Advances in Computer, Signals and Systems (2023)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2023.070307
ISSN 2371-8838 Vol. 7 Num. 3

Embedded and Single-chip System Development Mode

Hongbing Meng, Jun Chen, Andi Yang, Qinghua Han

Tarim University, Alar, Xinjiang, 843300, China

Keywords: SCM; Embedded System; System Development

Abstract: Due to the rapid development and application of computer technology in many fields in China in recent years, embedded systems have also been widely used. Based on strong functionality and excellent applicability, embedded systems have good application value and prospects. Therefore, improving the embedded system development capability and product quality plays an important role in the development of related enterprises and technologies. Based on this, this paper discusses the development of embedded system from the aspects of development process, boot loading design, etc. on the basis of a brief description of related concepts, hoping to provide valuable reference for related work.

The embedded system is based on the computer technology such as single chip microcomputer. Through the selection of software and hardware, it can control the system performance, volume, energy consumption and other aspects. It has the characteristics of good performance, small size, and strong timeliness. Compared with the general computer system, it has good stability, high computing speed, shorter development cycle, high cost performance ratio, and easy to carry and use. Therefore, it is applied in many fields such as production and national defense; when developing embedded systems, we need to deal with power consumption, interference and other issues reasonably to ensure the long-term stable operation of the system.

1. Brief analysis of relevant concepts of embedded system of single chip microcomputer

The embedded system is a computer system based on the application center, which applies computer technology and related software and hardware to reasonably control the function, reliability and cost according to the design requirements. IEEE defines it as a control device that controls, monitors or assists the operation of related equipment and machines. It can be embedded into the application system based on the controller in the form of single-chip microcomputer, and finally form an information technology product. Embedded systems are widely used in military, aviation, medical, communication, electrical and other fields [1].

Embedded system generally consists of microprocessor, memory, graphic controller, operating system, application software, peripheral hardware and ports, and is usually embedded in main equipment for operation. Embedded systems have developed for a long time. With the innovation of various circuit technologies and computer technologies, various microprocessors and embedded systems have been widely used and developed. Microcontroller is the first microprocessor used in embedded systems. Today, it is still used in electronic equipment in various fields such as automobiles, household appliances, communications, etc., mainly for simple functions related to control equipment[2]. With the gradual improvement of the requirements of related equipment for the

complexity of the control system, the single-chip microcomputer has also developed from 8-bit, 16 bit to 32-bit, and the embedded system based on the single-chip microcomputer has also been diversified, which has a positive significance for shortening the development cycle of related products and improving the control ability. The characteristics of the single-chip microcomputer can integrate the complete computing function into a chip, and the use of it to build an embedded system can simplify the device circuit It is of positive significance to improve the reliability of the system. The real-time nature of embedded technology improves the efficiency of relevant control information processing and transmission, which is conducive to shaping a good product experience.

2. Embedded system development mode

2.1. Development process

The general process of developing embedded system based on SCM is as follows: ① Establish a suitable development environment, install and develop the operating system and cross compiler. When selecting the operating system, it should be selected according to the running environment and operating environment. You can choose to use the embedded Linux system as the embedded operating system. You can choose to install all or customize the installation according to the development needs, download the matching cross compiler through the network, or directly install the cross compiler recommended or provided by the operating system product. ② Configure the parameters of the serial port terminal tuning tool. Take the Mnicom software as an example. As a Linux system display and keyboard input tool, it is mainly used to debug the information output of the development board. You need to debug the baud rate, data bit, stop bit and other parameters, perform parity check, and set the flow control parameters of software and hardware. ③ Network configuration: Linux system network configuration is similar to the network terminal configuration under Windows. Generally, NFS network files are used for configuration, the firewall is closed, or the system network debugging environment is effectively simplified. ④ Establish the system boot loader. Due to the problems of code copyright and open source, the program loading needs to be carried out on the basis of understanding its detailed working principle, or operate accurately according to the flow in the manual[3].

2.2. Performance test

In the actual operation of the embedded system, there may be problems such as system jam and system crash, which will lead to some electronic devices cannot be used. This is because the embedded system needs to continuously carry and process a huge amount of information. If the processor performance is general or the hardware design is unreasonable, the system is vulnerable to damage. Therefore, it is necessary to conduct a comprehensive and effective performance test on the entire system before installing the embedded system on related equipment, so as to ensure that the embedded system can run smoothly and stably. The general process of performance test is as follows: 1) Technicians should ensure that the working environment of the embedded system has certain compatibility or matches the target application scenario. Generally speaking, except for the embedded system developed for special scenarios, in order to improve the efficiency of system development, it is necessary to maximize its compatibility, especially software functions, output modules, and communication modules. 2) On the basis of testing the overall load condition of the system, conduct real application testing according to the target application scenario, so as to deeply judge whether the overall performance of the system, software and hardware performance, and circuit design are scientific, reasonable, safe and reliable. 3) The programmer should simulate the operation of the

system after long-term use, and conduct performance tests at different time points. If the performance tests at different time periods are not significantly different, or within a reasonable range, the test passes. If the test fails, the corresponding components need to be optimized and adjusted, and can be put into production only after the test is completed.

3. Specific development and design of embedded system program of single-chip microcomputer

3.1. Guided loading design

The general workflow of the boot loader is: after the embedded microcontroller system receives the command, the main system will receive the command signal that allows the boot loader to start, and the microcontroller will also be configured to accept the status of the program code. Finally, the code signal will be downloaded from the designated host to the corresponding embedded system program, and the code data will eventually be transmitted to the memory of the microcontroller, And rearrange and program in the main memory system. After the boot loader is downloaded and installed successfully, the relevant programs and functions can be executed automatically.

(1) In terms of hardware, after the connection between the host computer and the communication peripheral hardware of the single-chip computer is successful and the detection is normal, the single-chip computer starts to boot the loading program. First, the pin is assigned. The digital cross switch can be set as the digital peripheral interface port pin of the single-chip computer. It can be arbitrarily combined on the basis of meeting the software pin allocation rules. In order to allocate efficiency, the same script is generally used to allocate the boot loading program. Then the boot loading is allowed. In order to ensure that the system can successfully complete the system programming operation under reset or other conditions, it is usually necessary to set a pin as an input signal to notify the system to download and allow the transmission of signals with the host system that is booted to download the program. Finally, the host will notify the microcontroller to start the loading process when the allowed loading signal is read.

(2) Software. When the system receives the command to allow loading, it should ensure that the system can tell the microcontroller to do a good job of receiving data. First, the system needs to set up communication peripherals to meet the communication needs of the boot loader; secondly, the microcontroller should have a memory to download and save the text of the boot loader. In order to establish an appropriate communication link, the automatic baud rate and other parameters should be used to detect and adjust the bit rate; both the host computer and the microcontroller device need to use the designed baud rate. After the device receives the signal of the received data, ensure that the host computer can receive the corresponding notification to quickly send out the program data and relevant additional information.

3.2. Low power consumption design

Embedded systems that use SCM as the main control program are generally used in many environments where power supply is difficult, such as related electrical equipment in the field and underground, which can only carry batteries and small solar panels to power the system. Therefore, it is necessary to consider the power consumption when developing embedded systems. For example, the power consumption of embedded systems developed with CMOS devices is lower than that of general systems. CMOS devices have the characteristics of digital integration. All power consumption of their circuits during operation only includes static power consumption and dynamic power consumption. Dynamic power consumption is comprehensively affected by the load, operating frequency, power supply voltage, output level and other parameters. When the voltage value is large, the circuit operation requires a large power supply value. Based on the close relationship between

voltage and logic level, when the logic level of the system output has a large swing. The voltage required for operation is close to the supply voltage.

(1) During the hardware design, the dynamic power consumption, power supply voltage, clock frequency and other parameters of the system shall be mainly controlled, so as to reasonably control the power consumption of the system. Generally speaking, the clock frequency shall be selected to a smaller value, and most components shall be in the state of static power consumption. The power supply voltage shall be lower than high when the system operation is guaranteed. In terms of software design, the interrupt mode can be used to replace the query mode. Both the query mode and the interrupt mode in the program can better deal with simple applications. However, the power consumption characteristics of the two are quite different. Using the former processor requires continuous access to registers, which will increase the operating power consumption. However, the processor in the interrupt mode does not need high-frequency access to registers. If necessary, it can completely switch to the wait or stop access mode. Therefore, the power consumption will be much less.

(2) The timer timing function is used to replace the delay subprogram. The latter uses the processor to continuously carry out empty operation instructions and minus one operation transfer instructions to achieve delay, waiting and other operations. The processor workload increases, the dynamic operation time of related components increases, and the power consumption is naturally high. However, using the timer to meet the system timing logic requirements can significantly reduce unnecessary dynamic power consumption.

(3) Set the I/O module to operate intermittently or not to operate to reduce power consumption. If a single I/O pin of a single chip is used to control communication, when the system has no communication, the I/O module will be set to the fully closed state.

3.3. Anti-interference design

In the industrial environment, the main reasons for the interference of embedded systems in the form of pulses are space interference, power supply interference and channel interference. If the microcontroller or embedded system is interfered, it is very easy to cause control failures, data errors and other problems. Therefore, when developing an embedded system, it needs to be optimized from both hardware and software to deal with the interference.

(1) In terms of hardware, anti-interference can be achieved by means of isolation, shielding, etc. The isolation technology refers to the use of optocoupler to cut off the connection between the circuit connected to the microcontroller and other components, or the use of isolation transformer to achieve electromagnetic isolation. The shielding technology refers to the use of a metal box to cover the microcontroller and embedded system into the box to isolate or weaken the electromagnetic interference of the entire system from external appliances. Decoupling technology refers to the installation of decoupling capacitors between all integrated circuits and power supplies of the system, so as to store the capacitance on related circuits and absorb the electrical energy released by the power instantaneous switch in the circuit. For power supply, the power transformer can be optimized based on double isolation technology, and it can be paralleled to the primary coil and secondary coil of the power supply by using varistors to suppress surge and spike power supply voltage. The low channel filter can also be used to filter out the high-order harmonics carried by electromagnetic interference.

(2) In terms of software, NOP instructions, instruction redundancy, software traps and other technologies can be used to reduce interference. NOP instructions add single byte NOP instruction text to double byte and triple byte instructions, so as to avoid the problem of instruction operation errors caused by program runaway after interference, and ensure that the program can still run on track after interference. Instruction redundancy affects the flow direction of the program by repeatedly

writing decisive instructions. When the program is runaway and affects the running area. The relevant instructions will be repeated to ensure that the instructions are correctly executed. The software trap technology is to set the interceptor outside the program running area, so as to lead the out of control program area to the corresponding processing and repair address for handling errors, and use long branch instructions to restore the program to normal operation.

4. Conclusion

To sum up, this paper introduces in detail the relevant issues in the development of the single-chip embedded system. In the actual development work, we should reasonably carry out the design of low power consumption, anti-interference and other aspects according to the target application scenario of the product, and strengthen the performance test, so as to better balance the processing speed, energy consumption and other performance parameters of the system, and make it more suitable for the specific working environment.

Acknowledgement

Fund project: First class curriculum reform project of Tarim University (22010030910); Key Research Project of Higher Education Teaching Reform of Tarim University (TDGJZD2101).

References

[1] Xu D. Analysis on the application of embedded system in the Internet of things [J]. Popular Standardization 2020; (24): 196-197.

[2] Qi L, Zhao L, Wang Z. Design of computer embedded operating system based on the Internet of things [J]. Electronic Design Engineering 2020; 28 (22): 34-37+42.

[3] Meng H. Research on the development and application of embedded systems [J]. Electronic Testing 2019; (13): 104-105+60.