数字赛题说明
Digital Competition Description

竞赛课题：
基于AI智能体的NPU设计与验证
Competition Topic:
AI Agent-Based NPU Design and Verification

一、竞赛任务
I. Competition Task

本赛题要求参赛队伍设计一款NPU（神经网络处理器），实现FastViT-T8图像分类网络的推理功能。FastViT-T8是Apple于2023年在ICCV会议上发表的轻量级视觉Transformer模型，具有约400万参数，计算量为0.7 GFLOPs，在ImageNet-1K数据集上达到76.2%的Top-1准确率。

参赛队伍需要自行选择大语言模型（如Claude、GPT-4、Qwen等）并搭建智能体框架，通过智能体技术自动化或半自动化地完成NPU的设计、验证和优化工作。

This competition requires participating teams to design an NPU (Neural Processing Unit) that implements the inference function of the FastViT-T8 image classification network. FastViT-T8 is a lightweight vision transformer model published by Apple at the ICCV conference in 2023. It has approximately 4 million parameters, a computational cost of 0.7 GFLOPs, and achieves a Top-1 accuracy of 76.2% on the ImageNet-1K dataset.

Participating teams need to select a large language model (e.g., Claude, GPT-4, Qwen, etc.) and build an agent framework independently. The design, verification, and optimization of the NPU must be completed automatically or semi-automatically by the agent.

二、设计规范
II. Design Specifications

2.1 输入输出要求
2.1 Input/Output Requirements

NPU需要接收256×256像素的RGB图像作为输入，图像数据格式为INT8。完整运行FastViT-T8的推理流程，输出为1000类分类结果，对应ImageNet-1K数据集的类别。设计可以采用INT8量化推理，也可以选择FP16或混合精度方案。

The NPU shall receive a 256×256 pixel RGB image as input, with image data formatted as INT8. Upon completing the full inference process of FastViT-T8, it shall output classification results across 1000 classes, corresponding to the ImageNet-1K dataset categories. The design may use INT8 quantization inference, or opt for FP16 or mixed-precision schemes.

2.2 验证要求
2.2 Verification Requirements

参赛队伍需要提供完整的功能仿真环境，证明设计的正确性。验证时至少需要在5个测试样本上进行测试，并提供在ImageNet上的精度表现。

参赛队伍需要采用上海合见工业软件集团股份有限公司推出的数字设计AI智能平台UniVista Design Assistant (UDA) 作为参考基准（Baseline），在提交的汇报中需要详细分析相比于基准方案的改进效果以及改进措施。UDA工具的获取和部署方案后续另行通知。

Participating teams must provide a complete functional simulation environment to demonstrate the correctness of their design. Verification must be tested on at least five test samples, and the accuracy performance on ImageNet must be reported.

Participating teams are required to use the digital design AI intelligent platform UniVista Design Assistant (UDA), launched by Shanghai United Imaging Industry Software Group Co., Ltd., as a reference benchmark. The submitted report must include a detailed analysis of the improvements and measures compared to the baseline solution. The method for obtaining and deploying the UDA tool will be notified later.

三、评分体系
III. Scoring System

竞赛采用基础分加权重分的评分方式。完成基本功能并通过验证的队伍可获得基础分60分，剩余40分根据精度、性能、功耗和面积四个维度的表现进行加权评分。

The competition adopts a scoring method combining a base score and weighted scores. Teams that complete the basic functionality and pass verification will receive a base score of 60 points. The remaining 40 points will be awarded based on a weighted evaluation of performance across four dimensions: accuracy, performance, power, and area.

3.1 基础分（60分）
3.1 Base Score (60 points)

设计能够完整运行FastViT-T8推理流程，并在至少5个测试样本上得到合理的分类输出，即可获得60分基础分。这里的"合理输出"是指输出格式正确，数值在预期范围内，不要求达到特定的准确率。

A design that can successfully run the complete FastViT-T8 inference process and obtain reasonable classification outputs on at least five test samples will receive the 60-point base score. "Reasonable output" here means the output format is correct and the values are within an expected range; achieving specific accuracy rates is not required.

3.2 性能评分（40分）
3.2 Performance Score (40 points)

剩余40分按照以下权重分配：精度占50%（20分），性能和功耗各占25%（各10分），面积占较小比重，在性能和功耗评分中综合考虑，面积和功耗均在SKY130工艺下得到。

The remaining 40 points are allocated according to the following weights: Accuracy accounts for 50% (20 points), Performance and Power each account for 25% (10 points each). Area carries a smaller weight and is considered comprehensively within the Performance and Power scores. Both area and power are evaluated under the SKY130 process.

精度评分（20分）采用归一化方式计算。所有通过基础验证的队伍中，在ImageNet-1K验证集（或组委会提供的测试子集）上测试Top-1准确率，准确率最高的队伍得20分，其他队伍按比例折算。计算公式为：精度得分 = (本队准确率 / 最高准确率) × 20分。

Accuracy Score (20 points): Calculated using a normalization method. Among all teams that pass the basic verification, the Top-1 accuracy is tested on the ImageNet-1K validation set (or a test subset provided by the organizing committee). The team with the highest accuracy will receive 20 points. Other teams' scores will be scaled proportionally. Calculation formula: Accuracy Score = (Team's Accuracy / Highest Accuracy) × 20 points.

性能评分（10分）主要考察推理吞吐量和延迟。吞吐量定义为每秒能够处理的图像数量，延迟定义为处理单张图像所需的时间。评分时综合考虑这两个指标，吞吐量越高、延迟越低的设计得分越高。采用归一化方式，性能最优的队伍得10分，其他队伍按比例折算。

Performance Score (10 points): Primarily examines inference throughput and latency. Throughput is defined as the number of images processed per second, and latency is defined as the time required to process a single image. Scoring comprehensively considers both metrics; designs with higher throughput and lower latency receive higher scores. Normalization is used, with the best-performing team receiving 10 points, and other teams' scores scaled proportionally.

功耗评分（10分）考察NPU在推理过程中的平均功耗。功耗最低的队伍得10分，其他队伍按照最低功耗与本队功耗的比值折算得分。功耗评估需要基于综合后的门级网络进行功耗分析。

面积在性能和功耗评分中作为参考因素。在性能或功耗相近的情况下，面积更小的设计将获得更高的评价。

Power Score (10 points): Examines the average power consumption of the NPU during inference. The team with the lowest power consumption receives 10 points. Other teams' scores are calculated based on the ratio of the lowest power consumption to their own power consumption. Power evaluation must be based on power analysis performed on the post-synthesis gate-level netlist.

Area is considered as a reference factor within the Performance and Power scores. In cases where performance or power are similar, designs with smaller area will receive a higher evaluation.

创新加分（最高10分）
Innovation Bonus Points (Up to 10 points)

在智能体框架设计、NPU架构创新、优化技术等方面有突出表现的队伍可以获得额外加分，最高10分。智能体框架的自动化程度和通用性可获得最高5分加分，NPU架构的创新性（如新颖的数据流设计、存储层次优化）可获得最高3分加分，其他优化技术（如稀疏化、动态精度调整）可获得最高2分加分。

Teams demonstrating outstanding contributions in areas such as agent framework design, NPU architecture innovation, or optimization techniques may receive additional bonus points, up to a maximum of 10 points. The automation level and versatility of the agent framework can earn up to 5 bonus points. The innovativeness of the NPU architecture (e.g., novel dataflow designs, memory hierarchy optimization) can earn up to 3 bonus points. Other optimization techniques (e.g., sparsity, dynamic precision adjustment) can earn up to 2 bonus points.

四、提交材料
IV. Submission Materials

4.1 技术报告
4.1 Technical Report

参赛队伍需要提交一份的技术报告（PDF格式），内容包括智能体框架的设计思路、NPU架构方案、量化策略与精度分析、PPA优化方法、用户与智能体框架的交互日志和人工介入情况等。报告应清晰说明设计思路和关键技术选择。

Participating teams must submit a technical report (PDF format), including the design concept of the agent framework, the NPU architecture scheme, quantization strategy and accuracy analysis, PPA optimization methods, interaction logs between the user and the agent framework, and any manual intervention involved. The report should clearly explain the design ideas and key technical choices.

4.2 源代码与脚本
4.2 Source Code and Scripts

需要提交完整的RTL设计代码（支持Verilog、SystemVerilog、Chisel等硬件描述语言），以及智能体相关的脚本和配置文件。同时需要提供仿真测试平台的代码和综合脚本。所有代码应有适当的注释，便于评审。

Complete RTL design code (supporting hardware description languages such as Verilog, SystemVerilog, Chisel, etc.) must be submitted, along with agent-related scripts and configuration files. Simulation testbench code and synthesis scripts must also be provided. All code should have appropriate comments to facilitate review.

4.3 验证报告
4.3 Verification Report

验证报告需要包含功能仿真波形、至少五个端到端的推理样例、综合报告（面积、时序数据）以及功耗分析报告。

The verification report needs to include functional simulation waveforms, at least five end-to-end inference examples, a synthesis report (area, timing data), and a power analysis report.

五、参考资源
V. Reference Resources

5.1 模型与数据
5.1 Model and Data

FastViT-T8的预训练模型可以从PyTorch的timm库获取（模型名称为fastvit_t8.apple_in1k），也可以从Apple官方GitHub仓库（https://github.com/apple/ml-fastvit）或Hugging Face下载。

The pre-trained model for FastViT-T8 can be obtained from PyTorch's timm library (model name: fastvit_t8.apple_in1k), or downloaded from Apple's official GitHub repository (https://github.com/apple/ml-fastvit) or Hugging Face.