

内嵌 8051 的 USB 2.0 设备控制器 IP 设计*

陈亮,袁志坚,史大龙,黄鲁

(中国科学技术大学 电子科学与技术系 集成电路实验室,安徽 合肥 230027)

摘要: 基于 USB 2.0 协议规范提出了一个 USB 2.0 设备控制器串行接口引擎 SIE 的 IP 核的设计,并内嵌 8051 软核作为其微控制器进行 SoC 设计。所设计的 SIE 核在 FPGA 开发板上经过验证。

关键词: 串行接口引擎;通用串行总线;片上系统;微控制器

中图分类号: TN492

文献标识码: A

文章编号: 1674-7720(2012)17-0028-03

IP core design of USB 2.0 device controller with 8051

Chen Liang, Yuan Zhijian, Shi Dalong, Huang Lu

(IC Lab, Department of Electronic Science & Technology, University of Science and Technology of China, Hefei 230027, China)

Abstract: This paper presents the IP core design of a serial interface engine (SIE) of USB 2.0 device controller, which meets the USB 2.0 specification. It uses the 8051 IP as the MCU for SoC design. The SIE is accomplished and verified on FPGA board.

Key words: SIE; USB; SoC; MCU

凭借着每年超过 20 亿新单元的安装速度,通用串行总线 USB(Universal Serial Bus)如今已成为最为流行的计算机接口。只要与计算机打交道,人们的日常生活已经离不开 USB 端口,通过它们人们可与鼠标、键盘、扫描仪、数码相机、手机和平板电脑等人们想象的到的设备相连接。USB 凭借可靠、高速、方便和省电的优点,已经广泛得到主要的操作系统的支持。

目前大多数的 USB 设计都是进行系统集成,采用现成商用的 USB 芯片进行开发,并没有涉及到对 IP 核的设计与开发。本文通过分析 USB 2.0 协议,使用 Verilog HDL 硬件描述语言设计实现了一个 USB 2.0 设备控制器串行接口引擎 SIE(Serial Interface Engine)核,选取 MC8051 软核作为其微控制器,并通过 Wishbone 片上总线进行连接。

1 USB 2.0 设备控制器系统设计

1.1 系统设计

根据 USB 2.0 设备控制器所要实现的功能,本系统采用自顶向下(Top-to-Down)的设计结构,将设备控制器划分为 7 个主要功能模块:物理层收发器(USB PHY)、UTMI 接口、协议层 PL(Protocol Layer)、RAM 缓冲区、存储

器接口和仲裁器、控制和状态寄存器及功能总线接口。

USB 2.0 IP 核结构框图如图 1 所示。SIE 部分的 SSRAM、功能总线接口、PHY 模块为可更改的,通过对相应模块的更改可使本 IP 满足不同应用的需求。最下面接口用于连接微控制器。

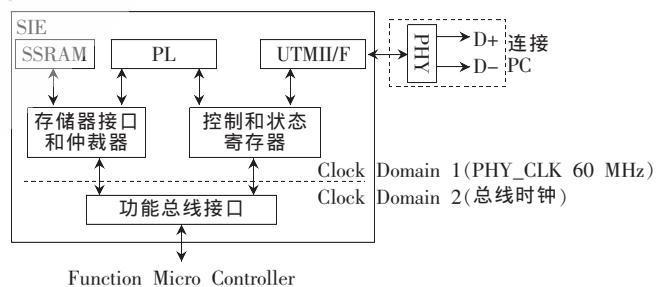


图 1 USB 2.0 IP 核总体框图

由于 USB 2.0 串行总线上最高速率高达 480 MHz,远高于 Standard Cell 电路的处理能力,因此需要全定制的 Transceiver 来进行数据恢复。PHY 的主要作用是将接收来的差分信号进行 NAZI 编码和位填充^[2]。

1.2 各模块具体实现

UTMI 接口是与 PHY 相连接的接口模块,它并不对输入和输出的数据进行处理,主要用于控制总线挂起/恢复模式和全速/高速模式的切换。

协议层 PL 是 USB 设备控制器中最核心的模块,它

* 基金项目:国家科技重大专项“脉冲体制超宽带高速无线通信芯片研发与应用示范”(2009ZX03006-009)

负责所有的 USB 数据 I/O 和通信的控制, 其中包括 DMA 和内存接口、协议引擎 PE (Protocol Engine)、组包 PA (Packet Assembly) 和解包 PD (packet Disassembly) 几个子模块。PL 内部框架如图 2 所示。

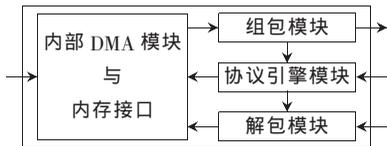


图 2 协议层模块内部结构

PA 和 PD 模块分别是对 USB 包进行装配和拆分。DMA 和内存接口是 PE 模块到内存的接口, 它提供了直接的内存访问 (Direct Memory Access) 和 DMA 块传输。协议引擎 PE 是 PL 中的核心模块, PE 处理所有标准 USB 协议握手和控制通信, 包括 SOF 标记、应答 (ACK、NACK 和 NYET) 以及对 PING 标记的回答。PE 模块将端点控制状态寄存器 (CSR) 的值解码成内部控制信号线, 将端点缓冲区 (Buffer) 寄存器 EP_BUF 的值解码成 Buffer 的大小和指针。图 3 描述了 PE 核心的基本操作。USB 设备总是等待一个来自 USB 总线主机的标记包, 才开始执行其他的操作。一旦接收到标记包, 解包模块将对其解码, PE 执行合适的操作。USB 主循环根据标记包的不同将操作分为特殊包处理、Setup 循环、IN 数据循环和 OUT 数据循环。

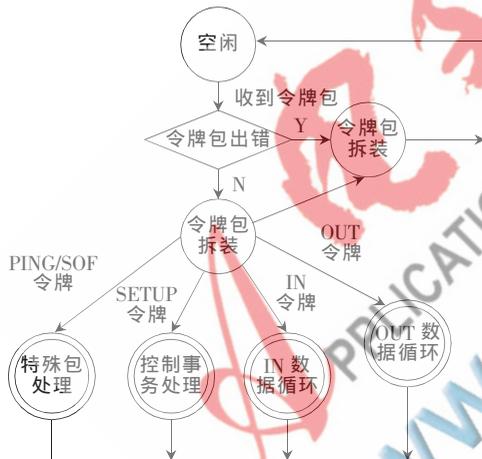


图 3 PE 核心操作流程

存储器接口和仲裁器实际上就是一个 2 选 1 的选择器, 选择 SRAM 与 Wishbone 总线 (及连接微控制器) 还是 SRAM 与 PL 层的 DMA 内存接口连接。

2 微控制器模块

2.1 MC8051 主要特性

在 USB 的通信过程中, 微控制器主要用来处理各种中断, 识别主机发送的是何种请求, 返回相应的数据, 从而完成设备的枚举和之后的数据传输。本设计采用了 MC8051 的 IP 软核, 该 IP 满足 USB 对微控制器功能的要求。通过修改 MC8051 中的固件 (Firmware) 来对 SIE 的寄存器进行配置, 以实现 USB 2.0 设备控制器不同的功能。

2.2 8051 与 USB 2.0 IP 核的连接

由于 SIE 与 MC8051 使用不同的时钟域 (SIE 使用 PHY 提供的 60 MHz 时钟, 而 MC8051 使用自身的时钟), 并且 MC8051 是个通用的 IP 核, 其接口并没有采用本 USB 2.0 IP 核的 Wishbone 兼容接口, 而且 MC8051 是个 8 bit 的通用 MCU (处理数据宽度是 8 bit), 因此 MC8051 和本 USB 2.0 IP 核不能直接相连。

本文设计了一个与 Wishbone 总线兼容的接口转换模块, 以使 MC8051 能正确地连接在 Wishbone 总线上, 完成与 SIE 的数据传输。转换模块主要实现了跨时钟域的数据传输、数据位数转换 (8 bit 到 32 bit 和 32 bit 到 8 bit) 以及 Wishbone 总线兼容信号的加入。实际上完成了一个异步 FIFO 的功能, 但比 FIFO 使用起来更加灵活, 因为 8051 执行每条指令会有一定的指令延时, 如果使用 FIFO, 很难保证 SIE 可以正确的接收到数据。跨时钟域的数据传输主要体现在 SIE 和微控制器之间, SIE 主要使用 PHY 提供的 60 MHz 的时钟, 而微控制器 MC8051 的工作频率不超过 40 MHz。为了防止跨时钟域的数据传输容易产生的亚稳态, 在接口转换模块里采用了“超前送数”的策略, 即 MCU 读写命令在有效信号到达之前把数据放入三态数据端口寄存, 读写信号到达之后进行数据传输。

微控制器与 SIE 之间的数据交换可以分为写入地址与数据和写地址并读数据两种方式, 分别对应着 USB 固件中两种最基本的命令函数。连接转换模块通过状态机来进行控制, 接收到数据后随即进行数据和地址的转换, 并将转换好的数据和地址存入寄存器中。转换完成后进入 WE 状态, 进行判断。如进行写入地址与数据则进入到 WE_AD 状态, 该状态会将 Wishbone 总线的控制位 CYC、STB 和 WE 置高同时输出转换好的地址与数据; 写地址并读数据则进入 WE_A, 该状态将控制位 CYC 与 STB 置高, WE 为低表示读过程, 同时输入要读数据的地址, WE_A 过后就进入 RE_DATA 读取数据并输出。

数据位数转换的原理实际上很简单, 采用标志位对其进行区分, 如图 4 所示。其中, WE 是写入信号, 低电平有效; flag 为 2 bit 的标志位; done 是转换完成信号; datain 和 dataout 分别是 8 bit 输入信号和转换好的 32 bit 输出信号。8 bit 转 32 bit 的过程为: 在 flag 为 00、01、10、11 每段分别向 datain 中写入 1 个 8 bit 数据, 最先写入的为低 8 bit, 然后是次 8 bit, 依次写完。等到 done 信号置高后, 再按照 Wishbone 总线数据的操作方式进行传输。32 bit 数据转 8 bit 与该过程相反。

该连接模块已成功通过 FPGA 验证, 能够正确转换与传输数据。

3 系统验证

3.1 FPGA 验证环境搭建

本文选用 Xilinx 公司的 Virtex-II Pro FPGA 开发板

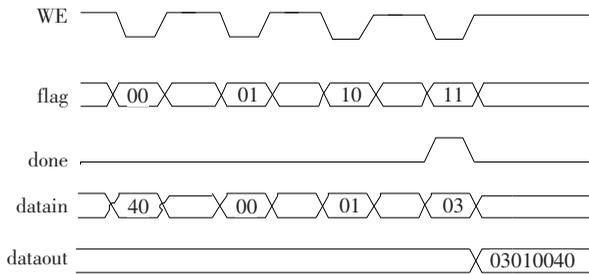


图4 数据位转换

作为验证 USB 2.0 SIE 核的平台。选用 Cypress 公司的 CY7C68000 芯片作为前端的收发器 (PHY), 将 USB 总线上 480 MHz 的串行数据流转换成 8 bit 60 MHz 的并行数据。系统测试平台架构如图 5 所示。

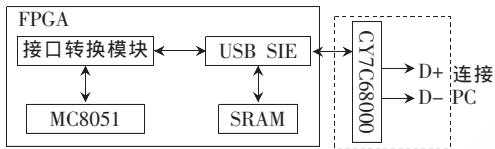


图5 测试平台架构图

CY7C68000 与 FPGA 开发板之间采用 100 脚 Hirose FX2 扩展插槽相连接, 该接口有 LVC MOS25 的 HS_CLKIN 高速时钟输入的接口来作为收发器 PHY_CLK 60 MHz 时钟的引入。缓冲区使用 ISE 软件自带的 RAM IP 核。本设计使用 ISE 10.1.03 对 USB SIE 模块综合后的 FPGA 资源使用情况为: Slices 数为 1 718 个, Slice Flip Flops 数为 1 801 个, 4 输入 LUT 数为 2 885, 最大时钟频率为 127.583 MHz, 满足了 USB 2.0 设备控制器工作所需的 60 MHz 时钟要求。

3.2 结果和分析

将程序下载到 FPGA 实验板上后, 将收发器 CY7C68000 连到主机, 主机马上会出现一个“检测到新硬件”的消息。这表示设备枚举过程的完成, 枚举就是从设备读取一些信息, 了解是什么样的设备, 如何进行通信, 这样主机就可以根据这些信息来加载合适的驱动程序。枚举过程属于控制传输, 一般分为 3 个阶段, 具体过程如下。

(1) 建立阶段。建立阶段如图 6 所示, USB 主机首先

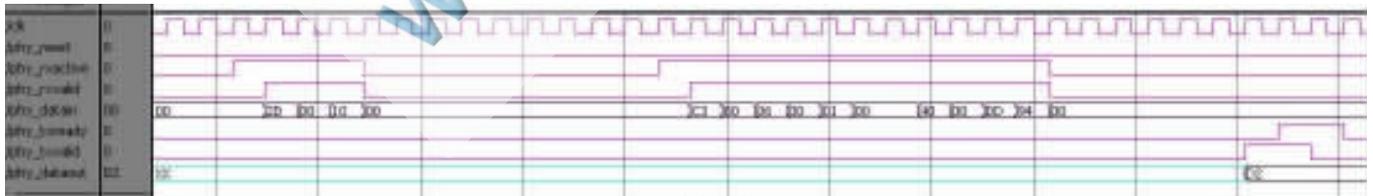


图6 建立阶段



图7 数据阶段

发送来一个 SETUP 令牌包, PID 为 0x2d, 设备地址为 0x00, 端点 0x10 (控制端点)。后面紧跟一个数据包, PID 为 0xc3 是 DATA0 的数据包, 后面的数据 0x80 0x06 0x00 0x01 0x00 0x00 0x40 0x00 表示这是一个 Get_Descriptor 标准设备请求的数据包。USB 设备收到并检测无误之后会返回一个 ACK (PID 为 0xd2) 握手包告诉主机已收到数据。

(2) 数据阶段。主机会发出一个 IN 包, 如图 7 所示, PID 为 0x69, 设备收到 IN 包后用数据包 DATA1 (PID 为 0x4b) 返回它自己的设备描述符, 描述符里有该设备自身特性的信息。主机收到数据包后会返回一个 ACK 握手包 (PID 为 0xd2), 表示数据接收正确。

(3) 状态阶段。主机会发出 OUT 包, 但与数据阶段不同, 状态阶段所发数据包内容为空。设备收到数据包后返回一下 ACK 握手包表示枚举过程的结束。

本文提出的 USB 2.0 设备控制器 IP 的架构具有很强的实用性和复用性。它既可以作为单独的设备芯片使用, 也可与其他 IP 一起进行 SoC 设计, 只需在片上总线上添加相应的模块。单独设计的接口转换模块成功完成内嵌 8051 微控制器核的设计, 也可以利用该模块外接一个 51 系列微控制器, 具有很强的灵活性。在操作系统上添加相应的驱动程序即可进行数据传输的实验。

参考文献

- [1] Universal serial bus specification (Rev 2.0) [S]. www.usb.org, 2002.4.
- [2] USB 2.0 transceiver macrocell interface (UTMI) Specification (Version 1.05) [S]. 2001.3.
- [3] 周立功. USB 2.0 与 OTG 规范及开发指南 [M]. 北京: 北京航空航天大学出版社, 2004.
- [4] AXELSON J. USB 开发大全 [M]. 北京: 人民邮电出版社, 2011.

(收稿日期: 2012-04-11)

作者简介:

陈亮, 男, 1987 年生, 硕士研究生, 主要研究方向: 数字集成电路设计。