

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2019 年 7 月 25 日 (25.07.2019)



(10) 国际公布号
WO 2019/141014 A1

- (51) 国际专利分类号:
G06F 9/30 (2006.01)
- (21) 国际申请号: PCT/CN2018/119732
- (22) 国际申请日: 2018 年 12 月 7 日 (07.12.2018)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201810041814.8 2018 年 1 月 16 日 (16.01.2018) CN
- (71) 申请人: 腾讯科技(深圳)有限公司 (TENCENT TECHNOLOGY (SHENZHEN) COMPANY LIMITED) [CN/CN]; 中国广东省深圳市南山区高新区科技中一路腾讯大厦 35 层, Guangdong 518057 (CN)。
- (72) 发明人: 王玉伟 (WANG, Yuwei); 中国广东省深圳市南山区高新区科技中一路腾讯大厦 35 层, Guangdong 518057 (CN)。于潇宇 (YU, Xiaoyu); 中国广东省深圳市南山区高新区科技中一路

腾讯大厦 35 层, Guangdong 518057 (CN)。张立鑫 (ZHANG, Lixin); 中国广东省深圳市南山区高新区科技中一路腾讯大厦 35 层, Guangdong 518057 (CN)。张博 (ZHANG, Bo); 中国广东省深圳市南山区高新区科技中一路腾讯大厦 35 层, Guangdong 518057 (CN)。

(74) 代理人: 北京三高永信知识产权代理有限责任公司 (BEIJING SAN GAO YONG XIN INTELLECTUAL PROPERTY AGENCY CO., LTD.); 中国北京市海淀区学院路蓟门里和景园 A 座 1 单元 102 室, Beijing 100088 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL,

(54) Title: CHIP-BASED INSTRUCTION SET PROCESSING METHOD AND APPARATUS, AND STORAGE MEDIUM

(54) 发明名称: 基于芯片的指令集处理方法、装置及存储介质

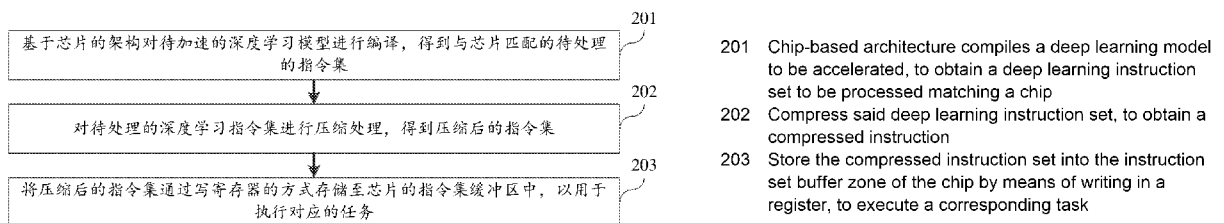


图 2

(57) Abstract: A chip-based instruction set processing method and apparatus, and a computer-readable storage medium. The method comprises: chip-based architecture compiles a deep learning model to be accelerated, to obtain a deep learning instruction set to be processed matching a chip (201); compressing said deep learning instruction set, to obtain a compressed instruction (202); and storing the compressed instruction set into the instruction set buffer zone of the chip by means of writing in a register, to execute a corresponding task (203). The compression of a deep learning instruction set to be processed can reduce the requirement for a storage space; the compressed instruction set is stored into the instruction set buffer zone of a chip by means of writing in a register, thereby implementing instruction registering, and reducing the overhead of instruction interaction.

(57) 摘要: 一种基于芯片的指令集处理方法、装置及计算机可读存储介质, 所述方法包括: 基于芯片的架构对待加速的深度学习模型进行编译, 得到与芯片匹配的待处理的深度学习指令集 (201); 对待处理的深度学习指令集进行压缩处理, 得到压缩后的指令集 (202); 将所述压缩后的指令集通过写寄存器的方式存储至所述芯片的指令集缓冲区中, 以用于执行对应的任务 (203)。通过对待处理的深度学习指令集进行压缩处理, 可以降低对存储空间的需求; 将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中, 实现指令寄存器化, 降低指令交互的开销。

PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,
SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG,
US, UZ, VC, VN, ZA, ZM, ZW。

- (84) 指定国 (除另有指明, 要求每一种可提供的地区
保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ,
NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM,
AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG,
CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU,
IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT,
RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI,
CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

- 包括国际检索报告 (条约第21条(3))。

基于芯片的指令集处理方法、装置及存储介质

本申请要求于 2018 年 1 月 16 日提交的申请号为 201810041814.8、发明名称为“基于芯片的指令集处理方法、装置及存储介质”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

本申请涉及数据处理技术领域，特别涉及一种基于芯片的指令集处理方法、装置及存储介质。

背景技术

随着深度学习的持续演进以及数据中心体系架构的重大变革，深度学习网络广泛应用在图像分类及检测中，各种深度学习模型也应运而生。然而，由于网络结构复杂、参数过多，计算时间过长使其应用受到了一定的限制。因此，对模型压缩、模型加速成为了深度学习领域比较热门的一个研究方向，而其中非常重要的一项研究就是高效的深度学习指令集的处理机制。

在相关技术中，主要通过编译器将深度学习模型编译成指令集，然后通过直接使用该指令集驱动芯片的各功能模块执行指令对应的任务。

在相关技术中，由于是直接使用指令集驱动芯片的各功能模块执行指令对应的任务，因而指令的交互较为频繁，开销大，特别是对于复杂模型，指令集占用较多存储空间。

发明内容

本申请实施例提供了一种基于芯片的指令集处理方法、装置及计算机可读存储介质，可以解决相关技术中存在的技术问题，技术方案如下：

一方面，提供了一种基于芯片的指令集处理方法，所述方法包括：

基于芯片的架构对待加速的深度学习模型进行编译，得到与所述芯片匹配的待处理的深度学习指令集；

对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；
对所述压缩后的指令集进行译码，得到译码后的指令信息；
将所述压缩后的指令集通过写寄存器的方式存储至所述芯片的指令集缓冲区中，以用于执行对应的任务。

一方面，提供了一种基于芯片的指令集处理方法，所述方法包括：
获取任务处理请求；
根据所述任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集；
对所述压缩后的指令集进行译码，得到译码后的指令信息；
将所述译码后的指令信息进行寄存器的寄存更新操作；
根据寄存更新操作控制对应的功能模块执行所述指令信息对应的任务。

一方面，还提供了一种基于芯片的指令集处理装置，所述装置包括：
模型指令编译模块，用于基于芯片的架构对待加速的深度学习模型进行编译，得到与所述芯片匹配的待处理的深度学习指令集；
指令集压缩模块，用于对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；
指令集缓存模块，用于将所述压缩后的指令集通过写寄存器的方式存储至所述芯片的指令集缓冲区中，以用于执行对应的任务。

一方面，提供了一种基于芯片的指令集处理装置，所述装置包括：
第一获取模块，用于获取任务处理请求；
第二获取模块，用于根据所述任务处理请求从指令集缓冲区中获取压缩后的指令集；
指令集译码模块，用于对所述压缩后的指令集进行译码，得到译码后的指令信息；
寄存更新模块，用于将所述译码后的指令信息进行寄存器的寄存更新操作；
流程控制模块，用于根据寄存更新操作控制对应的功能模块执行所述指令信息对应的任务。

还提供了一种计算机设备，所述计算机设备包含处理器和存储器，所述存储器中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由所述处理器加载并执行以实现上述基于芯片的指令集处理方法。

还提供了一种计算机可读存储介质，所述存储介质中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由处理器加载并执行以实现上述基于芯片的指令集处理方法。

通过对待加速的深度学习指令集进行压缩处理，可以降低对存储空间的需求；此外，将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中，实现指令寄存器化，从而后续在对压缩后的指令集进行译码得到指令信息后，通过指令信息实现任务的执行，能够降低执行任务时的指令交互的次数，减少交互开销。

附图说明

为了更清楚地说明本申请实施例中的技术方案，下面将对实施例描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

图1是本申请一示例性实施例示出的一种基于芯片的指令集处理系统架构示意图；

图2是本申请一示例性实施例示出的一种基于芯片的指令集处理方法流程图；

图3是本申请一示例性实施例示出的一种内存架构示意图；

图4是本申请一示例性实施例示出的一种深度学习指令集的压缩示意图；

图5是本申请一示例性实施例示出的一种基于芯片的指令集处理方法流程图；

图6是本申请一示例性实施例示出的一种深度学习指令集的处理过程示意图；

图7是本申请一示例性实施例示出的一种深度学习指令集的处理过程中的流程控制图；

图8是本申请一示例性实施例示出的一种基于芯片的指令集处理装置的结构示意图；

图9是本申请一示例性实施例示出的一种基于芯片的指令集处理装置的结构示意图；

图10是本申请一示例性实施例示出的一种终端的结构示意图；

图11是本申请一示例性实施例示出的一种服务器的结构示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。下面的描述涉及附图时，除非另有表示，不同附图中的相同数字表示相同或相似的要素。以下示例性实施例中所描述的实施方式并不代表与本申请相一致的所有实施方式。

随着深度学习的持续演进以及数据中心体系架构的重大变革，FPGA/ASIC（Application Specific Integrated Circuit，特定用途集成电路）等芯片作为一种异构加速解决方案逐渐走入到了互联网数据中心。由于在计算能力、响应延迟、可重构等方面的优势特性，FPGA/ASIC 在深度学习模型加速方面的应用愈加广泛；此外，以深度学习为代表的 AI（Artificial Intelligence，人工智能）算法依旧在快速的演进阶段，传统的以模型定制加速的方法无法适应算法的快速迭代，因此，设计具备一定通用性的基于芯片（如 FPGA/ASIC）的深度学习模型的加速方案十分必要。

然而，在兼备通用性的基础上，充分发挥 FPGA/ASIC 等芯片的加速性能需要考虑非常多的因素，其中非常重要的一项就是高效的深度学习指令集的处理机制，为此，本申请实施例针对 FPGA/ASIC 等芯片在深度学习模型的加速领域的应用提出了一种高效的基于芯片的指令集处理方法，该方法相较于相关技术能够以更少的资源来兼容不同模型的加速。

其中，本申请实施例提供的基于芯片的指令集处理方法可应用于如图1所示的系统架构中，该系统架构可以设置于终端或服务器等设备中。如图1所示，该系统架构包括深度学习模型101、编译器102、指令集压缩单元103、PCIe104、

指令集缓冲区 105、流程控制及指令译码单元 106、寄存器 107 以及功能模块。

其中，深度学习模型 101，即为待加速的深度学习模型。深度学习模型 101，用于发现数据的分布特征，常用于分类、检测等应用中；深度学习模型 101 可以根据应用场景进行选择，例如，根据应用场景选择 CNN (Convolutional Neural Network，卷积神经网络) 模型、DNN (Deep Neural Network，深度神经网络) 模型、或 RNN (recurrent neural network，递归神经网络) 模型等。

编译器 102，用于实现深度学习模型 101 到指令集的映射，即对深度学习模型 101 进行编译，得到待处理的深度学习指令集。

指令集压缩单元 103，用于对待处理的深度学习指令集进行压缩，压缩后的指令集通过 PCIe104 下发到指令集缓冲区 105。

PCIe104，是一种高速串行计算机扩展总线，用于指令集的传输。

指令集缓冲区 105，用于对指令集进行缓存。

流程控制及指令译码单元 106，用于当检测到有新的任务到来时，从指令集缓冲区 105 中获取指令并进行译码，得到指令信息之后，根据指令信息对寄存器 107 及功能模块进行流程控制。

寄存器 107 与功能模块可以看作是模型加速器，即芯片加速器，寄存器 107 与功能模块一一对应，不同的寄存器 107 用于寄存指令信息中的操作数信息，该操作数信息用于指示指令执行时所需数据的来源。

功能模块，用于在流程控制及指令译码单元 106 的控制下，根据与其对应的寄存器 107 中寄存的操作数信息执行相应的功能。

如图 1 所示，功能模块的数量可以为一个或多个，不同功能模块可以根据指令所需的处理方式设置。例如，功能模块包括但不限于基本计算单元 (PE) 1081、累加单元 1082、激活单元 1083、归一化 (Normalizations) 池化 (pooling) 单元 1084 以及内存访问单元 1085 等。

接下来，结合图 1 所示的系统架构，以 FPGA 芯片为例，对本申请实施例提供的基于芯片的指令集处理方法进行解释说明。如图 2 所示，该方法包括：

在步骤 201 中，基于芯片的架构对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的指令集；

在一种实现方式中，该步骤 201 可包括如下步骤 1 至步骤 3：

步骤 1：根据应用场景选择匹配的深度学习模型；

其中，应用场景包括但不限于图片分类、语音识别、智能推荐等。由于不同模型在不同场景下有各自的优势，因而可以根据应用场景选择匹配的深度学习模型。例如，在图片分类或图像处理的应用场景下，可以选择 CNN 模型；在语音识别的应用场景下，可以选择 DNN 模型或 RNN 模型。

当然，除了根据应用场景来选择深度学习模型，还可以提供设置入口，通过该设置入口获取设置的深度学习模型，并将设置的深度学习模型作为选择的深度学习模型。例如，提供设置入口，当检测到设置入口的选中指令后，显示设置界面，该设置界面上提供可供选择的深度学习模型。获取深度学习模型选择指令，将该选择指令所指示的深度学习模型作为选择的深度学习模型。

以设置界面上提供可供选择的深度学习模型分别为 CNN 模型、DNN 模型或 RNN 模型为例，如果 CNN 模型被选中，获取到用于指示 CNN 模型的深度学习模型选择指令，则将 CNN 模型作为选择的深度学习模型。

步骤 2：对选择的深度学习模型进行训练，得到待加速的深度学习模型；

当确定选择的深度学习模型之后，通过对选择的深度学习模型进行训练，可提高模型计算的准确性。训练时，可以选择 Caffe (Convolutional Architecture for Fast Feature Embedding, 快速特征嵌入的卷积结构)、Tensorflow 等主流框架，将复杂的数据结构传输至人工智能神经网络中进行分析 and 处理，从而得到待加速的深度学习模型。其中，Caffe 是一个清晰而高效的深度学习框架。TensorFlow 支持异构设备分布式计算，它能够在各个平台上自动运行模型，可被用于语音识别或图像识别等多项机器深度学习领域。选择哪种框架训练模型，可根据实际情况进行选择，本申请实施例对此不作限定。

步骤 3：基于芯片的架构对待加速的深度学习模型进行结构分析，根据分析结果对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的深度学习指令集。

基于芯片的架构对得到的待加速的深度学习模型进行结构分析时，本申请实施例以更高的 MAC (Multiply and Accumulate, 乘加) DSP (Digital Signal Processing, 数字信号处理) 计算效率以及更少的内存访问需求为目标来进行分析，主要包括数据依赖关系分析、模型结构的优化以及可变精度定点化这几部分。每个部分的实现方式如下：

1) 数据依赖关系分析

主要基于芯片的架构分析和挖掘深度学习模型中可流水化的计算以及可并

行化的计算。其中，流水化的计算可以提高模型加速器中的计算单元的利用率，并行化的计算可以在同一时刻利用尽量多的计算单元。

以 FPGA 芯片为例，对于流水化的计算，分析部分包括数据从 DDR(Double Data Rate, 双倍速率) 内存加载到 FPGA 芯片上（即加速器）SRAM (Static Random Access Memory, 静态随机存取存储器) 的操作与 PE 进行计算的流水，通过此项优化将内存访问的时间重叠；DSP 计算整列的计算控制过程，保证 DSP 利用率的提升。

关于并行化的计算，需要重点分析 PE 计算阵列与激活、pooling 以及归一化等功能模块之间的并行关系，如何确定好数据依赖关系以及防止冲突是此处设计关键。

2) 模型结构优化

主要涉及两个方面：寻找模型结构优化以及支持动态精度调整的定点化。

FPGA 是支持大量计算并行的器件，从模型结构上寻找更高维度的并行性，对于计算效率以及减少内存访问都十分有意义。例如，在一个深度学习模型中，在不同分支的第一层 1x1 卷积层，其输入数据完全一致，且卷积层的 stride（卷积核的步长）以及 pad（扩充边缘）均一致，那么，可以在输出特征图维度上对齐进行叠加，叠加后对输入数据的访存需求就降低到了原来的 1/3，即降低了访问内存的需求，增加了计算效率。

另一方面，本申请实施例提供的方法可采用支持动态精度调整的定点化 int16 方案。其中，Int16 是一种整数数据类型定义的标示符，表示值介于 -32768 到 +32767 之间的有符号整数。通过此种方法，训练好的模型可以直接通过编译器进行部署，而几乎无任何精度损失。

在上述分析基础上，带宽问题始终是计算机体系结构中制约性能的瓶颈之一，同时内存访问直接影响加速器功耗效率。本申请实施例提供了一种如图 3 所示的内存架构，以最大化的减少模型计算过程中的 DDR 访存：

- Input buffer（输入缓冲）以及 output buffer（输出缓冲） ping-pong 设计，最大化流水以及并行能力；
- 支持 Input buffer 和 output buffer 之间的 inner-copy（内部拷贝）操作；
- 支持 Input buffer 和 output buffer 之间的 cross-copy（交叉拷贝）操作；

由于现有的内存访问不支持上述 Input buffer 和 output buffer 之间的 inner-copy 操作以及 cross-copy 操作，因而当有数据处理请求从 PCIe DMA 进

入到 Input buffer，再通过功能模块处理，输出至 output buffer 之后，需要写入 DDR4 内存中。DDR4 属于片外内存，再次访问时，速度较慢。

对此，本申请实施例通过上述这种内存架构，在进行内存访问时，由于支持 Input buffer 和 output buffer 之间的 inner-copy 操作以及 cross-copy 操作，当有数据处理请求从 PCIe DMA 进入到 Input buffer，再通过功能模块处理，输出至 output buffer 之后，如果后续还需要对数据进行使用，则可以先不写入 DDR4 内存，而是再返回至 Input buffer。由于 Input buffer 和 output buffer 为芯片内缓冲，因而与 DDR4 的片外访问相比，不仅提高了访问速度，还降低了功耗。

基于上述分析，对于大多数目前主流的深度学习模型，加速器可以做到将中间数据全部缓存在 FPGA 片上，除了模型权重的加载外，中间无需消耗任何额外的内存操作。对于无法将中间层 feature map（特征图）完全存储在 FPGA 片上的模型，本申请实施例采用在 Channel（通道）维度上引入了 slice（切片）分片的概念，在 feature map 维度上引入了 part（部分）分片的概念。通过编译器将一次卷积或是池化/归一化操作进行合理的拆分，将 DDR 访存操作与 FPGA 加速计算进行流水化操作，在优先保证 DSP 计算效率的前提下尽量减少了 DDR 的访存需求。

综上所述，通过上述分析计算过程中的数据依赖及并行关系，根据深度学习模型各个层的参数信息，将一个完整的深度学习模型拆分成芯片可以支持的更小的计算单元。根据上述分析和拆分过程对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的深度学习指令集。

其中，与芯片匹配的待处理的深度学习指令集可以为根据芯片加速器中各功能模块的处理能力选择的粗粒度的指令集，从而减少模型映射后的指令集数量，进而使用较少的存储空间，进一步减少指令交互开销。

此外，由于对于某种特定的任务，模型训练完成上线部署运行之后的一段时间内保持不变，因此，深度学习模型对应的指令集可以在芯片加速器初始化时完成加载，无需针对每一次 workload（任务）下发，这样可以大大减少指令交互。

在步骤 202 中，对待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；

在步骤 201 的基础上，对于通用的深度学习加速器而言，深度学习模型被分解成了很小的可执行单元，并在硬件中被执行。以大矩阵乘法的执行为例：

假如深度学习加速器需要拆分执行 n 次完成计算，每一次计算需要有 k 条指令来完成。指令内容包括矩阵长宽高维度、矩阵源地址、矩阵目的地址、矩阵是否需要完成累加等信息。正常情况下，需要 $n*k$ 条指令来完成本次大矩阵乘法。但对于深度学习这种计算密度很大的应用场景，被拆分后的任务的指令集有很大一部分是相同的，也就有被压缩的空间。

因此，当获取到待处理的深度学习指令集后，本申请实施例采取了对待处理的深度学习指令集进行压缩处理的方式，以减少后续指令交互的开销。压缩处理的方式包括但不限于如下任一种：

第一种压缩处理方式：相同内容压缩。

考虑到待处理的深度学习指令集中的每个指令包括操作码和操作数信息，因此，对待处理的深度学习指令集进行压缩处理时，将待处理的深度学习指令集中具有相同操作码的指令进行压缩处理，得到压缩后的指令集。例如，如图4所示的深度学习指令集的压缩示意图。

第二种压缩处理方式：基于位宽压缩。

该种方式中，可以获取待处理的深度学习指令集中每条指令的位宽，将位宽小于参考数值的指令进行合并处理，得到压缩后的指令集。

其中，该参考数值可以根据实际情况进行设置，以参考数值设置为 32 bit（比特）为例，获取每条指令的位宽后，将不足 32bit 内容的指令进行合并。例如，一条指示 kernel size（内核大小）的指令的位宽为 5bit，一条指示 feature map size（特征图大小）的指令的位宽为 26bit，则可以将这两条指令进行合并，得到压缩后的一条指令。

当然，除上述压缩方式外，还可以采用其他压缩方式，本申请实施例不对压缩方式进行限定。

在步骤 203 中，将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中，以用于执行对应的任务。

无论采用哪种方式进行压缩，压缩后的指令集可以是一定位宽（例如 32）的寄存器数组，这些压缩后的指令集通过 PCIe 写寄存器的方式写入到指令集缓冲区。该指令集缓冲区可以由 BRAM 组成，则该过程的实现需要占用芯片的 BRAM 资源。然而，由于指令集进行了压缩，则使用的 BRAM 资源较不压缩的情况变少，因而降低了资源使用率，对产品开发来说具有积极的意义。

此外，压缩后的指令集存储至芯片的指令集缓冲区中之后，本申请实施例

提供的方法还可以通过该压缩后的指令集来执行对应的任务，执行任务的过程可参见下面图 5 所示的方法流程，此处暂不赘述。

本申请实施例提供的方法，通过对待加速的深度学习指令集进行压缩处理，可以降低对存储空间的需求；此外，将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中，实现指令寄存器化，从而在执行任务时，能够降低指令交互的次数，减少交互开销。

本申请实施例提供了一种基于芯片的指令集处理方法，该方法在上述图 2 所示实施例的基础上，针对执行任务时的指令集处理方式进行解释说明。如图 5 所示，该方法包括：

在步骤 501 中，获取任务处理请求；

由于深度学习模型由一个个 layer（层）构成，每个 layer 核心是由几种相对固定的计算算子构成，各种深度学习模型架构之间的主要区别在于如何排列组合这些算子，如一些深度学习模型的基本构成均是卷积计算以及激活函数，而芯片实现的是基本算子，例如如何完成卷积计算、矩阵乘矩阵计算以及激活函数等。因此，采用上述图 2 所示实施例的方法将深度学习模型映射为指令集后，每一组指令集可以驱动芯片中对应的功能模块执行一次计算任务，一次计算任务可以对应一个任务处理请求。

整个深度学习模型的完整计算流程可能需要有多次计算任务，则各个计算任务对应的任务处理请求可以构成任务请求队列。当一次任务执行完毕后，从任务请求队列中获取新的任务处理请求。

在步骤 502 中，根据任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集；

当获取到新的任务处理请求后，即触发从芯片的指令集缓冲区中获取压缩后的指令集。

如果各个计算任务对应的任务处理请求构成任务请求队列，则当一次任务执行完毕，从任务请求队列中获取到下一个新的任务处理请求后，根据该新的任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集。

在步骤 503 中，对压缩后的指令集进行译码，得到译码后的指令信息；

由于每个不同的指令由不同的操作码来区分，因而对压缩后的指令集进行译码时，可以根据操作码来获得指令中的操作数信息，即得到指令信息。

对于一般的深度学习指令集主要包括四种类型的指令信息：computational（计算），logical（逻辑），control（控制）以及 data transfer（数据传输）。而每个指令都有对应的 Instruction Type（指令类型）域，根据 Instruction Type 域的定义可以译码出对应的指令内容，即得到对应的指令信息。

当然，还可以根据编译器设置子字段来划分不同的指令类型，进而根据不同的指令类型译码得出对应的指令信息。对于译码方式，本申请实施例不做限定。

在步骤 504 中，将译码后的指令信息进行寄存器的寄存更新操作；

结合图 1 所示的系统架构，每一个功能模块均对应一个指令寄存器，当流程控制及指令译码单元分配过来新的指令信息时，控制寄存器进行寄存更新操作，并控制功能模块执行对应的任务。

其中，寄存器可以认为是一个变量，写入一个值后，如果不去重新赋值，该寄存器的值不会更改。译码后的指令信息中包括操作数信息，不同的寄存器保存有不同的操作数信息；将译码后的指令信息进行寄存器的寄存更新操作，包括：根据译码后的指令信息更新对应的寄存器所保存的操作数信息，即更新寄存器的值。

例如，针对代表 channel 个数的变量 channel_num，在上一次计算时其值是 512，即寄存器的值为 512；到了这次计算，这个变量是 1024，那么根据指令信息，将其赋值成为 1024，即将寄存器的值重新赋值为 1024，该过程即为寄存更新操作。

在 FPGA 加速器中，需要将除功能模块启动指令外的其他指令进行寄存器化，以支持上述指令集压缩的方式。通过寄存器化，每个功能模块通过寄存器将指令中的操作数信息进行保存。

例如，一个任务为进行一次矩阵乘法操作，需要 a 矩阵的长宽 m 和 n，以及 b 矩阵的长宽 n 和 k，则 m、n 和 k 这三个操作数信息被保存在不同的寄存器中。

在下一次功能模块执行任务启动前，如有新的 opcode（操作码）相同的指令，则更新对应的寄存器，无则继续保留相关操作数信息的寄存器状态。

在步骤 505 中，根据寄存更新操作控制对应的功能模块执行指令信息对应的任务。

由于不同功能模块对应不同的寄存器，指令寄存器化之后，FPGA 加速器

中的流程控制及指令译码单元通过指令信息控制各个功能模块的执行。其中，根据寄存更新操作控制对应的功能模块执行指令信息对应的任务，包括：控制进行寄存更新操作的寄存器所对应的功能模块，根据寄存更新操作后所保存的操作数信息执行对应的功能操作。例如，以图 1 所示的系统架构为例，若进行寄存更新操作的寄存器所对应的功能模块为累加单元 1082，则控制累加单元 1082 根据寄存更新操作后所保存的操作数信息执行对应的累加操作。

流程控制及指令译码单元通过判断各个功能模块的状态信息来控制指令的获取及执行，进而完成完整模型的加速任务。可选的，功能模块的状态信息可用于反映功能模块的执行状态，如是否正在执行任务，如果否，则可以继续控制指令的获取及执行。此外，各功能模块和芯片加速器流程控制及指令译码单元之间的基本接口包括 `fetch_inst`: 执行指令命令；`exe_busy`: 指令执行中；`inst_done`: 指令执行完毕。通过上述各个接口，以实现流程控制。

每一组指令集的粒度是可以驱动一个单独的功能模块完成一次单独的计算任务。例如，对于 Pooling 操作，如果需要完成模型中某层数据的 pooling 操作，无须用逐条的指令驱动一个接一个数据点计算；只需要将 pooling 计算的参数信息通过指令的方式译码到 pooling 接口的寄存器中，就可以完成一整次操作，中间的计算控制过程由 pooling 功能模块内部逻辑完成。

同 pooling 计算原理一样，对应于矩阵计算、卷积计算等也是如此。根据芯片中各功能模块设计的处理能力选择粗粒度的指令集设计可以大规模的减少模型映射后指令集的数量。

基于上述对指令集处理的方法介绍，处理指令集的加速栈可如图 6 所示，当执行任务时，指令集处理的流程控制过程可如图 7 所示。

例如，系统启动后，根据模型各个层的参数信息，通过上述分析计算过程中的数据依赖及并行关系，将一个完整的深度学习模型拆分成芯片可以支持的更小的计算单元。根据上述分析和拆分过程对待加速的深度学习模型进行编译，得到待处理的深度学习指令集。对深度学习指令集进行压缩处理之后，即完成模型到指令集的初始化过程。压缩后的指令集以写寄存器的方式缓存到指令集缓冲区内。当有任务时，根据任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集，再将压缩后的指令集发送至芯片的流程控制及指令译码单元，由该流程控制及指令译码单元对指令集进行译码，得到指令信息。接下来，便可以读取指令集的指令信息，执行任务。执行任务时，控制寄存器进行寄存

更新操作之后，控制功能模块执行对应操作，从而完成任务的处理。如果任务未结束，继续从指令集缓冲区读取指令集，执行任务，直至任务结束，发起中断。等待完成中断之后，返回中断响应。

上述指令集处理的流程控制过程，无需像相关技术那样针对每个指令逐一驱动功能模块执行，而是直接采用指令信息的方式实现一类或一组指令的执行。例如，将一个任务的整个操作的所有指令译码得到指令信息后，仅通过指令信息实现整个任务的执行，执行过程由功能模块根据指令信息内部实现，从而完成一次操作，减少了指令的交互。

本申请实施例提供的方法，通过根据任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集，在对压缩后的指令集进行译码得到指令信息后，通过将译码后的指令信息进行寄存器的寄存更新操作，实现指令寄存器化，从而后续通过指令信息实现任务的执行，能够降低指令交互的次数，减少交互开销。

需要说明的是，上述图 2 及图 5 所示的基于芯片的指令集处理方法可以应用于终端或服务器中，还可以应用于云端。例如，该方法可应用于智慧城市、无人机及无人车等场景中，也可以应用于云计算场景中，本申请不对上述方法的应用场景进行限定。

本申请实施例提供了一种基于芯片的指令集处理装置，用于执行上述图 2 所示的方法。如图 8 所示，本申请实施例提供的基于芯片的指令集处理装置包括：

模型指令编译模块 81，用于基于芯片的架构对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的深度学习指令集；

指令集压缩模块 82，用于对待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；

指令集缓存模块 83，用于将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中，以用于执行对应的任务。

在一种实施方式中，模型指令编译模块 81，用于根据应用场景选择匹配的深度学习模型；对选择的深度学习模型进行训练，得到待加速的深度学习模型；基于芯片的架构对待加速的深度学习模型进行结构分析，根据分析结果对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的指令集。

在一种实现方式中，模型指令编译模块 81，用于基于芯片的架构对待加速的深度学习模型进行依赖关系分析、模型结构的优化分析以及可变精度定点化分析中的至少一种分析。

在一种实现方式中，模型指令编译模块 81，用于基于芯片的架构分析和挖掘待加速的深度学习模型中可流水化的计算及可并行化的计算。

其中，待处理的深度学习指令集中的每个指令包括操作码和操作数信息；

在一种实施方式中，指令集压缩模块 82，用于将待处理的深度学习指令集中具有相同操作码的指令进行压缩处理，得到压缩后的指令集。

在一种实施方式中，指令集压缩模块 82，用于获取待处理的深度学习指令集中每条指令的位宽，将位宽小于预设数值的指令进行合并处理，得到压缩后的指令集。

本申请实施例提供的装置，通过对待加速的深度学习指令集进行压缩处理，可以降低对存储空间的需求；此外，将压缩后的指令集通过写寄存器的方式存储至芯片的指令集缓冲区中，实现指令寄存器化，从而在执行任务时，能够降低指令交互的次数，减少交互开销。

本申请实施例提供了一种基于芯片的指令集处理装置，用于执行如图 5 所示的方法。如图 9 所示，本申请实施例提供的装置包括：

第一获取模块 901，用于获取任务处理请求；

第二获取模块 902，用于根据任务处理请求从指令集缓冲区中获取压缩后的指令集；

指令集译码模块 903，用于对压缩后的指令集进行译码，得到译码后的指令信息；

寄存更新模块 904，用于将译码后的指令信息进行寄存器的寄存更新操作；

流程控制模块 905，用于根据寄存更新操作控制对应的功能模块执行指令信息对应的任务。

在一种实现方式中，译码后的指令信息包括操作数信息，不同的寄存器保存有不同的操作数信息；

寄存更新模块 904，用于根据译码后的指令信息更新对应的寄存器所保存的操作数信息。

在一种实施方式中，不同功能模块对应不同的寄存器；流程控制模块 905，

用于控制进行寄存更新操作的寄存器所对应的功能模块，根据寄存更新操作后所保存的操作数信息执行对应的功能操作。

本申请实施例提供的装置，通过根据任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集，在对压缩后的指令集进行译码得到指令信息后，通过将译码后的指令信息进行寄存器的寄存更新操作，实现指令寄存器化，从而后续通过指令信息实现任务的执行，能够降低指令交互的次数，减少交互开销。

需要说明的是：上述实施例提供的基于芯片的指令集处理装置，仅以上述各功能模块的划分进行举例说明，可选地，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。另外，上述实施例提供的基于芯片的指令集处理装置和基于芯片的指令集处理方法实施例属于同一构思，其具体实现过程详见方法实施例，这里不再赘述。

再有，上述图 8 和图 9 所示的装置还可以合并为一个装置，也就是说，本申请实施例提供的深度学习指令集的处理装置可以既包括图 8 所示的各个模块，也包括图 9 所示的各个模块。

图 10 示出了本申请一个示例性实施例提供的终端 1000 的结构框图。该终端 1000 可以是智能手机、平板电脑、MP3 播放器（Moving Picture Experts Group Audio Layer III，动态影像专家压缩标准音频层面 3）、MP4（Moving Picture Experts Group Audio Layer IV，动态影像专家压缩标准音频层面 4）播放器、笔记本电脑或台式电脑。终端 1000 还可能被称为用户设备、便携式终端、膝上型终端、台式终端等其他名称。

通常，终端 1000 包括有：处理器 1001 和存储器 1002。

处理器 1001 可以包括一个或多个处理核心，比如 4 核心处理器、8 核心处理器等。处理器 1001 可以采用 DSP（Digital Signal Processing，数字信号处理）、FPGA（Field — Programmable Gate Array，现场可编程门阵列）、PLA（Programmable Logic Array，可编程逻辑阵列）中的至少一种硬件形式来实现。处理器 1001 也可以包括主处理器和协处理器，主处理器是用于对在唤醒状态下的数据进行处理的处理器，也称 CPU（Central Processing Unit，中央处理器）；协处理器是用于对在待机状态下的数据进行处理的低功耗处理器。在一些实施

例中,处理器 1001 可以在集成有 GPU (Graphics Processing Unit, 图像处理器), GPU 用于负责显示屏所需要显示的内容的渲染和绘制。一些实施例中,处理器 1001 还可以包括 AI (Artificial Intelligence, 人工智能) 处理器, 该 AI 处理器用于处理有关机器学习的计算操作。

存储器 1002 可以包括一个或多个计算机可读存储介质, 该计算机可读存储介质可以是非暂态的。存储器 1002 还可包括高速随机存取存储器, 以及非易失性存储器, 比如一个或多个磁盘存储设备、闪存存储设备。在一些实施例中, 存储器 1002 中的非暂态的计算机可读存储介质用于存储至少一个指令, 该至少一个指令用于被处理器 1001 所执行以实现本申请中方法实施例提供的基于芯片的指令集处理方法。

在一些实施例中, 终端 1000 还可选包括有: 外围设备接口 1003 和至少一个外围设备。处理器 1001、存储器 1002 和外围设备接口 1003 之间可以通过总线或信号线相连。各个外围设备可以通过总线、信号线或电路板与外围设备接口 1003 相连。具体地, 外围设备包括: 射频电路 1004、触摸显示屏 1005、摄像头 1006、音频电路 1007、定位组件 1008 和电源 1009 中的至少一种。

外围设备接口 1003 可被用于将 I/O (Input /Output, 输入/输出) 相关的至少一个外围设备连接到处理器 1001 和存储器 1002。在一些实施例中, 处理器 1001、存储器 1002 和外围设备接口 1003 被集成在同一芯片或电路板上; 在一些其他实施例中, 处理器 1001、存储器 1002 和外围设备接口 1003 中的任意一个或两个可以在单独的芯片或电路板上实现, 本实施例对此不加以限定。

射频电路 1004 用于接收和发射 RF (Radio Frequency, 射频) 信号, 也称电磁信号。射频电路 1004 通过电磁信号与通信网络以及其他通信设备进行通信。射频电路 1004 将电信号转换为电磁信号进行发送, 或者, 将接收到的电磁信号转换为电信号。可选地, 射频电路 1004 包括: 天线系统、RF 收发器、一个或多个放大器、调谐器、振荡器、数字信号处理器、编解码芯片组、用户身份模块卡等等。射频电路 1004 可以通过至少一种无线通信协议来与其它终端进行通信。该无线通信协议包括但不限于: 万维网、城域网、内联网、各代移动通信网络 (2G、3G、4G 及 5G)、无线局域网和/或 WiFi (Wireless Fidelity, 无线保真) 网络。在一些实施例中, 射频电路 1004 还可以包括 NFC (Near Field Communication, 近距离无线通信) 有关的电路, 本申请对此不加以限定。

显示屏 1005 用于显示 UI (User Interface, 用户界面)。该 UI 可以包括图

形、文本、图标、视频及其它们的任意组合。当显示屏 1005 是触摸显示屏时，显示屏 1005 还具有采集在显示屏 1005 的表面或表面上方的触摸信号的能力。该触摸信号可以作为控制信号输入至处理器 1001 进行处理。此时，显示屏 1005 还可以用于提供虚拟按钮和/或虚拟键盘，也称软按钮和/或软键盘。在一些实施例中，显示屏 1005 可以为一个，设置终端 1000 的前面板；在另一些实施例中，显示屏 1005 可以为至少两个，分别设置在终端 1000 的不同表面或呈折叠设计；在再一些实施例中，显示屏 1005 可以是柔性显示屏，设置在终端 1000 的弯曲表面上或折叠面上。甚至，显示屏 1005 还可以设置成非矩形的不规则图形，也即异形屏。显示屏 1005 可以采用 LCD(Liquid Crystal Display, 液晶显示屏)、OLED(Organic Light-Emitting Diode, 有机发光二极管)等材质制备。

摄像头组件 1006 用于采集图像或视频。可选地，摄像头组件 1006 包括前置摄像头和后置摄像头。通常，前置摄像头设置在终端的前面板，后置摄像头设置在终端的背面。在一些实施例中，后置摄像头为至少两个，分别为主摄像头、景深摄像头、广角摄像头、长焦摄像头中的任意一种，以实现主摄像头和景深摄像头融合实现背景虚化功能、主摄像头和广角摄像头融合实现全景拍摄以及 VR (Virtual Reality, 虚拟现实) 拍摄功能或者其它融合拍摄功能。在一些实施例中，摄像头组件 1006 还可以包括闪光灯。闪光灯可以是单色温闪光灯，也可以是双色温闪光灯。双色温闪光灯是指暖光闪光灯和冷光闪光灯的组合，可以用于不同色温下的光线补偿。

音频电路 1007 可以包括麦克风和扬声器。麦克风用于采集用户及环境的声波，并将声波转换为电信号输入至处理器 1001 进行处理，或者输入至射频电路 1004 以实现语音通信。出于立体声采集或降噪的目的，麦克风可以为多个，分别设置在终端 1000 的不同部位。麦克风还可以是阵列麦克风或全向采集型麦克风。扬声器则用于将来自处理器 1001 或射频电路 1004 的电信号转换为声波。扬声器可以是传统的薄膜扬声器，也可以是压电陶瓷扬声器。当扬声器是压电陶瓷扬声器时，不仅可以将电信号转换为人类可听见的声波，也可以将电信号转换为人类听不见的声波以进行测距等用途。在一些实施例中，音频电路 1007 还可以包括耳机插孔。

定位组件 1008 用于定位终端 1000 的当前地理位置，以实现导航或 LBS (Location Based Service, 基于位置的服务)。定位组件 1008 可以是基于美国的 GPS (Global Positioning System, 全球定位系统)、中国的北斗系统或俄罗斯

的伽利略系统的定位组件。

电源 1009 用于为终端 1000 中的各个组件进行供电。电源 1009 可以是交流电、直流电、一次性电池或可充电电池。当电源 1009 包括可充电电池时，该可充电电池可以是有线充电电池或无线充电电池。有线充电电池是通过有线线路充电的电池，无线充电电池是通过无线线圈充电的电池。该可充电电池还可以用于支持快充技术。

在一些实施例中，终端 1000 还包括有一个或多个传感器 1010。该一个或多个传感器 1010 包括但不限于：加速度传感器 1011、陀螺仪传感器 1012、压力传感器 1013、指纹传感器 1014、光学传感器 1015 以及接近传感器 1016。

加速度传感器 1011 可以检测以终端 1000 建立的坐标系的三个坐标轴上的加速度大小。比如，加速度传感器 1011 可以用于检测重力加速度在三个坐标轴上的分量。处理器 1001 可以根据加速度传感器 1011 采集的重力加速度信号，控制触摸显示屏 1005 以横向视图或纵向视图进行用户界面的显示。加速度传感器 1011 还可以用于游戏或者用户的运动数据的采集。

陀螺仪传感器 1012 可以检测终端 1000 的机体方向及转动角度，陀螺仪传感器 1012 可以与加速度传感器 1011 协同采集用户对终端 1000 的 3D 动作。处理器 1001 根据陀螺仪传感器 1012 采集的数据，可以实现如下功能：动作感应（比如根据用户的倾斜操作来改变 UI）、拍摄时的图像稳定、游戏控制以及惯性导航。

压力传感器 1013 可以设置在终端 1000 的侧边框和/或触摸显示屏 1005 的下层。当压力传感器 1013 设置在终端 1000 的侧边框时，可以检测用户对终端 1000 的握持信号，由处理器 1001 根据压力传感器 1013 采集的握持信号进行左右手识别或快捷操作。当压力传感器 1013 设置在触摸显示屏 1005 的下层时，由处理器 1001 根据用户对触摸显示屏 1005 的压力操作，实现对 UI 界面上的可操作性控件进行控制。可操作性控件包括按钮控件、滚动条控件、图标控件、菜单控件中的至少一种。

指纹传感器 1014 用于采集用户的指纹，由处理器 1001 根据指纹传感器 1014 采集到的指纹识别用户的身份，或者，由指纹传感器 1014 根据采集到的指纹识别用户的身份。在识别出用户的身份为可信身份时，由处理器 1001 授权该用户执行相关的敏感操作，该敏感操作包括解锁屏幕、查看加密信息、下载软件、支付及更改设置等。指纹传感器 1014 可以被设置终端 1000 的正面、

背面或侧面。当终端 1000 上设置有物理按键或厂商 Logo 时，指纹传感器 1014 可以与物理按键或厂商 Logo 集成在一起。

光学传感器 1015 用于采集环境光强度。在一个实施例中，处理器 1001 可以根据光学传感器 1015 采集的环境光强度，控制触摸显示屏 1005 的显示亮度。具体地，当环境光强度较高时，调高触摸显示屏 1005 的显示亮度；当环境光强度较低时，调低触摸显示屏 1005 的显示亮度。在另一个实施例中，处理器 1001 还可以根据光学传感器 1015 采集的环境光强度，动态调整摄像头组件 1006 的拍摄参数。

接近传感器 1016，也称距离传感器，通常设置在终端 1000 的前面板。接近传感器 1016 用于采集用户与终端 1000 的正面之间的距离。在一个实施例中，当接近传感器 1016 检测到用户与终端 1000 的正面之间的距离逐渐变小时，由处理器 1001 控制触摸显示屏 1005 从亮屏状态切换为息屏状态；当接近传感器 1016 检测到用户与终端 1000 的正面之间的距离逐渐变大时，由处理器 1001 控制触摸显示屏 1005 从息屏状态切换为亮屏状态。

本领域技术人员可以理解，图 10 中示出的结构并不构成对终端 1000 的限定，可以包括比图示更多或更少的组件，或者组合某些组件，或者采用不同的组件布置。

图 11 是根据一示例性实施例示出的一种服务器的结构示意图。所述服务器 1100 包括中央处理单元（CPU）1101、包括随机存取存储器（RAM）1102 和只读存储器（ROM）1103 的系统存储器 1104，以及连接系统存储器 1104 和中央处理单元 1101 的系统总线 1105。所述服务器 1100 还包括帮助计算机内的各个器件之间传输信息的基本输入/输出系统（I/O 系统）1106，和用于存储操作系统 1113、应用程序 1114 和其他程序模块 1115 的存储设备 1107。

所述基本输入/输出系统 1106 包括有用于显示信息的显示器 1108 和用于用户输入信息的诸如鼠标、键盘之类的输入设备 1109。其中所述显示器 1108 和输入设备 1109 都通过连接到系统总线 1105 的输入输出控制器 1110 连接到中央处理单元 1101。所述基本输入/输出系统 1106 还可以包括输入输出控制器 1110 以用于接收和处理来自键盘、鼠标、或电子触控笔等多个其他设备的输入。类似地，输入输出控制器 1110 还提供输出到显示屏、打印机或其他类型的输出设备。

所述存储设备 1107 通过连接到系统总线 1105 的大容量存储控制器（未示出）连接到中央处理单元 1101。所述存储设备 1107 及其相关联的计算机可读介质为服务器 1100 提供非易失性存储。也就是说，所述大容量存储设备 1107 可以包括诸如硬盘或者 CD-ROM 驱动器之类的计算机可读介质（未示出）。

不失一般性，所述计算机可读介质可以包括计算机存储介质和通信介质。计算机存储介质包括以用于存储诸如计算机可读指令、数据结构、程序模块或其他数据等信息的任何方法或技术实现的易失性和非易失性、可移动和不可移动介质。计算机存储介质包括 RAM、ROM、EPROM、EEPROM、闪存或其他固态存储其技术，CD-ROM、DVD 或其他光学存储、磁带盒、磁带、磁盘存储或其他磁性存储设备。当然，本领域技术人员可知所述计算机存储介质不局限于上述几种。上述的系统存储器 1104 和存储设备 1107 可以统称为存储器。

根据本申请的各种实施例，所述服务器 1100 还可以通过诸如因特网等网络连接到网络上的远程计算机运行。也即服务器 1100 可以通过连接在所述系统总线 1105 上的网络接口单元 1111 连接到网络 1112，或者说，也可以使用网络接口单元 1111 来连接到其他类型的网络或远程计算机系统（未示出）。

所述存储器还包括一个或者一个以上的程序，所述一个或者一个以上程序存储于存储器中，中央处理器 1101 通过执行该一个或一个以上程序来实现本申请实施例提供的基于芯片的指令集处理方法。

在示例性实施例中，还提供了一种包括指令的非临时性计算机可读存储介质，例如包括至少一条指令、至少一段程序、代码集或指令集的存储器，上述至少一条指令、至少一段程序、代码集或指令集可由处理器执行以完成本申请实施例的全部或者部分步骤。例如，所述非临时性计算机可读存储介质可以是 ROM、随机存取存储器（RAM）、CD-ROM、磁带、软盘和光数据存储设备等。

以上所述仅为本申请可选实施例，并不用以限制本申请，凡在本申请的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本申请的保护范围之内。

权 利 要 求 书

1、一种基于芯片的指令集处理方法，其中，所述方法应用于终端或服务器中，所述方法包括：

基于芯片的架构对待加速的深度学习模型进行编译，得到与所述芯片匹配的待处理的深度学习指令集；

对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；

将所述压缩后的指令集通过写寄存器的方式存储至所述芯片的指令集缓冲区内，以用于执行对应的任务。

2、根据权利要求1所述的方法，其中，所述基于芯片的架构对待加速的深度学习模型进行编译，得到与所述芯片匹配的待处理的深度学习指令集，包括：

根据应用场景选择匹配的深度学习模型；

对选择的深度学习模型进行训练，得到待加速的深度学习模型；

基于芯片的架构对所述待加速的深度学习模型进行结构分析，根据分析结果对所述待加速的深度学习模型进行编译，得到与所述芯片匹配的待处理的深度学习指令集。

3、根据权利要求2所述的方法，其中，所述基于芯片的架构对所述待加速的深度学习模型进行结构分析，包括：

基于芯片的架构对所述待加速的深度学习模型进行依赖关系分析、模型结构的优化分析以及可变精度定点化分析中的至少一种分析。

4、根据权利要求3所述的方法，其中，所述基于所述芯片的架构对所述待加速的深度学习模型进行依赖关系分析，包括：

基于所述芯片的架构，分析和挖掘所述待加速的深度学习模型中可流水化的计算及可并行化的计算。

5、根据权利要求1至4中任一所述的方法，其中，所述待处理的深度学习指令集中的每个指令包括操作码和操作数信息；

所述对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集，包括：

将待处理的深度学习指令集中具有相同操作码的指令进行压缩处理，得到压缩后的指令集。

6、根据权利要求1至4中任一所述的方法，其中，所述对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集，包括：

获取所述待处理的深度学习指令集中每条指令的位宽，将位宽小于预设数值的指令进行合并处理，得到压缩后的指令集。

7、一种基于芯片的指令集处理方法，其中，所述方法应用于终端或服务器中，所述方法包括：

获取任务处理请求；

根据所述任务处理请求从芯片的指令集缓冲区中获取压缩后的指令集；

对所述压缩后的指令集进行译码，得到译码后的指令信息；

将所述译码后的指令信息进行寄存器的寄存更新操作；

根据寄存更新操作控制对应的功能模块执行所述指令信息对应的任务。

8、根据权利要求7所述的方法，其中，所述译码后的指令信息包括操作数信息，不同的寄存器保存有不同的操作数信息；

所述将所述译码后的指令信息进行寄存器的寄存更新操作，包括：

根据所述译码后的指令信息更新对应的寄存器所保存的操作数信息。

9、根据权利要求7或8所述的方法，其中，不同功能模块对应不同的寄存器；所述根据寄存更新操作控制对应的功能模块执行所述指令信息对应的任务，包括：

控制进行寄存更新操作的寄存器所对应的功能模块，根据寄存更新操作后所保存的操作数信息执行对应的功能操作。

10、一种基于芯片的指令集处理装置，其中，所述装置包括：

模型指令编译模块，用于基于芯片的架构对待加速的深度学习模型进行编

译，得到与所述芯片匹配的待处理的深度学习指令集；

指令集压缩模块，用于对所述待处理的深度学习指令集进行压缩处理，得到压缩后的指令集；

指令集缓存模块，用于将所述压缩后的指令集通过写寄存器的方式存储至所述芯片的指令集缓冲区中，以用于执行对应的任务。

11、根据权利要求 10 所述的装置，其中，所述模型指令编译模块，用于根据应用场景选择匹配的深度学习模型；对选择的深度学习模型进行训练，得到待加速的深度学习模型；基于芯片的架构对待加速的深度学习模型进行结构分析，根据分析结果对待加速的深度学习模型进行编译，得到与芯片匹配的待处理的指令集。

12、根据权利要求 11 所述的装置，其中，所述模型指令编译模块，用于基于芯片的架构对待加速的深度学习模型进行依赖关系分析、模型结构的优化分析以及可变精度定点化分析中的至少一种分析。

13、根据权利要求 12 所述的装置，其中，所述模型指令编译模块，用于基于芯片的架构分析和挖掘待加速的深度学习模型中可流水化的计算及可并行化的计算。

14、根据权利要求 11 至 13 中任一所述的装置，其中，所述待处理的深度学习指令集中的每个指令包括操作码和操作数信息；

所述指令集压缩模块，用于将待处理的深度学习指令集中具有相同操作码的指令进行压缩处理，得到压缩后的指令集。

15、根据权利要求 11 至 13 中任一所述的装置，其中，所述指令集压缩模块，用于获取待处理的深度学习指令集中每条指令的位宽，将位宽小于预设数值的指令进行合并处理，得到压缩后的指令集。

16、一种基于芯片的指令集处理装置，其中，所述装置包括：

第一获取模块，用于获取任务处理请求；

第二获取模块，用于根据所述任务处理请求从指令集缓冲区中获取压缩后的指令集；

指令集译码模块，用于对所述压缩后的指令集进行译码，得到译码后的指令信息；

寄存更新模块，用于将所述译码后的指令信息进行寄存器的寄存更新操作；

流程控制模块，用于根据寄存更新操作控制对应的功能模块执行所述指令信息对应的任务。

17、根据权利要求 16 所述的装置，其中，所述译码后的指令信息包括操作数信息，不同的寄存器保存有不同的操作数信息；

所述寄存更新模块，用于根据译码后的指令信息更新对应的寄存器所保存的操作数信息。

18、根据权利要求 16 或 17 所述的装置，其中，不同功能模块对应不同的寄存器；

所述流程控制模块，用于控制进行寄存更新操作的寄存器所对应的功能模块，根据寄存更新操作后所保存的操作数信息执行对应的功能操作。

19、一种计算机设备，其中，所述计算机设备包含处理器和存储器，所述存储器中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由所述处理器加载并执行以实现如权利要求 1 至 6 任一所述的基于芯片的指令集处理方法或者实现如权利要求 7-9 任一所述的基于芯片的指令集处理方法。

20、一种计算机可读存储介质，其中，所述存储介质中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由处理器加载并执行以实现如权利要求 1 至 6 任一所述的基于芯片的指令集处理方法或者实现如权利要求 7-9 任一所述的基于芯片的指令集处理方法。

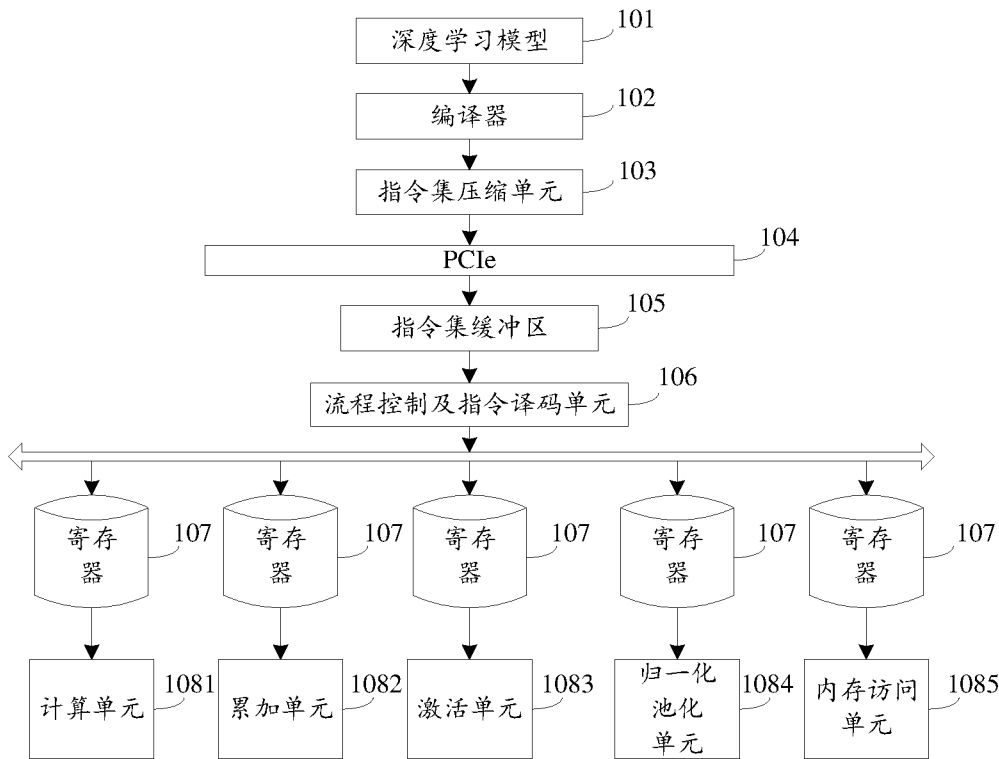


图 1

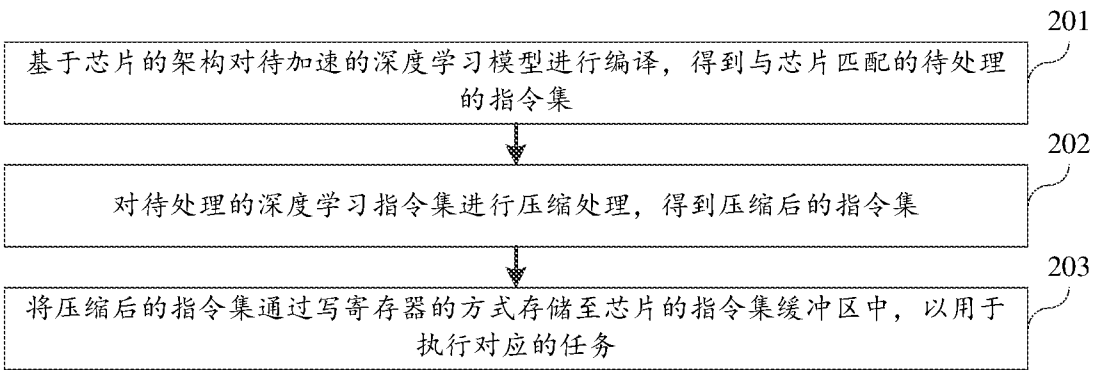


图 2

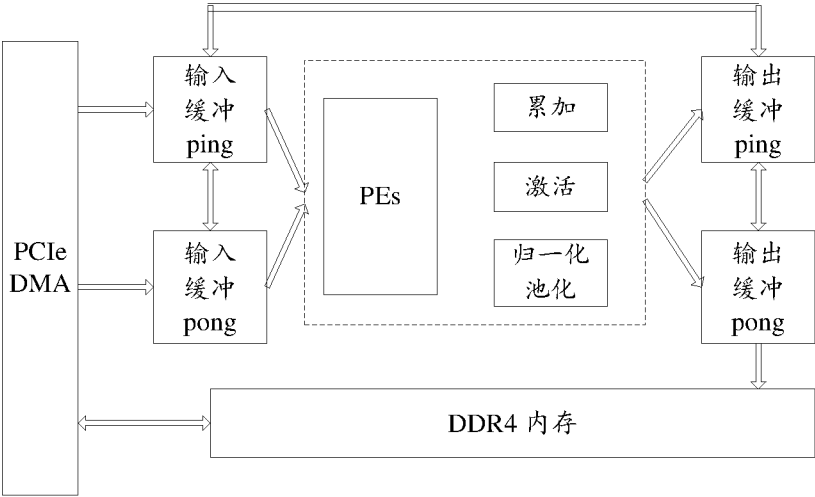


图 3

操作码 1	操作数信息
操作码 2	操作数信息
操作码 3	操作数信息
操作码 4	操作数信息

操作码 5	操作数信息
操作码 2	操作数信息
操作码 3	操作数信息
操作码 6	操作数信息

操作码 1	操作数信息
操作码 2	操作数信息
操作码 3	操作数信息
操作码 4	操作数信息
操作码 5	操作数信息
操作码 6	操作数信息

图 4

3/7

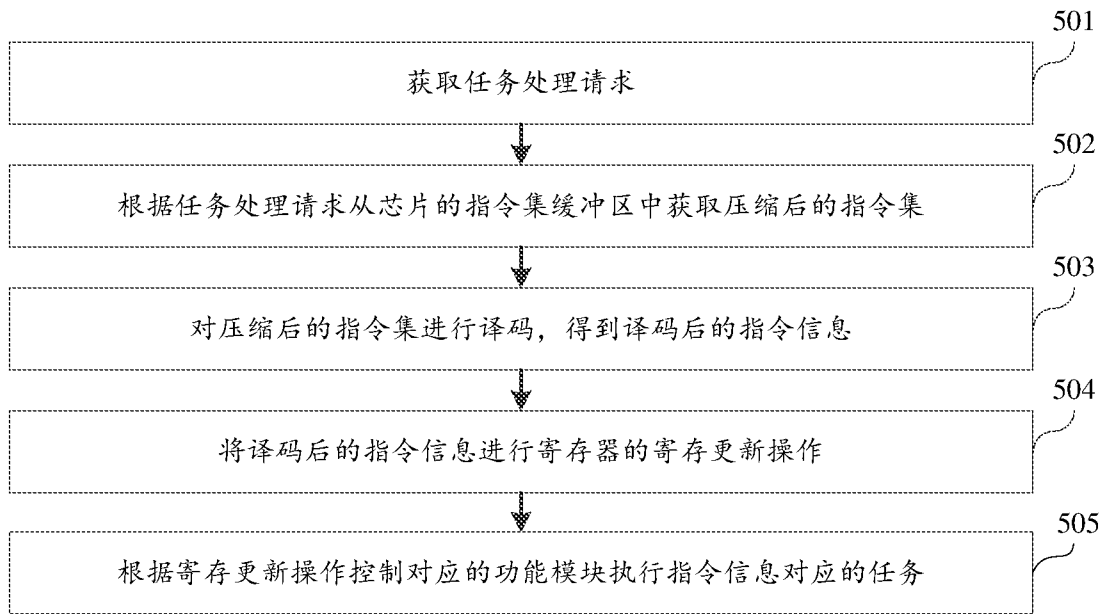


图 5

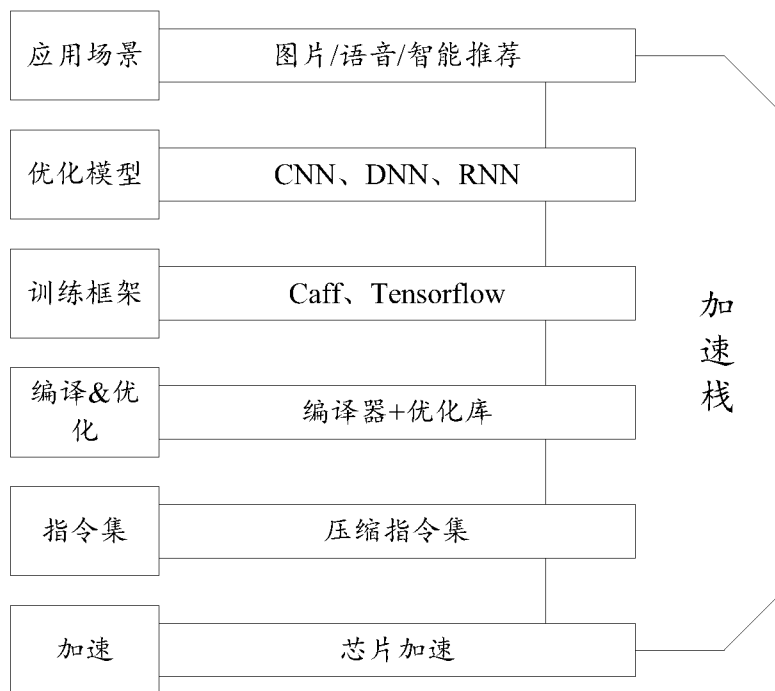


图 6

4/7

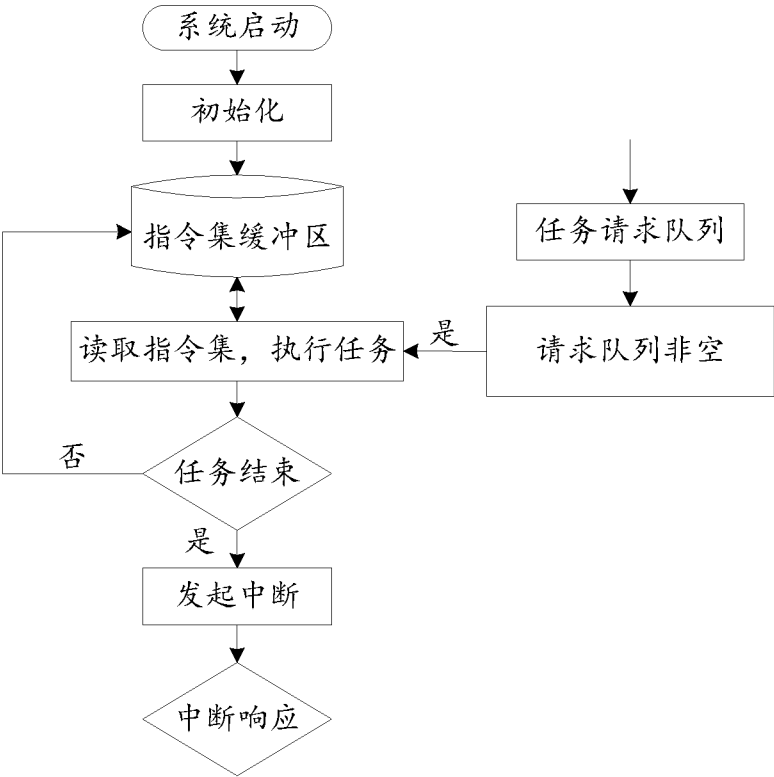


图 7

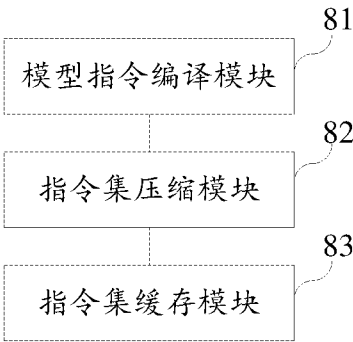


图 8

5/7



图 9

6/7

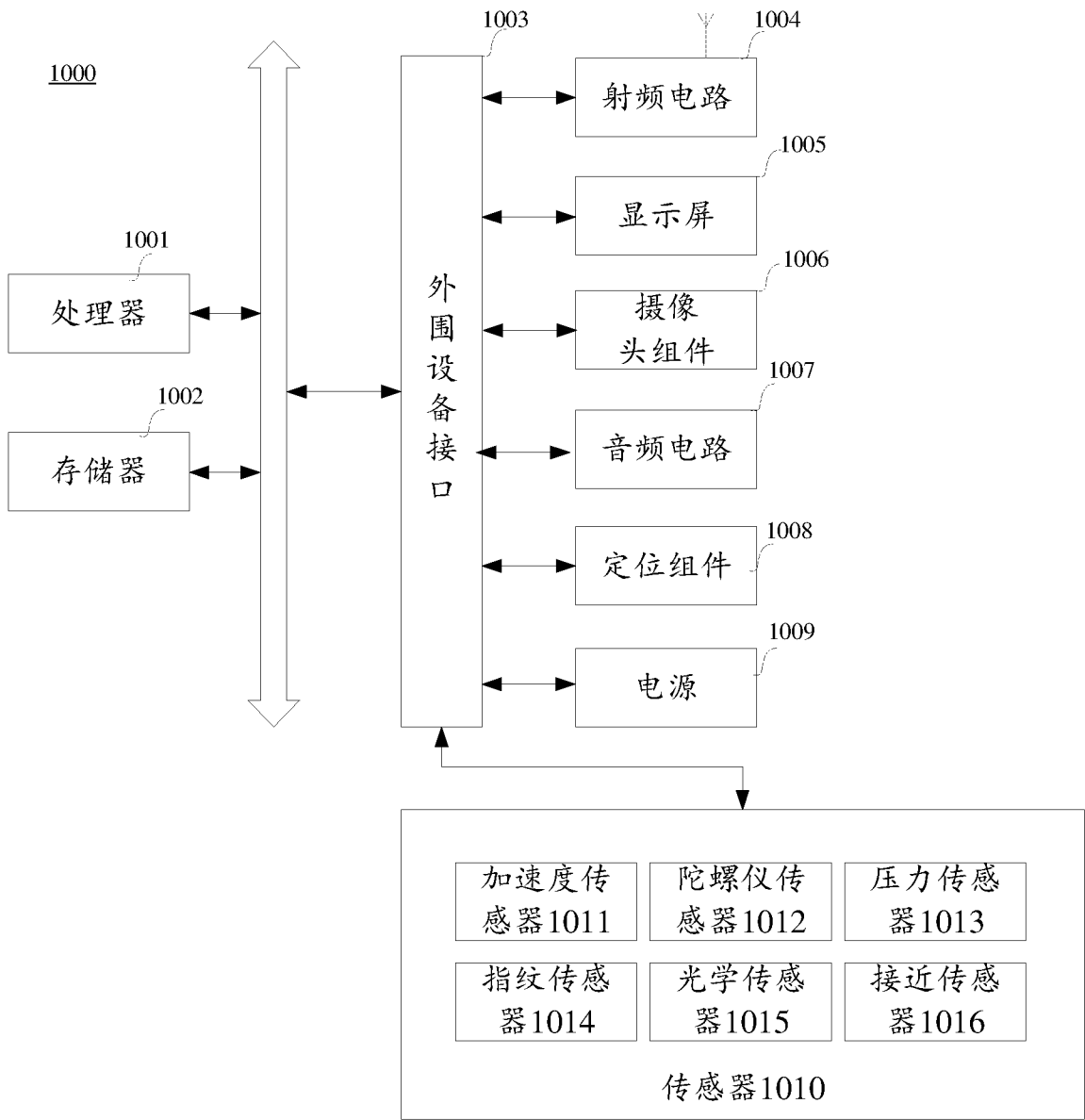


图 10

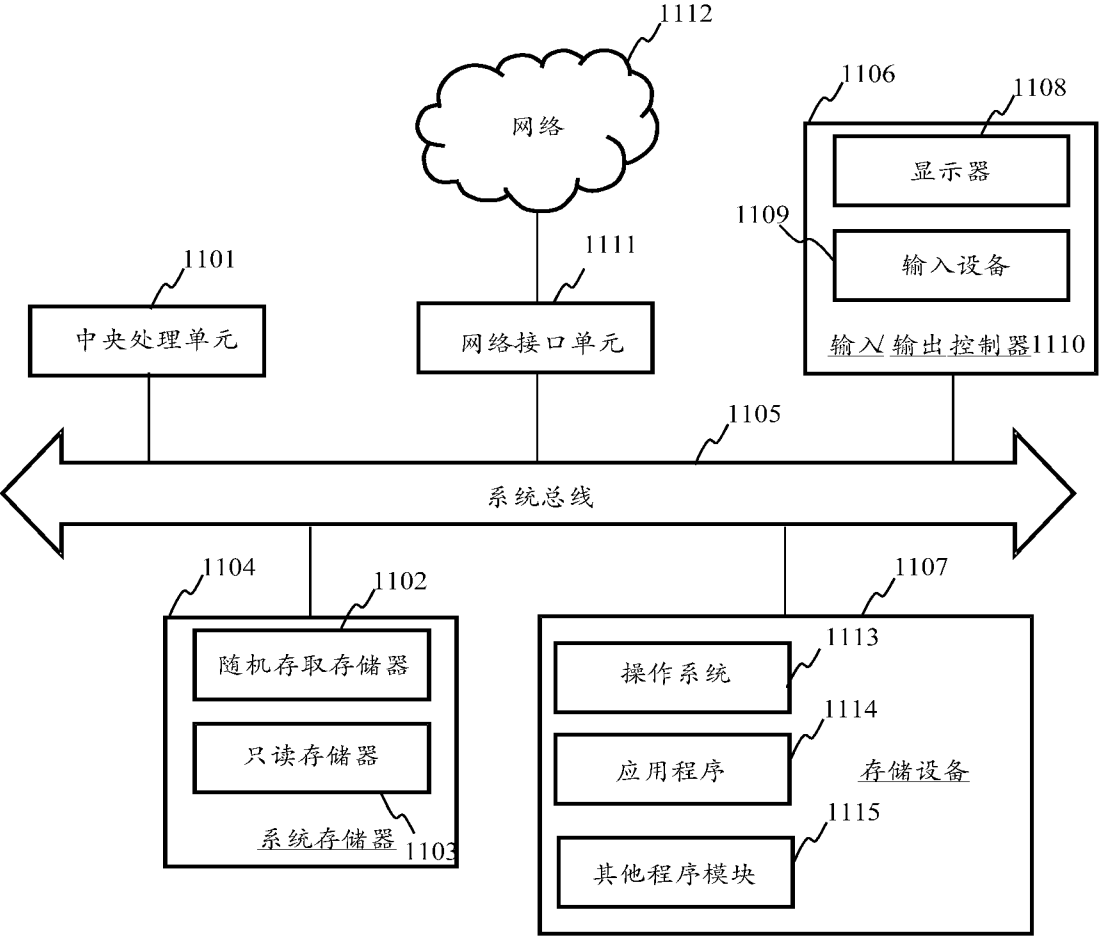


图 11

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2018/119732

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/30(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, WPI, EPODOC, CNKI, IEEE: 芯片, 指令集, 架构, 深度学习, 模型, 编译, 译码, 编译, 寄存器, 压缩, chip, instruction, set, architecture, learning, model, compile, encode, register, compress

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 104025042 A (INTEL CORPORATION) 03 September 2014 (2014-09-03) description, paragraphs [0002]-[0004] and [0012]-[0040], and figures 1-3	1-20
A	CN 107563512 A (TENCENT TECHNOLOGY (SHANGHAI) CO., LTD. ET AL.) 09 January 2018 (2018-01-09) entire document	1-20
A	CN 106022472 A (FUDAN UNIVERSITY) 12 October 2016 (2016-10-12) entire document	1-20
A	US 2010287359 A1 (MIPS TECHNOLOGIES, INC.) 11 November 2010 (2010-11-11) entire document	1-20

☐ Further documents are listed in the continuation of Box C.☒ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

22 February 2019

Date of mailing of the international search report

08 March 2019

Name and mailing address of the ISA/CN

State Intellectual Property Office of the P. R. China
No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing
100088
China

Facsimile No. (86-10)62019451

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2018/119732

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	104025042	A	03 September 2014	EP	2798479	A1	05 November 2014
				TW	201342227	A	16 October 2013
				US	2014082334	A1	20 March 2014
				WO	2013101149	A1	04 July 2013
CN	107563512	A	09 January 2018	None			
CN	106022472	A	12 October 2016	None			
US	2010287359	A1	11 November 2010	CN	101887357	A	17 November 2010
				US	2018173531	A1	21 June 2018
				US	2016147535	A1	26 May 2016

国际检索报告

国际申请号

PCT/CN2018/119732

A. 主题的分类

G06F 9/30 (2006.01) i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

CNPAT, WPI, EPDOC, CNKI, IEEE: 芯片, 指令集, 架构, 深度学习, 模型, 编译, 译码, 编译, 寄存器, 压缩, chip, instruction, set, architecture, learning, model, compile, encode, register, compress

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	CN 104025042 A (英特尔公司) 2014年 9月 3日 (2014 - 09 - 03) 说明书第[0002]-[0004], [0012]-[0040]段, 图1-3	1-20
A	CN 107563512 A (腾讯科技上海有限公司 等) 2018年 1月 9日 (2018 - 01 - 09) 全文	1-20
A	CN 106022472 A (复旦大学) 2016年 10月 12日 (2016 - 10 - 12) 全文	1-20
A	US 2010287359 A1 (MIPS TECHNOLOGIES, INC.) 2010年 11月 11日 (2010 - 11 - 11) 全文	1-20

☐ 其余文件在C栏的续页中列出。☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的)

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

2019年 2月 22日

国际检索报告邮寄日期

2019年 3月 8日

ISA/CN的名称和邮寄地址

中国国家知识产权局(ISA/CN)

中国北京市海淀区蓟门桥西土城路6号 100088

传真号 (86-10)62019451

授权官员

王莹

电话号码 86-(10)-53961411

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2018/119732

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	104025042	A	2014年 9月 3日	EP	2798479	A1	2014年 11月 5日
				TW	201342227	A	2013年 10月 16日
				US	2014082334	A1	2014年 3月 20日
				WO	2013101149	A1	2013年 7月 4日
CN	107563512	A	2018年 1月 9日	无			
CN	106022472	A	2016年 10月 12日	无			
US	2010287359	A1	2010年 11月 11日	CN	101887357	A	2010年 11月 17日
				US	2018173531	A1	2018年 6月 21日
				US	2016147535	A1	2016年 5月 26日