RESEARCH ARTICLE OPEN ACCESS

Embedded ARM Control Robotic Arm using BoaWeb server – a Survey

V.Sirisha*, V. Sachin Kumar**
*(Department of Electronics and Communication Engineering, Sreenidhi Institute Of Science And Technology, Jawaharlal Nehru Technological University, Hyderabad-501301)
**(Department of Electronics and Communication Engineering, Scient Institute Of Technology, Jawaharlal Nehru Technological University, Hyderabad)

ABSTRACT
In today's market, the competing microprocessors are ARM (Advanced Risc Microprocessor), Intel, AMD. ARM is preferred since it has been powering portable devices for decades and has simple architecture to keep the energy waste to be minimum. The electronics advancements and embedded technology advancements have become a challenging field in today's techno world. In paper, the diligent features of embedded systems are introduced. It deals about how a robot is controlled using embedded operating system and ARM. Based on the combination of ARM, DSP and ARM Linux, the robot is controlled. The paper introduces development of embedded robot control system using Wi-Fi and also IOT. The embedded control system design includes four aspects. i.e., system structure, functions, hardware design and software design. By using these aspects (hardware and software adjustments), many robotic applications can be developed. Due to the fast execution speed and reasonable Ethernet speed in ARM processor, this system can be used in industrial oriented applications where there is very much necessity of safety and security.
Keywords **-** ARM, ARM LINUX,Boa server, Embedded operating system, Embedded web server.

I. INTRODUCTION

1. Embedded Operating System

An embedded operating system is an operating system for embedded computer systems. These operating systems are designed to be compact, efficient at resource usage, and reliable for many functions that non-embedded computer operating systems provide, and which may not be used by the specialized applications they run. They are frequently also referred to as real-time operating systems .

An important difference between most embedded operating systems and desktop operating systems is that the application, including the operating system, is usually statically linked together into a single executable image. Unlike a desktop operating system, the embedded operating system does not load and execute applications. This means that the system is only able to run a single application.

The advantages of embedded Linux over proprietary embedded operating systems include multiple suppliers for software, development and support, no royalties or licensing fees, a stable kernel, the ability to read, modify and redistribute the source code. The technical disadvantages include a comparatively large memory footprint (kernel and root file system), complexities of user mode and kernel mode memory access, and a complex device driver's framework.

As more advanced control algorithms are becoming available for the control of robotic arms, traditional fixed controller boards and associated code generators are becoming less convenient way to test such control algorithms in real-time. The process of using such boards is complex, time consuming, and inflexible. By the advancement of electronics, embedded technology has become a challenging field in this modern age. The single functioned tightly constrained, reactive and real-time feature of these devices enhanced its importance in industrial, consumer applications.

1.1 Literature Survey

1. Alen Rajan in his paper [1] proposed that embedded technology is one of the emerging technologies in this most modern era. When networking technology is incorporated with the former, there is no doubt that the scope of embedded systems would be further more.
2. Alen Rajan has highlighted in the paper [2] that a robot arm is an Electro-mechanical device that performs various tasks ranging from simple mechanical jobs to highly complex tasks. It can be used to pick and place small parts on a production line.
3. Li Yanhong proposed that [3], it can replace the human operator in feeding industrial process with discrete components.

4. Reza Ezuan Samin in his paper [4] explained that, with the increase usage of wireless application, the demand for a system that could easily connect devices for transfer of data over a long distance - without cables, grew stronger. Robotic arms are used in diverse manufacturing processes including assembly, spot welding, laser processing, cutting, grinding, polishing, testing, painting and dispensing.
5. Tetsuya Akagia proved in his paper [5] that, Robots have proved to help automakers to be more agile, flexible and to reduce production lead times. The robot arm using in this paper was designed with DC motors which are driven by the driver circuit
6. V.Billy Rakesh Roy1, Sanket Dessai1, and S. G.Shiva Prasad Yadav in his paper [6] explained about The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are much simpler than those of micro programmed Complex Instruction Set Computers. This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective processor core. Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory ARM based embedded system will be more functional, reliable, cost effective, less in size and low power consumption. Microcontroller has low speed and poor memory, so it can only execute simple control tasks.
7. Mohd Ashiq Kamaril Yusoff in his paper [7] explained about a robotic arm saying that a robotic arm is a robot manipulator, usually programmable, with similar functions to a human arm. The links of such a manipulator are connected by joints allowing either rotational motion (such as in an articulated robot) or translational (linear) displacement. The links of the manipulator can be considered to form a kinematic chain. The business end of the kinematic chain of the manipulator is called the end effectors and it is analogous to the human hand. The end effectors can be designed to perform any desired task such as welding, gripping, spinning etc., depending on the application. The robot arms can be autonomous or controlled manually and can be used to perform a variety of tasks with great accuracy. The robotic arm can be fixed or mobile (i.e. wheeled) and can be designed for industrial or home applications. The wireless mobile robots also have been developing in previous years.
8. Junhua Yang in his paper [8] explains that Since ARM is not directly connected with the Internet system. A kind intelligence monitoring system based on 32 bit ARM processor LPC2138 and information fusion technology was provided. ARM has advantages of high integration and powerful information processing capability.
9. K.Bharath reddy, Ch. Rajendra Prasad in their paper [9] explains about the system structure of embedded Web server. The entire system uses B/S mode. The client PC is connected to the Internet through a browser and then gets access to the embedded Web server. Through this way, remote login and operation are realized. Embedded Web Server (EWS) is a Web server that runs on an embedded system with limited computing resources and serves embedded Web documents to a Web browser. By embedding a Web server into a network device, it is possible for a EWS to provide a powerful Web-based management user interface constructed using HTML, graphics and other features common to Web browsers. When applied to embedded systems, Web technologies offer graphical user interfaces, which are user-friendly, inexpensive, cross-platform, and network-ready. A Web server can be embedded in a device to provide remote access to the device from a Web browser if the resource requirements of the Web server are reduced. The end result of reducing the resource requirements of the Web server is typically a portable set of code that can run on embedded systems with limited computing resources.
10. Vaishak N. L,C.G. Ram Chandra explained about ARM processor in his paper[10] explained the software part was developed by using embedded C. Existing system robot generally works with microcontroller and it is basically wired robots which works on CISC microprocessor. Proposed System introduces the configuration of the embedded system, and then presents a robot control system based on an embedded operating system and ARM. Based on the combination of advanced RISC microprocessor (ARM), DSP and ARM-Linux, this project involves development of embedded robot control systems through Wi-Fi. Here we use ARM controller as the heart of the system. ARM has high speed of execution and powerful information processing capability. The capacity of multi-parameter execution, multi-level monitoring and networking of ARM processor makes it suitable for a wide variety of networking applications.

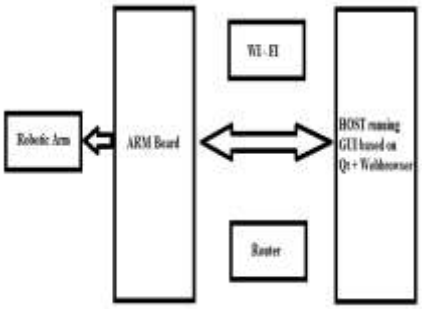

Fig.1 Functional Diagram

II. ARM MICROPROCESSOR

2. ARM Architecture

ARM architecture is the most widely used 32-bit ISA in terms of numbers produced. They were originally conceived as a processor for desktop personal computers by Acorn Computers, a market now dominated by the x 86 families used by IBM PC compatible and Apple Macintosh computers. The ARM is a 32-bit reduced instruction set computer (RISC) instruction set architecture (ISA) developed by ARM Holdings. It was known as the Advanced RISC Machine, and before that as the Acorn RISC Machine. The relative simplicity of ARM processors made them suitable for low power applications. This has made them dominant in the mobile and embedded electronics market, as relatively low cost, and small microprocessors and microcontrollers. By systematic maintenance it is possible to achieve substantial savings in money, material and manpower as every effort is directed towards avoiding catastrophic failures

2.1 RISC Features

1. The ARM architecture includes the following RISC features:
2. Load/store architecture.
3. No support for misaligned memory accesses (now supported in ARMv6 cores, with some exceptions related to load/store multiple word instructions).
4. Uniform 16 × 32-bit register file.
5. Fixed instruction width of 32 bits to ease decoding and pipelining, at the cost of decreased code density. Later, "the Thumb instruction set" increased code density.
6. Mostly single-cycle execution. To compensate for the simpler design, compared with contemporary processors like the Intel 80286 and Motorola 68020, some additional design features were used:
   a. Conditional execution of most instructions, reducing branch overhead and compensating for the lack of a branch predictor.
   b. Arithmetic instructions alter condition codes only when desired.
   c. 32-bit barrel shifter which can be used without performance penalty with most arithmetic instructions and address calculations.
   d. Powerful indexed addressing modes.
   e. A link register for fast leaf function calls.
   f. Simple, but fast, 2-priority-level interrupts subsystem with switched register banks.

2.1.1 Conditional Execution

The conditional execution feature (called predication) is implemented with a 4-bitcondition code selector (the predicate) on every instruction; one of the four-bit codes is reserved as an "escape code" to specify certain unconditional instructions, but nearly all common instructions are conditional. Most CPU architectures only have condition codes on branch instructions. This cuts down significantly on the encoding bits available for displacements in memory access instructions, but on the other hand it avoids branch instructions when generating code for small if statements. One of the ways that Thumb code provides a denser encoding is to remove that four bit selector from non-branch instructions.

2.1.2 Instruction Set

To keep the design clean, simple and fast, the original ARM implementation was hardwired without microcode, like the much simpler 8-bit 6502 processor used in prior Acorn microcomputers.

2.1.3 Pipelines and other Implementation Issues

The ARM7 and earlier implementations have a three stage pipeline; the stages being fetch, decode, and execute. Higher performance designs, such as the ARM9, have deeper pipelines: Cortex-A8 has thirteen stages. Additional implementation changes for higher performance include a faster adder, and more extensive branch prediction logic.

2.1.4 Coprocessors

The architecture provides a non-intrusive way of extending the instruction set using "coprocessors" which can be addressed using MCR, MRC, MRRC, MCRR, and similar instructions. The coprocessor space is divided logically into 16 coprocessors with numbers from 0 to 15, coprocessor 15 (cp15) being reserved for some typical control functions like managing the caches and MMU operation (on processors that have one).In ARM-based machines, peripheral devices are usually attached to the processor by mapping their physical registers into ARM memory space or into the coprocessor space or connecting to another device (a bus) which in turn attaches to the processor.

2.1.5 Debugging

All modern ARM processors include hardware debugging facilities; without them, software

debuggers could not perform basic operations like halting, stepping, and break pointing of code starting from reset.

2.1.6 Jazelle

Jazelle is a technique that allows Java Byte code to be executed directly in the ARM architecture as a third execution state (and instruction set) alongside the existing ARM and Thumb-mode. Support for this state is signified by the "J" in the ARMv5TEJarchitecture, and in ARM9EJ-S and ARM7EJ-S core names. Support for this state is required starting in ARMv6 (except for the ARMv7-M profile), although newer cores only include a trivial implementation that provides no hardware acceleration.

III. Embedded Web Server

If General web servers which were developed for general-purpose computers such as NT servers or UNIX and Linux workstations typically require megabytes of memory, a fast processor, a Pre-emptive multitasking operating system, and other Resources. A web server can be embedded in a device to Provide remote access to the device from a web browser.

The embedded system can be utilized to serve the embedded web documents, including static and dynamic Information about embedded systems, to web browsers.This type of web server is called an Embedded Web Server.

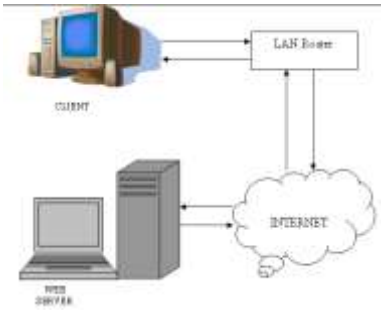

Fig 3.1 Client-Server Architecture

An embedded web server is a microcontroller that contains an Internet software suite as well as application Code for monitoring and controlling systems. Embedded Web servers are integral part of an embedded network and Paves way for faster time to market products. “Fig 3.1”. Shows the general-purpose web server where, it requires a huge amount of memory, special hardware, Software and an operating system. An embedded web Server can replace the “Fig 3.2” which is a single hardware With an RTOS and the application.

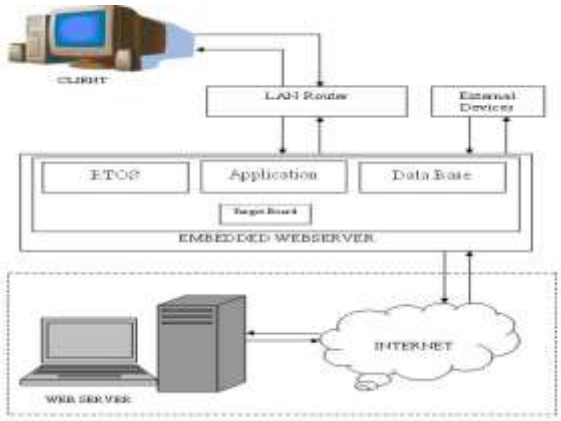

Fig 3.2 Embedded Web Server Architecture

The web server is the Board that has the application and the RTOS (µC/OS-II). The operating system manages all the tasks such as sending the HTML pages, connecting to new users etc.

The RTOS manages all the required tasks in parallel, and in small amounts of time. Web based management user Interfaces using embedded web servers have many Advantages: ubiquity, user-friendliness, low development Cost and high maintainability. Embedded web servers have different requirements, such as low resource usage, High reliability, security and portability, for which general Web server technologies are unsuitable. There are also design issues such as HTTP and embedded API.

3.1 The system diagram of Embedded Web server

The system structure of embedded Web server is shown in figure 3.1. The entire system uses B/S mode. The client PC is connected to the Internet through a browser and then gets access to the embedded Web server. Through this way, remote login and operation are realized. Compared with the traditional C/S mode, this mode is simple to use, convenient to maintain, and easy to extend Figure 3.1(The system diagram of Embedded Web server).:

3.2 The choice of Embedded Web server

The embedded devices have limited resources and don't need to handle the requests of many users simultaneously. Therefore they do not need to use the most commonly used Linux server Apache. Web Server which is specifically designed for embedded devices are applied in such case. This kind of Web Server requires relatively small storage space and less memory to run, which makes it quite suitable for embedded applications. The typical embedded Web server has three kinds, namely https, Boa and thttpd. As the simplest Web server, https has the weakest functions among the three. It does not support authentication and CGI technology while Boa and thttpd support these functions. If Web server only provides some static Web pages such as simple

online help and system Introduction, then a static server can be adopted. If you need to improve system security or interact with user Such as real-time status query and landing, then you have to use dynamic Web technologies.

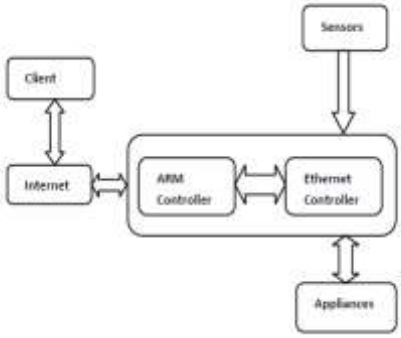

Fig 3.3- EWS With External Ethernet Controller

In such situation Boa can achieve these goals. This is vital important for embedded systems by saving the maximum extent possible system resources. Based on the above exposition, Boa applied to the embedded platform has many advantages. Therefore Boa is used as Web server .Its architecture is showed in Fig.3.3 EWS with External Ethernet Controller.In fig 3.4 - Architecture of boa server, if you need to improve system security or interact with users such as real-time status query and landing, then you have to use dynamic Web technologies. In such situation, either Boa or thttpd can achieve these goals. In the present research, we adopt Boa (the Web server) suitable for embedded system, because thttpd has less function and needs far more resources to run.

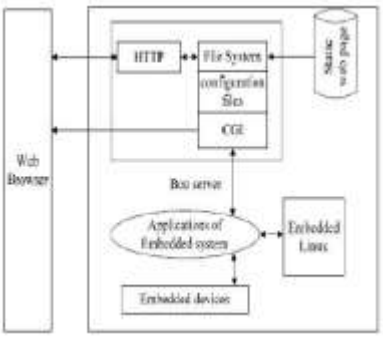

Fig 3.4 Architecture Of Boa Server

3.3 The principle of Embedded Web server Boa

Boa is a single task Web server. The difference between Boa and traditional Web server is that when a connection request arrives, Boa does not create a separate process for each connection, nor handle multiple connections by copying itself. Instead, Boa handles multiple connections by establishing a list of HTTP requests, but it only forks new process for CGI program. In this way, the system resources are saved to the largest extent. Like a common Web server, an embedded web server can accomplish tasks such as receiving requests from the client, analysing requests, responding to those requests, and finally returning results to the client. The following is its work process.

1. Complete the initialization of the Web server, such as creating an environment variable, creating socket, binding port, listening to a port, entering the loop, and waiting for connection requests form a client. When there is a connection request from a client, Web server is responsible for receiving the request and saving related information.
2. After receiving the connection request, Boa analyses the request, calls analysis module, and works out solutions, URL target, and information of the list. At the same time, it processes the request accordingly.
3. After the corresponding treatment is finished, the Web server sends responses to the client browser and then closes the TCP connection with the client. For different request methods, the embedded Web server Boa makes different responses. If the request method is HEAD, the response header will be sent to the browser. If the request method is GET, in addition to sending the response header, it will also read out from the server the URL target file of the client request and send it to the client browser. If the request method is POST, the information of the list will be sent to corresponding CGI program and then take the information as a CGI parameter to execute CGI program. Finally, the results will be sent to client browser.

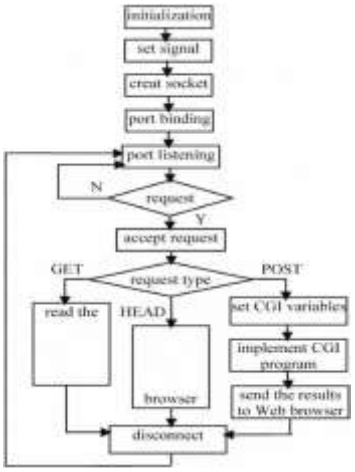

Fig 3.5 – Flow chart of Boa Server

IV. Hardware Design

S3C2440AL processor is used as core of the hardware platform in this paper. Fig. 6 is the block diagram of hardware system. Include: serial port, Ethernet interface, JTAG port, storage systems and so on. The frequency Samsung S3C2440AL is 400MHz and can up to 533MHz in the maximum. According

to its mode of internal circuit, 12 MHz chosen for the crystal. JTAG (Joint Test Action Group) is an international test protocol standard, software simulation, single-step debug and u-boot download can be carried out through the JTAG port, it's a simple and efficient means of developing and debugging embedded systems. The SDRAM capacity in the system is 64MB, working voltage is 3.3V, data bus is 32-bit, clock frequency up to100MHz, Auto-Refresh and Self-Refresh are both supported.

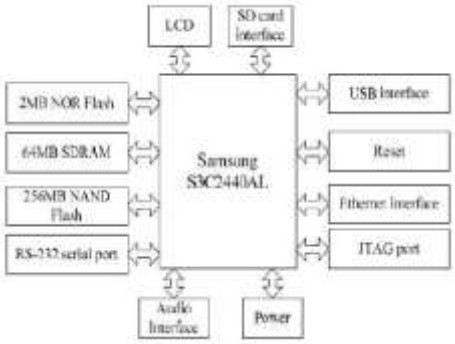

Fig 4-Block Diagram of Hardware Design

V. Software Desgin

Software development process based OS includes: the establishment of cross-compiler, the transplant of Boot loader, the transplant of embedded Linux, the development embedded Web server. To begin with, system cross-compiler environment using EABI-4.3.3 is established.uboot that developed by the German DEXN group is used as Boot loader. The function of Boot loader is to initialize the hardware devices, establish memory mapping tables, thus establish appropriate hardware and software environment and prepare for the final call to the operating system kernel. Besides, yaffs file system is made.

Linux is used as operating system because Linux system is a hierarchical structure and completely open its kernel source, the important feature of Linux is portability to support a wide range of hardware platforms, can run in most of the architecture. Contains a comprehensive set of editing, debugging and other development tools, graphical interface, a powerful network supporting and rich applications. In addition, the kernel can be reduced by configuring.

VI. Conclusion

When the case of monitoring multiple parameters comes, the EWS with integrated Ethernet is showing better performance when speed and reliability comes into picture. Thus EWS with integrated Ethernet is suitable for real-time monitoring of Industrial appliances. Moreover this system has a wide variety of Industrial applications such as remote monitoring and controlling etc. Since ARM processor has fast execution capability and Ethernet standard can provide internet access with reasonable speed and this system is suitable for enhancing security in industrial conditions by remotely monitoring various industrial appliances where high safety and care is a necessity. Finally, this system will be useful for a wide variety of industrial applications.

VII. Future Scope

This system can be extended with new web servers and by using Raspberry Pi and many applications can be developed.

REFERENCES

[1] Alen Rajan, Aby K. Thomas, "*ARM Based Embedded Web Server for Industrial Applications*", International Conference on Computing and Control Engineering (ICCCE 2012), 12 & 13 April, 2012.

[2] Alen Rajan, Aby K. Thomas, Rejin Mathew, "*A Comparative Performance Analysis of ARM based Web Servers with Integrated and External Ethernet Interfaces for Industrial Applications*", International Journal of Computer Applications(IJCA 0975 – 8887) Volume 44– No.21, April 2012.

[3] Li Yanhong, Li Shuliang, *"Based on the ARM and PID Control Free Pendulum Balance System",* 2012 International Workshop on Information and Electronics Engineering (IWIEE), Procedia Engineering 29 (2012) 3491 – 3495.

[4] Mohd Ashiq Kamaril Yusoff, Reza Ezuan Samin, Babul Salam Kader Ibrahim, "*Wireless Mobile Robotic Arm*", International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), Procedia Engineering 41 (2012) 1072 – 1078.

[5] Mohd Aliffa, Shujiro Dohtaa, Tetsuya Akagia, Hui Lia, "*Development of a simple-structured pneumatic robot arm and its control using low-cost embedded controller*", International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), Procedia Engineering 41 (2012) 134 – 142.

[6] V.Billy Rakesh Roy, Sanket Dessai, and S. G.Shiva Prasad Yadav;" *Design and Development of ARM Processor Based Web Server*," International Journal of Recent Trends in Engineering(IJRET), Vol. 1, No. 4, May 2009

[7] Mohd Ashiq Kamaril Yusoff, Reza Ezuan Samin, Babul Salam Kader Ibrahim*;" Wireless Mobile Robotic Arm*," International

Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)

[8] Junhua Yang; Zhien Shang and Tao Xing, *"Intelligence Monitoring System Based on ARM and Information Fusion,"* (IEEE) International Conference on Electric Information and Control Engineering, pp.487-490, April 2011.

[9] K.Bharath reddy, Ch. Rajendra Prasad; *"The Embedded Web server based Electrical Ethernet Monitoring system using ARM,"* International Journal of Advanced Research in Computer and Communication Engineering,Vol. 2, Issue 5, May 2013

[10] Vaishak N. L, C.G. Ram Chandra; "*Embedded Robot Control System Based On an Embedded Operating System, the Combination of Advanced RISC Microprocessor (ARM), DSP and ARM Linux, "*International Journal of Engineering and Innovative Technology (IJEIT) Vol. 2, Issue 6, December 2012.