MAS 433: Cryptography

Lecture 10

Stream Cipher

Wu Hongjun

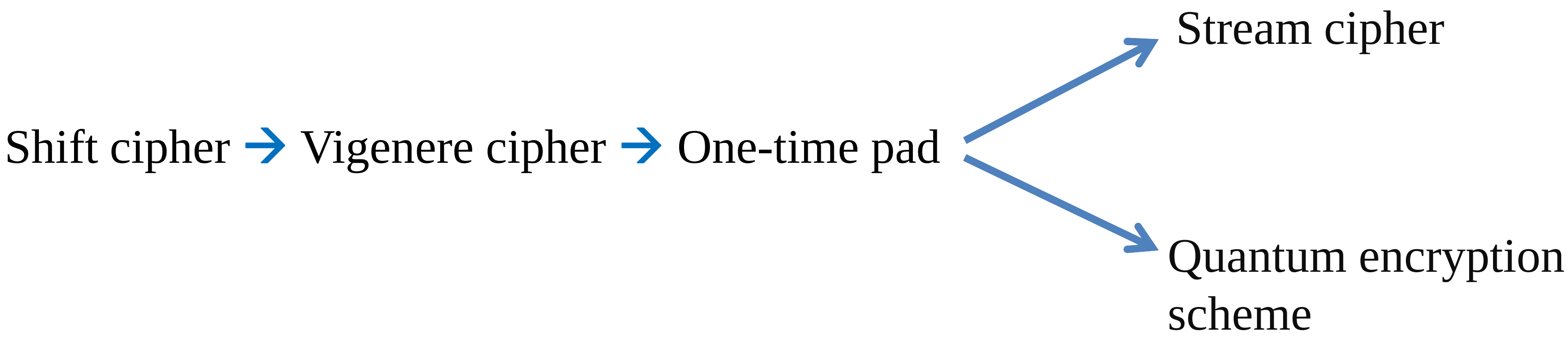
Lecture Outline

- Classical ciphers
- Symmetric key encryption
  - One-time pad & information theory
  - Block cipher
  - **Stream cipher**
    - **Block cipher based stream cipher**
    - **LFSR based stream cipher**
    - **NLFSR based stream cipher**
- Hash function and Message Authentication Code
- Public key encryption
- Digital signature
- Key establishment and management
- Introduction to other cryptographic topics

Recommended Reading

- HAC Chapter 6
- Wikipedia
  - Stream cipher
    http://en.wikipedia.org/wiki/Stream_cipher
  - A5/1
    http://en.wikipedia.org/wiki/A5/1
  - RC4
    http://en.wikipedia.org/wiki/RC4
  - eSTREAM
    http://en.wikipedia.org/wiki/ESTREAM

One-time pad → more practical



Shift cipher → Vigenere cipher → One-time pad

Stream cipher

Quantum encryption scheme

One-time pad:

key length = message length

Stream cipher:

generate a long keystream from a short key and IV

Quantum encryption:

generate a long keystream with quantum uncertainty principle,
a short key is required for ensuring the authenticity of keystream

Stream Cipher Classification

- Synchronous stream cipher
  - Keystream is generated independent of message/ciphertext
  - Example: Block cipher in OFB and CTR modes
- Self-synchronizing (asynchronous) stream cipher
  - Keystream is generated from the key and previous *N* ciphertext bits
  - Example: Block cipher in CFB mode
- Nowadays synchronous stream cipher more popular than self-synchronizing stream cipher

Synchronous Stream Cipher

- Advantages (of synchronous stream ciphers)
  - Keystream can be precomputed
    - Encryption/decryption can be extremely fast when plaintext or ciphertext arrived (only XOR)
      - Suitable for real-time applications
    - Keystream be generated at a secure place, and keystream be used at a less secure place for encryption/decryption
      - Suitable for some military applications
  - No partial block problem
  - Dedicated stream cipher can be much more efficient than block cipher for the same security level

Synchronous Stream Cipher

- Generally two major components
  - Initialization
    - Load the key and IV into the state
    - Mixing key and IV into a random state before generating keystream
  - Keystream generation
    - Update the state at each step
    - Generate keystream bit (or word) at each step

Block cipher based stream cipher

- CFB → Asynchronous stream cipher
- OFB
- CTR

$\left.\begin{array}{l}\text{OFB}\\ \text{CTR}\end{array}\right\}$ Synchronous stream cipher

Only as efficient as block cipher

LFSR based stream cipher

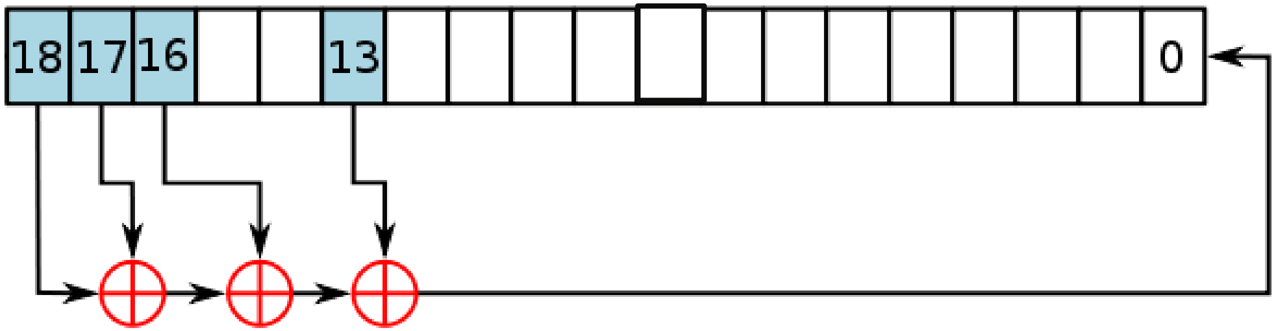
- Linear feedback shift register (LFSR)
  - Normally defined by a primitive polynomial over GF(2), called feedback polynomial or characteristics polynomial
    - A polynomial over GF(2) (with degree n) is primitive if it has order 2^n-1.
      - The order of a polynomial $f(x)$ for which $f(0)$ is not 0 is the smallest integer e for which $f(x)$ divides x^e+1.
        - » Example: x^2+x+1 has order 3 since $(x^2+x+1)(x+1)=x^3+1$.

          $2^2-1 = 3$, so x^2+x+1 is primitive
    - A primitive polynomial is also irreducible

LFSR based stream cipher

- Linear feedback shift register (LFSR)
  - Example: LFSR with primitive polynomial

$$x^{18} + x^{17} + x^{16} + x^{13} + 1$$



  - The period of the above LFSR is maximal: 2^{18}-1

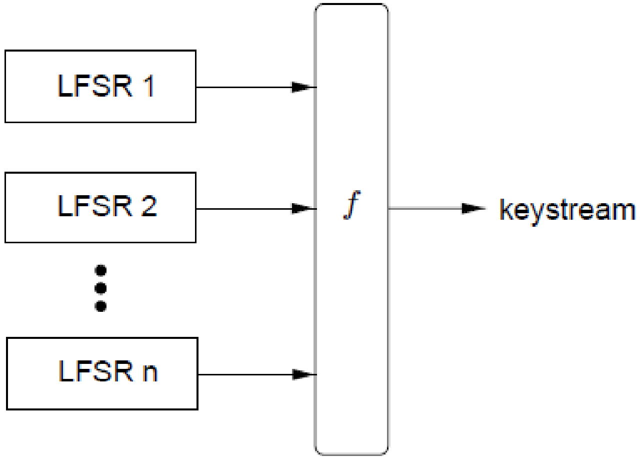
LFSR based stream cipher

- Linear feedback shift register (LFSR)
  - Advantages of using LFSR
    - Maximum period

      (with a primitive feedback polynomial)
    - Easy to construct in circuits
  - Disadvantage
    - linear

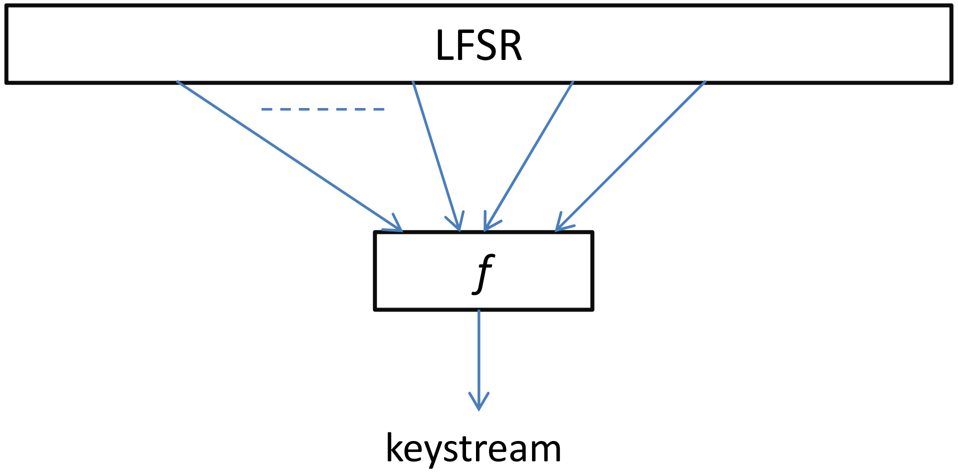
LFSR based stream cipher

- Two approaches to design LFSR based stream ciphers
  - Approach 1: Nonlinear combination
    - Several short LFSRs with co-prime periods
    - A nonlinear combination function is applied to generate keystream



LFSR based stream cipher

- Two approaches to design LFSR based stream ciphers
  - Approach two: nonlinear filter
    - A nonlinear filter function is applied to some bit positions (better with co-prime distances between them) of a long LFSR



LFSR based stream cipher

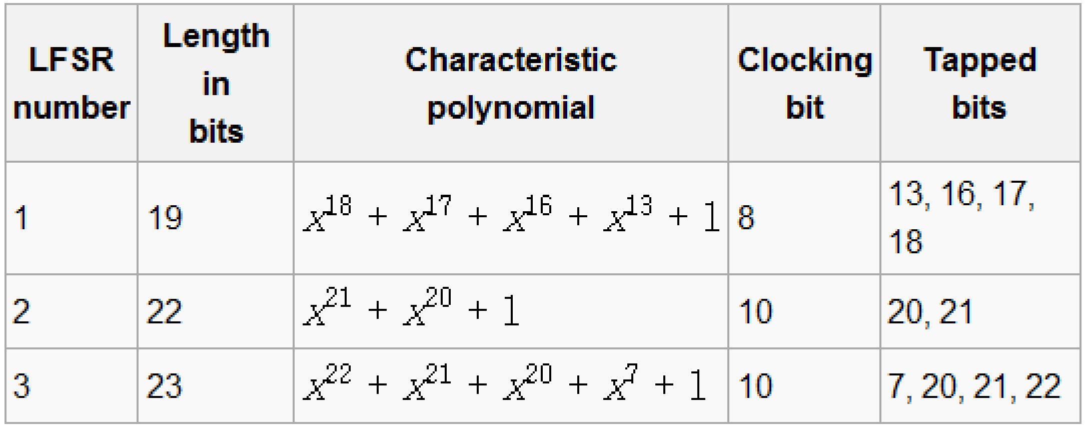
- Caution
  - The design of the nonlinear combination function or filter function is crucial
    - must satisfy a number of requirements:
      - Balanced
      - High nonlinearity
      - High algebraic degree
      - High algebraic immunity
      - …..
  - Researchers in academic have been actively working on LFSR-based stream ciphers for more than twenty years
    - but it is better to avoid this type of stream ciphers due to the linearity in LFSR.
    - LFSR based stream cipher SNOW-3G is a backup cipher for the 3G mobile telecommunication

Stream cipher based on irregularly clocked LFSR

- One way to design simple & strong stream cipher from LFSR is to clock LFSR irregularly, so as to introduce nonlinearity into the LFSR
  - Example: A5/1 for GSM mobile network (1987 --)
    - Simple structure,
    - Reasonable security
      - Some western European countries wanted a strong cipher, but some did not want …
      - Key size reduced from 64 bits to 56 bits
      - But with only 64-bit state in order to reduce the hardware cost.

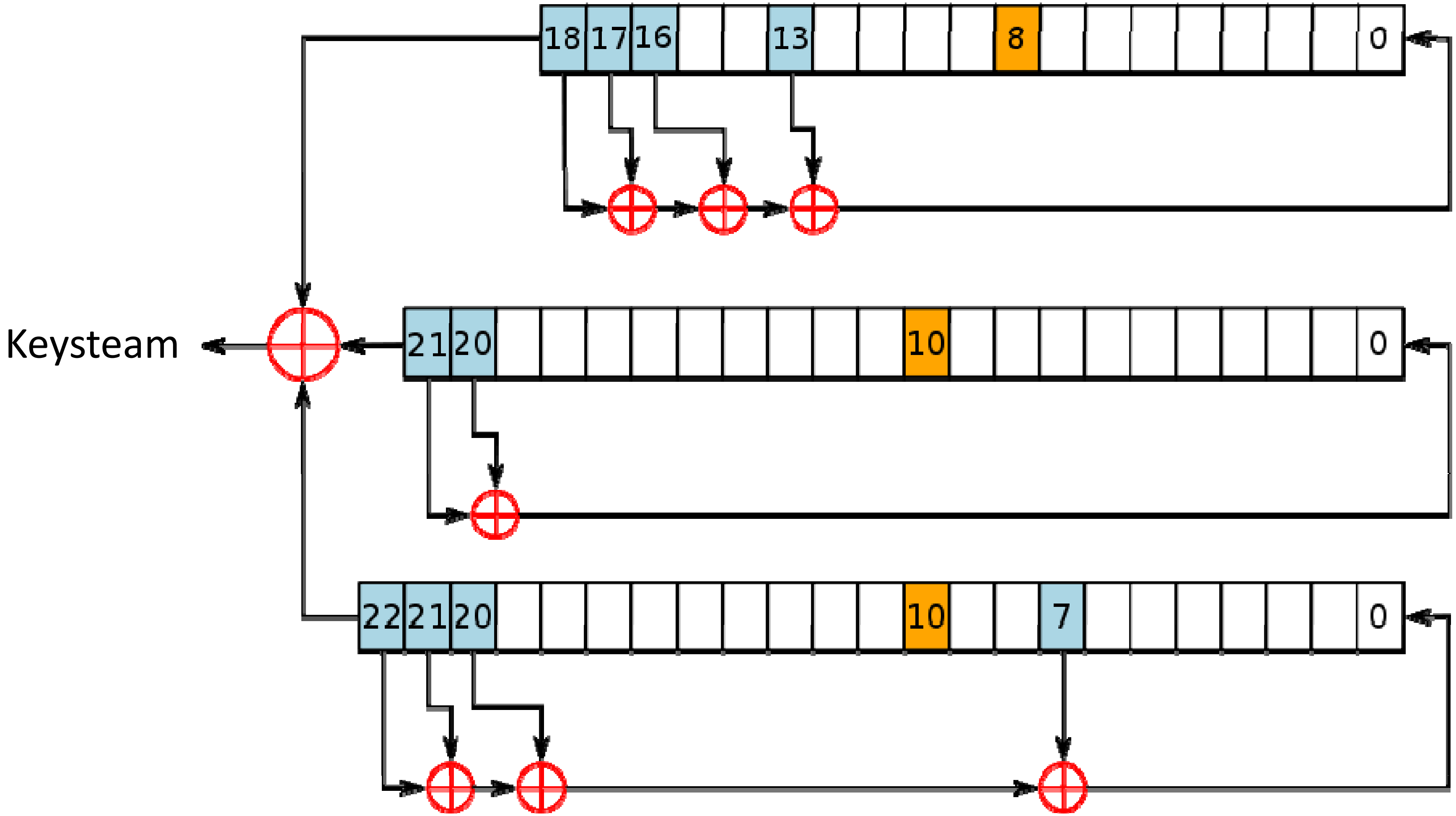
Stream cipher A5/1

- Three irregularly clocked LFSRs
- At each step, each LFSR provides one clocking bit
  - Compute the majority of those three bits
  - If the clocking bit of an LFSR is the majority, clock that LFSR. (each step, at least two LFSRs get clocked).



LFSR number	Length in bits	Characteristic polynomial	Clocking bit	Tapped bits
1	19	$x^{18} + x^{17} + x^{16} + x^{13} + 1$	8	13, 16, 17, 18
2	22	$x^{21} + x^{20} + 1$	10	20, 21
3	23	$x^{22} + x^{21} + x^{20} + x^{7} + 1$	10	7, 20, 21, 22

Stream cipher A5/1



Stream cipher A5/1

- Initialization
  - Load the 64-bit key (10 zero bits) into the LFSRs
    - 64 steps
    - At the *i*-th step, XOR the *i*-th bit of the key to the least significant bits of those three LFSRs
  - Load the 22-bit IV into the LFSRs
    - 22 steps
    - Similar to the key loading
  - Run the cipher 100 steps to mix key and IV
    - no output for these 100 steps
- Keystream generation
  - Update the state (clock those LFSRs according to the majority bit)
  - At each step, one keystream bit is generated
  - Only 228 bits are generated from each IV
    - First 114 bits for decrypting the received packet
    - Last 114 bits for encrypting the outgoing packet
    - About 217 packets/second

Stream cipher A5/1

- A5/1 is dedicated to mobile communication
  - The most widely used hardware stream cipher
- Not suitable for other applications
  - 64-bit key size (small)
  - 22-bit IV size (small)
  - 64-bit state size (too small)
  - Small keystream period
    - since the small state is updated in an non-invertible way

RC4: Another widely used stream cipher

- RC4
  - The most widely used software stream cipher
    - Such as in SSL
  - Designed by Ron Rivest (1987)
  - Extremely simple
  - Generally strong
    - But weak when a key is used with different IVs

Stream Cipher RC4

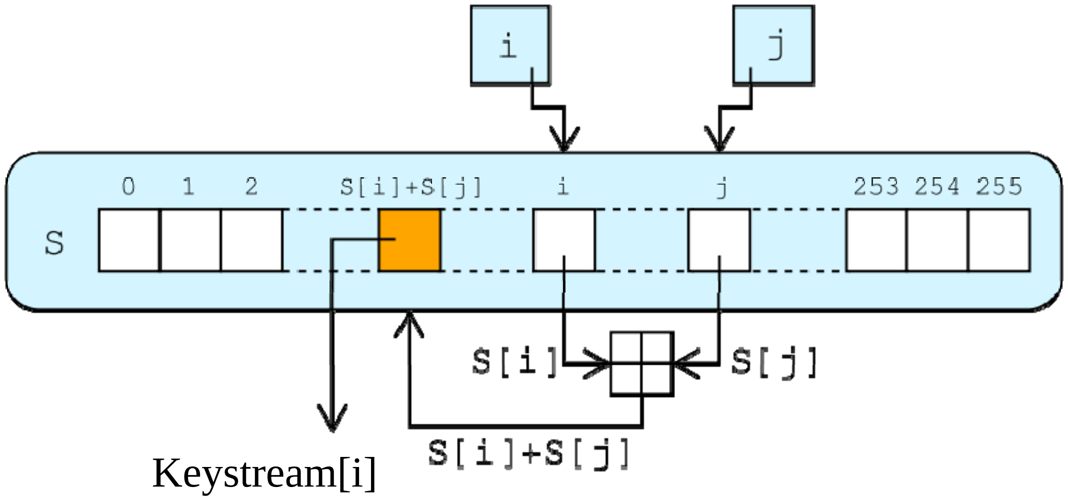
- The state
  - A secret table ***S*** with 256 elements + 2 indices
  - Each element is one-byte

RC4: Keystream generation

```
i = 0
j = 0
while Generating Keystream:
      i := (i + 1) mod 256
      j := (j + S[i]) mod 256
      swap values of S[i] and S[j]
      Keystream[i] := S[(S[i] + S[j]) mod 256]
endwhile
```

Update the state

Generate keystream



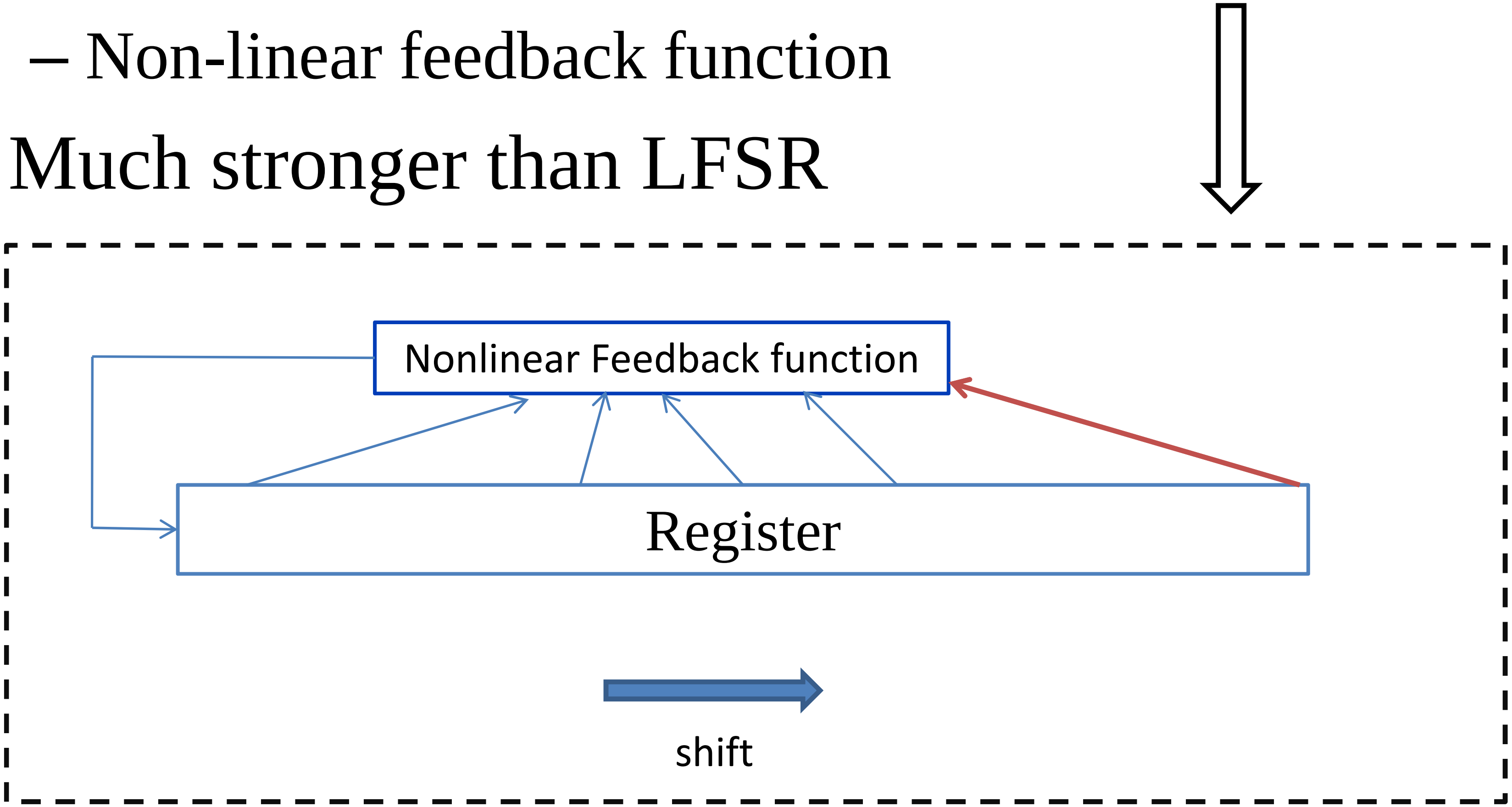
RC4: Initialization

```
for i from 0 to 255
    S[i] = i
endfor
j = 0
for i from 0 to 255
    j = (j + S[i] + key[i mod keylength]) mod 256
    swap values of S[i] and S[j]
endfor
```

- In the above initialization, there is no IV
- If an IV is used, it is considered as part of the key (with increased key length)
  - The above initialization is insufficient to mix well the key & IV

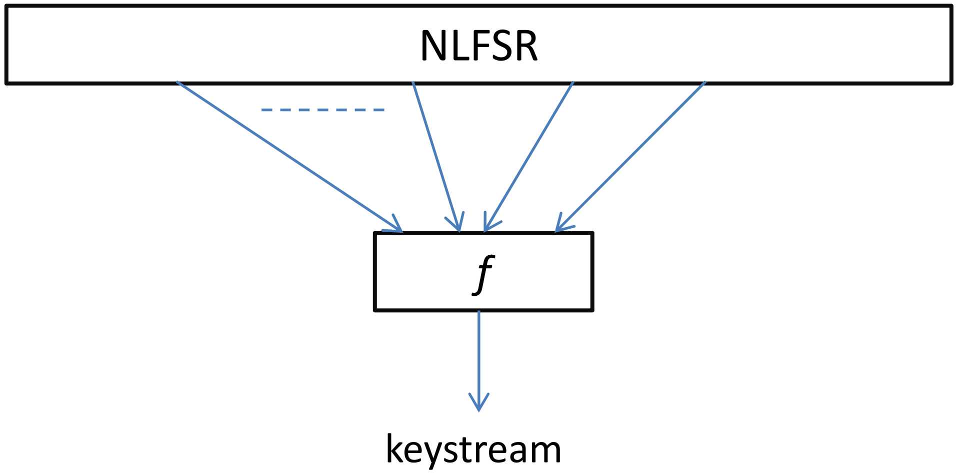
NLFSR Based Stream Cipher

- Non-linear feedback shift register (NLFSR)
  - Non-linear feedback function
- Much stronger than LFSR



NLFSR Based Stream Cipher

- Dominating the stream cipher design today
- The filtering type is better



Recent Developments on Stream Cipher

- eSTREAM project (2004 -- 2008)
  - – The stream cipher project of the European Network of Excellence for Cryptology (ECRYPT)
  - – To identify secure & efficient stream ciphers
  - – Around 35 submissions (around 50 ciphers)
    - 7 were selected in 2008
    - 4 for software: HC-128, Rabbit, Salsa20/12, SOSEMANUK
    - 3 for hardware: Grain, MICKEY, Trivium
    - Only SOSEMANUK is based on LFSR

Summary

- One-time pad → stream cipher
- Two types of stream ciphers
  - Synchronous stream cipher (more popular)
  - Asynchronous stream cipher
- Three main constructions
  - Block cipher based stream cipher
    - CFB, OFB, CTR
  - LFSR based stream cipher
  - NLFSR based stream cipher (now dominative)
- Two widely used stream ciphers
  - A5/1
  - RC4