[19]中华人民共和国专利局

[11]授权公告号
CN 1021380C

[12] 发明专利说明书

[21] 专利号 ZL 90103808

[51]Int.Cl[5]
G06F 13/42

[45]授权公告日 1993 年 6 月 23 日

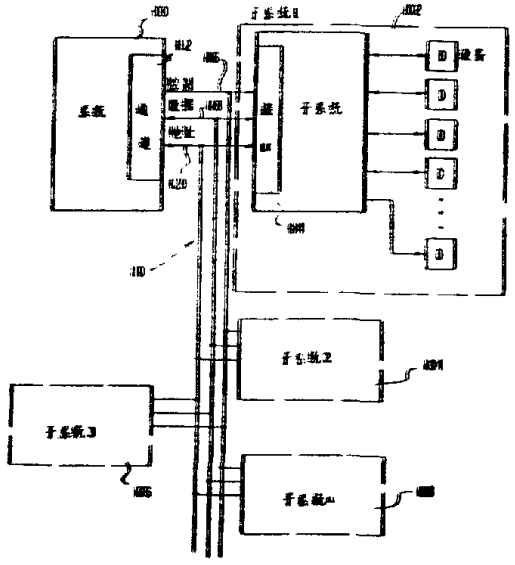
[24]颁证日 93.4.16
[21]申请号 90103808.3
[22]申请日 90.5.28
[30]优先权
[32]89.6.9 [33]US [31]07/364,931
[73]专利权人 国际商业机器公司
地 址 美国纽约
[72]发明人 弗朗西斯·迈克尔·博内维托
道格拉斯·罗德里克·奇泽姆
萨迈·戴维斯·多兹 迪鲁维克昂
阿·M·德萨伊 欧内斯特·纳尔逊
·曼德斯 安德鲁·博伊斯·麦克
尼尔 理查德·尼尔·门德尔斯奥
[74]专利代理机构 中国国际贸易促进委员会专利
代理部
代理人 范本国

说明书页数: 附图页数:

[54]发明名称 计算系统的命令传递

[57]摘要

一个命令接口包括若干端口用于在一个主处理器和一个可能带有若干附加设备的智能子系统之间传递信息。一个命令接口端口从主处理器接受一个直接命令或间接命令，这些命令指出要由该子系统或某一附加设备完成的操作的类型。一个引起注意端口从主处理器接受一个代码，它指出在命令接口端口接受的是直接命令还是间接命令，它还指出是这个智能子系统还是一个设备要去执行该命令。一个命令忙/状态端口从该智能子系统接受一个代码。



1．一个数据处理系统，包括主系统和至少一个可能有附加设备的子系统，其特点是：

一个命令接口，用于在所述主系统和所述一个子系统之间传送信息，该命令接口包括：

第一端口，用于接受来自所述主系统的直接命令或间接命令，这些命令指出要由该子系统或一个附加设备来完成的操作的类型；和

第二端口，用于接收来自所述主系统的代码，指出在所述第一端口接受的是所述直接命令还是所述间接命令，还指出是所述一个子系统还是一个附加设备要去执行在所述第一端口接受的命令。

2．如权利要求1所述的数据处理系统，其特点是：所述直接命令是一个立即命令，它包含了所述一个子系统或一个设备立即走执行该命令所必须的全部信息；而间接命令是一个子系统控制块（SCB）的地址，子系统要从主系统取出这个SCB供其后去执行。

3．权利要求2中的数据处理系统，其特点是：在所述第二端口接收的代码不仅指示出是否为间接命令，还编码指出所接收的是一组不同的立即命令中的哪一个。

4．权利要求2中的数据处理系统，其特点是：有一个第三端口用于从所述一个子系统向所述主系统提供代码，以指出所述第三端口和第二端口是否为“忙”，还指出是否一个子系统在接受或拒绝命令。

5．权利要求4中的数据处理系统，其特点是：有一个第四端口

用于从所述主系统向所述子系统提供控制信号；所述控制信号包括一个子系统复位信号。

6．权利要求5中的数据处理系统，其特点是：有一个第五端口，用于从所述子系统向所述主系统提供一个代码，指出是所述一个子系统还是一个设备在向主系统提供中断。

7．操作一个数据处理系统的方法，该系统包括一个主系统和至少一个可能带有附加设备的子系统，还有一个在所述主系统和所述一个子系统之间传递信息的命令接口。该方法以下述步骤为特征：

从所述主系统向所述命令接口上的第一端口提供一个直接命令或间接命令，这些命令指出要由所述子系统或一个附加设备完成的操作类型；以及

从所述主系统向所述命令接口的第二端口提供一个代码，它指出在所述第一端口接受的是所述直接命令还是所述间接命令，该代码指出是所述子系统还是某个附加设备要去完成在所述第一端口接受的命令。

8．权利要求7中的方法，其特点在于：所述直接命令是一个立即命令。它包含为所述子系统或某一设备要立即执行该命令所必须的全部信息，而间接命令是一个子系统控制块（ＳＣＢ）的地址，子系统从主系统中取出这个ＳＣＢ供其后执行。

9．权利要求8中的方法，其特点在于：在所述第二端口接受的代码不仅指出是否为间接命令，还编码指出是一组不同立即命令类型中的哪一个。

１０．权利要求8中的方法，以下述步骤为其特征：

从所述一个子系统向所述命令接口上的第三个端口提供一个代码

由所述主系统使用，该代码指示出所述一个子系统是否在接受或拒绝命令。

11．权利要求10中的方法，以下述步骤为特征：

从所述子系统向所述命令接口的第四个端口提供所述子系统需要的控制信号，这些控制信号包括一个子系统复位信号。

12．权利要求11中的方法，以下述步骤为特征：

从所述一个子系统向所述命令接口的第五个端口提供一个代码，指出是哪些设备正在向所述主系统提供中断。

计算系统的命令传递

本发明属于计算系统领域，特别是以命令传递机制为其目标。明确地说，本发明的目标是从个人计算机一类机器向一组可以带有附加设备的智能子系统传递命令。

本发明中参考了题目为“计算系统的中断处理”的美国专利申请。该项申请已转让给本专利申请的受让人，它讲述的中断处理机制可以被用于本项申请中的命令传递系统。

在一个由主系统及至少一个子系统构成的数据处理系统或计算系统中，通常需要经过输入／输出（I／O）通被将直接命令和间接命令传送给一个单个的子系统。

直接命令（*direct command*）是指能够在一次操作中直接送给一个子系统的命令，它是完全独立和完整的，完整地指定了要由子系统完成的一个操作。与此相反，一个间接命令（*indirect command*）为了指定一个操作所需要的数据量超过了一个I／O通道周期所能提供的数据量。间接命令本身是一个指向系统存贮器中一块（*block*）的指针（*pointer*），这一块中包含了要完整地指定由子系统完成的一个操作所需要的全部参数。这一个存贮块常被称作“控制块（*Control block*）”。

在所有的计算机系统中，都至少存在一个I／O指令（*instruction*），它使主系统中运行的程序能够通过外部接口或通道发送数据和／或命令。每次I／O操作所能发送的数据量以及

从通道或子系统收到的与接收或完成操作有关的状态都是取决于系统的。

“系列／1操作原理”（IBM出版物号GA34－0152－0）描述了属于IBM系列／1的那一类计算机中的操作，每个I/O操作把下述信息传送给安装在系列／1 I／O通道中的外部设备附加插件：

（1）命令所指向的子系统的地址。

（2）一个8位（*bit*）命令字段，用于指出该操作是直接的或是间接的。

（3）一个18位数据字，它或者包含了关于一个直接命令的数据，或者在间接命令情况下的一个设备控制块（DCB）的地址。

此外，在通道上执行每个I／O指令周期期间，还通过各种指示代码表示出状态，记录下代表I／O操作状态的状态代码。这种立即反映的实时的状态指示使主程序能够确定该命令是否被子系统成功地接收，或者是否出现误操作，例如“设备未联接”或“立即命令拒绝”在出错情况下，主程序能够在稍后再送入这个命令。

因为系列／1子系统能够立即接收或拒绝一个I／O命令，所以不需要使用优先中断或其他机制来向主程序报告立即状态。

为了保证整个系统的完整性，总是必须有一种机制来报告一个I／O命令是否为子系统接收去执行。主系统不能假定一个命令被子系统成功地接受，因为有很多种失败方式（*mode of failure*），它们不能被检测出来，因而也不能被修复。

现代化计算系统中使用的I／O子系统的复杂性不断增长，这表明在个人计算机一类机器上也需要有复杂的命令。

根据本发明，这种需要是这样满足的：提供一种以子系统控制块（SCB）的形式在主系统和一组子系统之间传送命令的机制。不象在系列／1中那样用一个独特的处理器指令来传递地址段和命令类型段，而是把这些字段提供给一个单一端口，称作“引起注意端口”（*attention port*）”，以适应于一般以Intel 80286／386／486为基础的处理器系统中当前使用的指令集。值得称道的是，本发明的实际应用中可以使用以其它处理器为基础的系统。

再有，如在系列／1类型机器中那样，在个人计算机类机器中不存在用于报告经I／O通道传送直接或间接命令时的立即状态的机制。

为满足这一需要，在每个子系统内部提供了一个“命令忙／状态端口（*Command Busy／status Port*）”，由它提供前一个I／O操作的状态以及子系统的当前状态，这个端口可以由主程序使用一个I／O读命令来读出，而该命令不会修改或更新该端口中的状态。只有指向子系统内其它端口的输出（写）命令才会影响“命令忙／状态端口”的状态。

本发明的一个目的是为计算系统提供一种命令传递机制。

本发明的又一目的是提供一种计算系统，其中的一个系统向至少一个可能带有附加设备的子系统提供直接或间接命令。

本发明的一个目的是提供一种计算系统，其中的主处理器向至少一个可能带有附加设备的子系统提供一个立即命令（*immediate Command*）或子系统控制块（SCB）地址，于是子系统执行这个立即命令或者取回SCB供下一步去执行。

本发明的再一个目的是提供一种计算系统，其中一个命令接口

（*Command interface*）在主处理器和至少一个智能子系统之间传送信息该智能子系统可能带有一组附加设备。这个命令接口包括一个“命令接口端口”和一个“引起注意端口”，命令接口端口接收一个立即命令或者存储在系统存储器中的ＳＣＢ的地址，而引起注意端口接收来自主处理器的代码，它指出在命令接口端口接收的是立即命令还是ＳＣＢ地址，它还指出是该智能子系统还是一个设备要去执行这一命令。一个初始命令和一个初始代码分别被锁存在命令接口端口和引起注意端口，这样，在初始命令被处理之前，下一个命令和代码便不能够重写各自的端口。

本发明还有一个目的是提供一个计算系统，其中的命令接口在主处理器和至少一个可能带有附加设备的智能子系统之间传送信息。该命令接口包括命令接口端口、引起注意端口和命令忙／状态端口。命令接口端口接收一个立即命令或存贮在系统存贮器中的ＳＣＢ的地址。引起注意端口接收来自主处理器的代码，指出在命令接口端口接收的是立即命令还是ＳＣＢ地址，还指出是该智能子系统还是一个设备要执行该命令。命令忙／状态端口接收来自一个子系统的代码，指出命令接口端口和引起注意端口是否处于智状态，还指出该智能子系统是否是在接收或拒绝命令。这个端口能由主处理器读出而不改变或更新该端口状态。

本发明还有一个目的是提供一种计算系统，其中的命令接口在主处理器和至少一个可能带有附加设备的智能子系统之间传送信息。该命令接口包括一个命令接口端口和一个引起注意端口。命令接口端口接收一个立即命令或存贮在系统存贮器中的ＳＣＢ的地址。引起注意端口接收来自主处理器的代码，指出在命令接口端口接收的是立即命

令还是SCB地址，还指出是该智能子系统还是一个设备要去执行这一命令。SCB命令可以按一种特定的顺序链接，并作为一个逻辑请求（*logical request*）来处理。SCB命令的链接使智能子系统接收和执行这些命令时不必引起与主处理器操作的过多的交接（*interfacing*）。对一个具体的命令，还可能有多个数据缓存器的链接。缓存器的链接是用间接表（*indirect list*）来构成的。每个SCB包含一个"终止状态块（Termination Status Block）"（TSB）在系统存贮器中的地址，正在处理SCB的那个子系统把SCB的完成或终止状态存贮到TSB之中。

本发明的目标是一个命令接口，它包括若干端口用于在主处理器和至少一个可能带有附加设备的智能子系统之间传送信息。一个命令接口端口接收来自主处理器的一个直接命令或一个间接命令，这里所说的直接命令包含了一个智能子系统要立即执行该命令所必需的全部信息，而间接命令是一个子系统控制块（SCB）的地址，该子系统控制块存贮在主处理器中系统存贮器的一个可寻地区间。一个引起注意端口接收来自主处理器的代码，代码的第一部分指出在命令接口端口所接收的是直接命令还是间接命令，代码的第二部分指出是这个智能子系统还是一个附加设备要去执行这一命令。一个命令忙／状态端口包含可由主处理器读出的信息，这些信息反映了要下一步执行命令的子系统或其附加设备是立即接收了还是拒绝了一个命令。根据写在引起注意端口的代码的第二部分，由一个智能子系统或所附加的设备来直接执行一个直接命令，如果是一个间接命令，该智能子系统便按照所指出的SCB地址直接访问系统存贮器。然后该智能子系统或一个设备去完成由被访问的SCB所指定的操作。

图1是一个计算系统的框图，该系统包括一个主系统联接一组智能子系统，每个子系统又可能联接一组设备；

图2的框图给出主系统和一个智能子系统的详细情况；

图3是在主系统和一个智能子系统之间交换信息的命令接口的框图；

图4是给出引起注意端口详细情况的框图；

图5是给出命令忙／状态端口详细情况的框图；

图6具体说明了用于立即命令的命令接口端口与引起注意端口的（信息）一般格式；

图7具体说明了用于子系统控制块（SCB）的命令接口端口和引起注意端口的（信息）一般格式；

图8具体说明了用于本发明的计算系统中的控制块SCB的详细格式；

图9给出一个详细的扩展SCB格式；

图10给出详细的SCB终止状态块（TSB）格式；

图11给出详细的扩展TSB格式；

图12是计算系统的框图，详细给出系统存贮器中的SCB格式；

图13是在提供命令期间子系统状态变换过程的状态图；

图14是本发明的命令传递系统的流程图；

图15A、15B和15C如图15所示的那样拼在一起，图16是写子系统控制端口过程的详细流程图，它的概况已示于图14中；

图17A和17B如图18所示的那样拼在一起，给出写引起注意端口过程的详细流程图，它的概况已示于图14；

图18是写接口端口过程的详细流程图，它的概况已示于图14；

图19A和19B合起来作为图19，图20A和20B合起来作为图20，它们给出命令译码处理的详细流程图。

本发明的总体目标是为主计算机系统中运行的程序提供一个命令传递系统。直接或间接命令经由命令接口传递给至少一个可能带有一组附加设备的智能子系统。为便于描述，把本发明置于在IBM个人系统／2（PS／2）计算机上运行的程序所使用的命令传递系统范围内来描述。个人系统2（Personal System 2）和PS／2各为IBM公司的注册商标。直接命令（诸如立即命令）和间接命令（诸如存贮在PS／2系统存贮器中的子系统控制块的地址）经由IBM微通道（Micro Channel（微通道）是IBM公司的注册商标）传送给一个智能子系统的命令接口，这个子系统可能有一组附加设备，例如磁带驱动器、磁盘驱动器、打印机、通讯设备和显示设备等等。然而，值得称道的是，本发明可以在其它主计算机系统中实现，经由接口网络把命令传递给智能子系统。

定义了一种子系统控制块（SCB）结构，它提供了主处理器和具有存贮器直接存取（DMA）能力以及处理命令级操作能力的子系统之间的一个控制块级接口。一个SCB内包含了一个单个命令。而命令链接可以把一个SCB指向一个或多个其它SCB的地址，而把这个链作为一个逻辑命令来处理。还有伴随一个给定命令的多个缓存器的数据链接。用一个间接表（indirect list）来构成这种缓存器链。

一个SCB包含系统存贮器中一个终止状态块（TSB）的地址。由处理命令的子系统把给定SCB的完成或终止状态放到TSB中。为了能在命令链中的任何一点处理终止操作，对链中的每个SCB都

定义了一个ＴＳＢ。

在下面对命令传递系统做一般性描述之后，将详细描述ＳＣＢ和ＴＳＢ。

现在参考图１，它是根据本发明绘出的一个计算系统的总体框图。一个主计算机系统１００通过总线结构１１０与一组智能子系统１０２、１０４、１０６和１０８交换信息。总线结构１１０把主计算机系统１００的通道１１２与每个子系统的一个接口（例如子系统１０２中的接口１１４）连接起来。子系统１０４、１０６、１０８的每一个都与子系统１０２相似。总线结构包括一个用于交换控制信号的控制总线１１６、交换数据信号用的数据总线１１８以及交换地址的地址总线１２０。

图２给出系统１００和子系统１０２的细节，其它子系统的构成方式与子系统１０２相似。

主系统１００包括一个系统处理器组合或主处理器１２２，它经由总线１２６与系统存贮器１２４相连。系统处理器组合１２２与一个个人计算机类微处理器组成，例如所提到的ＰＳ／２，由微通道１２６连到系统存贮器１２４上。ＰＳ／２和微通道的操作在“ＰＳ／２硬件接口技术参考书”（ＩＢＭ出版序号６８Ｘ－２３３０－０）描述。存贮器１２４可由主处理器１２２访问，也可由子系统１０２通过存贮器直接存取（ＯＭＡ）访问。

子系统１０２包含一个附加器（*attachmend*）或适配器，它包括一个与微通道１２６相连的微通道接口。该接口包含一组输入／输出（Ｉ／Ｏ）端口，它们的详细情况将由参考图３来描述，这个接口使系统主处理器能直接向子系统发出命令和接受来自系统的回答。

这些端口可以被映射（*map*）到系统存贮器空间或I／O空间。这个接口还向微通道提供适配器所需要的主控和／或受控功能。

命令包括直接（或立即）命令（如系统存贮器124中的子系统控制块（SCB）的地址）。SCB传递支持逻辑130把SCB命令提供给一个本机微处理器132，例如*Intel* 80188类微处理器（其操作在“*Intel* 微处理器和外围设备手册”中描述）在那里处理该SCB命令。SCB传递逻辑130还把本机处理器中断提供给处理器132中的中断逻辑134。图17—23中给出了详细的流程图来解释包含在SCB传递逻辑130和微处理器132中的命令处理器和译码处理器功能。

微处理器132响应SCB命令，通过附加设备接口146（例如，它可以是小型计算机系统接口（SCSI））对附加设备136、138、140和142的各项任务安排时间顺序。对SCSI接口操作的描述见“信息系统美国国家标准研究所（ANSI）——小型计算机系统接口（SCSI）”，其出版号为ANSI×3．131－1986。附加设备可以是磁盘驱动器、磁带驱动器、直接存贮器存取设备（DASD）、打印机、显示器以及其他类似设备。设备D（0）136用于把命令指向子系统。智能子系统136和设备138、140、142的每一个都被主处理器看作是一个逻辑设备，对每一个都指定一个设备标识号，分别为D（0）、D（1）、D（2）和D（3）、这些设备便借助这些标识号被访问。

图3给出微通道接口128的I／O端口。定义为控制一个子系统的端口必须按如下顺序分配在I／O地址空间中：

（1） 命令接口端口148。

（2） 引起注意端口１５０。

（3） 子系统控制端口１５２。

（4） 中断状态端口１５４。

（5） 命令状态／忙端口１５６。

这些端口能映射到系统存贮器地址空间而不放在Ｉ／Ｏ地址空间但为了便于描述，所以选择了Ｉ／Ｏ地址空间。因此，在本文中使用“ＯＵＴ（输出）”或“ＩＮ（输入）”命令的地方，如果各端口被映射到系统存贮器空间，则应该用“系统写”或“系统读”命令代替。

命令接口端口１４８也可以称作命令端口，它是一个“子系统读／系统写”寄存器。这个３２位寄存器用于把一个双字（３２位）指示（*Direct*）从系统传送到子系统。该双字指示可以是一个立即命令或者一个间接命令（如ＳＣＢ地址）。该立即命令或ＳＣＢ指定了要由子系统完成的操作。

系统程序把一个立即命令或一个ＳＣＢ地址写入命令接口端口１４８，然后去写引起注意端口１５０。子系统在命令忙／状态端口１５６写入“忙（*busy*）”位。当系统已经写入引起注意端口150时，便通过中断１３４向本机级处理器１３２发出了信号（图２）。子系统响应中断，读引起注意端口和命令接口端口，然后确定要完成的操作的类型,以及命令指向的设备。子系统或者将按命令接口端口提供的地址去取出ＳＣＢ，或者执行立即命令。

引起注意端口１５０是一个“子系统读／系统写”寄存器。这个8位或１６位寄存器由系统程序用于要求子系统注意去完成一个命令。寄存器中的各位指出所要求的动作及相应的设备号。换句话说，引起注意端口接收的代码有两部分，第一部分指出在命令接口端口１４８

接收的是立即命令还是ＳＣＢ地址，第二部分指出是子系统还是一个附加设备要去完成在命令接口端口１４８所接收的命令。当引起注意端口１５０由系统写入时，在命令忙／状态端口１５６中的“忙”位立即由子系统置1。这个“忙”位将一直保持置位，直到子系统读出注意请求（*attention request*）及相应的命令接口端口内容，并确定了它能否接受此命令时为止。在“忙”位为零时，子系统必须在任何时候都能接收新的命令。“忙”位也将保持置位直到子系统完成“子系统复位”时为止。

现在参考图４来说明引起注意端口（ＡＰ）的详细情况。主处理器具有对ＡＰ的读／写访问。当对ＡＰ进行“系统软件写（ＳＳＷ）”或“系统软件读（ＳＳＲ）”操作时，微通道Ｉ／Ｏ地址必须等于“建立（*set up*）”微通道来选定子系统时赋予ＡＰ的值。

ＡＰ在ＡＰ寄存器１４３的输入端接收8位引起注意命令１４１。当“与（ＡＮＤ）”门１５３为有效（*active*）时，输入的引起注意命令被记录在寄存器１４３中，其引起注意命令被提供给寄存器输出端１５７。每当门１５３有效时，便在线１５５上提供一个“置ＡＰ中断（ＳＥＴ－ＡＰ－ＩＮＴＲＰＴ”信号。这个信号是对子系统本机处理器的一个中断，通知它已经由主处理器向ＡＰ写入了一个命令。每当门１５３失效时，引起注意命令便不能被记录寄存器143，因而被子系统忽略。

有４个输入端来控制门１５３是否有“１”输出。用正号（＋）代表正逻辑输入，用负号（－）代表负逻辑输入。线１４５上的第一输入是一个＋ＭＣ－ＡＰ－ＳＥＬ信号，当它为１时，指出ＡＰ的微通道地址有效，而且主处理器已选定了该子系统的ＡＰ。在线１４７

上的第二输入是一个＋ＳＳＷ－ＲＥＧ－ＣＬＫ信号，当它为１时，指出对于这个子系统Ｉ／Ｏ地址空间其微通道Ｉ／Ｏ写是有效的。第三和第四个信号是由命令忙／状态端口（ＣＢＳＰ）提供的。在线１４９上的第三信号是一个－ＣＤＳＰ－ＢＩＴＯ－２Ｂ信号，它是一个“忙”位（Ｂ）信号。由于它是一个负逻辑信号，当Ｂ为零时线１４９上的信号为１，当Ｂ为１时线１４９上的信号是零。线１５１上的第四输入是一个－ＣＢＳＰ－ＢＩＴ４－（ＲＥＪ）信号，它是拒绝（ＲＥＪ）信号。由于这是一个负逻辑信号，当ＲＥＪ为零时在线１５１上的信号为１，而当ＲＥＪ为１时在线１５１上的信号为零可以看到，每当线１４５、１４７、１４９和１５１均为１时，线１４１上的引起注意命令便被记入寄存器１４３，而线１５５上的ＳＥＴ－ＡＰ－ＩＮＴＲＰＴ信号便通知本机处理器在ＡＰ已已写入了一个命令。反之，如果这些线上有任意个为零，则门１５３提供一个零输出，它禁止把命令写入ＡＰ寄存器１４３。具体地说，如果Ｂｕｓｙ（忙）位或ＲＥＧ位有任一个为１，则对ＡＰ寄存器１４３的命令写被封锁。

子系统控制端口１５２是一个子系统读／系统写寄存器。这是一个８位读／写端口，它提供了对子系统范围内几个功能的直接硬件控制。在这个端口包括的各位用于控制子系统和全部附加设备的复位，允许或不允许子系统向主处理器呈现物理中断，子系统ＤＭＡ允许／不允许功能，以及使命令被拒绝因而没有被子系统执行时所引起的错误复位的功能。子系统控制端口１５２的格式和功能示于图３中的位０－７，其功能如下：

位０－ＥＩ－使子系统允许中断。这一位置１时允许子系统向系

统发出物理中断。当子系统被接通电源或由“子系统复位”命令使子系统复位时，ＥＩ位被初始化成零。当ＥＩ位为零时，子系统不允许物理中断，不允许向主处理器发送一个物理中断。这一位保持不变，直到由发给该端口的“ＯＵＴ（输出）”命令或子系统复位命令来明确复位时为止。

位１－ＤＭＡ－允许ＤＭＡ。当这一位置零时，不允许子系统起动任何ＤＭＡ操作。当置该位为零时，便把该子系统置于这样的状态：它不能执行任何ＳＣＢ命令或存贮任何ＳＣＢ逻辑中断状态（ＴＳＢ）。当子系统接通电源时，这一位被置零，当执行“子系统复位”操作时这一位被复位为零。这一位保持置位，直到被发给该端口的“ＯＵＴ”命令或“子系统复位”来明确复位为止。（ＤＭＡ位预定作为硬件查错的手段。它也可以用作强迫子系统停止微通道活动的手段。）

位２和位３－Ｓ·Ｄ·－子系统独立。该端口的位２和位３被保留用于子系统的独立功能。

位４和位６－Ｒ－保留位。位４和位６被保留而且必须置零。

位５－ＲＲ－拒绝（状态）复位。当主处理器置这一位置为１，而同时子系统又处于拒绝（*Reject*）状态时，通过对ＲＲ位置“通（ＯＮ）”（也就是置４）将清除拒绝状态。当拒绝状态被清除时，在“命令忙状态”中的“忙”（Ｂ）位和拒绝（ＲＥＴ）位被置零。如果子系统不是处于拒绝状态，那么对ＲＲ写入１时会引起效应。当子系统接通电源或被子系统复位操作来复位时，ＲＲ位被置成零。这一位的设置将保持不变，直到由特定的“ＯＵＴ”命令发到该端口或由子系统复位操作才明确地收该位复位。对ＲＲ位写入零不会影响子系统的内部状态。

位7－RST－子系统复位，这一位用于提供一个由硬件控制的子系统及其附加的全部设备的复位。当该位被置1时，子系统进入复位未决状态（*Reset Pending State*），并对命令忙状态端口中的忙位（B）置1。在子系统上的全部设备的活动均被停止。程序必须在子系统指定的时间周期内对RST写入一个零。如果在复位未决状态时RST位从1转换到零值便把子系统放到“进行置位状态（*Reset in Progress State*）”。在这一状态，子系统完成对它本身及它的所有附加设备的复位。当复位完成时，“忙”位将置零，并将申请主处理器中断。

中断状态端口154是一个系统读／子系统写寄存器。这个8位或16位寄存器被子系统用于向程序呈送中断数据。在这一端口呈送的中断有立即命令中断、SCB命令带有严重硬件失效中断、无任何SCB命令中断以及SCB命令中断。在伴随每个SCB的终止状态块（TSB）区域提供了附加的SCB完成状态指示。TSB的功能在下文中描述。在ISP被系统软件明确复位之前，任何新的中断都不能通过ISP呈交给系统。

这个用于中断状态端口154的8位寄存器的构成示于图3，其各个位的定义如下：

位7－4－中断标识码（*Interrupt* ID）。这些位以一个数字编码，用以识别提交的中断的原因。程序借助检查TSB状态能够得到有关SCB中断的更多信息。

位3－0－设备地址。这是引起所提出的中断的设备地址。

16位中断状态端口（未画出）的格式如下：

位15－12－中断标识码。这些位以一个数字编码，用以识别

提交的中断的原因。程序借助检查ＴＳＢ状态能够得到有关ＳＣＢ中断的更多信息。

位１１－０－设备地址。这是引起所提出中断的设备地址。

命令忙／状态端口１５６是一个子系统写／系统读寄存器。这个８位端口有两个功能：第一、在每个命令被提交之前必须由系统读这个端口，以确定先前提交的命令的状态。在提交一个命令之后，程序必须允许子系统有一个规定的延迟时间段，然后再试图读此端口来获得命令状态。

第二、该端口指出子系统是否有一个有效的中断值呈交给中断状态端口ＩＳＰ１５４。除非命令忙／状态端口指出所呈交的是有效值即ＩＶ（中断有效标志）为１，这个ＩＳＰ不应被程序读出。

命令忙状态端口１５６的格式和功能如下：

位０－Ｂ－忙位。这一位指出当前命令接口和引起注意端口是否被使用。当忙（Ｂ）位为１时，程序不应该试图向子系统写入命令或引起注意端口。如果子系统处于下述状态，则忙位（Ｂ）为１：复位未决（*Reset Pending*），复位正在进行、拒绝、或译码（下文将解释这些状态）。如果主程序在忙位（Ｂ）为１的情况下去写引起注意端口和／或命令端口，那些向这些端口写的企图被子系统所忽略，而且最新的试图执行的命令被子系统忽略而不给出任何出错指示。

当子系统进入如下状态时这一位置１：复位未决、复位正在进行拒绝或译码。当子系统控制端口的ＲＳＴ位置１时，该子系统进入复位未决状态。如果子系统处于复位未决状态，而子系统控制端口的ＲＳＴ特被置零，那么子系统便进入“复位正在进行”状态。当向设备０（即子系统本身）发出一个“设备复位命令”时，该子系统也进

入“复位正在进行”状态。

当子系统完成了子系统复位命令后退出“复位正在进行”状态之后，忙位被置零。

当程序利用写引起注意端口而提交一个新命令时，如果该子系统又不处于复位未决、复位已在进行或译码状态，那么该子系统便进入译码状态。当子系统接收一个命令而退出译码状态时，忙位也被置零当 在译码状态被处理的命令遭到拒绝时，子系统退出译码状态而进入拒绝状态。在拒绝状态，拒绝位（ＲＥＪ）和忙位（Ｂ）均被置1。状态位S将指出该命令被子系统拒绝的原因。当子系统进入复位未决状态时，拒绝状态或译码状态便被清除掉。

位 1 －ＩＶ－ 中断有效。在子系统向该子系统的中断状态端口（ＩＳＰ）写入一个中断值之后，这一位便被置1。当ＩＶ为零时，ＩＳＰ中不包含一个有效的中断值。当ＩＶ为1时，读ＩＳＰ将给出一个有效的中断标识（ＩＤ）和设备号。

位2－保留

位3－保留

位4－ＲＥＪ位。当子系统决定拒绝经由命令接口端口和引起注意端口提交的一个命令时，这一位将被置1。于是子系统退出译码状态而进入拒绝状态。在命令忙状态端口的状态（Ｓ）位将包含一个编码值，用以指出命令被拒绝的原因。当子系统接受了提交给它的命令而退出译码状态时，在命令忙状态端口的拒绝（ＲＥＪ）位和忙（Ｂ）位都被置零。

当子系统处于拒绝状态时，程序不能靠向命令端口或引起注意端口写入来提交新的命令。这些写的企图是被忽略，而这些端口继续保

持被拒绝的命令的那些值。当子系统处于拒绝态时，拒绝（REJ）位和忙（B）位都被子系统置1。

对子系统控制端口的RST位置1可以使子系统脱离拒绝状态，这时执行了一个硬件控制的子系统复位；或者可以对子系统控制端口的RR位置1，这时发出一个硬件控制的“清除拒绝状态（*Reset Reject*）”。使用RR位是清除拒绝状态的通常方法。当子系统脱离拒绝状态之后，拒绝（REJ）位和忙（B）位都被子系统置零。

位5－7－状态位。当通过引起注意端口提交的命令在译码状态被拒绝之后，便在命令忙／状态端口置入一个三位编码值。于是子系统退出译码状态并进入拒绝状态，其拒绝（REJ）位和忙（B）位等于1。除非REJ位为1，否则状态（S）的各位之值是没有定义的，因而应该由主程序忽略。

S位的编码值是：

S＝000　被保留

S＝001“得不到设备，因而拒绝”。当子系统能够确定一个设备的功能不足以执行这个命令时，便出现这一状态。对于“设备复位”和“子系统复位”命令不能产生这一状态。这一状态可以被这两个命令之一来清除。

S＝010无效命令。当子系统不能认出引起注意端口的引起注意代码，或者在命令端口提交的一个立即命令的0至7位指定的立即命令代码无效的时候，便出现这一状态。

S＝011设备忙。当设备忙于执行一个命令而不能在此时接受一个新命令时。便出现这一状态。对“设备复位”和“子系统复位”命令不能产生这一状态。

S＝１００设备拒绝执行ＳＣＢ。当程序发出“挂起”命令，从而把设备置于不允许执行任何ＳＣＢ命令的状态时，便出现这一情况。这一状态可由“恢复（Ｒｅｓｕｍｅ）”、“设备复位”或“子系统复位”命令来清除。

S＝１０１设备地址无效。当系统在引起注意端口放入一个无效的、不存在的设备号时便出现这一状态。如果引起注意端口是8位宽的，则表明位３至位０包含一个无效的设备号。对于１６位宽的引起注意端口，则是位１１至位０包含了一个无效设备号。

S＝１１０　被保留。

S＝１１１设备中断排队满。设备为那些尚未存入子系统中断状态端口的那些中断保持了供中断排队用的内部存贮器。当设备已经把这些内部存贮器全部用完时，便出现这一状态。如果处于这一状态，那么除了“设备复位”、“子系统复位”和“由清除中断”命令之外，所有其它命令都将被该设备拒绝。一旦该设备能够从它的中断排队向子系统中断状态端口写入数据，该设备的这一状态便被清除。

现在参考图５，它详细说明了命令忙／状态端口（ＣＢＳＰ）１５６的各个位是怎样产生的。

忙位（Ｂ），即ＣＢＳＰ－ＢＩＴ０－（Ｂ）（ＣＢＳＰ的第０位），是由Ｓ／Ｒ寄存器３０２与极性（ＰＨ）寄存器３００的寄存器组合起来实现的。如果忙位是零或者是无效的（*inactive*）那些由如下三个特定动作之一能把它置成１。

１．如果在线３１４上的微通道ＣＨＲＥＳＥＴ信号为１，那么在或（ＯＲ）门３１０的输出便为１，它被提供给寄存器３０２的置位（Ｓ）输入端，这样便把寄存器３０２置为１，它被加到寄存器

300的D输入端。来自反向器312的反向MC－AP－SEL信号被加到寄存器300的计时（C）输入端，于是其输出端＋CBSP－BIT0－（B），即忙位（B），被置成1。

2．如果系统程序对子程序控制端口（SCP）的位7写入一个1，其表现是在线316上的＋SCP－BIT7（RST）信号为1，“或”门310的输出变为1，于是如上述第一项动作那样，寄存器302、300以及忙位（B）都被置1。

3．如果由软件向AP“写”，则在线304上的信号＋MC－AP－SEL及线306上的信号＋SSW－REG－CLK都成为1，从而使“与”门308的输出为1。来自门308的这个1使“或”门310输出为1，于是如上述第一项和第二项动作那样，寄存器302、300以及忙位（B）都被置1。

如果忙位（B）是1，那么它可以由下述两种方式之二来清掉，即置成零。

1．由系统程序向子系统控制端口（SCP）的位5（RR）写入1，便清除了忙位（B）。这也同时清掉拒绝（REJ）位（见下述）。当SCP中的RR位为1时，线328上的信号＋PIC－SCP－SEL、线326上的信号＋IC－BUU－RCVR－BIT5信号以及线306上的＋SSW－REG－CIK信号都变为1，于是在“与／或”门318的输出端产生一个1。这个1被加到寄存器302的“复位”（R）位，它的输出变成零，这个零又加到寄存器300的D输入端。此时在线304上的信号＋MC－AP－SEL为零，这个零由反向器312变换成1。这个1被加到寄存器300的C输入端，而在D输入端此时记入的是零，因此是寄存器

300为零。当寄存器300被置零时，忙位（B），即输出信号＋CBSP－BITO－（B）变为零。

2。在命令处理器从译码状态退出而进入“接受（Accept）”状态之后，忙位（B）也由子系统清掉，子系统的局部处理器对CBSP的位。写入一个零。在线320上的信号－IC－BVS－BITO、线322上的信号＋LP－CBSP－SEL以及线324上的信号＋LPW－REG－CLK都变为1，于是在“与／或”门318的输出端产生一个1。这使寄存器302和寄存器300被置成零，与前述情况相同。当寄存器300被置零时，忙位（B）变为零。

如果系统把SSCP位7写成1，即指出子系统正在进行子系统复位，当子系统已完成子系统复位时，忙位（B）将被子系统复位。假定忙位为ON（即置为1）而且子系统先处于译码状态然后进入拒绝状态。这时将由拒绝位（CBSP位4）经过SCP中的“拒绝复位”位5以如下方式来使忙位复位：系统把1写入SCP的位5（RR）位置。一旦由子系统检测到了向SCP的写入，线326（IC－BUS－RCVR－BIT5）即变为有效，线328（MC－SCP－SEL）和线306（SSW－REG－CLK）也变为有效。这些线处于有效状态使门318开启（ON），从而使寄存器302复位为“关闭（OFF）”。寄存器302变为OFF的同时反向器312变为ON，如前所述，便使寄存器300复位为OFF，置忙位（B）为OFF。

CBSP REJ 位（4）和CBSP状态位（5，6，7）分别由寄存器330和332代表。当子系统已进入拒绝状态，CBSP

位4便被置1，而CBSP的位5，6，7（状态）被置成拒绝代码。当线334、332和324为有效时，S／R寄存器被置成1。这是由线334、332和324使“与”门336变成ON，它又置S／R寄存器330为ON，使REJ位变成ON。把状态代码置于线326、336和338，使寄存器332的状态位被置成ON。线322和324为有效时使“与”门340变成ON，从而使寄存器332的C输入端变为有效，于是状态代码被记入寄存器332。

每当“或”门346为ON时拒绝（REJ）位便被置成OFF当线326（CHRSET）为有效，或线316（RST）为有效或“与”门342为ON，都会使“或”门346为ON。如果线326和线328和线306都为有效，则“与”门342为ON。这表示系统已把“拒绝复位”位5（SCP）置成1，使子系统从拒绝状态解脱出来。

CBSP－BIT1－IV是作为S／R寄存器348的输出来实现的，每当“与”门350为ON，寄存器348便被置1。如果线322和324和352均为有效的，则“与”门350变为ON。当子系统向中断状态端口（ISP）写入之后，IV位被置1。

主处理器读出IV位，作为由子系统要求的微通道中断的结果；或者可以用IV位来登记（*poll*）要求中断的子系统。在主处理器已读出ISP而且主中断处理器已处理了中断之后，主处理器经由引起注意端口向子系统发出清除中断注意代码（1110$_b$）命令。

每当“或”门354为ON时S／R寄存器348被置成OFF如果线314（CHRESET）或线316（CORESET）或“与”门356或“与”门358为ON，则“或”门354变为

ＯＮ。

如果在线３６０、３６２、３６４和３６６上的ＩＣ－ＢＵＳ－ＲＣＶＲ－位（７－４）分别由引起注意端口位７－４置成二进位值“００００”（立即设备复位），而且线３０４和３０６为有效，则“与”门３５６为ＯＮ。

如果线３３８、３３６、３２６和３６６上的ＩＣ－ＢＵＳ－ＲＣＶＲ位（７－４）为置成2进位值“１１１０”（中断复位），而且线３０４和３０６为有效，则“与”门３５８变为ＯＮ。

ＣＢＳＰ保留（Ｒ）位由寄存器３６８代表。当线３７０和372，ＩＣ－ＢＵＳ－Bits（位）（２，３）为有效时，两个数据（Ｄ）输入端被置位。当“与”门３７４为ＯＮ时这些被记入寄存器３６８。当线３２２和３２４为ＯＮ时，“与”门３７４为ＯＮ。

命令接口端口和引起注意端口使用的格式取决于主系统向子系统提供的是立即命令还是ＳＣＢ命令。

立即命令主要是指向设位的而且是控制定向的。参照图６来看一看由１６０指出的命令接口端口的一般格式以及由１６２、１６４和１６６分别指出的在引起注意端口三种类型立即命令的代码。如果一个命令是指向一个设备的，那么设备在执行该立即命令之前要确认接收了完整的命令。为确定命令类型，要检验引起注意端口的位７至位４。如果接受的是设备复位命令型，则代码如１６２处所指出的那样如果接受了一个立即命令（命令型），则代码如１６４处指出的那样；如果接受的是中断复位（ＥＯＩ）命令型，则代码如１６６处指出的那样。

命令接口端口的位７－０指定了作代码，如１６０处所指出的那

样。当接受命令时，设备使命令忙／状态端口（图中未画出）的忙位置位。在执行子系统复位命令时，子系统必须使端口保持忙状态，直至完成了复位为止。

对于立即命令，其命令接口端口的定义如下：

ＩＣ位８－－指定立即命令格式。ＩＣ位８用于指定所使用的立即命令格式类型。“如果ＩＣ位８＝１”，则该立即命令的格式定义为类型１。

ＩＣ位３１－１６－保留（在发出命令之前由软件置成零）。“如果ＩＣ位８＝０”，则立即命令格式定义为类型２。

“如果ＩＣ位８＝０”，ＩＣ位８用于指定使用的是立即命令格式类型２。

下文中将说明按位不同的定义。

ＩＣ位３１－１６—依赖于ＯＰ代码

ＩＣ位１５－使不能命令中断。如果位１５＝０，则在完成立即命令后中断并在中断状态端口（ＩＳＰ）报告状态。如果位１５＝１，则在立即命令完成时不中断，除非出现了错误。某些命令忽略这一指示。

ＩＣ位１４—使不能设备中断。如果位１４＝０，则能使系统为指定设备中断。如果位１４＝１，则使系统能为指定设备中断。

ＩＣ位１３－９—依赖于ＯＰ代码位。这些位允许修改命令字段。当前，位１３和１２用于设备复位命令。位１２－９用于立即清除ＳＣＢ中断命令来指定要清除的中断号。

ＩＣ位７－０—立即命令代码。它们的编码如下：

位	
7 6 5 4 3 2 1 0	
0 0 0 0 0 0 0 0	—立即设备复位
0 0 0 0 0 0 0 1	—保留
0 0 0 0 0 0 1 0	—空操作（*no-op*）
0 0 0 0 0 0 1 1	—保留
0 0 0 0 0 1 0 0	—中断状态端口复位
0 0 0 0 0 1 0 1	—保留
0 0 0 0 0 1 1 0	—保留
0 0 0 0 0 1 1 1	—保留
0 0 0 0 1 0 0 0	—SCB中断清除
0 0 0 0 1 0 0 1	—保留
0 0 0 0 1 0 1 0	—保留
0 0 0 0 1 0 1 1	—保留
0 0 0 0 1 1 0 0	—取决于设备
0 0 0 0 1 1 0 1	—取决于设备
0 0 0 0 1 1 1 0	—取决于设备
0 0 0 0 1 1 1 1	—取决于设备
0 0 0 1 0 0 0 0	—保留
0 0 0 1 0 0 0 1	—保留
0 0 0 1 0 0 1 0	—运行立即诊断
0 0 0 1 0 0 1 1	—取决于设备
0 0 0 1 0 1 0 0	—保留
0 0 0 1 0 1 0 1	—保留

00010110	—保留
00010111	—取决于设备
00011000	—保留
00011001	—保留
00011010	—保留
00011011	—保留
00011100	—保留
00011101	—保留
00011110	—保留
00011111	—挂起
00100000	—恢复

在发出命令之前，要由系统软件使所有被保留位置零。位0－7的所有其它编码都是被保留的。

参照图7，来看一看对于SCB命令在168处指出的命令接口端口一般格式及在170处指出的引起注意端口的一般格式。命令接口端口包含了SCB的起始地址。这个起始地址是一个物理地址，它必须处在双字边界。为起动SCB注意请求（SCB *attention request*），引起注意端口的格式在170处指出。

如前面所说明的，间接命令包含在SCB内。多个SCB可以按指定顺序链接并被当作一个逻辑请求来处理。数据缓存区可以用一个间接表（*indirect list*）来链接。在处理一个请求时如果出现错误，这个结构还能够用于处理复杂的状态信息。状态信息放在终止状态块（TSB）中。为了能在命令链中的任何一点处理终止，在链

中为每个SCB定义了一个TSB。

对于一个SCB配置，有两个细节不同的形态。它们称作基本格式（见图8）和扩展格式（见图9）。扩展格式是由基本格式在其末端增加若干字段构成的。这些字段的简要定义及所使用的简称如下：

1．命令字

用于定义正在被传递的SCB命令类型。这个字有三个字段。

—— OP代码—包含该SCB命令的8位操作码。

—— A－ 一位指示位，指明该SCB包含的命令代码是有结构的（*architected*）还是无结构的（*unarchitected*）。所展示的格式适用于有结构的和无结构的两种命令代码。

—— 保留段

2．使能字（*enable word*）

SCB中的各字段允许用于多种由命令指定的操作。使能字包含若干字段，用以指出SCB中的这些字段对该命令的用途。它还有一些指示位，用以定义要使用的回答指示的类型。

—— CH－如果无错误则链接到下一个SCB（下一个SCB的地址是在链地址的1字段）。

—— COND CH－位1指出在由位2、3和4定义的某些条件下链接的SCB的地址在链地址2之中。

—— A1－指出这个命令使用地址1来指向一个缓存区还是指向一个间接表（它有一组指向各缓存区的指针）（具体用途由该命令规定）。

—— A2－指出这个命令使用地址2来指向一个缓存区还是指向一个间接表（它有一组指向各缓存区的指针）（具体用途由该命令规

定）。

——I—当命令完成而无差错时清除中断。

——EXT—指出该SCB有扩展格式。

——R—保留

——SES—删除异常短（Suppress Exception Short）（当读入数据少于所提供的缓存区时）。

——SEL—删除异常长（Suppress Exception Long）（当从一个设备读入的数据超过该命令所提供的缓存区大小时。

——PT—指出该命令使用一个间接表指向一个缓存区链（数据链起用命令）。

——TSB—指出只有在出错时才去存贮TSB状态。

3．使能字2

当使用扩展格式时，位于扩展段的使能字2当中提供了附加指针。这些指针是：

——PT2—指出地址2指向一个间接表（表长度由字节计数1给出）。

——DEC1—指出存贮器地址1应该被减量而不是被增量。

——DEC2—指出存贮器地址2应该被减量而不是被增量。

4．地址1

它包含32位地址，指向数据缓存区或间接表。其使用由该命令规定。一个使能字指针（PT）指出正在使用的是哪一种地址形式。

5．地址2

它包含32位地址，指向数据缓存区或间接表。其使用由该命令规定。一个使能字指针（PT2）指出正在使用的是哪一种地址形式。

6．字节计数1

包含一个四字节计数，它规定了由地址1或地址2中的地址所指示的缓存区或间接表的长度。

7．字节计数2

包含一个四字节计数，它规定了由地址2中的地址向指示的缓存区或间接表的长度。这一字段位于扩展部。

8．ＴＳＢ地址

包含一个伴随该ＳＣＢ的３２位地址。

9．链地址1

在使用命令链接的请求中，用来指向下一个ＴＣＢ。

１０．链地址2

当条件链接所要求的条件被满足时，用来指向一个ＳＣＢ。

１１．链接标识符

包含该链的１６位标识符。该值对一个链请求中的所有ＳＣＢ都是相同的。

１２．扩展长度

除了在中断状态端口１５４中的有错／无错指示外，本发明还提供了对ＳＣＢ链中的每个命令要由子系统向计算机报告的详细状态信息。这是在终止状态块（ＴＳＢ）中报告的。每个ＳＣＢ包括系统存贮器中的一个ＴＳＢ地址，子系统向该地址指示的位置写入该ＳＣＢ的完成或终止状态。ＴＳＢ的基本格式示于图１０，而其扩展格式示于图１１。当对基本格式加上扩展部分时，用于指出依赖于设备的信息长度的计数在ＴＳＢ中的位置改变了。

下面是在基本ＴＳＢ格式和扩展ＴＳＢ中使用的各字段的简要定

义：

1．结束状态字

该字包含若干个指针，它们确定了与指向该TSB的那个SCB中所包含的命令的执行情况有关的具体状态信息。

——D—指出“请求被正确完成”。

——SES—指出“短长度”—伴随SCB使能字中的SES指针。

——R—保留

——SC—指示对SCB进行规格检查。SCB无效。

——SEL—指出“长长度”—伴随SCB使能字中的SEL控制。

——HLD—指出SCB命令链被停止了。

——INT—指出逻辑中断请求。

——ASA—指出可得到结构状态。

——DSA—指出可得到取决于设备的状态（使用ASA和DSA共同指出是扩展格式）。

——DO—指出设备运行过量（over run）。

——INI—指出设备未初始化。

——ERR—指出已发生了主要错误。

——CD—指出链接方向（遵循正常链还是条件链）。

——SUS—指出对SCB挂起的命令完成了。

——ES—指出使用了扩展状态。

2．扩展结束状态

这个字有增加的状态指针。当前只有一个指针被纳入结构。

—— CT－指出命令类型（SCB或立即命令）。

3．剩余缓存区计数

这是一个4字节计数，包含由剩余缓存区地址所指示的缓存区中剩余字节数。

4．剩余缓存区地址

这是在“请求”终止时使用的缓存区的32位地址。它可能是由这最后一个SCB给出的地址1或地址2缓存区地址，也可能是由这最后一个SCB中的地址1或地址2指向的一个间接表中的一个地址，其具体意义由命令指定。

5．剩余缓存区计数2

这是一个4字节计数（在TSB扩展部中），包含由剩余缓存区地址2所指示的缓存区的剩余字节数。

6．剩余缓存区地址2

这是在“请求”终止时使用的缓存区的32位地址（在TSB扩展部中）。它可能是由该最后一个SCB给出的地址2缓存区地址，也可能是由该最后一个SCB中的地址2指向的一个间接表中的一个地址。它是在一个命令使用二个地址时使用的。

7．与设备有关的状态信息区尺度（SIZE OF DEVICE DEPENDENT AREA）

除了一组纳入结构的状态外，还采取措施提供更多的信息。这些附加信息量取决于使用的是基本格式还是扩展格式。这一字段在基本格式和扩展格式中的位置不同。附加信息是与子系统有关的。

8．与设备有关的（状态信息）数据

这一信息区包含了与设备有关的状态信息。这一信息区的最大尺

度取决于使用哪一种格式。

使用刚才描述的ＳＣＢ格式的本发明的计算系统的详细情况显示于图１２。如前面描述的那样，主系统２００与至少一个智能子系统通道总线结构（如微通道２０４）交换信息。

主系统２００中的系统程序２０６经过线２１０在系统存贮器２０８中建立各ＳＣＢ块。在下列资料中描述了系统程序的操作："ＰＳ／２　ＢＩＯＳ接口技术参考书"（ＩＢＭ出版物号６８×－２３４１－００；"ＩＢＭ　ＰＳ／２　操作系统指南"（ＩＢＭ出版物号为ＧＧ２４－３２２５－０）。如前文所述，一个ＳＣＢ格式提供了缓存区地址、链地址和ＴＳＢ地址。在分页存贮器系统中，ＳＣＢ、缓存区地址、链地址及ＴＳＢ地址都是被操作系统"钉（*pinned*）"在系统存贮器中的。就是说，有一段系统存贮器为此目的而被锁住。为便于描述，图中给出一个ＳＣＢ　２１２，它包括一个ＴＳＢ　２１４、一个指向数据缓存区２１８和２２０的间接表２１６以及指向ＳＣＢ　２２２的链地址。需要理解，根据图８－１１所示的格式，在系统存贮器２０８中可以格式化一组类似的ＳＣＢ。

子系统２０２与图２中的子系统１０２相似，其唯一差别是子系统２０２显示得更详细些，以便于描述命令的传递。子系统２０２包括一个总线接口和控制单元２２８，它有一个命令传递逻辑２３０和总线控制器（*bus master*）２３２；子系统２０２还包括一个带有中断（Ｉ）逻辑２３６的本机（*local*）微处理器２３４本机存贮器２３８、以及设备接口和控制逻辑２４０，它与一组设备２４２、２４４、２４６及２４８相连。

下面先参照图１２来概括描述命令处理系统，它包括命令处理器和译码过程。然后再参照图１９－２５以流程图形式详细描述其功能。

系统程序读出命令忙／状态端口（图中未画）以确认引起注意端口和命令接口端口空闲。系统程序２０６通过命令接口端口（未画出）向总线接口与控制单元２２８的命令传递逻辑２３０提供一个ＳＣＢ的地址，通过线２５０向引起注意端口（未画出）提供命令类型和要执行命令的设备。

作为译码处理的一部分，该命令要被进行有效性检验进而被接受或被拒绝。假定该命令被接受了，则子系统把该状态写入命令忙／状态端口（图中未画出），并指向要执行的命令。

子系统利用总线控制器２３２中的总线控制功能启动微通道204上的总线周期。此时系统存贮器是一个受控微通道存贮器（*Micro Channel memory slave*），其微通道控制和地址线均由总线控制器２３２来驱动。系统存贮器驱动微通道数据总线，而总线控制器功能把ＳＣＢ（一个或多个）传送给子系统的本机存贮器２３８。一旦子系统接受了一个ＳＣＢ，在微处理器中运行的程序便对命令译码并把它指向指令的设备。这个程序操作将参考图１４－２０中的流程图来详细解释。为完成传送，要把缓存区地址装入总线控制器２３２。

具体地说，一旦ＳＣＢ地址被译码，该地址便被提供给系统存贮器２０８，而ＳＣＢ被取到子系统２０２。例如，如果ＳＣＢ２１２要被寻址，则总线控制器２３２以ＳＣＢ地址来启动地址线２５２，同时通过线２５４将ＳＣＢ的第一部分返回。在对地址增量之后再重复这一过程，直到整个ＳＣＢ被取完为止。对要处理的数据的访问

（*access*）是通过地址线256和258实现的，它们指向由间接表216指定的数据缓存区218和220。数据经线260和262从缓存区218和220返回。然后，总线控制器232按照引起注意端口（未画出）中的命令第二部分的规定，经接口240将数据传送给本机存贮器238或者设备之一。SCB的完成或终止状态经线264和266存贮于TSB214中，其状态数据经线268和270提供给TSB214。

SCB212指向SCB222，它经由线272寻址，而SCB222经线274返回信息。来自缓存区226的要处理的数据经由地址线256来访问，经线262将数据返回总线控制器232。然后总线控制器232经接口240（如前所述）将数据传送给本机存贮器238或设备之一。SCB222的完成或终止状态经由线264寻址存贮于TSB224，其状态数据经线268提供给TSB224。

子系统202还把SCB或SCB链的中断状态装入中断状态端口（未画出）。它经由线264向主系统提供了一个物理中断返回（*back*）。

在给出命令处理系统的流程图描述之前，先作一般性描述，以便使图14－20的流程图中使用的术语更易于理解。

提交命令方法与命令类型

本发明规定了两种向子系统传递命令的方法。某些子系统范围内的硬件规定功能可由硬件控制来调用，这个硬件控制可借助子系统控制端口中的程序修改位来访问。硬件控制提供了改变子系统状态的一种快速高优先级方式，以这种方式提出的子系统复位命令的优先级超

过把数据写入命令端口和引起注意端口而提交给子系统的所有其它命令。

利用发给子系统控制端口的OUT（输出）命令和修改某些规定位，可启动硬件控制。

对于RR、DMA以及EI位的修改不能由某些具体命令来说明但这些位对子系统也具有子系统范围内的效用它们将在下文中的“主处理器试图设置I／O端口的结果”一节以及附表1中的表中概要地描述。

通过写命令端口和引起注意端口来提交的命令归于两大类：间接命令（以SCB为基础）和直接命令。由引起注意请求代码3来区分以SCB为基础的命令。在SCB命令情况下，命令接口端口包含SCB起点的实际存贮器地址。直接命令有三种类型，由唯一的引起注意代码设置来决定。所有的直接命令在命令端口中有一个32位的值，它与引起注意代码一起定义了要由子系统完成的动作。提供了两个引起注意代码0（复位）和E（中断复位），以便能在具有引起注意代码“01h”的其他直接命令之上来对这些命令进行特殊情况处理。有时把直接命令用术语“立即命令”来描述，以指出命令端口中包含的32位值完全定义了必须完成的操作。这与SCB命令相反，SCB命令必须从命令端口给出的系统存贮器地址中取出来，以确定要完成的动作。

在体系结构中定义了一个为1的引起注意代码F（与设备有关的注意（*device dependent attention*）。提供引起注意代码F是为了使一个实际应用能为希望最优化的命令提供特殊情况处理。

提交命令过程中的子系统状态转换。

这一节中定义使用硬件控制或通过引起注意端口提交一个命令值所发生的那些状态变换与动作。在图13中的状态转换图里提供了这些资料。当子系统处于译码状态时，它对所提交命令的接受或拒绝将在下文中“命令处理与译码过程”一节中详细讨论，它对译码过程给出了更多的细节，并在附表2，3中提供了一套决策表，说明命令是这样被译码过程接受或拒绝的。

下述状态是为提交命令转换中的应用而定义的，关于子系统状态和译码过程的更多细节可参见有关“命令处理与译码过程的讨论。

子系统状态：

1。“接受命令”或ACCEPT。

子系统处于能够接受经命令端口和引起注意端口提交的命令的状态，在这一状态，在命令忙／状态端口中的拒绝位和忙位双双为零。

主处理器的引起注意端口写入将使子系统脱离这一状态而进入“进行译码”状态，在这一状态将决定该命令将是否被接受。

2．“子系统复位未决开始”或PENDS。

子系统控制端口的RST位已被主程序置1，而子系统已开始了对它的微处理器进行硬件控制的复位。在这一状态期间，主处理器对命令端口和引起注意端口的写人被忽略。子系统控制端口的由主系统写入的各位被忽略，而RST位不被忽略。当完成了硬件复位，子系统进入“子系统复位过程完成”状态。

当在这一状态时，命令忙／状态端口的忙（B）位为1，拒绝位（R）为零。

3．“子系统复位未决完成”或PENDC，此时子系统已经完

成了它的微处理器的硬件控制复位，正在等待主处理器向子系统控制端口的RST位写入一个零位。在这一状态时主处理器向命令端口和引起注意端口的写入均被忽略。但允许主处理器向子系统控制端口写。向子系统控制端口的RST位写入零将使子系统进入“子系统复位正在进行”状态。

在这一状态，命令忙／状态端口的忙位（B）为1，而拒绝位（R）为零。

某种实施方案中也许不能区分“复位未决开始”与“复位未决完成”两个状态的差别，因而可能只有一个“复位未决”状态。在这种情况下，如果对RST位置零将把子系统从未决状态转换到“复位正在进行”状态。

4．“子系统复位正在进行”或RESETS

子系统控制端口的RST位已经置零，在“子系统复位未决开始”状态之后经过了一段的具体实际系统所决定的延迟时间，便进入这一状态。一旦进入这一状态，子系统便完成了它对子系统的复位。当子系统处于这一状态时，主处理器被允许去修改子系统控制端口的内容。主处理器向命令端口和引起注意端口的写入均被忽略。

在这一状态的命令忙／状态端口的忙位是1，而拒绝位是零。

一旦子系统复位完成了，子系统便进入“接受命令”状态。这时命令忙／状态端口的忙位和拒绝位均是零、这时IV位置1，以指出ISP中包含有完成的子系统复位的中断值。

5．“子系统拒绝”或REJECT

经过命令端口和引起注意端口提交的命令以被译码过程拒绝。在拒绝状态被清除之前，不会通过命令端口和引起注意端口再接收任何

命令。当在这一状态时，主处理器向命令端口与引起注意端口的写入被忽略。而主处理器向子系统控制端口写入是允许的。命令忙／状态端口将使拒绝位和忙位都置成1。在命令忙／状态端口的S位将给出命令被译码过程拒绝的理由。

当子系统控制端口的RR位被置1，或者由硬件控制的子系统复位来置子系统控制端口的RSP位为1，可以使子系统退出拒绝状态。

6．“译码在进行”或DECODE

当检测到主处理器向引起注意端口写入时，子系统便从接受命令状态进入这一状态。在这一状态子系统决定它将是否接收该命令。

当子系统进入这一状态，命令忙／状态端口的忙位便被置1，而拒绝位被置零。当在这一状态时主处理器的命令端口和引起注意端口的写入被忽略。主系统向子系统控制端口的写入除了设变RST位之外也被忽略。

如果译码过程接受了该命令，子系统进入“接受命令”状态。如果命令未被接受执行，则子系统进入“拒绝”状态。

各个动作是：

a．主处理器向子系统控制端口RST位写入1，或者说RST＝1。

主处理器向子系统控制端口发出“OUT（输出）”命令，使RST位之值置1。

b．主处理器向子系统控制端口RST位写入零，或者说RST＝0。

主处理器向子系统控制端口发出“OVT”命令，使RST位之值置零。

c．主处理器向引起注意端口写，或者说W／A P。

主处理器以一个O U T 命令向引起注意端口写。

d．主处理器对子系统控制端口 R R 位写入1，或者说R R＝1。

主处理器向子系统控制端口说出“O U T ”命令，使R R位之值置1。

e．“复位未决完成”，或P E N D　C O M P L E T E。

子系统已经完成了对它的微处理器复位动作以及允许它等待完成子系统复位所需要的任何其它动作。

f。“子系统复位完成”或R E S E T　C O M P L E T E。

子系统已经完成了子系统复位动作。

g．“译码接受命令”或D E C O D E　A C C E P T。

子系统译码过程已经接受了通过命令端口和引起注意端口提交的命令。

h．“译码拒绝命令”或D E C O D E　R E J E C T。

子系统译码过程已经拒绝了通过命令端口和引起注意端口提交的命令。

在图１３的状态图中以这些状态和动作定义了各状态的转换。

主处理器试图置Ｉ／O端口的结果

在附表1的决策表中，概括了子系统处于不同状态时主处理器试图置Ｉ／O端口的结果。各状态是在图１２中的状态图里所使用的那些，其中S S C P的意义是“子系统控制端口”。

对附表1中表的注释所作的解释：

1．系统把R S P置1，使子系统处于“复位未决开始”状态。

2．如果RST被置1，则在子系统控制端口中的所有其他位均被子系统置1，这是进入“复位未决开始”状态的一部分。

3．当子系统处于“接受命令”状态时，系统对引起注意端口写把子系统带入“译码”状态。

4．当子系统处在“复位未决完成”状态时，系统对RST置零把子系统放入“复位正在进行”状态。

5．当子系统处于“拒绝”状态时，若系统置RR为1则把子系统置于“接受命令”状态，除非置RR的输出（OUT）命令也置RST为1。如果RST被置1，则RR的改变被忽略，于是子系统被置于“复位未决开始”状态。当子系统处于“拒绝”状态时，RR被置零将不会清除“拒绝”状态。

命令处理和译码处理过程

这一节中关于命令处理部分的讨论集中于译码过程。与译码过程中的接受命令有关的决策表示于表1、与译码过程中的拒绝命令有关的决策表示于表2、3和4中。

需要存在一个硬件控制的处理，子系统用它来监视主处理器对命令，引起注意及子系统控制三个端口的写入。这个处理称作命令处理器，它对写子系统控制端口的RST位这一类动作的反应是调用硬件控制来对子系统复位。命令处理器还负责调用一个在硬件或固件（*firmware*）中的一个处理过程，子系统用它来确定是否接受一个经引起注意和命令端口提交的命令。当子系统处于“接受命令”状态并检测到一个主处理器对引起注意端口的写入时，命令处理器便调用这一处理过程。命令处理器把子系统放在“译码”状态，并调用“译码处理过程”来决定它是否能接受这个新提交的命令。

当处于译码状态时，命令处理器还负责保证主处理器通过写命令端口或引起注意端口来试图提交的新命令被锁存起来。命令处理器还响应主程序对子系统控制器的写入，保证子系统在任何状态都允许由主程序来改变RST位。正是这一功能使来自主处理器的硬件控制的“子系统复位”命令能被接受。根据子系统的状态，命令处理器还允许进行其他硬件控制。

由于译码处理过程有确定子系统是否能接受一个指向设备的命令的任务，需要注意下列各类数据：

*命令类型，命令的引起注意代码决定了一个命令将被如何处理。命令被分成如下各类：设备复位、中断复位、SCB、立即命令。如果一个实际应用系统中支持“依赖设备的注意（Device Dependent Attention）”，它将根据该实际应用的规定，列入SCB类或立即命令类。不被支持的引起注意代码被译码处理过程所拒绝，这些不是SCB命令的无效命令注意代码被称作直接命令。

*设备号。命令指向的设备包含在引起注意端口。为了确定命令指向哪一个设备，故需要这个数据。译码处理过程还利用这个数据来设备状态信息，它使子系统确定这个新命令是否能被接受或拒绝。无效设备地址会使命令被拒绝。

*命令代码，当处理直接命令时，该命令的位0－7中的8位命令代码被检验，以确认命令代码。不能辨认的8位立即命令代码会使命令作为无效命令被拒绝。

*设备状态，这是设备的当前状态，它规定了在给定的当前设备状态下该命令能否被接受。本发明定义了一组状态，并指出了在一个给定状态下一个命令是否可以被接受。这一数据将在下文中的“译码

处理过程决策表”中详细陈述。

当译码处理过程决定接受一个命令时，它需要保证设备状态被更新以表明接受了新命令，还要使该设备定时开始执行新命令。译码处理过程还必须保证命令处理器把子系统从译码状态脱出并进入“接受命令”状态。本发明并不坚持要求设备状态复位及新命令定时要在译码处理过程中完成，但认为这种情况在概念上较为简单。

译码过程必须使所要求的新命令被安排好预定时间以便在所要求的设备上执行，认识到这一点也很重要。“安排预定时间（*scheduled*）”这个词用于“设备0复位”以外的所有其它命令，因为子系统处理器周期的使用需要在其它设备正在完成的各任务之间分。在“设备零复位”的情况下，这是一个特殊情况，因为这一动作产生一个由软件调用的子系统复位，并要求子系统当前为其它设备安排的所有活动全部停止。

当一个命令被告知已经接受时，是命令处理器把命令忙／状态端口置位，从而使忙位和拒绝位都被置0。

当译码过程拒绝一个命令时，命令处理器便得到拒绝的理由。子系统从译码状态过渡到拒绝状态。在命令忙／状态端口的忙位和拒绝位被置1。在拒绝位被置1的同时，该端口的S位被设置，使其包含的代码值指出命令被拒绝的原因。

在完成译码处理过程时，若干目标得到了满足：

*被译码处理过程接受的立即命令将不会在以后要求拒绝命令中断。除非有硬件故障，否则该立即命令由译码处理过程接受之后将正确地执行。

*任何由于当前设备状态的原因而不能被执行的命令都尽可能快

地被拒绝而不引起子系统中断。拒绝状态和拒绝理由被保持下来共主程序使用，直至明确地清除掉时为止。

*“设备复位”之类的特殊引起注意代码尽可能快地被安排执行。

*ＳＣＢ命令被安排时间取出。译码处理过程不检查ＳＣＢ中字段值的有效性。

对于当各种命令呈交给一个设备时该设备如何作出反应的描述都定义在“译码处理过程决策表”中的一组决策表里，对它将作简要讨论。作为总概括，这些规则可描述如下：

1．硬件控制的子系统复位总会被子系统接受。

2．子系统处于下列状态时，指向引起注意端口的命令将被忽略：译码在进行、子系统拒绝、子系统复位在进行、子系统复位未决开始、或子系统复位未决完成。

3．即使一个设备正忙于执行其它命令，该设备也将接受“设备复位”命令。对这条规则有三个例外。

——在引起注意端口指定的设备号无效。

——在位０－７中的８位立即命令代码不等于零。

一个系统处于上述第二款所规定的状态之一，就是说，它忽略企试图写入引起注意端口所提交的任何命令。

4．当一个设备正忙于执行一个命令，它将以设备忙代码拒绝一个新命令，但下列情况例外：

——新命令是“设备复位”。

——设备在忙于执行一个ＳＣＢ，而该新命令是“挂起”的。(这种情况下，在设备被置于“以不能执行ＳＣＢ”状态之前允许完成正在执行的ＳＣＢ。一个“挂起”总是使该设备中被执行的最后一个

SCB请求中断。）

——设备在忙于执行一个SCB，而新命令是“SCB中断复位”。（当前执行的SCB立即停止，转而执行SCB中断复位功能。被停止的SCB允许继续执行。）

5．当设备的中断排队满时，它将拒绝一个新命令，除非有下列情况：

——设备复位

——中断复位（SCB或“立即的”）

——子系统复位（硬件或软件控制的）

6．当一个设备没有忙于执行一个命令时，如果该设备已经被置于“因先前挂起的命令而不能执行SCB”的状态时，它将拒绝一个SCB命令。

7．如果设备处于不能可靠地执行命令的状态时，它将拒绝所有命令，除非是：

——设备复位

——子系统复位（硬件或软件控制）

——如果在引起注意端口的设备地址无效，该命令被拒绝。

译码处理过程决策表

译码处理过程的结构规定了进行拒绝检验的顺序，因此当存在多种拒绝理由时，它会指出要给定的是那一个拒绝理由。这一点在下表中给出：

译码处理过程步骤

步骤1—检查引起注意代码和命令的有效性。拒绝代码＝010

拒绝原因

不可识别的注意代码　　无效注意代码

引起注意复位代码（00）　　无效8位立即命令代码

中断注意复位代码（0E）　　无效8位立即命令代码

立即命令注意代码（01）　　无效8位立即命令代码

步骤2—检查“引起注意端口给出的设备有效”的有效性。拒绝代码＝101

拒绝理由

设备复位　　在引起注意端口口设备号无效

SCB或与设备有关的注意码作为SCB　　在引起注意端口的设备号无效

立即挂起、恢复空操作（NOOP）　　在引起注意端口的设备号无效

立即诊断　　在引起注意端口的设备号无效

中断状态端口复位　　在引起注意端口的设备号无效

SCB中断复位　　在引起注意端口的设备号无效

与设备有关的注意码作为非SCB　　在引起注意端口的设备号无效

立即命令　　在引起注意端口的设备号无效

步骤3—检验“设备处于可得到状态”的有效性拒绝代码＝001

拒绝理由

SCB或与设备有关的注意码作为SCB　　设备处于不可得到状态

立即、挂起、恢复空操作　　设备处于不可得到状态

立即诊断	设备处于不可得到状态
ＳＣＢ中断复位	设备处于不可得到状态
中断复位	设备处于不可得到状态
状态端口	设备处于不可得到状态
与设备有关的注意码作为非ＳＣＢ	设备处于不可得到状态
立即命令（上述任何一个）	设备处于不可得到状态

步骤４—检验“设备非忙”的有效性

拒绝代码＝０１１

拒绝原因

ＳＣＢ或与设备有关的注意码作为ＳＣＢ	设备忙
立即、恢复、空操作	设备忙
立即诊断	
中断状态端口复位	设备忙
设备中断复位、挂起	设备忙，（没有在执行ＳＣＢ）
与设备有关的注意码作为非ＳＣＢ	设备忙
立即命令（非上述任何一个）	设备忙

步骤５—检查“设备中断排队未满”的有效性

拒绝代码＝１１１

拒绝理由

SCB或与设备有关的　　中断排队满
注意码作为SCB
恢复、挂起、立即空操　　中断排队满
作
立即诊断　　中断排队满
与设备有关的注意作为　　中断排队满
非SCB
立即命令（非上述任何　　中断排队满
一个）
步骤6—检查“使能执行SCB”的有效性
拒绝代码＝100
拒绝理由
SCB或与设备有关的　　由挂起使不能执行SCB
注意码作为SCB

首先由于规定了详细的设备和子系统状态，然后又有一套决策表从而使译码结构体系被完全地确定了。

子系统及设备状态定义：

子系统有如下状态组分：

*子系统执行状态有：ACCEPT，PENDS，PENDC，RESETS，REJECT，DECODE。这些状态确定了该子系统是否能接受一个新命令或硬件控制。

*“可得到”状态，确定该子系统是否已进入了可以预料它能可靠地执行命令的状态。如果它的值是YES（是），该子系统是可得到的，若为NO（否），则该子系统为不能得到的。

*使能DMA。这一状态指出该子系统是否允许DMA活动，为YES或NO。当不允许DMA活动时；对于该子系统便不允许取出SCB命令、存贮TSB状态，或按SCB命令传送数据。这一状态相应于子系统控制端口中的DMA位。

*使能中断。这一状态指出该子系统能否在一个命令要求中断时向主处理器发出一个物理中断。它相应于子系统控制端口中的EI位当其值为YES时，可以发出物理中断信号。当其值为NO时，则不能发出物理中断信号。

*IV有效。这一状态相应于命令忙／状态端口中的IV位。如果其值为YES，则该子系统的中断状态端口在其中有一个有效中断值，它尚未被主程序清除。如果其值为NO，则中断状态端口清除了任何中断值。

*任何设备有效（*Any Device Active*）。这一状态分量由子系统用于确定在它已经处理了一个来自主处理器的请求之后如何恢复处理过程。这个主处理器请求或者是要处理一个硬件控制或者要处理一个通过引起注意端口提交的命令。例如，当子系统处于ACCEPT状态时写入RR位，而该状态分量为YES，则意味着子系统应该返回到在RR被置位之前所进行的访问。如果这一状态分量之值为NO，则子系统没有要完成的设备动作，于是它进入等待下一个主处理器动作的发生。

设备状态

设备被规定有如下状态分量

1．忙状态。这一状态确定了设备当前是否忙于执行一个命令。这个状态有两个值：忙（*busy*）和闭（*idle*）。如果值为“闭”，

表明该设备当前没有执行任何命令。如果其值为“忙”，则表明设备正在执行一个命令，而“忙”状态的子状态及CMD状态将进一步确定在该设备上正在进行的命令或硬件控制。

2。忙状态的子状态（*Busy Substates*），如果一个设备正忙于执行一个命令，则这个状态分量指出正在进行的命令的类型：如下立即命令，SCB、子系统复位（硬件控制的）、子系统复位（软件控制的）、设备复位、挂起、以及SCB中断复位。

当“忙状态的子状态”为立即命令或SCB时，则CMD段确定了在设备中将要执行的显式命令。当“忙状态的子状态”是子系统复位（硬设备控制的）、子系统复位（软件控制的）、设备复位、挂起或SCB中断复位时，则CMD状态分量指出这些命令之一被接受的时候在该设备中正在进行的那个命令。

当“忙状态的子状态”是子系统复位（硬件控制的）、子系统复位（软件控制的）、设备复位、挂起、或SCB中断复位时，设备状态分量“旧有的（*old*）忙状态子状态”包含有这些命令之一被接受时“忙状态的子状态”的行为。

当设备的忙状态是“闭”时，则对该设备不确定“忙状态的子状态”。

3．CMD状态，它给出设备中正在执行的命令的详细情况如下如果设备的“忙状态的子状态”是立即命令，则CMD状态给出在该设备中正在执行的命令。这时，它的值是：

立即运行诊断测试

立即空操作

中断状态端口复位

立即命令

如果“SCB忙子状态”，那么CMD状态是：

从主处理器存贮器中取下一个SCB，作为单个SCB。

从主处理器存贮器中取下一个SCB作为SCB链中的元素。

检验单个SCB的有效性（完成SCB的规格测试）。

检验SCB链元素的有效性（完成SCB的规格测试）。

执行单个SCB，

执行SCB链元素

这些CMD状态是在设备执行一个SCB时由设备任务来设置的就是说，命令状态从取SCB起持续到执行单个SCB或执行SCB链元素。当一个单个SCB在执行时，一旦SCB完成了4行，设备便进入“闭”状态。当设备执行一个SCB链时，从取出和执行SCB命令状态直到该链全部完成或SCB执行被“挂起”命令挂起为止，命令状态一直保持。“挂起”命令总是在完成“挂起”动作后迫使设备进入“闭”状态。

如果该设备的“忙状态的子状态”等于子系统复位（硬件控制的）、子系统复位（软件控制的）、设备复位、挂起、或SCB中断复位，CMD段的使用依赖于状态分量“旧有的忙状态子状态”。如果其值为立即命令或SCB，那么CMD段反映的是当子系统复位、设备复位、挂起或SCB中断复位终接受执行的时候曾经在动作的那个命令如果该设备的旧有的忙状态子状态是子系统复位（硬件或软件控制的）设备复位、挂起、或SCB中渐复位，则CMD被忽略，因为在接受时动作的命令完全由状态分量“旧有的忙状态子状态”所确定了。

4．“设备中断排队端”或“中断排队满”。这个状态分量指出该设备的中断排队空间是否满了。若值为“是（ＹＥＳ）”，则表明当前排队满，若值为“否（ＮＯ）”，则表明设备中断排队还有空间可供使用。当一个中断被设备排队时，被分配中断排队空间，而在这一点上中断排队可能变成满的了。

5．“使能执行ＳＣＢ”，或“ＳＣＢ Ｅ*xec* Ｅ*nable*”。这一状态分量指出该设备是否允许执行ＳＣＢ。如果已经不允许执行ＳＣＢ，这个分量为“否”。这是由执行“挂起”命令引起的。如果允许执行ＳＣＢ，这个分量之值为“是”。这个“是”是由恢复、设备或子系统复位命令设置的。

6．“ＳＣＢ链挂起”或“链挂起”，只有当对设备发出“挂起”命令时，这一状态的值才有效。如果发出了一个“挂起”命令，而且对链中的下一个ＳＣＢ的提取被“挂起”命令停止。则这一状态分量的值为“是（ＹＥＳ）”。这意味着在这一状态下向设备发出“恢复（Ｒ*esume*）”命令将使该设备重新进入ＳＣＢ忙子状态，并且利用挂起命令时在“ＳＣＢ取地址”中保存的ＳＣＢ地址来恢复执行被挂起的链。如果在一个ＳＣＢ链未曾活动（*active*）时发生了“挂起”命令，或者如果该链被当前ＳＣＢ的完成所停止，则这一状态分量之值将为“否（ＮＯ）”。在这种状态下向设备发出“恢复”命令将不会使设备重新进入ＳＣＢ忙子状态，它将退出“恢复”而进入“闭”状态。

7．ＳＣＢ取地址。当一个ＳＣＢ命令是首次被一个设备接受时在命令端口的地址被译码处理过程送到这一个区。当链中的下一个ＳＣＢ要被设备任务取出时，这一状态分量便被更新为那个ＳＣＢ的

地址。设备任务根据ＳＣＢ的结束状态来决定从ＳＣＢ中选择哪一个链地址。

8．“设备可得到”或“*Device Avail*”。如果设备处于能够可靠地执行命令的状态，则这个状态分量有值“是”。如果设备处于不能可靠地执行命令的状态，则这一状态值为“否”。

9．“立即命令值”或“ＩＣ”。当译码处理过程接受一个直接命令时，在命令端口的值便被传送到这一状态分量。它能被设备任务用于取直接命令中的值。当前只有位１５－０影响直接命令的执行。

１０．“使能设备中断”，这一状态分量确定一个设备是否能要求一个物理中断。如果这一分量的值为“使能（*enable*）”，则一个中断能够被请求。如果值为“使不能（*disable*）”，则不能有物理中断被请求。一个设备总是被要求把一个中断排列设备内部中断排队之中，而与这一分量之值无关。这一分量或者由发给该设备的立即命令的位１４设置，或者由子系统复位命令设置。

１１．“后中断（*intenupt after*”。这一状态分量确定一个立即命令在成功地完成之后是否将要求一个中断。对于子系统复位、ＳＣＢ中断复位、及立即中断复位，这一状态被忽略。这一状态分量有“是”和“否”两个值，由指向该设备的立即命令的位１５来设置。

１２．“旧有忙子系统（*old Busy Substate*）”这一状态分量包含着当接受了“挂起”、子系统复位（软件控制的）设备复位、子系统复位（硬件控制的）、或ＳＣＢ中断复位时曾经作用的“忙子状态”值。

１３．旧有的ＣＭＤ状态。这一状态分量包含的内容是：当因为

设备中断排队已满而不能取出一个SCB时曾经有效的SCB CMD值。当产生这一状态时设备被置于“闭”状态。SCB中断复位或立即中断复位能使该设备重新进入SCB状态，如果在执行这些之后中断排队不再满的话。

14．“中断排队满使SCB挂起”。如果因为设备中断排队已满而把该设备从SCB状态脱出而置于“闭”状态，则这一状态分量有值“是”。当中断复位使设备中断排队退出这一状态等于“是”时的排队状态时，状态分量“旧的CMD状态”和“SCB取地址”被用于重新启动SCB的执行。当设备不处于因中断排队满而使SCB执行被“挂起”的状态时，这一分量之值为“否”。

15．“在执行SCB的接受SCB中断复位”。当一个设备正忙于执行一个SCB时接受了SCC中断复位命令，则这一状态分量等于“是”。当该设备处于“闭”状态时接受了SCB中断复位，则该值等于“否”。

16．设备中断排队，这一状态分量为设备的内部中断排队分化空间。它得要包括当前最老的中断元素的位置以及为其后的中断所能提供的条目的位置和数量。

17．SCB逻辑中断计数。每当设备产生一个SCB逻辑中断这一状态分量便增1。

利用这些状态分量中的一些，译码处理过程按附表2、3、4、5所示决策表来接受或拒绝一个命令。在这些表中，符号DC意思是“不必管它（Don't Care）”，于是这个状态不被测试来确定译码处理过程的输出。符号NA意思是“不能应用的（Not Applicable）”，意思是说这一状态不能作为译码处理过程所需

要测试的一部分而存在。

命令传递系统包括一个命令处理器和一个译码处理器。这些功能可以在硬件和／或软件中实现。因此，它们的实施以流程图形式示于图１４—２０，作为前文中用于解释系统操作的图２和图１２所示框图的补充。特别是，在图１２中，命令处理器和译码处理器是以硬件形式作为举例包含在子系统２０２的命令传递逻辑２３０和微处理器２３４中。命令处理器处理在Ｉ／Ｏ端口接受的命令，而译码处理器对命令译码以供子系统或其附加设备之一去执行。

图１４是命令传递系统的总流程图。当子系统检测到主处理器向下列任何一个Ｉ／Ｏ地址的写入时，便以中断方式进入命令处理器。这些Ｉ／Ｏ地址是：子系统控制端口（ＳＳＣＰ），引起注意端口（ＡＰ），或命令接口端口（ＣＰ），（如块５００所示）。如块５０２所指出的，如果命令处理器能这样做了，它便使其处理主处理器对命令接口端口和引起注意端口的写入过程不能受任何中断。如果这类中断不能被阻止，那么对这些端口的写入便被忽略，直至其后命令处理器重新进入它的处理过程。在决策块５０４，确定进入命令处理器是否是试图写ＳＳＣＰ的结果。如果是，则此逻辑继续前进到块５０６去处理对ＳＳＣＰ的写入。这一过程参考图１５和图１６详细描述。在写完ＳＳＣＰ之后，该逻辑进入块５０８，在那里命令处理器重新使能处理主处理器对引起注意端口和命令接口端口的写入，然后循环返回块５００。

如果在决策块５０４发现不是对ＳＳＣＰ写入，则进入决策块５１０以确定主计算机是否是试图写命令接口端口。如果是这样，则该逻辑进入块５１２去处理对命令接口端口的写入。这一过程参照图

18详细描述。在写入命令接口端口之后，该逻辑再次进入块508在那里命令处理器重新使能处理主处理器对引起注意端口的写入，然后循环返回块500。

如果决策块500处发现不是写命令接口端口，便进入块514去处理对引起注意端口的写入。这一过程参照图17详细描述。

下面对照图15和图16来详细描述图14中的块506。对照图15，主处理器写入SSCP的值在块516被读入。在块518确定是否是试图对子系统复位。如果是，则进入块520，在那里对SSCP的RST位置1。再进入决策块522，在那里确定子系统执行状态是否接受命令状态。如果是，则子系统状态被置于“子系统复位未决开始”状态，如块524所指出的那样。然后，如块526所指出的那样，调用子系统硬件功能来执行子系统复位未决。当执行子系统复位未决功能的硬件功能完成时，子系统执行状态被置于“子系统未决完成”状态，如块528所示。然后返回图14中的块508使能主处理器中断，如前面所解释的那样。

如果在决策块522确定该子系统执行状态不是接受命令状态，则进入决策块530以确定子系统执行状态是否是“子系统复位未决开始”。如果是，则进行块532、重新使主处理器能够写入命令接口端口和引起注意端口。然后命令处理器把控制转向硬件处理过程，执行复位未决功能，如块535所示。

如果在决策块530确定了子系统执行状态不是“子系统复位未决开始”，则进行决策块536，以确定子系统执行状态是否是“子系统复位未决完成”。如果是，则返回图14中的块508，使能主处理器中断，如前面描述的那样。如果不是，则进入决策块538以

确定子系统执行状态是否是“拒绝”状态。如果是，进入块５２４，把子系统放在“复位未决开始”状态，然后继续前面描述过的过程。如果不是，则进行决策块５４０以确定子系统执行状态是否等于“译码”，如果是，进入块５４２。在那里，当主计算机把ＲＳＴ置１的时候正在进行的译码处理过程的执行被放弃了。对ＲＳＴ置１使子系统放弃了“复位在进行”状态，如果５４６所示，继续进行块５２４把子系统放入“复位未决开始”状态，如前已解释那样。

如果在决策块５７８确定了不曾试图使子系统复位，则进入块５４８，那里使ＳＳＣＰ的ＲＳＴ位置０。继续进入决策块５５０，在那里确定子系统的执行状态是不是“译码”。如果是，则进入块５５２，在那里命令处理器重新使能处理主处理器写命令端口和引起注意端口。然后命令处理器把控制转向译码处理过程，如块５５４所示。

如果在决策块５５０确定子系统执行状态不是“译码”，则进入决策块５５６，以确定子系统执行状态是否是“子系统复位未决开始”如果是，则进入块５５８去迫使所有ＳＳＣＰ位除ＲＳＴ位外均置零然后进入块５３２使能写入命令接口端口和引起注意端口，再继续如前面解释的过程。

如果在决策块５５６所作的判定是子系统执行状态不是“子系统复位未决开始”，则继续进入决策块５６０，在那里确定子系统决策状态是否是“复位未决结束”，如果是，则进入块５６２，在那里子系统执行状态被设置为“子系统复位在进行”状态。在块５６４，命令处理器重新使能处理主处理器写命令接口端口和引起注意端口，然后命令处理器调用“子系统复位在进行”功能，如块５６６所示。在

此时，子系统执行状态是“准备好接受命令”，如块５６８所示，然后返回图１４中的块５０８使能主处理器中断，如前面解释的那样。

如果在决策块５６０确定子系统执行状态不是“复位未决完成”则进入块５７０，在那里子系统执行为“拒绝”或“接受”。在块５７２，ＳＳＣＰ的位0被置成ＳＳＣＰ的ＥＩ位。子系统执行状态的中断使能分量就置成“是（ＹＥＳ）”，如块５７４所示，在块５７６，检测中断功能被调用来确定是否有任何设备（从设备0开始）能发生中断。ＳＳＣＰ的位１被置入子系统控制端口的ＤＭＡ，如块５７８所示。ＳＳＣＰ的位6、４、３和２被置入ＳＳＣＰ中的相应位置，如块５８０所示。ＳＳＣＰ的位5被置入ＳＳＣＰ的ＲＲ位，如块５８２所示。

写ＳＳＣＰ过程在图１６中在决策块５８４处继续，在那里确定ＳＳＣＰ中的ＲＲ位是否等于零。如果是，则进入决策块５８６，在那里确定子系统执行状态是否为“接受命令”。如果是，则进入决策块５８８以确定子系统是否有任何“设备在活动（*active*）”分量等于“否（ＮＯ）”。如果是，则进入图１４中的块５０８使能主处理器中断，如前文解释的那样。如果任何“设备在活动”分量等于“是（ＹＥＳ）”，则进入块５９０，在那里命令处理器重新使能处理器写命令接口端口和引起注意端口。在块５９２，命令处理器从处理它的活动返回，把控制转向子系统的被中断了的活动。

如果在决策块５８６确定子系统执行状态不是“接受命令”，则进入决策块５９４，在那里确定子系统执进状态是否是“系统复位在进行”状态。如果是，则进入块５９６，在那里命令处理器重新使能处理主处理器写命令接口端口和引起注意端口。然后命令处理器从

“中断”转向块５９８所指出的“子系统复位在进行”功能。

如果在决策块５９４确定子系统执行状态不是“复位在进行”，则进入决策块５８８，进面继续前面解释过的过程。

如果在决策块５８４确定ＲＲ位不等于零，则进入决策块６００在那里确定子系统执行状态是否等于“拒绝”。如果是，则进入块６０２，在那里子系统被从拒绝态脱出面放入接受命令状态。下一步进入决策块５８８，然后按前面解释的那样进行。如果在决策块６００确定子系统执行状态。不等于“拒绝”，则进入决策块５８８并按前面解释的那样继续进行。

下面对照图１７详细描述图１４中的块５１４，这是处理对引起注意端口写。在决策块６０４，确定子系统执行状态是否等于“译码”如果是，则如块６０６所示忽略主处理器写引起注意端口，并返回图１５Ａ所示块５５２继续进行前面解释的过程。如果在决策块６０４确定子系统执行状态不等于“译码”，则进入决策块６０８确定子系统执行状态是否等于“拒绝”。如果是，则如块６１０所示忽略主计算机写引起注意端口，并返回图１６中的块５８８，继续前面解释的过程。

如果在决策块６０８确定子系统执行状态不等于“拒绝”，则进入决策块６１２，在那里确定子系统状态是否等于“复位未决开始”如果是，则如块６１４所示忽略主计算机写引起注意端口并返回图１５ｃ中的块５３２，继续前面解释的过程。

如果在决策块６１２确定子系统执行状态不等于“复位未决”，则进入决策块６１６，以确定子系统状态是否等于“复位未决完成”如果是，则如块６１８所指示，主计算机对引起注意端口写被忽略，

然后返回图１４中的块５０８，使能主处理器中断，如前面解释的那样。

如果在决策块６１６确定子系统执行状态不等于“复位未决完成”则进入决策块６２０，在那里确定子系统执行状态是否等于“子系统复位在进行”。如果是，则如块６２２所示忽略主计算机对引起注意端口的写入，然后返回图１６中的块５９６，继续前面解释过的过程。

如果在块６２０确定子系统执行状态不等于“子系统复位在进行”则继续进行块６２４，在那里子系统执行状态是在“接受”状态，引起注意端口在块６２６中更新。然后子系统执行状态被设置成“译码”如块６２８所示。然后命令处理器在块６３０调用译码处理过程，其细节将在下文中参考图１９和２０来描述。

在决策块６３２确定译码处理过程是否接受了该命令。如果是，则进入块６３４，在那里子系统执行状态被设置成“接受命令”，然后返回图１６中的决策块５８８去进行前已解释的过程。

如果在决策块６３２中确定译码处理过程未曾接受命令，则进入块６３６，在那里将子系统执行状态置成“拒绝”。在块６３８，命令忙／状态端口中的Ｂ位和ＲＥＪ位均被置１，而Ｓ位的设置由译码处理过程来确定。然后返回决策块图１６中的决策块５８８，继续前面解释的过程。

下面对照图１８详细描述图１４中的块５１２，即处理对命令接口端口的写入。在决策块６４０，确定子系统执行状态是否等于“译码”。如果是，则如块６４２所示，忽略主计算机写命令接口端口，然后返回图１５Ａ中的块５５２，继续前面已解释的过程。如果在块

640确定子系统执行状态不等于“译码”，则进入决策块644，在那里确定子系统执行状态是否等于“拒绝”，如果是，主计算机对命令接口端口的写入被忽略，如块646所示，然后返回图16中的块588去继续前面解释过的过程。

如果在决策块644中确定子系统执行状态不等于“拒绝”，进入决策块643，在那里确定子系统执行状态是否等于“复位未决开始”。如果是，例如块650所示，忽略主计算机写命令接口端口，然后返回图15c中的块532，继续前面解释的过程。

如果在决策块648确定子系统执行状态不等于“复位未决开始”进入决策块652，在那里确定子系统执行状态是否等于“复位未决完成”。如果是，则如块654所示，忽略主计算机写命令接口端口然后返回图14中的块508，继续前面解释过的过程。

如果在决策块652确定子系统执行状态不等于“复位未决完成”则进入决策块656，在那里确定子系统执行状态是否等于“子系统复位在进行”。如果是，则如块658所示忽略主计算机写命令接口端口，而返回图16中的块596去继续前面已解释的过程。

如果在决策块656确定子系统执行状态不等于“子系统复位在进行”，则进入块660，在那里子系统执行状态是“接受”状态。于是如块662所示，命令接口端口被主计算机提供之值更新。然后返回图16中的决策块588，继续前面解释过的过程。

命令处理器使用一个“与调用（*call*）相似的”过程来调用“译码处理过程“，以确定一个命令是否能被接受。如果该命令被接受了，则译码处理过程安排一个设备任务，在它把控制转向由命令处理器指出一个命令是否被接受或拒绝之后，该设备任务便去执行该命

令。如果一个命令被接受去执行，则译码过程相应地设置设备状态。如果一个命令被拒绝，则译码过程回送一个值，命令处理器把它作为S位之怎写入命令忙／状态端口。

译码过程是通过来自命令处理器的调用而进入的。一旦进入了译码过程，则可以命令接口端口得到要被处理的命令。引起注意端口包含命令类型，并识别出命令所指向的设备。

下面参照图１９和图２０来详细描述译码处理过程。在决策　块６６４，确定该命令的引起注意代码是否是“设备复位”。如果是，则进入决策块６６６来确定命令接口端口的位７－０中的立即命令代码是否等于零。如果不是，则进入块６７２，在那里译码过程设定它的回送值以指出命令被拒绝。如果决策块６９８或７００的回答为“否（ＮＯ）”，也会发生这种情况，这些操作还将简要地解释。

如果在决策块６６６确定命令接口端口的位７－０中的立即命令代码等于零，进入决策块６６８，在那里确定：如果被访问的设备地址为“０”（即子系统地址），则该命令为一个软件控制的子系统复位。如果是这样，则进入块６８０，在那里子系统的执行状态被设置为“子系统复位在进行”。这个软件控制的子系统复位功能是在块６８２中由译码处理过程调用的。在块６８４，译码处理过程把它的回送代码设为“命令接受”。然后返回块６７６，返回命令处理器。

如果在决策块６６８确定设备地址不是地址“０”（子系统地址）则进入块６７０，在那里确定在引起注意端口的设备标识码（ＩＤ）是否有效。如果是，则进入块６８６，在那里被命令指定的设备的当前“忙子状态”被保存在设备状态分量”旧有的（ＯＬＤ）忙子状态之中。在块６８８，该命令指定的设备的设备状态分量ＩＣ被设置成

命令接口端口中的值。然后，在块690，被访问的设备的忙状态被设置成“忙”。然后，在块692调用子系统任务排序功能，把命令ICMD安排给被访问的设备供其后去执行。然后返回块684去继续前面解释过的过程。

如果在决策块670确定在引起注意端口的设备标识符（ID）无效，则进入块671，在那里命令回送码被设为“拒绝”，且说明理由为无效设备标识符。然后进入块676返回命令处理器。

如果在决策块664确定该命令的引起注意代码不是“设备复位”，则进入决策块694，在那里确定该命令的引起注意代码是否等于“中断复位”。如果不是，则进入决策块696，确定新命令的引起注意代码是否为立即命令。如果是，则进入决策块700，这一块将再简要地描述：如果不是，则进入决策块698，在那里确定新命令的引起注意代码是否是SCB。如果是，则进入决策块702，这也将再简要描述；如果不是，则进入块672，继续前面解释的过程。

如果在决策块694确定该命令的引起注意代码等于“中断复位”则进入决策块700，在那里确定命令接口端口中的命令的位7－0中的立即命令代码是否有效。如果该命令是无效的。则进入块672并继续前面解释过的过程。如果该命令有效，则进入决策块702，在那里确定在引起注意端口的设备标识符是否有效。如果不是，则进入块671并进行前面解释过的过程。如果是，则进入决策块704在那里确定该设备是否处于“可得到”状态。如果不是，则进入块706，在那里把回送代码设为“拒绝”。具体的拒绝代码为“设备不可得到”，如块708所示。然后返回块676继续前面解

释过的过程。

如果在决策块７０４确定该设备处于“可得到”状态，现等参考图２０Ａ，继续进入决策块７１０，在那里确定该设备是否处于“忙”状态。如果是，则进入决策块７１２，在那里确定这个新命令是否是“中断复位”。如果不是，则进入决策块７１４以确定新命令是否为“挂起”。如果不是，则在块７１６把回送码设为“拒绝”。其理由为“设备忙”，如块７１８所示。然后返回图１９Ｂ中的块６７６，继续前面解释过的过程。

如果在决策块７１２或７１４中有一个的决策结果为“是”，则进入决策块７２０，在那里确定当前该设备是否在执行一个ＳＣＢ。如果不是，则返回块７１６去继续前面解释过的过程。如果是，则进入决策块７２２。如果在决策块７１０确定设备“不忙”，则逻辑过程也进入决策块７２２，在决策块７２２确定该新命令的引起注意代码是否为“中断复位”。如果是，则返回图１９Ａ的块６８６。如果不是，则进入决策块７２４，在那里确定该设备中断排队是否满。如果是，则译码处理过程在块７２６中把它的回送代码设置为“拒绝”。具体拒绝代码为“中断排队满”，如块７２８中所示。然后返回图１９Ｂ中的块６７６去继续前面解释过的过程。

如果在决策块724确定设备中断排队不满，则进入决策块７３０，在那里确定新命令是否是“挂起”。如果是，则返回图１９Ａ中的块６８６并继续前面解释过的过程。如果不是，则新命令必定是ＳＣＢ，如块７３２所示。接下来，在决策块７３４确定设备状态是否允许ＳＣＢ的执行。如果是，则返回图１９Ａ中的块６８８，并继续前面解释过的过程。如果不是，则在块７３６中把回送代码置成“拒绝”。

拒绝原因是“ＳＣＢ执行被拒绝”。然后返回图１９Ｂ中的块６７６去继续前面解释过的过程。

总之，已经描述了在主计算机和至少一个可能有附加设备的智能子系统之间传送信息的命令接口。以主处理器发送的信息包括直接命令和间接命令，后者如子系统控制块（ＰＣＢ）的地址，该ＳＣＢ中包含了指定要由识别出的一个智能子系统或设备来完成的操作。从一个智能子系统传回主处理器的信息包括“忙信号”和状态信号。忙信号指示出一个智能子系统的命令接口端口和引起注意端口是否“忙”而状态信号指出这个智能子系统及其附加设备的操作状态。

附 表 1

当前状态	输出至SSCP置RST	输出至SSCP置EI或DMA	输出至SSCP置位6、4、3、2	输出至命令端口	输出至引起注意端口	输出至SSCP置RR
接受	置RST 注1	置EI或DMA 注2	SSCP中位置 注2	置命令端口	置引起注意端口 注3	置RR 注2
未决开始	置RST	忽略置0	忽略置0	忽略置0	忽略置0	忽略置0
未决开始	注4	忽略置0	忽略置0	忽略置0	忽略置0	忽略置0
复位开始	注1	置EI或DMA 注2	SSCP中置位 注2	忽略置0	忽略置0	忽略值0
译码	注1	忽略值不变 注2	忽略值不变 注2	忽略值不变	忽略值不变 注2	忽略值不变 注2
拒绝	注1	置EI或DMA 注2	SSCP中位置 注2	忽略值不变	忽略值不变 注2	置RR 注5

附　表　2

决策表1：在译码处理过程中接受命令

新的命令或引起注意代码	设备号有效	设备可得到	有效8位立即操作	忙状态	忙子状态	中断挂队满	使能执行SCB
设备复位或子系统复位	是	不必管它	是	不必管它	不必管它	不必管它	不必管它
中断状态端口复位	是	是	是	间	不能应用	不必管它	不必管它
SCB命令或取决于设备的注意码作为SCB	是	是	不能应用	间	不能应用	否	是
立即命令未挂起，设备复位子系统复位SCB中断复位中断状态端口复位	是	是	是	间	不能应用	否	不能应用
SCB　中断复位	是	是	是	间	不能应用	不必管它	不必管它
SCB　中断复位	是	是	是	忙	SCB	不必管它	是
挂　起	是	是	是	忙	SCB	否	不必管它
挂　起	是	是	是	间	不能应用	否	不必管它

附　表　3

决策表2：在译码处理过程中拒绝命令＝拒绝理由

新的命令或引起注意的代码	设备号有效	设备可得到	有效8位命令操作码	忙状态	忙子状态	中断排队满	使能执行SCB	由译码处理置拒绝代码
设备复位	不必管它	不必管它	否	不必管它	不必管它	不必管它	不必管它	无效命令
设备复位	不必管它	不必管它	是	不必管它	不必管它	不必管它	不必管它	无效命令
设备复位	不必管它	不必管它	是	不必管它	不必管它	不必管它	不必管它	无效命令
设备复位	否	不必管它	是	不必管它	不必管它	不必管它	不必管它	无效设备号
无效注意代码	不必管它	不必管它	不必管它	不必管它	不必管它	不必管它	不必管它	无效命令
有效注意非SCB或取决与设备的注意码作为SCB或设备复位	不必管它	不必管它	否	不必管它	不必管它	不必管它	不必管它	无效命令
有效注意非SCB或取决与设备的注意码作为SCB或设备复位	不必管它	不必管它	是	不必管它	不必管它	不必管它	不必管它	无效命令

决策表2（第2部分）：在译码处理过程中拒绝命令＝拒绝理由

新的命令或注意代码	设备号有效	设备可得到	有效8位命令操作码	忙状态	忙子状态	中断排队满	使能执行SCB	译码过程设置拒绝代码
有效注意非复位	否	不必管它	如果非SCB则是	不必管它	不必管它	不必管它	不必管它	无效设备号
有效注意非复位	是	否	如果非SCB则是	不必管它	不必管它	不必管它	不必管它	设备不能得到
任何命令但挂起、设备复位、SCB中断复位	是	是	如果非SCB则是	是	忙	不必管它	不必管它	设备忙
挂起或SCB中断复位	是	是	是	忙	非SCB	不必管它	不必管它	设备忙
SCB或取决于设备的注意码作为SCB	是	是	不能应用	间	不能应用	是	不必管它	中断排队满
挂起	是	是	是	间	不能应用	是	不必管它	中断排队满
挂起	是	是	是	忙	SCB	是	不必管它	中断排队满

决策表2（第3部分）：在译码处理过程中拒绝命令，拒绝理由

新的命令或注意代码	设备号有效	设备可得到	有效8位命令操作码	忙状态	忙子状态	中断排队满	使能执行SCB	译码过程设置拒绝代码
立即命令非设置复位SCB中断复位中断状态端口复位	是	是	是	间	不能应用	是 ****	不能应用	中断排队满
SCB或设备注意码作为SCB	是	是	不能应用	间	不能应用	否	否 ****	执行被挂起

专利号 90 1 03808
Int. Cl.[5] G06F 13/42
授权公告日 1993年6月23日

图1

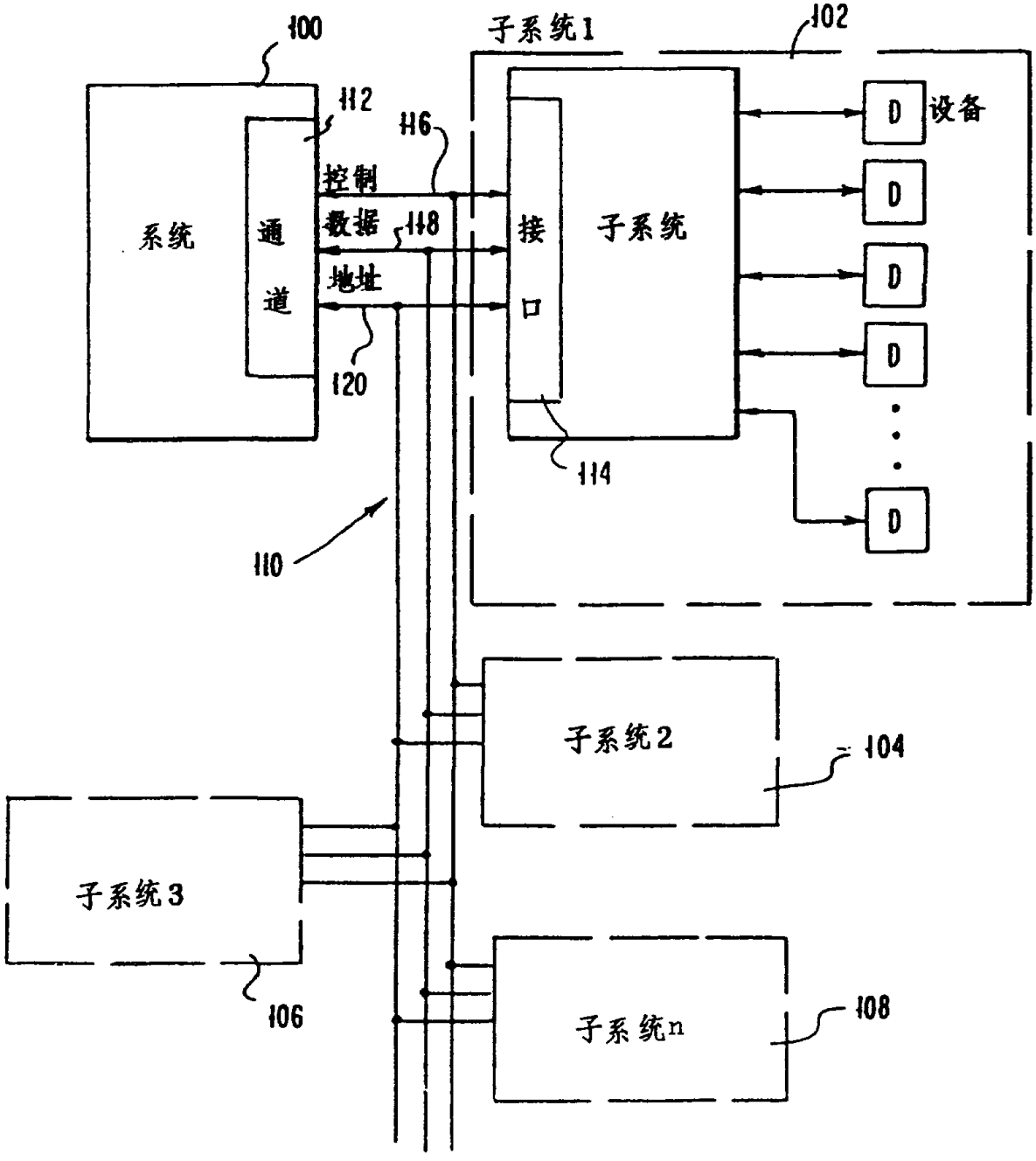


图2

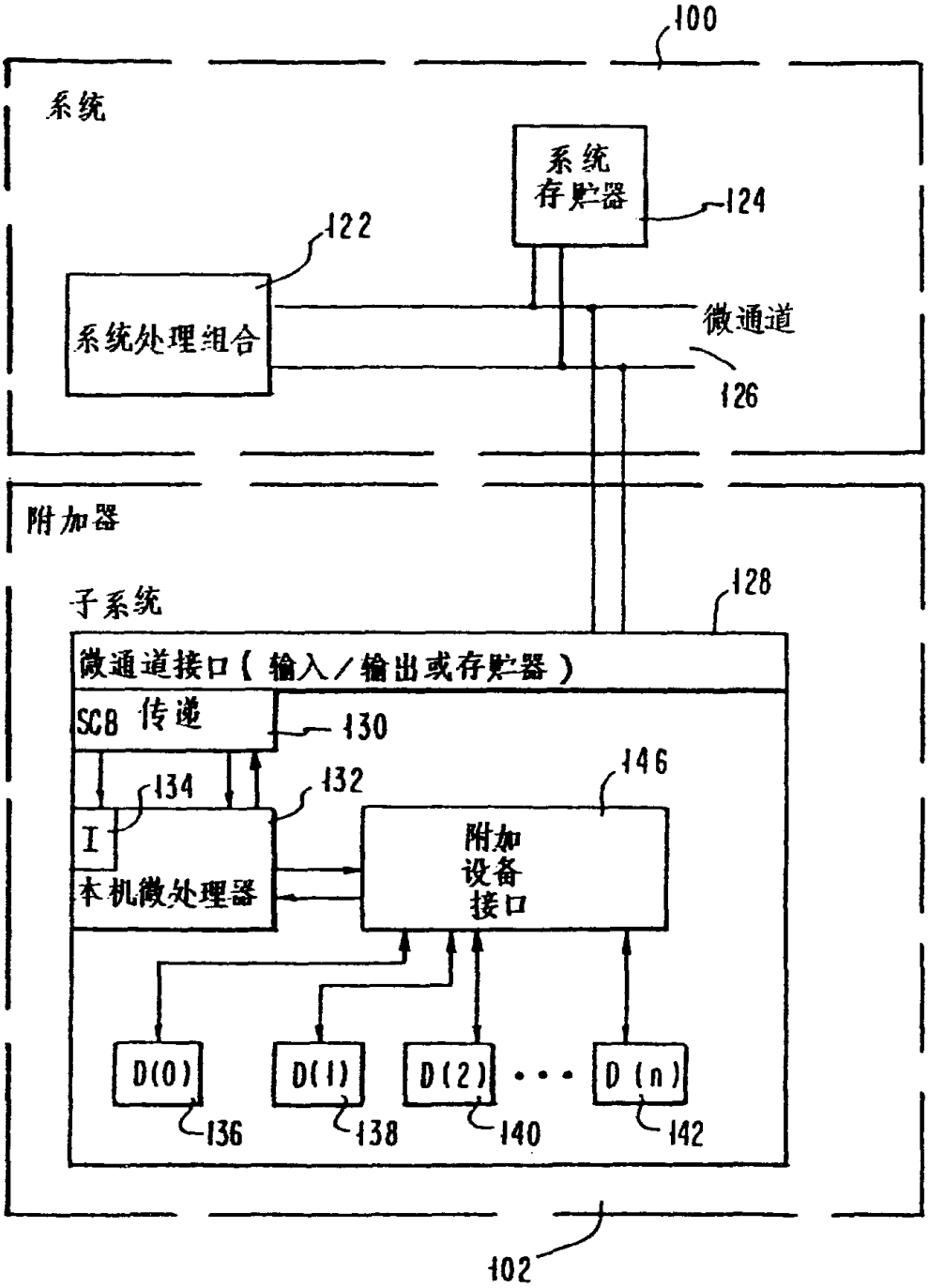
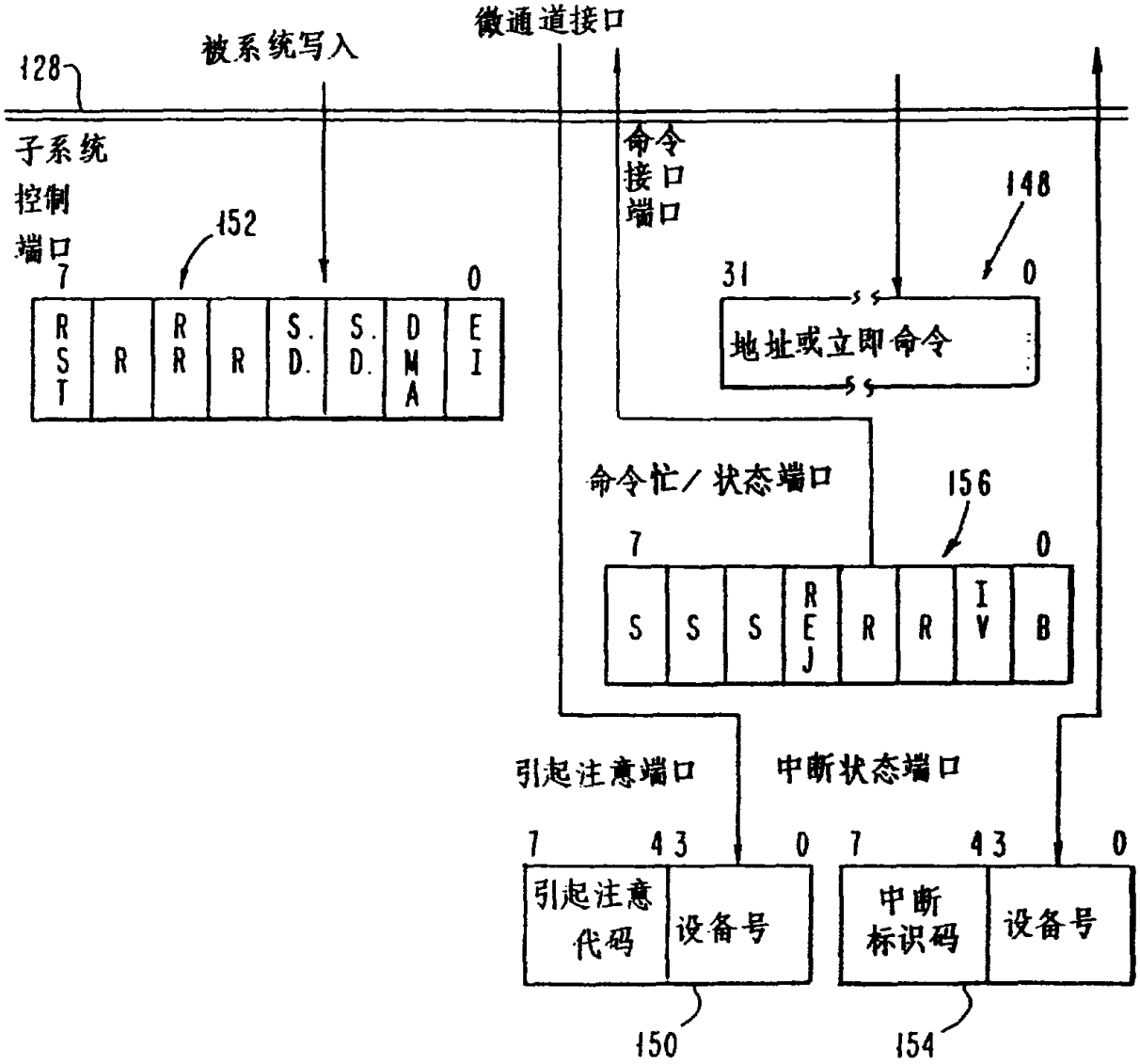

100
系统
系统存贮器
124
122
系统处理组合
微通道
126
附加器
子系统
128
微通道接口（输入／输出或存贮器）
SCB 传递
130
134
132
146
I
本机微处理器
附加设备接口
D(0)
D(1)
D(2)
D(n)
136
138
140
142
102

图3


微通道接口
被系统写入
128
子系统
控制
端口
152
7
0
RST
R
RR
R
S.D.
S.D.
DMA
EI
命令
接口
端口
148
31
0
地址或立即命令
命令忙／状态端口
156
7
0
S
S
S
REJ
R
R
IV
B
引起注意端口
中断状态端口
7
4 3
0
引起注意代码
设备号
中断标识码
设备号
150
154

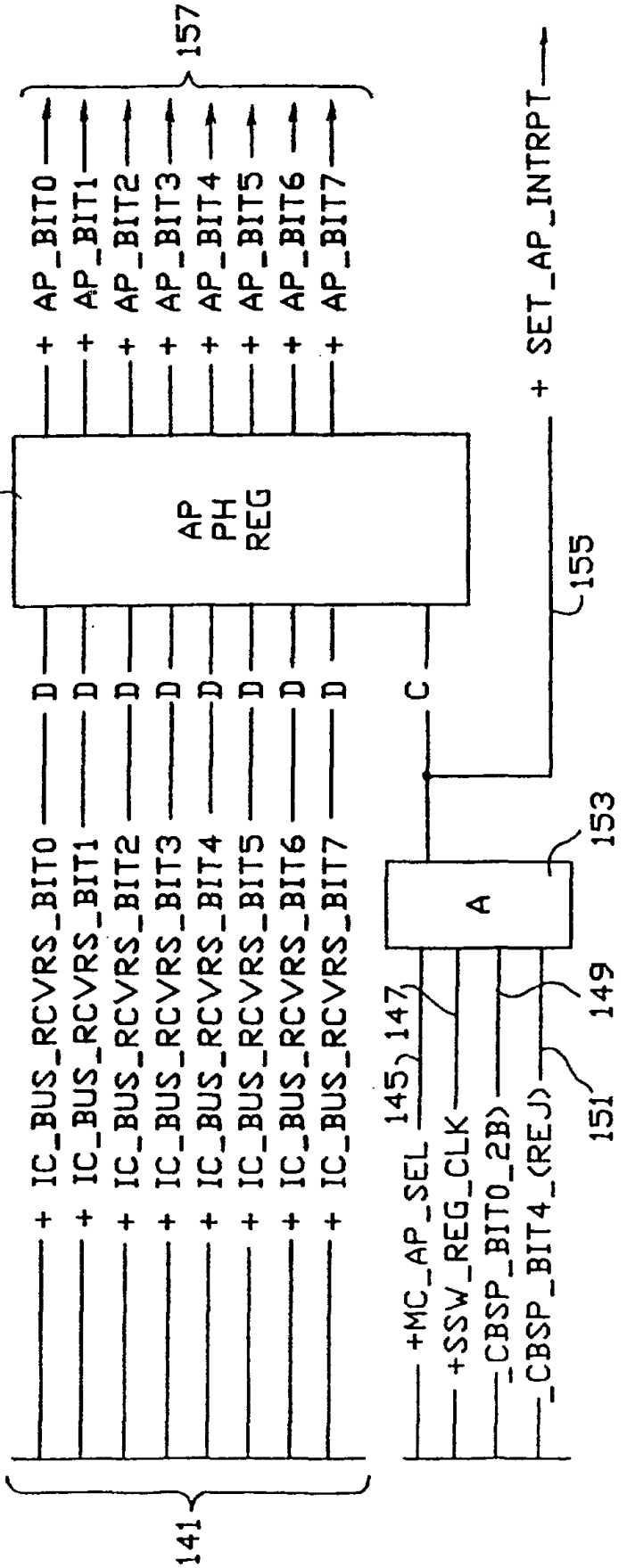

+ IC_BUS_RCVRS_BIT0
+ IC_BUS_RCVRS_BIT1
+ IC_BUS_RCVRS_BIT2
+ IC_BUS_RCVRS_BIT3
+ IC_BUS_RCVRS_BIT4
+ IC_BUS_RCVRS_BIT5
+ IC_BUS_RCVRS_BIT6
+ IC_BUS_RCVRS_BIT7
141
D
C
143
AP
PH
REG
+ AP_BIT0
+ AP_BIT1
+ AP_BIT2
+ AP_BIT3
+ AP_BIT4
+ AP_BIT5
+ AP_BIT6
+ AP_BIT7
157
+ SET_AP_INTRPT
155
A
153
+MC_AP_SEL
145
147
+SSW_REG_CLK
_CBSP_BIT0_2B)
149
_CBSP_BIT4_(REJ)
151

図4

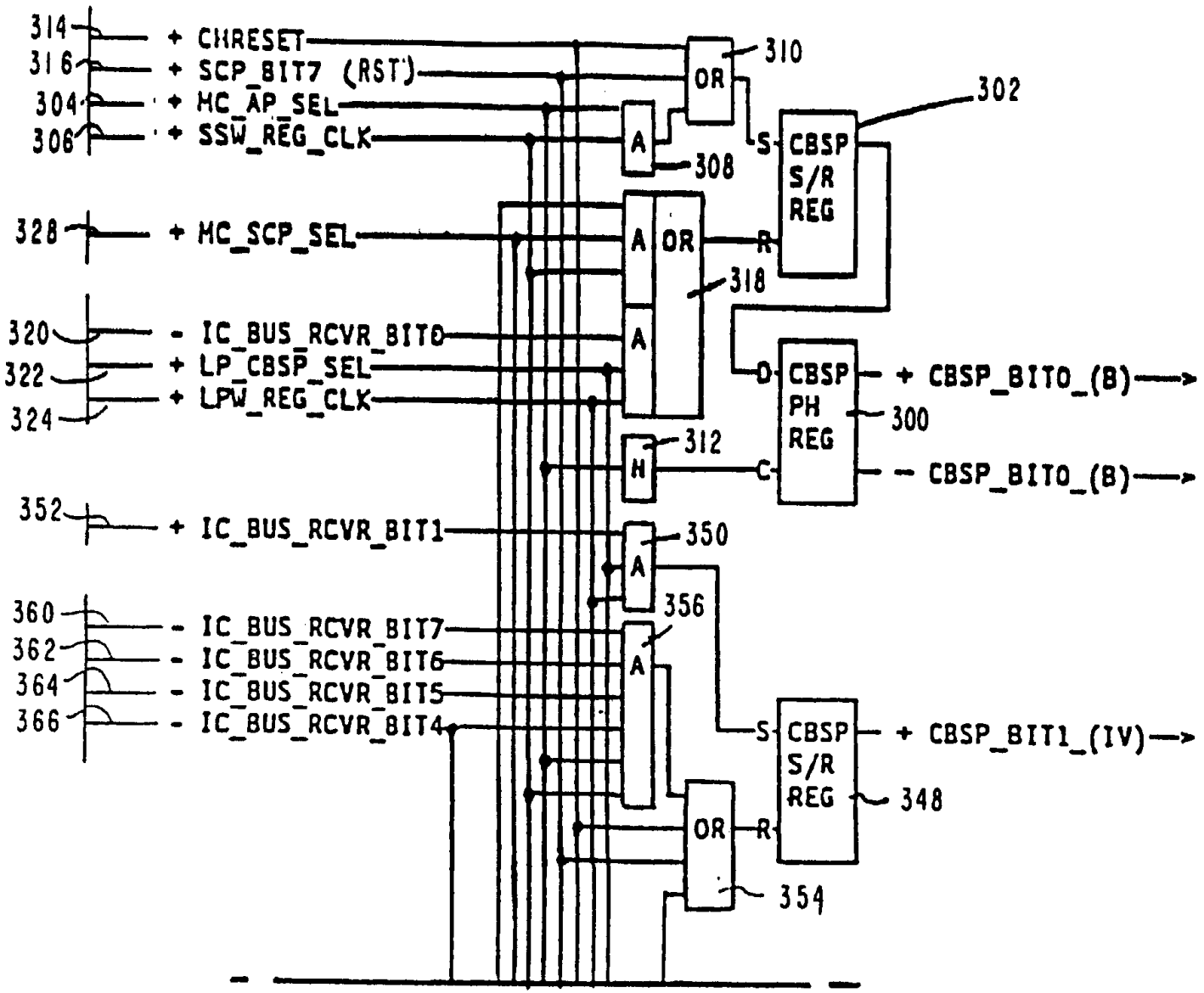


图5

图 5A
图 5B

图5A

図5B

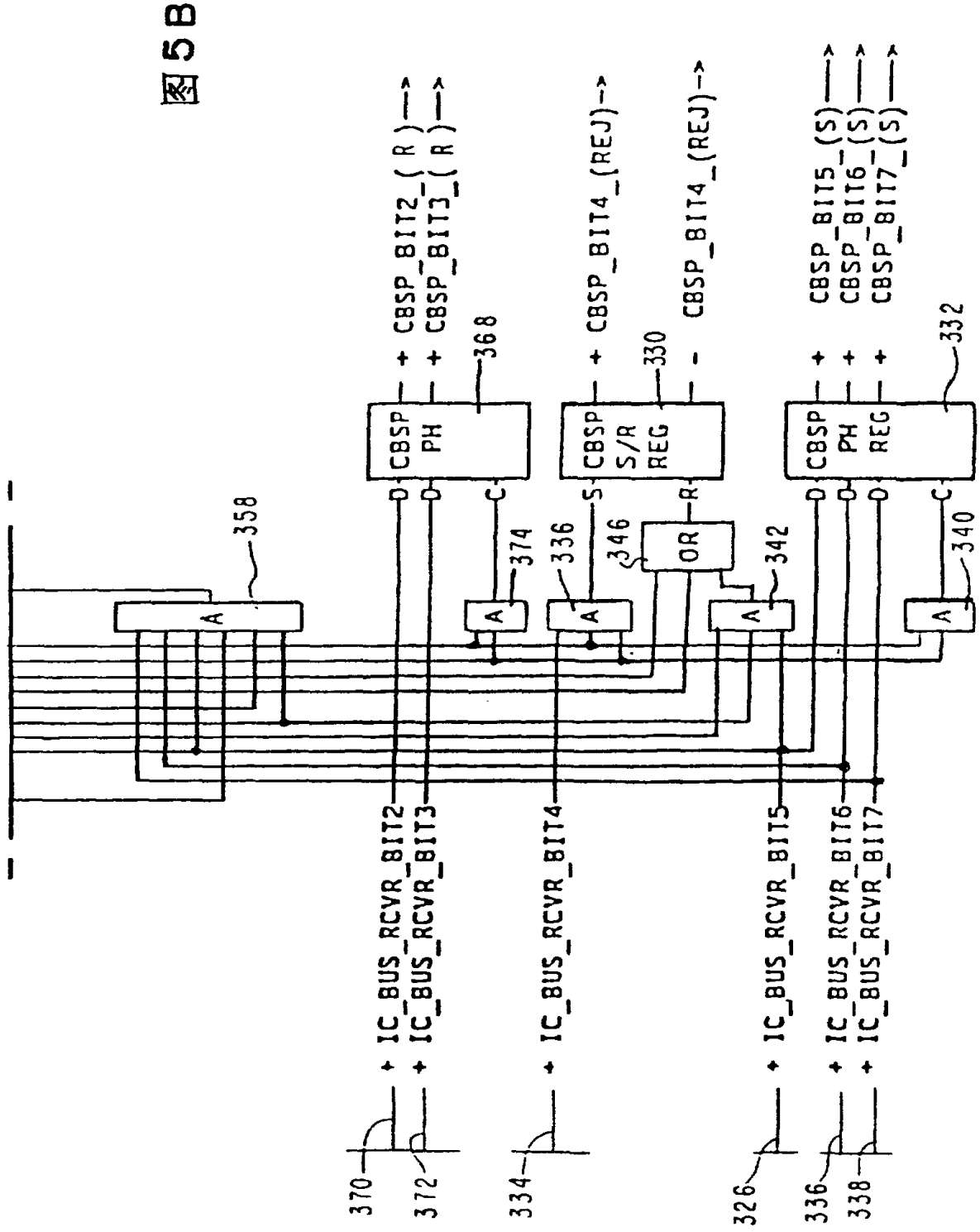
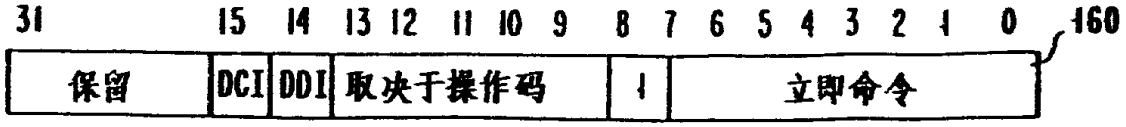

+ IC_BUS_RCVR_BIT2
+ IC_BUS_RCVR_BIT3
+ IC_BUS_RCVR_BIT4
+ IC_BUS_RCVR_BIT5
+ IC_BUS_RCVR_BIT6
+ IC_BUS_RCVR_BIT7
370
372
334
326
336
338
358
A
374
336
346
342
340
OR
CBSP PH
CBSP S/R REG
CBSP PH REG
368
330
332
+ CBSP_BIT2_(R)
+ CBSP_BIT3_(R)
+ CBSP_BIT4_(REJ)
- CBSP_BIT4_(REJ)
+ CBSP_BIT5_(S)
+ CBSP_BIT6_(S)
+ CBSP_BIT7_(S)

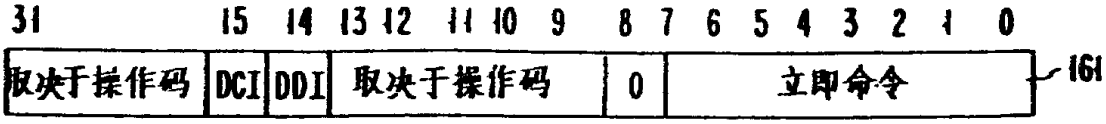
图6

命令接口端口格式

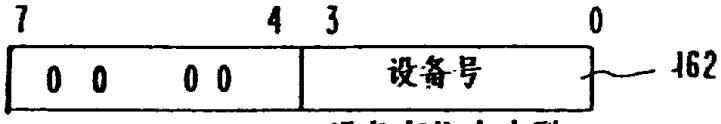


立即命令格式型1

（或）

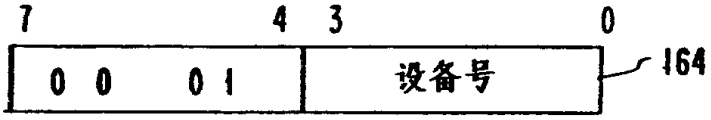


立即命令格式型2引起注意端口格式



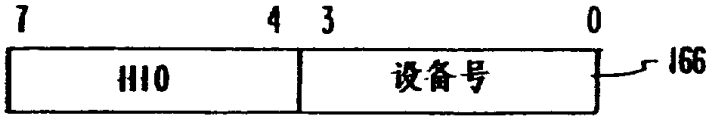
设备复位命令型

（或）



立即命令（命令型）

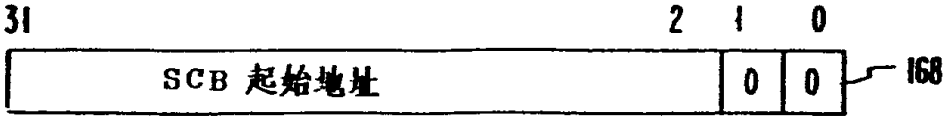
（或）



中断复位（EOI）命令型

图7

命令接口端口格式



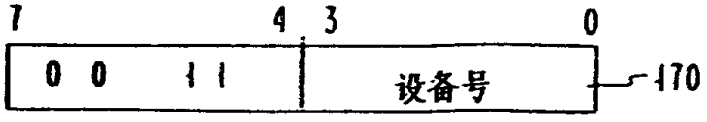


图8

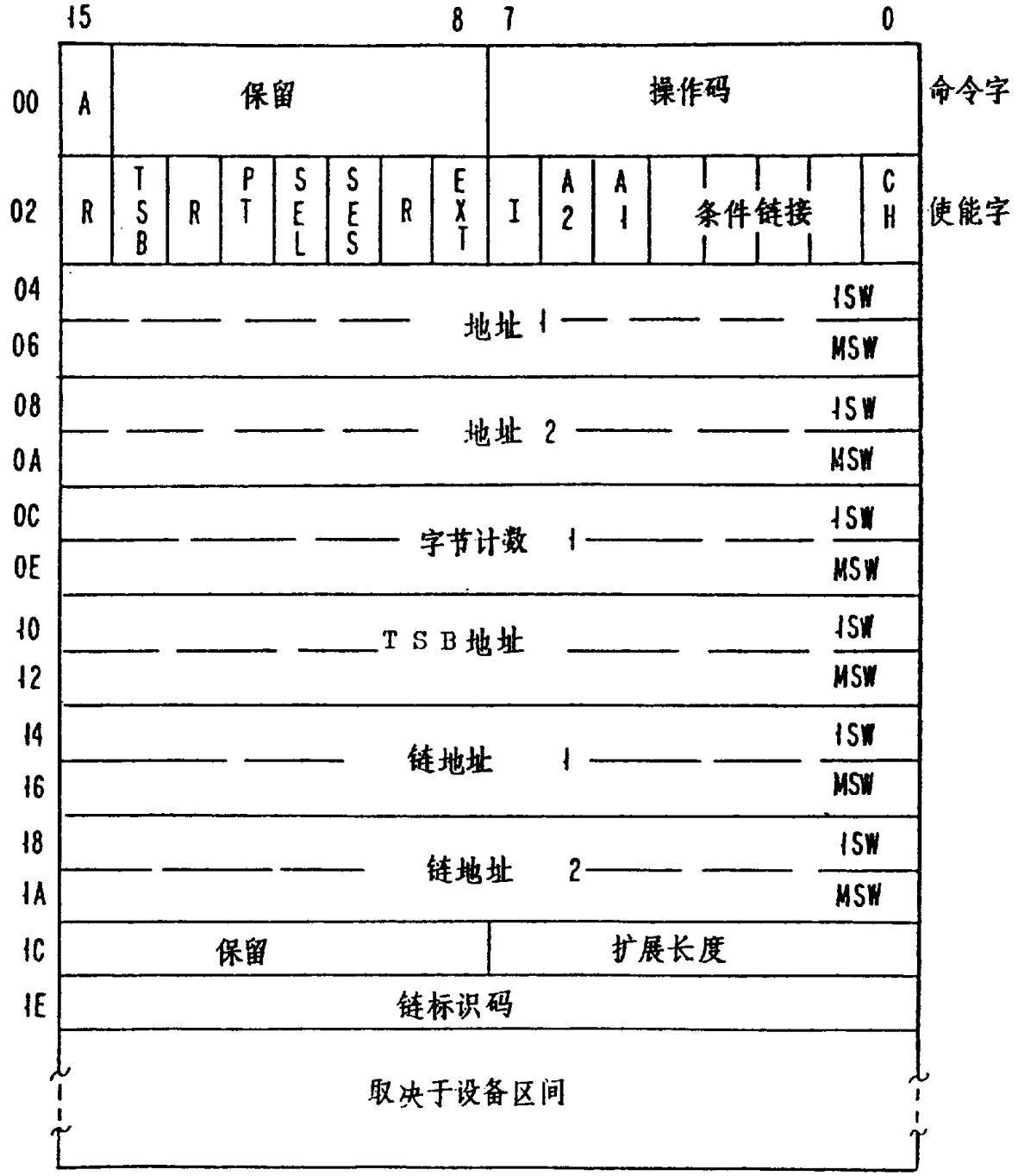

15
8
7
0
00
A
保留
操作码
命令字
02
R
TSB
R
PT
SEL
SES
R
EXT
I
A2
A1
条件链接
CH
使能字
04
06
地址 1
LSW
MSW
08
0A
地址 2
LSW
MSW
0C
0E
字节计数 1
LSW
MSW
10
12
TSB地址
LSW
MSW
14
16
链地址 1
LSW
MSW
18
1A
链地址 2
LSW
MSW
1C
保留
扩展长度
1E
链标识码
取决于设备区间

图9

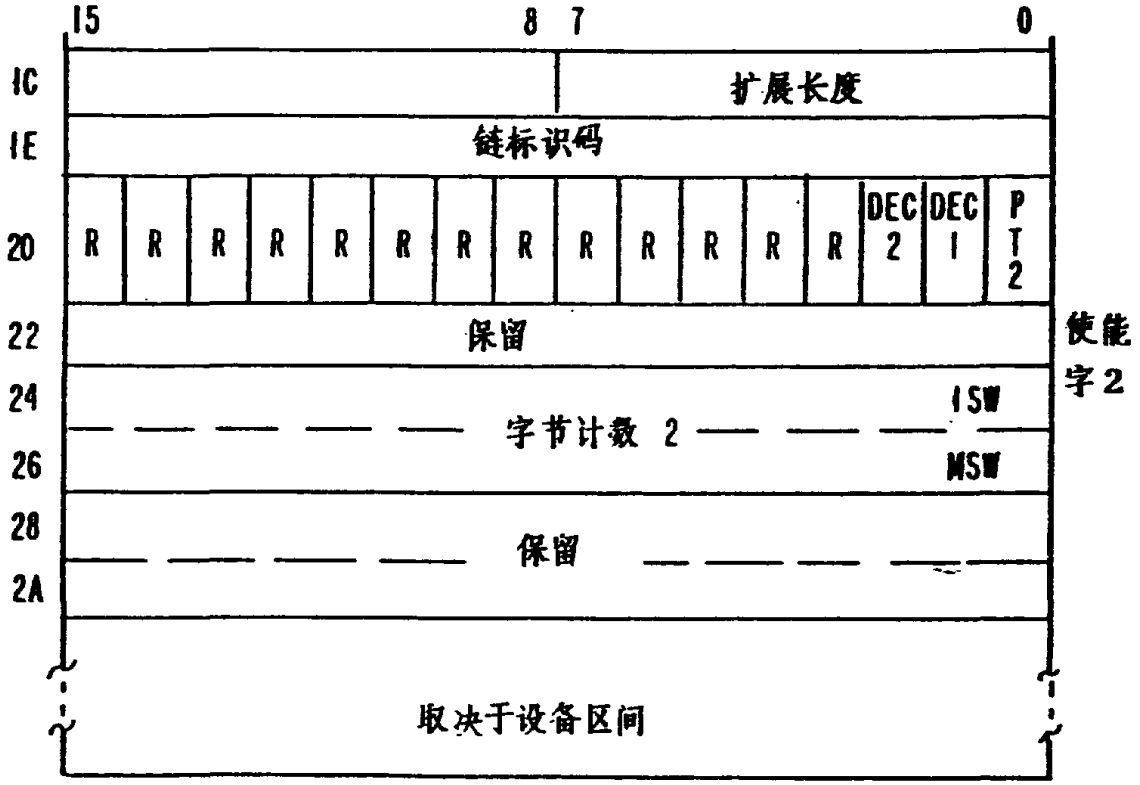

15
8
7
0
1C
扩展长度
1E
链标识码
20
R
R
R
R
R
R
R
R
R
R
R
R
R
DEC 2
DEC 1
PT 2
22
保留
使能字2
24
ISW
字节计数 2
26
MSW
28
保留
2A
取决于设备区间

图10

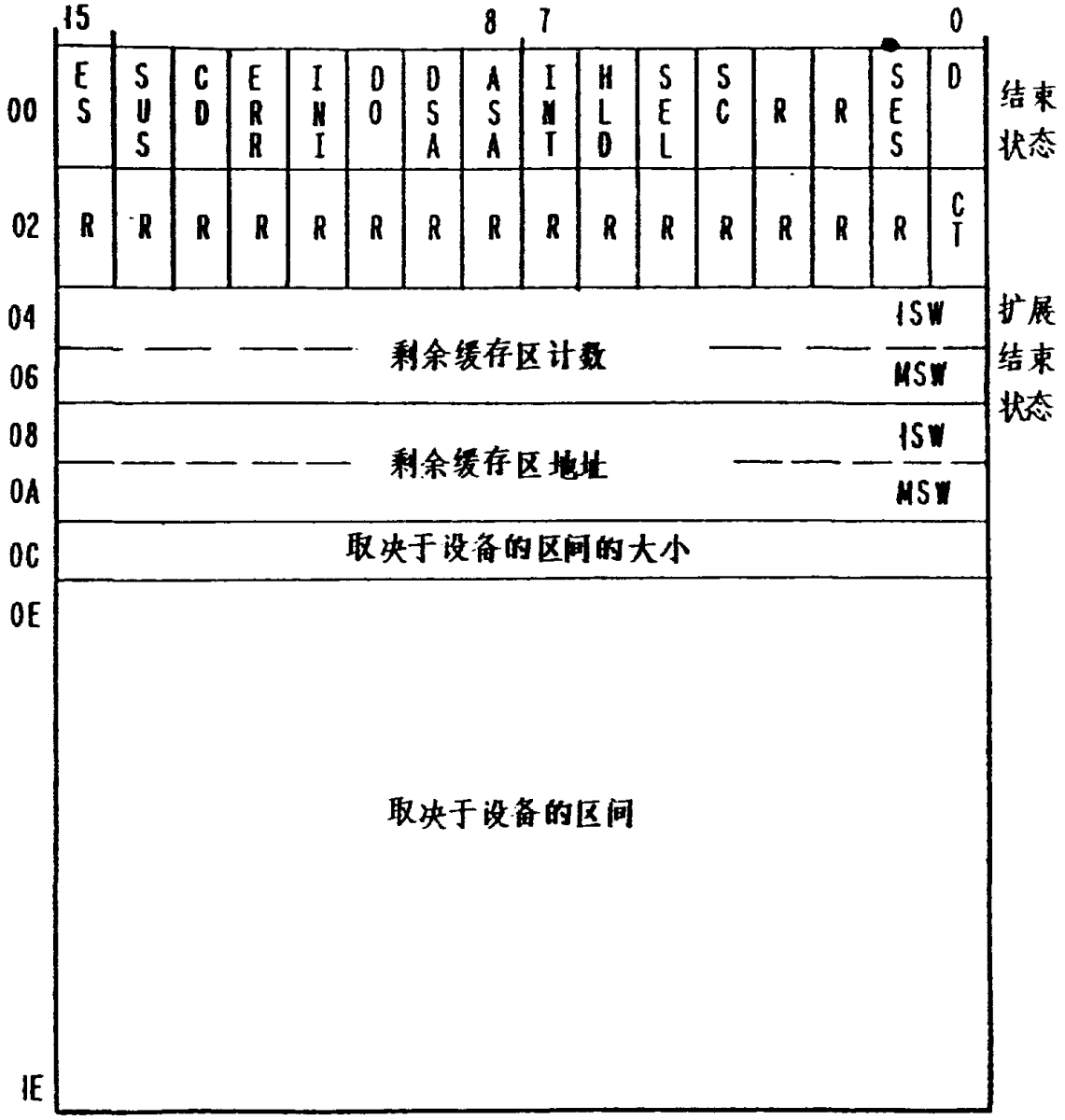

15
8
7
0
ES
SUS
CD
ERR
INI
DO
DSA
ASA
IMT
HLD
SEL
SC
R
R
SES
D
结束状态
00
02
R
CT
04
06
ISW
MSW
扩展结束状态
剩余缓存区计数
08
0A
剩余缓存区地址
0C
取决于设备的区间的大小
0E
取决于设备的区间
1E

图11

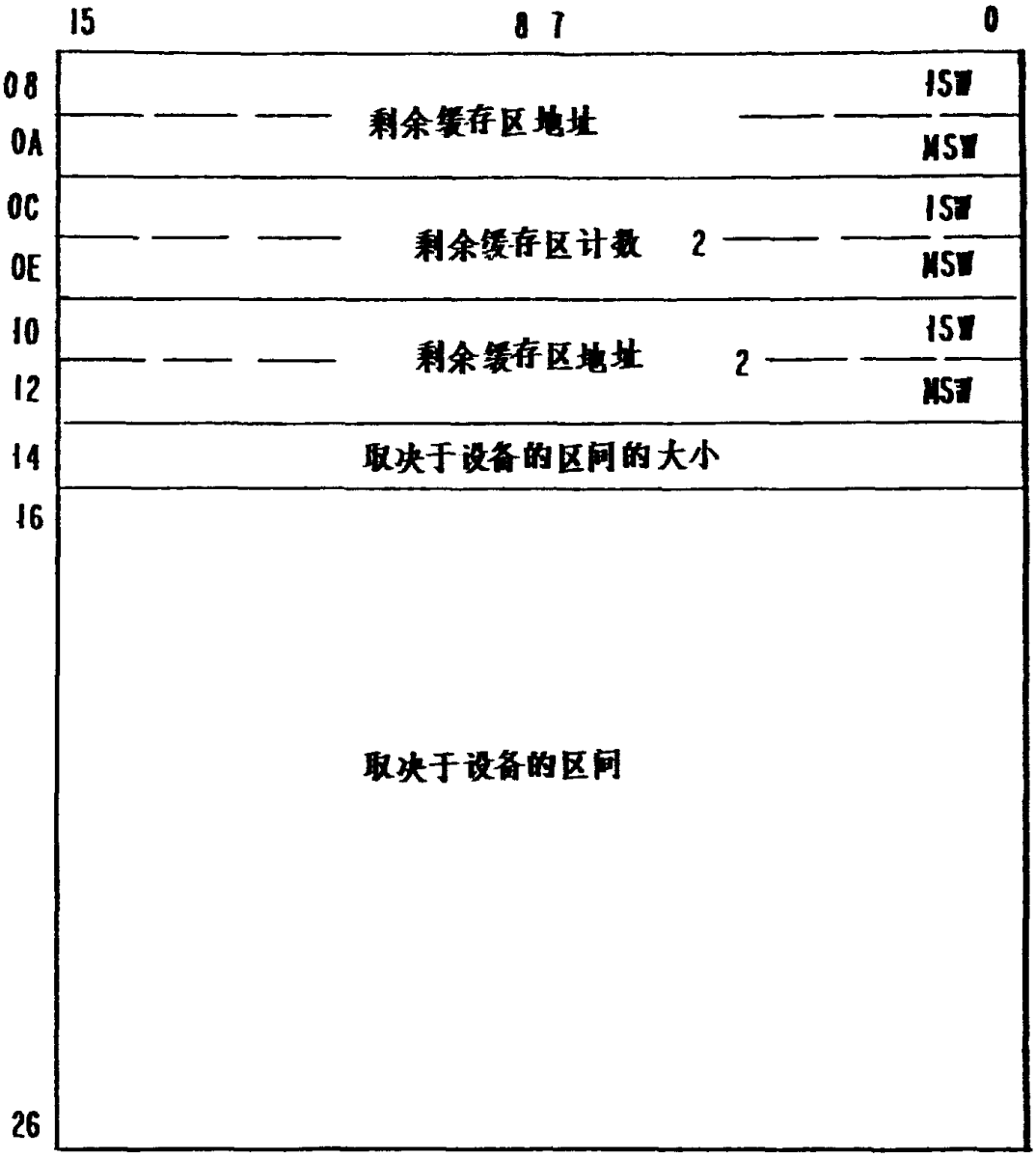
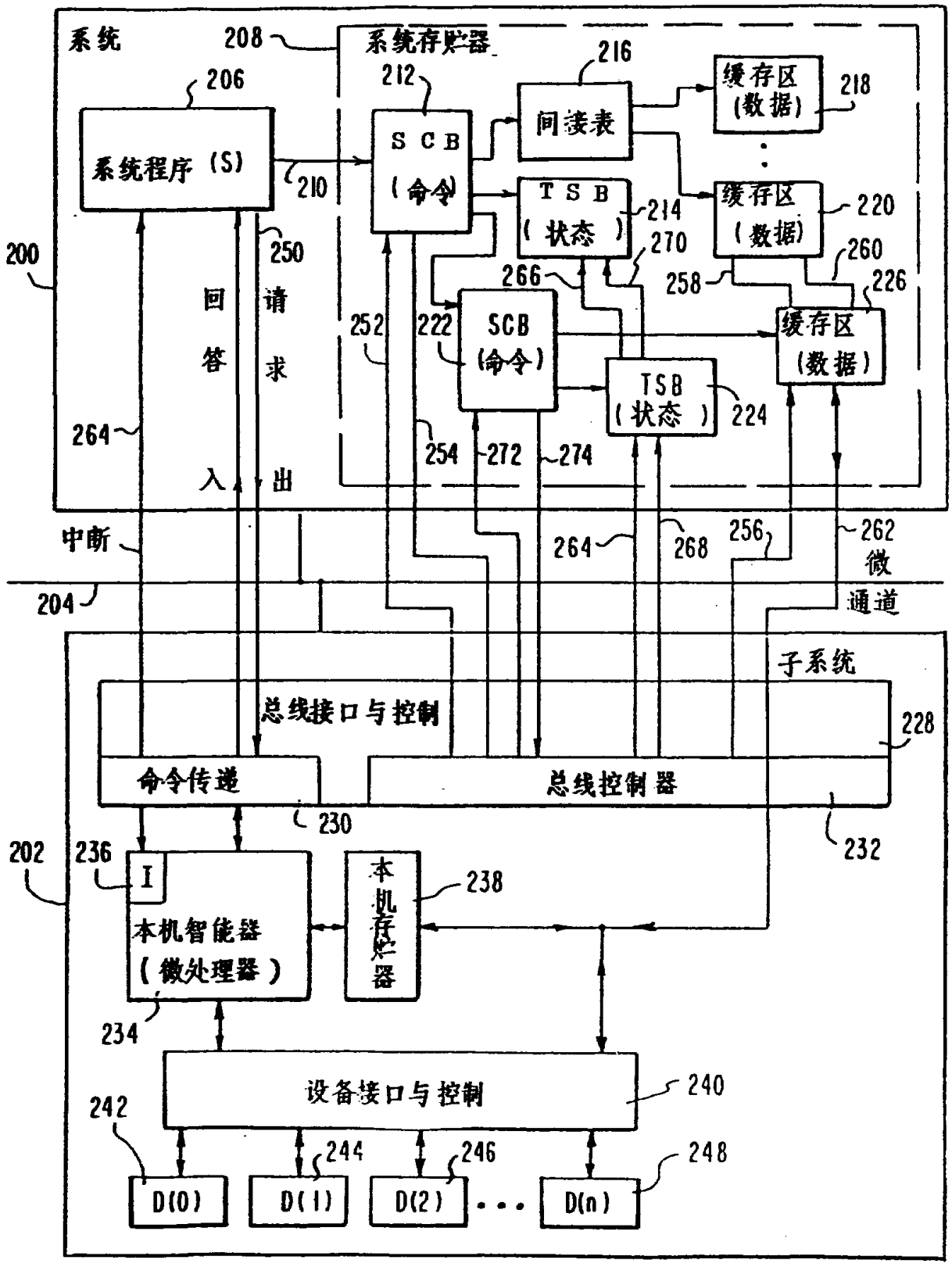

15
8 7
0
08
0A
0C
0E
10
12
14
16
26
剩余缓存区地址
LSW
MSW
剩余缓存区计数
2
剩余缓存区地址
2
取决于设备的区间的大小
取决于设备的区间

图12


系统
208
系统存贮器
206
212
216
缓存区
(数据)
218
系统程序 (S)
SCB
(命令)
间接表
210
TSB
(状态)
214
缓存区
(数据)
220
200
250
270
260
266
258
226
回答
请求
252
222
SCB
(命令)
缓存区
(数据)
TSB
(状态)
224
264
254
272
274
入
出
中断
264
268
256
262
微通道
204
子系统
总线接口与控制
228
命令传递
总线控制器
230
232
202
236
I
本机智能器
(微处理器)
本机存贮器
238
234
设备接口与控制
240
242
244
246
248
D(0)
D(1)
D(2)
D(n)

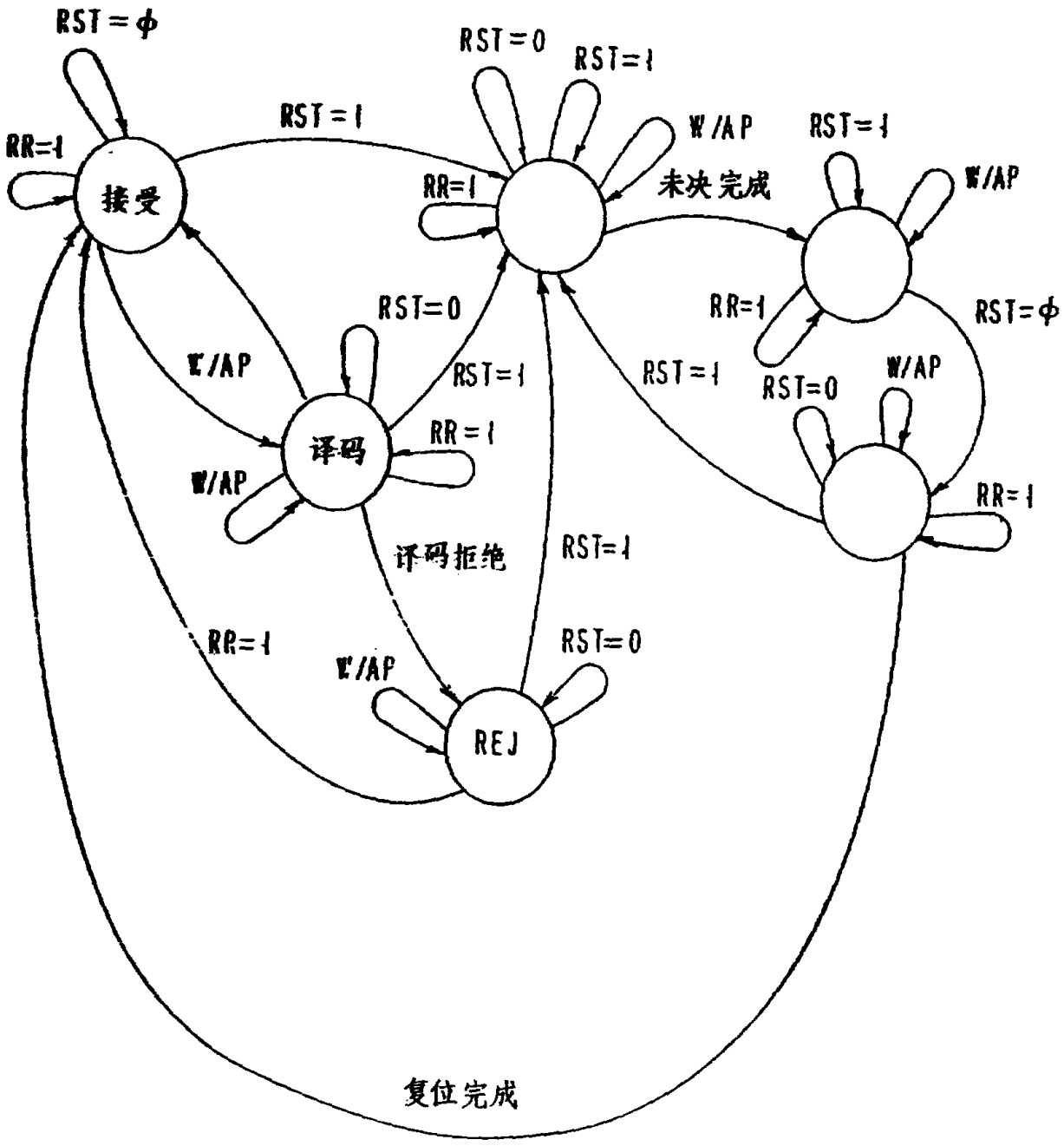
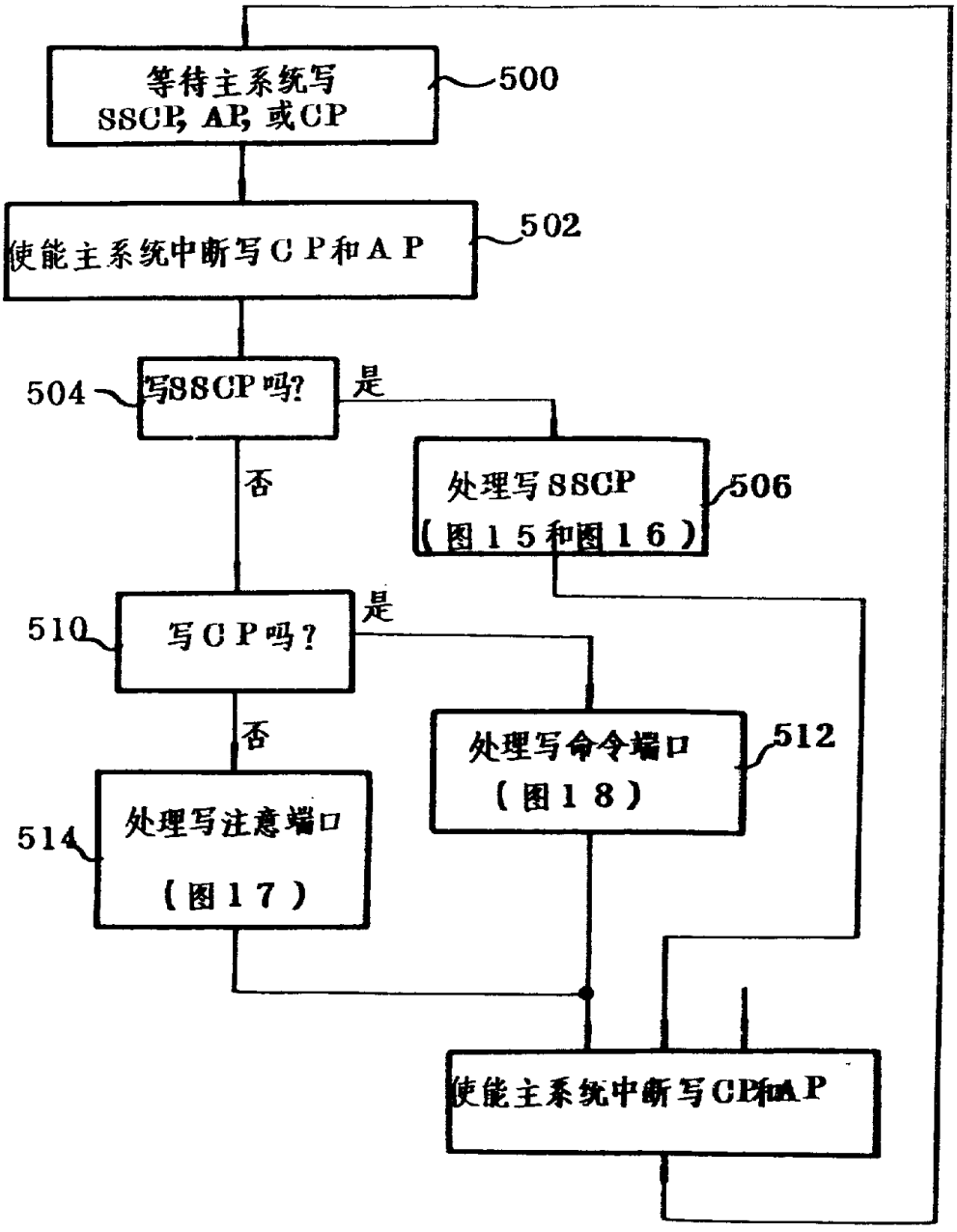

RST=φ
RST=0
RST=1
RST=1
W/AP
RST=1
W/AP
RR=1
接受
RR=1
未决完成
RST=0
RR=1
RST=φ
W/AP
RST=1
RST=1
W/AP
RST=0
译码
RR=1
W/AP
RR=1
译码拒绝
RST=1
RR=1
W/AP
RST=0
REJ
复位完成

图13

图14


等待主系统写
SSCP, AP, 或CP
500
使能主系统中断写CP和AP
502
504
写SSCP吗?
是
否
处理写SSCP
(图15和图16)
506
510
写CP吗?
是
否
处理写命令端口
(图18)
512
514
处理写注意端口
(图17)
使能主系统中断写CP和AP

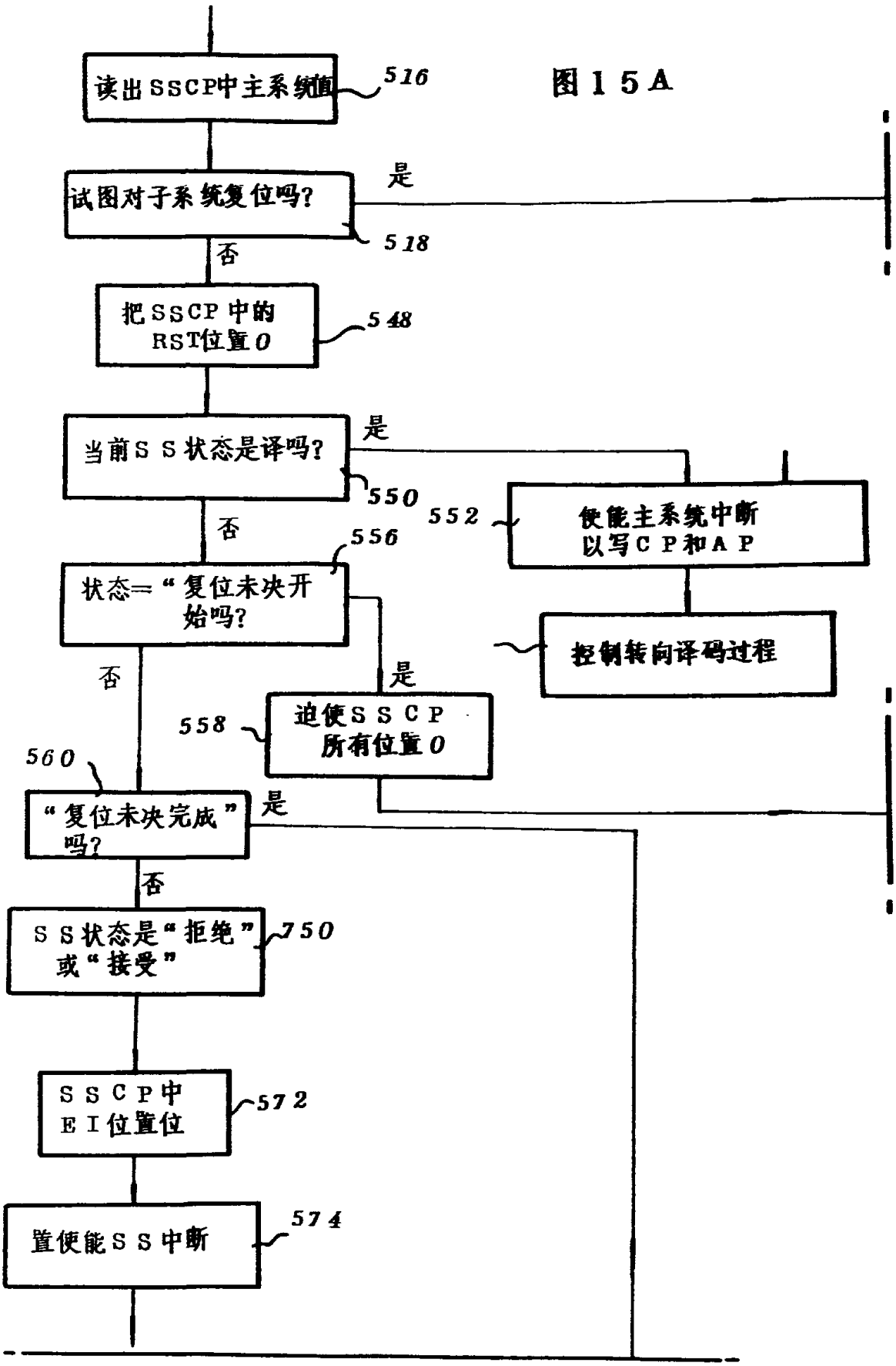

读出SSCP中主系统值
516
试图对子系统复位吗?
是
518
否
把SSCP中的
RST位置0
548
当前SS状态是译吗?
是
550
552
使能主系统中断
以写CP和AP
否
556
状态="复位未决开
始吗?
控制转向译码过程
否
是
558
迫使SSCP
所有位置0
560
"复位未决完成"
吗?
是
否
SS状态是"拒绝"
或"接受"
750
SSCP中
EI位置位
572
置使能SS中断
574

图15A

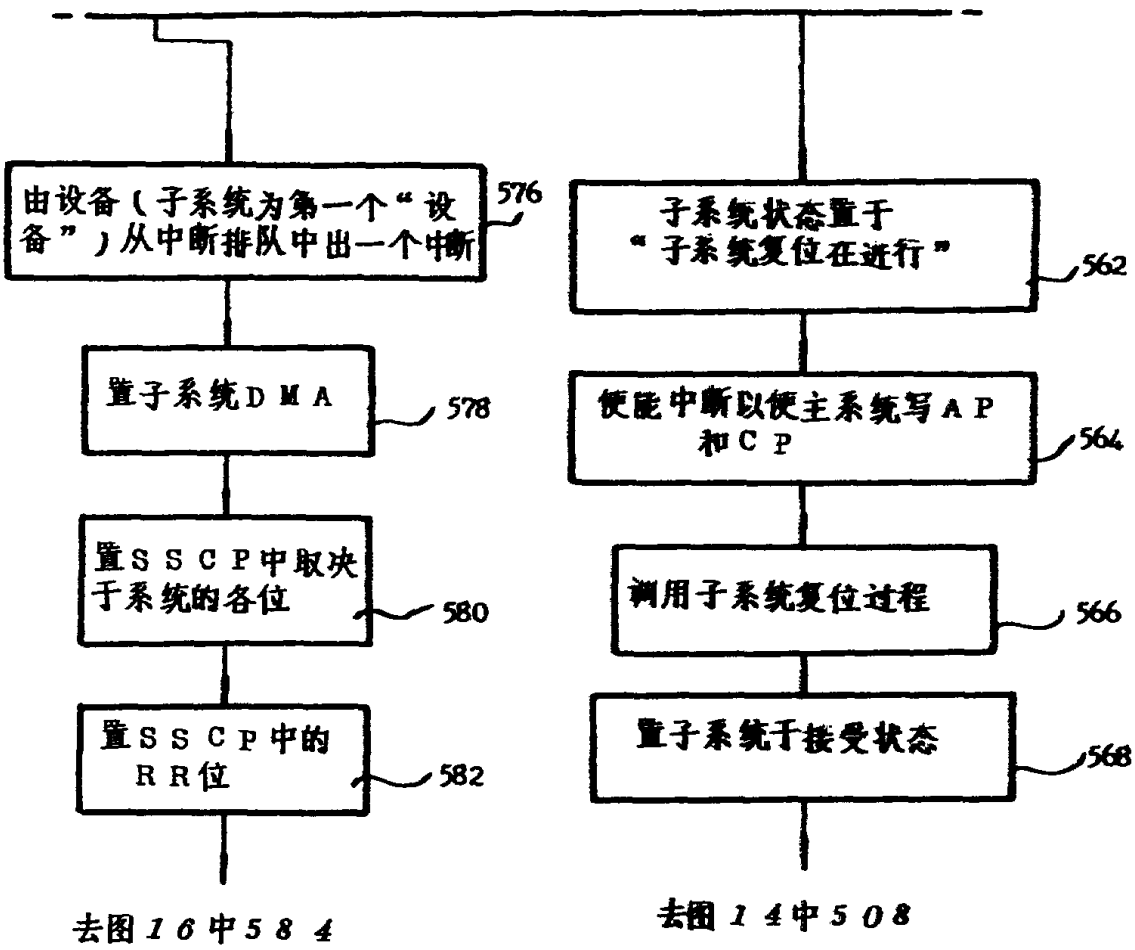

由设备（子系统为第一个“设备”）从中断排队中出一个中断
576
置子系统DMA
578
置SSCP中取决于系统的各位
580
置SSCP中的RR位
582
去图16中584
子系统状态置于“子系统复位在进行”
562
使能中断以便主系统写AP和CP
564
调用子系统复位过程
566
置子系统于接受状态
568
去图14中508

图15B

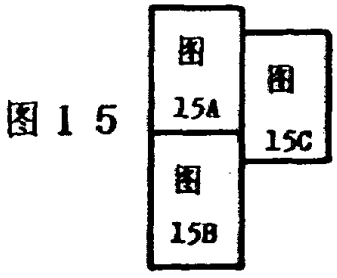

图15
图15A
图15B
图15C

图15C

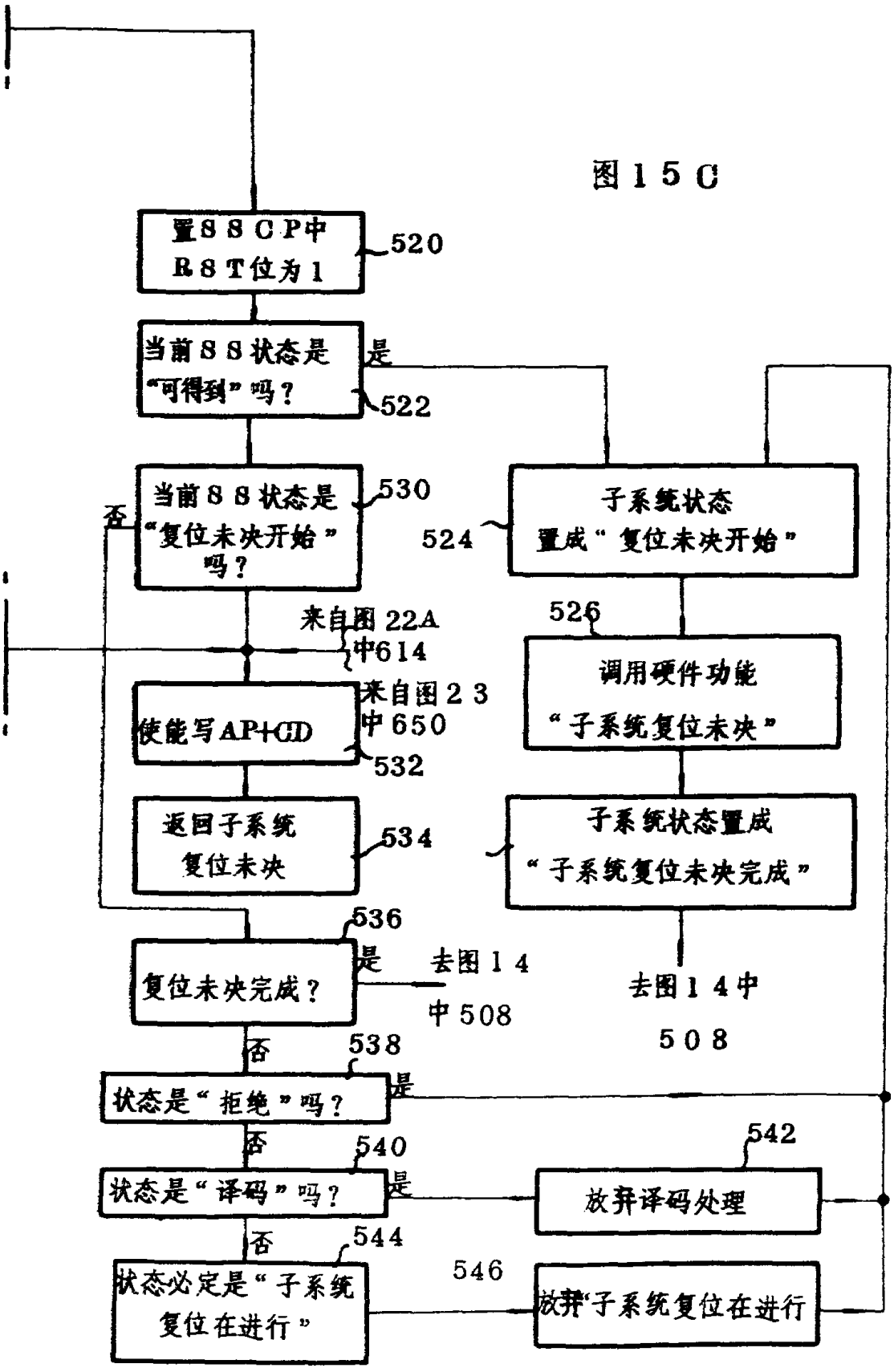

置SSCP中RST位为1
520
当前SS状态是"可得到"吗?
是
522
当前SS状态是"复位未决开始"吗?
530
否
子系统状态置成"复位未决开始"
524
来自图22A中614
526
调用硬件功能"子系统复位未决"
使能写APPCD
来自图23中650
532
返回子系统复位未决
534
子系统状态置成"子系统复位未决完成"
536
复位未决完成?
是
去图14中508
去图14中508
否
538
状态是"拒绝"吗?
是
否
540
状态是"译码"吗?
是
542
放弃译码处理
否
544
状态必定是"子系统复位在进行"
546
放弃"子系统复位在进行

图16

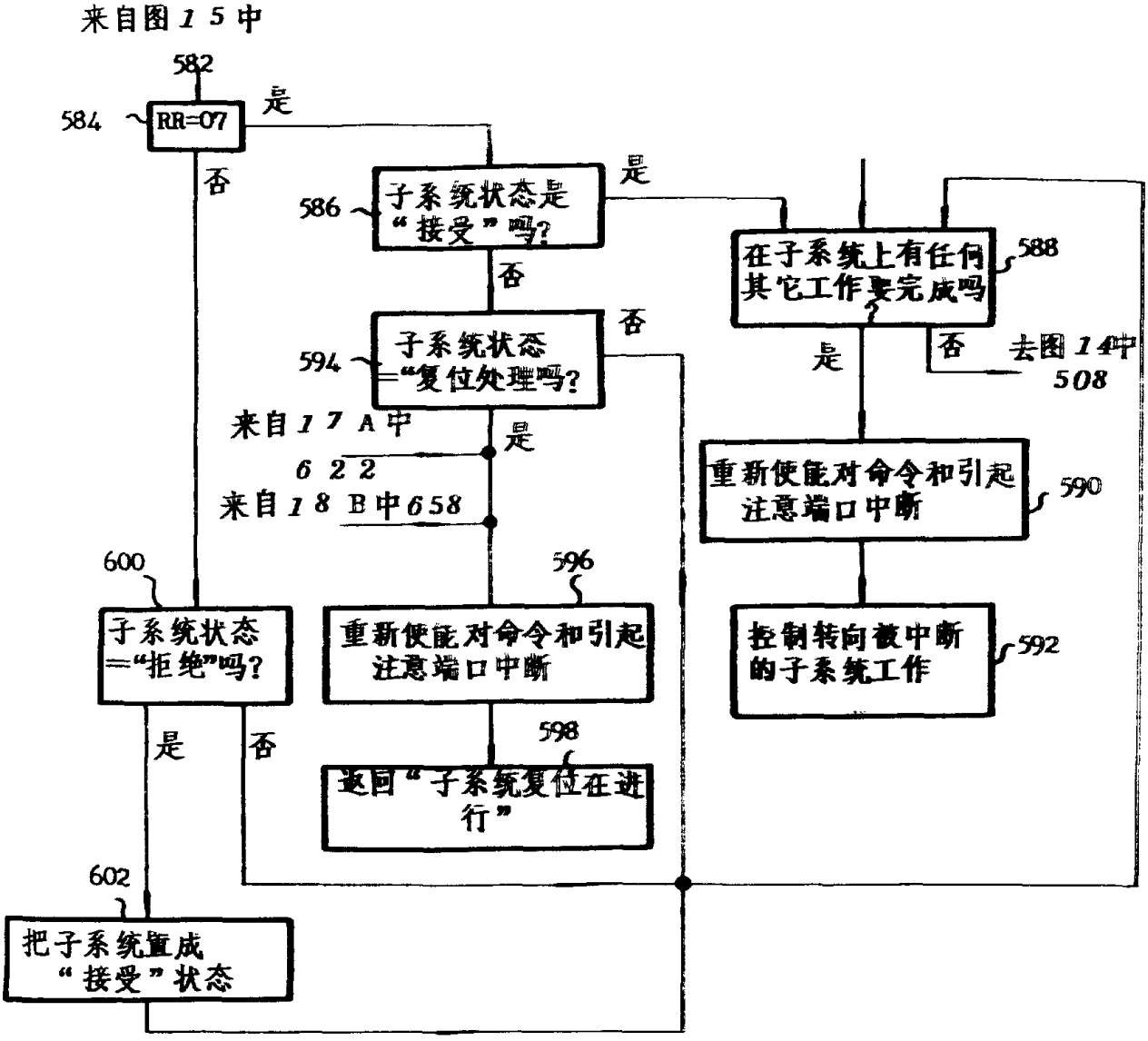
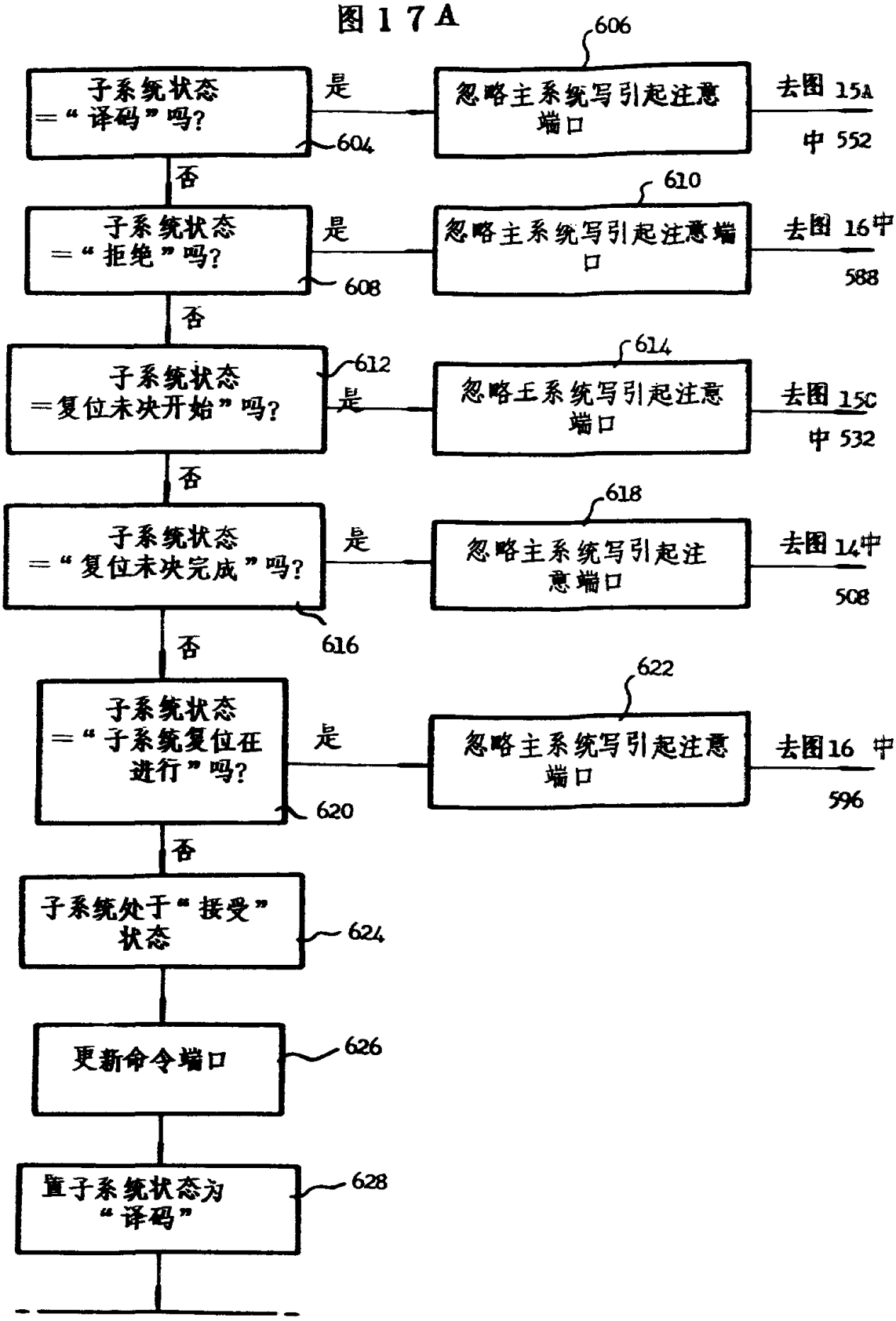

来自图15中
582
584
RR=07
是
否
586
子系统状态是"接受"吗?
是
否
588
在子系统上有任何其它工作要完成吗
否
去图14中508
是
590
重新使能对命令和引起注意端口中断
592
控制转向被中断的子系统工作
594
子系统状态="复位处理吗?
否
来自17A中622
来自18B中658
是
596
重新使能对命令和引起注意端口中断
598
返回"子系统复位在进行"
600
子系统状态="拒绝"吗?
是
否
602
把子系统置成"接受"状态

图 17A


子系统状态 = "译码"吗?
604
是
606
忽略主系统写引起注意端口
去图 15A 中 552
否
子系统状态 = "拒绝"吗?
608
是
610
忽略主系统写引起注意端口
去图 16中 588
否
子系统状态 = "复位未决开始"吗?
612
是
614
忽略主系统写引起注意端口
去图 15C 中 532
否
子系统状态 = "复位未决完成"吗?
616
是
618
忽略主系统写引起注意端口
去图 14中 508
否
子系统状态 = "子系统复位在进行"吗?
620
是
622
忽略主系统写引起注意端口
去图 16 中 596
否
子系统处于"接受"状态
624
更新命令端口
626
置子系统状态为"译码"
628

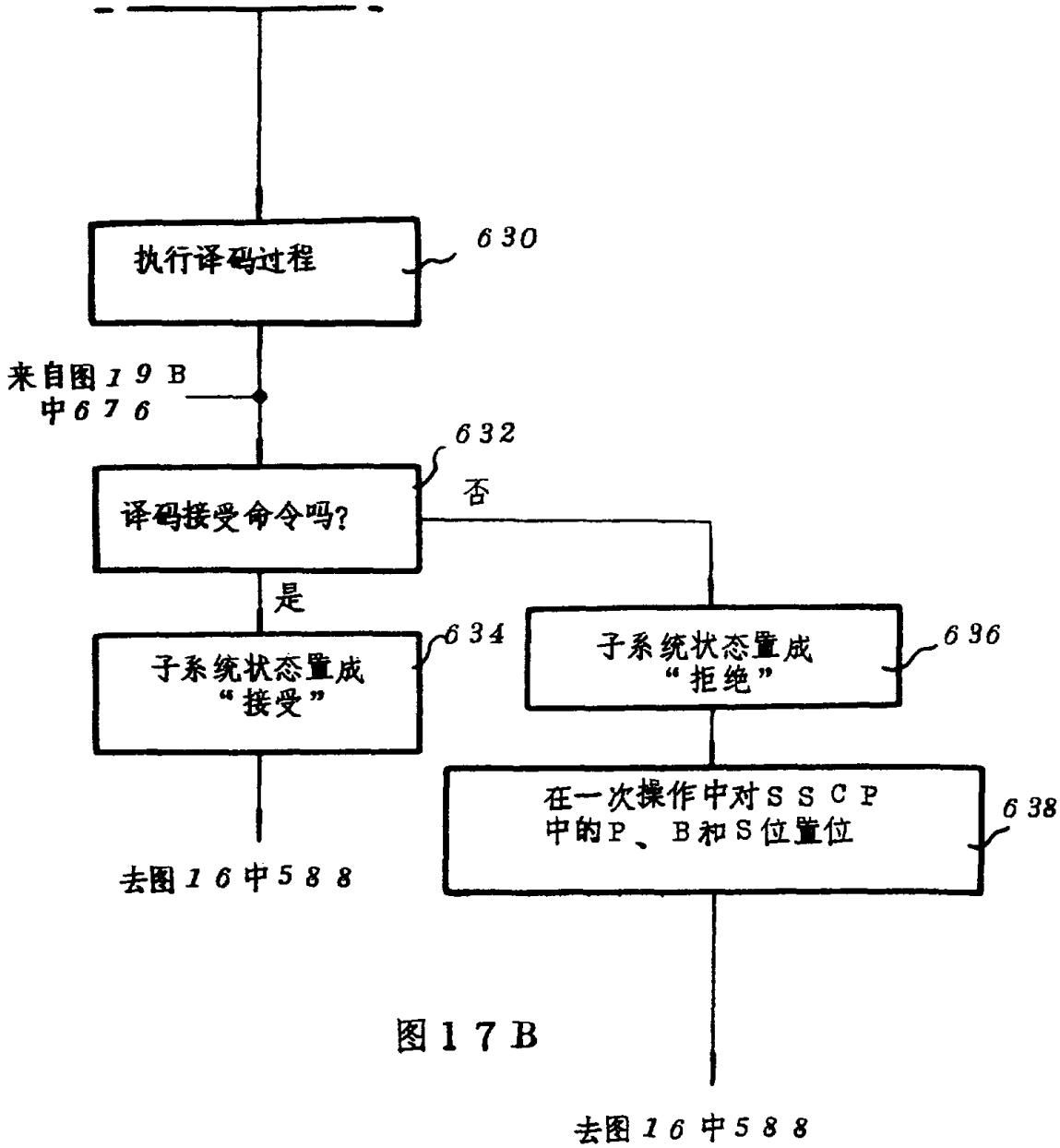

执行译码过程
630
来自图19B
中676
632
译码接受命令吗?
否
是
子系统状态置成
“接受”
634
子系统状态置成
“拒绝”
636
在一次操作中对SSCP
中的P、B和S位置位
638
去图16中588
去图16中588

图17B

图17

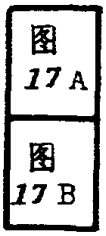
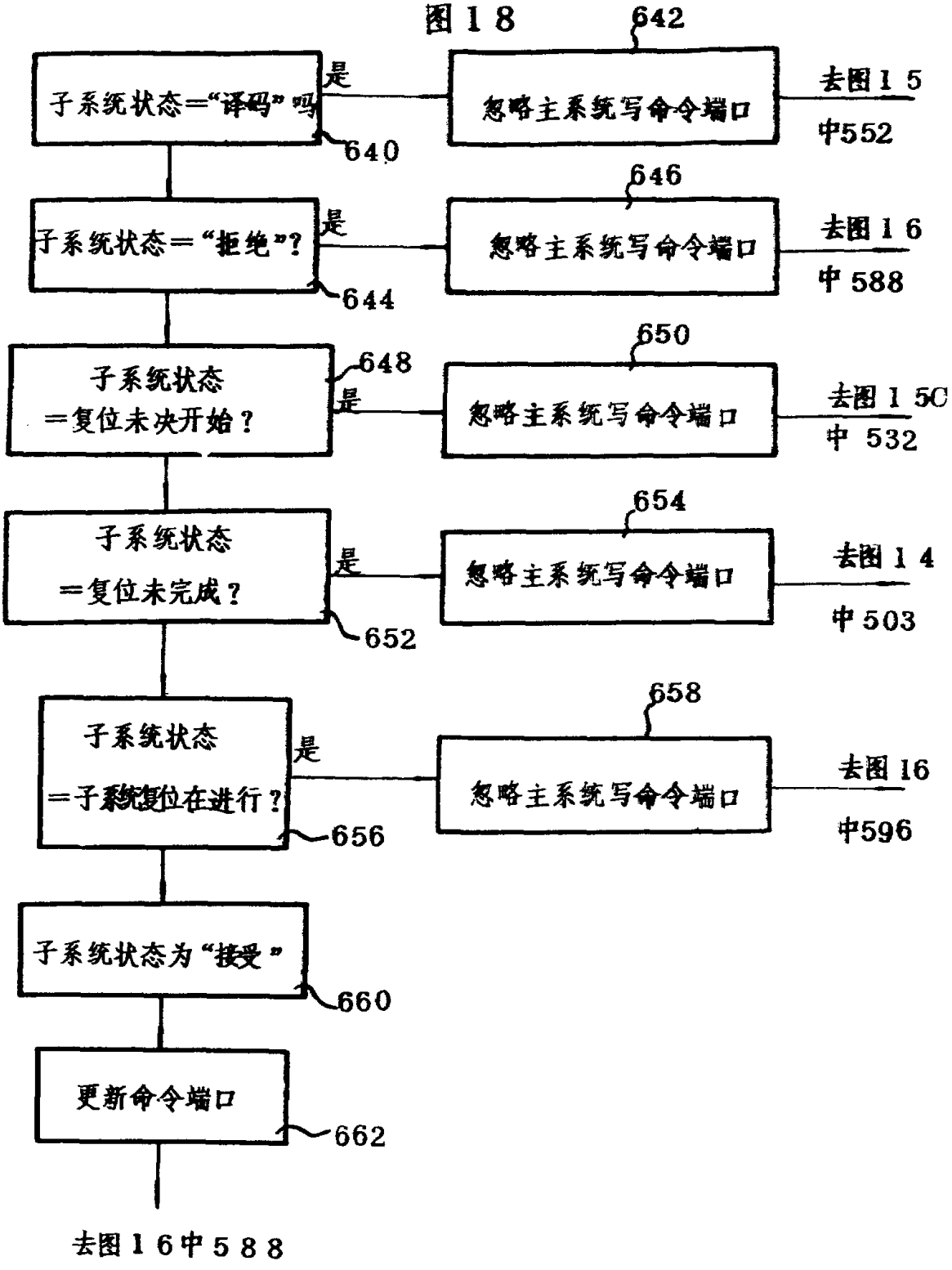

图
17A
图
17B

图 1 8


子系统状态="译码"吗
是
640
642
忽略主系统写命令端口
去图 1 5
中552
子系统状态="拒绝"?
是
644
646
忽略主系统写命令端口
去图 1 6
中 588
子系统状态
=复位未决开始?
是
648
650
忽略主系统写命令端口
去图 1 5C
中 532
子系统状态
=复位未完成?
是
652
654
忽略主系统写命令端口
去图 1 4
中 503
子系统状态
=子系统复位在进行?
是
656
658
忽略主系统写命令端口
去图 16
中596
子系统状态为"接受"
660
更新命令端口
662
去图 1 6中 5 8 8

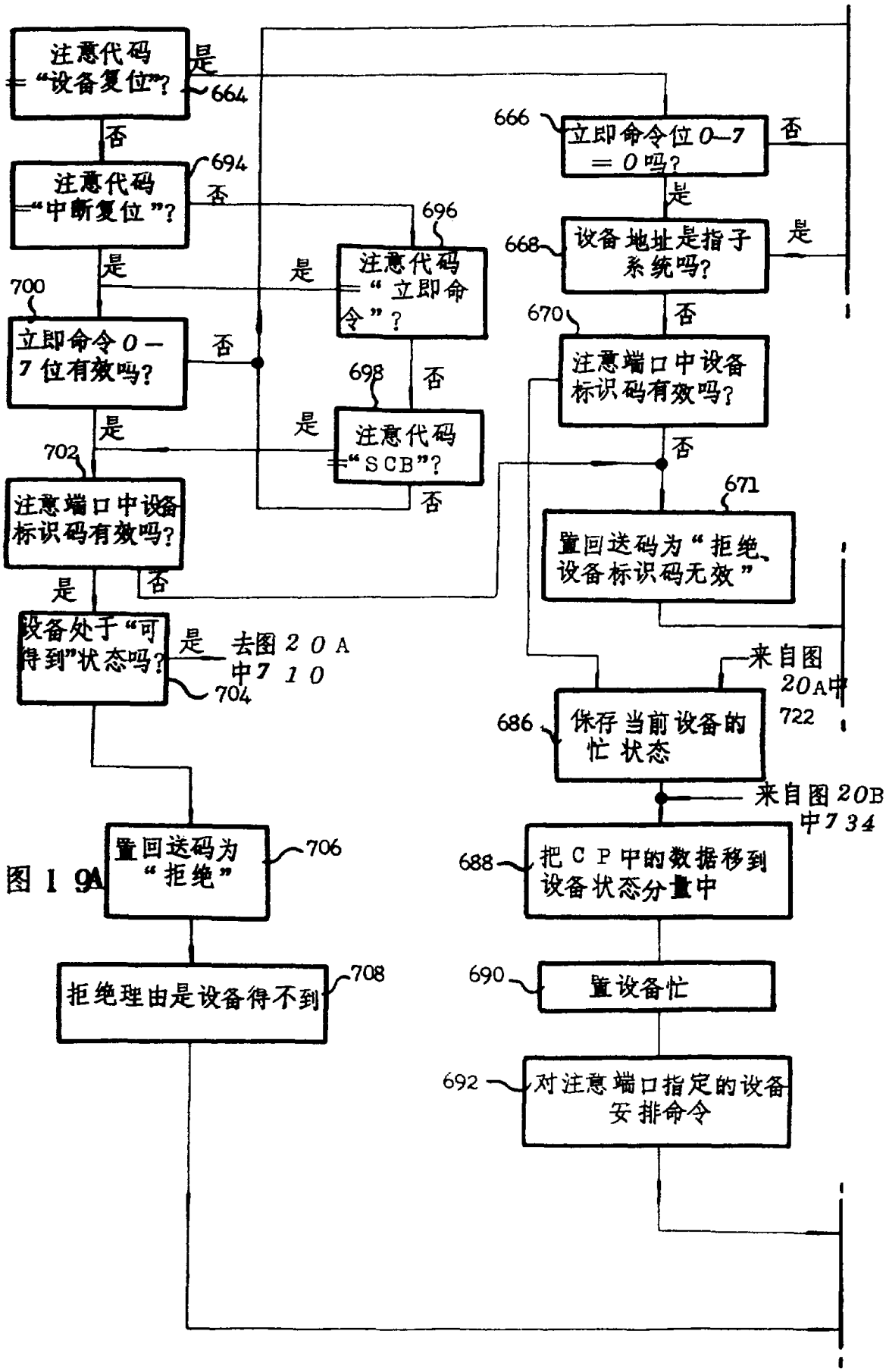

注意代码="设备复位"?
是
664
否
注意代码="中断复位"?
694
否
是
700
立即命令0—7位有效吗?
否
是
702
注意端口中设备标识码有效吗?
否
是
设备处于"可得到"状态吗?
是
去图20A中710
704
置回送码为"拒绝"
706
拒绝理由是设备得不到
708
注意代码="立即命令"?
696
是
否
注意代码="SCB"?
698
是
否
666
立即命令位0—7=0吗?
否
是
668
设备地址是指子系统吗?
是
否
670
注意端口中设备标识码有效吗?
否
671
置回送码为"拒绝、设备标识码无效"
来自图20A中722
686
保存当前设备的忙状态
来自图20B中734
688
把CP中的数据移到设备状态分量中
690
置设备忙
692
对注意端口指定的设备安排命令

图19A

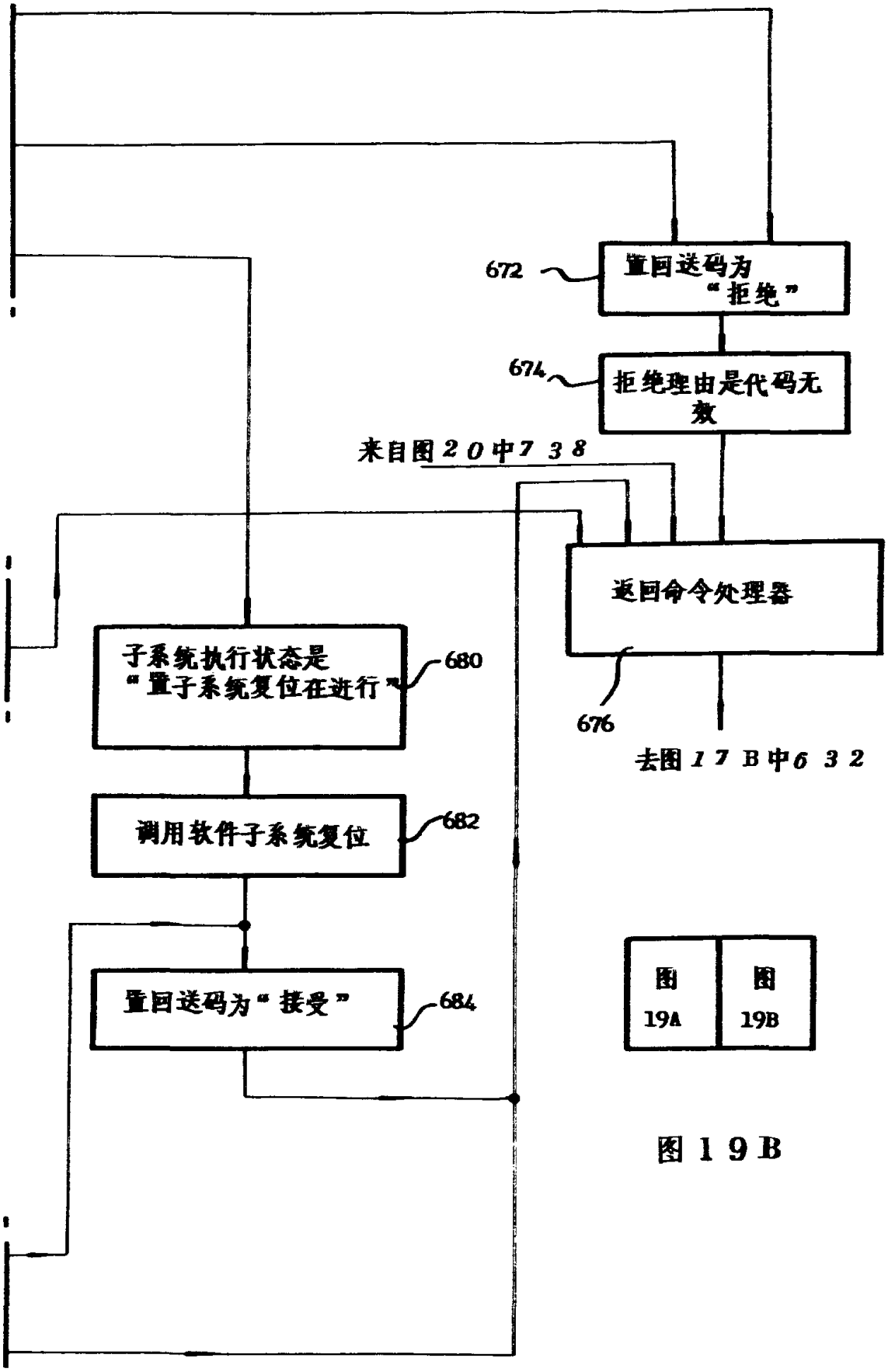


图 19A | 图 19B

图 19B

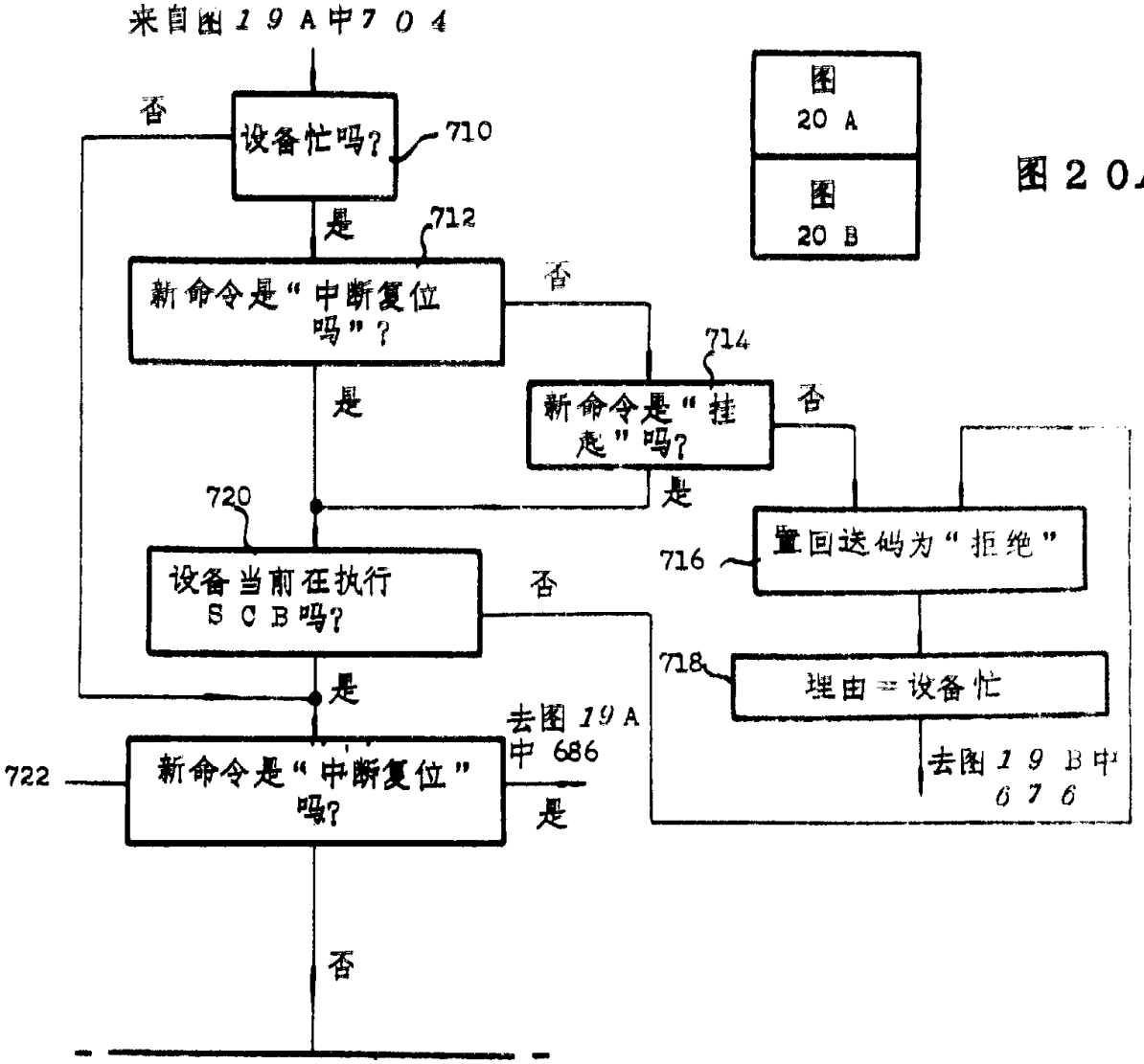

来自图19A中704
否
设备忙吗?
710
是
712
新命令是"中断复位吗"?
否
714
新命令是"挂起"吗?
否
是
是
720
设备当前在执行SCB吗?
否
716
置回送码为"拒绝"
718
理由=设备忙
是
去图19A中686
722
新命令是"中断复位"吗?
是
去图19B中676
否
图20A
图20B

图 20A

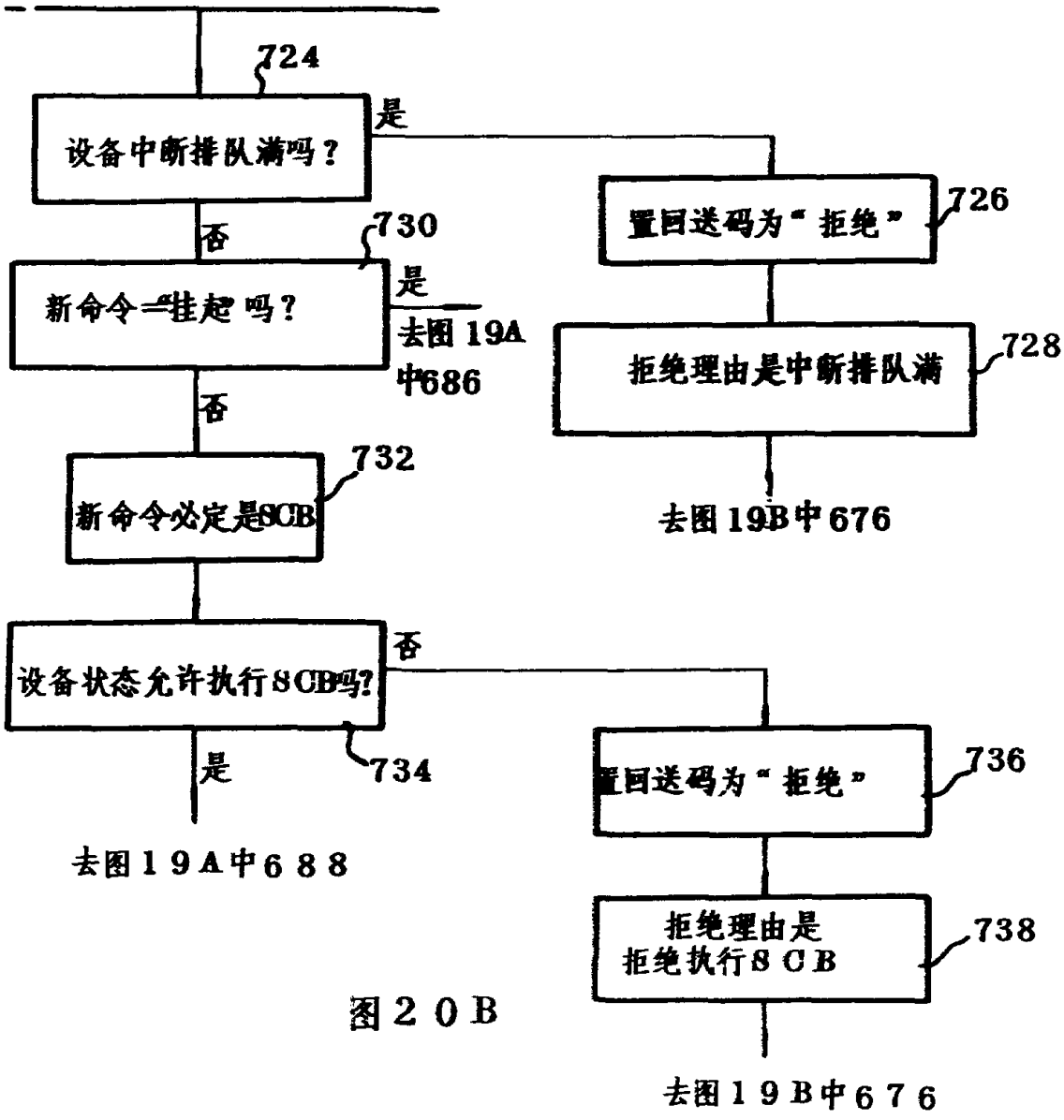

724
设备中断排队满吗？
是
否
730
新命令="挂起"吗？
是
去图19A中686
否
732
新命令必定是SCB
设备状态允许执行SCB吗?
否
是
734
去图19A中688
置回送码为"拒绝"
726
拒绝理由是中断排队满
728
去图19B中676
置回送码为"拒绝"
736
拒绝理由是
拒绝执行SCB
738
去图19B中676

图20B