

11
5 064
25752

646879

AMZ 8000

微處理機手冊

《AMZ8001/2》微處理機指令系統操作原理



成都科技大学图书馆
基本藏书

制导与控制 编辑组

52
209

译 者 的 话

本《参考手册》分为二大部份。第一部份为AMZ8001/2微处理机及指令系统介绍,第二部份为AMZ8001/2指令系统详解。

第一部份对AMZ8001/2CPU的内部寄存器结构,寻址方式作了介绍,并对指令系统详解中有关的名词和术语作了说明,可帮助读者顺利地阅读指令系统详解。第二部份详细介绍了处理机的指令系统。为了便于查阅,一条指令安排一页,每条指令都绘制有指令图解,以帮助读者对指令的理解,并加强记忆。

本资料对从事微处理机设计和应用的工程技术人员,以及使用微计算机的研究人员均有参考价值,对从事AMZ8001/2编程人员更是必不可少的工具书。

我们在翻译过程中得到了张成铭、治竹君等同志的协助,在此表示感谢。

由于我们水平所限,错误、缺点在所难免,恳切希望广大Z8000的用户及读者提出批评。

刘尚勤、虞秀玉译

刘尚勤校 一九八二年一月

先进的微处理机AMZ8000系列参考手册

——AMZ8001/2处理机指令系统的操作原理

前 言

目前MOS大规模集成半导体技术已发展到可将功能很强的，很复杂的通用处理机制作在一个硅片上。这预示将开创一个以廉价工具，来解决复杂问题的系统设计的新时代。这样可以不花很高的费用就可获得很强的计算能力，同时也可以提高程序设计人员在越来越复杂任务中的工作效率。

Amz8000系列是首先开发这个新时代的微处理机系列。它打破了过去由于制造工艺限制而折衷考虑系统性能的传统习惯。Amz8000系列中的Z8001/2集中了在小型计算机和大型计算机基础上发展起来的许多优良特性。这给应用程序人员，系统程序设计人员，系统设计人员提供了为解决今天复杂问题所需要的功能和灵活性。

本资料详细介绍了微处理机的指令系统。为了完整性以及便于查阅，本手册将一条指令安排一页。这种说明方法对于硬件设计者及程序设计者都很适用。本文对整个微处理机指令系统作了全面的、详尽的、易于理解的说明。

本文提供的资料，以后若有修改，将加在以后发行的Amz8000系列参考手册中有关指令系统一章中。

本资料是Amz8000系统几种基本文件之一。

目 录

微处理机结构

前 言	3
寄存器结构	3
堆栈指示器	7
程序计数器	7
处理机状态信息	7
新程序状态区指示器	9
刷新计数器	9
寻址方式	10
操作数寻址	19
存贮器寻址	19
中断与陷阱	24
指令格式	24
输入/输出	27
条件码	27
指令系统	27
按照助记符排列的指令表	223

微 处 理 机 结 构

前言

Amz8001和Amz8002是Amz8000 16位微处理机系列的最初产品。它们在软件上是相互兼容的。除加特殊说明外,本文件中所有内容对二者都适用。Amz 8001内存地址为23位,因此寻址能力为8兆字节(8,388,608字节),Amz8001系统的内存由128段组成,每段为64K字节(65,536字节),Amz8001也称之为分段型,而Amz8002仅具有16位(64K字节)的寻址能力,称之为非分段型。

寄存器结构

CPU的中心是16个16位通用寄存器,其分别称之为R0~R15。通常由指令中四位二进制数确定所希望的寄存器。指令的操作数一般可为字节(8位),字(16位)或长字(32位)。对于字节运算来说,头8个通用寄存器(R0至R7)可视为16个8位寄存器。它们分别表示为RL0,RH0,RL1等等直到RL7和RH7。用指令中的四位可确定所希望的字节寄存器。对于长字运算而言,两个16位寄存器分为一组,例如R0和R1这对寄存器定义为RR0,而R2,R3对寄存器称之为RR2,依此类推,直到R14,R15,对定义为RR14。因此,通用寄存器又可认为由8个32位寄存器组成。指令中,确定寄存器的4位中的最高3位可用来确定所需要的寄存器对,而第4位应填为零。对于某些64位操作数,通用寄存器也可4个为一组,例如,R0,R1,R2,R3定义为RQ0,R4,R5,R6,R7定义为RQ4依此类推,直到R12,R13,R14,R15定义为RQ12。指令中确定寄存器的4位中的最高2位用来确定所需要的四个寄存器组,其余二位则应为零。图1为Amz8001寄存器结构,图2表示Amz8002寄存器结构,表1为字节方式下,寄存器寻址的小结,表2为16位,32位,64位方式下,寄存器寻址的小结。寄存器可以存放操作数或地址信息。当用寄存器对存放长字操作数时,寄存器对中的偶数寄存器存放32位数的高16位数,而奇数寄存器存放低16位数。对于64位操作数,则要求4个寄存器,第一个寄存器存放最高的16位数,而这一组中最后一个寄存器存放最低的16位数。例如RQ0组寄存器中,RQ为第一个寄存器,而R3为最后一个寄存器,R4为RQ4组寄存器中第一个寄存器,R7则为最后一个寄存器,依次类推。

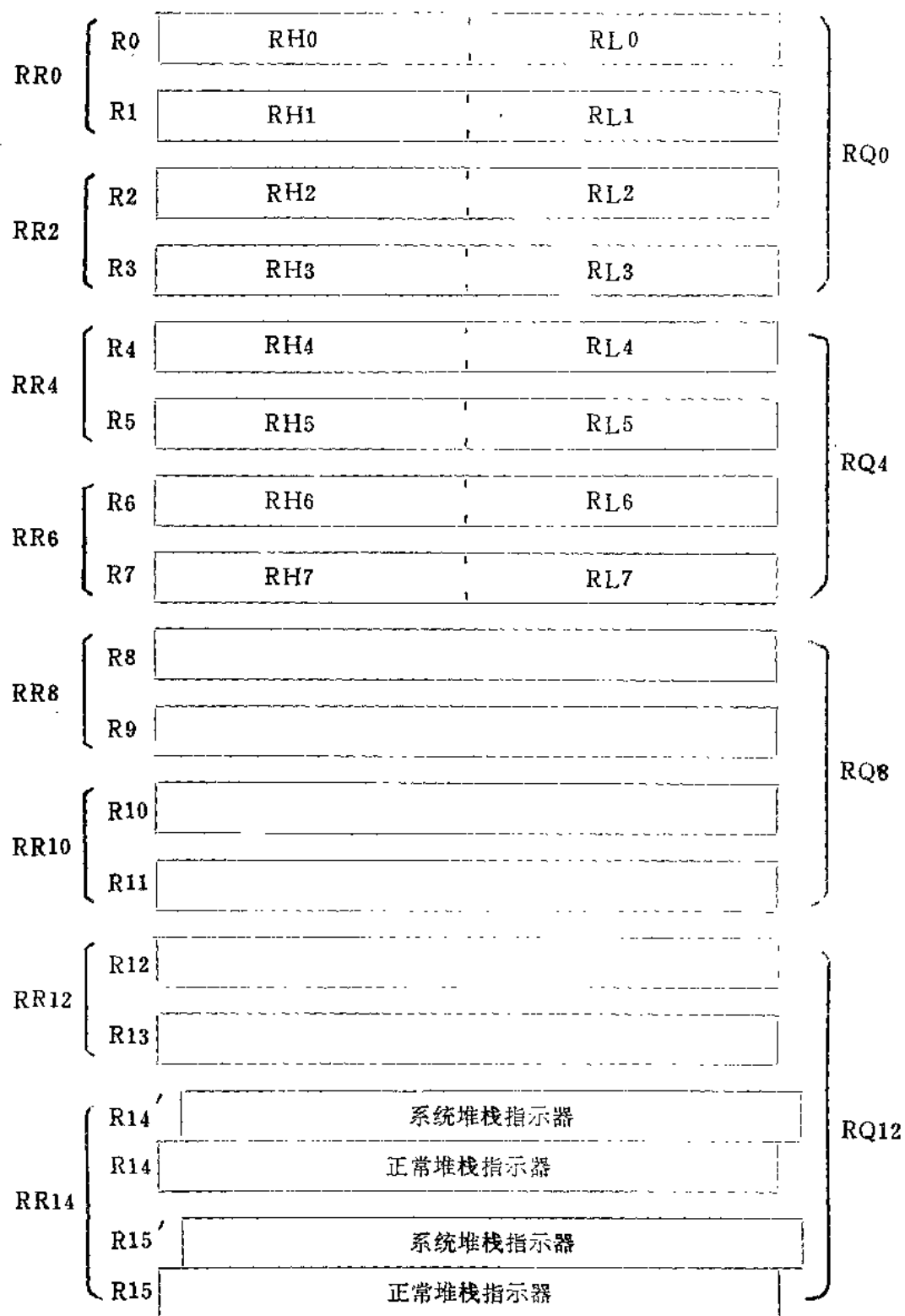


图 1 AMZ8001寄存器

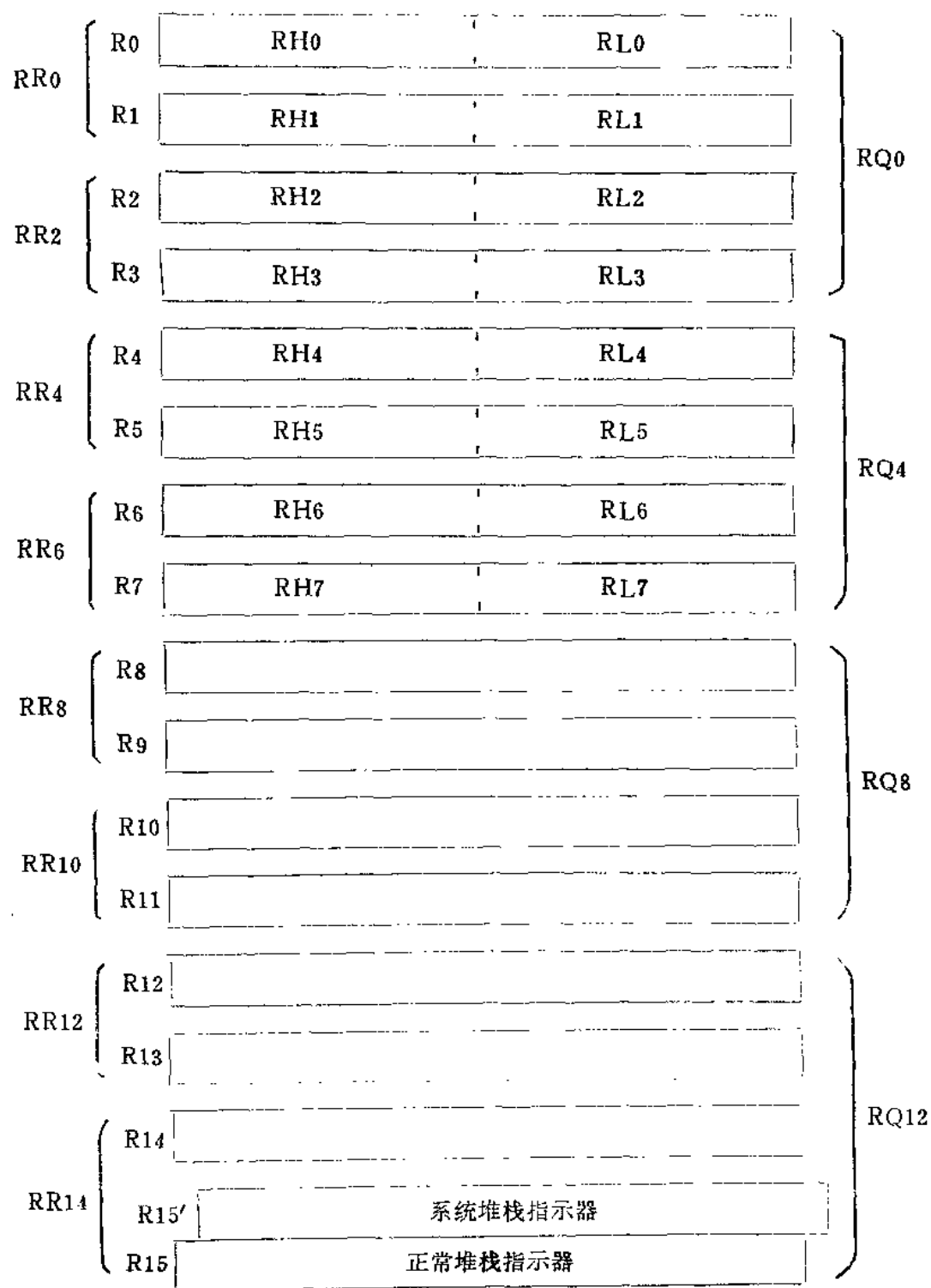


图 2 AMZ8002寄存器

指定位段				8 位 方式
0	0	0	0	RH0
0	0	0	1	RH1
0	0	1	0	RH2
0	0	1	1	RH3
0	1	0	0	RH4
0	1	0	1	RH5
0	1	1	0	RH6
0	1	1	1	RH7
1	0	0	0	RL0
1	0	0	1	RL1
1	0	1	0	RL2
1	0	1	1	RL3
1	1	0	0	RL4
1	1	0	1	RL5
1	1	1	0	RL6
1	1	1	1	RL7

表1 字节方式的寄存器寻址

指 定 位 段				16 位 方 式	32 位 方 式	64 位 方 式
0	0	0	0	R0	RR0	RQ0
0	0	0	1	R1		
0	0	1	0	R2	RR2	
0	0	1	1	R3		
0	1	0	0	R4	RR4	RQ4
0	1	0	1	R5		
0	1	1	0	R6	RR6	
0	1	1	1	R7		
1	0	0	0	R8	RR8	RQ8
1	0	0	1	R9		
1	0	1	0	R10	RR10	
1	0	1	1	R11		
1	1	0	0	R12	RR12	RQ12
1	1	0	1	R13		
1	1	1	0	R14	RR14	
1	1	1	1	R15		

表2 寄存器寻址

在Amz8001中,为了确定所要求的23位地址,需要用一寄存器对。其7位段号存放在寄存器对中编号为偶数的寄存器中,而16位偏移量则存放在寄存器对中编号为奇数的寄存器中。

堆栈指示器

Amz8000系列的结构体系允许在存储器中设立堆栈并能保持堆栈区。使用PUSH和POP指令时,任何通用寄存器(除Amz8001中的RR0寄存器对及Amz8002中的R0寄存器外)都可作为堆栈指示器使用。但是,对于CALL和RETURN指令则隐指特定的寄存器为堆栈指示器。

Amz8001中RR14通用寄存器对是隐含的栈指针,7位段号存放在R14中,16位偏移量存放在R15中,7位段号与偏移量一起构成23位分段式地址。Amz8002中,通用寄存器R15是隐含堆栈指示器,存放需要的16位地址。应注意,隐含堆栈指示器仍为通用寄存器,换句话说,某些隐含的通用寄存器,它们除具有通用寄存器的特征外,还可作为堆栈指示器用。

微处理器有二种可供选择的方式:系统方式和正常方式。系统方式有时称之为管理或特殊方式。正常方式有时又称为问题或非特殊方式。为了简化复杂系统的设计,可以将系统堆栈指示器与正常堆栈指示器分开。也就是说,除了正常堆栈指示器以外,还可以提供系统堆栈指示器。

Amz8001提供了与R14和R15相对应的两个附加寄存器,分别为R14'和R15'。当Amz8001工作在系统方式时,每当指定到R14'时,寄存器R14就代替它作为通用寄存器。同样地在系统工作方式下,Amz8001和Amz8002的R15'寄存器可以代替R15寄存器作为通用寄存器。因此,寄存器对R14',R15'(即为RR14')为Amz8001的隐含系统堆栈指示器,而R15'为Amz8002的隐含系统堆栈指示器。虽然在系统方式下不用R14和R15,但仍需给出相应的指令,以致于在不改变工作方式的情况下,就可以对这两个寄存器进行存取。在程序中断期间可利用系统堆栈指示器存放中断前的状态,而不受所选择的工作方式限制。

程序计数器

从存储器中取的指令可以控制CPU的操作方式。取指令的地址存放在程序计数器(PC)中。图3为Amz8001的程序计数器,它由两个字组成。头一个字的7位用来确定段号,第二个字为16位偏移量。段号可指定128个段中的任意一个段号,而16位偏移量可确定所指定的那一段中的存储单元。指令总是顺序排列的。当从存储器单元中取出指令时,PC自动加2。请注意,偏移量的增加不能影响段号,换句话说,偏移量的进位不能进到PC的段号寄存器中,即偏移量计数器只能进行加而不产生进位。图4表示Amz8002的PC格式,它由16位组成。除没有段号外,Amz8002的PC操作与Amz8001一样。复位时,存储器地址4的内容自动存入Amz8001PC的段号,而存储器地址6的内容自动存入PC的偏移量中。存储器地址2的内容自动存入Amz8002PC中。所有这些地址皆为零段内的单元。

处理机的状态信息

程序计数器的内容及标志,控制字(FCW)统称为处理器的状态信息。AMZ8001的

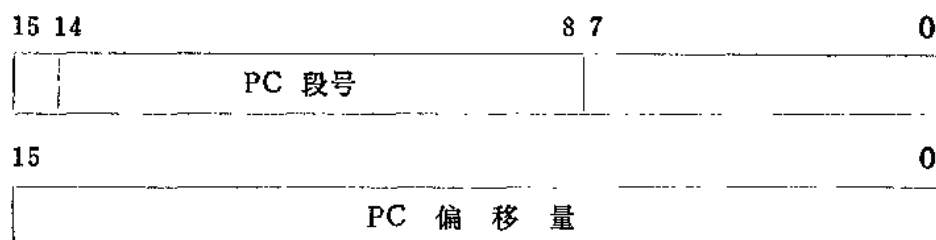


图3 AMZ8001程序计数器

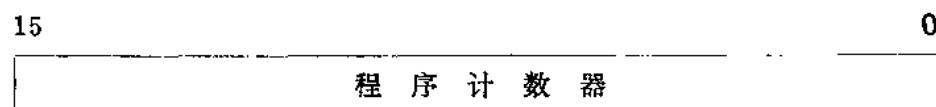


图4 AMZ8002程序计数器

FCW如图5所示。其高位字节有5个控制位：即段使能（SEG），正常/系统（N/S），停止使能（SE），矢量中断使能（VIE），非矢量中断使能（NVIE）。低位字节有6个CPU标志位：即进位（C），零（Z），符号（S），奇偶校验/溢出（P/V），十进制调整（DA），半进位（H），其余位是为将来扩展而保留的。AMZ8002的FCW如图6所示，AMZ8002中除没有SEG位外，其余均与图5完全一致。

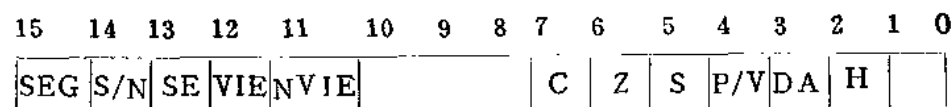


图5 AMZ8001标志和控制字

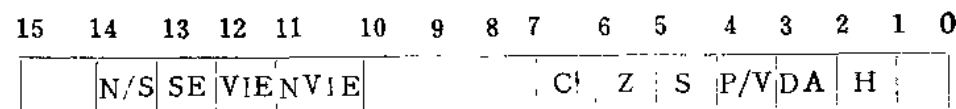


图6 AMZ8002标志和控制字

FCW的标志包含表示数据运算结果特征所需的全部标志。半进位标志仅在字节操作数算术运算时起作用。一个字节有两个四位数字组成。当低4位向高4位有进位时用H标志表示。

十进制调整标志DA有利于对BCD码进行必要的数转换。奇偶校验/溢出（P/V）标志在非算术运算时，表示运算结果的奇偶性，在算术运算时（P/V）标志表示是否产生溢出条件。如果参加加法运算的两个操作数的符号相同，但结果的符号与之相反，则表示产生了溢出。减法就是加减数的2的补码。

CPU采用2的补码表示法，因此最高位为符号位。如果最高位为“1”则表示为负数。符号反映了在执行算术或逻辑指令后结果的最高位状态。

零标志（Z）为1时，表示结果的所有位（包括符号位）都是零。一般说来，这个标志位对算

术和逻辑指令都有影响。进位标志 (C) 表示在执行算术指令后, 最高位位置上有进位。按照惯例, 位是从零开始编号, 因此位7 (第8位) 是字节的最高位, 位15 (第16位) 是字的最高位, 位31 (第32位) 是长字的最高位。

CPU能够处理三种外部中断: 非屏蔽中断, 非矢量中断和矢量中断。NVIE和VIE位控制后两种中断。CPU对非屏蔽中断不能控制。另一方面矢量中断和非矢量中断可以通过FCW的相应位清另而禁止。换句话说, 在收到相应的外部中断信号之前, 中断使能位将必须为1。非屏蔽中断有最高优先权, 矢量中断其次, 非矢量中断的优先权为最低。

停止使能位 (SE) 用于在执行单条指令后, 方便地使微处理器停机。对于微处理机有一个外部输入信号参加这种操作。当SE位为1时, 这个外部输入信号才起作用。

N/S位决定操作方式, 逻辑“1”表示处于系统方式, 逻辑“0”表示正常方式。在系统方式下, 所有指令都有效, 并且都可以执行。在正常方式下, 仅有那些不影响系统完整性的指令才是有效的。在正常方式下, 无效的指令称之为系统指令或特殊指令。系统指令有: 检查或修改FCW控制位指令, 微处理机内部信息交换指令和完成I/O操作指令。如果在正常方式下, 遇到系统指令, 则指令的执行暂停, 并产生陷阱使程序中断。

Amz8001具有23位分段式地址, 而Amz8002采用16位非分段式地址。FCW的SEG位可以使Amz8001工作在非分段寻址方式。当SEG位为1时, CPU工作在分段寻址方式, 当这位为零时CPU工作在非分段寻址方式。在Amz8002中这位总是零。

FCW的高字节为控制位, 因此对这些位进行操作的任何指令都是特殊指令, 亦即系统指令。FCW各位的意义归纳在表3中。

复位时, 存储器单元2 (0段) 的内容自动填入Amz8001的FCW, 而零单元的内容自动存入Amz8002的FCW中。

新程序状态区指示器

程序发生中断时, CPU自动地把当时程序状态存入系统堆栈。程序状态包括有处理器的状态信息和与中断的原因有关的信息。中断前程序状态存入堆栈之后, 新的程序状态被送入FCW和PC。这个新的程序状态存放在由新程序状态区指示器(NPSAP)指定的称之为新程序状态区的存储器内的预定单元内。Amz8001的NPSAP如图7所示。它由二个8位寄存器组成, 其中之一存放7位段号, 另一个则存放偏移量的高8位。显然, 在Amz8002中仅需一个8位寄存器, 如图8所示。仅用这16位地址的高8位。利用LDCTL指令可对NPSAP进行存取。

刷新计数器

Amz8001和Amz8002都有一个刷新计数器, 可有助于动态存储器的操作。如图9所示, 刷新计数器由一个9位二进制行计数器, 一个6位二进制速率计数器和一位刷新使能位组成。速率计数器为可编程的, 模数为64的计数器, 其计数频率为CPU时钟频率的25%。行计数器每当速率计数器溢出时计1。如果刷新使能位为零, 则刷新计数器自动失去刷新功能。CPU初始化复位时, 这位置1, 亦即可以刷新。使用LDCTL指令可对刷新计数器

存取。

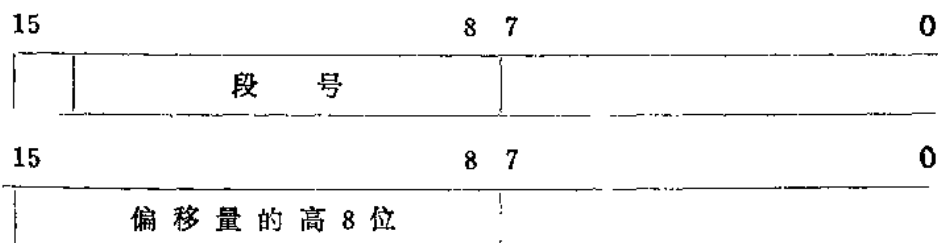


图7 Amz8001新程序状态区指示器

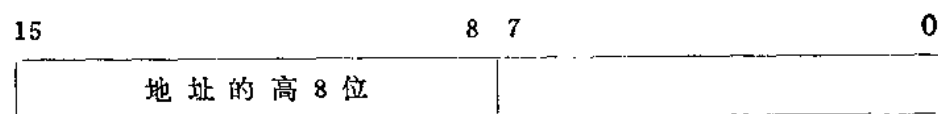


图8 AMZ8002新程序状态区指示器

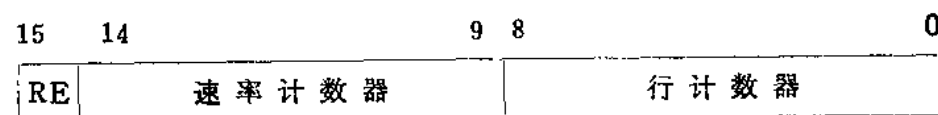


图9 刷新计数器

寻址方式

由寄存器地址,存储器地址或I/O地址确定执行指令所需要的操作数称之为寻址方式。给出指令的寻址方式不仅决定了有关的地址空间,而且决定了计算操作数地址的方法。寻址方式可以由指令明确地给出或由指令隐含地指出。有八种寻址方式:寄存器(R)寻址,间接寄存器(IR)寻址,直接地址(DA)寻址,立即(IM)寻址,变址(X)寻址,基地址(BA)寻址,基变址(BX)寻址和相对地址(RA)寻址。数据块和字符串指令操作中用到的自动递增与递减是隐含寻址方式。

下面详细说明寻址方式。

寄存器(R)寻址方式:指令所用的操作数存放在如图10所示的通用寄存器中,这条指令可确定操作数(字节,字,长字)的长度,指令中的4位位段可确定需用的寄存器。

间接寄存器(IR)寻址方式:这条指令可确定一个通用寄存器,但这个寄存器的内容,不是操作数,而是操作数的地址。由于Amz8001是23位分段地址,所以由指令确定寄存器对,其第一个寄存器内容为7位段号,而第二个寄存器内容为16位偏移量,如图11所示。除RR0以外的任何一个通用寄存器对都可以由这种寻址方式来确定,Amz8002如图12所示,仅要求16位地址,因而除R0以外任何通用寄存器都可以用IR这种寻址方式来确定。

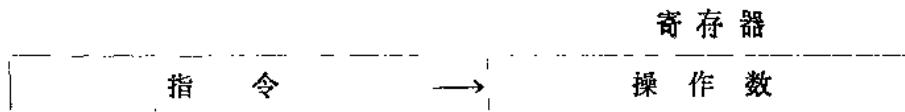


图10 AMZ8001和AMZ8002寄存器寻址方式

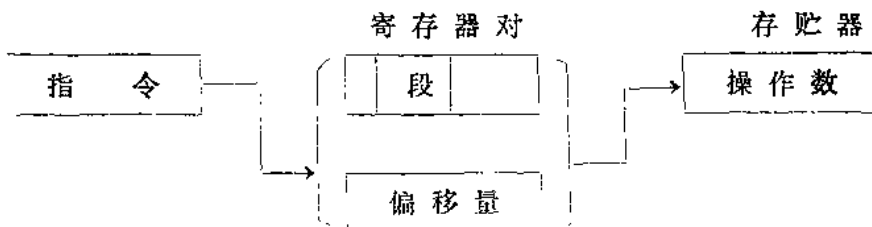


图11 AMZ8001间接寄存器寻址方式

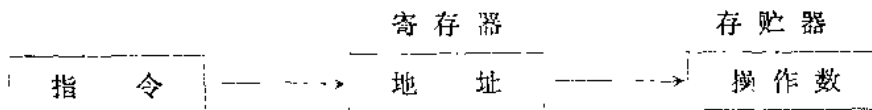


图12 AMZ8002间接寄存器寻址方式

直接地址 (DA) 寻址方式:

这条指令本身可明确地给出一个地址, 本条指令所使用的操作数就放在这个地址内。Amz8001的直接地址寻址方式有二种格式, 即为长偏移与短偏移。在长偏移的情况下, 跟在这条指令操作码后面的存贮单元的内容就是7位段号, 其后的存贮单元内容就是16位偏移量, 如图13A所示。

在短偏移的情况下, 直接跟在指令操作码后面的存贮单元内容即为7位段号和短偏移量 (8位) 如图13B所示。在Amz8002的情况下, 紧跟在这条指令操作码后面的存贮单元内容就是16位地址, 如图14所示。

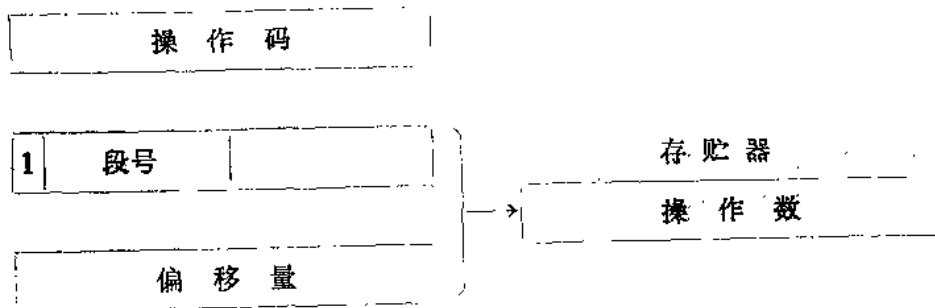


图13A AMZ8001直接寻址方式——长偏移

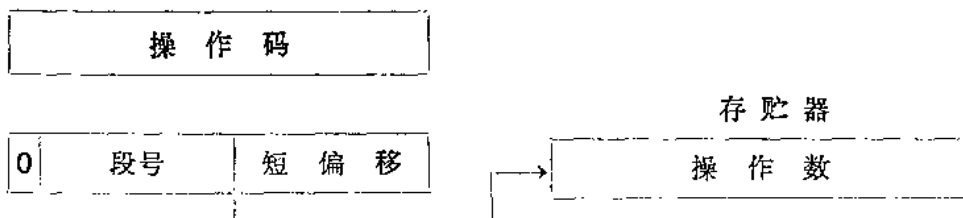


图13B AMZ8001直接寻址方式—短偏移

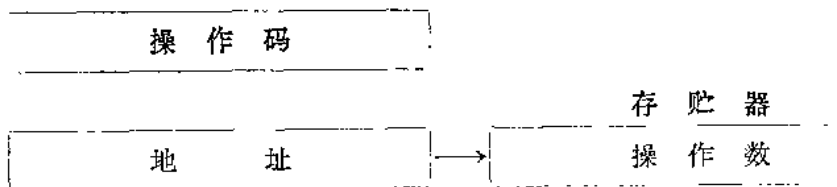


图14 Amz8002直接寻址

立即 (IM) 寻址方式:

这条指令本身就包括了操作数, 如图15所示。通常, 这条指令操作码后面的存储单元内容就是立即操作数。在32位立即操作数的情况下, 立即操作数为这条指令操作码后面二个存储单元的内容。

变址 (X) 寻址方式:

该指令指定一个16位通用寄存器作为变址寄存器, 除R0以外的任何通用寄存器都可以用作为变址寄存器。这条指令也可以象直接寻址方式那样指定一个地址。在Amz8001中, 把指定为变址寄存器的16位内容加到由这条指令确定的16位偏移量上。变址及16位偏移量都是不带符号的16位整数, 在这两个数相加过程中, 由高位产生的进位都可忽略不计。相加后得到的16位数与指令中指出的7位段号, 象图16A中说明的那样, 作为23位分段式地址。这条指令用的操作数就放在存储器的这个地址内。如果Amz8001的变址寻址方式采用短偏移方式的话, 紧跟在指令操作码后的存储字的内容就是7位段号和8位偏移量, 如图16B所示。由高8位为另和低8位为给定的偏移量构成一个不带符号的16位整数。把这样形成的16位数加到指定的通用寄存器内的不带符号的16位数上。在相加过程中由最高位产生的进位可忽略不计。所得到的16位数与给定的7位段号构成23位地址。这条指令所用的操作数就在存储器的这个地址里。在Amz8002中, 紧跟在指令操作码后面的存储字中存放的是16位地址, 如图17所示。这不带符号的16位整数与指定为变址寄存器中存放的不带符号的16位数相加, 在相加过程中, 由最高位产生的进位都忽略不计, 相加所得到的16位数就是操作数在存储器中存放的地址。

操 作 码	数 字 操 作 数
-------	-----------

4位立即操作数（数字）

操 作 码	字 节 操 作 数
-------	-----------

8位立即操作数（字节）

操 作 码

字 节 操 作 数

另一种8位立即操作数（字节）

操 作 码

字 操 作 数

16位立即操作数（字）

操 作 码

操 作 数 的 高 16 位

操 作 数 的 低 16 位

32位立即操作数（长字）

图15 AMZ8001和AMZ8002立即寻址方式

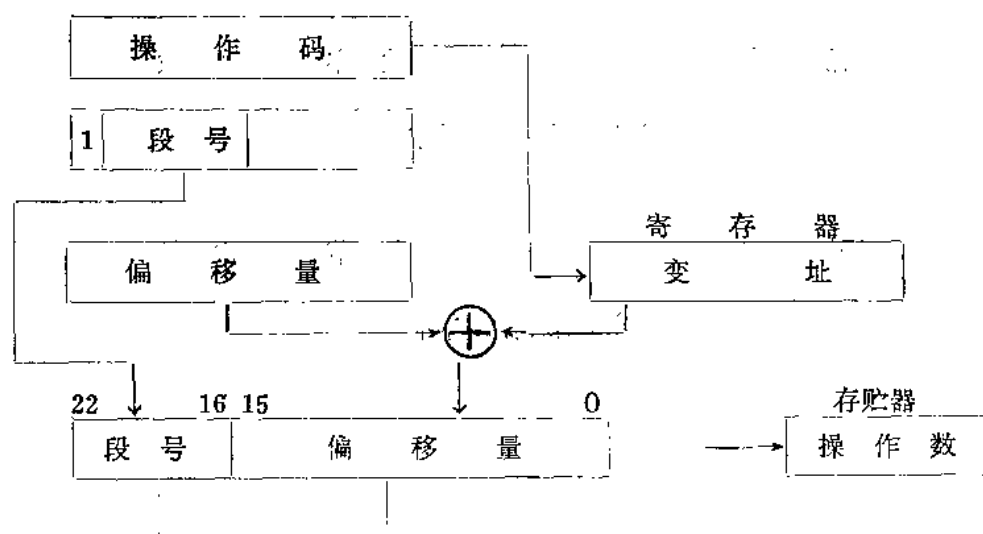


图16A AMZ8001变址寻址方式(长偏移)

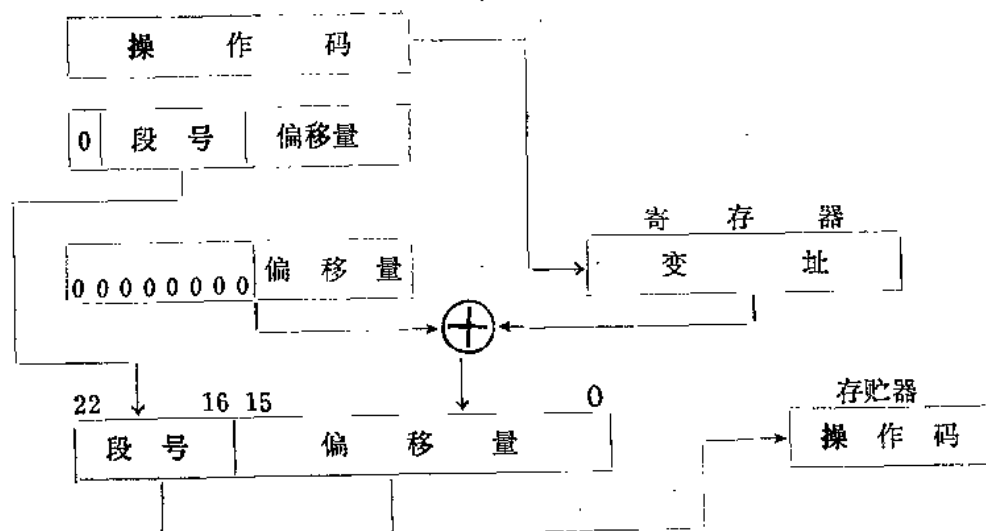


图16B AMZ8001变址寻址方式(短偏移)

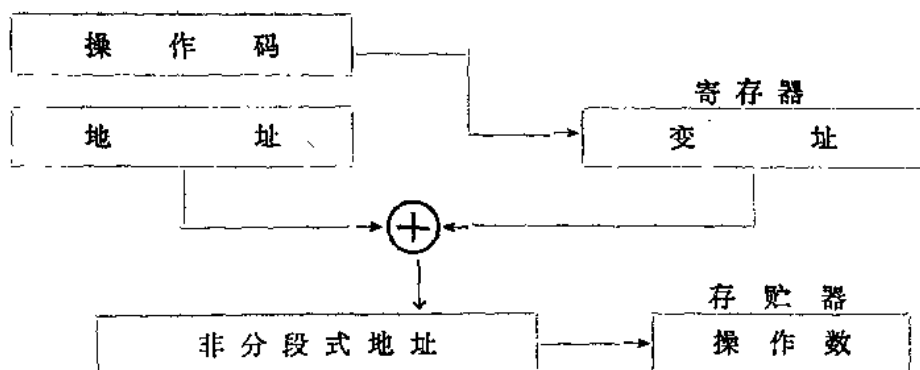


图17 AMZ8002变址寻址方式

基地址 (BA) 寻址方式:

这指令指定一个通用寄存器作为基地址寄存器。在Amz8001情况下, 这指令指定一个寄存器对作为基地址寄存器, 7位段号放在一个寄存器中, 16位偏移量放在另一个寄存器中, 如图18所示。在Amz8002中, 指定的基地址寄存器中存放16位地址, 如图19所示。除R0或RR0外任何通用寄存器或寄存器对都可以作为基地址寄存器, 紧跟在操作码后面的存储单元中的内容为16位位移量。位移量与基地址都是不带符号的二进制整数。16位位移量与16位基地址 (在Amz8001是16位偏移量) 相加, 相加过程中由最高位产生的进位忽略不计。所得的16位值 (与Amz8001基地址的段号一起) 就是操作数在存储器中的地址。

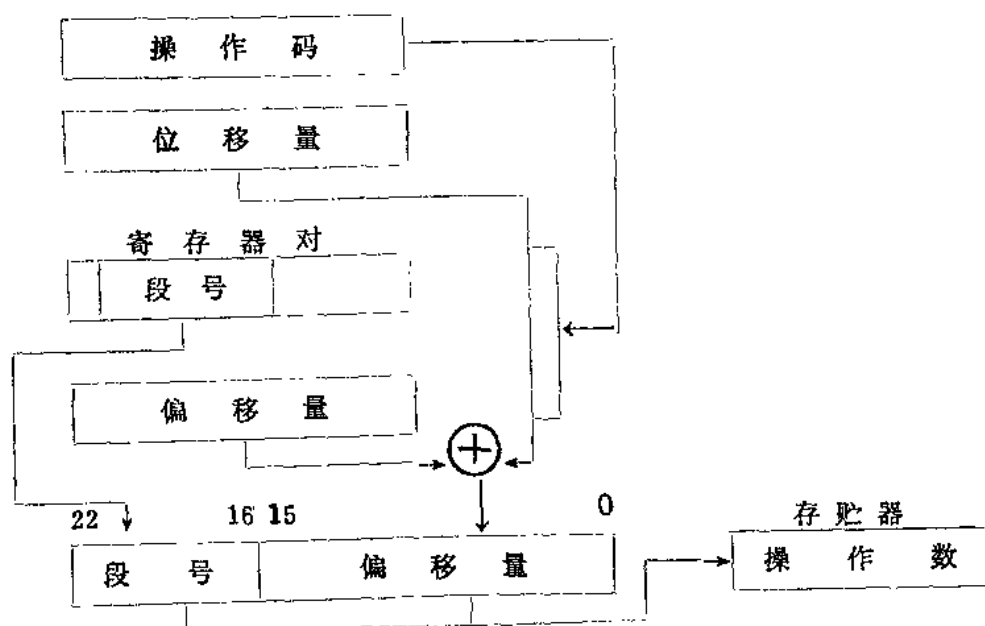


图18 AMZ8001基地址方式