Embedded Web Server for Real-Time Remote Control and Monitoring of a Raspberrypi based on Open Source Platform

G Ahmed Zeeshan¹, Dr. RSundar Guru², Syed Rahil³  
Assistant Professor¹, Ph.D, Professor and Head², M.Tech³  
Department of Electronics and Communication Engineering  
Global Institute of Engineering and Technology, Moinabad, Hyderabad, Telangana, India¹,³  
MVIT, Bangalore, Karnataka, India²

Abstract:  
Many FPGA-based embedded systems in space applications, automobiles, medical and industrial fields use an on-board computer system that needs remote emulation, monitoring, control and update. This paper describes an approach for the development of a real-time remote control and monitoring system using an embedded web server implemented in a MicroBlaze processor based software design, which takes Xilkernel as real-time operating system. We used the open source Lightweight IP (lwIP) standalone TCP/IP protocol stack to implement networking capability. The embedded web server will enable remote access to the on-board computer system by using a web browser for receive the downlink data, send the uplink commands or update the system. The embedded platform was implemented in Xilinx Spartan-6 FPGA SP605 Evaluation Kit.

Keywords: Embedded Web Server; lwIP; Xilkernel; RTOS; FPGA; MicroBlaze; HTTP; On-Board Computer

I. INTRODUCTION  
A new generation of reconfigurable FPGA-based on-board computer (OBC) has experienced a rapid development in a wide range of applications that include upgrade capability (Space and automobile applications, medical field, industrial control system). This was accompanied by an increased need to monitor, maintain and update these systems. On the other hand, Web technology becomes widely used and there is an ascending trend to migrate everything online and to remotely control systems. In this regard, several studies have emerged suggesting the implementation of the embedded Web server technology to provide a monitor and control system based on an Internet browser and an embedded TCP/IP protocol suite.

II. PROPOSED SYSTEM  
In comparison with PC, the embedded system is greatly improved in stability, reliability and safety etc. The embedded system transplanted web server can be called embedded web server. Through web page posted by embedded web server, remote users can obtain the real-time status information and control remote equipment’s without time and space restriction. In this project the monitoring and controlling of devices is achieved using TCP/IP protocol and the Raspberry Pi. The TCP/IP protocol is the vital part, as it is used for the implementation of Embedded Web server. This project consists of two parts, a server module which is connected to internet through RJ45 interface and a sensor module which always reads values from the sensors and is published on the internet. So the user can monitor and control the devices through Browser. When the client types the correct IP address in the browser, the webpage displays the sensors values to the user and if there are any discrepancies in the values, then the user can control the device through a relay connected to the sensor module. Apart from controlling the devices remotely, automatic controlling procedure is also provided to control the devices.

III. HARDWARE IMPLEMENTATION  
ARM Architecture:  
The Raspberry Pi is a credit-card-sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools. The Raspberry Pi is manufactured in two board configurations through licensed manufacturing deals with Newark element14.

Figure: 1. block Diagram

Figure: 2. Hard Ware Implementation
These companies sell the Raspberry Pi online. Egoman produces a version for distribution solely in China and Taiwan, which can be distinguished from other Pis by their red coloring and lack of FCC/CE marks. The hardware is the same across all manufacturers. The Raspberry Pi has a Broadcom BCM2835 system on a chip. It does not include a built-in hard disk or solid-state drive, but uses an SD card for booting and persistent storage.

Figure 3: Main programming language, with support for BBC BASIC

The Foundation provides Debian and Arch Linux ARM distributions for download. Tools are available for Python as the main programming language, with support for BBC BASIC (via the RISC OS image or the Brandy Basic clone for Linux), C, Java and Perl.

**Ethernet:**

**Ethernet LAN Features:**
- Bus topology, Wired LAN in IEEE 802.3 physical layer standard
- 10 Mbps, 100 Mbps (Unshielded and Shielded wires) and 4 Gbps (in twisted pair wiring mode)
- Broadcast medium-Passive, Wired connections based.
- Frame format like the IEEE 802.2
- SNMP (Simple Network Management Protocol) Open system (therefore allows equipment of different specifications)
- Each one connected to a common communication channel in the network listens and if the channel is idle then transmits. If not idle, waits and tries again.
- Multi access is like in a Packet switched network

**IV. SOFTWARE REQUIREMENTS**

**A. Linux Operating System:**

Linux or GNU/Linux is software operating for computers. The operating system is a collection of the basic instructions that tell the electronic parts of the computer what to do and how to work. Free and open source software (FOSS) means that everyone has the freedom to use it, see how it works, and changes it. There is a lot of software for Linux, and since Linux is free software it means that none of the software will put any license restrictions on users. This is one of the reasons why many people like to use Linux. A Linux-based system is a modular Unix-like operating system. It derives much of its basic design from principles established in UNIX during the 1970s and 1980s. Such a system uses a monolithic kernel, the Linux kernel, which handles process control, networking, and peripheral and file system access. Device drivers are either integrated directly with the kernel or added as modules loaded while the system is running.

![Diagram of the Linux Operating System](http://ijesc.org/)

**Figure 4: Architecture of Linux Operating System**

**B. Qt for Embedded Linux:**

Qt is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases Qt is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers. Qt uses standard C++ but makes extensive use of a special code generator (called the Meta Object Compiler, or moc) together with several macros to enrich the language. Qt can also be used in several other programming languages via language bindings. It runs on the major desktop platforms and some of the mobile platforms. Non-GUI features include SQL database access, XML parsing, thread management, network support, and a unified cross-platform application programming interface for file handling. It has extensive internationalization support.

**V. RESULTS**
VI. CONCLUSION
The project “Embedded Web Server for Real-time Remote Control and Monitoring of a RaspberryPi based on Open source Platform” has been successfully designed and tested. It has been developed by integrating features of all the hardware components and software used. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit. Secondly, using highly advanced ARM board and with the help of growing technology the project has been successfully implemented.

VII. REFERENCES


