

An intelligent plant growth monitoring system based on ESP32 and IoT resource explorer platform

Pinze Li^{1,a}, Xinxian Deng^{2,b,*}

¹International Division of Hubei Wuchang Experimental High School, 48 Huoju Road, Wuchang, Wuhan, China

²Wuhan University of Technology, Hongshan, Wuhan, China

^alameloli@163.com, ^b334941@whut.edu.cn

*Corresponding author

Keywords: ESP32; Internet of Things; humidity sensor; light sensor; Tencent Cloud

Abstract: Plant growth status monitoring is an important link in the process of plant cultivation. The traditional monitoring methods mainly rely on human labor. This paper focuses on the use of soil moisture sensors and light sensors to detect the plant growth status, and studies the way that can view the plant growth information in real time through the mobile phone program to achieve the purpose of intelligent detection. Considering the characteristics of convenient network transmission and large-capacity multi-channel, ESP32 is used as the main board for development, and a multi-sensor fusion perception device based on ESP32 is designed, which integrates the functions of data detection, storage, and transmission. At the same time, based on the Tencent Cloud IoT resource explorer platform, a set of programs for receiving sensor information is developed with a completed the panel design. Therefore, the purpose of data exchange between the microcontroller and Tencent Cloud is achieved, which makes it possible to display various plant growth status information on the mobile phone in real time, saving a lot of labor costs.

1. Introduction

With the advent of the information age and the rise of ecological civilization, people gradually pay attention to the development of home environment. Many people choose to cultivate some flowers at home, which can beautify the home environment and purify the air. However, modern people are also busy with work and other things, and may forget to take care of their plants, resulting in their plant death [1]. At the same time, there is also a demand for monitoring the plant growth status in many other scenarios, such as decorative plants in shopping malls, schools, and companies, which need long-term and regular care and cultivation. Therefore, if there is a device that can help detect the plant growth status, it can help save a lot of time and money costs.

The reason why sensors are widely used today is that they can help people detect a lot of data and monitor the environment. As early as 1979, NASA proposed the concept of intelligent sensors. But not until the 21st century, with the commercialization of 5G communication, intelligent sensing technology has developed rapidly and gradually formed industrialization [2]. In order to solve the problem of planting plants mentioned above, sensor technology is used to solve this problem. This

project aims to design and build a sensor to detect the plant condition [3]. The sensor will measure various parameters, such as soil moisture, temperature and light intensity, which are essential for plant growth. The sensor will be designed to be low-cost, easy to use and portable, making it widely used. The sensor is based on the Internet of Things technology, which can connect to the Internet and provide users with real-time data [4]. One of the key components of the intelligent plant growth monitoring system is the central controller, which is responsible for collecting, processing and transmitting data from the sensors. The research of “Design and Implementation of Intelligent Flower Management System” shows the basic working principle of software design and the selection of central controller [5]. The system uses Raspberry Pi as the central controller, which is a microcomputer based on Linux operating system, with only the size of an ID card. It has the characteristics of small size, complete functions and low price. Due to the powerful function of Raspberry Pi, it can perform all the functions of a PC, and can be used as a central controller.

In order to achieve accurate and real-time monitoring of plant growth status, this paper designs a multi-sensor fusion perception device based on ESP32, which integrates the functions of data detection, storage and transmission, and develops a program to display data and remind users [6]. In general, this sensor will provide residents with a practical tool to monitor their plant condition, so that they can get accurate information about the plants they plant, and take care of their plants accordingly.

2. Overall system scheme design

The production method of the plant status monitoring system includes hardware design, user interface design and response signal processing. This paper chooses ESP32 as the main board, which is a low-cost, low-power system-level chip that integrates Wi-Fi [7] and dual-mode Bluetooth functions. It adopts Tensilica Xtensa LX6 or LX7 dual-core or single-core microprocessor, which has high performance and flexibility. It supports Bluetooth 4.2 and BLE protocols, which can achieve low-power wireless communication, and also supports protocols such as GAP, GATT, ATT, SMP, L2CAP, etc. [8]. In this project, its WIFI module is mainly used to realize network communication with Tencent Cloud.

ESP32 integrates a variety of peripherals and interfaces, such as GPIO, ADC, DAC, SPI, I2C, I2S, UART, PWM, SDIO, CAN, etc., which can easily connect various sensors and modules. Therefore, for the above objectives, ESP32 can meet the performance requirements well while simplifying the design steps. The hardware layout of ESP32 is shown in Figure 1.

The hardware circuit is divided into different functional modules, including the main control module, the sensor module, the data acquisition module and the data transmission module. The hardware circuit mainly realizes the detection and conversion of sensor signals, as well as signal transmission functions. At the same time, since most of the project scenarios are indoors, sensors with high sensitivity and accuracy are selected for measurement. The overall design scheme of the system is shown in Figure 2.



Figure 1: ESP32 layout diagram

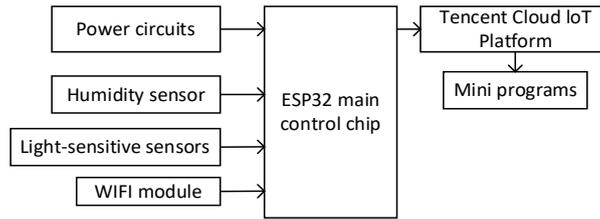


Figure 2: Overall design scheme block diagram

3. Hardware Design

The hardware design part includes hardware selection, circuit connection, embedded system integration and other parts. Firstly, hardware selection is performed, and suitable hardware modules are selected, including soil moisture sensor, light intensity sensor, etc. Secondly, according to the interface and function of the hardware module, a reasonable circuit connection scheme is designed, including power management, signal transmission, control logic, etc. Finally, embedded system integration is carried out, and embedded software is written on the ESP32 development board to realize the functions of soil moisture and light intensity collection, data processing and remote display.

3.1. Selection and connection of soil moisture sensor

The DFRobot SEN0193 is selected as soil moisture sensor in this project, which supports a wide voltage working environment of 3.3 ~ 5.5 V and can be directly compatible with ESP32. It uses the principle of capacitive induction to detect soil moisture. The circuit is not directly exposed to moist soil under the protection of insulating paint, which solves the problem of easy corrosion of resistive sensors. Compared with that, it has a longer service life. Its circuit diagram is shown in Figure 3. The structure ensures that it will not be affected by the salt and impurities in the soil, making the results of measurement more accurate and stable. It will not change the chemical properties of the soil and is harmless to plants. It does not need to be cleaned and maintained frequently, making it more convenient to use.

It uses a simple capacitor, whose two plates are the sensor itself and the surrounding soil. When the sensor is inserted into the soil, the moisture in the soil will affect the dielectric constant of the capacitor, thus changing the capacitance value of the capacitor. By measuring the capacitance value of the capacitor, the moisture content in the soil can be reflected.

When it comes to the connection of the capacitive soil moisture sensor, the monitored value is chosen to be called through ADC1 channel 7, which corresponds to the GPIO35 socket on the ESP32 board. Therefore, the IN port is connected to the GPIO35 port of the board. Then, the VCC port is connected to the 5V power supply port of the single board, and the GND port is connected to the GND port of the single board. Its soil moisture sensor connection diagram is shown in Figure 4.

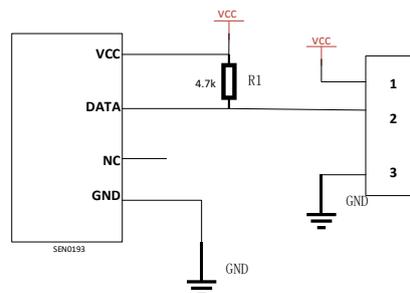


Figure 3: Soil moisture sensor circuit diagram

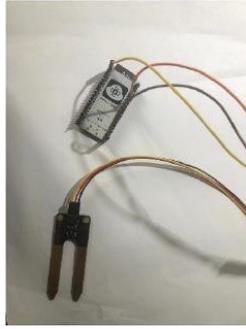


Figure 4: Soil moisture sensor connection diagram

3.2. Selection and connection of light sensor

The GL5528 is selected as photoresistor, which is a NTC (negative temperature coefficient) type photoresistor, whose resistance value changes with the change of light intensity. It supports a working voltage of 3.3 ~ 5 V and can be directly compatible with ESP32. The spectral response range of the resistor is 540 nm ~ 610 nm, which is suitable for the detection of visible light. It has the advantages of simple structure, easy to use, no need for complex circuits and calibration, etc. At the same time, it has fast response speed, good stability, and long life. It is also low-cost, easy to obtain and replace.

Its working principle is to use the characteristic of semiconductor materials to produce photoelectric effect under light, that is, when photons hit the semiconductor, they will excite electrons and holes. When an external voltage is applied to both ends of the semiconductor, electrons and holes will move in opposite directions, forming a current. When the light intensity increases, the generated electrons and holes also increase, resulting in an increase in current and a decrease in resistance. By measuring the resistance value or current value, the change of light intensity can be reflected. The circuit connection diagram is shown in Figure 5.

About the connection of the photoresistor, it has the same basic wiring principle with the soil moisture sensor, but to read the monitoring value through ADC1 channel 7, it is connected to the GPIO32 socket on the ESP32 board. The next steps are similar to connecting the soil moisture sensor. The IN port is connected to GPIO32. The VCC port is connected to the 3.3V power port of the board (because the 5V port is already occupied by the soil moisture sensor), and the GND port is connected to the GND port of the board. The connection diagram of optical sensor is shown in Figure 6.

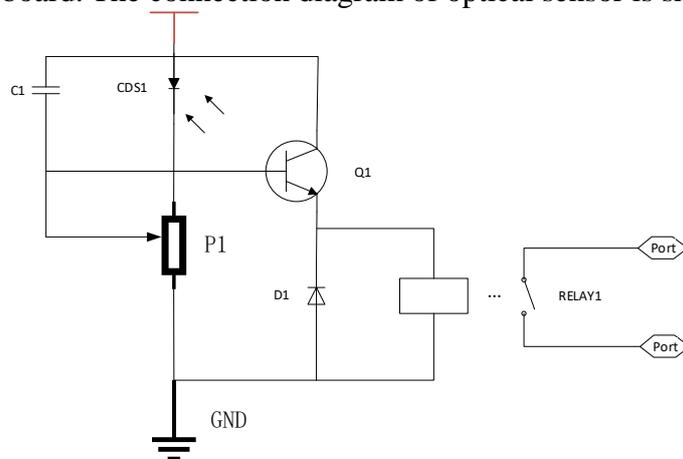


Figure 5: Photosensitive resistor circuit diagram

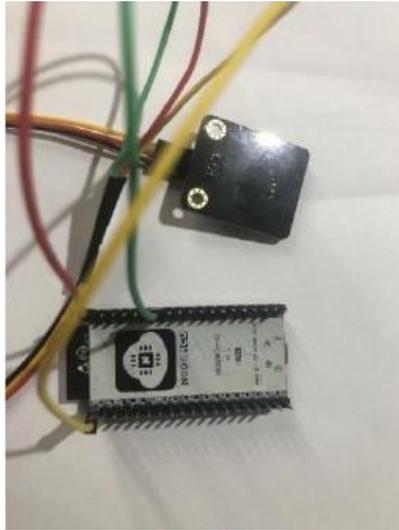


Figure 6: Connection diagram of optical sensor

The specific parameters of the soil moisture sensor and the light sensor are shown in Table 1.

Table 1: Technical indicators

	Working voltage	Output	PCB size
Soil moisture sensor	3.3V-5V	Analog signal output	6.3cm*2.3cm
Light sensor	3.3V-5V	Analog signal output	3cm*2.5cm

After completing all the preparations and having a clear understanding of the performance of each module, the project is started to be build and the program can be written.

4. Program Design

4.1. Main program design

The main program is designed to connect ESP32 microcontroller with Tencent Cloud platform to achieve the monitoring and remote control of the plant growth status. The basic logic of the main program includes initializing NVS, TCP/IP, event loop, Wi-Fi, IoT Hub and other necessary components, to prepare for the subsequent connection and communication. On the one hand, the device information, including product ID, device name, device key, etc. is set for identity verification and data encryption with the Tencent Cloud platform. On the other hand, the plant parameters, including temperature, humidity, light, light level, etc. are registered for data exchange and control instruction reception with the Tencent Cloud platform.

What's more, the callback function of the plant parameters is installed to get and set the values of the plant parameters. According to different parameter IDs, it can call the corresponding sensor function or actuator function. Through the smart networking method, the ESP32 can be connected to the local wireless network and can get the IP address and network status. Meanwhile, through the MQTT protocol, the ESP32 can be connected to the Tencent Cloud platform to subscribe and publish the corresponding topics, and then realize the function of data reporting and instruction issuing. As for the loop to handle events, the values of the plant parameters are periodically read and updated, which will be packaged into JSON format and sent to the Tencent Cloud platform through MQTT messages[9], while the control instructions from the Tencent Cloud platform can also be received and parsed to execute the corresponding operations.

The main program flow chart is shown in Figure 7:

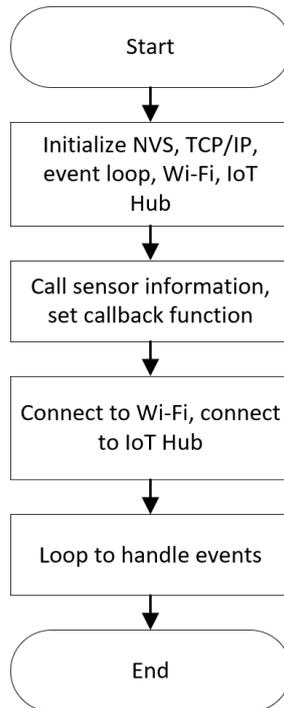


Figure 7: Main program flowchart

The main program code:

```

void app_main(void)
{
    ESP_ERROR_CHECK(nvs_flash_init());
    ESP_ERROR_CHECK(esp_netif_init());
    ESP_ERROR_CHECK(esp_event_loop_create_default());
    ESP_ERROR_CHECK(esp_qcloud_wifi_init());
    ESP_ERROR_CHECK(esp_qcloud_iothub_init());
    ESP_ERROR_CHECK(esp_qcloud_device_add(PRODUCT_ID, DEVICE_NAME,
    DEVICE_SECRET));
    ESP_ERROR_CHECK(esp_qcloud_device_add_param("temperature",
    QCLOUD_VAL_TYPE_NUMBER));
    ESP_ERROR_CHECK(esp_qcloud_device_add_param("moisture",
    QCLOUD_VAL_TYPE_NUMBER));
    ESP_ERROR_CHECK(esp_qcloud_device_add_param("light",
    QCLOUD_VAL_TYPE_NUMBER));
    ESP_ERROR_CHECK(esp_qcloud_device_add_param("light_high",
    QCLOUD_VAL_TYPE_BOOLEAN));
    ESP_ERROR_CHECK(esp_qcloud_device_set_callback(plantGetParam, plantSetParam));
    ESP_ERROR_CHECK(esp_qcloud_wifi_start_smartconfig());
    ESP_ERROR_CHECK(esp_qcloud_iothub_start());
    while (true) {
        esp_qcloud_iothub_yield(1000);
    }
}
  
```

4.2. Sensor signal calling program

The First step of the program is call various sensors, retrieve the values detected by the calling pins, including introducing relevant libraries and using some built-in functions or macros. Then, Define the pins connected to the sensors, as well as some other constants or variables, such as the maximum and minimum resistance values of the photoresistor, the maximum and minimum values of the light intensity, the storage variable of the light intensity, etc. Set the serial communication baud rate, so that the data can be printed on the serial monitor. Set the pins to input mode, so that their analog values can be read. The program is as follows:

```
/* Initialize the sensors */
static void initSensors()
{
    // Set GPIO16 to 3.3V
    gpio_set_direction(GPIO_NUM_16, GPIO_MODE_OUTPUT);
    // 1 is high level, pseudo low level
    gpio_set_level(GPIO_NUM_16, 1);
    adc1_config_width(ADC_WIDTH_BIT_12);
    // Humidity sensor GPIO35, measurement range 150mV~2456mV
    adc1_config_channel_atten(ADC1_CHANNEL_7, ADC_ATTEN_DB_11);
    // Light sensor GPIO32, measurement range 156mV~2450mV
    adc1_config_channel_atten(ADC1_CHANNEL_4, ADC_ATTEN_DB_11);
}
```

Next, use the formula to convert the monitored values into common units. The value of the soil moisture sensor is $*100 / 4095 + 0.5$, which gives the percentage of humidity.

```
/* Get the moisture value */
static int getMoisture()
{
    int moistureRaw = 0;
    moistureRaw = adc1_get_raw(ADC1_CHANNEL_7);
    // Convert the raw value to percentage
    return (int) (moistureRaw * 100 / 4095 + 0.5);
}

/* Get the light value */
static int getLight()
{
    int lightRaw = 0;
    lightRaw = adc1_get_raw(ADC1_CHANNEL_4);
    // Return the raw value
    return lightRaw;
}
```

4.3. Interaction program design

4.3.1. Tencent Cloud environment configuration and display panel design

After downloading the ESP-IDF environment, the appropriate version is chosen to build, and then the ESP Tencent Cloud code can be extracted from it. The Tencent Cloud design and distribution schematic is shown in Figure 8.

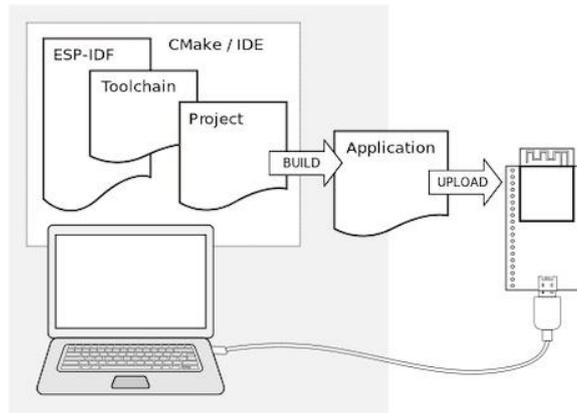


Figure 8: Design and distribution of Tencent Cloud

After completing the IDF environment setup, the ESP Tencent Cloud code from the command board is set to complete the development environment configuration [10]. The specific steps are as follows:

(1) Enter the Tencent IoT browser, build a new program, and get the corresponding sensor properties. As shown in Table 2:

Table 2: Sensor properties

Function Type	Function Name	Identifier	Data Type	Read Write Type	Data Definition
attribute	illuminance	light	INTEGER	read only	Value range: 0-4095 Step size: 1 Initial value: 0 Unit:LUX
attribute	soil moisture	moisture	INTEGER	read only	Value range: 0-100 Step size: 1 Initial value: 0 Unit:%RH

(2) Choose the device development method. This project uses the SDK-based development method.

(3) Perform interactive development, including panel configuration and connection method. For the panel design, set the values of the light sensor and soil moisture sensor as two modules with different units, such as the unit of soil moisture is percentage, and the unit of light is lux [11].

After the panel design is completed, a QR code can be generated. The panel data can be transferred to the phone by opening the WeChat mini program “Tencent Connect”.

4.3.2. Cloud application invocation program

For the back-end cloud application, this project referred to the sample code included in the ESP Tencent Cloud and made some modifications to form the cloud application of the project. These codes include callbacks, which are used to handle commands received from Tencent Cloud, assign network programs, initialize sensors and timer callbacks. The code is shown below:

```

/* Get the parameter value of the plant */
static esp_err_t plantGetParam(const char *id, esp_qcloud_param_val_t *val)
{
    if (!strcmp(id, "temperature"))

```

```

{ // If the parameter is temperature
    val->i = getTemperature(); // Get the temperature value
    ESP_LOGI(TAG, "Report id: %s, val: %d", id, val->i); // Log the report
}
else if (!strcmp(id, "moisture"))
{ // If the parameter is moisture
    val->i = getMoisture(); // Get the moisture value
    ESP_LOGI(TAG, "Report id: %s, val: %d", id, val->i); // Log the report
}
else if (!strcmp(id, "light"))
{ // If the parameter is light
    val->i = getLight(); // Get the light value
    ESP_LOGI(TAG, "Report id: %s, val: %d", id, val->i); // Log the report
}
else if (!strcmp(id, "light_high"))
{ // If the parameter is light_high
    val->b = (bool) (getLight() > 3000); // Get the boolean value of light > 3000
    ESP_LOGI(TAG, "Report id: %s, val: %d", id, val->b); // Log the report
}
return ESP_OK; // Return success}

```

After all the work is done, the detection device is packaged, and a waterproof case is designed to protect the main board from water damage.

By reading relevant materials to know the drought information of plants, a humidity environment reference table is made for reference. The specific parameters are shown in Table 3.

Table 3: Comparison Table of Humidity Environment

Soil moisture	
S.M≤30%	Extremely severe drought
30%<S.M 40%	Severe drought
40%<S.Ms 50%	Moderate drought
50%<S.Ms 60%	Mild drought
60%<S.M	No drought phenomenon

5. Experimental Results and Discussion

5.1. Experimental results

This article designs a multi-sensor fusion perception device based on ESP32, which is used to detect the plant growth environment, including soil moisture, light intensity, etc. At the same time, using the Tencent Cloud platform, it realizes the functions of data storage, transmission, display, control, etc., so that users can view and control the plant growth status in real time through the mobile program.

In order to verify the feasibility and effectiveness of the system, various tests were conducted, and many modifications and optimizations were made on the program, circuit connection, power supply, etc., and finally the monitoring content was displayed on the mobile program. The circuit connection is shown in Figure 9, and the final result is shown in Figure 10.

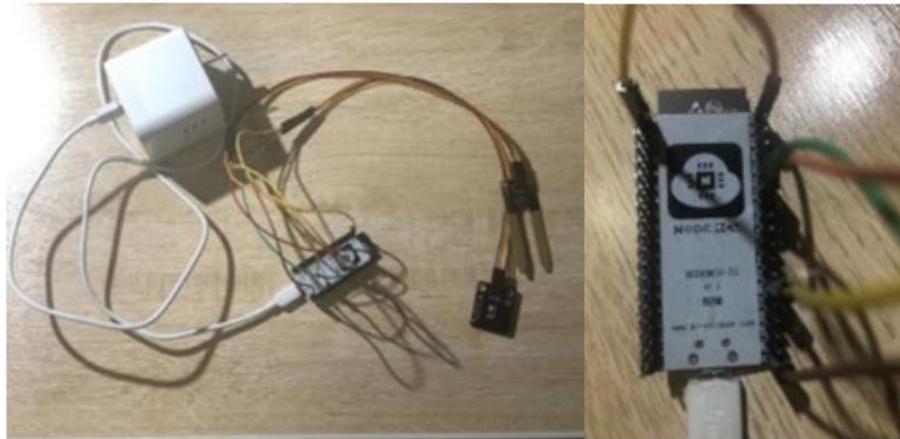


Figure 9: Circuit Connection



Figure 10: Program interface display before and after watering

As can be seen from Figure 10, the mobile app can display the parameters of the plant, such as humidity, light, etc. By comparing the parameter status before and after watering, it can be observed that the humidity has changed significantly, which shows that the device has achieved the function of detection and monitoring well.

By comparing experiments, this paper finds that the system designed has the following advantages:

- (1) The system has a simple structure, low cost, and is easy to deploy and maintain.
- (2) The system uses multi-sensor fusion to improve the accuracy and reliability of data.
- (3) The system uses Tencent Cloud platform, which can realize data security storage, fast transmission, beautiful display, flexible control and other functions.
- (4) The system provides a mobile program interface, which can facilitate users to check and control the growth status of plants anytime and anywhere.

5.2. Discussion

There are some problems in the process of finishing this project. The first problem is the choice of the main board. After comparing with ESP8266, Raspberry Pi Pico and Arduino, ESP32 is chosen as the main board. The pictures of the ESP8266 motherboard and Arduino motherboard are shown in Figures 11 and 12, respectively. Raspberry Pi Pico as a basic control board can accept many programming languages, but cannot get access to the Internet. Arduino's main board has the same problem with Raspberry Pi Pico to connect to the Internet, and it cannot accept Python language. As for ESP8266, it can connect to the network in actual use, but what is needed is a main board that can correspond to two channels, while ESP8266 only has one channel with a small size for memory in

only 2 megabytes. Finally, it is the ESP32 that overcomes all the problems existing in other main boards, and conveniently connected to the Internet or multiple channels with large memory, making it a good choice. Therefore, after many tests, ESP32 is chosen to use.



Figure 11: ESP8266 motherboard

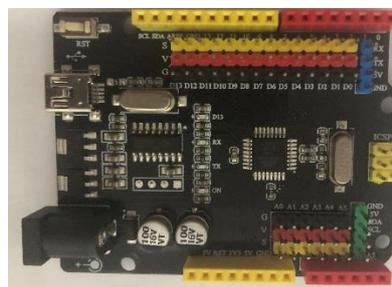


Figure 12: Arduino motherboard

In the initial project, temperature value is planned to be measured, and Raspberry Pi Pico is selected at the beginning of the project. Since this main board is equipped with a temperature sensor to detect the temperature effectively, and the algorithm and publicity can refer to the attached tutorial. However, as mentioned earlier, this main board does not meet the project requirements. Therefore, this main board and the temperature sensor inside are given up.

In the factory state monitoring sensor, the biggest problem is the transmission mode. When deciding to use the Raspberry Pi Pico main board firstly, the transmission mode doesn't be considered carefully. Since the Raspberry Pi Pico component cannot be connected to the network, the use of Bluetooth connection has been tried, and an application using MIT app INVENTOR software has been developed. However, considering the distance limitation and stability problem of Bluetooth transmission, it is not suitable. If the Bluetooth is too far away from the signal segment, the signal will not be received, and even at a certain distance, the signal transmission may be unstable. After replacing the original Raspberry Pi Pico with the ESP32 device currently used, which can connect to the network. In order to overcome the distance limitation, the transmission process will be completed by uploading the signal to Tencent Cloud, and then feedback to the mobile terminal.

In the project, the detection of some data of the plant has come true, but the product can still be more intelligent, for example, adding an expansion board and linking more sensors to monitor other values of the plant, such as temperature, trace elements, etc. Similarly, the irrigation module can also be accessed, which means that processing and analyzing the collected data to calculate the water demand of the plant and irrigate it.

6. Conclusion

Scientific planting of plants cannot be separated from the monitoring of plant growth status. Aiming at the problem of time-consuming and labor-intensive traditional monitoring of plant growth status, this paper designs an intelligent plant growth monitoring system based on ESP32 and Internet

of Things resource manager platform. The main components of the system include two modules: hardware circuit and user interface. The hardware circuit design realizes the signal acquisition function and signal transmission function, and the software design realizes the signal reception, signal processing and human-computer interaction function. Users can observe the light intensity and soil moisture content of plants in real time on the mobile phone screen through WeChat applet, which provides data reference for scientific planting, saves manpower, and has positive guidance significance for better planting of plants.

References

- [1] He, W.X, Li, S.L. (2022) *Design of Intelligent Home Potting System Technology*. *Modern computers*. <http://www.moderncomputer.cn/>.
- [2] Mi, S, Fang, C, Xu, M.S. (2022) *Seize the opportunity to accelerate the development of China's intelligent sensor industry*. *Economic Herald*.
- [3] Wu, G.J, Gao, M.L, Liu, X, Zhang, J.R, Liu, J.W. (2014) *Research and Development of Intelligent Maintenance System for Indoor Plants*. *Tianjin Agricultural Sciences* (02), 105-107.
- [4] Li, S.H. (2018) *Intelligent monitoring system for home potted plants based on STM32 and WIFI technology*. *Wuhan Light Industry University*. DOI:10.27776/d.cnki.gwhgy.2018.000174
- [5] Shen, C, Dai, X.P. (2017) *Design and Implementation of an Intelligent Flower Management System*. *Fujian Computer*, 33 (03): 26-27
- [6] Di, X. Y, Diao, Y. J, Zhu, H. Y, Zhou, R. S, Xie, C. (2023) *Design of Intelligent IoT Table Lamp Based on ESP32*. *Forestry Machinery & Woodworking Equipment*,47-51+57.
- [7] Li, B, Shi, Y. F, Yang, L. (2023) *Design of Indoor Air VOC Content Monitoring System Based on ESP32-E*. *Electronic technique*, 52(04):61-63.
- [8] ESPRESSIF, 'ESP32-DevKitC V4 Getting Started Guide' from *ESP32-DevKitC V4 Getting Started Guide - ESP32 - ESP-IDF Programming Guide latest documentation (espressif.com)(N.D)*
- [9] (2019) *Tencent Cloud releases multiple self-developed products to accelerate the construction of self-developed technology systems*. *Intelligent Manufacturing* (11):40-41.
- [10] Qiao, J. H, Liu, Z.C. (2021) *Design of an Intelligent Greenhouse Monitoring System Based on WeChat Mini Programs*. *Scientific and Technological Innovation* (28):172-174.
- [11] Tan, W.W. (2023) *Design of Smart Home Control Panel System Based on ESP32S3*. *World of Electronic Products*, 30(08):79-81.