

## 基于 ARCA3 CPU 的嵌入式 SoC 的 FPGA 原型验证

陈达燕,王进祥,孙 俊,杨 奕

(哈尔滨工业大学 微电子中心,黑龙江 哈尔滨 150001)

**摘 要:** 基于 FPGA 的验证是 SoC 功能验证的有效途径,建立一个基于 FPGA 的原型验证系统已成为 SoC 验证的重要方法. ARCA3 是一种高性能、低功耗,国产的嵌入式微处理器. 在 ARCA3 和 AMBA 架构上集成存储器控制器等 IP 核和外设,构建一个嵌入式 SoC,并在 FPGA 上实现 SoC 的原型验证系统和软硬件协同验证环境. 在 FPGA 原型机上运行 Bootloader 和操作系统,验证整个系统硬件的可操作性和软硬件之间的交互. 基于 FPGA 的原型验证系统的实现可以快速验证基于 ARCA3 的各种抽象层次的 IP 核和开发基于 ARCA3 的软件应用.

**关键词:** 嵌入式 SoC; 软硬件协同设计; Linux 操作系统

**中图分类号:** TN911. 22

**文献标识码:** A

**文章编号:** 1000-7180(2011)12-171-04

## FPGA Verification Platform for ARCA3 CPU Based Embedded SoC

CHEN Da-yan, WANG Jin-xiang, SUN Jun, YANG Yi

(Center of Microelectronics, Harbin Institute of Technology, Harbin 150001, China)

**Abstract:** FPGA-based verification platform is an effective verification way of SoC, and it is becoming very important to build a prototype of the SoC on FPGA. ARCA3 is a high-performance and low-power processor. Integrated EMC and IPs to ARCA3 CPU, build a embedded SoC. Implement its prototyping system and Hardware/Software Co-design system on FPGA for SoC verification. The system hardware and software can be fully verified by running bootloader and OS on FPGA prototyping machine. FPGA based verification platform can improve the efficiency of IPs verification and accelerate the development of software for ARCA3.

**Key words:** embedded SoC; hardware/Software co-design; linux OS

### 1 引言

随着微电子技术的飞速发展,单芯片可以集成更多的晶体管,将处理器、存储器、各类数字 IP 和模拟 IP 集成到一个芯片上成为可能,叫做 SoC<sup>[1]</sup>. 由于 SoC 的规模越来越大,功能越来越多,因此对 SoC 设计的验证工作变得越来越难. 一般的验证方法,是采用软件模拟的方式来验证. 这种方法的仿真程度很高,但由于 SoC 电路规模大、内嵌软件复杂(操作系统、驱动程序、通信协议和应用程序等),基于软件模拟的验证变得十分缓慢,而且调试极不方便.

近些年来,FPGA 快速发展,FPGA 的芯片规模

达到数千万门,里面还集成各种存储器、功能 IP 和时钟模块,为大规模的 SoC 的硬件原型验证提供了基础. FPGA 是实际的硬件电路,其验证速度比普通的软件模拟块几个数量级,可以在 FPGA 上运行大型的操作系统和驱动程序等软件,提高验证覆盖率,使得 SoC 产品的上市速度加快. 同时 FPGA 验证也可以减少设计的风险,在 SoC 的 ASIC 芯片实现之前,通过 FPGA 验证已经成为一个必要的过程.

ARCA3 是方舟公司开发的一种高性能、低功耗,具有自主知识产权的嵌入式微处理器. ARCA3 采用 RISC 指令集,具有 8 级流水线. ARCA3 集成了 MMU(存储器管理单元),大大减小了存储器访问延迟. ARCA3 具有 AMBA 协议接口,因此,只要

符合 AMBA 协议的 IP 核都可以和 ARCA3 集成。

本文论述基于 ARCA3 的嵌入式 SoC 平台的构建,在 FPGA 实现了原型验证系统和软硬件协同验证环境. 通过修改 U-Boot 和 Linux 底层硬件抽象和源代码,在 ARCA3 平台上运行 Bootloader 和操作系统,扩展了 SoC 的软件功能,也充分验证了 SoC 的功能(软硬件协同验证)。

2 基于 ARCA3 的 SoC 设计

2.1 SoC 的硬件结构

基于 ARCA3 处理器的 SoC 硬件结构见图 1. SoC 采用 AMBA 总线互连各个 IP 核和模块,包括 ARCA3 处理器,存储器控制器,DDR2 控制器,串口,中断控制器,定时器,以太网 MAC,USB,DMA 等。

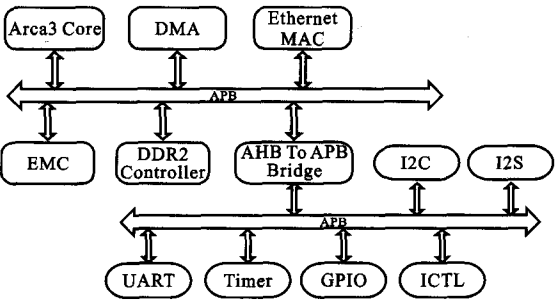


图 1 基于 ARCA3 的嵌入式 SoC 结构

EMC 是外部存储器控制器,片外接 SRAM 和 FLASH,用来装载启动程序和 Bootloader. ICTL 是中断控制器,用来管理 UART,TIMER,GPIO,以太网 MAC 等模块的中断,中断信号连到 ARCA3 处理器上. DDR2 控制器用来控制 FPGA 片外的 DDR2 存储器。

2.2 SoC 的软件仿真

SoC 构建完成后,需要进行系统级 RTL 仿真. 验证各个 IP 模块集成到 ARCA3 和 AMBA 架构后,总线接口信号和时序是否正确,各个模块的功能和数据传输是否正确。

SoC 中 DDR2 控制器是 Xilinx IP 核,没有 AHB 总线接口,需要对其添加 AHB 总线接口. 图 2 是 DDR2 控制器的框图。

经过封装后的 DDR2 控制器直接接入系统进行验证,对 DDR2 存储器进行各种读写操作以及对 DDR2 控制器中寄存器进行读写,完成 DDR2 控制器的验证。

其他 IP 是 Synopsys DesignWare IP,其功能经过了 IC 公司的验证,因此在 RTL 主要验证 IP 与

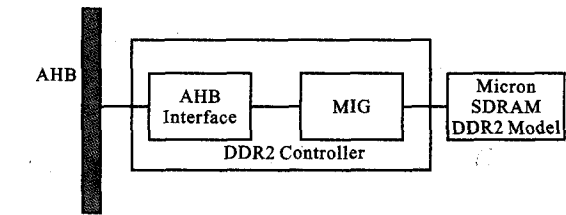


图 2 DDR2 控制器模块框图

ARCA3 之间的接口信号连接是否正确。

3 SoC 的 FPGA 原型验证

3.1 设计的 FPGA 实现

采用 FPGA 实现 SoC 之前,需要将 SoC 的 ASIC 实现转换为 FPGA 实现,具体有以下几个方面:

(1) FPGA 的存储器需要使用 FPGA 厂商提供的 EDA 工具生成,映射到 FPGA 的 Block Memory. ASIC 的存储器需要使用工艺厂商提供的 Memory compiler 工具生成。

(2)其他的单元,比如时钟 PLL 和一些 FPGA 原语调用。

原型验证系统采用 Xilinx 公司的 ML510 FPGA 评估板. ML510 的 FPGA 型号是 XC5VFX130T,拥有 20 480 个 Slice,每个 Slice 包含四个寄存器和四个六输入查找表,拥有10 728kB 的 Block RAM,以及两个 IBM PowerPC 440 处理器等资源,板上还带有以太网接口、NOR FLASH、DDR2 存储器、PCI 总线接口、USB 接口等,这个开发板是标准的 ATX 主板架构. 表 1 所示为整个系统在 FPGA 实现时所利用的硬件资源。

表 1 SoC 在 ML510 实现占用的 FPGA 资源

|                | Used   | Total  | 百分比/% |
|----------------|--------|--------|-------|
| Slice Register | 24 252 | 81 920 | 29    |
| Slice LUTs     | 40 439 | 81 920 | 49    |
| Slice          | 15 620 | 20 480 | 76    |
| Block RAM/FIFO | 141    | 298    | 47    |
| Memory(KB)     | 4 752  | 10 728 | 44    |

3.2 基于 FPGA 的原型验证

将 ISE 生成的硬件比特流下载到 FPGA 中. 编写测试程序,验证 SoC 的功能. 运行测试程序时,可把系统运行到特定时间和步骤的执行结果提取出来并自动比较是否正确,并把结果传递到 PC 主机. 由于 FPGA 是实际的硬件电路,因此 FPGA 功能验证

比系统级 RTL 仿真具有更快的速度、更大的验证范围和更可靠的验证结果<sup>[2]</sup>. 验证过程如下:

(1) 模块级验证:对 DDR2 存储器进行各种读写操作,验证 DDR2 控制器功能是否正确. 对串口, Timer, GPIO, 中断控制器等外设模块的寄存器进行读写访问,验证外设模块功能是否正确.

(2) 系统级验证:编写测试程序,观察 CPU、存储器、外设(串口或 TIMER)和中断控制器 各个模块的数据传输和功能协调. 验证 SoC 的功能和性能.

(3) 应用验证:编写 FFT、MUSIC 算法等应用程序在 SoC 上执行,验证整个 SoC 的功能和性能. 移植 U-boot 和 Linux 到 ARCA3 体系结构,让 SoC 运行 Bootloader 和操作系统,利用软硬件协同来验证 SoC,使 SoC 的原型验证具有更高的可靠性.

#### 4 SoC 的软硬件协同验证

操作系统和驱动程序可以为应用程序的执行提供服务,屏蔽其中的硬件细节,因此在 FPGA 验证时,在 FPGA 上执行操作系统和驱动程序可以为芯片的验证提供更深层次的验证范围,提高验证覆盖率<sup>[3]</sup>.

##### 4.1 mini\_boot 程序设计

FPGA 的软件调试过程,是通过将启动程序初始化到 FPGA 的 Block Memory 中,随硬件比特流一起编译下载,每次系统复位时,执行这段程序.

由于硬件比特流的编译过程需要较长时间,这样对调试软件极其不便,每次软件有错误,就需要重新编译比特流,花费几个小时时间. 因此本设计编写了一个小型的 boot 程序,这段程序在软件仿真时经过验证,通过初始化到 Block Memory 中,系统每次复位时,执行这段程序. mini\_boot 可以重新加载程序,跳转到新的应用程序去执行. 这样软件调试过程,变得简单、快速. 后面的 U-boot 和 Linux 操作系统内核移植都是通过 mini\_boot 来加载调试的.

##### 4.2 Bootloader 移植

Bootloader 代码是芯片复位后进入操作系统之前执行的一段代码,主要是为操作系统提供基本的运行环境,由它最终将操作系统启动起来并将控制权交给操作系统内核.

U-Boot 是 sourceforge 网站上的一个开源项目. 它是由德国的工程师 Wolfgang Denk 从 8XXROM 代码发展而来的,它支持很多处理器,比如 PowerPC、ARM、MIPS 和 x86. U-Boot 支持的操

作系统包括 VxWorks、QNX、RTEMS、NetBSD、Linux 等.

U-Boot 不支持 ARCA3 体系结构,因此需要修改 U-Boot 的源代码移植到 ARCA3 体系结构. 移植的主要工作是将 ARCA3 的底层硬件抽象加入到 U-Boot 里. 关键的 U-Boot 目录和移植过程有:

(1) board:平台相关,存放电路板相关的目录文件. 加入 ARCA3 平台的相关文件和 C 代码.

(2) cpu:平台相关,存放 CPU 相关的目录文件. 加入 ARCA3 的汇编初始化代码和其他硬件的底层代码. 有入口函数所在的文件 Start. S 和 Serial. c 文件.

(3) lib\_arca:平台相关,加入 ARCA3 的 C 主函数. 主要是 Board. c 文件.

(4) Include:通用头文件和开发板配置文件. 建立 ARCA3 的配置文件,有汇编宏定义和外设地址.

(5) 修改顶层目录的 Makefile 和链接脚本以及各子目录的 Makefile.

用 ARCA3 交叉编译器重新编译 U-Boot,生成可执行二进制文件. U-boot 执行分为以下两步:

(1) U-Boot 从 cpu/arca 目录下的 start. S 开始执行,主要是初始化 CPU,关闭中断,关闭 Cache,设置堆栈.

(2) 跳转到 lib\_arca/board. c 的 start\_arcaboot ()函数去,完成系统初始化工作,设置环境变量,初始化环境变量,设置串口波特率,初始化控制台,进入主循环,处理用户输入的命令.

U-Boot 启动后,设置串口为控制台,通过串口与 PC 主机进行交互. 等待用户输入命令. 图 3 所示为 U-Boot 在 FPGA 硬件原型上的运行,调试信息通过串口传递到 PC 主机.

```
U-Boot 1.1.6 (Jan 8 2010-19:01:22)
U-Boot code:90100000->90117390 BSS: ->9012909C
DRAM:64 MB
CPU:ARCA3
* * * Warning-bad CRE, using default environment

In: serial
Out: serial
Err: serial
### main_loop entered; bootdelay=5
### main_loop bootcmd="bootm"
Hit any key to stop autoboot:5
ARCA3#
```

图 3 U-Boot 在 FPGA 硬件原型的执行

### 4.3 Linux 操作系统内核移植

Linux 是一个开源的类 UNIX 操作系统,在嵌入式领域应用的很广泛. Linux 支持许多的嵌入式处理器结构,包括 ARM、PowerPC、MIPS 等. 由于 Linux 开放源代码,因此有很多软件开发工作都是基于 Linux 的.

Linux 不支持 ARCA3 体系结构,因此需要修改 Linux 的底层硬件驱动,将 Linux 移植到 ARCA3 体系结构. 需要修改的地方有 CPU 体系结构和外设的驱动.

用 ARCA3 交叉编译器重新编译 Linux,可以通过 U-Boot 来引导 Linux 操作系统. 图 4 所示为 Linux 在 FPGA 硬件原型的启动过程和 Linux shell 的执行,在 PC 主机通过串口控制台与 shell 交互.

```
Linux NET4.0 for Linux 2.4
Based upon Swansea University Computer Society NET3.039
Starting kswapd
Serial driver version 5.05c (2001-07-08) with no serial options enabled
ttyS00 at 0xe0001000x (irq = 7) is a Arcal6550
ttyS01 at 0xe0001400x (irq = 8) is a Arcal6550
ttyS02 at 0xe0001800x (irq = 9) is a 8250
ttyS03 at 0xe7ff03f8x (irq = 68) is a 8250
ttyS04 at 0xe7ff02f8x (irq = 67) is a 8250
block: 64 slots per queue, batch=16
RAMDISK driver initialized: 16 RAM disks of 8192K size 1024 blocksize
RAMDISK: Compressed image found at block 0
Freeing initrd memory: 384k freed
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 40k freed
Welcome to ARCA3_FPGA Linux

-/bin/sh: /bin/hostname: not found
# ls
bin      home     lost+found  sbin      var
dev      lib      proc       tmp
etc      linuxrc  root       usr
# cd bin
# vi a_
```

图 4 Linux 在 FPGA 硬件原型的运行

## 5 结束语

在 ARCA3 和 AMBA 架构基础上添加 DDR2

控制器、串口、定时器、中断控制器等一些外设和 IP 核,构建一个简单的嵌入式 SoC. 在 Xilinx 公司的 ML510 FPGA 评估板上实现整个系统的软硬件协同验证环境,成功移植 U-Boot 和 Linux 内核到 ARCA3 体系结构,充分利用软硬件协同验证了 SoC 的功能. FPGA 验证其实是一种硬件加速的方法,优点是速度快,短时间内可以跑大量的验证程序<sup>[3]</sup>.

SoC 的 FPGA 原型验证系统实现对基于 ARCA3 的软件开发有重要的意义,程序人员可以在 Linux 操作系统上快速开发基于 ARCA3 的应用程序. 还可以为基于 ARCA3 的各种抽象层次的 IP 验证提供软硬件协同仿真环境,缩短系统设计、IP 集成和验证的时间. 本文的创新之处还在于编写一个 mini\_boot 程序,用于加速 FPGA 的软件调试过程.

### 参考文献:

- [1] 赵刚, 侯立刚, 刘源. 基于 SoC 设计的软硬件协同验证方法学[J]. 微电子学与计算机, 2006, 23(6): 24-26.
- [2] 孔璐, 吴武臣, 侯立刚, 等. 基于 LEON3 的 SOC 平台设计与 SPI 嵌入[J]. 微电子学与计算机, 2010, 27(3): 172-176.
- [3] 虞致国, 魏敬和. 基于 JT AG 的 SoC 软硬件协同验证平台设计[J]. 微电子学与计算机, 2010, 27(10): 160-162.

### 作者简介:

陈达燕 男, (1987-), 硕士研究生. 研究方向为嵌入式系统、数字集成电路.

(上接第 170 页)

- [7] Rahm E. and Bernstein P A. A survey of approaches to automatic schema matching[J]. VLDB Journal, 2001, 10 (4): 334-350.
- [8] 李国徽, 杜小坤, 胡晓, 等. 基于函数依赖的结构匹配方法[J]. 软件学报, 2009, 20(10): 2667-2678.
- [9] David Aumuellner, Do Hong-Hai, Sabine Massmann, et al. Schema and ontology matching with COMA++[C] //Proc SIGMOD, Baltimore, Maryland: IEEE, 2005.

### 作者简介:

刘金岭 男, (1958-), 硕士生导师, 教授. 研究方向为数据库、数据仓库、数据挖掘.

倪晓红 女, (1974-), 硕士, 讲师. 研究方向为网络及数据库应用.

冯万利 男, (1973-), 硕士, 讲师. 研究方向为数据库、数据仓库、数据挖掘.