

## 论文标题

在数字采样示波器的内存访问仲裁中使用 OCP

## 作者

**Traian TULBURE** ([traian@easic.ro](mailto:traian@easic.ro))

Traian TULBURE 是 Brasov 的 Transilvania 大学的一名讲师兼博士。他毕业于罗马尼亚 Brasov 的 Transilvania 大学的电子和计算机学院，2000 年获得电子和计算机专业的硕士学位。他的毕业论文是“eASIC 技术中的 EDA 工具”。

他的研究方向是 VLSI 设计、HDL 建模、C/C++ 编程以及合成领域。

**Dan NICULA** ([dan@easic.ro](mailto:dan@easic.ro))

Dan NICULA 博士是罗马尼亚 Brasov 的 Transilvania 大学电子和计算机学院的一名教授。他毕业于罗马尼亚的布加勒斯特工艺学院，1990 年获得电子学硕士学位。

1997 年，Transilvania 大学授予他计算机科学专业的博士学位。他的研究兴趣主要在 VLSI 设计和 HDL RTL 级建模领域。教授的课程包括 HDL、微处理器和数字电子产品。他编写了多本与 VHDL/Verilog 有关的书。他还是位于罗马尼亚 Brasov 的 eASIC 分部的主管。

## 概述

以 SoC (片上系统) 的形式设计集成电路的最新方法基于对具有精确功能定义的模块反复使用模型。为了使互连更加容易，这些 IP 核心应拥有符合标准插槽规则的接口。开放核心协议 (OCP) 是知识产权 (IP) 核心接口 (或插槽) 的通用标准，便于实现“即插即用”片上系统 (SoC) 设计。

本论文介绍如何在通过单端口 SRAM 实现多端口访问内存时使用 OCP 协议。研究案例是在 eASIC FlexASIC 设备上实现的数字采样示波器 (DSO)。该芯片的常用功能包括单 USB 供电模块中的双通道数字采样示波器 (DSO) 和任意波形发生器 (AWG)。

该芯片在包含单端口 SRAM 的结构化 eASIC 组织上实现。片上 SRAM 在 DSO 和 AWG 之间共享。CPU 对内存的访问也是必不可少的。设计了时分多路复用内存访问。选择 OCP 协议在带宽为 96 位的数据总线上对 SRAM 访问进行访问和仲裁。

## 简介：如何使用 OCP？

eASIC 的 DSO 芯片 (项目名称为“eScope”) 设计用于 FlexASIC FA600 设备，使用 FlexASIC 设备上可用的所有块 SRAM (bRAM) 内存来存储获取通道和波形发生器通道的采样。因为 bRAM 内存只有一个端口，并且必须通过三个客户端访问内存，所以，实施了时分多路复用方案。

FA600 上所有可用的内存块均配置为 4kx96 内存，以 12 个 bRAM ( 4 行 x 3 列 x 1k x 32 位 ) 为单位构建。内存直接连接到总线转换器 (ocp2mem) 上。总线转换器提供对 bRAM 模块的 OCP 访问。

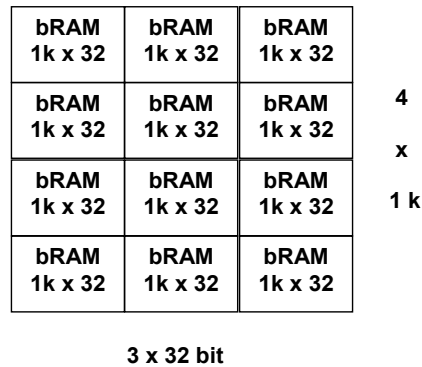


图 1：eScope 内存模块图

内存模块的带宽为 96 位，最多可以存储 8 个数据采样，每个采样 12 位。对于双采样通道模式，数据由来自两个通道的交叉采样组成。

将块内存配置为宽数据总线内存 ( 96 位 ) 并将其时钟设置为高频内部时钟 ( 最高 210 MHz ) ，可以确保 8 个 12\*210 Mbps 的时隙。8 个时隙分配给 3 种操作类型：

- 2 个时隙供触发器发生器逻辑将数据采样存储到内存中；
- 1 个时隙供波形发生器逻辑从内存中检索波形发生采样；
- 6 个时隙可供 USB 主机读/写数据采样。

项目约束是使内存访问技术独立。这意味着使用标准总线提供内存访问，与内存实现方式和接口无关。为了实现内存访问仲裁和标准核心互连，选择了 OCP 解决方案。

这样，在以后的 eScope 版本中，可以轻松地改进对块内存的时分多路复用，支持更大的内存或包含更多 MSPS 的模/数转换器。

### 体系结构：DSO 体系结构和 OCP 位置

eScope 围绕着 9 个模块进行设计，使用开放核心协议 (OCP) 作为将内存与三个可能的客户端进行互连的标准。子模块包括：

- **clkGen** : eScope 的时钟/重置发生器
- **adInInput** : 来自 ADC 的数据采样的同步器
- **dacOutput** : 传到数/模转换器的波形数据的 FIFO 同步器
- **trigGen** : 触发器发生器包含 OCP 发起器端口
- **waveGen** : 波形发生器逻辑包含 OCP 发起器端口
- **hostIf** : USB 模块主机接口包含 OCP 发起器端口

- **sampleMem** : 为存储数据采样和波形发生器数据共享的块内存
- **ocpMerge** : 3 个 OCP 发生器端口和 1 个 OCP 目标端口的 3:1 组合 OCP 合并
- **ocp2mem** : OCP/内存接口转换器包含 OCP 目标端口

子模块互连如图 2 所示：

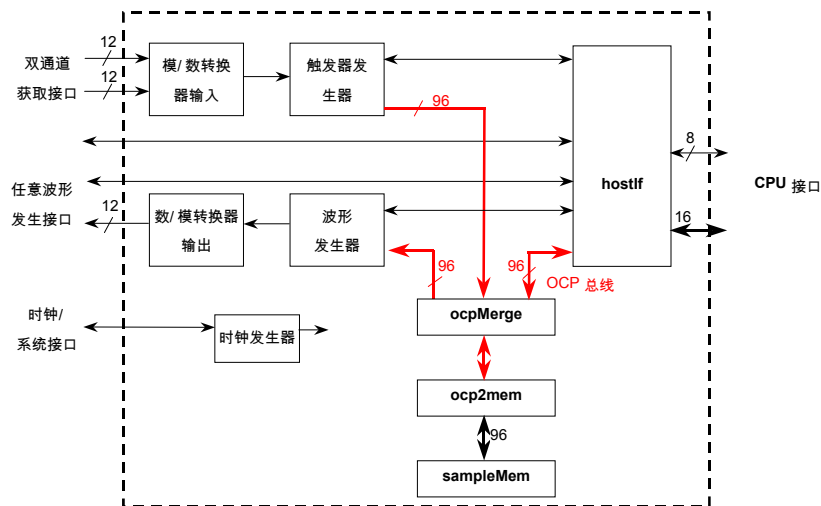


图 2：eScope 内部体系结构。OCP 位置。

ocpMerge 模块实现固定优先级的仲裁，优先级顺序为：

- trigGen (最高优先级)；
- hostIf；
- waveGen (最低优先级)。

trigGen (触发器发生器) 对采样内存发起 OCP 写入事务。

waveGen (波形发生器) 发起 OCP 读取事务。

hostIf (主机接口) 发起读取和写入 OCP 事务。

### 测试：测试 OCP 总线

eScope 已使用不同的方案进行测试，随机选择了包含各种少见案例的参数集中的所有配置参数。图 2 所示的测试环境包含 Matlab-Verilog 模拟。使用 OCP 监视器和检查器测试了 OCP 总线。在顶级测试台上添加了 OCP 监视器，用于检查 OCP 事务。在 RTL 模拟过程中检查所有内部 OCP 96 位宽的总线。在模拟结束后，还将脱机检查并反汇编转储文件。图 2 显示 eScope 测试环境：

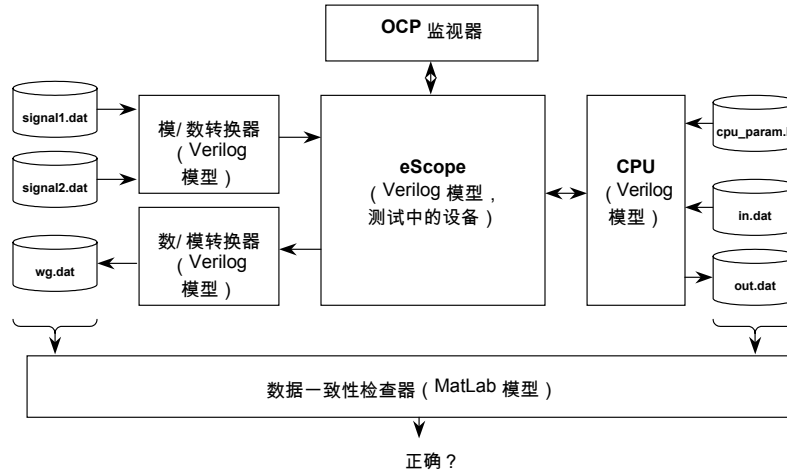


图 2：ModelSim/Matlab/CoreCreator 测试环境

外部 ADC、DAC 和 CPU 模型从外部文件读/写采样数据。使用与 HDL 代码功能相同的 MatLab 脚本完成所选方案的验证，使用与错误相同的一组参数和图形显示。以 MatLab 预期生成的信号与外部 Verilog 模型写入文件的信号之间的差别来评估该错误。零错误表示某个方案成功运行。

各种测试方案通过 eScope 参数定义，这些参数由 CPU 的哑元模型以及为模/数转换器和 HFIF 输入文件选择的波形进行控制。从一组预定义值中随机选择 eScope 参数和测试波形的值。此选择在 Verilog 测试环境中使用自定义版本的 \$random 任务完成。

### 实现：eASIC 技术特定的实现

eTool 测试流程如图 3 所示。此测试流程基于标准单元设计的正常 ASIC 设计流程。通过设计此方法来尽量利用现有的 ASIC 设计方法和基础结构。

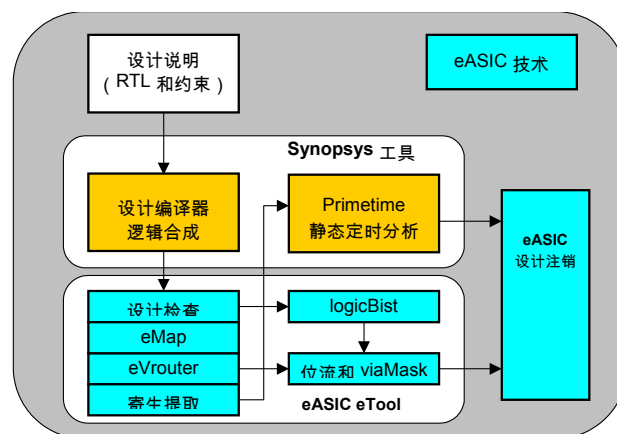


图 3：eTool 设计流程

逻辑合成使用 Synopsys 设计编译器合成工具执行，使用 eASIC 提供的自由格式合成库，该库包含全部两个、三个、四个和五个可以组合使用 eCell 中的 LUT 配置实现的

专有信息

本文档以及其中包含的数据归 eASIC Corp. 独有

输入函数，以及反转缓冲区模型和已在 eCell 逻辑中优化实现的许多 DesignWare 组件。

设计编译器生成的 netlist 通过 eASIC 开发的“技术映射和打包”工具（称为 eMap）传递，该工具尽可能使用速度较快的 nand 和 mux 函数替换速度较慢的、基于 LUT 的函数，然后尽可能将多个函数打包成一个 eCell。eMap 创建的 netlist 在放置和路由之前采用标准的线负载建模，与 Synopsys Primetime 静态定时分析配合使用。

表 4 和表 5 使用 Synopsys 流程在结构化 ASIC 上实现 eScope：

时钟	合成 (ns)	映射 (ns)	放置和路由 (ns)
scClk	3.95	4.72	7.53
wgClk	3	4.43	5.42
hfClk	26.37	28.97	35.62

表 4：eScope 定时结果

合成 (单元)	映射 (单元)	放置和路由 (单元)
5945	3923	3923

表 5：eScope 定域结果

后端设计流程使用 eASIC 开发的 eMap 和 eVrouter 工具来实现最终的放置和路由。执行寄生提取来注释最终的 netlist，以便使用 Primetime 进行后端（注销）静态定时分析。同时，创建 logicBIST 矢量来嵌入最终的位流和 Via Mask 数据库。使用 eDK 工具来执行位流汇编和配置 Via Mask 生成步骤。

## 结论：使用 OCP 的好处

在只使用单端口 SRAM 和时分多路复用为三端口内存建模时使用 OCP 标准总线，可以在短时间内开发 eScope 项目。

通过反复使用早期设计中的 OCP 开关和 ocp2mem 转换器，同时通过 OCP 监视器和检查器充实测试阶段，还使 eScope 的设计更加容易。

## 参考内容

[www.easic.com](http://www.easic.com)