

## PKU-DSPII 中 SPI 接口与 Bootrom 程序设计

王 腾, 周芝丽, 陈英瑞, 王新安

(北京大学深圳研究生院 集成微系统科学与工程应用重点实验室, 深圳 518055)

**摘 要:** 针对 PKU-DSPII 芯片, 设计了具有 AMBA AXI 3.0 接口的 SPI 主机接口, 支持四种传输模式、可变传输帧长度、频率可编程时钟输出和五种中断; 同时, 针对该 SPI 接口设计了 RISC DSP 的 bootrom 程序, 包括从 FLASH 正常启动和对 FLASH 系统程序进行升级两种工作模式. 通过功能仿真和 FPGA 原型验证, PKU-DSPII 可以利用该设计的 SPI 接口成功通过外部 FLASH 启动或进行系统升级, 证明了硬件设计和 bootrom 程序的正确性. 该设计已提交华润上华 0.18 $\mu\text{m}$  CMOS 工艺流片.

**关键词:** DSP; SPI 接口; AMBA AXI 协议; Bootrom 程序; FLASH

**中图分类号:** TP47

**文献标识码:** A

**文章编号:** 1000-7180(2012)12-0080-04

## Design of SPI Interface and Bootrom Program in PKU-DSPII

WANG Teng, ZHOU Zhi-li, CHEN Ying-rui, WANG Xin-an

(Key Lab of Integrated Micro-Systems Science Engineering and Application,  
Peking University Shenzhen Graduate School, Shenzhen 518055, China)

**Abstract:** An SPI master interface with AMBA AXI 3.0 interface is designed for PKU-DSPII single chip, which supports four transfer modes, variable frame length, frequency programmable clock output and five interrupt types. Simultaneously, the bootrom program for the RISC DSP is also proposed, providing normal booting from external FLASH and system updating of the boot-loader codes in the external FLASH. Both the function simulation and FPGA prototyping show that PKU-DSPII can boot and conduct system updating through the external FLASH successfully, which provides good evidence for the correctness of the design of the SPI master interface and the bootrom program. The design has been implemented with CSMC 0.18 $\mu\text{m}$  CMOS technology for taping out.

**Key words:** DSP; SPI interface; AMBA AXI protocol; bootrom program; FLASH

### 1 引言

图 1 是北京大学深圳研究生院集成微系统重点实验室设计的 PKU-DSPII SOC 芯片的系统结构图. 从图中可以看出, 该 SOC 芯片由自主设计开发的 16 位 RISC DSP 和相关接口模块组成, 采用 AMBA 总线结构. 其中 RISC、DDR2 控制器、USB 控制器、DMA 控制器、以太网 MAC 和 SPI0 由高速总线 AXI 相连, 并通过 AXI-APB 桥与 APB 总线上的相关系统控制模块和低速外设接口进行通信.

PKU-DSPII 中的 RISC 内包含 2KB 的片内掩

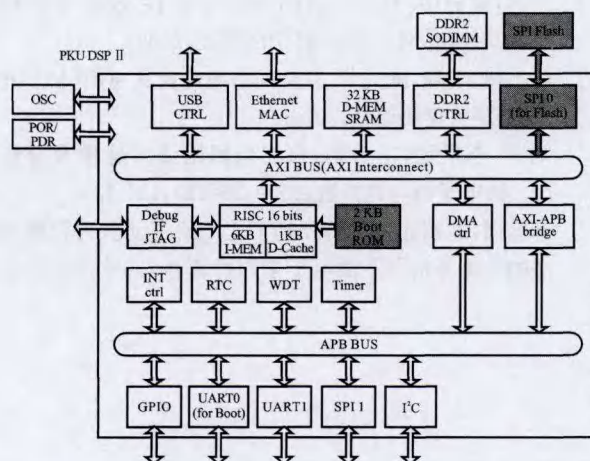


图 1 PKU-DSPII SOC 芯片系统结构图

收稿日期: 2012-04-11; 修回日期: 2012-05-12

基金项目: 深圳市科技计划基础研究资金支持(JC201005270278A)

膜 ROM,存储 RISC 的启动引导 bootrom 程序,该程序通过检测 GPIO 的状态,可引导 RISC 通过 SPI0 从片外 Flash 正常启动,也可以引导 RISC 通过调试串口 UART0 和 SPI0 升级外部 Flash 内的系统程序.

2 SPI 接口硬件设计

串行外围设备接口 (Serial Peripheral Interface, SPI) 是 Motorola 公司提出的一种高速、全双工、同步通信总线,目前广泛应用于各种 SOC 芯片中进行外设扩展和数据交换<sup>[1-2]</sup>. SPI 总线通常由四根线组成:串行时钟线(SCK),主机输出从机输入线(MOSI),主机输入从机输出线(MISO)和从机选择线(SSN).

高级可扩展接口协议(Advanced eXtensible Interface, AXI)是 ARM 公司提出的基于 AMBA(Advanced Microcontroller Bus Architecture)结构、面向高频高性能系统设计的系统总线结构,包含五个独立通道,支持非对齐和乱序数据操作,同时还支持低功耗操作的可选扩展<sup>[3]</sup>.

为满足 PKU-DSPII 芯片与外挂 FLASH 的高速数据传递,本文设计了带 AXI 接口的 SPI 主机模块 AXI-SPI. 其硬件结构如图 2 所示.

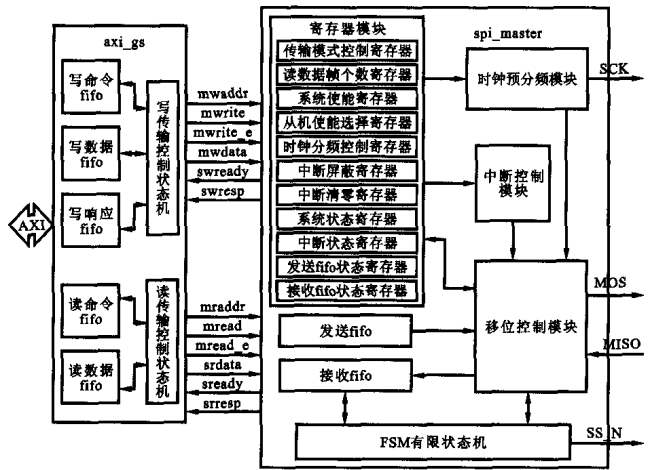


图 2 AXI-SPI 主机硬件结构图

AXI-SPI 主要由 axi\_gs 和 spi\_master 两部分组成,其中 axi\_gs 完成由 AXI 协议五个通道信号到通用读写数据信号的转换<sup>[4]</sup>. 由于 AXI 每一个突发传输由一个首地址和多个数据传输组成,同时 AXI 总线的数据位宽为 64 位而 spi\_master 读写数据位宽为 16 位,设计中采用状态机控制的命令 fifo 和数据 fifo 对 AXI 的命令和数据进行缓存并转换为通用接口的读写信号供 spi\_master 使用. 为进一步提高数据交换的效率,在通用接口上同样采用读传输

通道和写传输通道分离的设计.

spi\_master 模块由寄存器模块、时钟分频模块、中断控制模块、移位控制模块以及接收 fifo 和发送 fifo 组成. 其中,寄存器模块完成各控制寄存器和状态寄存器的更新,本设计中定义的主要控制寄存器和状态寄存器如表 1 所示.

表 1 spi\_master 寄存器列表

| 名称   | 地址   | 描述                 |
|------|------|--------------------|
| CTRL | 0x00 | 控制寄存器              |
| NDF  | 0x02 | 读数据帧个数寄存器          |
| ENR  | 0x04 | 使能寄存器              |
| BAUD | 0x08 | 波特率选择寄存器           |
| SR   | 0x0e | 状态寄存器              |
| IMR  | 0x10 | 中断屏蔽寄存器            |
| ISR  | 0x12 | 中断状态寄存器            |
| ICR  | 0x1c | 中断清零寄存器,读动作清除中断    |
| DR   | 0x20 | 数据寄存器,指向发送和接收 fifo |

其中控制寄存器 CTRL 分别控制自测模式、传输模式、时钟极性、时钟相位和数据帧长度. 这里共支持四种传输模式,包括读写模式、只写模式、只读模式和 EEPROM 读模式,而数据帧长度可支持 4—16 位. 所有的控制寄存器(包括 NDF、BAUD、IMR 等)只能在 spi 使能即 ENR 为 1 时进行修改,以确保传输的稳定性.

时钟分频模块根据控制寄存器中的时钟极性和相位以及波特率选择寄存器所设定的时钟分频值,产生用来驱动 MOSI 和对 MISO 进行采样的串行时钟信号 SCK,输出片外给从机使用,同时控制移位控制模块.

移位控制模块首先从发送 fifo 中获取发送数据放入移位寄存器中,并将从机选择 SS\_N 信号拉低;然后在串行时钟信号的控制下在一个时钟沿(由时钟极性和相位共同决定)将发送数据串行地移出并驱动到 MOSI 上,并在另一个时钟沿对 MISO 进行采样并移入移位寄存器;当一个数据帧传输结束后,将移位寄存器中的接收输入存入接收 fifo,同时从发送 fifo 中获取下一笔发送数据.

图 3 展示了在串行时钟相位为 1 时,输入输出的格式. 可以看出,在相位为 1 极性为 0 时,上升沿发送下降沿采样,而极性为 1 时相反.

在读写模式和只写模式,当发送 fifo 中最后一笔数据发送完成后传输结束;而在只读模式和 EEPROM 读模式下,当接收到 NDF+1 笔数据后传输结束. 传输结束后将 SS\_N 信号拉高.

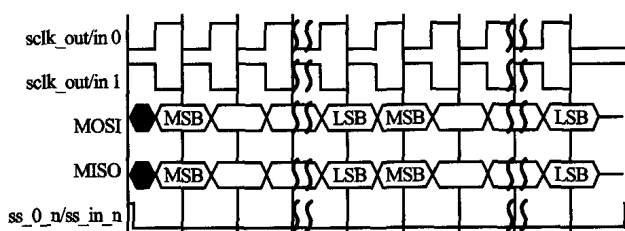


图 3 相位为 1 时 SPI 传输格式

状态寄存器 SR 的 bit4-0 分别标识接收 fifo 满、接收 fifo 不空、发送 fifo 空、发送 fifo 不满和 spi 忙状态。

spi\_master 共支持五种中断响应,分别由中断状态寄存器 ISR 的低五位标识,包括发送 fifo 空 (txe)、发送 fifo 溢出 (txo)、接收 fifo 满 (rxif)、接收 fifo 溢出 (rxo) 和接收 fifo 过载 (rxu)。这些中断可以由中断屏蔽寄存器 IMR 控制选择打开或者屏蔽,并通过 RISC 对中断清零寄存器 ICR 的读动作进行清除。

### 3 Bootrom 程序设计

RISC MCU 按其存储其类型可分为无片内 ROM 型和有片内 ROM 型,前者必须外接 EPROM 才能应用,后者又分为片内掩膜 ROM 型和片内 FLASH 型<sup>[5]</sup>。片内 FLASH 型虽然灵活性较高,但由于成本较高只在少数大型设计中使用;一般情况下 MCU 都采用片内 ROM 加片外扩展 FLASH 的结构,这样可以兼顾系统的灵活性和成本。

PKU-DSPII 即采用片内 ROM+片外 FLASH 这种结构,这样就需要设计片内 ROM 中的启动引导程序 bootrom。本文设计的 bootrom 程序分为三种模式:DEBUG 模式、NORMAL 模式和 UPDATE 模式。RISC 在 DEBUG 模式下可通过 UART0 进行人工调试,在 NORMAL 模式下通过本设计的 SPI 从 FLASH 启动,在 UPDATE 模式下通过 SPI 对 FLASH 中的系统程序和数据进行更新。

系统初始化之后,通过检测 GPIO 最高两位的值,RISC 分别进入 DEBUG、NORMAL 或 UPDATE 模式。

对于 NORMAL 模式,其启动流程为:

- ①配置 SPI 为 EEPROM 读模式,设定需要读取的指令长度;打开 SPI 使能;
- ②发送读命令和指令存储的首地址;
- ③检测 SPI 状态寄存器,当  $SR[3]=1$  即接收 fifo 不空时,说明接收到指令;
- ④RISC 从 SPI 接收 fifo 加载指令至 IMEM,指

令加载完成后关闭 SPI 使能;

⑤重复上述流程加载数据;

⑥当指令和数据加载完成后,跳转至 IMEM 入口,RISC 正常启动。

对于 UPDATE 模式,为了更新 FLASH 内的指令和数据,首先通过 UART0 将更新的指令和数据放入 DMEM,然后(1)对 FLASH 进行复位;(2)打开 FLASH 写使能位;(3)对 FLASH 进行区域擦除(Sector Erase);(4)再次打开 FLASH 写使能位;(5)从 DMEM 中读取指令和数据对 FLASH 进行页面编程(Page Program)。

以打开 FLASH 写使能位为例,其流程为:

- ①配置 SPI 为只发送模式;打开 SPI 使能;
- ②发送 FLASH 写使能命令(0x06);
- ③检测 SPI 状态寄存器,当  $SR[0]=1'b0$ ,即 SPI 忙结束表明发送完成;
- ④关闭 SPI 使能;配置 SPI 为只发送模式;
- ⑤打开 SPI 使能;发送 FLASH 读状态寄存器命令(0x05);
- ⑥检测 SPI 状态寄存器,当  $SR[3]=1$  即接收 fifo 不空时,说明接收到 FLASH 状态寄存器值;
- ⑦RISC 读取 FLASH 状态寄存器的值并判断,若 FLASH 使能位 WEL 为低,则回到④,重复上述流程;若 FLASH 使能位 WEL 为高,则关闭 SPI 使能位并结束。

当 UPDATE 模式完成后,跳转至 NORMAL 模式正常启动即可。

### 4 FPGA 原型验证

本文设计的 AXI-SPI 用 Verilog HDL 设计实现,bootrom 程序基于 PKU-DSPII 的专用指令集用汇编代码实现并转化为二进制码,两者最终集成于系统中进行功能仿真和 FPGA 原型验证。

为对系统进行功能仿真,系统中采用 SPAN- SION 公司的型号为 S25FL008K 的 FLASH 模型<sup>[6]</sup>。图 4 展示了在 UPDATE 和 NORMAL 模式下发送给和接收自 FLASH 的第一笔数据。

从图中可以看出,UPDATE 模式下,通过 AXI 总线给 0xb020(发送 fifo 的地址)写入数据 0x04de(对应汇编指令为 BR 0x4de,即跳转至应用程序入口),在 MOSI 上即串行地输出 0x04de 至 FLASH; NORMAL 模式下,FLASH 通过 MISO 串行返回 0x04de,通过 AXI 总线读取相应 0xb020(接收 fifo 的地址)位置时,rdata 数据线的低 16 位上获得 0x04de。仿真结果证明 AXI-SPI 主机接口工作正

确, bootrom 程序功能正确。

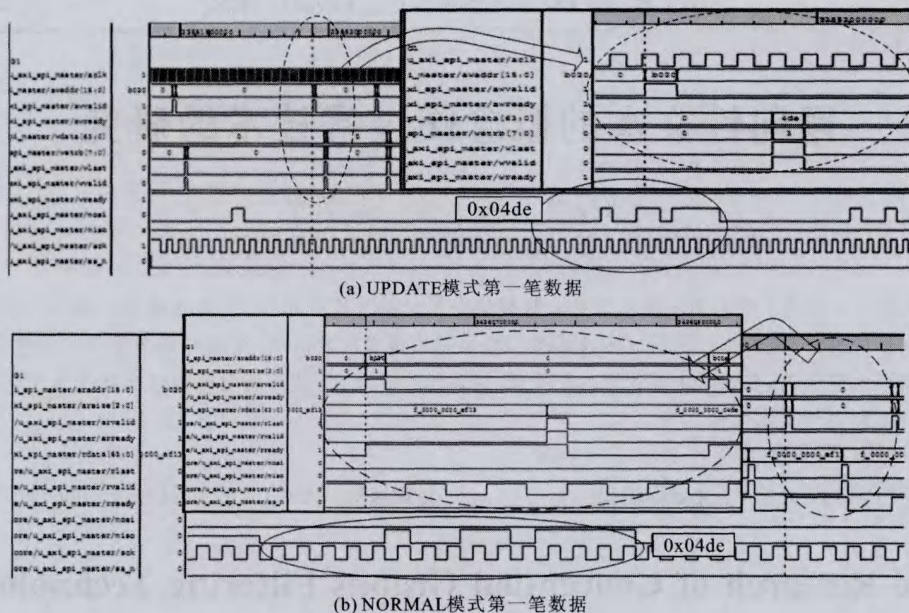


图4 功能仿真波形

进一步的FPGA原型验证采用Xilinx XUPV5-LX110T FPGA开发板, 外挂FLASH芯片采用WINBOND公司的W25Q32. 系统工作频率为50 MHz, SPI串行输出时钟频率为12.5 MHz, 系统共占用Virtex5-LX110T FPGA中44%逻辑资源和45%的块存储(Block Memory)。

图5展示了NORMAL模式下, 系统通过UART串口的打印信息, 可以看出, 系统启动正常, 应用程序执行正确。

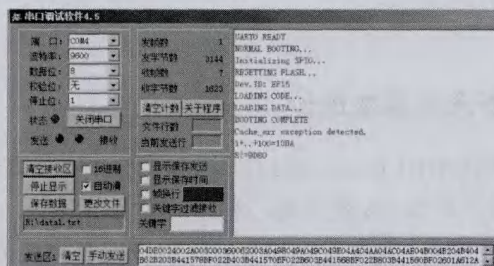


图5 NORMAL模式启动信息输出

## 5 结束语

本文针对PKU-4DSPII芯片设计了具有AXI接口的SPI主机模块, 同时设计了利用该SPI接口从外部FLASH进行系统启动和升级的bootrom程序. 通过功能仿真和FPGA原型验证, 证明了硬件设计和bootrom程序设计的正确性. 设计已集成在PKU-DSPII芯片提交上华0.18  $\mu\text{m}$  CMOS工艺流片。

## 参考文献:

- [1] 杨美刚, 李小文. SPI接口及其在数据交换中的应用[J]. 通信技术, 2007, 40(11): 385-387.
- [2] 夏继强, 周丽萍, 满庆丰. 基于FPGA的串行总线扩展平台[J]. 微电子学与计算机, 2004, 21(9): 127-130.
- [3] ARM, AMBA AXI Protocol Specification[S]. 2010.
- [4] 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): 173-176.
- [5] Patterson D A, Hennessy J L. 计算机组成与设计: 硬件/软件接口[M]. 郑纬民, 译. 北京: 机械工业出版社, 2007.
- [6] Spansion. S25FL004K/S25FL008K/S25FL016K Data Sheet[S]. 2011.

## 作者简介:



王 腾 男, (1988—), 博士研究生. 研究方向为系统集成芯片设计与设计方法学。

周芝丽 女, (1989—), 硕士研究生. 研究方向为系统集成芯片验证。

陈英瑞 男, (1989—), 硕士研究生. 研究方向为系统集成芯片设计与设计方法学。

王新安 男, (1963—), 博士, 教授, 博士生导师. 研究方向为系统集成芯片设计与设计方法学、阵列处理器架构与设计。