International Journal of
Scientific Engineering and Technology Research

www.ijsetr.com

ISSN 2319-8885
Vol.04,Issue.36,
September-2015,
Pages:7740-7743

Embedded Environment Control System using Linux OS

V. JYOSTHNA[1], K. SHIVA SUNDARI[2]

[1]PG Scholar, Dept of ECE, Sreenidhi Institute of Science and Technology, Hyderabad, TS, India,
E-mail: jyothsnavk@gmail.com.

[2]Associate Professor, Dept of ECE, Sreenidhi Institute of Science and Technology, Hyderabad, TS, India,
E-mail: kssundari@sreenidhi.edu.in.

Abstract: This paper realizes a Linux platform based Data Acquisition and Control System that helps the user to monitor and control the data. This project designs an environment control system with combining Embedded and ZigBee wireless sensor network technology. Using ZigBee wireless sensor network to complete acquisition and transmission of environment parameters and using Raspberry pi board to achieve precise control of the environment as system data controller. It consists of application program written in C and C++ in Keil and QT Integrated Development Environments for accessing data through the serial port and updating the data based on priority reversal and priority inheritance techniques. This system uses BCM3825 an ARM 11 architecture based Processor portability with Linux operating system (Raspbian OS). It makes the system more real time and handling various processes based on multi tasking and reliable priority mechanisms.

Keywords: Raspberry Pi, Mems, Zigbee, Linux.

I. INTRODUCTION

The embedded systems which use micro-controller such as 8-bit microcontroller as the main controller has been widely used in various fields, but most of these applications are still in the low-level stage of stand-alone use of the embedded system. It is feasible and forward-looking to apply the high-performance 32-bit microprocessors, embedded Linux system and Qt / embedded GUI application to practical industrial control in certain occasion. Embedded front end machine uses Raspberry Pi, which has an arm11 processor and embedded Linux operating system. The sensor node is responsible for collection of environment information (such as temperature, Position, and Vibration). The signals collected by the sensor through the A/D conversions are sent to MCU processing. The microcontroller is connected to LCD to display the values of. Controlling part is also included in the module by connecting the 5volts dc fan to the microcontroller which controls the fan. The Zigbee communication module changes the data into data packets of zigbee communication protocol which are transmitted to the coordinator node. The coordinator node after receiving the data packets from the sensor node performs handshake communication by sending a confirmation language source to the sensor node to complete a full Zigbee wireless communication process. On the other hand it should upload the data to the ARM master unit through the serial port. The ARM master unit gathered the collected information data.

II. SYSTEM ARCHITECTURE

We need to design a system that is collecting the information from the sensors and updating the latest information regarding the environmental condition. The main purpose is to design an application that will monitor, analyze and display the information continuously in a graphical format. Since the embedded system is resource-constrained, the design mode of the GUI of the traditional PC is not acceptable because its memory consumption is relatively large and takes up more CPU time. The control system uses Qt /Embedded under embedded Linux as its GUI development platform, which can fully satisfy the restriction of embedded system resources.

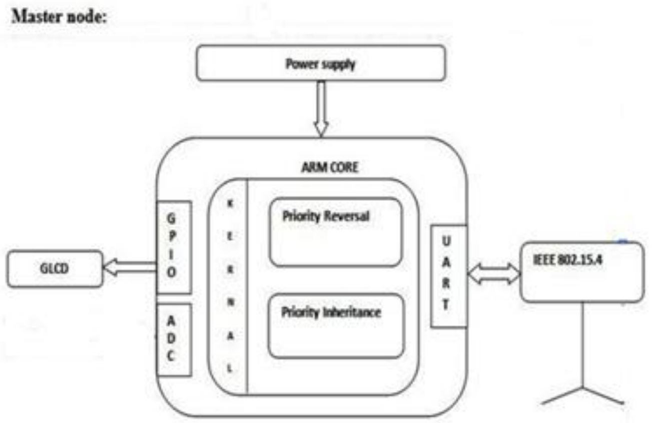


Fig.1. Master Node.

The various sensors are connected to the slave node via an ADC. The ADC is connected to the 8051 at the GPIO pins. The data from the sensors will be moved to the connected pins that will be taken as an input of the system, the received input data will be displayed and transmitted to the Master node through zigbee protocol as shown in Fig.1. On the other end the master will receive the data and using priority mechanisms. Qt is used as the programming GUI interface on the master side, as Qt uses C + + as its programming language, it can implement hybrid programming with linux-C. Write the linux system calls as parts of the slots functions which can respond to specific signals in order to achieve the combination of Qt / Embedded and linux-C. Of course, to achieve reading and writing of a specific device file, there must be device drivers which provide reading and writing operation interface functions. Therefore, we need to complete the configuration of the drivers of sensors and other external peripherals. The Intelligent Monitoring System uses QT to complete GUI on the ARM head-end machine of data collected by a variety of sensors as shown in Fig.2.

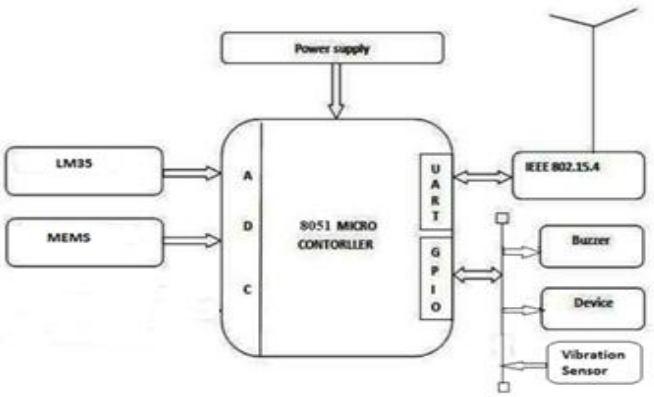


Fig.2. Slave Node.

III. IMPLEMENTATION

A. Hardware

Raspberry Pi (ARM 11): The Raspberry Pi is a credit-card sized computer that plugs into your TV and keyboard. It is a capable for little projects, and for many of the things that your desktop PC does, like spreadsheets, word-processing and games. It also plays high-definition videos as shown in Fig.3. We want to see it being used by kids all over the world to learn how computers work, how to manipulate the electronic world around them and, how to program. The original Raspberry Pi is based on the Broadcom BCM2835 system on a chip (SoC), which includes, Video Core IV GPU, RAM of 512 MB. The system has Secure Digital (SD) socket for boot media and persistent storage. A SoC consists of the hardware, described above, and the software controlling the microcontroller, microprocessor or DSP cores, peripherals and interfaces. The design flow for Soc aims to develop this hardware and software in parallel.

Fig 3. Raspberry Pi Board.

IEEE 802.15.4: The ZigBee network is defined by the ZigBee Alliance and based on the IEEE 802.15.4 standard, which is target data RF embedded applications that require a low data rate, long battery life and secure networking. It is intended to operate in the 2.4GHz unlicensed ISM band [1-2]. There is no large numbers of data which need to convey between the wireless ordering terminal build-in ZigBee module and the center node, and because of having no high requirement of data rate, so ZigBee is well suited for wireless ordering system. Each ZigBee modules includes an IEEE 802.15.4-compliant radio, an 8051 microcontroller, programmable I/O, flexible antenna and range solutions, Transmit range is up to 300m, which can meet the demand of wireless ordering system completely. ZigBee module can be configured in star, mesh, and cluster tree network topologies. IP-Net includes support for our innovative ‘serial mesh mode’, allowing RS232/RS485 data streams to be transmitted over multiple hops to improve data reliability and increase transmission range. ZigBee Wireless network of restaurant which is configured in star topology.

In this routing topology, data traffic and network commands are routed through a central node. Peripheral nodes require direct radio contact with the central node. An ordering end device acted as a peripheral node in the network is an RFD, it have stringent requirements for low power and memory space. An IEEE 802.15.4 network requires at least one FFD usually line powered to act as a network coordinator. The coordinator sets up a network, initializes a network, manages network nodes, stores network nodes information, and transmits to control center server via RS232.

MEMS: Micro electro mechanical systems (MEMS) are small integrated devices or systems that combine electrical and mechanical components. Their size range from the sub micrometer (or sub micron) level to the millimeter level and there can be any number, from a few to millions, in a particular system. MEMS extend the fabrication techniques developed for the integrated circuit industry to add mechanical elements such as beams, gears, diaphragms, and springs to devices. Examples of MEMS device applications include inkjet-printer cartridges, accelerometers, miniature robots, micro engines, locks, inertial sensors, micro transmissions, micro mirrors, micro actuators, optical scanners, fluid pumps, transducers and chemical, pressure and flow sensors. Many new applications are emerging as the existing technology is applied to the miniaturization and integration of conventional devices.

AT89S52 (8051): 8051 is an 8-bit processor, meaning that the CPU can work on only 8 bits of data at a time. Data larger than 8 bits has to be broken into 8-bit pieces to be processed by the CPU. 8051 is available in different memory types such as UV-EPROM, Flash and NV-RAM. Microprocessors and microcontrollers are widely used in embedded systems products. Microcontroller is a programmable device. A microcontroller has a CPU in addition to a fixed amount of RAM, ROM, I/O ports and a timer embedded all on a single chip. The fixed amount of on-chip ROM, RAM and number of I/O ports in microcontrollers makes them ideal for many applications in which cost and space are critical.The Intel

8051 is Harvard architecture, single chip microcontroller (µC) which was developed by Intel in 1980 for use in embedded systems. It was popular in the 1980s and early 1990s, but today it has largely been superseded by a vast range of enhanced devices with 8051-compatible processor cores that are manufactured by more than 20 independent manufacturers including Atmel, Infineon Technologies and Maxim Integrated Products.

Temperature Sensor (LM35): The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°C at room temperature and ±3⁄4°C over a full −55 to +150°C temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a −55° to +150°C temperature range.

B. Software

The software is used to design and developed is QT creator which is used to make efficient GUI application. Qt Creator is a good example of an application that mixes different user interface technologies. In fact, it uses all of the three different approaches described below. Qt Creator uses the traditional Qt Widgets such as menus and dialogs as a basis of the user interface, Qt Quick amongst others for the welcome screen, and Qt WebKit for presenting the Qt reference documentation. Qt Creator includes a project manager that uses a cross platform project file format (.pro). A project file can contain information such as what files are included into the project, custom build steps and settings for running the applications. Qt Creator includes a code editor and integrates Qt Designer for designing and building graphical user interfaces (GUIs) from Qt widgets. The code editor can parse code in C++ and QML languages... It is possible to compose and customize the widgets or dialogs and test those using different styles and resolutions directly in the editor. Widgets and forms created with Qt Designer are integrated with programmed code, using the Qt signals and slots mechanism.

Raspbian Operating System: Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi.The initial build of over 35,000 Raspbian packages, optimized for best performance on the Raspberry Pi, was completed in June of 2012. However, Raspbian is still under active development with an emphasis on improving the stability and performance of as many Debian packages as possible. The Raspberry Pi primarily uses Linux kernel-based operating systems Raspbian (recommended) – Maintained independently of the Foundation; based on ARM hard-float (armhf)-Debian 7 'Wheezy' architecture port, that was designed for a newer ARMv7 processor (or one with Jazelle RCT/ThumbEE, VFPv3 and NEON SIMD extensions built-in) whose binaries would not work on the Rapberry Pi, but Raspbian is compiled for the ARMv6 instruction set of the Raspberry Pi making it work but run more slowly. It provides some available deb software packages, pre-compiled software bundles. A minimum size of 2 GB SD card is required, but a 4 GB SD card or above is recommended. There is a Pi Store for exchanging programs. The Raspbian Server Edition is a stripped version with other software packages bundled as compared to the usual desktop computer oriented Raspbian.

Priority Inheritance and Reversal Protocol: Priority inversion problems are eliminated by using a method called priority inheritance and reversal methods. The process priority will be increased to the maximum priority of any process which waits for any resource which has a resource lock. This is the programming methodology of priority inheritance. When one or more high priority jobs are blocked by a job, the original priority assignment is ignored and execution of critical section at the highest priority level of jobs it blocks is performed. The job returns to the original priority level soon after executing the critical section using priority reversal. This is the basic idea of priority inheritance and Reversal protocol.

IV. RESULTS

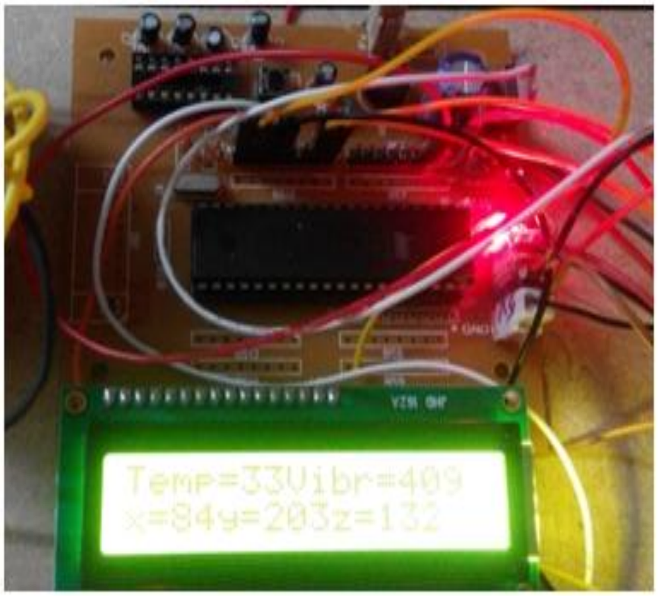
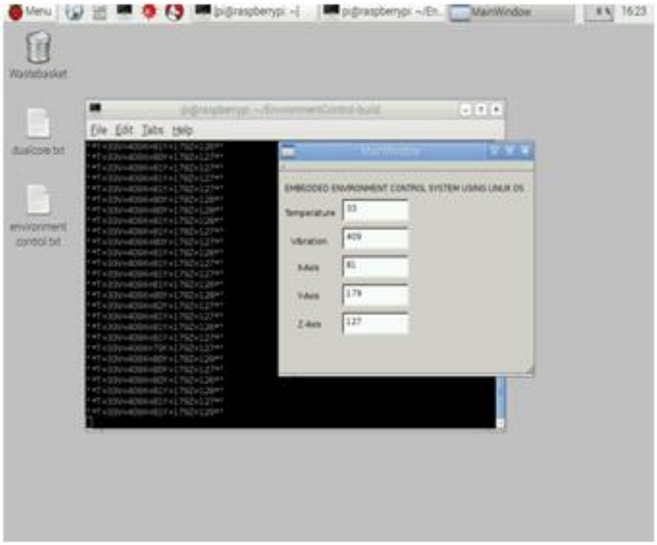


Fig.4. Sensor values on Slave.

The total system is divided into 2 parts, the first part is the considered as a slave circuit which will continuously monitor the sensor data as show in fig 4 and transmits the data over the zigbee network, which will be received by the master and displayed on the Qt GUI as shown in fig 5. These data will be continuously and appropriate priority will be allocated according to the threshold values set in the programming.



EMBEDDED ENVIRONMENT CONTROL SYSTEM USING LINUX OS

Temperature 33
Vibration 409
X-Axis 81
Y-Axis 179
Z-Axis 127

Fig.5. Received Data Display on the Master Side.

VI. CONCLUSION

In the post-PC era, the embedded system technology develops rapidly and the design of embedded GUI using Linux environment is important and indispensable components of it. This work focuses on solving the issues of poor real time kernel allocation, high cost, low precision and incapability of determining whether the sensors is in line with the in the environment management. It develops a monitoring system and GUI application based on Qt / Embedded that will periodically displays the monitored data in real time. With a perfect support of the embedded system technology, we believe that the monitoring system will have better performance and broader market prospect.

VII. REFERENCES

[1] Ni Jili, QT and window design of the Linux operating system, Beijing industry press, pp. 156-187, 2006.

[2]Sun Fang and Li Qiang, “A Study And Design Of A Wireless Serial Port Hub Based On ZigBee,” Control and Automation Publication Group,vol.24, 2008.

[3] [He Caijun and Fang Houhui, “Application of ZigBee Technology in Industrial Monitoring Network,” Computer system,” vol.19,2010.

[4] Wei Dongshan, Complete Guide to Embedded Linux Application Development, Post & telecom press, pp.170-180, 2008.

[5] Zhang Jianwu, LiJie, and Yao Yingbiao, Design and Development of Embedded System, XiDian university press, 2009.

[6] Xiguang, Li Bonian, Zhao Likai, Zhang Jie, Zhang Minghu ―An embedded real-time remote monitoring system based on B/S mode‖ 978-1-61284-722-1/11/$26.00 ©2011 IEEE.

[7]Yakun Liu Xiaodong Cheng ―Design and Implementation of Embedded Web Server Based on ARM and Linux‖ 978-1-4244-7656-511 01$26.00 ©201 0 IEEE.

Author’s Profile:

Ms V. Jyosthna,is working towards a Master of Technology in E.C.E in Digital Systems and Computer Electronics in prestigious Sreenidhi Institute of Science and Technology, Hyderabad, India. She received B.Tech in Electronics and Communication Engineering in Auroras Technological and Research Institute, Hyderabad, India

Mrs. K.Shiva Sundari presently working as an Associate Professor in ECE Department in prestigious Sreenidhi Institute of Science and Technology, Hyderabad, India.