Stream Ciphers for Constrained Environments

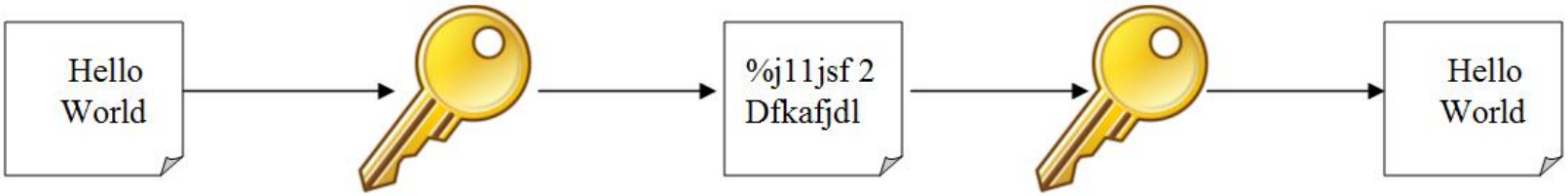
Meltem Sönmez Turan

National Institute of Standards and Technology

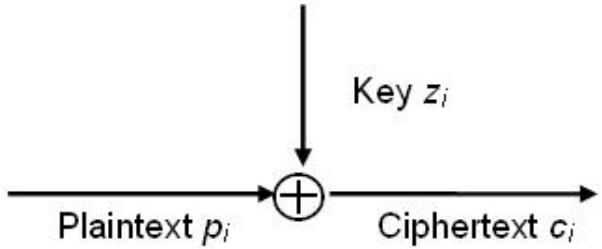
November 7, 2011

Stream Ciphers

- Symmetric key cryptosystems

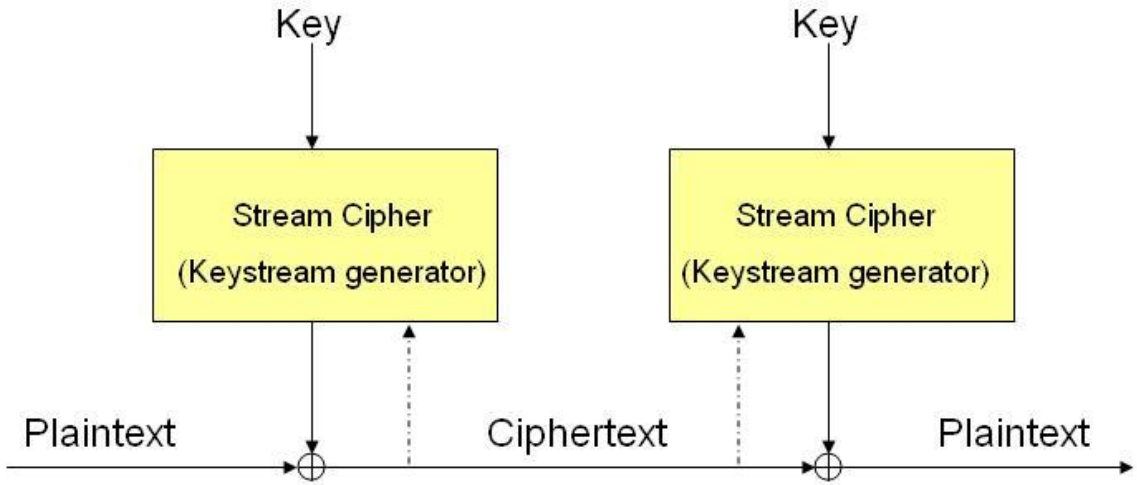


- Stream ciphers simulate the idea of unconditionally secure One Time Pad.



Stream Ciphers

Partition the plaintext into bits or words (e.g. 16, 32 bits) and encrypt each block using a **time-varying** encryption function.

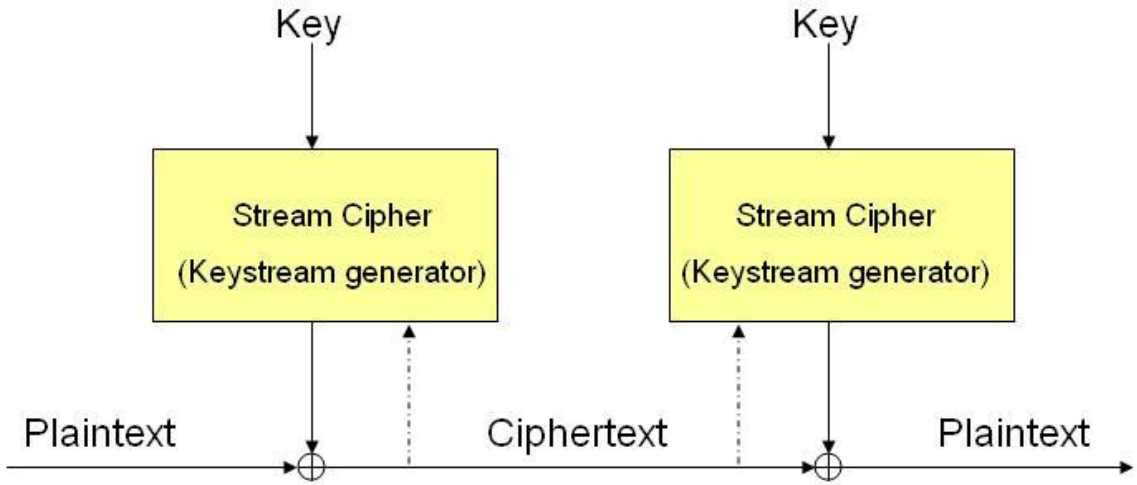


Two types of stream ciphers

- Synchronous stream ciphers
- Self-synchronizing stream ciphers

Stream Ciphers

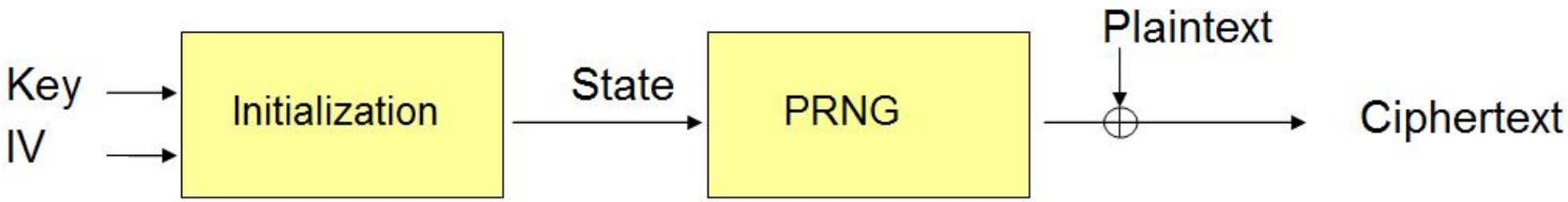
Partition the plaintext into bits or words (e.g. 16, 32 bits) and encrypt each block using a **time-varying** encryption function.



Two types of stream ciphers

- **Synchronous stream ciphers**
- Self-synchronizing stream ciphers

Generic Structure of Synchronous Stream Ciphers



Key/IV Initialization

- For correct decryption, sender and receiver must be synchronized, i.e. they must have the same internal state at time t.
- If ciphertext bits are deleted/inserted, then synchronization is lost and rest of the ciphertext is useless.
- A **Key/IV initialization function** is used for resynchronization.

PRNG

-

Properties of Synchronous Stream Ciphers

No Error Propagation

- A change in the ciphertext bit affects only the corresponding bit in the deciphered plaintext.

$$\begin{array}{r} \text{Encryption} \\ p_1, p_2, \ldots, p_n \\ \oplus \quad z_1, z_2, \ldots, z_n \\ \hline c_1, c_2, \ldots, c_n \end{array} \rightarrow \text{Insecure Channel} \rightarrow \begin{array}{r} \text{Decryption} \\ c_1, \ c_2', \ldots, c_n \\ \oplus \quad z_1, z_2, \ldots, z_n \\ \hline p_1, \ p_2', \ldots, p_n \end{array}$$

- Suitable for encrypting voice and video

Well-known Examples

- E0 used in Bluetooth
- RC4 used in Secure Socket Layer (SSL) protocol
- A5/1 used in the Global System for Mobile (GSM) Communication

Constrained Environments

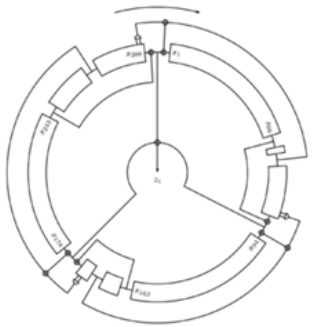
Requirements

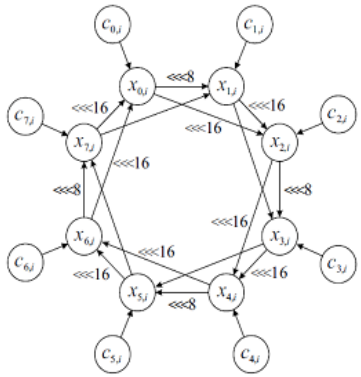
- Small chip sizes
- Less(peak and average) energy consumption (limited lifetime of the devices)
- Short processing times
- High throughput is not always necessary.

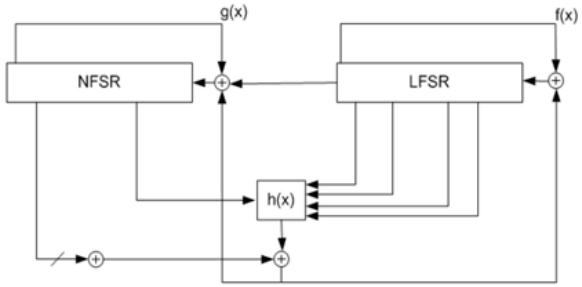
Stream Ciphers for Constrained Environments

Design approaches

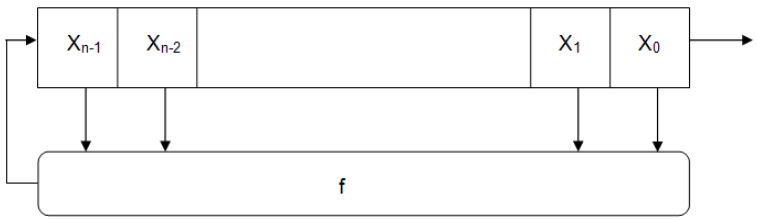
- Ad hoc designs







- Bit oriented
- Common building block: Feedback shift registers



Key Size

Key and State Size

- Key size determines the security level
- Large key size $\equiv$ High cost
- **Time Memory Tradeoff Attacks:** speed up exhaustive search by using memory. To resists TMTO attacks, the size of the internal state should at least be twice the key size.

Ecrypt eSTREAM Project

4-year (between October 2004 - May 2008) network of excellence funded project by European Network of Excellence for Cryptology (ECRYPT)
Goals:

- to identify new stream ciphers that might be suitable for widespread adoption
- to stimulate work in stream ciphers.

eStream Profile II

Call for primitives within two profiles

- Profile I: for software applications with high throughput requirements.
- Profile II: for hardware applications with restricted resources such as limited storage, gate count, or power consumption.

Profile II

- Key size is 80 bits.
- eStream received 25 candidates for Profile II.

eStream Profile II Finalists

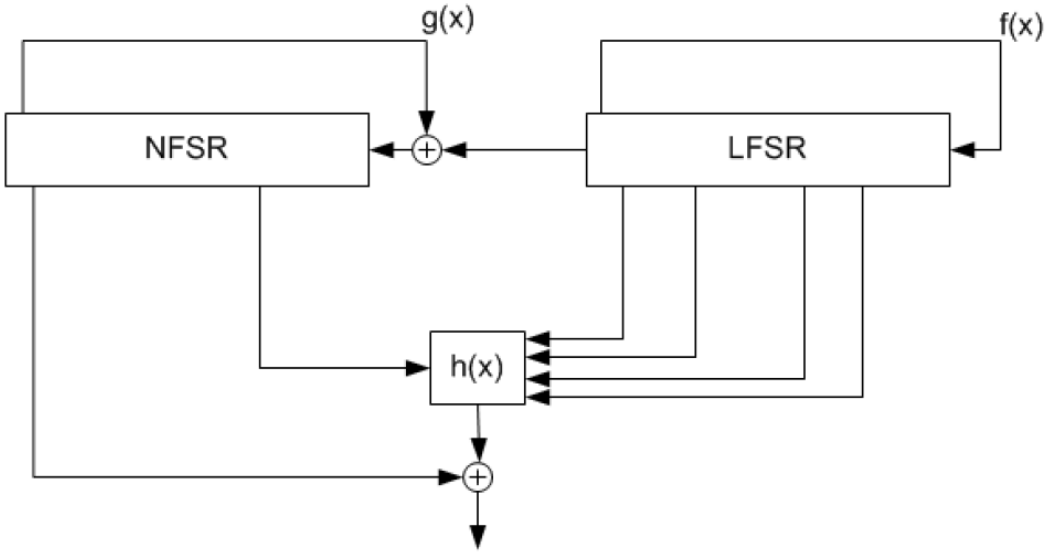
Finalists are;

- **Grain** by Hell, Johansson and Meier
- **F-FCSR**-H by Berger, Arnault and Lauradoux (Broken, and removed from the portfolio)
- **Trivium** by Canniere and Preneel
- **Mickey** by Babbage and Dodd

Grain

- Based on bit oriented FSRs. Main components: 80-bit LFSR, 80-bit NFSR, Boolean function
- Internal state size of 160 bits.
- Well studied

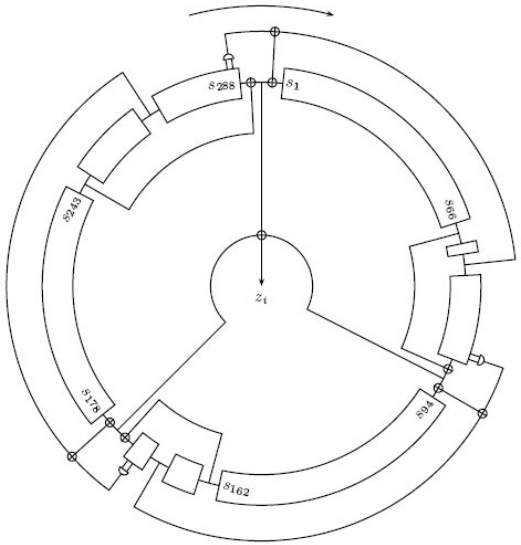
General Structure



Trivium

- Based on bit oriented FSRs. Main components: Three nonlinear shift registers
- Internal state size of 288 bits.
- Well studied

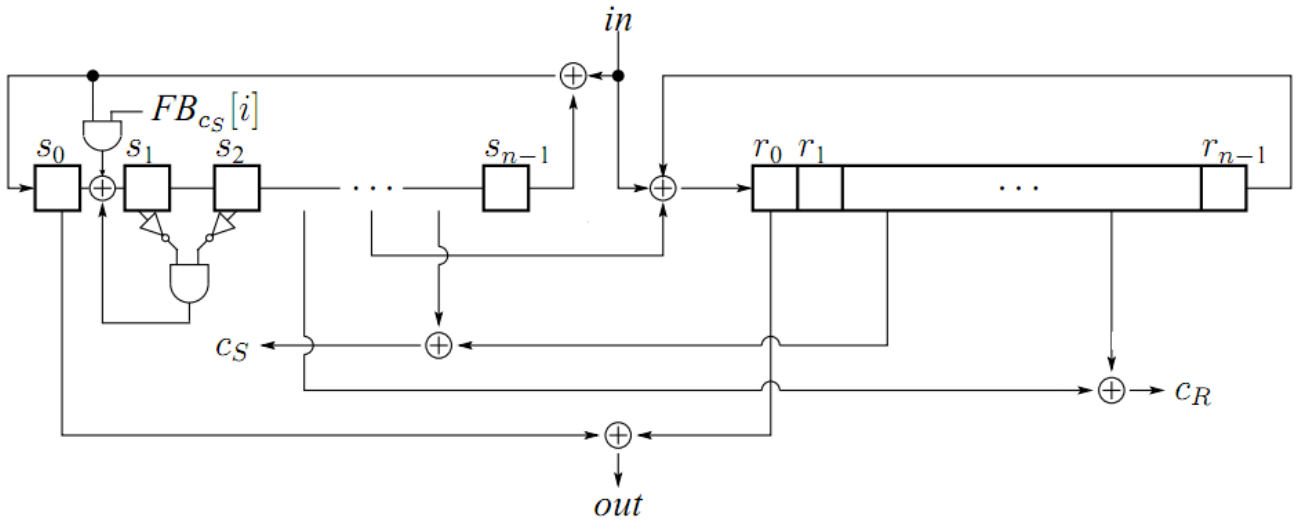
General Structure



Mickey

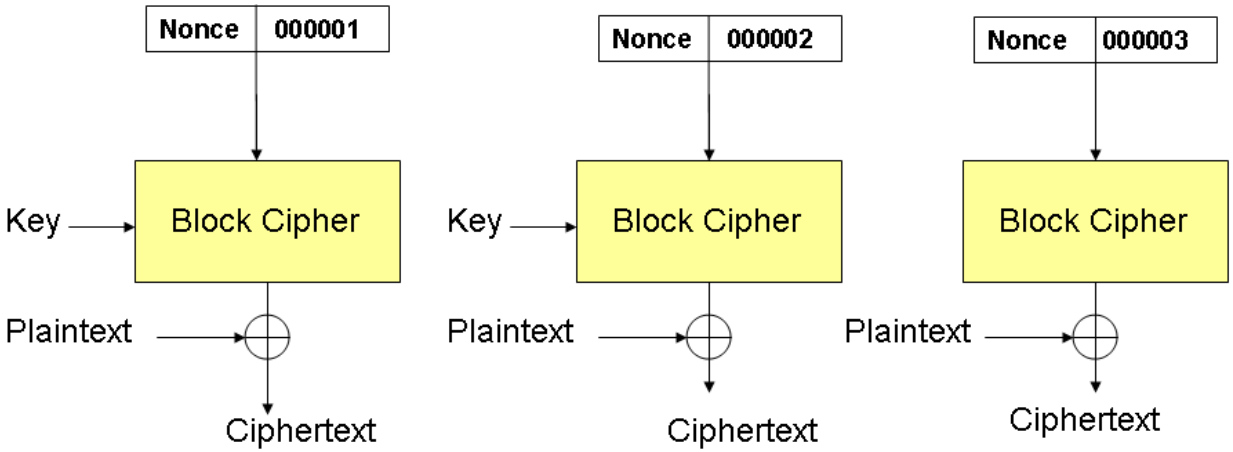
- Based on bit oriented registers. Main components: Two registers of size 100
- Internal state size of 200 bits.
- Not well studied.

General Structure



Advanced Encryption Standard

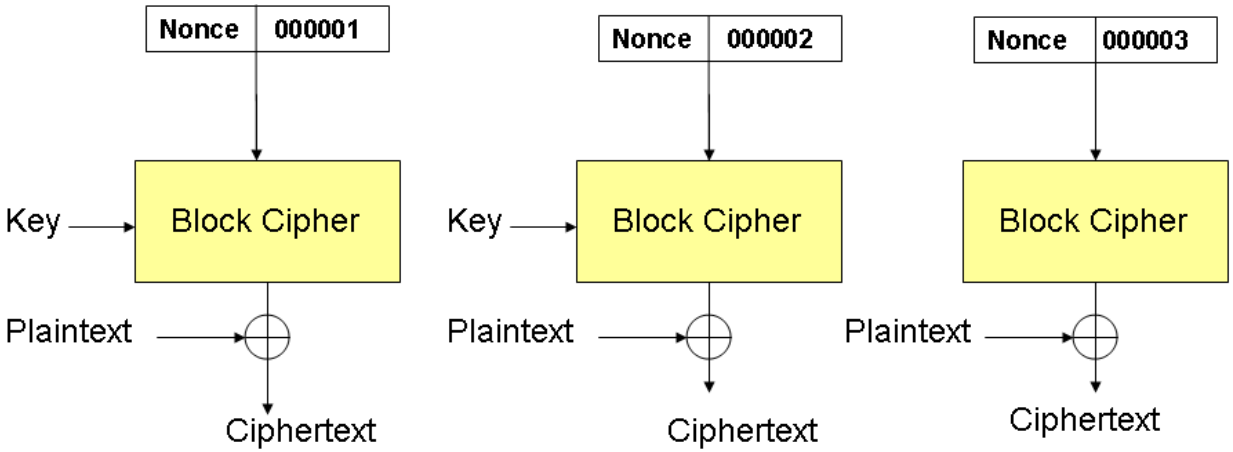
AES - Counter Mode



- Secure, Well understood, Standardized
- Fast and efficient
- Meets most needs.

Advanced Encryption Standard

AES - Counter Mode



- Secure, Well understood, Standardized
- Fast and efficient
- Meets most needs.

Not efficient enough for constrained devices!

Call for Feedback

There is no NIST approved stream cipher.

Do we need dedicated stream ciphers?

- More cryptanalytic results on stream ciphers and their impact on practical applications
- Performance comparison of lightweight stream ciphers and AES Counter Mode
- Experimental results (on power consumption, storage, complexity etc.) for environments that AES cannot be used.

THANKS!

streamcipher@nist.gov