International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 9 (2015) pp. 24729-24740

http://www.ripublication.com

Web Graphical User Interface for FPGA Modules using System on Chip

S S Sai Kiran[1] and Dr.Jayanta Biswas[2]

M.Tech VLSI Design, Associate Professor,
School of Computing, Sastra University, Thanjavur, 613401,India
sssk.025vlsi@gmail.com jayantabiswas@ict.sastra.edu

Abstract

Increasing trend of developing complex systems is observed for delivering high end lifestyle products. These systems use FPGA modules and it is desirable to have web enabled remote Graphical User Interface (GUI) for these complex FPGA Modules. This proposal develops web GUI using ALTERA *Nios Fast Core processor* in an efficient manner using System on Chip (SoC) mode. Existing proposals in the literature used SoC for this purpose. However, these proposals either port the webserver on the embedded software running in SoC mode or use CGI script to implement the webserver logic and these proposals are inefficient with respect to SoC performance. The current proposal uses socket interface to provide network connectivity between the host system and the embedded software system running in SoC mode. Socket Protocol provides flexibility to implement linux/windows as host and the minimum server functionality is implemented on the *Nios-f Processor*. The HTTP browser writes the configuration file, data is read from the file and whenever the data is modified client (host) transfers the data using socket protocols in an efficient manner without closing the file. The precision of execution is verified through the HTTP page with*set* operation which sets input value to leds on the FPGA Board and with *get* operation which reads displayed led data to the HTTP page in windows/linux host system.

I. INTRODUCTION

There is a need for complex lifestyle products in FPGAs which provides flexibility and integration for designing web based applications in the field of VLSI and embedded systems domain. This leads to the development of high end products.

The high end products are implemented using FPGA modules and user expects to use web GUI for remote access. To support this programmability, FPGA

devices contain three types of resources i.e. logic block, I/O block, and programmable interconnection.

In existing systems, CGI script and webserver are used for developing web applications. In case of webserver, cache is required which increases the load on embedded hardware. In CGI script the interface is implemented through file operations, which provides less flexibility and poor performance for ALTERA DE2 Education board without having memory management unit (MMU). The file operations are inefficient for nommu processors. To overcome this problems, Web GUI is implemented using SoC in an efficient manner in this proposal.

Web browsers are used to transfer information to the embedded systems to configure and to control the device. The Devices which are enabled with the web interface uses the HTTP standard protocol to transmit web data to the device and to get the status back from the device [3].

The design of a modern System-on-Chip (SoC) is a Complex task involving a range of skills and a deep Analysis of a hi-erarchy of perspectives on design, from processor architecture down to signal integrity. SoC which is used in this approach mainly consists of a Nios II processor on ALTERA DE-2 Education Board with the avalon bus interface and socket interface is used for communication between peripherals. The custom logic is interfaced to the processor by providing switches as inputs and leds as outputs. Fig.1 describes the block diagram of SoC Architecture.

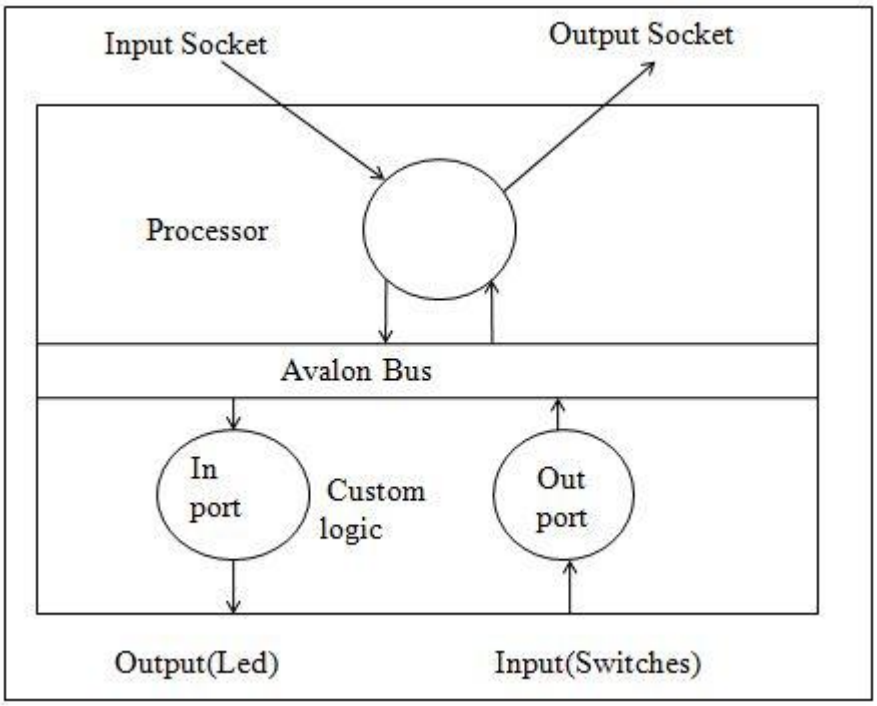

Fig. 1. Block Diagram of SoC Architecture

The aim of this work is to implement the webserver using a simple HTTP page and socket interface is used between client and server. In order to implement socket interface, a TCP/IP protocol is used which enables applications to communicate with each other. TCP sends data that appears to be transmitted in a character-by-character fashion, rather than as discrete packets.

In section I, we describe the need for high end products in FPGAs using system on chip and the technology that has provided support for implementation of web GUI. Section II, a brief description of implemented system using Nios II fast core processor, avalon bus interface, socket interface, SoPC builder and uClinux is provided. Section III, describes the architecture of proposed approach and implementation of web graphical user interface. Section IV describes the implementation setup andmethodology and Section V concludes the paper.

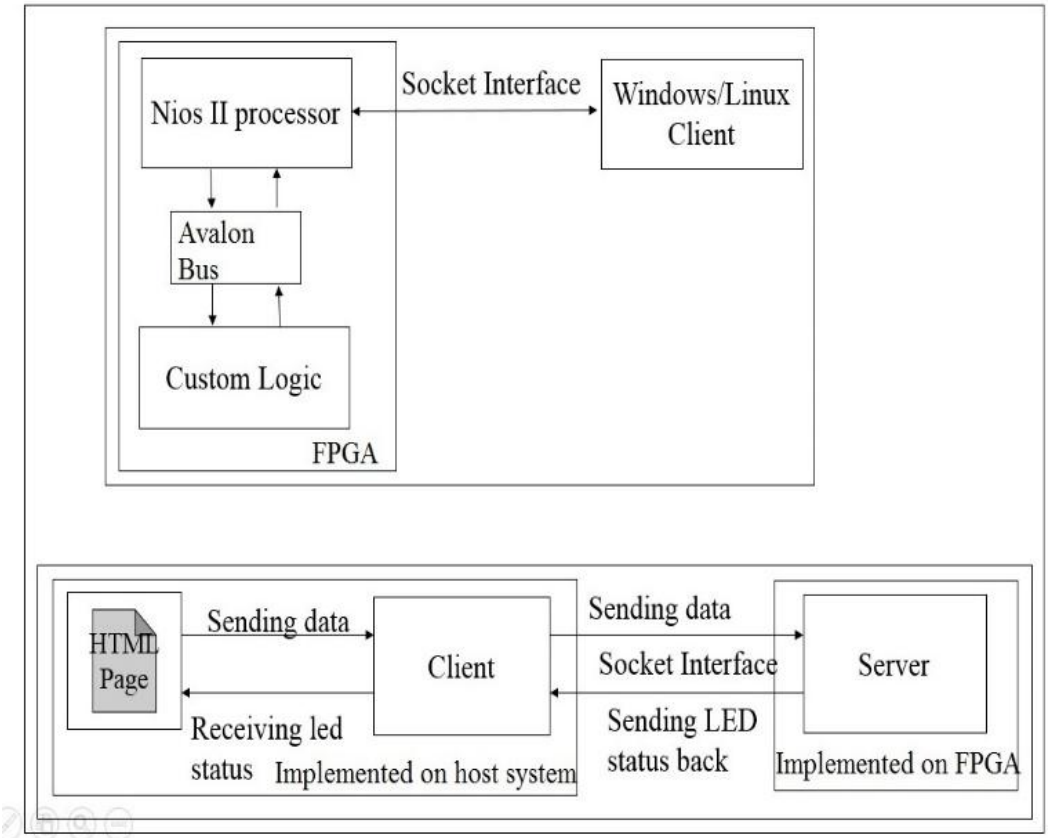

Fig. 2. General scheme of Implemented system

II. SYSTEM DESCRIPTION

A. Nios II fast core processor

A system is implemented using Nios II fast core processor, avalon bus interface, socket interface, SoPC builder and uClinux as shown in Fig.2

The Nios II family of embedded processor features a general Purpose RISC CPU architecture designed to address a widerange of embedded applications. The Nios II family consists of three cores- fast (Nios II /f), economy (Nios II /e), standard (Nios II/s) [4]. In this project, *Nios II fast core processor* is used. The soft core nature of the Nios II processor permits the system designer specify and generate a custom Nios II core that is targeted to any ALTERA FPGA family.

Many Architecture features of *Nios* like data path width, register file size, cache size, setup and hold parameters and custom instructions provides flexibility to the users to easily customize the processor to communicate with internal logic in FPGA and within the hardware level to perform specific applications [2].

B. Avalon Bus Interface

Avalon Interfaces are used to simplify the system design by allowing user to simply connect the different components in ALTERA FPGA. The avalon Interface family provides interfaces for numerous operations such as streaming of high-speed data, reading/writing operations between registers and controlling the peripheral devices.

In this paper, avalon memory mapped Interface is taken into account. avalon memory mapped interface (avalon-MM) is an address based read/write interfacetypically used for master slave connections [8]. For instance, SRAM interfaces have a fixed read/write transfers.

C. Socket Interface

Socket is an abstraction for network communication and it is defined by a socket descriptor. TCP/IP protocols define a communication endpoint which contains an IP address and a protocol host number. Active sockets are used by clients to initiate the connection. Sockets provide flexibility in windows/linuxplatform.

TCP/IP protocols use a single address representation with address family denoted by AF-INET. TCP/IP allows end to end connectivity by specifying how the data is converted to packets. TCP/IP assigns the source and destination addresses. TCP/IP provides a routing through the channel for the transmission of the packets and packets reception at the end. The functionality is organized into four abstraction layers in order to sort all related protocols according to the mode of networking used.

D. SoPC Builder

To implement a useful system, it is necessary to add functional units like input/output interfaces, memories, timers and communication interfaces. Altera SoPC Builder is a Computer aided design (CAD) software which is used for implementing the system on programmable chip (SoPC).

The basic components used to boot up the image are Nios

II/f core processor, SDRAM controller for memory interface, Timer and JTAG UART. The Timer core with avalon interface is a 32-bit timer for avalon based processor systems. It is used to control the start, stop and reset the timer operations. JTAG UART provides a peripheral circuitry for communication interfacing with the host systems. Ethernet provides a wired communication between host computer and DE-2 board. DM9000A component in SoPC Builder provides Ethernet support.

The input and output peripherals are assigned to switches and leds respectively. To interface with the custom logic, the base address of both switches and leds are defined in the given socket server program. The PIOs for both input and output are taken as 8-bit wide. The Avalon bus bridge acts as a FIFO between two interface buses. The Nios II processor core operate at a certain frequency compared to other peripherals. The Ethernet and PIOs operate at a different frequency. The Avalon bus bridge send and receive data based on the frequency of the peripheral as shown in Fig.3.

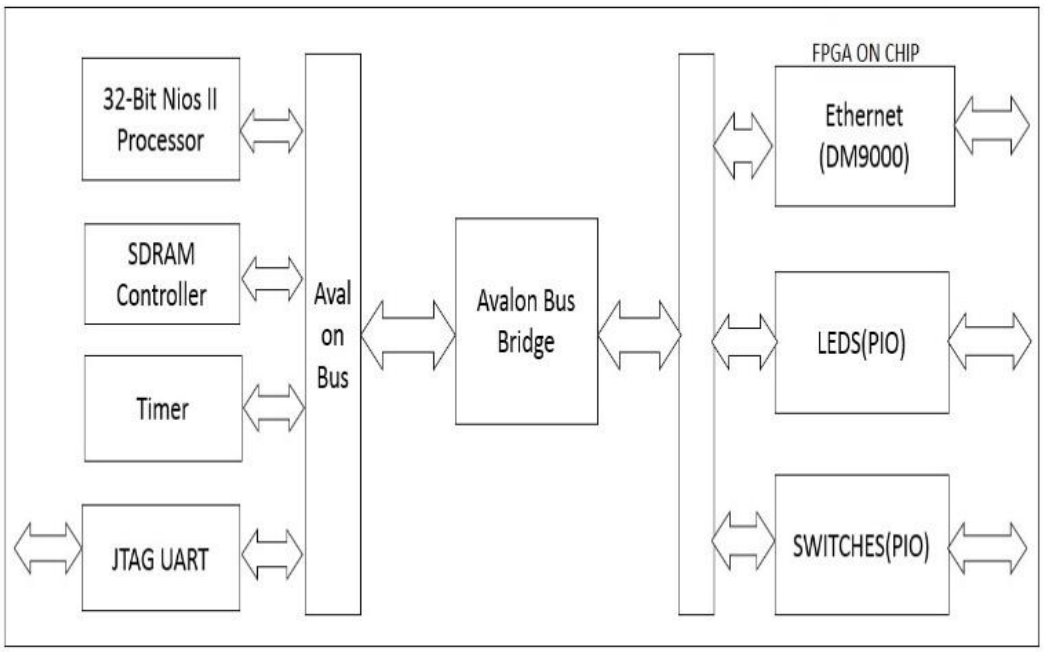

Fig. 3. System Implementation using SoPC Builder

E. uClinux (Embedded Linux)

uClinux is a fork of the linux kernel for microcontrollers without a memory management unit. uClinux is used as anoperating system which includes linux kernel releases of 2.0 and 2.6 along with the collection of user applications, libraries, driver interfaces and tool chains [9]. In this paper, uClinux is used for Nios II processors without memory management unit.

The main need for uClinux is to develop a embedded application which is user defined and the user logic can be modified at any time. The configuration of kernel settings according to the user perspective and creating the application interface with respect to the user logic along with the processor specifications obtained from the Altera SOPC Builder are necessary for full kernel compilation.

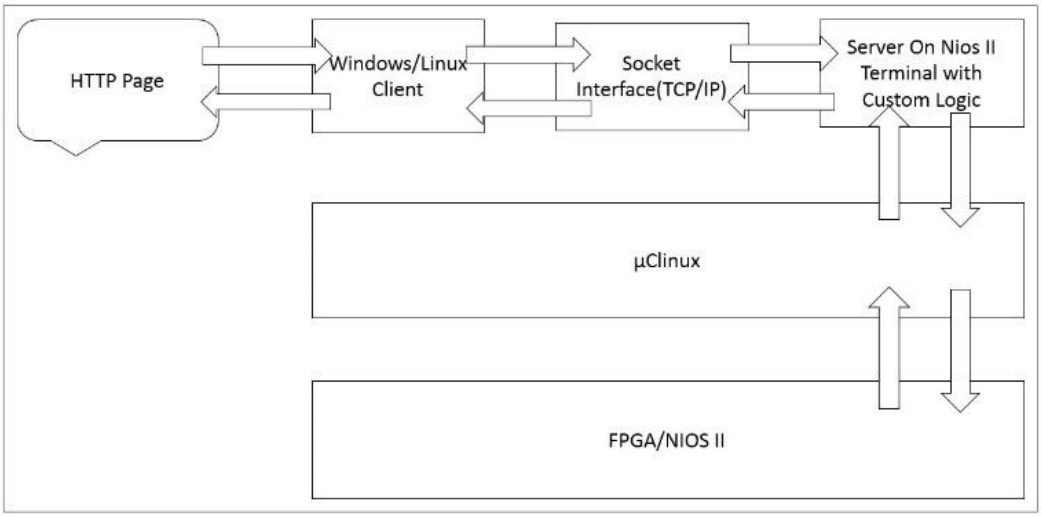

Fig. 4. Implemented system Architecture

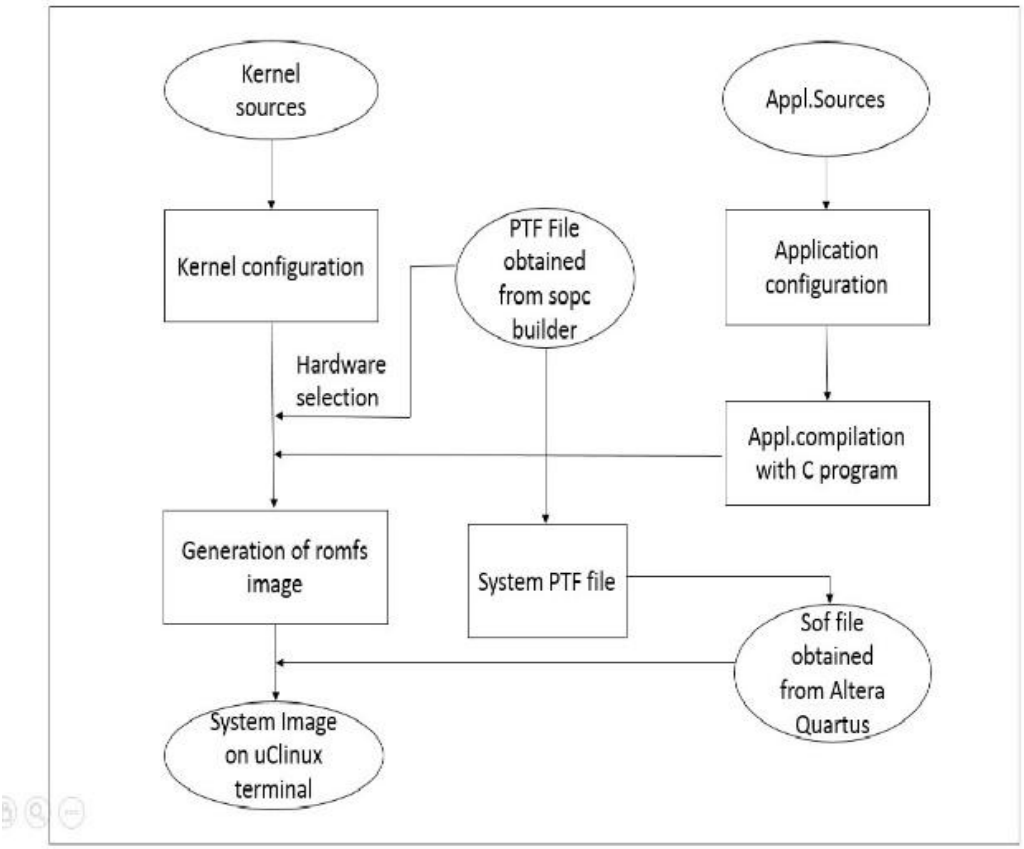

Fig. 5. Flowchart for obtaining the System Image

The zImage is saved in the uClinux directory, when the compilation is completed. The zImage is copied to the windows directory and sram output file obtained by Altera compilation is copied to the same directory [10]. Nios II command shell in windows is used for booting the image and running the application along with the image. Ethernet/wireless configuration between host computer and DE-2 board is achieved by configuring a unused mac and IP address.

III. PROPOSED APPROACH

In existing systems, CGI script is used for web application interface [1].The file operations for read and write must be performed everytime whenever the webpage access the CGI script and this increases the load on Nios II processor. In webserver, cache is required on processor and this increases the load on embedded hardware.

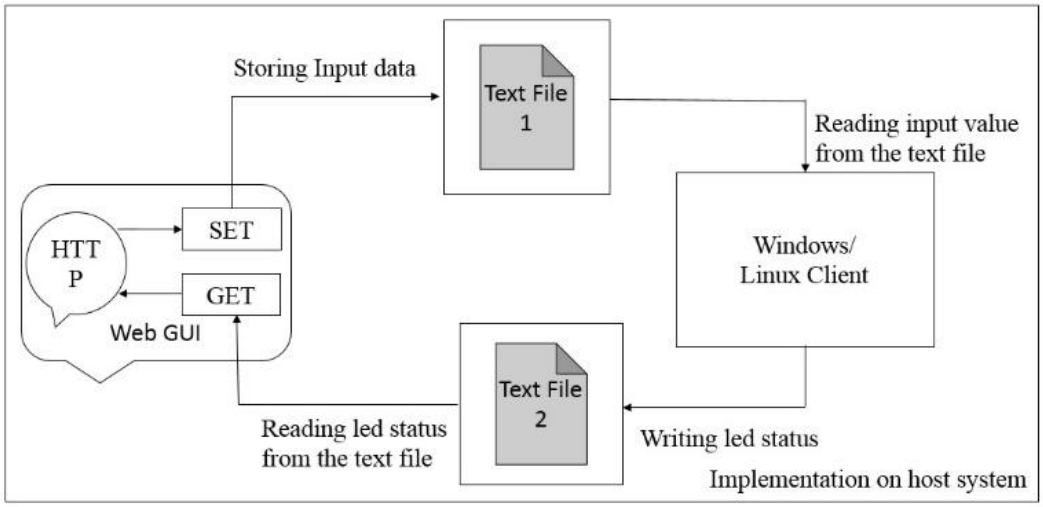

Fig. 6 .Implementation on host system

The Implemented System consists of a HTTP page with set and get operations which are used for writing data to client and reading led status back to client (host) as shown in Fig.4. The integration between HTTP page and client is implemented by writing data from the HTTP page to the text file. Client reads the data from text file and client never closes the server end file as shown in Fig.5. The input data is transmitted over the socket interface to the server on Nios II processor.

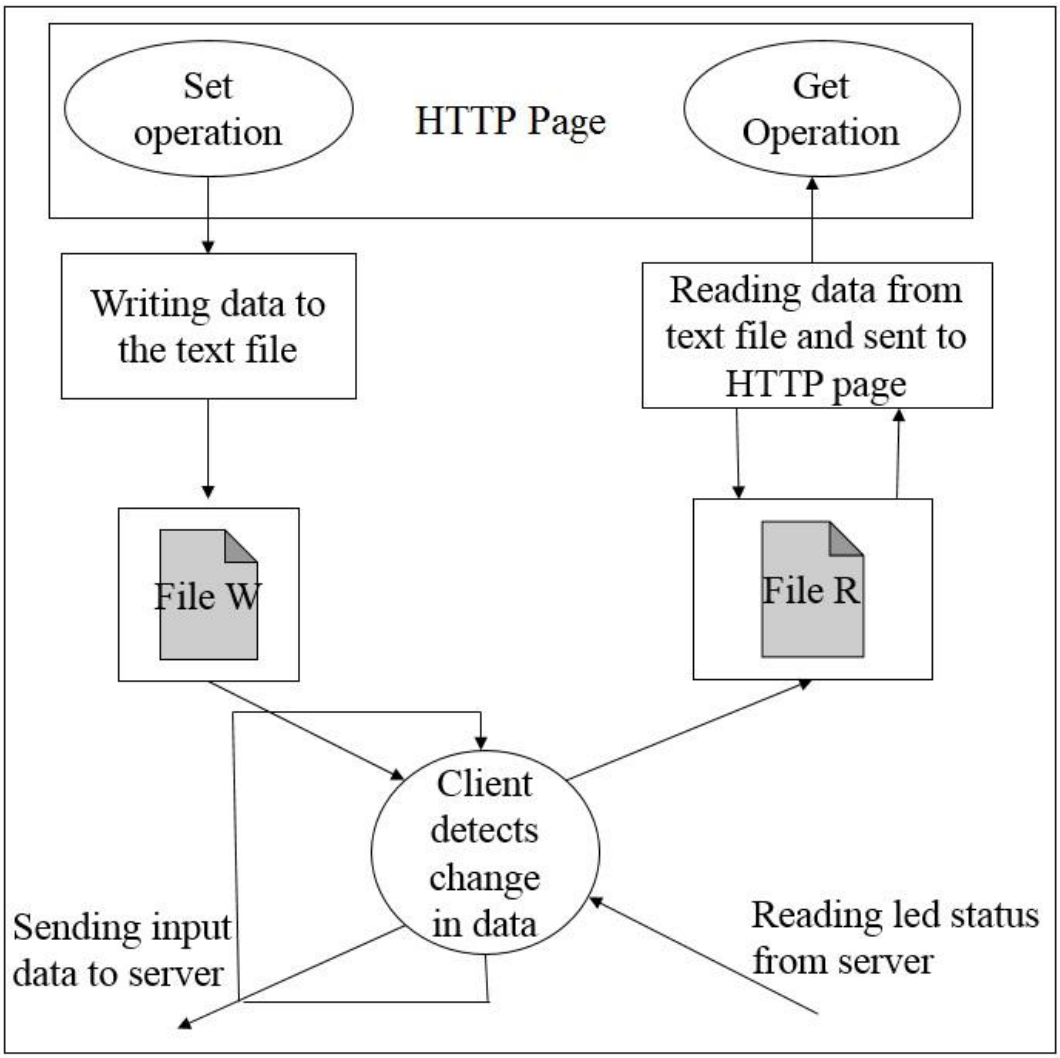

Fig. 7 File operations Implemented on the host system

The C application implemented for socket server functionality interacts with the custom logic on Nios II processor and uClinux is used as operating system. The socket server program receives input data and passes the data to custom logic over avalon bus. The data is displayed on the led on ALTERA DE2 Education board. C application reads the led data and passes the data to client (host).The client receives led data and stored in a text file. When the HTTP page is displayed on the browser, browser retrieves the data from the file and it is displayed on the HTTP page. The flowchart for booting the image using sram output file and *ptf* file obtained from SoPC builder is shown in Fig.5

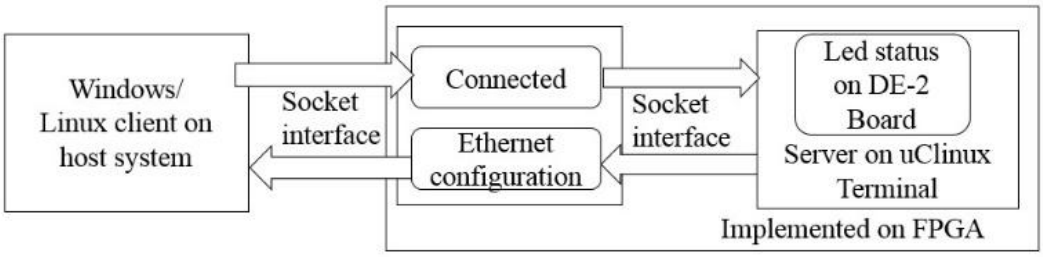

Fig. 8 .Implementation on server

The main advantage of this approach is implementation of bare minimum server functionality on Nios II processor and this approach provides flexibility because of socket interface. Socket protocol and HTTP browser is supported on windows/linux (host) operating systems.

In Common Gateway Interface, each URL is mapped to CGI java script. In case of embedded systems, the script is implemented by means of a function call to an embedded application. CGI script sends function call to a static HTML page, the parameters are appended to the HTML page by using java script [5] and input and output is achieved through files.

In case of web server, the contents of webpage and the software programs for different protocols to support HTTP is downloaded to the flash memory using flash programmer in Nios II development board. The TCP/IP stack which includes many protocols in TCP/IP suite must be taken into consideration.

In order to integrate the webGUI capability to the embedded applications, CGI (common Gateway Interface) and webserver approaches are used in the literature. CGI is low overhead in comparison to webserver but less flexible in design [5]. In case of webserver implementation, puts high overhead on embedded hardware [3]. In the proposed system, a simple HTTP page,webbrowser, socket interface and client/server communication on the host system through a TCP/IP protocol provide ultimate resource optimization for web GUI functionality and this approach is device independent.

Reading from file and writing to file operations between HTTP page and client is done as shown in Fig.6. Whenever the client detects the modified data from file W, The data is transferred to server over socket interface.File R is chosen to read the led status obtained from server is shown in Fig.7. The implementation on server and how the ethernet is configured between host system and DE-2 Board is shown in Fig.8.

IV. IMPLEMENTATION SETUP AND RESULTS

Web GUI is implemented between host system and Altera nommu Nios II processor using socket interface. uClinux uses a rich set of driver during system on chip boot mode. The passing data from HTTP page to a running C application in a host system is only possible in objective C which is part of development environment in Mac

operating system [11]. The total implementation undergoes three steps: *set* operation, *read* operation and *get* operation.

1) In set operation input data (85) from HTTP page is saved into text file. Client reads the input data (85) from text file and passes that data to socket server (ALTERA DE2 FPGA board) as shown in Fig.9.
2) Server receives the data (85) and programs the leds through avalon interface. Value of 85 implies that 1st, 3rd, 5th and 7th LED is on and it is demonstrated in Fig.10. Server reads the led status value (85) as demonstrated in Fig.11.and sends led status to client.
3) Client receives led status (85) and stores in a text file and led status is displayed on HTTP page using get operation as shown in Fig.9. The terminal output of client is shown in Fig.11.The demonstration of the proposed approach between host system and ALTERA DE2 Education board

Fig. 9. Set and get operation in http page.

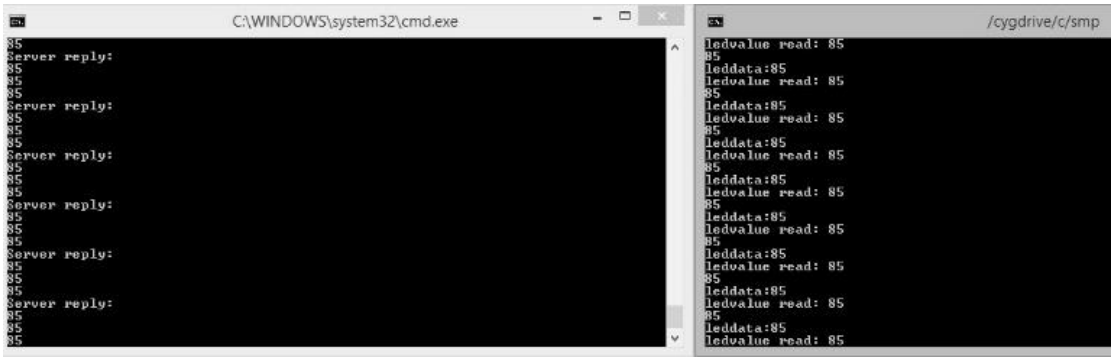

Fig. 10. Reading LED status in ALTERA DE2 FPGA board

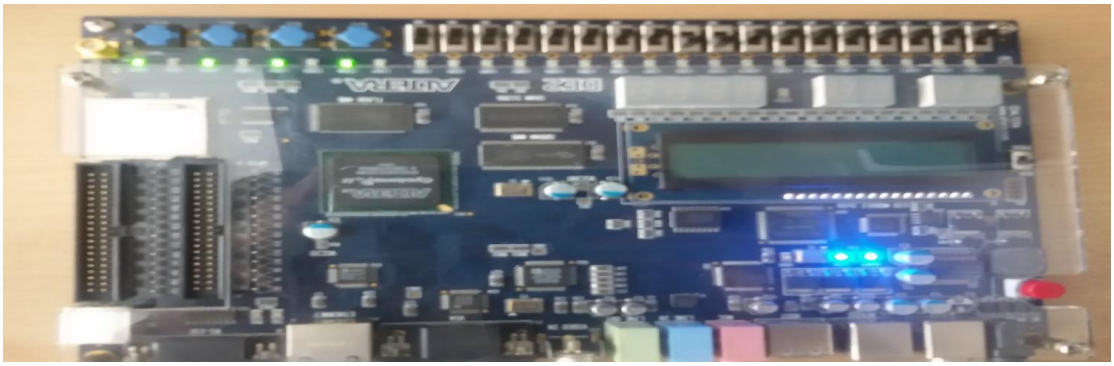

Fig. 11.Terminal output between client and server

The demonstration of the proposed approach between host system and ALTERA DE2 Education board is shown in Fig.12.

Fig. 12. Implementation Setup

V. CONCLUSION

In this paper, a WebGUI is designed with the AlTERA DE2 FPGA board using SoC environment and socket interface. The socket server application in Nios II processor and client application in host system is implemented in C language which reduces implementation complexity. The implementation may be carried out in any platform.

REFERENCES

[1] Eduardo Magdaleno 1, Manuel Rodrguez 1, Fernando Prez 2, David Hernndez 1 and Enrique Garca 1 A FPGA Embedded Web server for remote monitoring and control of Smart Sensor Networks. *sensors 2014*, 14, pp.416-430.

[2] Cheng, H.; Qin, H. A Design of IEEE 1451.2 Compliant Smart Sensor Based on the NIOS Soft-Core Processor. *In Proceedings. IEEE International Conference on Vehicular Electronics and Safety, Xian, China, 14-16 October 2005*, pp. 193-198.

[3] Agranat, I.D. Engineering web technologies for embedded applications. *IEEE Internet Comput. 1998*, 2, 40-45

[4] Joshi, N.; Dakhole, P.K.; Zode, P.P. Embedded Web Server on Nios II Embedded FPGA Platform. *Proceedings.2nd International Conference on Emerging Trends in Engineering and Technology, Nagpur, India, 16-18 December 2009*, pp. 372-377.

[5] Wang, Z.X.; Ren, X.Y. A Study on CGI of Embedded Webserver *Proceedings. International Symposium on Computer Science and Computational Technology, Shanghai, China, 20-22 December 2008; pp. 480-483.*

[6] Altera SOPC Builder datasheet, Altera Corporation, San Jose, CA , December 2010

[7] Nios II Hardware Development tutorial, Altera Corporation, San Jose, CA, May 2011

[8] Avalon Interface Specification, Altera Corporation, San Jose, CA, August 2010

[9] www.uClinux.org, uClinux resource website

[10] wwww.alteraforum.com, forum for issues and solutions for altera products.

[11] www.apple.com, Apple Inc. Consumer Electronics company