

浮点数据精度转换方法和装置

本申请要求于 2023 年 03 月 03 日提交国家知识产权局、申请号为 202310238789.3、申请名称为“浮点数据精度转换方法和装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请实施例涉及芯片技术领域，尤其涉及一种浮点数据精度转换方法和装置。

背景技术

随着人工智能（artificial intelligence, AI）领域的发展，神经网络（neural networks, NNS）的规模也越来越大，训练神经网络对算力的需求也越来越大。目前，模型训练算力的激增导致芯片成本开销较大，继续采用现有的浮点（floating point, FP）16 混合精度数据和脑浮点（brain floating point, BF）16 精度数据成本代价过高，而采用低比特数据进行计算可以完成变换器（transformer）大模型的无损训练，由此，低比特浮点训练是未来大模型训练的一个趋势。

但是，在神经网络训练过程中，针对混合精度计算场景，需要涉及到不同精度的浮点数据间的相互转换。低精度的浮点数据向高精度的浮点数据进行格式转换时，可以实现无误差的转换。高精度的浮点数据向低精度的浮点数据进行格式转换时，需要对高精度的浮点数据进行舍入（round）操作，由此会产生转换误差，从而影响训练模型的训练精度。

发明内容

本申请实施例提供一种浮点数据精度转换方法和装置，降低了高精度浮点数据转换为低精度浮点数据的转换误差的问题。

为达到上述目的，本申请实施例采用如下技术方案。

第一方面，本申请实施例提供一种浮点数据精度转换方法，第一浮点数据包括第一指数域和第一尾数域，第二浮点数据包括第二尾数域，第一浮点数据的精度高于第二浮点数据的精度，该方法包括：根据第一指数域的编码值确定第二尾数域的位宽；确定第一尾数域中的保留编码值和舍弃编码值，保留编码值包括第一尾数域中从最高位开始，且位宽与第二尾数域的位宽相同的编码值；若第一指数域的编码值大于或等于第一预设阈值，根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作，得到第二尾数域的编码值；若第一指数域的编码值小于第一预设阈值，根据舍弃编码值的最高位对保留编码值进行舍入操作，得到第二尾数域的编码值。

由此，本申请实施例提供的浮点数据精度转换方法，实现了将高精度的浮点数据转换为低精度的浮点数据。其中，根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作可以理解随机舍入方式，根据舍弃编码值的最高位对保留编码值进行舍入操作可以理解为远离 0 进位舍入方式。在数据格式转换中，对于随机舍入方式和远离 0 进位舍入方式的选择可以理解混合舍入方式。在第一指数域的编码值大于或等于第一预设阈值时，采用随机舍入方式，在第一指数域的编码值小于第一预设阈值时，采用远离 0 进位舍入方式。由此，通过提供不同的舍入方式，可以降低数据格式转换时的转换误差，提高数据转换的效率，提高训练模型的训练精度。

在一种可能的设计中，舍入操作包括进位操作和舍弃操作，根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作，得到第二尾数域的编码值，包括：舍弃编码值中从最高位开始，且位宽为预设位宽的编码值大于或等于第二预设阈值时，对保留编码值的最低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值；舍弃编码值中从最高位开始，且位宽为预设位宽的编码值小于第二预设阈值时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的编码值；其中，第二预设阈值为舍弃编码值中从最低位开始，且位宽为预设位宽的编码值。

这种设计中，对于随机舍入方式，用于比较的第二预设阈值为舍弃编码值中从最低为开始，且位宽为预设位宽的编码值，第二预设阈值的生成无需额外的随机数生成器，不存在随机数生成的性能瓶颈，提高了高精度的浮点数据向低精度的浮点数据的转换效率，同时硬件开销更低。

在一种可能的设计中，舍入操作包括进位操作和舍弃操作，根据舍弃编码值的最高位对保留编码值进行舍入操作，得到第二尾数域的编码值，包括：舍弃编码值的最高位大于或等于第三预设阈值时，对保留编码值的最低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编

作，保留编码值为第二尾数域的编码值。

这种设计中，第三预设阈值可以是 0 或 1，将舍弃编码值的最高位和第三预设阈值进行比较，属于远离 0 进位舍入方式。除了远离 0 进位舍入方式，还可以包括远离偶数进位舍入方式和远离奇数进位舍入方式等。但远离 0 进位舍入方式对于其他舍入方式硬件实现面积更小、功耗开销更小，且具有更高的数据分辨率。

在一种可能的设计中，第一预设阈值由遍历多个第一浮点数据确定。

这种设计中，第一预设阈值为可调参数，通过设置合适的第一预设阈值的值可以提高数据格式转换的准确率。

在一种可能的设计中，第一浮点数据还包括符号域，第二浮点数据还包括符号域、前缀码域和第二指数域，前缀码域用于指示第二指数域的位宽，根据第一指数域的编码值确定第二尾数域的位宽之前，方法还包括：根据第一指数域的编码值确定前缀码域的位宽、前缀码域的编码值、第二指数域的位宽和第二指数域的编码值。

这种设计中，在数据格式转换中，基于第一浮点数据的符号域可以得到第二浮点数据的符号域，基于第一浮点数据的第一指数域可以得到第二浮点数据的前缀码域和第二指数域，以及基于第一浮点数据的第一尾数域可以得到第二浮点数据的第二尾数域。第二浮点数据中通过较短的前缀码域指示第二指数域的位宽，可以有效提升第二浮点数据的尾数域的精度或位宽，同时对于只提供 1 位尾数的精度的第二浮点数据可以表示较大的数值范围，有效平衡了第二浮点数据位宽、范围和精度之间的关系。且前缀码域可以采用前缀码编码方式，占用位宽少，解析第二指数域和第二尾数域便捷。

第二方面，本申请实施例提供一种浮点数据精度转换装置，第一浮点数据包括第一指数域和第一尾数域，第二浮点数据包括第二尾数域，第一浮点数据的精度高于第二浮点数据的精度，装置包括：位宽计算单元，用于根据第一指数域的编码值确定第二尾数域的位宽；尾数域计算单元，用于确定第一尾数域中的保留编码值和舍弃编码值，保留编码值包括第一尾数域中从最高位开始，且位宽与第二尾数域的位宽相同的编码值；舍入操作单元，用于若第一指数域的编码值大于或等于第一预设阈值，根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作，得到第二尾数域的编码值；舍入操作单元，还用于若第一指数域的编码值小于第一预设阈值，根据舍弃编码值的最高位对保留编码值进行舍入操作，得到第二尾数域的编码值。

第二方面的有益效果可参见第一方面的说明。

在一种可能的设计中，舍入操作包括进位操作和舍弃操作，舍入操作单元还用于：舍弃编码值中从最高位开始，且位宽为预设位宽的编码值大于或等于第二预设阈值时，对保留编码值的最低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值；舍弃编码值中从最高位开始，且位宽为预设位宽的编码值小于第二预设阈值时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的编码值；其中，第二预设阈值为舍弃编码值中从最低位开始，且位宽为预设位宽的编码值。

在一种可能的设计中，舍入操作包括进位操作和舍弃操作，舍入操作单元还用于：舍弃编码值的最高位大于或等于第三预设阈值时，对保留编码值的最低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值；舍弃编码值的最高位小于第三预设阈值时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的编码值。

在一种可能的设计中，第一预设阈值由遍历多个第一浮点数据确定。

在一种可能的设计中，第一浮点数据还包括符号域，第二浮点数据还包括符号域、前缀码域和第二指数域，前缀码域用于指示第二指数域的位宽，位宽计算单元还用于：根据第一指数域的编码值确定前缀码域的位宽、前缀码域的编码值、第二指数域的位宽和第二指数域的编码值。

第三方面，本申请实施例提供了一种浮点数据精度转换装置，包括处理器核存储器，存储器存储计算机指令，处理器执行计算机指令后执行上述任一方面及任一项可能的实现方式中的浮点数据精度转换方法。

第四方面，本申请实施例提供了一种计算机可读存储介质，包括计算机指令，当计算机指令在电子设备上运行时，使得电子设备执行上述任一方面及任一项可能的实现方式中的浮点数据精度转换方法。

第五方面，本申请实施例提供了一种计算机程序产品，当计算机程序产品在计算机或处理器上

法。

第六方面，本申请实施例提供了一种系统，该系统可以包括以上任一方面的任一项可能的实现方式中的无线接入设备和至少一个电子设备。该电子设备和无线接入设备可以执行上述任一方面及任一项可能的实现方式中的浮点数据精度转换方法。

可以理解的是，上述提供的任一种浮点数据精度转换装置、计算机可读存储介质或计算机程序产品等均可以应用于上文所提供的对应的方法，因此，其所能达到的有益效果可参考对应的方法中的有益效果，此处不再赘述。

本申请的这些方面或其他方面在以下的描述中会更加简明易懂。

附图说明

图1为本申请实施例提供的一种数据结构示意图；

图2为本申请实施例提供的一种指数域-尾数域位宽的分布示意图；

图3为本申请实施例提供的一种浮点数据精度转换装置应用的系统或设备示意图；

图4为本申请实施例提供的一种SoC的结构示意图；

图5为本申请实施例提供的一种浮点数据精度转换方法的流程图；

图6为本申请实施例提供的一种基于SR舍入方式的FP32数据转换为HiFloat8数据的尾数域的示意图；

图7为本申请实施例提供的一种基于SR舍入方式的FP16数据转换为HiFloat8数据的尾数域的示意图；

图8为本申请实施例提供的一种混合舍入方式的分布示意图；

图9为本申请实施例提供的一种神经网络训练流程图；

图10为本申请实施例提供的另一种神经网络训练流程图；

图11为本申请实施例提供的另一种神经网络训练流程图；

图12为本申请实施例提供的一种浮点数据精度转换装置的组成示意图。

具体实施方式

为了便于理解，示例的给出了部分与本申请实施例相关概念的说明以供参考。如下所示：

神经网络训练的过程主要包括前向(forward)计算、反向(backward)计算和权重(weight)更新，训练过程中的计算模式包括通用矩阵乘法(general matrix multiplication, GEMM)和非通用矩阵乘法，通用矩阵乘法计算对应的层可以称为矩阵乘计算层，非通用矩阵乘法计算对应的层可以称为非矩阵乘计算层。具体的，通用矩阵乘法包括卷积(convolution)、反卷积(transposed convolution)、矩阵乘法(matmul)和批量矩阵乘法(batch matmul)，非通用矩阵乘法包括激活函数、归一化函数、优化器梯度更新计算等，激活函数包括sigmoid、tanh和relu，归一化函数包括批标准化(batch normalization)、层标准化(layer normalization)和实例标准化(instance normalization)等。

其中，通用矩阵乘法可以采用FP16、BF16或者更低精度的数据进行计算，非通用矩阵乘法需要采用FP32等较高精度的数据进行计算，FP16也可以称为半精度浮点数(half-precision floating-point)，BF16也可以称为16位脑浮点数(brain floating point)，FP32也可以称为单精度浮点数(full precise float)。如图1所示的数据结构示意图，具体的，FP16包括16个bit，其中最高位的1bit为符号域，中间5bit为指数域，剩余10bit为尾数域，尾数域用于表示小数。BF16包括16个bit，其中最高位的1bit为符号域，中间8bit为指数域，剩余7bit为尾数域。FP32包括32个bit，其中最高位的1bit为符号域，中间8bit为指数域，剩余23bit为尾数域。

下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行描述。其中，在本申请实施例的描述中，除非另有说明，“/”表示或的意思，例如，A/B可以表示A或B；本文中的“和/或”仅仅是一种描述关联对象的关联关系，表示可以存在三种关系，例如，A和/或B，可以表示：单独存在A，同时存在A和B，单独存在B这三种情况。另外，在本申请实施例的描述中，“多个”是指两个或两个以上。

以下，术语“第一”、“第二”仅用于描述目的，而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此，限定有“第一”、“第二”的特征可以明示或者隐含地包括一个或者多个该特征。在本实施例的描述中，除非另有说明，“多个”的含义是两个或两个以上。

目前，训练神经网络对算力的需求也越来越大，具体的，从最初的一次神经网络训练的过程需

一次神经网络训练的过程需要 10 亿 petaflops, 对算力的需求增长了 1000 万倍。由此, 为了降低训练神经网络时的功耗, 现有技术中定义了两种低精度浮点数据格式, 分别是 FP8(E5M2) 和 FP8(E4M3)。其中, FP8(E5M2) 的指数域 (exponent) 为 5 比特, 尾数域 (mantissa) 为 2 比特, FP8(E4M3) 的指数域为 4 比特, 尾数域为 3 比特。对于神经网络的前向计算、反向计算和权重更新的训练过程中采用混合精度进行训练, 其中, GEMM 中采用两种 FP8 数据格式进行计算, 非 GEMM 中采用 FP32 或 FP16 进行计算。可以理解的是, 通过采用上述 2 种低精度浮点数据格式的数据进行通用矩阵乘法计算, 与采用 FP16/BF16 的数据进行通用矩阵乘法计算相比, 由于此 2 种低精度浮点数据格式的总位宽是 FP16/BF16 的总位宽的一半, 因此通用矩阵乘法计算时数据量减小为一半, 从而能够降低神经网络芯片的功耗。

但是, 将高精度的浮点数据转换为上述 2 种低精度浮点数据格式进行通用矩阵乘法计算, 之后再再将数据转换为高精度的数据格式进行非通用矩阵乘法计算, 例如将数据转换为 FP16/BF16 的数据格式, 由于数据格式转换的过程中将会造成数据的精度损失, 将会导致神经网络的训练精度降低。为了减少数据的精度损失, 在训练过程中则会引入缩放 (scaling) 操作, 但是缩放操作后需要对张量核心区 (tensor core) 生成的数据进行分布统计, 增加了神经网络芯片的功耗。而且, 该 2 种低精度浮点数据在精度和动态范围的表现存在差异, 用户需要在该 2 种低精度浮点数据中, 选择一种数据格式进行神经网络训练, 目前该方法无法指示用户进行选择, 将导致用户体验较差, 泛化性较差。

另一种神经网络训练方法通过定义块浮点 (block float point) 用于低精度数据的表达, 在训练神经网络时采用混合精度进行训练, 从而降低神经网络芯片的功耗。具体的, 在进行通用矩阵乘法计算时将数据转换为块浮点数据格式之后再进行计算, 在进行非通用矩阵乘法计算时将数据转换为 FP32 或 FP16 数据格式之后再进行计算。其中, 将数据转换为块浮点数据格式时, 首先将高精度的数据分为多个块 (block), 然后统计每个块中的数据分布, 最后根据数据分布将块中的数据拆分为公共的指数域 (exponent)、以及各个数据的符号域 (sign) 和尾数域 (mantissa)。此时由于每个块中的数据都拥有公共的指数, 因此块中每个数据的总位宽减少, 从而采用该块浮点数据格式的数据进行训练时, 能够降低神经网络芯片的功耗。

但是, 采用块浮点数据格式的数据进行训练时, 由于块浮点数据格式限制了数据的表示范围, 将会造成数据的精度损失, 导致神经网络的训练精度降低。另外神经网络不同计算层存在不同的块划分方式, 将会导致算法流程更为复杂, 而且在进行通用矩阵乘法计算前后需要引入量化 (quant) 和反量化 (dequant) 操作, 需要对数据的分析进行统计, 将会影响神经网络训练的性能。

由此, 本申请实施例提供一种浮点数据精度转换方法, 该方法涉及第一浮点数据和第二浮点数据的数据格式的转换, 其中, 第一浮点数据的精度高于第二浮点数据的精度。该方法中采用混合舍入的方式, 即在第一指数域的编码值大于或等于第一预设阈值时, 采用随机舍入方式, 在第一指数域的编码值小于第一预设阈值时, 采用远离 0 进位舍入方式。通过提供不同的舍入方式, 可以降低数据格式转换时的转换误差, 提高数据转换的效率, 提高训练模型的训练精度。

该方法可以应用于 AI 领域。该浮点数据精度转换方法可以应用于中央处理器 (central processing unit, CPU)、图像处理器 (graphics processing unit, GPU)、张量处理器 (tensor processing unit, TPU) 和神经网络处理器 (neural network processing unit, NPU) 等芯片。

具体的, 本申请实施例提供的第二浮点数据可以是 HiFloat8 数据, 如表 1 所示, 表 1 为 HiFloat8 数据的编码方式。

表 1 HiFloat8 数据的编码方式

HiFloat8 数据	符号域 (S)	前缀码域 (D): {值}	指数域 (E)	尾数域 (M)
位宽	1	2: {2,3,4}	D	8-3-D
(width)	1	3: {0,1}	D	8-4-D

其中, 8 为 HiFloat8 数据的总位宽, 尾数域的位宽可以变化。符号域占据一个比特位, 0 表示正数, 1 表示负数, 或是 1 表示负数, 0 表示正数。前缀码域占据 2 个或 3 个比特位, 前缀码域可以表达 5 个不同信息, D 的值 (value) 可以是 0、1、2、3、4。指数域的位宽根据 D 的值变化, 以及尾数域占据剩余位宽。其中, 前缀码域可以采用整数 (integer) 编码, 此时 D 为固定值。前缀码域也可以采用前缀码 (prefix code) 编码, 此时 D 为有限值集合。前缀码编码中用 2 个比特位编码值 2、

码方式。

表 2 前缀码域的编码方式

值	编码	位宽
4	11	2
3	10	
2	01	
1	001	3
0	000	

根据表 2 可知，当点位域为 2 个 bit 时，可以用“11”编码数值“4”，可以用“10”编码数值“3”，可以用“01”编码数值“2”。当点位域为 3 个 bit 时，可以用“001”编码数值“1”，可以用“000”编码数值“0”。上述表 2 所示的编码方式仅为示例性说明，其并不相对本申请实施例构成限定。

进一步地，本申请实施例提供的 HiFloat8 数据和十进制的值(X)转换公式为： $X=(-1)^S*2^{E_i+E_c}*(1+M)$ 。

其中，Ec 为阶码对称中心，也是 FP32 数据中的偏置 (bias)。

当 D 为 0 时，表示指数域的值 0。当 D 不为 0 时，指数域采用符号量值 (signed magnitude) 编码，即符号位尾随原码 (true form, TF)，指数域的编码 $E_i=\{Se, 1'b1, TF[2: end]\}$, Se 为指数位符号位。TF 的最高位 1'b1 隐藏，不存储，则指数域的编码值为 $E_s=\{Se, TF[2: end]\}$ 。指数域对应于十进制的编码值为 $E_v=E_i+E_c$ 。

HiFloat(N,5,Ec)可配置为 HiFloat(8,5,0)，简写为 HiF8，也可以配置为其他情况。HiFloat8 编码数值分布如表 3 所示。

表 3HiFloat8 编码数值分布表

D	0	1	2	3	4
Es	None	Se	Se,TF[2]	Se,TF[2:3]	Se,TF[2:4]
Ei	0	Se,1	Se,1,TF[2]	Se,1,TF[2:3]	Se,1,TF[2:4]
Ev	0	±1	±[2,3]	±[4,7]	±[8,15]
M 位宽	4	3	3	2	1

结合表 1 和表 2，如表 3 所示，当 D=0 时，尾数域位宽=8-4-0=4。当 D=1 时，尾数域位宽=8-4-1=3。当 D=2 时，尾数域位宽=8-3-2=3。当 D=3 时，尾数域位宽=8-3-3=2。当 D=4 时，尾数域位宽=8-3-4=1。可以理解的是，当指数域的位宽越小（即数值范围越小）时，尾数域所占用的位宽就越大，数值精度也就越大。当指数域的位宽越大（即数值范围越大）时，尾数域所占用的位宽就越小，数值精度也就越小。

另外，如图 2 所示，为本申请实施例提供的一种指数域-尾数域位宽的分布示意图，根据图 2 可知当指数域的绝对值越小时，尾数域的位宽越大，因此该第二浮点数据也可以称为锥形低精度浮点数据。另外，锥形低精度浮点数据也可以兼顾数据的动态范围和数据精度，可以减少高精度数据向低精度数据格式转化过程中的转换误差，可以提高训练精度。

在一些可能的实施方式中，浮点数表示的数值 X 除了上述规格化表示外，还可以通过自定义设置，选择表示各类特殊值。

例如，当 S=0，D=4，Es=4'b1111=-15，M=1'b0 时，X=0 (zero)；当 S=1，D=4，ES=4'b1111=-15，M=1'b0 时，X 可以表示非数值 (not a number, NaN)；当 D=4，ES=4'b0111=15，M=1'b1 时，X=正负无穷大 (±infinity)。

在上述场景中，本申请的浮点数据精度转换方法和装置可以应用于不同的系统或设备中，如应用于图 3 所示的浮点数据精度转换装置 30，图 3 为本申请实施例提供的一种浮点数据精度转换装置应用的系统或设备示意图。该浮点数据精度转换装置可以是终端，如服务器 31、手机终端 32，平板电脑 33，笔记本电脑 34，增强现实技术 (augmented reality, AR) 设备 (图 3 中未示出)，虚拟现实技术 (virtual reality, VR) 设备 (图 3 中未示出) 和车载终端 (图 3 中未示出) 等。本申请提供的浮点数据精度转换方法可以应用于浮点数据精度转换装置 30 中涉及到 CPU、高性能计算 (high

单元、矩阵计算单元和张量计算单元等。

在一些实施例中，本申请提出的浮点数据精度转换装置可以为芯片，例如该芯片为系统级芯片（system-on-a-chip, SoC）。如图4所示，图4为本申请实施例提供的一种SoC的结构示意图。SoC包括处理器，该处理器可以为单核处理器或多核处理器、存储器和输入/输出（input/output, I/O）接口等。处理器可加载存储器中的数据和应用程序后，对数据进行处理，例如进行本申请中的计算处理。例如数据为FP32数据时，可通过读取FP32数据中第一指数域的编码值确定第二浮点数据中第二尾数域的编码值。

应用于上述系统或设备，下面对本申请实施例提供的浮点数据精度转换方法的流程进行介绍。

本申请实施例提供一种浮点数据精度转换方法，如图5所示，图5为本申请实施例提供的一种浮点数据精度转换方法的流程图。该方法应用于将第一浮点数据转换为第二浮点数据，第一浮点数据可以包括第一指数域和第一尾数域，第二浮点数据可以包括第二尾数域。该方法包括如下流程。

步骤501、浮点数据精度转换装置根据第一指数域的编码值确定第二尾数域的位宽。

示例性的，第一浮点数据的精度高于第二浮点数据的精度，其中，第一浮点数据可以是FP32数据或FP16数据，第二浮点数据可以是HiFloat8数据。

在转换的过程中，HiFloat8数据的尾数域的位宽是可变的，即第二尾数域的位宽是可变的，其位宽的范围为1-4。可以理解的是，尾数域的位宽越大，HiFloat8数据的精度也越大。

以第一浮点数据为FP32数据为例，若第一指数域的编码值为8'b01111100，表示十进制的124，去除FP32数据中的偏置（对于FP32数据，偏置为127），则该FP32数据的第一指数域为2的-3次方。结合表2和表3可以得知，第二尾数域的位宽为2。

步骤502、浮点数据精度转换装置确定第一尾数域中的保留编码值和舍弃编码值，保留编码值包括第一尾数域中从最高位开始，且位宽与第二尾数域的位宽相同的编码值。

示例性的，由于第一浮点数据的精度高于第二浮点数据的精度，第一浮点数据的第一尾数域的位宽大于第二浮点数据的第二尾数域的位宽。由此，若将第一浮点数据转换为第二浮点数据，由于第二浮点数据的第二尾数域的位宽有限，需要对第一尾数域中的编码值进行取舍。其中，将第一尾数域中从最高位开始，且位宽为第二尾数域的位宽相同的编码值确定为保留编码值，以及将第一尾数域中除保留编码值外剩余的编码值确定为舍弃编码值。

继续以第一浮点数据为FP32为例，若第一尾数域的编码值为23'b0100000000000000000000，由于第二尾数域的位宽为2，则保留编码值为2'b01。

步骤503、若第一指数域的编码值大于或等于第一预设阈值，浮点数据精度转换装置根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作，得到第二尾数域的编码值。

示例性的，根据舍弃编码值中从最高位开始，且位宽为预设位宽的编码值对保留编码值进行舍入操作的方式可以理解为随机舍入（stochastic rounding, SR）方式。其中，预设位宽可以是1至14中的整数。SR舍入方式中引入随机数，基于舍弃编码值中从最高位开始，且位宽为预设位宽的编码值和随机数的比较结果，以对保留编码值进行舍入操作。可以理解的是，SR舍入方式中的随机数和舍弃编码值相关，无需额外的随机数生成器，不存在随机数生成的性能瓶颈，由此SR舍入方式不仅提高了高精度数据向低精度数据的转换效率，同时硬件开销更低。

步骤504、若第一指数域的编码值小于第一预设阈值，浮点数据精度转换装置根据舍弃编码值的最高位对保留编码值进行舍入操作，得到第二尾数域的编码值。

示例性的，根据舍弃编码值的最高位对保留编码值进行舍入操作的方式可以理解为远离0进位（round half away from zero, TA）方式。TA舍入方式中，可以将舍弃编码值的最高位和预设阈值进行比较，基于比较结果对保留编码值进行舍入操作。其中，TA舍入方式对于其他舍入方式硬件实现面积更小、功耗开销更小，且具有更高的数据分辨率。

由此，SR舍入方式和TA舍入方式可以理解为混合舍入（hybrid rounding, HR）方式，其中，在混合舍入方式中，对于类高斯分布占比较小的两边数据，即影响均值的主要数据，采用SR舍入方式。而对于占比较大的中间数据，采用TA舍入方式。混合舍入方式可以同时应用于FP32数据和FP16数据，提供比单独使用TA舍入方式或SR舍入方式更好的均值不变性，且硬件开销更小。

可选的，舍入操作包括进位操作和舍弃操作，步骤503可以包括：舍弃编码值中从最高位开始，

低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值。舍弃编码值中从最高位开始，且位宽为预设位宽的编码值小于第二预设阈值时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的编码值。其中，第二预设阈值为舍弃编码值中从最低位开始，且位宽为预设位宽的编码值。

示例性的，对于SR舍入方式，预设位宽可以是1至14中的整数。以预设位宽为14位为例，第二预设阈值可以为舍弃编码值中从最低位开始，且位宽为14的编码值，则舍弃编码值中用于与第二预设阈值比较的部分舍弃编码值为从最高位开始，且位宽为14的编码值。继续以第一尾数域的编码值为23'b010000000000000000000000为例，此时第二尾数域的位宽为2，则第一尾数域中的保留编码值为2'b01，舍弃编码值为21'b0000000000000000000000，部分舍弃编码值为14'b0000000000000000，第二预设阈值为14'b0000000000000000。由于部分舍弃编码值等于第二预设阈值，则对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值，即第二尾数域的编码值为2'b01。

在一个示例中，如图6所示，图6为本申请实施例提供的一种基于SR舍入方式的FP32数据转换为HiFloat8数据的尾数域的示意图。其中，取FP32数据的尾数域中的最低有效位(least significant bit, LSB)，形成2比特的第二预设阈值 $T2=\{LSB, 1'b1\}$ ，可以取舍编码值中前4位比特形成F2，即F2为保留编码值，M为进行舍入操作后的保留编码值。

在另一个示例中，如图7所示，图7为本申请实施例提供的一种基于SR舍入方式的FP16数据转换为HiFloat8数据的尾数域的示意图。其中，取FP16数据的尾数域中的最低有效位(least significant bit, LSB)，形成2比特的第二预设阈值 $T2=\{LSB, 1'b1\}$ ，可以取舍编码值中前4位比特形成F2。

适用于图6和图7中的尾数域，SR舍入的定义为：if $F2 \geq T2$, $M+1$, else M 。即若F2大于等于T2，则保留编码值进行进位操作。若F2小于T2，则对舍弃编码值进行舍弃操作。

在另一种实现方式中，将第一浮点数据的第一尾数域中的任意位宽按照 $\{a_n, 1\}$ 或者 $\{\text{negate}(a_n), 1\}$ 方式作为第二预设阈值，其中，negate操作代表0, 1取反的意思， $\text{negate}(0)=1$, $\text{negate}(1)=0$ 。将第一尾数域中的部分舍弃编码值和第二预设阈值进行比较，即可得到SR舍入结果。在一个示例中，预设位宽为1位，若舍弃编码值中的最高位为0，则 $\{1'b0, 1'b1\}=0.25$ ，对舍弃编码值进行舍弃操作。若舍弃编码值中的最高位为1，则 $\{1'b1, 1'b1\}=0.75$ ，对保留编码值的最低位进行进位操作。

其中，该SR舍入方式的单数据的误差为0.75最小单位(unit in the last place, ulp)，该误差小于标准随机舍入的1ulp。另外，若数据均匀分布，SR舍入方式的均值不变性优于标准随机舍入。

可选的，舍入操作包括进位操作和舍弃操作，步骤504可以包括：舍弃编码值的最高位大于或等于第三预设阈值时，浮点数据精度转换装置对保留编码值的最低位进行进位操作，对舍弃编码值进行舍弃操作，保留编码值进位后的编码值为第二尾数域的编码值。舍弃编码值的最高位小于第三预设阈值时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的编码值。

示例性的，对于TA舍入方式，第三预设阈值可以为1。在一个示例中，舍弃编码值的最高位为1时，则对保留编码值的最低位进行进位操作，并对舍弃编码值进行舍弃操作。在另一个示例中，舍弃编码值的最高位为0时，则对舍弃编码值进行舍弃操作。

对于TA舍入方式，第三预设阈值也可以为0。舍弃编码值的最高位大于0时，对保留编码值的最低位进行进位操作，并对舍弃编码值进行舍弃操作，保留编码值进行进位操作后得到的编码值为第二尾数域的编码值。舍弃编码值的最高位小于或等于0时，对舍弃编码值进行舍弃操作，保留编码值为第二尾数域的第一编码值。

可选的，第一预设阈值由遍历多个第一浮点数据确定。

示例性的，通过遍历多个第一浮点数据，即遍历不同的第一指数域的编码值，可以确定第一预设阈值为4时，可以得到最低的转换误差。如图8所示，图8为本申请实施例提供的一种混合舍入方式的分布示意图。其中，HiFloat8的指数值 $E=[-15, 15]$ ，其中 $\text{abs}(E)=[0, 15]$ 。若 $E \geq 4$ ，则采用TA舍入方式，若 $E < 4$ ，则采用SR舍入方式。

可选的，第一浮点数据还包括符号域，第二浮点数据还包括符号域、前缀码域和第二指数域，前缀码域用于指示第二指数域的位宽。该方法还包括：根据第一指数域的编码值确定前缀码域的位宽、前缀码域的编码值、第二指数域的位宽和第二指数域的编码值。示例性的，在数据格式转换中，数据的正负是不变的，即第一浮点数据和第二浮点数据的符号域的位宽和编码值相同。

查看表 1, 可以基于第一指数域的指数值确定 D 的值。在一个示例中, 第一浮点数据为 FP32 数据, 假设第一指数域的编码值为 8'b01111100, 标识十进制的 124, 去除对于 FP32 数据的偏置 127 后, 得到十进制的 -3。其中, -3 为第一指数域的指数值 N, 利用公式 $D = \text{INT}[\log_2 |N|]$, 可以得到 D 为 2。通过查看表 2 可以确定与 2 对应的前缀码域的位宽为 2 和前缀码域的第一编码值为 01, 以及第二指数域的位宽为 2。继续查看表 3, 当 D 为 2 时, 且第一指数域的指数值为 -3, 即指数位符号位 S_e 为 1, 由于第二指数域的位宽为 2, 则确定的第二指数域的编码值为 2'b11。

可以理解的是, 若对保留编码值进行进位操作, 保留编码值有可能产生溢出。执行设备应先判断进位操作后的保留编码值是否溢出, 若进位操作后的保留编码值溢出, 则对第一指数域的编码值的最低位进行加 1 操作, 得到第一指数域的新的编码值。执行设备再基于第一指数域的新的编码值确定第二指数域新的位宽和前缀码域新的位宽。若前缀码域的新的位宽和前缀码域的原来的位宽不同, 则根据第一指数域的新的编码值确定前缀码域的新的编码值、第二指数域的新的编码值、第二尾数域的新的位宽和第二尾数域的新的编码值。若前缀码域的新的位宽和前缀码域的原来的位宽相同, 则判断第二指数域新的位宽和第二指数域的原来的位宽是否相同, 若第二指数域的新的位宽小于第二指数域的原来的位宽, 则对保留编码值的位宽进行加 1 操作, 得到第二尾数域的新的位宽和第二尾数域的新的编码值。若第二指数域的新的位宽大于第二指数域的原来的位宽, 则对保留编码值的最低位进行舍弃操作, 得到第二尾数域的新的位宽和第二尾数域的新的编码值。

本申请实施例提供的浮点数据精度转换方法可以应用于神经网络训练流程, 其中, 神经网络训练流程包括模型权重参数初始化、前向计算、反向计算、权重更新流程和多机多卡间数据通信流程。

模型权重参数初始化流程中权重采用高精度浮点数据格式, 针对不同的应用场景采用不同的初始化方案。其中, 对于从零开始的训练和预训练过程采用高精度浮点数据进行随机初始化, 随机化方式和 FP16 混合精度或 FP32 精度方式一致。对于基于 FP32 或 FP16 混合精度权重的重训练过程, 权重数据直接加载 FP32 或 FP16 混合精度训练过程得到的权重数据。模型中其他参数如归一化层参数、优化器参数等在训练与重训练过程初始化方式与权重初始化方式一致。

对权重参数初始化后, 如图 9 所示, 图 9 为本申请实施例提供的一种神经网络训练流程图。其中, L-1 表示第一矩阵乘计算层的上一个矩阵乘计算层对应的训练过程, L+1 表示第一矩阵乘计算层的下一个矩阵乘计算层对应的训练过程, L-1 和 L+1 的训练过程与第一矩阵乘计算层对应的数据矩阵和权重矩阵的训练过程相同。

在前向计算的流程中, 对于网络首层和中间层的 GEMM 操作, 需要将 FP32 数据或 FP16 数据为锥形低精度浮点数据格式后, 再输入至 GEMM 计算单元, 并输出 FP32 数据或 FP16 数据。对于激活层和归一化层, 则采用 FP32 数据或 FP16 数据进行计算。对于神经网络的尾层, 统一保持 FP32 数据类型或 FP16 数据类型。其中, 具体的流程为: 先根据第一浮点数据格式的数据转换矩阵和权重转换矩阵, 前向计算矩阵乘计算层得到第二浮点数据格式的第一输出转换矩阵。再根据第一输出转换矩阵前向计算得到第一非矩阵乘计算层, 并得到第一浮点数据格式的第二输出矩阵。该第二输出矩阵用作前向计算时, 第一非矩阵乘计算层的下一个计算层的输入矩阵。

在反向计算的流程中, 锥形低精度浮点混合精度反向流程和 FP16 训练反向流程保持一致, 采用自动缩放操作, 即对损失 (loss) 乘上适当的缩放值后, 再进行反向求导。对特征图 (feature map) 和权重的求导过程, GEMM 输入均为锥形低精度浮点数据。其中, 具体的流程为: 获取第二浮点数据格式的第一输出误差梯度矩阵, 该第一输出误差梯度矩阵用于更新第一非矩阵乘计算层对应的参数。并根据第一输出误差梯度矩阵反向传播第一非矩阵乘计算层, 得到第一浮点数据格式的第二输出误差梯度矩阵, 将第二输出误差梯度矩阵转换为第二浮点数据格式的第二输出误差梯度转换矩阵, 并根据第二输出误差梯度转换矩阵和权重转换矩阵的转置矩阵, 反向传播第一矩阵乘计算层, 得到第一浮点数据格式的第三输出误差梯度转换矩阵, 第三输出误差梯度转换矩阵用于更新神经网络前向计算时, 第一矩阵乘计算层的上一个计算层对应的参数。

权重梯度计算流程可以是: 将第二输出误差梯度转换矩阵和数据转换矩阵输入第一矩阵乘计算层, 得到第三数据格式的权重梯度转换数据, 并根据权重梯度转换数据更新权重矩阵。

在一种可能的实施例中, 如图 9 所示, 权重矩阵以及其他参数可以转换为高精度数据格式的数据, 备份存在本地存储设备中。例如, 高精度数据格式的权重矩阵可以存储在第一存储单元 w-master 中, 优化器的状态参数可以存储在第二存储单元 momentum 中, 其他参数可以存储在第三存储单元

另外，大规模训练模型常采用多机方式完成，涉及到数据并行、模型并行和流水线并行，以及上述三种并行方式的组合。多机多卡的训练方式需要进行卡间数据通信，通过锥形低精度浮点数据进行通信，可以有效解决传输带宽。

前向计算和反向计算可以采用不同的舍入方式将第一浮点数据转换为第二浮点数据，本申请实施例对此并不限定。

示例一，如图9所示，前向计算可以采用混合舍入方式，反向计算可以采用TA/SR方式。

示例二，如图10所示，图10为本申请实施例提供的另一种神经网络训练流程图。前向计算可以采用TA/SR方式，反向计算可以采用混合舍入方式。

示例三，如图11所示，图11为本申请实施例提供的另一种神经网络训练流程图。前向计算可以采用混合舍入方式，反向计算也可以采用混合舍入方式。

由此，本申请实施例提供的浮点数据精度转换方法，在卷积神经网络和transformer模型等人工智能网络模型，可以实现与FP32或FP16混合精度一致的训练精度，在双边模式（前向计算和反向计算均采用混合舍入方式）下，平均准确度比单边模式（前向计算或反向计算使用混合舍入方式）的准确度更高。

可以理解的是，为了实现上述功能，电子设备包含了执行各个功能相应的硬件和/或软件模块。结合本文中所公开的实施例描述的各示例的算法步骤，本申请能够以硬件或硬件和计算机软件的结合形式来实现。某个功能究竟以硬件还是计算机软件驱动硬件的方式来执行，取决于技术方案的特定应用和设计约束条件。本领域技术人员可以结合实施例对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

本实施例可以根据上述方法示例对电子设备进行功能模块的划分，例如，可以对应各个功能划分各个功能模块，也可以将两个或两个以上的功能集成在一个处理模块中。上述集成的模块可以采用硬件的形式实现。需要说明的是，本实施例中对模块的划分是示意性的，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式。

在采用对应各个功能划分各个功能模块的情况下，图12示出了上述实施例中涉及的浮点数据精度转换装置1200的一种可能的组成示意图，如图12所示，该浮点数据精度转换装置1200可以包括：位宽计算单元1201、尾数域计算单元1202、舍入操作单元1203。

其中，位宽计算单元1201可以用于支持浮点数据精度转换装置1200执行上述步骤501等，和/或用于本文所描述的技术的其他过程。

尾数域计算单元1202可以用于支持浮点数据精度转换装置1200执行上述步骤502等，和/或用于本文所描述的技术的其他过程。

舍入操作单元1203可以用于支持浮点数据精度转换装置1200执行上述步骤503和步骤504等，和/或用于本文所描述的技术的其他过程。

需要说明的是，上述方法实施例涉及的各步骤的所有相关内容均可以援引到对应功能模块的功能描述，在此不再赘述。

本实施例提供的浮点数据精度转换装置1200，用于执行上述浮点数据精度转换方法，因此可以达到与上述实现方法相同的效果。

在采用集成的单元的情况下，浮点数据精度转换装置1200可以包括处理模块、存储模块和通信模块。其中，处理模块可以用于对浮点数据精度转换装置1200的动作进行控制管理，例如，可以用于支持浮点数据精度转换装置1200执行上述位宽计算单元1201、尾数域计算单元1202和舍入操作单元1203执行的步骤。存储模块可以用于支持浮点数据精度转换装置1200存储程序代码和数据等。通信模块，可以用于支持浮点数据精度转换装置1200与其他设备的通信，例如与无线接入设备的通信。

其中，处理模块可以是处理器或控制器。其可以实现或执行结合本申请公开内容所描述的各种示例性的逻辑方框，模块和电路。处理器也可以是实现计算功能的组合，例如包含一个或多个微处理器组合，数字信号处理（digital signal processing, DSP）和微处理器的组合等等。存储模块可以是存储器。通信模块具体可以为射频电路、蓝牙芯片、Wi-Fi芯片等与其他电子设备交互的设备。

本申请实施例还提供一种电子设备，包括一个或多个处理器以及一个或多个存储器。该一个或多个存储器与一个或多个处理器耦合，一个或多个存储器用于存储计算机程序代码，计算机程序代

实现上述实施例中的浮点数据精度转换方法。

本申请的实施例还提供一种计算机存储介质，该计算机存储介质中存储有计算机指令，当该计算机指令在电子设备上运行时，使得电子设备执行上述相关方法步骤实现上述实施例中的浮点数据精度转换方法。

本申请的实施例还提供了一种计算机程序产品，当该计算机程序产品在计算机上运行时，使得计算机执行上述相关步骤，以实现上述实施例中电子设备执行的浮点数据精度转换方法。

另外，本申请的实施例还提供一种装置，这个装置具体可以是芯片，组件或模块，该装置可包括相连的处理器和存储器；其中，存储器用于存储计算机执行指令，当装置运行时，处理器可执行存储器存储的计算机执行指令，以使芯片执行上述各方法实施例中电子设备执行的浮点数据精度转换方法。

其中，本实施例提供的电子设备、计算机存储介质、计算机程序产品或芯片均用于执行上文所提供的对应的方法，因此，其所能达到的有益效果可参考上文所提供的对应的方法中的有益效果，此处不再赘述。

通过以上实施方式的描述，所属领域的技术人员可以了解到，为描述的方便和简洁，仅以上述各功能模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能模块完成，即将装置的内部结构划分成不同的功能模块，以完成以上描述的全部或者部分功能。

在本申请所提供的几个实施例中，应该理解到，所揭露的装置和方法，可以通过其它的方式实现。例如，以上所描述的装置实施例仅仅是示意性的，例如，所述模块或单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个装置，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，装置或单元的间接耦合或通信连接，可以是电性，机械或其它的形式。

所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是一个物理单元或多个物理单元，即可以位于一个地方，或者也可以分布到多个不同地方。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

另外，在本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

所述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用，可以存储在一个可读存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来，该软件产品存储在一个存储介质中，包括若干指令用以使得一个设备（可以是单片机，芯片等）或处理器（processor）执行本申请各个实施例所述方法的全部或部分步骤。而前述的存储介质包括：U盘、移动硬盘、只读存储器（read only memory, ROM）、随机存取存储器（random access memory, RAM）、磁碟或者光盘等各种可以存储程序代码的介质。

以上内容，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以所述权利要求的保护范围为准。

权利要求

1.一种浮点数据精度转换方法,其特征在于,第一浮点数据包括第一指数域和第一尾数域,第二浮点数据包括第二尾数域,所述第一浮点数据的精度高于所述第二浮点数据的精度,所述方法包括:

根据所述第一指数域的编码值确定所述第二尾数域的位置;

确定所述第一尾数域中的保留编码值和舍弃编码值,所述保留编码值包括所述第一尾数域中从最高位开始,且位宽与所述第二尾数域的位置相同的编码值;

若所述第一指数域的编码值大于或等于第一预设阈值,根据所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值;

若所述第一指数域的编码值小于所述第一预设阈值,根据所述舍弃编码值的最高位对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值。

2.根据权利要求1所述的方法,其特征在于,所述舍入操作包括进位操作和舍弃操作,所述根据所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值,包括:

所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值大于或等于第二预设阈值时,对所述保留编码值的最低位进行进位操作,对所述舍弃编码值进行舍弃操作,所述保留编码值进位后的编码值为所述第二尾数域的编码值;

所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值小于所述第二预设阈值时,对所述舍弃编码值进行舍弃操作,所述保留编码值为所述第二尾数域的编码值;

其中,所述第二预设阈值为所述舍弃编码值中从最低位开始,且位宽为所述预设位宽的编码值。

3.根据权利要求1所述的方法,其特征在于,所述舍入操作包括进位操作和舍弃操作,所述根据所述舍弃编码值的最高位对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值,包括:

所述舍弃编码值的最高位大于或等于第三预设阈值时,对所述保留编码值的最低位进行进位操作,对所述舍弃编码值进行舍弃操作,所述保留编码值进位后的编码值为所述第二尾数域的编码值;

所述舍弃编码值的最高位小于所述第三预设阈值时,对所述舍弃编码值进行舍弃操作,所述保留编码值为所述第二尾数域的编码值。

4.根据权利要求1-3任一项所述的方法,其特征在于,所述第一预设阈值由遍历多个所述第一浮点数据确定。

5.根据权利要求1所述的方法,其特征在于,所述第一浮点数据还包括符号域,所述第二浮点数据还包括所述符号域、前缀码域和第二指数域,所述前缀码域用于指示所述第二指数域的位置,所述根据所述第一指数域的编码值确定所述第二尾数域的位置之前,所述方法还包括:

根据所述第一指数域的编码值确定所述前缀码域的位置、所述前缀码域的编码值、所述第二指数域的位置和所述第二指数域的编码值。

6.一种浮点数据精度转换装置,其特征在于,第一浮点数据包括第一指数域和第一尾数域,第二浮点数据包括第二尾数域,所述第一浮点数据的精度高于所述第二浮点数据的精度,所述装置包括:

位置计算单元,用于根据所述第一指数域的编码值确定所述第二尾数域的位置;

尾数域计算单元,用于确定所述第一尾数域中的保留编码值和舍弃编码值,所述保留编码值包括所述第一尾数域中从最高位开始,且位宽与所述第二尾数域的位置相同的编码值;

舍入操作单元,用于若所述第一指数域的编码值大于或等于第一预设阈值,根据所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值;

舍入操作单元,还用于若所述第一指数域的编码值小于所述第一预设阈值,根据所述舍弃编码值的最高位对所述保留编码值进行舍入操作,得到所述第二尾数域的编码值。

7.根据权利要求6所述的装置,其特征在于,所述舍入操作包括进位操作和舍弃操作,所述舍入操作单元还用于:

所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值大于或等于第二预设阈值时,对所述保留编码值的最低位进行进位操作,对所述舍弃编码值进行舍弃操作,所述保留编码值进位后的编码值为所述第二尾数域的编码值;

所述舍弃编码值中从最高位开始,且位宽为预设位宽的编码值小于所述第二预设阈值时,对所述舍弃编码值进行舍弃操作,所述保留编码值为所述第二尾数域的编码值;

8.根据权利要求 6 所述的装置，其特征在于，所述舍入操作包括进位操作和舍弃操作，所述舍入操作单元还用于：

所述舍弃编码值的最高位大于或等于第三预设阈值时，对所述保留编码值的最低位进行进位操作，对所述舍弃编码值进行舍弃操作，所述保留编码值进位后的编码值为所述第二尾数域的编码值；

所述舍弃编码值的最高位小于所述第三预设阈值时，对所述舍弃编码值进行舍弃操作，所述保留编码值为所述第二尾数域的编码值。

9.根据权利要求 6-8 任一项所述的装置，其特征在于，所述第一预设阈值由遍历多个所述第一浮点数据确定。

10.根据权利要求 6 所述的装置，其特征在于，所述第一浮点数据还包括符号域，所述第二浮点数据还包括所述符号域、前缀码域和第二指数域，所述前缀码域用于指示所述第二指数域的位宽，所述位宽计算单元还用于：

根据所述第一指数域的编码值确定所述前缀码域的位宽、所述前缀码域的编码值、所述第二指数域的位宽和所述第二指数域的编码值。

11.一种浮点数据精度转换装置，其特征在于，包括处理器和存储器，所述存储器存储计算机指令，所述处理器执行所述计算机指令后执行上述权利要求 1-5 中的任一项所述的方法。

12.一种计算机可读存储介质，其特征在于，包括计算机指令，当计算机指令在电子设备上运行时，使得电子设备执行上述权利要求 1-5 中的任一项所述的方法。