



(12) 发明专利

(10) 授权公告号 CN 108255777 B

(45) 授权公告日 2021.08.06

(21) 申请号 201810056827.2

G06F 7/483 (2006.01)

(22) 申请日 2018.01.19

(56) 对比文件

(65) 同一申请的已公布的文献号  
申请公布号 CN 108255777 A

- CN 1818851 A, 2006.08.16
- CN 102819520 A, 2012.12.12
- CN 104246690 A, 2014.12.24
- CN 101706712 A, 2010.05.12
- CN 101847087 A, 2010.09.29
- US 2017322769 A1, 2017.11.09
- CN 1831753 A, 2006.09.13

(43) 申请公布日 2018.07.06

(73) 专利权人 中国科学院电子学研究所  
地址 100190 北京市海淀区北四环西路19号

王楠, 黄志洪, 杨海钢, 丁健1. 一种支持高效加法的FPGA嵌入式DSP IP设计.《太赫兹科学与电子信息学报》.2017,

(72) 发明人 赵赫 杨海钢 黄志洪 魏星  
李小龙

(74) 专利代理机构 中科专利商标代理有限责任公司 11021

审查员 倪霞

代理人 任岩

(51) Int. Cl.

G06F 15/78 (2006.01)

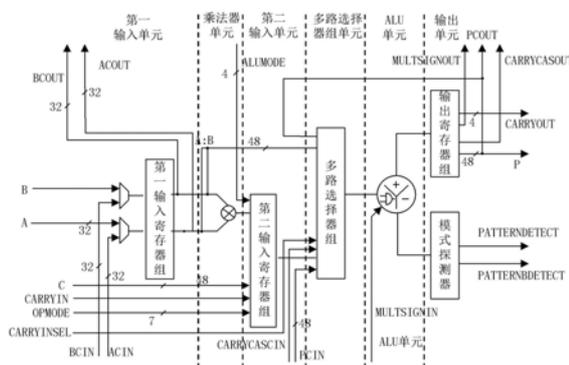
权利要求书2页 说明书7页 附图4页

(54) 发明名称

用于FPGA的嵌入式浮点型DSP硬核结构

(57) 摘要

本公开提供了一种用于FPGA的嵌入式浮点型DSP硬核结构,包括:第一输入单元,由输入寄存器组和浮点数乘法专用前加器构成,通过相对应的配置位,对输入数据进行输入寄存或者旁路选择;乘法器单元,连接到所述第一输入单元,接收前级经过寄存器的输入数据;第二输入单元,包括第二输入寄存器组,连接到乘法器单元的输出端;多路选择器组单元,输入端连接到所述第二输入单元的输出端、第一输入单元的输出端;ALU单元,包括加法器及逻辑运算单元,为浮点数和定点数提供加减以及乘法运算所使用,为定点数提供逻辑运算;输出单元。由于对数据的处理和运算都在该结构内部完成,运算效率要明显用软核的方式实现浮点数运算。



1. 一种用于FPGA的嵌入式浮点型DSP硬核结构,包括:

第一输入单元,由输入寄存器组和浮点数乘法专用前加器构成,通过相对应的配置位,对输入数据进行输入寄存或者旁路选择;

乘法器单元,连接到所述第一输入单元,接收前级经过寄存器的输入数据;

第二输入单元,包括第二输入寄存器组,连接到乘法器单元的输出端;

多路选择器组单元,由多个选择器组成,输入端连接到所述第二输入单元的输出端、第一输入单元的输出端;

ALU单元,包括加法器及逻辑运算单元,所述加法器为浮点数和定点数提供加减以及乘法运算所使用,同时也为定点数提供逻辑运算;

其中,ALU单元还包括调整电路、舍入单元、编码模块、探测树模块、初步移位模块、移位修正模块,其中所述调整电路包含前导零探测电路及一位误差调整电路;以及

输出单元,用于输出运算结果。

2. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,在进行浮点数乘法运算时,第一输入单元的前加器单元用于对输入的浮点数的指数部分进行求和,乘法器单元用于对尾数部分进行乘法操作,ALU单元用于完成浮点数的调整、规格化以及舍入操作。

3. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,在进行浮点数加减法运算时,在ALU单元中,输入的两个浮点数会分别送入两条路径,一路信号利用加法器对两个浮点数进行加减操作,得到的结果通过前导零探测单元探测尾数部分结果中0的个数,进行一个初步的移位和指数的调整,另一路信号被编码后送入探测树结构中,最终产生信号,指示是否需要初步移位的信号进行进一步的调整,最终得到浮点数加减法运算的结果。

4. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,所述输出单元包括:

输出寄存器组,输出寄存器组为前级ALU单元中的加法器单元提供了寄存单元,计算后的结果被寄存到加法器中,将结果用于累加的操作中;

模式探测器,所述模式探测器为可配置模块,用户通过配置模式探测器中的模式用以检测输出结果是否与模式相符,从而使DSP输出用户需要的特定数据。

5. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,

乘法器单元通过booth编码的方式对操作数进行乘法操作,并压缩部分积的个数,同时结合乘法器单元的树形加法器进一步对部分积进行压缩,并结合ALU单元中的前导零探测电路对所得的结果进行进一步的修正。

6. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,

所述乘法器单元在设计的过程中引入流水线的结构。

7. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,

第二输入单元还连接到包括:多路选择器组选择信号OPMODE、进位信号CARRYIN、端口C的数据输入端及ALU运算模式的配置信号ALUMODE。

8. 根据权利要求1所述的嵌入式浮点型DSP硬核结构,

多路选择器组单元的输入端还连接到包括DSP结果的级联信号PCIN,用于选择进位输入来源的CARRYINSEL信号,以及输出反馈信号PCOUT,并由相应的选通信号OPMODE对多路选择器组单元中的选择器进行选择,切换不同功能和/或改变输入到下一级加法器中的数据来源。

9. 根据权利要求1所述的嵌入式浮点型DSP硬核结构，  
第一输入单元和/或输出单元预留出DSP级联时所使用的端口。

## 用于FPGA的嵌入式浮点型DSP硬核结构

### 技术领域

[0001] 本公开涉及FPGA领域,尤其涉及一种用于FPGA的嵌入式浮点型DSP硬核结构。

### 背景技术

[0002] FPGA凭借自身可编程、并行度高、灵活性好等优势在通信、航天、军事等领域得到了广泛的应用。数字信号处理就是一个重要的应用领域,目前工业界主流的FPGA产品中基本都集成了可编程数字信号处理模块。例如,Xilinx公司的Virtex-7中含有3600个DSP48E1单元,支持乘加/乘减/乘累加等操作,Altera公司的Stratix-V含有532个DSP单元,单个的DSP IP核可根据应用的需求进行拆分以最小的资源实现最多的功能,支持乘加、乘减、乘累加等操作,但不支持加法、累加运算。FPGA在进行数字信号处理的过程中往往需要调用众多的DSP模块来对信号进行各种数学运算,但是随着数据量的不断增加,所需要处理的信号也由原来的定点数表示变为数值范围更大的浮点数表示,例如雷达信号、导航等都是采用浮点数的表示方式。

[0003] 在实际生活中,浮点数有着广泛的应用空间,例如,雷达信号就是采用浮点数的表示方法,收集到的雷达信号会被以浮点数的形式送入到计算机中进行信号的处理,正是由于这样的应用需求,Altera和Xilinx公司的FPGA产品都提供了浮点数运算的相关IP软核。以Xilinx公司为例,该公司开发了相关的浮点数运算模块IP软核,通过Vivado中的IP Catalog功能进行IP调用,能够支持包括多种浮点数运算,除浮点数的基本运算之外,还提供指数、对数、开方等运算,所提供的具体功能如下表1所示。采用硬件描述语言进行算法建模,产生硬核电路结构的方法是可行的,但过于繁琐,且开发周期长,所以FPGA厂商大多都是采用了一种传统的浮点数运算实现方式,即采用IP软核的方式,通过逻辑资源或DSP的方式实现相关的操作。

[0004] 表1 Xilinx浮点IP所支持的运算

运算类型	数学表达式
加/减运算	$a \pm b$
累加运算	$\sum_{i=0}^{n-1} a_i$
乘法运算	$a \times b$
乘加运算	$a \times b + c$
除法运算	$a \div b$
倒数运算	$1/x$
开方运算	$\sqrt{x}$
倒数开方运算	$1/\sqrt{x}$
绝对值运算	$ x $
自然对数运算	$\ln(x)$
指数运算	$e^x$

[0006] 目前的FPGA产品中的DSP模块都是采用定点型的DSP结构,在Altera的EDA软件Quartus II和Xilinx的EDA软件Vivado中将浮点数运算中的逻辑控制部分映射到FPGA的LUT表等逻辑资源中,将进行浮点数运算的乘法、加法等操作映射到DSP的定点乘法器、加法器中。这种方法虽然方便,但是占用的资源过多,且IP软核的运算效率不高。由于对浮点数高速运算的应用需求,Intel公司在其最新的产品中嵌入了浮点型硬核DSP模块来提高FPGA对浮点数计算的支持,但是目前为止未有芯片提供。

[0007] 针对上述问题,本公开提出了一种硬核的浮点型DSP结构,以提高浮点数的运算效率,同时减少FPGA中逻辑资源的使用。

[0008] 公开内容

[0009] (一)要解决的技术问题

[0010] 本公开提供了一种用于FPGA的嵌入式浮点型DSP硬核结构,以至少部分解决以上所提出的技术问题。

[0011] (二)技术方案

[0012] 根据本公开的一个方面,提供了一种用于FPGA的嵌入式浮点型DSP硬核结构,包括:第一输入单元,由输入寄存器组和浮点数乘法专用前加器构成,通过相对应的配置位,对输入数据进行输入寄存或者旁路选择;乘法器单元,连接到所述第一输入单元,接收前级经过寄存器的输入数据;第二输入单元,包括第二输入寄存器组,连接到乘法器单元的输出端;多路选择器组单元,由多个选择器组成,输入端连接到所述第二输入单元的输出端、第一输入单元的输出端;ALU单元,包括加法器及逻辑运算单元,所述加法器为浮点数和定点数提供加减以及乘法运算所使用,同时也为定点数提供逻辑运算;以及输出单元,用于输出运算结果。

[0013] 在本公开一些实施例中,其中,ALU单元还包括调整电路、舍入单元、编码模块、探测树模块、初步移位模块、移位修正模块,其中所述调整电路包含前导零探测电路及一位误

差调整电路。

[0014] 在本公开一些实施例中,在进行浮点数乘法运算时,第一输入单元的前加器单元用于对输入的浮点数的指数部分进行求和,乘法器单元用于对尾数部分进行乘法操作,ALU单元用于完成浮点数的调整、规格化以及舍入操作。

[0015] 在本公开一些实施例中,在进行浮点数加减法运算时,在ALU单元中,输入的两个浮点数会分别送入两条路径,一路信号利用加法器对两个浮点数进行加减操作,得到的结果通过前导零探测单元探测尾数部分结果中0的个数,进行一个初步的移位和指数的调整,另一路信号被编码后送入探测树结构中,最终产生信号,指示是否需要初步移位的信号进行进一步的调整,最终得到浮点数加减法运算的结果。

[0016] 在本公开一些实施例中,所述输出单元包括:输出寄存器组,输出寄存器组为前级ALU单元中的加法器单元提供了寄存单元,计算后的结果被寄存到加法器中,将结果用于累加的操作中;模式探测器,所述模式探测器为可配置模块,用户通过配置模式探测器中的模式用以检测输出结果是否与模式相符,从而使DSP输出用户需要的特定数据。

[0017] 在本公开一些实施例中,乘法器单元通过booth编码的方式对操作数进行乘法操作,并压缩部分积的个数,同时结合乘法器单元的树形加法器进一步对部分积进行压缩,并结合ALU单元中的前导零探测电路对所得的结果进行进一步的修正。

[0018] 在本公开一些实施例中,所述乘法器单元在设计的过程中引入流水线的结构。

[0019] 在本公开一些实施例中,第二输入单元还连接到包括:多路选择器组选择信号OPMODE、进位信号CARRYIN、端口C的数据输入端及ALU运算模式的配置信号ALUMODE。

[0020] 在本公开一些实施例中,多路选择器组单元的输入端还连接到包括DSP结果的级联信号PCIN,用于选择进位输入来源的CARRYINSEL信号,以及输出反馈信号PCOUT,并由相应的选通信号OPMODE对多路选择器组单元中的选择器进行选择,切换不同功能和/或改变输入到下一级加法器中的数据来源。

[0021] 在本公开一些实施例中,第一输入单元和/或输出单元预留出DSP级联时所使用的端口。

[0022] (三)有益效果

[0023] 从上述技术方案可以看出,本公开用于FPGA的嵌入式浮点型DSP硬核结构至少具有以下有益效果其中之一:

[0024] (1) 由于对数据的处理和运算都在该结构内部完成,相比于利用FPGA中的逻辑资源映射浮点数调整等电路结构来说运算更加高效,运算效率要明显优于xilinx 7系列用软核的方式实现浮点数运算;

[0025] (2) 通过使用专用的浮点型硬核DSP结构进行浮点数运算,可以减少FPGA中逻辑资源的消耗。

[0026] (3) 相比于浮点数运算的软核实现方式,在相同的浮点数操作的前提下,浮点型DSP硬核结构具有更低的功耗。

## 附图说明

[0027] 图1为本公开实施例用于FPGA的嵌入式浮点型DSP硬核结构的示意图。

[0028] 图2为本公开实施例用于FPGA的嵌入式浮点型DSP硬核实现浮点数乘法运算的原

理图。

[0029] 图3为本公开实施例浮点数乘法的DSP结构实现结构图。

[0030] 图4为本公开实施例尾数乘法的补位操作示意图。

[0031] 图5为本公开实施例用于FPGA的嵌入式浮点型DSP硬核实现浮点数加减法运算的原理图。

[0032] 图6本公开实施例浮点数加法的DSP结构实现结构图。

### 具体实施方式

[0033] 本公开提供了一种用于FPGA的嵌入式浮点型DSP硬核结构。为使本公开的目的、技术方案和优点更加清楚明白,以下结合具体实施例,并参照附图,对本公开进一步详细说明。

[0034] 本公开某些实施例于后方将参照所附附图做更全面性地描述,其中一些但并非全部的实施例将被示出。实际上,本公开的各种实施例可以许多不同形式实现,而不应被解释为限于此数所阐述的实施例;相对地,提供这些实施例使得本公开满足适用的法律要求。

[0035] 在本公开的第一个示例性实施例中,提供了一种用于FPGA的嵌入式浮点型DSP硬核结构。图1为本公开第一实施例用于FPGA的嵌入式浮点型DSP硬核结构的结构示意图。如图1所示,本公开用于FPGA的嵌入式浮点型DSP硬核结构包括:第一输入单元、乘法器单元、第二输入单元、多路选择器组单元、ALU单元、输出单元。

[0036] 以下分别对本实施例用于FPGA的嵌入式浮点型DSP硬核结构的各个组成部分进行详细描述。

[0037] 第一输入单元主要由输入寄存器组和浮点数乘法专用前加器构成,通过相对应的配置位,对输入数据A、B进行输入寄存或者旁路选择;在执行浮点数乘法的时候前加器会对浮点数的指数相加,得到的结果送到后一级进行进一步操作。该单元中引入多级流水线,对关键路径进行分割,缩短关键路径,保证了DSP的工作频率。同时,预留出DSP级联时所使用的端口,可通过级联的方式完成更多的操作,提高DSP的灵活性。

[0038] 乘法器单元连接到所述第一输入单元,用于接收前级经过寄存器的输入数据A和B。本实施例中乘法器单元为一个二输入的乘法器单元(A\*B),通过booth编码的方式对操作数进行乘法操作,并压缩部分积的个数,同时结合树形的加法器进一步对部分积进行压缩,通过上述方式减少面积开销,提高计算速度,该单元同时为浮点数和定点数的乘法提供器件支持,并结合ALU单元中的前导零探测电路(LZD)对所得的结果进行进一步的修正。优选地,所述乘法器单元在设计的过程中可以引入流水线的结构,提高工作频率。

[0039] 第二输入单元包括第二输入寄存器组,连接到乘法器单元的输出端,还包括多路选择器组选择信号OPMODE、进位信号CARRYIN、端口C的数据输入端及ALU运算模式的配置信号ALUMODE,该输入单元用于对输入进行输入寄存,同时分割乘法器到ALU单元之间的关键路径,提高工作频率。

[0040] 多路选择器组单元的输入端连接到所述第二输入单元的输出端、第一输入单元的输出端,同时包括DSP结果的级联信号PCIN,用于选择进位输入来源的CARRYINSEL信号,以及本DSP模块的输出反馈信号PCOUT,多路选择器组单元由多个选择器组成,由相应的选通信号OPMODE对选择器进行选择,从而完成不同功能之间的切换,也能改变输入到下一级加

法器中的数据来源。

[0041] ALU单元的主要单元是为浮点数和定点数提供加减以及乘法运算所使用的加法器,同时也为定点数提供逻辑运算。其中,该ALU单元为了能够支持浮点数的运算,对其结构进行了改进,除了运算结构以外,还加入了前导零探测电路(LZD)、一位误差调整电路以及舍入单元,从而使其在完成浮点数的运算的同时,将输出结果规格化从而产生符合标准的浮点数输出。

[0042] 输出单元包括输出寄存器组和模式探测器,所述输出单元通过输出寄存器组引入流水线结构,从而提高DSP的工作频率,同时寄存器为前级加法器单元提供了寄存单元,计算后的结果可以被寄存在加法器中,这样可以将结果用于累加的操作中。同时,输出单元预留了级联的端口,这样就可以通过级联DSP完成更多的操作。模式探测器是用户可配置的模块,用户通过配置模式探测器中的模式用以检测输出结果是否与模式相符,从而使DSP输出用户需要的特定数据。

[0043] 实现浮点数乘法的过程中,操作如图2所示。为了实现上述的计算过程,本结构在第一输入单元中加入前加器单元,对输入的浮点数A和B的指数部分进行求和,乘法器单元对尾数部分进行乘法操作,浮点数的调整、规格化以及舍入操作则交由ALU单元完成。算法的DSP结构实现框图如图3所示,其中,前加器在第一输入单元中,乘法器后面的加法器以及调整电路都在ALU单元中。前加器作为一个专用的指数加法器会对输入数据的32bit数据中的指数部分,即24位至31位的指数部分进行前加操作,得到的结果作为浮点数乘法的指数部分。

[0044] 根据IEEE-754协议的规定,任何规范浮点数的尾数部分为23bit,在这23bit之前还有一个数值为1的隐藏位不在浮点数中表示出来,但是在参与浮点数运算的时候这一位隐藏位需要补出来参与到运算中,为了兼顾定点数25\*25bit的乘法操作,还需要在隐藏位前补0,做为尾数的符号位,使其满足位宽需求,如图4所示。对尾数进行补0的操作是将浮点数全部看做是正数进行运算,但是这并不是结果的真正符号位,因此为了得到真正的符号位,就需要将输入的两个浮点乘数的符号取出,单独进行逻辑异或运算,从而得到结果真正的符号位。然后将尾数位进行乘法运算即可得到尾数部分的乘法结果。

[0045] 在得到指数部分和尾数部分的值之后,需要对指数部分和尾数部分进行调整,对相关的浮点数进行规格化处理。在进行规格化处理的过程中需要用到前导零探测电路(LZD)来探测尾数部分结果中0的个数,从而对尾数部分进行移位操作。在获得了前导0的个数之后,对尾数结果进行相应的移位操作,同时将指数的值减掉前导0的个数,再将最终的符号位、8bit指数部分和23bit的尾数部分组合起来即为最终浮点数乘法的结果。

[0046] 本公开实施例用于FPGA的嵌入式浮点型DSP硬核实现浮点数加减法运算的原理如图5所示,因此本公开采用了图6所示的结构对浮点数进行运算从而得到浮点数加减法的结果。两个浮点数输入到ALU单元中,会分别送入两条路径,一路信号利用加法器对两个浮点数进行加减操作,得到的结果通过LZD单元进行一个初步的移位和指数的调整,另一路信号被编码后送入探测树结构中,最终产生信号,指示是否需要进一步对初步移位的信号进行调整,最终得到浮点数加减法运算的结果。上述的操作都在ALU单元中完成。

[0047] 在进行实际的运算时,同符号数的加法是不需要通过编码和探测树单元,因为这种运算条件下结果不会产生误差,但是异号的加法或者同号的减法是需要进行结果的调整

的。在进行异号加法或者同号减法的过程中存在1bit误差。通过对输入的两个数据进行编码后,将其进行树形探测,从而最终确定是否需要进行一位误差的调整。

[0048] 本公开中提出的浮点型硬核DSP结构具有浮点数加法、乘法、以及累加的功能,采用标准中芯国际28nm CMOS工艺库,电压为0.945V,温度125摄氏度,利用DC工具完成浮点型硬核DSP的电路实现,通过布局布线得到的整体面积为11091 $\mu\text{m}^2$ 。

[0049] 对比同样工艺节点的xilinx 7系列FPGA,型号为xc7v585tffg1157-3,在相同的latency的情况下(本公开中采用的是6级流水线结构),调用浮点数加减法单元,选择速度优先的结构优化,该单元被映射到1个DSP48E1和逻辑资源上,分别设置性能和资源作为优化目标和本公开的结构进行比较,同时调用了浮点数乘法单元,比较结果如下表所示,其浮点数加法、乘法的对比情况如下表所示:

[0050] 表2 综合结果对比表

		本公开结构	Xilinx 7
[0051]	浮点数加法		
	最高工作频率	400MHz	220MHz
	功率	1.12mW	0.267W
	浮点数性能	0.4Gflops	0.22 Gflops
	浮点数乘法		
	最高工作频率	416MHz	290MHz
	功率	1.65mW	0.284W
	浮点数性能	0.416Gflops	0.29 Gflops

[0052] 通过比较可以看出,在执行浮点数加法的过程中,本公开所提出的浮点型硬核DSP结构的浮点数加法运算性能是性能优先条件下IP软核的1.8倍;浮点数乘法运算性能是性能优先条件下IP软核的1.43倍。这是由于IP软核实现的过程中需要片上逻辑资源的参与,数据由片上逻辑资源处理后送入DSP运算和DSP运算后的结果再送出到片上逻辑资源再处理这两个过程需要花费较长的时间,而本公开所提出的浮点型硬核DSP结构对数据的处理和运算都在该结构内部完成,其次利用FPGA中的逻辑资源映射浮点数调整等电路结构不如专用的电路设计高效。本公开所提出的浮点型硬核DSP结构的浮点数运算效率要明显优于xilinx 7系列用软核的方式实现浮点数运算。

[0053] 当然,上述硬件结构还应当包含电源模块(图未示)等功能模块,这些是本领域内的一般技术人员可以理解的,本领域内一般技术人员也可以根据功能的需要,添加相应的功能模块,在此不作赘述。

[0054] 至此,本公开第一实施例浮点型硬核DSP结构介绍完毕。

[0055] 至此,已经结合附图对本公开实施例进行了详细描述。需要说明的是,在附图或说明书正文中,未绘示或描述的实现方式,均为所属技术领域中普通技术人员所知的形式,并未进行详细说明。此外,上述对各元件和方法的定义并不仅限于实施例中提到的各种具体结构、形状或方式,本领域普通技术人员可对其进行简单地更改或替换。

[0056] 除非有所知名为相反之意,本说明书及所附权利要求中的数值参数是近似值,能够根据通过本公开的内容所得的所需特性改变。具体而言,所有使用于说明书及权利要求

中表示组成的含量、反应条件等等的数字,应理解为在所有情况中是受到「约」的用语所修饰。一般情况下,其表达的含义是指包含由特定数量在一些实施例中 $\pm 10\%$ 的变化、在一些实施例中 $\pm 5\%$ 的变化、在一些实施例中 $\pm 1\%$ 的变化、在一些实施例中 $\pm 0.5\%$ 的变化。

[0057] 再者,单词“包含”不排除存在未列在权利要求中的元件或步骤。位于元件之前的单词“一”或“一个”不排除存在多个这样的元件。

[0058] 说明书与权利要求中所使用的序数例如“第一”、“第二”、“第三”等的用词,以修饰相应的元件,其本身并不意味着该元件有任何的序数,也不代表某一元件与另一元件的顺序、或是制造方法上的顺序,这些序数的使用仅用来使具有某命名的一元件得以和另一具有相同命名的元件能做出清楚区分。

[0059] 此外,除非特别描述或必须依序发生的步骤,上述步骤的顺序并无限制于以上所列,且可根据所需设计而变化或重新安排。并且上述实施例可基于设计及可靠度的考虑,彼此混合搭配使用或与其他实施例混合搭配使用,即不同实施例中的技术特征可以自由组合形成更多的实施例。

[0060] 本领域那些技术人员可以理解,可以对实施例中的设备中的模块进行自适应性地改变并且把它们设置在与该实施例不同的一个或多个设备中。可以把实施例中的模块或单元或组件组合成一个模块或单元或组件,以及此外可以把它们分成多个子模块或子单元或子组件。除了这样的特征和/或过程或者单元中的至少一些是相互排斥之外,可以采用任何组合对本说明书(包括伴随的权利要求、摘要和附图)中公开的所有特征以及如此公开的任何方法或者设备的所有过程或单元进行组合。除非另外明确陈述,本说明书(包括伴随的权利要求、摘要和附图)中公开的每个特征可以由提供相同、等同或相似目的的替代特征来代替。并且,在列举了若干装置的单元权利要求中,这些装置中的若干个可以是通过同一个硬件项来具体体现。

[0061] 类似地,应当理解,为了精简本公开并帮助理解各个公开方面中的一个或多个,在上面对本公开的示例性实施例的描述中,本公开的各个特征有时被一起分组到单个实施例、图、或者对其的描述中。然而,并不应将该公开的方法解释成反映如下意图:即所要求保护的本公开要求比在每个权利要求中所明确记载的特征更多的特征。更确切地说,如下面的权利要求书所反映的那样,公开方面在于少于前面公开的单个实施例的所有特征。因此,遵循具体实施方式的权利要求书由此明确地并入该具体实施方式,其中每个权利要求本身都作为本公开的单独实施例。

[0062] 以上所述的具体实施例,对本公开的目的、技术方案和有益效果进行了进一步详细说明,所应理解的是,以上所述仅为本公开的具体实施例而已,并不用于限制本公开,凡在本公开的精神和原则之内,所做的任何修改、等同替换、改进等,均应包含在本公开的保护范围之内。

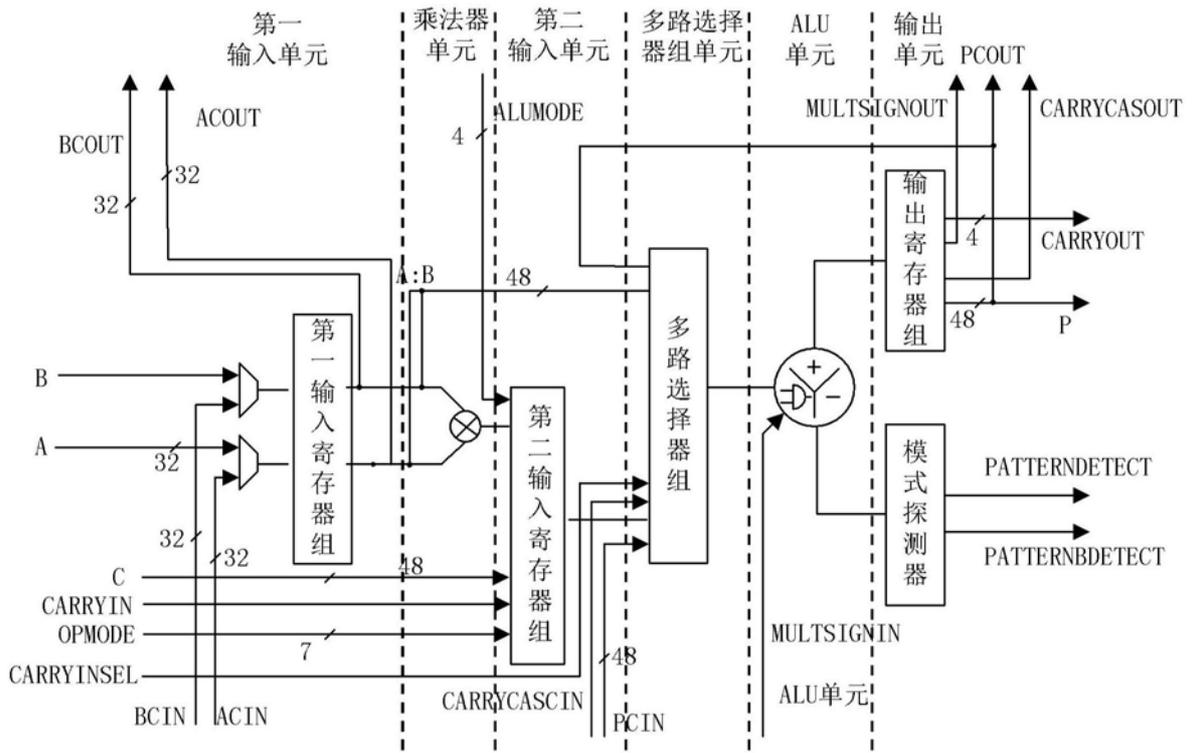


图1

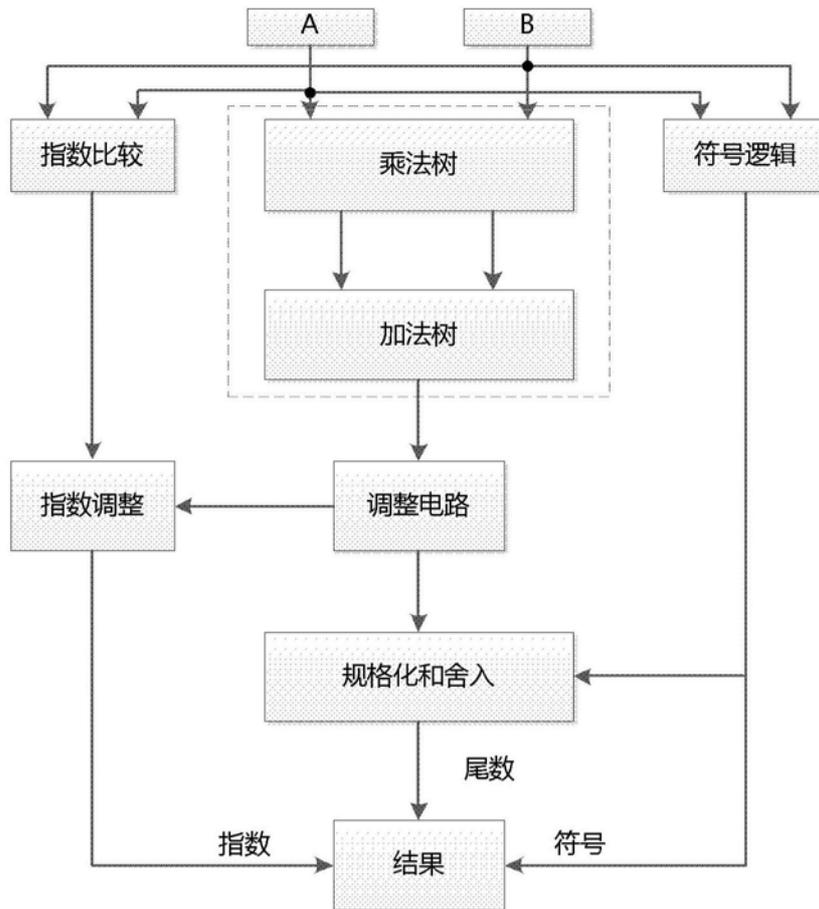


图2

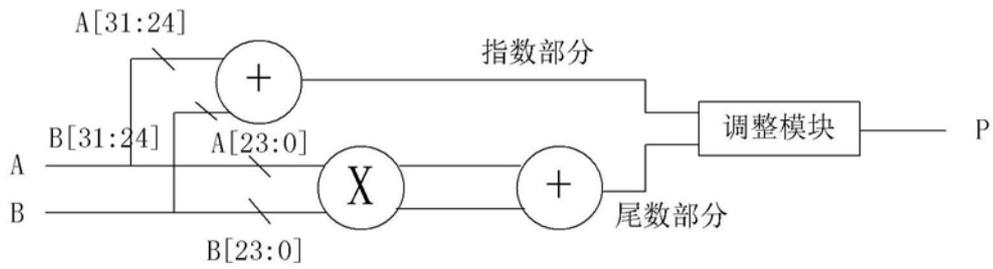


图3

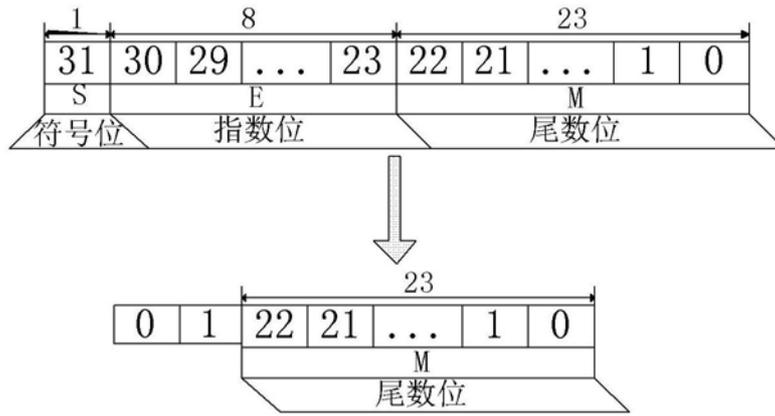


图4

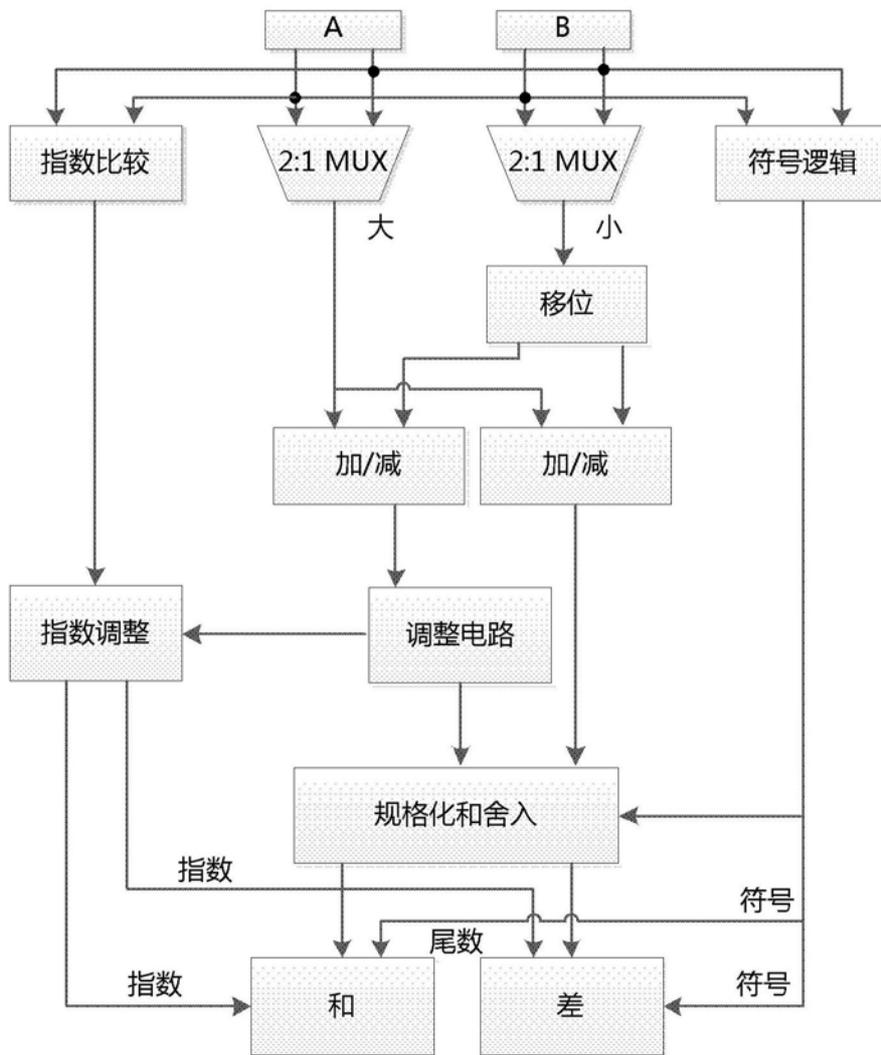


图5

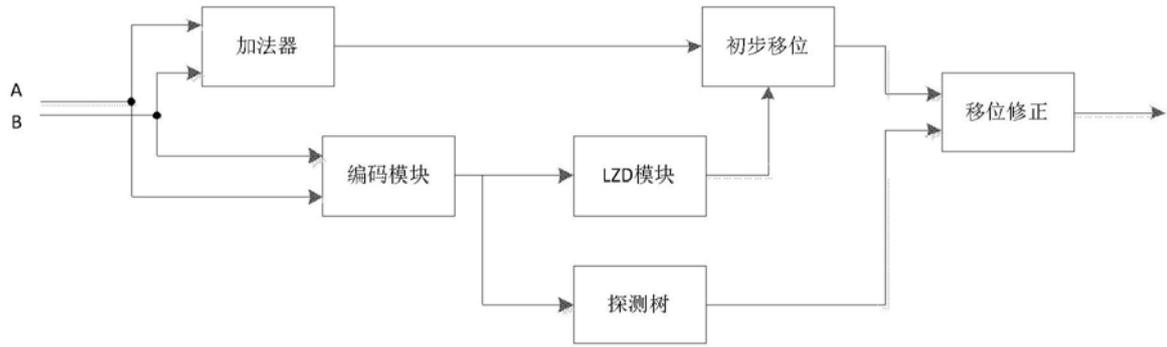


图6