

Design and Implementation of Real-Time Data Exchange Software of Maneuverable Command Automation System

Shi Chuan, Zhang Yang and Zhou Yuefei

1 Introduction

Command automation system provides an effective means of command under the conditions of information technology, it has been hailed as the “central nervous system” and “force multiplier” of the command, so command automation system confrontation determines the success of information confrontation [1]. For the rapidly changing situation, maneuverable command automation system can better adapt to the requirements motor performance and rapid response capability under real situation.

Maneuverable command automation system carries on the network exchange and data communication with external system or equipments through encipherment/decryption [2, 3]. There are many kinds of hardware maneuverable command automation system. The communication means and the frequency band are different. The interface is complex and the data throughput is large. In order to ensure the continuous and reliable operation and the proper performance of the system, there must be a stable real-time and high-efficiency data exchange system to guarantee the various types of information transmission, processing and other functions to achieve stability between hardware equipments. This paper designs and implements a real-time data exchange software for mobile maneuverable automation system based on requirement analysis using dynamic multi-threading and multicasting technology. It solves the problem of congestion and collapse caused by the large amount of data flooding, which have achieved good results in the test.

S. Chuan (✉) · Z. Yang

Luoyang Electronic Equipment Test Center, Luoyang 471003, China
e-mail: shi_chuan@163.com

Z. Yuefei

The Army of 66135, Beijing 100000, China

© Springer Nature Singapore Pte Ltd. 2018

Z. Deng (ed.), *Proceedings of 2017 Chinese Intelligent Automation Conference*,
Lecture Notes in Electrical Engineering 458,
https://doi.org/10.1007/978-981-10-6445-6_2

2 Software Requirements Analysis

According to the needs of a certain task, the maneuverable command automation system is mainly to achieve information collection, transmission, processing automation, and support decision-making and in the process, which could ensure that the command of commander to the participants and equipments. Its real-time data exchange software needs to complete the real-time data exchange among the working groups, and accomplish the tasks of complex data communication, transmission, processing, display and control.

Based on the analysis of actual demand and technical, the real-time data exchange software should have the following functions:

1. Data access and distribution: This function can forward the external data to the application seat of the system through internal and external network bridging way.
2. Protocol analysis and recording: This function can achieve the original data protocol analysis, and record the access data efficiently and completely. It can be distributed deployment and realize multiple backup to ensure data security.
3. Data processing and display: This function can complete the two-dimensional track display, data fusion and other required data preprocessing. It also can achieve the real-time display of variety target information and situation.

3 Software System Design

3.1 Structure Design

According to the system functional requirements analysis, the real-time data exchange software is divided into four configuration items: data access and distribution, protocol analysis and conversion, data recording, data processing and display. The main work flow of the real-time data exchange software applied to the maneuverable command automation system is as follows: Data access and distribution configuration item receives the external data, and send them to protocol analysis and conversion configuration item to finish the protocol conversion, and to data recording configuration item to finish data recording. Then the various types of data after analysis will be output in different directions. Data processing and display configuration item preprocesses the received data. The data results processed can be displayed visually based on the requirements. The structure design of this software is shown in Fig. 1.

- (1) Data access and distribution configuration item includes the parameter configuration module and data scheduling module.

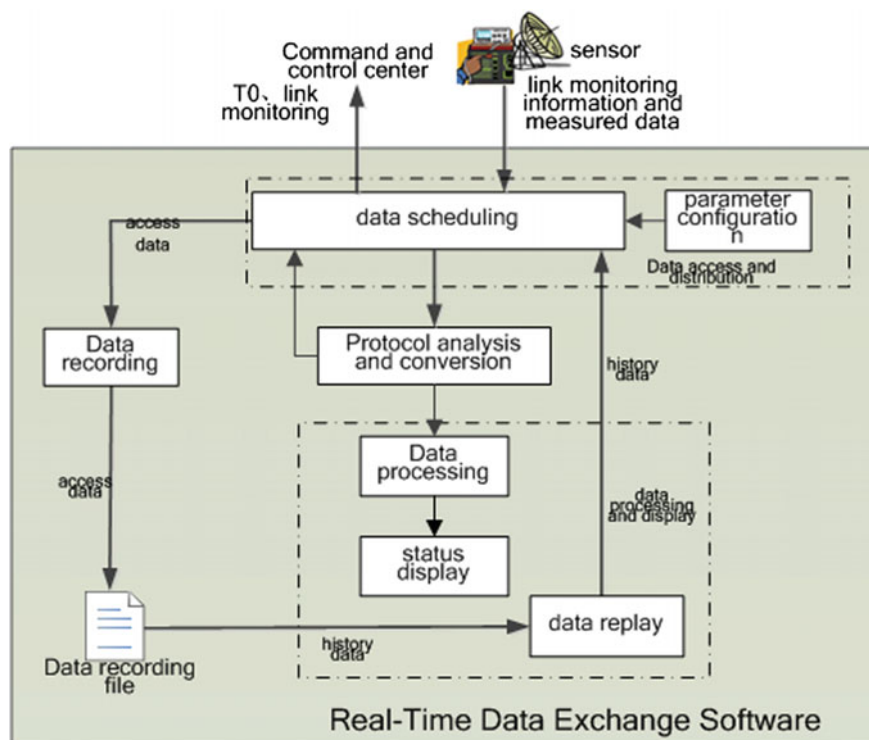


Fig. 1 Software structure design

Parameter configuration module mainly provides data distribution configuration table, operation and maintenance interface and API interface.

Data scheduling module mainly receives all the data sent from the system outside of the system and the data packets from the internal software. It checks the legitimacy of the received data and rejects the illegal data. Through configuring, it can finish the directional transmission and multicast transmission of the received data in the system.

- (2) Protocol analysis and conversion configuration item mainly finishes the send and receive protocol conversion of the data send by outside system. The external protocol format is converted to internal protocol format, which will be sent to system internal software for follow-up processing. It also finishes the send and receives protocol conversion of the data send by internal system. The internal protocol format is converted to external protocol format, which will be sent to system external software to ensure the entire data requirements.
- (3) Data record configuration item carries out efficient and complete record on the received data. Data scheduling module will receive all the data sent through the private forwarding channel to the data recording module. These dates will be

recorded by the data recording module in accordance with a custom format. Data recording module can be distributed deployment and achieves multi-backup, which ensure data security.

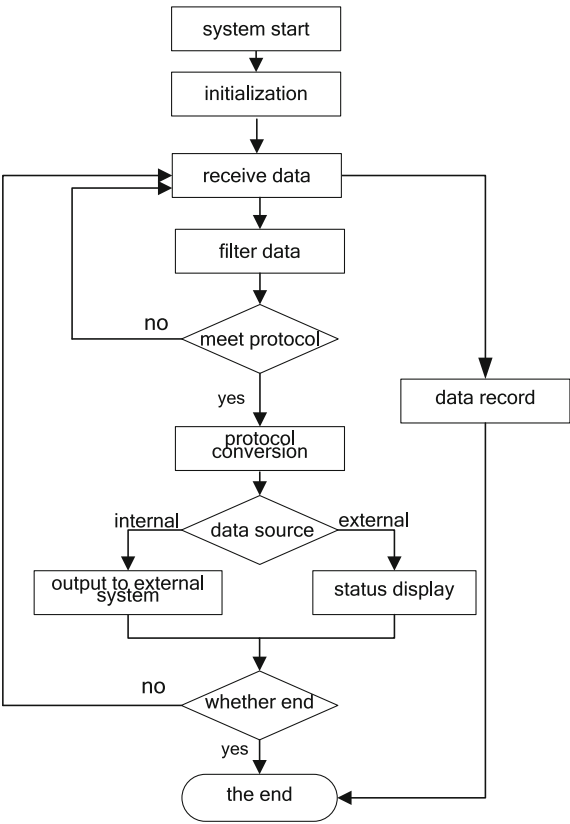
(4) Data processing and display configuration items include data processing module, status display module and data reproduction module.

The data processing module mainly processes the received data in real time. The status display module has the function of state display of system and each device. It can display the state information in real time. The data replay module is used to replay the data afterwards. The recorded data can be sent to the internal network according to the selected data interval, and the software in the system can be driven to run, which is help for the process analysis.

3.2 Software Running Process

Real-time data exchange software running process is shown in Fig. 2.

Fig. 2 Real-time data exchange software running flowchart



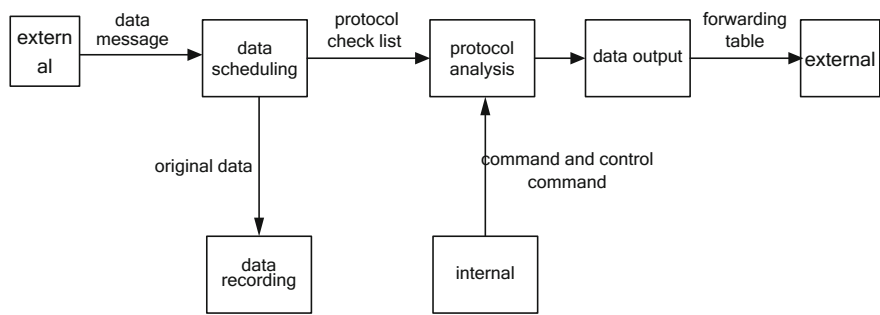


Fig. 3 Real-time data processing and management interface diagram

After the system starts, the system is initialized, and the parameter configuration is read, which includes the source IP address and port number, the destination IP address and port number, the address and device mapping table, the data distribution frequency, etc. Then the data is received. There are two operations for data at the same time. One is to record data directly, which is used for data analysis. The other one is to filter the data received. The data which do not meet the protocol will be discarded. The data which meet the protocol will be carried protocol conversion. According to different sources, the data will be output to different locations. The internal data received is output to external system. The external data received will be sent to status display module for real-time display. If the end has not been selected, the system is waiting for receiving data again. If the end has been selected, the software is end of operation.

3.3 Software Interface Design

Software interface design is shown in Fig. 3. All of the interface types are real-time data transmission. The operating states are normal. According to the interface connection the software data flow can be found out.

4 The Key Technologies and Solutions

4.1 Dynamic Multithreaded Programming

A thread is a single sequential control flow in a program that contains separate stack and CPU register states. Each thread shares all the process resources, including files opened, signal identifiers, and memory allocated dynamically, etc. Multi-threaded is that multiple threads are running simultaneously to complete different work in a

process, in which there is a main thread. If multiple tasks need to be dealt with at the same time, the use of multi-threaded is the ideal choice [4, 5].

Real-time data exchange software in the multi-channel real-time data processing applications, will transform a data packet received into a number of different formats of messages and forwarded to multiple destination addresses, in which case the use of dynamic multi-threaded approach can improve the system efficacy. According to the source address, the main thread can determine whether to open a new thread. If the source address is a new address, a new thread will be opened to receive and manage data send by this new source address. If there is no information of the source address for a long time, the thread corresponding to this source address will be deleted. So that system resources can be released in real-time. In high frequency data transmission of multiple data sources, dynamic multi-threaded approach can effectively improve the system operating efficiency and reduce congestion and other phenomena.

4.2 Multicast Planning

Multicast is a communication mode that enables point-to-multipoint network connectivity between the sender and each receiver. If a sender simultaneously transmits the same data to multiple receivers, only one copy of the same packet is replicated. Hosts can request to join or exit a group from a router. Hosts in the same group can receive all data of the group. Routers and switches in the network selectively copy and transmit data, and only transmit data to the group host, so that the data can be transferred to multiple groups in one time, and not affect the external host communication. Multicast improves data transmission efficiency and reduces the possibility of congestion in the backbone network [6–8].

In order to avoid network congestion caused by multi-channel high-speed data, real-time data exchange software groups the network nodes according to the data stream type. For example, the related information to measurement data of equipment 1 is assigned to a group 1 for transmission; command information is assigned to a group 2 for transmission; telemetry and remote control information is assigned to a group 3 for transmission. According to the data stream type, the network nodes are grouped, so that certain types of information can be propagated only among specific nodes, which do not affect the data transmission and reception of other nodes. The application of multicast technology improves the efficiency of the network, reduces the network congestion and improves the stability of the system.

4.3 Protocol Conversion

Because each equipment and the command system use different the real-time data information, it is need to convert protocol of the data when each equipment

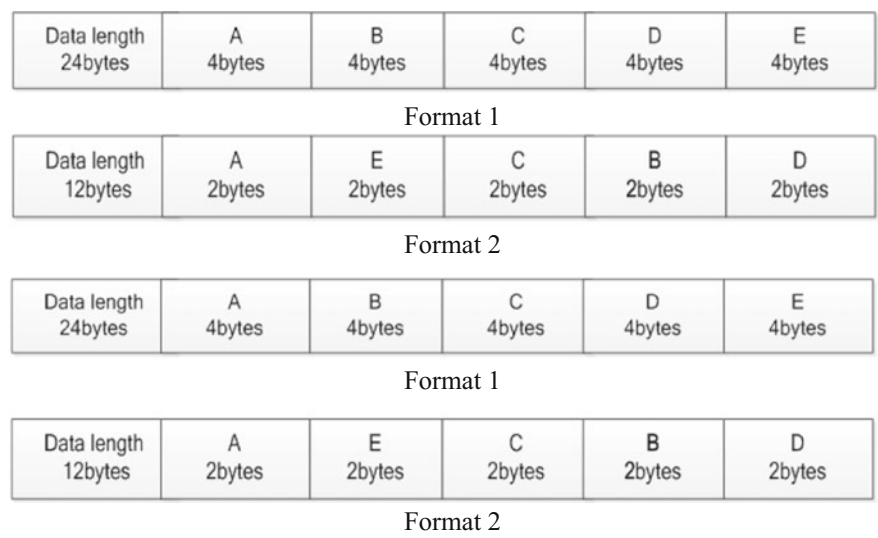


Fig. 4 Format conversion diagram

information is sharing. Specific conversions are as follows: The same information is described by two different structural forms and sequences. For example, there are five fields of A, B, C, D, and E, whose information are described in the following two ways, as shown in Fig. 4.

The information fields A, B, C, D and E in the format 1 are defined as INT data types. Each information field in the format 2 is defined as the short data type. In the data exchange process, it is need to do the following processing.

- (1) After the serial data packet is received from the network, it is copied into the structure of format 1.
- (2) According to the definition of the structure, the data of each information field is read.
- (3) The data type of each information is converted from INT data type to short data type corresponding to the information field of format 2.
- (4) The information is rearranged according to the format 2 structure.
- (5) The data after the reorganization is copied into the data packet to sent from the network.

5 Software System Implementation

According to the analysis and design, the software is completed using VC++ as the programming language. The software is deployed in a maneuverable mobile command automation system. Figure 5 shows two-dimensional situation display

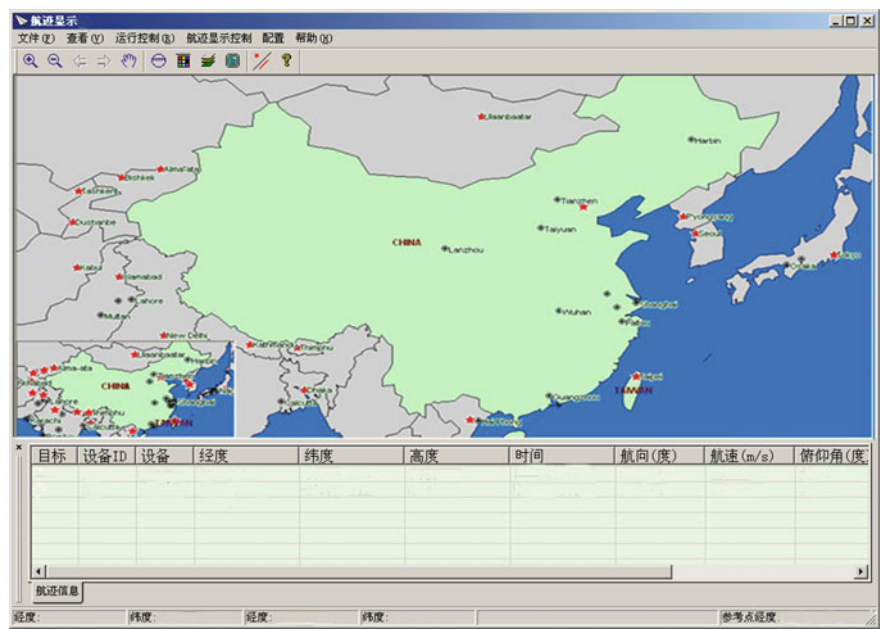


Fig. 5 Two-dimensional situation display



Fig. 6 Target information measurement results

interface in the real-time data exchange software. Figure 6 shows the target information measurement results displayed by the real-time data exchange software. The target information received is consistent with the target equipment state. The

simulation results show that the software can receive and display 25 batches and 100 targets track information correctly and in real time. There is no congestion and collapse when large amount of data is processed. The data records are established correctly in the background and good results are obtained.

6 Concluding Remarks

Based on multicast technology and dynamic multi-thread technology, a real-time data exchange software for maneuverable command automation system is designed and realized in this paper, which greatly improves the data throughput and greatly reduces the congestion in data transmission process, etc. It is proved through an application that the real-time data exchange and processing ability of the software can withstand the test of high-speed and large-flow data, and the software guarantees the completion of the task successfully.

References

1. Xiong Y, Teng ZP, Zheng HL (2006) Distributed artillery command automation system based on movable agent. *Fire Control Command Control* 31(1):52–54 (in Chinese)
2. Wang Y (2005) EMC design of the maneuverable C4I system. *Fire Control Command Control* 30(3):96–98 (in Chinese)
3. Wand SL, Dou B, Yang YM (2008) Power supply design for the motorized command automotive system. *Mob Power Veh* 4:8–10 (in Chinese)
4. Zhu HT, Yao M, Cai TY et al (2011) Multi-thread programming for image display of navigational radar. *Mod Electron Technol* 34(12):108–110 (in Chinese)
5. Hu M (2008) Windows network programming technology. Science Press, Beijing (in Chinese)
6. Huang K, Guo S (2011) Application of IP multicast in radar data processing. *Electron Design Eng* 19(17):16–18 (in Chinese)
7. Chen J, Chen W (2001) Visual C++ network advanced programming. People's Posts and Telecommunications Press, Beijing (in Chinese)
8. Gong LF, Ma Q, Pan CD (2007) Research on optimizing performance of multicast network. *Comput Modernization* 11:78–83 (in Chinese)

Proceedings of 2017 Chinese Intelligent Automation
Conference

Deng, Z. (Ed.)

2018, XI, 791 p. 372 illus., Hardcover

ISBN: 978-981-10-6444-9