



# [12] 发明专利说明书

专利号 ZL 00814443.5

[45] 授权公告日 2005 年 12 月 7 日

[11] 授权公告号 CN 1230735C

[22] 申请日 2000.12.4 [21] 申请号 00814443.5

[30] 优先权

[32] 1999.12.23 [33] US [31] 09/470,836

[86] 国际申请 PCT/US2000/032887 2000.12.4

[87] 国际公布 WO2001/048595 英 2001.7.5

[85] 进入国家阶段日期 2002.6.24

[71] 专利权人 英特尔公司

地址 美国加利福尼亚州

[72] 发明人 Y·寥 T·M·哈梅南蒂拉

D·B·罗伯茨

审查员 覃冬梅

[74] 专利代理机构 中国专利代理(香港)有限公司

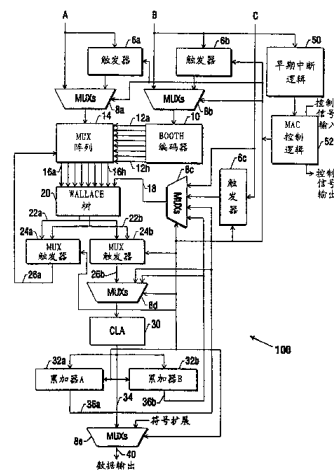
代理人 吴立明 王忠忠

权利要求书 2 页 说明书 10 页 附图 8 页

[54] 发明名称 在一个周期内处理乘累加运算

[57] 摘要

一个乘法累加器，或 MAC(100)，可以实现高吞吐量。MAC 不需要使用冗余硬件，如多个乘 Wallace 树，或流水线逻辑，然而，可能同时为不同操作执行 Wallace 树和进位预测加法器功能。



ISSN 1008-4274

1. 一种用于处理乘累加运算的设备, 包括:  
一个对第一操作数和第二操作数执行多个加运算的装置;  
一个加法器; 和
- 5 一个控制器, 它允许使用该装置对第一操作数的一部分执行第一数学运算, 同时加法器对第一操作数的另一个部分执行第二数学运算, 其中该控制器根据第二操作数的多个最高有效位是否相同来允许第一数学运算和第二数学运算。
  2. 如权利要求 1 所述设备, 进一步包括一个与所述加法器耦合的  
10 内部存储媒体。
    3. 如权利要求 1 所述设备, 其中所述装置包含 Wallace 树。
    4. 如权利要求 3 所述设备, 其中所述 Wallace 树包含多个进位保留加法器。
    5. 如权利要求 4 所述设备, 其中所述 Wallace 树是一个 4 阶段  
15 Wallace 树。
    6. 如权利要求 1 所述设备, 其中所述加法器是一个先行进位加法器。
    7. 如权利要求 1 所述设备, 进一步包括与所述装置耦合的 Booth 编码器。
    - 20 8. 如权利要求 4 所述设备, 其中所述 Wallace 树接收 9 个输入向量并产生两个输出向量。
    9. 如权利要求 2 所述设备, 其中所述内部存储媒体包括一或多个累加器。
    10. 如权利要求 9 所述设备, 其中所述一或多个累加器与所述加  
25 法器的输出耦合。
  11. 一种用于处理乘累加运算的设备, 包括:  
一个对操作数对的集合执行多个加运算以获得第一结果的装置;  
一个加法器; 和  
一个控制器, 其根据操作数对的第一集合中的至少一个集合的多个最高有效位是否相同, 来允许使用该装置对操作数对的第二集合执行第一数学运算, 同时加法器对操作数对的第一集合的第一结果执行第二数学运算。  
30

- 
12. 如权利要求 11 所述设备, 其中所述装置包含 Wallace 树。
  13. 如权利要求 11 所述设备, 其中所述加法器是一个先行进位加法器。
  14. 如权利要求 11 所述设备, 进一步包括与所述装置耦合的 Booth  
5 编码器。
  15. 如权利要求 11 所述设备, 其中一或多个累加器与所述加法器的输出耦合。

## 在一个周期内处理乘累加运算

### 技术领域

5 本发明关于基于处理器的系统，更特指，乘法累加单元。

### 背景技术

一个乘法累加单元，或 MAC，是为执行乘法操作设计的单元。在基于处理器的系统中，此类操作可能很多。

10 MAC 可能用多种逻辑实现。由于有多个乘法运算存在于应用程序中，通常选用最快的 MAC。任何 MAC 的设计成功可能与其耗电量，MAC 电路所在位置的大小限制，MAC 的用途及其它因素相关。

通常，然而，速度和成本之间的折衷使优化执行的 MAC 的设计很困难。其成本以附加硬件，容纳硬件所需的芯片空间或驱动硬件所需的耗电量的形式出现。

例如，在很多数字信号处理应用中，或 DSP，一些重要的操作涉及一系列的乘和加操作。因此，高吞吐量的 MAC 对获得高性能是重要的。然而，很多当今的 DSP 应用要求低耗电量，特别是在便携式电子市场。

因此，对能实现高吞吐量而又不过多耗电的 MAC 是一种持久的需求。

### 发明内容

本发明的一个实施例中，一种方法包括在单元的第一部分接收操作数的第一集，在操作数的第一集上执行多个操作产生第一个中间结果并在单元的第一部分接收操作数的第二集。之后，在第一中间结果上执行第二个多个操作产生第一个最终结果，同时第二个多个操作在操作数的第二集上执行产生第二个中间结果。

在本发明的另一个实施例中，一种用于处理乘累加运算的设备包括：一个对第一操作数和第二操作数执行多个加运算的装置；一个加法器；和一个控制器，它允许使用该装置对第一操作数的一部分执行第一数学运算，同时加法器对第一操作数的另一个部分执行第二数学运算，其中该控制器根据第二操作数的多个最高有效位是否相同来允许

第一数学运算和第二数学运算。

在本发明的另一个实施例中，一种用于处理乘累加运算的设备，包括：一个对操作数对的集合执行多个加运算以获得第一结果的装置；一个加法器；和一个控制器，其根据操作数对的第一集合中的至少一个集合的多个最高有效位是否相同，来允许使用该装置对操作数对的第二集合执行第一数学运算，同时加法器对操作数对的第一集合的第一结果执行第二数学运算。

本发明的优点和其他特性由下面叙述，图和权利要求中变得明显。

10

#### 附图说明

图 1 为按本发明一个实例的乘累加器的方块图；

图 2A 和 2B 是使用按照本发明的一个实例如图 1 的 MAC 执行的乘法操作的流程图；

15 图 3 是使用按照本发明的一个实例如图 1 的 MAC 的 16 位乘法操作的执行时间表；

图 4A-4C 是使用按照本发明的一个实例如图 1 的 MAC 的一个不指明的被执行的乘法操作的流程图；

20 图 5 是使用按照本发明的一个实例如图 1 的 MAC 的被执行的包装后数据的乘法操作的流程图。

#### 具体实施方式

从最基本的角度讲，MAC 接收两个整数值，把两个值相乘并送出操作的结果。对一些应用来说，附加的常数，一个累加值，可能被加到结果中。所以，MAC 将生成结果 d，由被乘数 a 乘以乘数 b 加累加值 c，如以下等式所示：

$$d=(a*b)+c$$

乘法操作可按一系列部分积的和的方式执行。为获得优化的性能，MAC 可能包括执行更快的附加操作或减少必备附加操作数量的特点。

30

例如，典型的 MAC 包括一个 Booth 编码器，也称为 Booth 重编码器或 Booth 乘法器。Booth 编码器可减少执行乘法步骤的数量。被相

加的部分积的数量也因此减少。例如，16 位长字，部分积的数量可由 16 减少到 8，除了二。

MAC 中另一个特性被称为 Wallace 树。Wallace 树是执行多个加操作的结构。Wallace 树通常包括特性，如加法器，帮助执行多个部分积的加法操作。

例如，Wallace 树可能包括一组进位保留 (carry-save) 加法器。一个 3 至 2 进位保留加法器，例如，接收 3 个输入 a, b 和 c, 产生两个输出，一个是和 y, 一个是进位 z, 如以下等式：

$$a+b+c=y+z*2$$

10 输出进位是为下个操作被省略的，因此命名。

Wallace 树可能组织了进位保留加法器或相似逻辑，如一组输入值通常产生两个结果，和值和进位值。例如，四阶段 Wallace 树可能包括四阶段 3 至 2 进位保留加法器。该四阶段 Wallace 树可能接收九个输入向量，并生成一个和向量和一个进位向量作为结果。

15 许多 MAC 设计的另一个特性是加所有两个向量位的逻辑以生成最后结果，例如，一个先行进位 (carry look-ahead) 加法器，或 CLA, 是一种这样的逻辑设备。CLA 能有效的执行两个多位操作数的加法。

特别是对于数字信号处理和其他应用，MAC 以很快的速度乘 16 位带符号整数的能力是一个很重要的考虑因素。为提高 16 位或更大整数操作数的乘法运算的速度，MAC 可包括附加电路。例如，MAC 的每个组件可被复制，如此多个操作可同时运行。控制并行操作的控制逻辑也是 MAC 的一部分。然而，附加硬件可能很贵，可获得空间可能有限，支持性硬件的附加电源可能被使用。

25 另一个 MAC 设计是使操作上流水线，这样一种新的乘操作可能在预先操作还没有被 MAC 硬件完成之前在流水线上某处被处理。流水线化可能涉及几个被执行操作的阶段，每个阶段独立操作。流水线化因此可能使用附加的控制逻辑和硬件，如寄存器，为暂时存储两个，三个或多个由 MAC 硬件同时执行的独立操作状态。虽然比提供并行硬件便宜，附加的控制逻辑可能增加开发成本和减少可获得的板上或芯片上空间。再有，支持流水线的 MAC 运行更慢。

30 更高吞吐量的 MAC 可能使用常规的结构特性实现，而又不过分的影响硬件的成本或电源需求。在本发明的一些实现中，MAC 设计提供的

为同时操作 MAC 的两个功能单元的 16 位乘法操作的单周期吞吐量。

图 1 中,按本发明的一个实现,MAC100 包括一个 Booth 编码器 10,一个多路复用器,或 MUX,阵列 14,一个 Wallace 树 20,和一个先行进位加法器,或 CLA,30。MAC100 可接收两个输入值,被乘数 A 和乘数 B,还有累加数据值 C。输入值 A,B,C 可为 16 位,32 位或更长的带符号整数。

在本发明的一个实施例中,Booth 编码器一次接收长至乘数 B 的 16 位,因此生成多至 8 个控制信号 12a-12h。MUX 阵列 14 在接收被乘数 A 的同时由 Booth 编码器 10 接收 8 个控制信号 12a-12h。该 MUX 阵列 14 产生多至 8 个部分积,16a-a6h,并被发送到 Wallace 树 20。

在本发明的一个实施例中,该 Wallace 树 20 是个四阶段单元,包括一些进位保留加法器(没有显示)。该 Wallace 树 20 因此接收 9 个输入信号并输出两个输出信号,一个和向量 22a 和进位向量 22b。和向量 22a 和进位向量 22b 被连续的输入 CLA30,并被相加。连续的或同时的,和向量 22a 和进位向量 22b 可被送回 MUX 阵列 14 进行进一步处理。

该 MAC100 包括 3 个触发器库 6a-6c。触发器 6 使被乘数 A,乘数 B 和累加数据 C 的部分被发送到 MAC100 的其他逻辑部分进一步处理。例如,如果 32 位乘数 B[31:0]被发送到 MAC100 作为乘法操作的一部分,触发器 6b 可在两个 16 位的部分, B[31:16]或 B[15:0],之间转换,控制乘数 B 的那部分被发送到 Booth 编码器 10。

该 MAC100 也包括 MUX8A-8E。该 MUX8 可通过 MAC100 的后来的单元控制信号的接收。例如, MUX8a-8c 与触发器 6a-6c 连接工作以控制被乘数 A,乘数 B,和累加数 C 的哪部分被 MUX 阵列 14, Booth 编码器 10 或 Wallace 树 20 分别一一对应的接收。该 MUX8d 通过 CLA30 控制和向量 22a 和进位向量 22b 的接收。该 MUX8e 使由 CLA30 来的输出值 34 在需要时成为符号扩展的。

该 MAC100 包括两个 MUX 的库和由 Wallace 树接收输出信号 22a 和 22b 的触发器 24a 和 24b。该 MUX 和触发器 24a 被用于输入和向量 22a 和进位向量 22b 回 MUX 阵列 14。依执行操作的不同,26a 回馈 MUX 阵列 14。该反馈是多阶段 MAC 操作的通常特点,如为执行 32 位乘法时。

该 MAC100 包括一个第二 MUX 和触发器 24b,和向量 22a 和进位向

量 22b 发到此处。然后，这些值被通过 MUX8d 发送到 CLA30。在本发明的一个实现中，和向量 22a 和进位向量 22b 被同时发送到单元 24a 和 24b。

在本发明的一些实施例中，该 MAC100 包括结构上的提高，此提高可能提高乘法操作的吞吐量。例如，本发明的一个实现中，该 MAC100 包括两个寄存器，累加器 A32a 和累加器 B32b。该累加器 32 可由 CLA30 接收结果向量 34。本发明的一个实现中，结果向量 34 可存于累加器 32 中。例如，在第一个操作中，可能为中间结果的结果向量，被存于累加器 A32a 中。下一步操作中，后一个结果向量被存于累加器 B32b 中。

该累加器 32 都为递交到 Wallace 树连接到 MUX8c。该配置允许存于任意累加器 32 的值被作为累加值递交到 Wallace 树而不是由 MAC100 的外部源 (C) 接收累加数据。依以下将显示，累加器 32 可帮助本发明一个实现中 MAC100 中执行操作的吞吐量。

该 MAC100 可支持不明确的指令如不明确的乘法指令。不明确的指令是一种指令在此一个或多个中间结果被存于 MAC 的内部，如在寄存器中。包括在 MAC 电路中的寄存器允许 MAC 避免通过外部总线发送中间结果到 MAC 外的存储位置。因此，支持不指明操作 MAC 可处理指令比无此种支持的 MAC 更快。

对很多乘法操作来说，MAC 电路在得到最终结果之前可提供多个中间结果。传统的 MAC 设计通常由 Wallace 树接收一个中间结果，一个中间和向量和一个中间进位向量，为下一步处理返回结果到 MUX 阵列。该方案在必要时被重复。最后，由 Wallace 树生成的和向量和进位向量代表了最终结果。然后，该值被发送到 CLA，在此他们被加到一起后被发送到 MAC 电路作为最终结果。

在图 1 的实现中，如需要的话，中间和向量 22a 和中间进位向量 22b 在一个周期内可被发送到 CLA30。该 CLA30 可能使中间和向量 22a 和中间进位向量 22b 相加，产生存于累加器 32 之一中的中间结果 34。

该 MAC100 设计中的反馈逻辑允许中间结果 34 作为累加数据 18，通过 MUX8c，由累加器 32 之一返回 Wallace 树 20。回想起 Wallace 树 20 包括为累加数据 18 的输入线 18。图 1 中，MUX8c 允许累加数据 18 的源到包括累加器 32。



本发明的一个实施例中,MAC100 提供允许同时操作 Wallace 树 20 和 CLA30 的电路。本发明的一个实现中,该电路包括早期中止逻辑 50 和 MAC 控制 52。该电路通过 MAC100 可控制多个乘法操作或单个乘法操作的多个部分的同时处理。

5 在典型的实现中,MAC 可接收带符号的二进制值作为操作数。例如,一个二的补充符号可能被使用。在二的补充符号中,最重要的值中的位可代表符号:零是正值,一为负值。因此,在二的补充符号中,16 位的操作数 FFC1h 可代表-63d 和操作数 003Fh 可代表+63d。

另外,在一些应用中,16 位值可由 32 位值代表。符号位(第十五  
10 位)可因此在十六位上扩展或复制。因此,符号扩展操作数 FFFFFFFC1h 可代表-63d,操作数 0000003Fh 可代表+63d。两个操作数的高十七位是一致的,如符号位(第十五位)被扩展。同样的,对两个值来说,高十六位不需要执行乘法操作。

在本发明的一个实施例中,早期终结逻辑 50 使用这一带符号二进  
15 制值的特性为 MAC100 的优势。在图 1 中,早期终结逻辑 50 在 32 位的乘数 B 的高十七位相同时被启用。在这种情况下,乘数 B 的高十六位在乘法运算中没有被用到。

一旦激活,早期终端逻辑 50 发送信号到 MAC 控制逻辑 52。该 MAC  
20 控制逻辑 52 控制触发器 6, MUX8 和 MUX 和触发器 24,还有累加器 32。该 MAC 控制逻辑 52 因此是下一个被乘数 A 通过触发器 6a 和 MUX8a 相连,下一个乘数 B 通过触发器 6b 和 MUX8b 相连,或下一个累加数 C 通过触发器 6c 和 MUX8c 相连。

另外,该 MAC 控制逻辑 52 可控制由 Wallace 树 30 接收的中间和  
25 向量 22a,中间进位向量 22b 和中间进位向量的路径。该 MAC 控制逻辑 52 可使 MUX 和触发器 24a, MUX 和触发器 24b,和 MUX8d 在需要时有效。

该 MAC 控制逻辑 52 进一步控制累加器 A 寄存器 32a 和累加器 B 寄  
30 存器 32b,以允许在其中的值被作为累加输入 18 送回 Wallace 树 20。最后,该 MAC 控制逻辑 52 可控制 MUX8e,如需要最终输出向量 34 可为符号扩展的,并生成输出值 40。

在本发明的一个实施例中,该 MAC 控制单元 52 可由 MAC100 的外  
部源接收控制信号,如一个指令解码器(没有显示)。该 MAC 控制单

元 52 也可提交控制信号到另一个 MAC100 的外部源。

依据被执行操作的不同，该 MAC100 可产生中间向量值。例如，本发明的一个实现中，Booth 编码器 10 由乘数 B 接收一个 16 位值。因此，为了乘两个 32 位数，或一个 16 位被乘数和一个 32 位乘数，该 Booth 编码器 10 可在第一个周期接收乘数的低 16 位然后在第二个周期接收乘数的高 16 位。同样，MAC100 的其他电路可在第一和第二周期都接收数据。该乘法操作因此只用了最少的两个周期。

对于一些操作来说，该 MAC100 可在完成单周期的输入输出。在本发明的一个实现中，MAC100 在执行 16 位乘法操作时，可获得单周期的输入输出。在操作数的第一集被接收到 MAC100 后，Wallace 树 66 和 CLA68 可同时处理后面的操作数。例如，Wallace 树可在 CLA68 为周期  $n-1$  处理数据时，处理周期  $n$  的数据。

图 2A 中，一个流程图阐明了 MAC100 逻辑如何处理乘法或乘/累加操作。在该例中，两个 16 位操作数是符号扩展后的如 32 位值的形式。该操作数在 DSP 应用中是常见的且可特别说明在 MAC100 中的早期中止逻辑 50 的性能。然而，MAC100 可在具有其他特点的操作数上执行乘或乘/累加操作。

第一个周期，即 0 周期，开始（块 200）。在本发明的一个实现中，乘数 B[15:0] 低十六位被发送到 Booth 编码器 10（块 202）。MUX 阵列 14 由 Booth 编码器 10 接收八个控制信号 12 和被乘数 A[31:0]（块 204）。MUX 阵列 14 处理这些信号，Wallace 树 20 从 MUX 阵列 14 接收 8 个部分积向量 16。另外，该 Wallace 树 20 接收累加数据 C[31;0]（块 206）。从这些数据中，Wallace 树 20 产生一个中间和向量（ISV）22a 和中间进位向量（ICV）22b（块 208）。该中间和向量 22a 和中间进位向量 22b 被发送到 MUX 和触发器 24a 和 24b（块 210）。

独立于前述操作，在 0 周期的开始（块 200），早期中止逻辑 50 可接收 32 位的乘数 B[31:0]（块 212）。因为操作数是符号扩展的 16 位值，乘数 B 的高 17 位是一样的。因此，早期中止逻辑 50 发送信号以连接触发器 6a，6b 和 6c 还有 MUX8a，8b 和 8c（块 216）。此方式中，MAC 控制逻辑 52 电路连接下一个 MAC 操作数 A，B 和 C。

在图 2B 中，第二周期，即 1 周期，是图 2A 操作的延续。在 1 周期开始时（块 230），MAC 控制逻辑 52 发送信号到 MUX8d 以发布中间

和向量 22a 和中间进位向量 22b 并发送到 CLA30(块 234)。因此,CLA30 将中间和向量 22a 和中间进位向量 22b 相加(块 236)。然后,结果 40 被发送出 MAC100(块 238)。因此,1 周期完成。

5 在图 3 中,按本发明的一个实施例使用 MAC100 的 16 位乘法操作的单周期输入输出可由图表说明。在图表的顶部,一系列 MAC100 执行的标记数字后的乘法操作被描述。在图表的左部,在该执行的周期是被特指的。按本发明的一个实现,该图表追踪 Wallace 树 20 执行的操作,被标为“W”,由 CLA30 执行的被标为“C”。

对第一个 16 位乘法操作,当 CLA30 在 1 周期执行时,Wallace 树 10 20 在 0 周期执行。对第二个 16 位乘法操作,当 CLA30 在 2 周期执行时,Wallace 树 20 在 1 周期执行。CLA30 对一个乘法操作的执行和 Wallace 树 20 对第二个 16 位乘法操作的执行是同时进行的。如该图表所示,后面的 16 位乘法操作如以上两个已描述的方式进行。在第 15 周期结束时,15 个 16 位乘法操作已完成。对于本发明一个实现的一些 15 乘法操作,MAC100 因此提供了单周期的输入输出。

累加器 32 允许 MAC100 执行不明确的乘法和乘/累加操作。在第二个实现中,MAC100 可包括一个单累加器。累加器 32 可由几个不明确指令以不指明的方式作参数。那种不指明的乘法/累加指令引用了累加器 32 而不是某个特殊的寄存器作为累加器。

20 例如,本发明一个实施例中的指令, MAR 或 MRA 可由 MAC 控制逻辑 52 为了累加器 32 可被分别读出和写入而接收。在图 4A 中,流程图显示了不指明的乘法/累加操作,在此早期中断被启用。回忆以前,为了早期中断被启用,乘数 B 的另外 17 位都为 0 或都为 1。

当 0 周期开始时(块 300),乘数 B[15:0]被发送到 Booth 编码器 25 10(块 302)。然后,MUX 阵列 14 由 Booth 编码器 10 接收 8 个控制信号 12 和被乘数 A[31:0](块 304)。

Wallace 树由 MUX 阵列 14 接收 8 个部分积。另外,Wallace 树 20 由累加器 A32a 和累加器 B32b 接收累加值。然后,Wallace 树产生一个中间和向量 22a 和一个中间进位向量 22b(块 308)。

30 在本发明的一个实施例中,先行进位加法器 30 是一个 40 位的 CLA,可执行 16 位的 DSP 操作。同样,累加器 A 寄存器 32a 和累加器 B 寄存器 32b 是 40 位寄存器。因此,在下一步操作中,中间和向量 22a

和中间进位向量 22b 的低 40 位被发送到触发器 24a 和 24b (块 310)。

5 仍然在 0 周期的开始, 整个 32 位的乘数 B 被发送到早期中断逻辑 (EFL) 50 (块 312)。因为乘数 B 的高 17 位是一致的, 早期中断逻辑 50 发送信号到 MAC 控制逻辑 52 (块 314)。然后, MAC 控制逻辑 52 发送信号以连接触发器 6a, 6b, 6c 和 MUX8a, 8b, 8c (块 316)。这指出 Wallace 树 20 将可以在下个周期接收下一个不指明的乘法/累加指令 (块 318)。

10 在图 4B 中, 1 周期开始 (块 330)。MAC 控制逻辑 52 发送信号到 MUX8d 以发布中间和向量 22a 和中间进位向量 22b (块 332)。然后, CLA30 使两个 22 的值相加产生一个中间结果 34 (块 334)。因为这是一个不明确的指令, 该结果 34 被存于累加器 A32a 或累加器 B32b (块 336) 中。这样就结束了 1 周期 (块 338)。

15 在本发明的一个实施例中, MAC100 可接收引用了累加器的不明确指令。这些指令可由最终控制累加器 A32a 和累加器 B32b 的 MAC 控制逻辑 52 接收。

20 例如, 在图 4C 中, MAC 控制逻辑 52 可为读取累加器 A32a 接收不指明的指令 (块 350)。在累加器 A32a 和累加器 B32b 中得知被发送到 CLA30 (块 352)。然后, CLA30 使值 36a 和 36b 相加 (块 354)。然后, CLA30 发送结果 40 作为输出数据 (块 356)。然后, 结果 40 可被发送到 MAC100 外部的寄存器文件。

25 在图 5 中, 一个带有不指明的累加操作的乘法使用打包后的数据指令。在本发明的一个实现中, MAC100 可接收 4 种不同的打包后的数据指令。这 4 个打包后的数据指令在 16 位或半个两个 32 位操作数上执行。例如, 一个指令可在乘数 B 和被乘数 A 的低 16 位上操作。最后, 每个打包后数据指令因此在两个 16 位操作数上执行一个 16 位乘法操作。

在 0 周期的开始 (块 400), 一个 32 位的被乘数 A [31:0] 和一个 32 位的乘数 B [31:0] 被发送到 MAC100 (块 402)。MAC 控制逻辑 52 也接收一个打包后数据指令 (块 404)。

30 被乘数 A 的 16 位被扩展到 32 位。例如, 如果打包后数据指令正在被乘数 A 的低 16 位上操作, 那么被乘数 A 的高 16 位由被乘数 A 的低 16 位的内容所代替。其实是符号被由第 16 位扩展到第 32 位。同样,

如果打包后数据指令在被乘数 A 的高 16 位上操作，那么高 16 位被移位到低 16 位且高 16 位被填入符号位的值。

乘数 B 的 16 位被发送到 Booth 编码器 (块 408)。例如，如果打包后数据指令在乘数 B 的低半部上操作，然后 B[15:0] 被发送到 Booth 编码器 10。同样，为在乘数 B 的高半部操作的打包后数据指令，B[31:16] 未被发送到 Booth 编码器 10。

由此来看，打包后数据指令正如任何其他不指明指令一样操作。因此，由 MAC100 执行的下一个操作在图 4A 中 (块 304)。

正如对其他操作，MAC100 同时决定是否早期中止逻辑 50 被启用。那就是乘数 B[31:0] 被发送到早期中止逻辑 50 (块 420)。因为乘数 B 的高 17 位是一致的，早期中止逻辑 50 发送信号到 MAC 控制逻辑 52 (块 422)。然后，MAC 控制逻辑 52 发送信号以连接触发器 6a, 6b, 6c 和 MUX8a, 8b, 8c (块 424)。这允许一个新的 MAC 指令在下一个周期被处理 (块 426)。

因此，一个乘累加期间可特定操作提供单周期输入输出。按照本发明的一个实现中，MAC 避免使用可能耗更多电能，其包括无可能运行更慢的流水线电路的冗余硬件。实际上，Wallace 树和 MAC 一部分的先行进位加法器可同时操作。这种同时操作的现实使更高吞吐量成为可能。

虽然本发明仅被描述为有限的实现上，本领域中的资深人士可认识到很多可做的改动和变化。我们试图使附加的权利要求涵盖所有落入本发明真正精神和范畴的改动和变化。

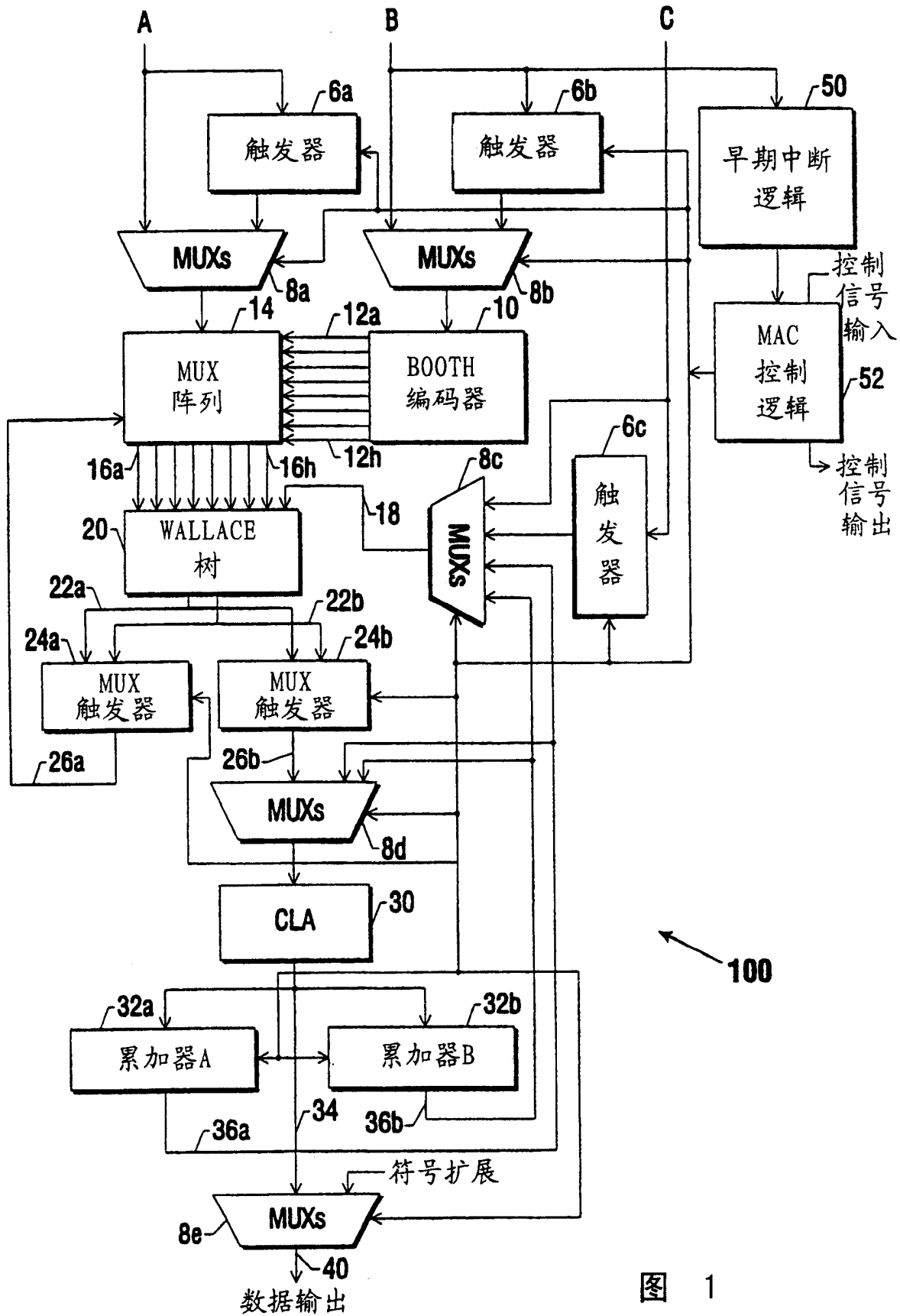


图 1

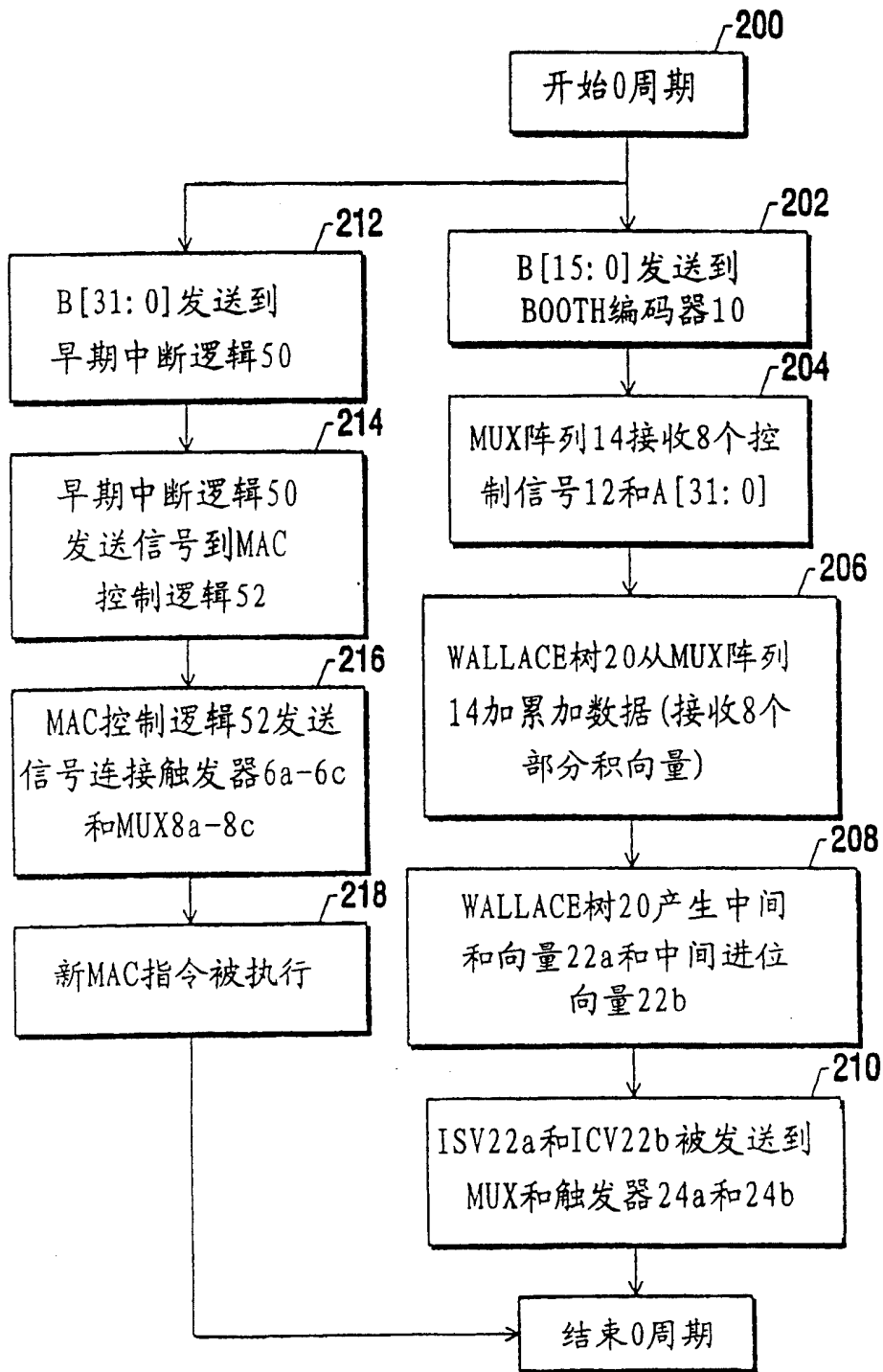


图 2A

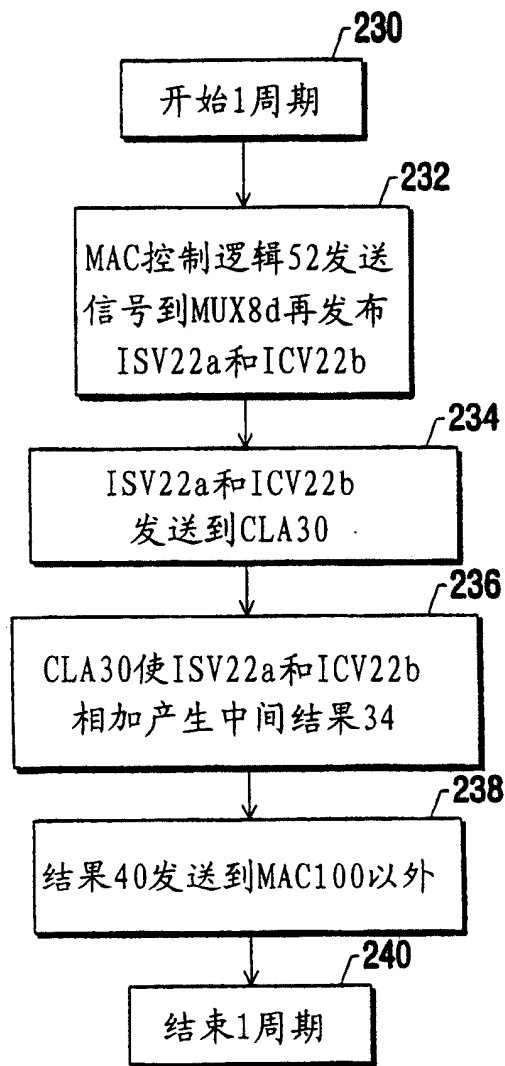
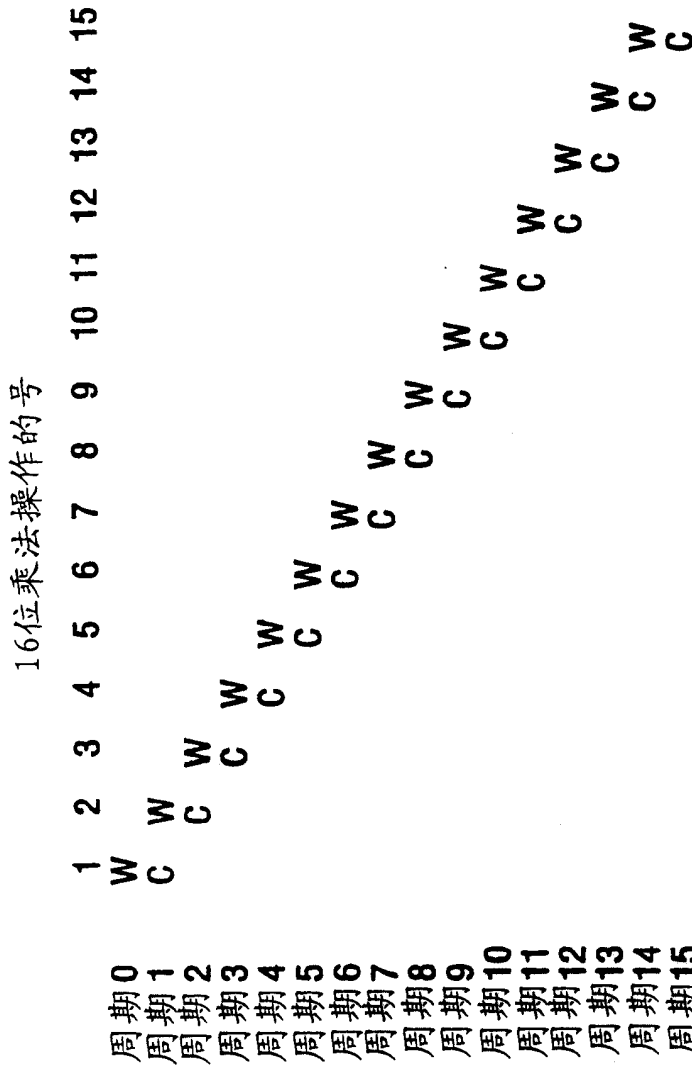


图 2B





W- WALLACE树  
 C- 进位预测加法器

图 3

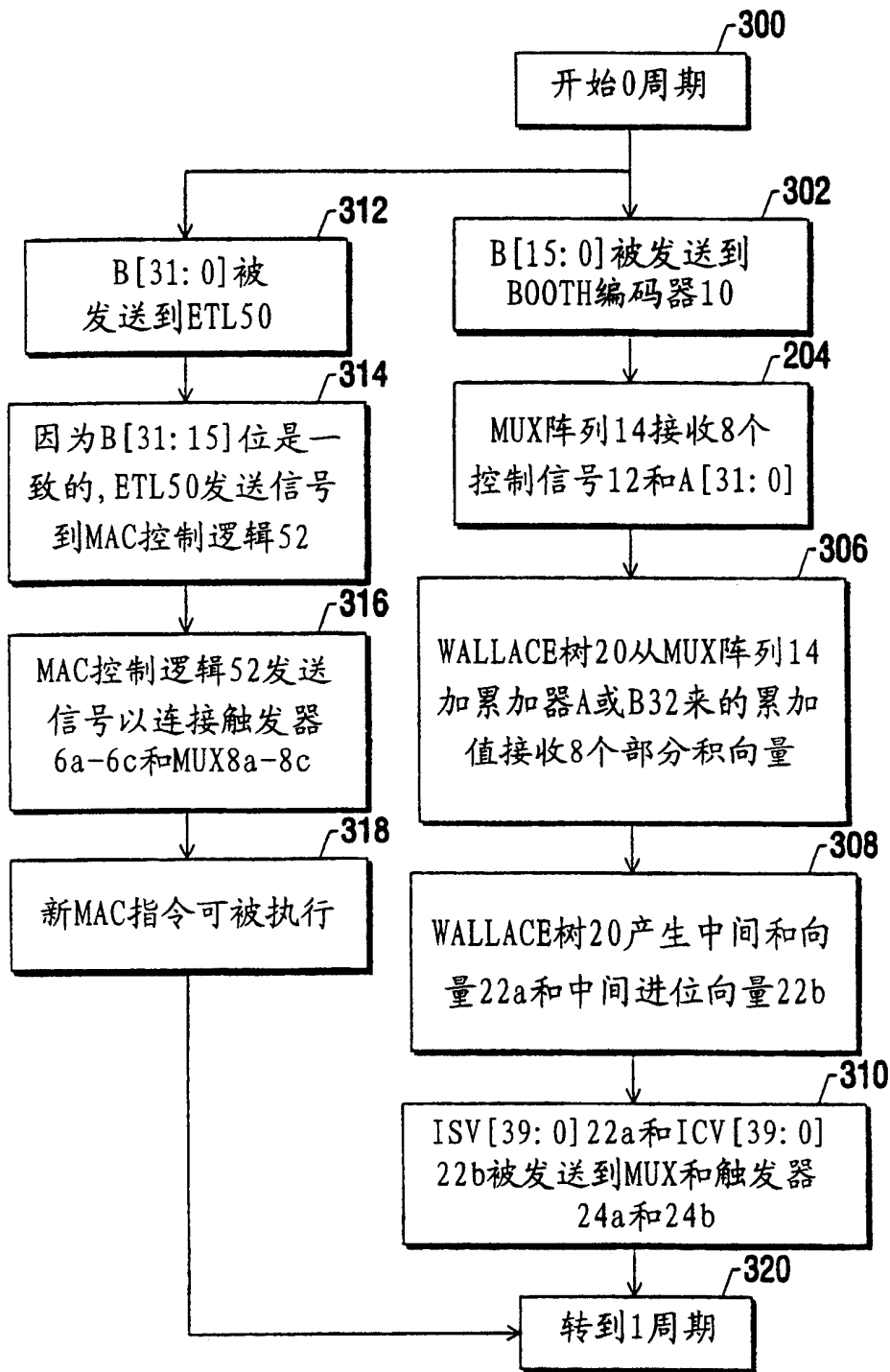


图 4A

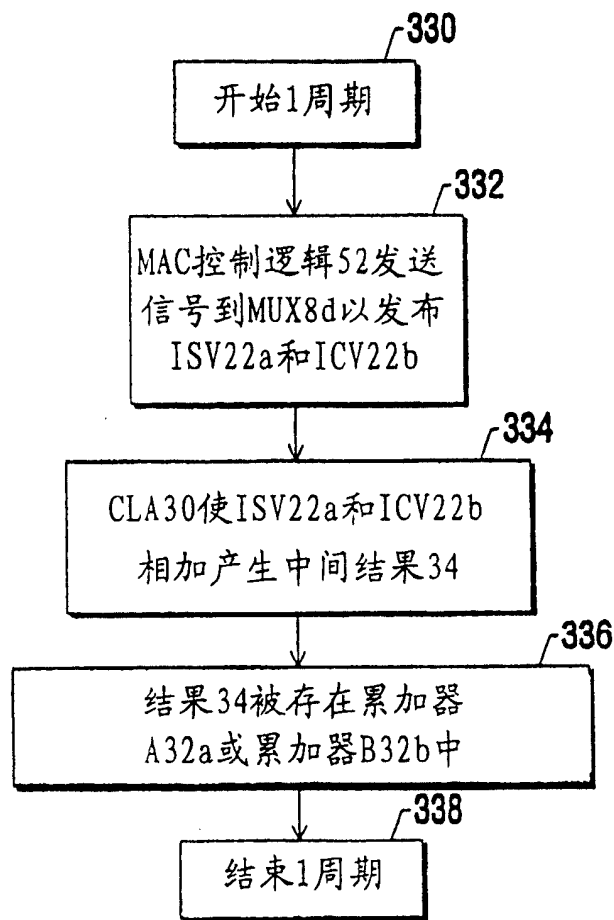


图 4B

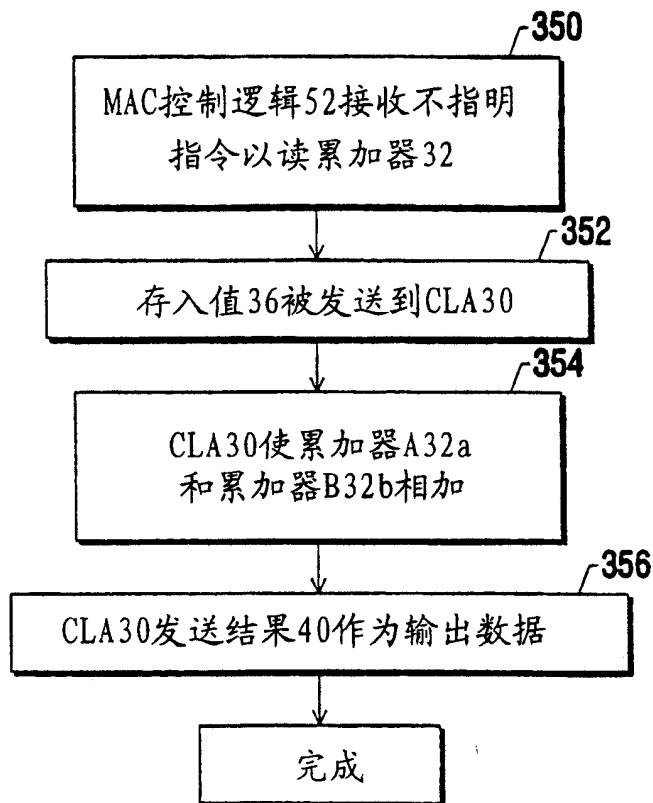


图 4C

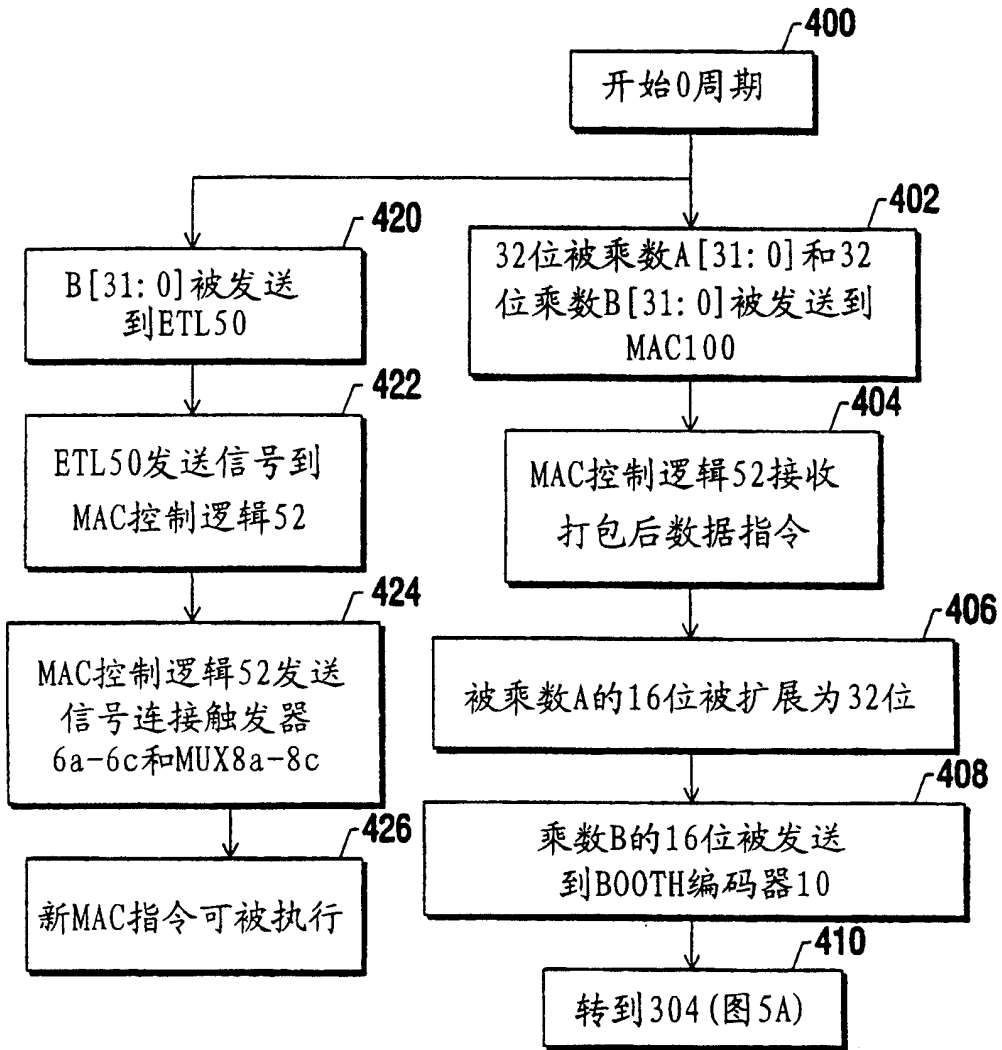


图 5