

DOI: 10.3969/j.issn.1007-2861.2012.05.014

SPI 总线用于双处理器通信的数据链路层设计及其实现

孙斌, 胡越黎, 王昆

(上海大学 机电工程与自动化学院, 上海 200072)

摘要: 提出一种基于串行外设接口(serial peripheral interface, SPI)总线的双处理器间通信的数据链路层设计方法. 该方法中的双向 SPI 总线具有占用引脚少(仅比 SPI 总线多一根线)、有完善的数据链路层协议控制的优点,能够保证数据透明地在双处理器之间进行传输. 以一款带有显示单元的车身总线控制器为应用对象,采用 ARM9 和 S12 作为双处理器实现双向通信,并对这一通信方法的实现和应用情况进行描述. 实验表明,采用该方法设计的协议比较完善,能够满足实际工程中的需求,对于其他 SPI 的应用也有很好的借鉴价值.

关键词: 双向通信; 串行外设接口; 双处理器通信; 数据链路层

中图分类号: TP 29

文献标志码: A

文章编号: 1007-2861(2012)05-0513-06

Data Link Layer Design and Implementation Based on SPI Bus for Dual-Processor Communication

SUN Bin, HU Yue-li, WANG Kun

(School of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China)

Abstract: This paper proposes a data link layer design method for dual-processor communication based on serial peripheral interface (SPI) bus. Advantages of the two-way SPI bus in this design include using less pins (only one line more than the SPI bus) and perfect data link layer protocol control. These can ensure transparent data transfer between the two processors. An automotive body bus controller with a display unit was used as an application. It uses ARM9 and S12 as dual-processors to achieve two-way communication. Implementation and application of the method are described. Experiments show that the protocol is effective in engineering practices, and provides a reference to other SPI applications.

Key words: two-way communication; serial peripheral interface (SPI); dual-processor communication; data link layer

系统级处理器间通信的方式有很多种,其中 RS-232, I2C 等串行通信总线都已经成为较为普遍

的选择之一,但对高速处理而言,这些串行通信总线速率都较低. 相对而言, 串行外设接口(serial

收稿日期: 2011-09-21

基金项目: 上海市科委集成电路专项资金资助项目(09706201300); 上海市经济与信息化委员会资助项目(090344); 上海市高新技术产业化新能源汽车资助项目(09625029); 上海大学研究生创新基金资助项目

通信作者: 胡越黎(1959~), 男, 教授, 博士生导师, 博士, 研究方向为微电子与集成电路设计、汽车电子与系统集成、机器视觉与片上多处理系统架构. E-mail: huyueli@shu.edu.cn

peripheral interface, SPI) 作为一种高速的、全双工、同步的通信总线,其通信速率基本可以按照通信双方处理器的处理速度来设定,满足了高速处理器对于通信的要求。

SPI 总线系统是一种同步串行外设接口总线,它采用主从工作方式,即在这种模式下通常有一个主设备和一个或多个从设备,其中主机控制数据传输,从机配合主机完成传输任务。在硬件资源上,SPI 通常采用四线制进行双向通信,三线制实现单向传输。在整个通信过程中信号线只能由主设备控制,从机对通信过程没有控制作用。但这种通信方式主要存在如下缺点:① 缺少组帧机制;② 通信过程中没有指定的数据流,缺少确认是否收到数据的应答机制^[1];③ 无任何校验,缺少控制传输错误的机制;④ 缺少对数据流的控制;⑤ 缺少对数据链路的维护机制,无建立、维持、释放的管理机制等。

从开放式系统互联(open system interconnect, OSI)七层模型的角度而言,SPI 通信本身仅仅规定了通信的物理层协议,而缺少数据链路层的协议。

1 研究现状及本研究的主要思路

就目前的研究和应用现状而言,SPI 通信作为一种物理层通信协议,对于其作为双向通信时的通信方式,已有了很多应用。Gao^[2]采用两组 SPI 总线为双 DSP 提供双向通信;Akos 等^[3]使用多从机选通信号(slave select, SS),将 SPI 总线作为多处理器通信的基础;桑伟等^[4]描述了一种基于 SPI 的双向通信方式,主设备和任一从设备之间至少包括了 4 条握手信号线;夏冰等^[5]描述了一种基于 SPI 总线的多核通信的方式,使用了一根带有下拉电阻 R1 的公共握手信号线。

本研究在上述应用的基础上,完成了一种用于双向通信的 SPI 的物理层修改和数据链路层设计,设计中主要包括如下处理与改进:① 针对缺少组帧机制,建立一组有效的帧结构,适应双处理器通信的环境;② 针对缺少应答机制等,基于 SPI 主从通信方式,采用基于单线握手信号的方式使其支持双向通信;③ 针对无校验的情况,建立基于循环冗余校验(cyclic redundancy check, CRC)方式的可靠通信;④ 针对缺少数据流控制的情况,采用缓冲区控制的方式,实现 SPI 流量控制;⑤ 针对缺少链路维护机制,建立基于握手信号和帧支持相结合的数据链路

维护机制。总的来说,就是建立起一套完整的数据链路层,以适应双处理器通信的环境。本研究将带有数据链路层的 SPI 协议称为 two-way SPI。

2 Two-way SPI 协议设计

2.1 物理层设计

为了能够实现 SPI 的数据链路层机制,本研究在 SPI 的四线制基础上增加一条接有上拉电阻的握手信号线——HandShaking(简称“HS”),这是一条单向(从机到主机)的信号线(见图 1)。HS 信号的作用主要是告知主机:从机有数据需要发送。

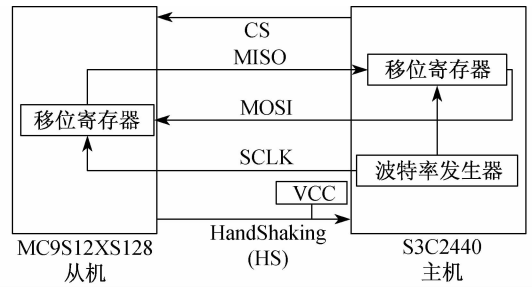


图 1 Two-way SPI 物理层

Fig. 1 Two-way SPI physical layer

2.2 信息传输格式

本研究采用了“0 比特插入法”,并将“确认字符”(0x7E)作为流量控制的一个标志。作为主从通信方式,本研究设计的帧格式主要包括发送方帧格式和接收方帧格式两种。

2.2.1 发送方帧格式

发送方帧结构如图 2 所示。为了能够满足发送方的业务需求,增加了地址段、控制段和 CRC 段,其中控制段信息包括功能 ID 和长度信息,在采用 8 位传输的情况下,功能 ID 和长度信息均为 4 bit;在采用 16 位传输的情况下,功能 ID 和长度信息均为 8 bit。这里的长度信息主要是指信息段的长度。

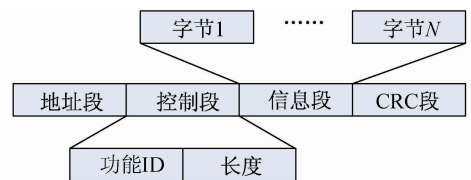


图 2 信息帧结构

Fig. 2 Information frame

2.2.2 接收方帧格式

为了能够有效控制数据流,考虑到 SPI 接收方

的特点,接收方的同步反馈信息为奇数字节时,为接收端处理器的“接收缓冲区剩余容量”;为偶数字节时,为接收端处理器之前接收到的数据是否处理完的标志,即“确认字符”(0x7E).接收端处理器在接收到发送端处理器发送的第 N 字节时,完整的反馈数据发送顺序及内容如图3所示.



图3 反馈字节
Fig.3 Feedback bytes

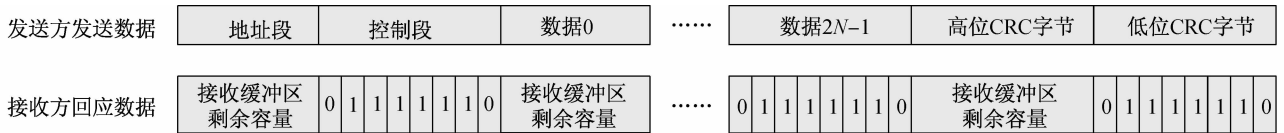


图4 发送及确认过程
Fig.4 Sending and confirmation process

2.3 同步机制

基于 SPI 总线已经规定的片选 (chip select, CS) 信号线和新建的 HS 握手信号线,整个同步过程按照主机发起和从机发起,结果有所差异.

2.3.1 主机起始发送方式

由主机发起的数据传送与普通的 SPI 传送并没有区别,也就是说,这种方式完全兼容传统的 SPI 通信,其时序图可参考文献[2].

2.3.2 从机起始发送方式

从机发起的通信由 HS 信号向主机申请开始发送,主机在接收到 HS 信号后,开始 SPI 发送(见图5).由于绝大多数微控制器(micro control unit, MCU)的中断处理以下降沿处理为主,因此,采用了外部上拉,由从机 MCU 置0的方式.在主机响应中断后,HS 信号在从机发送结束后被取消.

2.4 流量控制

本研究采用如下方式进行流量控制.在发送方发送过程中,每两个 Bytes 就有一个接收缓冲区剩余容量(C_b)单元(见图4).当接收方回应的接收缓冲区容量小于2($C_b < 2$)时,则发送方应停止发送;若当前正在发送过程中,则中断当前帧.对于主机起始发送来说,即将 CS 信号拉高;对于从机起始发送来说,即将 HS 信号拉高.发送方等待随机时间间隔后,重新尝试发起发送.

当发送方发送奇数字节(第1,3,5,⋯字节)时,接收方反馈回发送方的信息为“接收缓冲区剩余容量”;当发送方发送偶数字节(第2,4,6,⋯字节)时,接收方反馈回发送方的信息为“确认字符”(0x7E).当发送方连续发送多个字节时,接受方反馈过程如图4所示.

这种接收方式能够有效地控制每一个字的传输速度,避免前一字节尚未传输完成,就已经开始传输下一个字节的情况出现.同时,发送方对接收方的接收缓冲区大小也是能预测的,当接收缓冲区将要溢出时,发送方可以进行处理,减少接收方的负担.

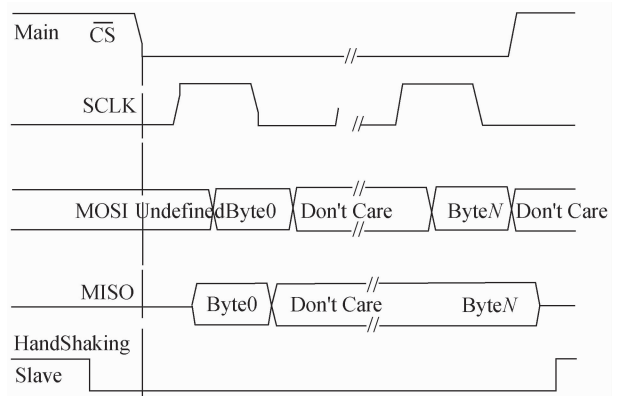


图5 从机发起的 SPI 通信流程
Fig.5 SPI communication processes launched by slave

2.5 错误处理

在这种链路层协议中,主要有以下两种错误:一是接收方 CRC 错误;二是确认字符错误.

本研究中的 CRC 采用 16 bit CRC 码,其生成多项式为 $X^{16} + X^{12} + X^5 + 1$.当出现 CRC 错误时,接收方将接收缓冲区剩余容量(C_b)置为0,即中止原发送方的继续发送;然后接收方发起发送,将控制段中的控制 ID 置为错误标识符 0xf,即控制段为 0xf0.发送的数据帧如图6所示.



图6 接收方发送的错误帧标识
Fig.6 Error frame mark sent by receiver

第二种错误是确认字符错误,即发送方没有在偶字节接收到“确认字符”(0x7E),则发送方中断本次发送,即将CS信号(主机端)或者HS信号(从机端)拉高;然后,在随机时间间隔后重新发起发送。

3 在实际工程中的应用

以上介绍了一套完整的基于 SPI 进行双处理器通信的数据链路层协议. 本研究将协议 two-way SPI 应用于一种新型的带有显示功能的车身总线网络控制器中,并取得了良好的效果。

3.1 系统硬件架构

在本研究提出的设计方案中,系统主要以高级精简指令集处理器 9 (advanced RISC machines 9, ARM9) 系列模块和 Freescale 生产的 S12 处理器模块为核心. 系统的架构如图 7 所示。

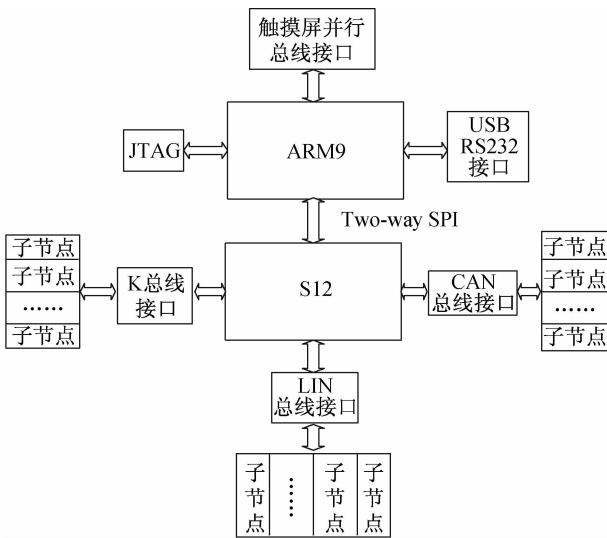


图 7 系统架构图

Fig. 7 System architecture

图 7 中的 ARM9 模块和 S12 处理器之间采用 two-way SPI 总线作为通信接口. 由 S12 处理器负责 KWP2000 总线、CAN 总线和 LIN 总线的通信,由 ARM9 模块负责触摸屏的显示和触摸操作,其中 ARM9 的地址段为 0x00, S12 处理器的地址段为 0x01,即 ARM9 为主处理器, S12 为从处理器。

系统中 ARM9 运行的是 Linux 操作系统, S12 运行在 uC/OS-II 操作系统上. 对于 Linux 操作系统而言,其中断处理模式与简单的裸机程序有所不同,需要进行相应的硬件驱动的设计. 在本研究中,主要是 SPI 总线硬件驱动的设计^[6-9]。

3.2 系统软件的实现

3.2.1 系统中的 two-way SPI 通信

ARM9 和 S12 的通信模块都通过 two-way SPI 总线协议进行通信,这一协议的具体实现过程如下。

主机发起通信的流程图包括图 8 所示的主机处理流程图和图 9 所示的从机接收处理流程图。

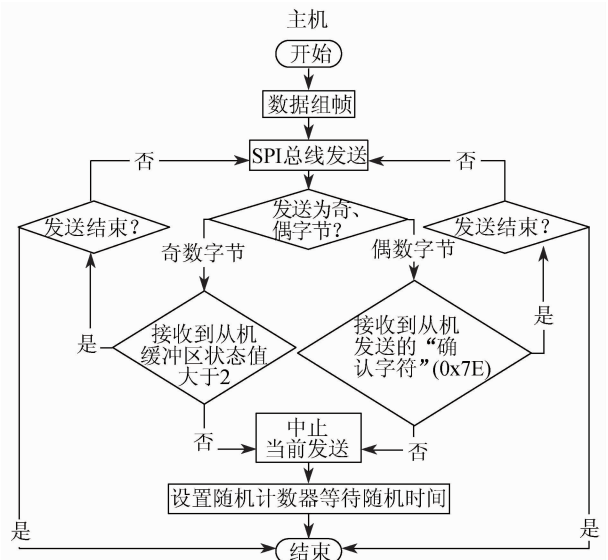


图 8 ARM9 发起的 two-way SPI 通信过程中的主机处理流程图
Fig. 8 Host processor sending flow chart in two-way SPI communication process launched by host processor (ARM9)

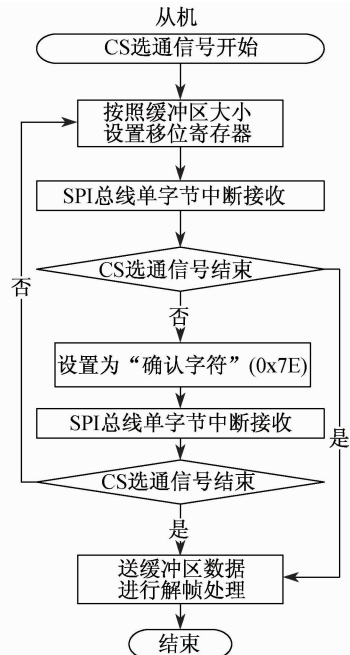


图 9 ARM9 发起的 two-way SPI 通信过程中的从机处理流程图
Fig. 9 Slave processor receive flow in two-way SPI communication process launched by host processor (ARM9)

图 10 为 ARM9 作为通信发起方的实验情况,发送的 ID 为 0x00,发送的数据段为 0xFC,0x7C. 经过 0 bit 插入后,变成 0xFa,0x3E 和 0x00. 为了保证反馈字节能够满足偶数,又加入了一个 0x00;然后是 CRC 段(0x51,0xb8). 经过组帧后,主机开始按照 two-way SPI 总线协议发送,并在发送过程中不断接收 S12 反馈回的接收状态信息,包括“接收缓冲区剩余容量”和“确认字符”(0x7E).



图 10 主机发起 two-way SPI 通信的测试情况
Fig. 10 Test case for two-way SPI communication launched by host

当从机发起通信时,其主从机处理流程和图 8、图 9 的通信处理流程正好互换,即主机使用图 9 的处理流程,而从机使用图 8 的通信流程.

图 11 为 S12 作为通信发起方的实验情况. 从图 10 和图 11 中可以看出,双方通过 two-way SPI 都可以很好地完成双向通信,步骤清楚,数据传输正常.

3.2.2 系统功能的实现

经过实际测试,two-way SPI 作为底层双处理器通信协议,在实际系统中运行稳定,并且能够满足 CAN 总线、LIN 总线和 KWP2000 总线在 1 MHz 通讯速率时的数据处理和网络分析等功能^[10]. 可见,本研究中的 two-way SPI 总线通信具有较高的实际工程应用价值.

3.2.3 Two-way SPI 总线稳定性分析

影响本研究中 two-way SPI 总线稳定性的因素主要包括:① 当主、从机同时需要进行数据发送时;② 从机在 HS 信号使用后,出现处理异常的情况,而没有将 HS 信号复位为高电平.

针对情况①,当主、从机同时需要进行数据发送



图 11 从机发起 two-way SPI 通信的测试情况

Fig. 11 Test case for two-way SPI communication launched by slave

时,从物理层上表述,即 HS 信号和 CS 信号同时被置为低电平. 从主机角度而言,其会开始传递数据,但是在 two-way SPI 数据链路层上,由于主机不能接收到从机反馈的“确认字符”,所以其会停止发送,开始等待随机时间间隔. 就从机角度而言,也会出现相同情况. 由于主、从机的随机时间间隔并不同,所以下一次出现冲突的概率就更低.

针对情况②,如果从机的 HS 信号始终没有置为高电平,主机在多次尝试标准 SPI 传输,并且从机发出的数据始终不符合规定的 CRC 后,主机应该提示从机端工作不正常. 这种状态实质上是主机系统崩溃的一种状态,该状态应该是单处理器系统设计的合理性考量,而非本研究通信过程稳定性的问题.

综上,本研究中的 two-way SPI 通信方式的稳定性还是较好的.

4 结束语

本研究提出了一种基于 SPI 总线的双处理器间通信的数据链路层设计,并将该设计应用到实际的工程实践中. 该 two-way SPI 总线具有占用很少引脚(仅比 SPI 总线多一根线)、有完善的数据链路层协议控制的优点,能够保证数据透明地在双处理器之间进行传输. 在实际工程应用中,本研究采用了 ARM9 和 S12 通信的方式,并对这一通信的过程和实际应用情况作了描述. 通过实验可以证明,该协议设计比较完善,能够满足实际工程中的需求,对于其他 SPI 的应用,也有很好的借鉴价值.

参考文献:

- [1] SUN B, HU Y L. Automatic body control model experimental platform design based on CAN-bus [J]. *Advanced Materials Research*, 2012, 383:4269-4273.
- [2] GAO Z. Implementation of SPI and CRC algorithm in data communication between two DSPs [J]. *Electronic Engineering & Product World*, 2011, 18(1):46-48.
- [3] AKOS S, TIBOR S, ZOLTAN H. Realising the SPI communication in a multiprocessor system [C] // 5th International Symposium on Intelligent Systems and Informations. 2007:213-216.
- [4] 桑伟,潘时林.一种基于SPI实现数据传输的方法和系统:中国,200910141461[P]. 2009-05-15.
- [5] 夏冰,曾敬,李荣正.基于SPI结构模型双向式身份识别及信息交互平台及方法:中国,200810039686[P]. 2008-06-26.
- [6] SANGTAE H, INJONG R. Taming the elephants: New TCP slow start [J]. *Computer Network*, 2011, 55(9): 2092-2110.
- [7] WANG T K, CHANG F R, LIN S Y. Multidevice time measurement system via a PTPD network [J]. *IEEE Transactions on Instrumentation and Measurement*, 2011, 60(7):2304-2307.
- [8] BAHLMANN C, BURKHARDT H. The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, 26(3):299-310.
- [9] SALAH K, QAHTAN A. Implementation and experimental performance evaluation of a hybrid interrupt-handling scheme [J]. *Computer Communications*, 2009, 32(1): 179-188.
- [10] SUN B, HU Y L, WANG K. Diagnosis design based on KWP2000 in body control module [C] // ICMTMA 2011. 2011:1071-1074.