

## Titan-TPU V2 核心知识速通指南 (NotebookLM 专用)

请将以下问题依次发送给 NotebookLM，以解锁 TPU 架构的核心知识。

### 第一阶段：架构宏观理解 (The Big Picture)

目标：理解 TPU 为什么长这样，以及它是如何工作的。

1. **Prompt:** "请用通俗易懂的语言解释 Titan-TPU V2 的 'Weight Stationary' (权重驻留) 数据流。在这个架构中，输入数据、权重和部分和(Partial Sum)分别是朝着哪个方向流动的？"
2. **Prompt:** "根据我上传的文档，解释一下 Systolic Array (脉动阵列) 相比于普通 CPU 核心，在做矩阵乘法时有什么巨大的优势？请重点从内存带宽的角度分析。"
3. **Prompt:** "Titan-TPU V2 的 Unified Buffer 是做什么用的？为什么不直接从 DDR 内存读取数据进入 PE？"

### 第二阶段：深入代码细节 (Code Deep Dive)

目标：建立代码与原理的联系，为看懂 `pe.sv` 做准备。

4. **Prompt:** "结合 `pe.sv` 代码和架构文档，解释 PE 模块中的 `weight_reg_active` 和 `weight_reg_inactive` 是什么关系？为什么需要两个权重寄存器？(提示：关键词 Shadow Register)"
5. **Prompt:** "在 PE 代码中，MAC (乘累加) 运算的具体逻辑是在哪几行实现的？当 `pe_switch` 信号到来时，硬件会发生什么具体的行为？"
6. **Prompt:** "解释一下数据在脉动阵列中的 'Skew' (错位/倾斜) 是什么意思？为什么输入数据不能同时到达第一列 PE，而需要延时进入？"

### 第三阶段：指令与控制 (ISA & Control)

目标：理解软件是如何指挥硬件干活的。

7. **Prompt:** "我想让 TPU 执行一次矩阵乘法。根据 ISA 手册，我需要按什么顺序配置哪些寄存器？请列出一个简单的步骤清单。"
8. **Prompt:** "解释一下 `load_weight` 指令和 `mat_mul_start` 指令在硬件执行层面的区别。哪一个会触发数据从 Unified Buffer 流向脉动阵列？"

### 第四阶段：魔改准备 (Magic Mod Prep)

目标：为后续添加 AXI 总线做准备。

9. **Prompt:** "根据 AXI 协议规范文档(如果已上传)，总结一下 AXI-Lite 接口的写操作握手流程。在我的 TPU 项目中，这个接口主要用于什么目的？"