

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 15/16

G06F 9/30 G06F 9/38

G06F 13/38



[12] 发明专利说明书

[21] ZL 专利号 96105644.4

[45] 授权公告日 2004 年 6 月 9 日

[11] 授权公告号 CN 1153155C

[22] 申请日 1996.4.26 [21] 申请号 96105644.4

[30] 优先权

[32] 1995.4.28 [33] JP [31] 106624/1995

[71] 专利权人 松下电器产业株式会社

地址 日本大阪府

[72] 发明人 森河彻 桧垣信生 宫地信哉

审查员 张 蕾

[74] 专利代理机构 中国专利代理(香港)有限公司

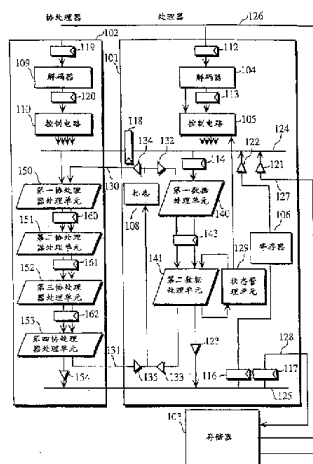
代理人 董 巍 张志醒

权利要求书 7 页 说明书 54 页 附图 11 页

[54] 发明名称 装有高效利用主处理器中的寄存器数据的协处理器的信息处理装置

[57] 摘要

一种信息处理装置，包含用于按照存储在存储器中的指令处理数据的一个主处理器及一个协处理器，它包括指令总线、第一总线、第二总线、指令检测装置、操作数标识装置、数据供给装置、数据存储器装置、协处理器指令检测装置、以及协处理器指令执行装置。



1. 一种信息处理装置，它包含按照存储器的指令处理数据的主处理器及协处理器，并包括：

5 一条指令总线，用于将指令从存储器传输到主处理器及协处理器；

一条第一总线，用于将主处理器内部的寄存器调整输出及协处理器内部的运算单元输入进行直接连接，另外将数据从主处理器传输到协处理器；

10 一条第二总线，用于将协处理器内部的运算单元输出与主处理器内部的寄存器调整输入进行直接连接，将数据从协处理器传输到主处理器；

指令检测装置，装设在主处理器中，用于通过指令总线从存储器接收的指令中检测出必须由协处理器执行的扩展运算指令；

15 操作数标识装置，装设在主处理器中，用于标识被检测到的扩展运算指令中的操作数所指定的源寄存器与目的寄存器；

20 数据供给装置，装设在主处理器中，用于通过第一总线将被标识源寄存器的数据供给协处理器；

数据存储装置，装设在主处理器中，用于通过第二总线将协处理器提供的运算结果存储在被标识的目的存储器中；

25 协处理器内的扩展指令检测装置，用于通过指令总线，从存储器接收的指令中检测出扩展运算指令；

协处理器内的指令执行装置，用于利用第一总线提供的数据执行扩展指令检测装置检测到的扩展运算指令，将执行结果提供给第二总线。

2. 权利要求1的信息处理装置，其特征在于

30 该协处理器包含一个用于暂时存储第一总线上发送的数据的输入缓冲器，以及该指令检测装置检测指示从主处理器到协处理器的数据传送的第一传送指令，

该操作数标识装置标识由检测到的第一传送指令的操作数所指定的源寄存器，

该数据供给装置将用于第一传送指令的被标识的源寄存器中的数据输出在第一总线上，

5 该协处理器扩展指令检测装置在检测到第一传送指令时，将第一总线上的数据存储在输入缓冲器中，以及

该协处理器指令执行装置在执行至少需要三个操作数的协处理器计算指令时，利用输入缓冲器中的数据。

3. 权利要求 2 的信息处理装置，其特征还在于还包括：

10 一个保存缓冲器，具有用于存储数据的区；

保存装置，装设在协处理器中，用于在接到一个中断信号时将输入缓冲器中的数据保存在保存缓冲器中；以及

返回装置，装设在协处理器中，用于在检测到从中断返回的指令时将保存缓冲器中的数据返回给输入缓冲器，

15 其中该主处理器在接收到一个中断请求时输出一个中断信号到保存装置，而协处理器扩展指令检测装置在检测到从中断返回的指令时通知返回装置。

4. 权利要求 2 的信息处理装置，其特征还在于

20 该协处理器还包含一个输出缓冲器，用于存储协处理器指令执行装置的一部分计算结果，这一部分计算结果是第二总线上的数据，

25 该协处理器扩展指令检测装置在检测到一条指示从协处理器到主处理器的数据传送的第二传送指令时，将作为协处理器指令执行装置的计算结果的一部分的输出缓冲器中的数据输出在第二总线上，

该指令检测装置检测第二传送指令，该操作数标识装置标识指定作为第二传送指令的操作数中的目的地的寄存器，以及该数据存储装置将来自第二总线的数据存储在所标识的寄存器中。

30 5. 权利要求 4 的信息处理装置，其特征还在于还包括：

一个保存缓冲器，它具有用于存储数据的区；

保存装置，装设在协处理器中，用于在接到一个中断

信号时将存储在输入缓冲器与输出缓冲器中的数据保存在保存缓冲器中；

5 返回装置，装设在协处理器中，用于在检测到一条从中断返回的指令时，将保存缓冲器中的数据返回给输入缓冲器与输出缓冲器，

其中主处理器在接到一个中断请求时输出一个中断信号到保存装置，而协处理器扩展指令检测装置在检测到一条从中断返回的指令时通知返回装置。

6. 权利要求 4 的信息处理装置，其特征在于

10 该协处理器扩展指令检测装置检测一条指令来按照标识一个任务的任务号执行任务切换，

输入缓冲器与输出缓冲器中包含多对存储区，以及

15 协处理器还包括任务管理装置，用于允许与检测到的任务切换指令中的任务号相对应的一对存储区进行数据输入与输出，而禁止所有其它存储区对的数据输入与输出。

7. 权利要求 1 的信息处理装置，其特征在于还包括：

20 主处理器阶段状态管理装置，用于管理主处理器的指令执行阶段中的周期数使之等于协处理器指令执行装置的执行周期数，其中

20 主处理器执行包含至少一个取指令阶段、一个指令解码阶段及指令执行阶段的流水线处理，而协处理器与主处理器同步操作。

25 8. 权利要求 7 的信息处理装置，其特征在于该协处理器指令执行装置在到达协处理器指令执行中的最后周期时通知主处理器阶段状态管理装置协处理器指令执行的结束，而主处理器阶段状态管理装置在接到协处理器指令执行结束通知时，便在接到该通知的执行周期中终止指令执行阶段。

30 9. 权利要求 7 的信息处理装置，其特征在于还包括主处理器确定装置，用于根据指令检测装置所检测到的协处理器计算指令的种类，确定主处理器的执行阶段中的周期数，其中该主处理器阶段状态管理装置令指令执行阶段延

续由主处理器确定装置所确定的周期数。

10. 权利要求 7 的信息处理装置，其特征在于

5 该协处理器包含一个用于暂时存储第一总线上发送的数据的输入缓冲器，以及指令检测装置检测指示从主处理器到协处理器的数据传送的第一传送指令，

操作数标识装置标识由检测到的第一传送指令的操作数所指定的源寄存器，

数据供给装置在第一总线上为第一传送指令输出被标识的源寄存器中的数据，

10 协处理器扩展指令检测装置在检测到第一传送指令时，将第一总线上的数据存储在输入缓冲器中，以及

协处理器指令执行装置在执行至少需要三个操作数的协处理器计算指令时，利用输入缓冲器中的数据。

11. 权利要求 10 的信息处理装置，其特征在于还包括：

15 一个保存缓冲器，具有用于存储数据的区；

保存装置，装设在协处理器中，用于在接到一个中断信号时，将输入缓冲器中的数据保存在保存缓冲器中；以及

20 返回装置，装设在协处理器中，用于在检测到一条从中断返回的指令时，将保存缓冲器中的数据返回给输入缓冲器，

其中该主处理器在接到一个中断请求时，输出一个中断信号给保存装置，而协处理器扩展指令检测装置在检测到从中断返回的指令时，通知返回装置。

25 12. 权利要求 10 的信息处理装置，其特征在于

该协处理器还包括一个输出缓冲器，用于存储协处理器指令执行装置的一部分计算结果，这些计算结果是第二总线上的数据，

30 协处理器扩展指令检测装置在检测到指示从协处理器到主处理器的数据传送的第二传送指令时，在第二总线上输出作为协处理器指令执行装置的一部分计算结果的输出缓冲器中的数据，

指令检测装置检测第二传送指令，操作数标识装置标识在第二传送指令的操作数中作为目的地指定的寄存器，以及数据存储装置将来自第二总线的数据存储在被标识的寄存器中。

5 13. 权利要求 12 的信息处理装置，其特征在于还包括：
一个保存缓冲器，具有用于存储数据的区；

保存装置，装设在协处理器中，用于在接到一个中断信号时，将输入缓冲器与输出缓冲器中的数据保存在保存缓冲器中；

10 返回装置，装设在协处理器中，用于在检测到从中断返回的指令时，将保存缓冲器中的数据返回给输入缓冲器与输出缓冲器，

其中该主处理器在接到一个中断请求时，输出一个中断信号给保存装置，以及协处理器扩展指令检测装置在检测到从中断返回的指令时，通知返回装置。

15 14. 权利要求 12 的信息处理装置，其特征在于

该协处理器扩展指令检测装置检测一条指令以按照标识一个任务的任务号执行任务切换，

输入缓冲器与输出缓冲器中包含多对存储区，以及

20 协处理器还包括任务管理装置，用于允许与检测到的任务切换指令中的任务号相对应的一对存储区进行数据输入与输出，而禁止所有其它存储区对的数据输入与输出。

15. 权利要求 1 的信息处理装置，其特征在于还包括：

25 一个主处理器标志寄存器，用于存储多个表示计算结果的状态的标志；

一条第一标志总线，用于将数据供给装置供给的标志传输给协处理器指令执行装置；及

一条第二标志总线，用于将来自协处理器指令执行装置的新标志传输给主处理器标志寄存器，

30 其中该数据供给装置将主处理器标志寄存器中的标志连同源寄存器中的数据供给协处理器，数据存储装置将协处理器传输的新标志存储在主处理器标志寄存器中，并将

计算结果存储在目的寄存器中，以及协处理器指令执行装置利用数据供给装置供给的标志与数据执行协处理器计算指令，然后将表示计算结果的状态的一个新标志输出到数据存储装置。

5 16. 权利要求 15 的信息处理装置，其特征在于还包括：

主处理器阶段状态管理装置，用于管理主处理器的一个指令执行阶段中的周期数，以便使之等于协处理器指令执行装置的执行周期数，其中

10 主处理器执行包括至少一个取指令阶段、一个指令解码阶段及指令执行阶段的流水线处理，而协处理器与主处理器同步操作。

17. 权利要求 16 的信息处理装置，其特征在于该协处理器指令执行装置在到达协处理器指令的执行中的最后周期时，通知主处理器阶段状态管理装置协处理器指令执行的结束，而主处理器阶段状态管理装置在接到协处理器指令执行结束的通知时，在接到通知的执行周期中终止指令执行阶段。

18. 权利要求 16 的信息处理装置，其特征在于还包括主处理器确定装置，用于按照指令检测装置检测到的协处理器计算指令的种类，确定主处理器的执行阶段中的周期数，其中该主处理器阶段状态管理装置令指令执行阶段延续由主处理器确定装置确定的周期数。

19. 权利要求 1 的信息处理装置，其特征在于还包括：

25 至少一个协处理器，各协处理器所执行的协处理器计算指令是用于不同类型的计算的，

30 其中该指令总线用于将取自存储器的指令传输给每一个协处理器中的协处理器扩展指令检测装置，第一总线用于将数据供给装置供给的数据传输给各协处理器中的协处理器指令执行装置，第二总线用于将执行协处理计算指令的一个协处理器中的协处理器指令执行装置的计算结果传输给数据存储装置，以及指令检测装置为各协处理器检测协处理器指令。

20. 权利要求 19 的信息处理装置，其特征在于还包括：

主处理器阶段状态管理装置，用于管理主处理器的一个指令执行阶段中的周期数，以便使之等于执行协处理器指令的协处理器的协处理器指令执行装置的执行周期数，
5 其中

主处理器执行包含至少一个取指令阶段、一个指令解码阶段及指令执行阶段的流水线处理，而各协处理器与主处理器同步操作。

21. 权利要求 20 的信息处理装置，其特征在于执行协处理器指令的一个协处理器中的协处理器指令执行装置在到达协处理器指令的执行中的最后一个周期时，通知主处理器阶段状态管理装置协处理器指令执行的结束，而主处理器阶段状态管理装置在接到协处理器指令执行结束的通知时，在接到通知的执行周期中终止指令执行阶段。
10

22. 权利要求 20 的信息处理装置，其特征在于还包括主处理器确定装置，用于根据指令检测装置检测到的协处理器计算指令的种类，确定主处理器的执行阶段中的周期数，其中该主处理器阶段状态管理装置令指令执行阶段延续由主处理器确定装置确定的周期数。
15

20

装有高效利用主处理器中的
寄存器数据的
协处理器的信息处理装置

本发明涉及信息处理装置，该装置由按照存储在存储器中的指令处理数据的一个主数据处理装置（主处理器）及一个从属的扩展处理装置（协处理器）构成。

近年来，通过在数据处理装置（此后称作“处理器”）上装设按照需要高速执行特殊计算的扩展计算装置（此后称作“协处理器”）而使处理能力有了极大的提高。

装设这种协处理器的主要传统方法包括在主处理器内部安装协处理器及根据需要将协处理器连接在主处理器上。在这两种方法中，前一装设方法具有在不使用协处理器时增加费用与功耗的缺点，因此希望用一种方法连接一个与所装设的协处理器性能相等的协处理器。

包含连接在主处理器上的协处理器的传统信息处理装置动作以便协处理器计算通过处理器为协处理器解码指令，然后通知协处理器解码后的命令。这一方法的缺点在于从处理器传送命令产生可观的开销，而使协处理器采用太长的处理时间。

根据这一问题，日本特许公开专利申请 1 - 240932 公开了一种信息处理装置，该装置通过令协处理器与执行流水线处理的处理器同时取出与解码同一条指令而达到命令传送与解码所需的时间的缩短。

然而，当这种传统信息处理装置中的协处理器利用保

持在主处理器的通用寄存器中的数据执行处理，或者当协处理器所处理的数据是存储在主处理器的通用寄存器中时，数据首先必须暂时存储在存储器中，将数据传送到存储器中要增加执行时间，从而降低系统性能。

类似地，当这种传统信息处理装置中的协处理器利用保持在主处理器中的标志存储寄存器中的标志信息执行处理，或协处理器所处理的标志信息存储在主处理器中的标志存储寄存器中时，必须首先将标志信息暂时存储在存储器中，将标志信息传送到存储器中要增加执行时间，从而降低系统性能。

同时，当协处理器所处理的数据与标志信息是存储在主处理器中的标志存储寄存器与通用寄存器中时，主处理器必须封锁流水线处理直到完成通用寄存器或标志存储寄存器中数据与标志的存储，这便增加了执行时间并从而降低了系统性能。

类似地，当在协处理器的处理期间产生中断过程时，由于不能存储当前正在处理的数据，因而在完成了中断过程之后协处理器必须重新执行数据处理。这便增加了处理时间，从而降低了系统性能。

最后，当在协处理器的处理期间产生任务切换过程时，由于当前正在处理的数据可能存储，因此协处理器必须重新执行任务切换前所执行的任务过程的处理。这便增加了执行时间并从而降低了系统性能。

本发明的第一目的为提供一种信息处理装置，该装置在使用主处理器的通用寄存器中的数据执行协处理器指令时，或将协处理器所处理的数据存储在主处理器的通用寄存器中时，并不导致执行时间的增加。

本发明的第二目的为提供一种信息处理装置，对于该装置可以根据需要方便地增加或拆除用于不同种类计算的多个协处理器。

本发明的第三目的为提供一种信息处理装置，该装置在使用主处理器的通信寄存器中的数据执行协处理器指令，或将协处理器所处理的数据存储在主处理器的通用寄存器中时，能高速执行处理而不打乱主处理器的流水线处理。

本发明的第四目的为提供一种信息处理装置，该装置在使用主处理器的标志存储寄存器中的标志信息执行协处理器指令或将协处理器所处理的标志信息存储在主处理器的通用寄存器中时，并不导致处理时间的增加及性能降低。

本发明的第五目的为提供一种信息处理装置，该装置能在协处理器使用多个操作数执行处理时抑制由于增加的电路规模或功耗引起的费用增加。

本发明的第六目的为提供一种信息处理装置，该装置能在协处理器处理期间出现中断过程时抑制执行时间增加及性能降低。

本发明的第七目的为提供一种信息处理装置，该装置能在协处理器处理期间出现任务切换过程时抑制执行时间增加及性能降低。

本发明的第一目的信息处理装置包括按照存储在存储器中的指令处理数据的一个主处理器及一个协处理器，所述信息处理装置包括：一条指令总线，用于将指令从存储器传输到主处理器及协处理器；一条第一总线，用于将数据从主处理器传输到协处理器；一条第二总线，用

于将数据从协处理器传输到主处理器；一个指令检测单元，装设在主处理器中，用于从经由指令总线自存储器接收的所有指令中检测出必须由协处理器执行的协处理器计算指令；一个操作数标识单元，装设在主处理器中，用于标识由指令检测单元检测到的指令中的操作数所指定的源寄存器与目的寄存器；一个数据供给单元，装设在主处理器中，用于通过第一总线将数据从所标识的源寄存器供给协处理器；一个数据存储单元，装设在主处理器中，用于通过第二总线将协处理器供给的计算结果存储在所标识的目的寄存器中；一个协处理器指令检测单元，装设在协处理器中，用于在通过指令总线从存储器接收的所有指令中检测出协处理器计算指令；以及一个协处理器指令执行单元，装设在协处理器中，用于使用第一总线供给的数据执行协处理器指令检测单元所检测到的协处理器计算指令，并将计算结果提供在第二总线上。

利用上述结构，由于数据能用设置在处理器与协处理器之间的第一与第二总线直接传送，所以在协处理器用主处理器的数据存储单元中的数据计算时，或将协处理器的计算结果存储在数据存储单元中时，不再需要通过存储器执行数据传输。这种结构便能实现高处理性能的信息处理装置。

同时，该信息处理装置还可包括：至少一个协处理器，各协处理器所执行的协处理器指令为不同类型的计算，其中的指令总线可用于将取自存储器的指令传输到每一协处理器中的协处理器指令检测单元，第一总线可用于将数据供给单元所供给的数据传输给各协处理器中的协处理器指令执行单元，第二总线可用于将执行该协处理器计算

指令的协处理器中的协处理器指令执行单元的计算结果传输给数据存储单元，以及指令检测单元可为各协处理器检测协处理器指令。

上述结构能够实现本发明的第二目的。这便是说，通过同时连接多个协处理器，便能高速地执行包含各种协处理器指令的程序。并且可根据需要做到方便地添加或拆除协处理器。在本例中，无论多少协处理器，主处理器的内部布线的电负荷不变，从而能自由地选择协处理器的不同配置。尤其是，这使得设计由单个芯片实现的信息处理装置非常容易。

同时，该信息处理装置还可包括：一个主处理器阶段状态管理单元，用于管理主处理器的指令执行阶段中的周期数目使之等于协处理器指令执行单元的周期数目，其中该主处理可执行包含至少一个取指令阶段、一个指令解码阶段及指令执行阶段的流水线处理，而协处理器可与主处理器同步操作。

上述结构能够实现本发明的第三目的。这便是说，通过管理主处理器与协处理器使它们的执行阶段的周期数相等，便能避免主处理器的流水线处理中的混乱。从而达到协处理器指令执行与主处理器指令执行之间的平滑转移。

同时，一旦协处理器指令执行单元在执行协处理器指令期间到达最终周期时，可通知主处理器阶段状态管理单元协处理器指令执行的结束，并且主处理器阶段状态管理单元在接收到协处理器指令执行的结束通知时，可在接收到通知的执行周期中终止指令执行阶段。

利用上述结构，设置一条信号线来传输执行完成信号

便能避免流水线处理中的混乱，并利用一个简单的结构来达到对协处理器指令执行的平滑转移。

同时，该信息处理装置还可包括一个主处理器确定单元，用于根据指令检测单元所检测到的协处理器计算指令的种类确定主处理器的执行阶段中的周期数，其中该主处理器阶段状态管理单元可令指令执行阶段延续由主处理器确定单元所确定的周期数。

利用上述结构，不再需要在主处理器与协处理器之间设置一条信号线来通知协处理器指令执行的结束，从而能够达到有利于连接多个协处理器的连接方法。

同时，该信息处理装置还可包括：一个主处理器标志寄存器，用于存储表示计算结果的状态的多个标志；一条第一标志总线，用于将数据供给单元所供给的标志传输给协处理器指令执行单元；以及一条第二标志总线，用于将来自协处理器指令执行单元的新标志传输给主处理器标志寄存器，其中该数据供给单元可将主处理器标志寄存器中的标志连同源寄存器中的数据一起提供给协处理器，数据存储单元可将协处理器所传输的新标志存储在主处理器标志寄存器中，并将计算结果存储在目的寄存器中，并且协处理器指令执行单元在将表示计算结果的状态的新标志输出到数据存储单元之前，用数据供给单元所供给的标志与数据执行协处理器计算指令。

上述结构能够实现本发明的第四目的。这便是说，通过与操作数数据相同的方式经由第一与第二标志总线在主处理器与协处理器之间直接传输多个标志的标志信息，便能得到高处理性能的信息处理装置。

同时，该协处理器可包含用于暂时存储第一总线上所

发送的数据的一个输入缓冲器，并且指令检测单元可检测表示从主处理器到协处理器的数据传送的第一传送指令，操作数标识单元可标识由检测到的第一传送指令的操作数所指定的源寄存器，数据供给单元可在第一总线上为第一传送指令输出所标识的源寄存器中的数据，协处理器指令检测单元在检测到第一传送指令时，可将第一总线上的数据存储在输入缓冲器中，而协处理器指令执行单元在执行至少需要三个操作数的协处理器计算指令中，可利用输入缓冲器中的数据。

上述结构能够实现本发明的第五目的。这便是说，能够实现有利于执行需要三个或以上操作数的协处理器指令的信息处理装置。并且，与利用为此目的设置的总线来提供多个操作数数据的装置相比，能够抑制由于增大电路规模及增加功耗引起的费用增加。

同时，该信息处理装置还可包括：具有用于存储数据的区域的一个保存缓冲器；一个保存单元，装设在协处理器中，用于在接收到一个中断信号时，将输入缓冲器与输出缓冲器中的数据保存在保存缓冲器中；以及一个返回单元，装设在协处理器中，用于在检测到从中断返回的指令时，将保存缓冲器中的数据返回给输入缓冲器与输出缓冲器，其中该处理器在接收到一个中断请求时可输出一个中断信号给保存单元，而协处理器指令检测单元在检测到从中断返回的指令时可通知返回单元。

上述结构能够实现本发明的第六目的。这便是说，当在协处理器处理中产生中断过程时，能够抑制执行时间的增加及性能降低，从而本信息处理装置能够快速转移到中断过程。

同时，该协处理器指令检测单元可检测一条根据标识一个任务的任务号执行任务切换的指令，输入缓冲器与输出缓冲器可包含多个存储器时，并且协处理器还可包含一个任务管理单元，用于允许对应于所检测到的任务切换指令中的任务号的一对存储区的数据输入与输出，及用于禁止对所有其它存储区对的数据输入与输出。

上述结构能够实现本发明的第七目的。这便是说，在执行多个任务时，上述结构利用输入缓冲器与输出缓冲器中与各任务对应的一对存储器，从而在出现任务切换过程时，能够抑制执行时间的增加及性能降低，从而实现高速的任务切换。

从下面结合展示本发明的一个特定实施例的附图所作的描述中，本发明的上述及其它目的、优点与特征将是显而易见的。附图中：

图1为展示本发明的第一实施例的信息处理装置的构造的方框图；

图2为示出用在该实施例中的协处理器102中的协处理器指令的实例的图；

图3为展示该实施例的信息处理装置的流水线处理的状态的时间图；

图4为展示本发明的第二实施例的信息处理装置的构造的方框图；

图5为展示当将通用寄存器106所保持的操作数写入该实施例的信息处理装置的数据处理单元输入缓冲器230时，协处理器202的流水线处理的状态的时间图；

图6为展示当利用存储在该实施例的信息处理装置的主处理器的通用寄存器106中的操作数及存储在数据处

理单元输入缓冲器 230 中的操作数的输出结果写入主处理器的通用寄存器 106 与数据处理单元输出缓冲器 231 时, 协处理器处理单元 211 的指令执行流水线处理的状态的时间图;

图 7 为展示将存储在该实施例的信息处理装置的数据处理单元输出缓冲器 231 中的数据写入主处理器的通用寄存器 106 的指令的流水线处理的时间图;

图 8 为展示发生对主处理器 201 的一个中断及将本发明的第二实施例的信息处理装置的协处理器的数据处理单元输入缓冲器 230 中的数据及协处理器的数据处理单元输出缓冲器 231 中的数据保存在保存数据存储缓冲器 232 中的操作的时间图;

图 9 为展示本发明的第二实施例的信息处理装置的中断返回指令的流水线处理的状态的时间图;

图 10 为展示本发明的第三实施例的信息处理装置的构造的方框图; 以及

图 11 为展示本发明的信息处理装置的 LSI (大规模集成) 配置的实例的图。

第一实施例

图 1 为展示本发明的第一实施例的信息处理装置的构造的方框图。可以看出该信息处理装置是由数据处理单元输入总线 124、数据处理单元输出总线 125、指令供给总线 126、数据总线 127、数据总线 128、标志数据总线 130 及标志数据总线 131 互连的处理器 101、协处理器 102 和存储器 103 构成的, 如图中所示。

处理器 101 由解码器 104、控制电路 105、通用寄存器 106、标志存储寄存器 108、指令寄存器 112、微指令

寄存器 113、数据锁存器 114、寄存器写缓冲器 116、存储器写缓冲器 117、协处理器数据锁存器 118、状态管理单元 129、流水线锁存器 143、驱动器 121 - 123 与 132 - 135、第一数据处理单元 140 与第二数据处理单元 141 组成。处理器 101 利用由取指令阶段、指令解码阶段、第一执行阶段、第二执行阶段及写阶段构成的五阶段流水线过程来处理存储器 103 中的指令。存储器 103 中的指令中包括处理器 101 的指令（此后称作“处理器指令”）及协处理器 102 的指令（此后称作“协处理器指令”）。当处理器 101 在指令解码阶段解码一条协处理器指令时，它从通用寄存器 106 中检索出指令中的操作数所指定的源数据并将这一数据供给协处理器 102，而协处理器 102 则将其计算结果存储在通用寄存器 106 中。这里将处理器 101 设定为 32 位处理器，并且除非明确说明，总线宽度是设定为 32 位的。

协处理器 102 由解码器 109、控制电路 110、指令寄存器 119、微指令寄存器 120、第一至第三流水线锁存器 160 - 162、第一至第四协处理器处理单元 150 - 153 及第一至第三流水线锁存器 160 - 162 组成。协处理器 102 是多个协处理器之一，它执行诸如乘、除、乘积和、位字段计算、字符串计算、饱和计算、排队计算、上下文切换过程与过滤过程等协处理器过程，并且包括取决于使用能支持所要求的各种计算的其它协处理器。在本实施例中，所要求的计算种类设定为乘法，这是利用由一个指令解码阶段及第一至第四执行阶段构成的流水线过程执行的。这里，表示乘法的协处理器指令的实例示出在图 2 中。在该图中，“MULQ Dm Dn”表示源寄存器 Dm（带有附加

代码的 32 位整数)与源寄存器 D_n (带有附加代码的 32 位整数)的相乘,将结果(64 位数)的低 32 位存储在目的寄存器 D_n 中,并将结果(64 位数)的高 32 位存储在通用寄存器 106 中的乘法寄存器 MR 中。在同一图中,“ $MULQ\ imm\ 8,\ D_n$ ”表示在应用代码的数据上扩充 8 位直接数据所给出的 32 位值与源寄存器 D_n (带有附加代码的 32 位整数)的乘法,将结果存储在 D_n 与 MR 中。“ $MULQ\ imm\ 16,\ D_n$ ”与“ $MULQ\ imm\ 32,\ D_n$ ”也一样。

存储器 103 中存储由处理器与协处理器指令构成的程序及处理器 101 与协处理器 102 所使用的数据。

解码器 104 在指令解码阶段中解码通过指令供给总线 126 与指令寄存器 112 从存储器 103 取出的处理器指令与协处理器指令,并输出微指令供实现所取出的指令。更具体地,在解码一条协处理器指令时(例如在图 2 中,当指令码的高四位为“FH”时),解码器 104 发布(1)一条读取由协处理器指令中的操作数所指示的源寄存器的微指令,(2)一条将读自源寄存器的数据锁存在协处理器数据锁存器 118 中的微指令,(3)一条令标志存储寄存器 108 中的标志数据通过驱动器 134 输出到标志数据总线 130 上的微指令,(4)一条向第一数据处理单元 140 及第二数据处理单元 141 指示 NOP (空操作)的微指令,(5)一条将协处理器 102 的计算结果存储在由一个操作数指示的目的寄存器中的微指令,以及(6)一条将来自协处理器 102 的标志数据通过驱动器 135 存储在标志存储寄存器 108 中的微指令。在这些微指令中,前面三条是不通过微指令寄存器 113 直接输出到控制电路 105 的,而其

余的微指令则是通过微指令寄存器 113 输出到控制电路 105 的。从而，在指令解码阶段中，首先执行前三条微指令。应指出，这些微指令可设置成上面（1）至（6）中所示，也可设置成使一条微指令由若干条微指令构成。

解码器 104 还根据协处理器指令的类型设置一个延续数，并将其输出到状态管理单元 129。这里，延续数表示处理器 101 的执行阶段数与协处理器 102 的执行阶段数之差。从而，这表示在协处理器的指令执行期间必须加在处理器 101 上的空闲状态（NOP）执行阶段的数目。在本实施例中，协处理器 102 的执行阶段数与处理器 101 的执行阶段数分别为 4 与 2，因此当解码器 104 确定解码结果为协处理器指令是乘法指令（对于图 2 中的实例，当指令码中的高 8 位为 F6H、F9H 与 FBH）时，解码器 104 便将延续数设定在 2 上。

处理器 101 中的控制电路 105 将直接从解码器 104 输入的或者通过微指令寄存器 113 间接从解码器 104 输入的微指令转换成各种控制信号，以及作为控制流水线处理的电路。当输入了作为一条协处理器指令的解码结果的上述指令（1）、（2）与（3）时，控制电路 105 输出用于通用寄存器 106 中所指示的寄存器的一个读信号、用于启动驱动器 122 的一个控制信号、用于令协处理器数据锁存器 118 执行锁存的一个控制信号、以及用于在所取出的指令的执行阶段前面的解码阶段中启动驱动器 134 的一个控制信号。以相同的方式，在输入指令（5）与（6）时，控制电路 105 输出一个用于令通用寄存器 106 执行锁存的控制信号，用于由通用寄存器 106 指定的寄存器的写信号、以及用于启动驱动器 135 的一个控制信号。

通用寄存器 106 由各保持 32 位数据的寄存器 D0 - D3 及保持乘积的高 32 位的乘法寄存器 MR 构成。

指令寄存器 112 受控制电路 105 控制以便将一条通过指令供给总线 126 从存储器 103 取出的指令输出到解码器 104。每次从取指令阶段切换到指令解码阶段时，这一输出便切换到下一条指令。

标志存储寄存器 108 存储表示处理器状态的各种标志，处理器 101 的第二数据处理单元 141 所生成的标志及协处理器 102 的第四协处理器处理单元 153 所生成标志两者都由控制电路 105 所进行的控制反映。

微指令寄存器 113 根据控制电路 105 的控制将来自解码器 104 的微指令输出到控制电路 105。每次从指令解码阶段切换到第一执行阶段时，这一输出便切换到下一条指令（或指令组）。

数据锁存器 114 按照控制电路 105 的控制，通过数据处理单元输入总线 124 将从通用寄存器 106 读取的操作数数据输出到第一数据处理单元 140。每次从指令解码阶段切换到第一执行阶段时，这一输出便切换到下一条指令。

寄存器写缓冲器 116 按照控制电路 105 的控制在完成第二执行阶段之后将来自数据处理单元输出总线 125 的数据输出到通用寄存器 106。

存储器写缓冲器 117 按照控制电路 105 的控制在完成第二执行阶段之后将数据从数据处理单元输出总线 125 输出到数据总线 128 上。

协处理器数据锁存器 118 在从指令解码阶段切换到第一执行阶段时，按照控制电路 105 的控制，通过数据处理单元输入总线 124 将寄存器数据输出到协处理器 102。

驱动器 121 具有一个按照控制电路 105 的控制开启的门, 并且是用于通过数据总线 127 在数据处理单元输入总线 124 上传输存储器 103 中的数据的驱动器。

驱动器 122 具有一个按照控制电路 105 的控制开启的门, 并且是用于在数据处理单元输入总线 124 上传输来自通用寄存器 106 的数据的驱动器。

驱动器 123 具有一个按照控制电路 105 的控制开启的门, 并且是用于在数据处理单元输出总线 125 上传输来自第二数据处理单元 141 的数据的驱动器。

数据处理单元输入总线 124 用于将来自处理器 101 内部的通用寄存器 106 及来自存储器 103 的数据传输给第一数据处理单元 140 及协处理器数据锁存器 118, 并且将数据从协处理器数据锁存器 118 传输到协处理器 102 中的第一协处理器处理单元 150。应特别注意, 数据处理单元输入总线 124 通过协处理器数据锁存器 118 将处理器 101 的通用寄存器 106 中的数据传输给协处理器 102 中的第一协处理器处理单元 150。在本实施例中, 该总线具有两个 32 位的宽度。

数据处理单元输出总线 125 用于将来自处理器 101 中的第二数据处理单元 141 及协处理器 102 中的第四协处理器处理单元 153 的执行结果传输给处理器 101 中的寄存器写缓冲器 116 与存储器写缓冲器 117。注意, 数据处理单元输出总线 125 是用于将处理器 101 的第二数据处理单元 141 及协处理器 102 的第四协处理器处理单元 153 处理过的数据传输给通用寄存器 106 与存储器 103 的总线。在本实施例中, 该总线具有两个 32 位宽度。

指令供给总线 126 是用于将存储在存储器 103 中的指

令传输给处理器 101 中的指令寄存器 112 及协处理器 102 中的指令寄存器 119 的总线。

数据总线 127 通过驱动器 121 在数据处理单元输入总线 124 上传输存储在存储器 103 中的数据。

数据总线 128 通过存储器写缓冲器 117 将来自数据处理单元输出总线 125 的数据传输给存储器 103。

状态管理单元 129 管理处理器 101 的流水线阶段的所有状态，并管理是否前进到当前阶段后面的周期中下一过程或者是否延续当前阶段。更具体地，状态管理单元 129 存储表示流水线阶段的状态的标志信息。这一标志信息表示指示前进到当前阶段后面的周期中的下一过程的“下一阶段执行状态”或者指示在下一周期中继续当前阶段的“当前阶段延续状态”。当解码器 104 解码一条处理器指令时，对于在由第一执行阶段及第二执行阶段构成的两个阶段中完成的指令执行，该标志信息显示“下一阶段执行状态”。当解码器 104 解码一条协处理器指令时，解码器 104 通知状态管理单元 129 延续数，在再下一个执行阶段中，标志信息显示“当前阶段延续状态”等于这一延续数的若干阶段。例如，当接到通知延续数 2 时，标志信息在显示“当前阶段延续状态”之前在第一执行阶段中显示“下一阶段执行状态”，然后“当前阶段延续状态”，然后在第二执行阶段中显示“下一阶段执行状态”。这样做，状态管理单元 129 管理处理器 101 的执行阶段数，使之与协处理器 102 的执行阶段数相等。

标志数据总线 130 是用于将标志存储寄存器 108 中的标志数据传输给协处理器 102 的总线。这一总线的位宽度可按照标志的数目设定。

标志数据总线 131 是用于将第四协处理器处理单元 153 处理过的标志数据传输给处理器 101 中的标志存储寄存器 108 的总线。这一总线的位宽度可按照标志的数目设定。

驱动器 132 具有一个按照控制电路 105 的控制开启的门，并且是用于将标志数据从标志存储寄存器 108 传输到第一数据处理单元 140 的驱动器。

驱动器 133 具有一个按照控制电路 105 的控制开启的门，并且是用于将标志数据从第二数据处理单元 141 传输到标志存储寄存器 108 的驱动器。

驱动器 134 具有一个按照控制电路 105 的控制开启的门，并且是用于将标志数据从标志存储寄存器 108 传输到协处理器 102 的驱动器。

驱动器 135 具有一个按照控制电路 105 的控制开启的门，并且是用于将标志数据从协处理器 102 传输到标志存储寄存器 108 的驱动器。

第一数据处理单元 140 按照控制电路 105 的控制，在处理器 101 中的第一执行阶段期间执行数据处理。当解码器 104 解码一条协处理器指令时，按照控制电路 105 的控制将第一数据处理单元 140 设定为空闲状态 (NOP)。

第二数据处理单元 141 在处理器 101 的第二执行阶段期间执行数据处理。当解码器 104 解码一条协处理器指令时，按照控制电路 105 的控制将第二数据处理单元 141 设定为空闲状态 (NOP)。

当在执行流水线控制中按照控制电路 105 的控制，从第一执行阶段切换到第二执行阶段并从第一数据处理单元 140 转移到第二数据处理单元 141 时，流水线锁存器 143

打开其锁存器。

至于图 1 中的协处理器 102，在指令解码阶段期间，解码器 109 解码通过指令供给总线 126 与指令寄存器 112 从存储器 103 取出的指令。当解码结果为一条协处理器指令时，解码器 109 便发布一条微指令来实现由该协处理器指令的 OP 码所示的计算内容。应指出，没有必要发布一条微指令来指示用于读与写协处理器指令中的操作数数据的寄存器。这是因为操作数寄存器的读与写操作是由处理器 101 执行的。

处理器中的控制电路 110 将通过微指令寄存器 120 从解码器 109 输入的微指令转换成各种控制信号，以及作为控制协处理器 102 的流水线处理的电路。

指令寄存器 119 将通过指令供给总线 126 从存储器 103 取出的指令输出给解码器 109。每次从取指令阶段切换到指令解码阶段时，这一输出便切换到下一条指令。

微指令寄存器 120 按照执行流水线控制中的控制电路 110 的控制，在从指令解码阶段切换到第一执行阶段时，将微指令从解码器 109 输出到控制电路 110。每次从指令解码阶段切换到第一执行阶段时，这一输出便切换到下一条指令（或指令组）。

第一协处理器处理单元 150 执行协处理器 102 的第一执行阶段中的数据处理。

第二协处理器处理单元 151 执行协处理器 102 的第二执行阶段中的数据处理。

第三协处理器处理单元 152 执行协处理器 102 的第三执行阶段中的数据处理。

第四协处理器处理单元 153 执行协处理器 102 的第四

执行阶段中的数据处理。

第一流水线锁存器 160，在按照执行流水线控制中的控制电路 110 的控制从第一执行阶段切换到第二执行阶段并从第一协处理器处理单元 150 转移到第二协处理器处理单元 151 时，打开其锁存器。

第二流水线锁存器 161，在按照执行流水线控制中的控制电路 110 的控制从第二执行阶段切换到第三执行阶段并从第二协处理器处理单元 151 转移到第三协处理器处理单元 152 时，打开其锁存器。

第三流水线锁存器 162，在按照执行流水线控制中的控制电路 110 的控制从第三执行阶段切换到第四执行阶段并从第三协处理器处理单元 152 转移到第四协处理器处理单元 153 时，打开其锁存器。

下面是上述构造的本发明的第一实施例的信息处理装置的说明。

图 3 为展示流水线处理的状态的时间图，这时对于协处理器 102，处理来自处理器 101 的通用寄存器 106 及标志存储寄存器 108 的数据的操作，然后将处理结果数据存储在通用寄存器 106 与标志存储寄存器 108 中。

在该图中，“IF1”为处理器 101 检索一条协处理器指令的阶段。“OP1”为处理器 101 解码一条协处理器指令的阶段。“EXP1”与“EXP2”分别为处理器 101 中一条协处理器指令的第一与第二执行阶段（或 NOP）。

“WB1”为处理器 101 写协处理器 102 的计算结果的阶段。“DE1”为协处理器 102 的解码一条协处理器指令的阶段。“EXE1”、“EXE2”、“EXE3”与“EXE4”分别为协处理器 102 中一条协处理器指令的第一、第二、

第三与第四执行阶段。“NXT - IF”为处理器 101 取出“IF1”所取的指令后面的一条指令的阶段。“NXT - DP”为处理器 101 解码下一条指令的阶段。“NXT - EXP1”与“NXT - EXP2”分别为处理器 101 中下一条指令的第一与第二执行阶段。

在时间间隔 1 中，存储器 103 在处理器 101 的取指令阶段（IF1）中将一条指令输出到导向处理器 101 与协处理器 102 的指令供给总线 126 上。

从时间间隔 1 切换到时间间隔 2 中，处理器 101 中的控制电路 105 从（IF1）前进到（DP1）并相应地打开处理器 101 的指令寄存器 112。以相同的方式，协处理器 102 中的控制电路 110 从（IF1）前进到（DE1）并相应地打开协处理器 102 的指令寄存器 119。

在时间间隔 2 中，存储器 103 在取指令阶段（NXT - IF1）中将下一条指令输出到导向处理器 101 与协处理器 102 的指令供给总线 126 上。

在指令解码阶段（DP1）中，在指令寄存器 112 打开以后，处理器 101 中的解码器 104 接收指令，然后解码指令。然后，控制电路 105 利用解码器 104 输出的微指令，启动驱动器 122 将通用寄存器 106 中的数据放置在数据处理单元输入总线 124 上，并启动驱动器 134 将标志存储寄存器 108 中的标志信息放置在标志数据总线 130 上。

在协处理器 102 的指令解码阶段（DE1）中，协处理器 102 中的解码器 109 在指令寄存器 119 打开之后接收指令，然后解码该指令。

从时间间隔 2 到时间间隔 3 的切换中，处理器 101 中的控制电路 105 从（NXT - IF）前进到（NXT - DP）

并相应地打开指令寄存器 112。

处理器 101 中的控制电路 105，从流水线阶段 (DP1) 前进到 (EXP1) 中打开微指令寄存器 113、数据锁存器 114 与协处理器数据锁存器 118。类似地，协处理器 110 中的控制电路 110 从流水线阶段 (DE1) 前进到 (EXE1) 中打开微指令寄存器 120。一旦打开了微指令寄存器 113，处理器 101 中的控制电路 105 接收指令解码阶段 (DP1) 中解码器 104 所解码的微指令。类似地，一旦打开了微指令寄存器 120，协处理器 102 中的控制电路 110 接收指令解码阶段 (DE1) 中协处理器 102 的解码器 109 所解码的微指令。

在时间间隔 3 中，在打开指令寄存器 112 后，处理器 101 的解码器 104 接收指令，然后在处理器 101 的指令解码阶段 (NXT - DP) 中解码该指令。

在处理器 101 的第一执行阶段 (EXP1) 中，处理器 101 中的第一数据处理单元 140 并不执行数据处理 (NOP)。

在协处理器 102 的第一执行阶段 (EXE1) 中，协处理器 102 中的第一协处理器处理单元 150 执行数据处理。

从时间间隔 3 切换到时间间隔 4 中，处理器 101 的控制电路 105 从 (NXT - DP) 到 (NXT - EXP) 的流水线阶段切换中打开微指令寄存器 113。打开了微指令寄存器 113 之后，处理器 101 的控制电路 105 便接收 (NXT - DP) 指令解码阶段中解码器 104 所解码的微指令。

从 (EXP1) 到 (EXP2) 的流水线阶段切换中，处理器 101 的控制电路 105 打开处理器 101 的流水线锁存器 143。

从 (EXE1) 到 (EXE2) 的流水线阶段切换中, 协处理器 102 的控制电路 110 打开协处理器 102 的第一流水线锁存器 160 .

在时间间隔 4 中, 处理器 101 中的第一数据处理单元 140 执行处理器 101 的第一指令执行阶段 (NXT - EXP1) 中下一条指令的第一数据处理.

在处理器 101 的第二指令执行阶段 (EXP2) 中, 第二数据处理单元 141 并不执行数据处理 (NOP) .

状态管理单元 129 从下一阶段执行状态改变到当前状态延续状态, 并传输一个当前阶段延续信号给控制电路 105 .

在协处理器 102 的第二指令执行阶段 (EXE2) 中, 协处理器 102 的第二协处理器处理单元 151 执行数据处理.

从时间间隔 4 切换到时间间隔 5 中, 由于状态管理单元 129 发送当前状态延续信号, 处理器 101 中的控制电路 105 并不打开流水线锁存器 143 . 从而, 当前阶段的状态延续到下一阶段.

接着, 从流水线阶段 (EXE2) 切换到 (EXE3) 中, 协处理器 102 中的控制电路 110 打开第二流水线锁存器 161 .

在时间间隔 5 中, 为处理器 101 延续第一指令执行阶段 (NXT - EXP1), 使得处理器 101 中的第一数据处理单元 140 继续下一指令的第一数据处理, 如上所述.

由于在处理器 101 的第二指令执行阶段 (EXP2) 中延续了阶段, 处理器 101 中的第二数据处理单元 141 并不执行数据处理 (NOP) .

在协处理器 102 的第三指令执行阶段 (EXE3) 中, 协处理器 102 的第三协处理器处理单元 152 执行数据处理。

从时间间隔 5 切换到时间间隔 6 中, 由于状态管理单元 129 发送一个当前阶段延续信号 (NOP), 处理器 101 中的控制电路 105 并不打开流水线锁存器 143。

接着在协处理器 102 中, 从流水线阶段 (EXE3) 切换到 (EXE4) 中, 控制电路 110 打开第三流水线锁存器 162。

在时间间隔 6 中, 为处理器 101 延续第一指令执行阶段 (NXT - EXP1), 使得处理器 101 中的第一数据处理单元 140 继续下一条指令执行的第一数据处理, 如上所述。

由于在处理器 101 的第二指令执行阶段 (EXP2) 中延续了阶段, 处理器 101 中的第二数据处理单元 141 并不执行数据处理 (NOP)。

通过令第二数据处理单元 141 将 (EXP2) 阶段延续到 3 个阶段 (换言之, 将这一阶段延长两个阶段), 在从 “当前阶段延续状态” 切换到 “下一阶段执行状态” 及结束将当前阶段延续信号传输给处理器 101 的控制电路 105 之前, 状态管理单元 129 令处理器 101 的阶段延续了与延续数等价的一个时间间隔。

在协处理器 102 的第四指令执行阶段 (EXE4) 中, 协处理器 102 的第四协处理器处理单元 153 执行数据处理因此在数据处理单元输出总线 125 上输出结果数据并在标志数据总线 131 上输出标志信息。然后, 协处理器 102 中的控制电路 110 启动协处理器 102 中的驱动器 154, 使得

经第四协处理器处理单元 153 处理过的数据在数据处理单元输出总线 125 上输出。

从时间间隔 6 切换到时间间隔 7 中，处理器 101 中的控制电路 105 在从流水线阶段 (NXT - EXP1) 切换到 (NXT - EXP2) 中打开处理器 101 中的流水线锁存器 143。

处理器 101 中的控制电路 105 从流水线阶段 (EXP2) 切换到 (WB1) 中启动驱动器 135，使得来自标志数据总线 131 的标志信息传输给标志存储寄存器 108。

在时间间隔 7 中，在处理器 101 的第二指令执行阶段 (NXT - EXP2) 中，第二数据处理单元 141 执行下一指令执行的第二数据处理。

在处理器 101 的写阶段 (WB1) 中，通用寄存器 106 接收来自数据处理单元输出总线 125 的数据并存储该数据。类似地，标志存储寄存器 108 接收标志数据总线 131 上协处理器 102 发送给处理器 101 的标志信息，并存储之。

以这一方式，协处理器 102 在计算中能直接使用通用寄存器 106 中的数据，从而过程能加快通过存储器传输数据所占用的时间，因为这一技术不再需要通过存储器传输。这里能够以相同的方式加快处理器 101 的标志存储寄存器 108 中的标志的引用。此外，由于避免了传输中的数据不等性及处理器 101 与协处理器 102 的流水线处理中的混乱，可以减少执行时间。

这里，在不需要协处理器 102 时，用于连接处理器 101 与协处理器 102 的数据处理单元输入总线 124 与数据处理单元输出总线 125、用于将存储器 103 连接在协处理器 102 上的一部分指令供给总线 126、用于连接处理器 101 到协

处理器 102 的标志数据总线 130 及用于连接协处理器 102 到处理器 101 上的标志数据总线 131 全都能够断开。

同时，与相关的技术实例相比时，在协处理器处理数据时不再需要通过存储器将待处理的数据传送给协处理器，这意味着能减少执行时间。

再者，在这一第一实施例中，处理器 101 中的解码器 104 根据协处理器指令的类型确定延续数，但也可能将处理器 101 用一条计算结束信号线连接在协处理器 102 的数据处理单元中任何一个上，通过传输一个计算结束信号便可以由状态管理单元 129 结束状态延续。此外，还可将协处理器指令本身设置成包含表示处理器 101 与协处理器 102 之间的流水线阶段中的差别的信息的，从而使解码器 104 将来自这一信息的状态延续数通知状态管理单元 129。

这里应指出，虽然本系统能构成为使处理器 101 连接在多个不同类型的协处理器 102 上，但上述说明是对处理器 101 只连接在一个协处理器 102 上的情况描述的。对于将处理器 101 连接在用于执行乘法的一个协处理器上，用于执行乘积之和的一个协处理器上及用于执行除法的一个协处理器上的一个实例，则可在处理器 101 与各协处理器之间设置由指令供给总线 126、数据处理单元输入总线 124、数据处理单元输出总线 125、标志数据总线 130 与标志数据总线 131 构成的五条总线。在这一情况中，有必要为协处理器指令的指令码中的位模式分配对各协处理器唯一的模式。此时处理器 101 便能以本实施例相同的方式为任何利用上述总线的协处理器执行供给与存储寄存器数据与标志。这里，这些协处理器是设计成只在解码结果表

示用于该特定协处理器的一条协处理器指令时，才以本实施例中的协处理器 102 的相同方式执行协处理器指令的。

同时，本实施例描述了处理器 101 执行 2 阶段流水线过程及协处理器 102 执行 4 阶段流水线过程的情况，但本结构不限于这些值。当协处理器 102 执行 b 阶段 ($1 < b$) 流水线过程而处理器 101 执行 c 阶段 ($1 < c < b$) 流水线过程时，可在 c 阶段上设置一个状态管理单元，由解码器通知这一状态管理单元延续数 $b-c$ ，并令第 c 阶段延续执行等于延续数的阶段数。当处理器的执行阶段数长于协处理器的数时 (即 $c > b$)，则可将延续数设定为零。在这一情况中，在完成了 c 个执行阶段的点上，处理器从协处理器取计算结果，而协处理器则从完成了 b 个执行阶段的点上开始输出计算结果，直到处理器完成 c 个执行阶段从而接收执行结果为止。

再者，指令供给总线 126 是描述为连接在处理器 101 的指令寄存器 112 的输入侧上及协处理器 102 的指令寄存器 119 的输入侧上的，但也可能设置一条只将处理器 101 的指令寄存器 112 的输入侧连接在存储器 103 上的指令供给总线，而用另一条总线将处理器 101 的指令寄存器 112 的输出侧连接在协处理器 102 的解码器 109 的输入侧上。以这一方式，能省去指令寄存器 119，从而指令供给总线的电负荷得以降低。

当本信息处理装置是由一块 LSI 构成时，则如图 11 中所示，处理器 101 的配置可分成由解码器 104 与控制电路 105 构成的一个控制模块单元；以及由第一数据处理单元 140、第二数据处理单元 141、数据总线 (数据处理单元输入总线 124、数据处理单元输出总线 125、指令供给总

线 126) 等构成的一个数据路径单元。协处理器 102 的配置也可分成由解码器 109 与控制电路 110 构成的一个控制模块单元及由第一至第四协处理器处理单元 150 - 153、数据总线等构成的一个数据路径单元。在这种配置中, 总存在着数据路径单元的配置的一边中的数据总线的位片(位宽度)的相关性, 从而由于在本信息处理装置中, 数据总线是处理器与协处理器共用的, 能够容易地增加用于不同类型计算的新的协处理器, 并不用考虑协处理器 102 是否已经存在。以这一方式, 可以容易地在一个芯片上实现带有各具不同功能的多个协处理器的信息处理装置。

第二实施例

图 4 为展示本发明的第二实施例的信息处理装置的构造的方框图。可以看出, 这一信息处理装置由处理器 201、协处理器 202、存储器 203、保存数据存储缓冲器 232 及驱动器 259 构成。注意与第一实施例中相同的构成元件给予了相同的参照数字并省略了它们的描述, 从而以下的描述着重在与第一实施例的差别上。

图 4 中, 处理器 201 是设置成执行由一个取指令阶段、一个指令解码阶段、一个指令执行阶段及一个写阶段构成的 4 阶段流水线过程的, 为了便于说明, 将执行阶段看成是只由一个阶段构成的。与第一实施例的主要差别在于涉及执行一条用于在处理器 201 与协处理器 202 之间的通用寄存器 106 中传送数据的传送指令, 以及涉及在中断过程开始时产生一个对通知的应答到协处理器之后, 分支到中断过程这两点上。结果, 构造中还包括中断请求信号线 240、中断信号线 241 及完成信号线 242。在本实施例中, 第一实施例中由两条 32 位总线构成的数据处理单元输出

总线 125 可以只由一条 32 位总线构成。

协处理器 202 是设置成执行由一个取指令阶段、一个指令解码阶段、一个指令执行阶段及一个写阶段构成的 4 阶段流水线过程的，为了便于说明将执行阶段看成是只由一个阶段构成的。与第一实施例的主要差别在于涉及协处理器传送指令的执行及涉及从处理器 201 接收中断过程开始的通知、在执行中途停止协处理器指令与保存数据、以及向处理器 201 传输表示就绪状态的应答这几点上。结果，该结构额外包括数据处理单元输入缓冲器 230、数据处理单元输出缓冲器 231、保存数据存储缓冲器 232 以及围绕它们的驱动器 250、251 及 253 - 259，这些是用数据处理单元输入缓冲器保存总线 243 及数据处理单元输出缓冲器返回总线 245 连接的。

本实施例中的协处理器传送指令包括从处理器 201 内部的通用寄存器 106 到协处理器 202 内部的数据处理单元输入缓冲器 230 的数据传送指令以及从数据处理单元输出缓冲器 231 到处理器 201 内部的通用寄存器 106 的数据传送指令。这些是以诸如下面所示的格式表示的，它指定传送指令（“MOVE”指令）的源与目的地，诸如数据处理单元输入缓冲器 230（缩略为 ex_in BUF）与数据处理单元输出缓冲器 231（缩略为 ex_out BUF）。

MOV Dn ex_in buf:从寄存器 Dn 到 ex_in BUF 的数据
传送

MOV ex_ouf BUF, Dn:从 ex_out BUF 到寄存器 Dn 的
数据传送

在本实施例中，协处理器 202 是设定为执行包含 3 个或更多的操作数数据的指定计算的协处理器指令的，诸如三

项计算 $D_n \cdot D_m + D_1$ 。这样做时，事先用协处理器传送指令将三项计算中的一个操作数数据 D_1 传送给协处理器 202。以相同的方式，当三项计算的计算结果增加到多个字长时，用传送指令将一部分计算结果传送给处理器 201。

处理器 201 中的中断请求信号线 240 将中断请求从处理器 201 外部传送到处理器 201 的控制电路 205。

中断请求信号线 241 将中断请求从处理器 201 的控制电路 205 传送到协处理器 202 的控制电路 210。

完成信号线 242 将中断请求期间的处理及完成中断处理的完成信号从协处理器 202 的控制电路 210 传送到处理器 201 的控制电路 205。

解码器 204 具备第一实施例中解码器 104 的功能，但差别在于具有解码协处理器传送指令的功能。更具体地，当作为解码结果，指定了通用寄存器 106 作为传送指令 ($MOV D_n, ex_in\ BUF$) 的源，且指定数据处理单元输入缓冲器 230 作为目的地时，解码器 204 发布下列微指令：

(p1.1) 从寄存器 D_n 读数据。

(p1.2) 启动驱动器 122，用协处理器数据锁存器 118 锁存读自源寄存器的数据。

当解码结果为 ($MOV ex_out\ BUF, D_n$) 时，解码器 204 发布下列微指令：

(p2.1) 用寄存器写缓冲器 116 锁存来自数据处理单元输出总线 125 的数据。

(p2.2) 将寄存器写缓冲器 116 中的数据写入寄存器 D_n 中。

当解码结果为从中断过程返回 (缩略成 (rti 指令))

时，解码器 204 发布下述微指令：

(p3.1) 返回过程 (返回到一个返回地址) 。

控制电路 205 具备第一实施例中的控制电路 105 的功能，但差别在于具有下述功能，当通过中断请求信号线 240 从处理器 201 外部接收到一个中断请求时，通过中断信号线 241 通知协处理器 202 的控制电路 210 该中断，并且在通过完成信号线 242 从协处理器 202 接收到表示已完成准备工作的应答时，控制分支到中断过程。它还输出与上面给出的各微指令 (p1.1)、(p1.2)、(p2.1)、(p2.2) 与 (p3.1) 对应的控制信号。

数据处理单元 207 在处理器 201 的执行阶段中执行处理。

协处理器 202 中的解码器 209 具备第一实施例中的解码器 109 的功能，但差别在于具有解码协处理器传送指令及中断返回程序的功能。更具体地，当解码结果为 (MOV Dn, ex_in BUF) 时，解码器发布下述微指令：

(E1.1) 启动驱动器 251，将来自数据处理单元输入总线 124 的数据输出到数据处理单元输入缓冲器 230。

(E1.2) 将通过驱动器 251 从数据处理单元输入总线 124 传输的数据锁存在数据处理单元输入缓冲器 230 中。

当解码结果为 (MOV ex_out BUF, Dn) 时，解码器发布下述微指令：

(E2.1) 启动驱动器 257，将数据处理单元输出缓冲器 231 中的数据输出到数据处理单元输出总线 125 上。

(E2.2) 将来自寄存器写缓冲器 116 的数据写入寄存器 Dn 中。

当解码结果为从中断过程的一条返回指令 (rti 指

令)时, 解码器发布下述微指令:

(E4.1) 将保存数据写入保存数据存储缓冲器 232 中并启动驱动器 259 .

(E4.2) 启动驱动器 251 , 写入数据处理单元输入缓冲器 230 .

(E4.3) 将其它存储数据写入保存数据存储缓冲器 232 , 并启动驱动器 259 .

(E4.4) 启动驱动器 256 , 写入数据处理单元输出缓冲器 231 .

(E4.5) 指示在完成信号线 242 上输出 .

协处理器 202 中的控制电路 210 具备第一实施例中的控制电路 110 的功能, 但差别在于还具有在中断过程开始时执行进入保存数据存储缓冲器 232 的保存过程的功能. 更具体地, 当通过中断信号线 241 接收到一个中断信号时, 控制电路 210 执行下述控制过程:

(E3.1) 启动驱动器 258 , 令数据从数据处理单元输入缓冲器 230 输出到数据处理单元输出总线 125 上 .

(E3.2) 将数据处理单元输出总线 125 上的数据写入保存数据存储缓冲器 232 中 .

(E3.3) 启动驱动器 257 , 将数据处理单元输出缓冲器 231 中的数据输出到数据处理单元输出总线 125 上 .

(E3.4) 将数据处理单元输出总线 125 上的数据写入保存数据存储缓冲器 232 中 .

(E3.5) 在完成信号线 242 上输出信号 .

控制电路 210 还输出与上述 (E1.1)、(E1.2)、(E2.1)、(E2.2)、(E1.1)、(E4.1)、(E4.1)、(E4.3) 及 (E4.5) 对应的各种控制信号 .

协处理器处理单元 211 在协处理器 202 的执行阶段中执行协处理器指令。

数据处理单元输入缓冲器 230 是为了协处理器处理单元 211 用多个操作数进行处理而存储要通过数据处理单元输入总线 124 从处理器 201 或协处理器 202 传送的操作数的缓冲器。

数据处理单元输出缓冲器 231 是为了多个数据要由协处理器处理单元 211 处理与输出而存储要通过数据处理单元输出总线 125 传送给处理器 201 或协处理器 202 的操作数的缓冲器。

保存数据存储缓冲器 232 是用于保存协处理器 202 的数据处理单元输入缓冲器 230 与数据处理单元输出缓冲器 231 中的数据的缓冲器，在本实施例中它由具有两个 32 位数据容量的一个 FIFO（先进先出）存储器构成。保存数据存储缓冲器 232 是示出为在处理器 201 与协处理器 202 外部的，但是它也可设置在处理器 201 或协处理器 202 内部。同样，保存数据存储缓冲器 232 是示出为通过数据处理单元输出总线 125 连接在处理器 201 与协处理器 202 上的，但它也可连接在数据处理单元输入总线 124、从存储器 103 到处理器 201 的数据总线 127 或者从处理器 201 到存储器 103 的数据总线 128 中任何一条上。

数据处理单元输入缓冲器保存总线 243 是用于在出现中断请求时，将来自协处理器 202 的数据处理单元输入缓冲器 230 的数据保存在保存数据存储缓冲器 232 中的总线。

数据处理单元输入缓冲器返回总线 244 是用于在完成中断过程时，将存储在保存数据存储缓冲器 232 中的数据

返回给协处理器 202 的数据处理单元输入缓冲器 230 的总线。

数据处理单元输出缓冲器返回总线 245 是在完成中断过程时，将存储在保存数据存储缓冲器 232 中的数据返回给协处理器 202 的数据处理单元输出缓冲器 231 的总线。

驱动器 250 是按照来自控制电路 205 的控制信号启动的，并将数据从数据处理单元输入总线 124 传输到协处理器处理单元 211。

驱动器 251 是按照来自控制电路 205 的控制信号启动的，并将数据从数据处理单元输入总线 124 传输到数据处理单元输入缓冲器 230。

驱动器 253 是按照来自控制电路 205 的控制信号启动的，并将数据从数据处理单元输入缓冲器 230 传输到协处理器处理单元 211。

驱动器 254 是按照来自控制电路 205 的控制信号启动的，并将数据从协处理器处理单元 211 传输到数据处理单元输出总线 125。

驱动器 255 是按照来自控制电路 205 的控制信号启动的，并将数据从协处理器处理单元 211 传输到数据处理单元输出缓冲器 231。

驱动器 256 是按照来自控制电路 205 的控制信号启动的，并将数据从数据处理单元输入总线 124 传输到数据处理单元输出缓冲器 231。

驱动器 257 是按照来自控制电路 205 的控制信号启动的，并将数据从数据处理单元输出缓冲器 231 传输到数据处理单元输出总线 125。

驱动器 258 是按照来自控制电路 205 的控制信号启动

的，并将数据从数据处理单元输入缓冲器 230 传输到数据处理单元输出总线 125 上。

驱动器 259 是在返回存储在保存数据存储缓冲器 232 中的保存数据时启动的，并将数据从保存数据存储缓冲器 232 传输到数据处理单元输入总线 124 上。

以下是如上述构成的本发明的第二实施例的信息处理装置的操作的说明，说明分成（2.1）从通用寄存器 106 到数据处理单元输入缓冲器 230 的一条协处理器传送指令的处理，（2.2）利用数据处理单元输入缓冲器 230 中的操作数数据的协处理指令的处理，（2.3）从数据处理单元输出缓冲器 231 到通用寄存器 106 的协处理器传送指令的处理，（2.4）在中断过程开始时，数据处理单元输入缓冲器 230 与数据处理单元输出缓冲器 231 中的数据的数据保存，以及（2.5）在完成中断过程时，将数据返回到数据处理单元输入缓冲器 230 与数据处理单元输出缓冲器 231。

（2.1）写入数据到数据处理单元输入缓冲器 230 中

图 5 为展示在执行将存储在处理器 201 中的通用寄存器 106 中的数据写入数据处理单元输入缓冲器 230 的一条协处理器传送指令时的流水线处理的状态的时间图。

在该图中，“IF1”为处理器 201 检索一条协处理器传送指令的阶段，“DP1”为处理器 201 解码一条协处理器传送指令的阶段，“EXP1”为处理器 201 中协处理器传送指令的第一指令执行阶段，处理器 201 中并不实际执行数据处理（NOP），“WB1”为处理器 201 写协处理器传送指令的计算结果的阶段，“DE1”为协处理器 202 解码一条协处理器传送指令的阶段，“EXE1”为在协处

理器 202 中的一条执行指令的执行阶段。这里，“输入存储操作数”表示用于存储写入数据处理单元输入缓冲器 230 的数据的理论定时。

在时间间隔 1 中，存储器 103 在处理器 201 的取指令阶段中将指令输出到导向处理器 201 与协处理器 202 的指令供给总线 126 上。

从时间间隔 1 切换到时间间隔 2 中，处理器 201 中的控制电路 205 从 (IF1) 前进到 (DP1) 并相应地打开处理器 201 的指令寄存器 112。以相同的方式，协处理器 202 中的控制电路 210 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 202 的指令寄存器 119。

在时间间隔 2 中，在处理器 201 的指令解码阶段 (DP1) 中，处理器 201 中的解码器 204 在指令寄存器 112 打开之后接收到所取出的指令，然后解码该指令。然后，控制电路 205 利用解码器 204 输出的微指令，并通过启动驱动器 122，将通用寄存器 106 中的操作数放置在数据处理单元输入总线 124 上。

在协处理器 202 的指令解码阶段 (DE1) 中，协处理器 202 中的解码器 209 在指令寄存器 119 打开之后接收指令，然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中，处理器 201 中的控制电路 205 在从流水线阶段 (DP1) 前进到 (EXP1) 中，打开微指令寄存器 113 与数据锁存器 114。结果，将微指令从解码器 204 传输到控制电路 205，从而用处理器 201 中的控制电路 205 执行流水线处理，将操作数从数据处理单元输入总线 124 传输到处理器 201 中的数据处理单元 207。这时，协处理器数据锁存器 118 是打开的，因此

将操作数从数据处理单元输入总线 124 传输到协处理器 202。类似地，协处理器 202 中的控制电路 210 在从流水线阶段 (DE1) 前进到 (EXE1) 中打开微指令寄存器 120。一旦打开了处理器 201 中的微指令寄存器 113，处理器 201 中的控制电路 205 便接收到解码器 204 在指令解码阶段 (DP1) 中解码的微指令。类似地，一旦打开了微指令寄存器 120，协处理器 202 中的控制电路 210 便接收到解码器 209 在指令解码阶段 (DE1) 中解码的微指令。

在时间间隔 3 中，在处理器 201 的执行阶段 (EXP1) 中，处理器 201 的数据处理单元 207 并不执行数据处理 (NOP)。

在协处理器 202 的指令执行阶段 (EXE1) 中，协处理器 202 中的控制电路 210 启动驱动器 251，后者在数据处理单元输入总线 124 上传输一个 32 位操作数给协处理器 202 中的数据处理单元输入缓冲器 230。然后将这一操作数存储在数据处理单元输入缓冲器 230 中。

(2.2) 利用数据处理单元输入缓冲器 230 中的操作数数据执行协处理器指令

图 6 为展示在利用存储在主处理器中的通用寄存器 106 中的数据及存储在数据处理单元输入缓冲器 230 中的数据执行一条 3 项计算指令时，协处理器 202 的流水线处理的状态的时间图，它将一部分计算结果传送给处理器 201 的通用寄存器 106 而将另一部分计算结果写入数据处理单元输出缓冲器 231 中。在本图中，图注“输出存储器数据”表示在数据处理单元输出缓冲器 231 中存储数据的定时。

在时间间隔 1 中，存储器 103 在处理器 201 的取指令

阶段 (IF1) 中将指令输出到导向处理器 201 与协处理器 202 的指令供给总线 126 上。

从时间间隔 1 切换到时间间隔 2 中, 处理器 201 中的控制电路 205 从 (IF1) 前进到 (DP1) 并相应地打开处理器 201 的指令寄存器 112。以相同的方式, 协处理器 202 中的控制电路 210 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 202 的指令寄存器 119。

在时间间隔 2 中, 在指令解码阶段 (DP1) 中, 处理器 201 中的解码器 204 在指令寄存器 112 打开以后接收指令, 然后解码该指令。然后, 控制电路 205 利用解码器 204 输出的微指令, 并通过启动驱动器 122, 将通用寄存器 106 中的数据放置在数据处理单元输入总线 124 上。

在协处理器 202 的指令解码阶段 (DE1) 中, 协处理器 202 中的解码器 209 在指令寄存器 119 打开之后接收指令, 然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中, 处理器 201 中的控制电路 205 在从流水线阶段 (DP1) 前进到 (EXP1) 中打开微指令寄存器 113 与数据锁存器 114。结果, 将微指令通过打开的微指令寄存器 113 从解码器 204 传输到控制电路 205, 从而用处理器 201 中的控制电路 205 执行流水线处理, 操作数是从数据处理单元输入总线 124 传输给处理器 201 中的数据处理单元 207 的。这时, 协处理器数据锁存器 118 是打开的, 因此将操作数从数据处理单元输入总线 124 传输给协处理器 202。类似地, 协处理器 202 中的控制电路 210 在从流水线阶段 (DE1) 前进到 (EXE1) 中打开微指令寄存器 120。一旦打开了处理器 201 中的微指令寄存器 113, 处理器 201 中的控制电路 205

便接收到解码器 204 在指令解码阶段 (DP1) 中所解码的微指令。类似地, 一旦打开了微指令寄存器 120, 协处理器 202 中的控制电路 210 便接收到解码器 209 在指令解码阶段 (DE1) 中所解码的微指令。

在时间间隔 3 中, 在处理器 201 的执行阶段 (EXP1) 中, 处理器 201 中的数据单元 207 并不执行数据处理 (NOP)。

在协处理器 202 的指令执行阶段 (EXE1) 中, 协处理器 202 中的控制电路 210 启动驱动器 250, 从而在数据处理单元输入总线 124 上将一个 32 位操作数传输给协处理器 202 中的协处理器处理单元 211, 并启动驱动器 253 而将数据处理单元输入缓冲器 230 中的一个操作数传输给协处理器处理单元 211。然后, 协处理器处理单元 211 利用来自数据处理单元输入总线 124 的操作数及来自数据处理单元输入缓冲器 230 的操作数进行处理, 并输出两个数据作为其处理结果。然后, 协处理器 202 中的控制电路 210 启动协处理器 202 中的驱动器 254, 并在数据处理单元输出总线 125 上传输协处理器处理单元 211 的处理结果之一, 将该处理结果传输给处理器 201。协处理器 202 中的控制电路 210 还启动协处理器 202 中的驱动器 255, 并将协处理器处理单元 211 的另一个处理结果传输给数据处理单元输出缓冲器 231, 将处理结果存储在那里。

从时间间隔 3 切换到时间间隔 4 中, 处理器 201 中的控制电路 205 在从流水线过程 (EXP1) 切换到 (WB1) 中打开寄存器写缓冲器 116, 结果, 处理结果之一便通过打开的寄存器写缓冲器 116 从数据处理单元输出总线 125 传输到处理器 201 中的通用寄存器 106 中。

在时间间隔 4 中，处理器 201 中的通用寄存器 106 存储一部分处理结果数据。

(2.3) 从数据处理单元输出缓冲器 231 中读数据及存储在通用寄存器 106 中的操作

图 7 为展示将存储在数据处理单元输出缓冲器 231 中的数据写入处理器 201 的通用寄存器 106 的操作的流水线处理的状态的时间图。

在时间间隔 1 中，存储器 103 将指令输出到指令供给总线 126 上，后者在处理器 201 的取指令阶段 (IF1) 中导向处理器 201 及协处理器 202 。

从时间间隔 1 切换到时间间隔 2 中，处理器 201 中的控制电路 205 从 (IF1) 切换到 (DP1) 并相应地打开处理器 201 的指令寄存器 112。以相同的方式，协处理器 202 中的控制电路 210 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 202 的指令寄存器 119。

在时间间隔 2 中，在指令解码阶段 (DP1) 中，处理器 201 中的解码器 204 在指令寄存器 112 打开之后接收指令，然后解码该指令。

在协处理器 202 的指令解码阶段 (DE1) 中，协处理器 202 中的解码器 209 在指令寄存器 119 打开之后接收指令，然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中，处理器 201 中的控制电路 205 在从流水线阶段 (DP1) 切换到 (EXP1) 中打开微指令寄存器 113 及数据锁存器 114。结果，微指令通过打开的微指令寄存器 113 从解码器 204 传输到控制电路 205，从而用处理器 201 中的控制电路 205 执行流水线处理，数据锁存器 114 是打开的，因此操作数从数据处

理单元输入总线 124 传输到处理器 201 中的数据处理单元 207 中。类似地，协处理器 202 中的控制电路 210 在从流水线阶段 (DE1) 前进到 (EXE1) 中打开微指令寄存器 120。一旦打开了处理器 201 中的微指令寄存器 113，处理器 201 中的控制电路 205 便接收到解码器 204 在指令解码阶段 (DP1) 中所解码的微指令。类似地，一旦打开了微指令寄存器 120，协处理器 202 中的控制电路 210 便接收到解码器 209 在指令解码阶段 (DE1) 中所解码的微指令。

在时间间隔 3 中，在处理器 201 的执行阶段 (EXP1) 中，处理器 201 中的数据处理单元 207 并不执行数据处理 (NOP)。

在协处理器 202 的指令执行阶段 (EXE1) 中，协处理器 202 中的控制电路 210 启动驱动器 257，从而在数据处理单元输出总线 125 上传输存储在数据处理单元输出缓冲器 231 中的处理结果数据。

从时间间隔 3 切换到时间间隔 4 中，处理器 201 中的控制电路 205 在从流水线过程 (EXP1) 切换到 (WB1) 中打开寄存器写缓冲器 116，结果，便通过打开的寄存器写缓冲器 116 将处理结果从数据处理单元输出总线 125 传输到处理器 201 中的通用寄存器 106。

在时间间隔 4 中，处理器 201 中的通用寄存器 106 存储一部分处理结果数据。

通过在协处理器 202 上装备这些种类的缓冲器，能使协处理器处理多个操作数而不增加由扩大芯片尺寸引起的功耗或成本。

(2.4) 在中断过程开始时保存数据处理单元输入缓冲器 230 与数据处理单元输出缓冲器 231 中的数据的操作

图 8 为展示出现对处理器 201 的一个中断及将协处理器 202 的数据处理单元输入缓冲器 230 中的数据与协处理器 202 的数据处理单元输出缓冲器 231 中的数据保存在保存数据存储缓冲器 232 中的操作的时间图。在这一时间图中，图注“ESC1”表示执行保存协处理器 202 的数据处理单元输入缓冲器 230 中的数据的指令，而图注“ESC2”则表示执行保存协处理器 202 的数据处理单元输出缓冲器 231 中的数据的指令。

在时间间隔 1 中，处理器 201 在中断请求信号线 240 上接收来自装置外部的一个中断请求信号。

在时间间隔 2 中，处理器 201 中的控制电路 205 在中断信号线 241 上传输一个中断信号给协处理器 202。然后，协处理器 202 中的控制电路 210 接收在中断信号线 241 上发送的这一信号。

在时间间隔 3 中，协处理器 202 中的控制电路 210 启动驱动器 258，因而在数据处理单元输出总线 125 上传输数据处理单元输入缓冲器 230 中的处理结果。然后，保存数据存储缓冲器 232 接收数据处理单元输出总线 125 上的这一数据，并更新其存储内容。

在时间间隔 4 中，协处理器 202 中的控制电路 210 启动驱动器 257，因而在数据处理单元输出总线 125 上传输数据处理单元输出缓冲器 231 中的处理结果。然后，保存数据存储缓冲器 232 接收数据处理单元输出总线 125 上的这一数据并存储之。

在时间间隔 5 中，协处理器 202 中的控制电路 210 通过完成信号线 242 传输一个中断准备完成信号给处理器 201，借此允许处理器 201 起中断过程。

(2.5) 完成中断过程后用于数据处理单元输入缓冲器 230 及数据处理单元输出缓冲器 231 的数据返回过程

图 9 为展示中断返回指令的流水线处理的状态的时间图。在这一时间图中，图注“RTI1”表示处理器 201 与协处理器 202 用于数据处理单元输入缓冲器 230 的数据返回过程，而图注“RTI2”则表示处理器 201 与协处理器 202 用于数据处理单元输出缓冲器 231 的数据返回过程。

下面是参照表示中断返回指令的流水线处理的状态的图 9 中的时间图，对一条中断返回指令的协处理器 202 的操作的说明。

在时间间隔 1 中，存储器 103 在处理器 201 的取指令阶段中将中断返回指令输出到导向处理器 201 与协处理器 202 的指令供给总线 126 上。

从时间间隔 1 切换到时间间隔 2 中，处理器 201 中的控制电路 205 从 (IF1) 前进到 (DP1) 并相应地打开处理器 201 的指令寄存器 112。以相同的方式，协处理器 202 中的控制电路 210 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 202 的指令寄存器 119。

在时间间隔 2 中，在处理器 201 的指令解码阶段 (DP1) 中，处理器 201 中的解码器 204 在指令寄存器 112 打开之后接收到所取出的中断返回指令，然后解码该指令。类似地，在协处理器 202 的指令解码阶段 (DE1) 中，协处理器 202 中的解码器 209 在指令寄存器 119 打开之后

接收到中断返回指令，然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中，处理器 201 中的控制电路 205 在 (RTI1) 过程中打开微指令寄存器 113 与数据锁存器 114。结果，中断返回微指令通过打开的微指令寄存器 113 从解码器 204 传输到控制电路 205。类似地，协处理器 202 中的控制电路 210 在前进到流水线阶段 (RTI1) 中打开微指令寄存器 120。一旦打开了处理器 201 中的微指令寄存器 113，处理器 201 中的控制电路 205 便接收到解码器 204 在指令解码阶段 (DP1) 中从 (RTI1) 指令解码出的微指令。类似地，一旦打开了微指令寄存器 120，协处理器 202 中的控制电路 210 便接收到解码器 209 在指令解码阶段 (DE1) 中从 (RTI1) 解码出的微指令。

在时间间隔 3 中，在处理器 210 的指令解码阶段 (RTI2) 中，重复地输出中断返回微指令。

在处理器 201 的指令执行阶段 (RTI1) 中，处理器 201 执行返回过程。

在协处理器 202 的指令解码阶段 (RTI2) 中，重复地输出中断返回微指令。

在协处理器 202 的指令执行阶段 (RTI2) 中，驱动器 259 是启动的，而保存数据存储缓冲器 232 将其数据放置在导向处理器 201 与协处理器 202 的数据处理单元输入总线 124 上。协处理器 202 中的控制电路 210 启动协处理器 202 中的驱动器 256，后者将数据处理单元输入总线 124 上的数据发送到协处理器 202 中的数据处理单元输入缓冲器 230 中。然后，数据处理单元输入缓冲器 230 存储从数据处理单元输入总线 124 接收的数据。

在时间间隔 4 中，随着处理器 201 的指令解码阶段 (RTI3)，重复地输出中断返回微指令。

在处理器 201 的指令执行阶段 (RTI2) 中，处理器 201 执行一个返回过程。

在协处理器 202 的指令执行阶段 (RTI2) 中，驱动器 259 是启动的，而保存数据存储缓冲器 232 将其数据放置在导向处理器 201 与协处理器 202 的数据处理单元输入总线 124 上。然后，协处理器 202 中的控制电路 210 启动协处理器 202 中的驱动器 256，后者将数据处理单元输入总线 124 上的数据发送到协处理器 202 中的数据处理单元输出缓冲器 231 中。然后，数据处理单元输出缓冲器 231 存储从数据处理单元输入总线 124 接收的数据。

第三实施例

图 10 为展示本发明的第三实施例的信息处理装置的结构方框图。可以看出这一信息处理装置由处理器 301、协处理器 302 及存储器 303 构成。注意，与图 4 中所示的第二实施例中相同的构造元件给予相同的参照数字并省略它们的说明，从而以下的说明集中在与第二实施例的差别上。

在本实施例中，处理器 301 具备第二实施例中的处理器 201 的功能，但差别在于具有切换任务的功能。这里，处理器 301 的任务切换是指通过执行一条任务切换指令而达到执行一个任务切换过程。处理器 301 的这一任务切换可根据普通技术做到。同时，在本实施例中，数据处理单元输出总线 125 可以和第二实施例一样用一条 32 位总线构成。

协处理器 302 包括数据处理单元输入缓冲器 330 与

331，数据处理单元输出缓冲器 332 与 333 以及周围的驱动器来取代第二实施例的数据处理单元输入缓冲器 230、数据处理单元输出缓冲器 231 及周围的驱动器，以及额外包括任务管理单元 361。

处理器 301 中的解码器 304 具备第二实施例中的解码器 204 的功能，但差别在于具有解码任务切换指令及发布考虑中的指令所指定的切换任务的微指令的功能。

处理器 301 中的控制电路 305 具备第二实施例中的控制电路 205 的功能，但差别在于具有输出与所接收的实现任务切换的微指令相对应的多种控制信号的功能。

协处理器 302 中的解码器 309 具备第二实施例中的解码器 209 的功能，但差别在于具有解码任务切换指令的功能。更具体地，当解码结果显示指令为一条任务切换指令时，解码器 309 便发布一条将所解码的指令所指定的任务号存储在任务管理单元 361 中的微指令。

处理器 301 的数据处理单元 207 执行处理器 301 的主要数据处理。

协处理器 302 中的控制电路 310 具备第二实施例中的控制电路 210 的功能，但差别在于具有在接收到作为一条任务切换指令的解码结果的微指令时输出控制信号、将该指令指定的任务号存储在任务管理单元 361 中的功能。同时，在接收到如第二实施例中所说明的作为一条协处理器传送指令的解码结果的微指令时，控制电路 310 按照存储在任务管理单元 361 中的任务号选用数据处理单元输入缓冲器 330 与数据处理单元输入缓冲器 331 中之一。例如，当存储在任务管理单元 361 中的任务号为 1 时，控制电路 310 执行控制来使用数据处理单元输入缓冲器 330 与数据

处理单元输出缓冲器 332，而当任务号为 2 时，控制电路 310 执行控制来使用数据处理单元输入缓冲器 331 与数据处理单元输出缓冲器 333。作为另一例子，当一条协处理器指令为用于执行一个三项计算的时，便选用数据处理单元输入缓冲器 330 与数据处理单元输入缓冲器 331，当协处理器指令的执行生成多个数据作为计算结果时，便按照任务号选用数据处理单元输出缓冲器 332 与数据处理单元输出缓冲器 333。

数据处理单元输入缓冲器 330 与 331 为在协处理器处理单元 211 使用多个操作数执行处理时用来存储数据处理单元输入总线 124 上发送的数据的缓冲器。这时，按照任务管理单元 361 中的任务号选择数据处理单元输入缓冲器 330 与 331。

数据处理单元输出缓冲器 332 与 333 是在作为协处理器处理单元 211 的计算结果生成若干数据时用来存储数据处理单元输出总线 125 上待发送的数据的缓冲器。这时，按照任务管理单元 361 中的任务号选择数据处理单元输出缓冲器 332 与 333。

驱动器 350 为按照控制电路 305 的控制启动的驱动器，并用于将数据从数据处理单元输入总线 124 传输到协处理器处理单元 211。

驱动器 351 为按照控制电路 305 的控制启动的驱动器，并用于将数据从数据处理单元输入总线 124 传输到数据处理单元输入缓冲器 330。

驱动器 352 为按照控制电路 305 的控制启动的驱动器，并用于将数据从数据处理单元输入总线 124 传输到数据处理单元输入缓冲器 331。

驱动器 354 为按照控制电路 305 的控制启动的驱动器，并用于将数据从协处理器处理单元 211 传输到数据处理单元输出总线 125 上。

驱动器 355 为按照控制电路 305 的控制启动的驱动器，并用于将数据从协处理器处理单元 211 传输到数据处理单元输出缓冲器 332。

驱动器 356 为按照控制电路 305 的控制启动的驱动器，并用于将数据从协处理器处理单元 211 传输到数据处理单元输出缓冲器 333。

驱动器 357 为按照控制电路 305 的控制启动的驱动器，并用于将数据从数据处理单元输出缓冲器 332 传输到数据处理单元输出总线 125 上。

驱动器 358 为按照控制电路 305 的控制启动的驱动器，并将数据从数据处理单元输出缓冲器 333 传输到数据处理单元输出总线 125 上。

选择器 360 受控制电路 305 控制以便选择数据处理单元输入缓冲器 330 或 331。

任务管理单元 361 受控制电路 305 控制以便存储一个任务号及将该任务号传输给控制电路 305。任务管理单元 361 所存储的任务号设定在任务 1 或任务 2 上，当任务号为任务 1 时，便选择数据处理单元输入缓冲器 330 与数据处理单元输出缓冲器 332，当任务号为任务 2 时，便选择数据处理单元输入缓冲器 331 与数据处理单元输出缓冲器 333。

下面是如上构成的本发明的第三实施例的信息处理装置的操作的说明，说明分成（3.1）任务切换操作的执行操作，（3.2）一条协处理器传送指令的操作，其中将

存储在通用寄存器 106 中的数据写入协处理器 302 的一个数据处理单元输入缓冲器中，(3.3) 一条协处理器指令的操作，它在将输出结果写入通用寄存器 106 与一个数据处理单元输出缓冲器之前，使用处理器 301 的通用寄存器 106 中的数据及一个数据处理单元输入缓冲器中的数据进行计算，以及 (3.4) 一条协处理器传送指令的操作，它将协处理器 302 的一个数据处理单元输出缓冲器中的数据写入处理器 301 的通用寄存器 106 中。

(3.1) 在协处理器 302 中执行任务切换的操作

在取指令阶段中，存储器 103 将一条任务切换指令放置在指令供给总线 126 上。在指令解码阶段中，控制电路 310 打开指令寄存器 119。一旦打开了指令寄存器 119，解码器 309 便解码该任务切换指令。在流水线阶段之间切换中，协处理器 302 中的控制电路 310 打开微指令寄存器 120。一旦打开了微指令寄存器 120，控制电路 310 便接收到解码器 309 解码的微指令，并且连同执行任务切换的控制，将原指令所指定的任务号存储在任务管理单元 361 中。在执行三个或更多项的计算的协处理器指令与协处理器指令时，当任务号为 1 时选择数据处理单元输入缓冲器 330 与数据处理单元输出缓冲器 332，而当任务号为任务 2 时，则选择数据处理单元输入缓冲器 331 与数据处理单元输出缓冲器 333。

(3.2) 一条协处理器指令的操作，其中将存储在通用寄存器 106 中的数据写入协处理器 302 的一个数据处理单元输入缓冲器中

下面是将任务号任务 1 存储在任务管理单元 361 中并将存储在通用寄存器 106 中的数据写入数据处理单元输入

缓冲器 330 时，协处理器 302 内部的操作的说明。本操作采用与图 5 中所示的流水线处理的时间图中的状态相同的时序，因此以下的说明也参照图 5。

在时间间隔 1 中，存储器 103 在处理器 301 的取指令阶段中将指令输出到指令供给总线 126 上，该总线导向处理 301 与协处理器 302。

从时间间隔 1 切换到时间间隔 2 中，处理器 301 中的控制电路 305 从 (IF1) 前进到 (DP1) 并相应地打开处理器 301 的指令寄存器 112。以相同的方式，协处理器 302 中的控制电路 310 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 202 的指令寄存器 119。

在时间间隔 2 中，在处理器 301 的指令解码阶段 (DP1) 中，处理器 301 中的解码器 304 在指令寄存器 112 打开之后便接收到所取出的指令，然后解码该指令。然后，控制电路 305 利用解码器 304 输出的微指令并通过启动驱动器 122，将通用寄存器 106 中的操作数放置在数据处理单元输入总线 124 上。

在协处理器 302 的指令解码阶段 (DE1) 中，协处理器 302 中的解码器 309 在指令寄存器 119 打开之后接收到指令，然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中，处理器 301 中的控制电路 305 在从流水线阶段 (DP1) 前进到 (EXP1) 中打开微指令寄存器 113、数据锁存器 114 与协处理器数据锁存器 118。结果，将微指令从解码器 304 传输到控制电路 305，从而用处理器 301 中的控制电路 305 执行流水线处理，将操作数从数据处理单元输入总线 124 传输到处理器 301 中的数据处理单元 207，然后将操作数从数据处

理单元输入总线 124 传输到协处理器 302。类似地，协处理器 302 中的控制电路 310 在从流水线阶段 (DE1) 切换到 (EXE1) 中打开微指令寄存器 120。一旦打开了处理器 301 中的微指令寄存器 113，处理器 301 中的控制电路 305 便接收到解码器 304 在指令解码阶段 (DP1) 中所解码的微指令。类似地，一旦打开了微指令寄存器 120，协处理器 302 中的控制电路 310 便接收到解码器 309 在指令解码阶段 (DE1) 中所解码的微指令。

在时间间隔 3 中，在处理器 301 的执行阶段 (EXP1) 中，处理器 301 的数据处理单元 207 并不执行数据处理 (NOP)。

在协处理器 302 的指令执行阶段 (EXE1) 中，协处理器 302 中的控制电路 310 启动驱动器 361，后者将数据处理单元输入总线 124 上的一个 32 位操作数传输给协处理器 302 中的数据处理单元输入缓冲器 330。然后将这一操作数存储在数据处理单元输入缓冲器 330 中。

(3.3) 在将结果写入通用寄存器 106 及一个数据处理单元输出缓冲器之前，利用通用寄存器 106 中的数据及一个数据处理单元输入缓冲器中的数据进行计算的协处理器指令的操作

下面是当任务管理单元 361 将任务 1 存储在协处理器 302 中时的说明，从而将存储在处理器 301 的通用寄存器 106 中的数据及存储在数据处理单元输入缓冲器 330 中的数据用在计算中，并将计算结果写入通用寄存器 106 与数据处理单元输出缓冲器 332 中。本操作使用与图 6 中所示的流水线处理的时间图中的状态相同的定时，从而下面的说明也参照图 6。

在时间间隔 1 中，存储器 103 在处理器 301 的取指令阶段（IF1）中将指令输出到指令供给总线 126 上，后者导向处理器 301 与协处理器 302。

从时间间隔 1 切换到时间间隔 2 中，处理器 301 中的控制电路 305 从（IF1）前进到（DP1）并相应地打开处理器 301 的指令寄存器 112。以相同的方式，协处理器 302 中的控制电路 310 从（IF1）前进到（DE1）并相应地打开协处理器 302 的指令寄存器 119。

在时间间隔 2 中，在指令解码阶段（DP1）中，处理器 301 中的解码器 304 在指令寄存器 112 打开后便接收到指令，然后解码该指令。然后，控制电路 305 利用解码器 304 输出的微指令，并通过启动驱动器 122，将通用寄存器 106 中的数据放置在数据处理单元输入总线 124 上。

在协处理器 302 的指令解码阶段（DE1）中，协处理器 302 中的解码器 309 在指令寄存器 119 打开之后便接收到指令，然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中，处理器 301 中的控制电路 305 在从流水线阶段（DP1）前进到（EXP1）中打开微指令寄存器 113、数据锁存器 114 与协处理器数据锁存器 118。结果，通过微指令寄存器 113 将微指令从解码器 304 传输到控制电路 305，从而用处理器 301 中的控制电路 305 执行流水线处理，将操作数从数据处理单元输入总线 124 传输到处理器 301 中的数据处理单元 207，并将操作数从数据处理单元输入总线 124 传输给协处理器 302。类似地，协处理器 302 中的控制电路 310 在从流水线阶段（DE1）前进到（EXE1）中打开微指令寄存器 120。一旦打开了处理器 301 中的微指令寄存器 113，处

理器 301 中的控制电路 305 便接收到解码器 304 在指令解码阶段 (DP1) 中所解码的微指令。类似地, 一旦打开了微指令寄存器 120, 协处理器 302 中的控制电路 310 便接收到解码器 309 在指令解码阶段 (DE1) 中所解码的微指令。

在时间间隔 3 中, 在处理器 301 的执行阶段 (EXP1) 中, 处理器 301 中的数据处理单元 207 并不执行数据处理 (NOP) 。

在协处理器 302 的指令执行阶段 (EXE1) 中, 将作为任务信息存储在任务管理单元 361 中的“任务 1”输出到协处理器 302 的控制电路 310 。

协处理器 302 中的控制电路 310 启动驱动器 350, 从而在数据处理单元输入总线 124 上将一个操作数传输给协处理器 302 中的协处理器处理单元 211, 而协处理器 302 中的选择器 360 则按照任务管理单元 361 输出的任务信息“任务 1”选择数据处理单元输入缓冲器 330, 从而将一个操作数从数据处理单元输入缓冲器 330 传输给协处理器处理单元 211。然后协处理器处理单元 211 利用来自数据处理单元输入总线 124 的操作数及来自数据处理单元输入缓冲器 330 的操作数进行处理, 并输出两个数据作为其处理结果。然后, 协处理器 302 中的控制电路 310 启动协处理器 302 中的驱动器 354, 在数据处理单元输出总线 125 上将协处理器处理单元 211 的处理结果数据中的一个传输给处理器 301。协处理器 302 中的控制电路 310 还按照任务信息“任务 1”启动协处理器 302 中的驱动器 255, 从而将协处理器处理单元 211 的另一个处理结果数据传输给数据处理单元输出缓冲器 332, 将处理结果存储在其中。

从时间间隔 3 切换到时间间隔 4 中，处理器 301 中的控制电路 305 在从流水线过程 (EXP1) 切换到 (WB1) 中打开寄存器写缓冲器 116，结果，处理结果之一便通过打开的寄存器写缓冲器 116 从数据处理单元输出总线 125 传输到处理器 301 中的通用寄存器 106。

在时间间隔 4 中，处理器 301 中的通用寄存器 106 存储一部分处理结果数据。

(3.4) 将协处理器 302 的一个数据处理单元输出缓冲器中的数据写入处理器 301 的通用寄存器 106 的协处理器传送指令的操作

下面说明一条协处理器传送指令的操作，该指令在任务管理单元 361 中存储任务信息“任务 2”时将存储在数据处理单元输出缓冲器 333 中的数据写入处理器 301 的通用寄存器 106 中。本操作采用与图 7 中所示的流水线处理的时间图中的状态相同的定时，因此下面的说明也参照图 7。

在时间间隔 1 中，存储器 103 在处理器 301 的取指令阶段 (IF1) 中将指令输出到导向处理器 301 与协处理器 302 的指令供给总线 126 上。

从时间间隔 1 切换到时间间隔 2 中，处理器 301 中的控制电路 305 从 (IF1) 前进到 (DP1) 并相应地打开处理器 301 的指令寄存器 112。以相同的方式，协处理器 302 中的控制电路 310 从 (IF1) 前进到 (DE1) 并相应地打开协处理器 302 的指令寄存器 119。

在时间间隔 2 中，在指令解码阶段 (DP1) 中，处理器 301 的解码器 304 在指令寄存器 112 打开之后接收指令，然后解码该指令。

在协处理器 302 的指令解码阶段 (DE1) 中, 协处理器 302 中的解码器 309 在指令寄存器 119 打开之后接收指令, 然后解码该指令。

从时间间隔 2 切换到时间间隔 3 中, 处理器 301 中的控制电路 305 在从流水线阶段 (DP1) 前进到 (EXP1) 中打开微指令寄存器 113 与数据锁存器 114。结果, 微指令便通过打开的微指令寄存器 113 从解码器 304 传输到控制电路 305, 从而用处理器 301 中的控制电路 305 执行流水线处理, 数据锁存器 114 是打开的, 并且操作数从数据处理单元输入总线 124 传输到处理器 301 中的数据处理单元 207。类似地, 协处理器 302 中的控制电路 310 在从流水线阶段 (DE1) 前进到 (EXE1) 中打开微指令寄存器 120。一旦打开了处理器 301 中的微指令寄存器 113, 处理器 301 中的控制电路 305 便接收到解码器 304 在指令解码阶段 (DP1) 中所解码的微指令。类似地, 一旦打开了微指令寄存器 120, 协处理器 302 中的控制电路 310 便接收到解码器 309 在指令解码阶段 (DE1) 中所解码的微指令。

在时间间隔 3 中, 在处理器 301 的执行阶段 (EXP1) 中, 处理器 301 中的数据处理单元 207 并不执行数据处理 (NOP)。

在协处理器 302 的指令执行阶段 (EXE1) 中, 将作为任务信息存储在任务管理单元 361 中的“任务 2”输出到协处理器 302 的控制电路 310。

协处理器 302 中的控制电路 310 按照任务管理单元 361 输出的“任务 2”启动驱动器 258, 从而在数据处理单元输出总线 125 上传输存储在数据处理单元输出缓冲器

333 中的处理结果数据。

从时间间隔 3 切换到时间间隔 4 中，处理器 301 中的控制电路 305 在从流水线过程 (EXP1) 切换到 (WB1) 中打开寄存器写缓冲器 116，结果，处理结果便通过打开的寄存器写缓冲器 116 从数据处理单元输出总线 125 传输到处理器 301 中的通用寄存器 106。

在时间间隔 4 中，处理器 301 中的通用寄存器 106 存储一部分处理结果数据。

应当指出，虽然在本实施例中控制电路 310 用任务管理单元 361 来控制任务的切换，也可能通过向该装置提供一条来自装置外部的任务切换信号线而进行任务切换，使得任务切换能从外部用一个任务切换信号控制。

类似地，虽然第一至第三实施例描述了指定寄存器作为协处理器指令的操作数数据的情况，在指定存储器单元作为操作数数据时也能完成相同的操作。在这一情况中，处理器必须执行存储器存取，这意味着执行速度不比传统的技术更好，但是不再需要在协处理器上装设用于控制存储器存取的电路，从而电路的规模并因而成本，是小于传统的电路的。

虽然已经参照附图用示例的方式全面地描述了本发明，应指出对于熟悉本技术的人员，各种改变与修正是显而易见的。因此，除非改变与修正脱离本发明的范围，都应视为是这里所包含的。

图1

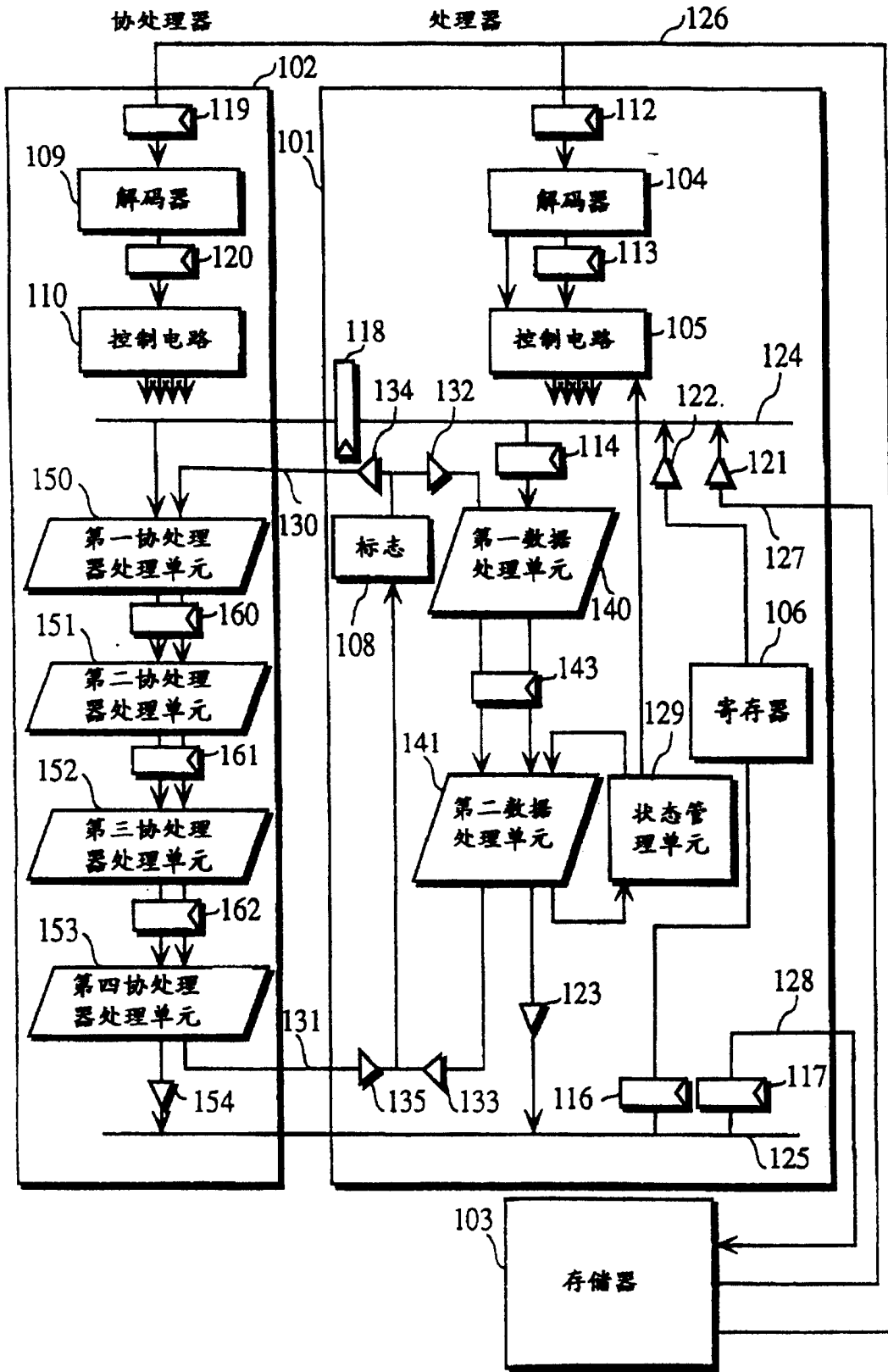


图2

| 助记符 | 指令码 | 操作 |
|---------------|--|--|
| MULQ Dm,Dn | $\begin{array}{c} \text{F60X(H)} \\ \uparrow \\ \text{源,目的} \end{array}$ | $\text{Dm} \times \text{Dn} \rightarrow \text{Dn(低 32),MR}$ |
| MULQ imm8,Dn | $\begin{array}{c} \text{F90XXXX(H)} \\ \uparrow \quad \uparrow \\ \text{imm8} \\ \text{源,目的} \end{array}$ | $\begin{array}{c} \text{(imm8 码扩充)} \\ \rightarrow \text{Dn(低 32),MR(高 32)} \end{array} \times \text{Dn}$ |
| MULQ imm16,Dn | $\begin{array}{c} \text{FB0XXXXXX(H)} \\ \uparrow \quad \uparrow \\ \text{imm16} \\ \text{源,目的} \end{array}$ | $\begin{array}{c} \text{(imm16 码扩充)} \\ \rightarrow \text{Dn(低 32),MR(高 32)} \end{array} \times \text{Dn}$ |
| MULQ imm32,Dn | $\begin{array}{c} \text{FD0XXXXXXXXXX(H)} \\ \uparrow \quad \uparrow \\ \text{imm32} \\ \text{源,目的} \end{array}$ | $\begin{array}{c} \text{(imm32 码扩充)} \\ \rightarrow \text{Dn(低 32),MR(高 32)} \end{array} \times \text{Dn}$ |

图3

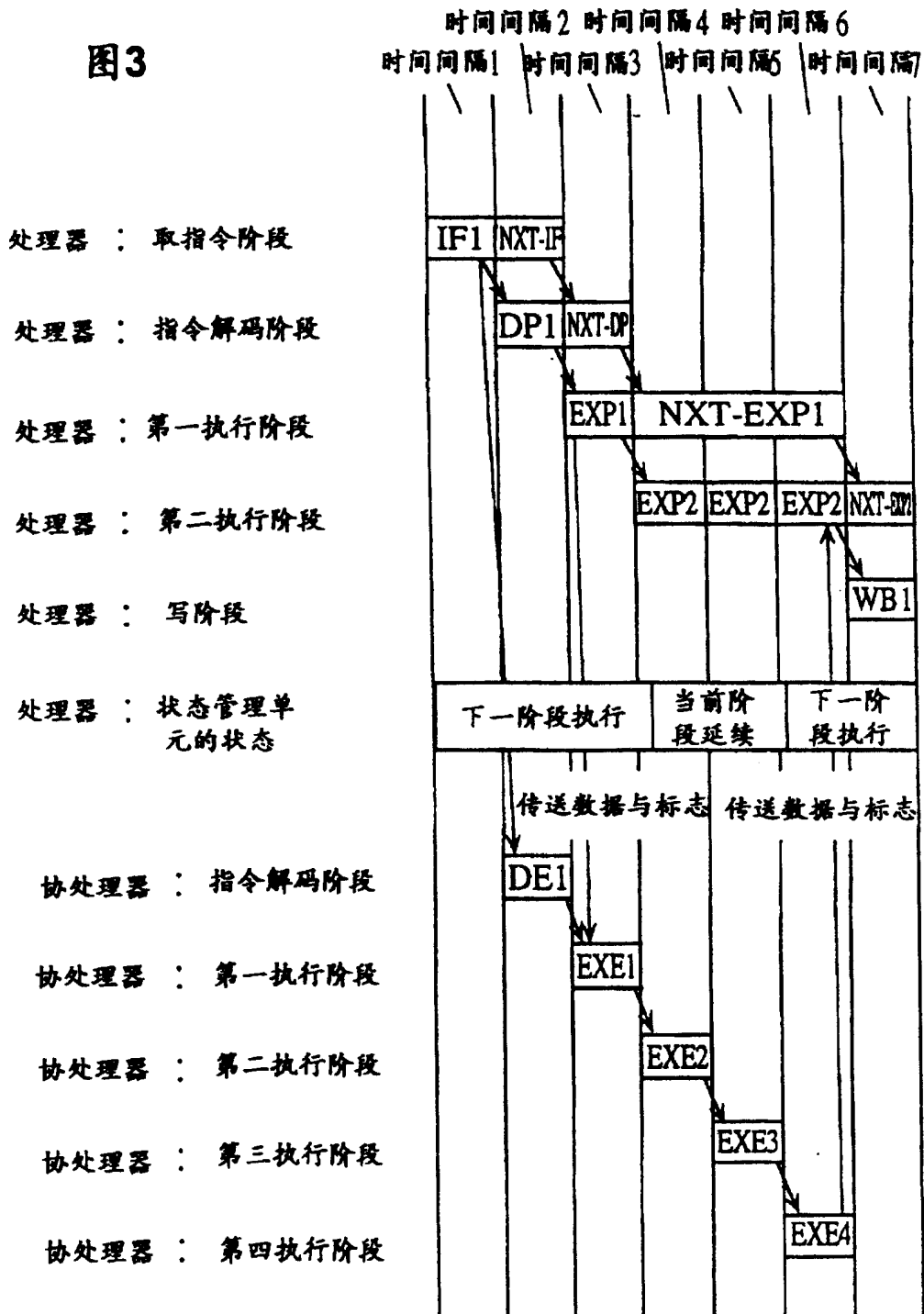


图4

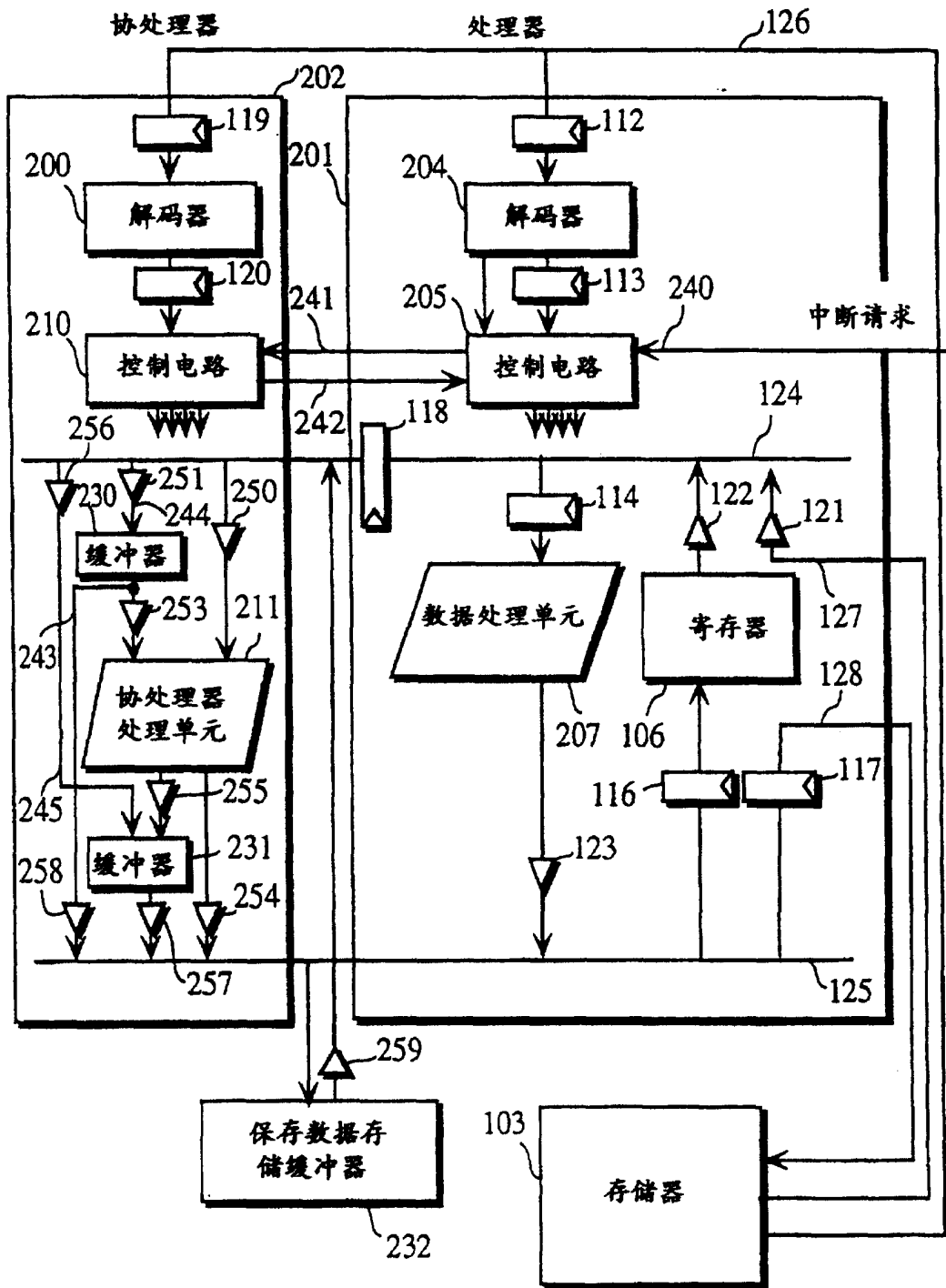


图5

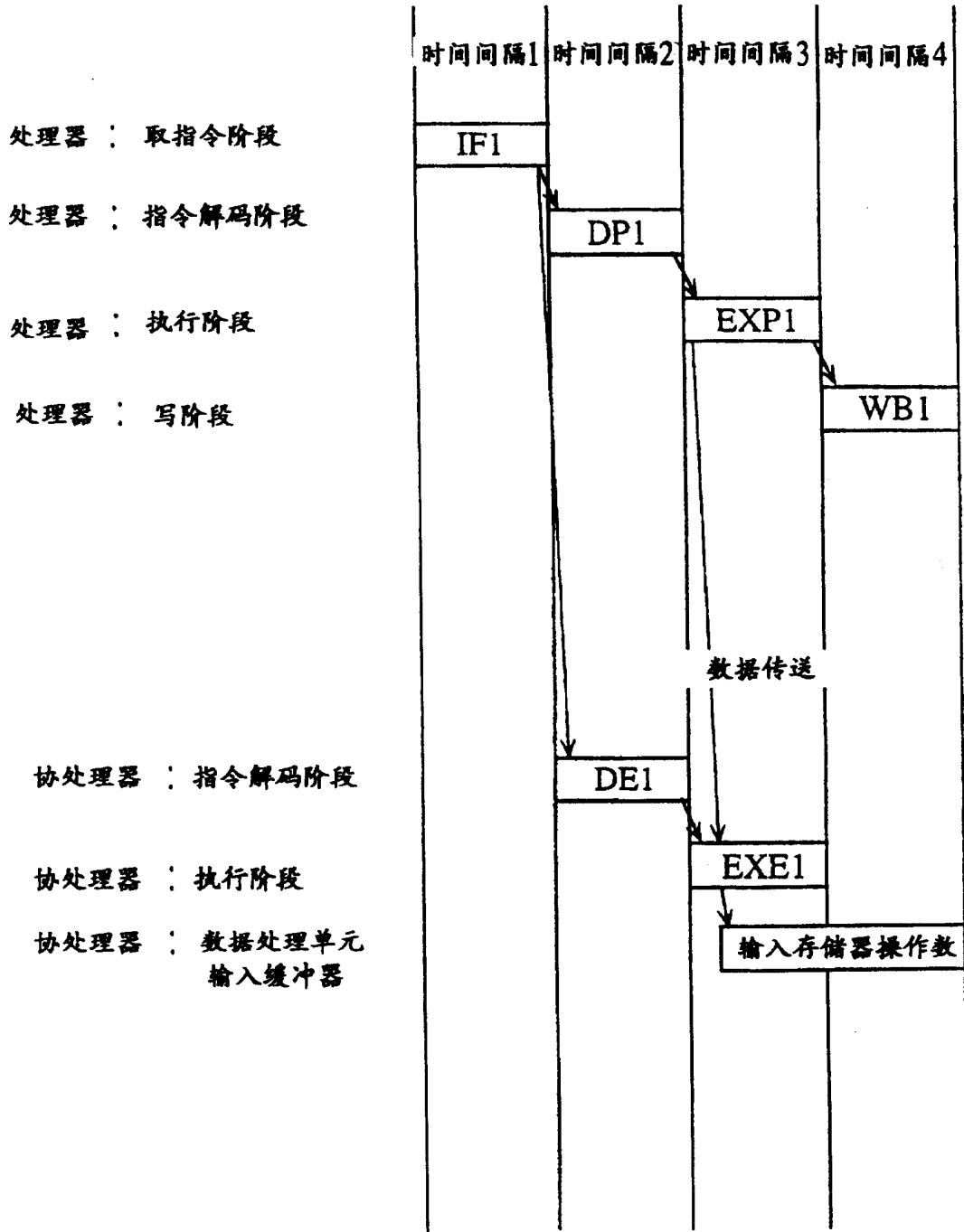


图6

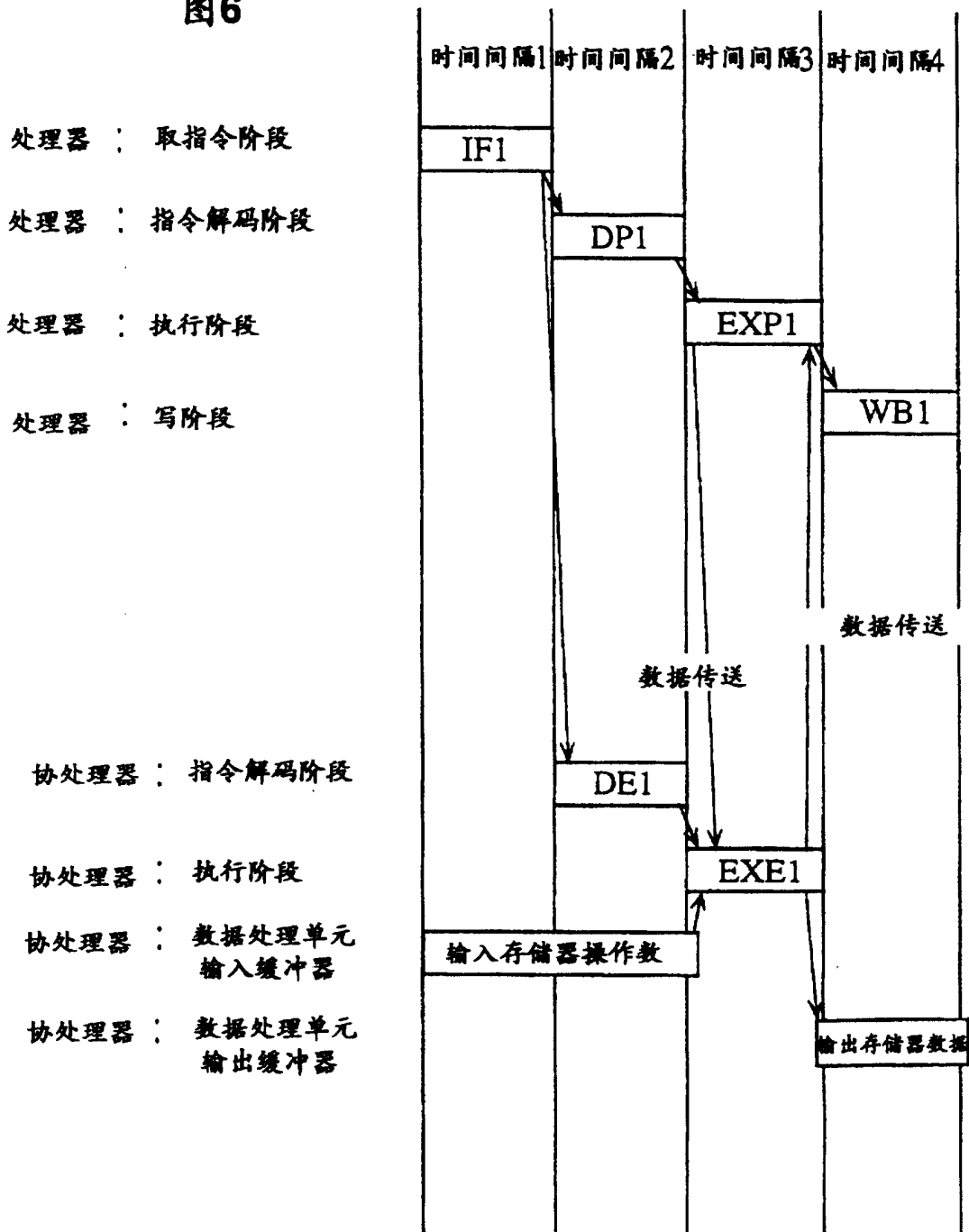


图7

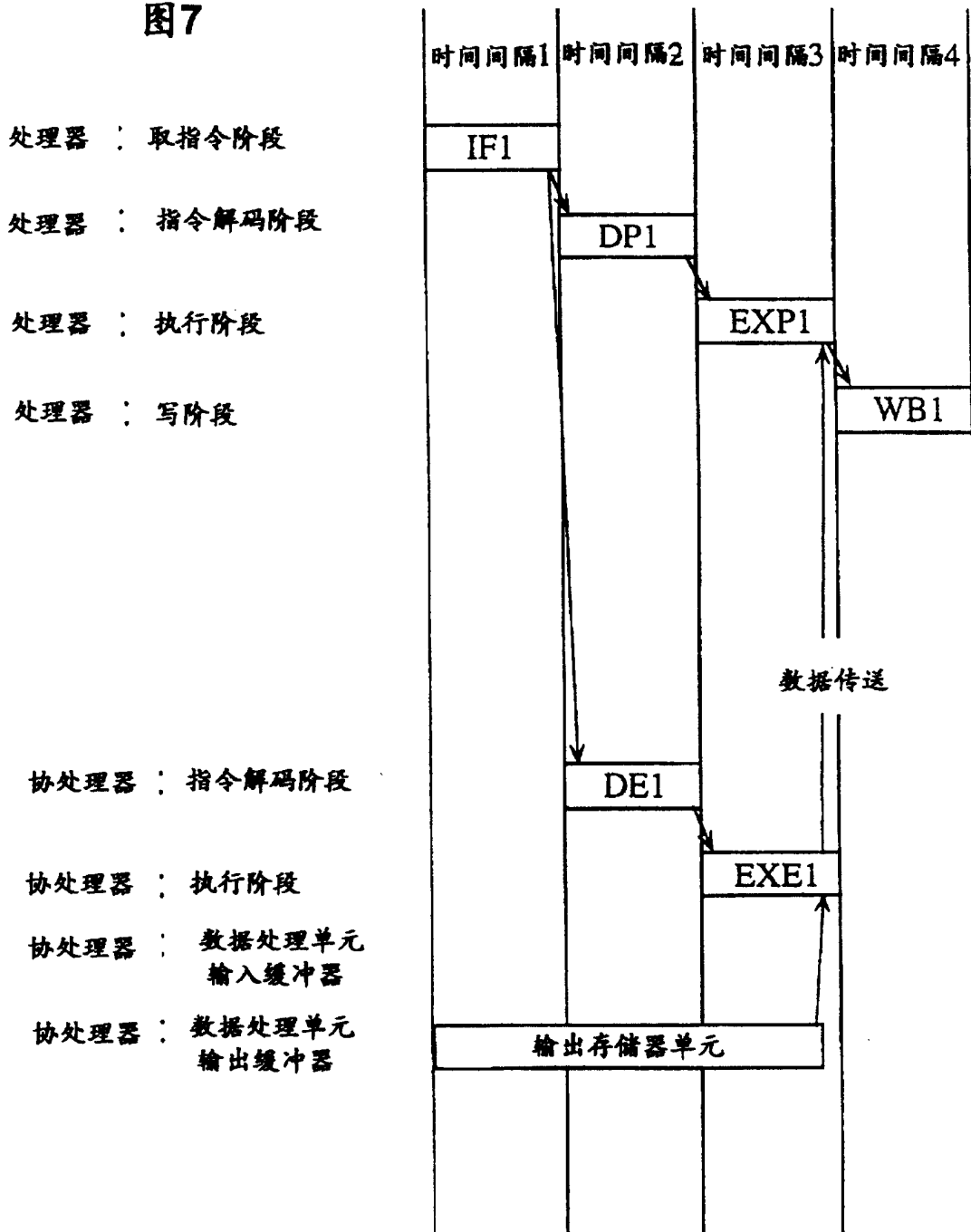


图8

处理器：来自外部的中断请求信号线

从处理器到协处理器的中断信号线

从处理器到协处理器的中断过程完成信号线

协处理器：控制电路

协处理器：数据处理单元输出总线

协处理器：数据处理单元输入缓冲器

协处理器：数据处理单元输出缓冲器

第一保存数据存储缓冲器

第二保存数据存储缓冲器

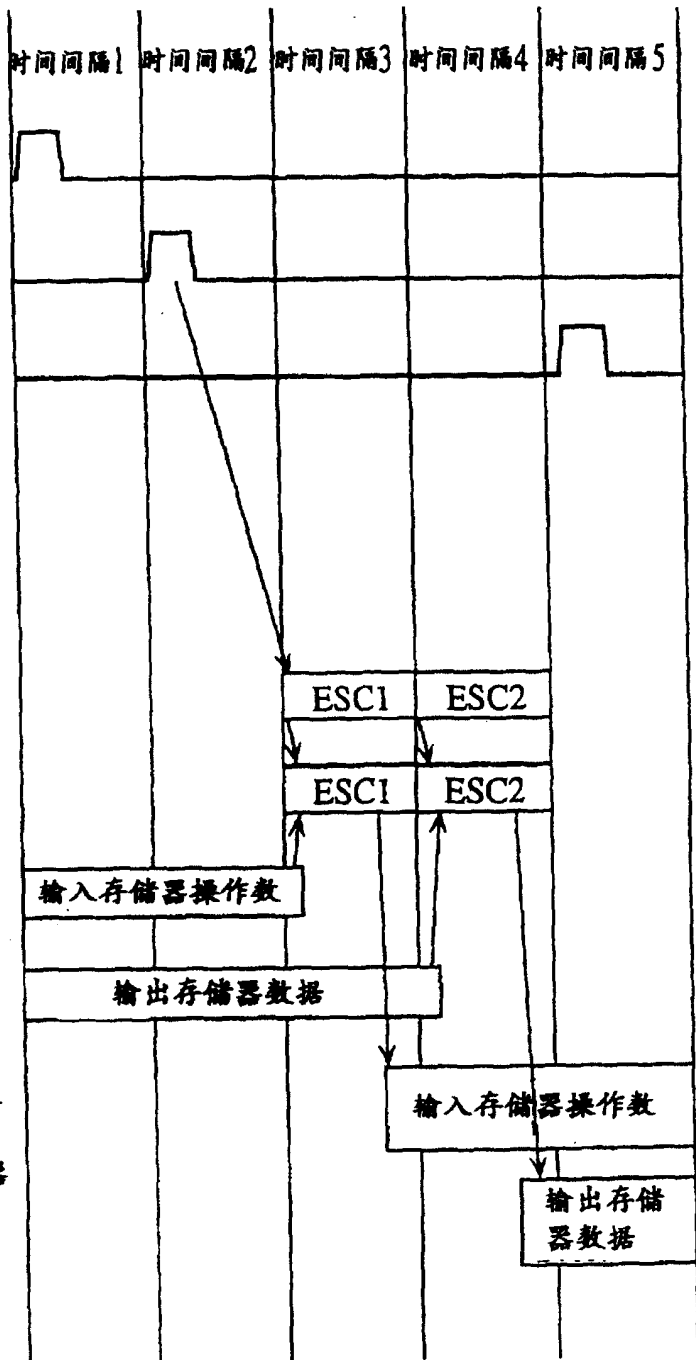


图9

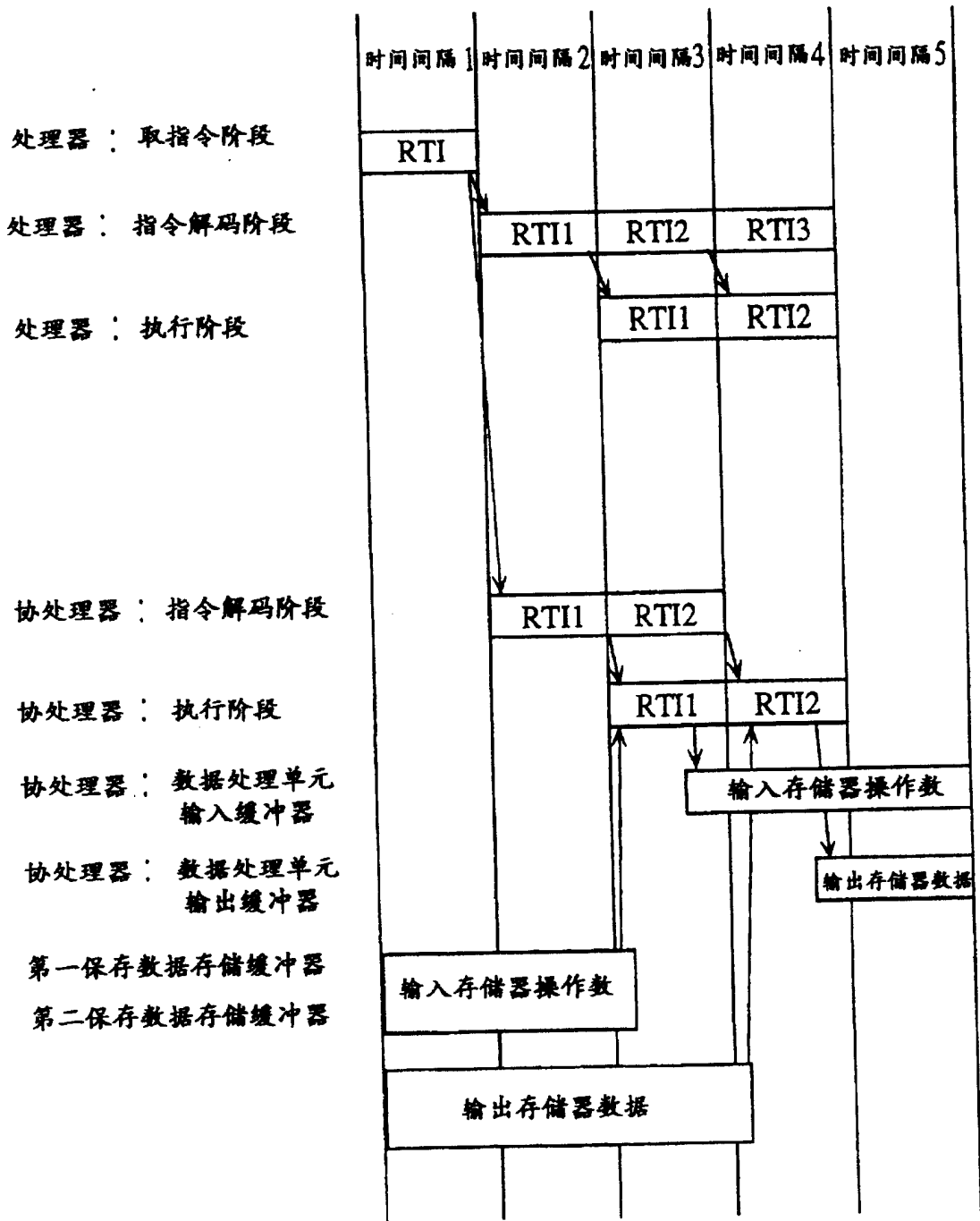


图10

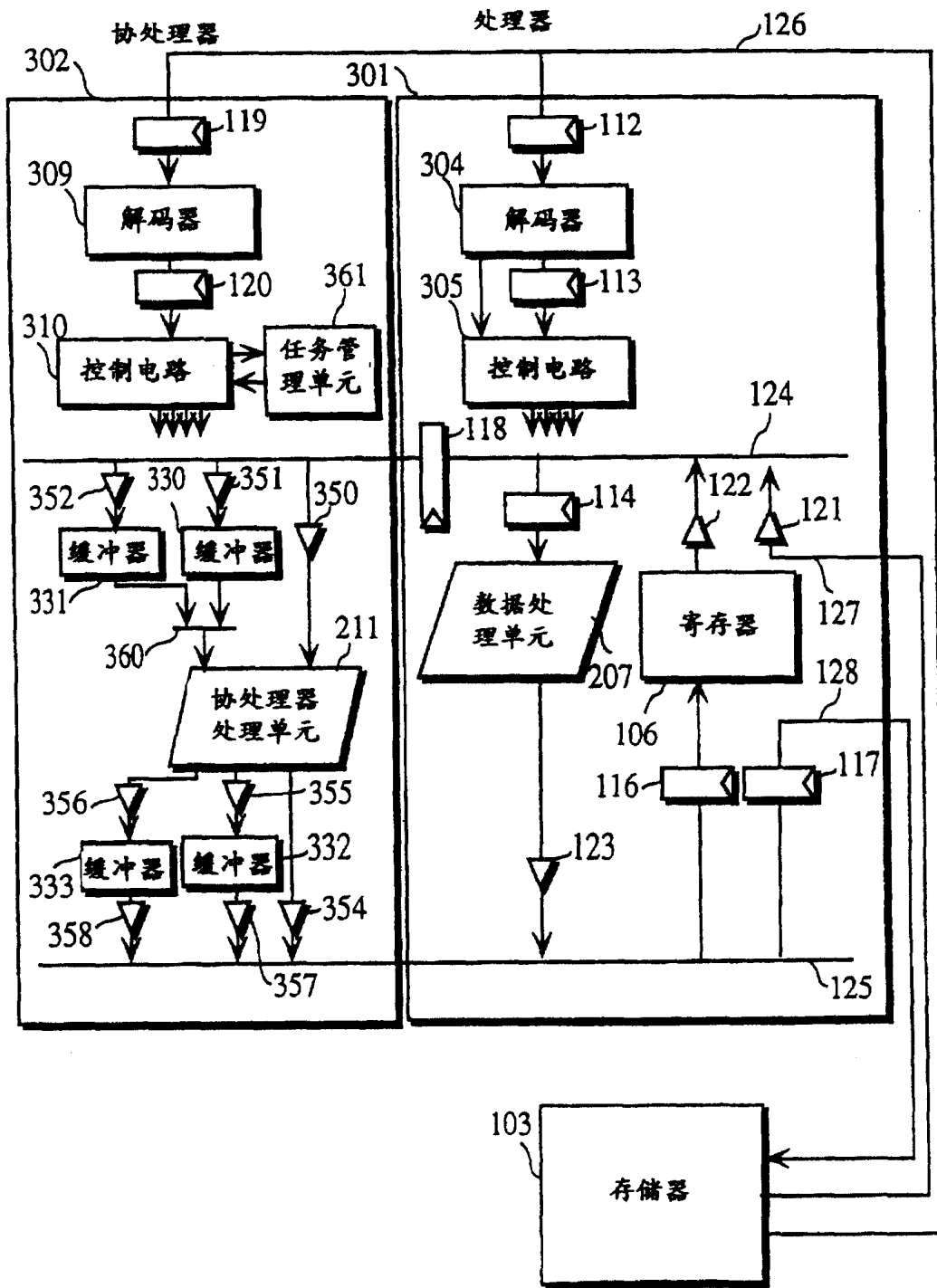


图11

