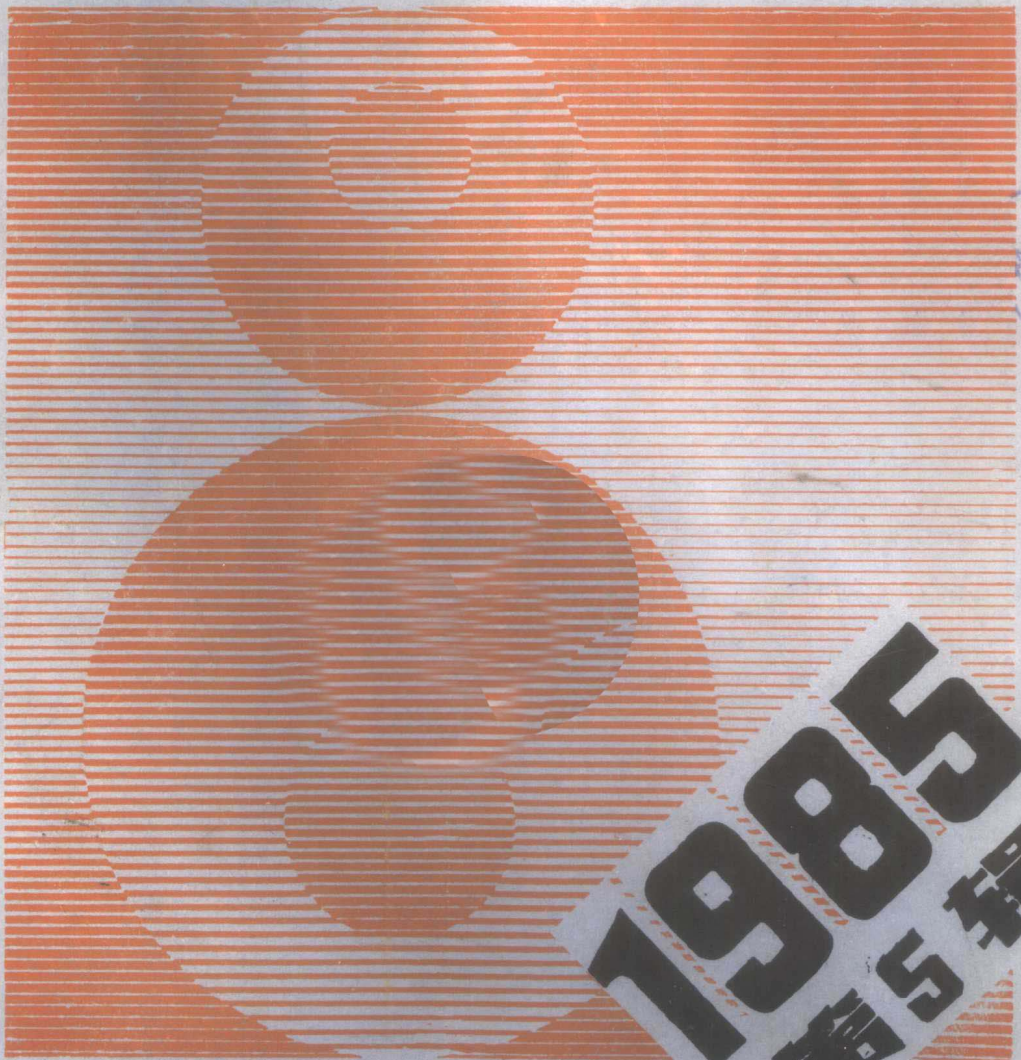


770020

6087
23034; 2
1985
T. 8

WEIXING DIANNAO YINGYONG

微型电脑应用



1985
第5辑

上海交通大学出版社

6087
23034; 2
1985
5

主 编 张钟俊
常务主编 白英彩
责任编辑 叶安琪
封面设计 朱天明

微 型 电 脑 应 用

(1985 年 第 5 辑)

上海交通大学出版社出版

(淮海中路 1984 弄 19 号)

新华书店上海发行所发行

常州村前印刷厂排印

开本 787×1092 毫米 1/16 印张 8.5 字数 211000

1985 年 11 月第 1 版 1985 年 11 月第 1 次印刷

印数: 1—15000 册

统一书号: 15324·40 科技书目: 101-250

定价: 1.70 元

目 录

1985 年第 5 辑 (总第 8 辑)

专 题 论 文

- 32位微处理机MC68020的结构和性能.....林匡定(1)
从单处理机多用户系统到多处理机多用户系统.....杨克忠(22)

微 机 应 用

- 多微机系统在黄浦江隧道交通监控中的应用.....王本初 朱梦熊等(29)
机械工厂生产管理信息系统.....罗希恒(35)
全自动单纱强力试验的微机控制及处理系统.....吕承柏 张国生等(41)
随机文件的应用.....潘振显(47)

操作系统专题

- 微机操作系统的种种问题.....[美] Harrey J. Hindin(53)
操作系统扩展连接不同的系统.....[美] John Row等(64)
微机操作系统的发展方向.....[美] Vincent Alia Gary Gysin(72)
实时工作的组合式操作系统.....[美] Gary Funck(80)
微机操作系统的模块化方法.....[美] John Little(90)
目标模型简化与系统设计加速.....[美] Gary L. Passon(98)

Modula-2程序设计(五)..... [瑞士] N. 沃斯(106)

字处理器和Wordstar.....方存正(115)

多 机 与 局 网

- 基于IDC网络的一个电子邮递系统EMP.....叶文川 陆春妮 孙树华(120)
局网(七): 局部网络的表示层.....吴竹辉 汪同英(126)

32位微处理机MC68020的结构和性能

上海第二医学院 林匡定

一、概 述

把16位微处理机扩充成32位机，MC68020是成功的代表。它不仅具有完全32位的数据和地址总线，而且具有32位的内部体系结构。作为M68000体系结构的32位实现，它代表了高性能微处理机设计的工艺水平。它可执行为M68000处理机系列各机种编制的用户目标代码，而且速度更快。这种新处理机除了高性能和兼容性以外，还有不少系统方面的提高。它支持一些新增加的寻址方式；含有一些新的指令；将所有的指令扩充成32位操作；具有更大的一组寄存器并且支持更多的数据类型。MC68020支持MC68010的虚拟存贮/虚拟机器功能，扩展了MC68010对模块式程序设计的支持，并为指令系统的扩充提供了一个协处理机接口。

尽管MC68000系列中的所有机种都已有32位的用户寄存器、操作数和内部寄存器，然而MC68020增加了完全32位的(内部和外部)数据通路、两条32位内部地址通路，一个32位的执行单元、3个32位的运算单元和一个片内指令高速缓冲存储器。设计的目标是使16MHz的MC68020的工作性能比8MHz的MC68000提高6到8倍。这一性能提高是通过MC68020工作的高时钟频率、32位数据总线宽度、片内指令高速缓存的使用和寻址方式的改进等几个方面来实现的。

MC68020的结构框图示于图1。各主要模块以高度独立的方式工作，因此能最大限度地实现并行操作。指令预取控制器将指令从数据总线上取来装入译码单元和片内高速缓存。时序控制单元负责全面的芯片控制，管理内部总线、寄存器和执行单元的各种功能。

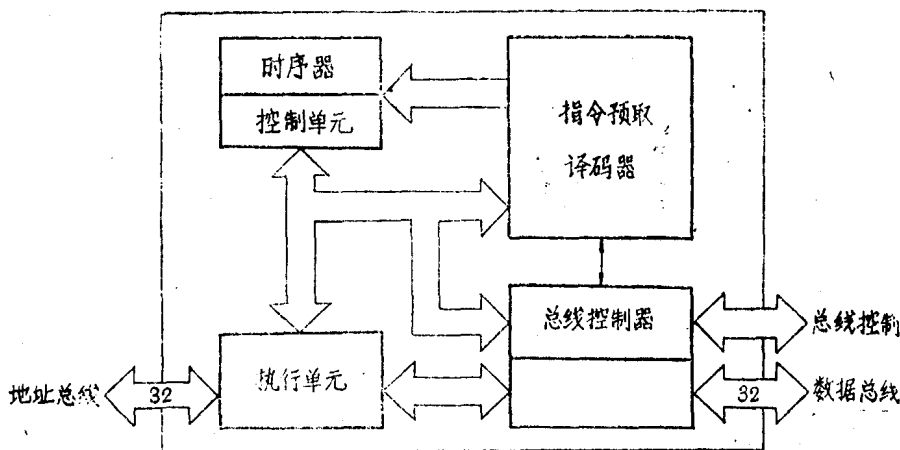


图1 MC68020 结构框图

MC68020的编程模式示于图2和图3，其中用户编程模式和M68000系列其它处理机相同，管理编程模式则有所扩充。从图2中可以看出MC68020共有16个32位的通用寄存器、一个32位的程序计数器、一个16位状态寄存器、一个32位向量基址寄存器、两个3位的替换功能码寄存器和两个32位的高速缓存操作(地址和控制)寄存器。

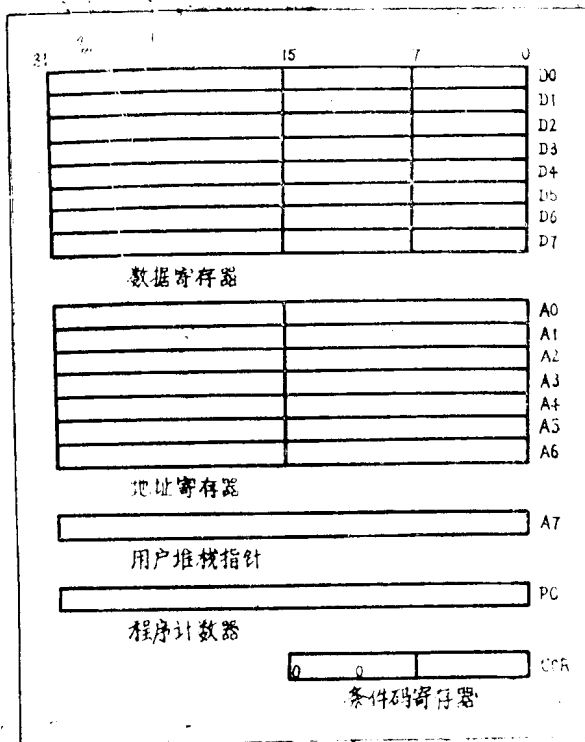


图2 用户编程模式

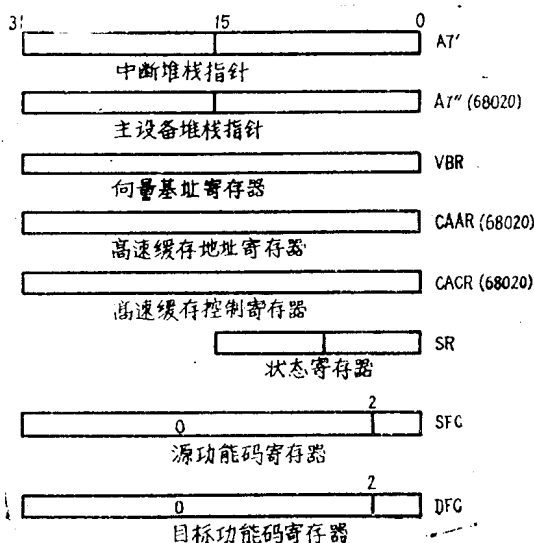


图3 管理编程模式

寄存器D0~D7用作数据寄存器，可按位、位字段（1~32位）、字节、字、长字（16位）和4倍长字（64位）操作。寄存器A0~A6和用户、中断及主设备堆栈指针都是地址寄存器，可用作软件堆栈指针或基址寄存器。地址寄存器可用于字或长字两种操作，所有16个寄存器（D0~D7和A0~A7）都可用作变址寄存器。向量基址寄存器用来确定异常处理向量表的内存地址，从而可支持多重向量表。替换功能码寄存器允许管理程序访问任何地址空间。高速缓存控制寄存器CACR用来控制高速缓存和存取高速缓存的状态。高速缓存地址寄存器CAAR用作高速缓存的指针，使得这些控制操作只影响高速缓存中的一项。

MC68020状态寄存器格式如图4所示。其中有两位是新定义的位，T0位与原先的跟踪方式位T（现为T1位），结合在一起，可定义三种跟踪状态：不跟踪、流向改变时跟踪和指令执行时跟踪。增加的流向改变时跟踪方式允许跟踪仅在执行了一条使程序计数器不按顺序递增的指令（也就是转移、JMP或RTS）之后才发生。增加的另外一位是主设备状态位M，用来指明何时处理机处于主设备状态。该位和管理状态位S结合使用，可指定一个寄存器作为系统堆栈指针，这一增加的堆栈（如果M位置位）仅用于异常处理有关的堆栈操作。然而兼容性仍得以维持，因为复位时M位被清除。M位清除时的表现与M68000系列中的其它处理机相同。

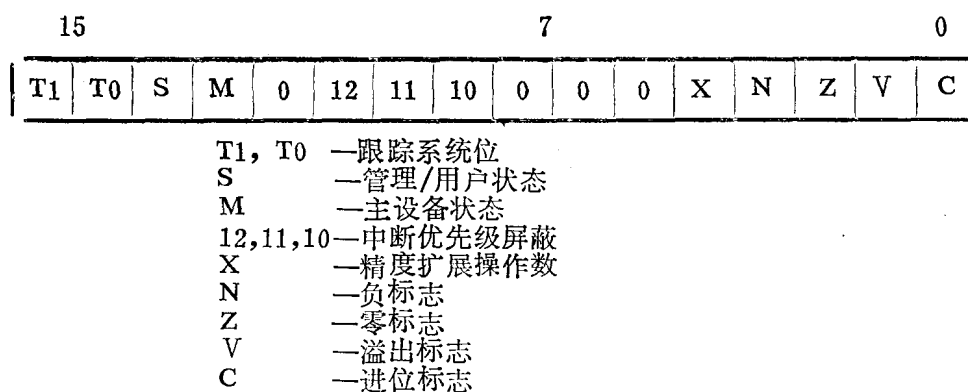


图 4 MC68020状态寄存器

M68000 的地址空间分割成五个,即用户数据、用户程序、管理数据、管理程序和 CPU 空间。在MC68020中,这种分割方法已作了重新定义和有了扩展,如表 1 所示。专门定义了一个用于用户应用程序的用户保留空间。这一空间只能由处理机空间传送指令 MOVES 来访问。

表 1 MC68020地址空间与功能码

FC2—FC0	空 间	FC2—FC0	空 间
000	未用(保留)	100	未用(保留)
001	用户数据	101	管理数据
010	用户程序	110	管理程序
011	用户保留	111	CPU 空间

此外, CPU 空间进一步分割给 16 类访问。CPU 访问的类型由地址位 A19~A16 和 CPU 空间功能码 (FC=111)来指出。表 2 指出了定义的不同类型 CPU 访问。所有未定义空间留作今后扩展时用。对 CPU 空间中这些区域进行了定义,就有可能响应断点和中断并提供一种手段和协处理机以及其它专用设备(如存储器管理部件)进行通讯。

表 2 CPU 空间访问类型

A19—16	访 问 类 型	A19—16	访 问 类 型	A19—16	访 问 类 型
0000	断点响应	0101	保留	1010	保留
0001	存储器管理部件	0110	保留	1011	保留
0010	协处理机	0111	保留	1100	保留
0011	保留	1000	保留	1101	保留
0100	保留	1001	保留	1110	保留
				1111	中断响应

二、指令系统

MC68020指令系统包括和扩展了MC68010的指令系统,并且增加了一些新的指令、新的数据类型和新的寻址方式。

1. 数据类型

M68000系列支持位、字节、长字和二进制(BCD)编码等数据类型。可变宽度位字段和压缩BCD码、4倍长字(8个字节)以及可变字节长度(Variable-byte-length)操作数是MC68020支持的4种新数据类型。位字段指令针对长度从1位到32位的任何位字段操作,而不管其位置在寄存器中还是在存储器中。4倍长字操作数为长除和乘法指令所需,它们分别处理64位的被除数和乘积。可变字节长度操作数用来支持协处理机接口,各协处理机可以定义适合于特定应用的操作数长度。

2. 寻址方式

M68000支持14种寻址方式,它们是寄存器直接、寄存器间接(可带前减量、后增量、位移和变址)、绝对、立即数、程序计数器相对和隐含有效地址等寻址方式。MC68020对变址寻址方式作了扩充,增加了存储器间接(memory indirection)、完全32位的位移值、位移值和变址操作数的符号扩展、变址值比例换算(index scaling)、前变址和后变址以及基址寄存器和/或变址值项的禁止(suppression),能够进一步地支持数组和表列操作。这些扩充对数据和程序地址空间的访问都适用。表3给出MC68020的全部18种寻址方式。

表 3

MC 68020 寻址方式

寻址方式	语 法
寄存器直接	
数据寄存器直接	Dn
地址寄存器直接	An
寄存器间接	
地址寄存器间接	(An)
带有后增量的地址寄存器间接	(An) +
带有前减量的地址寄存器间接	- (An)
带有位移的地址寄存器间接	(d16, An)
带有变址的寄存器间接	
带有变址(8位位移)的地址寄存器间接	(d8, An, Xn)
带有变址(基址位移)的地址寄存器间接	(bd, An, Xn)

续表

寻址方式	语 法
存储器间接	
后变址存储器间接	$([bd, An], Xn, od)$
前变址存储器间接	$([bd, An, Xn], od)$
带位移的程序计数器间接	$(d16, PC)$
带变址的程序计数器间接	
带变址(8位位移)的PC间接	$(d8, PC, Xn)$
带变址(基址位移)的PC间接	(bd, PC, Xn)
程序计数器存储器间接	
PC存储器后变址间接	$([bd, PC], Xn, od)$
PC存储器前变址间接	$([bd, PC, Xn], od)$
绝对	
绝对短	$\times \times \times .W$
绝对长	$\times \times \times .L$
立即	$\# \langle data \rangle$

说明

Dn = 数据寄存器, $D0 \sim D7$;

An = 地址寄存器, $A0 \sim A7$;

$d8, d16$ = 一个2的补码或带符号扩展的位移量, 作为有效地址计算的一部分加上; 长度为8或16位($d16$ 和 $d8$ 是16和8位位移值); 当省略此值时, 汇编程序用零值。

Xn = 用作为变址寄存器的地址或数据寄存器; 格式为 $Xn.sz*scl$, 其中 sz 为 W 或 L (指出变址寄存器长度), scl 为1、2、4或8(变址寄存器被 scl 乘), sz 和/或 scl 的使用可任选。

bd = 2的补码表示的基址位移量; 给出这个量时, 长度可为16或32位。

od = 外部位移量, 在任意一存储器间接之后, 作为部分有效地址加上去。其使用和16位还是32位可任选。

PC = 程序计数器。

$\langle data \rangle$ = 8、16或32位的立即值。

$()$ = 有效地址。

$[]$ = 用作为长字地址的间接地址。

这一表格并非一种完整的描述, 因为其中的每一项都可在计算时予以禁止(赋值0)。表中的变址寄存器可以是数据或地址寄存器, 其长度可取为字或长字, 还可以比例因子1、2、

4 或 8 来换算。存储器间接以方括号“[]”来表示，也就是说方括号中的地址用来取得一个长字地址，再用它来计算有效地址。在采用存储器间接方式时有两个位移值可用，当然在不需时也可被禁止。如果整个地址产生能力都不需要，则可采用一种简略格式来加快变址和换算，这时带有一个 8 或 16 位的位移值。

这些新的寻址方式很有意义，可寻址一些高级语言的数据结构。这些寻址方式不仅改进了执行性能，而且使编译程序产生的代码变得更加有效，因为它们能够更自然地表达和高级语言有关的各种结构。后面还要介绍有关这些寻址方式的一个实例。

3. 指令扩展

M68000 系列的指令系统只需作很少一些改动就可适用于 32 位操作。表 4 详细介绍了这一扩展所涉及的对 M68000 指令系统的增补，乘除指令原来就已针对 32 位操作数操作。现在可将两个 32 位操作数相乘而形成 32 位或 64 位的乘积。也允许进行 64 位除以 32 位和 32 位除以 32 位的除法操作。指令有关的位移值从 16 位扩展成 32 位。这是一种重要的能力——它使得分支和连接操作可以不受限制地在整个 4G 字节地址空间中发生。控制寄存器传送指令 (MOVEC) 也作了扩充以适应 MC68020 中的各种新的控制寄存器；断点指令也有所提高，可以更进一步支持实时程序调试。

表 4 MC68020 的指令扩展

指 令	扩 展
MULS, MULU, DIVS, DIVU	扩展成 32 位操作数
BCC, BRA, BSR, LINK	位移值扩展成 32 位
MOVEC	新的控制寄存器也可访问
BKPT	支持操作码替代

相对于 MC68000 而言，MC68020 增补了二十多条新指令来提供新的功能 (见表 5)。其中大多数用于通用处理，有的则对专门用途的处理更有意义。

表 5 MC68020 提供的新指令

指 令	功 能
BFCHG	位字段改变
BFCLR	位字段清除
BFEXTS	位字段带符号抽取
BFEXTU	位字段不带符号抽取
BFFFO	位字段寻找第一个置位
BFINS	位字段插入

续表

指 令	功 能
BFSET	位字段置位
BFTST	位字段测试
CALLM	调用模块
CAS	比较和对半交换
CAS2	比较和对半交换(双操作数)
CHK2	检查寄存器定位上下界
CMP2	比较寄存器定位上下界
cpBcc	协处理机条件转移
cpDBcc	协处理机条件测试、减量并转移
cpGEN	协处理机通用功能
cpRESTORE	协处理机恢复内部状态
cpSAVE	协处理机保存内部状态
cpSETcc	协处理机条件置位
cpTRAPcc	协处理机条件陷阱
PACK	压缩 BCD
RTM	模块返回
UNPK	非压缩BCD

大量的新指令用于提供可变长位字段的操作。利用这些指令可对一个长度可变(最高可达32位)的位字段(可位于数据寄存器中,也可以骑跨存贮器的5个字节),进行清除、置位、求反、抽提、插入、扫描或测试。这些指令在操作压缩数据和通讯及图形处理等场合都十分有效。这些指令的一个特点是,在处理存贮器中的一个位字段时,可指定任何一个1~32位的字段,而不必顾及其在存贮器中的定位。这种位字段在存贮器中的位置分配没有任何限制。如果象上面提到的那样,一个位字段骑跨了存贮器的5个字节,处理机内部能识别这种情况并作出适当的调整。

指令系统的主要增补之一是一组支持协处理机接口的通用协处理机指令。用MC68020这样的通用处理机来支持用户的各种专门用途的计算要求显然是不可能的,因此,处理各种非一般的应用必须有一种策略。为了满足这种要求,研制了协处理机接口用以提供一种机制。各种协处理机都可以通过这种机制而增加MC68020的能力,且不会增加体系结构的负担。在通用的体系结构中,高度专门化的指令和数据类型不可能直接得到完全的支持。MC68000体系结构的强大功能一部分就来源其通用性。MC68020的设计人员对这种通用性作了扩展,提供了可以用许多种方法组合的低级原语来支持各种专门应用,而不是企图对种类之多几乎没有任何限制的各种专门应用提供不完全的支持。

这些协处理机指令取代了F行异常机制,这种机制允许用户通过采取一个陷阱(一种异常)转向仿真例程来仿真这些专门功能。7条协处理机指令提供了和协处理机进行指令信息和数据通讯的一种手段。这些指令还得到一组18条响应原语的补充,后者可供协处理机用来请求有关它的处理机动作。在典型情况下,作为协处理机指令的结果,协处理机从处理机接收到一些指令信息。协处理机对需要的支持动作作出估计并通过向处理机发出原语响应来请求这些服务。

模块化程序设计得到调用模块(CALLM)和从模块返回(RTM)指令的支持。这些指令可以将用户状态进一步分成256级存取权限,然后定义一个协议来支持不同存取级之间的转换。这些指令中有两种不同的类型,分别支持不涉及存取级改变的简单模块调用和涉及存取级改变的复杂模块调用。

系列早期机种已配备的边界检查指令CHK和CMP亦已扩展成允许指定上下两条边界。利用CHK 2和CMP2指令,可以检查一个寄存器的值是否落在上下边界之间。两种指令的差别是,如果该值超出边界,CHK2除了置位条件码以外,还导致一次异常处理,而CMP2指令仅置位条件码。这种测试可针对有符号或无符号边界进行。处理机自动对两个边界之间的关系进行估计来确定适用何种类型的比较。

还有几条新指令用来提供对系统的支持,“比较和对半交换(CAS)”以及“比较和对半交换2(CAS2)”指令允许字节、字或长字数据元素的总线封锁操作来支持系统队列和堆栈功能。“压缩和解除压缩”指令PACK和UNPK方便了算术操作中从压缩的ASCII/EBCDIC数据格式到BCD格式或反方向的数据转换。“断点”指令BKPT通过断点响应机制给用户提供了系统调试帮助。采用这条指令就有可能断开一条指令。在处理机企图执行该断点指令时,就执行一个断点响应周期,取得一个适当的操作码,并在适当时代入断点处的指令管道,否则就执行断点例程。

表6以助记符字母顺序列出MC68020的全部指令。详细描述从简,可查看用户手册。

表 6

MC68020指令系统概要

助记符	说 明	助记符	说 明	助记符	说 明
ABCD	带扩展的十进制加	Bcc	条件转移	BFSET	位字段测试和置位
ADD	加	BCHG	位测试并改变	BFTST	位字段测试
ADDA	加地址	BCLR	位测试并清除	BRA	转移
ADDI	加立即数	BFCHG	位字段测试并改变	BSET	位测试和置位
ADDQ	快速加	BFCLR	位字段测试并清除	BSR	转移到子程序
ADDX	带扩展加	BFEXTS	不带符号位字段抽取	BTST	位测试
AND	逻辑与	BFEXTU	不带符号位字段抽取	CALLM	调用模块
ANDI	逻辑与立即数	BFFFO	寻找第一位字段	CAS	比较和对半交换操作数
ASL, ASR	算术左、右移	BFINS	位字段插入	CAS2	比较和对半交换双操作数

助记符	说 明	助记符	说 明	助记符	说 明
CHK	寄存器边界检查	MOVE SR	传送状态寄存器	RTS	从子程序返回
CHK2	寄存器上下界检查	MOVEUSR	传送用户栈指针	SBCD	带扩展的十进制减
CLR	清除	MOVEC	传送控制寄存器	Scc	条件置位
CMP	比较	MOVEM	多寄存器传送	STOP	停机
CMPA	比较地址	MOVEP	外设传送	SUB	减
CMPI	比较立即数	MOVEQ	快速传送	SUBA	减地址
CMPM	存储器与存储器比较	MOVES	备用地址空间传送	SUBI	减立即数
CMP2	比较寄存器上下界	MULS	带符号乘	SUBQ	快速减
DBcc	测试条件、减量和转移	MULU	不带符号乘	SUBX	带扩展的减
DIVS	带符号除法	NBCD	带扩展的十进制求补	SWAP	寄存器对半交换
DIVSL	带符号除法	NEG	求补	TAS	测试操作数和置位
DIVU	不带符号除法	NEGX	带扩展的求补	TRAP	自陷
DIULV	不带符号除法	NOP	空操作	TRAPcc	条件自陷
EOR	逻辑异或	NOT	逻辑反	TRAPV	溢出自陷
EORI	逻辑异或立即数	OR	逻辑或	TST	测试操作数
EXG	交换寄存器	ORI	逻辑或立即数	UNLK	链接解除
EXT	符号扩展	PACK	压缩BCD	UNPK	解除压缩BCD
JMP	跳转	PEA	压入有效地址		协处理机指令
JSR	跳转到子程序	RESET	复位外部设备	cpBcc	条件转移
LEA	取有效地址	ROL, ROR	左、右环移	cpDBcc	测试协处理机 条件, 减量和转移
LINK	链接定位	ROXL ROXR	带扩展的左右环移	cpGEN	协处理机通用指令
LSL, LSR	逻辑左、右移	RTD	返回和定位解除	cpREST ORE	恢复协处理机内部状态
MOVE	传送	RTE	从异常返回	cpSAVE	保存协处理机内部状态
MOVEA	传送地址	RTM	从模块返回	cpScc	条件置位
MOVE CCR	传送条件码寄存器	RTR	返回并恢复条件码	cpTRA Pcc	条件自陷

三、高速异步总线

MC68020保留了M68000系列快速、易用的非多路复用异步总线接口, 然而, 在MC68020总线接口设计中增加了一些有意义的特性。图5是MC68020的总线接口。

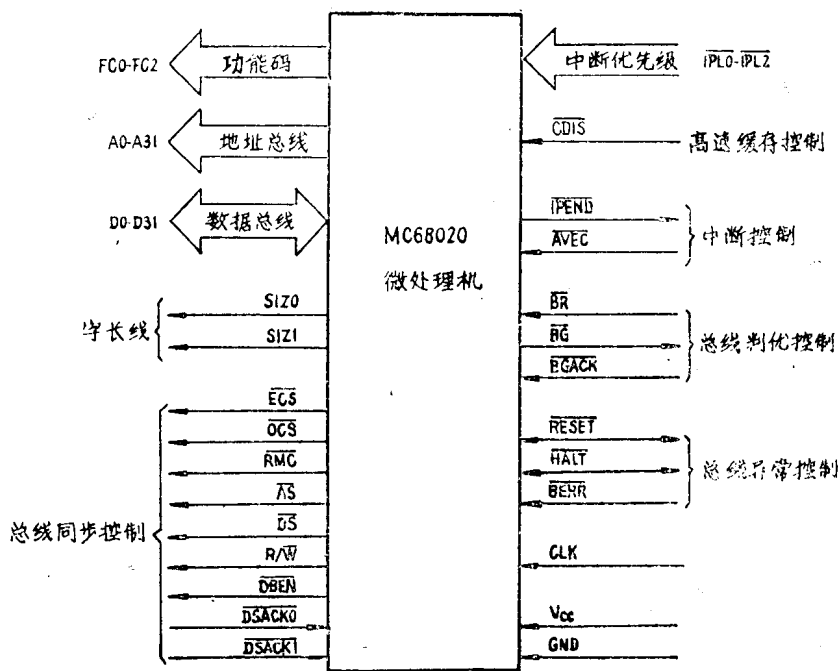


图 5 MC68020总线接口

组成这一总线接口的有完全32位的地址和数据总线、时序控制线和用于中断及判优控制等方面的一些线。32位地址和数据总线不作多路复用，以求有最大的工作性能，又能简化接口设计。这一点及其它一些改进使得这一接口可以支持16.67MHz主振频率下的180ns总线周期时限，而且地址有效到数据有效这一时限标称值仅120ns。典型的总线周期示于图6。

动态总线字长调整

MC68020总线接口支持一种称为动态总线字长调整(Dynamic bus sizing)的新特性。这一特性允许MC68020一个周期一个周期地调整总线端口的宽度，从而可以将数据分别传送给8、16和32位的总线端口，或者从这些端口取得数据。实现这一新概念的关键部分是数据多路开关、字长线输出端(SIZ0、SIZ1)和一些输入端(DSACK0、DSACK1)，MC68020就用它们和各种宽度的总线进行一个总线周期一个总线周期的动态接口。动态总线字长调整使得软件设计不必考虑存储器及I/O端口的硬件配置，并将地址空间看作任意大小的块连续，这些块可以是字节、字或长字。它除了提供一个功能强而又灵活可变的系统之外，还提供给用户一个能够在任何8或16位系统上运行，或者总线字长可随连接对象而变的高性能处理机。图7给出了采用MC68020有可能配置的各种不同端口，图8则给出采用动态字长调整来传送长字操作数给8和16位总线的典型操作数传送过程。MC68020访问存储器时，两条字长线(SIZ1、SIZ0)指出每一次传送可用的数据量(00=长字，01=字节，10=字，11=3个字节)。对于

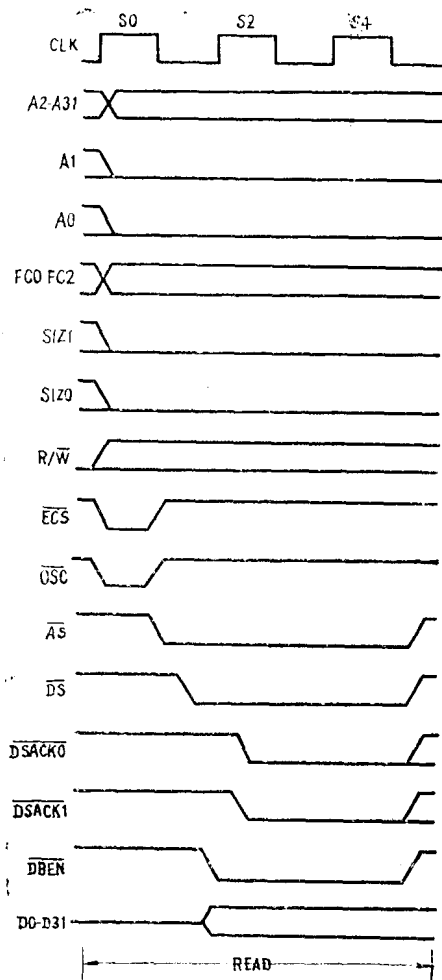


图 6 典型的MC68020总线周期

图 8 中每一例子，存贮控制都根据DSACK0和DSACK1来定义端口字长，并供处理机判断是否还要后续总线周期来完成整个数据的传送。由于MC68020能够动态调整总线字长，所以能容易地支持由 8 位、16 位和 32 位存贮器卡和 / 或 I/O 端口混合组成的系统配置。因此扩展成 32 位外部总线时不必将老的外围器件和存贮器卡丢弃。

除了提供动态字长调整特性外，MC68020还可对操作数在存贮器中的对位不加限制，就是说，即使数据操作数出现在奇地址字节边界上，处理机仍能正确地予以存贮。

图 9 给出一个非对位长字传送的实例。该例中带 $\times \times \times$ 的字节在传送过程中不受影响。在第一个周期中，地址总线的低 3 位为 001，指向奇地址字节，字长线为 00，表明要传送的操作数类型是长字。第二个周期中，地址指针指向 010，为偶地址，此时该周期中传送两个字节。第三周期地址指针虽含有偶地址，但字长线为 01，表明只剩一个字节待传送，因此仅传送一个字节。整个传送过程对程序员透明，因此程序员不必考虑数据的对位。

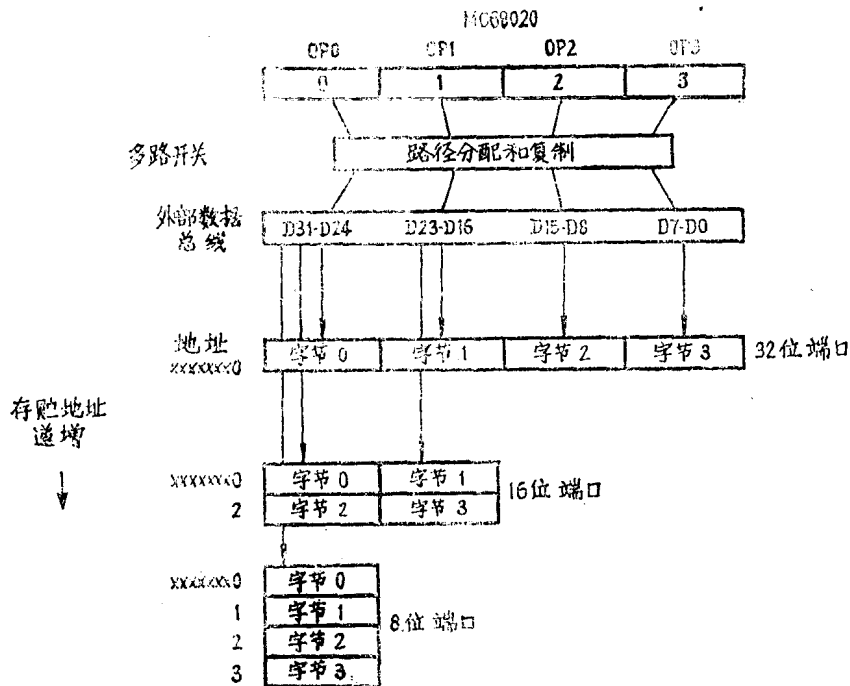
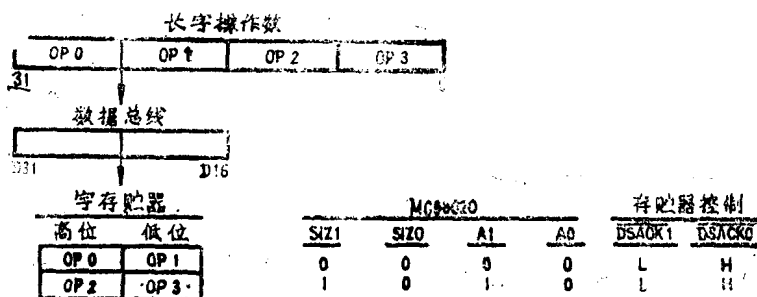
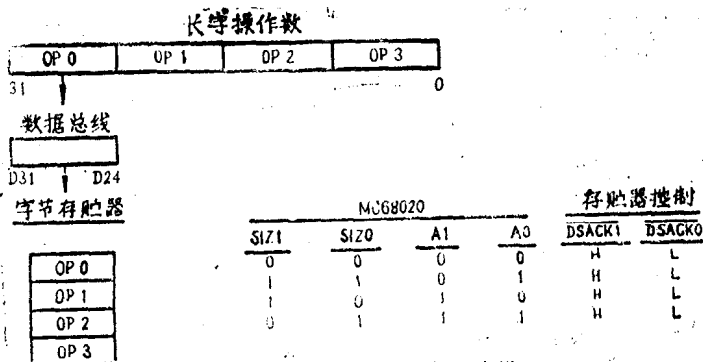


图7 MC68020数据总线与不同字长端口的接口



a 长字传送到字总线



长字传送到字节总线

图8 动态总线字长调整

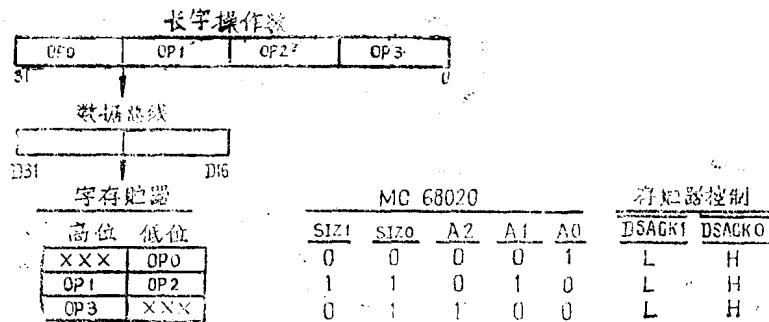


图9 非对位长字的传送

四、协处理机接口

MC68020的协处理机接口是一种能够扩展MC68020基本指令系统和支持新数据类型的有力而又灵活的机制。这种接口可以支持并发的和非并发的协处理机以及总线主设备和总线从处理机。通过提供这种功能，该接口为系统设计人员所面临的一些专门的处理问题提供了解决方法。采用这一机制，系统设计人员就可采用VLSI的协处理机来解决一些比较常见的专门用途的处理要求(如浮点计算)，或者用一种特定的板级设计来解决一些有限的但又非常重要的应用问题。该协处理机的基础是外部总线周期。CPU空间读写用来在主处理机和协处理机之间传送信息。协处理机操作由对这些协处理机控制寄存器的一系列读字操作来控制。这一接口尽管已根据MC68020环境作了优化处理，还是可以由系列中的其它较早机种来仿真，它们可以将协处理机作为一种外设来处理。最后一点，我们都知道不可能定义一个接口来解决所有的应用问题，有一些机制可用来对没有定义的原语进行软件仿真。有了这种方法，如果一种功能目前没有定义，它就可以后增补进去，或者至少可由用户来仿真。

来自于MC68020的协处理机指令(F行指令)传送至8个可能的协处理机之一来仿真。这种做法摆脱了主处理机与特定协处理机指令系统的紧密联系，使其更具一般性。对这一指令进行仿真以后，协处理机判断出需要请求主处理机为它提供哪些服务才能使它有条件执行这一指令。这些服务请求以原语的形式发送给主处理机，然后再由主处理机检查和执行这些原语。已经作出一些努力来增强这些原语的功能，从而使这一协处理机接口承担的工作量达到最低。例如，主处理机可以执行一个有效地址的计算，然后，或者将已求值的有效地址传送给协处理机，或者取得可变长度的操作数再将其传送给协处理机。通过提供这一服务，主处理机就使得协处理机的设计变得更加简单。

MC68020支持5种类型的协处理机指令：通用、转移、条件、保存状态和恢复状态(表7)。通用指令(cpGEN)用于数据传送和操作，转移指令(cpBcc)允许根据协处理机条件来转移。16位和32位两种转移位移值都可支持。条件指令(cpScc、cpTRAPcc和cpDBcc)允许检测64种协处理机条件之一，并根据这一条件来置位(cpScc)、自陷(cpTRAPcc)，或减量并转移(cpDBcc)。一个条件传送给协处理机求值，然后将结果告诉主处理机。除了通用、转移和条件这3种类型以外，保存类和恢复类都用于操作系统上下文转换支持。这两种类型指令允许在一种虚拟环境中保存和恢复协处理机状态。

协处理机指令从主处理机传送给协处理机求值。协处理机具备许多可为主处理机识别的原语,根据这些原语就能实现协处理机指令。主处理机支持18种基本原语,有的还有多种变型。这些原语可分成5种基本类型,分别涉及处理机/协处理机同步、指令流操作、异常处理、通用操作数传送和寄存器传送(表7)。

表 7

协处理机原语

处理机同步	忙于先前指令 忙于当前指令 非跟踪状态则开始处理下条指令 允许跟踪则开始处理下条指令 开始执行,条件为真/假	通用操作数传送	计算并传送有效地址 计算出有效地址并传送数据。 写入先前计算出的有效地址单元 接收地址,然后传送数据 栈顶单元数据传送
指令操作	传送操作字 从指令流中传送多个操作字	寄存器传送	传送单个CPU寄存器 传送单个CPU控制寄存器 传送多个CPU寄存器 传送多个协处理机寄存器 传送CPU状态寄存器和/或程序计数器
异常处理	S位未置位则进入特权违反异常处理 进入指令前异常处理 进入指令中异常处理 进入指令后异常处理		

处理机/协处理机同步原语允许协处理机将其状态告诉主处理机。协处理机可以用信号表示它正忙于先前的或当前的指令。协处理机忙时可以允许主处理机处理任何悬挂着的中断。它可请求主处理机将现行PC值传送给它。它可以要求主处理机根据指令跟踪是否被允许而开始处理下一条指令。最后一点,它还可指明条件测试的结果。

指令流操作原语允许协处理机发出将协处理机指令传送给它的请求。传送指令流原语允许协处理机请求主处理机将指令流中最多可达256字节的代码传送给它,并在必要时修改程序计数器的值。因此,协处理机指令的长度可达任意多的字数。

异常处理原语迫使主处理机检查是否有异常发生,和/或开始进入异常处理。协处理机可以请求主处理机在执行特权协处理机指令之前先作一次管态校核。如果主处理机未处于管态就进入特权违反异常处理。如果协处理机在执行一条协处理机指令之前、之中或之后检测到一次异常,它可迫使主处理机处理这一异常并保存足够的状态信息供以后从异常处理返回时用。

通用操作数传送原语用于请求主处理机和协处理机之间的操作数传送。这些原语分别用于为协处理机计算有效地址、将计算得到的有效地址传送给协处理机、有效地址单元间的数据传送以及将数据传给原先已计算出的有效地址单元。此外,协处理机还可将一个地址传送给主处理机并请求将操作数传送给这一地址单元,或从这一地址单元读取操作数。还有一种原语允许将操作数送到现行系统栈顶或从中取得操作数。

寄存器操作数传送原语允许在主处理机和协处理机之间传送单个主处理机地址/数据寄存器、单个主处理机控制寄存器、多个主处理机地址/数据寄存器和多个协处理机寄存器。此