Differential and Linear Properties of FCSRs

Gaël Thomas

XLIM – CNRS UMR no. 7252 – Limoges

October 8, 2012

This work was partially supported by the French National Agency of Research: ANR-11-INS-011.

Introduction

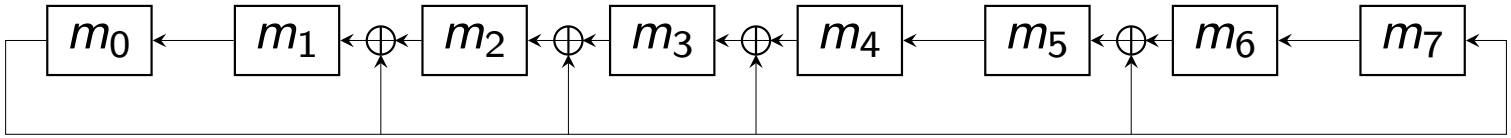
- LFSRs widely used in cryptography
- But linearity $\Rightarrow$ weaknesses
- FCSRs as an alternative

What we propose here :

- identify some differential biases
- explain what we want to do about linear biases

Summary

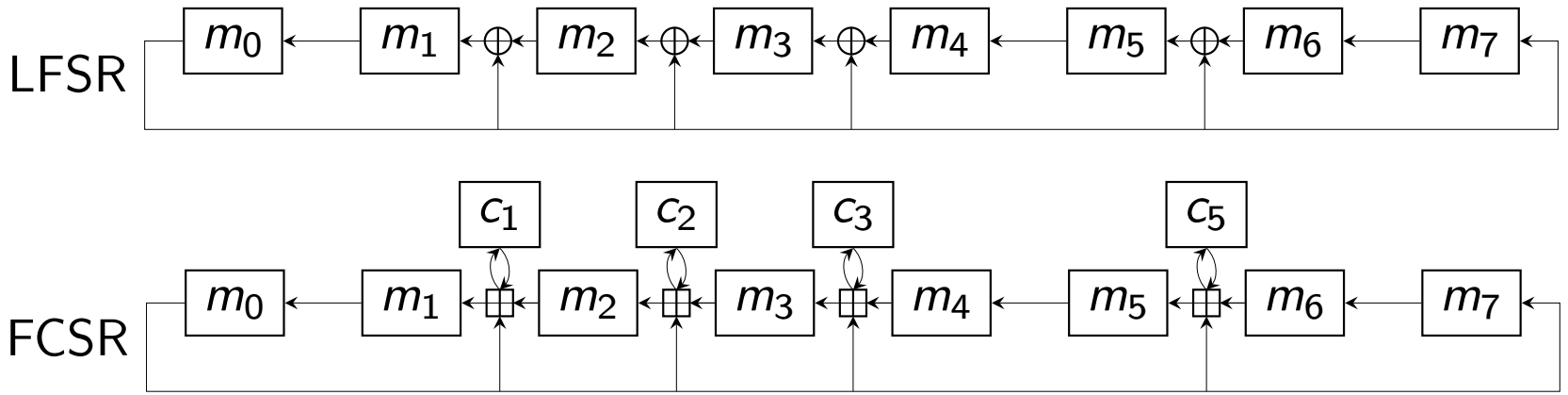
From LFSRs to FCSRs



LFSRs:

- modulo 2 addition

From LFSRs to FCSRs



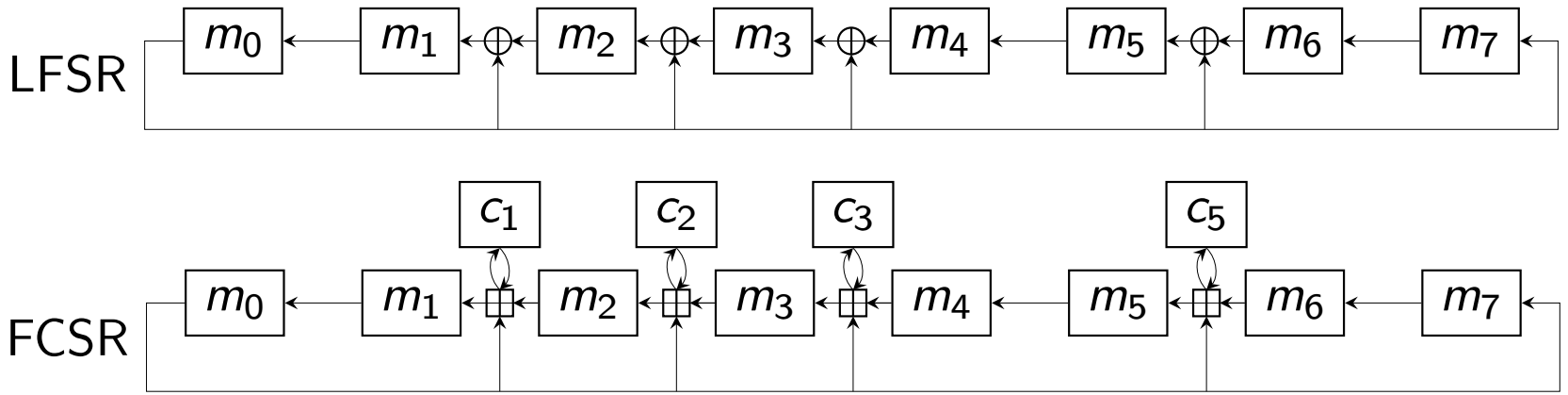
LFSRs:

- modulo 2 addition

FCSRs:

- addition with carry

From LFSRs to FCSRs



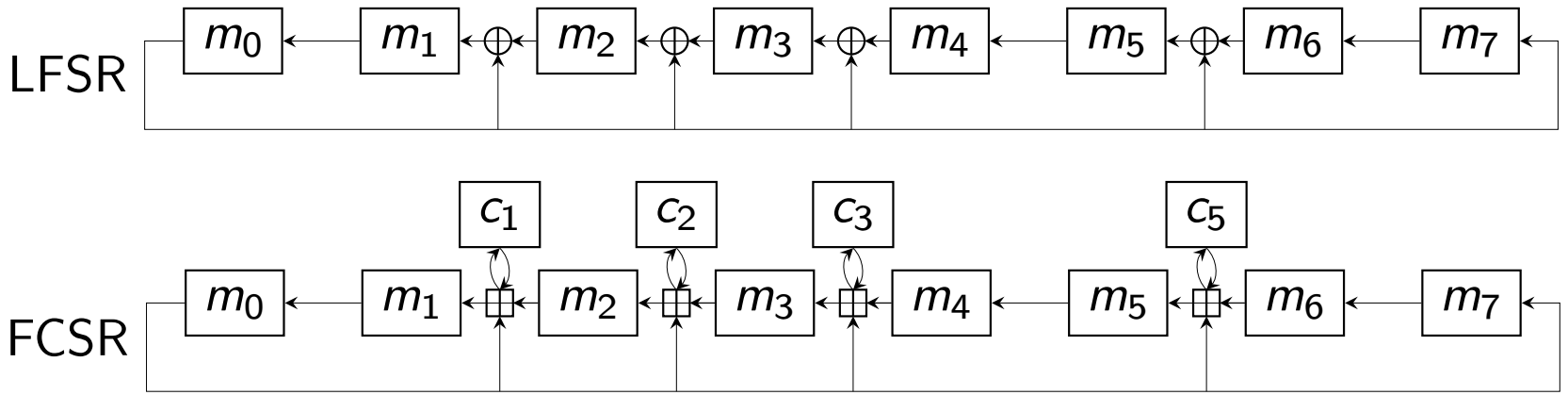
LFSRs:

- modulo 2 addition
- linear transition function

FCSRs:

- addition with carry
- *quadratic* transition function

From LFSRs to FCSRs



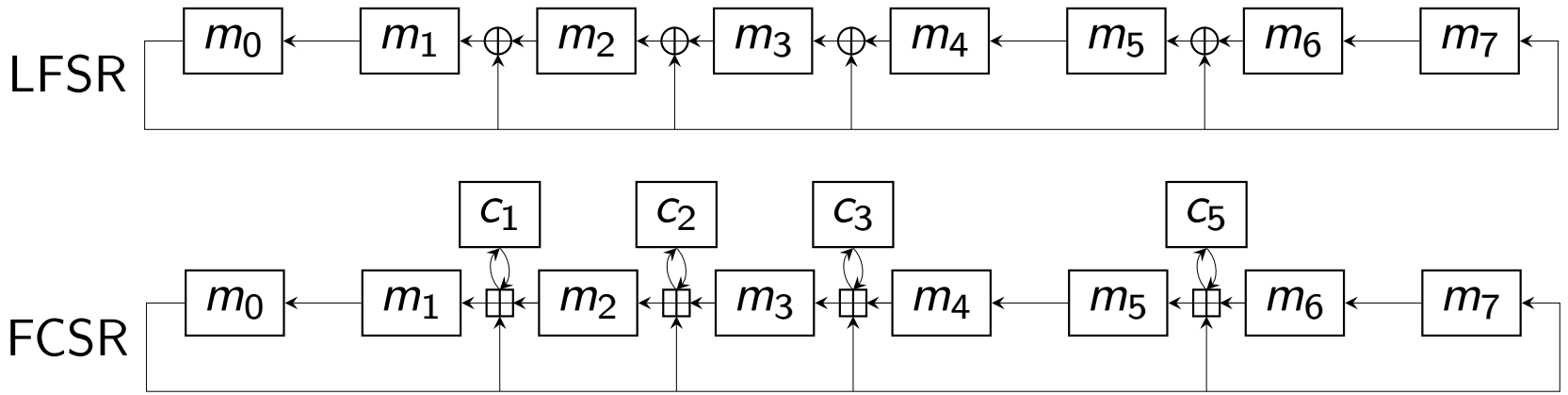
LFSRs:

- modulo 2 addition
- linear transition function
- m-sequence

FCSRs:

- addition with carry
- *quadratic* transition function
- ℓ-sequence

From LFSRs to FCSRs



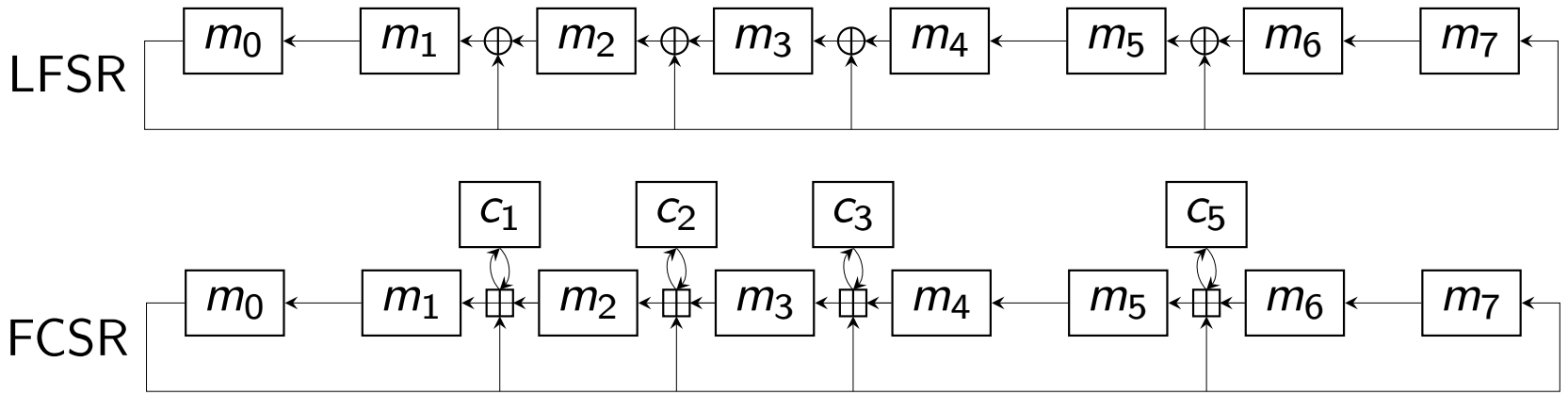
LFSRs:

- modulo 2 addition
- linear transition function
- m-sequence
- rational power series $\sum m(t)X^t = \frac{P(X)}{Q(X)}$

FCSRs:

- addition with carry
- *quadratic* transition function
- ℓ-sequence
- rational 2-adic numbers $\sum m(t)2^t = \frac{p}{q}$

From LFSRs to FCSRs



LFSRs:

- modulo 2 addition
- linear transition function
- m-sequence
- rational power series $\sum m(t)X^t = \frac{P(X)}{Q(X)}$

FCSRs:

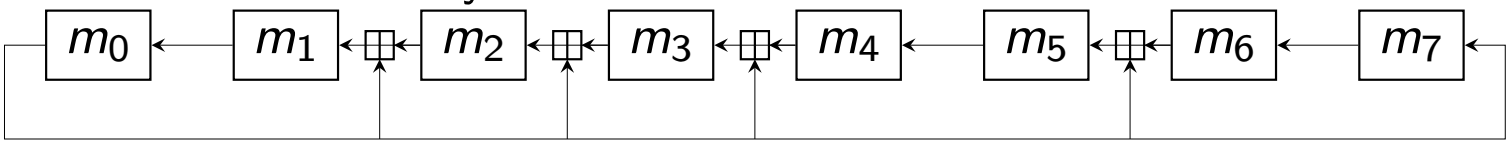
- addition with carry
- *quadratic* transition function
- ℓ-sequence
- rational 2-adic numbers $\sum m(t)2^t = \frac{p}{q}$

Both:

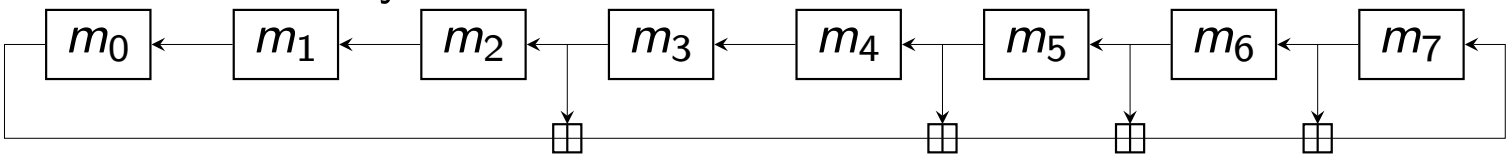
- known period
- good statistical properties

Ring FCSRs

- Galois : one to many

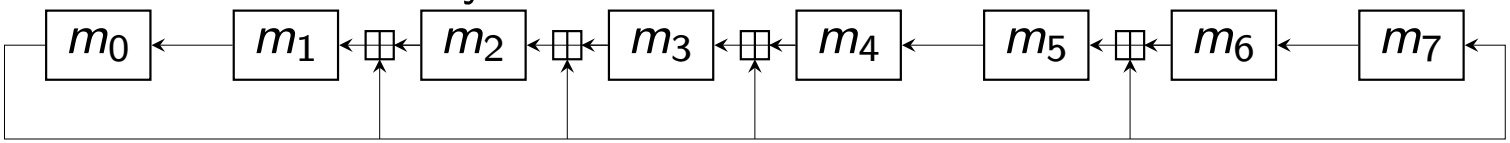


- Fibonacci : many to one

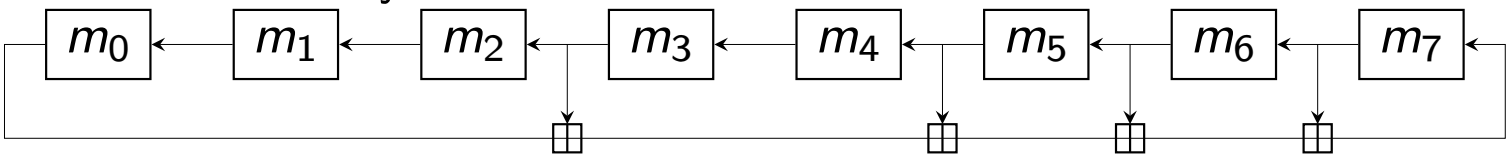


Ring FCSRs

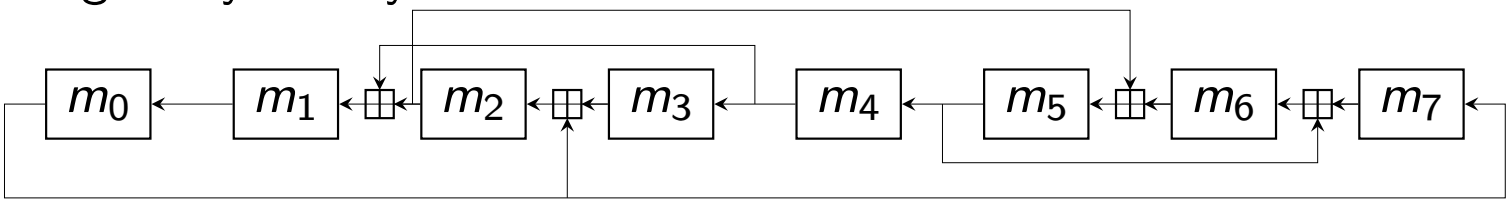
- Galois : one to many



- Fibonacci : many to one

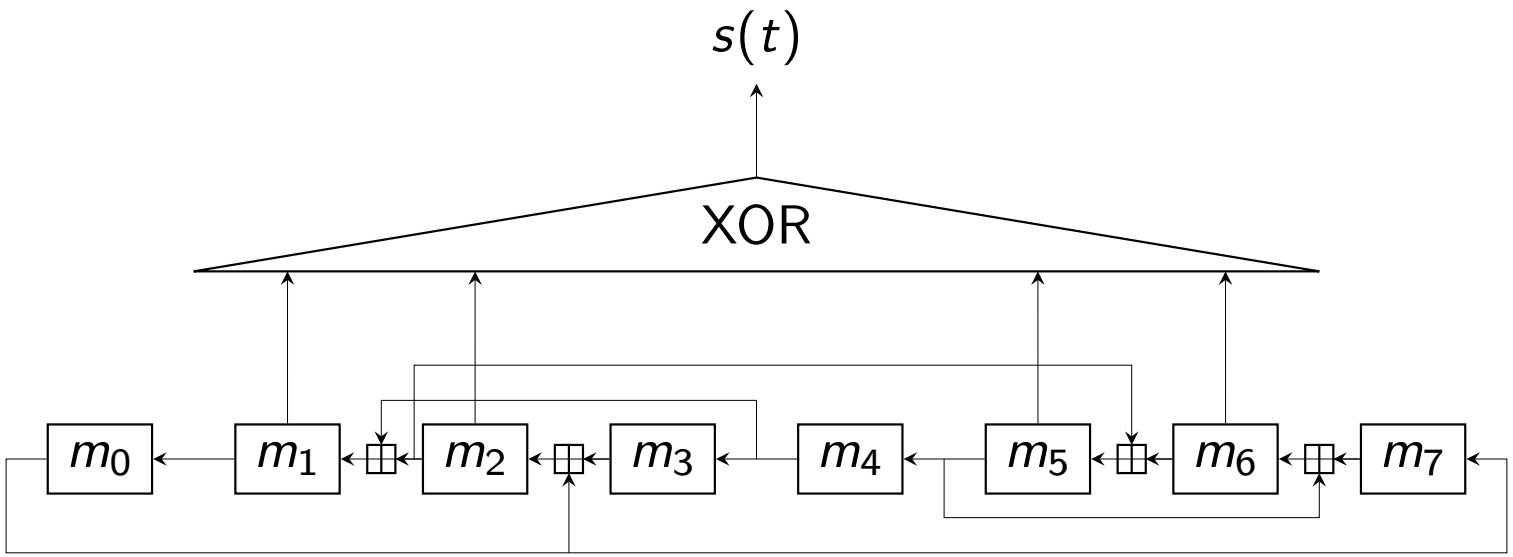


- Ring : any to any



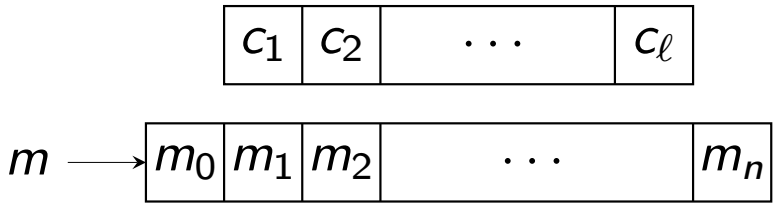
$\Rightarrow$ Same connection integer but better diffusion

F-FCSRs Stream Ciphers



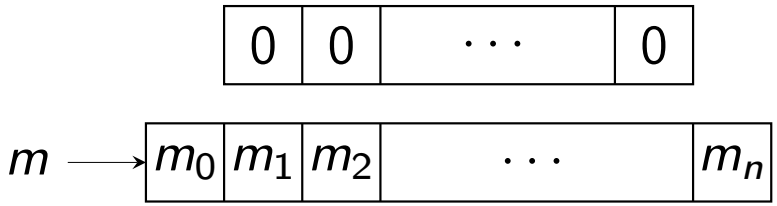
- Apply a linear filter to the main register
- Linearity $\Rightarrow$ highly (correlation) resilient, breaks the 2-adic structure
- As usual, the first bits of the stream are discarded

F-FCSRs as a Round-Function/Sbox



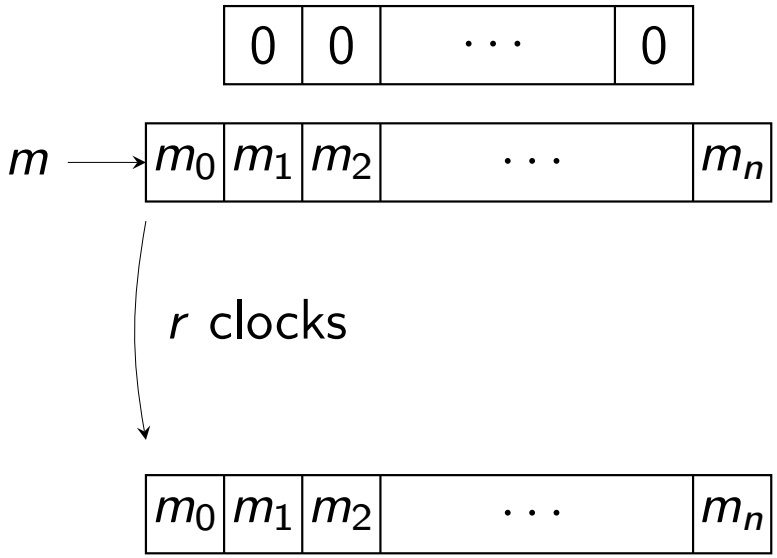
1. Load the input in the main register – possibly padded

F-FCSRs as a Round-Function/Sbox



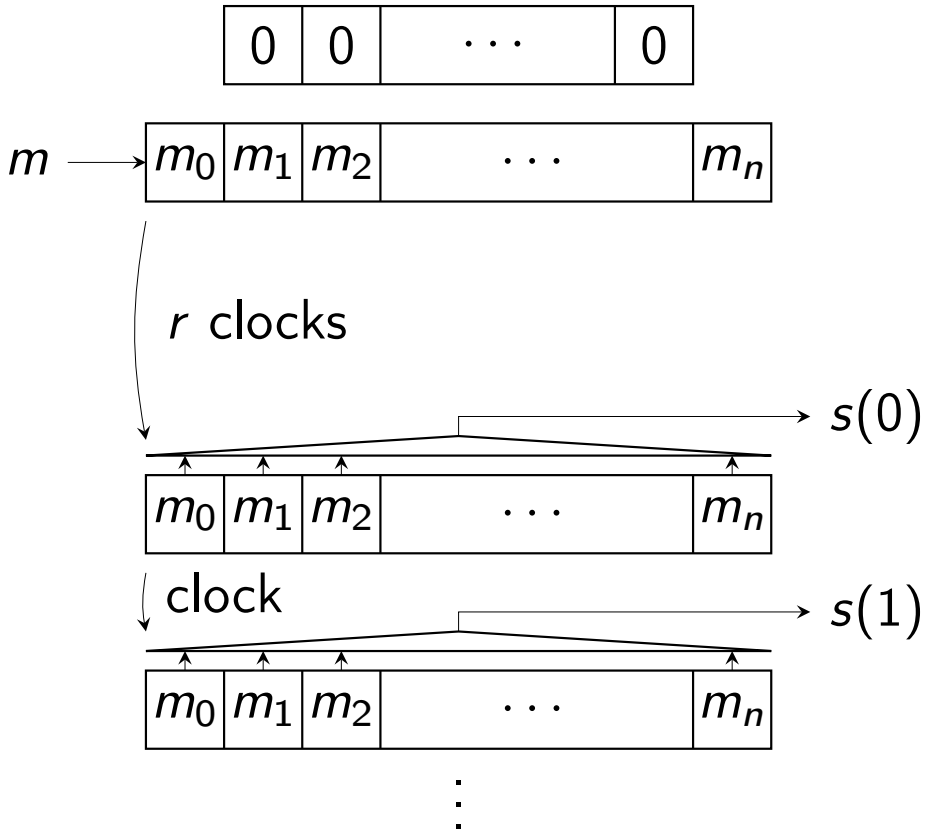
1. Load the input in the main register – possibly padded
2. Set the carry register to 0

F-FCSRs as a Round-Function/Sbox



1. Load the input in the main register – possibly padded
2. Set the carry register to 0
3. Clock the FCSR enough time to ensure full diffusion

F-FCSRs as a Round-Function/Sbox



1. Load the input in the main register – possibly padded
2. Set the carry register to 0
3. Clock the FCSR enough time to ensure full diffusion
4. Extract and Clock until enough bits are outputted

Summary

Notations and Goal

Notations :

- m an input message
- δ a differential
- $m' = m + \delta$
  - Note that if $Supp(m) \cap Supp(\delta) = \emptyset$ then $m' = m \oplus \delta$.

For $x \in \{m, \delta, m'\}$

- $x_i(t)$ the i-th bit of x after t clocks of the FCSR
- X_i the 2-adic serie associated with $x_i = x_i(0)$, e.g. $\Delta_i = \sum_{t=0}^{\infty} \delta_i(t) 2^t$

Notations and Goal

Notations :

- m an input message
- δ a differential
- $m' = m + \delta$
  - Note that if $Supp(m) \cap Supp(\delta) = \emptyset$ then $m' = m \oplus \delta$.

For $x \in \{m, \delta, m'\}$

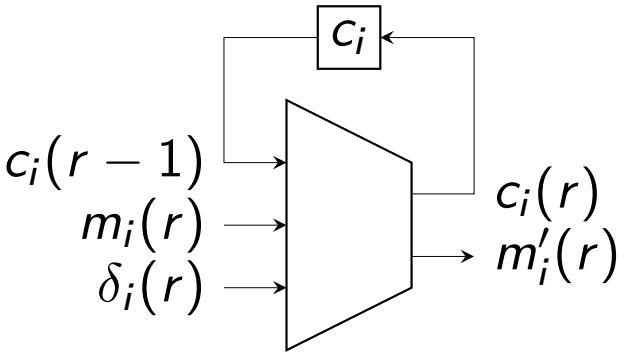
- $x_i(t)$ the i-th bit of x after t clocks of the FCSR
- X_i the 2-adic serie associated with $x_i = x_i(0)$, e.g. $\Delta_i = \sum_{t=0}^{\infty} \delta_i(t) 2^t$

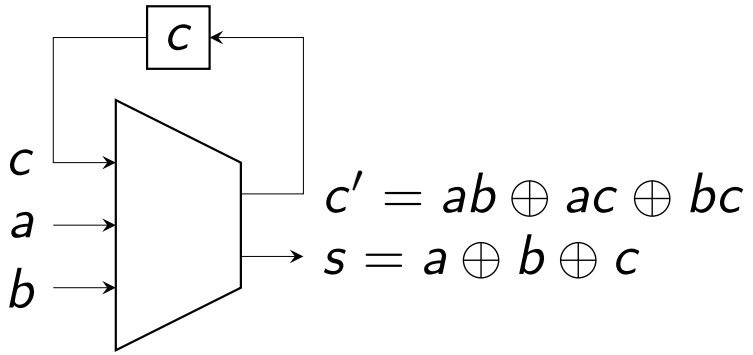
Goal

Estimate the probability $\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1]$.

Adding with carry

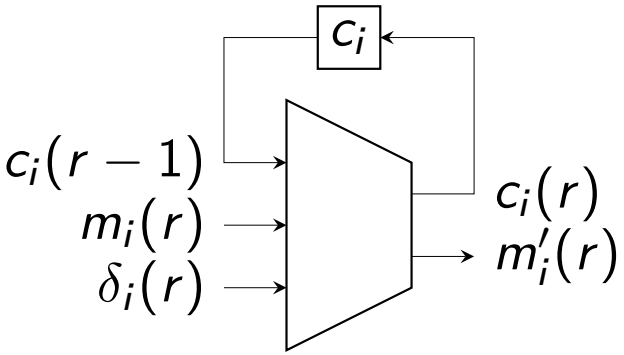
- FCSR linear toward $+ \Rightarrow M'_i = M_i + \Delta_i$
- Hence $m'_i(r)$ can be computed with an adder with carry:

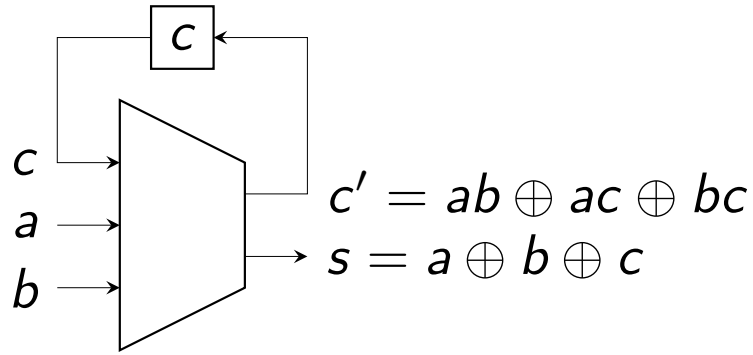




Adding with carry

- FCSR linear toward $+ \Rightarrow M'_i = M_i + \Delta_i$
- Hence $m'_i(r)$ can be computed with an adder with carry:

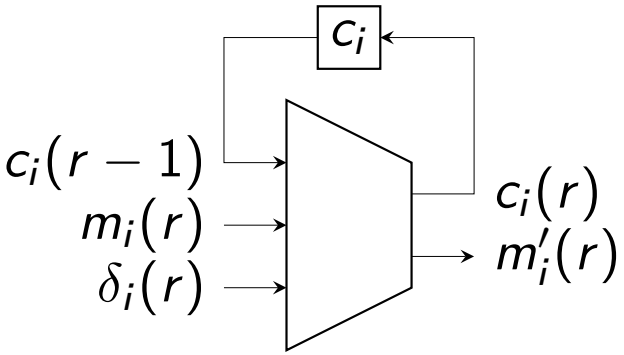


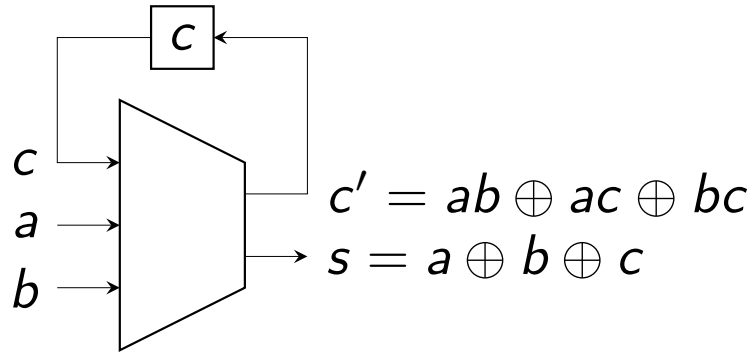


- $\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1] = \Pr[c_i(r-1) = 1]$
- Linear reccurence between $\Pr[c_i(r-1) = 1]$ and $\Pr[c_i(r-2) = 1]$

Adding with carry

- FCSR linear toward $+ \Rightarrow M'_i = M_i + \Delta_i$
- Hence $m'_i(r)$ can be computed with an adder with carry:



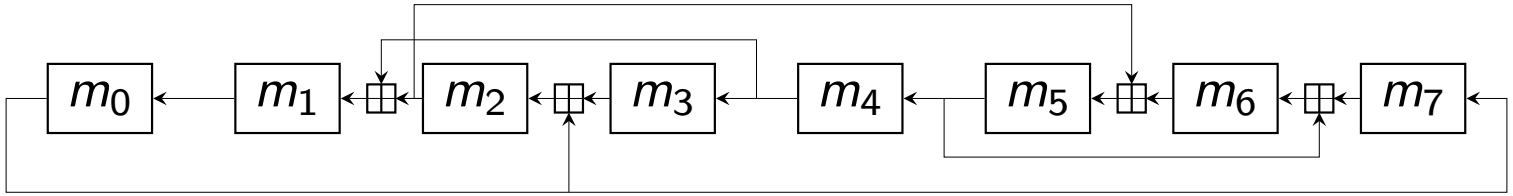


- $\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1] = \Pr[c_i(r-1) = 1]$
- Linear reccurence between $\Pr[c_i(r-1) = 1]$ and $\Pr[c_i(r-2) = 1]$

Result

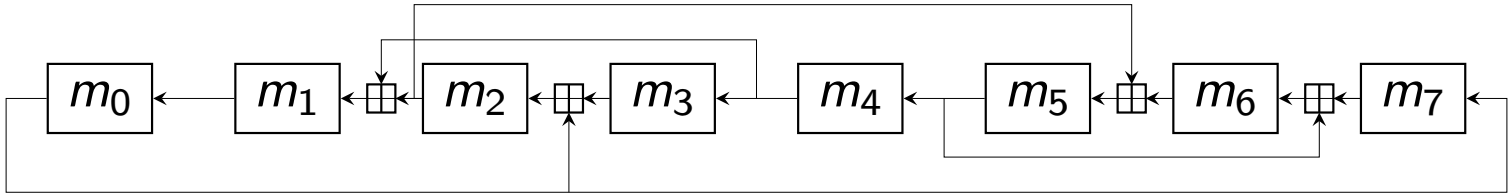
$$\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1] = \frac{1}{2^r}(\Delta_i \bmod 2^r)$$

Simulation on a small FCSR



- connection integer $q = 347$
- diameter $d = 5$ (full diffusion)
- clocked $r = d + 4 = 9$ times

Simulation on a small FCSR

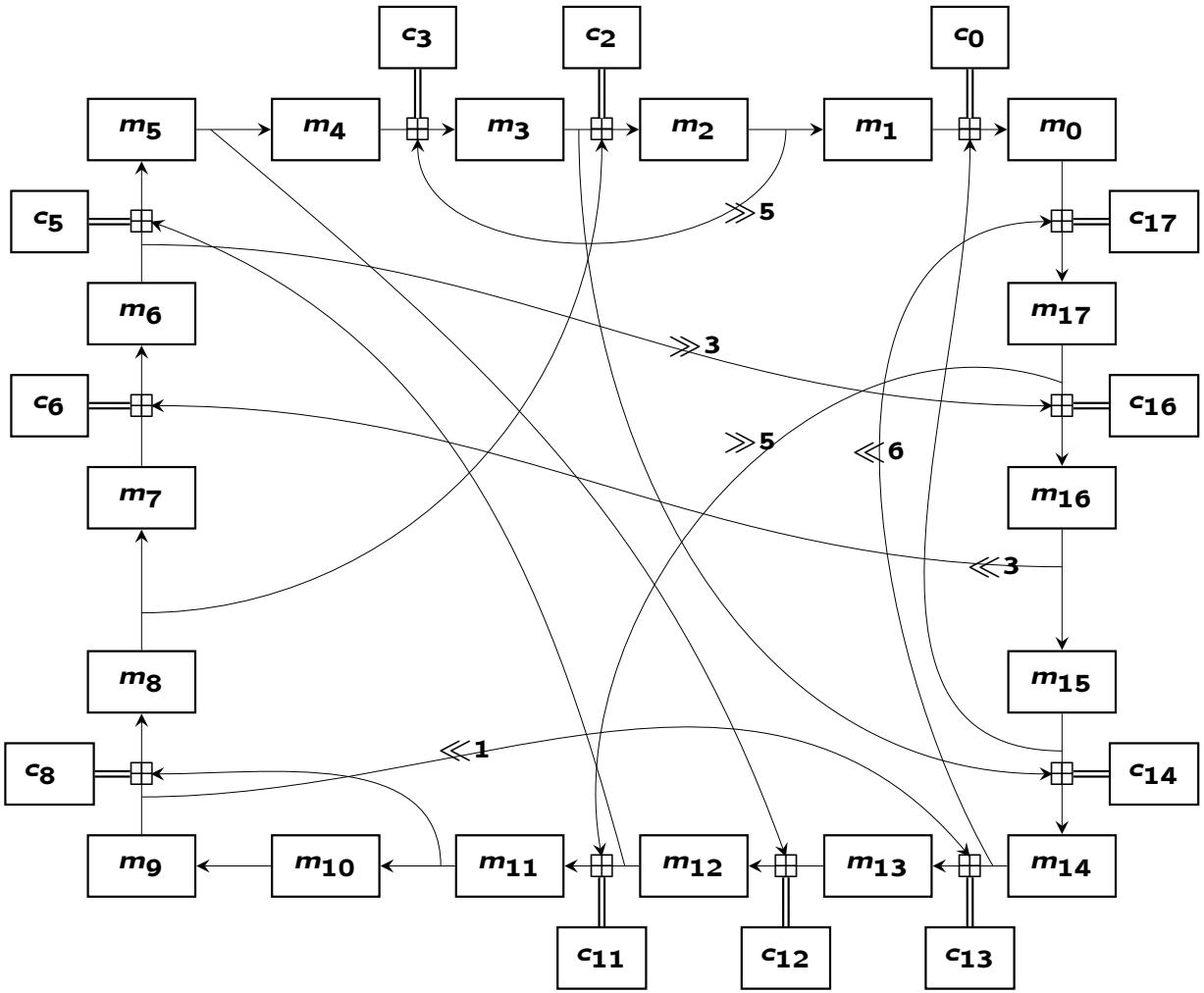


- connection integer $q = 347$
- diameter $d = 5$ (full diffusion)
- clocked $r = d + 4 = 9$ times

	$\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1]$ with $\delta = (0,1,0,0,0,0,0,0)$							
	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7
theoric	0.660	0.830	0.195	0.438	0.219	0.109	0.859	0.320
observed	0.656	0.828	0.203	0.461	0.227	0.117	0.867	0.313
difference ($\times 10^{-2}$)	0.391	0.195	0.781	2.34	0.781	0.781	0.781	0.781

$\Rightarrow$ Small difference between theory and observation

The GLUON-64 FCSR



18 blocs of 8 bits

FCSR of 144 bits

+ 73 bits of carries

Simulation on the GLUON-64 FCSR

GLUON : lightweight hash function family based on the sponge construction and FCSRs.

- size $n = 144$ bits with $\ell = 73$ carry bits
- connection integer
  $q = 270133361799904687777425461649779817670388 29$
- diameter $d = 29$
- clocked $r = d + 4 = 33$ times

Simulation on the GLUON-64 FCSR

GLUON : lightweight hash function family based on the sponge construction and FCSRs.

- size $n = 144$ bits with $\ell = 73$ carry bits
- connection integer
  $q = 27013336179990468777742546164977981767038829$
- diameter $d = 29$
- clocked $r = d + 4 = 33$ times

	$\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1]$ with $\delta = (1, 0, 0, ..., 0)$							
	bit 0	bit 1	bit 2	bit 3	bit 120	bit 121	bit 122	bit 123
theoric	0.623	0.868	0.329	0.402	0.817	0.308	0.214	0.402
observed	0.632	0.875	0.330	0.392	0.815	0.323	0.219	0.399
diff. ($\times 10^{-2}$)	0.928	0.671	0.0794	1.07	0.232	1.58	0.476	0.281

Again, small difference between theory and observation

Summary

A famous lemma

The linear filter :

- denoted f,
- xors some of the $m_i(r)$,
- output a single bit $s(r) = f\big(m(r)\big)$

A famous lemma

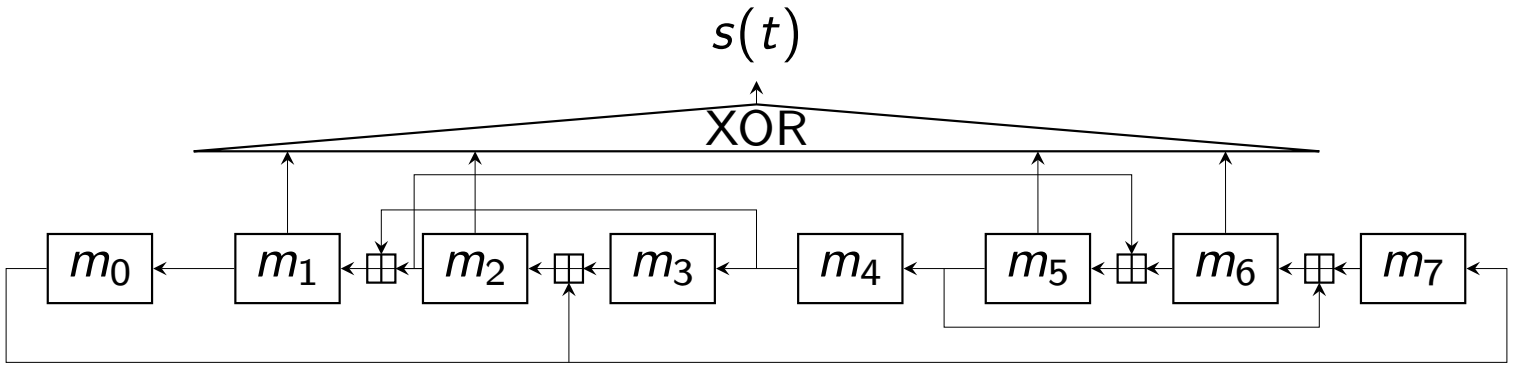
The linear filter :

- denoted f,
- xors some of the $m_i(r)$,
- output a single bit $s(r) = f\left(m(r)\right)$

But :

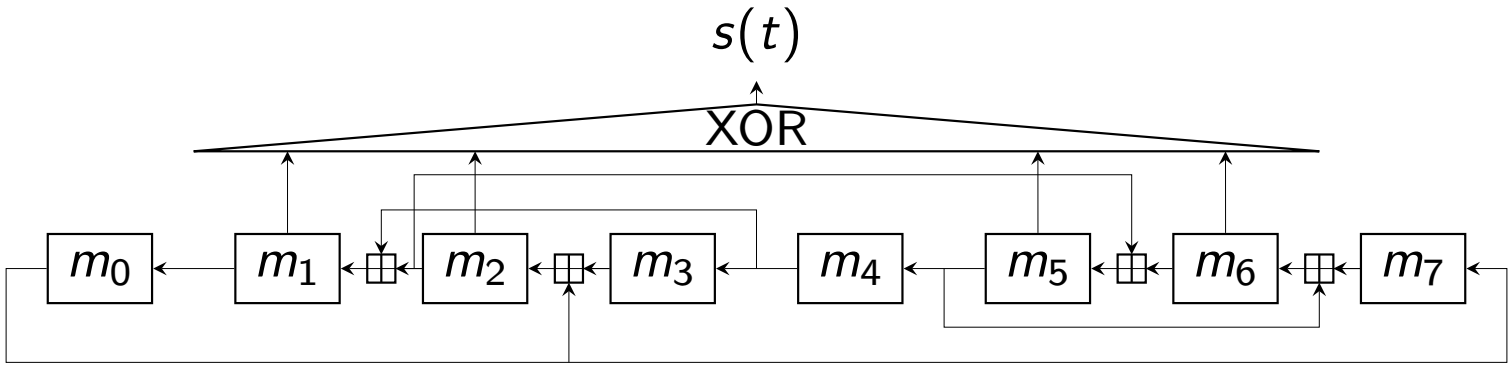
- $\Pr[m'_i(r) \oplus m_i(r) \oplus \delta_i(r) = 1]$ known for indices i at the filter input
- Matsui's *Piling-up lemma* $\Rightarrow \Pr\left[f\left(m'(r)\right) \oplus f\left(m(r)\right) \oplus f\left(\delta(r)\right) = 1\right]$

Simulation on the small FCSR



- connection integer $q = 347$
- clocked $r = 9$ times
- 4-bit filter $f(m) = m_1 \oplus m_2 \oplus m_5 \oplus m_6$

Simulation on the small FCSR



- connection integer $q = 347$
- clocked $r = 9$ times
- 4-bit filter $f(m) = m_1 \oplus m_2 \oplus m_5 \oplus m_6$

	$\Pr[f(m'(r)) \oplus f(m(r)) \oplus f(\delta(r)) = 1]$		
Differential δ	Theoric	Observed	Difference ($\times 10^{-2}$)
$(1,0,0,0,0,0,0,0)$	0.491	0.516	2.43
$(0,1,0,0,0,0,0,0)$	0.387	0.391	0.357
$(0,0,1,0,0,0,0,0)$	0.483	0.547	6.34
$(0,0,0,1,0,0,0,0)$	0.458	0.438	2.08

Problem : the bias toward 1/2 may be of wrongly signed.

Simulation on the GLUON-64 FCSR

- size $n = 144$ bits with $\ell = 73$ carry bits
- connection integer
  $q = 27013336179990468777742546164977981767038829$
- clocked $r = 33$ times
- filter $f : \mathbb{F}_2^{96} \to \mathbb{F}_2^8$

Simulation on the GLUON-64 FCSR

- size $n = 144$ bits with $\ell = 73$ carry bits
- connection integer
  $q = 27013336179990468777742546164977981767038829$
- clocked $r = 33$ times
- filter $f : \mathbb{F}_2^{96} \to \mathbb{F}_2^8$

	$\Pr[f(m'(r)) \oplus f(m(r)) \oplus f(\delta(r)) = 1]$ with $\delta = (1, 0, 0, ..., 0)$				
	bit 0	bit 1	bit 2	bit 3	bit 4
theoric	0.499961	0.504082	0.496059	0.500014	0.500464
observed	0.500000	0.498047	0.483887	0.496094	0.502930
difference ($\times 10^{-2}$)	0.00386238	0.603485	1.21722	0.392056	0.246525

Same Problem : the bias toward 1/2 may be of wrongly signed.

Can this break GLUON?

- Prblem with the bias sign
- Bias changes with each clock of the FCSR / each outputted word
- GLUON design $\Rightarrow$ differential only on the first input word
- Difficult to reverse the FCSR

Summary

What exists...

Paper of Tian Tian and Wen-Feng Qi : Linearity properties of binary FCSR sequences, vol. 52 of *Design, Codes and Cryptography* (2009).

- Large linear bias based on low-weight multiple of the connection integer q

E.g.:

- ℓ-sequence $\sum a(t)2^t$
- bias of $a(t+u) \oplus a(t+v) \oplus a(t) = a(t+r)$
- $p = 2^r + 2^{r-u} + 2^{r-v} - 1$
- $q|p \Rightarrow$ large bias (about $\frac{1}{3}$)

... And what we want to do

Tian and Qi:

- One ℓ-sequence:
  $\sum a(t)2^t$
- Temporal bias:
  $a(t+u) \oplus a(t+v) \oplus \cdots = 0$

... And what we want to do

Tian and Qi:

- One ℓ-sequence: $\sum a(t)2^t$
- Temporal bias: $a(t+u) \oplus a(t+v) \oplus \cdots = 0$

Us:

- Multiple ℓ-sequences: $\sum m_i(t)2^t$
- Spatial bias: $m_i(t) \oplus m_j(t) \oplus \cdots = 0$

... And what we want to do

Tian and Qi:

- One ℓ-sequence: $\sum a(t)2^t$
- Temporal bias: $a(t+u) \oplus a(t+v) \oplus \cdots = 0$

Us:

- Multiple ℓ-sequences: $\sum m_i(t)2^t$
- Spatial bias: $m_i(t) \oplus m_j(t) \oplus \cdots = 0$

Why it might work:

- Convert the spatial back to temporal
- Every ℓ-sequences $\sum m_i(t)2^t$: temporal shifts of a single ℓ-sequence

... And what we want to do

Tian and Qi:

- One ℓ-sequence: $\sum a(t)2^t$
- Temporal bias: $a(t+u) \oplus a(t+v) \oplus \cdots = 0$

Us:

- Multiple ℓ-sequences: $\sum m_i(t)2^t$
- Spatial bias: $m_i(t) \oplus m_j(t) \oplus \cdots = 0$

Why it might work:

- Convert the spatial back to temporal
- Every ℓ-sequences $\sum m_i(t)2^t$: temporal shifts of a single ℓ-sequence

Why it doesn't work:

- Some things are true modulo q but not modulo p

Great hope it will work, though.

Conclusion

- FCSRs : a non-linear primitive with many good properties
- Differential and linear bias inherent to the 2-adic structure, though

Yet to do:

- if linear works, find an efficient way to find low-weight multiple in our case
- find counter-measures that totally breaks the 2-adic structure of FCSRs