昇腾AI处理器架构

HUAWEI

目录

1. 华为AI全栈解决方案介绍
2. 昇腾AI处理器硬件架构
3. 昇腾AI处理器软件架构

华为AI全栈解决方案

应用使能

HiAI Service | General APIs | Advanced APIs | Pre-integrated Solutions

HiAI Engine | **ModelArts**

全流程服务，分层API和预集成方案

AI框架

端、边、云独立协同的统一训练和推理框架

MindSpore | TensorFlow | PyTorch | PaddlePaddle | …

降低核心代码量**20%**,效率提升**50%**以上

MindSpore

芯片使能

芯片算子库和高度自动化的算子开发工具

CANN* Compute Architecture for Neural Network

应对算子多样性，开发效率提升**3倍**

芯片

基于统一、可扩展架构的系列化AI IP和芯片

终端设备

Nano — Tiny — Lite – Mini

20 MOPS ~ 20 TOPS 1mW ~ 10W

耳机、手机、便携机、摄像头、TV、VR…

边缘计算

Mini / Multi–Mini

10~100 TOPS 3~100W

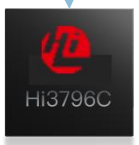
汽车、边缘计算服务器…

云服务

Ascend Max

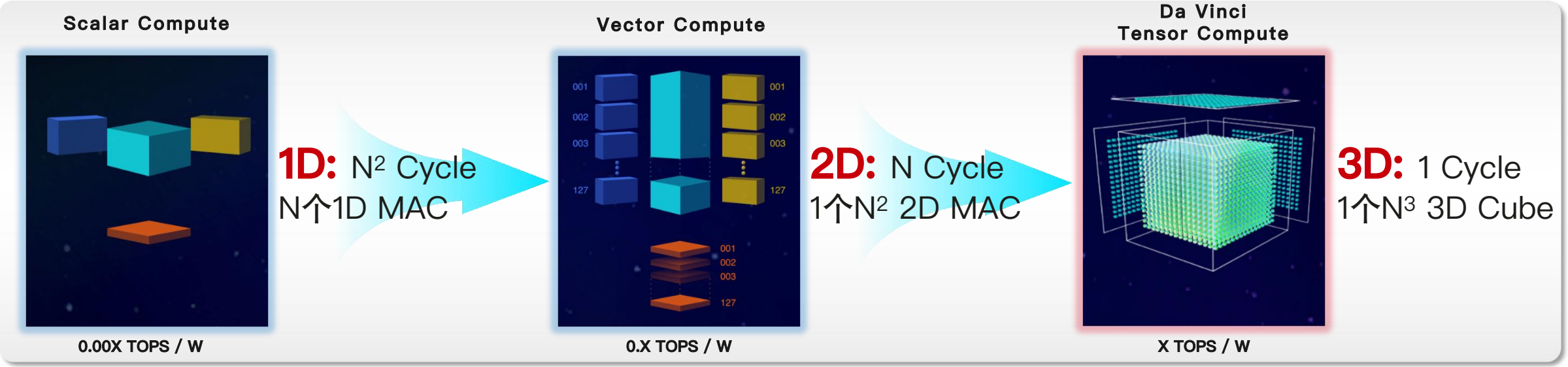
200+ TOPS 200W+

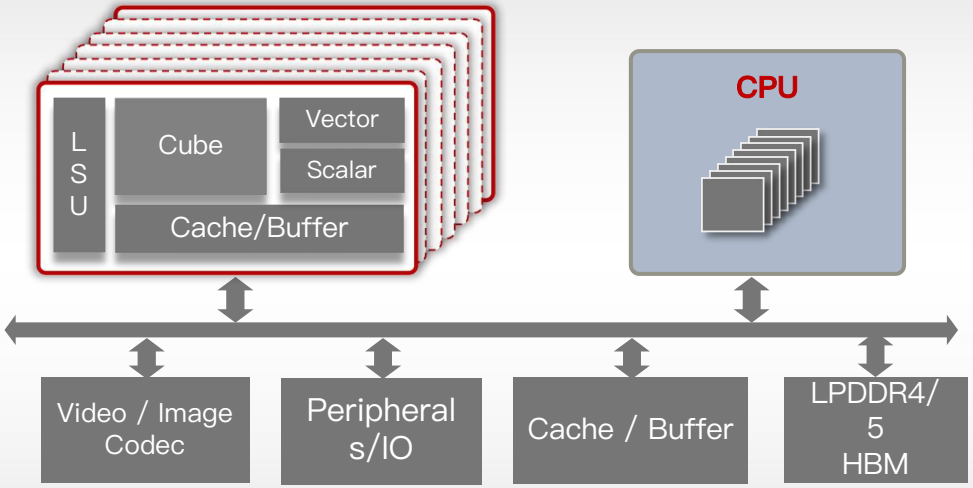
数据中心…



HUAWEI

芯片层：基于Da Vinci AI技术架构





3D Cube:16^3三维弹性立方体

- 高算力：可在一个时钟周期内完成4096个FP16 MAC 运算
- 高效：支持几十毫瓦IP到几百瓦芯片，适应端、边和云的平滑架构扩展

CANN层：高性能算子库和算子开发工具

CANN

Compute Architecture for Neural Networks

Fusion Engine 融合引擎

- Task信息
- 算子融合
- 算子管理

TBE 算子开发工具

- DSL
- TIK

CCE算子库

- 卷积类
- 矩阵乘
- 控制类
- Vector类

CCE Compiler

- Complier Front End
- AI Core
- AI CPU
- CPU

最优开发效率和算子最佳匹配昇腾芯片性能

融合引擎

算子融合，减少算子内存搬移，**提升性能17%**

TBE算子开发工具：预置丰富API接口，用户自定义算子开发和自动化调优，**工期缩短53%，人力投入减少42%**

CCE算子库：基于昇腾处理器，深度协同优化的高性能算子库，**性能提升10%**

CCE编译器

基于C/C++扩展的异构混合编程语言的编译器及二进制工具集，**极致性能、高效编程**，实现昇腾芯片全场景支持

HUAWEI

框架层：支持全场景AI计算的开源框架MindSpore

全场景AI应用生态

MindSpore

全场景统一API

自动微分	自动并行	自动调优

MindSpore IR 计算图表达

On-Device执行	Pipeline并行	深度图优化

端-边-云按需协作分布式架构（部署、调度、通信等）

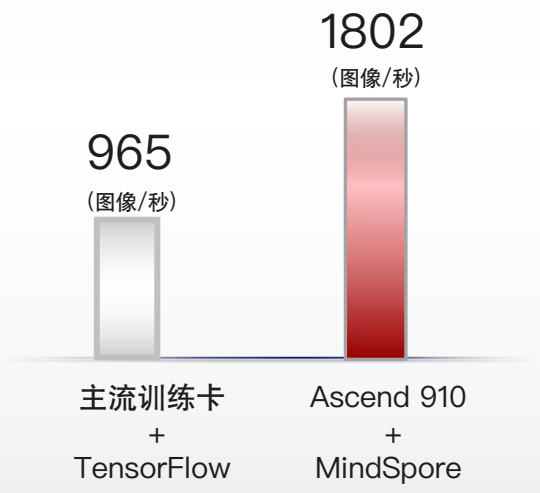
处理器：Ascend、GPU、CPU

MindSpore

开发态友好： AI 算法即代码

运行态高效： 面向昇腾AI处理器优化支持GPU

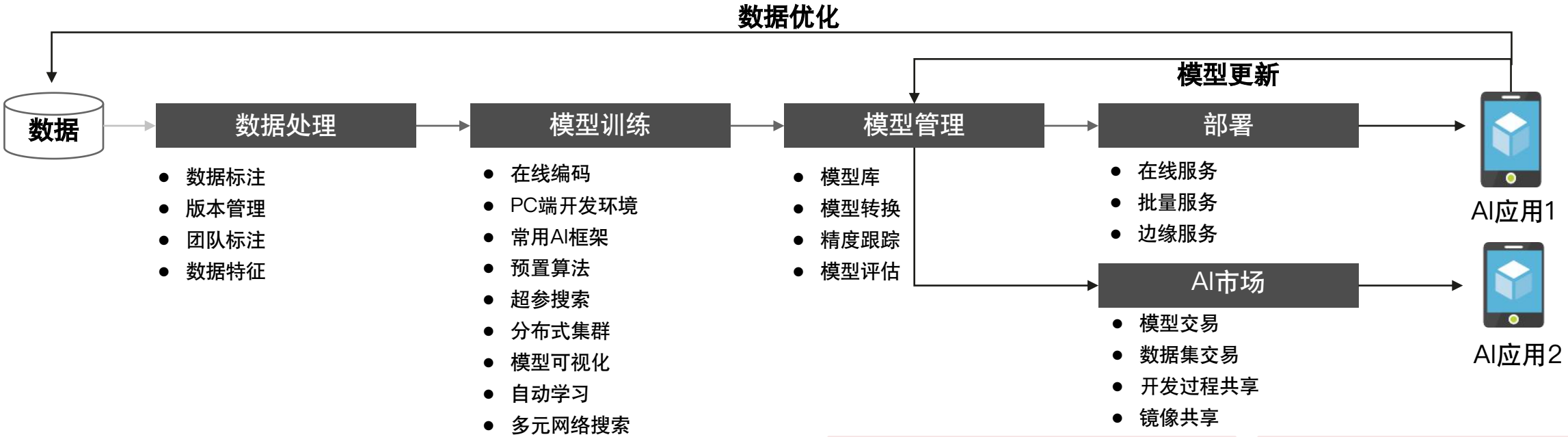
部署态灵活： 全场景按需协同



- ResNet 50 V1.5
- ImageNet 2012
- 按各自最佳的batch size

HUAWEI

应用层：普惠AI+急速性能的开发平台ModelArts



- ❑ 一站式AI平台，覆盖机器学习、深度学习、强化学习应用全流程。
- ❑ 支持分布式并行训练。
- ❑ 交互式Notebook，支持Python开发语言。
- ❑ 内置TensorFlow、Pytorch、MindSpore、Scikit-Learn, XGBoost,等AI引擎库。
- ❑ 支持难例挖掘，AI快速闭环。
- ❑ 模型资产统一管理，便捷应用。

效率高

- ❑ 自动学习，零AI基础零编码，可快速定制化AI模型。
- ❑ 预置50+机器学习算子，包含输入、数据转换、建模、评估、输出等多种类型。
- ❑ 可视化展示数据/模型/评估结果
- ❑ 集成TensorBoard，支持神经网络可视化
- ❑ 预置10+深度学习高精度模型，包含图像识别、物体检测、分割、NLP等多种常用类型

门槛低

- ❑ 自研MoXing和AIBox深度学习训练和推理高性能加速框架。
- ❑ 异构资源统一管理，海量任务容器化秒级调度。
- ❑ 支持分布式并行训练，1000GPU线性加速比可达0.8。
- ❑ 支持推理加速，提速可达x倍。
- ❑ 支持模型转换压缩，可部署端侧设备。
- ❑ 在线推理高并发场景实现任务动态扩缩容，负载均衡，高可用。

性能优

- ❑ 离线SDK/Pycharm插件，可本地PC调用。
- ❑ 模型仓库支持多厂商多框架多功能模型统一纳管。
- ❑ 服务详情监控（CPU/MEM/GPU等）
- ❑ 灰度发布，支持A/B Test。
- ❑ 模型滚动在线升级，业务不间断。
- ❑ 支持标准RestFul API，作为AI底座易于业务AI平台封装。
- ❑ 模型评估诊断,易于模型快速评估上线

运维易

目录

1. 华为AI全栈解决方案介绍
2. 昇腾AI处理器硬件架构
   - 概述
   - 达芬奇架构的AI Core
   - 处理器硬件架构（Ascend 310/Ascend 910）
3. 昇腾AI处理器软件架构

概述 —— 昇腾AI处理器现状

Ascend 310

- Ascend-Mini
- 架构: 达芬奇
- 半精度 (FP16): 8 Tera-FLOPS
- 整数精度 (INT8) : 16 Tera-OPS
- 16 通道 全高清 视频解码器 — H.264/265;
- 1 通道 全高清 视频编码器 — H.264/265;
- 最大功耗: 8W
- 12nm FFC

Ascend 910

- Ascend-Max
- 架构: 达芬奇
- 半精度 (FP16): 256 Tera-FLOPS
- 整数精度 (INT8) : 512 Tera-OPS
- 128 通道 全高清 视频解码器 — H.264/265;
- 最大功耗: 310W
- 7nm

当前已发布的有两款处理器：

Ascend 310 —— 面向推理场景

Ascend 910 —— 面向训练场景

本节内容将基于Ascend 310/910两款处理器深入介绍：

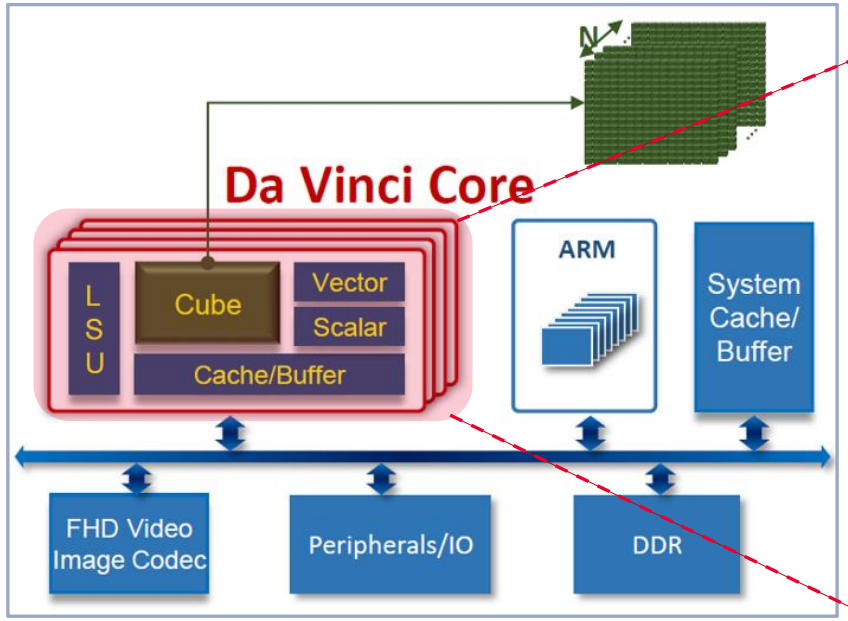
- 昇腾处理器的核心算力部件：达芬奇架构的AI Core
- Ascend 310、Ascend910的硬件架构
- 软件逻辑架构（适用Ascend 310和910）

HUAWEI

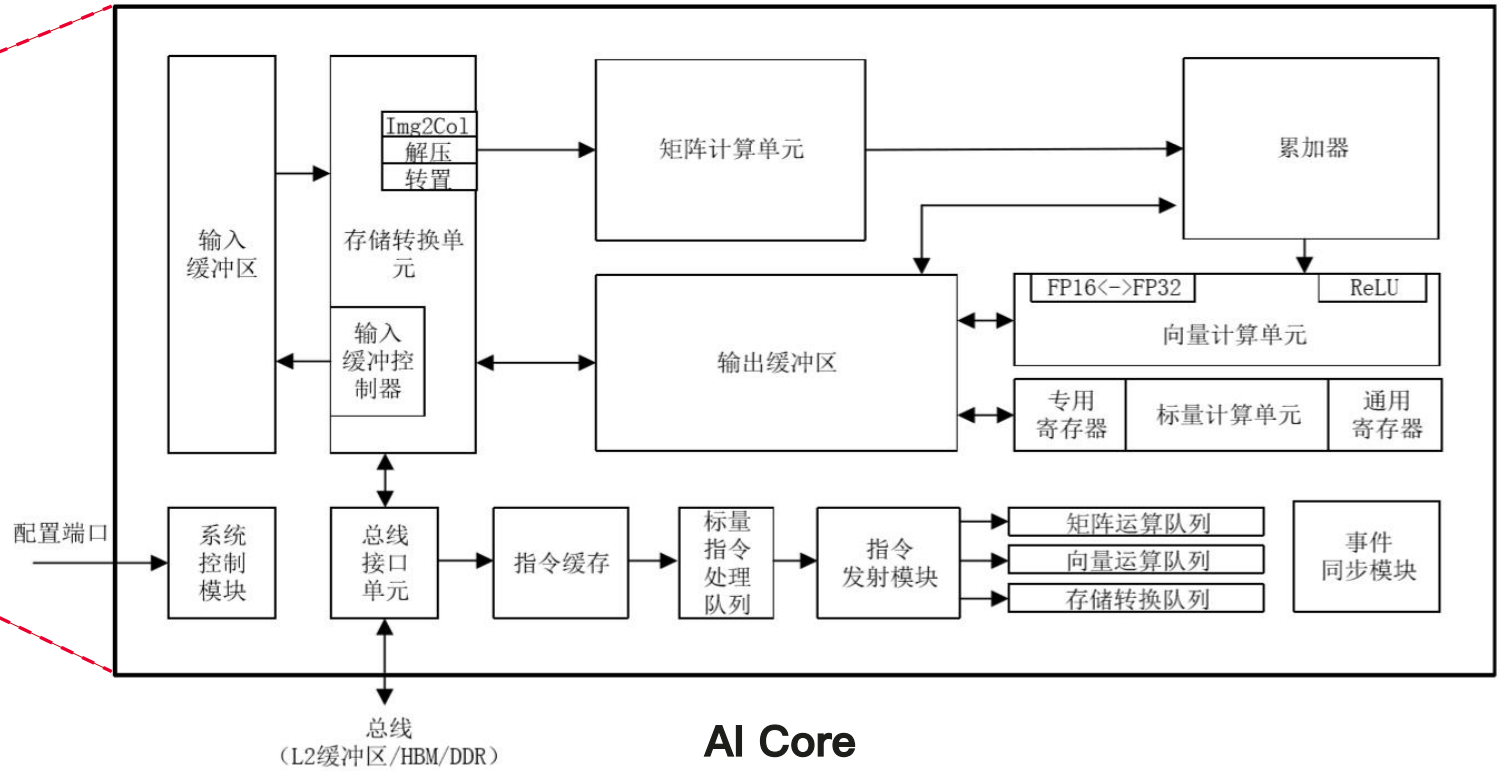
目录

1. 华为AI全栈解决方案介绍
2. 昇腾AI处理器硬件架构
   - 概述
   - 达芬奇架构的AI Core
   - 处理器硬件架构（Ascend 310/Ascend 910）
3. 昇腾AI处理器软件架构

AI Core：在昇腾处理器中的位置



昇腾处理器



AI Core

AI Core是昇腾AI处理器的计算核心，采用华为自研的达芬奇架构，通常也被叫做Da Vinci Core。

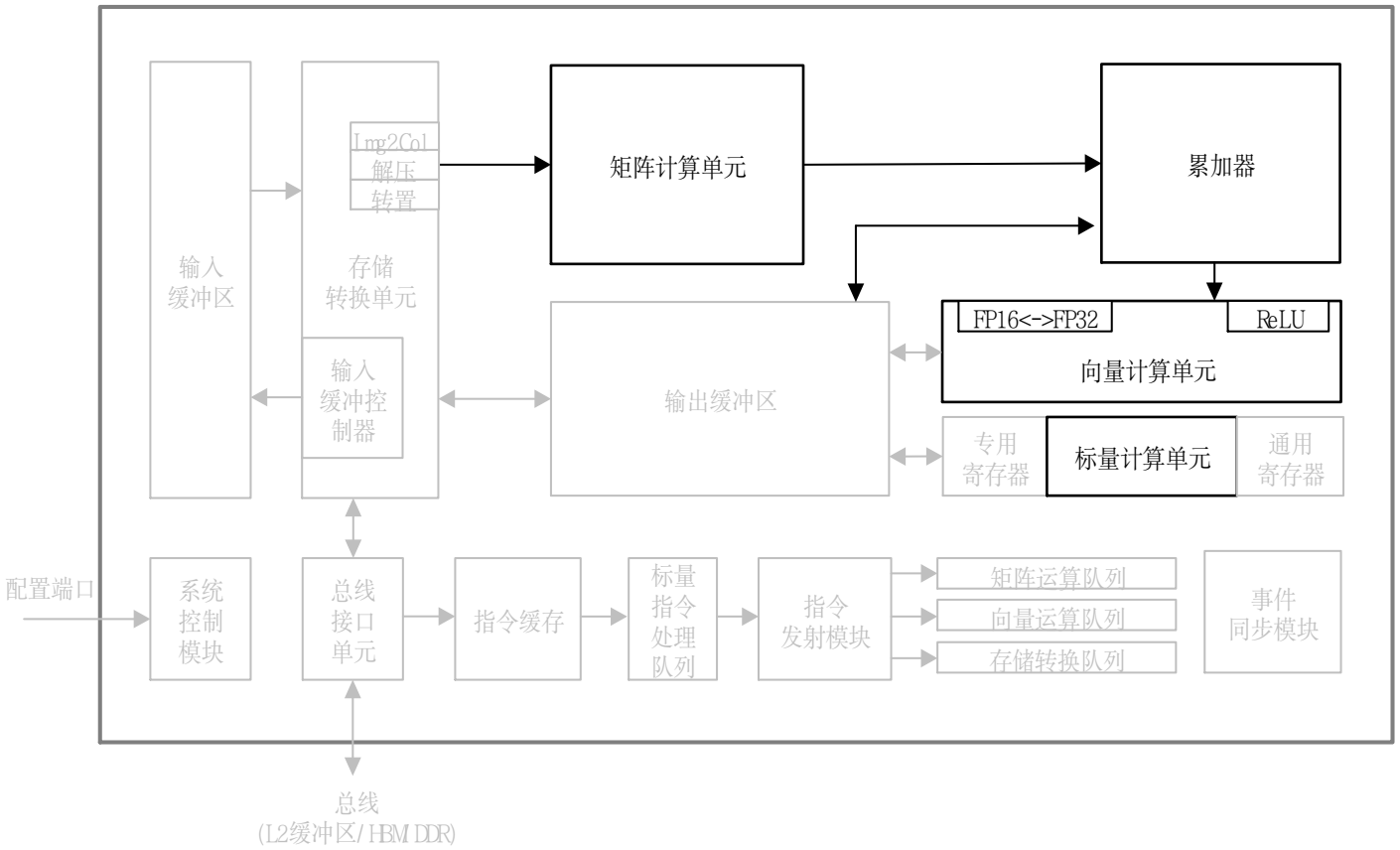
根据不同的处理器版本，AI Core里的计算、存储和带宽资源有不同的规格，在本课程中，主要是按照Ascend310/910 来介绍。

达芬奇架构主要部分：

- **计算单元：** 包含三种基础计算资源（矩阵计算单元、向量计算单元、标量计算单元）
- **存储系统：** AI Core的片上存储单元和相应的数据通路构成了存储系统。
- **控制单元：** 整个计算过程提供了指令控制，相当于AI Core的司令部，负责整个AI Core的运行。

AI Core：计算单元

三种基础计算资源：矩阵计算单元、向量计算单元和标量计算单元，分别对应矩阵、向量和标量三种常见的计算模式。



- **矩阵计算单元（Cube Unit）：**

  矩阵计算单元和累加器主要完成矩阵相关运算。一拍完成一个fp16的 16x16与16x16矩阵乘（4096）； 如果是int8输入，则一拍完成 16*32 与 32*16 矩阵乘（8192）。

- **向量计算单元（Vector Unit）：**

  实现向量和标量，或双向量之间的计算，功能覆盖各种基本的计算类型和许多定制的计算类型，主要包括FP16/FP32/Int32/Int8等数据类型的计算。一拍可以完成两个128长度fp16类型的向量相加/乘， 或者64个fp32/int32类型的向量相加/乘。

- **标量计算单元（Scalar Unit）：**

  相当于一个微型CPU，控制整个AI Core的运行，完成整个程序的循环控制、分支判断，可以为Cube/Vector提供数据地址和相关参数的计算，以及基本的算术运算。

- **累加器：**

  把当前矩阵乘的结果与前次计算的中间结果相加， 可以用于完成卷积中加bias操作。

AI Core：计算单元 —— 加速原理

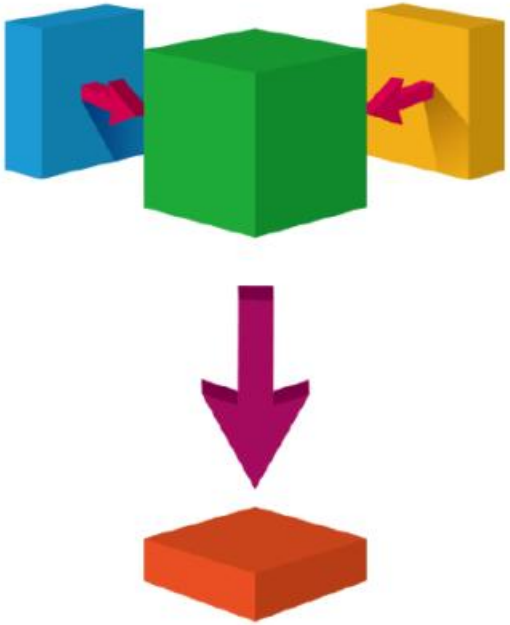
Scalar Unit
Full Flexibility Computation

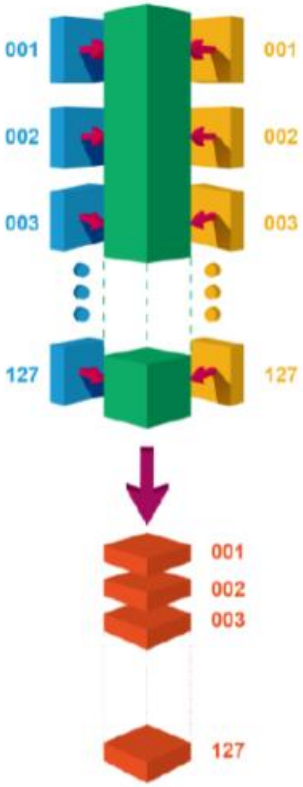
+

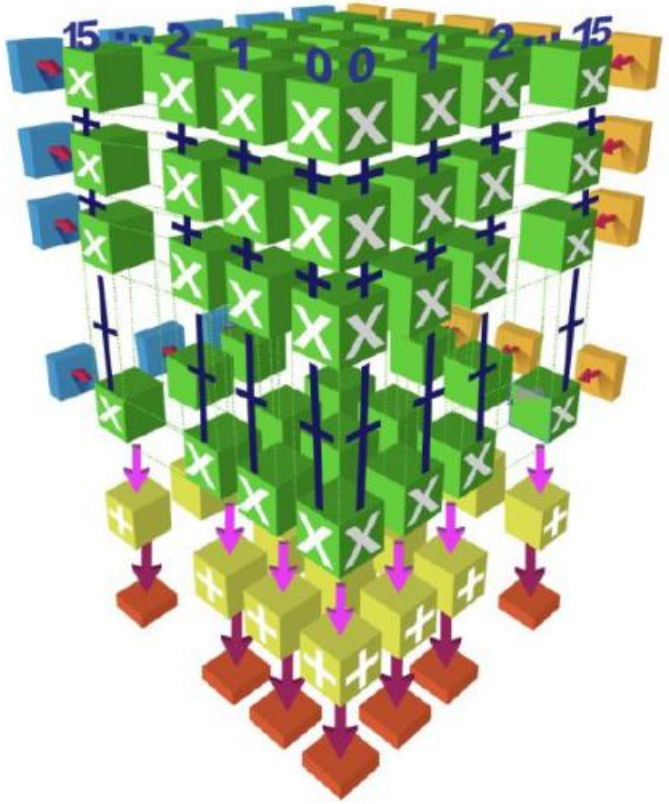
Vector Unit
Rich & Efficient Operations

+

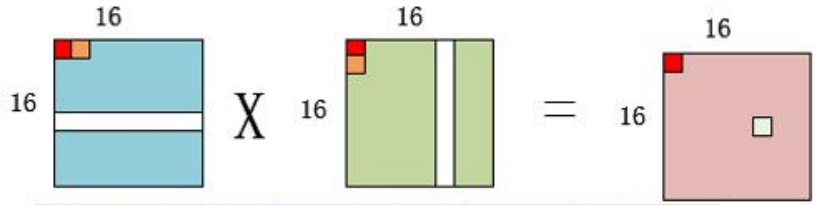
Cube Unit
High Intensity Computation







AI Core：计算单元 —— Cube Unit




```
float a[16][16], b[16][16], c[16][16];
```

CPU:

```
for(int i=0; i< 16; i++)
    for (int j=0; j<16; j++)
        for(int k=0; k<16; k++) {
            c[i][j] += a[i][k] * b[k][j];
        }
```

Cycle=16*16*16*2 = 8192
DataNum per cycle: Rd 2, Wr 1

算力密度高

灵活

Vector:

```
for(int i=0; i<16; i++)
    for ( int j =0; j<16; j++) {
        c[i][j] = a[i][:] *+  b[:][j]
    }
```

Cycle=16*16 = 256
DataNum per cycle: Rd 2*16, Wr 16

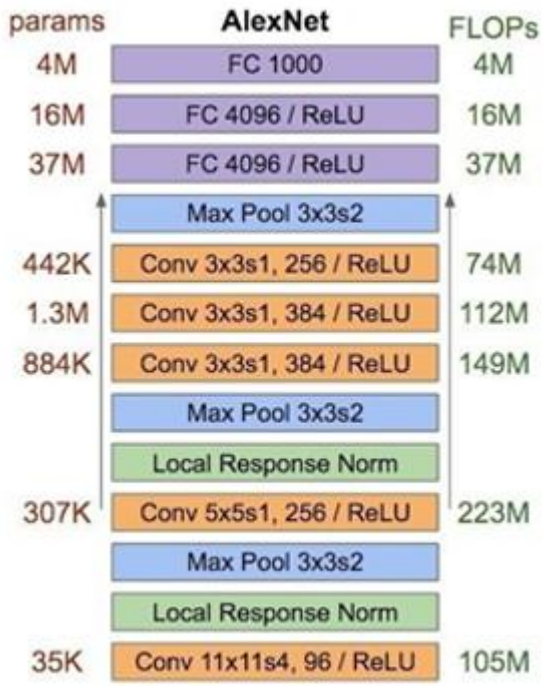
CUBE:

```
CUBE:  c[:][:] = a[:][:] X b[:][:]
```

Cycle=1
DataNum per cycle: Rd 2*16*16
Wr:16*16

上图示例为一个矩阵a和另一个矩阵b之间的乘法运算c=a*b。

在不同的计算单元中实现该矩阵乘，其复杂度和运算效率差别很大。



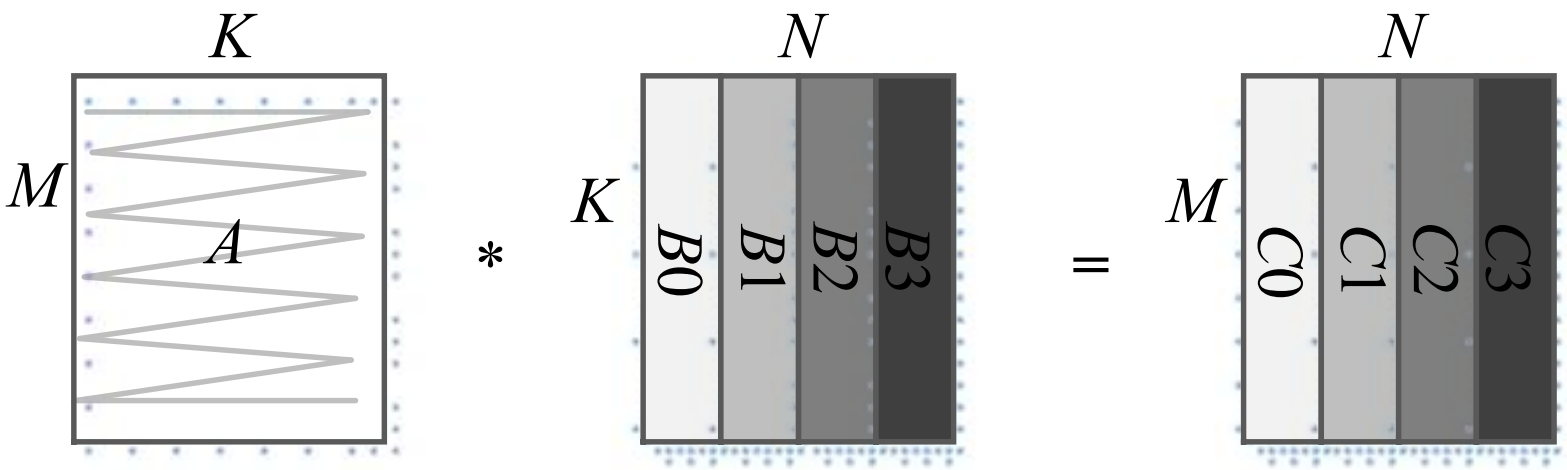
AlexNet模型每层每秒浮点运算次数及参数数量

CNN经典模型的内存，计算量和参数数量对比			
	AlexNet	VGG16	Inception-v3
模型内存(MB)	> 200	> 500	90–100
参数（百万）	60	138	23.2
计算量（百万）	720	15300	5000

从计算量角度，99%以上计算都是矩阵乘

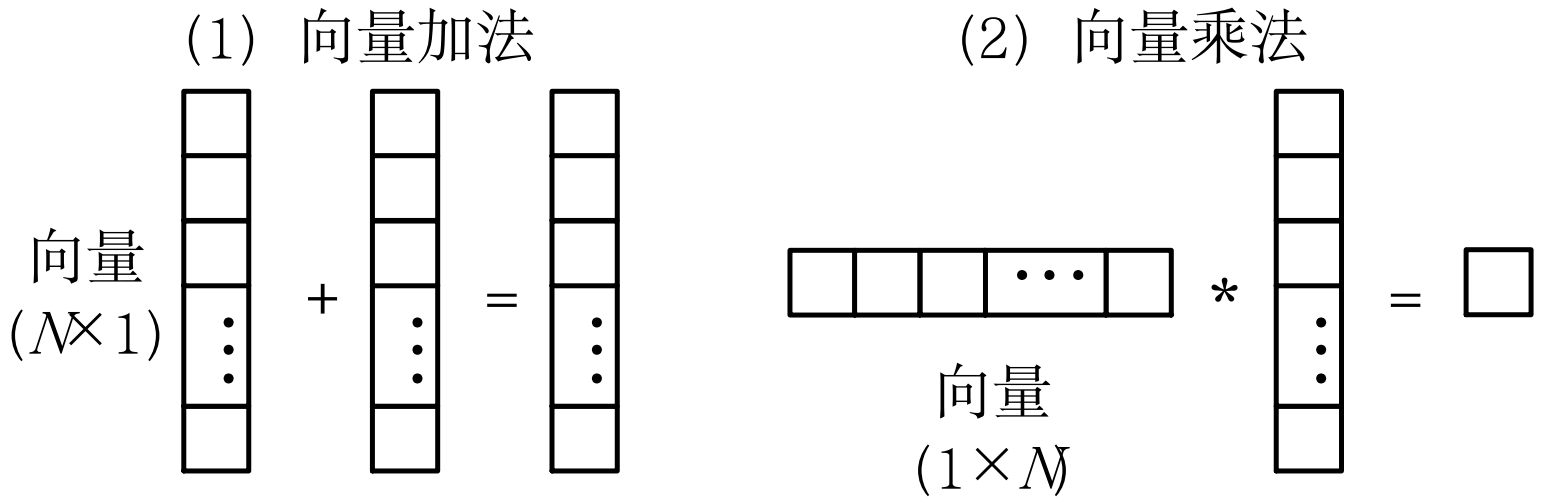
HUAWEI

AI Core：计算单元 —— Cube Unit（矩阵分块计算）



- 一般在矩阵较大时，由于芯片上计算和存储资源有限，往往需要对矩阵进行分块平铺处理（Tiling）。受限于片上缓存的容量，当一次难以装下整个矩阵B时，可以将矩阵B划分成为B0、B1、B2和B3等多个子矩阵。而每一个子矩阵的大小都可以适合一次性存储到芯片上的缓存中并与矩阵A进行计算从而得到结果子矩阵。这样做的目的是充分利用数据的局部性原理，尽可能的把缓存中的子矩阵数据重复使用完毕并得到所有相关的子矩阵结果后再读入新的子矩阵开始新的周期。如此往复可以依次将所有的子矩阵都一一搬运到缓存中，并完成整个矩阵计算的全过程，最终得到结果矩阵C。

AI Core：计算单元 —— Vector Unit



- AI Core中的向量计算单元主要负责完成和向量相关的运算，能够实现向量和标量，或双向量之间的计算，功能覆盖各种基本和多种定制的计算类型，主要包括FP32、FP16、INT32和INT8等数据类型的计算。
- 如上图所示，向量计算单元可以快速完成两个FP16类型的向量相加或者相乘。向量计算单元的源操作数和目的操作数通常都保存在输出缓冲器中。对向量计算单元而言，输入的数据可以不连续，这取决于输入数据的寻址模式。

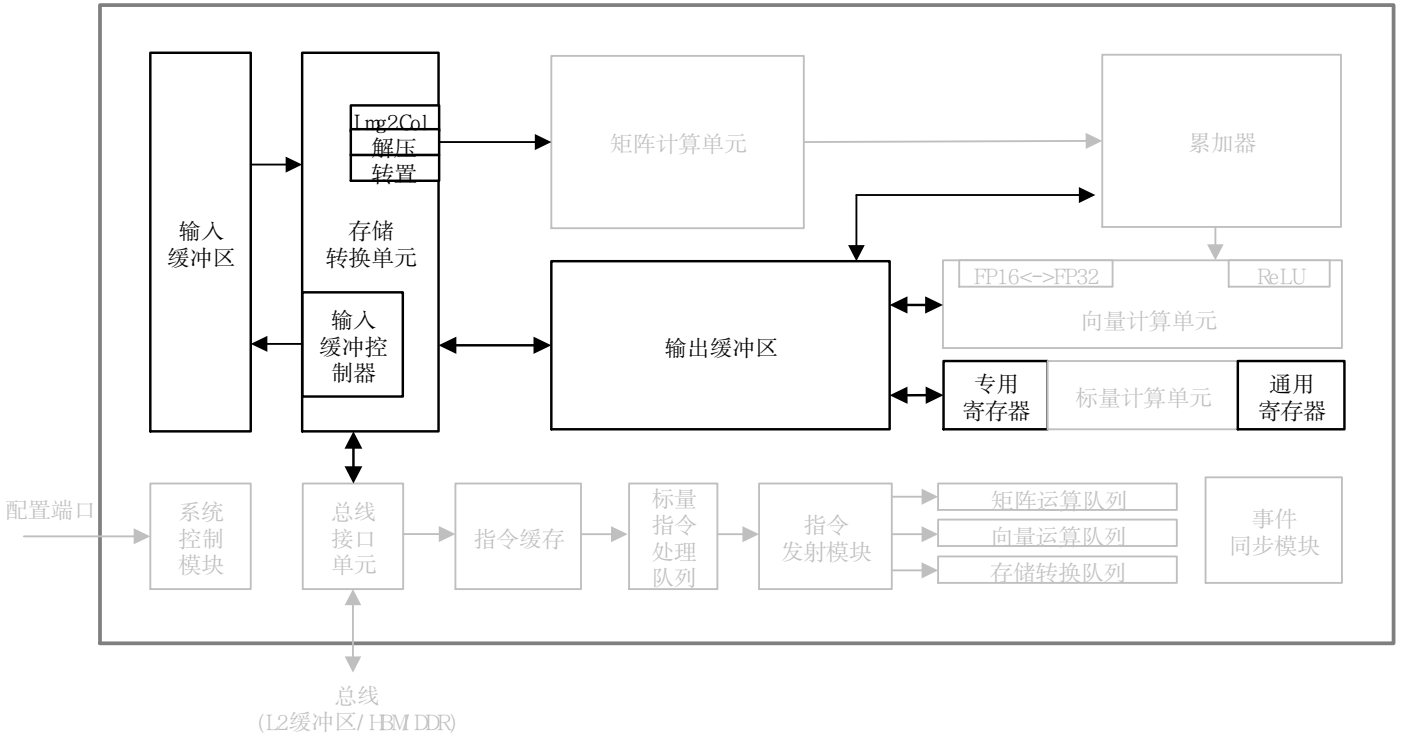
AI Core：计算单元 —— Scalar Unit

- 标量计算单元负责完成AI Core中与标量相关的运算。它相当于一个微型CPU，控制整个AI Core的运行。标量计算单元可以对程序中的循环进行控制，可以实现分支判断，其结果可以通过在事件同步模块中插入同步符的方式来控制AI Core中其它功能性单元的执行流水。它还为矩阵计算单元或向量计算单元提供数据地址和相关参数的计算，并且能够实现基本的算术运算。其它复杂度较高的标量运算则由专门的AI CPU通过算子完成。

- 在标量计算单元周围配备了多个通用寄存器（General Purpose Register，GPR）和专用寄存器（Special Purpose Register，SPR）。这些通用寄存器可以用于变量或地址的寄存，为算术逻辑运算提供源操作数和存储中间计算结果。专用寄存器的设计是为了支持指令集中一些指令的特殊功能，一般不可以直接访问，只有部分可以通过指令读写。

AI Core：存储系统

AI Core采用了大容量的片上缓冲区设计，通过增大的片上缓存数据量来减少数据从片外存储系统搬运到AI Core中的频次，从而可以降低数据搬运过程中所产生的功耗，有效控制了整体计算的能耗。 **存储单元和相应的数据通路，构成了AI Core的存储系统。**



- **数据通路：是指AI Core在完成一次计算任务时，数据在AI Core中的流通路径。**

  达芬奇架构数据通路的特点是**多进单出**，主要是考虑到神经网络在计算过程中，输入的数据种类繁多并且数量巨大，可以通过并行输入的方式来提高数据流入的效率。与此相反，将多种输入数据处理完成后往往只生成输出特征矩阵，数据种类相对单一，单输出的数据通路，可以节约芯片硬件资源。

存储单元由存储控制单元、缓冲区和寄存器组成。

- **存储控制单元：**

  通过总线接口直接访问AI Core之外的更低层级的缓存，也可以直通到DDR或HBM直接访问内存。其中还设置了**存储转换单元**，作为AI Core内部数据通路的传输控制器，负责AI Core内部数据在不同缓冲区之间的读写管理，以及完成一系列的格式转换操作，如补零，Img2Col，转置、解压缩等；

- **输入缓冲区：**

  用来暂时保留需要频繁重复使用的数据，不需要每次都通过总线接口到AI Core的外部读取，从而在减少总线上数据访问频次的同时也降低了总线上产生拥堵的风险，达到节省功耗、提高性能的效果；

- **输出缓冲区：**

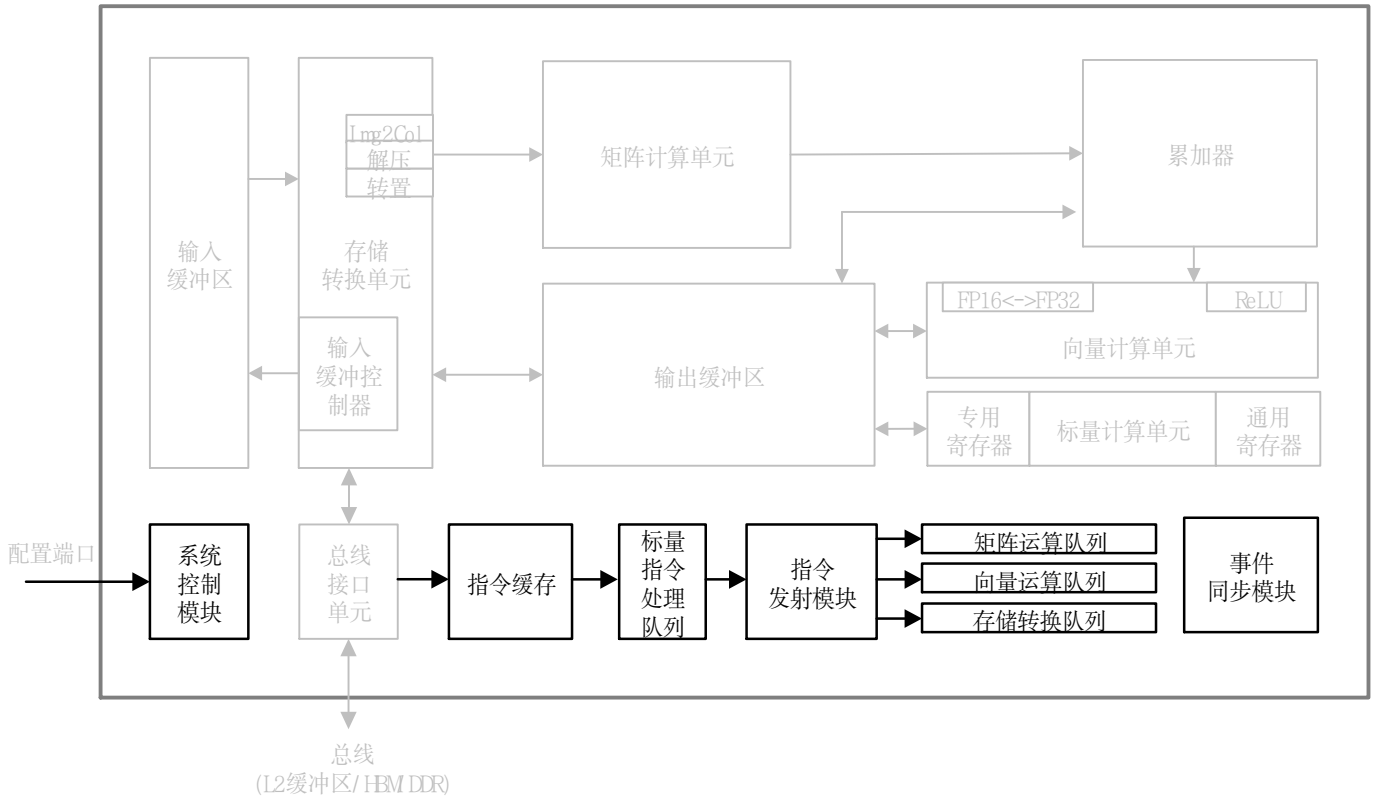
  用来存放神经网络中每层计算的中间结果，从而在进入下一层计算时方便的获取数据。相比较通过总线读取数据的带宽低，延迟大，通过输出缓冲区可以大大提升计算效率；

- **寄存器：**

  AI Core中的各类寄存器资源主要是标量计算单元在使用。

AI Core：控制单元

控制单元主要组成部分为系统控制模块、指令缓存、标量指令处理队列、指令发射模块、矩阵运算队列、向量运算队列、存储转换队列和事件同步模块。



- **系统控制模块：**
  控制任务块（AI Core最小任计算务粒度）的执行进程，在任务块执行完成后，系统控制模块会进行中断处理和状态申报。如果执行过程出错，会把执行的错误状态报告给任务调度器；
- **指令缓存：**
  在指令执行过程中，可以提前预取后续指令，并一次读入多条指令进入缓存，提升指令执行效率；
- **标量指令处理队列：**
  指令被解码后便会被导入标量队列中，实现地址解码与运算控制，这些指令包括矩阵计算指令、向量计算指令以及存储转换指令等；
- **指令发射模块：**
  读取标量指令队列中配置好的指令地址和参数解码，然后根据指令类型分别发送到对应的指令执行队列中，而标量指令会驻留在标量指令处理队列中进行后续执行；
- **指令执行队列：**
  指令执行队列由矩阵运算队列、向量运算队列和存储转换队列组成，不同的指令进入相应的运算队列，队列中的指令按进入顺序执行；
- **事件同步模块：**
  时刻控制每条指令流水线的执行状态，并分析不同流水线的依赖关系，从而解决指令流水线之间的数据依赖和同步的问题。

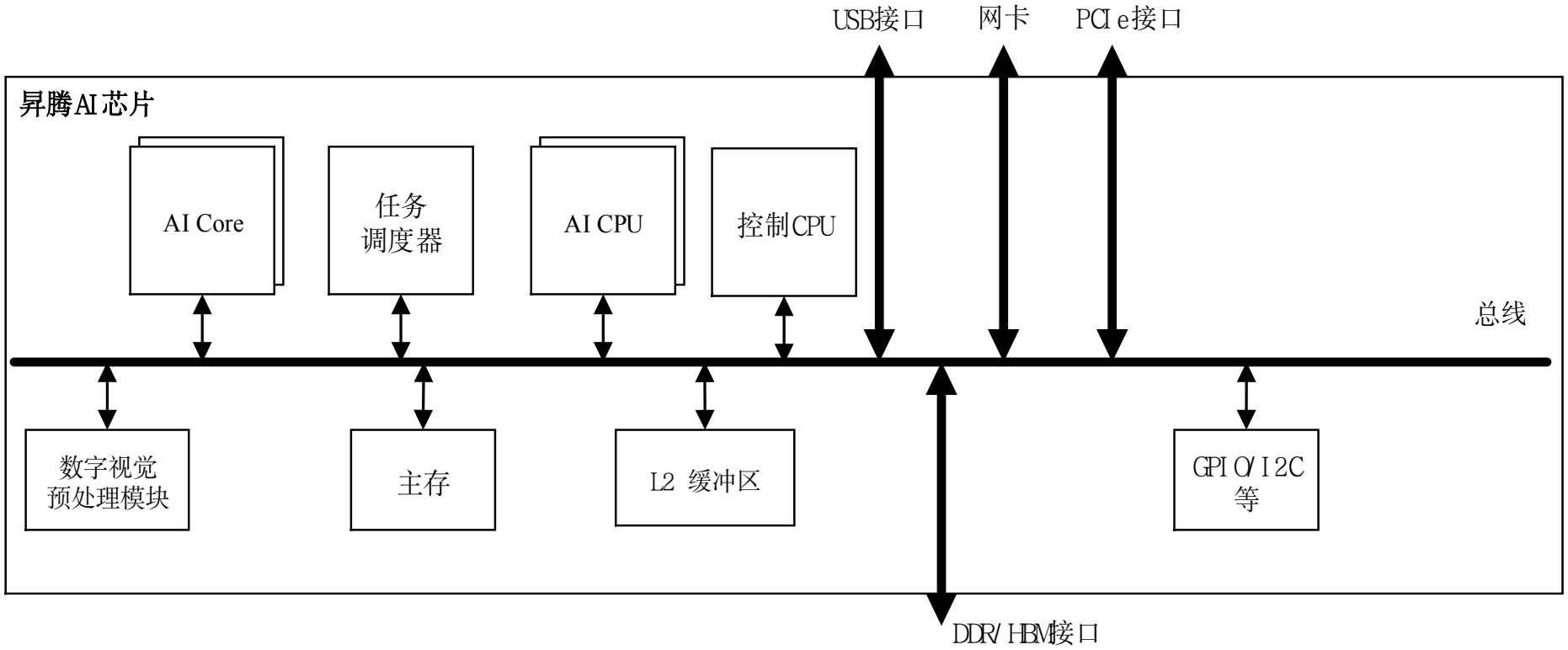
目录

1. 华为AI全栈解决方案介绍
2. 昇腾AI处理器硬件架构
   - 概述
   - 达芬奇架构的AI Core
   - 处理器硬件架构（Ascend 310/Ascend 910）
3. 昇腾AI处理器软件架构

Ascend 310处理器逻辑架构(AI Inference SoC)

Ascend 310处理器的主要架构组成：

- AI计算引擎（包括AI Core和AI CPU）
- 芯片系统控制CPU（Control CPU）
- 多层级的片上系统缓存（Cache）或缓冲区（Buffer）
- 数字视觉预处理模块（Digital Vision Pre-Processing，DVPP）等



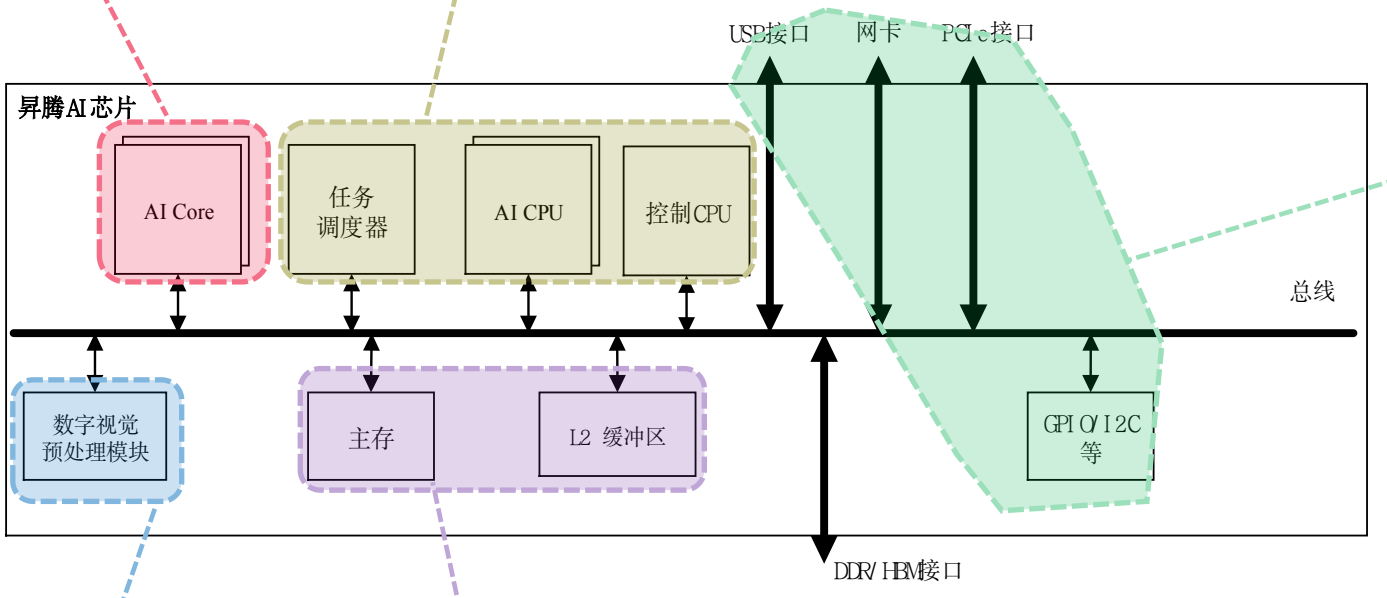
Ascend 310处理器逻辑架构(AI Inference SoC)

AI Core

昇腾AI芯片的计算核心，负责执行矩阵、向量、标量计算密集的算子任务，采用达芬奇架构。Ascend 310集成了2个AI Core。

ARM CPU核心

集成了8个ARM A55。其中一部分部署为**AI CPU**，负责执行不适合跑在AI Core上的算子（承担非矩阵类复杂计算）；一部分部署为专用于控制芯片整体运行的**控制CPU**。两类任务占用的CPU核数可由软件根据系统实际运行情况动态分配。此外，还部署了一个专用CPU作为任务调度器（**Task Scheduler**，TS），以实现计算任务在AI Core上的高效分配和调度；该CPU专门服务于AI Core和AI CPU，不承担任何其他的事务和工作。



对外接口

支持PCIE3.0、RGMII、USB3.0等高速接口、以及GPIO、UART、I2C、SPI等低速接口。

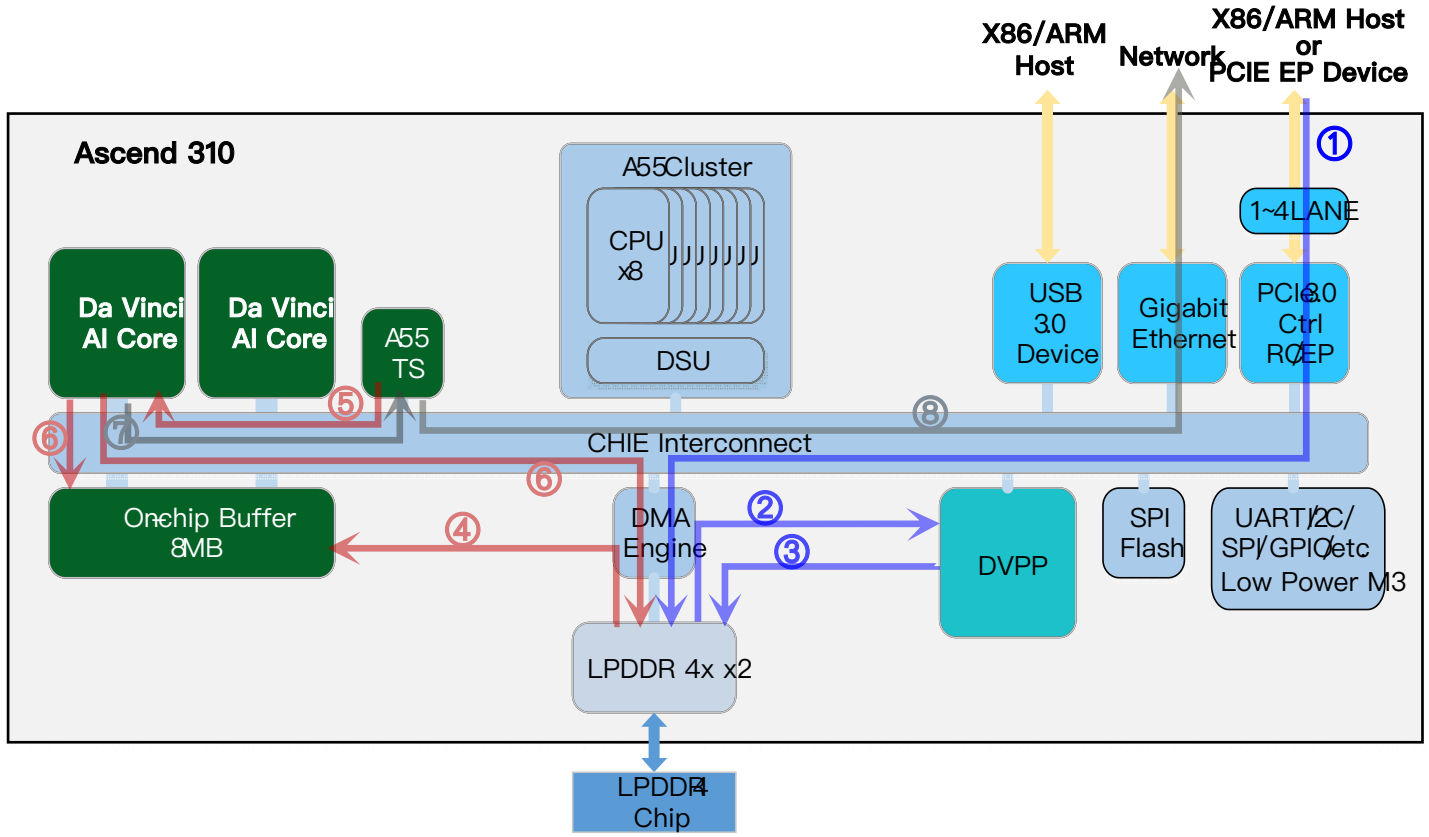
DVPP

数字视觉预处理子系统，完成图像视频的编解码。用于将从网络或终端设备获得的视觉数据，进行预处理以实现格式和精度转换等要求，之后提供给AI计算引擎。

Cache & Buffer

SOC片内有层次化的memory结构，AI core内部有两级memory buffer，SOC片上还有8MB L2 buffer，专用于AI Core、AI CPU，提供高带宽、低延迟的memory访问。芯片还集成了LPDDR4x控制器，为芯片提供更大容量的DDR内存。

数据流图参考 —— 以人脸识别推理应用为例



1、Camera数据采集和处理

① 从摄像头传入压缩视频流，通过PCIE存储至DDR内存中。

② DVPP将压缩视频流读入缓存。

③ DVPP经过预处理，将解压缩的帧写入DDR内存。

2、对数据进行推理

④ 任务调度器（TS）向直接存储访问引擎（DMA）发送指令，将AI资源从DDR预加载到片上缓冲区。

⑤ 任务调度器（TS） 配置AI Core以执行任务。

⑥ AI Core工作时，它将读取特征图和权重并将结果写入DDR或片上缓冲区。

3、人脸识别结果输出

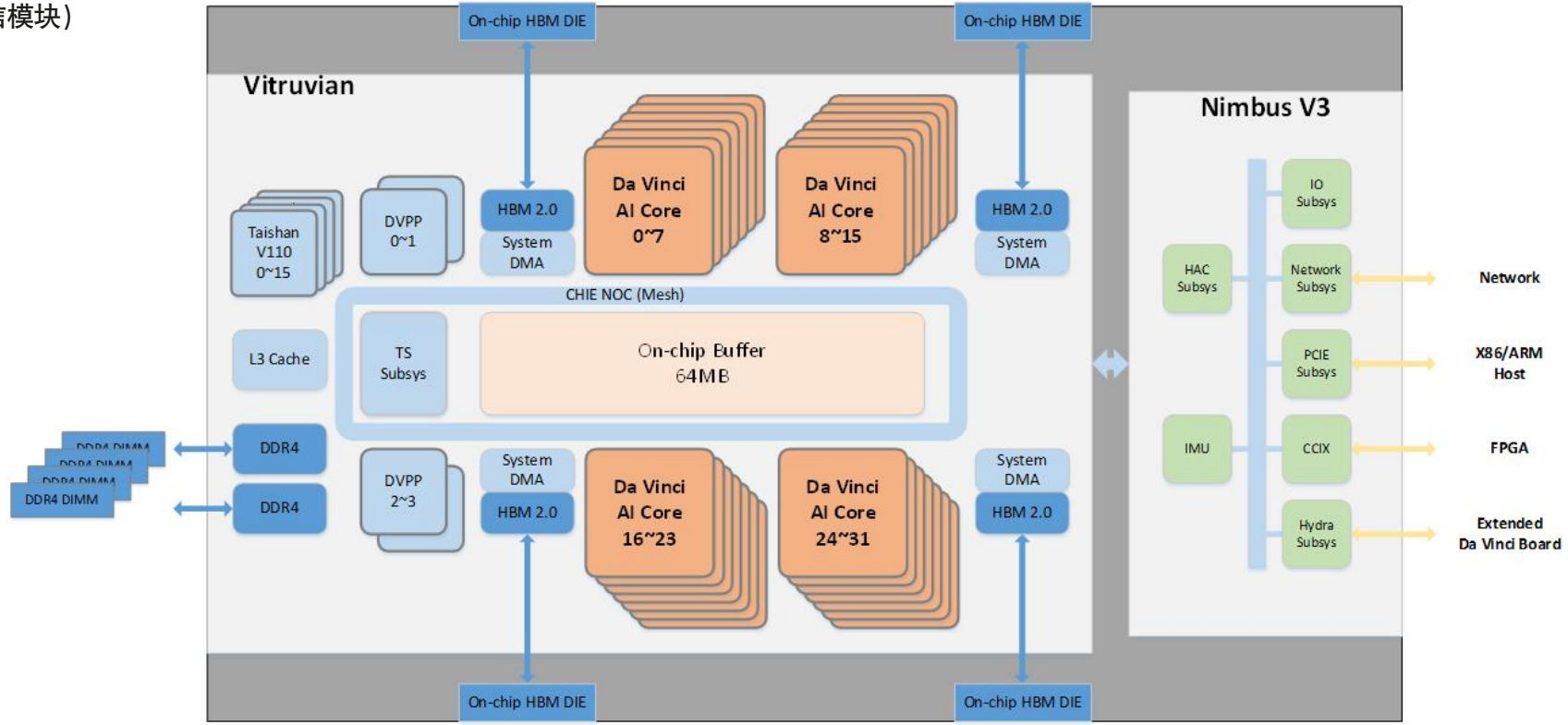
⑦ AI Core完成处理后，发送信号给任务调度器（TS），任务调度器检查结果，如果需要会分配另一个任务，并返回步骤④ 。

⑧ 当最后一个AI任务完成，任务调度器（TS）会将结果报告给Host。

Ascend 910处理器逻辑架构(AI Training SoC)

Ascend 910处理器的主要架构组成：

- AI数据处理子系统（AI Core）
- 计算子系统（CPU）
- 图像视频处理子系统（DVPP）
- 存储子系统（层次化的片上系统缓存Cache或缓冲区Buffer）
- 低速外设接口（Nimbus外部通信模块）



Ascend 910处理器逻辑架构(AI Training SoC)

CPU子系统

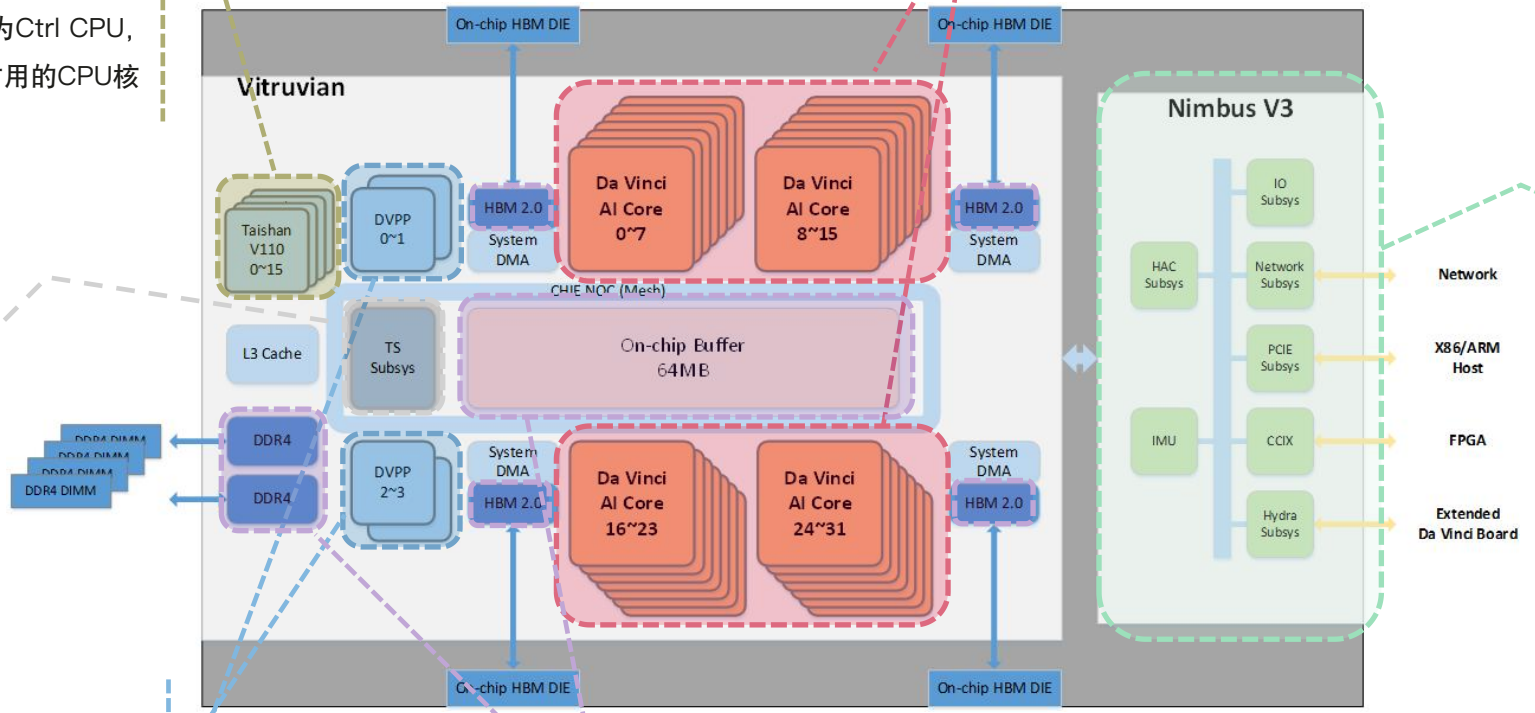
集成16个TaishanV110 Core（4个Core构成一个Cluster）。这些Taishan Core一部分部署为AI CPU，承担部分AI计算功能（负责执行不适合跑在AI Core上的算子）；一部分部署为Ctrl CPU，负责整SoC的控制功能。两类CPU占用的CPU核数由软件分配

AI Core

昇腾AI芯片的计算核心，主要负责执行矩阵、向量计算密集的算子任务，采用达芬奇架构。Ascend 910集成了32个AI Core。

TS CPU (Task Scheduler)

一个独立的4核A55 Cluster (ARMv8 64位架构)，负责任务调度，把算子任务切分之后，通过硬件调度器（HWTS），分发给AI Core或AI CPU。



Nimbus

提供x16 PCIe 4.0接口，和Host CPU对接，提供100G NIC（支持ROCE V2协议）用于跨服务器传递数据；集成1个A53 CPU核，执行启动、功耗控制等硬件管理任务。

DVPP

数字视觉预处理子系统，完成图像视频的编解码等预处理操作。

Cache & Buffer

片内有层次化的memory结构，AI Core内部有两级memory buffer，SOC片上还有64MB L2 buffer，专用于AI Core、AI CPU，提供高带宽、低延迟的memory访问。Virtruvian连接4个HBM 2.0颗粒，总计32GB容量。芯片还集成了DDR 4.0控制器，为芯片提供的DDR内存。

HUAWEI

基于昇腾AI处理器的产品形态

昇腾310

AI基础设施

X 1

X 1

X 4

X 1

Atlas 200(加速模块)	Atlas 200DK(开发者套件)	Atlas 300(PCIE加速卡)	Atlas 500(智能小站)
• 16TOPS INT8 @ 9.5W • 16路高清视频实时分析 • 4GB/8GB内存 ｜ PCIe 3.0 x 4接口 • 工作温度：-25℃ ~ +80℃ • 尺寸：52 x 38 x 10.2 mm	• 16TOPS INT8 @ 24W • 1*USB type-C｜2* CCM接口｜1*GE网口｜1* SD卡插槽 • 8GB内存 • 工作温度：0℃ ~ 45℃ • 尺寸：125 × 80 × 24 mm	• 64TOPS INT8 @ 67W • 64路高清视频实时分析 • 32GB内存 ｜ 204.8GB/s内存带宽 • PCIe 3.0 x16，半高半长卡	• 16 TOPS INT8 @ 25-40W • 支持WiFi & LTE • 16路高清视频实时分析 • 无风扇设计｜-40 ° C 至 +70 ° C｜TEC散热技术

应用场景

视频分析 ｜ OCR ｜ 语音识别 ｜ 精准营销 ｜ 医疗影像分析

平安城市
(人脸识别/车牌识别)

智慧交通
(信号灯优化/智能导流)

智慧安监
(越界检测/合规检测/看守

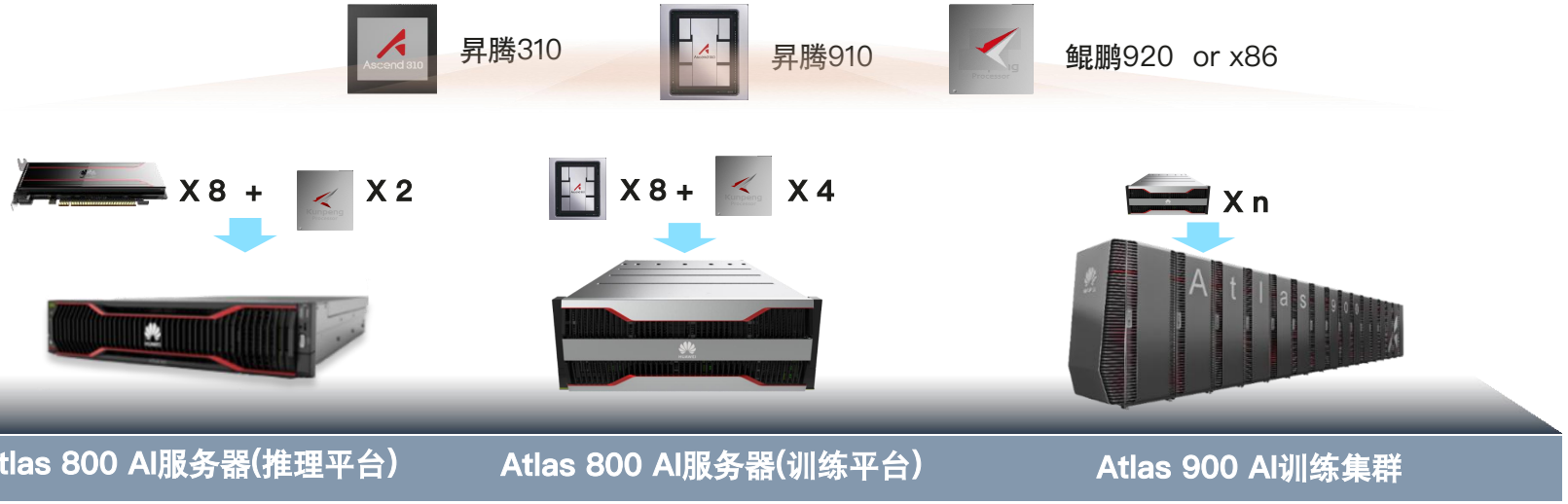
智能制造
(智能质检/柔性制造)

智能看护
(幼儿园/老人院)

无人零售
(无人货柜/智慧门店)

HUAWEI

基于昇腾AI处理器的产品形态



应用场景

政府	医疗	公共安全	制 造
OCR识别	图像识别	人脸识别 车辆识别 轨迹跟踪	智能质量检测
• 依托OCR识别技术，提升电子卷宗排版、归类和入卷效率，**人工录入工作量减少70%**。	• 依托肺部CT影像识别，将新冠肺炎诊断时间由传统方式的**15分钟降低至2分钟，准确性提升至98%**。	• 助力各市部署雪亮工程，**3秒返回8亿张**以图搜图结果 • 日均**千万级卡口图片解析**。 • 性能优于业界主流水平**2倍**以上。	• AI机器视觉可应对复杂多样场景，检测精度可由传统90%**提升至99%以上**。

深度学习训练

天文探索

石油勘探

HUAWEI

目录

1. 华为AI全栈解决方案介绍
2. 昇腾AI处理器硬件架构
3. 昇腾AI处理器软件架构

昇腾AI处理器软件逻辑架构

芯片使能层：实现解决方案对外能力开放，以及基于计算图的业务流的控制和运行。

- **AscendCL昇腾计算语言库**
  开放编程框架，提供Device/Context/Stream/内存等的管理、模型及算子的加载与执行、媒体数据处理、Graph管理等API库，供用户开发深度神经网络应用。
- **图优化和编译**
  统一的IR接口对接不同前端，支持TensorFlow/Caffe/MindSpore表达的计算图的解析/优化/编译，提供对后端计算引擎最优化部署能力。
  - Graph Engine：图编译和运行的控制中心
  - Fusion Engine：管理算子融合规则
  - AICPU Engine：AICPU算子信息管理
  - HCCL：HCCL算子信息管理
- **算子编译和算子库**
  - TBE：编译生成算子及算子开发工具
  - 算子库：神经网络加速库
- **数字视觉预处理**
  实现视频编解码(VENC/VDEC)、JPEG编解码(JPEGD/E)、PNG解码(PNGD)、VPC(预处理)
- **执行引擎**
  - Runtime：为神经网络的任务分配提供资源管理通道
  - Task Scheduler：计算图Task序列的管理和调度、执行

计算资源层：主要实现系统对数据的处理和对数据的运算执行。

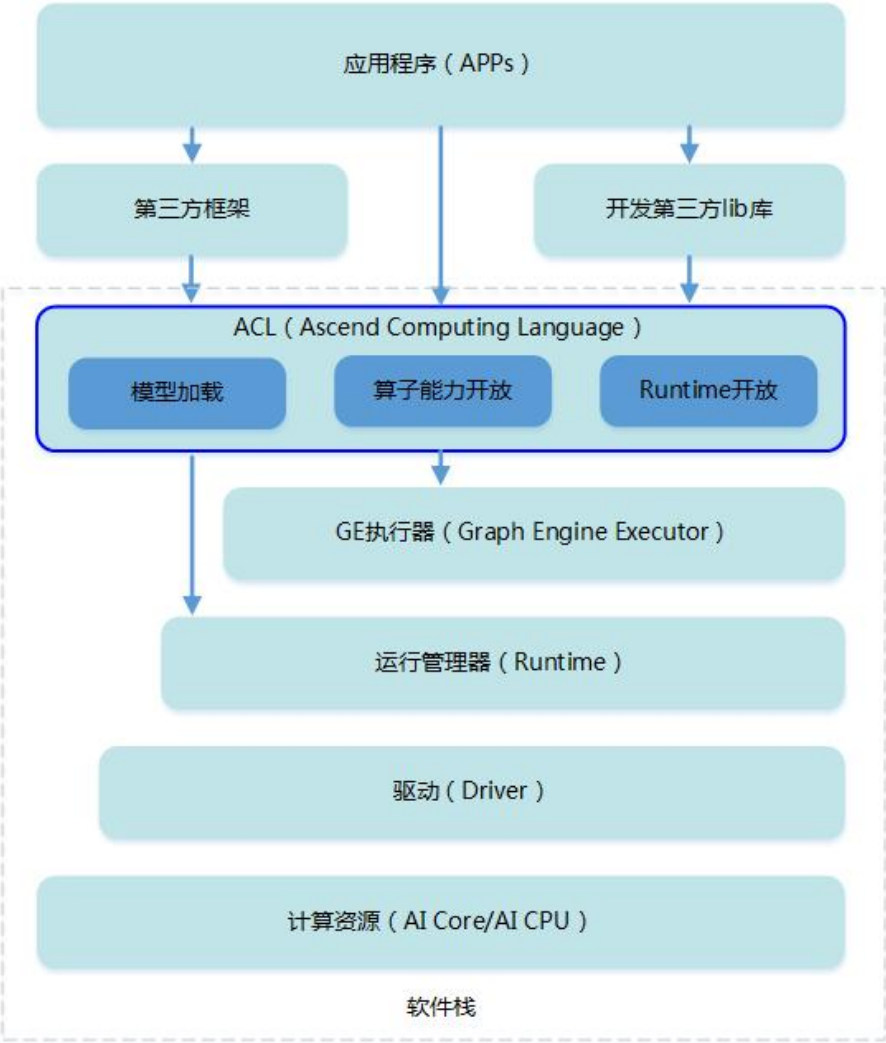
- **计算设备**
  - AI Core：执行NN类算子
  - AI CPU：执行CPU算子
  - DVPP：视频/图像编解码、预处理
- **通信链路**
  - PCIe：芯片间或芯片与CPU间高速互联
  - HCCS：实现芯片间缓存一致性功能
  - RoCE：实现芯片内存 RDMA功能

应用层：包括基于Ascend平台开发的各种应用，以及Ascend提供给用户进行算法开发、调优的应用类工具。

- **推理应用**
  基于AscendCL提供的API构建推理应用
- **AI框架**
  包括Tensorflow、Caffe、Mindspore以及第三方框架
- **模型小型化工具**
  实现对模型进行量化，加速模型
- **AutoML工具**
  基于MindSpore自动学习工具，根据昇腾芯片特点进行搜索生成亲和性网络，充分发挥昇腾性能
- **加速库**
  基于AscendCL构建的加速库（当前支持Blas加速库）
- **MindStudio**
  提供给开发者的集成开发环境和调试工具，可以通过MindStudio进行离线模型转换、离线推理算法应用开发调试、算法调试、自定义算子开发和调试、日志查看、性能调优、系统故障查看等

HUAWEI

软件架构 —— ACL子系统



ACL（Ascend Computing Language）：

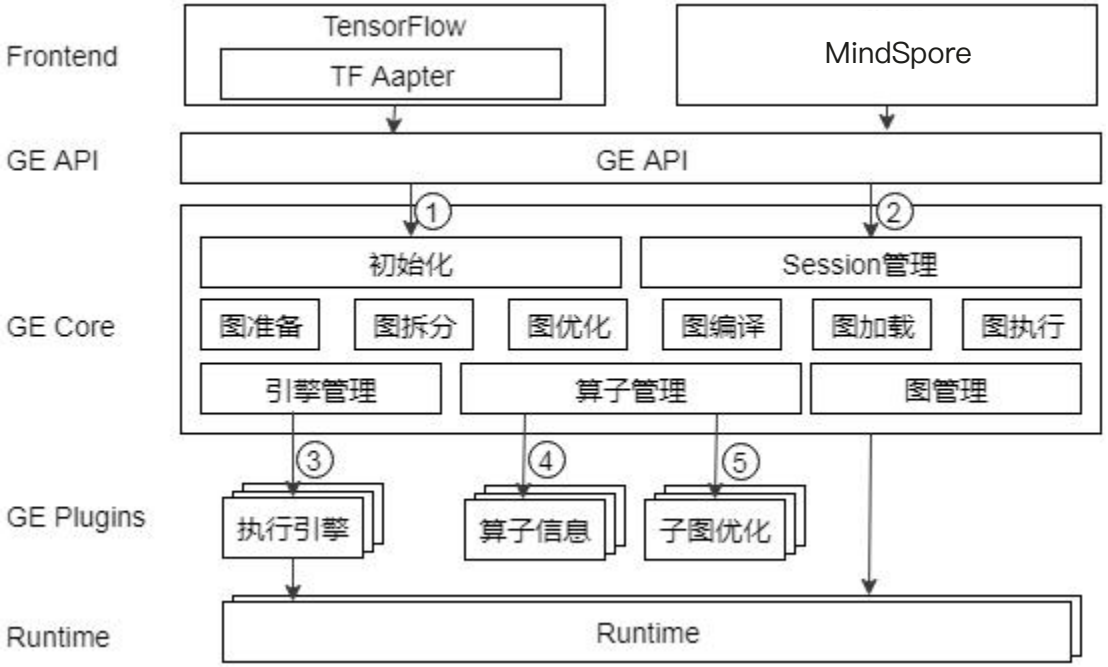
- 提供Device管理、Context管理、Stream管理、内存管理、模型加载与执行、算子加载与执行、媒体数据处理等C++ API库供用户开发深度神经网络应用，通过加载模型推理实现目标识别、图像分类等功能。用户可以通过第三方框架调用ACL接口，以便使用昇腾AI处理器的计算能力；用户还可以使用ACL封装实现第三方lib库，以便提供昇腾AI处理器的运行管理、资源管理能力。

- 在运行应用时，ACL调用GE执行器提供的接口实现模型和算子的加载与执行、调用运行管理器的接口实现Device管理/Context管理/Stream管理/内存管理等。

ACL提供的是分层开放能力的管控，通过不同的组件对不同的使能部件进行对接。包含GE能力开放、算子能力开放、Runtime能力开放、Driver能力开放等。

- **模型加载能力开放**：处理om模型加载，但接口的开放是通过ACL。
- **算子能力开放**：算子能力实现在CANN中，但算子能力开放是通过ACL。
- **Runtime能力开放**：处理基于stream的设备能力、内存、event等资源能力开发诉求，对app屏蔽底层实现。

软件架构 —— GE子系统



GE的核心功能组件：

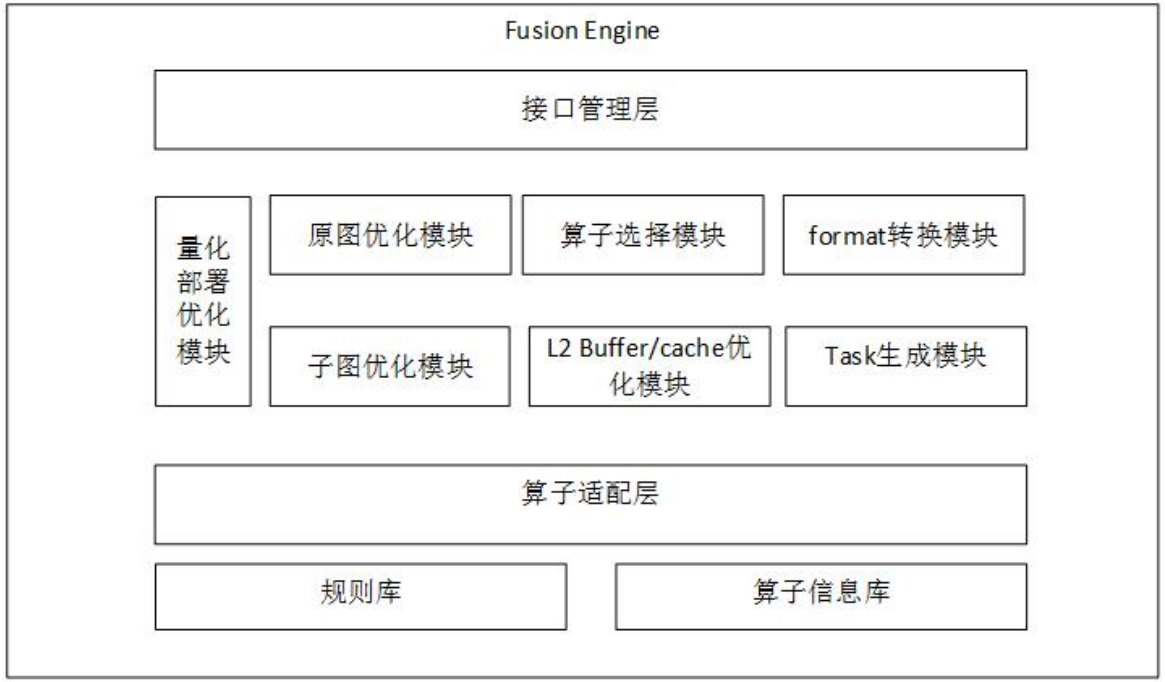
- 图准备：全局优化，完成shape推导，维测类算子并行拆分。
- 图拆分：引擎子图切分&边界连接
- 图优化：引擎/部件级优化，权值格式转换，图聚合（allreduce, L2Loss）
- 图编译：资源分配和Task生成
- 图加载：将Task加载到Runtime上
- 图执行：在Runtime上运行Task

GE（Graph Engine）：

GE作为图编译和运行的控制中心，提供运行环境管理、执行引擎管理、算子库管理、子图优化管理、图操作管理和图执行控制。

- GE通过统一的接口提供多前端的支持，不同的前端框架可以通过适配层完成不同格式图到GE IR Graph的转换。
- GE API对外呈现GE Core中初始化、Session管理模块的接口，支持运行环境初始化，Session创建、销毁，图添加执行。
- GE Core从上往下，共分为三层，分别是执行控制层、业务功能层、数据管理层。
  - **执行控制层**，提供API接口实现逻辑的控制，通过Runtime、业务功能模块、数据管理模块的接口，完成功能的实现。
  - **业务功能层**，为图执行提供最优执行引擎匹配，端到端执行路径优化，提供最低执行开销，支持不同的物理运行环境部署。
  - **数据管理层**，包含对外部插件的管理（执行引擎、算子库）及GE执行过程中需要的内部数据管理（图），对业务功能提供支持。

软件架构 —— FE子系统



FE的功能框架，从业务层次上划分，可以分为三层：

- 上层是接口层，对外提供各种接口；
- 中间是业务层，主要是两块功能：图优化和Task生成；
- 底层是数据层，维护着计算引擎信息、融合规则信息和算子信息库；

FE（Fusion Engine）：

为了业务的编排，抽象出执行引擎FE。它是一个逻辑概念，里面分三个部件：控制引擎、计算引擎、IO引擎。每个部件中都包含有算子库。

FE定位于AI Core的数据引擎，它提供图的优化分析、管理算子融合规则、算子融合功能、算子信息库管理、使能自定义算子等功能，提供如下功能：

- **接口管理层**：对GE提供了算子管理，优化管理，task生成等接口；
- **量化部署优化模块**：提供INT8量化的部署优化，辅助量化工具完成对图的INT8量化处理；
- **原图优化模块**：提供对整图的图融合处理；
- **算子选择模块**：提供了对图上各节点优先选择aicore算子实现；
- **Format转换模块**：提供了选择D芯片下算子实现支持的format和插入转换op的功能；
- **子图优化模块**： 对GE拆分的aicore引擎子图进行优化处理；
- **L2 buffer/cache优化模块**：对图上算子进行L2 buffer/cache的优化处理；
- **Task生成模块**：提供了生成taskinfo的功能；
- **算子适配层**：提供了适配TBE算子，自定义算子的功能；
- **规则库**：提供了融合的规则列表；
- **算子信息库**：提供了算子实现的信息库文件列表。

软件架构 —— TBE子系统

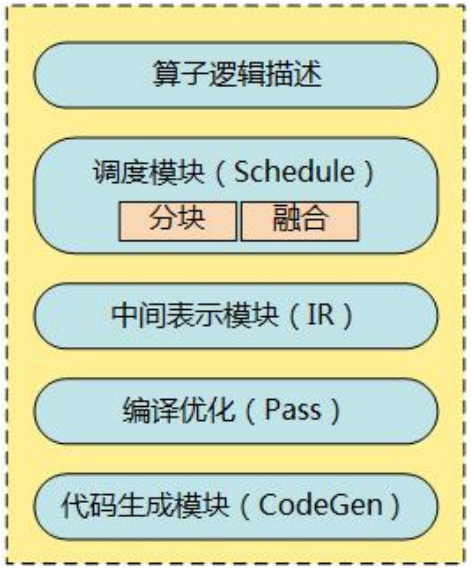
TBE (Tensor Boost Engine)：

提供了基于TVM框架的自定义算子开发能力，通过TBE提供的API和自定义算子编程开发界面可以完成相应神经网络算子的开发。

TBE在昇腾软件栈中的逻辑架构

- **前端框架**：包含第三方开源框架Tensorflow、Caffe。
- **图引擎（Graph Engine：GE)**：是基于昇腾AI处理器软件栈对不同的机器学习框架提供统一的IR接口，对接上层网络模型框架，例如Tensorflow、Caffe。主要功能包括图准备、图拆分、图优化、图编译、图加载、图执行和图管理等（此处图指网络模型拓扑图）。
- **融合引擎（Fusion Engine：FE）**：负责对接GE和TBE算子，具备算子信息库的加载与管理、融合规则管理、原图融合和子图优化的能力。GE在子图优化阶段将子图传递给FE，FE根据算子信息库以及FE融合优化进行预编译，例如修改数据类型、插入转换算子等，该子图将再次传递给GE进行子图合并及子图优化。
- **张量加速引擎（TBE）**：通过IR定义为GE的图推导提供必要的算子信息，通过算子信息库和融合规则为FE提供子图优化信息和TBE算子调用信息，TBE生成的二进制对接昇腾AI处理器，最终生成网络在昇腾AI处理器上的执行任务。

TBE功能框架

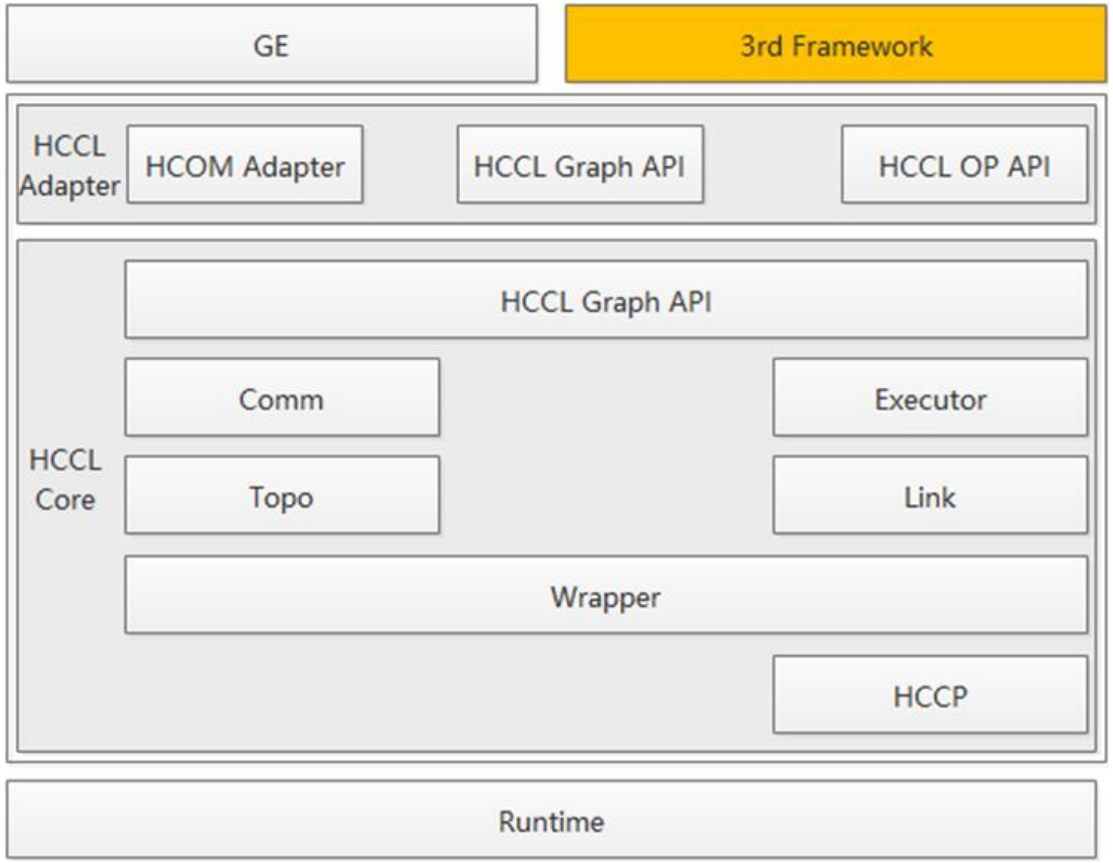


- **DSL模块**：面向开发者，提供算子逻辑的编写的接口（Compute接口），使用接口来编写算子。
- **Schedule模块**：用于描述指定shape下算子如何在昇腾AI处理器上进行切分，包括Cube类算子的切分、Vector类算子的切分，它们仍然使用的是社区提供的调度原语来描述。
- **IR模块**：借用社区的IR来表示的，包括IR变形、AST树的维护等功能。
- **编译优化（Pass）**：对生成的IR进行编译优化，优化的方式有双缓冲（Double Buffer）、流水线（Pipeline）同步、内存分配管理、指令映射、分块适配矩阵计算单元等。
- **代码生成模块（CodeGen）**：CodeGen生成类C代码的临时文件，这个临时代码文件可以通过编译器生成算子的实现文件，可被网络模型直接加载调用。

软件架构 —— HCCL子系统

HCCL (Huawei Collective Communication Library)：

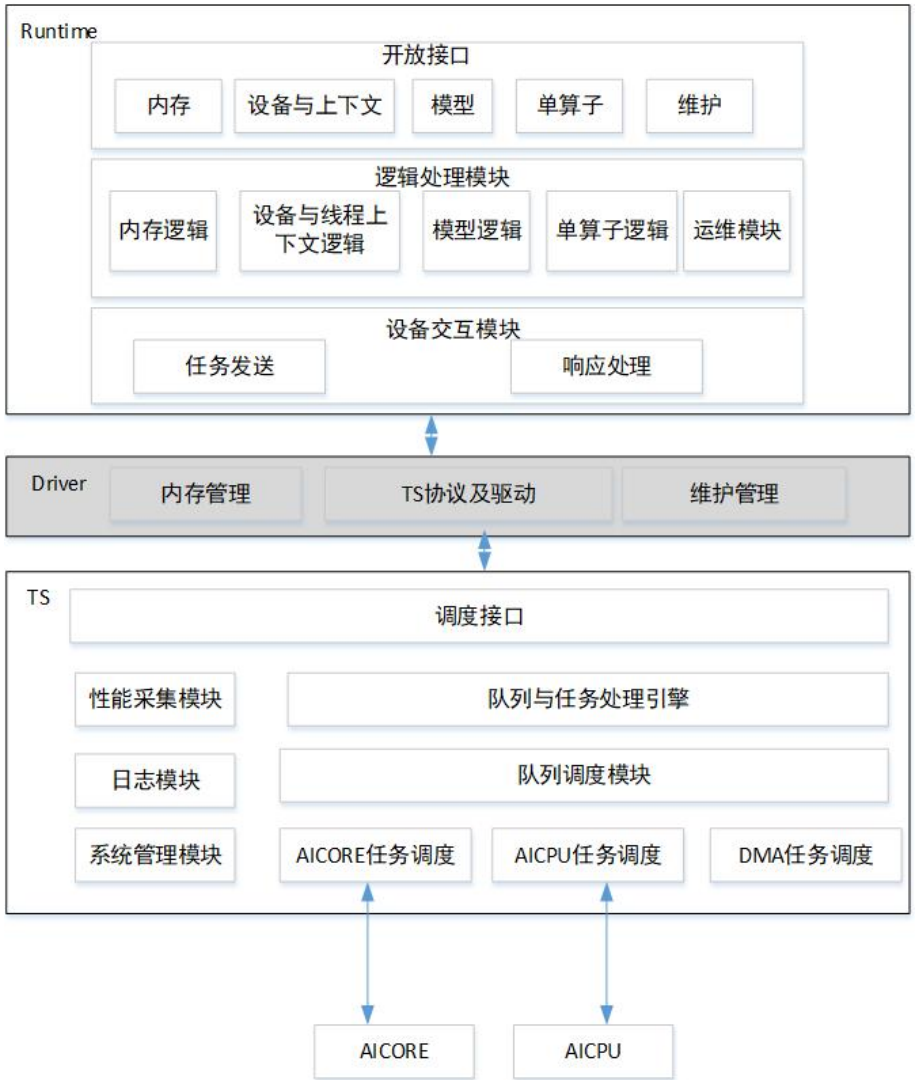
集合通信对外提供集合通信算子，支持网卡及集群不同节点间的RoCE传输功能，为分布式训练中不同npu之间提供高效的数据传输能力。



集合通信的功能框架主要包括以下部分：

- **HCOM适配器**：实现框架图模式新流程适配。提供初始化/算子运行时信息获取/算子编译加载/资源销毁等功能接口。
- **HCCL Graph API**：完成图模式下集合通信操作对外接口适配，通过图引擎支持不同的框架。功能包含通信域初始化/销毁；模型绑定/解绑定；提供Allreduce/allgather/broadcast/reduce-scatter集合通信操作。
- **HCCL OP API**：提供单算子模式下集合通信操作功能，支持不同的框架。
- **HCCL Core层API算法适配**：实现对外集合通信操作的分解，调用通信维护模块和操作执行模块提供的接口完成指定集合通信的功能。
- **Comm通信域维护模块**：提供不同拓扑算法的子通信域初始化接口，用户rank到对应子通信域rank的映射接口。
- **Executor操作执行模块**：实现集合通信操作和算法的抽象。
- **Link数据交互模块**：实现集合通信操作过程中的数据收、发以及所需的同步接口，为子通信域提供数据通信连接实例的创建和销毁接口。
- **Topo模块**：负责server内处理多npu的算法排序等。
- **Wrapper模块**：实现task下发和资源(stream/mem)的封装。
- **HCCP模块**：实现控制面的信息交互，以及NIC和RoCE相关功能。

软件架构 —— Runtime&TS子系统



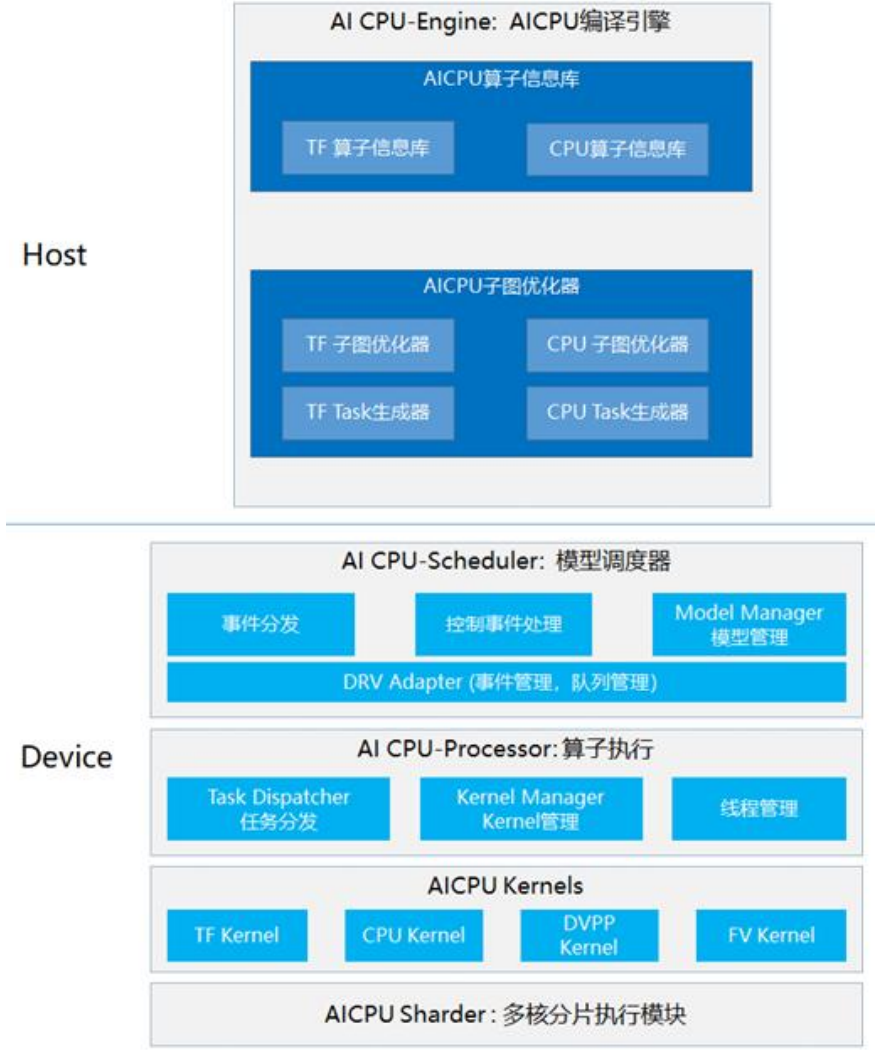
Runtime：

为神经网络的任务分配提供了资源管理通道。Runtime运行在应用程序的进程空间中，为应用程序提供了存储(Memory)管理、设备(Device)管理、执行流(Stream)管理、事件(Event)管理、核(Kernel)函数执行等功能。

Task Schedule：

运行在Device侧的任务调度CPU 上，负责将Runtime分发的具体任务进一步分发到AICPU 上。它也可以通过硬件任务调度器（HWTS）把任务分配到AI Core上执行，并在执行完成后返回任务执行的结果给运行管理器。通常Task Schedule处理的主要事务有AI Core任务、AI CPU 任务、内存复制任务、事件记录任务、事件等待任务、清理维护(Maintenance)任务和性能分析(Profiling)任务。

软件架构 —— AICPU子系统



AICPU子系统提供两大功能：

- **AICPU算子编译（部署在Host）**
  - AICPU算子信息库：包括TF算子信息库支持的TF算子名称、支持的format等。
  - AICPU图优化器：实现TF子图优化器，将TF子图优化成单function执行，减少task中断次数，提升AICPU算子执行效率。图优化器同时包含GE IR to TF算子的配置信息，可灵活的设置GE IR 与TF 算子的映射关系。
- **AICPU调度执行（部署在Device）**
  - AICPU–Scheduler：负责模型、stream等信息管理和模型状态管理；负责与TS交互控制命令，调度模型执行和结束。
  - AICPU–Processor：提供AICPU任务工作线程管理和Kernel注册管理。提供task分发功能，将CPU的task分发调度到各AICPU工作线程进行执行。
  - AICPU–Kernels：提供CPU算子，TF算子，DVPP算子，FV（短特征）算子的执行功能。

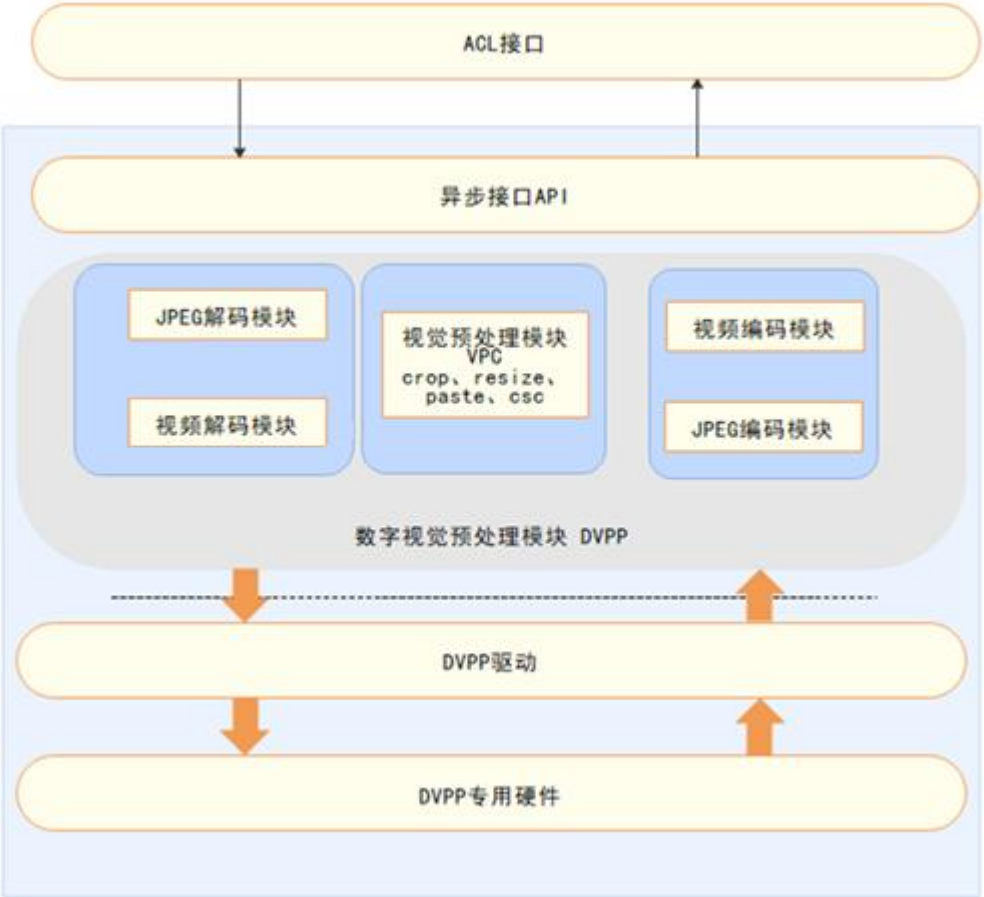
关于Host/Device的概念：

在异构计算架构中，昇腾AI处理器与CPU通过PCIe总线连接在一起来协同工作。

Host：CPU所在位置称为主机端（Host），是指与昇腾AI处理器所在硬件设备相连的X86服务器、ARM服务器或者WindowsPC，利用昇腾AI处理器提供的NN(Neural–Network)计算能力完成业务。

Device：是指安装了昇腾AI处理器的设备，利用PCIe接口与Host连接，为Host提供NN计算能力。

软件架构 —— DVPP子系统



DVPP（Digital Vision Pre-Processing）

主要实现视频解码（VDEC）、视频编码（VENC）、JPEG编解码（JPEGD/E）、VPC（预处理）。

DVPP对外提供的接口：

- **视频解码（VDEC）**：支持H264、H265两种视频格式的解码。
- **视频编码（VENC）**：实现YUV/YVU420图片数据的编码，支持H264、H265两种视频格式的编码。
- **JPEG编码（JPEGE）**：将YUV格式图片编码成.jpg图片，支持YUV422 Packed、YUV420SP（NV12,NV21）。
- **JPEG解码（JPEGD）**：实现.jpg、.jpeg、.JPG、.JPEG图片的解码，对于硬件不支持的格式，会使用软件解码。
- **VPC(Vision Preprocessing Core)：支持的功能有：抠图、缩放、叠加、黏贴、格式转换功能。**

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

HUAWEI