Embedded Web Server

Revathi A. S.
Assistant Professor
Department of EXTC,DJSCE
Mumbai University, Mumbai, India

Harshit Desai
Student
Department of EXTC,DJSCE
Mumbai University, Mumbai, India.

Ishan Kathuria
Student
Department of EXTC,DJSCE
Mumbai University, Mumbai, India.

ABSTRACT
ARM processor based servers are becoming trend of today's market instead of pc based servers. Ethernet module along with ARM processor form the Embedded Web Server, that provides a flexible management function and remote device monitoring based on Internet browser. Without the use of a computer, Ethernet module can communicate to the owner of the overall system, who is able to manage appliances from any location outside. This server provides a powerful networking solution and enables web access for automation and monitoring of different systems. This paper focuses on development of low cost system for automation, instrumentation and household devices control and the understanding of TCP/IP suite and user development platform for this embedded web server. Different sensors installed at working place help in sensing real time environmental conditions like temperature, light, humidity etc.

General Terms
TCP/IP, IP, Ethernet, ICMP, ARM Microcontrollers.

Keywords
ARM, Embedded, Web server, Ethernet, TCP/IP, Server, FGPA, Internet

1. INTRODUCTION
A general purpose web server is composed of an operating system, fast processor, large amount of memory, special purpose hardware, running applications and few web pages [5] etc. These web servers are developed using general purpose computers and use different kind of operating systems such as Unix, Linux Windows etc. A typical client-server architecture is highlighted in such systems where, the client accesses the server through the LAN router and the Internet. Client sends a request to the server which is then processed by the router to connect to the Internet. The web administers the request made and finally connects to the desired web server from where the requested data is sent to the client. An embedded web server is a microcontroller which includes the software and application code to monitor and control the systems. The ARM processor is an integral part of an embedded network since it creates an easy way for controlled activities of any device from any remote location. Such servers are designed using very low resource usage and are highly reliable, portable and secure systems.[4] For home and industrial automation, appliance such as lighting fixtures, air conditioners audio-visual equipment etc are fitted with servers which are connectable to the Internet making interactive communication with the device more convenient.[2] In recent years the exponential growth of Internet use has generated a strong trend toward using Internet protocols. In the paper Design an Embedded Web Server for Monitoring and Controlling Systems or Devices, G.Sunil Kumar, T.Swapna implemented the server on Samsung S3C2440A board for controlling various devices powered by embedded Linux. They concluded that the system can be connected to internet for monitoring and controlling devices and can be applicable in educational institutes, industries due its low cost and portability[6]. Janne Riihijarvi, Petri Mahonen, Mika J. Saaranen, Jussi Roivainen and Juha-Pekka Soininen, in their paper on Providing network connectivity for small appliances: a functionally minimized embedded Web server, took the WebChip approach for the design of embedded servers, which is a IPv6-compatible solution for the realization of embedded and minimized Web systems . The implementation is tested with an FPGA and can be later embedded into various ASIC chips. The code is written using C code library in VHDL. The WebChip proved to be much smaller and faster than what they expected. This was the result of minimizing functionality and employing parallel execution. Despite this minimization, the implementation worked well with standard browsers while the limitation of a single connection at a time did not really pose a problem for local networks.[3]

2. PROCESS DESCRIPTION
2.1 System Description and Working
Integration in digital equipment has its roots in the reference design of the embedded web server. Inputs from the sensor are connected through signal conditioning circuits to ARM Controller. It converts these analog signals to digital and then the parameter values are stored in the memory. Status of the working appliances is decided on the basis of these values stored in the memory. Relays can be made ON/OFF to change the status of appliances as per our requirement. By comparing this value with the standard values of these parameters, further status of devices is decided. This is how automation works. For controlling the devices using web browser, owner of the system has to access the webpage and change the settings. Code for the software will be written in Embedded C for all actions and will be transferred to processor using serial port. Serial port is interfaced using driver/receiver interface. Microcontroller processes the sensed data and provides feedback to the website continuously. The ethernet module, into which the server is plugged, plays an important role in transferring web pages to the client. Code for the web page is designed in HTML and uploaded on server using Ethernet module and TCP/IP address. The IP address is a unique identification for the web server which can be controlled remotely from anywhere in the world as long as the authorization is in order.[7]

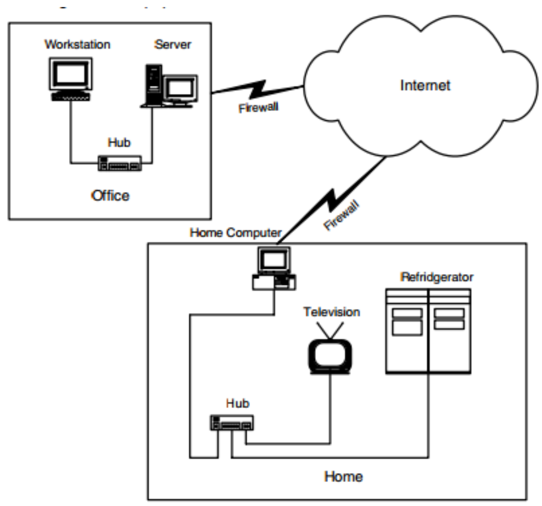

Fig 1: Monitoring Home Equipment from Office [7][8]

2.2 Hardware Structure

A web server can be embedded into an appliance to provide remote access for various devices located at home or industry. The embedded system stores dynamic and static information about the various parameters of the system and provides the same to the web browsers on demand. Such type of web server is called an Embedded Web Server. This server is a single hardware which contains a portable ARM processor along with an Ethernet module that contains the system software and application code for the overall process of automation. Embedded web servers are integral part of an embedded network. Fig. 2 shows concept of acquiring data from the system with an embedded web server on a single chip module. ARM processor is responsible for measuring signals and controlling the devices remotely. Measurements are done using DACs and the data is shared with clients through embedded web server. The system is completely designed to manage tasks such as data base updates, measuring signals, conversion of signals, sending HTML pages to client and communicating with the system owner using authentication password etc.

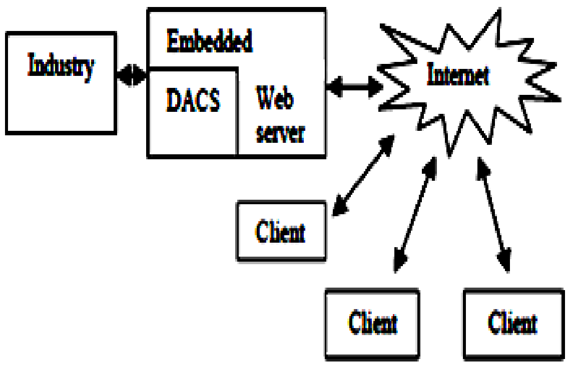

Fig 2: System Architecture

2.2.1 Memory

The web server has enough memory to develop large applications over the web server protocols. 32 Kbytes of external SRAM is used for buffering data. A 2-Mbit External Data Flash is used for storing web pages to allow a large amount of pages to be stored. The SRAM is connected to the address bus and data bus. The Serial Peripheral Interface (SPI) is used for communication with the Data Flash.

2.2.2 Ethernet-Controller

For this project selected Ethernet module is WIZ810MJ. This network module includes W5100 (TCP/IP hardwired chip, include PHY), MAG-JACK (RJ45 with X'FMR) with other glue logistics. The WIZ810MJ is an ideal option for those who want to develop their Internet enabling systems quickly .WIZ810MJ consists of W5100 and MAG-JACK. The Ethernet controller is configured as an 8-bit device. It features 4K bytes of internal memory which is accessed through the I/O registers or directly through memory mapping of the entire memory. Default operation on the Ethernet controller is I/O mode and address is 0300h. Since only address lines A0 - A12 are connected (need only 4 K bytes of address space), the I/O registers are mapped to address 8300h - 830Fh (I/O mode when the address lines into the PLD have the following configuration: bit 15 is high and bit 14 is low). By configuring the Ethernet controller through the I/O registers, the address can be changed and memory mode can be enabled. Memory mode operations can be mapped into address locations C000h - D000h.

2.2.2.1 Ethernet Specifications

- TCP/IP, MAC protocol layer : W5100
- Physical layer : Included in W5100
- Connector : MAG-JACK(RJ45 with Transformer)
- RJ-45 Connector RDA - 125BAG1A
- Input Voltage : 3.3V Internal Operation and

  5V Tolerant I/Os
- Power Consumption : 10/100 base T : Max 185mA (3.3V)

2.2.2.2 Features

- Supports 10/100 Base TX
- Supports half/full duplex operation
- Supports auto-negotiation and auto crossover
- Detection
- IEEE 802.3/802.3u Complaints
- Operates 3.3V with 5V I/O signal tolerance
- Supports network status indicator LEDs
- Includes Hardware Internet protocols: TCP, IP Ver.4, UDP, ICMP, ARP, PPPoE, IGMP
- Includes Hardware Ethernet protocols: DLC, MAC
- Supports 4 independent connections simultaneously
- Supports MCU bus Interface and SPI Interface
- Supports Direct/Indirect mode bus access
- Supports Socket API for easy application programming
- Interfaces with Two 2.0mm pitch 2 * 14 header pin

2.3. Software

The software running on the embedded web server follows the same layered structure as used in the TCP/IP protocol suite. The TCP/IP protocol suite allows computers of all sizes, running different operating systems, to communicate with each other. It forms the basis for what is called the worldwide Internet; a Wide Area Network (WAN) of several million computers .The TCP/IP protocol suite is a combination of different protocols at various layers .Every layer acts independently from each other. An Ethernet controller driver controls the Ethernet interface. The Address Resolution Protocol (ARP) translates IP addresses to Ethernet MAC addresses (and vice versa) The Internet Protocol (IP) delivers packets to Transmission Control Protocol (TCP), UDP, and Internet Control Message Protocol (ICMP), the ICMP answers to PING requests and TCP/UDP delivers data to the applications. The applications can communicate with the transport layer through buffers with data and variables with control information. This section explains how the TCP/IP protocol suite is built up in our approach.[7]

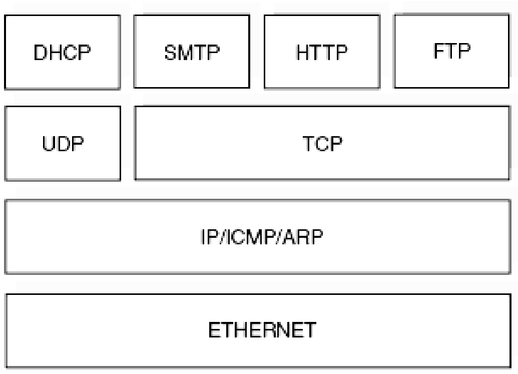

Fig 3: Protocol Stack[7][8]

2.3.1 Link Layer
Data-link or Network Interface Layer is another common name of this layer. The Link Layer normally includes the device driver in the operating system and the corresponding Network interface (card) in the computer. Together they handle all the hardware details of physically interfacing with the cable.[7] The Ethernet controller is configured to generate an interrupt every time a packet addressed directly to the Ethernet address arrives or when a broadcast arrives. When an interrupt occurs, the microcontroller reads the whole Ethernet frame into memory. A buffer of 1514 bytes, which is the maximum frame size on Ethernet, is reserved for this frame. Once the frame is transferred to the microcontroller, the Ethernet header is checked in order to ensure not receiving a misplaced frame. If the Ethernet address seen by the receiver is either a broadcast (all binary 1's) or addressed directly to specific Ethernet device, the frame is sent to the next layer or protocol according to the field, protocol type in the Ethernet header.

2.3.2 Network Layer
This layer is sometimes called the Internet Layer. It handles the movements of packets around the network. Routing of packets, for example, takes place here. IP (Internet Protocol) and ICMP (Internet Control Message Protocol) provides the Network Layer in the TCP/IP Protocol Suite.[7] The network layer controls the communication between hosts on the Ethernet. There is no form of transmission control to ensure that IP datagram's arrive to the host or that all IP datagram's from another host is received. This makes the layer rather easy to make. The ICMP sends messages between hosts and is only used to answer PING requests from a host. The IP handles communication for the overlaying Transport Layer.

2.3.3 Transport Layer
On the transport layer there are two major protocols which offer two different kinds of service; TCP which is a reliable delivery service and UDP which offers an unreliable service. TCP also offers flow control for retransmission of segments and acknowledgement of received segments.

2.3.4 Application Layer
The Application layer handles the details of a particular application.[7] Several applications may be implemented in the embedded web server. The main limitation is memory usage and performance. Running several Applications at once means lower performance.

2.3.5 Protocol Dependencies
Figure shows the protocol dependencies for the modules in the Embedded Web server. The protocols are described below:

- HTTPD needs TCP and a file-system/Data Flash to operate.[7]
- FTPD needs TCP and a file-system/Data Flash to operate.[7]
- SMTP needs a running TCP implementation to operate.
- DHCP needs IP and UDP to operate. In the initialization phase DHCP requires that the IP and UDP protocol forward any IP packets delivered before the IP address is configured.[7]

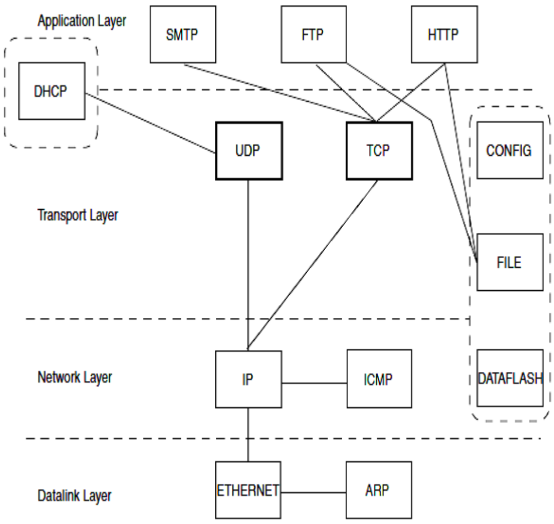

Fig 4: Protocol Dependencies[7][8]

- TCP needs IP to operate.[7]
- UDP needs IP to operate.[7]
- ICMP needs IP to operate.[7]
- IP needs ETHERNET to operate.
- ARP needs ETHERNET to operate
- ETHERNET depends only on the hardware Ethernet controller.[7]
- CONFIG needs file-system/Data Flash to read the configuration file. If the configuration file is unavailable it can return standard values.[7]
- FILE needs DATAFLASH to work.[7]
- DATAFLASH is only dependent on the hardware Data Flash.[7]
- MAIN must initialize Ethernet, Data Flash, and file-system, TCP, DHCP and HTTPD if these protocols are to be used.[7]
- DHCP, FTPD and HTTPD require repeatedly polling to operate.

2.3.6 User Developing Platform
When an application sends data using TCP, the data is sent down the protocol stack, through each layer, until it is sent as a stream of bits across the network. Each layer adds

information to the data by prep ending headers and adding trailers to the data it receives.

Some abbreviations:

- TCP segment: The unit of data that TCP sends to IP.
- IP datagram: The unit of data that IP sends to the network interface.
- Frame: The stream of bits that flows across the Ethernet.
- IP (Internet Protocol) adds an identifier to the IP header it generates to indicate which layer the data belongs to. IP handles this by storing an 8-bit value in its header called the protocol field. Similarly, many different applications can be using TCP or UDP at any time. The Transport Layer protocol stores an identifier in the header they generate to identify the application. Both TCP and UDP use 16-bit port numbers to identify applications. The TCP and UDP store the source port number and the destination port number in their respective headers. The network interface sends and receives frames on behalf of IP, ARP, and RARP. There must be some form of identification in the Ethernet header indicating which network layer protocol generates the data. To handle this, there is a 16-bit frame type field in the Ethernet header.

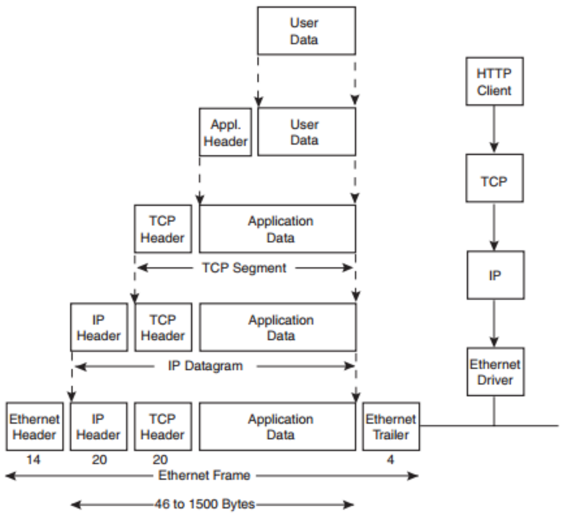

Fig 5: Encapsulation of Data as It Goes Down the Protocol Stack[7][8]

3. CONCLUSION

With the rapid development of the field of embedded systems and the wide range of applications of networks, digital distributed control system, intelligence, it is necessary for the data to be accurate and the reliability of the control system to be high. This embedded ARM system is designed to adapt to the strict requirements of the data and parameters in such a way that it can be utilized to control and automate all present systems. Their function, power consumption, reliability and remote access are managed by the existing system and can be extended to suite some additional features. This system can be widely applied to a lot of industries in the field of electric power, chemical, petroleum, metallurgy, steel, transportation, Electronic & Electrical industries, Automobiles and so on. The designed system can be used for controlling more no. of devices and the same can be tried out for wireless communications using GSM, Zigbee, GPRS etc. This Embedded ARM System can adapt to the requirements as a whole. By making use of LPC 2148 RISC machine - An embedded database, the collected data can be stored on the server and can be used for further processing. As ARM core is a fully accustomed and provided with USB Host device video processing can also be implemented. The classical service for field devices is done by a service engineer going into the field and visiting and maintaining every part of the equipment manually. A step forward is the usage of remote service capabilities offered by connecting to the device using a modem line. This gives the engineer the possibility to examine remotely the status of the device and to be better prepared if a visit is necessary. Often it will be enough to change some parameters or just to reset the device to ensure a proper function until the next maintenance visit. The benefit of this approach is limited in situations where in one place several different devices are located, often from several manufacturers. The limitation occurs due to usage of proprietary protocols to communicate and to the fact that the devices often could not be connected in a coherent network. Here the usage of standards concerning the networking (e.g. TCP/IP) and the higher level of communication (Internet protocols like http) could bring a major advantage in order to facilitate or even enable remote communication. The above system can be further used in industry to monitor the industrial system. By using more servo motor we can use this device as a haptic device. Further it can be used as patient monitoring system and can control and monitor the patient from anywhere.

4. REFERENCES

[1] Lixia Liu, Research on Technology of Embedded Web Server Application, IEEE, 2005.

[2] Chong FU, Zhi-liang ZHU, The Design & Implementation of General Reduced TCP/IP Protocol for EWS, IEEE, 2005.

[3] Janne Riihijarvi, Petri Mahonen, Provding Network connectivity for small appliances, IEEE, 2001.

[4] Masato Shimano, Futoshi Okazaki, Small Embedded web server for home appliances with embedded MPU & RTOS, IEEE, 2004.

[5] Tao Lin , Hai Zhao, An Embedded Web Server for Equipments, IEEE, 2004.

[6] G.Sunil Kumar, T.Swapna, Design an Embedded Web Server for Monitoring and Controlling Systems or Devices,IJETT,2013

[7] [EME99] Gary Desrosiers, 99. Embedded Ethernet. http://www.embeddedethernet.com/ [Accessed Feb 2000].

[8] [TCP86] Geoffrey H. Cooper. IMAGEN Corporation. http://www.csonline.net/bpaddock/tinytcp/default.htm [Accessed Feb 2000]

[9] Andrew S. Tanenbaum (2008): Computer Networks, 4th Edition.

[10] Cirrus Logic Product Data Sheet, Crystal LANTM ISA Etherenet Controller.

[11] Philips Semiconductors product data sheet: LPC2124.