On the Linear Complexity of Hard Knapsack Generator sequence

by

Dong-Hyun Choi

A Dissertation Submitted to the

Graduate School of Yonsei University

in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

Supervised by

Professor Hong-Yeop Song, Ph.D.

Department of Electrical and Electronic Engineering
The Graduate School

YONSEI University

January 2004

Contents

List of Figures

List of Tables

ABSTRACT

On the Linear Complexity of Hard Knapsack Generator sequence

Dong-Hyun Choi
Department of Electrical and Electronic Eng.
The Graduate School
Yonsei University

m-sequence is random sequence deterministically generated. Therefore, by combining linear feedback shift register(LFSR) with nonlinear filter or nonlinear combiner, we want to obtain higher linear complexity. One of these method is Rueppel's hard knapsack generator with real adder. Rueppel's hard knapsack generator sequence has maximal linear complexity equal to a period. However, because it has a long stage LFSR for a long period, its hardware complexity increases as long.

In this thesis we propose modified hard knapsack generator that has two additional LFSRs as knapsack weights. Thus modified hard knapsack generator has

much lower than rueppel's in hardware complexity. On the other hand, it has linear complexity profile as Rueppel's has maximal linear complexity equal to the period. In theory, modified hard knapsack generator sequence has a period, the least common multiple of periods which sequences generated by three LFSRs have. Also, the linear complexity of this sequence is derived as upperbound. Finally, we simulate and compare the linear complexity of Rueppel's and modified hard knapsack generator. As a result we know that they have maximal linear complexity equal to period.

Key words : Linear complexity, Period, Hard knapsack generator, Product sequence, m-sequence, Real adder

Chapter 1

Introduction

1.1 Motivation

In this thesis, We consider frequency hopping sequence which has a long period and high linear complexity. Maximal length linear feedback shift register sequence (m-sequence) has been used in spread spectrum communications. But it is very easy to predict given sequence by Berlekamp-Massey algorithm, because m-sequence is deterministically generated random sequences. Therefore the best one can hope for is to make the first period of a periodic sequence resemble the output of a binary symmetric source. A binary symmetric source realizes flipping an unbiased coin successively. Thus we are confronted with the problem of a finite sequence like a binary symmetric source.

Clearly the idea of randomness also reflects the impossibility of predicting the next digit of a sequence from all the previous ones. This property is measured by the length of the shortest linear feedback shift register (LFSR) which can generate the given finite sequence and the length of this LFSR is referred to as the

"linear complexity" associated to the sequence. For example m-sequence which a length L LFSR generates has a maximal period $2^L - 1$ but a linear complexity L.

In order to obtain high linear complexity many techniques have been developed. First method is that one or more LFSRs are combined by nonlinear filter or nonlinear combiner. Another is that clock controlled LFSR is used. In cryptographic systems these methods have been applied to key stream generator.

In this thesis nonlinear combiner and clock controlled LFSR are simultaneously applied to "Hard Knapsack Generator". And the period and linear complexity of hard knapsack generator are derived in theory. In addition I propose the modified hard knapsack generator and compare it with hard knapsack generator in view of the period and the linear complexity.

1.2 Overview

In chapter 2, basic theory of m-sequence and nonlinear function is introduced. Specially the period and the linear complexity of nonlinear function is included. In chapter 3, hard knapsack generator and its property are introduced. Also modified hard knapsack generator is proposed and the property is derived in theory. On the other hand, hard knapsack generator and modified hard knapsack generator are compared in two views of period and linear complexity by the simulation results. Finally this thesis is concluded by the summarizing and giving some remarks in chapter 4.

Chapter 2

Nonlinear theory

2.1 m-sequences

The necessary and sufficient condition for an L-stage linear feedback shift register (LFSR) to produce an m-sequence of period $2^L - 1$ is that the characteristic polynomial of degree L is primitive over GF(2). We will describe some basic properties of m-sequences of period $2^L - 1$. The first three properties, namely, balance, run distribution and ideal autocorrelation property are commonly known as "Golomb's postulates on random sequence." [1] [6] [4]

- **Balance Property** In one period of an m-sequence, the number of 1s and that of 0s are nearly same. Since the period is an odd integer, they cannot be exactly the same, but differ by one. This is called the balance property.

- **Ideal Autocorrelation Property** A periodic unnormalized autocorrelation

function $R(\tau)$ of a binary sequence $s(k)$ of period P is defined as

$$R(\tau) = \sum_{k=0}^{P-1} (-1)^{s(k)+s(k-\tau)}, \qquad \tau = 0, 1, 2, \ldots$$

where $k - \tau$ is computed mod P.

For any integer $L \geq 2$, and for any m-sequence $s(k)$ of period $P = 2^L - 1$, the ideal autocorrelation property of m-sequences refers to the following:

$$R(\tau) = \begin{cases} 2^L - 1, & \tau \equiv 0 \pmod{2^L - 1} \\ -1, & \tau \not\equiv 0 \pmod{2^L - 1} \end{cases}$$

- **Run Distribution Property** A string of the same symbol of length l surrounded by different symbols at both end is called a "run of length l." For example, a run of 1s of length 4 looks like $\ldots 0\underline{1111}0 \ldots$. The run distribution property of m-sequences refers to the fact that a shorter run appears more often than a longer run, and that the number of runs 1s is the same as that of 0s.

- **Span Property** If two vectors $\big((s(i), s(i+1), \ldots, s(i+L-1)\big)$ and $\big((s(j), s(j+1), \ldots, s(j+L-1)\big)$, of length L are distinct whenever $i \neq j$, then the sequence $s(k)$ is said to have this property. The indices of terms are considered mod P. For an m-sequence of period P, in addition, all the not-all-zero vectors of length L appear exactly once on the window of length L.

- **Decimation Property** Let d be a positive integer with $\gcd(2^L - 1, d) = 1$.

Then the sequence $u(t) = s(dt)$, called the $decimation\ by\ d\ of\ s(t)$, is also an m-sequence.

- **Constant on the Coset Property** For any m-sequence of period $2^L - 1$, there are $2^L - 1$ cyclically equivalent sequences corresponding to the $2^L - 1$ starting points. The term $constant\ on\ the\ coset\ property$ refers to the fact that there exists exactly one among all these such that it is fixed with 2 decimation. An m-sequence in this phase is said to be in the $characteristic\ phase$. Therefore the following relation is satisfied:

$$s(2k) = s(k), \quad \text{for all } k$$

- **Cycle and Add Property** When two distinct phases of an m-sequence are added term by term, a sequence of the same period appears and it is a different phase of the same m-sequence. In other words, for any given constants $\tau_1 \not\equiv \tau_2 (\text{mod } 2^L - 1)$, there exists yet another constant τ_3 such that

$$s(k - \tau_1) + s(k - \tau_2) = s(k - \tau_3), \quad k = 0, 1, 2, \ldots$$

This is the cycle-and-add property of m-sequence.

2.2 Nonlinear Theory of Periodic Sequences

A useful measure of unpredictability, or equivalently, randomness of a sequence is provided by the associated linear complexity. Thus there is a need for analyzing nonlinear combinations of (periodic) sequences in terms of linear complexity as well as in terms of period, statistics, leakage etc.

The simplest possible nonlinear transformation is the product of two binary digits. And the product of n variables is said to be an n-th order product. For example, $x_1x_2x_3$ is a third order product. The order of the function f is defined to be the maximum of the order of its product terms.

Then let us consider the product of distinct phase sequences in one m-sequence.

Lemma 2.1 (General upperbound on the linear complexity of nonlinearly filtered PN-sequences)

Let f be any kth-order function of k distinct phases $\tilde{s}^{t_1}$, $\tilde{s}^{t_2}$, ... , $\tilde{s}^{t_k}$ of an m-sequence $\tilde{s}$

$$\tilde{z} = f(\tilde{s}^{t_1}, \ldots, \tilde{s}^{t_k}).$$

Then, the linear complexity of $\tilde{z}$ is upperbound by

$$\Lambda(\tilde{z}) \leq \sum_{i=1}^{k} \binom{L}{i}$$

The proof is referred to [5], [2].

Then let us consider the product of two distinct binary m-sequences.

Lemma 2.2 Let $\tilde{r}$ and $\tilde{s}$ be nonzero sequences over GF(q) with irreducible minimal polynomials $m_{\tilde{r}}(x)$ and $m_{\tilde{s}}(x)$ $\in$GF(q)[x] whose degrees m and n are relatively prime. Let T_1 and T_2 be orders of $m_{\tilde{r}}(x)$ and $m_{\tilde{s}}(x)$.
If sequences are added or multiplied termwise, then the period T of the resulting sequence $\tilde{z}$ is lowerbound by

$$T \geq \frac{T_1 T_2}{(q-1)^2}$$

In particular, when q=2

$$T = T_1 T_2$$

The proof is referred to [5]. By lemma 2.2, for the product sequence which is the multiplication of two m-sequences in GF(2) the condition of relatively prime degrees is sufficient to guarantee the product of periods.
On the other hand, let us consider the linear complexity of product sequence.

Lemma 2.3 (Simple Product)
Let $\tilde{r}$ and $\tilde{s}$ be sequences with irreducible minimal polynomials $m_{\tilde{r}}(x)$ and $m_{\tilde{s}}(x)$ $\in$GF(q)[x] whose degrees m and n are relatively prime. Then the product sequence $\tilde{z} = \tilde{r}\tilde{s}$ has irreducible minimal polynomial $m_{\tilde{z}}(x)$ over GF(q) of degree mn.

The proof is referred to [5].

Example 2.1 To illustrate this principle, we make an example.

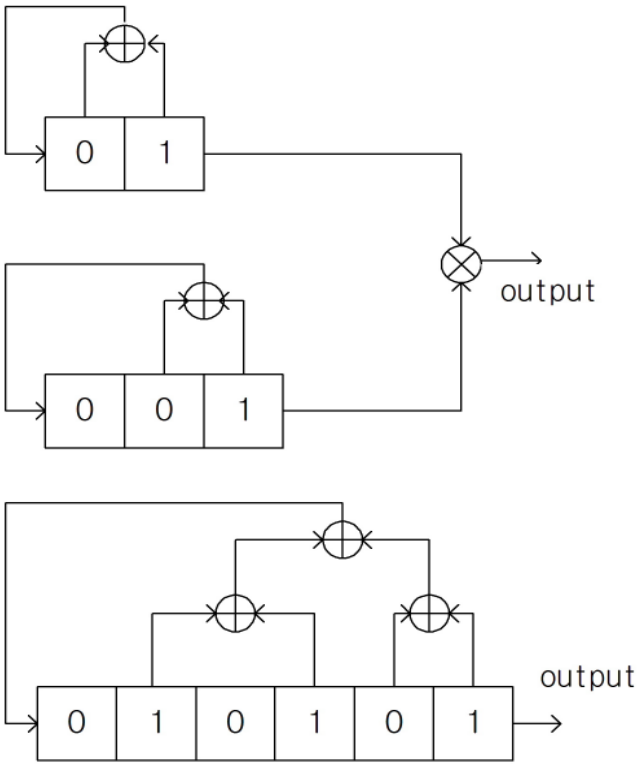

Figure 2.1: Product generator and its linear equivalent

In figure 2.1, two stages LFSR has a primitive root β for which $\beta^2 + \beta + 1 = 0$ and three stages LFSR has a primitive root γ for which $\gamma^3 + \gamma + 1 = 0$. Thus sequence a_n and b_n are

$$\begin{aligned} a_n &= (\beta + 1)\beta^n + \beta\beta^{2n} \\ b_n &= \gamma^n + \gamma^{2n} + (\gamma^2 + \gamma)^n \end{aligned}$$

The product sequence z_n is

$$\begin{aligned} z_n &= a_n b_n \\ &= (\beta+1)(\gamma\beta)^n + \beta(\gamma\beta^2)^n + (\beta+1)(\gamma^2\beta)^n \\ &\quad + \beta(\gamma^2\beta^2)^n + (\beta+1)(\gamma^4\beta)^n + \beta(\gamma^4\beta^2)^n \end{aligned}$$

Therefore the product sequence has the roots $\gamma\beta, \gamma\beta^2, \gamma^2\beta, \gamma^2\beta^2, \gamma^4\beta$ and $\gamma^4\beta^2$, which are the six conjugate roots of the irreducible polynomial $x^6 + x^4 + x^2 + x + 1 = 0$.

From now on, let us consider the knapsack function. It is defined as following.

Definition 2.1 Let N-integer weights be $w_1,\ w_2,\ \ldots\ ,\ w_N$. Then the knapsack function $F_w(x)$ for binary N-tuple vector $\underline{x} = (x_1,\ x_2,\ \ldots\ ,\ x_N)$ is as following.

$$\begin{aligned} F_w(x) &= w_1x_1 + w_2x_2 + \cdots + w_Nx_N \\ &= \sum_{i=1}^{N} w_ix_i \end{aligned} \tag{2.1}$$

It possesses the potential of realizing a nonlinear function with respect to GF(2). First, the knapsack is a very simple and fast transformation, but second, it is also a very flexible transformation(the weights are at our disposal). Thus the problem of implementing nonlinear functions could be solved very easily - by using integer addition.

Proposition 2.1 Let S be the integer sum of N-binary variables $b_1,\ b_2,\ \ldots\ , b_N$. Then

$$S = \sum_{i=1}^{N} b_i$$

and let

$$S = s_0 + s_1 2 + s_2 2^2 + \ldots + s_r 2^r$$

be the binary representation of S.

Then

$$s_i = \prod^{2^i}(b_1, b_2, \ldots, b_N)$$

where

$$\prod^{k}(b_1, b_2, \ldots, b_N) = \sum_{1 \le i_1 < \ldots < i_k \le N} b_{i_1} b_{i_2} \cdots b_{i_k}$$

N binary variables can sum to at most N. Hence their sum is always representable by $\llcorner \log_2 n \lrcorner + 1$ bits by proposition 2.1. To illustrate this principle, we make an example. [2] [4]

Example 2.2 Suppose N=4, then 3bits suffice to represent the integer sum of b_1, b_2, b_3 and b_4.

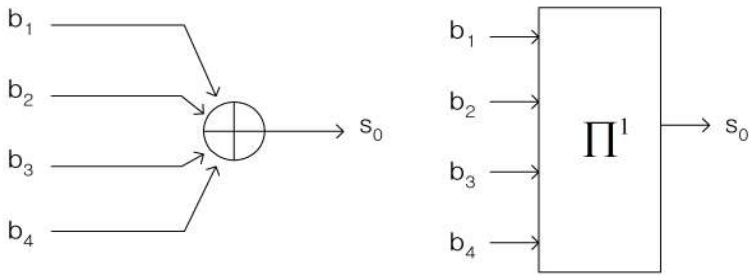

Figure 2.2: Least significant Bit function

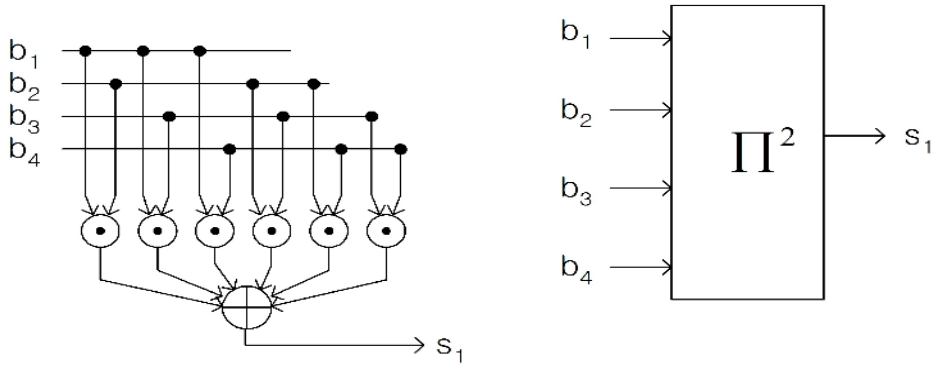

Figure 2.3: Second-least significant bit function

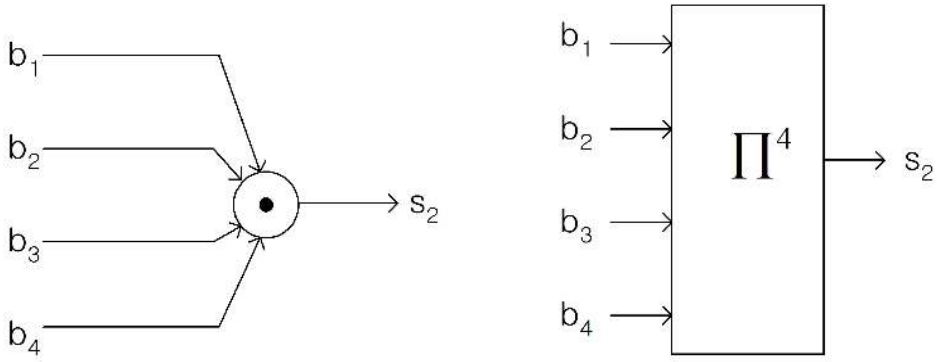

Figure 2.4: Most significant bit function

Then let us consider how many nonlinear orders the knapsack function has. Let the boolean function evaluating the jth bit of the partial sum S be $f_{w,j}(\underline{x})$ for $j = 1, 2, \ldots, N$.

Proposition 2.2 Let the knapsack equation in equation 2.1 be defined over the integers, or over the ring of integer modulo a power of 2. When the knapsack equation is transformed into GF(2)-arithmetic, then the function $f_{w,j}(\underline{x})$ has or-

der

$$\text{ord}[f_{w,j}(\underline{x})] \ \leq \ \min\{2^j, \ N\}$$

Let us consider the implementation of the real adder. When the two input shift registers in figure 2.5 are initially loaded with the binary representation of 2 integers and when the feedback memory cell is initially zero, then after $(n+1)$ clock cycles the $(n+1)$ bits corresponding to the binary representation of the real sum will have appeared serially at the output.

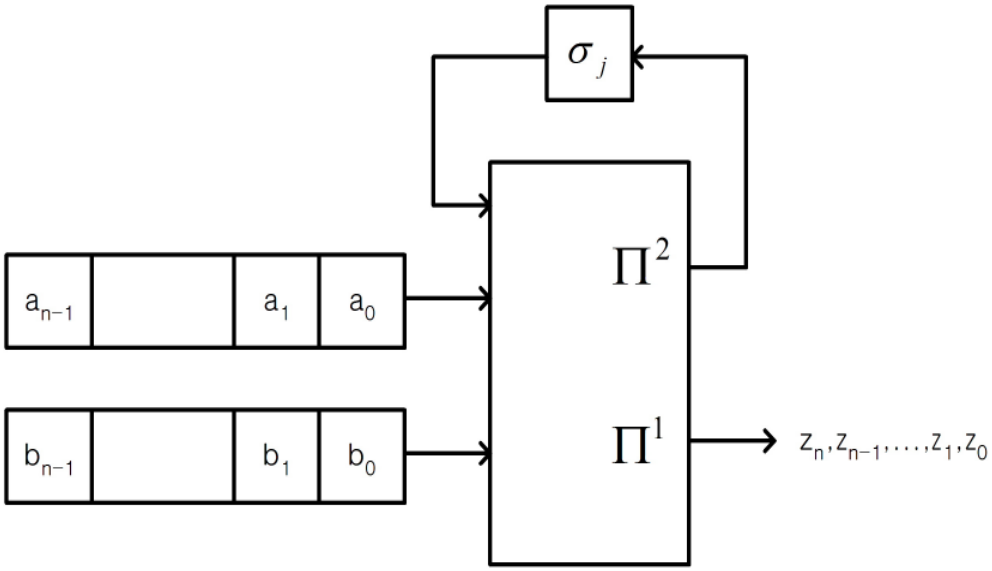

Figure 2.5: Time-sharing of a 3-bit adder to produce bit serially the real sum of two n-bit integers

Let $\tilde{a}$ and $\tilde{b}$ be two binary m-sequences whose primitive polynomials have relatively prime degree L_1 and L_2. Then the real sum sequence $\tilde{z}$ has the following properties. [5]

- **Period** The real sum sequence $\tilde{z}$ has period

$$(2^{L_1} - 1)(2^{L_2} - 1)$$

- **Linear Complexity** The real sum sequence $\tilde{z}$ exhibits linear complexity close to its period length,

$$\Lambda(\tilde{z}) \leq (2^{L_1} - 1)(2^{L_2} - 1) \qquad \text{with near equality.}$$

Chapter 3

Hard Knapsack Generator sequences

3.1 Properties of Hard Knapsack Generator

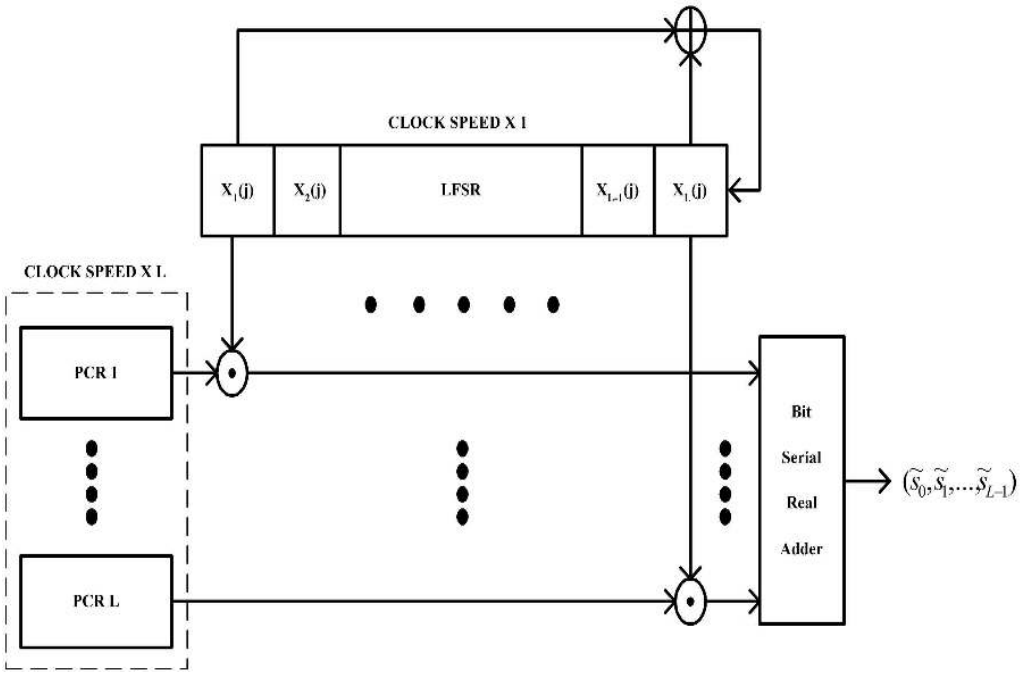

Figure 3.1: Rueppel's Hard Knapsack Generator

Rueppel's hard knapsack generator is described in figure 3.1. The key idea is to replace the nonlinear feedforward function which is applied to the each stages of the LFSR by knapsack whose input vector is taken to be the state of the LFSR and whose output , the integer partial sum, is converted to its binary representation to form key stream. Let the sequence $\tilde{s}_j$ produced at the jth output stage of the knapsack generator. The following is well known properties. [5]

- Let the LFSR which drives the knapsack generator have length L and a primitive connection polynomial. Also let N-stages PCRs be clocked by N-times clock speed and bit serial real adder be cleared after N clocks. Then output sequence $\tilde{s}_j$ has a period $2^L - 1$ and linear complexity as following.

$$\Lambda(\tilde{s}_j) \leq \sum_{i=1}^{2^j} \binom{L}{i} \qquad j < \lceil \log L \rceil$$

$$\Lambda(\tilde{s}_j) \leq \sum_{i=1}^{L} \binom{L}{i} = 2^L - 1 \qquad j \geq \lceil \log L \rceil$$

The period of Rueppel's hard knapsack generator depends on the length of driving LFSR. Thus a long period for Rueppel's hard knapsack generator needs a long stage LFSR and many PCRs. Also it needs a bit serial adder which has high hardware complexity. For example, generally it is unfeasible to attack such a sequence by means of a linear equivalent when the length L of LFSR is about 50. Thus Rueppel's hard knapsack generator has many product terms in table 3.1. In that case although Rueppel's hard knapsack generator has maximum linear complexity, its hardware complexity becomes extremely high.

Table 3.1: Hardware complexity for $L = 50$

order	The number of product terms
1	50
2	$\binom{50}{2} = 1225$
4	$\binom{50}{4} = 230300$
8	$\binom{50}{8} = 536878650$
16	$\binom{50}{16} \approx 4.92 \times 10^{12}$
32	$\binom{50}{32} \approx 1.8 \times 10^{13}$

Therefore we need to change the Rueppel's hard knapsack genenrator for reducing the hardware complexity but preserving the linear complexity.

3.2 Analysis of Modified Hard Knapsack

In this thesis we propose the modified hard knapsack generator to reduce hardware complexity. Instead of two PCRs two LFSRs are replaced in figure 3.2. Let LFSR 1,LFSR 2 and LFSR 3 have pairwise relatively prime $L_1,\ L_2\ and\ L_3$ stages and primitive connection polynomials in figure 3.2. Then the output sequence consists of the real sum of L_1 product sequences.Suppose $L_1 = \llcorner \frac{L}{3} \lrcorner$. Then we compare the hardware complexity between Rueppel's and modified hard knapsack generator.

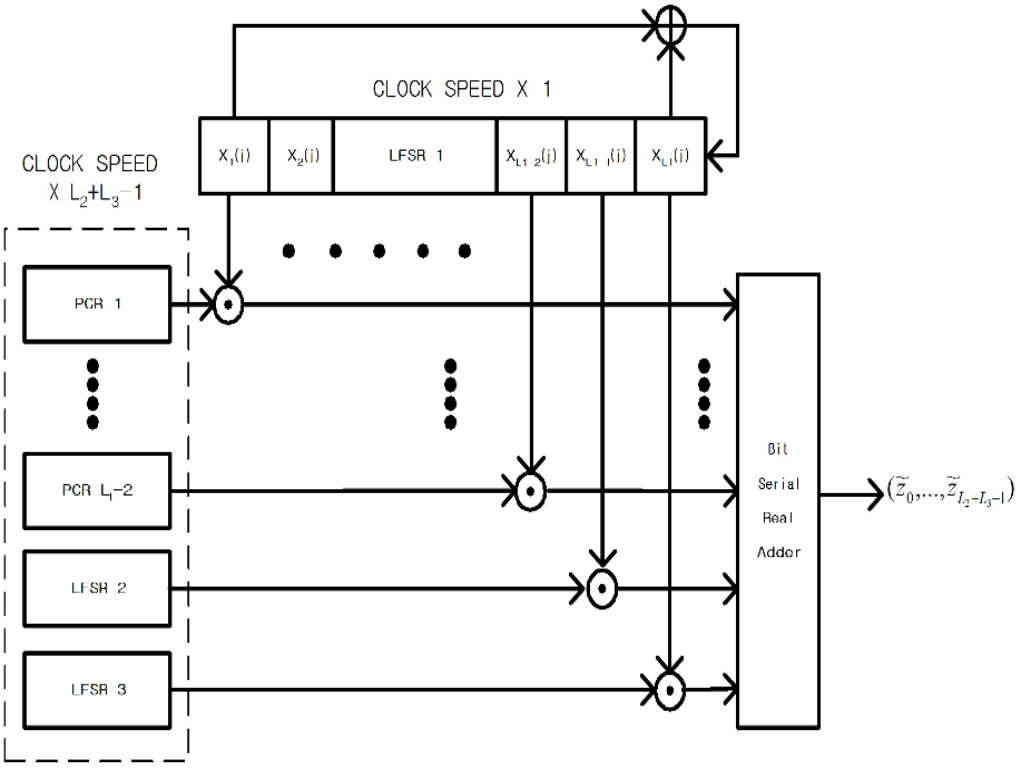

Figure 3.2: Modified Hard Knapsack Generator

Table 3.2: Hardware complexity for $L = 50$ and $L_1 = 16$

order	The number of product terms	
	(Rueppel's)	(Modified)
1	50	16
2	$\binom{50}{2} = 1225$	$\binom{16}{2} = 120$
4	$\binom{50}{4} = 230300$	$\binom{16}{4} = 1820$
8	$\binom{50}{8} = 536878650$	$\binom{16}{8} = 12870$
16	$\binom{50}{16} \approx 4.92 \times 10^{12}$	$\binom{16}{16} = 1$
32	$\binom{50}{32} \approx 1.8 \times 10^{13}$	

We can see the fact that modified hard knapsack generator has much less hardware complexity than Rueppel's in table 3.2. Then we need to guarantee that modified knapsack generator has mostly same performance as Rueppel's in views of period and linear complexity.

First, let us consider the period of modified hard knapsack generator sequence in figure 3.2. In the product of two distinct m-sequences, the resulting sequence has the following properties.

Lemma 3.4 Let $\tilde{\sigma}^1$ and $\tilde{\sigma}^2$ be two distinct binary m-sequences with each period $2^{L_1} - 1$ and $2^{L_2} - 1$. If L_1 and L_2 are relatively prime, then the product sequence $\tilde{\sigma}^3 = \tilde{\sigma}^1\tilde{\sigma}^2$ has period $(2^{L_1} - 1)(2^{L_2} - 1)$ and irreducible minimal polynomial $m_{\tilde{\sigma}^3}(x)$ over GF(2) of degree L_1L_2.

Lemma 3.4 can be derived by lemma 2.3. Thus by lemma 3.4 two product sequences in figure 3.2 have irreducible minimal polynomials and period $(2^{L_1} -$

$1)(2^{L_2}-1)$ and $(2^{L_1}-1)(2^{L_3}-1)$.

Then we consider the sum of two product sequences which are the multiplication of two m-sequences.

Lemma 3.5 For each $i = 1, 2, \ldots, h$, let σ^i be a homogeneous linear recurring sequence with minimal polynomial m_i in GF(2)[x] and least period r_i. If the polynomials $m_1(x), m_2(x), \ldots, m_h(x)$ are pairwise relatively prime, then the least period of the sum sequence $\sigma^1 + \sigma^2 + \ldots + \sigma^h$ is equal to the least common multiple of $r_1, r_2, \ldots, r_h$.

The proof is referred to [3].

Let the output sequence produced at the m-th output stage of modified knapsack generator be $\tilde{z}_m$.

Theorem 3.1 In figure 3.2 the period of output sequence $\tilde{z}_m \quad for\, m = 0, 1, 2, \ldots, L_2 + L_3 - 1$ is $(2^{L_1}-1)(2^{L_2}-1)(2^{L_3}-1)$.

Proof: The product sequences which are multiplications of PCRs and LFSR 1 has period $2^{L_1}-1$. Thus the period of output sequence $\tilde{z}_m\, for\, m = 1, 2, \ldots, L_2 + L_3 - 1$ depends on the two product sequences which are the multiplication of two m-sequences. By lemma 3.4 the product sequences of two m-sequences have irreducible minimal polynomials and period $(2^{L_1}-1)(2^{L_2}-1)$ and $(2^{L_1}-1)(2^{L_3}-1)$. Thus by lemma 3.5 the period of output sequence $\tilde{z}_m$ is $(2^{L_1}-1)(2^{L_2}-1)(2^{L_3}-1)$.

From now on, we will consider the linear complexity. Let the linear complexity of output sequence $\tilde{z}_m$ produced at the m-th output stage of the knapsack generator be $\Lambda(\tilde{z}_m)$. Suppose that modified knapsack generator has no PCR and only 3 LFSRs. Then $\Lambda(\tilde{z}_m)$ is as following.

Theorem 3.2 In figure 3.2 suppose that no PCR exists. Let two distinct phase sequences of LFSR 1 be s_0 ,s_1 and sequences of LFSR 2,3 be a_0,b_0. Then linear complexity of the output sequence $z_m = a_m s_0 + b_m s_1$ at m-th stage output bit is as following.

$$\begin{aligned}
\Lambda(z_m) &\leq (L_2+L_3)L_1 + \left(\binom{L_1}{2} + L_1\right) \sum_{m'=1}^{m} \sum_{k=1}^{m'} \binom{L_2}{k} \binom{L_3}{m'+1-k} \\
&\quad if\ m < L_2 + L_3 - 1 \qquad (3.1) \\
&= \left(\binom{L_1}{2} + L_1\right)(2^{L_2}-1)(2^{L_3}-1) \qquad if\ m \geq L_2 + L_3 - 1 \qquad (3.2)
\end{aligned}$$

Proof: Initially,

$$\begin{aligned}
z_0 &= a_0 s_0 + b_0 s_1 \\
\Lambda(z_0) &= (L_2+L_3)L_1
\end{aligned}$$

In next stage,

$$\begin{aligned}
z_1 &= a_1 s_0 + b_1 s_1 + a_0 b_0 s_0 s_1 \\
\Lambda(z_1) &= (L_2+L_3)L_1 + L_2 L_3 \left(\binom{L_1}{2} + L_1\right)
\end{aligned}$$

From now on, considering the product term of a and b and neglecting the others

$$\begin{aligned} z_0 &= 0 \\ z_1 &= a_0 b_0 s_0 s_1 \end{aligned}$$

In 2nd stage,

$$\begin{aligned} z_2 &= a_1 b_1 s_0 s_1 + a_1 a_0 b_0 s_0 s_1 + b_1 a_0 b_0 s_0 s_1 \\ &= a_1 b_1 s_0 s_1 + a_0 b_0 s_0 s_1 (a_1 + b_1) \\ &= s_0 s_1 \big(a_1 b_1 + a_0 b_0 (a_1 + b_1) \big) \end{aligned}$$

In 3rd stage,

$$\begin{aligned} z_3 &= s_0 s_1 \big(a_2 b_2 + a_1 b_1 (a_2 + b_2) + (a_2 + b_2)(a_1 + b_1) a_0 b_0 + a_1 b_1 a_0 b_0 (a_1 + b_1) \big) \\ &= s_0 s_1 \big(a_2 b_2 + a_1 b_1 (a_2 + b_2) + (a_2 + b_2)(a_1 + b_1) a_0 b_0 \big) \end{aligned}$$

Generally in m-th stage with $m \geq 2$,

$$\begin{aligned} z_m &= s_0 s_1 \big(a_{m-1} b_{m-1} + (a_{m-1} + b_{m-1}) z_{m-1} \big) \\ &= s_0 s_1 \big(a_{m-1} b_{m-1} + (a_{m-1} + b_{m-1}) a_{m-2} b_{m-2} + (a_{m-1} + b_{m-1})(a_{m-2} + b_{m-2}) z_{m-2} \big) \\ &= s_0 s_1 \big(a_{m-1} b_{m-1} + (a_{m-1} + b_{m-1}) a_{m-2} b_{m-2} + (a_{m-1} + b_{m-1})(a_{m-2} + b_{m-2}) a_{m-3} b_{m-3} \\ &\quad + \cdots + (a_{m-1} + b_{m-1})(a_{m-2} + b_{m-2}) \cdots (a_2 + b_2)(a_1 + b_1) a_0 b_0 \big) \end{aligned}$$

Thus the linear complexity of z_m

$$\begin{aligned}
\Lambda(z_m) &\leq \left(\binom{L_1}{2} + L_1\right)\left(L_2L_3 + (L_2\binom{L_3}{2} + L_3\binom{L_2}{2}) + \cdots + \sum_{k=1}^{m'}\binom{L_2}{k}\binom{L_3}{m'+1-k}\right) \\
&= \left(\binom{L_1}{2} + L_1\right)\sum_{m'=1}^{m}\sum_{k=1}^{m'}\binom{L_2}{k}\binom{L_3}{m'+1-k} \\
&= \left(\binom{L_1}{2} + L_1\right)(2^{L_2}-1)(2^{L_3}-1) \qquad if\ m \geq L_2 + L_3 - 1
\end{aligned}$$

with near equality.

3.3 Simulations and Results

Modified hard knapsack generator has 3 LFSRs and two of them operate as knapsack weights. Without LFSR 1 modified hard knapsack is the same as the real summation generator in figure 2.5. But it has LFSR 1 whose bits switch the knapsack weights.

Let the number of stages in LFSR i for i=1,2,3 be L_i. First when $L_1 = 2,\ L_2 = 5,\ L_3 = 7$ and no PCR exists, let observe the linear complexity of output sequence $\tilde{z}$ in table 3.3. Because of no PCR and $L_1 = 2$, the theoretic value of eq-3.1 and eq-3.2 is exactly same as the simulation result from 0th bit to 4th bit and almost same in other bits.

Table 3.3: Linear complexity of each bit for $L_1 = 2$, no PCR

Bit Position	Linear Complexity
0	24
1	129
2	654
3	2019
4	4329
5	7066
6	9439
7	10914
8	11574
9	11772
10	11808
11	11811

But if $L_1 > 2$ and PCRs exist, exact linear complexity can't be calculated. If the number of input in real adder is k, then real adder has $\llcorner \log k \lrcorner$ carry outputs and higher order product terms. In figure 3.3 we compare linear complexity for $L_1 = 2, 3, 4$ and the number of PCRs =$0, 1, 2$. When $L_1 = 2$ and no PCR exists, the period of output sequence is $(2^2 - 1)(2^5 - 1)(2^7 - 1) = 11811$. From 7th bit to 11th bit the output sequences have nearly maximum linear complexity. Also in other case from 7th bit to 11th bit they do.

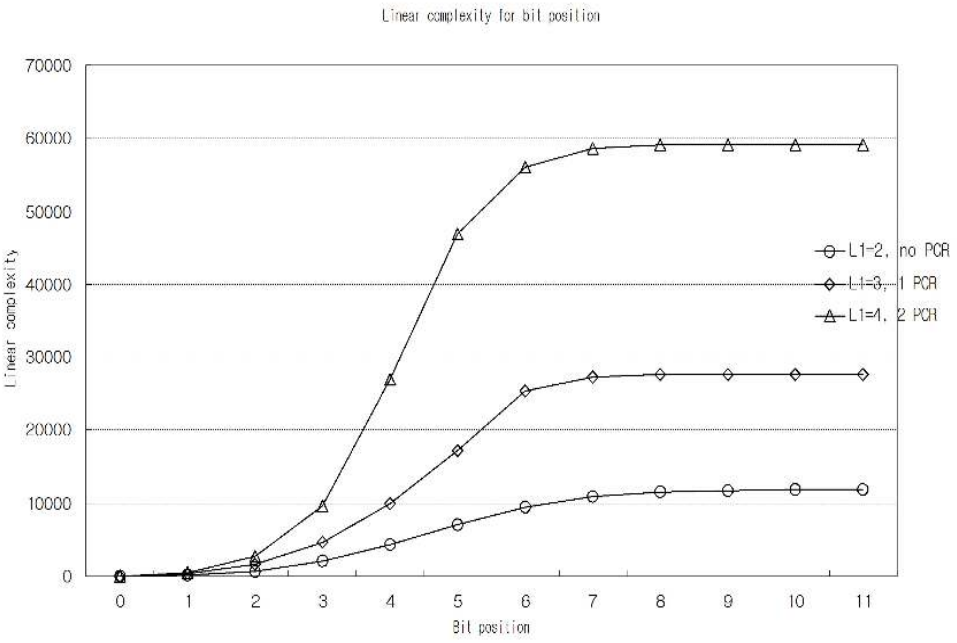

Figure 3.3: Linear complexity of each bit for $L_1 = 2, 3, 4$

Finally, we compare modified knapsack generator with Rueppel's hard knapsack generator. Suppose that the total number of stages in all LFSRs of modified hard

knapsack generator is equal to the number of stages in LFSR of Rueppel's hard knapsack generator. Maximum Linear complexity is equal to whole period of output sequence $\tilde{z}$. But Rueppel's generator has more bits which have maximum linear complexity than modified generator. In figure 3.4 Rueppel's generator has 15 stages LFSR 1. Thus the period of output sequence is $2^{15} - 1$ and maximum linear complexity is also $2^{15} - 1$ from 5th bit to 12th bit. On the other hand modified hard knapsack generator has 3 stages LFSR 1, 5 stages LFSR 2 and 7 stages LFSR 3. The period and maximum linear complexity are $(2^3 - 1)(2^5 - 1)(2^7 - 1)$. But the bits which have maximum linear complexity are from 7th bit to 12th bit. In figure 3.5 Rueppel's generator has 18 stages LFSR and modified

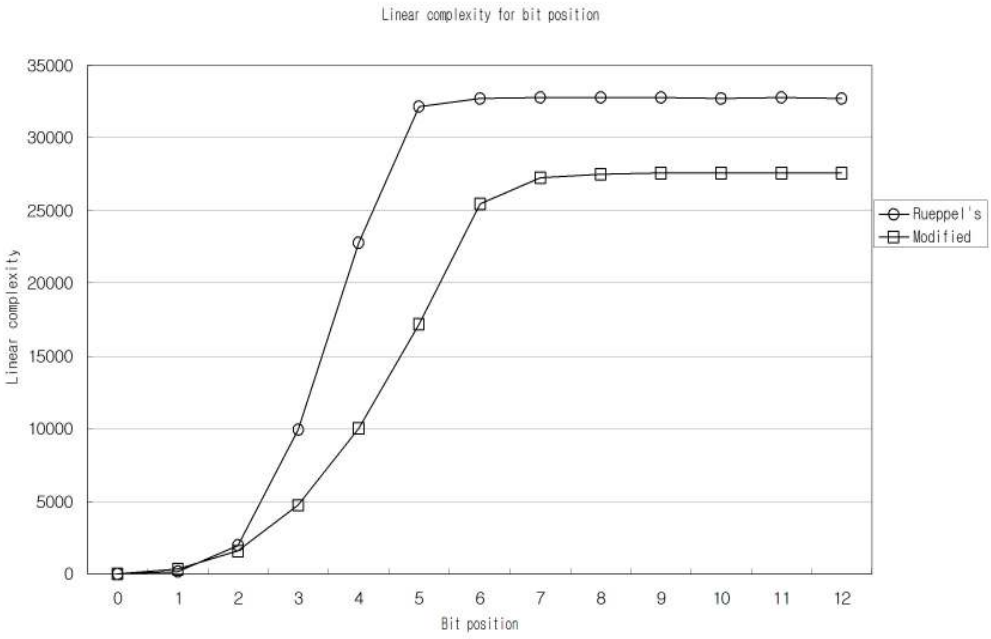

Figure 3.4: Linear complexity of modified knapsack vs Rueppel's generator $L = 15, L_1 = 3, L_2 = 5$ and $L_3 = 7$

generator has 5 stages LFSR 1, 6 stages LFSR 2 and 7 stages LFSR 3. Also both of two generator has maximum linear complexity closed to period. Therefore we can see that modified hard knapsack generator has lower hardware complexity than Rueppel's generator but nearly same linear complexity profile as Rueppel's generator.

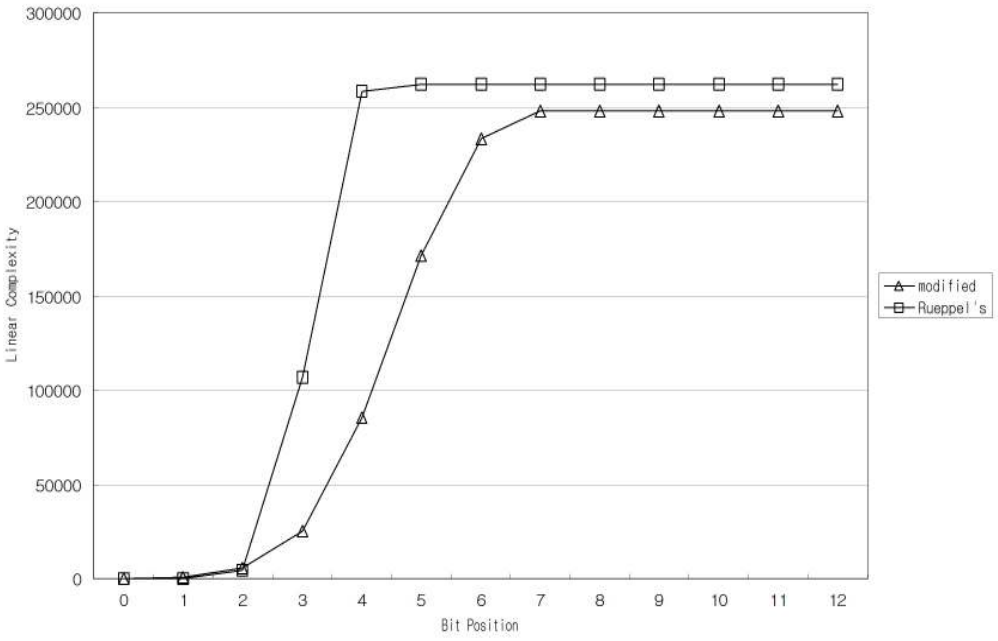

Figure 3.5: Linear complexity of modified knapsack vs Rueppel's generator $L = 18, L_1 = 5, L_2 = 6$ and $L_3 = 7$

Chapter 4

Concluding Remarks

We know that modified hard knapsack generator also has maximum linear complexity closed to the period by theorem 3.2. And the period of output sequence becomes the product of periods which m-sequences have by theorem 3.1. Thus by controlling the number of stages in LFSRs we have the sequence generator with low hardware complexity but nearly same linear complexity and period as Rueppel's. But in theorem 3.2, I can't generalize it for the number of product sequences. Also we need to study the case that the product sequences and PCR sequences are summed.

In this thesis we propose modified hard knapsack generator which has two more LFSRs as knapsack weights than Rueppel's. However, there are many combinations that LFSRs operate as knapsack weights. In example, the distinct phase sequences of one m-sequence or more are used as knapsack weights. Or the generator which has no PCR and only LFSRs are used as knapsack weights. In that case there may be generators which have higher performance and simple structure.

Bibliography

[1] S. W. Golomb. *Shift Register Sequences*. Aegean Park Press, P.O.Box 2837, revised edition, 1982. With portions co-authored by Lloyd R. Welch, Richard M. Goldstein, and Alfred W. Hales.

[2] Edwin L. Key. An analysis of the structure and complexity of nonlinear binary sequence generators. *IEEE Tr. on Information Theory*, IT-22,No.6:732–736, Nov. 1976.

[3] R. Lidl and H. Niederreiter. *Introduction to Finite Fields and Their Applications*. Cambridge University Press, 1988.

[4] Robert J. McElice. *Finite Fields for Computer Scientists and Engineers*. Kluwer Academic Publishers, 2nd edition, 1995.

[5] Rainer A. Rueppel. *Analysis and Design of Stream Ciphers*. Springer-Verlag, 1986.

[6] H.-Y. Song. Feedback shift register sequences. In John G. Proakis, editor, *Wiley Encyclopedia of Telecommunications*, volume 2, pages 789–802. John Wiley and Sons Publication, 2003.

국 문 요 약

변형된 Hard Knapsack 발생기 수열의 선형복잡도

m-sequence는 고정적으로 발생되는 의사 잡음 수열이다. 따라서 선형 귀환 축차 발생기와 비선형 함수의 결합에 의해서 우리는 선형복잡도를 높이고자 한다. 정수 덧셈기를 결합한 Rueppel의 Hard Knapsack 발생기 또한 이와 같은 의도에 의해 제안되었다. 하지만, 주기가 길어지면 하드웨어의 복잡도가 높아져서 구현이 어려워지는 단점이 있다.

본 논문에서는 기존의 Rueppel의 발생기에서 2개의 Pure Cycling 발생기를 선형 귀환 축차 발생기로 대체함으로써, 변형된 Hard Knapsack 발생기를 제안하였다. 변형된 Hard Knapsack 발생기는 하드웨어 복잡도가 Rueppel의 발생기보다 낮으면서, 그 수열의 최대 선형복잡도가 주기와 같은 특성을 유지한다. 변형된 Hard Knapsack 발생기 수열의 주기는 3개의 선형 귀환 축차 발생기에서 생성되는 수열의 주기의 최소 공배수와 같음을 증명하고, 선형복잡도의 상한 값을 유도하였다. 실험을 통해서 각각의 수열의 선형복잡도를 비교한 결과, 변형된 Hard Knapsack 발생기 수열은 선형복잡도의 관점에서 기존의 Rueppel의 발생기 수열과 대등한 성능을 갖음을 확인하였다.

핵심되는 말: 선형복잡도, 주기, Hard Knapsack 발생기, 곱셈 수열, m-sequence, 정수 덧셈기