SECURE REMOTE PROTOCOL FOR FPGA RECONFIGURATION

G.Balasubramanian[1], N.Keerthana[2]

[1]Assistant Professor, [2]PG Scholar, ECE Department, Arasu Engineering College, Kumbakonam-612 001

Abstract

In most of the wireless sensor network nodes main functionality in software are implemented using CPU. Since total energy consumption of periodic measurement and network listening are considerably increased. In order to tackle this problem novel reconfigurable peripheral blocks are introduced into the WSN. For ultra-low-power sensor networks, finite state machines are used for simple tasks where the system's microprocessor would be overqualified. This FSM unit autonomously handles simple sub-tasks of the periodic activities, Microprocessor to remain in a sleep state, thus saving energy. This work presents for implementing transition based reconfigurable FSM

Keywords: - *Finite State Machine (FSM), Reconfiguration, Wireless Sensor Network (WSN).*

----------***----------

1. INTRODUCTION

Wireless sensor network development was motivated by military application such as battlefield surveillance. These networks are used in many industrial and consumer application, such as industrial process monitoring and machine health monitoring and so on. In this paper secure remote protocol has been proposed for FPGA Reconfiguration.. Reconfiguration platform has feature that allow easy reuse of the node in several application avoiding redesigning the system from scratch. The node includes an FPGA which is the core of the reconfigurable capabilities of the node. Reconfiguration area can be remotely or dynamically configured [7]. Previously microprocessor and microcontroller has been used [1] which consume more power. In order to reduce power consumption and to transmit the bit stream confidently we go for remote configuration.

In high volume application most challenging trend is to reduce power. But it is difficult in sensor node, there are two ways to reduce power consumption (i) activity of node to a portion of time (ii) integrate most component into a single chip. Sensor node consists of multiple individual commercially available components waste lot of energy in voltage level adaptation [8]. In this paper serial communication protocol is used (Which transmit data one at a time).

The paper outline is as follows. Section 2 presents Remote Configuration and their feature. Section 3 presents Reconfigurable architecture. Section 4 presents MD5 algorithm for data integrity check. Section 5 presents Transition based FSM. Section 6 presents Experimental setup. Section 7 presents security scheme. Section 8 presents simulation and result. Section 9 presents comparison table for power consumption. Section 10 presents conclusion and future work.

2. REMOTE CONFIGURATION

Remote update for hardware system is enabled by Field Programmable Gate Array (FPGA). In this work, previous ideas [6, 9] are improved and implemented in order to achieve flexibility. Remote configuration has following feature: fix software bugs, adapt to changing user needs and environmental condition in which the network is deployed, shorten software development phase, make software robust, complete application replacement.

3. RECONFIGURABLE ARCHITECTURE

3.1 Need for Network Reconfiguration

- Node gets failed or low response time
- Signal strength seems to be quite weak
- Change the router configuration when number end device increase or decrease
- Add a new device with high secure.

Generally there are two type of configuration (i)STATIC (The device is not active during reconfiguration process. While data is send into FPGA the rest of device is stopped and brought up after configuration complete) (ii)DYNAMIC (Active reconfiguration permit to change the part of the device while reset of an FPGA is still running). In this paper dynamic configuration is used.

3.2 Reconfigurable Architecture

We introduce reconfigurable hardware block to Wireless Sensor Network. Which independently conduct simple sub-task instead of the CPU. Software can be modified by reprogramming the code memory; synthesized logic cores require a redesign of the chip.

3.3 Proposed Block Diagram

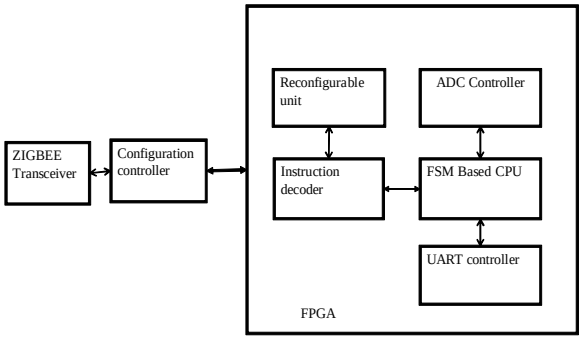

Fig 1: Hardware Implementation Block Diagram

3.3.1 Block Diagram Description

ADC Controller

Generate the start of conversion, clock signal to external ADC. This controller monitor the output enable signal from ADC. It will read the sensor data which is in hex format and convert the hex value into decimal and ASCII format.

UART Controller

UART Controller generate require baud rate for transmit the data (9600). Compress the sensor data and discard the redundant bits. UART arrange the sensor data into 10bit frame(8 bit data one start bit and stop bit) and shift the data in vitwise each 1/9600 clock period.

FSM based CPU

Generate control signal to UART controller and ADC controller. FSM read the opcode from instruction decoder and execute the task and monitor the reconfigurable unit for update the new bit stream.

Configuration Controller

This controller will read the bits stream from Zigbee transceiver. Decrypt the bit stream data, controller will verify the data integrity through MD5 algorithm and generate reconfiguration boot command to FPGA.

3.3.2 Feature of Proposed System

Reconfigurability

Central control unit can set the run-time parameter of the device and/or update/upgrade the firmware of the system over the air(OTA).

Plug –N-Play

Depending on the application needs/requirements, different infrastructure and their configuration can be deployed quickly.

Self-Identification

Necessary for high volume data collection system as multiple sensor/actuator may be read at one time.

Self-Calibration

The intelligent adaptive sensor can accurately measure data and self- calibrate without significant user intervention.

Wireless Connectivity

Provide bi-directional communication over a wireless connection.

4. MD5 ALGORITHM

Previously MAC (message authentication code) value is calculated when making configure with remote update server. This security analysis the integrity and the confidentiality of the bitstream for remote updating process [3,6&9]. In this paper MD5 algorithm is used for data integrity.

MD5 algorithm was developed by Professor Ronald L. Rivest in 1991. According to RFC 1321, “MD5 message-digest algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input …The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA”.

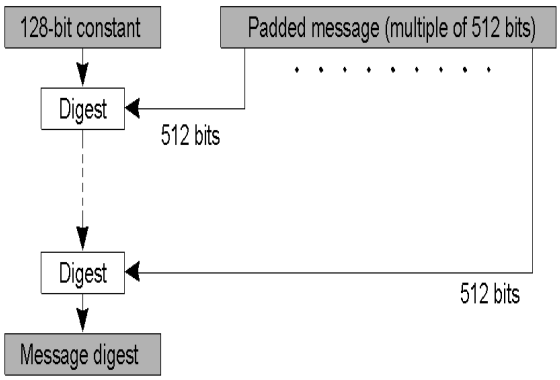

Fig 2: MD5 Algorithm Structure

5. TRANSITION BASED FSM

Reconfigurable logic block are implemented usually as a FSM. We can implement any possible FSM for given input, output and state transition. Reconfiguration take place by writing appropriate content in RAM. For whole state transition large reconfigurable block are needed which are complex in order to reduce the complexity we construct reconfigurable block for each transition.

[8]Transition instead of state function is known as transition based FSM. Transition based FSM is a two phase process. In first phase, necessary resource are specified. In second phase, transition based FSM implemented in embedded chip is configured. Transition based FSM consists of (i) State Register (ii) Input Switching Matrix (iii) State Selection Gate (iv) Input Pattern Gate (v) Next State.

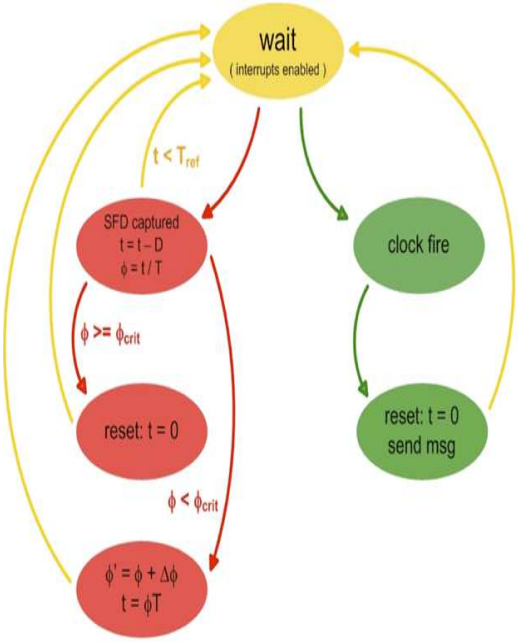

Fig 3: FSM CPU work mode

5.1 Data Forwarding

The aggregator receives data from both the devices & CCU and forwards it to its in-tended recipients. Data from devices are marked with their specific device IDs at the aggregator and sent to CCU. Data from CCU is marked with device IDs and the aggregator extracts this information and forwards the data to the respective-device.

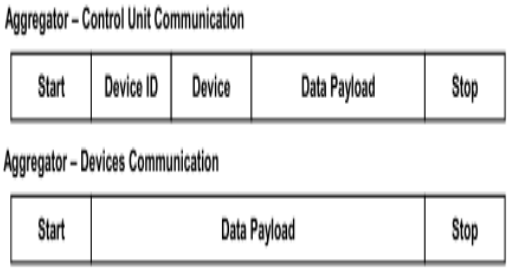

Fig 4: Payload over wireless link

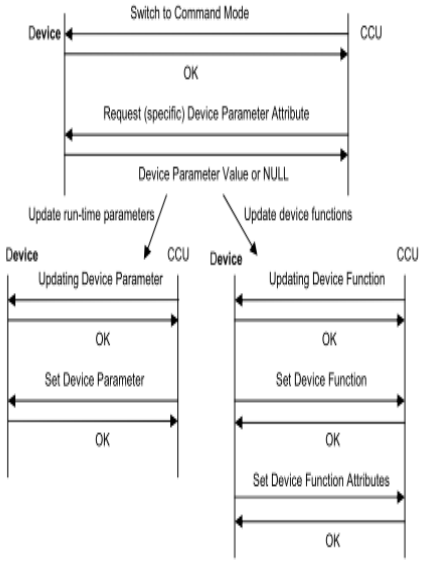

Fig 5: Message Exchange Flowchart for Device Reconfiguration (OTA)

6. EXPERIMENTAL SETUP

Hardware implementation is realized with FPGA. FPGA are programmable digital logic chip, which means you can program them to do almost any digital function. Altera is the second FPGA heavyweight. Altera focus is on easy of use with an HDL software suite that has traditional been very good their silicon has a bit less feature their architecture is not open. Altera quartus is a programmable logic device software from altera. In this paper we are using quartus II software version 9.0.

6.1 ALTERA DE1 Board

The DE1 (Development and Education) board has many features that allow the user to implement a wide range of designed circuits, from simple circuits to various multimedia projects. To provide maximum flexibility for the user, all connections are made through the Cyclone II FPGA device.

Thus, the user can configure the FPGA to implement any system design.

6.2 Cyclone II FPGA Configuration

There are two ways to configure the cyclone II FPGA (i) JTAG Programming (In this method of programming, named after the IEEE standards Joint Test Action Group, the configuration bit stream is downloaded directly into the Cyclone II FPGA. The FPGA will retain this configuration as long as power is applied to the board; the configuration is lost when the power is turned off.) (ii) AS Programming (In this method, called Active Serial programming, the configuration bit stream is downloaded into the Altera EPCS4 serial EEPROM chip. It provides non-volatile storage of the bit stream, so that the information is retained even when the power supply to the DE1 board is turned off. When the board's power is turned on, the configuration data in the EPCS4 device is automatically loaded into the Cyclone II FPGA). In this paper we are using AS Programming.

6.3 XILINX ISE VS ALTERA QUARTUS II

Altera quartus II has better GUI(Graphical User Interface) than Xilinx, it provide better HDL support than Xilinx.

7. SECURITY

- In this reconfiguration security is provided using three different secure algorithms.
- Public key encryption-RSA algorithm- for secure key sharing between host and reconfigurable node.
- Private key encryption-AES algorithm- 12b bit AES encryption for both host and receiver reconfigurable node.
- HASH function-(MD5/CRC/SHA1)- for integrity check at reconfigurable node.

8. SIMULATION AND RESULT

For simulation we are using Proteus(Design Software). Proteus is software for microprocessor simulation, schematic capture and printed circuit board. Here ATMEL(89S51) microcontroller is used. 89S51 is a 8bit CMOS Microcontroller. It has following Feature (i) 4 K bytes of Flash (ii) 32 i/o lines (iii) 128 bytes of RAM (iv) in system programmable flash and so on. Assembly language are used for developing program. Coding is synthesized using Keil Vision IDE. Simulation result are shown in figure. Figure shows two microcontrollers which can be remotely configured from host system.

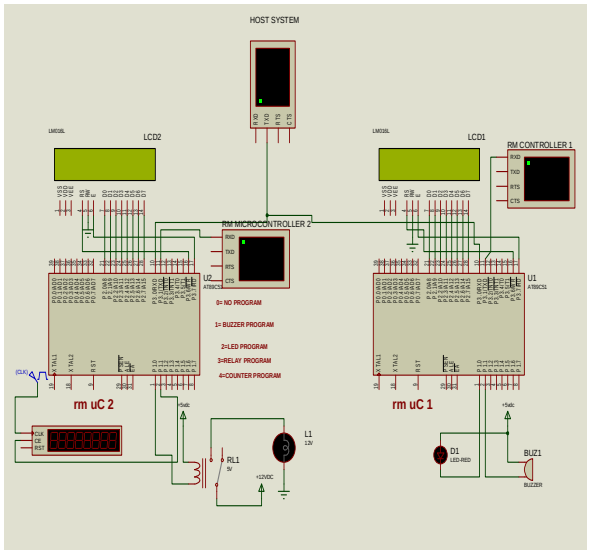

Fig 6: Simulation Module

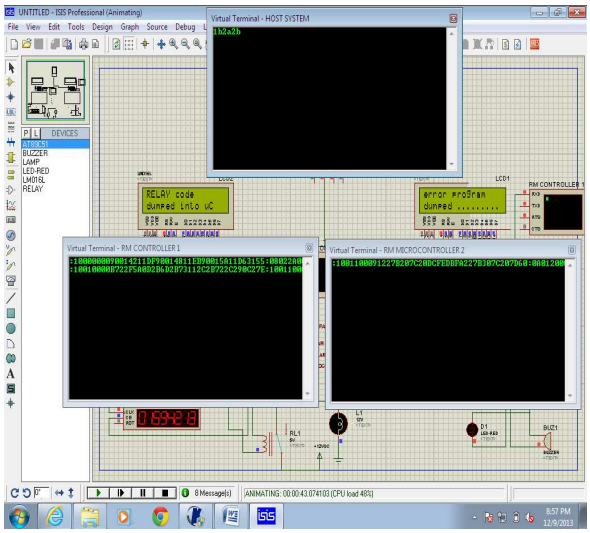

Fig 7: Simulation Output

9. COMPARISION

Table 1: Power consumption by various devices

S.NO	DEVICE	POWER CONSUMPTION(nJ)
1	MPSP430F1232	191.48
2	MPSP430F2232	222.63
3	MPSP430F5418	189.15
4	ATmega	225.90
5	FPGA	90 times less energy than MPSP430F5418

10. CONCLUSION AND FUTURE WORK

The optimization potential of the power consumption of a sensor interface for WSN node was depicted .The CPU with additional reconfigurable hardware blocks that take over simple tasks from the CPU to release it from frequent wake-ups. A novel reconfigurable block are introduced in to WSN that tackle the problem due to wakeup overhead and waiting periods the total energy consumption of periodic measurement and network listening.

Future work include implementation of Transistion based Reconfigurable FSM(TR-FSM) and Hardware Performance Analysis. FSM are used for ultra low power sensor nodes. In this work reconfigurable Finite State Machines: Transition-based Reconfigurable FSM (TR-FSM) is used. A simple solution for reconfigurable FSM is the use of a RAM in read only mode. This architecture allows to implement any possible FSM with the given number of inputs, outputs and state signals and is reconfigurable by writing the appropriate RAM content. However, the size of the RAM grows exponentially with the number of inputs and state signals. On the other hand, the implemented FSM can have transitions from any state to any other.

REFERENCES

[1]. Ann Gordon-Ross, Alan D. George and Rafael Garcia (2009), “Exploiting Partially Reconfigurable FPGAs for Situation-Based Reconfiguration in Wireless Sensor Networks”, ISBN: 978-0-7695-3716-0.

[2]. Andreas Engel, Andreas Koch and Bjorn Liebig (2012), “Energy- Efficient Hetrogenous Reconfigurable Sensor Node For Distributed Structural Health Monitoring”, E-ISBN : 978-2-9539987-4-0.

[3]. An Braken, Nele Mentens, Jo Vliegen and Ingrid Verbauhede,”Secure Remote Reconfiguration of FPGA”. http://drops.dagstuhl.de/opus/volltexte/2010/2839.

[4]. Carlos Eduardo Pereira, David Cemin and Marcelo Gotz (2012),” Reconfigurable Agents for Heterogeneous Wireless Sensor Networks”, ISBN: 978-1-4673-5747-0.

[5]. T,Casstro,A and Riesgo (2007), ”A Reconfigurable FPGA-Based Architecture For Modular Nodes In Wireless Sensor Networks,” ISBN:1-4244-0606-4.

[6]. Florian Devi, Lionel Torres (2010),”Secure Protocol for Remote Bitstream Update Preventing Replay Attacks on FPGA”, International Conference On Field Programmable Logic and Application.

[7]. Jia, Z, Liu, Xie, s (2012), ”Hardware reconfigurable wireless sensor network node with power and area efficiency ”

[8]. Johann Glaser, Jan Haase, Markus Damm, “A Novel Reconfigurable Architecture for Wireless Sensor Network Nodes’’.

[9]. Run-feng Huan (ICAISE 2013),”Secure Mechanism for Remote Updates of Reconfigurable Computing Platform”.