The Design of the Intelligent Car Based on the Arduino UNO and LabVIEW

Lili Tang¹, Wei Huang² and Jie You¹

¹ Department of Computer and Information Engineering, Zhixing College of Hubei University, Wuhan, 430011, China.
Email: toney2001@126.com
² The Fourth Laboratory, System Design Institute of Hubei Aerospace Technology Academy, Wuhan, 430011, China.
Email: 34446549@qq.com

Abstract. The design of wireless remote control intelligent car uses PC-side LabVIEW host computer control software responsible for the interaction interface with the car, uses serial communication module APC220-43, one is connected to the PC to send commands from the host computer, the other is connected to the Arduino UNO to receive instructions from the host computer, and uses open source MCU controller Arduino UNO lower computer is responsible for receiving, parsing and executing instructions sent by LabVIEW. The upper computer and the lower computer carry out long-distance transmission of instructions through APC220-43 wireless serial communication module to realize the remote control of the car and the measurement of temperature, humidity and distance data. This design scheme provides certain reference value for the design fields of smart home, smart transportation, remote assistance, etc.

1. Introduction
The smart car is a small robot that can automatically drive by wheel movement. Due to its intelligent features, it can be applied to environments that are not suitable for human work, such as disaster relief, outdoor adventure, etc.

This smart car can be divided into three major components: command section, communication section, and execution section. The actuator plate is realized by a DC geared motor to control the direction and speed of the car running. The driving of DC motor generally has the following two ways: First, the driving module and the MCU development board are integrated on one circuit board, which can realize integrated control; Second, the car control core board and the driving circuit are separated from other circuits. In order to avoid the trouble in the debugging process, the second scheme was adopted.

2. Overall design
The upper computer software adopts the graphic block diagram program research platform introduced by National Instruments. The main functions include communication serial port selection, speed adjustment dial and trolley motion control keys. The communication part adopts APC220-43 serial communication module, which can send a large amount of data, and the user can accurately transmit the instruction of controlling the trolley without writing a program. The lower computer adopts Arduino UNO, which is simple and resource-rich. This design utilizes open source hardware typical representation of Arduino, graphical programming software LabVIEW and multi-channel embedded wireless communication module APC220-43 to realize innovative smart car design.
From the communication point of view, the communication part is realized by the APC220-43 wireless data transmission module. The module is cyclically interleaved and error-corrected, which makes the anti-interference ability strong and sensitive, and also provides multiple channel selection. Supports online modification of various parameters such as serial port rate, transmit power, and radio frequency rate. It can also transparently transport relatively large data, users do not need to design a software program, it also has a long transmission distance, and has a wide range of applications[1][2].

From the perspective of control, the upper computer is the control center of the trolley. The upper computer sends a command to the trolley through the “button”. After receiving the command, the trolley returns the relevant data to the upper computer. The upper computer receives the data returned by the trolley. Parse and display it.

2.1. Upper-computer Software Design

The front panel of LabVIEW mainly includes serial port selection, mode selection, temperature measurement, ranging, and humidity measurement. Temperature measurement, gear selection and direction selection. The car running state can be selected from the remote state and the autonomous state. In remote mode, it can respond to forward, backward, left, right, stop, and shift control and behavioral controls. In autonomous mode, it can drive autonomously and has obstacle avoidance function. Distance, temperature and humidity measurements can be made in both remote mode and autonomous mode[3][4].

2.2. Lower-computer Software Design

The PWM wave output from the Arduino UNO development board controls the voltage of the input motor to adjust the vehicle speed. The digital ports 5 and 6 of the Arduino UNO control board serve as the output ports of the PWM pulse wave, and the output signals serve as the enable control signals of the L298N driver chip to control the motor’s transfer. When the voltage is kept constant, the voltage will change with the change of the duty value. The average value of the output voltage is determined according to the duty ratio, so as to adjust the speed of the motor, that is, PWM speed regulation[5].

The digital ports 2, 3, 4 and 7 of the Arduino UNO development board serve as inputs to the L298N to control the steering of the motor. According to the L298N input logic, when ENA=0, the input signal cannot control the motor. When ENA=1, if the input signal is high or low, the motor state is forward or reverse; if both are low or high Usually, you can park quickly. The ENB function is the same as ENA. L298N can drive two DC motors, corresponding to the left and right wheels of the car, to realize the forward, reverse, left turn, right turn, brake and multi-speed control of the car. The Arduino UNO control panel reads distance, temperature and humidity measurements and obstacle avoidance by reading data from the ultrasonic module, DHT11 temperature and humidity sensor, and infrared obstacle avoidance module.

![Diagram](image-url)

**Figure 1.** Lower-computer software plan.
3. Design Scheme Features and Extension Instructions

3.1. Features Introduction
This design is based on the concept of combining computer software, serial communication and single-chip technology. From the selection of the MCU, the determination of the communication scheme, and the design of the upper-computer software, there are many differences with the traditional “smart car”. The features of this program are as follows:

The Arduino development board is selected as the core of the design to achieve the design requirements and achieve the desired set goals. The connection between the upper computer and the lower computer is well completed, and the Arduino complements LabVIEW, which makes the solution a lot more convenient.

The communication method adopts serial communication, and uses the serial port digital transmission module to complete the wireless connection between Arduino and LabVIEW, which expands the application range of Arduino and LabVIEW, and can select different types and different power wireless transmission modules to achieve different distances. Wireless data transmission.

Using the computer-side LabVIEW host computer to realize the wireless control of the car, realize real-time remote monitoring of data, and effectively combine serial communication and computer software. The use of upper-computer software to manage the car is in line with the design concept of simplifying and integrating complex control. Give the complex control and maintenance to the computer, and the user only needs to pay attention to the data that is useful to them, and the useful data is expressed in an intuitive form such as a chart, which is simple and clear.

3.2. Extension Description
Arduino's powerful and rich open source platform has laid a strong foundation for the expansion of this design. In addition to the functions that have been implemented, the car can also expand the following functions:

- infrared tracking function.
- using the wifi module to implement the Internet of Things function.
- using the ZigBee module to achieve short-range, low-power wireless data transmission.

4. Program Difficulties and Key Technologies
The basic functions of this design have been realized. The main difficulties of the scheme are:

- Arduino UNO controller analyzes serial communication data.
- When the distance between the upper computer and real-time display is used, temperature compensation is used to reduce the error of distance measurement.
- There is a delay after the upper computer sends the command, and a streamlined communication protocol needs to be established to complete the corresponding action.

The key technologies are summarized as follows:

- the designation of the communication protocol.
- the method of collecting and processing information by the lower computer.
- the graphical programming of the upper computer.
- the dynamic display of the distance, temperature and humidity of the upper computer.

5. System Simulation and Results Analysis

5.1. Overall System of the Car
The system of the car is mainly composed of the upper computer and the small car, and the trolley control circuit is placed on the trolley. The trolley controller mainly includes motor control, ranging, temperature measurement, humidity measurement, and obstacle avoidance.
5.2. Overall System Debugging

The upper computer interface of the car control system mainly consists of serial port selection, mode selection, temperature measurement, ranging, and humidity measurement. It consists of buttons for temperature measurement, gear selection and direction selection. The main debugging contents are as follows:

**Table 1. LabVIEW Upper-Computer Instructions.**

| Button name          | Command | Function description                  |
|----------------------|---------|---------------------------------------|
| Remote mode          | 0xAA    | Remote command                        |
| Autonomous mode      | 0xFF    | Autonomous command                    |
| go ahead             | 0x00    | Control the car forward movement      |
| Back                 | 0x10    | Control the car to move backwards     |
| Turn left            | 0x20    | Control the car to turn left          |
| Turn right           | 0x30    | Control the car turn right            |
| Stop                 | 0x40    | Control the car stop                  |
| Gear selection       | 0x50, 0x60, 0x70, 0x80, 0x90 | Control car speed (a total of 5 gears) |
| Measuring distance   | 0x11    | Measuring distance                    |
| Measuring humidity   | 0x12    | Measuring humidity                    |
| Measuring temperature| 0x13    | Measuring temperature                 |
| Automatic temperature| 0x13    | Automatic temperature measurement     |

5.2.1. LabVIEW Upper-Computer Debugging. Open the virtual serial port application and add ports COM1 and COM2, as shown in Figure 4 below. Open the VI program, select the virtual serial port COM1 just generated, and run the program. Open the serial port debugging assistant, select the serial port number COM2, open the serial port, and accept the data, “automatic line feed display” and “hexadecimal display”, and receive the data from the host computer and receive the data.

5.2.2. Lower-Computer Debugging. After assembling the trolley system according to the design and simulation circuit diagram, check the connection between the circuit VCC and GND, mainly check whether the power supply of each chip is normal or not; after power supply, check the voltage of the power supply connection of each module with a multimeter, and power off after all normal. Then, according to the circuit diagram, each module is inserted in turn, and the power supply part, Arduino UNO controller, APC220-43 board, L298N driving part, DHT11, ultrasonic ranging module HSR04 and infrared obstacle avoidance module are sequentially checked, and each part of the power supply display system is detected step by step. Normal; after the connection is completed, mark the connection of each module to avoid the wrong connection of the line; bundle the connecting wires to avoid wire entanglement when the car moves.

5.2.3. System Debugging and Troubleshooting. After the host computer sends the command, if the car does not operate normally, system debugging and troubleshooting are required.

Check the power supply. Check whether the working indicator of the Arduino UNO control board is normal. If it is on, it works normally. If it is not bright, use a multimeter to check whether the power supply of the Arduino UNO control board is normal. Since the Arduino UNO control board is powered by the L298N motor drive module, it is necessary to detect the voltage from the L298N when detecting the power supply, and check whether the voltage is normal.
Check the communication. Connect the Arduino UNO development board directly to the PC with a USB cable. After the VI program outputs the command, if the car is in normal operation, check whether the APC220-43 and Arduino UNO digital pins 0 and 1 are cross-connected, TXD is connected to 0 pin, RXD is connected to 1 foot.

6. Conclusion
Through repeated trial and error and unremitting efforts, all the functions of the smart car preset can be successfully realized. The significance of the scheme is that the upper computer and the lower computer transmit long-distance transmission commands through the APC220-43 wireless serial communication module, realizing remote control of the trolley and measuring data such as temperature, humidity and distance. This design can be applied to smart home, intelligent transportation, remote assistance and other fields.

7. Acknowledgments
In this paper, the research was sponsored by the Science and Technology Research Program of 2018Hubei Provincial Education Department (Project No. B2018405).

8. References
[1] Chen Lvzhou 2014 Arduino Programming Foundation (The 2nd Edition) (Beijing: BEIHANG UNIVERSITY PRESS)
[2] Cheng Chen 2012 Arduino Development Guide (AVR) (Beijing: CHINA MACHINE PRESS)
[3] Peng Yong 2011 LabVIEW virtual instrument design and analysis (Beijing: TSINGHUA UNIVERSITY PRESS)
[4] Zhang Jiarong 2013 Proficient in LabVIEW virtual instrument programming and case realization (Beijing: POSTS & TELECOM PRESS)
[5] Shen Jinxin 2014 Arduino and LabVIEW development combat (Beijing:CHINA MACHINE PRESS)