

微型计算机



microcomputer

MC 68000 十六位 微型计算机

杜 毅 仁 杨 克 忠

上海交通大学

1984-1

(总 36 期)

前 言

本书是为 MIC-68K 单板微型机用户编写的一份讲义，分成 MC68000 和 MIC-68K 这样上、下二部分。

MC68000 是目前国际上功能最强、应用最广的三种先进的 16 位微处理器之一，也是我国采用的二个微型机优选机种之一。MIC-68K 是上海交通大学在分析解剖了国外产品以后自行设计的一种高性能的 68000 单板机，该机在软件上与 Motorola 公司的 MEX68KECB 单板兼容。

本书共包括五章，其中第一、二、三章介绍 MC68000 的指令系统、程序设计、硬件操作和系统组成，第四、五章介绍 MIC-68K 的硬件、软件以及操作使用，末尾还带有四个非常实用的附录。在介绍 MC68000 的章节中，笔者尽量从原理上和原则上进行广泛的和一般性的介绍，而并不局限于某一个具体的系统，因此本书对于那些非 MIC-68K 的用户但希望学习 MC68000 的读者仍然是一本有益的参考书。在材料组织方面，每个章节都是笔者在学习 MC68000 和设计 MIC-68K 的基础上写成的，并且书中所采用的硬件框图、软件程序及上机操作都经过实践(或以实体为基础)，故本书对于初学者来说可能较其翻他译本或编译本更容易阅读和理解。

由于 MC68000 必竟是一个很新的元件，各种参考资料和使用经验也相对缺乏，而要在短短一年多一点的时间内完成从学习、熟悉、设计、开发直到写出完备资料的全部过程确实是太匆忙了。故恳请读者批评指正，以便重印时更正。

编者 1983.10 于上海交通大学

目 录

引 言	(1)
第一章 MC68000 指令系统和程序设计	(7)
§ 1-1 操作数据	(7)
§ 1-2 寻址方式	(9)
§ 1-3 状态寄存器和优先状态	(14)
§ 1-4 指令系统概述	(17)
§ 1-5 指令详细说明	(23)
§ 1-6 汇编语言	(111)
§ 1-7 汇编程序设计	(115)
第二章 MC68000 硬件操作	(126)
§ 2-1 引脚定义	(126)
§ 2-2 总线操作	(128)
§ 2-3 异常处理(EXCEPTION PROCESSING)	(137)
第三章 以 MC68000 为 MPU 的系统组成	(148)
§ 3-1 基本系统	(148)
§ 3-2 与 M6800 系列外围芯片的配合	(150)
§ 3-3 带有存贮管理部件的 68000 系统	(155)
§ 3-4 若干 68000 系统的实例	(159)
第四章 MIC-68K 的硬件结构和系统连接	(166)
§ 4-1 硬件结构	(166)
§ 4-2 系统的硬件连接和开机过程	(173)
§ 4-3 检查和维护	(175)
第五章 MIC-68K 监控命令	(177)
§ 5-1 监控命令的一些约定	(177)
§ 5-2 监控命令详述	(179)
§ 5-3 常驻汇编/反汇编软件	(196)
§ 5-4 TRAP14 系统功能调用	(202)
§ 5-5 监控程序的操作信息	(211)
附 录	
附录A 符号说明	(212)
附录B MC68000 机电性能参数	(213)
附录C MC68000 指令执行时间	(220)
附录D MC68000 系列其他芯片介绍	(227)

引言

MC68000 (图 1)是目前国际上最流行的三种先进的 16 位微处理器之一, MC68000 于 1979 年研制成功, 1980 年进入市场。MC68000 不仅在其本身具有强大的硬件资源和指令系统, 而且在外部享有丰富的配套芯片和支持软件。正因为如此, 大批原小型机和 8 位微型机的用户纷纷转向 MC68000, 各种以 68000 为 MPU 的单板机和微型机系统也已成为国内外应用面最广、用户最多的微型机种之一。

MC68000 在设计上吸取了 PDP-11 和 IBM360 等国际流行的中小型计算机系统的成功经验, 同时又继承和发展了 8 位/16 位微型机的惯例和传统。MC68000 大胆地突破了与 8 位 M6800 兼容这一束缚, 从而给芯片的统系设计、逻辑设计、电路设计以及工艺设计带来了充分的自由。MC68000 在功能上较 M6800 提高 10 倍以上, 这将为用户提供了更为广泛的应用舞台(图 2)。



图 1 MC68000 引脚配置图

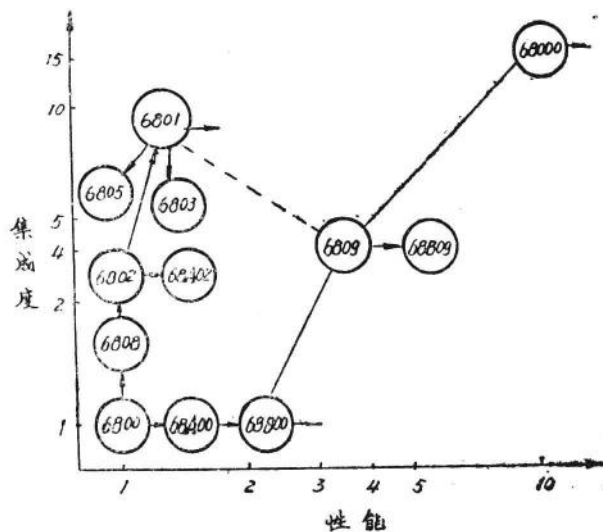


图 2 68 系列产品的发展

MC68000 在制造工艺上采用 HMOS(高密度、短沟道、金属—氧化物半导体)工艺, 该工艺的速度功耗乘积较 NMOS 高 4 倍, 电路密度高 2 倍, 片内共集成了 68000 个管子, 因而是一种 VLSI 超大规模集成器件。

MC68000 的主要资源有:

- 18 个 32 位寄存器(包括 8 个数据寄存器、8 个地址寄存器和 1 个 16 位状态寄存器)

- 直接寻址 16 M字节
- 56 种基本指令
- 5 种主要的数据类型
- 存贮器映象 I/O
- 14 种寻址方式
- 能与 M6800 系列外设直接连接
- 64 条引脚双列直插式封装
- 主频 4MHz/6MHz/8MHz/10MHz

MC68000 用户所能使用的寄存器如图 3 所示。32 位的数据寄存器($D_0 \sim D_7$)可作 32 位长字(第 31~0 位)、16 位字(第 15~0 位)或 8 位字节(第 7~0 位)三种长度使用。八个数据寄存器功能完全相同,且都可作累加器使用。32 位的地址寄存器($A_0 \sim A_7$)能以 32 位和 16 位二种长度使用,地址寄存器 $A_0 \sim A_6$ 功能相同,通用于全部寻址方式。 A_7 不但具有 $A_0 \sim A_6$ 的功能,而且还兼作系统/用户堆栈指针:当系统状态时, A_7 为系统堆栈指针;用户状态时, A_7 为用户堆栈指针。32 位程序计数器 PC 实际提供 24 位有效地址,故能直接寻址 16M 字节。当将之与功能码 $FC_0 \sim FC_2$ 配合时,寻址范围可达 64M 字节。

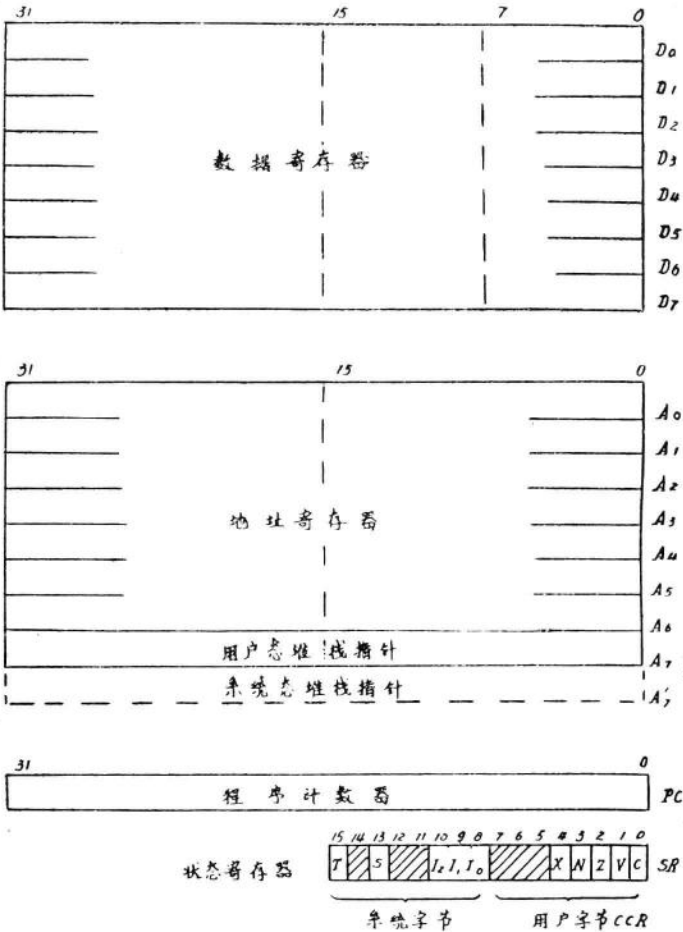


图 3 MC68000 的寄存器

MC68000 的状态寄存器 SR 为 16 位, 分成高八位字节(系统字节)和低八位字节(用户字节)两段。系统字节包括中断级别指示($I_2I_1I_0$)、系统状态/用户状态指示(S)和跟踪位(T)。用户字节又称为条件码 CCR, 其中包括进位位(C)、溢出位(V)、零位(Z)、符号位(N)和扩展位(X)。

与其他的 8 位/16 位微处理机相比较, MC68000 有如下的一些特点:

1. 硬件功能强大。这是很明显的, 单从 MC68000 MPU 芯片内包含的 18 个 32 位寄存器和 16 兆直接寻址能力这一点来看, 它就足以跟小型机比美。MC68000 内部数据总线 32 位, 因而又是一种内部 32 位/外部 16 位的准 32 位超级微处理机。

2. 结构的一致性。这不仅体现在硬件的组成(如 8 个数据寄存器和 8 个地址寄存器字长统一, 互换通用), 而且体现在指令系统和寻址方式的规整性。例如: 通过改变寻址方式, 带操作数的指令可同样适用于寄存器对寄存器、寄存器对存贮器和存贮器对存贮器之间的操作, 并且存贮器操作中还包括映象的外设 I/O。因此, MC68000 的所有指令可压缩成一个由 56 条基本指令组成的内核, 通过有规律的扩充便成为一个包含千余条指令的功能强大的指令系统。这种强化和浓缩的指令系统, 既增强了指令的功能, 又使记忆和使用都十分方便。

3. 支持先进的操作系统和模块化程序设计。模块化程序设计是微型计算机中最常用的方法。模块化程序设计需要将每一个功能段、每一个进程分解成一个个尺寸更小的子程序模块, 而每一模块应该是良好定义的。模块化程序设计要求有先进的操作系统、结构化的汇编和模块化的高级语言, 而这一些软设备的基础则需要提供软件模块以高效的参数(数据)传送。MC68000 指令系统有力地支持了程序设计的这一需要。例如: LINK 和 UNLK 指令能连系二个不同堆栈数据区, MOVEM 指令能一次从堆栈中压入或弹出多个数据等等, 这些都极大地减少了子程序调用等操作的开销。系统共提供 255 种异常处理、7 级(192 个)硬件中断、16 个软件 TRAP 指令, 这些都将成为操作系统设计的宝贵资源。

此外, MC68000 还设有二种运行状态: 系统状态和用户状态。并对所有的指令赋予“权”。这一功能对于先进的操作系统以及存贮管理无疑是极为有利的。正因为如此, 大多数以 MC68000 为 MPU 的微型机系统都配有 UNIX 操作系统和 PASCAL 高级语言。

4. 提高了组成系统的可靠性。MC68000 采用软硬件结合的方式提供了多种查错处理手段, 它们是:

- 总线出错(访存贮器无回答等)
- 地址出错(存取奇数地址单元的字或长字)
- 非法指令
- 不可执行的指令
- 除法中除数为 0
- 溢出
- 寄存器超出边界

5. 软硬件支持和与 M6800 外设兼容。在 MC68000 开发的同时, 还推出了各种支持软件和硬件, 如 EXORmacs68000 开发系统就是一个带有磁盘操作系统、汇编语言、多种高级语言的开发工具。为了提高构成系统的整机效率, MC68000 还有一套与之配合的外围芯片, 如存贮管理部件 68451、并行接口/定时器 68230 等。由于 MC68000 在结构上与

PDP-11 十分相似, 各种原在 PDP-11 系列机上运行的软件稍加改造即可移植到 68000 微型机系统中, 因此各种以 68000 为 MPU 的微型机可充分享用丰富的小型机软件, 这是其他机种所不具备的特点。

为了使 MC68000 能与原 M6800 系列的外围兼容, 芯片上专门设置了 E、VPA、VMA 等三个引脚信号, 这一机构使得原来使用 8 位外围芯片的系统和用户, 当采用了先进的 16 位 MC68000 MPU 以后, 仍可以保留其硬件和软件, 特别是宝贵的使用经验。

MIC-68K 是上海交通大学自行研制的一种高性能的 16 位单板微型机。MIC-68K 采用先进的 MC68000 作为中央处理器, 在软件上与 Motorola 公司的 MEX68KECB 单板完全兼容, 并且在硬件配置和软件操作等方面有所扩充。该机的主要性能指标是:

硬件

- 16 位微处理器 MC68000L4(内部寄存器 32 位, 直接寻址 16 兆字节)
- 晶振 24MC, MPU 主频 4MC
- 32K 字节动态 RAM($M4116 \times 16$, 其中用户区 30K)
- 32K 字节 EPROM(2764×4 , 其中 16K 供用户编程)
- 二对串行编口($M6850 \times 2$), 波特率 9600~75 八档选用
- 六组并行端口($M6821 \times 3$)
- 三组定时器($M6840 \times 1$)
- 56 总线引出, 供系统扩充
- 56 线插头三个, 50 线插头座引线区一个

软件

- 固化监控程序(16K 字节, 含监控命令 40 条)
- 常驻编辑/汇编/反汇编软件
- TRAP14 系统调用

可选配外设

- CRT 终端(RS-232 接口标准)
- 主机 HOST 通讯(RS-232 接口标准)
- 行式打印机(Centronics 接口标准)
- 音频磁带录音机(调频方式, 2400 波特)
- EPROM 编程器

其他

- 电源: +5V(500MA), +12V(50MA), -12V(50MA)
- 单板尺寸: $305 \times 200 \times 20\text{MM}^3$

MIC-68K 构成系统的硬件配置如图 4 所示。

在系统设计上, MIC-68K 有以下的一些特点:

1. MIC-68K 的设计思想是: 方便用户, 降低成本, 提高软硬件使用效率, 力求在系统设计上取得尽可能高的性能价格比。如在 MIC-68K 的存储器设计上, 通过对提高芯片集成度和降低造价成本这二个方面反复权衡, 决定采用先进的 8K 字节 EPROM 2764 作固定存储器。这样, 仅两片器件就能容纳下整个监控程序(共 16K), 另扩充二片 2764(16K)供

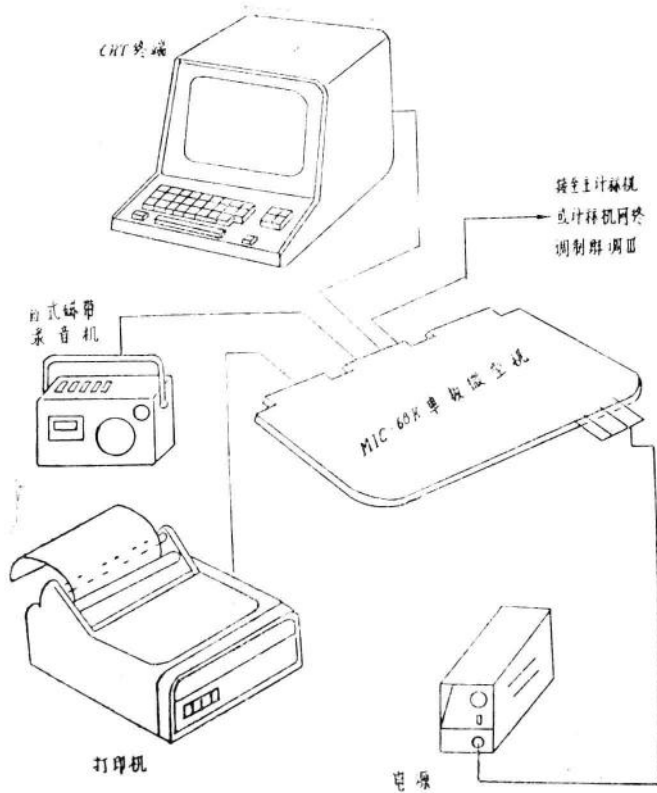


图 4 MIC-68K 单板微型机的系统构成

用户编程。板内读写存储器拟采用通用的 $16K \times 1$ 位 4116 动态 RAM。以 16 片 4116 组成一个 32K 字节的读写存储体。对比一般单板机所配备的 2~8K 静态 RAM 而言, MIC-68K 则可以为用户提供一个体积小、容量大、价格低、功耗小的存储资源。

2. 采用字长 8 位的外围芯片与 16 位 MPU 配接是 MIC-68K 的又一特点。这是考虑到目前已有的各种 M6800 配套外围芯片在制造技术上已经成熟、用户使用经验比较丰富、而且价格低廉、容易得到。本单板机选用异步串行 I/O 芯片 M6850 与 CRT 终端和 HOST 接口, 选用并行 I/O 芯片 M6821 配接行式打印机和盒式磁带机, 并选用定时器芯片 M6840 以提供 3 组定时/计数器。在电路设计中, 将这些芯片的数据线分别连入系统数据总线的高 8 位或低 8 位, 并且使二片 M6850 的地址前后连续, 这样一次读/写就能同时或分别对 CRT 终端和 HOST 进行输入输出。在定时设计上, 充分利用了 MC68000 微处理器总线操作定时的异步性能。总线传送由信号交换方式进行, 因而不需要使系统同步地进行取样。同时, EPROM 存储器和 RAM 存储器分别采用两组不同的定时。

在接口电路的设计方面, 尽量考虑与各种现有的国际标准兼容。每片与 CRT 终端和 HOST 配接的 M6850 器件的 I/O 线位上, 分别添加了电平驱动器 1488 和电平转换器 1489, 以使 MIC-68K 内部的 $0 \sim +3V$ TTL 逻辑电平转换成 $+3 \sim -3V$, 并通过在插件端子规定适当的引脚, 从而使之和 EIA 串行通讯标准 RS-232 标准兼容。在与打印机配接的 M6821 芯片输出端, 增加了若干 TTL 器件从提高对较长连线的行式打印机的驱动能力, 并专门选用一个端口提供联络信号, 使之与 Centronics 并行通讯标准兼容。通过这样的设计, 使 MIC-68K 能配接各种通用的 I/O 外部设备。

3. 16K 字节固化于 EPROM 的常驻监控程序。这是一个基于 CRT 终端作为人机交换窗口的多功能监控软件。由于采用了 CRT 终端,从而可以输入多参数多功能的监控命令。MIC-68K 一共有 40 条命令,分为存储器命令、寄存器命令、运行控制命令、I/O 控制命令以及辅助命令、单键命令等六大类。

命令行中可插入选用项,这是 MIC-68K 命令的一个特点。通过加入不同的选用项,可使同一条命令得到不同的扩充功能。例如,在 M 命令、S 命令之后插入选用项 B、W 或 L 可规定操作数字分别为 8 位字节、16 位字或 32 位长字。又如,在 M 命令或 D 命令之后添入选用项 DI,可调入汇编/反汇编软件。这类选用项将为用户提供极大的便利。

4. MIC-68K 的汇编/反汇编软件是系统监控程序的一部分。该软件也同监控程序一并被固化在 EPROM 中,是一种常驻的分行汇编反汇编兼编辑程序。与一般的 MC68000 汇编程序相比, MIC-68K 具有以下不同点:

① 调入的方式不同。该汇编/反汇编程序由 DI 选用项来进入:通过存储器显示命令(D; DI)和存储器修改命令(M; DI)调用反汇编;通过存储器修改命令(M; DI)的写入过程中调用汇编,并具有一定的编辑功能。

② 工作方式不同。该汇编/反汇编软件采用分行扫描工作方式。对输入的汇编源程序,分行独立进行汇编而后生成机器代码,并将生成的机器码存入存储器中。因此,该汇编/反汇编软件不保留汇编源程序,也没有语句标号或符号常数。

③ 控制方式简单。该汇编/反汇编软件仅有一条伪指令“DC.W”以在存储器中填入数据。软件中不设置宏指令和条件汇编,也只给出十分有限的出错显示信息。

不难看出, MIC-68K 的汇编/反汇编软件是十分简易的,但其执行速度却比一般的二次扫描方式为快,而且操作简便、直接,因此与 MIC-68K 的单板微型机的结构配置是十分和谐的。

5. 使用 TRAP 14 实现子程序调用。MIC-68K 的监控程序中含有丰富的软件资源,如 I/O 处理、数据/字符转换、二/十进制转换、内部初始化等。为此, MIC-68K 将各种常用的功能子程序进行编号列表,并采用陷井指令 TRAP #14 来调用。与常规的采用转子指令(如 BSR 或 JSR)调用相比较, TRAP 调用不但记忆方便(不必记忆地址而是记忆有规律的编号),而且使生成的用户程序的目的码仍可浮动。这一特点是常规的调用方式所不具备的。

基于 MC68000 和 MIC-68K,我们编写了这本书。在编写过程中,我们所设想的读者应对微处理器和微计算机的基础知识有一定的了解,如果他能有 8 位或 16 位微型机的上机经验那自然更好一些。本书的前一部分,主要阐述 MC68000 中央处理器芯片的有关内容,其中包括硬件结构、指令系统、寻址方式、汇编语言及编程特点、与外设的连接等等。本书的后一部分转入讲解 MIC-68K 单板微型计算机,其中包括系统的硬件设计、软件设计、监控命令、汇编/反汇编、以及 TRAP 14 子程序调用等。此外,书末还附有较为齐全的图表附录,以满足读者的实际应用之需要。

在材料的组织方面,尽量使每章具有相对的独立性。对于 MIC-68K 的用户,如果对 MC68000 已经有所了解,则可以跳过前几章而浏览第四、五章。在介绍 MC68000 软硬件的一、二、三章中,内容完全按照 MC68000 的原始资料编写,因此对于非 MIC-68K 的用户但需要了解 MC68000 的读者也有一定的参考价值。

第一章 MC 68000 指令系统和程序设计

MC68000 内部数据总线 32 位, 对外数据总线 16 位, 直接寻址能力 16 兆字节(24 位), 这一特点使它能有力地支持 8 位/16 位/32 位等多种字长的操作数据, 并提供 14 种不同的寻址方式。

§1-1 操作数据

MC68000 能处理以下种类的数据:

- 位 (Bit): 1 位二进制
- 字节 (Byte): 8 位二进制
- 字 (Word): 16 位二进制
- 长字 (LongWord): 32 位二进制(对于 24 位地址, 以高位补 0 同作长字处理)
- 二十进制数 (BCD Digit): 8 位二进制表示 2 个 BCD 数

当这些数据在寄存器中被处理时, 它们分别占用寄存器的低 8 位, 低 16 或全部(见图 1.1)。

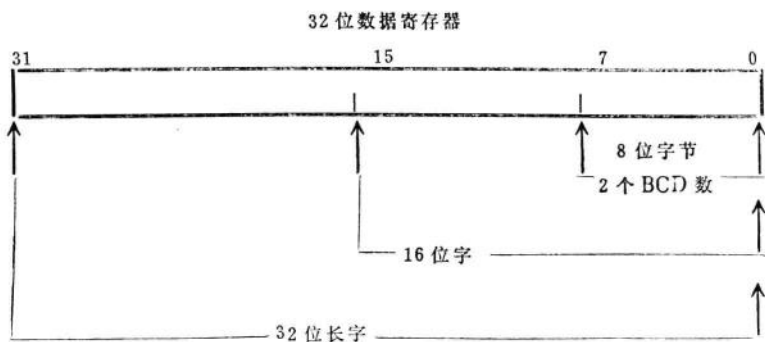


图 1.1 寄存器中的数据

而当这些数据存放在存贮区中时, 数据的物理结构(从硬件的角度上看)和逻辑结构(从程序设计的角度上看)是有所不同的。

从程序设计的角度上看, 字节、字、长字都是一个顺序排列的单元序列, 不同之处仅在于信息的长度和编址的约定。

字节是编址的最小单位, MC68000 可对 000000 到 \$FFFFFF 中任意一个地址直接寻址, 其逻辑结构见图 1.2。

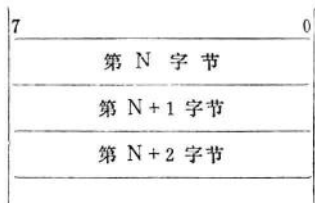


图 1.2 字节的逻辑序列

字是 68000 系统中数据的基本单位。1 个字等于 2 个字节，为了与字节的寻址方式统一，现规定：高 8 位采用偶数字节地址，低 8 位采用奇数字节地址，并选取 16 位字的高 8 位偶数字节地址作为该字的地址。因此，68000 的字地址范围是 000000 到 \$FFFFFE 中的偶数(图 1.3)。

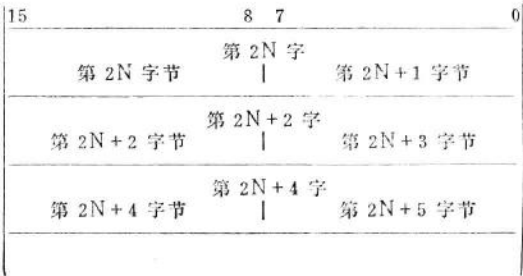


图 1.3 字的逻辑序列

长字等于 2 个字或 4 个字节(对 24 位地址，将高 8 位补 0，也作长字处理)。为了与上述寻址方式统一，规定其地址等于高 16 位字的地址。因此，68000 的长字地址范围是 000000 到 \$FFFFFC 中的偶数(图 1.4)。

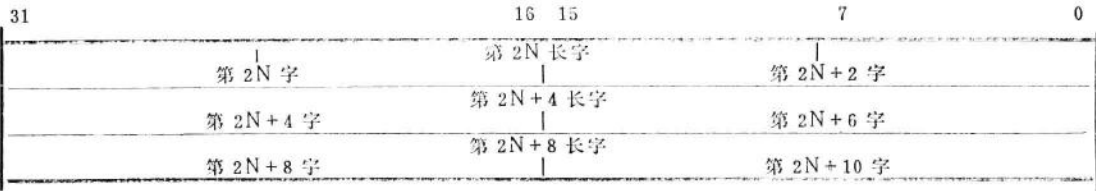


图 1.4 长字的逻辑序列

从硬件的角度来看，由于 68000MPU 的对外数据总线为 16 位，与之配套的存贮部件也必然以 16 位字宽作为基本读写单位。其物理结构是：16 位字占一个字，8 位字节占半个字(先高8位后低 8 位)，32 位长字占 2 个字(先高 16 位后低 16 位)。详见图 1.5。

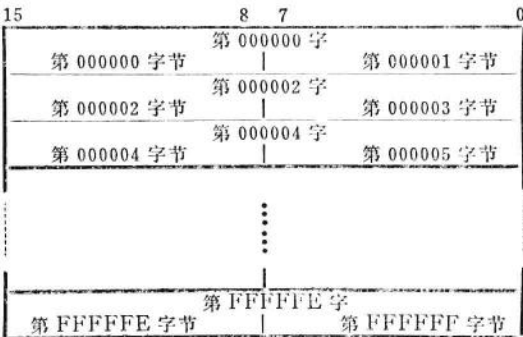


图 1.5 存贮器中数据的物理结构

- 这一结构特点使得 MC68000 在存贮器操作时：
- ① 访问 8 位字节与访问 16 位字所需时间相同，访问一个长字的时间等于访问二次字的时间。
 - ② 读写字或长字时，所指地址一定要为偶数，MC68000 中称之为“字边界”。当指令以奇数地址访问字/长字时，系统将作为地址出错。
- 以上这二点在用户编写程序时应充分给于重视。

§1-2 寻址方式

MC68000 为了适应 16 兆字节的寻址范围,采用了多种寻址方式。与其他 16 位微处理器相比,MC68000 除了具有寄存器直接寻址、寄存器间接寻址、立即地址等常规寻址方式外,还设有“前减量地址寄存器间址”和“后减量地址寄存器间址”二种多功能寻址方式。当与地址寄存器配合后可在系统内同时建立起八个堆栈或四个队列。为了缩短指令中寻址占用的字节数,MC68000 采用了一个由 6 位二进制位组成的有效地址段(Effective Address,缩写为 EA 段)。并设有立即快数这一简略的立即数表达方式。

MC68000 的指令有 2、4、6、8 字节四种长度,但第一、二字节恒为指令的操作码。当指令操作需要有操作数参加时,操作码段中便设有一个有效地址段来指示操作数的来源。有效地址段又分为两个小段:Mode 段(3 位)用于指示寻址类型,Register 段(3 位)用于指示寄存器编号(当 Mode = 111 时)或类型信息(当 Mode = 111 时)。图 1.6 是某个单指令的例子。

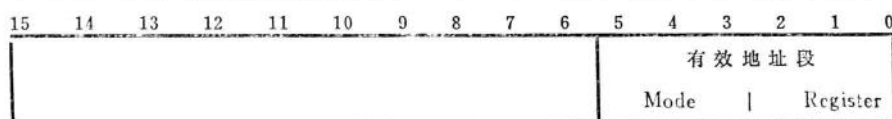


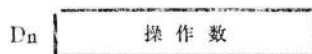
图 1.6 有效地址段

下面将对 14 种寻址方式分别加以说明。其中采用的符号可参见附录。

1. 数据寄存器直接寻址(Mode = 000, Register = n)

汇编符号: Dn

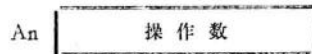
说明: 操作数在 Dn 中



2. 地址寄存器直接寻址(Mode = 001, Register = n)

汇编符号: An

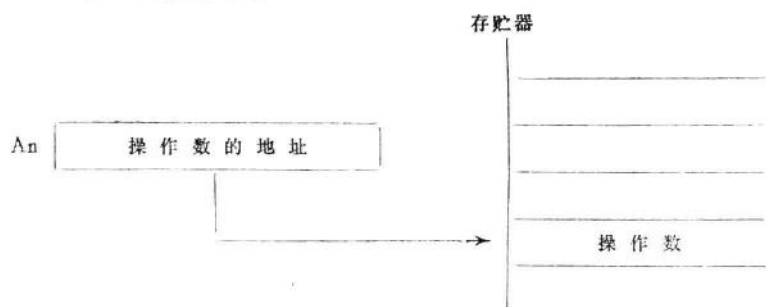
说明: 操作数在 An 中



3. 地址寄存器间接寻址(Mode = 010, Register = n)

汇编符号: (An)

说明: An 中为操作数的地址



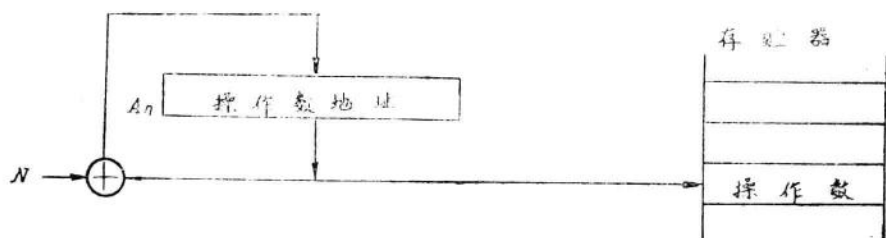
4. 后增量地址寄存器间接寻址(Mode = 011, Register = n)

汇编符号: (An) +

说明: An 中为操作数地址。

每进行一次读写以后,根据操作数的长度(字节、字或长字)而将 An 加以 1、2 或 4。但对于 A7 没有加 1 的操作,这是因为 A7 是系统/用户堆栈指针,

为了保证始终指向偶数“字边界”，即使是字节读写， A_7 也加 2 而不是 1。这样，当出现中断或陷井处理时，不会因为地址等信息(为长字和字)入栈而遇到 A_7 指向奇数地址发生错误。

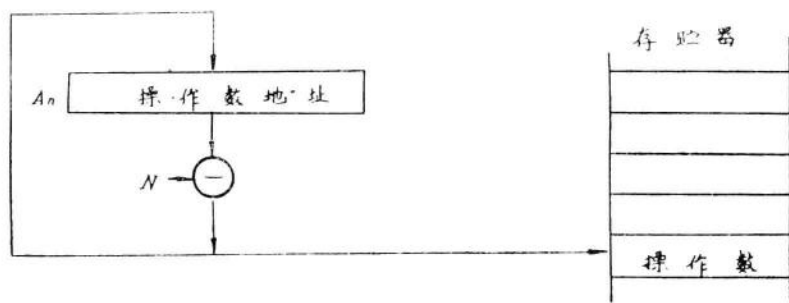


5. 前减量地址寄存器间接寻址 (Mode = 100, Register = n)

汇编符号: $-(A_n)$

说明: A_n 中为操作数地址。

进行读写以前，先根据操作数的长度(字节、字或长字)分别将 A_n 减去 1、2 或 4，然后再根据修改后的新地址读写操作数。但对于 A_7 没有减 1 的操作，即使读写字节也减去 2，以保证系统栈指针 A_7 始终指向偶数“字边界”。

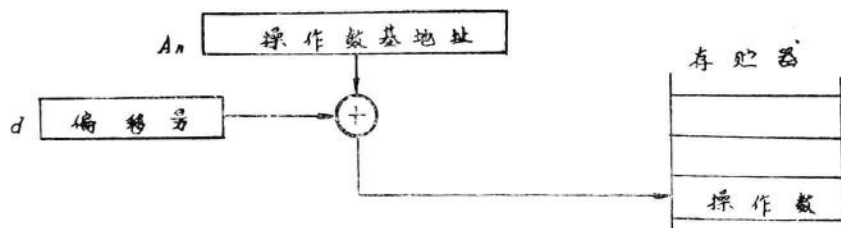


6. 带偏移地址寄存器间接寻址, (Mode = 101, Register = n)

汇编符号: $d(A_n)$

说明: 由 A_n 提供基地址, d 提供相对于基地址的偏移量, 将偏移量符号扩充后与基地相加得到操作数地址。相加结果不回送(A_n 不变)。

偏移量有长、短之分。短偏移占 1 字节, 范围为 $+\$7F \sim -\80 ; 长偏移占 2 字节, 范围为 $+\$7FFF \sim -\8000 。在机器的指令码中分别以字节(或字)的补码形式跟在操作码字节(或字)之后。

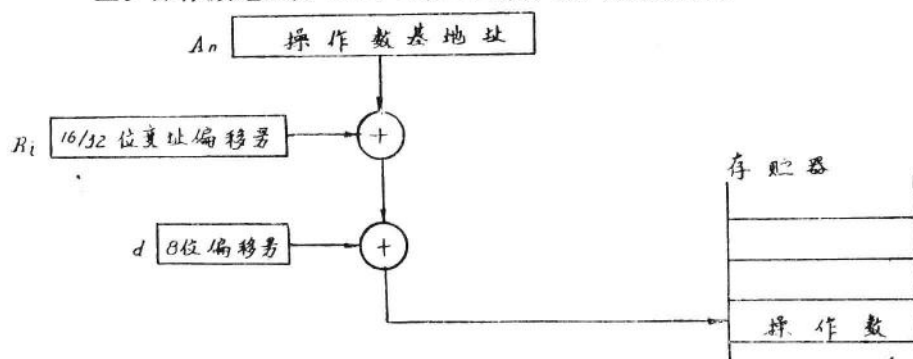


7. 带变址器地址寄存器间接寻址 (Mode = 110, Register = n)

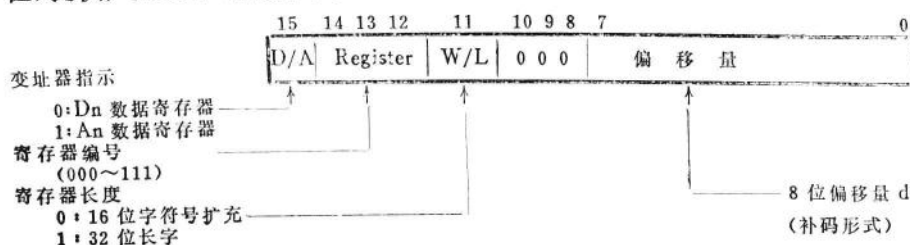
汇编符号: $d(A_n, R_i)$

$d(An, Ri, L)$

说明：由 An 提供基地址， d 提供 8 位偏移量， Ri 提供 16 位(或 32 位)变址偏移量。操作数地址为 An 、 d 和 Ri 的符号扩充相加和。



在机器指令代码中，偏移量 d 和变址偏移量 Ri 由操作码字以后扩充的 2 个字节(16 位)提供，其具体结构如下：



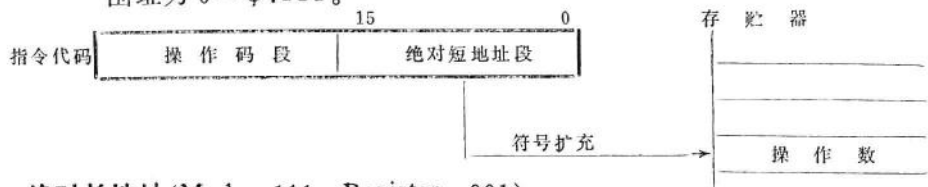
8. 绝对短地址 (Mode = 111, Register = 000)

汇编符号: XXX (采用 10 进制表示地址)

$\$XXX$ (采用 16 进制表示地址)

说明：地址包含在指令中。将指令中 16 位绝对短地址符号扩充便得到操作数的有效地址。

在机器的指令码中，绝对短地址占 16 位(2 字节)，符号扩充得到的有效地址范围为 $0 \sim \$7FFF$ 。

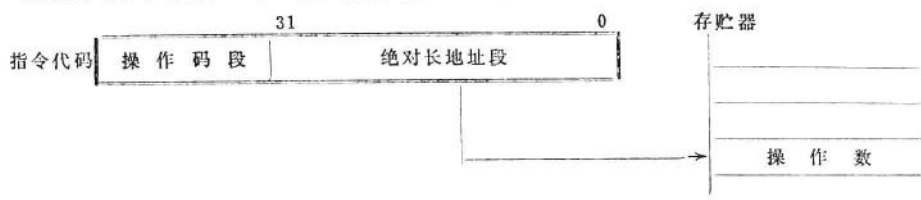


9. 绝对长地址 (Mode = 111, Register = 001)

汇编符号: XXX (采用 10 进制表示地址)

$\$XXX$ (采用 16 进制表示地址)

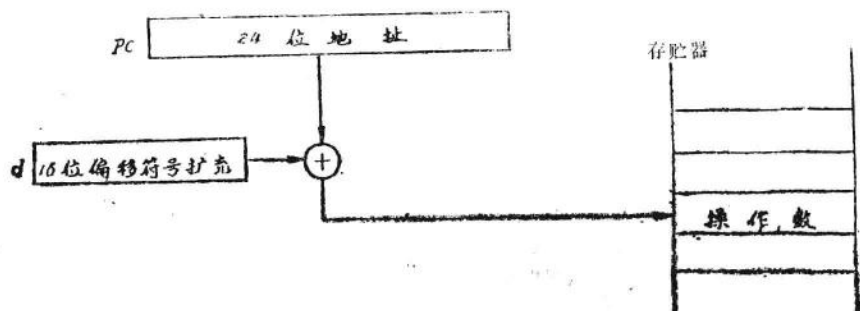
说明：指令的后 32 位即为操作数的绝对地址



10. 带偏移量指令计数器相对寻址 (Mode = 111, Register = 010)

汇编符号: d(PC)

说明: 本指令需要一个字的扩充段, 以存放 16 位偏移量。寻址时, 将当前指令计数器的 PC 值与符号扩充的 16 位偏移量相加便得到操作数的有效地址。

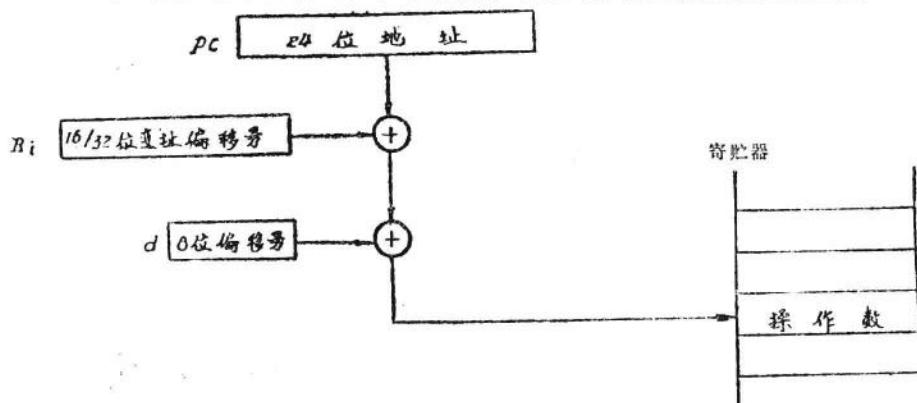


11. 带变址器指令计数器相对寻址 (Mode = 110, Register = 011)

汇编符号: d(PC, Ri)

d(PC, Ri, L)

说明: 寻址时, 将 PC 提供的地址与 d 提供的 8 位偏移量、Ri 提供的 16 位 (或 32 位) 变址偏移量符号扩充后相加, 从而得到操作数的有效地址。



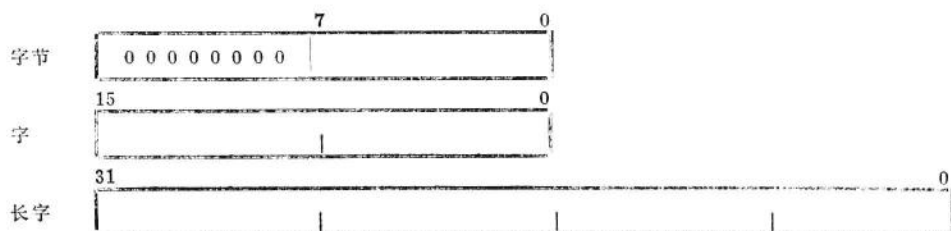
在机器指令代码中, 偏移量 d 和变址偏移量 Ri 由操作码字段后扩充的 2 个字节 (16 位) 提供, 具体结构与 7 (Mode = 110) 相同。

12. 立即数 (Mode = 111, Register = 100)

汇编符号: #XXX (10 进制数)

\$XXX (16 进制数)

说明: 立即数需要在指令代码的操作码段后面扩充 2 至 4 个字节, 以存放立即数字节、字或长字。当采用字节时, 仍占用 16 位, 但高字节为 0。



13. 立即快数

汇编符号: #XXX

说明: 立即快数是立即数的一种压缩表达形式, 仅能用于 ADDQ, SUBQ 和 MOVEQ 三条指令。这些指令是利用指令操作码中的 3 位来表示一个取值为 1~8 的字节、字或长字(对 ADDQ 和 SUBQ), 或是利用操作码中的后 8 位表一个取值为 -128~127 的扩充长字(对 MOVEQ)。这些指令没有扩充字部分, 都是双字节指令。因此, 其指令代码的长度将比采用立即数寻址方式缩短了 2 或 4 个字节。显然这是一种节省存贮空间的有效手段。

14. 隐含寻址

隐含寻址指令中并不出现寻址寄存器、地址偏移量等信息, 而由指令操作码来决定如何寻址。隐含寻址这类指令共 21 种, 详见表 1.1。

表 1.1 隐含寻址指令

指 令	所 指 的 寄 存 器
条件相对转移(Bcc), 无条件相对转移(BRA)	PC
相对转子(BSR)	PC, SP
检查寄存器边界(CHK)	SSP, SR
检查条件, 减量并转移(DBcc)	PC
有符号除法(DIVS)	SSP, SR
无符号除法(DIVU)	SSP, SR
无条件转移(JMP)	PC
转子(JSR)	PC, SP
连接并定位(LINK)	SP
条件码传送(MOVE CCR)	SR
状态寄存器传送(MOVE SR)	SR
用户栈指针传送(MOVE USP)	USP
有效地址入栈(PEA)	SP
异常处理返回(RTE)	PC, SP, SR
返回并恢复条件码(RTR)	PC, SP, SR
子程序返回(RTS)	PC, SP
陷阱(TRAP)	SSP, SR
溢出陷阱(TRAPV)	SSP, SR
断开连接(UNLK)	SP
逻辑立即数与 CCR 操作	SR
逻辑立即数与 SR 操作	SR

作为小结, 现将 MC68000 的 14 种寻址方式列表于表 1.2

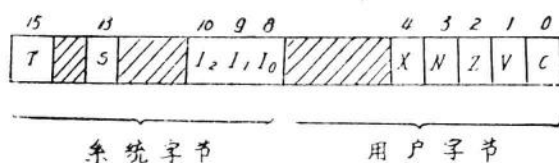
表 1.2 MC68000 寻址方式一览

Mode	Register	寻 址 方 式	汇编表示法	寻 址 对 象
0 0 0	寄存器编号	数据寄存器直接寻址	Dn	操作数在寄存器中
0 0 1	寄存器编号	地址寄存器直接寻址	An	
0 1 0	地址寄存器编号	地址寄存器间接寻址	(An)	操作数在存贮器中
0 1 1	地址寄存器编号	后增量地址寄存器间接寻址	(An) +	
1 0 0	地址寄存器编号	预减量地址寄存器间接寻址	-(An)	

Mode	Register	寻 址 方 式	汇编表示法	寻 址 对 象
101	地址寄存器编号	带偏移地址寄存器间接寻址	$d(An)$	
110	地址寄存器编号	变址、偏移地址寄存器间接寻址	$d(An, Ri)$	
111	000	绝对短地址	$\$XXX.S$	
111	001	绝对长地址	$\$XXX.L$	
111	010	带偏移指令计数器相对寻址	$d(PC)$	
111	011	变址、偏移指令计数器相对寻址	$d(PC, Ri)$	
111	100	立即数	$\bullet XXX$	操作数在 指令代码中
—	—	立即快数	$\bullet XXX$	
—	—	隐含寻址		

§1-3 状态寄存器和优先状态

MC68000 有一个 16 位的状态寄存器 SR, 其中高 8 位为系统字节, 低 8 位为用户字节, 又称为条件码 CCR。



用户字节也只使用了 5 位, 它们的意义是:

进位位(C): 当在执行指令时(如加、减法运算等)超过最高位而引起进位, 则标志位置位; 否则复位。

溢出位(V): 在进行算术运算时, 如果产生溢出, 则置位; 否则复位。

零位(Z): 若操作结果是 0, 则标志位置位; 否则复位。

符号位(N): 若操作结果的最高位是 1, 则标志位置位; 否则复位。

扩展位(X): 该位通常与进位标志位(C)有相同的作用, 但在移位指令和循环指令等指令中具有独特的作用。该标志位在多倍字长精度运算中也有效。详见指令说明部分。

对于具体的指令执行过程, C、V、Z、N 和 X 标志有不同变化情况。现将各种有关指令对条件码的影响列于表 1.3。读者阅此表时, 可参阅附录中每一条指令的具体说明。

表 1.3 条件码变化情况

指 令	X	N	Z	V	C	特 殊 定 义
ABCD	•	U	?	U	?	$C = \text{十进制进位}$ $Z = Z \cdot \bar{R}_m \cdot \dots \cdot \bar{R}_0$
ADD.I ADD ADDQ	•	•	•	?	?	$V = S_m \cdot D_m \cdot \bar{R}_m + \bar{S}_m \cdot \bar{D}_m \cdot R_m$ $C = S_m \cdot D_m + R_m \cdot \bar{D}_m + \bar{S}_m \cdot R_m$
ADDX	•	•	?	?	?	$V = S_m \cdot D_m \cdot \bar{R}_m + \bar{S}_m \cdot \bar{D}_m \cdot R_m$ $C = S_m \cdot D_m + R_m \cdot \bar{D}_m + \bar{S}_m \cdot R_m$ $Z = Z \cdot \bar{R}_m \cdot \dots \cdot \bar{R}_0$