

ARM Based Interactive Electronic Board

Sayali Kale, Ruchira Gujar, Priyanka Karanje, Juili Cholachgudd, *Prof. Meenal Mungi*
Department of Computer Engineering, MMCOE , Pune-52

Abstract - Using chalk is not only laborious, but it also produces a lot of chalk dust, polluting the air and adversely affecting the health of the students and teachers. This design uses ARM9 board for hand-held terminal hardware and uC/OS-II embedded real time operating system for hand-held terminal software. Handwriting signal will be converted to electrical signal and transferred to the PC terminal by wireless transmission module. Then the PC terminal software will process the handwriting signal and display it in a larger form by projector. The Linux operating system is used as its free ware and has support of the device drivers of the interfacing various modules.

I. INTRODUCTION

At present, multimedia is widely used in classroom teaching. However, it cannot fulfill the needs of teaching. Teachers still have to write on the blackboard with chalk. Using chalk is not only laborious, but it also produces a lot of chalk dust, polluting the air and adversely affecting the health of the students and teachers. These few years, many prominent firms are dedicated to developing the technology of electric board such as: Hitachi, Panasonic.[1,2] But these companies just replace traditional blackboards with larger electric boards which cost more than ten thousand Yuan. At the same time, these electric boards are ineffective. In this design, Wireless Electric Board costs just one-fifth of the ordinary electric board. In addition, it makes classroom teaching be more efficient.

This design uses ARM processor as a heart of this hand-held terminal hardware and Linux operating system for hand-held terminal software. Handwriting signal written touch screen will be transferred to the PC terminal by wireless transmission module display it in a larger form by projector. In many embedded operating systems, Linux, with its unique advantages, takes a large share of the embedded field. The Linux operating system is used as its free ware and has support of the device drivers of the interfacing various modules.

Qt is a framework to create cross-platform applications. Qt is used to create amazing GUI applications quickly and fairly easily. Visual Studio or X code is used to create applications in Windows and Mac respectively. Qt is similar to these tools which help to design and code your application. But, the real advantage of Qt lies in the fact that your application can be made to run on a variety of operating systems without you having to change your code. The application will run on Windows, Mac and Linux in the same way.

II. SYSTEM OVERVIEW

This design uses ARM9 board for hand-held terminal hardware and embedded real time operating system for hand-held terminal software. Handwriting signal on a book-size touch screen will be converted to electrical signal and transferred to the PC terminal by nRF2401 wireless transmission module in the band of frequency 2.4GHz, then the PC terminal software will process the handwriting signal and display it in a larger form by projector. The structure diagram of this system is shown in Fig.1.

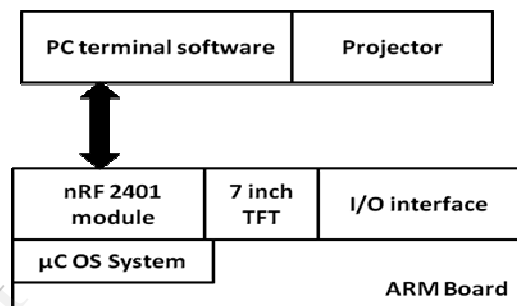


Figure 1. Structure diagram of this system

III. HARDWARE DESIGN

Hardware design includes hand-held terminal hardware and PC terminal. Hand-held terminal hardware consists of ARM9 processor on board and NAND and NOR Flash memory. The processor is interfaced to 7 inch Touch Screen, memories, and various input output modules along with the wireless transceiver. The hand-held terminal hardware design structure of the system is shown in fig .2.

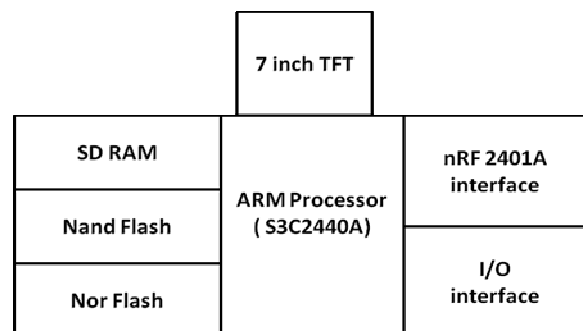


Figure 2. Hand-held terminal hardware design structure

A. ARM9 board

ARM9 board is constituted by Samsung's S3C2440A, 7 inch TFT, 64M SDRAM, 64M Nand Flash, 2M Nor Flash and four interfaces, etc. The S3C2440A is developed with ARM920T core, 0.13um CMOS standard cells and a memory compiler. Its low-power, simple, elegant and fully static design is particularly suitable for cost-low and power sensitive applications. It adopts a new bus architecture known as Advanced Micro controller Bus Architecture (AMBA). 7 inch TFT is LCD touch screen, is used for receiving handwriting signal. Nor Flash memory is used for BIOS; Nand Flash is used for installing the operating system of uCIOS-II. [3.4]

B. NAND Flash Controller

S3C2440A boot code can be executed on an external NAND flash memory. In order to support NAND flash boot loader, the S3C2440A is equipped with an internal SRAM buffer called 'Steppingstone'. When booting, the first 4 Kbytes of the NAND flash memory will be loaded into Steppingstone and the boot code loaded into Steppingstone will be executed. Generally, the boot code will copy NAND flash content to SDRAM. Using hardware ECC, the NAND flash data validity will be checked. Upon the completion of the copy, the main program will be executed on the SDRAM.

C. SRAM Specification

The S3C2410A's memory controller provides memory control signals required for external memory access. It has total 8 memory banks: Six memory banks for ROM, SRAM, Remaining two memory banks for ROM, SRAM, SDRAM, etc.

IV. SOFTWARE DESIGN

Software design includes hand-held terminal software and PC terminal software. Hand-held terminal software is designed to collect, process and transmit handwriting signal. On the other hand, PC terminal program software design is used for receiving signals from Hand-held terminal by bluetooth wireless transmission module.

1) Hand-held terminal software architecture

Hand-held terminal software is divided into four parts: Boot Loader, Drivers, Kernel, and Applications. Primarily, Boot Loader is used for initializing necessary peripherals and interrupt vector table in processor. Secondly, Drivers, which refer to LCD driver, touch screen driver, Nand Flash driver and wireless module driver are used for supporting communication with Kernel. Thirdly, Kernel chooses real time multitasking kernel μ Cos-II. Finally, Application can be available based on these above mentioned three dispensable parts: Boot Loader, Kernel and Drivers. Software architecture is shown in Fig 3.

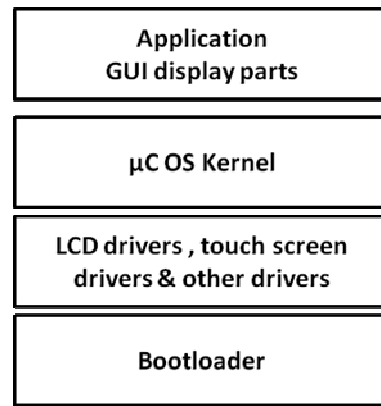


Figure 3. Hand-held Software architecture diagram

a) Boot Loader design

After connecting the power supply, the NAND Flash controller of SOC will automatically copy the primary 4KB of data to Stepping Stone, while the Boot Loader will be preserved in the beginning 4KB of NAND Flash. Here as Boot Loader will realize these functions: shutting off watchdog, setting system timer, initializing MMU, defining stack and modifying NAND Flash controller.

b) Transplantation of Linux Kernel

Linux is a real-time multitasking kernel, responsible for process creating, scheduling and communication. The porting of Embedded Linux is divided into the following steps: obtaining source codes; building cross compiler environment; porting Linux boot loader (Uboot or VIVI), then configure and compile the kernel, porting and loading the embedded file system, developing and debugging of application program, program downloading. In this paper, we mainly introduce the porting technology of Linux kernel. [5]

c) Drives design

Drivers provide the interfaces of hardware operating modules for Kernel and Application, including LCD driver, touch screen driver, USB drivers, NAND Flash driver, and wireless transmission module driver.

d) Application Design

Application is composed of four parts: Qt GUI display, touch information gathering, processing program and wireless sending. The touch screen controller will not lunch interrupt signal to the processor until receiving touch handwriting signal. At the same time, processor opens interrupt mode. In this mode, processor extracts information of signals in relative registers by inquiring Interrupt Request Number. The information will be kept in Global Touch message queue. After these steps, processor will return to initial mode. Meanwhile, the processing program starts to inquire and process the information in the Global Touch message queue as long as it's non-empty. In

succession, processing program will launch the information to GUI display part for displaying and wireless sending part for transmitting. The flow chart of application design is shown in Fig.4

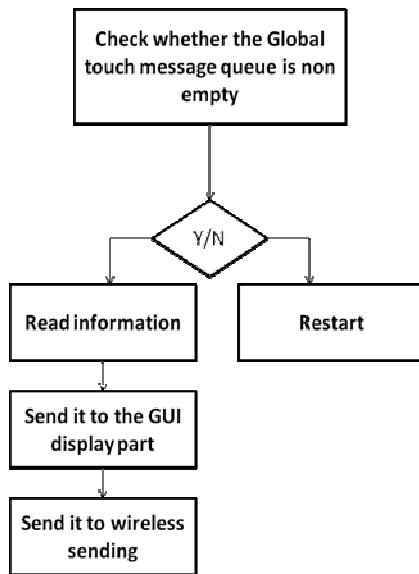


Figure 4. Application program design

2) PC terminal software architecture

PC terminal software includes PC terminal handler and USB wireless receiver module driver. PC terminal handler receives wireless signal by USB wireless receiver module thus it can get the information coming from the hand-held terminal. After being analyzed and processed, the information will be displayed in PC terminal and projected by API interface. The flow chart of the PC terminal handler design is shown in Fig.5 :

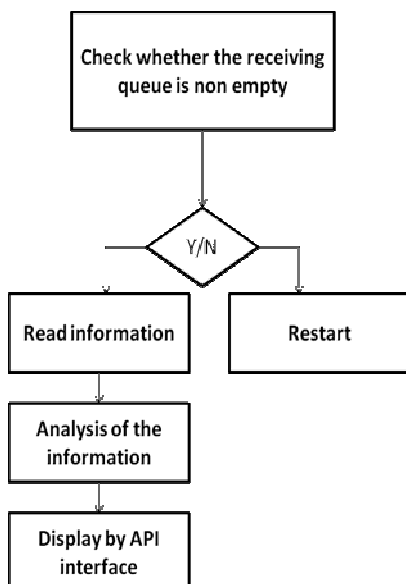


Figure 5. PC terminal handler design

V. BLUETOOTH as WIRELESS MODULE

Bluetooth is an open standard for short-range radio frequency (RF) communication. Bluetooth is a low-cost, low-power technology that provides a mechanism for creating small wireless networks on an ad hoc basis, known as piconets. It operates in the unlicensed 2.4 gigahertz (GHz) to 2.4835 GHz Industrial, Scientific, and Medical (ISM) frequency band. Bluetooth employs frequency hopping spread spectrum (FHSS) technology for all transmissions. FHSS reduces interference and transmission errors and provides a limited level of transmission security. Bluetooth also provides for radio link power control, where devices can negotiate and adjust their radio power according to signal strength measurements.

Each device in a Bluetooth network can determine its received signal strength indication (RSSI) and make a request of the other network device to adjust its relative radio power level (i.e., have the transmission power incrementally increased or decreased). This is performed to conserve power and/or to keep the received signal characteristics within a preferred range. Linux has two popular Bluetooth stacks, BlueZ and Affix. The BlueZ stack is included with most Linux kernels and was originally developed by Qualcomm. The Affix stack was developed by Nokia.

VI. APPLICATION SOFTWARE

Qt is a cross platform development framework written in C++, but makes extensive use of a special code generator (called the *Meta Object Compiler*, or *moc*) together with several macros to enrich the language. Qt can also be used in several other programming languages via language bindings. It runs on the major desktop platforms and some of the mobile platforms. It has extensive internationalization support. Non-GUI features include SQL database access, XML parsing, thread management, network support, and a unified cross-platform application programming interface (API) for file handling. Qt is made up of modules. All modules have a common scheme and are built from the same API design ideas. Qt extends C++ with macros and introspection, all code is still plain C++. Qt has Cross platform applications built from one source. It builds native applications with native look and feel. Easy to (re)use API, high developer productivity, openness,. Qt is open source software, but the development is led by Qt development Frameworks Embedded target platforms: Windows CE, Symbian, Maemo, Embedded Linux (Direct frame buffer access)

VII. SYSTEM OPERATION

1. We have to download the uboot, kernel and the file system of so that the entire operating system is present and ready to use.
2. The device drivers for various interfaces are configured in kernel.

3. We compile these using tool chain and the download it through the serial port step-wise by configuring the HyperTerminal for 115200 baud rate and at com1 port.
4. Once the HyperTerminal is configured we give the various commands on the command prompt (#) of the serial port and thus entire Operating system is downloaded.
5. We compile the Code for Qt and then download the application through the USB ports on the PC and the board.
6. The board is power on and first configures the board for the Wireless communication.
7. We turn on the board Bluetooth connection and connect it to PC through the bluetooth dongle connected to PC Thus a network is formed between PC, router and the board.
8. Once the connection is established we turn on the serial for and give the command on the root # prompt as #./start.sh to start the application on the board by terminating the server.
9. After the connection has been established between the board and PC we can see what is written on the scrabble pad of the application on the PC.
10. The PC is connected to projector and the written data is projected.

VIII. CONCLUSION

This design of hand held equipment is build using ARM 9 processor S3C2240 which can interface to wireless module and 7 inch touch screen. The data written on the screen is transferred to PC through Bluetooth. The use of Linux operating system helps in specification for multitasking and also for writing the application software on the kernel. The application software is written in Qt which is ported to the operating system. The programming in 'Qt' is similar to C++ which can be easily compiled. VNC server is also implemented in the application and VNC viewer is used for the PC through which data written on the hand held equipment is transferred to PC which is displayed on projector.

Moreover this technology helps in avoiding use of traditional chalk & board thus eliminating health issues caused by chalk dust. This hand-held equipment is cost effective compared to present interactive white boards and is portable many features.

REFERENCES

- [1] <http://panasonic.neti>
- [2] <http://www.hitachi.com>
- [3] <http://www.samsungsemi.com>
- [4] Di WU, Yang Zhang, Baoding Zhou "Wireless Electric Board Based on an ARM-based Embedded System ", IEEE International conference computer science and information technology (ICCSIT) Volume 5, pp 268-270 .September 2010.
- [5] Ian McLoughlin, Anton Aendenroomer "Linux as a Teaching Aid for Embedded Systems", IEEE, 978-1-

4244-1890- July 2007.

- [6] Shao Guojin, Shen Yunqin "Embedded Linux Core Transplantation and Development of Handheld Terminals System" IEEE, Volume 5, pp 583- 586 978-1-4244-5586-October 2010.
- [7] Christof Ebert, "Using Linux for Real-Time Applications", Published by the IEEE Compute Society, 0740 -7459 April 2004.
- [8] Chun-yue Bi, Yun-peng Liu , Ren-fang Wang , "Research of Key Technologies for Embedded Linux Based on ARM", International Conference on Computer Application and System Modeling (ICCASM), Volume 8 , pp 373 -378, June 2010