

·信号与信息处理·

基于STM32F103的HDLC通信接口设计与实现

李永强, 郭少雷

(中国电子科技集团公司光电研究院, 天津 300308)

摘要:在中高速同步半双工和全双工通信领域,高级数据链路控制(HDLC)协议因其强大的差错检测、高可靠性、高效率、透明传输等优点而被广泛使用。在航空、航天等特殊的使用场合,由于对于数据传输正确性和可靠性的要求更高,在实际使用中往往对HDLC协议进行改进。介绍了一种采用HDLC特殊数据帧结构实现单路半双工数据通信的接口设计方案。该方案采用STM32F103微处理器和差分收发芯片架构,通过软件编程的方式,实现了HDLC特殊格式数据帧的解析与生成。测试结果表明,该通信接口设计方案的合理、正确和可行。

关键词:高级数据链路控制;STM32F103;软件编程

中图分类号:TP311.56

文献标识码:A

文章编号:1673-1255(2018)-05-0044-05

Design and Implementation of HDLC Communication Driver Based on STM32F103

LI Yong-qiang, GUO Shao-lei

(Academy of Opto-Electronics, China Electronic Technology Group Corporation (AOE CETC), Tianjin 300308, China)

Abstract: Because of the advantages of powerful error detection, high reliability, high efficiency and transparent transmission, high level data link control (HDLC) protocol is widely used in the field of medium and high speed synchronous half-duplex and full-duplex communication. In special occasions such as aviation and aerospace, due to higher requirements of transmission correctness and reliability, HDLC protocol always needs to be improved. A HDLC interface design using special data frame which completed single channel half-duplex data communication is described. Using STM32F103 microprocessor and differential driver and receive pair chips, this design achieves parsing and generation of HDLC data using special frame by the way of software programming. The test results show that the communication interface design is reasonable, correct and feasible.

Key words: high level data link control (HDLC); STM32F103; software programming

高级数据链路控制(high level data link control, 简称HDLC)协议,由国际标准化组织ISO制定,是一种在同步网上传输数据、面向比特的数据链路层协议^[1]。

与面向字符的基本型传输控制协议相比较,HDLC协议具有以下几个特点^[2]:

(1)不依赖任何一种字符编码集,数据报文可透明传输;

(2)支持全双工通信,可不必等待确认便可连

续发送数据,具有较高的数据链路传输效率;

(3)所有帧均采用CRC校验,对信息帧进行编号,可防止漏收或重发,传输可靠性高;

(4)传输控制功能与处理功能分立,具有较大灵活性和完善的控制功能。

由于以上的优点,HDLC已广泛应用于军事及航空、航天等领域;同时,也成为其他许多重要的数据链路层协议的基础。

1 HDLC 帧结构

HDLC 的基本帧格式^[1]如图 1 所示。它由六个字段组成,这六个字段可以分为 5 种类型,即标志字段(F)、地址字段(A)、控制字段(C)、信息字段(I)、帧校验字段(FCS)。在基本帧结构中允许不包含信息字段(I)。

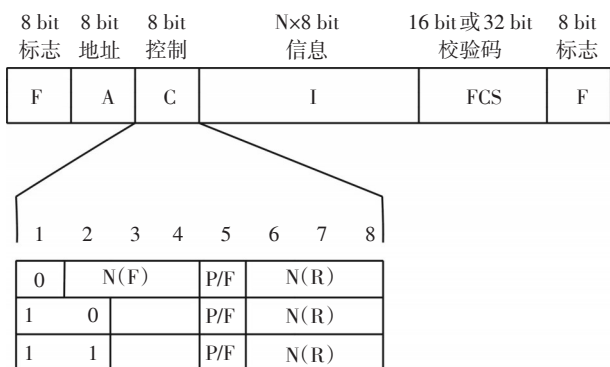


图 1 HDLC 标准帧结构

(1)标志字段(F)

HDLC 指定采用 01111110 为标志序列,称为 F 标志。所有的帧必须以 F 标志开始和结束。接收设备不断地搜寻 F 标志,以实现帧同步,从而保证接收部分对后续字段的正确识别。另外,在帧与帧的空载期间,可以连续发送 F,用来作时间填充。

在一串数据比特中,有可能产生与标志字段的码型相同的比特组合。为了防止这种情况产生,保证对数据的透明传输,采取了比特填充技术。当采用比特填充技术时,在信码中连续 5 个“1”以后插入一个“0”;而在接收端,则去除 5 个“1”以后的“0”,恢复原来的数据序列,如图 2 所示。



图 2 比特填充

比特填充技术的采用排除了在信息流中出现的标志字段的的可能性,保证了对数据信息的透明传输。

当连续传输两帧时,前一个帧的结束标志字段

F 可以兼作后一个帧的起始标志字段。当暂时没有信息传送时,可以连续发送标志字段,使接收端可以一直保持与发送端同步。

(2)地址字段(A)

地址字段表示链路上站的地址。在使用不平衡方式传送数据时(采用 NRM 和 ARM),地址字段总是写入从站的地址;在使用平衡方式时(采用 ABM),地址字段总是写入应答站的地址。

地址字段的长度一般为 8 bit,最多可以表示 256 个站的地址。在许多系统中规定,地址字段为“11111111”时,定义为全站地址,即通知所有的接收站接收有关的命令帧并按其动作;全“0”比特为无站地址,用于测试数据链路的状态。因此有效地址共有 254 个之多,这对一般的多点链路是足够的。但考虑在某些情况下,例如使用分组无线网,用户可能很多,可使用扩充地址字段,以字节为单位扩充。在扩充时,每个地址字段的第 1 位用作扩充指示,即当第 1 位为“0”时,后续字节为扩充地址字段;当第 1 位为“1”时,后续字节不是扩充地址字段,地址字段到此为止。

(3)控制字段(C)

控制字段用来表示帧类型、帧编号以及命令、响应等。由图 1 可见,由于 C 字段的构成不同,可以把 HDLC 帧分为三种类型:信息帧、监控帧、无编号帧,分别简称 I 帧(information)、S 帧(supervisory)、U 帧(unnumbered)。在控制字段中,第 1 位是“0”为 I 帧,第 1、2 位是“10”为 S 帧,第 1、2 位是“11”为 U 帧,它们具体操作复杂,文中不予以具体介绍。

(4)信息字段(I)

信息字段内包含了用户的数据信息和来自上层的各种控制信息。在 I 帧和某些 U 帧中,具有该字段,它可以是任意长度的比特序列。在实际应用中,其长度由收发站的缓冲器的大小和线路的差错情况决定,但必须是 8 bit 的整数倍。

(5)帧校验字段(FCS)

帧校验序列用于对帧进行循环冗余校验,其校验范围从地址字段的第 1 比特到信息字段的最后一比特的序列,并且规定为了透明传输而插入的“0”不在校验范围内。

2 HDLC 协议的实现方法

解析和生成 HDLC 协议的方式主要有以下四种:

(1)专用HDLC协议接口芯片+微处理器^[4],如采用80C51单片机和MT8952B接口芯片实现了HDLC协议通信,但这种方式专用性强,难以通用于不同版本,缺乏灵活性;

(2)直接使用芯片集成的HDLC接口^[5],如三星的S3C4510B器件集成了两通道的HDLC控制器,但这些芯片集成的HDLC控制器都包含了较多的接口控制寄存器,需要配备相应的协处理器,其结构复杂,专业性太强,也会带来成本高等问题;

(3)现场可编程门阵列FPGA+接口芯片^[6],FPGA采用硬件技术处理信号,可通过软件反复编程,能够兼顾速度和灵活性,并能并行处理多路信号,但其成本略高,更适用于中小批量多通道HDLC通信产品中;

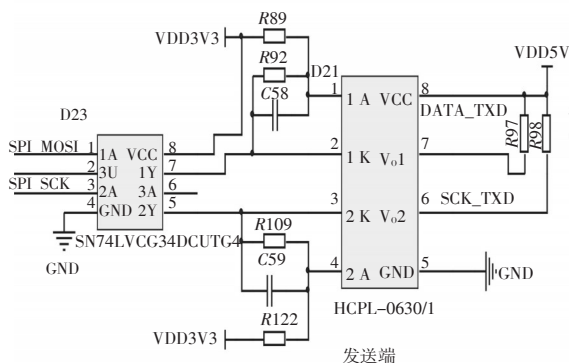
(4)软件编程^[1],由于微控制器或信号处理器是基于字的处理,而HDLC协议帧的解析和生成是面向比特的处理,这种方法会增加微处理器或信号处理器的负荷,执行速度慢,处理的通道数量有限,但软件编程方式更为灵活,仅需修改程序就可以适用于不同的HDLC应用,对于个别路数的低速信号处理更具优势。

由上所述可知,对于单路或双路采用特殊帧格式的HDLC协议的解析和生成,采用软件编程方式更为合适。

设计了一种半双工模式的HDLC协议接口,并通过软件编程,实现了特殊帧结构(帧格式如图3所示)的HDLC协议通信。

32 bit 标志	8 bit 地址	8 bit 控制	N×8 bit 信息	16 bit 校验	32 bit 标志
7E7E7E7E	A	C	I	FCS	7E7E7E7E

图3 特殊帧格式



3 硬件电路设计

整个电路由微处理器、总线驱动、光电隔离、差分接口和保护电路组成,组成框图如图4所示。

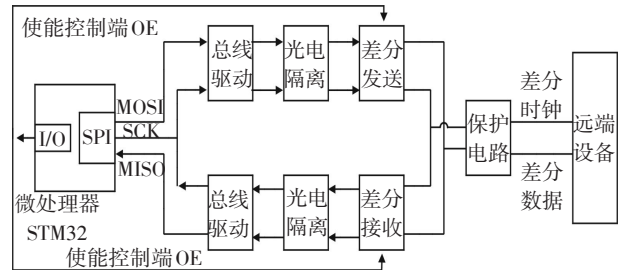


图4 电路组成框图

微控制器选用ST公司生产的STM32F103ZET6。该芯片具有以下特点^[7]:

- (1)使用高性能的ARM® Cortex™-M3 32位的RISC内核,工作频率最高为72 MHz;
- (2)内置512 K字节的闪存和64 K字节的SRAM;
- (3)外设配置丰富:最多可支持112路I/O端口、3个12位的ADC、4个通用16位定时器和2个PWM定时器、2个I²C接口、3个SPI接口、2个I²S接口、1个SDIO接口、5个USART接口、一个USB接口和一个CAN接口。

总线驱动选用TI公司生产的SN74LVC3G34,该芯片体积小,具有3路驱动端口,最大输出电流±50 mA,最大输出时延4.1 ns,输入电压范围1.65~5.5 V^[8]。

光电隔离选用Fairchild公司生产的HC-PL0630,最小输入电流5 mA,最高速率为10 Mbit/s,隔离电压为1 000 V^[9]。

总线驱动和光电隔离电路原理图如图5所示。

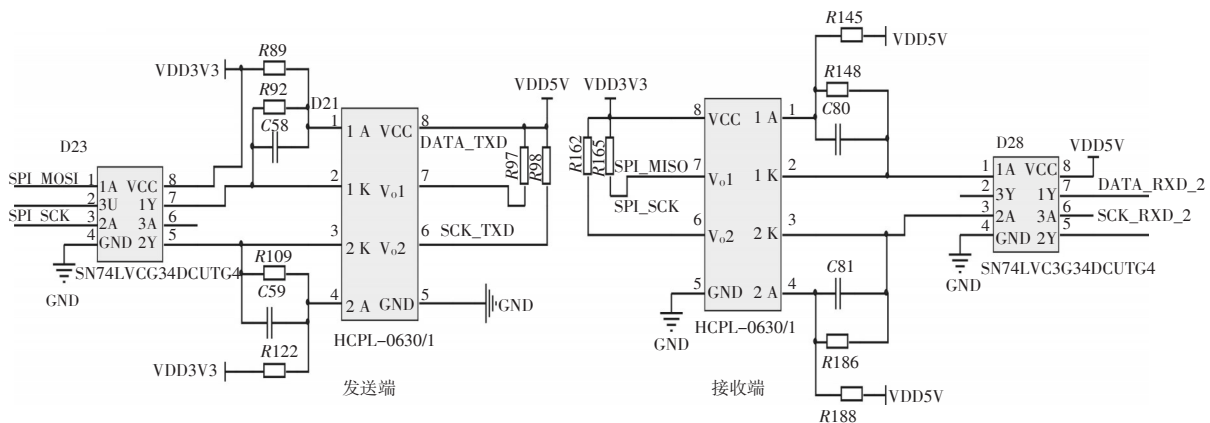


图5 总线驱动和光电隔离电路

差分发送选用NS公司生产的DS96F174MW/883,该芯片符合EIA-485和EIA-422标准,4路差分发送接口,2个使能控制端口^[10];差分接收选用NS公司

生产的DS96F175MW/883,该芯片符合EIA-485、EIA-422和EIA-423标准,4路差分接收接口,2个使能控制端口^[11]。差分发送和接收电路原理图如图6所示。

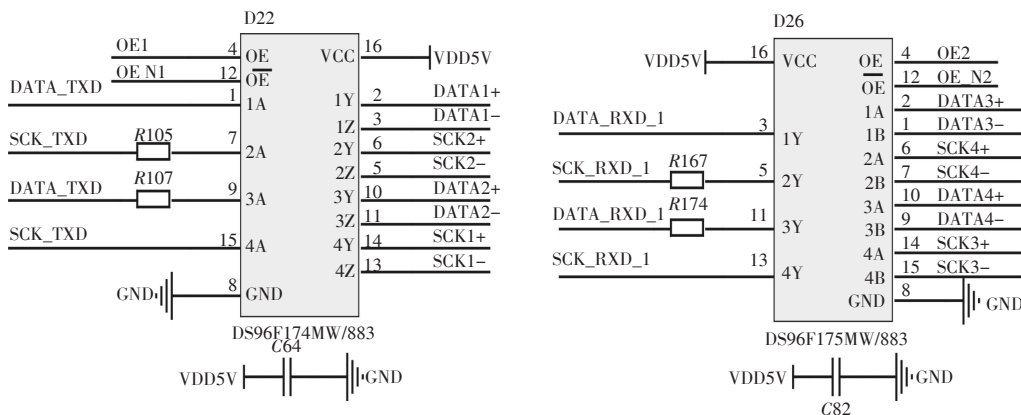


图6 差分发送和接收电路

保护电路由二级管和阻容器件组成,电路原理图如图7所示。

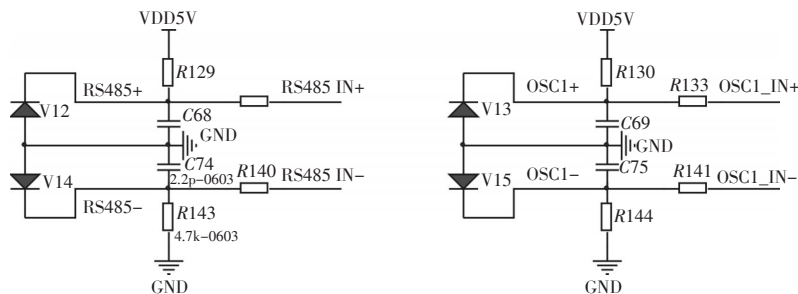


图7 保护电路

4 软件程序设计

在Keil开发环境下,使用C语言编程,实现HDLC数据帧的解析和生成,程序流程图如图8所示。

系统上电后,首先对使能控制端口OE、SPI接口、时钟、定时器等进行初始化,待初始化完成后,系统转入接收状态(默认状态,从机模式),等待远端设备发送指令。当接收到指令后,进入SPI中断服务程序,由软件判断同步帧头是否正确。如同步帧头正确,将接收的数据存入数据队列(FIFO),待数据接收完毕后,解析数据(去掉插入的“0”位);如同步帧头错误,丢弃该无效指令。

数据解析完成后,首先判断同步帧尾、地址字、控制字、CRC校验码是否正确。仅当同步帧尾、地址字、控制字和CRC校验码都正确时,系统方才响

应指令,形成应答数据;反之,丢弃该无效指令。

由软件将应答数据按位处理(每5个“1”后插入“0”),生成应答数据流(含同步帧头和帧尾)后,系统转入发送状态(主机模式)发送应答数据,待数据发送完成后,系统转入接收状态(从机模式)。

5 测试

为验证接口设计的合理、正确性,使用Tektronix公司生产的DPO 4104B示波器和TDP3500探头对本接口设计方案进行了测试。

测试时,首先由软件设置数据在时钟的上升沿有效,通信频率1 Mbit/s,再由远端设备发送测试数据(“FF FF FF FF”)至自研板卡,自研板卡将接收的数据解析后将解析的数据再发送给远端设备。自研板卡生成的波形如图9所示。

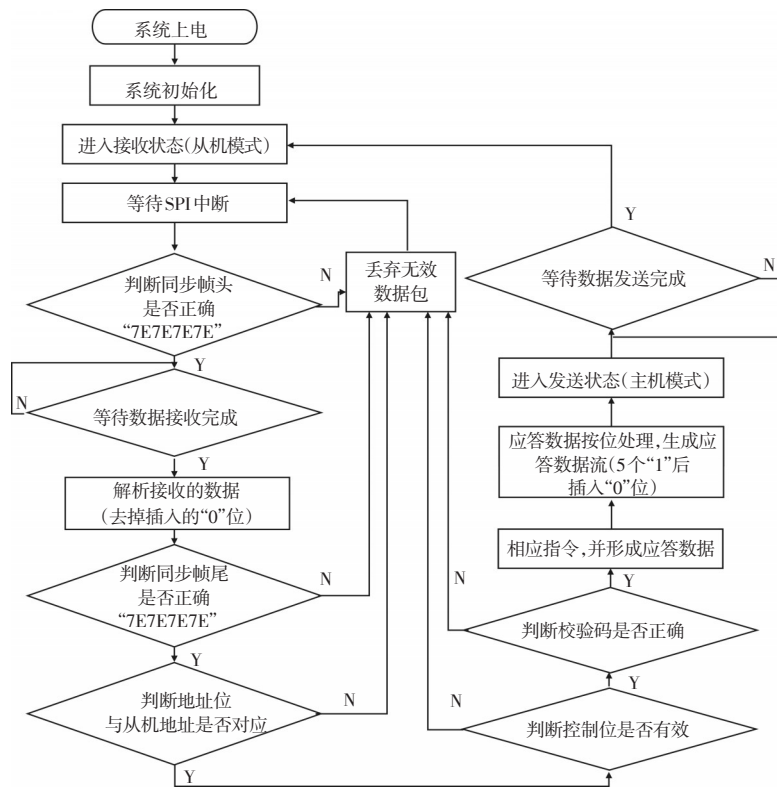
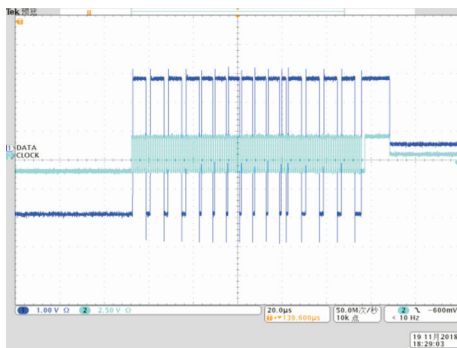
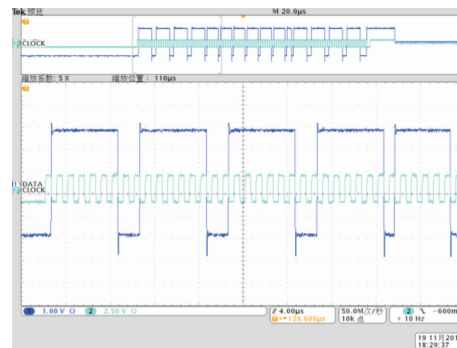


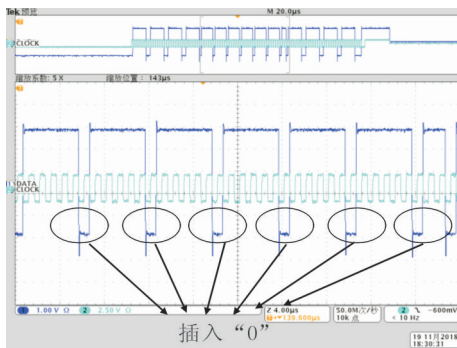
图8 程序流程图



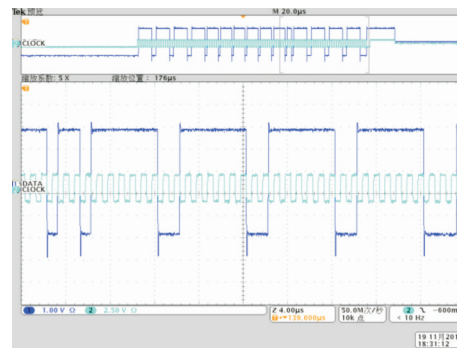
数据帧总览



同步帧头(7E7E7E7E)



数据



同步帧尾(7E7E7E7E)

图9 示波器实测波形

(下转第76页)

synchronization of a complex network with different nodes[J]. Acta Physica Sinica, 2012, 61(9): 1-13.

[30] Sarpturk S Z, Istefanopoulos Y, Kaynak O. On the stability

of discrete-time sliding mode control systems[J]. IEEE Transactions on Automatic Control, 1987, 32 (10): 930-937.

(上接第40页)

and compressive sensing[J]. Rice University, 2009, 9.

[4] 王强, 李佳. 压缩感知中确定性测量矩阵构造算法综述[J]. 电子学报, 2013, 10(10): 2041-2050.

[5] 吴赞. 压缩感知测量矩阵的研究[D]. 西安: 西安电子科技大学, 2012.

[6] 华翌辰. 基于压缩感知的鬼成像研究[D]. 南京: 南京理工大学, 2014.

[7] Tropp J, Gilbert A C. Signal recovery from partial information via orthogonal matching pursuit[J]. 2005.

[8] Tropp J A, Gilbert A C. Signal recovery from random mea-

surements via orthogonal matching pursuit [J]. Information Theory, IEEE Transactions on, 2007, 53(12): 4655-4666.

[9] Donoho D L, Tsaig Y, Drori I, et al. Sparse solution of underdetermined systems of linear equations by stage wise orthogonal matching pursuit[J]. Information Theory, IEEE Transactions on, 2012, 58(2): 1094-1121.

[10] Needell D, Vershynin R. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit[J]. Foundations of computational mathematics, 2009, 9(3): 317-334.

(上接第48页)

由示波器波形可以看出, 自研板卡的数据信号以“7E7E7E7E”开始, 以“7E7E7E7E”结尾, 与图3所示的特殊数据帧格式一致; 地址、控制、信息和校验码字段的数据序列每隔5个“1”插入了1个“0”, 符合HDLC协议的比特填充要求; 发出的数据信号和时钟信号同步产生和结束, 符合串行同步通信的要求。

分析数据序列波形, 可得出自研板卡发出的原始数据为“FF FF FF FF”, 与远端设备发来的测试数据一致。

综上所述, 自研板卡可正确解析和生成HDLC协议数据, 本接口设计合理、正确、可行。

6 结论

采用STM32F103微处理器和差分收发芯片架构实现HDLC数据帧的解析与生成, 所需的硬件配置资源少, 利于实现研发成本控制和功能集成; 同步帧头和同步帧尾数量、数据触发模式(即时钟上升沿有效和时钟下降沿有效)和数据通信速率可通过软件进行配置, 使用灵活, 适应性强; 在此接口设计方案基础上稍加修改即可实现全双工模式的HDLC协议通信, 但由于STM32F103微处理器的处理能力有限, 通信速率不大于1 Mbit/s。

综上所述, 本方法在成本控制较高、通信速率小于1 Mbit/s的单路或双路采用HDLC协议通信的场合, 具有很高的应用价值。

参考文献

- [1] 张伟. 基于FPGA的HDLC协议控制系统实现[J]. 指挥信息系统与技术, 2011, 2(2): 48-52.
- [2] 宋飞, 李志蜀. HDLC协议在FPGA通信系统中的实现[J]. 计算机应用, 2009, 29(4): 1092-1094.
- [3] 孟静, 王冬青, 丁明伟, 等. HDLC/UART通信网关的设计[J]. 自动化技术与应用, 2013(4): 21-24, 55.
- [4] 闫肖鹏, 余义的, 张洪昶. 一种基于单片机的HDLC通信卡设计[J]. 自动化技术与应用, 2016, 35(10): 156-158.
- [5] 刘鹏, 金野. 基于S3C4510B和 μ Clinux的HDLC接口设计实现[J]. 微计算机信息, 2007, 23(2-2): 1-3.
- [6] 钱黎明, 刘富发, 魏敬和. 支持PCI总线的HDLC协议专用控制器设计[J]. 中国电子科学研究院学报, 2014, 9(6): 648-652.
- [7] ST公司. STM32F103xCDE数据手册[M]. 2009.
- [8] TI公司. SN74LVC3G34数据手册[M]. 2013.
- [9] Fairchild公司. HCPL 06xx系列数据手册[M]. 2006.
- [10] NS公司. DS96F172/DS96F174数据手册[M]. 2000.
- [11] NS公司. DS96F172/DS96F174数据手册[M]. 2000.