RESEARCH ARTICLE OPEN ACCESS

Hardware Realization of interfaces in Wireless Sensor Node Implementation on FPGA

S. R. Adhav, Prof. U. R. More,
Department of Electronics &Telecommunication
SCOE, Pune.

Abstract:

This paper presents interfaces required in wireless sensor node (WSN) implementation. Here keyboard, LCD, ADC and Wi-Fi module interfaces are presented. These interfaces are developed as hardware prototypes in the application of wireless sensor node as a single chip solution. Protocols of these interfaces have been described with the help of their hardware simulations and synthesis reports.

The end application is proposed to monitor physical parameters remotely using wireless protocol. The sensor node has to be implemented on Field Programmable Gate Array (FPGA). The proposed node design is reconfigurable, and hence flexible in context of future modification. Xilinx platform is proposed for synthesis, simulation and implementation.

Keywords — FPGA, wireless sensor node.

1. Introduction

Implementation of digital designs on FPGA is advantageous over controller based programmable based implementations. This is because configurable platforms like PLD takes less power, less memory, less space. More over these designs are flexible for changing specifications implementations. [4]

WSNs have great potential for many applications in scenarios such as remote monitoring of physical parameter like temperature, humidity, pressure etc. where user can save time, money, manpower efforts and provide overall control immediately.[3] Less power consumption leads long time battery back-up. [6]

Proposed WSN node is reconfigurable in nature. It can be used as a generalized data acquisition

platform for wireless sensor networks.[5] It also facilitates us to add design for testability (DFT) in it.

This paper gives detailed realization of following interfaces.

a. Key board interface.
b. LCD (Liquid crystal Display) interface.
c. 4 Channel ADC (Analog to Digital Converter) interface.
d. WiFi interface.

2. Structure of Hardware of Interfaces

I] Keyboard Interface Realization.

4X4 matrix key pad interface has been designed to interface to FPGA. It has been modeled in VHDL. [2]

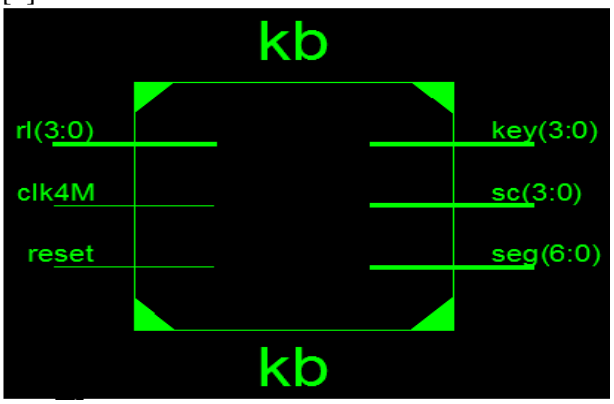

Figure 1: RTL Schematic

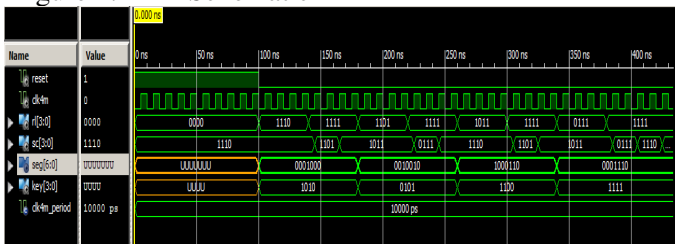

Figure 2: Timing diagram

In figure 2, it has been demonstrated that if all bits of return lines (rl) are 1, means none of the keys is pressed, and sc lines advance to next scan. If key is detected then it is converted into seven segment format and displayed on seven segment display.

Synthesis Report of keyboard interface:

Top Level Output File Name : kb
Output Format : NGC
Optimization Goal : Speed
Keep Hierarchy : No

Design Statistics:
IOs : 21

Cell Usage:
BELS : 44
GND : 1
INV : 1
LUT2 : 4
LUT3 : 6
LUT4 : 25
MUXF5 : 6
XORCY : 1
FlipFlops/Latches : 14
FDC : 3
LD : 11
Clock Buffers : 1
BUFGP : 1
IO Buffers : 20
IBUF : 5
OBUF : 15

Device utilization summary:
Selected Device: 3s250ecp132-5

Number of Slices:	21 out of	2448	0%
Number of Slice Flip Flops:	3 out of	4896	0%
Number of 4 input LUTs:	36 out of	4896	0%
Number of IOs:	21		
Number of bonded IOBs:	21 out of	92	22%
IOB Flip Flops:	11		
Number of GCLKs:	1 out of	24	4%

In this interface, key de-bounce is considered in implementation for faithful key stroke recognition. Therefore scanning rate of scan lines is kept as low as 2 ms.

It is necessary to down convert the system clock up to 2 mili-second or 512 Hz for row scanning. Key board interface is required for setting threshold levels for various physical parameters like temperature, humidity, light. It makes the system controllable and measurable too.

ii] ADC interface Realization

Figure 3. Shows RTL schematic of ADC interface to PLD. An interface has been modeled in VHDL [1] using Xilinx platform.[1] Sampling clock is derived from system clock. Minimum 20 KHz sampling frequency is sufficient, for the parameters like temperature, light, and humidity.

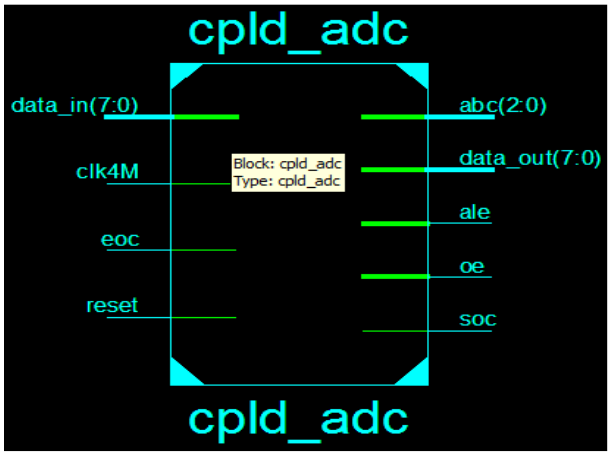

Figure 3: RTL Schematic

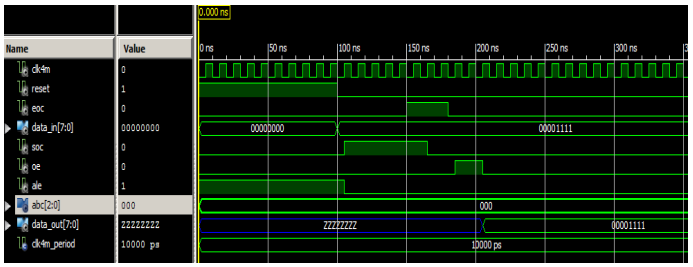

Figure 4: Timing diagram

In timing diagram, it is clearly indicated that, channel zero is being sampled by putting channel address 000. Followed by latch pulse and raising SOC. ADC replies in terms of EOC, after its conversion time. Here SOC is pulled down by PLD it raises pulse of OE. At falling edge of OE, digital data from ADC is placed on data_out bus.

Synthesis Report:

Top Level Output File Name : cpld_adc
Output Format : NGC
Optimization Goal : Speed
Keep Hierarchy : No

Design Statistics

IOs : 39

Cell Usage:
BELS : 166
GND : 1
INV : 7
LUT1 : 33
LUT2 : 13
LUT3 : 15
LUT4 : 17
LUT4_L : 1
MUXCY : 41
MUXF5 : 2
VCC : 1
XORCY : 35
FlipFlops/Latches : 62
FDC : 46
FDCE : 2
FDP : 1
LD : 13
Clock Buffers : 1
BUFGP : 1
IO Buffers : 38
IBUF : 15
OBUF : 23

Device utilization summary:

Selected Device: 3s250ecp132-5

Number of Slices:	45 out of	2448	1%
Number of Slice Flops:	54 out of	4896	1%
Number of 4 input LUTs:	86 out of	4896	1%
Number of IOs:	39		
Number of bonded IOBs:	39 out of	92	42%
IOB Flip Flops:	8		
Number of GCLKs:	1 out of	24	4%

iii] LCD interface Realization.

Figure 5 Shows LCD interface to PLD. It has been modeled in VHDL using Xilinx platform.[1] In this interface, 16X2 character LCD is implemented. First of all LCD is initialized. This is done by sending control word information to LCD by PLD. [2]

This information has been passed to LCD with minimum 16 msec clock. It is also derived from system clock. Once it is initialized, cursor blinks on LCD. Then user sends a character by having strobe signal. LCD can receive data from internal memory and display it.

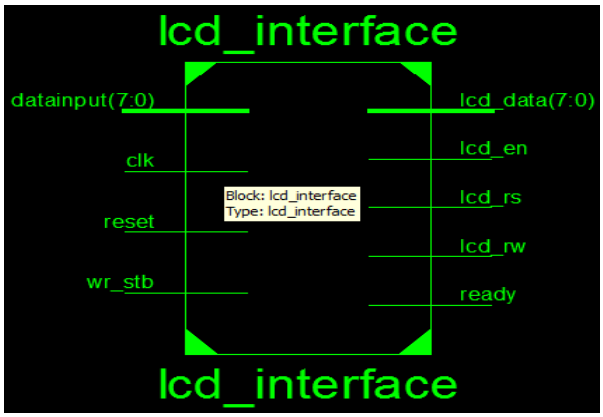

Figure 5: RTL schematic

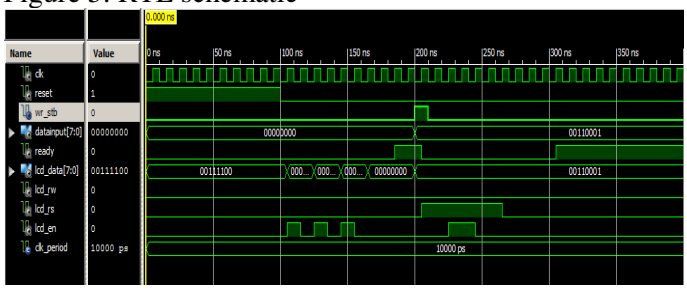

Figure 6: Timing diagram

Timing diagram shows that LCD is initialized first by sending commands. 'Ready' signal is asserted. Then if valid data is present on input data bus, strobe is applied by user. At strobe detection, 'lcd_rs' becomes one, followed by 'lcd_en'. Then 'lcd_en' is pulled down, followed by 'lcd_rs'. Here 'lcd_data' is displayed on LCD module.

Synthesis report:

Top Level Output File Name : lcd_interface
Output Format : NGC
Optimization Goal : Speed
Keep Hierarchy : No

Design Statistics

IOs : 23

Cell Usage:

BELS : 79
GND : 1
INV : 2
LUT1 : 19
LUT2 : 4
LUT2_L : 1
LUT3 : 7
LUT4 : 4
MUXCY : 19
MUXF5 : 1
VCC : 1
XORCY : 20
FlipFlops/Latches : 30

# FDC	: 27
# FDCE	: 2
# FDP	: 1
# Clock Buffers	: 1
# BUFGP	: 1
# IO Buffers	: 22
# IBUF	: 10
# OBUF	: 12

Device utilization summary:

Selected Device: 3s250ecp132-5

Number of Slices:	19 out of 2448	0%
Number of Slice Flops:	30 out of 4896	0%
Number of 4 input LUTs:	37 out of 4896	0%
Number of IOs:	23	
Number of bonded IOBs:	23 out of 92	25%
Number of GCLKs:	1 out of 24	4%

iv] WiFi Module Interface Realization.

Figure 7. and 8 Shows RTL schematic of WiFi interface at transmitter and receiver side to PLD. It has been modeled in VHDL [2] using Xilinx platform.[1]

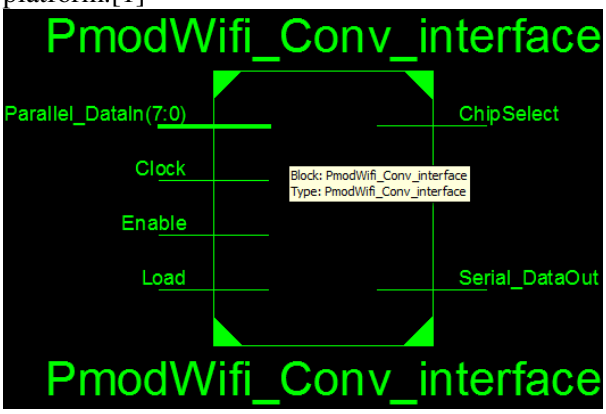

Figure 7. RTL schematic of WiFi Transmitter Interface.

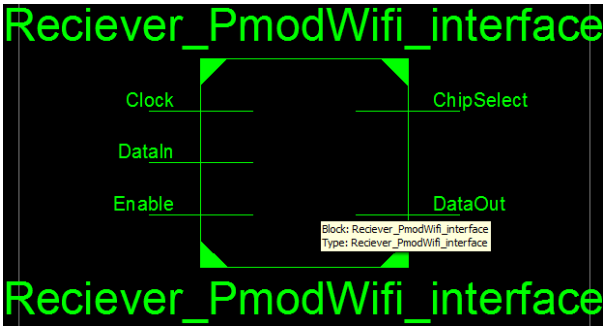

Figure 8. RTL schematic of WiFi Receiver Interface.

Figure 9 shows that, serially received data with serial clock is presented at DataOut pin when 'Enable' pin is high.

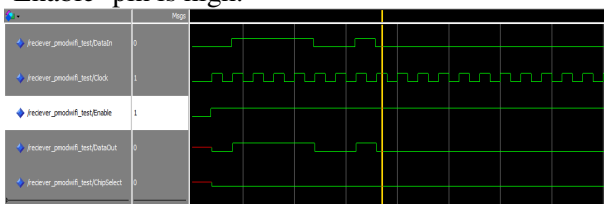

Figure 9. Transmitter Timing Diagram

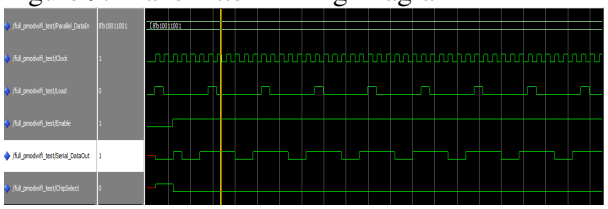

Figure 10. Receiver Timing Diagram

In figure 10 it has been shown that parallel loaded data is converted into serial form and with Enable pin high and with load pulse it is kept on Serial_DataOut.

Synthesis Report:

Top Level File Name: PmodWifi_Conv_interface

Output Format	: NGC
Optimization Goal	: Speed
Keep Hierarchy	: No

Design Statistics

# IOs	: 13
Cell Usage :	
# BELS	: 11
# INV	: 2
# LUT3	: 8
# VCC	: 1
# FlipFlops/Latches	: 11
# FD	: 8
# FDE	: 1
# FDR	: 2
# Clock Buffers	: 1
# BUFGP	: 1
# IO Buffers	: 12
# IBUF	: 10
# OBUF	: 2

Device utilization summary:

Selected Device: 3s250ecp132-5

Number of Slices:	6 out of 2448	0%
Number of Slice Flops:	10 out of 4896	0%
Number of 4 input LUTs:	10 out of 4896	0%
Number of IOs:	13	
Number of bonded IOBs:	13 out of 92	14%
IOB Flip Flops:	1	
Number of GCLKs:	1 out of 24	4%

3. Proposed System Architecture

Figure11. Shows proposed system block diagram, whose interfaces have been realized. Emphasis is on interface modeling for ADC, keyboard, LCD, and communication module.

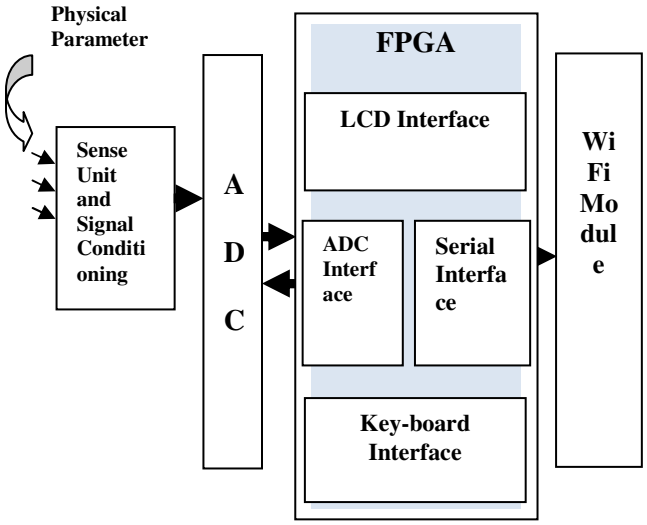

Figure 11. **System Block Diagram**

ADC accepts conditioned physical parameters from various sensors. FPGA based processing unit has also a local buffer to retain these. These values are displayed with LCD at node side as well as receiving node side. Key board is interfaced to set various threshold values during processing if required. These values are communicated through wireless protocol over long/short distance in wireless media.

4. Conclusion

Work showed here presents hardware realization of proposed system architecture's block sets. These are integrated blocks of tailor made wireless sensor node. Many features become merits of the system viz. programmability; granularity and reconfigurability. System may consume low power and area. Parallelism is biggest merit of PLD based architecture which increases throughput. All block sets of the design have been functionally tested using simulator.

5. References

[1] Karen parnell and Nick Mehata , Programmable logic design Quick Start Hand book, June 2003, Forth edition, ISE 5.1i. E XILINX.

[2] Synopsys Inc. FPGA Compiler II / FPGA Express, VHDL Reference Manual, Version 1999.05, May 1999.

[3] EL-Medany,W.M., "FPGA based MIMO system for wireless sensor network". System of System Engineering , 2009. SOSE 2009, IEEE. International conference. Publication Year: 2009, pages(s) 1-5, IEEE conference publications.

[4] Yan Sun, Le li, Hong Luo, "Design of FPGA – Based Multimedia node for WSN". Wireless communication, networking and mobile computing (WICOM), 2011, 7 th International conference on Digital object identifier : Publication Year : 2011, Page (1-5) IEEE conference publications.

[5] A. Abdaoui, K. Gurram, M. Singh, A. Errandani, E. Châtelet, A. Doumar and T. Elfouly. "Video Acquisition between USB 2.0 CMOS Camera and Embedded FPGA system". 978-1-4577-1180-0/11/©2011 IEEE

[6] Leonardo Gasparini, Massimo Gottardi, Roberto Manduchi, Nicola Massari, Dario Petri, "FPGA Implementation of a People Counter for an Ultra-Low-Power Wireless Camera Network Node". 978-1-4244-9137-7/11/ ©2011 IEEE