

# Design and Implementation of Embedded Web Server in Industrial Control Systems

Jianming Liu<sup>1</sup>, Yunjie Zhang<sup>2</sup>, Lili Xu<sup>3</sup>, Jincui Wang<sup>1</sup>

<sup>1</sup>Computer Department, Weifang Medical University, Weifang, China

<sup>2</sup>Electronic & Information Engineering Department, Tianjin Institute of Urban Construction, Tianjin, China

<sup>3</sup>Weifang Vocational College, Weifang, China

Email: [sdwfljm@126.com](mailto:sdwfljm@126.com)

Received 2012

## ABSTRACT

In the traditional project of industry control, the field operator is the only man who watches production condition at the production spot. Enterprise administrators cannot obtain real-time production status and make real-time decision if they don't reach the production spot. Because of the development of Embedded Web Server such status is being changed. The remote monitor and control mode based on Embedded Web Server provides complete exact real-time information for managers, and makes it possible to make a decision in real time. In the paper, the design and implements of Embedded Web Server are presented and the key realization technology of it is described in detail. EWS (Embedded Web Server) will effectively lower system operating cost and improve maintaining efficiency, which has wide application prospect and popularization value in industrial control systems.

**Keywords:** Embedded Web Server (EWS); TCP/IP; HTTP; Industrial Control Systems

## 1. Introduction

With the rapid development of the computer and network technology, it has caused the technical revolution with the industrial control systems. Now, the network, opening structure becomes the main developing trend of the industrial control systems. At present, World-Wide Web (WWW) is one of the most widely used internet applications. The technology is very rapidly penetrating many social and business areas. Now most of embedded systems are still in use alone, so the embedded equipments should be improved as soon as possible. On the other hand, a huge amount of embedded equipments are based on 8-bit microcontrollers, such as instrument and apparatus, real-time application and process controlling, home automation, etc. Without exception, the industrial control systems can be equipped with Web servers for providing Web-based network communication. This type of Web server is called an Embedded Web Server (EWS). Based on EWS, the remote monitor and control mode provides complete exact real-time information for managers, and makes it possible to make a decision in real time [1].

In the conditions of EWS-equipped devices, the users can use web browser to manage devices and to receive and send data to browsers anywhere in the world in order to configure, monitor and control them. By the Web browsers, users can manage individual devices remotely.

This scheme provides a user with a simple but powerful user interface. Most embedded devices based on 8-bit microcontrollers, contain a small amount of memory, so in the design and implementation of EWS, some optimization techniques must be applied. In this paper, a kind of design and implementation of EWS has been introduced and the key realization technology of it is described in detail. The technology uses the 8-bit microcontrollers and network interface controller, such as UMC9005, RTL8019AS, CS8900, etc., and adopts thin TCP/IP protocols to transmit data. The result of experiment proves that it is feasible to realize the target at very low costs [2].

## 2. System Architecture of the EWS

The architecture of whole system is shown in the **Figure 1**. A EWS is a web server that has been designed to embed in different kinds of small devices. Anyone who is authorized may access the system if he knows the IP of the installed web server. The client doesn't need install any special software besides the internet Explorer or other browsers. By browser users who have been authorized can configure, control the embedded device, and also set up network of the web server.

A EWS communicates via thin HTTP to a Web browser by receiving a request which indicates the operation that the user wants to perform, and sends a re-

sponse in the form of a Web page for viewing. The URL determines what operation to perform, which is executed by the embedded system application. Further, to incorporate living data into a response, the EWS must dynamically create the Web page to return to the Web browser request. The living data is extracted from the embedded system application by the EWS [3].

### 3. Hardware Implementation

This Hardware structure of EWS is mainly made up of MCU, EEPROM, I/O interface and Ethernet communicate controller and isolate wave filter, as **Figure 2** shows. In this system we adopt AT89S52 as MCU, 24c02 as EEPROM, and RTL8019AS as network interface controller. The Hardware structure of EWS is shown in **Figure 2**.

RTL8019AS is a high-integrated 10Mbps Ethernet control chip produced by Realtek Company of Taiwan. RTL8019AS has finished all functions based on MAC layer of Ethernet protocol. AT24C02 is a 256B E2PROM, which is used for keeping the configuration information of EWS, for instance the IP address, MAC address, etc. So in different environment, the parameter of the EWS can be changed freely. In order to enhance the anti-interference ability of the TCP/IP network, segregation wave filter is used between RTL8019AS and RJ45 interface. For different application, the I/O interface and the

device are added to the EWS [1].

For some system, 8051 MCU is not suitable to process the task. Then the arm microprocessor is adopted widely in the Industrial control systems. The design of the EWS using S3C44B0X is show in **Figure 3**.

The embedded web server implementations usually choose two kinds of methods: 1) Embedded processor and network interface chip; 2) Embedded processor with Ethernet interface. In the design of this paper, it adopts the scheme 2. In the scheme, S3C44B0X processor and RTL901AS network interface chip are chosen. Its connection structure is as shown in Figure 3. RTL901AS is connected to S3C44B0X through the data bus and address bus. Besides, interrupt of the RTL901AS is connected to the NO 3, external interrupt the S3C44B0X.

### 4. Protocol Consideration

Web servers bring the desired HTML pages and pictures over the world wide internet to the Web browser. At the same time, TCP/IP communication happens. TCP/IP is the standard protocol used in internet, which is too large and too costly for most embedded applications. To make it sense to have a web server in every device, the TCP/IP stack used in device must be designed with reduced code size for embedded applications. The TCP/IP that we reduced is called thin TCP/IP.

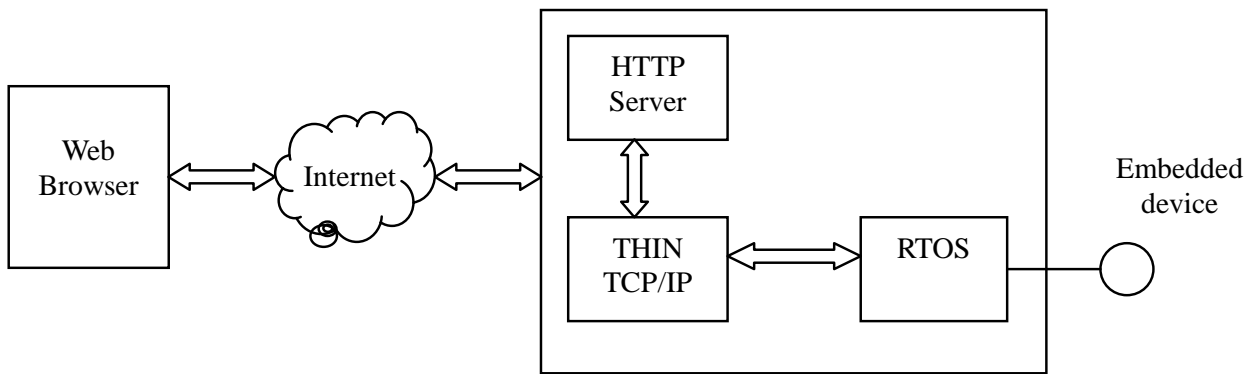


Figure 1. The architecture of EWS.

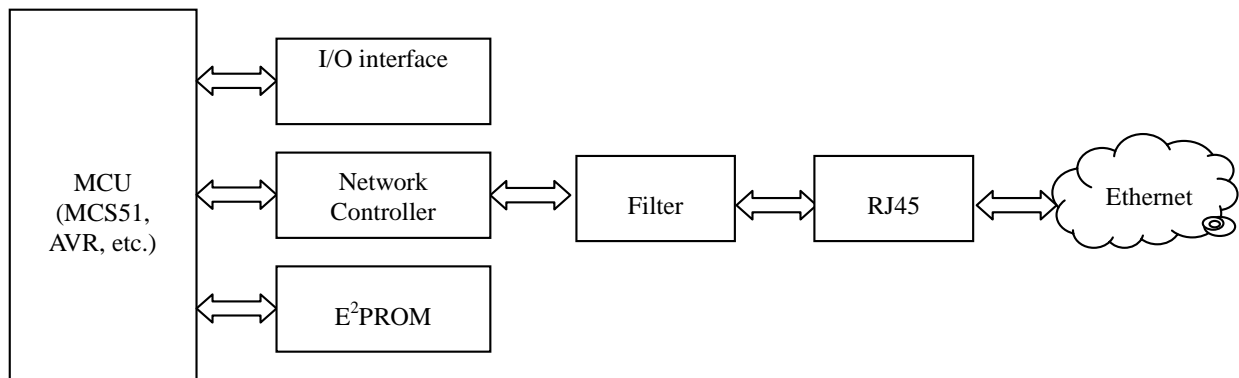


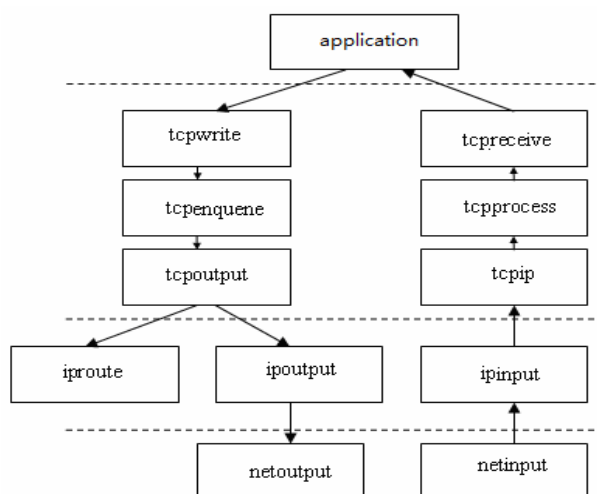
Figure 2. Hardware structure of EWS.

Simplicity and versatility are a major concern when designing thin TCP/IP. It is designed to be uncomplicated, straightforward, and flexible. Using a TCP/IP connection, the web server is accessed over the internet by a web browser to provide a graphical window into embedded devices. The web server generates web pages that are graphically displayed in the web browser. The server and the browser communicate via the HTTP protocol.

TCP/IP follows seven layers in OSI which is adopted into four layers (see **Figure 4**).

From the bottom of TCP/IP, network layer is the definition type of physical network media. Such as Ethernet, frame relay, ATM, and Token Ring. In internet layer there are protocols: IP, ARP, IGMP, and ICMP. In EWS we can only implement IP, ARP, and ICMP. IGMP does not need to be implemented, because we assume that routing process is conducted in routing table at local server. For Diagnostic functions, ICMP can be added. Then in Transport Layer there are two protocols, TCP and UDP. We only implement TCP, because web servers only need TCP. The implement of TCP protocol is shown in **Figure 5**.

TCP transmission control protocol provides a reliable byte stream oriented service. The basic TCP task is divided into 6 functional blocks, which includes tcpinput, tcpprocess, tcpreceive, tcpwrite, tcpenqueue, and tcpoutput. When the application wants to send TCP packets, tcpwrite function is called. The tcpwrite received data call tcpenqueue function, which is used to put the data



**Figure 5. The implement of TCP protocol.**

into the appropriate TCP data message if the network connection is ready. Tcp output function is used to check it can send these data. For example, the receiving end whether or not has adequacy receive buffer, and if condition is satisfied it then call the iproute and ipoutput function.

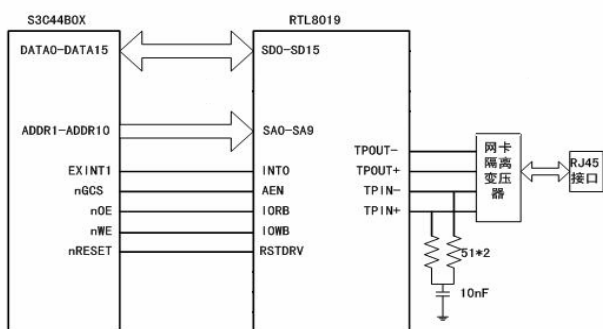
Address conversion if referred to Top address mapped to a physical address. ARP (Address Resolution Protocol) is responsible for the address conversion. The ARP protocol working process is described as follows: when the host generates an IP datagram, the datagram is sent to the destination host. The 32bit the IP address should be transformed into a 48bit Ethernet address, which is from a logical Internet address to a corresponding physical hardware.

The structure of the ARP Packets is as follows:

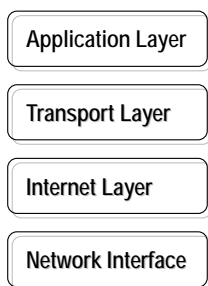
```

struct arp {
    unsigned int hardware; //hardware type
    unsigned int ARPopcode; //ARP operation
    unsigned int prot; //protocol type
    unsigned char hardwarelen; //hardware length
    unsigned char protlen; //protocol length
    unsigned char targeta[5]; //target's address
    unsigned char targetpa[3]; //target's protocol address
    unsigned char sendha[5]; //sender's address
    unsigned char sendpa[3]; //sender's protocol address
}
    
```

In the network layer, the IP protocol is the basis and the core of TCP/IP, which is responsible for sending the data in the sequence of operations. For different network between the host, such as TCP, UDP, ICMP and IGMP, the IP datagram format is used for transmission. IP is providing unreliable connectionless datagram transport services. The IP protocol function, is consisted of the



**Figure 3. The design of the EWS using S3C44B0X.**



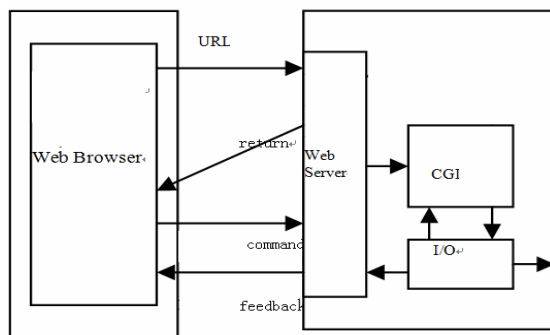
**Figure 4. TCP/IP protocol architecture layers.**

data to be transmitted combining with IP header processing, then packed IP datagram sending to the MAC layer. At the same time, the IP layer receives datagrams from the lower layer (such as Ethernet device driver). After header checksum and message processing, the received datagrams are sent to the higher level -- TCP or UDP layer.

The structure of the IP header is as following:

```
struct ipheader {
    unsigned char ipversionlen; //ip version &
header length
    unsigned char iptypeser; //type of Service
    unsigned int ippacketlen; //total packet length
    unsigned int ipdatagramid; //datagram id
    unsigned int ipfragmentoffset; //fragment offset
    unsigned char iptlmetollve; //tlmetollve
    unsigned char ipprotocol; //ip protocol
    unsigned int ipchecksum; //header checksum
    unsigned char ipsource[3]; //ip address of
source
    unsigned char ipdest[3]; //ip address of destina-
tion
};
```

The application layer of the TCP/IP is HTTP, which is made to send HTML file. HTTP is a standard protocol used for transferring documents between machines over TCP/IP. HTTP is used to manipulate interconnected documents around the globe. Each document is retrieved from a web server operating on some computers with access to the EWS. The document is used to refer to any type of data, which can be text, graphics, sound or video. **Figure 6** shows the model of the HTTP process. As in **Figure 4**, the most common HTTP client is the familiar web browser [4]. The web browser issues an URL request to the web server in order to access a document. The server must be operating before the browser initiates its requested HTML pages. The server is the depository for the web pages, and it handles requests and passes data back to the browser. The browser does more difficult work of presenting the text, displaying graphics, and running java applets.



**Figure 6.** The model of HTTP process.

## 5. The Program Flow Diagram of the EWS

Next **Figure 5** shows the program flow diagram of EWS. After data pack is delivered to the RTL8019AS and MCU through the RJ45 interface, the thin TCP/IP of system carries on a data analyzed and a judgment of the data flow direction, and wraps or unwraps data packets. So the rest work will be going on continually. Obviously, the end processing result is dealt with by web Server [6].

In the flow chat of micro programs, a box represents one micro instruction. If one operate should be performed, the corresponding field will be set '1', otherwise '0'. When the micro instruction is sent to data path from control memory, the corresponding module will work correctly. The machine instruction is interpreted and executed after the IR receives it. That is to say, it will execute the corresponding micro program segment. In accordance with the 3 higher bits (IR7-IR5), several branches are produced. Each of the branches presents one machine instruction. Finally, all of the codes of the micro instructions are obtained.

## 6. Conclusions

The EWS is a good media to attach device to internet. There are lots of benefits in applying EWS technology to industrial control systems field. The paper has presented and the key realization technology of EWS. This system is built with low cost and is easy to install. Many kinds of microcontrollers (8 bit) can be adopted to realize the system. In 21 century, termed Post-PC stage, it can be imagined that the traditional sever will become lean sever which can be operated without human intervention.

## REFERENCES

- [1] X. Peng, J. W. Xia and J. M. Liao, "An Embedded Internet Interface System," *Mobile Robots*, Vol. 8, No. 2, 2001, pp. 520-531.
- [2] Web Server for Embedded Systems, Klaus-D. Walter, 2005
- [3] G. -J. Han, M. Guan and H. Zhao. "EWS: Providing Internet Connectivity for non-PC Devices," *2004 IEEE Internet Conference on Networking, Sensing and Control (IEEE ICNSC'04)*, Taipei, Taiwan, Vol. 1, March 21 - 23, 2004, pp. 349-354.
- [4] G. J. Han, J. D. Wang, T. Lin and H. Zhao, "Design and Realization of Embedded Web Server Based on Web Management," *Journal of Northeastern University (Natural Science)*, Vol. 23, No.11, 2002, pp. 1021-1024.
- [5] J. Wang and S. J. Guo, "Implementation of Embedded Web Server and Its Application of CGI," *Microcontrollers and Embedded Systems*, Vol. 32, 2003, pp. 116-117.
- [6] D. Q. He, "Remote Multimedia Monitoring System Based on Embedded Web Server for Networked Manufacturing," *Journal of Chongqing University-Engineering*, Vol. 29, 2002, p.10.