

Intelligent Improvement in Multi-Task Data Acquisition System Employing Low Cost Embedded Web Server

N. Kumaresan,
Assistant Professor,
Department of ECE,
Anna University, Regional Centre,
Coimbatore- 641 047

J. Suganthi, PhD.
Vice Principal, Professor & Head, Department of
CSE, Hindusthan College of Engineering &
Technology, Coimbatore-641 032

ABSTRACT

Web-based user interfaces are state-of-the-art in network-based embedded systems. These Web-based user interfaces makes configuration, control and monitoring of a device from every PC, smart phone or tablet device running a Web browser possible. This paper proposed the design and development of on-line Interactive Data Acquisition and Control System (IDACS) using an ARM based embedded Web server. As a result, this provides a need for Web services being deployed on various embedded processors such as Advanced RISC Machine (ARM). Single chip IDACS improves processing capabilities and rectifies the problem of poor reliability in real time context. This system uses ARM9 Processor, which is a real time processor that handles various processes based on multi tasking and reliable scheduling mechanisms. Web server application is ported into an ARM processor using Visual Basic, Web pages are written by Visual Studio; it is beneficial for real time IDACS, Mission critical applications, ATM networks, Power plants.

Keywords

ARM, Embedded Web Server, TCP/IP Protocol, Windows CE 6.0

1. INTRODUCTION

The Web server offer services through the Web that brings availability and accessibility of services on the internet backbone. A Web server composed an operating system that leads the Web pages or the application and even memory of a special hardware. Currently, the hardware data acquisition systems are quasi real-time data transferring capability [1].

1.1 Embedded Web Servers

The general Web servers that were developed in NT servers or UNIX and LINUX workstations, requires megabytes of memory, a fast processor and a preemptive multitasking OS with its associated resources [2] [3]. An embedded Web server [4] to provide a remote access to the device from a Web browser, a Web server has to be embedded in a device. It can be utilized to serve the embedded Web documents which include Static and Dynamic information about the embedded systems.

It has certain requirements, which includes, low resource usage, high reliability, security and portability, which differ from general Web server technologies. It also has some design issues in case of HTTP and embedded API.

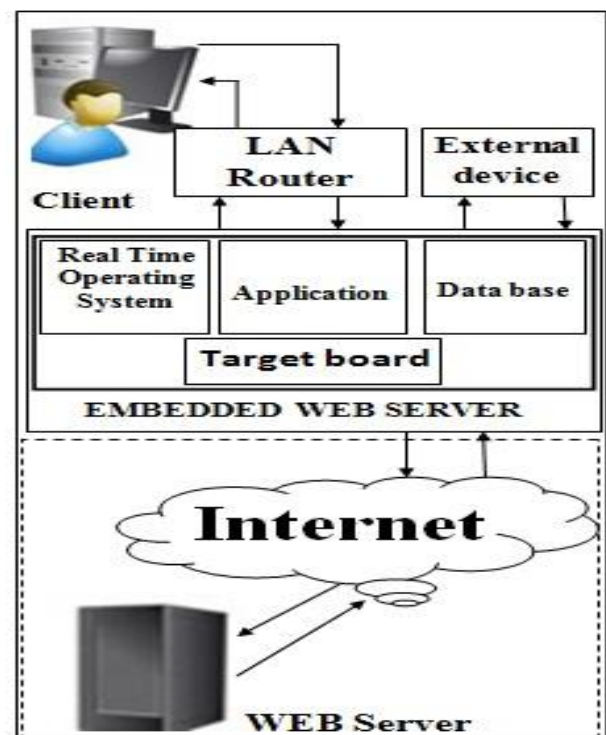


Fig 1: Embedded Web server architecture

2. HARDWARE DESIGN OF THE SYSTEMS

2.1 IDACS Design

IDACS design begins with the measurement of physical properties such as temperature, pressure, load voltage, load current. Irrespective of the type of physical properties to be measured, the first step is to transform those properties to be measured into a unified form that can be sampled by the data-acquisition system. This task can be performed by the device called sensors [5]. These sensors measure the differing properties suited to detect its associated properties.

DAQ systems have various signal conditioning mechanisms that adequately modifies the electrical signals into its equivalent voltage signals which can be digitized using an Analog-to-digital converter (ADC).

2.2 Hardware Requirement

A sensor, which is a type of transducer, is a device that converts a physical property into a corresponding electrical signal (e.g., a voltage or current) or, in many cases, into a corresponding electrical characteristics (e.g., resistance or capacitance) which is easily be converted into an equivalent electrical signal. To develop an embedded Web server, components used here are listed below:

- Mini2440 ARM9 Processor
- 10KΩ Thermistor
- MPX2010-Pressure sensor
- Voltage and Current transformer.

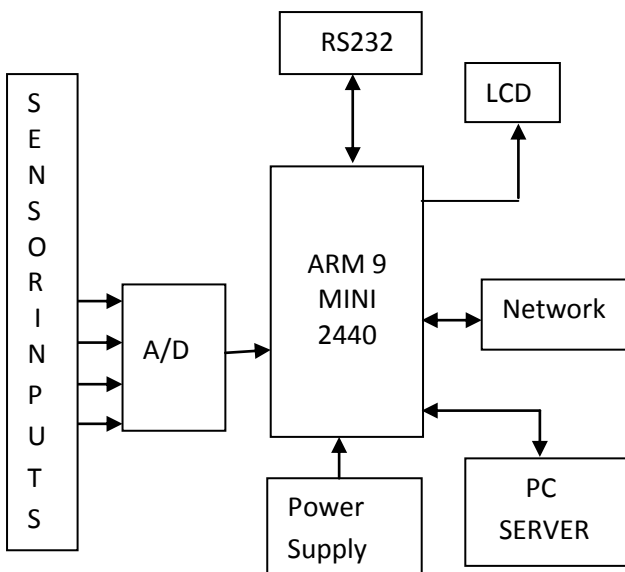


Fig 2: Block diagram of an embedded system

2.2.1 Mini 2440

The friendly ARM Mini 2440 is a single board computer based on Samsung S3C2440 ARM9 microprocessor [6]. The board measures 10cmX10cm, ideal for earning about ARM systems or integrating into the numerous products. S3C2440 supports two startup modes: one is from NAND flash startup and other from NOR flash startup. The chip select memory space allocation is differed for different startup modes.

2.2.1.1 Startup Mode Selection

To choose the development board startup mode:

S2 DIP switch governs the target board tips to choose the development board startup mode. Although,

- Switch S2 “NOR” side logo, the system will start with NOR flash.
- Switch S2 “NAND” side logo, the system will start with NAND flash.

The NOR flash and NAND flash of the development board has been burned into the same bios from the factory S2 has been receiving side of NAND flash, the system boot from start up operation of NAND flash system.

2.2.2 A/D Output

On the mini 2440, totally there are 4 A/D conversion channel that can be led out, which are located on the CON4.GPIO. The ADC chip should be synchronized with the processor’s clock while communicating with ARM processor.

2.2.3 Thermistor

A thermistor is a type of negative temperature co-efficient resistor, whose resistance decreases with increasing temperature. Its application’s areas are, used as inrush current limiters, temperature sensors, self-resetting, over current protectors and self-regulating heating elements. It differs from various temperature detectors in the form of materials used and its range of temperature sensing etc. In particular thermistor typically achieves a higher precision within a limited temperature range, typically –90°C to 130°C.

2.2.4 MPX2010-Pressure Sensor

The MPX2010 family of pressure sensors appeals to measure small gauge, vacuum or applications presents design-in challenges for these sensors. For very low pressure sensing, large amplification is required. Two pairs of operational amplifiers were packed in single 14 pin quad package [7]. The first gain stage is accomplished by feeding both pressure sensor outputs (Vs- & Vs+) into the non-inverting inputs of the operational amplifiers.

2.2.5 Voltage and Current Transformer

Voltage and current transformers are also called as instrument transformers. These transformers are necessary to protect the devices in the circuit from heavy current flow. It is also necessary for isolating, protecting, control and measuring equipments from the high voltage and current of a power system and for supplying the equipment with the appropriate values of current and voltage-generally this is 1A or 5A for the current coils, and 120V for the voltage coils.

2.3 Software design of the System

The software system designed for the distributed I/O Windows CE based embedded Web server data acquisition system is an open source software which includes the functions such as task management, system management, timer management, information management, queue management and so on. The above mentioned management functions can be serviced as though GUI core functions.

2.3.1 Software Requirement

- Visual Basic 6.0
- Visual Studio 2010

2.3.2 Embedded Web server Process Structure

The architectural diagram in the Fig.3 shows the way how to communicate between a Web server layer and the other layers, which uses different communication methods and protocols that are connected with the embedded Web server by bus structure and it adopts master and slave access mode during communication between them. The embedded Web server is master machine, as well as local equipments are slave machines. Each of equipment has its unique address. Its status flag is set as WAIT once reset.

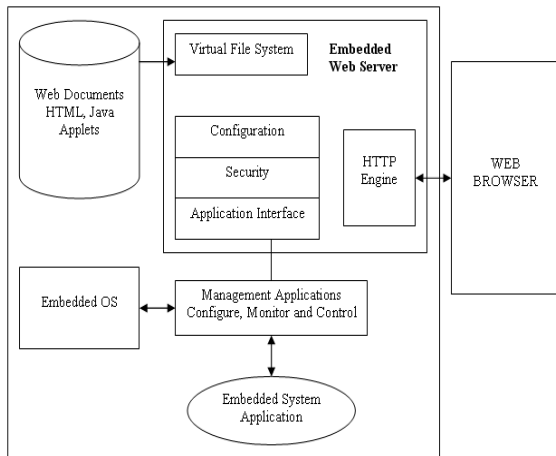


Fig 3: Overall System Design

The communication process described here uses an embedded Web server which transmits the access address via serial protocol. Each of the equipment receives the access address and establishes the connection with embedded Web server. In the communication process, the equipment's status is set as UNWAIT. The status will be reset as WAIT since the communication process gets over.

The Monitoring Server Module exchanges the information with SQL which associates in to traditional Web through TCP/IP protocol. In order to simultaneously support the two data dynamic interactive mechanism, the Monitoring Server Module respectively listens to control and monitoring requests via controlling port and monitoring port at the same time.

Receiving the control request on controlling port, Monitoring Server Module may create a new thread to response. Consequently the new thread analyzes request string to get equipment number and command, and then invokes Equipment Controlling Module to execute.

Receiving monitoring request on monitoring port, Monitoring Server Module may establish the special socket for communication and store it into correlative queue according to status type. After that, Monitoring Server Module is always in wait status until receiving a new status arriving message from Status Acquisition Module. Once receiving the message, Monitoring Server Module traverse the corresponding queue according to status type and sends information to the relevant client.

The Embedded Web server processes an HTTP request in discrete steps as though a finite state machine (FSM). Fig.4 shows the embedded Web server process structure. To enhance multiple connections in a single threaded environment, multiple finite state machines which uses a light weight task structure that are run by a scheduling system. The above task structure consists of the following heuristics,

- A pointer which holds the function being run by the system.
- A variable holds the state of the function
- A flag that indicates the states being run or blocked by the finite state machine (FSM).

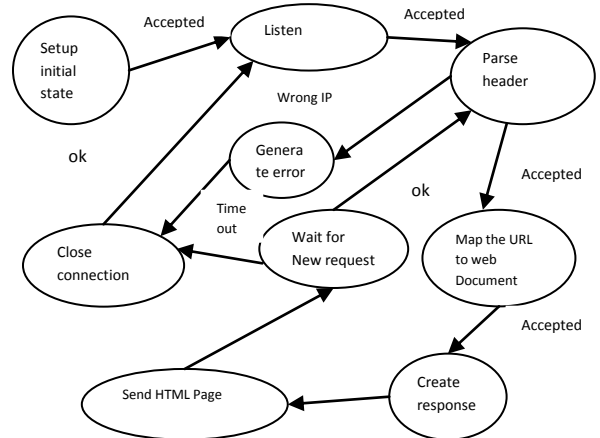


Fig 4: Embedded Web server Process Structure

The scheduling system allocates an available finite state machine for an accepted connection, checks each finite state machine to see if it is blocked or runnable and move the finite state machine.

3. EXPERIMENTAL SETUP

3.1 Experimental Setup of the Embedded Web server

The experimental setup of the embedded Web server which consists of the target board mini2440 is connected to the sensor inputs via A/D Converter.

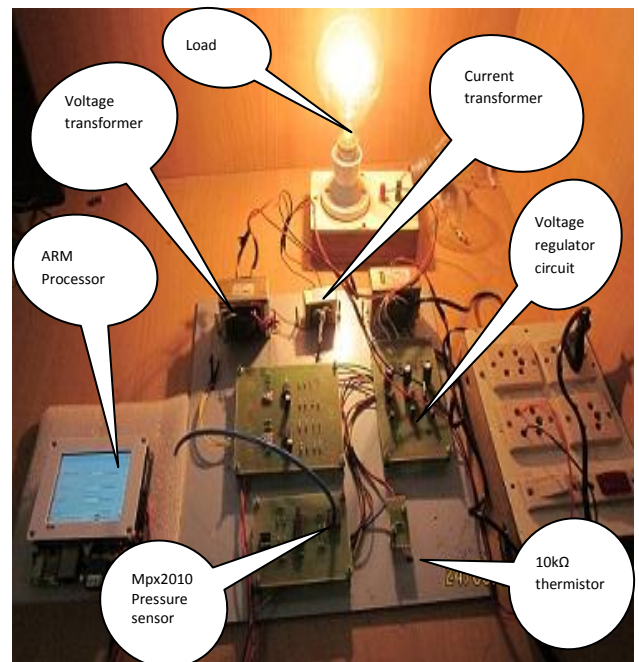


Fig 5: Experimental Setup

The server PC connected with ARM mini2440 which updates the sensor information and stores in a buffer, the sensor information includes temperature, pressure values, load voltage and load current. Here, the small load of 100W bulb is connected with the experimental setup.

3.2 Experimental Setup of the Embedded Web server

The Windows CE 6.0 and they are ready and embedded Web server application are to be ported on the target.

- Run DNW for WinCE
- Check the settings (Configuration → Options)Baud Rate: 115200, Download Address: 0x30000000
- Open the COM-Port
- Set switch (S2) the boot mode switch to NOR while the board is off.
- On board power.
- The DNW which appears on the boot loader menu shows the USB connection [USB:OK]
- Press f to format the NAND flash.
- Press v to download supervivi via USB.
- Now the boot loader is waiting for data.
- Press n to download Nboot for WinCE via USB
- Now the boot loader is waiting for data.
- Press l to download WinCE boot-logo via USB
- Now the boot loader is waiting for data.
Select USB Port → Transmit and choose the file to send to the board: bootlogo.bmp
- Press w to download WinCE NK.bin via
Now the boot loader is waiting for data.
- Select USB Port → Transmit and choose the file to send to the board: nk_A70.bin, nk_T35.bin or nk_N35.bin
- Power off the board and set the boot mode switch (S2) to NAND.
- Power ON the board to start WinCE.

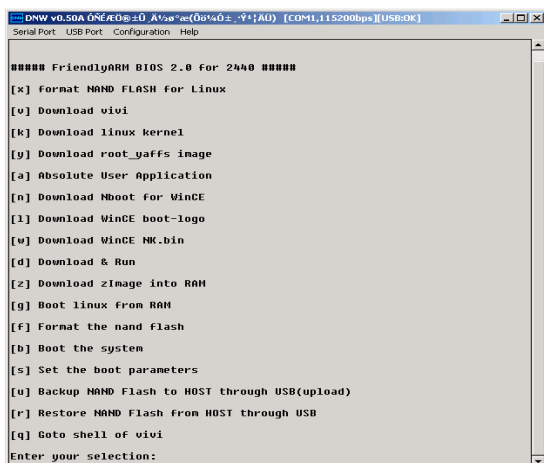


Fig 6: System Boot Configuration

Now, the code is loaded on the target, and the target is configured over the network, to work as an embedded Web server.

3.3 Testing the Embedded Web server

Initially, the target system is tested for the working of an operating system that can be done by booting the target system using a hyper terminal. Henceforth the embedded Web server responds to the clients request by typing the IP address of the server in the client’s browser. Then the controller sends the request to the router which processes and checks for the system connected to the network with the particular IP address. If the IP address entered is correct and matches to that of the server, a request is sent to the controller of the server and a session is established, using a TCP/IP connection and the server starts sending the Web pages to the client.

4. RESULTS AND DISCUSSION

Fig.7. shows execution results of ARM embedded Web server based on IDACS system. Server sends a response by providing user login information, when a client enters correct user id and password. Server provides sensor information’s and can monitor from anywhere using TCP/IP Protocol.

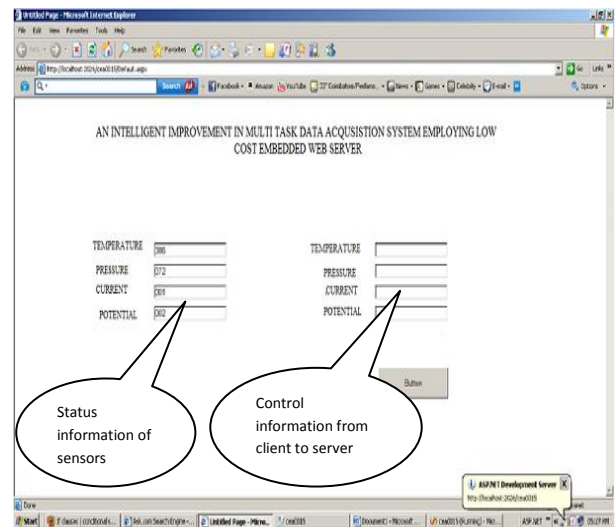


Fig 7: Output of four channels input signal and status

Table 1. shows implementation of Web server application in two different processor and parameters comparison. Compared to AT89C51 and LPC2132, mini2440 number of sensors and sampling rate is more and consumes less power.

Table 1. Parametric Comparison of different processor implemented on Web server application

Parameter Processor	No. of Bits	No. of Sensors	Protocol	Sampling Rate	Txn. Rate	Throughput Rate	Power Consumption
AT89C51	8 bits	2	Zigbee	10KSPS	250 Kbps	-	High
LPC2132	16/32 bits	1	TCP/IP	400KSPS	-	400Kbps	Low
MINI2440	16/32 bits	4	TCP/IP	500KSPS	-	1.90Mbps	Low

5. CONCLUSION

This paper proposes the architecture of the embedded remote monitoring system based on the Internet, because the system adopts embedded Web server as a central monitoring node,

the system is provided with not only, excellent cost performance but also running steadily and reliably. An embedded device with a standard Web browser can be accessed, controlled and managed by the remote users over the traditional internet. Moreover, utilization of dynamic monitoring Web based on SQL server improves the response capability and brings convenience for complex monitoring Web design. The system has achieved the design of data collector successfully by debugging of the hardware and software.

Future work includes implementing the data acquisition system in Android OS, so that one can monitor and control an application with the hand held device such as Android Mobile phones, tablet PCs and iPhones.

6. REFERENCES

- [1] Li, S., Jiarong R lu., Yichun C. Wu., Guiming M. li., Feng Wang and Yong Wang, "Continuous and Real – Time Data Acquisition Embedded System for EAST", Vol. 57, No. 2, April 2010.
- [2] Jacker, K. and Mckinney, J, "TkDAS – A data acquisition system using RTLinux, COMEDI, and Tcl/Tk", in Proc. Third Real – Time Linux workshop 2001. [Online]. Available: The Real –Time Linux Foundations: 2001/papers.html.
- [3] Siever, A., Weber, S., Figgins and Love, R. "Linux in a Nutshell". Sebastopol, CA: O' Reilly, 2005.
- [4] Tao Lin, Hai Zhao, Jiyong Wang, Ganglia Han and Jindong Wang, "An Embedded Web server for Equipment", School of Information Science & Engineering, Northeastern University, Shenyang, Liaoning, China.
- [5] Zhou, Q., Wu, W., and Ma, Y. "The Embedded Data Acquisition System for Mossbauer Spectrum", in Proc. Third Real – Time Linux Workshop, Embedded Linux Expo Conf. Real – Time Embedded Comput. Conf., Milan, Italy, Nov. 2001, pp26-29.
- [6] The ARM Processor website. [Online]. Available: <http://www.friendlyarm.com>
- [7] MPX 2010 Datasheet, 2007. [Online]. Available: http://www.atmicroprog.com/download/metrologie/MPX_2010.pdf