

# Design of AGV 3D Real-Time Simulation Control System Based on Wireless Network

Sun Hui\*, Yu Hongfei, Nie Qixuan, Wang Manfei, Yan Yanfei

College of transportation, Shandong University of Science and Technology, Qingdao, china

## Email address:

576507258@qq.com (Sun Hui), yuhongfeiqing@qq.com (Yu Hongfei), 609362504@qq.com (nie Qixuan),

15764255942@163.com (Wang Manfei), 824661974@qq.com (Yan Yanfei)

\*Corresponding author

## To cite this article:

Sun Hui, Yu Hongfei, Nie Qixuan, Wang Manfei, Yan Yanfei. Design of AGV 3D Real-Time Simulation Control System Based on Wireless Network. *International Journal of Systems Engineering*. Vol. 2, No. 1, 2018, pp. 15-20. doi: 10.11648/j.ijse.20180201.14

Received: June 28, 2018; Accepted: July 9, 2018; Published: July 30, 2018

---

**Abstract:** For most of the AGV superior control system is planar two-dimensional, poor visibility and lack of equipment operating efficiency statistics, A three-dimensional AGV simulation control system based on wireless communication and Flexsim logistics simulation software is proposed.. Through the socket connection zigbee wireless transmission module serial port and Flexsim software completes the data communication between the processes, thus establishes AGV and Flexsim data communication. Build the 3D system simulation model of AGV and write the control algorithm through Flexsim. At the same time, call the SQL database data to complete the real-time simulation and analyze the running status in real time. Mainly studied the data communication interaction between the AGV wireless communication module and Flexsim, using the AGV of the college laboratory to conduct field tests. The results show that the design program basically achieves the expected functions and effects.

**Keywords:** 3D Control System, Real-Time Simulation, Socket, Flexsim, AGV, Automatic Control

---

## 1. Introduction

With the deepening of the concept of Industry 4.0, the production, equipment, and logistics industries are facing an overall upgrade. Many companies are transforming and upgrading the manufacturing industry, and they are moving towards automation, intelligence, and flexibility [1]. Smart AGV systems are widely used in the transformation and upgrading of smart logistics in the manufacturing industry. In the process of industrial transformation and upgrade under the new normal background, the manufacturing industry and logistics companies have introduced new technologies and new methods to improve their logistics efficiency. This has become a major trend, and improving the efficiency of AGVs is an important part of transforming and intelligent production. [2].

## 2. AGV 3D Simulation Control System Overview

The AGV upper control system is the core of the entire AGV control system and controls the movement of the AGV. The

hardware of the control system is composed of a computer and related peripherals, and the communication equipment and the AGV communicate wirelessly. In terms of software, the control system software can perform intelligent assignment of tasks and can observe the operational status of the robot AGV in real time. If the AGV fails or has other unexpected conditions, the control system can implement automatic detection and alarm measures. The control system can also implement multiple AGV cluster control scheduling [3].

At this stage, most of the AGV upper control system interface is flat two-dimensional, poor visibility, and lack of statistical analysis output of system operation efficiency, unfavorable follow-up to the overall evaluation and optimization of the running automation system. Aiming at this disadvantage, this paper presents a three-dimensional simulation control system based on socket process. The software architecture uses Flexsim three-dimensional simulation software as a platform to establish a three-dimensional simulation control system based on socket process communication to AGV.

Flexsim is an object-oriented visual 3D logistics modeling and simulation software for visual modeling and simulation of dynamic discrete events and systems [4].

The structure of the three-dimensional simulation control system researched and designed in this paper is mainly divided into three parts: three-dimensional simulation monitoring module, control system module, signal data communication module, and data statistics module.

The three-dimensional simulation monitoring module is mainly based on equipment model parameters, real-time status information and other parameters, based on the custom task sequence of Flexsim simulation engine to generate the corresponding task discrete logistics system, to build the basis of three-dimensional dynamic simulation; the control system module is mainly in the control algorithm. On the basis of decision feedback and can be manually controlled through the secondary development of the interface GUI; signal data communication module based on Socket process communication, combined with RS-232 serial communication Zigbee wireless communication module to achieve signal communication with the AGV; data statistics module based on The Flexsim platform internal data analysis and statistics system can output data and status statistics that users care about after secondary development.

The main workflow of the three-dimensional simulation control system is:

- (1) Set up a logistics simulation model based on actual equipment and site environment parameters using Flexsim.
- (2) Flexsim simulation engine and AGV exchange data in real time, including AGV location information, loading status, running status, etc., and store this information in the database

of the system platform.

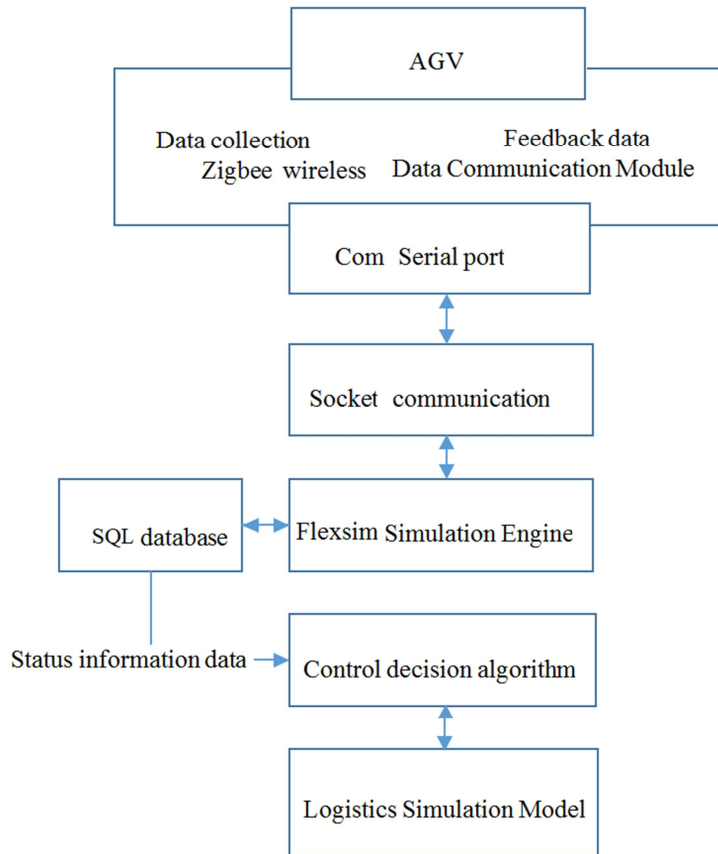
- (3) When the system is running, the simulation engine invokes the control algorithm to calculate and make decisions on the acquired signal data information, and feedback the decision-making control signal to the AGV to complete the control process.

- (4) The three-dimensional simulation model acquires the relevant information in the database and converts it into data required for operation of the simulation model, such as action information, and completes secondary conversion and processing of the signal data, thereby real-time simulation.

- (5) Simultaneously run the data statistics system during the real-time simulation to perform intuitive and concise data output.



**Figure 1.** AGV industry application case.



**Figure 2.** Flowchart of 3D simulation control system.

### 3. Signal Data Communication Module Design and Implementation

#### 3.1. The Composition of the Signal Data Communication Module

The signal data communication module hardware is mainly composed of an AGV wireless communication module (frequency 2.4GHz), a zigbee wireless module (frequency 2.4GHz), and a Flexsim simulation control system server.

#### 3.2. The Working Principle of the Signal Data Communication Module

Signal data communication module implementation ideas: When receiving AGV signal communication data, the data received by the Zigbee wireless module connected to the RS-232 serial port of the PC platform where the simulation control system server is located is read into the Buffer, and then the data in the Buffer is written In the Socket's Server (Flexsim), vice versa, the data acquisition and data feedback functions are completed to establish the data communication function between the AGV device and the simulation control system.

Socket is a session-level application programming interface (API) built into the operating system kernel that allows a process to communicate with other processes, allowing applications to share TCP/IP connections to the network. Socket data transmission is actually a special type of I/O [5].

#### 3.3. Data Interconnection Between Serial Port Wireless Communication Module and Flexsim

Based on the TCP protocol, using Flexsim's Socket function to achieve data exchange with the serial Zigbee wireless communication module. First need to start Flexsim, create a Socket socket; Bind the socket (Bind) to the specified port; Then send a Connect request to the serial communication module; After the connection request response is successful, you can transfer data.. The SendData and GetData methods are used for data transmission and data reception, respectively, and are applicable to the Server server (Flexsim) and Client client (Com serial). At the end of the connection request, the connection is closed and the socket is closed, ending Socket communication [6].

Connect the serial port in three steps:

- (1) initialization of object parameters;
- (2) Com serial port creation and parameter settings;

```
hCom = CreateFileA (name, GENERIC_READ |
GENERIC_WRITE, 0, NULL, OPEN_EXISTING, 0,
NULL);
```

- (3) Set the parameters for the device name, such as Com1, access mode, sharing mode, security settings, etc. [6];

```
DCB dcb;
```

```
dcb.DCBlength = sizeof (DCB);
```

```
GetCommState (hCom, &dcb);
```

```
dcb.BaudRate = baudrate;
```

```
dcb.ByteSize = 8;
```

```
dcb.Parity = NOPARITY;
```

```
dcb.StopBits = 0;
```

- (4) Update the object parameters after successful connection, start the thread to exchange the communication data.

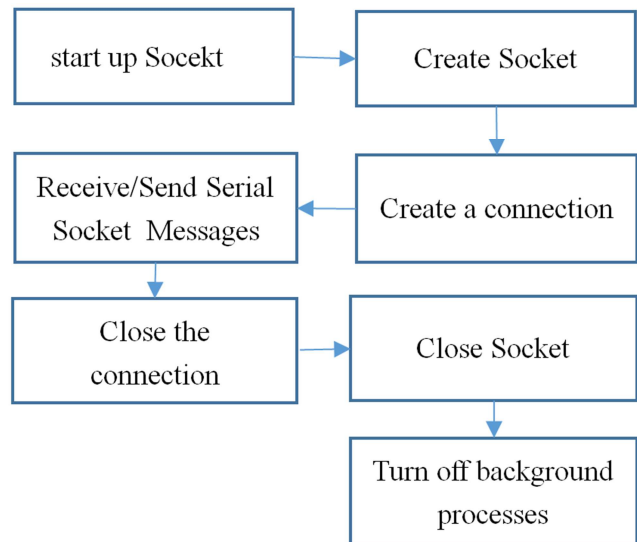


Figure 3. Data Communication Module Design Process.

As shown in Figure 3, the steps to communicate with Flexsim via the socket Socket (and the corresponding FlexSim command) are as follows:

1. Socketinit ()// Start the Windows background processes
2. Servercreatemain (int port)// Create a socket
3. Serveraccept (intnoblocking)// Create a connection
4. Serverreceive (intconnection, char\*buffer, intbufferSize, intnoblocking)// Accept messages
5. Servercloseconnection (int connection)// Close the connection created in step 4
6. Serverclosemain ()//Close the socket created in step 2
7. Socketend ()// Stop the Windows background process

After the port is created and connected to the Socket, Flexsim script triggers call the Senddata (send data) and Getdata (receive data) functions cyclically to check the notification data on the port, thus completing the communication data sending and receiving functions.

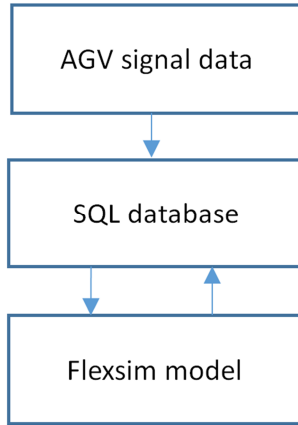
The socket Socket is based on the TCP/IP protocol and establishes the connection mechanism between Flexism and serial communication so as to establish a wireless communication connection with the AGV and complete the data signal communication with the device.

### 4. Design and Implementation of 3D Simulation Monitoring Module

Using Flexsim can indirectly simulate the running status of

AGV. At the same time, it can also use Flexsim to perform real-time online simulation of AGV.

The implementation process of the real-time simulation model mainly includes the data acquisition of the AGV equipment control site and the data call required for running the Flexsim model [7]. The data transfer between the device and Flexsim is based on the database SQL, as shown in Figure 4.



**Figure 4.** Simulation model data call flow.

First, set up a logistics simulation model according to the actual equipment parameters. In the data transmission process, there is a one-to-one correspondence between the device signal data, the SQL database, and the Flexsim model. During data acquisition, the device signal data is saved in the corresponding database variable. When the data is called, the device variable value corresponding to the Flexsim device name is found from the database, and the call of the simulation entity corresponds to the variable [8].

Flexsim can store information into a SQL database. Similarly, Flexsim can access SQL databases through ODBC interfaces, read relevant data information from the database [8], and perform dynamic applications in Flexsim. Its main operating steps are:

Create ODBC Alias on Windows (data source), set a database name and connection database called in Flexsim.

(2) Flexsim script functions can be used to call database data in Flexsim. For example, the `dbopen ()` command

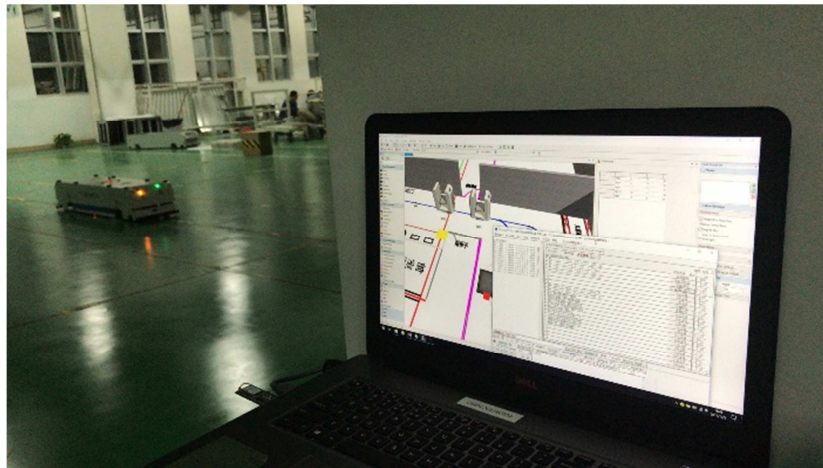
accesses the database and opens it for data writing or reading. Such as: Enter "`dbopen ("flexsimdata", "agvstate", 1, 0)`" in the Flexsim entity script;" The first parameter is the name of the database, and the command in the windows ODBC Data Source Administrator (ODBC) data source manager same. The second parameter is the table that we want to read in and read out the data. The third parameter is to switch between using table mode (1) and SQL mode (0).

(3) Use `dbgettextnum (num row, num col)` to read the numerical data in the database, and use `dbsettablenum (int row, int col, double val)` to write the database data.

(4) Use `dbclose ()` to shut down the database. This approach establishes a link between Flexsim and a database called "Flexsimdata" and makes calls to the relevant data on the database. In addition, it uses Flexsim's real-time triggered scripts. Therefore, with Flexsim software as a simulation platform, the use of script triggers can read the data in the database in real time, thus completing the implementation of data refresh, thus realizing the function of real-time simulation and monitoring [9]. When AGV is loaded and unloaded, the motion state is used. The method of inserting a custom task sequence is implemented. When receiving the signal value of the unloaded load or the moved position, the AGV immediately executes the task instruction, thereby completing the function of the three-dimensional simulation monitoring [10].

## 5. AGV Simulation Control Example

The simulation control example is designed based on the requirements of the AGV simulation control system in the AGV project of the Logistics Equipment and Control Laboratory of Shandong University of Science and Technology. Its main equipment includes two two-wheel differential drive submersible AGVs, and the navigation method is RFID magnetic guidance. The simulation control system requires that the AGV can be started and stopped at the designated card point, and the lifting rod of the AGV can be lifted to complete the loading and unloading of the goods, and real-time simulation of the AGV state is performed. The running state needs to be statistically calculated and output.



**Figure 5.** Lab Field Test.



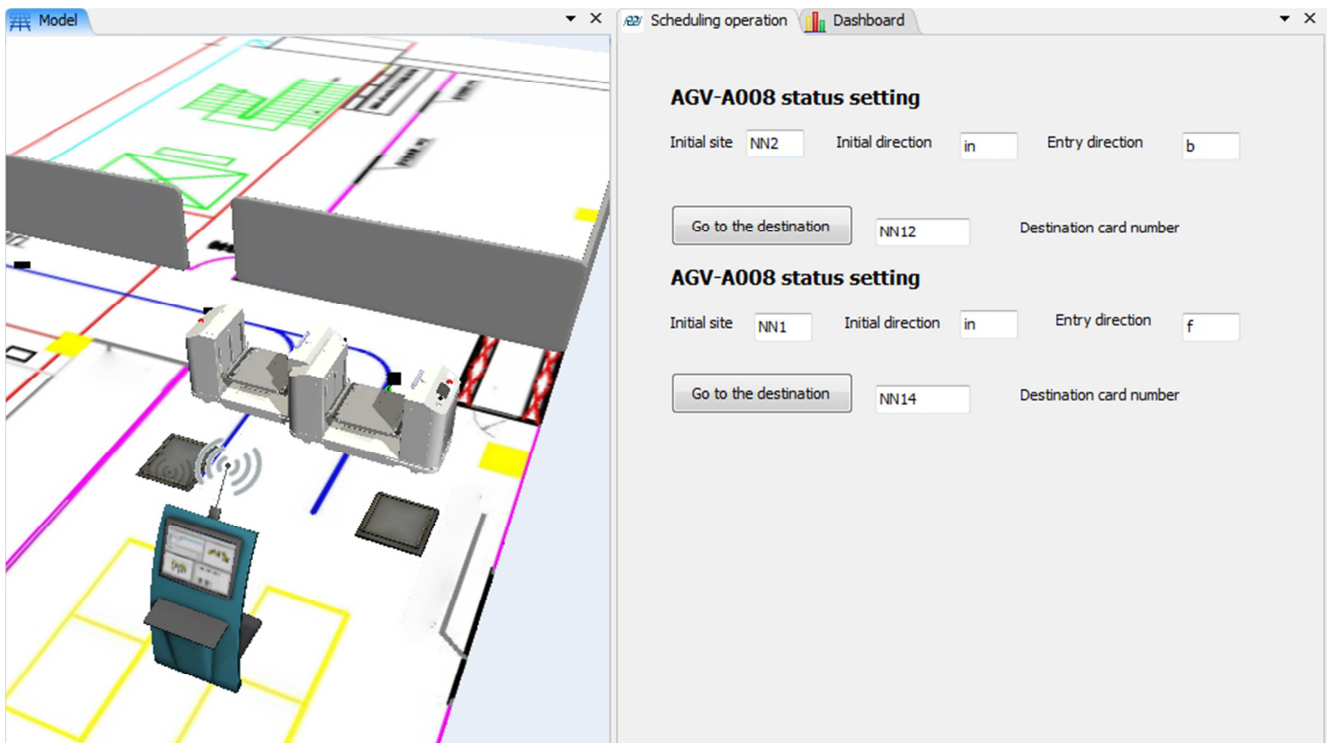


Figure 6. Simulation control system monitoring interface.

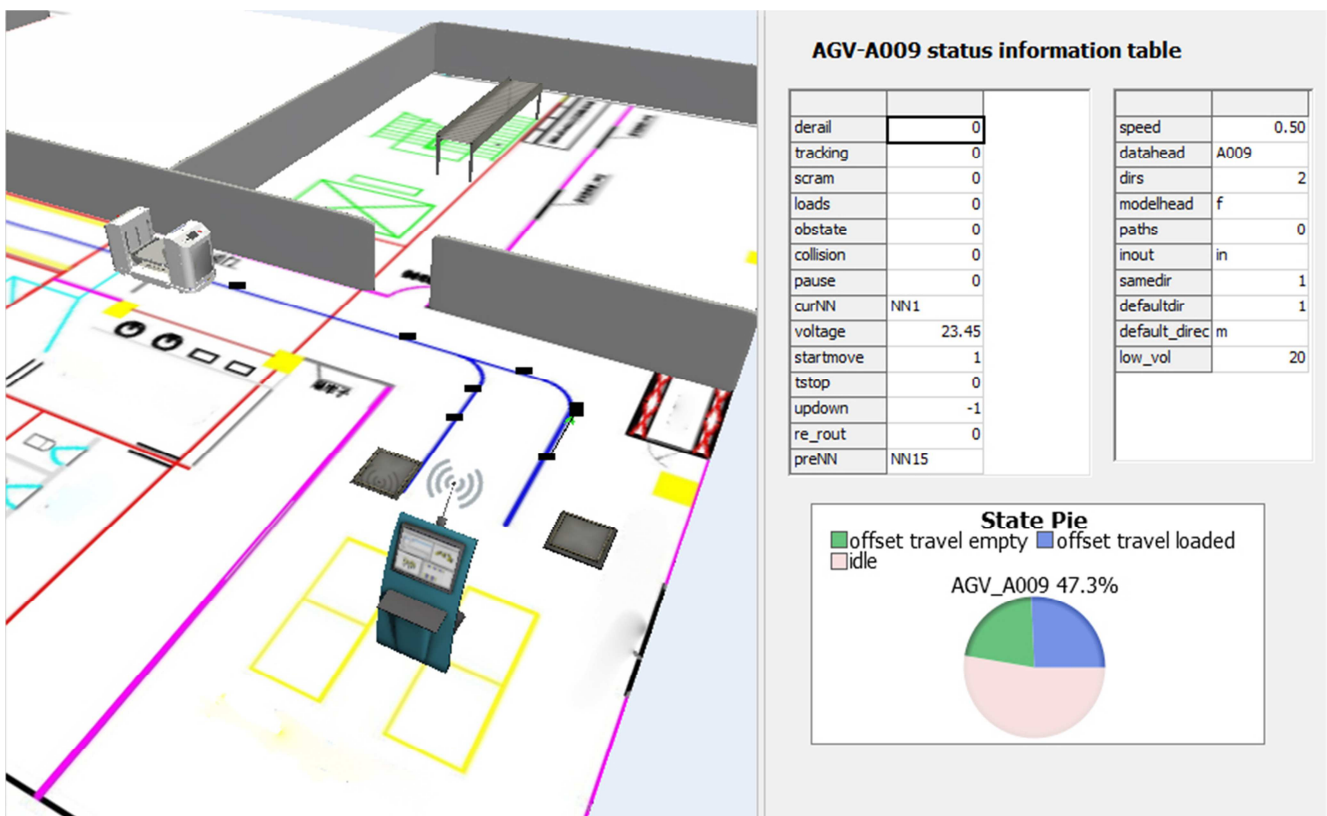


Figure7. Real-time statistics of AGV status.

When the loading and unloading points of the AGV are set in the GUI control interface, click on "Go to the destination", the system will calculate the path according to the written path algorithm and send the path information to the AGV.

Perform the task of loading and unloading the goods. When at the point of loading, the AGV will "rise the lift bar" to complete the loading of the cargo. When the unloading point is reached, the "lifting rod falls" to complete the unloading of

the goods. In the actual operation of the AGV, the AGV sends its current status (current card number position, arrival time, AGV status, etc.) to the control simulation system through the communication module each time it passes through one card point in the path, and at the same time controls the simulation system. The received status information is processed in real time, and Flexsim also performs real-time simulation of the three-dimensional interface according to the operating data in the database. At the same time, the status information of the AGV is output. The status information panel in the lower part of Figure 7 is a real-time status information statistics dashboard, and the information of the “empty”, “loaded”, and “idle” of the AGV system is separately collected. Therefore, the real-time running status of the AGV system can be obtained.

## 6. Conclusion

This paper establishes a three-dimensional simulation control system for the AGV operation process of logistics automated handling equipment. This system not only realizes the automatic and manual control of automatic equipment AGV, but also performs three-dimensional real-time simulation and achieves high-quality monitoring results. The statistical output of AGV operating status data during system operation provides the basis for data and decision making and helps to further analyze and optimize the actual system. The realization of the manual control function also extends the application of logistics simulation software in the field of logistics automation control, so that the simulation can participate in the whole life process of the entire logistics system for data acquisition and simulation optimization, which will help improve the logistics efficiency.

---

## References

- [1] Hou Yizhuo. Future of China's Supply Chain in the Age of Industry 4.0 [J]. Beijing: Modern Manufacturing, 2014 (29):20-20.
- [2] Wang Biao, Zhang Xibiao. Design of Intelligent Logistics System Based on AGV Powertrain Plant [J]. Maanshan: Automation Application, 2017 (11):143-144+152.
- [3] Zhang Yuchun. Research and design of AGV positioning guidance and control software system [D]. Chengdu: University of Electronic Science and Technology, 2017.
- [4] Zhou Dongqing, Deng Endi, Cao Xiaohua. Visual monitoring technology of container crane based on Flexsim 3D simulation [J]. Wuhan: Port Handling, 2015 (4):1-4.
- [5] Liu Xinyu. Research on AGVS Control Technology and Path Planning Method Based on Wireless Network [D]. Shanghai: East China University of Science and Technology, 2016: 60-64.
- [6] Zhou Yang, Zhou Meijiao, Huang Bo et al. Research and design of serial communication system based on C#[J]. Shanghai: Electronic Measurement Technology, 2015 (07):135-140.
- [7] Xie Xiaocheng, Jiang Lili, Xu Yuzhi. Design of real-time simulation system for PLC automatic control production process [J]. Guangzhou: Journal of North University of China (Natural Science Edition), 2012 (3):282-287.
- [8] Hu Hui, Zeng Tao, Xuan Dengdian et al. Simulation of a Comprehensive Passenger Terminal Based on Flexsim Secondary Development [J]. Xi'an: Journal of Chang'an University (Natural Science Edition), 2012 (4):76-81.
- [9] Yin Chao, Zhang Fei, Li Xiaobin, etc. Visual dynamic monitoring system of production tasks for multi-variety small batch machine shop [J]. Chongqing: Computer Integrated Manufacturing Systems, 2013 (01):46-54.
- [10] Tang Haochen. Research on Path Planning of Multi-AGV System [J]. Nanjing: Jiangsu Science and Technology Information, 2018 (11):57-59.