A Fast Stream Cipher with Huge State Space and Quasigroup Filter for Software

Makoto Matsumoto (Dept. Math. Hiroshima Univ.)
Mutsuo Saito (Dept. Math. Hiroshima Univ.)
Takuji Nishimura (Dept. Math. Yamagata Univ.)
Mariko Hagita (Dept. Info. Ochanomizu Univ.)

August 16 – 17, SAC2007

This study is supported in part by
JSPS Core-to-Core Program 18005

1. Introduction

In this talk, we propose a new kind of stream cipher combining

(1) an LFSR generator (Mother Generator) with huge period and high dimensional equidistribution, and

(2) a filter with memory, based on a quasi group (Quasi group filter).

1. Introduction

In this talk, we propose a new kind of stream cipher combining

(1) an LFSR generator (Mother Generator) with huge period and high dimensional equidistribution, and

(2) a filter with memory, based on a quasi group (Quasi group filter).

We prove that such a combination assures long period and high dimensional equidistribution.

1. Introduction

In this talk, we propose a new kind of stream cipher combining

(1) an LFSR generator (Mother Generator) with huge period and high dimensional equidistribution, and

(2) a filter with memory, based on a quasi group (Quasi group filter).

We prove that such a combination assures long period and high dimensional equidistribution.

At the end of this talk, we propose a concrete instance: CryptMT ver. 3, which is one of the fastest stream ciphers among the candidates of eSTREAM phase 3.

2. Combined Generator

Here we treat a so-called combined generator, namely,

- an Automaton without input as a sequence generator (called Mother Generator, MG),
- an Automaton with input that transforms the sequence into a more secure stream (Filter, F)

2-1. Automaton without Input

A finite state automaton M without input (M for Mother) consists of

- the set of states S_M,
- the state transition function $f_M : S_M \to S_M$,
- the set of the output symbols O_M,
- the output function $o_M : S_M \to O_M$.

2-1. Automaton without Input

A finite state automaton M without input (M for Mother) consists of

- the set of states S_M,
- the state transition function $f_M : S_M \to S_M$,
- the set of the output symbols O_M,
- the output function $o_M : S_M \to O_M$.

For a given initial state s_0, M changes the state by the recursion

$$s_n := f_M(s_{n-1}) \quad (n = 1, 2, 3, \ldots)$$

and generates the sequence

$$o_M(s_0), o_M(s_1), o_M(s_2), \ldots \in O_M.$$

2-2. Automaton with Input

A finite state automaton F with input (F for Filter) consists of

- the set of states S_F,
- the set of input symbols I_F,
- the state transition function $f_F : I_F \times S_F \longrightarrow S_F$,
- the set of the output symbols O_F,
- the output function $o_F : S_F \to O_F$.

2-2. Automaton with Input

A finite state automaton F with input (F for Filter) consists of

- the set of states S_F,
- the set of input symbols I_F,
- the state transition function $f_F : I_F \times S_F \to S_F$,
- the set of the output symbols O_F,
- the output function $o_F : S_F \to O_F$.

For an initial state s_0 and an input sequence $i_0, i_1, \ldots \in I$, F changes the state by

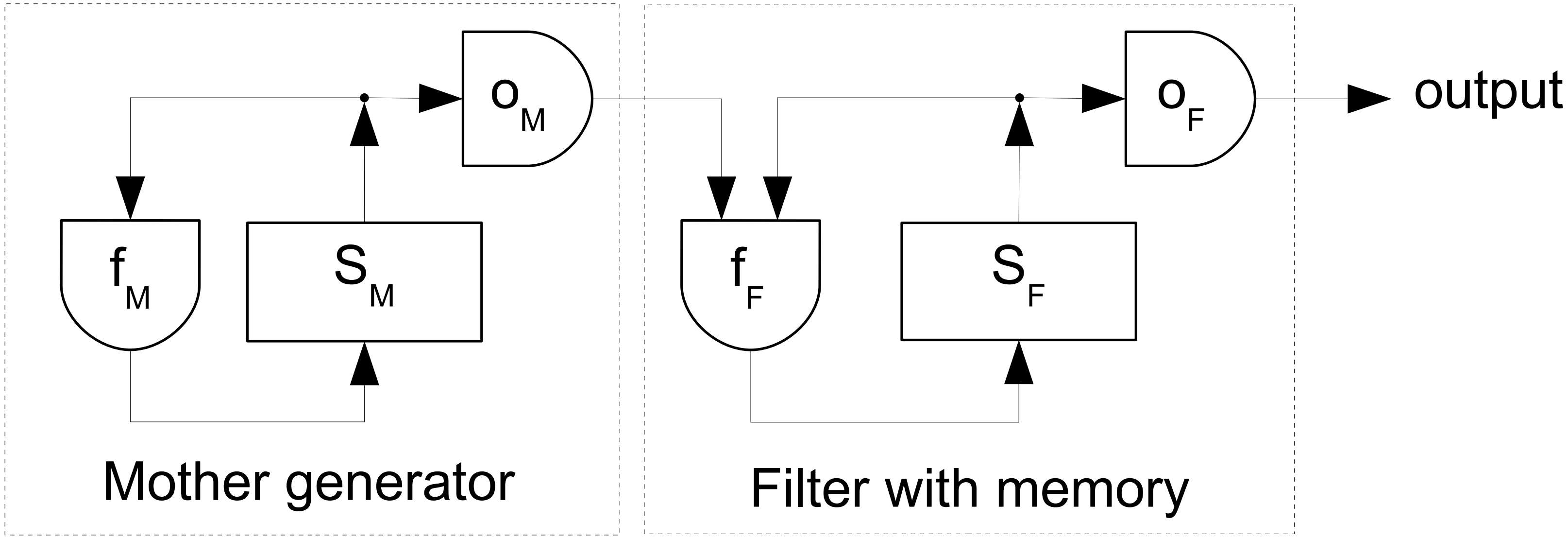
$$s_n = f_F(i_{n-1}, s_{n-1}) \quad (n = 1, 2, 3, \ldots),$$

and generates the sequence

$$o_F(s_0), o_F(s_1), o_F(s_2), \ldots \in O_F.$$

2-3. Combination of two automata

A combined generator is the combination of two automata.



The combined generator can be regarded as an automaton with state space $S_M \times S_F$ without input.

This output is used for stream cipher.

3. Mother Generator

To assure the period and distribution of the combined generator, Mother Generator is assumed to have

- period of large prime number (or having a big prime factor),
- high dimension of equidistribution (later in section 5).

3. Mother Generator

To assure the period and distribution of the combined generator, Mother Generator is assumed to have

- period of large prime number (or having a big prime factor),
- high dimension of equidistribution (later in section 5).

Mersenne Twister (MT, Matsumoto-Nishimura ’97) and
SIMD-oriented Fast Mersenne Twister (SFMT, Saito-Matsumoto ’06)
satisfy these conditions.

4. Equidistribution of automaton without input

Let k be an integer, and let M be an automaton without input.

We consider k-tuple output function $o_M^{(k)}$:

$$o_M^{(k)} : S_M \to O^k \quad s \mapsto (o_M(s), o_M(f(s)), o_M(f^2(s)), \ldots, o_M(f^{k-1}(s)))$$

(i.e. $o_M^{(k)}$ maps the state to the next k outputs).

4. Equidistribution of automaton without input

Let k be an integer, and let M be an automaton without input.
We consider k-tuple output function $o_M^{(k)}$:

$$o_M^{(k)} : S_M \to O^k \quad s \mapsto (o_M(s), o_M(f(s)), o_M(f^2(s)), \ldots, o_M(f^{k-1}(s)))$$

(i.e. $o_M^{(k)}$ maps the state to the next k outputs).
Consider the multi-set of the possible output k-tuples for all states:

$$O_M^{(k)} := \{o_M^{(k)}(s) \mid s \in S_M\}.$$

This is the image of S_M by $o_M^{(k)}$ counted with multiplicities.

4. Equidistribution of automaton without input

Let k be an integer, and let M be an automaton without input.

We consider k-tuple output function $o_M^{(k)}$:

$$o_M^{(k)} : S_M \to O^k \quad s \mapsto (o_M(s), o_M(f(s)), o_M(f^2(s)), \ldots, o_M(f^{k-1}(s)))$$

(i.e. $o_M^{(k)}$ maps the state to the next k outputs).

Consider the multi-set of the possible output k-tuples for all states:

$$O_M^{(k)} := \{o_M^{(k)}(s) \mid s \in S_M\}.$$

This is the image of S_M by $o_M^{(k)}$ counted with multiplicities.

The automaton M is said to be ***k*-dimensionally equidistributed** if the multiplicity of each element in $O_M^{(k)}$ is same.

4. Equidistribution of automaton without input

Let k be an integer, and let M be an automaton without input.
We consider k-tuple output function $o_M^{(k)}$:

$$o_M^{(k)} : S_M \to O^k \quad s \mapsto (o_M(s), o_M(f(s)), o_M(f^2(s)), \ldots, o_M(f^{k-1}(s)))$$

(i.e. $o_M^{(k)}$ maps the state to the next k outputs).
Consider the multi-set of the possible output k-tuples for all states:

$$O_M^{(k)} := \{o_M^{(k)}(s) \,|s \in S_M\}.$$

This is the image of S_M by $o_M^{(k)}$ counted with multiplicities.
The automaton M is said to be ***k*-dimensionally equidistributed** if the multiplicity of each element in $O_M^{(k)}$ is same.

Remark: If M is an $\mathbb{F}_2$-linear generator (this is our case), this coincides with surjectivity of $o_M^{(k)}$, hence computable.

5. Quasigroup Filter

A **quasigroup filter** is an automaton with input where the state transition function

$$f_F : I_F \times S_F \to S_F$$

is bi-bijective.

5. Quasigroup Filter

A **quasigroup filter** is an automaton with input where the state transition function

$$f_F : I_F \times S_F \to S_F$$

is bi-bijective.

A function

$$f : X \times Y \to Z$$

is said to be **bi-bijective** if

$$f(-, y) : X \to Z,\ x \mapsto f(x, y)$$

is bijective for any fixed y, and so is

$$f(x, -) : Y \to Z,\ y \mapsto f(x, y)$$

for any fixed x. If $X = Y = Z$, this coincides with the notion of a quasigroup.

6. Theorems

6-1. Theorem on distribution

A mapping

$$g : X \to Y$$

is **uniform** if the cardinality of $g^{-1}(y)$ is independent of $y \in Y$.

A bijection and a surjective group homomorphism are uniform.

A composition of uniform mappings is uniform.

6. Theorems

6-1. Theorem on distribution

A mapping

$$g : X \to Y$$

is **uniform** if the cardinality of $g^{-1}(y)$ is independent of $y \in Y$.

A bijection and a surjective group homomorphism are uniform.

A composition of uniform mappings is uniform.

Theorem 1. If a combined generator has

- k-dimensionally equidistributed Mother Generator,
- quasigroup filter F,
- uniform output function o_F of F,

then the combined generator is $(k+1)$-dimensionally equidistributed. (Proof is easy)

6-2. Theorem on period

Assume the following:

- Mother Generator is k-dimensionally equidistributed,
- period of Mother Generator is Qq, where Q is a (big) prime, (q is a not large integer,)
- F is a quasigroup filter,
- and o_F is uniform.

6-2. Theorem on period

Assume the following:

- Mother Generator is k-dimensionally equidistributed,
- period of Mother Generator is Qq, where Q is a (big) prime, (q is a not large integer,)
- F is a quasigroup filter,
- and o_F is uniform.

If

$$(\#(O_F))^{k+1} > q \cdot (\#(S_F))^2,$$

then the period of the output sequence of the combined generator is a nonzero multiple of Q.

Period of MG $= Qq$, dimension of equidistribution of $M = k$

$$\#(O_F)^{k+1} > q \cdot \#(S_F)^2 \Rightarrow \text{period} \geq Q.$$

Roughly saying: If

- k : large: (MG has sufficiently high-dimensional equidistribution,)
- Qq : (the period of MG has a dominating big prime factor,)
- $\#(S_F)$: (quasigroup filter has relatively small state space S_F,)

then the period of combined generator is a multiple of the prime Q.

Period of MG $= Qq$, dimension of equidistribution of $M = k$

$$\#(O_F)^{k+1} > q \cdot \#(S_F)^2 \Rightarrow \text{period} \geq Q.$$

Roughly saying: If

- k : large: (MG has sufficiently high-dimensional equidistribution,)
- Qq : (the period of MG has a dominating big prime factor,)
- $\#(S_F)$: (quasigroup filter has relatively small state space S_F,)

then the period of combined generator is a multiple of the prime Q.

Theorem requires huge MG and smaller F.
CryptMT ver. 3 satisfies these conditions:

- $Q = 2^{19937} - 1, \quad 1 \leq q \leq 2^{31} - 1,$
- $k = 155, \quad \#(S_F) = 2^{128}, \quad$ and $\#(O_F) \geq 2^2$.

7. Multiplicative Filter

Here we show two examples of quasigroup filter.

Example 1. Multiplicative filter

Let $I_F = S_F = (\mathbb{Z}/2^{32})^\times$ be the set of odd integers in $Z/2^{32}$.

Let

$$f_F : I_F \times S_F \to S_F$$

be the integer multiplication modulo 2^{32}.

This is a multiplicative group of the ring, hence a quasigroup.

7. Multiplicative Filter

Here we show two examples of quasigroup filter.

Example 1. Multiplicative filter

Let $I_F = S_F = (\mathbb{Z}/2^{32})^\times$ be the set of odd integers in $Z/2^{32}$.

Let

$$f_F : I_F \times S_F \to S_F$$

be the integer multiplication modulo 2^{32}.

This is a multiplicative group of the ring, hence a quasigroup.

We choose

$$o_F : S_F \to O_F = \{0,1\}^8$$

as the function taking the 8 MSBs from the 32-bit integer.

We used this filter in CryptMT ver. 1.

Example 2. Modified multiplicative filter

Through a bijection

$$\begin{aligned} \mathbb{Z}/2^{32} &\longrightarrow (\mathbb{Z}/2^{33})^{\times} \\ x &\longmapsto 2x+1, \end{aligned}$$

$\mathbb{Z}/2^{32}$ has a group structure $\tilde{\times}$ induced from that of $(\mathbb{Z}/2^{33})^{\times}$:

$$\tilde{\times} : (x, y) \mapsto x \tilde{\times} y := 2xy + x + y \bmod 2^{32}.$$

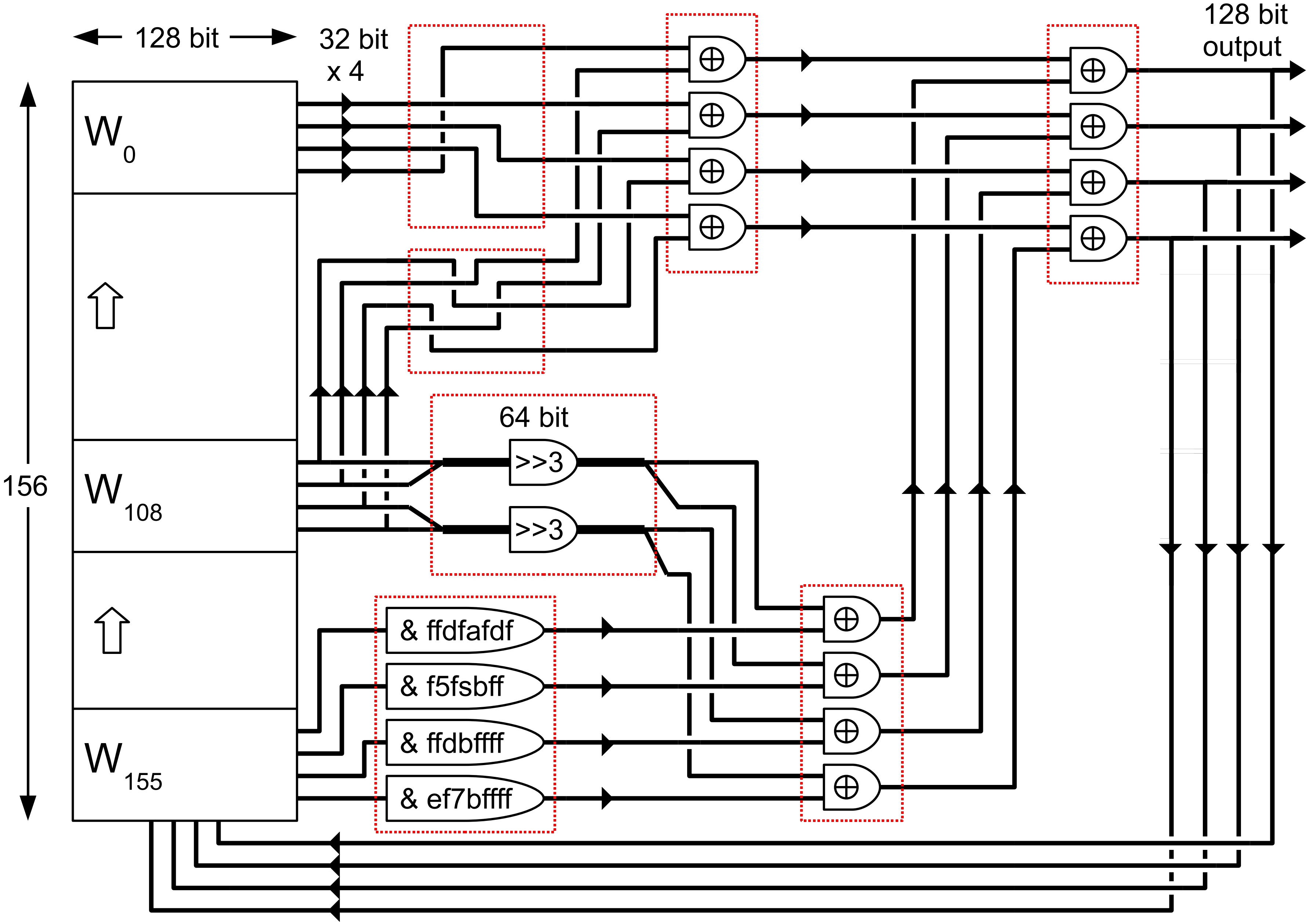
We can consider the corresponding multiplicative filter with $I_F = S_F = \mathbb{Z}/2^{32} =$ 32-bit integers.

8. CryptMT version 3

CryptMT version 3 is such a combined generator.

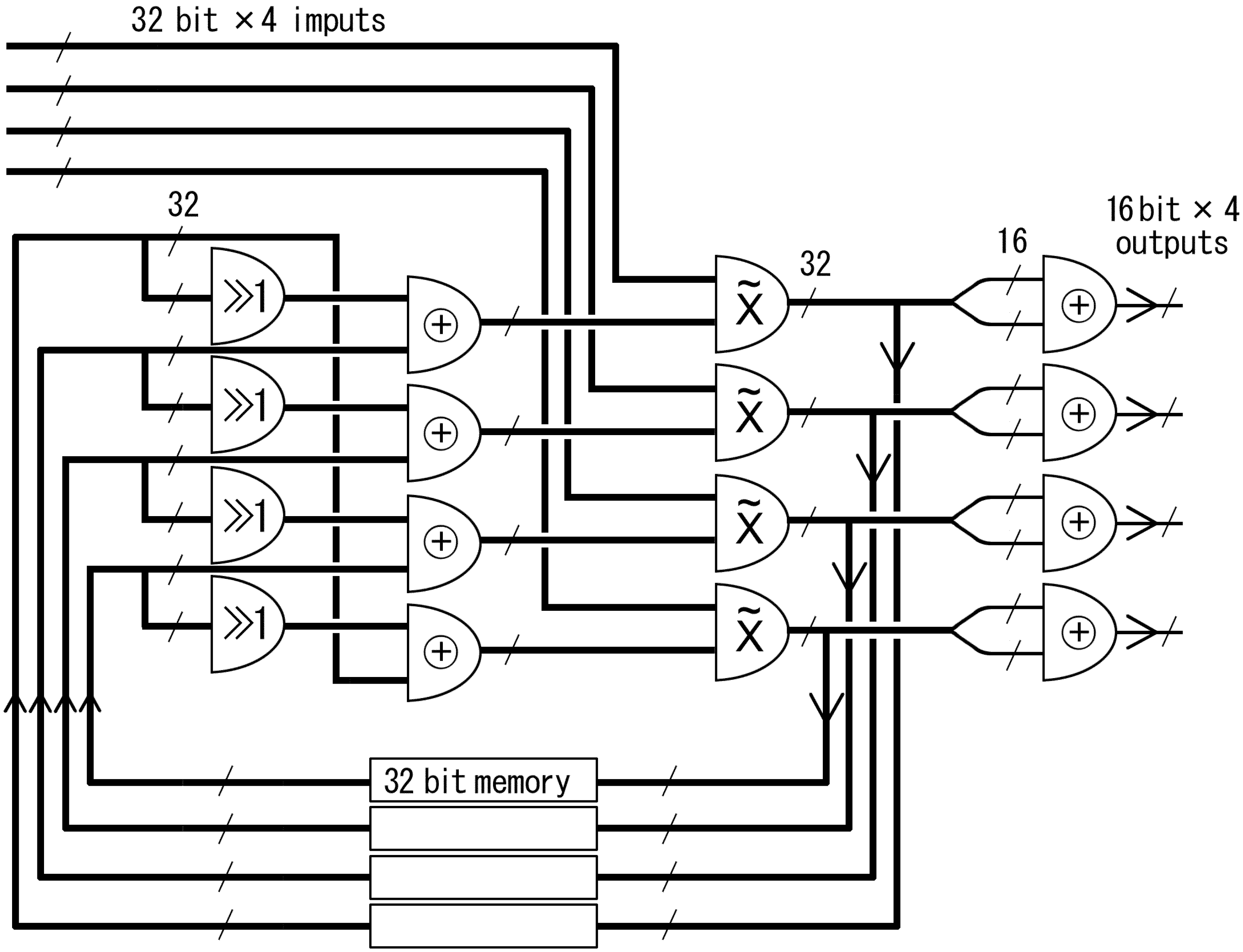
8-1. Mother Generator: SIMD-oriented Fast Mersenne Twister (SFMT)

- a 128-bit sequence LFSR generator,
- using 128-bit operations: Single Instruction Multiple Data (SIMD) instruction set,
- period is a multiple of $2^{19937} - 1$,
- dimension of equidistribution is at least 155.


128 bit
32 bit
x 4
128 bit
output
156
W_0
⇧
W_{108}
⇧
W_{155}
64 bit
>>3
>>3
& ffdfafdf
& f5fsbff
& ffdbffff
& ef7bffff
⊕

8-2. Filter: A Modified Multiplicative Filter using SIMD

- S_F is a 128-bit memory,
- use group operation $\tilde{\times}$ on each of four 32-bit integers.
- additional 128-bit bit mixing.
- its output function extracts 64 bit from 128 bit.


32 bit ×4 imputs
32
≫1
≫1
≫1
≫1
⊕
⊕
⊕
⊕
X̃
X̃
X̃
X̃
32
16
16bit × 4
outputs
⊕
⊕
⊕
⊕
32 bit memory

8-3. Properties

CryptMT ver. 3 has theoretically assured properties:

- period of a multiple of $2^{19937} - 1$,
- at least 1241 dimensional equidistribution as 8 bit sequence,

and plausibly

- high algebraic degree (we omit here.)

8-4. Comparison of the speed

Compared generators:

- CryptMT3,
- SOSEMANUK,
- Dragon,
- SNOW 2.0,
- HC-256,
- Salsa20,
- AES (counter-mode)

Five candidates in eSTREAM software cipher phase 3 [2007, ECRYPT] permitting 256-bit Key, and two reference ciphers SNOW 2.0 and AES.

CPUs:

- Intel Core 2 Duo 2137MHz,
- AMD Athlon X2 2000MHz,
- Motorola PowerPC G4 533MHz.

The number of cycles consumed per byte encryption.

	Core 2 Duo	Athlon 64 X2	PowerPC G4
CryptMT3	2.95	4.73	9.23
HC-256	3.42	4.26	6.17
SOSEMANUK	3.67	4.41	6.17
SNOW-2.0	4.03	4.86	7.06
Salsa20	7.12	7.64	4.24
Dragon	7.61	8.11	8.39
AES-CTR	19.08	20.42	34.81

This table is copied from Bernstein's Homepage:
http://cr.yp.to/streamciphers/timings.html.

CryptMT3 is the fastest in Intel Core 2 Duo CPU, reflecting the efficiency of SIMD operations there.
(PowerPC lacks 32-bit multiplication instruction in SIMD.)

9. The booter

Problem: Mother Generator with huge state space needs long time for initialization.

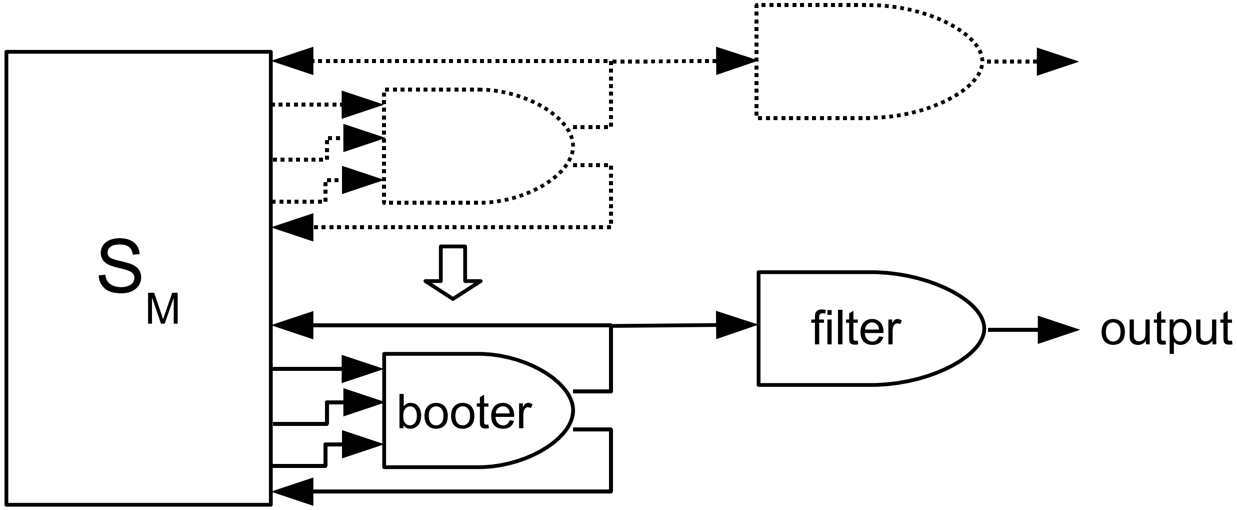
Answer: During MG is being initialized, the sequence used for initialization is outputted through the filter.

9. The booter

Problem: Mother Generator with huge state space needs long time for initialization.

Answer: During MG is being initialized, the sequence used for initialization is outputted through the filter.

The **booter** is a smaller sequence generator to fill up the state array of MG, and its output is passed to the filter at the same time. When the state array of MG filled, then booter switches to MG.



10. Conclusions

We proposed a combination of an LFSR and a uniform quasigroup filter as a stream cipher in software.

- Period and dimension of equidistribution are theoretically assured,
- Booter for initialization and generate initial sequence,
- A fast example CryptMT ver. 3.

10. Conclusions

We proposed a combination of an LFSR and a uniform quasigroup filter as a stream cipher in software.

- Period and dimension of equidistribution are theoretically assured,
- Booter for initialization and generate initial sequence,
- A fast example CryptMT ver. 3.

We did not mention in this talk, but the following properties are shown in the paper in preproceedings:

- high algebraic degree of multiplicative filter
  (both by an analysis on multiplication and by a toy model).
- ordinary attacks seem to be difficult because of the huge state space and high non-linearity.

Thank you for your kind attention.