Design and Implementation of Remote Control System for Mobile Platform Based on Interoperation of Android and Arduino

Xu Ri[1], Liang Jing-Chen[2], Yang Hao[3] and Teng Jing-Wen[4]

[1,2,3,4]*Beijing Electronic Science and Technology Institute*
[1]*4978368@qq.com,* [2]*447328916@qq.com,* [3]*604291518@qq.com,*
[4]*183124370@qq.com*

Abstract

The project designed and implemented an interoperability mobile platform with a system based on Android open source software platform and Arduino open source hardware platform , an Android program on the host terminal receiving environment data from the mobile platform and sending message through 3G channel using XMPP protocol, and an Android program on the remote control terminal which could process the environment data sent from the mobile platform according to the algorithm and send back control commands. With this system, the mobile intelligent terminal can collect environment data of moving objects in any area through the 3G signal and control the behavior of distant moving objects to avoid collision. Meanwhile, the system which has broad application prospect is prototype of self-driving car, robot designed for dangerous mission and Mars exploration rover.

Keywords: Arduino, Android, XMPP Protocol, Automatic Control

1. Introduction

Android is a Linux-based open source operation system released by Google on November 5, 2007, which allows developers to get and modify the source code freely. Android system is claimed to be the first open-source and completeѕ mobile software designed for mobile terminals [1]. Android has got a rapid development currently. As of November 2013, Android has occupied 76% of the global smartphone operating system market, 90% of the Chinese market.

Arduino is an open source hardware platform which developed by teachers in a high-tech design school in Ivrea Italy. The development environment of Arduino is Processing/Wiring where Java-like programming language could be used [2]. It can be used to develop interactive products and read a lot of signals from both switchings and sensors. It can also control lights, motors and other forms of physical devices. Besides, It not only can be developed as hardware devices to meet the functional requirements of the users, but also communicate with PC software and mobile intelligent software [3, 4].

Binding properties of these two systems, Arduino system can be mounted on a mobile platform. This project could implement the interoperation of Android and Arduino by using specific sensors to collect environment data. With Android system, the remote control terminal and the host terminal could swap data with each other. Research on combination of these two systems has rarely been touched at both home and abroad.

2. General Platform Architecture

The project aims to implement an interoperability mobile platform by using Android, Arduino and 3G network technology. Android system has good interactive interface while Arduino system is able to control a large number of electronic components. Android and Arduino can be integrated by using standard of Arduino-based Android Open Accessory and ADK tools released by Google.

General architecture of the platform is shown in Figure 1. The part of remote control terminal can achieve automatic control of security and command control functions. And Android control module is mainly used to implement this. The host terminal part has two modules: the environment data collecting module and the motion control module. The host terminal is a mobile platform used for receiving information from remote control terminal. The environment data collecting module whose core technology is sensor technology is to monitor the environment data. The motion control module whose core technology is DC motor drive technology controls the actions of the mobile platform.

As the bridge of information exchanging, the communication module is set on both remote control terminal and host terminal. It aims to exchange the remote data through 3G channel and XMPP protocol. The architecture of the module is shown in Figure 2.
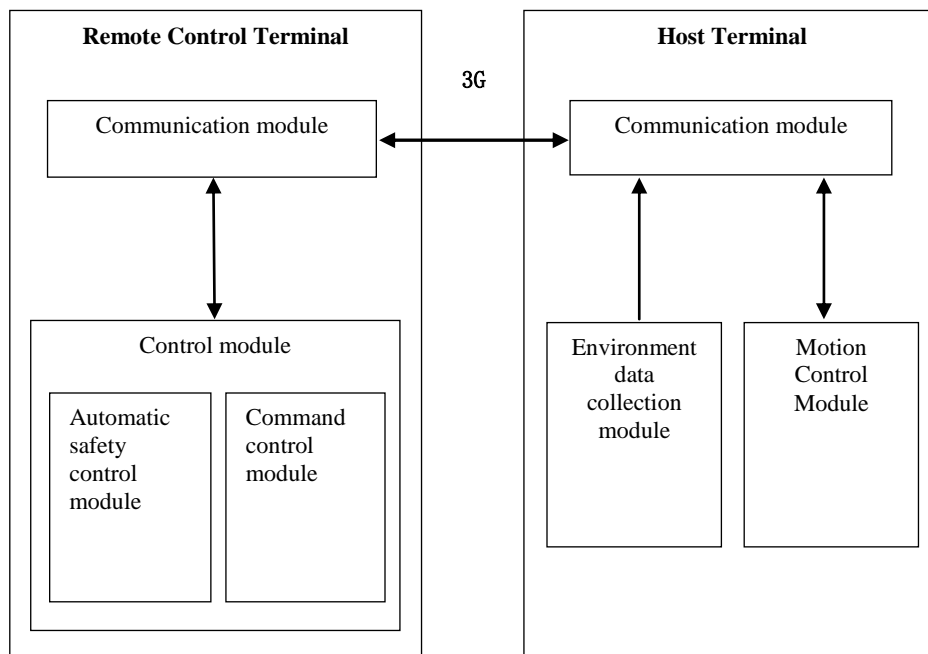

Figure 1. General Architecture of the Platform

XMPP (The Extensible Messaging and Presence Protocol) is a protocol based on Extensible Markup Language (XML). It is used for instant messaging (IM) and real-time online information detection [5]. It can promote impending instantaneous operations of clients between servers. XMPP defines three roles, client, server, and gateway. Bidirectional communication can occur between any two of the three [6]. Domestic 3G network access point only can get an intranet IP provided by the service provider. Therefore, the two devices cannot achieve point to point communication within the 3G network. For

these reasons, we select XMPP protocol to achieve communication. XMPP protocol uses C / S architecture- in this project, the client is developed on the Android platform, while the server uses Google's Gtalk server.
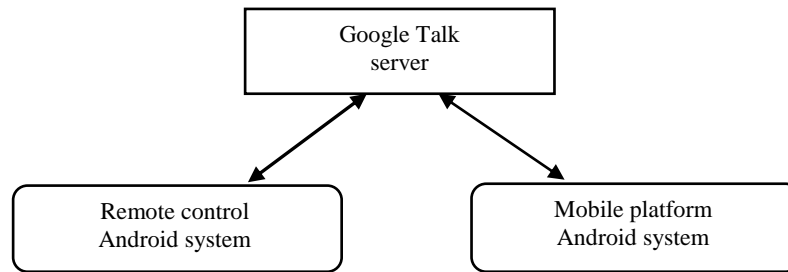
```
            ┌─────────────────┐
            │  Google Talk    │
            │    server       │
            └─────────────────┘
             ↗               ↘
┌─────────────────┐     ┌─────────────────┐
│ Remote control  │     │ Mobile platform │
│ Android system  │     │ Android system  │
└─────────────────┘     └─────────────────┘
```

Figure 2. 3G Communication Mechanism to Access Applications

3. Design and Implementation of the System

After collecting the analog data of the distance between the car and the obstacles which is detected by a sensor mounted on the mobile platform, the environment data collection module implements analog-to-digital conversion. Then it sends the digital data to the host Android communication module. The host Android communication module delivers it to communication module in remote control terminal through 3G channel. The remote control communication module delivers the date of environment data to the safety control module. Safety control module makes corresponding judgments based on the environment data and the security algorithms. According to the result or the instructions of the control module, remote control terminal sends motion control instructions to the host communication module through the communication module in the remote control terminal. Meanwhile, the instructions are delivered to the motion control module by the communication module in the host terminal. The motion module will make mobile platform to make corresponding actions according to instructions.

3.1. Design and Implementation of the Communication Module

In this project, the core of the design and implementation of communication module is XMPP protocol. XMPP protocol is implemented by Android-based XMPP (The Extensible Messaging and Presence Protocol) Service. The XMPP protocol's main function is initiating the request to create a connection to the server and ensuring the instant communication unblocked. The communication module will establish a connection with Internet network through the 3G network. After the connection is done, the communication modules in control terminal and controlled terminal can exchange data with each other through the Internet server. Firstly, the host communication module initializes the connection by acquisition system permissions in DEBUG mode. Secondly, the host communication module connects the server using the method new XMPP Connection. There're two states in the communication. One is to monitor real-time information, when there is a message being sent to the control terminal, the module read it, the whole process is implemented by using the method addMessageListener. The other is to establish communication with the account of remote control terminal when it is needed. After finding and clarifying the account of the remote Android terminal, it can create a real-time conversation connection named newChat using the method creatChat() and send message using the method sendMessage.

Due to the characteristics of the Android system, there only can be one Activity active in the forefront at one time. However, there are two real-time active activities required in this program. To solve the conflict, we change the XMPP protocol into a service running in background and the protocol which accepts signals from Arduino into an Activity. These two communicate with each other through Intent and Message. DroidGeigerActivity in the forefront is always at the most active state. When DroidGeigerActivity receives the distance signal from Arduino through broadcast reception mechanism, then it will start XMPP service by Intent and deliver the distance information to the remote control terminal using XMPP protocol [7]. The implementation and process in the remote control terminal is basically the same as the host Android terminal in terms of the implementation of XMPP protocol.

3.2. Implementation of the Automatic Safety Control and Control Module

The main function of automatic safety control module in remote control terminal is to accept the distance data from the host on-board Android terminal on mobile platform. It will determine the possibility of collision in the mobile platform according to the velocity and distance data. According to the above result, a control signal will be generated to change the velocity in order to avoid collision. Command control module is to send data such as advance and stop to the host platform through the human-computer interaction of remote control terminal. The control commands in this project are discrete digital data. While the behaviors and the speeds of the mobile platform are continuous analog data in actual movement [8]. Considering such features and the actual needs of the project, we designed the automaton model using the formal system based on hybrid interface automaton shown in Figure 3, which could always prevent the mobile platform from colliding while moving. Also the feasibility was verified by associated test algorithm of hybrid automaton.
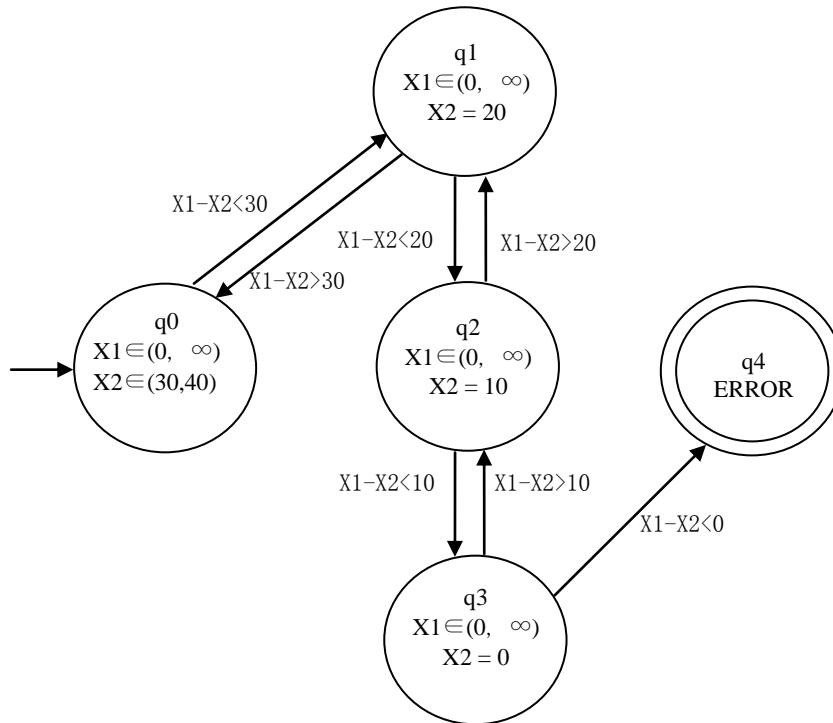

Figure 3. Safety Automation Model

In the automata model of this project, the initial velocity of mobile platform is 30 to 40 mm/s. The automat will turn into different states according to the distance between obstacle and the platform. Eventually, the velocity of the mobile platform will be altered. As the results of the validation algorithm described previously shows, as well as actual operating results, the state q4 has never been reached. That is to say the collision situation will never happen.

3.3. Design of the Environment data Collection Module

The core of the environment data collection module is ultrasonic sensors. The ultrasonic sensors are used to measure the distance data. The principle is to measure the time interval from the acoustic waves being launched to them being back during which they are reflected from an obstacle. As the velocity of ultrasonic in air is known to all, we can multiply the time interval and the velocity to get an actual distance we want.

The distance measurement formula is: $L = C \times T$; where L is the measured length; C is the velocity of ultrasonic in air; T is the time difference to measure the distance(T is half of the time between the transmission and reception).

The port of Arduino I / O will provide a pulse trigger signal more than 10uS to the Trig port (control port) of the ultrasonic sensor. The module will send eight 40KHz cycle level and detects the echo. It will output a response signal [9] once an echo signal is detected. The pulse width of the response signal is proportional to the measured distance. Thus the distance can be calculated using the time from transmitting to receiving the response signal, according to the formula $L = C \times T$ mentioned above.

3.4. Motion Control Module

In the Arduino, the duty ratio of the PWM can be adjusted to output different voltage values at the pin. PWM (pulse width modulation) is a method of digitally encoding for analog level. Modulating duty ratio of the square wave is used to encode a particular level of the analog signal by using the high-resolution calculator. A 130 DC motor is used in the project, it speeds up to 10,000 models rev / min. It's voltage is 4.5-6V. Motor gearbox reduction ratio is 1:120, it's maximum speed is up to 68cm / s. The motor is connected to the analog port on Arduino platform. Arduino is equipped with a motor drive expansion board so it can control output voltage value of the analog port, thus it can control the speed of the motor which decides the speed of the mobile platform.

4. System Test

Testing whether mobile platform could make predefined actions and measure the distance from the obstacles in front is the primary task in the testing phase. Re-burn the code into the Arduino board when testing and we can make the mobile platform to go forward, go backward, turn left, turn right, turn left front, right front turn, stop etc. All the above actions are based on the functions in library running on the platform coded before. While doing these actions, the mobile platform will stop when the distance from the obstacles is less than 40CM which is measured by the ultrasonic sensor.

Testing phase for the Android host terminal is mainly about whether the distance information of mobile platform can be received properly, whether it can communicate with remote Android through XMPP protocol, whether it can send commands to the mobile

platform. In the test, firstly, we connect the host Android to the platform. Secondly, the mobile platform will start APP automatically. Thirdly, distance information of mobile platform is displayed in the interface of program on host terminal after starting up, meanwhile it's sent to the remote control terminal.Then the remote control terminal will give the orders to the host terminal. Fourthly, the host terminal accepts the commands sent from the remote control terminal and transmit them to Arduino system. Lastly, the mobile platform makes corresponding actions based on the commands received from the host terminal.

Testing purpose of Android on the remote control terminal is mainly to test whether the system is able to receive distance information from host Android system and whether it can produce normal control commands and send them to the host Android system.

The main purpose of System Test is to test whether the system can run normally as expected, whether it can stop in front of an obstacle to avoid a collision.

The test results are shown in Table 1.

Table 1. Test Result

Test module	Test result	Whether achieve expected result
Mobile platform	make corresponding actions as expected, measure distance	Yes
Host terminal	Receive and display distance information, communicate with remote control terminal, send commands to mobile platform	Yes
Remote control terminal	Communicate with host terminal, produce commands	Yes
General system test	Avoid collision between mobile platform and obstacle	Yes

5. Conclusion

Remote control system for mobile platform based on Android and Arduino implemented the design of a mobile platform which could move continuously and measure the distance from the obstacles in front while moving. The system also includes two Android programs, one can receive environment data and send it through 3G channel on the host terminal, the other can receive and process 3G information and send command on the remote control terminal. With this system prototype, the mobile intelligent terminal can collect environment data of moving objects in any area where the 3G signal covers and control the behavior of distant moving objects.

Currently, the software developers in China know little about hardware, as the hardware developers know little about software. Our project combines the Arduino and Android platform together perfectly taking the advantages of the both, which also provides a very beneficial experience for the future design and development of interaction between software and hardware.

The project contains both software and physical entities. The physical entities interact via a wired or wireless network. The system can be widely used in safety-critical areas, such as Distributed robotics, telemedicine equipment and systems, modern grid system, modern traffic control systems, networked autonomous vehicle systems and *etc*.

References

[1]   Y F Wu, B M Du, Y G Su. Typical examples of android programming and project development, Publishing House of Electronics Industry. Beijing **(2011)**.

[2]   Bender P，Kussmann K. Arduino based projects in the computer science capstone course. Journal of Computing Sciences in Colleges. 5, 27 **(2012)**:152-157.

[3]   C Cheng, Arduino practical guide. China Machine Press. Beijing **(2012)**.

[4]   LORSCHEITER ，   Aguirre T ，   PAIM. Using Ms-Visualstudio and Arduino to do temperature measurements. Periódico Tchê Química. 8, 16 **(2011)**:60-66.

[5]   RFC6121， Extensible messaging and presence protocol (XMPP): instant messaging and presence. P. Saint-Andre, Cisco **(2011)**.

[6]   C G He, H G Li. Introduction to Programming for Android 4. Tsinghua University Press, Beijing **(2012)**.

[7]   ZHENG Meng. Detailed Java programming under Android system. Publishing House of Electronics Industry, Beijing **(2012)**.

[8]   Y.Zhang, T.Zhang. Hybrid interface automata. Proceedings of the 19th Asia-Pacific Software Engineering Conference (APSEC 2012), **(2012)** December 4–7; Hong Kong.

[9]   Massimo Banzi. Getting Started with Arduino. O'Reilly Media Inc, USA, California **(2011)**.

About the works

The works this paper based on, won first prize of the 2013 North five provinces (municipalities and autonomous regions), and Hong Kong, Macao and Taiwan college undergraduate Computer Application Contest.

Authors

XU Ri, (1978－), male, born in Jilin, teacher in Department of Computer Science and Technology, Beijing Electronic Science and Technology Institute, lecturer, vice director of Fundamentals and Applications laboratory.

LIANG Jing-Chen, (1991－), male, born in Hebei, student in Department of Computer Science and Technology, Beijing Electronic Science and Technology Institute.

YANG Hao, (1992－), male, born in Guangdong, student in Department of Computer Science and Technology, Beijing Electronic Science and Technology Institute.

TENG Jing-Wen, (1991－), female, born in Shandong, student in Department of Computer Science and Technology, Beijing Electronic Science and Technology Institute.