Implementation of Wireless Sensor Network using Microblaze and Picoblaze Processors

Seema M. Borawake
PG student: E & TC,
SKNCOE Pune, Maharashtra, India
E-mail: seema.borawake@gmail.com

Prof. P. G. Chilveri
Asst. Prof.: E & TC,
SKNCOE Pune, Maharashtra, India
e-mail: pgchilveri@gmail.com

Abstract— **A wireless sensor network has a substitute for traditional wired monitoring systems. Wireless monitoring systems helps to reduce the complex wiring and power consumption of the system in tern installation cost of the system. It increases the reliability of the system for long term applications such as structural health monitoring, Health care, control and security etc. Wireless sensor network (WSN) consist spatially distributed sensors, computing and communicating elements. This system is implemented using Field Programmable Gate Array (FPGA). It has master-slave architecture. Master is designed with Picoblaze and slave with Microblaze soft core processor. These two softcore processors are optimized for Xilinx FPGA and implemented using ISE and EDK software suite. To implement WSN on FPGA is the challenging task of the system. This will overcome precincts of traditional processor such as reliability, flexibility and very important aspect durability with high processing speed. It reduced the processing time to nanoseconds. Wireless sensor nodes are communicating using ZigBee (IEEE 802.15.4 standard). It is RF transceiver operate in 2.4-2.843 GHz ISM (industrial, scientific, and medical) band.**

Key words – Field Programmable Gate Arrays, Wireless Sensor Network, Softcore processor, ZigBee.

I. INTRODUCTION

The locations where manual monitoring is infeasible, Wireless Sensor Networks (WSN) will be the great solution. Conventional monitoring system used for structural and health monitoring applications involve significant wiring (fiber-optic cables or copper cables) and centralized data acquisition systems. This might lead to high installation cost and noncontiguous service due to complexity in wired network [1]. The complexity will go on increasing as the number of nodes and the distance between two nodes of a network increases. Therefore recent years have an increasing interest in the sensors technology, based on WSN. WSN is already implemented using different hardcore processors causes slow response with dense network. Due to all these reasons power consumption of the system is more. Thus we switched to softcore processor. A softcore processor is a customized processor with a software code using C, C++, and VHDL etc. on hardware platform. It is an emerging concept in digital logic design such as Field Programmable Gate Array (FPGA). It allows us to change the logic of the system with intelligent coding. Xilinx Embedded development kit (EDK) and ISE (Xilinx project navigator) helps to complete the task of softcore processors. User has choice of Picoblaze or Microblaze softcore processors which are intended for Xilinx, Altera FPGA platforms. Spartan 3 complete series, Vertex5 and 6 are examples of Xilinx FPGA hardware platforms. It also endow a system with reconfigure ability which reduces power consumption, number of resources required for particular application and size of the system. It allows using same core with simple modification if the application is to be changed; also it provides compatibility with other processing unit for some extraordinary applications. It will overcome the limitations of traditional monitoring system [5]. In the countries like United States, France, Germany the wireless sensor network (WSN) already deployed in the huge structure of metallic bridges, towers, dams also in automobile plants, healthcare etc. It is playing a vital role in safety of lives. Also reduce maintenance & repair cost of the civil structure. In turn trim down national economic downturn.

The performance of the module presented in a paper is tested with the load on a prototype metallic structure using a load cell. With variation in the load the change in voltage is noted and the data is send to master module using wireless communication. When the load limit is crossed an error message is displayed at the master.

II. BACKGROUND

Interest in wireless sensors networks was initially motivated by their low-cost. The suppression of extensive lengths of coaxial wires, in a structure results in low installation costs and an uninterrupted service. Also it has less complexity and capable to support robust mesh network.WSN is having various components such as sensors, transmitters, receivers, processing units at both ends. These are connected in different topologies.

In India such technology is not adopted at social level. Whereas in other countries research is done with different technologies for structural health monitoring (SHM). Let's have a look on some examples Ping Wang,Yan Yan, Gui Yun Tian, Omer Bouzid, Zhiguo Ding write a paper on investigation of wireless sensor network for structural health monitoring[1] Dae-Man Han and Jae-Hyun Lim design a home security system using WSN with the help of 8051 MCU[6]. Prayati, Tyler Harms, Sahra Sedigh, and Filippo Bastianini implement a vibration sensor for metallic bridge structure using mote based sensor supported by

978-1-4799-3070-8/14 $31.00 © 2014 IEEE
DOI 10.1109/CSNT.2014.216

TinyOS [3]. Also systems are implemented with Atmel AVR-8 bit, Intel PXA271, TI-MSP430, ATMEGA-1281 processor.

Some other application using softcore processor is: Katteb, Aiyappa Ramesh and Azzawi design a WSN using softcore processor by processing an image of the metallic bridge structure .They used a SPIN (Sensor Protocol for Information via Negotiation)[5]. EDK Implemented Temperature Controller by Ioan Lie, he used Microblaze for this application [6]. Input/output peripheral devices Control through serial Communication using Microblaze Processor is designed by B.Muralikrishna, K.Gnana Deepika [4]. But WSN on FPGA platform for different application is currently under research.

Previous WSN implemented with above mentioned techniques used hardcore processors faces three main drawbacks: 1. lack of flexibility in terms of development due to low resource 2. Rigid and large architecture hence require more size on chip 3. Low capability of expansion and hence poor performance when network is more dense [7]. Theses drawbacks are eliminate using softcore processor. As it are having features like

- It is preeminent hardware and software configuration environment hence number of nodes can easily interfaced to the processor with few changes in the code.
- As it is reconfigurable number of resources and external hardware required is less.
- It is compatible to latest versions of Xilinx FPGA devices hence the capability of the system can be enhanced for dense network.

In this system the sensor module is implemented with Microblaze. Embedded Development Kit (EDK) supports soft-core processor Microblaze and the hard-core embedded processor PowerPC. The silent features of this Microblaze soft processor core are: It is a32-bit RISC microprocessor having separate instruction and data buses [8]. The processor functional unit consists of basic sequential and combinational elements .As the multiple instructions are executing at the same time additional registers are used. Program counter, Instruction program counter, register files, data register, instruction register ALU out register etc. [17]. The model for the soft core processor architecture is developed using EDK 10.1 software design tools .But for Microblaze some tools from ISE also required to be installed. Then with the help of EDK tool known as Xilinx Platform Studio (XPS) is used to add an embedded processor to ISE project. The code written to control the operation of the Microblaze processor must be done in C/C++ language. C/C++ is the format that the Xilinx EDK accepts. EDK is having in-built C-compiler which generates the .vhd file for the FPGA platform.

In this work Master module implemented with Picoblaze processor. Xilinx ISE project navigator helps to develop Picoblaze softcore processor. It is an 8-bit RISC microcontroller optimized for Xilinx Spartan and Vertex families. It is provided with the KCPSM3.exe as synthesizable code [9]. The pBlazIDE assembler has a code import function which cans reads the KCPSM3 syntax [ug129].It just required 96 slices on FPGA target device and 0.5-1block RAM. Instructions are stored in BRAM (block RAM) and they are fetched and executed by KCPSM3 as per the application. New source file are created and stored in BRAM together with KCPSM3.vhd [10]. User write code in this new source file as per functional requirement of the system.

Microblaze and Picoblaze architecture design was mapped to Spartan 3A and 3E prototype boards respectively and functionality of the complete system is tested using performance characteristics of sensor connected to the sensor node.

The protocols used in wireless sensor network are Wi-Fi, Bluetooth and ZigBee etc. The ZigBee module can operate in worldwide 2.4-2.4835GHz ISM band with maximum bandwidth of 250kbps.It is supported by 802.15.4 IEEE protocol [1]. ZigBee is selected due to low power consumption, short range personal area network, and more communication channel amongst three.

III DESIGN AND IMPLEMENTATION

A. Architecture and data flow

Master-slave architecture is adopted for the system. Depend on the ZigBee module we can say, it is a broadcast network with one master(M) node and number of slave modes (S1 to SN). At the slave sensors are deployed which are collected and processed at slave and send to master. Master receives it analyze and displays a message on LCD display if any limiting condition dominate. The remote observer at the master module now able to take necessary action in case of any unexpected situation. The system is to be designed with short range communication link. If the remote observer is beyond 80-100m range we need to switch for long range communication device

The master and slave are having destination and source address respectively. In this network the code is written such as each slave can only communicate to master.

B. Hardware and software configuration

The hardware required for the system is presented in the Figure 1. There are two section of the hardware

I. Sensor/Slave module
II. Master module

Sensor module: It consists of Sensors, data processing unit and a RF transmitter.

- Sensor: We have selected a load cell (model no- CZL 601) to measure stress. The load cell used is a single point, Aluminum compressed type of cell when the load on the cell increased the output voltage of the cell is increased. Hence an analog to digital convertor (ADC) is required. An ADC is designed by adding custom IP facility available in EDK. Load cell is interfaced to the

channel A of the ADC connector. Where ADC linear Technology LTC1407A-1 dual channel Analog to

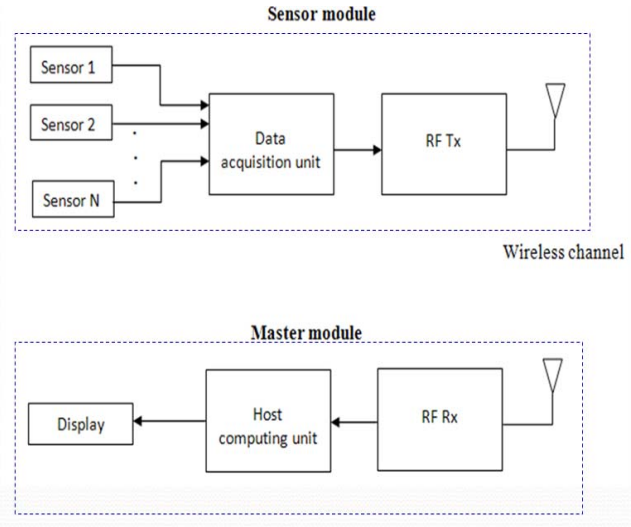

Figure 1: Block Diagram of the System

Digital Converter and a gain control amplifier linear technology LTC6912 dual channel programmable amplifier are converting the analog input to the digital and control the gain respectively. The inverting gain control amplifier is used. The equation 1 defines the digital output for the respective analog input at channel A of the ADC

$$Digital\ o/p(in\ Hex) = \frac{(V_{in} - 1.65) \times gain}{1.25} \times 8192 \quad --1$$

V_{in}- Analog input at channel A of ADC
1.65- Reference voltage of amplifier
1.25-Supply voltage of ADC

The output is 14-bit twos complement. Gain is varying from -1 to -100; we had set it to -1 for this amplifier.

- Data Acquisition System: A Microblaze softcore processor is used in a data acquisition unit. It is implemented on the Spartan 3E prototype board using the device 3S500EFG320- 4. The implementation of this processor has been realized in the EDK and ISE project environment. The most important tool of EDK10.1 environment is the Xilinx Platform Studio (XPS) which XPS includes a graphical user interface (GUI), along with a set of tools to develop hardware and software of the system. Figure 2 shows the interfacing of the Microblaze soft processor with other peripherals. Implementation began with ISE and embedded processor is added to ISE using XPS. XPS used to develop an embedded system hardware and software. SDK is the suggested Software Development Environment for simple and complex software applications. Configuration of the microprocessor, peripherals, and their interconnection, along with their intellectual property assignments, takes place in XPS.

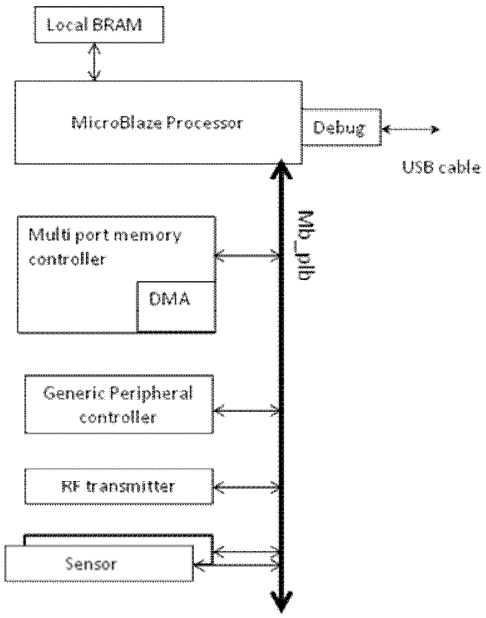

Figure 2: Architecture of Microblaze

The functionality verification of the hardware simulation is done with the help of Hardware description Language (HDL) simulator. A typical embedded system design project involves: hardware platform creation, hardware platform verification (simulation), software platform creation, software application creation, and software verification.

Base System Builder (BSB) is the wizard used to create a foundation for embedded System project. It creates new project with *.xmp (Xilinx microprocessor Project) file. It is a top level description file all the information including location of Microprocessor Hardware Specification (MHS) and Microprocessor Software Specification (MSS) files is saved in it. XMP file also contain C source and header files that XPS has to compile and also files that SDK has to compile. Project also includes FPGA architecture family and the device type which is to be targeted. BSB allows us to select this board available from Xilinx and lets us to select resources from board then automatically match the FPGA pin out to the board. After adding board and internal peripherals such as memory, registers etc., BSB displays a system summary page. After verifying the hardware now it's time to software setting. XPS also create a software file simultaneously with user defined application source file. It is defined by Microprocessor Software Specification (MSS) file which defines software element of the system driver and library

customization parameters for peripherals, processor customization parameters, standard devices, interrupt handler routines. User creates proper firmware for an acquisition unit of sensor module including other peripherals such as sensor and ZigBee transmitter. This collection of files, used in conjunction with EDK installed libraries and drivers, and drivers for custom peripherals you provide. It allows SDK to compile user defined applications. The compiled software routines are available as an Executable and Linkable Format (ELF) file. Figure 3 shows the flow of ELF file generation.

- RF transmitter: As Picoblaze is a single chip solution the RF ZigBee transmitter is interfaced using RS232 port C of same targeted device on FPGA board. It is connected to Microblaze trough mb_plb (microblaze_Processor Local bus)

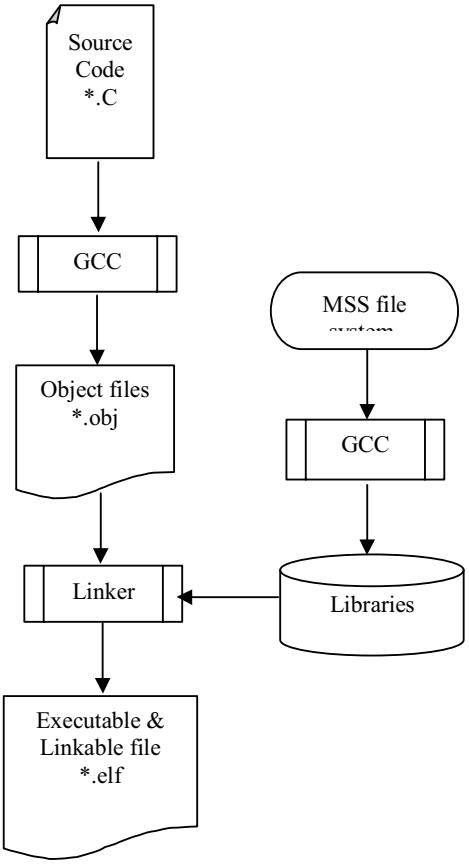

Figure 3: ELF File Generation Flow

Master module: It is implemented with a Picoblaze softcore processor which is again a single chip solution. External peripherals such as RF receiver and on board LCD display are interfaced to this processor.

- RF receiver: It is interfaced to the Picoblaze softcore processor with 8-bit input port. It receives data from slave and forwards it to processor.
- Host computing Unit: As mentioned above Picoblaze is handling the analysis of the data received. And displays appropriate message on the LCD display. Architecture of the Picoblaze is as shown in figure 4. It has two main blocks KCPSM3 and Block RAM and two external peripheral RF receivers and LCD display.

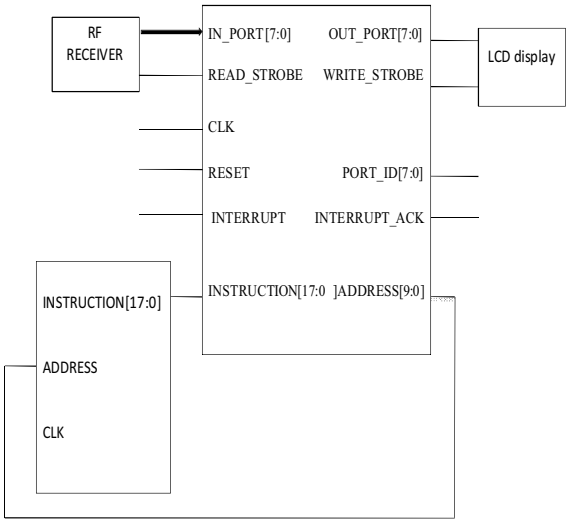

Figure 4: Architecture of Picoblaze

- KCPSM3 and Block RAM: The KCPSM 3(Ken Chapman programmable State Machine Version 3) is a.psm file. It provides ALU, registers, scratchpad RAM etc. The block memory consists of instructions which are fetched and executed by KSPSM3. One Block RAM is 1024 bytes in size. Now a new directory is created named as lcd_rs232, KCPSM assembler runs the .spam file and generates 15 types of different file such as .vhd, .hex, .txt file etc n this directory. Copy the following file to assembler directory KCPSM3.exe, ROM_FORM.vhd, ROM_FORM. coe, ROM_FORM.v and KCPSM.vhd to VHDL directory of lcd_rs232. Now create new project with located at lcd_rs232 directory using ISE 10.1 (Xilinx project navigator 10.1) suite. Using new project wizard set the device and project properties. Here we are using Spartan 3A Xilinx board having device is XC3S700AFG484-4. Now source files are added named as processor. Vhd, control.vhd. To add other entities of project such as RF receiver, LCD display some create a top level entity named as TOP.BEHAVE. Define input/output modules and add it in new source file. This file can be edited by user to define specific application which is expected to be performing by processor. Depending on number of peripheral to be used user can add source files having .vhd type. We added files display_232 _test.vhd and rx.vhd for LCD display and RF receiver respectively. At the end the code is synthesized and a bit file is downloaded on Spartan 3E using USB JTAG cable.

- LCD Display: It is interfaced to the Picoblaze and after simulation it displays the error message if limiting condition is crossed.

IV RESULTS

As this technology is new with the help of synthesis report the resource utilization, processing time and operating frequency for both Picoblaze and Microblaze are measured. Also performance characteristic of load cell is plotted.

TABLE1: RESULT-SENSOR MODULE

Resource utilization (Microblaze)		
Resources	Used/ Available	% utilization
Number of slices(FPGA)	1527/4656	32
Number of flip flop	1807/9312	19
Number of 4 I/p LUT	2471/9312	26
Number of bonded IOBs	41/232	17
Number of BRAM	17/20	85
Number of GCLKs	4/24	16
Number of DCMs	1/4	25
Time constraints		
Total data processing time(sensor to display)		14.41nsec
Total CPU time to XST completion		33.1sec
Maximum operating frequency		249.13MHz

TABLE 2: RESULT-MASTER MODULE

Resource utilization (Picoblaze)		
Resources	Used/available	% utilization
Number of slices (FPGA)	143/4656	3
Number of flip flop	162/9312	1
Number of 4 I/p LUT	256/9312	2
Number of bonded IOBs	19/232	8
Number of BRAM	1/20	5
Time constraints		
Total data processing time		19 nsec
Total CPU time to router completion		17sec
Maximum operating frequency		215MHz

From table 1 and table 2 information we can say that data processing time for both the processor is in tens of nano-second and operating frequency is 200+ MHz which is 150MHz for hardcore processor. Eventhough the number sensor increased the processing time will reach to few seconds

Figure 6 shows the block schematic of Microblaze (sensor module) generated by the EDK-XPS tool. This shows the actual resources used for architecture of a sensor module.

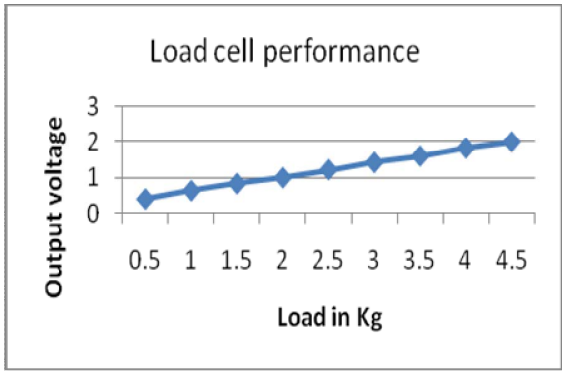

Figure 5: Performance of Load Cell

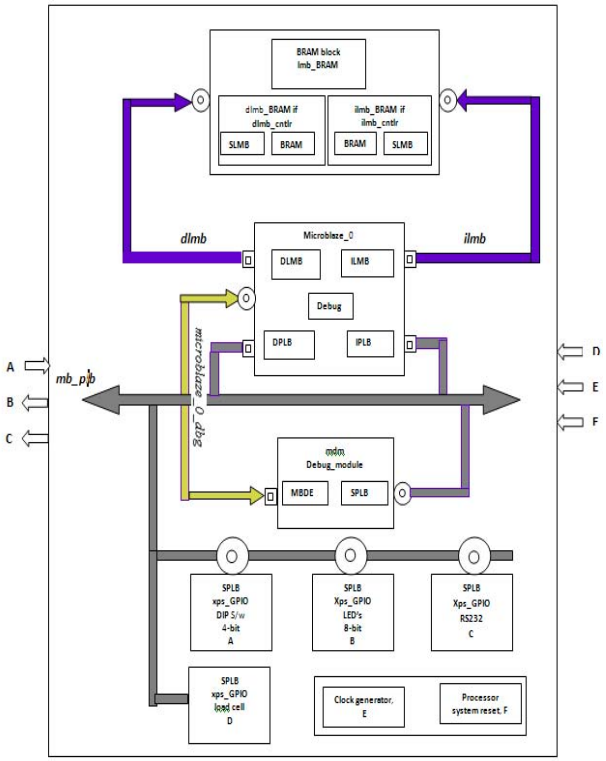

Figure 6: Block schematic of Micoblaze

V CONCLUSIONS

Softcore architectures Picoblaze and Microblaze implemented successfully on the Spartan 3E and 3A standard prototype board respectively. One complete module is implemented on single chip hence size is very small. Software and hardware combined architecture of soft processor reduced the processing and routing time of the system to nanoseconds. As system is implemented effectively on FPGA platform with Xilinx software environment i.e. ISE 10.1 and EDK 10.1 it is more durable as compare to other existing system. A lossless and secure

wireless link is established successfully and tested for the distance of 30m (Short distance communication) using ZigBee RF module.

VI FUTURE SCOPE

Wireless Sensor Network is having number of sensor nodes which required batteries of 12 V supply. Thus it will function till the battery is working else sensor node service will interrupt. For long term application such as SHM these modules may powered with energy harvesting systems with power management scheme so that power consumption can be reduce further. And system will provide uninterrupted service. Also the data processed at sensor module can be placed over internet for long distance communication.

REFERENCES

[1] Ping Wang,Yan Yan, Gui Yun Tian, Omer Bouzid, Zhiguo Ding "Investigation of wireless sensor network for structural health monitoring", Journal of Sensor, Vol 2012

[2] Danielle Reichel, Antoine Druilhe and Tuan Dang, "An Innovative Wireless Sensor Network Protocol Implementation using a Hybrid FPGA Technology ",World Academy of Science, Engineering and Technology, Vol-53, 2011

[3] Prayati, Tyler Harms, Sahra Sedigh, Filippo Bastianini "Structural health monitoring of bridges using wireless sensor network" IEEE instrument and measurement magazine Dec 2010.

[4] B.Muralikrishna, K.Gnana Deepika, "Input/output Peripheral Devices Control Through Serial Communication Using Microblaze Processor", 978-1-4577-1545-7 ,IEEE,2012

[5] Ali El Kateeb, Aiyappa Ramesh L. Azzawi, "Wireless sensor nodes Processor architecture and Design", 987-1-4244-1643, IEEE 2009

[6] Ioan Lie , Cristian Ionici, Aurel Gontean , Mihail Cernăianu, "EDK Implemented Temperature Controller", 33rd Int. Spring Seminar on Electronics Technology, IEEE 2010.

[7] Dae-Man Han and Jae-Hyun Lim " Smart Home Energy Management System using IEEE 802.15.4 and ZigBee", IEEE transaction on consumer electronics, vol 56 no. 3, August 2010.

[8] http://www.xilinx.com/support/documentation/sw_manuals/xilinx 13_1/mb_ref_guide.pdf

[9] http://www.xilinx.com/products/intellectualproperty/Picoblaze.htm

[10] http://disi.unitn.it/~picco/papers/tower.pdf