2025

# BRICS SKILLS COMPETITION

## (BRICS+ FUTURE SKILLS & TECH CHALLENGE)

# Internet of Things

BRICS-FS-16

# Technical Description

(International Final_Offline)

Jun, 2025

# Contents

# Precautions

1.Check whether the hardware equipment and computer devices are functioning properly. Check all the equipment, software and competition materials required for the competition;

2.All kinds of software tools, software installation files and other competition materials used in the competition tasks have been copied to a USB flash drive. Please have the contestants copy the competition materials from the USB drive to the D drive of the computer and use them as required by the competition tasks.

3.During the competition, please strictly follow the descriptions in the competition tasks to install, configure, operate and use each Internet of Things device. For devices that have been connected before the competition, they may be related to subsequent competition tasks. Please do not make any changes.

4.After completing the competition tasks, it is necessary to save the device configuration. Do not turn off any devices, do not disconnect the hardware connection lines, and do not encrypt the devices at will.

5.Contestants should complete the contents required by the task book within the prescribed time. The files and materials formed during the task implementation process must be stored in the designated folder under the root directory of the "D Drive" on the server computer, and at the same time, a copy of the "Submitted Materials" should be made to the root directory of the USB drive. Files not stored in the designated location will not be scored.

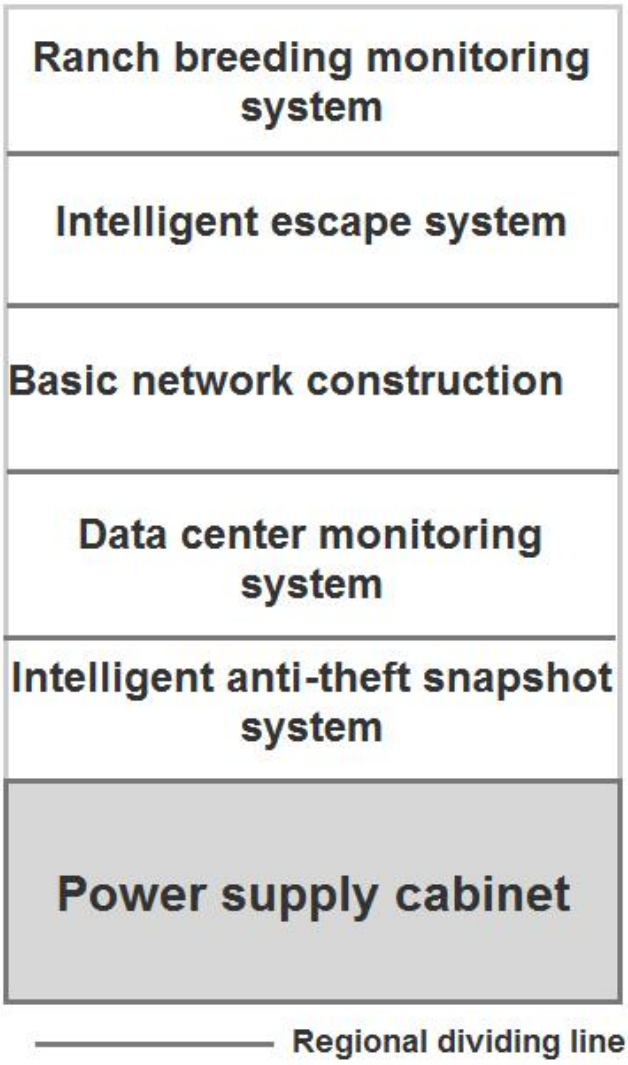
# Hardware environment

| No. | Device Name | Quantity |
|---|---|---|

| | | |
|---|---|---|
| 1 | Internet of Things System Integration Engineering Training Platform (NLE-ISE840) | 1 |
| 2 | Iot toolbox and consumables package | 1 |
| 4 | Server Computer(There are annotations on the computer) | 1 |
| 5 | Workstation Computer(There are annotations on the computer) | 1 |

# Module A: IoT Engineering Design and Implementation(55 Points)

Note: According to the description requirements of each subsystem, complete the corresponding tasks. The result files of this module need to be saved to the "D:\Submit Materials\Module A" folder on the server computer. For reusable equipment, the equipment area layout can take one of the subsystems.




Equipment area layout map

Note: It is required to stick black electrical tape on the cover of the wire trough for dividing the area, indicating that the wire trough is a regional division line. In the

designated area, contestants can appropriately add additional cable troughs based on the actual situation for the wiring of the equipment.

# 1.Network link system

Set up the Internet of Things network link environment and install the corresponding equipment in the basic network construction area: switches, routers, central gateways, serial port servers, and NLE-4150.

**Note: The relay actuating devices used must be installed within each subsystem area.**

**Task requirements:**

- According to the "Equipment Area Layout Map", install the corresponding equipment in the network link system area to build a local area network. It is required that the network devices within the local area network can access the cloud platform。
- Configure the router according to "Appendix: Router Configuration Table".
- Assign the IP addresses of each network device according to the "Appendix: Device IP Address Table".

**After completing the above tasks, please follow these steps:**

- Take A screenshot of the router network Settings interface and save it as A-1-1.jpg.
- Take A screenshot of the router's local area network Settings interface and save it as A-1-2.jpg.
- A screenshot of the scanning result with the IP scanning tool (the IP address should at least show: gateway, serial server, server, workstation), and be

saved as A-1-3.jpg.

- Open the browser, take A screenshot of the first interface of the cloud platform, and save it as A-1-4.jpg.

## 2. Ranch breeding monitoring system

In modern chicken farms, light sensors act like sharp "eyes", precisely monitoring every ray of light and automatically adjusting the on and off of the lights according to the different growth cycle needs of laying hens. The noise sensor remains vigilant at all times. Once it detects an abnormally high decibel (greater than 70 decibels) sound, it promptly alerts the system to protect the chicken flock from being disturbed. The temperature and humidity sensor acts like an indefatigable "guardian", tracking the changes in temperature and humidity in the environment in real time. When the data shows that the temperature is too high (above 30℃) and the air is stuffy, the fan starts immediately. A strong airflow circulates in the chicken coop, dispelling the summer heat and reducing humidity, creating a comfortable and healthy growth environment for the chicken flock. Ensure that every chicken can thrive under scientific and intelligent management.

**Task requirements:**

- Complete the selection of hardware equipment for this system and install it in the corresponding area.
- Complete the configuration of various sensors and actuators on the central gateway to achieve real-time data reporting to the cloud platform.
- When the illuminance is less than 100Lux, the light is turned on; otherwise, it is turned off.

- Create automated strategies on the cloud platform to implement the relevant functions of the system description.
- Create an application named "Ranch Breeding Monitoring System" on the cloud platform, and require a reasonable and aesthetically pleasing interface layout.
- All actuators are required to be equipped with relays in accordance with the specifications.

**After completing the above tasks, please follow these steps:**

- Take screenshots of the real-time data pages of each device that can be seen in the central gateway data monitoring page and save them as A-2-1.jpg.
- Take A screenshot of the configuration interface of the central gateway configuration data sent to the cloud platform and save it as a-2-2.jpg.
- Take A screenshot of the interface on the cloud platform where you can view the real-time data of each sensor and the real-time status of the actuator device, and save it as A-2-3.jpg.
- Take A screenshot of the configuration result interface of the lighting automatic control in the strategy management interface of the project generator and save it as a-2-4.jpg.
- Take A screenshot of the application design result interface of the "Ranch Breeding Monitoring System" and save it as a-2-5.jpg.

## 3. Intelligent escape system

In the fire safety system of large buildings, the intelligent escape system builds a multi-level protection framework relying on professional equipment such as smoke

detectors and warning lights. When the smoke detector detects that the environmental smoke concentration exceeds the preset threshold, it will immediately send a fire alarm to the central control center. After receiving the alarm signal, the central control center immediately activated the emergency response mechanism: the warning light group quickly switched to the alarm mode. The electric push rod rapidly extends according to the control instructions, pushing the escape passage door to open. It works in conjunction with the limit switch to collect real-time data on the door's position. Once the door is fully opened, it feeds back a confirmation signal to the control system.

During the personnel evacuation process, the carbon dioxide transmitter continuously monitors the gas concentration in the escape passage. Once it detects that the carbon dioxide concentration exceeds the safety standard, it will send a rescue request to the central control center. In addition, the microswitch installed on the door frame can manually trigger the electric push rod. If an abnormal closure of the door is detected, the electric push rod will be immediately triggered to re-perform the door opening operation, ensuring that the evacuation passage remains unobstructed throughout.

**Task requirements:**

- Complete the selection of hardware equipment for this system and install it in the corresponding area.
- All sensors and actuators are configured on the central gateway, and the data is reported to the cloud platform in real time.
- Create automated strategies on the cloud platform to implement the relevant functions of the system description.
- Create an application on the cloud platform named "Intelligent Escape

System", which is required to display the carbon dioxide data of the past hour in a line chart, with a reasonable and aesthetically pleasing interface layout.

**After completing the above tasks, please follow these steps:**

- Take screenshots of the real-time data of each device on the central gateway device monitoring page and save them as A-3-1.jpg.
- Take screenshots of the real-time data of each device on the cloud platform and save them as A-3-2.jpg. The screenshots should show the current values of each sensor and actuator device.
- Take A screenshot of the strategy editing interface and save it as a-3-3.jpg. The screenshot should show that the configuration of the smoke detector and the alarm light device is linked.
- Take A screenshot of the strategy editing interface and save it as A-3-4.jpg. The screenshot should show that the configuration of the smoke detector is linked with the electric push rod device.
- Take A screenshot of the strategy editing interface and save it as a-3-5.jpg. The screenshot should show that the configuration of the microswitch and the electric push rod device is linked.
- Take A screenshot of the application design result interface of the "Intelligent Escape System" and save it as a-3-6.jpg.

# 4. Software prototype design

Use Axure prototyping software to complete the prototype design based on the content shown in the effect drawing.

**Task requirements:**

- Contestants use the materials provided in the competition materials and refer to the design renderings to complete the prototype design.
- Complete the corresponding page linkage functions based on the effect drawing.
- After the design is completed, the project needs to be generated into an HTML page.

**After completing the above tasks, please follow these steps:**

- After completing the above functions, please save the generated Axure project file as "Protodesign.rp".
- Package the generated HTML page into a compressed file and save it as "Prototype Design HTML.rar".

# Module B: IoT Engineering Maintenance and Optimization(20 Points)

Note: According to the description requirements of each subsystem, complete the implementation and deployment of the corresponding system. The result files of this module need to be saved to the "D:\Submit Materials\Module B" folder on the server computer. At the same time, the entire folder should be copied to the root directory of the USB drive uniformly distributed by the event. After the competition, this USB drive will be submitted as the competition result.

## 1. Windows system maintenance

**Task requirements:**

- To ensure the security of the server computer, please enable the account password security policy. The password must be at least 8 characters long and can be used for up to 30 days.
- Set the account to be locked for 5 minutes when the user logs in incorrectly 5 times.
- Complete a system performance test on the server computer.
- Query the list of items that start automatically at startup in the system registry.

**After completing the above tasks, please follow these steps:**

- Take a screenshot of the result interface of setting the account password policy that meets the requirements and save it as B-1-1.jpg.
- Take a screenshot of the interface of the policy editing for setting user login errors that meet the requirements and locking them, and save it as B-1-2.jpg.

- Take a screenshot of the report interface of the performance test and save it as B-1-3.jpg.
- A screenshot of the auto-startup item list interface of the registry at startup. The complete registry form and the startup item list should be visible in the screenshot. Save the screenshot as B-1-4.jpg.

## 2. Ubuntu system maintenance

In the Internet of Things (iot) system, some security issues usually occur. As an iot engineer, it is necessary to configure the system securely. Please carry out security maintenance work on the virtual machine Ubuntu operating system deployed on the server computer, and log in to the system with your account and password to complete the specified functional configuration.

**Task requirements:**

- Log in to the Ubuntu system and set the network IP address according to the "Device IP Address Table" in the appendix.
- Use the command in the terminal to create a folder named BRICS in the current directory.
- Use commands in the terminal to query the information of the installed packages.
- Implement the use of the iptables command in the Ubuntu system to forward the traffic from port 80 to port 8080, and perform the same forwarding for the local loopback interface (OUTPUT chain).
- Configure a scheduled task in the Ubuntu system to clean up the BRICS directory every two hours.

**After completing the above tasks, please follow these steps:**

- Log in to the Ubuntu system via SSH, take a screenshot of the successful login interface, and save it as B-2-1.jpg.
- To query the network address configuration results in the terminal using commands, please take a screenshot of the query result interface (the IP information to be configured should be visible in the screenshot), and save it as B-2-2.jpg.
- Take a screenshot of the interface for creating the BRICS folder using the command and save it as B-2-3.jpg.
- Take a screenshot of the iptables command that forwards the traffic from port 80 to port 8080 and performs the same forwarding for the local loopback interface (OUTPUT chain). Save the screenshot as B-2-4.jpg and ensure that the iptables command and the successful execution result are visible.
- Take a screenshot of the scheduled task list in Ubuntu and save it as B-2-5.jpg. The screenshot shows that the scheduled configuration includes a scheduled task to clean up the /BRICS directory every two hours.

# 3. Application and maintenance of data-assisted software

MySQL plays a core role in data management in Internet of Things (IoT) systems, especially demonstrating significant advantages in scenarios with medium to low concurrency and structured data. Contestants complete the deployment, application and maintenance of the MySQL database according to the task requirements.

**Task requirements:**

- Log in to the Ubuntu system and create a folder named "MySQL" in the

current directory. By transferring the provided "mysql.tar.gz" to the MySQL folder, use the command to decompress the file.

- Complete the installation of the MySQL database, log in to the MySQL database in the terminal, and import the "BRICS" database given in the data.
- Write an sql statement to query which country has the most students in the student table and display the names and genders of all students in that country. The query results should be sorted in ascending order by name. The competition materials provide a result reference interface for the contestants to use.

**After completing the above tasks, please follow these steps:**

- Take a screenshot of the interface where you successfully logged into the MySQL database using commands in the terminal and save it as B-3-1.jpg.
- Take a screenshot of the page where you import the database named "BRICS" into MySQL using the command in the terminal and save it as B-3-2.jpg.
- Take screenshots of the sql statements written to query the results that meet the requirements, and save the screenshots as B-3-3.jpg. The screenshots should show the query statements and the corresponding query results.

# 4. Hardware equipment debugging and maintenance

**Task requirements:**

- It is required to debug whether the communication light of the multi-in-one sensor is normal through the serial port assistant.
- Send the command "FA 01 00 FE" to control the flashing of the

communication light.

- Sending the command "FA 01 01 FE" can control the communication light to go out.

**After completing the above tasks, please follow these steps:**

- Take a screenshot of the result interface of the serial port assistant sending "FA 01 00 FE" and save it as B-4-1.jpg.
- Take a screenshot of the result interface of the serial port assistant sending "FA 01 01 FE" and save it as B-4-2.jpg.

# Module C: IoT Engineering Application Development and Debugging(20 Points)

Note: According to the description requirements of each subsystem, complete the implementation and deployment of the corresponding system. The result file of this module should be saved to the "D:\Submit Materials\Module C" folder on the server computer. At the same time, all copies of this folder should be made to the root directory of the USB drive uniformly distributed by the event. After the competition, this USB drive will be submitted as the competition result.

## 1. OpenHarmony Low-power Monitoring System

This module uses the HarmonyOS Light Device (STM32 M4). It is required to write code in the provided engineering source code of the HarmonyOS Light Device to realize the acquisition and conversion of environmental photoresistors, the numerical calibration of the matrix keyboard, and the remote monitoring and remote control of the pan-tilt. It also displays the network connection status, the IP address of the cloud platform domain name resolution, and the light intensity value on the OLED, and realizes the control of local and remote low-power screen-off solutions.

**Task requirements:**

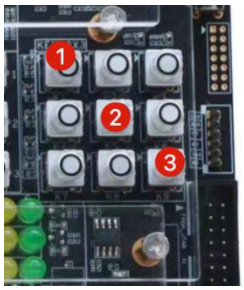
- Set up the HarmonyOS development environment in the Ubuntu system. Request to engineering project code on the source code of the device/board/newland/nlef407/example/lowpowr_system/directory.
- Build a new device named "Intelligent Low-Power Environmental Monitoring System" on the cloud platform, and create sensors and actuators in the device.

It is required that through these sensors and actuators, the intensity of environmental light can be monitored in real time and the device can be set to enter low-power mode. (Note: When creating a device on the cloud platform, in addition to the required parameters, other parameters can be customized and filled in. As long as the final effect can be achieved, it is fine.

- OLED displays the network connection status, the IP address of the domain name resolution of the cloud platform, light intensity data, and the current low-power mode. Among them, the network connection status and the connection status with the cloud platform are updated every 3 seconds. The light intensity data needs to be collected 20 times after the light intensity is collected, and then the average value is calculated and updated. The current low-power mode setting status of the device only needs to be updated when it changes. The OLED displays for each line are as follows:

| | |
|---|---|
|  | a).NetSta indicates the network connection status. 1 means connected and 0 means not connected<br>b).Cloud ip is the IP address of the cloud platform, such as 192.168.1.123<br>c).Light represents the value of light intensity<br>d).lopowrmode is the current low-power mode of the device, either 0, 5 or 10. 0 indicates that the low-power screen remains on constantly when turned off, and 5 indicates that the screen is off for 5 seconds and then turns on for 5 seconds in a cycle. 10 indicates that the screen will turn off for 10 seconds and then light up for 10 seconds in a cycle. |

- Connect to the network, either Ethernet or WIFI is acceptable.
- The on-board photosensitive sensor module values are collected every 800 milliseconds.
- The device must be capable of entering low-power mode via the matrix keyboard:




a).Realize the development of matrix keyboard functions;

b).Press the 1 key to turn off the low-power screen constantly on. Press the 2 key to turn off the screen for 5 seconds and then light it up for 5 seconds in a cycle. Press the 3 key to turn off the screen for 10 seconds and then light it up for 10 seconds in a cycle.

c).Press keys 1, 2, and 3 on the OLED screen in LowPowrMode to display 0, 5, and 10 respectively.

- The device must be capable of being remotely triggered to enter low-power mode.
- The cloud platform sends control command strings 0, 5, or 10 to achieve low-power switching between different states. 0 indicates that the low-power screen remains on constantly when turned off, and 5 indicates that the screen is off for 5 seconds and then turns on for 5 seconds in a cycle. 10 indicates that the screen will turn off for 10 seconds and then light up for 10 seconds in a cycle.

- Cyclic reporting: Report the light intensity data to the cloud platform once every 5 seconds.
- The HarmonyOS light device automatically runs the "Intelligent Low-Power Environmental Monitoring System" program upon startup.

**After completing the above tasks, please follow these steps:**

- Take a screenshot of the written program and box out the program segments that can reflect the following requirements:

  1.The data of the photosensitive sensor is collected every 800 milliseconds and saved as C-1-1.jpg.

  2.Matrix keyboard scan triggers key code, save as C-1-2.jpg.

  3.Every 3 seconds, OLED, refresh the network connection status and the IP address with the cloud platform, and save it as C-1-3.jpg.

  4.Report the light intensity data to the cloud platform every 5 seconds and save it as C-1-4.jpg.

  5.Cloud platform control command processing, save as C-1-5.jpg.

- Open the real-time data function of the cloud platform, take a screenshot of the entire page, and save it as C-1-6.jpg. The screenshot shows the control of lighting and low-power mode switching.
- Enter the "Historical Sensor Data" page of the relevant devices on the cloud platform, take a screenshot of the entire page, and save it as C-1-7.jpg. The screenshot shows the historical data of light intensity.
- Enter the "Historical Online Data" page of the relevant device on the cloud platform, take a screenshot of the entire page, and save it as C-1-8.jpg. The screenshot shows the online historical records of the HarmonyOS light device.

- After the development is completed, burn the program onto the HarmonyOS light device, then disconnect the power supply and wait for the referee's judgment.
- In the "Submit Directory\Module C", create a new folder named "Intelligent Low Power Environmental Monitoring System", and under this folder, create the "Source Code", "Program Firmware" and "Screenshot" folders. Copy the lowpowr_system folder to the "Source Code" folder. Copy the program firmware OHOS_Image.bin to the "program firmware" folder and copy the previous screenshot to the "screenshot" folder. Be sure to keep the data complete for future reference.

## 2. Data center monitoring system

In the data center, temperature and humidity transmitters are closely deployed near the server cabinets to monitor the operating environment of the equipment in real time. Once the temperature in a certain area exceeds the safety threshold (30℃), the RGB light strip immediately emits a strong red light (otherwise, the RGB light strip returns to its default state). At the same time, on the visualization management large screen in the data center, the corresponding area is highlighted in red, allowing operation and maintenance personnel to quickly locate the overheated position and promptly initiate heat dissipation measures. Infrared beam devices are installed around important equipment and at the entrances and exits of the computer room. When unauthorized personnel approach the sensitive area and trigger the infrared sensor, the RGB light strips turn flashing white, and at the same time, the security management system issues an alarm to ensure the safety of equipment and data stability in the data center.

**Task requirements:**

- Create a new Python application and utilize the documentation, images and other resources provided in the competition materials to develop the program and simulate the control of the device by the application software.
- Install the Python development environment and PyCharm software on the server computer.
- The competition materials provide the necessary materials, development packages and renderings for this question for the contestants to use.
- It is required to realize the real-time collection of temperature, humidity and infrared beam state data from the cloud platform by writing Python programs.
- The RGB light strips are controlled to emit strong red light based on temperature through a Python program.
- The RGB light strips are controlled to turn into flashing white based on infrared beam through a Python program.
- When the temperature exceeds the threshold or the infrared beam is triggered, the program interface should respectively display the effects of "Area Temperature Alarm" and "Illegal Intrusion Alarm".
- The RGB light strip is in the off state by default.

**After completing the above tasks, please follow these steps:**

- Package the project into an exe executable file and save it as "C2.exe".
- Make sure that "C2.exe" can run normally on the server computer during the scoring stage.
- Package the project source code into a compressed file and save it as "C2.rar".

# 3. Intelligent anti-theft snapshot system

With the advancement of smart city construction, intelligent security systems have become an important guarantee for community safety. A certain science and technology park plans to upgrade its security system and requires the design of an intelligent anti-theft snapshot system to achieve the functions of identifying, alerting and recording unauthorized personnel.

The system should create identity cards for authorized personnel through a UHF desktop card issuer, input personal information and permission levels, and store them in the database. At the entrance of the park, human infrared sensors monitor the entry and exit status of people in real time, trigger high-frequency card readers to read the identity card information, and at the same time, intelligent facial recognition cameras synchronously collect facial images for feature comparison. The system needs to automatically verify the identity of personnel: when an unauthorized person is detected, it immediately triggers the three-color alarm light to sound a red alarm, and records the time, location and on-site captured images. Authorized personnel will be allowed to pass according to their permission levels.

**Task requirements:**

- Create a new Python application and utilize the documentation, images and other resources provided in the competition materials to develop the program and simulate the control of the device by the application software.
- The number of people entering is counted through human body infrared sensors.
- Use a desktop card issuer to create two identity cards (one temporary identity

and one official identity), and mark each identity card with stickers.

- When an unauthorized card is used, the camera is triggered to take a snapshot and the three-color light turns red as an alarm.
- When a temporary identity card is used, it triggers the camera to take a snapshot and triggers the yellow alarm of the three-color light.
- Realize that when using the official identity card, only the green light of the three-color light is triggered.

**After completing the above tasks, please follow these steps:**

- Package the project into an exe executable file and save it as "C3.exe".
- Make sure that "C3.exe" can run normally on the server computer during the scoring stage.
- Package the project source code into a compressed file and save it as "C3.rar".

# Module D: Work organization and management and professionalism(5 Points)

During the project construction process, it is necessary to safely and reliably select and use tools, correctly choose equipment, install it stably, evenly arrange the equipment components, align the equipment, have equal spacing, and make it neat and beautiful. The wiring is reasonable and all the wires are installed in cable trays. After the construction is completed, it is necessary to clean the floor, organize the tables and restore the tools and equipment.

**Task requirements:**

- Cleaning of the floor, tables and other areas in the competition venue.
- The tools used are neatly restored, the equipment is neatly arranged, and the equipment carrying cases are neatly organized, etc.
- The equipment at the workstations is neatly installed, the equipment components are evenly arranged, and the wiring is reasonable and aesthetically pleasing, etc.
- Technical specification requirements related to teamwork and safe production operations, etc.

# Appendix: Router Configuration Table

| Parameter information | |
|---|---|
| Static IP of the WAN port of the wireless router | 192.168.0. [Workstation Number] |
| SSID of wireless router | Close |
| Wireless router gateway IP | 192.168.0.254 |
| Cloud platform address | 192.168.0.138 |
| Docker private repository | 192.168.0.139 |
| Cloud platform login account and password | Register and authorize yourself according to the key information form distributed on site |
| The TCP server port number of the cloud platform | 8600, 8700, 8800 |
| Cloud platform MQTT server port number | 1883 |
| Cloud platform Modbus server port | 5500, 15000 |
| IP address of the LAN port of the wireless router | 172.20.[Workstation Number].1 |

# Appendix: Device IP Address Table

| Device Name | IP Information |
|---|---|
| Server computer | 172.20.[Workstation Number].2 |
| Workstation computer | 172.20.[Workstation Number].3 |
| Central gateway | 172.20.[Workstation Number].56 |
| Serial port server | 172.20.[Workstation Number].200 |
| Webcam | 172.20.[Workstation Number].13 |
| Virtual Machine (Ubuntu | 172.20.[Workstation Number].17 |
| Other devices that require IP Settings | Set the IP by yourself |

# Appendix: Parameters of Cloud Platform Equipment

| No. | DeviceName | ApiTags | No. | DeviceName | ApiTags |
|---|---|---|---|---|---|
| 1 | Temperature | m_temp | 11 | Call button | m_ring |
| 2 | Humidity | m_hum | 12 | Confirm button | m_comfirm |
| 3 | Lighting | m_light | 13 | The three-color light is red | m_redlight |
| 4 | air conditioner | m_thermostat | 14 | Three-color light green | m_greenlight |
| 5 | lamp | m_lamp | 15 | Three-color light yellow light | m_yellowlight |
| 6 | Electric curtains | m_pushrod | 16 | Carbon dioxide | f_co2 |
| 7 | Curtain opening button | m_open | 17 | Noise | f_noice |
| 8 | Curtain close button | m_close | 18 | Smoke sensor | m_smoke |
| 9 | Electronic fence | m_fence | 19 | Human body sensor | m_body |
| 10 | Alarm light | m_alarm | 20 | Exhaust fan | m_fan |

Note: If the hardware device does not appear in this table, set the parameters by yourself