



(12) 发明专利申请

(10) 申请公布号 CN 102087590 A

(43) 申请公布日 2011.06.08

(21) 申请号 200910155140.5

(22) 申请日 2009.12.03

(71) 申请人 浙江大学

地址 310027 浙江省杭州市西湖区浙大路  
38 号

(72) 发明人 傅可威 高金加 孟建熠 严晓浪

(74) 专利代理机构 杭州天正专利事务所有限公司 33201

代理人 王兵 王利强

(51) Int. Cl.

G06F 7/57(2006.01)

G06F 9/38(2006.01)

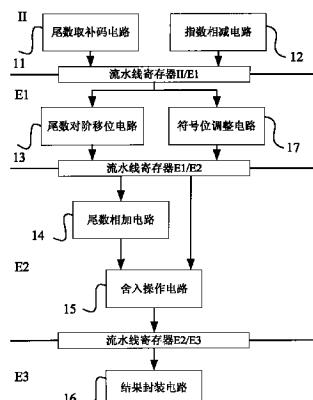
权利要求书 2 页 说明书 10 页 附图 8 页

(54) 发明名称

资源复用的浮点 SIMD 指令的执行装置

(57) 摘要

一种资源复用的浮点 SIMD 指令的执行装置，包括尾数取补码电路、指数相减电路、尾数对阶移位电路、尾数相加电路、尾数和舍入操作电路和结果封装电路；还包括 SIMD 逻辑运算指令；在各个具体执行电路中，SIMD 逻辑运算指令的低路操作数和高路操作数均复用双精度浮点数的运算的硬件资源。本发明提供一种能够加快单精度浮点运算的执行速度的资源复用的浮点 SIMD 指令的执行装置。



1. 一种资源复用的浮点 SIMD 指令的执行装置, 该执行装置包括 :

尾数取补码电路, 用于将操作数尾数取补码, 统一加减法运算逻辑, 包括双精度浮点数的取补加法器, 单精度指令复用双精度浮点数的取补加法器的低路 ;

指数相减电路, 用于得到两组操作数的指数差和大小关系, 为尾数对阶移位电路准备控制信号, 包括指数相减加法器, 所述指数相减加法器分为高路和低路, 单精度指令复用指数相减加法器的低路 ;

尾数对阶移位电路, 用于选择较小的操作数尾数进行对阶移位, 使得浮点加减法的两个操作数指数相等, 提供尾数相加电路的输入数据, 包括双精度浮点数的尾数对阶移位器, 所述尾数对阶移位器包括高路的尾数对阶小移位器和低路的尾数对阶小移位器, 单精度指令复用低路的尾数对阶小移位器,

尾数相加电路, 用于完成两个操作数尾数的相加, 得到补码表示的尾数和, 为舍入操作做准备, 包括双精度尾数相加加法器, 单精度指令复用所述双精度尾数相加加法器的低路 ;

尾数和舍入操作电路, 用于完成浮点加减法的舍入操作, 包括双精度舍入加法器, 单精度指令复用所述双精度舍入加法器 ;

结果封装电路, 用于对尾数和指数进行规格化调整, 将运算结果表示成规格化的浮点形式, 包括双精度尾数和规格化调整的大移位器, 所述大移位器包括低路规格化小移位器和高路规格化小移位器, 单精度指令复用低路规格化小移位器 ; 还包括指数规格化调整加法器, 所述指数规格化调整加法器包括低路指数规格化加法器和高路指数规格化加法器, 单精度指令复用低路指数规格化加法器 ;

其特征在于 : 所述执行装置还包括 SIMD 逻辑运算指令 ;

在所述尾数取补码电路中, SIMD 逻辑运算指令的低路操作数的尾数取补码操作复用双精度浮点数的取补加法器的低路, SIMD 逻辑运算指令的高路操作数的尾数取补码操作复用双精度浮点数的取补加法器的高路 ;

在所述指数相减电路中, SIMD 逻辑运算指令的低路操作数的指数相减操作复用双精度浮点数的指数相减加法器的低路, SIMD 逻辑运算指令的高路操作数的指数相减操作复用双精度浮点数的指数相减加法器的高路 ;

在所述尾数对阶移位电路中, SIMD 逻辑运算指令的低路操作数的尾数对阶移位操作复用双精度浮点数的尾数对阶移位器低路的小尾数对阶移位器, SIMD 逻辑运算指令的高路操作数的尾数对阶移位操作复用双精度浮点数的数相减加法器高路的小尾数对阶移位器 ;

在所述尾数相加电路中, SIMD 逻辑运算指令的低路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的低路, SIMD 逻辑运算指令的高路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的高路 ;

在所述尾数和舍入操作电路中, SIMD 逻辑运算指令的低路操作数的尾数和舍入操作复用双精度舍入加法器的低路, SIMD 逻辑运算指令的高路操作数的尾数和舍入操作复用双精度舍入加法器的高路 ;

在所述结果封装电路中, SIMD 逻辑运算指令的低路操作数的对尾数和进行规格化调整操作复用低路小规格化移位器, SIMD 逻辑运算指令的高路操作数的对尾数和进行规格化调整操作复用高路小规格化移位器 ; SIMD 逻辑运算指令的低路操作数的对指数进行规格化

调整操作复用低路规格化加法器, SIMD 逻辑运算指令的高路操作数的对指数进行规格化调整操作复用高路规格化加法器。

2. 如权利要求 1 所述的资源复用的浮点 SIMD 指令的执行装置, 其特征在于 : 所述执行装置还包括 : 符号位调整电路, 用于针对浮点操作数的符号位进行取反或置 0。

3. 如权利要求 1 所述的资源复用的浮点 SIMD 指令的执行装置, 其特征在于 : 所述 SIMD 逻辑运算指令包括浮点加法 SIMD 指令或者浮点减法 SIMD 指令。

4. 如权利要求 2 所述的资源复用的浮点 SIMD 指令的执行装置, 其特征在于 : 所述 SIMD 逻辑运算指令包括浮点取绝对值 SIMD 指令或者浮点取反 SIMD 指令。

5. 如权利要求 1-4 之一所述的所述的资源复用的浮点 SIMD 指令的执行装置, 其特征在于 : 所述执行装置还包括 : 预数零逻辑电路, 用于计算尾数和中第一个 1 出现时的偏移量。

6. 如权利要求 1-4 之一所述的所述的资源复用的浮点 SIMD 指令的执行装置, 其特征在于 : 双精度指令复用所述指数相减加法器的低路和部分高路, 双精度指令复用低路指数规格化加法器和部分的高路指数规格化加法器。

## 资源复用的浮点 SIMD 指令的执行装置

### 技术领域

[0001] 本发明涉及浮点 SIMD(单指令多数据) 指令的算术逻辑执行装置, 尤其是复用单精度或双精度浮点运算资源的算术逻辑执行装置。

### 背景技术

[0002] 现有技术中, 根据 Oberman 的技术报告, 在浮点运算中浮点加减执行单元的使用频度约为 55%。浮点加减法执行速度的加快对提升浮点算术逻辑部件的性能具有重要意义。

[0003] 浮点加减法的执行包含以下步骤: 指数相减、尾数对阶移位、尾数相加、舍入操作和结果封装。图 1 示出了典型的单精度或双精度的浮点加减运算的数据通路。其中 II(Instruction Issue) 表示该流水线级为指令发射级, E1(Execution 1<sup>st</sup>) 表示该流水线级为运算执行第一级, E2(Execution 2<sup>nd</sup>) 表示该流水线级为运算执行第二级, E3(Execution 3<sup>rd</sup>) 表示该流水线级为运算执行第三级。

[0004] 尾数取补码电路 11 根据运算类型(加法或减法)来计算其中一个浮点源操作数的补码, 从而统一加法运算和减法运算。该电路需要一个 25/54 位的加法器来完成单精度 / 双精度浮点数的尾数取补运算。

[0005] 指数相减电路 12 计算两个浮点源操作数的指数差, 以控制尾数对阶移位电路 13。单精度 / 双精度浮点数的指数相减运算需要一个 9/12 位的加法器。

[0006] 尾数对阶移位电路 13 对指数较小的浮点源操作数进行移位操作, 使两个操作数的指数相等。该电路需要一个 48/106 位的移位器来完成单精度 / 双精度浮点数的尾数移位操作。

[0007] 尾数相加电路 14 计算出两个操作数的尾数和。单精度 / 双精度浮点数的尾数相加运算需要一个 27/56 位的加法器。

[0008] 舍入操作电路 15 根据尾数和以及尾数对阶移位 13 提供的信息对尾数和进行舍入操作。该电路需要一个 26/55 位的舍入加法器来完成舍入操作。

[0009] 结果封装电路 16 对舍入操作后的尾数和以及指数进行规格化调整, 得到最终的计算结果。结果封装时, 单精度 / 双精度浮点数的尾数移位需要一个 26/55 位的移位器; 单精度 / 双精度浮点数的指数调整需要一个 8/11 位的加法器。

### 发明内容

[0010] 为了克服已有的算术逻辑执行装置处理单精度和双精度浮点运算时执行速度较慢的不足, 本发明提供一种能够加快单精度浮点运算的执行速度的资源复用的浮点 SIMD 指令的执行装置。

[0011] 本发明解决其技术问题所采用的技术方案是:

[0012] 一种资源复用的浮点 SIMD 指令的执行装置, 该执行装置包括:

[0013] 尾数取补码电路, 用于将操作数尾数取补码, 统一加减法运算逻辑, 包括双精度浮

点数的取补加法器,单精度指令复用双精度浮点数的取补加法器的低路;

[0014] 指数相减电路,用于得到两组操作数的指数差和大小关系,为尾数对阶移位电路准备控制信号,包括指数相减加法器,所述指数相减加法器分为高路和低路,单精度指令复用指数相减加法器的低路;

[0015] 尾数对阶移位电路,用于选择较小的操作数尾数进行对阶移位,使得浮点加减法的两个操作数指数相等,提供尾数相加电路的输入数据,包括双精度浮点数的尾数对阶移位器,所述尾数对阶移位器包括高路的尾数对阶小移位器和低路的尾数对阶小移位器,单精度指令复用低路的小移位器;

[0016] 尾数相加电路,用于完成两个操作数尾数的相加,得到补码表示的尾数和,为舍入操作做准备,包括双精度尾数相加加法器,单精度指令复用所述双精度尾数相加加法器的低路;

[0017] 尾数和舍入操作电路,用于完成浮点加减法的舍入操作,包括双精度舍入加法器,单精度指令复用所述双精度舍入加法器;

[0018] 结果封装电路,用于对尾数和和指数进行规格化调整,将运算结果表示成规格化的浮点形式,包括双精度尾数和规格化调整的大移位器,所述大移位器包括低路规格化小移位器和高路规格化小移位器,单精度指令复用低路规格化小移位器,还包括指数规格化调整的加法器,所述加法器包括低路指数规格化加法器和高路指数规格化加法器,单精度指令复用低路指数规格化加法器,双精度指令复用低路指数规格化加法器和部分的高路指数规格化加法器;

[0019] 所述执行装置还包括 SIMD 逻辑运算指令;

[0020] 在所述尾数取补码电路中,SIMD 逻辑运算指令的低路操作数的尾数取补码操作复用双精度浮点数的取补加法器的低路,SIMD 逻辑运算指令的高路操作数的尾数取补码操作复用双精度浮点数的取补加法器的高路;

[0021] 在所述指数相减电路中,SIMD 逻辑运算指令的低路操作数的指数相减操作复用指数相减加法器的低路,SIMD 逻辑运算指令的高路操作数的指数相减操作复用指数相减加法器的高路;

[0022] 在所述尾数对阶移位电路中,SIMD 逻辑运算指令的低路操作数的尾数对阶移位操作复用双精度浮点数的尾数对阶移位器低路的尾数对阶小移位器,SIMD 逻辑运算指令的高路操作数的尾数对阶移位操作复用双精度浮点数的数相减加法器高路的尾数对阶小移位器;

[0023] 在所述尾数相加电路中,SIMD 逻辑运算指令的低路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的低路,SIMD 逻辑运算指令的高路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的高路;

[0024] 在所述尾数和舍入操作电路中,SIMD 逻辑运算指令的低路操作数的尾数和舍入操作复用双精度舍入加法器的低路,SIMD 逻辑运算指令的高路操作数的尾数和舍入操作复用双精度舍入加法器的高路;

[0025] 在所述结果封装电路中,SIMD 逻辑运算指令的低路操作数的对尾数和进行规格化调整操作复用低路规格化小移位器,SIMD 逻辑运算指令的高路操作数的对尾数和进行规格化调整操作复用高路规格化小移位器,SIMD 逻辑运算指令的低路操作数的对指数进行规格

化调整操作复用低路规格化加法器, SIMD 逻辑运算指令的高路操作数的对指数进行规格化调整操作复用高路规格化加法器。

[0026] 所述 SIMD 逻辑运算指令包括浮点加法 SIMD 指令或者浮点减法 SIMD 指令。

[0027] 作为优选的一种方案:所述执行装置还包括:符号位调整电路,用于针对浮点操作数的符号位进行取反或置 0。

[0028] 进一步,所述 SIMD 逻辑运算指令包括浮点取反 SIMD 指令或者 浮点取绝对值 SIMD 指令。当执行浮点取绝对值 SIMD 指令时,符号调整电路进行置 0 操作;当执行浮点取反 SIMD 指令时,符号调整电路进行取反。

[0029] 再进一步,所述执行装置还包括:预数零逻辑电路,用于计算尾数和中第一个 1 出现时的偏移量。通过采用预数零和规格化校准逻辑,加快运算结果的获得。

[0030] 双精度指令复用所述指数相减加法器的低路和部分高路,双精度指令复用低路指数规格化加法器和部分的高路指数规格化加法器。

[0031] 本发明的技术构思为:单指令多数据 (Single Instruction MultipleData) 操作是一种微处理器设计中常用的用于提升运算速度、减少指令数量的技术。该技术的本质是利用双精度数据的存储空间存放两组相互独立的单精度数据,两组同时进行运算。该技术的存在减少了完成大量数据进行相同运算的任务(例如多媒体图形、图像处理等)所必须执行的指令的总数量,使得单位时间内浮点算术逻辑单元的数据吞吐量增加了一倍。浮点算术逻辑单元数据吞吐量的增加对提高该任务的完成速度、降低完成该任务所需功耗具有重要意义。

[0032] 浮点取绝对值 SIMD 指令和浮点取反 SIMD 指令在指令类型上归属于浮点加减法 SIMD 指令。通过将浮点加减法 SIMD 指令的第二个源操作数固定为零,这两种指令可以完全复用浮点加减法 SIMD 指令的执行电路。例如:浮点数 A 的绝对值  $|A| = |A+0|$ ;浮点数 A 的取反值  $\sim A = \sim (A+0)$ 。唯一需要增加的是符号位调整电路 17。符号调整电路 17 根据指令类型是取反指令还是取绝对值指令分别对符号位进行取反或置 0(0 表示正数)。

[0033] 为支持浮点运算的微处理器指令集在增加很少硬件资源的前提下,增加一条浮点加法 SIMD 指令(形如 FADDM),用于加快浮点加法的执行速度。

[0034] 为支持浮点运算的微处理器指令集在增加很少硬件资源的前提下,增加一条浮点减法 SIMD 指令(形如 FSUBM),用于加快浮点减法的执行速度。

[0035] 为支持浮点运算的微处理器指令集在增加很少硬件资源的前提下,增加一条浮点取绝对值 SIMD 指令(形如 FABSM),用于加快浮点取绝对值运算的执行速度。

[0036] 为支持浮点运算的微处理器指令集在增加很少硬件资源的前提下,增加一条浮点取反 SIMD 指令(形如 FNEGM),用于加快浮点取反运算的执行速度。

[0037] 本发明的有益效果主要表现在:加快执行装置的运算速度。

## 附图说明

[0038] 图 1 是典型的单精度或双精度浮点加减法运算的数据通路的示意图。

[0039] 图 2 是四条浮点 SIMD 指令的活动的示意图,其中,图 2(a) 表示 FADDM  $R_d, R_m, R_n$ , 图 2(b) 表示 FSUBM  $R_d, R_m, R_n$ , 图 2

(c) 表示 FABSM  $R_d, R_n$ , 图 2(d) 表示 FNEGM  $R_d, R_n$ 。

- [0040] 图 3 是硬件资源复用的尾数取补码电路的示意图。
- [0041] 图 4 是硬件资源复用的指数相减电路的示意图。
- [0042] 图 5 是硬件资源复用的尾数对阶移位电路的示意图。
- [0043] 图 6 是硬件资源复用的尾数相加电路的示意图。
- [0044] 图 7 是硬件资源复用的尾数和舍入操作电路的示意图。
- [0045] 图 8 是硬件资源复用的结果封装电路的示意图。
- [0046] 图 9 是双精度指令和 SIMD 指令的符号位调整电路的示意图, 其中, 图 9(a) 表示双精度取反 / 取绝对值指令符号位调整电路, 图 9(b) 表示 SIMD 取反 / 取绝对值指令符号位调整电路。

### 具体实施方式

- [0047] 下面结合附图对本发明作进一步描述。
- [0048] 实施例 1
- [0049] 参照图 2 ~ 图 8, 一种资源复用的浮点 SIMD 指令的执行装置, 该执行装置包括:
- [0050] 尾数取补码电路, 用于将操作数尾数取补码, 统一加减法运算逻辑, 包括双精度浮点数的取补加法器, 单精度指令复用所述双精度浮点数取补加法器的低路;
- [0051] 指数相减电路, 用于得到两组操作数的指数差和大小关系, 为尾数对阶移位电路准备控制信号, 包括指数相减加法器, 单精度指令复用所述指数相减加法器的低路;
- [0052] 尾数对阶移位电路, 用于选择较小的操作数尾数进行对阶移位, 使得浮点加减法的两个操作数指数相等, 提供尾数相加电路的输入数据, 包括双精度浮点数的尾数对阶移位器, 所述尾数对阶移位器包括高路的尾数对阶小移位器和低路的尾数对阶小移位器, 单精度指令复用低路的小移位器;
- [0053] 尾数相加电路, 用于完成两个操作数尾数的相加, 得到补码表示的尾数和, 为舍入操作做准备, 包括双精度尾数相加加法器, 单精度指令复用所述双精度尾数相加加法器的低路;
- [0054] 尾数和舍入操作电路, 用于完成浮点加减法的舍入操作, 包括双精度舍入加法器, 单精度指令复用所述双精度舍入加法器;
- [0055] 结果封装电路, 用于对尾数和和指数进行规格化调整, 将运算结果表示成规格化的浮点形式, 包括双精度尾数和规格化调整的大移位器, 所述大移位器包括低路规格化小移位器和高路规格化小移位器, 单精度指令复用低路规格化小移位器; 还包括指数规格化调整加法器, 所述指数规格化调整加法器包括低路指数规格化加法器和高路指数规格化加法器, 单精度指令复用低路指数规格化加法器;
- [0056] 所述执行装置还包括 SIMD 逻辑运算指令;
- [0057] 在所述尾数取补码电路中, SIMD 逻辑运算指令的低路操作数的尾数取补码操作复用双精度浮点数的取补加法器的低位, SIMD 逻辑运算指令的高路操作数的尾数取补码操作复用双精度浮点数的取补加法器的高位;
- [0058] 在所述指数相减电路中, SIMD 逻辑运算指令的低路操作数的指数相减操作复用双精度浮点数的指数相减加法器的低位, SIMD 逻辑运算指令的高路操作数的指数相减操作复用双精度浮点数的指数相减加法器的高位;
- [0059] 在所述尾数对阶移位电路中, SIMD 逻辑运算指令的低路操作数的尾数对阶移位操

作复用双精度浮点数的尾数对阶移位器低路的尾数对阶小移位器, SIMD 逻辑运算指令的高路操作数的尾数对阶移位操作复用双精度浮点数的数相减加法器高路的尾数对阶小移位器;

[0060] 在所述尾数相加电路中, SIMD 逻辑运算指令的低路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的低路, SIMD 逻辑运算指令的高路操作数的尾数相加操作复用双精度浮点数的尾数相加加法器的高路;

[0061] 在所述尾数和舍入操作电路中, SIMD 逻辑运算指令的低路操作数的尾数和舍入操作复用双精度舍入加法器的低路, SIMD 逻辑运算指令的高路操作数的尾数和舍入操作复用双精度舍入加法器的高路;

[0062] 预数零逻辑电路, 用于计算尾数和中第一个 1 出现时的偏移量;

[0063] 在所述结果封装电路中, SIMD 逻辑运算指令的低路操作数的对尾数和进行规格化调整操作复用低路规格化小移位器, SIMD 逻辑运算指令的高路操作数的对尾数和进行规格化调整操作复用高路规格化小移位器; SIMD 逻辑运算指令的低路操作数的对指数进行规格化调整操作复用低路规格化加法器, SIMD 逻辑运算指令的高路操作数的对指数进行规格化调整操作复用高路规格化加法器。

[0064] 双精度指令复用所述指数相减加法器的低路和部分高路, 双精度指令复用低路指数规格化加法器和部分的高路指数规格化加法器。

[0065] 本实施例的 SIMD 逻辑运算指令包括浮点加法 SIMD 指令或者浮点减法 SIMD 指令。

[0066] 图 2 示出了本装置所执行的指令活动。本装置支持的指令集包括浮点加法的 SIMD 指令 (FADDM)、浮点减法的 SIMD 指令 (FSUBM)、浮点取绝对值的 SIMD 指令 (FABSM) 和浮点取反的 SIMD 指令 (FNEGM)。

[0067] SIMD 指令的源操作数利用 64 位的双精度 (double) 浮点数的存储空间存放两个 32 位的单精度 (Single) 浮点数。SIMD 低路 (双精度浮点格式寄存器的低 32 位) 和 SIMD 高路 (双精度浮点格式寄存器的高 32 位) 在逻辑上完全独立。

[0068] 图 2(a) 示出了浮点加法 SIMD 指令的数据流。其汇编语言表示类似 :FADDM R<sub>d</sub>, R<sub>m</sub>, R<sub>n</sub>

[0069] 这里, “FADDM”代表操作码, 标识了必须执行的操作, R<sub>d</sub>、R<sub>m</sub>、R<sub>n</sub> 代表双精度浮点格式的寄存器, 每个寄存器的内容都被看做两个独立的 32 位的单精度操作数。该指令表示将源寄存器 R<sub>m</sub>、R<sub>n</sub> 中的低 32 位 (即图 2(a) 中的 OP11 和 OP21) 相加后存放于 R<sub>d</sub> 的低 32 位, 将源寄存器 R<sub>m</sub>、R<sub>n</sub> 中的高 32 位 (即图 2(a) 中的 OP1h 和 OP2h) 相加后存放于 R<sub>d</sub> 的高 32 位, 从而用一条指令完成了两个浮点加法运算。

[0070] 图 2(b) 示出了浮点减法 SIMD 指令的数据流。其汇编语言表示类似 :FSUBM R<sub>d</sub>, R<sub>m</sub>, R<sub>n</sub>

[0071] 这里, “FSUBM”代表操作码, 标识了必须执行的操作, R<sub>d</sub>、R<sub>m</sub>、R<sub>n</sub> 的意义与浮点加法的 SIMD 指令相同。该指令表示将源寄存器 R<sub>m</sub>、R<sub>n</sub> 中的低 32 位 (即图 2(b) 中的 OP11 和 OP21) 相减后存放于 R<sub>d</sub> 的低 32 位, 将源寄存器 R<sub>m</sub>、R<sub>n</sub> 中的高 32 位 (即图 2(b) 中的 OP1h 和 OP2h) 相减后存放于 R<sub>d</sub> 的高 32 位, 从而用一条指令完成了两个浮点减法运算。

[0072] 图 2(c) 示出了浮点取绝对值 SIMD 指令的数据流。其汇编语言表示类似 :

[0073] FABSM R<sub>d</sub>, R<sub>n</sub>

[0074] 这里,“FABSM”代表操作码,标识了必须执行的操作, $R_d$ 、 $R_n$ 的意义与浮点加法的 SIMD 指令相同。该指令表示将源寄存器  $R_n$  中的低 32 位(即图 2(c) 中的 OP1l)取绝对值后存放于  $R_d$  的低 32 位,将源寄存器  $R_n$  中的高 32 位(即图 2(c) 中的 OP1h)取绝对值后存放于  $R_d$  的高 32 位,从而用一条指令完成了两个浮点取绝对值运算。

图 2(d) 示出了浮点取反 SIMD 指令的数据流。其汇编语言表示类似:

FNEGM  $R_d$ ,  $R_n$

[0075] 这里,“FNEGM”代表操作码,标识了必须执行的操作, $R_d$ 、 $R_n$ 的意义与浮点加法的 SIMD 指令相同。该指令表示将源寄存器  $R_n$  中的低 32 位(即图 2(d) 中的 OP1l)符号取反后存放于  $R_d$  的低 32 位,将源寄存器  $R_n$  中的高 32 位(即图 2(d) 中的 OP1h)符号取反后存放于  $R_d$  的高 32 位,从而用一条指令完成了两个浮点取反运算。图 3 至图 8 分别是本装置的各个子电路,其具体实施电路是按照流水线组织的。

[0076]

[0077] 图 3 示出了本装置的操作数尾数取补码电路,其特征是 SIMD 指令完全复用双精度浮点数取补码运算的加法器(图中标号为 35)。对于双精度浮点数来说,取补码的操作数包括{1bit 符号位,1bit 规格化浮点数的前导 1,52bits 尾数位},因此,至少需要一个 54 位取补加法器。对单精度浮点数而言,取补码的操作数包括{1bit 符号位,1bit 规格化浮点数的前导 1,23bits 尾数位},共 25 位,而 SIMD 指令共需  $2 \times 25 = 50$  位的取补加法器。因此,单精度指令和 SIMD 指令的操作数取补加法器可以完全复用双精度浮点数的 54 位取补加法器。在一个具体实施例中,双精度指令的尾数取补码操作需要使用整个 54 位的加法器,单精度指令的操作数取补加法器复用双精度浮点数取补加法器的低 25 位。SIMD 指令低路操作数的尾数取补码操作可以复用加法器 35 的低 25 位,SIMD 指令高路操作数的尾数取补码操作可以复用加法器 35 的高 25 位,中间用 4 个零间隔。其他合理的比特位分布方案也可以达到复用的目标。

[0078] 图 3 中标号为 33 的操作数尾数选择逻辑根据指令类型(single/double/SIMD,即单精度/双精度/单指令多数据)从 64 位浮点寄存器  $R_{2m}$  和  $R_{2n}$  中提取出正确的操作数。对于实际减法的指令(实际减法是指同号两数相减或异号两数相加,实际加法是指同号两数相加或异号两数相减),需要对第二个操作数进行取补码操作,由数据选择器(MUX,图中标号为 34)选取正确数据。取补码操作包括取反和加 1 两步,因此取补加法器的输入进位信号可能包括以下五种情况:

[0079] ①单精度和双精度指令实际减法的情况:取补加法器输入进位位为 1;

[0080] ②单精度、双精度指令和 SIMD 指令两路均为实际加法的情况:取补加法器输入进位位为 0;

[0081] ③ SIMD 指令两路均为实际减法的情况:取补加法器输入进位位在两个对应尾数的最低位上加 1;

[0082] ④ SIMD 指令高路为实际减法、低路为实际加法的情况:取补加法器在高路尾数的最低位上加 1,而在低路尾数的最低位上加 0;

[0083] ⑤ SIMD 指令高路为实际加法、低路为实际减法的情况:取补加法器在高路尾数的最低位上加 0,而在低路尾数的最低位上加 1。

[0084] 图 3 中标号为 35 的加法器输入进位选择电路根据以上 5 中情况选择对应的取补

加法器输入进位信号。

[0085] 最后该电路将取补码后的操作数存放于流水线寄存器中用于下一级尾数的对阶移位电路使用。

[0086] 图4示出了本装置的操作数指数相减电路,其特征是 SIMD 指令完全复用单精度和双精度浮点指令的指数相减加法器(图中标号为44)。对于双精度浮点数来说,加法器44需要{1bit 符号位,11bits 指数位},共 12 位;对于单精度浮点数来说,加法器44需要{1bit 符号位,8bits 指数位},共 9 位;对于 SIMD 指令来说,指数相减加法器需要  $2 \times 9 = 18$  位。因此在一种可行的实施方案中,使用 18 比特的指数相减加法器,双精度指令使用该加法器的低 12 位,单精度指令使用加法器的低 9 位, SIMD 指令低路使用该加法器的低 9 位, SIMD 指令高路使用该加法器的高 9 位。其它合理的方案,例如“将加法器 44 扩展位 21 位,双精度指令占用低 12 位,单精度指令占用高 9 位, SIMD 指令高低路分别占用高低 9 位,中间用零间隔”或者改变各类型指令占用加法器的位置等,均可以达到目标。其共同的特征是,在双精度和单精度指令的硬件基础上,增加少量硬件开销,实现 SIMD 指令指数相减的功能。

[0087] 图4中标号为43的操作数指数选择逻辑根据指令类型(single/double/SIMD,即单精度 / 双精度 / 单指令多数据)从64位浮点寄存器R<sub>2m</sub>和R<sub>2n</sub>中提取出正确的操作数指数,为指数相减加法器准备输入。

[0088] 在一个具体实施例中,图3和图4示出的操作数尾数取补码操作和指数相减操作安排在同一级流水线中并行执行,是本装置流水线的第一级。

[0089] 图5示出了操作数尾数的对阶移位电路,其特征是 SIMD 指令完全复用双精度指令的尾数对阶移位器(图中标号为56、57)。双精度浮点数的有效数字共有 53 位,因此双精度尾数对阶移位器至少需要 106 比特,其原因是:对于就近舍入(round to nearest, Std. IEEE754-1985 浮点运算标准中定义的基本舍入模式),需要保留足够的移位数字用于正确舍入,若移位位数小于等于 53 位,则 106 比特的移位器自然可以满足要求;若移位位数大于 53 位,则移出数 字的最高位必定为 0,则可以判定移出数字必定小于 0.5,所以 106 比特移位器可以完全保证舍入信息的完整性。同理,对于单精度浮点数而言,尾数对阶移位器只需要  $2 \times 24 = 48$  位;对于 SIMD 指令而言,尾数对阶移位器需要  $2 \times 48 = 96$  位,因此单精度和 SIMD 指令可以完全复用双精度指令的对阶移位器,而不需额外硬件开销。

[0090] 在一个具体实施例中,106位的尾数对阶移位器由两个53位的尾数对阶小移位器组成。当当前指令为双精度类型时,两个尾数对阶小移位器合成为一个大移位器;当当前指令为单精度指令时,只需使用低路的尾数对阶小移位器;当当前指令为 SIMD 时, SIMD 指令低路数据使用低路的尾数对阶小移位器(图中标号为57),高路数据使用高路的尾数对阶小移位器(图中标号为56)。这样实施的优势在于,即节省了硬件开销,又有利于低功耗的设计。其他合理的比特位分布方案也可以达到复用的目标。

[0091] 图5中标号52的操作数选择逻辑与图3、图4类似,根据当前指令类型从流水线寄存器中选取正确的尾数。标号53的操作数尾数交换逻辑根据上级指数相减得到的两操作数的大小情况,选择较小的操作数放入对阶移位器中进行对阶移位,同时将较大的操作数存入流水线寄存器中。

[0092] 图5示出的操作数尾数对阶移位电路是本具体实施例的第二级流水线。

[0093] 图6示出了尾数相加电路,其特征是 SIMD 指令完全复用双精度指令的尾数相加加

法器（图中标号为 65）。该加法器的两个输入是，流水线寄存器 E1/E2 中的未移位操作数 (op\_fix, 图中标号为 62) 和经操作数选择逻辑（图中标号为 64）选出的已移位操作数 (op shift, 图中标号为 63)。

[0094] 在一个具体实施例中，对于双精度类型指令，加法器 65 的输入是 {1bit 符号位, 1 比特进位位, 53bit 有效数位, 1bit 移出数字的最高位}，共 56 位；同理，对于单精度类型指令，加法器 65 的输入是 {1bit 符号位, 1 比特进位位, 24bit 有效数位, 1bit 移出数字的最高位}，共 27 位；对于 SIMD 类型指令，加法器 65 的输入共  $2 \times 27 = 54$  位。因此单精度指令和 SIMD 指令可以完全复用 56 位的双精度尾数相加加法器。在具体实施例中，单精度指令尾数相加复用加法器 65 的低 27 位；SIMD 指令低路的尾数相加复用加法器 65 的低 27 位，SIMD 指令高路的尾数相加复用加法器 65 的高 27 位，中间用 2 比特零间隔。其他合理的比特位分布方案也可以达到复用的目的。

[0095] 加法器 65 的运算结果即为用补码表示的尾数和，用于下一步的舍入操作和预数零操作。

[0096] 图 7 示出了尾数和的舍入操作电路，其特征是 SIMD 指令完全复用双精度指令的舍入加法器（图中标号为 77）。舍入加法器的特征是将补码表示的尾数和（图中标号为 73）求原码过程（取反加 1）中的加 1 操作和舍入进位逻辑相统一，共用一个舍入加法器。其原因是：求原码过程中的加 1 操作是在无限精度的尾数和的最低位上加 1，而舍入进位则是在规格化浮点数的最低有效位上加 1，因此求原码过程中的加 1 操作不一定会影响最终的结果。在一个具体实施例中，通过一种算法将这两个加 1 操作相统一，由舍入控制逻辑（图中标号为 74）判断是否需要在最低有效位上加 1，并以此来控制数据选择器 76 选择 0 或 1。数据选择器 75 则根据当前尾数和是否为负数来选取尾数的原始值或后尾数和的取反值。

[0097] 舍入准备逻辑（图中标号为 72）为舍入控制逻辑提供必要的舍入信息，即对阶移位过程中移出数字与 0.5 比较的大小情况，包括四种信息：移出数字大于 0.5，移出数字等于 0.5，移出数值小于 0.5 且不等于 0，移出数字等于 0。

[0098] 舍入控制逻辑（图中标号为 74）根据舍入准备逻辑提供的四条信息以及补码表示的尾数和，决定尾数和的最低有效位以及是否在尾数和的最低有效位上加 1。

[0099] 在一个具体实施例中，对于双精度类型指令，舍入加法器的输入是 55 位尾数和（尾数相加加法器的 56 位运算结果去掉一位符号位）；对于单精度类型指令，舍入加法器的输入是 26 位尾数和（尾数相加加法器的 27 位运算结果去掉一位符号位）；对于 SIMD 类型指令，舍入加法器的输入是  $2 \times 26 = 52$  位尾数和，因此，单精度和 SIMD 类型指令可以完全复用该 55 位舍入加法器。一种合理的比特位分配方案是：单精度类型指令占用舍入加法器的低 26 位；SIMD 指令的低路占用舍入加法器的低 26 位，SIMD 指令的高路占用舍入加法器的高 26 位，中间用三个零间隔。其他合理的比特位分配方案也可以达到复用目的。

[0100] 在一个具体实施例中，经舍入操作后的尾数和存放在流水线寄存器 E2/E3 中为下一级做准备。图 6 和图 7 示出的尾数相加电路和尾数和舍入电路组成了本装置的第三级流水线。

[0101] 图 8 示出了指数和尾数的规格化调整电路，其特征在于 SIMD 指令指数规格化调整的两个加法器（图中标号为 86、87）是在复用双精度类型指令的指数规格化调整加法器的基础上增加少量位宽实现的；SIMD 指令尾数和规格化调整的移位器复用双精度类型指令

的尾数和规格化移位器（图中标号为 88）。本电路的另一个特征是在上一级（E2 级）采用预数零逻辑，得到尾数和中第一个 1 出现时数过的数字个数，称为偏移量（bias，图中标号为 83），在本级（E3 级）进行校准，从而加快了运算结果的获得速度。

[0102] 在一个具体实施例中，对于实际加法的情况，尾数和只可能有两种情况：有一位进位、既无进位也无借位，所以数据选择器（图中标号为 85）选通零，表示尾数和的第一个 1 无偏移。因为预数零逻辑的输入是舍入前的尾数和，舍入操作可能造成进位，因此若舍入造成了进位，则应向指数加上 1，数据选择器（图中标号为 89）选择加法器 87 的结果作为最终结果的指数；若舍入无进位，则指数不需校准，数据选择器 89 选择加法器 86 的结果作为最终结果的指数。舍入是否有进位的信息存于流水线寄存器 E2/E3 中，由该信号控制数据选择器 89。需要注意的是，若预数零逻辑的输入是舍入后的尾数和，则不需要校准，但对时序延迟非常不利，本发明包括了这两种预数零逻辑。

[0103] 对于实际减法的情况，尾数和可能有以下三种情况：既无进位也无借位、有一位借位、有多位借位。对于第三种情况，则结果必然精确，舍入操作不影响结果。对于实际减法的情况，指数需要减去预数零得到的偏移量。若舍入造成了进位，则应向指数加上 1 校准，数据选择器 89 选择加法器 87 的结果作为最终结果的指数；若舍入没有造成进位，则指数不需校准，数据选择器 89 选择加法器 86 的结果作为最终结果的指数。

[0104] 在一个具体实施例中，双精度类型指令的指数规格化调整需要两个 11 位的加法器，单精度类型指令的指数规格化调整需要两个 8 位 加法器，SIMD 类型指令的指数规格化调整需要两个 16 位的加法器，因此一种可行的方案是采用两个 16 位的加法器来实现图中的加法器 86、87。一种可行的比特位分配方案是：双精度类型指令的指数规格化调整占用加法器 86、87 的低 11 位；单精度类型指令的指数规格化调整占用加法器 86、87 的低 8 位；SIMD 类型指令低路指数规格化调整占用加法器 86、87 的低 8 位，SIMD 类型指令高路指数规格化调整占用加法器 86、87 的高 8 位。这样在单精度和双精度指令执行的基础上，增加很少硬件资源，就实现了 SIMD 指令指数的规格化调整。其他合理的比特位分配方案也可以达到复用的目的。

[0105] 在一个具体实施例中，规格化移位器（图中标号为 88）的输入是上一级（E2 级）舍入操作的输出（存放在流水线寄存器 E2/E3 中）。因此双精度类型指令的尾数规格化移位需要一个 55 位的移位器；单精度类型指令的尾数规格化移位只需一个 26 位的移位器；SIMD 类型指令的尾数规格化移位需要两个 26 位的移位器。55 位的规格化移位器（图中标号为 78）由两个规格化小移位器组成：29 位的高路移位器和 26 位的低路移位器。当当前指令为双精度指令时，两个规格化小移位器合成为一个大移位器；当当前指令为单精度指令时，只需使用低路的规格化小移位器；当当前指令为 SIMD 指令时，SIMD 指令低路数据使用低路的规格化小移位器，高路数据使用高路的规格化小移位器。这样实施的优势在于，即节省了硬件开销，又有利于低功耗的设计。其他合理的比特位分布方案也可以达到复用的目标。

[0106] 实施例 2

[0107] 参照图 9，本实施例的执行装置还包括：符号位调整电路，用于 针对浮点操作数的符号位进行取反或置 0。

[0108] 所述 SIMD 逻辑运算指令包括浮点取绝对值 SIMD 指令或者浮点取反 SIMD 指令。

[0109] 浮点取绝对值 SIMD 指令和浮点取反 SIMD 指令在指令类型上归属于浮点加减法 SIMD 指令。通过将浮点加减法 SIMD 指令的第二个源操作数固定为零,这两种指令可以完全复用浮点加减法 SIMD 指令的执行电路。例如:浮点数 A 的绝对值  $|A| = |A+0|$ ;浮点数 A 的取反值 $\sim A = \sim (A+0)$ 。唯一需要增加的是符号位调整电路。图 9 分别示出了浮点取反 / 取绝对值双精度指令和浮点取反 / 取绝对值 SIMD 指令的符号位调整电路。浮点取反 / 取绝对值 SIMD 指令的符号位调整电路的特征是在复用浮点取反 / 取绝对值双精度的符号位调整电路的基础上,通过增加少量逻辑实现其功能。双精度取反指令只需对双精度浮点数最高位(即符号位,双精度浮点数的第 63 位)取反即可,双精度取绝对值指令只需将双精度浮点数最高位设置为 0(符号位为 0 表示浮点数为正,符号位为 1 表示浮点数为负)。同理,浮点取反 / 取绝对值单精度指令可通过将单精度浮点数放到双精度浮点数寄存器的高 32 位来复用双精度指令的符号位调整电路。对于浮点取反 / 取绝对值 SIMD 指令,其高路数据同单精度指令一样可以完全复用双精度指令的符号位调整电路;而低路数据则需增加额外逻辑来实现符号位调整的功能。

[0110] 在一个具体实施例中,符号位调整电路通过判断指令为取反或取绝对值,以及指令类型是否为 SIMD 指令来调整符号位。无论指令类型为单精度、双精度或者 SIMD,双精度浮点数寄存器的最高位都需要进行调整。因此需要使用数据选择器 92;当指令为取反指令时,数据选择器 92 选择取反后的符号位;当指令为取绝对值指令时,数据选择器 92 选择 0 作为符号位的值。当指令类型为 SIMD 指令时,还需对低路数据的符号位进行调整。数据选择器 93 实现了同数据选择器 92 类似的功能:当指令为取反指令时,数据选择器 93 选择取反后的符号位;当指令为取绝对值指令时,数据选择器 93 选择 0 作为符号位的值。在此基础上,当指令类型为 SIMD 指令时,数据选择器 94 选择调整过后的符号位作为低路数据的符号位;否则数据选择器 94 选择该位的原始数据写入流水线寄存器。

[0111] 在一个具体实施例中,调整过后的符号位存放在流水线寄存器 E1/E2 中供 E2 级流水线使用。

[0112] 应该认识到,本实施例根据各个电路的输出结果限定了上述以及如权利要求定义的硬件资源复用的电路结构。应该注意,可用多种不同的比特位分布策略来获得相同或部分相同的结果。应该注意,根据集成电路工艺的不同和主频设计指标的不同,流水线的划分有多种不同方式。本发明包括了所有这些资源复用的浮点 SIMD 指令执行装置电路结构的所有变化形式。

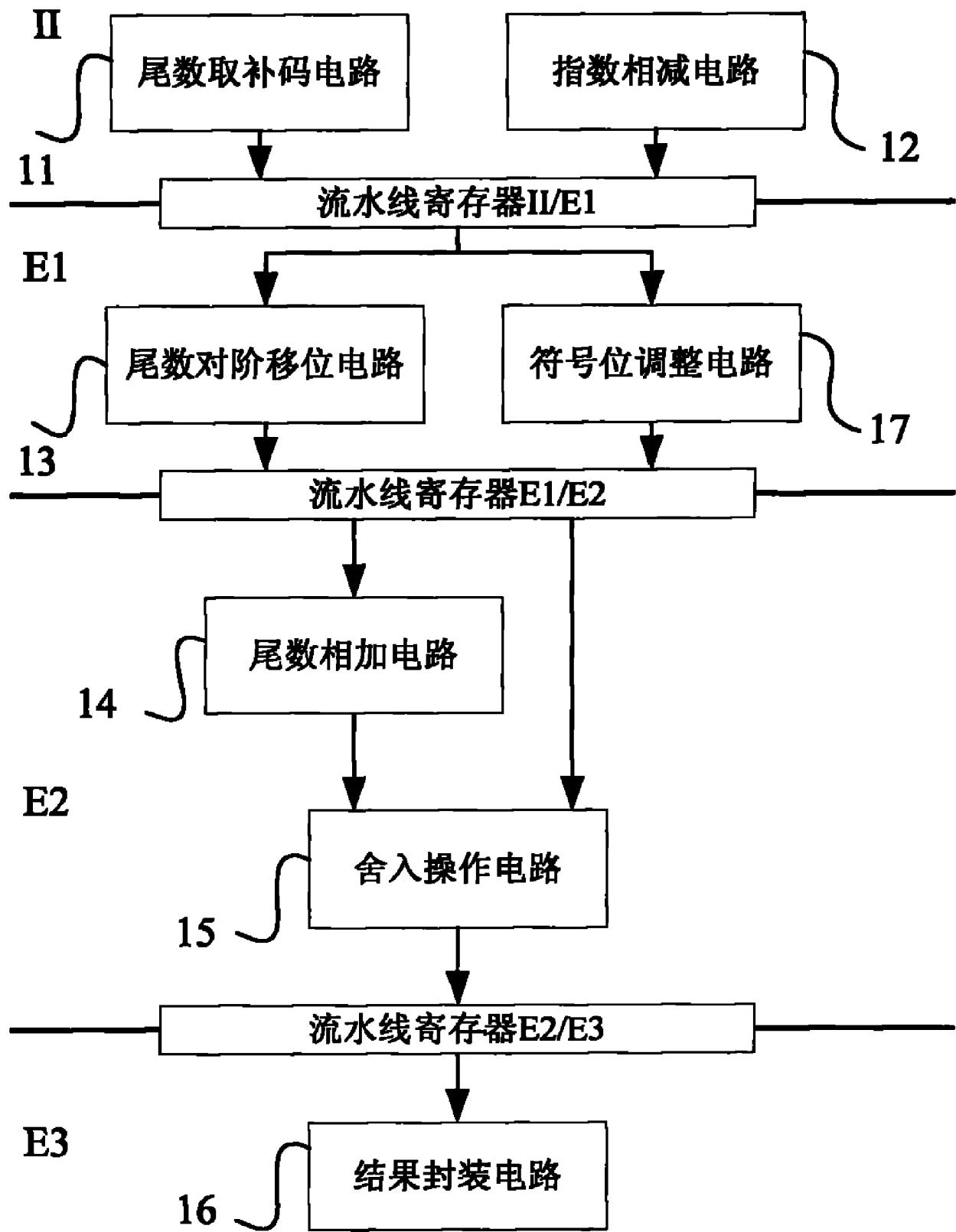


图 1

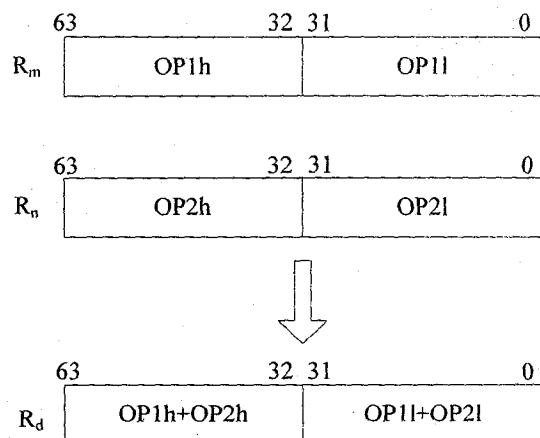


图2 (a)

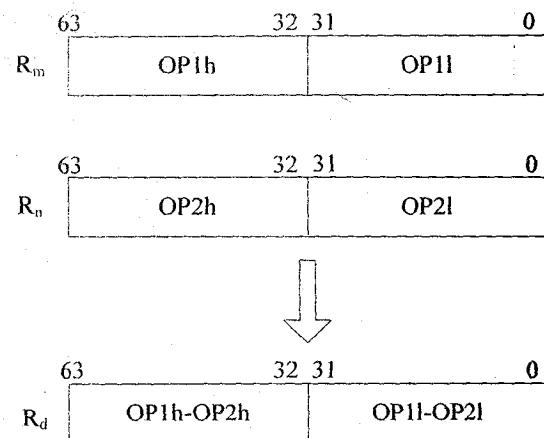


图2 (b)

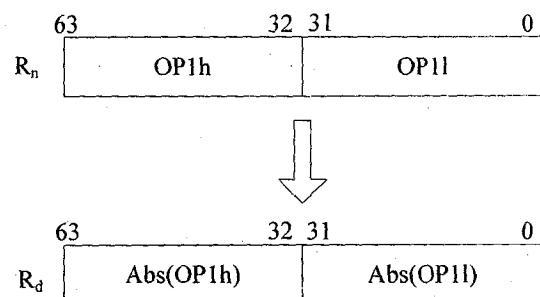


图2 (c)

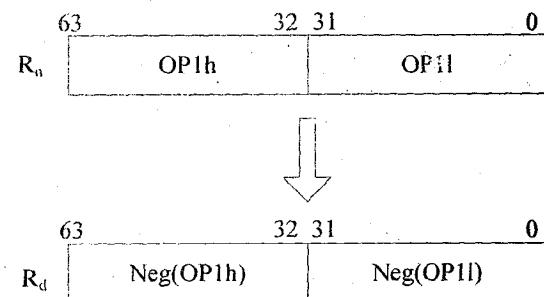


图2 (d)

图 2

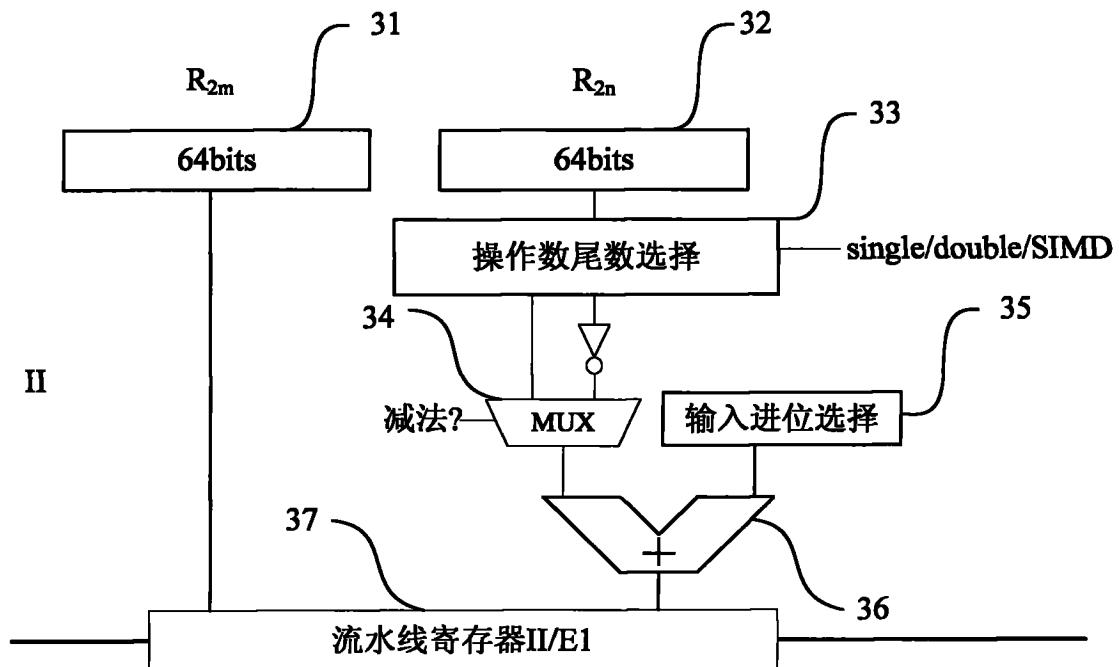


图 3

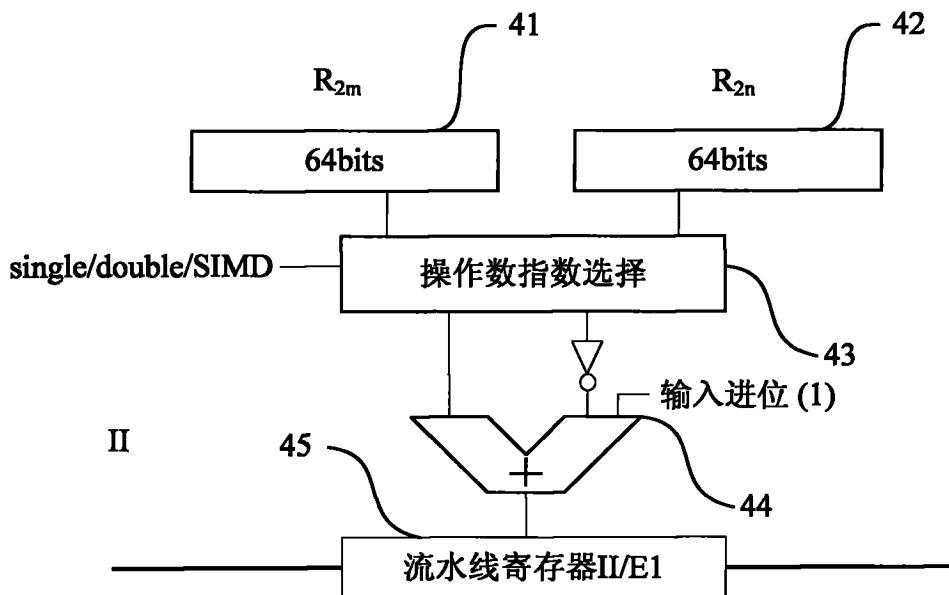


图 4

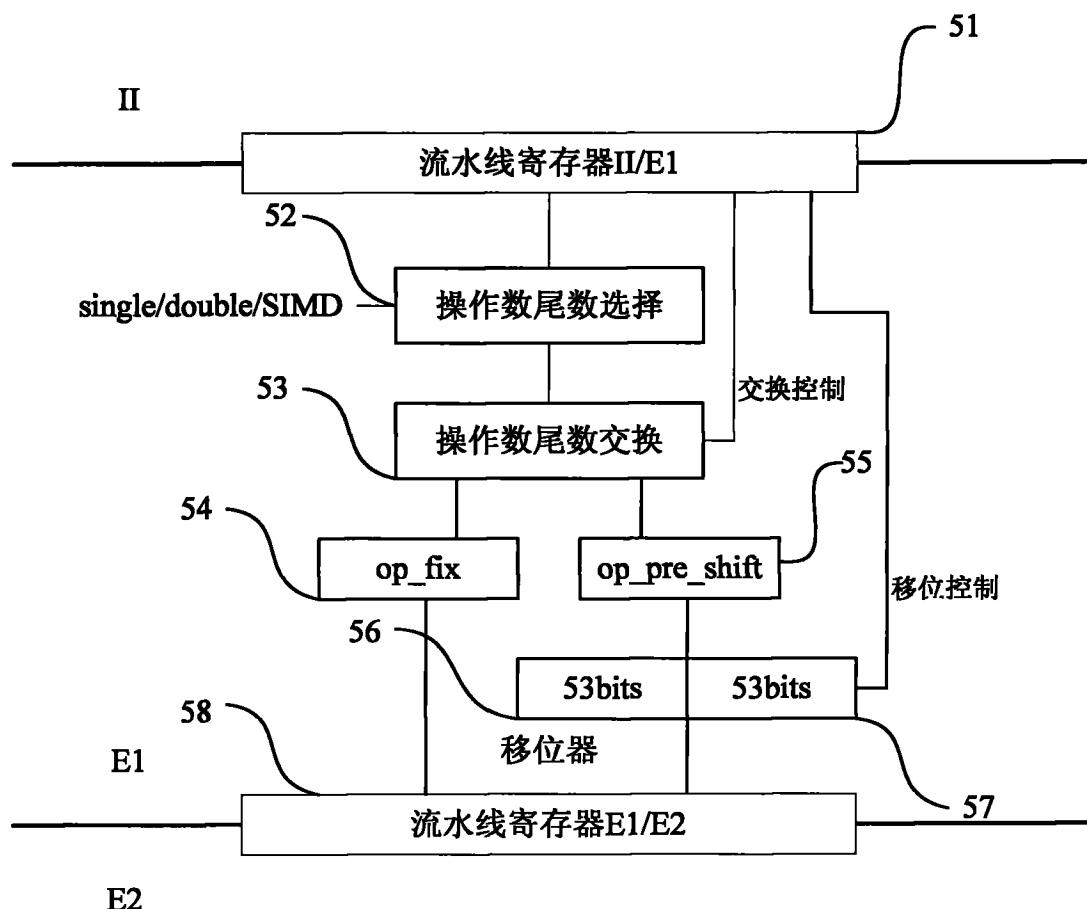


图 5

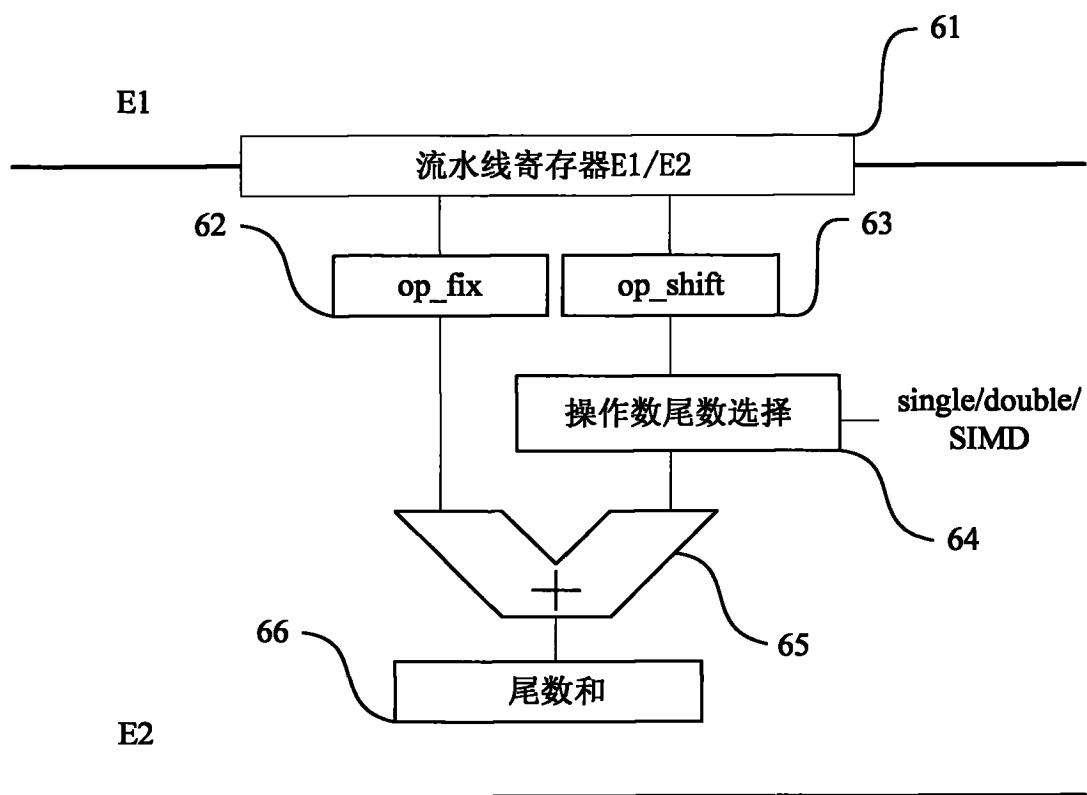


图 6

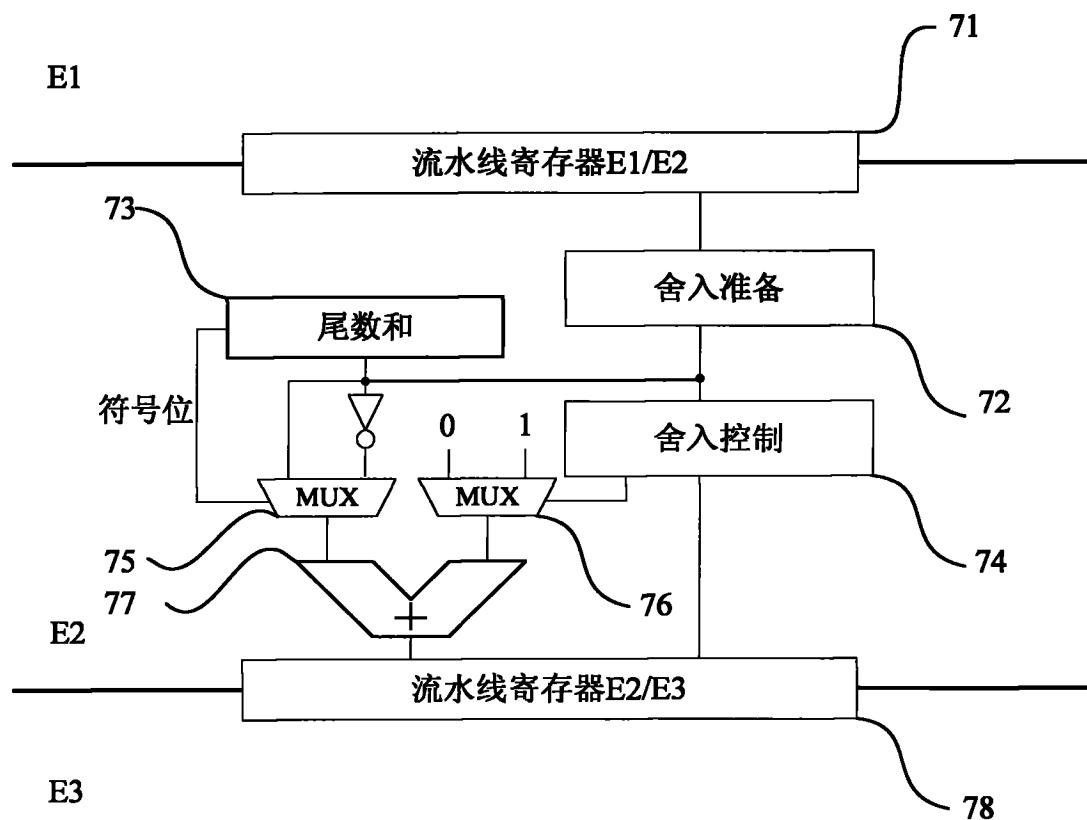


图 7

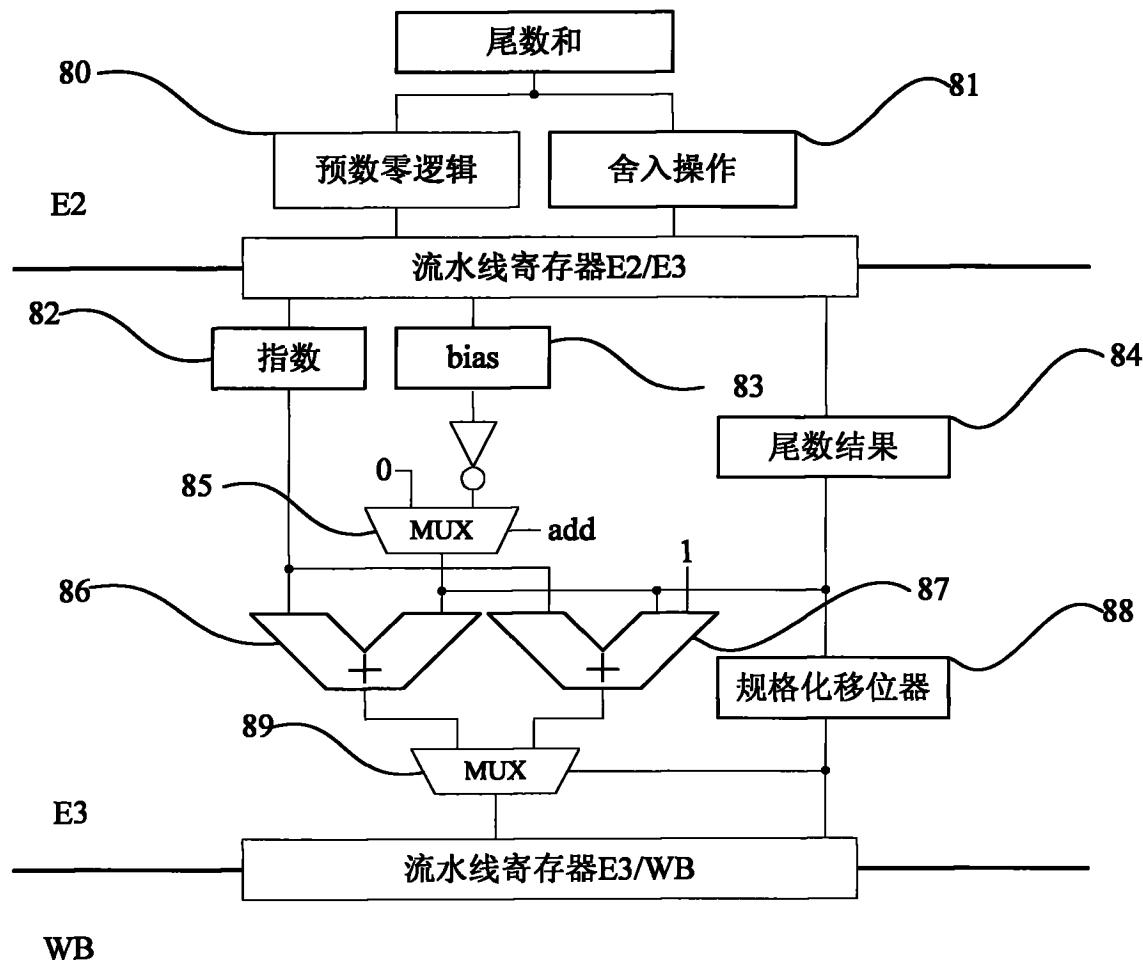


图 8

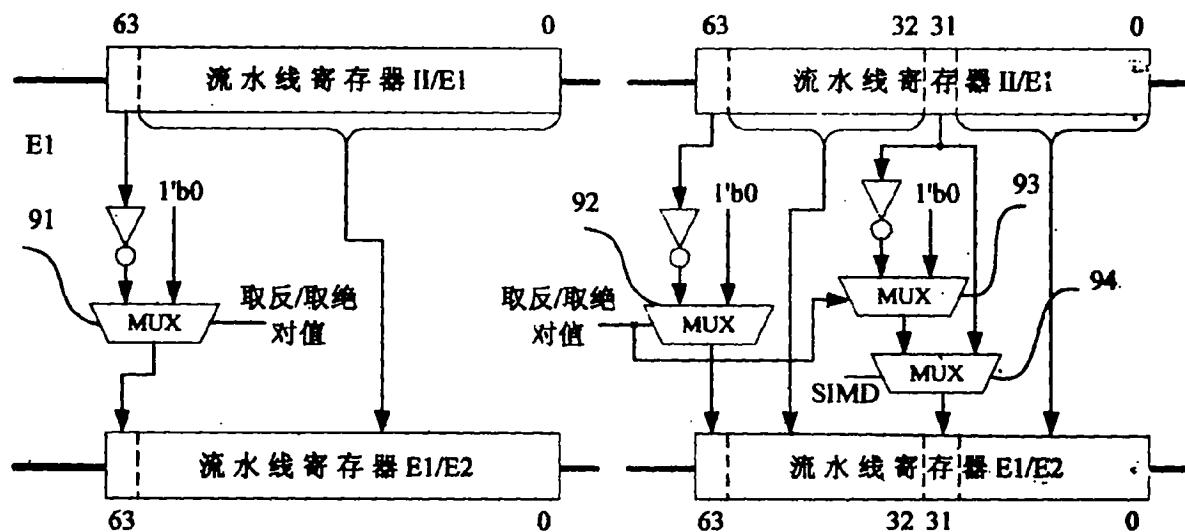


图 9