PRF功能描述和实现原理

功能描述

在本次处理器设计中，物理寄存器文件（PRF）系统由两个核心子模块PRF\_Value和PRF\_Valid协同构成。PRF\_Value负责物理寄存器值的存储与读写操作，维护深度为PRF\_DEPTH的寄存器阵列，支持6个功能单元写回端口和5个指令发射读取端口。PRF\_Valid管理物理寄存器的状态有效性，跟踪寄存器就绪状态并在指令提交时释放寄存器资源。两者协同实现寄存器重命名机制，为乱序执行架构提供寄存器操作数服务，同时通过AMT接口与重命名单元保持架构寄存器映射同步，支持分支预测错误时的精确状态恢复。

实现原理

PRF\_Value模块使用向量寄存器阵列regBank存储物理寄存器值，其中寄存器0硬连线为0值。该模块聚合5类功能单元（2个ALU、1个BU、1个MUL、1个DIV和1个LDU）的写回操作，通过地址/数据分离总线（writeAddr/writeData）实现同步写入。读取端口分为三组：双发射EXU支持两路并行双操作数读取（4个端口），ST指令读取基址/偏移量操作数，LD指令读取基址操作数，所有读取均为组合逻辑直接索引。

PRF\_Valid模块通过prf\_valid向量跟踪寄存器状态，初始化时仅寄存器0有效。当功能单元写回结果时，对应目标寄存器标记为有效。指令提交阶段，根据ROB提交信号解析需要释放的寄存器：通过hasPd函数筛选算术/跳转/CSR类指令的目标寄存器，当双提交槽目标寄存器冲突时释放最新物理寄存器。分支预测错误时触发状态恢复：用ROB提交的旧映射覆盖当前AMT，通过in\_amt函数检测AMT关联寄存器，仅保留映射寄存器状态其余置无效。

系统采用值-状态解耦架构优化时序路径，通过rob\_valid\_bits状态机处理双提交冲突。写端口聚合管理简化数据通路，零寄存器（x0）硬连线实现符合RISC-V规范。参数化配置支持不同规模处理器实现，包括PRF深度和功能单元数量等关键参数。