

架构演进对比报告：v2.3 (Legacy) vs v3.1.2 (Async Core)

日期: 2025-12-21

分析对象:

- 旧版: `log/okx_deepseek.py` (v2.3 单体脚本)
- 新版: `src/OKXBot_Plus.py` (v3.1.2 异步微服务架构)

1. 核心架构模式 (Core Architecture)

特性	v2.3 (旧版 - <code>okx_deepseek.py</code>)	v3.1.2 (新版 - Async Core)
编程模型	同步阻塞 (Synchronous) 依赖 <code>time.sleep</code> 和 <code>requests</code> , 串行处理。	异步非阻塞 (Asynchronous) 基于 <code>asyncio</code> + <code>aiohttp</code> , 全并发处理。
代码结构	单体巨石 (Monolithic) 所有逻辑 (API、策略、执行、风控) 挤在 1900 行的一个文件中。	模块化微服务 (Modular) 按领域拆分为 <code>core</code> , <code>strategy</code> , <code>execution</code> , <code>risk</code> 多个独立模块。
并发能力	低 (Low) 轮询 10 个币种需要 10-20 秒, 容易错过瞬时行情。	高 (High) 轮询 10 个币种仅需 1-2 秒, 网络 I/O 全部并行。
AI 推理	阻塞式 等待 DeepSeek 回复时, 整个程序卡死, 无法处理其他币种。	非阻塞式 等待 AI 时 CPU 继续处理其他任务, 吞吐量提升 10 倍。

2. 交易逻辑与风控 (Trading Logic & Risk)

功能点	v2.3 (旧版)	v3.1.2 (新版)	改进评价
下单数量	被动且保守 AI 经常给出 0.01 (最小单位), 代码仅做最低限额修正。	主动且智能 Prompt 注入最小单位上下文, 引导 AI 在高信心时大胆买入 (半仓/满仓)。	☆☆☆
双向交易	不对称 (跛脚) SELL 信号只平多, 不反手开空。	完全对称 SELL 信号 = 平多仓 + 开空仓 (Close Long + Open Short)。	☆☆☆
微利风控	死板 只要利润覆盖不了手续费就拦截, 哪怕大难临头也不让跑。	灵活 引入 AI 信心加权。如果 AI 喊救命 (HIGH Confidence), 强制放行逃顶。	☆☆
平仓通知	缺失 卖出/平仓时静悄悄, 用户以为机器人没动。	全覆盖 开仓、平仓、止损、止盈全都有 Webhook 通知。	☆☆
资金计算	有缺陷 合约模式下重复计算持仓市值, 导致基准资金虚高。	修正 直接读取 OKX 统一账户权益, 剔除重复计算, PnL 更精准。	☆☆

3. 用户体验 (UX & Operations)

体验点	v2.3 (旧版)	v3.1.2 (新版)
启动速度	慢 启动时串行预热 K 线, 币种多时要等很久。	快 并行预热, 几乎秒开。
日志查看	原始 依赖 <code>print</code> 输出, 需要手动 <code>tail -f</code> 。	现代 启动脚本自动进入实时监控, 日志格式化更好。
通知样式	纯文本 干巴巴的文字堆砌。	富文本卡片 飞书/钉钉卡片样式, 带标题和排版, 更专业。
配置管理	硬编码混杂 部分配置写在代码里, 部分在 JSON。	统一配置中心 所有配置收敛至 <code>config.json</code> 和 <code>.env</code> , 代码无硬编码。

4. 总结 (Conclusion)

v2.3 (okx_deepseek.py) 是一个典型的“个人脚本”，适合单币种、低频、小资金的实验性运行。它的逻辑虽然跑得通，但在并发性能、资金安全和交易灵活性上存在明显瓶颈。

v3.1.2 (Async Core) 则是一个“工程化系统”，具备了商业级量化软件的雏形：

1. 快：异步并发，不错过行情。
2. 稳：完善的异常捕获和超时熔断。
3. 准：修复了资金计算和交易逻辑的缺陷。
4. 狠：AI 敢于重仓（基于信心），风控敢于熔断。

建议：彻底弃用 `log/okx_deepseek.py`，全力维护 `src/` 下的新架构。