ZUC STREAM CIPHER USING FEEDBACK CARRY SHIFT REGISTER

A.VIJAYA BHASKAR
Department of Electronics and Communication Engineering,
SASI Institute of Technology and Engineering,
Tadepalligudem, Andhra Pradesh, India-534101,
lachavusache@gmail.com

C.RAVI SHANKAR HANUMAN
Department of Electronics and Communication Engineering,
SASI Institute of Technology and Engineering,
Tadepalligudem, Andhra Pradesh, India-534101,
crshanumanster@gmail.com

Abstract:

This paper addresses the usage of FCSR in place of LFSR in a stream cipher. To demonstrate the usage ZUC stream cipher is taken as example in this paper. A good stream cipher should have good randomness, high period, linear span, and security against any known attack. FCSR's provide greater non-linearity than LFSR's.

Keywords: ZUC, FCSR, Stream Cipher.

1. Introduction

The development of a good random number generator has been a hot topic in cryptology. Feedback shift registers have been introduced to fill this requirement. The shift registers must have a very high degree of non-linearity. An LFSR consists of shift register and feedback function. Due to the linearity of LFSR, we can determine the LFSR which generates any output sequence using the Berlekamp–Massey algorithm by knowing 2n output bits only.

Klapper and Goresky [7],[8] proposed a new type of pseudo random binary sequence generator called Feedback with Carry Shift Register (FCSR), FCSR has a shift register, feedback function, and a small amount of memory cells (to store the carry). The bits of the register are added together with the current contents of the memory to form sum. The parity bit of the sum is fed back into the first cell, and the higher order bits are retained for the new value of the memory. It has good non-linearity so the stream ciphers using FCSR cannot be vulnerable to some attacks.

2. ZUC Stream Cipher

ZUC stream cipher is the heart of encryption and decryption algorithms used in cryptography. The structure of ZUC stream cipher is shown in fig.1 and it consists of three layers Linear Feedback Shift Register(LFSR), Bit Reorganization layer(BR), Non-linear function(F).

The LFSR is constructed from 16 register cells, each holding 31 bits, and the feedback is defined by a primitive polynomial over the finite field GF(2^{31}-1). The bit-reorganization extracts 128 bits from the cells of the LFSR and forms four 32-bit words which will be used by the nonlinear function *F* and the output of the key stream. The nonlinear function *F* is based upon two 32-bit memory cells *R*1 and *R*2. The nonlinear function *F* takes 3 of 32-bit words from the bit-reorganization as its inputs and uses two S-boxes *S*0 and *S*1. It also involves different operations such as the exclusive-OR, the cyclic shift and the addition modulo 2^{32}.

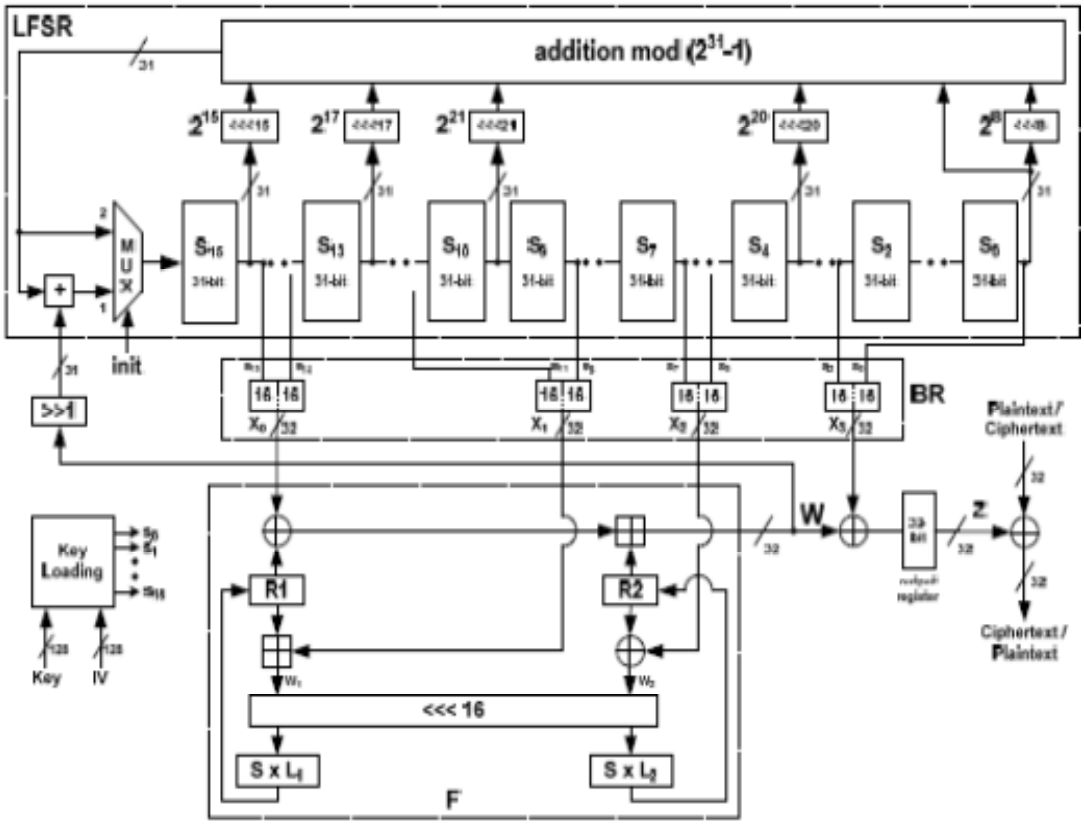

Fig. 1: ZUC Stream Cipher architecture

2.1 Operation

ZUC is a word-oriented stream cipher [2] that takes a 128-bit Key and a 128-bit Initial Vector (IV) as input, and outputs a key stream of 32-bit words. ZUC has three logical layers. The top layer is a Linear Feedback Shift Register (LFSR) of 16 cells, the middle layer is for bit reorganization (BR), and the bottom layer is a nonlinear function F. The LFSR has 16 of 31-bit cells (s0, s1… s15). This LFSR has two stages of operations: the initialization stage and the working stage. In the initialization, the LFSR receives a 31-bit input word u, which is obtained by removing the rightmost bit from the 32-bit output W of the nonlinear function F, (u=W>>1).

The bit-reorganization layer extracts 128-bit from the cells of the LFSR and forms 4 of 32-bit words, where the first three will be used by the nonlinear function F in the bottom layer, and the last word will be involved in producing the key stream.

The nonlinear function F has two 32-bit memory cells R1 and R2. Let the inputs to F be X0, X1 and X2, which come from the outputs of the bit-reorganization. Then function F outputs a 32-bit word W. The 32×32 S-box S is composed of four 8×8 mini S boxes, i.e., S = (S0, S1, S2, S3), where S0=S2, S1=S3. The definitions of S0 and S1 can be found in the official cipher specifications. L1 and L2 are linear transformations from 32-bit words to 32-bit words.

For the cipher operation firstly the key loading procedure expands the initial key and the initial vector into 16 of 31-bit integers as the initial state of the LFSR and then two stages are executed; initialization stage and working stage. In the first stage, a Key/IV initialization is performed and the cipher is clocked without producing output. The second stage is a working stage in which every clock cycle produces a 32- bit word of output.

3. Feedback Carry Shift Register (FCSR)

FCSRs can be designed using Fibonacci or Galois architectures. The main parameters related to the analysis of FCSR are: the connection integer "q", the number of register cells.

3.1 Fibonacci Architecture

In the FCSR architecture (Fig 2), introduced in [6], the basic shift register is provided with a small amount of auxiliary memory m which is a nonnegative integer. The contents (0 or 1) of the tapped cells are added as integers to the current contents of the memory to form an integer sum σ. The parity bit σ mod 2 is fed back into the first cell of the shift register while the higher order bits σ/2 are retained for the new value of the memory.

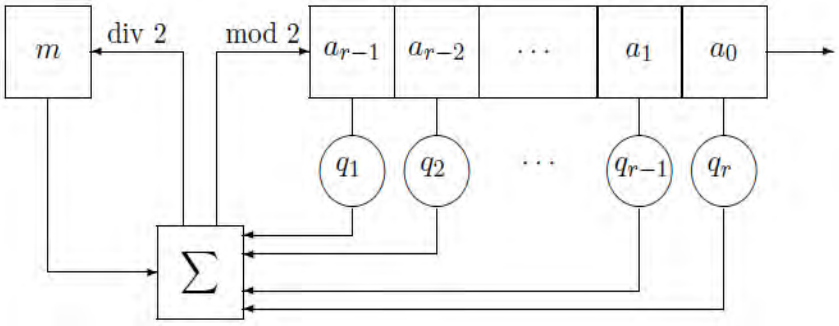

Fig. 2: Fibonacci FCSR

3.2 Galois Architecture

The Galois representation [6] for an FCSR is illustrated in the Fig 3. Here, the bits q_1, q_2. . . q_r are multipliers. The cells denoted c_1, c_2. . . c_{r-1} are the memory (or "carry") bits. The Σ sign represents a full adder. At the j-th adder, the following input bits are received :

- a_j from the preceding cell
- $a_0 q_j$ from the feedback line
- c_j from the memory cell,

which are added to form a sum σ_j (with 1 j r − 1). At the next clock cycle, this sum modulo 2 is passed on to the next cell in the register and the higher order bit is used to replace the memory.

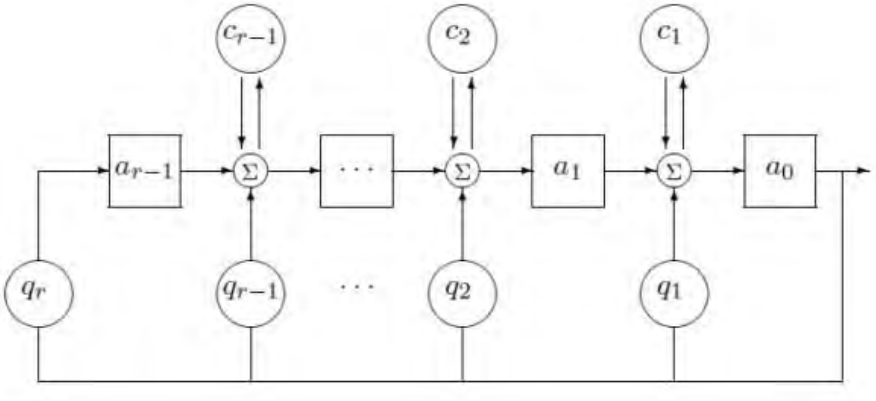

Fig.3: Galois FCSR

4. Proposed ZUC Stream Cipher using FCSR

The block diagram of the proposed ZUC stream cipher using FCSR is shown in Fig.4. Similar to the architecture shown in fig.1 this also consists of three layers Feedback Carry Shift Register (FCSR), Bit Reorganization layer (BR) and Non-Linear function (F). The result of replacing the LFSR with FCSR is improved security because of the inherent non-linearity in the FCSR. So now the proposed ZUC stream cipher is not vulnerable to linear attacks, algebraic attacks…etc.

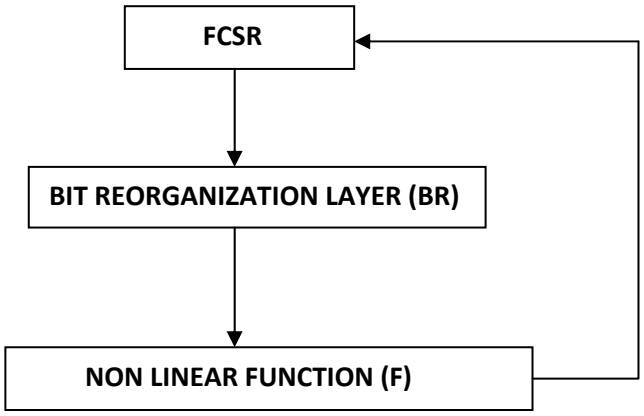

Fig.4: ZUC Stream Cipher using FCSR

5. Results

The ZUC stream cipher operates in two modes where the first mode is initialization mode in which the cipher is clocked for 32 times and the output is ignored. After finishing the initialization mode the cipher operates in working mode where the output z of the stream cipher is used for the encryption and decryption of the data.

The simulation results for the ZUC stream cipher using LFSR and FCSR both are shown in the figures below.

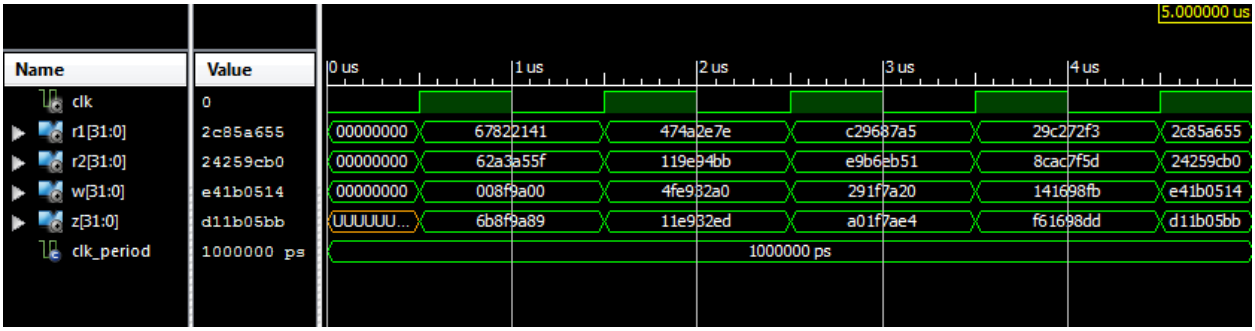

Fig.5: Simulation result of ZUC stream cipher using LFSR in initialization mode

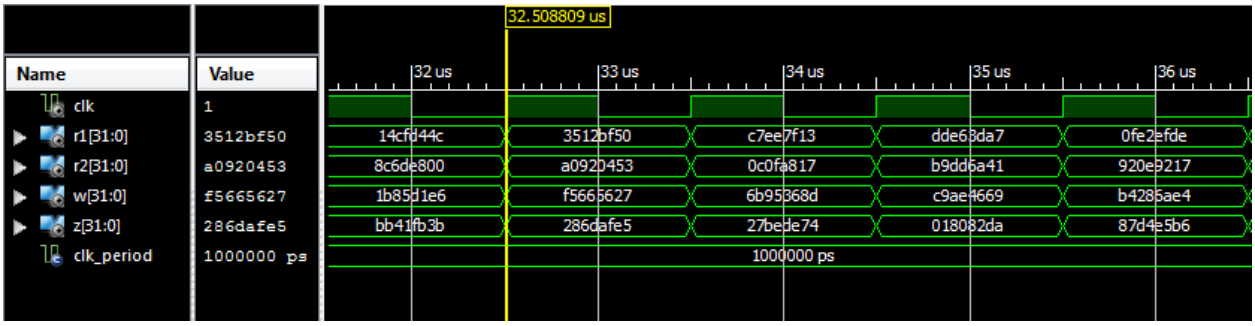

Fig.6: Simulation result of ZUC stream cipher using LFSR in working mode

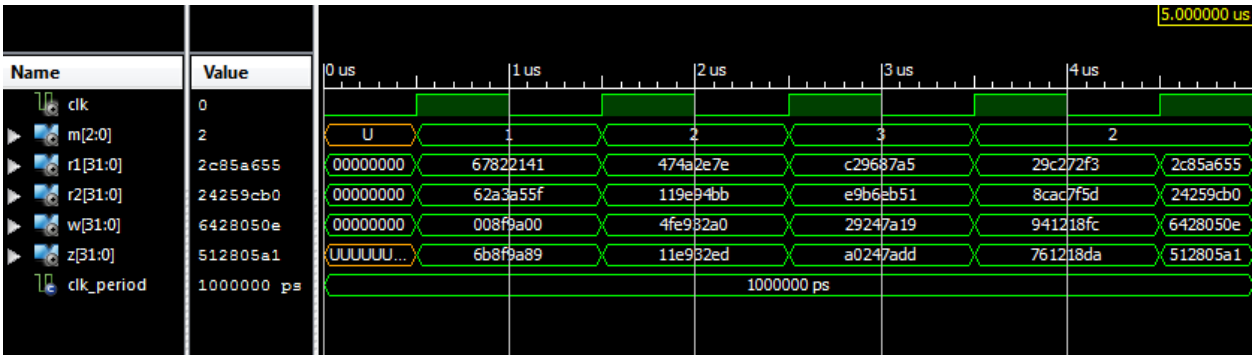

Fig.7: Simulation result of ZUC stream cipher using FCSR in initialization mode

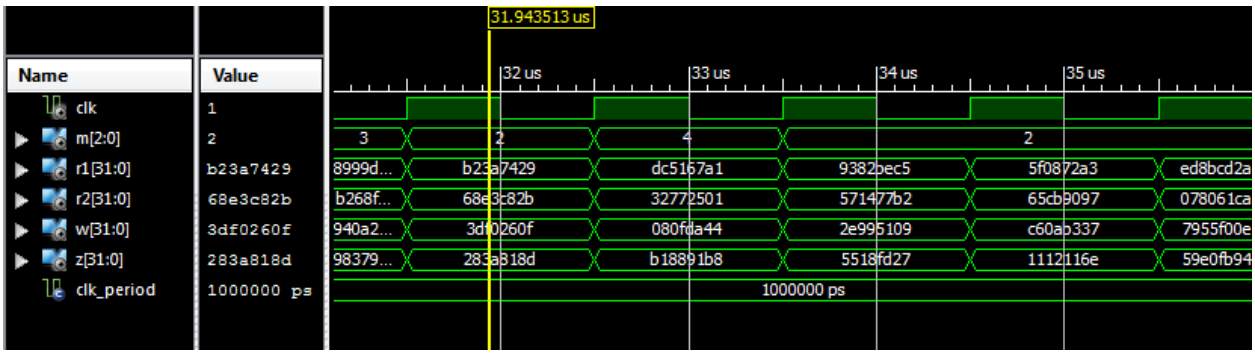

Fig.8: Simulation result of ZUC stream cipher using FCSR in working mode

6. Conclusion

In this paper I have proposed a modified structure of ZUC stream cipher. The security of ZUC stream cipher can be improved by using FCSR in place of LFSR, because LFSR sequences can be predicted using Berlekamp–Massey algorithm. In the future the security of the cipher can be further improved by varying the design of S-boxes which play key role in the non-linear function of the stream cipher.

7. References

[1] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 1: 128- EEA3 and 128-EIA3 Specification; Version: 1.5; Date: 4th January 2011.

[2] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 2: ZUC Specification; Version: 1.5; Date: 4th January 2011.

[3] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 3: Test data; Version: 1.5; Date: 4th January 2011.

[4] An FPGA implementation of ZUC stream cipher: digital system design 2011 14th euro micro conference, Publication Year: 2011 , Page(s): 814 – 817.

[5] Fields and Galois Theory by J.S. Milne, Version 4.22, March 30, 2011

[6] Fibonacci and Galois Representations of Feedback with Carry Shift Registers: Mark Goresky_ Andrew Klapper December 4, 2004

[7] A. Klapper and M. Goresky, Feedback shift registers, 2-adic span, and combiners with memory, J. Crypt. 10 (1997), 111-147.
[8] A. Klapper and M. Goresky, Arithmetic cross-correlation of FCSR sequences, IEEE Trans. Info. Theory 43 (1997) 1342-1346.