



产品特点介绍 · 客户参考指南

FPGA / 嵌入式功能与向量知识库

产品特点与客户知识复用能力介绍

Xg-Stellar 面向 FPGA/RTL 与嵌入式研发团队，将自然语言任务、工程工具、验证证据和企业知识连接为可复用的研发闭环。产品强调工程文件内执行、结果可检查、知识需确认、后续可回查。

文档导读

内容	阅读重点
FPGA / RTL	从工程识别、设计生成到仿真、检查和修复回归
嵌入式	从模块基线、单元测试到覆盖率、静态分析和板级准备
向量知识库	资料导入、语义检索、来源治理和领域隔离
客户知识复用	知识候选、人工确认、正式入库和回查验证

适用范围

适用于 Windows

桌面端及内网离线部署场景。知识检索结果用于研发参考，当前工程仍需执行适用的仿真、测试和检查。



01 | 产品定位

Xg-Stellar 是面向 FPGA/RTL 验证与嵌入式 C/C++ 开发的桌面端 AI 研发助手。它以自然语言交互为入口，在客户工程工作区内组织任务、调用确定性工具、反馈失败原因、迭代修复，并沉淀可追溯的验证记录。

核心价值

把“问模型”升级为“让智能体在工程中完成可检查的工作”，同时让企业既有资料和历史经验在后续项目中持续复用。

能力层	客户获得的能力	主要价值
智能交互	用自然语言描述模块、接口、测试与排障目标	降低工具使用门槛，减少上下文切换
FPGA/RTL	RTL/测试台生成、仿真、波形、静态规范、跨时钟域检查	形成设计-验证-修复闭环
嵌入式	测试工程、单元测试、编译运行、覆盖率、静态分析、板级准备	提升测试完整性与问题定位效率
知识复用	文档导入、语义检索、知识候选、确认入库、回查验证	把个人经验转化为团队可检索资产
部署安全	本地工作区、本地知识库、内网模型服务、可裁剪离线部署	研发数据与知识资产留在客户环境

02 | FPGA / RTL 开发辅助特点

FPGA 能力围绕真实工程文件工作，覆盖从需求整理到验证证据的连续过程。FPGA 与嵌入式保持任务和结论隔离，避免跨领域状态混用。

完整工作路径

- 1 识别当前工程中的 RTL、测试台、约束、脚本、波形和顶层模块候选。
- 2 把需求整理为接口、时钟、复位、位宽、关键时序和测试场景。
- 3 生成或修改 RTL 与自检测试台，优先修改真实文件，不重复创建平行实现。
- 4 执行仿真并按需生成波形，同时完成静态规范和跨时钟域检查。
- 5 根据报告区分 RTL、测试台和环境问题，修复后重跑相关检查。
- 6 输出修改文件、执行结果、未执行项、剩余风险和本轮验证记录。

能力亮点

- 支持 RTL 与自检测试台协同生成，减少“有代码、无验收”的情况。
- 支持仿真、波形、静态规范、跨时钟域检查等多类验证动作。
- 失败时按原因进入修复循环，不把环境缺失误判成设计缺陷。
- 验证记录与工程文件一起落位，便于复现、评审和回归。
- 可结合 FPGA 专用知识集合检索接口、CDC、复位、时序和板级经验。

结论边界

产品输出开发辅助和基础验证结果，不替代器件厂商原始规范、正式认证、板级验收或项目责任人的工程签字。

03 | 嵌入式开发辅助特点

嵌入式能力聚焦 C/C++ 模块开发与质量提升：先识别公共接口和测试环境，再生成或修改代码与测试，最后以编译、断言、覆盖率和静态分析结果推动修复。

阶段	主要动作	典型输出
工程识别	识别源码、头文件、测试、构建配置和目标平台	模块基线与缺失信息
任务整理	明确 API、输入输出、边界、错误处理和时间行为	开发任务卡
生成/修改	生成头文件、实现和基础测试；保留兼容接口	源码与测试变更
测试验证	编译运行、断言检查、覆盖率、静态分析	测试结果与风险清单
修复回归	按报告修复并重跑相关项目	可复现的验证记录
板级准备	检查制品、串口和调试条件；真实烧录需人工确认	板级调试记录

适用场景

- 算法模块、通信协议、驱动适配层和硬件抽象层的单元测试。
- 已有模块补测试、补覆盖率、定位越界、未初始化和资源管理风险。
- 新模块骨架生成、接口约束整理和回归用例构建。
- 板卡调试前的制品检查、串口验收条件和人工授权流程准备。

04 | XG-Lmy 向量知识库

XG-Lmy 将技术文档、芯片手册、接口规范、历史代码说明和已验证经验转换为可按语义检索的知识。用户不必记住文件名或关键词，也能通过问题找到含义接近的资料片段。

知识生命周期

环节	机制	治理重点
导入	解析文档、分块、生成向量并写入领域集合	记录来源、版本和内容哈希
检索	语义检索结合关键词，提高专有名词命中率	结果用于参考，不替代实际验证
引用	返回来源文档和片段依据	避免答案与资料来源脱节
候选	任务结束后提炼问题、根因、办法、证据和适用范围	模型推断不能直接成为企业知识
确认	用户审阅候选并使用专属确认短语授权入库	普通“可以”不视为授权
回查	入库后立即按标题、模块和问题现象检索	确认新知识确实可被找到

知识集合隔离

- FPGA 经验进入 fpga-experience，嵌入式经验进入 embedded-experience。
- 同领域、同项目、同模块和同主题使用稳定 sourceId，新版本替换旧分块。
- 客户敏感信息、密钥、口令、许可证和个人信息不得进入知识库。
- PDF 仅阅读或总结时不自动入库；存在 OCR 或解析警告时先核对再确认。

05 | 客户知识复用功能

客户知识复用不是简单上传文档，而是一条“来源可识别、内容可检索、经验需确认、结果可回查”的受控链路。

复用闭环

项目资料 → 分领域导入 → 研发任务检索 → 形成知识候选 → 用户确认 → 正式入库 → 回查验证 → 后续项目复用

复用对象	示例	后续用途
规范资料	芯片手册、接口协议、编码规范	生成设计前检索约束和边界
设计经验	复位策略、同步器选择、资源权衡	相似模块设计时提供依据
验证经验	自检测试台结构、断言、覆盖缺口	生成测试和回归计划
故障经验	现象、根因、修复、验证方法	相似故障快速定位
交付模板	评审清单、验证报告、上板记录	统一项目输出格式

对团队直接收益

- 减少资料散落在个人电脑、聊天记录和旧项目目录中的情况。
- 让新人可通过问题直接找到团队既有方法和适用边界。
- 复用经过验证的排障方法，同时保留来源和剩余风险。
- 同一问题的新版本知识替换旧分块，降低过期资料混检。
- 支持全离线或内网共享部署，知识资产不需要发送到公网。

06 | 部署、安全与适用边界

Xg-Stellar 支持 Windows 10/11 x64 桌面环境，可按客户授权和场景裁剪部署 FPGA、嵌入式及知识库能力。模型服务既可使用公网接口，也可连接内网私有化推理网关。

安全维度	产品机制
工程文件	智能体以当前工作区为根访问文件，生成物和记录落在工程内。
知识数据	知识库可完全本地运行；共享模式可使用令牌保护的内网服务。
模型接入	支持内网 OpenAI 兼容接口，研发数据可留在客户网络。
入库控制	阅读不等于入库；经验候选必须经过用户明确确认。
领域隔离	FPGA 与嵌入式不共享业务规格、任务状态和验证结论。
板级操作	真实烧录必须在桌面板级闭环页面由用户明确授权。

建议落地方式

- 1 选择一个 FPGA 项目和一个嵌入式模块作为试点，建立基础工程流程。
- 2 导入少量高价值资料，优先选择接口规范、芯片手册和验证清单。
- 3 用真实任务产生第一批知识候选，并由项目负责人审核入库。
- 4 约定 sourceId、版本、审核状态和敏感信息排除规则。
- 5 在后续项目中统计检索命中、复用次数和问题定位时间变化。

推荐试点指标

验证准备时间、测试补全时间、问题复现时间、知识检索命中率、已确认经验复用次数。