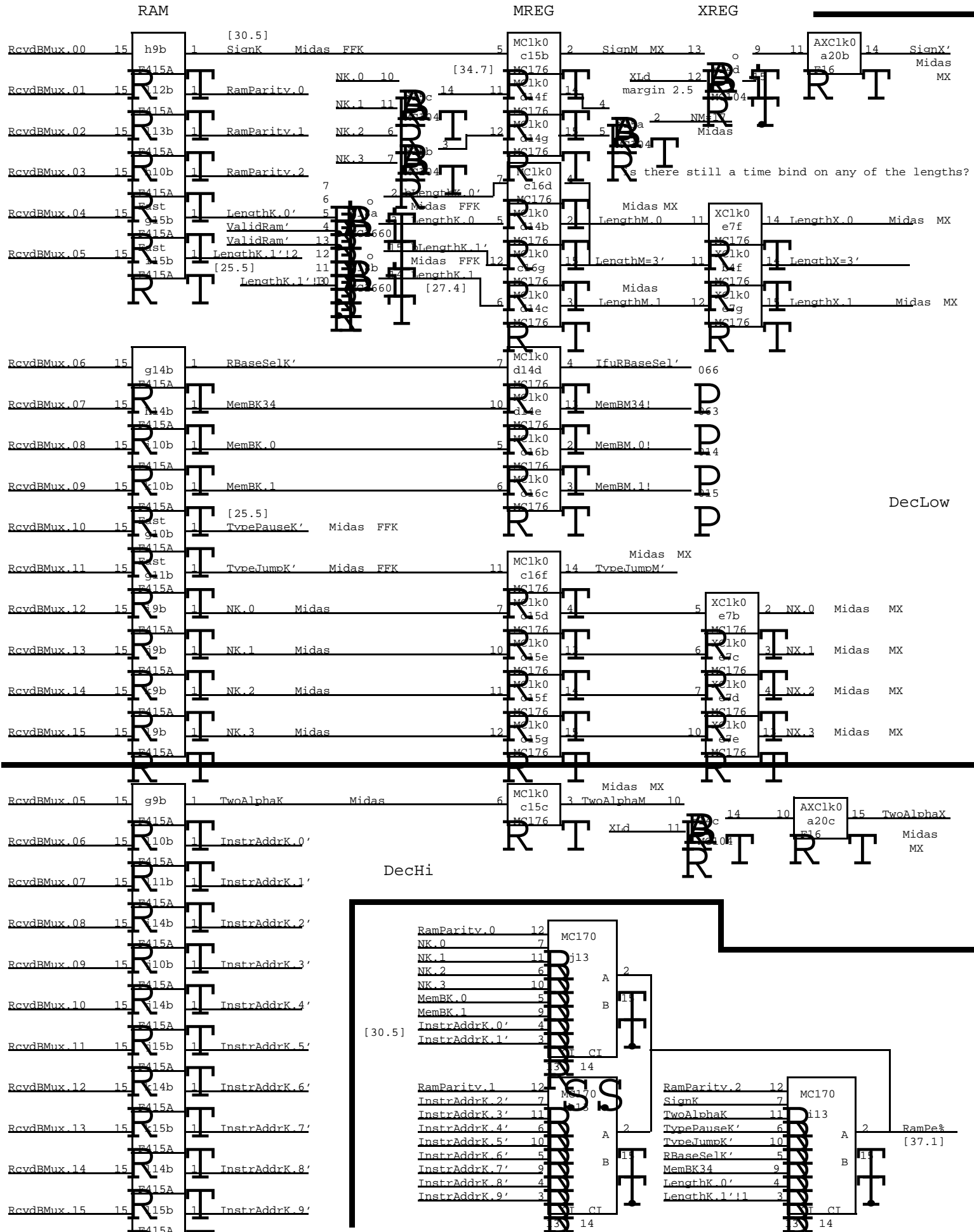
Proc using Mar





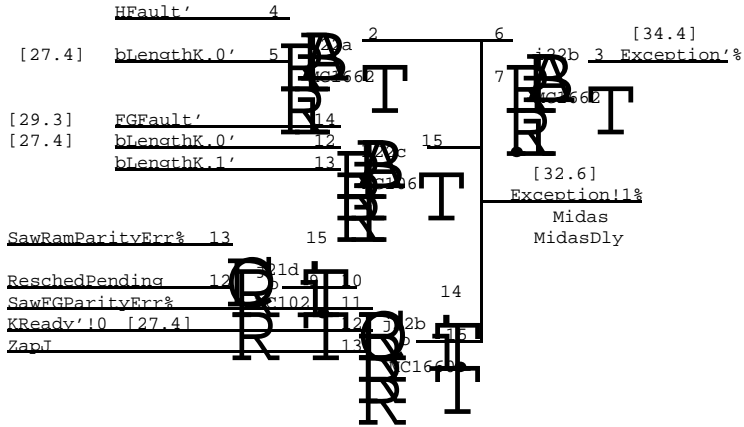




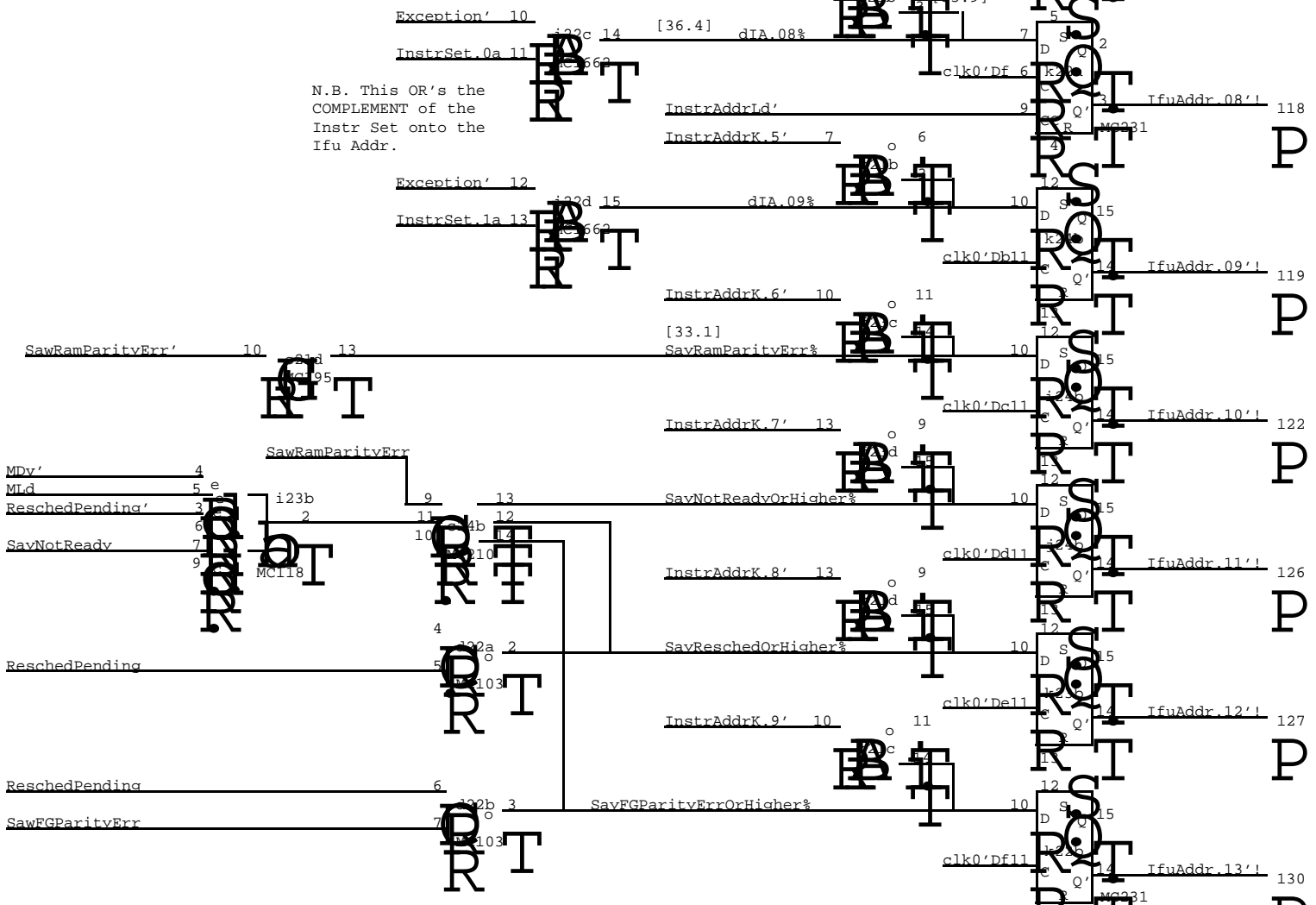


- 0 K Fault low priority
- 4 FG Parity
- 14 Resched
- 34 Not Ready
- 74 Ram Parity high priority

KFault is JFault OR HFault&Length>1
OR FGFault&Length=3



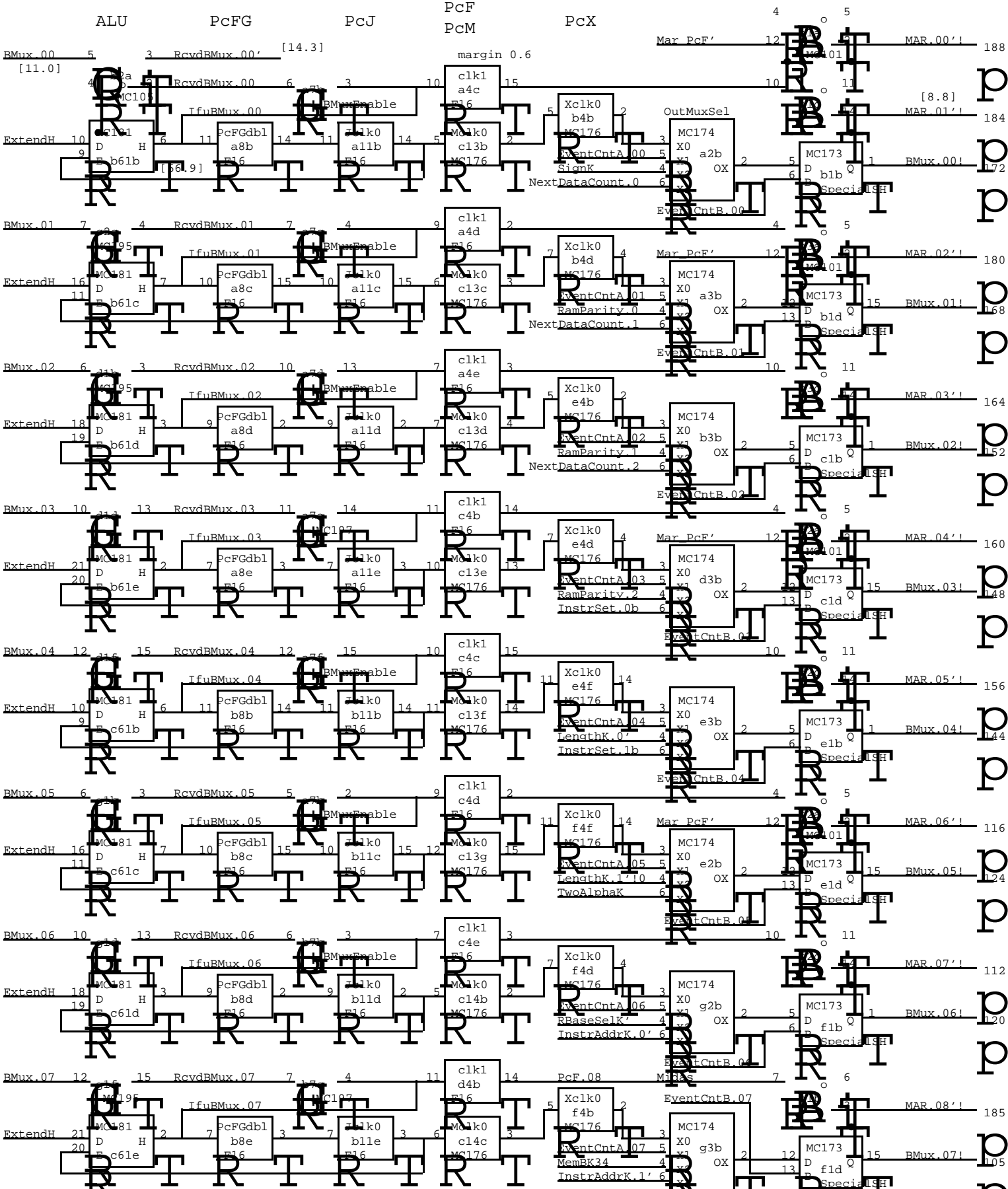
N.B. This OR's the
COMPLEMENT of the
Instr Set onto the
Ifu Addr.



XEROX PARC	Project Dorado	File InstrAddr Bit Slice	Designer S. Ornstein	Rev Cg	Date 8/28/80	Page 07
---------------	-------------------	-----------------------------	-------------------------	-----------	-----------------	------------

THE HIGH ORDER BYTE OF THE PC PIPE

PcJ loads on JldAnd not J_H, increments on Jld
 PcF Loads on ZapFGH and increments on IncPcF
 ALU turns of on BMuxEnable or BrkPending or IfuReset1
 PcFG loads on PcFGLd and increments on IncPcFG



XEROX PARC	Project Dorado	PC Bit Slice(High byte)	File Ifu08.sil	Designer S. Ornstein	Rev Cg	Date 8/23/79	Page 08
---------------	-------------------	-------------------------	-------------------	-------------------------	-----------	-----------------	------------

THE LOW ORDER BYTE OF THE PC PIPE

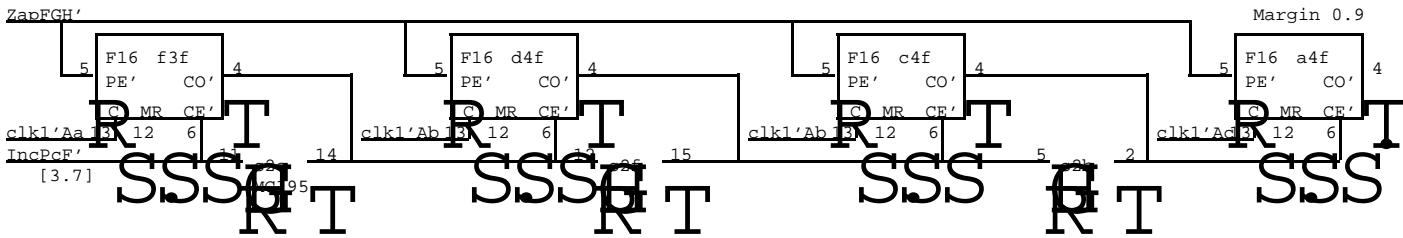
PcJ loads on Jld and not J_H, increments on Jld
 PcF loads on ZapFGH and increments on IncPcF
 ALU turns off on BMuxEnable or BrkPending or IfuReset
 PcFG loads on PcFGLd and increments on IncPcFG



send the memory a bugged PcFG15
 to stay compatible with alto Mesa
 Cut trace to d7:5 when this gets fixed
 so that all inst sets are left/right

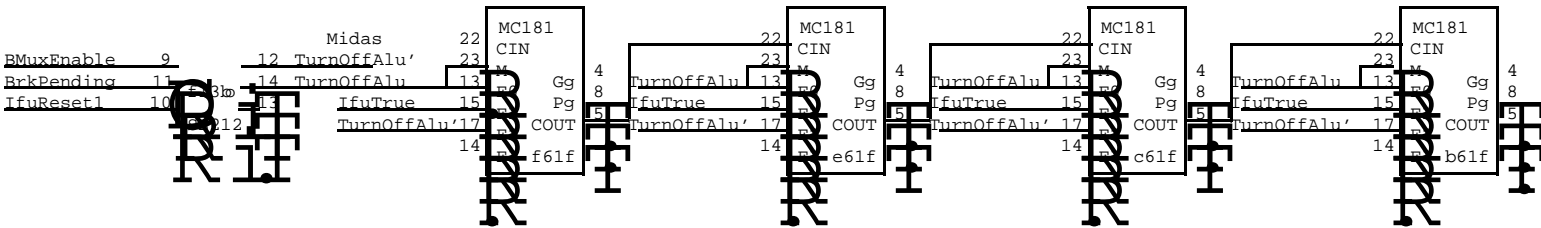
XEROX PARC	Project Dorado	PC Bit Slice(low byte)	File ifu09.sil	Designer S. Ornstein	Rev Cg	Date 8/14/79	Page 09
---------------	-------------------	------------------------	-------------------	-------------------------	-----------	-----------------	------------

PcF Control



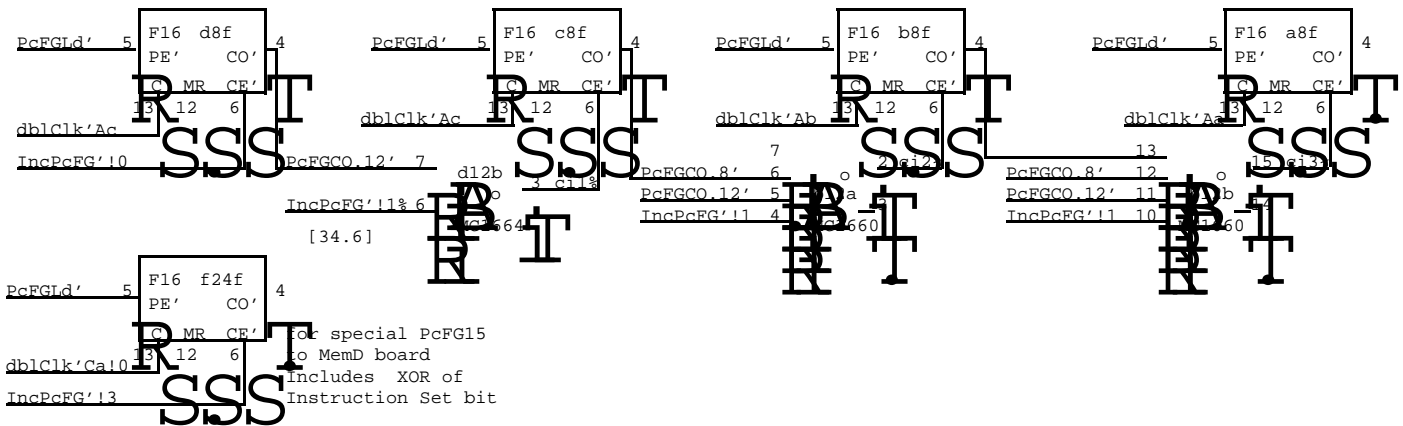
ALU Control

Low data in to high data out = 26.4



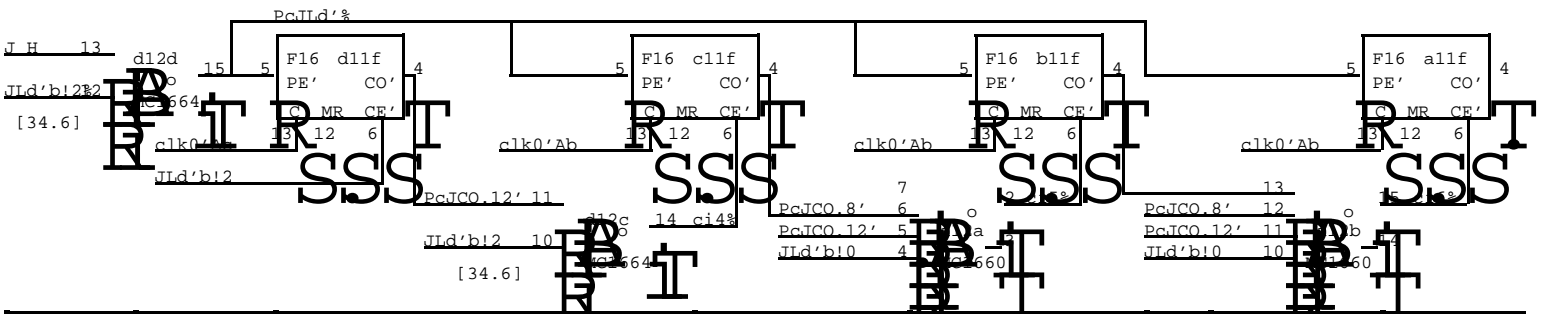
PcFG Control

[Margin 0.7 via Inc]

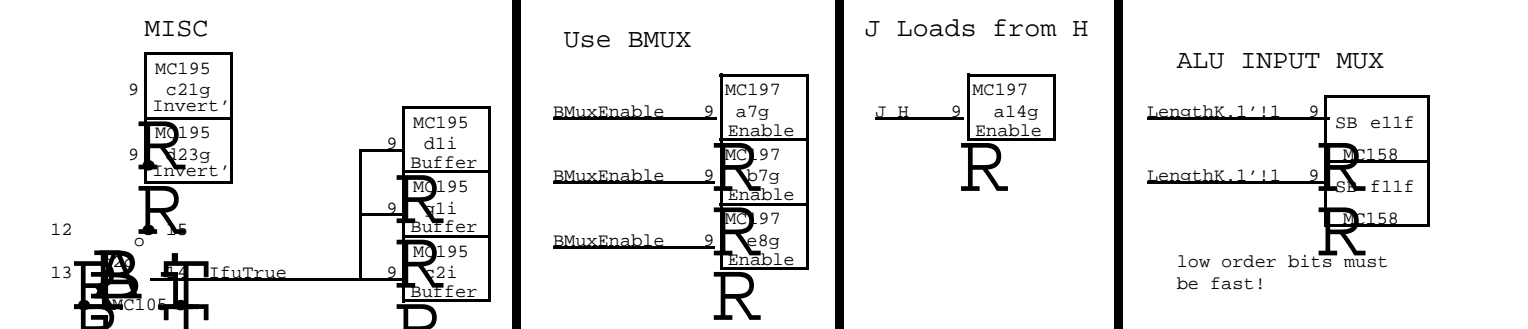
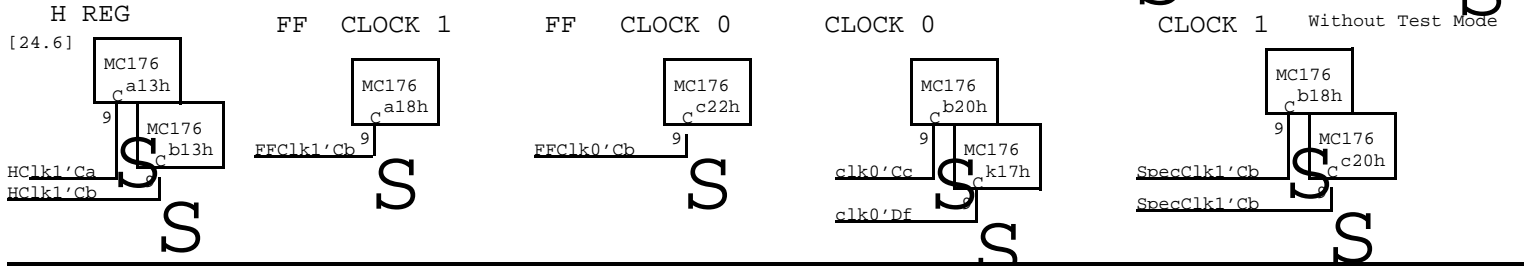
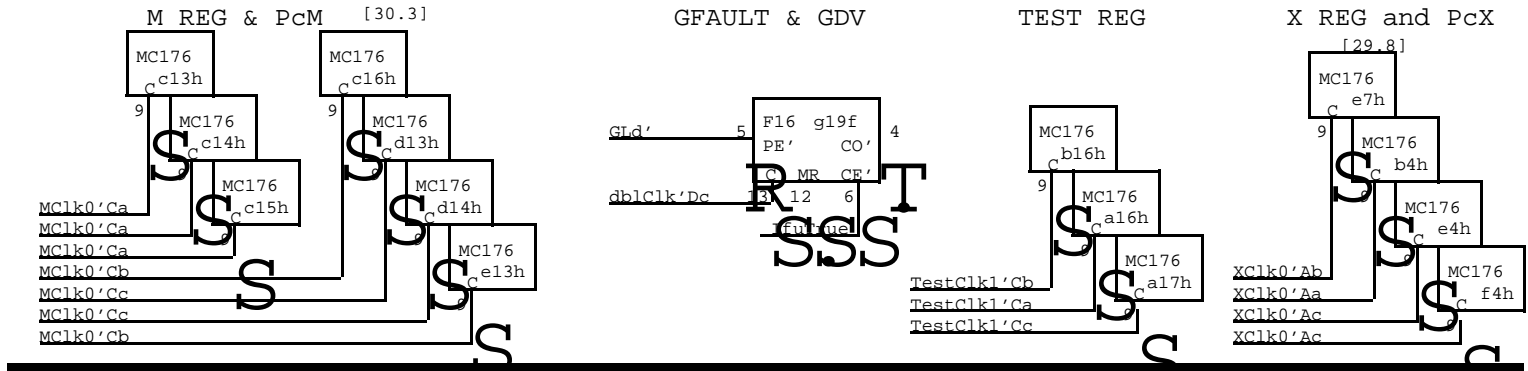
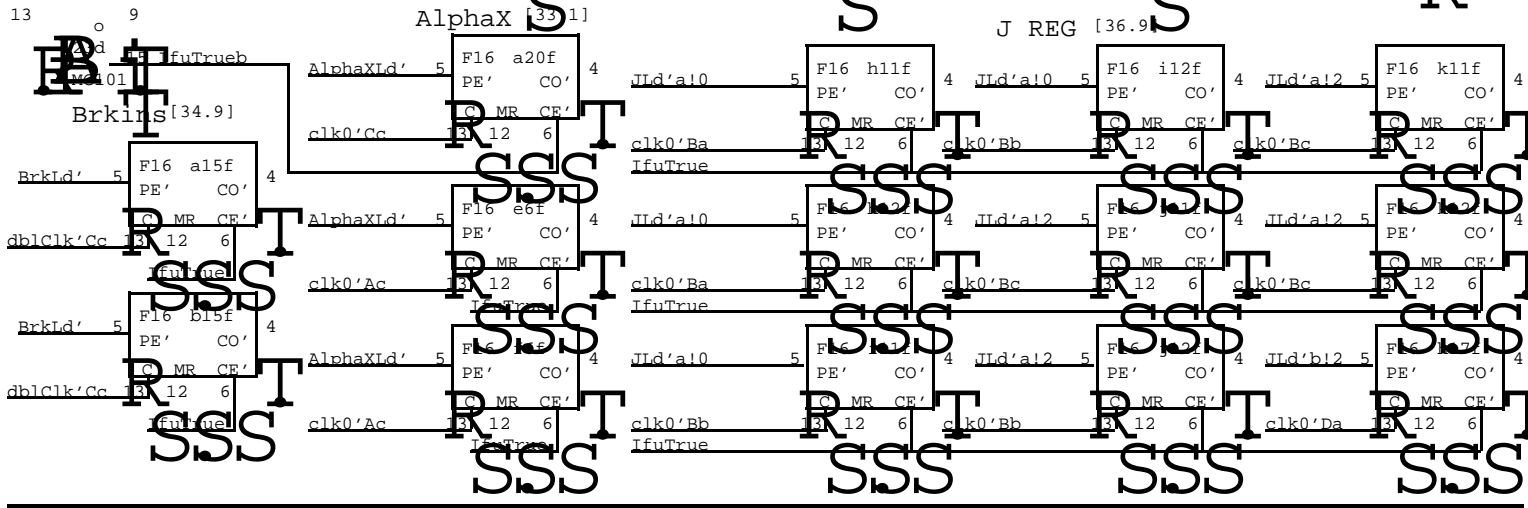
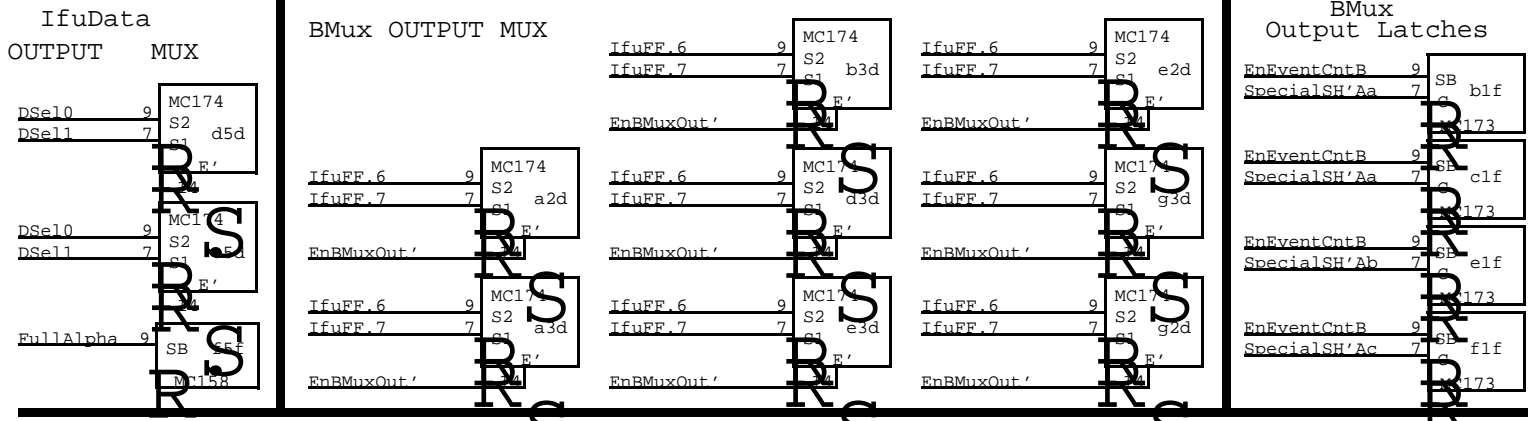


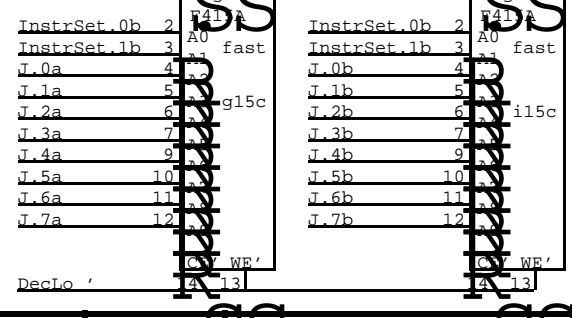
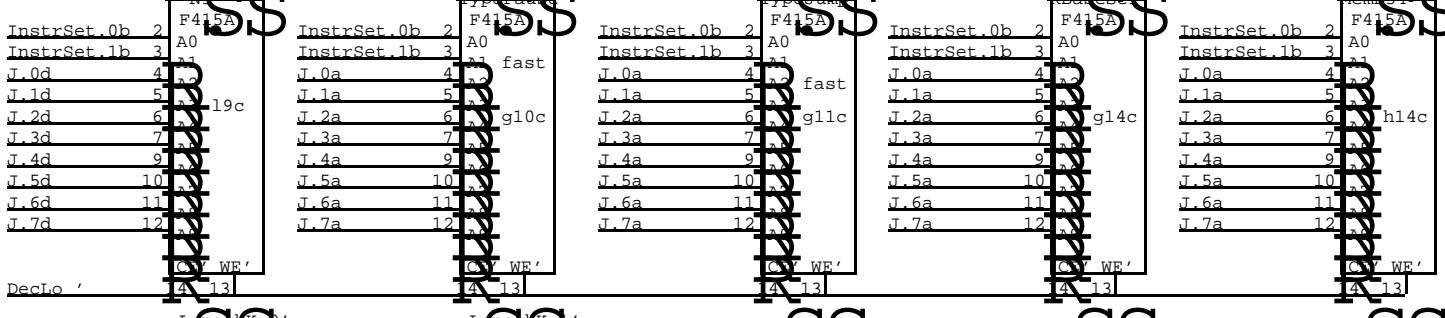
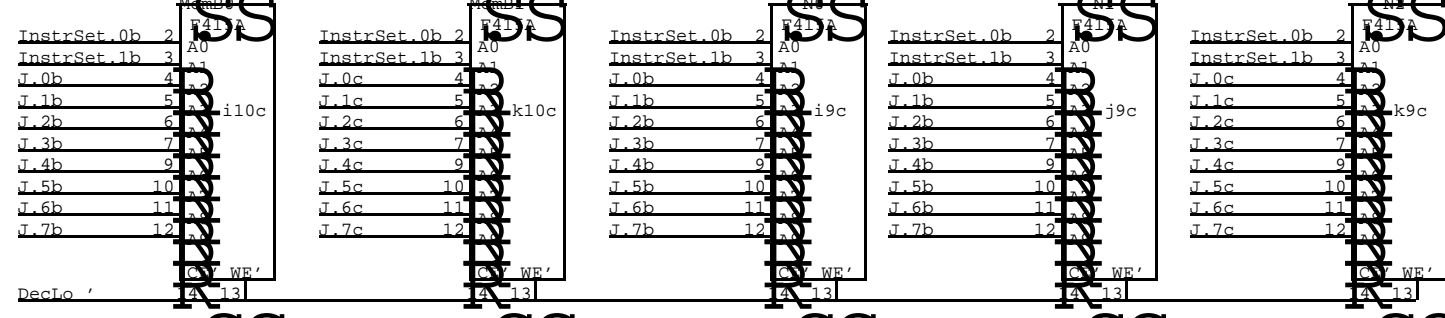
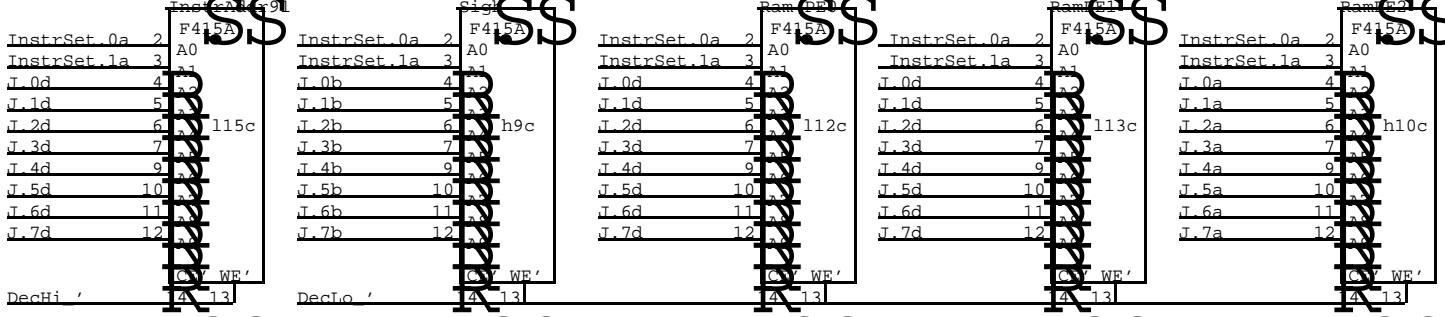
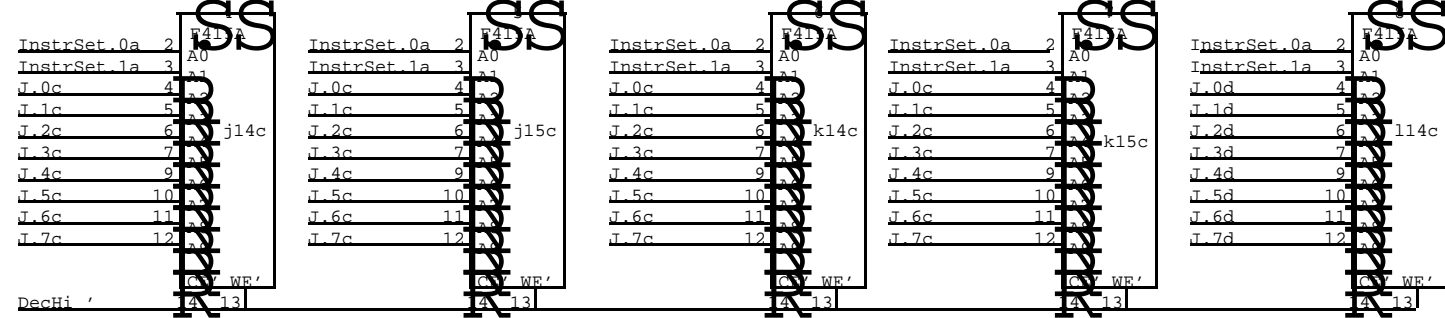
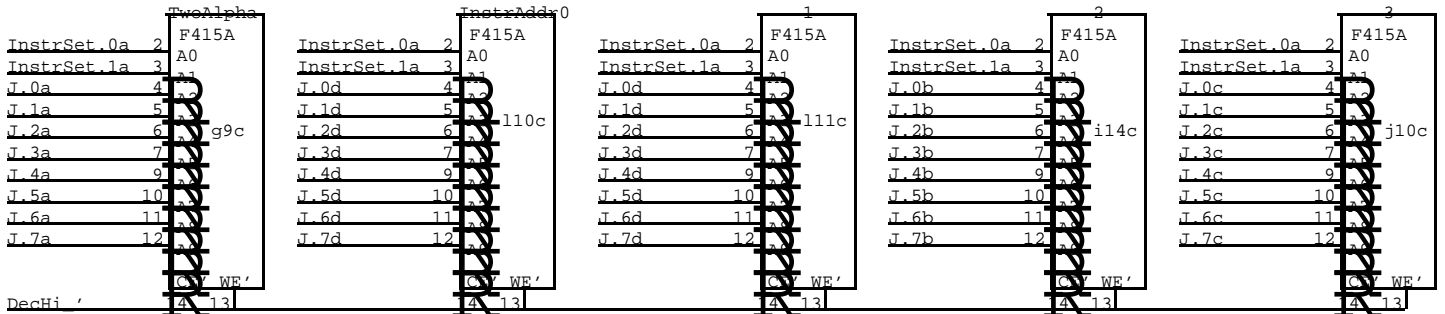
PcJ Control

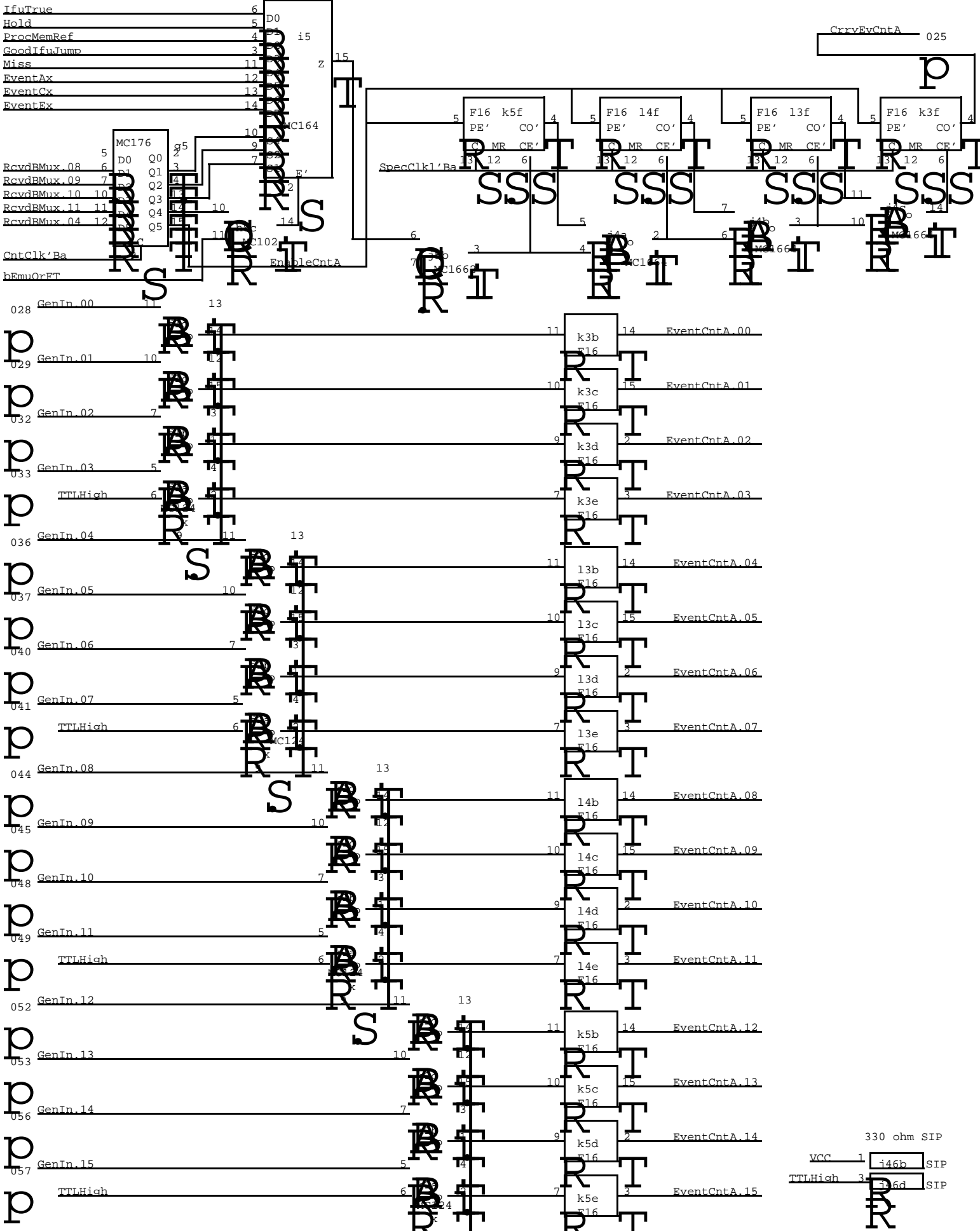
[Margin 0.7 via load and Inc]



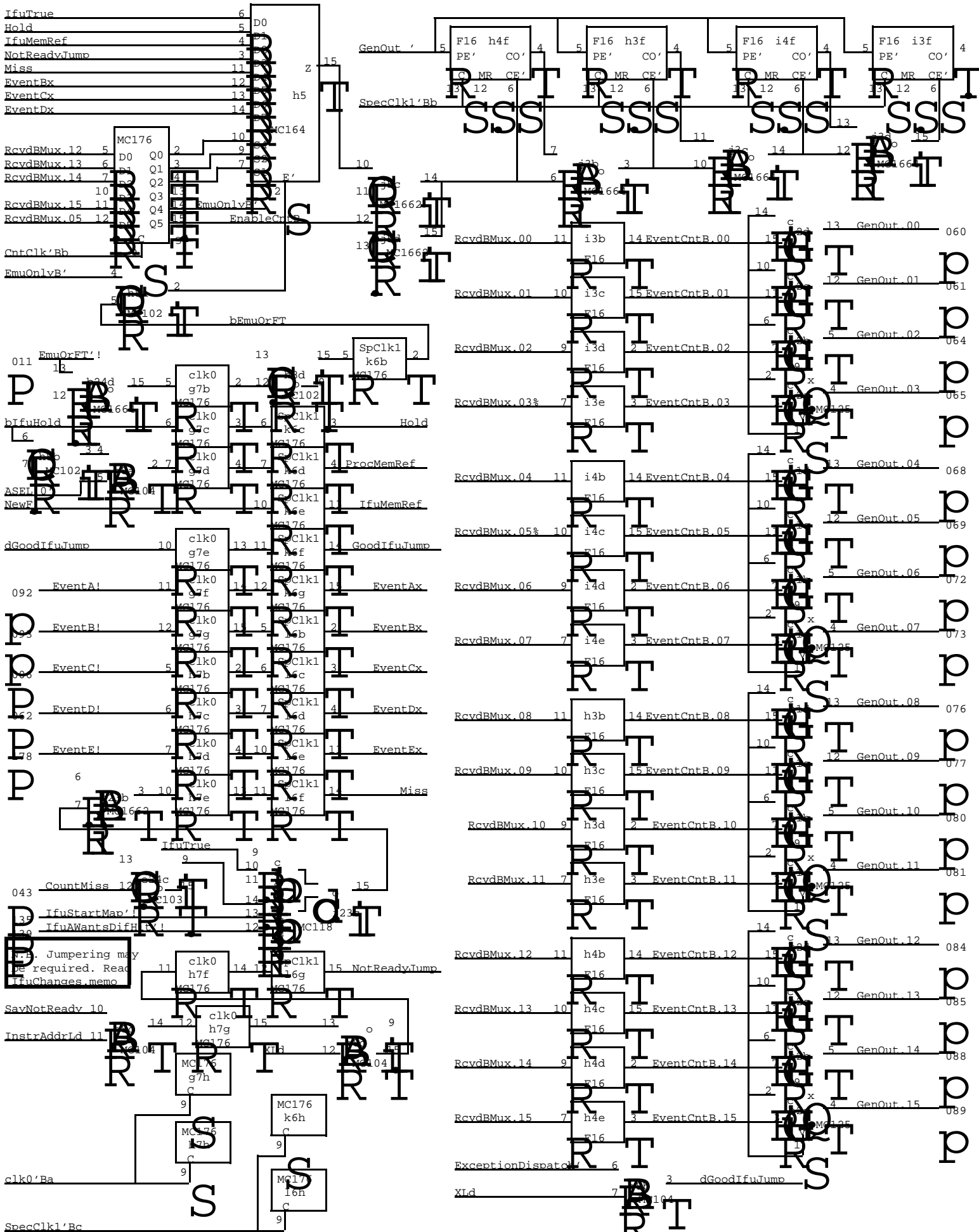
XEROX PARC	Project Dorado	Pc Control	File ifu10.sil	Designer S. Ornstein	Rev Cg	Date 8/23/79	Page 10
---------------	-------------------	------------	-------------------	-------------------------	-----------	-----------------	------------



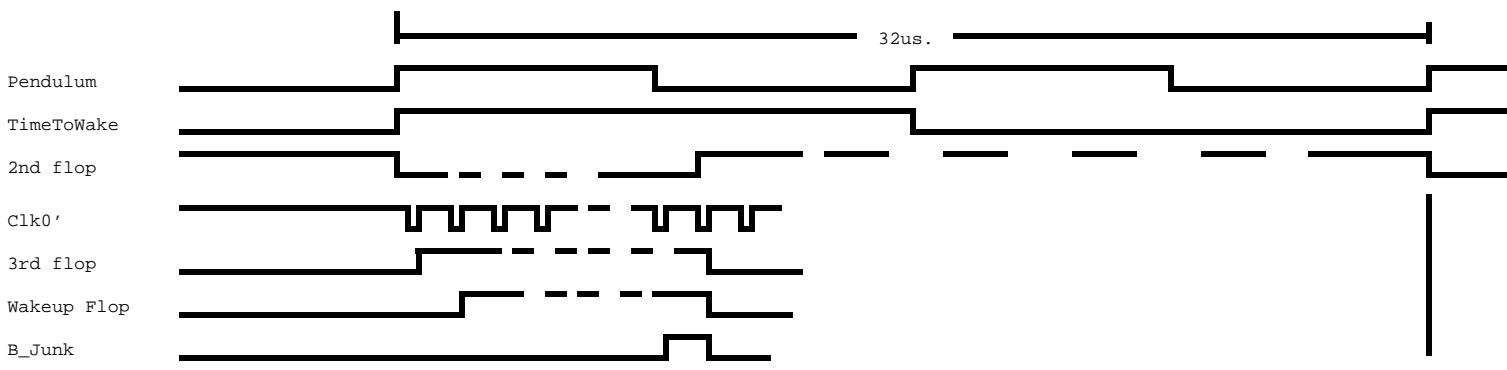
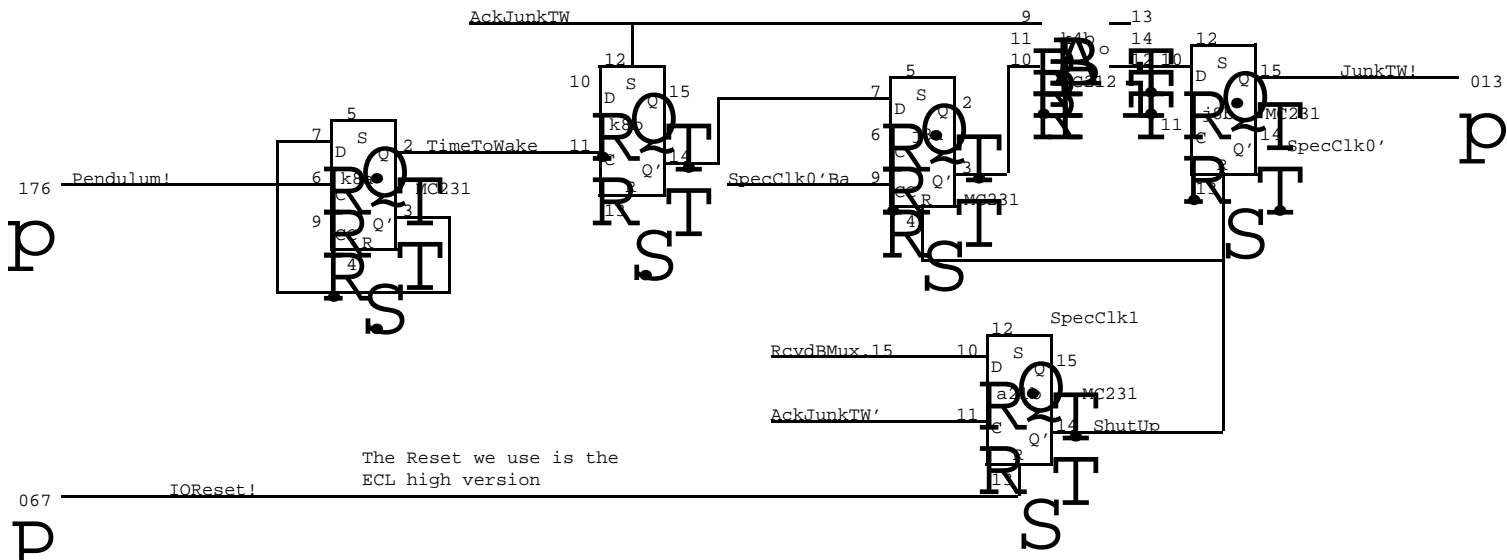




XEROX PARC	Project Dorado	File Event CounterA	Designer S. Ornstein	Rev Cg	Date 1/12/80	Page 13
---------------	-------------------	------------------------	-------------------------	-----------	-----------------	------------

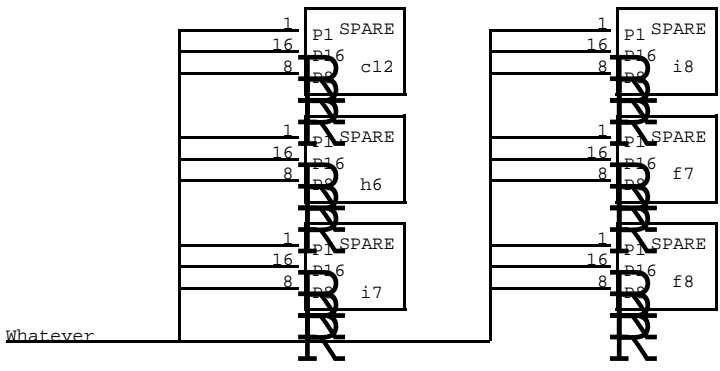
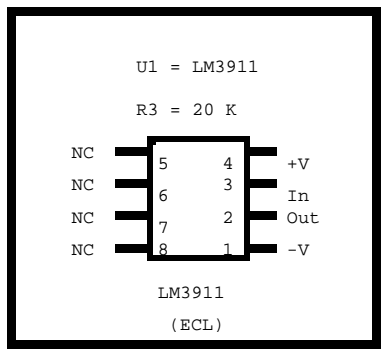
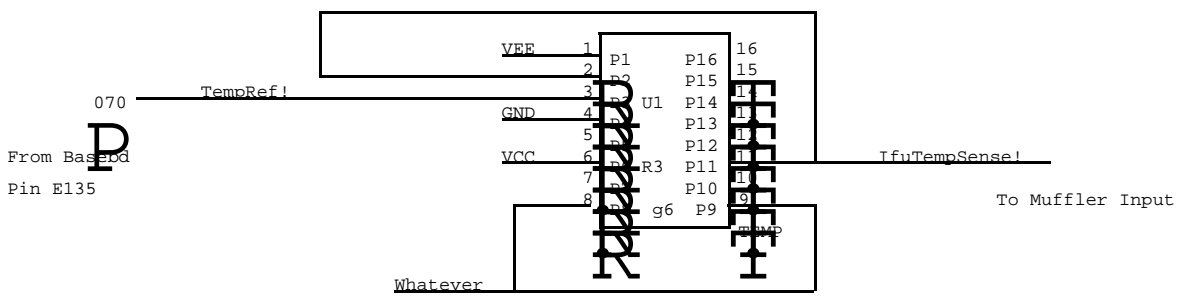


XEROX PARC	Project Dorado	Event Counter B	File ifu14.sil	Designer S. Ornstein	Rev Cg	Date 12/21/79	Page 14
---------------	-------------------	-----------------	-------------------	-------------------------	-----------	------------------	------------

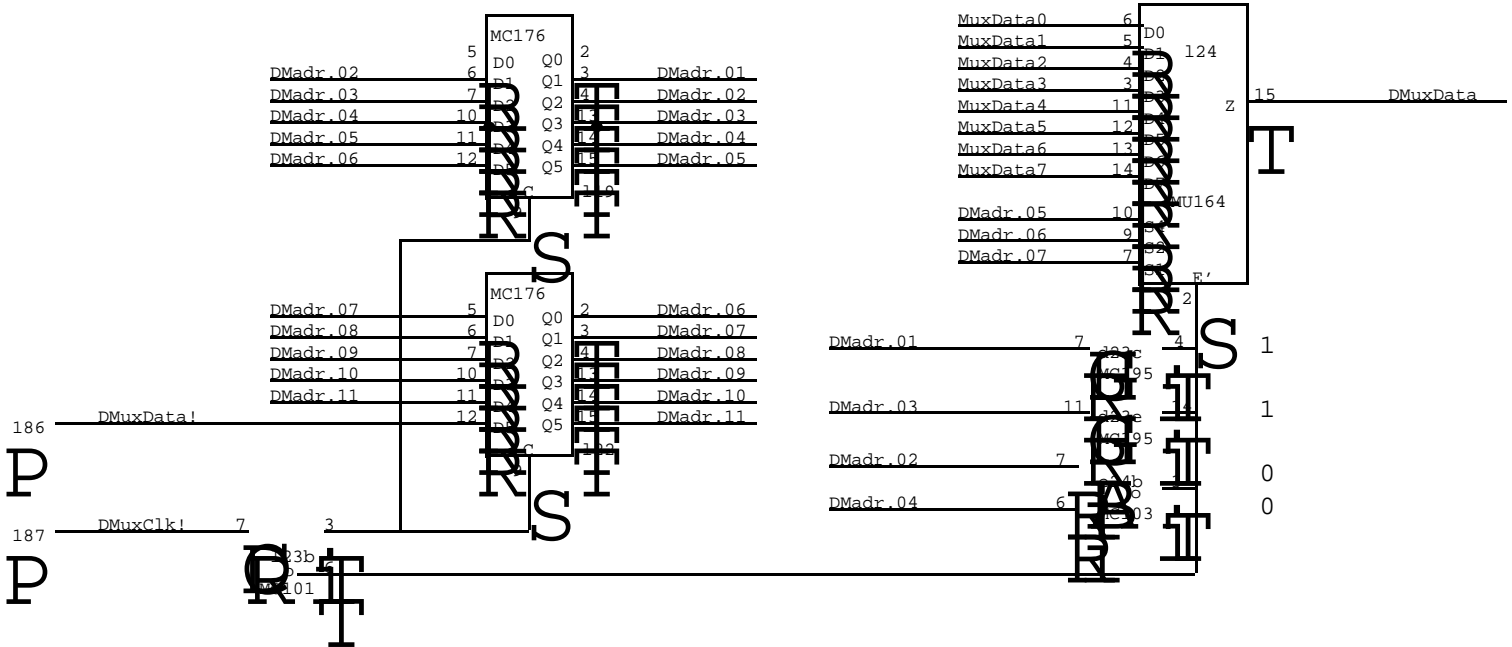


Do it again here

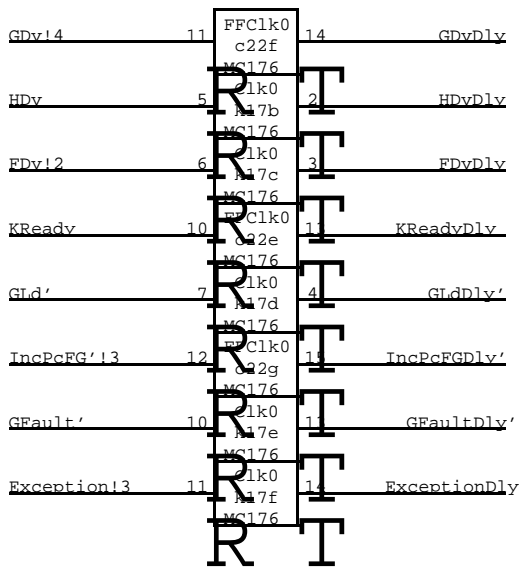
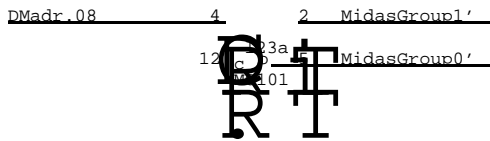
Temperature Sensor

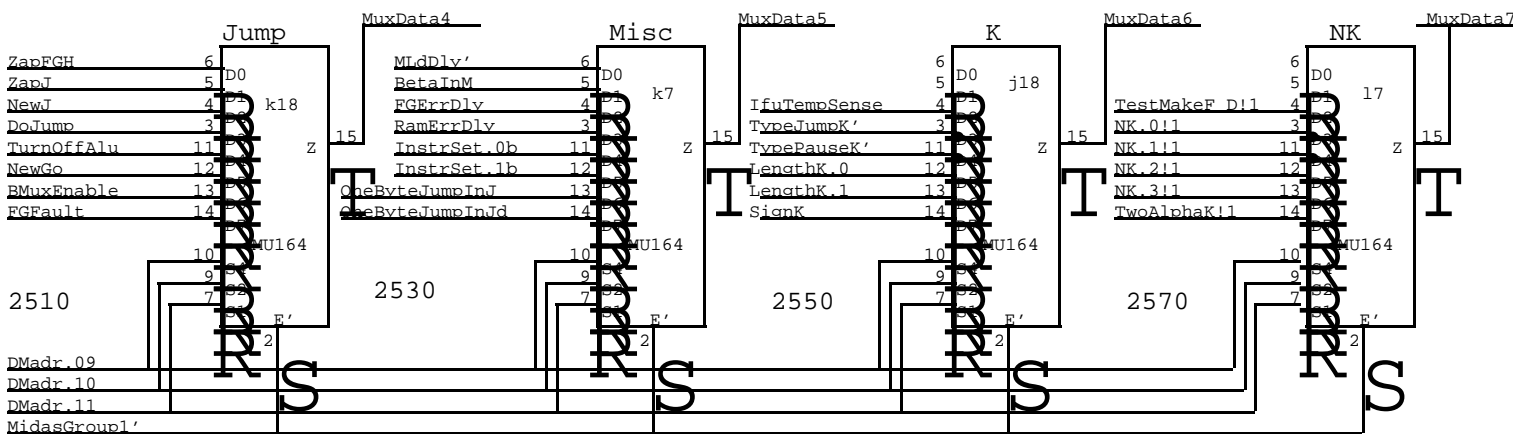
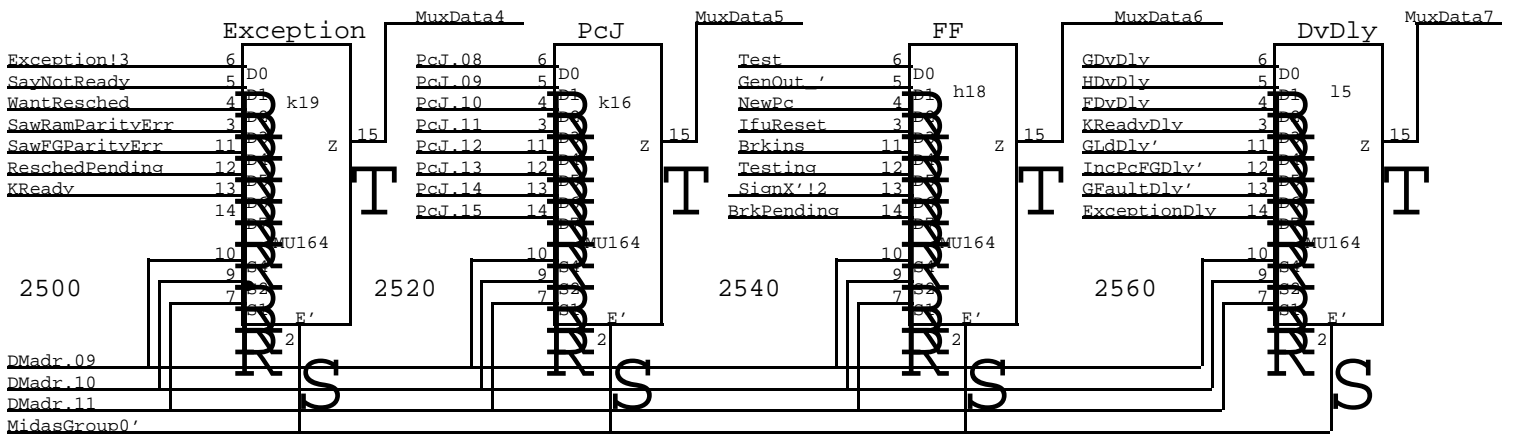
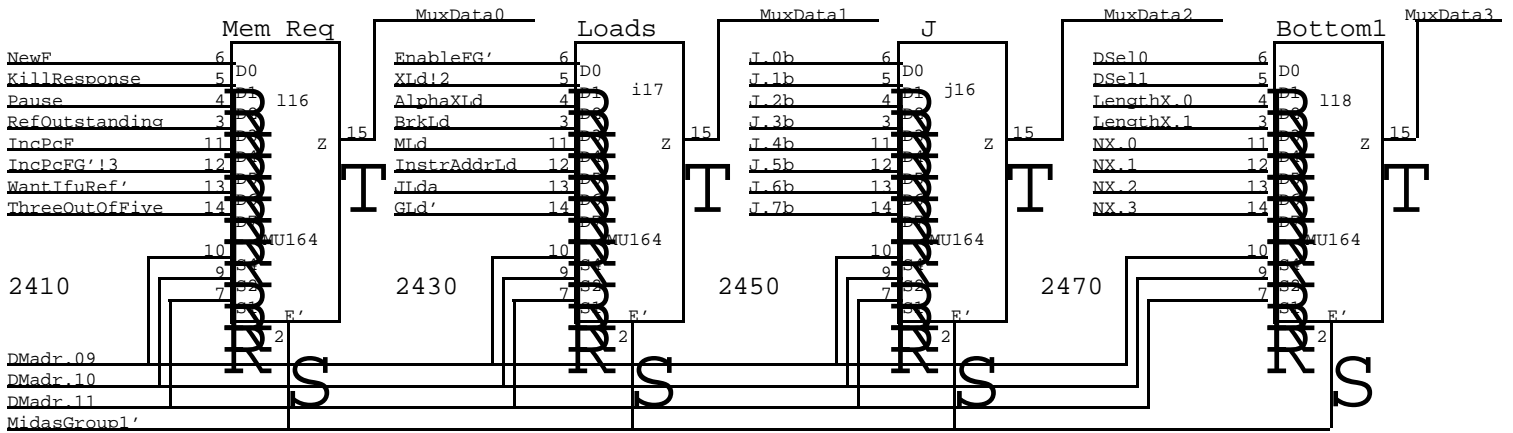
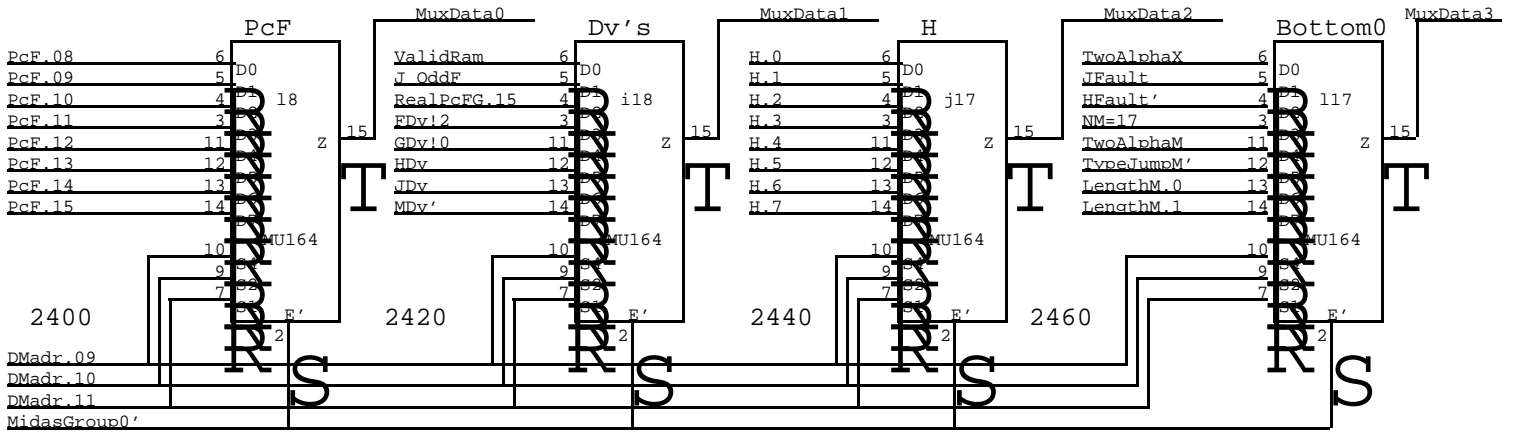


Muffler Control

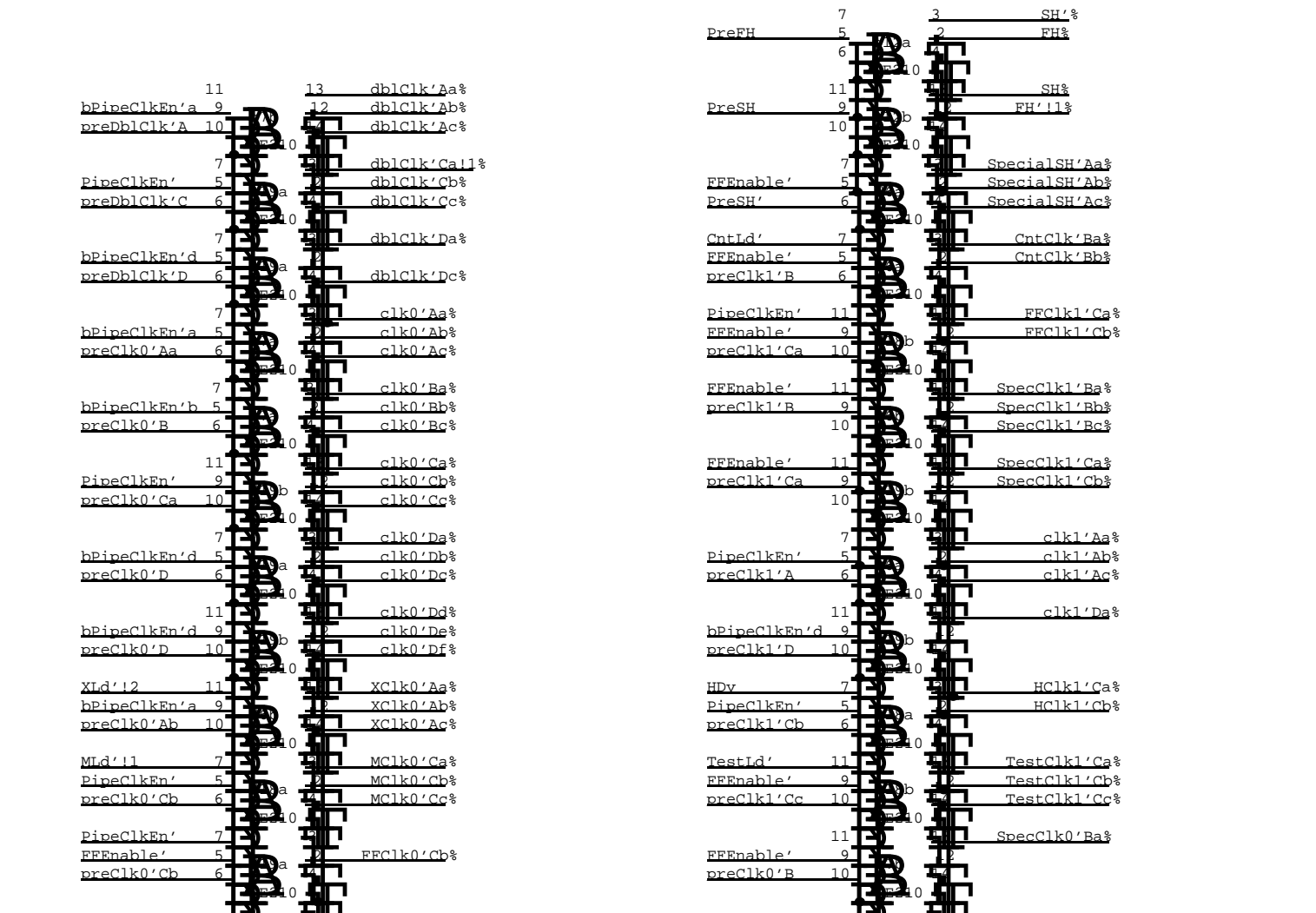
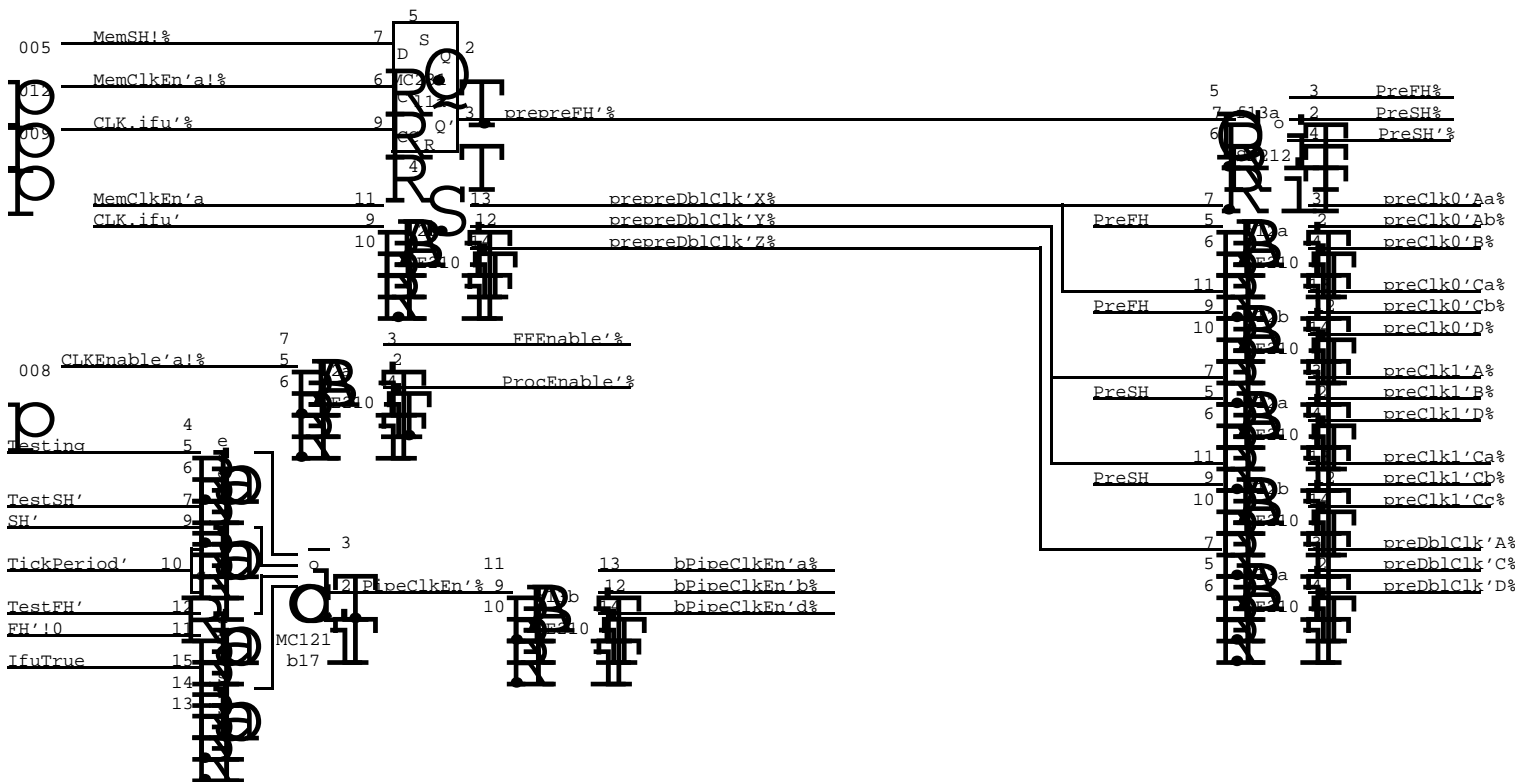


We respond to 2400 thru 2577
Tho we are allocated to 2777





Word0: MemRQ Word4: JmpExc
 Word1: Loads Word5: PcJ
 Word2: HJ Word6: FFK
 Word3: MX Word7: DvNK



01	A MAR 00 b MAR 08 c MAR 01 d MAR 09 101	b BMux 00 SS c BMux 08 d BMux 01 e BMux 09 173	b BMux 02 SS c BMux 10 d BMux 03 e BMux 11 173	b RcvdBmux 02 c RcvdBmux 10 d RcvdBmux 03 e RcvdBmux 11 f RcvdBmux 04 h RcvdBmux 12 195	b BMux 04 SS c BMux 12 d BMux 05 e BMux 13 173	b BMux 06 SS c BMux 14 d BMux 07 e BMux 15 173
02	b dBMux 00 c dBMux 08 174	a RcvdBmux 00 B FullAlpha c IfuTrue 105	b RcvdBmux 08 c RcvdBmux 01 d RcvdBmux 09 e PcFCarry f PcFCarry h PcFCarry 195	A MAR 04 b MAR 12 c MAR 05 d MAR 13 101	b dBMux 05 c dBMux 13 174	A MAR 06 b MAR 14 c MAR 07 d MAR 15 101
03	b dBMux 01 c dBMux 09 174	b dBMux 02 c dBMux 10 174	A MAR 02 b MAR 10 c MAR 03 d MAR 11 101	b dBMux 03 c dBMux 11 174	b dBMux 04 c dBMux 12 174	b PcF 12 C1 c PcF 13 d PcF 14 e PcF 15 F16
04	b c PcF 01 C1 d PcF 02 e PcF 03 F16	b PcX 00 c PcX 08 X0 d PcX 01 e PcX 09 f LengthX=3' g 176	b PcF 04 C1 c PcF 05 d PcF 06 e PcF 07 F16	b PcF 08 C1 c PcF 09 d PcF 10 e PcF 11 F16	b PcX 02 X0 c PcX 10 d PcX 03 e PcX 11 f PcX 04 g PcX 12 176	b PcX 07 X0 c PcX 15 d PcX 06 e PcX 14 f PcX 05 g PcX 13 176
05	b NextData 0 C0 c NextData 1 d NextData 2 F16	DSel0 C0 141	A EnEventCntB b SignIfuData c AlphaX' 106	b IfuData 7 c IfuData 6 174	b IfuData 5 c IfuData 4 174	b IfuData 3 c IfuData 2 d IfuData 1 e IfuData 0 158
06	DSelProm 139	DSel1 C0 141	a clk1' A b 210	a SpecSH A b XClk0 A 210	b AlphaX 07 C0 c AlphaX 06 d AlphaX 05 e AlphaX 04 F16	b AlphaX 03 C0 c AlphaX 02 d AlphaX 01 e AlphaX 00 F16
07	b IfuBMux 00 c IfuBMux 01 d IfuBMux 02 e IfuBMux 03 f IfuBMux 04 h IfuBMux 05 197	b IfuBMux 06 c IfuBMux 07 d IfuBMux 08 e IfuBMux 09 f IfuBMux 10 h IfuBMux 11 197	a clk0' A b dbclk A 210	a PcFG.15 b dSaw FGPE c 1672	b NX.0 X0 c NX.1 d NX.2 e NX.3 f LengthX.0 g LengthX.1 176	
08	b PcFG 00 C2 c PcFG 01 d PcFG 02 e PcFG 03 F16	b PcFG 04 C2 c PcFG 05 d PcFG 06 e PcFG 07 F16	b PcFG 08 C2 c PcFG 09 d PcFG 10 e PcFG 11 F16	b PcFG 12 C2 c PcFG 13 d PcFG 14 e PcFG 15 F16	b IfuBMux 12 c IfuBMux 13 d IfuBMux 14 e IfuBMux 15 f h 197	
09		b ALU 00 c ALU 01 d ALU 02 e ALU 03 181	b ALU 04 c ALU 05 d ALU 06 e ALU 07 181		b ALU 08 c ALU 09 d ALU 10 e ALU 11 181	b ALU 12 c ALU 13 d ALU 14 e ALU 15 181
10						
11	b PcJ 00 C0 c PcJ 01 d PcJ 02 e PcJ 03 F16	b PcJ 04 C0 c PcJ 05 d PcJ 06 e PcJ 07 F16	b PcJ 08 C0 c PcJ 09 d PcJ 10 e PcJ 11 F16	b PcJ 12 C0 c PcJ 13 d PcJ 14 e PcJ 15 F16	b dALU 08 c dALU 09 d dALU 10 e dALU 11 158	b dALU 12 c dALU 13 d dALU 14 e dALU 15 158
12	a PcJCarry b PcJCarry 1660	a PcFGCarry b PcFGCarry 1660		a dIncPcFG b PcFGCarry c PcJCarry d PcJLd 1664	a FH b SH 210	a preClk1 b preClk1 210

A

B

C

D

E

F

01	b RcvdBmux 05 c RcvdBmux 13 d RcvdBmux 06 e RcvdBmux 14 f RcvdBmux 07 h RcvdBmux 15 195	A GenOut.11 b GenOut.10 c GenOut.09 d GenOut.08 125	A GenOut.07 b GenOut.06 c GenOut.05 d GenOut.04 125	A Genin.03 b Genin.02 c Genin.01 d Genin.00 124	A Genin.11 b Genin.10 c Genin.09 d Genin.08 124	A preprePH'a b See Pg. 2 231
02	b dBmux 06 c dBmux 14 174	A GenOut.15 b GenOut.14 c GenOut.13 d GenOut.12 125	A GenOut.03 b GenOut.02 c GenOut.01 d GenOut.00 125	A Genin.07 b Genin.06 c Genin.05 d Genin.04 124	A Genin.150 b Genin.14 c Genin.13 d Genin.12 124	a ClkEnable b prepreDblClk 210
03	b dBmux 07 c dBmux 15 174	Spec 1 EventCntB F16	Spec 1 EventCntB F16	a BCntCarry b BCntCarry c BCntCarry d BCntCarry 1664	Spec 1 EventCntA F16	Spec 1 EventCntA F16
04	CntClk BCntMuxCntrl 176	Spec 1 EventCntB F16	Spec 1 EventCntB F16	a ACntCarry b ACntCarry c ACntCarry d 1664	a b dTWReq01 212	Spec 1 EventCntA F16
05	CntClk ACntMuxCntrl 176	BCntMux 164	ACntMux 164	a dCntA b dCntB c dCntB d dCntB 1662	Spec 1 EventCntA F16	DvDly MU164
06	Temp Sensor		a CntClk B b SpecClk1 B 210	A IfuReset b NewGo SpecClk0 231	b bEmuOrFT Spec 1 c Hold Spec 1 d ProcMemRef e IfuMemRef f GoodIfuJmp 176 g EventAx 176	b EventBx c EventCx Spec 1 d EventDx e EventEx f Miss g NotRdyJmp 176
07	c0 Events 176	c0 Events 176		a clk0 B b SpecClk0 B 210	Misc MU164	NK MU164
08	a EventCntB b EventCntB c EventCntB D EventCntB 104	a BCntMuxEnb b bIfuHold' c ACntMuxEnb D bEmuOrFT 102		SpecClk0 A SyncedTTW b TWReq01 231	A TimeToWake b del TTW Pendulum 231	PCF MU164
09	TwoAlphaK 415	SignK 415	NK.0 415	NK.1 415	NK.2 415	NK.3 415
10	TypePauseK 415	Parity2 415	MEMBK.0 415	IfuAddr 3 415	MemBK.1 415	IfuAddr 0 415
11	TypeJumpK 415	b J0a c J1a c0 d J2a e J3a F16	b J0b c J1b c0 d J2b e J3b F16	b J0c c J1c c0 d J2c e J3c F16	b J0d c J1d c0 d J2d e J3d F16	IfuAddr 1 415
12	a preClk0 b preClk0 210	b J4a c J5a c0 d J6a e J7a F16	b J4b c J5b c0 d J6b e J7b F16	b J4c c J5c c0 d J6c e J7c F16	b J4d c J5d c0 d J6d e J7d F16	Parity0 415

G

H

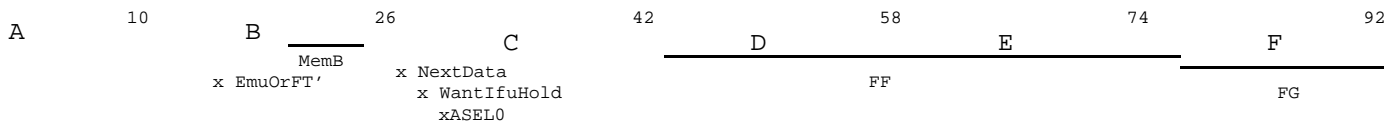
I

J

K

L

13	b H 0 c H 1 d H 2 e H 3 f H 4 g H 5 176	HCl	b H.6 c H.7 d HFault e f g 176	HCl	b PcM 00 c PcM 01 d PcM 02 e PcM 03 f PcM 04 g PcM 05 176	MC0	b AlphaM 0 c AlphaM 1 d AlphaM 2 e AlphaM 3 f AlphaM 4 g AlphaM 5 176	MC0	b AlphaM 6 c AlphaM 7 d PcM 12 e PcM 13 f PcM 14 g PcM 15 176	MC0	a preFH b TurnOffAlu 212
14	b EnableH 0 c EnableH 1 d EnableH 2 e EnableH 3 f EnableH 4 h EnableH 5 197		a EnableH 6 b EnableH 7 c dSignX D dTwoAlphaX 104		b PcM 06 c PcM 07 d PcM 08 e PcM 09 f PcM 10 g PcM 11 176	MC0	b LengthM.0 c LengthM.1 d SpareM.0 e SpareM.1 f dNM=17a g dNM=17b 176	MC0	FGParityErr 170		b InstrSet 0a c InstrSet 0b d InstrSet 1a e InstrSet 1b FFc1 F16
15	b Brkins 0 c Brkins 1 d Brkins 2 e Brkins 3 F16	c2	b Brkins 4 c Brkins 5 d Brkins 6 e Brkins 7 F16	c2	b SignM c TwoAlphaM d NM 0 e NM 1 f NM 2 g NM 3 176	MC0	a NM=17 b NK2&NK3 c NK0&NK1 D IfuMemAck 104		a dHJ4 b dHJ5 c dHJ6 d dHJ7 1664		a InstrSetLd b CntLd C d TestOrReset 103
16	b TestReg 0 c TestReg 1 d TestReg 2 e TestReg 3 f TestReg 4 g TestReg 5 176	Test1	b TestReg 6 c TestReg 7 d e TestFH f TestSH g Testing 176	Test1	b MemBM 0 c MemBM 1 d LengthM=3 e MDv f TypeJumpM g LengthM=3 176	MC0	A BrkPending b 231	c2	a dHJ0 b dHJ1 c dHJ2 d dHJ3 1664		IncPcFG 119
17	b TestIfuAck c TestMemAck d TestMakeF_D e TestFault f g 176	Test1	PipeClkEn 121		A SawFGPe b (delayed clock) 231	c2	b AlphaXLd c XShift 117		A Pause b IncPcF 231	c0	a dPause b PcFGLd c bIfuHold d dAckJunkTW 1664
18	b NewPcl c AckJnkTW d DecLo_ e DecHi_ f g Brkins 176	FF1	b c GenOut_ d InstrSetOrJunk_ e IfuReset1 f g 176	Spec1	a HClk1' b TestClk1' 210	C	a MClk0' b FFClk1' 210	C	a XLd b KReady 1660		InstrAddrLd 121
19	a BMuxEnable B NextBeta 109		a dNewGo B EnBMuxOut c TestLd 105		a dblclk' b SpecClk1' 210	C	a FFClk0' b Clk0' 210	C	a PcFG15 b KReady c Kready d dBrkLd 1662		a JKillResponse b FGDv 212
20	b SignX c TwoAlphaX d BetaInM e F16	AXC0	b bIfuFault c NewF d NewF' e MLdDly f DoJump g DoJump' 176	c0	b Test_ c dTickPeriod d RamErrDly e FGErrDly f WantIfuHoldDly g 176	Spec1	b ExtendH c sIncPcF 117		a B dSawFGPe c InstrAddrLd 105		A HDv b FDv c2 135
21	A TickPeriod b ShutUp (junk) Spec1 231		b c WantIfuHold 117		b InstrSetOrJunk_ c ReschedPending d SayRamParityErr e DecLo_ f DecHi_ h TestMemAck' 195	inv	b dJDv c ZapJ 117		b BrkLd c SayNotReady 117		A NewJ b RefOutstanding c0 135
22	A FF130 b FF131 c FF132 d FF133 100		a dReschedPending b dReschedPending 1660		b ReschedPending c NewPc_ d WantResched e KReadyDly f GDvDly g IncPcFGDly 176	ff0	a SayResched b SayFGParityErr C BetaInM' d IfuNextData 103		a BetaInH B J FDv c NewGoOrNoM 105		a dSayNotReady b K NewJ c dGDv d K FDv 1664
23	A FF134 b FF135 c FF136 d FF137 100		A FF100 b FF101 c FF102 d dReschedPending 100		FC 161		b GenOut_ c DMuxEnable d AckJnkTW' e DMuxEnable f ReschedPending' h SayNotReady 195	inv	a ZapFGH B JHDv c SeeJump 105		a EnableFG' b J_H 1660
24	FB 161		a FF5 b FF6 c FF7 d EmuOrFT' 1664		a FA=1 b SayStuff 210		a J_OddF b SayNotReady1 1660		a EnableFG' b DMuxEnable C dMiss d lByteJumpInJ 103		b c d e specialPcFG15 F16 c2



XEROX PARC	Project Dorado	reference Lower Left Quadrant	File ifu21.sil	Designer S. Ornstein	Rev Cg	Date 12/21/79	Page 21
---------------	-------------------	----------------------------------	-------------------	-------------------------	-----------	------------------	------------

N	Two Alpha	JMP	Length	Address	Contents	Numeric Sequence	Effective Sequence		
N	Two Alpha	q	0	00	11111111	3, 3, 3, 3,	3IL, IL, IL, IL, IL.....		
			1	01	11111111				
			2	02	11111111				
		3	03	11111111					
		r	0	04	11111111				
			1	05	11110111			2, 3, 3, 3,	3N, IL, IL, IL, IL.....
			2	06	10110011			2, 0, 3, 3,	3N, Alpha, IL, IL, IL.....
	3		07	10010001	2, 0, 0, 3,	3N, Alpha, Beta, IL, IL.....			
	Two Alpha	q	0	10	11111111	3, 3, 3, 3,	3IL, IL, IL, IL, IL.....		
			1	11	11111111				
			2	12	11111111				
		3	13	11111111					
		r	0	14	11111111				
			1	15	11111111				
			2	16	10010101			2, 1, 0, 3,	NB, Alpha[0:3], Alpha[4:7], IL, IL, IL, IL.....
	3		17	10000100	2, 1, 0, 0,	NB, Alpha[0:3], Alpha[4:7], Beta, IL, IL, IL, IL.....			
	no N	r	q	0	20	11111111	3, 3, 3, 3,	3IL, IL, IL, IL, IL.....	
1				21	11111111				
2				22	11111111				
3			23	11111111					
r			0	24	11111111				
			1	25	11111111				
			2	26	01110111	0, 3, 3, 3,			3Alpha, IL, IL, IL, IL.....
		3	27	00110011	0, 0, 3, 3,	3Alpha, Beta, IL, IL, IL.....			
Two Alpha		q	0	30	11111111	3, 3, 3, 3,	3IL, IL, IL, IL, IL.....		
			1	31	11111111				
			2	32	11111111				
		3	33	11111111					
		r	0	34	11111111				
			1	35	11111111				
			2	36	00111011			1, 0, 3, 3,	Alpha[0:3], Alpha[4:7], IL, IL, IL, IL.....
3			37	00011001	1, 0, 0, 3,	Alpha[0:3], Alpha[4:7], Beta, IL, IL, IL, IL.....			



No zero length instructions

No 3 - Byte Jumps

Jumps never export anything other than the length (no "N" on 2 - byte Jumps).

A one - byte instruction specifying Two Alpha is meaningless