

# 胡傲东

18085741680 [hu.aodong@iloud.com](mailto:hu.aodong@iloud.com) GitHub 个人主页

## 核心竞争力

**全栈工程师 + AI 多模态开发者**, 侧重于**客户端架构设计与端到端 AI 应用落地**。具备复杂系统分析与设计能力, 擅长将 AI 能力与工程实践深度融合, 具有良好的项目规范管理与团队协作能力。

## 教育背景

华中科技大学 · 集成电路设计与集成系统 (本科)

2022.09 - 2026.06

## 专业技能

语言 / 框架

C++、Python/PyTorch/FastAPI、Swift/SwiftUI、  
TypeScript/React Native/Expo

AI / 多模态

LLM Agent (微调/Tool Calling)、SD/SVD 图像视频生成、  
TTS 语音合成、GraphRAG、pHash/LSH 相似检测

系统 / 平台

macOS 原生开发 (App Sandbox/Notarization/XPC)、  
Linux 运维、Docker 容器化、PostgreSQL/Redis/MySQL

架构 / 工具

MVVM/Clean Architecture、GDB/LLDB 调试、  
CMake/Makefile 构建、Git 协作、CI/CD

## 项目经历

### 全栈 故事成片：跨平台 AI 视频生成系统

字节工程营 · 小组长 (剪映客户端/多模态) · 2025.11 - 至今

**项目定位:** 端到端 AI 视频生成系统 (文本→智能分镜→图像/配音/视频→FFmpeg 合成), Windows/macOS 桌面端支持实时编辑/预览/导出  
**核心职责:** 小组长 (5人), 主导架构设计与任务分工, 建立 Code Review + CI/CD 流程, 2 周内完成 MVP 上线

**技术挑战:** 多模态任务 (图像/语音/视频) 并行执行易导致 OOM 和任务阻塞; 设计任务网关 + 优先级队列实现智能调度 (暂停/恢复/取消/重试/降级), 通过任务拆分 + 异步执行解决阻塞问题, 长任务中断恢复率 95%+, 内存峰值降低 30%, prompt 优化, 提升生成视频质量

**核心成果:** 8GB 显存下稳定生成 30s 720p 视频; 缓存复用率从 25% 提升至 65% (+40%), 减少重复 SD 推理; 渲染帧率从 30fps 优化至 60fps

**技术栈:** Qt/QML/C++ · Go/Gin · FastAPI · Redis/Async · Docker · FFmpeg · Qwen/SD/SVD/TTS · MySQL/MinIO

### 全栈 AI 智能教学平台

本科毕业设计 · 全栈开发 · 2025.11 - 至今

**项目定位:** 以学生为中心的高校 AI 教学平台, 支持学习状态追踪、个性化答疑/批改与学习建议; 覆盖桌面端、企业微信 H5 及 iOS/Android 原生 App, 与导师合作在华中科技大学《电磁场与电磁波》(本科) 与《学术规范与论文写作》(研究生) 课程试点 (初期预计覆盖 200+ 研究生)

**核心职责:** 独立完成全栈开发 (React 前端 / Go 后端 / Python AI 服务), 构建跨端应用, 形成移动端/桌面端多端覆盖

**架构重构:** 发现现有平台以课程为核心, 无法追踪学生跨课程学习进展; 重构为学生中心架构, 设计学生画像 + 学习档案, 支持长期学情追踪与知识点薄弱分析, 并引入 LLM Agent (微调/Tool Calling) 实现个性化答疑, 引入 FPGA/NPU 加速 AI 推理

**技术选型:** 对比 BM25/向量检索/Hybrid 检索后选择 GraphRAG, 解决知识点关联性检索问题, 答疑相关性从 68% 提升至 85%+

**核心成果:** Docker Compose 一键部署, 支持 30+ 并发用户; AI 答疑准确率 85%+, 作业批改覆盖 80% 常见题型

**技术栈:** React/Vite · Go/Gin/GORM · Python/FastAPI · MySQL/MinIO · Docker Compose · Qwen3 · GraphRAG · iOS/Android

### Apple SwiftSweep · macOS 系统优化工具

个人开源 · 2025.12 - 至今

**项目定位:** 原生 macOS 系统优化工具 (GUI + CLI), 覆盖系统监控/磁盘分析/卸载残留/包与仓库管理/相似媒体检测等 14+ 模块

**安全设计:** 为避免用户误删系统文件, 设计统一清理链路 (dry-run 预览 + allowlist 白名单 + symlink 逃逸检测), 零误删事故

**权限隔离:** 为避免主进程获取 root 权限带来安全风险, 采用 XPC Helper 实现特权操作隔离, 通过 SMApService 注册系统级 Helper

**核心成果:** pHash + LSH 实现相似图片/视频检测; 实现鉴权机制和用户数据隔离; 实现插件系统; 通过 Developer ID 签名 + Notarization 发布

**技术栈:** Swift 5.9 · SwiftUI · SwiftPM · XPC Helper/SMApService · Argument Parser · pHash/LSH

### 系统 RISC-V CPU 设计

集创赛二等奖 · 2025.03 - 2025.07

**项目定位:** 基于 RV32IMACFG 指令集的四路超标量处理器设计与验证, 支持动态分支预测与乱序执行

**技术选型:** 引入动态分支预测, 选择 gshare 算法, 分支预测准确率从 50% 提升至 70% (+20%), IPC 从 0.8 提升至 1.2

**核心成果:** 基于 Python + Makefile 构建 CI 流水线, 设计 50+ 测试用例覆盖指令集; 成功移植 RT-Thread RTOS 并运行 Dhrystone 基准测试

**技术栈:** SystemVerilog · C++ · Python · Makefile · Scala/SpinalHDL

## 校园经历

启明学院 Dian 团队 · 华为海思组 (技术): 参与 C++ 网络仿真平台开发, 负责时延算法 | 媒体组 (运营): 运营团队公众号

华中科技大学贵州省招生组 · 核心骨干: 主导 500+ 人社群管理, 组织线上线下宣讲, 覆盖数百人