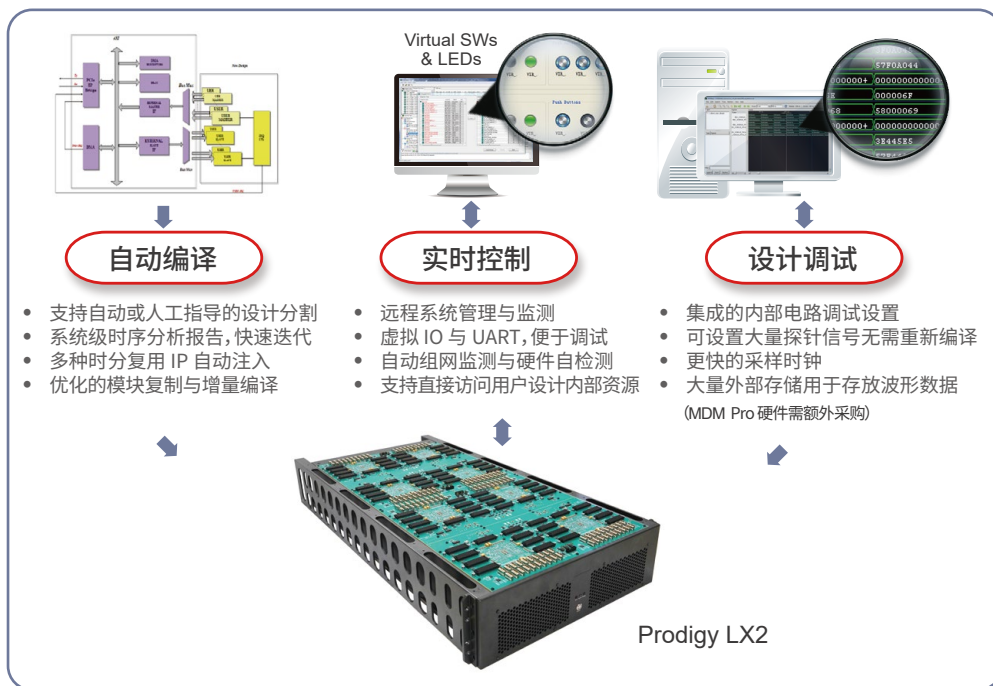


Prodigy™ Player Pro™

芯神瞳 FPGA 原型设计和调试工具

芯神瞳自动原型编译软件 - Prodigy Player Pro 是思尔芯针对其芯神瞳逻辑矩阵、逻辑系统和逻辑模块原型验证平台而设计的自动原型编译与调试 EDA 工具。包含完整的设计编译、运行控制、设计调试等功能，可将用户设计自动编译并加载至原型验证平台，实现高效的系统级验证和软件开发，加速芯片的开发进程。



自动编译 Player Pro - CompileTime

集成的 GUI 和 Tcl 脚本接口，可自动实现 RTL 至 FPGA 执行文件的完整编译流程，包括设计综合、自动分割、布局布线等，直至生成设计 bin 文件。

自动编译流程

Prodigy Player Pro - CompileTime 可通过图形化的界面引导，逐步完成所有编译步骤，并可以通过Tcl 模式自动执行 ECO 流程。

- 设计导入
- I/O 管脚分配
- 设置内部探针信号
- 运行布局布线
- 调用综合工具综合
- 生成执行文件
- 设计分割

I/O管脚分配

自动集成 S2C 子板 I/O 管脚对应关系的库文件，并可自动匹配到 I/O 连接器，提升效率并最小化了人为出错的几率。

设计分割

- 自动或人工指导分割设计到多颗FPGA
 - 用户指导分割配置以实现性能优化
 - 用户可灵活配置互连线缆的连接方案
- 支持自动设置黑盒, 节省设计编译时间
- 多种 TDM IP 自动注入, 提升系统性能
- 管脚复用逻辑自动插入, 待复用管脚信号预检, 以获得更好的性能
- 全系统时序分析, 加速设计迭代

实时控制 Player Pro - RunTime

Prodigy Player Pro - RunTime 可通过以太网或 USB 连接方式实现对芯神瞳逻辑矩阵、逻辑系统和逻辑模块原型验证平台的远程管理、控制与监测功能。

多配置方式

- 支持 USB 和以太网接口将设计文件直接下载到 FPGA 内
- 支持写入原型验证平台板载的 SD 卡内, 并从 SD 卡快速加载

虚拟“SWs 和 LED”

提供丰富的虚拟开关、指示灯、串口等, 便于远程调试。

- 虚拟状态灯可用于快速监测设计的状态
- 虚拟按钮和开关可用于快速设置设计的输入条件
- 虚拟串口用于设计的固件调试

硬件自检测

通过软件向导, 引导并帮助用户快速诊断 I/O 引脚、互连线、及全局时钟线等资源的状态。

远程系统控制

所有系统功能都可以通过 USB 或以太网进行远程控制。

- SD 卡插入时系统自动监测
- 轻松地监测 I/O 电压、电流和温度
- 自动识别插入子卡类别
- 在同一控制台下, 方便地控制多个原型验证平台

设计调试 Player Pro - DebugTime

Prodigy Player Pro - DebugTime 支持预先选择或设定待触发或追踪的内部信号, 从而实现多 FPGA 系统的联合调试。在运行调试过程中, 所选择的探针信号将被追踪和存储在专属 DDR4 内存, 进行数据分析与调试。

集成的内部电路调试设置

- 可标记 FPGA 内部信号, 以防综合时被优化
- 内部探针信号会根据设计分割的结果自动分配到多颗 FPGA 内
- 通过同一控制台设置多颗 FPGA 的触发和跟踪信号

设置大量的内部探针, 无需重新编译

- 可标记不限数量的 FPGA 内部探针信号
- 在 FPGA 无需重新编译的情况下, 每 FPGA 可以追踪到最多 16K 探针信号 (8 组, 每组 2K 探针信号)

运行调试

当设计调试的编译流程完成时, 可通过选配的 Prodigy Multi-Debug Module Pro (MDM Pro) 调试设备, 通过 Prodigy Player Pro - DebugTime 进行触发条件设置, 捕获并上传波形数据进行设计调试。

触发条件设置

用户可以通过 Prodigy Player Pro - DebugTime 调试面板设置触发事件和组合事件。

- 触发事件支持: ==, !=, >=, <=, >, < 和计数
- 组合活动支持: ! &, |, ^ - > 和计数
- 支持最多 8 个事件触发模块的组合
- 支持状态机脚本语言

多 FPGA 同时调试 (需要 MDM Pro 硬件)

- 使用单个逻辑分析器调试多颗 FPGA
- 通过高速收发器将多 FPGA 的触发和跟踪数据传输给 MDMHW
- 编写 VCD/FST 格式的采样数据以供分析
- 外部存储器支持最多存储 64GB 的波形数据

产品规格

适用的硬件

- 编译: VU+, VU, KU, S10 和 A10
- 实时控制: VU+, VU, KU, S10 和 A10
- 调试设置: VU+ 和 VU

适用的语言

- Verilog / VHDL
- System Verilog
- EDIF
- 混合编程语言

适用的 OS

- 自动编译:
64 位 RHEL7.7/7.8, Ubuntu 16.04/18.04 & CentOS 7.3/7.6
- 实时控制:
64 位 Windows 10, RHEL7.6, Ubuntu 16.04/18.04 & CentOS 7.4
- 设计调试:
64 位 Windows 10, RHEL7.8, & CentOS 7.3