Embedded Web Server based on ARM Cortex for DAC System

Smita Salgaonkar, Mrs. J. D. Bhosale, Mrs. Padmaja Bangde

Abstract— The scope of embedded devices is increasing day by day and the demand will be further more when networking technology is incorporated into these devices. Network embedded system have become quite important; especially for monitoring and control of the industrial devices or home appliances. HTTP protocol is able to provide communication between larger distant client servers. In this paper a hard-wired web server is proposed using ARM cortex M3 micro controller. This microcontroller is 32 bit and supporting 10 X 100 Mbps Ethernet communications. Since ARM processor has fast execution capability and Ethernet standard can provide internet access with reasonable speed, this system is suitable for enhancing security in industrial conditions by remotely monitoring various industrial appliances where high safety and care is a necessity. In this paper we design a system using arm cortex micro controller to read data on HTTP protocol continuously and this design can be used for monitoring and controlling various devices which supports internet connectivity. This architecture is suitable for distributed measurement/control system and provides scalable networking solutions which can be optimized for educational laboratories, instrumentation and even in medical field.

Index Terms— ARM cortex Microcontroller, Ethernet interface, Data Acquisition, LPC 1768, Network Communication.

---

1. Introduction

THIS web server is a computer that stores data on the internet and delivers web pages to viewers upon request. This service is referred to as web hosting. Every web server has a unique address called as an internet protocol address that tells other computers connected to the internet where to find the server in the vast network. Web host rents out space on their web servers for people or businesses to set up their own website and web server allocates a unique website address to each website it hosts.

When someone connects to the internet, his personal computer also receives a unique IP address assigned by his internet service provider (ISP). This address identifies the computer location on the network. When he clicks on the link to visit a website his browser sends out a request to that IP address. This request includes return information and functions like a postal letter sent across town; in this case, the information is transferred across a network. The communication passes through several computers on the way to that website, each routing it closer to its ultimate destination.

As the technology of electronic measure and network communication has made great progress, DCS based on embedded multiprocessor has been used widely in industrial control. Whereas, embedded equipment for acquisition and control have wide variety of kinds, while interface protocols used in equipment become more sophisticated. Therefore it is highly necessary that a uniform interface be provided between every automation system vendors [3].

A person that needs to access any data must first access the server. An indirect access to the data-acquisition unit makes the system unattractive for real-time control applications, where direct interaction with the system may be required. The need to maintain an additional server will also increase the setup costs and the costs to maintain the acquisition systems, such as regular maintenance costs, system updates, etc. Therefore, the central server has to be eliminated for a real time system. The closet to this idea is published in [1].

The proposed system uniquely reduces the costs occurring from frequently requested data and eliminates the need for a well-established server.

The system uses a dummy server for static information, thus optimizing the transfer of large data. The user can directly log in and interact with the embedded device in real time without the need to maintain an additional server [1].

The system is modularly built, allowing different modules to be added. In addition, it is flexible to accommodate a wide range of measurement devices with appropriate interfaces.

An embedded web server is a web server which runs on an embedded system with limited computing resources to serve embedded web documents to a web browser. By embedding a web server into a network device, it is possible to provide a web based management user interface, which are user friendly, inexpensive, cross-platform and network ready [2].

- *Smita Salgaonkar is currently pursuing masters degree program in electronics engineering in Mumbai University, Countr. E-mail: smitassalgaonkar@gmail.com*
- *Mrs. J. D. Bhosale has done masters in electronics and telecommunication engineering. E-mail: jbhosale@mes.ac.in*
- *Mrs. Padmaja Bangade has done masters in instrumentation engineering. E-mail: pbangde@mes.ac.in*

2. Interactive Data Acquisition System

Interactive Internet-based systems provide a way to monitor and adjust using standard web browsers and a PC. The target systems can be monitored and controlled independent from the location and the platform since standard web browsers can be used on the client side. A typical data-acquisition system is connected to web clients via the Internet.

The data acquisition system needs to relay the acquired information to the requesting clients. Digitally acquired data are stored in web server's data base. Whenever the client wants to access data, it sends the request to server; this request is taken by the router, which is connected to the internet. The web processes the request made and finally connects to the desired web server, access the requested data and sends the data to the client.

2.1 Embedded Web Server Architecture

The system consists of three different parts. First is the Embedded Web Server. Second is the user part or client and third one is the sensors and appliances to be monitored. The client thus monitors various parameter status through sensors and also controls many industrial appliances with the help of the web server.

An embedded web server is an ARM processor that contains an internet software suite as well as application code for monitoring and controlling machines/systems. Embedded web servers are integral part of an embedded network. Fig 1. Shows the proposed concept of DACS with embedded web server.

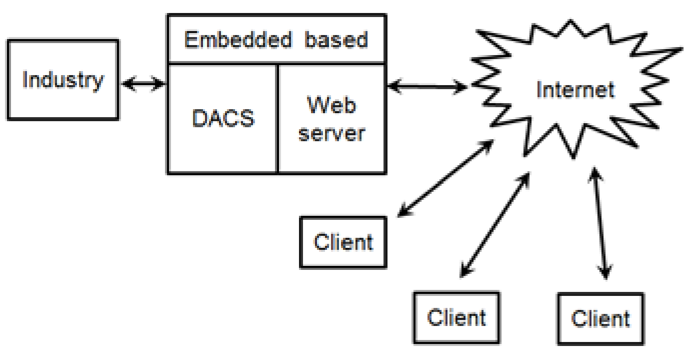

Fig 1. Embedded Web Server Architecture

ARM processor is the responsible part for measuring signals and controlling the devices remotely. Measurements can be done by DACS mode and the data are shared with clients through embedded web server by embedded web server mode. The real time operating system manages all the tasks such as measuring signals, conversion of signals, data base updation, sending HTML pages and connecting/communicating with new users etc.

Web based management user interfaces using embedded web server have many advantages: ubiquity, user-friendly, low-development cost and high maintainability. Embedded web server has different requirements, such as low resource usage, high reliability, security, portability and controllability for which general web server technologies are unsuitable.

2.2 Establishing a Direct Communication Link Between the Client and the Embedded Device

GSM and GPRS are developed for cellular mobile communication. Hence it is very much easy to be getting connected to the Internet world. Once a GPRS connection has been established queried data can be relayed to the client via a central server. Using a central server to relay the acquired data has some disadvantages. First, a central server needs a client interface framework.

An additional data transfer corresponds to time delays before the data are made available to the client. In addition, since the server acts as a relay, no direct bidirectional communication between the client and the embedded system can be established. This makes the system unsuitable for real-time control applications. The basic idea behind real-time processing is that the embedded system is expected to respond to the queries in time. Real time should be fast enough in the context in which the system is operating and reliable as well.

Real-time system correctness depends not only on the correctness of the logical result of the computation but also on the result delivery time. This method also increases the data transfer cost as the number of clients increases due to the access amount of data transfers via GPRS. Direct communication, on the other hand, enables access to only relevant information in the embedded system by preprocessing the data.

The embedded system should also handle the web services. This eliminates the need for a central server and reduces the amount of data sent from the remote unit since only the queried data will be transferred.

In the proposed system, the GPRS architecture and protocols are compliant with. This system is configured to be virtually online at all times in a GSM network. An admin script is executed after the boot of the operating system, initiating the GPRS connection software module. A PPP connection is established by a GPRS modem that works at 900/1800/1900 MHz operating frequencies. A PPP daemon (PPPD) is used to manage the PPP network connections between the client and the embedded module.

The PPPD is responsible for setting up the GPRS parameters, such as the connection speed and compression. To directly access an embedded system, the IP address of the embedded device should be made available to the client side. There are two choices available. A static (hard-coded) IP could be used, or the remote device should initiate a connection by reporting its IP. This choice is quite straightforward and simple. Although the usage cost remains unchanged, it requires a static IP setup by the service provider and involves monthly recurring costs.

The static IP is preferred for its simplicity in designing a system; however, its overhead may be impractical. The other choice is to use a dynamic IP assigned through a Dynamic Host Configuration Protocol (DHCP) server of the GSM provider for every connection established. However, this IP needs

ISSN 2229-5518

to be known by any client requesting an access to the embedded server. One solution is to broadcast this IP to a dummy FTP server.

The FTP server is a dummy server and does not require regular software updates or maintenance. A script on the embedded device is configured to update its IP address on the FTP server in Hypertext Meta-Language as an index.htm file, under a folder uniquely named by its hostname. This script simply parses the current IP for that embedded device and sends an html file with the IP information of the embedded device to the FTP server.

Once this file is in place, a direct connection can be established with the desired embedded device by a simple query. For example if an embedded system, named DAS, is queried from the FTP server by a simple command.

The web browser processes the (index.htm) file in the specified folder as default; therefore, a file name is not needed for referencing.

The DHCP approach is more flexible and works better compared with the static approach as a cost-effective solution, despite the necessity for a script running on the embedded server, one-time broadcasting its IP to the FTP server. With this mechanism in place, the embedded system updates it IP information on the FTP server upon every reboot, which causes an IP refresh from the GSM service- provider.

2.3 Data Management in the System

The Internet server is used to decrease the management costs by sending all the pictures (logo, picture, bar graphics, etc.) to the client through a server on the Internet.

Text data such as coordinates, temperature, and altitude are served from the embedded system. If bulky data are going to be sent, the embedded module is set to send the image only once via GPRS and placed on an FTP server.

This approach eliminates the transfer of large data through GPRS more than once, thus reducing the transfer costs, particularly if more than one client is involved or multiple requests to the same data are needed.

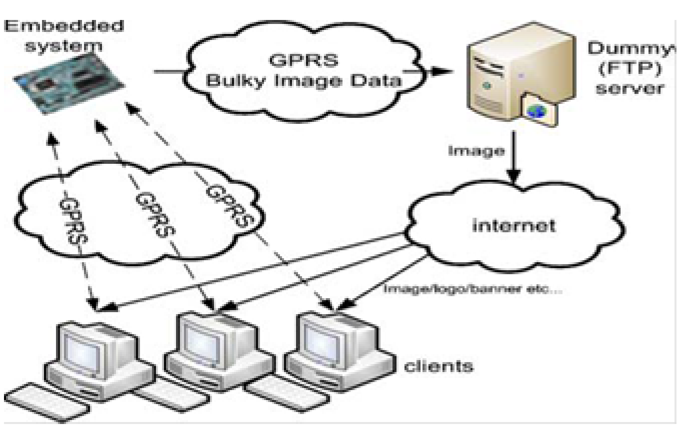

Fig 2. Data management in the proposed system

3. Hardware

The general hardware structure of the remote I/O data acquisition and control system based on ARM processor is shown in Fig 7. The remote I/O data acquisition and control system based on embedded ARM platform has high universality, each acquisition and control device equipped with 24-way acquisition/control channels and isolated from each other. Each I/O channel can select a variety of electrical and non-electrical signals like current, voltage, resistance etc., Digital acquisition are done by special ADC. The measured data are stored in external memory in which the memory is act as a data base during web server mode. The ARM processor directly supports the Ethernet service and RS485 communication. Hence the data has been stored and controlled by some other PCs or network via RS485 and Ethernet. ARM processor has internal I2C module. So it has the ability to communicate with any other peripherals. A Data acquisition module and a GPRS are integrated into the embedded board to form a sample application, as shown in fig 3.

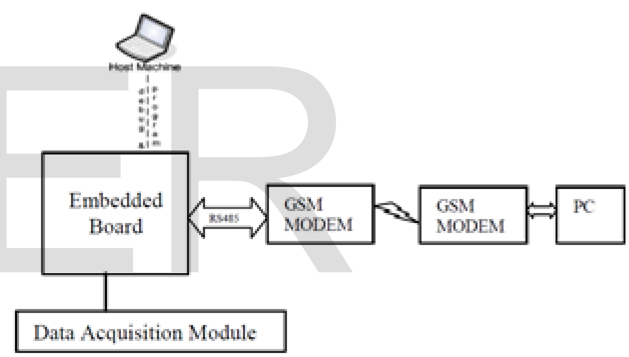

Fig 3. Block diagram of the embedded system with sample devices attached

3.1 LPC1768 Processor

This is an ARM Cortex M3 based microcontroller for embedded applications with a high level of integration and low power consumption. CPU operating frequency is 100MHz with 3-stage pipeline and internal prefetch unit. It follows Harvard architecture. It has a 512kB of flash memory, 64kB of data memory, Ethernet MAC, USB data interface, 8 channel general purpose DMA controller, 4 UART's, 2 CAN channels, 2 SSP controllers, SPI interface, 3 I2C bus interfaces, 8 channel 12-bit ADC, 10-bit DAC, Ultra low power Real Time Clock, 6-output general purpose PWM, four general purpose timers and up to 70 general purpose I/O pins. Moreover this controller has In- System Programming and In-Application Programming functionality through in-chip boot loader software. RMII interface is provided to Ethernet MAC. Fig 3 shows the system structure of the microcontroller.

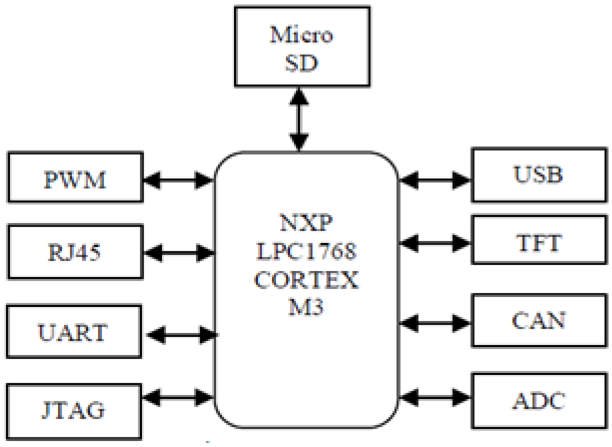

Fig 4. System Structure

The ADC is of successive approximation type with a 12-bit conversion rate of 200 kHz. Individual result registers is there for each ADC channel to avoid interrupt overhead. Moreover burst conversion mode is adopted for single or multilevel inputs [4].

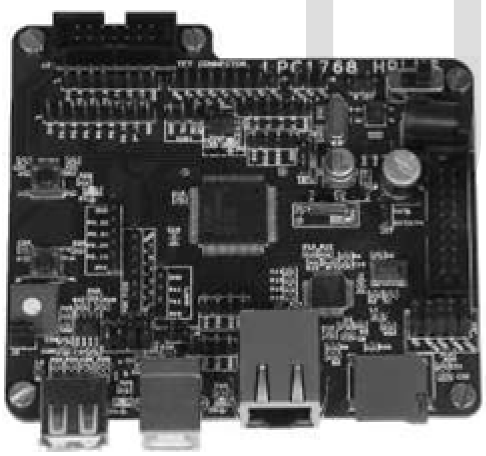

Fig 5. Embedded Web Server

Devices are connected through fast general purpose parallel I/O pins. GPIO registers are accessed through AHB multilayer bus so that the fastest possible I/O timing can be achieved.

4. System Software

The software coding for the hardware functionality is written in embedded C language in Keil Software. The TCP/IP layered model of IEEE 802.3 standard is successfully implemented in embedded C language.

4.1 Ethernet Standard

A typical Ethernet frame format is shown in Table 1. It consists of 7 bytes of preamble and this is used for bit synchronization. Start frame delimiter is of size 1 byte which is a frame flag that indicates the start of the frame. Source and destination addresses are of same size that is 2 to 6 bytes which is the 48-bit MAC address. Data size of 1500 bytes is allowed and frame checksum for checking errors is of 4 byte size that cyclic redundancy check.

The Ethernet block of the system 10Mb/s or 100Mb/s Ethernet MAC that provides optimal performance through the use of DMA hardware acceleration. It has automatic frame transmission or reception with half or full duplex operation. The Ethernet block interfaces between the off-chip Ethernet PHY using Reduced MII protocol and the on-chip Media Independent Interface management serial bus.

Filtering of received frame, multicast and broadcast support for both frame transmit and receive, power management by clock switching, automatic collision back-off and frame retransmission are some of the features of the Ethernet controller used for network communication in this system.

4.2 Client Server Communication

As we have seen before, when the client wants to access the equipments connected with the server remotely, it enters the configured IP address and it can monitor the equipment through the displayed HTML web page. For the communication to take place, before reception of data, initially CRC is performed with the checksum bytes. Once it is identified that the checksum is valid, now it checks whether the user is the authorized one or not by comparing the logged in password with the stored password in the program. If the client is authorized one, now the system allows collecting data say sensor data through the specified ADC channel. Client server synchronization is performed at specific time intervals to get the most recent information. The program flow chart is shown in Fig 6.

4.3 HTTP Protocol

The protocol used for the communication between web server and web browser is Hyper Text Transfer Protocol or HTTP protocol. This protocol defines all the basic frame work of web communications by handling requests and also by providing control information to be transferred between browser and server. To obtain a web document, the browser and server should establish a connection at Port 80 [2].

5. Results

Fig 7 and 8 shows execution results of ARM embedded web server based on DACS system. The web pages shown are requested by the client and served by the embedded web server which is ported on ARM processor. Client can interact with the machine through its own browser via these embedded web pages. Every client's control has been executed in industry via the embedded web server.

ISSN 2229-5518

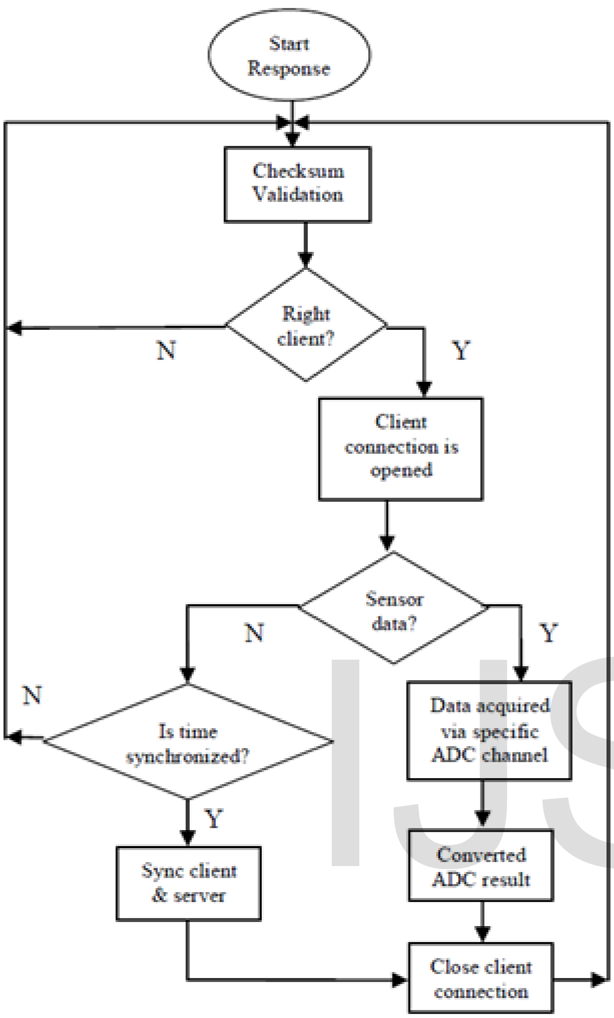

Fig. 6. Client Server Communication Flowchart

Fig. 7. Online processing web page

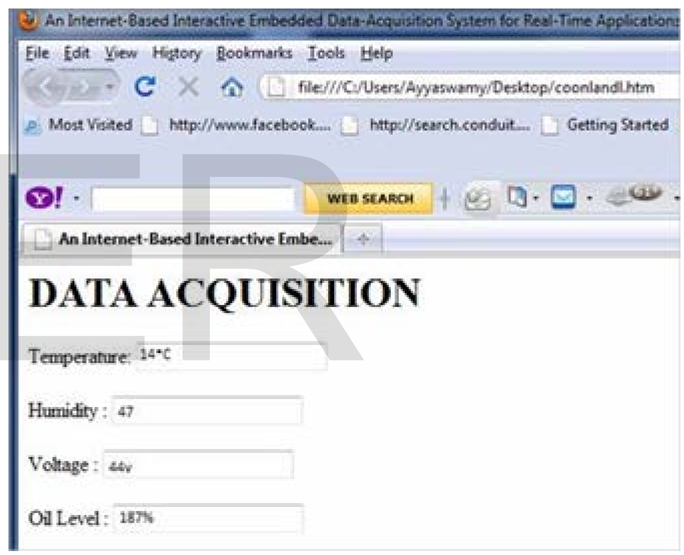

Fig.8. Client requested web page (issued by ARM web server)

6. Conclusion

In this application, a low-cost, Internet-based data acquisition and control system has been designed and implemented that should find interest from researchers. The application possibilities are virtually unlimited by attaching modules with appropriate interfaces, although the usage of the system is demonstrated with only a few sample devices. Compared with other applications, this system has advantages in terms of allowing direct bidirectional communication and reducing overhead, which can be vitally important for some real-time applications. The operational costs have been reduced by relinquishing the storage of large data to an FTP server on the Internet. The system is designed to support both static and dynamic IPs. A method to distribute the IP information has been developed. This cost-minimization effort is a big concern for mobile systems using wireless communication methods and has not been discussed before. The overall cost advantage of the system in terms of the components used makes it an attractive choice for data-acquisition applications. The power demand of the device is still in the process of being improved by putting the attached devices into sleep mode at times when they are not in use to conserve power.

ACKNOWLEDGMENT

I express my deep and sincere sense of gratitude to my guide Mrs. Bhosale J. D., Asst. Prof. Electronics Dept. and my co-guide Mrs. Padmaja Bangde, Asst. Prof. Electronics Dept. Their enthusiasm and optimism made this experience both rewarding and enjoyable. The feedback and editorial comments were also invaluable for the writing of this paper.

REFERENCES

[1] M Poongothai; "ARM Embedded Web Server Based on DAC System" 987-1-61284-764-1/11 2011 IEEE.

[2] Alen Rajan and Aby K. Thomas; " ARN Based Embedded Web Server For Industrial Applications", ISBN 978-1-4675-2248-9 2012 published by Coimbatore Institute of Information Technology.

[3] N. Kumaresan and J. Suganthi; "Intelligent Improvement in Multi-Task Data Acquisition System Employing Low Cost Embedded Web Server", International Journal of Computer Applications (0975-8887) Volume 58 – N0. 8, November 2012

[4] Rui Yang, Hong Cai and Ming zhang; "Research and Implement of Ethernet Interface Based on Embedded System," Second International Symposium on Computational Intelligence and Design, pp.288-291, Dec2009

[5] Manivannan, M and Kumaresan, N; "Design of On-line Interactive Data Acquisition and Control System for Embedded Real Time Applications," International Conference on Emerging trends in Computer and Information Technology, pp.551-556, March 2011.

[6] Datasheet Reference Manual of LPC1768 processor.