HUMAN-COMPUTER INTERFACE DEVELOPMENT OF WIRELESS MONITORING SYSTEM BASED ON MINIGUI

Zhihua Diao[1], Chunjiang Zhao[1,2], Xiaojun Qiao[2,*], Cheng Wang[2], Gang Wu[1], Xin Zhang[2]

[1] *University of Science and Technology of China, china, 230024;*

[2] *National Engineering Research Center for Information Technology in Agriculture, china, 100097*

* Corresponding author, *Address: Shuguang Huayuan Middle Road 11#, National Engineering Research Center for Information Technology in Agricultur, Beijing, China,100097, Tel:+86-10-51503348, Fax:+86-10-51503449, Email: qiaoxj@nercita.org.cn*

Abstract: As the development of embedded GUI, the human-computer interface is more and more important in embedded system. In order to achieve simple operation and visual display, the better to achieve human-computer interaction，we use MiniGUI to develop human-computer interface of the wireless monitoring system. The hardware of the system includes the S3C2410X、TFT LCD and touch screen with four line resister, the relative software is ARM-Linux. This human-computer interface is used in the wireless monitoring system, provides a visible and friendly platform for customers, and runs steady in the real system.

Keywords: MiniGUI, Human-computer interface, Monitoring system, Wireless communications, Installation agriculture.

1. INTRODUCTION

As the constant development of the facilities agriculture, the requirements to the supervises-control system of agriculture are more and more high. In order to realize the human-computer interface better, set the system procedure and equipment more convenient, look into the system real time data immediately, manipulate correspond of control equipments, such human-computer interface with some merits such as easy operation, reliable , and taking up a little resource, is required to realize the control of modernization, intelligence and precise. The user interface is the foundation of human-computer interaction; the user control and computer respond all carry out through the user interface. Unceasing development and wide application of embedded system have injected new vitality to agricultural fields. Embedded system develops from the initial systems in the form of single-chip as the core of the programmable controller to the present embedded system based on Internet. The present hardwares of embedded system include ARM, DSP, Power PC and so on, the operating systems running on them include ARM-Linux, ucLinux, uc/OS, Wince and so on(An et al., 2005). From the 90s of the 20th century, the embedded systems introduced graphical user interface support system, which runs on embedded Linux operating systems are Nano-X, OpenGUI, QT - Embedded and MiniGUI. Because of networking function components without proper adjustment, Nano-X has not too many ready-made application programs to be used. Because the kernel of OpenGUI is achieved by compiled language, its portability will be affected. QT-Embedded has high hardware requirements and the structure is too complex, is hard to be bottom expansion, customization and transplant.

MiniGUI is such a light open source graphic interface support system with the features of less resources occupancy, high performance, high reliability and configurable(Sun et al., 2005). It can not only run on the VxWork, pSOS, uc/OS, but also on the Linux operating environment which the kernel is 2.4 edition. We use ARM9 to realize this wireless monitoring system, and the operating system is ARM-Linux. We can conveniently set system program and parameters of the control devices, display Real-Time system working conditions and environmental information according to human-computer interface developed by MiniGUI. In order to satisfy the real-time requirements of system, MiniGUI is compiled to Threads mode.

2. SYSTEM HARDWARE STRUCTURE

In order to overcome the less coordination, wiring and maintenance disadvantages of the traditional monitoring system as a result of using PC - controller working mode and wire communications, we design the universal wireless monitoring system. It is a new type of monitoring system developed by the combination of wireless communication technology and information-processing technology, adopts monitoring center-front-end wireless module, its whole structure is as shown in Fig. 1. Monitoring center is the core of the entire system, realized by ARM9 system which core is S3C2410X. LCD is a

6.4-inch TFT-LCD display produced by yuan tai company, its type is PD064VT5, its pixel is 640X480. Monitoring center is responsible for receiving the required monitoring information from wireless modules, sending operational orders to wireless modules, to control the action of control devices linked to wireless modules. We use the existing module JN5121 as wireless modules, and the wireless technology is Zigbee. Wireless modules are deployed away from the monitoring center in the monitoring point, are responsible for completion of communication link to monitoring center, front-end monitoring information collection and responding to control orders that the monitoring center sends.

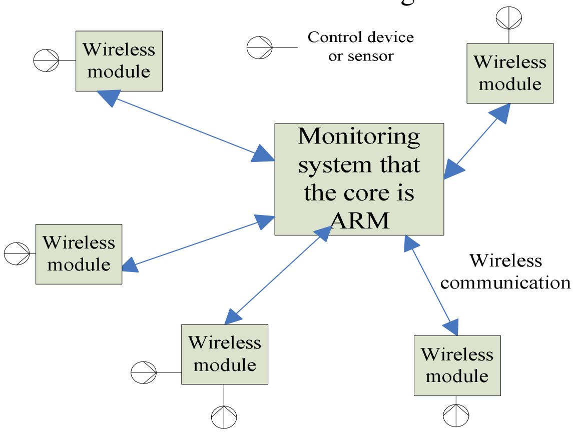

Fig. 1. system whole structure chart

System can not only be set the parameters of control devices and alarm parameters, but also be set logical condition and system operation programs. Then system will control the corresponding devices to work according to the setting program. System can not only store the setting parameters in databases, but also store the data that received from front-end modules. Meanwhile system also supports the control of the most primitive ways: manual control mode.

3. DESIGN OF SYSTEM INTERFACE

According to the system requirements, we design system software structure as shown in Fig. 2. Then we focus on the introduction 、realization and development of system human-computer interface module below.

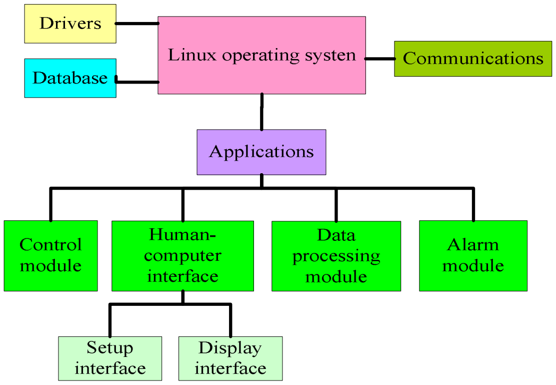

Fig. 2. system software structure chart

3.1 Introduction of system interface

To achieve system functions, we design system human-computer interface as shown in Fig. 3.

System input setting interface can be set the parameters of control equipment and the definition of various types of sensors.

We can choose control methods according to the actual needs in program setting interface, for example, using poll irrigation as a mean of achieving the purpose of irrigation.

Setting Logic condition is set up in interface of logic condition setting, besides using linked conditions to achieve more complex logic control conditions.

Alarm interface can be set up to achieve the output numbers and the setting of alarm time.

Display of real-time data interface can show real-time irrigation program working, sensor information and some information of control equipment.

Manual operation interface can be achieved by controlling the most primitive ways: manual control, including manual suspension and manual operation.

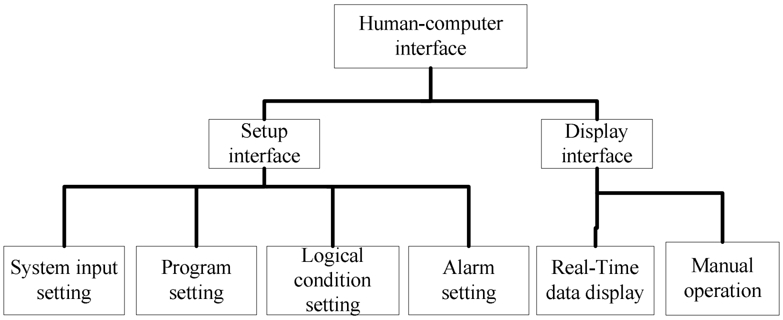

Fig. 3. System human-computer interface chart

3.2 Realization of system interface

MiniGUI provides three kinds of window, Main window, Dialogue box and Control window (Feynman, 2003).

Generally, every MiniGUI application should sets up a main window as the main-interface or start-interface. Main window usually includes some sub-windows, which are usually control window or user-defined window class, and the application can also create other types of windows, such as dialogue box and news box.

In MiniGUI, Dialogue box is a special kind of the main window, which just concerned communications with users---output information to users, but more are for input. It can be interpreted as a kind of main window after sub-classified.

Controls can be understood as the sub-window of the main window, and it acts as the main window. Not only receive the keyboard, mouse and other external input, but also could output in its region-- were just all of its activities restricted in the main window.

In this wireless monitoring system, using Modal Dialog create MiniGUI main window as the main interface, and the six sub-pages which representatives of the functional modules all express with a single button in this main interface. All these buttons are taking the form of arrays in the main interface controls layout definition. When opening the main interface, and clicking a button, then a sub-interface will be pop up correspondingly, according to the sub-interface functions, all those setup of system input, irrigation program, real-time-data display and other operations can be carry out..

3.3 Development of system interface

MiniGUI is a graphical user interface support system, and the concepts of GUI programming are also applicable to MiniGUI programming, such as windows and other time-driven programming (Kang et al., 2006). The system is used in Threads model, which is based on messages and window management mechanism of POSIX thread, MiniGUI intercommunicates with outside through receiving messages. Messages are produced by system or application procedures, system produces message for input event and response of input event meanwhile. Applications could complete a particular task through message, or communicate with other application window. System send message to the process of application window which contains four parameters: window handle, message identifier and two 32-bit message parameters. Applications must promptly deal with the messages that delivered to its message queue, and the application generally handles the message queue news through a message cycle in MiniGUIMain function

(Feynman, 2003). Message cycle is the cycle of body. Programs obtain message from the message queue continuously using GetMessage function in the body, then send message to the specified window using DispatchMessage function, which is same to call the designated window process, and impart information and parameters.

The various functional modules program in this system is realized by modular design, and the interfaces of modules are realized in the form of model dialog. Meanwhile system uses information-driven mechanism to complete the realization of the system function. Callback function of the application procedures verdicts incidents occurred on the current interface based on receiving message identifier, then transfers the corresponding processing function to operate. There are six buttons totally in the main interface, and each button has an identifier. When the corresponding button is pressed, the system will receive message and send this message to main system process function, then call the corresponding functional modules and show the corresponding sub window based on different identifiers. The specific system main callback function is as follows:

```
   static int MainwindowProc (HWND hDlg, int message, WPARAM wParam, LPARAM
lParam)
   {   switch (message) {
         case MSG_INITDIALOG:     return 1;
         case MSG_COMMAND:    {
           int id = LOWORD(wParam);
           int nc = HIWORD(wParam);
  if (id == IDC_SYSTEMINPUTSETBUTTON && nc == BN_CLICKED)
   {   //enter into system input set sub interface when pushed
    DlgSysteminputset.controls = CtrlSysteminputset;
    DialogBoxIndirectParam (&DlgSysteminputset, hDlg, SysteminputsetProc, 0L);}
  if (id == IDC_IRRIGATIONSETBUTTON && nc == BN_CLICKED)
  {   //enter into procedure set sub interface when pushed
    DlgIrrigationset.controls = CtrlIrrigationset;
    DialogBoxIndirectParam (&DlgIrrigationset, hDlg, IrrigationsetProc, 0L);}
  if(id==IDC_LOGICALCONDITIONSETBUTTON && nc == BN_CLICKED)
  {   // enter into system logical condition set sub interface when pushed
  DlgLogicalconditionset.controls=CtrlLogicalconditionset;
   DialogBoxIndirectParam(&DlgLogicalconditionset, hDlg, LogicalconditionsetProc, 0L);}
  if (id == IDC_ALARMSETBUTTON && nc == BN_CLICKED)
  {  // enter into alarm set sub interface when pushed
    DlgAlarmset.controls = CtrlAlarmset;
    DialogBoxIndirectParam (&DlgAlarmset, hDlg, AlarmsetProc, 0L);}
  if (id == IDC_DATADISPLAYBUTTON && nc == BN_CLICKED)
  {// enter into data display sub interface when pushed
     DlgDatadisplay.controls = CtrlDatadisplay;
     DialogBoxIndirectParam (&DlgDatadisplay, hDlg, DatadisplayProc, 0L);}
  if (id == IDC_MANUALLYOPERATEBUTTON && nc == BN_CLICKED)
  {   // enter into manual operation sub interface when pushed
  DlgManuallyoperate.controls= CtrlManuallyoperate;
  DialogBoxIndirectParam (&DlgManuallyoperate, hDlg, ManuallyoperateProc, 0L);}
```

4. CONCLUDING REMARKS

In this system, we have transplanted MiniGUI - 1.3.3 version to the ARM development transplant board successfully, received executable file using ARM-Linux-GCC compiler to cross compile system procedure on the PC. We have copied executable file to the development board and set it as startup operating procedure. Then system human-computer interface have run on the development board and worked well on the ARM-Linux operating system. In order to satisfy the characteristics of operating system easily and adapt it as the habit of operation of touch screen input, therefore, in the choice of man-machine interface controls, the use of button controls, frame composition controls, the drop-down controls will be adopted regularly to make the users operate conveniently. Based on the embedded system MiniGUI developed can achieve beautiful interface, complete functions, good real-time, better scalability and maintainability, and other advantages can be widely used in the development of man-machine interface of embedded system(Yang, 2004).

ACKNOWLEDGEMENTS

This research is supported by the National High Technology Research and Development Program of China (863 Program, Grant 2006AA10A311 and Grant 2006AA10Z253) and Beijing Science & Technology Program（Z0006321001391）.

REFERENCES

Beijing Feynman Software Technology Co., Ltd., (2003). MiniGUI Program Guide For MiniGUI Version 1.3.x.

Beijing Feynman Software Technology Co., Ltd., (2003). MiniGUI User Manual For MiniGUI Version 1.3.x.

Chengjin An, Maoyang Sun, Li Po. 2005. Development of Graphical Interface Based on Embedded Linux System, Modern Electronics Technique, 20:108-113.

Jianwu Zhang, Zuopeng Xu, Dong Ping. 2006. Multi-serial Port Communication Based on MiniGUI-Threads and MiniGUI-Lite, Chinese Journal of Electron Devices, 3:985-988.

Shaohua Sun, Lizhong Xu. 2005. Graphics User Interface Based on Embedded Linux System, Microcomputer Development, 10:123-125.

Tieying Kang, Li Wei, Wengang Zheng etc. 2006. Human-computer Interface Design of Irrigation Controller Based on MiniGUI, Journal of Liaoning Normal University (Natural Science Edition), 1:46-48.

Xiaojun Yang. 2004. Research of Embed System Based on Linux in GUI, Modern Electronics Technique, 15:89-91.

Xiaoyan Zhao, Ma Qi. 2005. GUI Support System for Embedded System Based on Linux-MiniGUI, Computer and modernization, 1:10-12.