Design and Implementation of Networked Control System Based on ARM and Zigbee

Xudong Yang. Jinsong Tan
Chongqing Vocational Institute of Safety & Technology. Chongqing Wanzhou,China. xudongyang8@163.com

Abstract. With the rapid development of computer network technology, Network Control System has gradually become a hot point in the control field. AT91RM9200 micro-processor, CC2530 RF chip, DC motor and optical encoders were used to build a miniature embedded networked control system. Then some embedded software developments and also control algorithm were designed to make sure that the system can control the motor through a wireless network. Finally, network transmission test and control experiment were conducted to observe the actual network traffic conditions. In the Motor non-load condition, the local control effect was compared to the networked control to get made a series of experimental data, and these data were analyzed and summarized.

1. Hardware Design
The Embedded Networked control system designed in this paper is to construct a closed-loop control structure to adjust the speed of the controlled object DC motor so as to achieve the desired speed. The control method is to change the duty cycle of square wave pulse supplied to the motor by means of pulse width modulation (PWM), and adjust the average voltage loaded on both ends of the motor, so as to achieve the purpose of adjusting the motor speed.

The hardware of the whole system consists of three parts: the local controller with AT91RM9200 as the core, the Zigbee wireless communication module with CC2530 as the core and the motor drive module with L298N as the core.

1.1. Controller circuit design
The core of the controller is AT91RM9200 chip based on ARM920T kernel produced by ATMEL Company. Its main work is to read the data from remote controller from Zigbee wireless network. After calculation, the data is converted into PWM duty cycle, so as to control the output PWM waveform with adjustable duty cycle, after being amplified by the motor drive module. It can be added to both ends of the motor to adjust the speed of the motor. The overall hardware structure of the controller part is shown in Fig. 1. In the figure, the power circuit is used to supply power to the controller circuit, providing 12V and 5V voltages. The external storage circuit is a 4MB NOR FLASH memory that is extended beyond the on-chip 16KB SRAM and 128KB ROM for storing boot code, Linux kernel, applications, etc., and maps to Chi Select 0 of AT91RM9200. In addition, 32MB SDRAM is also extended as the program's running space. It is composed of two 16-bit data-wide SDRAM memories connected in parallel, and is mapped to the Chip Select 1 of the AT91RM9200. In order to provide data storage for the system, a 64 MB NDFLASH memory is specially extended and mapped to Chip Select 3.
The overall hardware design of the controller is divided into a 6-layer core board and a 4-layer bottom board. The core board integrates some important units, including the AT91RM9200 microprocessor, NOR FLASH, SDRAM, NAND Flash, system clock and so on. The core board is connected to the bottom board by 2 * 100 double-row pins. The bottom board outputs 3.3V and 1.8V DC voltages after the 12V and 5V DC voltages are processed by a linear regulator, and provides 1.8V chip voltage and 3.3V IO working voltage for the core board through a double row of pins.

1.2. Wireless communication module circuit design

The core part of wireless communication module is an advanced RF chip produced by TI company [4]. In the design process of this system, its main function is to package and analyze the data transmitted in the network, and transmit and receive through the antenna. The power supply circuit supplies both the 3.3V and 1.8V voltages to the module. The JTAG interface is the interface for downloading and debugging user programs. UART is not only the channel of data transmission between wireless module and controller, but also provides a necessary way for users to observe data transmission. The balun circuit is a functional circuit designed to provide impedance matching for RF (radio frequency) signals to ensure proper signal transmission and reception. In addition, almost all the PIO ports available in the system are deliberately introduced for user expansion. The overall hardware structure of the Zigbee module is shown in Fig. 2.
1.3. Motor drive module circuit design
In this system, the timing counter TC of the controller module works in the waveform mode and outputs the PWM waveform through GPIO. This waveform is essentially a TTL logic level signal, and the operating voltage of the system controlled object DC motor is 12V, it is obviously impossible to directly supply the GPIO output signal to the motor. Therefore, a device that converts the TTL logic level into the required driving voltage of the motor is needed. The L298N can meet this requirement. L298N is a motor driver chip produced by SGS Company. It contains two H-bridge full-bridge drivers with high voltage and high current. It can accept standard TTL logic level signals and drive two-phase or four-phase motors [5]. The internal schematic diagram of L298N is shown in Fig. 3.

![Fig. 3 internal schematic diagram of L298N](image-url)

2. System software design

2.1. Overall design
The software part of the system adopts a modular design method, which is divided into several discrete parts to design, write and debug respectively according to the functional characteristics and the operating platform, and the system is coordinated after each part is tested separately, which ensures that errors can be quickly located in which parts, greatly facilitating later functional testing.

The design of the system software part is implemented in the following parts:

(1) Zigbee Application design, mainly to achieve networking - serial data reading - packaging and sending through the wireless network, while also to receive data from the wireless network - parsing and writing to the serial port to send to the controller.

(2) The driver of PWM device and motor speed measuring device based on Linux kernel, the PWM device driver is used to generate the PWM waveform for PIO of the controller and TC operation of the timer counter, and the motor speed measuring device is used to read the output pulse of the photoelectric speed measuring encoder, and then calculate and convert it into the actual speed data of the motor.

(3) Based on Linux, this application program mainly completes the operation of the PWM device, the motor speed measuring device and the serial port. The function of this application is: reading control instructions - modifying PWM duty cycle - outputting PWM waveform - reading photoelectric encoder output pulse - converting to motor speed - writing serial port.

(4) PC user interface and simple PID algorithm. It is used to read the motor speed data in real time and display the motor speed curve. The motor speed is adjusted according to the PID parameters set by the user.

2.2. Main program design of controller
The above driver only hides the hardware concepts and configuration details from the timer counter, general IO interface, advanced interrupt manager and so on, and abstracts the hardware in the form of file description. In order to realize the functions of measuring the speed of the controller, communicating with the wireless module, generating the PWM waveform, etc., the user process is also
written on the upper layer of the controller to access the device driver file. This application consists of three functional modules: waveform generation, pulse measurement, and serial communication. The program flow is shown in Fig. 4.

![controller program flow chart]

**Fig. 4** controller program flow chart

2.3. *Wireless communication program design*

The coordinator and router router are separated, and their applications are designed separately. The coordinator is the founder of a network, which has the highest privileges within the network. All routing nodes and terminal nodes must be licensed by the coordinator. Each Zigbee network can only have one coordinator. The router is regarded as a node that can transmit data directly and bidirectionally to the coordinator. In this paper, a coordinator and a simple network of routers are used to realize the communication between the local controller and the remote controller. The choice of coordinator and router is originally decided by jumper in TI's original system, but now in our own system, it depends on the value of the variable zgDeviceLogicalType. If it is ZG_DEVICETYPE_COORDINATOR, it means to start with the coordinator. If it is ZG_DEVICETYPE_ROUTER, it means to start with the router.

An important difference between the coordinator program and the router program lies in the part of the network formation. In fact, in the implementation process, the difference between the two is only one device identification difference, that is, the device's intranet short address, which is contained in the data structure, the short address in the network of the coordinator is 0x0000. In the same communication group, the short addresses of the remaining devices are allocated by the coordinator and are invisible to the user. In this paper, the Zigbee node program is modified based on the application of
the SampleApp light node which is circulated on the Internet, and several key functions in the program are introduced.

The SampleApp_SPI_Send() function, called during serial data processing, is used to transmit data from the serial buffer to the air through an antenna. The data structure closely related to this function is Sadr, which defines the mode of data transmission (unicast or broadcast), the device name of the target device, and the short address of the target device, and then call AF_Data_Request() function of the AF layer to implement data packaging and sending.

SampleApp_MSG_CB() function is used to receive wireless signals from the air, save data to the buffer, and then call the HAL (Hardware Abstract Layer) function HalUARTWrite() to write the received data to the serial port, thus completing the whole node "serial-wireless-serial port" transmission process.

3. Control algorithms and experimental results

3.1. Data Network Testing
Before starting the control experiment of the system, the wireless network was first tested to ensure that the data communication of the system can be carried out normally. In this paper, a basic wireless network is constructed at room temperature of 25 degrees Celsius and without obvious obstacles. The coordinator node is 15 meters away from the router node, and the coordinator node and the router node are connected to the serial port of two PCs respectively, so as to observe the data interaction. Coordinator node and router node send data in the sending area of the serial port to each other, where the communication mode used in the network is unicast, that is, coordinator and router Single-to-Single transmission.

The network protocol analysis software smart sniffer is used to capture a frame of data, which is sent from the router node to the coordinator. The short address in the segment of the router node is 0x0002, which is allocated by the coordinator, and the payload of the data frame is an array {01, 02, 03, 04, and 05}.

3.2. Simulation and experiment
The user interface is designed on the remote controller, i.e. PC, and the PID algorithm designed above is run. The user can adjust the PID parameters to achieve better control effect, and then adjust the speed of the motor through the network. The operation sequence of the system simulation experiment is as follows:

(1) Connect the local controller, motor drive module, motor and encoder, wireless module with wires and cables, the motor power cable is connected to OUT1 and OUT2 of the drive module, and the drive module input IN1 is connected to the controller's PA19, and the IN2 is connected to the COM of the regulated power supply. The encoder only uses phase A and connects the A to the PA21 of the controller.

(2) Power on the local controller, open the Hyper Terminal, cd to the /test/ directory, run the NetCon_AT91 program, that is, the local controller program, enter the PWM preset period at the prompt, the value is 1400.

(3) Power on the wireless module, since the node communication program has been programmed into the flash in advance, the node communication program automatically runs after power-on and opens the serial debugging assistant in the upper computer to observe the data transmission and reception.

(4) Power on the motor drive module, motor and encoder, the motor starts to rotate.

(5) Open the host computer client, set the speed, set the PID parameters, and obtain the speed curve. Two sets of test results are given respectively. The results give the rise curves of the networked control system when the set values are 1200n/min and 1500n/min, and the corresponding local control curves are given. The comparison of the group results can be seen: in local control, the motor will eventually stabilize at a given speed value after a few tens of milliseconds of adjustment process. Because the

5
motor and the speed encoder are not strictly coaxial and other unknown mechanical reasons, the speed curve is not very smooth, the rise curve has obvious overshoot. When joining the network, it can be seen that the adjustment process is prolonged, and the overshoot is significantly increased, and after the motor runs smoothly, the obtained curve burr is increased, so the addition of the network causes the control effect to decrease.

Assuming that the transmission time of one frame of data between the coordinator node and the router node is τ, there is a time delay τ between the motor speed information transmitted from the coordinator node through the network to the router node and the actual speed of the current motor. The control information sent by the controller after the operation is transmitted to the coordinator node via the wireless network through the router node, and then the delay τ is superimposed, so that there is a 2τ delay in the data transmission line which only exists in the wireless network part.

If the payload of the data packet is 2 bytes, the length of the data packet can be calculated as 23, and the maximum transmission rate of the Zigbee module is 250 kbps. The data transmission of the system is limited by the serial port baud rate, and the maximum baud rate of the serial port is 115200 bps. Therefore, the upper limit of the data transmission speed of the system is 115.2 kbps, which is 14.8 kByte/s. Dividing this value by 23, the effective transmission rate of the system control information is 0.6 kByte/s, so it can be seen that the value of τ is at least 1.6 ms. With the delay caused by the transmission of two RS-232 cables connected with the wireless module, the overall network delay of the system can reach nearly 10ms, and the DC motor used in this system has a maximum speed of 1800rpm/min. To control such a high-speed object, the delay of 10ms will inevitably reduce the control effect greatly, and the best baud rate of Zigbee serial port transmission is 38400 bps. In this case, in order to improve the transmission rate of the system, the baud rate is forced to be changed to 115200 in the underlying protocol stack file MT_UART.c, which may lead to packet loss in the process of network transmission and degrade the network performance. Delay and packet loss are inherent characteristics of Zigbee networks. To achieve better control results as much as possible, a series of work needs to be done in network optimization and control algorithm design.

4. Conclusion
The research of networked control system is a hot spot in the field of control in recent years. The purpose of this paper is to provide an experimental and simulation platform for networked control research. Designed for the hardware circuit of the whole system, and designed and developed the software under the embedded controller environment with AT91RM9200 as the core. Designed and programmed the wireless communication program, set up Zigbee network, design the remote controller program to realize the transparent transmission of data through serial port, and the system is adjusted to achieve remote speed regulation of DC motor.

Further work requires remote controller client program development, in the form of simulation software, to design common control and filtering modules for users, so that users can design control strategies according to their own needs and wishes. At the same time, add network nodes, increase system network complexity, and observe the communication status and control effects of the system.

References
[1] Wenzhong Wu, Wanlei Li. Smart home system based on ARM and Zigbee [J]. Computer Engineering and Design, 2011, 32(6): 1987-1990.
[2] Liubin Li, Guoyi Li, Peng Du. Design of embedded precise positioning system based on ARM7 and Zigbee [J]. Information Communication, 2012, 01: 82-84.
[3] Zuocheng Wang, Yulong Bai. Research and implementation of intelligent garage system based on Zigbee [J], Measurement & Control Technology, 2013(01): 69-71+75.
[4] Yuandong Liu, Lin Liu, Ruichi Zhang. DC motor PID control experimental system based on ARM9 [J] Industrial Instrument and Automation Device, 2011, (2):88-91.
[5] Zhenyu Cui, Rongliang Wang. Remote sensing control system based on Zigbee technology [J]. Computer Measurement and Control, 2010, 18(8): 1802-1804