OPTi

82C556M/82C557M/82C558M

Viper-M (Multimedia) Chipset

Preliminary Data Book

Revision: 1.0
912-2000-010
April 1995

OPTi Inc.
2525 Walsh Avenue
Santa Clara, CA 95051
Tel: (408) 980-8178
Fax: (408) 980-8860
BBS: (408) 980-9774

Table of Contents

1.0 Features 1

2.0 Overview 2

2.1 82C556M (DBC) Data Buffer Controller 3

2.2 82C557M (SYSC) System Controller 3

2.3 82C558M (IPC) Integrated Peripherals Controller 3

3.0 Signal Definitions 6

3.1 Signal Description Definitions 6

3.2 82C556M Signal Descriptions 11

3.2.1 CPU Bus Interface Signals 11

3.2.2 82C557M Interface Signals 11

3.2.3 DRAM Interface Signals 12

3.2.4 Power and Ground Pins 12

3.3 82C557M Signal Descriptions 19

3.3.1 Reset and Clock Signals 19

3.3.2 CPU Bus Interface Signals 19

3.3.3 Cache Control Interface Signals 21

3.3.4 DRAM Interface Signals 22

3.3.5 AT Bus Interface Signals 23

3.3.6 Bus Arbiter Interface Signals 24

3.3.7 PCI/VL Bus Interface Signals 25

3.3.8 82C556M/82C558M Interface Signals 26

3.3.9 Power and Ground Pins 27

3.4 82C558M Signal Descriptions 35

3.4.1 Reset and Clock Signals 35

3.4.2 IDE Interface Signals 35

3.4.3 82C557M and 82C556M Interface Signals 38

3.4.4 PCI Bus Interface Signals 38

3.4.5 CPU Interface Signals 40

3.4.6 ISA Bus Interface Signals 41

Table of Contents (cont.)

3.4.7 ISA DMA Arbiter Signals 43
3.4.8 Interrupt Control Signals 45
3.4.9 RTC and Timer Signals 46
3.4.10 Miscellaneous Signals 47
3.4.11 Power and Ground Pins 47

4.0 Functional Description 52

4.1 Reset Logic 52

4.2 System Clocks 52

4.2.1 CPU and 82C557M Clocks 52
4.2.2 PCI and VL Bus Clocks 53
4.2.3 AT Bus Clocks 53

4.3 Cache Subsystem 54

4.3.1 CPU Burst Mode Control 54
4.3.1.1 Cyrix Linear Burst Mode Support 54
4.3.2 Cache Cycle Types 54
4.3.3 Cache Operation 55
4.3.3.1 L2 Cache Read Hit 55
4.3.3.2 L2 Cache Write Hit Cycle 59
4.3.3.3 L2 Cache Read Miss 62
4.3.3.4 L2 Cache Write Miss 66
4.3.3.5 Adaptive Write-Back Policy 68
4.3.3.5.1 Write-Through on Page Hit and RAS# Active (AWB Mode 1) 68
4.3.3.5.2 Write-Through on Page Hit (AWB Mode 2) 68
4.3.3.6 Tag Compare Table 68
4.3.3.7 Tag and Dirty RAM implementations 69
4.3.3.7.1 Separate Tag/Dirty RAM Implementation 69
4.3.3.7.2 Combined Tag/Dirty RAM Implementation 69
4.3.3.8 Cache Initialization 71
4.3.3.9 Write Back Cache with DMA/ISA Master/PCI Master Operation 71
4.3.3.10 Cacheability and Write Protection 76
4.3.4 Synchronous SRAM Support 77
4.3.4.1 Pipelined Synchronous SRAM support 77
4.3.4.2 SONY SONIC-2WP (Cache Module) Support 77
4.3.5 SRAM Requirements 77

Table of Contents (cont.)

4.4 DRAM Controller 80
4.4.1 EDO Support 80
4.4.2 Programming the DRAM Parameters 80
4.4.2.1 Number of DRAM banks 80
4.4.2.2 DRAM Configurations 81
4.4.2.3 Timing Parameters 81
4.4.2.4 Drive Strengths 81
4.4.3 DRAM Cycles 81
4.4.3.1 DRAM Read Cycle 81
4.4.3.2 DRAM Write Cycle 82
4.4.3.3 DRAM Parity Generation/Detection Logic 85
4.4.3.4 DRAM Refresh Logic 85
4.4.4 DRAM Address Muxing 85
4.4.5 DRAM DMA/Master Cycles 85
4.5 PCI Bus Interface 87
4.5.1 PCI Master Cycles 87
4.5.1.1 System Memory Access 87
4.5.1.2 Non-Local Memory Access 87
4.5.1.3 PCI Master Pre-snoop 88
4.5.2 PCI Slave Cycles 88
4.5.2.1 CPU Master Cycles 88
4.5.2.2 PCI Byte/Word Merge 88
4.5.2.3 ISA Master Cycles 88
4.6 VL Bus Interface 88
4.7 AT Bus Interface 99
4.8 XD Bus Interface 99
4.9 Bus Arbitration Logic 99
4.10 Data Bus Conversion/Data Path Control Logic 99
4.11 Internal Integrated 82C206 100
4.11.1 Top Level Decoder and Configuration Register 100
4.11.2 DMA Subsystem 100
4.11.2.1 DMA Operation 100
4.11.2.1.1 Idle Mode 100
4.11.2.1.2 Program Mode 100
4.11.2.1.3 Active Mode 102

Table of Contents (cont.)

4.11.2.2 DMA Transfer Modes 102
4.11.2.2.1 Single Transfer Mode 102
4.11.2.2.2 Block Transfer Mode 102
4.11.2.2.3 Demand Transfer Mode 102
4.11.2.2.4 Cascade Mode 102
4.11.2.3 Transfer Types 103
4.11.2.3.1 Auto-initialization 104
4.11.2.3.2 DREQ Priority 104
4.11.2.3.3 Address Generation 104
4.11.2.3.4 Compressed Timing 105
4.11.3 DMA Register Descriptions 105
4.11.3.1 Current Address Register 105
4.11.3.2 Current Word Count Register 105
4.11.3.3 Base Address Register 105
4.11.3.4 Base Word Count Register 105
4.11.3.5 Command Register 105
4.11.3.6 Mode Register 106
4.11.3.7 Request Register 106
4.11.3.8 Request Mask Register 106
4.11.3.9 Status Register 106
4.11.3.10 Temporary Register 106
4.11.4 Special Commands 106
4.11.5 Interrupt Controller Subsystem 107
4.11.5.1 Interrupt Controller Subsystem Overview 107
4.11.5.2 Interrupt Controller Operation 108
4.11.5.3 Interrupt Sequence 108
4.11.5.4 End of Interrupt (EOI) 109
4.11.5.5 Priority Assignment 109
4.11.5.5.1 Fixed Priority Mode 109
4.11.5.5.2 Specific Rotation Mode 109
4.11.5.5.3 Automatic Rotation Mode 109
4.11.5.6 Programming the Interrupt Controller 110
4.11.5.6.1 Initialization Command Words (ICWs) 110
4.11.5.6.2 Operational Command Words (OCWs) 111
4.11.5.6.3 IRR, ISR, & Poll Vector 111
4.11.6 Counter/Timer Subsystem 112
4.11.6.1 Counter Description 112
4.11.6.1.1 Programming the Counter/Timer 113
4.11.6.1.2 Read/Write Counter Command 113
4.11.6.1.3 Counter Latch Command 113
4.11.6.1.4 Read-Back Command 113

Table of Contents (cont.)

4.11.6.2 Counter Operation 113
4.11.6.2.1 Mode 0 - Interrupt on Terminal Count 114
4.11.6.2.2 Mode 1 - Hardware Retriggerable One-Shot 114
4.11.6.2.3 Mode 2 - Rate Generator 114
4.11.6.2.4 Mode 3 - Square Wave Generator 114
4.11.6.2.5 Mode 4 - Software Triggered Strobe 114
4.11.6.2.6 Mode 5 - Hardware Triggered Strobe 114
4.11.6.2.7 GATE2 114

4.12 Power Management 116

4.12.1 Definition of Power Management Modes 116
4.12.1.1 Normal Mode 116
4.12.1.2 AUTO_GREEN Mode 116
4.12.1.3 SMI_GREEN Mode 116

4.12.2 System Activity Detection 116
4.12.2.1 GREEN Events 116
4.12.2.2 Reload GET/Wake-up Events 117

4.12.3 System Management Interrupt (SMI) 117
4.12.3.1 SMI Implementation 117

4.12.4 Hardware Power Management Support without an External Latch 117

4.13 Fast GATEA20 and Reset Emulation 118

4.14 Shadow ROM & BIOS Cacheability 119

4.15 Scatter/Gather 119

4.16 Type F DMA 119

4.17 IDE Interface 120

4.17.1 Overview 120

4.17.2 MIDE Register Descriptions 120
4.17.2.1 MIDE Configuration Registers 120
4.17.2.2 I/O Registers 126
4.17.2.2.1 I/O Registers for Primary IDE 126
4.17.2.2.2 I/O Registers for Secondary IDE 129

4.17.3 Physical Region Descriptor Table 132
4.17.3.1 Physical Region Descriptor 132
4.17.3.2 Bus Master IDE Registers 132

Table of Contents (cont.)

4.17.4 Programming the MIDE Module Registers 136
4.17.4.1 Standard Programming Sequence for Bus Mastering Operations 136
4.17.4.2 Programming the IDE Mode Timing 136
4.17.4.3 Programming the IDE Interrupt Routing 136

4.18 Special Cycles 141
4.18.1 System ROM BIOS Cycles 141
4.18.2 System Shutdown/Halt Cycles 141

5.0 Register Descriptions 142

5.1 82C557M Register Space 143
5.1.1 82C557M PCI Configuration Register Space 143
5.1.2 82C557M System Control Register Space 146
5.1.3 82C557M I/O Registers 166

5.2 82C558M Register Space 167
5.2.1 82C558M PCI Configuration Register Space 167
5.2.2 82C558M System Control Register Space 170
5.2.3 82C558M I/O Registers 182
5.2.4 82C558M Internal Integrated 82C206 Register Descriptions 184
5.2.4.1 DMA Subsystem Registers 185
5.2.4.2 Interrupt Controller Subsystem 187
5.2.4.3 Counter/Timer Subsystem 191
5.2.5 82C558M Power Management Registers 194

6.0 Maximum Ratings 211

6.1 Absolute Maximum Ratings 211
6.2 DC Characteristics: 5.0 Volt (VCC = 5.0V ±5%, TA = 0°C to +70°C) 211
6.3 DC Characteristics: 3.3 Volt (VDD = 3.3V ±5%, TA = 0°C to +70°C) 211
6.4 82C556M AC Characteristics (66MHz - Preliminary) 212
6.5 82C557M AC Characteristics (66MHz - Preliminary) 213
6.6 82C558M AC Characteristics (66MHz - Preliminary) 215
6.7 AC Timing Diagrams 216

Table of Contents (cont.)

7.0 Test Mode Information 219

7.1 82C556M Testability 219

7.1.1 82C556M NAND Tree Test (Test Mode 1) 219

7.2 82C557M Testability 219

7.2.1 82C557M NAND Tree Test (Test Mode 1) 219

7.3 82C558M Testability 222

7.3.1 82C558M NAND Tree Test (Test Mode 1) 222

7.3.2 Drive High/Drive Low Test (Test Modes 2 and 3) 222

8.0 Mechanical Package Outlines 225

A.0 Accessing the BBS 229

A.1 Paging the SYSOP 229

A.2 System Requirements 229

A.3 Calling In/Hours of Operation 229

A.4 Logging On for the First Time 229

A.5 Log On Rules and Regulations 229

A.6 Using the BBS 229

A.6.2 Sending/Receiving Messages 230

A.6.3 Finding Information 230

A.6.4 Downloading Files From OPTi 230

A.6.5 Uploading Files To OPTi 230

A.6.6 Logging Off 230

A.6.7 Logging Back on Again 230

A.7 The Menus 230

A.7.1 Menu Selections 231

List of Figures

Figure 1-1	System Block Diagram	1
Figure 2-1	82C556M Block Diagram	3
Figure 2-2	82C557M Block Diagram	4
Figure 2-3	82C558M Block Diagram	5
Figure 3-1	82C556M Pin Diagram	7
Figure 3-2	82C557M Pin Diagram	15
Figure 3-3	82C558M Pin Diagram	30
Figure 3-4	PCI Interrupts Mapping Matrix	48
Figure 4-1	CPU and 82C557M Clock Distribution	52
Figure 4-2	Clock Distribution Method for VL Bus and PCI Connectors (Async. PCI and VL Bus)	53
Figure 4-3	Clock Distribution Method for VL Bus and PCI Connectors (Sync. PCI and VL Bus)	53
Figure 4-4	L2 Cache Read Hit Cycle - Async. SRAMs (Double Bank)	56
Figure 4-5	L2 Cache Read Hit Cycle - Sync. SRAMs	57
Figure 4-6	L2 Cache Read Hit Cycle Async. SRAMs (Single Bank)	58
Figure 4-7	Write Hit Burst Cycle for Write-Back Mode - Async. SRAM (Double Bank)	59
Figure 4-8	Write Hit Burst Cycle for Write-Back Mode (Single Bank) - Async. SRAM	60
Figure 4-9	Write Hit Burst Cycle for Write-Back Mode - Sync. SRAM	61
Figure 4-10	L2 Cache Read Miss Clean Burst of 8-3-3-3 (Linefill Cycle) - Async. SRAMs	63
Figure 4-11	L2 Cache Read Miss Clean Burst of 8-3-3-3 (Linefill Cycle) - Sync. SRAMs	64
Figure 4-12	L2 Cache Read Miss Dirty Cycle - Async. SRAMs	65
Figure 4-13	L2 Cache Write Miss Burst Cycle (4-3-3-3) - Async. SRAMs	66
Figure 4-14	L2 Cache Write Miss Burst Cycle - Sync. SRAMs	67
Figure 4-15	Block Diagram of Internal Tag Comparator	68
Figure 4-16	Separate 32Kx8 and 32Kx1 Split Tag/Dirty RAM Implementation	69
Figure 4-17	32Kx9 Combined Tag/Dirty RAM Implementation	70
Figure 4-18	32Kx8 Combined Tag/Dirty RAM Implementation	70
Figure 4-19	32Kx8 and 32Kx8 Combined Tag/Dirty RAM Implementation (Separate Devices)	70
Figure 4-20	AT DMA/Master Read (L1 cache with non-modified line)	73
Figure 4-21	AT DMA/Master Read (L1 cache with modified line)	74
Figure 4-22	AT DMA/Master Write (L1 cache with modified line)	75
Figure 4-23	DRAM Read Page Miss with RAS Active Read Cycle	83
Figure 4-24	DRAM Page Miss with RAS Active Write Cycle	84

List of Figures (cont.)

Figure 4-25	AT Master Synchronization	86
Figure 4-26	PCI Master Read from Local Memory	89
Figure 4-27	CPU Read from PCI, 32 Bits	90
Figure 4-28	CPU Write to PCI, 32 Bits	91
Figure 4-29	CPU Read from PCI, 64 Bits	92
Figure 4-30	CPU Read from VESA Slave	93
Figure 4-31	CPU Write to VESA Slave	94
Figure 4-32	ISA Master Read from PCI	95
Figure 4-33	ISA Master Write to PCI	96
Figure 4-34	ISA Master Read from ISA Slave	97
Figure 4-35	ISA Master Write to ISA Slave	98
Figure 4-36	Cascade Mode Interconnect	103
Figure 4-37	Internal Cascade Interconnect	107
Figure 4-38	Interrupt Controller Block Diagram	108
Figure 4-39	Interrupt Sequence	108
Figure 4-40	Initialization Sequence	110
Figure 4-41	Counter/Timer Block Diagram	112
Figure 4-42	Connections with Keyboard Emulation disabled	118
Figure 4-43	MIDE Interface Primary Channel Programming Flow Chart	140
Figure 6-1	Setup Timing Waveform	216
Figure 6-2	Hold Timing Waveform	217
Figure 6-3	Output Delay Timing Waveform	217
Figure 6-4	Float Delay Timing Waveform	218
Figure 8-1	160-Pin Plastic Quad Flat Pack (PQFP)	225
Figure 8-2	208-Pin Plastic Quad Flat Package (PQFP)	226
Figure 8-3	208-Pin Thin Quad Flat Pack (TQFP)	227
Figure A-1	The Main Menu	230
Figure A-2	The Bulletin Menu	231
Figure A-3	The File Menu	231
Figure A-4	The Message Menu	231

List of Tables

Table 3-1	82C556M 160-Pin Numerical Pin Cross-Reference List	8
Table 3-2	82C556M 160-Pin Alphabetical Cross-Reference List	10
Table 3-3	82C556M Strap Pin Mapping Functions	13
Table 3-4	82C556M Operating Voltage Groups	13
Table 3-5	82C556M Encoded Commands	14
Table 3-6	82C557M Numerical Pin Cross-Reference List	16
Table 3-7	82C557M Alphabetical Pin Cross-Reference List	18
Table 3-8	82C557M Multiplexed Function Pins	28
Table 3-9	82C557M Strap Function Pins	28
Table 3-10	82C557M System Operating Voltage Groups	29
Table 3-11	82C558M Numerical Pin Cross-Reference List	31
Table 3-12	82C558M Alphabetical Pin Cross-Reference List	34
Table 3-13	82C558M Strap Function Pins	48
Table 3-14	82C558M Multiplexed Function Pins	49
Table 3-15	82C558M Programmable Functions Pins	50
Table 3-16	82C558M Operating Voltage Groups	51
Table 4-1	Burst Modes	54
Table 4-2	Tag Compare Table	68
Table 4-3	DMA/Master Read Cycle Summary	72
Table 4-4	DMA/Master Write Cycle Summary	72
Table 4-5	Cacheability Methods	76
Table 4-6	SRAM Requirements	77
Table 4-7	Signal Functionality for Synchronous SRAM Implementation	77
Table 4-8	Data SRAM Asynchronous Configurations	78
Table 4-9	Data SRAM (Asynchronous) and Tag SRAM Speed Requirements	78
Table 4-10	Data SRAM (Synchronous) Configurations	78
Table 4-11	Data SRAM (Synchronous) Speed Requirements	78
Table 4-12	Tag SRAM Speed Requirements for Synchronous SRAMs	78
Table 4-13	SRAM Comparisons	79
Table 4-14	DRAM Programmable Control	81
Table 4-15	DRAM Timing Mode Summary	82
Table 4-16	DRAM Row/Column MA to Address Bit Map	86

List of Tables (cont.)

Table 4-17	DMA I/O Address Map	101
Table 4-18	Rotating Priority Scheme	104
Table 4-19	DMA Page Register I/O Address Map	105
Table 4-20	Interrupt Request Source	107
Table 4-21	Interrupt Vector Byte	109
Table 4-22	Counter/Timer I/O Address Map	113
Table 4-23	GATE2 Pin Function	115
Table 4-24	Vendor ID Register - 01h-00h	120
Table 4-25	Device ID Register - 03h-02h	120
Table 4-26	Command Register - 05h-04h	121
Table 4-27	Status Register - 07h-06h	121
Table 4-28	Revision ID Register - 08h	121
Table 4-29	Class Code Register - 0Bh-09h	122
Table 4-30	Reserved Register - 0Dh-0Ch	122
Table 4-31	Header Type Register - 0Eh	122
Table 4-32	Built-In Self-Test Register - 0Fh	122
Table 4-33	Command Block Base Address Register - 13h-10h (Primary IDE)	123
Table 4-34	Control Block Base Address Register - 17h-14h (Primary IDE)	123
Table 4-35	Command Block Base Address Register - 1Bh-18h (Secondary IDE)	123
Table 4-36	Control Block Base Address Register - 1Fh-1Ch (Secondary IDE)	123
Table 4-37	Bus Master IDE Base Address Register - 23h-20h	123
Table 4-38	Interrupt Line Register - 3Ch	124
Table 4-39	Interrupt Pin Register - 3Dh	124
Table 4-40	IDE Initialization Control Register - 42h-40h	124
Table 4-41	IDE Enhanced Mode Register - 43h	125
Table 4-42	Internal ID Register - 1F2h	126
Table 4-43	Read Cycle Timing Register-A - 1F0h, Index-0	126
Table 4-44	Read Cycle Timing Register-B - 1F0h, Index-1	126
Table 4-45	Write Cycle Timing Register-A - 1F1h, Index-0	127
Table 4-46	Write Cycle Timing Register-B - 1F1h, Index-1	127
Table 4-47	Control Register - 1F3h	127
Table 4-48	Strap Register - 1F5h	128

List of Tables (cont.)

Table 4-49	Miscellaneous Register - 1F6h	128
Table 4-50	Internal ID Register - 172h	129
Table 4-51	Read Cycle Timing Register-A - 170h, Index-0	129
Table 4-52	Read Cycle Timing Register-B - 170h, Index-1	129
Table 4-53	Write Cycle Timing Register-A - 171h, Index-0	130
Table 4-54	Write Cycle Timing Register-B - 171h, Index-1	130
Table 4-55	Control Register - 173h	130
Table 4-56	Strap Register - 175h	131
Table 4-57	Miscellaneous Register - 176h	131
Table 4-58	Physical Region Descriptor Table Entry	132
Table 4-59	Bus Master IDE Registers	132
Table 4-60	Bus Master IDE Command Register for Primary IDE - Base Address + 00h	133
Table 4-61	Bus Master IDE Status Register for Primary IDE - Base Address + 02h	133
Table 4-62	Descriptor Table Pointer Register for Primary IDE - Base Address + 04h	134
Table 4-63	Bus Master IDE Command Register for Secondary IDE - Base Address + 08h	134
Table 4-64	Bus Master IDE Status Register for Secondary IDE - Base Address + 0Ah	135
Table 4-65	Descriptor Table Pointer Register for Secondary IDE - Base Address + 0Ch	135
Table 4-66	REGTIMx Programming Options	136
Table 4-67	16-Bit Timing Parameters with PCI Bus @ 33MHz	137
Table 4-68	16-Bit Timing Parameters with PCI Bus @ 25MHz	138
Table 4-69	IDE Interrupt Routing Chart	139
Table 5-1	Register Locations and Access Mechanisms Summary	142
Table 5-2	57VIN, 82C557M Vendor Identification Register - Address Offset 01h-00h	143
Table 5-3	57DID, 82C557M Device Identification Register - Address Offset 03h-02h	143
Table 5-4	57COM, 82C557M Command Register - Address Offset 05h-04h	143
Table 5-5	57STAT, 82C557M Status Register - Address Offset 07h-06h	144
Table 5-6	57REV, 82C557M Revision Identification Register - Address Offset 08h	144
Table 5-7	57CLASS, 82C557M Class Code Register - Address Offset 0Bh-09h	144
Table 5-8	57RESV1, 82C557M Reserved Register 1 - Address Offset 0Ch	144
Table 5-9	57MLTMR, 82C557M Master Latency Timer Register - Address Offset 0Dh	144
Table 5-10	57HEAD, 82C557M Header Type Register - Address Offset 0Eh	144
Table 5-11	57BIST, 82C557M Built-In Self-Test Register, Address Offset 0Fh	144
Table 5-12	57RESV2, 82C557M Reserved Register 2 - Address Offset 3Fh-10h	145

List of Tables (cont.)

Table 5-13 57MEMCTL, 82C557M Memory Control Register - Address Offset 43h-40h 145
Table 5-14 57DRCCTL, 82C557M DRAM Data Latching Control Register - Address Offset 47h-44h 145
Table 5-15 Index 00h 146
Table 5-16 57DRMCTL, 82C557M DRAM Control Register 1 - Index 01h 147
Table 5-17 57CHCTL1, 82C557M Cache Control Register 1 - Index 02h 147
Table 5-18 57CHCTL2, 82C557M Cache Control Register 2 - Index 03h 148
Table 5-19 57SHCTL1, 82C557M Shadow RAM Control Register 1 - Index 04h 148
Table 5-20 57SHCTL2, 82C557M Shadow RAM Control Register 2 - Index 05h 149
Table 5-21 57SHCTL3, 82C557M Shadow RAM Control Register 3 - Index 06h 149
Table 5-22 57TAGTST, 82C557M Tag Test Register - Index 07h 149
Table 5-23 57CPUCH, 82C557M CPU Cache Control Register - Index 08h 150
Table 5-24 57SYSMEM, 82C557M System Memory Function Register - Index 09h 151
Table 5-25 57DRAMA, 82C557M DRAM Hole A Address Decode Register 1 - Index 0Ah 151
Table 5-26 57DRAMB, 82C557M DRAM Hole B Address Decode Register 2 - Index 0Bh 151
Table 5-27 57EDMA, 82C557M Extended DMA Register - Index 0Ch 151
Table 5-28 57CLKCTL, 82C557M Clock Control Register - Index 0Dh 152
Table 5-29 57CYCTL1, 82C557M Cycle Control Register 1 - Index 0Eh 153
Table 5-30 57CYCTL2, 82C557M Cycle Control Register 2 - Index 0Fh 153
Table 5-31 57MISC1, 82C557M Miscellaneous Control Register 1 - Index 10h 154
Table 5-32 57MISC2, 82C557M Miscellaneous Control Register 2 - Index 11h 155
Table 5-33 57REFCTL - 82C557M Refresh Control Register - Index 12h 156
Table 5-34 57MEMDC1, 82C557M Memory Decode Control Register 1 - Index 13h 156
Table 5-35 57MEMDC2, 82C557M Memory Decode Control Register 2 - Index 14h 157
Table 5-36 DRAM Configurations 158
Table 5-37 57PCICY1, 82C557M PCI Cycle Control Register 1 - Index 15h 159
Table 5-38 57DRTY - 82C557M Dirty/Tag RAM Control Register - Index 16h 160
Table 5-39 57PCICY2, 82C557M PCI Cycle Control Register 2 - Index 17h 161
Table 5-40 57TRICTL, 82C557M Tristate Control Register - Index 18h 162
Table 5-41 57MEMDC3, 82C557M Memory Decode Control Register 3 - Index 19h 162
Table 5-42 57SGCTL1, 82C557M Memory Shadow Control Register 1 - Index 1Ah 163
Table 5-43 57SGCTL2, 82C557M Memory Shadow Control Register 2 - Index 1Bh 163
Table 5-44 57EDOCTL, 82C557M EDO DRAM Control Register - Index 1Ch 164
Table 5-45 57RESV3, 82C557M Reserved Register 3 - Index 1Dh 164
Table 5-46 57BOFF, 82C557M BOFF# Control Register - Index 1Eh 165

List of Tables (cont.)

Table 5-47	57RSVD1, 82C557M Reserved Register - Index 1Fh	165
Table 5-48	57REFPG, 82C557M Refresh Page Register (Write-Only) - Index 8Fh	166
Table 5-49	58VIN, 82C558M Vendor Identification Register - Address Offset 01h-00h	167
Table 5-50	58DID, 82C558M Device Identification Register - Address Offset 03h-02h	167
Table 5-51	58COM, 82C558M Command Register - Address Offset 05h-04h	167
Table 5-52	58STAT, 82C558M Status Register - Address Offset 07h-06h	168
Table 5-53	58REV, 82C558M Revision Identification Register - Address Offset 08h	168
Table 5-54	58CLASS, 82C558M Class Code Register - Address Offset 0Bh-09h	168
Table 5-55	58RESV1, 82C558M Reserved Register 1 - Address Offset 0Ch	168
Table 5-56	58MLTMR, 82C558M Master Latency Timer Register - Address Offset 0Dh	168
Table 5-57	58HEAD, 82C558M Header Type Register - Address Offset 0Eh	169
Table 5-58	58BIST, 82C558M Built-In Self-Test Register - Address Offset 0Fh	169
Table 5-59	58RESV2, 82C558M Reserved Register 2- Address Offset 3Fh-10h	169
Table 5-60	58KBDCTL, 82C558M Keyboard Control Register - Address Offset 41h-40h	170
Table 5-61	58MISC1, 82C558M Miscellaneous Control Register 1 - Address Offset 43h-42h	171
Table 5-62	58PINFU1, 82C558M Pin Functionality Register 1 - Address Offset 45h-44h	172
Table 5-63	58CYCTL1, 82C558M Cycle Control Register 1 - Address Offset 47h-46h	174
Table 5-64	58PINFU2, 82C558M Pin Functionality Register 2 - Address Offset 49h-48h	175
Table 5-65	58ROMCS, 82C558M ROMCS# Range Control Register - Address Offset 4Bh-4Ah	176
Table 5-66	58RESV3, 82C558M Reserved Register 3 - Address Offset 4Dh-4Ch	176
Table 5-67	58MISC2, 82C558M Miscellaneous Control Register 2 - Address Offset 4Fh-4Eh	177
Table 5-68	58TRIG, 82C558M Trigger Control Register - Address Offset 51h-50h	178
Table 5-69	58INTMCL, 82C558M Interrupt Multiplexing Control Register - Address Offset 53h-52h	179
Table 5-70	58DMACTL, 82C558M DMA Control Register - Address Offset 55h-54h	180
Table 5-71	Pin Multiplexing Chart 1	180
Table 5-72	Pin Multiplexing Chart 2	181
Table 5-73	Pin Multiplexing Chart 3	181
Table 5-74	Port B Register - I/O Port 61h	182
Table 5-75	PS/2 Reset Control - Port 92h	182
Table 5-76	DMA System I/O Register - Address 40Bh	183
Table 5-77	DMA System I/O Register - Address 483h-481h	183
Table 5-78	DMA System I/O Register - Address 48Bh-487h	183
Table 5-79	DMA System I/O Register - Address 4D6h	183
Table 5-80	Configuration Register (Index Port 022h, Data Port 023h) - Index: 01h	184

List of Tables (cont.)

Table 5-81	Command Register	185
Table 5-82	Mode Register	185
Table 5-83	Request Register Write Format	186
Table 5-84	Request Register Read Format	186
Table 5-85	Request Mask Register Set/Reset Format	186
Table 5-86	Request Mask Register Read/Write Format	186
Table 5-87	Status Register	186
Table 5-88	ICW1 Register - Address 020h (0A0h)	187
Table 5-89	ICW2 Register - Address 021h (0A1h)	187
Table 5-90	ICW3 Register - Format for INTC1 - Address 021h	187
Table 5-91	ICW3 Register - Format for INCT1 - Address 0A1h	187
Table 5-92	ICW4 Register - Address 021h (0A1h)	188
Table 5-93	OCW1 Register - Address 021h (0A1h)	188
Table 5-94	OCW2 Register - Address 020h (0A0h)	188
Table 5-95	OCW3 Register - Address 020h (0A0h)	189
Table 5-96	IIR Register - Address 020h (0A0h)	189
Table 5-97	ISR Register - Address 020h (0A0h)	189
Table 5-98	Poll Vector - Address 20h (0A0h)	189
Table 5-99	Interrupt Controller Subsystem Shadow Registers	190
Table 5-100	Control Word Format (Write Only) - Address 043h	191
Table 5-101	Counter Latch Command Format (Write Only) - Address 043h	191
Table 5-102	Read-Back Command Format (Write Only) - Address 043h	191
Table 5-103	Status Format - Address 043h	192
Table 5-104	Counter 0 Count Value - Address 40h	192
Table 5-105	Counter 1 Count Value - Address 41h	192
Table 5-106	Counter 2 Count Value - Address 42h	192
Table 5-107	Counter/Timer Subsystem Shadow Registers	193
Table 5-108	58GRNCTL, GREEN Mode Control/Enable Status - Index E0h	194
Table 5-109	58GETCTL, EPMI Control / GREEN Event Timer - Index E1h Register	195
Table 5-110	58COUNT, GREEN Event Timer Initial Count Register - Index E2h	196
Table 5-111	58IRQ1, IRQ Event Enable Register 1 - Index E3h	196
Table 5-112	58IRQ2, IRQ Event Enable Register 2 - Index E4h	196
Table 5-113	58DREQ, DREQ Event Enable Register - Index E5h	196
Table 5-114	58ACCDET, Device Cycle Monitor Enable Register - Index E6h	197

List of Tables (cont.)

Table 5-115	58MASK, Wake-up Source/Programmable I/O/Memory Address Mask Register - Index E7h	198
Table 5-116	58IOMEM1, Programmable IO/MEM Address Range Register - Index E8h	198
Table 5-117	58IOMEM2, Programmable IO/MEM Address Range Register - Index E9h	198
Table 5-118	58GPP, Enter GREEN State Port Register - Index EAh	199
Table 5-119	58NPP, Return to NORMAL State Configuration Port Register - Index EBh	199
Table 5-120	58PP, Shadow Register for External Power Control Latch Register - Index ECh	199
Table 5-121	58ACCMON, Device Cycle Detection Enable / Status Register - Index EDh	200
Table 5-122	58STPCLK, STPCLK# Modulation Register - Index EEh	202
Table 5-123	58MISC3, Miscellaneous Register - Index EFh	203
Table 5-124	58TMRCLK, Device Timer CLK Select / Enable Status Register - Index F0h	204
Table 5-125	58DV0TMR, Device 0 Timer Initial Count Register - Index F1h	205
Table 5-126	58DV1TMR, Device 1 Timer Initial Count Register - Index F2h	205
Table 5-127	58IOMSK, Device Timer IO/MEM Select, Mask Bits Register - Index F3h	205
Table 5-128	58DV0AD1, Device 0 IO/MEM Address Register - Index F4h	205
Table 5-129	58DV0AD2, Device 0 IO/MEM Address Register - Index F5h	206
Table 5-130	58DV1AD1, Device 1 IO/MEM Address Register - Index F6h	206
Table 5-131	58DV1AD2, Device 1 IO/MEM Address Register - Index F7h	206
Table 5-132	58RSVD1, 82C558M Reserved Register 1 - Index FBh-FAh	206
Table 5-133	58PMCTL1, Power Management Control Register 1 - Index FCh	207
Table 5-134	58PMCTL2, Power Management Control Register 2 - Index FDh	208
Table 5-135	58PMCTL3, Power Management Control Register 3 - Index FEh	209
Table 5-136	58GPCCTL, General Purpose Chip Select Control Register - Index FFh	210
Table 7-1	82C556M NAND Tree Test Mode Pins	220
Table 7-2	82C557M NAND Tree Test Mode Pins	221
Table 7-3	Drive High/Drive Low Test Mode Pins	222
Table 7-4	82C558M NAND Tree Test Mode Pins	223

1.0 Features

CPU Interface

- Fully supports Intel's 3.3V Pentium™ Processor and dual processor configuration at 50, 60, and 66.667MHz
- Supports P55C™, K5™, and M1™ processors
- Supports the Cyrix® M1 Processor linear burst mode
- Three chip solution:
  - 82C556M DBC (Data Buffer Controller) in a 160-pin PQFP (Plastic Quad Flat Pack)
  - 82C557M System Controller (SYSC) in a 208-pin PQFP or TQFP (Thin Quad Flat Pack)
  - 82C558M IPC (Integrated Peripherals Controller) in a 208-pin PQFP or TQFP
- Supports CPU address pipelining

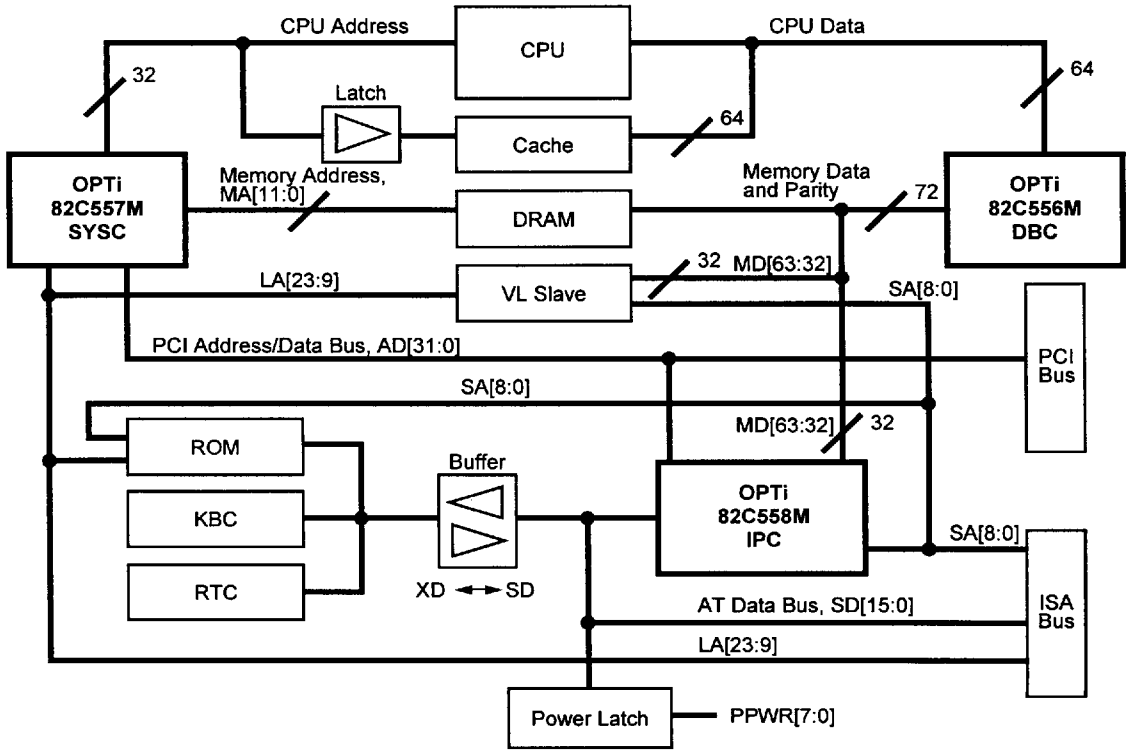
Cache Interface

- Write-back/write-through, direct-mapped cache with size selections: 64KB, 128KB, 256KB, 1MB and 2MB
- Support for synchronous and asynchronous SRAMs, pipelined synchronous SRAMs, and Intel standard BSRAMs (BiCMOS SRAMs)
- Support for the Sony SONIC-2WP™ Cache Module
- Programmable cache write policy
  - Write-back
  - Write-through
  - Adaptive write-back
- Built-in tag auto-invalidation circuitry
- Fully programmable 3-2-2-2 cache burst read/write cycles. 2-1-1-1 burst read/write support at 50MHz
- Options for cacheable, write protected, system and video BIOS

DRAM Interface

- Supports six banks of 64-bit wide DRAMs with 256KB, 512KB, 1MB, 2MB, 4MB, 8MB and 16MB addressing page mode DRAMs
- Supports DRAM configurations up to 512MB
- EDO DRAM support provides both OE# and WE# control
- Supports mixing of EDO SIMMs and fast page mode DRAM SIMMs among different banks

Figure 1-1 System Block Diagram



Features (cont.)

- Supports 2-2-2-2 pipelined DRAM burst cycles when using EDO DRAMs
- Programmable drive currents for the DRAM control signals
- 64-bit DRAM post write buffer
- Hidden refresh with CAS-before-RAS refresh supported
- Self-refresh supported during Suspend mode
- Support for two programmable non-cacheable memory regions

PCI Interface

- Interfaces the CPU and standard buses to both Peripheral Component Interconnect (PCI) and VESA local bus operating in synchronous/asynchronous modes, with VESA bus always operating at PCI bus operating frequency
- Supports four PCI masters, one VESA slave, and six ISA slots
- Supports PCI pre-snoop for PCI masters
- PCI byte/word merge support for CPU accesses to PCI bus, and support for PCI pre-fetch
- Burst mode PCI accesses to local memory supported

IDE Interface

- Integrated master mode IDE
- Two channels supported (up to four devices)
- PIO mode transfer support (up to Mode 5)
- Enhanced ATA Specification support
- Single- and/or Multi-Word DMA Mode 2 timing
- Scatter/Gather feature
- Built-in FIFOs
- Data prefetch and post write supported
- Compliant to the PCI Bus Mastering Guideline

DMA Interface

- Type F DMA support

Power Management

- "True" GREEN power management support, with support for STPCLK# modulation and the CPU stop clock state

Miscellaneous

- Support for flash ROM
- Shadow RAM option
- Transparent 8042 emulation for fast CPU reset and Gate A20 generation
- Supports Port 92h, fast Gate A20 and fast reset
- Includes a fully integrated 82C206 with external real-time clock (RTC) interface

2.0 Overview

OPTi's Viper-M (82C556M/82C557M/82C558M) Chipset provides a highly integrated solution for fully compatible, high performance PC/AT platforms based on Intel's 3.3V Pentium Processor, Cryix's M1 Processor, and AMD's K5 Processor. The chipset provides 64-bit core logic, integrated PCI and VESA support, and sophisticated power management features. This highly integrated approach supplies the foundation for a cost effective platform without compromising performance. Its feature set furnishes an array of control and status monitoring options that are accessed through a simple and straightforward interface. All major BIOS vendors provide extensive software hooks that allow system designers to integrate their own special features with minimal effort.

The Viper-M Chipset is comprised of three chips:

- 82C556M Data Buffer Controller (DBC)
- 82C557M System Controller (SYSC)
- 82C558M Integrated Peripherals Controller (IPC)

2.1 82C556M (DBC) Data Buffer Controller

The 82C556M DBC performs the task of buffering the CPU to the DRAM memory data path. It also performs parity checking.

- CPU to memory data buffer
- CPU to local bus buffer
- Memory to local bus buffer
- 160-pin PQFP

Figure 2-1 shows a block diagram of the 82C556M DBC.

2.2 82C557M (SYSC) System Controller

The 82C557M SYSC provides the control functions for the host CPU interface, the 64-bit Level-2 (L2) cache, the 64-bit DRAM bus, the VL bus interface, and the PCI interface. (The 82C557M also controls the data flow between the CPU bus, the DRAM bus, the local buses, and the 8/16-bit ISA bus.) The 82C557M interprets and translates cycles from the CPU, PCI bus master, ISA master, and DMA to the host memory, local bus slave, PCI bus slave, or ISA bus devices.

- 3.3V CPU interface
- DRAM controller
- L2 cache controller
- L1 cache controller
- PCI interface
- Arbitration logic
- Data bus buffer control (memory data bus to and from host data bus)
- VL bus interface
- 208-pin PQFP or TQFP

Figure 2-2 shows a block diagram of the 82C557M SYSC.

2.3 82C558M (IPC) Integrated Peripherals Controller

The 82C558M IPC contains the AT bus controller and includes an 82C206, RTC interface, DMA controller, and a sophisticated system power management unit.

- AT bus controller
- Master mode IDE
- Type F DMA support
- Integrated 82C206 IPC
- System power management functions
- PCI local bus interface
- PCI to ISA expansion bridge
- Keyboard emulation of A20M# and CPU warm reset
- Port B and Port 92h Register
- Local bus interface
- 208-pin PQFP or TQFP

Figure 2-3 shows a block diagram of the 82C558M IPC.

Figure 2-1 82C556M Block Diagram

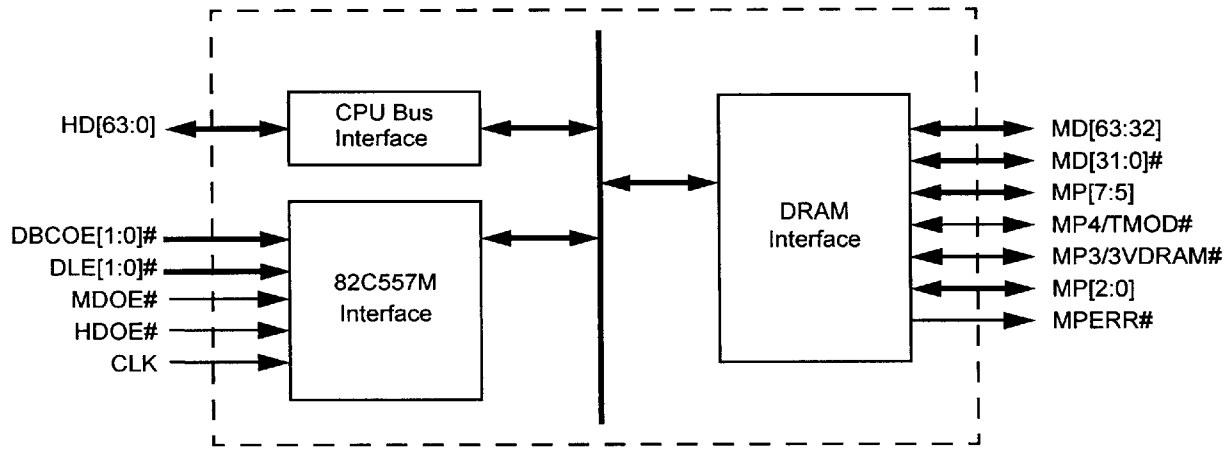


Figure 2-2 82C557M Block Diagram

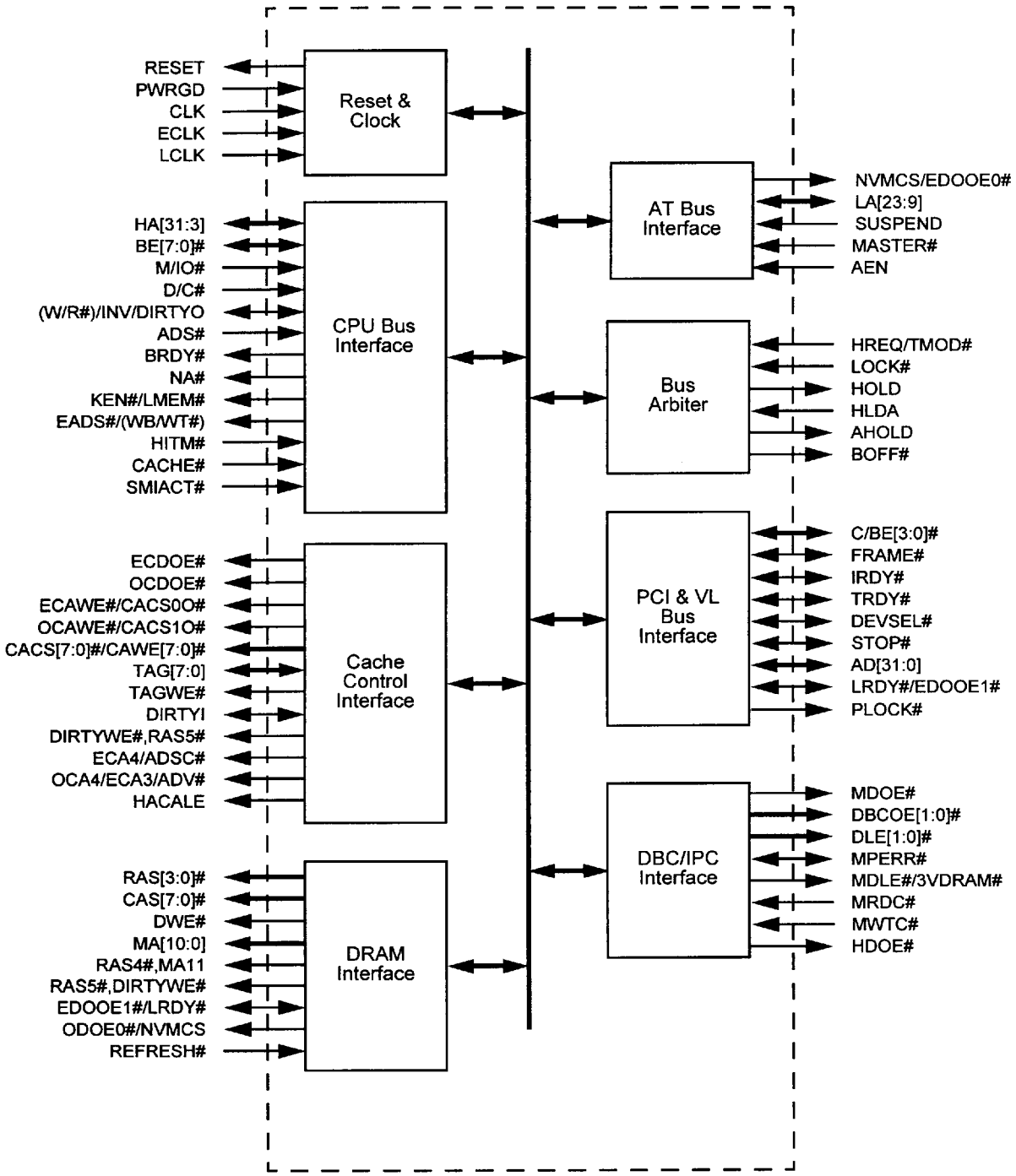
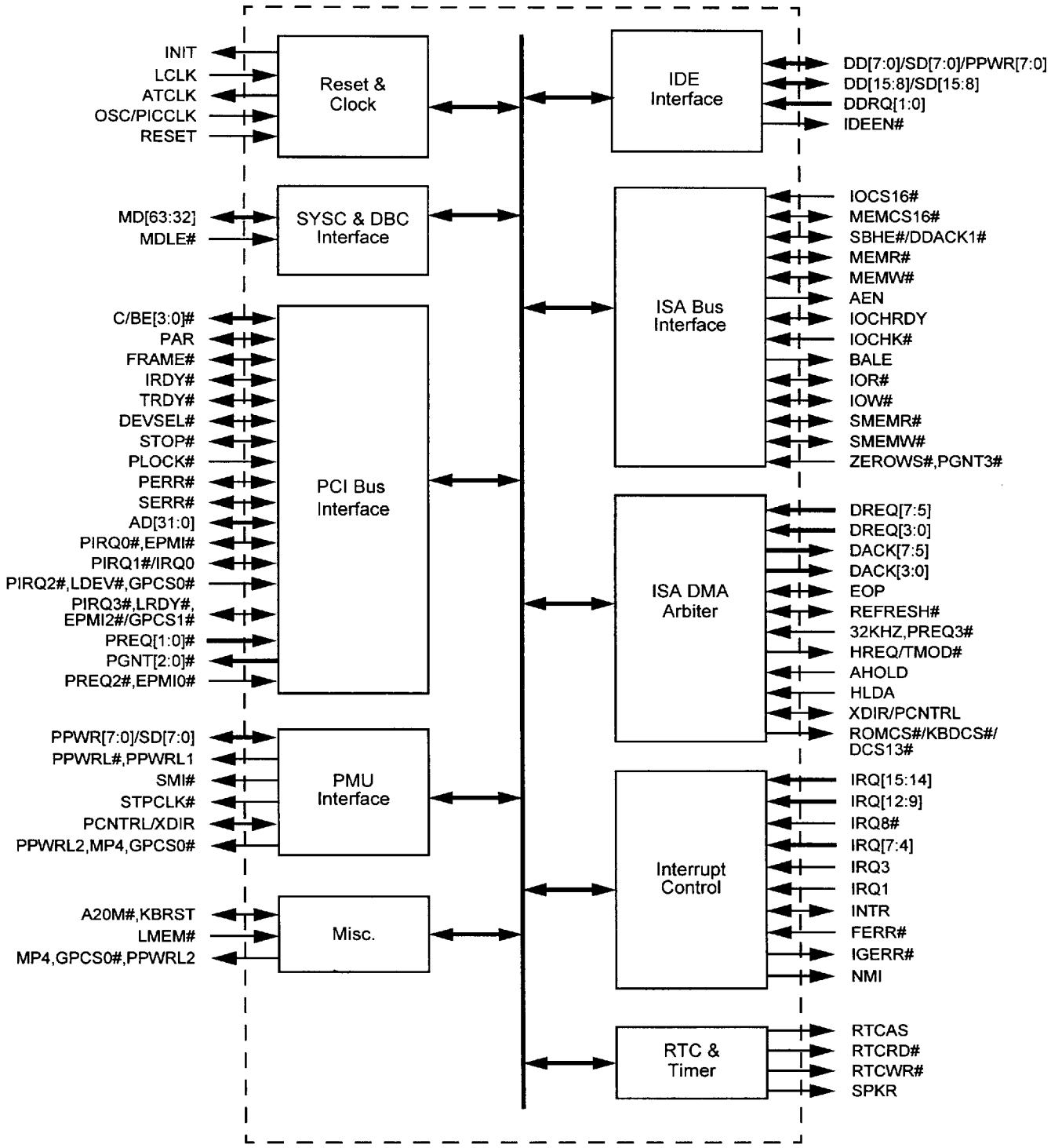


Figure 2-3 82C558M Block Diagram



3.0 Signal Definitions

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 either "group-wise" or "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.

Figure 3-1 82C556M Pin Diagram

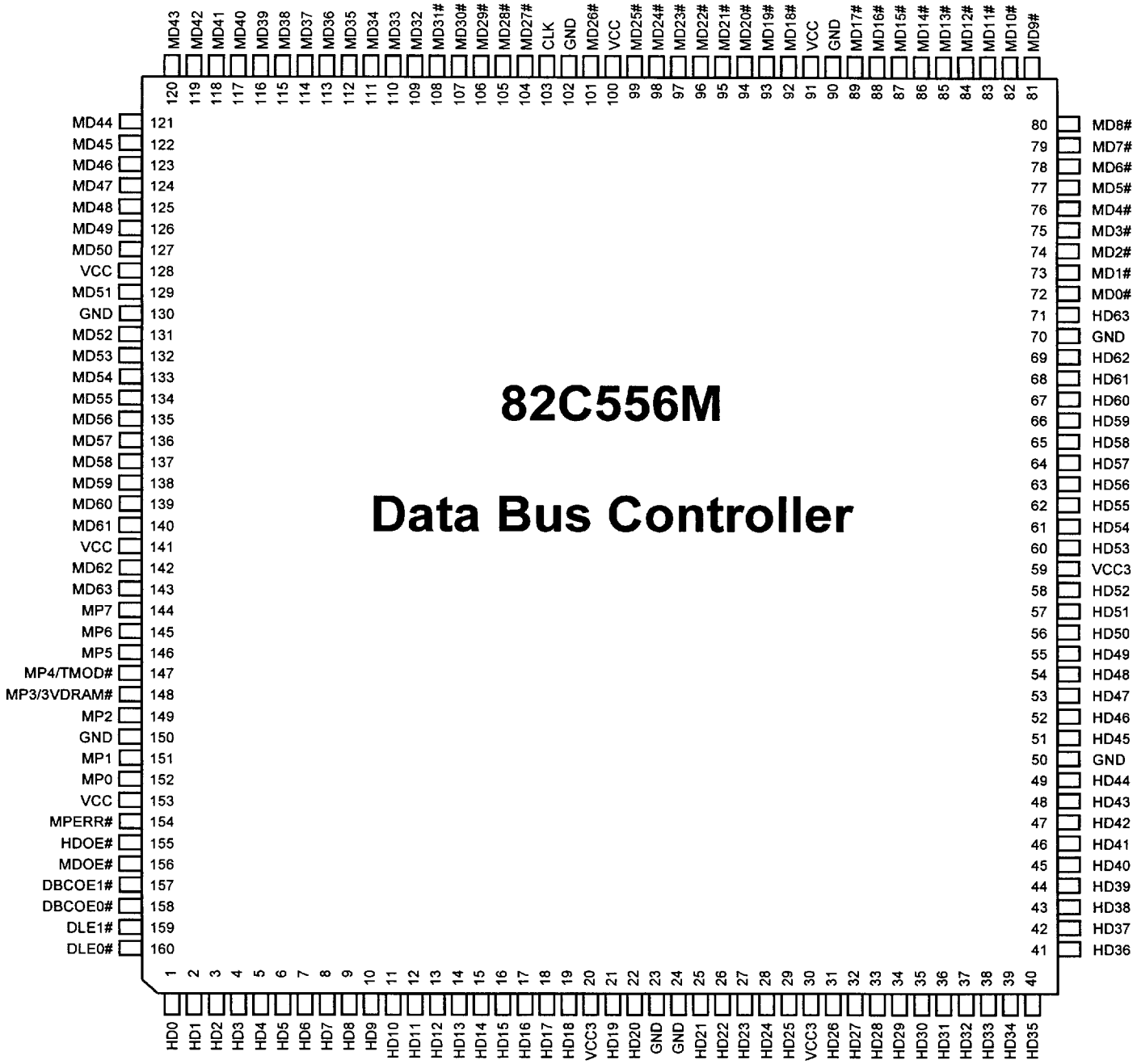


Table 3-1 82C556M 160-Pin Numerical Pin Cross-Reference List

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
1	HD0	I/O	TTL	4	Int PD
2	HD1	I/O	TTL	4	Int PD
3	HD2	I/O	TTL	4	Int PD
4	HD3	I/O	TTL	4	Int PD
5	HD4	I/O	TTL	4	Int PD
6	HD5	I/O	TTL	4	Int PD
7	HD6	I/O	TTL	4	Int PD
8	HD7	I/O	TTL	4	Int PD
9	HD8	I/O	TTL	4	Int PD
10	HD9	I/O	TTL	4	Int PD
11	HD10	I/O	TTL	4	Int PD
12	HD11	I/O	TTL	4	Int PD
13	HD12	I/O	TTL	4	Int PD
14	HD13	I/O	TTL	4	Int PD
15	HD14	I/O	TTL	4	Int PD
16	HD15	I/O	TTL	4	Int PD
17	HD16	I/O	TTL	4	Int PD
18	HD17	I/O	TTL	4	Int PD
19	HD18	I/O	TTL	4	Int PD
20	VCC3	I	P		
21	HD19	I/O	TTL	4	Int PD
22	HD20	I/O	TTL	4	Int PD
23	GND	I	G		
24	GND	I	G		
25	HD21	I/O	TTL	4	Int PD
26	HD22	I/O	TTL	4	Int PD
27	HD23	I/O	TTL	4	Int PD
28	HD24	I/O	TTL	4	Int PD
29	HD25	I/O	TTL	4	Int PD
30	VCC3	I	P		
31	HD26	I/O	TTL	4	Int PD
32	HD27	I/O	TTL	4	Int PD
33	HD28	I/O	TTL	4	Int PD
34	HD29	I/O	TTL	4	Int PD
35	HD30	I/O	TTL	4	Int PD
36	HD31	I/O	TTL	4	Int PD
37	HD32	I/O	TTL	4	Int PD
38	HD33	I/O	TTL	4	Int PD
39	HD34	I/O	TTL	4	Int PD
40	HD35	I/O	TTL	4	Int PD
41	HD36	I/O	TTL	4	Int PD
42	HD37	I/O	TTL	4	Int PD
43	HD38	I/O	TTL	4	Int PD
44	HD39	I/O	TTL	4	Int PD
45	HD40	I/O	TTL	4	Int PD
46	HD41	I/O	TTL	4	Int PD
47	HD42	I/O	TTL	4	Int PD
48	HD43	I/O	TTL	4	Int PD
49	HD44	I/O	TTL	4	Int PD
50	GND	I	G		
51	HD45	I/O	TTL	4	Int PD
52	HD46	I/O	TTL	4	Int PD
53	HD47	I/O	TTL	4	Int PD
54	HD48	I/O	TTL	4	Int PD
55	HD49	I/O	TTL	4	Int PD
56	HD50	I/O	TTL	4	Int PD
57	HD51	I/O	TTL	4	Int PD
58	HD52	I/O	TTL	4	Int PD
59	VCC3	I	P		
60	HD53	I/O	TTL	4	Int PD
61	HD54	I/O	TTL	4	Int PD
62	HD55	I/O	TTL	4	Int PD
63	HD56	I/O	TTL	4	Int PD
64	HD57	I/O	TTL	4	Int PD
65	HD58	I/O	TTL	4	Int PD
66	HD59	I/O	TTL	4	Int PD
67	HD60	I/O	TTL	4	Int PD
68	HD61	I/O	TTL	4	Int PD
69	HD62	I/O	TTL	4	Int PD
70	GND	I	P		
71	HD63	I/O	TTL	4	
72	MD0#	I/O	CMOS	4	Int PU
73	MD1#	I/O	CMOS	4	Int PU
74	MD2#	I/O	CMOS	4	Int PU
75	MD3#	I/O	CMOS	4	Int PU
76	MD4#	I/O	CMOS	4	Int PU
77	MD5#	I/O	CMOS	4	Int PU
78	MD6#	I/O	CMOS	4	Int PU
79	MD7#	I/O	CMOS	4	Int PU
80	MD8#	I/O	CMOS	4	Int PU
81	MD9#	I/O	CMOS	4	Int PU
82	MD10#	I/O	CMOS	4	Int PU
83	MD11#	I/O	CMOS	4	Int PU
84	MD12#	I/O	CMOS	4	Int PU
85	MD13#	I/O	CMOS	4	Int PU
86	MD14#	I/O	CMOS	4	Int PU
87	MD15#	I/O	CMOS	4	Int PU
88	MD16#	I/O	CMOS	4	Int PU

82C556M 160-Pin Numerical Pin Cross-Reference List (cont.)

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
89	MD17#	I/O	CMOS	4	Int PU
90	GND	I	G		
91	VCC	I	P		
92	MD18#	I/O	CMOS	4	Int PU
93	MD19#	I/O	CMOS	4	Int PU
94	MD20#	I/O	CMOS	4	Int PU
95	MD21#	I/O	CMOS	4	Int PU
96	MD22#	I/O	CMOS	4	Int PU
97	MD23#	I/O	CMOS	4	Int PU
98	MD24#	I/O	CMOS	4	Int PU
99	MD25#	I/O	CMOS	4	Int PU
100	VCC	I	P		
101	MD26#	I/O	CMOS	4	Int PU
102	GND	I	G		
103	CLK	I	TTL		
104	MD27#	I/O	CMOS	4	Int PU
105	MD28#	I/O	CMOS	4	Int PU
106	MD29#	I/O	CMOS	4	Int PU
107	MD30#	I/O	CMOS	4	Int PU
108	MD31#	I/O	CMOS	4	Int PU
109	MD32	I/O	CMOS	4	Int PU
110	MD33	I/O	CMOS	4	Int PU
111	MD34	I/O	CMOS	4	Int PU
112	MD35	I/O	CMOS	4	Int PU
113	MD36	I/O	CMOS	4	Int PU
114	MD37	I/O	CMOS	4	Int PU
115	MD38	I/O	CMOS	4	Int PU
116	MD39	I/O	CMOS	4	Int PU
117	MD40	I/O	CMOS	4	Int PU
118	MD41	I/O	CMOS	4	Int PU
119	MD42	I/O	CMOS	4	Int PU
120	MD43	I/O	CMOS	4	Int PU
121	MD44	I/O	CMOS	4	Int PU
122	MD45	I/O	CMOS	4	Int PU
123	MD46	I/O	CMOS	4	Int PU
124	MD47	I/O	CMOS	4	Int PU
125	MD48	I/O	CMOS	4	Int PU
126	MD49	I/O	CMOS	4	Int PU
127	MD50	I/O	CMOS	4	Int PU
128	VCC	I	P		
129	MD51	I/O	CMOS	4	Int PU
130	GND	I	G		
131	MD52	I/O	CMOS	4	Int PU
132	MD53	I/O	CMOS	4	Int PU
133	MD54	I/O	CMOS	4	Int PU
134	MD55	I/O	CMOS	4	Int PU
135	MD56	I/O	CMOS	4	Int PU
136	MD57	I/O	CMOS	4	Int PU
137	MD58	I/O	CMOS	4	Int PU
138	MD59	I/O	CMOS	4	Int PU
139	MD60	I/O	CMOS	4	Int PU
140	MD61	I/O	CMOS	4	Int PU
141	VCC	I	P		
142	MD62	I/O	CMOS	4	
143	MD63	I/O	CMOS	4	
144	MP7	I/O	CMOS	4	Int PU
145	MP6	I/O	CMOS	4	Int PU
146	MP5	I/O	CMOS	4	Int PU
147	MP4/TMOD#	I/O	CMOS	4	Int PU
148	MP3/3VDRAM#	I/O	CMOS	4	Int PU
149	MP2	I/O	CMOS	4	Int PU
150	GND	I	G		
151	MP1	I/O	CMOS	4	Int PU
152	MP0	I/O	CMOS	4	Int PU
153	VCC	I	P		
154	MPERR#	O		4	
155	HDOE#	I	TTL		
156	MDOE#	I	TTL		
157	DBCOE1#	I	TTL		
158	DBCOE0#	I	TTL		
159	DLE1#	I	TTL		
160	DLE0#	I	TTL		

Legend:

CMOS	CMOS-level compatible	P	Power
Ext	External	PD	Pull-down resistor
G	Ground	PU	Pull-up resistor
Int	Internal	Sch	Schmitt-trigger
I/O	Input/Output	TTL	TTL-level compatible
OD	Open drain		

Table 3-2 82C556M 160-Pin Alphabetical Cross-Reference List

Pin No.	Pin Name
103	CLK
158	DBCOE0#
157	DBCOE1#
160	DLE0#
159	DLE1#
23	GND
24	GND
50	GND
70	GND
90	GND
102	GND
130	GND
150	GND
1	HD0
2	HD1
3	HD2
4	HD3
5	HD4
6	HD5
7	HD6
8	HD7
9	HD8
10	HD9
11	HD10
12	HD11
13	HD12
14	HD13
15	HD14
16	HD15
17	HD16
18	HD17
19	HD18
21	HD19
22	HD20
25	HD21
26	HD22
27	HD23
28	HD24
29	HD25
31	HD26

Pin No.	Pin Name
32	HD27
33	HD28
34	HD29
35	HD30
36	HD31
37	HD32
38	HD33
39	HD34
40	HD35
41	HD36
42	HD37
43	HD38
44	HD39
45	HD40
46	HD41
47	HD42
48	HD43
49	HD44
51	HD45
52	HD46
53	HD47
54	HD48
55	HD49
56	HD50
57	HD51
58	HD52
60	HD53
61	HD54
62	HD55
63	HD56
64	HD57
65	HD58
66	HD59
67	HD60
68	HD61
69	HD62
71	HD63
155	HDOE#
72	MD0#
73	MD1#

Pin No.	Pin Name
74	MD2#
75	MD3#
76	MD4#
77	MD5#
78	MD6#
79	MD7#
80	MD8#
81	MD9#
82	MD10#
83	MD11#
84	MD12#
85	MD13#
86	MD14#
87	MD15#
88	MD16#
89	MD17#
92	MD18#
93	MD19#
94	MD20#
95	MD21#
96	MD22#
97	MD23#
98	MD24#
99	MD25#
101	MD26#
104	MD27#
105	MD28#
106	MD29#
107	MD30#
108	MD31#
109	MD32
110	MD33
111	MD34
112	MD35
113	MD36
114	MD37
115	MD38
116	MD39
117	MD40
118	MD41

Pin No.	Pin Name
119	MD42
120	MD43
121	MD44
122	MD45
123	MD46
124	MD47
125	MD48
126	MD49
127	MD50
129	MD51
131	MD52
132	MD53
133	MD54
134	MD55
135	MD56
136	MD57
137	MD58
138	MD59
139	MD60
140	MD61
142	MD62
143	MD63
156	MDOE#
152	MP0
151	MP1
149	MP2
148	MP3/3VDRAM#
147	MP4/TMOD#
146	MP5
145	MP6
144	MP7
154	MPERR#
153	VCC
91	VCC
100	VCC
128	VCC
141	VCC
20	VCC3
30	VCC3
59	VCC3

3.2 82C556M Signal Descriptions

3.2.1 CPU 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 82C557M Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
DBCOE[1:0]#	157, 158	I	**DBC Output Enables:** These lines are connected to the 82C557M's DBCOE[1:0]# pins. These signals, along with the MDOE# and the HDOE# signals, form the encoded command sent from the 82C557M to the 82C556M. These commands indicate the type of cycle currently underway and it enables the 82C556M to perform the appropriate data steering, latching and direction control. The encoded commands are defined in Table 3-5.
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 82C557M. 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 82C557M. 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 82C557M's DLE[1:0]# pins and used to latch the HD and MD data bus depending on which cycle is occurring.

82C556M 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 82C558M. 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 DS[31:0] bus or the MD[63:32] bus. These lines have internal pull-up resistors.
MP[7:5], 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 3-3 details the strap functionality for these pins. MP[7:0] have internal pull-up resistors.
MPERR#	154	O	**Memory Parity Error Indication:** This pin is connected to the MPERR# input of the 82C557M. It indicates the detection of a parity error during a read from DRAM and is qualified within the 82C557M when parity is enabled.
CLK	103	I	**Clock:** This clocking input is used for EDO (extended data out) DRAM support.

3.2.4 Power and Ground Pins

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

Table 3-3 82C556M Strap Pin Mapping Functions

Signals	High	Low
MP0	Enable MP[7:4] as outputs from the 82C556M during PCI master writes	Disable MP[7:4] as outputs from the 82C556M 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 3-4 82C556M Operating Voltage Groups

TTL (5.0V)	DRAM (5.0V)	CPU/Cache (3.3V)
CLK	MD[63:32]	HD[63:0]
DBCOE[1:0]#	MD[31:0]#	
DLE[1:0]#	MP[7:5]	
HDOE#	MP4/TMOD#	
MDOE#	MP3/3VDRAM#	
MPERR#	MP[2:0]	

Table 3-5 82C556M 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 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, performs the required data bus steering, buffering and latching control.
10	0	1	This code indicates that the following cycle is underway: - PCI read low order data from DRAM The 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, 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 82C556M then, along with the appropriate control signals from the 82C557M, performs the required data bus steering, buffering and latching control
01	1	0	This code indicates that the system is in the reset state.

Figure 3-2 82C557M Pin Diagram

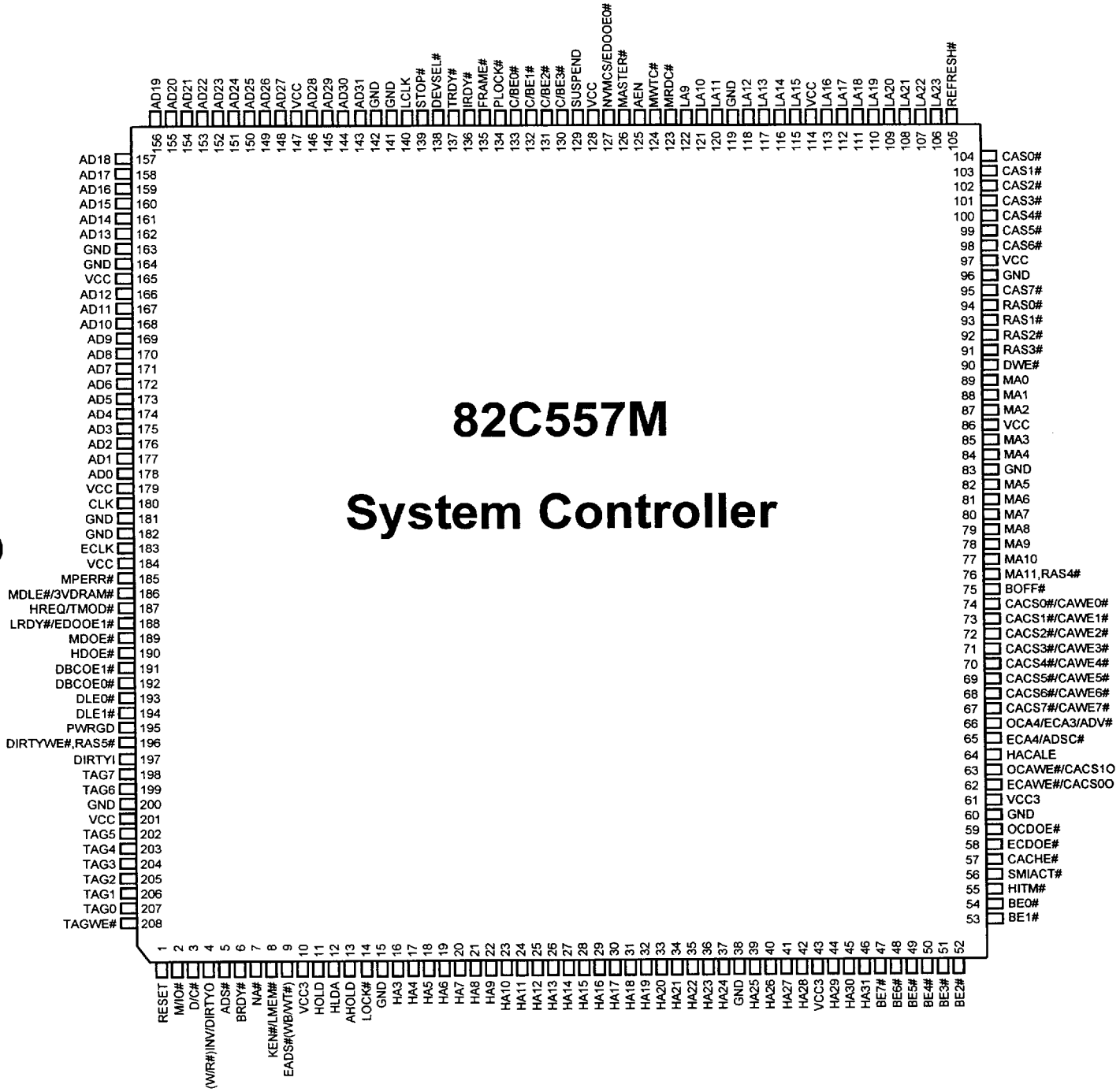


Table 3-6 82C557M Numerical Pin Cross-Reference List

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
1	RESET	O		8	
2	M/IO#	I	TTL		
3	D/C#	I	TTL		
4	(W/R#)/INV/DIRTYO	I/O	TTL	4	
5	ADS#	I	TTL	8	
6	BRDY#	O		8	
7	NA#	O		8	
8	KEN#/LMEM#	O		8	
9	EADS#/(WB/WT#)	O		8	
10	VCC3	I	P		
11	HOLD	O		8	
12	HLDA	I	TTL		
13	AHOLD	O		8	
14	LOCK#	I	TTL		Int PD, activated when HLDA is active
15	GND	I	G		
16	HA3	I/O	TTL	4	Int PD, however, ext PU should be used
17	HA4	I/O	TTL	4	Int PD, however, ext PU should be used
18	HA5	I/O	TTL	4	Int PD
19	HA6	I/O	TTL	4	Int PD
20	HA7	I/O	TTL	4	Int PD
21	HA8	I/O	TTL	4	Int PD
22	HA9	I/O	TTL	4	Int PD
23	HA10	I/O	TTL	4	Int PD
24	HA11	I/O	TTL	4	Int PD
25	HA12	I/O	TTL	4	Int PD
26	HA13	I/O	TTL	4	Int PD
27	HA14	I/O	TTL	4	Int PD
28	HA15	I/O	TTL	4	Int PD
29	HA16	I/O	TTL	4	Int PD
30	HA17	I/O	TTL	4	Int PD
31	HA18	I/O	TTL	4	Int PD
32	HA19	I/O	TTL	4	Int PD
33	HA20	I/O	TTL	4	Int PD
34	HA21	I/O	TTL	4	Int PD
35	HA22	I/O	TTL	4	Int PD
36	HA23	I/O	TTL	4	Int PD
37	HA24	I/O	TTL	4	Int PD
38	GND	I	G		
39	HA25	I/O	TTL	4	Int PD
40	HA26	I/O	TTL	4	Int PD
41	HA27	I/O	TTL	4	Int PD
42	HA28	I/O	TTL	4	Int PD
43	VCC3	I	P		
44	HA29	I/O	TTL	4	Int PD
45	HA30	I/O	TTL	4	Int PD
46	HA31	I/O	TTL	4	Int PD
47	BE7#	I	TTL		Int PD, activated when HLDA is active
48	BE6#	I	TTL		Int PD, activated when HLDA is active
49	BE5#	I	TTL		Int PD, activated when HLDA is active
50	BE4#	I	TTL		Int PD, activated when HLDA is active
51	BE3#	I	TTL		Int PD, activated when HLDA is active
52	BE2#	I	TTL		Int PD, activated when HLDA is active
53	BE1#	I	TTL		Int PD, activated when HLDA is active
54	BE0#	I	TTL		Int PD, activated when HLDA is active
55	HITM#	I	TTL		
56	SMIACT#	I	TTL		
57	CACHE#	I	TTL		
58	ECDOE#	O		8	
59	OCDOE#	O		8	
60	GND	I	G		
61	VCC3	I	P		
62	ECAWE#/CACS0O#	O		8	
63	OCAWE#/CACS1O#	O		8	
64	HACALE	O		8	
65	ECA4/ADSC#	O		8	
66	OCA4/ECA3/ADV#	O		8	
67	CACS7#/CAWE7#	O		4	
68	CACS6#/CAWE6#	O		4	
69	CACS5#/CAWE5#	O		4	
70	CACS4#/CAWE4#	O		4	
71	CACS3#/CAWE3#	O		4	
72	CACS2#/CAWE2#	O		4	
73	CACS1#/CAWE1#	O		4	
74	CACS0#/CAWE0#	O		4	
75	BOFF#	O			
76	MA11,RAS4#	O		4*	
77	MA10	O		4*	
78	MA9	O		4*	
79	MA8	O		4*	
80	MA7	O		4*	
81	MA6	O		4*	
82	MA5	O		4*	
83	GND	I	G		
84	MA4	O		4*	
85	MA3	O		4*	
86	VCC	I	P		
87	MA2	O		4*	
88	MA1	O		4*	
89	MA0	O		4*	
90	DWE#	O		16	
91	RAS3#	O		4*	
92	RAS2#	O		4*	
93	RAS1#	O		4*	
94	RAS0#	O		4*	
95	CAS7#	O		8	
96	GND	I	G		
97	VCC	I	P		
98	CAS6#	O		8	
99	CAS5#	O		8	
100	CAS4#	O		8	
101	CAS3#	O		8	
102	CAS2#	O		8	
103	CAS1#	O		8	

82C557M Numerical Pin Cross-Reference List (cont.)

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
104	CAS0#	O		8	
105	REFRESH#	I	CMOS		
106	LA23	I/O	CMOS	8	
107	LA22	I/O	CMOS	8	
108	LA21	I/O	CMOS	8	
109	LA20	I/O	CMOS	8	
110	LA19	I/O	CMOS	8	
111	LA18	I/O	CMOS	8	
112	LA17	I/O	CMOS	8	
113	LA16	I/O	CMOS	8	
114	VCC	I	P		
115	LA15	I/O	CMOS	8	
116	LA14	I/O	CMOS	8	
117	LA13	I/O	CMOS	8	
118	LA12	I/O	CMOS	8	
119	GND	I	G		
120	LA11	I/O	CMOS	8	
121	LA10	I/O	CMOS	8	
122	LA9	I/O	CMOS	8	
123	MRDC#	I	CMOS		
124	MWTC#	I	CMOS		
125	AEN	I	CMOS		Ext PU
126	MASTER#	I	CMOS		
127	NVMCS/EDOOE0#	O		4	
128	VCC	I	P		
129	SUSPEND	I	TTL		
130	C/BE3#	I/O	TTL	PCI	
131	C/BE2#	I/O	TTL	PCI	
132	C/BE1#	I/O	TTL	PCI	
133	C/BE0#	I/O	TTL	PCI	
134	PLOCK#	O	TTL	PCI	
135	FRAME#	I/O	TTL	PCI	
136	IRDY#	I/O	TTL	PCI	
137	TRDY#	I/O	TTL	PCI	
138	DEVSEL#	I/O	TTL	PCI	
139	STOP#	I/O	TTL	PCI	
140	LCLK	I	TTL		
141	GND	I	G		
142	GND	I	G		
143	AD31	I/O	TTL	PCI	
144	AD30	I/O	TTL	PCI	
145	AD29	I/O	TTL	PCI	
146	AD28	I/O	TTL	PCI	
147	VCC	I	P		
148	AD27	I/O	TTL	PCI	
149	AD26	I/O	TTL	PCI	
150	AD25	I/O	TTL	PCI	
151	AD24	I/O	TTL	PCI	
152	AD23	I/O	TTL	PCI	
153	AD22	I/O	TTL	PCI	
154	AD21	I/O	TTL	PCI	
155	AD20	I/O	TTL	PCI	
156	AD19	I/O	TTL	PCI	
157	AD18	I/O	TTL	PCI	
158	AD17	I/O	TTL	PCI	
159	AD16	I/O	TTL	PCI	
160	AD15	I/O	TTL	PCI	
161	AD14	I/O	TTL	PCI	
162	AD13	I/O	TTL	PCI	
163	GND	I	G		
164	GND	I	G		
165	VCC	I	P		
166	AD12	I/O	TTL	PCI	
167	AD11	I/O	TTL	PCI	
168	AD10	I/O	TTL	PCI	
169	AD9	I/O	TTL	PCI	
170	AD8	I/O	TTL	PCI	
171	AD7	I/O	TTL	PCI	
172	AD6	I/O	TTL	PCI	
173	AD5	I/O	TTL	PCI	
174	AD4	I/O	TTL	PCI	
175	AD3	I/O	TTL	PCI	
176	AD2	I/O	TTL	PCI	
177	AD1	I/O	TTL	PCI	
178	AD0	I/O	TTL	PCI	
179	VCC	I	P		
180	CLK	I	TTL		
181	GND	I	G		
182	GND	I	G		
183	ECLK	I	TTL		
184	VCC	I	P		
185	MPERR#	I/O	TTL	4	
186	MDLE#/3VDRAM#	I/O	TTL	4	Ext PU
187	HREQ/TMOD#	I	TTL		Ext PU
188	LRDY#/EDOOE1#	I/O	TTL	4	Ext PU
189	MDOE#	O		4	
190	HDOE#			4	
191	DBCOE1#	O		4	
192	DBCOE0#	O		4	
193	DLE0#	O		4	
194	DLE1#	O		4	
195	PWRGD	I	Sch		
196	DIRTYWE#,RAS5#	O		8	
197	DIRTYI	I/O			
198	TAG7	I/O			
199	TAG6	I/O			
200	GND	I	G		
201	VCC	I	P		
202	TAG5	I/O	TTL	4	
203	TAG4	I/O	TTL	4	
204	TAG3	I/O	TTL	4	
205	TAG2	I/O	TTL	4	
206	TAG1	I/O	TTL	4	
207	TAG0	I/O	TTL	4	
208	TAGWE#	O		8	

*Default drive is 4ma, however, by setting Index 18h[4] = 1, it can be increased to 16mA.

Legend:

CMOS	CMOS-level compatible	P	Power
Ext	External	PD	Pull-down resistor
G	Ground	PU	Pull-up resistor
Int	Internal	Sch	Schmitt-trigger
I/O	Input/Output	TTL	TTL-level compatible
OD	Open drain		

Table 3-7 82C557M Alphabetical Pin Cross-Reference List

Pin Name	Pin No.
AD0	178
AD1	177
AD2	176
AD3	175
AD4	174
AD5	173
AD6	172
AD7	171
AD8	170
AD9	169
AD10	168
AD11	167
AD12	166
AD13	162
AD14	161
AD15	160
AD16	159
AD17	158
AD18	157
AD19	156
AD20	155
AD21	154
AD22	153
AD23	152
AD24	151
AD25	150
AD26	149
AD27	148
AD28	146
AD29	145
AD30	144
AD31	143
ADS#	5
AEN	125
AHOLD	13
BE0#	54
BE1#	53
BE2#	52
BE3#	51
BE4#	50
BE5#	49
BE6#	48
BE7#	47
BOFF#	75
BRDY#	6
C/BE0#	133
C/BE1#	132
C/BE2#	131
C/BE3#	130
CACHE#	57
CACS0#/CAWE0#	74
CACS1#/CAWE1#	73

Pin Name	Pin No.
CACS2#/CAWE2#	72
CACS3#/CAWE3#	71
CACS4#/CAWE4#	70
CACS5#/CAWE5#	69
CACS6#/CAWE6#	68
CACS7#/CAWE7#	67
CAS0#	104
CAS1#	103
CAS2#	102
CAS3#	101
CAS4#	100
CAS5#	99
CAS6#	98
CAS7#	95
CLK	180
D/C#	3
DBCOE0#	192
DBCOE1#	191
DEVSEL#	138
DIRTYI	197
DIRTYWE#,RAS5#	196
DLE0#	193
DLE1#	194
DWE#	90
EADS#/(WB/WT#)	9
ECA4/ADSC#	65
ECAWE#/CACS0O#	62
ECDOE#	58
ECLK	183
FRAME#	135
GND	15
GND	38
GND	60
GND	83
GND	96
GND	119
GND	141
GND	142
GND	163
GND	164
GND	181
GND	182
GND	200
HA3	16
HA4	17
HA5	18
HA6	19
HA7	20
HA8	21
HA9	22
HA10	23
HA11	24

Pin Name	Pin No.
HA12	25
HA13	26
HA14	27
HA15	28
HA16	29
HA17	30
HA18	31
HA19	32
HA20	33
HA21	34
HA22	35
HA23	36
HA24	37
HA25	39
HA26	40
HA27	41
HA28	42
HA29	44
HA30	45
HA31	46
HACALE	64
HDOE#	190
HITM#	55
HLDA	12
HOLD	11
HREQ/TMOD#	187
IRDY#	136
KEN#/LMEM#	8
LA9	122
LA10	121
LA11	120
LA12	118
LA13	117
LA14	116
LA15	115
LA16	113
LA17	112
LA18	111
LA19	110
LA20	109
LA21	108
LA22	107
LA23	106
LCLK	140
LOCK#	14
LRDY#/EDOOE1#	188
M/IO#	2
MA0	89
MA1	88
MA2	87
MA3	85
MA4	84

Pin Name	Pin No.
MA5	82
MA6	81
MA7	80
MA8	79
MA9	78
MA10	77
MA11,RAS4#	76
MASTER#	126
MDLE#/3VDRAM#	186
MDOE#	189
MPERR#	185
MRDC#	123
MWTC#	124
NA#	7
NVMCS/EDOOE0#	127
OCA4/ECA3/ADV#	66
OCAWE#/CACS1O#	63
OCDOE#	59
PLOCK#	134
PWRGD	195
RAS0#	94
RAS1#	93
RAS2#	92
RAS3#	91
REFRESH#	105
RESET	1
SMIACT#	56
STOP#	139
SUSPEND	129
TAG0	207
TAG1	206
TAG2	205
TAG3	204
TAG4	203
TAG5	202
TAG6	199
TAG7	198
TAGWE#	208
TRDY#	137
VCC	128
VCC	147
VCC	165
VCC	179
VCC	184
VCC	201
VCC	86
VCC	97
VCC	114
VCC3	10
VCC3	43
VCC3	61
(W/R#)/INV/DIRTYO	4

3.3 82C557M Signal Descriptions

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. When the PWRGD signal makes a low-to-high transition, RESET is asserted 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 82C557M 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 VL bus state machine within the 82C557M. The same clock or another identical signal is used by the local bus devices. For a synchronous PCI/VL implementation, this signal can be skewed from the CLK input by a margin of ±2ns.

3.3.2 CPU 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 82C557M. They are used for address decoding and second level cache tag lookup sequences. During inquire cycles, the HA[31:5] are outputs from the 82C557M to the CPU to snoop the first level cache tags. They also are outputs from the 82C557M to the L2 cache. HA[31:3] have internal pull-downs, however, external pull-ups should be used on HA3 and HA4.
BE[7:0]#	47:54	I	**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 82C557M 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 when HLDA is active.

82C557M 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 82C557M'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. ADS# has an internal pull-up resistor that is disabled when the system is in the Suspend mode.
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 82C557M 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 82C557M asserts this signal to inform the 82C558M that local system memory needs to be accessed. The 82C558M 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).

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

82C557M Signal Descriptions (cont.)

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.
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.

82C557M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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.
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#.

82C557M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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.
EDOOE1#/LRDY#	188	I/O	**EDO Data RAM Output Enable 1 and Local Ready:** Refer to the LRDY#/EDOOE1# signal description in Section 3.3.7 for details.
EDOOE0#/ NVMCS	127	O	**EDO Data RAM Output Enable 0 and NVRAM Chip Select:** Refer to the NVMCS#/EDOOE0# signal description in Section 3.3.5 for details.

3.3.5 AT Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
NVMCS/ EDOOE0#	127	O	**NVRAM Chip Select and EDO Data RAM Output Enable 0:** 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. This signal also functions as EDOOE0# for accesses to the lower 32 bits of EDO DRAM memory. If the current cycle has been decoded as a local system memory access and if EDO DRAM is present, then this signal is activated for reads.

82C557M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
LA[23:9]	106:113, 115:118, 120:122	I/O	**System Address Bus:** LA[23:9] and SA[8:0] on the 82C558M provide the memory and I/O access on the ISA bus and VL bus. The addresses are outputs when the 82C557M 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 82C558M'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 82C557M about getting into the Suspend mode. SUSPEND needs to be pulled low to resume normal operation.

3.3.6 Bus Arbiter 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 82C558M. 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.
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.
BOFF#	75	O	**IDE DMA Back-off:** This pin is connected to the BOFF# input of the CPU. This signal is asserted by the 82C557M when it senses a potential deadlock condition in systems that incorporate PCI-to-PCI bridges.

82C557M Signal Descriptions (cont.)

3.3.7 PCI/VL 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 82C557M 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 82C557M to a PCI cycle if it is not a local memory cycle. FRAME# is asserted by the bus master, 82C557M (CPU) or PCI to indicate the beginning and the duration of an access. FRAME# is an input when the 82C557M 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 82C557M during CPU cycles to the PCI bus. IRDY# is an input when the 82C557M 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 82C557M when the 82C557M is the PCI slave. TRDY# is an input when the 82C557M 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 82C557M when 82C557M 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 82C557M 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 82C557M. 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 82C557M will drive the AD lines by default. This bus also serves as a conduit for receiving address information during ISA master cycles. The 82C558M conveys the SA[8:0] information to the 82C557M on the AD lines.

82C557M 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#/EDOOE1#	188	I/O	**Local Ready and EDO Data RAM Output Enable 1:** The VL bus cycle will be terminated by asserting LRDY#. The 82C558M terminates VL memory requests by asserting LRDY# when other masters own the bus. This signal also functions as EDOOE1# for accesses to the upper 32 bits of EDO DRAM memory. If the current cycle has been decoded as a local system memory access and if EDO DRAM is present, then this signal is activated for reads. This signal should be pulled up externally.

3.3.8 82C556M/82C558M Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
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 82C556M. These commands inform the 82C556M 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 3-5 (in the 82C556M Signal Description Section).
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 82C556M. 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 82C556M. 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 82C556M's DLE[1:0]# pins and used to latch the HD and MD data bus depending on which cycle is occurring.
MPERR#	185	I/O	**Memory Parity Error:** This signal is an input to the 82C557M from the 82C556M. The 82C557M generates PEN# internally if the corresponding register is programmed to enable parity. The 82C557M qualifies the MPERR# signal with the internally generated PEN#.

82C557M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
MDLE#/ 3VDRAM#	186	I/O	**Memory Data Latch Enable and strap option for 3.3V DRAM:** It is connected to the 82C558M'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-M Desktop Chipset.
MRDC#	123	I	**Memory Read Command:** This input is connected to the 82C558M's MEMR# pin to monitor ISA memory read operations.
MWTC#	124	I	**Memory Write Command:** This input is connected to the 82C558M's MEMW# pin to monitor ISA memory write operations.

3.3.9 Power and Ground Pins

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

Table 3-8 82C557M 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	
LRDY#/EDOOE1#	188	Bidirectional	
NVMCS/EDOOE0#	127	Output	

Table 3-9 82C557M 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-M Desktop Chipset.
MPERR#		This pin must be strapped low for EDO DRAM functionality.
MASTER#		Strap low for ATE (automatic test equipment) Test Modes 0 and 1.
AEN#		Strap low for ATE Test Mode 0. Strap high for ATE Test Mode 1.
REFRESH#		Strap high for ATE Test Modes 0 and 1.

Table 3-10 82C557M System Operating Voltage Groups

TTL (5.0V)	DRAM (5.0V)	AT (5.0V)	CPU/Cache (3.3V)
AD[31:0]	CAS[7:0]#	AEN	ADS#
C/BE[3:0]#	DWE#	LA[23:9]	AHOLD
CLK	MA[10:0]	MASTER#	BE[7:0]#
DBCOE[1:0]#	MA11,RAS4#	MRDC#	BOFF#
DEVSEL#	RAS[3:0]#	MWTC#	BRDY#
DIRTYI		NVMCS/EDOOE0#	CACHE#
DIRTYWE#,RAS5#		REFRESH#	CACS[7:0]#/CAWE[7:0]#
DLE[1:0]#			D/C#
ECLK			EADS#/WB
FRAME#			ECA4/ADSC#
HDOE#			ECAWE#/CACS0O#
HREQ/TMOD#			ECDOE#
IRDY#			HA[31:3]
LCLK			HACALE
LRDY#/EDOOE1#			HITM#
MDLE#/3VDRAM#			HLDA
MDOE#			HOLD
MPERR#			KEN#/LMEM#
PLOCK#			LOCK#
PWRGD			M/IO#
STOP#			NA#
SUSPEND			OCDOE#
TAG[7:0]			OCAWE#/CACS1O#
TAGWE#			OCA4/OCA3/ADV#
TRDY#			SMIACT#
			RESET
			(W/R#)/INV/DIRTYO

Figure 3-3 82C558M Pin Diagram

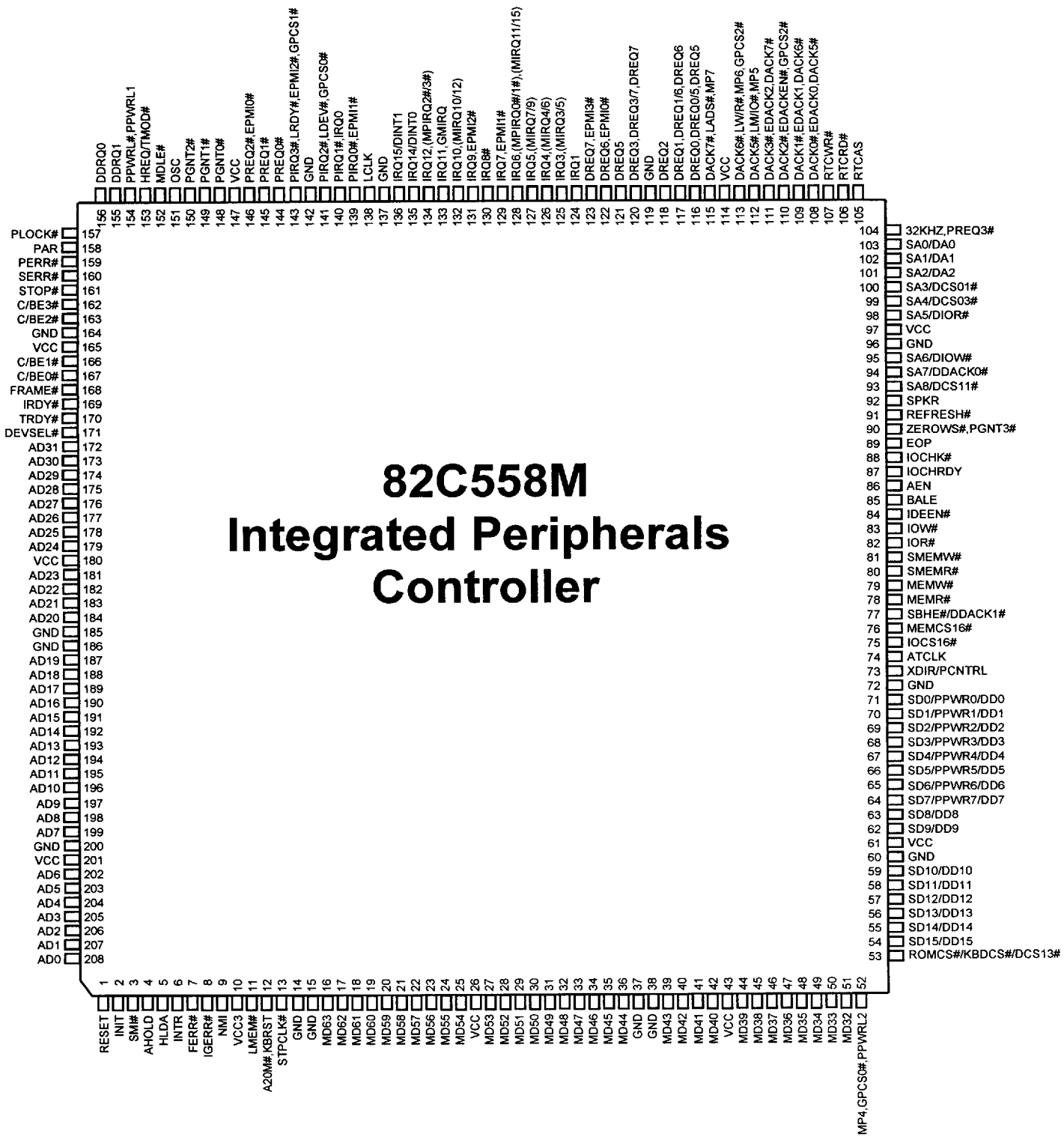


Table 3-11 82C558M Numerical Pin Cross-Reference List

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
1	RESET	I	Sch		
2	INIT	O		4	
3	SMI#	O		4	
4	AHOLD	I	TTL		
5	HLDA	I	TTL		
6	INTR	I/O	TTL	4	
7	FERR#	I	TTL		
8	IGERR#	O		4	
9	NMI	O		4	
10	VCC3	I	P		
11	LMEM#	I	TTL		
12	A20M#,KBRST	I/O	TTL	4	
13	STPCLK#	O		4	If using an M1 processor, leave pin unconnected.
14	GND	I	G		
15	GND	I	G		
16	MD63	I/O	TTL	4	
17	MD62	I/O	TTL	4	
18	MD61	I/O	TTL	4	
19	MD60	I/O	TTL	4	
20	MD59	I/O	TTL	4	
21	MD58	I/O	TTL	4	
22	MD57	I/O	TTL	4	
23	MD56	I/O	TTL	4	
24	MD55	I/O	TTL	4	
25	MD54	I/O	TTL	4	
26	VCC	I	P		
27	MD53	I/O	TTL	4	
28	MD52	I/O	TTL	4	
29	MD51	I/O	TTL	4	
30	MD50	I/O	TTL	4	
31	MD49	I/O	TTL	4	
32	MD48	I/O	TTL	4	
33	MD47	I/O	TTL	4	
34	MD46	I/O	TTL	4	
35	MD45	I/O	TTL	4	
36	MD44	I/O	TTL	4	
37	GND	I	G		
38	GND	I	G		
39	MD43	I/O	TTL	4	
40	MD42	I/O	TTL	4	
41	MD41	I/O	TTL	4	
42	MD40	I/O	TTL	4	
43	VCC	I	P		
44	MD39	I/O	TTL	4	
45	MD38	I/O	TTL	4	
46	MD37	I/O	TTL	4	
47	MD36	I/O	TTL	4	
48	MD35	I/O	TTL	4	
49	MD34	I/O	TTL	4	
50	MD33	I/O	TTL	4	
51	MD32	I/O	TTL	4	
52	MP4,GPCS0#, PPWRL2	O		4	If MP4, ensure that the MP0 line to 82C556M has ext PD.
53*	ROMCS#/KBDCS#/ DCS13#	O		4	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
54	SD15/DD15	I/O	TTL	8	
55	SD14/DD14	I/O	TTL	8	
56	SD13/DD13	I/O	TTL	8	
57	SD12/DD12	I/O	TTL	8	
58	SD11/DD11	I/O	TTL	8	
59	SD10/DD10	I/O	TTL	8	
60	GND	I	G		
61	VCC	I	P		
62	SD9/DD9	I/O	TTL	8	
63	SD8/DD8	I/O	TTL	8	
64	SD7/PPWR7/DD7	I/O	TTL	8	Ext PU
65	SD6/PPWR6/DD6	I/O	TTL	8	Ext PU
66	SD5/PPWR5/DD5	I/O	TTL	8	Ext PU
67	SD4/PPWR4/DD4	I/O	TTL	8	Ext PU
68	SD3/PPWR3/DD3	I/O	TTL	8	Ext PU
69	SD2/PPWR2/DD2	I/O	TTL	8	Ext PU
70	SD1/PPWR1/DD1	I/O	TTL	8	Ext PU
71	SD0/PPWR0/DD0	I/O	TTL	8	Ext PU
72	GND	I	G		
73	XDIR/PCNTRL	I/O	TTL	4	
74	ATCLK	O		8	
75	IOCS16#	I	TTL		
76	MEMCS16#	I/O	TTL	8	
77*	SBHE#/DDACK1#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
78	MEMR#	I/O	TTL	8	
79	MEMW#	I/O	TTL	8	
80	SMEMR#	O			
81	SMEMW#	O			
82	IOR#	I/O	Sch	8	
83	IOW#	I/O	Sch	8	
84	IDEEN#	O		8	Should be connected to the output enable of the 74F244.
85	BALE	O		8	
86	AEN	I/O	TTL	8	
87	IOCHRDY	I/O	TTL	8	If local bus IDE is being used, DCHRDY from the IDE drives could be connected to this signal directly or through a buffer.
88	IOCHK#	I	TTL		
89	EOP	I/O	TTL	8	
90	ZEROWS#,PGNT3#	I/O	TTL	PCI	
91	REFRESH#	I/O	TTL	8	
92	SPKR	O		16	

82C558M Numerical Pin Cross-Reference List (cont.)

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
93*	SA8/DCS11#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
94*	SA7/DDACK0#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
95*	SA6/DIOW#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
96	GND	I	G		
97	VCC	I	P		
98*	SA5/DIOR#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
99*	SA4/DCS03#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
100*	SA3/DCS01#	I/O	TTL	8	This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M.
101	SA2/DA2	I/O	TTL	8	
102	SA1/DA1	I/O	TTL	8	
103	SA0/DA0	I/O	TTL	8	
104	32KHZ,PREQ3#	I	Sch		
105	RTCAS	O		4	
106	RTCRD#	O		4	
107	RTCWR#	O		4	
108	DACK0#,EDACK0, DACK5#	O		6	If EDACK0, ext decoder required.
109	DACK1#,EDACK1, DACK6#	O		6	If EDACK1, ext decoder required.
110	DACK2#, EDACKEN#,GPCS2#	O		6	If EDACKEN#, ext decoder required.
111	DACK3#,EDACK2, DACK7#	O		6	If EDACK2, ext decoder required.
112	DACK5#, LM/IO#,MP5	O		6	If MP5, ensure that the MP0 line to 82C556M has ext PD.
113	DACK6#,LW/R#, MP6,GPCS2#	O		6	If MP6, ensure that the MP0 line to 82C556M has ext PD.
114	VCC	I	P		
115	DACK7#,LADS#,MP7	O		6	If MP7, ensure that the MP0 line to 82C556M has ext PD.
116	DREQ0,DREQ0/ 5,DREQ5	I	TTL		If DREQ0/5, ext multiplexer is required.
117	DREQ1, DREQ1/6,DREQ6	I	TTL		If DREQ1/6, ext multiplexer required.
118	DREQ2	I	TTL		
119	GND	I	G		
120	DREQ3, DREQ3/7,DREQ7	I	TTL		If DREQ3/7, ext multiplexer required.
121	DREQ5	I	TTL		
122	DREQ6,EPMI0#	I	TTL		If EPMI0#, ext PU required.
123	DREQ7,EPMI3#	I	TTL		If EPMI3#, ext PU required.
124	IRQ1	I	TTL		
125	IRQ3,(MIRQ3/5)	I	TTL		
126	IRQ4,(MIRQ4/6)	I	TTL		If MIRQ4/6, ext multiplexer required.
127	IRQ5,(MIRQ7/9)	I	TTL		
128	IRQ6,(MPIRQ0#/1#), (MIRQ11/15)	I	TTL		If multiplexed input, ext multiplexer required.
129	IRQ7,EPMI1#	I	TTL		If EPMI1#, ext PU required.
130	IRQ8#	I	TTL		
131	IRQ9,EPMI2#	I	TTL		If EPMI2#, ext PU required.
132	IRQ10,(MIRQ10/12)	I	TTL		If MIRQ10/12, ext multiplexer required.
133	IRQ11,GMIRQ	I	TTL		
134	IRQ12, (MPIRQ2#/3#)	I	TTL		If MPIRQ2#/3#, ext multiplexer required.
135	IRQ14/DINT0	I	TTL		If the local bus IDE is being used, then the DINT0 output from the primary drive should be directly wired to this input.
136	IRQ15/DINT1	I	TTL		If the local bus IDE is being used, then the DINT1 output from the primary drive should be directly wired to this input.
137	GND	I	G		
138	LCLK	I	TTL		
139	PIRQ0#,EPMI1#	I/O	TTL	4	
140	PIRQ1#,IRQ0	I/O	TTL	4	
141	PIRQ2#,LDEV#, GPCS0#	I	TTL	4	If LDEV#, should have ext PU.
142	GND	I	G		
143	PIRQ3#,LRDY#, EPMI2#,GPCS1#	I/O	TTL	4	If LRDY# or EPMI2#, ext PU required.
144	PREQ0#	I	TTL		Ext PU
145	PREQ1#	I	TTL		Ext PU
146	PREQ2#,EPMI0#	I	TTL		Ext PU
147	VCC	I	P		
148	PGNT0#	O		PCI	
149	PGNT1#	O		PCI	
150	PGNT2#	O		PCI	
151	OSC	I	TTL		
152	MDLE#	I	TTL		
153	HREQ/TMOD#	I/O	TTL	4	
154	PPWRL#,PPWRL1	O		4	
155	DDRQ1	I	TTL		
156	DDRQ0	I	TTL		
157	PLOCK#	I	TTL		
158	PAR	I/O	TTL	PCI	
159	PERR#	I/O	TTL	PCI	

82C558M Numerical Pin Cross-Reference List (cont.)

Pin No.	Pin Name	Pin Type	Signal Type	Drive (mA)	Comment
160	SERR#	I/O	TTL	PCI	
161	STOP#	I/O	TTL	PCI	
162	C/BE3#	I/O	TTL	PCI	
163	C/BE2#	I/O	TTL	PCI	
164	GND	I	G		
165	VCC	I	P		
166	C/BE1#	I/O	TTL	PCI	
167	C/BE0#	I/O	TTL	PCI	
168	FRAME#	I/O	TTL	PCI	
169	IRDY#	I/O	TTL	PCI	
170	TRDY#	I/O	TTL	PCI	
171	DEVSEL#	I/O	TTL	PCI	
172	AD31	I/O	TTL	PCI	
173	AD30	I/O	TTL	PCI	
174	AD29	I/O	TTL	PCI	
175	AD28	I/O	TTL	PCI	
176	AD27	I/O	TTL	PCI	
177	AD26	I/O	TTL	PCI	
178	AD25	I/O	TTL	PCI	
179	AD24	I/O	TTL	PCI	
180	VCC	I	P		
181	AD23	I/O	TTL	PCI	
182	AD22	I/O	TTL	PCI	
183	AD21	I/O	TTL	PCI	
184	AD20	I/O	TTL	PCI	
185	GND	I	G		
186	GND	I	G		
187	AD19	I/O	TTL	PCI	
188	AD18	I/O	TTL	PCI	
189	AD17	I/O	TTL	PCI	
190	AD16	I/O	TTL	PCI	
191	AD15	I/O	TTL	PCI	
192	AD14	I/O	TTL	PCI	
193	AD13	I/O	TTL	PCI	
194	AD12	I/O	TTL	PCI	
195	AD11	I/O	TTL	PCI	
196	AD10	I/O	TTL	PCI	
197	AD9	I/O	TTL	PCI	
198	AD8	I/O	TTL	PCI	
199	AD7	I/O	TTL	PCI	
200	GND	I	G		
201	VCC	I	P		
202	AD6	I/O	TTL	PCI	
203	AD5	I/O	TTL	PCI	
204	AD4	I/O	TTL	PCI	
205	AD3	I/O	TTL	PCI	
206	AD2	I/O	TTL	PCI	
207	AD1	I/O	TTL	PCI	
208	AD0	I/O	TTL	PCI	

*These signals are multiplexed on the Viper-M Chipset. For IDE cycles, IDEEN# will drive the 74F244 active allow the signals to pass to the IDE interface. For all other cycles, IDEEN# will remain inactive and the corresponding signals will drive to the appropriate destinations.

Legend:

CMOS	CMOS-level compatible	P	Power
Ext	External	PD	Pull-down resistor
G	Ground	PU	Pull-up resistor
Int	Internal	Sch	Schmitt-trigger
I/O	Input/Output	TTL	TTL-level compatible
OD	Open drain		

Table 3-12 82C558M Alphabetical Pin Cross-Reference List

Pin Name	Pin No.
AD0	208
AD1	207
AD2	206
AD3	205
AD4	204
AD5	203
AD6	202
AD7	199
AD8	198
AD9	197
AD10	196
AD11	195
AD12	194
AD13	193
AD14	192
AD15	191
AD16	190
AD17	189
AD18	188
AD19	187
AD20	184
AD21	183
AD22	182
AD23	181
AD24	179
AD25	178
AD26	177
AD27	176
AD28	175
AD29	174
AD30	173
AD31	172
AEN	86
AHOLD	4
ATCLK	74
A20M#,KBRST	12
BALE	85
C/BE0#	167
C/BE1#	166
C/BE2#	163
C/BE3#	162
DACK0#,EDACK0, DACK5#	108
DACK1#,EDACK1, DACK6#	109
DACK2#,EDACKEN#, GPCS2#	110
DACK3#,EDACK2, DACK7#	111
DACK5#,LM/IO#,MP5	112
DACK6#,LW/R#, MP6,GPCS2#	113
DACK7#,LADS#,MP7	115
DDRQ0	156
DDRQ1	155

Pin Name	Pin No.
DEVSEL#	171
DREQ0,DREQ0/5, DREQ5	116
DREQ1,DREQ1/6, DREQ6	117
DREQ2	118
DREQ3,DREQ3/7, DREQ7	120
DREQ5	121
DREQ6,EPMI0#	122
DREQ7,EPMI3#	123
EOP	89
FERR#	7
FRAME#	168
GND	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
HLDA	5
HREQ/TMOD#	153
IDEEN#	84
IGERR#	8
INIT	2
INTR	6
IOCHK#	88
IOCHRDY	87
IOCS16#	75
IOR#	82
IOW#	83
IRDY#	169
IRQ1	124
IRQ3,(MIRQ3/5)	125
IRQ4,(MIRQ4/6)	126
IRQ5,(MIRQ7/9)	127
IRQ6,(MPIRQ0#/1#), (MIRQ11/15)	128
IRQ7,EPMI1#	129
IRQ8#	130
IRQ9,EPMI2#	131
IRQ10,(MIRQ10/12)	132
IRQ11,GMIRQ	133
IRQ12,(MPIRQ2#/3#)	134
IRQ14/DINT0	135
IRQ15/DINT1	136
LCLK	138

Pin Name	Pin No.
LMEM#	11
MD32	51
MD33	50
MD34	49
MD35	48
MD36	47
MD37	46
MD38	45
MD39	44
MD40	42
MD41	41
MD42	40
MD43	39
MD44	36
MD45	35
MD46	34
MD47	33
MD48	32
MD49	31
MD50	30
MD51	29
MD52	28
MD53	27
MD54	25
MD55	24
MD56	23
MD57	22
MD58	21
MD59	20
MD60	19
MD61	18
MD62	17
MD63	16
MDLE#	152
MEMCS16#	76
MEMR#	78
MEMW#	79
MP4,GPCS0#,PPWRL2	52
NMI	9
OSC	151
PAR	158
PERR#	159
PGNT0#	148
PGNT1#	149
PGNT2#	150
PIRQ0#,EPMI1#	139
PIRQ1#,IRQ0	140
PIRQ2#,LDEV#, GPCS0#	141
PIRQ3#,LRDY#, EPMI2#,GPCS1#	143
PLOCK#	157
PPWRL#,PPWRL1	154
PREQ0#	144
PREQ1#	145

Pin Name	Pin No.
PREQ2#,EPMI0#	146
REFRESH#	91
RESET	1
ROMCS#/KBDCS#/ DCS13#	53
RTCAS	105
RTCRD#	106
RTCWR#	107
SA0/DA0	103
SA1/DA1	102
SA2/DA2	101
SA3/DCS01#	100
SA4/DCS03#	99
SA5/DIOR#	98
SA6/DIOW#	95
SA7/DDACK0#	94
SA8/DCS11#	93
SBHE#/DDACK1#	77
SD0/PPWR0/DD0	71
SD1/PPWR1/DD1	70
SD2/PPWR2/DD2	69
SD3/PPWR3/DD3	68
SD4/PPWR4/DD4	67
SD5/PPWR5/DD5	66
SD6/PPWR6/DD6	65
SD7/PPWR7/DD7	64
SD8/DD8	63
SD9/DD9	62
SD10/DD10	59
SD11/DD11	58
SD12/DD12	57
SD13/DD13	56
SD14/DD14	55
SD15/DD15	54
SERR#	160
SMEMR#	80
SMEMW#	81
SMI#	3
SPKR	92
STOP#	161
STPCLK#	13
TRDY#	170
VCC	26
VCC	43
VCC	61
VCC	97
VCC	114
VCC	147
VCC	165
VCC	180
VCC	201
VCC3	10
XDIR/PCNTRL	73
ZEROWS#,PGNT3#	90
32KHZ,PREQ3#	104

3.4 82C558M 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 82C557M. It is used by the PCI and the local bus state machine within the 82C558M and the 82C557M. It is also used by the 82C558M 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	151	I	**Timer Oscillator:** This is the main clock used by the internal 8254 timers. It is connected to the 14.31818MHz oscillator.
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 82C557M in response to a PWRGD input.

3.4.2 IDE Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
DCS13#/ ROMCS#/ KBDCS#	53*	O	**Secondary Drive Chip Select 3 / BIOS ROM Chip Select / Keyboard Chip Select:** This output has multiple functions. When configured as DCS13#, this pin functions as the chip select signal for the secondary IDE. After the 82C557M translates the CPU cycle to a PCI cycle, the 82C558M decodes the address on the AD lines and asserts this signal to select the command block register for the primary IDE. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the KBDCS#/ROMCS#/DCS13# signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the KBDCS#/ROMCS# functions of this pin.
DDACK1#/SBHE#	77*	I/O	**Secondary Drive DMA Acknowledge / System Byte High Enable:** When configured as DDACK1#, this pin functions as the secondary drive DMA acknowledge signal. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SBHE#/DDACK1# signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SBHE# function of this pin.

*These signals are multiplexed on the Viper-M Chipset. For IDE cycles, IDEEN# will drive the 74F244 active allow the signals to pass to the IDE interface. For all other cycles, IDEEN# will remain inactive and the corresponding signals will drive to the appropriate destinations.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
DCS11#/SA8	93*	I/O	**Secondary Drive Chip Select 1 / System Address Bus Line 8:** When configured as DCS11#, this pin functions as the chip select signal for the secondary IDE. After the 82C557M translates the CPU cycle to a PCI cycle, the 82C558M decodes the address on the AD lines and asserts this signal to select the command block register for the primary IDE. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal descriptions in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA8 function of this pin.
DDACK0#/SA7	94*	I/O	**Primary Drive DMA Acknowledge / System Byte High Enable:** When configured as DDACK0#, this pin functions as the primary drive DMA acknowledge signal. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA7 function of this pin.
DIOW#/SA6	95*	I/O	**IDE I/O Write Strobe / System Address Bus Line 6:** When configured as DIOW#, this pin functions as the write strobe signal for the primary and secondary IDE drives. The rising edge of DIOW# samples data from the IDE data bus (DA[15:0]) into a register or the data port of the drive. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA6 function of this pin.
DIOR#/SA5	98*	I/O	**IDE I/O Read Strobe / System Address Bus Line 5:** When configured as DIOR#, this pin functions as the read strobe signal for the primary and secondary IDE drives. The low level of DIOR# enables data from a register or the data port of the drive on the data bus (DD[15:0]). This signal should be buffered with a 74F244. This buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA5 function of this pin.
DCS03#/SA4	99*	I/O	**Primary Drive Chip Select 3 / System Address Bus Line 4:** When configured as DCS03#, this pin functions as the chip select signal for the primary IDE. After the 82C557M translates the CPU cycle to a PCI cycle, the 82C558M decodes the address on the AD lines and asserts this signal to select the command block register for the primary IDE. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA4 function of this pin.

*These signals are multiplexed on the Viper-M Chipset. For IDE cycles, IDEEN# will drive the 74F244 active allow the signals to pass to the IDE interface. For all other cycles, IDEEN# will remain inactive and the corresponding signals will drive to the appropriate destinations.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
DCS01#/SA3	100*	I/O	**Primary Drive Chip Select 1 / System Address Bus Line 3:** When configured as DCS01#, this pin functions as the chip select signal for the primary IDE. After the 82C557M translates the CPU cycle to a PCI cycle, the 82C558M decodes the address on the AD lines and asserts this signal to select the command block register for the primary IDE. This signal should be buffered and this buffered output will be valid only when IDEEN# is asserted by the 82C558M. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA3 function of this pin.
DA[2:0]/SA[2:0]	101:103	I/O	**IDE Drive Address Lines 2-0 / System Address Bus Lines 2-0:** This 3-bit binary coded address is asserted by the 82C558M to access a register or a data port in the primary/secondary IDE drive. It is recommended that these signals be buffered and driven to the IDE drives. Refer to the SA[8:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the SA[2:0] function of this pin.
IDEEN#	84	O	**IDE Enable:** This signal is asserted by the 82C558M for all accesses to and from the IDE drives. Most of the IDE control signals need to be buffered via a 74F244 and driver to the IDE drives. IDEEN# should be connected to the output enable of the 244's. This will prevent the IDE drives from responding to any other cycle.
DD[7:0]/SD[7:0]/ PPWR[7:0]	64:71	I/O	**Disk Data Lines 7-0 / System Data Bus Lines 7-0 / Peripheral Power Control Lines 7-0:** DD[7:0] along with DD[15:8] provide the 16-bit data path for the IDE disk drives. Refer to the SD[7:0]/PPWR[7:0]/DD[7:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the other functions of these pins.
DD[15:8]/SD[15:8]	54:59, 62, 63	I/O	**Disk Data Lines 7-0 / System Data Bus Lines 7-0:** DD[7:0] along with DD[15:8] provide the 16-bit data path for the IDE disk drives. Refer to the SD[7:0]/PPWR[7:0]/DD[7:0] signal description in Section 3.4.6, ISA Bus Interface Signals, for details regarding the other functions of these pins.
DDRQ1	155	I	**Secondary IDE Drive Data Request:** The secondary master IDE drive asserts this signal to the 82C558M when it needs control of the bus. It is recommended that this line be pulled low externally.
DDRQ0	156	I	**Primary IDE Drive Data Request:** The primary master IDE drive asserts this signal to the 82C558M when it needs control of the bus. It is recommended that this line be pulled low externally.

*These signals are multiplexed on the Viper-M Chipset. For IDE cycles, IDEEN# will drive the 74F244 active allow the signals to pass to the IDE interface. For all other cycles, IDEEN# will remain inactive and the corresponding signals will drive to the appropriate destinations.

82C558M Signal Descriptions (cont.)

3.4.3 82C557M and 82C556M 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 82C557M'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	I/O	**Hold Request / Test Mode:** This signal is connected to the 82C557M's HREQ pin to indicate that there is a master or DMA cycle request from the 82C558M. During power-up reset if TMOD is high, it means normal operation, if TMOD is low, the system enters test mode.
LMEM#	11	I	**Local Memory Accessed Indication:** This signal serves as a local device memory accessed indication during local bus master cycles.

3.4.4 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 82C558M 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 82C558M when the 82C558M is the target. FRAME# is an output when the 82C558M is the initiator.
IRDY#	169	I/O	**Initiator Ready:** IRDY# indicates the 82C558M'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 82C558M when the 82C558M is the target and an output when the 82C558M is an initiator.
TRDY#	170	I/O	**Target Ready:** TRDY# indicates the 82C558M'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 82C558M when the 82C558M is the initiator and an output when the 82C558M is a target.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
DEVSEL#	171	I/O	**Device Select:** The 82C558M asserts DEVSEL# to claim a PCI transaction. As an output, the 82C558M asserts DEVSEL# when it samples configuration cycles to the 82C558M 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 82C558M asserts DEVSEL# to claim the cycle and initiates an ISA cycle.
STOP#	161	I/O	**STOP:** STOP# indicates that the 82C558M, as a target, is requesting a master to stop the current transaction. As a master, STOP# causes the 82C558M to stop the current transaction. STOP# is an output when the 82C558M is a target and an input when the 82C558M is an initiator
PLOCK#	157	I	**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.
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 82C557M during ISA master cycles. It conveys the SA[8:0] information to the 82C557M on these lines.
PIRQ0#,EPMI1#	139	I/O	**PCI Interrupt Request 0 or External PMI 1:** This pin can be programmed to be PCI interrupt request 0, or an external power management interrupt 1. If configured to be an EPMI1# input, this signal should have an external pull-up.
PIRQ1#,IRQ0	140	I/O	**PCI Interrupt Request 1 or Interrupt Request 0:** This pin can be programmed to PCI interrupt request 1 or interrupt request 0. If IRQ6 has been programmed to take on the MPCIRQ0#/1# functionality, this pin should be tied high.
PIRQ2#, LDEV#, GPCS0#	141	I	**PCI Interrupt 2, or VL Device Indicator, or General Purpose Chip Select 0:** This pin can be programmed to be PCI interrupt 2, a VL device indicator, or as general purpose chip select 0. Address Offset 45h-44h[12, 1:0] determine the functionality of this pin. If configured as LDEV#, it should have an external pull-up.
PIRQ3#, LRDY#, EPMI2#, GPCS1#	143	I/O	**PCI Interrupt 3, or VL Ready, or External PMI 2, or General Purpose Chip Select 1:** This pin can be programmed to be PCI interrupt 3, VL ready input, external power management interrupt 2, or as general purpose chip select 1. Address Offset 45h-44h[13, 1:0] determine the functionality of this pin. If configured as LRDY# or EPMI2#, 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.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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#, EPMI0#	146	I	**PCI Request 2 or External PMI 0:** 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. This pin can also be programmed as an external power management interrupt.
PGNT[2:0]#	150:148	O	**PCI Grants 2-0:** An active low assertion indicates that one of the initiators on PCI slot 1, 2, or 3 has been granted use of the PCI bus.

3.4.5 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 82C557M'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	6	I/O	**Interrupt Request:** INTR is driven by the 82C558M 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.
NMI	9	O	**Non-Maskable Interrupt:** 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 82C558M also generates an NMI when either PERR# or SERR# is asserted.
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 82C558M in response generates an INIT to the CPU. Index 41h-40h[12] determines the functionality of this pin.

82C558M Signal Descriptions (cont.)

3.4.6 ISA Bus Interface Signals

Signal Name	Pin No.	Signal Type	Signal Description
ROMCS#/ KBDCS#/DCS13#	53	O	**BIOS ROM Chip Select / Keyboard Chip Select / Secondary Drive Chip Select 13:** This output has multiple 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. Refer to the DCS13#/KBDCS#/ROMCS# signal description in Section 3.4.2, IDE Interface Signals, for the for details regarding the DCS13# function of this pin.
SBHE#/DDACK1#	77	I/O	**System Byte High Enable / Secondary Disk Drive DMA Acknowledge 1:** 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 82C558M owns the ISA bus. Refer to the DDACK1#/SBHE# signal description in Section 3.4.2, IDE Interface Signals, for details regarding the DDACK1# function of this pin.
SA8/DCS11# SA7/DACK0# SA6/DIOW# SA5/DIOR# SA4/DCS03# SA3/DCS01# SA2/DA2 SA1/DA1 SA0/DA0	93 94 95 98 99 100 101 102 103	I/O	**System Address Bus Lines 8-0 / Various IDE control signals:** The SA[8:0] and LA[23:9] signals on the 82C557M provide the address for memory and I/O accesses on the ISA bus. The addresses (SA[8:0]) are outputs when the 82C558M owns the ISA bus and are inputs when an external ISA master owns the ISA bus. SA[8:0] can be pin multiplexed with various IDE control signals. Refer to Section 3.4.2, IDE Interface Signals, for details.
SD[7:0]/ PPWR[7:0]/ DD[7:0]	64:71	I/O	**System Data Bus Lines 7-0 / Peripheral Power Lines 7-0 / Disk Data Lines 7-0:** 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. Refer to the DD[7:0]/SD[7:0]/PPWR[7:0] signal description in Section 3.4.2, IDE Interface Signals, for details regarding the DD[7:0] function of these pins.
SD[15:8]/DD[15:8]	54:59, 62, 63	I/O	**System Data Bus Lines 15-8 / Disk Data Lines 15-8:** SD[15:8] are used along with SD[7:0] to provide the 16-bit data path for devices residing on the ISA bus. Refer to the DD[15:8]/SD[15:8] signal description in Section 3.4.2, IDE Interface Signals, for details regarding the DD[15:8] function of these pins.
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 82C558M 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 82C558M is a master on the ISA bus. MEMR# is an input when an ISA master, other than 82C558M, owns the ISA bus.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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 82C558M owns the ISA bus. MEMW# is an input when an ISA master, other than the 82C558M, owns the ISA bus.
AEN	86	I/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.
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. If the local bus IDE is being used, the DCHRDY signal from the IDE drives could be connected to this signal directly or through a buffer.
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 82C558M 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 82C558M 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 82C558M owns the ISA bus. IOW# is an input when an external ISA master owns the ISA bus.
SMEMR#	80	O	**System Memory Read:** The 82C558M asserts SMEMR# to request a memory slave to provide data. If the access is below the 1MB range (00000000h-000FFFFFh) during DMA compatible, 82C558M master, or ISA master cycles, the 82C558M asserts SMEMR#.
SMEMW#	81	O	**System Memory Write:** The 82C558M 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, 82C558M master, or ISA master cycles, the 82C558M asserts SMEMW#.
ZEROWS#, PGNT3#	90	I/O	**Zero Wait States or PCI Grant 3:** 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. When this pin functions as PGNT3# and it is asserted, it indicates that the device on PCI slot 4 has been granted use of the PCI bus.
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.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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.

3.4.7 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 82C558M's DMA controller. It can be programmed to be DMA request 0, a multiplexed DMA request 0/5, or DMA request 5 through Address Offset 49h-48h[3:2]. 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 82C558M's DMA controller. It can be programmed to be DMA request 1, a multiplexed DMA request 1/6, or DMA request 6 through Address Offset 49h-48h[5:4]. 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 82C558M'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 82C558M's DMA controller. It can be programmed to be DMA request 3, a multiplexed DMA request 3/7, or DMA request 7 through Address Offset 49h-48h[7:6]. 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 82C558M's DMA controller.
DREQ6, EPMI0#	122	I	**DMA Request 6 or External PMI 0:** The DREQ is used to request DMA service from the 82C558M's DMA controller. Address Offset 43h-42h[0] can be programmed to control the functionality of this pin (0 = DREQ6 and 1 = EPMI0#). If configured as EPMI0#, this signal should be pulled up externally.
DREQ7, EPMI3#	123	I	**DMA Request 7 or External PMI 3:** The DREQ is used to request DMA service from the 82C558M's DMA controller. This pin can be programmed as an external power management interrupt. If configured as EPMI3#, this signal should be pulled up externally.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
DACK0#, EDACK0, DACK5#	108	O	**DMA Acknowledge 0, or Encoded DACK0, or DMA Acknowledge 5:** This pin can be programmed to be DMA acknowledge 0, encoded DACK0, or DMA acknowledge 5. Address Offset 45h-44h[7:6] determines the functionality of this pin. If configured as an encoded DMA acknowledge signal, an external decoder is required.
DACK1#, EDACK1, DACK6#,	109	O	**DMA Acknowledge 1, or Encoded DACK1, or DMA Acknowledge 6:** This pin can be programmed to be DMA acknowledge 1, encoded DACK1, or DMA acknowledge 6. Address Offset 45h-44h[5:4] and [1:0] determine the functionality of this pin. If configured as an encoded DMA acknowledge signal, an external decoder is required.
DACK2#, EDACKEN#, GPCS2#	110	O	**DMA Acknowledge 2, or Encoded DACK Enable, or General Purpose Chip Select 2:** This pin can be programmed to be either DMA acknowledge 2, an encoded DACK enable signal, or general purpose chip select 2. Address Offset 45h-44h[1:0] 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. Address Offset 45h-44h[7:6] and [1:0] determine the functionality of this pin. 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. Address Offset 45h-44h[1:0] determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the 82C556M 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 as general purpose chip select 2. Address Offset 45h-44h[1:0] determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the 82C556M 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. Address Offset 45h-44h[1:0] determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the 82C556M has an external pull-down.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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 82C558M to terminate DMA cycles. The 82C558M 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 82C557M to refresh the local DRAM. When another bus master has control of the bus, this pin is an input to the 82C558M.
32KHZ, PREQ3#	104	I	**32KHz Clock or PCI Request 3:** This signal can be used as a 32KHz clock input or as PCI request 3.

3.4.8 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, (MIRQ3/5)	125	I	**Interrupt Request 3 or Multiplexed IRQ3/5:** This pin can be programmed to be IRQ3 or a multiplexed IRQ3/5. The 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. Address Offset 49h-48h[10] determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ5, (MIRQ7/9)	127	I	**Interrupt Request 5 or Multiplexed IRQ7/9:** This pin can be programmed to be IRQ5 or a multiplexed IRQ7/9. This IRQ5 signal provides parallel port 2 with a mechanism for asynchronously interrupting the CPU.
IRQ6, (MPIRQ0#/1#), (MIRQ11/15)	128	I	**Interrupt Request 6, or Multiplexed PCI Interrupt 0/1, or Multiplexed IRQ11/15:** This pin can be programmed to be interrupt request 6, a multiplexed PCI interrupt 0/1, or a multiplexed IRQ11/15. The IRQ6 signal provides the floppy disk controller with a mechanism for asynchronously interrupting the CPU. Address Offset 49h-48h[11] 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,EPMI2#	131	I	**Interrupt Request 9 or External PMI 2:** This pin is used to provide interrupt request 9 to the CPU. This pin can be programmed as an external power management interrupt. If configured as EPMI2#, this signal should be pulled up externally.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
IRQ7,EPMI1#	129	I	**Interrupt Request 7 or External PMI 1:** This IRQ7 signal provides parallel port 1 with a mechanism for asynchronously interrupting the CPU. This pin can be programmed as an external power management interrupt. If configured as EPMI1#, this signal should be pulled up externally.
IRQ10, (MIRQ10/12)	132	I	**Interrupt Request 10 or Multiplexed IRQ10/12:** This pin can be programmed to be interrupt request 10 or a multiplexed interrupt 10/12. Address Offset 49h-48h[12] determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ11,GMIRQ	133	I	**Interrupt Request 11 or General Purpose Motherboard Interrupt Request:** This pin used to provide interrupt request 11 to the CPU. This pin can also function as a general purpose IRQ line on the motherboard. Through register settings, this line can be mapped onto any of the IRQ lines.
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. Address Offset 49h-48h[14:13] determines the functionality of this pin. If configured as a multiplexed input, an external multiplexer is required.
IRQ14/DINT0	135	I	**Interrupt Request 14 or Disk Interrupt 0:** This IRQ14 signal provides the expansion slot with a mechanism for asynchronously interrupting the CPU. If the local bus IDE is being used, then the DINT0 output from the primary drive should be directly wired to this input. If the local bus IDE is enabled, this interrupt will not be available for use on the ISA bus.
IRQ15/DINT1	136	I	**Interrupt Request 15 Disk Interrupt 1:** This IRQ15 signal provides the expansion slot with a mechanism for asynchronously interrupting the CPU. If the local bus IDE is being used, then the DINT1 output from the secondary drive should be directly wired to this input. If the local bus IDE is enabled, this interrupt will not be available for use on the ISA bus.
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.
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.

3.4.9 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.

82C558M Signal Descriptions (cont.)

Signal Name	Pin No.	Signal Type	Signal Description
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.

3.4.10 Miscellaneous Signals

Signal Name	Pin No.	Signal Type	Signal Description
MP4,GPCS0#, PPWRL2	52	O	**Memory Parity 4 or General Purpose Chip Select 0:** This pin can be programmed to be memory parity bit 4 or general purpose chip select 0. Address Offset 45h-44h[1:0] determine the functionality of this pin. If configured as an MP output, ensure that the MP0 line to the 82C556M 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.

3.4.11 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	G	**Ground Connection**
VCC3	10	P	**3.3V Power Connection**
VCC	26, 43, 61, 97, 114, 147, 165, 180, 201	P	**5.0V Power Connection**

Figure 3-4 PCI Interrupts Mapping Matrix

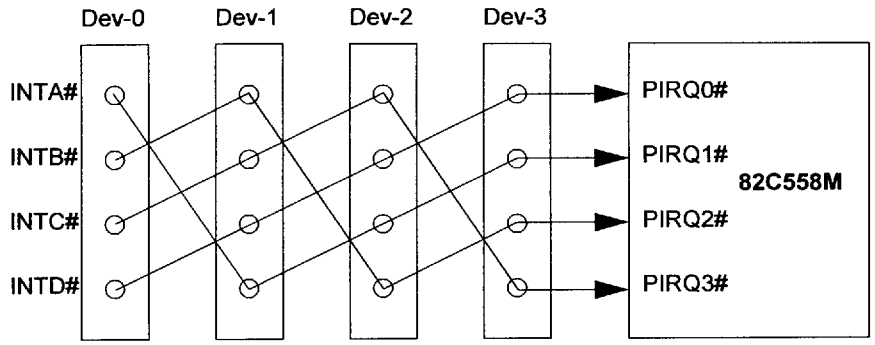


Table 3-13 82C558M Strap Function Pins

Default	Strap	Function
HREQ	TMOD#	Low = Test Mode High = Normal Mode
XDIR	PCNTRL	Low = Internal power management scheme High = External power management scheme.
IRQ0		Strap low for ATE (automatic test equipment) Test Modes 0 and 1. Strap high for ATE Test Modes 3 and 2.
IRQ1		Strap low for ATE Test Modes 2 and 0. Strap High for ATE Test Modes 3 and 1.

Table 3-14 82C558M Multiplexed Function Pins

Type	Multiplexed Functions	Note
Bidirectional	C/BE[3:0]	
Bidirectional	SD[7:0]/PPWR[7:0]/DD[7:0]	Valid ONLY when PPWRL# is selected.
Bidirectional	SD[15:8]/DD[15:8]	
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#	
Output	KBDCS#/ROMCS#/DCS13#	
Bidirectional	SBHE#/DDACK1#	
Bidirectional	SA8/DCS11#	
Bidirectional	SA7/DDACK0#	
Bidirectional	SA6/DIOW#	
Bidirectional	SA5/DIOR#	
Bidirectional	SA4/DCS03#	
Bidirectional	SA3/DCS01#	
Bidirectional	SA[2:0]/DA[2:0]	

Table 3-15 82C558M Programmable Functions Pins

Type	Default	Group-Wise Programmable (Index 44h[1:0])			Pin-Wise Programmable		
		01	10	11			
Input	IRQ1					Level	
Input	IRQ3					Level	MIRQ3/5
Input	IRQ4				MIRQ4/6	Level	
Input	IRQ5					Level	MIRQ7/9
Input	IRQ6				MPIRQ0#/1#	Level	MIRQ11/15
Input	IRQ7					Level	EPMI1#
Input	IRQ9					Level	EPMI2#
Input	IRQ10				MIRQ10/12	Level	
Input	IRQ11					Level	GMIRQ
Input	IRQ12				MPIRQ2#/3#	Level	
Input	IRQ14					Level	
Input	IRQ15					Level	EPMI#
Input	DREQ0				DREQ0/5	DREQ5	
Input	DREQ1				DREQ1/6	DREQ6	
Input	DREQ3				DREQ3/7	DREQ7	
Input	DREQ6				EPMI0#		EPMI#
Input	DREQ7				EPMI3#		
Input	PIRQ0#				EPMI1#		
Input	PIRQ1#						IRQ0
Bidirectional	PIRQ2#	PIRQ2#	LDEV#	PIRQ2#	GPCS0#		
Bidirectional	PIRQ3#	PIRQ3#	LRDY#	PIRQ3#	GPCS1#	EPMI2#	
Output	DACK0#	DACK0#	EDACK0	EDACK0	DACK5#		GPCS0#
Output	DACK1#	DACK1#	EDACK1	EDACK1	DACK6#		GPCS2#
Output	DACK2#	DACK2#	EDACKEN#	EDACKEN#	GPCS2#		
Output	DACK3#	DACK3#	EDACK2	EDACK2	DACK7#		
Output	MP4	GPCS0#	MP4	MP4			
Output	MP5	DACK5#	LMIO	MP5			
Output	MP6	DACK6#	LWR	MP6			GPCS2#
Output	MP7	DACK7#	LADS#	MP7			
Output	PGNT2#						GPCS1#
Input	PREQ2#						EPMI0#
Output	PPWRL#					PPWRL1	

Note:
1. A whole group is enabled by one bit for PnP (plug and play).
2. GMIRQ is the general purpose IRQ for PnP, it can be routed to any of the 11 ISA IRQs (IRQ[7:3], IRQ[12:9], IRQ14 and IRQ15) or be disabled. Default is disabled.
3. During PnP IRQ14, PIRQ0#, and PIRQ1# are not muxed, and PIRQ2# and PIRQ3# are muxed in order to use GPCS0# and GPCS1#.

Table 3-16 82C558M Operating Voltage Groups

TTL (5.0V)
AD[31:0]
C/BE[3:0]#
DEVSEL#
FRAME#
HREQ/TMOD#
IRDY#
LCLK
MDLE#
OSC
PAR
PEN#
PERR#
PGNT[1:0]#
PGNT2#,GPCS1#
PIRQ0#,EPMI#
PIRQ1#
PIRQ2#,LDEV#,GPCS0#
PIRQ3#,LRDY#,EPMI#, GPCS1#
PLOCK#
PREQ0#
PREQ1#
PREQ2#
SERR#
STOP#
TRDY#

AT (5.0V)
AEN
ATCLK
BALE
DACK0#,EDACK0, DACK5#
DACK1#,EDACK1, DACK6#
DACK2#,EDACKEN#
DACK3#,EDACK2, DACK7#
DACK5#,LM/IO#,MP5
DACK6#,LW/R#, MP6,GPCS2#
DACK7#,LADS#,MP7
DDRQ1
DDRQ0
DREQ0,DREQ0/5, DREQ5
DREQ1,DREQ1/6, DREQ6
DREQ2
DREQ3,DREQ3/7, DREQ7
DREQ5
DREQ6,EPMI#
DREQ7
EOP
IDEEN#
IOCHRDY
IOCHK#
IOCS16#
IOR#
IOW#
IRQ1
IRQ3
IRQ5
IRQ4,(MIRQ4/6)
IRQ6,(MPIRQ0#/1#)

AT (5.0V)
IRQ7
IRQ8#
IRQ9
IRQ10,(MIRQ10/12)
IRQ11
IRQ12,(MPIRQ2#/3#)
IRQ14/DINT0
IRQ15/DINT1
MEMCS16#
MEMR#
MEMW#
MP4,GPCS0#,PPWRL2
PPWRL#,PPWRL1
REFRESH#
ROMCS#/KBDCS#
RTCAS
RTCRD#
RTCWR#
SA8/DCS11#
SA7/DDACK0#
SA6/DIOW#
SA5/DIOR#
SA4/DCS03#
SA3/DCS01#
SA[2:0]/DA[2:0]
SBHE#
SD[7:0]/PPWR[7:0]/ DD[7:0]
SD[15:8]/DD[15:8]
SPKR
SMEMR#
SMEMW#
XDIR/PCNTRL
ZEROWS#,PREQ3#
32KHZ,PGNT3#

DRAM (5.0V)
MD[63:32]

CPU/Cache (3.3V)
AHOLD
HLDA
INIT
INTR
FERR#
IGERR#
LMEM#
NMI/LINT1
RESET
SMI#
STPCLK#

4.0 Functional Description

4.1 Reset Logic

The PWRGD input to the 82C557M 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 82C558M 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 82C557M Clocks

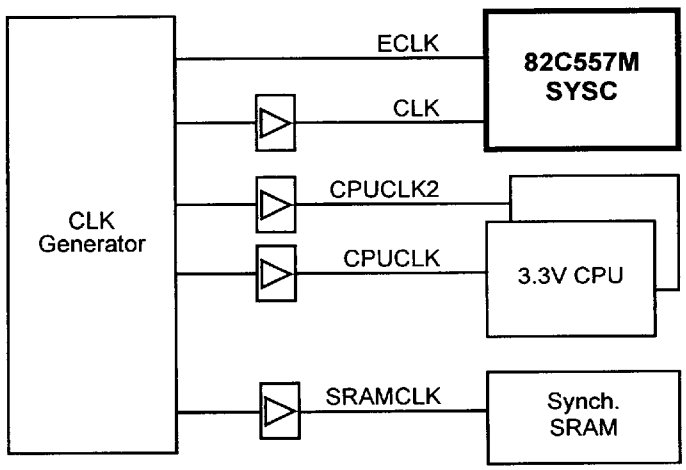
The 82C557M uses two high frequency clock inputs, CLK and ECLK.

The clock signals that go to the CPU and the 82C557M's CLK inputs are required to be in the same phase and have minimum skew between them. The skew between the CLK input to the 82C557M and the CLK input to the CPU should not exceed 2ns. The 82C557M CLK is a single phase clock which is used to sample all host CPU synchronous signals and for clocking the 82C557M'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 82C557M's CLK delay requirements.

Figure 4-1 shows the relationship between CPUCLK, CLK, and ECLK and the typical CPU and 82C557M clock distribution circuit.

Figure 4-1 CPU and 82C557M Clock Distribution



4.2.2 PCI and VL Bus Clocks

The 82C557M and the 82C558M require LCLK for the PCI and VL bus interface. The phase and frequency of the LCLK input to the 82C557M and 82C558M 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 82C558M 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)

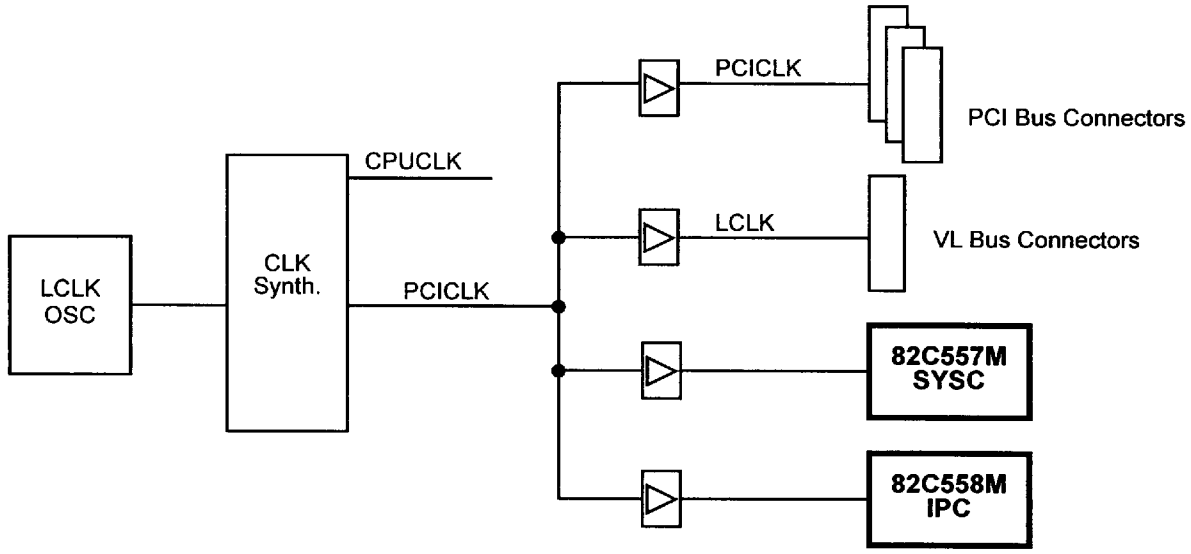
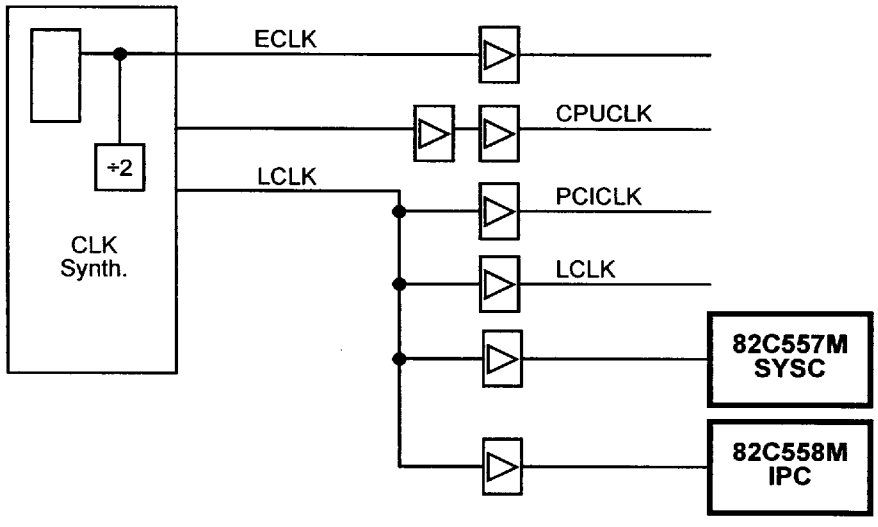


Figure 4-3 Clock Distribution Method for VL Bus and PCI Connectors (Sync. PCI and VL Bus)



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 82C557M'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-M Chipset fully supports the 64-bit wide data path for the CPU burst read and burst write cycles. The 82C557M'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 82C557M 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-M Chipset supports the Cyrix linear burst mode. Index 17h[0] in the 82C557M 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 82C557M 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-M 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 82C557M 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 82C557M 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)

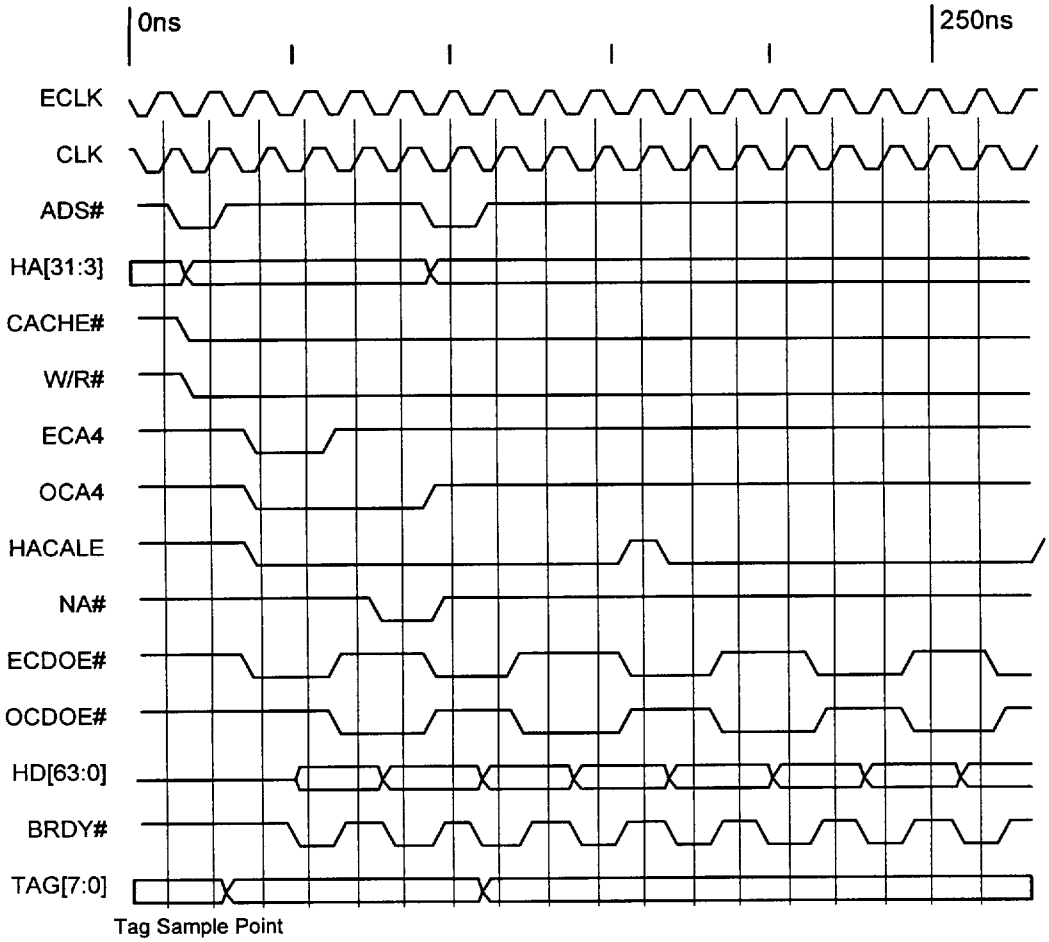


Figure 4-5 L2 Cache Read Hit Cycle - Sync. SRAMs

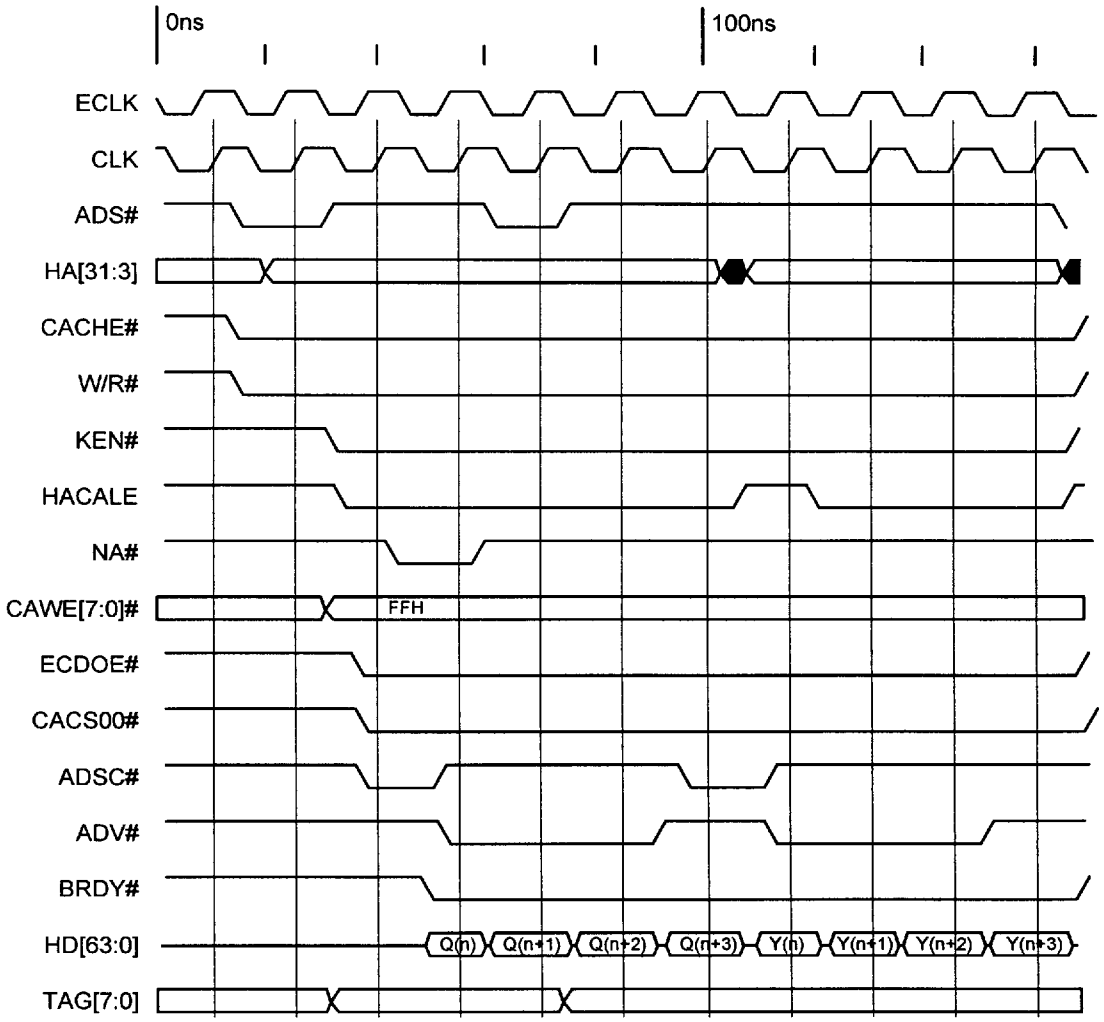
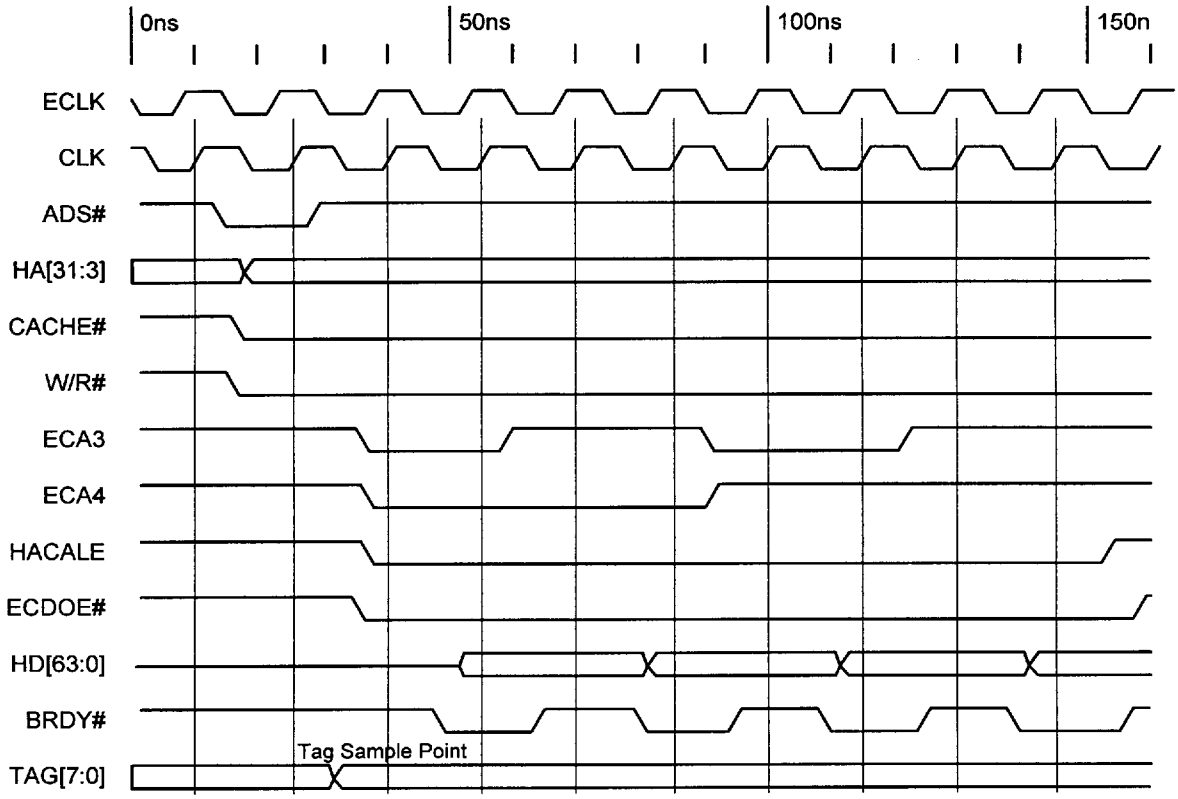


Figure 4-6 L2 Cache Read Hit Cycle Async. SRAMs (Single Bank)



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)

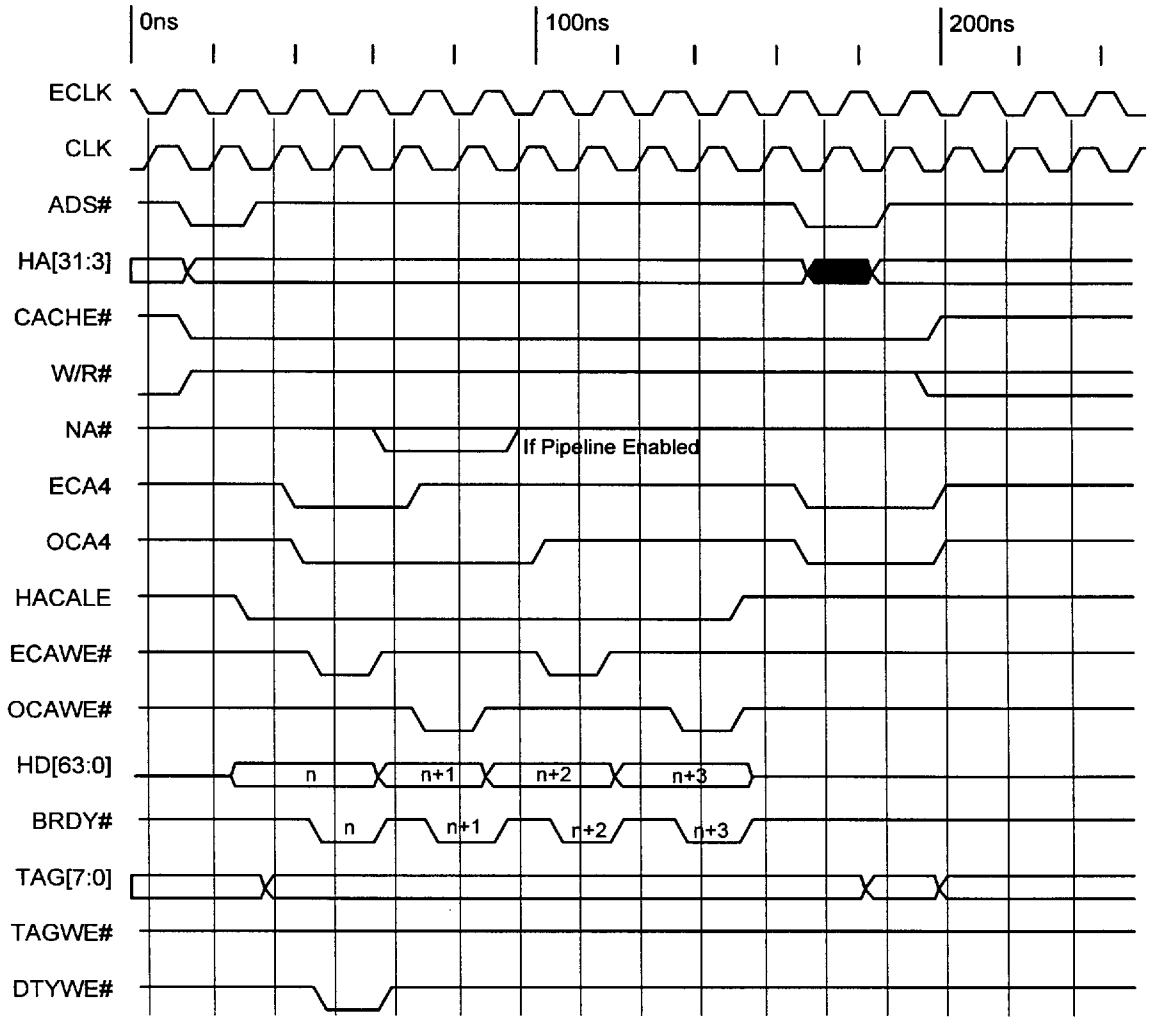


Figure 4-8 Write Hit Burst Cycle for Write-Back Mode (Single Bank) - Async. SRAM

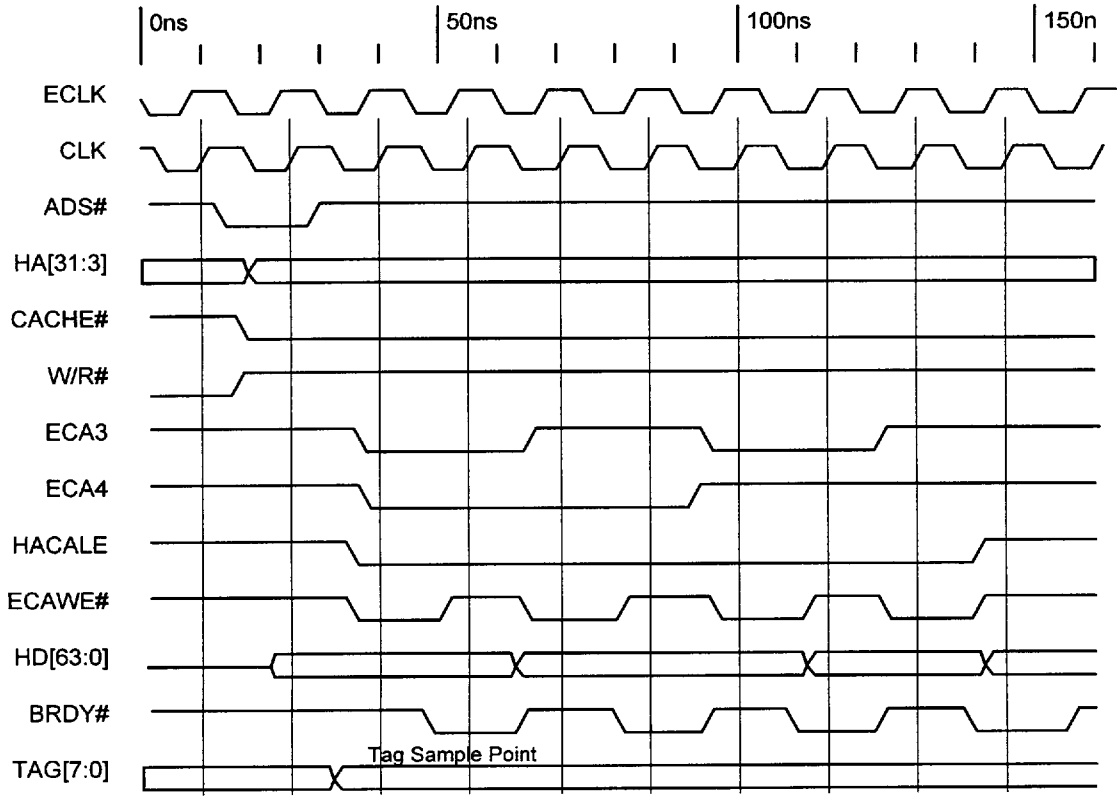
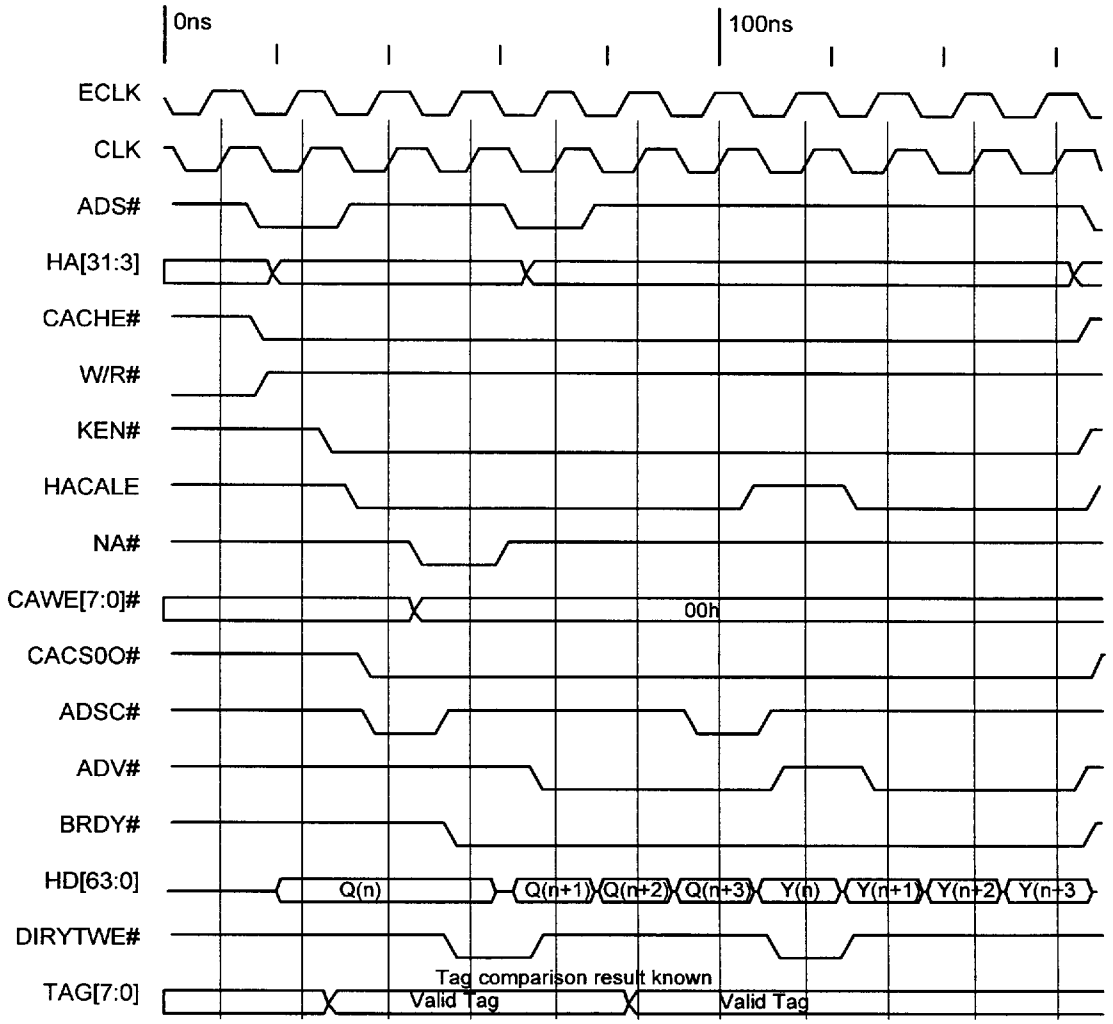


Figure 4-9 Write Hit Burst Cycle for Write-Back Mode - Sync. SRAM



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.

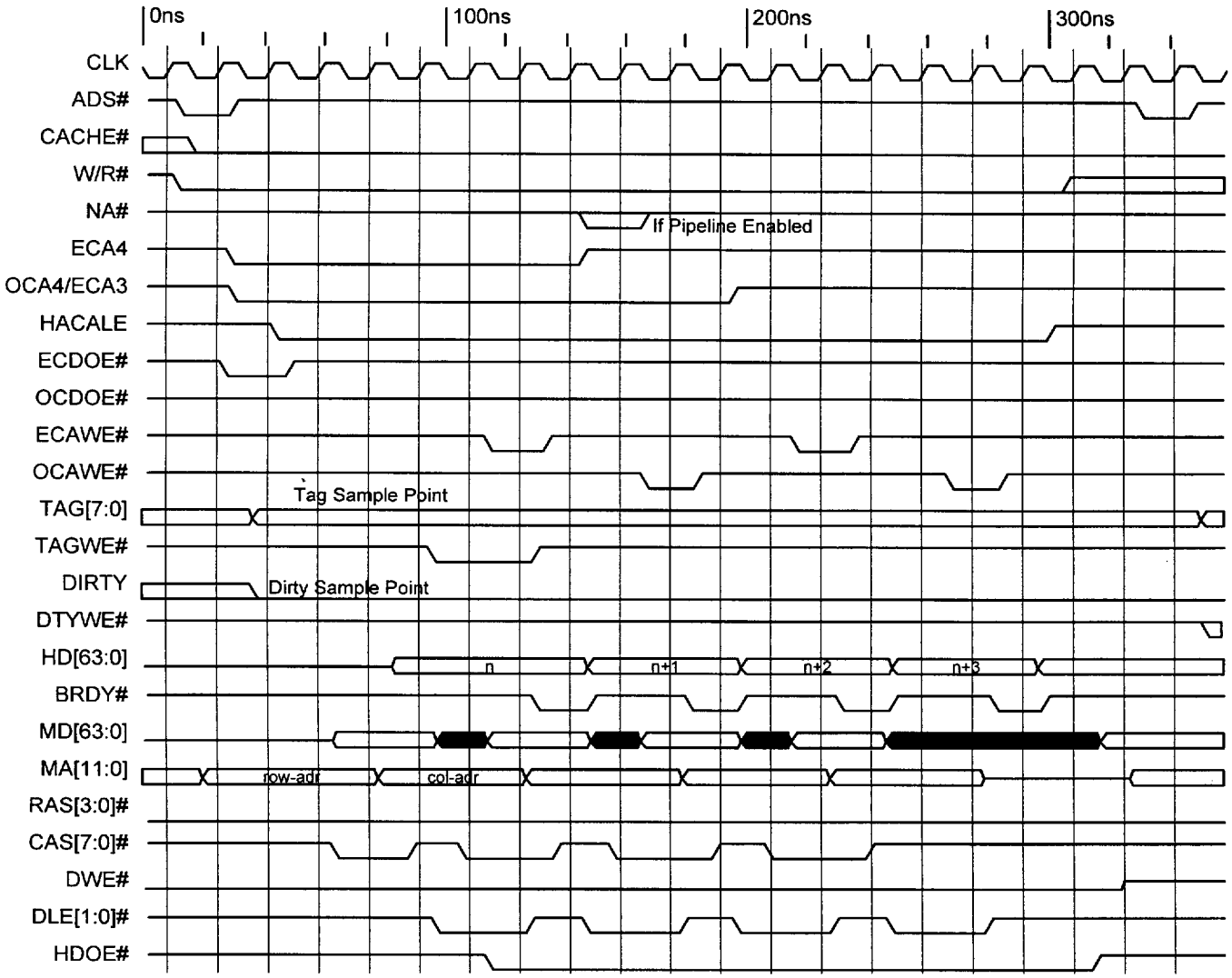
The cache line for this case has been modified and only the L1 and L2 cache have the updated copy of the data. Before this line is overwritten in the cache, the modified line must first be written to the system memory by performing a castout cycle. After the completion of the castout cycle, a linefill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

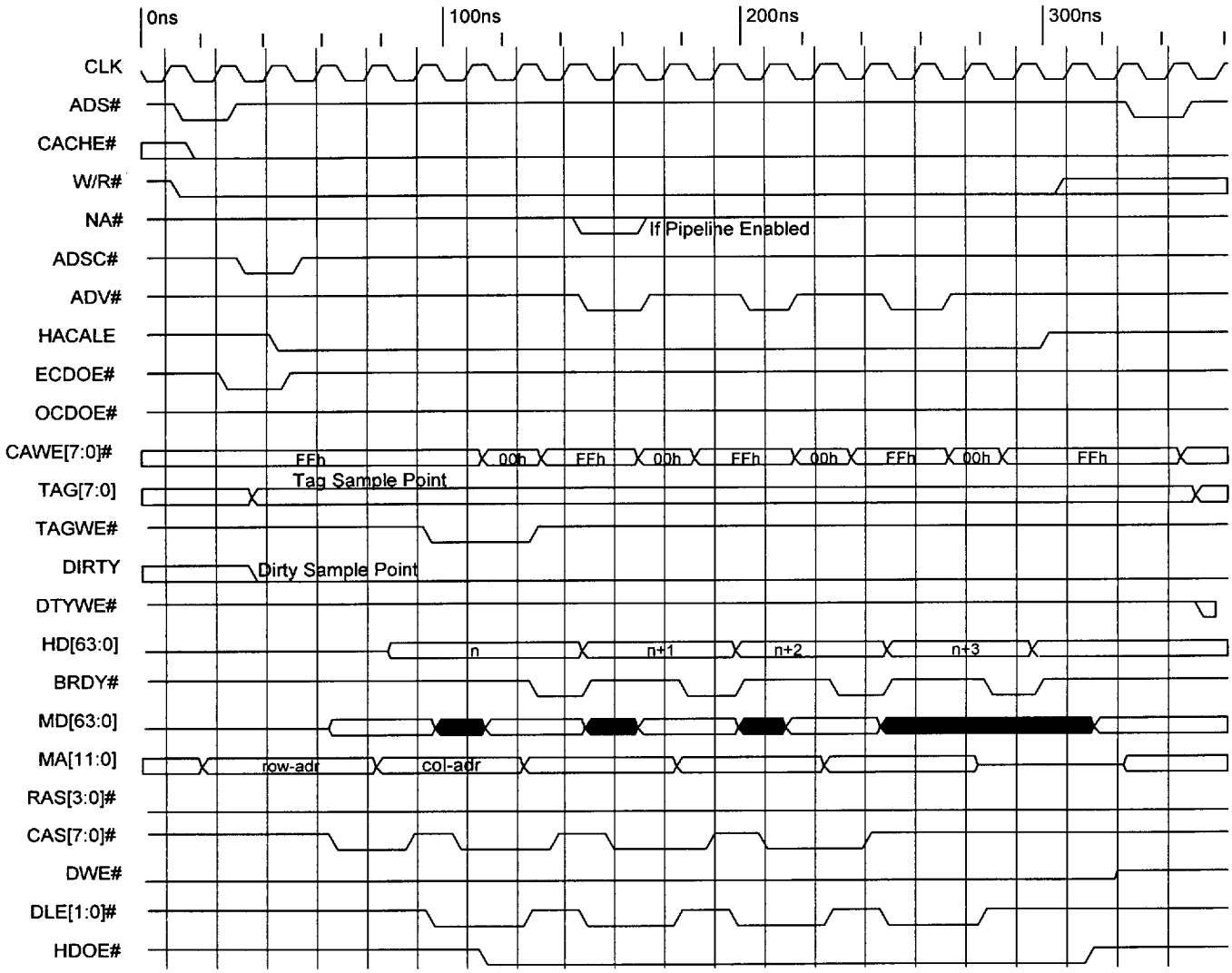


Figure 4-12 L2 Cache Read Miss Dirty Cycle - Async. SRAMs

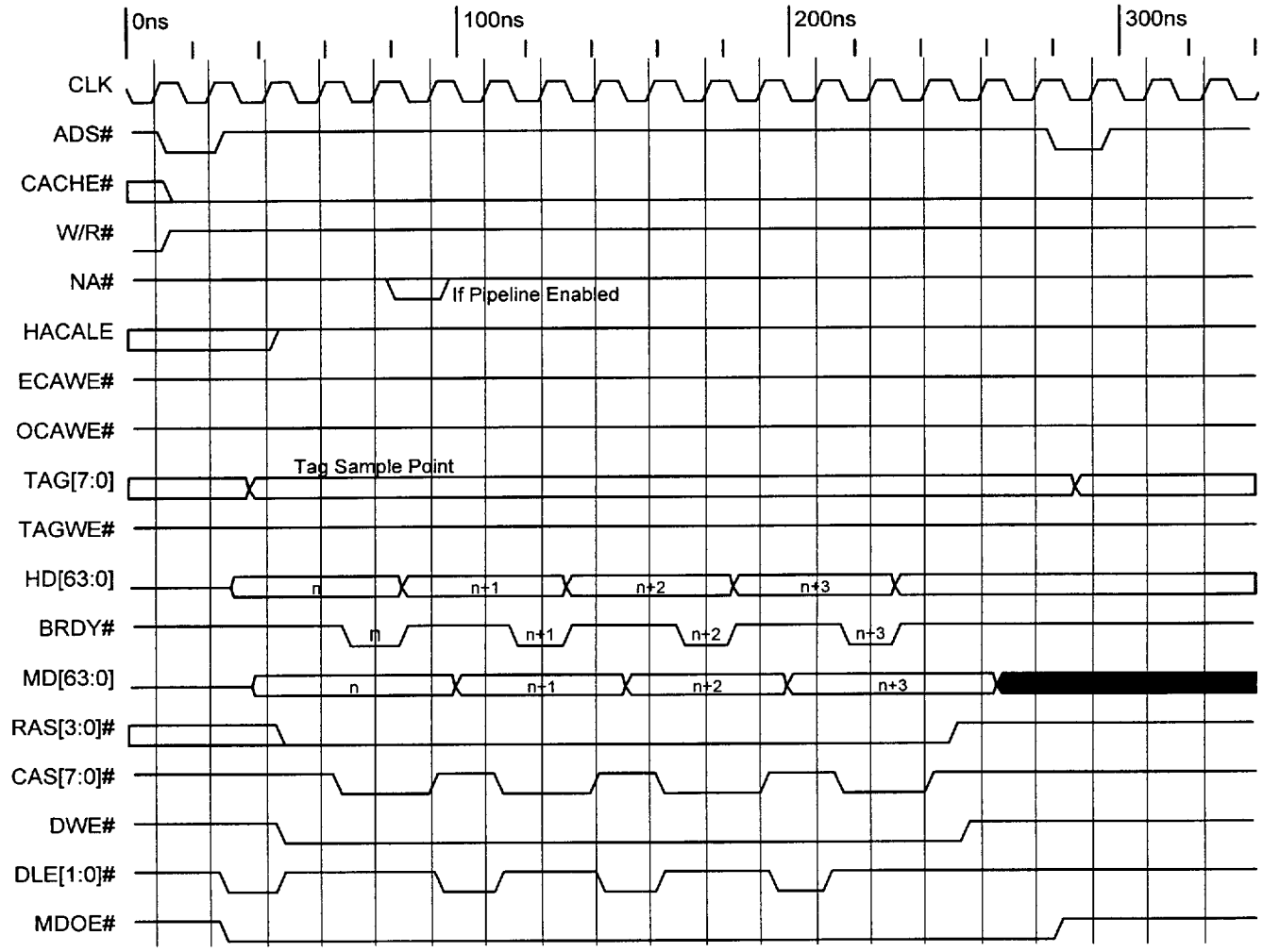
4.3.3.4 L2 Cache Write Miss

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.

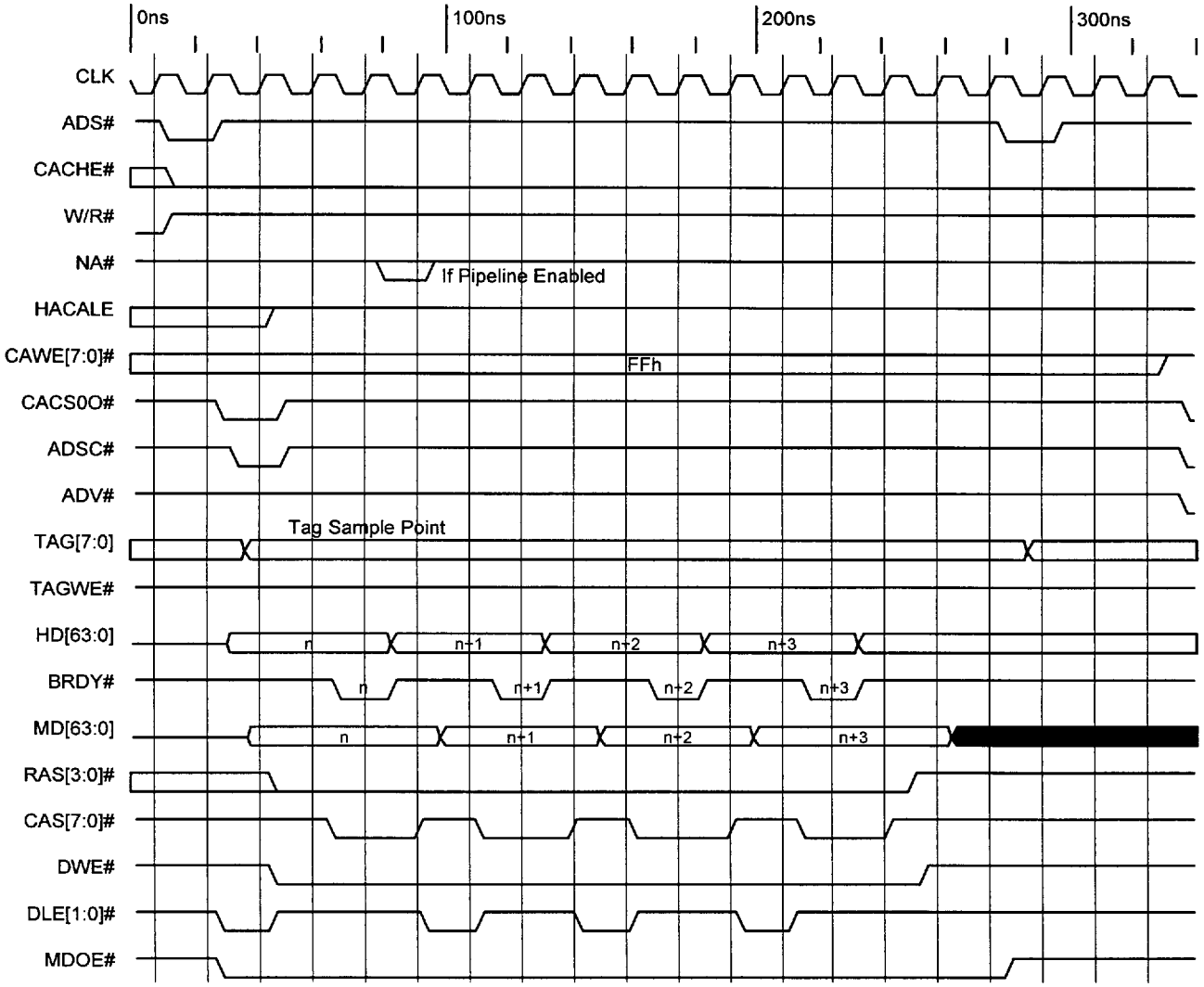
Figures 4-13 and 4-14 show L2 cache write miss cycles.

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-M Chipset can be chosen: write-back, write-through, and adaptive write-back, by programming bits [5:4] of the 82C557M'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.

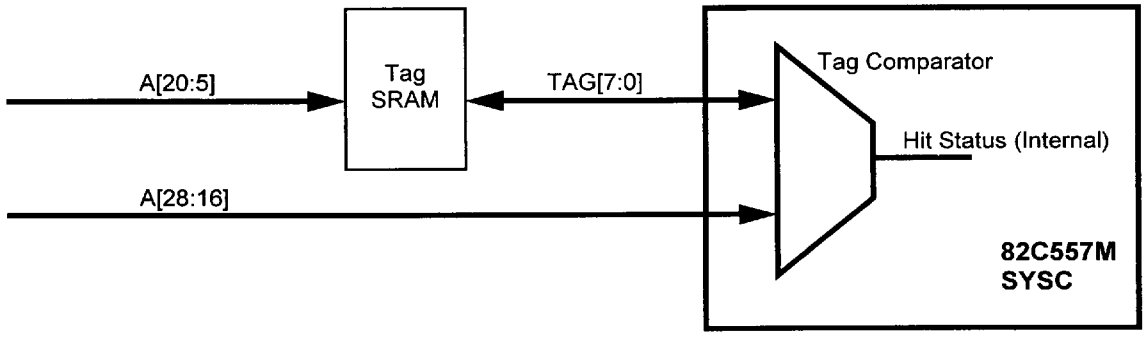
4.3.3.6 Tag Compare Table

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 82C557M's 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-M 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 82C557M is used to update the tag RAM and the DIRYTWE# signal from the 82C557M 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 82C557M 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 82C557M or it can be tied to GND. The DIRYTI signal of the 82C557M 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 82C557M 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 82C557M 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

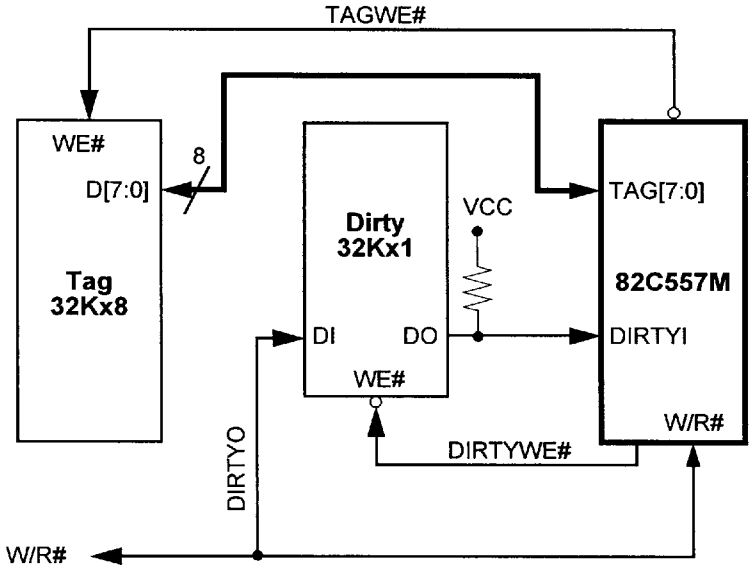


Figure 4-17 32Kx9 Combined Tag/Dirty RAM Implementation

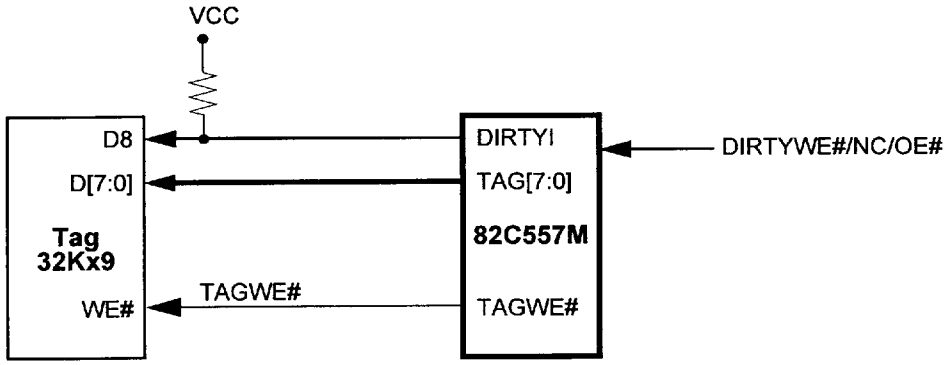


Figure 4-18 32Kx8 Combined Tag/Dirty RAM Implementation

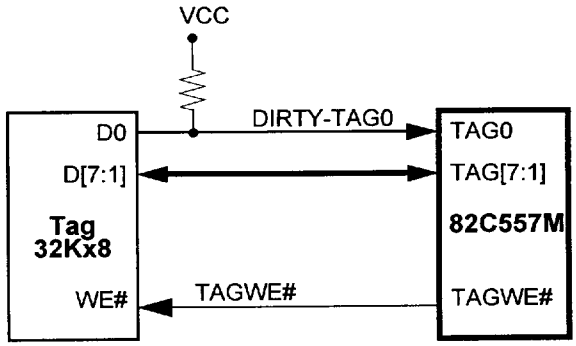
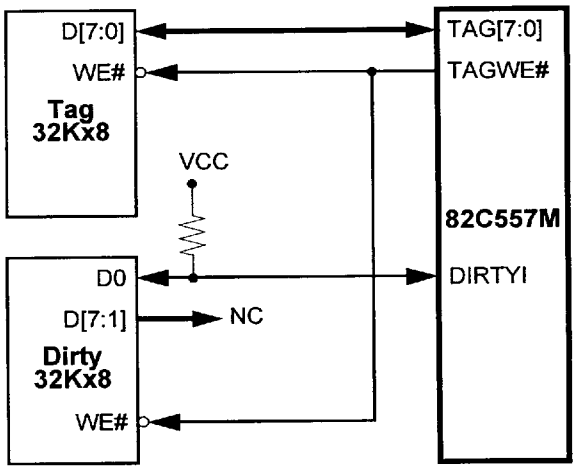


Figure 4-19 32Kx8 and 32Kx8 Combined Tag/Dirty RAM Implementation (Separate Devices)



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 82C557M's 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 82C557M's 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 82C557M 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-M 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 82C557M will assert AHOLD. The address will flow from the master to the CPU bus and the 82C557M will assert EADS# for one CPU clock. If the CPU does not respond with the assertion of HITM#, the 82C557M will complete the cycle from the L2 cache or the system memory. If HITM# was asserted, the 82C557M 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 82C557M 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 82C557M 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)

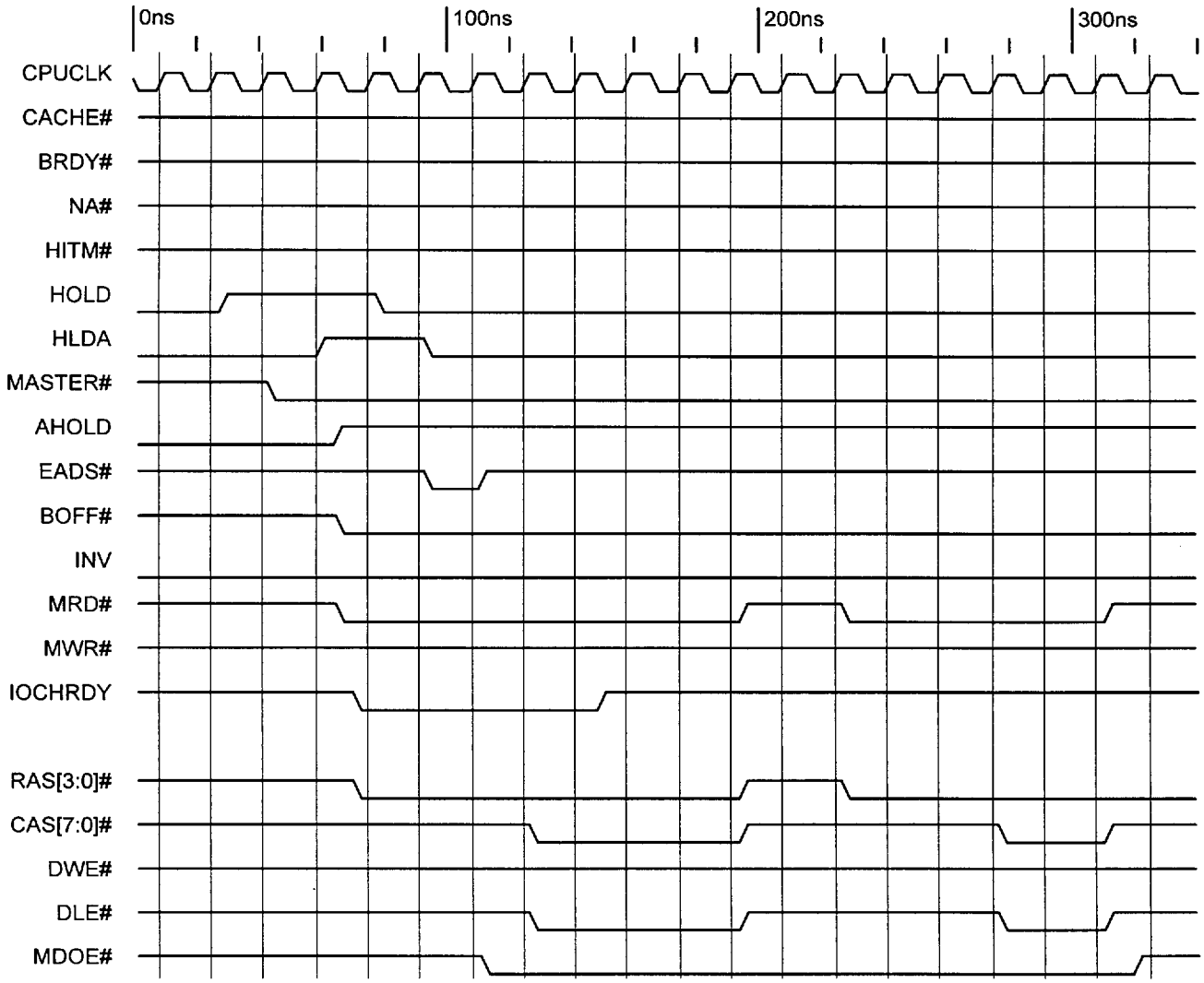


Figure 4-21 AT DMA/Master Read (L1 cache with modified line)

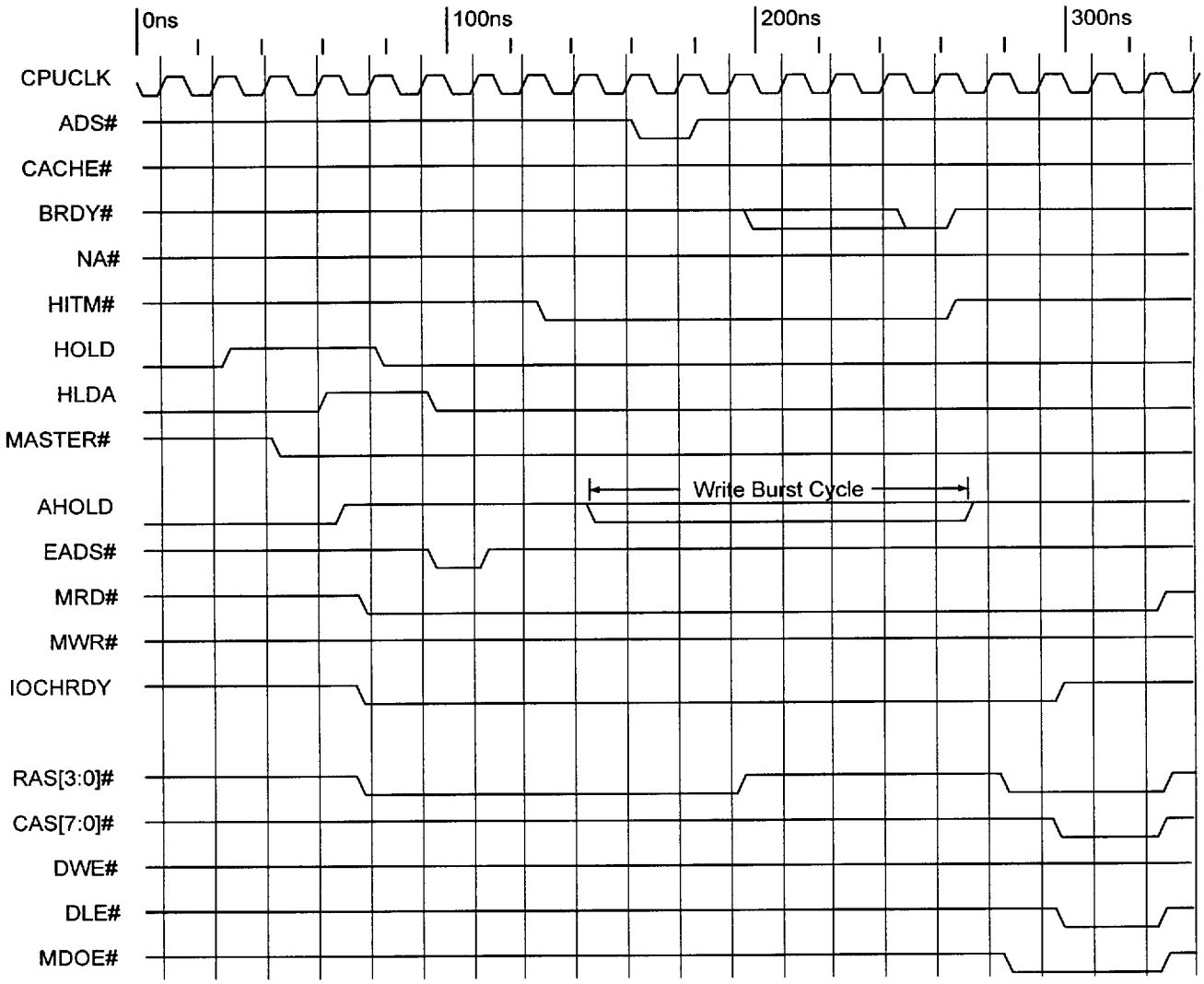
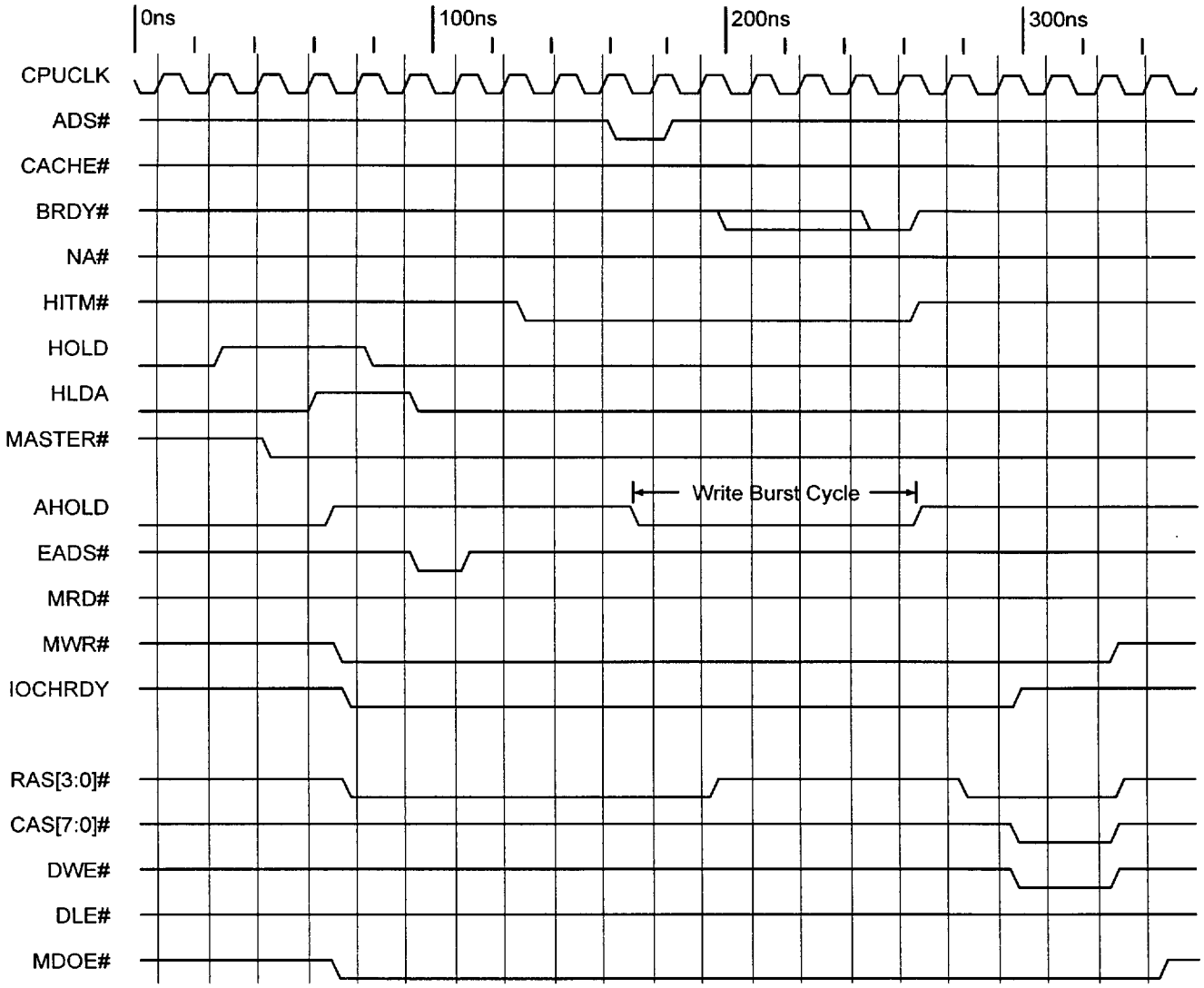


Figure 4-22 AT DMA/Master Write (L1 cache with modified line)



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-M 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 82C557M's 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 82C557M'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

	System DRAM		System BIOS		Video BIOS		Write Enabled Shadow RAM		Write Protected Shadow RAM	
Method	**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-M 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-M 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-M 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-M 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 82C557M. 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 82C557M 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 82C557M are not required and should be no connects. The ADS# input of the 82C557M 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-M 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 82C557M OE# control.
*** L2 needs "castout" dirty line before master access.

4.4 DRAM Controller

The Viper-M 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-M 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 EDO Support

The Viper-M Chipset offers the user the capability to use EDO DRAMs in the system. EDO devices are very similar to devices that incorporate fast page mode accesses. However, the use of EDO DRAMs boosts the system performance by a minimum of 30% over conventional fast page mode DRAMs. This boost in performance stems from the different way in which the memory bus is controlled. In conventional fast page mode DRAMs the memory bus is turned on by the falling edge of CAS# and turns the bus off (High-Z) when CAS# returns to high. The fast page mode DRAMs only guarantee data to be valid for 5ns (which is typically too brief for systems operating at full speed). To compensate, system designers hold CAS# low for an extended period until the data can be read of the bus. In contrast, EDO devices turn on the memory bus when CAS# falls low but do not turn off the bus when CAS# returns to high. Instead, the data remains valid until the next falling edge of CAS#. Because the data remains valid until the falling edge of the next CAS#, the transfer of memory data to the latch in the memory controller can be overlapped with the next column precharge.This extra time that the data remains valid resolves the system problem described above. The extended data time allows the system to run with a minimum CAS# low time. This increases the system performance by decreasing the page access cycle time. Control of the memory bus can be obtained by the OE# and CAS# signals.

The Viper-M Chipset allows the user to populate the system with up to six banks of EDO DRAMs. Individual bits in the register whose Index is 1Ch in the 82C557M SYSC of the chipset needs to be set to a "1" for each bank that uses EDO DRAMs. Timing can be programmed to achieve a 6-2-2-2 read cycle at 50MHz when the user uses EDO DRAMs with the Viper-M Chipset.

Users of the Viper-M Chipset are also provided with the flexibility to mix and match EDO DRAM SIMMs and conventional fast page mode DRAMs among the different banks.

As an example, the user could use EDO DRAMs in Banks 0 and 2 and fast page mode DRAMs in the other banks. There is no restriction placed upon the user in terms of which bank(s) could contain EDO SIMMs and which should contain fast page mode DRAMs.

Care must be taken by the user to ensure that those SIMMs that make-up the 64-bit data path to DRAM are all EDO DRAMs or fast page mode DRAMs.

The DRAM controller is capable of interfacing to EDO DRAMs that are EO# controlled as well as WE#.

In a system, all banks that have been populated with EDO DRAMs will have the same DRAM timings. Likewise, all banks that are populated by fast page mode DRAMs will have the same DRAM timings.

4.4.2 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.

4.4.2.1 Number of DRAM banks

The Viper-M 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.

4.4.2.2 DRAM Configurations

The Viper-M 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.

4.4.2.3 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	1 to 2
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

4.4.2.4 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.

4.4.3 DRAM Cycles

The fastest possible burst read is 6-2-2-2 which means the first quad-word is received in six clocks and the next three quad-words are received after two 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.

4.4.3.1 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 82C557M 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 82C557M to the 82C556M. Data from the DRAM is latched by the 82C556M on the rising edge of CLK or DLE (when using EDO DRAMs or fast page mode DRAMs for CPU reads from DRAM). 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 82C556M 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-M 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 82C556M 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 7-2-2-2.

4.4.3.2 DRAM Write Cycle

Posted write to the DRAM improves the write cycle timing relative to the CPU and allows the Viper-M Chipset to perform an independent write burst cycle to DRAM without holding the CPU. The Viper-M 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 82C556M is controlled by the DLE[1:0]# signals from the 82C557M. 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.

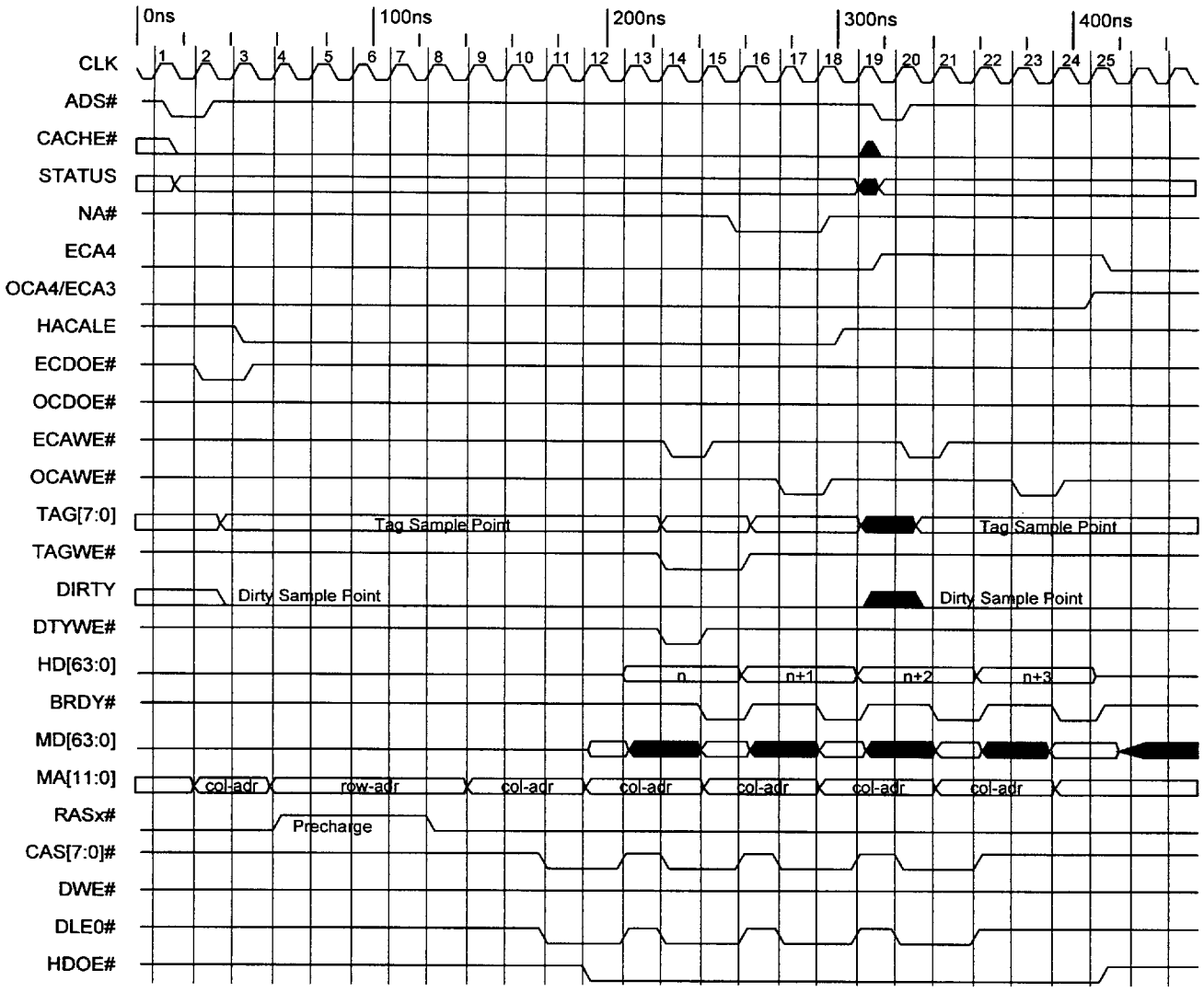
Single level posted write cycles are employed to achieve a 4-3-3-3 burst at 66MHz. The data from the CPU is latched in the 82C556M's write buffer until CAS goes active one T-state after the first T2 (on a page hit). This provides a fast write mechanism and two wait state writes are maintained for the 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 Bus Speed (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	7 cycles	10 cycles	10 + precharge	5 clocks	2-2-2	60
80	yes	8 cycles	12 cycles	12 + precharge	5 clocks	4-4-4	60

CPU Bus Speed (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.

4.4.3.3 DRAM Parity Generation/Detection Logic

During local DRAM write cycles, the 82C556M 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 82C556M will assert the MPERR# signal to the 82C557M. This will be qualified by an internal PEN# signal in the 82C557M, if parity is enabled. The 82C557M then triggers a PCI configuration cycle to the 82C558M. On detecting this configuration cycle, the 82C558M will assert an NMI interrupt to the CPU.

4.4.3.4 DRAM Refresh Logic

The 82C556M 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 82C558M) asserts the RFSH# and MEMR# commands and outputs the refresh address during AT bus refresh cycles.

The 82C557M 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 82C558M 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.

4.4.4 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.

4.4.5 DRAM DMA/Master Cycles

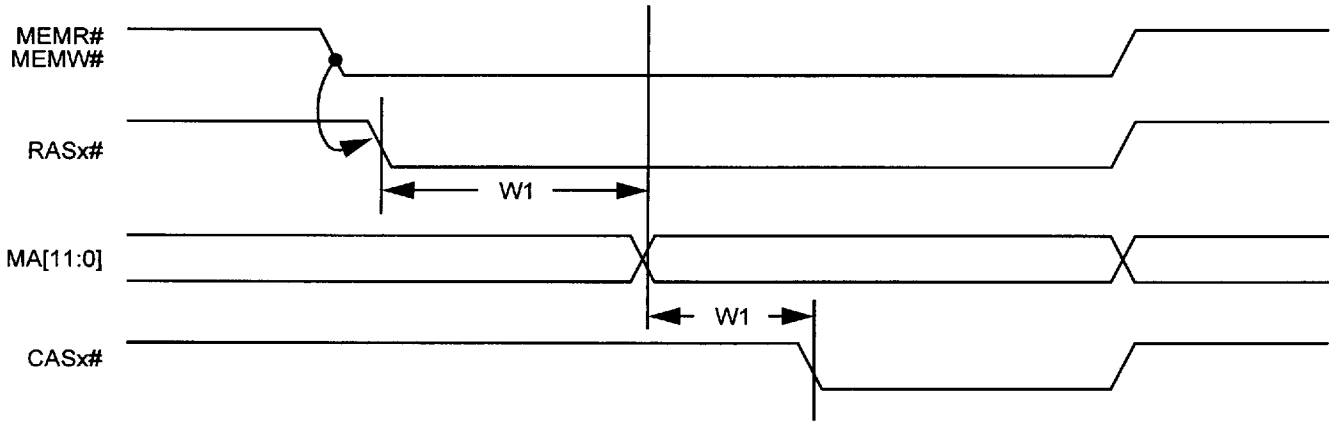
For DMA and master cycles, the DRAM controller operates such that the MEMR# and MEMW# signals generate RASx# synchronously. The generation of the DRAM column address is then synchronized with LCLK. The synchronization can be programmed to be 0.5 to 1.5 LCLKs and 1 to 2 LCLKs. The generation of CASx# 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-M 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-M 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 82C557M (SYSC), and the PCI data bus buffering is done within the 82C558M (IPC). The 82C558M 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 at Address Offset 43h-42h in the 82C558M.

4.5.1.1 System Memory Access

The PCI master asserts FRAME# and puts out the address on the AD[31:0] bus. The 82C557M decodes that address and asserts LMEM# to the 82C558M if the access is to the system memory area. The 82C558M then provides the data path to the PCI master to access system memory. If the access is to the system memory space, then the 82C557M 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 82C556M performs the data steering and latching based on the control information received from the 82C557M, 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 3-5 describes the state of the control signals from the 82C557M to the 82C556M for all cycles.

For a low order DRAM read, the DRAM puts out the data on the MD[31:0]# lines. The 82C556M latches the data, inverts it and puts it out onto the MD[63:32] lines and drives it out to the 82C558M. The 82C558M 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 82C558M latches the data and puts it out on the MD[63:32] lines. The 82C556M 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 82C556M latches the data onto the MD[63:32] lines and drives it out to the 82C558M. The 82C558M 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 82C558M latches the data and puts it out on the MD[63:32] lines. The 82C556M 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 82C558M and the 82C556M does not have to perform any latching or steering of data. The 82C558M 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 82C558M latches the data and this puts it out on the MD[63:32] lines. The 82C556M 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 82C556M latches the data onto the MD[63:32] lines and drives it out to the 82C558M. The 82C558M 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 82C558M latches the data and puts it out on the MD[63:32] lines. The 82C556M 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 82C557M does not assert LMEM# to the 82C558M. The 82C558M 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 82C558M 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 82C558M 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 82C558M 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 82C558M 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 82C558M 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 82C558M 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 82C557M 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-M 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 82C557M 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 82C557M. The number of MDLE# pulses sent out by the 82C557M 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 82C558M should be set to 1.
- Index 17h[2] of the 82C557M's system control register space should be set to 1.
- Index 0h[4:3] of the 82C557M's 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 82C558M 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 82C558M handles all the data bus conversion and steering logic.

4.6 VL Bus Interface

The Viper-M 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 82C558M and the data path is controlled by buffers in the 82C556M. The 82C557M also does the data bus conversion to interface the 32-bit VL bus to the 64-bit CPU bus. The Viper-M 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 82C557M generates a PCI cycle and the 82C558M a VL cycle. If the Viper-M Chipset has been configured to support VL bus slaves, then the 82C558M 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 82C558M asserts DEVSEL# to the 82C557M 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 82C558M asserts DEVSEL# to the 82C557M. After LRDY# is sampled active, the Viper-M 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

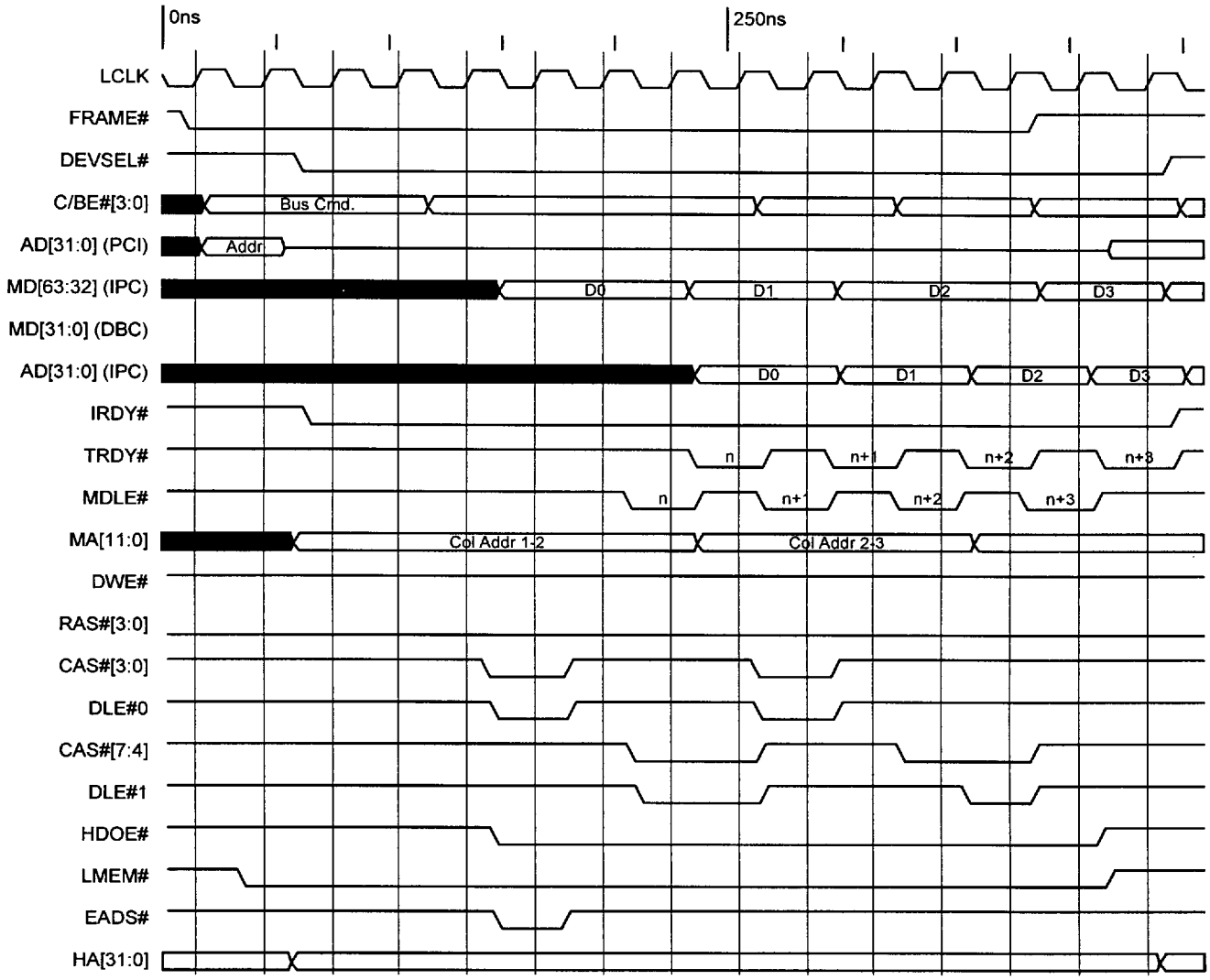


Figure 4-27 CPU Read from PCI, 32 Bits

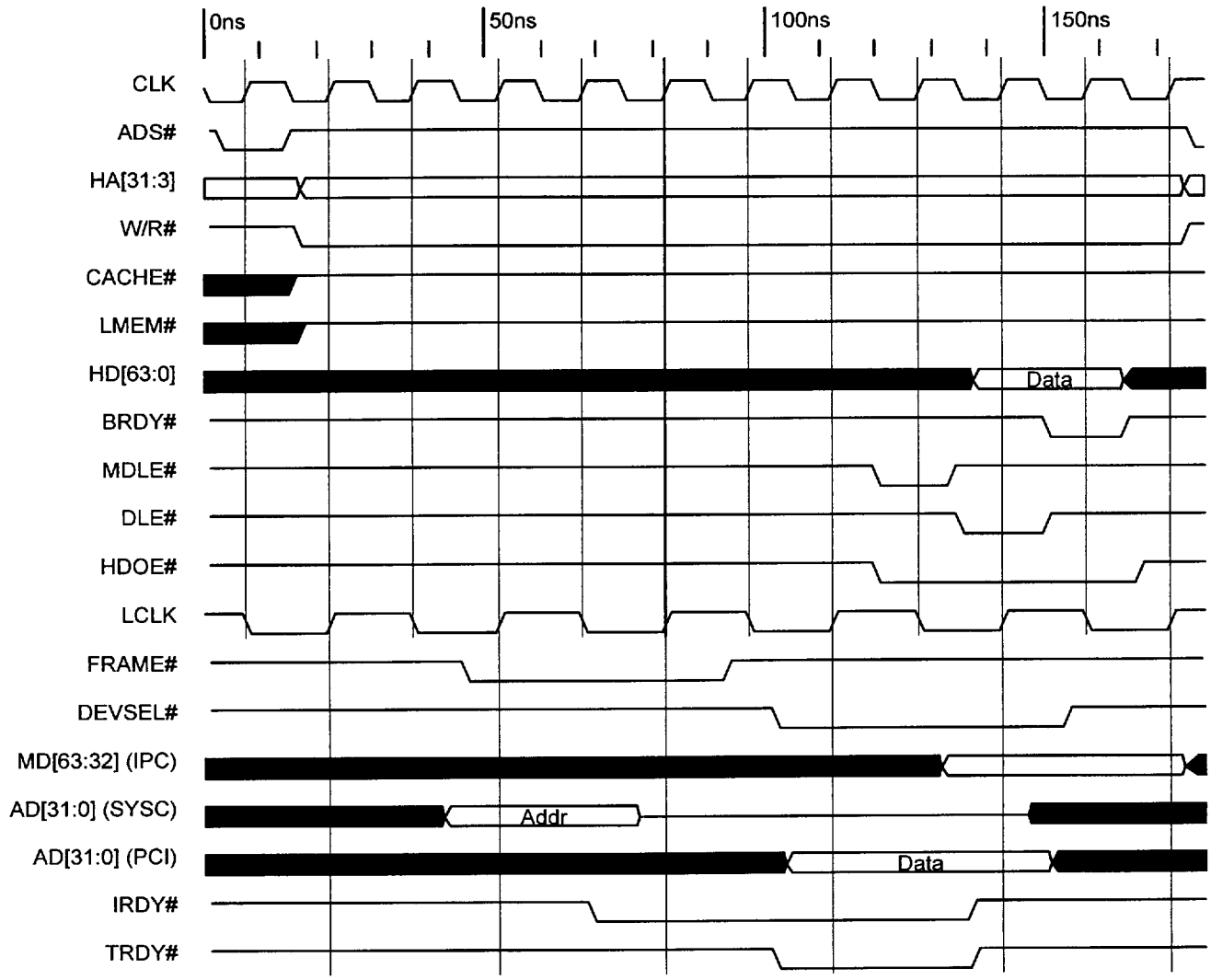


Figure 4-28 CPU Write to PCI, 32 Bits

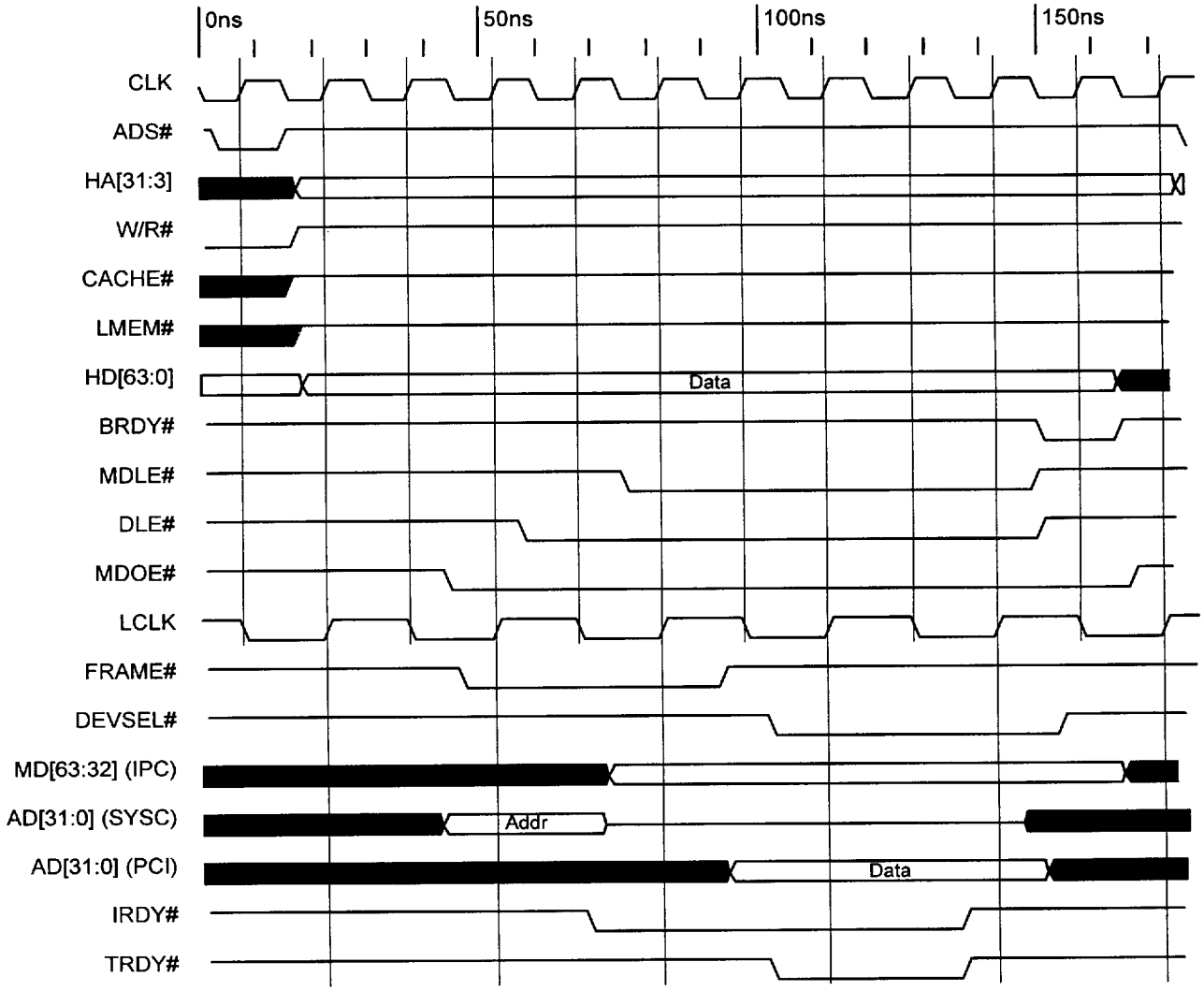


Figure 4-29 CPU Read from PCI, 64 Bits

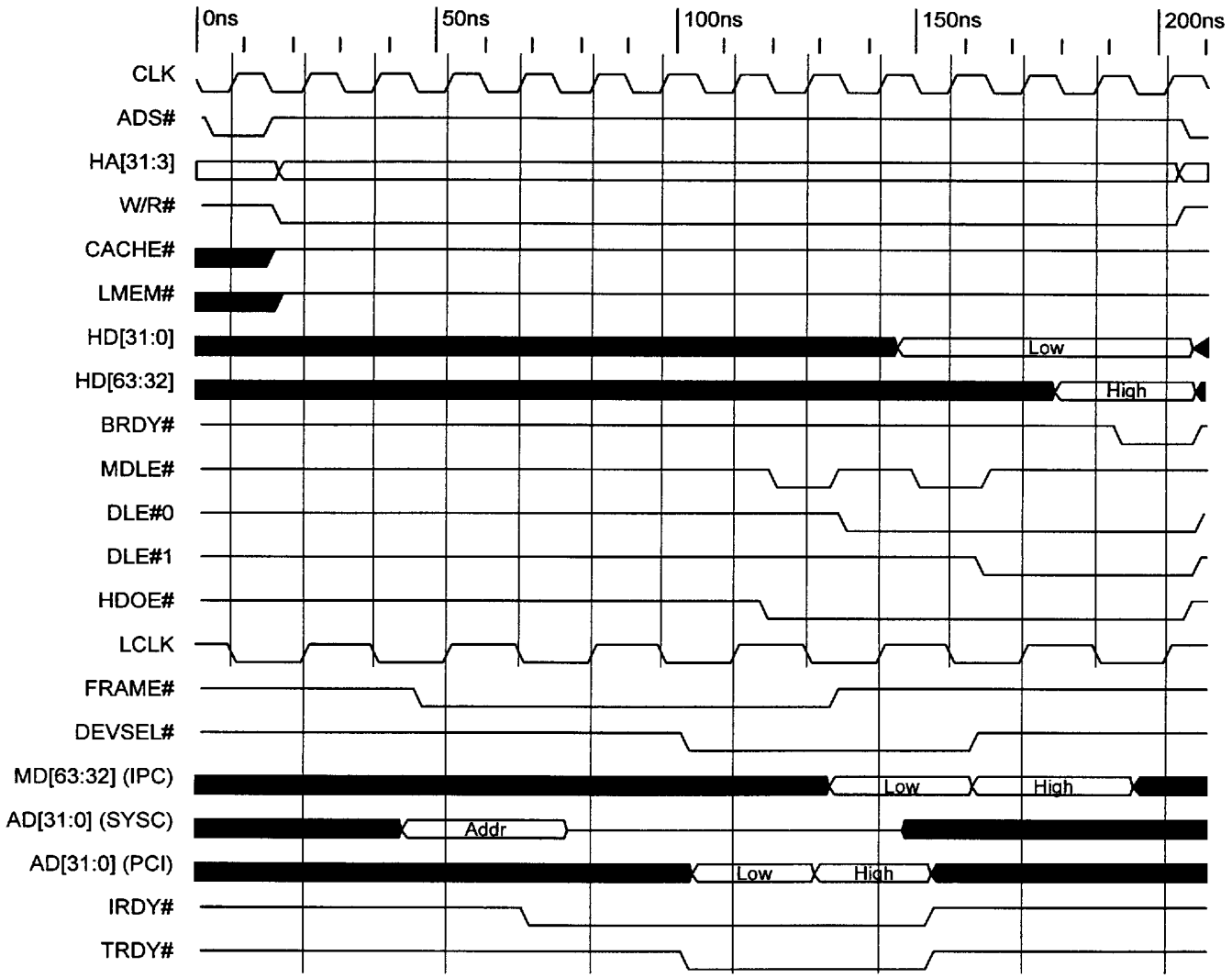


Figure 4-30 CPU Read from VESA Slave

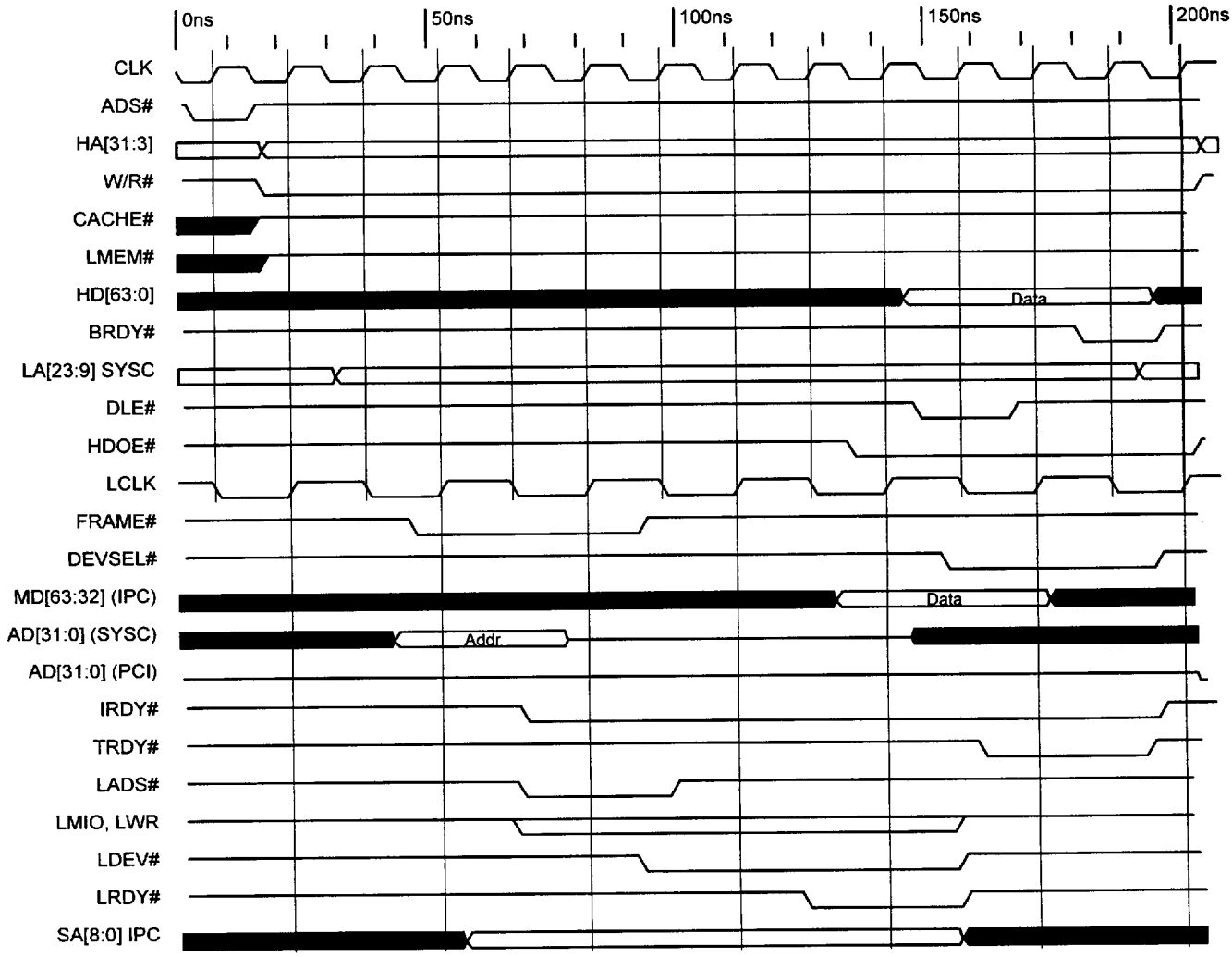


Figure 4-31 CPU Write to VESA Slave

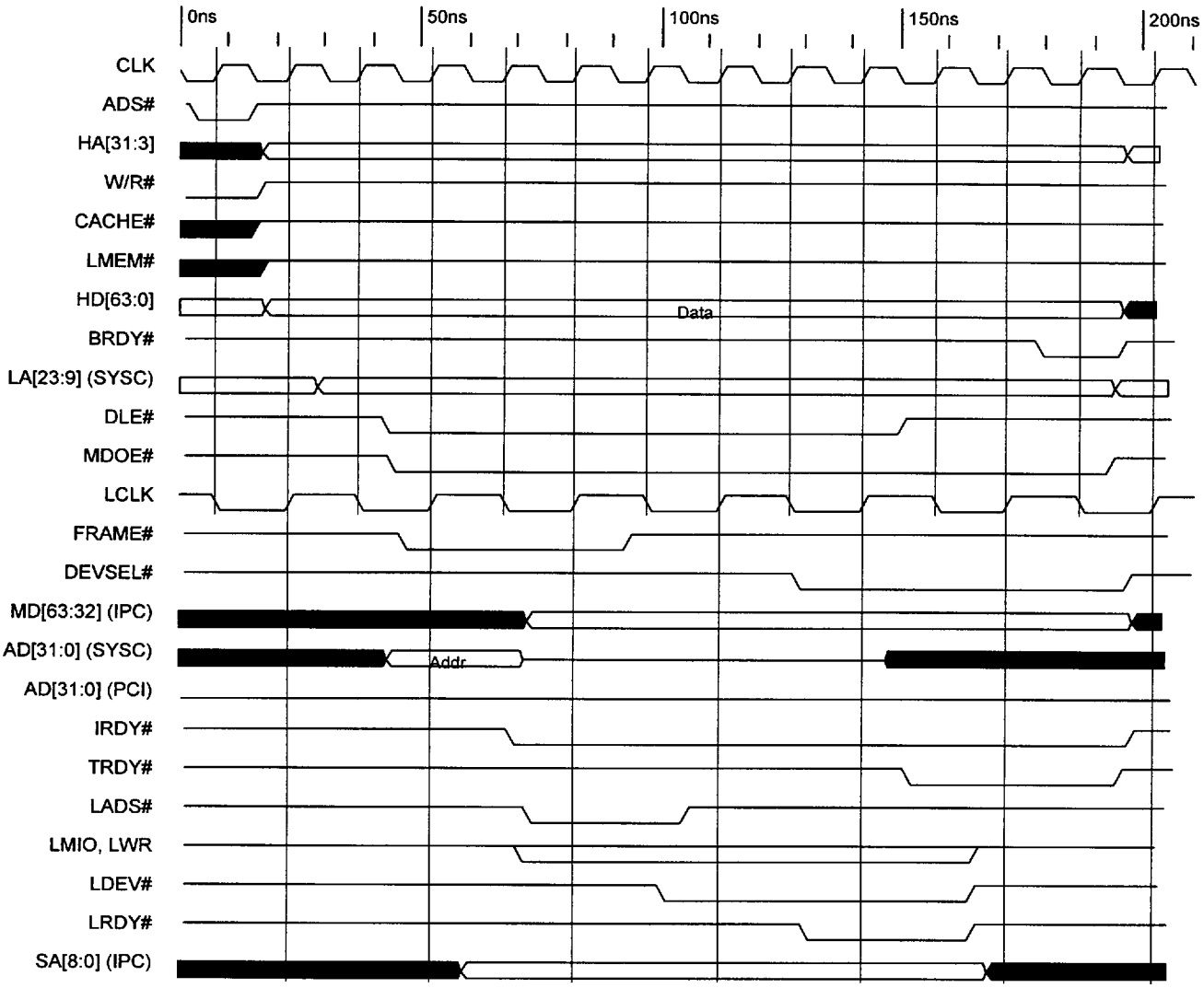


Figure 4-32 ISA Master Read from PCI

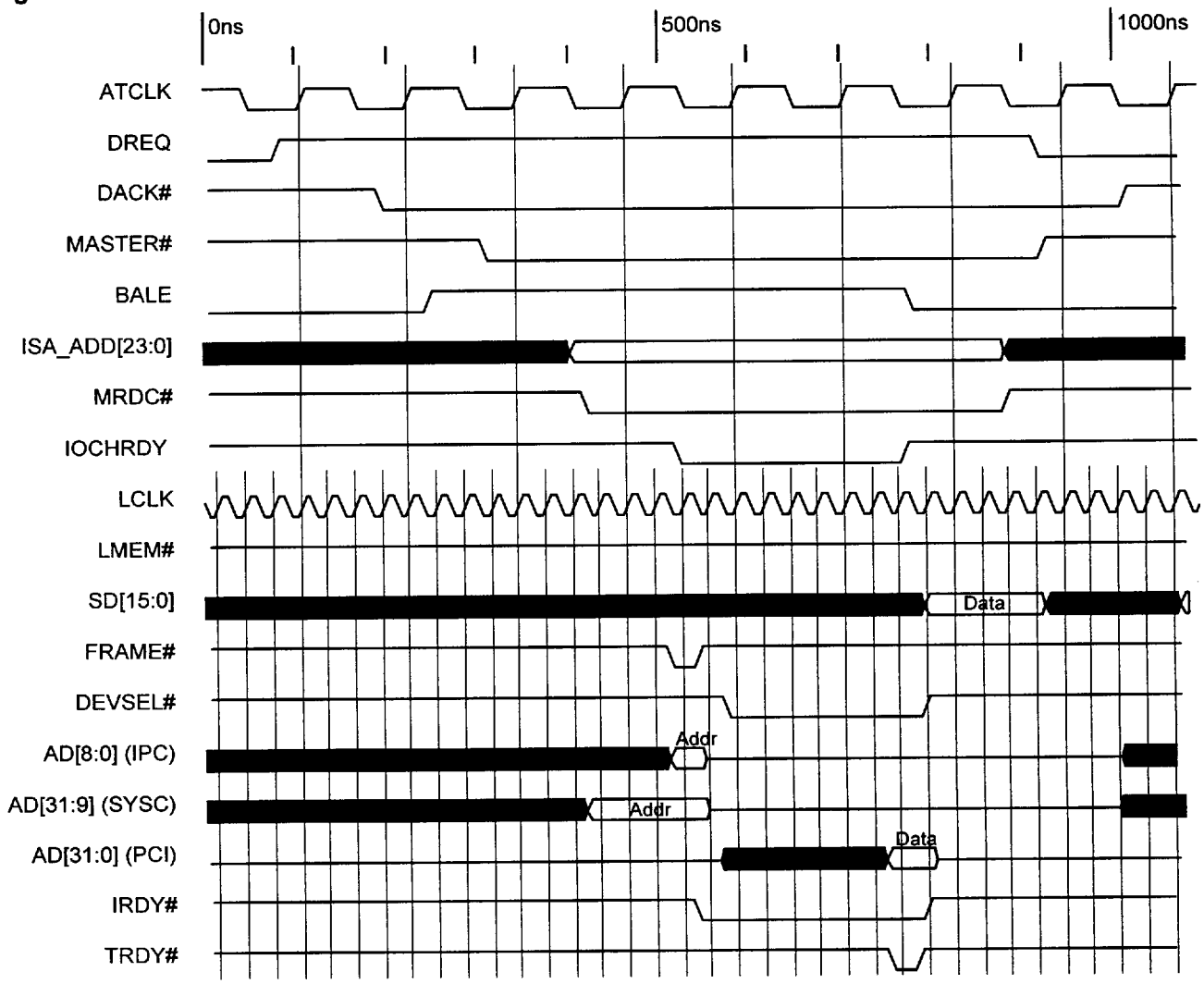


Figure 4-33 ISA Master Write to PCI

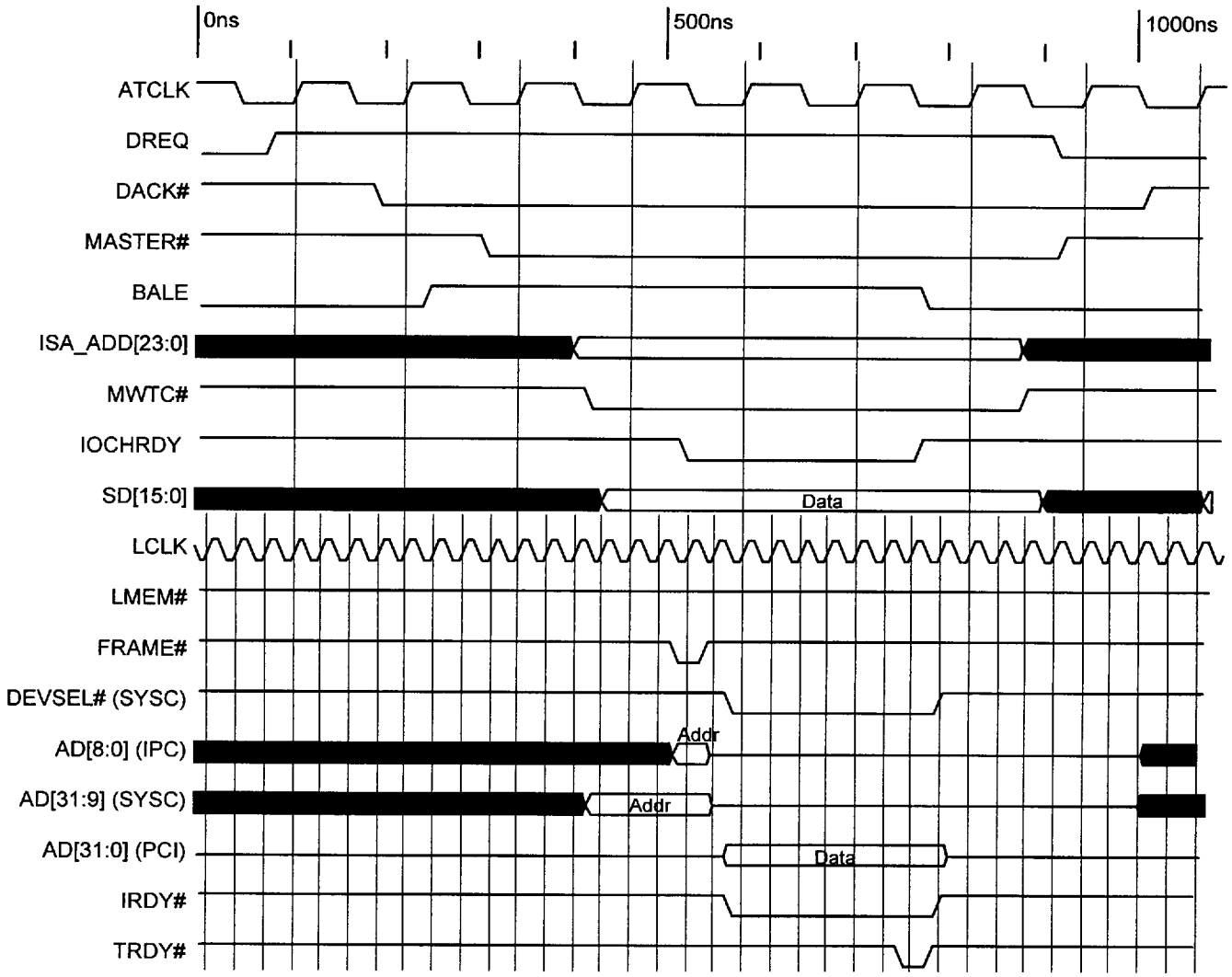


Figure 4-34 ISA Master Read from ISA Slave

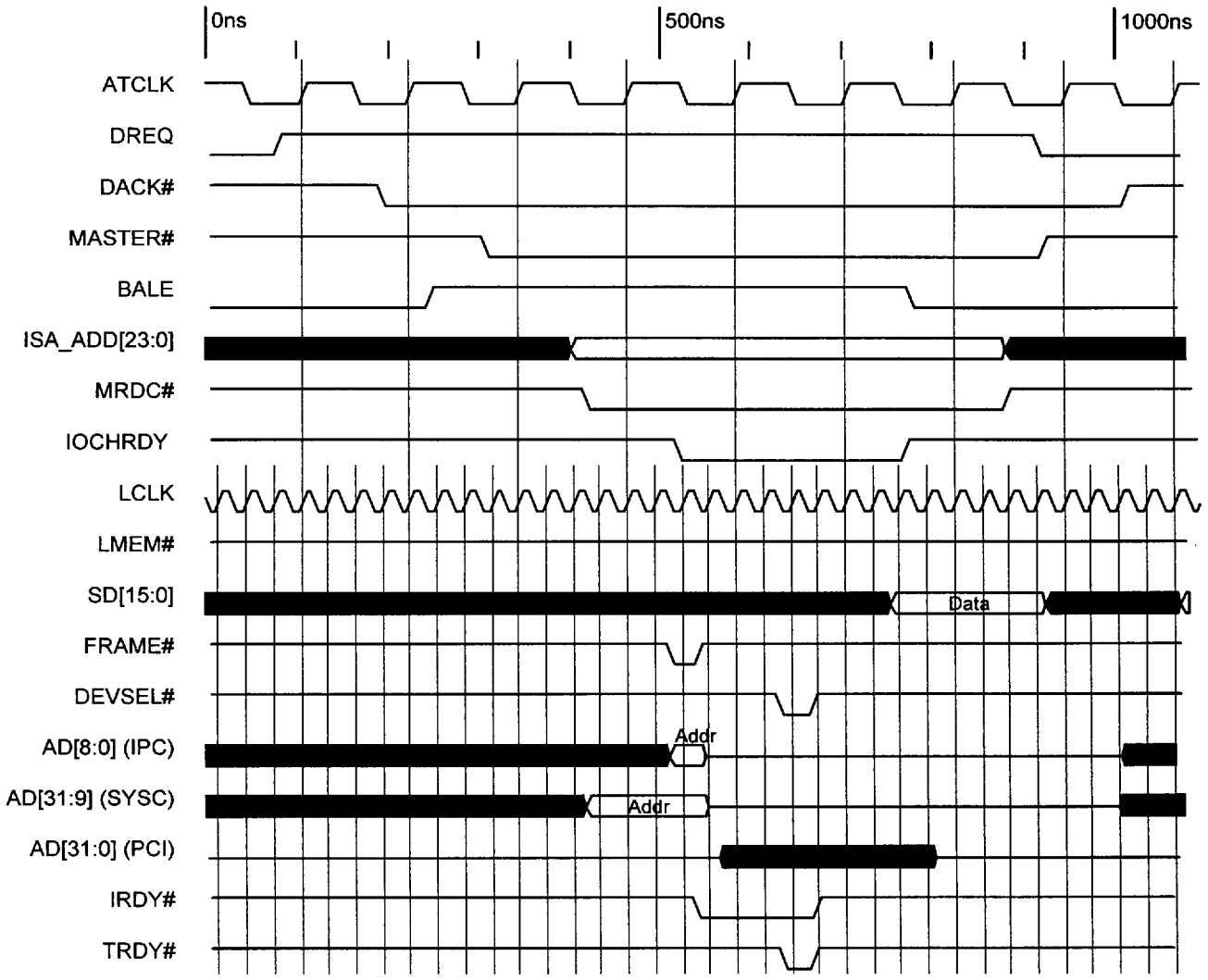
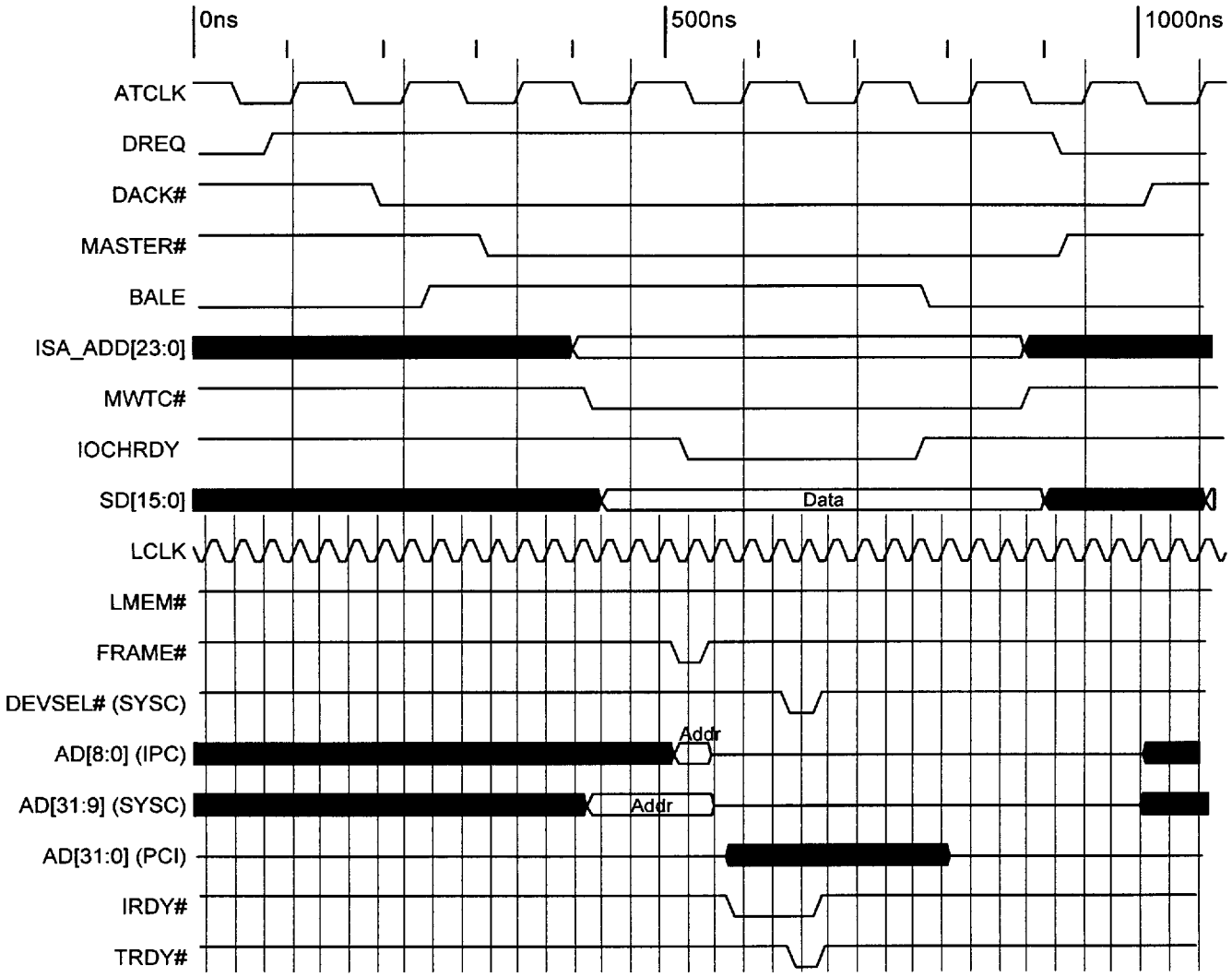


Figure 4-35 ISA Master Write to ISA Slave



4.7 AT Bus Interface

The AT bus state machine gains control when the 82C558M'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-M 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 82C558M's 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 82C558M 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 82C557M 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 82C557M 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 82C557M 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 82C557M 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 82C557M. The 82C557M 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 82C556M (based on control signals from the 82C557M). The data bus conversion from the higher order MD bus to the AD bus is done by the 82C558M, and the conversion to a 8/16-bit AT bus is also done by the 82C558M. The 82C557M 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 82C558M. The 82C558M 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 82C558M 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 82C557M provides all of the signals to control external bidirectional data buffers.

4.11 Internal Integrated 82C206

The following subsections give detailed operational information about the 82C558M'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		Byte Pointer	Register Function
DMA8	DMA16	XIOR#	XIOW#		
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.

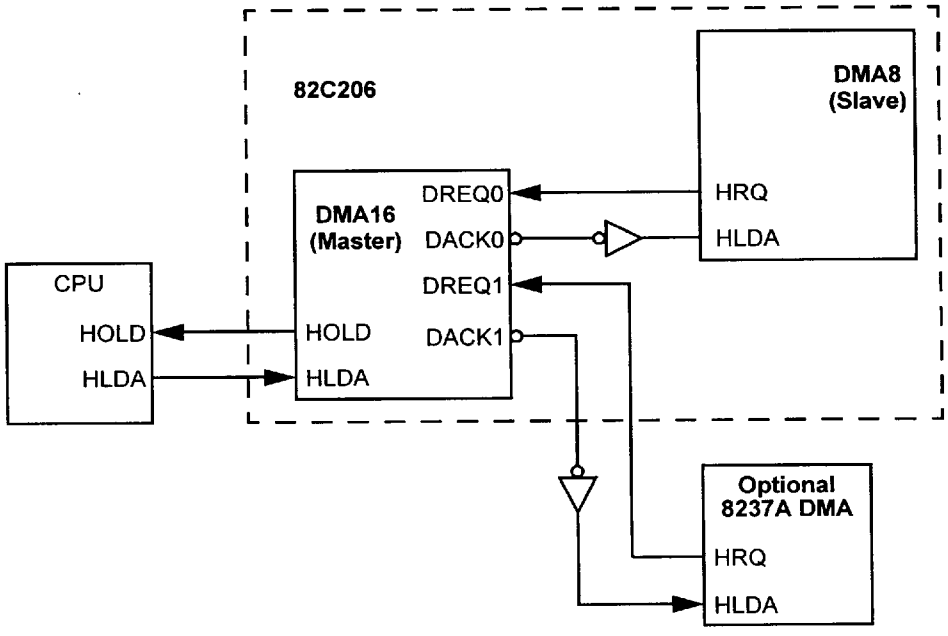
Read Transfers move data from memory to an I/O peripheral by generating the memory address and asserting MEMR# and IOW# during the same transfer cycle.

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

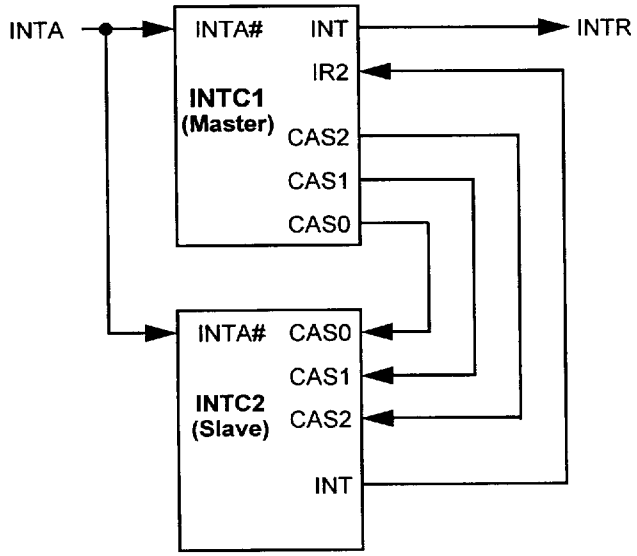
There are two interrupt controllers, INTC1 and INTC2, included in the 82C206. Each of the interrupt controllers is equivalent to an 8259A device operating in X86 mode. The two devices are interconnected and must be programmed to operate in the Cascade mode for all 16 interrupt channels to operate properly.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

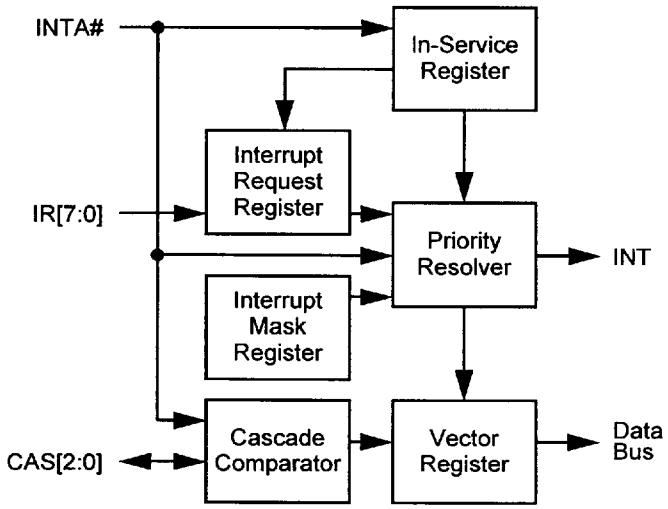


Figure 4-39 Interrupt Sequence

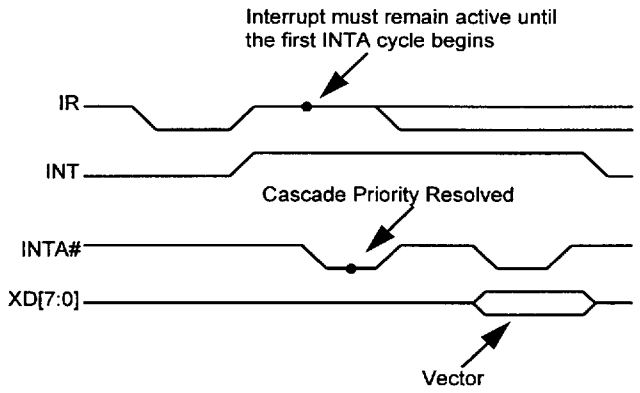


Table 4-21 Interrupt Vector Byte

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).

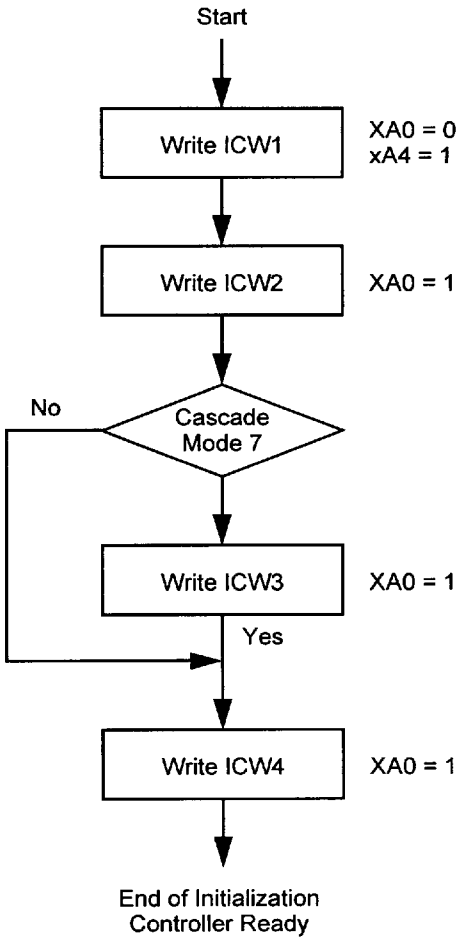
4.11.5.6.1 Initialization Command Words (ICWs)

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

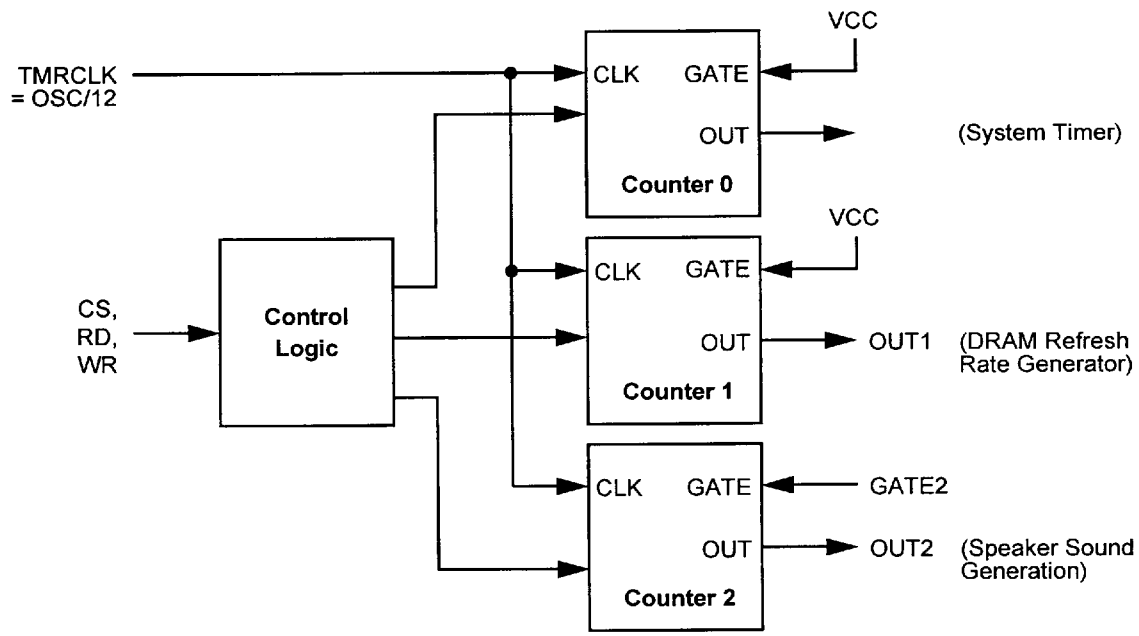
The internal timer counter use an internal signal TMRCLK which is derived from the OSC input of the 82C206. For the sake of simplicity, all references to the timer counter clock will be TMRCLK in the following description. All three counters are driven from a common clock input, TMRCLK (TMRCLK = OSC/12) 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 (0SC/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-M 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-M 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-M 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-M Chipset to the CPU. In response, the CPU saves the state of all its internal registers, asserts SMIACT# to the Viper-M Chipset and then begins executing the SMI code. In the SMI code, the 82C558M'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-M Chipset can support hardware power management through an external latch or internally through programmable pins on the 82C558M IPC. At power-on reset, if 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 82C558M change functionality to support hardware power management without a latch.

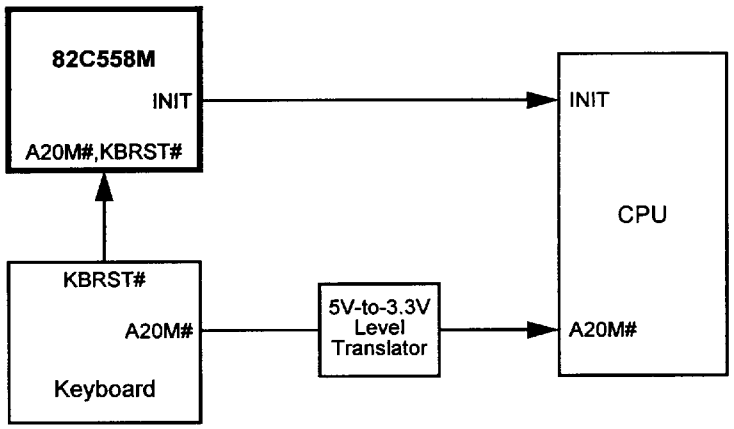
The PPWRL# pin becomes PPWRL1 and the MP4,GPCS0# becomes PPWR1. With this implementation, the 82C558M will not be able to generate memory parity for PCI master writes. Hence, maximum performance for PCI master writes will be sacrificed.

4.13 Fast GATEA20 and Reset Emulation

The 82C557M 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 82C558M's system control register space) then the keyboard has to generate the GATEA20 and KBRST signals. In this case, the A20M#,KBRST# signal on the 82C558M 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



4.14 Shadow ROM & BIOS Cacheability

When using the Viper-M 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 Offset 4Bh-4Ah of the 82C558M's System Control Registers.
2) Enable ROM contents to be copied into DRAM. To do this, the appropriate bits in Index 04h, 05h, and 06h in the 82C557M's 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 Index 04h, 05h, and 06h in the 82C557M's 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 82C557M's 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.

4.15 Scatter/Gather

The Viper-M Chipset provides another useful feature in the form of the scatter/gather functionality. Primarily scatter/gather functionality provides the user with the ability to transfer multiple buffers of data between memory and I/O without any intervention by the CPU. This facilitates the CPU to continue its processes without any interruption, thereby increasing the overall performance of the system.

In scatter/gather, the DMA reads memory addresses and word counts from multiple buffer descriptors located in the system memory (PCI or ISA). These buffer descriptors are called the scatter/gather descriptor (SGD) table. With the aid of these descriptors, the DMA controller can sustain DMA transfers until all the buffers in the descriptor table are transferred. The SGD table pointer address holds the address of the next buffer descriptor in the SGD table.

In order to perform a scatter/gather transfer the following needs to be done:

1) Software prepares the SGD table in system memory. A typical SGD will consist of an address pointer to the starting address and the transfer count of the memory buffer to be transferred.
2) Initialize the DMA channel mode registers with transfer specific information like 8/16-bit I/O device, transfer mode, etc.
3) Software provides the starting address of the SGD table by loading the SGD table pointer register.
4) Initiate the scatter/gather function by writing a start command to the Scatter/Gather Command Register.
5) The Mask Register should be the last register to be cleared to prevent the DMA from starting the DMA cycle with a partially loaded command description.
6) Once the register set is loaded and the channel is unmasked, DMA can begin.

Refer to the Section 4.17.4, Programming the MIDE Module Registers, for more information.

4.16 Type F DMA

The Viper-M Chipset provides a form of compressed timing on the DMA called "Type F DMA". This mode provides ISA compatible timing for fast DMA slave devices. Type F timing basically runs at 360ns/cycle or three ISA clock cycles during the repeated portion of a Block or Demand mode transfer. Timing on DMA cycles is controlled by programming bits [5:4] of registers in the I/O space. Programming bits [5:4] of the register located at the I/O address of 040Bh to a "00" will yield compatible timings on DMA channels 0 through 3. Programming these aforementioned bits to a "11" will yield compressed timing or Type F DMA timing on channels 0 through 3. Likewise, programming bits [5:4] of the register located at the I/O address of 04D6h to a "00" will yield compatible timings on DMA channels 5 through 7. Programming these aforementioned bits to a "11" will yield compressed timing or Type F DMA timing on channels 5 through 7. (Refer to Table 5-76 through Table 5-79 in the Register Description Section for information on the DMA System I/O Registers.)

4.17 IDE Interface

OPTi's Bus Master PCI IDE (MIDE) Module is designed as a fast and flexible interface between the PCI bus and two channels of IDE devices (up to four devices). An integrated 8-level (32-byte) read prefetch FIFO and an 8-level (32-byte) posted write FIFO supports bus mastering burst read and write operations on the PCI bus, substantially improving the performance over the typical slave IDE implementations. The Enhanced ATA Specification can be supported by programming the internal registers up to IDE PIO Mode 5 and Single- and/or Multi-Word DMA Mode 2 timing.

4.17.1 Overview

The Viper-M Chipset provides a full function PCI local bus IDE controller capable of master mode operation. The chipset is capable of arbitrating for the ownership of the PCI local bus and transfer data between the IDE device and local memory. The IDE controller in the Viper-M Chipset conforms to the ATA standard for IDE disk controllers.

By performing the IDE data transfers as a bus master instead of a slave, the chipset off-loads the CPU from having to perform the transfers. This benefit is realized in the form of the CPU not having to perform programmed I/O transfers to effect the data transfer between the disk and the memory. This feature improves system performance dramatically, especially in systems that operate in multitudinous environments.

Since the Viper-M Chipset implements this feature, the user's need to develop indigenous software to provide a complete product is greatly reduced.

The master mode of operation is an extension to the standard IDE controller model. Thus, systems can still revert back to slave mode IDE if they so desire. The master mode of operation is designed to work with any IDE device that supports DMA transfers on the IDE bus. Devices that do not support DMA on the IDE bus can revert to transfers across the IDE bus using programmed I/O.

The master mode of operation simply defines a simple scatter/gather mechanism allowing large blocks of data to be scattered to or gathered from memory. Such a mechanism improves system performance by reducing the number of interrupts to the CPU and the number of interventions by the CPU.

The following subsections will cover the MIDE's register formats and programming information.

4.17.2 MIDE Register Descriptions

4.17.2.1 MIDE Configuration Registers

The configuration space for the MIDE Module can be mapped in different locations in different OPTi chipsets. In the case of the 82C558M (Viper-M Chipset), it can be mapped into two locations. It is controlled by Index FFh[4]. If this bit is set to 0, the configuration space is mapped as Device 14h (AD31 = 1,) Function 0. If FFh[4] is 1, the MIDE is mapped as Device 01h (AD12 = 1), Function 1. This section describes the registers implemented in the 256-byte configuration space. All registers not implemented always return zero during read cycles.

Table 4-24 Vendor ID Register - 01h-00h

Bit(s)	Name	Type	Default	Function
15:0	VID	RO	1045h	***Vendor ID:*** This register identifies the OPTi ID.

Table 4-25 Device ID Register - 03h-02h

Bit(s)	Name	Type	Default	Function
15:0	DID	RO	C621h	***Device ID:*** This register identifies the ID of the bus master IDE controller. If Index FFh[4] is set to 1, this register returns C558h.

Table 4-26 Command Register - 05h-04h

Bit(s)	Name	Type	Default	Function
15:7	---	RO	0000 0000 0	***Reserved***
6	PEN	R/W	0	***Parity Checking Enable:*** When this bit is set, the MIDE Module generates PERR# if a parity error occurs during I/O write cycles. If the bit is reset, parity checking is ignored. For I/O read cycles, the MIDE Module always generates the parity bit.
5	---	RO	0	***Reserved***
4	MWI	R/W	0	***Memory Write and Invalid:*** When this bit is 1, the MIDE Module may generate the command. When it is 0, memory write will be used instead.
3	---	RO	0	***Reserved***
2	MSTR	R/W	0	***Master Enable:*** When this bit is 1, the MIDE Module is a PCI master to generate PCI accesses.
1	---	RO	0	***Reserved***
0	IOEN	R/W	1	***Input/Output Enable:*** When this bit is set, the MIDE Module enables the I/O accesses. If reset, all I/O accesses are disabled.

Table 4-27 Status Register - 07h-06h

Bit(s)	Name	Type	Default	Function
15	PER	R/W	0	***Parity Error:*** This bit is set whenever the MIDE Module detects a parity error. This bit cleared by writing 8000h to this register.
14	---	RO	0	***Reserved***
13	MABORT	R/W	0	***Master Abort:*** As a PCI master, the MIDE Module sets this bit to 1 when its transaction is terminated with a master abort.
12	TABORT	R/W	0	***Target Abort:*** As a PCI master, the MIDE Module sets this bit to 1 when its transaction is terminated with a target abort.
11	---	RO	0	***Reserved***
10:9	SELTIM [1:0]	RO	01	***Select Timing:*** These are read only bits indicating allowable timing assertion for DEVSEL#.
8	DPER	R/W	0	***Data Parity:*** As a PCI master, the MIDE Module sets this bit to 1 when it detected a data parity error. 0 = No data parity detected 1 = Data parity detected
7	BTB	RO	1	***Back-to-Back Transactions:*** When this bit is set to 1, it allows fast back-to-back transactions.
6:0	---	RO	0000 000	***Reserved***

Table 4-28 Revision ID Register - 08h

Bit(s)	Name	Type	Default	Function
7:0	REVID	RO	00h	***Revision ID:*** This register identifies the revision number of the bus master IDE controller.

Table 4-29 Class Code Register - 0Bh-09h

Bit(s)	Name	Type	Default	Function
23:8	CCODE	RO	0101h	***Class Code:*** The MSB indicates the base class code for the mass storage controller. The middle byte indicates the sub-class code (IDE controller).
7	PI7	RO	1	***Bus-Mastering IDE Signature:*** This bit is set to 1 to indicate master mode support.
6:4	---	RO	000	***Reserved***
3	PI3	RO	0	***Writability of the Native/Legacy Bit for Secondary IDE:*** This bit controls whether bit 2 of this register is read only or read/write. 0 = bit 2 is read only 1 = bit 2 is read/write This bit is set only when both the FNC0 and the RELOC bits (Configuration Address Offset 40h[3:2]) are set.
2	PI2	R/W	0	***Native/Legacy for Secondary IDE:*** This bit controls whether the secondary IDE is in the Native or Legacy Mode. 0 = Legacy Mode 1 = Native Mode
1	PI1	RO	0	***Writability of the Native/Legacy Bit for Primary IDE:*** This bit controls whether bit 0 of this register is read only or read/write. 0 = bit 0 is read only 1 = bit 0 is read/write If the RELOC bit (Configuration Address Offset 40h[2]) is reset, this bit is 0. When the RELOC bit is set, this bit is 1.
0	PI0	R/W	0	***Native/Legacy for Primary IDE:*** This bit controls whether the primary IDE is in the Native or Legacy Mode. 0 = Legacy Mode 1 = Native Mode

Table 4-30 Reserved Register - 0Dh-0Ch

Bit(s)	Name	Type	Default	Function
15:0	---	RO	0000h	***Reserved***

Table 4-31 Header Type Register - 0Eh

Bit(s)	Name	Type	Default	Function
7:0	HDR	RO	00h	***Header Type:*** Configuration bit for single (default) or multi-function device. If Index FFh[4] is set to 1, this register returns 80h denoting a multi-function device.

Table 4-32 Built-In Self-Test Register - 0Fh

Bit(s)	Name	Type	Default	Function
7:0	BIST	RO	00h	***Built-In Self-Test Register***

Table 4-33 Command Block Base Address Register - 13h-10h (Primary IDE)

Bit(s)	Name	Type	Default	Function
31:0	IO1	R/W	1F1h w/ RELOC=1	***Command Block Base Address:*** This register is the I/O space indicator for the Drive Command Block. The address block has a size of eight bytes. Bits [2:0] of this register are read only and default to 001. Bits [31:3] are writable if the RELOC bit is set to 1. If the RELOC bit is set to 0, bits [31:0] are read only and return 0.

Table 4-34 Control Block Base Address Register - 17h-14h (Primary IDE)

Bit(s)	Name	Type	Default	Function
31:0	IO2	R/W	3F5h w/ RELOC=1	***Control Block Base Address:*** This register is the I/O space indicator for the Drive Control Block. The address block has a size of four bytes. Bits [1:0] of this register are read only and default to 01. Bits [31:2] are writable if the RELOC bit is set to 1. If the RELOC bit is set to 0, bits [31:0] are read only and return 0.

Table 4-35 Command Block Base Address Register - 1Bh-18h (Secondary IDE)

Bit(s)	Name	Type	Default	Function
31:0	IO3	R/W	171h w/ RELOC=1 FNC0=0	***Command Block Base Address:*** This register is the I/O space indicator for the Drive Command Block. The address block has a size of eight bytes. Bits [2:0] of this register are read only and default to 001. Bits [31:3] are writable if the RELOC bit is set to 1. If the RELOC bit is set to 0, bits [31:0] are read only and return 0.

Table 4-36 Control Block Base Address Register - 1Fh-1Ch (Secondary IDE)

Bit(s)	Name	Type	Default	Function
31:0	IO4	R/W	375h w/ RELOC=1 FNC0=0	***Control Block Base Address:*** This register is the I/O space indicator for the Drive Control Block. The address block has a size of four bytes. Bits [1:0] of this register are read only and default to 01. Bits [31:2] are writable if the RELOC bit is set to 1. If the RELOC bit is set to 0, bits [31:0] are read only and return 0.

Table 4-37 Bus Master IDE Base Address Register - 23h-20h

Bit(s)	Name	Type	Default	Function
31:0	IO5	R/W	8000 0001h	***Bus Master IDE Base Address:*** This register is the I/O base address indicator for the Bus Master IDE registers. The address block has a size of 16 bytes. Bits [3:0] of this register are read only and default to 0001. Bits [31:4] are writable.

Table 4-38 Interrupt Line Register - 3Ch

Bit(s)	Name	Type	Default	Function
7:0	INTL	R/W	14h	***Interrupt Line:*** This register indicates which input of the system interrupt controller the PCIIRQ3# interrupt pin is routed to.

Table 4-39 Interrupt Pin Register - 3Dh

Bit(s)	Name	Type	Default	Function
7:0	INTP	RO	1	***Interrupt Pin:*** The contents of this register is 1.

Table 4-40 IDE Initialization Control Register - 42h-40h

Bit(s)	Name	Type	Default	Function
23:6	----			***Reserved:*** Must be written 0.
5	ESLAVE	R/W	0	***Enhanced Slave:*** 0 = 82C621A-compatible mode, uses a 16-byte FIFO 1 = Enhanced mode, uses a 32-byte FIFO
4	IDEP	R/W	0	***Default ISA Ownership:*** 0 = MIDE Module requests the ISA bus only when it needs to access IDE cable. 1 = MIDE Module always requests the ISA bus.
3	FNC0	R/W	0	***Function 0:*** Function 0 enables or disables the secondary IDE: 0 = Enabled 1 = Disabled
2	RELOC	R/W	0	***Address Relocation:*** RELOC decides whether the I/O space addresses are relocatable through programming configuration space registers. 0 = Fixed I/O addresses (1F0h-1 F7h, 3F6h for primary; 170h-177h, 376h for secondary) 1 = Relocatable I/O addresses
1:0	MODE [1:0]	R/W	00	***Mode:*** These bits control the default 16-bit cycle times for all the IDE devices and can be overridden by programming the IDE I/O Registers. 00 = ≥ 600ns cycle time (PIO Mode 0) 01 = ≥ 383ns cycle time (PIO Mode 2) 10 = ≥ 240ns cycle time (PIO Mode 1) 11 = ≥ 180ns cycle time (PIO Mode 3)

Table 4-41 IDE Enhanced Mode Register - 43h

Bit(s)	Name	Type	Default	Function
7:6	SMODE1 [1:0]	R/W	00	***Enhanced Mode for Drive 1 on Secondary Channel:*** These two bits set 16-bit cycle times for IDE PIO Modes 4 and 5, or Multi-Word DMA Modes 1 and 2. 00 = Disabled, control by corresponding Timing Registers Set 01 = PIO Mode 4 or Multi-Word DMA Mode 1, command inactive for 2 LCLKs 10 = PIO Mode 5 or Multi-Word DMA Mode 2, command inactive for 1 LCLK 11 = Reserved The corresponding 170h/171h[3:0] must be set to 0 before these two bits are set to 01 or 10.
5:4	SMODE0 [1:0]	R/W	00	***Enhanced Mode for Drive 0 on Secondary Channel:*** These two bits set 16-bit cycle times for IDE PIO Modes 4 and 5, or Multi-Word DMA Modes 1 and 2. 00 = Disabled, control by corresponding Timing Registers Set 01 = PIO Mode 4 or Multi-Word DMA Mode 1, command inactive for 2 LCLKs 10 = PIO Mode 5 or Multi-Word DMA Mode 2, command inactive for 1 LCLK 11 = Reserved The corresponding 170h/171h[3:0] must be set to 0 before these two bits are set to 01 or 10.
3:2	PMODE1 [1:0]	R/W	00	***Enhanced Mode for Drive 1 on Primary Channel:*** These two bits set 16-bit cycle times for IDE PIO Modes 4 and 5, or Multi-Word DMA Modes 1 and 2. 00 = Disabled, control by corresponding Timing Registers Set 01 = PIO Mode 4 or Multi-Word DMA Mode 1, command inactive for 2 LCLKs 10 = PIO Mode 5 or Multi-Word DMA Mode 2, command inactive for 1 LCLK 11 = Reserved The corresponding 1F0h/1F1h[3:0] must be set to 0 before these two bits are set to 01 or 10.
1:0	PMODE0 [1:0]	R/W	00	***Enhanced Mode for Drive 0 on Primary Channel:*** These two bits set 16-bit cycle times for IDE PIO Modes 4 and 5, or Multi-Word DMA mode 1 and 2. 00 = Disabled, control by corresponding Timing Registers Set 01 = PIO Mode 4 or Multi-Word DMA Mode 1, command inactive for 2 LCLKs 10 = PIO Mode 5 or Multi-Word DMA Mode 2, command inactive for 1 LCLK 11 = Reserved The corresponding 1F0h/1F1h[3:0] must be set to 0 before these two bits are set to 01 or 10.

4.17.2.2 I/O Registers

4.17.2.2.1 I/O Registers for Primary IDE

The register addresses are referred to in this section by their power-up default addresses If the power-up default is modified by writing to Configuration Register IO1, then these registers will be relocated accordingly.

The MIDE Module contains registers at seven I/O ports accessible after two consecutive 16-bit I/O reads from address 1F1h, followed by a byte write 03h to 1F2h. Any other I/O cycle between these two reads will disable access to the MIDE Module registers. Refer to Section 4.17.4.2, Programming the IDE Mode Timing, for programming details.

Table 4-42 Internal ID Register - 1F2h

Bit(s)	Name	Type	Default	Function
7	CNFDIS	WO	1	***Configuration Disable:*** This bit must be set to 0 in order to access the MIDE Module's internal registers. Any write to this register with CNFDIS = 1 will disable all accesses to the MIDE Module registers until another two consecutive I/O reads from 1F1h.
6	CNFOFF	WO	0	***Configuration Off:*** This bit must be set to 0 in order to access MIDE Module internal registers. Any write to this register with CNFOFF = 1 will disable all accesses to the MIDE Module registers until power down or reset.
5:2	---	RO	x	***Reserved:*** Must be written 0.
1:0	---		00	***Reserved:*** These two bits must be written 11. If not, all writes to IDE I/O Registers will be blocked.

Table 4-43 Read Cycle Timing Register-A - 1F0h, Index-0

Bit(s)	Name	Type	Default	Function
7:4	RDPW [3:0]	R/W	xxxx	***Read Pulse Width:*** The value programmed in this register plus one determines the DRD# pulse width in LCLKs (for a 16-bit read from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	RDREC [3:0]	R/W	xxxx	***Read Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DRD# and the next DA[2:0]/DCSx# being presented (after a 16-bit read from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68

Read Cycle Timing Register-A shares the I/O address with Read Cycle Timing Register-B, indexed by bit 0 of the Miscellaneous Register. It controls the read cycle timing of the IDE Data Register for the drive selected by bits [3:2] of the Control Register.

Table 4-44 Read Cycle Timing Register-B - 1F0h, Index-1

Bit(s)	Name	Type	Default	Function
7:4	RDPW [3:0]	R/W	xxxx	***Read Pulse Width:*** The value programmed in this register plus one determines the DRD# pulse width in LCLKs (for a 16-bit read from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	RDREC [3:0]	R/W	xxxx	***Read Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DRD# and the next DA[2:0]/DCSx# being presented (after a 16-bit read from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Read Cycle Timing Register-B shares the I/O address with Read Cycle Timing Register-A, indexed by bit 0 of the Miscellaneous Register. It controls the read cycle timing of the IDE Data Register for the drive not selected by bits [3:2] of the Control Register if bit 7 of the Control Register is set.

Table 4-45 Write Cycle Timing Register-A - 1F1h, Index-0

Bit(s)	Name	Type	Default	Function
7:4	WRPW [3:0]	R/W	xxxx	***Write Pulse Width:*** The value programmed in this register plus one determines the DWR# pulse width in LCLKs (for a 16-bit write from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	WRREC [3:0]	R/W	xxxx	***Write Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DWR# and the next DA[2:0]/DCSx# being presented (after a 16-bit write from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Write Cycle Timing Register-A shares the I/O address with Write Cycle Timing Register-B, indexed by bit 0 of the Miscellaneous Register. It controls the write cycle timing of the IDE Data Register for the drive selected by bits [3:2] of the Control Register.

Table 4-46 Write Cycle Timing Register-B - 1F1h, Index-1

Bit(s)	Name	Type	Default	Function
7:4	WRPW [3:0]	R/W	xxxx	***Write Pulse Width:*** The value programmed in this register plus one determines the DWR# pulse width in LCLKs (for a 16-bit write from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	WRREC [3:0]	R/W	xxxx	***Write Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DWR# and the next DA[2:0]/DCSx# being presented (after a 16-bit write from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Write Cycle Timing Register-B shares the I/O address with Write Cycle Timing Register-A, indexed by bit 0 of the Miscellaneous Register. It controls the write cycle timing of the IDE Data Register for the drive not selected by bits [3:2] of the Control Register if bit 7 of the Control Register is set.

Table 4-47 Control Register - 1F3h

Bit(s)	Name	Type	Default	Function
7	REGTIM2	R/W	0	***Enable Timing Registers-B:*** When set, this bit enables Cycle Timing Registers-B (1F0h and 1F1h of Index-1) and Miscellaneous Timing Register 1F6h[5:1] to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for any drive not selected by 1F3h[3:2].
6:5	---	RO	00	***Reserved:*** Must be written with 0.
4	EN1WSRD	R/W	0	***Enable One Wait State Read:*** 0 = 2 WS minimum 1 = 1WS minimum for data reads
3	REGTIM1	R/W	0	***Enable Timing Registers-A, Drive 1:*** When set, this bit enables Cycle Timing Registers-A (1F0h and 1F1h of Index-0) to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for Drive 1.
2	REGTIM0	R/W	0	***Enable Timing Registers-A, Drive 0:*** When set, this bit enables Cycle Timing Registers-A (1F0h and 1F1h of Index-0) to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for Drive 0.
1	---		0	***Reserved:*** Must be written 0.
0	---	RO	1	***Reserved:*** Must be written 1.

Note: For all new software controls the IDE timing through registers programming, bits 2, 3 and 7 of the Control Register should be enabled after the Cycle Timing Registers and Miscellaneous Register are programmed. See Table 4-66 for programming options.

Table 4-48 Strap Register - 1F5h

Bit(s)	Name	Type	Default	Function
7	---	RO	0	***Reserved:*** Must be written 1.
6:5	REV [1:0]	RO	11	***Revision Number:*** When the value of this register is set to 11, the contents of REVID Register should be used to find the revision level of the chip.
4	DINTR	RO		***DINTR Status:*** Returns the state of the DINTR input.
3:2	MODE [1:0]	RO	00	***Mode:*** Returns information about drive speed as determined by MODE[1:0] bits at PCI Configuration Address Offset 40h[1:0]. (Refer to the Mode Bit description for specific information.)
1	---	RO	1	***Reserved:*** Must be written 1.
0	SPD0	R/W	0	***PCI CLK Speed:*** PCI bus clock frequency select. 0 = 33MHz 1 = 25MHz

Table 4-49 Miscellaneous Register - 1F6h

Bit(s)	Name	Type	Default	Function
7	---		0	***Reserved:*** Must be written 0.
6	ENPREF	R/W	0	***Enable Read Prefetch:*** Enables/disables read prefetch. 0 = Disable 1 = Enable
5:4	ASU[1:0]	R/W	xx	***Address Setup Time:*** The value programmed in this register plus one determines the address setup time between DRD# or DWR# going active and DA[2:0], DCS3#, DCS1# being presented, measured in LCLKs. See Table 4-67 or Table 4-68.
3:1	DRDY [2:0]	R/W	xxx	***Delay:*** The value programmed in this register plus two determines the minimum number of LCLKs between DRDY# going high and DRD# or DWR# going inactive. See Table 4-67 or Table 4-68.
0	INDEX-0	R/W	0	***Index-0:*** This bit is used to select between Cycle Timing Registers-A and -B located at 1F0h and 1F1h.

4.17.2.2.2 I/O Registers for Secondary IDE

The register addresses are referred to in this section by their power-up default addresses. If the power-up default is modified by writing to Configuration Register IO3, then these registers will be relocated accordingly.

The MIDE Module contains registers at seven I/O ports accessible after two consecutive 16-bit I/O reads from address 171h, followed by a byte write 03h to 1F2h. Any other I/O cycle between these two reads will disable access to the MIDE Module registers. Refer to Section 4.17.4.2, Programming the IDE Mode Timing, for programming details.

Table 4-50 Internal ID Register - 172h

Bit(s)	Name	Type	Default	Function
7	CNFDIS	WO	1	***Configuration Disable:*** This bit must be set to 0 in order to access the MIDE Module's internal registers. Any write to this register with CNFDIS = 1 will disable all accesses to the MIDE Module registers until another two consecutive I/O reads from 171h.
6	CNFOFF	WO	0	***Configuration Off:*** This bit must be set to 0 in order to access the MIDE Module's internal registers. Any write to this register with CNFOFF = 1 will disable all accesses to the MIDE Module registers until power-down or reset.
5:2	---	RO	xxxx	***Reserved:*** Must be written 0.
1:0	---		00	***Reserved:*** These two bits must be written 11. If not, all writes to IDE I/O Registers will be blocked.

Table 4-51 Read Cycle Timing Register-A - 170h, Index-0

Bit(s)	Name	Type	Default	Function
7:4	RDPW [3:0]	R/W	xxxx	***Read Pulse Width:*** The value programmed in this register plus one determines the DRD# pulse width in LCLKs (for a 16-bit read from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	RDREC [3:0]	R/W	xxxx	***Read Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DRD# and the next DA[2:0]/DCSx# being presented (after a 16-bit read from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Read Cycle Timing Register-A shares the I/O address with Read Cycle Timing Register-B, indexed by bit 0 of the Miscellaneous Register. It controls the read cycle timing of the IDE Data Register for the drive selected by bits [3:2] of the Control Register.

Table 4-52 Read Cycle Timing Register-B - 170h, Index-1

Bit(s)	Name	Type	Default	Function
7:4	RDPW [3:0]	R/W	xxxx	***Read Pulse Width:*** The value programmed in this register plus one determines the DRD# pulse width in LCLKs (for a 16-bit read from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	RDREC [3:0]	R/W	xxxx	***Read Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DRD# and the next DA[2:0]/DCSx# being presented (after a 16-bit read from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Read Cycle Timing Register-B shares the I/O address with Read Cycle Timing Register-A, indexed by bit 0 of the Miscellaneous Register. It controls the read cycle timing of the IDE Data Register for the drive not selected by bits [3:2] of the Control Register if bit 7 of the Control Register is set.

Table 4-53 Write Cycle Timing Register-A - 171h, Index-0

Bit(s)	Name	Type	Default	Function
7:4	WRPW [3:0]	R/W	xxxx	***Write Pulse Width:*** The value programmed in this register plus one determines the DWR# pulse width in LCLKs (for a 16-bit write from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	WRREC [3:0]	R/W	xxxx	***Write Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DWR# and the next DA[2:0]/DCSx# being presented (after a 16-bit write from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Write Cycle Timing Register-A shares the I/O address with Write Cycle Timing Register-B, indexed by bit 0 of the Miscellaneous Register. It controls the write cycle timing of the IDE Data Register for the drive selected by bits [3:2] of the Control Register.

Table 4-54 Write Cycle Timing Register-B - 171h, Index-1

Bit(s)	Name	Type	Default	Function
7:4	WRPW [3:0]	R/W	xxxx	***Write Pulse Width:*** The value programmed in this register plus one determines the DWR# pulse width in LCLKs (for a 16-bit write from the IDE Data Register). See Table 4-67 or Table 4-68.
3:0	WRREC [3:0]	R/W	xxxx	***Write Recovery Time:*** The value programmed in this register plus two determines the recovery time between the end of DWR# and the next DA[2:0]/DCSx# being presented (after a 16-bit write from the IDE Data Register), measured in LCLKs. See Table 4-67 or Table 4-68.

Write Cycle Timing Register-B shares the I/O address with Write Cycle Timing Register-A, indexed by bit 0 of the Miscellaneous Register. It controls the write cycle timing of the IDE Data Register for the drive not selected by bits [3:2] of the Control Register if bit 7 of the Control Register is set.

Table 4-55 Control Register - 173h

Bit(s)	Name	Type	Default	Function
7	REGTIM2	R/W	0	***Enable Timing Registers-B:*** When set, this bit enables Cycle Timing Registers-B (170h and 171h of Index-1) and Miscellaneous Timing Register 176h[5:1] to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for any drive not selected by 173h[3:2].
6:4	---		000	***Reserved:*** Must be written with 0.
3	REGTIM1	R/W	0	***Enable Timing Registers-A, Drive 1:*** When set, this bit enables Cycle Timing Registers-A (170h and 171h of Index-0) to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for Drive 1.
2	REGTIM0	R/W	0	***Enable Timing Registers-A, Drive 0:*** When set, this bit enables Cycle Timing Registers-A (170h and 171h of Index-0) to override the IDE timing set by the MODE[1:0] bits in PCI Configuration Address Offset 40h for Drive 1.
1	---		0	***Reserved:*** Must be written 0.
0	---	RO	1	***Reserved:*** Must be written 1.

Note: For all new software controls the IDE timing through registers programming, bits 2, 3, and 7 of the Control Register should be enabled after the Cycle Timing Registers and Miscellaneous Register are programmed. See Table 4-66 for programming options.

Table 4-56 Strap Register - 175h

Bit(s)	Name	Type	Default	Function
7	---		0	***Reserved:*** Must be written 1.
6:5	REV [1:0]	RO	11	***Revision Number:*** When the value of this register is set to 11, the contents of REVID Register should be used to find the revision level of the chip.
4	SDINTR	RO	x	***SDINTR Status:*** Returns the state of the SDINTR input.
3:2	---		00	***Reserved:*** Must be written 0.
1	---	RO	0	***Reserved:*** Must be written 1.
0	---	RO	0	***Reserved:*** Must be written 0.

Table 4-57 Miscellaneous Register - 176h

Bit(s)	Name	Type	Default	Function
7	---		0	***Reserved:*** Must be written 0.
6	ENPREF	R/W	0	***Enable Read Prefetch:*** Enables/disables read prefetch. 0 = Disable 1 = Enable
5:4	ASU[1:0]	R/W	xx	***Address Setup:*** The value programmed in this register plus one determines the address setup time between DRD# or DWR# going active and DA[2:0], DCS3#, DCS1# being presented, measured in LCLKs. See Table 4-67 or Table 4-68.
3:1	DRDY [2:0]	R/W	xxx	***DRDY Delay:*** The value programmed in this register plus two determines the minimum number of LCLKs between DRDY# going high and DRD# or DWR# going inactive. See Table 4-67 or Table 4-68.
0	INDEX-0	R/W	0	***Index-0:*** This bit is used to select between Cycle Timing Registers-A and -B located at 170h and 171h.

4.17.3 Physical Region Descriptor Table

Before the MIDE controller starts a master transfer it is given a pointer to a Physical Region Descriptor Table. This table contains some number of Physical Region Descriptors (PRDs) which describe areas of memory that are involved in the data transfer. The descriptor table must be aligned on a 4-byte boundary and the table cannot cross a 64K boundary in memory.

4.17.3.1 Physical Region Descriptor

The physical memory region to be transferred is described by a Physical Region Descriptor (PRD). The data transfer will proceed until all the regions described by the PRDs in the table have been transferred.

Each Physical Region Descriptor entry is eight bytes in length. The first four bytes specify the start address of a physical memory region. The next four bytes specify the size of the region in bytes (64K byte limit per region). A value of zero in these two bytes indicates 64K. Bit 7 of the last byte indicates the end of the table; bus master operation terminates when the last descriptor has been retired.

4.17.3.2 Bus Master IDE Registers

The bus master IDE function uses 16 bytes of I/O space. The base address of this block of I/O space is pointed to by the Bus Master IDE Base Address Register (IO5) in the PCI Configuration space. All bus master IDE I/O space registers can be accessed as byte, word, or dword quantities. The description of the 16 bytes of I/O registers is shown in Table 4-59.

Table 4-58 Physical Region Descriptor Table Entry

Bit(s)	Name	Default	Function
Byte-0, bit 0	---	0	0 (RO)
Byte-[3:1] Byte-0, bits [7:1]	BASE	xxxx xxxx	Memory Region Physical Base Address [31:1]
Byte-4, bit 0	---	0	0 (RO)
Byte-5 Byte-4, bits [7:1]	COUNT	xxxx	Byte Count [15:1]
Byte-6	---	xx	Reserved
Byte-7, bits [6:0]	---	xx	Reserved
Byte-7, bit 7	EOT	x	End of Table

Note: The memory region specified by the descriptor is further restricted such that the region cannot straddle a 64K boundary. This means the byte count is limited to 64K and the incrementer for the current address register only extends from bit 1 to bit 15.

Table 4-59 Bus Master IDE Registers

Offset from Base Address	Register Access	Register Name/Function
00h	R/W	Bus Master IDE Command Register for Primary IDE
01h		Device-specific
02h	RWC	Bus Master IDE Status Register for Primary IDE
03h		Device-specific
04h-07h	R/W	Bus Master IDE PRD Table Address for Primary IDE
08h	R/W	Bus Master IDE Command Register for Secondary IDE
09h		Device-specific
0Ah	RWC	Bus Master IDE status Register for Secondary IDE
0Bh		Device-specific
0Ch-0Fh	R/W	Bus Master IDE PRD Table Address for Secondary IDE

Table 4-60 Bus Master IDE Command Register for Primary IDE - Base Address + 00h

Bit(s)	Type	Default	Function
7:4		0000	***Reserved:*** Must be written 0.
3	R/W	0	***Read or Write Control:*** This bit sets the direction of the bus master transfer. When set to 0, PCI bus master reads are performed. When set to 1, PCI bus master writes are performed. This bit must NOT be changed when the bus master function is active.
2:1		00	***Reserved:*** Must be written 0.
0	R/W	0	***Start/Stop Bus Master:*** Writing a 1 to this bit enables bus master operation of the controller. Bus master operation begins when this bit is detected changing from 0 to 1. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation can be halted by writing 0 to this bit. All state information is lost when a 0 is written; master mode operation cannot be stopped and then resumed. If this bit is reset while bus master operation is still active (i.e., the Bus Master IDE Active bit of the Bus Master IDE Status Register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status Register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded before being written to memory. This bit is intended to be reset after the data transfer is completed, as indicated by either the Bus Master IDE Active bit or the Interrupt bit of the Bus Master IDE Status Register for that IDE channel.

Table 4-61 Bus Master IDE Status Register for Primary IDE - Base Address + 02h

Bit(s)	Type	Default	Function
7	RO	0	***Simplex Only:*** This bit indicates that both bus master channels (primary and secondary) can be operated at the same time.
6	R/W	0	***Drive 1 DMA Capable:*** This read/write bit is set by device-dependent code (BIOS or device driver) to indicate that Drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance.
5	R/W	0	***Drive 0 DMA Capable:*** This read/write bit is set by device-dependent code (BIOS or device driver) to indicate that Drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance.
4:3		00	***Reserved:*** Must be written 0.
2	R/W	0	***Interrupt:*** This bit is set by the rising edge of the IDE interrupt line. It is cleared when a 1 is written to it by software. Software can use this bit to determine if an IDE device has asserted its interrupt line. When this bit is read as a 1, all data transferred from the drive is visible in system memory.
1	R/W	0	***Error:*** This bit is set when the controller encounters an error transferring data to/from memory. The exact error condition is bus-specific and can be determined in a bus-specific manner. This bit is cleared when a 1 is written to it by software.
0	R/W	0	***Bus Master IDE Active:*** This bit is set when the Start bit is written to the Command Register. It is cleared when the last transfer for a region is performed, where EOT (end of transfer) for that region is set in the region descriptor. It is also cleared when the Start bit is cleared in the Command Register. When this bit is read as 0, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted.

Table 4-62 Descriptor Table Pointer Register for Primary IDE - Base Address + 04h

Bit(s)	Type	Default	Function
31:2	R/W	0000h	***Base Address of Descriptor Table:*** Corresponds to A[31:2].
1:0	R/W	00	***Reserved***

Note: The Descriptor Table must be dword aligned and must not cross a 64K boundary in memory.

Table 4-63 Bus Master IDE Command Register for Secondary IDE - Base Address + 08h

Bit(s)	Type	Default	Function
7:4		0000	***Reserved:*** Must be written 0.
3	R/W	0	***Read or Write Control:*** This bit sets the direction of the bus master transfer. When set to 0, PCI bus master reads are performed. When set to 1, PCI bus master writes are performed. This bit must NOT be changed when the bus master function is active.
2:1		00	***Reserved:*** Must be written 0.
0	R/W	0	***Start/Stop Bus Master:*** Writing a 1 to this bit enables bus master operation of the controller. Bus master operation begins when this bit is detected changing from 0 to 1. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation can be halted by writing 0 to this bit. All state information is lost when a 0 is written; master mode operation cannot be stopped and then resumed. If this bit is reset while bus master operation is still active (i.e., the Bus Master IDE Active bit of the Bus Master IDE Status Register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status Register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded before being written to memory. This bit is intended to be reset after the data transfer is completed, as indicated by either the Bus Master IDE Active bit or the Interrupt bit of the Bus Master IDE Status Register for that IDE channel.

Table 4-64 Bus Master IDE Status Register for Secondary IDE - Base Address + 0Ah

Bit(s)	Type	Default	Function
7	RO	0	***Simplex Only:*** This bit indicates that both bus master channels (primary and secondary) can be operated at the same time.
6	R/W	0	***Drive 1 DMA Capable:*** This read/write bit is set by device dependent code (BIOS or device driver) to indicate that Drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance.
5	R/W	0	***Drive 0 DMA Capable:*** This read/write bit is set by device dependent code (BIOS or device driver) to indicate that Drive 0 for this channel is capable of DMA transfers and that the controller has been initialized for optimum performance.
4:3		00	***Reserved:*** Must be written 0.
2	R/W	0	***Interrupt:*** This bit is set by the rising edge of the IDE interrupt line. It is cleared when a 1 is written to it by software. Software can use this bit to determine if an IDE device has asserted its interrupt line. When this bit is read as a 1, all data transferred from the drive is visible in system memory.
1	R/W	0	***Error:*** This bit is set when the controller encounters an error transferring data to/from memory. The exact error condition is bus-specific and can be determined in a bus-specific manner. This bit is cleared when a 1 is written to it by software.
0	R/W	0	***Bus Master IDE Active:*** This bit is set when the Start bit is written to the Command Register. It is cleared when the last transfer for a region is performed, where EOT (end of transfer) for that region is set in the region descriptor. It is also cleared when the Start bit is cleared in the Command Register. When this bit is read as 0, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted.

Table 4-65 Descriptor Table Pointer Register for Secondary IDE - Base Address + 0Ch

Bit(s)	Type	Default	Function
31:2	R/W	0000h	***Base Address of Descriptor Table:*** Corresponds to A[31:2].
1:0		00	***Reserved***

Note: The Descriptor Table must be dword aligned and must not cross a 64K boundary in memory.

4.17.4 Programming the MIDE Module Registers

4.17.4.1 Standard Programming Sequence for Bus Mastering Operations

To initiate a bus master transfer between memory and an IDE DMA slave device, the following steps are required:

1) Software prepares a PRD Table in system memory. Each PRD is eight bytes long and consists of an address pointer to the starting address and the transfer count of the memory buffer to be transferred. In any given PRD Table, two consecutive PRDs are offset by eight bytes and are aligned on a 4-byte boundary. Bit 7 of the eighth byte in the PRD Table will be set to a 1 if that is the last entry in the table.
2) Software provides the starting address of the PRD Table by loading the PRD Table Pointer Register. The direction of the data transfer is specified by setting the Read/Write Control bit. Clear the Interrupt bit and Error bit in the Status Register.
3) Software issues the appropriate DMA transfer command to the disk device.
4) Engage the bus master function by writing 1 to the Start bit in the Bus Master IDE Command Register for the appropriate channel.
5) The controller transfers data to/from memory responding to DMA requests from the IDE device.
6) At the end of the transfer, the IDE device signals an interrupt.
7) In response to the interrupt, software resets the Start/Stop bit in the Command Register. Next it reads the controller's status and then the drive's status to determine if the transfer completed successfully.

4.17.4.2 Programming the IDE Mode Timing

Table 4-67 and Table 4-68 show the timing and recommended register settings for various IDE modes defined in the Enhanced IDE Specifications. They include PIO transfer, Single-Word DMA transfer, and Multi-Word DMA transfer modes. The actual cycle time equals the sum of actual command active time and actual command inactive (command recovery and address setup) time. These three timing requirements shall be met. In some cases, the minimum cycle time requirement is greater than the sum of the command pulse and command recovery time. This means either the command active (command pulse) or command inactive time (command recovery and address setup) can be lengthened to ensure that the minimum cycle times are met.

Figure 4-43 is a flow chart that describes how to program the primary channel of the MIDE interface. For the secondary channel, a similar procedure can be done by changing all the indexes from 1Fxh to 17xh.

4.17.4.3 Programming the IDE Interrupt Routing

Table 4-69 details the interrupt routing mechanism for the MIDE Module while in the Legacy and Native Modes. The system BIOS needs to program them accordingly.

Table 4-66 REGTIMx Programming Options

REGTIM0	REGTIM1	REGTIM2	Drive 0 Control	Drive 1 Control
1[1]	0	1	Index-0	Index-1
0	1	1	Index-1	Index-0
0	0	1	Index-1	Index-1
1	0	0	Index-0	Default[2]
0	2	0	Default[2]	Index-0
0	0	0	Default[2]	Default[2]
1	1	x	Index-0	Index-0

1. Recommended configuration.
2. Refer to PCI Configuration Address Offset 40h[1:0] for default values.

Table 4-67 16-Bit Timing Parameters with PCI Bus @ 33MHz

Parameter: Register Bits	Dimension	IDE Transfer Modes											
		PIO Modes						Multi-Word DMA Modes			Single-Word DMA Modes		
		0	1	2	3	4	5	0	1	2	0	1	2
Address Setup: 1F6h/176h[5:4]	Bit values in hex	2	1	1	0	0	0	0	0	0	0	0	0
	Timing in LCLKs[1]	3	2	2	1	1	1	1	1	1	1	1	1
	Enhanced IDE Spec in ns[2]	70	50	30	30	25	N/S	N/A	N/A	N/A	N/A	N/A	N/A
R/W Command Pulse: 1F0h/170h/1F1h/ 171h[7:4], Index-0/1	Bit values in hex	5	4	3	2	2	2	7	2	2	F	7	3
	Timing in LCLKs[1]	6	5	4	3	3	3	8	3	3	16	8	4
	Enhanced IDE Spec in ns[2]	165	125	100	80	70	N/S	215	80	70	480	240	120
R/W Recovery Time: 1F0h/170h/1F1h/ 171h[3:0], Index-0/1	Bit values in hex	9	4	0	0	0	0	6	0	0	D	5	1
	Timing in LCLKs[1]	11	6	2	2	1	0	8	1	0	15	7	3
	Enhanced IDE Spec in ns[2]	N/S	N/S	N/S	70	25	N/S	215	50	25	NS	NS	NS
Enhanced Mode: PCI Config. Addr. Offset 43h bits [7:6], [5:4], [3:2], or [1:0]	Bit values in hex	0	0	0	0	1	2	0	1	2	0	0	0
DRDY: 1F6h/176h[3:1]	Bit values in hex	0	0	0	0	0	0	0	0	0	0	0	0
	Timing in LCLKs[1]	2	2	2	2	2	2	2	2	2	2	2	2
Cycle Time	Timing in LCLKs	20	13	8	6	5	4	17	5	4	32	16	8
	Enhanced IDE Spec in ns[2]	600	383	240	180	120	N/S	480	150	120	960	480	240

N/S = Not Specified, N/A = Not Applicable

1. The actual timing (in LCLKs) that will be generated by the MIDE Module if the recommended bit values in hex are programmed.
2. The timing (in ns) as specified in the Enhanced IDE Specification.

Table 4-68 16-Bit Timing Parameters with PCI Bus @ 25MHz

Parameter: Register Bits	Dimension	IDE Transfer Modes											
		PIO Modes						Multi-Word DMA Modes			Single-Word DMA Modes		
		0	1	2	3	4	5	0	1	2	0	1	2
Address Setup: 1F6h/176h[5:4]	Bit values in hex	1	1	0	0	0	0	0	0	0	0	0	0
	Timing in LCLKs[1]	2	2	1	1	1	1	1	1	1	1	1	1
	Enhanced IDE Spec in ns[2]	70	50	30	30	25	N/S	N/A	N/A	N/A	N/A	N/A	N/A
R/W Command Pulse: 1F0h/170h/1F1h/ 171h[7:4], Index-0/1	Bit values in hex	4	3	2	1	1	1	5	1	1	C	5	2
	Timing in LCLKs[1]	5	4	3	2	2	2	6	2	2	12	6	3
	Enhanced IDE Spec in ns[2]	165	125	100	80	70	N/S	215	80	70	480	240	120
R/W Recovery Time: 1F0h/170h/1F1h/ 171h[3:0], Index-0/1	Bit values in hex	-6	2	0	0	0	0	4	0	0	9	3	0
	Timing in LCLKs[1]	8	4	2	2	1	0	6	1	0	11	5	2
	Enhanced IDE Spec in ns[2]	N/S	N/S	N/S	70	25	N/S	215	50	25	NS	NS	NS
Enhanced Mode: PCI Config. Addr. Offset 43h bits [7:6], [5:4], [3:2], or [1:0]	Bit values in hex	0	0	0	0	1	2	0	1	2	0	0	0
DRDY: 1F6h/176h[3:1]	Bit values in hex	0	0	0	0	0	0	0	0	0	0	0	0
	Timing in LCLKs[1]	2	2	2	2	2	2	2	2	2	2	2	2
Cycle Time	Timing in LCLKs	15	10	6	5	4	3	13	4	3	24	12	6
	Enhanced IDE Spec in ns[2]	600	383	240	180	120	N/S	480	150	120	960	480	240

N/S = Not Specified, N/A = Not Applicable

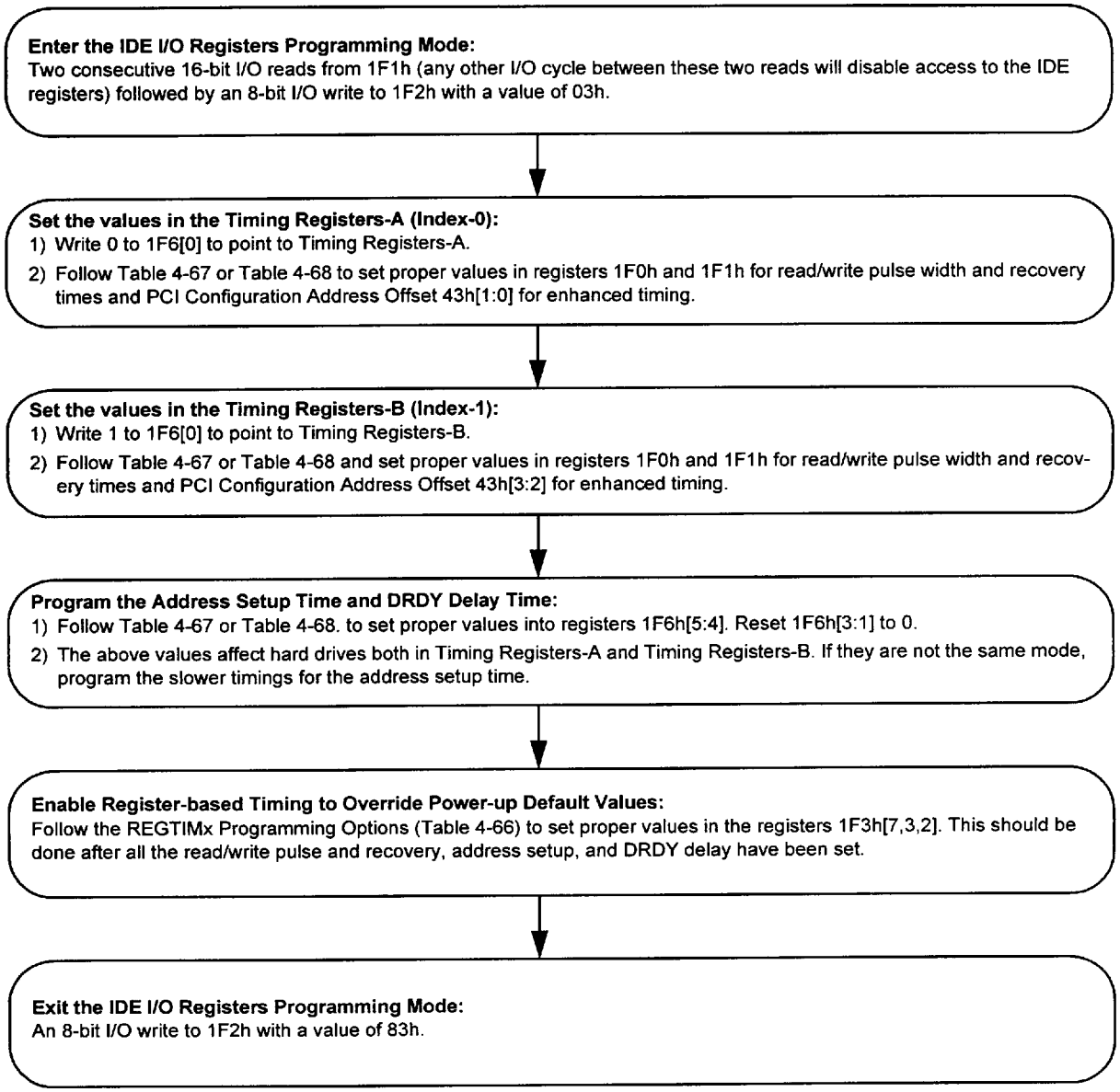
1. The actual timing (in LCLKs) that will be generated by the MIDE Module if the recommended bit values in hex are programmed.
2. The timing (in ns) as specified in the Enhanced IDE Specification.

Table 4-69 IDE Interrupt Routing Chart

Functions		82C558M PCI Configuration Address Offset Setting					Interrupt Controller			
		PCI Bus Intrf, Dev = 01h Func = 0	IDE Module, Dev = 01h Func = 1[1]				82C558M Interrupt Input		IDE Interrupts Output	
IDE Modes		4Fh[6]	04h[0]	40h[3]	40h[2]	09h[3:0]	Pin 135[2]	Pin 136[2]	Primary	Secondary
Primary	Secondary	IDE Module Enable	IDE I/O Enable	2nd IDE Disable	Native Mode Enable	Native/Legacy Mode	IRQ14 or DINT0	IRQ15 or DINT1	8259 or PCI INCT	8259 or PCI INCT
Disable		0	PCI Config. Register Space cannot be accessed				ISA IRQ14	ISA IRQ15	N/A	N/A
		1	0	x	x	xxxx				
Legacy[3]	Disable	1	1	1	0	xxxx	DINT0	ISA IRQ15	8259 IRQ14	N/A
		1	1	1	1	xx10				
Native	Disable	1	1	1	1	xx11	DINT0	ISA IRQ15	PIRQ3#[4]	N/A
Legacy[3]	Native	1	1	0	1	1110	DINT0	DINT1	8259 IRQ14	PIRQ3#[4]
Native	Legacy[3]	1	1	0	1	1011	DINT0	DINT1	PIRQ3#[4]	8259 IRQ15
Legacy[3]	Legacy[3]	1	1	0	0	xxxx	DINT0	DINT1	8259 IRQ14	8259 IRQ15
		1	1	0	1	1010				
Native	Native	1	1	0	1	1111	DINT0	DINT1	PIRQ3#[4]	PIRQ3#[4]

1. It assumes that Index FFh[4] of the 82C558M is set to 1, the IDE module is mapped as Device #01h, Function #1.
2. The ISA IRQ14 (ISA IRQ15) will not be available to the ISA bus if the on-board primary (secondary) IDE is enabled.
3. The 8259 IRQ14 (8259 IRQ15) will not be available for PIRQ[3:0]# if the on-board primary (secondary) IDE is enabled.
4. In Native mode, IDE interrupts are shared with PIRQ3# from the PCI bus. It is routed in the same way as PIRQ3# to the interrupt controller and is controlled by PCI Configuration Address Offsets 40h[11:9], 72h[7:1], and 50h[7:6] of the 82C558M (Device #01h, Function #0). Using this mode requires that the IDE device's Interrupt Service Routine support interrupt sharing.

Figure 4-43 MIDE Interface Primary Channel Programming Flow Chart



4.18 Special Cycles

4.18.1 System ROM BIOS Cycles

The 82C557M 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 82C558M 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.

4.18.2 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 82C557M. The Viper-M Chipset will generate INIT for a CPU shutdown cycle.

5.0 Register Descriptions

There are three broad classes of configuration registers in the Viper-M Desktop Chipset:

1) PCI Configuration Registers
2) System Control Registers
3) Power Management Registers.

The PCI Configuration Registers 00h-3Fh exist on both the 82C557M and the 82C558M. They pertain only to PCI operation and are accessed by special PCI configuration mechanisms.

The System Control Registers are distributed between the 82C557M and the 82C558M. Some of the System Control Registers are accessed by an indexing scheme and some are located in the device-specific PCI address space of the 82C557M and the 82C558M and are accessed via a PCI configuration mechanism.

The Power Management Registers are located in the 82C558M and are accessed by the same indexing scheme used to access the System Control Registers of the 82C557M. Table 5-1 summarizes the locations and access mechanisms for all the registers.

Note: In all the register format tables that follow, the most significant bit (MSB) corresponds to the upper Address Offset/Index. RO = Read Only, R/W = Read/Write, and W/O = Write Only.

Table 5-1 Register Locations and Access Mechanisms Summary

Parameter	PCI-Specific Configuration Registers		System Control Registers		PMU Registers
Location	82C557M (Address Offset 00h-3Fh)	82C558M (Address Offset 00h-3Fh)	82C557M (Index 00h-19h)	82C558M (Address Offset 40h-51h)	82C558M (Index E0h-FFh)
Classification	82C557M PCI Registers (Address Offset 00h-3Fh)	82C558M PCI Registers (Address Offset 00h-3Fh)	82C557M System Control Registers (Index 00h-19h)	82C558M System Control Registers (Address Offset 40h-51h)	82C558M (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 022h, Data to/from index through 024h	PCI Configuration Mechanism #1, Bus #0, Device #1, Function #0	Indexed Method: Index loaded in 022h, Data to/from index through 024h
Reference Section	Section 5.1.1	Section 5.2.1	Section 5.1.2	Section 5.2.2	Section 5.2.5

5.1 82C557M Register Space

The 82C557M has two register spaces:

1) PCI Configuration Space
2) System Control Register Space

5.1.1 82C557M PCI Configuration Register Space

The 82C557M's PCI Configuration Registers are used to implement PCI-specific functions and are accessed through:

- Configuration Mechanism #1 as:
  - Bus #0
  - Device #0
  - Function #0

Table 5-2 57VIN, 82C557M Vendor Identification Register - Address Offset 01h-00h

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

Table 5-3 57DID, 82C557M Device Identification Register - Address Offset 03h-02h

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

Table 5-4 57COM, 82C557M Command Register - Address Offset 05h-04h

Bit(s)	Type	Default	Function
15:10	R/W	0000 00	Reserved: 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: 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 82C557M.
3	RO	0	Special cycles: Must always = 0. The 82C557M does not respond to the PCI special cycle.
2	RO	1	Bus master operations: Must always = 1. This allows the 82C557M to perform bus master operations at any time.
1	RO	1	Memory access: Must always = 1. The 82C557M allows a PCI bus master access to memory at anytime.
0	RO	1	I/O access: Must always = 1. The 82C557M allows a PCI bus master I/O access at any time.

Table 5-5 57STAT, 82C557M 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 82C557M 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: Must be written to 0.

Table 5-6 57REV, 82C557M Revision Identification Register - Address Offset 08h

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

Table 5-7 57CLASS, 82C557M Class Code Register - Address Offset 0Bh-09h

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

Table 5-8 57RESV1, 82C557M Reserved Register 1 - Address Offset 0Ch

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

Table 5-9 57MLTMR, 82C557M Master Latency Timer Register - Address Offset 0Dh

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

Table 5-10 57HEAD, 82C557M Header Type Register - Address Offset 0Eh

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

Table 5-11 57BIST, 82C557M Built-In Self-Test Register, Address Offset 0Fh

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

Table 5-12 57RESV2, 82C557M Reserved Register 2 - Address Offset 3Fh-10h

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

Table 5-13 57MEMCTL, 82C557M Memory Control Register - Address Offset 43h-40h

Bit(s)	Type	Default	Function
31:16	R/W	0000h	Reserved: 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.
5:4	R/W	00	Reserved: Must be written to 0.
3	R/W	0	If this bit is set to 0, then control of writes being posted on the PCI bus is determined by the settings of Index 15h[5:4]. If this bit is set to 1, then no writes will be posted on the PCI bus except writes to the video memory and frame buffer areas.
2	R/W	0	Video frame buffer write post control: If Address Offset 43h-40h[3] = 0: 0 = Enable write posting to the video frame buffer 1 = Disable write posting to the video frame buffer If Address Offset 43h-40h[3] = 1: 0 = Disable write posting to the video frame buffer 1 = Enable write posting to the video frame buffer
1	R/W	0	Video memory (A0000h-BFFFFh) write post control: If Address Offset 43h-40h[3] = 0: 0 = Enable write posting to the video memory (A0000h-BFFFFh) 1 = Disable write posting to the video memory (A0000h-BFFFFh) If Address Offset 43h-40h[3] = 1: 0 = Disable write posting to the video memory (A0000h-BFFFFh) 1 = Enable write posting to the video memory (A0000h-BFFFFh)
0	R/W	0	I/O cycle write post control: 0 = Disable 1 = Enable

Table 5-14 57DRCCTL, 82C557M DRAM Data Latching Control Register - Address Offset 47h-44h

Bit(s)	Type	Default	Function
31:1	R/W	00h	Reserved
0	R/W	0	This bit determines the usage of the CPU clock to latch data coming into the 82C556M from the DRAM during reads of DRAM. 0 = CPU clock is not used to latch data into the 82C556M. 1 = CPU clock is used by the 82C556M to latch data into its internal data path during reads of DRAM. This bit must be set if EDO DRAMs are being used.

5.1.2 82C557M System Control Register Space

An indexing scheme is used to access the 82C557M's System Control Registers. Port 022h is used as the Index Register and Port 024h is the Data Register. Each access to a register within this space consists of:

1) a write to Port 022h, specifying the desired register in the data byte,
2. followed by a read or write to Port 024h with the actual register data.

The index resets after every access; so every data access (via Port 024h) must be preceded by a write to Port 022h even if the same register is being accessed consecutively.

Index 023h 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 5-15 Index 00h

Bit(s)	Type	Default	Function
57PYTHON, 82C557M 82C546/82C547 Compatible DRAM Configuration Register 1 (if Index 13h[7] = 0)			
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. These configurations maintain backward compatibility with the 82C546/82C547 chipset, and they do not allow the user too much flexibility. For 100% flexibility, Index 13h[7] should be set to 1.
57CMOD, 82C557M Byte Merge/Prefetch & Sony Cache Module Control Register (if Index 13h[7] = 1)			
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

Note: In Table 5-15 Index 00h can take on the functionality of two different registers. The setting of bit 7 at Index 13h decides on the functionality of this register.

Table 5-16 57DRMCTL, 82C557M 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, 82C557M 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 Index 0Fh[0] = 0:** / **If Index 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 07h) 10 = Test Mode 2; External Tag Read (Tag data read from Index 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, 82C557M 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 01 = X-3-3-3 10 = X-2-2-2 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 01 = 4-X-X-X 10 = 3-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 01 = X-3-3-3 10 = X-2-2-2 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 01 = 4-X-X-X 10 = 3-X-X-X 11 = 2-X-X-X* *Sync. SRAM double bank implementation does not support this timing

Table 5-19 57SHCTL1, 82C557M 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, 82C557M 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 **Bit Decode** — **Shadowed Area** 00 = Read/write PCI bus — Bits [7:6] = DC000h-DFFFFh 01 = Read from DRAM / write to PCI — Bits [5:4] = D8000h-DBFFFh 10 = Read from PCI / write to DRAM — Bits [3:2] = D4000h-D7FFFh 11 = Read from DRAM / write to DRAM — Bits [1:0] = D0000h-D3FFFh

Table 5-21 57SHCTL3, 82C557M 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 82C557M 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, 82C557M 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, 82C557M 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 82C557M keeps PEN# to the 82C558M asserted for all memory read cycles. If the 82C556M asserts MPERR# to the 82C558M then the 82C558M asserts NMI to the CPU. If this is not set, the assertion of MPERR# will not cause the 82C558M 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, 82C557M 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, 82C557M 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, 82C557M 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, 82C557M Extended DMA Register - Index 0Ch

Bit(s)	Type	Default	Function
7	R/W	0	Reserved: 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, 82C557M 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 82C557M 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 82C557M 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 82C557M asserts STOP# to the current master. Immediately after doing that, if this bit is set, the 82C557M claims the PCI bus and starts a dummy cycle. 0 = 82C557M does not claim the PCI cycle after it asserts STOP# 1 = 82C557M 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, 82C557M 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, 82C557M 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: 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	Write pulse duration control for operation with asynchronous SRAM: This bit is used when the write cycle takes the form of 3-X-X-X. 0 = Write pulse to RAMs are "one" CPU clock wide 1 = Write pulse is one-half a CPU clock cycle plus the delay of an internal delay line
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, 82C557M 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 82C557M 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, 82C557M Miscellaneous Control Register 2 - Index 11h

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved: 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 - 82C557M 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 82C558M or ISA master is source of the REFRESH# input 1 = Source of REFRESH# pulse is a 32KHz clock
6	R/W	0	Reserved: 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	82C556M 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 82C558M or the 82C556M. For better performance during PCI master writes, these parity bits should be generated by the 82C558M. If the option has been chosen to generate the MP[7:4] lines from the 82C558M, then the 82C557M can disable the 82C556M from generating the same. It does this via an encoded command that it sends to the 82C556M over the DBCOE#[1:0], MDOE#, and HDOE# lines 0 = Disable the 82C556M from generating the MP[7:4] lines during PCI master writes 1 = Enable the 82C556M 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 82C556M). If set to 1, there must be a pull-down on MP0 (strap option for 82C556M).

Table 5-34 57MEMDC1, 82C557M 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 Index 00h's register format - 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 Index 13h[7] is 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 Index 13h[7] is set: Refer to Index 13h[6:4] for decode settings.

Table 5-35 57MEMDC2, 82C557M Memory Decode Control Register 2 - Index 14h

Bit(s)	Type	Default	Function
7	R/W	0	Generation of DLE1# to the 82C556M during configuration cycles: The DLE1# signal may be used to latch data into the 82C556M during reads of DRAM. 0 = 82C556 (of Viper-DP Chipset) is used 1 = 82C556M (of Viper-M Chipset) is used
6:4	R/W	000	Full decode for logical bank 3 (RAS3#) if Index 13h[7] is set: Refer to Index 13h[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

43210	SIMM 0	SIMM 1	SIMM 2	SIMM 3	Total
00000	256K	256K			2M
00001	512K	512K	-----	-----	4M
00010	1M	1M	-----	-----	8M
00011	2M	2M		-----	16M
00100	4M	4M		-----	32M
00101	8M	8M			64M
00110	256K	256K	256K	256K	4M
00111	256K	256K	512K	512K	6M
01000	512K	512K	512K	512K	8M
01001	256K	256K	1M	1M	10M
01010	512K	512K	1M	1M	12M
01011	1M	1M	1M	1M	16M
01100	256K	256K	2M	2M	18M
01101	512K	512K	2M	2M	20M
01110	1M	1M	2M	2M	24M
01111	2M	2M	2M	2M	32M
10000	256K	256K	4M	4M	34M
10001	512K	512K	4M	4M	36M
10010	1M	1M	4M	4M	40M
10011	2M	2M	4M	4M	48M
10100	4M	4M	4M	4M	64M
10101	256K	256K	8M	8M	66M
10110	512K	512K	8M	8M	68M
10111	1M	1M	8M	8M	72M
11000	2M	2M	8M	8M	80M
11001	4M	4M	8M	8M	96M
11010	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 Index 13h[7], 13h[2:0, 6:4], and 14h[2:0, 6:4] for greater flexibility in DRAM configurations.

Table 5-37 57PCICY1, 82C557M 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: Must be written to 0.
0	R/W	0	PCI cycle, FRAME# timing control for pipelined cycles: During pipelined PCI cycles, the 82C557M 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 - 82C557M 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 82C557M. 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: 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	Synchronization between the PCI bus clock (LCLK) and the CPU clock (CLK): It should be noted that LCLK could be asynchronous to CLK also. This bit therefore implies that the PCI clock is either synchronous to the CPU clock with a skew not to exceed -2ns to 15ns, or that the PCI clock is asynchronous to the CPU clock. 0 = PCI clock is asynchronous to the CPU clock. 1 = PCI clock is synchronous to the CPU clock with a skew not to exceed -2ns to 15ns.
1	R/W	00	CPU to VL read access, 82C556M 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, 82C557M 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, 82C557M Tristate Control Register - Index 18h

Bit(s)	Type	Default	Function
7:6	R/W	0	Reserved: Must be written to 0.
6	R/W	0	Drive capability on the RAS lines going to DRAM: 0 = 4mA 1 = 16mA
5	R/W	0	Voltage selection for the CAS[7:0]# lines: 0 = 5.0V 1 = 3.3V
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, 82C557M 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 57SGCTL1, 82C557M Memory Shadow Control Register 1 - Index 1Ah

Bit(s)	Type	Default	Function
7	R/W	0	Reserved: Must be written to 0.
6:5	R/W	00	These bits allow the user to guarantee the CPU a percentage of the total available bus bandwidth, if he so desires. When these bits are programmed, the CPU is ensured of utilization of the bus for up to 4µs of every 15µs of operation of the system. This is achieved by not granting bus ownership to other requesting devices. 00 = No CPU bandwidth guarantee. 01 = CPU has a 1µs guarantee of utilization of the bus for every 15µs period. 10 = CPU has a 2µs guarantee of utilization of the bus for every 15µs period. 11 = CPU has a 4µs guarantee of utilization of the bus for every 15µs period.
4	R/W	0	Shadowing granularity: 0 = C8000h-DFFFFFh has a 16KB granularity. 1 = C8000h-DFFFFFh has an 8KB granularity.
3:0	R/W	0000	CX000h-CXFFFh read/write control if S8K = 1: These bits provide the read and write control of the region CX000h-CXFFFh for shadowing, if the granularity of shadowing is 8k. The "S8K" reference is to bit 4 of the same register, explained below, which provides the shadowing granularity. **Bit Decode** / **Controlled Area** 00 = Read/write PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM 01 = Read from DRAM / write to PCI Bits [3:2] = CE000h-CFFFFh Bits [1:0] = CA000h-CBFFFh

Table 5-43 57SGCTL2, 82C557M Memory Shadow Control Register 2 - Index 1Bh

Bit(s)	Type	Default	Function
7:0	R/W	00h	DX000h-DXFFFh read/write control if S8K = 1: These bits provide the read and write control of the region DX000h-DXFFFh for shadowing, if the granularity of the shadowing is 8k. The "S8K" reference is to bit 4 of the register whose Index is 1Ah, explained above, which provides the shadowing granularity. **Bit Decode** / **Controlled Area** 00 = Read/write PCI 10 = Read from PCI / write to DRAM 11 = Read from DRAM / write to DRAM 01 = Read from DRAM / write to PCI Bits [7:6] = DE000h-DFFFFh Bits [5:4] = DA000h-DBFFFh Bits [3:2] = D6000h-D7FFFh Bits [1:0] = D2000h-D3FFFh

Table 5-44 57EDOCTL, 82C557M EDO DRAM Control Register - Index 1Ch

Bit(s)	Type	Default	Function
7:2	R/W	0000 00	EDO DRAMs usage: Each bit is set to a 1 if the user is using EDO DRAMs in each of the available six banks. Bit 2 corresponds to Bank 0 and bit 7 corresponds to Bank 5, yielding a total of six banks that the user can populate. 0 = Standard page mode DRAM 1 = EDO DRAM
1	R/W	0	50MHz operating frequency: This bit is set by the user when the chipset is operating at a frequency of 50MHz. The setting of this bit could potentially improve DRAM access times even if the user is not using EDO DRAMs. 0 = The chipset is not being operated at 50MHz. 1 = The chipset is being operated at 50MHz.
0	R/W	0	CAS pulse width during DRAM accesses: If this bit is set to 0, then the width of the CAS pulse is determined by the setting of Index 01h[3]. This is the default setting. If this bit is set to a 1, the width of the pulse is one CPU clock if the Viper-M Chipset is operating at 50MHz. The Viper-M Chipset will operate at 50MHz by setting Index 1Ch[1] to 1. The CAS pulse width will also be one CPU clock wide if the Viper-M Chipset is interfaced to EDO DRAMs. This is done by setting Index 1Ch[7:2]. 0 = CAS pulse width determined by the setting of Index 01h[3]. 1 = CAS pulse width is one CPU clock wide if the Viper-M Chipset is operating at 50MHz or if the chipset is interfaced to EDO DRAMs.

Table 5-45 57RESV3, 82C557M Reserved Register 3 - Index 1Dh

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved: Must be written to 0.
5	R/W	0	DWE# timing selection: 0 = DWE# timing is normal 1 = DWE# is removed on clock earlier When using a buffered DWE# solution and the DRAM load is substantial, this bit may have to be set if the system begins to malfunction.
4	R/W	0	DRAM read leadoff cycle: 0 = Normal 1 = Reduced by 1 clock
3	R/W	0	DMA accesses from system memory: 0 = Enable 1 = Disable
2	R/W	0	PEN# pin functionality: 0 = PEN# (output only) 1 = MPERR# (input only) If parity checking needs to be done, then this bit has to be set to 1.
1	R/W	0	Accesses to B0000h-BFFFFh area during SMM mode: 0 = Accesses go to main memory 1 = Go out on to the PCI bus
0	R/W	0	Accesses to A0000h-AFFFFh area during SMM mode: 0 = Accesses go to main memory 1 = Go out on to the PCI bus

Table 5-46 57BOFF, 82C557M BOFF# Control Register - Index 1Eh

Bit(s)	Type	Default	Function
7:5	R/W	00	Reserved: Must be written to 0.
4	R/W	0	BOFF# generation if the PCI retry cycle is in A0000h-BFFFFh range: 0 = BOFF# is not generated if bit 3 of this register is set to 1 1 = BOFF# is generated if bit 3 of this register is set to 1 If bit 3 of this register is not set to 1, then the setting of this register has no effect.
3	R/W	0	In a situation where there is a PCI-to-PCI bridge in a system and that bridge supports write posting, the following deadlock condition can occur. The bridge posts data from a master on the secondary PCI bus into its FIFO. If at the same time the 82C557M is accessing the bridge as a target, then the bridge will tell the 82C557M to retry its request after it has serviced out its FIFO. This will result in a deadlock situation. 0 = There is no way to avert a deadlock situation if the write posting buffer on the PCI-to-PCI bridge has been enabled. 1 = The 82C557M asserts BOFF# to the CPU in the event that this deadlock situation occurs. This bit needs to be set to 1 if a DEC 21050 PCI-to-PCI bridge (or a similar chip) is being used.
2	R/W	0	Reserved: Should always be written 0.
1	R/W	0	When set to 1, this will disable PCI bursting if BE[7:4]# and/or BE[3:0]are not all 0.
0	R/W	0	Reserved: Should always be written 0.

Table 5-47 57RSVD1, 82C557M Reserved Register - Index 1Fh

Bit(s)	Type	Default	Function
7:0	R/W	00h	Reserved: Should always be written 0.

5.1.3 82C557M I/O Registers

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

Table 5-48 57REFPG, 82C557M Refresh Page Register (Write-Only) - Index 8Fh

Bit(s)	Type	Default	Function
7:0	WO	00h	Values programmed into this register gets driven on LA[23:17] during refresh cycles. It is recommended that the user write 00h into this register.

57NVMADR, 82C557M Configuration/NVM Address Register - Index CF8h

- 32-bit writes to this register are captured by the 82C557M'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, 82C557M 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 82C558M Register Space

The 82C558M has four register spaces:

1) PCI Configuration Register Space
2) System Control Register Space
3) I/O Register Space
4) Power Management Register Space

5.2.1 82C558M PCI Configuration Register Space

The 82C558M's PCI Configuration Registers to implement PCI-specific functions are accessed through:

- Configuration Mechanism #1 as:
  - Bus #0
  - Device #1
  - Function #0

Table 5-49 58VIN, 82C558M Vendor Identification Register - Address Offset 01h-00h

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

Table 5-50 58DID, 82C558M Device Identification Register - Address Offset 03h-02h

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

Table 5-51 58COM, 82C558M Command Register - Address Offset 05h-04h

Bit(s)	Type	Default	Function
15:10	R/W	0000 00	Reserved: 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: 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 82C558M.
3	R/W	0	Special cycles: 0 = Disable 1 = Enable 82C558M 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 82C558M allows a PCI bus master access to memory at any time.
0	RO	1	I/O access: Must always = 1. The 82C558M allows a PCI bus master to access I/O at any time.

Table 5-52 58STAT, 82C558M 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 82C558M 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
6:0	R/W	0000 000	Reserved: Must always be written to 0.

Table 5-53 58REV, 82C558M Revision Identification Register - Address Offset 08h

Bit(s)	Type	Default	Function
7:0	RO	10h	Revision Identification Register

Table 5-54 58CLASS, 82C558M Class Code Register - Address Offset 0Bh-09h

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

Table 5-55 58RESV1, 82C558M Reserved Register 1 - Address Offset 0Ch

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

Table 5-56 58MLTMR, 82C558M Master Latency Timer Register - Address Offset 0Dh

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

Table 5-57 58HEAD, 82C558M Header Type Register - Address Offset 0Eh

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

Table 5-58 58BIST, 82C558M Built-In Self-Test Register - Address Offset 0Fh

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

Table 5-59 58RESV2, 82C558M Reserved Register 2- Address Offset 3Fh-10h

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

5.2.2 82C558M System Control Register Space

These registers are accessed via the same configuration mechanism used to access the PCI Configuration Registers of the 82C558M (i.e., through Configuration Mechanism #1 as 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 82C558M's PCI Configuration Register Address Space.

Table 5-60 58KBDCTL, 82C558M 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 PIRQx# has been triggered: **Bit Decode** / **Triggered PIRQx# Signal** 000 = IRQx* / Bits [11:9] = PIRQ3# 001 = IRQ5 / Bits [8:6] = PIRQ2# 010 = IRQ9 / Bits [5:3] = PIRQ1# 011 = IRQ10 / Bits [2:0] = PIRQ0# 100 = IRQ11 101 = IRQ12 110 = IRQ14 111 = IRQ15 *Selection controlled by Address Offset 50h[7:0].

Table 5-61 58MISC1, 82C558M Miscellaneous Control Register 1 - Address Offset 43h-42h

Bit(s)	Type	Default	Function
15	R/W	0	ISA IRQ14 recognition control: If the user desires to use IDE on the PCI bus, then the ISA IRQ14 signal needs to be blocked. 0 = ISA IRQ14 is honored 1 = ISA IRQ14 is ignored
14	R/W	0	ISA IRQ15 recognition control: If the user desires to use secondary IDE on the PCI bus, then the ISA IRQ15 signal needs to be blocked. 0 = ISA IRQ15 is honored 1 = ISA IRQ15 is ignored
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 **Note:** When bits [11:10] take on the combination of 11, all back-to-back cycles will be delayed by 12 AT clocks. This is different from the combinations of 00 and 01 because in the latter case, the delay will be inserted only when an I/O access is followed by a second I/O access with no other type of access occurring in between (e.g., a memory access).
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 IRQx: 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). **Bit Decode** 0 = Edge 1 = Level **Level Triggered IRQx** Bit 7 = IRQ15, Bit 6 = IRQ14, Bit 5 = IRQ12, Bit 4 = IRQ11, Bit 3 = IRQ10, Bit 2 = IRQ9, Bit 1 = IRQ5
0	R/W	0	DREQ6 pin functionality: DREQ6 is pin-wise programmable with EPMI#. This bit determines the functionality that this pin takes on. 0 = DREQ6 1 = EPMI0#

Table 5-62 58PINFU1, 82C558M Pin Functionality Register 1 - Address Offset 45h-44h

Bit(s)	Type	Default	Function
15	R/W	0	DREQ5 pin functionality: 0 = DREQ5 1 = DREQ0 takes on the functionality of DRQ0/5 or DRQ5
14	R/W	0	PGNT2# pin functionality: This bit determines the group-wise functionality of the PGNT2#/ GPCS1# pin. 0 = PGNT2# 1 = GPCS1# Note: If this pin is programmed as GPCS1# it should be pulled high externally.
13	R/W	0	PIRQ3# pin functionality: This pin can take on any of the these functionalities - PIRQ3#, LRDY#, or EPMI2#. PIRQ3# and LRDY# functionalities are group-wise programmable, and those two functionalities are pin-wise programmable with EPMI2#. 0 = Controlled by bits [1:0] and Address Offset 51h-50h[12] = 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 = EPMI2# - If this bit is set and Address Offset 51h-50h[12] = 0, then this pin takes on the functionality of EPMI2#, 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 group-wise programmable. 0 = Controlled by bits [1:0] and Address Offset 51h-50h[11] = 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: 00 = PIRQ1# 01 = OUT0 The rest are reserved.
9:8	R/W	00	PIRQ0# pin functionality: This pin can take on any of the following functionalities - PIRQ0#, EPMI1#. PIRQ0# and EPMI1# are pin-wise programmable. 00 = PIRQ0# 01 = EPMI1# 1X = Reserved
7:6	R/W	00	DACK3# pin functionality: This pin can take on the following functionalities - DACK3#, EDACK2, or DACK7#. DACK3# and EDACK2 are group-wise programmable, and both of them are pin-wise 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, 82C558M 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 group-wise programmable, and both of them are pin-wise 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#,EDACK0,DACK5# pin functionality: This field of bits determines the functionality of pin number 108 on the 82C558M in conjunction with bits [1:0] of this register. 0X = Controlled by bits [1:0] 10 = DACK5# 11 = Reserved (the user should not program the bits to have this combination)
1:0	R/W	00	DACK/MP/PIRQ[3:2]# pin functionality: These bits determine the functionality of the group-wise 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]# Pin-wise, these functions may be overridden by GPCS[x]#, EPMI#, and DACK[7:5]# (for DACK[3,1,0]).

Table 5-63 58CYCTL1, 82C558M 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: **Preempt after unserviced request pending for:** 000 = No preemption 100 = 36 LCLKs 001 = 260 LCLKs 101 = 20 LCLKs 010 = 132 LCLKs 110 = 12 LCLKs 011 = 68 LCLKs 111 = 5 LCLKs
7	R/W	0	DMA/ISA access to PCI/VL slave: 0 = Never 1 = When LMEM# is not asserted **Note:** Master Retry always unmasked after 16 LCLKs.
6	R/W	0	XDIR control: 0 = XDIR is achieved for accesses to/from ROM, keyboard controller, RTC, or NVRAM 1 = XDIR achieved only during access to/from ROM or NVRAM
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:** Reserved: Must be written to 0.
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 5-64 58PINFU2, 82C558M Pin Functionality Register 2 - Address Offset 49h-48h

Bit(s)	Type	Default	Function
15	R/W	0	IRQ15 pin functionality: 0 = IRQ15 1 = Reserved
14:13	R/W	00	IRQ12 pin functionality: This pin can take on the functionality of IRQ12 or MPIRQ2#/3#. These functionalities are pin-wise 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 pin-wise 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 pin-wise 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 pin-wise programmable. 0 = IRQ4 1 = MIRQ4/6
9:8	R/W	00	DREQ7,EPMI1# pin functionality: 00 = DREQ7 01 = EPMI1# 1X = Reserved
7:6	R/W	00	DREQ3 pin functionality: This pin can take on the functionality of DREQ3, DREQ3/7, or DREQ7. These functionalities are pin-wise 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 pin-wise 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 pin-wise programmable. 00 = DREQ0 10 = DREQ5 01 = DREQ0/5 11 = Reserved
1:0	R/W	00	PREQ1# pin functionality: Bit 1 of this register pair needs to be set a 0 for the pin to provide the functionality of PREQ1#. All other combinations on these bits are reserved. 0X = PREQ1# 1X = Reserved

Table 5-65 58ROMCS, 82C558M 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 5-66 58RESV3, 82C558M Reserved Register 3 - Address Offset 4Dh-4Ch

Bit(s)	Type	Default	Function
15:0	R/W	00h	Reserved: Must be written to 0.

Table 5-67 58MISC2, 82C558M Miscellaneous Control Register 2 - Address Offset 4Fh-4Eh

Bit(s)	Type	Default	Function
15	R/W	0	Reserved: Must be written to 0.
14	R/W	0	IDE support. 0 = Disable 1 = IDE functionality enabled
13	RO	1	82C558M silicon revision usage in system: 0 = 82C558M Rev 0 1 = 82C558M Rev 1
12	R/W	0	Extend the cycle on the AT bus: Usually a cycle on the ISA bus, during 8-bit accesses, consists of a command phase that is one clock long, followed by four wait states, and then finally followed by a one clock long data phase. Setting this bit to a 0 will cause the number of wait states inserted in the cycle to increase by one to five clocks, making the duration of the entire cycle to seven clocks instead of six clocks. 0 = One wait state is inserted in AT commands. 1 = No wait states are inserted in AT commands.
11	R/W	0	Reserved: Must be written to 0.
10	R/W	0	Reserved: Must be written to 0.
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 group-wise programmable, and all three of them are pin-wise programmable with GPCS2#. 0 = The functionality is decided by the group-wise 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: 0 = 14.318MHz clock is used for demultiplexing the IRQs 1 = LCLK clock is used for demultiplexing the IRQs
7:4	R/W	0000	Reserved: 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 5-68 58TRIG, 82C558M Trigger Control Register - Address Offset 51h-50h

Bit(s)	Type	Default	Function
15	R/W	0	32KHZ,PREQ3# pin functionality: 0 = 32KHz 1 = PREQ3# pin
14	R/W	0	ZEROWS#,PGNT3# pin functionality: 0 = ZEROWS# 1 = PGNT3#
13	R/W	0	DACK2#,EDACKEN#,GPCS2# pin functionality: 0 = DACK2# 1 = GPCS2#
12	R/W	0	PIRQ3#,LRDY#,EPMI#,GPCS1# pin functionality: 0 = PIRQ3# 1 = GPCS1#
11	R/W	0	PCIRQ2#,LDEV#,GPCS0# pin functionality: 0 = PIRQ2# 1 = GPCS0#
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 5-69 58INTMCL, 82C558M Interrupt Multiplexing Control Register - Address Offset 53h-52h

Bit(s)	Type	Default	Function
15	R/W	0	Pin functionality: 0 = IRQ3, IRQ5, IRQ7, IRQ9, and IRQ11 function as they are 1 = IRQ3 functions as MIRQ3/5, IRQ5 functions as MIRQ7/9, IRQ6 functions as MIRQ11/15, IRQ7 functions as EPMI1#, IRQ9 functions as EPMI2#, and IRQ11 functions as GMIRQ
14	R/W	0	PREQ2#,EPMI0# pin functionality: 0 = PREQ2# 1 = EPMI0#
13	R/W	0	IRQ1 latching: 0 = Disable 1 = Enable
12	R/W	0	IRQ12 latching: 0 = Disable 1 = Enable
11	R/W	0	DACKEN#: 0 = Active low 1 = Active high
10	R/W	0	System bus ownership: 0 = External device cannot own system bus 1 = External device can own system bus When this bit is set to 1, preemption of PGNT0# is disabled, EOP is tristated, BALE is driven high, and internal decoding of the DMA subsystem registers is blocked.
9	R/W	0	Locking of flash ROM: Setting this bit to a 1 will block writes to flash ROM until the next RESET pulse. 0 = Disable 1 = Enable
8	R/W	0	Reserved: Must be written to 0.
7	R/W	0	Triggering for IRQ6: 0 = Edge triggered 1 = Level triggered
6:3	R/W	00	IRQ signal generation when GMIRQ is triggered: 0000 = Disabled 1000 = Reserved 0001 = Reserved 1001 = IRQ9 0010 = Reserved 1010 = IRQ10 0011 = IRQ3 1011 = IRQ11 0100 = IRQ4 1100 = IRQ12 0101 = IRQ5 1101 = Reserved 0110 = IRQ6 1110 = IRQ14 0111 = IRQ7 1111 = IRQ15
2	R/W	0	Reserved: Must be written to 1.
1	R/W	0	Priority scheme: 0 = Disable 1 = Enable A setting of 1 on this bit will employ a priority scheme that guarantees higher priority for PCI masters during arbitration over DMA and ISA masters for the first 7µs interval after every refresh cycle.
0	R/W	0	Concurrent refresh and IDE cycle: 0 = Disable 1 = Enable ISA devices that rely on accurate refresh addresses for proper operation should disable this bit.

Table 5-70 58DMACTL, 82C558M DMA Control Register - Address Offset 55h-54h

Bit(s)	Type	Default	Function
15:0	R/W	00h	Reserved: Must be written to 0.

Table 5-71 Pin Multiplexing Chart 1

Default Pin Names	Address Offset																	
	42h	44h								48h						4Fh	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 5-72 Pin Multiplexing Chart 2

Default Pin Names	Address Offset 42h													Address Offset 51h						Address Offset 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
L: Level trigger

Table 5-73 Pin Multiplexing Chart 3

Default Pin Names	Address Offset 41h-40h								Address Offset 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#)												
PIRQ0# (PCI_IRQ0#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[A]	IRQ4[A]	IRQ3[A]	Disable
PIRQ1# (PCI_IRQ1#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[B]	IRQ4[B]	IRQ3[B]	Disable
PIRQ2# (PCI_IRQ2#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[C]	IRQ4[C]	IRQ3[C]	Disable
PIRQ3# (PCI_IRQ3#)[E]	IRQ15	IRQ14	IRQ12	IRQ11	IRQ10	IRQ9	IRQ5	Disable	IRQ7[D]	IRQ4[D]	IRQ3[D]	Disable

*: Default
A: Only valid with Address Offset: 40h[11:9] - 000.
B: Only valid with Address Offset: 40h[8:6] - 000.
C: Only valid with Address Offset: 40h[5:3] - 000.
D: Only valid with Address Offset: 40h[2:0] - 000.
E: Refer to Table 5-72 for additional muxing options on IRQ6 and IRQ12.
F: Address Offset: 41h-40h[11:9] for PIRQ0#, 41h-40h[8:6] for PIRQ1#, 41h-40h[5:3] for PIRQ2#, 41h-40h[2:0] for PIRQ3#.
G: Address Offset: 50h[7:6] for PIRQ0#, 50h[5:4] for PIRQ1#, 50h[3:2] or PIRQ2#, 50h[1:0] for PIRQ3#.

5.2.3 82C558M I/O Registers

Table 5-74 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 82C558M 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 5-75 PS/2 Reset Control - Port 92h

Bit(s)	Type	Default	Function
7:2		0000 00	Reserved
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 82C557M propagates the CPU Stop Grant cycle as a configuration write to this register. The BIOS should never write to this configuration space. The 82C558M itself should ignore any write to this space unless it has driven the STPCLK# active.

Table 5-76 DMA System I/O Register - Address 40Bh

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved: Must be written to 0.
5:4	R/W	00	A setting of 00 on these bits gives compatible timing on DMA transfers. When set to 11, Type F DMA timing can be obtained.
3:2	R/W	00	Reserved: Must be written to 0.
1:0	R/W	00	DMA channel select: 00 = Channel 0 10 = Channel 2 01 = Channel 1 11 = Channel 3

Table 5-77 DMA System I/O Register - Address 483h-481h

Bit(s)	Type	Default	Function
23:16	R/W	00h	DMA Channel 1 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.
15:8	R/W	00h	DMA Channel 3 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.
7:0	R/W	00h	DMA Channel 2 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.

Table 5-78 DMA System I/O Register - Address 48Bh-487h

Bit(s)	Type	Default	Function
39:32	R/W	00h	DMA Channel 5 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.
31:24	R/W	00h	DMA Channel 7 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.
23:16	R/W	00h	DMA Channel 6 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.
15:8	R/W	00h	Reserved: Must be written to 0.
7:0	R/W	00h	DMA Channel 0 high page register: These bits from the eight most significant address bits AD[31:24] when formatting the address for a DMA transfer.

Table 5-79 DMA System I/O Register - Address 4D6h

Bit(s)	Type	Default	Function
7:6	R/W	00	Reserved: Must be written to 0.
5:4	R/W	00	A setting of 00 on these bits gives compatible timing on DMA transfers. When set to 11, Type F DMA timing can be obtained.
3:2	R/W	00	Reserved: Must be written to 0.
1:0	R/W	00	DMA channel select: 00 = Channel 4 10 = Channel 6 01 = Channel 5 11 = Channel 7

5.2.4 82C558M Internal Integrated 82C206 Register Descriptions

The internal integrated 82C206 registers are accessed by indexing I/O Registers 022h and 023h. Index Register 01h should be set to the default value of C0h.

Following Table 5-80 are tables that explain the subsystem registers of the 82C206.

Table 5-80 Configuration Register (Index Port 022h, Data Port 023h) - 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 82C558M'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 82C558M'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 5-81 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 5-82 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 5-83 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 5-84 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 5-85 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 5-86 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 5-87 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 5-88 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 5-89 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 5-90 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 5-91 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 5-92 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 5-93 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 5-94 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 5-95 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 5-96 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 5-97 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 5-98 Poll Vector - Address 20h (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.

The remaining registers listed in this section (Interrupt Controller Subsystem) are accessed with Port 022h used as the Index Register and Port 024h as the Data Register. Each access to a register within this space consists of:

1) a write to Port 022h, specifying the desired register in the data byte,
2. followed by a read or write to Port 024h with the actual register data.

The index resets after every access; so every data access (via Port 024h) must be preceded by a write to Port 022h even if the same register is being accessed consecutively.

These registers shadow the write-only registers in the Interrupt Controller Subsystem. The values programmed into the "write-only" registers can be read back by accessing these registers.

Table 5-99 Interrupt Controller Subsystem Shadow Registers

Bit(s)	Type	Function
Index 80h: PIC1ICW1 Register		
7:0	RO	Reads back the contents of the ICW1 Register at Address 020h.
Index 81h: PIC1ICW2 Register		
7:0	RO	Reads back the contents of the ICW2 Register at Address 021h.
Index 82h: PIC1ICW3 Register		
7:0	RO	Reads back the contents of the ICW3 Register at Address 021h.
Index 83h: PIC1ICW4 Register		
7:0	RO	Reads back the contents of the ICW4 Register at Address 021h.
Index 84h: Reserved		
Index 85h: PIC1OCW2 Register		
7:0	RO	Reads back the contents of the OCW2 Register at Address 020h.
Index 86h: PIC1OCW3 Register		
7:0	RO	Reads back the contents of the OCW3 Register at Address 020h.
Index 87h: Reserved		
Index 88h: PIC2ICW1 Register		
7:0	RO	Reads back the contents of the ICW1 Register at Address 0A0h.
Index 89h: PIC2ICW2 Register		
7:0	RO	Reads back the contents of the ICW2 Register at Address 0A1h.
Index 8Ah: PIC2ICW3 Register		
7:0	RO	Reads back the contents of the ICW3 Register at Address 0A1h.
Index 8Bh: PIC2ICW4 Register		
7:0	RO	Reads back the contents of the ICW4 Register at Address 0A1h.
Index 8Ch: Reserved		
Index 8Dh: PIC2OCW2 Register		
7:0	RO	Reads back the contents of the OCW2 Register at Address 0A0h.
Index 8Eh: PIC2OCW3 Register		
7:0	RO	Reads back the contents of the OCW3 Register at Address 0A0h.

5.2.4.3 Counter/Timer Subsystem

Table 5-100 Control Word Format (Write Only) - Address 043h

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 5-101 Counter Latch Command Format (Write Only) - Address 043h

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 5-102 Read-Back Command Format (Write Only) - Address 043h

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 5-103 Status Format - Address 043h

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.

Table 5-104 Counter 0 Count Value - Address 40h

Bit(s)	Type	Function
15:0	W	Contains the initial count value of Counter 0.

Table 5-105 Counter 1 Count Value - Address 41h

Bit(s)	Type	Function
15:0	W	Contains the initial count value of Counter 1.

Table 5-106 Counter 2 Count Value - Address 42h

Bit(s)	Type	Function
15:0	W	Contains the initial count value of Counter 2.

The remaining registers listed in this section (Counter/Timer Subsystem) are accessed with Port 022h used as the Index Register and Port 024h as the Data Register. Each access to a register within this space consists of:

1) a write to Port 022h, specifying the desired register in the data byte,
2. followed by a read or write to Port 024h with the actual register data.

The index resets after every access; so every data access (via Port 024h) must be preceded by a write to Port 022h even if the same register is being accessed consecutively.

These registers shadow the write-only registers in the Counter/Timer Subsystem. The values programmed into the "write-only" registers can be read back by accessing these registers.

Table 5-107 Counter/Timer Subsystem Shadow Registers

Bit(s)	Type	Function
Index 90h: CTSC0LB Register		
7:0	RO	Reads back the lower byte value of Counter 0 at Address 40h.
Index 91h: CTSC0HB Register		
7:0	RO	Reads back the higher byte value of Counter 0 at Address 40h.
Index 92h: CTSC1LB Register		
7:0	RO	Reads back the lower byte value of Counter 1 at Address 41h.
Index 93h: CTSC1HB Register		
7:0	RO	Reads back the higher byte value of Counter 1 at Address 41h.
Index 94h: CTSC2LB Register		
7:0	RO	Reads back the lower byte value of Counter 2 at Address 42h.
Index 95h: CTSC2HB Register		
7:0	RO	Reads back the higher byte value of Counter 2 at Address 42h.
Index 96h: Byte Pointer Register		
7:0	RO	The contents of this register is the Byte 2 pointer value.

5.2.5 82C558M Power Management Registers

The Power Management Registers are located in the 82C558M and are accessed by an indexing scheme.

Port 022h is used as the Index Register and Port 024h is the Data Register. Each access to a control register consists of:

1) a write to Port 022h, specifying the desired register in the data byte,
2) followed by a read or write to Port 024h with the actual register data.

The index resets after every access; so every data access (via Port 024h) must be preceded by a write to Port 022h even if the same register is being accessed consecutively. Note that all reserved bits are to be set to zero unless otherwise noted.

The tables that follow give the Power Management Register's bit formats.

Table 5-108 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 5-109 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 10 = 1.94s CLK period 01 = 12.25ms 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	This bit is used in conjunction with bit 5 of this same register to determine the polarity of the EPMI0# signal. If this bit is programmed to be a 0, then the polarity of the EPMI0# signal is determined by the setting of bit 5. On the other hand, if bit 3 is programmed to be a 1, then regardless of the setting of bit 5, EPMI0# will be triggered on the falling edge as well as the rising edge. However, care must be taken to ensure that the level on EPMI0# stays unchanged for a minimum period of 20ms in order for the Viper-M Chipset to recognize the level change and act on it. In addition, since this pin can also take on the functionality of DREQ6 (pin number 122 on the 82C558M), the polarity on DREQ6, when this pin is used for the DREQ6 functionality, will be active low when this bit is a 0 and will be active high when this bit is a 1. When the combination of bits 3 and 5 is 00, then a falling edge on EPMI0# will be recognized as an interrupt. When the combination on the aforementioned bits is 01, then a rising edge on the EPMI0# signal will be recognized as an interrupt. When the combination of the bits is 1X, then EPMI0# will be recognized on a transition to the opposite level. However, as noted earlier, the new level to which EPMI0# switches to will have to be maintained for a minimum period of 20ms for the internal circuit to recognize the transition and act on it.
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 5-110 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 5-111 58IRQ1, IRQ Event Enable Register 1 - Index E3h

Bit(s)	Type	Default	Function
7:3	R/W	0000 0	IRQx monitor: Disables/enables IRQx from/to becoming a reload GET / wake-up event. **Bit Decode** — **IRQx Signal** 0 = Disables; 1 = Enables Bit 7 = IRQ7, Bit 6 = IRQ6, Bit 5 = IRQ5, Bit 4 = IRQ4, Bit 3 = IRQ3
2	R/W	0	IRQ[15-0] deglitch select: 0 = No deglitch 1 = Sample the IRQ[15-0] lines using an internal clock and reduce glitching
1:0	R/W	00	IRQx monitor: Disables/enables IRQx from/to becoming a reload GET / wake-up event. **Bit Decode** — **IRQx Signal** 0 = Disables 1 = Enables Bit 1 = IRQ1 Bit 0 = IRQ0

Table 5-112 58IRQ2, IRQ Event Enable Register 2 - Index E4h

Bit(s)	Type	Default	Function
7:0	R/W	0000 0000	IRQx monitor: Disables/enables IRQx from/to becoming a reload GET / wake-up event. **Bit Decode** — **IRQx Signal** 0 = Disables 1 = Enables Bit 7 = IRQ15, Bit 6 = IRQ14, Bit 5 = IRQ13, Bit 4 = IRQ12, Bit 3 = IRQ11, Bit 2 = IRQ10, Bit 1 = IRQ9, Bit 0 = IRQ8

Table 5-113 58DREQ, DREQ Event Enable Register - Index E5h

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

Table 5-114 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 Index 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 5-115 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: Must be written to 0.
3	R/W	0	IO/MEM (non-system memory) selection for the programmable address ranges specified in Index E8h, E9h: This bit determines whether the address specified in Index E8h and E9h is an I/O or non-system memory address. 0 = The address specified in Index E8h and E9h is an I/O address. 1 = The address specified in Index 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 5-116 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 5-117 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 5-118 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 5-119 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 5-120 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 5-121 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 Index E8h and E9h), to become a wake-up event. If Index 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 Index 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 Index 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 Index 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 Index 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 Index 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 Index 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 Index 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 5-122 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: 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# = Reserved 110 = STPCLK# = Reserved 111 = STPCLK# = Reserved

Table 5-123 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: 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 Index F4h-F5h and F6h-F7h. 0 = Disable GPCS#1, GPCS#2 generation 1 = Enable GPCS#1, GPCS#2 generation
3	R/W	0	Reserved: Must be written to 0.
2	R/W	0	Clock selection bit to initiate a hardware power-down condition. 0 = 14MHz clock is selected to initiate the hardware PPWRL# signal. 1 = 32KHz clock is selected to initiate the hardware PPWRL# signal. **Note:** If CLK SYNTH is in the power-down mode, then a clock that is still running will have to be selected in order for the wake-up event to trigger a hardware PPWRL# signal.
1	R/W	0	Read current count of Timers: This bit has to be set to a 0 if the user wishes to read the current count of the various timers used in the power management of the Viper-M Chipset. It may be recalled that the user needs to program various initial counts in the registers whose Index Addresses are E0h, E1h, E2h, EDh, F0h, F1h, F2h, FCh, FDh, and FEh. The current count value of the timers associated with these registers can be read back from these registers if bit 1 of this register is set to a 0. If bit 1 of this register is set to a 1, reads from the aforementioned registers will return the originally programmed initial count of the respective registers. 0 = Return the current count value on reads to registers. 1 = Return the originally programmed value into the register on reads.
0	R/W	0	Reserved: Must be written to 0.

Table 5-124 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 (Index 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 (Index 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 5-125 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 5-126 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 5-127 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 Index 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 Index 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 5-128 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 5-129 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 5-130 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 5-131 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

Table 5-132 58RSVD1, 82C558M Reserved Register 1 - Index FBh-FAh

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

Table 5-133 58PMCTL1, Power Management Control Register 1 - Index FCh

Bit(s)	Type	Default	Function
7	R/W	0	EPMI1# becomes GREEN event: 0 = Disable EPMI1# from triggering a GREEN event. 1 = Enable triggering of a GREEN event when EPMI1# is active. This bit and bit 6 of this register cannot be set to a 1 at the same time, as the EPMI1# signal is interpreted differently for each of the settings. During reads of the register (reads of the register is primarily done to determine status) if bit 7 returns a 1, the user should interpret that as a GREEN event is invoked by the EPMI1# signal and if a 0 is returned, the user should interpret that as a GREEN event not being invoked by the EPMI1# signal.
6	R/W	0	Reload wake-up GREEN state timer whenever EPMI1# is active: During writes: 0 = Disable the EPMI1# signal from loading the wake-up GREEN state timer. 1 = Enable the loading of the wake-up GREEN state timer on the EPMI1# signal. During reads: 0 = GREEN event not invoked by EPMI1#. 1 = GREEN event has been invoked by EPMI1#. As this interpretation of the EPMI1# signal is different from when bit 7, above, is set to a 1, the user should never set both bits 7 and 6 of this register to a 1 at the same time.
5	R/W	0	EPMI1# polarity: This bit determines the polarity of the EPMI1# signal in conjunction with bit 4 of this register. When this bit is set to a 0, polarity of the EPMI1# signal is determined by the setting on bit 4 of this register. However, if this bit is set to a 1, then regardless of the setting of bit 4 of this register a transition on the EPMI1# signal will trigger an event based on the settings of bits 7 and 6 (explained above). It should be noted that when this bit is set to a 1, it is the user's responsibility to guarantee that the transition on EPMI1# signal lasts for a minimum period of 20ms before it departs from the current logic level. 0 = EPMI1# polarity will be determined by the setting on bit 4 of this register. 1 = An event will be triggered, dictated by the settings on bits 7 and 6 of this register on a transition of the EPMI1# signal, provided timing requirements are met on the EPMI1# signal.
4	R/W	0	EPMI1# polarity: This bit will determine the polarity of the EPMI1# signal as long as bit 5 of this register is not set. Listed is the combination of bits 5 and 4 of this register and the polarity on the EPMI1# signal that they define. If the combination is 00, then a falling edge on the EPMI1# signal will cause a triggering of an event. If the combination is 01, then a rising edge on the EPMI1# signal will cause a trigger of the event. If the combination is 1X, then a transition on EPMI1# signal will cause a trigger of the event as long as the timing requirements mentioned above are met.
3	R/W	0	EPMI1# debounce effect: This bit is used to select the debounce effect on the EPMI1# signal. Normally, external interrupts to the power management unit can cause glitches in the system. The user can avoid this glitching by setting this bit to a 1. Note that to avoid the glitching, the EPMI1# signal needs to stay stable for a minimum period of 5ms. 0 = No debounce effect has been enabled on the EPMI1# signal. 1 = Debounce effect has been selected on the EPMI1# signal. (Again, the EPMI1# signal should remain stable for a minimum period of 5ms for proper recognition.)
2:1	R/W	00	Reserved: Should be written to 0.
0	R/W	0	Pin functionality: 0 = Pin numbers 117, 140, and 143 all function as EPMI0#. 1 = Pin number 117 functions as EPMI0#, pin number 140 functions as EPMI1#, and pin number 143 functions as EPMI2#.

Table 5-134 58PMCTL2, Power Management Control Register 2 - Index FDh

Bit(s)	Type	Default	Function
7	R/W	0	EPMI2# becomes GREEN event: During writes: 0 = Disable the EPMI2# signal from loading the wake-up GREEN state timer. 1 = Enable the loading of the wake-up GREEN state timer on the EPMI2# signal. During reads: 0 = GREEN event not invoked by EPMI2#. 1 = GREEN event has been invoked by EPMI2#. This bit and bit 6 of this register cannot be set to a 1 at the same time, as the EPMI2# signal is interpreted differently for each of the settings. During reads of the register (done to determine the status of the event) if bit 7 returns a 1, the user should interpret that as a GREEN event is invoked by the EPMI2# signal and if a 0 is returned, the user should interpret that as a GREEN event not being invoked by the EPMI2# signal.
6	R/W	0	Reload wake-up GREEN state timer whenever EPMI2# is active: 0 = Disable the EPMI2# signal from loading the wake-up GREEN state timer. 1 = Enable the loading of the wake-up GREEN state timer on the EPMI2# signal. As this interpretation of the EPMI2# signal is different from when bit 7, above, is set to a 1, the user should never set both bits 7 and 6 of this register to a 1 at the same time.
5	R/W	0	EPMI2# polarity: This bit determines the polarity of the EPMI2# signal in conjunction with bit 4 of this register. When this bit is set to a 0, polarity of the EPMI2# signal is determined by the setting on bit 4 of this register. However, if this bit is set to a 1, then regardless of the setting of bit 4 of this register, a transition on the EPMI2# signal will trigger an event based on the settings of bits 7 and 6 (explained above). It should be noted that when this bit is set to a 1, it is the user's responsibility to guarantee that the transition on EPMI2# signal lasts for a minimum period of 20ms before it departs from the current logic level. 0 = EPMI2# polarity will be determined by the setting on bit 4 of this register. 1 = An event will be triggered, dictated by the settings on bits 7 and 6 of this register on a transition of the EPMI2# signal, provided timing requirements are met on the EPMI2# signal.
4	R/W	0	EPMI2# polarity: This bit will determine the polarity of the EPMI2# signal as long as bit 5 of this register is not set. Listed is the combination of bits 5 and 4 of this register and the polarity on the EPMI2# signal that they define. If the combination is 00, then a falling edge on the EPMI2# signal will cause a triggering of an event. If the combination is 01, then a rising edge on the EPMI2# signal will cause a trigger of the event. If the combination is 1X, then a transition on EPMI2# signal will cause a trigger of the event as long as the timing requirements mentioned above are met.
3	R/W	0	EPMI2# debounce effect: This bit is used to select the debounce effect on the EPMI2# signal. Normally, external interrupts to the power management unit can cause glitches in the system. The user can avoid this glitching by setting this bit to a 1. Note that to avoid the glitching, the EPMI2# signal needs to stay stable for a minimum period of 5ms. 0 = No debounce effect has been enabled on the EPMI2# signal. 1 = Debounce effect has been selected on the EPMI2# signal. (Again, the EPMI2# signal should remain stable for a minimum period of 5ms for proper recognition.)
2:0	R/W	000	Reserved: Should be written to 0.

Table 5-135 58PMCTL3, Power Management Control Register 3 - Index FEh

Bit(s)	Type	Default	Function
7	R/W	0	EPMI3# becomes GREEN event: During writes: 0 = Disable the EPMI3# signal from loading the wake-up GREEN state timer. 1 = Enable the loading of the wake-up GREEN state timer on the EPMI3# signal. During reads: 0 = GREEN event not invoked by EPMI3#. 1 = GREEN event has been invoked by EPMI3#. This bit and bit 6 of this register cannot be set to a 1 at the same time, as the EPMI3# signal is interpreted differently for each of the settings. During reads of the register (done to determine the status of the event) if bit 7 returns a 1, the user should interpret that as a GREEN event is invoked by the EPMI3# signal and if a 0 is returned, the user should interpret that as a GREEN event not being invoked by the EPMI3# signal.
6	R/W	0	Reload wake-up GREEN state timer whenever EPMI3# is active: 0 = Disable the EPMI3# signal from loading the wake-up GREEN state timer. 1 = Enable the loading of the wake-up GREEN state timer on the EPMI3# signal. As this interpretation of the EPMI3# signal is different from when bit 7, above, is set to a 1, the user should never set both bits 7 and 6 of this register to a 1 at the same time.
5	R/W	0	EPMI3# polarity: This bit determines the polarity of the EPMI3# signal in conjunction with bit 4 of this register. When this bit is set to a 0, polarity of the EPMI3# signal is determined by the setting on bit 4 of this register. However, if this bit is set to a 1, then regardless of the setting of bit 4 of this register, a transition on the EPMI3# signal will trigger an event based on the settings of bits 7 and 6 (explained above). It should be noted that when this bit is set to a 1, it is the user's responsibility to guarantee that the transition on EPMI3# signal lasts for a minimum period of 20ms before it departs from the current logic level. 0 = EPMI3# polarity will be determined by the setting on bit 4 of this register. 1 = An event will be triggered, dictated by the settings on bits 7 and 6 of this register on a transition of the EPMI3# signal, provided timing requirements are met on the EPM3# signal.
4	R/W	0	EPMI3# polarity: This bit will determine the polarity of the EPMI3# signal as long as bit 5 of this register is not set. Listed is the combination of bits 5 and 4 of this register and the polarity on the EPMI3# signal that they define. If the combination is 00, then a falling edge on the EPMI3# signal will cause a triggering of an event. If the combination is 01, then a rising edge on the EPMI3# signal will cause a trigger of the event. If the combination is 1X, then a transition on EPMI3# signal will cause a trigger of the event as long as the timing requirements mentioned above are met.
3	R/W	0	EPMI3# debounce effect: This bit is used to select the debounce effect on the EPMI3# signal. Normally, external interrupts to the power management unit can cause glitches in the system. The user can avoid this glitching by setting this bit to a 1. Note that to avoid the glitching, the EPMI3# signal needs to stay stable for a minimum period of 5ms. 0 = No debounce effect has been enabled on the EPMI3# signal. 1 = Debounce effect has been selected on the EPMI3# signal. (Again, the EPMI3# signal should remain stable for a minimum period of 5ms for proper recognition.)
2:0	R/W	000	Reserved: Should be written to 0.

Table 5-136 58GPCCTL, General Purpose Chip Select Control Register - Index FFh

Bit(s)	Type	Default	Function
7	R/W	0	CPU type: This bit provides CPU vendor information to the chipset so it can provide the stop clock functionality. 0 = CPU vendor is Intel/AMD. 1 = CPU vendor is Cyrix and the CPU is M1.
6:5	R/W	00	Reserved: Must be written to 0.
4	R/W	0	Reserved: Must be written to 1.
3	R/W	0	Reserved: Must be written to 0.
2	R/W	0	Address bit masking capability: This bit is used in conjunction with Index F3h[6:4] to increase the flexibility in the address granularity of the address space of external devices. This table controls GPCS2#.

Index FFh[2]	Index F3h[6:4]	Address Bits Masked	Index FFh[2]	Index F3h[6:4]	Address Bits Masked
0	000	Mask no bit	1	000	Mask lowest 8 bits
0	001	Mask first lowest bit	1	001	Mask lowest 9 bits
0	010	Mask lowest 2 bits	1	010	Mask lowest 10 bits
0	011	Mask lowest 3 bits	1	011	Mask lowest 11 bits
0	100	Mask lowest 4 bits	1	100	Mask lowest 12 bits
0	101	Mask lowest 5 bits	1	110	Mask lowest 13 bits
0	110	Mask lowest 6 bits	1	110	Mask lowest 14 bits
0	111	Mask lowest 7 bits	1	111	Mask lowest 15 bits

1	R/W	0	Address bit masking capability: This bit is used in conjunction with Index F3h[2:0] to increase the granularity in the address space of external devices This table controls GPCS1#.

Index FFh[1]	Index F3h[2:0]	Address Bits Masked	Index FFh[1]	Index F3h[2:0]	Address Bits Masked
0	000	Mask no bit	1	000	Mask lowest 8 bits
0	001	Mask first lowest bit	1	001	Mask lowest 9 bits
0	010	Mask lowest 2 bits	1	010	Mask lowest 10 bits
0	011	Mask lowest 3 bits	1	011	Mask lowest 11 bits
0	100	Mask lowest 4 bits	1	100	Mask lowest 12 bits
0	101	Mask lowest 5 bits	1	110	Mask lowest 13 bits
0	110	Mask lowest 6 bits	1	110	Mask lowest 14 bits
0	111	Mask lowest 7 bits	1	111	Mask lowest 15 bits

0	R/W	0	Address bit masking capability: This bit is used in conjunction with Index E7h[2:0] to increase the granularity in the address space of external devices This table controls GPCS0#.

Index FFh[0]	Index E7h[2:0]	Address Bits Masked	Index FFh[0]	Index E7h[2:0]	Address Bits Masked
0	000	Mask no bit	1	000	Mask lowest 8 bits
0	001	Mask first lowest bit	1	001	Mask lowest 9 bits
0	010	Mask lowest 2 bits	1	010	Mask lowest 10 bits
0	011	Mask lowest 3 bits	1	011	Mask lowest 11 bits
0	100	Mask lowest 4 bits	1	100	Mask lowest 12 bits
0	101	Mask lowest 5 bits	1	110	Mask lowest 13 bits
0	110	Mask lowest 6 bits	1	110	Mask lowest 14 bits
0	111	Mask lowest 7 bits	1	111	Mask lowest 15 bits

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:
  - 82C556M = 400mW
  - 82C557M = 600mW
  - 82C558M = 600mW

6.4 82C556M 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 82C557M 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	

82C557M 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 82C558M 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	

82C558M 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

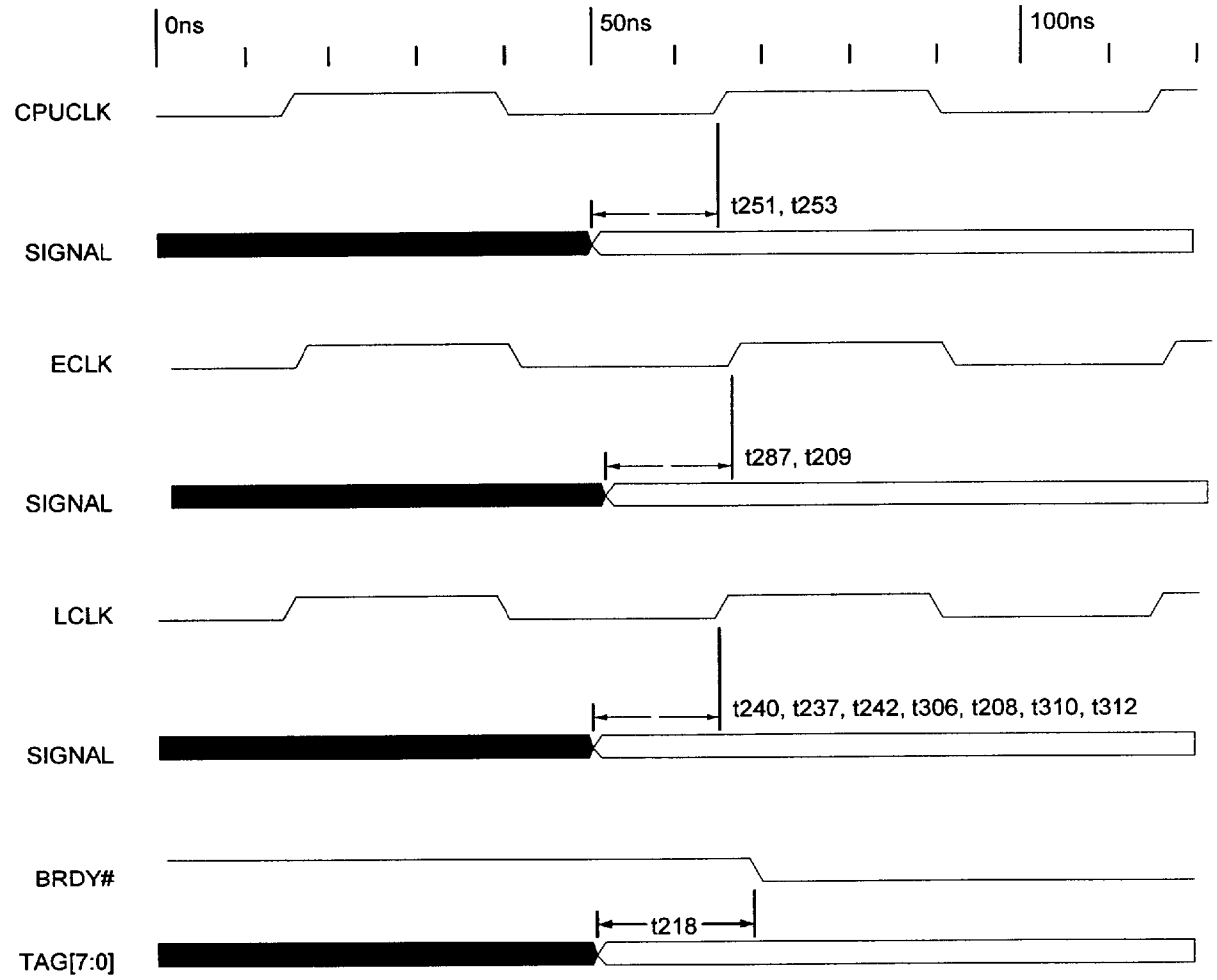


Figure 6-2 Hold Timing Waveform

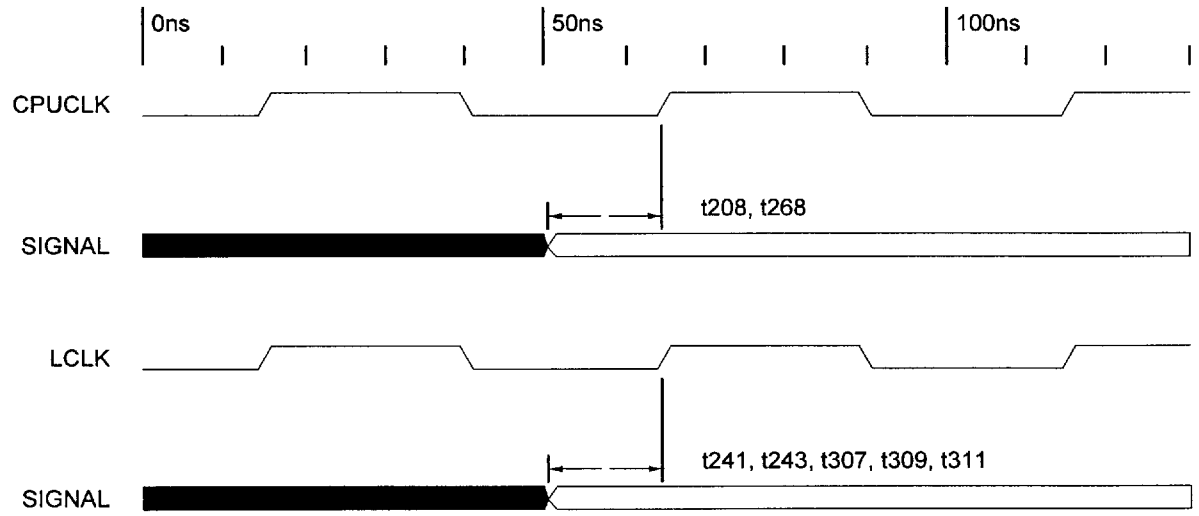


Figure 6-3 Output Delay Timing Waveform

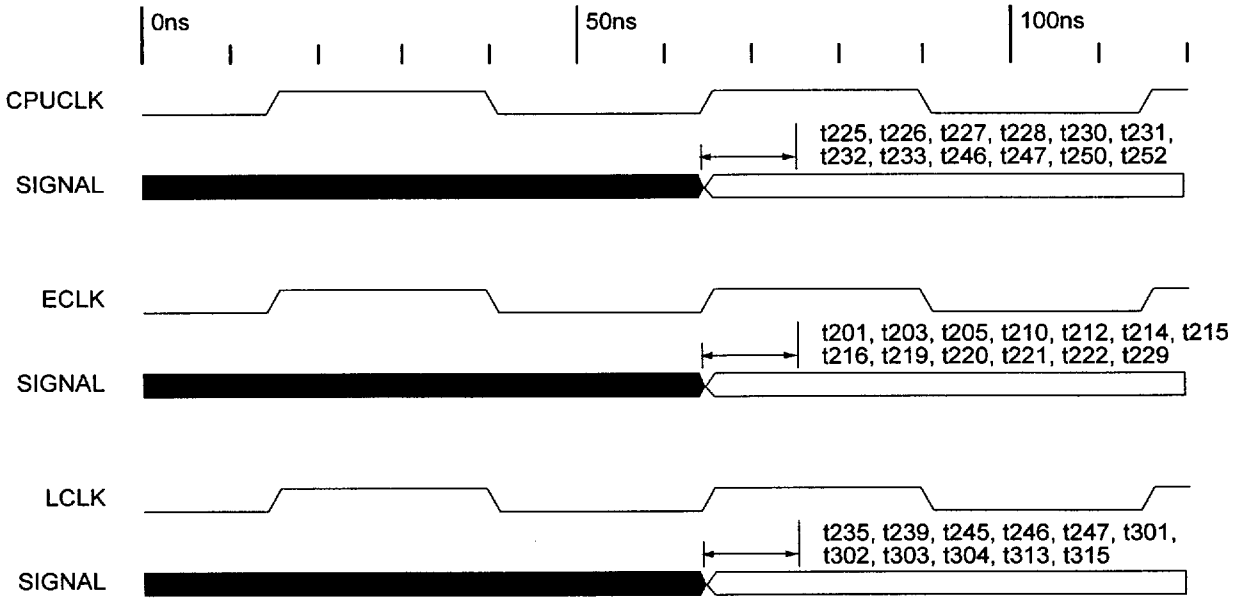
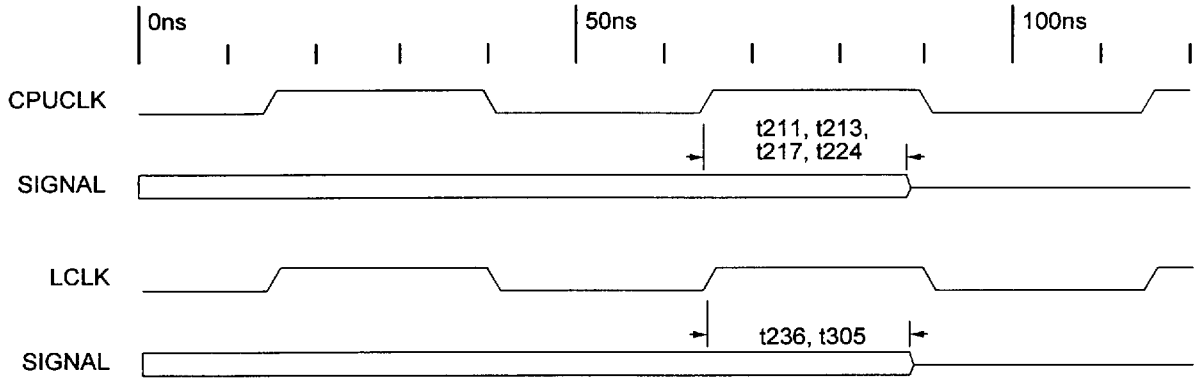


Figure 6-4 Float Delay Timing Waveform



7.0 Test Mode Information

Each device in the Viper-M Chipset can be forced into various test modes for board-level testing automatic test equipment (ATE).

- 82C556M
  - Test Mode 0: All outputs and bidirectional pins are tristated.
  - Test Mode 1 (NAND tree test): All bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 154.
- 82C557M
  - Test Mode 0: All outputs and bidirectional pins are tristated.
  - Test Mode 1 (NAND tree test): All bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 127.
- 82C558M
  - Test Mode 0: All outputs and bidirectional pins are tristated.
  - Test Mode 1 (NAND tree test): All bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 92.
  - Test Mode 2: All even numbered output pins are driven high and all odd numbered output pins are driven low.
  - Test Mode 3: All even numbered output pins are driven low and all odd numbered output pins are driven high.

The following subsections will explain how to enable the devices in the Viper-M Chipset into the above mentioned test modes.

7.1 82C556M Testability

When the 82C556M decodes the RESET combination, it generates an internal reset signal and straps in the value of the MP[5:4] lines. The RESET combination defined below show the minimum number of required clocks. It is recommended that the user program higher values than these.

- DBCOE0#, DBCOE1#, HDOE#, MDOE#, DLE1#, and DLE0# all = 1
- 2 CPUCLKs later DBCOE1# changes to 0
- 2 CPUCLKs later HDOE# changes to 0
- 10 CPUCLKs later DLE1# changes to 0
- 20 CPUCLKs later DLE1# changes to 1
- 10 CPUCLKs later HDOE# changes to 1
- 2 CPUCLKs later DBCOE0# changes to 1

If MP4 is sampled low at the point that this RESET condition is decoded, then the 82C556M enters the test mode:

MP4 = 0 - test mode is enabled
MP4 = 1 - test mode is disabled

If the test mode has been enabled, then the strap information on MP5 decides which test mode has been enabled:

- MP5 = 0: Test Mode 0 is enabled (i.e., all outputs and bidirectional pins are tristated)
- MP5 = 1: Test Mode 1 (NAND tree test) is enabled (i.e., all bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 154).

7.1.1 82C556M NAND Tree Test (Test Mode 1)

The NAND tree testing is enabled if the test mode has been enabled and if MP5 is held high, which enables Test Mode 1. The NAND tree mode is used to test input and bidirectional pins which will be part of the NAND tree chain. The NAND tree chain starts at Pin 155 and the output of the chain is Pin 154. Table 7-1 gives the pins of the NAND tree chain.

7.2 82C557M Testability

The 82C557M samples all its strap information during RESET. If the HREQ pin is sampled low during RESET, the 82C557M enters the test mode. Strap information is sampled 4096 LCLKs after the rising edge of PWRGD.

- HREQ = 0 at the rising edge of RESET - test mode is enabled
- HREQ = 1 at the rising edge of RESET - test mode is disabled

If the test mode has been enabled, then the strap information on the MASTER#, REFRESH#, and AEN lines decide which test mode has been enabled:

- MASTER# = 0, AEN = 0, and REFRESH# = 1: Test Mode 0 is enabled (i.e., all outputs and bidirectional pins are tristated)
- MASTER# = 0, AEN = 1, and REFRESH# = 1: Test Mode 1 (NAND tree test) is enabled (i.e., all bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 127).

7.2.1 82C557M NAND Tree Test (Test Mode 1)

The NAND tree testing is enabled if the test mode has been enabled and if MASTER# = 0, AEN = 1, and REFRESH# = 1, to enable Test Mode 1. The NAND tree test is used to test input and bidirectional pins which will be part of the NAND tree chain. The NAND tree chain starts at Pin 126 and the output of the chain is Pin 127. Table 7-2 gives the pins of the NAND tree chain.

Table 7-1 82C556M NAND Tree Test Mode Pins

Pin No.	Remarks
155	NAND tree input start
156	NAND tree input
157	NAND tree input
158	NAND tree input
159	NAND tree input
160	NAND tree input
1	NAND tree input
2	NAND tree input
3	NAND tree input
4	NAND tree input
5	NAND tree input
6	NAND tree input
7	NAND tree input
8	NAND tree input
9	NAND tree input
10	NAND tree input
11	NAND tree input
12	NAND tree input
13	NAND tree input
14	NAND tree input
15	NAND tree input
16	NAND tree input
17	NAND tree input
18	NAND tree input
19	NAND tree input
21	NAND tree input
22	NAND tree input
25	NAND tree input
26	NAND tree input
27	NAND tree input
28	NAND tree input
29	NAND tree input
31	NAND tree input
32	NAND tree input
33	NAND tree input
34	NAND tree input

Pin No.	Remarks
35	NAND tree input
36	NAND tree input
37	NAND tree input
38	NAND tree input
39	NAND tree input
40	NAND tree input
41	NAND tree input
42	NAND tree input
43	NAND tree input
44	NAND tree input
45	NAND tree input
46	NAND tree input
47	NAND tree input
48	NAND tree input
49	NAND tree input
51	NAND tree input
52	NAND tree input
53	NAND tree input
54	NAND tree input
55	NAND tree input
56	NAND tree input
57	NAND tree input
58	NAND tree input
60	NAND tree input
61	NAND tree input
62	NAND tree input
63	NAND tree input
64	NAND tree input
65	NAND tree input
66	NAND tree input
67	NAND tree input
68	NAND tree input
69	NAND tree input
71	NAND tree input
72	NAND tree input
73	NAND tree input

Pin No.	Remarks
74	NAND tree input
75	NAND tree input
76	NAND tree input
77	NAND tree input
78	NAND tree input
79	NAND tree input
80	NAND tree input
81	NAND tree input
82	NAND tree input
83	NAND tree input
84	NAND tree input
85	NAND tree input
86	NAND tree input
87	NAND tree input
88	NAND tree input
89	NAND tree input
92	NAND tree input
93	NAND tree input
94	NAND tree input
95	NAND tree input
96	NAND tree input
97	NAND tree input
98	NAND tree input
99	NAND tree input
101	NAND tree input
104	NAND tree input
105	NAND tree input
106	NAND tree input
107	NAND tree input
108	NAND tree input
109	NAND tree input
110	NAND tree input
111	NAND tree input
112	NAND tree input
113	NAND tree input
114	NAND tree input

Pin No.	Remarks
115	NAND tree input
116	NAND tree input
117	NAND tree input
118	NAND tree input
119	NAND tree input
120	NAND tree input
121	NAND tree input
122	NAND tree input
123	NAND tree input
124	NAND tree input
125	NAND tree input
126	NAND tree input
127	NAND tree input
129	NAND tree input
131	NAND tree input
132	NAND tree input
133	NAND tree input
134	NAND tree input
135	NAND tree input
136	NAND tree input
137	NAND tree input
138	NAND tree input
139	NAND tree input
140	NAND tree input
141	NAND tree input
142	NAND tree input
143	NAND tree input
144	NAND tree input
145	NAND tree input
146	NAND tree input
147	NAND tree input
148	NAND tree input
149	NAND tree input
151	NAND tree input
152	NAND tree input
154	NAND tree output

Table 7-2 82C557M NAND Tree Test Mode Pins

Pin No.	Remarks
126	NAND tree input start
125	NAND tree input
124	NAND tree input
123	NAND tree input
122	NAND tree input
121	NAND tree input
120	NAND tree input
118	NAND tree input
117	NAND tree input
116	NAND tree input
115	NAND tree input
113	NAND tree input
112	NAND tree input
111	NAND tree input
110	NAND tree input
109	NAND tree input
108	NAND tree input
107	NAND tree input
106	NAND tree input
105	NAND tree input
57	NAND tree input
56	NAND tree input
55	NAND tree input
54	NAND tree input
53	NAND tree input
52	NAND tree input
51	NAND tree input
50	NAND tree input
49	NAND tree input
48	NAND tree input
47	NAND tree input
46	NAND tree input

Pin No.	Remarks
45	NAND tree input
44	NAND tree input
42	NAND tree input
41	NAND tree input
40	NAND tree input
39	NAND tree input
37	NAND tree input
36	NAND tree input
35	NAND tree input
34	NAND tree input
33	NAND tree input
32	NAND tree input
31	NAND tree input
30	NAND tree input
29	NAND tree input
28	NAND tree input
27	NAND tree input
26	NAND tree input
25	NAND tree input
24	NAND tree input
23	NAND tree input
22	NAND tree input
21	NAND tree input
20	NAND tree input
19	NAND tree input
18	NAND tree input
17	NAND tree input
16	NAND tree input
14	NAND tree input
12	NAND tree input
5	NAND tree input
4	NAND tree input

Pin No.	Remarks
3	NAND tree input
2	NAND tree input
207	NAND tree input
206	NAND tree input
205	NAND tree input
204	NAND tree input
203	NAND tree input
202	NAND tree input
199	NAND tree input
198	NAND tree input
197	NAND tree input
195	NAND tree input
188	NAND tree input
187	NAND tree input
186	NAND tree input
185	NAND tree input
183	NAND tree input
180	NAND tree input
178	NAND tree input
177	NAND tree input
176	NAND tree input
175	NAND tree input
174	NAND tree input
173	NAND tree input
172	NAND tree input
171	NAND tree input
170	NAND tree input
169	NAND tree input
168	NAND tree input
167	NAND tree input
166	NAND tree input
162	NAND tree input

Pin No.	Remarks
161	NAND tree input
160	NAND tree input
159	NAND tree input
158	NAND tree input
157	NAND tree input
156	NAND tree input
155	NAND tree input
154	NAND tree input
153	NAND tree input
152	NAND tree input
151	NAND tree input
150	NAND tree input
149	NAND tree input
148	NAND tree input
146	NAND tree input
145	NAND tree input
144	NAND tree input
143	NAND tree input
140	NAND tree input
139	NAND tree input
138	NAND tree input
137	NAND tree input
136	NAND tree input
135	NAND tree input
134	NAND tree input
133	NAND tree input
132	NAND tree input
131	NAND tree input
130	NAND tree input
129	NAND tree input
127	NAND tree output

7.3 82C558M Testability

The 82C558M samples all its strap information on the falling edge of RESET. If the HREQ pin is sampled low on the falling edge of RESET, the 82C558M enters the test mode.

- HREQ = 0 at the falling edge of RESET - test mode is enabled
- HREQ = 1 at the falling edge of RESET - test mode is disabled

If the test mode has been enabled, then the strap information on the IRQ1 and IRQ3 lines decide which test mode has been enabled:

IRQ1 = 0 and IRQ3 = 0: Test Mode 0 is enabled (i.e., all outputs and bidirectional pins are tristated)

IRQ1 = 0 and IRQ3 = 1: Test Mode 1 (NAND tree test) is enabled (i.e., all bidirectionals are tristated and the end of the input and bidirectional NAND chain is present on Pin 92).

IRQ = 1 and IRQ3 = 0: Test Mode 2 is enabled (i.e., all even numbered output pins are driven high and all odd numbered output pins are driven low).

IRQ = 1 and IRQ3 = 1: Test Mode 3 is enabled (i.e., all even numbered output pins are driven low and all odd numbered output pins are driven high).

7.3.1 82C558M NAND Tree Test (Test Mode 1)

The NAND tree testing is enabled if the test mode has been enabled and IRQ1 = 0 and IRQ3 = 1, to enable Test Mode 1. The NAND tree test is used to test input and bidirectional pins which will be part of the NAND tree chain. The NAND tree chain starts at Pin 2 and the output of the chain is Pin 92. Table 7-4 gives the pins of the NAND tree chain.

7.3.2 Drive High/Drive Low Test (Test Modes 2 and 3)

The 82C558M supports two type of drive high/drive low tests. If the test mode has been enabled, and if IRQ1 = 1 and IRQ3 = 0, then Test Mode 2 is enabled. In this mode, all even numbered output pins are driven high and all odd numbered output pins are driven low. Table 7-3 shows which pins are driven high/low during this test.

If the test mode has been enabled and if IRQ1 = 1 and IRQ3 = 1, then Test Mode 3 is enabled. In this mode, all odd numbered output pins are driven high and all even numbered output pins are driven low. Table 7-3 shows which pins are driven high/low during this test.

Table 7-3 Drive High/Drive Low Test Mode Pins

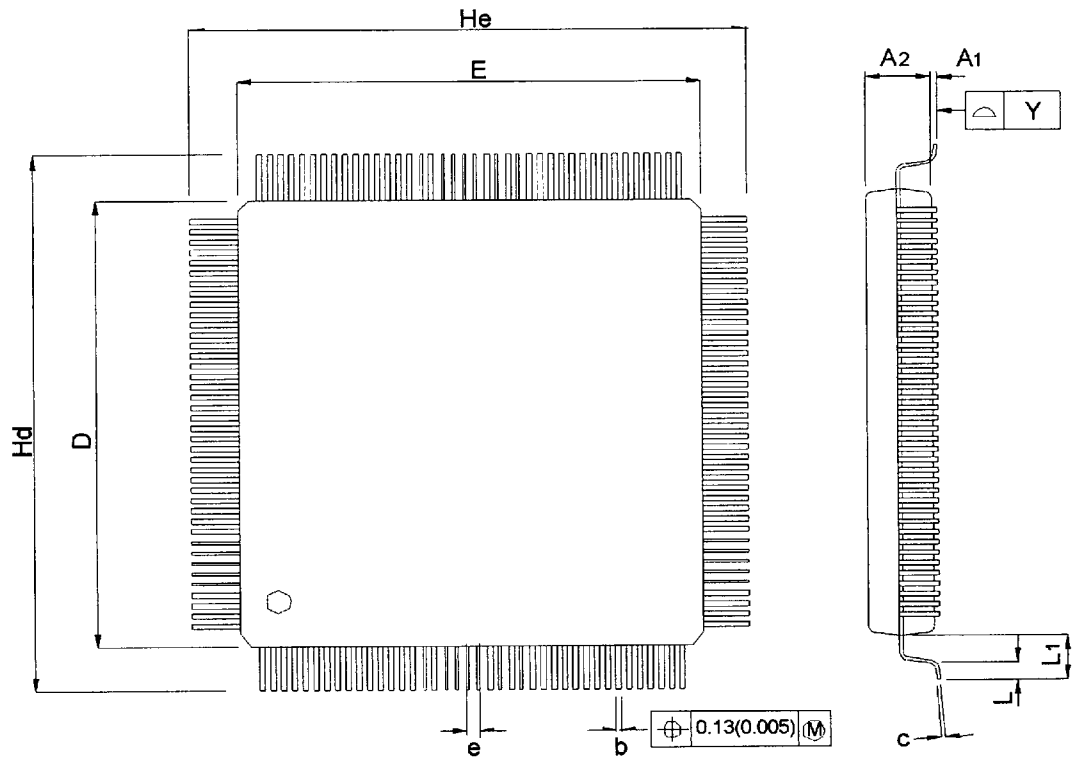
Even Number Pins	Odd Number Pins
2	3
8	9
52	13
74	53
80	81
84	85
92	105
106	107
108	109
110	111
112	113
148	115
150	149
154	

Table 7-4 82C558M NAND Tree Test Mode Pins

Pin No.	Remarks	Pin No.	Remarks	Pin No.	Remarks	Pin No.	Remarks
4	NAND tree input start	55	NAND tree input	117	NAND tree input	166	NAND tree input
5	NAND tree input	56	NAND tree input	118	NAND tree input	167	NAND tree input
6	NAND tree input	57	NAND tree input	120	NAND tree input	168	NAND tree input
7	NAND tree input	58	NAND tree input	121	NAND tree input	169	NAND tree input
11	NAND tree input	59	NAND tree input	122	NAND tree input	170	NAND tree input
12	NAND tree input	62	NAND tree input	123	NAND tree input	171	NAND tree input
16	NAND tree input	63	NAND tree input	124	NAND tree input	172	NAND tree input
17	NAND tree input	64	NAND tree input	125	NAND tree input	173	NAND tree input
18	NAND tree input	65	NAND tree input	126	NAND tree input	174	NAND tree input
19	NAND tree input	66	NAND tree input	127	NAND tree input	175	NAND tree input
20	NAND tree input	67	NAND tree input	128	NAND tree input	176	NAND tree input
21	NAND tree input	68	NAND tree input	129	NAND tree input	177	NAND tree input
22	NAND tree input	69	NAND tree input	130	NAND tree input	178	NAND tree input
23	NAND tree input	70	NAND tree input	131	NAND tree input	179	NAND tree input
24	NAND tree input	71	NAND tree input	132	NAND tree input	181	NAND tree input
25	NAND tree input	73	NAND tree input	133	NAND tree input	182	NAND tree input
27	NAND tree input	75	NAND tree input	134	NAND tree input	183	NAND tree input
28	NAND tree input	76	NAND tree input	135	NAND tree input	184	NAND tree input
29	NAND tree input	77	NAND tree input	136	NAND tree input	187	NAND tree input
30	NAND tree input	78	NAND tree input	138	NAND tree input	188	NAND tree input
31	NAND tree input	79	NAND tree input	139	NAND tree input	189	NAND tree input
32	NAND tree input	82	NAND tree input	140	NAND tree input	190	NAND tree input
33	NAND tree input	83	NAND tree input	141	NAND tree input	191	NAND tree input
34	NAND tree input	86	NAND tree input	143	NAND tree input	192	NAND tree input
35	NAND tree input	88	NAND tree input	144	NAND tree input	193	NAND tree input
36	NAND tree input	89	NAND tree input	145	NAND tree input	194	NAND tree input
39	NAND tree input	90	NAND tree input	146	NAND tree input	195	NAND tree input
40	NAND tree input	91	NAND tree input	151	NAND tree input	196	NAND tree input
41	NAND tree input	93	NAND tree input	152	NAND tree input	197	NAND tree input
42	NAND tree input	94	NAND tree input	153	NAND tree input	198	NAND tree input
44	NAND tree input	95	NAND tree input	155	NAND tree input	199	NAND tree input
45	NAND tree input	98	NAND tree input	156	NAND tree input	202	NAND tree input
46	NAND tree input	99	NAND tree input	157	NAND tree input	203	NAND tree input
47	NAND tree input	100	NAND tree input	158	NAND tree input	204	NAND tree input
48	NAND tree input	101	NAND tree input	159	NAND tree input	205	NAND tree input
49	NAND tree input	102	NAND tree input	160	NAND tree input	206	NAND tree input
50	NAND tree input	103	NAND tree input	161	NAND tree input	207	NAND tree input
51	NAND tree input	104	NAND tree input	162	NAND tree input	208	NAND tree input
54	NAND tree input	116	NAND tree input	163	NAND tree input	92	NAND tree output

8.0 Mechanical Package Outlines

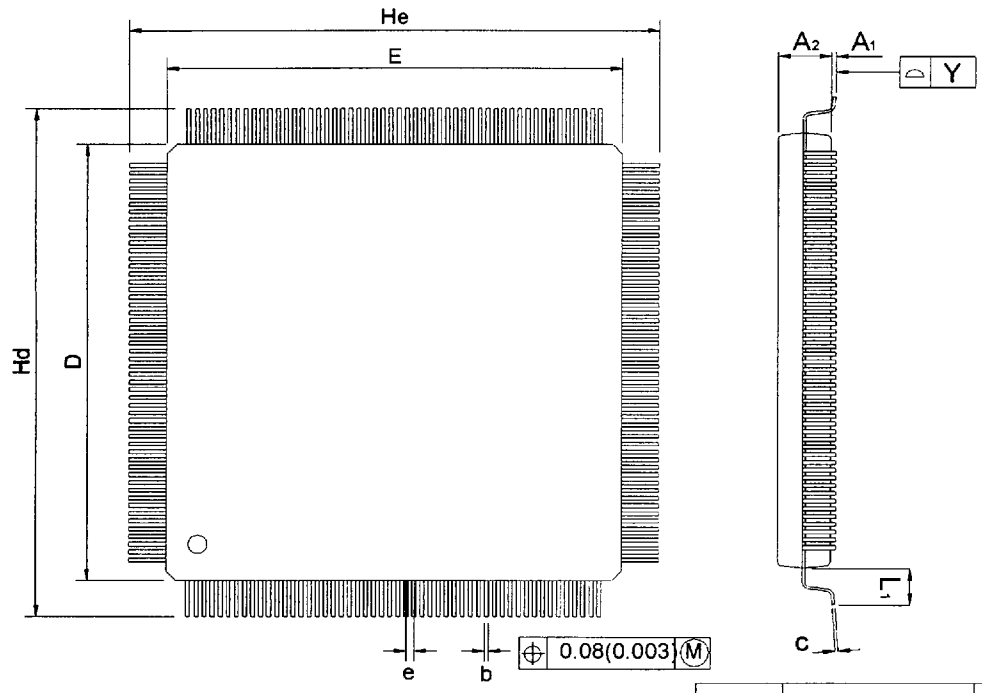
Figure 8-1 160-Pin Plastic Quad Flat Pack (PQFP)

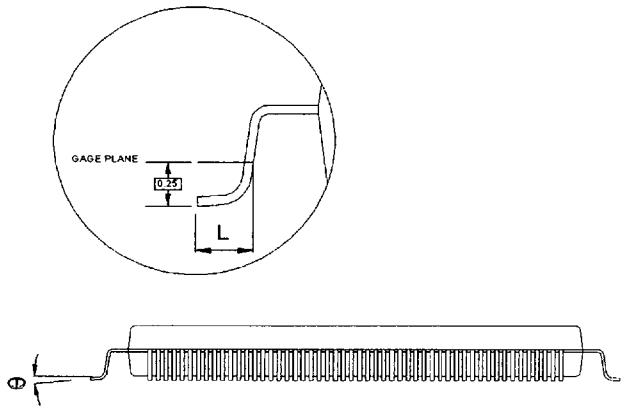


SYMBOL	MILLIMETER MIN.	MILLIMETER NOM.	MILLIMETER MAX.	INCH MIN.	INCH NOM.	INCH MAX.
A1	0.05	0.25	0.50	0.002	0.010	0.020
A2	3.17	3.32	3.47	0.125	0.131	0.137
b	0.20	0.30	0.40	0.008	0.012	0.016
c	0.10	0.15	0.20	0.004	0.006	0.008
D	27.90	28.00	28.10	1.098	1.102	1.106
E	27.90	28.00	28.10	1.098	1.102	1.106
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		7	0		7

Dwg. No.: AS160PQFP-001
Dwg. Rev.: A0 Unit: MM / INCH

Figure 8-2 208-Pin Plastic Quad Flat Package (PQFP)

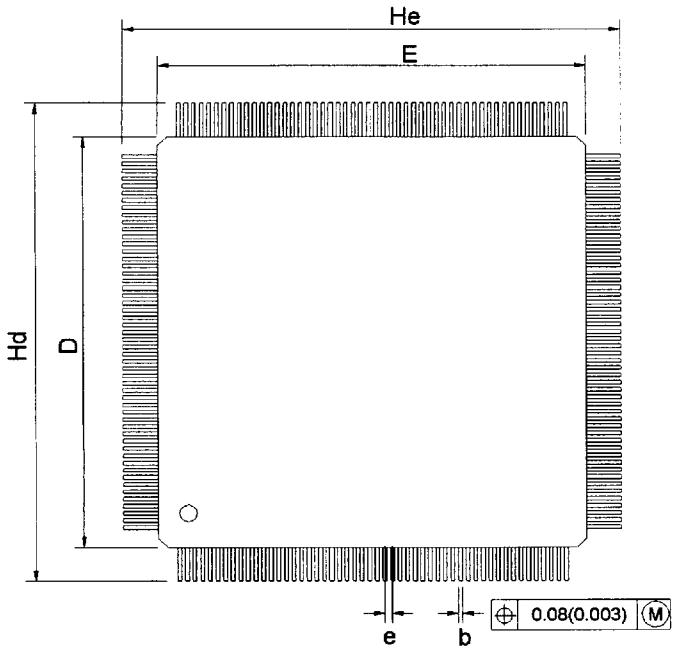


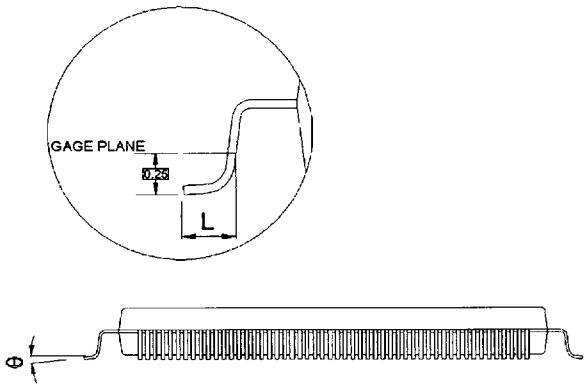


SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A_1	0.05	0.25	0.50	0.002	0.010	0.020
A_2	3.17	3.32	3.47	0.125	0.131	0.137
b	0.10	0.20	0.30	0.004	0.008	0.012
c	0.10	0.15	0.20	0.004	0.006	0.008
D	27.90	28.00	28.10	1.098	1.102	1.106
E	27.90	28.00	28.10	1.098	1.102	1.106
e		0.50			0.020	
Hd	30.35	30.60	30.85	1.195	1.205	1.215
He	30.35	30.60	30.85	1.195	1.205	1.215
L	0.50	0.60	0.75	0.020	0.024	0.030
L_1		1.30			0.051	
Y			0.08			0.003
θ	0		7	0		7

Dwg. No.:	AS208PQFP-001		
Dwg. Rev.:	A1	Unit:	MM / INCH

Figure 8-3 208-Pin Thin Quad Flat Pack (TQFP)





SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A_1	0.05	0.10	0.15	0.002	0.004	0.006
A_2	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
L_1		1.00			0.039	
Y			0.08			0.003
θ	0		7	0		7

Dwg. No.:	AS208TQFP-001
Dwg. Rev.: A0	Unit: MM / INCH

A.0 Accessing the BBS

The OPTi BBS offers a wide range of useful files and utilities to our customers, from Evaluation PCB Schematics to HPGL/ PostScript format Databooks that you can copy directly to your laser printer. The only requirements for accessing and using the BBS is a modem and an honest response to our questionnaire.

A.1 Paging the SYSOP

Currently, Paging the SYSOP is not a valid choice for the OPTi BBS. Once a full-time SYSOP is created, then there will be hours available for paging the SYSOP and getting immediate help.

For now, you must send [C] Comments to the SYSOP with any questions or problems you are experiencing. They will be answered promptly.

Note: Each conference has its own Co-SYSOP (the application engineer responsible for that product line), so specific conference questions can be addressed that way, but general, BBS-wide, questions should be sent to the SYSOP from the [0] - Private E-Mail conference.

A.2 System Requirements

The OPTi BBS will support any PC modem up to 14,400 baud, with 8 bits, no parity, and 1 stop bit protocol. The baud rate, handshaking, and system type will automatically be detected by the OPTi BBS.

A.3 Calling In/Hours of Operation

The OPTi BBS phone number is (408) 980-9774. The BBS is on-line 24 hours a day, seven days a week. Currently there is only one line, but as traffic requires additional lines will be installed.

A.4 Logging On for the First Time

To log on to the BBS for the first time,

1. Call (408) 980-9774 with your modem.
2. Enter your first name.
3. Enter your last name.
4. Verify that you have typed your name correctly.
5. Select a password (write it down).
6. Reenter the password to verify spelling.
7. You must then answer the questionnaire that follows.

After you have answered the questionnaire, you are given Customer rights. To change your profile (security level, password, etc.), you must send a [C]omment to the SYSOP explaining why.

After you have logged on for the first time, each subsequent log on will bypass the questionnaire and put you directly at the bulletin request prompt. As bulletins will be added on a regular basis in the future, it is recommended that you read the new bulletins on a regular basis.

A.5 Log On Rules and Regulations

- As a FULLUSER you can download from any conference.
- You will be limited to 45 minutes per day of access time (note that once a download has started, it will finish, even if the daily time limit is exceeded). If you have not entered any keystrokes after 5 minutes, you will automatically be logged off.
- You can upload to the Customer Upload Conference[1] only. This area is used for our customers/contacts to send data to OPTi. You will not be able to download any files from this area.

A.6 Using the BBS

This section will describe how to use the BBS on a daily basis.

The BBS is divided into Conferences that are specific to a product (for example, the Viper Desktop Chipset), or an application group (for example, the Field Application Conference is used by OPTi Field Application Engineers to send data to their contacts in the field). As a general rule, the files in the application specific areas will be for specific application and may contain a password. If a file is password protected, and you know you need that file, you must contact your OPTi sales representative for the password.

The files in the Product Conferences are released data that can be used for evaluating the OPTi product line.

To access a feature of the BBS, you should type the letter in brackets that precedes each menu item. This document places the appropriate letter in brackets whenever you are told to access a feature.

A.6.1 Reading Bulletins

The OPTi BBS will present you with a set of bulletins each time you log on that are global bulletins applying to OPTi in general. In addition to these, each Product Conference will have its own set of bulletins that apply to that product. These bulletins will announce new product information, documentation updates, and bug fixes and product alerts.

1. See Section A.6.5 for more information on uploading.

It is recommended that you read any new bulletins on a regular basis to keep up to date on the OPTi product line.

A.6.2 Sending/Receiving Messages

The Message Menu can be used to send and receive messages from OPTi employees, or other BBS users. The Message menu can also be used to attach files for the receiver to download after they read the message. This method will be used often to send customer specific files to OPTi customers.

Messages to the SYSOP depend upon the conference you are in. Each Product Conference sends Comments and Messages to the SYSOP to the Application Engineer responsible for that conference.

A.6.3 Finding Information

To find information on the OPTi BBS, you must use the [J] Join a Conference option and then list all of the conferences available. They are arranged by product number and name.

Once you are in the correct conference, you should read all applicable bulletins and messages. Then you can [L] List all the files that are available from the File Menu.

A.6.4 Downloading Files From OPTi

The easiest way to download files from OPTi is to [L] List the files from the File Menu, the [M] Mark and files you want from the list. After you have marked all the files you need, you can [D] Download all the marked files and then logoff automatically.

A.6.5 Uploading Files To OPTi

There are two ways to upload a file to OPTi. The first is similar to the download option. You should [J] Join the Customer Upload Conference (this is the only conference that allows uploads from users) and [U] Upload the file to this conference.

If you are sending the file to a specific person, you should use the Message Menu to [E] Enter a new message to that person and then [A] Attach the file to the message. This way, the person receiving the message can download the file to his or her system without leaving behind a file that will not be used by anyone else on the BBS.

A.6.6 Logging Off

Once you have completed your visit to the OPTi BBS, you must say [G] Goodbye.

A.6.7 Logging Back on Again

To log back on to the BBS,

1. Call (408) 980-9774 with your modem.
2. Enter your first name.
3. Enter your last name.
4. Verify that you have typed your name correctly.
5. Enter your password.

You will not have to answer the questionnaire after the initial log-on. You will also be in the conference you were in when you last logged-on.

A.7 The Menus

There are four major menus that OPTi customers will use, the Main Menu, the File Menu, the Bulletin Menu and the Message Menu.

Note: The following menus are for the Customer Profile (FULLUSER) only, if your user profile has been changed, you may see slightly different menus.

Figure A-1 The Main Menu

```
MAIN MENU:
[ J ] Join a conference              [ F ] File menu
[ M ] Message menu                   [ B ] Bulletin menu
[ C ] Comments to the sysop          [ U ] Userlog list
[ Y ] Your settings                  [ G ] Goodbye & logoff

Conf: "[0] - Private E-Mail", time on 0, with 45 remaining.
MAIN MENU: [J F M B C P U Y G] ?
```

Figure A-2 The Bulletin Menu

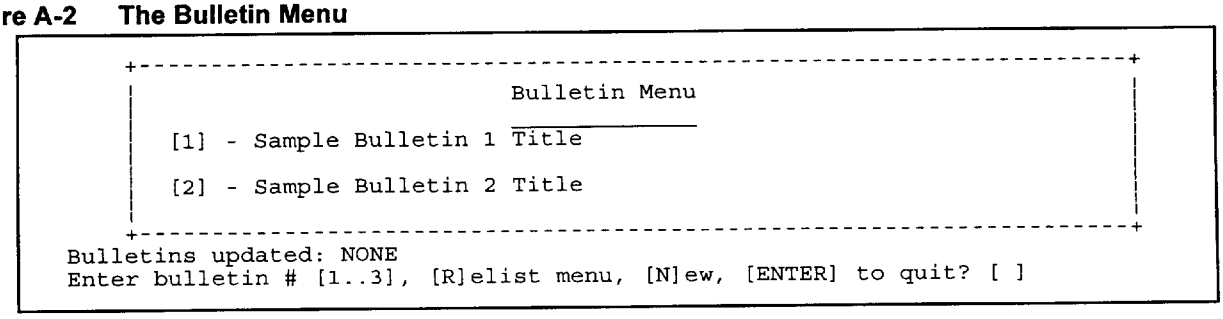


Figure A-3 The File Menu

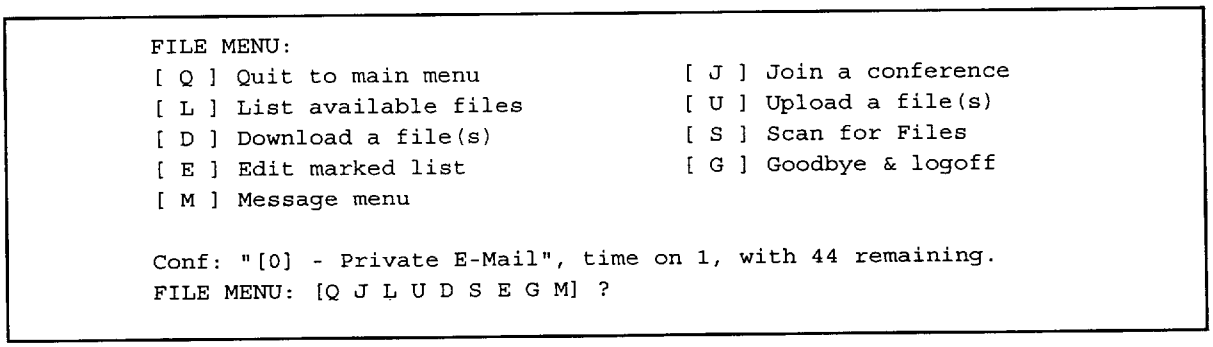
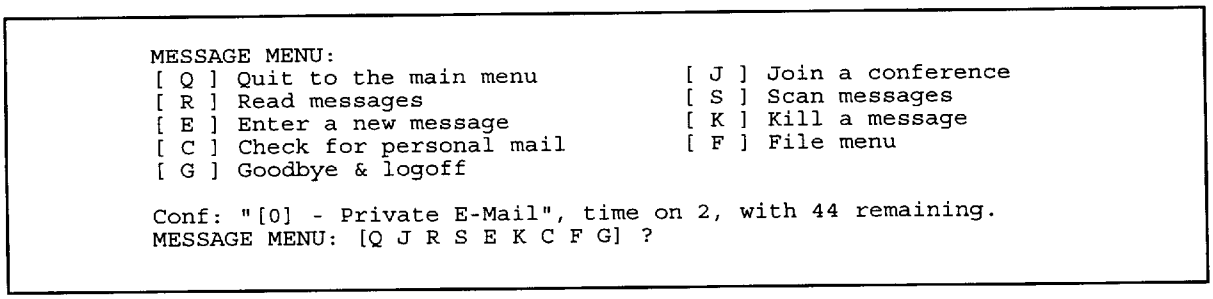


Figure A-4 The Message Menu



A.7.1 Menu Selections

- [B] Bulletin MenuMenu(s): main
  Access the Bulletin Menu.
- [C] Check for personal mailMenu(s): message
  See if you have any mail.
- [C] Comments to the sysopMenu(s): main
  Leave a private comment for the SYSOP.
- [D] Download a file(s)Menu(s): file
  Download a file from the BBS to your computer. If you have marked files it will display these files. If you have not marked any files, it will ask you for a file name. The file must be present in the current conference for you to be able to enter its name.
- [E] Edit Marked ListMenu(s): file
  Change the entries that you have selected as Marked for downloading.

- [E] Enter a new messageMenu(s): message
  Send a new message to someone on the BBS.
- [F] File MenuMenu(s): main, message
  Access the File Menu.
- [G] Goodbye and logoffMenu(s): main, message, file
  Logoff the system.
- [J] Join a ConferenceMenu(s): main, message, file
  Change conferences (product areas).
- [K] Kill a messageMenu(s): message
  Delete a message.
- [L] List available filesMenu(s): file
  List the files in the current conference. Note that most conferences have sub-categories of files (Schematics, JOB, etc.) that you will be asked for (or you can press enter the list all of the categories).
- [M] Message MenuMenu(s): main, file
  Access the Message Menu.
- [Q] Quit to Main MenuMenu(s): message, file
  Leave current menu and return to the Main Menu.
- [R] Read MessagesMenu(s): message
  Read messages in the current conference or all conferences.
- [S] Scan for FilesMenu(s): file
  Scan for particular files (by name, or extension, etc.).
- [S] Scan messagesMenu(s): message
  Search for message by specific qualifier (date, sender etc.).
- [U] Upload a file(s)Menu(s): file
  Send a file from your computer to OPTi. This can only be done in the Customer Upload Conference.
- [U] Userlog ListMenu(s): main
  Lists the user database, in order of logon. This is useful if you are sending a message and are looking for the spelling of a persons name.
- [Y] Your settingsMenu(s): main
  Show you settings and allow you to make changes. These include password, name, address, etc.