NIOS II Processor Based Customized Soft Microcontroller Unit for Web Server Implementation

Manikandan. K[1], Purushothaman. V[2]

[1]Student, M.E. Applied Electronics, Thiruvalluvar College of Engineering and Technology, Vandavasi, India
[2]Student, M.E. Embedded System Technologies, Krishnasamy College of Engineering and Technology, Cuddalore, India

Abstract: ***It is an undeniable fact that the embedded system domain is growing rapidly and almost on a daily basis there are new applications. Various microcontrollers' manufacturers have come up with various microcontroller families to cater to these requirements. But the problem with these is that of interoperability i.e. the architecture, the programming tools, languages, debugging and testing tools varies vastly and hence becomes a major limiting factor. Using the NIOS II software processor platform the user can select every aspect of a microcontroller unit from the ALU to the memories and even the peripherals that are required. Thus any user is free to design their own controller depending on the specific application. In this project the whole cycle of design and construction of a soft MCU using the NIOS II processor will be followed to implement a customized MCU. This implementation will be used for industrial control and measurement using multi channel PWM. Embedded Webserver is implemented using lab VIEW, which will enable the control through internet using graphical programming.***

Keywords: NIOS II, Webserver, MCU unit

1. Introduction

This paper aims at demonstrating systems which could be a solution to these problems of compatibility and availability. All of these problems would be gone if the user has a tool using which they could make controllers with the features they want with as much memory and peripherals they want. NIOS II is a processor platform which can be customized by a user to their requirements. Using the NIOS II software processor platform the user can select every aspect of a MCU (microcontroller unit) from the ALU to the memories and even the peripherals that are required. Once the user has selected and configure the appropriate amount of features required for the application, the soft MCU can then be transferred to a appropriate FPGA to get the controller working in hardware and in real time. Thus a user is free to design their own controller depending on the specific application. In this project the whole cycle of design and construction of a soft MCU using the NIOS II processor will be followed to implement a customized MCU on a FPGA. The application that will be demonstrated using the project will be multi axis control for industrial control and measurement using multi channel PWM. For this a custom microcontroller will be designed with the required peripherals, ALU and the memories. Additionally in the paper an Embedded Webserver will be implemented, which will be controlled over the internet. This over all control will be done using LAB view graphical programming; the application could be changed for various requirements.

1.1 Project Specifications and Description

Our system NIOS II processor based customized soft MCU consist of four main parts;

1) NIOS II processor
2) Altera Cyclone FPGA
3) SOPC builder
4) Lab VIEW

NIOS II is a processor platform which can be customized by a user to their requirements. Using the NIOS II software processor platform we can select every aspect of a MCU (microcontroller unit) from the ALU to the memories and even the peripherals that are required. Once we had selected and configured the appropriate amount of features required for the application, the soft MCU can then be transferred to appropriate FPGA to get the controller working in hardware and in real time. Thus we can free to design their own controller depending on the specific application. In this project the whole cycle of design and construction of a soft MCU using the NIOS 2 processor will be followed to implement a customized MUC on a FPGA. The application that will be demonstrated using the project will be multi axis control for industrial control and measurement using multi channel PWM. For this a custom microcontroller will be designed with the required peripherals, ALU and the memories. Additionally in the project an Embedded Webserver will be implemented, which will be controlled over the internet .This over all control will be done using LAB view graphical programming.

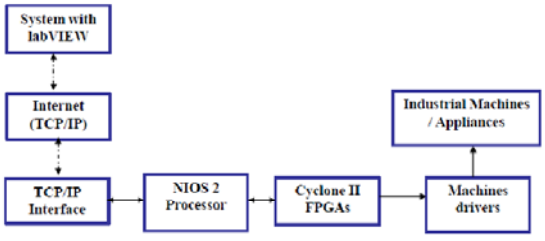

Figure 1: Block diagram

Web server has been run over the standard features of the Nios II processor system. Figure 1 shows the block diagram of Nios II processor and some added features. It is altera's sopc builder design tool responsibility to configure the processor features and generate a hardware design automatically in an FPGA. After system generation, the design has been downloaded onto FPGA board

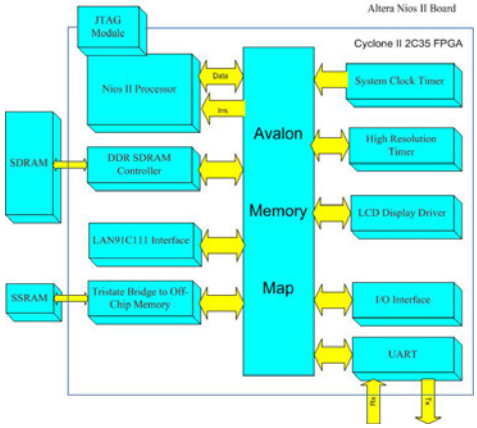

Figure 2: Nios II and peripherals block diagrams

The Nios II/s standard core is designed for small size while maintaining performance. The Nios II processor is a general-purpose RISC processor core, providing:

- Full 32-bit instruction set, data path, and address space.
- 32 general-purpose registers.
- Optional shadow register sets.
- 32 interrupt sources.
- External interrupt controller interface for more interrupt sources.
- Single-instructions 32 X 32 multiply and divide producing a 32-bit result.
- Dedicated instructions for computing 64-bit and 128-bit products of multiplication.
- Floating-point instructions for single-precision floating-point operations.
- Single-instruction barrel shifter.
- Access to a variety of on-chip peripherals, and interfaces to off-chip memories and peripherals.
- Hardware-assisted debug module enabling processor start, stop, step, and trace under control of the Nios II software development tools.
- Optional memory management unit (MMU) to support operating systems that require MMUs
- Optional memory protection unit (MPU)
- Software development environment based on the GNU
- Integration with Altera's Signal Tap II Embedded Logic Analyzer, enabling real-time analysis of instructions and data along with other signals in the FPGA design.
- Instruction set architecture (ISA) compatible across all Nios II processor systems.
- Performance up to 250 DMIPS.

A Nios II processor system is equivalent to a microcontroller or "computer on a chip" that includes a processor and a combination of peripherals and memory on a single chip. A Nios II processor system consists of a Nios II processor core, a set of on-chip peripherals, on-chip memory, and interfaces to off-chip memory, all implemented on a single Altera device. Like a microcontroller family, all Nios II processor systems use a consistent instruction set and programming model Figure 2 shows the NIOS II processor and its peripherals are built using SOPC builder, which are then implemented on the FPGA.

Avalon Interface: Simplify system design by allowing easily connection for several components in an FPGA. The Avalon interface family defines interfaces for use in both high-speed streaming and memory-mapped applications.

System Clock Timer: An interval timer for Avalon-based processor systems, it provides 32-bit and 64-bit counters, Controls to start, stop, and reset the timer, two count modes: count down once and continuous count-down., Count-down period register, Option to enable or disable the interrupt request (IRQ) when timer reaches zero, Optional watchdog timer feature that resets the system if timer ever reaches zero, Optional periodic pulse generator feature that outputs a pulse when timer reaches zero, Compatible with 32-bit and 16-bit processors.

LAN91C111: is designed to facilitate the implementation of a third generation of fast Ethernet connectivity solutions for embedded applications. For the third generation of products, flexibility and integration dominate the design requirements. The LAN91C111 is a mixed signal Analog/Digital device that implements the MAC and PHY portion of the CSMA/CD protocol at 10 and 100 Mbps.

Optrex 16207 LCD: controller core with Avalon Interface (LCD controller core) provides the hardware interface and software driver required for a Nios II processor to display characters on an Optrex 16207 (or equivalent) 162-character LCD panel. Device drivers are provided in the HAL system library for the Nios II processor. Nios II programs access the LCD controller as a character mode device using ANSI C standard library routines.

UART: The UART core with Avalon interface implements a method to communicate serial character streams between an embedded system on an Altera FPGA and an external device. The core implements the RS-232 protocol timing, and provides adjustable baud rate, parity, stop, and data bits, and optional RTS/CTS flow control signals. The feature set is configurable, allowing designers to implement just the necessary functionality for a given system.

I/O Interface: The parallel input/output (PIO) core with Avalon interface provides a memory-mapped interface

between an Avalon Memory-Mapped (Avalon-MM) slave port and general-purpose I/O ports. The I/O ports connect either to on-chip user logic, or to I/O pins that connect to devices external to the FPGA. The PIO core provides easy I/O access to user logic or external devices in situations where a"bit banging" approach is sufficient.

Some example uses are:

- Controlling LEDs.
- Acquiring data from switches.
- Controlling display devices.
- Configuring and communicating with off-chip devices, such as application-specific standard products (ASSP).

Memory-Mapped Tri-state Interfaces: Avalon-MM tristate slave interfaces allow Avalon-MM masters to drive off-chip devices. The interface allows data and address pins to be shared across multiple tri-state devices. Sharing is valuable in systems that have multiple external memory devices and limited pins.

DDR SDRAM Controller MegaCore Function: The Altera DDR and DDR2 SDRAM Controller Compiler comprise the DDR SDRAM Controller MegaCore function and the DDR2 SDRAM Controller MegaCore function. The MegaCore functions provide simplified interfaces to industry-standard DDR SDRAM and DDR2 SDRAM devices. The DDR and DDR2 SDRAM Controllers handle the complex aspects of using DDR or DDR2 SDRAM-initializing the memory devices, managing SDRAM banks, and keeping the devices refreshed at appropriate intervals. The DDR and DDR2 SDRAM Controllers translate read and write requests from the local interface into all the necessary SDRAM command signals.

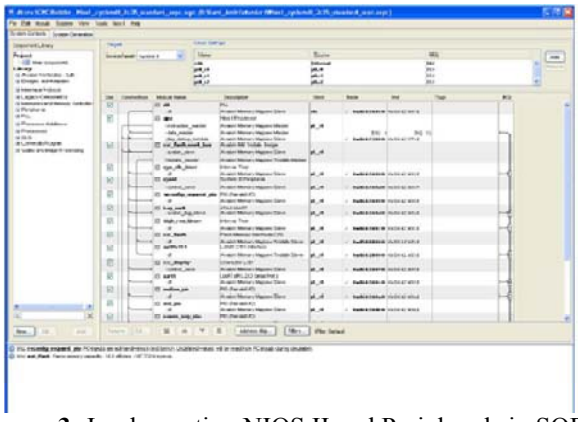

Figure 3: Implementing NIOS II and Peripherals in SOPC Builder

Lab VIEW: In Programmers develop software applications every day in order to increase efficiency and productivity in various situations. LabVIEW, as a programming language, is a powerful tool that can be used to help achieve these goals. LabVIEW (Laboratory Virtual Instrument Engineering Workbench) is a graphically-based programming language developed by National Instruments. Its graphical nature makes it ideal for test and measurement (T&M), automation, instrument control, data acquisition, and data analysis applications. This results in significant productivity improvements over conventional programming languages. National Instruments focuses on products for T&M, giving them a good insight into developing LabVIEW.

Virtual Instruments: A Virtual Instrument (VI) is a LabVIEW programming element. A VI consists of a front panel, block diagram, and an icon that represents the program. The front panel is used to display controls and indicators for the user, while the block diagram contains the code for the VI. The icon, which is a visual representation of the VI, has connectors for program inputs and outputs. Programming languages such as C and BASIC use functions and subroutines as programming elements. Lab VIEW uses the VI. The front panel of a VI handles the function inputs and outputs, and the code diagram performs the work of the VI. Multiple VIs can be used to create large-scale applications; in fact, large scale applications may have several hundred VIs. A VI may be used as the user interface or as a subroutine in an application. User interface elements such as graphs are drag and- drop easy in LabVIEW.

2. TCP/IP Communication in Lab VIEW

Internet Protocol (IP), User Datagram Protocol (UDP), and Transmission Control Protocol (TCP) are the basic tools for network communication. The name TCP/IP comes from two of the best-known protocols of the internet protocol suite, the Transmission Control Protocol and the Internet Protocol. With TCP/IP you can communicate over single networks or interconnected networks (Internet). TCP/IP communication provides a simple user interface that conceals the complexities of ensuring reliable network communications. Use the TCP/IP functions located on the Functions»Communication»TCP palette for TCP communication in Lab VIEW. As with DAQ, instrument, and File I/O communication, the process involves opening the connection, reading and writing the information, and closing the connection. With most I/O communication, the processor is always the client that initiates a connection to the disk drive server, the external instrument server, or the DAQ board server. With TCP/IP connections, a computer can function either as the client or the server. The following block diagram represents a client application that initiates a connection to a remote server with TCP Open Connection. The server, or daemon, listens for remote connections and responds appropriately. LabVIEW users can develop custom applications for TCP/IP communication. The programme is responsible for developing both the client and the server. Because anyone can initiate a connection to a server, you might want server access control. The following block diagram shows how the server uses the remote

address output value of the TCP Listen VI to determine whether a remote client has permission to access the server.

3. Conclusion

Implementing NIOS II processor on FPGA allows more flexibility on both hardware and software. Such flexibility on hardware side is represented by adding different type of CPUs, peripherals, interfaces, and controllers without changing the FPGA. On the software side, Altera provides network stack that works with NIOS II processor to be targeted to different types of networking application Therefore, embedded NIOS II processor provides reconfigurable environment, with lower coast, lower power consumption, and high performance.

4. Acknowledgment

We would like to thank Altera Corporation for offering programmable logic solutions: FPGAs, CLPDs, and ASICs with their software tools, and intellectual property that allows us to access a wide variety of IP blocks of differing size and complexity. Which help designers around the world to innovate new systems rapidly and cost effectively Furthermore, we thank the department of electrical engineering and embedded system technology, Krishnasamy college of engineering technology.

References

[1] Altera Corporation,"Nios II Software Developer's Handbook, Chapter 11: Ethernet and the NicheStack TCP/IP Stack - Nios II Edition, July 2010.
[2] N. Joshi, P. Dakhole, P. Zode, "Embedded Web Server on Nios II Embedded FPGA Platform.", In Second International Conference on Emerging Trends in Engineering and Technology.
[3] Website, HTTP://www.altera.com.
[4] http://www.iniche.com/nichestack.php.
[5] www.Altera.com,"Nios II Processor Reference Handbook", 2010
[6] www.Altera.com,"Avalon Interface Specifications", 2010
[7] www.Altera.com,"Quartus II Handbook Version 9.1 Volume 5: Embedded Peripherals", 2009
[8] www.Altera.com,"Embedded Peripherals IP User Guide", 2010
[9] www.Altera.com,"DDR and DDR2 SDRAM Controller Compiler User Guide"', 2009
[10] T. S. Hall, J. Hamblen,"System-on-a-Programmable-Chip Development Platforms in the Classroom." IEEE transactions on Education, Volume 47, issue 4, Nov. 2004,
[11] www.ni.com/support Knowledgebase – searchable database of tips, common questions, and LabVIEW Measurements Manual.