

基于 STM32 嵌入式的无线通信远程数据传输控制系统设计

史磊

(陕西机电职业技术学院 电子与信息学院, 陕西 宝鸡 721001)

摘要: 无线通信远程数据传输控制系统对提高数据传输稳定性起到积极作用, 大量数据传输任务给系统的控制功能带来巨大挑战, 为此利用 STM32 嵌入式技术实现无线通信远程数据传输控制系统的优化设计; 将 STM32 芯片以嵌入式的方式安装到数据传输控制器中, 改装传输数据处理器, 加设时钟发生器; 通过系统电路的调整与连接, 完成数据传输控制硬件系统的设计; 在硬件设备的支持下, 采集并处理无线通信数据资源, 设置远程数据传输控制协议; 在协议约束下, 根据信道特征合理分配远程传输数据资源, 最终从传输速率、传输流量等方面, 实现系统的无线通信远程数据传输控制功能; 通过系统测试实验得出结论: 在优化设计系统的控制下, 远程数据传输速率误差、丢包率控制和传输拥塞总时长均低于预设值, 且吞吐率高于 88%, 由此说明优化设计系统的控制功能和运行性能均满足设计与应用要求。

关键词: STM32 嵌入式; 无线通信; 远程数据传输; 传输控制系统

Design of Embedded Wireless Communication Remote Data Transmission Control System Based on STM32

SHI Lei

(School of Electronics and Information, Shanxi Institute of Mechatronic Technology, Baoji 721001, China)

Abstract: The wireless communication remote data transmission control system plays a positive role in improving the stability of data transmission. A large number of data transmission tasks bring great challenges to the control function of the system. For this reason, the STM32 embedded technology is used to realize the optimal design of the wireless communication remote data transmission control system. The STM32 chip is installed into the data transmission controller in an embedded manner, the transmission data processor is modified, and a clock generator is added. Through the adjustment and connection of the system circuit, the design of data transmission control hardware system is completed. With the support of hardware equipment, the wireless communication data resources is collected and processed, and the remote data transmission control protocols is set up. Under the constraints of protocol, the remote transmission data resources are allocated reasonably according to the channel characteristics, finally the wireless communication remote data transmission control function of the system is realized from the aspects of transmission rate and transmission flow. Through the system test experiment, it is concluded that, under the control of the optimized design system, the remote data transmission rate error, the packet loss rate control and the total duration of transmission congestion are all lower than the preset value, and the throughput rate is higher than 88%, which shows that the control function and operation performance of the optimized design system meet the design and application requirements.

Keywords: STM32 embedded; wireless communication; remote data transmission; transmission control system

0 引言

无线通信是指多个节点间不通过导体传播而实现远距离传输通讯的技术, 与有线通信相比, 无线通信方式具有结构简单、传输距离远等优点, 不会受到光缆、导线等硬件设备的约束, 能够在很大程度节约硬件成本, 完成有线通信设备无法完成的数据传输任务, 因此网络通信信息的获取和传输逐步从有线过渡到无线, 并应用到诸多工作领域中^[1]。一般来讲, 无线通信主要利用无线电波作为数据

传输的媒介, 将发送端与接收端设备的数据信息调制到相同的载波频率上, 从而实现数据的定向传输。

然而从当前无线通信远程数据传输的应用情况来看, 受到通信传输环境的影响, 会出现传输中断、传输数据丢失等现象, 随着使用用户数量的增加, 无线通信服务也容易出现传输拥塞等问题, 不仅会影响无线通信的运行效率, 还有可能造成信息泄露等严重后果。为了最大程度地保证无线通信远程数据传输的安全稳定性以及时效性, 有必要设计并应用相应的数据传输控制系统。

收稿日期: 2022-04-06; 修回日期: 2022-05-09。

基金项目: 2022 年度陕西省教育厅专项科研计划项目(22JK0026)。

作者简介: 史磊 (1991-), 女, 陕西延安人, 硕士, 讲师, 主要从事通信与信息系统方向的研究。

引用格式: 史磊. 基于 STM32 嵌入式的无线通信远程数据传输控制系统设计[J]. 计算机测量与控制, 2022, 30(11): 111-115, 146.

现有的数据传输控制系统存在明显的控制功能不佳、运行性能差的问题，主要体现在数据传输丢失量大、拥塞概率高、吞吐量差等方面，为了解决上述传统控制系统存在的问题，引入 STM32 嵌入式概念。将 STM32 以嵌入式的方式应用到无线通信远程数据传输控制系统中，改装传输数据处理器，加设时钟发生器，提升无线通信远程数据传输控制效果。通过系统电路的连接实现数据传输控制硬件系统的设计，采集并处理无线通信数据资源，设置远程数据传输控制协议并通过信道特征分配远程传输数据资源，提高系统的传输控制功能与运行性能。

1 无线通信远程数据传输控制硬件系统设计

无线通信远程数据传输控制系统以无线信息远程控制为主要手段，通过对传输信道、传输数据等主体的控制，完成对数据传输流量以及传输速度的控制。此次无线通信远程数据传输控制系统分别从硬件、软件功能程序两个方面进行分析优化设计，其中硬件部分的作用是为数据库以及软件功能的运行提供支持，硬件系统的基本框架如图 1 所示。

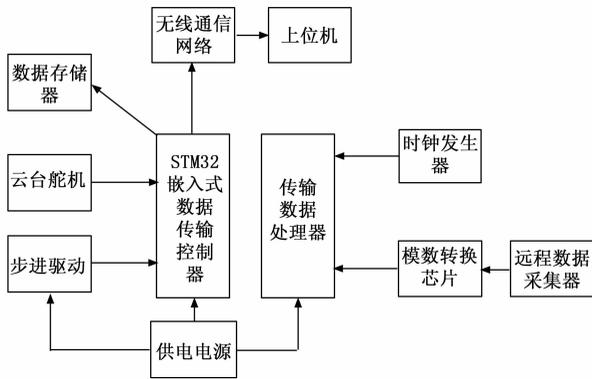


图 1 无线通信远程数据传输控制硬件系统框图

在传统传输控制硬件系统的基础上，利用 STM32 对控制器以及微处理器等硬件设备进行优化，同时对部分硬件元件和连接电路进行改装和调整，提高硬件设备的运行效率。

1.1 STM32 嵌入式数据传输控制器

STM32 嵌入式芯片基于 Cortex-m3 核心，结合了多种工业标准的高性能接口，并与 STM32 系列产品的管脚和软件兼容，使得 STM32 适配于多种应用程序^[2]。为满足系统的互联功能和实时性能，选择 STM32107 型号的芯片进行传输控制器优化。图 2 表示的是 STM32107 嵌入式芯片的内部结构。

图 2 所示的 STM32 控制芯片，包括 256KB 的 Flash 存储器，6 个 USART 接口、3 个 16 位 DAC 转换器、12 个 DMA 通道、以太网 IEEE1588 接口等。STM32 嵌入式芯片相对于传统的嵌入式控制系统，采用 IEEE1588 协议，内置双 CAN 总线具有良好的网络连接能力^[3]。STM32 芯片采用了嵌入式的方式连接到传统控制器上，实现一种经济的

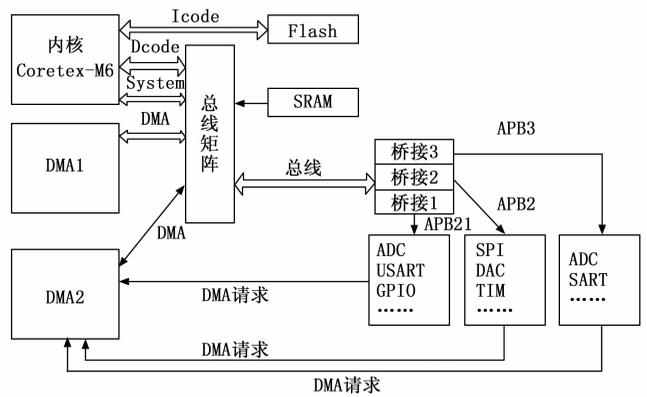


图 2 STM32 嵌入式芯片结构图

网络互联。在硬件方面，它支持 IEEE1588 精确计时协议，能够在网络上进行准确的时钟同步，为更多的嵌入式应用程序预留 CPU 空间。STM32 中采用的双 CAN20B 模块是一种更为便捷的 CAN 桥接方式。

1.2 传输数据处理器

优化设计的无线通信远程数据传输控制系统采用 ARM7 型信号处理装置进行了优化设计，包括核心、总线、外部接口、调试等部分。内核是执行运算、内存存取、周界面控制、中断控制等功能的核心。ARM7 处理器采用了一个简单的二极管处理命令，一个命令被内核同时解码，而另一个命令又被读取到了内存中，所以程序计数器总是指向下一个命令，而非目前被执行的命令。ARM7 核心包含 38 个寄存器，32 个通用寄存器，6 个寄存器不能用于使用者和程序^[4]。目前可使用的暂存器组合需要通过处理器状态、运行、闲置和运行方式来确定。

1.3 时钟发生器

硬件系统中的所有元件均为时序电路，需要一个时钟信号才能工作，为此在硬件设备中内置一个型号为 LPC2210 晶体振荡器，LPC2210 晶体振荡器的工作模式可以分为从属模式和振荡模式两种^[5]，从属模式下的工作原理如图 3 所示。

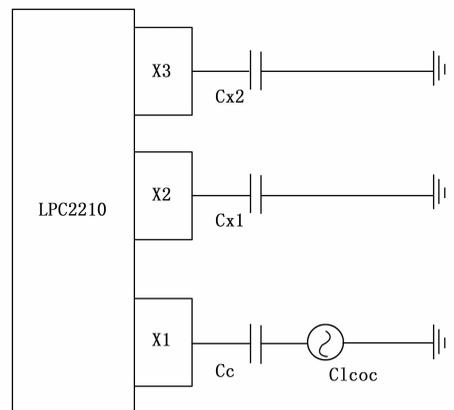


图 3 时钟发生器工作原理图

输入时钟信号被耦合到一个大于 100 pF 的电容器 Cc，

该电容器的振幅小于 200 米 Vrms, X2 管脚未被连接。若选取从属模式, 则将振荡器输出 Fosc 信号的频率限定为 1 至 50 MHz^[6]。在振荡模式下, LPC2210 芯片内部有一个反馈电阻, 因此, 只要将 Xtal 与 Cx1, Cx2 的外部相连, 即可实现基本的振荡^[7]。

1.4 系统外围电路设计

1.4.1 电源电路

无线通信远程数据传输控制系统的控制器是在通讯网络中正常工作的, 其稳定、可靠的电力供应是确保其可靠工作的先决条件^[8]。该系统的供电电路采用线性稳压器, 线性稳压电源具有响应迅速、噪音低、输出波动小等特点^[9]。控制器的 IC 晶片均为 5 V 及 3.3 V。控制器及物理层 PHY 晶片的工作电源为 3.3 V DC, 而储存晶片及 I/O 控制电路则需 5 V 供电。由于控制电路由 5 V DC 供电, 所以由 LM1117-3.3 V 的稳压芯片构成, LM1117 具有过热保护、限制过流的作用, 保障电压稳定度达到了电路工作的需要。

1.4.2 通信接口电路

优化设计的数据传输控制系统需保证最基础的串口通讯, 只需 RXD, TXD, GND 三种通信接口电路即可实现。这种连接只需要三条线路, 也就是模拟了 MCU 的串行通讯方式, 需要用 SP232 实现控制器与电脑之间的电平变换。

1.4.3 液晶显示电路

为直观地显示无线数据采集终端的采集结果, 同时也便于系统的调试, 系统使用 128 × 64 的液晶显示设备, 并采用 ST7920 驱动 128x64 LCD 模组, 既能实现 32 路 com 的输入, 又能实现 64 路 Sg 的输出^[10]。该系统采用 ST7921 驱动 256 × 32 的点阵 LCD, 适用于汉字及图形的显示, 内置 2 M 中文字型 ROM^[11]。另外, 系统还具有 4 位并行功能, 以满足不同的微处理器和 MCU 的接口要求。将显示器设备与系统其他元件按图 4 方式进行连接。

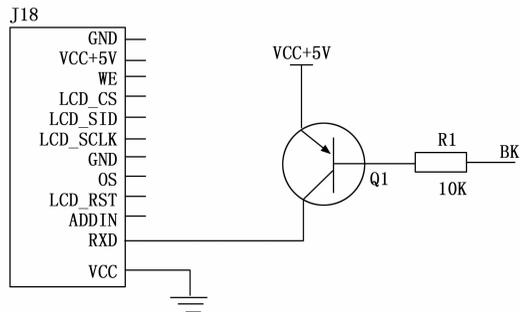


图 4 液晶显示连接电路图

LCD 显示器采用 3 线串联模式, PSB 管脚固定在地面上, 使 I/O 占用较小, 程序编制简便。同时, 通过 STM32 的 IO 来实现对背光的控制, 延长 LCD 屏幕的使用寿命。

除了上述硬件设备和电路外, 数据采集器、通信网络等设备均沿用传统无线通信远程数据传输控制硬件系统中的设备。

2 无线通信远程数据传输控制系统软件功能设计

2.1 无线通信数据收集与处理

启动硬件系统中的数据采集器, 按照图 5 表示流程进行初始无线通信数据的采集。

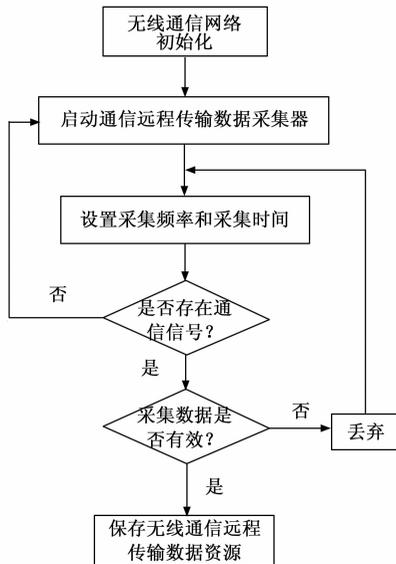


图 5 无线通信数据采集流程图

在采集动态无线通信数据过程中, 通过 SPI 中断服务和中断服务程序来实现, STM32 控制器通过 SPI 中断来接收初始数据, SPI 中断服务中, 将所收到的数据存储在对应的全局阵列变量 spi_data 中, 从而保证当前采集数据的实时性^[12]。为了满足无线传输通信信道的传输规则, 需要对初始采集的数据进行处理, 规范数据的报文格式。首先利用公式 (1) 对数据进行无量纲规范化处理。

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

式中, x 为初始采集的无线通信远程传输数据, x_{max} 和 x_{min} 为数据的最大值和最小值。在此基础上将数据转换为 ACK 报文格式, ACK 报文包含了一个消息的回复和一个特定的序列号的回复^[13]。当接收端收到消息时, 报文的序列被存储在响应包清单中, 并且基于该清单来计算下一次传送的 ACK 域的值。若传送代理传送 1、2、3、4、5 五个消息, 当报文到达接收端传送代理的次序是 1、2、4、5, 则接收端的 acklist 中存储 [1、2、4、5], 则 ACK 是 3, 表明所有小于 3 的消息都被接收。在下次 ACK 报文被传送时, ACK 字段置为 3, t 字段为 001, 并且将 $sn=4$ 、 $sn=5$ 的消息确认应答置于 ACK 的负载中, 通知发送端, 序号小于 3 的消息已被接收, 而序号为 4 和 5 的消息也已接收, 发送者仅需重新发送 3 号消息^[14]。采用该格式和响应机制, 可以极大地降低重复发送的消息, 间接地提高了数据传输的效率。

2.2 设置无线通信远程数据传输控制协议

无线通信远程数据传输选择 TCP/IP 协议作为控制约束

标准，传输控制协议的工作流程如图 6 所示。

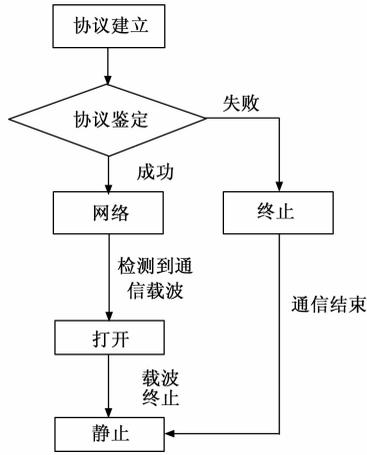


图 6 数据传输控制协议工作流程图

若传输信道处于静态状态，则无实体层连接。在调制解调器的载频信号被侦测并且物理层连接被建立之后，LCP 就会开始进行一些选择。谈判完毕，就进入鉴定阶段。如果通信双方识别成功，就可以进行网络访问。NCP 设定网路层，指定 IP 位址，进入开放状态，进行资料通讯。在完成了数据传送之后，进入了终止状态。当载波停止时，恢复到静止状态。在实际运行过程中，IP 协议用来约束数据包的接收与发送、数据传输信道的搜索、路由以及差错处理等^[15-17]。IP 协议的作用就是把数据包传递给目标主机，因为在某些中间节点上都没有验证的程序，因此无法保证数据包的正确性。在传送过程中，若出现了由路由器缓冲溢出引起的包丢失等差错，协议会依据自己的纠错算法，将此资料包删除，并以响应消息报文的方式传送至源主机，只有高层传送控制协议才能保证其可靠性。

2.3 分配无线通信远程数据

在无线通信远程数据传输控制协议的约束下，假设无线通信环境的初始通道是随机生成的，在此期间，该节点不再变换通道，而是一直在这个通道内。该节点利用伪随机序列生成的种子来计算下一时段的频道，信道是频道的总数。利用伪随机序列生成的种子，下一个时间周期内使用的信道可以表示为：

$$n_{\text{Channel}} = (\beta_{\text{Channel}} + \text{seed}) \bmod (N_{\text{Channels}}) \quad (2)$$

式中， n_{Channel} 和 N_{Channels} 分别为节点将要切换的新信道编号和信道总数量， β_{Channel} 为节点初始信道， seed 为伪随机序列产生的种子。以初始采集并处理完成的数据作为分配目标，则其分配结果可以表示为：

$$b = \ln \left(1 - \frac{3P_{k,m}(x_{k,m}^i)^2}{2N_{\text{Channels}} \ln(5R_{\text{target}})} \right) \quad (3)$$

式中， $P_{k,m}$ 表示的是无线通信数据的发送功率， $x_{k,m}^i$ 为待分配的数据资源， N_{Channels} 和 R_{target} 分别为无线通信网络环境中存在的信道数量以及目标误比特率，另外角标 k 和 m 为数据接收端数量和子载波量。分别将信道、采集数据等相关数据代入到式 (3) 中，便可完成对目标传输数据资源的合

理分配。在实际信道选择与传输过程中，要判断传输信道当前的状态，若信道当前处于运行状态，则需要考虑当前传输数据量，并对传输数据的分配量进行调整。

2.4 实现无线通信远程数据传输控制

根据无线通讯的远程数据传送流程，客户机 $_a$ 和客户机 $_b$ 被定义为发送方， agent_A 和 agent_B 作为接收方，首先在发送方和接收方之间建立 TCP 连接，当接到客户端 $_a$ 的 Connection 请求时， agent_A 将代理响应于 client_a ，并将 client_a 传送到发送队列中，同时向 agent_B 询问 client_b 的状况， agent_B 先查询接收会话信息表 RIB，然后主动地与 client_b 建立联系，并通过 RESPONSE 消息反馈给 agent_B 。一旦 agent_A 接收到 agent_B 和 client_b 之间的连接成功，则告知发送端传送代理将数据从发送队列中提取，并将其封装在 UTP 包的数据区进行发送^[18]。相应地，在接收 UTP 包之后，接收端传送代理按照包类型进行处理，若发现为数据包，则从 UTP 包中分析该数据，通过 agent_B 向客户机 $_b$ 传送一个应答包，并且将响应包传送到发送代理，若发现为响应包，则按照响应包从传送队列中删除相应的数据。在完成了数据传送之后，客户机 $_a$ 主动启动中断连接请求， agent_A 将 FINISH 报文发送给 agent_B ，告知 agent_B 切断与客户机 $_b$ 的连接， agent_B 在收到 FINISH 消息后，切断与客户机 $_b$ 的连接并从接收会话表中删除传输相关会话信息^[19]。按照上述数据传输流程，利用安装的控制设备，在传输协议的约束下，以选择的信道为控制目标，实现系统的无线通信远程数据传输控制功能。无线通信远程数据传输流量控制的目的是最大程度地避免传输信道出现拥塞现象，首先利用式 (4) 计算无线通信网络中任意信道的最大链路容量。

$$Q_i = RTT_i \times \text{Bandwidth}_i \quad (4)$$

式中，变量 RTT_i 和 Bandwidth_i 分别表示的是链路往返时延向量和测量代理查询每个端口的带宽信息。在实际流量控制过程中，根据式 (4) 的计算结果生成一个流量拥塞控制窗口，任意时刻无线通信传输信道流量受限如下公式：

$$\text{awnd} = \min(\text{creditwnd}, Q) \quad (5)$$

式中， awnd 和 creditwnd 分别为允许窗口和拥塞窗口大小。若当前无线通信信道的数据总传输量高于 awnd ，则利用控制设备启动流量控制窗口程序，保证该信道通过流量低于 awnd ，直到信道传输数据低于 $0.6Q$ 为止^[20]。另外通过对流量窗口大小的设置与调节，实现数据实时传输速率的控制，由此从流量和速率两个方面完成系统的数据传输控制功能。

3 系统测试

在系统设计中，每一个模块都要经过验证、测试、检查和不断地修正，从而使各个功能模块能够根据设计的要求来工作，从而实现目标和结果。在对各功能模块进行了测试后，根据对系统的功能需求，对经过测试的各模块进行了装配和测试，以确定系统在运行时是否能够满足所需

的性能指标, 并对其进行改进。为了测试优化设计的基于 STM32 嵌入式的无线通信远程数据传输控制系统的控制功能和运行性能, 设计系统测试实验, 并验证优化设计结果是否满足设计与应用要求。

3.1 搭建无线通信环境

此次系统测试实验所针对的无线通信环境由网关部署环境、服务器运行环境、用户端运行环境 3 个部分组成。其中, 网关与 LoRa 模块的数据输出接口通过串口进行连接, 使之能与 LoRa 网络相连通。可移动网关将路由器与无线网络相连, 由此进入互联网。另外, 在实验室的局域网下面还配置了一个监控 PC, 以实现现场的实时监控。Tomcat 和 MySQL 被部署在服务器中, 用于提供业务支持, 并在开放式系统中使用每个端口进行数据传送。设置多个远程用户端作为无线通信远程数据传输的接收端。

3.2 准备远程数据传输样本

为了给系统测试实验提供充足的数据样本, 收集某个有线通信网络的发送传输信息数据, 以此作为系统测试实验的传输数据样本。表 1 为部分实验数据传输样本的设置结果。

表 1 部分实验传输数据样本设置表

传输任务编号	发送端 IP 地址	接收端 IP 地址	数据发送时间	传输数据大小/MB
1	192.101.225.4	192.101.225.16	8:00:14	79.74
2	192.101.225.7	192.101.225.3	8:05:26	129.35
3	192.101.225.1	192.101.225.7	8:11:08	37.11
4	192.101.225.5	192.101.225.11	8:17:22	207.30
5	192.101.225.12	192.101.225.6	8:21:37	98.13
6	192.101.225.6	192.101.225.4	8:25:52	47.62
7	192.101.225.8	192.101.225.13	8:31:08	105.95

系统测试实验共设置 200 个数据传输任务, 通过传输数据的大小确定合适的传输信道, 以此作为系统控制效果测试的对比标准。

3.3 描述实验过程

根据基于 STM32 嵌入式的无线通信远程数据传输控制系统的优化设计目的, 分别从控制功能和运行性能两个方面进行分析。其中控制功能的测试内容是检测控制系统能否完成数据在规定速率下无误码传输, 同时避免出现数据传输拥塞现象。而控制系统的运行性能主要就是测试该系统能够同时控制无线通信远程传输数据量的多少。在实验环境中安装优化设计系统中的硬件设备, 并执行硬件设备调试工作。启动数据传输控制系统软件程序, 逐一调取数据传输任务。以第一个传输任务为例, 服务器端传送的是 0XFFF10101 的数据, 前面 4 个 16 进制的数据是一个 FFF1, 如果服务器端的 4 个 16 进制的数据是一个 FFF1, 则确认这个网络的数据是有效的, 然后把后面的 4 个 16 进制的数据传送到一个控制 I/O 控制任务以进行对应的控制, 使用相同的方式可以得出其他任务的数据传输与控制结果。通过 Xilinx ISE12.2 开发包中的 Chip Scope, 通过 Chip

Scope 来实现对系统内的信号进行实时探测, 当错误标记比特被触发时, 该错误标记比特信号会一直等待, 表明在发送数据时未发生任何数据传输错误。

3.4 设置系统测试指标

此次系统测试实验设置传输速率控制误差、丢包率、传输拥塞总时长 3 个指标来反映系统的数据传输控制功能, 其中传输速率控制误差和丢包率的数值结果为:

$$\begin{cases} \epsilon_v = |\nu_{\text{target}} - \nu_{\text{control}}| \\ \eta_{\text{loss}} = \left(1 - \frac{N_{\text{receive}}}{N_{\text{send}}}\right) \times 100\% \end{cases} \quad (6)$$

式中, 变量 ν_{target} 和 ν_{control} 分别为设置的数据传输速率目标值和实际控制结果, N_{receive} 和 N_{send} 对应的是数据的接收量和发送量, 另外传输拥塞总时长测试指标的测试结果为:

$$\tau = \sum_i^n \tau_i \quad (7)$$

式中, τ_i 为第 i 次拥塞的维持时长, 参数 n 为传输过程中发生拥塞事故的次数。为了满足系统设计对控制功能的要求, 设置 ϵ_v 、 η_{loss} 和 τ 的预设值分别为 0.5 MB/s、1% 和 3.0 s, 并要求系统控制功能实际测试指标结果不高于预设值。另外, 设置吞吐率作为系统运行性能量化测试指标, 其数值结果可以表示为:

$$B(p) = \frac{L}{RTT \sqrt{\frac{2N\eta_{\text{loss}}}{3}} + T_o \eta_{\text{loss}} (1 + 32\eta_{\text{loss}}^2)} \cdot 100\% \quad (8)$$

式中, L 为无线通信信道的长度, RTT 、 N 和 T_o 分别为数据传输的往返时间、单个 ACK 确认数据量和数据传输超时重传定时器参数, T_o 取值为定值。另外 η_{loss} 为数据丢包率, 可以通过式 (6) 的计算得出该变量的具体取值, 实验要求吞吐率不得低于 88%。

3.5 系统测试结果分析

通过相关数据的提取与统计, 得出系统控制功能的测试结果, 如表 2 所示。

表 2 系统控制功能测试数据表

传输任务编号	数据传输速率目标值/(MB/s)	数据传输速率实际控制值/(MB/s)	传输数据接收量/MB
1	30.0	30.0	79.61
2	40.0	39.7	129.14
3	10.0	9.8	37.08
4	50.0	49.6	204.25
5	30.0	29.7	97.07
6	20.0	20.0	47.35
7	30.0	29.9	105.23

将表 2 中的数据代入到式 (6) 中, 可以得出控制系统下无线通信远程数据传输速率控制误差和丢包率对的平均测试结果分别为 0.18 MB/s 和 0.6%。另外通过式 (7) 的计算, 得出传输拥塞总时长的测试结果如图 7 所示。

从图 7 中可以直观地看出, 在设计系统的控制下, 远程数据传输发生拥塞时间的最大值为 2.6 s。由此可见系统

(下转第 146 页)