IMPACT: International Journal of Research in Engineering & Technology (IMPACT: IJRET)
ISSN (E): 2321-8843; ISSN (P): 2347-4599
Vol. 4, Issue 2, Feb. 2016, 7-24
© Impact Journals

FPGA IMPLEMENTATION OF 30GBPS SECURITY MODULE FOR GPON SYSTEMS

KALYANI PEDDINA, SUNIL KUMAR DASARI & SATISH KUMAR REDDY M V

Department of Electronics and Communications, GITAM School of Technology, GITAM University

Bangalore, Karnataka, India

ABSTRACT

The need for privacy has become a high priority for both governments and civilians desiring protection from signal and data interception. Widespread use of personal communications devices has only increased demand for a level of security on previously insecure communications. Both DES (Data Encryption Standard) and AES are defined as symmetric key block ciphers [1], with the main difference being the bit length of the key (56 bit for DES).

These symmetric-key encryption schemes use the same key for both the sender and receiver, and as a result eliminate the need for the verification server needed in public keying. Symmetric keying lends itself to work independently of an open network and in turn a higher level of system interoperability.

Ever since DES [2] was phased out in 2001 and its successor, the Advanced Encryption Standard (also known as Rijndael) took its place, various AES implementations have been proposed both in software and hardware. This paper presents low cost and low power hardware architecture for the Advanced Encryption Standard (AES). In 1997, the National Institute of Standards and Technology promoted worldwide research into a replacement for DES, or the widely accepted Data Encryption Standard. In this brief, we present an efficient and cost-effective AES co-processor design [3]. To minimize cost, focusing on efficiency reduced overall hardware complexity.

A VHDL hardware implementation [4] is also presented, utilizing a field programmable gate array (FPGA) as a prototyping platform. In this architecture, the main priority was not to increase throughput or decrease processing time but to balance these factors in order to minimize cost. A focus on low power and cost allows for scaling of the architecture towards vulnerable portable communications devices in consumer and military applications such as cellular phones, PDAs, digital radios, pagers, and similar lower speed communication embedded systems.

KEYWORDS: Module for GPON, Security

INTRODUCTION

Motivation

Cryptography is best known as a way of keeping the contents of a message secret. Better cryptographic techniques will have to be developed to protect business transactions. DES was one of the encryption standard, with block size 64-bits and key size 56-bits. But these are very small and slower performance. Next standard is 3DES with 112 and 168 bits key size and 64-bits block size. But it also gives slower performance.AES was developed in 2001 with 128-bit block size and 128,192 and 256-bits key sizes. It is Stronger & faster than triple DES ,Active life of 20-30 years, Provide full specification & design details.In order to achieve high performance and to provide the more security for data transmission in GPON security module , AES encryption standard can be used.

GOAL

In this report, It presents an implementation of very high speed security module for GPON on FPGA. The aim of this dissertation is to design and implement a suitable architecture for the GPON security module, because GPONs are attractive for cost-effective delivery of high-bandwidth data directly to building, curb, and home. This creates a strong requirement for access network to be trustworthy, secure, and reliable. The performance of our design is well suitable for encryption applications of GPON systems. However, only AES with counter mode (CTR-AES) can be used for GPON payload encryption. In this project, we present a GPON security module using CTR-AES algorithm which is implemented by a 3 stage full-pipelined architecture for area and performance optimization.

LITERATURE SURVEY

Passive Optical Network

Descriptions

A passive optical network (PON) is a point-to-multipoint, fiber to the premises network architecture in which unpowered optical splitters utilizing Brewster's angle principles are used to enable a single optical fiber to serve multiple premises, typically 32-128.

A PON consists of an optical line terminal (OLT) at the service provider's central office and a number of optical network units (ONUs) near end users.

A PON configuration reduces the amount of fiber and central office equipment required compared with point to point architectures.

A passive optical network is a form of fiber-optic access network. Downstream signals are broadcast to each premises sharing a single fiber. Encryption is used to prevent eavesdropping. Upstream signals are combined using a multiple access protocol, usually time division multiple access (TDMA).

The basic principle of a PON network [5] is to share the Central Office equipment (Optical Line Terminal or OLT) and the feeder fiber among as many end units (Optical Network Termination or ONT) as possible within the physical and bandwidth constraints. Since this solution requires less fiber layout in order to cover a specific area, as well as less costly optical interfaces at the CO (one optical interface serves the entire network), the solution offered enables high speed optical connections for businesses or residential units in scenarios that could not be served in an economical manner using traditional point-to-point or ring architectures.

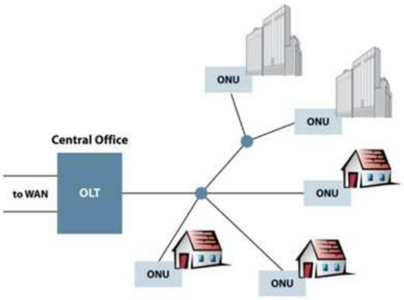


Figure 1: Basic PON Network

STANDARDS

ITU-T G.983

- **APON** (ATM Passive Optical Network) : This was the first Passive optical network
- Standard. It was used primarily for business applications, and was based on ATM.
- **BPON** (Broadband PON): It is a standard based on APON. It adds support for WDM
- Dynamic and higher upstream bandwidth allocation, and survivability. It also created a
- Standard management interface, called OMCI, between the OLT and ONU/ONT,
- Enabling mixed-vendor networks.

IEEE 802.3ah

EPON or GEPON (Ethernet PON): It is an IEEE/EFM standard for using Ethernet for packet data. 802.3ah is now part of the IEEE 802.3 standard. There are currently over 25 million installed EPON subscribers. Commercial upgrade capability to 10G EPON will become available in 2010 (see IEEE 802.3av, 10G-EPON).

ITU-T G.984

GPON (Gigabit PON): It is an evolution of the BPON standard [5]. It supports higher rates, enhanced security, and choice of Layer 2 protocol (ATM, GEM, and Ethernet). In early 2008, Verizon having installed over 800 thousand lines by mid year. British Telecom, mobily-saudiarabia, Etisalat-uae, and AT&T are in advanced trials.

IEEE 802.3AV

10G-EPON (10 Gigabit Ethernet PON): It is an IEEE Task Force for 10Gbit/s, backward compatible with 802.3ah EPON. 10GEPON will use separate wavelengths for 10G and 1G downstream. 802.3av will continue to use a single wavelength for both 10G and 1G upstream with TDMA separation. The 802.3av task force has concluded with the .3av inclusion in the IEEE 802.3 standard. Commercial 10G EPON equipment is expected in 2010.

SCTE IPS910

RFoG (RFoverGlass): It is an SCTE Interface Practices Subcommittee standard in development for Point to Multipoint (P2MP) operations that has a proposed wavelength plan compatible with data PON solutions including EPON,GEPON and 10G-EPON. RFoG offers an FTTH PON like architecture for MSOs without having to select or deploy a PON technology.

NETWORK ELEMENTS

A PON takes advantage of wavelength division multiplexing (WDM), using one wavelength for downstream traffic and another for upstream traffic on a single [Non-dispersion shifted fiber] (ITU-T G.652). BPON, EPON, GEPON, and GPON have the same basic wavelength plan and use the 1490 nanometer (nm) wavelength for downstream traffic and 1310 nm wavelength for upstream traffic. 1550 nm is reserved for optional overlay services, typically RF (analog) video.

A PON consists of a central office node, called an optical line terminal (OLT), one or more user nodes, called optical network units (ONUs) or optical network terminals (ONTs), and the fibers and splitters between them, called the optical distribution network (ODN). ONT is an ITU-T term, whereas ONU is an IEEE term. In Multiple Tenant Units, the ONT may be bridged to a customer premise device within the individual dwelling unit using technologies such as Ethernet over twisted pair, G.hn (a high-speed ITU-T standard that can operate over any existing home wiring - power lines, phone lines and coaxial cables) or DSL. An ONT is a device that terminates the PON and presents customer service interfaces to the user. Some ONUs implement a separate subscriber unit to provide services such as telephony, Ethernet data, or video.

COMPARISON

Table 1: Comparison of Pons

	B-PON	GE-PON	G-PON
Standard	ITU-T G.983	IEEE 802.3ah	ITU-T G.984
Downstream data rate	600 Mbit/s	1 Gbit/s	2.4 Gbit/s
Upstream data rate	150 Mbit/s	1 Gbit/s	1.2 Gbit/s
Transmission Format	ATM	Ethernet	Ethernet TDM +ATM

GIGA-BIT PASSIVE OPTICAL NETWORK

Descriptions

GPON carries a two-fold promise of both higher bit rates and higher efficiency when carrying multiple services over the PON. When initiated, the GPON was intended as a complete bottom-up reconsideration of PON applications and requirements and, as such, laid the foundation for new solutions that are not based upon the previous APON standard.

While much of the functionality that is not directly related to the PON is preserved, such as OAM messages, DBA, etc., GPON is based [5] upon a completely new Transmission Convergence (TC) layer. The FSAN has recently selected the proposal put forward by Flex Light and numerous additional vendors, for a frame based protocol using GEM for service mapping, as the next GPON protocol.

Starting with the GPON work, the following objectives were put forward:

- Scalable framing structure for 622Mbps to 2.5Gbps, as well as asymmetric bit rates support.
- Exceptionally high bandwidth utilization/efficiency for any type of service
- GPON Encapsulation Method (GEM) encapsulation of any type of service (both TDM and packet) into 125 µ sec Periodic frames.
- High efficiency with no overhead transport of native TDM traffic required.
- Dynamic Allocation of upstream bandwidth via bandwidth maps (pointers) for each ONT.

- GPON Encapsulation Method (GEM) is based on the ITU GFP standard (ITU-T G.7041), with some minor modifications to make it optimized for PON topologies. GEM provides a generic mechanism to adapt traffic from higher layer client signals over a transport network.
- Summarizing the design choices for the GPON protocol, the following items can be mentioned:
- Frame-based, multi service (ATM, TDM, Data) transport over PON.
- Upstream bandwidth allocation mechanism via slot assignments through pointers.
- Support for asymmetric line rate operation. 2.488 Gbit/s downstream and 1.244 Gbit/s upstream rates.
- Line coding will be non-return-to-zero (NRZ) with scrambling.
- Fragmentation and concatenation of data frames for bandwidth efficiency.
- Upstream burst mode preamble, including clock and data recovery (CDR), will not be long

GPON DOWN STREAM FRAME FORMAT

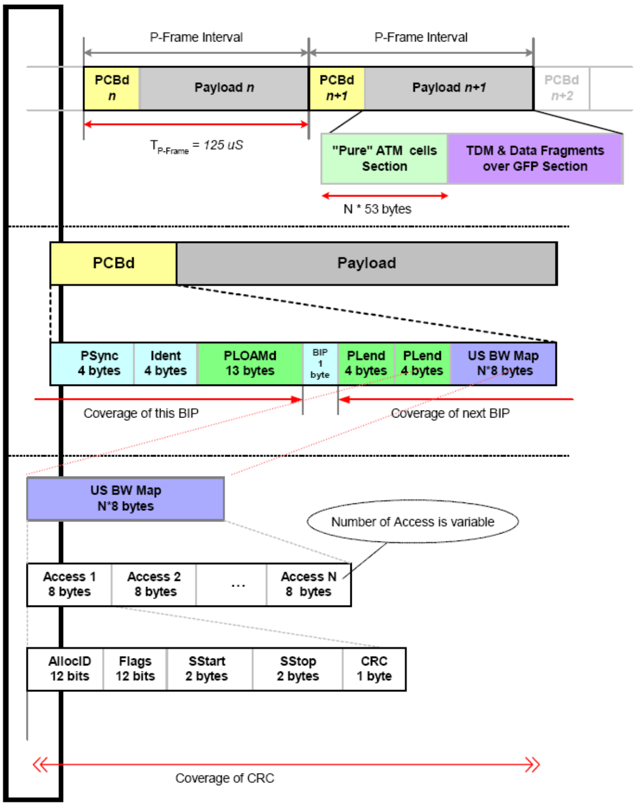
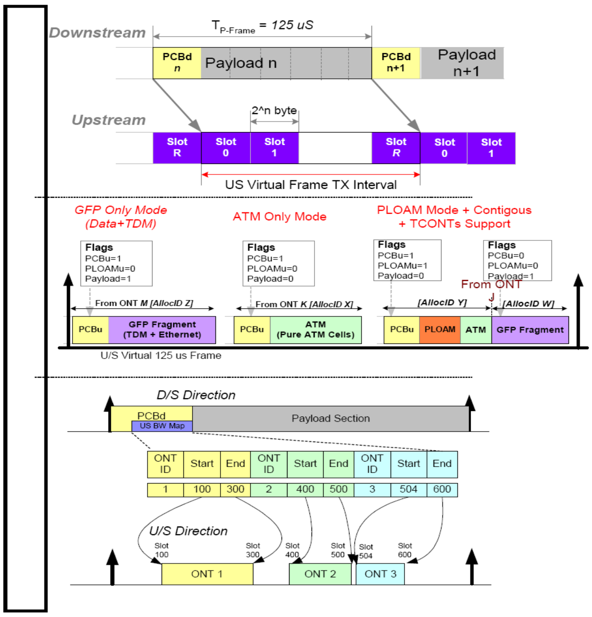


Figure 2: GPON Down Stream Frame Format

GPON UP STREAM FRAME FORMAT



ONT ID	Start	End	ONT ID	Start	End	ONT ID	Start	End
1	100	300	2	400	500	3	504	600

Figure 3: Gpon Upstream Frame Format

THE ARCHITECTURE OF THE GPON SECURITY MODULE

Architecture of Security Module

The GPON security module is implemented to guarantee a secure communication in Tx/Rx link of GPON [5]. Using the module, the transmission data are ensured to be confidentiality, integrity, and origin authenticity of each frame sent and received by the OLT (Optical Line Termination) / ONT (Optical Network Termination).

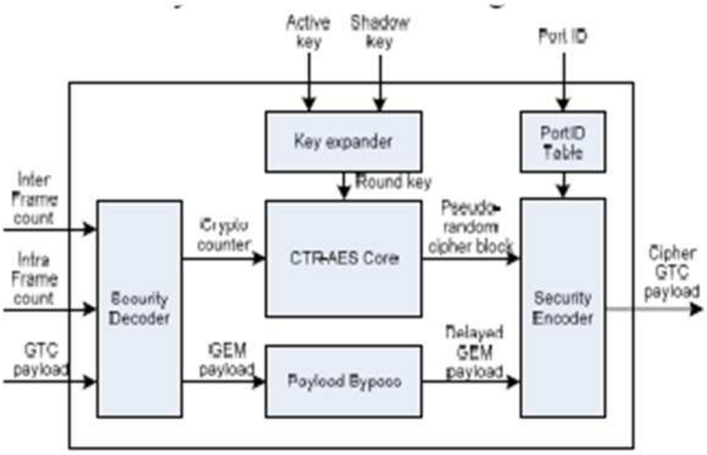


Figure 4: Architecture of GPON Security Module

PORT-ID TABLE

It is implemented as 4K 12-bit registers to store the port identifier. Only frames with the appropriate Port-ID are encrypted by CTR-AES core.

A port number is a way to identify a specific process to which an Internet or other network message is to be forwarded when it arrives at a server. For the Transmission Control Protocol and the User Datagram Protocol, a port number is a 16-bit integer that is put in the header appended to a message unit. Some services or processes have conventionally assigned permanent port numbers. These are known as well-known port numbers. In other cases, a port number is assigned temporarily (for the duration of the request and its completion) from a range of assigned port numbers. This is called an ephemeral port number.

SECURITY DECODER

It generates Crypto counter with the format: (Inter Frame Count [19:0] & Intra Frame Count [15:0]) & (Inter Frame Count [29:0] & Intra Frame Count [15:0]) & (Inter Frame Count [29:0] & Intra Frame Count [15:0]). It also registers 128-bit GTC Payload for the Payload Bypass.

In digital electronics, a decoder can take the form of a multiple-input, multiple-output logic circuit that converts coded inputs into coded outputs, where the input and output codes are different. e.g. n-to-2^n, binary-coded decimal decoders.

KEY EXPANDER

It restores the initial key and generates round keys for CTR-AES from 128-bit key input. The total bit number of round keys is 1408 = 128*(10+1). The shadow key is used if the OLT require key exchange. The ONT responds by generating, storing and sending a new key. When the new key is transferred successfully to OLT, both the OLT and ONU (Optical Network Unit) begin using the new key at precisely the same frame boundary.

CTR-AES CORE

It is the same process of AES algorithm except input values which is crypto counter. The crypto counter increases at every 128-bit data block. 128-bit input blocks are transformed into 128-bit pseudorandom cipher blocks.

PAYLOAD BYPASS

It delivers the insecure payload without an authentication encryption. It has the same delay as encryption time to synchronize with the cipher GEM payload at the output.

SECURITY ENCODER

It multiplexes the cipher GEM (G-PON Encapsulation Method) Payloads from Bypass GEM Payload and Encrypted GEM Payload depending whether security function is enabled. For the authentic frames, the encoder performs XORed 128bits Pseudorandom Cipher block with delayed GEM payload to generate cipher GEM payload.

The AES algorithm in GPON security module uses counter mode to encrypt data . In counter mode encryption, the forward cipher function is invoked on each counter blocks, and the resulting output blocks are exclusive-ORed with the corresponding plaintext blocks to produce the ciphertext blocks. The forward cipher function is used in both CTR decryption and CTR encryption. Therefore, only one hardware implementation is used for both encryption and decryption. The XORed operation is executed in security encoder block.

ARCHITECTURE IMPLEMENTATION OF CTR-AES MODULE

The Fullpipelined Architecture for Aes Algorithm

In order to achieve very high throughput, we apply pipeline technique both for outer round and inner round of AES architecture [17]. For outer round pipelining, the pipeline registers are placed between the data path instances of each round. For the inner round pipelining, we decompose four processes SubByte, ShiftRow, MixColumn and AddRoundKey into sub-pipelined [18] stages with equivalent delay.

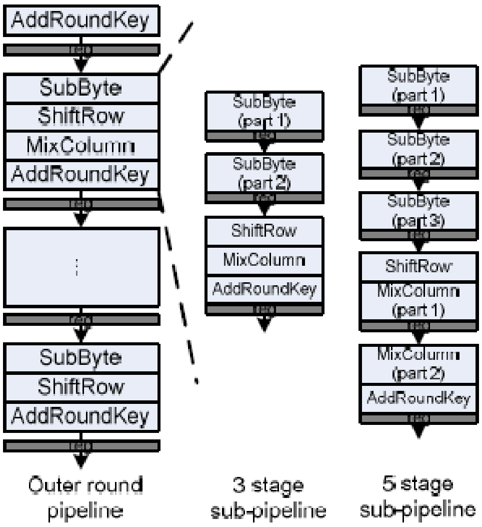


Figure 5: Full Pipelined Architecture

Among round processes of AES algorithm, the SubByte phase has the most delay. Therefore, the number of sub-stages of this block is more than that of other phases. We implemented two full-pipelined architectures which have 2-stage sub-pipeline and 5-stage sub-pipeline for each round process. Thus, the SubByte block has to be decomposed into 2 stages and 3 stages, respectively. We can achieve a very high throughput when using 5-stage sub-pipelined for AES architecture [19].

SUBBYTE TRANSFORMATION

In the SubByte transformation (Sbox) [20], the input is considered as an element of GF(2^8). First, the multiplicative inverse in GF(2^8) is calculated. Then, an affine transformation over GF(2^8) is applied. The implementation of a SBox can be done by a look-up table, but it consumes much resource. Nevertheless, we can implement a SBox using Galios Field operations . Field arithmetic GF(2^4) is used instead of GF(2^8) to optimize area. In this architecture, the input values is mapped to two elements of GF(2^4). Then, the multiplicative inverse is calculated using GF(2^4) operation. Next, the two GF(2^4) elements are inverse mapped to one element in GF(2^8). Last, the affine transformation is performed. Although the composite field implementation of Sbox is very efficient in area, it suffers from a long critical path. To overcome this drawback, further pipelining can be used. By using the 2-stage pipelined architecture with three 8-bit registers, the critical path is broken in half. To reduce more path delay, the 3-stage pipelined architecture can be also applied

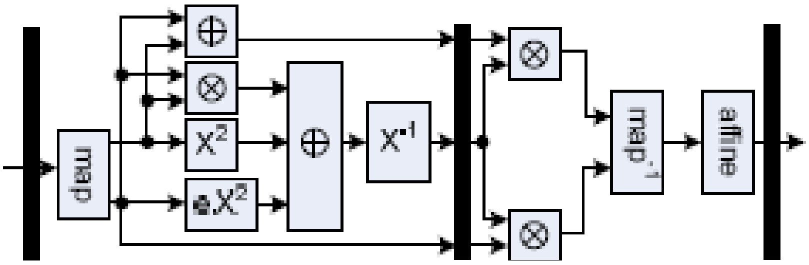


Figure 6: 2-Stage Pipelined S-Box Using GF (2$^{8)}$ Operation

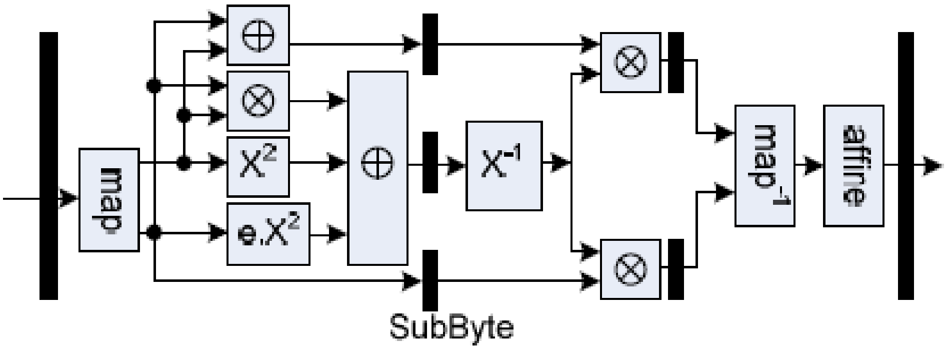
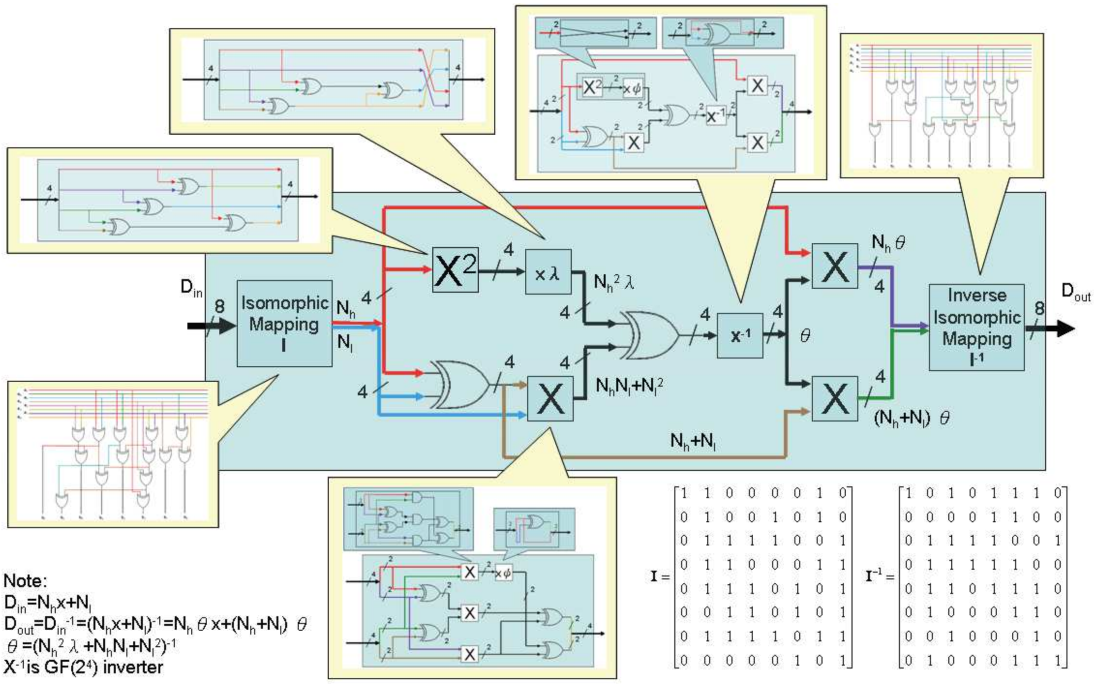


Figure 7: 3-Stage Pipelined S Box Using GF(2$^{8)}$ Operation



Note:
$D_{in}=N_hx+N_l$
$D_{out}=D_{in}^{-1}=(N_hx+N_l)^{-1}=N_h\,\theta\,x+(N_h+N_l)\,\theta$
$\theta=(N_h^2\,\lambda+N_hN_l+N_l^2)^{-1}$
$X^{-1}$is GF(2^4) inverter

$$\mathbf{I}=\begin{bmatrix}1&1&0&0&0&0&1&0\\0&1&0&0&1&0&1&0\\0&1&1&1&1&0&0&1\\0&1&1&0&0&0&1&1\\0&1&1&1&0&1&0&1\\0&0&1&1&0&1&0&1\\0&1&1&1&1&0&1&1\\0&0&0&0&0&1&0&1\end{bmatrix}\quad \mathbf{I}^{-1}=\begin{bmatrix}1&0&1&0&1&1&1&0\\0&0&0&0&1&1&0&0\\0&1&1&1&1&0&0&1\\0&1&1&1&1&1&0&0\\0&1&1&0&1&1&1&0\\0&1&0&0&0&1&1&0\\0&0&1&0&0&0&1&0\\0&1&0&0&0&1&1&1\end{bmatrix}$$

Figure 8: Architecture of GF (2^{8}) Inverter

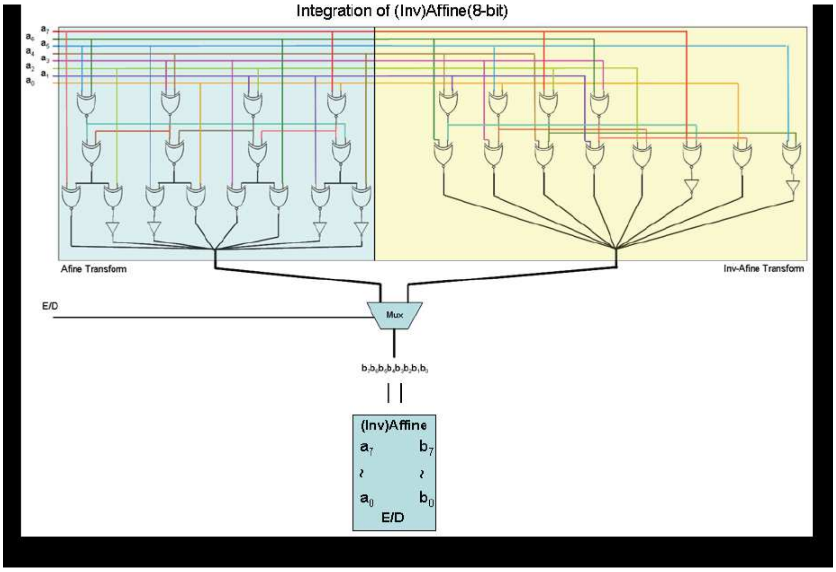


Figure 9: Circuit of the Integration of Affine and Inverse Affine Transformations

MIX COLUMN

In MixColumn transformation, the columns of the State are considered as polynomials over GF(28) and multiplied modulo x4 + 1 with a fixed polynomial c(x) = ‘03’ x3 + ‘01’ x2 + ‘01’ x + ‘02’. In direct form, the MixColumn transformation can be expressed as

$$\begin{cases} s'_{0,c} = (\{02\} \bullet s_{0,c}) \oplus (\{03\} \bullet s_{1,c}) \oplus s_{2,c} \oplus s_{3,c} \\ s'_{1,c} = s_{0,c} \oplus (\{02\} \bullet s_{1,c}) \oplus (\{03\} \bullet s_{2,c}) \oplus s_{3,c} \\ s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus (\{02\} \bullet s_{2,c}) \oplus (\{03\} \bullet s_{3,c}) \\ s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c}) \end{cases}$$

In our implementation, we also use substructure sharing techniques to implement an efficient hardware for MixColumn transformation. To apply this technique, the equation (1) should be rewritten in an efficient way as

$$\begin{cases} s'_{0,c} = \{02\} \bullet (s_{0,c} \oplus s_{1,c}) \oplus s_{1,c} \oplus (s_{2,c} \oplus s_{3,c}) \\ s'_{1,c} = \{02\} \bullet (s_{1,c} \oplus s_{2,c}) \oplus s_{0,c} \oplus (s_{2,c} \oplus s_{3,c}) \\ s'_{2,c} = \{02\} \bullet (s_{2,c} \oplus s_{3,c}) \oplus s_{3,c} \oplus (s_{0,c} \oplus s_{1,c}) \\ s'_{3,c} = \{02\} \bullet (s_{3,c} \oplus s_{0,c}) \oplus s_{2,c} \oplus (s_{0,c} \oplus s_{1,c}) \end{cases}$$

The equation for MixColumn transformation is now more symmetrical, and we can apply substructure sharing to optimize area for hardware implementation. The {02} constant multiplication is computed by the function denoted by a = xtime(b). The xtime() function can be implemented at the byte level as a left shift anda subsequent conditional bitwise XOR with {1b} if the most significant of input byte is one (b7 = 1). The xtime() block can be implemented by 3 2-bit XOR gate. By using efficient architecture of xtime() and applying XOR-sharing, the MixColumn transformation can be implemented.

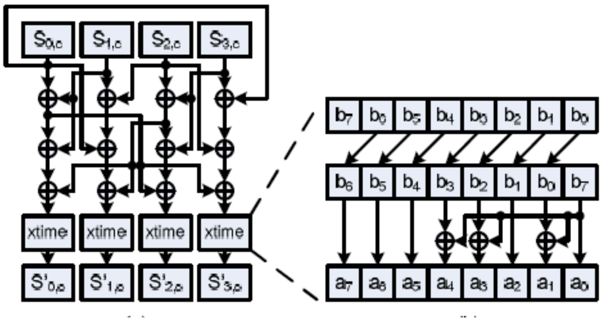


Figure 10: (A) Efficient Implementation of Mixcolumn

Implementation of xtime() function

The total number of gate counts for MixColumn transformation is 324, which includes 108 2-bit XOR gates (each XOR gate contains 3 gates).

KEY-EXPANDER

The Key Expansion routine generates a total of 11 round keys from an initial key in 128-bit AES algorithm. For pipelined AES architecture, all round keys must be available at the same time. Therefore, some researchers implemented a key expansion routine to compute a round key, and duplicate this hardware 10 times for total 10 rounds [16]. These architectures can calculate all round keys at the same time, but they consume much area.

Some other researchers has proposed key expander that can operate in on-the-fly manner [16]. The data encryption and the key expansion can start simultaneously. Inherited from that architecture, we implement an area-efficient key expander which also can compute round key in on-the-fly manner. In order to operate synchronously with the sub-pipelined round process, the key expander is divided into r sub-stages. We use 11 registers to store 11 round keys. It is different from the architecture of the key expansion in which the author used r sets of registers all round keys and temporary values for sub-pipelined stage. By this scheme, we can reduce more area than the previous architecture. The sub-pipelined architecture for on-the-fly key expander with 3 substage (r=3) since round keys are generated on the fly, the number of sub-pipelined stages for key expansion must be the same with the number of encryption sub-stages. After r clock cycles, a new round key is generated, so all the round keys are available after (r×Nr) +1 clock cycles.

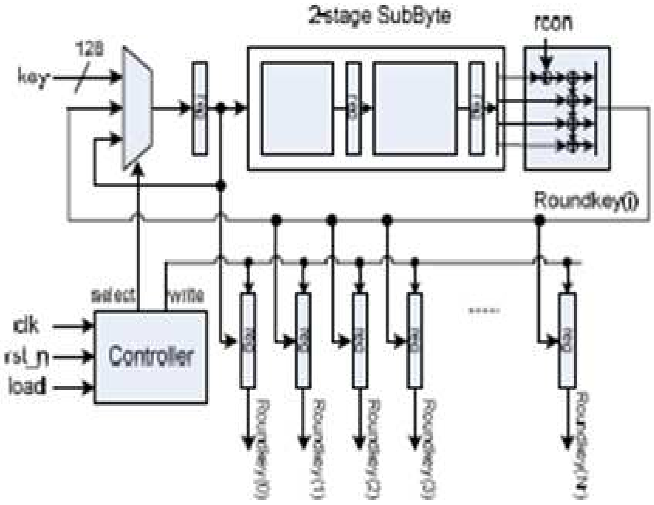


Figure 11: Architecture of on the Fly Key Expander

SIMULATION RESULTS

Gpon Security Module Top Simulation Results

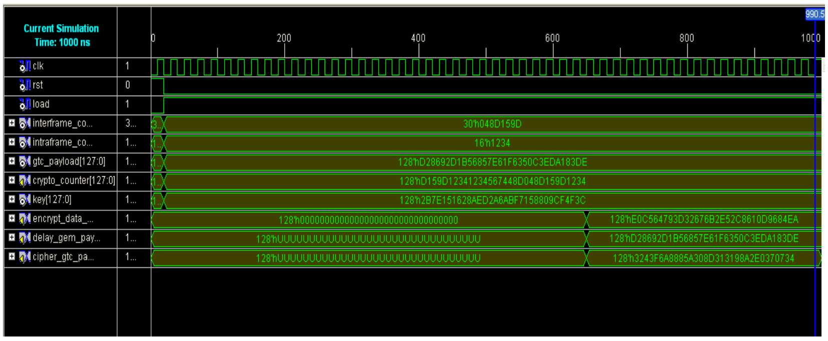

Current Simulation
Time: 1000 ns
990.5
0
200
400
600
800
1000
clk 1
rst 0
load 1
interframe_co... 3... 30'h048D159D
intraframe_co... 1... 16'h1234
gtc_payload[127:0] 1... 128'hD28692D1B56857E61F6350C3EDA183DE
crypto_counter[127:0] 1... 128'hD159D12341234567448D048D159D1234
key[127:0] 1... 128'h2B7E151628AED2A6ABF7158809CF4F3C
encrypt_data_... 1... 128'h00000000000000000000000000000000 128'hE0C564793D32676B2E52C8610D9684EA
delay_gem_pay... 1... 128'hUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU 128'hD28692D1B56857E61F6350C3EDA183DE
cipher_gtc_pa... 1... 128'hUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU 128'h3243F6A8885A308D313198A2E0370734

Figure 12: Waveform of Top Level GPON Security Module

The above figure gives the top level waveform for GPON security module. The main aim of this dissertation is providing the high performance and more security to GPON security module. For this GPON security module, the inputs are Clk, rst, load, interframe_count[29:0], intraframe_count[15:0], gtc_payload[127:0], key[127:0], and the output is Cipher_gtc_payload[127:0].

SECURITY DECODER SIMULATION RESULT

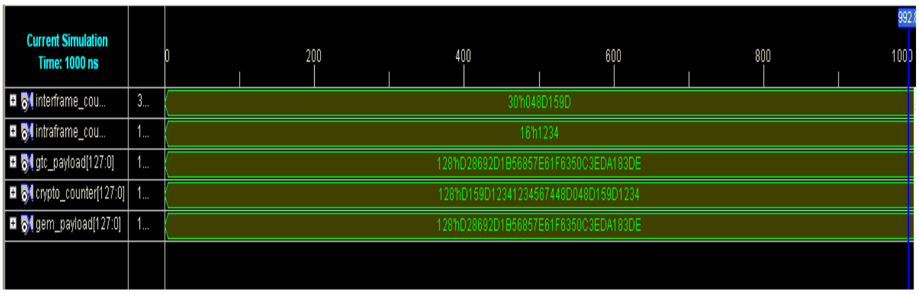

Current Simulation
Time: 1000 ns
992.
0
200
400
600
800
1000
interframe_cou... 3... 30'h048D159D
intraframe_cou... 1... 16'h1234
gtc_payload[127:0] 1... 128'hD28692D1B56857E61F6350C3EDA183DE
crypto_counter[127:0] 1... 128'hD159D12341234567448D048D159D1234
gem_payload[127:0] 1... 128'hD28692D1B56857E61F6350C3EDA183DE

Figure 13: Waveform of Security Decoder

For Security Decoder, given input format is Inter Frame Count [29:0] & Intra Frame Count [15:0] and generates Crypto counter with the format: (Inter Frame Count [19:0] & Intra Frame Count [15:0]) & (Inter Frame Count [29:0] & Intra Frame Count [15:0]) & (Inter Frame Count [29:0] & Intra Frame Count [15:0]). It also registers 128-bit GTC Payload for the Payload Bypass to generate the gem_payload[127:0].

AES CORE SIMULATION RESULTS

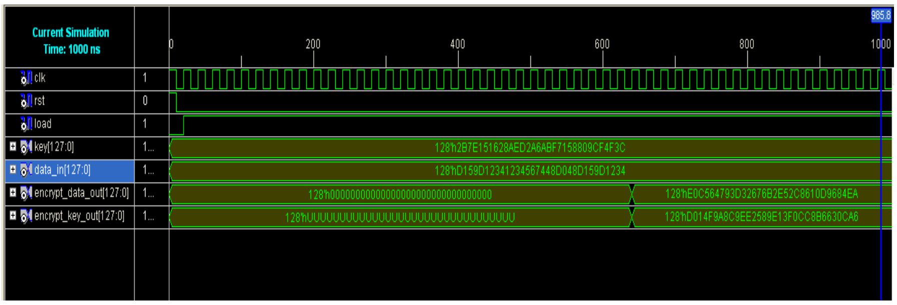

Current Simulation
Time: 1000 ns
985.8
0
200
400
600
800
1000
clk
1
rst
0
load
1
key[127:0]
1...
128'h2B7E151628AED2A6ABF7158809CF4F3C
data_in[127:0]
1...
128'hD159D12341234567448D048D159D1234
encrypt_data_out[127:0]
1...
128'h00000000000000000000000000000000
128'hE0C564793D32676B2E52C8610D9684EA
encrypt_key_out[127:0]
1...
128'hUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
128'hD014F9A8C9EE2589E13F0CC8B6630CA6

Figure 14: Waveform of AES Core

CTR AES core, is the same process of AES algorithm except input values which is crypto counter. The crypto counter increases at every 128-bit data block. 128-bit input blocks are transformed into 128-bit pseudorandom cipher blocks. For this AES core clk, rst, load will be used in the key expander to control the keys. In this key[127:0] and data_in[127:0] which is crypto counter generated from security decoder. The output of AES core is Encrypt_data_out[127:0],encrypt_key_out[127:0].

PAY LOAD BYPASS SIMULATION RESULTS

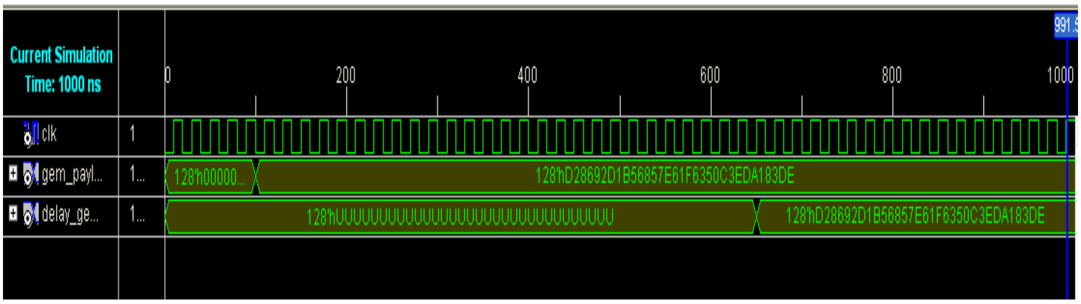
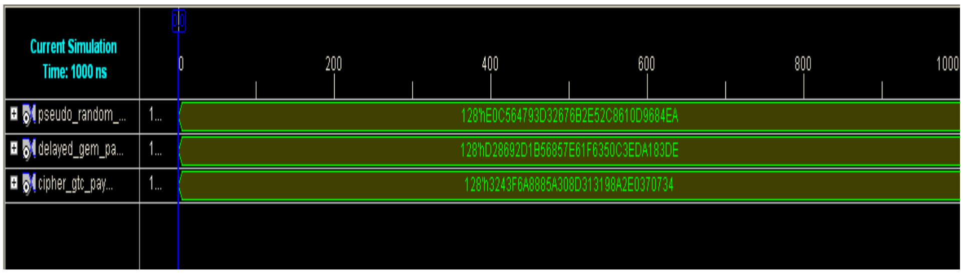

Current Simulation
Time: 1000 ns
991.5
0
200
400
600
800
1000
clk
1
gem_payl...
1...
128'h00000...
128'hD28692D1B56857E61F6350C3EDA183DE
delay_ge...
1...
128'hUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
128'hD28692D1B56857E61F6350C3EDA183DE

Figure 15: Waveform of Payload Bypass.

Payload bypass delivers the insecure payload without an authentication encryption. It has the same delay as encryption time to synchronize with the cipher GEM payload at the output. In order to provide the same delay as encryption time for the gem_payload [127:0], clk is used as the one of the input to the Payload Bypass. The output of the payload bypass is delay_gem_payload[127:0].

SECURITY ENCODER SIMULATION RESULT



Current Simulation Time: 1000 ns

```
pseudo_random_...  1...  128'hE0C564793D32676B2E52C8610D9684EA
delayed_gem_pa...  1...  128'hD28692D1B56857E61F6350C3EDA183DE
cipher_gtc_pay...  1...  128'h3243F6A8885A308D313198A2E0370734
```

Figure 16: Waveform of Security Encoder

In the waveform of security encoder, inputs are pseudo_random_data[127:0] which is output of AES core and delayed_gem_payload[127:0] which is coming from payload bypass. Security encoder performs XORed 128bits Pseudorandom Cipher block with delayed GEM payload to generate cipher GTC payload. The output of security encoder is cipher_gtc_payload[127:0]. This is final output for GPON transmitter.

RTL SCHEMATIC DIAGRAM

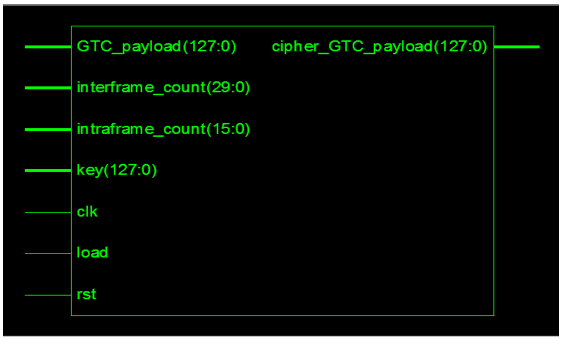
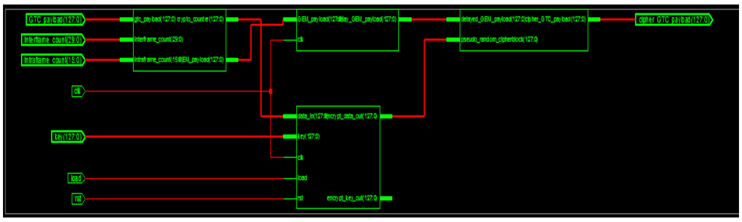


Figure 17: RTL Schematic Diagram

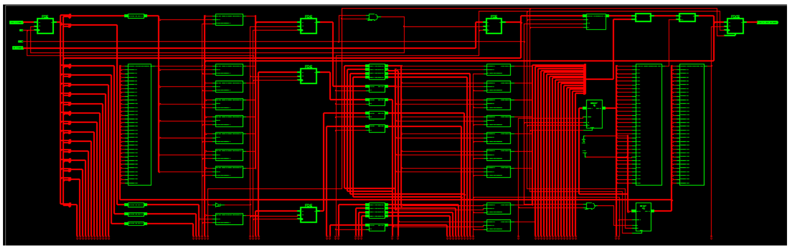
RTL SCHEMATIC INTERNAL DIAGRAM



GTC payload(127:0)
interframe count(29:0)
intraframe count(15:0)
clk
key(127:0)
load
rst
gtc_payload(127:0) crypto_counter(127:0)
interframe_count(29:0)
intraframe_count(15:0) GEM_payload(127:0)
GEM_payload(127:0) delay_GEM_payload(127:0)
clk
data_in(127:0) encrypt_data_out(127:0)
key(127:0)
clk
load
rst
encrypt_key_out(127:0)
delayed_GEM_payload(127:0) cipher_GTC_payload(127:0)
pseudo_random_cipherblock(127:0)
cipher GTC payload(127:0)

Figure 18: RTL Schematic Internal Diagram

AES INTERNAL RTL



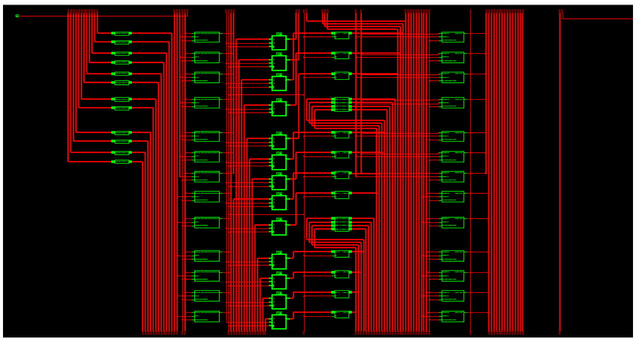


Figure 19: AES Internal RTL

CONCLUSION AND FUTURE WORK

CONCLUSIONS

Cryptography is best known as a way of keeping the contents of a message secret. Better cryptographic techniques will have to be developed to protect business transactions. DES was one of the encryption standard, with block size 64-bits and key size 56-bits. But these are very small and slower performance. Next standard is 3DES with 112 and 168 bits key size and 64-bits block size. But it also gives slower performance.AES was developed in 2001 with 128-bit block size and 128,192 and 256-bits key sizes. It is Stronger & faster than triple DES ,Active life of 20-30 years, Provide full specification & design details.In order to achieve high performance and to provide the more security for data transmission in GPON security module , AES encryption standard is used.

FPGA implementation of the high speed GPON security module using counter mode AES algorithm was done. A new AES VLSI architecture is developed to reduce the cost using a minimalist bit serial approach. This design has three main efficient features: composite field arithmetic SubByte, area-efficient MixColumn, and on-the-fly sub-pipelined Key-Expander and all the operations are integrated into a simple architecture in which all the operations are concurrently done by using the same blocks using different control signals, which is very important develop an architecture to minimize the cost of the implementation. By using these improvement features, this design has optimal area and maximum throughput. This architecture can be used in many military, industrial, and commercial applications that require compactness and low cost.

FUTURE SCOPE

In this architecture we developed the encryption output using 128-bit key length. To get more secure transmission of data it is possible to go with 192 key, 256 key length in which more number of rounds are to get the encrypted output which very high secure because generally this type of architectures are used in defense applications to transmit very high secure data. And also we can implement this security module with 5 stages, to get high through put data.

REFERNCES

1. J. Daemen, L. R. Knudsen, and V. Rijmen, “The block cipher square”, in *Fast Software Encryption*, E. Biham, Ed. 1997, vol. 1267 of *LNCS*, pp. 149–165, Springer-Verlag.
2. National Institute of Standards and Technology (NIST), *Data Encryption Standard (DES)*, National Technical Information Service, Springfield, VA 22161, Oct. 1999.
3. C.-P. Su, T. -F. Lin, C. -T. Huang, and C.-W. Wu, "A high-throughput lowcost AES processor", IEEE Communications Magazine, vol. 41, no. 12, pp. 86-91, Dec. 2003.
4. X. Zhang and K. Parhi, "High-speed VLSI architecture for the AES algorithm", IEEE Trans. On VLSI Systems, vol. 12, no. 9, pp. 957-967, 2004.
5. “Gigabit-capable Passive Optical Networks (G-PON): Transmission convergence layer specification”, ITU-T G.984.3 Amendment 1, July. 2005.
6. W. Stallings, *Cryptography and Network Security: Principles and Practice. 3rd ed.*, Prentice-Hall Inc., Upper Saddle River, N.J., 2003.
7. National Institute of Standards and Technology (NIST), *Advanced Encryption Standard (AES)*, National Technical Information Service, Springfield, VA 22161, Nov. 2001.
8. J. Daemen and V. Rijmen, AES Proposal: Rijndael, AES algorithm submission, Sept. 1999. (http://www.nist.gov/CryptoToolkit)
9. Shuenn-Shyang Wang, Wan-Sheng Ni, “An efficient FPGA implementation of advanced encryption standard algorithm”, Proceedings of the International Symposium on Circuits and
10. Systems, vol. 2, pp. 597-600, May 2004.
11. Jae-Gon Lee, Woong Hwangbo, Seonpil Kim, Chong-Min Kyung,Top-down implementation of pipelined AES cipher and its verification with FPGA-based simulation accelerator”, Proceedings of 6th International Conference on ASIC , pp. 68-72, Oct. 2005.
12. Morris Dworkin, “Recommendation for Block Cipher Modes of Operation”, NIST Special Publication, http://csrc.nist.gov/ CryptoToolkit/modes/, 2001.
13. Yongzhi Fu, Lin Hao, Xuejie Zhang, Rujin Yang, “Design of an extremely high performance counter mode AES reconfigurable processor”, Second International Conference on Embedded Software and Systems, Dec. 2005.
14. N. Ferguson, R. Schroeppel, and D. Whiting, “A simple algebraic representation of rijndael”, in *Selected Areas in Cryptography (SAC) 2003*. 2003, vol. 2259 of *LNCS*, pp. 103–111, Springer-Verlag.
15. B. Song and J. Seberry, “Further observations on the structure of the AES algorithm”, in *Fast Software Encryption (FSE) 2003*. 2003, vol. 2887 of *LNCS*, pp. 223–234, Springer-Verlag.
16. J. Wolkerstorfer, E. Oswald, and M. Lamberger, “An ASIC Implementation of the AES Sboxes”, Proceeding of RSA Conference, pp.29-52, Feb. 2002. 872.

17. S. Yoo, D. Kotturi, D. Pan, and J. Blizzard “An AES crypto chip using a high-speed parallel pipelined architecture,” Microprocessors and Microsystems, vol. 29, no. 7, pp. 317-326, Sep. 2005.

18. C.-L. Homg, "An AES cipher chip design using on-the-fly key scheduler", Master Thesis, Dept.Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan, June 2004.

19. Nedjah, N., de Macedo Mourelle, L., Cardoso, M.P., “A Compact Pipelined Hardware Implementation of the AES-128 Cipher”, Third International Conference on InformationTechnology: New Generations, pp. 216-221, April 2006.

20. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A compact Rijndael hardware architecture with S-box optimization", in ASIACRYPT 2001. 2001, vol. 2248 of LNCS, pp. 239-254, Springer-Verlag.

21. Saqib, N.A., Rodriguez-Henriquez, F., Diaz-Perez, A., “AES algorithm implementation - an efficient approach for sequential and pipeline architectures”, Proceedings of the Fourth Mexican International Conference on Computer Science , pp. 126-130, Sept. 2003.