Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

MOTOROLA

MC68020
MC68EC020

MICROPROCESSORS USER'S MANUAL

First Edition

PREFACE

The *M68020 User's Manual* describes the capabilities, operation, and programming of the MC68020 32-bit, second-generation, enhanced microprocessor and the MC68EC020 32-bit, second-generation, enhanced embedded microprocessor.

Throughout this manual, “MC68020/EC020” is used when information applies to both the MC68020 and the MC68EC020. “MC68020” and “MC68EC020” are used when information applies only to the MC68020 or MC68EC020, respectively.

For detailed information on the MC68020 and MC68EC020 instruction set, refer to M68000PM/AD, *M68000 Family Programmer's Reference Manual*.

This manual consists of the following sections:

NOTE

In this manual, *assert* and *negate* are used to specify forcing a signal to a particular state. In particular, *assertion* and *assert* refer to a signal that is active or true; *negation* and *negate* indicate a signal that is inactive or false. These terms are used independently of the voltage level (high or low) that they represent.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

Section 6
Exception Processing

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

Paragraph Number	Title	Page Number

Section 8
Instruction Execution Timing

Section 9
Applications Information

TABLE OF CONTENTS (Concluded)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Concluded)

LIST OF TABLES

LIST OF TABLES (Continued)

MC68020/EC020 ACRONYM LIST

BCD — Binary-Coded Decimal
CAAR — Cache Address Register
CACR — Cache Control Register
CCR — Condition Code Register
CIR — Coprocessor Interface Register
CMOS — Complementary Metal Oxide Semiconductor
CPU — Central Processing Unit
CQFP — Ceramic Quad Flat Pack
DDMA — Dual-Channel Direct Memory Access
DFC — Destination Function Code Register
DMA — Direct Memory Access
DRAM — Dynamic Random Access Memory
FPCP — Floating-Point Coprocessor
HCMOS — High-Density Complementary Metal Oxide Semiconductor
IEEE — Institute of Electrical and Electronic Engineers
ISP — Interrupt Stack Pointer
LMB — Lower Middle Byte
LRAR — Limited Rate Auto Request
LSB — Least Significant Byte
MMU — Memory Management Unit
MPU — Microprocessor Unit
MSB — Most Significant Byte
MSP — Master Stack Pointer
NMOS — n-Type Metal Oxide Semiconductor
PAL — Programmable Array Logic
PC — Program Counter
PGA — Pin Grid Array
PMMU — Paged Memory Management Unit
PPGA — Plastic Pin Grid Array
PQFP — Plastic Quad Flat Pack
RAM — Random Access Memory
SFC — Source Function Code Register
SP — Stack Pointer
SR — Status Register
SSP — Supervisor Stack Pointer
SSW — Special Status Word
UMB — Upper Middle Byte
USP — User Stack Pointer
VBR — Vector Base Register
VLSI — Very Large Scale Integration

SECTION 1 INTRODUCTION

The MC68020 is the first full 32-bit implementation of the M68000 family of microprocessors from Motorola. Using VLSI technology, the MC68020 is implemented with 32-bit registers and data paths, 32-bit addresses, a rich instruction set, and versatile addressing modes.

The MC68020 is object-code compatible with earlier members of the M68000 family and has the added features of new addressing modes in support of high-level languages, an on-chip instruction cache, and a flexible coprocessor interface with full IEEE floating-point support (the MC68881 and MC68882). The internal operations of this microprocessor operate in parallel, allowing multiple instructions to be executed concurrently.

The asynchronous bus structure of the MC68020 uses a nonmultiplexed bus with 32 bits of address and 32 bits of data. The processor supports a dynamic bus sizing mechanism that allows the processor to transfer operands to or from external devices while automatically determining device port size on a cycle-by-cycle basis. The dynamic bus interface allows access to devices of differing data bus widths, in addition to eliminating all data alignment restrictions.

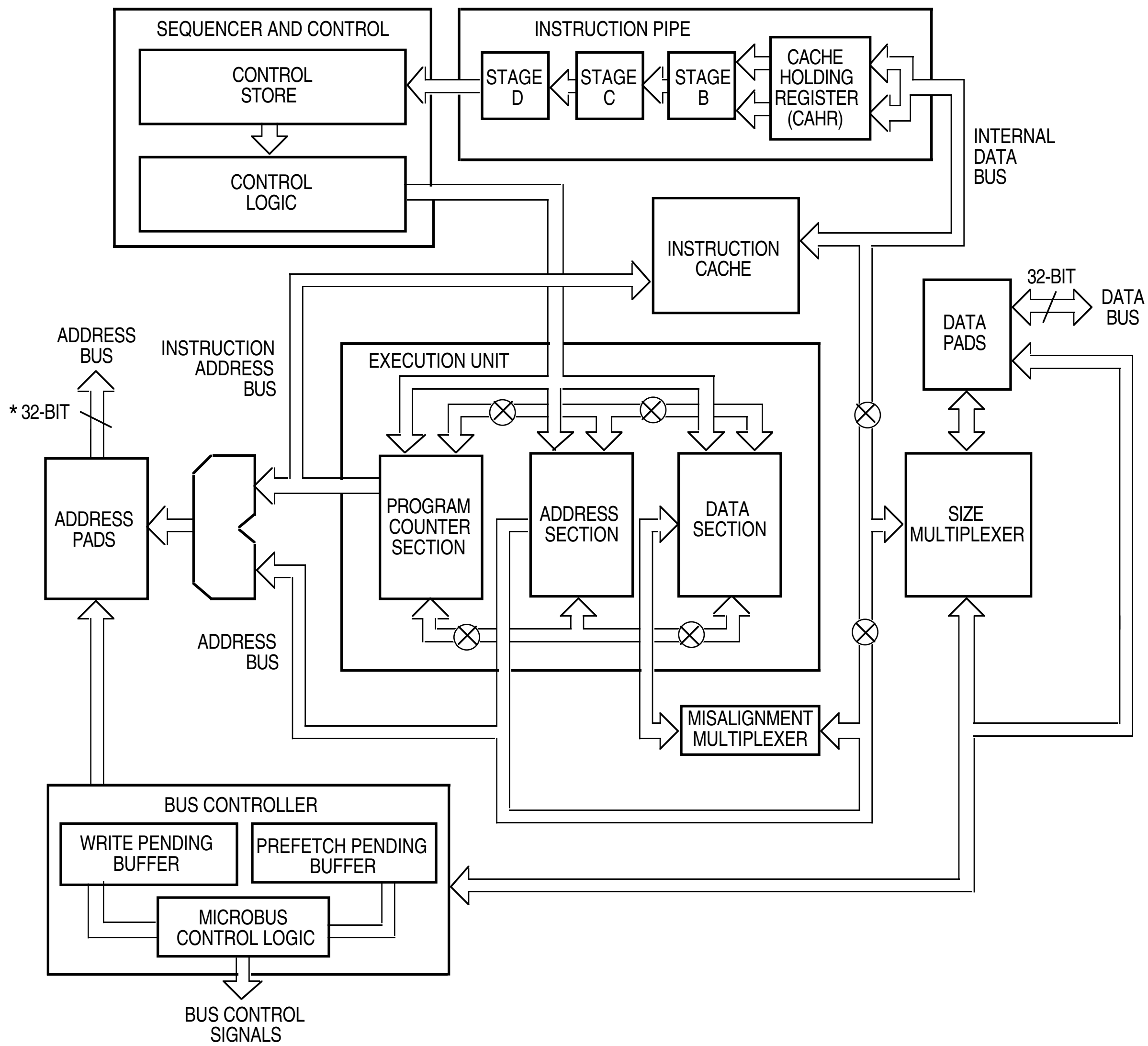
The MC68EC020 is an economical high-performance embedded microprocessor based on the MC68020 and has been designed specifically to suit the needs of the embedded microprocessor market. The major differences in the MC68EC020 and the MC68020 are that the MC68EC020 has a 24-bit address bus and does not implement the following signals: ECS, OCS, DBEN, IPEND, and BGACK. Also, the available packages and frequencies differ for the MC68020 and MC68EC020 (see **Section 11 Ordering Information and Mechanical Data**.) Unless otherwise stated, information in this manual applies to both the MC68020 and the MC68EC020.

1.1 FEATURES

The main features of the MC68020/EC020 are as follows:

- Object-Code Compatible with Earlier M68000 Microprocessors
- Addressing Mode Extensions for Enhanced Support of High-Level Languages
- New Bit Field Data Type Accelerates Bit-Oriented Applications—e.g., Video Graphics
- An On-Chip Instruction Cache for Faster Instruction Execution
- Coprocessor Interface to Companion 32-Bit Peripherals—the MC68881 and MC68882 Floating-Point Coprocessors and the MC68851 Paged Memory Management Unit
- Pipelined Architecture with High Degree of Internal Parallelism Allowing Multiple Instructions To Be Executed Concurrently
- High-Performance Asynchronous Bus Is Nonmultiplexed and Full 32 Bits
- Dynamic Bus Sizing Efficiently Supports 8-/16-/32-Bit Memories and Peripherals
- Full Support of Virtual Memory and Virtual Machine
- Sixteen 32-Bit General-Purpose Data and Address Registers
- Two 32-Bit Supervisor Stack Pointers and Five Special-Purpose Control Registers
- Eighteen Addressing Modes and Seven Data Types
- 4-Gbyte Direct Addressing Range for the MC68020
- 16-Mbyte Direct Addressing Range for the MC68EC020
- Selection of Processor Speeds for the MC68020: 16.67, 20, 25, and 33.33 MHz
- Selection of Processor Speeds for the MCEC68020: 16.67 and 25 MHz

A block diagram of the MC68020/EC020 is shown in Figure 1-1.



* 24-Bit for MC68EC020

Figure 1-1. MC68020/EC020 Block Diagram

1.2 PROGRAMMING MODEL

The programming model of the MC68020/EC020 consists of two groups of registers, the user model and the supervisor model, that correspond to the user and supervisor privilege levels, respectively. User programs executing at the user privilege level use the registers of the user model. System software executing at the supervisor level uses the control registers of the supervisor level to perform supervisor functions.

As shown in the programming models (see Figures 1-2 and 1-3), the MC68020/EC020 has 16 32-bit general-purpose registers, a 32-bit PC two 32-bit SSPs, a 16-bit SR, a 32-bit VBR, two 3-bit alternate function code registers, and two 32-bit cache handling (address and control) registers.

The user programming model remains unchanged from earlier M68000 family microprocessors. The supervisor programming model supplements the user programming model and is used exclusively by MC68020/EC020 system programmers who utilize the supervisor privilege level to implement sensitive operating system functions. The supervisor programming model contains all the controls to access and enable the special features of the MC68020/EC020. All application software, written to run at the nonprivileged user level, migrates to the MC68020/EC020 from any M68000 platform without modification.

Registers D7–D0 are data registers used for bit and bit field (1 to 32 bits), byte (8 bit), word (16 bit), long-word (32 bit), and quad-word (64 bit) operations. Registers A6–A0 and the USP, ISP, and MSP are address registers that may be used as software stack pointers or base address registers. Register A7 (shown as A7 in Figure 1-2 and as A7′ and A7″ in Figure 1-3) is a register designation that applies to the USP in the user privilege level and to either the ISP or MSP in the supervisor privilege level. In the supervisor privilege level, the active stack pointer (interrupt or master) is called the SSP. In addition, the address registers may be used for word and long-word operations. All of the 16 general-purpose registers (D7–D0, A7–A0) may be used as index registers.

The PC contains the address of the next instruction to be executed by the MC68020/EC020. During instruction execution and exception processing, the processor automatically increments the contents of the PC or places a new value in the PC, as appropriate.

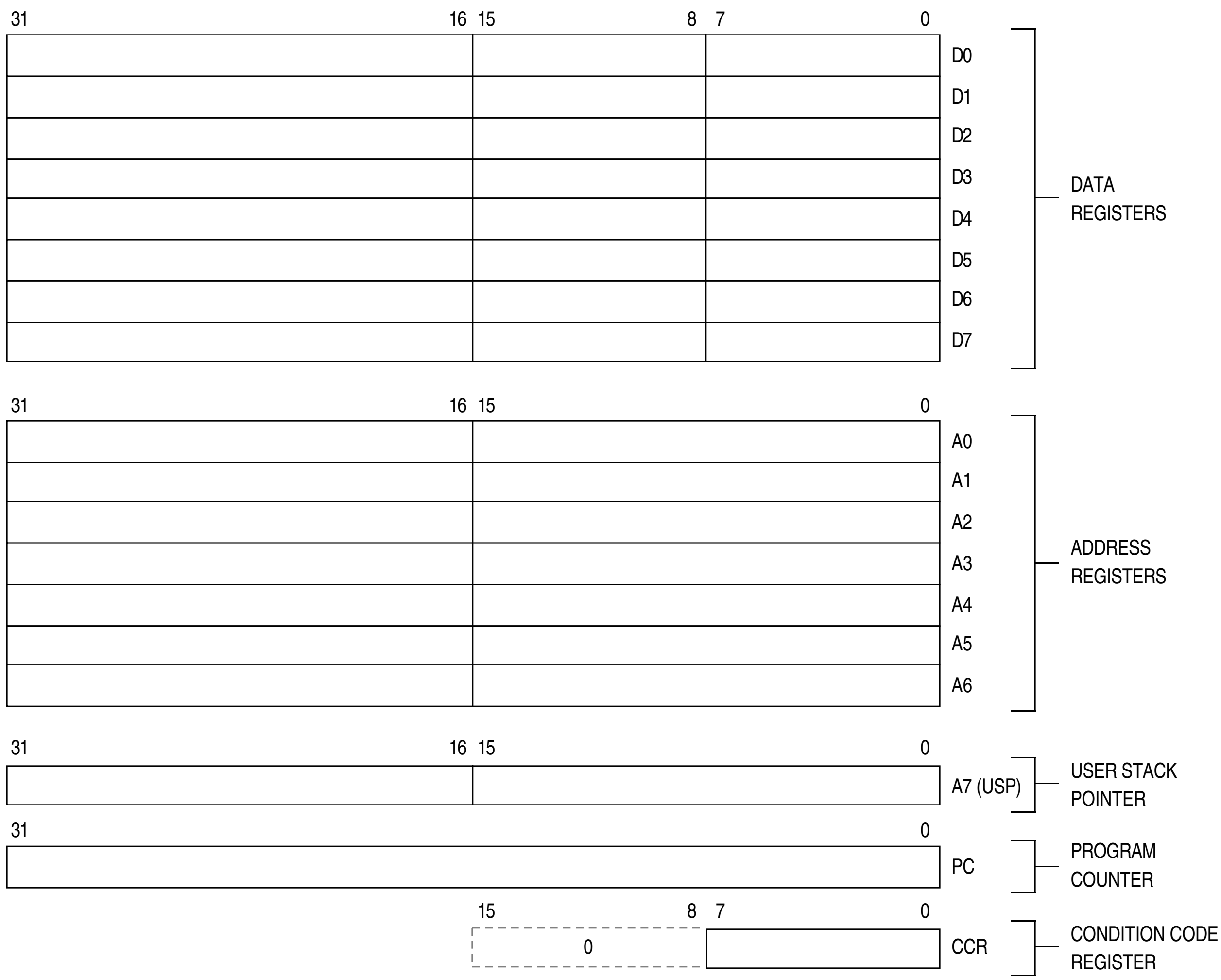


Figure 1-2. User Programming Model

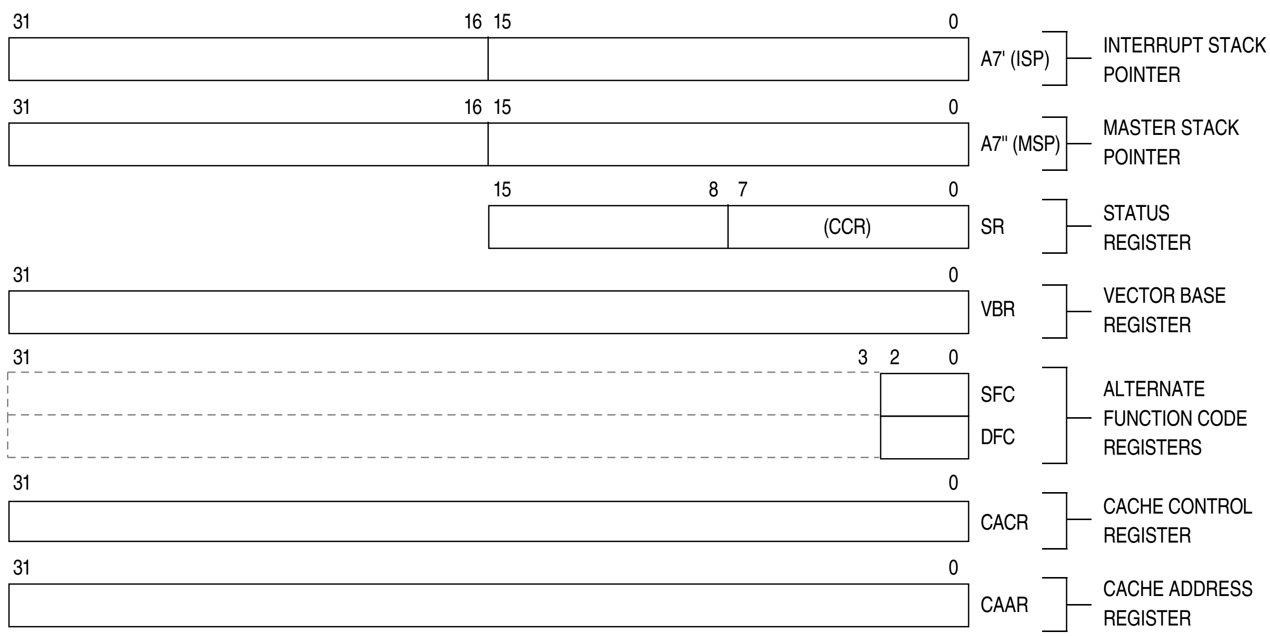


Figure 1-3. Supervisor Programming Model Supplement

The SR (see Figure 1-4) stores the processor status. It contains the condition codes that reflect the results of a previous operation and can be used for conditional instruction execution in a program. The condition codes are extend (X), negative (N), zero (Z), overflow (V), and carry (C). The user byte, which contains the condition codes, is the only portion of the SR information available in the user privilege level, and it is referenced as the CCR in user programs. In the supervisor privilege level, software can access the entire SR, including the interrupt priority mask (three bits) and control bits that indicate whether the processor is in:

1. One of two trace modes (T1, T0)
2. Supervisor or user privilege level (S)
3. Master or interrupt mode (M)

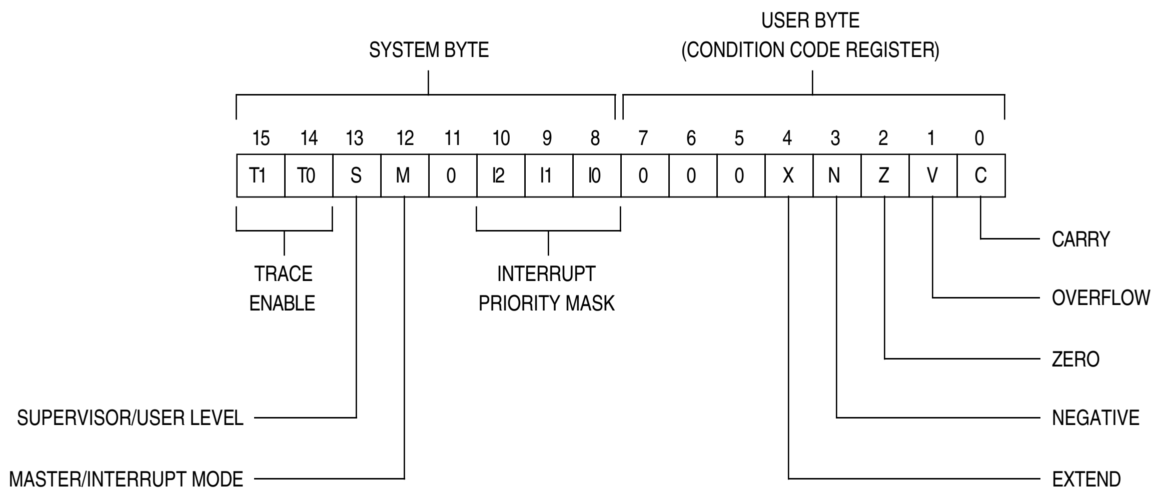


Figure 1-4. Status Register (SR)

The VBR contains the base address of the exception vector table in memory. The displacement of an exception vector is added to the value in this register to access the vector table.

The alternate function code registers, SFC and DFC, contain 3-bit function codes. For the MC68020, function codes can be considered extensions of the 32-bit linear address that optionally provide as many as eight 4-Gbyte address spaces; for the MC68EC020, function codes can be considered extensions of the 24-bit linear address that optionally provide as many as eight 16-Mbyte address spaces. Function codes are automatically generated by the processor to select address spaces for data and program at the user and supervisor privilege levels and to select a CPU address space for processor functions (e.g., coprocessor communications). Registers SFC and DFC are used by certain instructions to explicitly specify the function codes for operations.

The CACR controls the on-chip instruction cache of the MC68020/EC020. The CAAR stores an address for cache control functions.

1.3 DATA TYPES AND ADDRESSING MODES OVERVIEW

For detailed information on the data types and addressing modes supported by the MC68020/EC020, refer to M68000PM/AD, *M68000 Family Programmer's Reference Manual*.

The MC68020/EC020 supports seven basic data types:

1. Bits
2. Bit Fields (Fields of consecutive bits, 1–32 bits long)
3. BCD Digits (Packed: 2 digits/byte, Unpacked: 1 digit/byte)
4. Byte Integers (8 bits)
5. Word Integers (16 bits)
6. Long-Word Integers (32 bits)
7. Quad-Word Integers (64 bits)

In addition, the MC68020/EC020 instruction set supports operations on other data types such as memory addresses. The coprocessor mechanism allows direct support of floating-point operations with the MC68881 and MC68882 floating-point coprocessors as well as specialized user-defined data types and functions.

The 18 addressing modes listed in Table 1-1 include nine basic types:

1. Register Direct
2. Register Indirect
3. Register Indirect with Index
4. Memory Indirect
5. PC Indirect with Displacement
6. PC Indirect with Index
7. PC Memory Indirect
8. Absolute
9. Immediate

The register indirect addressing modes have postincrement, predecrement, displacement, and index capabilities. The PC modes have index and offset capabilities. Both modes are extended to provide indirect reference through memory. In addition to these addressing modes, many instructions implicitly specify the use of the CCR, stack pointer, and/or PC.

Table 1-1. Addressing Modes

Addressing Modes	Syntax

Register Direct
  Data
  Address
	
	Dn
	An

Register Indirect
  Address
  Address with Postincrement
  Address with Predecrement
  Address with Displacement
	
	(An)
	(An)+
	–(An)
	(d_{16}, An)

Address Register Indirect with Index
  8-Bit Displacement
  Base Displacement
	
	(d_8, An, Xn)
	(bd, An, Xn)

Memory Indirect
  Postindexed
  Preindexed
	
	([bd, An], Xn, od)
	([bd, An, Xn], od)

PC Indirect with Displacement	(d_{16}, PC)

PC Indirect with Index
  8-Bit Displacement
  Base Displacement
	
	(d_8, PC, Xn)
	(bd, PC, Xn)

PC Indirect
  Postindexed
  Preindexed
	
	([bd, PC], Xn, od)
	([bd, PC, Xn], od)

Absolute Data Addressing
  Short
  Long
	
	(xxx).W
	(xxx).L

Immediate	#<data>

NOTE:

Dn = Data Register, D7–D0

An = Address Register, A7–A0

d_8, d_{16} = A twos complement or sign-extended displacement added as part of the effective address calculation; size is 8 (d_8) or 16 (d_{16}) bits; when omitted, assemblers use a value of zero.

Xn = Address or data register used as an index register; form is Xn.SIZE*SCALE, where SIZE is .W or .L (indicates index register size) and SCALE is 1, 2, 4, or 8 (index register is multiplied by SCALE); use of SIZE and/or SCALE is optional.

bd = A twos-complement base displacement; when present, size can be 16 or 32 bits.

od = Outer displacement added as part of effective address calculation after any memory indirection; use is optional with a size of 16 or 32 bits.

PC = Program Counter

<data> = Immediate value of 8, 16, or 32 bits

( ) = Effective Address

[ ] = Use as indirect access to long-word address.