

学校编码: 10384

学号: 19820061151796

分类号_密级_

UDC_____

厦 门 大 学

硕 士 学 位 论 文

USB2.0 设备控制器在 PowerPC 架构上的
集成设计与验证

Design and Verification of the USB2.0 Device Controller
integrated in PowerPC Architecture

杨 娟

指导教师姓名: 李晓潮 副教授

郭东辉 教授

专 业 名 称: 凝聚态

论文提交日期: 2009 年 月

论文答辩时间: 2009 年 月

学位授予日期: 2009 年 月

答辩委员会主席: _____

评 阅 人: _____

2009 年 月

厦门大学学位论文原创性声明

兹提交的学位论文，是本人在导师指导下独立完成的研究成果。本人在论文写作中参考的其他个人或集体的研究成果，均在文中以明确方式标明。本人依法享有和承担由此论文而产生的权利和责任。

声明人（签名）：

2009 年 月 日

摘要

随着 USB 技术的广泛应用，市场对兼容 USB 协议的设备控制器芯片的需求大量增加。由于电子系统设备接口的数据传输速度不断提升，原有的 USB1.1 协议（12Mb/s）已经逐渐被高速的 USB2.0 传输协议所替代，接口数据传输速度可提高到 480Mb/s，因此，USB2.0 成为了目前电子系统设备接口的标准配置之一。本论文工作旨在设计一种基于 FPGA 内部 PowerPC 架构的 USB 设备控制器，即设计实现了将处理器、USB 设备接口 IP 核和其他外围模块等集成在一个 SoC（片上系统）芯片内。本文首先介绍了 USB2.0 设备控制器开发的相关知识和 FPGA 平台的构建；然后设计和验证了 USB2.0 设备控制器；最后将 USB2.0 设备控制器集成到 PowerPC 架构中对其进行 FPGA 验证。其中，在 FPGA 验证过程阶段，本文对 USB 传输协议和 PowerPC 架构进行了研究，设计实现了 PowerPC 处理器外设总线 OPB 与 USB IP 核接口 Wishbone 总线之间的桥接；并将该桥接应用于 USB 设备控制器的设计中，实现了 OPB 接口的 USB 设备控制器；此外，还对 PowerPC 处理器通过该桥接与 USB 设备接口 IP 核的通信进行验证。本论文工作有实际参考价值的内容主要体现在：

（1）完成了 USB IP 核的时序约束和综合，以保证外部 USB 设备控制器与外部收发器同行时序的正确性。

（2）根据 OPB 总线和 Wishbone 总线协议，设计了 OPB to Wishbone 桥接。

（3）设计和制作了外部物理收发器电路。

总之，随着 SoC 芯片的集成度越来越高，电子系统设备的运行频率与接口数据传输越来越快，且产品面市时间要求越来越短，因此，本文设计实现 USB 设备接口 IP 核及其 SoC 芯片集成，可为电子系统设备配备 USB2.0 标准接口的产品原型开发，提供一种有实际参考价值的 SOPC（片上可编程系统）技术。

关键词：USB 设备接口 IP 核；SOPC；PowerPC 处理器；OPB；EDK

ABSTRACT

With the extensive application of USB technology, there is a significant increase in demand for the device controller chip which is compatible with USB protocol. As the upgrade of data transmission speed of electronic system device interface, the original USB1.1 protocol (12Mb/s) has gradually been replaced by high-speed USB2.0 transfer protocol, and the interface data transfer is speeded up to 480Mb/s. Therefore, USB2.0 has become one of the standard configurations for the current electronic system device interface.

This paper aims to design a USB device controller based on PowerPC architecture inside FPGA, which is to design and implement the project how to integrate processor, USB Device IP core and other peripheral modules into a SoC (system-on-chip) chip. First, this paper introduces the relevant knowledge of a USB2.0 device controller and the FPGA Platform; and then design and verify the correctness of USB2.0 device controller; Finally, in order to integrate USB2.0 device controller into the PowerPC architecture to implement its FPGA verification, we design a OPB to Wishbone bridge according to the transfer protocol of OPB and Wishbone bus. Then this bridge is applied to the design of USB device controller, which realize the USB device controller with OPB interface. And at last we verify that PowerPC processor can communicate with USB device controller core through the bridge. In this paper, the practical reference work is mainly in:

- (1) Complete the timing constraint and synthesis of USB IP core, which is to ensure the correct transfer timing between the external transceiver and USB device controller.
- (2) Design OPB to Wishbone Bridge according to Wishbone bus and OPB bus protocol.
- (3) Design and produce the external physical transceiver circuit

Generally, with much higher degree of integration of the SoC chip, operating frequency and data transmission of electronic systems equipment are much faster, and it takes less time for products to come to market. Therefore, the USB Device Interface IP core and its SoC chip integration designed in this paper can provide a SOPC (Programmable System on Chip) technology with practical reference for electronic

system equipment prototype development with standard USB2.0 interface.

Key Words: USB Device Interface IP Core;SOPC;PowerPC Processor;OPB;EDK

厦门大学博硕士学位论文摘要库

目 录

第 1 章 绪论	1
1.1 USB 发展概况.....	1
1.2 研究背景和意义	2
1.3 USB 设备控制器实现方案.....	3
1.4 关键技术和研究现状	4
1.4.1 IP 核验证技术	4
1.4.2 总线桥接技术.....	5
1.5 主要研究内容	7
第 2 章 USB2.0 开发的相关知识	8
2.1 USB 系统.....	8
2.1.1 USB 总线布局.....	9
2.1.2 USB 拓扑结构.....	10
2.1.3 USB 通信流.....	12
2.2 USB 事务处理.....	14
2.2.1 字段的格式.....	15
2.2.2 信息包的格式.....	19
2.3 USB 数据传输.....	22
2.3.1 传输类型.....	22
2.3.2 事务传输方式.....	23
2.4 FPGA 平台构建.....	26
2.4.1 Xilinx EDK 平台构建工具	27
2.4.2 Xilinx Virtex-II Pro 平台组成.....	29
第 3 章 USB2.0 设备控制器设计与验证	36
3.1 设备控制器的架构	36
3.1.1 模块划分.....	36

3.1.2	顶层引脚定义.....	38
3.2	UTMI 接口模块.....	42
3.2.1	UTMI 接口设计	42
3.2.2	总线事件检测实现.....	44
3.2.3	UTMI 接口模块仿真	47
3.3	协议处理模块	48
3.3.1	解包模块.....	49
3.3.2	打包模块.....	52
3.3.3	协议引擎模块.....	53
3.3.4	协议引擎模块仿真.....	57
3.4	寄存器管理模块	58
3.4.1	控制寄存器组.....	59
3.4.2	端点寄存器组.....	61
3.4.3	寄存器管理模块仿真.....	64
3.5	Wishbone 接口模块	65
3.5.1	Wishbone 接口模块设计	65
3.5.2	Wishbone 接口模块仿真	66
3.6	存储器接口和仲裁模块	68
3.6.1	存储器接口和仲裁模块设计.....	68
3.6.2	存储器接口和仲裁模块仿真.....	69
3.7	USB 设备接口 IP 核 TOP 层功能仿真	71
第 4 章	USB2.0 设备控制器的 FPGA 验证	73
4.1	FPGA 验证系统的组成.....	73
4.2	USB2.0 设备控制器的 FPGA 实现.....	74
4.2.1	USB IP 核的综合	74
4.2.2	USB 设备控制器的实现.....	79
4.3	OPB to Wishbone 桥接实现.....	85
4.3.1	Wishbone 总线	85
4.3.2	OPB IPIF 服务.....	86
4.3.3	OPB to Wishbone 桥接设计.....	87

4.3.4 OPB to Wishbone 桥接的验证结果.....	92
4.4 UTMI 物理收发器.....	95
4.5 USB 验证实物图.....	100
第 5 章 总结与展望	101
5.1 工作总结	101
5.2 工作展望	102
参考文献.....	错误！未定义书签。

厦门大学博硕士论文摘要

CONTENTS

1	Introduction.....	1
1.1	The Overview USB development.....	1
1.2	The Background and Meaning of Research	2
1.3	Implementation Schemec for USB Device Controller	3
1.4	Key Technologies and Progress.....	4
1.4.1	IP Core Verification Technology.....	4
1.4.2	Bus Conversion Technology	5
1.5	Main Research Works	7
2	Related Knowledge of USB2.0 Device Controller.....	8
2.1	USB System.....	8
2.1.1	USB Bus Layout	9
2.1.2	USB Topology Structure.....	10
2.1.3	USB Data Flow Model.....	12
2.2	USB Transaction	14
2.2.1	Packet Field Formats.....	15
2.2.2	Packet Formats.....	19
2.3	USB Data Transfer	22
2.3.1	Transfer Types.....	22
2.3.2	Transaction Types	23
2.4	FPGA Platform Construction	26
2.4.1	Xilinx EDK Platform Construction Tool	27
2.4.2	Xilinx Virtex-II Pro Platform Composition	29
3	The Design and Verification of USB2.0 Device Controller	36
3.1	The Structure of Device controller	36
3.1.1	Module division	36
3.1.2	The definition of Top-level Pin.....	38
3.2	UTMI Interface Module	42
3.2.1	UTMI Interface Design.....	42
3.2.2	Bus Incident Detection.....	44

3.2.3	Simulation of UTMI Interface Module.....	47
3.3	Protocol Layer Module.....	48
3.3.1	Packet Disassembly	49
3.3.2	Packet Assembly	52
3.3.3	Protocol Engine.....	53
3.3.4	Simulation of Protocol Layer Module	57
3.4	Register Management Module.....	58
3.4.1	Control Register Group.....	59
3.4.2	Endpoint register group.....	61
3.4.3	Simulation of Register Management Module	64
3.5	Wishbone InterfaceModule.....	65
3.5.1	Design of Wishbone Interface.....	65
3.5.2	Simulation of Wishbone Interface	66
3.6	Memory Interface and Arbiter	68
3.6.1	Design of Memory Interface and Arbiter.....	68
3.6.2	Simulation of Memory Interface and Arbiter.....	69
3.7	Simulation of The TOP Layer of USB Device IP Functional Core.....	71
第 4 章	The FPGA Verification of USB2.0 Device Controller	73
4.1	Composition of FPGA Validation System.....	73
4.2	FPGA Implementation of USB2.0 Device Controller	74
4.2.1	Synthesis of USB IP Core	74
4.2.2	Implementation of USB Device Controller	79
4.3	Implementation of OPB_to _Wishbone Bridge.....	85
4.3.1	Wishbone Bus	85
4.3.2	OPB IPIF Service.....	86
4.3.3	Design of OPB_to_Wishbone Bridge	87
4.3.4	The Verification of OPB _to_Wishbone Bridge	92
4.4	UTMI Physical transceiver	95
4.5	The Photo of USB Verification System.....	100
第 5 章	Summay and Future Work	101
5.1	Summary.....	101
5.2	Future Work	102
References	错误！未定义书签。

厦门大学博硕士学位论文摘要库

第 1 章 绪论

1.1 USB 发展概况

USB (Universal Serial Bus 通用串行总线) 规范是由以 Intel 为首的七家公司 Compaq、Intel、Microsoft, IBM、DEC、Northern Telecom 以及日本 NEC 于 1994 年 11 月共同提出的, 但直到 1997 年 Microsoft 公司推出 Win95/97 之后, USB 才开始进入实用阶段[1], 而且这个版本对 USB 的支持属于外挂式模块。1998 年 Compaq、Intel、Microsoft、Nec 四个公司联合发布 USB 协议 1.1 版本[2], 规定了两种速度 (1.5Mb/s 和高速 12Mb/s) 以满足不同需要, 同时支持热插拔和即插即用; 有同步、中断、控制和批传输四种方式; 连接线最长距离可达 5 米; 采用星型拓扑结构, 理论上可以同时连接 127 个设备; 各种外设分享 USB 带宽; 同时, 其它外设和主机正在通信时, 总线也允许添加、设置、使用以及拆除外设。同年, Microsoft 推出了 WIN98 系统, 系统内置了支持 USB 接口的驱动模块[2]。正是在 Intel 和 Microsoft 的大力支持下, USB 设备开始了突飞猛进的发展。使用 USB 接口的设备在以惊人的速度发展, 据统计数据显示, 1998 年的一年内, 全球有 1000 万台 USB 设备售出, 到 2000 年已超过一亿台[3]。

随着电子系统的不断发展, 传统 USB1.1 协议 (12Mb/s) 已经不能满足高速数据传输的要求, 因此在 2000 年 Compaq、Hewlett-Packard、Intel、Lucent、Microsoft、NEC、Philips 推出了新的 USB 2.0 版本[4], 它增加了最快为 480 Mb/s 的高速传输模式, 并且可以向下兼容 USB 1.1, 即使用同样的电缆、同样的连接器、同样的连接方式就能到达 480 Mb/s 的接口速度, 使其成为 PC 机外围设备扩展中应用日益广泛的接口标准之一, 从海量存储类, 如移动存储, 移动硬盘, CD/DVD, 读卡器存储设备等; 音频类, 如音箱等音频设备; 监视器类, 如显示器类; 通信类, 如 Modem; 有线连接类, 如 USB 网卡、DSL 路由器、共享器等; 蓝牙及无线类, 如蓝牙转换器、无线网卡等到影音图像类, 如 MP3、USB 音像等; 还有 PDA, modem、电话等, USB2.0 几乎应用于所有种类的外设。据 DigiTimes 科技网的评论预测 USB2.0 产品的出货量从 2003 年的 4 亿 9000 万套成长至 2008

年的 11 亿 3000 万套，年复合增长率为 18.1%。

1.2 研究背景和意义

USB2.0 接口设备的广泛使用催生了一个巨大的需求市场，即对兼容 USB 协议的设备控制器芯片的大量需求。在这一领域，国外的公司如 Intel、Cypress、Philips 等已经赶在了前面，他们推出的各类芯片已经得到了广泛的使用，Cypress 公司 2000 年推出的世界上第一块 USB2.0 功能设备控制器芯片 CY7C68013[5]，以及后续推出的 ISD-300A1[6]、CY7C68000[7]，CY7C683000[8]和 EZ-USB[9] 等高速设备芯片；Philips 公司推出的高速设备控制器芯片 ISP1581[10]；NetChip 公司的高速设备控制器芯片 Net2270[11]、Net2280[12]。但是国内的 IC 设计、生产厂商没有投入足够的精力在这一领域，丧失了很大的市场空间。尤其是兼容协议 USB 的接口，几乎完全依赖于国外的产品，根据公开的文献资料，目前国内还没有 USB2.0 设备控制器芯片开发成功的报道，市场上可以看到的国内设计公司的产品只有极少数的几款 USB1.1 设备控制器芯片，包括南京沁恒出品的 CH372[13]、CH375[14]和 CH374[15]控制芯片。

目前国内关于 USB 的研究主要集中在 USB 系统集成和 USB 芯片设计两个方面。其中，USB 系统集成是指利用现有的接口芯片，编写固件程序和驱动来完成整个应用系统的设计，这方面的研究已经比较成熟。如文献[16] [17] [18]介绍了采用 Philips 的 PDIUSBD12 USB 接口芯片，进行了 USB 通信模块设计以及固件编程；文献[19][20][21]介绍了采用 Cypress 的 CY7C68013 作为 USB 接口芯片的 USB 接口设计以及驱动程序开发；在 USB 芯片设计方面，目前国内的的研究还只处于研发阶段，虽然有一些单位或研究机构在探索独立设计 USB 芯片和 IP 核，但是还没有实际的产品推向市场[22][23][24][25][26]。

在 USB 设备控制器的设计方面，随着片上系统（SoC）芯片[27]集成度越来越高，运行速度越来越快，面市时间越来越短，且随着其依赖的 USB IP 核的大量推出，如 OpenCores 上的免费 USB IP 资源[28]、CPLD/FPGA 业者 Xilinx 提供的范例 DHL 程序[29]、Altera 的 MegaCore IP[30]、Synopsys, Faraday 及台湾的虹晶科技，松瀚科技等推出的商业版 USB IP 核[31]，SoC 已经成为主流的 USB 设计方式。

SoC 将系统的全部功能模块集成到单一半导体芯片上,包括 CPU、I/O 接口、存储器以及一些重要的模拟集成电路。最近发展起来的 SOPC (片上可编程系统) [27] 技术为 SOC 的实现提供了有效的解决方案,即用大规模可编程器件 FPGA 来实现 SOC 的功能,此系统不仅具有 Soc (片上系统) 的特点,即由单个芯片完成整个系统的主要逻辑功能,同时,它还具有可编程,设计灵活,可重复修改和软硬件在线编程的优点。

本文在 Xilinx Virtex-II Pro[32] 开发平台上,采用 SOPC 开发技术,设计实现了一个 USB2.0 设备控制器,该控制器内部通过采用自主设计的总线桥实现 PowerPC OPB 总线[33]与 USB 设备接口 IP 核的 Wishbone 总线[34]之间的桥接,将 PowerPC 处理器和 USB 设备接口 IP 核集成到一片 FPGA 芯片内。这种方案具有较大的灵活性,可以随意地对其功能进行裁剪和扩展,同时由于采用软硬件协同开发技术,有利于挖掘系统潜能、缩小产品的体积、降低系统成本、提高系统整体性能。

1.3 USB 设备控制器实现方案

USB 设备控制器硬件结构一般包括微控制器和 USB 设备接口,硬件电路是软件开发的基础,在 USB 系统的开发中起着决定性的作用,直接决定了 USB 系统的功能、性能。目前,USB 设备控制器的硬件实现主要有以下三种方案:

1) 采用 SoC 或 SOPC 器件,在一个芯片内完成 USB 设备控制器的所有功能。文献[24][35][36][37]介绍了基于 FPGA 的 USB 设备接口 IP 核的实现,在一片 SOC 芯片内集成 USB 设备接口 IP 核、存储器和其它功能模块。这种方案具有较大的灵活性,既可以最大限度地发挥 USB 总线的性能,又可以随意地对其功能进行裁剪,达到器件的最高使用效率。

2) 使用嵌入 MCU (微控制器) 的 USB 控制器,如 Cypress 公司的 EZ-USB, Microchip 公司的 PIC16C745[38], STMicroelectronics 公司的 ST7277[39]等。如文献[40]采用 USB2.0 控制器 EZ-USB-FX2 作为 USB 设备控制器,详细讨论了设备端固件及主机应用程序的实现方案;文献[41]介绍了 EZ-USB-FX2 芯片的内部 FIFOs 和通用可编程接口(GPIF),并提出了其在固件程序中的具体应用和固件程序的实现方案;文献[42]这介绍了采用 EZ-USB SXTM High-Speed US 设备为

数控电源的方案，实现了 FPGA-DSP 高速总线通信。这种微控制器 MCU 内嵌 USB 设备接口引 IP 核的方案是在单芯片内同时集成了 USB 接口 IP 核和基于该核的单片机，虽然可以节省芯片资源，但制作专用集成电路的周期长，不利于产品的升级。

3) 使用专用 USB 接口芯片，如 Lucent 的 USS-820FD[43]; NetChip 的 NET2890[44]; Sipex 的 SP5301[45]; Philips 的 PDIUSBD12 [46]，它是目前 USB 外设设计中常用的芯片，符合 USB1.1 协议，支持高速 12Mb/s，仅处理通信中的 USB 标准协议包，需要外接 MCU（微控制器）来构成 USB 设备控制器，传统的 MCU 外接的方法大多是由单片机来完成微控制器控制作用的，文献[47] [48]介绍了采用 USB 接口芯片 PDIUS-BD12 外接单片机来实现 USB 外设的方案。这种方案虽然开发时间比较短，但由于单片机本身存在的资源匮乏的局限性，开发时可发挥空间比较小，集成度低。由于需要专用的微控制器，而且 USB 设备控制器作为一个“黑匣子”调试起来也比较麻烦。

本文采用第一种方案实现 USB 设备控制器，设计方式如下，借助 FPGA 供应商提供的集成开发环境 EDK (Embedded Development Kit)[49]，通过可视化界面向导灵活选择处理器核以及外围 IP 核，快速方便地搭建所需要的硬件平台，在开源 USB 设备接口 IP 核的基础上，根据需求进行 USB 硬件系统的集成。这种 USB 设备控制器实现方案，即不需要设计或采用专用的单片，又能提高设计的集成度，还能兼顾开发周期和开发成本。

1.4 关键技术和研究现状

1.4.1 IP 核验证技术

随着 HDL IP 软核的出现使电路设计能力有了质的飞跃，集成电路的规模进入到千万门级，验证已经成为 IP 核设计的关键技术。验证所需要的时间、人力、物力甚至超过了芯片本身所耗费的精力，70%以上的芯片失败的原因是验证的不充分[50]，因此验证技术，特别是功能验证技术成为 IC 设计中一个非常重要的环节。功能验证即逻辑仿真，是指针对硬件描述语言描述电路进行的功能验证，是不考虑器件延时和布线延时的情况下对源代码进行逻辑功能的验证，属于前仿真。IP 软核验证就是采用功能验证技术来检验和证明所设计的软核功能是正确的。

Degree papers are in the "[Xiamen University Electronic Theses and Dissertations Database](#)". Full texts are available in the following ways:

1. If your library is a CALIS member libraries, please log on <http://etd.calis.edu.cn/> and submit requests online, or consult the interlibrary loan department in your library.
2. For users of non-CALIS member libraries, please mail to etd@xmu.edu.cn for delivery details.

厦门大学博硕士论文摘要库