doi:10.3969/j.issn.1671-637X.2017.03.022

基于USB的软件综合安全模块设计及应用

章超超, 陈元林, 卢 娜, 安博文
(上海海事大学信息工程学院,上海 201306)

摘 要:设计的软件安全模块是一个外置的硬件模块,提供版权保护、核心数据加密、稳定运行监控等功能。外置安全模块采用USB与PC主机连接,通过将安全模块内的微控制器虚拟为USB大容量存储设备,实现了免驱动的便携数据通道。设计了一种基于报文的分层通信协议栈作为安全模块与PC的通信规约,实现安全可靠的数据通信,并提供通用的调用接口。以电缆载流量分析软件为应用实例,说明安全模块的使用方法。

关键词:数据加密;数据通信;USB;版权保护;通信规约;载流量分析
中图分类号:TP309 **文献标志码**:A **文章编号**:1671-637X(2017)03-0093-05

Design and Application of a Comprehensive Software Security Module Based on USB

ZHANG Chao-chao, CHEN Yuan-lin, LU Na, AN Bo-wen
(Information Engineering College, Shanghai Maritime University, Shanghai 201306, China)

Abstract: A software security module is designed, which is an external hardware module, and can provide such functions as copyright authentication, kernel data encryption, stable operation monitoring and so on. The external module uses USB to connect with host PC, and a free driver portable data channel is achieved by taking the microcontroller of security module as a virtual USB mass storage device. A communication protocol is designed for use between the security module and PC based on packets layered communication protocol stack, and thus to implement safe and reliable data communication. The usage of the security module is illustrated by taking the software for cable ampacity analysis as an application example.

Key words: data encryption; data communication; USB; copyright protection; communication protocol; ampacity analysis

0 引言

软件的版权保护、核心数据加密、稳定运行监控功能是开发人员亟待解决的重要问题。根据加密方式,目前软件版权保护及数据加密技术分为软加密和硬加密[1]。软加密不依赖特定硬件,但软件实现手段不利于授权管理[2];硬加密依赖特定的硬件,通过硬件和软件相结合的方式来实现加密保护,其优势在于利用硬件的不可复制性达到较高强度的加密保护[3]。目前市面上的加密狗属于硬加密的一种方式,但是市场上的加密狗不能同时满足上述3个功能需求。本文设计的综合安全模块采用通用USB传输接口,具备认证、存储、运算、故障恢复、继电输出等功能。重点是设计一套通信协议,来高效地实现上述功能,同时易于二次开发和应用,提高稳定性。

收稿日期:2016-03-31 **修回日期**:2016-04-18
作者简介:章超超(1990—),男,安徽芜湖人,硕士生,研究方向为嵌入式应用、光纤通信。

1 系统概述

为方便系统描述,本文把运行应用软件的PC机称为上位机,外置安全模块称为下位机,两者结合组成一个包含软硬件的综合安全系统。

综合安全系统包括上位机通信协议栈、下位机通信协议栈、下位机硬件模块、下位机伺服程序以及相关控制装置,其中,下位机硬件模块以微控制芯片为核心。综合安全系统则具有版权保护、透明传输、稳定运行监控等功能。

1.1 版权保护功能

版权保护功能包括:下位机监测上位机的运行次数及单次运行时间,限定软件用户的使用权限;上位机监测下位机是否已与软件连接,防止软件运行时未接

入配套的下位机。

1）合法授权情况下上位机与下位机的通信：上位机定时产生随机数据包并加密，将密文数据包发送给下位机，下位机收到后对其解密，再将解密数据包发送给上位机，上位机比对随机数据包和解密数据包，以此来核准身份。上位机启动时，向下位机发送认证数据包，下位机开始记录上位机本次运行时间及上位机运行次数。

2）非法授权情况下上位机与下位机的通信：超过一定时间，上位机没有收到解密数据包或者比对不一致时，上位机软件启动自锁功能。运行次数超过限定次数或本次运行时间达到单次的运行限定时间时，下位机发送控制数据包，控制上位机自锁功能。

1.2 数据加密功能

1）加密传输：上位机与下位机之间传输的数据均采用 AES256 加密算法加密后进行传输，实现透明传输的功能。

2）核心数据加密：系统软件中一些核心参数等数据，将其加密后存放在下位机中，当上位机需要时再进行读取以保证核心数据的安全。

1.3 稳定运行监控功能

稳定运行是系统软件的基本要求，外置稳定模块具有保障软件的稳定运行的功能。图 1 为其整体框图。

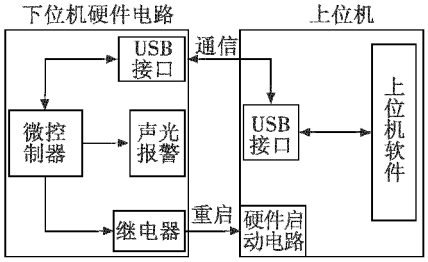

图 1 外置稳定模块整体框图

Fig. 1 The external stable module

图 2 为控制逻辑程序框图，图中，$t1$，$t2$，$t3$，$t4$，n，m，p 均为根据用户需求设置的变量。

重启条件 I：超过 $t1$ 分钟未收到上位机交互数据包，硬件重启软件。

重启条件 II：检测到上位机在 $t2$ 分钟之内“软重启”n 次，硬件重启上位机。

在连续 $t2$ 分钟内，当连续 n 次出现重启条件 I 时，不再硬件重启上位机，关闭软件系统功能，开始声光报警。

当在重启条件 II 连续出现 m 次时，不再硬件重启上位机，关闭上位机软件系统功能，开始声光报警。

稳定工作时，上位机与下位机的通信：每间隔 $t3$ 分钟，上位机会给下位机硬件模块发送交互数据告知其正常运行。

非稳定工作时，上位机与下位机的通信：1）当上位机发生故障时，下位机由于没有接收到数据会通过硬件启动电路重启上位机，保障系统正常运行。2）在一定时间内重启次数过多，系统会自动声光报警提醒工作人员检查设备。

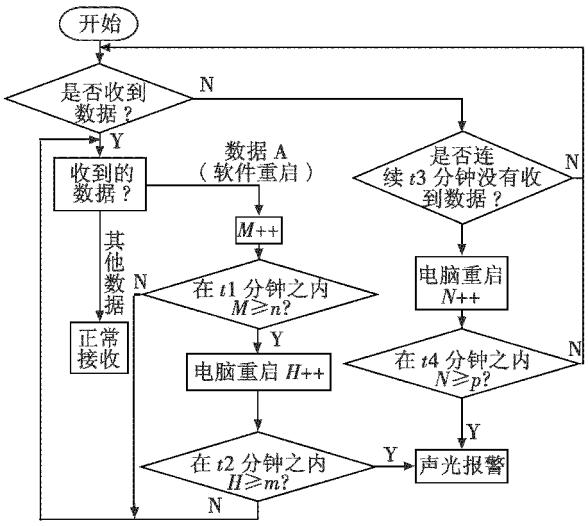

图 2 控制逻辑程序框图

Fig. 2 The control logic program

2 硬件设计

2.1 微控制芯片选型

由于系统需要具备 USB 接口、大容量 SRAM 等特性，因此选用微控制芯片 STM32F103。

2.2 硬件电路构成

硬件电路以 STM32F103 微控制器为控制核心，采用 ORCAD 软件进行原理图设计和 PCB 绘制[4]。电路采用 USB 与上位机进行通信，通过继电器控制重启服务器实现硬件看门狗功能。

核心电路板主要由电源模块、硬件启动电路、USB 接口、RS232 接口、STM32F103VCT6 微控制器系统等组成。电路板直接由 PC 机的 USB 接口供电。

硬件电路的具体组成如图 3 所示。

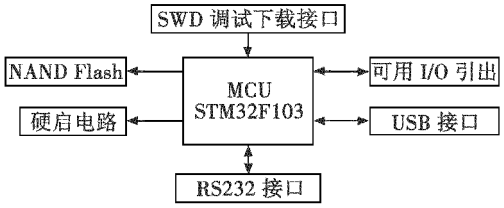

图 3 硬件电路组成框图

Fig. 3 Hardware circuit

以下主要描述一下硬件启动电路，它是控制上位机及服务器重启的重要控制电路。硬件启动电路如图 4 所示。图 4 中，微控制器通过控制 9013 三极管基极

的电平高低来控制三极管是否导通[5]。当基极收到高电平时，三极管导通，继电器中的线圈通过电流，触点簧片合上，继电器处于“关”的状态。同理，当基极收到低电平时继电器断开。通过这样的“开关”控制电脑的重启，保障上位机系统的稳定运行。每一个继电器都有一个保护二极管，防止断开继电器后线圈的自感电压烧坏器件。

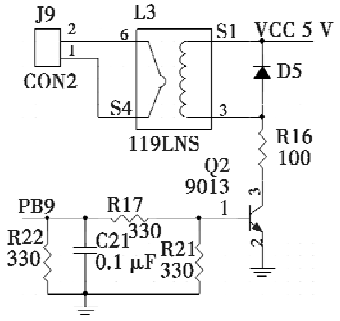

图 4　硬件启动电路

Fig. 4　Hardware start-up circuit

3　数据通道设计

上位机与下位机采用 USB 进行通信，需要编写 USB 设备驱动程序。由于 USB 采集设备的驱动在不同操作系统中会出现不兼容现象，所以设计免驱动的便携数据通道，将 STM32 内置存储器虚拟成 U 盘，保证安全模块的适用性和通用性。虚拟的 U 盘作为下位机与上位机数据交互的公共空间，采用 Windows API 函数对虚拟 U 盘进行读写数据操作[6]，上位机与下位机按照采集设备通信协议实现通信过程。

本系统采用的控制芯片的内置 SRAM 内存为 48 kB，前 16 kB 用于处理器处理程序存储变量，剩余 32 kB 作为虚拟 U 盘。

使用 Win hex 软件查看 U 盘物理地址处数据，经过多次试验得知：1）上位机向虚拟 U 盘读写数据字节，单次读写字节数只能是 2 kB 的整数倍；2）无法向 4 kB 的起始扇区中写数据，只能向剩余 28 kB 区域中读写数据。

根据功能用途，将 28 kB 区域分成 5 个区域，分别为 A 域、B 域、C 域、D 域和 E 域。内置 SRAM 区域划分情况如图 5 所示。

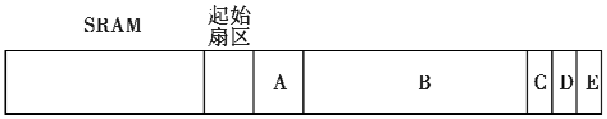

图 5　区域划分示意图

Fig. 5　Zoning schematic

虚拟 U 盘中数据读写分区如表 1 所示。

1）A 域是上位机向下位机发送数据包的存储区域，因为上位机向下位机发送的数据为请求或认证数据，字节数较少，所以 A 域分配 4 kB。

2）B 域为下位机向上位机发送数据包的存储区域，发送加密的核心数据较多，所以 B 域分配了 18 kB。

3）C 域为上位机在 A 域中写完数据后，再发送结束数据包的存储区域。

4）D 域为下位机在 B 域中写完数据后，再发送结束数据包的存储区域。

5）E 域为下位机主动向上位机报告状态信息的存储区域，C，D，E 域中数据字节长度很小，设置为单次读写字节数最少的 2 kB。

表 1　虚拟 U 盘中数据读写分区

Table 1　Partition read-write areas in the U disk

虚拟 U 盘	A 域	B 域	C 域	D 域	E 域
内存大小/kB	4	18	2	2	2
偏移地址	0x1000 ~ 0x1FFF	0x2000 ~ 0x67FF	0x6800 ~ 0x6FFF	0x7000 ~ 0x77FF	0x7800 ~ 0x7FFF

4　软件设计

综合安全系统软件要实现的功能包括认证、存储、运算、监控和继电控制等。

4.1　软件框架设计

综合安全系统软件设计分为上位机软件设计和下位机伺服软件设计。上位机负责调用 Windows API 函数通信及实现上位机通信协议工作，下位机伺服软件负责虚拟 U 盘及实现下位机通信协议的工作。

4.2　通信协议设计

4.2.1　通信协议分层结构设计

通信协议设计过程参考 OSI 网络模型，通信协议采用层次结构实现，各层采用统一接口实现通信[7]。按照功能的不同分为 4 层：即硬件通信层、加密传输层、传输控制层和应用层。图 6 所示为通信协议分层结构模型。

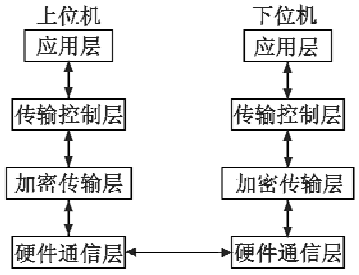

图 6　通信协议分层结构模型图

Fig. 6　Communication protocol layered model

通信协议各层主要功能如下。

1）硬件通信层。主要功能是接收上层协议传输层传输下来的数据包，硬件通信层对数据包做判断（判

断是否分包),将不足2048字节整数倍的数据包拼接成2048整数倍的数据包,并发送到指定地址,在虚拟U盘中实现读写数据工作。该层的工作包括调用Windows API底层读写函数直接在虚拟U盘内存中读写数据,同时预留函数接口,来接收加密传输层传递下来的密文数据包。

2)加密传输层。加密传输层采用AES256加密算法,将上位机和下位机之间传输的数据进行加密,保证数据的安全性。

3)传输控制层。传输控制层的主要功能是完成数据封包/解析、校验差错控制、传输控制工作。

4)应用层。应用层的工作是把要传输的数据提交到传输控制层和提供通用的调用接口。

4.2.2 逻辑控制

下位机不会主动发送数据,上位机是所有数据和命令发起者。逻辑控制如下:

1)上位机向A域中发送命令数据包,发送完成后,再向C域中发送结束标识数据包;

2)下位机定时轮询读取C域,查看是否有发送结束标识数据包,如果读取到发送结束标识数据包,则下位机读取A域中命令数据包;

3)下位机响应上位机的命令,向B域中发送响应数据包,发送完成后,再向D域发送结束标识数据包;

4)上位机同下位机一样,定时轮询读取B域数据,查看下位机是否响应,如果有响应数据包,则读出响应数据包,进行后续操作。

4.2.3 数据包的格式设计

数据包中包含5个组成部分,即包标识符字段、数据字段、扩展字段、校验字段和包结束符字段[8-9]。本文设计的通信协议数据包格式如表2所示。

表2 通信协议数据包格式

Table 2 Communication protocol packet format B

包标识符字段	数据字段	扩展字段	校验字段	结束字段
5	视情况而定	494	2	1

数据字段包含5个组成部分,即功能码、包序号、数据长度、真实数据长度和真实数据,其中数据长度包括真实数据和扩展字节数据。数据字段的格式如表3所示。

表3 数据字段格式

Table 3 Data numeric field format B

功能码	包序号	数据长度	真实数据长度	真实数据
2	4	2	2	视情况而定

数据包的格式说明如下。

1)标识符字段。表示本数据包属于该设备通信协议,作为一个数据包的包头,同时作为是否接收此数据包的标志,为了防止与正文数据混淆,在本系统中约定为AABBCC789FH,占5个字节,这样就能保证与正文数据相同的概率降至$1/(2^{40})$。

2)数据字段。

① 功能码。数据包所执行功能的标志码,如"0000H"代表上位机应用程序要求下位机开始计算本次软件运行时间和软件运行次数的功能码。占2个字节,可供使用的功能码有65536个,可以根据实际应用需求进行扩充。

② 包序号。因为虚拟U盘空间比较小,如果发送的数据包超过了U盘的区域内存大小,就需要分成几个数据包发送。占4个字节,前2个字节表示发送数据的分包数,后2个字节表示当前发送的包序号。

③ 数据长度。表示本次发送的整个数据包的字节长度,包括包标识符和包结束符,便于接收方校验。占2个字节。

④ 真实数据长度。表示后面的真实数据所占字节长度,与数据长度相比,不包含扩展字节、包标识符和包结尾符。占2个字节。

⑤ 真实数据。表示一个数据包所要发送的数据内容。

3)扩展字段。保留字段,方便以后可扩展功能,同时保证数据包中真实数据前后共占512字节,由于对U盘读写操作是以扇区为单位,一个扇区为512字节,方便对真实数据的读写操作。

4)校验字段。这部分内容是对前面内容的校验码,接收方通过判断这部分的内容来确定传送数据是否正确,如果不正确,则发送请求重发数据包,请求发送方重传。在本系统中,数据包校验字段使用的是16位循环冗余校验CRC,校验的范围为校验字段之前的所有字节[10],占2个字节。

5)包结束符字段。表示此数据包的结束,约定为16H,占1个字节。

4.2.4 数据包的封包与解析

数据发送方在发送数据前,需要按照通信协议的规则,将要发送的数据打包好才能发送,接收方接收到数据包后,需要对数据包进行解析,抽取出数据包中的真实数据。

4.2.5 差错控制

校验字段放在包结束符字段之前、扩展字段之后。上位机与下位机数据交互过程中检测纠错采用循环冗余校验方法。

数据接收方接收到发送方的数据后进行差错校验,若不正确,则发送请求重发数据包,请求数据发送方重传数据包。

4.3 下位机伺服程序设计

系统采用 USB 接口通信，图 7 为 USB 通信程序框图。下位机伺服程序包括将 SRAM 虚拟成 U 盘、启动控制程序、实现下位机通信协议以及运算存储等功能。

程序通过 USB 中断完成虚拟 U 盘过程。U 盘虚拟完成后继续等待定时器中断，当中断发生后，下位机伺服程序读取 U 盘特定地址处是否有来自上位机的数据包，若没有则继续等待中断，若有则响应上位机的命令。

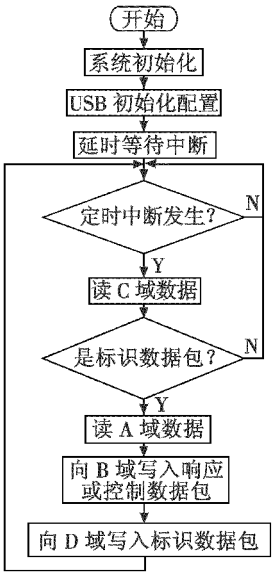

图 7　USB 通信程序框图

Fig. 7　Flow chart of USB communication program

5 系统应用

本文以电缆载流量分析软件为应用实例，介绍安全模块的使用方法。电缆载流量分析中核心数据如光纤温度、铜芯温度及其之间的映射关系，这些核心数据加密后存放在下位机硬件模块中，当载流量分析软件需要核心数据时从下位机读取。

当载流量分析软件开始运行时，调用安全模块上位机的接口，向下位机发送认证数据包、随机密文数据包及交互数据包，下位机能定时收到交互数据包并且上位机能定时收到下位机解密数据包，即实现了下位机对载流量分析软件的版权认证保护和稳定运行模块功能。图 8 为数据传输示意图。

当载流量分析软件在需要调用核心数据时，向下位机发送请求核心数据包，下位机收到请求数据包后调出加密的核心数据并发送给上位机，实现核心数据加密功能。

载流量分析软件系统中，数据加密过程、数据存储转移过程比较耗时，实际测得速率保持在1.1 MB/s，即 1 min 可以采集 66 MB 的数据，载流量分析软件每 1 min 采集 1 次数据，每 3 min 分析 1 次数据，上位机与下位机之间通信与数据交互的数据量较少，实际速率能满足系统软件实时性要求。

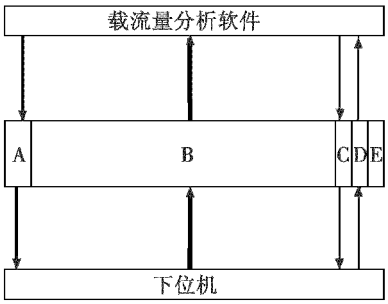

图 8　数据传输示意图

Fig. 8　Data transmission schematic

6 结论

综合安全模块采用 USB 与 PC 主机连接，将 STM32 微控制器内置 SRAM 虚拟成 U 盘，设计了免驱动的便携式数据通道，保证了安全模块的实用性和通用性，然后设计了上位机与下位机通信规约，实现了安全可靠的数据通信。本文以载流量分析软件为应用实例，实现了对电缆载流量分析软件的版权保护、核心数据加密及稳定运行监控功能，取得了很好的使用效果。

参考文献

[1] 魏光村，孙忠林，徐燕妮. 软件加密技术研究[J]. 福建电脑，2006(9)：44-45.

[2] 任克强，刘晖. 单片机系统硬件及软件加密技术[J]. 电子设计应用，2003(7)：61-63.

[3] 王海春，李均，邓珊. 基于混沌加密的 RFID 认证协议设计[J]. 数字技术与应用，2015(11)：206-207，209.

[4] 信侃. 基于 Cadence 软件的高速 AD 电路设计与仿真[D]. 哈尔滨：哈尔滨工业大学，2008.

[5] 童诗白，华成英. 模拟电子技术基础[M]. 北京：高等教育出版社，2006.

[6] 范文庆，周彬彬，安靖. 精通 Windows API：函数、接口、编程实例[M]. 北京：人民邮电出版社，2009.

[7] 谢希仁. 计算机网络[M]. 北京：电子工业出版社，2008.

[8] 方旭. 基于 STM32 处理器和 PC 主机的 USB 通信协议的实现[D]. 济南：山东大学，2009.

[9] 刘爱东，张永强，杨健，等. USB 设备互连通信协议设计[J]. 电光与控制，2011，18(1)：69-72.

[10] 胡方家，周双娥，曾军. 基于可靠度的循环冗余校验算法[J]. 计算机应用，2015，35(3)：629-632.