IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735. Volume 7, Issue 6 (Sep. - Oct. 2013), PP 47-50
www.iosrjournals.org

Ethernet Enabled Digital I/O Control in Embedded Systems

L.S.Narayana Reddy[1], B.T.Vedanth[2]
[1]Student,[2]Assistant Professor
[1] *Ece Department, Mallareddy Institute Of Engineering And Technology.*
[2]*hod, Ece Department, Mallareddy Institute Of Engineering And Technology.*
Hyderabad, Ap, India.[1], Hyderabad, Ap.India[21]

Abstract: *This paper presents very simple and economical way to provide Ethernet connectivity to micro controller based embedded systems. This system uses arm7 controller to store the main application source code, web pages and TCP/IP stack which is a vital element of the system software. an Ethernet controller chip,ENC28J60 is used to handle the Ethernet controller chip.ENC28J60 is used to handle the Ethernet communications and is interfaced with the host micro controller using SPI pins. There are several I/O pins available at the micro controller which are used to interface with sensors and relays for monitoring and controlling operations. Now a day, internet has spread worldwide and most of the internet connections use Ethernet as media for data transfer. in industries or in home appliances ,most of the time we need to monitor and control different parameters using micro controllers. Once we enable Ethernet interface to such systems, we can communicate with them remotely over the internet*

Keywords:*Ethernet controller; TCP/IP stack; serialperipheral interface (SPI),ENC28J60, Triac,Sensors.*

I. Introduction

For many years, embedded systems and Ethernet networks existed in separate worlds. Ethernet was available only to desktop computers and other large computers. Embedded systems that needed to exchange information with other computers were limited to interfaces with low speed, limited range, or lack of standard application protocols. But developments in technology and the marketplace now make it possible for embedded systems to communicate in local Ethernet networks as well as on the Internet. Network communications can make an embedded system more powerful and easier to monitor and control [I].

Ethernet solves the problem of remote communication with the embedded application. Ethernet is a data link and physical layer protocol defined by the IEEE 802.3 specification [3]. Ethernet is versatile enough to suit many purposes. Ethernet can transfer any kind of data, from short messages to huge files. An Ethernet communication can take advantage of existing higher-level protocols such as TCP and IP, or it can use an application-specific protocol. Ethernet doesn't require a large or fast computer. Challenges like application monitoring, control, diagnostics and data logging can all be accomplished from a remote, centralized location. With the ability to access the application remotely, corporations can eliminate the need to send a service person to the application and thus save labor time and money.

The ENC28J60 is a stand-alone Ethernet controller with an industry standard Serial Peripheral Interface SPI.It has on-chip [10] Mbps Ethernet Physical Layer Device (PHY) and Medium Access Controller (MAC), providing reliable packet-data transmission/reception based on an industry standard Ethernet protocol [4]. The PHY contains analog circuitry to encode and decode the data on the twisted pair interface while the MAC contains digital circuitry to control when to transmit, handle automatic retransmission when a collision occurs, calculates and validates CRCs (Cyclical Redundancy Check), and do other necessary tasks.There are compelling reasons behind considering Ethernet for remote communication. Ethernet is the most widely deployed network in offices and industrial buildings. Ethernet's infrastructure, interoperability and scalability ensure ease of development. Once equipment is connected to a Ethernet network, it can be monitored or controlled through the Internet removing any distance barrier that may have inhibited remote communication previously.

II. System Design Model

A. Software design module

It is possible to create the source files in a text editor such as Notepad, run the Compiler on each C source file, specifying a list of controls, run the Assembler on each Assembler source file, specifying another list of controls, run either the Library Manager or Linker (again specifying a list of controls) and finally running the Object-HEX Converter to convert the Linker output file to an Intel Hex File. Once that has been completed the Hex File can be downloaded to the target hardware and debugged. Alternatively KEIL can be used to create source files; automatically compile, link and covert using options set with an easy to use user interface and finally simulate or perform debugging on the hardware with access to C variables and memory. Unless you have to use the tolls on the command line, the choice is clear. KEIL Greatly simplifies the process of creating and testing an embedded application.

The user of KEIL centers on "projects". A project is a list of all the source files required to build a single application, all the tool options which specify exactly how to build the application, and – if required – how the application should be simulated. A project contains enough information to take a set of source files and generate exactly the binary code required for the application. Because of the high degree of flexibility required from the tools, there are many options that can be set to configure the tools to operate in a specific manner. It would be tedious to have to set these options up every time the application is being built; therefore they are stored in a project file. Loading the project file into KEIL informs KEIL which source files are required, where they are, and how to configure the tools in the correct way. KEIL can then execute each tool with the correct options. It is also possible to create new projects in KEIL. Source files are added to the project and the tool options are set as required. The project can then be saved to preserve the settings. The project is reloaded and the simulator or debugger started, all the desired windows are opened. KEIL project files have the extension

B. Hardware design module

BLOCK DIAGRAM

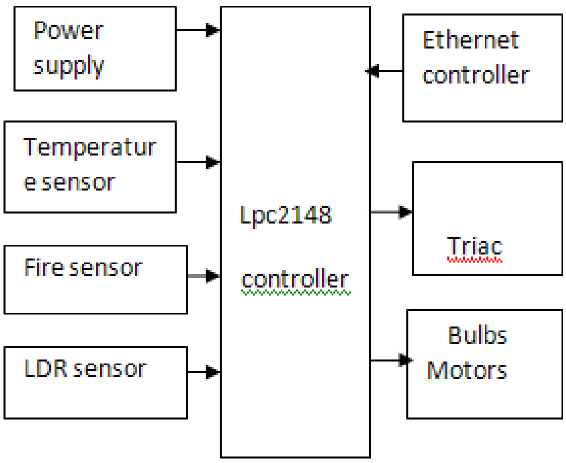

Figure: experimental set-up block diagram

The project "Ethernet enabled digital i/o control in embedded systems". Consists of hardware devices like arm7 controller, Ethernet controller, temperature sensor, LDR sensor, fire sensor, loads (bulb, dc motor), traicIC.In this project we control the home appliances through web page which is operated through remote communication of Ethernet which is interfaced with the micro controller. Whenever we switch on and off the bulb and motor via web page,it occurs through sensors. When LDR is sensed then bulb is in off condition. LDR means light dependent resistor.If the temperature sensor is sensed then motor will be on.likewise for the fire sensor also. Entire operation is monitored and controlled through arm7 controller and Ethernet controller.

III. Experimental Results

Ethernet enabled digital I/O control system is designed formultiple input and output arrangements for industrial as well as non-industrial applications. The module is small, simpleand flexible which can perform different I/O operationsremotely over Ethernet.Our system can be extended for sensing malfunctioning inindustrial machines and making corrective measures in it.Wireless Ethernet enabled interface can also be developed.There is no limit for future scope in the monitoring andcontrol operation. Industrial automation is no longer limitedby the walls of the production facility. More and moreautomation is being handled via remote communication.

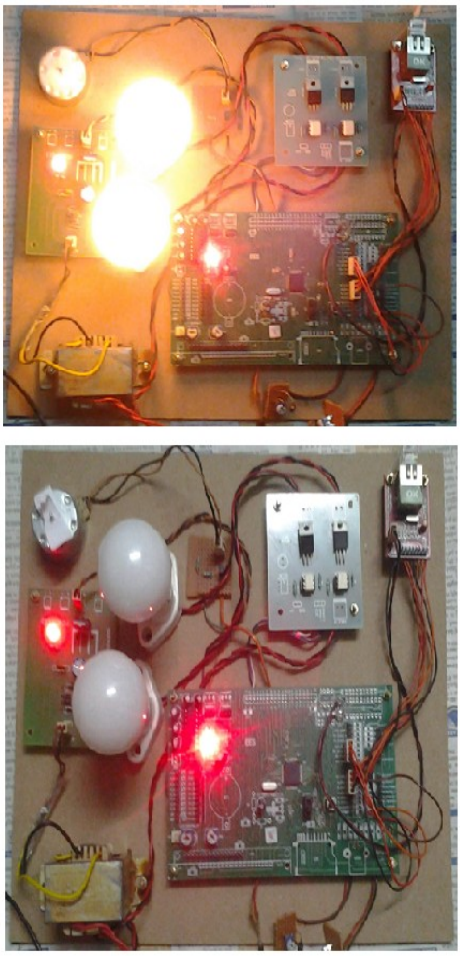

Figure: experimental set-up, hardware implementation

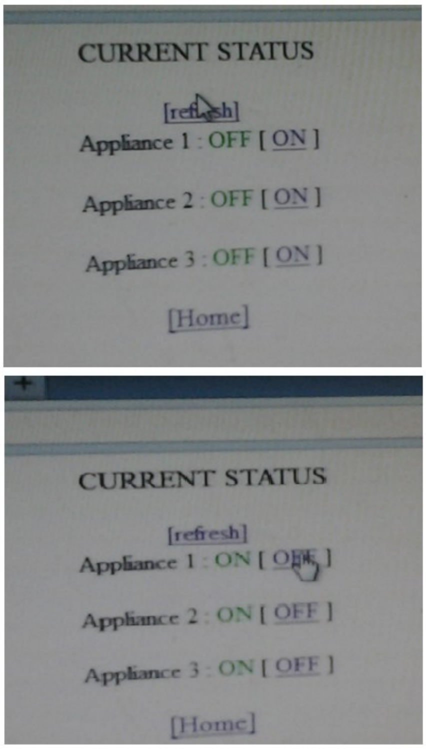

Figure: experimental status indicator.

IV. Conclusion

The project"Ethernet enabled digital i/o control in embedded systems"has been successfully designed and tested. Integrating features of all the hardware components used have developed it. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit. Secondly, using highly advanced IC's and with the help of growing technology the project has been successfully implemented.

References

[I] Jan Axelson ,"Embedded Ethernet and Internet Complete", 2003.
[2] Andrew Tanenbaum, "Computer Networks", 4th Edition, 2003, pp. 271-292.
[3] www.microchip.com.