

基于 Xilinx MicroBlaze 多核嵌入式系统的设计

何宾, 王瑜

(北京化工大学 信息科学与技术学院, 北京 100029)

摘要: MicroBlaze 核是嵌入在 Xilinx FPGA 之中的属于 32 位 RISC Harvard 架构软处理器核。针对 Xilinx MicroBlaze 软处理器的核间互连, 实现多处理器核之间的快速通信的目的, 采用了 PLB 和 FSL 总线混连的方法, 利用 xps_mailbox 和 xps_mutex 核完成核间的通信与同步, 通过在 Xilinx EDK 平台下, 将 3 个软处理器核嵌入到 FPGA Spartan-3E 芯片上的试验, 开发出了一个运行在 FPGA 上的基于多处理器的嵌入式可编程片上系统, 得出此种多核处理器混连的可行性与实用性, 核间通信速度得到提升的结论。

关键词: Xilinx; 多核处理器; MicroBlaze; 嵌入式; 核间通信

中图分类号: TP368.1

文献标识码: A

文章编号: 1674-6236(2011)13-0141-04

Design of dual-core embedded system based on Xilinx MicroBlaze

HE Bin, WANG Yu

(School of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China)

Abstract: MicroBlaze core, which is embedded in the Xilinx FPGA, is among the 32-bit RISC Harvard architecture soft processor core. In this paper, for the Xilinx MicroBlaze soft processor interconnect and in order to realize the faster communication between the processors, with the method that PLB bus is mixed with the FSL bus interconnect, using the xps_mailbox, xps_mutex to complete the inter-core communication and synchronization, under the Xilinx EDK platform, by embedding three soft-core processor into the FPGA Spartan-3E chip, a embedded programmable system on a chip based on multi-core processor running on the FPGA is developed. It concluded that this kind of multi-core interconnect is practicality and feasibility, it improves the speed of inter-core communication.

Key words: Xilinx; multi-core processor; MicroBlaze; embedded; inter-core communication

随着嵌入式处理需求的快速增长, 单处理器系统复杂度太高且计算能力不足的缺点不能满足需求, 嵌入式系统架构开始转向多处理器的协同工作, 这样针对同一系统的多任务需求, 协同工作的处理器可以充分完成各自管辖的不同功能应用。近年来, FPGA 以其高逻辑集成度和内嵌的高性能硬件模块的优势, 使得功能强大的芯片多处理 (CMP) 解决方案成为现实, 如今关键问题是如何高效地将各种 IP 核互联在一起, 构成复杂的系统, 实现多核处理器的设计。

本文提出了将 3 个可编程的 32 位微处理器软核 MicroBlaze 内嵌到 FPGA 芯片上, 实现了一个运行在 FPGA 上的基于多处理器软核的可编程片上系统, 并对采用 PLB 总线与 FSL 总线连接处理器, 实现核间通信进行了比较。一个 FPGA 芯片内部可以使用主从两个或多个 MicroBlaze 处理器, 而 FPGA 最大的特点就是能够缩短产品开发的周期, 它的可编程配置特性可以将系统开发的风险降到最低。而 Xilinx 公司的 MicroBlaze 软核处理器的优点在于, 不像 ARM 是由固定的硬芯片来实现, 用户可以根据实际系统的需求添加必要的外设来实现基于 MicroBlaze 的嵌入式可编程片上

系统, 从而提高了设计的灵活性。

1 系统总体结构设计

系统整体架构如图 1 所示。本系统是基于 MicroBlaze 的三核系统, 采用嵌入 3 个 32 位处理器软核 MicroBlaze 的 SOPC 是由 Xilinx 公司提供的 Spartan-3E 芯片^[1]。

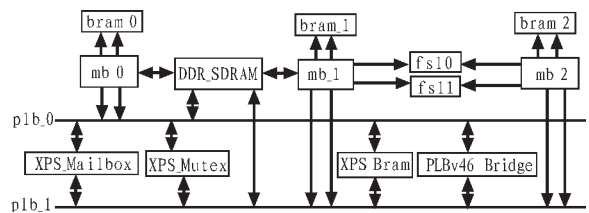


图 1 三核处理器互连架构

Fig. 1 Interconnect architecture of three processor

在本系统中, 2 个 MicroBlaze 软核协同处理工作, 核间采用 PLBv46 总线互连, 并利用系统自带的 XPS_Mailbox 核实现核间的通信, 第 2 个核与第 3 个核之间采用 FSL 总线实现点对点的通信。在核不多的情况下, 采用共享内存结构可以有效地减少核间的通信量, 避免数据一致性的问题。本系统

收稿日期: 2011-04-18

稿件编号: 201104082

作者简介: 何宾 (1975—), 男, 河南济源人, 博士, 讲师。研究方向: 嵌入式系统、高速实时信号处理。

中,数据传送到第1个核,第1个核与第2个核之间进行通信,第2个核再将数据发送给第3个核,并通过PC机超级终端接收UART的输出,得到输出信息,经过分析和比较,可以对系统进行调试和最后结果的验证。

1.1 Spartan-3E 开发板

Spartan-3E 是 Xilinx 公司在 Spartan-3 成功的基础上改进的产品,提供了比 Spartan-3 更多的 I/O 端口和更低的单位成本,是 Xilinx 公司性价比最高的 FPGA 芯片。其应用广泛:支持 32 位 RISC 处理器;支持 DDR 接口的应用;内嵌 Xilinx 的 MicroBlaze 软核,可用于嵌入式系统的开发;支持基于 Ethernet 网络的应用。本系统采用的 FPGA 芯片是 XC3S500E-4FG320。

2 多 MicroBlaze 软处理器核的系统设计

2.1 EDK 开发平台介绍及其开发流程

一般而言,由于嵌入式系统涉及了软件和硬件的开发以及两者的综合设计,因此其开发是较为复杂的。Xilinx 为了简化基于 FPGA 的嵌入式开发流程,提供了功能强大、操作简单的工具 ISE 和 EDK。

EDK (Embedded Development Kit) 是 XILINX 公司针对 FPGA 内部 32 位嵌入式处理器开发而推出的工具套件。EDK 的工具包中集成了硬件平台产生器 (platgen)、软件平台产生器 (libgen)、仿真模型生成器 (simgen)、软件编译器 (GNU) 和软件调试 (GDB) 等工具,同时配有丰富的可重用 IP 核,利用其集成开发环境 XPS (Xilinx Platform Studio) 可以方便、快速地完成嵌入式系统开发的整个流程。该开发流程主要分为硬件开发流程和软件开发流程,整个流程如图 2 所示。其中,嵌入式处理器硬件系统的构建由微处理器硬件规范 (MHS) 文件定义,用于描述硬件系统结构,定义处理器类型、总线接口、外设接口、中断处理和地址空间;软件系统结构由微处理器软件规范 (MSS) 文件定义,用于装载软件库、驱动程序和文件系统。Data2BRAM 的作用是把软件代码文件 (*.Elf)、FPGA 位流文件 (*.bit) 和块存储器 (BRAM) 初始化数据文件 (*.bmm) 转换成新的 FPGA 位流文件 (*.bit) 和存储器数据文件 (*.mem)^[2]。

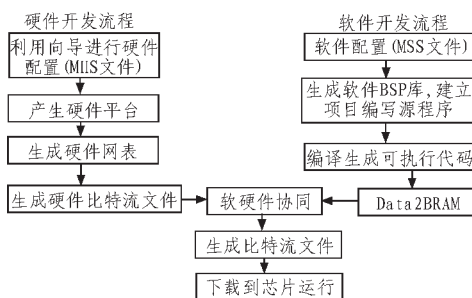


图 2 EDK 开发流程

Fig. 2 Development process of EDK

2.2 系统硬件平台的搭建与设计

2.2.1 MicroBlaze 软核

MicroBlaze 处理器是 Xilinx 公司针对嵌入式处理器开发

应用推出的一种 32 位嵌入式处理器内核,采用 RISC 架构和哈佛结构的 32 位指令和数据总线,可以全速执行存储在片上存储器和外部存储器中的程序,并访问其中的数据。简单但灵活性强,与其他外设 IP 核及用户 IP 核一起,可以完成片上可编程系统芯片 (SOPC) 的设计。MicroBlaze 处理器的内部结构如图 3 所示^[3]。

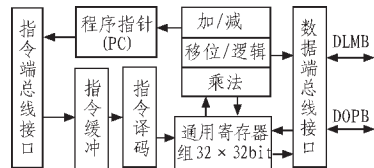


图 3 MicroBlaze 内部结构框图

Fig. 3 Internal block diagram of MicroBlaze

2.2.2 硬件平台搭建步骤及其设置

利用 Xilinx 公司提供的嵌入式开发工具 EDK 中的基本系统创建器 (Base System Builder, BSB) 向导来快速创建一个工程。通过进入 XPS 图形用户界面,在系统的组件面板中,通过 Bus interface 子窗口建立硬件单元与总线的连接,在硬件单元上单击鼠标右键,可配置其参数、阅读相应的数据手册以及查看底层代码。通过 Ports 子窗口配置端口参数,包括顶层模块和各个子模块的端口,并可对其重命名。单击相应信号行 Net 列的下拉框,可选择连接的网表名。通过 Address 子窗口设置硬件单元的绝对地址和大小,用户还可设置指令缓存 (ICache) 和数据缓存 (DCache) 的位置。

2.2.3 PLBv46 总线

通过 Xilinx 公司 128bit 宽度的 PLB 总线可将多个 PLB 主设备和 PLB 从设备连接到整个 PLB 系统中。PLB 总线结构包括总线控制单元、看门狗计时器、读/写数据通路以及独立的地址空间,此外,还带有一个可选的 DCR 从接口来访问总线错误状态寄存器。PLB 总线的主要特点包括以下几点:

- 1) 支持任意数目的 PLB 主设备。
- 2) 可支持 128 bit、64 bit 及 32 bit 位宽的从设备和主设备。
- 3) 仲裁时间小于 3 个周期。
- 4) PLB 地址流水线技术,支持共享总线模式或端到端的配置模式。
- 5) 自带 PLB 看门狗计时器。
- 6) 效率远高于 OPB 总线,特别是在大吞吐量应用场合。

此外,PLB 总线支持单拍读写 8 位、16 位、32 位数据传输和低延时的 PLB 点到点拓扑结构。PLB 总线桥的功能就是将 2 个独立的 PLB 总线连接起来,实现外设资源的共享^[4]。

2.2.4 FSL 总线

FSL_V20 快速单链接 (FSL) 总线是一个基于 FIFO 的单向点对点通信总线,可用来连接 FPGA 上的任意两个带有 FSL 总线接口的设计元素并提供两者间的快速通信信道。Xilinx MicroBlaze 处理器最多可支持 8 个 FSL 主设备接口和 8 个 FSL 从设备接口。这些接口被用于处理器的寄存器堆和 FPGA 片内的硬件之间的数据传输,单个数据传输操作消耗的 CPU 时间不超过 2 个时钟周期^[4]。

FSL 总线具备以下主要特征:

- 1)实现了单向的、点对点的、基于 FIFO 的通信;
- 2)提供了不共享的、无需仲裁的通信机制,可以用于在带有 FSL 接口的主设备和从设备之间快速传送数据;
- 3)提供了额外的控制位用于注释被传输数据,这个控制位对于从设备有多种用途,例如,将被传输数据作为一个控制字解码,或者用这个位来标识一个数据帧的起始或结束;
- 4)支持的 FIFO 深度最小为 1,最大为 8 k;
- 5)支持同步和异步 2 种 FIFO 模式,这允许 FSL 的主从端能以不同的速率工作;
- 6)支持 2 种 FIFO 实现方式:双端口 Block RAM 或 LUT RAM, SRL16 移位寄存器。

本系统处理器间通信采取的方案为:处理器 1 与处理器 2 之间通过 XPS_mailbox 核实现,处理器 2 与处理器 3 之间的双向通信通过一对 FSL 总线实现,一个核负责发送,另一个核负责接收,不存在阻塞而导致无法运行的问题。处理器核间的 FSL 总线相当于一个简单的基于 FSL 的数据通道 FIFO_LINK。它用 VHDL 语言描述的实现过程如下:

```
architecture example of fifo_link is
begin
FSL_M_Data<=FSL_S_Data;
FSL_M_Write<=FSL_S_Exists and (not FSL_M_Full);
FSL_S_Read<=FSL_S_Exists and (not FSL_M_Full);
End architecture example;
```

2.3 系统软件的开发与设计

2.3.1 内存地址的映射

在多核处理器的设计中,任何一个共享内存必须进行地址映射。本系统在 FPGA 内部的 BRAM 中放置了共享代码,其中,软件应用程序接口与系统的设备通过平台工作室(XPS)设备驱动,首先在将要运行的应用程序名处右击,选中“Mark to Initialize BRAMs”,然后再右击应用程序名,点击“Generate Linker Script”,对链接脚本文件进行合理的设置。本系统链接脚本文件各个段配置如表 1 所示,实验表明,该配置能够满足系统实际运行要求及代码的高效执行^[5]。

表 1 程序各段配置空间

Tab.1 Configuration space of programming sections

段名	配置位置	说明
.text	DDR_SDRAM_C_MP MC_BASEADDR	存放可执行代码
.rodata	ilmb_cntlr_dlmb_cntlr	存放只读数据
.data	ilmb_cntlr_dlmb_cntlr	存放读写数据
.sdata2	ilmb_cntlr_dlmb_cntlr	存放访问频率高的读写数据
.bss	DDR_SDRAM_C_MP MC_BASEADDR	存放堆栈段可读写数据
.vectors.reset	ilmb_cntlr_dlmb_cntlr	复位段地址空间
.vectors.interrupt	ilmb_cntlr_dlmb_cntlr	中断段地址空间

2.3.2 核间通信

1) 共享内存的通信 多个处理核之间的通信可以使用

进程间通信机制、共享内存或中断。共享内存是最基本的实现不同核上的线程间通信的机制(队列和消息)。在内存中有一个共享的全局变量或数据结构,则在处理器中的软件可以轻易地更新该变量的值,且对于其他处理器是可见的。所有这一切需要的是变量的地址或指向共享区域的指针。

在两个处理器处理共享资源的时候,为了对处理器的工作进行同步,避免出现共同资源占用的冲突,造成功能实现混乱,采用了 XPS_Mutex 核。对于共享资源,如一些常见的外设和对内存的共享访问,经常需要进行同步。XPS Mutex 模式提供了一些可配置的内存映射互斥寄存器,它包括一个权值部分和一个处理器编号的组成部分。为了获得 Mutex,处理器写入了一个软件分配的处理器 ID,该 ID 与互斥寄存器和权值 0 相对应。Mutex 判断两个同时访问 Mutex 的处理器,并在互斥寄存器中存储获胜处理器写下来的权值。如果 Mutex 已经被锁住,Mutex 中的权值不会改变。每个处理器用读回 Mutex 的权值与自己的处理器 ID 进行比较,以验证该处理器有没有获得 Mutex。

当 Mutex 成功挂载到系统中,程序开始会出现“#include<xmutex.h>”的头文件。使用“XMutex mutex”,mutex 表明是 mb_0 和 mb_1 之间的一个互斥,利用该语句进行对 Mutex 核的引用。对于 Mutex 核的初始化,使用语句如下:

```
“cfg = XMutex_LookupConfig (XPAR_XPS_MUTEX_0_
DEVICE_ID);
XMutex_LookupConfig(XPAR_XPS_MUTEX_0_DEVICE_I
D);”
```

利用“XMutex_Lock_CPU (XMutex *InstancePtr, u8 MutexNumber,u8 CPUID)”函数对处理器加锁。利用“XMutex_Unlock_CPU(XMutex *InstancePtr, u8 MutexNumber, u8 CPUID)”对处理器进行解锁^[6]。

2)消息的传递原理 利用 XPS_Mailbox 核来传递两个处理器通信时需要的信息和握手信号。XPS Mailbox 硬件最初是在硬件设计部分描述的,它的软件驱动程序在处理器之间可以帮助提供消息传递功能。该软件的 API 是面向 read()和 write()调用的,因此软件可以将邮箱看作是发送和接收的一个串行访问的文件。该软件库提供阻塞和非阻塞的 API 版本。Mailbox 是从一个或多个发送者传输信息到一个接收者的方案。它形成一种渠道,通过这个渠道信息从发送者端排列进入一个 FIFO 流,然后到达接收端,它可以被看作成两核之间一个简单的 TCP/IP 形式的信息通道。

Xilinx 利用 XPS Mailbox 核实现核间的通信,该核具有以下几个特点:

①每一个 Mailbox 核都有一对 Mailbox FIFO,一个用来发送数据,另一个用来从特定的处理器接收数据。

②FIFO 的深度由用户定义。

③FIFO 利用的是分配的 RAM 或是 BRAM 资源。

④每一个 Mailbox 都有一对连接处理器用来通信的接口。虽然,多处理器都可以连接到接口上,但是推荐的方法是每一个 Mailbox 连接 2 个处理器^[6]。

Xilinx 的 XPS Mailbox 核适于中小规模的数据传输(数据

一般小于 100 Bytes)。发送端处理器需要从本地或外部内存复制整条信息,再将其写入 FIFO;接收端处理器将信息复制到内存之中。在拷贝的过程中处理器浪费了一定的时间。

当信息到达 Mailbox 时,Mailbox 接收端发送一个中断;当 Mailbox 不能再接收信息的时候,这个中断是无效的,中断控制器上可以使中断无效,且可以选择开或者关闭,所以,发送端与接收端的通信可以是同步的,也可以是异步的。

当 Mailbox 配置成功时会出现“#include 和 #include”。前者是 Mailbox 的函数库,后者是中断的函数库。在程序调用 Mailbox 时需要用语句“XMbox mbox;”对其进行引用。对于 Mailbox 的初始化,使用语句如下:

```
cfg = XMbox_LookupConfig (MBOX_INTR_ID);
```

```
XMbox_CfgInitialize (&mbox, cfg, cfg->BaseAddress);”
```

对于异步消息传递功能允许一个处理器上的软件去前行,而不必浪费在一个旋转环数据到达的邮箱,从而隔离了从具有其他时间临界任务执行的快速接收机到缓慢发送者。异步消息传递功能也可以用作处理器间中断的一种形式。

3 多核系统的调试

在系统工程中更新比特流,将比特流下载到目标 FPGA 芯片中,经综合、编译后,运行应用程序,进行调试。通过串口连接到目标板上应用系统中的 XMD 调试接口,在超级终端窗口查看观察结果。超级终端的配置为:每秒位数(B)=9600,数据位(D)=8,奇偶校验(P)=无,停止位(S)=1,数据流控制(F)=无。

核间共享内存的调试结果如图 4 所示。

```

-- Entering main() --
Starting MemoryTest for DDR_SDRAM:
Running 32-bit test...PASSED!
Running 16-bit test...PASSED!
Running 8-bit test...PASSED!
-- Exiting main() --

```

图 4 DDR_SDRAM 存储器测试结果

Fig. 4 Testing result of DDR_SDRAM memorizer

核间消息传递的调试结果如图 5、图 6 所示。

```

-- Entering main() --
Running MailboxExample() for XPS_MAILBOX_0
CPU 0: Starts.
CPU 0: Write call
CPU 0: Successfully sent the message.
-- Exiting main() --

```

图 5 处理器发送数据显示结果

Fig. 5 Display result of Processor sending datas

```

-- Entering main() --
Running FifoExample() for FSL_0...
CPU 2: Starts.
CPU 2: Read call
CPU 2: Successfully receive the message.
-- Exiting main() --

```

图 6 处理器接收数据显示结果

Fig. 6 Display result of processor receiving datas

由调试结果知,系统可以正常工作,满足设计需求。

4 结束语

通过实验调试结果得知,基于 Xilinx MicroBlaze 多核系统的嵌入式系统开发^[7],提高了系统的性能,由于 MicroBlaze 软核的可移植性,增加了系统设计的灵活性。文中将多个 MicroBlaze 软核嵌入到 FPGA 芯片中,得知多核比单核的性能有了显著地提高,且验证了 PLB 总线和 FSL 总线混连处理器的可行性,提高了核间的通信速度,降低了系统的开发成本,且表明该系统的设计具有一定的实用性。

参考文献:

- [1] 李剑阳,付宇卓,刘婷.基于 MicroBlaze 多核系统的设计及在 FPGA 上的实现[J].微电子学与计算机,2009,26(7):160-163.
LI Jian-yang,FU Yu-zhuo,LIU Ting.Design multi-core system using MicroBlaze and realize on the FPGA[J].Microelectronics & Computer,2009,26(7):160-163.
- [2] 陈锐,徐建华,蒋亚东.MicroBlaze 嵌入式处理器的硬件加速模块设计[J].微处理机,2009(5):3-6.
CHEN Rui,XU Jian-hua,JIANG Ya-dong.Design of hardware accelerator in MicroBlaze embedded processor [J]. Microprocessors,2009(5):3-6.
- [3] 王玉峰,胥保华.基于 FSL 总线的 UART 外设 IP 核设计[J].单片机与嵌入式系统应用,2008(2):45-47.
WANG Yu-feng, XU Bao-hua. Design of UART peripheral IP core based on FSL[J].Microcontrollers & Embedded Systems, 2008(2):45-47.
- [4] 田耘,徐文波.Xilinx FPGA 开发实用教程[M].北京:清华大学出版社,2008.
- [5] 王尔申,张淑芳,张芝贤.基于 Xilinx MicroBlaze 的嵌入式 GPS 接收机研究[J].通信技术,2010,43(10):28-30.
WANG Er-shen, ZHANG Shu-fang, ZHANG Zhi-xian.Study on embedded GPS receiver based on Xilinx MicroBlaze [J]. Communication Technology,2010,43(10):28-30.
- [6] 张益嘉.一种基于 FPGA 的 MPSoC 架构的设计方法与实现[D].大连:大连理工大学,2009.
- [7] 蔡勉,田健生.向多核平台移植操作系统的研究[J].现代电子技术,2010(05):182-184.
CAI Mian,TIAN Jian-sheng.Research on transplanting of operating system to multi-core environment [J].Modern Electronics Technique,2010(05):182-184.