

基于 STM32 的 USB OTG 数据通信的实现

龚跃玲, 汪玲彦

(武汉理工大学 自动化学院, 湖北 武汉 430070)

摘要: 针对一般 USB 装置只能在 PC 机周围进行数据通信的限制, 提出了基于 USB OTG 接口与 STM32 微控制器相结合的智能通信的设计方法。该方法是在基于 Cortex 内核的 32 bit RAM 芯片 STM32F107 上集成 USB OTG 接口上实现, 它利用了 STM32 集成度高、成本低、功耗低、开发方便、性能可靠和 USB OTG 双角色特性等优点, 解决了无 PC 机时 USB 从设备之间数据传送难的问题。并在金牛开发板上实现它与 PC 机以及 U 盘之间的通信, 完成了 USB 嵌入式设备之间数据的传输。

关键词: USB OTG; STM32F107; U 盘

中图分类号: TG434

文献标识码: A

文章编号: 1674-7720(2011)10-0032-03

Implementation of USB OTG data communication based on STM32

Gong Yueling, Wang Lingyan

(Automation School of Wuhan University of Technology, Wuhan 430070, China)

Abstract: For USB devices completing data communication only around the PC, this paper introduces the design approach of intelligent communications based on combination with USB OTG interface and STM32 microcontroller. It realizes on the STM32F107 chip of 32 bit RAM chip based on the Cortex core. The design make of the advantage of STM32, such as it's high integration, low cost, low power consumption, the facility of development, the reliable performance, and the dual role of USB OTG. It solve the difficult problem of data transmission between USB devices broken away from PC. It realized the communication between the USB OTG interface and PC or U disk, which completes data transmission of the embedded device.

Key words: USB OTG; STM32F107; U disk

移动数据的交换和存储, 是近年来 IT 行业的热点。随着 USB 技术的产生, USB 技术已逐渐应用在移动数据领域中。目前使用的 USB 移动设备都只能通过 PC 机进行相互的文件和数据交换。随着 USB 技术的逐渐成熟, 人们希望通过移动设备直接与 USB 外设通信, 使得 USB 能应用在没有 PC 的领域中。

USB OTG (On The Go)^[1] 可满足这些要求, 即一个拥有 OTG 功能的设备既可以扮演主机的角色操纵其他 USB 从设备, 同时又可以担当从设备的角色接受主机的支配。例如, 照相机可以直接连到 USB 打印机上打印数据。当今 USB 主机的嵌入式应用就成了 USB 领域新的研究焦点。该技术对于便携式设备、野外作业设备间的数据传输将是一个重要突破。基于 Cortex-M3 内核的 STM32 系列的 ARM 嵌入式处理器因其高的性价比而正被广泛应用。将 USB 与 ARM 相结合是进行数据采集、处理与传输的理想解决方案。本文重点讨论基于 STM32

的 USB OTG 的数据通信。

1 USB OTG 规范

USB OTG 规范是 USB 2.0 规范的补充, 它完全遵守 USB 2.0 规范的外围设备。在 USB OTG 规范之下, 任意两个符合规范的 USB 产品都可以互相连接进行通信, 而不是必须通过 PC 机才能通信, 从而实现了各类 USB 产品完全脱离电脑进行数据传输的功能。

双重角色设备必须具备有限的主机能力, 同时还需要一个 OTG 目标列表, 即包括与之通信的设备信息、设备类型、制造商等。它支持会话请求协议 SRP 和主机交换协议 HNP。SRP 协议用于 B 设备向 A 设备请求开始 USB 会话, HNP 协议用于 USB 会话过程中在 A 设备和 B 设备之间传递主机地位^[2]。

2 STM32 的 USB OTG 全速控制器

ST 公司在基于 ARM 公司的 Cortex-M3 架构上研发出 32 bit STM32 处理器后, 凭借其丰富的外围模块、高

硬件纵横 Hardware Technique

速指令执行速度、低功耗、低价格而备受市场青睐。2009年又推出了 STM32F107 互联型系列微控制器,它集成了 USB OTG 接口,配合了 USB HOST 和 SPI 功能,让微控制器可以读取 USB 大容量外存、MP3 播放器、SD 记忆卡等文件。STM32F107 芯片内集成了 USB OTG 模块功能图如图 1 所示。

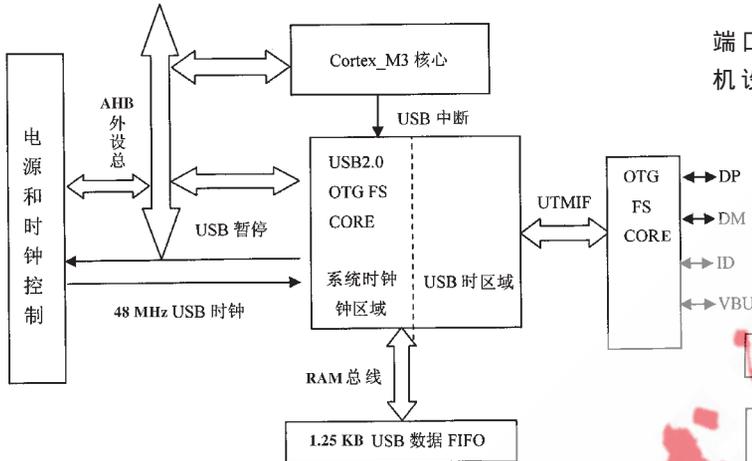


图 1 USB OTG 模块功能图

USB OTG 全速控制器从复位和时钟控制模块(RCC)中获得时钟,微控制器内核(CPU)通过 AHB 外设总线访问 OTG 全速控制器的寄存器,USB 事件由单独的 USB OTG 中断控制线通知微控制器内核^[3]。

每个发送 FIFO 都配置了一个 PUSH 寄存器,微控制器以向 PUSH 寄存器写 32 bit 数据的方式向 USB 控制器传输数据,每一个 OUT 端点或 IN 通道都有一个 POP 寄存器,微控制器通过读 POP 寄存器获得来自 USB 总线的 32 bit 数据,这些数据自动从共享的接收 FIFO 中载入,接收 FIFO 位于总共 1.25KB 的 USB 数据 RAM 区。

USB 协议层由串行接口控制器(SIE)驱动,并连接到由内置物理层(PHY)支持的 USB 全速/低速收发模块。OTG PHY 由 OTG 全速控制器控制,并通过 UTMIF+总线(UTMIFS)的全速子集来收发控制和数据信号。它包括了上拉/下拉电阻的 ECN 电路。内置了 ID 线的上拉电阻,用于区分是主机状态还是设备状态,若 ID 线浮空,内置的上拉电阻将检测到 ID 线的高电平,此时控制器处于默认的设备模式下。若 ID 线接地,自动切换到主机模式,并需要软件初始化主机模式。

DP/DM 线内置了上拉和下拉电阻^[4],在设备模式下,当 VBUS 线上出现了有效的电平,控制器使能 DP 线的上拉电阻,向主机通告接入一个 USB 全速设备。在主机模式下,控制器同时使能 DP 和 DM 线的下拉电阻。上拉和下拉电阻可以在控制器通过主机协商协议(HNP)切换角色类型时动态地切换。

3 USB OTG 软件设计

STM32F107 芯片集成了 USB OTG 功能,USB OTG 固

件程序的设计可以完成数据传输的所有操作及功能。其结构图如图 2 所示。其中驱动程序栈包括主机驱动程序栈、从机驱动程序栈、USB OTG 驱动程序栈。当完成固定程序设计后,系统软件会根据硬件的连接检查 ID 线的状态来选择工作模式是设备模式还是主机模式。程序流程图如图 3 所示。

如果工作于主机模式下,则进行相应初始化、检测端口,在检测到有设备接入时复位总线、枚举并配置从机设备,在完成对从机识别后,可通过查询从机的 OTG 性能描述符判断是否支持 HNP 协议(即是否为两用 OTG 设备)。当工作在从机状态时则等待主机对其完成枚举。本文是在 STM32 开发工具 RealView MDK 的基于 STM32 固件库上进行开发,实现与 U 盘、PC 机间的通信。

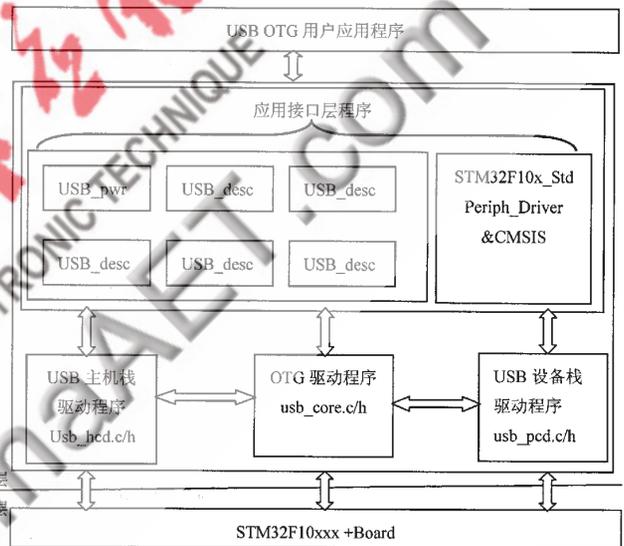


图 2 USB OTG 固件程序设计

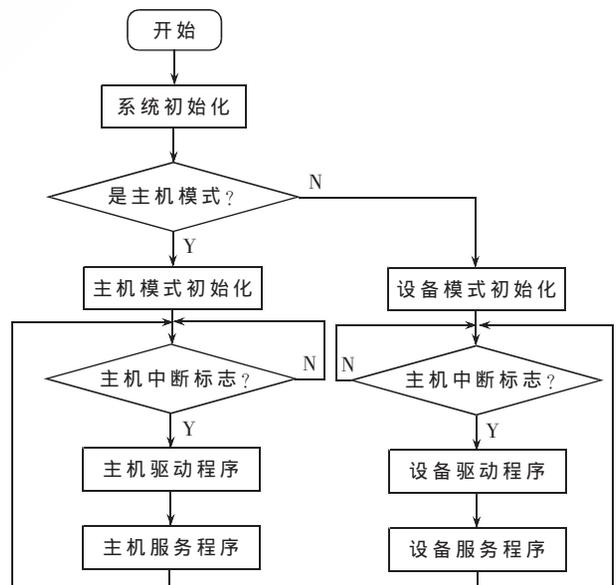


图 3 USB OTG 驱动程序流程图

3.1 U 盘读写的实现

U 盘属于 USB 大容量存储设备,它具有容量大、可移动、数据交互方便等优点,因而实现与 U 盘的读写具有很强的应用价值和市场前景。要实现 U 盘读写,USB 主机必须具有相应的驱动^[5],对各种读写指令进行封装、解释和执行。

在系统进入主机模式前应先给开发板提供 5 V 供电,开启系统时钟、USB OTG 时钟,然后调用 void HOST_Init (USB_OTG_CORE_DEVICE *pdev) 函数将 USB OTG 初始化为主机模式,接着调用 HOST_ChannelInit (USB_OTG_CORE_DEVICE *pdev, USB_OTG_HC *pHost Channel) 初始化传输通道,其中包括总传输长度、期望接收到的数据包数、设备端点传输类型、速度、方向的配置。最后根据主机传输协议在初始化通道内进行传输,可以调用下列函数 uint32_t HOST_StartXfer (USB_OTG_CORE_DEVICE *pdev, USB_OTG_HC *pHostChannel) 进行通信。

U 盘是大容量存储设备,支持 Bulk-Only 传输协议,USB 主机在能够正确操纵 U 盘之前必须先完成 USB 总线的枚举,在控制传输阶段先获取从设备的信息,然后根据这些信息对从设备进行重配置后,才能建立主从通信。此外,程序中还要实现标准 Mass Storage 类协议中的磁盘操作命令 UFI,它能完成读、写、格式化磁盘等。最后建立了 FATFS 文件系统,它兼容了 FAT16、FAT32 文件系统,它是 UFI 与移动盘上文件数据连接的桥梁,UFI 命令的所有数据流只有按照 FATFS 标准协议传输,才能顺利实现文件读、写等功能。

程序中实现 U 盘的 BOT 传输,除了规定控制传输端点 0 之外,还定义了输入、输出端点,BOT 状态机的 5 个状态,两个状态寄存器 CBW、CSW。端点输出中断程序完成写 U 盘,端点输入完成读 U 盘操作。其流程图分别如图 4、图 5 所示。

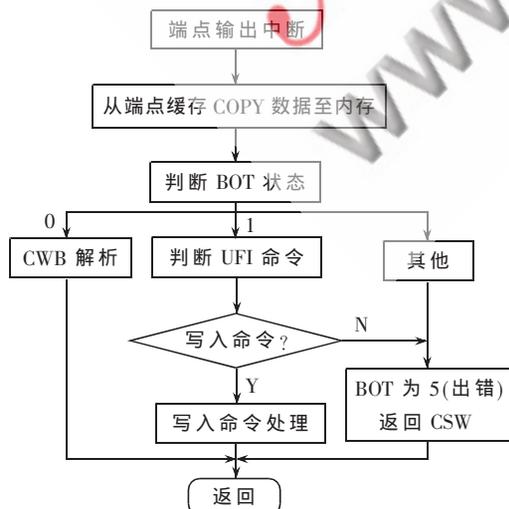


图 4 写 U 盘流程图

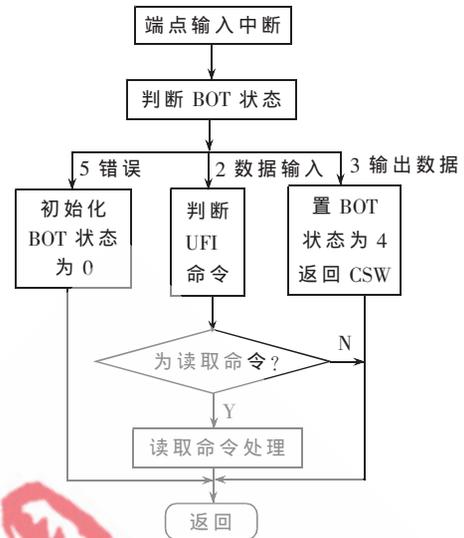


图 5 读 U 盘流程图

3.2 与 PC 机之间通信

与 PC 机进行通信,系统作为 USB 从设备时,要应答 PC 主机的标准请求、处理 USB 总线事务和用户功能^[6]。首先调用 void USB_OTG_USBD_Init () 将 USB OTG 工作模式配置成从机模式,接着调用底层驱动 API 函数 USB_OTG_USBD_EP_Open () 来激活端点进行数据传输,USB OTG 定义了三个端点,程序中定义端点 0 在控制传输中应答设备枚举,端点 1 的功能为向 PC 机发送数据,端点 2 的功能为接受 PC 机发送的数据。PC 主机枚举系统设备时,必须先获得 USB OTG 端点的配置参数。最后通过调用如下的 PCD 层 API 函数完成 SETUP 包、IN 包、OUT 包的传输。

```

uint32_t USB_OTG_USBD_EP_Read();
uint32_t USB_OTG_USBD_EP_Write();
uint32_t USB_OTG_USBD_EP_Stall();
uint32_t USB_OTG_USBD_EP_ChrStall();
uint32_t USB_OTG_USBD_EP_Flush();
  
```

本文采用意法半导体公司互型控制芯片 STM32F107 集成的 USB OTG 接口,在基于金牛开发板上实现了从设备和主机的数据传输,作为从设备,PC 机端的应用程序可以通过 USB OTG 开发板的 SDRAM 进行数据的读写,可稳定在 500 kb/s;作为主机,可以枚举连接到 USB OTG 接口上的 U 盘实现了对 U 盘的读写,最高读写速率可达 800 kb/s。

本文利用了 STM32F107 芯片高度集成的 USB OTG 接口,其开发系统性能好、可靠性高、开发方便,开发者只需在软件上编程实现数据传输。同时 STM32 芯片提供了相关固件库,在此基础上进行开发提高了效率。本文的创新点就在于充分结合和利用了 STM32 处理器低成本、低功耗、高可靠性与 USB OTG 技术的优点,对于已经编程实现的主机、从机栈程序有很好的移植性,对

于以后数据传输的开发具有很好的参考性。将 USB 与 STM32 相结合实现 USB 嵌入式设备之间的数据传输,将会在移动数据传输领域具有很好的应用前景。

参考文献

[1] On The Go Supplement to the USB2.0 Specification,Rev1.0 [S].www.usb.org,2001.
[2] 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.
[3] 王永虹,徐炜,郝立平.STM32 系列 ARM Cortex-M3 微控制器原理与实践[M].北京:北京航空航天大学出版社,2008.
[4] ST Microelectronics. STM32 Reference manual[Z].2009.

[5] 宋宇宁,周兆英,赵焕军. USB OTG 扩展子板的实现[J]. 电子技术应用,2006,32(5):2-3.
[6] 张洪波,江海河,贾先德. USB OTG 技术在数据采集系统的应用[J]. 微计算机信息,2006,(22):2-3.

(收稿日期:2011-01-10)

作者简介:

龚跃玲,女,1958 年生,副教授,主要研究方向:新型传感技术与智能仪器、嵌入式系统、远程监控及远程故障诊断。

汪玲彦,女,1985 年生,硕士研究生,主要研究方向:嵌入式系统、智能控制与智能系统。

