Wireless control of bubble display with Android terminal

Shogo Maeda*, Kouhei Hada*, Masahumi Hayashibara*, Eikou Gonda*, Hitoshi Miyata*

*Department of Electrical and Computer Engineering
National Institute of Technology, Yonago College
4448, Hikona-cho, Yonago-shi, Tottori-ken, 683-8502 Japan

gonda@yonago-k.ac.jp

Abstract

The bubble display controls the solenoid valve by PLC, generates the bubble in water, and shows the character and the figure with it. It is chiefly used in the several events. A present problem is a software problem and the operation when the display is changed because of its cable communication. The fact is that the displayed things are limited. Moreover, there is a fault that PLC is expensive. At this present research, both the application to communicate the data of the character and the figure drawn with the touch panel of the Android terminal with the control part on the bubble display by using Bluetooth and the control circuit that controlled the bubble display, were developed from the information. It became possible to output the pattern to the bubble display by drawing the pattern on the application of the Android terminal at present, and sending this to the control circuit.

Keywords: bubble display, wireless control, Arduino, solenoid valve

1. Introduction

A display through water is noticed as new expression method in recent years. This is called a water display, and there exist the various forms specialized in the respective contents. For example, there is a fog screen by which a picture is projected onto the fog which made spout and produce (1). This is the water display which enabled to correspond to a change in the viewpoint location of the observer by projecting a picture different from fog from manifoldness. The water display proposed by Eitoku and others makes light scattered on the drop of water which output from a valve, and it enables to describe a picture (2). Besides these, there is a bubble display which shows a character and a figure after generating bubbles in the water. The bubble display outputs the bubble pattern in an water tank by controlling a solenoid valve using the PLC connected to a PC. A survey of a bubble display is indicated on figure 1.1. Matsumura's and others "Water Canvas" is to assign character input by a keyboard and to sway a device with a speaker held in his hand by the user in front of the display, and to command to open a valve of source falling plumb down from a computer leads to expressing a movement of a hand as a track of a bubble which possesses interactivity (3). But in this system, since the described bubble patterns are limited to a track of input from a keyboard and from a bubble, it has low degree of freedom and is limited in information. Following 3 points are noted and have been developed about a bubble display in this research. First, for output with the various bubble patterns, improvement of the degree of freedom was aimed by drawing and a two-

Fig 1.1 An Overview of the Bubble Display

(a) Solenoid Valve  (b) Bubble Display
Fig 1.2 Bubble Display Used in This Experiment
dimensional picture using a touch panel. We worked with Logicom Corporation Ltd. That produces bubble display and other companies in the same trade of an enterprise in a joint research destination indicating a picture besides the ASCII code. But this can't change the indication freely. Secondly, the control by the cable, to which a migration sphere is limited, isn't desirable because the operation from the remote is needed for a bubble display of this research which assumed its use at a large place. Also, a communication wire has to manage itself. Therefore, between the Android terminal and the bubble display control circuit, Bluetooth communication is made and wirelessness of a control system is planned. Thirdly, in the previous study jointly made with Logicom, PLC control was performed, but by changing this to PIC, cost cutting of control circuit and its miniaturization are being planned. The bubble display and the solenoid valve used in this research are indicated on the figure 1.2.

2. Flow of research

The valve control circuit that controlled the solenoid valve and the Android application that sent to it the control signal were developed at this present research.

First of all, the development of the solenoid valve control circuit is described. The prototype was made by Arduino before the control circuit was constructed with PIC, though the valve control circuit is composed by PIC. Here, there are two reasons to make the prototype. Firstly, the purpose is to select the hardware newly mounted on the control circuit. The second reason is that the construction of a system was assumed to be a priority purpose. At this present research, it takes much time to take the correspondence of the communication with the Android terminal, if it takes time to produce the control circuit to operate the control circuit with the Android terminal. Therefore, to take the correspondence with the Android application, making the prototype was decided. Next, the development of the Android application is described. First of all, to take the correspondence with the prototype previously described, the valve control test application program was made that had both the telecommunication facility and the valve unit control function. The paint function was added to this, after the communication and the valve control by the test application program became possible. Then the Android application was developed.

3. Prototype Using Arduino

〈3·1〉 Arduino

Arduino Uno Rev3 (Figure 3.1(a)) adopted here is a microcomputer board that operates in a board equipped with the AVR microcomputer and peripherals and a special development setting. As its character, 20digital I/O pins exist, and by connecting the shield, it can connect hardware easily. As for software, the library exists a lot and enhancing hardware is easy. 〈3·2〉 Bluetooth communications by Arduino

The communication between Arduino-Android terminals was done by Bluetooth. As for Bluetooth and Wi-Fi, it is installed without fail, and it is inevitable in the Android terminal to select these today if it communicates with the Android terminal. The control from the distance through the network is not so necessary for the bubble display from the usage pattern. Therefore, Bluetooth that direct communication is personally easy in was adopted.

Wireless Proto Shield (Figure 3.1(b)) was connected with Arduino, and RN42XVP-I/RM (Figure 3.2) that was the Bluetooth communication module was installed in the shield. Wireless Proto shield was originally a shield to install Xbee that was the wireless telecommunications module, and the Bluetooth communication module that was used this time was the same as Xbee because of the same pin layout and the same pitch distance of RN42XVP-I/RM.

First of all, a connected test was done between Bluetooth USB dongles connected to Arduino and PC. However, inputs and outputs are actually done by serial communications, though it is a Bluetooth communication.
Therefore, the Bluetooth USB dongle and serial communications were done by using Tera Term that was the Windows terminal emulator software that the COM port and serial communications are possible for PC. Moreover, RN42XVP-I/RM and serial communications were done by using the library of special serial communications on the Arduino side. Next, the content of connected test is described. First of all, the pairing of the Bluetooth dongle and RN42XVP-I/RM is done according to the application only for Bluetooth. Next, the character is transmitted from Tera Term on PC. And data to communicate the cereal with RN42XVP-I/RM by using the cereal monitor that connects Arduino with PC are displayed, built into the development setting of Arduino. It was possible to communicate by setting the transmission rate of serial communications as a result of the experiment by 11520bps.

(3.3) Selection of Enhancing Hardware Using Arduino

Hardware besides the Bluetooth communication module was actually connected to Arduino, and the selection and the operation test of suitable hardware for the present research were done. It is microSD, Daughting connection transistor array TD62084APG, a liquid crystal display, and a controller that are actually built in. As a result, because the liquid crystal display and the controller are combined by all means with the control circuit, it contradicted the wireless, one of the purposes of the present research, and the adoption was put off. It was two (microSD and the transistor array) that were actually adopted. As for microSD, the received pattern data are usefully preserved, and the pins are few by connecting SPI ends. Next as for transistor array, because

---

Fig 3.3  Control system that was produced

Fig 3.4  Output pattern by Arduino

---

Fig 4.1  Circuit Diagram of a Control circuit using PIC
this was originally necessary for PLC to control the taking the place of solenoid valve, the main was its operation test. Because the rate of the solenoid valve were 24V, and 0.1A, the transistor array by eight circuits was used to attempt the miniaturization of the use of the transistor and the circuit of the Daulington connection.

(3・4) Bubble Display Control by Arduino

Bluetooth communication module, a transistor array, and an actual bubble display were connected to Arduino, and the bubble display was controlled. Here, it communicates the control circuit with the Android terminal equipped with the valve control test application program, not by PC. The detail is described later. Single-engined was able to be controlled by this experiment. However, if this is not continuously displayed, it is not actually significant. Then, the pattern was written, and was output to Arduino. At this time, the appearance of the circuit made here is Figure 3.3, and the appearance output to the bubble display is Figure 3.4. It is a problem of the Android application that was not continuously able to control the valve here. However, it was considered that the purpose was possible to achieve, and the stage was shifted to the control circuit construction with PIC, because it was able to take the correspondence of the control circuit and the Android application.

4. Control Circuit that Used PIC(4)

The control circuit was made by using PIC16F886 with 24 existence of the I/O port. Figure 4.1 shows the schematic diagram. First of all, the connection to the Bluetooth communication module is described. Wiring was not especially needed because it had been connected to Arduino by using Wireless Proto shield in the prototype of Arduino. However, Wireless Proto shield cannot be used with PIC. Therefore, it is necessary to wire. However, it is different from 2.54mm because the pin interval of RN42XVP-1/RM is 2mm, and is the pin intervals of the substrate. Therefore, the pitch was converted by using 2.54mm pitch conversion substrate for Xbee of Akizuki Electronic Commerce. Because the pin array of RN42XVP-1/RM is the same as Xbee, it is possible to use without trouble, though the regulator for Xbee and for the power supply stabilization is mounted as for this pitch conversion substrate. Moreover, because the communication is made with the communication module with USART as the standard of serial communications, it only has to connect it with tx and rx on the communication module side respectively with PIC rx and tx. Here, it is impossible in the internal clock of PIC16F886 to communicate by this transmission rate though the transmission rate of serial communications is done from the result of the prototype by 11520bps. Therefore, the lock of the selah of 20MHz was newly used as an external clock. Finally, microSD was connected to SPI. However, it was not possible to operate with the program problem of PIC.

5. Android Application

Android application to control the bubble display was made by using Bluetooth. Development environment is the Pleiades All in One that help in the development, such as Japanized plug-ins containing many factors. Eclipse version is 4.4 Luna of Java integrated development environment. Moreover, the execution environment is not a built-in virtual device, which was carried out in Android4.2.2 equipped actual smartphone. Development of Android application is divided into two stages like the control circuit: the first stage that communicates with the prototype control circuit using Arduino and the second stage that expands and adds the function.

(5・1) Valve Control Testing Application(5)

Valve control application is composed of a Bluetooth communication function and a valve control function. The two functions are those that form the basis of a system constructed in this research, and take the integrity of the system which is the object of manufacturing a prototype in a minimum functions required to achieve the goal. The execution screen of the valve control application is shown in Figure 5.1 (a). The operation procedure of this application is described here. First, the selection of a pairing destination device from hopping up away by pressing the Connect button. Next, press all the buttons in which the valve number is opened. In this case, Set button for all open, and then press the Reset button if all to close. Finally, the send button is pressed, the opening and closing information of the valve is transmitted. 3Byte information in a single communication with this application is sent to the control circuit. 2Byte data
on this, corresponding each 16 portion closing and opening information of the respective solenoid valves so as to correspond to each bit; the head of 1Byte is a header section. However, the header portion indicates only the start of communications and the other information is not included.

5-2 Cooperation with Paint Functions, etc. (5)

The second stage of the Android application fabrication, is the additional pattern of drawing function by paint function and the additional ability to change valve of basic opening and closing time. The execution screen of this application is shown in Figure 5.2(b). As can be seen from the figure, the paint area for the upper part pattern drawing, and the buttons for the valve of basic opening and closing time setting have been added to the middle. The operation procedure of this application is described. First, it is to set the Bluetooth communication initially, but it will be omitted because it is the same as the valve control application. Then, the setting of the valve of basic opening and closing time is done. Choose the valve of basic opening and closing time and from among the 10 buttons in the middle, the time of setting opening and closing is set. All the units are msec. Next is a drawing of the pattern. Draw a pattern to be output to the display bubbles in square at the top. Redrawing to perform, the deletion of the pattern at the erase button should be selected. Finally, press the send button to perform the transmission of the valve opening and closing information. The operation principle of the present application is described. Some difference from the valve control application is that the information is transmitted even when a user presses the valve of basic opening and closing time by setting button. Data to be transmitted at this time are a signal of the header portion 1Byte data unit 1Byte. Header is different from the value at the time of sending the valve of opening and closing information. If the value is confirmed as a header, it becomes the valve basic opening and closing time setting mode, reading time from the value of the data unit, setting the basic closing time. Then, the process returns to the normal mode. As for the valve of opening and closing information, the basic is not changed from the structure of the valve control application, and by sending a one-off control signal with continuity, it displays some kinds of pattern on the bubble display.

6. Operating Results of the System

Actually the control circuit by the PIC connected to bubble display, two patterns (Figure 6.1) on the control application of Android terminal are drawn. It is as shown in Figure 6.2. As can be seen from the figure, it has different sizes of the generated bubble. Patterns by the buoyancy difference are collapsed somewhat by this. The cause of this is considered to be something due to the opening and closing time of the valves. The closed state of the valve raises the pressure of the air subsequently. Then, when opening the valve in a state where the pressure remains increasing, greater bubbles are generated by the increasing pressure. On the contrary, as it continued opening, pressure of the air is reduced, and small bubbles are generated. This phenomenon can be verified especially revealed in the pattern of Figure 6.2, which is performing the continuous opening of the valve. In other words, if the fewer the collapse of the simple figure patterns such as lines at present, the less it is and when it comes to complex shapes, the size of the bubble extremely changes and the pattern is lost largely. Several proposals exist as a solution to this problem. First, one pawl is to be briefly and repeatedly opened and closed. Although at the present scheme, a minimum off-time of the valve is 200msec, a large bubble is generated as in FIG even at this time. Therefore, for example, when the open command of 200msec came, the opening and closing of the valves per 1msec is performed finely cutting the bubbles. This method, however, is a method largely depending on whether the valve can follow much faster operation. This method also causes to shorten the life of the valve. Then, the second is to attach the porous material to the air discharge nozzle of the bubble display. In this case, the pursuit of a shape for fine bubbles by 3D printer is also necessary, although it is uncertain whether it makes the bubble finer or not. In addition, the current communication method to add each of the headers to the data is wasteful since it is transmitted with continuity, and there is a possibility of reading headers and data section wrongly. There may be no problems at present, but long-term watching the system is inefficient and should be improved because it is unstable.

![Figure 6.1 Pattern Input Screen](image1)

![Figure 6.2 Pattern in the Bubble Display Output State](image2)
As for Specific solutions, by increasing the header portions, are described a valve of basic opening and closing.

describe a valve of basic opening and closing time for the header portion. In addition, by one header, so that all patterns can be sent, are described the value for identifying the end of a number of data and the pattern data in the header. Here in order to identify the pattern data, the value set in the header is added to the end of the data. Thus, now it can freely increase or decrease patterns that only 16 rows can output. Along with this, there is a need to make improvements to the paint application. Specifically, although it now depicts a pattern on the grid of 16 × 16, eliminate square, this application draws a pattern freely and converts the transmission data by the arbitrary resolutions.

7. Summary

In this research, drawing patterns and performing the transmission by Android device, are its objective of the radio control of the bubble display. As a result, although the resolution is 16 × 16 on the Android terminal by drawing the pattern from the application that was developed in this research, has become possible to output a pattern by communicating with the control circuit by this Bluetooth. However, several new problems, are found. When First Pattern is output, it will make a difference in the size of the bubbles due to the difference in pressure, whereby the buoyancy difference problem occurs in the pattern, whose resolution is considered low. The second problem is of low resolution in the 16 × 16. In the future challenges, it is necessary to solve the newly discovered problem. First, as the solution of the pattern of the collapse of the problem, the solenoid valve should be operated at high speed, including two points of proposal to make improvements of the air discharge nozzle. Then, as for the problem that the resolution is low, the painting function, should be improved in order to attempt to improve the vertical resolution of the bubble display. Besides these problems, in addition to the implementation of the read / write function to the microSD, some mode—such as a pattern continuous display mode—should be added to the Android application.

Acknowledgment

We wish to thank Associate Prof. Gonda Eikou, Associate Prof. Hayamizu Yasutaka, and Prof. Aoki Kaoru for advice on this research. In addition, Mr. Sasaki Norimasa and Teshima Atsushi at Logicom., Ltd., who gave advice to the progress of my research.

References

(1) Asuka Yagi, Masataka Imura, Yoshihiro Kuroda, Oshiro Osamu : “360-Degree Observable”, International Processing Society of Japan Interaction, 1INH-5, 2011(in Japanese)
(2) Shin-ichiro Eitoku, Kunihiro Nisimura, Tomohiro Tanikawa, and Michitaka Hirose : “Study on Design of Controllable Particle Display Using Water Drops Suitable for Light Environment”, Proc. of the 16th ACM Symposium on Virtual Reality Software and technology, pp. 23-26, 2009(in Japanese)
(3) Seichirou Matsumura, Tarou Suzuki, Chuichi Arakawa, Itou : “The Intoroduction of The Interactive Art Using Bubbles And Sound”, Institute of Environmental Art and Design, Vol.2, pp.29-36, 2002(in Japanese)
(4) Tetsuya Gokan : “PIC16F1 Family Use Guide Book for Electronic Work”, Gijutsuhyoronsha, pp256-273, 2013(in Japanese)
(5) Syunya Kanashiro : “Android Programming Perfect Master“, Hidekazu System, pp756-757, 2013(in Japanese)