



(12) 发明专利申请

(10) 申请公布号 CN 103119548 A

(43) 申请公布日 2013. 05. 22

(21) 申请号 201180046023. 0

(51) Int. Cl.

(22) 申请日 2011. 08. 12

G06F 5/16 (2006. 01)

(30) 优先权数据

61/385, 919 2010. 09. 23 US

(85) PCT申请进入国家阶段日

2013. 03. 22

(86) PCT申请的申请数据

PCT/IB2011/002510 2011. 08. 12

(87) PCT申请的公布数据

W02012/038829 EN 2012. 03. 29

(71) 申请人 马维尔以色列 (M. I. S. L.) 有限公司

地址 以色列约克尼穆

(72) 发明人 E·沙姆斯凯 J·库什尼尔

(74) 专利代理机构 北京市金杜律师事务所

11256

代理人 鄢迅 李峥宇

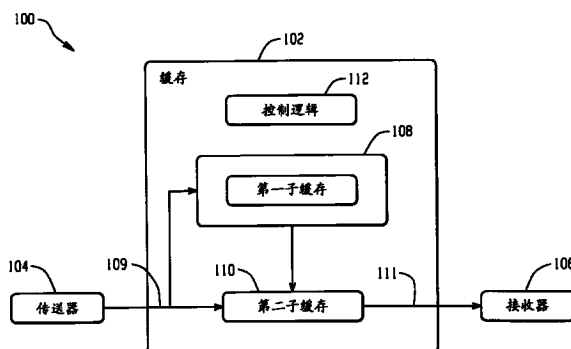
权利要求书3页 说明书7页 附图9页

(54) 发明名称

低等待时间先进先出 (FIFO) 缓存

(57) 摘要

提供了用于先进先出缓存的系统和方法。缓存包括配置用于存储从缓存输入接收的数据的第一子缓存, 以及第二子缓存。所述第二子缓存被配置用于存储从所述缓存输入或第一子缓存接收的数据, 并且以与在所述缓存输入处接收数据相同的顺序, 将数据输出到缓存输出。缓存控制逻辑被配置用于选择性地从所述缓存输入或第一子缓存将数据路由到所述第二子缓存, 使得在所述缓存输入处接收的数据可以按先进先出方式从所述第二子缓存输出。



1. 一种缓存,包括:

第一子缓存,被配置用于存储从缓存输入接收的数据;

第二子缓存,被配置用于存储从所述缓存输入或所述第一子缓存接收的数据,并且以与在所述缓存输入处接收数据相同的顺序将数据输出到缓存输出;

缓存控制逻辑,被配置用于选择性地从所述缓存输入或所述第一子缓存将数据路由到所述第二子缓存,使得在所述缓存输入处接收的数据可以按先进先出方式从所述第二子缓存输出。

2. 根据权利要求1所述的缓存,其中所述缓存控制逻辑响应于所述第一子缓存和所述第二子缓存中至少一个的空/满状态。

3. 根据权利要求1所述的缓存,其中所述缓存控制逻辑被配置用于当所述第一子缓存为空时,通过旁路所述第一子缓存而从所述缓存输入向所述第二子缓存路由数据。

4. 根据权利要求1所述的缓存,其中所述缓存控制逻辑还被配置用于当所述第二子缓存为满时,从所述缓存输入向所述第一子缓存路由数据。

5. 根据权利要求1所述的缓存,其中所述缓存控制逻辑还被配置用于当其他数据当前被存储在所述第一子缓存中时,从所述缓存输入向所述第一子缓存路由数据。

6. 根据权利要求1所述的缓存,其中所述控制逻辑还被配置用于当所述第二子缓存不满时,从所述第一子缓存向所述第二子缓存路由数据。

7. 根据权利要求1所述的缓存,其中所述第二子缓存的等待时间小于所述第一子缓存的等待时间。

8. 根据权利要求1所述的缓存,其中所述第一子缓存和所述第二子缓存是不同类型的独立先进先出(FIFO)队列。

9. 根据权利要求1所述的缓存,其中所述第二子缓存是具有所述缓存输出所响应于的采样器输出的采样器,所述采样器输出直接连接到所述缓存输出。

10. 根据权利要求1所述的缓存,其中所述第一子缓存包括输出比接收到数据请求之后一个周期多的数据的先进先出(FIFO)队列,并且其中所述第二子缓存被配置用于在接收到数据请求之后一个时钟周期内将数据输出到所述缓存输出。

11. 根据权利要求10所述的缓存,其中所述第一子缓存包括触发器存储器位置的阵列,其中所述第一子缓存跟踪哪个触发器存储器位置在FIFO队列前面,其中所述第一子缓存被配置用于访问在FIFO队列前面的所述触发器存储器位置,用于在接收到读命令时向所述第二子缓存传送数据。

12. 根据权利要求10所述的缓存,其中所述第一子缓存包括随机访问存储器,其中所述第一子缓存跟踪随机访问存储器的哪个存储器位置在FIFO队列前面,其中所述第一子缓存被配置用于获取访问在FIFO队列前面的所述随机访问存储器位置,用于在接收到读命令时向所述第二子缓存传送。

13. 根据权利要求1所述的缓存,其中所述控制逻辑使用计数器确定所述第一子缓存是否为空、以及所述第二子缓存是否为满,其中当数据在所述缓存输入处被接收时所述计数器递增,并且其中当数据从所述缓存输出处被输出时所述计数器递减。

14. 根据权利要求1所述的缓存,还包括:

随机访问存储器预缓存,被配置用于选择性地向所述第一子缓存和所述第二子缓存传

送数据 ; 以及

第二控制逻辑, 被配置用于 :

当所述第一子缓存和所述第二子缓存包含比所述预缓存的时钟周期等待时间少的数据值时, 将数据路由到所述第一子缓存或所述第二子缓存 ;

当所述第一子缓存和所述第二子缓存包含比所述预缓存的所述时钟周期等待时间多的数据值时, 将数据路由到所述预缓存 ; 以及

当所述第一子缓存不空时, 从所述预缓存向所述第一子缓存路由数据。

15. 根据权利要求 14 所述的缓存, 其中所述第二控制逻辑使用第二计数器确定所述第一子缓存和所述第二子缓存是否包含比所述预缓存的时钟周期等待时间少的数据值。

16. 根据权利要求 1 所述的缓存, 还包括 :

预采样器, 被配置用于向所述第一子缓存、所述第二子缓存和所述缓存输出传送数据 ; 以及

第二控制逻辑, 被配置用于 :

当所述第一子缓存和所述第二子缓存为空时, 从所述预采样器向所述缓存输出路由数据 ;

当所述第一子缓存和所述第二子缓存中至少一个不空时, 从所述预采样器向所述第一子缓存或所述第二子缓存路由数据。

17. 根据权利要求 16 所述的缓存, 其中所述控制逻辑和所述第二控制逻辑使用单个计数器确定所述第一子缓存和所述第二子缓存的状态。

18. 根据权利要求 1 所述的缓存, 其中所述缓存是集成电路上的组件, 并且其中所述集成电路包括 :

存储器电路 ; 以及

处理器电路 ;

其中所述缓存耦合到所述存储器电路和所述处理器电路中的至少一个。

19. 一种缓存数据的方法, 包括 :

在缓存输入处接收输入数据 ;

确定第一子缓存和第二子缓存的存储器满 / 空状态 ;

基于所述存储器满 / 空状态, 将数据选择性地从所述缓存输入或所述第一子缓存路由到所述第二子缓存, 使得在所述缓存区输入处接收的数据可以按先进先出方式从所述第二子缓存输出。

20. 根据权利要求 19 所述的方法, 其中选择性地路由包括 :

当所述第一子缓存为空时, 将所述输入数据路由到所述第二子缓存 ; 以及

当所述第二子缓存为满时或者当所述第一子缓存不空时, 将所述输入数据路由到所述第一子缓存。

21. 根据权利要求 19 所述的方法, 还包括 :

当所述第一子缓存和所述第二子缓存包含比所述预缓存的时钟周期等待时间的数目少的数据值时, 将所述输入数据路由到随机访问存储器预缓存 ; 以及

当所述第一子缓存和所述第二子缓存包含比所述预缓存的时钟周期等待时间的数目多的数据值时, 将所述输入数据路由到所述第一子缓存或所述第二子缓存 ; 以及

当所述第一子缓存或所述第二子缓存不满时,将数据从所述预缓存路由到所述第一子缓存或所述第二子缓存。

22. 根据权利要求 19 所述的方法,还包括:

将所述输入数据路由到预采样器;

当所述第一子缓存和所述第二子缓存为空时,将所述输入数据从预采样器路由到输出;以及

当所述第一子缓存和所述第二子缓存中的一个不空时,将所述输入数据从所述预采样器路由到所述第一子缓存或所述第二子缓存。

## 低等待时间先进先出 (FIFO) 缓存

[0001] 相关申请的交叉引用

[0002] 本申请要求于 2010 年 9 月 23 日提交的发明名称为“jFIFO”的美国临时申请 No. 61/385, 919 的优先权, 通过引用将其整体并入于此。

### 技术领域

[0003] 在此所述的技术概括地涉及数据传送, 且更具体地涉及低等待时间数据缓存。

### 背景技术

[0004] 缓存提供用于在从一个位置向下一个位置传送期间临时保持数据的存储区域。例如, 在计算机系统内的硬件和 / 或进程之间移动数据时使用缓存。当传送实体和接收实体之间传送和消耗速率不匹配时利用缓存。缓存提供临时存储, 这使得所述传送实体可以在不关心接收实体能接受被传送的数据的速率的情况下发送数据。接收实体因此能根据接收实体的访问和处理能力, 访问存储在缓存中的、来自传送实体的数据。

[0005] 以上说明呈现为这个领域中相关技术的一般概要, 并且不应理解为承认它包含的任何信息组成与本专利申请相对的现有技术。

### 发明内容

[0006] 根据在此的教程, 提供了用于先进先出缓存的系统和方法。缓存包括配置用于存储从缓存输入接收的数据的第一子缓存, 以及第二子缓存。所述第二子缓存被配置用于存储从所述缓存输入或第一子缓存接收的数据, 并且以与在所述缓存输入处接收数据相同的顺序, 将数据输出到缓存输出。缓存控制逻辑被配置用于选择性地从所述缓存输入或第一子缓存将数据路由到所述第二子缓存, 使得在所述缓存输入接收的数据可以按先进先出方式, 从所述第二子缓存输出。所述缓存控制逻辑响应于第一子缓存和第二子缓存中至少一个的空 / 满状态。

[0007] 作为另一示例, 在缓存数据的方法中, 输入数据在缓存输入处被接收。第一子缓存和第二子缓存的存储器满 / 空状态被确定。基于所述存储器满 / 空状态, 数据被选择性地从所述缓存输入或所述第一子缓存路由到所述第二子缓存, 使得在所述缓存输入处接收的数据可以按先进先出方式从所述第二子缓存输出。

[0008] 作为又一个示例, 一种集成电路包括处理器电路、存储器电路和缓存。所述缓存包括: 第一子缓存, 被配置用于从缓存输入接收的数据; 以及第二子缓存, 被配置用于存储从所述缓存输入或第一子缓存接收的数据, 并且以与在所述缓存输入处接收数据相同的顺序将数据输出到缓存输出。所述缓存还包括缓存控制逻辑, 被配置用于选择性地从所述缓存输入或第一子缓存将数据路由到所述第二子缓存, 使得在所述缓存输入处接收的数据可以按先进先出方式从所述第二子缓存输出。

### 附图说明

- [0009] 图 1 为绘出低等待时间 FIFO 缓存的实现的框图。
- [0010] 图 2 为绘出具有触发器第一子缓存和采样器第二子缓存的缓存实现实施方式的框图。
- [0011] 图 3 为绘出一种缓存数据的方法的流程图。
- [0012] 图 4 为绘出基于随机访问存储器 (RAM) 和低等待时间触发器缓存的实施方式的流程图。
- [0013] 图 5 为绘出使用预缓存来缓存数据的方法的流程图。
- [0014] 图 6 为绘出具有相关采样器预缓存的基于 RAM 或触发器的缓存实施方式的框图。
- [0015] 图 7 为绘出一种缓存数据的方法的流程图。
- [0016] 图 8 为绘出缓存集成电路实施方式的框图。
- [0017] 图 9 为绘出另一集成电路实施方式的框图。

### 具体实施方式

[0018] 先进先出 (FIFO) 是在传送实体和接收实体之间缓存数据的一种方法。缓存在如下配置中是有用的,例如,在传送实体以大于接收实体的消耗能力的突发来传送数据,以及接收实体正处于不可用的时段。FIFO 缓存在多种环境下是有用的。例如,FIFO 缓存广泛应用于专用集成电路 (ASIC) 设计中。

[0019] 传统的 FIFO 缓存实现面临若干缺陷。例如,从数据被传送到 FIFO 缓存的输入的时间到该数据可用于在 FIFO 缓存的输出处访问的时间,很多 FIFO 缓存实现面临过多的等待时间。在由传送实体发送数据与由接收实体接收该数据之间,很多 FIFO 缓存具有至少两个时钟周期的等待时间。

[0020] 除了等待时段过长,出现另一个问题是等待时段在 FIFO 缓存实现中不一致。例如,与基于 RAM 的 FIFO 缓存相比,在一些配置中的基于触发器的 FIFO 缓存具有不同的等待时间。导致缓存等待时间改变的电路中 FIFO 缓存实现中的改变(例如增加错误校正代码 (ECC) 保护)需要在传送和 / 或接收实体处修改以应对等待时间改变。

[0021] 传统 FIFO 缓存还存在定时隔离问题。例如,在 ASIC 环境中,FIFO 缓存即插即用通常在逻辑和物理方面都是重要的。这意味着 FIFO 缓存不应对外围元件施加定时限制。例如,当 FIFO 缓存使得在特定时钟周期期间在输出处可获得数据时,优选使得在尽可能接近特定时钟周期的开始处获得该数据,以便不延迟接收外设对该数据的访问。

[0022] 传统的 FIFO 缓存还缺乏在缓存和外设单元之间的标准的接口(例如,发送实体和接收实体)。跨多个实现标准化 FIFO 缓存接口的尝试通常导致跨各个 FIFO 缓存实现的更高的等待时间和 / 或定时隔离问题。传统的缓存也可以实现高成本的组件以尝试改进速度性能。

[0023] 图 1 为绘出根据本公开的一个实施方式实现低等待时间 FIFO 缓存的框图。缓存 102 从传送器 104 接收数据,并且按先进先出方式将接收的数据传送至接收器 106。缓存 102 包括第一子缓存 108,其被配置用于存储从缓存输入 109 接收的数据。缓存 102 还包括第二子缓存 110,其被配置用于存储从缓存输入 109 或第一子缓存 108 接收数据,并且按与在缓存输入 109 处接收数据相同的顺序将数据输出到缓存输出 111。缓存 102 还包括控制逻辑 112,其选择性地从缓存输入 109 或第一子缓存 108 将数据路由到第二子缓存 110,使

得在缓存输入 109 接收的数据可以按先进先出方式从第二子缓存 110 输出。

[0024] FIFO 缓存 102 使用多种机制提供改进的性能。例如, 在一些实现中, 第一子缓存 108 具有比第二子缓存 110 更大量的存储空间。示例第一子缓存 108 还工作在比第二子缓存 110 更高的等待时间(例如, 第一子缓存以 2 个时钟周期等待时间提供数据, 而第二子缓存以 1 个时钟周期等待时间提供数据)。较高速度的第二子缓存 110 提供用于当缓存 102 为空或接近空时使用的旁路机制, 以使从传送器 104 接收的数据立即可用于接收器 106(例如, 以 1 个时钟周期等待时间)。在一个实施方式中, 较大较慢的第一子缓存 108 提供较廉价的存储作为备选, 以用于当缓存 102 为非空或接近空时使用。当数据从第二子缓存 110 传送到接收器 106 时, 第一子缓存 108 再填充第二子缓存 110。

[0025] 在这个配置中, 避免了通过单独使用较慢的第一子缓存 108 将产生的瓶颈。当缓存 102 接近空时, 较慢的第一子缓存 108 被旁路, 使得在缓存输入 109 处接收的数据被直接传送到较快的第二子缓存 110 用于在缓存输出 111 处快速访问。当缓存 102 处于更满的状态时, 在缓存输入 109 处接收的数据被提供至较慢的第一子缓存 108 用于存储。存储在较慢的第一子缓存 108 处的数据被按需要传送至第二子缓存 110 以再填充第二子缓存, 以继续使得在缓存输出 111 处可获得数据。

[0026] 控制逻辑 112 监视第一子缓存 108 和第二子缓存 110 的状态, 并且选择性地从缓存输入路由数据以提供快速的数据吞吐量。例如, 当缓存 102 为空时(即, 第一子缓存 108 和第二子缓存 110 都不包含数据), 控制逻辑通过旁路第一子缓存而从缓存输入 109 向较快的第二子缓存 110 路由数据, 使得在缓存输出 111 立即可获得接收的数据。当第二子缓存 110 为满时, 控制逻辑 112 从缓存输入向第一子缓存 108 路由数据。这个控制方案使得当缓存 102 为空或接近空时, 通过使用第二子缓存 110 来支持快速访问数据, 而当缓存 102 处于更满的状态时, 利用较低成本、较慢、较大存储的第一子缓存 108。

[0027] 控制逻辑 112 还路由数据以便保持缓存 102 的先进先出特性。例如, 缓存控制逻辑被配置用于当其他数据当前存储在第一子缓存 108 中时(即, 当第一子缓存 108 不空时), 从缓存输入 109 向第一子缓存 108 路由数据。这个逻辑保证在缓存输入处接收的数据不会跳到已存储在第一子缓存 108 中的数据之前。控制逻辑还被配置用于当第二子缓存 110 不满时, 从第一子缓存 108 向第二子缓存 110 路由数据。这种操作提供再填充效果, 其中第二子缓存 110 利用当前存储在第一子缓存 108 中的行中的下一数据再次被装满。

[0028] 图 2 为绘出具有触发器第一子缓存和采样器第二子缓存的缓存实现(jFIFO)的框图。基于触发器的缓存 202 通过 load\_in 缓存输入 204 来接收数据。接收的数据被选择性地路由到触发器第一子缓存 206 和采样器第二子缓存 208 中的一个。在一个实施方式中, 触发器第一子缓存 206 包括触发器阵列和内部逻辑用于识别序列中下一触发器, 以及多路复用器用于在接收到读请求时从该下一触发器访问数据。采样器第二子缓存 208 对在一个或多个输入接收的数据进行采样, 并且将检测的数据以低等待时间(例如, 在来自 ready\_in 输入 212 的读请求的一个时钟周期内)传输至 load\_out 输出 210。采样器 208 为缓存 202 提供希望的定时隔离特征, 这是因为普通的基于采样器/触发器的缓存输出 210 中的实施方式中不存在逻辑电路(即, 相对于常规的基于触发器的缓存而言, 该常规的基于触发器的缓存在触发器存储元件和基于触发器的缓存输出之间包括多路复用器-触发器选择逻辑)。

[0029] 基于触发器的缓存 202 还包括缓存控制逻辑 214、216、218、220、222、224，其选择性地从缓存输入 204 或第一子缓存 206 向第二子缓存 208 路由数据，以支持先进先出数据传输。在图 2 的示例中，控制逻辑 214 基于缓存输入 204 设置递增字段 (inc) 和就绪信号 226。由触发器 228 提供的就绪信号 226 从基于触发器的缓存 202 输出，以向传送实体表明缓存 202 准备就绪以接收数据。inc 数据信号被控制逻辑 216 接收，该逻辑 216 确定 inc 数据信号是否将经由 232 被路由到第一子缓存 206，或经由 234 被路由到第二子缓存 208。使用填充计数器 (fill counter) 230 处维护的计数来确定 inc 数据信号的路由。当计数大于 1 时，其中基于一个存储位置的子缓存 208 容量选择 1，inc 数据信号被路由到第一子缓存 206 用于存储。当计数小于或等于 1 时，控制逻辑通过旁路第一子缓存，将 inc 数据信号直接路由到第二子缓存 208。在接收到 inc 数据信号时，控制逻辑 218 更新存储在计数器 230 中的计数值。当接收到 inc 数据信号时 (即，情况 1,0)，计数器增加 1。

[0030] 在接收到就绪信号 212 时，第二子缓存 208 经由缓存输出 210 传送数据。当缓存输出 210 和缓存读请求 212 线为激活时，通过使用递减 (dec) 数据信号，控制逻辑 220 命令缓存 202 中数据的计数递减。dec 数据信号被传送至控制逻辑 218 用于调节计数。在接收到 dec 数据信号时，控制逻辑 218 更新存储在计数器 230 中的计数值。当 dec 数据信号被控制逻辑 218 接收时 (即，情况 0,1)，计数器减 1。dec 数据信号还被控制逻辑 222 接收并与来自计数器 230 的计数一起，确定第二子缓存 208 是否应该用来自第一子缓存 206 的数据再填充。当计数器大于 1 时，表示数据存在于第一子缓存中，再填充信号通过 236 被传送到第一子缓存 206 以及第二子缓存 208，命令第一子缓存 206 中的行中下一数据经由 238 被传送到第二子缓存 208。

[0031] 控制逻辑 218 提供的计数器更新也被传送至控制逻辑 224 用于产生缓存就绪信号 226。当计数器更新表示值小于总缓存大小时 (即，第一子缓存 206 和第二子缓存 208 的存储大小的和)，就绪信号经由触发器 228 被传送至就绪输出 226。当计数器更新是不小于总缓存大小时，则就绪输出 226 表明不能接受新数据的满缓存 202。

[0032] 图 3 为绘出一种缓存数据的方法的流程图。在 302，输入数据在缓存输入处被接收。在 304，针对第一子缓存和第二子缓存确定存储器满 / 空状态，比如通过使用计数值进行。基于存储器满 / 空状态，数据被选择性地从缓存输入或第一子缓存路由到第二子缓存，使得在缓存输入接收的数据可以按先进先出方式从第二子缓存输出。例如，在 306，当第一子缓存为空时，通过旁路第一子缓存，输入数据被直接传送至第二子缓存。在 308，当第一子缓存不空时或当第二子缓存为满时，输入数据被传送至第一子缓存。在 310，在接收到数据时，计数值被更新。在 312，接收到数据请求。在 314，在接收到请求时，数据从第二子缓存被传送 (例如，在接收到读请求的一个时钟周期内)。在 316，第二子缓存利用来自第一子缓存的数据被再填充，并且在 318，计数器被更新以反映输出的数据。

[0033] 图 4 为绘出基于随机访问存储器 (RAM) 和低等待时间触发器的缓存的框图。RAM 和基于触发器的缓存 402 (另一 jFIFO 实现) 经由 load\_in 缓存输入 404 接收数据。接收到的数据被选择性地路由到基于 RAM 的第一子缓存 406 (预缓存) 和基于低等待时间触发器的第二子缓存 408 中的一个。基于 RAM 的第一子缓存 406 包括：存储器位置的阵列、和用于识别序列中下一存储器位置的内部逻辑、以及用于在接收到读请求时从下一存储器位置访问数据的多路复用器。基于低等待时间触发器的第二子缓存 408 经由一个或多个输入接



收数据,并且将接收的数据以低等待时间传送至 load\_out 输出 410(例如,在来自 ready\_in 输入 412 的读请求的一个时钟周期内)。基于低等待时间触发器的第二子缓存采用多种形式。例如,基于 jFIFO 触发器的缓存(比如以上结合图 2 所述)用作图 4 的示例中的第二子缓存 408。

[0034] 缓存 402 还包括缓存控制逻辑 414、416、418、420、422、424,其选择性地将数据从缓存输入 404 或第一子缓存 406 路由到第二子缓存 408,以支持先进先出数据传输和其他操作。在图 4 的示例中,控制逻辑 414 基于缓存输入 404 设置递增字段 (inc) 和就绪信号 426。由触发器 428 传送的就绪信号被从缓存 402 输出,以向传送实体表明缓存 402 准备就绪以接收数据。inc 数据信号被控制逻辑 416 接收,其确定 inc 数据信号是否将经由 432 被路由到第一子缓存 406,或经由 434 被路由到第二子缓存 408。使用在填充计数器 430 处维护的计数,来确定 inc 数据信号的路由。在图 4 的示例中,控制逻辑 416 基于计数值和基于 RAM 的第一子缓存 406 的等待时间来确定 inc 数据信号的路由。当计数小于第一子缓存等待时间加 1 时,则 inc 数据信号被直接路由到第二子缓存 208。当计数大于第一子缓存等待时间加 1 时,则 inc 数据信号被路由到第一子缓存 406。当接收到 inc 数据信号时,控制逻辑 418 更新存储在计数器 430 中的计数值。当接收到 inc 数据信号时(即,情况 1,0),计数器增加 1。

[0035] 在接收到就绪信号 412 时,第二子缓存 408 经由缓存输出 410 传送数据。当缓存输出 410 和缓存读请求 412 线为激活时,通过使用递减 (dec) 数据信号,控制逻辑 420 命令缓存 402 中的数据的计数递减。dec 数据信号被传送至控制逻辑 418 用于调节计数。在接收到 dec 数据信号时,控制逻辑 418 更新存储在计数器 430 中的计数值。当控制逻辑 418 接收到 dec 数据信号时(即,情况 0,1),计数器减少 1。dec 数据信号也被控制逻辑 422 接收并与来自计数器 430 的计数一起,以确定是否应该用来自第一子缓存 406 的数据再填充第二子缓存 408。当计数大于第一子缓存等待时间加 1 时,再填充信号经由 436 被传送至第一子缓存 406 以及第二子缓存 408,命令第一子缓存 406 中的行中下一数据经由 438 被传送到第二子缓存 408。第一子缓存 406 传送数据有效性信号 440,其表明在 438 处的数据对于由第二子缓存 408 读是有效的。

[0036] 由控制逻辑 418 提供的计数器更新也被传送至控制逻辑 424,用于产生缓存就绪信号 426。当计数器更新表示值小于总缓存大小时(即,第一子缓存 406 和第二子缓存 408 的存储大小的和),就绪信号经由触发器 428 被传送至就绪输出 426。当计数器更新表示值不小于总缓存大小时,则就绪输出 426 表明不能接受新数据的满缓存 402。

[0037] 图 5 为绘出使用预缓存以增强图 3 的方法的缓存数据的方法的流程图。在 502 接收输入数据。在 504,确定第一子缓存和第二子缓存的存储器满/空状态,如图 3 所述。在 506,当第一子缓存和第二子缓存包含比预缓存的时钟周期等待时间数目多的数据值时(例如,第一子缓存和第二子缓存包含 4 个数据值,并且基于 RAM 的预缓存具有 2 个时钟周期等待时间),输入数据被传送到随机访问存储器预缓存。在 508,当第一子缓存和第二子缓存包含比预缓存的时钟周期等待时间数目多的数据值时(例如,根据图 4 在 416 所示的控制逻辑),输入数据被传送到第一子缓存和第二子缓存中的一个。在 510,当第一子缓存和第二子缓存不满时,数据从预缓存被路由到第一子缓存或第二子缓存。

[0038] 图 6 为绘出具有相关采样器预缓存的基于 RAM 或触发器的缓存(jFIFO)的框图。

jFIFO 601 可以采用多种形式,比如以上结合图 2 和 4 所述的缓存之一的形式。缓存 602 经由 load\_in 缓存输入 604 接收数据。接收到的数据被路由到采样器预缓存 606,其对来自 604 的输入数据进行采样,并且在 608 将该输入数据的采样输出。在 608 采样的数据被传送至控制逻辑 610 和控制逻辑 612。当 jFIFO 601 为空并且经由 ready\_in 线 618 请求数据时,控制逻辑 610 向缓存输出 614 提供来自 608 的所采样的数据。jFIFO 601 的状态经由 620 传送至控制逻辑 610。当 jFIFO 601 不空时,来自 jFIFO 601 的数据通过控制逻辑 610 经由 622 被提供至缓存输出 614。

[0039] 控制逻辑 612 确定在 608 的采样数据是否应该被输入到 jFIFO601。当 jFIFO 601 不满并且还当缓存 602 没准备就绪用于传送时,控制逻辑 612 将采样的数据 608 添加到 jFIFO 601,如就绪输入 618 所表明的。控制逻辑 624 确定 set\_ready\_in 线 626 是否应该被激活,其命令预缓存采样并输出数据。当 jFIFO 601 为空且缓存就绪输入 618 是激活时(即,用于直接向缓存输出 614 传送),或者当 jFIFO601 不满且可用于从预采样输出 608 填充/再填充时,set\_ready\_in 线 626 和预采样 606 被激活。

[0040] 在图 6 的示例中,控制逻辑 628 基于缓存输入 604 和就绪信号 630 设置递增字段(inc)。由触发器 632 传送的就绪信号从缓存 602 输出,以向传送实体表明缓存 602 准备就绪用于接收数据。inc 数据信号被控制逻辑 634 接收,其更新存储在计数器 636 中的计数值。当接收到 inc 数据信号时(即,情况 1,0),计数器增加 1。

[0041] 当接收到就绪信号 618 时(读请求),当缓存输出 614 和缓存读请求 618 线是激活时,通过使用递减(dec)数据信号,控制逻辑 638 命令数据的计数递减。dec 数据信号被传送至控制逻辑 634 用于调节计数。在接收到 dec 数据信号时,控制逻辑 634 更新存储在计数器 636 中的计数值。当控制逻辑 634 接收到 dec 数据信号时(即,情况 0,1),计数器减 1。

[0042] 由控制逻辑 634 提供的计数器更新还被传送到控制逻辑 640,用于产生缓存就绪信号 630。当计数器更新表示值小于 jFIFO 601 缓存大小时,就绪信号经由触发器 632 传送到就绪输出 630。当计数器更新不小于总缓存大小时,则就绪输出 630 表明不能接受新数据的满缓存 602。在一个实施方式中,缓存依靠 jFIFO 601 以识别其满/空存储器状态(例如,使用 jFIFO 601 的内部计数器)。在这个实施方式中,缓存 602 不包括图 6 所示的某些元件,比如计数器 636。

[0043] 图 7 为绘出一种缓存数据的方法的流程图。在 702,输入数据被提供至预缓存。在 704,第一子缓存或第二子缓存的满/空存储器状态(比如结合图 2 和 4 所述的 jFIFO 电路中包含的那些)被确定,比如使用图 3 所述的过程。在 706,当第一子缓存或第二子缓存被确定为空时,输入数据被提供至缓存的输出。当在 704 确定非空状态时,在 708,输入数据被提供至第一子缓存或第二子缓存中的一个。

[0044] 图 8 为绘出集成电路实施方式的框图。集成电路 802 包括缓存 804,其中缓存包括第一子缓存 806,其被配置用于存储从缓存输入 808 接收的数据。缓存 804 还包括第二子缓存 810,其被配置用于存储从缓存输入 808 或第一子缓存 806 接收的数据。第二子缓存 810 还被配置用于按与在缓存输入 808 处接收数据相同的顺序,将数据输出到缓存输出 812。缓存 804 还包括缓存控制逻辑 814,其被配置用于选择性地将数据从缓存输入 808 或第一子缓存 806 路由到第二子缓存 810,使得在缓存输入 808 接收的数据可用于以先进先出方式从第

二子缓存 810 输出。缓存 804 可以在多种组件之间促进通信。例如,缓存 804 可以缓存从处理器、存储器或其他电路 816 到接收器 818 的通信,其可以是集成电路 802 的组件或集成电路 802 外部的组件。

[0045] 图 9 为绘出另一集成电路实施方式的框图。集成电路 902 包括缓存 904,其中缓存包括第一子缓存 906,其被配置用于存储从缓存输入 908 接收的数据。缓存 904 还包括第二子缓存 910,其被配置用于存储从缓存输入 908 或第一子缓存 906 接收的数据。第二子缓存 910 还被配置用于按与在缓存输入 908 处接收数据相同的顺序,将数据输出到缓存输出 912。缓存 904 还包括缓存控制逻辑 914,其被配置用于选择性地从缓存输入 908 或第一子缓存 906 路由到第二子缓存 910,使得在缓存输入 908 接收的数据可用于以先进先出方式,从第二子缓存 910 输出。缓存 904 可以促进在多种组件之间的通信。例如,缓存 904 可以缓存从传送器 916 到处理器、存储器或其他电路 918 的通信,其中传送器可以是集成电路 902 的组件或集成电路 902 外部的组件。

[0046] 已参考特定示范实施方式说明了本发明。但是,本领域技术人员将容易理解,可以以上述示范实施方式之外的具体形式实现本发明。所述实施方式仅是示意性的且不应理解为限制性的。在权利要求中而不是上述说明书中反映本发明的范围,并且旨在包括落在权利要求范围中的所有变型和等效设置。

[0047] 例如,所述系统和方法可以利用经由网络(例如,局域网、广域网、因特网,其组合等)传递的数据信号、光纤介质、调制的载波、无线网络等,用于与一个或多个数据处理设备通信。数据信号可以承载从/向设备提供的在此公开的数据中的任一或全部。

[0048] 应该理解,如在此的说明书所使用以及在下列权利要求中,除非上下文明确表明以外,“一个”、“一种”和“所述”包括复数引用。而且,如在此的说明书中所用并且在下列权利要求中,除非上下文明确表明以外,“之中”的意思包括“之中”和“之上”。此外,除非上下文明确表明以外,使用词语“每个”不必表示“每一个和每个”。最后,如在此的说明中所用并且在下列权利要求中,除非上下文明确表明以外,“和”和“或”的含义包括连接和转折且可互换地使用;短语“异或”可用于表明其中仅可应用转折意思的情况。

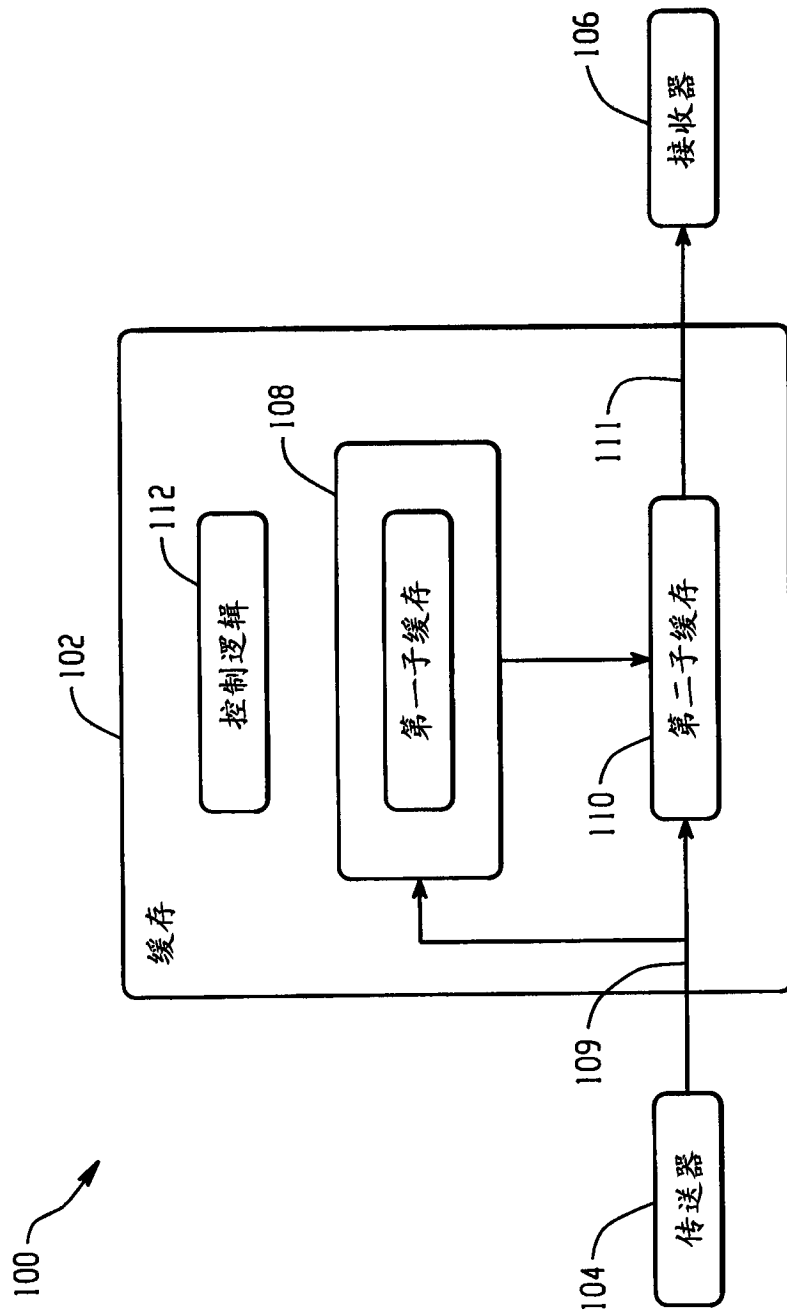


图 1

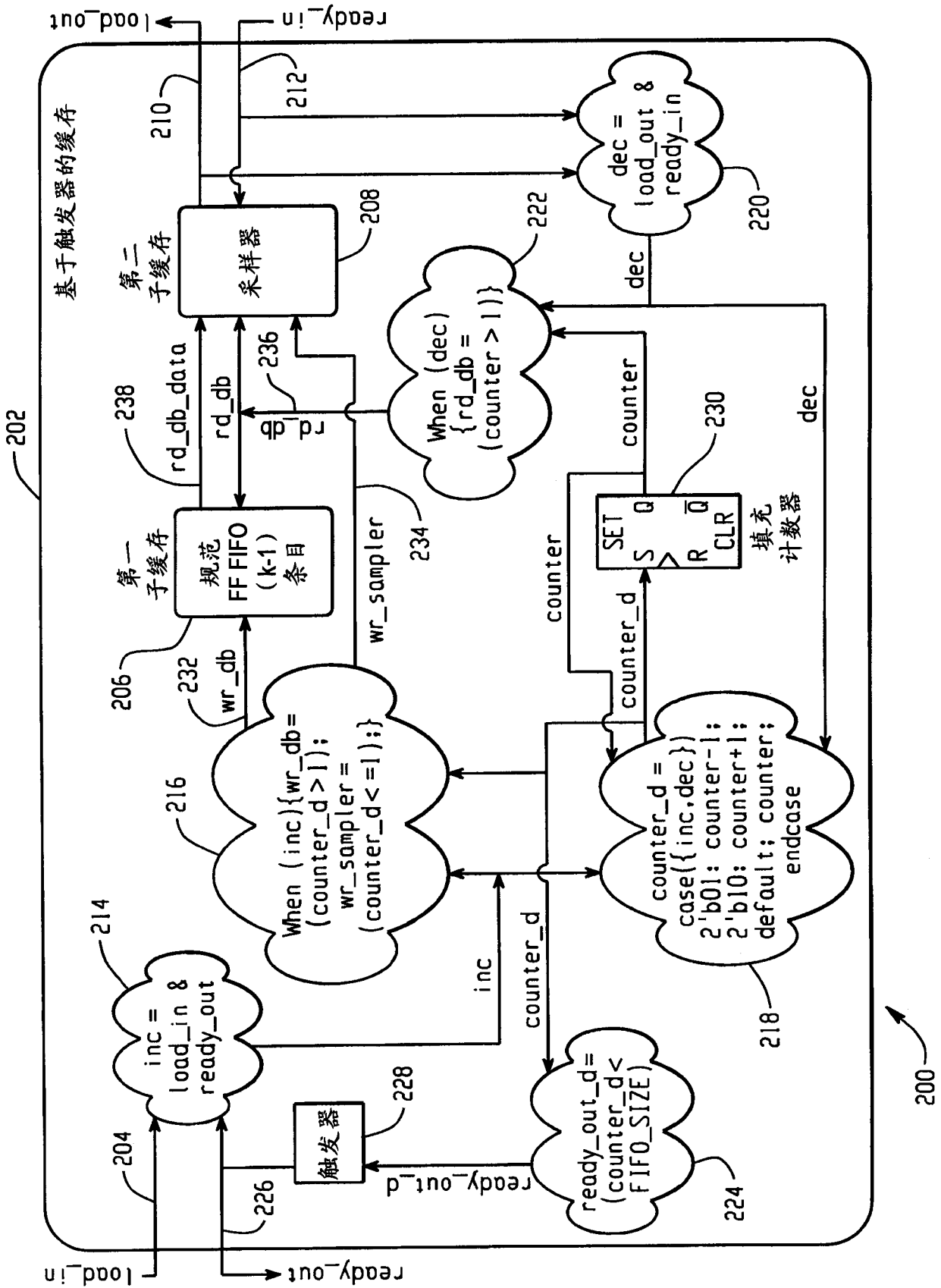


图 2

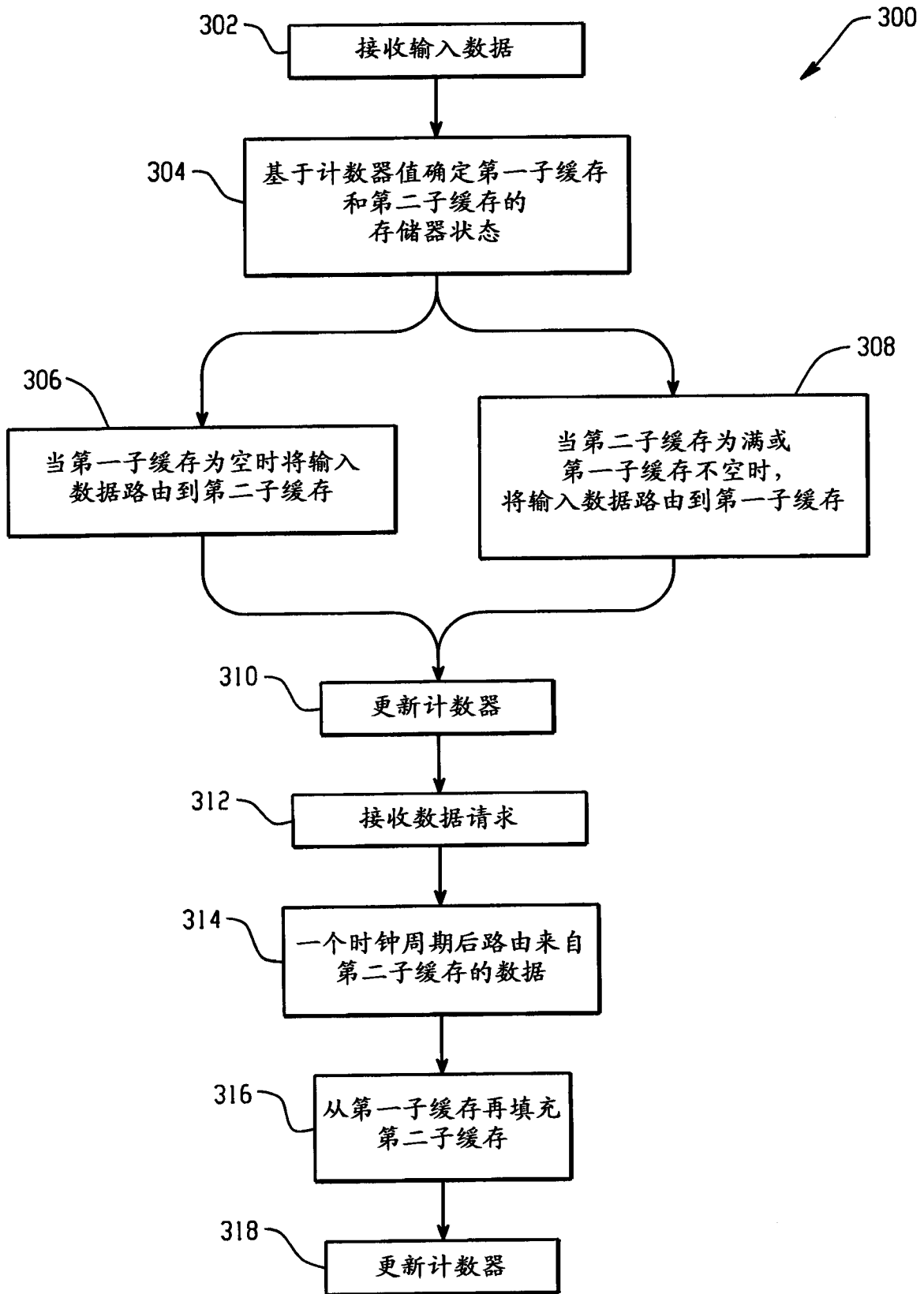


图 3

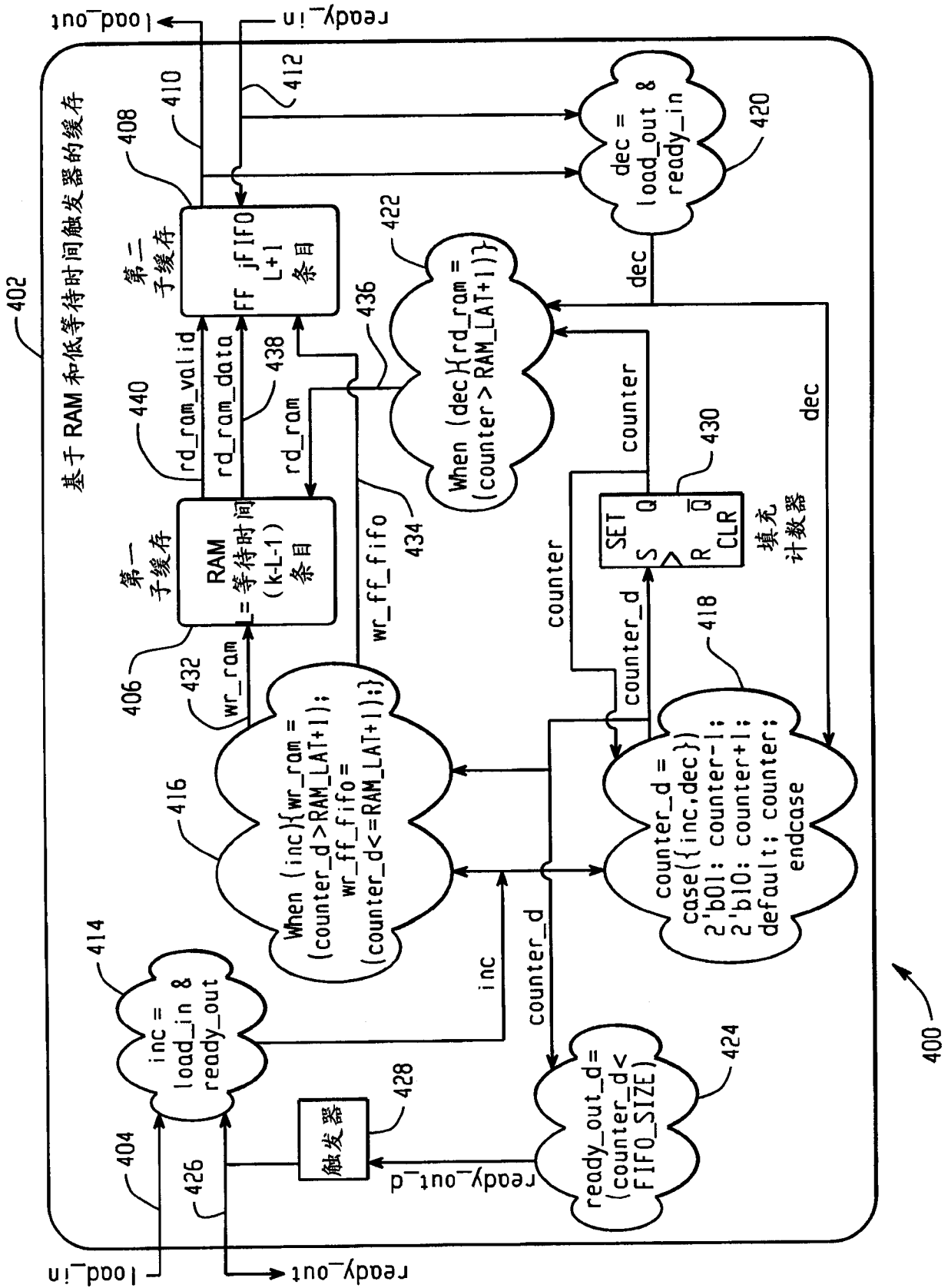


图 4

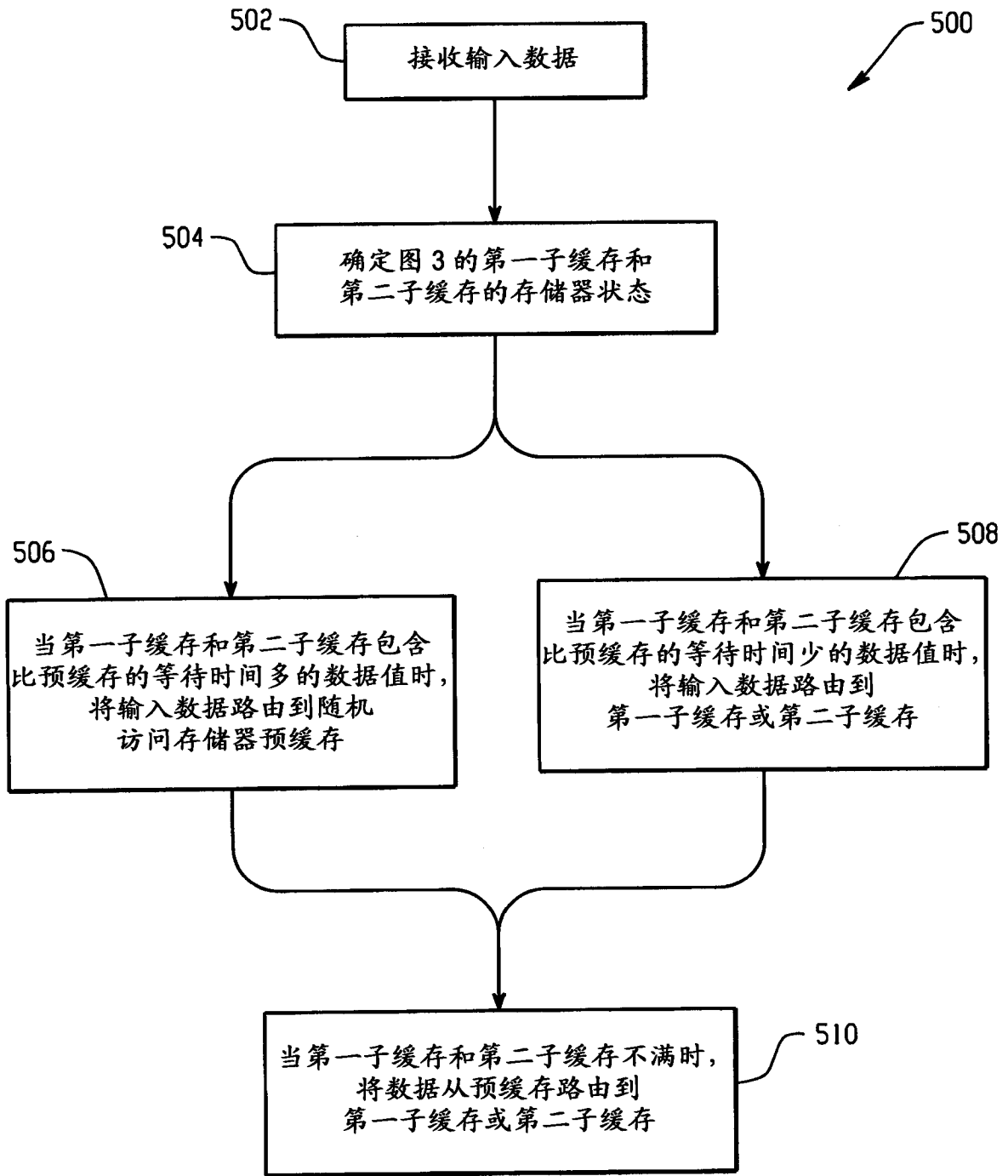


图 5



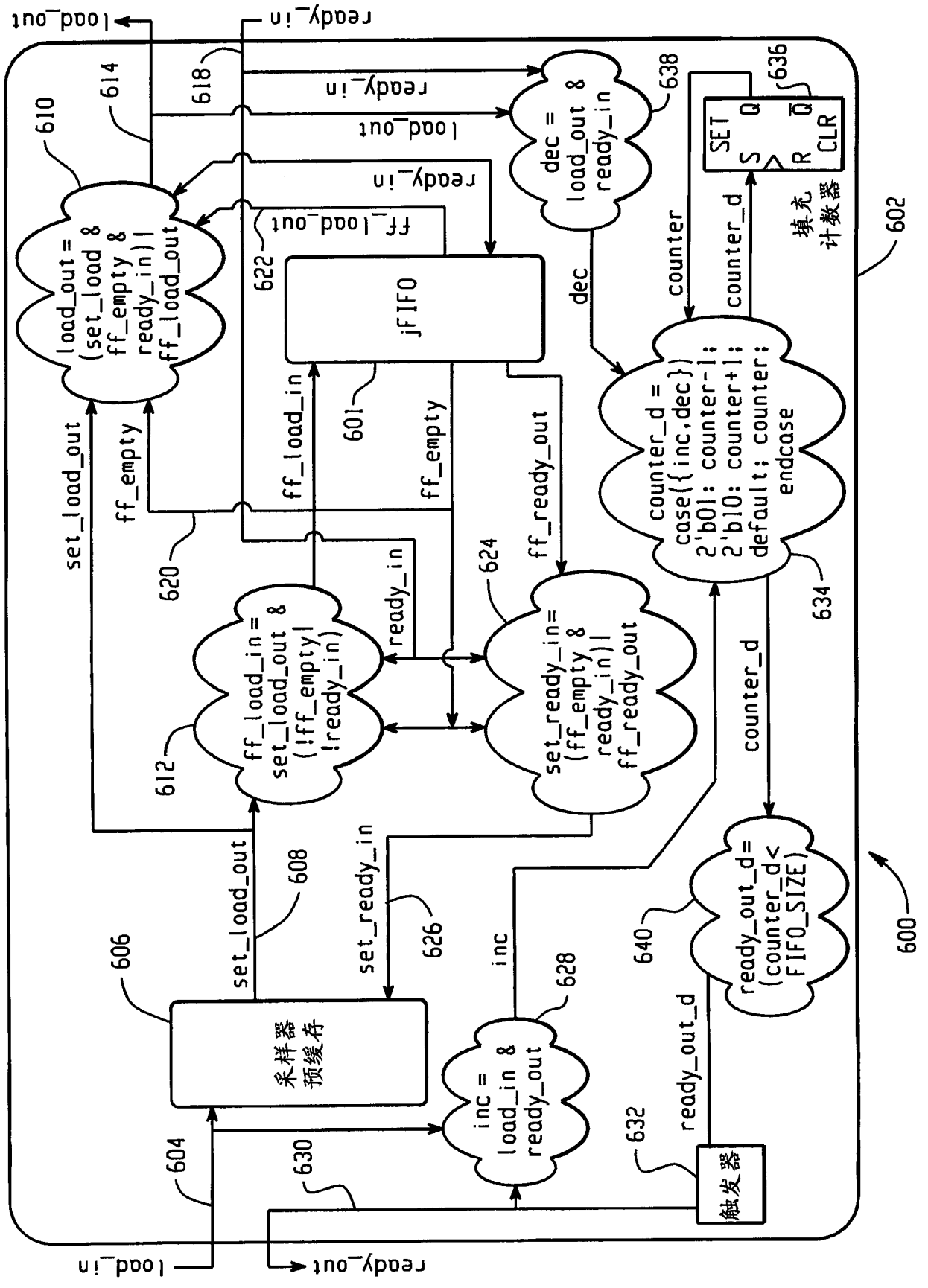


图 6

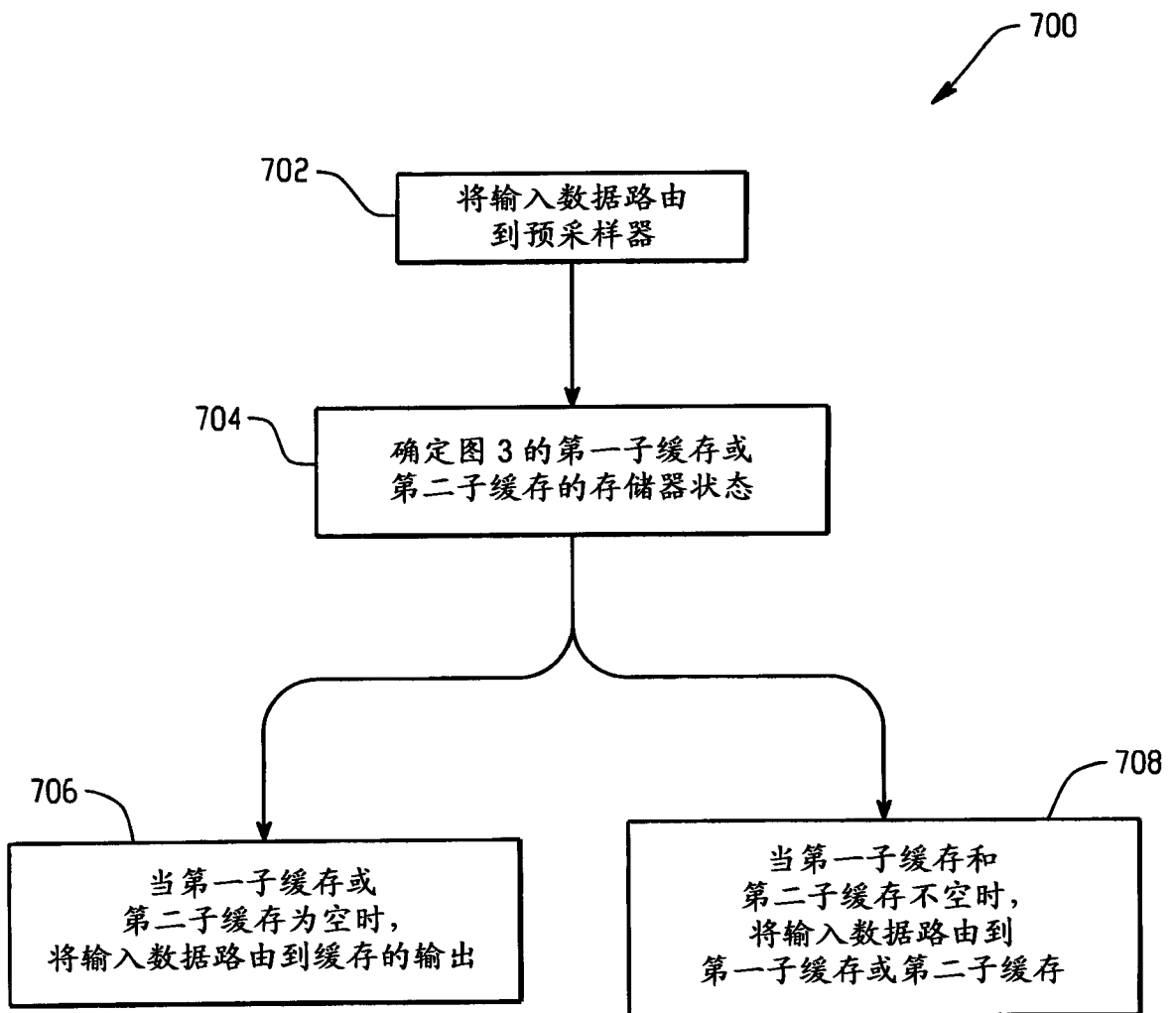


图 7

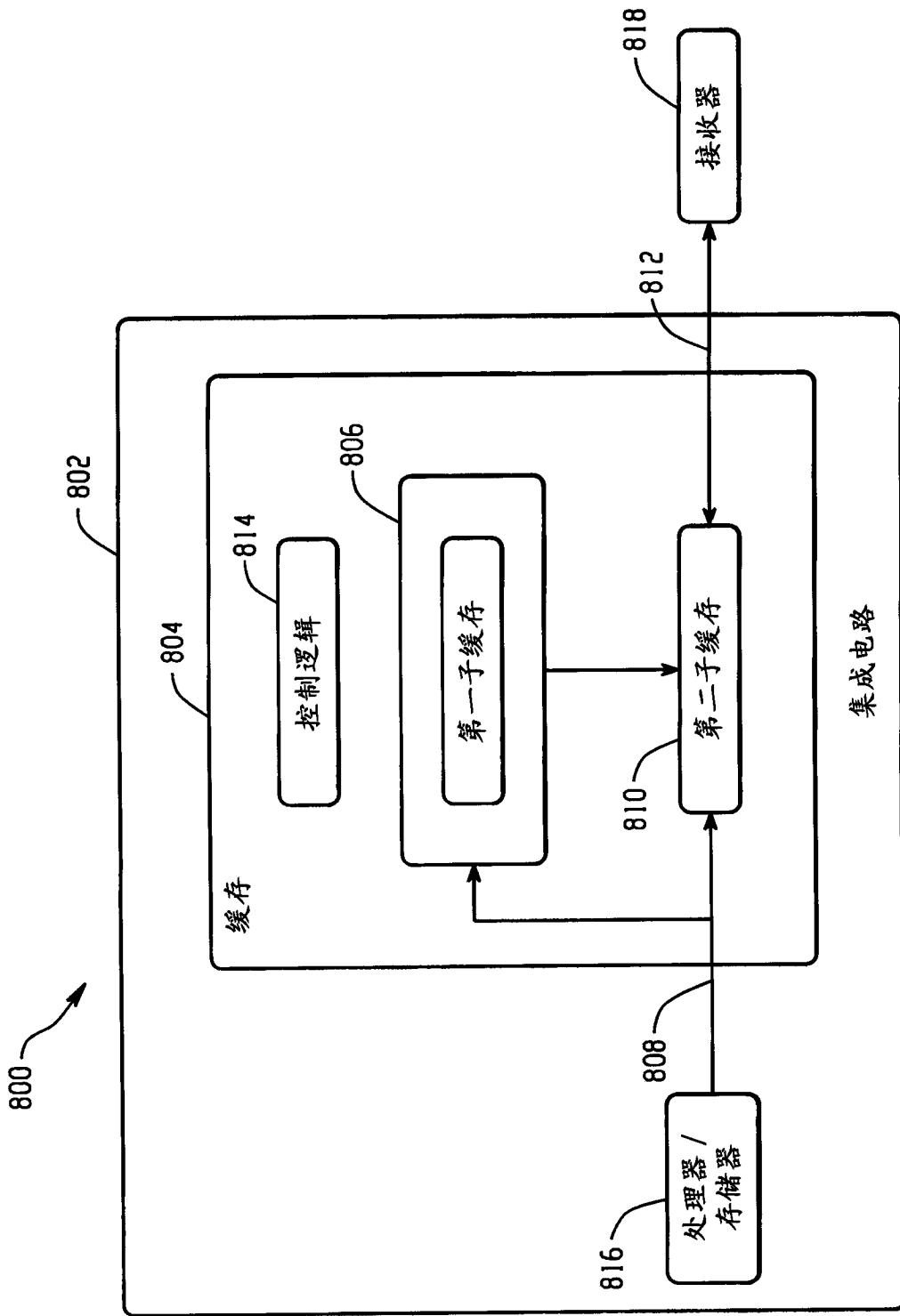


图 8

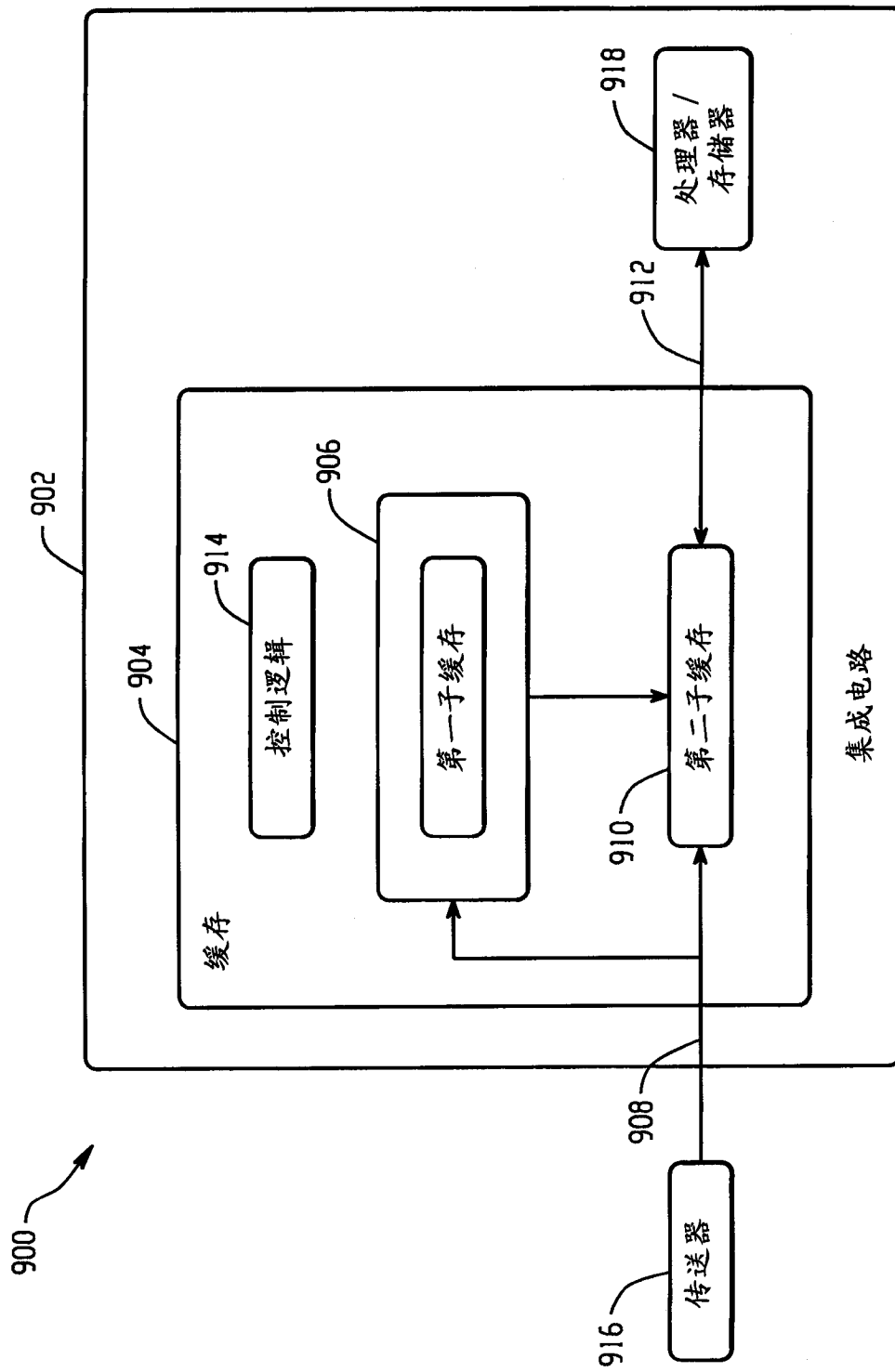


图 9