TPU-lite-SoC 模型结构描述

Ⅰ. 模型基础信息

模型类型：MLP；　推理模式：多 batch 推理，batch_size = 16

Ⅱ. 数据定义

张量名称	维度	数据类型	说明
模型输入	(16, 256)	INT8	16 张图片*256 个像素输入
权重矩阵 1（W1）	(256, 64)	INT8	第 1 层全连接层权重
权重矩阵 2（W2）	(64, 32)	INT8	第 2 层全连接层权重
权重矩阵 3（W3）	(32, 16)	INT8	第 3 层全连接层权重
偏置 1（b1）	(1, 64)	INT32	第 1 层全连接层偏置
偏置 2（b2）	(1, 32)	INT32	第 2 层全连接层偏置
偏置 3（b3）	(1, 16)	INT32	第 3 层全连接层偏置
缩放因子 1（scale1）	-	FP32	第 1 层输出量化缩放因子
缩放因子 2（scale2）	-	FP32	第 2 层输出量化缩放因子
缩放因子 3（scale3）	-	FP32	第 3 层输出量化缩放因子

Ⅲ. 多 batch 推理计算流程

 # 第 1 层全连接计算

1. 矩阵乘法：输入矩阵 × W1 → 矩阵 A (16, 64)，数据类型 INT32
2. 偏置相加：矩阵 A 逐行+ b1 → 矩阵 B (16, 64)，数据类型 INT32
3. 量化缩放：矩阵 B 逐元素 × scale1 → 矩阵 C (16, 64)，数据类型 INT8
4. 激活函数：矩阵 C 经过 ReLU 激活 → 矩阵 D (16, 64)，数据类型 INT8

 # 第 2 层全连接计算

5. 矩阵乘法：矩阵 D × W2 → 矩阵 E (16, 32)，数据类型 INT32
6. 偏置相加：矩阵 E 逐行 + b2 → 矩阵 F (16, 32)，数据类型 INT32
7. 量化缩放：矩阵 F 逐元素 × scale2 → 矩阵 G (16, 32)，数据类型 INT8
8. 激活函数：矩阵 G 经过 ReLU 激活 → 矩阵 H (16, 32)，数据类型 INT8

 # 第 3 层全连接计算

9. 矩阵乘法：矩阵 H × W3 → 矩阵 I (16, 16)，数据类型 INT32
10. 偏置相加：矩阵 I 逐行 + b3 → 矩阵 J (16, 16)，数据类型 INT32
11. 量化缩放：矩阵 J 逐元素 × scale3 → 矩阵 K (16, 16)，数据类型 INT8

Ⅳ. 分类结果输出

1. 张量特征：矩阵 K[i] 的后 6 个元素恒为 0
2. 结果判定：矩阵 K[i]前 10 个元素中最大值对应的索引即为数字 0-9 分类结果

Ⅴ．第 1 层全连接计算流程细化

用 **A**，**W** 表示一个矩阵，大小是 16 × 16 个 INT8 数据，那么

输入矩阵可以被分块表示为：$[A_1, A_2, A_3, \dots, A_{16}]$

权重矩阵可以被分块表示为：

$$\begin{bmatrix} W_{11}, & W_{12}, & W_{13}, & W_{14} \\ W_{21}, & W_{22}, & W_{23}, & W_{24} \\ \vdots & & & \vdots \\ W_{16,1}, & W_{16,2}, & W_{16,3}, & W_{16,4} \end{bmatrix}$$

下述是按模块划分的分步计算流程，绿色表示上一步中 VPU 得到/保存的数据

Systolic Array 计算得到 A_1W_{11}，A_1W_{12}，A_1W_{13}，A_1W_{14}

VPU 存入数据

Systolic Array 计算得到 A_2W_{21}，A_2W_{22}，A_2W_{23}，A_2W_{24}

VPU 计算得到 $A_1W_{11}+A_2W_{21}$，$A_1W_{12}+A_2W_{22}$，$A_1W_{13}+A_2W_{23}$，$A_1W_{14}+A_2W_{24}$ 并保存

Systolic Array 计算得到 A_3W_{31}，A_3W_{32}，A_3W_{33}，A_3W_{34}

VPU 计算得到 $A_1W_{11}+A_2W_{21}+A_3W_{31}$，$A_1W_{12}+A_2W_{22}+A_3W_{32}$，$A_1W_{13}+A_2W_{23}+A_3W_{33}$，$A_1W_{14}+A_2W_{24}+A_3W_{34}$ 并保存

……

Systolic Array 计算得到 $A_{16}W_{16,1}$，$A_{16}W_{16,2}$，$A_{16}W_{16,3}$，$A_{16}W_{16,4}$

VPU 计算得到 $A_1W_{11}+\dots+A_{16}W_{16,1}$，$A_1W_{12}+\dots+A_{16}W_{16,2}$，$A_1W_{13}+\dots+A_{16}W_{16,3}$，$A_1W_{14}+\dots+A_{16}W_{16,4}$ 并保存

VPU 将数据视为 16 × 64 个 INT32 的矩阵，每行与偏置 1 相加

VPU 将数据每个元素 × scale1，处理到 INT8

VPU 将数据每个元素进行 ReLU

VPU 将数据（16 × 64 个 INT8）写回 Unified Buffer