

# 实现 FPGA 回读功能的可重构系统设计

周盛雨<sup>1,2</sup>, 孙辉先<sup>1</sup>, 陈晓敏<sup>1</sup>, 安军社<sup>1</sup>, 张健<sup>1</sup>

(1. 中国科学院空间科学与应用研究中心, 北京 100080; 2. 中国科学院研究生院, 北京 100039)

**摘要:** Xilinx Virtex 系列 FPGA 具有配置逻辑可重构、配置数据可回读的特点, 该文设计了基于 Virtex FPGA 的一种可重构系统。FPGA 采用 SelectMAP 配置方式, 在 CPU 和 CPLD 控制下实现了配置数据加载和回读的功能。给出了系统配置 FPGA 和回读其配置数据的流程及相应的波形图。

**关键词:** 可重构; FPGA; SelectMAP; CPLD; 回读; Virtex

## Design of Reconfigurable System with FPGA Readback Function

ZHOU Shengyu<sup>1,2</sup>, SUN Huixian<sup>1</sup>, CHEN Xiaomin<sup>1</sup>, AN Junshe<sup>1</sup>, ZHANG Jian<sup>1</sup>

(1. Center for Space Science and Applied Research, Chinese Academy of Sciences, Beijing 100080;  
2. Graduate School, Chinese Academy of Sciences, Beijing 100039)

**【Abstract】** As FPGA is reconfigurable in logic and can be readback in data, this paper designs a reconfigurable system based on Xilinx Virtex™ FPGA. CPU and CPLD are used to configure FPGA which selects a SelectMAP configuration mode and readback configuration data in it. The paper introduces the process of the system to implement the configuration and readback function. Part of the results are shown in wave charts.

**【Key words】** Reconfigurable; FPGA; SelectMAP; CPLD; Readback; Virtex

可重构思想最早可追溯到 20 世纪 60 年代, 美国加利福尼亚大学的Geraid Estrin提出了可重构计算的概念, 研制了原型系统, 奠定了可重构计算系统的基础<sup>[1]</sup>。20 世纪 70 年代末, Suetlana P. Kartashev和Steven I. Kartashev 博士提出了动态可重构系统的概念, 对集成电路动态可重构系统的软、硬件进行了研究。

近年来可重构技术成为国际上计算系统研究中的一个热点。它的出现使过去传统意义上硬件和软件的界限变得模糊, 使硬件系统具有软件化的特征。电路重构的本质是利用可编程 FPGA 具有可多次配置的特性, 根据需要调用不同的配置文件来改变电路逻辑结构, 从而使系统具有灵活、简捷、硬件资源可重用、易于升级等多种优良性能。基于此技术设计的可重构系统在高速数字滤波器、图像压缩、硬件演化计算、定值计算、嵌入式系统等方面, 都有着广泛的应用前景。

可重构系统通常在两种情况下对 FPGA 进行重新配置: FPGA 逻辑功能需要转换, 如算法改进或是发现设计上的错误; FPGA 配置数据发生错误而导致功能失效, 如在空间辐射环境下的单粒子翻转 (SEU) 效应导致配置数据错误。本文介绍了基于 Virtex FPGA 的可重构系统的设计, 此系统能够实现 FPGA 配置数据的回读。

### 1 系统设计

本文设计的可重构系统由 CPU、CPLD、FPGA 和存储器 (FLASH) 构成, 系统设计框图如图 1 所示。设计中 FPGA 选用了 Xilinx 公司 Virtex 系列芯片 (Virtex-E XCV600E), 该系列 FPGA 基于 SRAM 工艺, 具有配置逻辑可重构、配置数据可回读的功能<sup>[2]</sup>。Virtex FPGA 共有 4 种配置方式: 主串 (master serial)、从串 (slave serial)、SelectMAP 和边界扫描 (boundary scan)。FPGA 设计一般采用串行 (主串或从串) 方式, 这种方式

需要的配置信号少 (PROGRAM、CCLK、DIN、INIT、DONE), FPGA 在配置时钟的上升沿接收 1 位配置数据, 采用 SelectMAP 方式, FPGA 在配置时钟的上升沿接收 8 位配置数据, 因此 SelectMAP 配置速度快, 重构时间短, 同时可以通过此接口回读出 FPGA 的配置数据, 综上选用 SelectMAP 配置方式, 它的编程信号如表 1 所示。

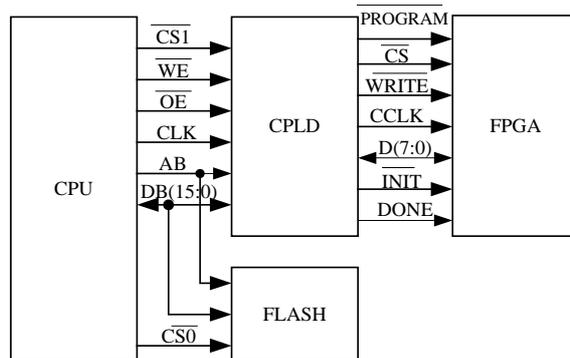


图 1 可重构系统框图

表 1 中 BUSY 信号是握手信号, 只有当配置时钟的频率超过 50MHz 时才起作用, 本设计的配置时钟频率低于 50MHz, 因此不使用 BUSY 信号。M(2:0) 是模式选择信号, 在 SelectMAP 配置模式下, M(2:0) 应置为 0x110。

FLASH 用于存储系统的启动代码及应用程序, 同时还利

**作者简介:** 周盛雨 (1976 - ), 女, 博士生, 主研方向: 基于 FPGA 的嵌入式动态可重构系统; 孙辉先、陈晓敏, 研究员、博导; 安军社、张健, 博士、副研究员

**收稿日期:** 2006-08-04 **E-mail:** zsyrain@sina.com.cn

用 FLASH 的部分存储空间存储 FPGA 配置数据。

表 1 SelectMAP 配置信号

信号名称	信号方向	描述
PROGRAM	I	异步配置复位, 置低时清除片内配置存储器, 初始化配置序列
CS	I	片选
WRITE	I	写使能
CCLK	I/O	配置时钟
D(7:0)	I/O	字节宽度配置数据输入, 回读数据输出
INIT	I/O	准备接收配置数据
DONE	I/O	配置加载完成
BUSY	O	握手信号, 控制配置速率
M(2:0)	I	配置模式选择

CPLD 作为 CPU 控制、监测 FPGA 的配置和回读 FPGA 配置数据的接口电路, 内部有 4 个功能模块: 编程寄存器, 配置寄存器, 监测寄存器和回读寄存器模块。其逻辑框图如图 2 所示。CPU 数据线 DB 在 CPLD 内部寄存器的构成如表 2 所示。

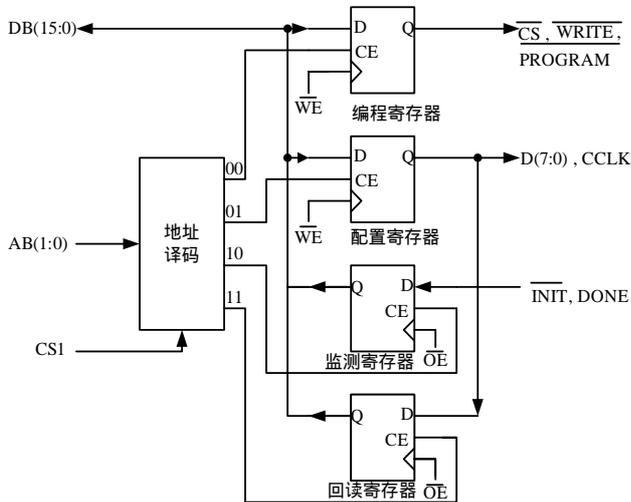


图 2 CPLD 内部逻辑框图

表 2 CPLD 寄存器表

寄存器地址	寄存器名称	寄存器数据								
		DB(15:8)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
00	编程寄存器	NA	NA	NA	NA	NA	NA	CS	WRITE	PROGRAM
01	配置寄存器	DB(7:0)	NA	CCLK						
10	监测寄存器	NA	NA	NA	NA	NA	NA	NA	DONE	INIT
11	回读寄存器	D(7:0)	NA	NA						

CPLD 完成以下 4 方面的功能: 启动 FPGA 配置时序; 向 FPGA 内部传入配置数据; 监测 FPGA 是否正常配置; 回读出 FPGA 配置数据。

CPU 配置 FPGA 的流程如下:

(1) 启动 FPGA 的配置时序: CPU 地址线选中 CPLD 内部编程寄存器, 数据线输出 FPGA 配置控制信号 CS、WRITE 和 PROGRAM, 将 CS 和 WRITE 置低, PROGRAM 信号置低至少 300ns 后置高<sup>[3]</sup>;

(2) 监测 FPGA 的配置状态: CPU 地址线选中 CPLD 内部监测寄存器, 接收 FPGA 的输出信号 INIT, 若其值为高, 那么 CPU 开始发出 FPGA 的配置数据, 否则持续检查 INIT 的值, 直到其值为 1;

(3) 向 FPGA 内传送配置数据: CPU 地址线选中 CPLD 内部配置寄存器, 数据线 DB(15:8) 输出 FPGA 的配置数据, DB(0) 产生配置时钟 CCLK, 在 CCLK 的上升沿 FPGA 接收配置数据。配置数据传送分两步, 首先 DB(0) 置 0, 然后 DB(0) 置 1, CCLK 产生一个上升沿, 在此过程中 DB(15:8) 数据保持不变。重复此过程, CPU 输出 FPGA 全部配置数据;

(4) 检查 FPGA 是否配置成功: FPGA 的配置数据被传送到结束时, CPU 地址线选中 CPLD 内部监测寄存器, 接收 FPGA 的输出信号 DONE, 若其值为高时, 说明 FPGA 配置成功, 否则需要重新配置 FPGA。

每次重构 FPGA 时, CPU 根据需要调用 FLASH 中存储的不同配置文件, 从上述过程的第(1)步开始顺序执行。

本系统利用 Xilinx Virtex 系列 FPGA 具有回读功能的特性, 完成读出其配置数据的功能。图 3 是 SelectMAP 配置方式下 FPGA 的回读时序图<sup>[4]</sup>。回读过程不影响 FPGA 正常工作。

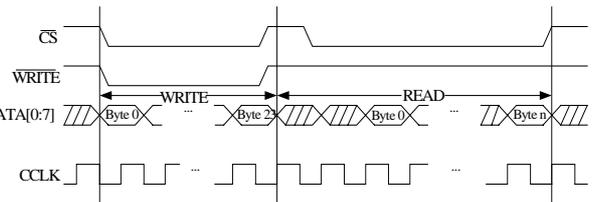


图 3 SelectMAP 配置及回读时序图

CPU 回读 FPGA 的过程如下:

(1) CPU 地址线选中 CPLD 内部编程寄存器, 将 FPGA 配置控制信号 CS 和 WRITE 置低;

(2) CPU 地址线选中 CPLD 内部配置寄存器, 向 FPGA 发送回读命令;

(3) CPU 地址线选中 CPLD 内部编程寄存器, 将 CS 置低, WRITE 置高, FPGA 进入回读时序;

(4) CPU 地址线选中 CPLD 内部配置寄存器, 利用 DB(0) 产生回读时钟 CCLK, 在 CCLK 的上升沿, FPGA 输出配置数据, 存储到 CPLD 内部回读寄存器中;

(5) CPU 地址线选中 CPLD 内部回读寄存器, 读出 FPGA 回读数据。

重复上述(4)、(5), CPU 将 FPGA 的配置数据全部读出。

## 2 仿真和分析

CPLD 选用 Xilinx 公司的 XC95144XL, 内部功能逻辑由 VHDL 语言实现。图 4 是在 Xilinx ISE6.1 软件下 CPLD 控制 FPGA 重构时的仿真波形图。由于 Virtex-E XCV600E FPGA 的配置数据共 3 961 632B, 因此仿真出 FPGA 完整的配置过程是不现实的, 图 4 中用 6B 的配置数据来代表 FPGA 的完整配置, 在 1 775ns 处, FPGA 开始重构。为了在图中看到信号 DB(15:0) 的值, 将其分为 DBH(15:8) 和 DBL(7:0) 两个信号。

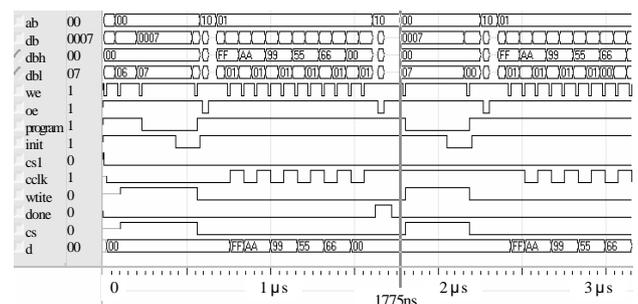


图 4 重构仿真波形图

(下转第 274 页)