Figure 2-3 82C557 SYSC Block Diagram

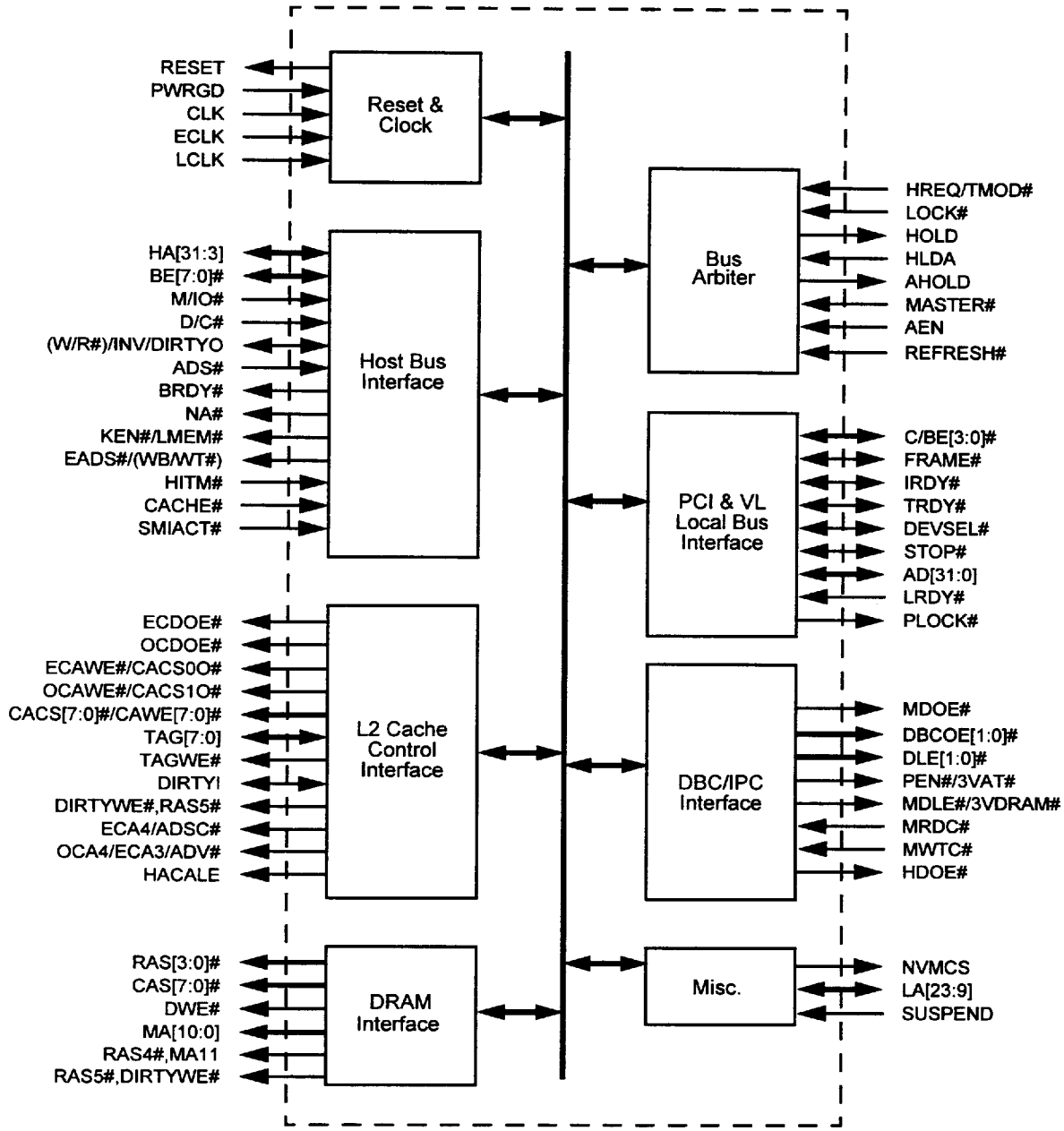
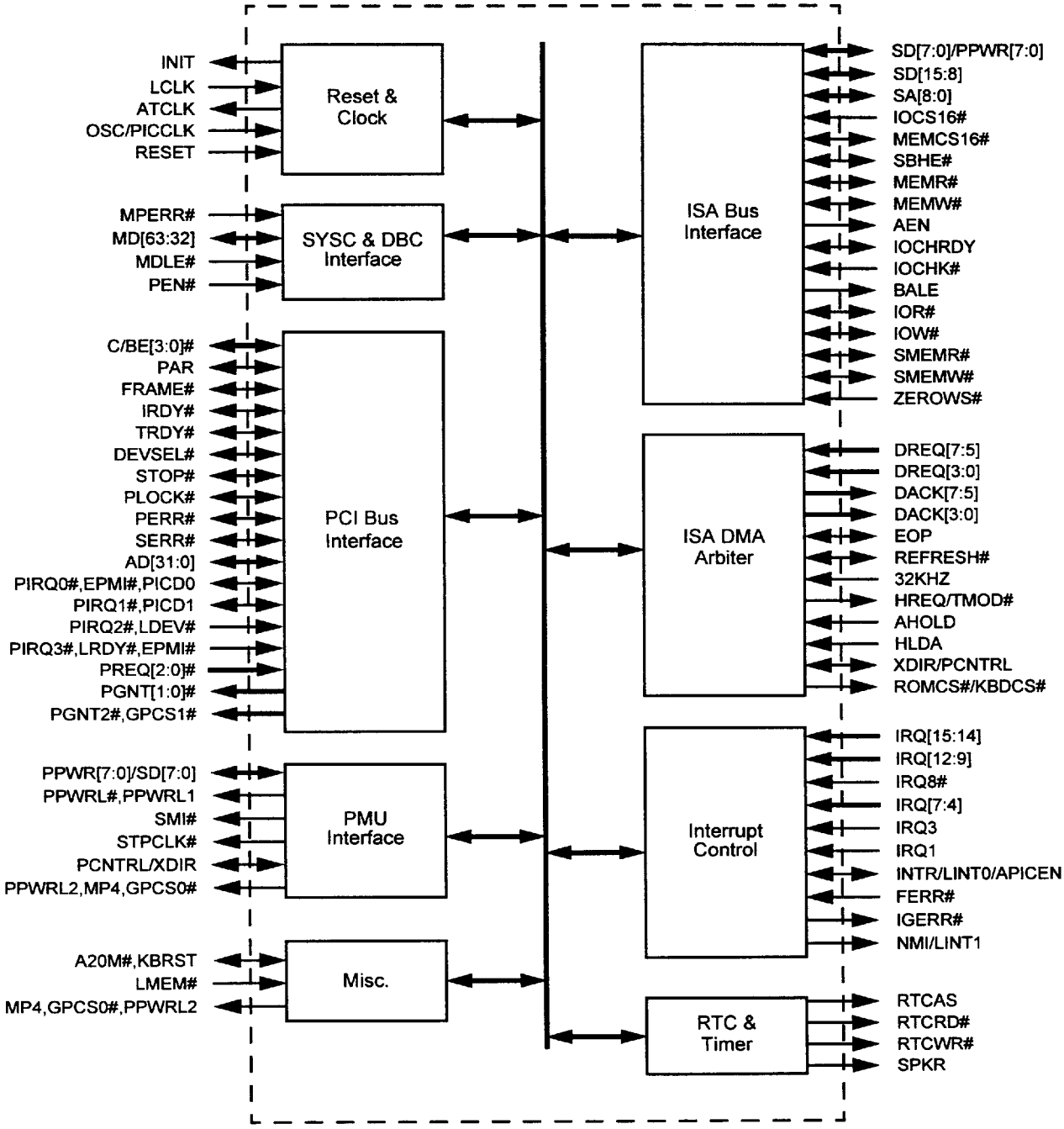


Figure 2-4 82C558 IPC Block Diagram



3.0 Signal Definitions

Figure 3-1 82C556 Pin Diagram

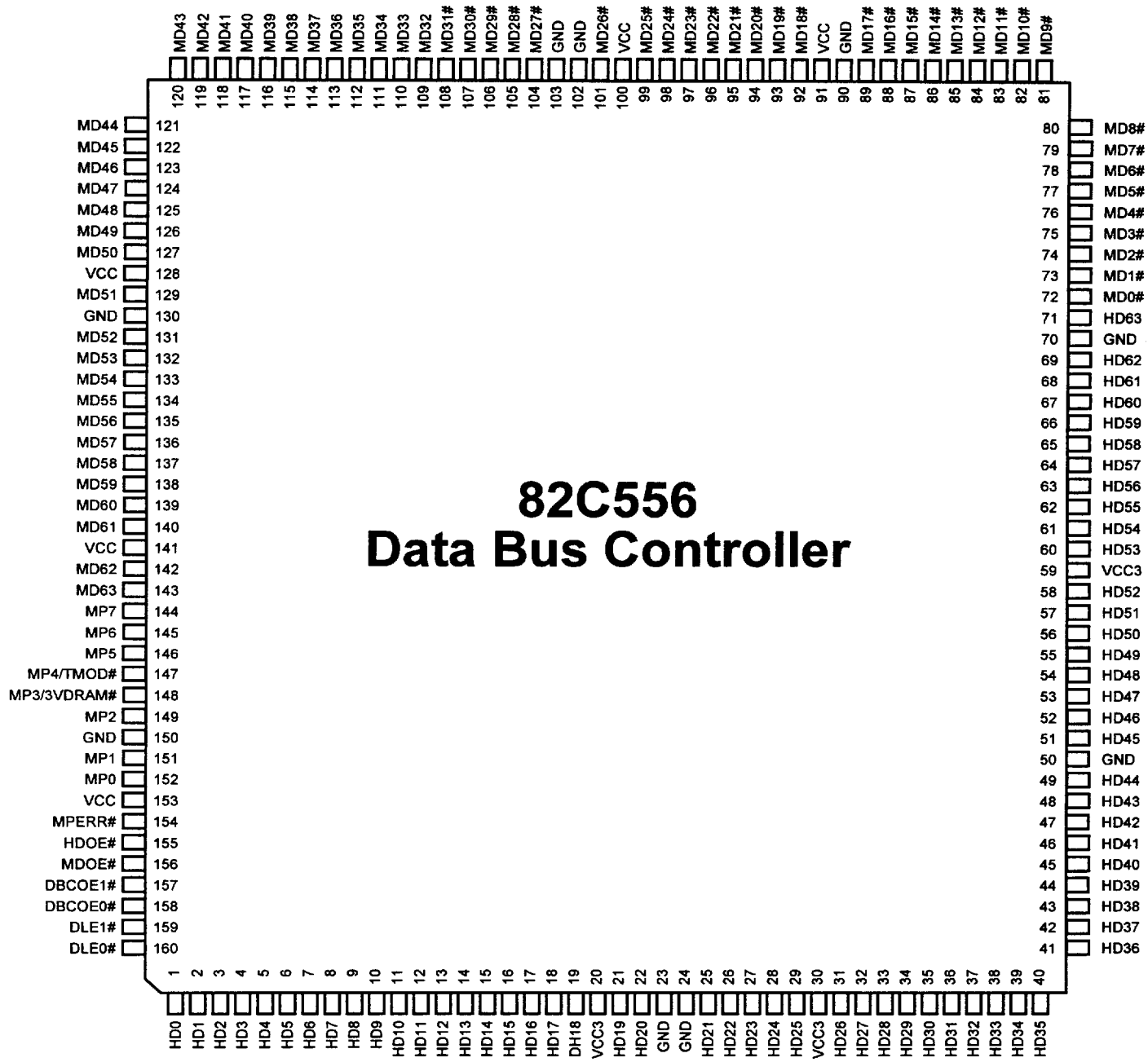


Table 3-1 82C556 160-Pin Numerical Pin Cross-Reference List

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	HD0	41	HD36	81	MD9#	121	MD44
2	HD1	42	HD37	82	MD10#	122	MD45
3	HD2	43	HD38	83	MD11#	123	MD46
4	HD3	44	HD39	84	MD12#	124	MD47
5	HD4	45	HD40	85	MD13#	125	MD48
6	HD5	46	HD41	86	MD14#	126	MD49
7	HD6	47	HD42	87	MD15#	127	MD50
8	HD7	48	HD43	88	MD16#	128	VCC
9	HD8	49	HD44	89	MD17#	129	MD51
10	HD9	50	GND	90	GND	130	GND
11	HD10	51	HD45	91	VCC	131	MD52
12	HD11	52	HD46	92	MD18#	132	MD53
13	HD12	53	HD47	93	MD19#	133	MD54
14	HD13	54	HD48	94	MD20#	134	MD55
15	HD14	55	HD49	95	MD21#	135	MD56
16	HD15	56	HD50	96	MD22#	136	MD57
17	HD16	57	HD51	97	MD23#	137	MD58
18	HD17	58	HD52	98	MD24#	138	MD59
19	HD18	59	VCC3	99	MD25#	139	MD60
20	VCC3	60	HD53	100	VCC	140	MD61
21	HD19	61	HD54	101	MD26#	141	VCC
22	HD20	62	HD55	102	GND	142	MD62
23	GND	63	HD56	103	GND	143	MD63
24	GND	64	HD57	104	MD27#	144	MP7
25	HD21	65	HD58	105	MD28#	145	MP6
26	HD22	66	HD59	106	MD29#	146	MP5
27	HD23	67	HD60	107	MD30#	147	MP4/TMOD#
28	HD24	68	HD61	108	MD31#	148	MP3/3VDRAM#
29	HD25	69	HD62	109	MD32	149	MP2
30	VCC3	70	GND	110	MD33	150	GND
31	HD26	71	HD63	111	MD34	151	MP1
32	HD27	72	MD0#	112	MD35	152	MP0
33	HD28	73	MD1#	113	MD36	153	VCC
34	HD29	74	MD2#	114	MD37	154	MPERR#
35	HD30	75	MD3#	115	MD38	155	HDOE#
36	HD31	76	MD4#	116	MD39	156	MDOE#
37	HD32	77	MD5#	117	MD40	157	DBCOE1#
38	HD33	78	MD6#	118	MD41	158	DBCOE0#
39	HD34	79	MD7#	119	MD42	159	DLE1#
40	HD35	80	MD8#	120	MD43	160	DLE0#

Table 3-2 82C556 160-Pin Alphabetical Cross-Reference List

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
158	DBCOE0#	32	HD27	74	MD2#	119	MD42
157	DBCOE1#	33	HD28	75	MD3#	120	MD43
160	DLE0#	34	HD29	76	MD4#	121	MD44
159	DLE1#	35	HD30	77	MD5#	122	MD45
23	GND	36	HD31	78	MD6#	123	MD46
24	GND	37	HD32	79	MD7#	124	MD47
50	GND	38	HD33	80	MD8#	125	MD48
70	GND	39	HD34	81	MD9#	126	MD49
90	GND	40	HD35	82	MD10#	127	MD50
102	GND	41	HD36	83	MD11#	129	MD51
103	GND	42	HD37	84	MD12#	131	MD52
130	GND	43	HD38	85	MD13#	132	MD53
150	GND	44	HD39	86	MD14#	133	MD54
1	HD0	45	HD40	87	MD15#	134	MD55
2	HD1	46	HD41	88	MD16#	135	MD56
3	HD2	47	HD42	89	MD17#	136	MD57
4	HD3	48	HD43	92	MD18#	137	MD58
5	HD4	49	HD44	93	MD19#	138	MD59
6	HD5	51	HD45	94	MD20#	139	MD60
7	HD6	52	HD46	95	MD21#	140	MD61
8	HD7	53	HD47	96	MD22#	142	MD62
9	HD8	54	HD48	97	MD23#	143	MD63
10	HD9	55	HD49	98	MD24#	156	MDOE#
11	HD10	56	HD50	99	MD25#	152	MP0
12	HD11	57	HD51	101	MD26#	151	MP1
13	HD12	58	HD52	104	MD27#	149	MP2
14	HD13	60	HD53	105	MD28#	148	MP3/3VDRAM#
15	HD14	61	HD54	106	MD29#	147	MP4/TMOD#
16	HD15	62	HD55	107	MD30#	146	MP5
17	HD16	63	HD56	108	MD31#	145	MP6
18	HD17	64	HD57	109	MD32	144	MP7
19	HD18	65	HD58	110	MD33	154	MPERR#
21	HD19	66	HD59	111	MD34	153	VCC
22	HD20	67	HD60	112	MD35	91	VCC
25	HD21	68	HD61	113	MD36	100	VCC
26	HD22	69	HD62	114	MD37	128	VCC
27	HD23	71	HD63	115	MD38	141	VCC
28	HD24	155	HDOE#	116	MD39	20	VCC3
29	HD25	72	MD0#	117	MD40	30	VCC3
31	HD26	73	MD1#	118	MD41	59	VCC3

Figure 3-2 82C557 SYSC Pin Diagram

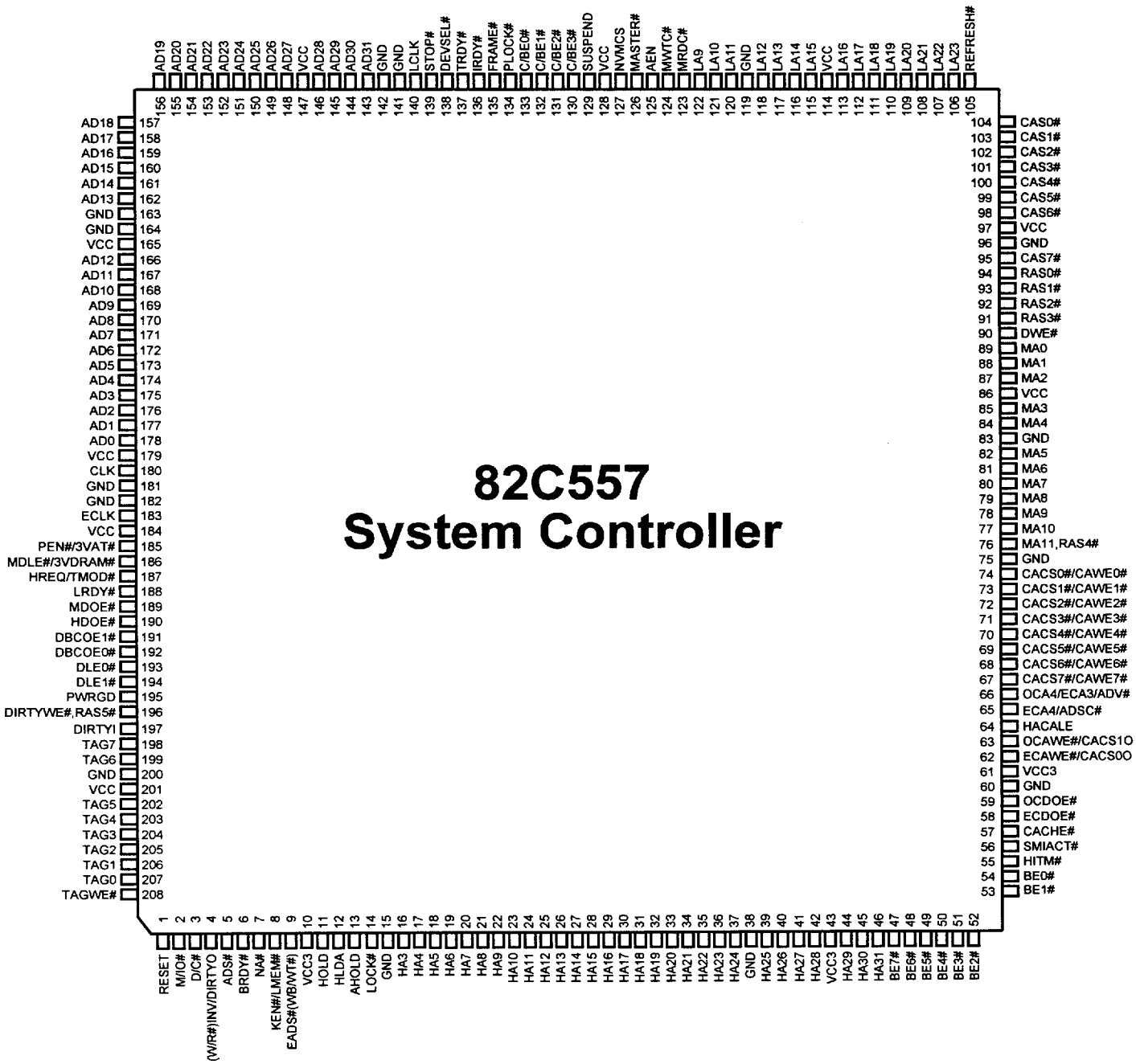


Table 3-3 82C557 Numerical Pin Cross-Reference List

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	RESET	53	BE1#	105	REFRESH#	157	AD18
2	M/IO#	54	BE0#	106	LA23	158	AD17
3	D/C#	55	HITM#	107	LA22	159	AD16
4	(W/R#)/INV/DIRTYO	56	SMIACT#	108	LA21	160	AD15
5	ADS#	57	CACHE#	109	LA20	161	AD14
6	BRDY#	58	ECDOE#	110	LA19	162	AD13
7	NA#	59	OCDOE#	111	LA18	163	GND
8	KEN#/LMEM#	60	GND	112	LA17	164	GND
9	EADS#/(WB/WT#)	61	VCC3	113	LA16	165	VCC
10	VCC3	62	ECAWE#/CACS0O#	114	VCC	166	AD12
11	HOLD	63	OCAWE#/CACS1O#	115	LA15	167	AD11
12	HLDA	64	HACALE	116	LA14	168	AD10
13	AHOLD	65	ECA4/ADSC#	117	LA13	169	AD9
14	LOCK#	66	OCA4/ECA3/ADV#	118	LA12	170	AD8
15	GND	67	CACS7#/CAWE7#	119	GND	171	AD7
16	HA3	68	CACS6#/CAWE6#	120	LA11	172	AD6
17	HA4	69	CACS5#CAWE5#	121	LA10	173	AD5
18	HA5	70	CACS4#/CAWE4#	122	LA9	174	AD4
19	HA6	71	CACS3#/CAWE3#	123	MRDC#	175	AD3
20	HA7	72	CACS2#/CAWE2#	124	MWTC#	176	AD2
21	HA8	73	CACS1#/CAWE1#	125	AEN	177	AD1
22	HA9	74	CACS0#/CAWE0#	126	MASTER#	178	AD0
23	HA10	75	GND	127	NVMCS	179	VCC
24	HA11	76	MA11,RAS4#	128	VCC	180	CLK
25	HA12	77	MA10	129	SUSPEND	181	GND
26	HA13	78	MA9	130	C/BE3#	182	GND
27	HA14	79	MA8	131	C/BE2#	183	ECLK
28	HA15	80	MA7	132	C/BE1#	184	VCC
29	HA16	81	MA6	133	C/BE0#	185	PEN#/3VAT#
30	HA17	82	MA5	134	PLOCK#	186	MDLE#/3VDRAM#
31	HA18	83	GND	135	FRAME#	187	HREQ/TMOD#
32	HA19	84	MA4	136	IRDY#	188	LRDY#
33	HA20	85	MA3	137	TRDY#	189	MDOE#
34	HA21	86	VCC	138	DEVSEL#	190	HDOE#
35	HA22	87	MA2	139	STOP#	191	DBCOE1#
36	HA23	88	MA1	140	LCLK	192	DBCOE0#
37	HA24	89	MA0	141	GND	193	DLE0#
38	GND	90	DWE#	142	GND	194	DLE1#
39	HA25	91	RAS3#	143	AD31	195	PWRGD
40	HA26	92	RAS2#	144	AD30	196	DIRTYWE#,RAS5#
41	HA27	93	RAS1#	145	AD29	197	DIRTYI
42	HA28	94	RAS0#	146	AD28	198	TAG7
43	VCC3	95	CAS7#	147	VCC	199	TAG6
44	HA29	96	GND	148	AD27	200	GND
45	HA30	97	VCC	149	AD26	201	VCC
46	HA31	98	CAS6#	150	AD25	202	TAG5
47	BE7#	99	CAS5#	151	AD24	203	TAG4
48	BE6#	100	CAS4#	152	AD23	204	TAG3
49	BE5#	101	CAS3#	153	AD22	205	TAG2
50	BE4#	102	CAS2#	154	AD21	206	TAG1
51	BE3#	103	CAS1#	155	AD20	207	TAG0
52	BE2#	104	CAS0#	156	AD19	208	TAGWE#

Table 3-4 82C557 Alphabetical Pin Cross-Reference List

Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.
AD0	178	CACS3#/CAWE3#	71	HA12	25	MA5	82
AD1	177	CACS4#/CAWE4#	70	HA13	26	MA6	81
AD2	176	CACS5#/CAWE5#	69	HA14	27	MA7	80
AD3	175	CACS6#/CAWE6#	68	HA15	28	MA8	79
AD4	174	CACS7#/CAWE7#	67	HA16	29	MA9	78
AD5	173	CAS0#	104	HA17	30	MA10	77
AD6	172	CAS1#	103	HA18	31	MA11,RAS4#	76
AD7	171	CAS2#	102	HA19	32	MASTER#	126
AD8	170	CAS3#	101	HA20	33	MDLE#/3VDRAM#	186
AD9	169	CAS4#	100	HA21	34	MDOE#	189
AD10	168	CAS5#	99	HA22	35	MRDC#	123
AD11	167	CAS6#	98	HA23	36	MWTC#	124
AD12	166	CAS7#	95	HA24	37	NA#	7
AD13	162	CLK	180	HA25	39	NVMCS	127
AD14	161	D/C#	3	HA26	40	OCA4/ECA3/ADV#	66
AD15	160	DBCOE0#	192	HA27	41	OCAWE#/CACS1O#	63
AD16	159	DBCOE1#	191	HA28	42	OCDOE#	59
AD17	158	DEVSEL#	138	HA29	44	PEN#/3VAT#	185
AD18	157	DIRTYI	197	HA30	45	PLOCK#	134
AD19	156	DIRTYWE#,RAS5#	196	HA31	46	PWRGD	195
AD20	155	DLE0#	193	HACALE	64	RAS0#	94
AD21	154	DLE1#	194	HDOE#	190	RAS1#	93
AD22	153	DWE#	90	HITM#	55	RAS2#	92
AD23	152	EADS#/(WB/WT#)	9	HLDA	12	RAS3#	91
AD24	151	ECA4/ADSC#	65	HOLD	11	REFRESH#	105
AD25	150	ECAWE#/CACS0O#	62	HREQ/TMOD#	187	RESET	1
AD26	149	ECDOE#	58	IRDY#	136	SMIACT#	56
AD27	148	ECLK	183	KEN#/LMEM#	8	STOP#	139
AD28	146	FRAME#	135	LA9	122	SUSPEND	129
AD29	145	GND	15	LA10	121	TAG0	207
AD30	144	GND	38	LA11	120	TAG1	206
AD31	143	GND	60	LA12	118	TAG2	205
ADS#	5	GND	75	LA13	117	TAG3	204
AEN	125	GND	83	LA14	116	TAG4	203
AHOLD	13	GND	96	LA15	115	TAG5	202
BE0#	54	GND	119	LA16	113	TAG6	199
BE1#	53	GND	141	LA17	112	TAG7	198
BE2#	52	GND	142	LA18	111	TAGWE#	208
BE3#	51	GND	163	LA19	110	TRDY#	137
BE4#	50	GND	164	LA20	109	VCC	128
BE5#	49	GND	181	LA21	108	VCC	147
BE6#	48	GND	182	LA22	107	VCC	165
BE7#	47	GND	200	LA23	106	VCC	179
BRDY#	6	HA3	16	LCLK	140	VCC	184
C/BE0#	133	HA4	17	LOCK#	14	VCC	201
C/BE1#	132	HA5	18	LRDY#	188	VCC	86
C/BE2#	131	HA6	19	M/IO#	2	VCC	97
C/BE3#	130	HA7	20	MA0	89	VCC	114
CACHE#	57	HA8	21	MA1	88	VCC3	10
CACS0#/CAWE0#	74	HA9	22	MA2	87	VCC3	43
CACS1#/CAWE1#	73	HA10	23	MA3	85	VCC3	61
CACS2#/CAWE2#	72	HA11	24	MA4	84	(W/R#)/INV/DIRTYO	4

Figure 3-3 82C558 Pin Diagram

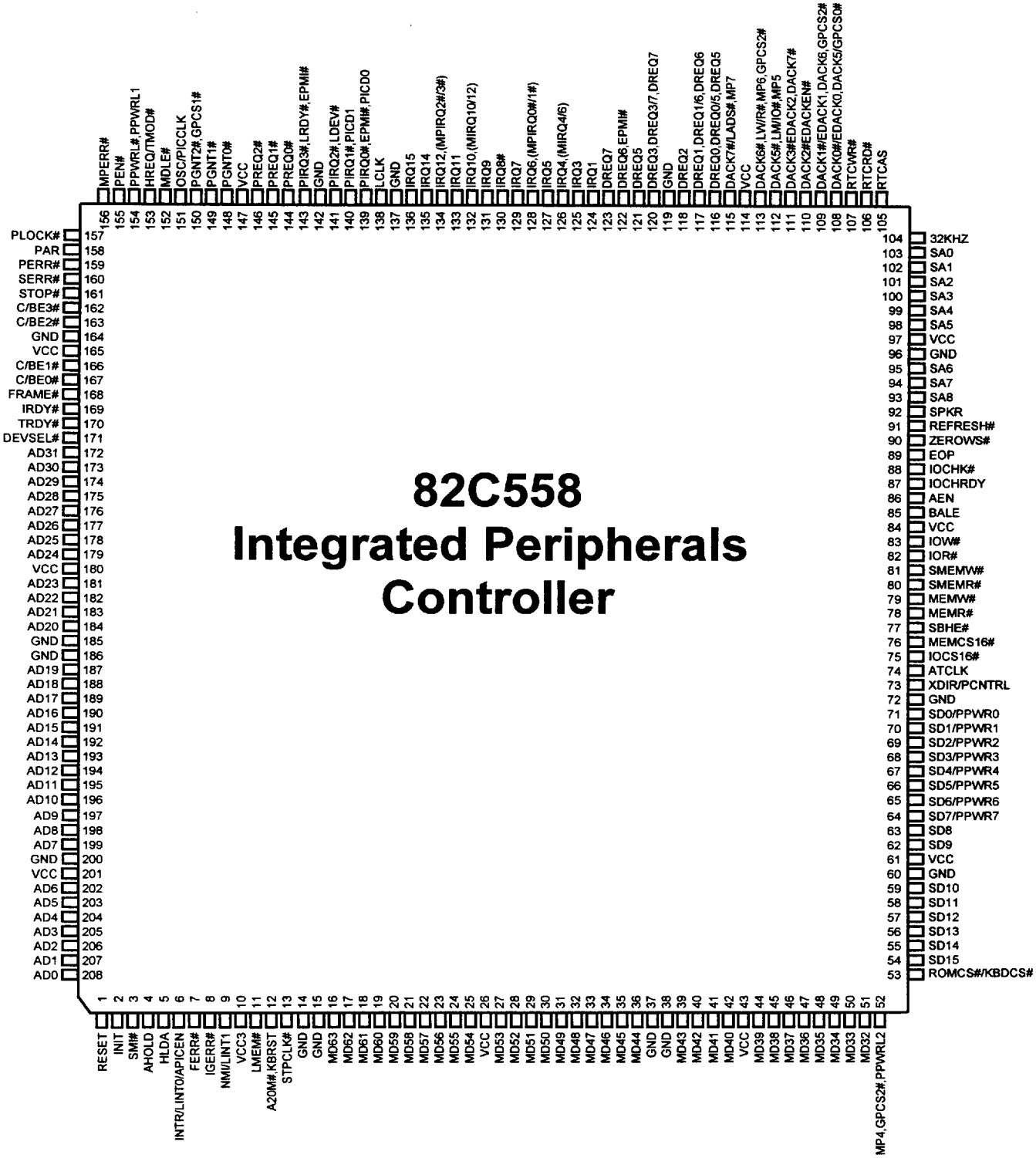


Table 3-5 82C558 Numerical Pin Cross-Reference List

Pin No.	Pin Name
1	RESET
2	INIT
3	SMI#
4	AHOLD
5	HLDA
6	INTR/LINT0/APICEN
7	FERR#
8	IGERR#
9	NMI/LINT1
10	VCC3
11	LMEM#
12	A20M#,KBRST
13	STPCLK#
14	GND
15	GND
16	MD63
17	MD62
18	MD61
19	MD60
20	MD59
21	MD58
22	MD57
23	MD56
24	MD55
25	MD54
26	VCC
27	MD53
28	MD52
29	MD51
30	MD50
31	MD49
32	MD48
33	MD47
34	MD46
35	MD45
36	MD44
37	GND
38	GND
39	MD43
40	MD42
41	MD41
42	MD40
43	VCC
44	MD39
45	MD38
46	MD37
47	MD36
48	MD35
49	MD34
50	MD33
51	MD32
52	MP4,GPCS0#,PPWRL2
53	ROMCS#/KBDCS#
54	SD15
55	SD14
56	SD13
57	SD12
58	SD11
59	SD10
60	GND
61	VCC
62	SD9
63	SD8
64	SD7/PPWR7
65	SD6/PPWR6
66	SD5/PPWR5
67	SD4/PPWR4
68	SD3/PPWR3
69	SD2/PPWR2
70	SD1/PPWR1
71	SD0/PPWR0
72	GND
73	XDIR/PCNTRL
74	ATCLK
75	IOCS16#
76	MEMCS16#
77	SBHE#
78	MEMR#
79	MEMW#
80	SMEMR#
81	SMEMW#
82	IOR#
83	IOW#
84	VCC
85	BALE
86	AEN
87	IOCHRDY
88	IOCHK#
89	EOP
90	ZEROWS#
91	REFRESH#
92	SPKR
93	SA8
94	SA7
95	SA6
96	GND
97	VCC
98	SA5
99	SA4
100	SA3
101	SA2
102	SA1
103	SA0
104	32KHZ
105	RTCAS
106	RTCRD#
107	RTCWR#
108	DACK0#,EDACK0, DACK5#,GPCS0#
109	DACK1#,EDACK1, DACK6#,GPCS2#
110	DACK2#,EDACKEN#
111	DACK3#,EDACK2, DACK7#
112	DACK5#,LM/IO#,MP5
113	DACK6#,LW/R#, MP6,GPCS3#
114	VCC
115	DACK7#,LADS#,MP7
116	DREQ0,DREQ0/5, DREQ5
117	DREQ1,DREQ1/6, DREQ6
118	DREQ2
119	GND
120	DREQ3,DREQ3/7, DREQ7
121	DREQ5
122	DREQ6,EPMI#
123	DREQ7
124	IRQ1
125	IRQ3
126	IRQ4,(MIRQ4/6)
127	IRQ5
128	IRQ6,(MPIRQ0#/1#)
129	IRQ7
130	IRQ8#
131	IRQ9
132	IRQ10,(MIRQ10/12)
133	IRQ11
134	IRQ12,(MPIRQ2#/3#)
135	IRQ14
136	IRQ15
137	GND
138	LCLK
139	PIRQ0#,EPMI#,PICD0
140	PIRQ1#,PICD1
141	PIRQ2#,LDEV#
142	GND
143	PIRQ3#,LRDY#,EPMI#
144	PREQ0#
145	PREQ1#
146	PREQ2#
147	VCC
148	PGNT0#
149	PGNT1#
150	PGNT2#,GPCS1#
151	OSC/PICCLK
152	MDLE#
153	HREQ/TMOD#
154	PPWRL#,PPWRL1
155	PEN#
156	MPERR#
157	PLOCK#
158	PAR
159	PERR#
160	SERR#
161	STOP#
162	C/BE3#
163	C/BE2#
164	GND
165	VCC
166	C/BE1#
167	C/BE0#
168	FRAME#
169	IRDY#
170	TRDY#
171	DEVSEL#
172	AD31
173	AD30
174	AD29
175	AD28
176	AD27
177	AD26
178	AD25
179	AD24
180	VCC
181	AD23
182	AD22
183	AD21
184	AD20
185	GND
186	GND
187	AD19
188	AD18
189	AD17
190	AD16
191	AD15
192	AD14
193	AD13
194	AD12
195	AD11
196	AD10
197	AD9
198	AD8
199	AD7
200	GND
201	VCC
202	AD6
203	AD5
204	AD4
205	AD3
206	AD2
207	AD1
208	AD0

Table 3-6 82C558 Alphabetical Pin Cross-Reference List

Pin No.	Pin Name
208	AD0
207	AD1
206	AD2
205	AD3
204	AD4
203	AD5
202	AD6
199	AD7
198	AD8
197	AD9
196	AD10
195	AD11
194	AD12
193	AD13
192	AD14
191	AD15
190	AD16
189	AD17
188	AD18
187	AD19
184	AD20
183	AD21
182	AD22
181	AD23
179	AD24
178	AD25
177	AD26
176	AD27
175	AD28
174	AD29
173	AD30
172	AD31
86	AEN
4	AHOLD
74	ATCLK
12	A20M#,KBRST
85	BALE
167	C/BE0#
166	C/BE1#
163	C/BE2#
162	C/BE3#
108	DACK0#,EDACK0, DACK5#,GPCS0#
109	DACK1#,EDACK1, DACK6#,GPCS2#
110	DACK2#,EDACKEN#
111	DACK3#,EDACK2, DACK7#
112	DACK5#,LM/IO#,MP5
113	DACK6#,LW/R#, MP6,GPCS2#
115	DACK7#,LADS#,MP7
171	DEVSEL#
116	DREQ0,DREQ0/5, DREQ5
117	DREQ1,DREQ1/6, DREQ6
118	DREQ2
120	DREQ3,DREQ3/7, DREQ7
121	DREQ5
122	DREQ6,EPMI#
123	DREQ7
89	EOP
7	FERR#
168	FRAME#
14	GND
15	GND
37	GND
38	GND
60	GND
72	GND
96	GND
119	GND
137	GND
142	GND
164	GND
185	GND
186	GND
200	GND
5	HLDA
153	HREQ/TMOD#
8	IGERR#
2	INIT
6	INTR/LINT0/APICEN
88	IOCHK#
87	IOCHRDY
75	IOCS16#
82	IOR#
83	IOW#
169	IRDY#
124	IRQ1
125	IRQ3
126	IRQ4,(MIRQ4/6)
127	IRQ5
128	IRQ6,(MPIRQ0#/1#)
129	IRQ7
130	IRQ8#
131	IRQ9
132	IRQ10,(MIRQ10/12)
133	IRQ11
134	IRQ12,(MPIRQ2#/3#)
135	IRQ14
136	IRQ15
138	LCLK
11	LMEM#
51	MD32
50	MD33
49	MD34
48	MD35
47	MD36
46	MD37
45	MD38
44	MD39
42	MD40
41	MD41
40	MD42
39	MD43
36	MD44
35	MD45
34	MD46
33	MD47
32	MD48
31	MD49
30	MD50
29	MD51
28	MD52
27	MD53
25	MD54
24	MD55
23	MD56
22	MD57
21	MD58
20	MD59
19	MD60
18	MD61
17	MD62
16	MD63
152	MDLE#
76	MEMCS16#
78	MEMR#
79	MEMW#
52	MP4,GPCS0#,PPWRL2
156	MPERR#
9	NMI/LINT1
151	OSC/PICCLK
158	PAR
155	PEN#
159	PERR#
148	PGNT0#
149	PGNT1#
150	PGNT2#,GPCS1#
139	PIRQ0#,EPMI#,PICD0
140	PIRQ1#,PICD1
141	PIRQ2#,LDEV#
143	PIRQ3#,LRDY#,EPMI#
157	PLOCK#
154	PPWRL#,PPWRL1
144	PREQ0#
145	PREQ1#
146	PREQ2#
91	REFRESH#
1	RESET
53	ROMCS#/KBDCS#
105	RTCAS
106	RTCRD#
107	RTCWR#
103	SA0
102	SA1
101	SA2
100	SA3
99	SA4
98	SA5
95	SA6
94	SA7
93	SA8
77	SBHE#
71	SD0/PPWR0
70	SD1/PPWR1
69	SD2/PPWR2
68	SD3/PPWR3
67	SD4/PPWR4
66	SD5/PPWR5
65	SD6/PPWR6
64	SD7/PPWR7
63	SD8
62	SD9
59	SD10
58	SD11
57	SD12
56	SD13
55	SD14
54	SD15
160	SERR#
80	SMEMR#
81	SMEMW#
3	SMI#
92	SPKR
161	STOP#
13	STPCLK#
170	TRDY#
26	VCC
43	VCC
61	VCC
84	VCC
97	VCC
114	VCC
147	VCC
165	VCC
180	VCC
201	VCC
10	VCC3
73	XDIR/PCNTRL
90	ZEROWS#
104	32KHZ

3.1 Signal Description Definitions

The "#" symbol at the end of a signal name indicates that the active, or asserted state occurs when the signal is at a low voltage level. When "#" is not present after the signal name, the signal is asserted when at the high voltage level.

The terms "assertion" and "negation" are used extensively. This is done to avoid confusion when working with a mixture of "active-low" and "active-high" signals. The term "assert", or "assertion" indicates that a signal is active, independent of whether that level is represented by a high or low voltage. The term "negate", or "negation" indicates that a signal is inactive.

The "/" symbol between signal names indicates that the signals are "multiplexed" and they use the same pin to deliver the various functions. These signals are time-multiplexed and take on different functions at different instances of time. The "( )" symbol is used to provide a grouping information for the "multiplexed" signals. The "," symbol between signal names indicates that signals are "pin-wise" programmable depending on the configuration registers. The functions that these signals take on has to be decided on boot-up and can only be changed by reprogramming the registers.

Abbreviations: I = TTL level input, O = CMOS level output, and IS = Schmitt-trigger level input.

3.2 82C556 Signal Descriptions

3.2.1 Host Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
HD[63:0]	71, 69:60, 58:51, 49:31, 29:25, 22, 21, 19:1	I/O	**Host Data Bus:** These pins are bidirectional and connected directly to the CPU data bus and L2 cache data lines. There are internal pull-downs on these lines which can be engaged during the Suspend mode or if the HD/MD lines are idle, depending on the strap information sampled on the MP1 and MP2 lines during power-on reset.

3.2.2 SYSC Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
DBCOE[1:0]#	157, 158	I	**DBC Output Enables:** They are connected to the 82C557's DBCOE[1:0]# pins. These signals, along with the MDOE# and the HDOE# signals, form the encoded command sent from the SYSC to the DBC. These commands indicate the type of cycle currently underway and it enables the DBC to perform the appropriate data steering, latching and direction control. The encoded commands are defined in Table 3-9.
MDOE#	156	I	**Memory Data Output Enable:** This signal is used along with the DBCOE[1:0]# signals and the HDOE# signal to form the encoded commands that are sent out by the SYSC. When asserted, this signal enables data to be outputted on the MD bus. MDOE# is asserted for CPU writes to cache/DRAM, CPU writes to PCI, PCI reads from cache/DRAM, L2 cache write-back cycles, and PCI writes to DRAM.
HDOE#	155	I	**Host Data Output Enable:** This signal is used along with the DBCOE[1:0]# signals and the MDOE# signal to form the encoded commands that are sent out by the SYSC. When asserted, this signal enables data to be outputted on the HD bus. HDOE# is asserted for CPU reads from DRAM/PCI/VL bus, PCI writes to cache, CPU linefills, Suspend mode indication, and reset state indication.
DLE[1:0]#	159, 160	I	**Data Latch Enables:** These lines are connected to the SYSC's DLE[1:0]# pins and used to latch the HD and MD data bus depending on which cycle is occurring.

82C556 Signal Descriptions (Cont.)

3.2.3 DRAM Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
MD[63:32]	143, 142, 140:131, 129, 127:109	I/O	**Higher Order Memory Data Bus:** These pins are connected directly to the higher order DRAM data bus and the IPC. This bus serves as a conduit for all high order reads/writes to and from system memory, CPU writes/reads to/from PCI/VL bus/ISA. These lines have internal pull-up resistors.
MD[31:0]#	108:104, 101, 99:92, 89:72	I/O	**Lower Order Memory Data Bus:** These pins are connected directly to the lower order DRAM data bus. During lower order CPU/PCI writes to DRAM, this bus carries the inverted version of the MD[31:0] bus or the MD[63:32] bus. These lines have internal pull-up resistors.
MP[7:6], MP5, MP4/TMOD#, MP3/3VDRAM#, MP[2:0]	144:149, 151, 152	I/O	**Memory Parity:** These pins are connected directly to the system DRAM data bus. As outputs, these lines are only driven when DWE# is active. MP[7:4] can be configured as outputs for PCI master writes with the aid of the strap option on MP0. During power-up reset, MP[4:0] are used to provide strap functions. Table 3-7 details the strap functionality for these pins. MP[7:4] have internal pull-up resistors and MP[3:0] have internal pull-down resistors.
MPERR#	154	O	**Memory Parity Error Indication:** This pin is connected to the MPERR# input of the IPC. It indicates the detection of a parity error during a read from the system DRAM and is qualified within the IPC only when the PEN# output from the SYSC is active.

3.2.4 Power and Ground Pins

Signal Name	Pin No.	Signal Type	Signal Description
GND	23, 24, 50, 70, 90, 102, 103, 130, 150	I	**Ground Connection**
VCC3	20, 30, 59	I	**3.3V Power Connection**
VCC	91, 100, 128, 141, 153	I	**5.0V Power Connection**

Table 3-7 DBC Strap Pin Mapping Functions

Signals	High	Low
MP0	Enable MP[7:4] as outputs from the DBC during PCI master writes.	Disable MP[7:4] as outputs from the DBC during PCI master writes.
MP1	Enable internal pull-downs or pull-ups when HD/MD are not driven.	Disable internal pull-downs or pull-ups when HD/MD are not driven.
MP2	Disable internal pull-downs and pull-ups in Suspend mode.	Enable internal pull-downs and pull-ups in Suspend mode.
MP3/3VDRAM#	5.0V DRAM operation: Should always be high.	3.3V DRAM operation: Only for notebooks.
MP4/TMOD#	Normal Mode	Test Mode

Table 3-8 DBC Operating Voltage Groups

5.0V TTL	CPU/Cache (3.3V)	DRAM (5.0V)
DBCOE[1:0]#	HD[63:0]	MD[63:32]
DLE[1:0]#		MD[31:0]#
MPERR#		MP[7:6]
MDOE#		MP5
HDOE#		MP4/TMOD#
		MP3/3VDRAM#
		MP2
		MP1
		MP0
7	64	72

Table 3-9 DBC Encoded Commands

DBCOE[1:0]#	MDOE#	HDOE#	Description
01	0	1	This code indicates that one of the following cycles is underway: - CPU write low order data to PCI - PCI read low order data from cache The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
11	0	1	This code indicates that one of the following cycles is underway: - L2 cache write-back cycle - CPU write to cache/DRAM - CPU write high order data to PCI - PCI read high order data from cache The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
10	0	1	This code indicates that the following cycles is underway: - PCI read low order data from DRAM The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
10	1	0	This code indicates that currently the system is in the Suspend state and all the buffer outputs should be tristated.
00	1	0	This code indicates that one of the following cycles is underway: - CPU read 64 bits of data from PCI/VLB - CPU read high order data from PCI/VLB - CPU read low order data from PCI/VLB The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
00	0	0	This code indicates that the following cycle is underway: - PCI write to cache/DRAM The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
11	1	0	This code indicates that one of the following cycles is underway: - CPU read from DRAM, and the cache line being replaced is not dirty - CPU linefill The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control
11	1	1	This code indicates that one of the following cycles is underway: - PCI read from PCI - PCI write to PCI - CPU read/write cache hit - PCI read VLB - PCI write VLB - Idle The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control.
10	1	1	This code indicates that the following cycle is underway. - PCI read high order data from DRAM The DBC then, along with the appropriate control signals from the SYSC, performs the required data bus steering, buffering and latching control
01	1	0	This code indicates that the system is in the reset state.

3.3 82C557 SYSC Signal Description

3.3.1 Reset and Clock Signals

Signal Name	Pin No.	Signal Type	Signal Description
RESET	1	O	**System Reset:** When asserted, this signal resets the CPU. RESET is asserted in response to a PWRGD only and is guaranteed to be active for 1ms such that CLK and VCC are stable.
PWRGD	195	IS	**Power Good:** This input reflects the "wired-OR" status of the external reset switch and the power good status from the power supply.
CLK	180	I	**Clock:** This input is used as the master single frequency clock. This signal has to be identical to the clock signal sent to the CPU.
ECLK	183	I	**Early Clock:** This input clock is required to be 3 to 6ns earlier than CLK. This signal is used by the SYSC to generate some critical signals for the host CPU and the cache controller logic.
LCLK	140	I	**Local Bus Clock:** This clock is used by the PCI and local bus state machine within the SYSC. The same clock or another identical signal is used by the local bus devices. For a synchronous PCI/VL implementation, the skew between this input and the CLK input should satisfy the following requirements: LCLK < 1/2 CLK period ahead in phase of CLK LCLK < 0.5ns behind in phase of CLK

3.3.2 Host Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
HA[31:3]	46:44, 42:39, 37:16	I/O	**Host Address Bus:** HA[31:3] are the address lines of the CPU bus. HA[31:3] are connected to the CPU A[31:3] lines. Along with the byte enable signals, the HA[31:3] lines define the physical area of memory or I/O being accessed. During CPU cycles, the HA[31:3] lines are inputs to the SYSC. They are used for address decoding and second level cache tag lookup sequences. During inquire cycles, the HA[31:5] are outputs from the SYSC to the CPU to snoop the first level cache tags. They also are outputs from the SYSC to the L2 cache. HA[31:3] have internal pull-downs, however, external pull-ups are required on HA3 and HA4.
BE[7:0]#	47:54	I/O	**Byte Enable:** The byte enables indicate which byte lanes on the CPU data bus are carrying valid data during the current bus cycle. They are inputs to the SYSC for CPU cycles and outputs for master or DMA cycles. In the case of cacheable reads, all eight bytes of data are driven to the CPU, regardless of the state of the byte enables. The byte enable signals indicate the type of special cycle when M/IO# = D/C# = 0 and W/R# = 1. BE[7:0]# have internal pull-downs that are activated during Suspend or when HLDA is active.

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
M/IO#, D/C#, (W/R#)/INV/ DIRTYO[a]	2, 3, 4	I, B(W/R#)	**Bus Cycle Definition (Memory/Input-Output, Data/Control, Write/Read):** M/IO#, D/C#, W/R# define CPU bus cycles. Interrupt acknowledge cycles are forwarded to the PCI bus as PCI interrupt acknowledge cycles. All I/O cycles and any memory cycles that are not directed to memory controlled by the SYSC's DRAM interface are forwarded to PCI. The W/R# pin serves also as an output signal and is used as INV for L1 cache and DIRTYO for L2 cache during an inquire cycle. If a combined Tag/Dirty RAM implementation is being used, then the W/R# pin does not serve as a DIRTYO pin.
ADS#	5	I	**Address Strobe:** The CPU asserts ADS# to indicate that a new bus cycle is beginning. ADS# is driven active in the same clock as the address, byte enables, and cycle definition signals.
BRDY#	6	O	**Burst Ready:** BRDY# indicates that the system has responded in one of three ways: 1) Valid data has been placed on the CPU data bus in response to a read, 2) CPU write data has been accepted by the system, or 3) the system has responded to a special cycle.
NA#	7	O	**Next Address:** This signal is connected to the CPU's NA# pin to request pipelined addressing for local memory cycle. The SYSC asserts NA# for one clock when the system is ready to accept a new address from the CPU, even if all data transfers for the current cycle have not completed. The 3.3V Pentium processor and the M1 processor support pipelined memory accesses, however, the K5 processor does not support this feature.
KEN#/LMEM#	8	O	**Cache Enable or Local Memory Accessed**: This pin is connected to the KEN# input of the CPU and is used to determine whether the current cycle is cacheable. During master cycles, the SYSC asserts this signal to inform the IPC that local system memory needs to be accessed. The IPC is then responsible for providing the data path to the corresponding master.
EADS#/ (WB/WT#)	9	O	**External Address Strobe or Write-Back/Write-Through:** This output has two functions. It indicates that a valid address has been driven onto the CPU's address bus by an external device. This address will be used to perform an internal cache inquiry cycle when the CPU samples EADS# active. It is also used to control write-back or write-through policy for the primary cache during CPU cycles.
HITM#	55	I	**Hit Modified:** Indicates that the CPU has had a hit on a modified line in its internal cache during an inquire cycle. It is used to prepare for write-back.
CACHE#	57	I	**Cacheability:** It is connected to the CPU's CACHE# pin. It goes active during a CPU initiated cycle to indicate when, an internal cacheable read cycle or a burst write-back cycle, occurs.
SMIACT#	56	I	**System Management Interrupt Active:** The CPU asserts SMIACT# in response to the SMI# signal to indicate that it is operating in System Management Mode (SMM).

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
HOLD	11	O	**CPU Hold Request:** This output is connected to the HOLD input of the CPU. This signal requests that the CPU allow another bus master complete control of its buses. In response to HOLD going active, the CPU will float most of its output and I/O pins and then assert HLDA.
HLDA	12	I	**CPU Hold Acknowledge:** This input is connected to the CPU's HLDA line. This signal indicates, in response to a HOLD, when the CPU has relinquished bus control to another bus master.
AHOLD	13	O	**Address Hold:** This signal is used to tristate the CPU address bus for internal cache snooping.
LOCK#	14	I	**CPU Bus Lock:** The processor asserts LOCK# to indicate the current bus cycle is locked. It is used to generate PLOCK# for the PCI bus. LOCK# has an internal pull-down resistor that is engaged when HLDA is active.

a. In this case the " , " does not mean that they are group-wise programmable, they are separate pins

3.3.3 Cache Control Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
ECDOE#	58	O	**Even Bank Cache Output Enable:** It is connected to the output enables of the SRAMs in the even bank of the L2 cache to enable data read.
OCDOE#	59	O	**Odd Bank Cache Output Enable:** It is connected to output enables of the SRAMs in the odd bank of the L2 cache to enable data read.
ECAWE#/ CACS0O#	62	O	**Even Bank Cache Write Enable or Bank 0 Synchronous SRAM Chip Select:** For asynchronous L2 cache operations this pin becomes ECAWE#, and is connected to the write enables of the SRAMs in the even bank of the L2 cache to enable data update. For synchronous L2 cache operation, this pin provides the chip select for the second bank (synchronous L2 cache is always non-interleaved).
OCAWE#/ CACS1O#	63	O	**Odd Bank Cache Write Enable or Bank 1 Synchronous SRAM Chip Select:** For asynchronous L2 cache operations this pin becomes OCAWE#, and is connected to the write enables of the SRAMs in the odd bank of the L2 cache to enable data update. For synchronous L2 cache operation, this pin provides the chip select for the first bank (synchronous L2 cache is always non-interleaved).
CACS[7:0]#/ CAWE[7:0]#	67:74	O	**Cache Chip Selects 7-0 or Cache Write Enables 7-0:** For asynchronous L2 cache operations these pins become chip selects and are connected to the chip selects of the SRAMs in the L2 cache in both banks to enable data read/write operations. For synchronous L2 cache operation these pins become cache write enables for the SRAMs.

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
TAG[7:0]	198:199, 202:207	I/O	**Tag RAM Data Bits 7-0:** Normally input signals, they become outputs whenever TAGWE# is activated to write new Tags to the Tag RAM. If using a combined Tag/Dirty RAM implementation and a 7-bit Tag is used, then TAG0 functions as the Dirty I/O bit. If using the Sony cache module, then TAG1 and TAG2 are connected to the START# output from the module and TAG3 is connected to the BOFF# output from the module. The remaining TAG bits are unused.
TAGWE#	208	O	**TAG RAM Write Enable:** This control strobe is used to update the Tag RAM with the valid Tag of the new cache line that replaces the current one during external cache read miss cycles. If using a combined Tag/Dirty RAM implementation, this signal functions as both the TAGWE# and DIRTYWE#.
DIRTYI	197	I/O	**Dirty Bit:** This input signal represents the dirty bit of the TAG RAM and is used to indicate whether a corresponding cache line has been overwritten. If using a combined Tag/Dirty implementation, this pin becomes bidirectional. If using a 7-bit Tag in a combined Tag/Dirty RAM implementation, then this pin is not used.
DIRTYWE#, RAS5#	196	O	**Dirty RAM Write Enable or Row Address Strobe bit 5:** This control strobe is used to update the dirty bit RAM when a cache write hit occurs. A cache write hit will set the dirty bit for the currently accessed cache line. If using a combined Tag/Dirty implementation, this signal is not used to update the Dirty RAM. If Index 19h[7] = 0, this pin functions as DIRTYWE#. If Index 19h[7] = 1, this pin functions as RAS5#.
ECA4/ADSC#	65	O	**Even Cache Address 4 or Controller Address strobe:** This pin can be used as even cache address 4 for asynchronous L2 cache operation or as the controller address strobe for synchronous L2 cache operation. For an asynchronous L2 cache, if a single bank is used, this pin is mapped from HA4 and connected to the second LSB of the cache SRAMs' address inputs. For a double bank configuration, it is connected to the LSB of the cache SRAMs' address input in the even bank. For a synchronous L2 cache, this pin is connected to the ADSC# input of the synchronous SRAMs.
OCA4/ECA3/ ADV#	66	O	**Odd Cache Address 4, or Even Cache Address 3, or Advance Output:** This pin can be used as odd cache address 4 or even cache address 3 for asynchronous L2 cache operation, or as the advance pin for synchronous L2 cache operation. For asynchronous caches in a single bank configuration, this pin takes on the functionality of ECA3 and is mapped from HA3 and connected to the cache SRAMs LSB address input. For a double bank configuration, this pin takes on the functionality of OCA4 and is mapped from HA4 and connected to the LSB address input of the SRAMs in the odd bank. For synchronous caches, this pin becomes the advance output and is connected to the ADV# input of the synchronous SRAMs.

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
HACALE	64	O	**Cache Address Latch Enable:** It is used to latch the CPU address and generate latched cache addresses for the L2 cache.

3.3.4 DRAM Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
RAS[3:0]#	91:94	O	**Row Address Strobe bits 3 through 0:** Each RAS# signal corresponds to a unique DRAM bank. Depending on the kind of DRAM modules being used, these signals may or may not need to be buffered externally. These signals, however, should be connected to the corresponding DRAM RAS# lines through a damping resistor. The default drive current on these lines is 4mA, however, by setting Index 18h[4] = 1, it can be increased to 16mA. RAS4# is pin-wise programmable with MA11 and RAS5# is pin-wise programmable with DIRTYWE#.
CAS[7:0]#	95, 98:104	O	**Column Address Strobe bits 7 through 0:** CAS[7:0]# outputs correspond to the eight bytes for each DRAM bank. Each DRAM bank has a 64-bit data bus. These signals are typically connected directly to the DRAMs CAS# inputs through a damping resistor.
DWE#	90	O	**DRAM Write Enable:** This signal is typically buffered externally before connection to the WE# input of the DRAMs. The default drive current on this line is 4mA, however, by setting Index 18h[4] = 1, it can be increased to 16mA.
MA11,RAS4#	76	O	**Memory Address bit 11 or Row Address Strobe bit 4:** If Index 19h[3] = 0, then this pin functions as MA11. In this case, 8Mx36 and 16Mx36 SIMMs will be supported. If Index 19h[3] = 1, this pin functions as RAS4#. In this case, SIMM sizes above 4Mx36 will not be supported and a maximum of 192MB of DRAM will be supported. Depending on the kind of DRAM modules being used, this signal may or may not need to be buffered externally. The default drive current on the MA[11:0] lines is 4mA, however, by setting Index 18h[4] = 1, it can be increased to 16mA.
MA[10:0]	77:82, 84, 83, 87:89	O	**Memory Address Bus:** Multiplexed row/column address lines to the DRAMs. Depending on the kind of DRAM modules being used, these signals may or may not need to be buffered externally. The default drive current on the MA[11:0] lines is 4mA, however, by setting Index 18h[4] = 1, it can be increased to 16mA.
REFRESH#	105	I	**Refresh:** This signal is generated once every 15µs and is derived from the 14.318MHz clock input.

82C557 Signal Descriptions (Cont.)

3.3.5 Local Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
C/BE[3:0]#	130:133	I/O	**PCI Bus Command and Byte Enables 3 through 0:** C/BE[3:0]# are driven by the current bus master (CPU or PCI) during the address phase of a PCI cycle to define the PCI command, and during the data phase as the PCI byte enables. The PCI commands indicate the current cycle type, and the PCI byte enables indicate which byte lanes carry meaningful data. C/BE[3:0]# are outputs from the SYSC during CPU cycles that are directed to the PCI bus. C/BE[3:0]# are inputs during PCI master cycles.
FRAME#	135	I/O	**Cycle Frame:** Every CPU cycle is translated by the SYSC to a PCI cycle if it is not a local memory cycle. FRAME# is asserted by the bus master, SYSC (CPU) or PCI to indicate the beginning and the duration of an access. FRAME# is an input when the SYSC acts as a slave.
IRDY#	136	I/O	**Initiator Ready:** The assertion of IRDY# indicates the current bus master's ability to complete the current data phase. IRDY# works in conjunction with TRDY# to indicate when data has been transferred. A data phase is completed on each clock that TRDY# and IRDY# are both sampled asserted. Wait states are inserted until both IRDY# and TRDY# are asserted together. IRDY# is an output from the SYSC during CPU cycles to the PCI bus. IRDY# is an input when the SYSC acts as a slave.
TRDY#	137	I/O	**Target Ready:** TRDY# indicates the target device's ability to complete the current data phase of the transaction. It is used in conjunction with IRDY#. A data phase is completed on each clock that TRDY# and IRDY# are both sampled asserted. Wait states are inserted on the bus until both IRDY# and TRDY# are asserted together. TRDY# is an output from the SYSC when the SYSC is the PCI slave. TRDY# is an input when the SYSC is a master.
DEVSEL#	138	I/O	**Device Select:** When asserted, DEVSEL# indicates that the driving device has decoded its address as the target of the current access. DEVSEL# is an output of the SYSC when SYSC is a PCI slave. During CPU-to-PCI cycles, DEVSEL# is an input. It is used to determine if any device has responded to the current bus cycle, and to detect a target abort cycle. Master abort termination results if no decode agent exists in the system, and no one asserts DEVSEL# within a fixed number of clocks.
STOP#	139	I/O	**Stop:** STOP# indicates that the current target is requesting the master to stop the current transaction. This signal is used in conjunction with DEVSEL# to indicate disconnect, target abort, and retry cycles. When the SYSC is acting as a master on the PCI bus, if STOP# is sampled active on a rising edge of LCLK, FRAME# is negated within a maximum of three clock cycles. STOP# may be asserted by the SYSC. Once asserted, STOP# remains asserted until FRAME# is negated.
AD[31:0]	143:146, 148:162, 166:178	I/O	**PCI Address and Data:** AD[31:0] are bidirectional address and data lines of the PCI bus. The AD[31:0] signals sample or drive the address and data on the PCI bus. During power-up reset, the SYSC will drive the AD lines by default. This bus also serves as a conduit for receiving address information during ISA master cycles. The IPC conveys the SA[8:0] information to the SYSC on the AD lines.

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
PLOCK#	134	O	**PCI Lock:** PLOCK# is used to indicate an atomic operation that may require multiple transactions to complete. When PLOCK# is asserted, non-exclusive transactions may proceed to an address that is not currently locked.
LRDY#	188	I	**Local Ready:** The VL local bus cycle will be terminated by asserting LRDY#. The IPC terminates VL memory requests by asserting LRDY# when other masters own the bus. This signal should be pulled up externally.

3.3.6 DBC/IPC Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
HREQ/ TMOD#	187	I	**Hold Request or Strap Signal for Test Mode Operation:** Master or DMA cycle request from the Integrated Peripheral Controller (IPC). During power-up reset, this is the strap pin to enter the test mode operation. If TMOD ="high" during power-up reset, it means normal operation. If TMOD ="low", the system enters test mode. An external pull-up is required for normal operation.
DBCOE[1:0]#	191, 192	O	**DBC Output Enables 1 and 0:** These two signals along with the MDOE# and the HDOE# signals, form the encoded commands that are sent out to the DBC. These commands inform the DBC about the current cycle type and enable it to perform the appropriate data steering, latching and direction controls. The encoded commands are defined in Table 3-9.
MDOE#	189	O	**Memory Data Output Enable:** This signal is used along with the DBCOE[1:0]# signals and the HDOE# signal to form the encoded commands that are sent out to the DBC. When asserted, this signal enables data to be outputted on the MD bus. MDOE# is asserted for CPU writes to cache/DRAM, CPU writes to PCI, PCI reads from cache/DRAM, L2 cache write-back cycles, and PCI writes to DRAM.
HDOE#	190	O	**Host Data Output Enable:** This signal is used along with the DBCOE[1:0]# signals and the MDOE# signal to form the encoded commands that are sent out to the DBC. When asserted, this signal enables data to be outputted on the HD bus. HDOE# is asserted for CPU reads from DRAM/PCI/VL bus, PCI writes to cache, CPU linefills, Suspend mode indication, and reset state indication.
DLE[1:0]#	194, 193	O	**Data Latch Enables:** These lines are connected to the DBC's DLE[1:0]# pins and used to latch the HD and MD data bus depending on which cycle is occurring.
PEN#/3VAT#	185	O	**Parity Enable and strap option for 3.3V AT:** This signal is connected to the IPC's PEN# pin and controls the qualification of the memory parity error (MPERR#) signal. At power-up reset, this pin functions as a strapping option for 3.3V or 5.0V AT operation. This pin should be pulled up externally for the Viper-DP Desktop Chipset.

82C557 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
MDLE#/ 3VDRAM#	186	O	**Memory Data Latch Enable and strap option for 3.3V DRAM:** It is connected to the IPC's MDLE# pin to control the data flow from PCI AD[31:0] bus to the high 32-bit memory data bus, MD[63:32], and vice versa. It is used to latch the data during CPU writes to PCI and PCI writes to DRAM and L2 cache. At power-up reset, this pin functions as a strapping option for 3.3V or 5.0V DRAM operation. This pin should be pulled up externally for the Viper-DP Desktop Chipset.
MRDC#	123	I	**Memory Read Command:** This input is connected to the IPC's MEMR# pin to monitor ISA memory read operations.
MWTC#	124	I	**Memory Write Command:** This input is connected to the IPC's MEMW# pin to monitor ISA memory write operations.

3.3.7 AT Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
NVMCS	127	O	**NVRAM Chip Select:** If the current cycle has been decoded as an access to the NVRAM, then this pin is used to issue the chip select signal. NVRAM is used for storing the system configuration information and is required for "plug and play" support. The NVRAM must sit on the XD bus.
LA[23:9]	106:113, 115:118, 120:122	I/O	**System Address Bus:** LA[23:9] and SA[8:0] on the IPC provide the memory and I/O access on the ISA bus and VL bus. The addresses are outputs when the SYSC owns the ISA bus and are inputs when an external ISA master owns the bus. LA[23:9] have internal pull-ups which are disabled when in the Suspend mode.
MASTER#	126	I	**Master:** An ISA bus master asserts MASTER# to indicate that it has control of the ISA bus. Before the ISA master can assert MASTER#, it must first sample DACK# active. Once MASTER# is asserted, the ISA master has control of the ISA bus until it negates MASTER#.
AEN	125	I	**Address Enable:** It is connected to IPC's AEN pin to monitor the ISA bus activity. During power-on reset, if TMOD# is sample low, the AEN pin will be floated. This pin requires an external pull-up.
SUSPEND	129	I	**Suspend:** This signal is used to inform the SYSC about getting into the Suspend mode. SUSPEND needs to be pulled low to resume normal operation.

82C557 Signal Descriptions (Cont.)

3.3.8 Power and Ground Pins

Signal Name	Pin No.	Signal Type	Signal Description
GND	15, 38, 60, 75, 83, 96, 119, 141, 142, 163, 164, 181, 182, 200	I	**Ground Connection**
VCC3	10, 43, 61	I	**3.3V Power Connection**
VCC	86, 97, 114, 128, 147, 165, 179, 184, 201	I	**5.0V Power Connection**

Table 3-10 SYSC Multiplexed Function Pins

Multiplexed Functions	Pin No.	Type	Note
M/IO#	2	Input	
D/C#	3	Input	
(W/R#)/INV/DIRTYO	4	Bidirectional	
KEN#/LMEM#	8	Output	
EADS#/(WB/WT#)	9	Output	
OCA4/ECA3/ADV#	66	Output	
HREQ/TMOD#	187	Input	Multiplexed with TMOD# during power-up reset.
C/BE[3:0]#	130:133	Bidirectional	
ECA4/ADSC#	65	Output	
CACS[7:0]#/CAWE[7:0]#	67:74	Output	
ECAWE#/CACS0O#	62	Output	
OCAWE#/CACS1O#	63	Output	
MDLE#/3VDRAM#	186	Output	
PEN#/3VAT#	185	Output	

Table 3-11 SYSC Strap Function Pins

Default	Strap	Function
HREQ	TMOD#	Low = Test Mode, High = Normal Mode.
MDLE#	3VDRAM#	At power-up reset, this pin functions as a strapping option for 3.3V or 5.0V DRAM operation. This pin should be pulled up externally for the Viper-DP Desktop Chipset.
PEN#	3VAT#	At power-up reset, this pin functions as a strapping option for 3.3V or 5.0V AT operation. This pin should be pulled up externally for the Viper-DP Desktop Chipset.

Table 3-12 System Operating Voltage Groups

5.0V TTL	CPU/Cache (3.3V)	DRAM (5.0V)	AT (5.0V)
CLK	RESET	RAS[3:0]#	MASTER#
ECLK	LOCK#	CAS[7:0]#	AEN
HREQ/TMOD#	HA[31:3]	DWE#	REFRESH#
DBCOE[1:0]#	BE[7:0]#	MA[10:0]	MRDC#
DLE[1:0]#	M/IO#	MA11,RAS4#	MWTC#
PEN#/3VAT#	D/C#		NVMCS
MDLE#/3VDRAM#	(W/R#)/INV/DIRTYO		LA[23:9]
SUSPEND	ADS#		
PWRGD	BRDY#		
LCLK	NA#		
C/BE[3:0]#	KEN#/LMEM#		
FRAME#	EADS#/WB		
IRDY#	HITM#		
TRDY#	CACHE#		
DEVSEL#	SMIACT#		
STOP#	ECDOE#		
AD[31:0]	OCDOE#		
LRDY#	ECAWE#/CACS0O#		
PLOCK#	OCAWE#/CACS1O#		
MDOE#	CACS[7:0]#/CAWE[7:0]#		
HDOE#	ECA4/ADSC#		
TAG[7:0]	OCA4/OCA3/ADV#		
TAGWE#	HACALE		
DIRTYI	HOLD		
DIRTYWE#,RAS5#	HLDA		
	AHOLD		
57	79	25	21

3.4 82C558 IPC Signal Descriptions

3.4.1 Reset and Clock Signals

Signal Name	Pin No.	Signal Type	Signal Description
LCLK	138	I	**Local Bus Clock:** This is the same CLK signal that is also fed into the SYSC. It is used by the PCI and the local bus state machine within the IPC and the SYSC. It is also used by the IPC to derive the AT clock signal. Another identical clock signal is used to clock the PCI and local bus devices.
ATCLK	74	O	**AT Bus Clock:** This signal is derived from an internal division of LCLK. It is used to sample and drive all ISA synchronous signals.
OSC/PICCLK	151	I	**Timer Oscillator or APIC Clock:** This is the main clock used by the internal 8254 timers. It is connected to the 14.31818MHz oscillator. This clock is also used by the APIC when the system is running the 3.3V Pentium/3.3V PentiumM configuration.
INIT	2	O	**CPU Initialize:** A shutdown cycle will trigger INIT, or a low-to-high transition of I/O Port 92h bit 0 will trigger INIT. If keyboard emulation is enabled (default), an INIT will be generated when a Port 64h write cycle with data FEh is decoded. If keyboard emulation has been disabled, then this signal will be triggered when it sees the KBRST from the keyboard.
RESET	1	I	**CPU Reset:** Output from the SYSC in response to a PWRGD input.

3.4.2 SYSC and DBC Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
MD[63:32]	16:25, 27:36, 39:42, 44:51	I/O	**High 32-Bit Memory Data Bus:** These pins are connected directly to the higher order 32 bits of the system DRAM data bus. This is the conduit for all PCI, VLB and ISA device data communication to and from the system.
MDLE#	152	I	**Memory Data Latch Enable:** This input is connected to the SYSC's MDLE# pin to control the data flow from the PCI AD[31:0] bus to the higher order 32-bit memory data bus and vice versa. It is used to latch the data during CPU writes to PCI and PCI writes to DRAM and L2 cache.
HREQ/TMOD#	153	O	**Hold Request / Test Mode:** This signal is connected to the SYSC's HREQ pin to indicate that there is a master or DMA cycle request from the IPC. During power-up reset if TMOD is high, it means normal operation, if TMOD is low, the system enters test mode.
PEN#	155	I	**Parity Enable:** This input is connected to the SYSC's PEN# pin to qualify the memory parity error (MPERR#) signal from DBC.
MPERR#	156	I	**Memory Parity Error:** This input is from the DBC. It is qualified by the parity enable (PEN#) signal from the SYSC to generate an NMI if parity checking is enabled and a parity error occurs.
LMEM#	11	I	**Local Memory Accessed Indication:** This signal serves as a local device memory accessed indication during local bus master cycles.

82C558 Signal Descriptions (Cont.)

3.4.3 PCI Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
C/BE[3:0]#	162, 163, 166, 167	I/O	**PCI Bus Command and Byte Enables:** During the address phase of a transaction, C/BE[3:0]# defines the PCI command. During the data phase, C/BE[3:0]# are used as the PCI byte enables. The PCI commands indicate the current cycle type, and the PCI byte enables indicate which byte lines carry meaningful data. The IPC drives C/BE[3:0]# as an initiator of a PCI bus cycle and monitors C/BE[3:0]# as a target.
PAR	158	I/O	**Calculated Parity Signal:** PAR is "even" parity and is calculated on 36 bits, AD[31:0] plus C/BE[3:0]#. PAR is generated for address and data phases and is only guaranteed to be valid on the PCI clock after the corresponding address or data phase.
FRAME#	168	I/O	**Cycle Frame:** FRAME# is driven by the current bus master to indicate the beginning of a PCI cycle and is maintained asserted for the entire duration of an access. FRAME# is asserted to indicate that a bus transaction is beginning. FRAME# is an input to the IPC when the IPC is the target. FRAME# is an output when the IPC is the initiator.
IRDY#	169	I/O	**Initiator Ready:** IRDY# indicates the IPC's ability, as an initiator, to complete the current data phase of the transaction. It is used in conjunction with TRDY#. A data phase is completed on each clock that both IRDY# and TRDY# are sampled asserted. IRDY# is an input to the IPC when the IPC is the target and an output when the IPC is an initiator.
TRDY#	170	I/O	**Target Ready:** TRDY# indicates the IPC's ability to complete the current data phase of the transaction. It is used in conjunction with IRDY#. A data phase is completed on each clock that TRDY# and IRDY# are both sampled asserted. TRDY# is an input to the IPC when the IPC is the initiator and an output when the IPC is a target.
DEVSEL#	171	I/O	**Device Select:** The IPC asserts DEVSEL# to claim a PCI transaction. As an output, the IPC asserts DEVSEL# when it samples configuration cycles to the IPC configuration registers. As an input, DEVSEL# indicates the response to a transactions. If no slave claims the cycle within four PCICLKs after the assertion of FRAME#, the IPC asserts DEVSEL# to claim the cycle and initiates an ISA cycle.
STOP#	161	I/O	**STOP:** STOP# indicates that the IPC, as a target, is requesting a master to stop the current transaction. As a master, STOP# causes the IPC to stop the current transaction. STOP# is an output when the IPC is a target and an input when the IPC is an initiator
PLOCK#	157	I/O	**PCI Lock:** PLOCK# is used to indicate an atomic operation that may require multiple transactions to complete. When PLOCK# is asserted, non-exclusive transactions may proceed to an address that is not currently locked.
PERR#	159	I/O	**Parity Error:** PERR# may be pulsed by any agent that detects a parity error during an address phase, or by the master or the selected target during any data phase in which the AD[31:0] lines are inputs.
SERR#	160	I/O	**System Error:** SERR# can be pulsed active by any PCI device that detects a system error condition.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
AD[31:0]	172:179, 181:184, 187:199, 202:208	I/O	**PCI Address and Data:** AD[31:0] are bidirectional address and data lines for the PCI bus. The AD[31:0] signals sample or drive the address and data on the PCI bus. This bus also serves as a conduit for transferring address information to the SYSC during ISA master cycles. It conveys the SA[8:0] information to the SYSC on these lines.
PIRQ0#, EPMI#,PICD0	139	I/O	**PCI Interrupt 0, or External PMI Source, or APIC data bit 0:** This pin can be programmed to be PCI interrupt A, or an external power management interrupt source. It can also be programmed to be set up as APIC data bit 0 when the system is running the 3.3V Pentium/3.3V PentiumM configuration. At power-on reset, if the IOAPIC enable strap (on the INTR line) is sampled high, then this signal takes on the PICD0 functionality. Bits 9 and 8 of the register at Address Offset 45h-44h determine the functionality of this pin. If configured to be an EMPI# input, this signal should have an external pull-up. If configured to be PICD0, then it should be connected to the 3.3V Pentium CPU's PICD0 line through a 47ohm series resistor.
PIRQ1#, PICD1	140	I/O	**PCI Interrupt 1 or APIC data bit 1:** If the system is not running the 3.3V Pentium/3.3V PentiumM configuration, then this pin functions as PCI interrupt 1. Otherwise it is used as APIC data bit 1. At power-on reset, if the IOAPIC enable strap (on the INTR line) is sampled high, then this signal takes on the PICD1 functionality. Bits 9 and 8 of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as PICD1, then it should be connected to the 3.3V Pentium's PICD1 line through a 47ohm series resistor.
PIRQ2#, LDEV#	141	I	**PCI Interrupt 2 or VL Device Indicator:** This pin can be programmed to be PCI interrupt 2 or a VL device indicator. Bits 12, 1, and 0 of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as LDEV#, it should have an external pull-up.
PIRQ3#, LRDY#, EPMI#	143	I	**PCI Interrupt 3, or VL Ready, or External PMI Source:** This pin can be programmed to be PCI interrupt 3, a VL ready, or an external power management interrupt source. Bits 13, 1, and 0 of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as LRDY# or EPMI#, it should be pulled up externally.
PREQ0#	144	I	**PCI Request 0:** An active low assertion indicates that the device on PCI slot 1 desires the use of the PCI bus. This signal should be pulled up externally.
PREQ1#	145	I	**PCI Request 1:** An active low assertion indicates that the device on PCI slot 2 desires the use of the PCI bus. This signal should be pulled up externally.
PREQ2#	146	I	**PCI Request 2:** An active low assertion indicates that the device on PCI slot 3 desires the use of the PCI bus. This signal should be pulled up externally.
PGNT[1:0]#	149, 148	O	**PCI Grant 1 and 0:** An active low assertion indicates that one of the initiators on PCI slot 1 or 2 has been granted use of the PCI bus.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
PGNT2#, GPCS1#	150	O	**PCI Grant 2 or General Purpose Chip Select 1:** An active low assertion indicates that the initiator on PCI slot 3 has been granted the use of the PCI bus. This signal can also be programmed to function as general purpose chip select 1. Bit 14 of the register at Address Offset 45h-44h determines the functionality of this pin. 0 = PGNT2# and 1 = GPCS1#. If configured to be GPCS1#, then only two PCI masters can be supported.

3.4.4 CPU Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
SMI#	3	O	**System Management Interrupt:** This signal is used to request System Management Mode (SMM) operation from the CPU.
STPCLK#	13	O	**Stop Clock:** This signal is connected to the STPCLK# input of the CPU. It causes the CPU to get into the STPGNT# state. The M1 processor does not support this signal, hence this signal should remain unconnected if using an M1 processor.
AHOLD	4	I	**Address Hold:** This signal is connected to the SYSC's AHOLD pin and is used to monitor bus arbitration.
HLDA	5	I	**CPU Hold Acknowledge:** This input is connected to the CPU's HLDA line. This signal indicates when the CPU has relinquished bus control to a bus master.
INTR/ LINT0/ APICEN	6	I/O	**Interrupt Request / Local Interrupt 0 / I/O APIC Enable:** INTR is driven by the IPC to signal the CPU that an interrupt request is pending and needs to be serviced. The interrupt controller must be programmed following a reset to ensure that INTR is at a known state. This pin also can be used as a local interrupt 0 when the system is running the 3.3V PentiumC/M configuration. During power-up reset, this pin also serves as a strap signal to enable the local I/O APIC interface in the IPC. If sampled high, the local IOAPIC is enabled.
NMI/ LINT1	9	O	**Non-Maskable Interrupt or Local Interrupt 1:** This signal is activated when a parity error from a local memory read is detected or when the IOCHK# signal from the ISA bus is asserted and the corresponding control bit in Port B is also enabled. The IPC also generates an NMI when either PERR# or SERR# is asserted. This pin can also be used as local interrupt 1 when the system is running a 3.3V PentiumC/M configuration.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
A20M#, KBRST	12	I/O	**Address 20 Mask or Keyboard Reset:** This signal has two functions. It is derived from the keyboard GATEA20 emulation and Port 92h bit 1. It also monitors the keyboard reset signal. If keyboard emulation has been enabled, this pin takes on the A20M# functionality. It outputs A20M# whenever it decodes a Port 92h fast GATEA20 or a keyboard GATEA20. If keyboard emulation has been disabled, this pin takes on the KBRST functionality. It takes the KBRST from the keyboard, and the IPC in response generates an INIT to the CPU. Bit 12 of the register at Index 41h-40h determines the functionality of this pin.

3.4.5 ISA Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
SD[7:0]/ PPWR[7:0]	64:71	I/O	**System Data Bus:** SD[7:0] along with SD[15:8] provides the 16-bit data path for devices residing on the ISA bus. The SD[7:0] pins are also used as the peripheral power control signals latched externally with the PPWRL# signal. These lines should be pulled up externally.
SD[15:8]	54:59, 62, 63	I/O	**System Data Bus:** SD[15:8] are used along with SD[7:0] to provide the 16-bit data path for devices residing on the ISA bus.
SA[8:0]	93:95, 98:103	I/O	**System Address Bus:** The SA[8:0] and LA[23:9] signals on the SYSC provide the address for memory and I/O accesses on the ISA bus. The addresses (SA[8:0]) are outputs when the IPC owns the ISA bus and are inputs when an external ISA master owns the ISA bus.
IOCS16#	75	I	**16-Bit I/O Chip Select:** This signal is driven by I/O devices on the ISA bus to indicate that they support 16-bit I/O bus cycles.
MEMCS16#	76	I/O	**16-Bit Memory Chip Select:** ISA slaves that are 16-bit memory devices drive this signal low. MEMCS16# is an input when the IPC owns the ISA bus.
SBHE#	77	I/O	**System Byte High Enable:** When asserted, SBHE# indicates that a byte is being transferred on the upper byte (SD[15:8]) of the data bus. SBHE# is negated during refresh cycles. SBHE# is an output when the IPC owns the ISA bus.
MEMR#	78	I/O	**Memory Read:** MEMR# is the command to a memory slave that it may drive data onto the ISA data bus. MEMR# is an output when the IPC is a master on the ISA bus. MEMR# is an input when an ISA master, other than IPC, owns the ISA bus.
MEMW#	79	I/O	**Memory Write:** MEMW# is the command to a memory slave that it may latch data from the ISA data bus. MEMW# is an output when the IPC owns the ISA bus. MEMW# is an input when an ISA master, other than the IPC, owns the ISA bus.
AEN	86	O	**Address Enable:** AEN is asserted during DMA cycles to prevent I/O slaves from misinterpreting DMA cycles as valid I/O cycles. When asserted, AEN indicates to an I/O resource on the ISA bus that a DMA transfer is occurring. This signal is also asserted during refresh cycles.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
IOCHRDY	87	I/O	**I/O Channel Ready:** Resources on the ISA bus negates IOCHRDY to indicate that wait states are required to complete the cycle.
IOCHK#	88	I	**I/O Channel Check:** When asserted, this signal indicates that a parity or an uncorrectable error has occurred for a device or memory on the ISA bus.
BALE	85	O	**Bus Address Latch Enable:** BALE is an active high signal asserted by the IPC to indicate that the address, AEN, and SBHE# signal lines are valid. BALE remains asserted throughout ISA master and DMA cycles.
IOR#	82	I/O	**I/O Read:** IOR# is the command to an ISA I/O slave device that the slave may drive data on to the ISA data bus (SD[15:0]). The I/O slave device must hold the data valid until after IOR# is negated. IOR# is an output when the IPC owns the ISA bus. IOR# is an input when an external ISA master owns the ISA bus.
IOW#	83	I/O	**I/O Write:** IOW# is the command to an ISA I/O slave device that the slave may latch data from the ISA data bus (SD[15:0]). IOW# is an output when the IPC owns the ISA bus. IOW# is an input when an external ISA master owns the ISA bus.
SMEMR#	80	I/O	**System Memory Read:** The IPC asserts SMEMR# to request a memory slave to provide data. If the access is below the 1MB range (00000000h-000FFFFFh) during DMA compatible, IPC master, or ISA master cycles, the IPC asserts SMEMR#.
SMEMW#	81	I/O	**System Memory Write:** The IPC asserts SMEMW# to request a memory slave to accept data from the data lines. If the access is below the 1MB range (00000000h-000FFFFFh) during DMA compatible, IPC master, or ISA master cycles, the IPC asserts SMEMW#.
ZEROWS#	90	I	**Zero Wait States:** An ISA slave asserts ZEROWS# after its address and command signals have been decoded to indicate that the current cycle does not require any wait states.
PPWRL#, PPWRL1	154	O	**Peripheral Power Latch Control Signal or Peripheral Power Latch 1:** This signal is used to control the external latching of the peripheral power control signals PPWR[7:0]. This signal is pulsed after reset to preset the external latch. If XDIR is sampled high at reset, then this pin functions as PPWRL#. If XDIR is sampled low at reset, this pin functions as PPWRL1 and can be used to put the clock synthesizer into the power down mode.
XDIR/ PCNTRL	73	I/O	**X Bus Direction / Power Control:** This signal is connected directly to the direction control of a 74F245 that buffers the utility data bus. During power-on reset, this pin is a strap option to decide on what kind of power management scheme is required. If sampled high on reset, the PPWRL# pin functions as a power latch control strobe and MP4,GPCS0#,PPWRL2 takes on its programmed functionality. If sampled low at reset, and PPWRL# functions as PPWRL1 and MP4,GPCS0#,PPWRL2 as PPWRL2. In this case, the external power control latch cannot be used.
ROMCS#/ KBDCS#	53	O	**BIOS ROM Chip Select / Keyboard Chip Select:** This output has two functions. It goes active on both reads and writes to the ROM area to support flash ROM. It is also used to decode accesses to the keyboard controller.

82C558 Signal Descriptions (Cont.)

3.4.6 ISA DMA Arbiter Signals

Signal Name	Pin No.	Signal Type	Signal Description
DREQ0, DREQ0/5, DREQ5	116	I	**DMA Request 0, or Multiplexed DMA Request 0/5, or DMA Request 5:** The DREQ is used to request DMA service from the IPC's DMA controller. It can be programmed to be DMA request 0, a multiplexed DMA request 0/5, or DMA request 5 by bits[3:2] of the register at Address Offset 49h-48h. If configured as a multiplexed input, an external multiplexer is required.
DREQ1, DREQ1/6, DREQ6	117	I	**DMA Request 1, or Multiplexed DMA Request 1/6, or DMA Request 6:** The DREQ is used to request DMA service from the IPC's DMA controller. It can be programmed to be DMA request 1, a multiplexed DMA request 1/6, or DMA request 6 by bits [5:4] of the register at Address Offset 49h-48h. If configured as a multiplexed input, an external multiplexer is required.
DREQ2	118	I	**DMA Request 2:** The DREQ is used to request DMA service from the IPC's DMA controller.
DREQ3, DREQ3/7, DREQ7	120	I	**DMA Request 3, or Multiplexed DMA Request 3/7, or DMA Request 7:** The DREQ is used to request DMA service from the IPC's DMA controller. It can be programmed to be DMA request 3, a multiplexed DMA request 3/7, or DMA request 7 by bits [7:6] of the register at Address Offset 49h-48h. If configured as a multiplexed input, an external multiplexer is required.
DREQ5	121	I	**DMA Request 5:** The DREQ is used to request DMA service from the IPC's DMA controller.
DREQ6, EPMI#	122	I	**DMA Request 6 or External PMI Source:** The DREQ is used to request DMA service from the IPC's DMA controller. Bit 0 of the register at Address Offset 43h-42h can be programmed to control the functionality of this pin (0 = DREQ6 and 1 = EPMI#). If configured as EPMI#, this signal should be pulled up externally.
DREQ7	123	I	**DMA Request 7:** The DREQ is used to request DMA service from the IPC's DMA controller.
DACK0#, EDACK0, DACK5#, GPCS0#	108	O	**DMA Acknowledge 0, or Encoded DACK0, or DMA Acknowledge 5, or General Purpose Chip Select 0:** This pin can be programmed to be DMA acknowledge 0, encoded DACK0, DMA acknowledge 5, or general purpose chip select 0. Bits [7:6] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an encoded DMA acknowledge signal, an external decoder is required.
DACK1#, EDACK1, DACK6#, GPCS2#	109	O	**DMA Acknowledge 1, or Encoded DACK1, or DMA Acknowledge 6, or General Purpose Chip Select 2:** This pin can be programmed to be DMA acknowledge 1, encoded DACK1, or DMA acknowledge 6, or general purpose chip select 2. Bits [5:4] and bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an encoded DMA acknowledge signal, an external decoder is required.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
DACK2#, EDACKEN#	110	O	**DMA Acknowledge 2 or Encoded DACK Enable:** This pin can be programmed to be either DMA acknowledge 2 or an encoded DACK enable signal. Bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an encoded DMA acknowledge signal, an external decoder is required.
DACK3#, EDACK2, DACK7#	111	O	**DMA Acknowledge 3, or Encoded DACK2, or DMA Acknowledge 7:** This pin can be programmed to be DMA acknowledge 3, encoded DACK2, or DMA acknowledge 7. Bits [7:6] and bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If Address Offset 4Fh-4Eh[9] = 1, then this pin takes on the GPCS2# functionality irrespective of the setting in 45h-44h If configured as an encoded DMA acknowledge signal, an external decoder is required.
DACK7#, LADS#, MP7	115	O	**DMA Acknowledge 7, or VL Address Strobe, or Memory Parity 7:** This pin can be programmed to be DMA acknowledge 7, a VL address strobe for a VL bus device, or memory parity bit 7. Bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the DBC has an external pull-down.
DACK6#, LW/R#, MP6, GPCS2#	113	O	**DMA Acknowledge 6, or VL Write/Read, or Memory Parity 6, or General Purpose Chip Select 2:** This pin can be programmed to be DMA acknowledge 6, a VL write/read signal for a VL bus device, memory parity bit 6, or general purpose chip select 2. Bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the DBC has an external pull-down.
DACK5#, LM/IO#, MP5,	112	O	**DMA Acknowledge 5, or VL Memory/Input-Output, or Memory Parity 5:** This pin can be programmed to be DMA acknowledge 5, a VL memory/input-output signal for a VL bus device, or memory parity bit 5. Bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the DBC has an external pull-down.
EOP	89	I/O	**End of Process:** EOP is bidirectional, acting in one of two modes, and is directly connected to the TC line of the ISA bus. DMA slaves assert EOP to the IPC to terminate DMA cycles. The IPC asserts EOP to DMA slaves as a terminal count indicator.
REFRESH#	91	I/O	**Refresh:** As an output, this signal is used to inform the SYSC to refresh the local DRAM. When another bus master has control of the bus, this pin is an input to the IPC.
32KHZ	104	I	**32KHz Clock:** This signal is used as a 32KHz clock input.

82C558 Signal Descriptions (Cont.)

3.4.7 Interrupt Control Signals

Signal Name	Pin No.	Signal Type	Signal Description
IRQ1	124	I	**Interrupt Request 1:** This IRQ1 signal provides the keyboard controller with a mechanism for asynchronously interrupting the CPU.
IRQ3	125	I	**Interrupt Request 3:** This IRQ3 signal provides serial port 2 with a mechanism for asynchronously interrupting the CPU.
IRQ4, (MIRQ4/6)	126	I	**Interrupt Request 4 or Multiplexed IRQ4/6:** This pin can be programmed to be IRQ4 or a multiplexed IRQ4/6. The IRQ4 signal provides serial port 1 with a mechanism for asynchronously interrupting the CPU. The IRQ6 signal provides the floppy disk controller with a mechanism for asynchronously interrupting the CPU. Bit 10 of the register at Address Offset 49h-48h determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ5	127	I	**Interrupt Request 5:** This IRQ5 signal provides parallel port 2 with a mechanism for asynchronously interrupting the CPU.
IRQ6, (MPIRQ0#/1#)	128	I	**Interrupt Request 6 or Multiplexed PCI Interrupt 0/1:** This pin can be programmed to be interrupt request 6 or a multiplexed PCI interrupt 0/1. The IRQ6 signal provides the floppy disk controller with a mechanism for asynchronously interrupting the CPU. Bit 11 of the register at Address Offset 49h-48h determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ8#	130	I	**Interrupt Request 8:** This IRQ8 signal provides the real-time clock with a mechanism for asynchronously interrupting the CPU.
IRQ9	131	I	**Interrupt Request 9:** This pin is used to provide interrupt request 9 to the CPU.
IRQ7	129	I	**Interrupt Request 7:** This IRQ7 signal provides parallel port 1 with a mechanism for asynchronously interrupting the CPU.
IRQ10, (MPIRQ10/12)	132	I	**Interrupt Request 10 or Multiplexed PCI IRQ10/12:** This pin can be programmed to be interrupt request 10 or a multiplexed interrupt 10/12. Bit 12 of the register at Address Offset 49h-48h determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ11	133	I	**Interrupt Request 11:** This pin used to provide interrupt request 11 to the CPU.
IRQ12, (MPIRQ2#/3#)	134	I	**Interrupt Request 12 or Multiplexed PCI Interrupt 2/3:** This pin can be programmed to be interrupt request 12 for a mouse device, or a multiplexed PCI interrupt 2/3. Bits [14:13] of the register at Address Offset 49h-48h determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ14	135	I	**Interrupt Request 14:** This IRQ14 signal provides the expansion slot with a mechanism for asynchronously interrupting the CPU.
IRQ15	136	I	**Interrupt Request 15:** This IRQ15 signal provides the expansion slot with a mechanism for asynchronously interrupting the CPU
FERR#	7	I	**Floating Point Coprocessor Error:** This input causes two operations to occur. IRQ13 is triggered and IGERR# is enabled. An I/O write to Port F0h will set IGERR# low when FERR# is low.

82C558 Signal Descriptions (Cont.)

Signal Name	Pin No.	Signal Type	Signal Description
IGERR#	8	O	**Ignore Coprocessor Error:** Normally high, IGERR# will go low after FERR# goes low and an I/O write to Port 0F0h occurs. When FERR# goes high, IGERR# is driven high.

3.4.8 RTC and Timer Signals

Signal Name	Pin No.	Signal Type	Signal Description
RTCAS	105	O	**RTC Address Strobe:** This output is connected to the external real-time clock's address strobe.
RTCRD#	106	O	**RTC Read:** This pin is used to drive the external real-time clock's read signal.
RTCWR#	107	O	**RTC Write:** This pin is used to drive the external real-time clock's write signal.
SPKR	92	O	**Speaker Data:** This pin is used to drive the system board speaker. This signal is a function of the internal Timer-0 Counter-2 count and bit 1 of Port 61h.

3.4.9 Miscellaneous Signals

Signal Name	Pin No.	Signal Type	Signal Description
MP4, GPCS0#, PPWRL2	52	O	**Memory Parity 4, or General Purpose Chip Select 0, or Peripheral Power Latch 2:** This pin can be programmed to be memory parity bit 4 or general purpose chip select 0. Bits [1:0] of the register at Address Offset 45h-44h determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the DBC has an external pull-down. If XDIR is sampled low at reset, this pin functions as PPWRL2 and can be used to put the clock synthesizer into the Doze mode.

3.4.10 Power and Ground Pins

Signal Name	Pin No.	Signal Type	Signal Description
GND	14, 15, 37, 38, 60, 72, 96, 119, 137, 142, 164, 185, 186, 200	I	**Ground Connection**
VCC3	10	I	**3.3V Power Connection**
VCC	26, 43, 61, 84, 97, 114, 147, 165, 180, 201	I	**5.0V Power Connection**

Figure 3-4 PCI Interrupts Mapping Matrix

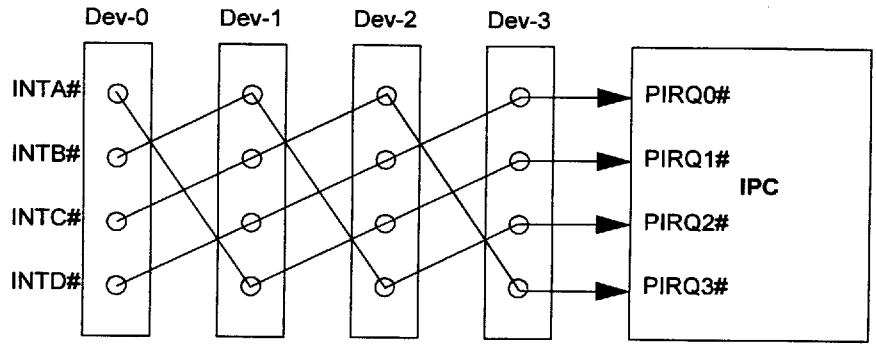


Table 3-13 IPC Programmable Functions Pins

Type	Default	Group-Wise Programmable (Index 44h[1:0])			Pin-Wise Programmable			INTR/ APICEN Strap Option = 1
		01	10	11				
Input	IRQ1				Level			
Input	IRQ3				Level			
Input	IRQ4				Level	MIRQ4/6		
Input	IRQ5				Level			
Input	IRQ6				Level	MPIRQ0#/1#		
Input	IRQ7				Level			
Input	IRQ9				Level			
Input	IRQ10				Level	MIRQ10/12		
Input	IRQ11				Level			
Input	IRQ12				Level	MPIRQ2#/3#		
Input	IRQ14				Level			
Input	IRQ15				Level		EPMI#	
Input	DREQ0				DREQ0/5	DREQ5		
Input	DREQ1				DREQ1/6	DREQ6		
Input	DREQ3				DREQ3/7	DREQ7		
Input	DREQ6						EPMI#	
Input	PREQ2#							
Input	PIRQ0#				EPMI#			PICD0
Input	PIRQ1#							PICD1
Bidirectional	PIRQ2#	PIRQ2#	LDEV#	PIRQ2#				
Bidirectional	PIRQ3#	PIRQ3#	LRDY#	PIRQ3#	EPMI#			
Output	DACK0#	DACK0#	EDACK0	EDACK0	DACK5#		GPCS0#	
Output	DACK1#	DACK1#	EDACK1	EDACK1		DACK6#	GPCS2#	
Output	DACK2#	DACK2#	EDACKEN#	EDACKEN#				
Output	DACK3#	DACK3#	EDACK2	EDACK2		DACK7#		
Output	MP4	GPCS0#	MP4	MP4	PPWRL2			
Output	MP5	DACK5#	LMIO	MP5				
Output	MP6	DACK6#	LWR	MP6	GPCS2#			
Output	MP7	DACK7#	LADS#	MP7				
Output	PGNT2#				GPCS1#			
Output	PPWRL#				PPWRL1			

Refer to Tables 5-64 through 5-66 for more details.

Table 3-14 IPC Multiplexed Function Pins

Type	Multiplexed Functions	Note
Input	OSC/PICCLK	
Bidirectional	C/BE[3:0]	
Bidirectional	SD[7:0]/PPWR[7:0]	Valid ONLY when PPWRL# is selected.
Bidirectional	INTR/LINT0/APICEN	INTR is selected in the single processor configuration and LINT0 is selected in 3.3V Pentium/3.3V PentiumM mode. During power-up reset, this pin provides the APIC enable strap signal. When APICEN is pulled high, the I/O APIC is enabled.
Output	NMI/LINT1	NMI is selected in the single processor configuration and LINT1 is selected in 3.3V Pentium/3.3V PentiumM mode.
Input	MIRQ4/6	Valid ONLY when IRQ4 is programmed to MIRQ4/6 function.
Input	MPIRQ0#/1#	Valid ONLY when IRQ6 is programmed to MPIRQ0#/1# function.
Input	MPIRQ2#/3#	Valid ONLY when IRQ12 is programmed to MPIRQ2#/3# function.
Input	DREQ0/5	Valid ONLY when DREQ0 is programmed to DREQ0/5 function.
Input	DREQ1/6	Valid ONLY when DREQ1 is programmed to DREQ1/6 function.
Input	DREQ3/7	Valid ONLY when DREQ3 is programmed to DREQ3/7 function.
Bidirectional	XDIR/PCTNRL	
Bidirectional	HREQ/TMOD#	

Table 3-15 IPC Strap Function Pins

Default	Strap	Function
HREQ	TMOD#	Low = Test Mode High = Normal Mode
XDIR	PCNTRL	Low = Internal power management scheme High = External power management scheme.
INTR/LINT0	APICEN	Low = Local IOAPIC disabled High = Local IOAPIC enabled.

Table 3-16 IPC Operating Voltage Group

5.0V TTL	CPU/Cache (3.3V)	DRAM (5.0V)	AT (5.0V)
OSC/PICCLK	RESET	MD[63:32]	ATCLK
MDLE#/3VDRAM#	SMI#	MP4,GPCS0#,PPWRL2	PPWRL#,PPWRL1
HREQ	STPCLK#		SD[7:0]/PPWR[7:0]
PEN#	AHOLD		SA[8:0]
MPERR#	HLDA		IOCS16#
LCLK	INTR/LINT0/APICEN		MEMCS16#
C/BE[3:0]#	FERR#		SBHE#
PAR	IGERR#		MEMR#
FRAME#	NMI/LINT1		MEMW#
IRDY#	LMEM#		AEN
TRDY#	INIT		IOCHRDY
DEVSEL#			IOCHK#
STOP#			BALE
PLOCK#			IOR#
PERR#			IOW#
SERR#			SMEMR#
AD[31:0]			SMEMW#
PIRQ0#,EPMI#,PICD0			ZEROWS#
PIRQ1#,PICD1			DREQ0,DREQ0/5,DREQ5
PIRQ2#,LDEV#			DREQ1,DREQ1/6,DREQ6
PIRQ3#,LRDY#,EPMI#			DREQ2
PREQ0#			DREQ3,DREQ3/7,DREQ7
PREQ1#			DREQ5
PREQ2#			DREQ6,EPMI#
PGNT[1:0]#			DREQ7
PGNT2#,GPCS1#			DACK0#,EDACK0,DACK5#,GPCS0#
			DACK1#, EDACK1,DACK6#,GPCS2#
			DACK2#,EDACKEN#
			DACK3#, EDACK2,DACK7#
			DACK5#, LM/IO#,MP5
			DACK6#, LW/R#, MP6
			DACK7#, LADS#, MP7
			EOP
			REFRESH#
			32KHZ
			IRQ1
			IRQ3
			IRQ5
			IRQ4,(MIRQ4/6)
			IRQ6,(MPIRQ0#/1#)
			IRQ8#
			IRQ9
			IRQ10,(MIRQ10/12)
			IRQ11
			IRQ12,(MPIRQ2#/3#)
			IRQ14
			IRQ15
			RTCAS
			RTCRD#
			RTCWR#
			SPKR
			ROMCS#/KBDCS#
			XDIR/PCNTRL
			SD[15:8]
			IRQ7
9	11	33	77

Table 3-13 IPC Programmable Functions Pins

Type	Default	Group-Wise Programmable (Index 44h[1:0])			Pin-Wise Programmable			INTR/ APICEN Strap Option = 1
		01	10	11				
Input	IRQ1				Level			
Input	IRQ3				Level			
Input	IRQ4				Level	MIRQ4/6		
Input	IRQ5				Level			
Input	IRQ6				Level	MPIRQ0#/1#		
Input	IRQ7				Level			
Input	IRQ9				Level			
Input	IRQ10				Level	MIRQ10/12		
Input	IRQ11				Level			
Input	IRQ12				Level	MPIRQ2#/3#		
Input	IRQ14				Level			
Input	IRQ15				Level		EPMI#	
Input	DREQ0				DREQ0/5	DREQ5		
Input	DREQ1				DREQ1/6	DREQ6		
Input	DREQ3				DREQ3/7	DREQ7		
Input	DREQ6						EPMI#	
Input	PREQ2#							
Input	PIRQ0#				EPMI#			PICD0
Input	PIRQ1#							PICD1
Bidirectional	PIRQ2#	PIRQ2#	LDEV#	PIRQ2#				
Bidirectional	PIRQ3#	PIRQ3#	LRDY#	PIRQ3#	EPMI#			
Output	DACK0#	DACK0#	EDACK0	EDACK0	DACK5#		GPCS0#	
Output	DACK1#	DACK1#	EDACK1	EDACK1		DACK6#	GPCS2#	
Output	DACK2#	DACK2#	EDACKEN#	EDACKEN#				
Output	DACK3#	DACK3#	EDACK2	EDACK2		DACK7#		
Output	MP4	GPCS0#	MP4	MP4	PPWRL2			
Output	MP5	DACK5#	LMIO	MP5				
Output	MP6	DACK6#	LWR	MP6	GPCS2#			
Output	MP7	DACK7#	LADS#	MP7				
Output	PGNT2#				GPCS1#			
Output	PPWRL#				PPWRL1			

Refer to Tables 5-64 through 5-66 for more details.

Table 3-14 IPC Multiplexed Function Pins

Type	Multiplexed Functions	Note
Input	OSC/PICCLK	
Bidirectional	C/BE[3:0]	
Bidirectional	SD[7:0]/PPWR[7:0]	Valid ONLY when PPWRL# is selected.
Bidirectional	INTR/LINT0/APICEN	INTR is selected in the single processor configuration and LINT0 is selected in 3.3V Pentium/3.3V PentiumM mode. During power-up reset, this pin provides the APIC enable strap signal. When APICEN is pulled high, the I/O APIC is enabled.
Output	NMI/LINT1	NMI is selected in the single processor configuration and LINT1 is selected in 3.3V Pentium/3.3V PentiumM mode.
Input	MIRQ4/6	Valid ONLY when IRQ4 is programmed to MIRQ4/6 function.
Input	MPIRQ0#/1#	Valid ONLY when IRQ6 is programmed to MPIRQ0#/1# function.
Input	MPIRQ2#/3#	Valid ONLY when IRQ12 is programmed to MPIRQ2#/3# function.
Input	DREQ0/5	Valid ONLY when DREQ0 is programmed to DREQ0/5 function.
Input	DREQ1/6	Valid ONLY when DREQ1 is programmed to DREQ1/6 function.
Input	DREQ3/7	Valid ONLY when DREQ3 is programmed to DREQ3/7 function.
Bidirectional	XDIR/PCTNRL	
Bidirectional	HREQ/TMOD#	

Table 3-15 IPC Strap Function Pins

Default	Strap	Function
HREQ	TMOD#	Low = Test Mode High = Normal Mode
XDIR	PCNTRL	Low = Internal power management scheme High = External power management scheme.
INTR/LINT0	APICEN	Low = Local IOAPIC disabled High = Local IOAPIC enabled.

Table 3-16 IPC Operating Voltage Group

5.0V TTL	CPU/Cache (3.3V)	DRAM (5.0V)	AT (5.0V)
OSC/PICCLK	RESET	MD[63:32]	ATCLK
MDLE#/3VDRAM#	SMI#	MP4,GPCS0#,PPWRL2	PPWRL#,PPWRL1
HREQ	STPCLK#		SD[7:0]/PPWR[7:0]
PEN#	AHOLD		SA[8:0]
MPERR#	HLDA		IOCS16#
LCLK	INTR/LINT0/APICEN		MEMCS16#
C/BE[3:0]#	FERR#		SBHE#
PAR	IGERR#		MEMR#
FRAME#	NMI/LINT1		MEMW#
IRDY#	LMEM#		AEN
TRDY#	INIT		IOCHRDY
DEVSEL#			IOCHK#
STOP#			BALE
PLOCK#			IOR#
PERR#			IOW#
SERR#			SMEMR#
AD[31:0]			SMEMW#
PIRQ0#,EPMI#,PICD0			ZEROWS#
PIRQ1#,PICD1			DREQ0,DREQ0/5,DREQ5
PIRQ2#,LDEV#			DREQ1,DREQ1/6,DREQ6
PIRQ3#,LRDY#,EPMI#			DREQ2
PREQ0#			DREQ3,DREQ3/7,DREQ7
PREQ1#			DREQ5
PREQ2#			DREQ6,EPMI#
PGNT[1:0]#			DREQ7
PGNT2#,GPCS1#			DACK0#,EDACK0,DACK5#,GPCS0#
			DACK1#, EDACK1,DACK6#,GPCS2#
			DACK2#,EDACKEN#
			DACK3#, EDACK2,DACK7#
			DACK5#, LM/IO#,MP5
			DACK6#, LW/R#, MP6
			DACK7#, LADS#, MP7
			EOP
			REFRESH#
			32KHZ
			IRQ1
			IRQ3
			IRQ5
			IRQ4,(MIRQ4/6)
			IRQ6,(MPIRQ0#/1#)
			IRQ8#
			IRQ9
			IRQ10,(MIRQ10/12)
			IRQ11
			IRQ12,(MPIRQ2#/3#)
			IRQ14
			IRQ15
			RTCAS
			RTCRD#
			RTCWR#
			SPKR
			ROMCS#/KBDCS#
			XDIR/PCNTRL
			SD[15:8]
			IRQ7
9	11	33	77

4.0 Functional Description

4.1 Reset Logic

The PWRGD input to the SYSC is used to generate the CPU and the system reset (CPURST). PWRGD is a "cold reset" which is generated when either PWRGD goes low (from the power supply, indicating a low power condition) or the system reset button is activated. When PWRGD makes a low-to-high transition, CPURST will go active and will remain active for at least 1ms after PWRGD goes high.

The INIT signal is used to initialize the 3.3V CPU during warm resets. INIT is generated for the following cases:

- When a shutdown condition is decoded from the CPU bus definition signals, the IPC will assert INIT for 15 T-states.
- Keyboard reset to I/O Port 64h.
- Fast reset to I/O Port 92h.

4.2 System Clocks

4.2.1 CPU and SYSC Clocks

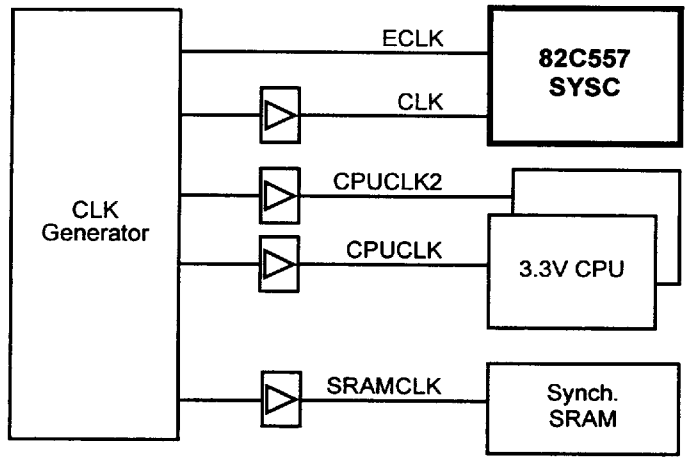
The SYSC uses two high frequency clock inputs, CLK and ECLK.

The clock signals that go to the CPU and the SYSC CLK inputs are required to be in the same phase and have minimum skew between them. The skew between the CLK input to the SYSC and the CLK input to the CPU should not exceed 2ns. The SYSC CLK is a single phase clock which is used to sample all host CPU synchronous signals and for clocking the SYSC's internal state machines.

ECLK literally means "Early Clock". ECLK is used for generating some critical signals for the host CPU and the cache controller logic. Its main use is to clock signals out earlier so that the signals are guaranteed to meet setup times of the CPU and cache. ECLK is required to be in the same phase as CLK but ahead of CLK. The delay from ECLK to CLK must meet the delay timing of a minimum of 3ns and a maximum of 6ns. Typically, a one gate delay from ECLK to CLK meets the SYSC CLK delay requirements.

Figure 4-1 shows the relationship between CPUCLK, CLK, and ECLK and the typical CPU and SYSC clock distribution circuit.

Figure 4-1 CPU and SYSC Clock Distribution



4.2.2 PCI and VL Bus Clocks

The SYSC and the IPC require LCLK for the PCI and VL bus interface. The phase and frequency of the LCLK input to the 82C557 and 82C558 and the LCLK inputs to the PCI and VL bus is required to be the same and the maximum skew should not exceed 2ns. Figure 4-2 and Figure 4-3 show possible clock generation and distribution schemes for LCLK. The local bus can be asynchronous/synchronous to the CPU bus, but must be synchronous to the PCI bus.

4.2.3 AT Bus Clocks

The IPC generates the AT bus clock (ATCLK) from an internal division of LCLK. The ATCLK frequency is programmable and can be set to any of the four clock division options: LCLK/1, LCLK/2, LCLK/3, LCLK/4. This allows the system designer to tailor the AT bus clock frequency to support a wide range of system designs and performance platforms.

Figure 4-2 Clock Distribution Method for VL Bus and PCI Connectors (Async. PCI and VL Bus)

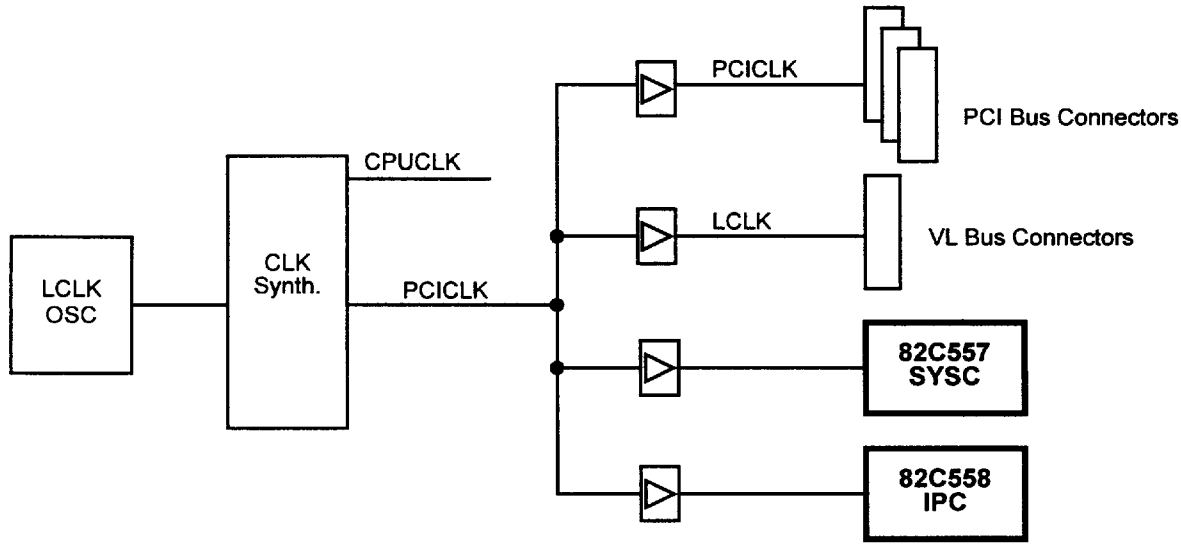
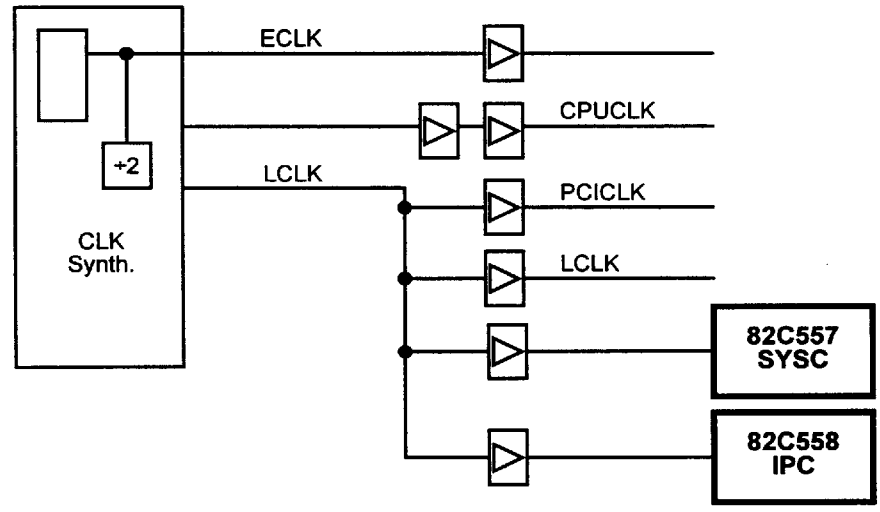


Figure 4-3 Clock Distribution Method for VL Bus and PCI Connectors (Sync. PCI and VL Bus)



4.3 Cache Subsystem

The integrated cache controller, which uses a direct-mapped, bank-interleaved scheme (for an asynchronous SRAM implementation) dramatically boosts the overall performance of the local memory subsystem by caching writes as well as reads (write-back mode). Cache memory can be configured as one or two banks of asynchronous SRAMs and sizes of 64KB, 128KB, 256KB, 512KB, 1MB and 2MB are supported. In addition, the cache controller also supports 256KB, 512KB, 1MB, and 2MB of synchronous SRAM in a single/double bank configuration. Two programmable non-cacheable regions are provided. The cache controller operates in a non-pipelined or a pipelined mode, with a fixed 32-byte line size (optimized to match a CPU burst linefill) in order to simplify the motherboard design without increasing cost or degrading system performance. The secondary cache operates independently and in addition to the CPU's internal cache.

The SYSC's cache controller has a built-in tag comparator which improves system performance while reducing component count on the system board. The controller features a 64-bit wide data bus with 32-byte CPU burst support. The cache controller supports both write-back, adaptive write-back, and write-through schemes.

The cache controller uses a 32-byte secondary cache line size. It supports read and write bursting in 3-2-2-2 bursts for the asynchronous SRAM, and 3-1-1-1 burst read/write for synchronous SRAMs. 2-1-1-1 burst read/write cycles are supported for synchronous SRAMs at 50MHz. In this case, the ADSC# output of the processor needs to be connected to the ADSC# input of the synchronous SRAM. The 8-bit tag has a "dirty" bit option for the write-back cache. The cache controller uses standard single bank SRAMs or dual bank SRAMs with interleaving (only in the case for asynchronous SRAM) for optimum cache performance.

4.3.1 CPU Burst Mode Control

The Viper-DP Desktop Chipset fully supports the 64-bit wide data path for the CPU burst read and burst write cycles. The SYSC's cache and DRAM controllers ensure that data is burst into the CPU whenever the CPU requests a burst linefill or a burst write to the system memory.

The SYSC contains separate burst counters to support DRAM and external cache burst cycles. The DRAM controller performs a burst for the L2 cache read miss linefill cycle (DRAM to L2 cache and CPU) and the cache controller burst supports the CPU burst linefill (3.3V Pentium and K5 burst linefill and the Cyrix M1 linear burst linefill) for the L2 cache hit cycle (L2 cache to the 3.3V Pentium CPU). Depending on the kind of processor being used, either the 3.3V Pentium quad word burst address sequencing or the Cyrix M1 quad word linear burst address sequencing is used for all system memory burst cycles.

4.3.1.1 Cyrix Linear Burst Mode Support

The Viper-DP Desktop Chipset supports the Cyrix linear burst mode. Index 17h[0] in the 82C557 SYSC determines which burst mode is to be implemented - the Intel 3.3V Pentium CPU burst mode or the Cyrix linear burst mode. No additional hardware is required for supporting either of these modes.

When using a synchronous SRAM solution, care must taken that the synchronous SRAM burst protocol complements the processor's burst protocol. Table 4-1 shows the burst mode sequence for both of these processors.

Table 4-1 Burst Modes

1st Address	2nd Address	3rd Address	4th Address
Cyrix Linear Burst Mode			
0	8	10	18
8	10	18	0
10	18	0	8
18	0	8	10
Intel Burst Mode			
0	8	10	18
8	0	18	10
10	18	0	8
18	10	8	0

4.3.2 Cache Cycle Types

Some cache terminology and cycle definitions that are chipset specific:

The cache hit/miss status is generated by comparing the high-order address bits (for the memory cycle in progress) with the stored tag bits from previous cache entries. When a match is detected and the location is cacheable, a cache hit cycle takes place. If the comparator does not detect a match or a non-cacheable location is accessed (based on the internal non-cacheable region registers), then the current cycle is a cache miss.

A cache hit/miss decision is always made at the end of the first T2 for a non-pipeline cycle and at the end of the first T2P for a pipeline cycle, so the SRAM read/write cycle will begin after the first T2 or T2P. The cacheable decision is based on the DRAM bank decodes and the Chipset's configuration registers for non-system memory areas and non-cacheable area definitions. If the access falls outside the system memory area, it is always non-cacheable.

The dirty bit is a mechanism for monitoring coherency between the cache and system memory. Each tag entry has a corresponding dirty bit to indicate whether the data in the represented cache line has been modified since it was loaded from system memory. This allows the SYSC to determine whether the data in the system memory is "stale" and needs to be updated before a new memory location is allowed to overwrite the currently indexed cache entry. The Viper-DP supports several Tag/Dirty schemes and those are described in Section 4.3.3.7.

A linefill cycle occurs for a cache read miss cycle. It is a data read of the new address location from the system memory and a corresponding write to the cache. The tag data will also be updated with the new address.

A castout cycle occurs for a cache read miss cycle, but only if the cache line that is being replaced is "dirty". In this cycle, the dirty cache line is read from the cache and written to the system memory. The upper address bits for this cycle are provided by the tag data bits.

A write-back cycle consists of performing a castout cycle followed by a linefill cycle. The write-back cycle causes an entire cache line (32 bytes) to be written back to memory followed by a line burst from the new memory location into the cache and to the CPU simultaneously. The advantages of performing fast write cycles to the cache (for a write hit) typically outweigh the cycle overhead incurred by the write-back scheme.

4.3.3 Cache Operation

The following discussion pertains to asynchronous SRAMs, but is valid for the synchronous SRAM as well, except that the synchronous SRAM supports 3-1-1-1 cycles and 2-1-1-1 cycles at 50MHz instead of 3-2-2-2 cycles.

4.3.3.1 L2 Cache Read Hit

On an L1 read miss and an L2 read hit, the secondary cache provides data to the CPU. The SYSC follows either the 3.3V Pentium CPU's burst protocol or the M1's linear burst mode protocol to fill the processor's internal cache line.

The cache controller will sample CACHE# from the CPU at the end of T1 and perform a burst read if CACHE# is sampled active. The first cache read hit for a cycle is always one wait state. If a read cycle can be converted to a burst, the read cycle is extended for the additional three words continuing at one wait state per cycle. To achieve the burst at this rate, the hit or miss decision must be made before BRDY# is returned to the CPU at the end of the second T2. The cache hit comparator in the SYSC compares the data from the tag RAM with the higher address bits from the CPU bus. The output of this comparator generates the BRDY# signal to the 3.3V Pentium CPU. The tag comparator's output is sampled at the end of the first T2, and BRDY# is generated one clock later for cache hits, resulting in a leadoff of three cycles. BRDY# will go inactive to add wait states depending on the wait states programmed. Please refer to Table 4-2 for the tag compare table.

If two SRAM banks are used, address bit A4 from the CPU will be the least significant address bit that goes to the data SRAMs. The data output for each SRAM bank is controlled by a separate output enable for each SRAM bank (OCDOE# and ECDOE#). The OCDOE#/ECDOE# generation for the leadoff cycle is based on address bit A3 from the CPU. The two signals OCDOE# and ECDOE# will interleave the data read from the two cache banks in a burst cycle. If one SRAM bank is used, address bit A3 from the CPU will be the least significant address bit that goes to the SRAMs and the output enable ECDOE# will be active for the complete cycle.

Figures 4-4 through 4-6 show various L2 cache read hit cycles.

Figure 4-4 L2 Cache Read Hit Cycle - Async. SRAMs (Double Bank)

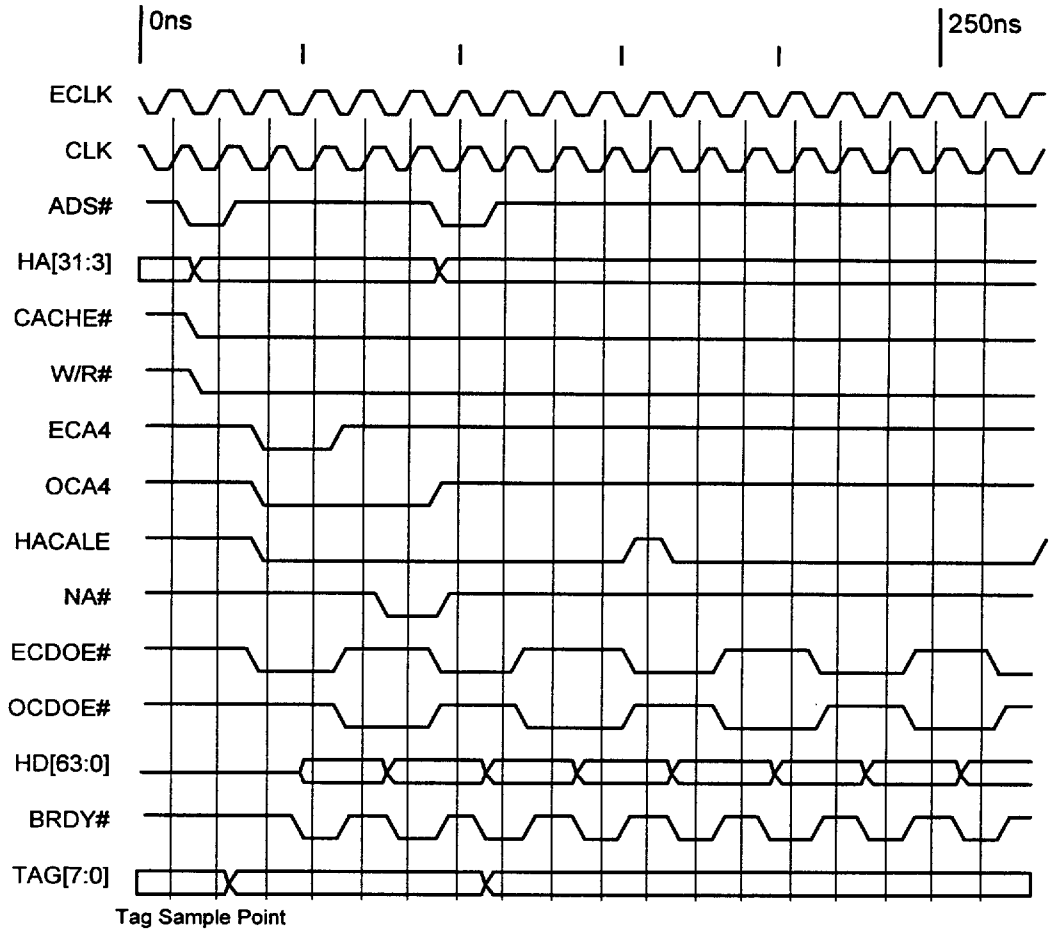


Figure 4-5 L2 Cache Read Hit Cycle - Sync. SRAMs

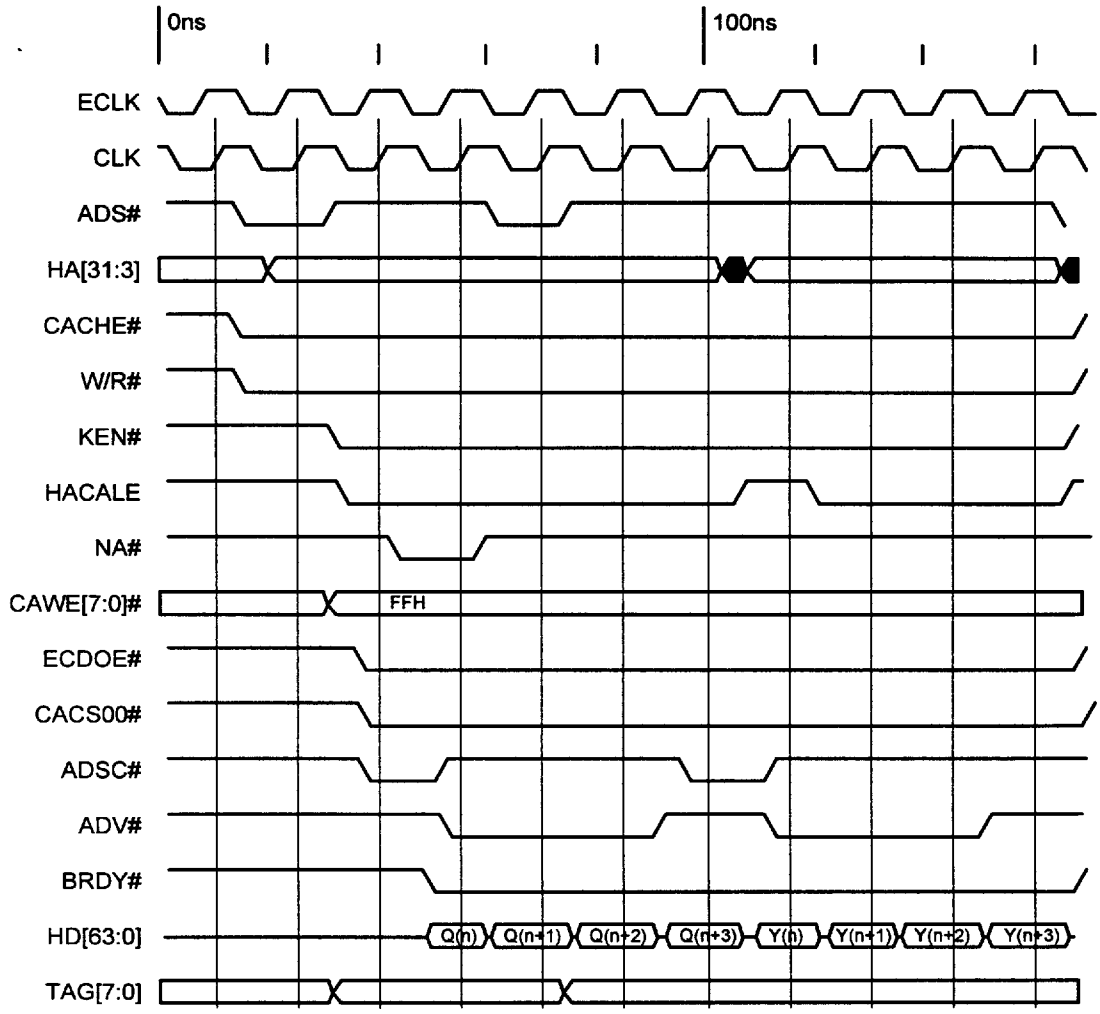
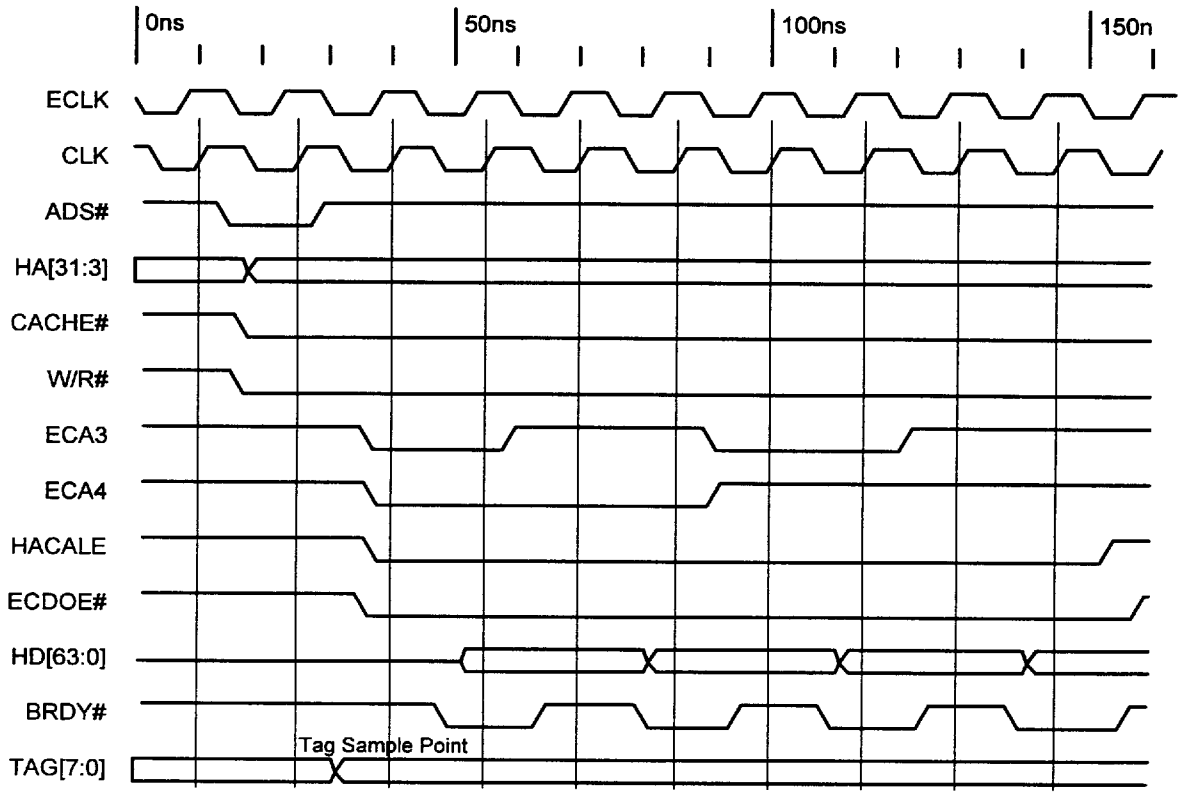


Figure 4-6 L2 Cache Read Hit Cycle Async. SRAMs (Single Bank)



4.3.3.2 L2 Cache Write Hit Cycle

Write-through Mode: In this mode, data is always written to the L2 cache and to the system memory. The dirty bit is not used. When the write to the system memory is completed, BRDY# is returned to the CPU.

Write-back Mode: For a write hit case, the data is written only to the L2 cache (the system memory is not updated) and the dirty bit is always made dirty. The cache controller will sample CACHE# from the CPU at the end of T1 and execute a burst write if CACHE# is sampled active, otherwise the cycle will end in a single write. In this mode, the write cycle is completed in a 3-2-2-2 burst. For synchronous SRAMs, the cycle can be completed in a 2-1-1-1 burst if operating at 50MHz. The write enable signals OCAWE# and ECAWE# to the SRAM odd and even banks respectively, are based on address bit A3 from the CPU and will interleave writes to the two banks.

For writes, only the byte requested by the CPU can be written to the cache. This is done by using the BEx# from the CPU to control the SRAM chip selects.

Refer to Figures 4-7 through 4-9 show various write hit burst cycles.

Figure 4-7 Write Hit Burst Cycle for Write-Back Mode - Async. SRAM (Double Bank)

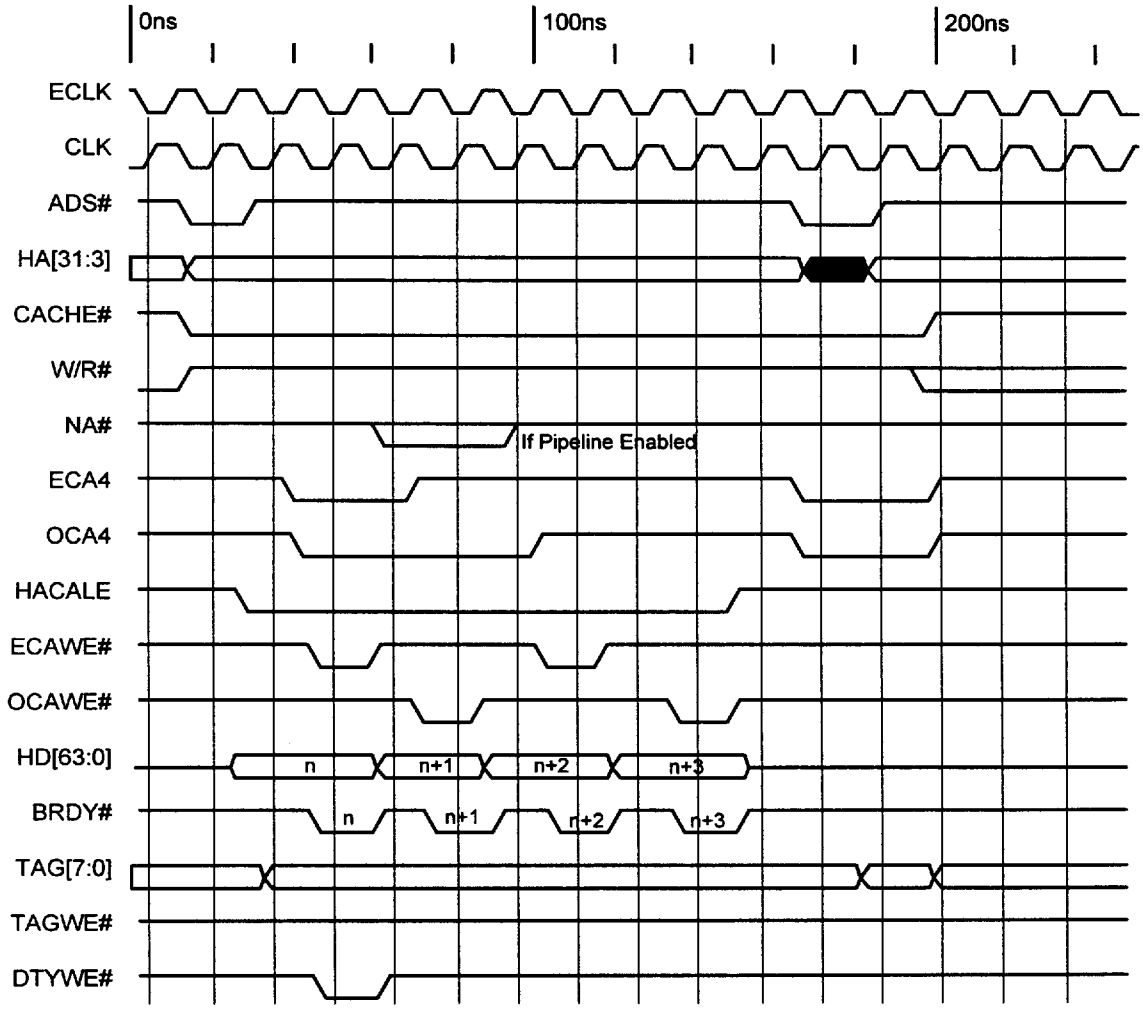


Figure 4-8 Write Hit Burst Cycle for Write-Back Mode (Single Bank) - Async. SRAM

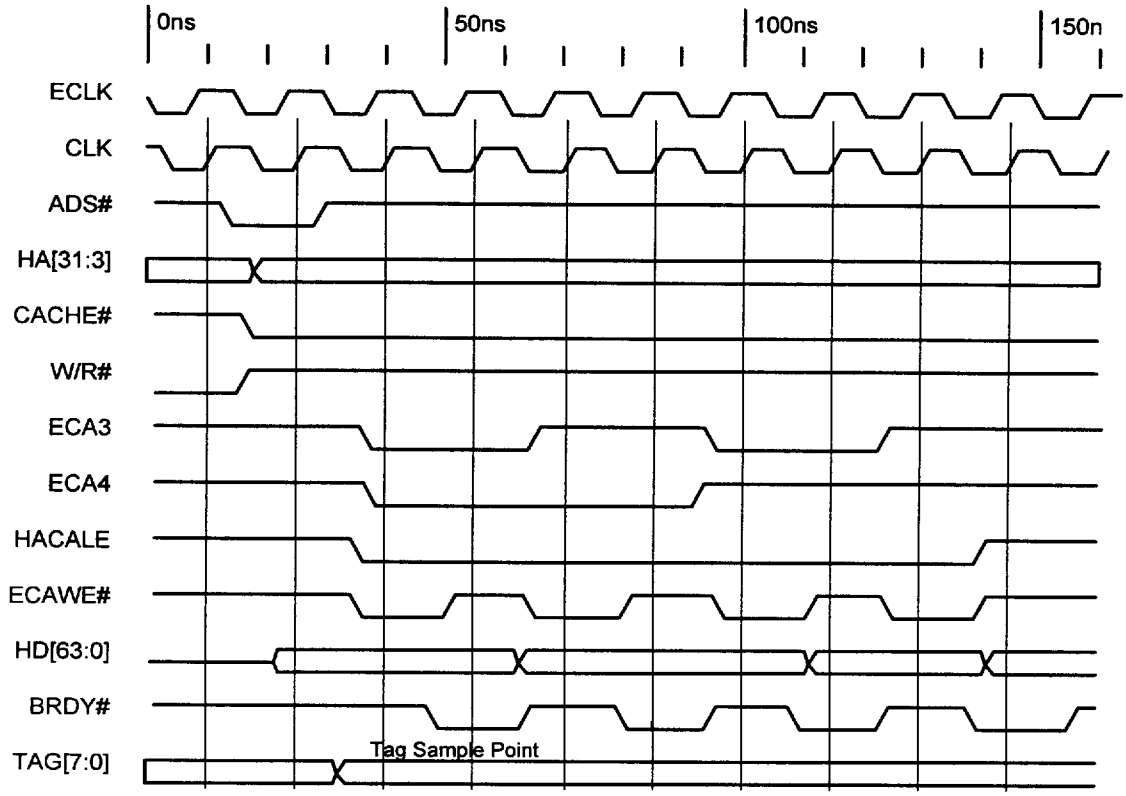
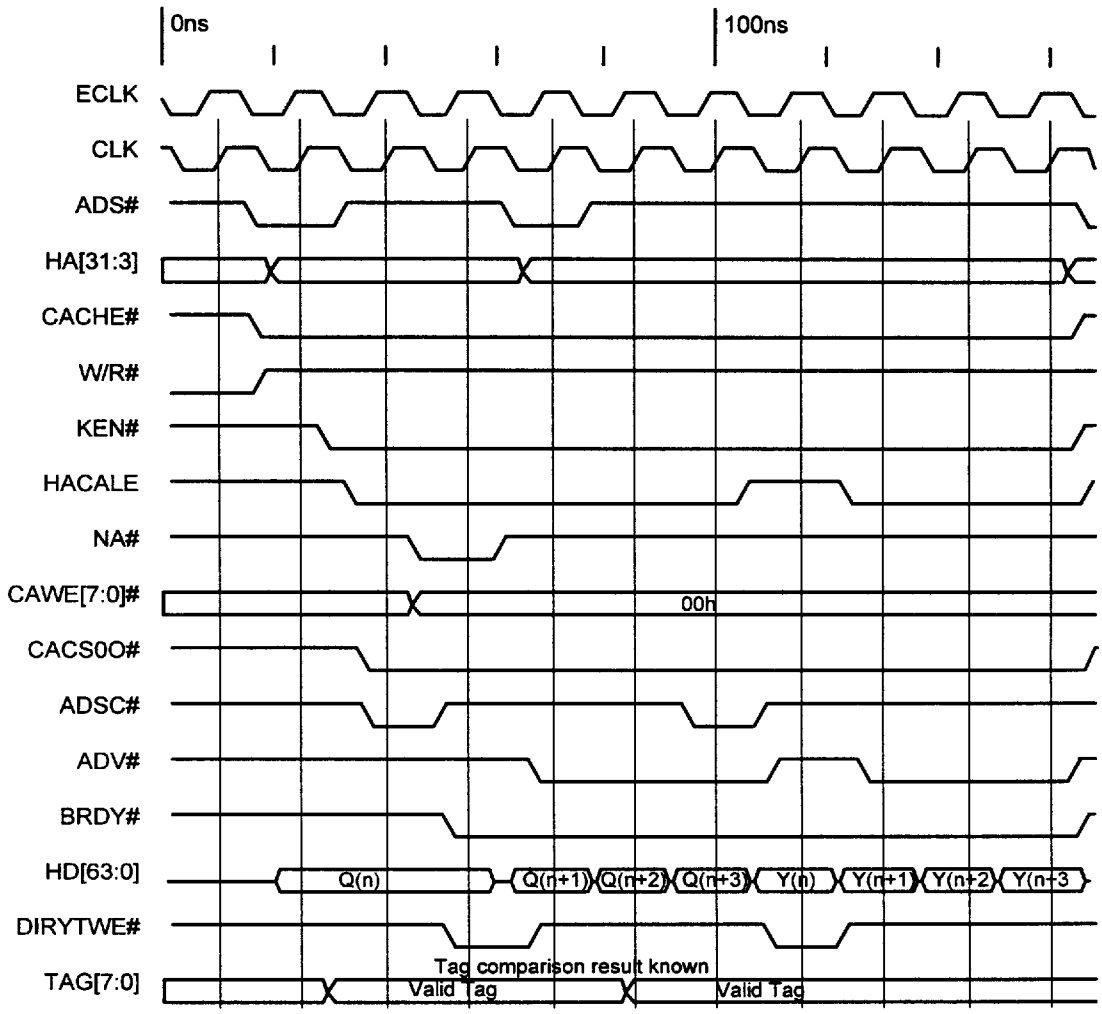


Figure 4-9 Write Hit Burst Cycle for Write-Back Mode - Sync. SRAM



4.3.3.3 L2 Cache Read Miss

Write-back Mode: There are two cache read miss cases depending on the status of the dirty bit.

CASE 1: Read miss of a "clean" cache line.

In this case, only a linefill cycle is executed. The L2 cache line that is to be replaced with a new line from the DRAM will just be overwritten. The linefill cycle is done by reading the new data from the system memory first and then the data is simultaneously written to both the CPU and the secondary cache.

The sequence for CASE 1 linefill is: System memory read ⇨ write to the L2 cache + CPU read.

The cache controller will update the tag data bits and the dirty bit in the background during the linefill cycle. At the end of T1, if the CACHE# signal from the CPU is negated, a linefill cycle will not be executed. Instead, only the eight bytes requested by the CPU will be read from the system memory. The tag and the dirty bit will not be updated.

CASE 2: Read miss with cache line dirty.

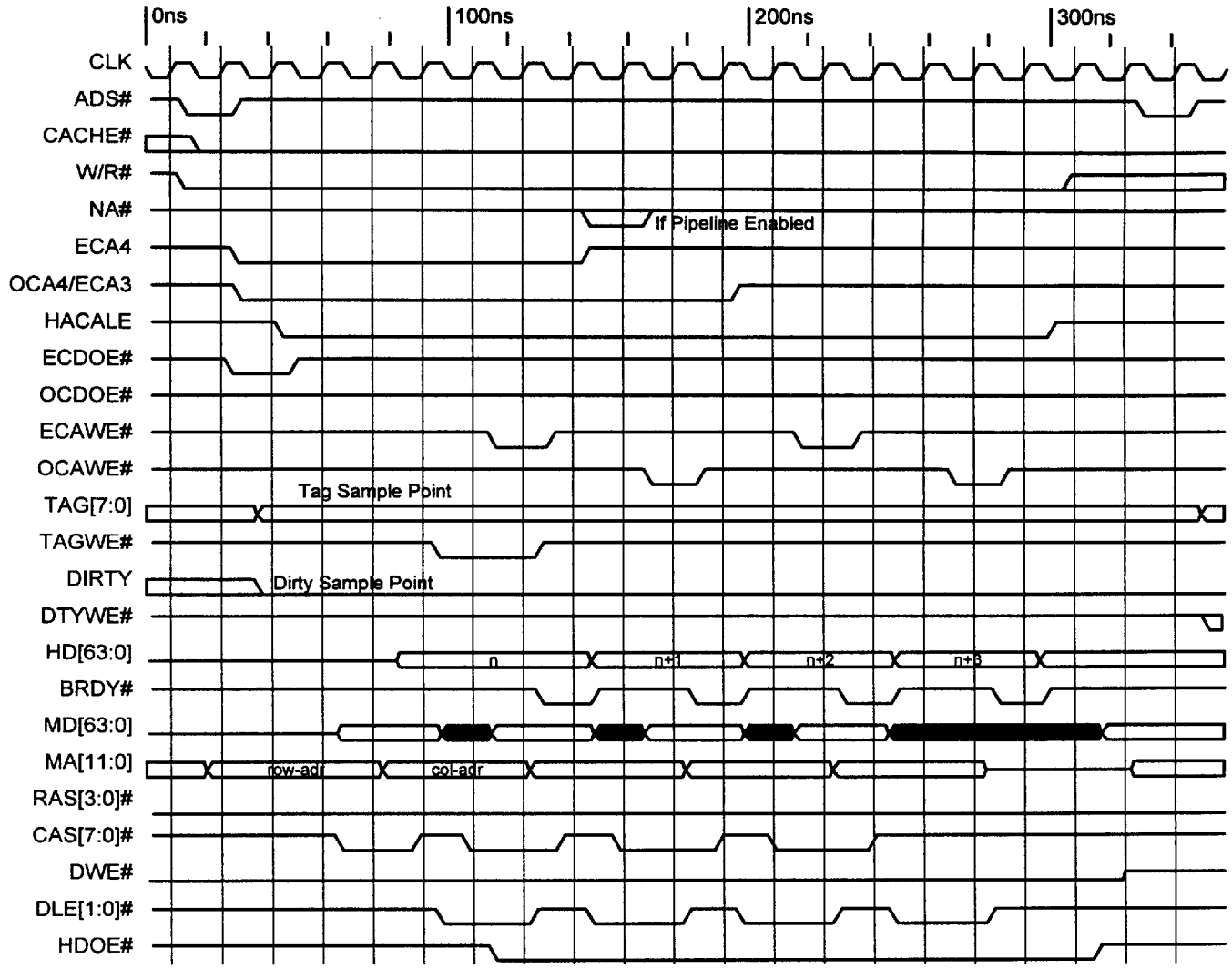
The cache line for this case has been modified and only the L1 and L2 cache have the updated copy of the data. Before this line is overwritten in the cache, the modified line must first be written to the system memory by performing a castout cycle. After the completion of the castout cycle, a linefill cycle is executed. The linefill cycle is performed by reading the new data from the system memory and then simultaneously writing this data to the CPU and the secondary cache.

The sequence for CASE 2 is: Read the dirty line from L2 cache ⇨ write to the system memory ⇨ new line read from system memory ⇨ write to the L2 cache + CPU read.

The cache controller will update the tag data bits and the dirty bit in the background during the castout cycle. If the CACHE# signal from the CPU is inactive, then the eight bytes requested by the CPU will be read from the system memory. The tag and the dirty bit are not updated.

Figures 4-10 through 4-12 show various L2 cache read miss cycles.

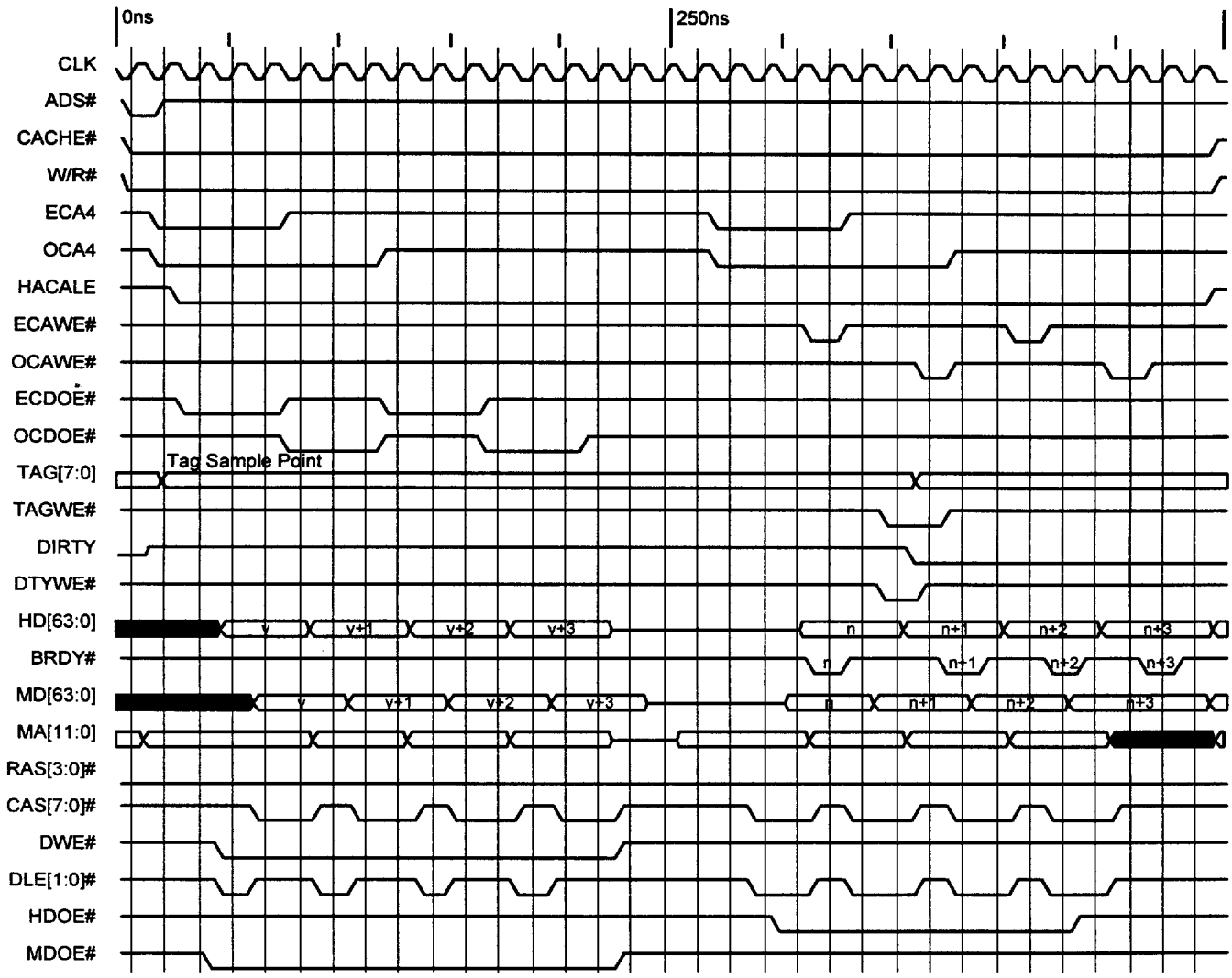
Figure 4-10 L2 Cache Read Miss Clean Burst of 8-3-3-3 (Linefill Cycle) - Async. SRAMs



Note: This diagram is also for "DRAM Read Page Hit Cycle".

Figure 4-11 L2 Cache Read Miss Clean Burst of 8-3-3-3 (Linefill Cycle) - Sync. SRAMs

Figure 4-12 L2 Cache Read Miss Dirty Cycle - Async. SRAMs



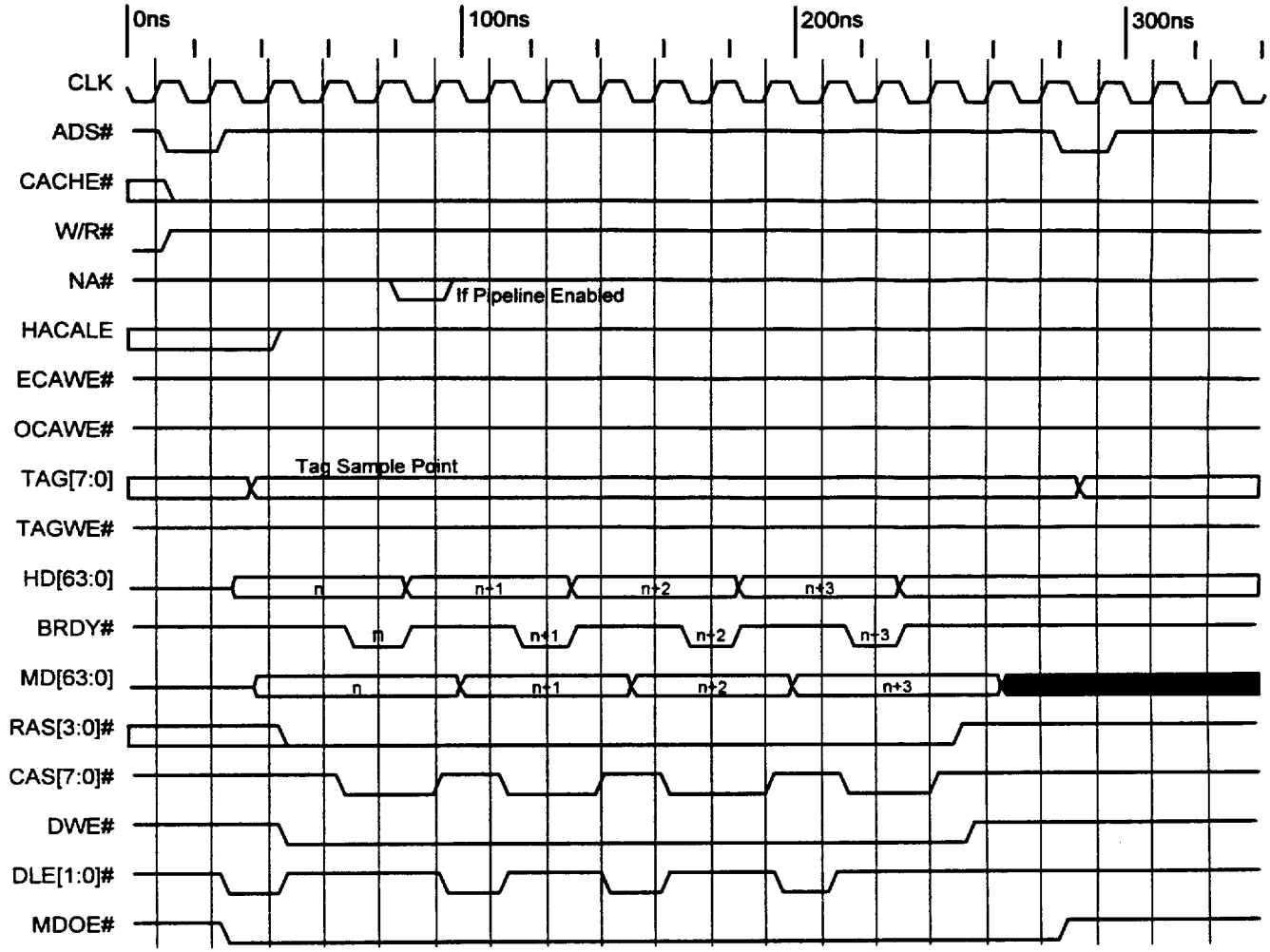
4.3.3.4 L2 Cache Write Miss

Write-back or Write-through Cases: The data is not written to the SRAM and the tag data remains unchanged. The data is written only to the system memory.

If the write buffer and DRAM posted write is enabled then is available, it is stored there and the cycles are posted writes to the DRAM. If the target is on the PCI bus, VL bus, or AT bus the cache controller will not be active.

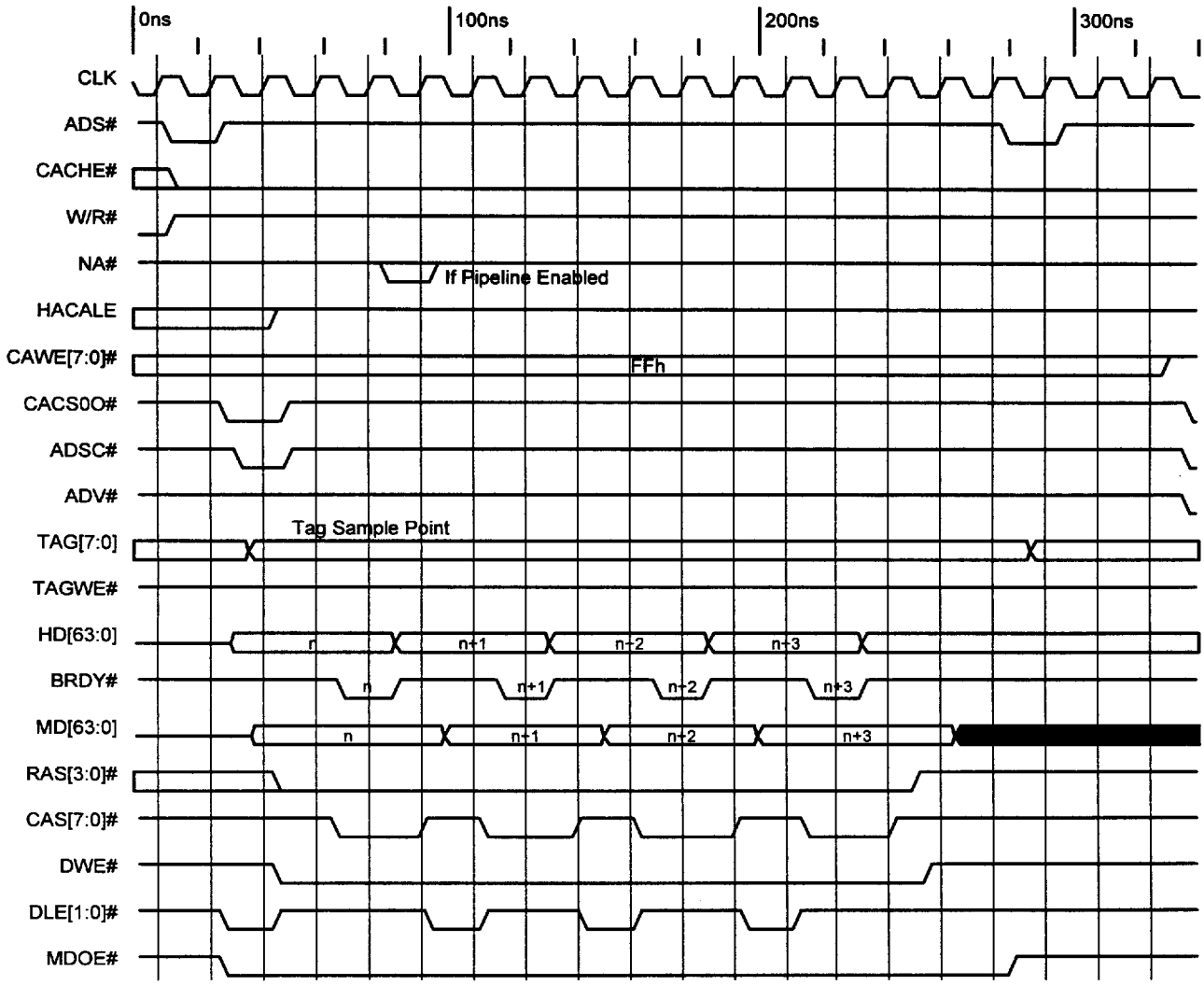
Figures 4-13 and 4-14 show L2 cache write miss cycles.

Figure 4-13 L2 Cache Write Miss Burst Cycle (4-3-3-3) - Async. SRAMs



Note: This diagram is also for "DRAM Write Page Hit Cycle".

Figure 4-14 L2 Cache Write Miss Burst Cycle - Sync. SRAMs



4.3.3.5 Adaptive Write-Back Policy

Any of the following three write policies supported by the Viper-DP Desktop Chipset can be chosen: write-back, write-through, and adaptive write-back, by programming bits [5:4] of the SYSC's I/O register at Address Offset 02h. Depending on the state of these bits and the type of DRAM cycle that would be required to complete the write hit cycle, the cache controller decides whether to update the DRAM memory, however, the cache is always updated. The adaptive write-back policy tries to reduce the disadvantages of both the write-through and the write-back schemes to a minimum. The best case cache write burst timing (for an asycnhronous cache) is 3-2-2-2, and the best case DRAM page hit write burst timing is 4-3-3-3. The adaptive write-back scheme converts a write hit cycle to a write through cycle only if the address location being written to corresponds to a page hit. In this manner this scheme incurs a 4 CLK penalty for a burst write cycle, but it saves a 13 CLK penalty (for a castout cycle) that would have occurred later due to a read miss access. There are two adaptive write-back modes.

4.3.3.5.1 Write-Through on Page Hit and RAS# Active (AWB Mode 1)

In this mode, the data is written through to the DRAM on a write hit, if the address being written to causes a page hit and the corresponding RAS# signal is active. The data will not be written through if, either the RAS# is inactive or if it's a page miss. In this case the write hit cycle completes in the same manner as in a write-back scheme.

4.3.3.5.2 Write-Through on Page Hit (AWB Mode 2)

In this mode data is written through to the DRAM on a write hit, if the address being written to causes a page hit. RAS# being active/inactive does not come into consideration when making this decision.

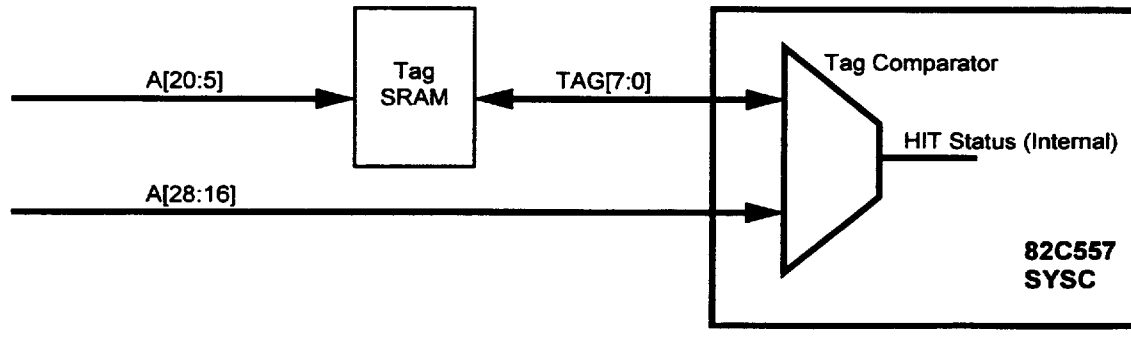
4.3.3.6 Tag Compare Table

The upper address bits used to compare for a L2 cache hit status will depend on the total L2 cache size. Table 4-2 shows the address bits from the CPU bus and the tag data bit used in the SYSC tag comparator. Figure 4-15 shows the block diagram of the L2 cache tag structure.

Table 4-2 Tag Compare Table

Tag Data	L2 Cache Size					
	64KB	128KB	256KB	512KB	1MB	2MB
TAG0	A16	A24	A24	A24	A24	A24
TAG1	A17	A17	A25	A25	A25	A25
TAG2	A18	A18	A18	A26	A26	A26
TAG3	A19	A19	A19	A19	A27	A27
TAG4	A20	A20	A20	A20	A20	A28
TAG5	A21	A21	A21	A21	A21	A21
TAG6	A22	A22	A22	A22	A22	A22
TAG7	A23	A23	A23	A23	A23	A23
Dirty Bit	Dirty	Dirty	Dirty	Dirty	Dirty	Dirty

Figure 4-15 Block Diagram of Internal Tag Comparator



4.3.3.7 Tag and Dirty RAM implementations

There are various tag/dirty RAM implementations supported by the Viper-DP Desktop Chipset.

4.3.3.7.1 Separate Tag/Dirty RAM Implementation

If a 32Kx1 part is used for the dirty RAM, there has to be a separate dirty input bit and a separate dirty output bit. In this implementation, the TAGWE# signal from the 82C557 is used to update the tag RAM and the DIRYTWE# signal from the 82C557 is used to update the dirty RAM. Only this implementation can provide a 3-2-2-2 write burst cycle at 66MHz. This scheme is shown in Figure 4-16.

4.3.3.7.2 Combined Tag/Dirty RAM Implementation

There are various ways of achieving a combined tag/dirty RAM implementation. In all these implementations, the best write burst performance obtainable is a 4-2-2-2/5-2-2-2 cycle.

A 32Kx9 SRAM can be used to implement eight tag bits and one dirty bit. In this case, the TAGWE# signal from the 82C557 is used to update both the tag and dirty information. The OE# signal of the 32Kx9 SRAM can be connected to the DIRYTWE# signal from the 82C557 or it can be tied to GND. The DIRYTI signal of the 82C557 becomes a bidirectional signal and it now serves as the dirty I/O bit. This scheme is shown in Figure 4-17.

A 32Kx8 SRAM can be used, wherein seven bits are used for the tag RAM and one bit is used for the dirty RAM. In this case, the TAGWE# signal from the 82C557 is used to update both the tag and dirty information. The OE# of the 32Kx8 SRAM can be connected to the DIRYTWE# signal from the 82C557 or it can be tied to GND. TAG[7:1] convey the tag information and TAG0 becomes the dirty I/O bit. In this scheme, the amount of main memory that can be cached reduces by half as compared to an 8-bit tag implementation. This scheme is shown in Figure 4-18.

A 32Kx8 SRAM can be used to implement the eight tag bits and another 32Kx8 SRAM used to implement the single dirty I/O bit. This scheme is identical to the 32Kx9 implementation and is shown in Figure 4-19.

Figure 4-16 Separate 32Kx8 and 32Kx1 Split Tag/Dirty RAM Implementation

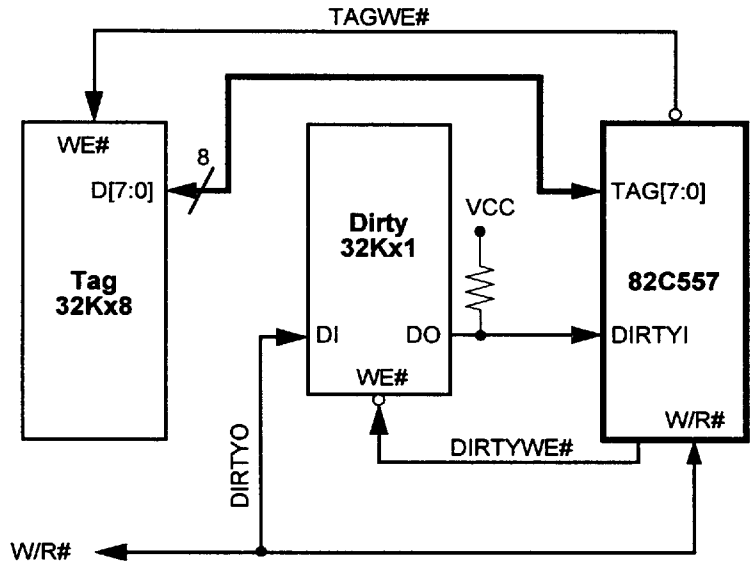


Figure 4-17 32Kx9 Combined Tag/Dirty RAM Implementation

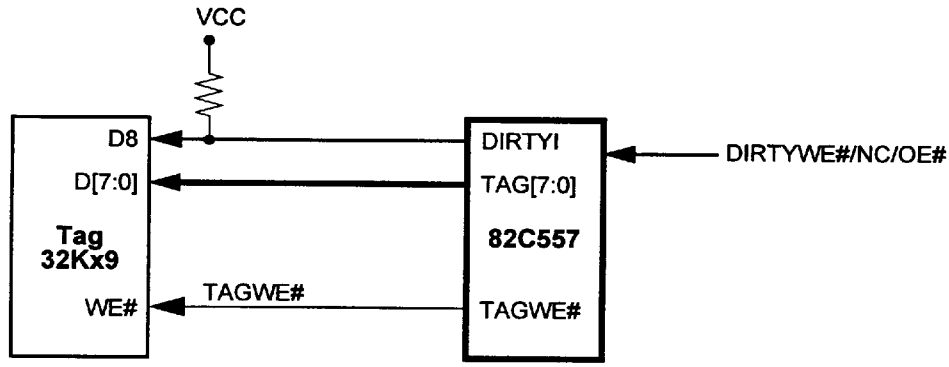


Figure 4-18 32Kx8 Combined Tag/Dirty RAM Implementation

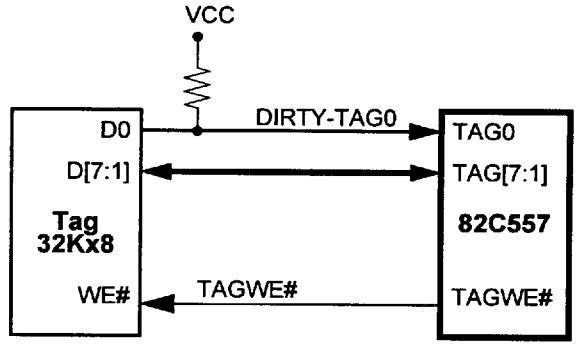
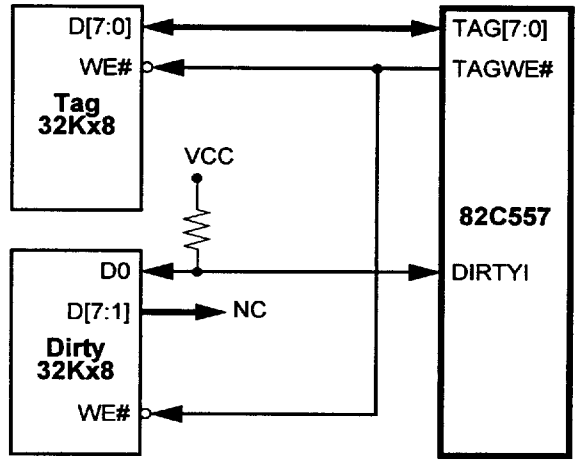


Figure 4-19 32Kx8 and 32Kx8 Combined Tag/Dirty RAM Implementation (Separate Devices)



4.3.3.8 Cache Initialization

On power-up, the tag RAM will contain random data and the L2 cache will contain no valid data. Therefore, the cache must be initialized before it is enabled.

Initializing Procedure 1: The cache is initialized by configuring the cache controller to the write-through mode. This will cause all the cache read miss cycles to fill the cache with valid data. This can be done by reading a block of system memory that is greater than or equal to the size of the cache. Once the cache is initialized, it is always valid. After this is done, the L2 cache can be set up for write-back operation by initializing the dirty bits. This is done by first enabling the cache controller to the write-back mode. Then, by reading a block of system memory that is greater than or equal to *twice* the size of the cache, the dirty bits will be cleared and the L2 cache will be valid.

Initializing Procedure 2: This procedure uses the cache controller in Test Mode 1 and Test Mode 2 as defined in the SYSC I/O registers at Indices 02h and 07h.

The upper bits of an address is written to Index 07h. The cache controller is now set to Test Mode 2. Writing a block equal to the size of the cache to the system memory will write the contents of Index 07h to the tag. The cache controller is now configured in the write-through mode and reading a block of system memory equal to the size of the cache will make the data in the cache valid. Next, by reading a block of system memory which is greater than or equal to *twice* the size of the cache, the dirty bits will be cleared and the L2 cache will be valid.

Disabling the Cache: Disabling of a write-back cache cannot be done by just turning off the cache enable bit in the SYSC register. There may still be valid data in the cache that has not been written to the system memory. Disabling write-back cache without flushing this valid data usually causes a system crash.

This situation can be avoided by first reading a cacheable memory block *twice* the size of the cache. "Twice the size" of the cache is required to make sure every location gets a read miss, which will cause a castout cycle if the cache line is dirty. The cache can then be disabled. ***Note: No writes should occur during this process.***

4.3.3.9 Write Back Cache with DMA/ISA Master/ PCI Master Operation

The L1 and the L2 cache contain the only valid copy (modified) of the data. The SYSC will execute an inquire cycle to the L1 cache for all master accesses to the system memory area. This will increase the bus master cycle time for every access to the system memory which will also decrease the bus master performance. The Viper-DP Desktop Chipset provides the option of a snoop-line comparator (snoop filtering) to increase the performance of a bus master with the L1 cache.

L1 Cache Inquire Cycle: This cycle begins with the CPU relinquishing the bus with the assertion of HLDA. On sampling HLDA active, the SYSC will assert AHOLD. The address will flow from the master to the CPU bus and the SYSC will assert EADS# for one CPU clock. If the CPU does not respond with the assertion of HITM#, the SYSC will complete the cycle from the L2 cache or the system memory. If HITM# was asserted, the SYSC will expect a castout cycle from the L1 cache and in response AHOLD is negated until the end of the castout cycle.

DMA/Master Read Cycle: Table 4-3 shows the action taken by the SYSC based on the L1 and L2 cache status for bus master reads from the system memory area. The L1 cache castout cycle will be completed in the burst order provided by the CPU and will be written to the L2 cache or the system memory based on the L2 cache status. The required bytes are then read back for the completion of the master read cycle. A read hit in the L1 cache will always invalidate the L1 cache line. Refer to Figures 4-20 and 4-21.

DMA/Master Write Cycle: Table 4-4 shows the action taken by the SYSC based on the L1 and L2 cache status for bus master writes to the system memory area. A master write to the L2 cache will always be in the write-through mode. The L1 cache castout cycle will be completed in the CPU burst sequence and the data will be written to the L2 cache or to the system memory based on the L2 cache status. Data from the master is always written to the system DRAM memory and is written to the L2 cache only if it is a L2 cache hit. Refer to Figure 4-22.

Table 4-3 DMA/Master Read Cycle Summary

DMA/Master Read Cycle		Data Source	Type of Cycle for L1 Cache	Type of Cycle for L2 Cache	Type of Cycle for DRAM
L1 Cache	L2 Cache				
Hit	Hit	L2 Cache	Invalidate	Read the Bytes Requested	No Change
hitM	Hit	L1 Cache	Castout, invalidate	Write CPU Data, Read Back the Bytes Requested	No Change
Hit	Miss	DRAM	Invalidate	No Change	Read the Bytes Requested
hitM	Miss	L1 Cache	Castout, invalidate	No Change	Write CPU Data, Read Back the Bytes Requested
Miss	Hit	L2 Cache	No Change	Read the Bytes Requested	No Change
Miss	Miss	DRAM	No Change	No Change	Read

Note: hitM - L1 cache modified

Table 4-4 DMA/Master Write Cycle Summary

DMA/Master Read Cycle		Data Destination	Type of Cycle for L1 Cache	Type of Cycle for L2 Cache	Type of Cycle for DRAM
L1 Cache	L2 Cache				
Hit	Hit	DRAM, sec	Invalidate	Write Master Data	Write Master Data
hitM	Hit	DRAM, sec	Castout, Invalidate	Write CPU Data, Write Master Data	Write Master Data
Hit	Miss	DRAM	Invalidate	No Change	Write Master Data
hitM	Miss	DRAM	Castout, Invalidate	No Change	Write CPU Data, Write Master Data
Miss	Hit	DRAM, sec	No Change	Write Master Data	Write Master Data
Miss	Miss	DRAM	No Change	No Change	Write Master Data

Figure 4-20 AT DMA/Master Read (L1 cache with non-modified line)

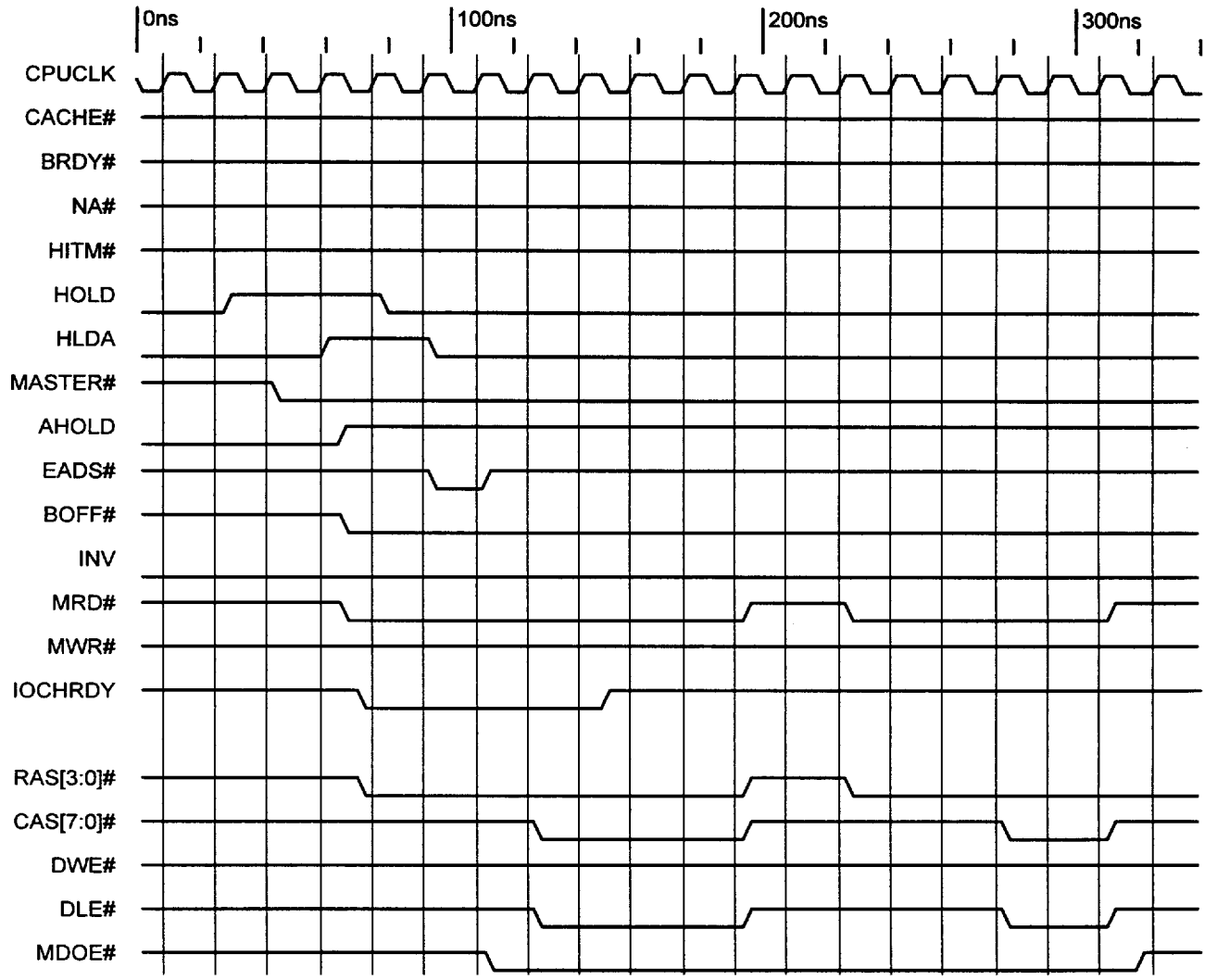


Figure 4-21 AT DMA/Master Read (L1 cache with modified line)

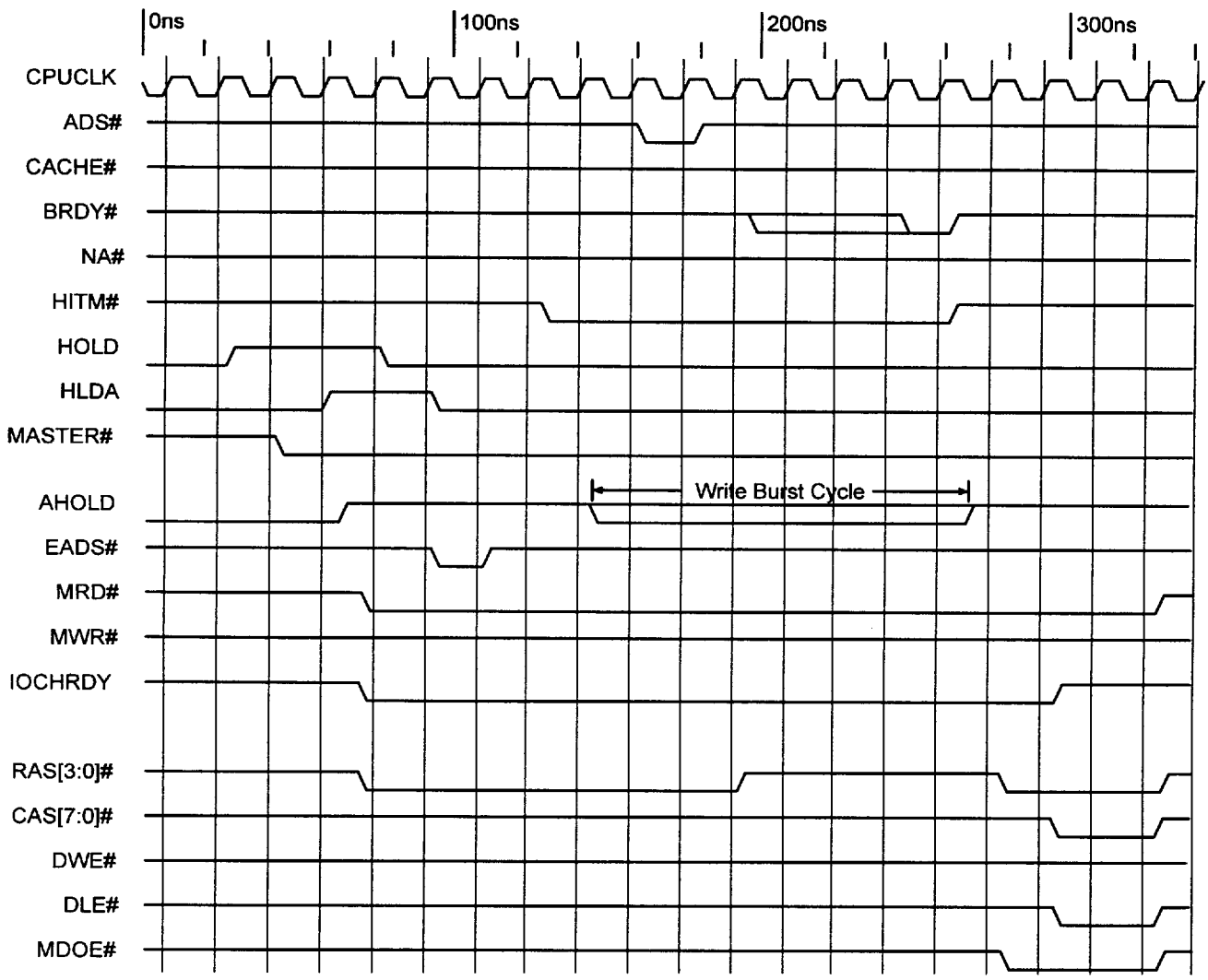
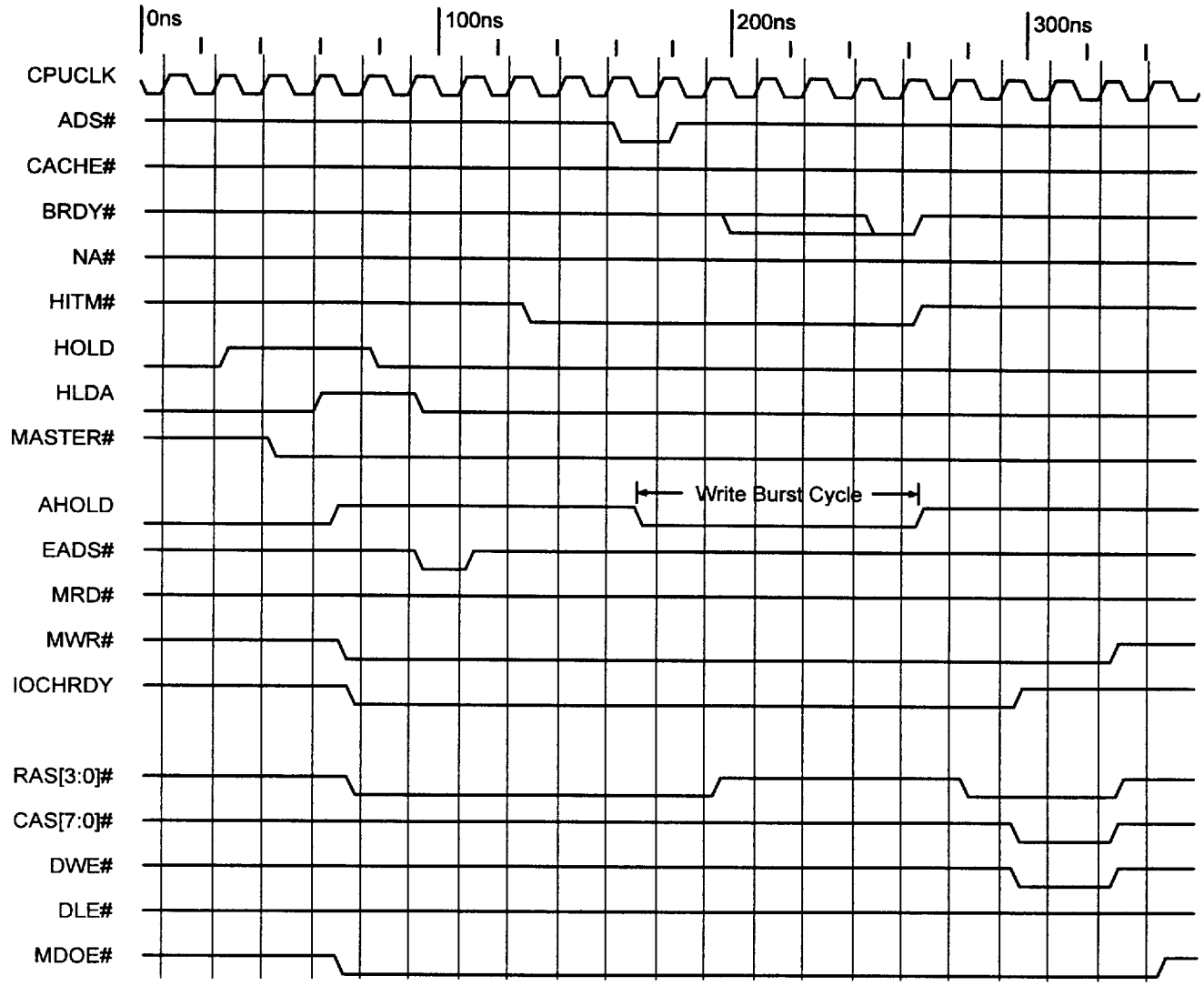


Figure 4-22 AT DMA/Master Write (L1 cache with modified line)



4.3.3.10 Cacheability and Write Protection

Both system DRAM and shadow RAM are cacheable in both the primary (L1) and/or secondary (L2) cache. Of these two areas, only the shadow RAM areas (system BIOS, video BIOS and DRAM) have the capability of being write-protected (Non-shadowed BIOS ROM areas are implicitly write-protected). Since the possibility exists that write-protected shadow RAM can be cached, there also exists the possibility that this data might be modified inside the cache and subsequently executed. To prevent this from occurring, an explicit control mechanism must be used that prevents the unexpected from happening. There are three methods for controlling write protection in the Viper-DP Desktop Chipset. (See Table 4-5 for a summary of these methods.)

METHOD 1: In this method, the write protected areas are **not** cached in the L1 or the L2 cache. This is implemented by driving KEN# high for the first word with BRDY#, which will cause the CPU to not cache the data in it's L1 cache and not do burst cycles. Data in the L2 cache is also not updated, so all reads and writes to this area will go directly to or from the system memory or to/from System BIOS/Video BIOS (if they are not shadowed). Please refer to the SYSC I/O registers at Index 05h and 06h for further information.

METHOD 2: In this method, the write protected areas can be cached in the L2 cache but not in the L1 cache. This is implemented by driving KEN# high for the first word with BRDY#, which will cause the CPU to not cache the data in the L1 cache or do a burst cycle. This data can then be stored in the L2 cache, but only subsequent read requests by the CPU are serviced (discarding all writes), thus effectively write-protecting the data in the L2 cache. Read miss cycles are serviced by first performing a linefill burst from the DRAM into the L2 cache and then performing a normal non-cacheable (and non-burst) cycle to the CPU. In this method, writes to the system memory and to the L2 cache are write protected.

METHOD 3: This method is implemented by driving EADS#/WT# high during the read cycle. Data read from write protected areas are stored in both the L1 and L2 caches. Accesses from the CPU that are L2 cache read hits are serviced in burst mode and L2 cache read miss cycles are serviced by first performing a linefill burst read to the L2 cache from the write protected area and then performing a normal burst cycle to the CPU. Write cycles from the CPU to these areas are write-through and are discarded by the SYSC's cache controller. ***However, L1 cache writes occur internally to the CPU in this mode and are therefore not write protected***. Please refer to the register at Index 08h for further information.

Table 4-5 Cacheability Methods

Method	System DRAM		System BIOS		Video BIOS		Write Enabled Shadow RAM		Write Protected Shadow RAM	
	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write
1	L1,L2	L1,L2	Single	None	Single	None	L1,L2	L1,L2	Single	None
2	L1,L2	L1,L2	L2	None	L2	None	L1,L2	L1,L2	L2	None
3	L1,L2	L1,L2	L1,L2	L1	L1,L2	L1	L1,L2	L1,L2	L1,L2	L1

Note: **L1** = accessible to primary cache, **L2** = accessible to secondary cache, **none** = no cycle performed (or discard). **int** = internal cycle to CPU, **WT** = write-through cycle, **single** = single word (non-burst) cycle, **burst** = burst cycle

4.3.4 Synchronous SRAM Support

The Viper-DP Desktop Chipset supports almost all the varieties of synchronous SRAMs available. As shown in Table 4-6, 3-1-1-1 read/write cycles are supported at 66MHz. Table 4-7 shows which signals change functionality to support a synchronous SRAM implementation.

In addition to the standard synchronous SRAMs, the Viper-DP Desktop Chipset supports pipelined synchronous SRAMs as well as the Intel standard BSRAM.

Tables 4-8 through 4-12 give additional details regarding SRAM usage in a Viper-DP Desktop Chipset implementation.

4.3.4.1 Pipelined Synchronous SRAM support

Pipelined synchronous SRAMs are cheaper than their counterpart BiCMOS synchronous SRAMs (standard synchronous SRAMs). The timing requirement of the ADV# pin assertion is different for these SRAMs, and this is enabled by setting Index 17h[1] = 1 (i.e., enabling pipelined synchronous SRAM).

In a two bank synchronous SRAM implementation, there could be data contention when switching between banks. To avoid this, Intel has proposed a BSRAM standard. This standard requires the insertion of one "idle" cycle when switching between banks. The BSRAMs that support a one clock disable and a two clock enable timing, meet this standard. The Viper-DP Desktop Chipset supports this standard. To enable this feature, Index 17h[5] should be set to 1.

4.3.4.2 SONY SONIC-2WP (Cache Module) Support

The Sony SONIC-2WP is a single chip, write-back cache subsystem that integrates 256Kbytes of cache memory, tag RAM and all other associated control logic. The integrated 256Kbyte cache is direct-mapped and it supports 3-1-1-1 burst cycles, and operates at 3.3V. If this chip is used, Index 0h[5] should be set to 1. This causes a few changes in the signal functions of the 82C557. The TAG1 and the TAG2 signals are connected to the START# signal from the Sony cache module.This signal is asserted by the Sony cache module when a CPU cycle translates to a read miss, write miss, or a write-through cycle. The assertion of this signal by the cache module causes the 82C557 to take control of the KEN# and BRDY# signals which it shares with the cache module. The TAG3 signal is connected to the BOFF# signal from the Sony cache module. The remainder of the TAG lines should be unconnected. All the other cache control signals of the 82C557 are not required and should be no connects. The ADS# input of the 82C557 should be connected to the SADS# output from the cache module. One note of caution, CPU pipelining must be disabled if using this cache module.

4.3.5 SRAM Requirements

The data RAMs are quad-word interleaved for the two bank configuration, which requires 64-bit wide SRAM. This allows systems based on the Viper-DP Desktop Chipset to perform a full 3-2-2-2 burst for reads and writes. If a single bank of DRAM is to be used, the cache controller will increase the burst wait state.

Table 4-6 SRAM Requirements

Speed	Asynchronous SRAMs		Synchronous SRAMs	
	Cycles	Operation	Cycles	Operation
50MHz	3-2-2-2	Burst Read/Write	2-1-1-1	Burst Read/Write
60MHz	3-2-2-2	Burst Read/Write	3-1-1-1	Burst Read/Write
66.6MHz	3-2-2-2	Burst Read/Write	3-1-1-1	Burst Read/Write

Table 4-7 Signal Functionality for Synchronous SRAM Implementation

Asynchronous SRAM	Synchronous SRAM
CACS[7:0]#	CAWE[7:0]#
ECAWE#	CACS0O# (Cache chip select 0)
OCAWE#	CACS1O# (Cache chip select 1)
ECA4	ADSC#
OCA4/ECA3	ADV#
ECDOE#	ECDOE#
OCDOE#	OCDOE#

Table 4-8 Data SRAM Asynchronous Configurations

Cache Size	Data SRAMs		Tag SRAMs				Cacheable Range
	Qty	Type	Qty	Tag Address Field	Qty	Tag Dirty Bit Field	
64K Bytes	8	8Kx8	1	8Kx8	1	8Kx1	16MB
128K Bytes	16	8Kx8	1	8Kx8	1	8Kx1	32MB
256K Bytes	8	32Kx8	1	8Kx8	1	8Kx1	64MB
512K Bytes	16	32Kx8	1	16Kx8	1	16Kx1	128MB
1M Bytes	8	128Kx8	1	32Kx8	1	32Kx1	128MB
2M Bytes	16	128Kx8	1	64Kx8	1	64Kx1	128MB

Table 4-9 Data SRAM (Asynchronous) and Tag SRAM Speed Requirements

Parameter	Description	33MHz	50MHz	60MHz	66MHz
Data Async. SRAMs					
tAA	Address Access Time	35ns	25ns	15ns	15ns
tOE	OE# Access Time	20ns	12ns	8ns	8ns
tWP	Write Pulse Width	30ns	25ns	14.5ns	14.5ns
Tag RAMs					
tAA	Address Access Time	35ns	20ns	15ns	12ns

Table 4-10 Data SRAM (Synchronous) Configurations

Cache Size	Qty	Size
256K Bytes	4	32Kx18
512K Bytes	4	64Kx18

Table 4-11 Data SRAM (Synchronous) Speed Requirements

Parameter	Description	33MHz	50MHz	60MHz	66MHz
tCD/	Clock Access Time	18ns	12ns (2-1-1-1)/ 12ns (3-1-1-1)	9ns	9ns

Table 4-12 Tag SRAM Speed Requirements for Synchronous SRAMs

Parameter	Description	33MHz	50MHz	60MHz	66MHz
tAA	Address Access Time	25ns	10ns (2-1-1-1)/ 20ns (3-1-1-1)	15ns	12ns

Table 4-13 SRAM Comparisons

Cycles	Async.	Sync.	Pipelined Sync.	Pipelined BSRAM	Sony Cache Module
Read hit	3-2-2-2	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1
CPU piped RH	2-2-2-2	1-1-1-1	1-1-1-1	1-1-1-1	3-1-1-1*
2 BKs piped RH	2-2-2-2	1-1-1-1**	2-1-1-1**	2-1-1-1	3-1-1-1*
Write hit	3-2-2-2	3-1-1-1	3-1-1-1	3-1-1-1	3-1-1-1
Write-back	N	N	N+4	N+4	N+BOFF
PCI read	x-2-2-2	x-2-2-2	x-3-3-3	x-3-3-3	x-2-2-2***
PCI write	x-2-2-2	x-2-2-2	x-2-2-2	x-2-2-2	x-2-2-2***
Cost	Lowest	High	Low	Low	High

* No CPU pipelined for Sony Cache Module.
** Data bus conflict for sync. SRAM, minimum data bus conflict for pipelined SRAM with 82C557 OE# control.
*** L2 needs "castout" dirty line before master access.

4.4 DRAM Controller

The Viper-DP Desktop Chipset DRAM controller uses a 64-bit wide DRAM data bus interface. It also uses the page mode technique for faster data access from the DRAMs.

Page mode is always used in the Viper-DP Desktop Chipset for CPU accesses, both for bursts and between bursts. Page mode is performed by keeping RAS active while reading or writing multiple words within a DRAM page by changing only the column address and toggling CAS with the new column address. The DRAM page size is fixed at 4KB.

Hidden refresh is used to increase the CPU bandwidth by not having to put the CPU on hold every 15µs to refresh the DRAM. The DRAM can be refreshed in the background while the CPU is accessing the internal cache.

4.4.1 Programming the DRAM Parameters

There are various parameters that can be obtained in the DRAM state machine - number of banks, DRAM configurations, timing parameters and drive strengths.

4.4.1.1 Number of DRAM banks

The Viper-DP Desktop Chipset supports up to six banks of DRAM. The default condition is four banks of DRAM supporting up to 512Mbytes of system memory. MA11 is multiplexed with RAS4# and DIRYTWE# is multiplexed with RAS5#.

If MA11 is used as RAS4#, then the maximum memory size supported decreases to 192Mbytes.

If DIRYTWE# is used as RAS5#, then there are several scenarios:

If a separate tag/dirty RAM implementation is used, then the L2 cache write-back functionality is lost. If a combined tag/dirty RAM implementation is used, then the L2 cache can still be used in the write-back mode.

If RAS5# is used and MA11 is not used as RAS4#, then the maximum amount of memory supported is 512Mbytes, with not more than 3 banks populated with 128Mbytes.

If both RAS4# and RAS5# are being used (i.e six banks of DRAM), then the maximum amount of memory supported is 192Mbytes.

4.4.1.2 DRAM Configurations

The Viper-DP Desktop Chipset provides the maximum flexibility for DRAM configurations, if Index 13h[7] = 1. Asymmetric as well as symmetric DRAM sizes are supported and there are no restriction on which banks need to be populated as long as each logical bank has a 64-bit data path (*Asymmetric support is limited to DRAMs which have their number of row address bits = number of column address bits + 1*). To maintain backward compatibility with OPTi's 82C546/82C547 (Python) Chipset, the fixed DRAM configurations of that chipset are also supported. If Index 13h[7] = 0, then only the fixed DRAM configurations listed out in Table 5-36 (see Register Section 5.0) are supported.

4.4.1.3 Timing Parameters

The timing constraints to achieve optimum performance at 66MHz are met without making the system design overly critical. Timing variations that are required for different system speeds are handled by a selection of timing modes that vary the wait states used. Table 4-14 summarizes these timing modes.

Table 4-14 DRAM Programmable Control

DRAM Timing Being Controlled	Variation in CLK
RAS address hold time	1 to 2
CAS pulse width for reads	2 to 3
CAS pulse width for writes	2 to 3
Address setup time to CAS for write page hit	1 to 2
CAS precharge time	1 to 2
RAS precharge time	3 to 6
RAS pulse width for refresh	4 to 7

4.4.1.4 Drive Strengths

Programmable current drive for the MA[11:0], RAS[5:0]# and the DWE# lines is provided. If Index 18h[4] = 0, then the current drive on these lines is 4mA. In this case, two F244 buffers will be required to drive each pair of DRAM banks. If Index 18h[4] = 1, then the current drive on these lines is increased to 16mA and it should be possible to drive the first pair of DRAM banks without any buffers.

4.4.2 DRAM Cycles

The fastest possible burst read is 8-3-3-3 which means the first quad-word is received in eight clocks and the next three quad-words are received after three clocks each. For a cache based system, it would mean the bursting to the cache and CPU for read miss cycles or write miss cycles. Table 4-15 summarizes the DRAM timing modes for read and write cycles respectively.

4.4.2.1 DRAM Read Cycle

The DRAM read cycle begins with the DRAM controller detecting a page hit or a page miss cycle at the end of the first T2. Based on the status of the current open page and the active RASx#, a page hit, a page miss with RAS inactive, or a page miss with RAS active cycle is executed.

Page Miss with RAS High Cycle: The row address is generated from the CPU address bus. Table 4-16 gives the row/column address mux map. After RASx# goes active, the row address is changed on the next clock edge (programmable to be two CLKs) to the column address. The CASx# will be active two CLKs after the column address is generated. (Refer to Figure 4-23.)

Page Miss with RASx# Low Cycle: RAS is first precharged for the programmed number of CLKs and then driven active, after which it will be the same as a page miss with RAS high cycle.

Page Hit Cycle: The 82C557 generates the column address from the CPU address bus and CASx# is driven active for two clocks. Data flow from the CPU data bus to the memory data bus and vice versa is controlled by the DBCOE#[1:0], MDOE#, and HDOE# signals from the SYSC to the DBC. Data from the DRAM is latched by the DBC on the rising edge of each DLE (for CPU reads from DRAM, the DLE signals are identical to the CAS signal). The latched data is valid on the CPU data bus until the next rising edge of CASx#. During this time, the next read is started, CASx# signals are precharged for one or two clocks (programmable), and the next data from the DRAM is accessed and latched. The DBC latches the data from the DRAM and holds the data for the CPU while the DRAM controller begins the read for the next word in the burst cycle. The burst read from the DRAM is in effect pipelined into the CPU data bus by the Viper-DP Desktop Chipset. This scheme reduces the constraints on the board layout so that routing for the CPU data bus, MD data bus, and CASx# signal lines are less critical and performance can be maintained.

Page Hit Cycle (Extended): Wait states can be added if slower DRAMs are used. In this mode, data from the DRAM is latched by the DBC at the end of each CAS cycle similar to the default mode. The only difference between the two modes is that the CAS low time on reads is increased by one T-state. This eases up on the page mode cycle time and CAS access time parameters.

The DRAM read cycle uses a CAS signal that is active for multiples of T-state boundaries rather than half T-state boundaries. This allows additional address decode setup time and MA bus setup time at the start of the cycle, making the fastest burst cycle 8-3-3-3.

4.4.2.2 DRAM Write Cycle

Posted write to the DRAM improves the write cycle timing relative to the CPU and allows the Viper-DP Desktop Chipset to perform an independent write burst cycle to DRAM without holding the CPU. The Viper-DP Desktop Chipset maintains a one quad-word deep data buffer for DRAM writes so that the CPU write cycle is completed without waiting for the external DRAM cycle. For a burst write cycle, the leadoff cycle time is reduced to four clocks even if the cycle is a non-page hit cycle. For a page hit cycle, the burst write can be completed in 4-3-3-3 with posted write enabled. The posted write buffer in the DBC is controlled by the DLE[1:0]# signals from the SYSC. Effectively, the rising edge of these signals will latch the high 32-bit and the low 32-bit new data respectively, from the CPU bus to the posted write buffer.

Single level posted write cycles are employed to achieve a 4-3-3-3 burst at 66MHz. The data from the CPU is latched in the DBC's write buffer until CAS goes active one T-state after the first T2 (on a page hit). This provides a fast write mechanism and two wait state writes are maintained for the leadoff cycle within a page (even at 66MHz). The CAS pulse width can be extended by one more T-state to ease the timing constraints on the CAS pulse width requirement for speeds above 66MHz.

Table 4-15 DRAM Timing Mode Summary

CPU Speeds (MHz)	Can be Used w/o Cache?	Page Hit	Page Miss RAS High	Page Miss RAS Active	CPU Pipeline Reduces Leadoff Cycle by:	Burst Cycle	DRAM Speed (ns)
Read Cycle							
33 to 66	yes	8 cycles	11 cycles	11+ precharge	5 clocks	3-3-3	70
80	yes	9 cycles	13 cycles	13+ precharge	5 clocks	5-5-5	70

CPU Speeds (MHz)	Can be Used w/o Cache?	Burst Page Hit	Page Miss Burst RAS High	Page Miss Burst RAS Active	CPU Pipeline Reduces Leadoff Cycle by:	DRAM Speed (ns)
Write Cycle						
33 to 66	yes	4-3-3-3	4-7-3-3	4-(7+pre)-3-3	1 clock	70
80	yes	6-5-5-5	6-9-5-5	6-(9+pre)-5-5	1 clock	70

Figure 4-23 DRAM Read Page Miss with RAS Active Read Cycle

Note: For RAS inactive cycle, clocks 4 through 7 will not exist.

Figure 4-24 DRAM Page Miss with RAS Active Write Cycle

Note: For RAS inactive cycle, clocks 4 through 7 will not exist.

4.4.2.3 DRAM Parity Generation/Detection Logic

During local DRAM write cycles, the DBC generates a parity bit for each byte written by the processor. Parity bits are stored in the local DRAM along with each data byte. During a DRAM read, the parity bit is checked for each data byte. If the logic detects incorrect parity, the DBC will assert the MPERR# signal to the IPC. If index 08h[4] = 1 in the SYSC I/O register space (i.e parity has been enabled), then the SYSC keeps the PEN# signal to the IPC asserted. When the IPC senses that MPERR# has been asserted by the DBC and if PEN# is also asserted, then it will assert a NMI interrupt to the CPU.

4.4.2.4 DRAM Refresh Logic

The DBC supports both normal and hidden refresh. Normal refresh refers to the classical refresh implementation which places the CPU on "hold" while a refresh cycle takes place to both the local DRAM and any AT bus memory. This is the default condition at power-up. Hidden refresh is performed independently of the CPU and does not suffer from the performance restriction of losing processor bandwidth by forcing the CPU into its hold state.

Hidden refresh delivers higher system performance and is recommended over normal refresh. As long as the CPU does not try to access local memory or the AT bus during a hidden refresh cycle, refresh will be transparent to the CPU. The CPU can continue to execute from its internal and secondary caches as well as execute internal instructions during hidden refresh without any loss in performance due to refresh arbitration. If a local memory or AT bus access is required during a hidden refresh cycle, wait states will be added to the CPU cycle until the resource becomes available. Hidden refresh also separates refreshing of the AT bus and local DRAM.

The DRAM controller arbitrates between CPU DRAM accesses and DRAM refresh cycles, while the AT bus controller arbitrates between CPU accesses to the AT bus, DMA and AT refresh. The AT bus controller (the IPC) asserts the RFSH# and MEMR# commands and outputs the refresh address during AT bus refresh cycles.

The SYSC implements refresh cycles to the local DRAM using CAS-before-RAS timing. The CAS-before-RAS refresh uses less power than RAS-only refresh which is important when dealing with large memory arrays. CAS-before-RAS refresh is used for both normal and hidden refresh to DRAM memory.

The periodic refresh request signal output, from the IPC that occurs every 15µs, originates from the counter/timer of the integrated 82C206. Requests for refresh cycles are generated by two sources: the counter/timer of the integrated 82C206 or 16-bit ISA masters that activate refresh when they have bus ownership. These ISA masters must supply refresh cycles because the refresh controller cannot preempt the bus master to perform the necessary refresh cycles. 16-bit ISA masters that hold the bus longer than 15µs must supply refresh cycles.

4.4.3 DRAM Address Muxing

Table 4-16 shows the DRAM address (MA) muxing. Note that the column address is the same for all configurations since this is the speed path. A3 and A4 must go through an internal burst counter, for the generation of the MA address to the DRAMs. The table shows MA line to address bit mapping for each DRAM size configuration.

4.4.4 DRAM DMA/Master Cycles

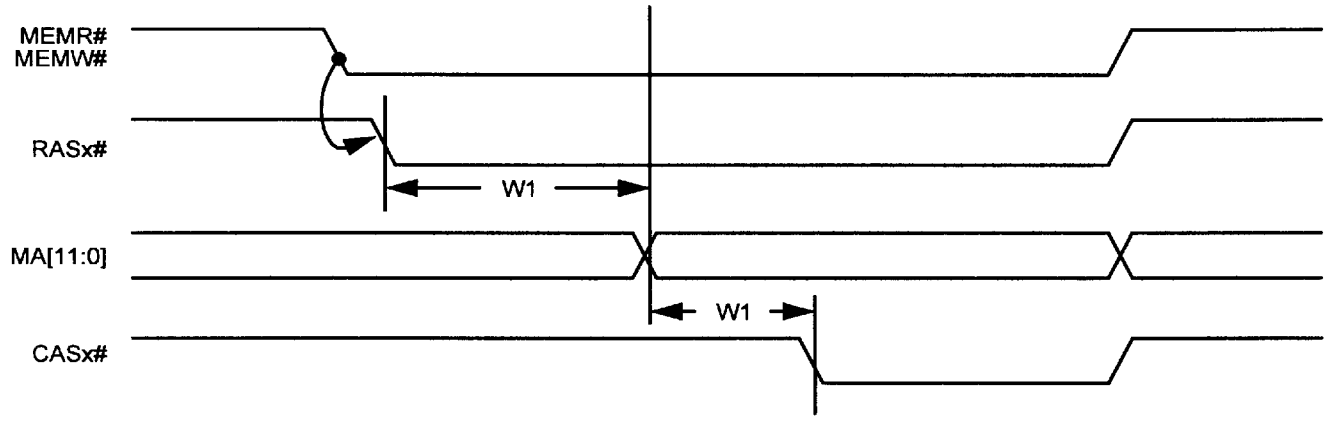
For DMA and master cycles, the DRAM controller operates such that the MEMR# and MEMW# signals generate RASx# synchronously. The generation of the DRAM column address is then synchronized with LCLK. The synchronization can be programmed to be 0.5 to 1.5 LCLKs and 1 to 2 LCLKs. The generation of CASx# is always one LCLK after the generation of the column address. The cycles can thus be completed without adding wait states. For cases when the CPU write-back cache is enabled, wait states need to be added to the DMA/master cycles. This is because the CPU can request a primary cache castout (always a burst write to the DRAMs) and only after the castout is completed can the requested data from the DRAM be fetched.

Note: ISA masters which ignore IOCHRDY may not work when CPU write-back is enabled.

Table 4-16 DRAM Row/Column MA to Address Bit Map

	256KB		512KB		1MB		2MB		4MB		8MB		16MB	
Addr.	Col	Row	Col	Row	Col	Row	Col	Row	Col	Row	Col	Row	Col	Row
MA0	A3	A12	A3	A12	A3	A12	A3	A12	A3	A12	A3	A12	A3	A12
MA1	A4	A13	A4	A13	A4	A13	A4	A13	A4	A13	A4	A13	A4	A13
MA2	A5	A14	A5	A14	A5	A14	A5	A14	A5	A14	A5	A14	A5	A14
MA3	A6	A15	A6	A15	A6	A15	A6	A15	A6	A15	A6	A15	A6	A15
MA4	A7	A16	A7	A16	A7	A16	A7	A16	A7	A16	A7	A16	A7	A16
MA5	A8	A17	A8	A17	A8	A17	A8	A17	A8	A17	A8	A17	A8	A17
MA6	A9	A18	A9	A18	A9	A18	A9	A18	A9	A18	A9	A18	A9	A18
MA7	A10	A19	A10	A19	A10	A19	A10	A19	A10	A19	A10	A19	A10	A19
MA8	A11	A20	A11	A20	A11	A20	A11	A20	A11	A20	A11	A20	A11	A20
MA9	-	-	-	A21	A22	A21	A22	A21	A22	A21	A22	A21	A22	A21
MA10	-	-	-	-	-	-	-	A23	A24	A23	A24	A23	A24	A23
MA11	-	-	-	-	-	-	-	-	-	-	-	A25	A26	A25

Figure 4-25 AT Master Synchronization



W1 = 0.5 to 1.5 and 1 to 2 LCLKs
W2 = 1 LCLK

4.5 PCI Bus Interface

The Viper-DP Desktop Chipset supports up to three PCI bus masters. Both synchronous and asynchronous modes of operation of the PCI bus, with respect to the CPU, are supported. The Viper-DP Desktop Chipset supports a 32-bit PCI implementation and supports PCI bus operating frequencies up to 33MHz. The PCI local bus controller is present in the 82C557 (SYSC), and the PCI data bus buffering is done within the 82C558 (IPC). The IPC also functions as the PCI to ISA expansion bridge and performs the required data path conversion between the 32-bit PCI bus and the 8/16-bit ISA bus.

4.5.1 PCI Master Cycles

A PCI master is always allowed to access the system memory and system I/O spaces. Accesses to the AT bus and the VESA local bus space can be individually disabled/enabled by programming the appropriate bits in the IPC register at Address Offset 43h-42h.

4.5.1.1 System Memory Access

The PCI master asserts FRAME# and puts out the address on the AD[31:0] bus. The SYSC decodes that address and asserts LMEM# to the IPC if the access is to the system memory area. The IPC then provides the data path to the PCI master to access system memory. If the access is to the system memory space, then the SYSC acts as the PCI slave and it generates the appropriate control signals to snoop the L1 cache for every access, or for every access to a new line (if the line comparator is enabled). The DBC performs the data steering and latching based on the control information received from the SYSC, over the DBCOE[1:0]#, MDOE#, HDOE#, and the DLE[1:0]# lines.

Table 4-3 and Table 4-4 describe the sequence of events that take place during a master read/write cycle from/to system memory. Listed below is the data flow path for all such accesses by a PCI master. Table 3-9 describes the state of the control signals from the SYSC to the DBC for all cycles.

For a low order DRAM read, the DRAM puts out the data on the MD[31:0]# lines. The DBC latches the data, inverts it and puts it out onto the MD[63:32] lines and drives it out to the IPC. The IPC then latches the data and puts it out on the AD[31:0] lines for the PCI master. For a low order DRAM write, the PCI master puts out the data on the AD[31:0] lines and the IPC latches the data and puts it out on the MD[63:32] lines. The DBC then latches the data, inverts it and puts it out onto the MD[31:0]# lines and presents it to the DRAM. A low order cache read would cause the cache to put out the data on the HD[31:0] lines. The DBC latches the data onto the MD[63:32] lines and drives it out to the IPC. The IPC then latches the data and puts it out on the AD[31:0] lines for the PCI master. For a low order cache write, the PCI master puts out the data on the AD[31:0] lines and the IPC latches the data and puts it out on the MD[63:32] lines. The DBC latches this data and puts it out on the HD[31:0] lines for the cache.

For a high order DRAM read, the DRAM puts out the data on the MD[63:32] lines. In this case, there is a direct path from the DRAM to the IPC and the DBC does not have to perform any latching or steering of data. The IPC latches the data available on the MD[63:32] lines and puts it out on the AD[31:0] lines for the PCI master. For a high order DRAM write, the PCI master puts out the data on the AD[31:0] lines. The IPC latches the data and this puts it out on the MD[63:32] lines. The DBC does not to have perform any steering or latching and the data is written directly to the DRAM. A high order cache read would cause the cache to put out the data on the HD[63:32] lines. The DBC latches the data onto the MD[63:32] lines and drives it out to the IPC. The IPC then latches the data and puts it out on the AD[31:0] lines for the PCI master. For a high order cache write, the PCI master puts out the data on the AD[31:0] lines and the IPC latches the data and puts it out on the MD[63:32] lines. The DBC latches this data and puts it out on the HD[63:32] lines for the cache.

4.5.1.2 Non-Local Memory Access

The PCI master asserts FRAME# and puts out the address on the AD[31:0] bus. If the access is not to the system memory area then the SYSC does not assert LMEM# to the IPC. The IPC then translates all PCI cycles to the VL bus and generates the local bus signals one LCLK after the assertion of FRAME#. The LDEV# signal is sampled at the end of the next LCLK and the IPC asserts DEVSEL# if the LDEV# was sampled asserted. For a read access from the VL bus, the local bus device puts out the data on the MD[63:32] lines. This data is latched by the IPC and put out on the AD[31:0] lines for the PCI master. For a write access to the VL bus, the PCI master puts out the data on the AD[31:0] lines. This data is latched by the IPC and put out on the MD[63:32] lines for the VL bus.

All other PCI slaves have up to three PCI CLKs after the start of the PCI cycle to assert DEVSEL#. All read/write access from/to PCI slaves is done directly over the AD[31:0] lines.

If neither a PCI slave nor a local bus device responds within three PCI CLKs after the start of the cycle, then the IPC starts an ISA cycle. For a read access from the ISA bus, the ISA device puts out the data on the SD[15:0] or the SD[7:0] lines depending on whether it is a 16- or 8-bit slave. The IPC latches this data and then performs the appropriate data bus conversions and steering (based on the IOCS16#, MEMCS16#, SBHE# signals) and puts the data out on the AD[31:0] lines. For a write access to the ISA bus, the PCI master puts out the data on the AD[31:0] lines. The IPC latches this data and then performs the appropriate data bus conversions and steering (based on the IOCS16#, MEMCS16#, SBHE# signals) and puts out the data on the SD[15:0] or the SD[7:0] lines depending on whether it is a 16- or 8-bit slave.

4.5.1.3 PCI Master Pre-snoop

Pre-snooping is a technique with the aid of which a PCI master can sustain bursting to the local memory till a 4K page boundary is reached. If pre-snooping is enabled, then on the first TRDY# of the PCI master cycle, the state machine within the 82C557 increments the HA[12:5] address lines by 1 and asserts EADS# to the CPU after that. By this time, the earlier cache address would have been latched by HACALE. If the CPU responds with a HITM#, then the current PCI master cycle will be terminated at the line boundary to allow the write-back cycle to occur. Enabling pre-snooping allows the Viper-DP Desktop Chipset to continue bursting past a line boundary.

4.5.2 PCI Slave Cycles

4.5.2.1 CPU Master Cycles

Any CPU cycle that is not an access to the system memory area, the SYSC translates that cycle to a PCI cycle and asserts FRAME# on the PCI bus. All PCI slaves have up to three PCI CLKs after the start of the cycle within which to assert DEVSEL#. The data flow path would be similar to the ones described in the previous section.

4.5.2.2 PCI Byte/Word Merge

This feature, if turned on, allows successive 8-/16-bit writes from the CPU to a PCI slave, to be merged into a 32-bit entity and then sent out to the PCI slave. This enhances PCI video performance by a substantial margin. Byte/word merge is controlled by the MDLE# and the IRDY# signal from the 82C557. The number of MDLE# pulses sent out by the 82C557 before it asserts IRDY# determines how much data was sent out with each pulse. There is one additional control provided (in Index 0h[2:1]) for the byte/word merge implementation. This setting determines the maximum time difference within which consecutive PCI bytes/words could be merged.

To enable byte/word merge and to obtain the maximum performance benefit the following should be done:

- Address Offsets 4Eh[3] and 4Eh[1] in the IPC should be set to 1.
- Index 17h[2] of the SYSC system control register space should be set to 1.
- Index 0h[4:3] of the SYSC system control register space should be set to 11.

4.5.2.3 ISA Master Cycles

If the ISA master cycle is not a system memory access, then the IPC becomes the initiator and commences a PCI cycle. The data flow path for an ISA master to a PCI slave access is between the SD[15:0]/SD[7:0] lines and the AD[31:0] lines. The IPC handles all the data bus conversion and steering logic.

4.6 VESA VL Bus Interface

The Viper-DP Desktop Chipset supports VL bus slaves only. The VL bus always operates at the PCI bus operating frequency. All the control and status signals are generated by the IPC and the data path is controlled by buffers in the DBC. The SYSC also does the data bus conversion to interface the 32-bit VL bus to the 64-bit CPU bus. The Viper-DP Desktop Chipset supports VL bus speeds up to 33MHz, independent of the CPU speed. It assumes that an access outside the system memory area is either a PCI cycle, a VL slave cycle, or an AT cycle. If the cycle is not a system memory cycle, then the SYSC generates a PCI cycle and the IPC a VL cycle. If the Viper-DP Desktop Chipset has been configured to support VL bus slaves, then the IPC generates LADS# and other VL bus status signals one LCLK after FRAME# has been asserted. The VL slave can claim such an access by asserting LDEV#, which is sampled at the end of the next LCLK. If LDEV# is active when sampled, the IPC asserts DEVSEL# to the SYSC and it will not execute an AT cycle but will instead wait for the VL slave to generate LRDY#. On receiving an active LDEV#, the IPC asserts DEVSEL# to the SYSC. After LRDY# is sampled active, the Viper-DP Desktop Chipset will terminate the cycle of the current active bus master by returning BRDY# or IOCHRDY.

Figure 4-26 PCI Master Read from Local Memory

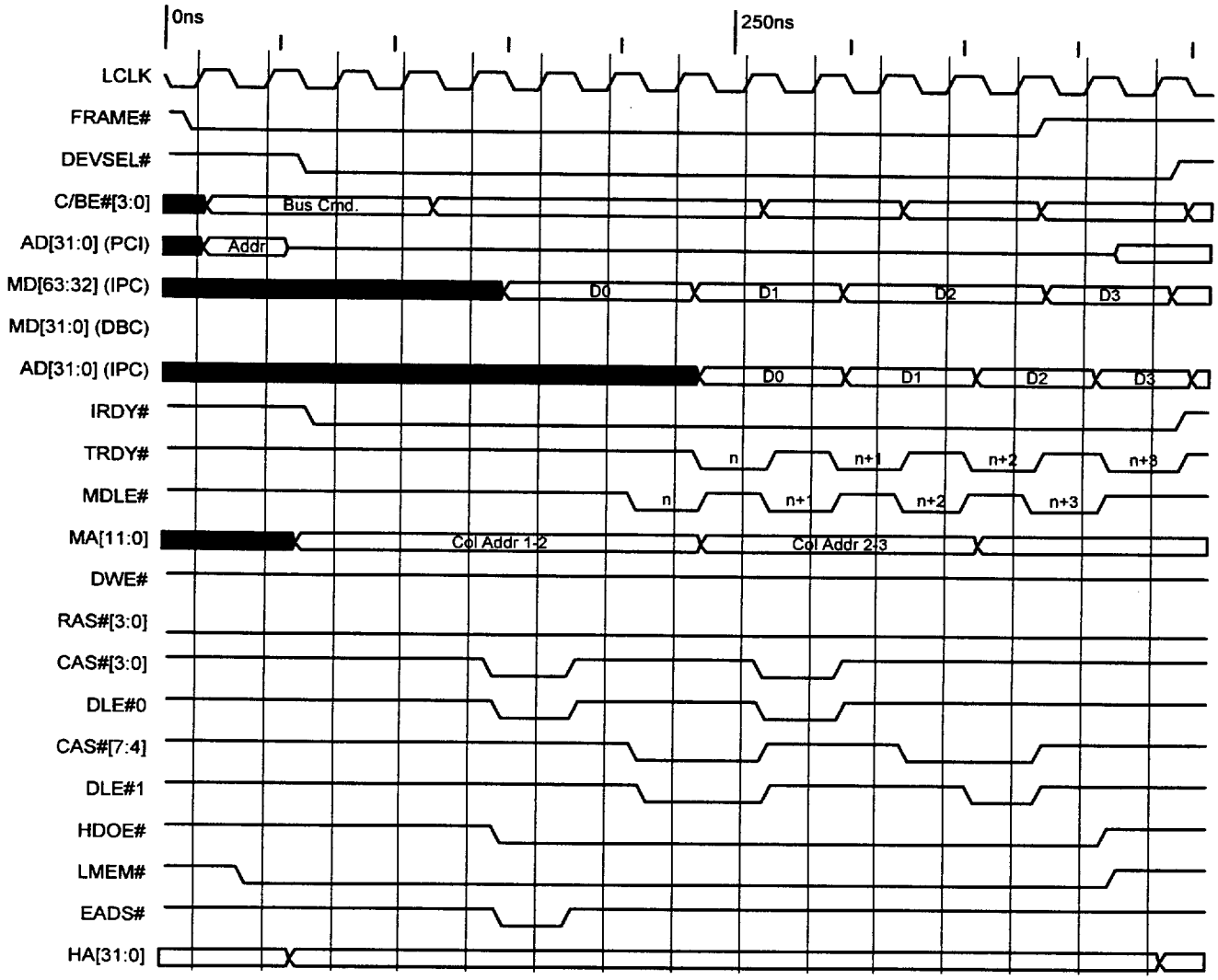


Figure 4-27 CPU Read from PCI, 32 Bits

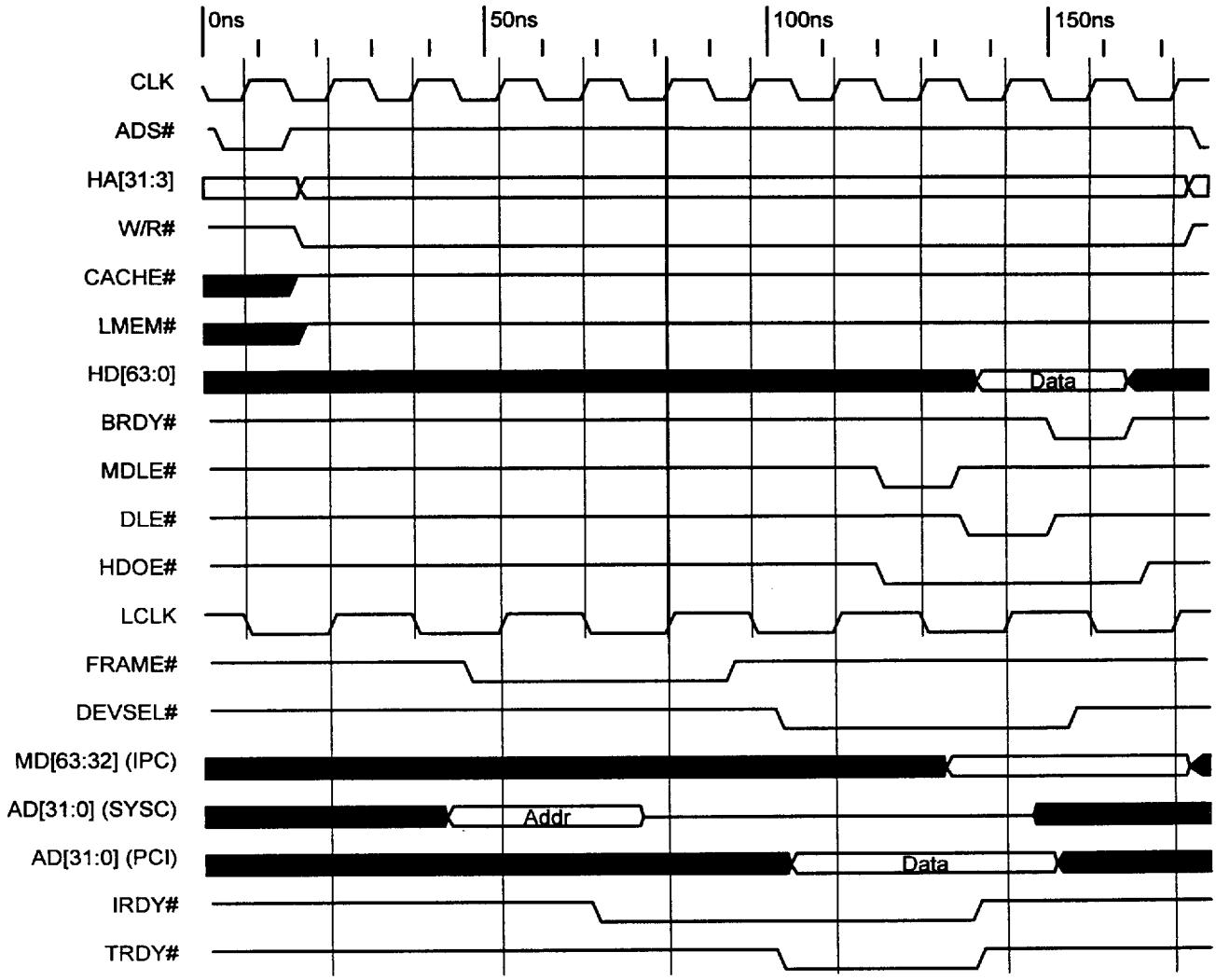


Figure 4-28 CPU Write to PCI, 32 Bits

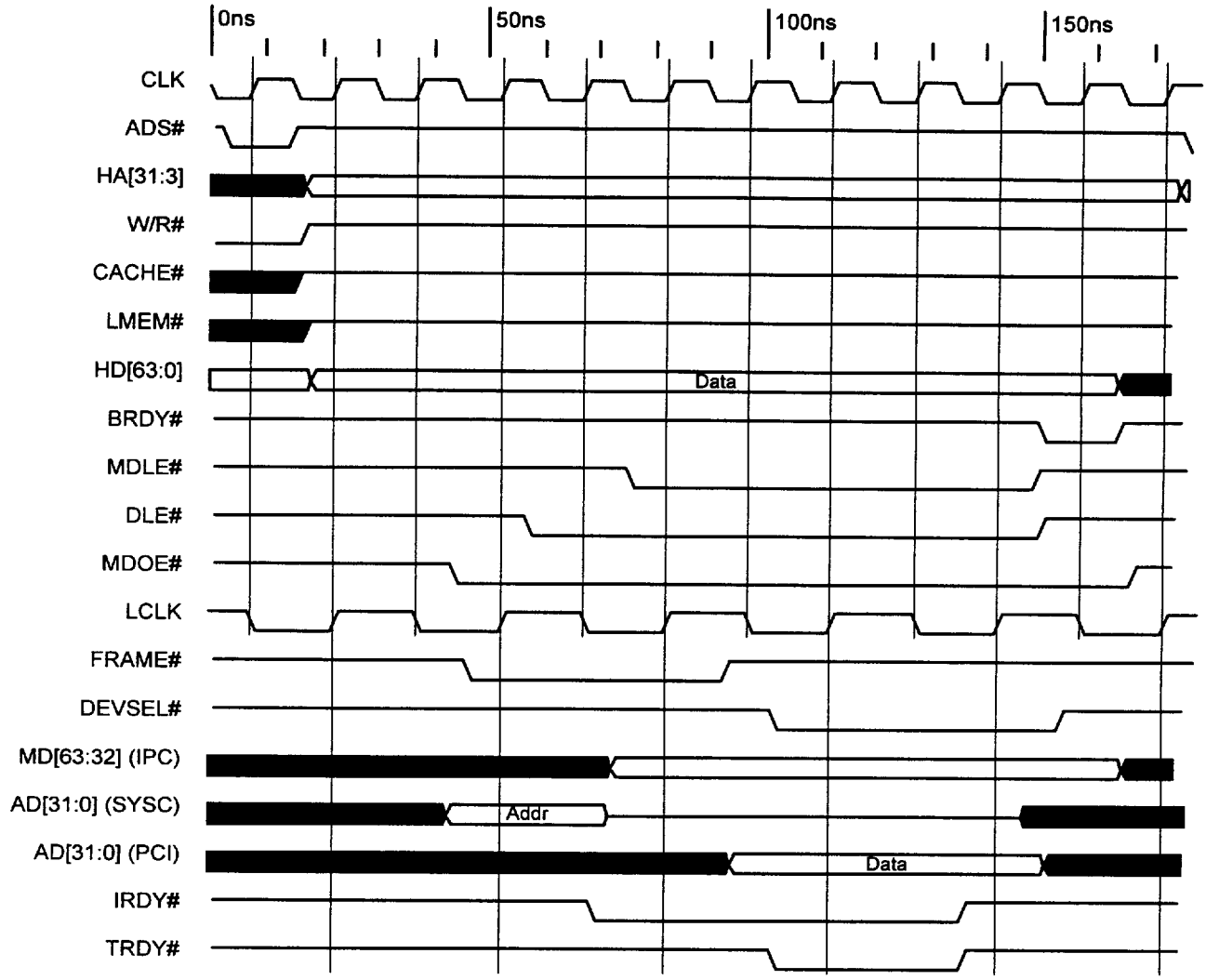


Figure 4-29 CPU Read from PCI, 64 Bits

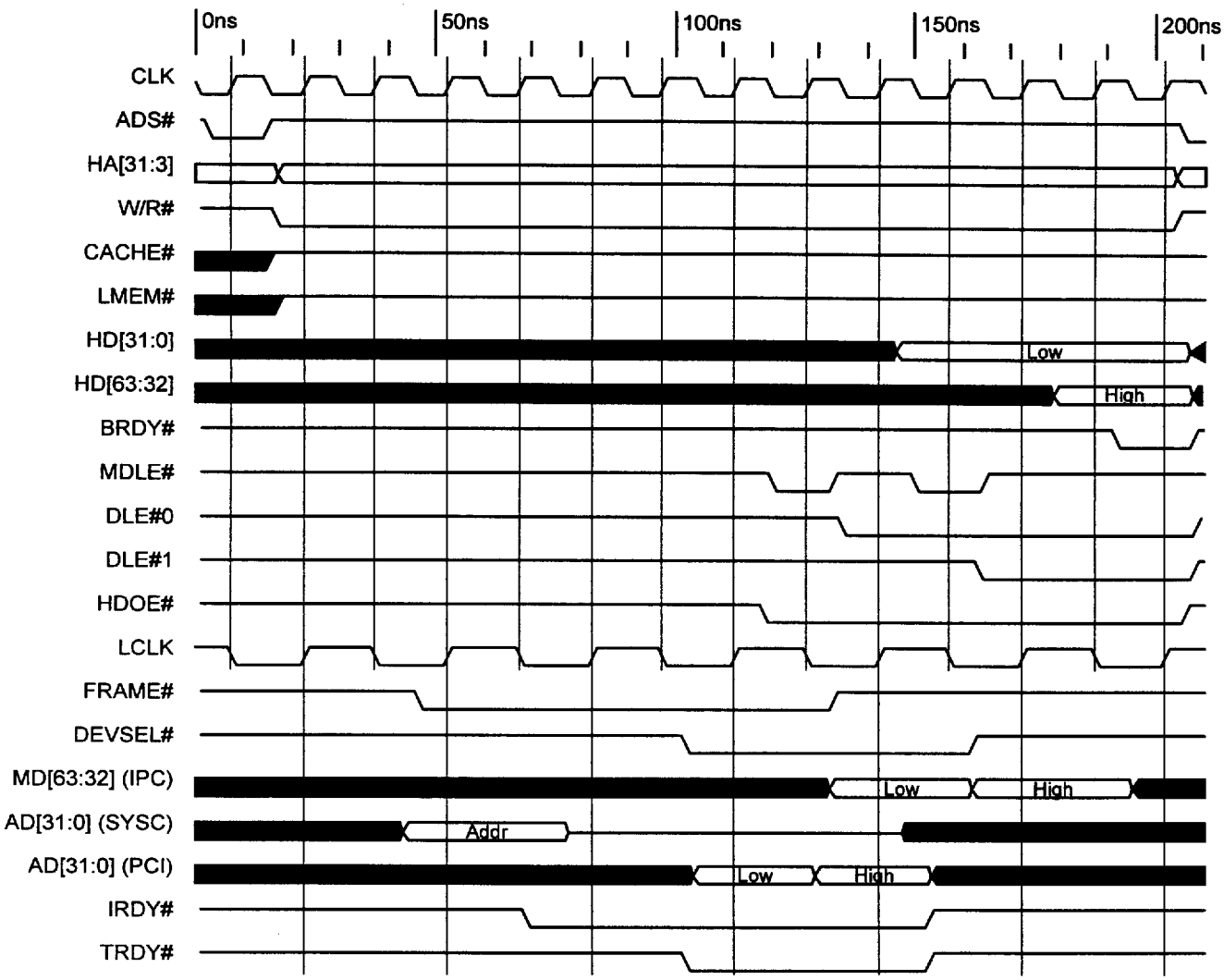


Figure 4-30 CPU Read from VESA Slave

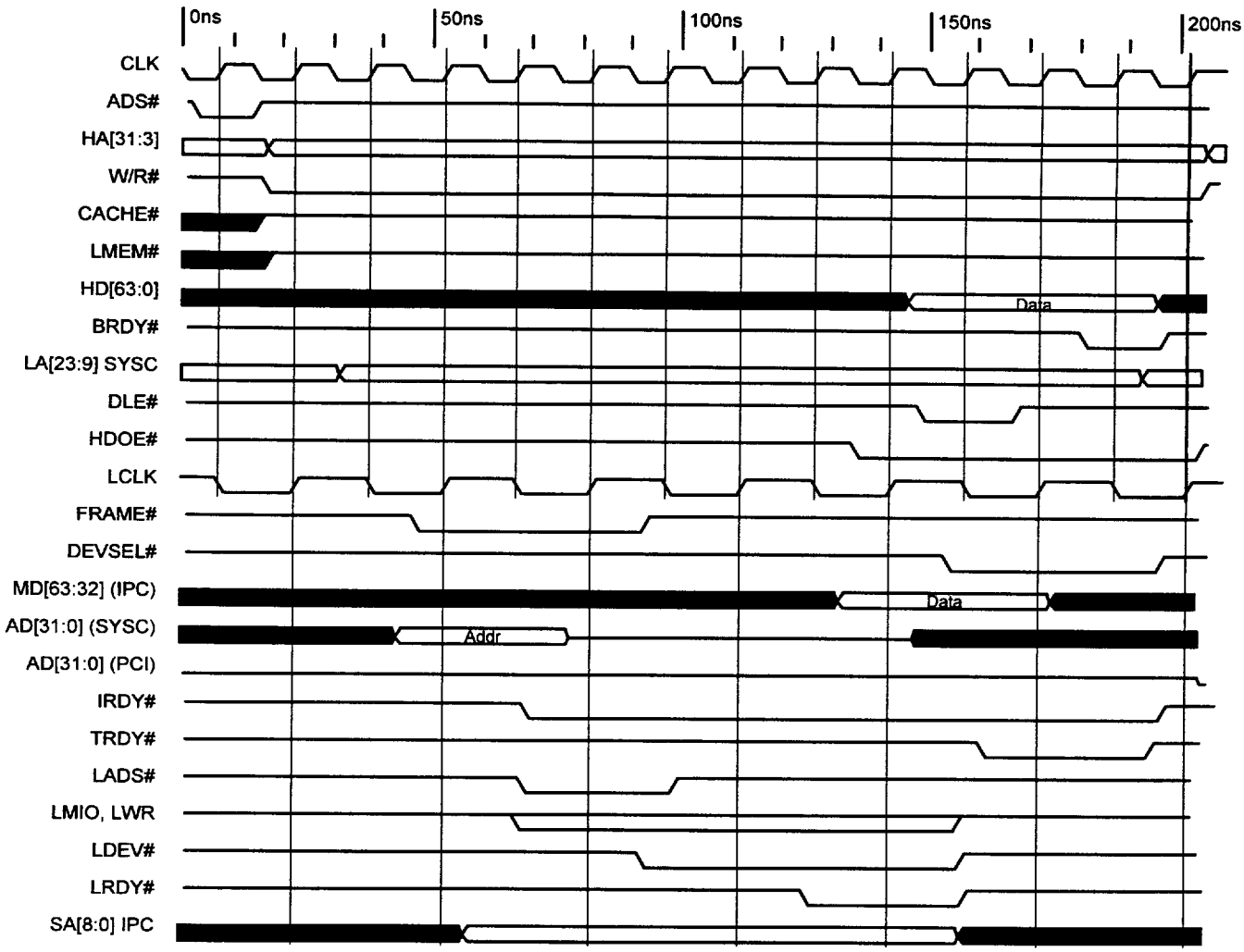


Figure 4-31 CPU Write to VESA Slave

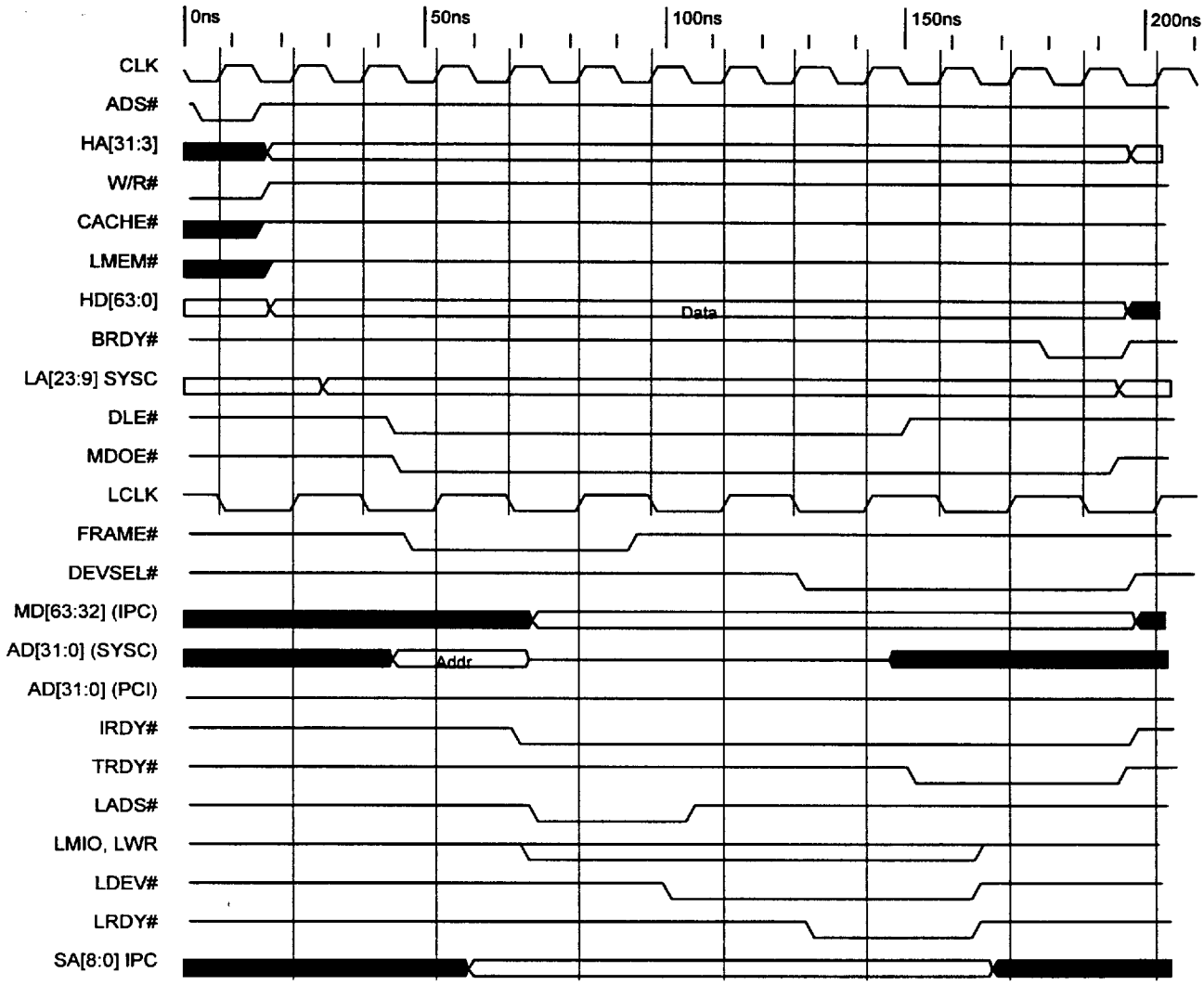


Figure 4-32 ISA Master Read from PCI

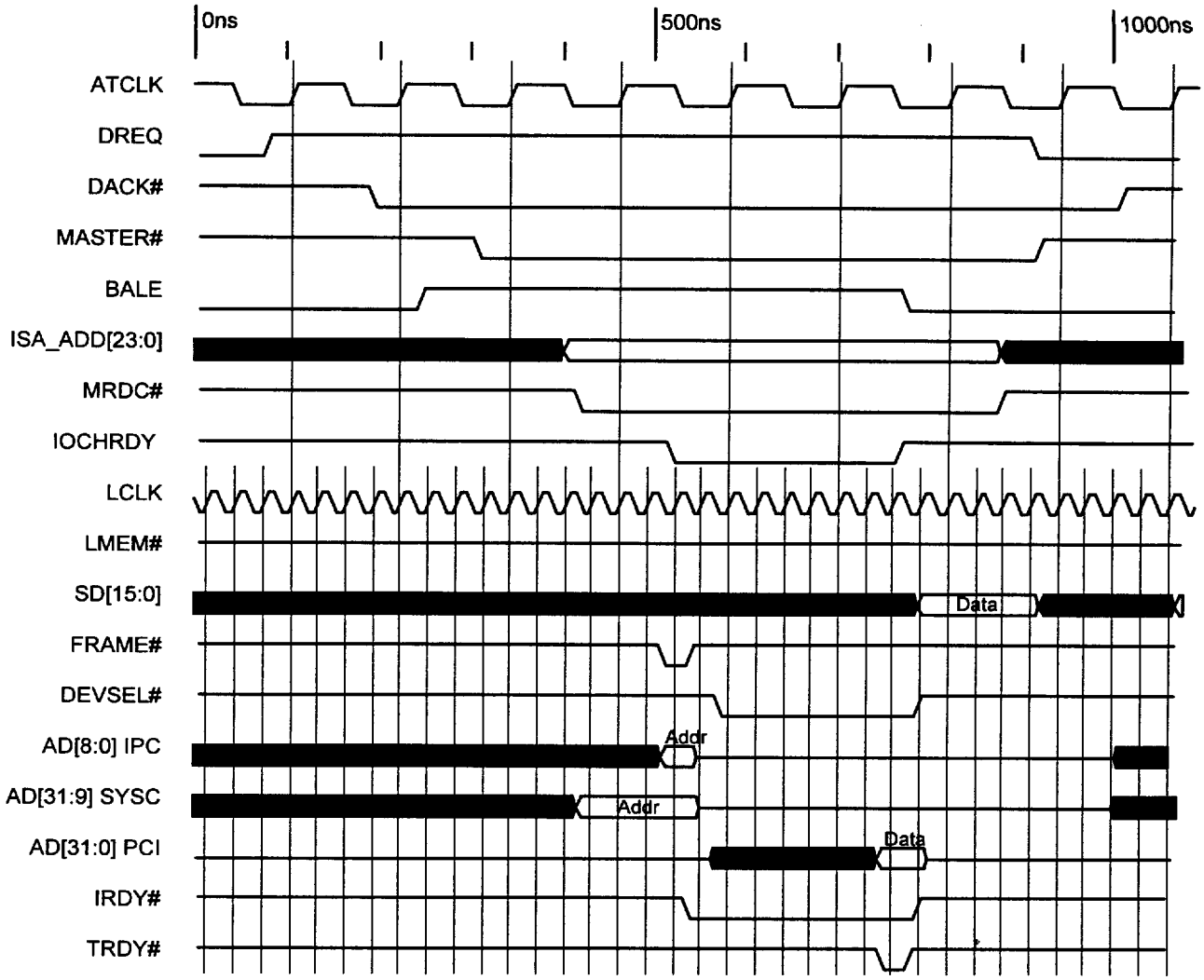


Figure 4-33 ISA Master Write to PCI

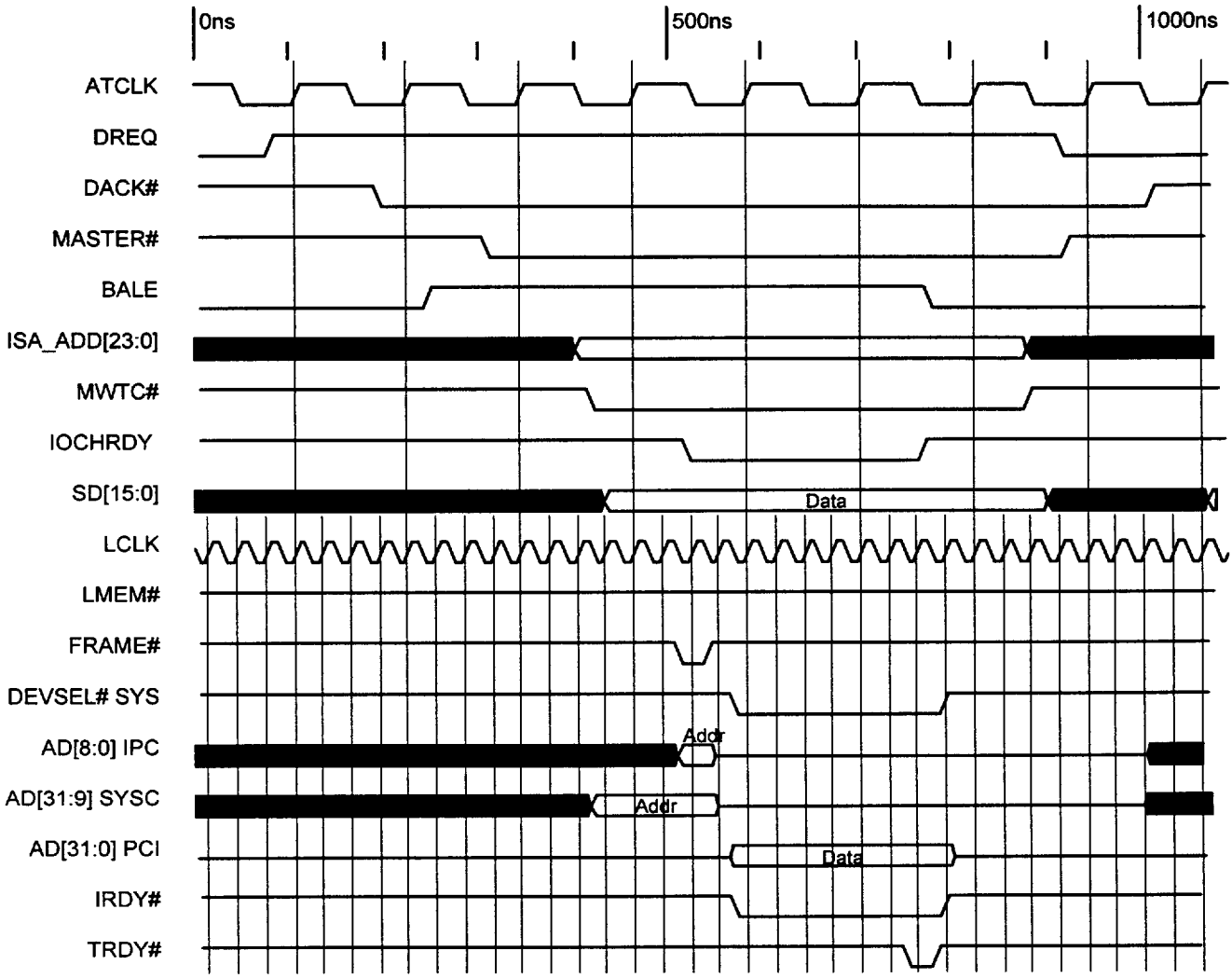


Figure 4-34 ISA Master Read from ISA Slave

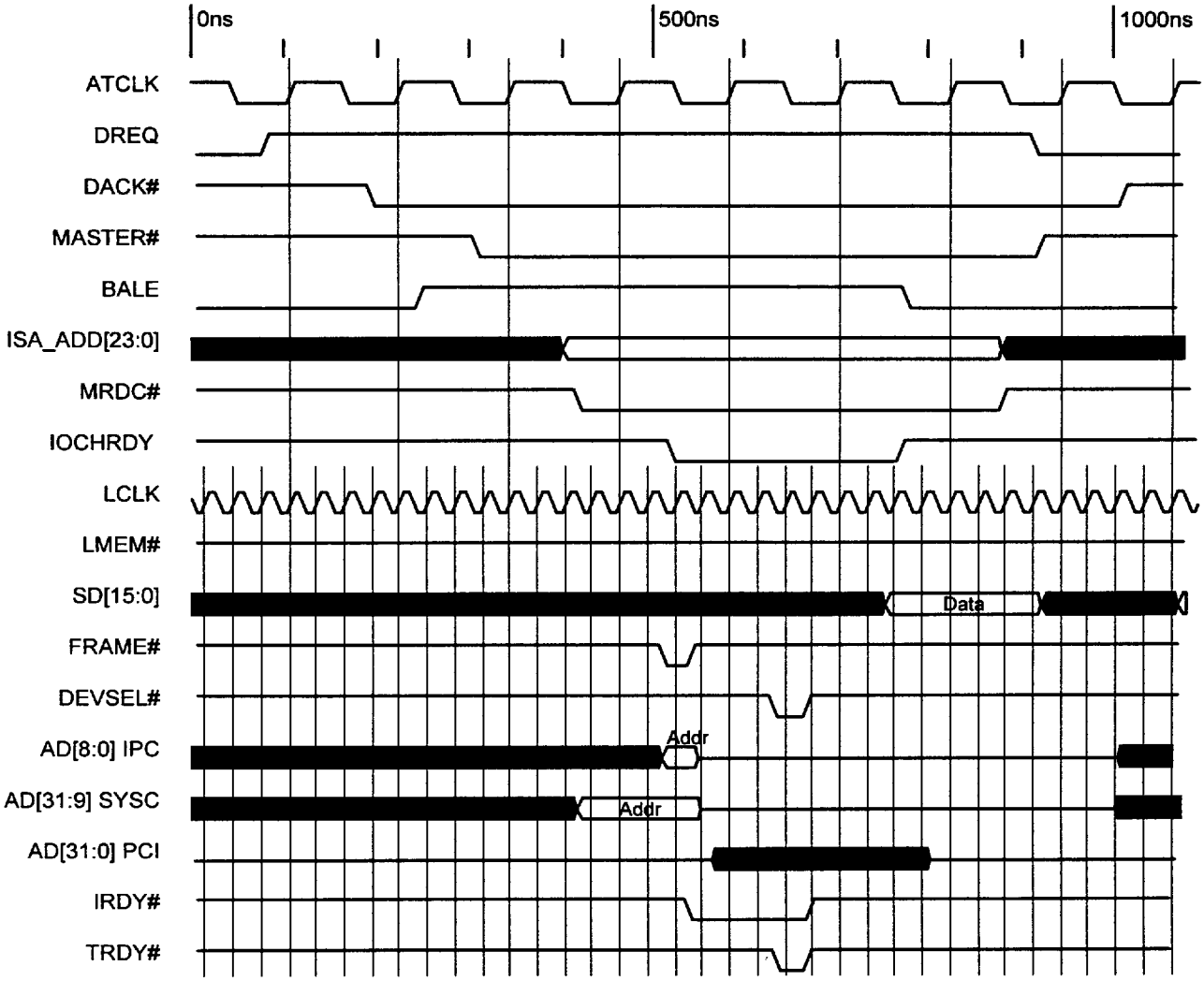
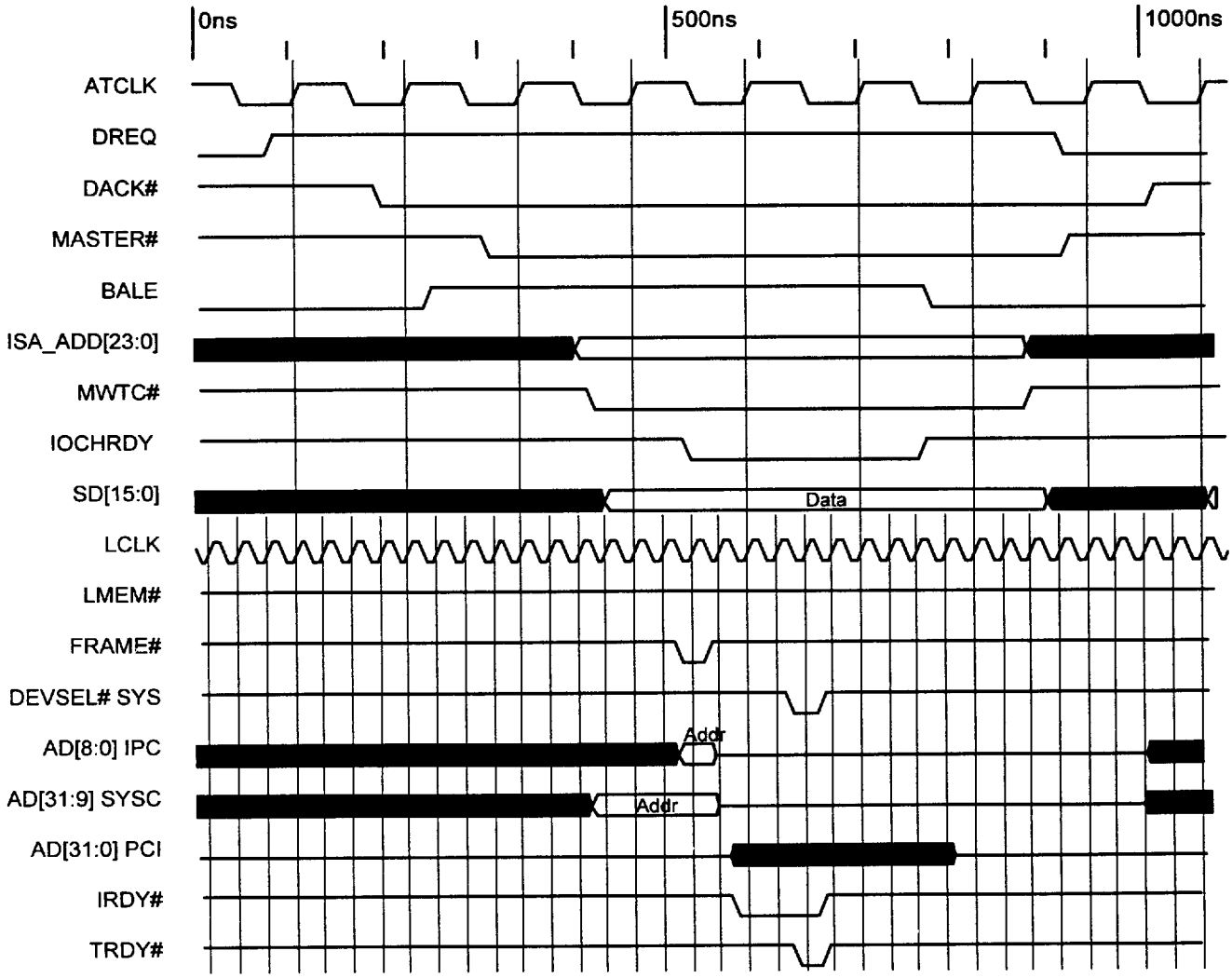


Figure 4-35 ISA Master Write to ISA Slave



4.7 AT Bus Interface

The AT bus state machine gains control when the IPC's decoding logic detects that no PCI/VL device has claimed the cycle. It monitors status signals MEMCS16#, IOCS16#, IOCHRDY, and ZEROWS# and performs the necessary synchronization of control and status signals between the AT bus and the microprocessor. The Viper-DP Desktop Chipset supports 8- and 16-bit memory and I/O devices located on the AT bus.

An AT bus cycle is initiated by asserting BALE in AT-TS1 state. On the trailing edge of BALE, M16# is sampled for a memory cycle to determine the bus size. It then enters AT-TC state and provides the command signal. For an I/O cycle, IO16# is sampled after the trailing edge of ALE until the end of the command. The command cycle is extended when IOCHRDY is detected inactive or the cycle is terminated when the zero wait state request signal (ZEROWS#) from the AT bus is active. Upon expiration of the wait states, the AT state machine terminates itself and passes internal READY to the CPU state machine to output a synchronous BRDY# to the CPU. The AT bus state machine also routes data and address when an AT bus master or DMA controller accesses system memory.

The delay between back to back ISA cycles is programmable and can be configured by programming the appropriate bits in Index 43h[3:2] of the IPC system control registers.

4.8 XD Bus Interface

The XD bus is an 8-bit utility that is used to access the 8-bit keyboard controller, BIOS ROM, the real-time clock, and the Non-Volatile RAM (NVRAM). The XDIR output signal from the IPC is used for the XD bus data buffer direction control. A 1 indicates data transfer from the SD bus to the XD bus. Normally high, it is low for the following conditions:

1) during BIOS ROM accesses, when ROMCS# and MEMR# are both active and
2) during reads from I/O Ports 60h, 64h, 70h, and 71h.
3) during read accesses from the NVRAM

4.9 Bus Arbitration Logic

The SYSC provides arbitration between the CPU, DMA controller, AT bus masters, PCI bus masters, and the refresh logic. During DMA, AT bus master cycles, PCI bus master cycles and conventional refresh cycles, the SYSC asserts HOLD to the CPU. The CPU will respond to an active HOLD signal by generating HLDA (after completing its current bus cycle) and placing most of its output and I/O pins in a high impedance state. After the CPU relinquishes the bus, the SYSC responds by issuing REFRESH# (refresh cycle) or AHOLD (PCI master, AT bus master or DMA cycle), depending on the requesting device. During hidden refresh, HOLD remains negated and the CPU continues its current program execution as long as it services internal requests or achieves cache hits.

Normally, refresh cycles, DMA cycles and master cycles are serviced on a first in-first out (FIFO) priority, but DRAM refresh requests (REFRESH#) are internally latched and serviced immediately after the current DMA or master finishes its request, if the refresh request was queued behind an AT DMA or master (HREQ) request. The SYSC will now request the CPU bus by asserting HOLD to the CPU. The CPU will complete the ongoing cycle and when it gives up the CPU bus, it will assert HLDA to the SYSC. The SYSC will grant the CPU bus to the PCI master, ISA DMA or master and assert AHOLD. The HREQ signal must remain active to be serviced if a refresh request comes first. DMA and bus masters share the same request pin; HREQ. To distinguish between DMA and bus master requests during an active AHOLD period, the AEN signal can be used to distinguish between DMA and master cycles. If AEN is active, then it is a DMA cycle. When these signals are inactive, then an external bus master controls the system bus.

4.10 Data Bus Conversion/Data Path Control Logic

Data bus conversion from the 64-bit CPU bus to the memory bus is done by the DBC (based on control signals from the SYSC). The data bus conversion from the higher order MD bus to the AD bus is done by the IPC, and the conversion to a 8/16-bit AT bus is also done by the IPC. The SYSC converts the CPU byte enable BE[7:0]# to address A2 and four byte enable signals C/BE[3:0]#, for the PCI bus, the VL bus and the IPC. The IPC uses the C/BE[3:0]#, A2 and the other AT address (A[1:0], SBHE# and IOCS16#/MEMCS16#) information to complete the 64-bit to 8/16-bit data conversion for the AT bus. The IPC performs data bus conversion when the CPU accesses 16- or 8-bit devices through 16- or 32-bit instructions. It also handles DMA and AT master cycles that transfer data between local DRAM or cache memory and locations on the AT bus. The SYSC provides all of the signals to control external bidirectional data buffers.

4.11 Internal Peripherals Controller

The following subsections give detailed operational information about the 82C558's internal integrated 82C206.

4.11.1 Top Level Decoder and Configuration Register

The 82C206's top level decoder provides eight separate enables to various internal subsystems. The following is a truth table for the top level decoder.

Address Range	Selected Device
000h-00Fh	DMA8 - 8-bit DMA Controller
020h-021h	INTC1 - Interrupt Controller 1
022h-023h	CONFIG - Configuration Register
040h-043h	CTC - Counter/Timer
080h-08Fh	DMAPAGE - DMA Page Register
0A0h-0A1h	INTC2 - Interrupt Controller 2
0C0h-0DFh	DMA16 - 16-Bit DMA Controller

Refer to Section 5.0, Register Descriptions, to program the various 82C206 registers.

4.11.2 DMA Subsystem

The 82C206 contains two 8237 DMA controllers. Each controller is a four channel DMA device which will generate the memory address and control signals necessary to transfer data between a peripheral device and memory directly. The two DMA controllers are internally cascaded to provide four DMA channels for transfers to 8-bit peripherals (DMA8) and three channels for transfers to 16-bit peripherals (DMA16). Channel 0 of DMA16 provides the cascade interconnection of the two DMA controllers, hence maintaining PC/AT compatibility. Hereafter, the description of the DMA subsystem pertains to both DMA8 and DMA16 unless otherwise noted.

Table 4-17 gives the I/O address map of the 82C206's DMA subsystem. The mapping is fully PC/AT compatible.

4.11.2.1 DMA Operation

During normal operation, the DMA subsystem of the 82C206 will be in one of three modes: the Idle mode, Program mode, or the Active mode. When the DMA controller is in the Idle mode, it only executes idle state cycles. The DMA controller will remain in the Idle mode unless it has been initialized to work and one of the DMA request pins has been asserted. In this case, the DMA controller will exit the Idle mode and enter the Active mode. The DMA controller will also exit the Idle mode and enter the Program mode when the CPU attempts to access its internal registers.

4.11.2.1.1 Idle Mode

If no peripheral requests service, the DMA subsystem will enter the Idle mode and perform only idle states. During this time, the 82C206 will sample the DREQ input pins every clock cycle to determine if any peripheral is requesting a DMA service. The internal select from the top level decoder and HLDA input pin will also sample at the same time to determine if the CPU is attempting to access the internal registers. With either of the above conditions, the DMA subsystem will exit the Idle mode and enter either the Program or Active mode. Note that the Program mode has priority over the Active mode since a CPU cycle has already started before the DMA was granted use of the bus.

4.11.2.1.2 Program Mode

The DMA subsystem will enter the Program mode whenever HLDA is inactive and an internal select from the top level decoder is active. During this time, the address lines A[3:0] become inputs if DMA8 is selected or A[4:1] become inputs if DMA16 is selected. These address inputs are used to decode which registers in the DMA controller are to be accessed. The IOR# and IOW# signals are used to select and time the CPU reads or writes. When DMA16 is selected, A0 is not used to decode and is ignored. Due to the large number and size of the internal registers of the DMA controller, an internal byte pointer flip-flop is used to supplement the addressing of the 16-bit word and count address registers. This byte pointer is used to determine the upper or lower byte of word count and address registers and is cleared by a hardware reset or a master clear command. It may also be set or cleared by the CPU's set byte pointer flip-flop or clear byte pointer flip-flop commands.

The DMA subsystem supports some special commands when in the Program mode. These commands do not use the data bus, but are derived from a set of address, the internal select, and IOR# or IOW#. These commands are listed at the end of Table 4-17. Erratic operation of the 82C206 can occur if a request for service occurs on an unmasked DMA channel which is being programmed. The channel should be masked or the DMA should be disabled to prevent the 82C206 from attempting to service a peripheral with a channel which is only partially programmed.

Table 4-17 DMA I/O Address Map

Address		Operation			
DMA8	DMA16	XIOR#	XIOW#	Byte Pointer	Register Function
000h	0C0h	0	1	0	Read Channel 0's current address low byte
		0	1	1	Read Channel 0's current address high byte
		1	0	0	Write Channel 0's base and current address low byte
		1	0	1	Write Channel 0's base and current address high byte
001h	0C2h	0	1	0	Read Channel 0's current word count low byte
		0	1	1	Read Channel 0's current word count high byte
		1	0	0	Write Channel 0's base and current word count low byte
		1	0	1	Write Channel 0's base and current word count high byte
002h	0C4h	0	1	0	Read Channel 1's current address low byte
		0	1	1	Read Channel 1's current word count high byte
		1	0	0	Write Channel 1's base and current address low byte
		1	0	1	Write Channel 1's base and current address high byte
003h	0C6h	0	1	0	Read Channel 1's current word count low byte
		0	1	1	Read Channel 1's current word count high byte
		1	0	0	Write Channel 1's base and current word count low byte
		1	0	1	Write Channel 1's base and current word count high byte
004h	0C3h	0	1	0	Read Channel 2's current address low byte
		0	1	1	Read Channel 2's current address high byte
		1	0	0	Write Channel 2's base and current address low byte
		1	0	1	Write Channel 2's base and current address high byte
005h	0CAh	0	1	0	Read Channel 2's current word count low byte
		0	1	1	Read Channel 2's current word count high byte
		1	0	0	Write Channel 2's base and current word count low byte
		1	0	1	Write Channel 2's base and current word count high byte
006h	0CCh	0	1	0	Read Channel 3's current address low byte
		0	1	1	Read Channel 3's current address high byte
		1	0	0	Write Channel 3's base and current address low byte
		1	0	1	Write Channel 3's base and current address high byte
007h	0CEh	0	1	0	Read Channel 3's current word count low byte
		0	1	1	Read Channel 3's current word count high byte
		1	0	0	Write Channel 3's base and current word count low byte
		1	0	1	Write Channel 3's base and current word count high byte
008h	0D0h	0	1	X	Read Status Register
		1	0	X	Write Command Register
009h	0D2h	0	1	X	Read DMA Request Register
		1	0	x	Write DMA Request Register
00Ah	0D4h	0	1	X	Read Command Register
		1	0	X	Write single bit DMA Request Mask Register
00Bh	0D6h	0	1	X	Read Mode Register
		1	0	X	Write Mode Register
00Ch	0D8h	0	1	X	Set byte pointer flip-flop
		1	0	X	Clear byte pointer flip-flop
00Dh	0DAh	0	1	X	Read Temporary Register
		1	0	X	Master clear
00Eh	0DCh	0	1	X	Clear Mode Register counter
		1	0	X	Clear all DMA Request Mask Register bits
00Fh	0DEh	0	1	X	Read all DMA Request Mask Register bits
		1	0	X	Write all DMA Request Mask Register bits

4.11.2.1.3 Active Mode

The DMA subsystem will enter the Active mode whenever a software request occurs or a DMA request occurs on an unmasked channel which has already been programmed. An example of this would be a DMA read cycle. After receiving a DREQ, the 82C206 will issue HOLD to the CPU. Until an HLDA is returned from the CPU, the DMA subsystem will remain in an idle state. On the next clock cycle, the DMA will exit the idle state and enter an S0 state. During S0, the DMA will resolve priority and issue DACK on the highest priority channel which is requesting service. The DMA then enters the S1 state where the multiplexed addresses are output and latched. Next, the DMA enters the S2 state where the 82C206 asserts the MEMR# command. Then the DMA will enter the S3 state where the 82C206 asserts the IOW# command. The DMA will then remain in the S3 state until the wait state counter has expired and IOCHRDY is high. Note that at least one additional S3 will occur unless compressed timing is programmed. Once a ready condition is detected, the DMA will enter S4 where MEMR# and IOW# are negated.

In the Compressed and Demand modes, subsequent transfers will begin in S2 unless the intermediate addresses require updating. In these subsequent transfers, the lower addresses are changed in S2.

4.11.2.2 DMA Transfer Modes

There are four transfer modes supported by the DMA subsystem: Single, Block, Demand, and Cascade. The DMA subsystem can be programmed on a channel-by-channel basis to operate in one of these four modes.

4.11.2.2.1 Single Transfer Mode

In the Single Transfer mode, the DMA will execute only one cycle at a time. DREQ must be held active until DACK becomes active in order to be recognized. If DREQ is held active throughout the Single Transfer, the 82C206 will negate HOLD and release the bus to the system once the transfer is complete. After HLDA has gone inactive, the 82C206 will again assert HOLD and execute another transfer on the same channel unless a request from a higher priority channel has been received.

During the Single Transfer mode, the CPU is ensured of at least one full machine cycle execution between DMA transfers. Following each transfer, the Word Count Register is decreased and the Address Register is increased or decreased (depending on the DEC bit of the Mode Register). When the word count decrements from 0000h to FFFFh, the terminal count bit in the Status Register is set and a pulse is output to the TC pin. If auto-initialization is selected, the channel will reinitialize itself for the next service - otherwise, the DMA will set the corresponding DMA request bit mask and suspend transferring on that channel.

4.11.2.2.2 Block Transfer Mode

In the Block Transfer mode, the DMA will begin transfers in response to either a DREQ or a software reset. If DREQ starts the transfer, it needs to be held active until DACK becomes active. The transfers will continue until the word count decrements from 0000h to FFFFh, at which time the TC pin is pulsed and the terminal count bit in the Status Register is set. Once more, an auto-initialization will occur at the end of the last service if the channel has been programmed to do so.

4.11.2.2.3 Demand Transfer Mode

In the Demand Transfer mode, the DMA will begin transfers in response to the assertion of DREQ and will continue until either the terminal count is reached or DREQ becomes active. The Demand Transfer mode is normally used for peripherals which have limited buffering capacity. The peripheral can initiate a transfer and continue until its buffer capacity is exhausted. The peripheral may then re-establish service by again asserting DREQ. During idle states between transfers, the CPU is released to operate and can monitor the operation by reading intermediate values from the Address and Word Count Registers. Once DREQ is negated, higher priority channels are allowed to intervene. Reaching the terminal count will result in the generation of a pulse on the TC pin, the setting of the terminal count bit in the Status Register, and auto-initialization if programmed to do so.

4.11.2.2.4 Cascade Mode

The Cascade mode is used to interconnect more than one DMA controller to extend the number of DMA channels while preserving the priority chain. While in this mode, the master DMA controller does not generate address or control signals. The DREQ and DACK signals of the master are used to interface the HOLD and HLDA signals of the slave DMA devices. Once the master has received an HLDA from the CPU in response to a DREQ caused by the HOLD from a slave DMA controller, the master DMA controller will ignore all inputs except HLDA from the CPU and DREQ on the active channel. This prevents conflicts between the DMA devices.

Figure 4-36 shows the cascade interconnection for two levels of DMA devices. Note that Channel 0 of DMA16 is internally connected for the Cascade mode to DMA8. Additional devices can be cascaded to the available channels in either DMA8 or DMA16 since the Cascade mode is not limited to two levels of DMA controllers.

When programming cascaded controllers, begin with the device which is actually generating HRQ to the system (first level device) and then proceed to the second level devices. RESET causes the DACK outputs to become active low and are placed in the inactive state. To allow the internal cascade between DMA8 and DMA16 to operate correctly, the active

low state of DACK should not be modified. The first level device's DMA request mask bits will prevent the second level cascaded devices from generating unwanted hold requests during the initialization process.

4.11.2.3 Transfer Types

There are three types of transfers:

- Read Transfers
- Write Transfers
- Verify Transfers

The Single, Block, and Demand Transfer modes can perform any of the three transfer types.

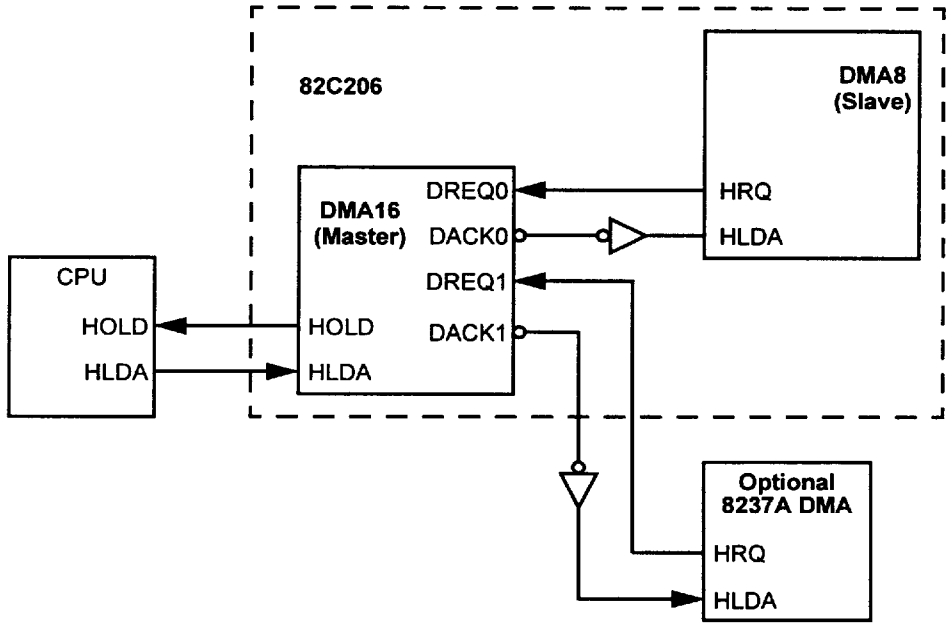
Read Transfers move data from memory to an I/O peripheral by generating the memory address and asserting MEMR# and IOW# during the same transfer cycle.

Write Transfers move data from an I/O peripheral to memory by generating the memory address and asserting MEMW# and IOR# during the same transfer cycle.

Verify Transfers are pseudo transfers. In this type of transfer, the DMA will operate as in Read or Write Transfers by generating HOLD, DACK, memory addresses and respond to the terminal count, but it does not activate the memory or I/O command signals. Since no transfer actually takes place, IOCHRDY is also ignored during Verify Transfers.

In addition to the three transfer types mentioned above, there is also a memory-to-memory transfer which can only be used on DMA Channels 0 and 1. The memory-to-memory transfer is used to move a block of memory from one location in memory to another. DMA Channels 0 and 1 may be programmed to operate as memory-to-memory channels by setting a bit in the DMA Command Register. Once programmed, the transfer can be started by generating either a software or an external request to Channel 0. During the transfer, Channel 0 provides the address for the source block during the memory write portion of the same transfer. During the read portion of the transfer, a byte of data is latched in the internal Temporary Register of the DMA. The contents of this register are then output on the SD[7:0] output pins during the write portion of the transfer and subsequently written to the memory location. Channel 0 may be programmed to maintain the same source address on every transfer. This allows the CPU to initialize large blocks of memory with the same value. The DMA subsystem will continue performing transfers until Channel 1 reaches the terminal count.

Figure 4-36 Cascade Mode Interconnect



4.11.2.3.1 Auto-initialization

The Mode Register of each DMA channel contains a bit which will cause the channel to reinitialize after reaching the terminal count. During auto-initialization, the Base Address and Base Word Count Registers (which were originally programmed by the CPU) are reloaded into the Current Address and Current Word Count Registers. The Base Registers remain unchanged during DMA active cycles and can only be changed by the CPU. If the channel has been programmed to auto-initialize, the request mask bit will remain cleared upon reaching the terminal count. This allows the DMA to continue operation without CPU intervention. In memory-to-memory transfers, the Word Count Registers of Channels 0 and 1 must be programmed with the same starting value for full auto-initialization.

4.11.2.3.2 DREQ Priority

The 82C206 supports two types of software programmable priority schemes: fixed and rotating. Fixed priority assigns priority based on channel position. With this method, Channel 0 is assigned the highest priority and Channel 3 is the lowest. After the recognition of any one channel for service, the other channels are prevented from interfering with that service until it is completed.

In the rotating priority scheme, the ordering of priority from Channel 0 to Channel 3 is maintained, but the actual assignment of priority changes. The channel most recently serviced will be assigned the lowest priority and since the order of priority assignment remains fixed, the remaining three channels rotate accordingly. Table 4-18 shows the rotating priority scheme. In cases where multiple requests occur at the same time, the 82C206 will issue HOLD but will not freeze the priority logic until HLDA is returned. After HLDA becomes active, the priority logic is frozen and DACK is asserted on the highest requesting channel. Priority will not be reevaluated until HLDA has been deactivated.

4.11.2.3.3 Address Generation

During active cycles of the DMA, eight intermediate bits of the address are multiplexed onto the data lines. This reduces the number of pins required by the DMA subsystem. During an S1 state, the intermediate addresses are output on data lines SD[7:0]. These addresses should be externally latched and used to drive the system address bus. Since DMA8 is used for 8-bit transfers and DMA16 is used for 16-bit transfers, a one bit skew occurs in the intermediate address fields. DMA8 will therefore output address on LA[15:8] on the data bus at this time whereas DMA16 will output LA[16:9]. A separate set of latch and enable signals are provided for both DMA8 and DMA16 to accommodate the address skew.

During 8-bit DMA transfers in which DMA8 is active, the 82C206 will output the lower eight bits of address on SA[7:0]. LA[23:16] are also generated at this time from a DMA page register in the 82C206. Note that A16 is output on the A16 pin of the device.

During 16-bit DMA transfers in which DMA16 is active, the 82C206 will output the lower eight bits of address on SA[8:1]. LA[23:17] are also generated at this time from a DMA page register in the 82C206. Note that SA0 and LA16 remain tristated during 16-bit DMA transfers

The DMA page registers are a set of 16 8-bit registers in the 82C206 which are used to generate the high order addresses during DMA cycles. Only eight of the registers are actually used, but all 16 were included to maintain PC/AT compatibility. Each DMA channel has a page register associated with it except Channel 0 of DMA16 which is used for cascading to DMA8. Assignment of each of these registers is shown in Table 4-19 along with its CPU I/O read/write address.

Table 4-18 Rotating Priority Scheme

Priority	First Arbitration	Second Arbitration	Third Arbitration
Highest	Channel 0	Channel 2 - Cycle Grant	Channel 3 - Cycle Grant
	Channel 1 - Cycle Grant	Channel 3	Channel 0
	Channel 2	Channel 0	Channel 1
Lowest	Channel 3	Channel 1	Channel 2

Channel X = Requested Channel

During Demand and Block Transfers, the 82C206 generates multiple sequential transfers. For most of these transfers, the information in the external address latches will remain the same, thus eliminating the need to be relatched. Since the need to update the latches occurs only when a carry or borrow from the lower eight bits of the address counter exists, the 82C206 will only update the latch contents when necessary. The 82C206 execute an S1 state only when necessary and improve the overall system throughput.

Table 4-19 DMA Page Register I/O Address Map

I/O Addr	Type	Register Function
080h	R/W	Unused
081h	R/W	DMA8 Channel 2 (DACK2)
082h	R/W	DMA8 Channel 3 (DACK3)
083h	R/W	DMA8 Channel 1 (DACK1)
084h	R/W	Unused
085h	R/W	Unused
086h	R/W	Unused
087h	R/W	DMA8 Channel 0 (DACK0)
088h	R/W	Unused
089h	R/W	DMA16 Channel 2 (DACK6)
08Ah	R/W	DMA16 Channel 3 (DACK7)
08Bh	R/W	DMA16 Channel 1 (DACK5)
08Ch	R/W	Unused
08Dh	R/W	Unused
08Eh	R/W	Unused
08Fh	R/W	DRAM Refresh Cycle

4.11.2.3.4 Compressed Timing

The DMA subsystem in the 82C206 can be programmed to transfer a word in as few as two DMA clock cycles. Normal transfers require four DMA clock cycles since S3 is executed twice (due to the one wait state insertion). In systems capable of supporting higher throughput, the 82C206 can be programmed to omit one S3 and assert both commands in S2. S2 begins the cycle by generating the address and asserting both commands. One S3 cycle is executed and the cycle terminates in S4. If compressed timing is selected, TC will be output in S2 and S1 cycles which will be executed as necessary to update the address latch. Note that compressed timing is not allowed for memory-to-memory transfers.

4.11.3 DMA Register Descriptions

The following subsections are descriptions of the 82C206's internal peripherals controller DMA registers. The complete bit descriptions to these registers can be found in Section 5.0, Register Descriptions.

4.11.3.1 Current Address Register

Each DMA channel has a 16-bit Current Address Register which holds the address used during transfers. Each channel can be programmed to increment or decrement this register whenever a transfer is completed. This register can be read or written by the CPU in consecutive 8-bit bytes. If auto-initialization is selected, this register will be reloaded from the Base Address Register upon reaching the terminal count in the Current Word Count Register. Channel 0 can be prevented from incrementing or decrementing by setting the address hold bit in the Command Register.

4.11.3.2 Current Word Count Register

Each channel has a Current Word Count Register which determines the number of transfers. The actual number of transfers performed will be one greater than the value programmed into the register. The register is decremented after each transfer until it goes from 0 to FFFFh. When this rollover occurs, the 82C206 will generate TC and either suspend the operation on that channel and set the appropriate request mask bit, or auto-initialize and continue.

4.11.3.3 Base Address Register

Associated with each Current Address Register is a Base Address Register. This is a write-only register which is loaded by the CPU when writing to the Current Address Register. The purpose of this register is to store the initial value of the Current Address Register for auto-initialization. The contents of this register are loaded into the Current Address Register whenever the terminal count is reached and the auto-initialize bit is set.

4.11.3.4 Base Word Count Register

This register preserves the initial value of the Current Word Count Register. It too is a write-only register which is loaded by writing to the Current Word Count Register. The Base Word Count Register is loaded into the Current Word Count Register during auto-initialization.

4.11.3.5 Command Register

The Command Register controls the overall operation of the DMA subsystem. This register can be read or written by the CPU and is cleared by either a reset or master clear command.

4.11.3.6 Mode Register

Each DMA channel has a Mode Register associated with it. All four Mode Registers reside at the same I/O address. Bits 0 and 1 of the Write Mode Register command determine which channel Mode Register gets written. The remaining six bits control the mode of the selected channel. Each channel Mode Register can be read by sequentially reading the Mode Register location. A Clear Mode Register Counter command is provided to allow the CPU to restart the mode read process at a known point. During mode read operations, bit 0 and 1 will both equal 1.

4.11.3.7 Request Register

This 4-bit register is used to generate software requests (DMA service can be requested either externally or under software control). Request Register bits can be set or reset independently by the CPU. The register mask has no effect on software generated requests. All four bits are read in one operation and appear in the lower four bits of the byte. Bits 7 through 4 are read as 1s. All four request bits are cleared to 0 by a reset.

4.11.3.8 Request Mask Register

The Request Mask Register is a set of four bits which are used to inhibit external DMA requests from generating transfer cycles. This register can be programmed in two ways. Each channel can be independently masked by writing to the Write Single Mask bit location.

Alternatively, all four mask bits can be programmed in one operation by writing to the write all mask bits address.

All four mask bits are set following a reset or a Master Clear command. Individual channel mask bits will be set as a result of the terminal count being reached, if auto-initialize is disabled. The entire register can be cleared, enabling all four channels by performing a Clear Mask Register operation.

4.11.3.9 Status Register

The status of all four channels can be determined by reading the Status Register. Information is available to determine if a channel has reached the terminal count and whether an external service request is pending.

4.11.3.10 Temporary Register

The Temporary Register is used as a temporary holding register for data during memory-to-memory transfers. The register is loaded during the first cycle of a memory-to-memory transfer from SD[7:0]. During the second cycle of the transfer, the data in the Temporary Register is output on the SD[7:0] pins. Data from the last memory-to-memory; transfer will remain in the register.

4.11.4 Special Commands

Five special commands are provided to make the task of programming the 82C206 easier. These commands are activated as a result of a specific address and assertion of either IOR# or IOW#. For these special commands, the data bus is ignored by the 82C206 whenever an IOW# activated command is issued. Data returned on IOR# activated commands is undefined.

- Clear Byte Pointer Flip-Flop: This command is normally executed prior to reading or writing to the Address or Word Count Registers. This initializes the flip-flop to point to the low byte of the register and allows the CPU to read or write the register bytes in correct sequence.
- Set Byte Pointer Flip-Flop: Setting the byte pointer flip-flop allows the CPU to adjust the pointer to the high byte of an Address or Word Count Register.
- Master Clear: This command has the same effect as a hardware reset. The Command Register, Status Register, Request Register, Temporary Register, Mode Register counter, and byte pointer flip-flop are cleared and the Request Mask Register is set. Immediately following a Master Clear or reset, the DMA will be in the Idle mode.
- Clear Request Mask Register: This command enables all four DMA channels to accept requests by clearing the mask bits in the register.
- Clear Mode Register Counter: In order to allow access to the four Mode Registers while only using one address, an internal counter is used. After clearing the counter, all four Mode Registers may be read by successive reads to the Mode Register. The order in which the registers are read is Channel 0 first and Channel 3 last.

4.11.5 Interrupt Controller Subsystem

The programmable interrupt controllers in the 82C206 serve as a system wide interrupt manager. They accept requests from peripherals, resolve priority on pending interrupts and interrupts in service, issue an interrupt request to the CPU, and provide a vector which is used as an index by the CPU to determine which interrupt service routine to execute.

A variety of priority assignment modes are provided which can be reconfigured at any time during system operation. This allows the complete subsystem to be restructured based on the system environment.

4.11.5.1 Interrupt Controller Subsystem Overview

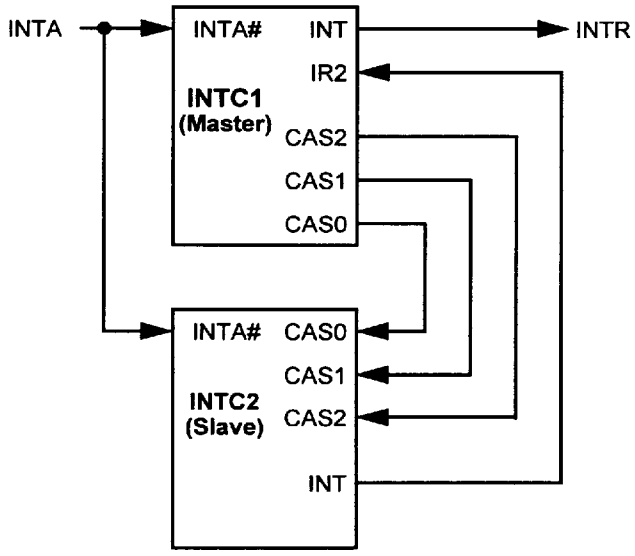
There are two interrupt controllers, INTC1 and INTC2, included in the 82C206. Each of the interrupt controllers is equivalent to an 8259A device operating in X86 mode. The two devices are interconnected and must be programmed to operate in the Cascade mode for all 16 interrupt channels to operate properly. Figure 4-37 shows the internal Cascade interconnection.

INTC1 is located at addresses 020h-021h and is configured for master operation in the Cascade mode. INTC2 is a slave device and is located at 0A0h-0A1h. The interrupt request output signal (INT) from INTC2 is internally connected to the interrupt request input Channel 2 (IR2) of INTC1. The address decoding and cascade interconnection matches that of the PC/AT.

Two additional interconnections are made to the interrupt request inputs of the interrupt controllers. The output of Timer 0 in the counter/timer subsystem is connected to Channel 0 (IR0) of INTC1. Interrupt request from the real-time clock is connected to Channel 0 (IR0) of INTC2. Table 4-20 lists the 16 interrupt channels and their interrupt request sources.

Description of the interrupt subsystem will pertain to both INCT1 and INCT2 unless otherwise noted. Wherever register addresses are used, the address for the INTC1 Register will be listed first and the address for the INTC2 Register will follow in parenthesis. Example: 02h (0A0h).

Figure 4-37 Internal Cascade Interconnect



Note: INTA will be active when the CPU initiates an interrupt acknowledge cycle.

Table 4-20 Interrupt Request Source

Interrupt Controller	Channel Name	Interrupt Request Source
INTC1	IR0	Counter/Timer OUT0
INTC1	IR1	IRQ1 input pin
INTC1	IR2	INTC2 cascade interrupt
INTC1	IR3	IRQ3 input pin
INTC1	IR4	IRQ4 input pin
INTC1	IR5	IRQ5 input pin
INTC1	IR6	IRQ6 input pin
INTC1	IR7	IRQ7 input pin
INTC2	IR0	Real-time clock IRQ
INTC2	IR1	IRQ9 input pin
INTC2	IR2	IRQ10 input pin
INTC2	IR3	IRQ11 input pin
INTC2	IR4	IRQ12 input pin
INTC2	IR5	IRQ13 input pin
INTC2	IR6	IRQ14 input pin
INTC2	IR7	IRQ15 input pin

4.11.5.2 Interrupt Controller Operation

Figure 4-38 is a block diagram of the major components in the interrupt controller subsystem. The Interrupt Request Register (IRR) is used to store requests from all of the channels which are requesting service. The IRR's bits are labeled using the channel name IR[7:0]. The In-Service Register (ISR) contains all the channels which are currently being serviced (more than one channel can be in service at a time). The ISR's bits are labeled IS[7:0] and correspond to IR[7:0]. The Interrupt Mask Register (IMR) allows the CPU to disable any or all of the interrupt channels. The Priority Resolver evaluates inputs from the IRR, ISR, and IMR, issues an interrupt request, and latches the corresponding bit into the ISR. During interrupt acknowledge cycles, a master controller outputs a code to the slave device which is compared in the Cascade Buffer/Comparator with a 3-bit ID code previously written. If a match occurs in the slave controller, it will generate an interrupt vector. The contents of the Vector Register are used to provide the CPU with an interrupt vector during interrupt acknowledge (INTA) cycles.

4.11.5.3 Interrupt Sequence

The 82C206 allows the CPU to perform an indirect jump to a service routine in response to a request for service in response to a request for service from as peripheral device. The indirect jump is based on a vector which is provided by the 82C206 on the second of two CPU generated INTA cycles (the first INTA cycle is used for resolving priority and the second is for transferring the vector to the CPU (see Figure 4-39). The events which occur during an interrupt sequence are as follows:

1. One or more of the interrupt requests (IR[7:0]) becomes active, setting the corresponding IRR bit(s).
2. The interrupt controller resolves priority based on the state of the IRR, IMR, and ISR and asserts the INTR output if needed.
3. The CPU accepts the interrupt and responds with an INTA cycle.
4. During the first INTA cycle, the highest priority ISR bit is set and the corresponding IRR bit is reset. The internal cascade address is generated.
5. The CPU will execute a second INTA cycle, during which the 82C206 will drive an 8-bit vector onto the data pins XD[7:0], which is read by the CPU. The format of this vector is shown in Table 4-21. Note that V[7:3] in Table 4-21 are programmable by writing to ICW2 (Initialization Command Word 2).
6. At the end of the second INTA cycle, the ISR bit will be cleared if the Automatic End of Interrupt Mode is selected (see below). Otherwise, the ISR bit must be cleared by an End of Interrupt (EOI) command from the CPU at the end of the interrupt service routine to allow further interrupts. If no interrupt request is present at the beginning of the first INTA cycle (i.e., a spurious interrupt), INCT1 will issue an interrupt level 7 vector during the second INTA cycle.

Figure 4-38 Interrupt Controller Block Diagram

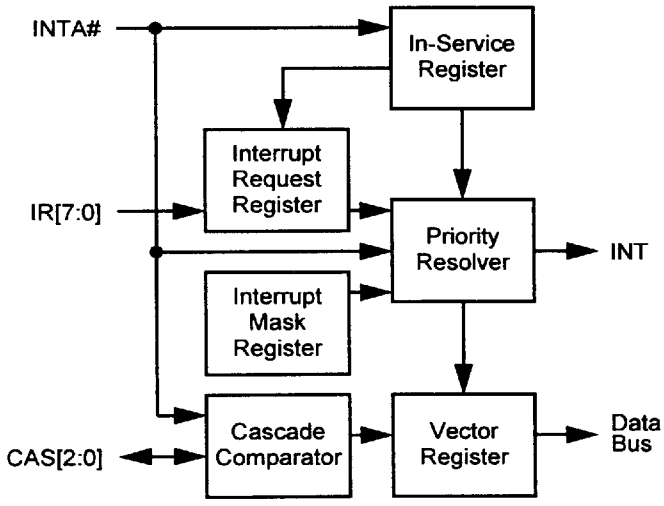


Figure 4-39 Interrupt Sequence

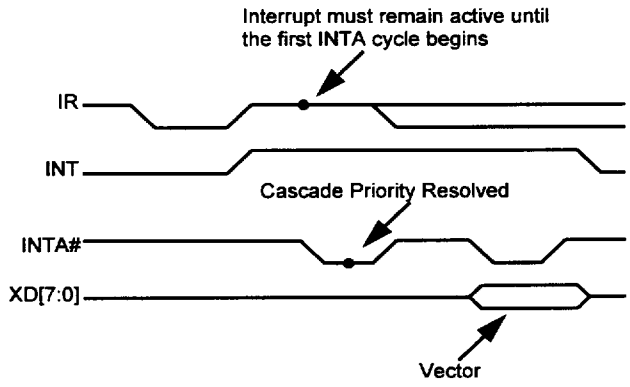


Table 4-21 Interrupt Vector Byte

Interrupt	D7	D6	D5	D4	D3	D2	D1	D0
IR7	V7	V6	V5	V4	V3	1	1	1
IR6	V7	V6	V5	V4	V3	1	1	0
IR5	V7	V6	V5	V4	V3	1	0	1
IR4	V7	V6	V5	V4	V3	1	0	0
IR3	V7	V6	V5	V4	V3	0	1	1
IR2	V7	V6	V5	V4	V3	0	1	0
IR1	V7	V6	V5	V4	V3	0	0	1
IR0	V7	V6	V5	V4	V3	0	0	0

4.11.5.4 End of Interrupt (EOI)

EOI is defined as the condition which causes an ISR bit to be reset. Determination of which ISR bit is to be reset can be done by a CPU command (specific EOI) or the Priority Resolver can be instructed to clear the highest priority ISR bit (nonspecific EOI). The 82C206 can determine the correct ISR bit to reset when operated in modes which do not alter the fully nested structure since the current highest priority ISR bit is the last level acknowledged and serviced. In conditions where the fully nested structure is not preserved, a specific EOI must be generated at the end of the interrupt service routine. An ISR bit that is masked, in the Special Mask Mode by an IMR bit, will not be cleared by a nonspecific EIO command. The interrupt controller can optionally generate an Automatic End of Interrupt (AEOI) on the trailing edge of the second INTA cycle.

4.11.5.5 Priority Assignment

Assignment of priority is based on an interrupt channel's position relative to the other channels in the interrupt controller. After the initialization sequence, IR0 has the highest priority, IR7 the lowest, and priority assignment is Fixed. Priority assignment can be rotated either manually (Specific Rotation Mode) or automatically (Automatic Rotation Mode) by programming Operational Command Word 2 (OCW2).

4.11.5.5.1 Fixed Priority Mode

This is the default condition which exists unless rotation (either manual or automatic) is enabled, or the controller is programmed for Polled Mode. In the Fixed Priority Mode, interrupts are fully nested with priority assigned as shown:

	Lowest							Highest
Priority Status	7	6	5	4	3	2	1	0

Nesting allows interrupts of a higher priority to generate interrupt requests prior to the completion of the interrupt service. When an interrupt is acknowledged, priority is resolved, the highest priority request's vector is placed on the bus, and the ISR bit for that channel is set. This bit remains set until an EIO (automatic or CPU generated) is issued to that channel. While the ISWR bit is set, all interrupts of equal or lower priority are inhibited. Note that a higher priority service routine will only be acknowledged if the CPU has internally re-enabled interrupts.

4.11.5.5.2 Specific Rotation Mode

Specific Rotation allows the system software to reassign priority levels by issuing a command which redefines the highest priority channel. Before rotation:

	Lowest							Highest
Priority Status	7	6	5	4	3	2	1	0

(Specific Rotation command issued with Channel 5 specified.) After rotation:

	Lowest							Highest
Priority Status	5	4	3	2	1	0	7	6

4.11.5.5.3 Automatic Rotation Mode

In applications where a number of equal priority peripherals are requesting interrupts, Automatic Rotation may be used to equalize the priority assignment. In this mode, after a peripheral is serviced it is assigned the lowest priority. All peripherals connected to the controller will be serviced at least once in eight interrupt requests to the CPU from the controller. Automatic Rotation will occur, if enabled, due to the occurrence of an EOI (automatic or CPU generated).

Before rotation (IR3 is the highest priority request being serviced):

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	1	1	0	0	1	0	0	0
	Lowest						Highest	
Priority Status	7	6	5	4	3	2	1	0

(Specific Rotation command issued with Channel 4 specified.) After rotation:

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	1	1	0	0	0	0	0	0
	Lowest						Highest	
Priority Status	3	2	1	0	7	6	5	4

4.11.5.6 Programming the Interrupt Controller

Two types of commands are used to control the 82C206's interrupt controllers: Initialization Command Words (ICWs) and Operational Command Words (OCWs).

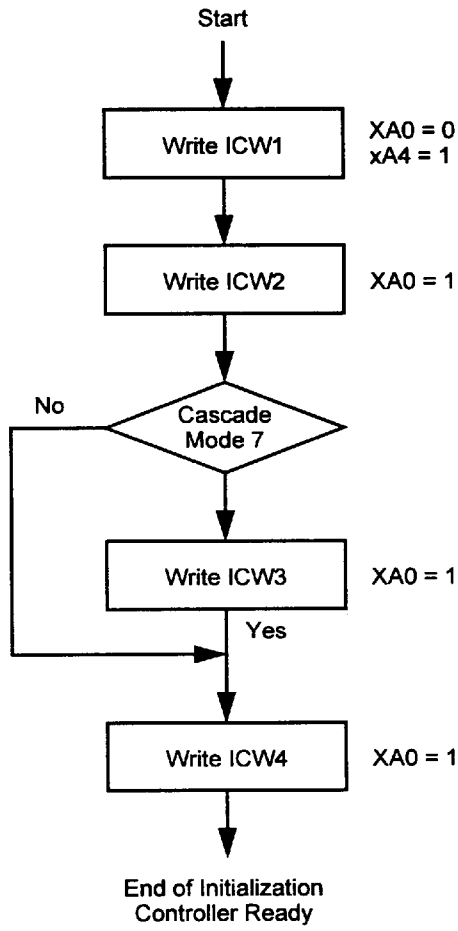
4.11.5.6.1 Initialization Command Words (ICWs)

The initialization process consists of writing a sequence of four bytes to each interrupt controller. The initialization sequence is started by writing the first Initialization Command Word (ICW1) to address 020h (0A0h) with a 1 on bit 4 of the data byte. The interrupt controller interprets this as the start of an initialization sequence and does the following:

1) The Initialization Command Word Counter is reset to 0.
2) ICW1 is latched into the device.
3) Fixed Priority Mode is selected.
4) IR0 is assigned the highest priority.
5) The Interrupt Mask Register is cleared.
6) The Slave Mode Address is set to 7.
7) Special Mask Mode is disabled.
8) IRR is selected for status read operations.

The next three I/O writes to address 021h (0A1h) will load ICW2 through ICW4. See Figure 4-40 for a flow chart of the initialization sequence. The initialization sequence can be terminated at any point (all four bytes must be written for the controller to be properly initialized) by writing to address 020h (0A0h) with a 0 in data bit 4. Note this will cause OCW2 or OCW3 to be written.

Figure 4-40 Initialization Sequence



4.11.5.6.2 Operational Command Words (OCWs)

Operational Command Words (OCWs) allow the 82C206's interrupt controllers to be controlled or reconfigured at any time while operating. Each interrupt has three OCWs which can be programmed to affect the proper operating configuration and a status register to monitor controller operation.

OCW1 is located at address 021h (0A1h) and may be written any time the controller is not in the Initialization Mode. OCW2 and OCW3 are located at address 020h (0A0h). Writing to address 020h (0A0h) with a 0 in bit 4 will place the controller in the operating mode and load OCW2 (if data bit 3 = 0) or OCW3 (if data bit 3 = 1).

4.11.5.6.3 IRR, ISR, & Poll Vector

IRR, ISR, and Poll Vector are the same address, 020h (0A0h). The selection of the registers depends on the programming of ITC. If the latest OCW3 issued the poll command (PM = 1), the poll vector is selected for the next read. Before another poll command is issued, subsequent reads to the address will select IRR or ISR depending on the latest OCW3, if RR = 1 and RIS = 0, ISR is selected. Note that the poll command is cleared after the first read to the ITC. After initialization (ICW1 or reset), IRR is selected.

4.11.6 Counter/Timer Subsystem

The 82C206 contains an 8254 compatible counter/timer. The counter/timer can be used to generate accurate time delays under software control. It contains three 16-bit counters (Counters 2 through 0) which can be programmed to count in binary or binary-coded decimal (BCD). Each counter operates independently of the other and can be programmed for operation as a timer or a counter.

All counters in this subsystem are controlled by a common control logic as shown in Figure 4-41. The control logic decodes and generates the necessary commands to load, read, configure, and control each counter. Counter 0 and Counter 1 can be programmed for all six modes, but Mode 1 and Mode 5 have limited usefulness because their gate is hard-wired to GND internally. Counter 2 can be programmed to operate in any of the six modes:

- Mode 0 - Interrupt on terminal count
- Mode 1 - Hardware retriggerable one-shot
- Mode 2 - Rate generator
- Mode 3 - Square wave generator
- Mode 4 - Software triggered strobe
- Mode 5 - Hardware retriggerable strobe

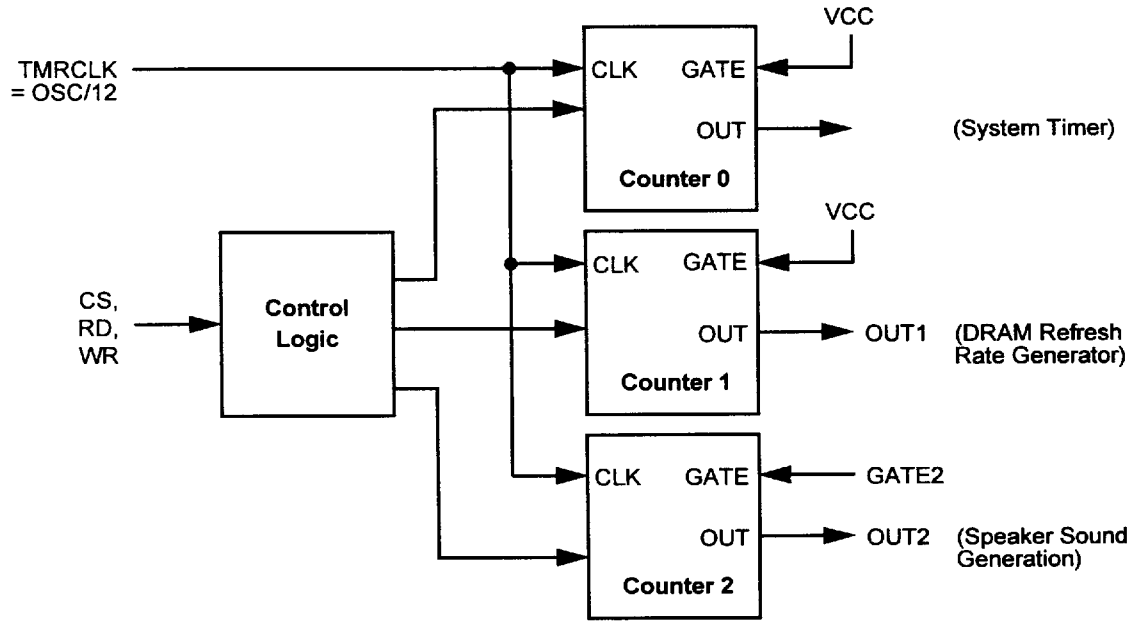
The internal timer counter use an internal signal TMRCLK which is derived from the OSC input of the 82C206. For the sake of simplicity, all references to the timer counter clock will be TMRCLK in the following description. All three counters are driven from a common clock input, TMRCLK (TMRCLK = OSC/12) Counter 0's output (OUT0) is internally connected to IRQ of INTC1 and is used as an interrupt to the system for time keeping and task switching. Counter 1 may be programmed to generate pulses or square waves for external devices. Counter 2 is a full function counter/timer. It can be used as an interval timer, a counter, or as a gated rate/pulse generator. In a PC/AT compatible design, Counter 0 is used as a system timer, Counter 1 is used as a DRAM refresh rate generator, and Counter 2 is used for speaker sound generation.

4.11.6.1 Counter Description

Each counter in this subsystem contains a control register, a status register, a 16-bit counting component, a pair of 8-bit counter input latches, and a pair of 8-bit counter output latches. Each counter shares the same clock input (TMRCLK). GATE0, GATE1, and OUT0 are not externally accessible. This is fully compatible with a PC/AT-based design. Output of OUT0 is dependent on the counter mode.

The control register stores the mode and command information used to control the counter. It may be loaded by writing a byte to the write control word at Port 043h. The status register allows the software to monitor counter conditions and read back the contents of the control register.

Figure 4-41 Counter/Timer Block Diagram



The 16-bit counting component is a loadable synchronous down counter. It is loaded or decremented on the falling edge of TMRCLK. The counting component contains a maximum count when a 0 is loaded, which is equivalent to 65536 in binary operation or 1000 in BCD. The counting component does not stop when it reaches 0. In Modes 2 and 3, the counting component will be reloaded and in all other modes it will wrap around to 0FFFFh in binary operation or 9999 in BCD.

The counting component is indirectly loaded by writing one or two bytes (optional) to the counter input latches, which are in turn loaded into the counting component. Thus, the counting component can be loaded or reloaded in one TMRCLK cycle. The counting component is also read indirectly by reading the contents of the counter output latches. The counter output latches are transparent latches which can be read while transparent or latched (see Latch Counter Command).

4.11.6.1.1 Programming the Counter/Timer

After a system reset, the contents of the control registers, counter registers, counting components, and the output of all counters are undefined. Each counter must be programmed before it can be used. Each counter is programmed by writing its control register with a control word and then giving an initial count to its counting component. Table 4-22 lists the I/O address map used by the counter/timer subsystem.

Table 4-22 Counter/Timer I/O Address Map

Address	Function
040h	Counter 0 read/write
041h	Counter 1 read/write
042h	Counter 2 read/write
043h	Control register write only

4.11.6.1.2 Read/Write Counter Command

Each counter has a write only control register. This control register is written with a control word to the I/O address 043h.

When programming to a counter, the following steps must sequentially occur:

1) Each counter's control register must be written with a control word before the initial count is written.
2) Writing the initial count must follow the format specified in the control word (least significant bit only, most significant bit only, or least significant bit and then most significant bit.

A new initial count can be written into the counter at any time after programming without rewriting the control word.

4.11.6.1.3 Counter Latch Command

When a counter latch command is issued, the counter's output latches latch the current state of the counting component. The counter's output latches remain latched until read by the CPU or the counter is reprogrammed. After that, the output latches then returns to a "transparent" condition. Counter latch commands may be issued to more than one counter before reading the first counter to which this command was issued. Also, multiple counter latch commands issued to the same counter without reading the counter will cause all but the first command to be ignored.

4.11.6.1.4 Read-Back Command

The read-back command allows the user to check the count value, mode, and state of the OUT signal and null count flag of the selected counter(s).

Each counter's latches remain latched until either the latch is read or the counter is reprogrammed. If both LSTATUS and LCOUNT are 0, the status will be returned on the next read from the counter. The next one or two reads (depending on whether the counter is programmed to transfer one or two bytes) from the counter result in the count being returned. Multiple read-back commands issued to the same counter without reading the counter will cause all but the first command to be ignored.

4.11.6.2 Counter Operation

Since Counter 1 and 0 have limitations in some of their operation modes, Counter 2 will be used to describe the various counter operating modes. However, the description of Modes 0, 2, 3, and 4 are suitable for all counters. The following terms are defined for describing the counter/timer operation.

- TMRCLK pulse - A rising edge followed by a falling edge of the 82C206's TMRCLK (OSC/12).
- Trigger - The rising edge of the GATE2 input.
- Counter Load - the transfer of the 16-bit value in counter input latches to the counting element.
- Initialized - A control word written and the counter input latches loaded.
- Counter 2 can operate in one of the following modes:
  - Mode 0 - Interrupt on terminal count
  - Mode 1 - Hardware retriggerable one-shot
  - Mode 2 - Rate generator
  - Mode 3 - Square wave generator
  - Mode 4 - Software triggered strobe
  - Mode 5 - Hardware triggered strobe

4.11.6.2.1 Mode 0 - Interrupt on Terminal Count

Mode 0 is usually used for event counting. After a counter is written with the control word, OUT2 of that counter goes low and remains low until the counting element reaches 0, at which time it goes back high and remains high until a new count or control word is written. Counting is enabled when GATE2 = 1 and disabled when GATE2 = 0. GATE2 has no effect on OUT2.

The counting component is loaded at the first TMRCLK pulse after the control word and initial count are loaded. When both initial count bytes are required, the counting component is loaded after the high byte is written. This TMRCLK pulse does not decrement the count, so for an initial count of N, OUT2 does not go high until (N + 1) TMRCLK pulses after initialization. Writing a new initial count to the counter reloads the counting element on the next TMRCLK pulse and counting continues from the new count. If an initial count is written with GATE2 = 0, it will still be loaded on the next TMRCLK pulse. But counting does not progress until GATE2 = 1. When GATE2 goes high, OUT2 will go high after N TMRCLK pulses later.

4.11.6.2.2 Mode 1 - Hardware Retriggerable One-Shot

Writing the control word causes OUT2 to go high initially. Once initialized, the counter is armed and a trigger causes OUT2 to go low on the next TMRCLK pulse. OUT2 then remains low until the counter reaches 0. An initial count of N results in a one-shot pulse N TMRCLK cycles long. Any subsequent triggers while OUT2 is low cause the counting component to be reloaded, extending the length of the pulse. Writing a new count to the counter input latches will not affect the current one-shot pulse unless the counter is retriggered. In the latter case, the counting component is loaded with the new count and the one-shot pulse continues until the new count expires.

4.11.6.2.3 Mode 2 - Rate Generator

This mode functions as a divide-by-N counter. After writing the control word during initialization, the counter's OUT2 is set to high. When the initial count is decremented to 1, OUT2 goes low on the next TMRCLK pulse. The following TMRCLK pulse returns OUT2 high, reloads the CE, and the process is repeated. In Mode 2, the counter continues counting (if GATE2 = 1) and will generate an OUT2 pulse every N TMRCLK cycles. Note that a count of 1 is illegal in Mode 2.

GATE2 = 0 disables counting and forces OUT2 high immediately. A trigger reloads the CE on the next TMRCLK pulse. Thus, GATE 2 can be used to synchronize the counter to external events.

Writing a new count while counting does not affect current operation unless a trigger is received. Otherwise, the new count will be loaded at the end of the current counting cycle.

4.11.6.2.4 Mode 3 - Square Wave Generator

Mode 3 is similar to Mode 2 in every respect except for the duty cycle of OUT2. OUT2 is set high initially and remains high for the first half of the count. When the first half of the initial count expires, OUT2 goes low for the remainder of the count.

If the counter is loaded with an even count, the duty cycle of OUT2 will be 50% (high = low = N/2). For odd count values, OUT2 is high one TMRCLK cycle longer than it is low. Therefore, high = (N + 1)/2 and low = (N - 1)/2.

4.11.6.2.5 Mode 4 - Software Triggered Strobe

Writing the Control Word causes OUT2 To go high initially. Expiration of the initial count causes OUT2 to go low for one TMRCLK cycle. GATE2 = 0 disables counting but has no effect on OUT2. Also, a trigger will not reload CE.

The counting sequence is started by writing the initial count. The CE is loaded on the TMRCLK pulse after initialization. The CE begins decrementing one TMRCLK pulse later, OUT2 will go low for one TMRCLK cycle, (N + 1) cycles after the initial count is written.

If a new initial count is written during a counting sequence, it is loaded into the CE on the next TMRCLK pulse and the sequence continues from the new count. This allows the sequence to be "retriggerable" by software.

4.11.6.2.6 Mode 5 - Hardware Triggered Strobe

Writing the Control Word causes OUT2 to go high initially. Counting is started by a trigger. The expiration of the initial count causes OUT2 to go low for one TMRCLK cycle. GATE2 = 0 disables counting.

The CE is loaded on the TMRCLK pulse after a trigger. Since loading the CE inhibits decrementing, OUT2 will go low for one TMRCLK cycle, (N + 1) TMRCLK cycles after the trigger.

If a new count is loaded during counting, the current counting sequence will not be affected unless a trigger occurs. A trigger causes the counter to be reloaded from CIL and CIH making the counter "retriggerable".

4.11.6.2.7 GATE2

In Modes 0, 2, 3, and 4 GATE2 is level-edge sensitive and is sampled on the rising edge of TMRCLK. In Modes 1, 2, 3, and 5 the GATE2 input is rising-edge sensitive. This rising edge sets an internal flip-flop whose output is sampled on the rising edge of TMRCLK. The flip-flop resets immediately after being sampled. Note that in Modes 2 and 3, the GATE2 input is both edge and level sensitive. Table 4-23 details this operation.

Table 4-23 GATE2 Pin Function

Mode	GATE2		
	Low	Rising	High
0	Disables counting		Enables counting
		A) Initiates counting B) Reset OUT2 pin	
2	A) Disables counting B) Forces OUT2 pin high	Initiates counting	Enables counting
3	A) Disables counting B) Forces OUT2 pin high	Initiates counting	Enables counting
4	Disables counting		Enables counting
5		Initiates counting	

4.12 Power Management

The Viper-DP Desktop Chipset supplies an optimum GREEN solution by providing a green power management port for controlling desktop subsystems which includes clock control to the CPU's clock, STPCLK# signal to the CPU, and monitoring shutdown.

The Viper-DP Desktop Chipset has a Green Event Timer (GET) used to activate the AUTO_GREEN or SMI_GREEN modes. The GET can be reloaded by any IRQ, PCI bus requests, DMA requests, keyboard, video, hard disk and floppy accesses, programmable I/O subsystem activity and optional external sources.

The AUTO_GREEN mode is available for dynamic CPUs which do not support the SMI protocol. The SMI_GREEN mode enables a much higher degree of software control for GREEN capabilities. The SMI_GREEN mode can only be utilized in systems that support the SMI# protocol.

4.12.1 Definition of Power Management Modes

The following subsections will define the various power management modes used when configuring systems with the Viper-DP Desktop Chipset to run in the AUTO_GREEN and SMI_GREEN modes.

4.12.1.1 Normal Mode

In this mode, the system is running at full speed. No power management features have been activated.

4.12.1.2 Auto Green Mode

This mode is used to accommodate non-SL Enhanced CPUs. It allows for power management through hardware control. The AUTO_GREEN mode is entered when any enabled GREEN event occurs. When any enabled GREEN event occurs, the power control information in Index EAh is put out onto the SD[7:0] bus and the external power control latch is pulsed to match this value. The system can resume out of the AUTO_GREEN mode by any wake-up event that has been enabled in the power management registers. When a wake-up event takes place, the power control information in Index EBh is put out onto the SD[7:0] bus and the external power control latch is pulsed to latch this value. While returning to the NORMAL mode, the CPU clock first runs at full speed for 20ms before the AT bus clock is switched back to the synchronous mode.

4.12.1.3 SMI_GREEN Mode

The SMI_GREEN mode is used to accommodate SMI supported CPUs. It allows power management through the SMI# protocol. When any event, that has been configured to generate a SMI#, occurs then a SMI# is generated from the Viper-DP Desktop Chipset to the CPU. In response, the CPU saves the state of all its internal registers, asserts SMIACT# to the Viper-DP Desktop Chipset and then begins executing the SMI code. In the SMI code, the 82C558's Index ECh port can be directly written to the external power control latch, thus allowing power management through hardware too.

The system can resume out of the SMI_GREEN mode by any enabled wake-up event programmed in the power management register. During this Resume state, the system can be allowed to return to the NORMAL mode. The CPU clock first runs at full speed for 20ms before the AT bus clock is switched back to the synchronous mode.

4.12.2 System Activity Detection

4.12.2.1 GREEN Events

The following is a list of events that can be programmed to be GREEN.

- GREEN Event Timer (GET) time-out
- EPMI# trigger
- Software bit trigger - This is a bit in the system power management registers which if set, causes a GREEN event
- Device Timer time-out - There are two general purpose timers that can be programmed (in Index F0h-F7h) to monitor user specified address locations. When either of them times out, a GREEN event is generated.

4.12.2.2 Reload GET/Wake-up Events

Any of the following events, if enabled, will cause the GREEN Event Timer to reload its initial count from Index E2h. These events, if enabled, will cause the system to generate hardware PPWRL# (AUTO_GREEN mode) or SMI# (SMI_GREEN Mode) or both.

- All IRQs (except IRQ2)
- One programmable IO/MEM range access
- PREQ# signals from the PCI bus
- All DREQs (except DREQ4)
- Keyboard access:
  - I/O Ports 60h and 64h
- Video access:
  - 0A0000-0BFFFF address trap (graphics buffer)
  - I/O Port 3B0h-3DFh (VGA command registers)
- Hard/floppy disk access:
  - I/O Port 1F0h-1F7h and/or 3F6h, 170h-177h (hard disk)
  - I/O Port 3F5h (floppy)
- COM Ports:
  - COM1/3: COM1 (3F8-3FF) and COM3 (3E8-3EF)
  - COM2/4: COM2 (2F8-2FF) and COM4 (2E8-2EF)
- LPT Ports:
  - LPT1 (3B0-3BF), LPT2 (378-37F), and LPT3 (278-27F)
- External EPMI source

4.12.3 System Management Interrupt (SMI)

Most modern processors offer a System Management Interrupt (SMI) that allows external logic to signal to the CPU that a high-priority event has occurred and must be serviced but should not in any way interfere with the application currently being processed. When the CPU senses its SMI input active, it saves the context of its current application and loads the context of its System Management Mode (SMM) handler routine from a protected part of RAM. SMM code can then proceed to determine the reason for the interrupt, service it appropriately, and return to application processing through a special RESUME instruction that restores the context as it originally was before the SMI. Entry to and exit from SMM is completely hardware-controlled.

4.12.3.1 SMI Implementation

During the NORMAL mode of operation, CPU accesses in the A0000h-BFFFFh are diverted to the AT bus or the local bus. During SMM, the SMIACT# signal is used for recognizing SMM addresses and these addresses are always mapped to the A0000h-BFFFFh range in DRAM (which is initialized with SMM code during boot-up). It is not required to flush the cache before executing SMM code due to the following reasons:

- The A0000-BFFFFh range is always made non-cacheable whether the CPU is in SMM mode or not.
- The CPU's SMBASE Register is always initialized to A0000h.

If the CPU's SMBASE Register were programmed with an address other than an address in the A0000h-BFFFFh range, it would be necessary to flush the cache.

4.12.4 Hardware Power Management Support without an External Latch

The Viper-DP Desktop Chipset can support hardware power management through an external latch or internally through programmable pins on the 82C558 IPC. At power-on reset, if the XDIR is sampled high, then hardware power management is done through an external latch. If XDIR is sampled low, then the PPWRL# and MP4,GPCS0# pins on the IPC change functionality to support hardware power management without a latch.

The PPWRL# pin becomes PPWRL1 and the MP4,GPCS0# becomes PPWRL2. With this implementation, the IPC will not be able to generate memory parity for PCI master writes. Hence, maximum performance for PCI master writes will be sacrificed.

4.13 I/O APIC Interface

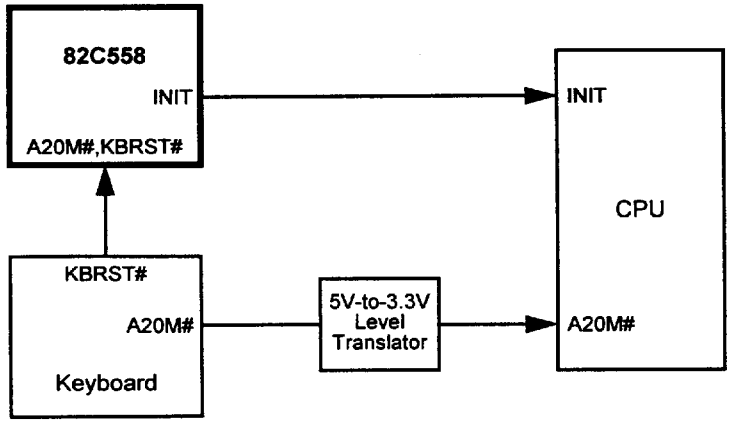
The Viper-DP Desktop Chipset supports dual processor implementation with the aid of an I/O APIC controller, which resides in the 82C558 (IPC). During power-up reset, if the strap on the INTR/LINT0/APICEN is sampled high, then the I/O APIC interface is enabled. The I/O APIC controller captures all the system interrupts and routes them to the appropriate processor over the APIC bus. The APIC bus consists of the following signals - the data lines PICD0 and PICD1, the local interrupt lines LINT0 and LINT1, and the APIC clock. The local interrupt lines LINT0 and LINT1 are multiplexed with the INTR and the NMI signals respectively, and the data lines PICD0 and PICD1 are pin multiplexed with PIRQ0# and PIRQ1# respectively. If the APIC enable strap is sampled high on power-up reset, then these lines take on the APIC bus functions.

4.14 Fast GATEA20 and Reset Emulation

The SYSC will intercept commands to Ports 60h and 64h so that it can emulate the keyboard controller, allowing the generation of the fast GATEA20 and fast INIT signals. The decode sequence is software transparent and requires no BIOS modifications to function. The fast GATEA20 generation sequence involves writing "D1h" to Port 64h, then writing data "02h" to Port 60h. The fast CPU "warm reset" function is generated when a Port 64h write cycle with data "FEh" is decoded. A write to Port 64h with data "D0h" will enable the status of GATEA20 (bit 1 of Port 60h) and the warm reset (bit 0 of Port 60h) to be readable.

If keyboard emulation is disabled, (i.e., Index 41h[12] =1 in the IPC system control register space) then the keyboard has to generate the GATEA20 and KBRST signals. In this case, the A20M#,KBRST# signal on the IPC functions as a KBRST# input signal. It samples the KBRST output from the keyboard and generates an INIT to the CPU. The keyboard GATEA20 signal should be connected to the CPU A20M# input through a voltage translator. The keyboard GATEA20# signal is a 5V output signal and the CPU A20M# signal is a 3.3V input signal. Figure 4-42 shows the connectivity when keyboard emulation has been disabled.

Figure 4-42 Connections with Keyboard Emulation disabled



4.15 Shadow ROM and BIOS Cacheability

When using the Viper-DP Desktop Chipset, the procedures listed below should be followed for proper setup and configuration of shadow RAM utilities.

1. Enable ROMCS# generation for the segment to be shadowed. Although the F0000h-FFFFFh segment defaults to ROMCS# generation, the C,D and E0000h ROM segments must have ROMCS# generation enabled by setting the appropriate bits in Address Offsets 4B-4Ah of the IPC system control registers.
2. Enable ROM contents to be copied into DRAM. To do this, the appropriate bits in Indices 04h, 05h, and 06h in the SYSC system control register space should be set. These bits must be set so that reads from these segments will be executed out of ROM but will be written to DRAM.
3. Enable shadow RAM areas to permit DRAM read/write accesses. At this point, the ROMCS# generation bits that were previously necessary to access the original ROM code, must be disabled.
4. Write protect shadow RAM areas. To do this, the appropriate bits in Indices 04h, 05h, and 06h in the SYSC system control register space should be set. These bits must be set so that reads from these segments will be executed out of DRAM, but writes will be directed to the ROM.
5. Cache shadow RAM areas in L2/L1 caches (optional). Caching of the individual code segments can be accomplished by setting the appropriate bits in Index 06h in the SYSC system control register space. Although write protection control for the L2 cache is provided, the L1 cache does not have a write protection mechanism and the ROM code may be overwritten or modified if stored in the L1 cache.

4.16 Special Cycles

4.16.1 System ROM BIOS Cycles

The 82C557 supports both 8- and 16-bit EPROM cycles. If the system BIOS is 16 bits wide, ROMCS# should be connected to MEMCS16# through an open collector gate indicating to the IPC that a 16-bit EPROM is responding. The system BIOS resides on the XD bus.

ROMCS# is generated for both the E0000h-EFFFFh and F0000h-FFFFFh segments. If a combined video/system ROM BIOS is desired, these two segments should be used.

4.16.2 System Shutdown/Halt Cycles

The CPU provides special bus cycles to indicate that certain instructions have been executed or certain conditions have occurred internally. These special cycles, such as shutdown and halt, are covered by dedicated handling logic inside the SYSC. The Viper-DP Desktop Chipset will generate INIT for a CPU shutdown cycle.

5.0 Register Descriptions

There are three broad classes of configuration registers in the Viper-DP Desktop Chipset; (i) PCI Configuration Registers, (ii) System Control Registers, and (iii) Power Management Registers. The PCI Configuration Registers 00h-3Fh are mandatory and exist on both the SYSC and the IPC. The System Control Registers are distributed between the SYSC and the IPC. The ones that are in the SYSC are accessed by an indexing scheme, whereas the ones in the IPC are accessed via the same PCI configuration mechanism used to access the PCI Configuration Registers of the IPC. The Power Management Registers are located in the IPC and are accessed by the same indexing scheme used to access the System Control Registers of the SYSC. Table 5-1 summarizes the locations and access mechanisms for all the registers.

Table 5-1 Register Locations and Access Mechanisms Summary

Parameter	PCI Specific Configuration Registers		System Control Registers		PMU Registers
Location	SYSC (00h-3Fh)	IPC (00h-3Fh)	SYSC (Index 00h-19h)	IPC (40h-51h)	IPC (Index E0h-FFh)
Classification	SYSC PCI Registers (00h-3Fh)	IPC PCI Registers (00h-3Fh)	SYSC System Control Registers (Index 00h-19h)	IPC System Control Registers (40h-51h)	IPC (Index E0h-FFh)
Access Mechanism	PCI Configuration Mechanism #1, Bus #0, Device #0, Function #0	PCI Configuration Mechanism #1, Bus #0, Device #1, Function #0	Indexed Method: Index loaded in 22h, Data to/from index through 24h	PCI Configuration Mechanism #1, Bus #0, Device #1, Function #0	Indexed Method: Index loaded in 22h, Data to/from index through 24h
Reference Section	Section 5.1.1	Section 5.2.1	Section 5.1.2	Section 5.2.2	Section 5.2.5

5.1 82C557 SYSC Register Space

The 82C557 has two register spaces - the PCI Configuration Space and the System Control Register Space.

5.1.1 82C557 SYSC PCI Configuration Register Space

The 82C557 SYSC's PCI Configuration Registers are accessed by Configuration Mechanism #1, on Bus #0, Device #0, Function #0.

Notes: In the address offsets given below, the most significant bit (MSB) corresponds to the upper address offset.
RO = Read Only, R/W = Read/Write, and WO = Write Only.

Table 5-2 57VIN, 82C557 Vendor Identification Register - Address Offset 01h-00h

Bit(s)	Type	Default	Function
15:0	RO	1045h	Vendor Identification Register

Table 5-3 57DID, 82C557 Device Identification Register - Address Offset 03h-02h

Bit(s)	Type	Default	Function
15:0	RO	C557h	Device Identification Register

Table 5-4 57COM, 82C557 Command Register - Address Offset 05h-04h

Bit(s)	Type	Default	Function
15:10	R/W	0000 00	Reserved Bits: Must be written to 0.
9	R/W	0	Enable fast back-to-back to different slaves: 0 = Disable 1 = Enable
8	RO	0	SERR# output pin: 0 = Disable always
7	RO	0	Address/data stepping: 0 = Disable always
6	R/W	0	PERR# output pin: 0 = Disable always
5	R/W	0	Reserved Bit: Must be written to 0.
4	RO	0	Memory write and invalidate cycle generation: Must always = 0. No memory write and invalidate cycles will be generated by the SYSC.
3	RO	0	Special cycles: Must always = 0. The SYSC does not respond to the PCI special cycle.
2	RO	1	Bus master operations: Must always = 1. This allows the SYSC to perform bus master operations at any time.
1	RO	1	Memory access: Must always = 1. The SYSC allows a PCI bus master access to memory at anytime.
0	RO	1	I/O access: Must always = 1. The SYSC allows a PCI bus master I/O access at any time.

Table 5-5 57STAT, 82C557 Status Register - Address Offset 07h-06h

Bit(s)	Type	Default	Function
15	RO	0	Detected parity error: Must always = 0.
14	RO	0	SERR# status: Must always = 0.
13	RO	0	Master abort status: Must always = 0.
12	RO	0	Received target abort status: 0 = No target abort 1 = Target abort occurred
11	RO	0	Signaled target abort status: Must be 0 always
10:9	RO	01	DEVSEL# timing status: These bits must always = 01 and indicates medium timing selection. The SYSC asserts the DEVSEL# based on medium timing.
8	RO	0	Data parity detected: Must always = 0.
7	RO	1	Fast back-to-back capability: 1 = Capable 0 = Not Capable
6:0	R/W	0000 000	Reserved Bits: Must be written to 0.

Table 5-6 57REV, 82C557 Revision Identification Register - Address Offset 08h

Bit(s)	Type	Default	Function
7:0	RO	00h	Revision Identification Register

Table 5-7 57CLASS, 82C557 Class Code Register - Address Offset 0Bh-09h

Bit(s)	Type	Default	Function
23:0	RO	060000h	Class Code

Table 5-8 57RESV1, 82C557 Reserved Register 1 - Address Offset 0Ch

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved Bits: Must be written to 0.

Table 5-9 57MLTMR, 82C557 Master Latency Timer Register - Address Offset 0Dh

Bit(s)	Type	Default	Function
7:0	RO	00h	Master Latency Timer Register

Table 5-10 57HEAD, 82C557 Header Type Register - Address Offset 0Eh

Bit(s)	Type	Default	Function
7:0	RO	00h	Header Type

Table 5-11 57BIST, 82C557 Built-In Self-Test Register, Address Offset 0Fh

Bit(s)	Type	Default	Function
7:0	RO	00h	BIST

Table 5-12 57RESV2, 82C557 Reserved Register 2 - Address Offset 3Fh-10h

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved Bits: Must be written to 0.

Table 5-13 57MEMCTL, 82C557 Memory Control Register - Address Offset 43h-40h

Bit(s)	Type	Default	Function
31:16	R/W	0000h	Reserved Bits: Must be written to 0.
15:6	R/W	00h	VL video frame buffer write posting hole: These bits map onto address bits A[31:22] to define the 4MB window where write posting can be masked.
5:3	R/W	0h	Reserved Bits: Must be written to 0.
2	R/W	0	Write posting hole: The write posting hole defined by bits [15:6], can be enabled/disabled by this bit. 0 = Enable 1 = Disable
1	R/W	0	Video memory (A0000h-BFFFFh) write post control: 0 = Enable 1= Disable
0	R/W	0	I/O cycle write post control: 0 = Disable 1 = Enable

5.1.2 82C557 SYSC System Control Register Space

An indexing scheme is used to access all of these registers. Port 22h is used as the Index Register and Port 24h is the Data Register. Each access to a configuration register within this space consists of a write to Port 22h, specifying the desired register in the data byte, followed by a read or write to Port 24h with the actual register data. The index resets after every access; so every data access (via Port 24h) must be preceded by a write to Port 22h even if the same register is being accessed consecutively. Index 23h is the Data Register for DMA Clock Select.

Note: All reserved bits are set to zero by default and must be set to zero for future purposes.
Table 5-14 and Table 5-15 represent the same register. The setting of bit 7 at Index 13h decides on the functionality of this register. If 13h[7] = 1, then Table 5-13 is valid for Index 00h. If 13h[7] = 0, then Table 5-14 is valid for Index 00h.

Table 5-14 57CMOD, 82C557 Byte Merge/Prefetch and Sony Cache Module Control Register - Index 00h

Bit(s)	Type	Default	Function
7	R/W	0	Enable pipelining of single CPU cycles to memory: 0 = Disable 1 = Enable
6	R/W	0	Video memory byte/word read prefetch enable: This setting enables/disables the prefetching of bytes/words/from PCI video memory by the CPU. 0 = Disable 1 = Enable
5	R/W	0	Sony SONIC-2WP support enable: If this bit is set, then ensure that the L2 cache has been disabled (i.e., set Index 02h[3:2] = 00). 0 = No Sony SONIC-2WP installed 1= Sony SONIC-2WP installed
4	R/W	0	Byte/word merge support: 0 = Disable 1 = Enable
3	R/W	0	Byte/word merging with CPU pipelining (NA# generation) support: 0 = Disable 1 = Enable
2:1	R/W	00	Time-out counter for byte/word merge: This setting determines the maximum time difference between two consecutive PCI byte/word writes to allow merging. 00 = 4 CPU CLKs 10 = 12 CPU CLKs 01 = 8 CPU CLKs 11 = 16 CPU CLKs
0	R/W	0	Enable internal hold requests to be blocked while performing byte merge: 0 = Disable 1 = Enable

Table 5-15 57PYTHON, 82C557 82C546/82C547 Compatible DRAM Configuration Register 1 - Index 00h

Bit(s)	Type	Default	Function
7	R/W	0	Enable pipelining of single CPU cycles to memory: 0 = Disable 1 = Enable
6	R/W	0	Second bank: 512x36 SIMM - These SIMMs need to be single sided: 0 = Single sided SIMM not installed in bank 1 1 = Single sided SIMM installed in bank 1
5	R/W	0	First bank: 512x36 SIMM - These SIMMS need to be single sided: 0 = Single sided SIMM not installed in bank 0 1 = Single sided SIMM installed in bank 0
4:0	R/W	0000 0	Memory configurations select (default = 2MB) - These settings are detailed in Table 5-36. **Note:** These configurations maintain backward compatibility with the 82C546/82C547 chipset, and they do not allow the user too much flexibility. For 100% flexibility, bit 7 of the register at Index 13h should be set to 1.

Table 5-16 57DRMCTL, 82C557 DRAM Control Register 1 - Index 01h

Bit(s)	Type	Default	Function
7	R/W	0	Row address hold after RAS# active in CLKs: 0 = 2 CLKs 1 = 1 CLK
6	R/W	0	RAS# active/inactive on entering master mode: 0 = Normal page mode when starting a master cycle. In this case, RAS# will remain active when starting a master cycle. 1 = RAS# inactive when starting a master cycle.
5:4	R/W	00	RAS pulse width used during refresh in CLKs: 00 = 7 CLKs 10 = 5 CLKs 01 = 6 CLKs 11 = 4 CLKs
3	R/W	0	CAS pulse width during reads in CLKs: 0 = 3 CLKs 1 = 2 CLKs
2	R/W	0	CAS pulse width during writes in CLKs: 0 = 3 CLKs 1 = 2 CLKs
1:0	R/W	00	RAS precharge time in CLKs: 00 = 6 CLKs 10 = 4 CLKs 01 = 5 CLKs 11 = 3 CLKs

Table 5-17 57CHCTL1, 82C557 Cache Control Register 1 - Index 02h

Bit(s)	Type	Default	Function
7:6	R/W	00	Cache size selection: This setting determines the size of the L2 cache. **If 0Fh[0] = 0:** / **If 0Fh[0] = 1** 00 = 64K / 00 = 1MB 01 = 128K / 01 = 2MB 10 = 256K / 10 = Reserved 11 = 512K / 11 = Reserved
5:4	R/W	00	Cache write policy: This setting determines the write policy for the L2 cache. 00 = L2 cache write-through 01 = Adaptive Write-back Mode 1(see Section 4.3.3.5.1) 10 = Adaptive Write-back Mode 2 (see Section 4.3.3.5.2) 11 = L2 cache write-back
3:2	R/W	00	Cache mode select: This setting determines the operating mode of the L2 cache. 00 = Disable 01 = Test Mode 1; External Tag Write (Tag data write-through Index Register 07h) 10 = Test Mode 2; External Tag Read (Tag data read from Index Register 07h) 11 = Enable L2 cache
1	R/W	0	DRAM posted write: 0 = Disable posted write / 1 = Enable posted write
0	R/W	0	CAS precharge time in CLKs: 0 = 2 CLKs / 1 = 1 CLK

Table 5-18 57CHCTL2, 82C557 Cache Control Register 2 - Index 03h

Bit(s)	Type	Default	Function
7:6	R/W	00	Cache write burst mode CLKs: This setting determines the timing for burst writes to the L2 cache. 00 = X-4-4-4 / 10 = X-2-2-2 01 = X-3-3-3 / 11 = X-1-1-1
5:4	R/W	00	Leadoff cycle for cache writes: This setting determines the leadoff time for writes to the L2 cache. 00 = 5-X-X-X / 10 = 3-X-X-X 01 = 4-X-X-X / 11 = 2-X-X-X* *Sync. SRAM double bank implementation does not support this timing
3:2	R/W	00	Cache read burst mode CLKs: This setting determines the timing for burst reads to the L2 cache. 00 = X-4-4-4 / 10 = X-2-2-2 01 = X-3-3-3 / 11 = X-1-1-1
1:0	R/W	00	Leadoff cycle for cache reads: This setting determines the leadoff time for reads to the L2 cache. 00 = 5-X-X-X / 10 = 3-X-X-X 01 = 4-X-X-X / 11 = 2-X-X-X* *Sync. SRAM double bank implementation does not support this timing

Table 5-19 57SHCTL1, 82C557 Shadow RAM Control Register 1 - Index 04h

Bit(s)	Type	Default	Function
7:6	R/W	00	CC000h-CFFFFh read/write control: This setting determines the read/write control for the CC000h-CFFFFh portion of the shadow RAM. 00 = Read/write PCI bus 01 = Read from DRAM / write to PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM
5:4	R/W	00	C8000h-CBFFFh read/write control: 00 = Read/write PCI bus 01 = Read from DRAM / write to PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM
3	R/W	0	Synchronous SRAM pipelined read cycle 1-1-1-1 enable: If Index 11h[3] = 1 (i.e., sync. SRAM chosen) and if Index 03h[3:2] = 11, then this register setting comes into play. 0 = This implies that the leadoff T-state for the read piped cycle = 2 (i.e., it will be a 3-1-1-1 cycle followed by a 2-1-1-1 cycle, or a 3-1-1-1 cycle for successive piped cycles). 1 = This enables the leadoff T-state for the read piped cycle = 1 (i.e., it will be a 3-1-1-1 cycle followed by a 1-1-1-1 cycle for successive piped cycles. This is valid only for a single bank case.)
2	R/W	0	E0000h-EFFFFh range selection: This bit determines whether this region will be treated like the F0000 BIOS area or whether it will always be non-cacheable. 0 = E0000h-EFFFFh area will always be non-cacheable 1 = E0000h-EFFFFh area will be treated like the F0000h BIOS area. If this bit is set then Index 06h[1:0] and Index 06h[3:2] should be set identically.
1:0	R/W	00	C0000h-C7FFFh read/write control: This setting determines the read/write control for the C0000h-C7FFFh portion of the shadow RAM. 00 = Read/write PCI bus 01 = Read from DRAM / write to PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM

Table 5-20 57SHCTL2, 82C557 Shadow RAM Control Register 2 - Index 05h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	DX000h-DXFFFh read/write control: These settings determine the read/write control for the DX000h-DXFFFh segments of the shadow RAM 00 = Read/write PCI bus 01 = Read from DRAM / write to PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM Bits [7:6] = DC000h-DFFFFh Bits [5:4] = D8000h-DBFFFh Bits [3:2] = D4000h-D7FFFh Bits [1:0] = D0000h-D3FFFh

Table 5-21 57SHCTL3, 82C557 Shadow RAM Control Register 3 - Index 06h

Bit(s)	Type	Default	Function
7	R/W	0	DRAM hole in system memory from 80000h-9FFFFh: This setting gives the user the option to have some other device in the address range 80000h-9FFFFh instead of system memory. When this bit is set, the SYSC will not start the system DRAM controller for accesses to this particular address range. 0 = No hole in memory 1 = Enable hole in memory
6	R/W	0	Wait state addition for PCI master snooping: 0 = Do not add a wait state for the cycle access finish to do the snooping 1 = Add a wait state for the cycle access to finish and then do the snooping
5	R/W	0	Range C0000h-C7FFFh cacheability: 0 = Not Cacheable 1 = Cacheable in L1 and L2 (L1 disabled by Index 08h[0])
4	R/W	0	Range F0000h-FFFFFh cacheability: 0 = Not Cacheable 1 = Cacheable in L1 and L2 (L1 disabled by Index 08h[0])
3:2	R/W	00	F0000h-FFFFFh read/write control: This setting determines the read/write control for the F0000h-FFFFFh portion of the shadow RAM 00 = Read/write PCI bus 10 = Read from PCI / write to DRAM 01 = Read from DRAM / write to PCI 11 = Read from DRAM / write to DRAM **Note:** If Index 04h[2] = 1, then the E0000h-EFFFFh read/write control should have the same setting as this.
1:0	R/W	00	E0000h-EFFFFh read/write control: This setting determines the read/write control for the E0000h-EFFFFh portion of the shadow RAM 00 = Read/write PCI bus 10 = Read from PCI / write to DRAM 01 = Read from DRAM / write to PCI 11 = Read from DRAM / write to DRAM

Table 5-22 57TAGTST, 82C557 Tag Test Register - Index 07h

Bit(s)	Type	Default	Function
7:0	R/W	00h	Tag Test Register

Data from this register is written to the tag, if in Test Mode 1 (refer to Index 02h).
Data from the tag is read into this register, if in Test Mode 2 (refer to Index 02h).

Table 5-23 57CPUCH, 82C557 CPU Cache Control Register - Index 08h

Bit(s)	Type	Default	Function
7	R/W	0	L2 cache single/double bank select: This bit determines whether there are one or two banks of L2 cache. 0 = Double bank (If async. SRAM, then the banks are interleaved. If sync. SRAM, they are not interleaved.) 1 = Single bank (non-interleaved)
6	R/W	0	Snoop filtering for bus masters: For a master request if the subsequent read/write is within the same cache line, CPU 'Inquire' cycles are not done until there is a cache line miss (i.e., line comparator not activated for accesses within the same cache line). 0 = Disables snoop filtering 1 = Enables snoop filtering
5	R/W	0	CPU HITM# pin sample timing: 0 = Delay one clock, therefore HITM# sampled on the third rising edge of LCLK after EADS# has been asserted. 1 = Do not delay, therefore HITM# sampled on the second rising edge of LCLK after EADS# has been asserted.
4	R/W	0	Parity checking: If this bit is set, the SYSC keeps PEN# to the IPC asserted for all memory read cycles. If the DBC asserts MPERR# to the IPC then the IPC asserts NMI to the CPU. If this is not set, the assertion of MPERR# will not cause the IPC to assert NMI. 0 = Disable 1 = Enable
3	R/W	0	Tag/Dirty RAM implementation: This bit reflects the Tag/Dirty RAM implementation 0 = Tag and Dirty are on separate chip (i.e., one could be using a separate x1 or x8 SRAM for the Dirty RAM) 1 = Tag and Dirty are on the same chip (i.e., it could be either a x9 or x8 Tag/Dirty RAM)
2	R/W	0	CPU address pipelining: 0 = Disable 1 = Enable
1	R/W	0	L1 cache write-back and write-through control: 0 = Write-through only 1 = Write-back enabled
0	R/W	0	BIOS area cacheability in L1 cache: This bit determines whether the system BIOS area E0000h-FFFFFh (if Index 04h[2] = 1) or F0000h-FFFFFh (if Index 04h[2] = 0), and video BIOS area C0000h-C7FFFh is cacheable in L1 or not. 0 = Cacheable 1 = Non-Cacheable

Table 5-24 57SYSMEM, 82C557 System Memory Function Register - Index 09h

Bit(s)	Type	Default	Function
7:6	R/W	00	DRAM Hole B size: The address for this hole is specified in Index 0Bh[7:0] and Index 0Ch[2:3] 00 = 512KB 01 = 1MB 10 = 2MB 11 = 4MB
5:4	R/W	00	DRAM Hole B control mode: 00 = Disable 01 = Write-through for L1 and L2 cache 10 = Non-cacheable for L1 and L2 cache 11 = Enable hole in DRAM
3:2	R/W	00	DRAM Hole A size: The address for this hole is specified in Index 0Ah[7:0] and Index 0Ch[1:0] 00 = 512KB 01 = 1MB 10 = 2MB 11 = 4MB
1:0	R/W	00	DRAM Hole A control mode: 00 = Disable 01 = Write-through for L1 and L2 cache 10 = Non-cacheable for L1 and L2 cache 11 = Enable hole in DRAM

Table 5-25 57DRAMA, 82C557 DRAM Hole A Address Decode Register 1 - Index 0Ah

Bit(s)	Type	Default	Function
7:0	R/W	00h	DRAM Hole A starting address: These bits along with bits [1:0] of Index 0Ch are used to specify the starting address of DRAM Hole A. These bits - AST[7:0] map onto HA[26:19] lines.

Table 5-26 57DRAMB, 82C557 DRAM Hole B Address Decode Register 2 - Index 0Bh

Bit(s)	Type	Default	Function
7:0	R/W	00h	DRAM Hole B starting address: These bits along with bits [3:2] of Index 0Ch are used to specify the starting address of DRAM Hole B. These bits - BST[7:0] map onto HA[26:19] lines

Table 5-27 57EDMA, 82C557 Extended DMA Register - Index 0Ch

Bit(s)	Type	Default	Function
7	R/W	0	Reserved Bit: Must be written to 0.
6	R/W	0	Fast BRDY# generation for DRAM write page hits: 0 = BRDY# for DRAM writes is generated on the fourth clock 1 = BRDY# for DRAM writes is generated on the third clock
5	R/W	0	HACALE one-half a clock cycle earlier: 0 = HACALE normal timing maintained 1 = HACALE one-half a clock cycle early enabled
4	R/W	0	Wider cache WE# pulse: 0 = Cache WE# pulse width is normal (i.e., ~15ns) 1 = Cache WE# pulse width is made wider (i.e., ~17.5ns)
3:2	R/W	00	DRAM Hole B starting address: These bits are used in conjunction with the bits in Index 0Bh to specify the starting address of DRAM Hole B. These bits BST[9-8] map onto HA[28-27] lines.
1:0	R/W	00	DRAM Hole A starting address: These bits are used in conjunction with the bits in Index 0Ah to specify the starting address of DRAM Hole A. These bits AST[9-8] map onto HA[28-27] lines

Table 5-28 57CLKCTL, 82C557 Clock Control Register - Index 0Dh

Bit(s)	Type	Default	Function
7	R/W	0	Clock source for generating the sync. SRAM timing: 0 = CPU clock is the clock source for generating the sync. SRAM timing and control signals 1 = ECLK is the clock source for generating the sync. SRAM timing and control signals
6	RO	0	The 82C557 sets this bit if the skew between ECLK and CPU CLK is too large. This bit is a read only bit.
5	R/W	0	Auto skew detect: Setting this bit will cause Index 0Dh[4] to be set to 1 automatically if the skew between CLK and ECLK is too large. 0 = No auto detection of the skew between ECLK and CLK takes place. 1 = Automatic detection of the skew between CLK and ECLK takes place and if the skew is too large then Index 0Dh[4] is set to 1.
4	R/W	0	ECLK - CLK skew: If synchronous SRAMs are being used, this bit comes into play. 0 = Skew between CLK and ECLK is not too large 1 = Skew between CLK and ECLK is too large
3	R/W	0	Enable A0000h-BFFFFh as system memory: 0 = A0000h-BFFFFh is not enabled as system memory 1 = A0000h-BFFFFh is enabled as system memory
2	R/W	0	Wait state addition for PCI master doing address toggling in the 486 style: If the PCI master does its address toggling in the style of the Intel 486 burst, rather than a linear burst mode style, then one wait state needs to be added 0 = Linear burst mode style address toggling - no wait state addition 1 = Intel 486 burst style address toggling - one wait state needs to be added
1	R/W	0	PCI cycle claimed by the 82C557 during PCI pre-snoop cycle: If pre-snoop has been enabled, Index 0Fh[7] = 1, then this bit comes into play. During a PCI pre-snoop cycle if the CPU asserts HITM#, then the 82C557 asserts STOP# to the current master. Immediately after doing that, if this bit is set, the 82C557 claims the PCI bus and starts a dummy cycle. 0 = 82C557 does not claim the PCI cycle after it asserts STOP# 1 = 82C557 claims the PCI cycle after it asserts STOP#
0	R/W	0	Slow CPU clock: This bit should be set if the CPU clock frequency has been reduced. 0 = CPU clock frequency is normal 1 = CPU clock has been slowed down

Table 5-29 57CYCTL1, 82C557 Cycle Control Register 1 - Index 0Eh

Bit(s)	Type	Default	Function
7:6	R/W	00	PCI master read burst wait state control: 00 = 4 cycles 10 = 2 cycles 01 = 3 cycles 11 = Reserved
5:4	R/W	00	PCI master write burst wait state control: 00 = 4 cycles 10 = 2 cycles 01 = 3 cycles 11 = Reserved
3	R/W	0	Master cycle parity enable: If Index 08h[4] = 1, then this bit becomes applicable. 0 = Enable parity check during master cycles 1 = Disable parity check during master cycles
2	R/W	0	HACALE timing control: 0 = HACALE high during HITM# before CPU ADS# 1 = HACALE low and CA4 always enabled during HITM cycle, to save external F126
1	R/W	0	Write protection for L1 BIOS: 0 = L1 BIOS not write protected 1 = L1 BIOS write protected
0	R/W	0	PCI line comparator: If Index 08h[6] = 1,then this bit becomes applicable. 0 = Use line comparator in PCI master 1 = Generate inquire cycle for every new FRAME#

Table 5-30 57CYCTL2, 82C557 Cycle Control Register 2 - Index: 0Fh

Bit(s)	Type	Default	Function
7	R/W	0	PCI pre-snoop: Setting this bit enables the PCI pre-snooping feature. 0 = Disable 1 = Enable
6	R/W	0	AT master wait state control: 0 = Do not add any wait states for AT master cycles 1 = Add wait states for AT master cycles
5	R/W	0	Wait state addition for synchronous SRAM even byte access: 0 = Do not add a wait state for a synchronous SRAM even byte access 1 = Add one wait state for a synchronous SRAM even byte access
4	R/W	0	PCI wait state addition for the synchronous SRAM L2 cache implementation: If synchronous SRAMs are being used, then setting this bit enables the conservative mode. 0 = Master does not wait for end of current cycle + CPU-PCI clock to become synchronous 1 = Master waits for end of the current cycle + wait for CPU-PCI clock to become synchronous.
3	R/W	0	Reserved Bit: Must be written to 0.
2	R/W	0	ADSC# generation for synchronous SRAM read cycle: 0 = Generate ADSC# immediately after CPU ADS# goes active 1 = Generate ADSC# one clock after CPU ADS# goes active
1	R/W	0	Reserved Bit
0	R/W	0	Cache size selection: This bit along with Index 02h[1:0] defines the L2 cache size. 0 = Below 1MB 1 = 1MB and above

Table 5-31 57MISC1, 82C557 Miscellaneous Control Register 1 - Index 10h

Bit(s)	Type	Default	Function
7	R/W	0	Early decode of PCI/VL/AT cycle: 0 = CPU to PCI/VL/AT slave cycle triggered after second T2 1 = CPU to PCI/VL/AT slave cycle triggered after first T2
6	R/W	0	Cache modified write cycle timing: 0 = Use the old address changing method (i.e., as in the 82C546/82C547) 1 = Two bank cache, CA4 delayed one-half a clock for write cycles
5	R/W	0	Pipelined read cycle timing: This bit determines the leadoff cycle for a pipelined read 0 = 3-X-X-X read followed by a 3-X-X-X piped read cycle 1 = 3-X-X-X read followed by a 2-X-X-X piped read cycle
4	R/W	0	Write hit pipelined enable: 0 = Do not enable 2-X-X-X pipelined write hit cycles 1 = Enable 2-X-X-X pipelined write hit cycles
3	R/W	0	Write pulse timing control for cache write hit cycles: 0 = Do not change the write pulse timing during X-2-2-2 write hit cycles 1 = Move the write pulse one-half a clock later in X-2-2-2 write hit cycles
2	R/W	0	Write pulse timing control for cache write hit cycles: 0 = Do not change the write pulse timing during 3-X-X-X write hit cycles 1 = Move the write pulse one-half a clock earlier in 3-X-X-X write hit cycles
1	R/W	0	External 74F126 select: 0 = An external 74F126 is installed for CA3 and CA4 1 = An external 74F126 is not installed for CA3 and CA4 This bit should always be set to 1.
0	R/W	0	LCLK select control: If this bit is set, (i.e., a synchronous PCI/VL implementation is being used) then the timing constraints between the LCLK and CPUCLK inputs to the SYSC need to be met. LCLK <= 1/2 CPUCLK period before CPUCLK LCLK <= 0.5ns after CPUCLK 0 = LCLK is asynchronous to the CPUCLK 1 = LCLK is synchronous to the CPUCLK **Note:** In the synchronous LCLK option, LCLK = CPUCLK/2.

Table 5-32 57MISC2, 82C557 Miscellaneous Control Register 2- Index 11h

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved Bits: Must be written to 0.
5	R/W	0	Cache inactive during Idle state control: This bit controls the chip selects of the SRAMs. 0 = SRAM active always 1 = SRAM inactive during Idle state
4	R/W	0	Next address (NA#) mode control: If the CPU is used at a 50MHz operating frequency, then a 2-1-1-1 cycle on read/write hits to the synchronous SRAM can be obtained. To obtain this performance, the ADS# output of the CPU needs to be connected to the ADSP# input of the synchronous SRAM directly and this bit needs to be set. By setting this bit, generation of the NA# signal from the chipset to the CPU is controlled. 0 = Normal NA# timing used with asynchronous SRAMs 1 = New NA# timing for synchronous SRAMs - used only when CPU operating at 50MHz
3	R/W	0	SRAM type: This bit selects what type of SRAM is being used in the system. 0 = Asynchronous SRAM 1 = Synchronous SRAM
2	R/W	0	Page miss posted write: 0 = Enable 1 = Disable
1	R/W	0	ISA/DMA IOCHRDY control: 0 = Old mode, no IOCHRDY during line hit 1 = Drive IOCHRDY low until cycle is finished
0	R/W	0	Delay start: 0 = Old mode, do not delay internal master cycle cycles after an inquire cycle 1 = Delay internal master cycles by one LCLK after inquire cycle

Table 5-33 57REFCTL - 82C557 Refresh Control Register - Index 12h

Bit(s)	Type	Default	Function
7	R/W	0	REFRESH#/32KHz selection: This bit determines the source for REFRESH#. 0 = REFRESH# pulse from the 82C558 or the ISA master is the source of the REFRESH# input 1 = Source of REFRESH# pulse is a 32KHz clock
6	R/W	0	Reserved Bit: Must be written to 0.
5:4	R/W	00	Suspend mode refresh: 00 = From CLK state machine 01 = Self refresh based on 32KHz only 10 = Normal refresh based on 32KHz only 11 = Undefined
3:2	R/W	00	Slow refresh: 00 = Refresh on every REFRESH#/32KHz falling edge 01 = Refresh on alternate REFRESH#/32KHz falling edge 10 = Refresh on one in four REFRESH#/32KHz falling edge 11 = Refresh on every REFRESH#/32KHz toggle
1	R/W	0	LA[23:17] enable from 8Fh during refresh: 0 = Disable 1 = Enable
0	R/W	0	DBC MP[7:4] output enable during PCI master write: During a PCI master write cycle, the MP[7:4] parity bits can be generated by either the 82C558 IPC or the 82C556 DBC. For better performance during PCI master writes, these parity bits should be generated by the 82C558. If the option has been chosen to generate the MP[7:4] lines from the IPC, then the 82C557 can disable the DBC from generating the same. It does this via an encoded command that it sends to the DBC over the DBCOE#[1:0], MDOE#, and HDOE# lines 0 = Disable the DBC from generating the MP[7:4] lines during PCI master writes 1 = Enable the DBC to generate the MP[7:4] lines during PCI master writes If set to 0, there must be a pull-up on MP0 (strap option for DBC). If set to 1, there must be a pull-down on MP0 (strap option for DBC).

Table 5-34 57MEMDC1, 82C557 Memory Decode Control Register 1 - Index 13h

Bit(s)	Type	Default	Function
7	R/W	0	Memory decode select: This bit is used to determine various DRAM options. - It determines whether Table 5-14 or Table 5-15 will be used for Index 00h - It determines the different DRAM configuration options 0 = Compatible to 82C547 DRAM configurations as listed out in Table 5-36. In this case, there is not much flexibility in choosing different DRAM configurations. ***1 = Full decode - This gives the user maximum flexibility in choosing different DRAM configurations***
6:4	R/W	000	Full decode for logical bank 1 (RAS1#), if bit 7 set 000 = 0Kx36 001 = 256Kx36 010 = 512Kx36 011 = 1Mx36 100 = 2Mx36 101 = 4Mx36 110 = 8Mx36 111 = 16Mx36
3	R/W	0	SMRAM: 0 = Disable 1 = Enable
2:0	R/W	000	Full decode for logical bank 0 (RAS0#), if bit 7 set: Refer to Index 13h[6:4] for decode settings.

Table 5-35 57MEMDC2, 82C557 Memory Decode Control Register 2 - Index 14h

Bit(s)	Type	Default	Function
7	R/W	0	Reserved Bit: Must be written to 0.
6:4	R/W	000	Full decode for logical bank 3 (RAS3#) if Index 13h[7] is set: Refer to Index 14h[6:4] for decode settings.
3	R/W	0	SMRAM control: When SMIACT# is inactive: 0 = Disable SMRAM 1 = Enable SMRAM (if Index 13h[3] is set) When SMIACT# is active: 0 = Enable SMRAM for both Code and Data (if Index 13h[3] is set) 1 = Enable SMRAM for Code only (if Index 13h[3] is set)
2:0	R/W	000	Full decode for logical bank 2 (RAS2#) if Index 13h[7] is set: Refer to Index 13h[6:4] for decode settings.

Table 5-36 DRAM Configurations

4 3 2 1 0	SIMM 0	SIMM 1	SIMM 2	SIMM 3	Total
0 0 0 0 0	256K	256K			2M
0 0 0 0 1	512K	512K	----	----	4M
0 0 0 1 0	1M	1M	----	----	8M
0 0 0 1 1	2M	2M		----	16M
0 0 1 0 0	4M	4M		----	32M
0 0 1 0 1	8M	8M			64M
0 0 1 1 0	256K	256K	256K	256K	4M
0 0 1 1 1	256K	256K	512K	512K	6M
0 1 0 0 0	512K	512K	512K	512K	8M
0 1 0 0 1	256K	256K	1M	1M	10M
0 1 0 1 0	512K	512K	1M	1M	12M
0 1 0 1 1	1M	1M	1M	1M	16M
0 1 1 0 0	256K	256K	2M	2M	18M
0 1 1 0 1	512K	512K	2M	2M	20M
0 1 1 1 0	1M	1M	2M	2M	24M
0 1 1 1 1	2M	2M	2M	2M	32M
1 0 0 0 0	256K	256K	4M	4M	34M
1 0 0 0 1	512K	512K	4M	4M	36M
1 0 0 1 0	1M	1M	4M	4M	40M
1 0 0 1 1	2M	2M	4M	4M	48M
1 0 1 0 0	4M	4M	4M	4M	64M
1 0 1 0 1	256K	256K	8M	8M	66M
1 0 1 1 0	512K	512K	8M	8M	68M
1 0 1 1 1	1M	1M	8M	8M	72M
1 1 0 0 0	2M	2M	8M	8M	80M
1 1 0 0 1	4M	4M	8M	8M	96M
1 1 0 1 0	8M	8M	8M	8M	128M

Note: The fixed configurations shown in Table 5-36 are for maintaining compatibility with OPTi's 82C546/82C547 (Python) Chipset. Please refer to Indices 13h[7], 13h[2:0, 6:4], and 14h[2:0, 6:4] for greater flexibility in DRAM configurations.

Table 5-37 57PCICY1, 82C557 PCI Cycle Control Register 1 - Index 15h

Bit(s)	Type	Default	Function
7:6	R/W	00	CPU master to PCI memory slave, write IRDY# control: 00 = 3 LCLKs after end of address phase 01 = 2 LCLKs after end of address phase 10 = 1 LCLK after end of address phase 11 = 0 LCLK after end of address phase
5:4	R/W	00	CPU master to PCI slave write posting, bursting control: 00 = PCI slave write, no posting, no bursting 01 = PCI slave write, posting enabled, no bursting 10 = PCI slave write, posting enabled, conservative bursting 11 = PCI slave write, posting enabled, aggressive bursting
3:2	R/W	00	Master retry timer: 00 = Retries unmasked after 10 PCICLKs 01 = Retries unmasked after 18 PCICLKs 10 = Retries unmasked after 34 PCICLKs 11 = Retries unmasked after 66 PCICLKs
1	R/W	0	Reserved Bit: Must be written to 0.
0	R/W	0	PCI cycle, FRAME# timing control for pipelined cycles: During pipelined PCI cycles, the 82C557 SYSC can assert FRAME# either after the last BRDY# of the ongoing cycle or it can assert FRAME# on receiving ADS# for the next cycle. The former case is the conservative mode, whereas the latter is the aggressive mode. 0 = PCI cycle FRAME# assertion is done in the conservative mode style 1 = PCI cycle FRAME# assertion is done in the aggressive mode style

Table 5-38 57DRTY - 82C557 Dirty/Tag RAM Control Register - Index 16h

Bit(s)	Type	Default	Function
7	R/W	0	Dirty pin selection: This bit reflects the kind of SRAM that has been chosen for implementing the Dirty RAM. It also determines the functionality of the DIRTYI pin of the 82C557. If using a x1 SRAM for the Dirty RAM in which there is a separate DirtyIn and a separate DirtyOut bit, then the DIRTYI pin becomes an input only. If using a standard x8 or x9 SRAM, where there is no separate pin for input and output, then the DIRTYI pin becomes an I/O pin 0 = DIRTYI pin is an input only pin 1 = DIRYTI pin is an I/O pin
6	R/W	0	Reserved Bit: Must be written to 0.
5	R/W	0	Tag RAM size selection: This bit determines whether a 7- or 8-bit Tag RAM is used. If a 7-bit Tag is being used and a combined Tag/Dirty RAM is being used, then TAG0 functions as the DIRTYIO signal. In this case, the DIRTYI pin is unused. 0 = 8-bit Tag being used 1 = 7-bit Tag being used
4	R/W	0	Write hit cycle leadoff time when combining Dirty/Tag RAM: If the Dirty RAM is not implemented using a x1 SRAM, then the Tag and Dirty RAM implementation is considered combined. This bit decides the leadoff cycle time for a write hit in a system that has a combined Dirty/Tag implementation. 0 = Single write hit leadoff cycle = 5 cycles 1 = Single write hit leadoff cycle = 4 cycles
3	R/W	0	Pre-snoop control: 0 = Pre-snoop for starting address 0 only 1 = Pre-snoop for all addresses except those on the line boundary
2	R/W	0	Reserved Bit: Must be written to 0.
1	R/W	00	CPU to VL read access, DBC DLE#[1:0] timing: 0 = LCLK high 1 = LCLK low
0	R/W	0	HDOE# timing control: 0 = HDOE# is negated normally 1 = HDOE# is negated one clock before the cycle finishes

Table 5-39 57PCICY2, 82C557 PCI Cycle Control Register 2 - Index 17h

Bit(s)	Type	Default	Function
7	R/W	0	NA# assertion control for PCI slave accesses, when using synchronous PCI clock: 0 = No pipelining for accesses to PCI slave when using a synchronous PCI solution. 1 = Pipelining enabled for accesses to PCI slave for both asynchronous and synchronous PCI solutions. If this bit is set, it overrides bit 6.
6	R/W	0	NA# assertion control for PCI slave accesses, when using an asynchronous PCI clock: 0 = No pipelining for accesses to PCI slave when using an asynchronous PCI solution. 1 = Pipelining enabled for accesses to PCI slave for an asynchronous PCI implementation. **Note:** This bit will be overridden if bit 7 is set.
5	R/W	0	Support for Intel standard BSRAM: 0 = No support for Intel standard BSRAM 1 = Support for Intel standard BSRAM - This bit should be set only if using two banks of synchronous SRAM.
4	R/W	0	Fast BRDY# generation for PCI cycles: 0 = Normal timing of BRDY# generation for PCI cycles 1 = Fast generation of BRDY# for PCI cycles
3	R/W	0	Fast FRAME# generation for PCI cycles: 0 = Normal generation of FRAME# for PCI cycles 1 = Fast generation of FRAME# for PCI cycles
2	R/W	0	Byte merge/piping control: 0 = No pipelining when byte merging is on 1 = Pipelining enabled along with byte merging
1	R/W	0	Pipelined synchronous SRAM support: 0 = Standard synchronous SRAM installed 1 = Pipelined synchronous SRAM installed **Note:** This bit becomes applicable only if Index 11h[3] = 1.
0	R/W	0	Cyrix Linear burst mode support: 0 = Normal Intel standard burst mode 1 = Support for Cyrix linear burst mode

Table 5-40 57TRICTL, 82C557 Tristate Control Register - Index 18h

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved Bits: Must be written to 0.
5	R/W	0	Voltage selection for the CAS[7:0]# lines: 0 = CAS[7:0]# lines are driven out at 5.0V logic level 1 = CAS[7:0]# lines are driven out at 3.3V logic level
4	R/W	0	Programmable current drive for the MA[X], RAS[X]#, and the DWE# lines: 0 = Driving capability on these lines is 4mA 1 = Driving capability on these lines is 16mA
3	R/W	0	Tristate CPU interface during Suspend and during CPU power-off: 0 = Disable the tristate control 1 = Enable the tristate control
2	R/W	0	Tristate PCI interface during Suspend and during PCI power-off: 0 = Disable the tristate control 1 = Enable the tristate control
1	R/W	0	Tristate cache interface during Suspend and during cache power-off: 0 = Disable the tristate control 1 = Enable the tristate control
0	R/W	0	Enable the pull-up/pull-down resistors during Suspend and power-off: 0 = Disable the pull-up/pull-down resistors during Suspend and power-off 1 = Enable the pull-up/pull-down resistors during Suspend and power-off

Table 5-41 57MEMDC3, 82C557 Memory Decode Control Register 3 - Index 19h

Bit(s)	Type	Default	Function
7	R/W	0	DIRYTWE#/RAS5# selection: If six DRAM banks have been chosen, then the DIRTYWE# line will become RAS5#, if this bit is set. 0 = DIRTYWE# functions as DIRYTWE# (i.e., six banks of DRAM are not chosen) 1 = DIRTYWE# functions as RAS5# (i.e., six banks of DRAM are chosen) **Note:** If six banks of DRAM are chosen, then a combined Dirty/Tag SRAM solution must be implemented or else it will not have a Dirty RAM.
6:4	R/W	000	Full decode for logical bank 5 (RAS5#), if Index 13h[7] is set and Index 19h[7] is set: 000 = 0Kx36 100 = 2Mx36 001 = 256Kx36 101 = 4Mx36 010 = 512Kx36 110 = 8Mx36 011 = 1Mx36 111 = 16Mx36
3	R/W	0	MA11/RAS4# selection: If five DRAM banks have been chosen, then the MA11 line will become RAS4#, if this bit is set. 0 = MA11 functions as MA11 (i.e., the fifth bank of DRAM has not been chosen) 1 = MA11 functions as RAS4# (i.e., five banks of DRAM have been chosen) **Note:** If this bit is set to 1, then none of the DRAM banks will support the 8Mx36 or 16Mx36 options.
2:0	R/W	000	Full decode for logical bank 4 (RAS4#), if Index 13h[7] is set and Index 19h[3] is set: 000 = 0Kx36 100 = 2Mx36 001 = 256Kx36 101 = 4Mx36 010 = 512Kx36 110 = Undefined 011 = 1Mx36 111 = Undefined

Table 5-42 57RESV3, 82C557 Reserved Register 3 - Index 1Ah-1Fh

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved Bits: Must be written to 0.

5.1.3 Other SYSC I/O Registers

These registers are accessed normally, not via any indexing scheme.

Table 5-43 57REFPG, 82C557 Refresh Page Register (Write-Only) - Index 8Fh

Bit(s)	Type	Default	Function
7:0	WO		

57NVMADR, 82C557 Configuration/NVM Address Register - Index CF8h

- 32-bit writes to this register are captured by the 82C557 SYSC's internal configuration address register.
- When bit 31 of this register is set, access to configuration data register is translated to PCI configuration or special cycle according to Configuration Mechanism #1
- When bit 31 of this register is reset, access to configuration data register goes out as I/O access to the PCI bus.
- When bit 31 of this register is reset and bits [30:14] are set, NVMCS is generated during Configuration Data Register access and the content of Configuration Address Register bits [13:0] is put out as address [15:12, 9:0] and address [11:10] are driven to 0 on PCI and ISA buses.
- When bit 31 is reset and any of the bits [30:14] are reset, access to the Configuration Data Register proceeds as a normal I/O cycle.

57NVMDAT, 82C557 Configuration/NVM Data Register - Index CFCh

Note: CPU to VL read: BRDY# 2 CLKs after LRDY# if sync.
CPU to PCI read: BRDY# 1 LCLK + 1 CLK after TRDY# if sync.

5.2 82C558 IPC Register Space

The 82C558 IPC has four register spaces - PCI configuration register space, the system control configuration register space, the IPC I/O register space, and the power management register space.

5.2.1 82C558 IPC PCI Configuration Register Space

The 82C558 IPC's PCI Configuration Registers are accessed by Configuration Mechanism #1, on Bus #0, Device #1, Function #0.

Note: In the address offsets given below, the most significant bit (MSB) corresponds to the upper address offset.

Table 5-44 58VIN, 82C558 Vendor Identification Register - Address Offset 01h-00h

Bit(s)	Type	Default	Function
15:0	RO	1045h	Vendor Identification Register

Table 5-45 58DID, 82C558 Device Identification Register - Address Offset 03h-02h

Bit(s)	Type	Default	Function
15:0	RO	C558h	Device Identification Register

Table 5-46 58COM, 82C558 Command Register - Address Offset 05h-04h

Bit(s)	Type	Default	Function
15:10	R/W	0000 00	Reserved Bits: Must be written to 0.
9	RO	0	Enable fast back-to-back to different slaves: 0 = Disabled always.
8	R/W	0	SERR# output pin: 0 = Disable 1 = Enable
7	RO	0	Address/data stepping: 0 = Disable always.
6	R/W	0	PERR# output pin: 0 = Disable 1 = Enable
5	R/W	0	Reserved Bit: Must be written to 0.
4	RO	0	Memory write and invalidate cycle generation: Must always = 0. No memory write and invalidate cycles will be generated by the IPC.
3	R/W	0	Special cycles: 0 = Disable 1 = Enable IPC responds to Stop Grant special cycle.
2	R/W	1	Bus master operations: 0 = Disable 1 = Enable PCI cycle generation during DMA/ISA master may be disabled by this bit.
1	RO	1	Memory access: Must always = 1. The IPC allows a PCI bus master access to memory at any time.
0	RO	1	I/O access: Must always = 1. The IPC allows a PCI bus master to access I/O at any time.

Table 5-47 58STAT, 82C558 Status Register - Address Offset 07h-06h

Bit(s)	Type	Default	Function
15	RO	0	Parity error status: 0 = No parity error 1 = Parity error has occurred
14	RO	0	SERR# status: 0 = No system error 1 = System error has occurred
13	RO	0	Master abort status: Must always = 0.
12	RO	0	Received target abort status: 0 = No target abort 1 = Target abort occurred
11	RO	0	Signaled target abort status: Must always = 0.
10:9	RO	01	DEVSEL# timing status: These bits must always = 01. Medium timing is selected. The IPC asserts the DEVSEL# based on medium timing.
8	RO	0	Data parity status: 0 = No data parity detected 1 = Data parity detected
7	RO	1	Fast back-to-back capability: 0 = Not Capable 1 = Capable **Note:** Can be set to 0 only when Offset 44h[2] is set to 0; for debugging purposes only.
6:0	R/W	0000 000	Reserved Bits: Must always be written to 0.

Table 5-48 58REV, 82C558 Revision Identification Register - Address Offset 08h

Bit(s)	Type	Default	Function
7:0	RO	00h	Revision Identification Register

Table 5-49 58CLASS, 82C558 Class Code Register - Address Offset 0Bh-09h

Bit(s)	Type	Default	Function
23:0	RO	060100h	Class Code

Table 5-50 58RESV1, 82C558 Reserved Register 1 - Address Offset 0Ch

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved Bits: Must be written to 0.

Table 5-51 58MLTMR, 82C558 Master Latency Timer Register - Address Offset 0Dh

Bit(s)	Type	Default	Function
7:0	RO	00h	Master Latency Timer Register

Table 5-52 58HEAD, 82C558 Header Type Register - Address Offset 0Eh

Bit(s)	Type	Default	Function
7:0	RO	00h	Header Type

Table 5-53 58BIST, 82C558 Built-In Self-Test Register - Address Offset 0Fh

Bit(s)	Type	Default	Function
7:0	RO	00h	BIST

Table 5-54 58RESV2, 82C558 Reserved Register 2- Address Offset 3Fh-10h

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved Bits: Must be written to 0.

5.2.2 82C558 IPC System Control Configuration Register Space

These registers are accessed via the same configuration mechanism used to access the PCI Configuration Registers of the IPC (i.e., by Configuration Mechanism #1, on Bus #0, Device #1, Function #0). These registers are used to configure the system I/O, however, they are present physically as part of the IPC PCI Configuration Register Address Space.

Table 5-55 58KBDCTL, 82C558 Keyboard Control Register - Address Offset 41h-40h

Bit(s)	Type	Default	Function
15	RO	0	Keyboard port read: This is a read only bit. 0 = Does not say anything 1 = Keyboard controller has received command D0h and has not received the following 60h read
14	RO	0	Keyboard port write: This is a a read only bit. 0 = Does not say anything 1 = Keyboard controller has received command D1h and has not received the following 60h write
13	R/W	0	Immediate INIT generation: Generate INIT immediately on FEh command. 0 = Generate INIT immediately on FEh command 1 = Wait for halt before generating INIT on receiving the keyboard RESET
12	R/W	0	Enable keyboard emulation: This bit disables/enables keyboard emulation. 0 = Enable keyboard emulation In this case, the A20M#,KBRST# pin functions as an A20M# output pin. 1 = Disable keyboard emulation In this case, the A20M#,KBRST# pin functions as a KBRST# input pin.
11:0	R/W	000 000 000 000	Selects which IRQ signal is to be generated when PIRQ[X]# has been triggered: 000 = Disabled 001 = IRQ5 010 = IRQ9 011 = IRQ10 100 = IRQ11 101 = IRQ12 110 = IRQ14 111 = IRQ15 Bits [11:9] = PIRQ3# Bits [8:6] = PIRQ2# Bits [5:3] = PIRQ1# Bits [2:0] = PIRQ0#

Table 5-56 58MISC1, 82C558 Miscellaneous Control Register 1 - Address Offset 43h-42h

Bit(s)	Type	Default	Function
15:14	R/W	00	Reserved Bits: Must be written to 0.
13	R/W	0	Enable DMA/ISA master preemption: Enable DMA/ISA master to preempt PCI master. 0 = Disable 1 = Enable
12	R/W	0	Fixed/rotating priority amongst PCI masters: 0 = Rotating priority amongst PCI masters 1= Fixed priority amongst PCI masters - Priority sequence is PREQ0#, PREQ1#, PREQ2#
11:10	R/W	00	Back-to-back ISA I/O and memory delay: 00 = Delay all back to back ISA I/O cycles by 3 ATCLKs 01 = Delay all back to back ISA I/O cycles by 12 ATCLKs 10 = No delay between back-to-back ISA (I/O and memory) cycles 11 = Delay all back-to-back ISA (I/O and memory) cycles by 12 ATCLKs
9	R/W	0	PCI master access to VESA/AT devices: 0 = Enable 1 = Disable
8	R/W	0	AT bus control signals for memory access greater than 16M and for I/O accesses greater than 64K. 0 = Enable 1 = Disable
7:1	R/W	0000 000	Level triggering for IRQ[X]: If a PCI interrupt has been routed to a particular ISA IRQ, then that ISA IRQ would need to become level triggered (except IRQ14 for IDE). 0 = Edge 1 = Level Bit 7 = IRQ15 Bit 3 = IRQ10 Bit 6 = IRQ14 Bit 2 = IRQ9 Bit 5 = IRQ12 Bit 1 = IRQ5 Bit 4 = IRQ11
0	R/W	0	DREQ6 pin functionality: DREQ6 is pinwise programmable with EPMI#. This bit determines the functionality that this pin takes on. 0 = DREQ6 1 = EPMI#

Table 5-57 58PINFU1, 82C558 Pin Functionality Register 1 - Address Offset 45h-44h

Bit(s)	Type	Default	Function
15	R/W	0	Reserved Bit: Must be written to 0.
14	R/W	0	PGNT2# pin functionality: This bit determines the groupwise functionality of the PGNT2#/ GPCS1# pin. 0 = PGNT2# 1 = GPCS1#
13	R/W	0	PIRQ3# pin functionality: This pin can take on any of the these functionalities - PIRQ3#, LRDY#, or EPMI#. PIRQ3# and LRDY# functionalities are groupwise programmable, and those two functionalities are pinwise programmable with EPMI#. 0 = Controlled by bits [1:0] - if bits [1:0] are 00, 01, or 11 then this pin takes on PIRQ3# functionality. If bits [1:0] are 10, then this pin takes on LRDY# functionality. 1 = EPMI# - If this bit is set, then this pin takes on the functionality of EPMI#, regardless of the setting of bits [1:0].
12	R/W	0	PIRQ2# pin functionality: This pin can take on any of the following functionalities - PIRQ2# or LDEV#. These two functionalities are groupwise programmable. 0 = Controlled by bits [1:0] - if bits [1:0] are 00, 01, or 11 then this pin takes on PIRQ2# functionality. If bits [1:0] are 10, then this pin takes on LDEV# functionality. 1 = Reserved
11:10	R/W	00	PIRQ1# pin functionality: (This functionality is valid only if APIC is disabled.) This pin can take on any of the following functionalities - PIRQ1# or PICD1. If the APIC enable strap is sampled high at power-up reset, then this pin takes on PICD1 functionality, otherwise it takes on PIRQ1# functionality. 00 = PIRQ1# The rest are reserved. **Note:** If the local I/O APIC is enabled, then this pin takes on the functionality of PICD1, irrespective of the status of these bits.
9:8	R/W	00	PIRQ0# pin functionality: (These functionalities are valid only if APIC is disabled.) This pin can take on any of the following functionalities - PIRQ0#, EPMI#, or PICD1. PIRQ0# and EPMI# are pinwise programmable. If the APIC enable strap is sampled high at power-up reset, then this pin takes on PICD1 functionality, otherwise it takes on either PIRQ1# functionality or EPMI# functionality. 00 = PIRQ0# 01 = EPMI# 1X = Reserved **Note:** If the local I/O APIC is enabled, then this pin takes on the functionality of PICD0, irrespective of the status of these bits
7:6	R/W	00	DACK3# pin functionality: This pin can take on the following functionalities - DACK3#, EDACK2, or DACK7#. DACK3# and EDACK2 are groupwise programmable, and both of them are pinwise programmable with DACK7#. 0X =Controlled by bits [1:0] - If bits [1:0] are 00 or 01 then this pin takes on DACK3# functionality, if bits [1:0] are 10 or 11, then this pin takes on EDACK2 functionality. 10 = DACK7# **Note:** If set to 10 or 11, then the setting on bits [1:0] will not affect the functionality that this pin takes on.

58PINFU1, 82C558 Pin Functionality Register 1 - Address Offset 45h-44h (Cont.)

5:4	R/W	00	DACK1# pin functionality: This pin can take on the following functionalities - DACK1#, EDACK1, DACK6#, or GPCS2#. DACK1# and EDACK1 are groupwise programmable, and both of them are pinwise programmable with DACK6# and GPCS2#. 0X = Controlled by bits [1:0] - If bits [1:0] are 00 or 01, then this pin takes on DACK1# functionality. If bits [1:0] are 10 or 11, then this pin takes on EDACK1 functionality. 10 = DACK6 11 = GPCS2# **Note:** If set to 10 or 11, then the setting on bits[1:0] will not affect the functionality that this pin takes on.
3:2	R/W	00	DACK0# pin functionality: This pin can take on the following functionalities - DACK0#, EDACK0, DACK5#, or GPCS0#. DACK0# and EDACK0 are groupwise programmable, and both of them are pinwise programmable with DACK5# and GPCS0#. 0X = Controlled by bits [1:0] - If bits[1:0] are 00 or 01, then this pin takes on DACK0# functionality. If bits [1:0] are 10 or 11, then this pin takes on EDACK0 functionality. 10 = DACK5 11 = GPCS0# **Note:** If set to 10 or 11, then the setting on bits[1:0] will not affect the functionality that this pin takes on.
1:0	R/W	00	DACK/MP/PIRQ[3:2]# pin functionality: These bits determine the functionality of the groupwise programmable pins 00 = Explicit DACK[3:0], MP[7:4], PIRQ[3:2]# 01 = Explicit DACK[7:5,3:0], GPCS0#, PIRQ[3:2]# 10 = Encoded EDACK[2:0], EDACKEN#, LADS#, LW/R#, LM/IO#, LDEV#, LRDY# 11 = Encoded EDACK[2:0], EDACKEN#, MP[7:4], PIRQ[3:2]# Pinwise, these functions may be overridden by GPCS[X]#, EPMI#, and DACK[7:5] (for DACK[3,1,0]).

Table 5-58 58CYCTL1, 82C558 Cycle Control Register 1 - Address Offset 47h-46h

Bit(s)	Type	Default	Function
15	R/W	0	Write protect AT bus ROM: 0 = Disable ROMCS# for writes 1 = Enable ROMCS# for writes
14	R/W	0	Hidden refresh enable: 0 = Normal refresh 1 = Hidden refresh
13:12	R/W	00	ATCLK frequency select: (ATCLK frequency is) 00 = LCLK divided by 4 10 = LCLK divided by 2 01 = LCLK divided by 3 11 = LCLK
11	R/W	0	CPU master to PCI slave write: 0 = 1 LCLK turnaround between address and data phases 1 = 0 LCLK turnaround between address and data phases
10:8	R/W	000	PCI master to PCI master preemption timer: 000 = No preemption 001 = Preempt after unserviced request pending for 260 LCLKs 010 = Preempt after unserviced request pending for 132 LCLKs 011 = Preempt after unserviced request pending for 68 LCLKs 100 = Preempt after unserviced request pending for 36 LCLKs 101 = Preempt after unserviced request pending for 20 LCLKs 110 = Preempt after unserviced request pending for 12 LCLKs 111 = Preempt after unserviced request pending for 5 LCLKs
7:6	R/W	00	DMA/ISA access to PCI/VL slave: 00 = Never 01 = When both LMEM# and M16# are not asserted 10 = When LMEM# is not asserted 11 = Reserved
5	R/W	0	Conversion of PERR# to SERR#: 0 = Disable 1 = Enable
4	R/W	0	Address parity checking: 0 = Disable 1 = Enable
3	R/W	0	SERR# generation for target abort: 0 = Disable 1 = Enable
2	R/W	1	Fast back-to-back capability: 0 = Disable 1 = Enable **Note:** The change on this bit will reflect on bit 7 in Address Offset 06h.
1	R/W	1	Subtractive decoding sample point: 0 = Typical sample point 1 = Slow sample point
0	R/W	0	Wait for IRDY# to generate VL cycle: The data for the VL bus should be valid one LCLK after LADS# has been asserted. Once IRDY# is asserted, the data will definitely be available on the next clock. This bit gives the user the option of starting a VL cycle before IRDY# has been asserted (aggressive), or start a VL cycle after IRDY# has been asserted (conservative). 0 = Disable - Keep aggressive mode, where VL the start of the VL cycle does not depend on IRDY# 1 = Enable - Keep conservative mode, where a VL cycle is started only after IRDY# has been asserted

Table 5-59 58PINFU2, 82C558 Pin Functionality Register 2 - Address Offset 49h-48h

Bit(s)	Type	Default	Function
15	R/W	0	Reserved Bit: Must be written to 0.
14:13	R/W	00	IRQ12 pin functionality: This pin can take on the functionality of IRQ12 or MPIRQ2#/3#. These functionalities are pinwise programmable. 0X = IRQ12 10 = MPIRQ2#/3# 11 = Reserved
12	R/W	0	IRQ10 pin functionality: This pin can take on the functionality of IRQ10 or MIRQ10/12. These functionalities are pinwise programmable. 0 = IRQ10 1 = MIRQ10/12
11	R/W	0	IRQ6 pin functionality: This pin can take on the functionality of IRQ6 or MPIRQ0#/1#. These functionalities are pinwise programmable. 0 = IRQ6 1 = MPCIRQ0#/1#
10	R/W	0	IRQ4 pin functionality: This pin can take on the functionality of IRQ4 or MIRQ4/6. These functionalities are pinwise programmable. 0 = IRQ4 1 = MIRQ4/6
9:8	R/W	00	Reserved Bits: Must be written to 0.
7:6	R/W	00	DREQ3 pin functionality: This pin can take on the functionality of DREQ3, DREQ3/7, or DREQ7. These functionalities are pinwise programmable 00 = DREQ3 10 = DREQ7 01 = DREQ3/7 11 = Reserved
5:4	R/W	00	DREQ1 pin functionality: This pin can take on the functionality of DREQ1, DREQ1/6, or DREQ6. These functionalities are pinwise programmable. 00 = DREQ1 10 = DREQ6 01 = DREQ1/6 11 =Reserved
3:2	R/W	00	DREQ0 pin functionality: This pin can take on the functionality of DREQ0, DREQ0/5, or DREQ5. These functionalities are pinwise programmable. 00 = DREQ0 10 = DREQ5 01 = DREQ0/5 11 = Reserved
1:0	R/W	00	Reserved Bits: Must be written to 0.

Table 5-60 58ROMCS, 82C558 ROMCS# Range Control Register - Address Offset 4Bh-4Ah

Bit(s)	Type	Default	Function
15	R/W	0	ROMCS# generation for FFFF8000h-FFFFFFFFh segment: 0 = Enable 1 = Disable
14	R/W	0	ROMCS# generation for FFFF0000h-FFFF7FFFh segment: 0 = Enable 1 = Disable
13	R/W	0	ROMCS# generation for FFFE8000h-FFFEFFFFh segment: 0 = Disable 1 = Enable
12	R/W	0	ROMCS# generation for FFFE0000h-FFFE7FFFh segment: 0 = Disable 1 = Enable
11	R/W	0	ROMCS# generation for FFFD8000h-FFFDFFFFh segment: 0 = Disable 1 = Enable
10	R/W	0	ROMCS# generation for FFFD0000h-FFFD7FFFh segment: 0 = Disable 1 = Enable
9	R/W	0	ROMCS# generation for FFFE8000h-FFFCFFFFh segment: 0 = Disable 1 = Enable
8	R/W	0	ROMCS# generation for FFFC0000h-FFFC7FFFh segment: 0 = Disable 1 = Enable
7	R/W	0	ROMCS# generation for F8000h-FFFFFh segment: 0 = Enable 1 = Disable
6	R/W	0	ROMCS# generation for F0000h-F7FFFh segment: 0 = Enable 1 = Disable
5	R/W	0	ROMCS# generation for E8000h-EFFFFh segment 0 = Disable 1 = Enable
4	R/W	0	ROMCS# generation for E0000h-E7FFFh segment: 0 = Disable 1 = Enable
3	R/W	0	ROMCS# generation for D8000h-DFFFFh segment 0 = Disable 1 = Enable
2	R/W	0	ROMCS# generation for D0000h-D7FFFh segment: 0 = Disable 1 = Enable
1	R/W	0	ROMCS# generation for C8000h-CFFFFh segment: 0 = Disable 1 = Enable
0	R/W	0	ROMCS# generation for C0000h-C7FFFh segment: 0 = Disable 1 = Enable

Table 5-61 58APIC, 82C558 APIC Control Register - Address Offset 4Dh-4Ch

Bit(s)	Type	Default	Function
15:14	R/W	00	Reserved Bits: Must be written to 0.
13:10	R/W	0000	X-base address bits: These bits are used for APIC RAM relocation. (See Table 5-62.)
9:8	R/W	00	Y-base address bits: These bits along with the X-base address bits are used for APIC RAM relocation. (See Table 5-62.)
7:2	R/W	000000	Reserved Bits: Must be written to 0.
1	R/W	0	SMI# connection: This bit determines whether the SMI# to the CPU will be given through the SMI# pin or given through the APIC lines. 0 = SMI# routed via physical SMI# pin to the CPU 1 = SMI# routed via the APIC lines to the CPU
0	R/W	0	INTR enable/disable: 0 = INTR enable 1 = INTR disable - In this case, the APIC must be enabled as the INTR is routed to the CPU via the APIC lines.

Table 5-62 I/O APIC Base and Relocation Address

X-Base Address, C.R.: 4Dh[13:10]	Y-Base Address, C.R.: 4Dh[9:8]			
	Y - 0000 (0h)	Y - 0100 (4h)	Y - 10000 (8h)	Y - 1100 (Ch)
x - 0000 (0h)	FEC0 0000h * FEC0 0010h	FEC0 0400h FEC0 0410h	FEC0 0800h FEC0 0810h	FEC0 0C00h FEC0 0C10h
x - 0001 (1h)	FEC0 1000h FEC0 1010h	FEC0 1400h FEC0 1410h	FEC0 1800h FEC0 1810h	FEC0 1C00h FEC0 1C10h
x - 0010 (2h)	FEC0 2000h FEC0 2010h	FEC0 2400h FEC0 2410h	FEC0 2800h FEC0 2810h	FEC0 2C00h FEC0 2C10h
x - 0011 (3h)	FEC0 3000h FEC0 3010h	FEC0 3400h FEC0 3410h	FEC0 3800h FEC0 3810h	FEC0 3C00h FEC0 3C10h
x - 0100 (4h)	FEC0 4000h FEC0 4010h	FEC0 4400h FEC0 4410h	FEC0 4800h FEC0 4810h	FEC0 4C00h FEC0 4C10h
x - 0101 (5h)	FEC0 5000h FEC0 5010h	FEC0 5400h FEC0 5410h	FEC0 5800h FEC0 5810h	FEC0 5C00h FEC0 5C10h
x - 0110 (6h)	FEC0 6000h FEC0 6010h	FEC0 6400h FEC0 6410h	FEC0 6800h FEC0 6810h	FEC0 6C00h FEC0 6C10h
x - 0111 (7h)	FEC0 7000h FEC0 7010h	FEC0 7400h FEC0 7410h	FEC0 7800h FEC0 7810h	FEC0 7C00h FEC0 7C10h
x - 1000 (8h)	FEC0 8000h FEC0 8010h	FEC0 8400h FEC0 8410h	FEC0 8800h FEC0 8810h	FEC0 8C00h FEC0 8C10h
x - 1001 (9h)	FEC0 9000h FEC0 9010h	FEC0 9400h FEC0 9410h	FEC0 9800h FEC0 9810h	FEC0 9C00h FEC0 9C10h
x - 1010 (Ah)	FEC0 A000h FEC0 A010h	FEC0 A400h FEC0 A410h	FEC0 A800h FEC0 A810h	FEC0 AC00h FEC0 AC10h
x - 1011 (Bh)	FEC0 B000h FEC0 B010h	FEC0 B400h FEC0 B410h	FEC0 B800h FEC0 B810h	FEC0 BC00h FEC0 BC10h
x - 1100 (Ch)	FEC0 C000h FEC0 C010h	FEC0 C400h FEC0 C410h	FEC0 C800h FEC0 C810h	FEC0 CC00h FEC0 CC10h
x - 1101 (Dh)	FEC0 D000h FEC0 D010h	FEC0 D400h FEC0 D410h	FEC0 D800h FEC0 D810h	FEC0 DC00h FEC0 DC10h
x - 1110 (Eh)	FEC0 E000h FEC0 E010h	FEC0 E400h FEC0 E410h	FEC0 E800h FEC0 E810h	FEC0 EC00h FEC0 EC10h
x - 1111 (Fh)	FEC0 F000h FEC0 F010h	FEC0 F400h FEC0 F410h	FEC0 F800h FEC0 F810h	FEC0 FC00h FEC0 FC10h

*Default Address (FEC0 0000h for Register Select and FEC0 0010h for Window Register.

IO_APIC base address and relocation addresses:

a) FEC0 XY00h Register Select Register
b) FEC0 XY10h Window Register

Programming I/O APIC:

1. Write Register Select Register with Index - 00h (select IO_APIC ID Register).
2. Write Window Register with data (write data to IO_APIC ID Register).
3. Write Register Select Register with Index - 01h (select IO_APIC Version Register).
4. Write Window Register with data (write data to IO_APIC Version Register).
5. Write Register Select Register with Index - 10h, 12h, ..., FEh (select redirection table entry 0, 1, ..., 120 bit 0-31).
6. Write Window Register with data (with data to redirection table entry 0, 1, ..., 120 bit 0-31).
7. Write Register Select Register with Index - 11h, 13h, ..., FFh (select redirection table entry 0, 1, ..., 120 bit 0-31).
8. Write Window Register with data (write data to redirection table entry 0, 1, ..., 120 bit 32-63).

Table 5-63 58MISC2, 82C558 Miscellaneous Control Register 2- Address Offset 4Fh-4Eh

Bit(s)	Type	Default	Function
15:12	R/W	0000	Reserved: Must be written to 0.
11	R/W	0	APIC RAM write enabled/disabled: 0 = APIC RAM write enabled 1 = APIC RAM write disabled
10	R/W	0	MP7 pin functionality: This pin can take on the following functionalities - MP7, DACK7#, or LADS#. 0 = The functionality is decided by the groupwise programming option in Index 44h[1:0]. If those bits are 00 or 11, then this pin takes on the functionality of MP7. If those bits are 01 then it takes on the functionality of DACK7#, or if those bits are 10 then it takes on the functionality of LADS#. 1 = Should never be set to 1.
9	R/W	0	MP6 pin functionality: This pin can take on the following functionalities - MP6, DACK6#, LW/R#, or GPCS2#. The first three are groupwise programmable, and all three of them are pinwise programmable with GPCS2#. 0 = The functionality is decided by the groupwise programming option in Index 44h[1:0]. If those bits are 00 or 11, then this pin takes on the functionality of MP6. If those bits are 01 then it takes on the functionality of DACK6#, or if those bits are 10 then it takes on the functionality of LW/R#. 1 = This pin takes on the functionality of GPCS2#. **Note:** If this bit is set to 1 then this pin will take on the functionality of GPCS2#, irrespective of the setting in index 44h[1:0
8	R/W	0	Clock source for multiplexing/demultiplexing IRQs and DREQs: 0 = The 14.318MHz clock is used for demultiplexing the IRQs 1 = The LCLK clock is used for demultiplexing the IRQs
7:4	R/W	0000	Reserved Bits: Must be written to 0.
3	R/W	0	Pipelining with byte merge: 0 = Disable 1 = Enable
2	R/W	0	EOP direction configuration: 0 = EOP configured as output 1 = EOP configured as input
1	R/W	0	Byte merging: 0 = Disable 1 = Enable
0	R/W	0	ISA master data swap: 0 = Enable 1 = Disable

Table 5-64 58TRIG, 82C558 Trigger Control Register - Address Offset 51h-50h

Bit(s)	Type	Default	Function
15:11	R/W	00000	Reserved Bits: Must be written to 0.
10	R/W	0	Triggering for IRQ3: 0 = IRQ3 is edge triggered 1 = IRQ3 is level triggered
9	R/W	0	Triggering for IRQ4: 0 = IRQ4 is edge triggered 1 = IRQ4 is level triggered
8	R/W	0	Triggering for IRQ7: 0 = IRQ7 is edge triggered 1 = IRQ7 is level triggered
7:6	R/W	00	Selects which IRQ signal is generated when PIRQ3# has been triggered: 00 = Disabled 10 = IRQ4 01 = IRQ3 11 = IRQ7 **Note:** These bits are used along with the bits [11:9] in Index 41h-40h. If PIRQ3# is routed onto any of these ISA IRQs, then make sure that 41h-40h[11:9] = 000.
5:4	R/W	00	Selects which IRQ signal is generated when PIRQ2# has been triggered: 00 = Disabled 10 = IRQ4 01 = IRQ3 11 = IRQ7 **Note:** These bits are used along with the bits [8:6] in Index 41h-40h. If PIRQ2# is routed onto any of these ISA IRQs, then make sure that 41h-40h[8:6] = 000.
3:2	R/W	00	Selects which IRQ signal is generated when PIRQ1# has been triggered: 00 = Disabled 10 = IRQ4 01 = IRQ3 11 = IRQ7 **Note:** These bits are used along with the bits [5:3] in Index 41h-40h. If PIRQ1# is routed onto any of these ISA IRQs, then make sure that 41h-40h[5:3] = 000.
1:0	R/W	00	Selects which IRQ signal is generated when PIRQ0# has been triggered: 00 = Disabled 10 = IRQ4 01 = IRQ3 11 = IRQ7 **Note:** These bits are used along with the bits [2:0] in Index 41h-40h. If PIRQ0# is routed onto any of these ISA IRQs, then make sure that 41h-40h[2:0] = 000.

Table 5-65 Pin Multiplexing Chart 1

Default Pin Names	C.R.: 42h	Configuration Register: 44h								Configuration Register: 48h						C.R.: 4Fh	C.R.: 45h	
	Bit 0	Bits [7:6]	Bits [5:4]		Bits [3:2]		Bits [1:0][A]			Bits [7:6]		Bits [5:4]		Bits [3:2]		Bit 1	Bit 6	Bit 5
	1[B]	10[B]	10[B]	11[B]	10[B]	11[B]	11[C]	10[C]	01[C]	10[B]	01[B]	10[B]	01[B]	10[B]	01[B]	1[B]	1[B]	1[B]
DREQ0														DREQ 5	DREQ 0/5			
DREQ1												DREQ 6	DREQ 1/6					
DREQ2																		
DREQ3										DREQ 7	DREQ 3/7							
DREQ5																		
DREQ6	EPMIO#																	
DREQ7																		
DACK0#					DACK 5#	GPCS 0#	EDACK 0#	EDACK 0	DACK 0#									
DACK1#			DACK 6#	GPCS 2#			EDACK 1	EDACK 1	DACK 1#									
DACK2#							EDACK EN#	EDACK EN#	DACK 2#									
DACK3#		DACK 7#					EDACK 2	EDACK 2	DACK 1#									
MP4							MP4	MP4	GPCS 0#									
MP5							MP5	LM/IO	DACK 5#									
MP6							MP6	LW/R#	DACK 6#									
MP7							MP7	LADS#	DACK 7#									
PIRQ2#							PIRQ 2#	LDEV#	PIRQ 2#									
PIRQ3#							PIRQ 3#	LRDY#	PIRQ 3#									EPMI#
PGNT2#																	GPCS 1#	

A. Group-wise programming can be overridden by pin-wise programming.
B. Pin-wise programming
C. Group-wise programming

Table 5-66 Pin Multiplexing Chart 2

Default Pin Names	Configuration Register: 42h														C.R.: 51h						Configuration Register: 49h			
	Bit 7		Bit 6		Bit 5		Bit 4		Bit 3		Bit 2		Bit 1		Bit 2		Bit 1		Bit 0		Bits [6:5]	Bit 4	Bit 3	Bit 2
	1	0*	1	0*	1	0*	1	0*	1	0*	1	0*	1	0*	1	0*	1	0*	1	0*	10	1	1	1
IRQ1[A]																								
IRQ3															L	E								
IRQ4																	L	E						IRQ4/6
IRQ5													L	E										
IRQ6[A]																							PCIRQ0/1#	
IRQ7																			L	E				
IRQ8#[A]																								
IRQ9											L	E												
IRQ10									L	E												IRQ10/12		
IRQ11							L	E																
IRQ12					L	E																		
IRQ14			L	E																	PCIRQ2/3#			
IRQ15	L	E																						

*: Default
A: IRQ1, IRQ6 and IRQ8# are hard-wired as edge trigger (internal signal IRQ0, IRQ13 are also edge trigger).
E: Edge trigger (only applicable to 8259A/APIC required to program redirection table)
L: Level trigger (only applicable to 8259A/APIC required to program redirection table)

Table 5-67 Pin Multiplexing Chart 3

Default Pin Names	Configuration Register: 41h-40h								Configuration Register 50h			
	PCI IRQ Internal Routing[F]								PCI IRQ Internal Routing[G]			
	111	110	101	100	011	010	001	000*	11	10	01	00
PREQ0# (PCI_REQ0#)												
PREQ1# (PCI_REQ1#)												
PREQ2# (PCI_REQ2#)												
PGNT0# (PCI_GRANT0#)												
PGNT1# (PCI_GRANT1#)												
PGNT2# (PCI_GRANT2#)												
PCIRQ0# (PCI_IRQ0#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[A]	IRQ4[A]	IRQ3[A]	Disable
PCIRQ1# (PCI_IRQ1#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[B]	IRQ4[B]	IRQ3[B]	Disable
PCIRQ2# (PCI_IRQ2#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[C]	IRQ4[C]	IRQ3[C]	Disable
PCIRQ3# (PCI_IRQ3#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[D]	IRQ4[D]	IRQ3[D]	Disable

*: Default
A: Only valid with Configuration Register: 40h[11:9] - 000.
B: Only valid with Configuration Register: 40h[8:6] - 000.
C: Only valid with Configuration Register: 40h[5:3] - 000.
D: Only valid with Configuration Register: 40h[2:0] - 000.
E: Refer to Table 5-66 for additional muxing options on IRQ6 and IRQ12.
F: C.R.: 41h-40h[11:9] for PCIRQ0#, 41h-40h[8:6] for PCIRQ1#, 41h-40h[5:3] for PCIRQ2#, 41h-40h[2:0] for PCIRQ3#.
G: C.R.: 50h[7:6] for PCIRQ0#, 50h[5:4] for PCIRQ1#, 50h[3:2] or PCRIQ2#, 50h[1:0] for PCIRQ3#.

5.2.3 IPC I/O Registers

Table 5-68 Port B Register - I/O Port 61h

Bit(s)	Type	Function
7	RO	System parity check
6	RO	I/O channel check
5	RO	Timer OUT2 detect
4	RO	Refresh detect
3	R/W	I/O channel check: 0 = Enable 1 = Disable
2	R/W	Parity check: 0 = Enable 1 = Disable
1	R/W	Speaker output: 0 = Disable 1 = Enable
0	R/W	Timer 2 Gate: 0 = Disable (from CPU address) 1 = Enable

Keyboard I/O Control Registers - Ports 60h and 64h

The 82C558 will intercept commands to Ports 60h and 64h so that it may emulate the keyboard controller, allowing the generation of the fast GATEA20 and fast CPURST signals. The decode sequence is software transparent and requires no BIOS modifications to function. The fast GATEA20 generation sequence involves writing 'D1h' to Port 64h, then writing data '02h' to Port 60h. The fast CPU warm reset function is generated when a Port 64h write cycle with data 'FEh ' is decoded. A write to Port 64h with data D0h will enable the status of GATEA20 (bit 1 of Port 60h) and the system reset (bit 0 of Port 60h) to be readable.

Table 5-69 PS/2 Reset Control - Port 92h

Bit(s)	Type	Default	Function
7:2		0000 00	Reserved Bits
1		0	A20M# register: 0 = A20M# active 1 = A20M# inactive
0		0	Fast Reset (automatically clears back to 0): 1 = INIT sent to the 3.3V CPU

Address Offset FEh

A byte write to this register indicates a Stop Grant cycle. The data that is written is don't care. A read from this register is undefined. The SYSC propagates the CPU Stop Grant cycle as a configuration write to this register. The BIOS should never write to this configuration space. The IPC itself should ignore any write to this space unless it has driven the STPCLK# active.

5.2.4 82C558 Internal Integrated 82C206 Register Descriptions

The internal integrated 82C206 registers are accessed by indexing I/O Registers 22h and 23h. Index Register 01h should be set to the default value of C0h.

Following Table 5-70 are tables that explain the subsystem registers of the 82C206.

Table 5-70 Configuration Register (Index Port 22h, Data Port 23h) - Index: 01h

Bit(s)	Type	Default	Function
7:6	R/W	11	These bits control the number of wait states inserted when the CPU accesses the registers of the 82C206. Wait states are counted as SYSCLK cycles and are not affected by the DMA clock selection. 00 = One R/W wait state 01 = Two R/W wait states 10 = Three R/W wait states 11 = Four R/W wait states (Default)
5:4	R/W	00	These bits control the number of wait states inserted in 16-bit DMA cycles. Further control of the DMA cycle length is available through the use of the 82C558's IOCHRDY pin. During DMA cycles, this pin is used as an input to the wait state generation logic to extend the cycle if necessary. 00 = One 16-bit DMA wait state (Default) 01 = Two 16-bit DMA wait states 10 = Three 16-bit DMA wait states 11 = Four 16-bit DMA wait states
3:2	R/W	00	These bits control the number of wait states inserted in 8-bit DMA cycles. Further control of the DMA cycle length is available through the use of the 82C558's IOCHRDY pin. During DMA cycles, this pin is used as an input to the wait state generation logic to extend the cycle if necessary. 00 = One 8-bit DMA wait state (Default) 01 = Two 8-bit DMA wait states 10 = Three 8-bit DMA wait states 11 = Four 8-bit DMA wait states
1	R/W	0	This bit enables the early internal DMAMEMR# function. In a PC/AT-based system, DMAMEMR# is delayed one clock cycle later than SMEMR#. If set to 1, it will start DMAMEMR# at the time as SMEMR#. If set to 0, it will start DMAMEMR#.
0	R/W	0	If this bit is set to 0, the SYSCLK input is divided by two and is used to drive both 8- and 16-bit DMA subsystems. If this bit is set to 1, SYSCLK will directly drive the DMA subsystems. Whenever the state of this bit is changed, an internal synchronizer controls the actual switching of the clock to prevent a short clock pulse from causing a DMA malfunction.

5.2.4.1 DMA Subsystem Registers

Table 5-71 Command Register

Bit(s)	Type	Function
7	R/W	The setting of this bit determines if the DACK output pin will be active low or active high. 0 = Active low 1 = Active high
6	R/W	The setting of this bit determines if the DREQ input pin will be active low or active high. 0 = Active low 1 = Active high
5	R/W	Extended write: The extended write feature is controlled by this bit. When enabled, it causes the write command to be asserted one DMA cycle earlier during a transfer. Thus, read and write commands both begin in the S2 state. 0 = Disabled 1 = Enabled
4	R/W	Rotating priority: This bit selects wether the priority scheme is fixed or rotating. 0 = Fixed (Default) 1 = Rotating
3	R/W	Compressed timing: This bit enables the compressed timing feature. 0 = Compressed timing 1 = Normal timing (Default)
2	R/W	Controller disable: Setting this bit to 1 disables the DMA subsystem (DMA8 or DMA16). This function is normally used whenever the CPU needs to reprogram one of the channels to prevent DMA cycles from occurring. 0 = Enable 1 = Disable
1	R/W	Address hold: Setting this bit to 1 enables the address hold feature in Channel 0 when performing memory-to-memory transfers. 0 = Disable 1 = Enable
0	R/W	Memory-to-Memory: This bit enables Channel 0 and 1 to be used for memory-to-memory transfers. 0 = Disable 1 = Enable

Table 5-72 Mode Register

Bit(s)	Type	Function
7:6	R/W	Mode select bits 1 and 0: These bits are used to select the mode for each channel. 00 = Demand mode 10 = Block mode 01 = Single cycle mode 11 = Cascade mode
5	R/W	Decrement: Writing a 1 to this bit decrements the address after each transfer.
4	R/W	Auto-initialization: Writing a 1 to this bit enables the auto-initialization function.
3:2	R/W	Transfer type bits 1 and 0: These bits control the type of transfer to performed. 00 = Verify 10 = Read transfer 01 = Write transfer 11 = Illegal
1:0	R/W	Channel selection bits 1 and 0: These bits determine which channel's Mode Register will be written. Read back of a Mode Register will cause these bits to both be 1. 00 = Select Channel 0 10 = Select Channel 2 01 = Select Channel 1 11 = Select Channel 3

Table 5-73 Request Register Write Format

Bit(s)	Type	Function
7:3	W	Don't care
2	W	Request bit: Writing a 1 to this bit sets the request bit.
1:0	W	Request select bits 1 and 0: These bits determine which channel's request bit will be set. 00 = Select Channel 0 10 = Select Channel 2 01 = Select Channel 1 11 = Select Channel 3

Table 5-74 Request Register Read Format

Bit(s)	Type	Function
7:4	R	Reserved: Always reads 1.
3:0	R	Request channel bits 3 through 0: These bits contain the state of the request bit associated with each request channel. The bit position corresponds to the channel number.

Table 5-75 Request Mask Register Set/Reset Format

Bit(s)	Type	Function
7:3		Don't Care
2		Mask bit: Writing a 1 to this bit sets the request mask bit and inhibits external requests.
1:0		Mask select bits 1 and 0: These bits determine which channel's request bit will be set. 00 = Select Channel 0 10 = Select Channel 2 01 = Select Channel 1 11 = Select Channel 3

Table 5-76 Request Mask Register Read/Write Format

Bit(s)	Type	Function
7:4	R/W	Reserved: Always reads 1.
3:0	R/W	Mask Bits 3 through 0: These bits contain the state of the request mask bit associated with each request channel. The bit position corresponds to the channel number.

Table 5-77 Status Register

Bit(s)	Type	Function
7:4	R	Data Request bits 3 through 0: These bits show the status of each channel request and are not affected by the state of the Mask Register bits. Reading a 1 means "request" occurs and bits 7 through 4 represent Channels 3 through 0, respectively. These bits can be cleared by a reset, Master Clear, of the pending request being deasserted.
3:0	R	Terminal Count bits 3 through 0: These bits indicate which channel has reached the terminal count reading 1. These bits can be cleared by a reset, Master Clear, or each time a status read takes place. The channel number corresponds to the bit position.

5.2.4.2 Interrupt Controller Subsystem

Table 5-78 ICW1 Register - Address: 020h (0A0h)

Bit(s)	Type	Function
7:5	W	Don't Care
4	W	Must be set to 1 for ICW1 since ICW1, OCW2, and OCW3 share the same address, 020h (0A0h).
3	W	Level trigger mode: This bit selects either the level triggered mode or edge triggered mode input to the IR. If a 1 is written to LTM, a high level on the IR input will generate an interrupt request and the IR must be removed prior to EOI to prevent another interrupt. In the edge triggered mode, a low-to-high will generate an interrupt request. In either mode, IR must be held high until the first INTA cycle is started in order to generate the proper vector. IR7 vector will be generated if the IR input is negated early.
2	W	Don't Care
1	W	Single Mode: This bit selects between the Single and Cascade modes. The Single mode is used whenever only one interrupt controller (INTC1) is used and is not recommended for this device. The Cascade mode allows the two interrupt controllers to be connected through IR2 of INTC1. INTC1 will allow INTC2 to generate its own interrupt vectors if the Cascade mode is selected and the highest priority IR pending is from an INTC2 input. INTC1 and INTC2 must be programmed for the Cascade mode.
0	W	Don't Care

Table 5-79 ICW2 Register - Address: 021h (0A1h)

Bit(s)	Type	Function
7:3	W	Vector bits 5 through 0: These bits are the upper five bits of the interrupt vector and are programmable by the CPU. INTC1 and INTC2 need not be programmed with the same value in ICW2. Usually INTC1 is programmed with 08h and INTC2 with 70h.
2:0	W	Vector bits 2 through 0: The lower three bits of the vector are generated by the Priority Resolver during INTA

Table 5-80 ICW3 Register - Format for INTC1 - Address: 021h

Bit(s)	Type	Function
7:0	W	Slave mode bits 7 through 0: These bits select which IR inputs have Slave mode controller connected. ICW3 in INTC1 must be written with 04h (IRQ2) for INTC2 to function correctly.

Table 5-81 ICW3 Register - Format for INCT1 - Address: 0A1h

Bit(s)	Type	Function
7:3	W	Don't Care
2:0	W	Identify bits 2 through 0: Determines the Slave mode address the controller will respond to during the cascade INTA sequence. ICW3 in INTC2 should be written with a 02h (IRQ2 of INTC1) for operation in the Cascade mode.

Table 5-82 ICW4 Register - Address 021h (0A1h)

Bit(s)	Type	Function
7:5	W	Don't Care
4	W	Enable multiple interrupts: This bit will enable multiple interrupts from the same channel in the Fixed Priority mode. This allows INTC2 to fully nest interrupts when the Cascade and Fixed Priority mode are both selected, without being blocked by INTC1. Correct handling in this type of mode requires the CPU to issue a non-specific EOI command to zero when exiting an interrupt service routine. If zero, a non-specific EOI command should be sent to INTC1. If non-zero, no command is issued.
3:2	W	Don't Care
1	W	Auto end of interrupt: An AEOI is enabled when this bit is 1. The interrupt controller will perform a non-specific EOI on the trailing edge of the second INTA cycle. Note this function should not be used in a device with fully nested interrupts unless the device is a cascade master type.
0	W	Don't Care

Table 5-83 OCW1 Register - Address: 021h (0A1h)

Bit(s)	Type	Function
7:0	R/W	Mask bits 7 through 0: These bits control the state of the Interrupt Mask Register. Each Interrupt Register can be masked by writing a 1 in the appropriate bit position (M0 controls IR0, etc.). Setting an IMR bit has no affect on lower priority requests. All IMR bits are cleared by writing ICW1.

Table 5-84 OCW2 Register - Address: 020h (0A0h)

Bit(s)	Type	Function
7:5	W	These bits are used to select various operating functions. Writing a 1 in bit 7 causes one of the rotate functions to be selected. Writing a 1 in bit 6 causes a specific or immediate function to occur. All specific commands require L[2:0] to be valid except no operation. Writing a 1 in bit 5 causes a function related to EOI to occur. **7 6 5 Function** 0 0 0 Clear Rotate in Auto-EOI mode 0 0 1 Non-specific EOI Command 0 1 1 No Operation 0 1 1 Specific EOI Command* 1 0 0 Set Rotate in Auto-EOI Mode 1 0 1 Rotate on Non-specific EOI Command 1 1 0 Set Priority Command* 1 1 1 Rotate on specific EOI Command *L[2:0] are used by these commands.
4:3	W	These bits must be set to 0 to indicate that OCW2 is selected, because ICW1, OCW2, and OCW3 share the same address. 020h (0A0h).
2:0	W	These three bits are internally decoded to select which interrupt channel is to be affected by the Specific command. L[2:0] must be valid during three of the four specific cycles.

Table 5-85 OCW3 Register - Address 020h (0A0h)

Bit(s)	Type	Function
7	W	Reserved: This bits must be set to 0.
6:5	W	Enable Special Mask mode: Writing a 1 in bit 5 enables the set/reset Special Mask mode function. ESMM allows the other functions in OCW3 to be accessed and manipulated without affecting the Special Mask mode (SMM) state. During SMM, writing a 1 to any bit position of OCW1 inhibits interrupts and a 0 enables interrupts on the associated channel by causing the Priority Resolver to ignore the condition of the ISR. **6 5 Function** 0 X No operation 1 0 Reset Special Mask mode to Normal Mask mode 1 1 Set Special Mask mode
4:3	W	These bits must be set to 0 to indicate that OCW3 is selected because ICW1, OCW2, and OCW3 share the same address, 020h (0A0h).
2	W	Polled mode: Writing a 1 to this bit of OCW3 enables the Polled mode. Writing OCW3 with the Polled mode acts like the first INTA cycle, freezing all interrupt request lines and resolving priority. The next read operation to the controller acts like a second INTA cycle and polled vector is output to the data bus. The format of polled vector is described later.
1:0	W	Read Register: A 1 to this bit enables the contents of IRR or ISR (determined by RIS) to be placed on XD[7:0] when reading the Status Port at address 020h (0A0h). Asserting PM forces RR to reset. **1 0 Function** 0 X No Operation 1 0 Read IRR on the next read 1 1 Read ISR on the next read

Table 5-86 IIR Register - Address: 020h (0A0h)

Bit(s)	Type	Function
7:0		Interrupt request bits 7 through 0: These bits correspond to the interrupt request bits of the Interrupt Request Register. A 1 on these bits indicate that an interrupt request is pending on the corresponding line.

Table 5-87 ISR Register - Address 020h (0A0h)

Bit(s)	Type	Function
7:0		Interrupt service bits 7 through 0: These bits correspond to the interrupt service bits of the Interrupt Service Register. A 1 on these bits indicate that an interrupt is being serviced on the corresponding IS bits of the ISR.

Table 5-88 Poll Vector - Address 200h (0A0h)

Bit(s)	Type	Function
7		Interrupt: A 1 on this bit indicates that a pending interrupt is polled. If there is no pending interrupt request or the request is removed before the poll command, this bit is 0.
6:3		Don't Care
2:0		Vector bits 2 through 0: These bits are the binary encoding of the highest priority level pending interrupt request being polled. If no pending interrupt has been polled, all three bits are equal to 1.

5.2.4.3 Counter/Timer Subsystem

Table 5-89 Control Word Format (Write Only)

Bit(s)	Type	Function
7:6	W	Select counter bits 1 and 0: These bits select which counter this control word is written to. 00 = Select Counter 0 01 = Select Counter 1 10 = Select Counter 2 11 = Reserved for read-back command
5:4	W	Read/write bits 1 and 0: These bits determine the counter read/write word size. 00 = Reserved for counter latch command 01 = Read/write LSB only 10 = Read/write MSB only 11 = Read/write LSB first, then MSB
3:1	W	Mode select bits 2 through 0: These bits select the counter operating mode. 000 = Select Mode 0 001 = Select Mode 1 X10 = Select Mode 2 X11 = Select Mode 3 100 = Select Mode 4 101 = Select Mode 5
0	W	Binary coded decimal: During read/write counter commands control word writing, a 1 selects binary coded decimal count format. A 0 selects binary counting format. During read-back command word writing, this bit must be 0.

Table 5-90 Counter Latch Command Format (Write Only)

Bit(s)	Type	Function
7:6	W	Select counter bits 1 and 0: These bits select which counter is being latched. 00 = Select Counter 0 01 = Select Counter 1 10 = Select Counter 2 11 = Reserved for read-back command
5:4	W	These bits must be 0 for the counter latch command.
3:0	W	Don't care

Table 5-91 Read-Back Command Format (Write Only)

Bit(s)	Type	Function
7:6	W	These bits must be 1 for the read-back command
5	W	Latch count: A 0 in this bit will latch the count of the counting component of the selected counter(s);
4	W	Latch status: A 0 in this bit will latch the status information of the selected counter(s).
3:1	W	Counter select bits 2 through 0: These bits select which counter(s) the read-back command is applied to. 0XX = Select Counter 2 X0X = Select Counter 1 XX0 = Select Counter 0
0	W	Reserved: Write as 0.

Table 5-92 Status Format

Bit(s)	Type	Function
7	R	Out: This bit contains the state of the OUT signal of the counter.
6	R	Null Count: This bit contains the condition of the null count flag. This flag is used to indicate that the contents of the counting element are valid. It will be set to 1 during a write to the control register or the counter. It is cleared to a 0 whenever the counter is loaded from the counter input register.
5:4	R	Read/Write Word bits 1 and 0: These bits indicate the counter read/write word size. This information is useful in determining where the high byte, the low byte, or both must be must be transferred during counter read/write operations.
3:1	R	Mode bits 2 through 0: These bits reflect the operating mode of the counter and are interpreted in the same manner as in the write control word format.
0	R	Binary Coded Decimal: This bit indicates the counting element is operating in binary format or BCD format.

5.2.5 82C558 Power Management Registers

Table 5-93 58GRNCTL, Green Mode Control/Enable Status - Index E0h

Bit(s)	Type	Default	Function
7	R/W	0	SMI# generation enable bit: Setting this bit allows power management to be run through the SMI# protocol. 0 = Disable SMI# generation - the only way power management functions can now be utilized is by setting bit 3 to 1. 1 = Enable SMI# generation
6	R/W	0	GREEN event SMI# generation/status: When written to: Setting this bit allows generation of an SMI# on the occurrence of any GREEN event (if bit 7 is set). 0 = Disable - Green event occurrence will not cause an SMI# to be generated (this will also cause the status bit to be 0, and a read from the status bit will always yield a 0). 1 = Enable - Green event will cause an SMI# to be generated. When read from: This bit reflects whether or not a GREEN event has caused the SMI#. 0 = SIM# is not caused by a GREEN event. 1 = GREEN event has caused the system to generate a SMI#. The BIOS should read this bit to identify whether a GREEN event caused the generation of the SMI#.
5	R/W	0	Wake-up event SMI# generation/status: When written to: Setting this bit allows generation of an SMI# on the occurrence of any reload GET / wake-up event (if bit 7 is set). 0 = Disable - A wake-up event occurrence will not cause a SMI# to be generated (this will also cause the status bit to be 0, and a read from the status bit will always yield a 0). 1 = Enable - A wake-up event will cause a SMI# to be generated. When read from: This bit reflects whether or not a wake-up event has caused the SMI#. 0 = SMI# is not caused by a wake-up event. 1 = Wake-up event has caused the system to generate a SMI#. The BIOS should read this bit to identify whether a wake-up event caused the generation of the SMI#
4	RO	0	GREEN status bit: This is a read only bit. 0 = System is in NORMAL state 1 = System is in GREEN state
3	R/W	0	Hardware PPWRL# generation enable bit: Setting this bit allows a GREEN / wake-up event to generate a PPWRL# pulse. 0 = Disable 1 = Enable
2	R/W	0	Hardware PPWRL# generation for GREEN events: This bit allows generation of a hardware PPWRL# for GREEN events. 0 = Disable - This will cause the system not to generate a hardware PPWRL# for GREEN events. 1 = Enable - This will cause the system to generate a hardware PPWRL# for GREEN events (if bit 3 =1).
1	R/W	0	Hardware PPWRL# generation for Wake-up events: This bit allows you to control hardware PPWRL# generation for reload GET / wake-up events. 0 = Disable - This will cause the system not to generate a hardware PPWRL# for wake-up events. 1 = Enable - This will cause the system to generate a hardware PPWRL# for wake-up events (if bit 3 = 1).
0	R/W	0	Software GREEN event generation (can be used by APM): Setting this bit via software generates a GREEN event (if Index E1h, bit 0 is enabled). 0 = No action is taken. 1 = This will cause the generation of a GREEN event (if Index E1h[0] = 1).

Table 5-94 58GETCTL, EPMI Control / Green Event Timer - Index E1h Register

Bit(s)	Type	Default	Function
7:6	R/W	00	GREEN Event Timer (GET) CLK selection: This specifies the period of the CLK used by the GET. 00 = 119µs CLK period 01 = 12.25ms CLK period 10 = 1.94s CLK period 11 = 62.5s CLK period
5	R/W	0	EPMI polarity select: This allows the External Power Management Interrupt pin to be an active high or an active low input. 0 = EPMI is an active low input 1 = EPMI is an active high input
4	R/W	0	EPMI debounce select: Setting this bit allows debouncing of the EPMI input signals. 0 = Disable debounce 1 = Enable debounce (EPMI width needs to stay stable for >= 5ms to be recognized)
3	R/W	0	Reserved Bit: Must be written to 0.
2	R/W	0	Timer time-out into GREEN enable/status bit: When written to: This bit enables the GREEN Event Timer (GET) time-out to generate a GREEN event. 0 = Disables the GET time-out from causing a GREEN event 1 = Enables the GET time-out to cause a GREEN event When read from: This bit reflects whether a GREEN event has been invoked by the timer time-out (if 0 had been written to it then reading this bit will always show a 0). 0 = GREEN event has not been caused by the GET time-out 1 = GREEN event has been caused due to the GET time-out The BIOS should read this bit to identify the cause of the GREEN event. If it is caused by the GET time-out, then the BIOS should clear the bit by writing a 0.
1	R/W	0	EPMI trigger into GREEN enable/status bit: When written to: This bit enables a EPMI trigger to generate a GREEN event. 0 = Disables an EPMI trigger from causing a GREEN event 1 = Enables an EPMI trigger to cause a GREEN event When read from: This bit reflects whether a GREEN event has been invoked by an EPMI trigger (if 0 had been written to it then reading this bit will always show a 0). 0 = GREEN event is not invoked due to an EPMI trigger 1 = GREEN event is invoked due to an EPMI trigger The BIOS should read this bit to identify the cause of the GREEN event. If it is caused by an EPMI trigger, then the BIOS should clear the bit by writing a 0.
0	R/W	0	Software trigger into GREEN enable/status bit: When written to: This bit enables a software trigger to generate a GREEN event. 0 = Disables a software trigger from causing a GREEN event 1 = Enables a software trigger to cause a GREEN event When read from: This bit reflects whether a GREEN event has been invoked by a software trigger (if 0 had been written to it then reading this bit will always show a 0). 0 = GREEN event is not invoked by a software trigger 1 = GREEN event is invoked by a software trigger The BIOS should read this bit to identify the cause of the GREEN event. If it is caused by a software trigger, then the BIOS should clear the bit by writing a 0. After that, to enable the functionality the BIOS should write a 1.

Table 5-95 58COUNT, Green Event Timer Initial Count Register - Index E2h

Bit(s)	Type	Default	Function
7:0	R/W	00h	Green Event Timer count: Specifies the initial count (01h-FFh) for the GET. This count, along with Index E1h[7:6], specify the time-out period for the GET. **Note:** There is a latency of two clocks, so the actual time-out period will be: (the count selected +2) * (clock period defined in Index E1h[7:6])

Table 5-96 58IRQ1, IRQ 7-0 Event Enable Register - Index E3h

Bit(s)	Type	Default	Function
7:3	R/W	0000 0	IRQ[X] monitor: 0 = Disables IRQ[X] from becoming a reload GET / wake-up event 1 = Enables IRQ[X] to become a reload GET / wake-up event Bit 7 = IRQ7 Bit 4 = IRQ4 Bit 6 = IRQ6 Bit 3 = IRQ3 Bit 5 = IRQ5
2	R/W	0	IRQ15-0 deglitch select: 0 = No deglitch 1 = Sample the IRQ15-0 lines using an internal clock and reduce glitching
1:0	R/W	00	IRQ[X] monitor: 0 = Disables IRQ[X] from becoming a reload GET / wake-up event 1 = Enables IRQ[X] to become a reload GET / wake-up event Bit 1 = IRQ1 Bit 0 = IRQ0

Table 5-97 58IRQ2, IRQ15-8 Event Enable Register - Index E4h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	IRQ[X] monitor: 0 = Disables IRQ[X] from becoming a reload GET / wake-up event 1 = Enables IRQ[X] to become a reload GET / wake-up event Bit 7 = IRQ15 Bit 3 = IRQ11 Bit 6 = IRQ14 Bit 2 = IRQ10 Bit 5 = IRQ13 Bit 1 = IRQ9 Bit 4 = IRQ12 Bit 0 = IRQ8

Table 5-98 58DREQ, DREQ 7-0 Event Enable Register - Index E5h

Bit(s)	Type	Default	Function
7:5, 3:0	R/W	000 0000	DREQ[X] monitor: 0 = Disables DREQ[X] from becoming a reload GET / wake-up event. 1 = Enables DREQ[X] to become a reload GET if Index EFh[6] = 1 then this becomes a wake-up event. Bit 7 = DREQ7 Bit 3 = DREQ3 Bit 6 = DREQ6 Bit 2 = DREQ2 Bit 5 = DREQ5 Bit 1 = DREQ1 Bit 0 = DREQ0
4	R/W	0	Reserved Bit: Must be written to 0.

Table 5-99 58ACCDET, Device Cycle Monitor Enable Register - Index E6h

Bit(s)	Type	Default	Function
7	R/W	0	Programmable IO/MEM range: Setting this bit enables the IO/MEM address ranges specified in Indices E7h, E8h, and E9h to be monitored. 0 = Disables the device cycle to the address range specified from becoming a reload GET / wake-up event. 1 = Enables the device cycle to the address range specified to become a reload GET / wake-up event.
6	R/W	0	Parallel ports access detection: 0 = Disables accesses to the parallel ports from becoming a reload GET / wake-up event. 1 = Enables accesses to the parallel ports (3B0h-#BFh, 378h-37Fh, 278h-27Fh) to become a reload GET / wake-up event.
5	R/W	0	Video access detection: 0 = Disables accesses to video (A0000-BFFFFh, 3B0h-3DFh) from becoming a reload GET / wake-up event. 1 = Enables accesses to video (A0000-BFFFFh, 3B0h-3DFh) to become a reload GET / wake-up event.
4	R/W	0	Hard disk access detection: 0 = Disables accesses to the hard disk ports from becoming a reload GET /wake-up event. 1 = Enables accesses to hard disk (170h-177h, 376h, 1F0h-1F7h, 3F6h) to become a reload GET / wake-up event.
3	R/W	0	Floppy disk access detection: 0 = Disables accesses to the floppy disk to become a reload GET / wake-up event. 1 = Enables accesses to floppy disk (3F5h) from becoming a reload GET / wake-up event.
2	R/W	0	Keyboard access detection: 0 = Disables accesses to the keyboard from becoming a reload GET / wake-up event. 1 = Enables accesses to keyboard (60h, 64h) to become a reload GET / wake-up event.
1	R/W	0	COM ports 1/3 access detection: 0 = Disables accesses to these COM ports from becoming a reload GET / wake-up event. 1 = Enables accesses to the COM ports 1/3 (3F8h-3FFh, 3E8h-3EFh) to become a reload GET / wake-up event.
0	R/W	0	COM ports 2/4 access detection: 0 = Disables accesses to these COM ports from becoming a reload GET / wake-up event. 1 = Enables accesses to the COM ports 2/4 (2F8h-2FFh, 2E8h-2EFh) to become a reload GET / wake-up event.

Table 5-100 58MASK, Wake-up Source/Programmable I/O/Memory Address Mask Register - Index E7h

Bit(s)	Type	Default	Function
7	R/W	0	PREQ# detection: This bit, if set, will allow PCI bus requests to be monitored. 0 = Disables a PCI bus master cycle from generating a reload GET / wake-up event. 1 = Enables a PCI bus master cycle to generate a reload GET / wake-up event if Index EFh, bit 7 is set.
6	R/W	0	LDEV#/DEVSEL# detection: This bit if set will allow all accesses to the local bus be monitored 0 = Disables the generation of a reload GET / wake-up event on any local bus slave access. 1 = Enables the generation of a reload GET / wake-up event on any local bus slave access.
5	R/W	0	Reload GET on an EPMI# trigger: 0 = Disables the generation of a reload GET / wake-up event on an EPMI# trigger. 1 = Enables the generation of a reload GET / wake-up event on an EPMI# trigger.
4	R/W	0	Reserved Bit: Must be written to 0.
3	R/W	0	IO/MEM (non-system memory) selection for the programmable address ranges specified in Indices E8h, E9h: This bit determines whether the address specified in Indices E8h and E9h is an I/O or non-system memory address. 0 = The address specified in Indices E8h and E9h is an I/O address. 1 = The address specified in Indices E8h and E9h is a non-system memory address.
2:0	R/W	000	Mask bits for the programmable IO/MEM (non-system memory) address range (Index E8h): 000 = Mask no bits 001 = Mask the lowest bit 010 = Mask the lowest 2 bits 011 = Mask the lowest 3 bits 100 = Mask the lowest 4 bits 101 = Mask the lowest 5 bits 110 = Mask the lowest 6 bits 111 = Mask the lowest 7 bits

Table 5-101 58IOMEM1, Programmable IO/MEM Address Range Register - Index E8h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Programmable IO/MEM (non-system memory) address range: Depending on the selection in Index E7h[3], the address range specified in this register and in Index E9h corresponds to an I/O or a non-system memory address. Bits [7:0] map onto address lines A[7:0] for an I/O address and map onto address lines A[23:16] for a non-system memory address. Index E7h[2:0] specify the masking range.

Table 5-102 58IOMEM2, Programmable IO/MEM Address Range Register - Index E9h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Programmable IO/MEM (non-system memory) address range: Bits [7:0] map onto address lines A[15:8] for an I/O address and map onto address lines A[31:24] for a non-system memory address.

Table 5-103 58GPP, Enter GREEN State Port Register - Index EAh

Bit(s)	Type	Default	Function
7:0	R/W	0000 0001	Hardware power management information: This port provides the GREEN state values for the external power control latch (Index ECh). When the hardware PPWRL# is strobed to enter the GREEN state, this register will transfer its contents to Index ECh. GREEN state Power Port (GPP) bits [7:0] contain the information that is transferred to Index ECh.

Table 5-104 58NPP, Return to NORMAL State Configuration Port Register - Index EBh

Bit(s)	Type	Default	Function
7:0	R/W	0000 0001	NORMAL state configuration information: This port provides the return to NORMAL state values for the external power control latch (Index ECh). When the hardware PPWRL# is strobed to return to the NORMAL mode, this register will transfer its contents to Index ECh. NORMAL state Power Port (NPP) bits [7:0] contain the NORMAL state configuration information that is transferred to Index ECh.

Table 5-105 58PP, Shadow Register for External Power Control Latch Register - Index ECh

Bit(s)	Type	Default	Function
7:0	R/W	0000 0001	This port shadows the value of the external power control latch (Index ECh). Power Port (PP) bits [7:0] contain the value that the external power control latch has. A write to Index ECh will generate a PPWRL# pulse.

Table 5-106 58ACCMON, Device Cycle Detection Enable / Status Register - Index EDh

Bit(s)	Type	Default	Function
7	R/W	0	Programmable IO/MEM range monitor for SMI# generation/status: When written to: This bit, if set, will allow an access to the programmed IO/MEM address range (specified by Indices E8h and E9h), to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables an access to this programmed range from generating a SMI#. 1 = Enables an access to this programmed range to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by an access to this programmed range. 1 = SMI# has been invoked by an access to this programmed range. The BIOS should read this bit to identify the cause of the SMI#. If it is caused by a device cycle to the above programmed address range, then the BIOS should clear the bit by writing a 0.
6	R/W	0	LPT access monitor for SMI# generation/status: When written to: This bit, if set, will allow an LPT access to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables an LPT access from generating a SMI#. 1 = Enables an LPT access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by an LPT access cycle. 1 = SMI# has been invoked by an LPT access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to an LPT access, then the BIOS should clear the bit by writing a 0.
5	R/W	0	Video access monitor for SMI# generation/status: When written to: This bit, if set, will allow a video access to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a video access from generating a SMI#. 1 = Enables a video access to generate a SMI#. When read from: If a 0 has been written into this bit then a read from it will always show a 0. 0 = SMI# has not been invoked by a video access cycle. 1 = SMI# has been invoked by a video access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to a video access, then the BIOS should clear the bit by writing a 0.
4	R/W	0	Hard disk access monitor for SMI# generation/status: When written to: This bit, if set, will allow a hard disk access to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a hard disk access from generating a SMI#. 1 = Enables a hard disk access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by a hard disk access cycle. 1 = SMI# has been invoked by a hard disk access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to a hard disk access, then the BIOS should clear the bit by writing a 0.

58ACCMON, Device Cycle Detection Enable / Status Register - Index EDh (Cont.)

3	R/W	0	Floppy disk access monitor for SMI# generation/status: When written to: This bit, if set, will allow a floppy disk access to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a floppy disk access from generating a SMI#. 1 = Enables a floppy disk access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by a floppy disk access cycle. 1 = SMI# has been invoked by a floppy disk access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to a floppy disk access, then the BIOS should clear the bit by writing a 0.
2	R/W	0	Keyboard access monitor for SMI# generation/status: When written to: This bit, if set, will allow a keyboard access to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a keyboard access from generating a SMI#. 1 = Enables a keyboard access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by a keyboard access cycle. 1 = SMI# has been invoked by a keyboard access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to a keyboard access, then the BIOS should clear the bit by writing a 0.
1	R/W	0	COM ports 1/3 access monitor for SMI# generation/status: When written to: This bit, if set, will allow an access to COM ports 1/3 to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a COM1/COM3 access from generating a SMI#. 1 = Enables a COM1/COM3 access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by a COM1/COM3 access cycle. 1 = SMI# has been invoked by a COM1/COM3 access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to an access to COM ports 1/3, then the BIOS should clear the bit by writing a 0.
0	R/W	0	COM ports 2/4 access monitor for SMI# generation: When written to: This bit, if set, will allow an access to COM ports 2/4 to become a wake-up event. If Indices E0h[5] and E0h[7] are set to 1, then it will cause a SMI# to be generated. 0 = Disables a COM2/COM4 access to generate a SMI#. 1 = Enables a COM2/COM4access to generate a SMI#. When read from: If a 0 has been written into this bit, then a read from it will always show a 0. 0 = SMI# has not been invoked by a COM2/COM4 access cycle. 1 = SMI# has been invoked by a COM2/COM4 access cycle. The BIOS should read this bit to identify the cause of the SMI#. If it is caused due to an access to COM ports 2/4, then the BIOS should clear the bit by writing a 0.

Table 5-107 58STPCLK, STPCLK# Modulation Register - Index EEh

Bit(s)	Type	Default	Function
7	R/W	0	CPU STOPCLK state support: This bit, if set, will enable CPU STOPCLK state support. 0 = CPU STOPCLK state support disabled 1 = CPU STOPCLK state support enabled
6	R/W	0	CPU on hold when in STOPCLK state: This setting is mainly used for further lowering power consumption. 0 = CPU not on HOLD when in STOPCLK state 1 = CPU on HOLD when in STOPCLK state
5:4	R/W	00	Reserved Bits: Must be written to 0.
3	R/W	0	STPCLK# modulation enable bit: This bit, if set, will allow STPCLK# modulation. 0 = Disable STPCLK# modulation 1 = Enable STPCLK# modulation
2:0	R/W	000	STPCLK# modulation duty cycle: This comes into effect only if bit 3 = 1. 000 = STPCLK# = 1 always (i.e., no modulation) 001 = STPCLK# = 1 for 1/2 period 010 = STPCLK# = 1 for 1/4 period 011 = STPCLK# = 1 for 1/8 period 100 = STPCLK# = 1 for 1/16 period 101 = STPCLK# = 1 for 1/32 period 110 = STPCLK# = 1 for 1/64 period 111 = STPCLK# = 1 for 1/128 period

Table 5-108 58MISC3, Miscellaneous Register - Index EFh

Bit(s)	Type	Default	Function
7	R/W	0	PREQ# wake-up enable: This bit is used in conjunction with Index E7h[7] to enable a local bus request to wake-up the system. 0 = Disables PREQ# from waking up the system 1 = Enables PREQ# to wake-up the system
6	R/W	0	DREQ# wake-up enable: This bit is used in conjunction Index E5h to enable any DREQ# to wake-up the system. 0 = Disables all DREQ# from waking up the system 1 = Enables any DREQ# to wake-up the system
5	R/W	0	Reserved Bit: Must be written to 0.
4	R/W	0	Enable GPCS#1, GPCS#2 generation: Setting this bit enables generation of GPCS#1 and GPCS#2 for the address ranges specified in Indices F4h-F5h and F6h-F7h. 0 = Disable GPCS#1, GPCS#2 generation 1 = Enable GPCS#1, GPCS#2 generation
3:0	R/W	0000	Reserved Bits: Must be written to 0.

Table 5-109 58TMRCLK, Device Timer CLK Select / Enable Status Register - Index F0h

Bit(s)	Type	Default	Function
7:6	R/W	00	Device timer 1 CLK selection: This is one of the two general purpose device timers. These bits determine the clock period for this device timer clock. 00 = 119µs clock period 01 = 12.25ms clock period 10 = 1.94s clock period 11 = 62.5s clock period
5:4	R/W	00	Device timer 0 CLK selection: This is the second of the two general purpose device timers. These bits determine the clock period for this device timer clock. 00 = 119µs clock period 01 = 12.25ms clock period 10 = 1.94s clock period 11 = 62.5s clock period
3	R/W	0	Device timer 1 time-out GREEN event enable / status: When written to: If this bit is set, then this device timer time-out becomes a GREEN event. 0 = Disables device timer 1 time-out from becoming a GREEN event. 1 = Enables device timer 1 time-out to become a GREEN event. When read from: If this has not been enabled (i.e., not set to 1), then a read from it will always fetch 0. 1 = GREEN event has been invoked by device timer 1 time-out. 0 = GREEN event has not been invoked by device timer 1 time-out. The BIOS should read this bit to identify the cause of the GREEN event. If it is caused due to device timer 1 time-out, then the BIOS should clear the bit by writing a 0. After that, to enable the functionality again, the BIOS should write a 1.
2	R/W	0	Device timer 0 time-out GREEN event enable / status: When written to: If this bit is set, then this device timer time-out becomes a GREEN event. 0 = Disables device timer 0 time-out from becoming a GREEN event. 1 = Enables device timer 0 time-out to become a GREEN event. When read from: If this has not been enabled (i.e., not set to 1) then a read from it will always fetch 0. 1 = GREEN event has been invoked by device timer 0 time-out. 0 = GREEN event has not been invoked by device timer 0 time-out. The BIOS should read this bit to identify the cause of the GREEN event. If it is caused due to device timer 0 time-out, then the BIOS should clear the bit by writing a 0. After that, to enable the functionality again, the BIOS should write a 1.
1	R/W	0	Device 1 cycle generate wake-up event enable / status: When written to: This bit, if enabled, will allow any access to the device 1 address range (Indices F6h, F7h) to become a wake-up event. 0 = Disables a device 1 cycle from becoming a wake-up event. 1 = Enables a device 1 cycle to become a wake-up event. When read from: If a 0 had been written to it, then a read from it will always fetch 0. 0 = Wake-up event has not been invoked by an access cycle to device 1. 1 = Wake-up event has been invoked by an access cycle to device 1. The BIOS should read this bit to identify the cause of the wake-up event. If it is caused by an access to the device 1 address range, then the BIOS should clear the bit by writing a 0. After that, to enable the functionality again, the BIOS should write a 1. **Note:** If device timers 1 and 2 are not utilized, then the F1h-F7h register space can be used as scratch pad space.
0	R/W	0	Device 0 cycle generate wake-up event enable / status: When written to: This bit, if enabled, will allow any access to the device 0 address range (Indices F5h, F4h) to become a wake-up event. 0 = Disables a device 0 cycle from becoming a wake-up event. 1 = Enables a device 0 cycle to become a wake-up event. When read from: If a 0 had been written to it, then a read from it will always fetch 0. 0 = Wake-up event has not been invoked by an access cycle to device 0. 1 = Wake-up event has been invoked by an access cycle to device 0. The BIOS should read this bit to identify the cause of the wake-up event. If it is caused by an access to the device 0 address range, then the BIOS should clear the bit by writing a 0. After that, to enable the functionality again, the BIOS should write a 1.

Table 5-110 58DV0TMR, Device 0 Timer Initial Count Register - Index F1h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 0 timer initial count: Bits [7:0] specify the initial count (00h-FFh) for device 0 timer. These bits, in conjunction with Index F0h[5:4], define the time-out period for device 0 timer.

Table 5-111 58DV1TMR, Device 1 Timer Initial Count Register - Index F2h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 1 timer initial count: Bits [7:0] specify the initial count (00h-FFh) for device 1 timer. These bits, in conjunction with Index F0h[7:6], define the time-out period for device 1 timer.

Table 5-112 58IOMSK, Device Timer IO/MEM Select, Mask Bits Register - Index F3h

Bit(s)	Type	Default	Function
7	R/W	0	IO/MEM selection for device 1: This bit determines whether the address range specified for device 1 in Indices F6h and F7h, is an I/O address or a memory address. 0 = The address range specified is an I/O address 1 = The address range specified is a memory address
6:4	R/W	000	Mask bits for device 1 IO/MEM address range: 000 = Mask no bits 001 = Mask the lowest bit 010 = Mask the lowest 2 bits 011 = Mask the lowest 3 bits 100 = Mask the lowest 4 bits 101 = Mask the lowest 5 bits 110 = Mask the lowest 6 bits 111 = Mask the lowest 7 bits
3	R/W	0	IO/MEM selection for device 0: This bit determines whether the address range specified for device 0 in Indices F5h and F4h, is an I/O address or a memory address 0 = The address range specified is an I/O address 1 = The address range specified is a memory address
2:0	R/W	000	Mask bits for device 0 IO/MEM address range: 000 = Mask no bits 001 = Mask the lowest bit 010 = Mask the lowest 2 bits 011 = Mask the lowest 3 bits 100 = Mask the lowest 4 bits 101 = Mask the lowest 5 bits 110 = Mask the lowest 6 bits 111 = Mask the lowest 7 bits

Table 5-113 58DV0AD1, Device 0 IO/MEM Address Register - Index F4h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 0 IO/MEM address: Bits [7:0] of this register are used in conjunction with bits [7:0] of Index F5h to specify the I/O or the memory address of device 0. If Index F3h[3] = 0, then these bits map onto A[7:0] as an I/O address. If Index F3h[3] = 1, then these bits map onto A[23:16] as a memory address. Index F3h[2:0] specify which of these bits are to be masked.

Table 5-114 58DV0AD2, Device 0 IO/MEM Address Register - Index F5h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 0 IO/MEM address: Bits [7:0] of this register are used in conjunction with bits [7:0] of Index F4h to specify the I/O or the memory address of device 0. If Index F3h[3] = 0, then these bits map onto A[15:8] as an I/O address. If Index F3h[3] = 1, then these bits map onto A[31:24] as a memory address. **Note:** If Index EFh[4] = 1, then an access to the address range specified here causes GPCS#1 to be asserted

Table 5-115 58DV1AD1, Device 1 IO/MEM Address Register - Index F6h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 1 IO/MEM address: Bits [7:0] of this register are used in conjunction with bits[7:0] of Index F7h to specify the I/O or the memory address of device 1. If Index F3h[7] = 0, then these bits map onto A[7:0] as an I/O address. If Index F3h[7] = 1, then these bits map onto A[23:16] as a memory address. Index F3h[6:4] specify which of these bits are to be masked.

Table 5-116 58DV1AD2, Device 1 IO/MEM Address Register - Index F7h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	Device 1 IO/MEM address: Bits [7:0] of this register are used in conjunction with bits [7:0] of Index F6h to specify the I/O or the memory address of device 0. If Index F3h[7] = 0, then these bits map onto A[15:8] as an I/O address. If Index F3h[7] = 1, then these bits map onto A[31:24] as a memory address. **Note:** If Index EFh[4] = 1, then an access to the address range specified here causes GPCS#2 to be asserted

5.3 Register Summaries

Table 5-117 82C557 Register Space Summary

Addr.	7	6	5	4	3	2	1	0
PCI Configuration Register Space								
01h-00h	82C557 Vendor Identification Register							
03h-02h	82C557 Device Identification Register							
05h-04h	Reserved						Fast back-to-back to different slaves	SERR# output pin
	Address/data stepping	PERR# output pin	Reserved	Memory write & invalidate cycle generation	Special cycles	Bus master operations	Memory access	I/O access
07h-06h	Detected parity error	SERR# status	Master abort status	Received target abort status	Signaled target abort status	DEVSEL# timing status		Data parity detected
	Fast back-to-back capability	Reserved						
08h	82C557 Revision Identification Register							
0Bh-09h	82C557 Class Code Register							
0Ch	Reserved							
0Dh	82C557 Master Latency Timer Register							
0Eh	82C557 Header Type Register							
0Fh	82C557 BIST Register							
3Fh-10h	Reserved							
43h-40h	Reserved							
	Reserved							
	VL video frame buffer write posting hole							
			Reserved			Write posting hole	Video memory (A0000h-BFFFFh) write post control	I/O cycle write post control

System Control Register Space								
Index	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
00h (or, see below)	Enable pipelining of single CPU cycles to memory	Video memory byte/word read prefetch enable	Sony SONIC-2WP support enable	Byte/word merge support	Byte/word merging with CPU pipelining (NA# generation) support	Time-out counter for byte/word merge		Enable internal hold requests to be blocked while performing byte merge
00h (or, see above)	Enable pipelining of single CPU cycles to memory	Second bank: 512x36 SIMM	First bank: 512x36 SIMM	Memory configurations select (default = 2MB)				
01h	Row address hold after RAS# active in CLKs	RAS# active/ inactive on entering master mode	RAS pulse width used during refresh in CLKs		CAS pulse width during reads in CLKs	CAS pulse width during writes in CLKs	RAS precharge time in CLKs	

82C557 Register Space Summary (Cont.)

Index	7	6	5	4	3	2	1	0
02h	Cache size selection		Cache write policy		Cache mode select		DRAM posted write	CAS pre-charge time in CLKs
03h	Cache write burst mode CLKs		Leadoff cycle for cache writes		Cache read burst mode CLKs		Leadoff cycle for cache reads	
04h	CC000h-CFFFFh read/write control		C8000h-CBFFFh read/write control		Synchronous SRAM pipe-lined read cycle 1-1-1-1 enable	E0000h-EFFFFh range selection	C0000h-C7FFFh read/write control	
05h	DX000h-DXFFFh read/write control							
06h	DRAM hole in system memory from 80000h-9FFFFh	Wait state addition for PCI master snooping	Range C0000h-C7FFFh cacheability	Range F0000h-FFFFFh cacheability	F0000h-FFFFFh read/write control		E0000h-EFFFFh read/write control	
07h	82C557 Tag Test Register							
08h	L2 cache single/double bank select	Snoop filtering for bus masters	CPU HITM# pin sample timing	Parity checking	Tag/Dirty RAM implementation	CPU address pipelining	L1 cache write-back and write-through control	BIOS area cacheability in L1 cache
09h	DRAM Hole B size		DRAM Hole B control mode		DRAM Hole A size		DRAM Hole A control mode	
0Ah	DRAM Hole A starting address							
0Bh	DRAM Hole B starting address							
0Ch	Reserved	Fast BRDY# generation for DRAM write page hits	HACALE one-half a clock cycle earlier	Wider cache WE# pulse	DRAM Hole B starting address		DRAM Hole A starting address	
0Dh	Clock source for generating the sync. SRAM timing	The 82C557 sets this bit if the skew between ECLK and CPU CLK is too large	Auto skew detect	ECLK - CLK skew	Enable A0000h-BFFFFh as system memory	Wait state addition for PCI master doing address toggling in the 486 style	PCI cycle claimed by the 82C557 during PCI pre-snoop cycle	Slow CPU clock
0Eh	PCI master read burst wait state control		PCI master write burst wait state control		Master cycle parity enable	HACALE timing control	Write protection for L1 BIOS	PCI line comparator
0Fh	PCI pre-snoop	AT master wait state control	Wait state addition for synchronous SRAM even byte access	PCI wait state addition for the synchronous SRAM L2 cache implementation	Reserved	ADSC# generation for synchronous SRAM read cycle	Reserved	Cache size selection
10h	Early decode of PCI/VL/AT cycle	Cache modified write cycle timing	Pipelined read cycle timing	Write hit pipelined enable	Write pulse timing control for cache write hit cycles	Write pulse timing control for cache write hit cycles	External 74F126 select	LCLK select control
11h	Reserved		Cache inactive during Idle state control	Next address (NA#) mode control	SRAM type	Page miss posted write	ISA/DMA IOCHRDY control	Delay start

82C557 Register Space Summary (Cont.)

Index	7	6	5	4	3	2	1	0
12h	REFRESH#/ 32KHz selection	Reserved	Suspend mode refresh		Slow refresh		LA[23:17] enable from 8Fh during refresh	DBC MP[7:4] output enable during PCI master write
13h	Memory decode select	Full decode for logical bank 1 (RAS1#), if bit 7 set			SMRAM	Full decode for logical bank 0 (RAS0#), if bit 7 set		
14h	Reserved	Full decode for logical bank 3 (RAS3#) if Index 13h[7] is set			SMRAM control	Full decode for logical bank 2 (RAS2#) if Index 13h[7] is set		
15h	CPU master to PCI memory slave, write IRDY# control		CPU master to PCI slave write posting, bursting control		Master retry timer		Reserved	PCI cycle, FRAME# timing control for pipelined cycles
16h	Dirty pin selection	Reserved	Tag RAM size selection	Write hit cycle leadoff time when combining Dirty/Tag RAM	Pre-snoop control	Reserved	CPU to VL read access, DBC DLE#[1:0] timing	HDOE# timing control
17h	NA# assertion control for PCI slave accesses, when using synchronous PCI clock	NA# assertion control for PCI slave accesses, when using an asynchronous PCI clock	Support for Intel standard BSRAM	Fast BRDY# generation for PCI cycles	Fast FRAME# generation for PCI cycles	Byte merge/ piping control	Pipelined synchronous SRAM support	Cyrix Linear burst mode support
18h	Reserved		Voltage selection for the CAS[7:0]# lines	Programmable current drive for the MA[X], RAS[X]#, and the DWE# lines	Tristate CPU interface during Suspend and during CPU power-off	Tristate PCI interface during Suspend and during PCI power-off	Tristate cache interface during Suspend and during cache power-off	Enable the pull-up/pull-down resistors during Suspend and power-off
19h	DIRYTWE#/ RAS5# selection	Full decode for logical bank 5 (RAS5#), if Index 13h[7] is set and Index 19h[7] is set			MA11/RAS4# selection	Full decode for logical bank 4 (RAS4#), if Index 13h[7] is set and Index 19h[3] is set		
1Ah-1Fh	Reserved							

Table 5-118 82C558 Register Space Summary

Addr.	7	6	5	4	3	2	1	0
PCI Configuration Register Space								
01h-00h	82C558 Vendor Identification Register							
03h-02h	82C558 Device Identification Register							
05h-04h	Reserved						Fast back-to-back to different slaves	SERR# output pin
	Address/data stepping	PERR# output pin	Reserved	Memory write & invalidate cycle generation	Special cycles	Bus master operations	Memory access	I/O access
07h-06h	Parity error status	SERR# status	Master abort status	Received target abort status	Signaled target abort status	DEVSEL# timing status		Data parity status
	Fast back-to-back capability	Reserved						
08h	82C558 Revision Identification Register							
0Bh-09h	82C558 Class Code Register							
0Ch	Reserved							
0Dh	82C558 Master Latency Timer Register							
0Eh	82C558 Header Type Register							
0Fh	82C558 BIST Register							
3Fh-10h	Reserved							
System Control Register Space								
41h-40h	Keyboard port read	Keyboard port write	Immediate INIT generation	Enable keyboard emulation	Selects which IRQ signal is to be generated when PIRQ[X]# has been triggered			
43h-42h	Reserved		Enable DMA/ISA master preemption	Fixed/rotating priority amongst PCI masters	Back-to-back ISA I/O and memory delay		PCI master access to VESA/AT devices	AT bus control signals for memory access greater than 16M and for I/O accesses greater than 64K
	Level triggering for IRQ[X]							DREQ6 pin functionality
45h-44h	Reserved	PGNT2# pin functionality	PIRQ3# pin functionality	PIRQ2# pin functionality	PIRQ1# pin functionality		PIRQ0# pin functionality	
	DACK3# pin functionality		DACK1# pin functionality		DACK0# pin functionality		DACK/MP/PIRQ[3:2]# pin functionality	
47h-46h	Write protect AT bus ROM	Hidden refresh enable	ATCLK frequency select		CPU master to PCI slave write	PCI master to PCI master preemption timer		
	DMA/ISA access to PCI/VL slave		Conversion of PERR# to SERR#	Address parity checking	SERR# generation for target abort	Fast back-to-back capability	Subtractive decoding sample point	Wait for IRDY# to generate VL cycle

82C558 Register Space Summary (Cont.)

Addr.	7	6	5	4	3	2	1	0
49h-48h	Reserved	IRQ12 pin functionality		IRQ10 pin functionality	IRQ6 pin functionality	IRQ4 pin functionality	Reserved	
	DREQ3 pin functionality		DREQ1 pin functionality		DREQ0 pin functionality		Reserved	
4Bh-4Ah	ROMCS# generation for FFFF8000h-FFFFFFFFh	ROMCS# generation for FFFF0000h-FFFF7FFFh	ROMCS# generation for FFFE8000h-FFFEFFFFh	ROMCS# generation for FFFE0000h-FFFE7FFFh	ROMCS# generation for FFFD8000h-FFFDFFFFh	ROMCS# generation for FFFD0000h-FFFD7FFFh	ROMCS# generation for FFFE8000h-FFFCFFFFh	ROMCS# generation for FFFC0000h-FFFC7FFFh
	ROMCS# generation for F8000h-FFFFFh	ROMCS# generation for F0000h-F7FFFh	ROMCS# generation for E8000h-EFFFFh	ROMCS# generation for E0000h-E7FFFh	ROMCS# generation for D8000h-DFFFFh	ROMCS# generation for D0000h-D7FFFh	ROMCS# generation for C8000h-CFFFFh	ROMCS# generation for C0000h-C7FFFh
4Dh-4Ch	Reserved		X-base address bits				Y-base address bits	
	Reserved						SMI# connection	INTR enable/disable
4Fh-4Eh	Reserved				APIC RAM write enabled/disabled	MP7 pin functionality	MP6 pin functionality	Clock source for multiplexing/demultiplexing IRQs and DREQs
	Reserved				Pipelining with byte merge	EOP direction configuration	Byte merging	ISA master data swap
51h-50h	Reserved					Triggering for IRQ3	Triggering for IRQ4	Triggering for IRQ7
	Selects which IRQ signal is generated when PIRQ3# has been triggered		Selects which IRQ signal is generated when PIRQ2# has been triggered		Selects which IRQ signal is generated when PIRQ1# has been triggered		Selects which IRQ signal is generated when PIRQ0# has been triggered	
Index	**7**	**6**	**5**	**4**	**3**	**2**	**2**	**1**
Power Management Register Space								
E0h	SMI# Generation	Green event SMI# generation/status	Wake-up event SMI# generation/status	Green status bit	Hardware PPWRL# generation	Green event HW PPWRL# generation	Wake-up event HW PPWRL# generation	Software trigger Green event generation
E1h	GET CLK selection	GET CLK selection	EPMI polarity select	EPMI# debounce	Reserved	GET time-out Green event enable/status	EPMI# trigger Green event enable/status	Software trigger Green event enable/status
E2h	GET initial count bit 7	GET initial count bit 6	GET initial count bit 5	GET initial count bit 4	GET initial count bit 3	GET initial count bit 2	GET initial count bit 1	GET initial count bit 0
E3h	IRQ7 monitor	IRQ6 monitor	IRQ5 monitor	IRQ4 monitor	IRQ3 monitor	IRQ deglitch	IRQ1 monitor	IRQ0 monitor
E4h	IRQ15 monitor	IRQ14 monitor	IRQ13 monitor	IRQ12 monitor	IRQ11 monitor	IRQ10 monitor	IRQ9 monitor	IRQ8 monitor
E5h	DRQ7 monitor	DRQ6 monitor	DRQ5 monitor	Reserved	DRQ3 monitor	DRQ2 monitor	DRQ1 monitor	DRQ0 monitor
E6h	Programmable I/O address detection	LPT access detection	Video access detection	Hard disk detection	Floppy detection	Keyboard detection	COM1/COM3 access detection	COM2/COM4 access detection
E7h	LREQ# detection	LDEV# detection	EPMI# trigger, reload GET / wake-up event	Reserved	IO/MEM selection for Indices E8h, E9h	Mask bit 2 for programmable IO/MEM range	Mask bit 1 for programmable IO/MEM range	Mask bit 0 for programmable IO/MEM range

82C558 Register Space Summary (Cont.)

Index	7	6	5	4	3	2	2	1
E8h	Programmable IO/MEM address bit 7	Programmable IO/MEM address bit 6	Programmable IO/MEM address bit 5	Programmable IO/MEM address bit 4	Programmable IO/MEM address bit 3	Programmable IO/MEM address bit 2	Programmable IO/MEM address bit 1	Programmable IO/MEM address bit 0
E9h	Programmable IO/MEM address bit 15	Programmable IO/MEM address bit 14	Programmable IO/MEM address bit 13	Programmable IO/MEM address bit 12	Programmable IO/MEM address bit 11	Programmable IO/MEM address bit 10	Programmable IO/MEM address bit 9	Programmable IO/MEM address bit 8
EAh	GPP bit 7	GPP bit 6	GPP bit 5	GPP bit 4	GPP bit 3	GPP bit 2	GPP bit 1	GPP bit 0
EBh	NPP bit 7	NPP bit 6	NPP bit 5	NPP bit 4	NPP bit 3	NPP bit 2	NPP bit 1	NPP bit 0
ECh	PP bit 7	PP bit 6	PP bit 5	PP bit 4	PP bit 3	PP bit 2	PP bit 1	PP bit 0
EDh	GNR access SMI# enable/ status	LPT access SMI# enable/ status	Video access SMI# enable/ status	Hard disk access SMI# enable/status	Floppy disk access SMI# enable/status	Keyboard access SMI# enable/status	COM1/COM3 access SMI# enable/status	COM2/COM4 access SMI# enable/status
EEh	STPCLK# state enable	Reserved	Reserved	Reserved	STPCLK# modulation enable	STPCLK modulation bit 2	STPCLK modulation bit 1	STPCLK modulation bit 0
EFh	PREQ# enable wake-up	DRQ# enable wake-up	Reserved	GPCS1,2 generation	Reserved	Reserved	Reserved	Reserved
F0h	Device timer 1 CLK selection bit 1	Device timer 1 CLK selection bit 0	Device timer 0 CLK selection bit 1	Device timer 0 CLK selection bit 0	Timer 1 time-out green enable/status	Timer 0 time-out green enable/status	Device 1 cycle SMI# enable/status	Device 0 cycle SMI# enable/status
F1h	Device 0 timer initial count bit 7	Device 0 timer initial count bit 6	Device 0 timer initial count bit 5	Device 0 timer initial count bit 4	Device 0 timer initial count bit 3	Device 0 timer initial count bit 2	Device 0 timer initial count bit 1	Device 0 timer initial count bit 0
F2h	Device 1 timer initial count bit 7	Device 1 timer initial count bit 6	Device 1 timer initial count bit 5	Device 1 timer initial count bit 4	Device 1 timer initial count bit 3	Device 1 timer initial count bit 2	Device 1 timer initial count bit 1	Device 1 timer initial count bit 0
F3h	IO/MEM selection for device 1	Mask bit 2 for device 1 IO/ MEM address range	Mask bit 1 for device 1 IO/ MEM address range	Mask bit 0 for device 1 IO/ MEM address range	IO/MEM selection for device 0	Mask bit 2 for device 0 IO/ MEM address range	Mask bit 1 for device 0 IO/ MEM address range	Mask bit 0 for device 0 IO/ MEM address range
F4h	Device 0 IO/ MEM address bit 15	Device 0 IO/ MEM address bit 14	Device 0 IO/ MEM address bit 13	Device 0 IO/ MEM address bit 12	Device 0 IO/ MEM address bit 11	Device 0 IO/ MEM address bit 10	Device 0 IO/ MEM address bit 9	Device 0 IO/ MEM address bit 8
F5h	Device 0 IO/ MEM address bit 7	Device 0 IO/ MEM address bit 6	Device 0 IO/ MEM address bit 5	Device 0 IO/ MEM address bit 4	Device 0 IO/ MEM address bit 3	Device 0 IO/ MEM address bit 2	Device 0 IO/ MEM address bit 1	Device 0 IO/ MEM address bit 0
F6h	Device 1 IO/ MEM address bit 15	Device 1 IO/ MEM address bit 14	Device 1 IO/ MEM address bit 13	Device 1 IO/ MEM address bit 12	Device 1 IO/ MEM address bit 11	Device 1 IO/ MEM address bit 10	Device 1 IO/ MEM address bit 9	Device 1 IO/ MEM address bit 8
F7h	Device 1 IO/ MEM address bit 7	Device 1 IO/ MEM address bit 6	Device 1 IO/ MEM address bit 5	Device 1 IO/ MEM address bit 4	Device 1 IO/ MEM address bit 3	Device 1 IO/ MEM address bit 2	Device 1 IO/ MEM address bit 1	Device 1 IO/ MEM address bit 0

6.0 Maximum Ratings

Stresses above those listed in the following tables may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification are not implied.

6.1 Absolute Maximum Ratings

Symbol	Parameter	5.0 Volt		3.3 Volt		Unit
		Min	Max	Min	Max	
VCC	5.0V Supply Voltage		+6.5			V
VDD	3.3V Supply Voltage				+4.0	
VI	Input Voltage	-0.5	VCC + 0.5	-0.5	VDD + 0.5	V
VO	Output Voltage	-0.5	VCC + 0.5	-0.5	VDD + 0.5	V
TOP	Operating Temperature	0	+70	0	+70	°C
TSTG	Storage Temperature	-40	+125	-40	+125	°C

6.2 DC Characteristics: 5.0 Volt (VCC = 5.0V ±5%, TA = 0°C to +70°C)

Symbol	Parameter	Min	Max	Unit	Condition
VIL	Input low Voltage	-0.5	+0.8	V	
VIH	Input high Voltage	+2.0	VCC + 0.5	V	
VOL	Output low Voltage		+0.4	V	IOL = 4.0mA
VOH	Output high Voltage	+2.4		V	IOH = -1.6mA
IIL	Input Leakage Current		+10.0	µA	VIN = VCC
IOZ	Tristate Leakage Current		+10.0	µA	
CIN	Input Capacitance		+10.0	pF	
COUT	Output Capacitance		+10.0	pF	
ICC	Power Supply Current		240	mA	In a 60/90MHz system

6.3 DC Characteristics: 3.3 Volt (VDD = 3.3V ±5%, TA = 0°C to +70°C)

Symbol	Parameter	Min	Max	Unit	Condition
VIL	Input low Voltage	-0.5	+0.8	V	
VIH	Input high Voltage	+2.0	VDD + 0.5	V	
VOL	Output low Voltage		+0.4	V	IOL = 4.0mA
VOH	Output high Voltage	+2.4		V	IOH = -1.6mA
IIL	Input Leakage Current		+10.0	µA	VIN = VDD
IOZ	Tristate Leakage Current		+10.0	µA	
CIN	Input Capacitance		+10.0	pF	
COUT	Output Capacitance		+10.0	pF	
ICC	Power Supply Current		115	mA	In a 60/90MHz system

- Average power dissipation for a system running at 60/90MHz:
  - 82C556 = 400mW
  - 82C557 = 600mW
  - 82C558 = 600mW

6.4 82C556 AC Characteristics (66MHz - Preliminary)

Symbol	Parameter	Min	Max	Unit	Condition
t101	HD[63:0] to MD[63:32] bus valid	2	22	ns	
t102	HD[31:0] to MD[31:0]# bus valid	2	22	ns	
t103	DLE1# high to HD[63:32] bus valid	2	22	ns	
t104	DLE0# high to HD[31:0] bus valid	2	22	ns	
t105	DLE1# high to MPERR# valid	2	22	ns	
t106	DLE0# high to MPERR# valid	2	22	ns	
t107	MD[31:0]# setup to DLE0# high	5		ns	
t108	MD[63:32] setup to DLE0# high	5		ns	
t109	MD[63:32] setup to DLE1# high	5		ns	
t110	HD[31:0] setup to DLE0# high	5		ns	
t111	HD[63:32] setup to DLE0# high	5		ns	
t112	MP[3:0]# setup to DLE0# high	5		ns	
t113	MP[7:4] setup to DLE0# high	5		ns	
t114	MP[7:4] setup to DLE1# high	5		ns	
t115	MD[31:0]# hold from DLE0# high	5		ns	
t116	MD[63:32] hold from DLE0# high	5		ns	
t117	MD[63:32] hold from DLE1# high	5		ns	
t118	HD[31:0] hold from DLE0# high	8		ns	
t119	HD[63:32] hold from DLE0# high	8		ns	
t120	MP[3:0]# hold from DLE0# high	5		ns	
t121	MP[7:4] hold from DLE0# high	5		ns	
t122	MP[7:4] hold from DLE1# high	5		ns	
t123	HDOE# high to HD[63:32] high-Z	2	11	ns	
t124	HDOE# high to HD[31:0] high-Z	2	11	ns	
t125	MDOE# high to MD[63:32] high-Z	2	15	ns	
t126	MDOE# high to MD[31:0]# high-Z	2	15	ns	

6.5 82C557 AC Characteristics (66MHz - Preliminary)

Symbol	Parameter	Min	Max	Unit	Condition
t201	ECLK to BRDY# active delay	5	15	ns	
t202	ECLK to BRDY# inactive delay	5	15	ns	
t203	ECA4, OCA4 delay from ECLK rising	5	15	ns	
t204	HACALE delay from ECLK rising	5	15	ns	
t205	ECDOE#, even bank cache, falling edge valid delay from ECLK rising	5	15	ns	
t206	OCDOE#, odd bank cache, falling edge valid delay from ECLK rising	5	15	ns	
t207	ADS# setup to CLK high	2		ns	
t208	ADS# hold time from CLK high	1		ns	
t209	M/IO#, D/C#, W/R#, CACHE# setup to CLK high	1		ns	Sampled one CLK after ADS#
t210	ECLK to DIRTYWE# active delay	5	14	ns	
t211	ECLK to DIRTYWE# inactive delay	5	14	ns	
t212	ECLK to TAGWE# active delay	5	14	ns	
t213	ECLK to TAGWE# inactive delay	5	14	ns	
t214	ECAWE#, even bank cache, falling edge valid delay from ECLK high	5	15	ns	
t215	OCAWE#, odd bank cache, falling edge valid delay from ECLK high	5	15	ns	
t216	ECLK to NA# active delay	5	15	ns	
t217	ECLK to NA# inactive delay	5	15	ns	
t218	TAG[7:0] data read to BRDY# low		5	ns	
t219	ECLK to ADSC# active delay (for sync. SRAM)	5	15	ns	
t220	ECLK to ADV# active delay (for sync. SRAM)	5	15	ns	
t221	ECLK to SYNCS0#, SYNCS1# active delay (for sync. SRAM)	5	15	ns	
t222	ECLK to CAWE[7:0]# active delay (for sync. SRAM)	5	15	ns	
t223	HA[31:3] valid delay from LCLK high	2	18	ns	
t224	HA[31:3] Float delay from LCLK high	2	18	ns	
t225	AHOLD valid delay from CLK high	5	15	ns	
t226	EADS# valid delay from CLK high	5	15	ns	
t227	RESET rising edge valid from CLK high	5	15	ns	
t228	RESET falling edge valid delay from CLK high	5	15	ns	
t229	KEN#/LMEM# valid delay from ECLK high	5	15	ns	
t230	RAS[3:0]# valid delay from CPUCLK high/LCLK high	2	15	ns	

82C557 AC Characteristics (66MHz - Preliminary) (Cont.)

Symbol	Parameter	Min	Max	Unit	Condition
t231	CAS[7:0]# valid delay from CPUCLK high/LCLK high	2	15	ns	
t232	MA[11:0] valid delay from CPUCLK high/LCLK high	2	15	ns	
t233	DWE# valid delay from CPUCLK high/LCLK high	2	15	ns	
t234	MA[11:0] propagation delay from HA[28:3]	2	22	ns	
t235	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, PLOCK#, DEVSEL# valid delay from LCLK rising	2	11	ns	
t236	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, PLOCK#, DEVSEL# active to float delay from LCLK rising	2	15	ns	
t237	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, DEVSEL# setup time to LCLK rising	7		ns	
t238	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, DEVSEL# hold time from LCLK rising	0		ns	
t239	AD[31:0] valid delay from LCLK high	2	11	ns	
t240	AD[31:0] setup time to LCLK high	7		ns	
t241	AD[31:0] hold time from LCLK high	0		ns	
t242	LRDY# setup time to LCLK high	5		ns	
t243	LRDY# hold time from LCLK high	2		ns	
t244	CLK delay from ECLK	3	6	ns	
t245	DBCOE[1:0]#, MDOE#, HDOE# valid delay from CLK/ LCLK high	2	15	ns	
t246	DLE[1:0]# valid delay from CLK\ LCLK high	2	15	ns	
t247	MDLE# valid delay from CLK\LCLK high		15	ns	
t248	PEN# valid delay from CLK\LCLK high		15	ns	
t249	NVMCS delay from CLK (2nd or 3rd T2)	2	35	ns	
t250	LA[23:9] valid delay from CLK high (2nd or 3rd T2)	2	25	ns	
t251	HREQ setup time to CLK high	2		ns	
t252	HOLD valid delay from CLK high	5	15	ns	
t253	HLDA setup time to CLK high	2		ns	
t268	HLDA hold time from CLK high	1		ns	

6.6 82C558 AC Characteristics (66MHz - Preliminary)

Symbol	Parameter	Min	Max	Unit	Condition
t301	FRAME#, TRDY#, IRDY#, STOP#, DEVSEL#, PAR, SERR#, PERR# valid delay from LCLK rising	2	11	ns	
t302	PGNT[2:0]# valid delay from LCLK rising	2	12	ns	
t303	MP7/LADS#, PIRQ[3:0]#/LRDY# valid delay from LCLK rising	2	16	ns	
t304	MD[63:32] valid delay from LCLK rising	2	20	ns	
t305	FRAME#, TRDY#, IRDY#, STOP#, DEVSEL#, PAR, SERR#, PERR# float delay from LCLK rising	2	20	ns	
t306	C/BE[3:0]#, AD[31:0], FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, PLOCK#, PAR, SERR#, PERR# setup time to LCLK rising	7		ns	
t307	C/BE[3:0]#, AD[31:0], FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, PLOCK#, PAR, SERR#, PERR# hold time from LCLK rising	0		ns	
t308	PREQ[2:0]# setup time to LCLK rising	12		ns	
t309	PREQ[2:0]# hold time from LCLK rising	0		ns	
t310	PIRQ[3:0]#/LRDY# setup time to LCLK rising	5		ns	
t311	PIRQ[3:0]#/LRDY# hold time from LCLK rising	3		ns	
t312	LMEM# setup time to LCLK rising	5		ns	
t313	INIT valid delay from LCLK rising	2	15	ns	
t314	ATCLK rising edge delay from LCLK rising edge	5	20	ns	
t315	SMI# valid delay from LCLK rising	2	15	ns	
t316	IOR#, IOW# high valid delay from ATCLK rising		15	ns	
t317	MEMR#, MEMW#, SMEMR#, SMEMW# valid delay from ATCLK rising		15	ns	
t318	BALE low valid delay from ATCLK rising		15	ns	
t319	XDIR valid delay from ATCLK rising		15	ns	
t320	STPCLK# valid delay from ATCLK rising		15	ns	
t321	RTCAS, RTCRD#, RTCWR#, ROMCS#/KBDCS# valid delay from ATCLK rising		15	ns	
t322	BALE high valid delay from ATCLK falling		15	ns	
t323	IOR#, IOW#, MEMR#, MEMW#, SMEMR#, SMEMW# low valid delay from ATCLK falling		15	ns	
t324	PPWRL# valid delay from ATCLK falling		15	ns	
t325	ZEROWS# setup time to ATCLK falling	0		ns	
t326	ZEROWS# hold time from ATCLK falling	5		ns	
t327	IOCHRDY setup time to ATCLK falling	5		ns	
t328	IOCHRDY hold time from ATCLK falling	5		ns	
t329	MD[63:32] setup time to MDLE# rising	5		ns	

82C558 AC Characteristics (66MHz - Preliminary) (Cont.)

Symbol	Parameter	Min	Max	Unit	Condition
t330	MD[63:32] hold time from MDLE# rising	5		ns	
t331	MPERR# setup time to PEN# rising	3		ns	
t332	MPERR# hold time from PEN# rising	3		ns	

6.7 AC Timing Diagrams

Figure 6-1 Setup Timing Waveform

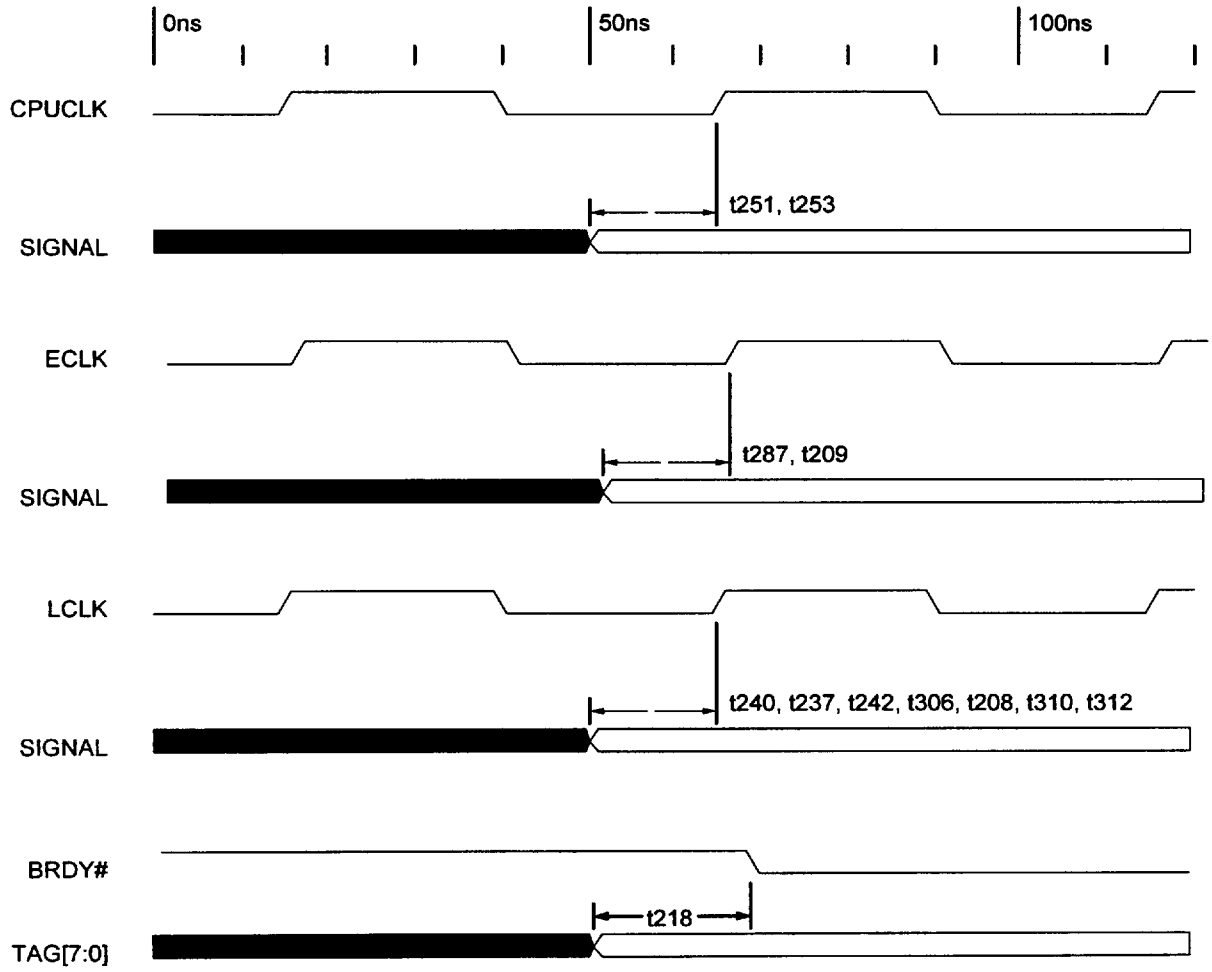


Figure 6-2 Hold Timing Waveform

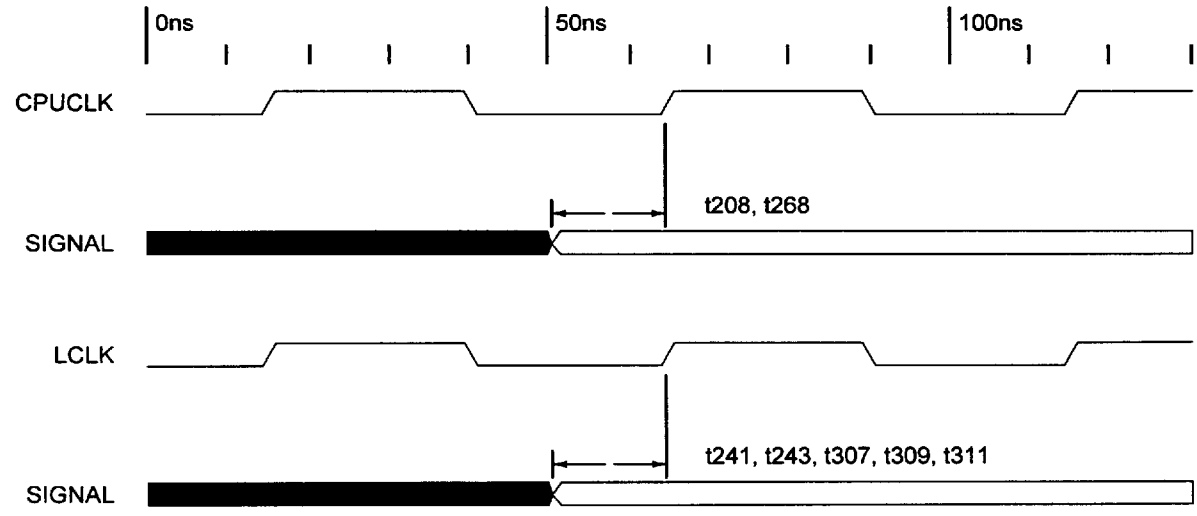


Figure 6-3 Output Delay Timing Waveform

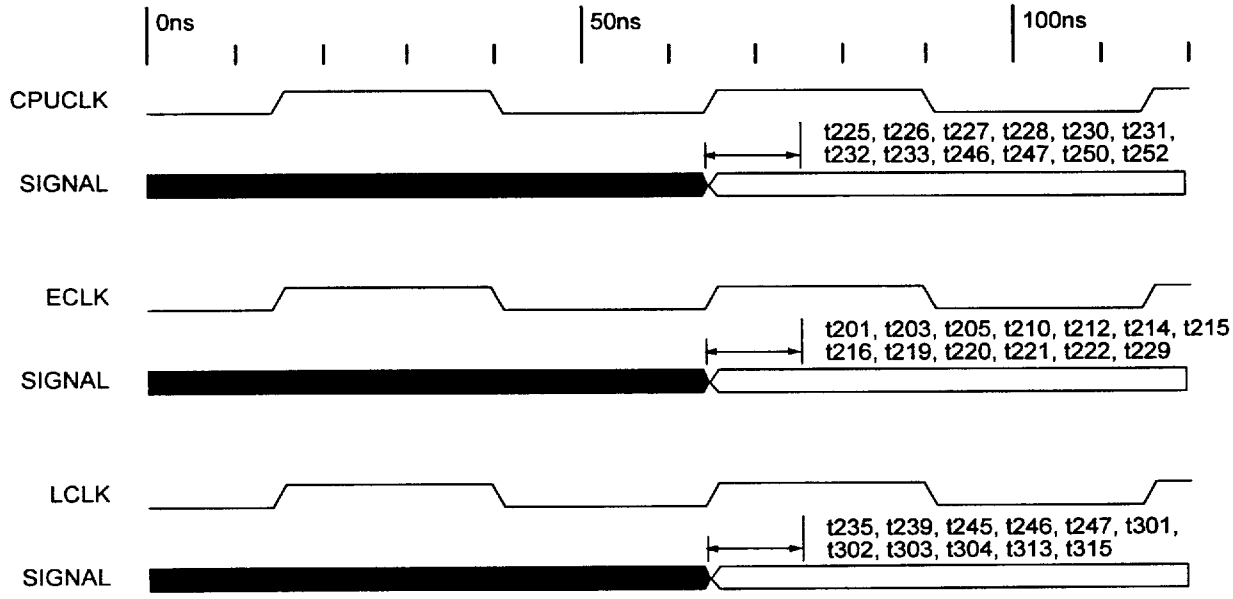
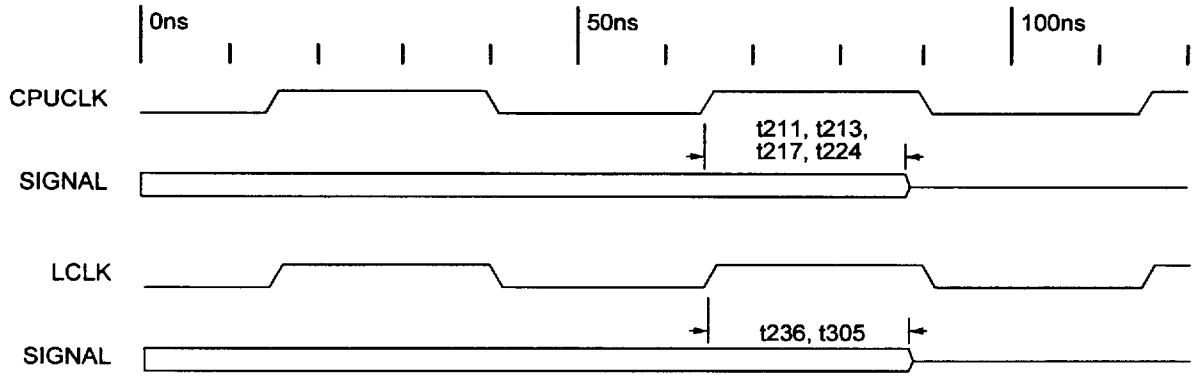
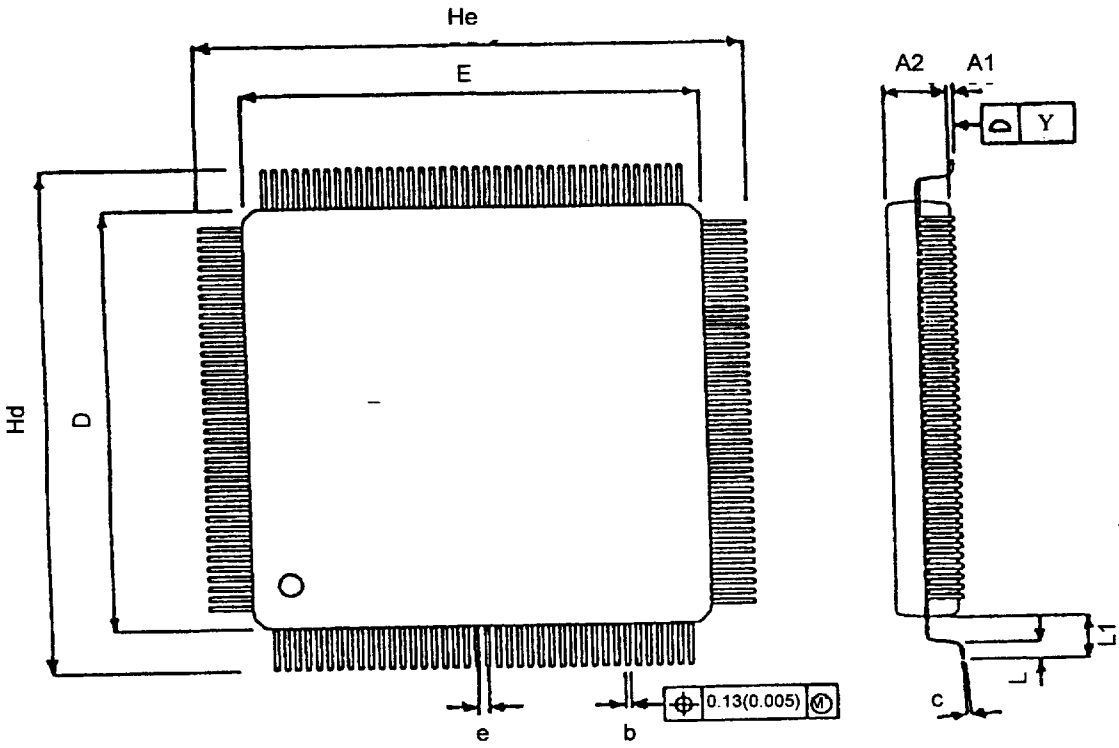


Figure 6-4 Float Delay Timing Waveform



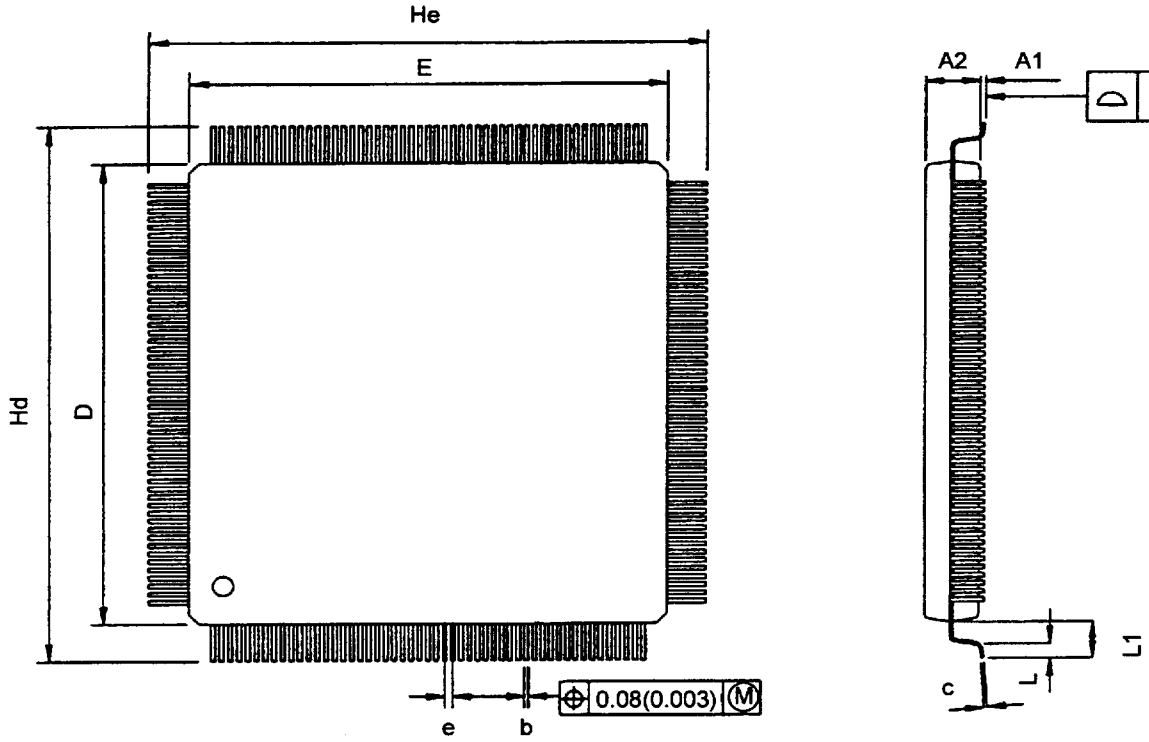
7.0 Mechanical Package Outlines

Figure 7-1 160-Pin Plastic Quad Flat Pack (PQFP)



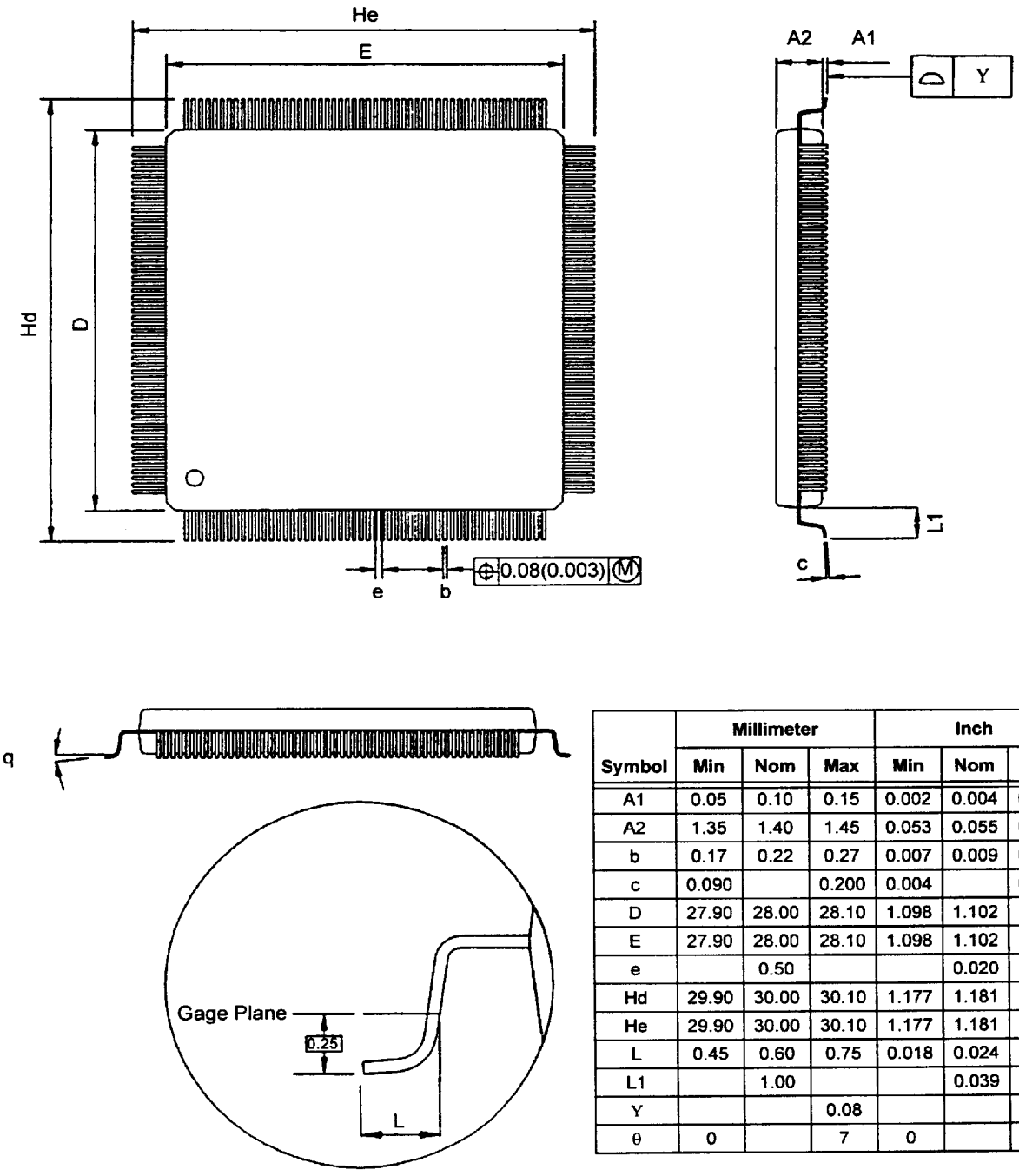
Symbol	Millimeter			Inch		
	Min	Nom	Max	Min	Nom	Max
A1	0.05	0.25	0.50	0.002	0.010	0.020
A2	3.17	3.32	3.47	0.125	0.131	0.137
b	0.20	0.30	0.40	0.008	0.012	0.016
c	0.10	0.15	0.20	0.004	0.006	0.008
D	27.90	28.00	28.10	1.098	1.102	1.106
E	27.90	28.00	28.10	1.098	1.102	1.106
e		0.65			0.026	
Hd	31.65	31.90	32.15	1.246	1.256	1.266
He	31.65	31.90	32.15	1.246	1.256	1.266
L	0.65	0.80	0.95	0.025	0.031	0.037
L1		1.95			0.077	
Y			0.08			0.003
θ	0		10	0		10

Figure 7-2 208-Pin Plastic Quad Flat Package (PQFP)



Symbol	Millimeter			Inch		
	Min	Nom	Max	Min	Nom	Max
A1	0.05	0.25	0.50	0.002	0.010	0.020
A2	3.17	3.32	3.47	0.125	0.131	0.137
b	0.10	0.20	0.30	0.004	0.008	0.012
c	0.10	0.15	0.20	0.004	0.006	0.008
D	27.90	28.00	28.10	1.098	1.102	1.106
E	27.90	28.00	28.10	1.098	1.102	1.106
e		0.50			0.020	
Hd	30.35	30.60	30.85	1.195	1.205	1.215
He	30.35	30.60	30.85	1.195	1.205	1.215
L	0.35	0.50	0.65	0.014	0.020	0.026
L1		1.30			0.051	
Y			0.08			0.003
θ	0		10	0		10

Figure 7-3 208-Pin Thin Quad Flat Pack (TQFP)



Symbol	Millimeter			Inch		
	Min	Nom	Max	Min	Nom	Max
A1	0.05	0.10	0.15	0.002	0.004	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.17	0.22	0.27	0.007	0.009	0.011
c	0.090		0.200	0.004		0.008
D	27.90	28.00	28.10	1.098	1.102	1.106
E	27.90	28.00	28.10	1.098	1.102	1.106
e		0.50			0.020	
Hd	29.90	30.00	30.10	1.177	1.181	1.185
He	29.90	30.00	30.10	1.177	1.181	1.185
L	0.45	0.60	0.75	0.018	0.024	0.030
L1		1.00			0.039	
Y			0.08			0.003
θ	0		7	0		7