An AES Evaluation for 2.4GHz Wireless Network Based on SOPC

Xiaoying Liang
Guangdong Women's Polytechnic College, minnielxy@gmail.com

Abstract

In modern systems, data security is needed more than ever before and many cryptographic algorithms are utilized for security services. Wireless Sensor Networks (WSN) is an example of such technologies. In this paper an innovative SOPC-based approach for the security services evaluation in WSN is proposed that addresses the issues of scalability, flexible performance, and silicon efficiency for the hardware acceleration of encryption system. The design includes a Nios II processor together with custom designed modules for the Advanced Encryption Standard (AES) which has become the default choice for various security services in numerous applications. The objective of this mechanism is to present an efficient hardware realization of AES using very high speed integrated circuit hardware description language (Verilog HDL) and expand the usability for various applications. As compared to traditional customize processor design, the mechanism provides a very broad range of cost/performance points.

Keywords: FPGA, AES, 2.4GHz wireless network, embedded systems, SOPC

1. Introduction

With the continual and rapid expansion of internet and wireless-based communications across open networks security problem has become most important over the networks. Therefore, plenty of cryptographic algorithms (e.g. TEA, NTUREncryp, SEA, and ECC) have recently emerged. Cryptography is one of the strongest tools for controlling against many kinds of security threats [1]. However, these cryptographic algorithms are very computationally intensive. For this reason, research on hardware implementation of cryptographic algorithms has been intensive during the recent years. Field Programmable Gate Arrays (FPGAs) are very attractive platforms for implementing cryptographic algorithms. Many FPGA implementations of the AES algorithm have been introduced [2,3]. With the exponential increase in FPGA size over time (Moore's law), FPGA gives an opportunity to create, test and prototype ASIC silicon devices' blocks. FPGAs are now able to hold multi-million gate designs, opening up the possibility of implementing a full embedded system on a single programmable chip [4]. This is the idea behind the concept of system-on-a-programmable-chip (SOPC), a FPGA-base embedded system which may integrates hard-core or soft-core CPU, digital signal processor (DSP), memory, I/O and different communication busses. Since its high flexibility and cheap price, SOPC is very competitive and is considered as the future of semiconductor industry [5-10]. Naturally, it is possible to implement a relatively high performance, user parameterizable cryptographic algorithms at low cost on FPGA.

In this paper, we propose an innovative SOPC-based implementation mechanism for the AES algorithm evaluation. The mechanism is optimized especially for Altera FPGAs and it is design to make use of embedded memory blocks inside Cyclone II. The objective of this mechanism is to reduce the cost of the algorithm implementation, expand the usability for various applications and increase the flexibility of system.

The rest of this paper is organized as follows: Section 2 presents an overview of related technology. In section 3 we describe hardware and software co-design and the related work of AES algorithm implementation in FPGA. Conclusion is provided in section 4.

2. Overview

Our approach to achieve above goals is based on Altera Nios II embedded processor technology that can be configured to be 16/32 bits, have data or instruction pipelining, hardware multipliers, custom instructions with hardware acceleration to increase system performance. In our design, the encryption
rounds are implemented by hardware in FPGA chip, other are implemented by software using Nios II embedded processor. We provide some preliminary information in brief for this design as follow:

2.1. Field Programmable Gate Arrays

Field Programmable Gate Arrays (FPGAs) are programmable semiconductor devices that are based around a matrix of configurable logic blocks (CLBs) connected via a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together". The continuous improvement of integrated circuit technology has had a dramatic impact on the architecture of FPGAs. The architecture of a Field Programmable Gate Array (FPGA) is illustrated in Fig.1. Field Programmable Gate Arrays (FPGAs) are becoming increasingly popular as a deployment implementation for circuit designs. FPGAs can be configured by the customer or designer after manufacturing and extensively used in rapid prototyping and verification of a conceptual design. They are also used in electronic systems when the mask-production of a custom IC becomes prohibitively expensive due to the small quantity. As opposed to Application Specific Integrated Circuits (ASICs) where the device is custom built for the particular design, FPGAs are another solution from the market, which require much less development costs which is a very important issue as far as limited production volumes are concerned. The implementation of the AES algorithm based on FPGA devices has a lot of advantages over the implementation based on ASICs, such as a shorter time to market, ability to reprogram in the field to fix bugs, and lower non-recurring engineering costs.

![Figure 1. Structure of an FPGA](image)

2.2. Why Nios II Embedded Processor?

We chose Altera system-on-a-programmable-chip (SOPC) solution including the Nios II processor for the following reasons:

- The Altera NIOS II softcore processor designed to address a wide range of embedded applications is a 32-bits scalar RISC with Harvard architecture, 6 stages pipeline, 1-way direct-mapped 64KB data cache, 1-way direct-mapped 64KB instruction cache, maximum 2-Gbytes of accessible address space and can execute up to 150 MIPS.
- Nios II processor is called a configurable soft-core embedded processor that means it is not physically existed. The main interest of this softcore processor is its extensibility and adaptability. The Altera Nios II processor supports flexible product development designs at low cost.
The Nios II processor's customizable instruction set can accelerate time-critical software algorithms and reduce a complex sequence of standard instructions to a single instruction implemented in hardware, which provides faster execution than implementing these operations in software.

Based on the Nios II processor, we can customize the peripheral intellectual property (IP) based on Avalon bus using customized instructions. By doing so, we have greatly improved the digital signal processing capacity of the system and this technology supports multiple parallel data channels to achieve high throughput in security services evaluation.

Altera provides a complete set of FPGA development tools including the Quartus II software, SOPC Builder, ModelSim-Altera software, and SignalTap II embedded logic analyzer. The Nios II Integrated Development Environment (IDE) provides the tools to accomplish software development tasks such as program editing and debugging. With the Nios II IDE, it is possible to simplify software design and all software development tasks.

2.3. Hardware platform

For the main processing stage, a NIOS II development board was chosen (Fig.2). Key features include:

- A Low-power consumption Altera Cyclone II EP2C35 chip in a 484-pin FineLine BGA package.
- The Cyclone II board provides a set of memory, including 128-Mbyte DDR SDRAM (which can operating at 167MHz), 1-Mbyte SRAM with 10ns High-speed access time, 32-Mbyte nand flash device for storage, a 16-Mbyte parallel nor flash device for storage, a 2-Mbyte parallel nor flash for config.
- A 10/100M Ethernet PHY/ MAC with RJ45.
- USB 2.0 Hi-speed controller with 2 HOST, one DEVICE and one OTG interfaces.
- An Altera SANTA CRUZ Daughter Card connector with 3.3V/5V compatibility.
- A 32bit 33/66M PCI edge with 3.3V and 5V compatibility.

For the AES algorithm evaluation, we are using flash memory, SRAM, DDR SDRAM, UART, timer, SPI and a 2.4GHz transceiver for transmission.

![Cyclone II development board](image)

**Figure 2.** Cyclone II development board

2.4. 2.4GHz Module

The nRF24L01 is a unique wireless solution for compact, battery operated applications with stringent requirements on battery lifetime and cost. The transceiver operates in the license free worldwide 2.4GHz ISM band. The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a crystal oscillator, a demodulator, modulator and Enhanced ShockBurst protocol engine. Output power, frequency channels, and protocol setup are easily programmable through a SPI interface. No external loop filters, resonators or VCO varactor diodes are required, only a low cost ±60ppm 16MHz crystal, matching circuitry and the antenna. Current consumption is very low, only 9.0mA at an output power of -6dBm and 12.3mA in RX mode. Built-in Power Down and Standby modes makes power saving easily realizable. The nRF24L01 module is shown in Fig.3.
The National Institute for Standards and Technology (NIST) ran a contest for the new Advanced Encryption Standard (AES) to replace the Data Encryption Standard (DES) which expired in 1998 for the United States beginning in January 1997. Fifteen submissions meeting basic criteria were received, not just from the US but from many other countries as well. In fifteen official candidate algorithms, five had been selected as finalists. In October 2000, Rijndael algorithm, designed by Joan Daemen and Vincent Rijmen, was announced as the winner of the five finalist algorithms. NIST chose Rijndael algorithm as the Advanced Encryption Standard (AES). This new AES has replaced the Data Encryption Standard (DES). AES is a symmetric block cipher with a fixed block length of 128 bit. The AES cipher treats the input 128-bit block as a group of 16 bytes organized in 4×4 matrix of bytes referred to as State. It supports key sizes of 128, 192 and 256 bits and consists of 10, 12 or 14 iteration rounds, respectively. This algorithm starts with initial transformation of state matrix followed by nine iteration of rounds. A round consists of four transformations: SubBytes (SB), ShiftRows (SR), MixColumns (MC) and followed by AddRoundKey (ARK). The last round is slightly different because MixColumns is not present. Figure 4 illustrates the AES cipher algorithm flow of encrypting one block of input data utilizing a given user key. In this paper we will focus on the 128-bit version with 10 rounds.
3. Implementation

3.1. The architecture

SOPC Builder tool is an exclusive Altera Quartus II design software tool that helps users create a complete system. Using the SOPC Builder, designers can use a variety of free components that are integrated in SOPC Builder or define their own peripherals or commands. Using powerful easy-to-use tools to integrate these cores into designs, the Altera FPGA platform provides a unique value to system designers.

Furthermore, SOPC Builder automatically generates interconnect logic, generates headers compliant with subscribed system features for successive software design, and creates a testbench to verify functionality, reducing the required time to set up a SOPC and enabling to construct and design in hours instead of weeks. When the hardware system change, SOPC Builder automatically updates these headers and provides software designers with an automatic configuration interface.

In our design, a system has been developed with SOPC Builder tool. It has the following elements, as shown in Figure 5:

- **Nios II processor**, which is referred to as a Central Processing Unit (CPU); The NIOS II configuration chosen is the NIOS II/fast, to provide the best performance to the processing unit.
- **AES encryption.** This module takes in 128-bit blocks of data, performs AES encryption with a user-entered 128-bit key. The results of this process are stored in the SDRAM.
- **UART Controller**, which performs nRF24L01 module initialization, receives and sends packets.
- **JTAG UART.** Communicates serial character streams between a host PC and an SOPC Builder system using the JTAG circuitry built into Altera.
- **Flash Interface**, which is a hardware component that facilitates the use of flash memory devices.
- The **DDR SDRAM controller**, which handles the complex aspects of using DDR SDRAM, such as initializing the memory devices, managing SDRAM banks, and keeping the devices refreshed at appropriate intervals.
- **Some input and output ports**, which is designed for test, timing and synchronization.
- **System Software**, which provides hardware core component control, peripheral control, operation control, data flow control, interface control, interrupt control, etc.

![Figure 5. System structure diagram](image)

3.2. System Implementation

Fig.6 shows the test platform for the system. It uses the nRF24L01 to transfer encoding data between RF node, which is composed of MCU, 2.4GHz module and the PC validation platform, and the NIOS II Development Board on 2.4GHz ISM band. The flow is also described as follows.
• The NIOS II development board stores a Plain Message, which is the data source to be encoded and encrypted, to be sent to Receiver RF node.

• The Nios II processor implements the AES encoding to encrypt the Plain Message to Ciphered Message in order to protect the eavesdropping before sending it off. The AES is more robust than DES but it is also more complex than DES, which demands more computational effort from an AES device. This complexity demands more computational effort from an AES device. To improve the effectiveness of AES algorithm is implemented in HW (hardware) in HDL language since the software implementation is unable to satisfy the higher throughput requirement.

• The system operates in the license free worldwide 2.4GHz ISM band. Transmit the Ciphered Message from the NIOS II development board to the Receiver RF node for validation.

• When the ciphered message arrives at receiving-end, the MCU included as part of a receiver RF node sends it to PC by uart and PC validation platform executes the decryption process to recover the plain message.

![System test platform](image)

**Figure 6. System test platform**

### 3.3. Design of Communication Software

![Wireless Data Receiving and Transmission Flow](image)

**Figure 7. Wireless Data Receiving and Transmission Flow**
Wireless data receiving and transmission flowchart is showed in Fig.7. In data transmission flowchart diagram, NIOS II development board configures nRF24L01 as PRX to receive REQ command 10h, if REQ command equal to 10h, starts data transmission mode. Then NIOS II development board configures nRF24L01 as PTX for sending data. When all required data is sent, nRF24L01 is set as PRX again waiting for REQ. In data receiving flowchart diagram, MCU development board configures nRF24L01 as PTX to send REQ command 10h for receiving ciphered message after powered on and initialized. Then nRF24L01 is configured as PRX waiting for receiving data. When all required data have been received and sent to PC validation platform for executing the decryption process to recover the plain message, the nRF24L01 enter into stand-by mode.

3.4. Design of communication software

In data transmission phase, the nRF24L01 is configured as PRX to receive REQ command 10h by NIOS II development board through a Serial Peripheral Interface (SPI). When it receives REQ command equal to 10h, it starts data transmission mode. NIOS II development board then configures nRF24L01 as PTX for sending data. When all required image data is sent, nRF24L01 is set as PRX again waiting for REQ.

In data receiving phase, the nRF24L01 which works in the fingerprint authentication center is configured as PTX to send REQ command 10h for receiving image after powered on and initialized. Then nRF24L01 is configured as PRX waiting for receiving data. When all required data have been received and sent to the fingerprint authentication center for executing the fingerprint authentication process and reacts the reference score to embedded fingerprint acquisition unit, the nRF24L01 enter into stand-by mode. The nRF24L01 driver contains functions to send/receive data via RF is prensent in Table 1.

<table>
<thead>
<tr>
<th>Function name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPI_RW()</td>
<td>SPI low level operation</td>
</tr>
<tr>
<td>SPI_Read_Reg()</td>
<td>Read register operation</td>
</tr>
<tr>
<td>SPI_Write_Reg()</td>
<td>Write register operation</td>
</tr>
<tr>
<td>SPI_Read_Buf()</td>
<td>Sequential read registers operation</td>
</tr>
<tr>
<td>SPI_Write_Buf()</td>
<td>Sequential write registers operation</td>
</tr>
<tr>
<td>NRFSetRXMode()</td>
<td>Configure nRF24L01 in Rx mode</td>
</tr>
<tr>
<td>NRFSetNPipeRXMode()</td>
<td>Configure nRF24L01 in Multi-channel Rx mode</td>
</tr>
<tr>
<td>NRFSetTxMode()</td>
<td>Configure nRF24L01 in Tx mode and send data on RF part</td>
</tr>
<tr>
<td>CheckACK()</td>
<td>Acknowledgement signal detection operation</td>
</tr>
<tr>
<td>NRFRevDate()</td>
<td>Receive data on RF part of nRF24L01</td>
</tr>
<tr>
<td>NRFRevNPipeDate()</td>
<td>Receive Multi-pipe data on RF part of nRF24L01</td>
</tr>
<tr>
<td>init_NRF24L01()</td>
<td>Init nRF24L01 operation</td>
</tr>
</tbody>
</table>

4. Conclusion

Cryptographic algorithms play a central role in modern cryptosystems, providing encryption for millions of sensitive financial, government, and private transactions daily. FPGAs provide a relatively low-cost, high-performance method of implementing cryptographic algorithms. Also, as FPGAs grow in size, they allow a designer to use a single reconfigurable platform to instantiate both the processor and the required dedicated logic. In this paper we present the AES coprocessor implementation for 2.4GHz Wireless Network evaluation based on SOPC technology. Results show that the presented architecture delivers good price/performance and lowers technical development hurdles.

7. References


