Real Time Monitoring System based on Embedded Linux Application

[1]Unnati Patel, [2]Vijay N. Chavda

[1]ME Scholar, [2]Assistant Professor,
VLSI and Embedded system, GTU PG School, Gandhinagar, India
[1]patelunnu27@gmail.com, [2]vjychavda@gmail.com

***Abstract*—In today's world, the management of domestic laboratories in the research institute and universities has issues of poor real time, high cost and low precision. It is difficult to determine the quality of environment of the laboratory. So the Laboratory Monitoring System should be developed to implement early warning, remote control, real time monitoring and other functions. Thus we focus on the process and difficult points in the application of embedded GUI based on Qt / Embedded in the laboratory monitoring system. The design of GUI for embedded systems is different from that of traditional data computing, which often handles mouse or keyboard events to complete a specific calculation, while for embedded systems the events are caused by external devices. Because the embedded systems is resource-constrained, the design mode of the GUI of the traditional PC has relatively larger memory consumption and take up more CPU time, which is not suitable for embedded systems. The intelligent monitoring system uses Qt / Embedded under embedded Linux as its GUI development platform, which can fully satisfy the restriction of embedded system resources.**

***Index Terms*—Real Time Monitoring, Raspberry pi, Embedded/Qt**

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 that manages the following:

- The sensor networks include a variety of monitoring sensors such as temperature sensor, current sensor, and voltage sensor. These sensors automatically monitor the temperature, current and voltage in the laboratory. The microprocessor collects the data for the environment of the lab and processes it.
- The electrical equipment switches control various electrical equipment in the laboratory. When the laboratory environmental data is abnormal, such as the temperature is too high, the system will cut off the large-scale power equipment; switch on the air conditioning for cooling and when an illegal invasion occurs, the burglar alarm will ring to notify the administrators.
- The System is equipped with a LCD monitor, which enables us to develop friendly GUI with QT under embedded Linux system to provide functions such as querying and setting the laboratory environment parameters. The monitoring software real-time receives data of all monitored points, so as to track the data statistics, analyze and respond to abnormal condition. All the data received will be displayed on the monitor graphically in real time.

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 of the lab. 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 lab monitoring system uses Qt /Embedded under embedded Linux as its GUI development platform, which can fully satisfy the restriction of embedded system resources.

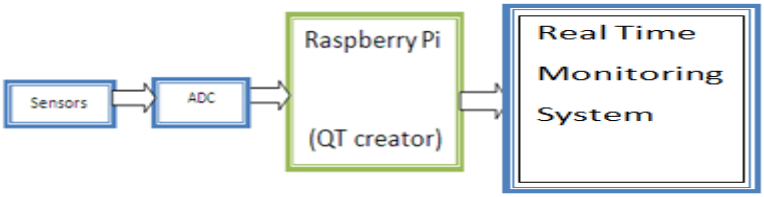

Fig1. System architecture of real time monitoring system

The various sensors are connected to the Raspberry Pi via an ADC. The ADC is connected to the Raspberry Pi at the SPI pin. The data from the sensors will be moved to the SPI pin that will be taken as an input of the system and is sent to the Qt creator Platform. At the Qt Creator application, the received input will be displayed. The main task is to create a GUI application that will display the input not only in text format. But as the data changes, it will display the change in value of the input in a graphical representation. There will be a plot created that will display the changes in data on the sensors immediately as the value on the sensors gets updating. The GUI application is such that the changes monitored on the sensors will be immediately plotted in a graphical format. That is, the graph will be updating in Real time. The changes will be made on the graph in real time as the values on the sensors are updated.

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 to achieve the graphical display of data collected by a variety of sensors.

III. IMPLEMENTATION

A. Hardware Design

The Raspberry Pi is a low cost single-board, credit card sized Linux computer. It is developed by the Raspberry Pi Foundation in the UK. It is controlled by a modified version of Debian Linux optimized for the ARM architecture.It has two models model A and model B. The Model B has 512 MB RAM, BCM2385 ARM11, 700 Mhz System on chip processor.The Raspberry Pi is connected with the ADC that will be in turn connected to the various sensors. The sensors will have an analog output which can be converted into digital with the help of ADC. The ADC is connected with the Raspberry Pi through SPI configuration. The ADC used here is ADS7841 that is 12 bit 4 channels ADC.

The current and voltage are measured using Current Transformer sensor and Voltage Transformer sensor. Using these sensors, continous monitoring of voltage and current is carried out. The result of monitoring is saved automatically in a pdf format at regular interval.

If the readings exceed the normal limit, an automatic control system is designed that will invoke the relay. The relay is connected to the device whose current is being monitored. Whenever the result exceeds a certain limit, relay will cut off the connection with the hardware device connected to it.

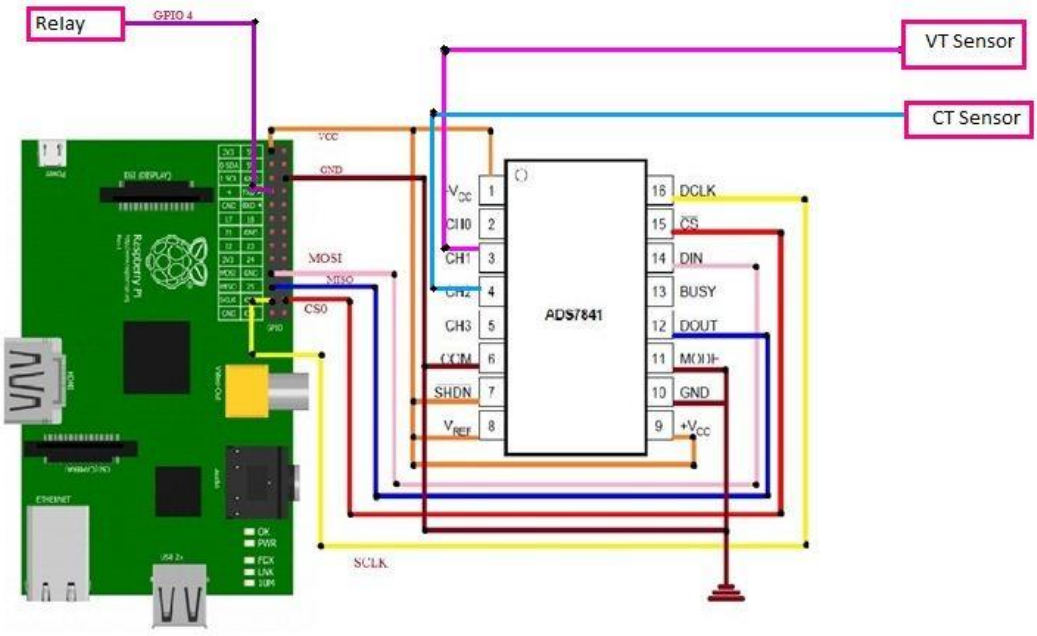

Fig 2. Hardware Configuration

B. Software Design

In this system, the user interface is designed by QT Creator. Qt/Embedded is developed by Trolletch Company in Netherlands for embedded systems. It is popular GUI that is mainly used in embedded Linux. It is easy to transplant to Linux as well as Microsoft Windows. A friendly man machine system is very important for monitoring system. Using QT Creator, the man machine interface looks much sophisticated and easy for monitoring the voltage and current of any system.

The GUI is designed to display the desired sensors results in a graphical format. This means that the monitoring is achieved in real time. As the sensors data is updated, the changes can be immediately seen on the graph in real time. The graph will update itself and add the latest results continuously. Various power related parameters are calculated and stored in a

text file. The text file consists of the readings of voltage and current and all the related parameters calculated in numerical format. The real time graph is stored in a PDF format.

The various sensors event handling can be explained by the following flowchart:

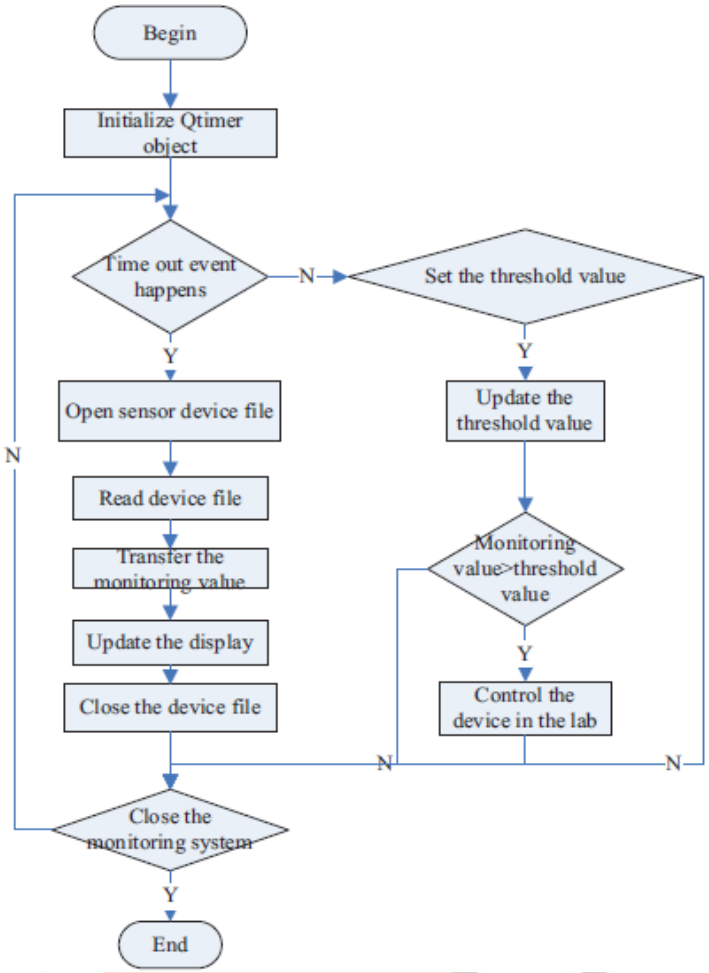

Fig 3: Event handling of various sensors

REAL TIME PLOTTING

In research labs, a lot of different computer simulations are created. The simulations usually produce a lot of data, which is normally analyzed after the simulation has finished. The analysis is a lengthy process. During the first stages, a lot of graphs which tell us various things about the data are generated. Often the most important conclusion drawn from a data plot is that we need to re-run the simulation with a different set of parameters. This can be frustrating, as sometimes the simulations have to run for many hours, or even several days. We realized that almost every time we created a new simulation, we also created a new data visualization module in order to be able to watch how the simulation variables change during the simulation's live-time. In the vast majority of cases these visualization modules are some kind of graph plotters. There are many plotting tools around, some of which offer very advanced data analysis features. However, there is no tool suitable for our requirements - being able to plug into various sensors data changes and plot their output in real-time.

Real Time Graph is a professional solution for exploratory data analysis. It is particularly useful while visually searching data at particular instant or interesting features. It allows for live monitoring of real-time data. The focus lies on a very fast and easy to use interface, rather than on powerful analysis tools.

IV. IMPLEMENTATION RESULTS.

The results obtained are shown below:

1. When the current sensor is connected with a 100W bulb. And the voltage transformer sensor is measuring the input voltage.
2. A pop –up that asks if we want to save the result obtained in PDF format. The pop-up comes up automatically after every 100 sec result is been displayed on the monitor.

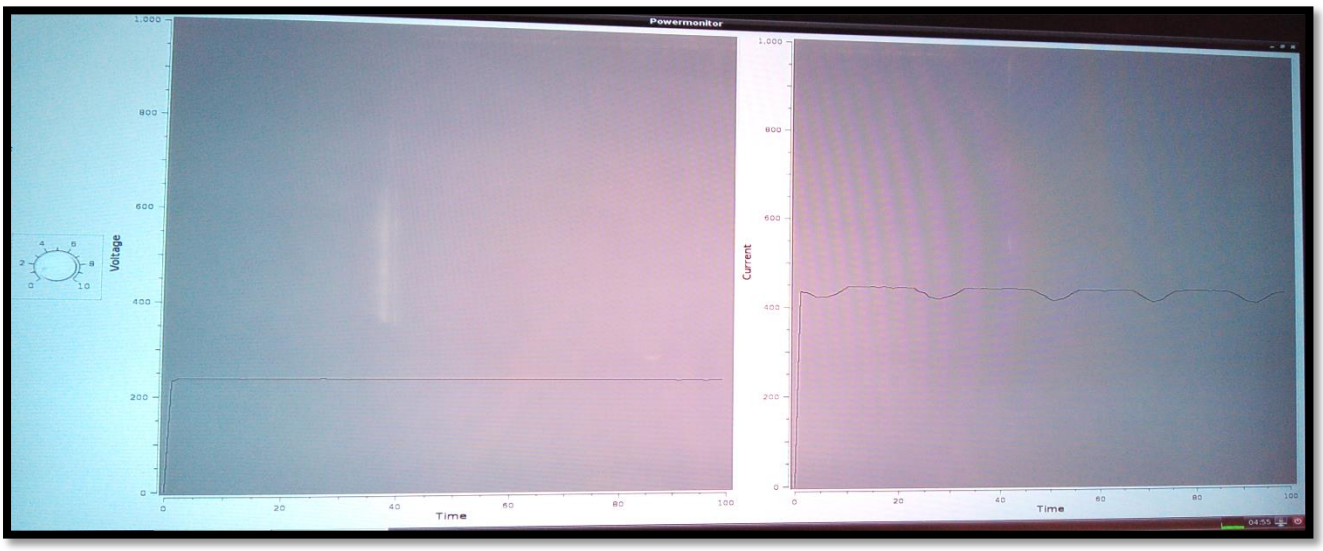

Fig 4: Implementation Result

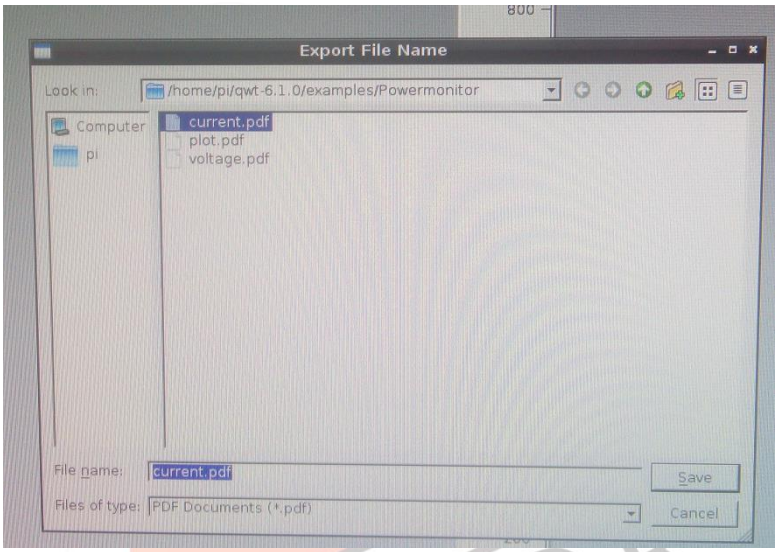

Fig 5: Export File

V. CONCLUSION

In the post-PC era, the embedded system technology develops rapidly and the design of embedded GUI is important and indispensable components of it. This work focuses on solving the issues of poor real time, high cost, low precision and incapability of determining whether the lab environment is in line with the body's health indicators in the laboratory management. It develops a laboratory monitoring system and GUI application based on Qt / Embedded that will graphically 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.

VI. ACKNOWLEDGEMENT

Apart from the efforts of myself, the success of any project depends largely on the encouragement and guidelines of my guide, Prof. Vijay N. Chavda. I take this opportunity to express my gratitude towards him for his useful comments, remarks and engagement through the learning process of this master thesis.. I can't say thank him enough for the tremendous support and help. I feel motivated and encouraged every time I attend his meeting. Without his encouragement and guidance this project would not have materialized.

REFRENCES

[1] Chen Kun, Chen Yun-qiu, Liu Xin. Application design based on Qt / Embedded and embedded Linux [J]. Computer and Digital,Engineering, 2009,37 (1): 156-161.

[2] WEI Dong-shan, "Complete Guide to Embedded Linux Application Development," Posts & Telecom Press, Beijing, 2008, 8. pp.21-23.

[3] Jasmin Blanchette and Mark SummerfieldI, C++ GUI Programming with Qt 4 (2nd Edition). New Jersey: Prentice Hall, February 2008, pp. 279-280

[4] Linux Frame Buffer Driver Developer [EB/OL], http://www.linux-fbdev.org/

[5] S.S.Sadistap, S.A.Akbar, B.A.Botre, "Design and Development of Digital PID Controller for DC Motor Drive System Using Embedded Platform for Mobile Robot." International Advance Computing Conference (IACC) 2013 IEEE 3rd international

[6] M.Shankar, B.Lalitha, Akhil C.V, Nijin Karthik. C, Praveen T.T, A. Sasi kumar " Real Time Controlling and Testing In an Embedded Processor under a Multitasking Environment" International Journal of Engineering Research and development (ijerd) Volume 6, Issue 10 (April 2013), PP. 54-59 54

[7] M. Chiandone, S. Cleva, A. Massi Pavan and G. Sulligoi "Monitoring applications of electrical and climate data for PV systems using Linux RTAI" Clean Electrical Power, 2009 international conference.

[8] Zhang Qinghui, Li Xudong "GUI Design of Grain Monitoring and Control System Based on QT" 2010 2nd International Conference on Signal Processing Systems (ICSPS) v2-20, v2 -22

[9] Fan Shengwen, Chen Licong, Wen Chunxue, "Software Design of Wind Power Supervisory Control System Based on Embedded Qt" Electrical and Control Engineering (ICECE) 2011 international conference

[10] Srinivas Raja, G. SrinivasBabu, "Design of Web based Remote Embedded Monitoring system" International Journal of Technology and Engineering system(IJTES),Jan-March 2011-Vol.2,No.2.

[11] Fang Hongping, Fang KangLing, "The Design of Remote Embedded Monitoring System based on Internet" International Conference on Measuring Technology and Mechatronics Automation,,2010.