WP506 (v1.0.2) 2018 年 10 月 3 日

赛灵思 AI 引擎及其应用

对于 5G 蜂窝和机器学习 DNN/CNN 这样的计算密集型应用，赛灵思的新型向量处理器 AI 引擎由一系列 VLIW SIMD 高性能处理器构成，可提供高达 8 倍的芯片计算密度，功耗却比传统可编程逻辑解决方案低 50%。

摘要

本白皮书探讨了将赛灵思新 AI 引擎用于计算密集型应用（如 5G 蜂窝和机器学习 DNN/CNN）的架构、应用和优势。

与前几代相比，5G 的计算密度要高 5 到 10 倍；AI 引擎已针对 DSP 进行了优化，可满足吞吐量和计算要求，从而提供无线连接所需的高带宽和加速速度。

许多产品中机器学习的采用（通常采用 DNN/CNN 网络的形式）大大增加了计算密度要求。AI 引擎针对线性代数进行了优化，可提供满足这些要求的计算密度，同时与可编程逻辑中执行的类似功能相比，功耗也降低了 50%。

AI 引擎使用了许多程序员所熟悉的 C/C++ 范例进行编程。AI 引擎与赛灵思的自适应与标量引擎集成，可提供高度灵活且功能强大的整体解决方案。

赛灵思深厚的计算历史

赛灵思产品在计算密集型应用方面坐拥数十年的实施历史，该领域的开拓始于 90 年代初的高性能计算 (HPC) 和数字信号处理 (DSP)。FPGA 的赛灵思 XC4000 系列 FPGA 已成为支持商用和航空航天与国防无线通信系统数字前端 (DFE) 解决方案的关键技术。这些早期实践者使用 LUT 和加法器 实现计算元素（例如乘法器），以构建 DSP 功能数、FIR 滤波器和 FFT。

随着更多的客户采用赛灵思器件来处理严苛的新型计算应用，特定的计算密集型单元应运而生，例如 2001 年为 Virtex®-II 系列 FPGA 开发的第一个“DSP 片”。遵照摩尔定律，赛灵思将 LUT 数量从 XC4000 FPGA 中的 400 个增加到现有器件中的 370 多万个 LUT 和超过 12,200 个 DSP 片，将可用资源增加了 9,500 多倍。随着计算资源的不断增加，赛灵思产品始终能够提供所需的计算密度和逻辑资源，紧跟新兴信号处理市场的步伐。

技术进步推动计算密度提高

多种技术的进步推动了对非线性更高计算密度的需求。采样速率为每秒千兆赫的数据转换器能够对 RF 信号直接采样，这虽然简化了模拟系统，但需要相应数量级更高的 DSP 计算密度支持。直接 RF 采样结合多个天线的使用，例如具有数万个天线的高级雷达系统。

围绕 5G 无线的炒作多年来不断酝酿，5G 将环境中的所有内容连接到比蜂窝连接速度快一百倍的网络，比最快速的家庭宽带服务也要快十倍，该技术有望改变人们的生活。毫米波、大规模 MIMO、全双工、波束成形和小型蜂窝只是实现超高速 5G 网络的几项技术。5G 的两大优势在于速度和低时延是，能够支持从自动驾驶汽车到虚拟现实的多种新应用。这些技术将计算密度和存储器要求提高到超越 4G 的数量级。

使用 5G、大规模 MIMO、多天线和频段等新技术也将复杂性提高至 4G 的 100 倍。复杂性的增加直接推动了计算密度、存储器要求和 RF 数据转换器的性能。参见图 1。

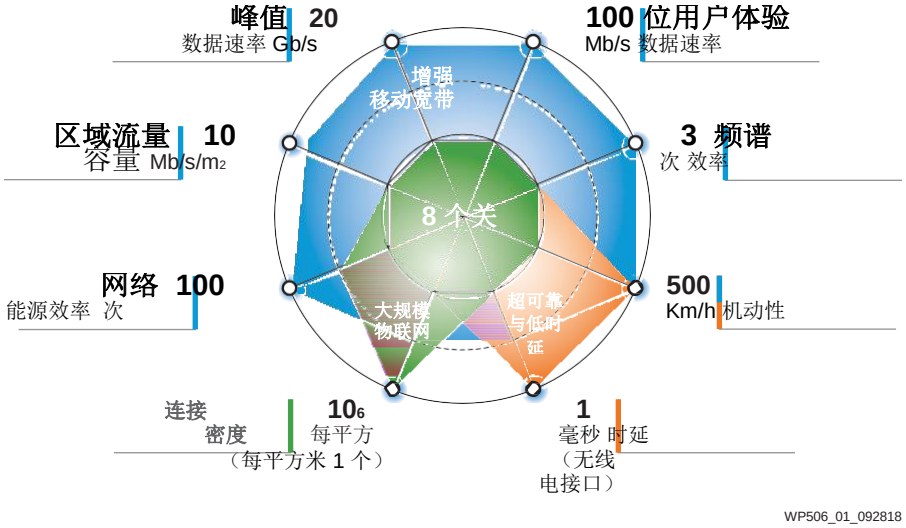

图 1： **5G 复杂性与 4G1**

1. ETRI RWS-150029、5G 视觉和关键技术：ETRI Perspective 3GPP RAN Workshop Phoenix，2015 年 12 月：http://www.3gpp.org/ftp/tsg_ran/TSG_RAN/TSGR_70/Docs

摩尔定律的衰亡

1965 年，英特尔联合创始人戈登·摩尔 (Gordon Moore) 观察到集成电路中的元件数量每两年翻一番。在 1965 年，这意味着每个芯片中布局 50 个晶体管将提供最低的每晶体管成本；摩尔预测，到 1970 年，每芯片将增加 1,000 个元件，每个晶体管的价格将下降超过 90%。摩尔后来将此修改为每两年增加一倍的资源，从 1975 年到 2012 年大致符合预测。(1)摩尔定律预测每个新型更小的工艺节点均将提供更大的密度、更高的性能和更低的功耗，以及更低的成本。该观察结果称为“摩尔定律”，并持续了大约 50 年。摩尔定律的原则是提高 IC 密度、性能和可负担性的推动因素，也是赛灵思用于日益降低器件成本的原则。

随着 IC 工艺节点达到 28 nm 及以下，出现了“违反”摩尔定律的现象；在较小的工艺节点上构建的器件的功耗、成本与性能不再易于预测。5G 蜂窝系统的计算需求与可编程逻辑计算密度之间存在差距。第 5 代蜂窝所需的成本、功耗和性能超过了可编程逻辑满足系统级目标的能力。

AI 引擎的面世

为响应下一代无线和机器学习应用对提高计算密度与降低功耗要求的非线性需求增长，赛灵思开始研究创新架构，从而开发出 AI 引擎。AI 引擎以及自适应引擎（可编程逻辑）和标量引擎（处理器子系统）形成一个紧密集成的异构计算平台。AI 引擎为基于向量的算法提供了高达五倍的计算密度。自适应引擎提供灵活的自定义计算和数据移动。标量引擎提供了复杂的软件支持。参见图 2。

1.Wikipedia.org, “Moore’s law,” https://en.wikipedia.org/wiki/Moore%27s_law,于 2018 年检索。

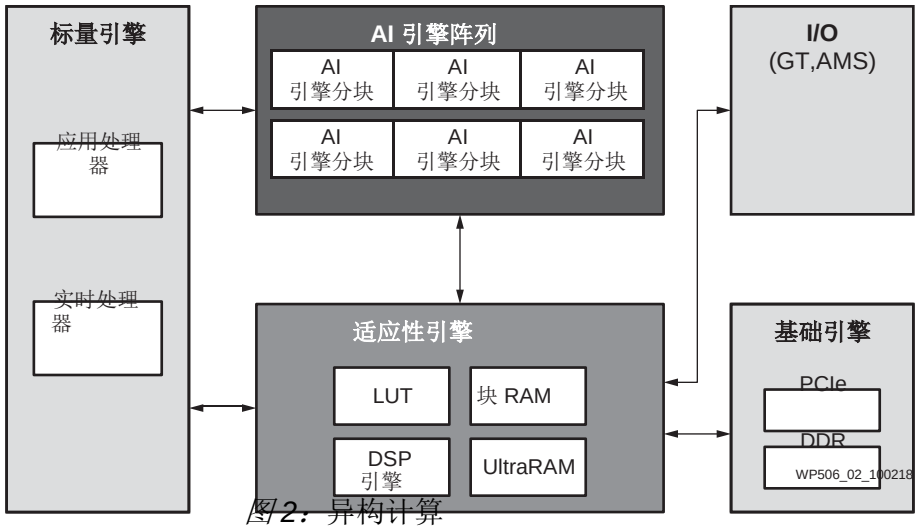

图 2：异构计算

图 3 说明了 AI 引擎接口分块到 2D 阵列的组成。

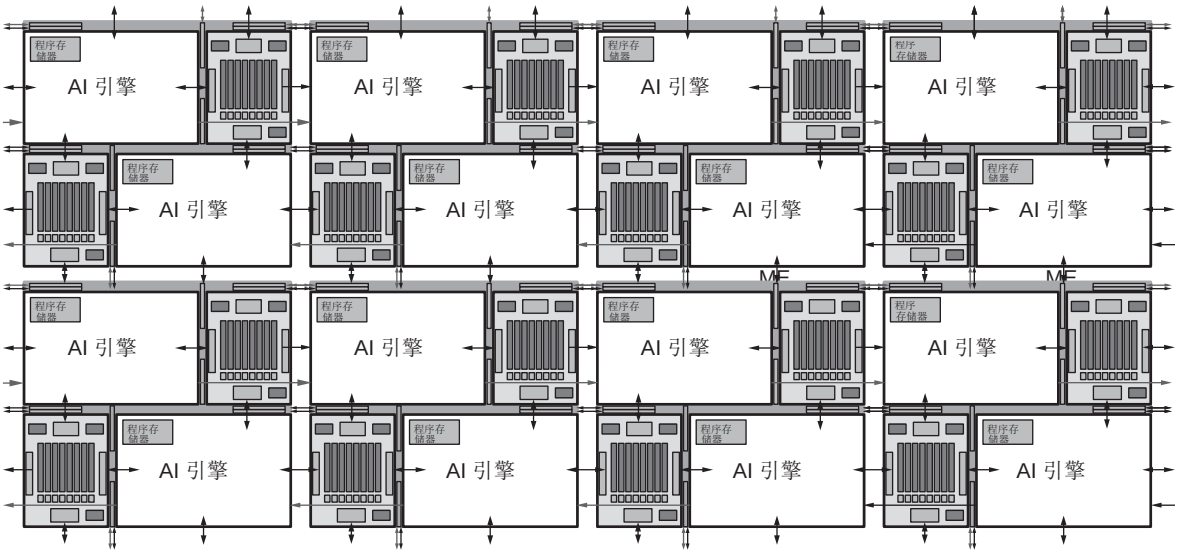

*图 3：**AI 引擎阵列***

每个 AI 引擎分块包括用于定点和浮点运算的向量处理器、标量处理器、专用程序和数据存储器、专用 AXI 数据移动通道、以及 DMA 和锁止。AI 引擎是一组单指令多数据 (SIMD)；和超长指令字 (VLIW),提供多达 6 路指令并行性，包括每个时钟周期两/三次标量运算，两次矢量加载和一个次写操作、以及一次定点或浮点向量运算

。

AI 引擎阵列针对实时 DSP 和 AI/ML 计算进行了优化，通过专用数据和指令存储器、DMA、锁止和软件工具的组合提供确定性时序。专用数据和指令存储是静态的，消除了由于缓存失败和相关填充而产生的不一致性。

AI 引擎目标和目的

AI 引擎的目标和目的来自使用 DSP 和 AI/ML 的计算密集型应用。其他市场需求包括更高的开发人员生产力与抽象级别，这些都推动了开发工具的发展。AI 引擎的开发旨在提供四个主要优势：

- 与计算密集型应用的 PL 实现相比，每个芯片面积的计算容量提高了 3 到 8 倍
- 与 PL 中实现的相同功能相比，能将计算密集型功耗降低 50%
- 提供确定性、高性能的实时 DSP 功能
- 显著改善开发环境并提高设计人员的工作效率

AI 引擎原始图像分块架构细节

要想真正掌握 AI 引擎的巨大功能，必须对其架构和功能有一个大致的了解。图 4 中显示的 AI 引擎分块提供了每个分块中资源的详细计算：

- 专用的 16 KB 指令存储器和 32 KB RAM
- 32b RISC 标量处理器
- 512b 定点和 512b 浮点向量处理器，带有相关的向量寄存器
- 同步处理程序
- 跟踪和调试

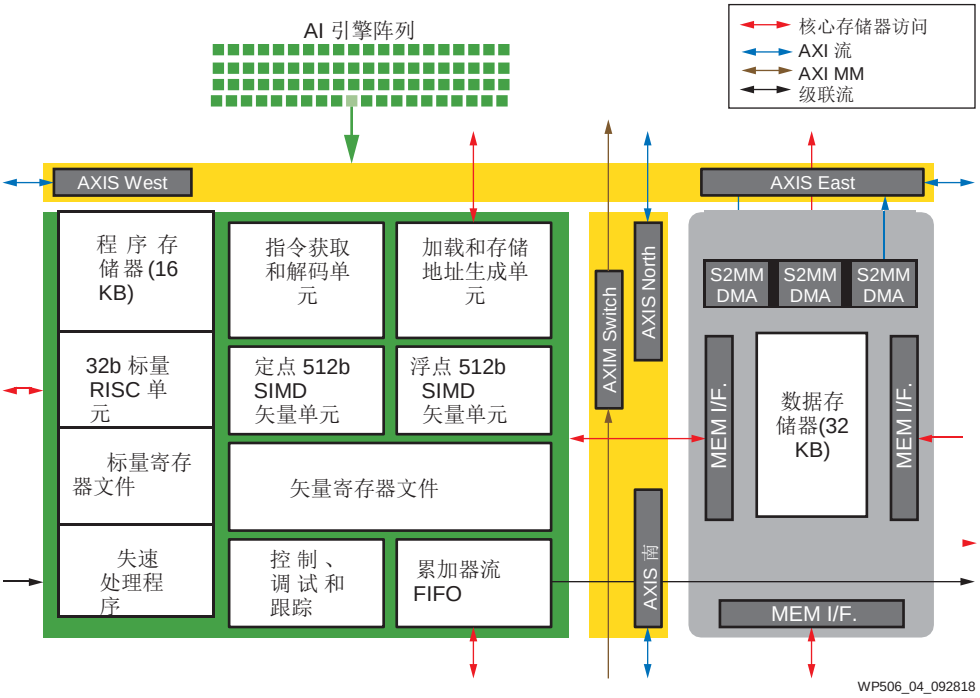

图 4： **AI 引擎分块细节**

通过使用专用 AXI 总线路由和直连相邻 AI 引擎分块的组合，将具有专用指令和数据存储器的 AI 引擎与其他 AI 引擎分块互联。针对数据移动，专用 DMA 引擎和锁止可直接连接专用 AXI 总线连接、数据移动和同步。

操作数精确支持

向量处理器由整数和浮点单元组成。支持 8 位、16 位、32 位和单精度浮点 (SPFP)的操作数。对于不同的操作数，每个时钟周期的操作数都会发生变化，如表 1 所示。

表 1： **AI 引擎向量精度支持**

操作数 A	操作数 B	输出	MAC/时钟数
8b real	8b real	16b real	128
16b real	8b real	48b real	64
16b real	16b real	48b real	32
16b real	16b complex	48b complex	16
16b complex	16b complex	48b complex	8
16b real	32b real	48/80 real	16
16b real	32b complex	48/80 complex	8
16b complex	32b real	48/80 complex	8
16b complex	32b complex	48/80 complex	4
32b real	16b real	48/80 complex	16
32b real	16b complex	48/80 complex	8
32b complex	16b real	48/80 complex	8
32b complex	16b complex	48/80 complex	4
32b real	32b real	80b real	8
32b real	32b complex	80b complex	4
32b complex	32b real	80b complex	4
32b complex	32b complex	80b complex	2
32b SPFP	32b SPFP	32b SPFP	8

指令和数据并行

通过指令级和数据级并行性实现多级并行。

指令级并行性如图 5 所示。对于每个时钟周期、两个标量指令、两个向量读取、一个单向量写入和单向量指令执行的 6 路 VLIW。

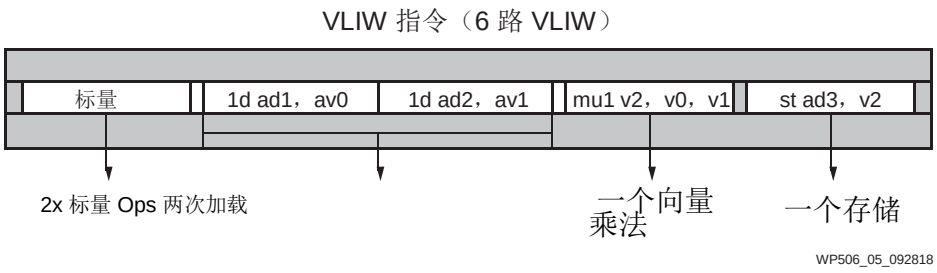

图 5： **AI 指令级并行**

数据级并行是通过向量级操作实现的，其中可以在每个时钟周期的基础上操作多组数据，如表 1 所示。

确定性性能与连接性

AI 引擎架构是为实时处理应用开发的，这些应用需要确定性的性能。两大关键的架构特性确保了确定性的时间：

- 专用指令和数据存储器
- 专用连接性与 DMA 引擎配合使用，以利用 AI 引擎分块之间的连接性进行预定数据移动

直接存储器 (DM)接口提供 AI 引擎分块与其邻近分块间的直接访问，AI 引擎分块数据存储器可以直接访问北、南和西。这通常用于在整个处理链产生和/或消耗数据的同时，将结果移动到向量处理器或从向量处理器移开结果。实施数据存储器以实现“乒乓”缓存方案，将存储器争用对性能的影响降至最低。

AI 引擎分块之间的 AXI-Stream 和 AXI-Memory 映射性连接

最简单的 AI 引擎到 AI 引擎数据移动形式是通过直接相邻 AI 引擎分块之间的共享存储器。但是，当分块距离较远时，AI 引擎分块需要使用 AXI-Streaming 数据流。AXI-Streaming 连接是由 AI 引擎编译器工具根据数据流图进行预定义和编程的。这些流接口也可用于直接与 PL 和 NoC 连接。参见图 6。

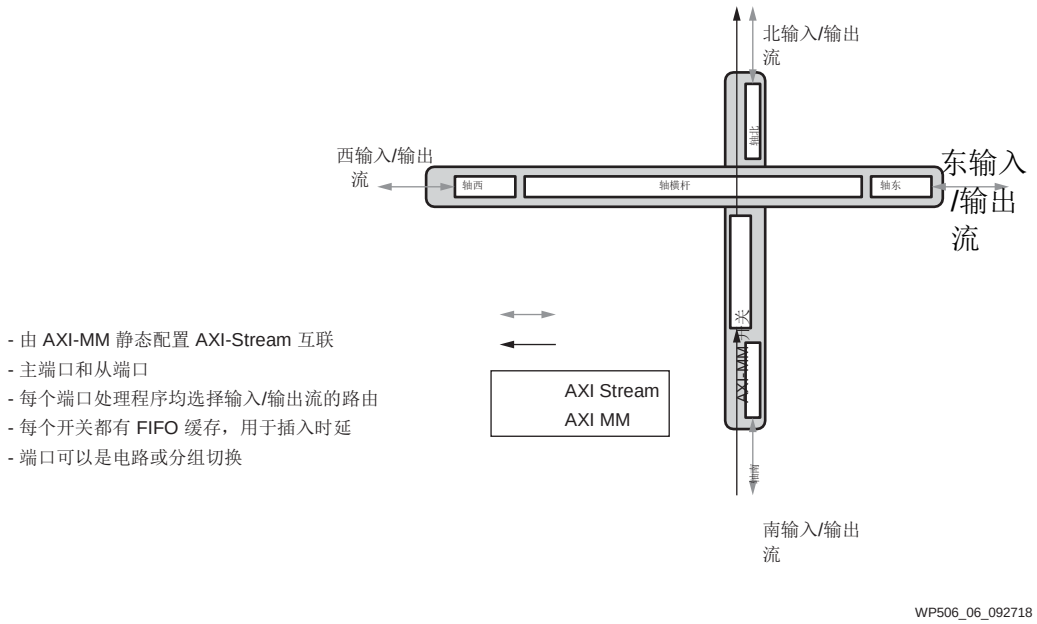

图 6： **AI 引擎阵列 AXI-MM 和 AXI-Stream 互联**

AI 引擎和 PL 连接性

Versal 产品组合的最高价值主张之一是能够在自适应引擎中使用 AI 引擎阵列和可编程逻辑。这一资源组合为在最佳资源、AI 引擎、自适应引擎或标量引擎中实现功能提供了极大的灵活性。图 7 显示了 AI 引擎阵列和可编程逻辑之间的连接，称为“AI 引擎阵列接口”。AXI-Streaming 连接存在于 AI 引擎阵列接口的每一侧，并将连接扩展到可编程逻辑中，并分别扩展到片上网络 (NoC)。

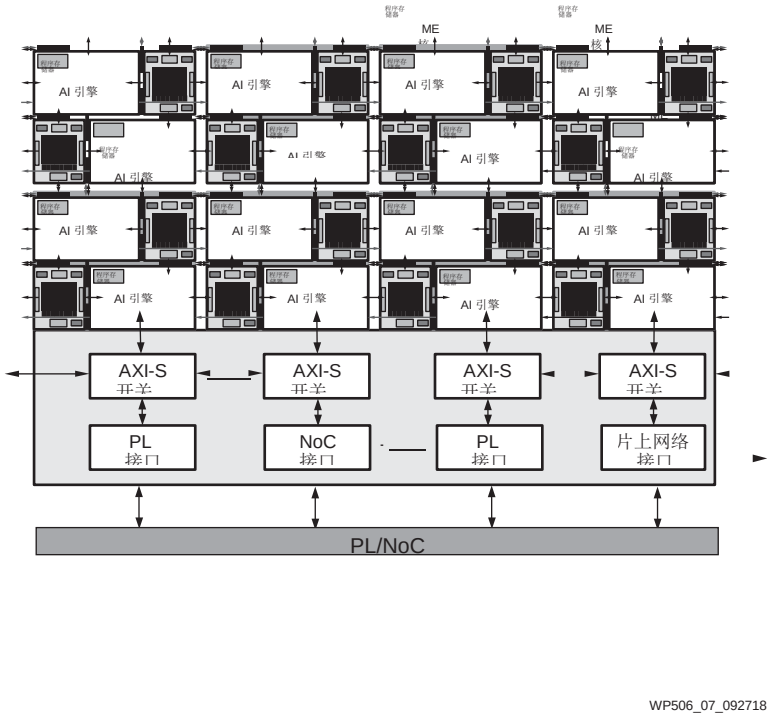

图 7： **AI 引擎阵列接口**

AI 引擎控制、调试和跟踪

在每个 AI 引擎分块中集成了控制、调试和跟踪功能，为调试和性能监控和优化提供了可视性。通过 Versal 产品组合中引入的高速调试端口，可以访问调试功能。

AI 引擎和可编程逻辑实现对比

AI 引擎目标和目的章节提供了评估是否能满足应用和市场需求所需的指标。可以通过在 PL 和 AI 引擎中实现 4G 和 5G 蜂窝来测算架构的有效性。结果总结表明，基于 AI 引擎的解决方案可以提供：

- 与在相同工艺节点上的 PL 中实现的相同功能相比，芯片面积要小 3-8 倍
- 功耗约为 PL 实现的 50%

对于那些不适合向量实现的功能，AI 引擎的效率要低得多，因此 AI 引擎通常不太适合此类功能。在这些情况下，PL 将是更好的解决方案。AI 引擎和 PL 旨在作为计算对等体运行，二者各自处理与其优势相匹配的功能。PL 非常适合数据移动、面向比特的功能和非基于向量的计算；还可以为非 AI 引擎支持的操作实现自定义加速器。PL 和 AI 引擎相互补充，能够形成更强大的系统级解决方案。在大多数计算密集型应用中，可编程逻辑仍然是一种非常有价值的资源；AI 引擎/PL 组合可提供灵活性、高计算性能和高带宽数据移动和存储。

使用 AI 引擎架构的 Versal 产品组合概述

Versal 器件包括三种类型的可编程处理器：Arm® 处理器子系统 (PS)、可编程逻辑 (PL) 和 AI 引擎。各自提供用于满足整个系统中不同部分的各种计算能力。Arm 处理器通常用于控制面应用、操作系统、通信接口、以及更低级别或更复杂的计算。PL 执行数据操作和传输、非基于向量的计算和连接。AI 引擎通常用于向量实现中的计算密集型功能。

图 8 提供了 Versal 器件的高级视图，其中 AI 引擎阵列位于器件顶部。直接和通过片上网络 (NOC)支持 AI 引擎阵列和 PL 之间的连接。

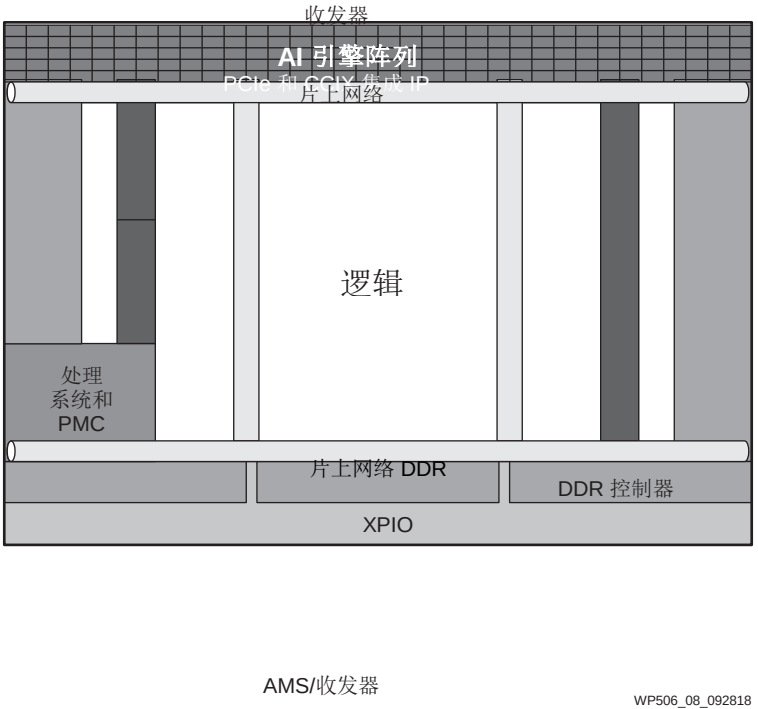

图 8： **带 AI 引擎架构的 Versal ACAP 概述**

AI 引擎开发环境

近年来，赛灵思非常重视使用高级语言 (HLL) 来帮助提高赛灵思器件开发的抽象级别。Versal 架构有三个存在根本性差异可编程元素：PL、PS 和 AI 引擎。这三个都可以使用 C/C++ 进行编程。

使用基于 x86 的仿真环境，AI 引擎仿真可以具备功能性或循环精确性。对于系统级仿真，可以使用支持所有三个处理域的 System-C 虚拟平台。

开发环境中的关键元素是支持 DSP 和无线功能、ML 和 AI、线性代数和矩阵数学的 AI 引擎库。这些库针对效率和性能进行了优化，使开发人员能够充分利用 AI 引擎功能。

AI 引擎应用

AI 引擎针对计算密集型应用进行了优化，特别是数字信号处理 (DSP)和一些人工智能 (AI) 技术，如机器学习 (ML) 和 5G 无线应用。

使用 AI 引擎的数字信号处理

无线解决方案验证套件

实时 DSP 广泛用于无线通信。赛灵思比较了经典窄带和宽带无线电设计原理、大规模 MIMO 以及基带和数字前端概念的实现，验证了 AI 引擎架构非常适合构建无线解决方案。

示例：100 MHz 5 通道 LTE20 无线解决方案

在 Versal 器件的一部分中实现了 100 MHz 5 通道 LTE20 无线。五个 16b 输入数据通道以 30.72 MSPS 的速率进行流传输，并在 89 抽头通道滤波器中处理。然后使用两级半带滤波器（23 和 11 抽头）对信号进行四次上采样，得到 122.88 MSPS 的采样率。

然后将上采样流与直接数字综合 (DDS)正弦/余弦波函数混合并求和。另外两个半带滤波器（47 和 27 抽头）进行四次上采样，产生 491.52 MSPS 输入流，以达到波峰因数降低 (CFR)功能。由 41 抽头滤波器提供的五上/四下分数速率变化，导致 614.4 MSPS 输入采样率达到数字预失真 (DPD)功能要求。

在 PL 中实现峰值检测器/比例查找 (PD/SF) 电路；491.52 MSPS DUC 的输出和混合器级包括其输入之一，而 CFR 第二级提供其第二输入。在 PL 中实现的 PD/SF 电路能高效利用资源；相反，如果在 AI 引擎中实现，则资源效率低下。对于为设计的不同功能块利用最佳资源而采取的架构决策，这提供了很好的说明。参见图 9。

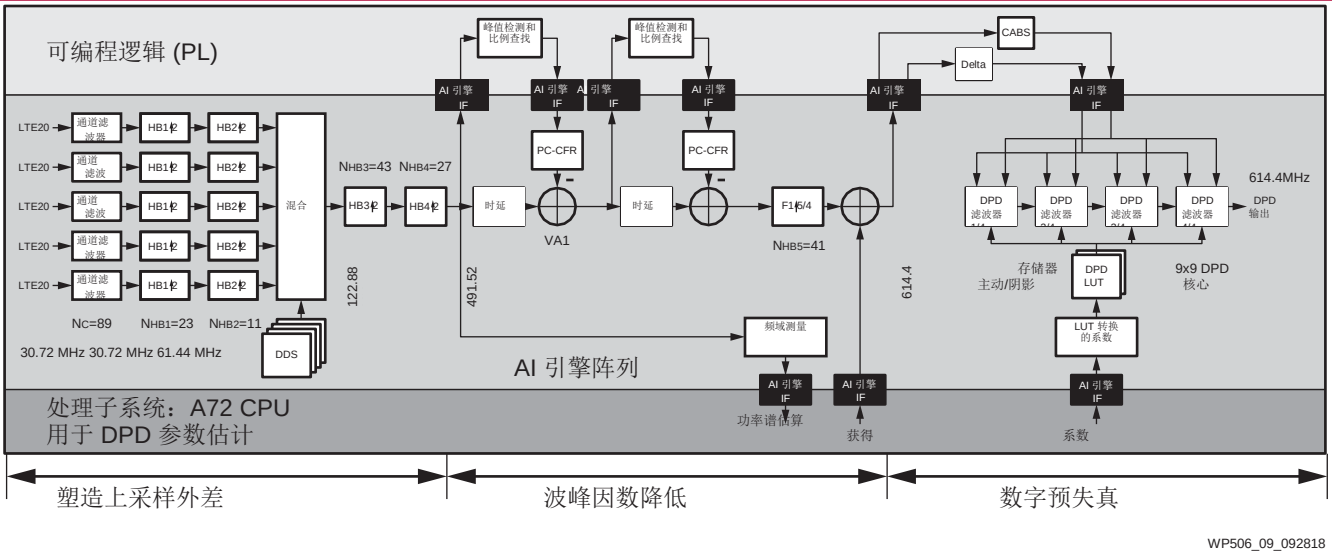

*图 9：***框图：带 DSP 的 100 MHz 5 通道 LTE20 无线解决方案**

DPD 功能需要定期重新计算系数。使用模数转换器 (ADC) 对来自发送数模转换器 (DAC) 输出的反馈路径进行采样，并进行缓缓存。将缓存的样本数据集传递给 PS，并用于每秒十次计算一组新的 DPD 系数。使用片上网络和 AXI 总线互联将新系数集写回 DPD。

机器学习和 AI 引擎

在机器学习中，卷积神经网络 (CNN) 是一类深度前馈人工神经网络，最常用于分析视觉图像。随着计算机被用于从自动驾驶车辆到视频监控以及图像和视频的数据中心分析等各种应用，CNN 变得至关重要。CNN 提供了技术突破，使视觉图像识别的可靠性足够精确，可用于安全地引导车辆。

CNN 技术尚处于起步阶段，几乎每周都会公布新的突破。该领域的创新步伐令人震惊，这意味着未来几年内可能会支持新型、以前无法实现的应用。

然而，CNN 面临的挑战在于需要大量的计算，通常需要多个 TeraOPS。AI 引擎经过优化，能够通过低成本、高能效的方式提供这一计算密度。

AI 引擎 CNN/DNN 覆盖

赛灵思正在开发基于 AI 引擎的机器学习推断引擎，并将作为应用覆盖来应用。可编程逻辑用于有效地移动和管理数据。AI 引擎应用覆盖提供了一个经定义的结构，用于实现诸多主流 CNN/DNN 网络所需的计算和其他操作，例如 ResNet、GoogLeNet 和 AlexNet。

从用户的角度来看，覆盖方法具有许多优点，包括随着更新的网络架构的出现而进行修改的能力。AI 引擎和 PL 的可编程组合提供了一个高效且非常灵活的平台，可随着 ML 应用空间的发展而增长和扩展。

将 AI 引擎 CNN/DNN 覆盖用于数据中心应用，能够加速 ML 网络推断和嵌入式系统。集成是将解决方案例化为用户整体设计的简单问题。然后使用 TensorFlow 或 Caffe 开发 CNN/DNN 网络，并将其编译成在 AI 引擎 CNN/DNN 覆盖上运行的可执行程序。

总结

AI 引擎代表了一类新的高性能计算。AI 引擎集成在 Versal 类器件中，可与 PL 和 PS 完美结合，在单个赛灵思 ACAP 中实现高复杂度系统。实时系统需要确定性行为，其中 AI 引擎通过结合专用数据和编程存储器、DMA 和锁止以及编译器工具等架构特性提供上述行为。

与传统的可编程逻辑 DSP 和 ML 实现相比，AI 引擎的芯片面积计算密度提高了 3 到 8 倍，同时降低了 50% 的功耗。C/C++ 编程范例提高了抽象级别，并承诺显著提高开发人员的工作效率。

从具有 30 个 AI 引擎和 80K LUT 的小型器件到具有 400 个 AI 引擎和近 100 万个 LUT 的器件，通过一系列器件提供系统性能可扩展性。这些器件之间的封装尺寸兼容性使产品系列内的迁移能够满足不同的性能和价格目标。

如需了解更多信息，敬请访问：

WP505, *Versal: 首款自适应计算加速平台 (ACAP)*

WP504，使用赛灵思 Alveo 加速器卡加速 DNN

修订历史

下表列出了本文档的修订历史:

日期	版本	修订描述
2018/10/3	1.0.2	仅编辑更新
2018/10/2	1.0.1	仅编辑更新
2018/10/2	1.0	赛灵思初始版本。

免责声明

本文向贵司/您所提供的信息（下称“资料”）仅在对赛灵思产品进行选择和使用时参考。在适用法律允许的最大范围内：（1）资料均按“现状”提供，且不保证不存在任何瑕疵，赛灵思在此声明对资料及其状况不作任何保证或担保，无论是明示、暗示还是法定的保证，包括但不限于对适销性、非侵权性或任何特定用途的适用性的保证；
且（2）赛灵思对任何因资料发生的或与资料有关的（含对资料的使用）任何损失或赔偿（包括任何直接、间接、特殊、附带或连带损失或赔偿，如数据、利润、商誉的损失或任何因第三方行为造成的任何类型的损失或赔偿），均不承担责任，不论该等损失或者赔偿是何种类或性质，也不论是基于合同、侵权、过失或是其他责任认定原理，即便该损失或赔偿可以合理预见或赛灵思事前被告知有发生该损失或赔偿的可能。赛灵思无义务纠正资料中包含的任何错误，也无义务对资料或产品说明书发生的更新进行通知。未经赛灵思公司的事先书面许可，贵司/您不得复制、修改、分发或公开展示本资料。部分产品受赛灵思有限保证条款的约束，请参阅赛灵思销售条款：http://china.xilinx.com/legal.htm#tos； IP 核可能受赛灵思向贵司/您签发的许可证中所包含的保证与支持条款的约束。赛灵思产品并非为故障安全保护目的而设计，也不具备此故障安全保护功能，不能用于任何需要专门故障安全保护性能的用途。如果把赛灵思产品应用于此类特殊用途，贵司/您将自行承担风险和责任。请参阅赛灵思销售条款：http://china.xilinx.com/legal.htm#tos。

关于与汽车相关用途的免责声明

如将汽车产品（部件编号中含“XA”字样）用于部署安全气囊或用于影响车辆控制的应用（“安全应用”），除非有符合 ISO 26262 汽车安全标准的安全概念或冗余特性（“安全设计”），否则不在质保范围内。客户应在使用或分销任何包含产品的系统之前为了安全的目的全面地测试此类系统。在未采用安全设计的条件下将产品用于安全应用的所有风险，由客户自行承担，并且仅在适用的法律法规对产品责任另有规定的情况下，适用该等法律法规的规定。