

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION TO THE INTEL ARCHITECTURE MMX™ TECHNOLOGY

1.1.	ABOUT THE INTEL ARCHITECTURE MMX™ TECHNOLOGY	1-1
1.1.1.	Single Instruction, Multiple Data (SIMD) Technique.....	1-1
1.1.2.	Performance Improvement	1-2
1.2.	ABOUT THIS MANUAL	1-2
1.3.	RELATED DOCUMENTATION	1-3

CHAPTER 2

INTEL ARCHITECTURE MMX™ TECHNOLOGY FEATURES

2.1.	NEW FEATURES.....	2-1
2.2.	NEW DATA TYPES.....	2-1
2.3.	MMX™ REGISTERS.....	2-2
2.4.	EXTENDED INSTRUCTION SET	2-3
2.4.1.	Packed Data	2-3
2.4.2.	Saturation Arithmetic Vs. Wrap Around.....	2-4
2.4.3.	Instruction Group Overview	2-5
2.4.3.1.	ARITHMETIC INSTRUCTIONS.....	2-5
2.4.3.2.	COMPARISON INSTRUCTIONS	2-6
2.4.3.3.	CONVERSION INSTRUCTIONS.....	2-6
2.4.3.4.	LOGICAL INSTRUCTIONS.....	2-6
2.4.3.5.	SHIFT INSTRUCTIONS.....	2-7
2.4.3.6.	DATA TRANSFER INSTRUCTIONS	2-7
2.4.3.7.	EMMS (EMPTY MMX™ STATE) INSTRUCTION	2-7
2.4.4.	Instruction Operand.....	2-7
2.5.	COMPATIBILITY	2-8

CHAPTER 3

APPLICATION PROGRAMMING MODEL

3.1.	DATA FORMATS.....	3-1
3.1.1.	Memory Data Formats	3-1
3.1.2.	IA MMX™ Register Data Formats	3-2
3.1.3.	IA MMX™ Instructions and the Floating-Point Tag Word	3-2
3.2.	PREFIXES.....	3-3
3.3.	WRITING APPLICATIONS WITH IA MMX™ CODE	3-3
3.3.1.	Detecting IA MMX™ Technology Existence Using the CPUID Instruction.....	3-3
3.3.2.	The EMMS Instruction.....	3-4
3.3.3.	Interfacing with IA MMX™ Technology Procedures and Functions.....	3-5
3.3.4.	Writing Code with IA MMX™ and Floating-Point Instructions	3-5
3.3.4.1.	RECOMMENDATIONS AND GUIDELINES	3-6
3.3.5.	Multitasking Operating System Environment.....	3-7
3.3.5.1.	COOPERATIVE MULTITASKING OPERATING SYSTEM.....	3-7
3.3.5.2.	PREEMPTIVE MULTITASKING OPERATING SYSTEM	3-7
3.3.6.	Exception Handling in IA MMX™ Application Code	3-8
3.3.7.	Register Mapping	3-8

CHAPTER 4

SYSTEM PROGRAMMING MODEL

4.1.	CONTEXT SWITCHING.....	4-1
4.1.1.	Cooperative Multitasking Operating System	4-1
4.1.2.	Preemptive Multitasking Operating System.....	4-1
4.2.	EXCEPTIONS.....	4-3
4.3.	COMPATIBILITY WITH EXISTING SOFTWARE ENVIRONMENTS	4-4
4.3.1.	Register Aliasing	4-4
4.3.2.	The Effect of Floating-Point and MMX™ Instructions on the Floating-Point Tag Word.....	4-7
4.3.2.1.	ALIASING SUMMARY	4-8
4.3.3.	Context Switch Support.....	4-8
4.3.4.	Floating-Point Exceptions.....	4-8
4.3.5.	Debugging	4-9
4.3.6.	Emulation of the Instruction Set.....	4-9
4.3.7.	Exception handling in Operating Systems.....	4-9

CHAPTER 5

INTEL ARCHITECTURE MMX™ INSTRUCTION SET

5.1. INSTRUCTION SYNTAX.....	5-1
5.2. INSTRUCTION FORMAT	5-2
5.3. NOTATIONAL CONVENTIONS.....	5-3
5.4. HOW TO READ THE INSTRUCTION SET PAGES.....	5-4
EMMS—Empty MMX State.....	5-8
MOVD—Move 32 Bits.....	5-10
MOVQ—Move 64 Bits	5-12
PACKSSWB /PACKSSDW—Pack with Signed Saturation	5-14
PACKUSWB—Pack with Unsigned Saturation	5-16
PADDB/PADDW/PADDQ—Packed Add	5-18
PADDSB/PADDSD—Packed Add with Saturation.....	5-21
PADDUSB/PADDUSW—Packed Add Unsigned with Saturation	5-23
PAND—Bitwise Logical And.....	5-26
PANDN—Bitwise Logical And Not.....	5-28
PCMPEQB/PCMPEQW/PCMPEQD—Packed Compare for Equal.....	5-30
PCMPGTB/PCMPGTW/PCMPGTD—Packed Compare for Greater Than	5-33
PMADDWD—Packed Multiply and Add	5-36
PMULHW—Packed Multiply High.....	5-38
PMULLW—Packed Multiply Low.....	5-40
POR—Bitwise Logical Or.....	5-42
PSLLW/PSLLD/PSLLQ—Packed Shift Left Logical	5-44
PSRAW/PSRAD—Packed Shift Right Arithmetic.....	5-47
PSRLW/PSRLD/PSRLQ—Packed Shift Right Logical.....	5-50
PSUBB/PSUBW/PSUBD—Packed Subtract.....	5-53
PSUBSB/PSUBSW—Packed Subtract with Saturation	5-56
PSUBUSB/PSUBSW—Packed Subtract Unsigned with Saturation	5-58
PUNPCKHBW/PUNPCKHWD/PUNPCKHDQ—Unpack High Packed Data	5-60
PUNPCKLBW/PUNPCKLWD/PUNPCKLDQ—Unpack Low Packed Data.....	5-63
PXOR—Bitwise Logical Exclusive OR.....	5-66

APPENDIX A

IA MMX™ INSTRUCTION SET SUMMARY

APPENDIX B

IA MMX™ INSTRUCTION FORMATS AND ENCODINGS

APPENDIX C

ALPHABETICAL LIST OF IA MMX™ INSTRUCTION SET MNEMONICS

APPENDIX D

IA MMX™ INSTRUCTION SET OPCODE MAP

Figures

Figure	Title	Page
2-1.	Packed Data Types.....	2-2
2-2.	MMX™ Register Set	2-3
3-1.	Eight Packed Bytes in Memory (at address 1000H)	3-2
4-1.	Example of FP and MMX State Saving in Operating System.....	4-2
4-2.	Aliasing of MMX™ to Floating-Point Registers	4-5
4-3.	Mapping of MMX™ Registers to Floating Point Registers	4-6
5-1.	Floating Point Tag Word Format.....	5-8
B-1.	Key to Codes for Datatype Cross-Reference	B-3

Tables

Table	Title	Page
2-1.	Data Range Limits for Saturation.....	2-4
3-1.	IA MMX™ Instruction Behavior with Prefixes Used by Application Programs.....	3-3
4-1.	Effect of the FP and MMX Instructions on the FP Tag Word	4-7
4-2.	Effects of MMX™ Instruction on FP State.....	4-8
A-1.	IA MMX Instruction Set Summary, Grouped into Functional Categories	A-2
B-1.	Encoding of Granularity of Data (gg) Field	B-1
B-2.	Encoding of 32-bit General Purpose (reg) Field for Register-to-Register Operations	B-2
B-3.	Encoding of 64-bit MMX™ Register (mmxreg) Field	B-2
B-4.	IA MMX Instruction Formats and Encodings	B-3
C-1.	IA MMX™ Instruction Set Mnemonics	C-1
D-1.	Opcode Map (First Byte is (0FH)	D-3
D-2.	Opcodes Determined by Bits 5, 4, 3 of Mod R/M Byte	D-5

Examples

Example	Title	Page
3-1.	Partial sequence of IA MMX™ technology detection by CPUID.....	3-4
3-2.	Floating-point and MMX™ Code.....	3-7