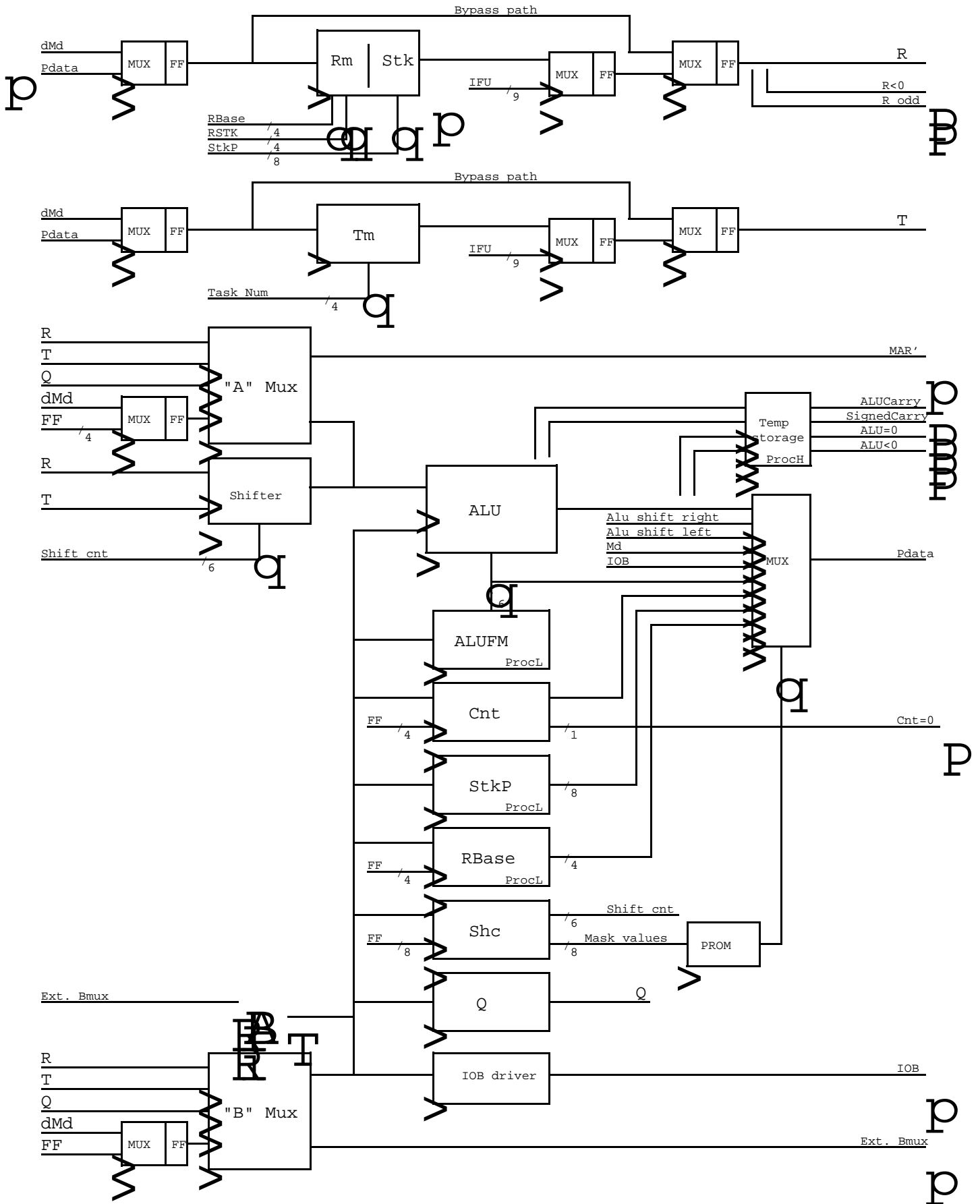


D O R A D O S C H E M A T I C S
 L o w B y t e
 P R O C E S S O R

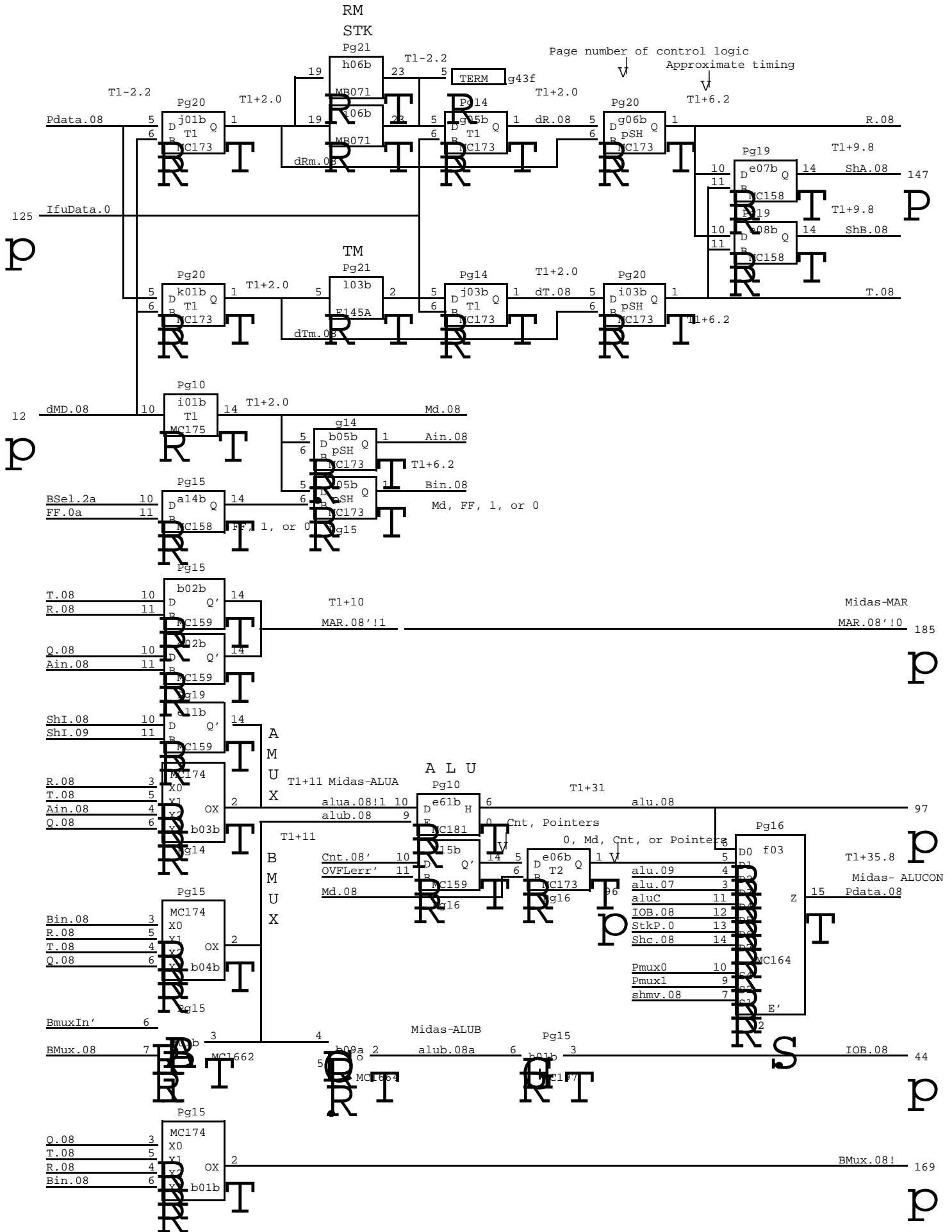
Table of contents

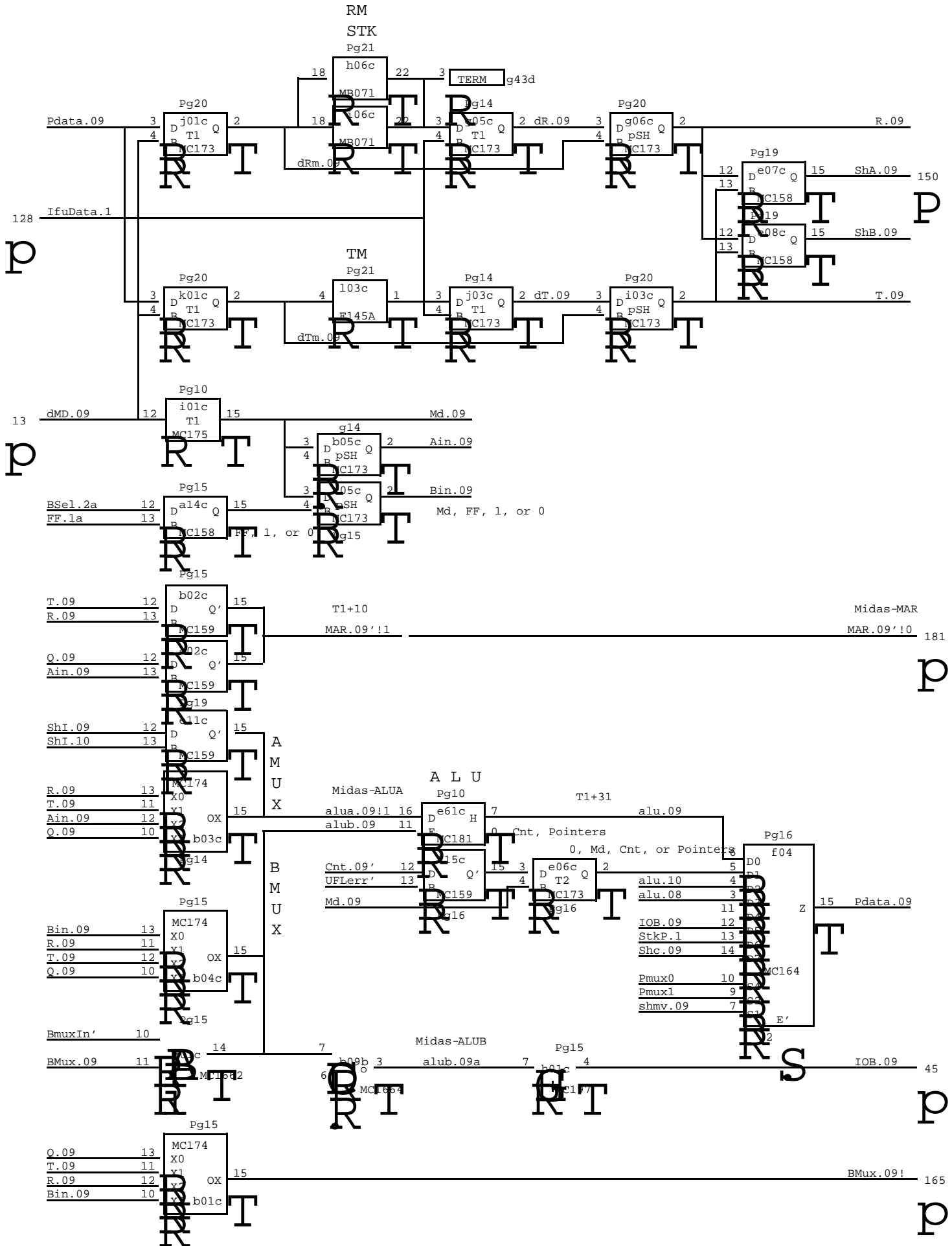
<u>TITLE</u>	<u>Page</u>
Block Diagram	01
Bit slice drawings	02
ALU carry and RM fast-branch	10
ALU function Ram	11
Task & Sub-Task logic	12
Function Field decoding	13
"A" multiplexor & Mar multiplexor control	14
"B" multiplexor control	15
"Processor" multiplexor control	16
Q and Count Registers	17
Shift Control register	18
Shifter multiplexors	19
Rm & T write control	20
Rm, Stk, and T address lines	21
Rm Address control & Bypass	22
Stk Address control & Stk Ufl/Ovfl tests	24
Hold Simulator and Temp Sence	26
Parity logic	27
Midas control and multiplexors	28
Clock distribution	30
Termination resistors for Multiwire	31
Layout	32
Loading Information	33



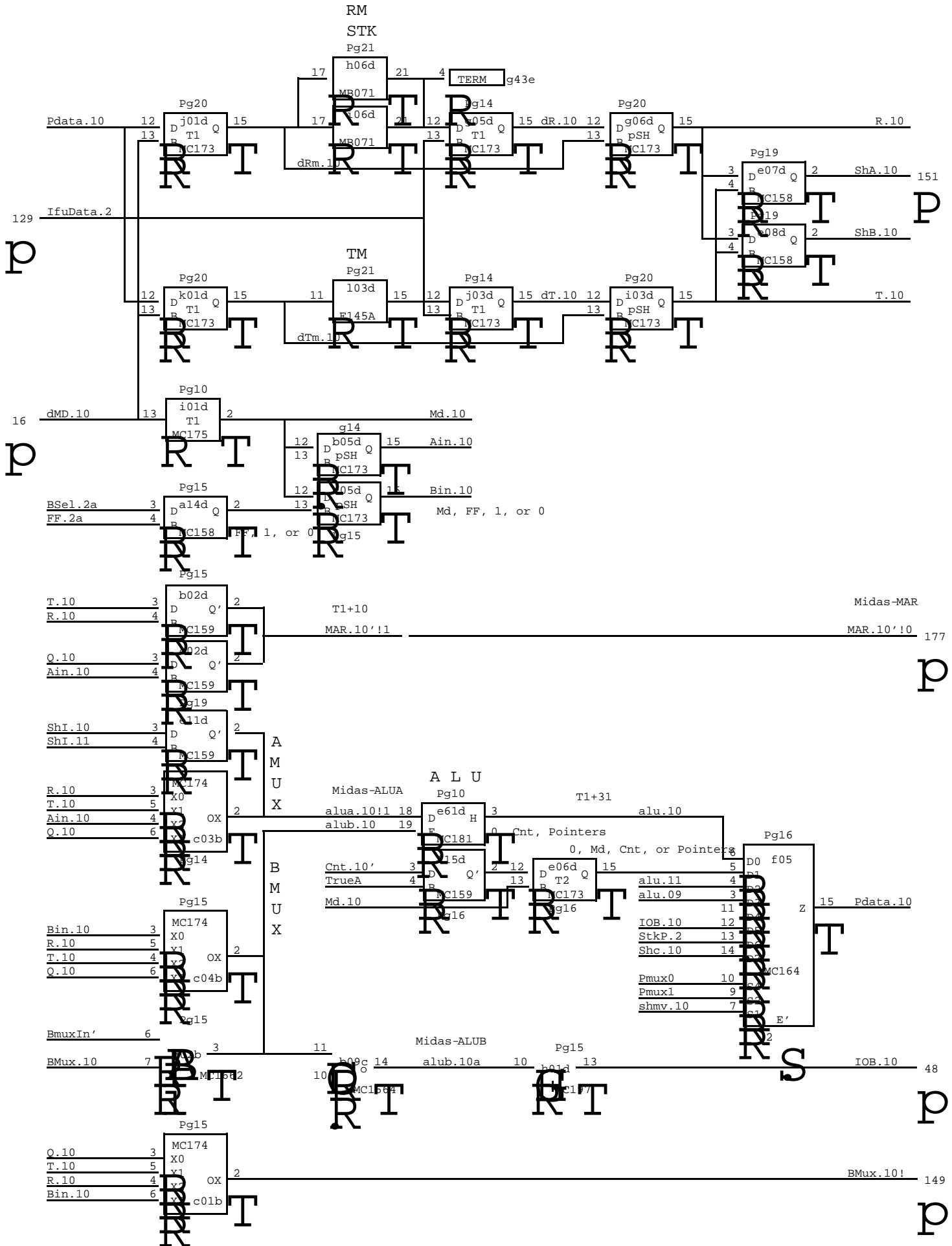
Note: all busses 16 wide unless noted otherwise

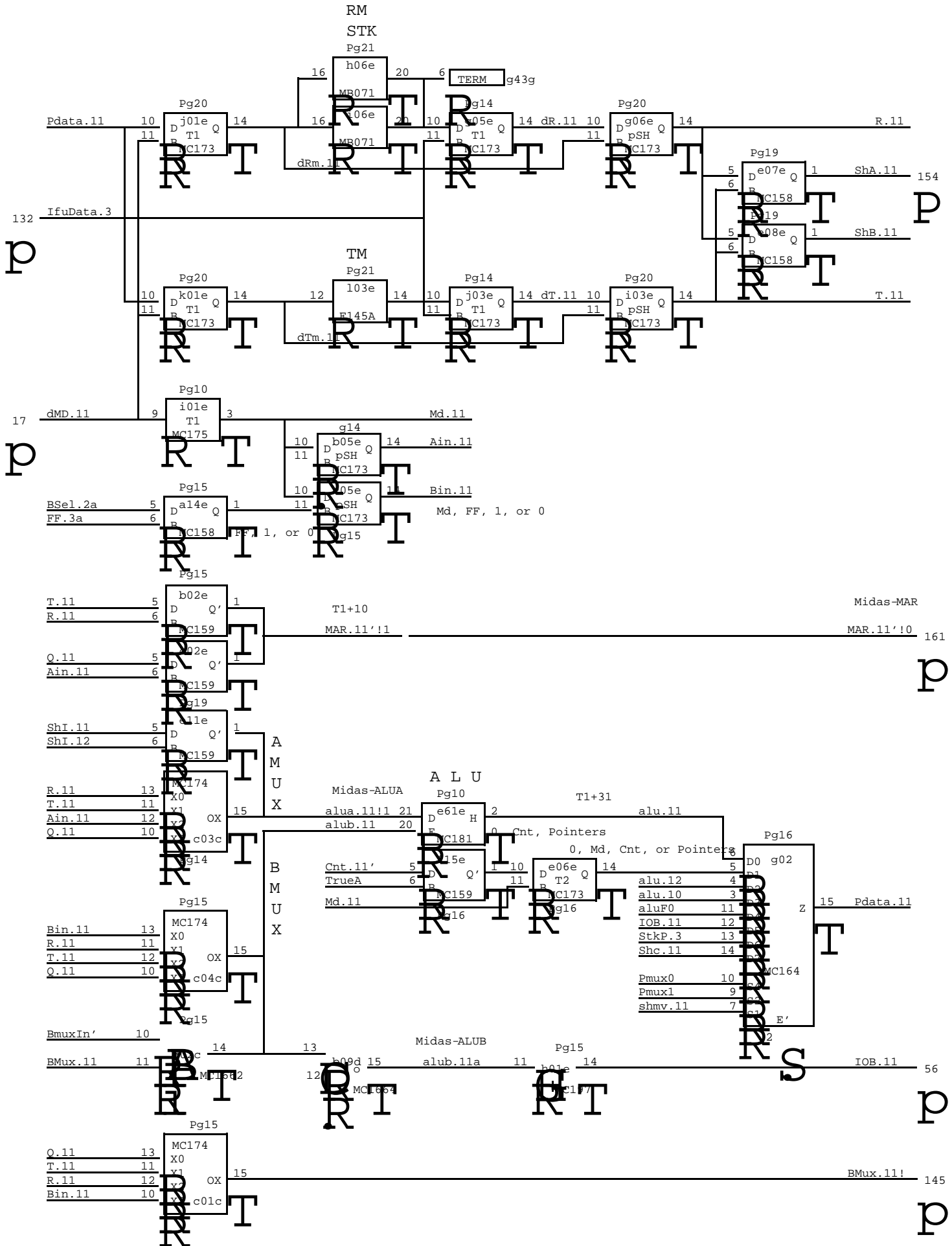
XEROX	Project	Reference	File	Designer	Rev	Date	Page
PARC	Dorado	Block Diagram	ProcL01.sil	R Bates	Ci	6/27/79	01

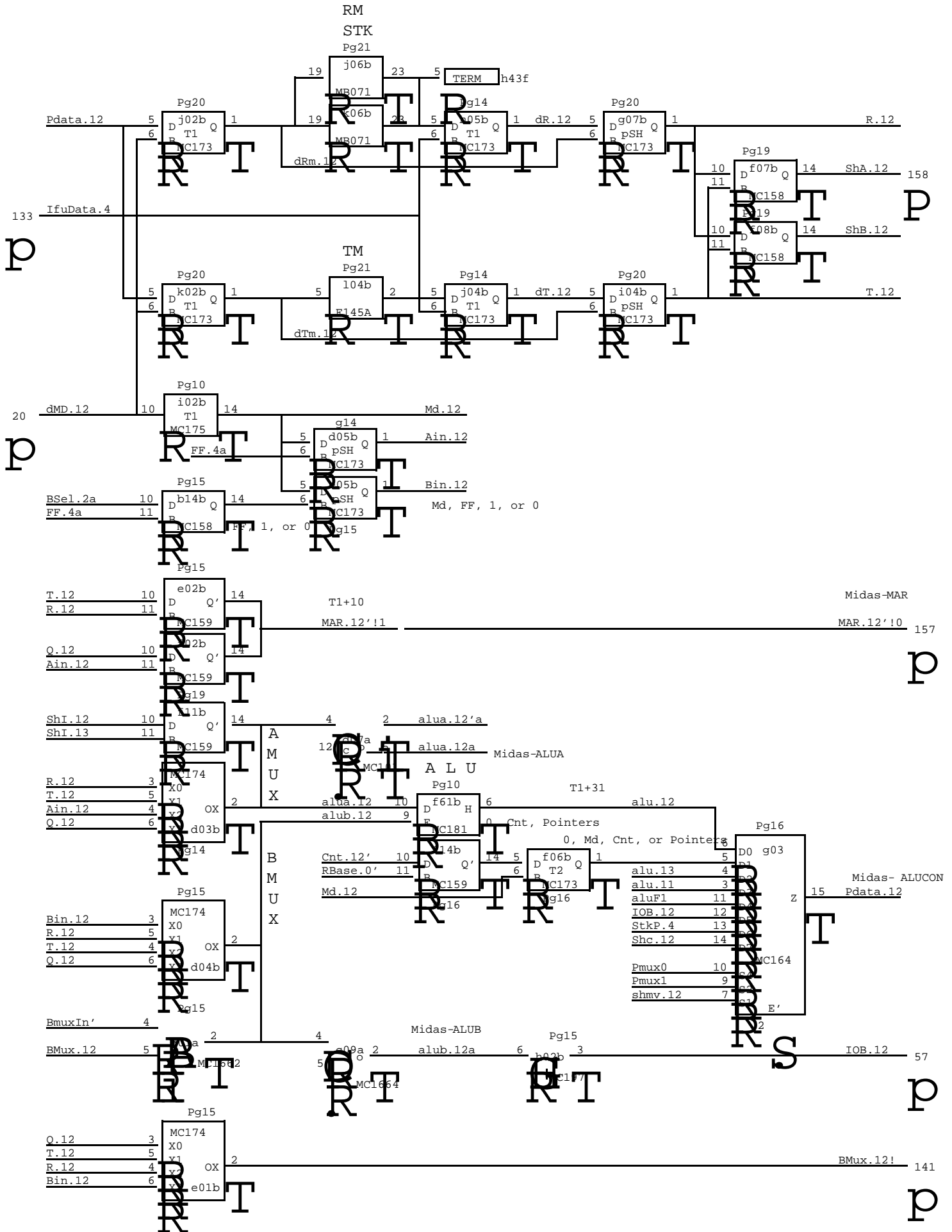




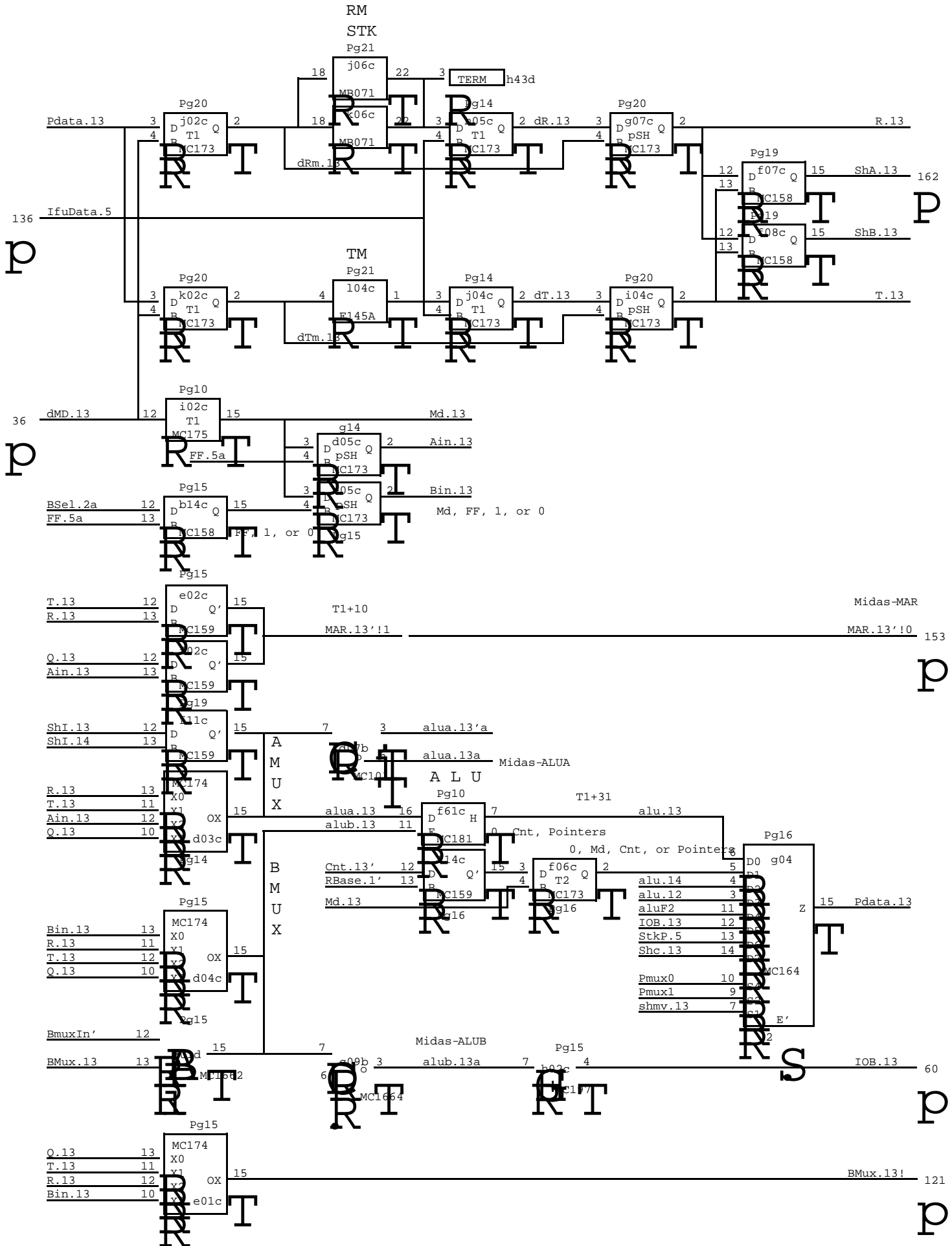
XEROX PARC	Project Dorado	Drawing BIT SLICE 09	File ProcL03.sil	Designer R Bates	Rev Ci	Date 6/27/79	Page 03
---------------	-------------------	-------------------------	---------------------	---------------------	-----------	-----------------	------------



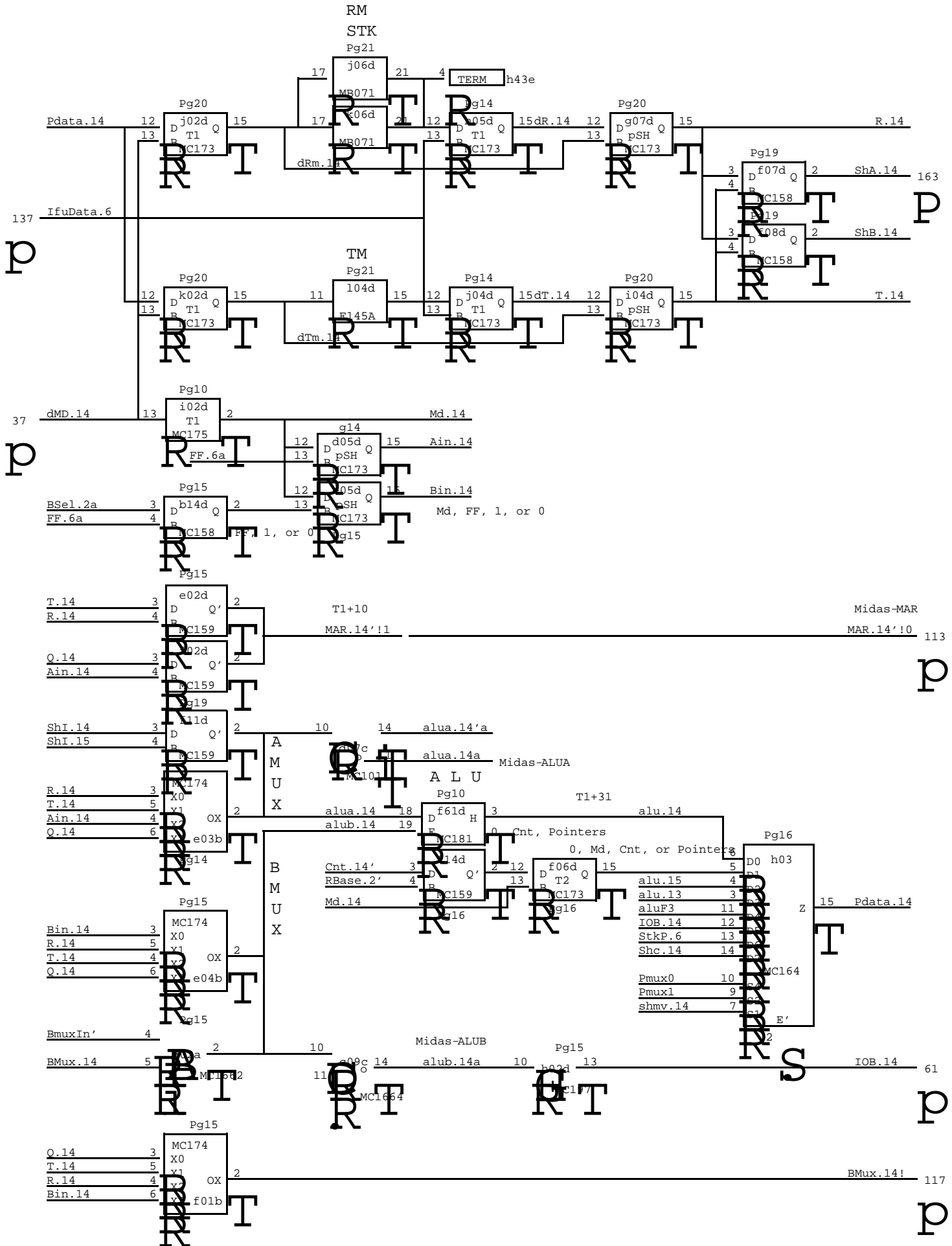


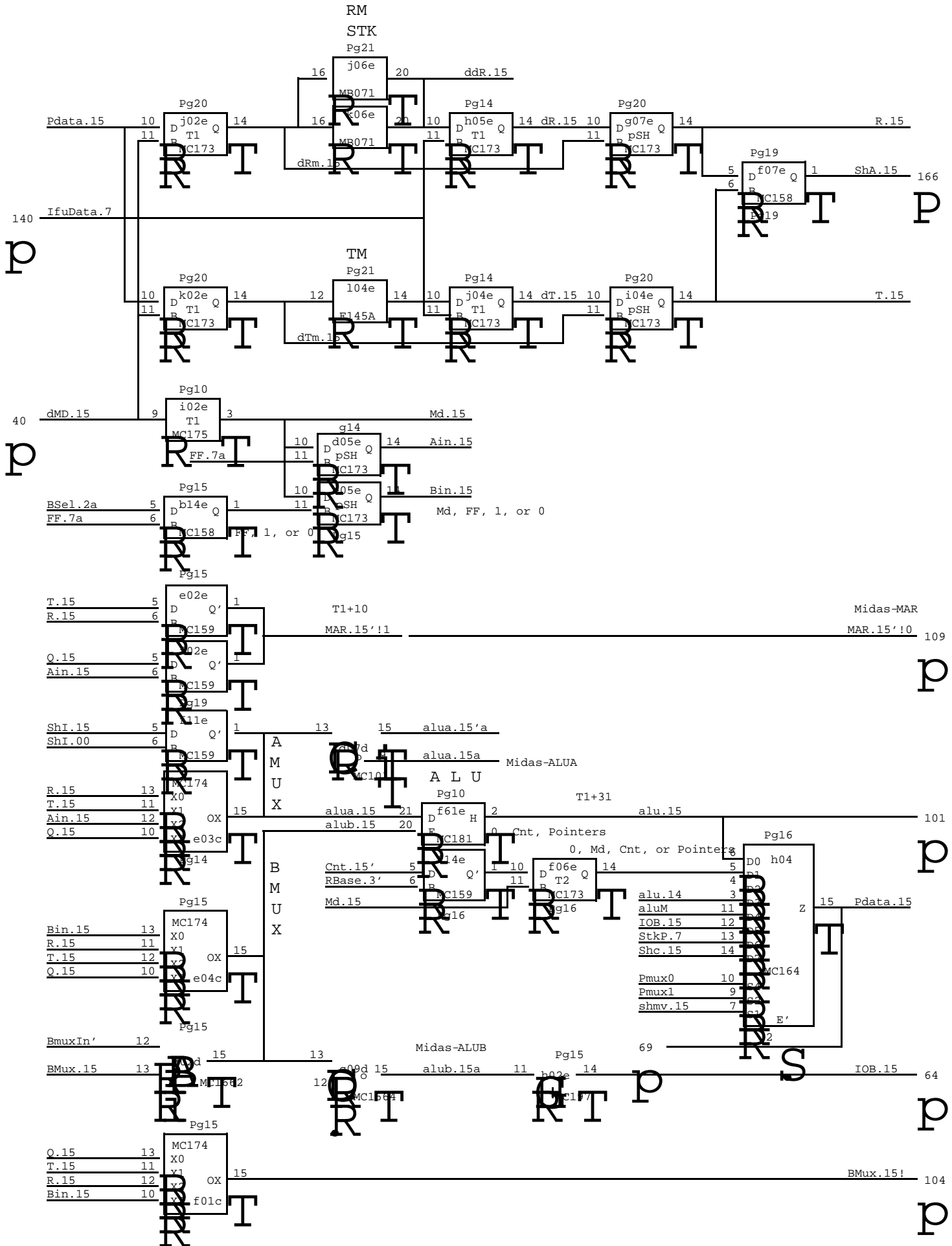


XEROX PARC	Project Dorado	Drawing BIT SLICE 12	File ProcL06.sil	Designer R Bates	Rev Ci	Date 6/27/79	Page 06
---------------	-------------------	-------------------------	---------------------	---------------------	-----------	-----------------	------------



XEROX PARC	Project Dorado	Drawing BIT SLICE 13	File ProcL07.sil	Designer R Bates	Rev Ci	Date 6/27/79	Page 07
---------------	-------------------	-------------------------	---------------------	---------------------	-----------	-----------------	------------



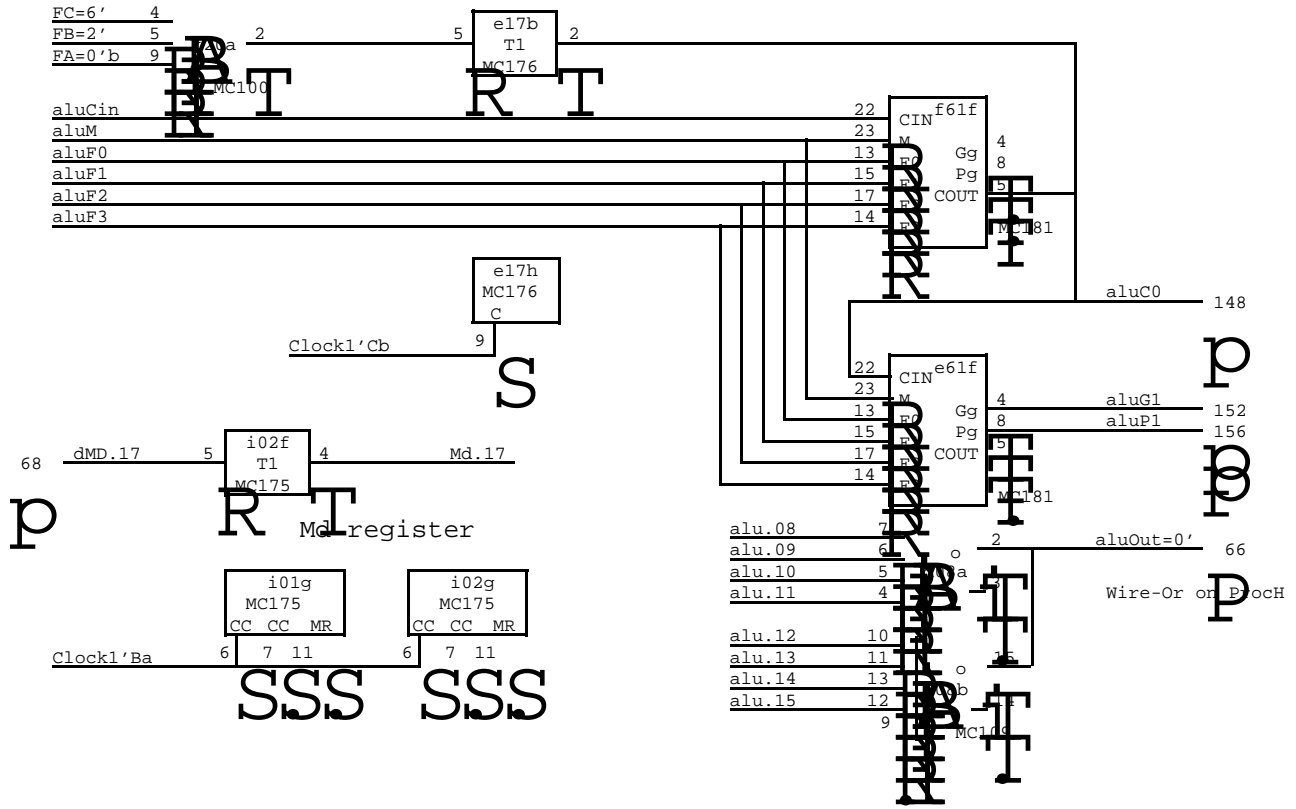


ALU delays

Logical function to output = 11.9
 Arithmetic operation to data = 20.0
 Arithmetic operation to carry = 17.9

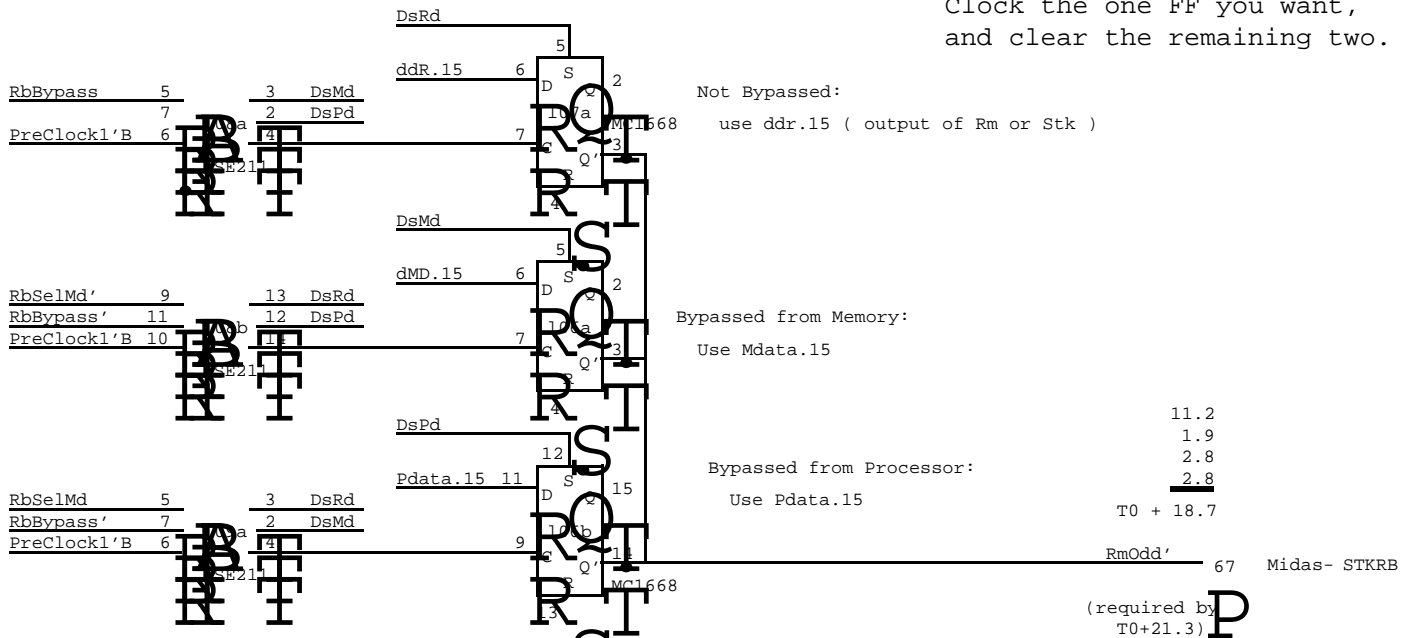
ALU output (assuming 10.2 ns to output of BMux)

Logical Function = 22.0
 Arithmetic operation to data = 30.1
 Arithmetic operation to carry = 28.0



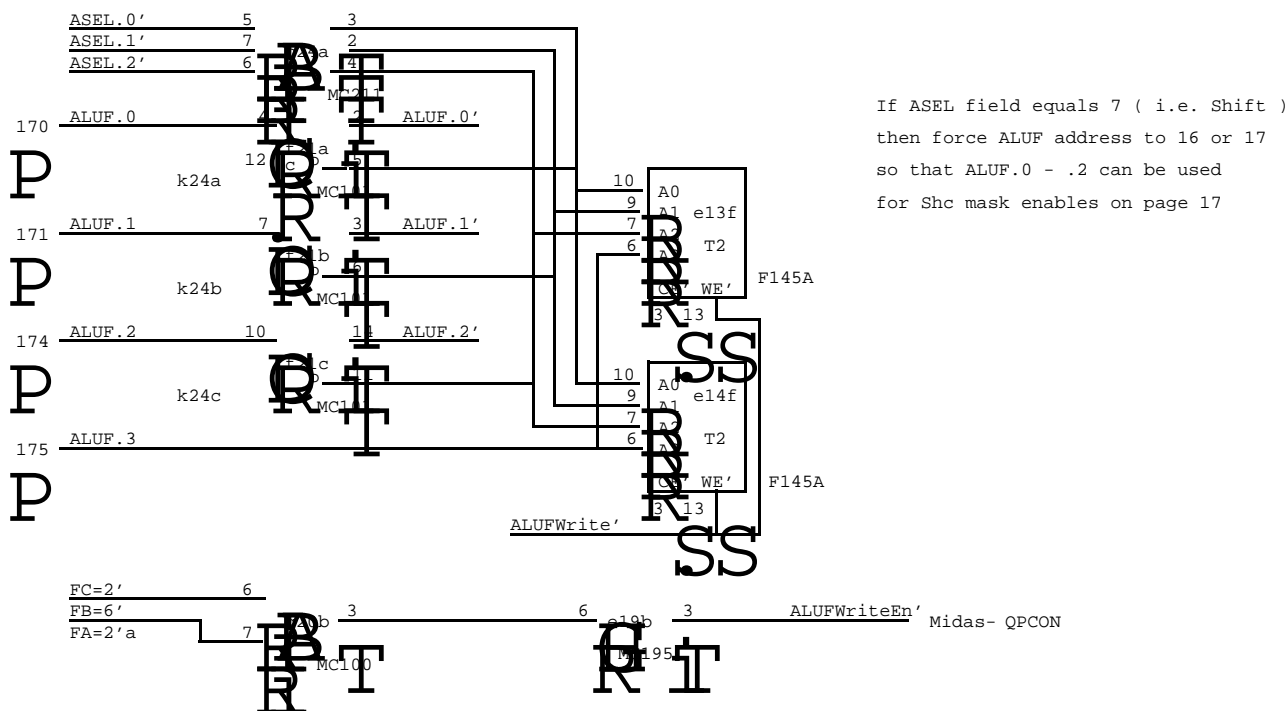
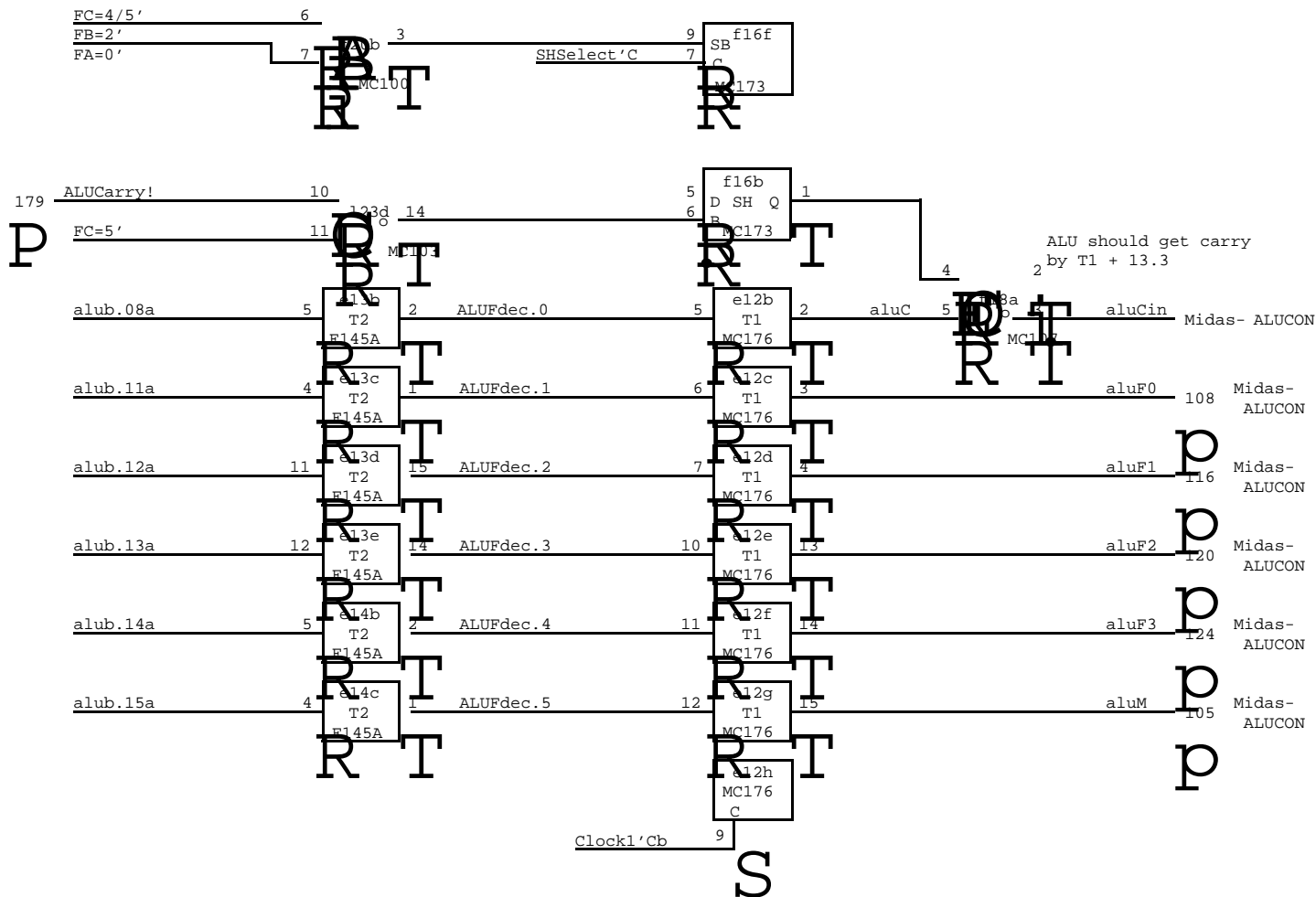
Fast Branch
 on Rm odd

Clock the one FF you want,
 and clear the remaining two.



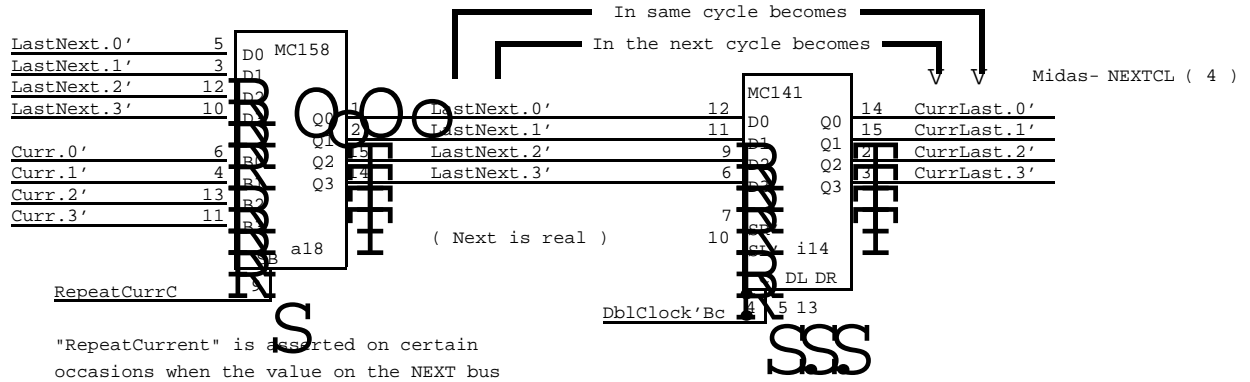
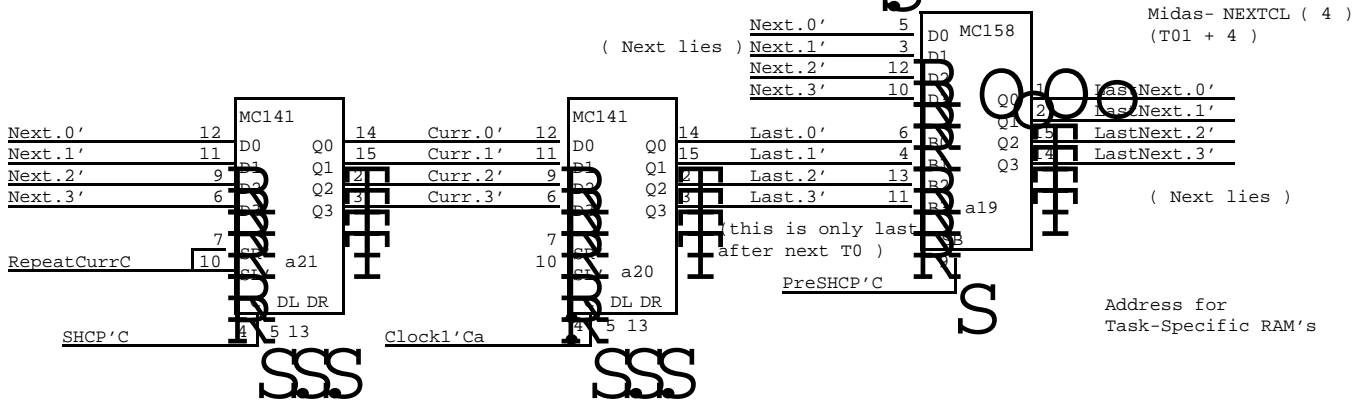
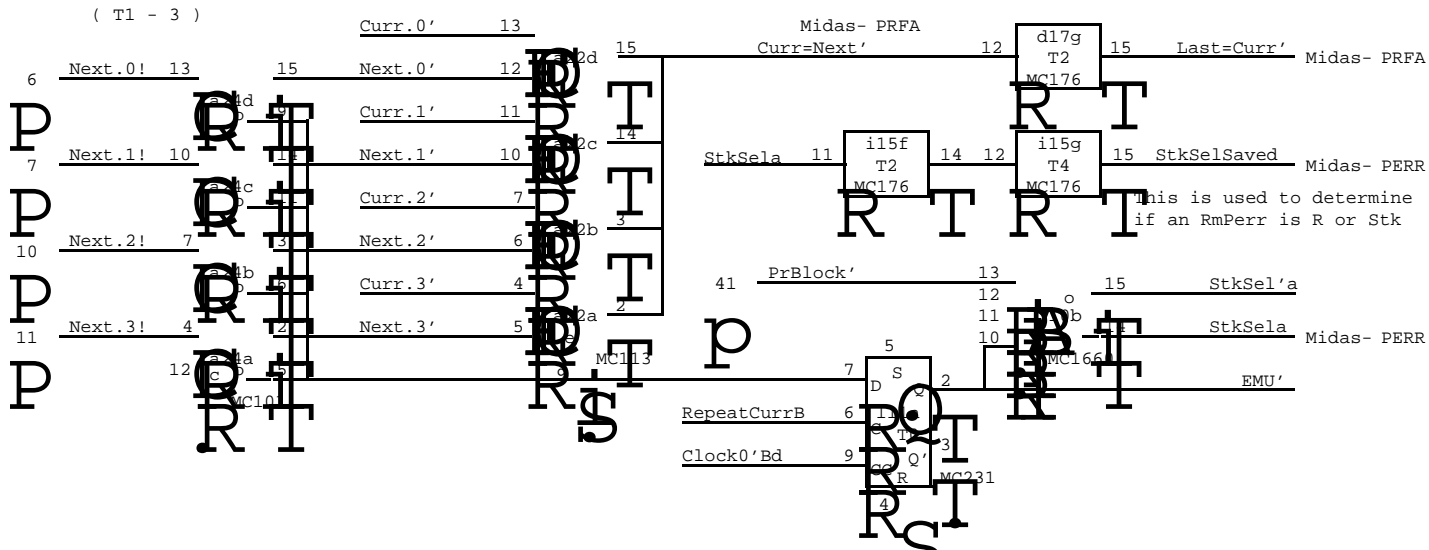
This circuit will correctly
 bypass R from Pdata or Mdata
 When "RisIfData" is in effect, the fast branch
 will be based on the contents of the addressed
 RM or STK, bypassed if necessary.

FF = Cin xor 1
 or
 FF = Cin xor Cout



NOTE: Moving k24 to f21 saves 0.4 ns

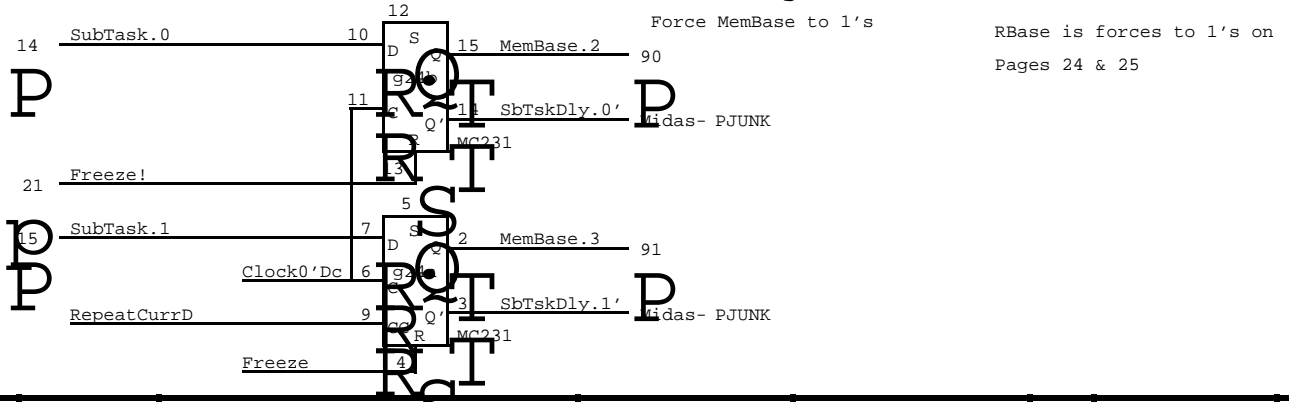
XEROX PARC	Project Dorado	Drawing ALU FUNCTION RAM	File ProcL11.sil	Designer R Bates	Rev Ci	Date 6/27/79	Page 11
---------------	-------------------	-----------------------------	---------------------	---------------------	-----------	-----------------	------------

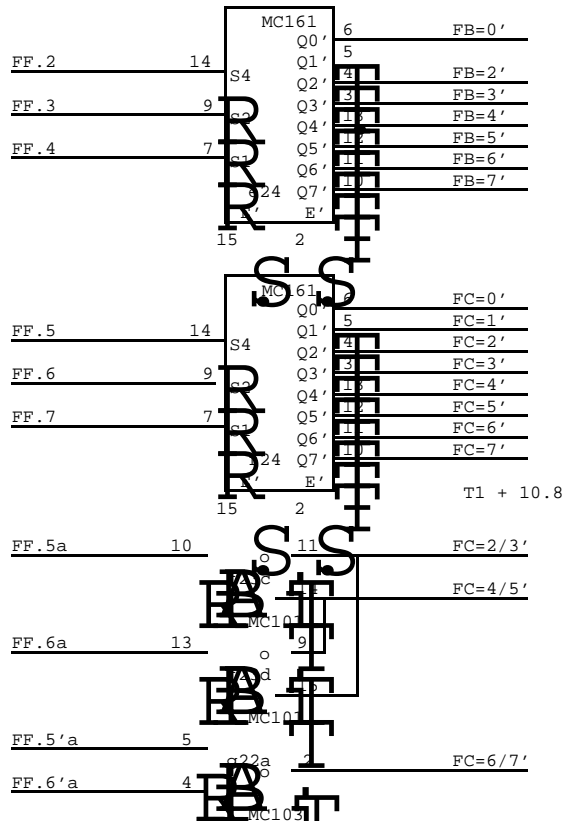
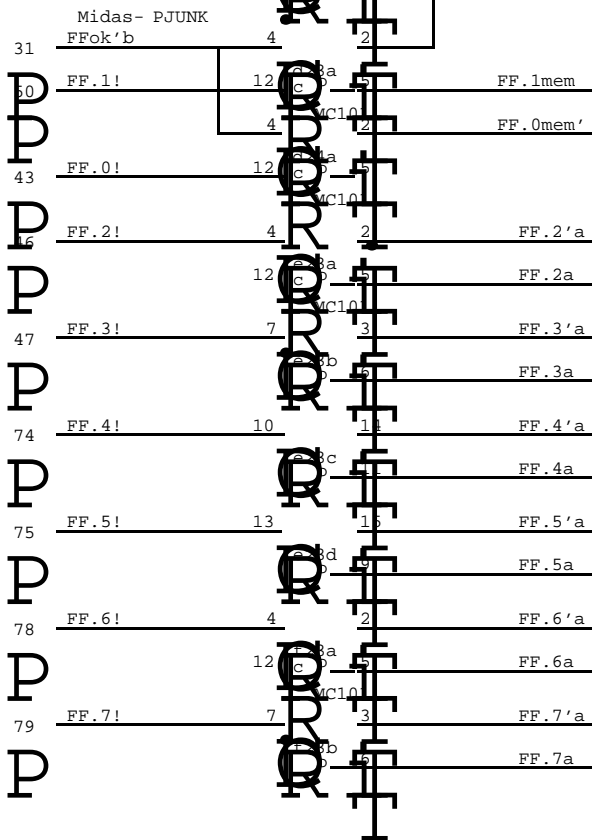
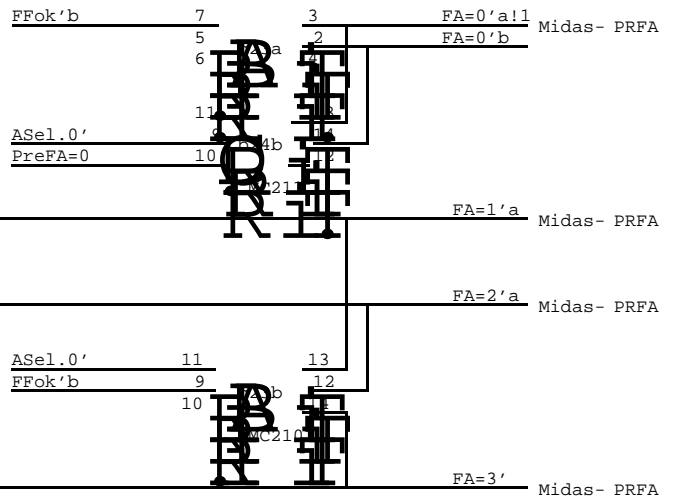
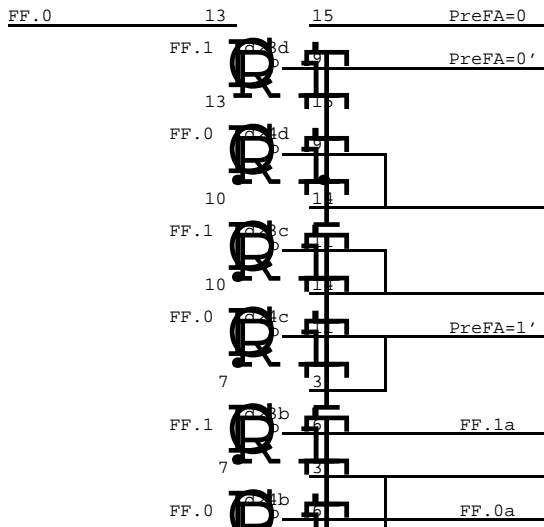
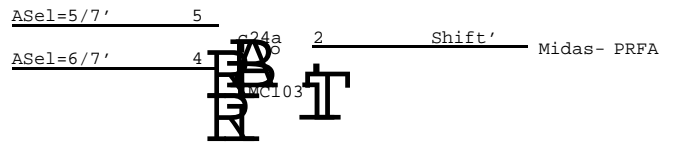
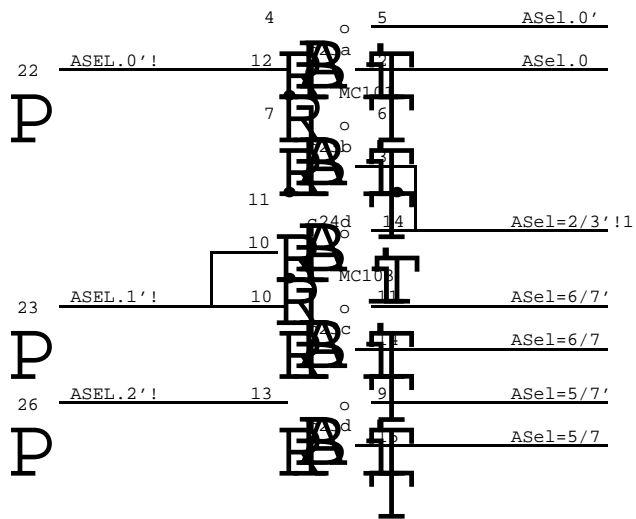


"RepeatCurrent" is asserted on certain occasions when the value on the NEXT bus may be invalid ("Next Lies") due to the combination of Block and Hold.

Task number tracking logic

Sub-Task wire-or logic





AMux decoding

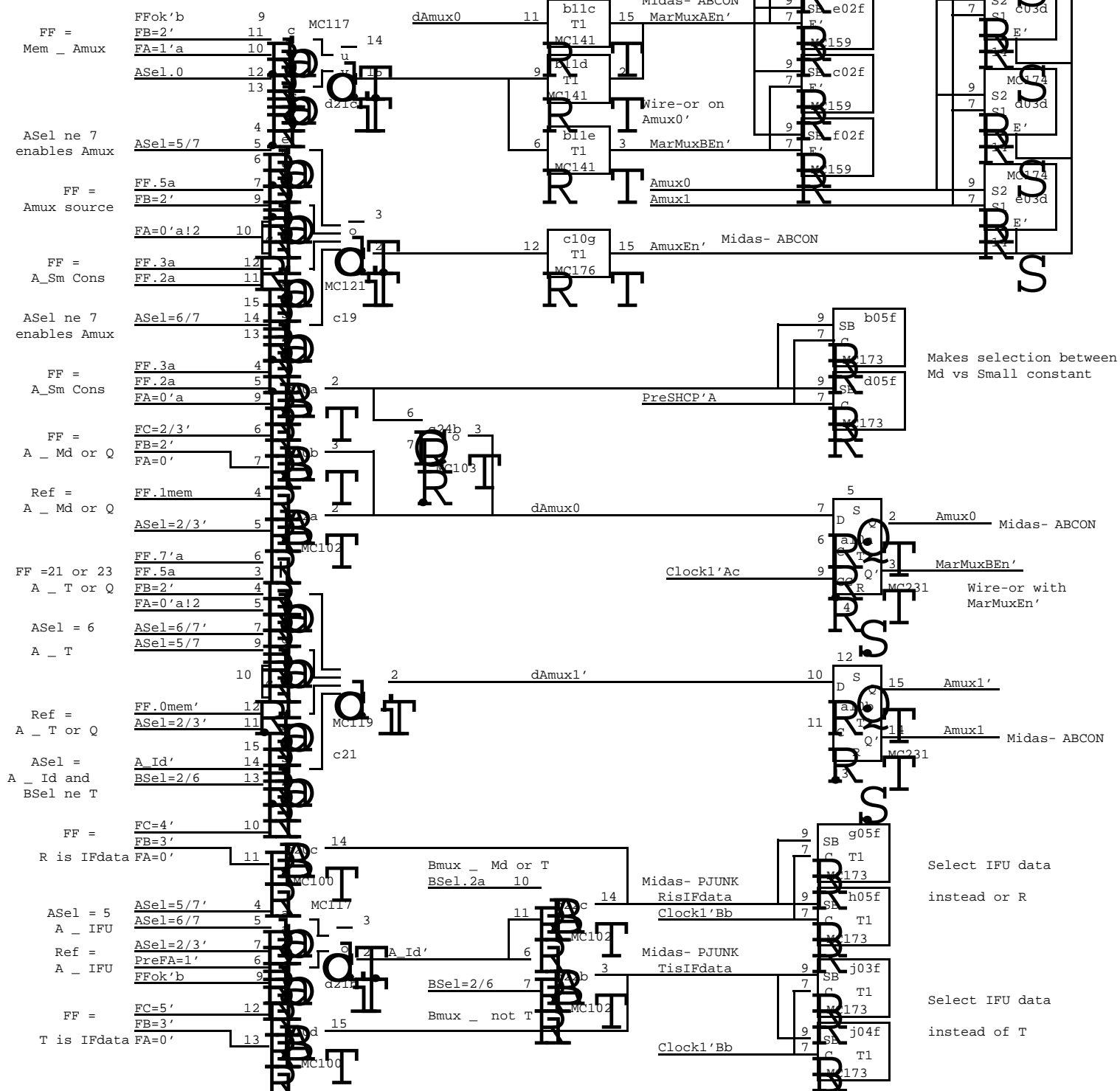
AMux _ FF:	FF=0-17 *
AMux _ T:	FF=021 * ASEL=2or3 & FAmem=3 ASEL=6
AMux _ Md:	FF=022 * ASEL=2or3 & FAmem=0
AMux _ Q:	FF=023 * ASEL=2or3 & FAmem=2
AMux _ IFU:	ASEL=5 ASEL=2or3 & FAmem=1
AMux _ R:	FF=020 * None of the above

AMux encoding

Mux Input	Source
0	R or IFU data
1	T or IFU data
2	Md or Small Const.
3	Q

* The Amux is disabled by ASEL=7 unless one of these codes are in effect

NOTE: ASEL selects and FF selects for the AMUX are "OR'd" by this hardware. Thus ASEL codes selecting non-Rm sources of Amux must not be used when an FF specifies an ASEL source. Likewise for FF when ASEL specifies non-Rm AMux sources.

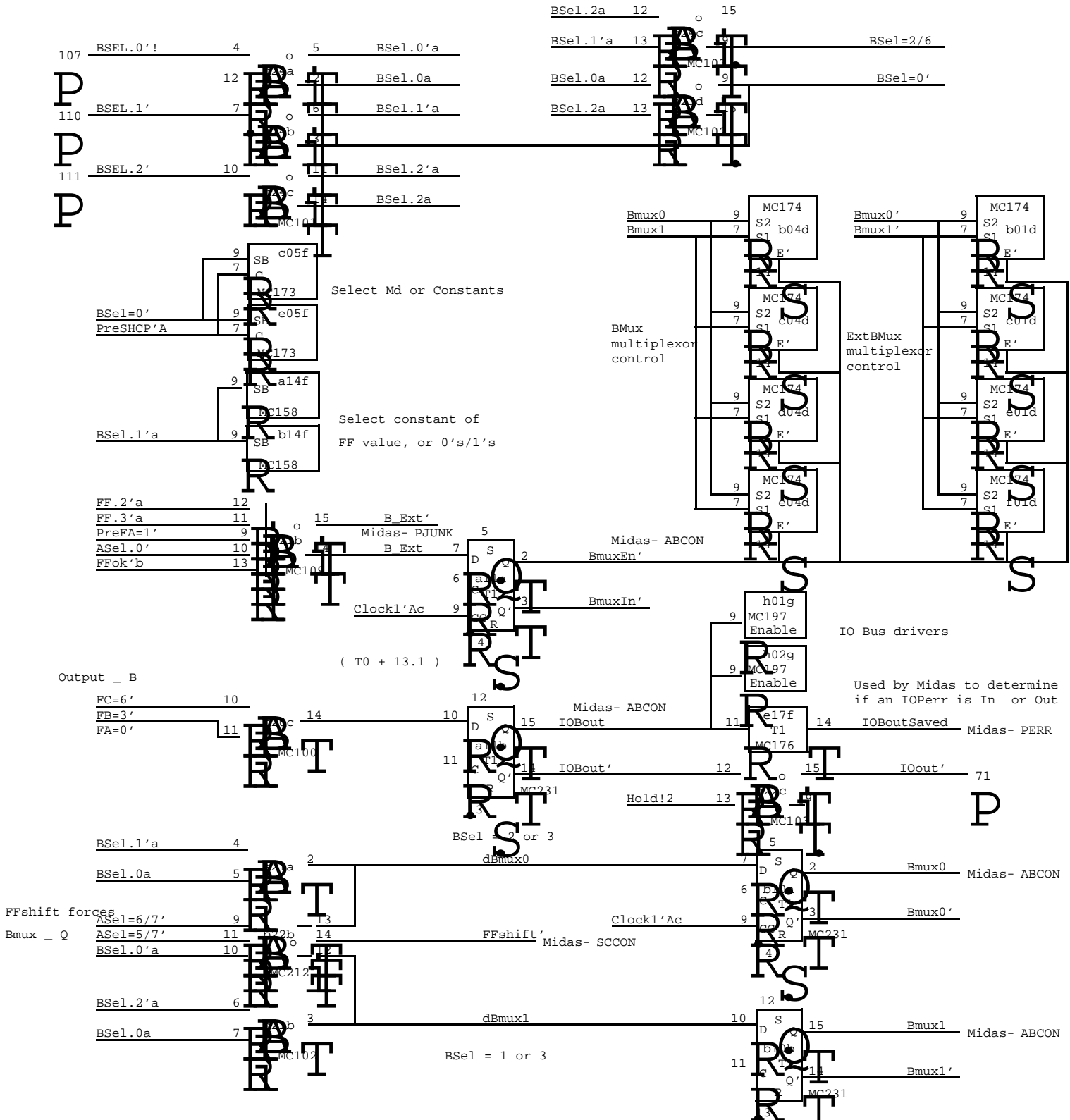


BSEL field decoding

BSEL	INTERNAL	EXTERNAL
0	Md	--
1	R	--
2	T	Hold_B
3	Q	Q _ B
4	0,,FF	--
5	377,FF	--
6	FF,,0	--
7	FF,,377	--

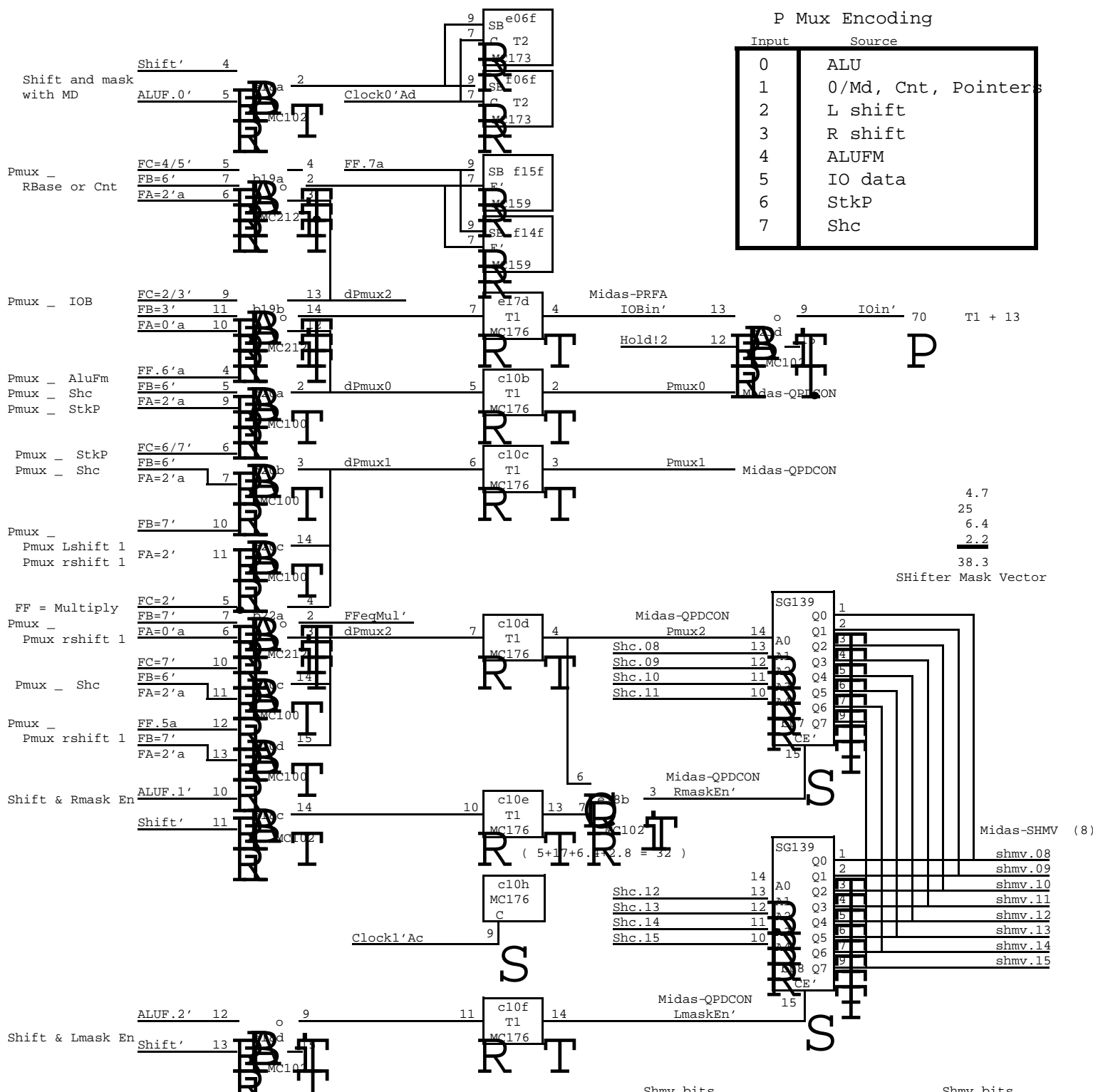
BMux encoding

BMux	
0	Md or Constant
1	R
2	T
3	Q

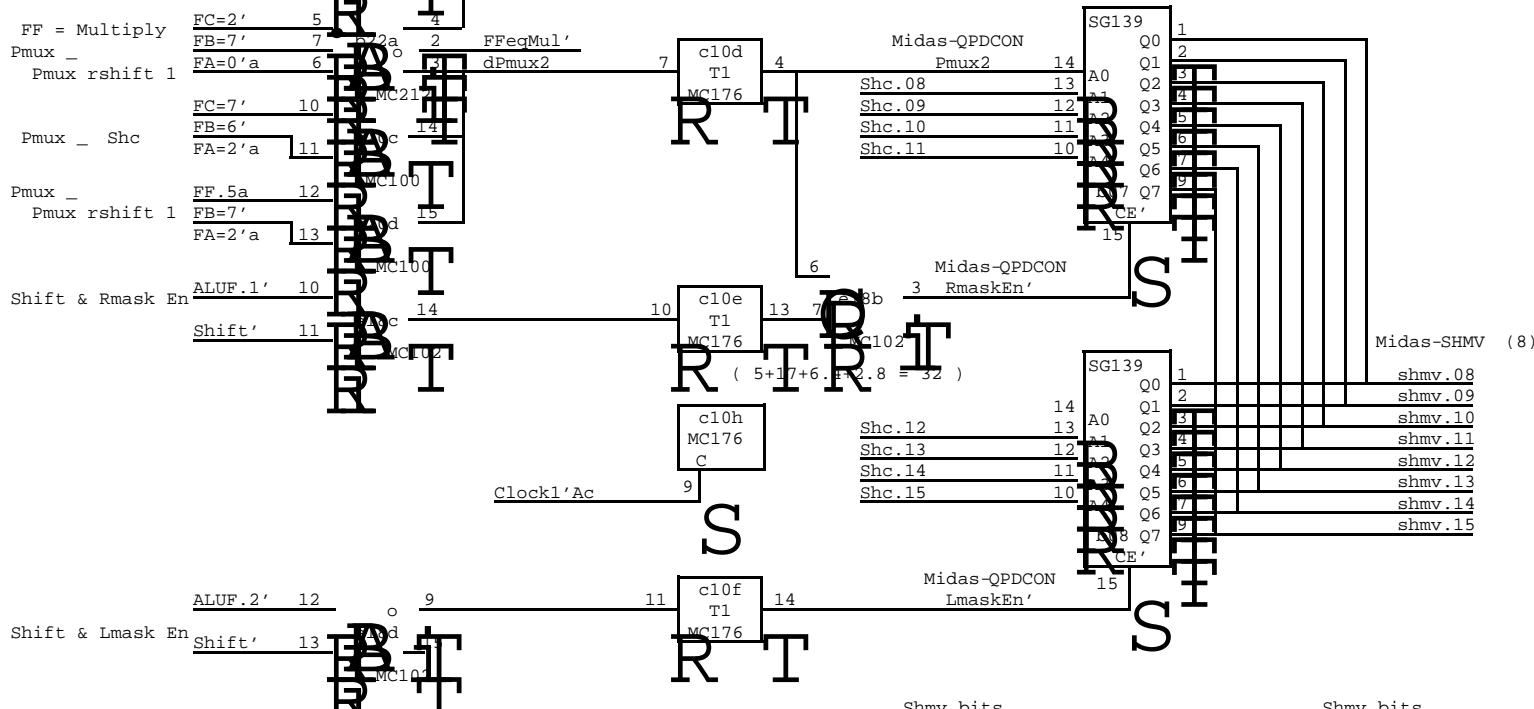


P Mux Encoding

Input	Source
0	ALU
1	0/Md, Cnt, Pointers
2	L shift
3	R shift
4	ALUFM
5	IO data
6	StkP
7	Shc

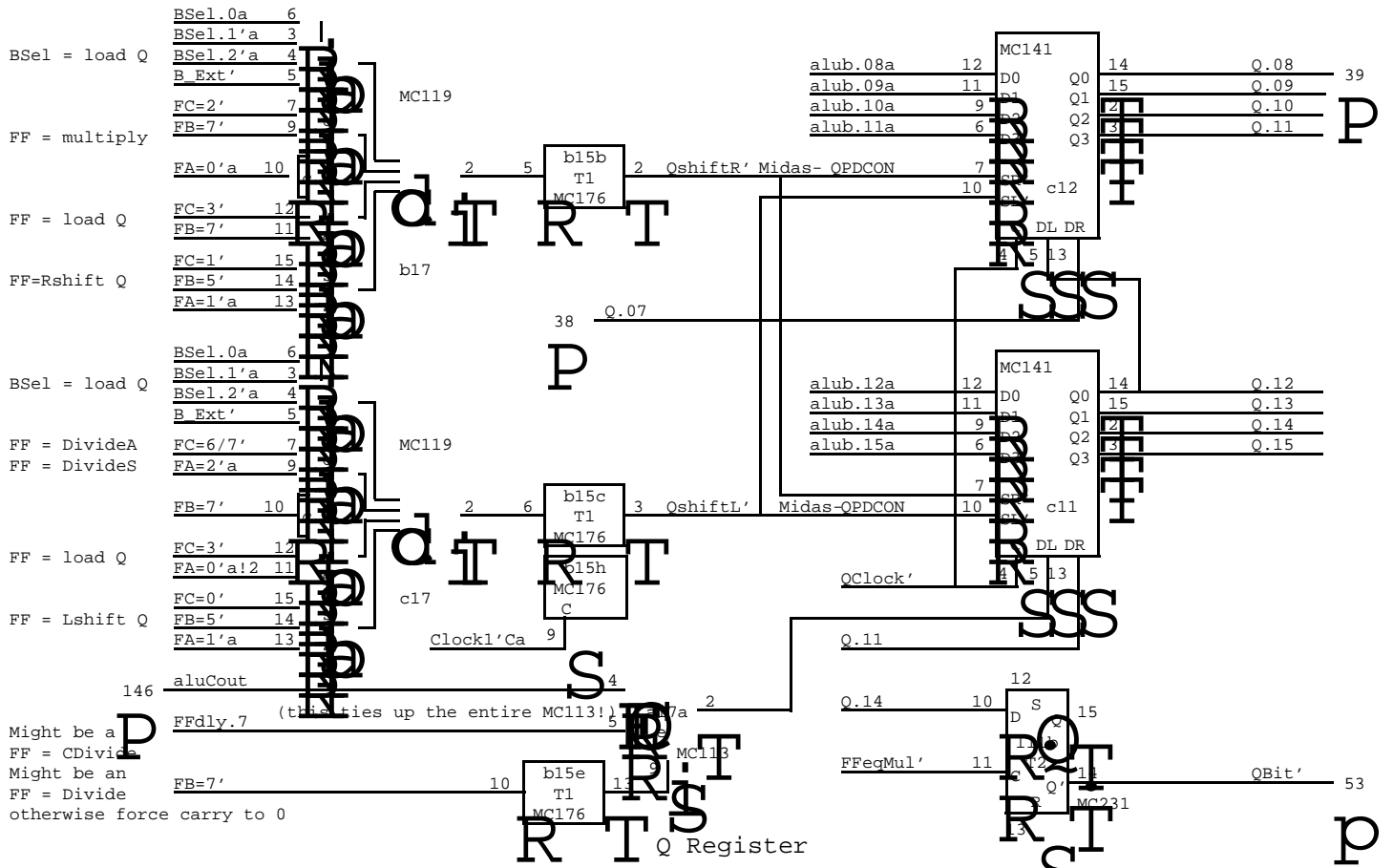


4.7
25
6.4
2.2
38.3
Shifter Mask Vector

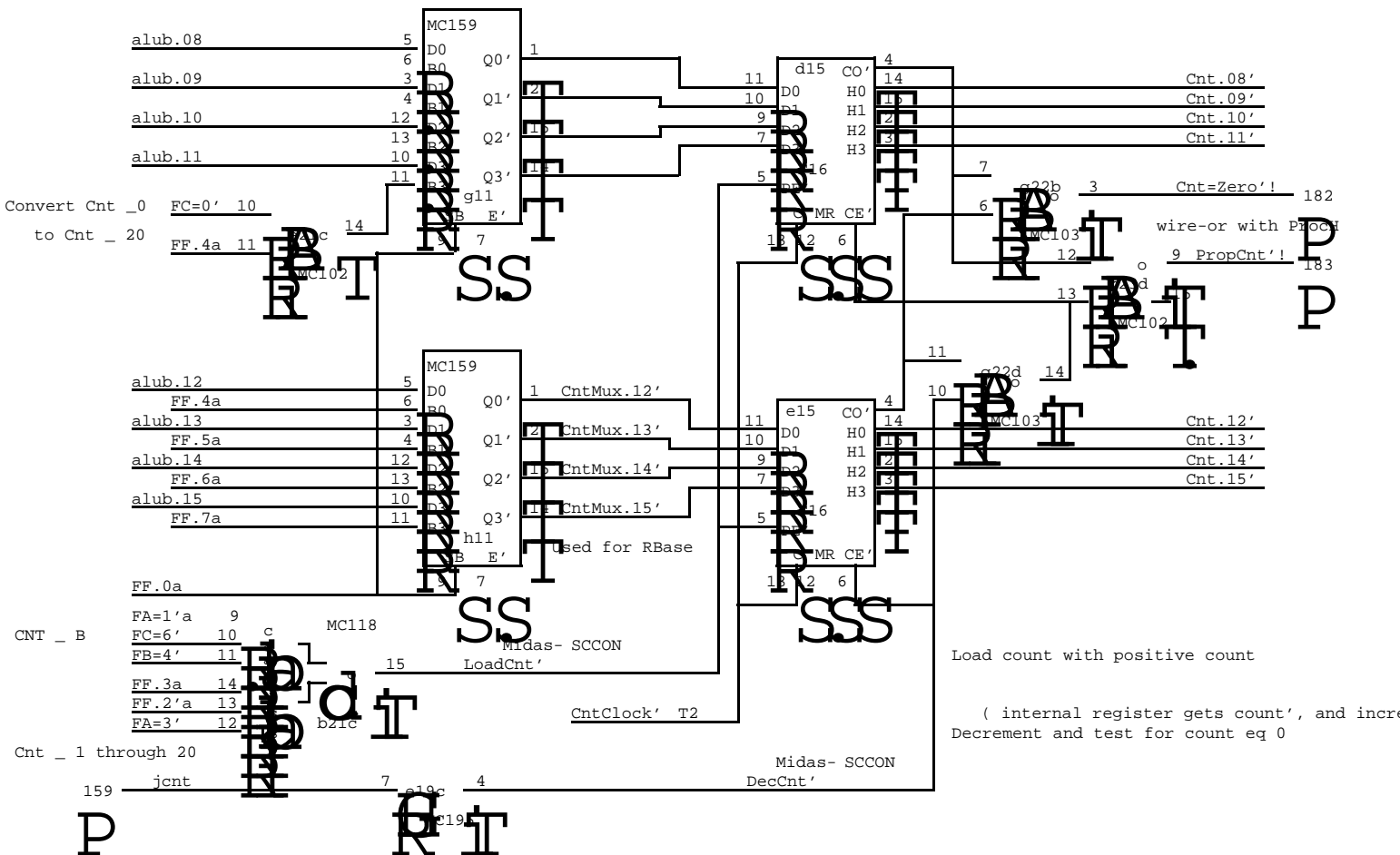


Lmask	Shmv bits								Rmask	Shmv bits							
	08	09	10	11	12	13	14	15		08	09	10	11	12	13	14	15
Shc.12-15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
	2	0	0	0	0	0	0	0	0	2	0	0	0	0	0	1	
	3	0	0	0	0	0	0	0	0	3	0	0	0	0	0	1	
	4	0	0	0	0	0	0	0	0	4	0	0	0	0	1	1	
	5	0	0	0	0	0	0	0	0	5	0	0	0	1	1	1	
	6	0	0	0	0	0	0	0	0	6	0	0	1	1	1	1	
	7	0	0	0	0	0	0	0	0	7	0	1	1	1	1	1	
	10	0	0	0	0	0	0	0	0	10	1	1	1	1	1	1	
	11	1	0	0	0	0	0	0	0	11	1	1	1	1	1	1	
	12	1	1	0	0	0	0	0	0	12	1	1	1	1	1	1	
	13	1	1	1	0	0	0	0	0	13	1	1	1	1	1	1	
	14	1	1	1	1	0	0	0	0	14	1	1	1	1	1	1	
	15	1	1	1	1	1	0	0	0	15	1	1	1	1	1	1	
	16	1	1	1	1	1	1	0	0	16	1	1	1	1	1	1	
	17	1	1	1	1	1	1	0	0	17	1	1	1	1	1	1	
20-37	1	1	1	1	1	1	1	1	1	20-37	1	1	1	1	1	1	

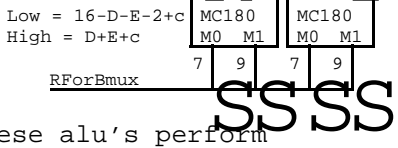
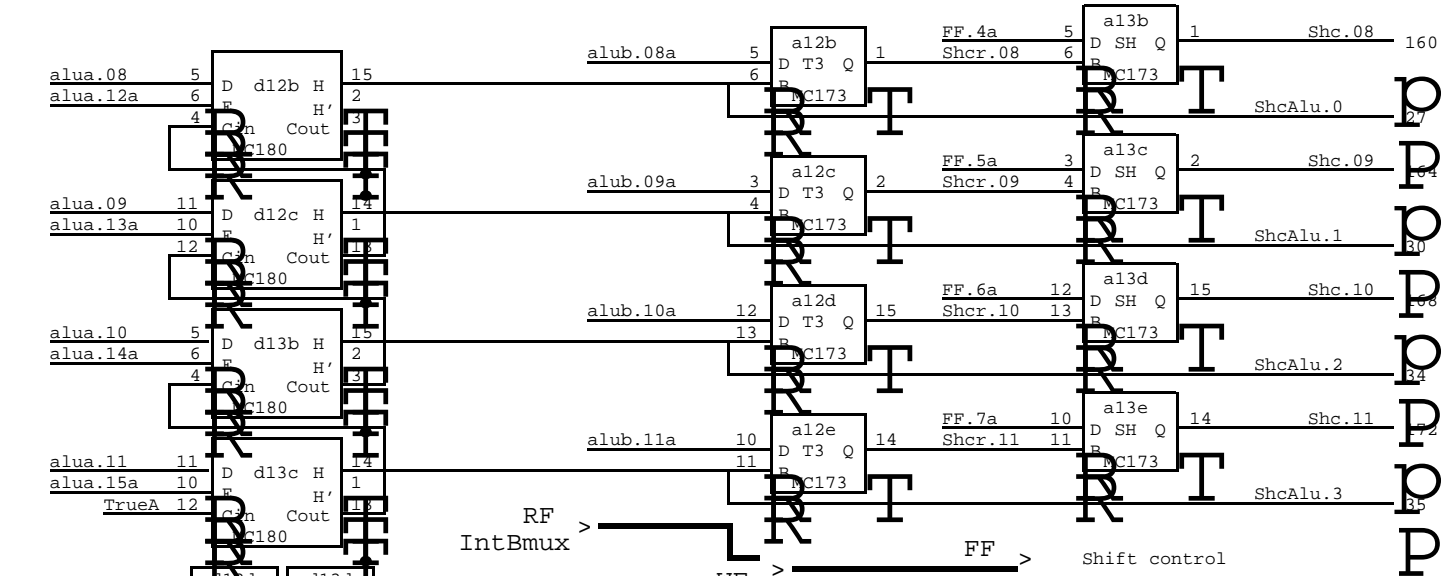
NOTE: The prom patterns are designed so that a one into address bit 0 will produce all 1's on the output. This allows the odd address inputs to Pmux to be selected.



Count Register



RIGHT mask

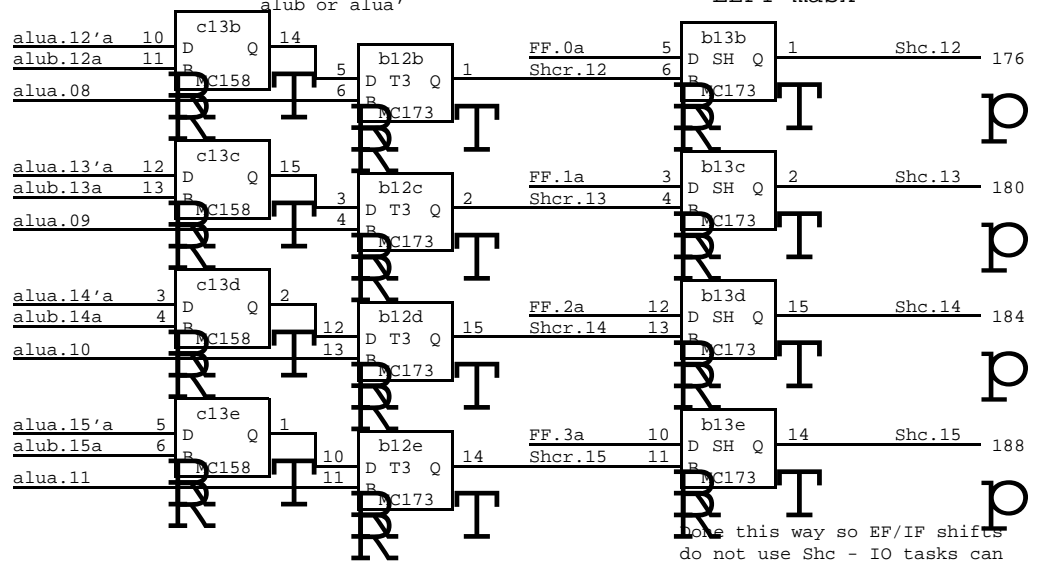


These alu's perform

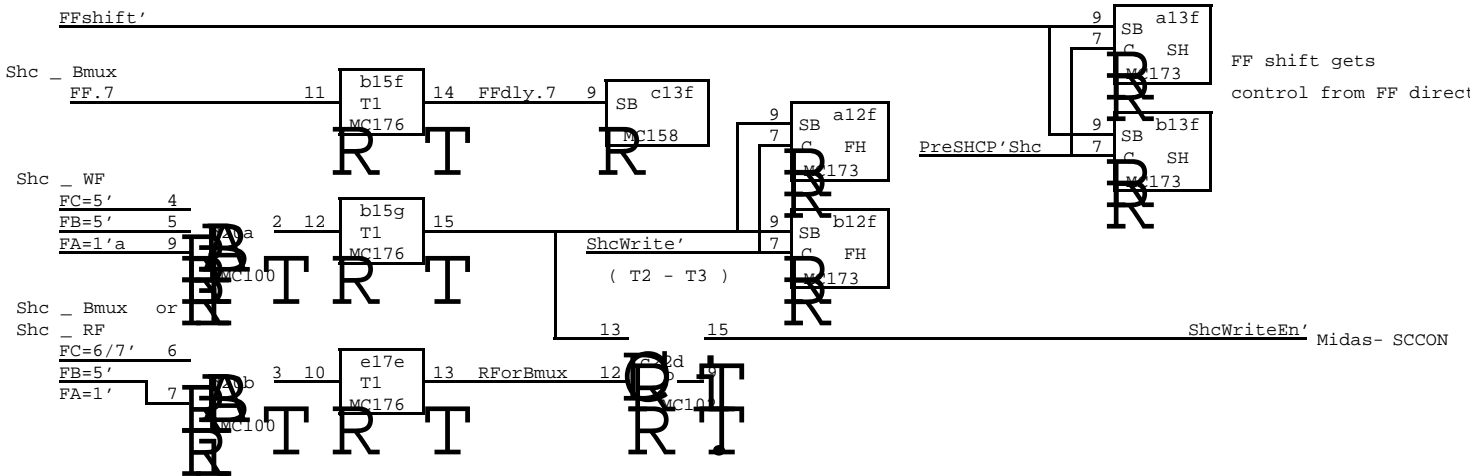
	RF	WF
Shift Count	P+S+1	16-P-S-1
Right Mask	Don't Care	16-P-S-1
LeftMask	16-S-1 (ie S')	P

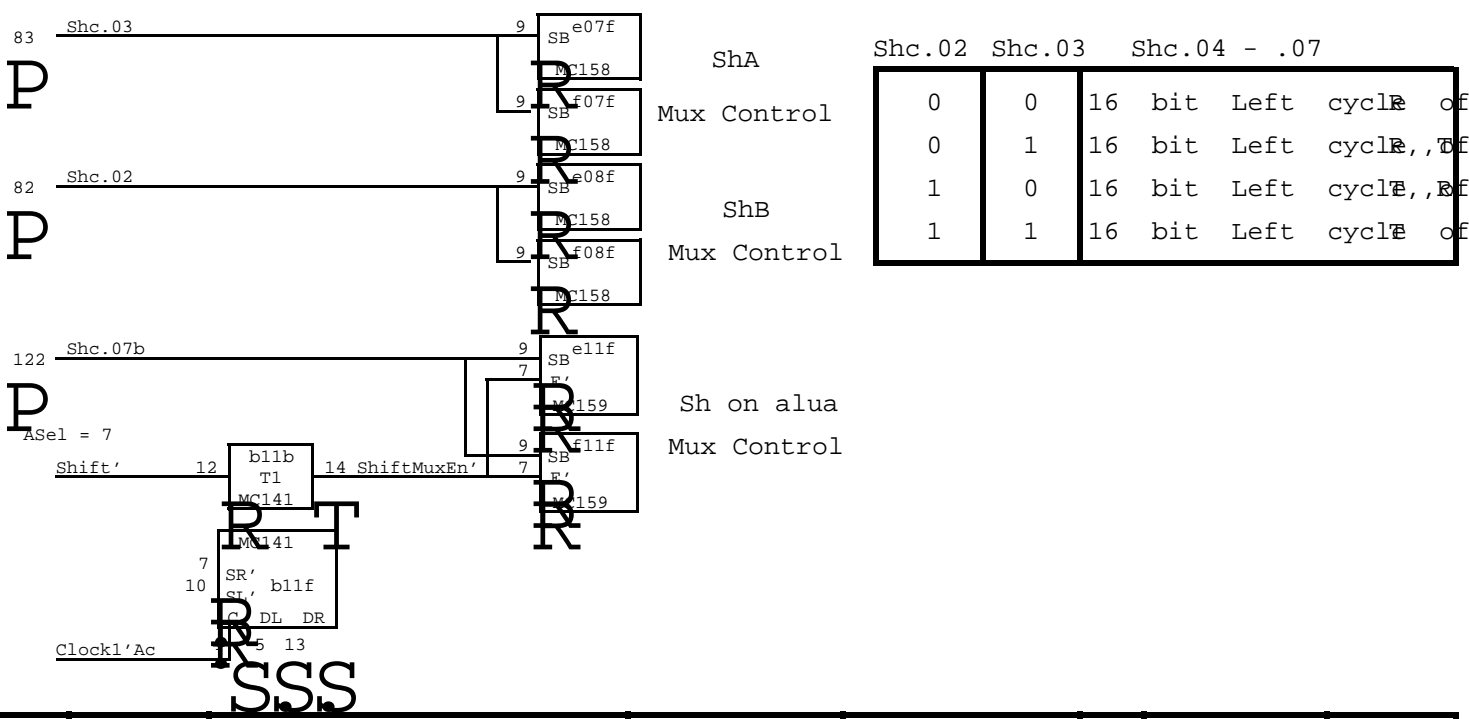
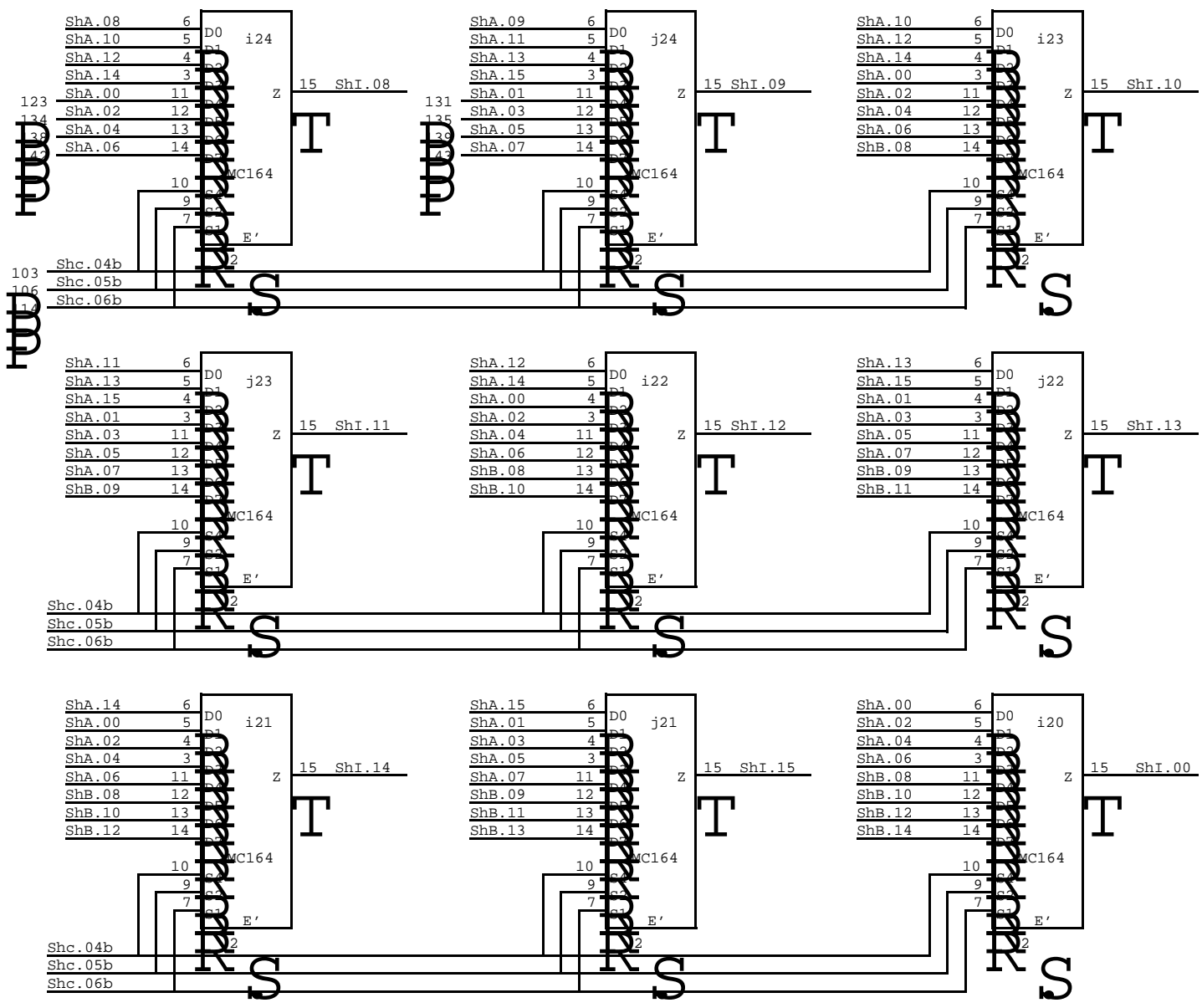
"P" = 8:11
"S" = 12:15

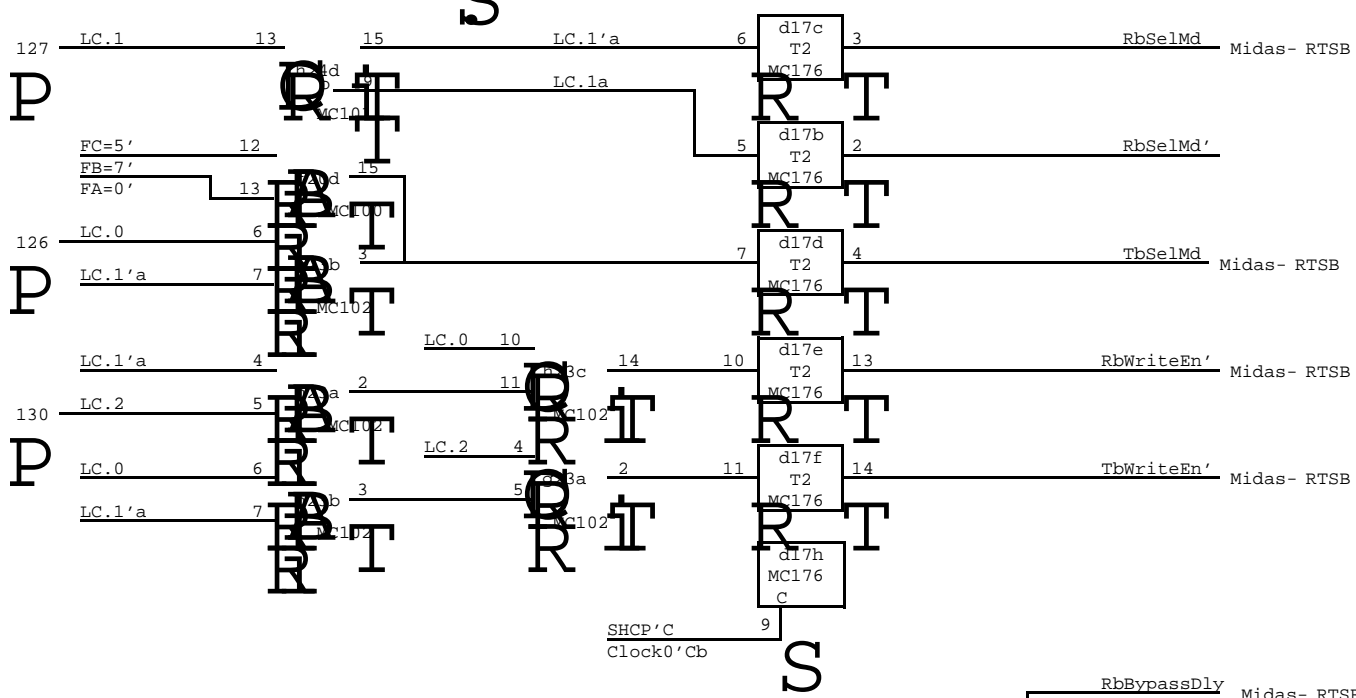
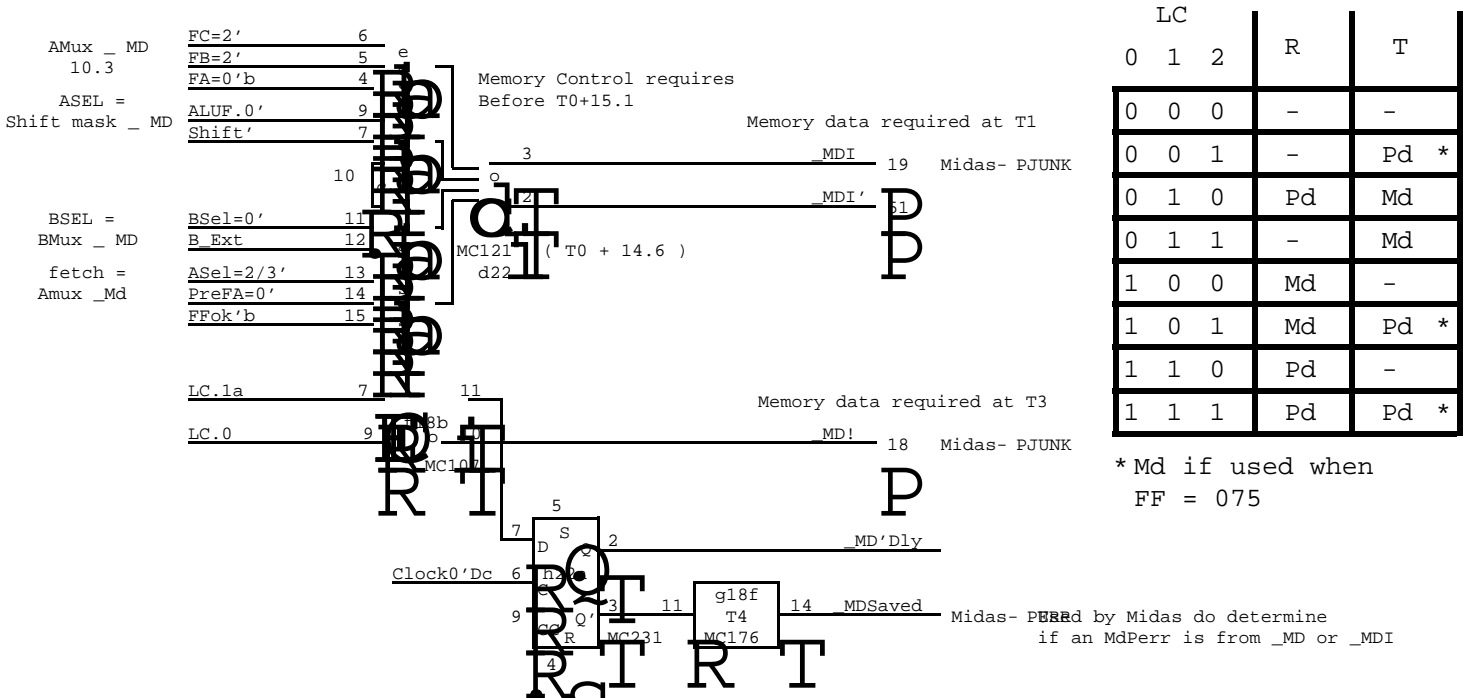
LEFT mask



Done this way so EF/IF shifts do not use Shc - IO tasks can use the shifter without saving and restoring Shc.

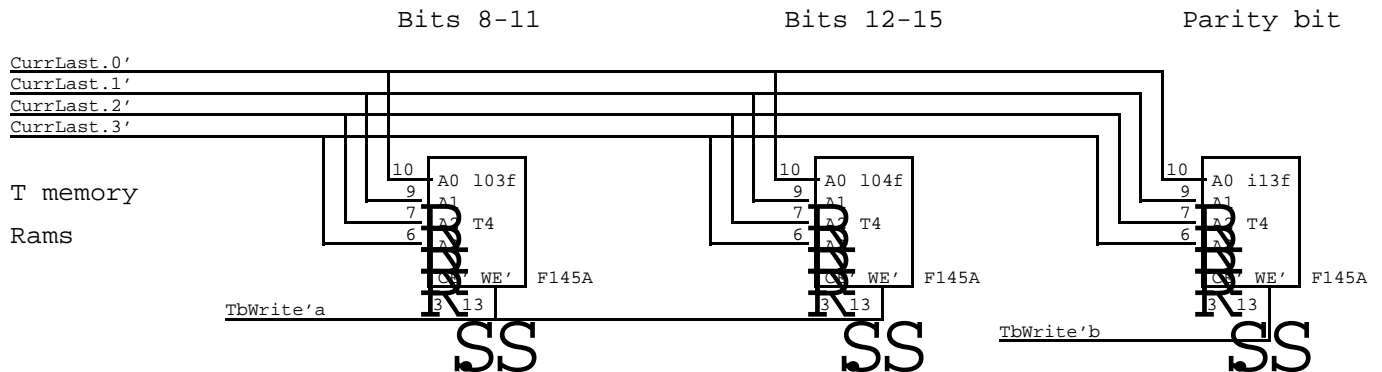
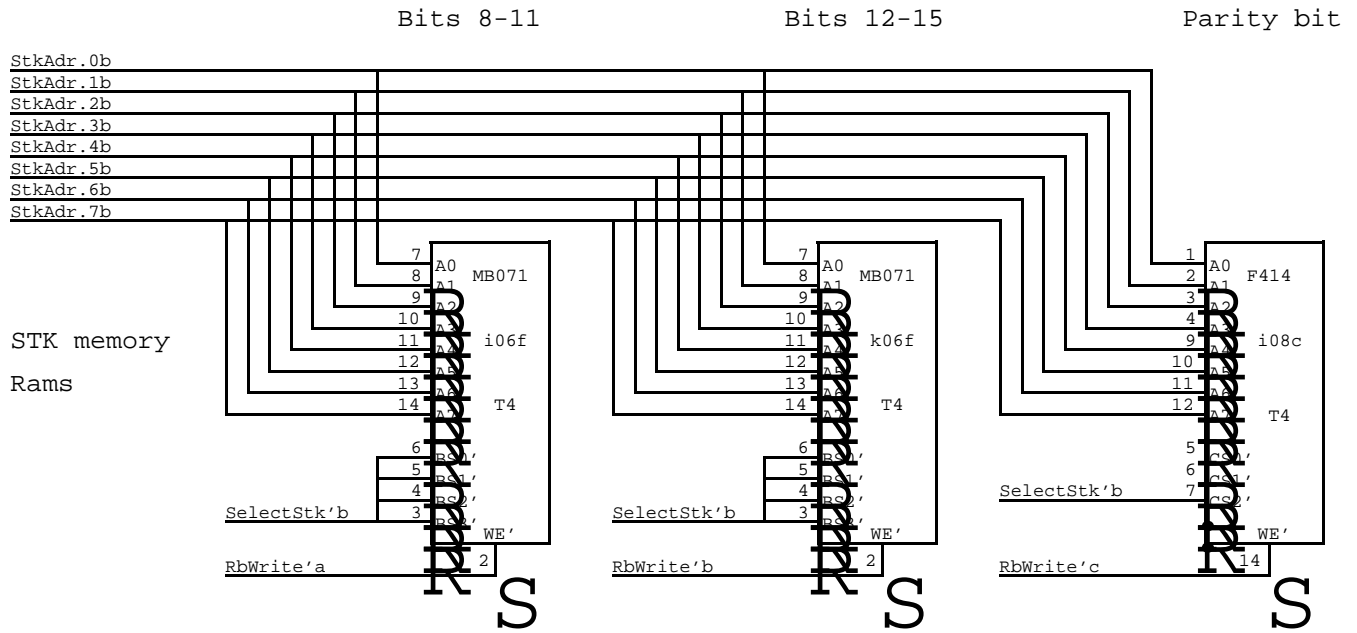
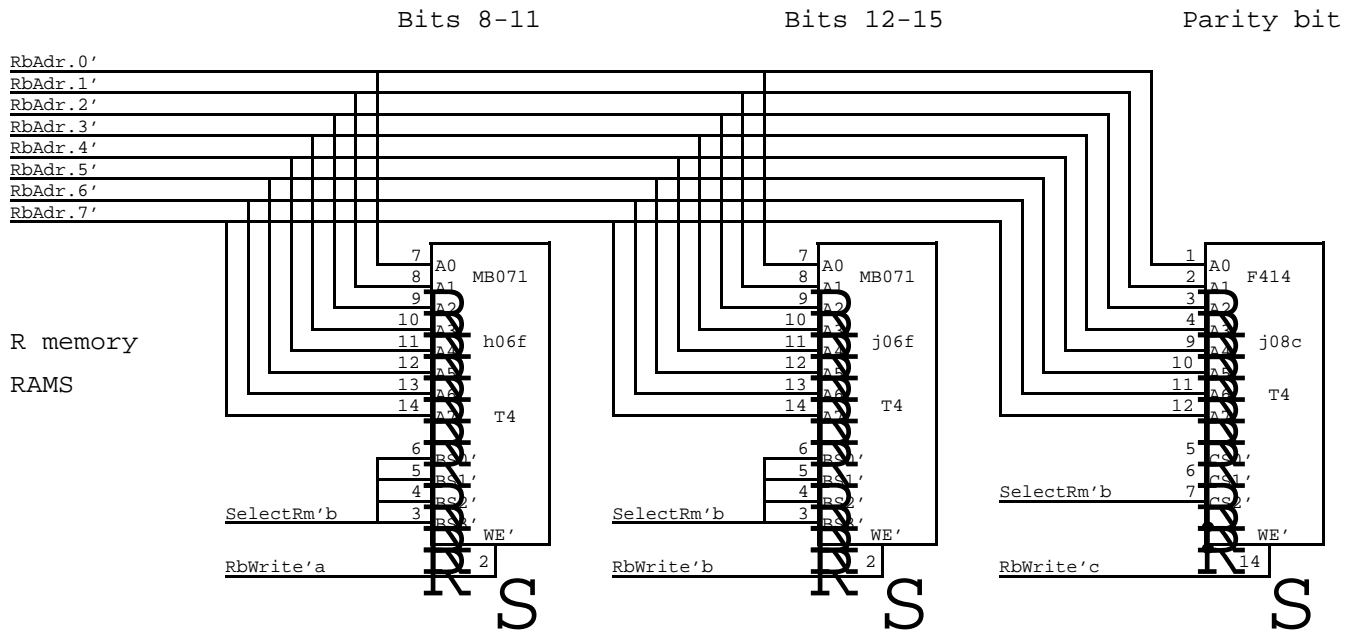


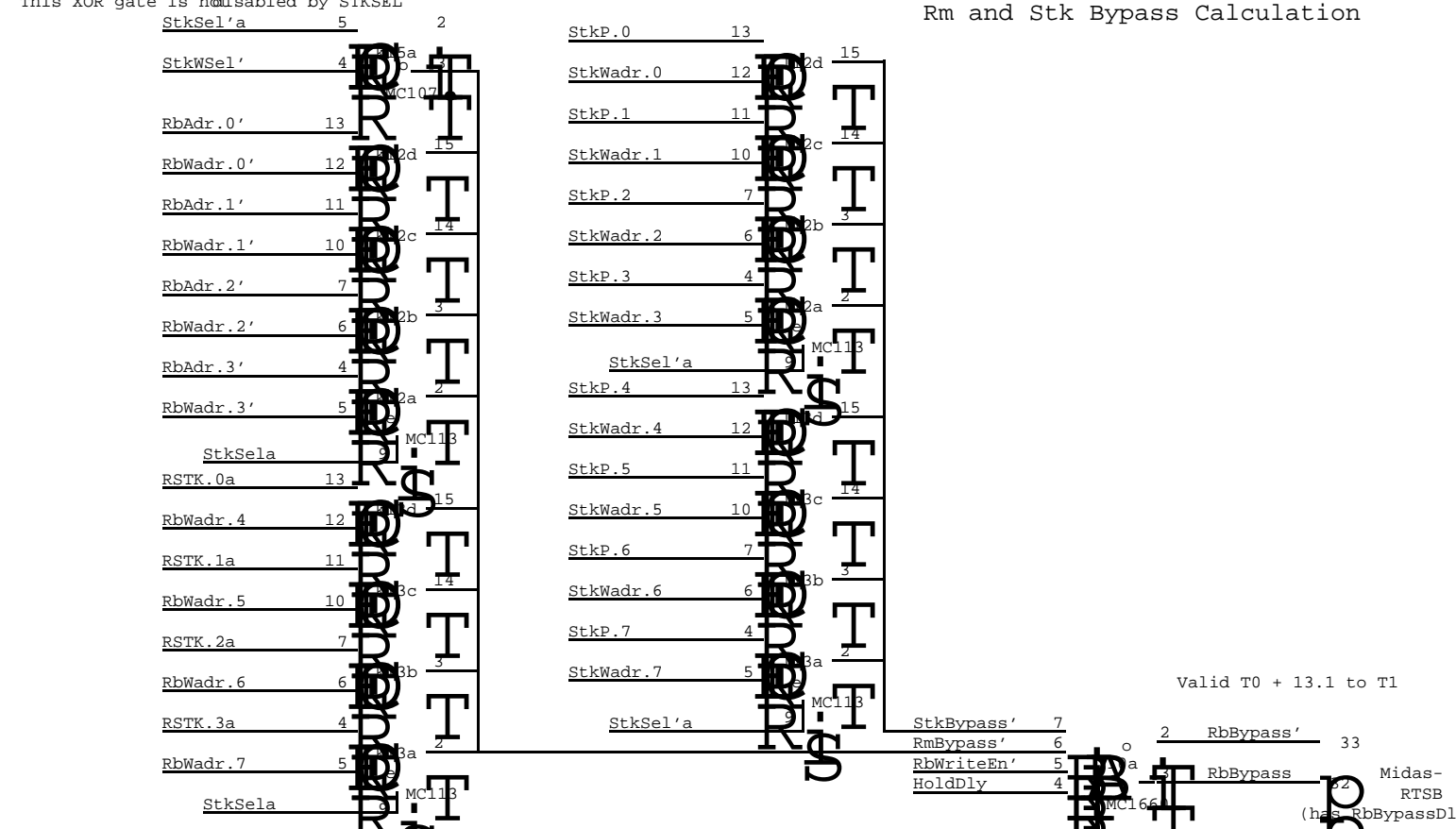
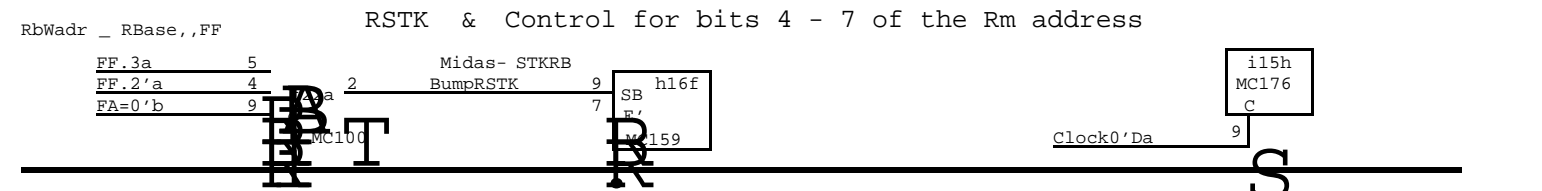
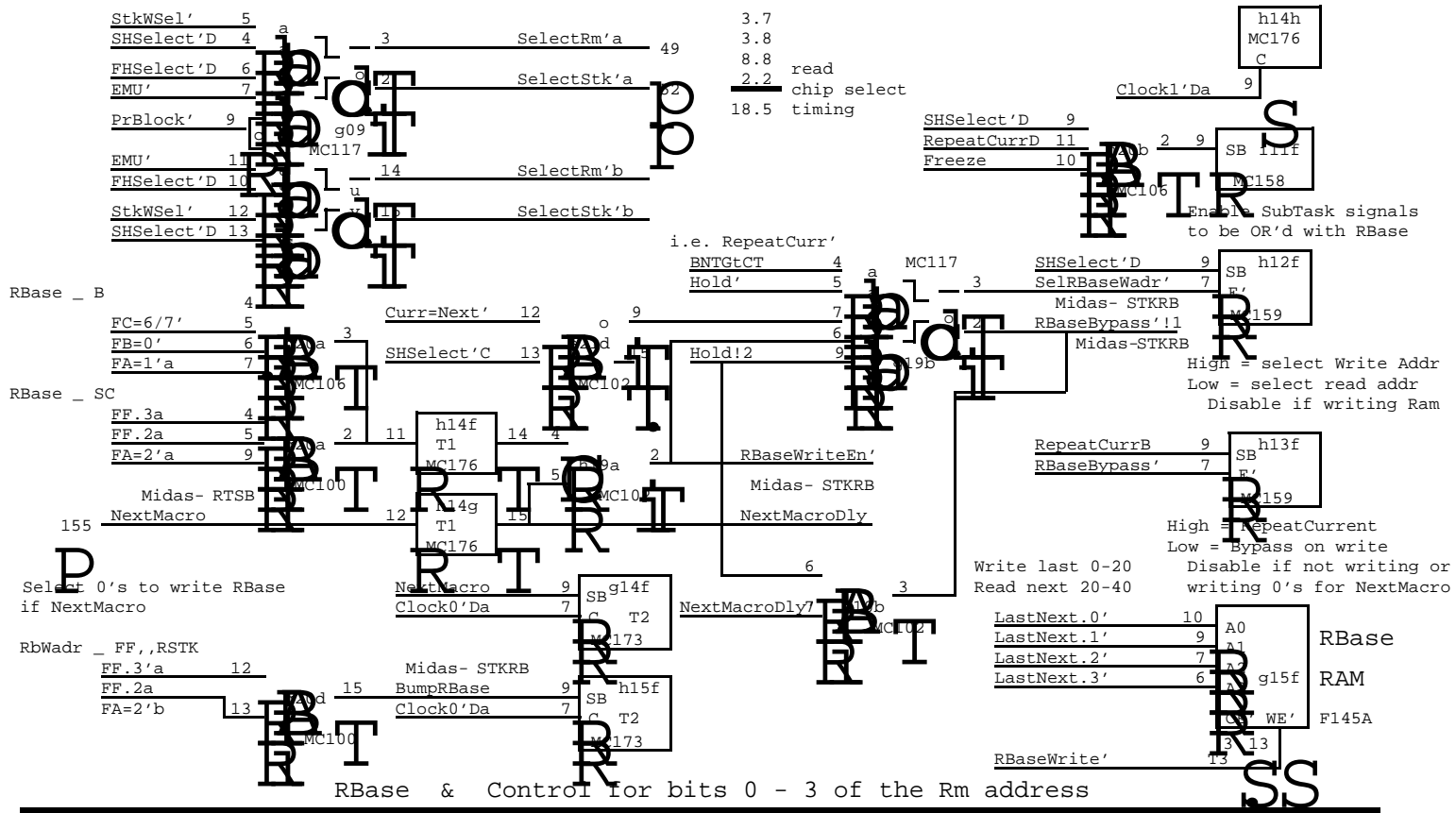


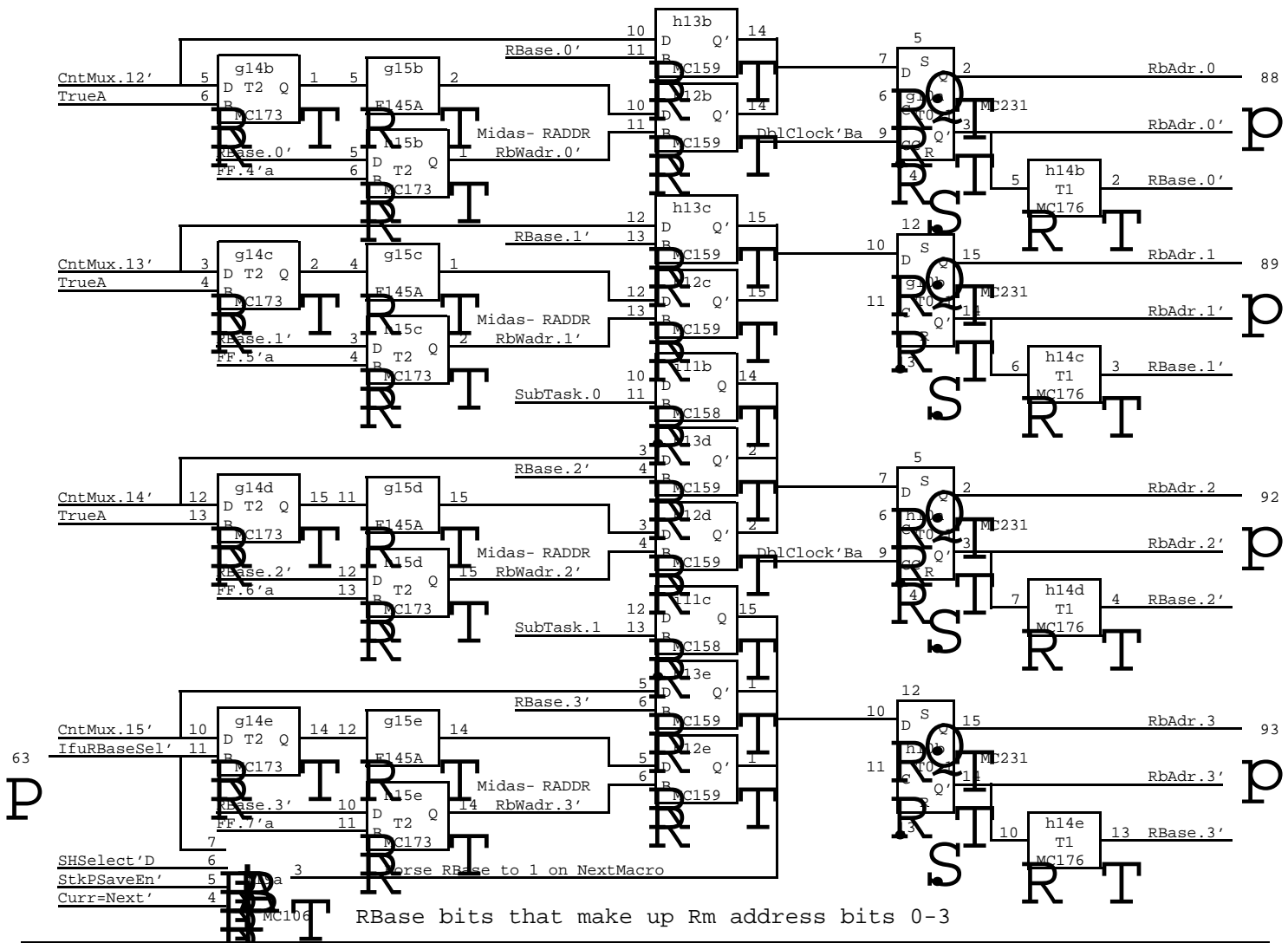


Write RFrom
Pdata or Mdata

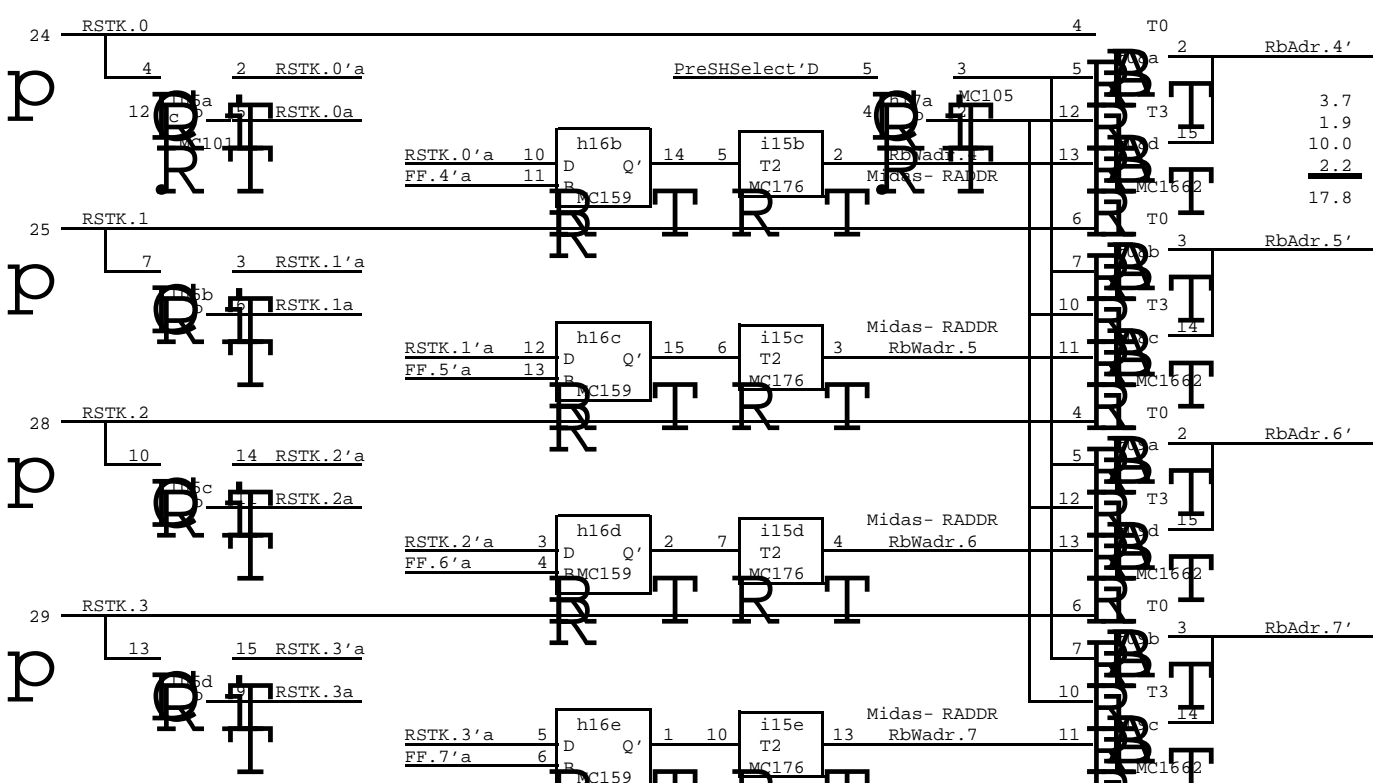
Write TFrom
Pdata or Mdata

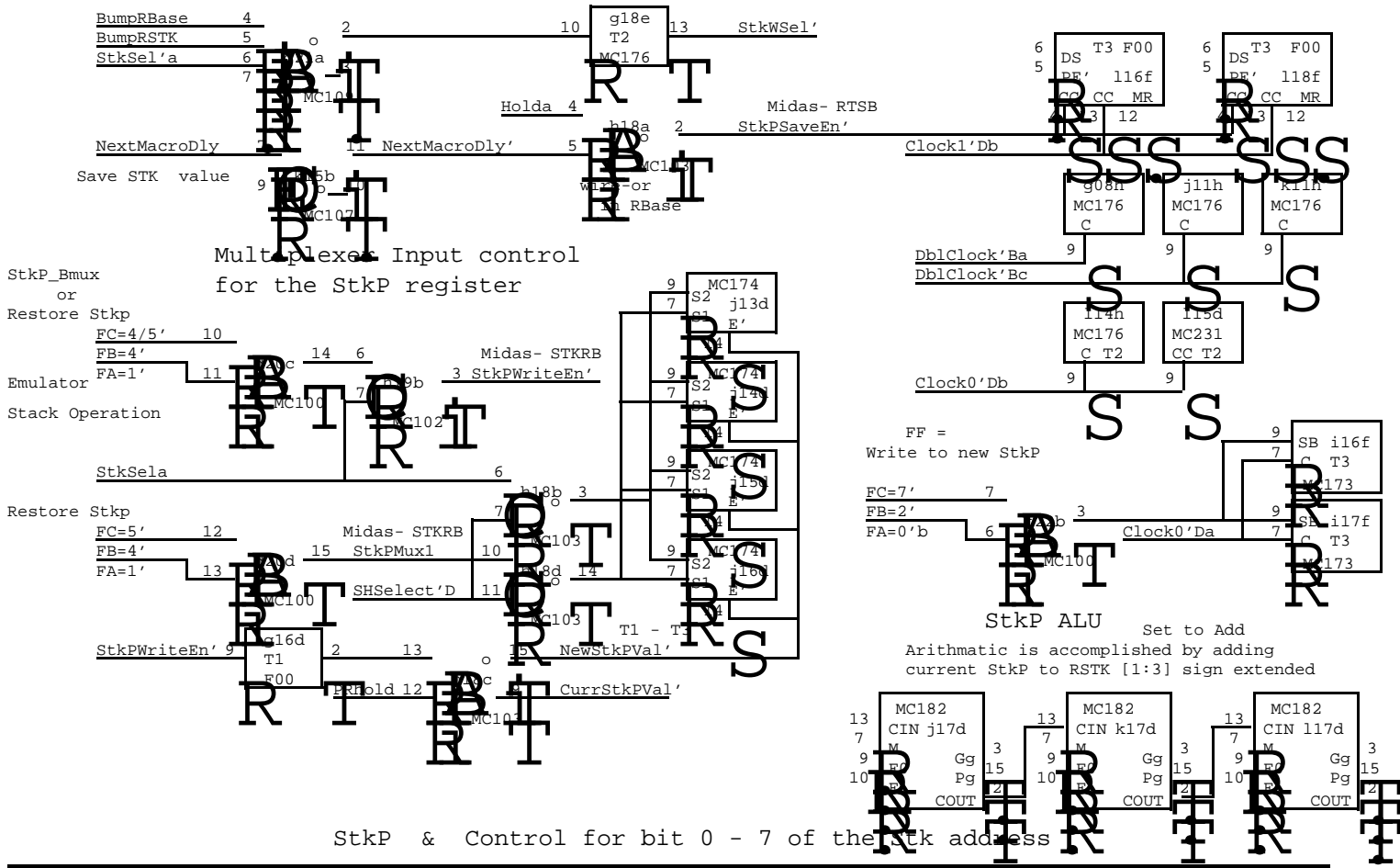






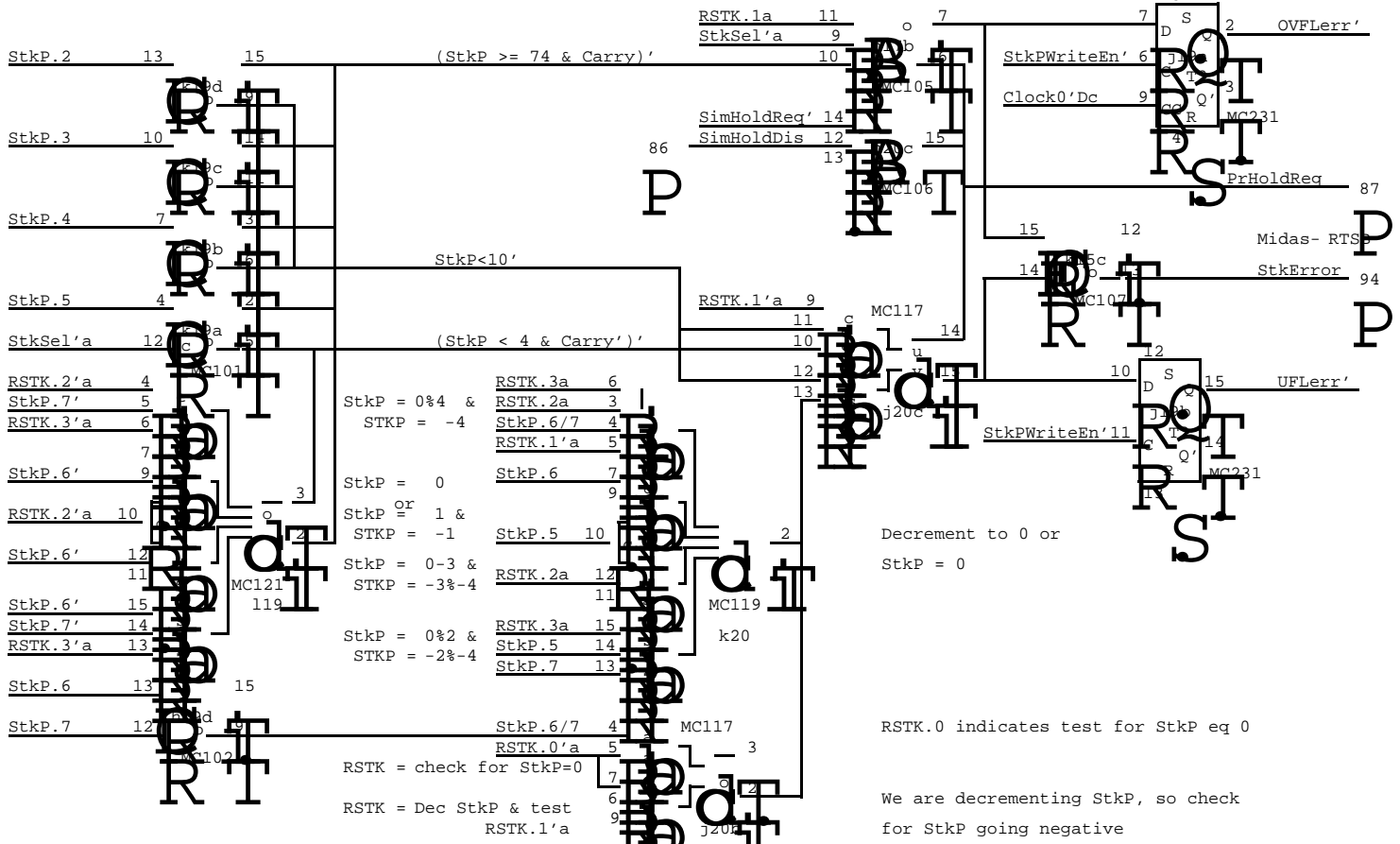
RSTK bits that make up Rm address bits 4-7

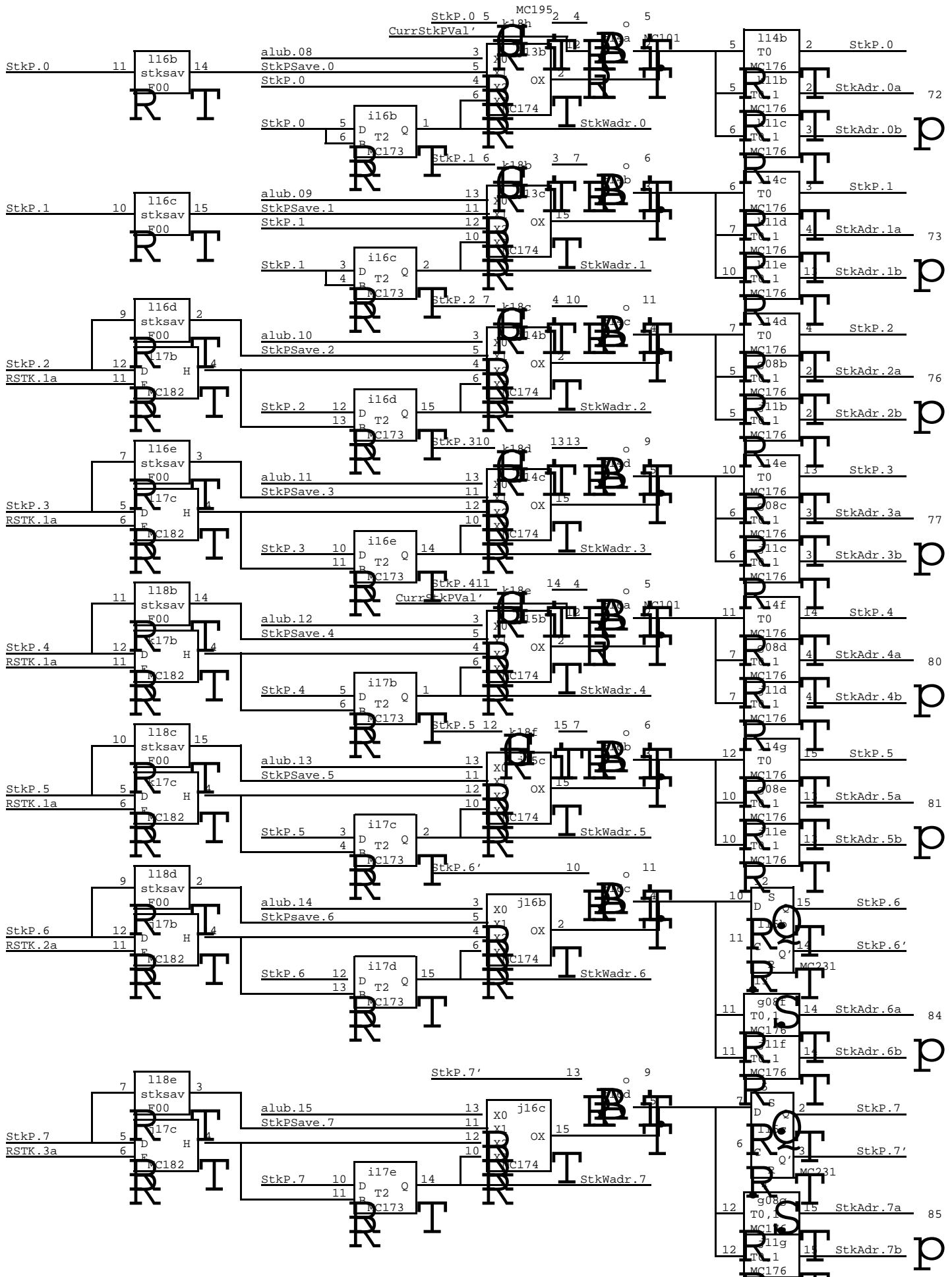


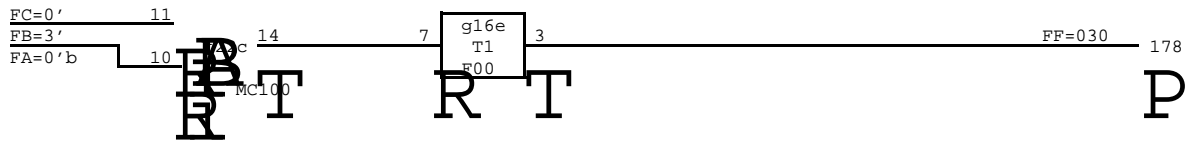


Stk Overflow and Underflow logic

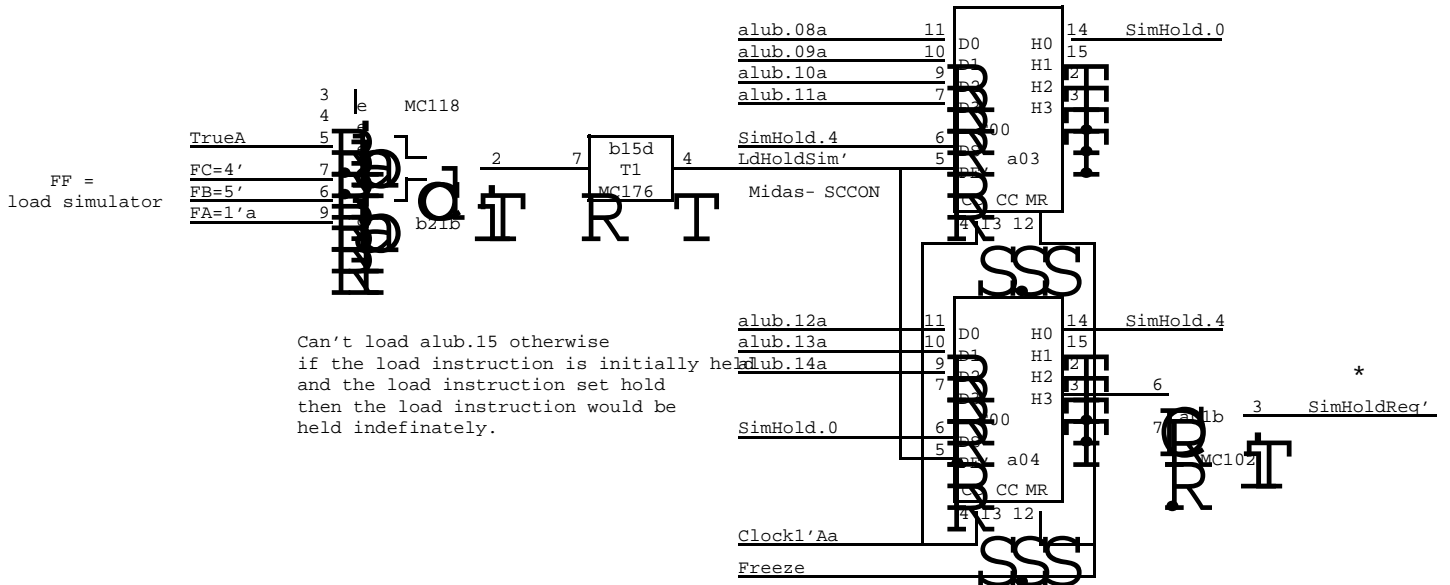
Save the status from the previous stack operation (status if also cleared on an FF=StkP_)





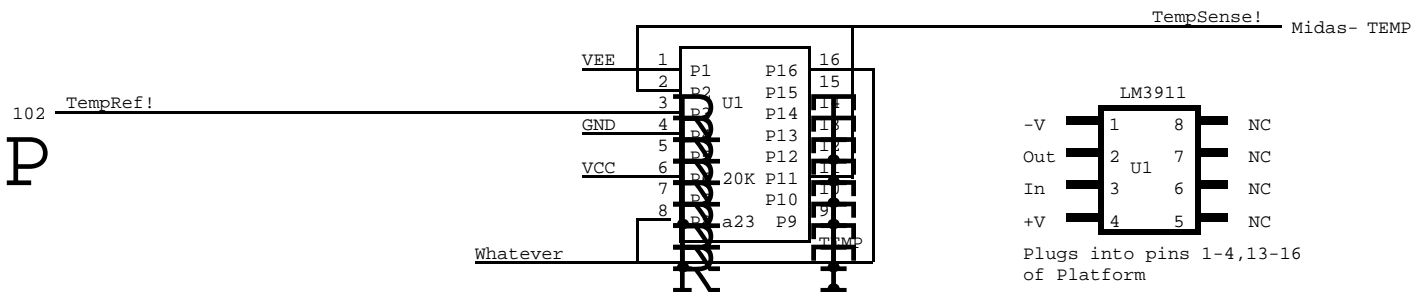


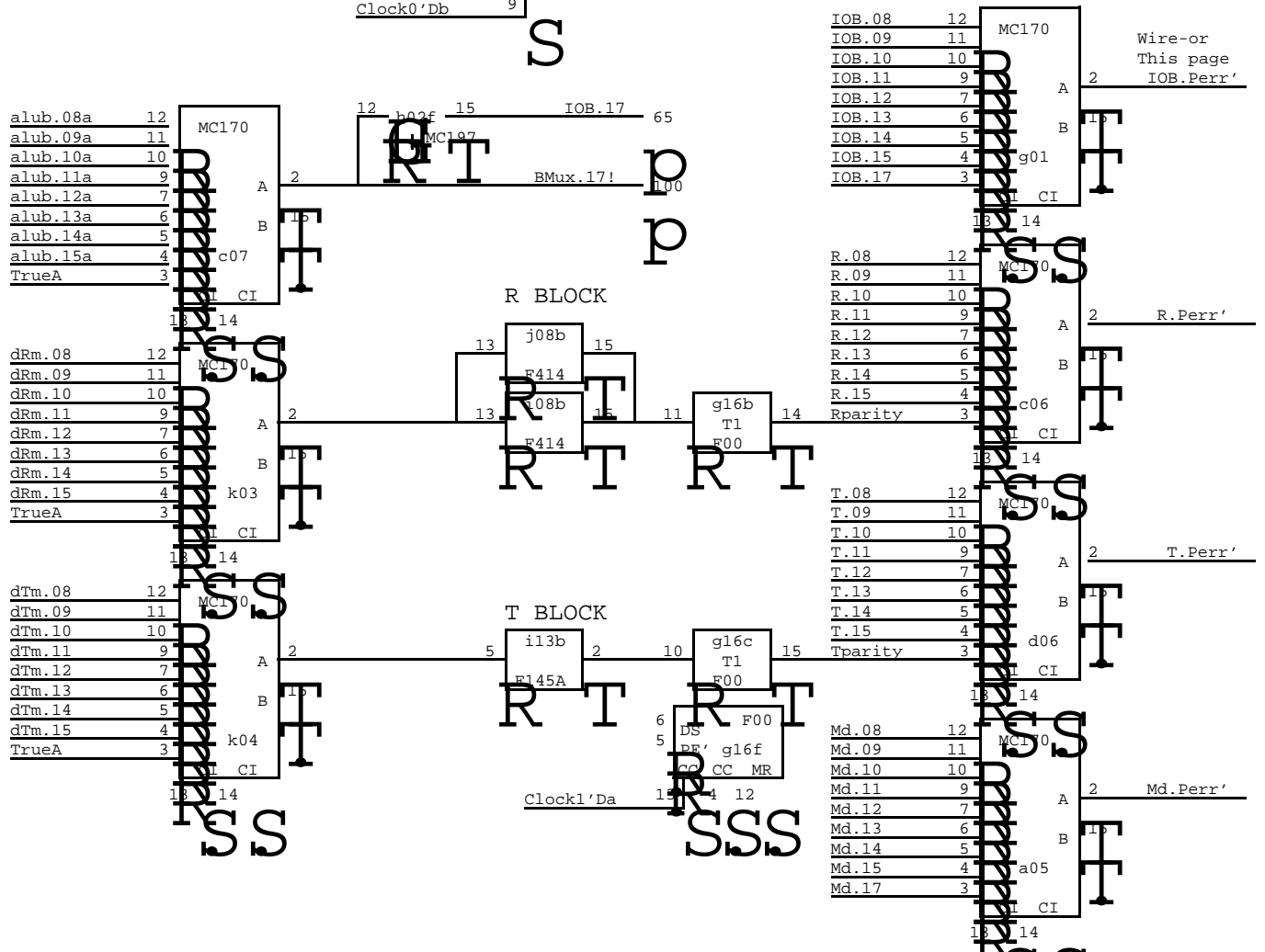
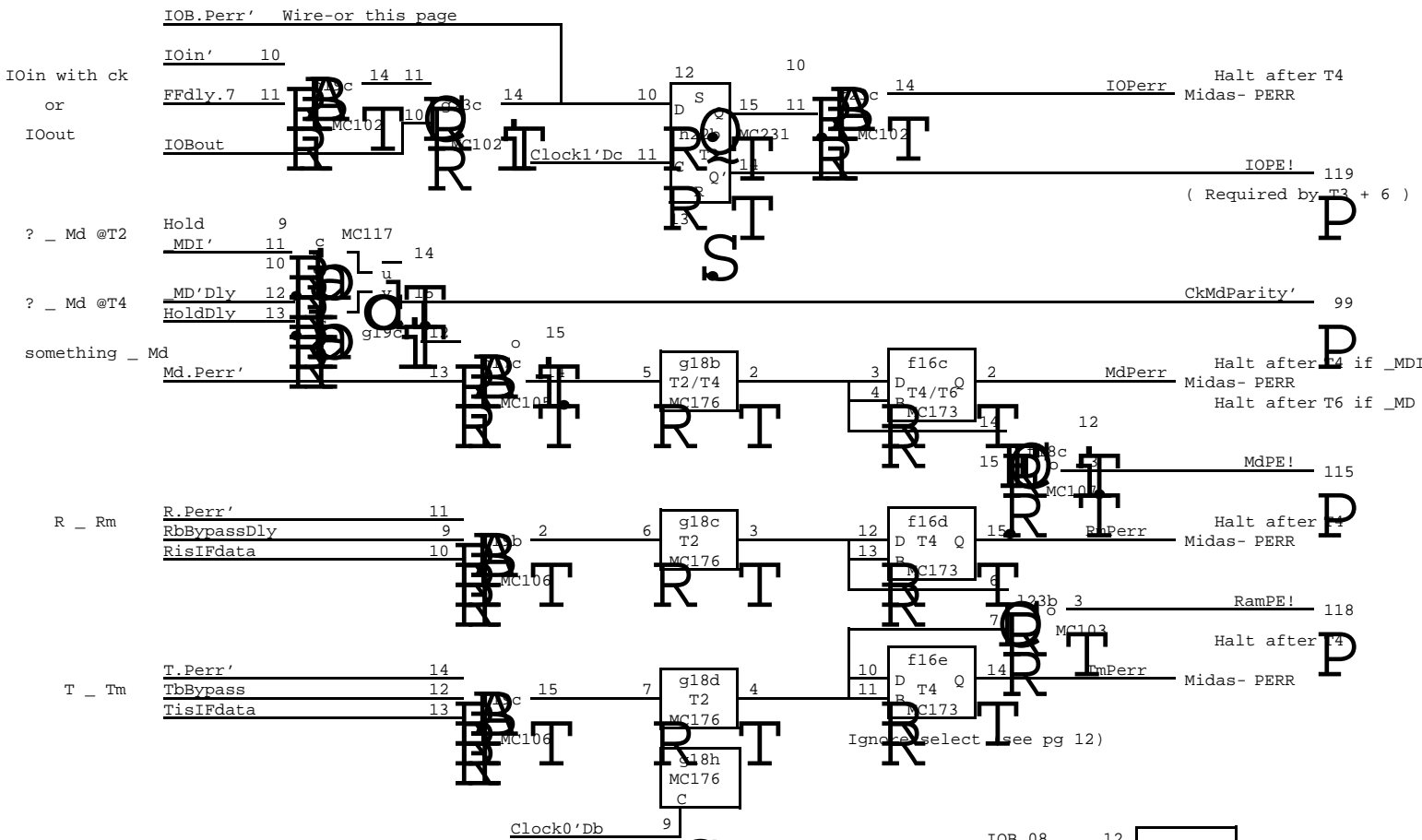
Spare FF Decode



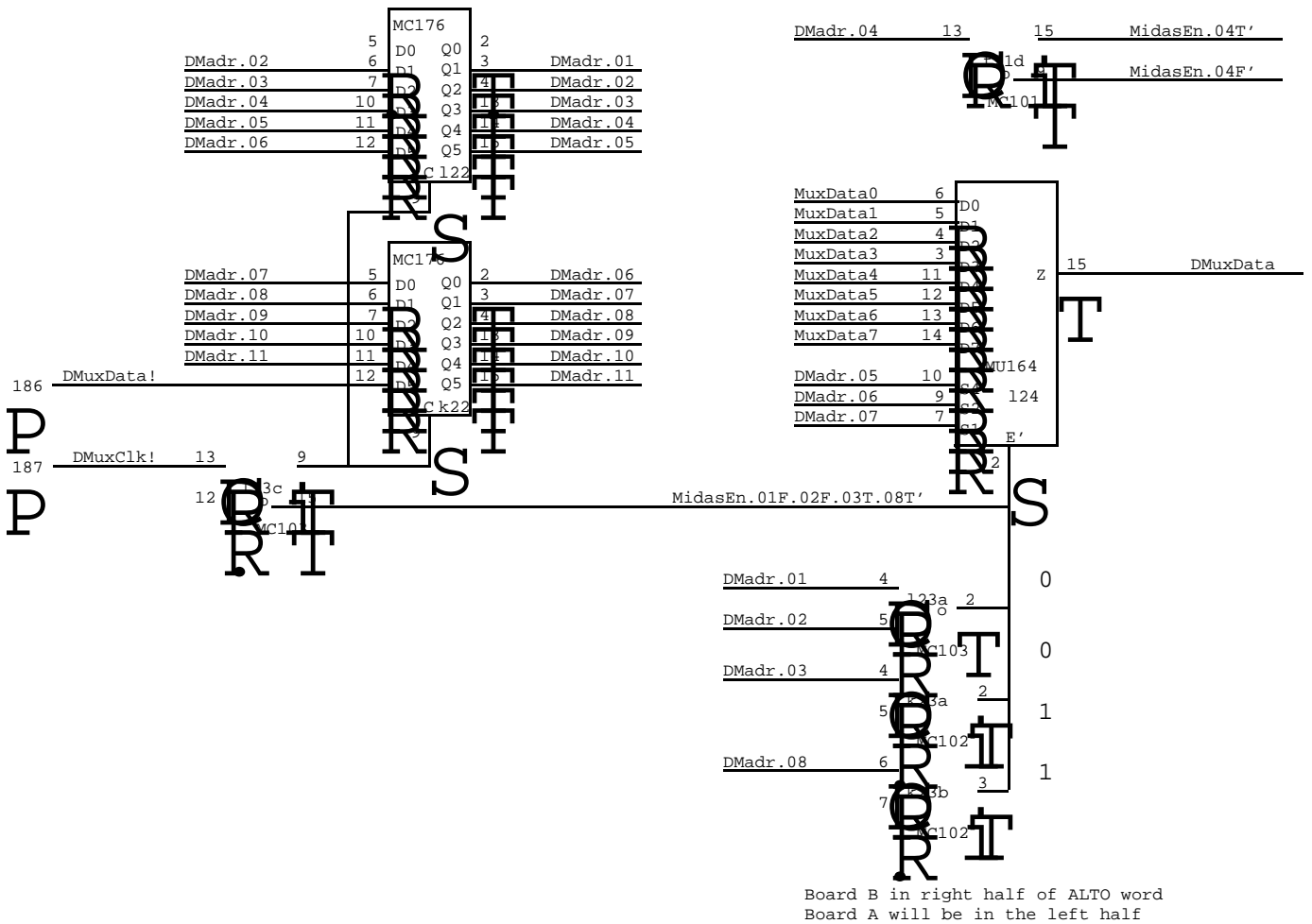
Hold Simulator

Temperature Sensing Ckt





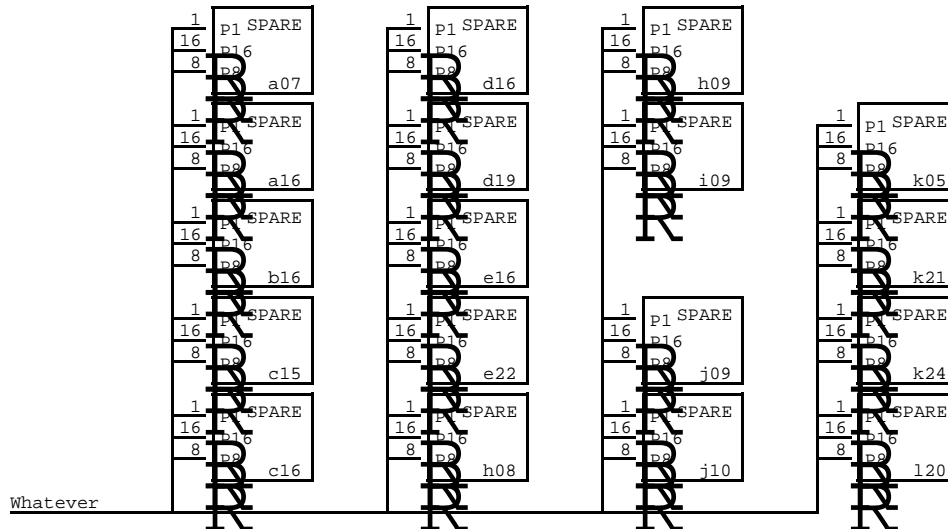
XEROX PARC	Project Dorado	Drawing Parity Logic	File ProcL27.sil	Designer R Bates	Rev Ci	Date 6/27/79	Page 27
---------------	-------------------	-------------------------	---------------------	---------------------	-----------	-----------------	------------



Board B in right half of ALTO word
Board A will be in the left half

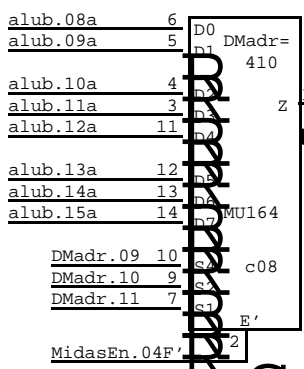
Midas Control

Spair Socket Declarations

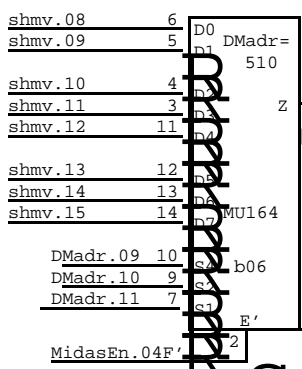


These should be the entire collection of unused locations. This declaration will cause holes to be drilled in these locations on multiwire boards.

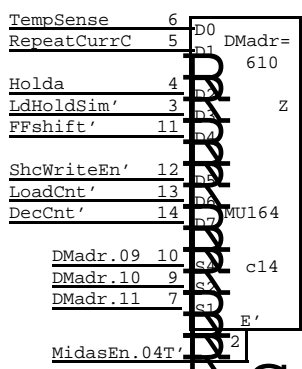
ALUB



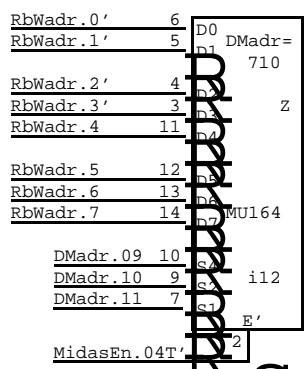
SHMV



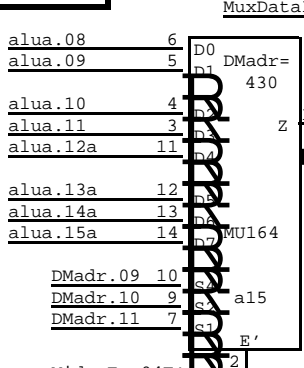
SCCON



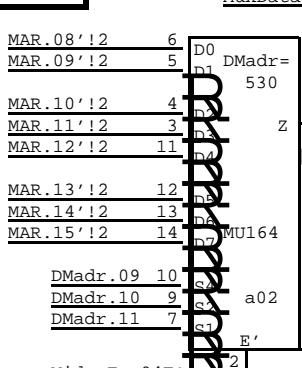
RADDR



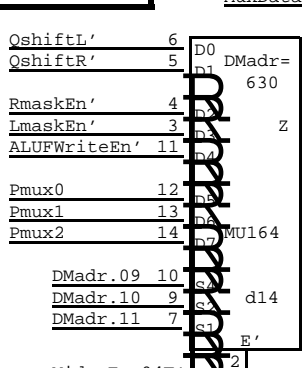
ALUA



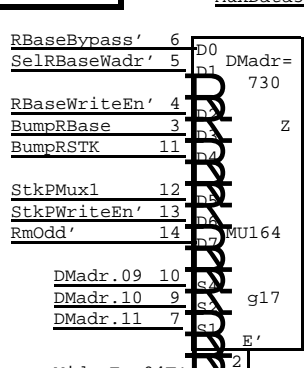
MAR



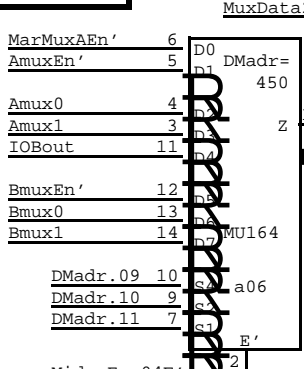
QPDCON



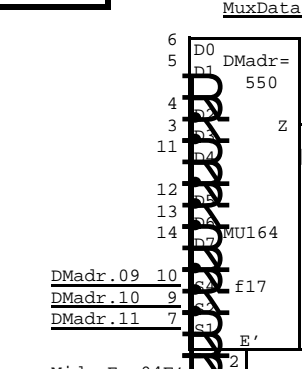
STKRB



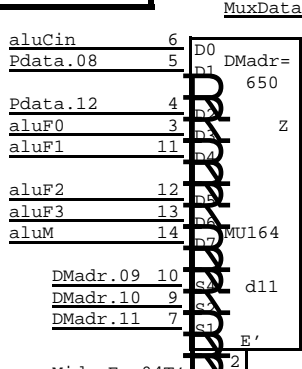
ABCON



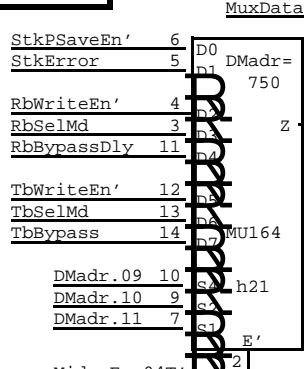
SPAIR



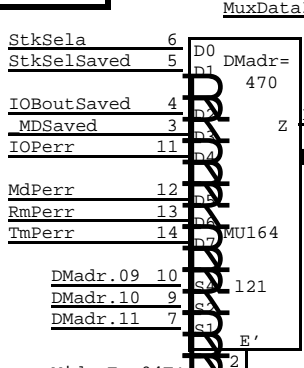
ALUCON



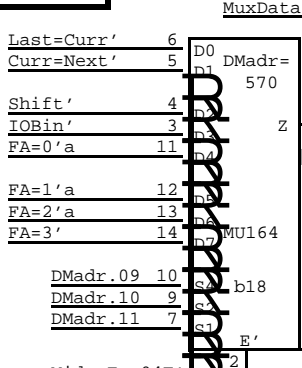
RTSB



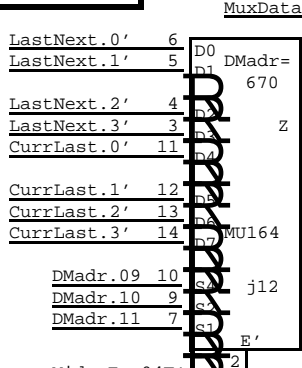
PERR



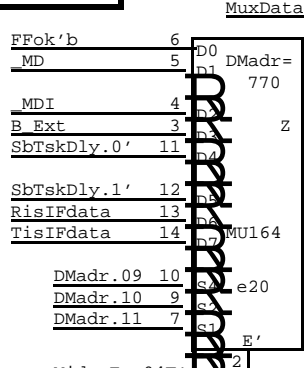
PRFA

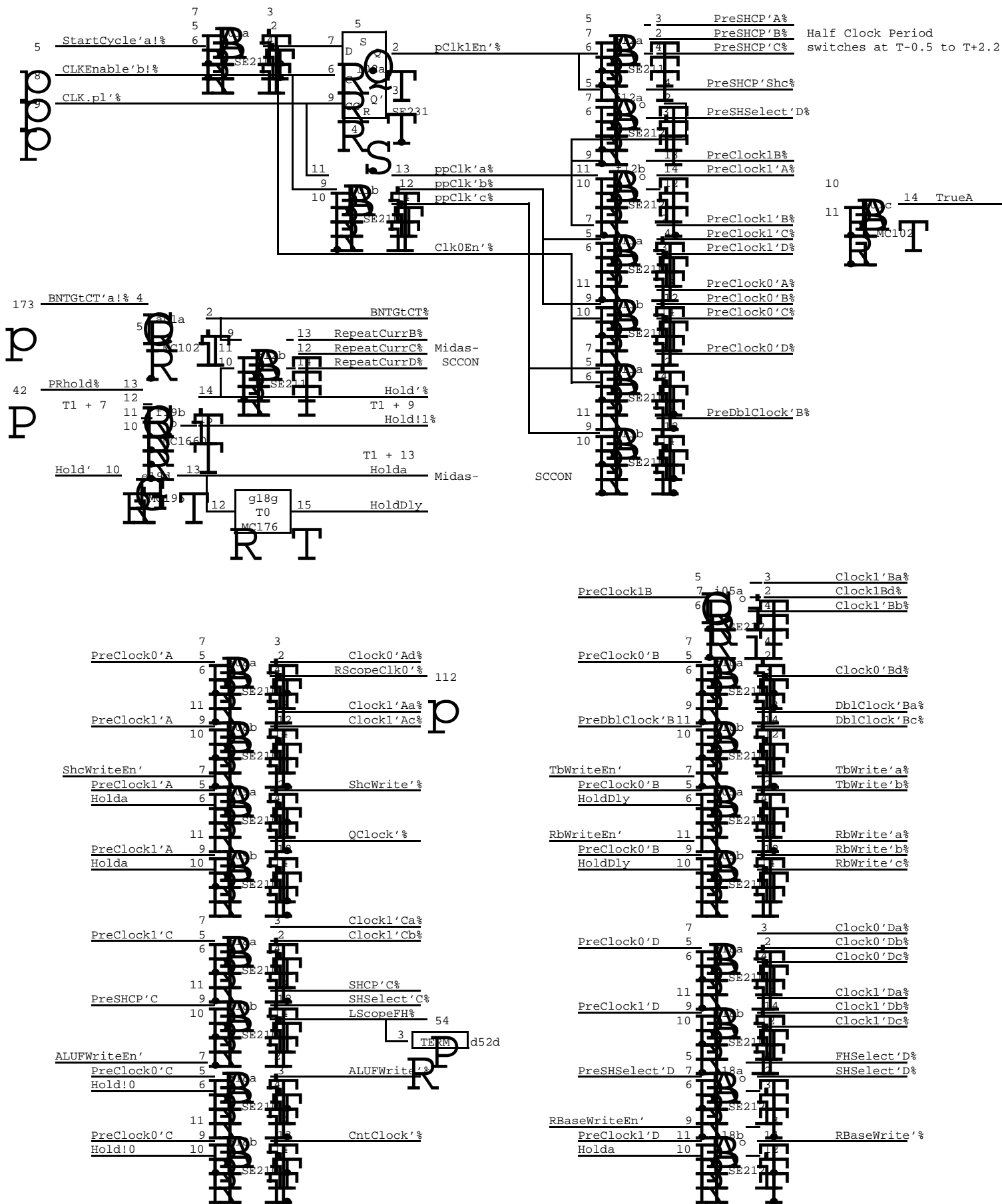


NEXTCL



PJUNK





The Following Signals are provided 50 ohm termination values
by the addition of 100 ohm 1/4 w discrete resistors in parallel with the normal signal

StkSel'a
StkSela
RbAdr.0'
RbAdr.1'
RbAdr.2'
RbAdr.3'
RSTK.0a
RSTK.1a
RSTK.2a
RSTK.3a
RbWadr.0'
RbWadr.1'
RbWadr.2'
RbWadr.3'
RbWadr.4'
RbWadr.5'
RbWadr.6'
RbWadr.7'

For additional information or the rework instructions see
[IVY]<DoradoLogic>ProcL-MWRev-Ch-to-Ci-Rework.sil

XEROX	Project	Reference	File	Designer	Rev	Date	Page
PARC	Dorado	Multiwire Termination Resistors	ProcL31.sil	R. Bates	Ci	3/18/80	31

	184	168	152	136	120	102	84	68	52	36	20	
	A	B	C	D	E	F	G	H	I	J	K	L
1	MC102 III,	Ex Bmux MC174	Ex Bmux MC174	Ex Bmux MC1662	Ex Bmux MC174	Ex Bmux MC174	IOB par MC170	IOB MC197	Md MC175	dRm Reg. MC173	dTm Reg. MC173	Clocks SE210
41	MAR MU164	Mar Mux MC159	Mar Mux MC159	Ex Bmux MC1662	Mar Mux MC159	Mar Mux MC159	Pdata MC164	IOB MC197	Md MC175	dRm Reg. MC173	dTm Reg. MC173	Clocks SE231
3	SimTW F00	Amux MC174	Amux MC174	Amux MC174	Amux MC174	Pdata MC164	Pdata MC164	Pdata MC164	T Reg. MC173	dT Reg. MC173	dRm Par. MC170	T Mem. MC145
42	SimTW F00	Bmux MC174	Bmux MC174	Bmux MC174	Bmux MC174	Pdata MC164	Pdata MC164	Pdata MC164	T Reg. MC173	dT Reg. MC173	dT par. MC170	T Mem. MC145
5	Md Parity MC170	Amux In MC173	Bmux In MC173	Amux In MC173	Bmux In MC173	Pdata MC164	dR Reg. MC173	dR Reg. MC173	Clocks SE212	Clocks SE210		RSTK MC101
43	Mux Cont MU164	Shmv MU164	R Parity MC170	T Parity MC170	Pdata in MC173	Pdata in MC173	R Reg. MC173	Rm Mem. MB071	Stk Mem. MB071	Rm Mem. MB071	Stk Mem. MB071	Rodd MC1668
7		Shmv MC139	alub Par MC170	alua-a MC101	ShA MC158	ShA MC158	R Reg. MC173	X	X	X	X	Rodd MC1668
44	Clocks SE210	Shmv MC139	alub MU164	alu=0 MC109	AhB MC158	AhB MC158	StkPdrT01 MC176		Stk parity F414	Rm parity F414	RbAdr MC1662	Rodd MC211
8	Clocks SE210	alub-a MC1664	alub-a MC1664	alu	alu		R/stk sel MC117				RbAdr MC1662	Rodd MC211
9	II	X	X	II	X	X	X				X	II
45	Amux T1 MC231	Bmux T1 MC231	Mux T1 MC176	MC181		MC181		RbAdr MC231	RbAdr MC231	RBase MC102	Clocks SE210	RbBypass MC1660
10	II	II	IIIIII	X	X		X	X	X		II	II
11	Amux T1 MC231	Mux T1 MC141	Q Reg. MC141	ALUFM MU164	alua sh MC159	alua sh MC159	CntMux MC159	CntMux MC159	RBase MC158	StkAdr T01 MC176	StkAdr T01 MC176	Misc. MC231
46	II	IIII	X	X	X	X	X	X	X	IIIIII	IIII	II
12	Shc MC173	Shc MC173	Q Reg. MC141	Shc MC180	ALUFM MC176	Clocks SE212	Clocks SE211	RBase MC159	RmWadr MU164	NextLast MU164	RmBypass MC113	StkBypass MC113
13	X	X	X	X	IIIIII	II	II	X	X	X	X	X
13	Shc MC173	Shc MC173	Shc MC158	Shc MC180	ALUFM MC145	Clocks SE210	Clocks SE210	RBase MC159	T mem-P MC145	StkAdr MC174	RmBypass MC113	StkBypass MC113
47	X	X	X	X	X	II	II	X	X	X	X	X
14	Bmux In MC158	Bmux In MC158	Shc MU164	P mux MU164	ALUFM MC145	Pdata in MC159	RBase MC173	RbAdr T1 MC176	CurrLast' MC141	StkAdr MC174	StkAdr MC158	StkP MC176
15	X	X	X	X	X	X	X	IIIIII	X	X	X	X
15	alua MU164	Q Reg. MC176		Cnt Reg. F16	Cnt Reg. F16	Pdata in MC159	RBase MC145	RBase MC173	RBase T0 MC176	StkAdr MC174	Misc. MC107	StkP MC231
48	X	IIIIII		X	X	X	X	X	IIIIII	X	III	X
16						aluCin MC173	Parity T1 F00	RBase MC159	StkAdr MC173	StkAdr MC174	StkAdr MC158	StkPsave F00
17	Q Reg. MC113	Q Reg. MC119	Q Reg. MC119	Misc. T0 MC176	Misc. T1 MC176	SPAIR MU164	Rm cont MU164	Misc. MC105	StkAdr MC173	StkP MC182	StkP MC182	StkP MC182
49	X	X	X	IIIIII	I,III,	X	X	III	X	X	X	X
18	LastNext' MC158	FA MU164	Clocks SE210	Clocks SE210	P mux MC102	Misc. MC107	Misc. T0 MC176	RBase MC103	Clocks SE212	Clocks SE210	StkP MC195	StkPsave F00
19	X	X	II	II	IIII	III	IIIIII	IIII	II	II	X	X
19	LastNext' MC158	Misc. MC212	Amux MC121		Misc. MC195	MC1660	Parity MC117	Misc. MC102	Parity MC106	Ovfl-Ufl MC231	Ovfl-Ufl MC101	Ovfl-Ufl MC121
50	X	II	X		III,,	II	II	IIII	III	II	IIII	X
20	Last' MC141	FA=0 MC100	FA=2 MC100	FA=1 MC100	Misc. MU164	FA=0 MC100	FA=2 MC100	FA=0 MC106	ShI MC164	Ovfl-Ufl MC117	Ovfl-Ufl MC119	
21	X	IIII	IIII	IIII	X	IIII	IIII	IIII	X	X	X	Parity MU164
51	Curr' MC141	FA=1 MC118	Amux MC119	Amux MC117	Misc. MC102	ALUF MC101	B_Ext MC109	Rm cont MU164	ShI MC164	ShI MC164		X
51	X	II	X	X	IIII	IIII	II	X	X	X		X
22	Next=Curr MC113	P mux MC212	Misc. MC102	_MDI MC121		FA=0 MC100	Misc. MC103	Parity MC231	ShI MC164	ShI MC164	Midas MC176	Midas MC176
52	X	II	IIII	X		III,	IIII	II	X	X	X	X
23	Temp LM3911	ASel MC101	FF dec. MC210	FF dec. MC101	FF-a MC101	FF-a MC101	Misc. MC102	LC dec. MC102	ShI MC164	ShI MC164	Misc. MC102	Misc. MC103
52	X	IIII	II	X	X	X	IIII	IIII	X	X	IIII	IIII
24	Next MC101	ALUF MC211	Misc. MC103	FF dec. MC101	FF dec. MC161	FF dec. MC161	MemBase MC231	Bsel dec. MC101	ShI MC164	ShI MC164		Midas MU164
52	X	II	IIII	X	X	X	II	IIII	X	X	X	X

Spare = 19

XEROX PARC	Project Dorado	Reference Board Layout	File Procl32.sil	Designer R. Bates	Rev Ci	Date 3/18/80	Page 32
---------------	-------------------	---------------------------	---------------------	----------------------	-----------	-----------------	------------

Use Dorado Proms to define the following Proms:

Board Name	Prom Name	location
PorcL	Lmask (low bite)	b07
	Rmask (low bite)	b08