Advanced WG and MOWG Stream Cipher with Secured Initial vector

Dijomol Alias

Pursuing M.Tech in VLSI and Embedded Systems, Viswajyoythi College of Engineering and Technology, Vazhakulam, Kerala

Abstract- Synchronous stream ciphers are lightweight symmetric-key cryptosystems which encrypt a plain-text or decrypt a cipher-text. This project includes two new hardware designs of the Welch–Gong (WG)−128 cipher, one for the multiple output WG (MOWG) version, and the other for the single output version WG based on type−II optimal normal basis representation. The proposed MOWG design uses signal reuse techniques to reduce hardware cost in the MOWG transformation, whereas it increases the speed by eliminating the inverters from the critical path. This is accomplished through reconstructing the key and initial vector loading algorithm and the feedback polynomial of the linear feedback shift register. The proposed WG design uses properties of the trace function to optimize the hardware cost in the WG transformation. The security of WG and MOWG ciphers are increased by providing one way encryption to the initial vector using cryptographic hash functions. The proposed designs have less area and power consumptions than the existing implementations of the WG cipher. The software used for simulation is Xilinx ISE and the programming language used is VHDL. Hardware implementation of the project is done using Spartan-3E FPGA.

Index Terms- Linear feedback shift registers(LFSR), Optimal Normal Basis(ONB), Stream ciphers, Welch-Gong transformation.

I. INTRODUCTION

Traditionally, many hardware-oriented stream ciphers have been built using linear feedback shift registers (LFSRs) and a filter/combiner Boolean function. However, the discovery of algebraic attacks made such a way of design insecure. Many nonlinear feedback shift registers-based stream ciphers have been proposed in the eSTREAM stream cipher project, which have limited theoretical results about their randomness and cryptographic properties, and therefore, their security depends on the difficulty of analyzing the design itself. In addition, the arrival of the 4G mobile technology has triggered another initiative for new stream ciphers.

Synchronous stream ciphers are lightweight symmetric-key cryptosystems. These ciphers encrypt a plain-text, or decrypt a cipher-text, by XORing the plaintext/ cipher-text bit-by-bit with the generated key-stream bits. The key-stream bits are produced using a pseudorandom sequence generator and a seed (secret key). Stream ciphers are heavily used in wireless communication and restricted in resources applications such as 3GPP LTE Advanced security suite, network protocols (Secure Socket Layer, Transport Layer Security, Wired Equivalent Privacy, and Wi-Fi Protected Access), radio frequency identification (RFID) tags, and bluetooth, to name some.

The Welch–Gong (WG)(29, 11) [29 corresponds to GF (2^{29}) and 11 is the length of the LFSR] is a stream cipher submitted to the hardware profile in phase 2 of the eSTREAM project. It has been designed based on the WG transformations to produce key bit-streams with mathematically proved randomness aspects. Such properties include balance, long period, ideal tuple distribution, large linear complexity, ideal two-level autocorrelation, cross correlation with an m-sequence has only three values, high nonlinearity, Boolean function with high algebraic degree, and 1-resilient. The new WG (29, 11) does not suffer the chosen initial value (IV) attack. The number of key-stream bits per run is strictly less than the number of key-stream bits required to perform the attack. In addition, the WG cipher is secure against any attacks because secured initial vector is used. Therefore, the WG(29, 11) is secure and has the randomness properties that cannot be offered by other ciphers and, hence, it has a potential that the WG stream cipher will be adopted in practical applications.

Despite of its attractive randomness and cryptographic properties, few designs have been proposed for the hardware implementations of the WG(29, 11). A direct design using computation in the optimal normal basis (ONB), requires seven multiplications and an inversion over GF(2^{29}). In the proposed system, the inversion operation requires four multiplications and reduced the other seven multiplications of the WG transformation by one through signal reuse.

In this project, two designs are proposed. One for multiple-bit output version of the WG cipher, called multiple output WG (MOWG) and the other one for the WG cipher both with secured initial vector. The secured initial vector is obtained by performing one way encryption to the IV using cryptographic hash functions. The MOWG reduces the hardware cost through signal reuse by removing one multiplier from the WG permutation, whereas it generates $d \leq 17$ output bits. Furthermore, it improves the hardware cost and throughput of the cipher. The key-stream sequences generated by the MOWG cipher possess many of the WG key-stream randomness properties. Also the proposed designs optimize the area by reducing the number of multiplications in the MOWG/WG transforms. This is done through signal reuse for the MOWG and through using the new trace properties for the WG. The proposed WG design has significant area and power consumption reductions and an improved speed compared with existing systems.

II. ADVANCED WG CIPHER

The Welch–Gong (WG)(29, 11) [29 corresponds to GF(2^{29}) and 11 is the length of the LFSR] is a stream cipher submitted to the hardware profile in phase 2 of the eSTREAM project. The proposed WG(29, 11) is a stream cipher in which the properties of the trace function when the elements of GF(2^m) are represented in ONB of type-II is used. The proposed WG design uses these properties to minimize the hardware complexity of its transform and this design eliminates some necessary signals for the generation of the initial feedback, which is required to conduct the key initialization phase of the cipher. Missing of the initial feedback signal is recovered by introducing a serialized scheme to generate it.

The general block diagram for WG stream cipher is shown in figure 3.2. To describe the WG cipher and its operation we use F_2 = GF(2) is a finite field with two elements 0 and 1 and F_2^{29} = GF(2^{29}) is an extension field of GF(2) with 2^{29} elements. Each element in this field is represented as a 29 bit binary vector.

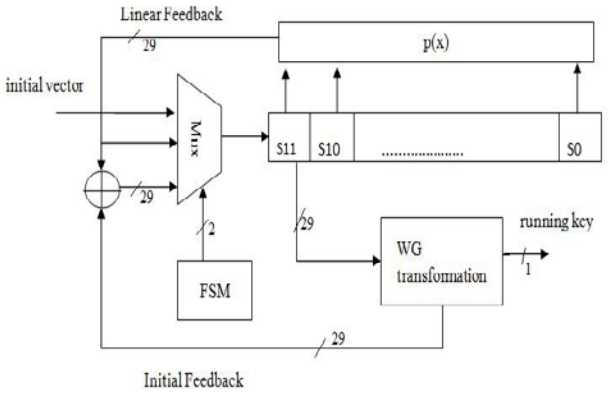

Fig. 1 Block diagram of WG cipher

The WG(29,11) is a bit-oriented filter generator. The WG cipher can be used with keys of length 80, 96, 112 and 128 bits. An initial vector (IV) of size 32 or 64 bits can be used with any of the above key lengths. To increase security, IVs of the same length as the secret key can also be used. In the proposed method, 128 bit key and 128 bit initial vector is used. The sequence generator consists of an orthogonal 29-bit WG transform which is applied to the leftmost cell of a primitive LFSR of degree 11 over GF(2^{29}).

A. WG Generator

WG cipher is a synchronous stream cipher which consists of a WG key-stream generator. A simple block diagram of the WG key-stream generator is shown in figure 3.3. Which consists of a 11-stage LFSR and a transformation block. The key-stream produced by the generator is added bitwise to the plaintext to produce the cipher text. We now describe the WG key-stream generator. As shown in figure 3.3, the key-stream generator consists of a 11 stage linear feedback shift register (LFSR) over F_2^{29} .

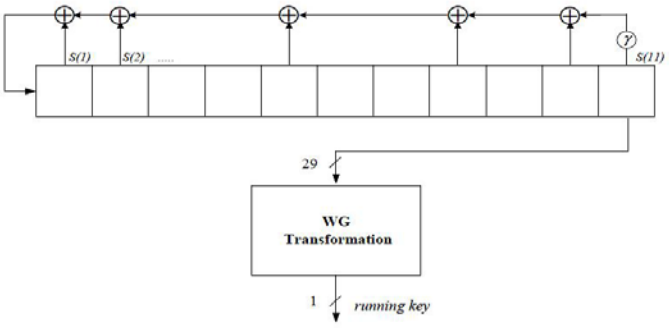

Fig. 2 WG Generator

B. Trace Function

This section presents a method for computing the trace of a multiplication of two field elements when the representation is in the type-II ONB.

Let {β, β^2, β^{2^2}, . . . , $\beta^{2^{m-1}}$} be a type-II ONB in GF (2^m). Then

Tr( ) =1, i=0,1,.....,m-1

Tr($\beta^{2^i}\beta^{2^j}$) =0 , i,j=0,1,.....,m-1.

A type-II ONB is a dual bias. In a In a type-II ONB, the trace of the field multiplication of any two GF (2^m) elements A = (a_0, a_1, . . . , a_{m-1}) and B = (b_0, b_1, . . . , b_{m-1}) is computed as the inner product of A and B as follows:

$$Tr(AB) = \sum_{i=0}^{m-1} a_i b_i \qquad (3.1)$$

Also in type-II ONB, the two relations below are valid for any two elements A and B in GF(2^m).

Tr(AB) = Tr((A>>n)(B>>n)) = $\sum_{i=0}^{m-1} a_{i-n} b_{i-n}$

Tr(AB) = Tr((A<<n)(B<<n)) = $\sum_{i=0}^{m-1} a_{i+n} b_{i+n}$

The trace of the field multiplication of any two elements A and B, represented in type-II ONB, does not change if an *n*-bit cyclic shift (left or right) is applied to both elements in the same direction.

C. WG Transformation

The WG transformation form F_2^{29}□F2 can be regarded as a boolean function in 29 variables. The exact boolean representation depends on the basis used for computation in F_2^{29}. The optimal normal basis provided in previous selections has been selected in such a way so that the corresponding boolean representation of WG transformation is 1-order resilient. The WG transformation is obtained as

WGTrans = Tr(1⊕ X ⊕ X^{r1})+ Tr($X^{2^{2K}}$ (X^{r1} ⊕ X^{2^K-1} ⊕ $X^{2^{sK(2k-1)}}$))

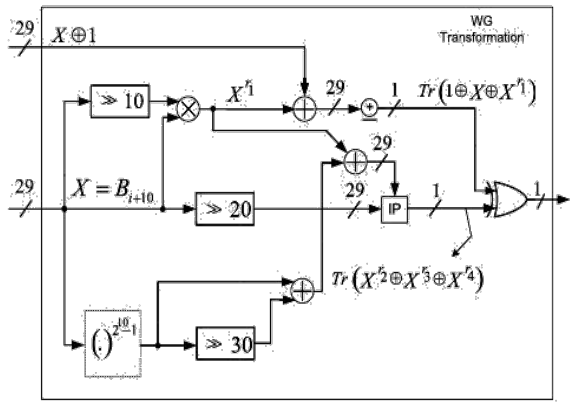

Fig. 3 Advanced WG Transformation

III. MOWG CIPHER

The Multi-Output WG(29, 11, 17) cipher, where 29 corresponds to GF(2^{29}), 11 is the number of stages in the LFSR, and 17 is the number of output bits. In this design, the MOWG transform uses seven multipliers, reducing the number of multipliers comparing to existing MOWG cipher. In addition, in an attempt to improve the overall speed of the cipher, the LFSR is reconstructed to remove the inverters from the critical paths during the PRSG phase/initialization phase. The MOWG transform design has reduced area and the LFSR/key and initial vector loading algorithm (KIA) changes for speed improvement.

A MOWG cipher can be regarded as a nonlinear filter. A MOWG cipher consists of a linear feedback shift register, followed by a MOWG permutation transform. The MOWG permutation transform MOWG(x) is defined in terms the permutation WG_{perm}.

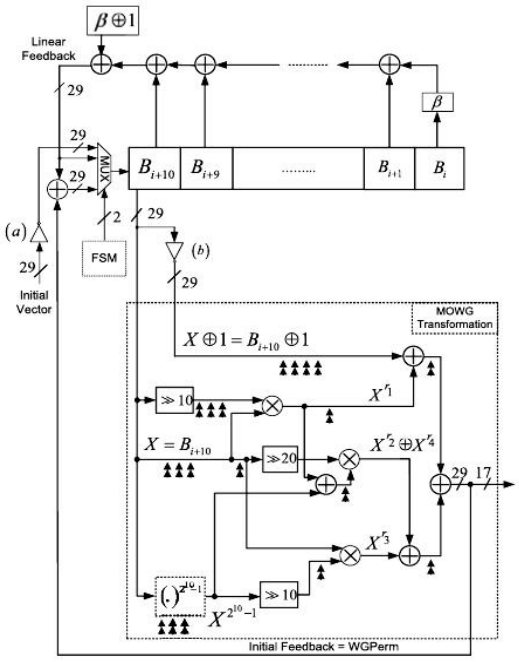

Fig. 4 MOWG Stream cipher

The overall proposed architecture of the MOWG(29, 11, 17) cipher is shown in figure 3.8. In this figure, the FSM controls the input to the LFSR for each phase of operation. In the same figure, because of the bit-wise complement operator, the LFSR receives the complemented IV during the loading phase. Hence, after 11 clock cycles, the initial state of this LFSR, (B_0, B_1, . . . , B_{10}), is basically the complement of the initial state of the LFSR. i.e., Bi = Ai $\oplus$ 1, $0 \le i < 11$. When the key initialization phase starts, the bit-wise XOR of the initial feedback and linear feedback applies to the input of the LFSR.

The input to the LFSR during the key initialization phase in figure 3.8 is complemented with respect to the one.

Throughout the PRSG phase, the only input to the LFSR is the linear feedback signal Bi = Ai $\oplus$ 1, $33 \le i < 2^{319} - 1$. This sets the MOWG transform of figure 3.8 to generate the key-stream bits. The maximum delay of the MOWG transformation is reduced by an amount equivalent to the delay of two inverters.

A MOWG Transformation

The hardware cost of the MOWG cipher is dominated by its transform's field multipliers. Any decrease in the number of these multipliers would minimize the area of the overall cipher. The figure shows the architecture of the proposed MOWG transform, where the number of field multipliers is reduced by 1 through signal reuse technique.

The architecture of the proposed MOWG transform is shown in figure 3.7. Where the WG permutation is

computed as,

$$WGPerm = (1 \oplus X \oplus X^{2^K+1} \oplus X^{2^K(2^K-1)+1} \oplus X^{(2^{2K} \oplus X^{(2^K+1)} \oplus X^{2^K-1}))}$$

(3.7)

In the MOWG(29,11,17), k=10 and, hence ,the signal X^{2^K-1} requires four multiplications and four squaring operations. In figure 3.9, the coordinates of the output of $X \oplus X^{r1} \oplus X^{r2} \oplus X^{r3} \oplus X^{r4}$ in GF(2^{29}) are complemented by the inverter to generate all 29 bits of the WG Perm function, which forms the initial feedback. Seventeen bits of the WGPerm are the output of the MOWG in the run phase. The time delay through the MOWG transform dominates the delay of the overall cipher.

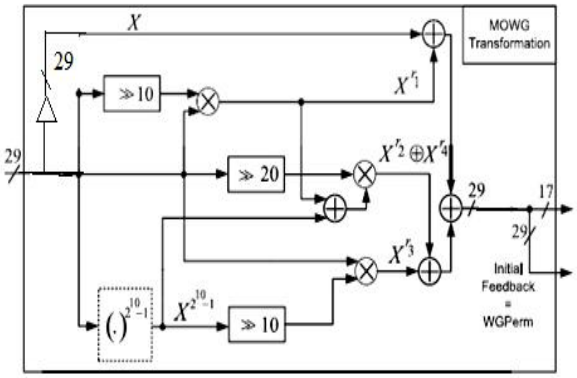

Fig. 5 MOWG transformation block

The modified KIA algorithm is modifying the MOWGs LFSR requires its left most stage to hold the complement of the IV during the loading phase. Therefore, it is required to complement the IV input before it is loaded to the modified LFSR. This can easily be implemented by inserting 29 inverters at the multiplexer's input that receives the IV.

B FSM

This subsection exposes the architecture of the FSM and describes how it schedules the input to the LFSR throughout the three phases of operation. figure 3.10 shows the components of the FSM. The FSM has two inputs, namely clk and reset, 1-bit each, whereas there are two outputs denoted as op0 and op1. The reset input is pulled down before each run of the cipher. This forces the 11-bit one-hot counter to initialize to (1, 0, . . . , 0), i.e., output 0 is the only bit set to a high logic level. In addition, when the reset signal is low, the 2-bit binary counter resets its state to (0, 0). Because of the 1-bit Register connected to the AND gate at the reset input of the 11-bit one-hot counter, this counter starts incrementing one clock cycle after the reset signal gets pulled up. This assures that the 11-bit one-hot counter returns to its initial state after 11 clock cycles. Then, it triggers the 2-bit binary counter to increment that starts the initialization phase.

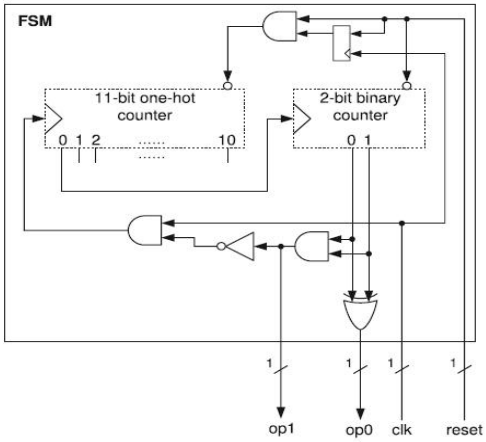

Fig. 6 FSM

IV. SECURED INITIAL VECTOR

The advanced Welch-Gong and Multi-output Welch-Gong stream cipher can be modified by using a secured initial vector to increase the security. The initial vector and a secret key are the two inputs loaded to the LFSR. So by securing the initial vector, the security of the cipher will increase. The secured initial vector is obtained by providing an initial vector hashing using cryptographic hash functions. This is done by developing an algorithm based on cryptographic hash functions.

A cryptographic hash function is a hash function or one way encryption which is considered practically impossible to invert, that is, to recreate the input data from its hash value alone. The input data is often called the message, and the hash value is often called the message digest or simply the digest. Cryptographic hash functions have many information security applications, notably in digital signatures. Hash functions are also commonly employed by many operating systems to encrypt passwords. Here it is used to secure the initial vector.

Initial vector is the input used for one way encryption using cryptographic hash functions. The output obtained is the secured form of initial vector. The algorithm used for hashing technique is given below.

- 128-bit initial vector is the input which is assigned to a temporary variable word.
- h1,h2,h2,h4,h5,h6,h7,h8 are the eight constant hash words each of 16-bit length. assign random values to the hash words.
- The initial vector is divided into 16-blocks and according to the conditions adding, shifting are done and hash values are computed.
- By computing the eight hash values, finally the 128-bit output will obtained in the secured form.

This security of the cipher can be increased by this method. since it is a one way encryption, the initial vector cannot be reconstructed.

V. RESULTS

The simulation results of WG encryption and decryption are shown below.

Fig. 7 WG encrption

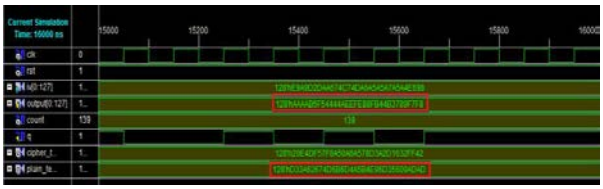

Fig. 8 WG decryption

VI. CONCLUSION

Advanced WG-128 and MOWG-128 stream cipher with secured initial vector are the new stream ciphers were implemented. The proposed MOWG will reduces the number of field multipliers in the transform by signal reuse. In addition, it increases the speed by eliminating two inverters delay from the critical path. This is accomplished by reconstructing the KIA and feedback polynomial of the LFSR. The proposed WG have properties of the trace function for type-II ONB and it will reduce the area and power consumption. The security of the WG and MOWG can be increased by adding the initial vector hashing to the initial vector using cryptographic hash functions. The comparison of WG and MOWG is also implemented in this

project. The software implementation is done in Xilinx ISE and hardware implementation on Spartan-3E FPGA.

REFERENCES

[1] Hayssam El-Razouk, Arash Reyhani-Masoleh, Member, IEEE, and

[2] Guang Gong:"New Implementations of the WG Stream Cipher‖, IEEE transactions on very large scale integration (vlsi) systems, vol. 22, no. 9, september 2014.

[3] Yassir Nawaz and Guang Gong ― The WG stream cipher‖ ECRYPT Stream Cipher Project. University of Waterloo, Canada.

[4] Y. Luo, Q. Chai, G. Gong, and X. Lai, ―A lightweight stream cipher WG-7 for RFID encryption and authentication,‖ in Proc. IEEE Global Telecommun.Conf., Dec. 2010,pp. 1-6.

[5] Y. Nawaz and G. Gong, ―WG: A family of stream ciphers with designed randomness properties,‖ Inf. Sci., vol. 178, no. 7, pp. 1903–1916, 2008.

[6] G. Gong and Y. Nawaz. "New Hardware Implementations of WG(29,11) and WG-16 Stream Ciphers Using Polynomial Basis" [Online]Available: http://www.ecrypt.eu.org/stream/wgp2.htm l

[7] C. Lam, M. Aagaard, and G. Gong, ―Hardware implementations of multi-output Welch-Gong ciphers,‖ Dept. Department of Electr. Comput. Eng., Univ. Waterloo, Waterloo, ON, Canada. 2009.

[8] Guang Gong, Member, IEEE, and Amr M. Youssef:"Cryptographic Properties of the Welch-Gong Transformation Sequence

[9] Generators‖, IEEE transactions on information theory, vol. 48, no. 11, November 2002.

[10] E. Krengel, ―Fast WG Stream Cipher,‖ in Proc. IEEE Region 8 Int. Conf. Comput. Technol. Electron.Eng., Jul.2008,pp.31-35.

AUTHORS

First Author – Dijomol Alias, Pursuing M.Tech in VLSI and Embedded Systems, Viswajyoythi College of Engineering and Technology, Vazhakulam, Kerala email: dijomolalias93@gmail.com