AI / ML 处理器和应用程序的体系结构探索

一．人工智能处理器架构探索简介

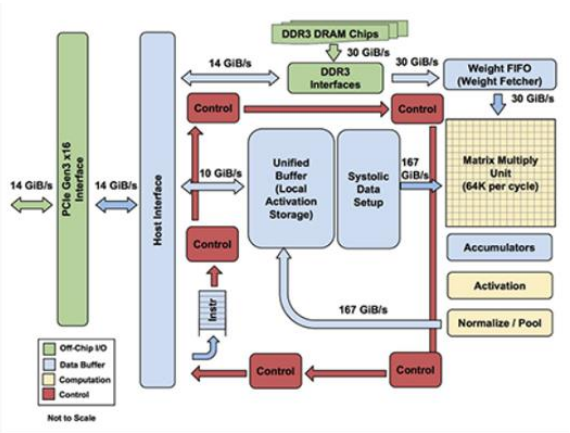
人工智能（AI）应用程序处理可以分布在处理器内的多核，PCIe 骨干网上的多个处理器板，分布在以太网网络中的计算机，高性能计算机或跨数据中心的系统中。 AI 处理器具有巨大的内存大小要求，访问时间限制，在模拟和数字之间的分布以及硬件-软件分区。 AI 处理器和系统的体系结构探索具有挑战性，因为它在硬件的全部功能上应用了数据密集型任务图。 它考虑了计算，存储，内存，管道，通信接口，软件和控制。

GenesisArch 是用于复杂计算系统（如 AI 处理器和系统）的体系结构探索的建模和仿真软件。 GenesisArch 提供了一个平台，可以探索和权衡硬件和软件的体系结构，以创建一个完全经过验证的系统，该系统可以满足时序截止日期和成本方面的考虑

不得将这些体系结构模型与在开发过程中或集成期间构建的虚拟原型混淆。 虚拟原型用于性能验证，该过程为正在开发或已经开发的系统提供定时号。

二．Google Tensor Processor 的体系结构探索示例

图 1 显示了 Google Tensor Processor 的内部视图。 处理器通过 PCIe 接口接收来自主机的请求。 权重存储在片外 DDR3 中，并调入权重 FIFO。 到达的请求在统一本地缓冲区中存储和更新，并发送给矩阵多个单元进行处理。 通过 AI 管道处理了请求后，请求将返回到统一缓冲区以响应主机。



TPU-1 block diagram. Source: Google.

Figure 1 TPU-1 from Google

该框图已转换为图 2 中的体系结构模型。

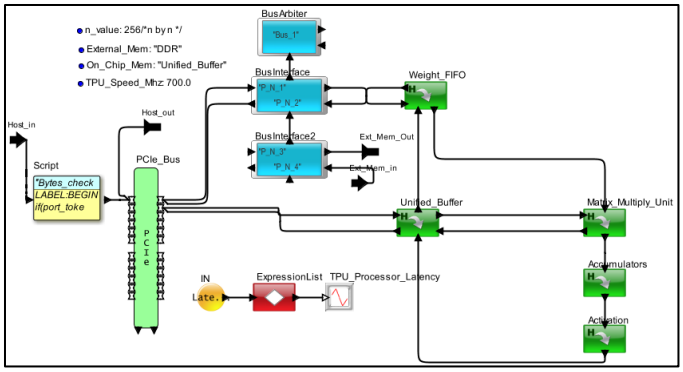


Figure 2 Top view of a GenesisArch Model of the AI hardware architecture

在图 3 中，您可以查看片外 DDR3 中的延迟和反向传播权重管理。 延迟是从主机发送请求到接收响应的时间。 您将看到 TG3 和 TG4 能够保持低延迟，直到分别达到 200 us 和 350 us。 MM 和 TG2 在仿真初期开始缓冲。 这表明 TPU 配置不足以处理到达的负载和所需的处理。 TG3 和 TG4 的更高优先级已帮助其将运营维持了更长的时间。 MM，TG2，TG3 和 TG4 是来自独立主机的不同请求流

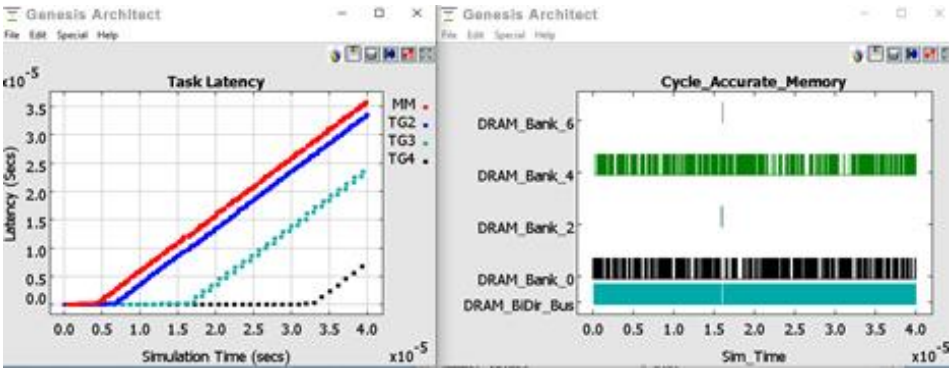


Figure 3 Statistics for the Architecture Exploration trade-off

三．AI 处理器的探索重点

AI 处理器和系统的设计师使用应用程序类型，训练与推论，成本点，功耗和尺寸限制进行实验。 例如，设计人员可以

- 将子网络分配给流水线阶段，
- 权衡深度神经网络（DNN）与常规机器学习算法之间的关系，
- 在 GPU，TPU，AI 处理器，FPGA 和常规处理器上测量算法性能，
- 评估融合计算和内存的好处 在芯片上，
- 计算类似于人脑功能的模拟技术对电源的影响，并
- 使用针对单个应用程序的部分功能构建 SoC。

在该示例中，架构探索的重点是内存访问。 有这么多的选择-SRAM 与 DRAM，本地与分布式存储，内存中计算以及缓存反向传播系数与丢弃。

1) 第二评估扇区是总线或网络拓扑。 虚拟原型可以具有用于处理器内部的片上网络，Tilelink 或 AMBA AXI 总线，用于连接多处理器板和机箱的 PCIe 或以太网，以及用于访问数据中心的 Wifi / 5G / Internet 路由器。

   - 使用虚拟原型的第三项研究是计算。 可以将其建模为处理器内核，多处理器，加速器，FPGA，多累加和模拟处理。

2) 最后一部分是传感器，网络，数学运算，DMA，自定义逻辑，仲裁器，调度程序和控制功能的接口。

用例和流量模式适用于组装成硬件，RTOS 和网络的组合的体系结构模型。 交通概况可以是周期性的，例如雷达，激光雷达和照相机，而用例可以是自动驾驶，聊天机器人，搜索，学习，推理，大数据操纵，图像识别和疾病检测。 对于输入速率，数据大小，处理时间，优先级，相关性，先决条件，反向传播循环，系数，任务图和内存访问，用例和流量会有所不同。 通过改变属性在系统模型上模拟用例。 这将导致生成各种统计信息和图表，包括高速缓存命中率，管线利用率，拒绝的请求数，每条指令或任务的瓦特数，吞吐量，缓冲区占用率和状态图。

图 4 显示了系统或芯片的功耗。 除了散热，电池电量消耗率和电池寿命周期变化外，该模型还可以捕获动态功率变化。

Figure 4 Measure the power consumption in real-time for an AI processor

该模型绘制了每个设备的状态活动，相关的瞬时尖峰和系统的平均功率。 尽早获得有关功耗的反馈，有助于热力和机械团队设计外壳和冷却方法。 大多数机箱对每个板都有最大的功率限制。 此早期功耗信息可用于在性能与性能之间进行权衡，从而寻找降低功耗的方法。

以下是一些其他 SoC 示例，重点介绍了 AI 体系结构模型和分析的使用。

1.自动驾驶系统

-将 360 度激光扫描仪，立体摄像机，鱼眼镜头，毫米波雷达，声纳和激光雷达连接到通过网关连接的多个 IEEE802.1Q 网络上的 20 个 ECU。

-该模型用于测试功能包的 OEM 硬件配置，以确定硬件和网络要求。 主动安全措施的响应时间是主要标准。

2.用于学习和推理任务的 AI 处理器

- 是使用片上网络主干定义的，该主干构建有 32 个内核，32 个加速器，4 个 HBM2.0、8 个 DDR5，多个 DMA 和完整的缓存一致性。
- 该模型在 RISC-V，ARM Z1 和专有内核上进行了试验。 实现的目标是在链路上达到 40Gbps，同时保持较低的路由器频率并重新训练网络路由。

3. 32 层深度神经网络

- 需要将内存从 40GB 减少到 7GB 以下。 数据吞吐量和响应时间未更改。
- 通过行为的功能流程图以及处理和反向传播的内存访问来设置模型。
- 对于不同的数据大小和任务图，该模型确定了数据的丢弃量以及各种片外 DRAM 大小和 SSD 存储选项。任务图随任意数量的图以及几个输入和输出而变化。

4. 使用 ARM 处理器和 AXI 总线进行低成本 AI 处理的通用 SoC。

- 目标是获得最低的每瓦功率，从而最大化内存带宽。 乘法累加功能被卸载到矢量指令，加密到 IP 内核，定制算法到加速器。
- 构建模型的明确目的是评估不同的缓存内存层次结构，以增加命中率和总线拓扑结构，以减少延迟。

5. 模数 AI 处理器

- 需要对功耗进行彻底分析，并对获得的吞吐量进行准确分析。
- 在此模型中，非线性控制在离散事件模拟器中建模为一系列线性函数，以加快仿真时间。
- 在这种情况下，对功能进行了测试，以检查行为并测量真正的节能效果。

四．人工智能系统示例 – 自动驾驶

考虑自动驾驶（ADAS）应用程序，它是图 5 中 AI 部署的一种形式。ADAS 应用程序与计算机或电子控制单元（ECU）以及网络上的许多应用程序共存。 为了使 ADAS 任务正确运行，还需要依赖这些现有系统的传感器和执行器。

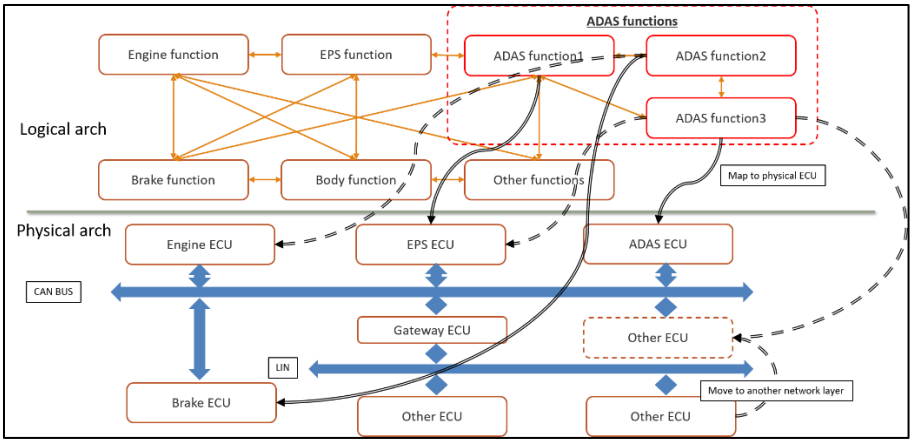


Figure 5 Logical to Physical Architecture of the AI Applications in an Automotive Design

早期的架构权衡可以测试和评估假设，以快速识别瓶颈，并优化规格以满足时序，吞吐量，功率和功能要求。 在图 1 中，您将看到该体系结构模型需要硬件，网络，应用程序任务，传感器，衰减器和流量激励来获得整个系统运行的可见性。

图 6 显示了此 ADAS 逻辑体系结构映射到物理体系结构的实现。 体系结构模型的一个不错的功能是能够分离设计的所有部分，从而可以研究单个操作的性能。 您会注意到现有任务是单独列出的，具有 ECU，传感器生成和 ADAS 逻辑任务组织的网络。 ADAS 任务图中的每个功能都映射到 ECU。

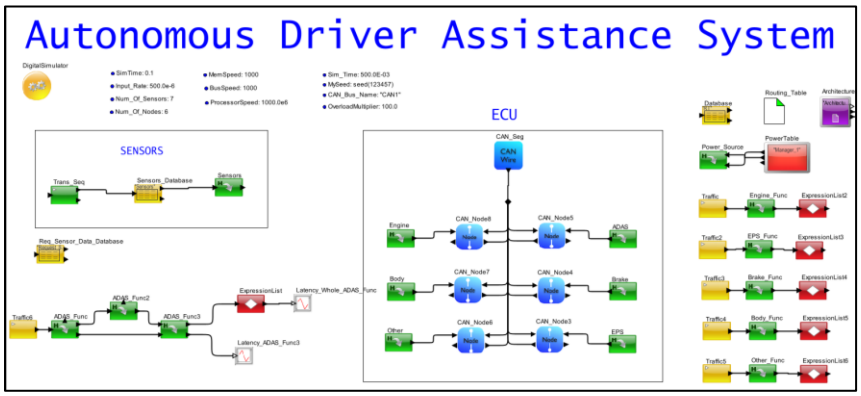


Figure 6 System Model of Automotive System with ADAS Mapped to the ECU Network

模拟 ADAS 模型后，您可以获得各种报告。 在图 7 中，显示了完成 ADAS 任务的等待时间以及与此任务相关的电池散热。 其他感兴趣的图可以是测量的功率，网络吞吐量，电池消耗，CPU 利用率和缓冲区占用率。

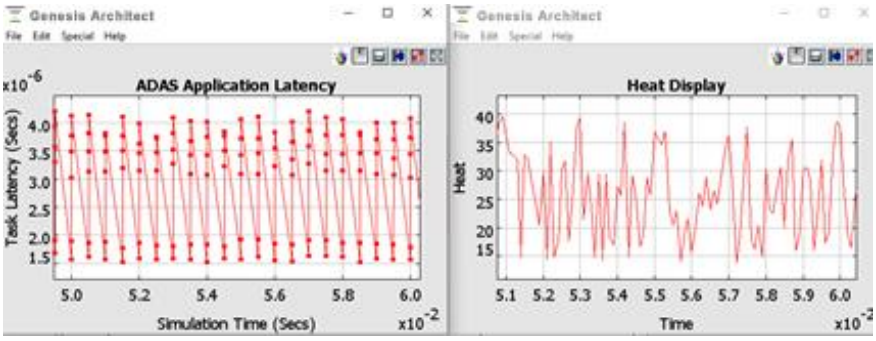


Figure 7 Analysis reports from the ADAS Architecture Model

图 8 是包含 ECU，CAN-FD，以太网和网关的网络框图。

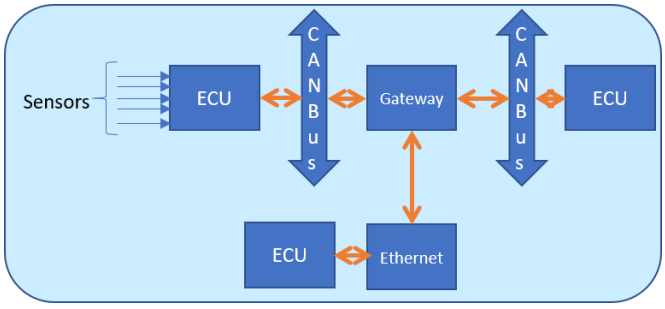


Figure 1 Automotive Network with CAN Bus, Sensors and ECU

图 9 捕获了图 8 的一部分，该部分将 CAN-FD 网络与包含多个 ARM 内核和 GPU 的高性能 Nvidia DrivePX 集成在一起。 从模型中删除了以太网/ TSN / AVB 和网关，以简化视图。

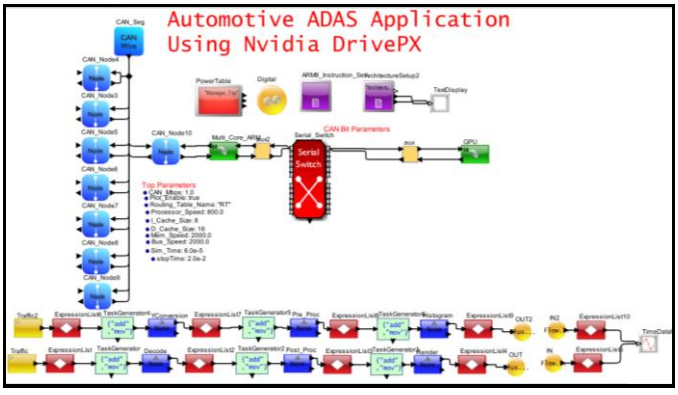


Figure 2 GenesisArch model of an Autonomous Driving and E/E Architecture

在此模型中，重点是了解 SoC 的内部行为。 该应用程序是由车辆上的摄像头传感器触发的 MPEG 视频捕获，处理和渲染。 图 10 显示了 AMBA 总线和 DDR3 内存的统计信息。 您可以看到跨多个主服务器的工作负载分配。 可以评估应用程序管道是否存在瓶颈，确定最高循环时间任务，内存使用情况配置文件以及每个任务的延迟。

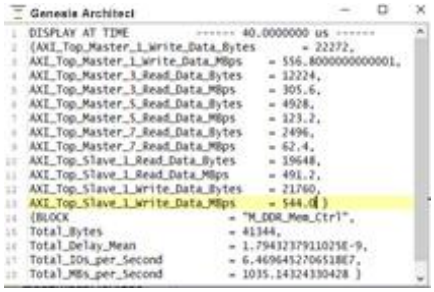


Figure 3 Bus and memory activity report

五．总结

GenesisArch 的用户可以在带有大型库 AI 硬件和软件建模组件的图形离散事件模拟平台中非常快速地构建架构模型。 该模型可用于进行时序，吞吐量，功耗和服务质量的权衡。 GenesisArch 中可用的库支持模拟，内存，处理器，RTOS，加速器，乘法累加单元，DMA，网络接口，总线，网桥，FIFO，缓冲区，调度程序和仲裁方案。 提供了 20 多个 AI 处理器和嵌入式系统模板（参考设计），以加速新 AI 应用程序的开发。 为在 AI 系统中进行权衡而生成的报告包括响应时间，吞吐量，缓冲区占用率，平均功率，能耗和资源效率。

该体系结构模型还可以生成文档并用于演示目的。 GenesisArch 能够以最小的配置，成本，功耗以及按时完成任务的能力来提供更高质量的产品。