Nodemcu-based Low-cost Smart Home Node Design

Tong Qiang[1], Gao Guangling[2], Liu hai*[1],Cao Lina[1] and Wang Han[1]

[1]School of Mechanical, Electrical and Information Engineering, Shandong University (Weihai), Weihai 264209, China
[2]State Grid of China Technology College, Jinan 250002, China

liangLikai@sdu.edu.cn

Abstract. This design proposes a low-cost node solution based on the Wi-Fi wireless sensor network. It aims to solve the problem of the popularity of smart homes due to high costs in real life and is suitable for applications in the smart home field. This solution saves cost, uses SOC solution, and has few peripheral circuits. Nodemcu is used to collect sensor data directly, which saves a separate main control chip, thereby further reducing costs. Various hardware interfaces are provided to facilitate the expansion of different types of sensor functions. The hardware of different functional nodes is designed, and the data is uploaded to a back-end server. Based on the software processing, the data between different nodes is aggregated and uploaded to the cloud platform for display to achieve remote access and control.

1. Introduction

In recent years, the vocabulary closely related to smart homes has been understood by most families, such as smart home local area networks, home gateways and smart devices. Some domestic universities have also begun research and development of smart home appliances systems.S3C2410 series high-performance single-chip microcomputers are used as central control cores to extend the Bluetooth module interface and home gateway solutions for GSM module interfaces, but they do not relate to home node networking solutions.In addition, in a smart home network where a large number of nodes are deployed, the use of the Bluetooth networking technology is complicated and the cost is high. Most domestic smart homes use ZigBee as the inter-node communication module. ZigBee nodes have the advantages of flexible networking and reliable operation. However, the ZigBee protocol stack with high network efficiency needs to be charged and the cost of a single node is high. The gateway is required to convert the network during operation. You can upload data to the Internet. In view of the above situation, we use the Nodemcu chip as the main control and use GSM to upload data to form a low-cost smart home solution.

2. Overall system design

2.1. Overall system framework design

2.1.1.Overview of the overall plan example.The smart home system in this design is based on the ESP8266 chip as a hardware platform, secondary development based on Nodemcu firmware and a gateway control platform through a public cloud server.Through the TCP protocol, the cloud server

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

gateway is connected to various home Wi-Fi nodes and accesses to the Internet network. A smart home network system that remotely monitors and controls hand-held terminals is implemented [1].

2.1.2.Networking scheme. Each Wi-Fi module connects to the cloud server gateway through a router. The nodes are not affected and each connection remains independent.

2.1.3.Overall functional design. The smart home system designed in this paper includes intelligent lighting control system, intelligent temperature control system, etc[2].The overall functional design of the smart home is shown in Figure1.

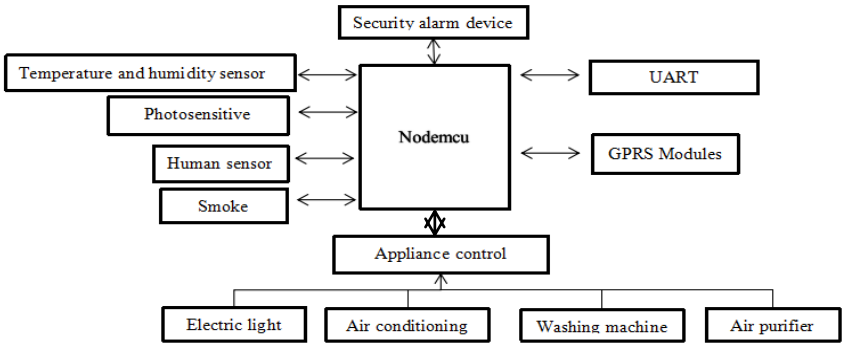


Figure 1. Overall function design of smart home

2.2. System main module hardware and software components

2.2.1.System main module hardware components.The main modules of the system consist of smoke sensors, human sensors, temperature and humidity sensors, lighting controllers, cloud server gateways, and GPRS modules [3].The specific role of each part is as follows:

- Smoke sensor: It is mainly installed in places such as home kitchens where main electricity is used. It is used to detect the presence of smoke in the surrounding environment. The fire is discovered in time and a signal is sent to the main control.
- Human Sensor: It carries out security auxiliary inspection and automatically controls the light.
- Temperature and humidity sensor: The temperature and humidity of the ambient air are detected every ten seconds, and the air temperature and humidity data are returned to the main control chip.
- Lighting controller: According to the main control chip to control the light brightness.
- Cloud Server Gateway: The gateway is used to implement the data processing aggregation function of each node. Users can access the cloud platform through the Internet to obtain various data and perform effective monitoring. The discovery data exception immediately informs the gsm module to alert the user.
- GPRS module: Through the Internet without Wi-Fi, the user is notified of the real-time situation by SMS and phone.

2.2.2. Main module software components. This design adopts a cloud server as the gateway of the smart home system. Because the node used in this design is a Wi-Fi connection directly to the Internet, no local other device is required to forward data across the network. Therefore, cloud servers can be directly used for data processing, storage, and control commands. Our operating system uses Win Server 2016. This system is used because it has the following advantages:

- Development, the system can install many development environments.
- Low cost, operators have lower cost control on cloud servers.

- Multiple users, with public network IP, can directly send and receive data from many nodes without forwarding.
- Safe and reliable, the cloud server has special maintenance and high security.

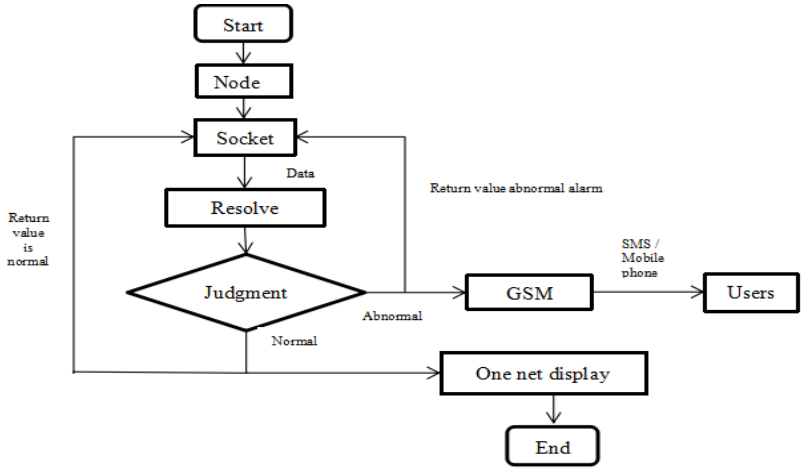


Figure 2. Server software design flow chart

The JDK is installed on the cloud server, and the temperature and humidity data of each Wi-Fi node is processed through a Java script, uploaded to the one net cloud platform, and displayed on the free cloud platform. Users can monitor data through the platform, and can also send commands to the cloud server to control the smart home system. When the data is abnormal, the program connects the GPRS module to enable the platform to have the ability to notify the user of text messages and telephones. The server software design flow is shown in Figure 2.

This design is based on Nodemcu's low-cost smart home node, which realizes the construction of a smart home system at a lower cost, completes the user's remote monitoring of home environment parameters, and controls some of the controlled home devices[4].By connecting home control equipment and cloud servers, the node realizes the exchange of information data between home equipment and nodes, nodes and owners, and finally realizes the regulation and remote real-time monitoring of indoor equipment.

3. System design and implementation

3.1. Overall system framework design

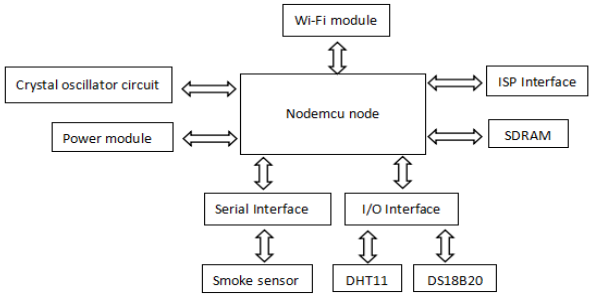


Figure 3. Smart home system structure

The Nodemcu-based low-cost smart home node design mainly uses the Nodemcu as the master. Figure 3 shows the composition of the associated hardware platform. Mainly by the MCU within the ESP8266 to control and process, which can be through the various communications port sensor data acquisition and processing, and data through the Wi-Fi module connected to the Internet to upload

data to the cloud server gateway. In this way, effective information transmission can be achieved and control information actions can be completed in real time. The node chip designed in this paper is ESP8266. The implementation of all functions is related to the master control. Instead of using other masters to control the ESP8266, it is directly controlled by the ESP8266. The biggest advantage of using this scheme is its low cost. The node design includes the ESP8266 and CP2102 USB-to-serial modules.

3.1.1.Functions of ESP8266 module. The work of ESP8266 needs 5V power supply and crystal oscillator circuit to ensure its normal operation, and it needs to design the ISP download port to facilitate the software debugging. In this design, the ESP8266 is programmed directly to control the peripheral modules. This not only improves resource utilization but also reduces node hardware costs.

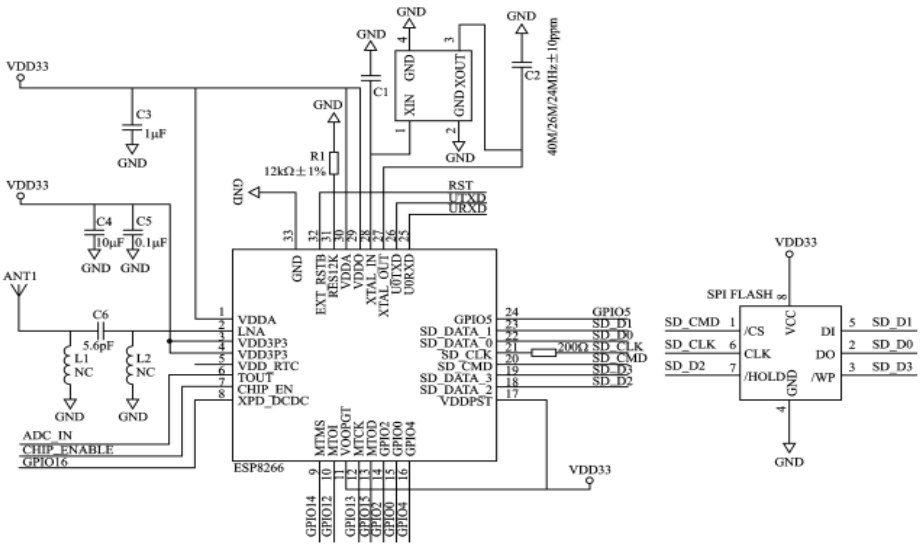


Figure 4. ESP8266 module circuit diagram

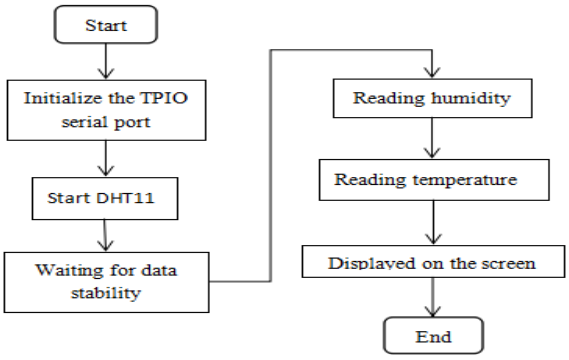


Figure 5. DHT11 data collection process

3.1.2.ESP8266 module circuit design. Because of the high degree of integration, the peripheral design of the ESP8266 module is very simple. In addition to the main chip, only one passive crystal, one SPI Flash, and several resistors, capacitors and inductors are required.ESP8266 module circuit diagram shown in Figure 4.

3.2. System software design and implementation

3.2.1.Smart home node software design. Based on the ESP8266 IoT, IoT SDK provides users with a simple, fast and efficient software platform for the development of IoT products. The SDK provides users with a set of function interfaces that can receive and send data. Users can only focus on the development of the upper application of the Internet of Things, without having to care about the structure of the underlying network. All network functions are implemented in the library, and being opaque to users is a major feature of the ESP8266.Users can do this in user_main.c. Void user_init(void) is the entry function of the upper program, which provides the user with an initialization interface. The user can increase the hardware initialization and other functions. The specific software implementation flow is shown in Figure 5.

3.2.2.Introduction to Nodemcu. Nodemcu is a programmable firmware based on the ESP8266 chip. It allows developers to implement programming logic through simple upper-level code to operate the underlying hardware to achieve software requirements. At the same time Nodemcu also provides a network API, through the event driver can greatly facilitate the developers to achieve their own design features. The Arduino IDE that this design uses carries on the development to the chip, the programming style is succinct, facilitates the developer to realize the rapid development.

3.2.3.Smart home background server development. This design adopts the currently popular cloud server as the smart home gateway server designed by the company. It aggregates all data, monitors each sensor data in real time and creates messages for users. The initialization of the GSM module registrants the network first, waits for the module to register successfully, connects to the background server port after successfully connecting to the Internet, and completes the three-way handshake in the TCP/IP protocol and then successfully connects to the cloud server. The master obtains data such as temperature, humidity, and smoke concentration every second. It averages the data and packages the data that needs to be sent to the server. The data format is shown in Table 1.

Table 1. Data frame format

Send direction	Baotou	ID	Command code	Data length	Data pack	Terminator
Server → Node	0x68	ADDR	0x10	0x01	0x10	0x16
Node → Server	0x69	ADDR	0x10	LEN	DATA	0x16

A communication module is started every ten seconds to upload data to the server. After the server receives a normal return value, the connection is closed and the module is set to sleep mode. Configure the operating mode so that the master chip also enters the standby state.Java as a background programming language, Server Socket class in Java has a variety of methods to provide calls to achieve socket connection. Create a Server Socket on a specific port. After creation, the server starts listening for requests on port 8899.The node connects to the server via TCP/IP, and the server creates a new socket to communicate with the node. After the server calls the accept() method, it enters a blocking state and waits for the connection of the node. When receiving a request from a node, the server creates a thread pool to allocate threads. Each thread creates a new socket to communicate with the node. After the communication is completed, the thread is closed. This allows the server to connect to multiple nodes[5]. After a socket connection is established, the receiving node sends data and analyzes the contents of the packet. Table 1shows the contents of the packet resolution. Each node is defined with a unique address. When the data check is normal, it will be sent to the one net cloud platform for data display. If the data is abnormal, the gateway will not only notify the cloud platform of the abnormality, thus notifying the user through the platform APP, and will also notify the GSM node to notify the user via SMS or by telephone.

4. System test results

An important part of this design is software testing, which is also the key to ensuring the normal operation of the program. The smart home access network is connected to Wi-Fi and IoT

connections[6]. Users can check the relevant test data in real time by logging in to the mobile phone or browser to monitor the relevant home appliances in the house.

4.1. System function test and analysis instructions

4.1.1.Hardware function test. According to the hardware requirements, design the hardware scheme and connect the circuit according to the hardware function interface. The appearance of the node is shown in Figure 6.Connect the node to the 5V circuit, test the voltage of the power supply port is normal, the brightness of the power indicator on the Nodemcu motherboard is normal, and the power supply of the DHT11 sensor module is normal. The chip can be found by the debugger, and both the chip and the peripherals can operate normally.

4.1.2. Software function test. First, the user logs in to the mobile phone client app. The smart home platform system builds it on the basis of the login command so that it can come to the login display interface. Based on the user name and password entered to obtain the system verification match, the corresponding security login can be completed. After entering the system, the smart home that has accessed the cloud server can be discovered. The connected smart home device can be operated in real time. The data observed by the user in real time through the mobile phone is shown in Figure 7.

4.2. System test results

According to the above related content, the relevant functions of this system are tested, and the test results are as follows:

- The GSM module networking is not affected by the communication distance, geographical location, etc., to meet the general family needs and other special needs;
- The main control chip has a rich interface and is compatible with common data acquisition modules. After the expansion of the connection module, the system can work normally.
- The system is running normally, and the temperature and humidity data can be uploaded to the server, and the monitoring data can be reflected in the monitoring interface in real time;
- There is no mutual interference between each module to maintain stable operation, which improves system stability.

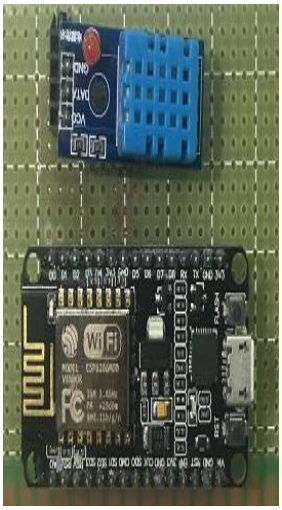


Figure 6. Node appearance



Figure 7. Users observe relevant data in real time through mobile phones

Acknowledgment
This work was supported by the National Natural Science Foundation of China under Grant No. 61473174 and Weihai Science and Technology Project (2018). Corresponding author: Liu hai.

Reference

[1] Meng Kun. Smart home system design based on MSP430 [D]. Master's degree thesis, Suzhou: Suzhou University, 2014.05, 9-12.
[1] Liu Yan, Zhao Xumin, Jiang Jiawei. Development Status and Development Trend of Smart Home[J]. Household Appliances, 2016.07, (07): 17-19.
[3]Liu Miao. Research and Implementation of Intelligent Home Control System Based on Internet of Things Technology [D]. Master thesis, Lanzhou: Lanzhou Jiaotong University, 2015.10, 18-25.
[4] Wei Qingmei. Design and implementation of smart home system based on wireless sensor network [D]. Master's thesis, Xi'an Xidian University, 2015.09, 5-6.
[5] Chen Guojun, Chen Lei, Chen Xixi. Java 2 Programming Design Foundation [M]. Beijing: Tsinghua University Press, 2006.01, 206-210.
[6] Xu Longjie, Ying Wenda. Analysis and Application Practice of Smart Home Application Development[J]. Telecommunication Science, 2015.11, 31(11): 116-122.