Applications of Temperature Control Based on Android Platform

Kun-Wei Lin, Zong-Han Wu, and Yuan-Li Lin

***Abstract*—In this study, we have implemented the temperature control system based on Android platform for industrial use. Our studied temperature control system includes Android platform, data server, and external control unit. Based on established standards, RS485 and RS232 communicate the necessary information from PID controller and Android platform, respectively, to each control unit to complete the temperature control system. In addition, the application programming interface (API) and RS232 driver were developed to enhance the maintenance and extension of the studied system. Our study reveals the advantages including, low cost, friendly interface, integrated record function, database, external control switches, and multiple temperature detecting sensors. Therefore, this research is suitable for using as an industrial temperature control.**

***Index Terms*—Android, API, Graphical, User interface.**

I. Introduction

As time progresses, industry has become more and more advanced. As electronic products are increasingly popular, traditional mechanical control system has dramatic change in operation. In recent years, more and more researches have been applied Android platform [1] in industry; many researchers proposed industrial machines control based on Android platform successively, through wired and wireless network transmission [2], [3]. However, temperature controller based on Android was rarely found. Temperature control has been widely used in industrial, such as pottery industry, heat treatment, drying equipment, food processing, and devices which were required to operate with stable temperature. Currently, most of traditional PID temperature controllers operate by button and display on seven segment display or LCD, which is difficult to operate since the using interface is unfriendly. To solve this problem, we developed a temperature control system with graphical user interface (GUI). In addition, the record was integrated in our temperature control system. The setting parameters, alarms, and the related information during the operating process can be saved in the database. Furthermore, a query program which provides user to query information saved in database was designed. In order to use RS232 to communicate Android platform and external control circuitry [4], [5], we developed RS232 driver by compiling android kernel. We also designed application programming interface (API) of Android platform by using native development kit (NDK). Through the designed API, commends and data can be communicated between Android platform and external circuit. In this work, the study is organized as follows: Section 2 presents our system architecture. System functions and features are introduced in Section 3. Finally, some conclusions and future works are summarized in Section 4.

Manuscript received January 8, 2014. This work is supported in part by the National Science Council of Taiwan, R.O.C., under Grant NSC 101-2221-E-324 -025.

Kun-Wei Lin, Zong-Han Wu and Yuan-LiLin are with the Department and Graduate Institute of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung, Taiwan, R.O.C. Post code: 413. (e-mail: kwlin@cyut.edu.tw, s10127621@cyut.edu.tw, s10027603@cyut.edu.tw).

II. System Architecture

The designed platform of system consists of Android, data server, and external control unit. The system architecture is shown in Fig. 1. A friendlly user-interface (UI) was provided in this system. For data server, the database which can store data and parameters has been developed. We also developed a query program for user to query data and parameter from database. The external control unit will work after received the user's commands from Android. The architecture of Android temperature control system is shown in Fig. 1. The Android system consists of five components: curve, product list, alarm, parameter and start/stop proceeding. Data server used to store operating data and user's setting parameters. A query program was developed to help the user query the record in the database. External control unit was composed of PID controller and control circuit. How to contol the external circuitry and switches are describe as follow. First, the user set operating parameters in the android platform. Second, the PID controller received the command parameters from Android. Third, the heater and switches which were connected to external equipments were controlled by PID controller. Then, the operating temperature will rise or down according to the setting command parameters.

III. Method

This system consists of Android, data server and external control unit, which are described as follows. And, our designed RS232 API will be described at last section.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

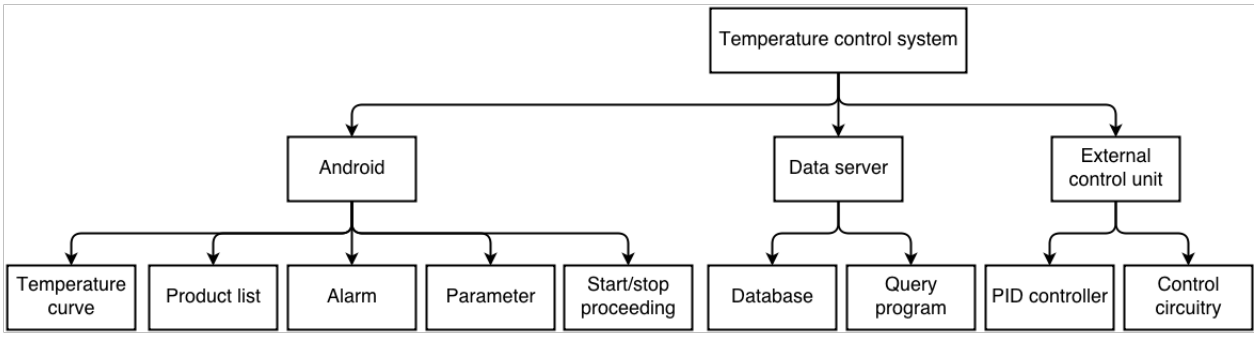
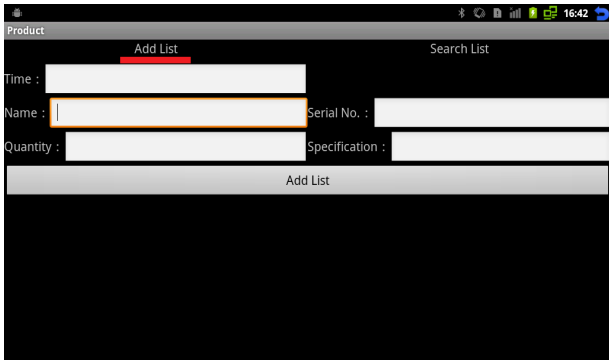
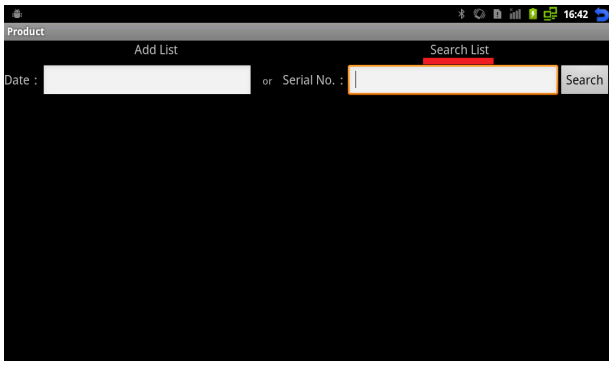


Fig. 1. The architecture of temperature control system based on Android platform.

Fig. 2. Temperature curve

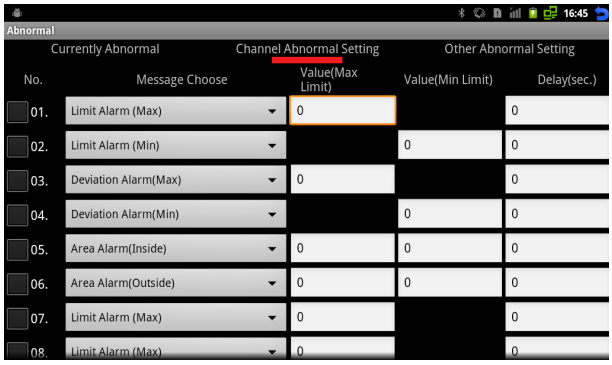


(a)

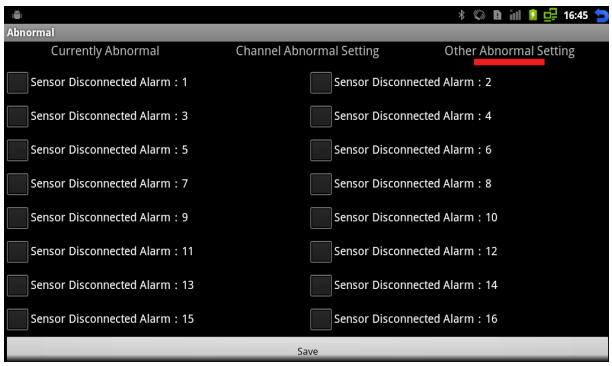


(b)

Fig. 3. Product list: (a) Add a new list, (b) Product search.



(a)



(b)

Fig. 4. Alarm of: (a) Channel abnormal, (b) Other abnormal conduction.

A. Android

(i) Temperature curves

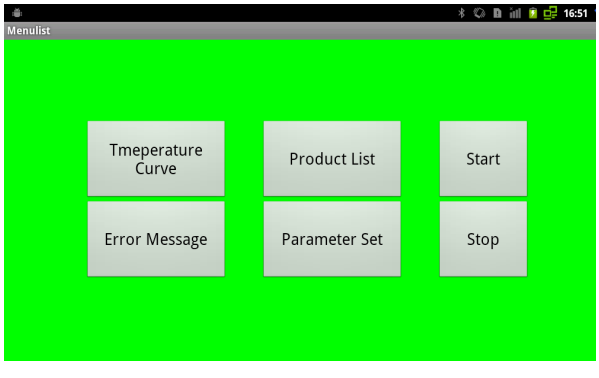
We used free chart library, "AChartEngine", to draw the temperature curves in real-time, as shown in Fig. 2. In the page of temperature cure, we can find the temperature curves, stage, present time, set temperature, and temperature curves during the whole manufacture period.

(ii) Product list

Users must set parameters of production list before the program starts. Production list also allows user query the production information which was stored in database. In this work, the production information includes date of manufacture, name of production, series number of production, quantity, specification of production. Production list is shown in Fig.3 (a) and (b)..

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Fig. 5. Parameter setting: (a) External control setting, (b) Step parameter setting, (c) PID setting, (d) Other setting, (e) Step control setting, (f) Input switch setting.



(b)

Fig. 6. Background changed when system was operated.

(iii) Alarm

The studied system provides multiple alarms as pre-warning. User can choose different sounds to distinguish hazardous conditions. Fig. 4(a) shows the setting of abnormal sensing channels which was divided into seven types including, high absolute alarm, low absolute alarm, maximum deviation alarm, minimum deviation alarm, in-zone alarm, out-zone alarm, and alarm delay. Other setting is shown in Fig. 4(b). Users can set sixteen break alarms. When the sensing value was over the setting thresholds, a pop-up window of warning appeared.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

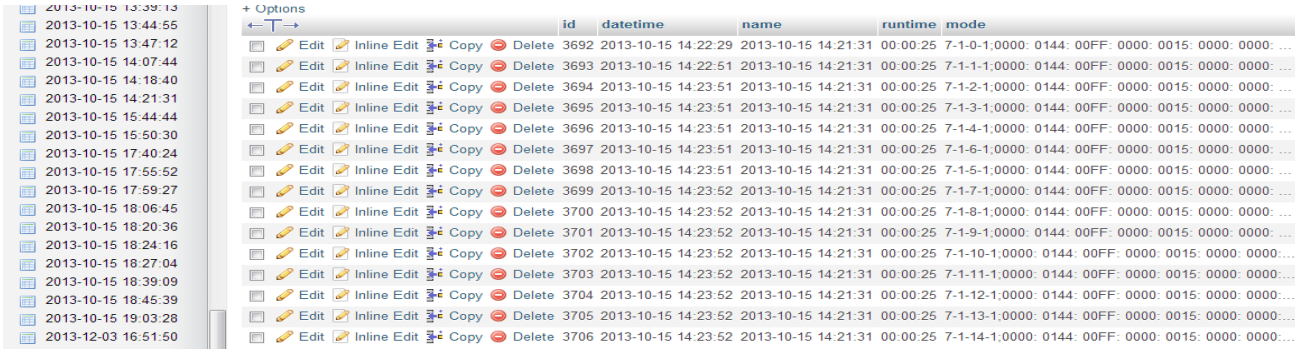
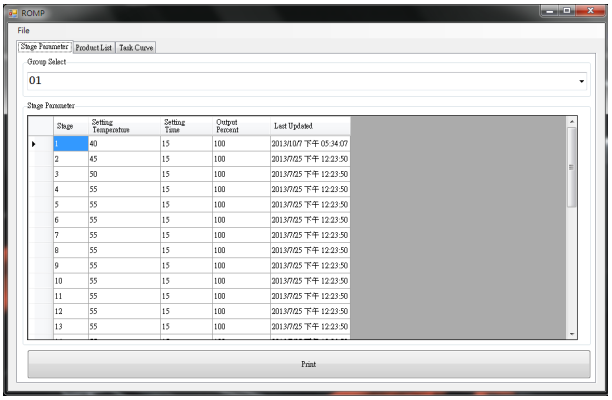
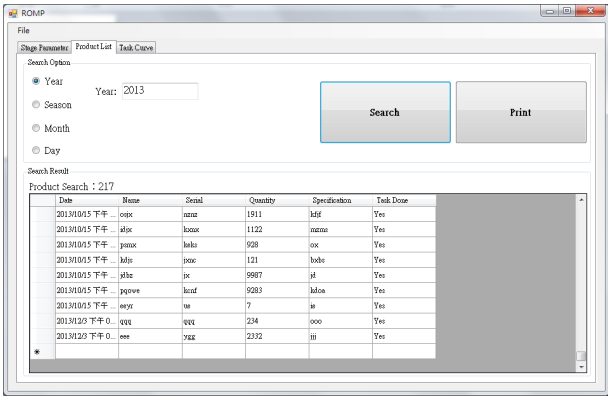


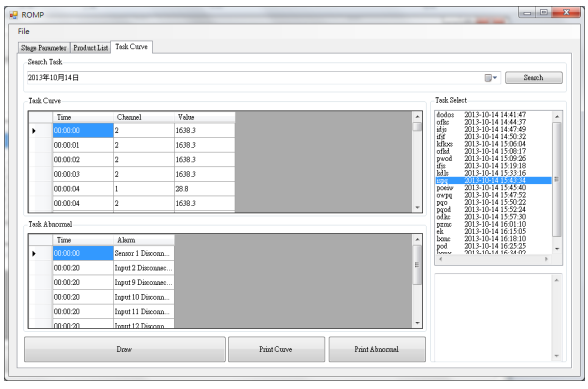
Fig. 7. Database with operating data and stored parameter.



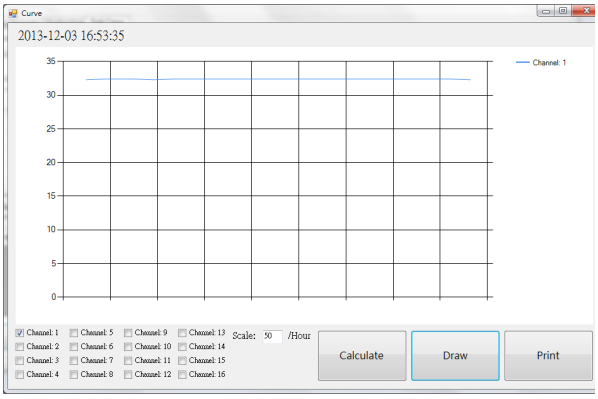
(a)



(b)



(c)



(d)

Fig. 8. Query program: (a) Stored parameter, (b) Product list, (c) Proceeding information, (d) Temperature curve draw.

(iv) Control Parameter

Fig. 5 (a-f) shows the parameters which were used in this work. Fig. 5(a) shows the setting page of external control switches. Each number represents a switch. When the number was chosen, the external control switch will open. Fig. 5(b) shows the setting interface of temperature control for different groups of different stages. For example, if we filled 5 in the blank of group and 3 in the blank of stage, it represents the temperature condition of production of No.3 of 3th group. Fig. 5(c) shows the setting of temperature value and temperature compensation. Fig. 5(d) shows the setting interface of PID parameters. It has the functions of read the current PID parameters, automatic calculation of PID, and setting of PID parameters. Fig. 5(e) shows the setting interface of different steps in stages. Fig. 5(f) shows the setting interface of input switches. In this work, the heating system has thirty-two steps can be set, each step has sixteen switches can be chosen. If the switches were opened, then the outer equipment can work to assist the temperature control.

(v) Start/Stop proceeding

These comments and feedback signal were transmitted between Android control unit and external control unit through RS232. The operating system will broadcast the services to other functions in background, and record the data to data server by

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

network. The background color changed from black to green when system was operated, as shown in Fig.6.

B. *Data server*

The data server consists of two components: database and query program. Database can store data and parameter. Query program allows user to query the recorded data and parameter from database.

(i) Database

Our proposed database developed by PHP and My SQL. With security issues for Android access database directly, PHP was used as a bridge between Android and MySQL. Also, using PHP can easily present information by web in feature. The developed database is shown in Fig. 7.

(ii) Query program

This program provided user to query all of information from database. Fig. 8 (a) shows parameters which were stored in database. As shown in Fig. 8 (b), user can query product list from database. The abnormal status extracted from database is shown in Fig. 8 (c). The temperature curve also can be presented, as shown in Fig. 8 (d).

C. *External control unit*

The architecture of external control unit is shown in Fig. 9. The external unit was composed by PID controller and control circuitry.

(i) PID controller

As shown in Fig. 9, the PID controller was composed of input signal, two output controllers, motor control, phase controller and communication unit. The different types of temperature sensors were used to measure sensing current and temperature of the chamber. The temperature in the chamber was controlled by a heat control circuitry and fan. Motor controller was used to control fan to let the temperature of chamber uniformly. A three-phase motor was used to make the received electric power more stable

(ii) Control circuitry

The main responsibility of control circuitry was to control external components. The external control

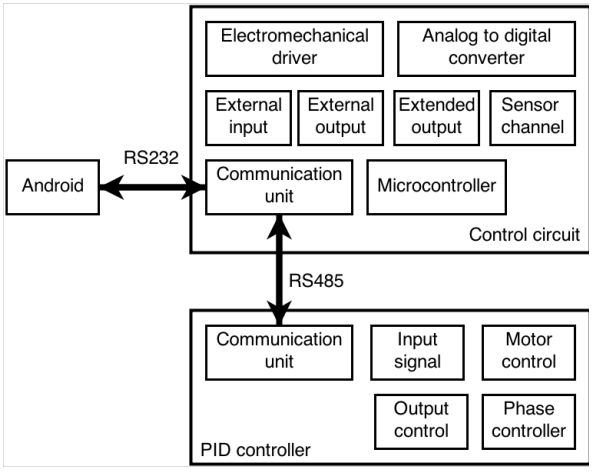


Fig. 9. The architecture of external control unit.

```
JNIEXPORT jint JNICALL Java_com_romp_itrsii_Linuxc_openUart(JNIEnv *env, jobject mc, jint i, jint mode)
{
    Connect RS232
}
JNIEXPORT void JNICALL Java_com_romp_itrsii_Linuxc_closeUart(JNIEnv *env, jobject mc, jint fd)
{
    Disconnect RS232
}
JNIEXPORT jint JNICALL Java_com_romp_itrsii_Linuxc_setUart(JNIEnv *env,jobject mc,jint fd,jint i)
{
    Set RS232
}
JNIEXPORT jint JNICALL Java_com_romp_itrsii_Linuxc_sendMsgUart(JNIEnv *env,jobject mc,jint fd,jstring str)
{
    Send message
}
JNIEXPORT jint JNICALL Java_com_romp_itrsii_Linuxc_sendMsgUartHex(JNIEnv *env, jclass mc,jint fd,jstring str, jint len)
{
    Send hex message
}
JNIEXPORT jstring JNICALL Java_com_romp_itrsii_Linuxc_receiveMsgUartHex(JNIEnv *env,jobject mc,jint fd)
{
    Receive hex message
}
JNIEXPORT jstring JNICALL Java_com_romp_itrsii_Linuxc_receiveMsgUart(JNIEnv *env,jobject mc,jint fd)
{
    Receive message
}
```

Fig. 10. Pseudo code of our RS232 API.

circuitry was composed of: analog to digital converter (ADC) [6], eight electromechanical drivers, sixteen external input switches, sixteen external output switches, four extended output switches, sixteen temperature sensors which can use different type temperature sensor including: J, K, B…etc. types of sensor. RS232 communication standard was used to transmit the command and data between Android control unit and control circuitry. RS485 communication standard was used to transmit the command and data between PID controller and control circuitry. We used digital commands to control the electromechanical drivers, external input switches, external output switches and extended output switches. Electromechanical drivers were used to control high current components such as fan or motor, etc. Sensor channels were used to measure temperature, and convert the sensing data by ADC into digital signal.

D. *RS232 API*

Because the Android standards don't provide API and drivers, we developed the RS232 API to communicate with RS232 through compiling Android kernel. We implement the API by Android NDK. Our developed pseudo code is shown in Fig. 10.

The functions of first two sub programs were used to open and close the transmission of RS232, respectively. The function of third sub program was to set up and spread the message through RS232. The chart array will be clear at first time. The set message will be send by RS232 designed in the fourth sub program. The fifth sub program was used to send hex codes to Android kernel. The hex codes were converted by receiving message. The kernel will convert hex codes and import to our developed application after receiving the hex codes. The received hex codes will be sent from Android kernel as shown in sixth sub program. This hex codes were converted by application process commands. Last sub program was used to receive message from RS232 serial port mode, then store this message into chart array. As the processes described above, Android platform with RS232 serial port has been implemented.

IV. CONCLUSION

In this work, we developed a novel temperature control system which integrates Android platform, data server, PID controller and control circuitry. For traditional PID temperature control system, user may confused by a number of entity buttons and Seven-segment display. Using

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

paper to record the operating information will cause the waste of paper and environmental damage. Also, traditional temperature control system was divided into controller and recorder. These will lead to increased cost of equipment in company. According to these defects, we have developed a novel temperature control system based on Android platform. Our studied temperature control system consists of three parts including, Android platform, output PID temperature control circuitry, and data sever. The main features of our studied system provide:

1. Based on Android platform, the studied system has a friendly graphic user interface. Hence, user can operate the studied system more easily.
2. Integrating temperature recorder into our studied system, a traceable system to product has been created. So, traceability and analysis to products are more easily.
3. Data and Curves were presented on the screen of database, which can reduce paper consumption.
4. Programmable multiple input/output switches make the temperature control more flexible.
5. Multiple temperature detecting sensors make the analysis of temperature field more easily and accurately.

Therefore, our studied temperature control system based on Android platform exhibits potential candidate for industrial temperature control.

REFERENCES

[1] Android Platform Official Site, http://www.android.com

[2] Andrei Drumea, "Control of Industrial Systems Using Android–Based Devices,"*Proceedings of the 36th International Spring Seminar on Electronics Technology,May 2013, pp. 405-408.*

[3] Nguyen-Vu Truong, Duc-Lung Vu, "Remote Monitoring and Control of Industrial Process Via WirelessNetwork and Android Platform", International Conference on Control, Automation and Information Sciences (ICCAIS), Nov. 2012, pp.340 – 343.

[4] Barry M. Cook, "IEEE 1355 data-strobe links: ATM speed at RS232 cost,"*Microprocessors and Microsystems, Volume 21, Issues 7–8, 30 March 1998, Pages 421-428.*

[5] Haruo Kobayashi, Kensuke Kobayashi, Hiroshi Sakayori and Yasuyuki Kimura,"ADC standard and testing in Japanese industry", Computer Standards & Interfaces, Volume 23, Issue 1, March 2001, Pages 57-64.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)