TCVN **TIÊU CHUẨN QUỐC GIA**

TCVN 11367-4:2016
ISO/IEC 18033-4:2011

Xuất bản lần 1

CÔNG NGHỆ THÔNG TIN - CÁC KỸ THUẬT AN TOÀN - THUẬT TOÁN MẬT MÃ - PHẦN 4: MÃ DÒNG

Information technology -- Security techniques -- Encryption algorithms -- Part 4: Stream ciphers

HÀ NỘI - 2016

Mục Lục

Lời nói đầu

TCVN 11367-4:2016 hoàn toàn tương đương với ISO/IEC 18033-4:2011.

TCVN 11367-4:2016 do Cục Quản lý mật mã dân sự và Kiểm định sản phẩm mật mã biên soạn, Ban Cơ yếu Chính phủ đề nghị, Tổng cục Tiêu chuẩn Đo lường Chất lượng thẩm định, Bộ Khoa học và Công nghệ công bố.

Bộ tiêu chuẩn TCVN 11367 *Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã* gồm 04 phần:

- TCVN 11367-1:2016 (ISO/IEC 18033-1:2015) Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã – Phần 1: Tổng quan.

- TCVN 11367-2:2016 (ISO/IEC 18033-2:2006) Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã – Phần 2: Mật mã phi đối xứng.

- TCVN 11367-3:2016 (ISO/IEC 18033-3:2010) Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã – Phần 3: Mã khối.

- TCVN 11367-4:2016 (ISO/IEC 18033-4:2011) Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mật mã – Phần 4: Mã dòng.

Giới thiệu

Tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) bao gồm các thuật toán mã dòng. Mã dòng là cơ chế mã hóa sử dụng khóa dòng để mã hóa bản rõ theo cách từng bit hoặc từng khối. Có hai loại mã dòng: mã dòng đồng bộ, trong đó khóa dòng chỉ được tạo ra từ khóa bí mật (và véc tơ khởi tạo) và mã dòng tự đồng bộ, trong đó khóa dòng được tạo ra từ khóa bí mật (và véc tơ khởi tạo). Tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) mô tả cả hai bộ tạo số giá ngẫu nhiên để sinh ra khóa dòng và hàm đầu ra kết hợp khóa dòng với bản rõ

Tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) bao gồm hai hàm đầu ra:

- Hàm đầu ra cộng nhị phân; và
- Hàm đầu ra MULTI-S01.

Tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) bao gồm năm bộ tạo khóa dòng chuyên dụng:

- Bộ tạo khóa dòng MUGI;
- Bộ tạo khóa dòng SNOW 2.0;
- Bộ tạo khóa dòng Rabbit;
- Bộ tạo khóa dòng Decim^{v2}; và
- Bộ tạo khóa dòng Kcipher-2(K2).

TIÊU CHUẨN QUỐC GIA **TCVN 11367-4:2016**

Công nghệ thông tin - Các kỹ thuật an toàn - Thuật toán mật mã - Phần 4: Mã dòng

Information technology - Security techniques - Encryption algorithms - Part 4: Stream ciphers

1 Phạm vi áp dụng

Tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) quy định

a) Hàm đầu ra để kết hợp một khóa dòng với bản rõ,

b) Bộ tạo khóa dòng để sinh khóa dòng, và

c) Định danh đối tượng được gán cho bộ tạo khóa dòng chuyên dụng phù hợp với tiêu chuẩn ISO/IEC 9834.

CHÚ THÍCH 1 Danh sách định danh đối tượng được gán đưa ra trong Phụ lục A.

CHÚ THÍCH 2 Bất kỳ thay đổi nào của đặc tả các thuật toán này làm thay đổi hành vi chức năng sẽ dẫn đến thay đổi đối tượng định danh gán cho thuật toán có liên quan.

2 Tài liệu viện dẫn

Các tài liệu viện đẫn sau rất cần thiết cho việc áp dụng tiêu chuẩn này. Đối với các tài liệu viện dẫn ghi năm công bố thì áp dụng phiên bản được nêu. Đối với các tài liệu viện dẫn không ghi năm công bố thì áp dụng phiên bản mới nhất, bao gồm cả các sửa đổi, bổ sung (nếu có).

ISO/IEC 18033-1, *Information technology — Security techniques — Encryption algorithms — Part 1: General*

3 Thuật ngữ và định nghĩa

Trong tiêu chuẩn này áp dụng các thuật ngữ và định nghĩa dưới đây:

3.1

Big-endian (big-endian)

Phương pháp lưu trữ các số nhiều byte với byte trọng số cao nhất tại các địa chỉ bộ nhớ thấp nhất

[ISO/IEC 10118-1:2000]

3.2

Bản mã (ciphertext)

Dữ liệu đã được biến đổi để giấu thông tin chứa trong đó

[ISO/IEC 10116:2006]

3.3

Tính bí mật (confidentiality)

Thuộc tính mà thông tin không ở dạng sẵn sàng hoặc bị tiết lộ cho cá nhân, thực thể hoặc quy trình không được phép

3.4

Tính toàn vẹn dữ liệu (data integrity)

Thuộc tính mà dữ liệu không bị thay đổi hoặc phá hủy một cách trái phép

[ISO/IEC 9797-1:2011]

3.5

Giải mã (decryption)

Phép toán ngược với phép mã hóa tương ứng

[ISO/IEC 10116-1:2006]

3.6

Mã hóa (encryption)

Phép biến đổi (khả nghịch) dữ liệu bởi thuật toán mật mã để tạo ra bản mã, tức là giấu nội dung thông tin của dữ liệu.

[ISO/IEC 9797-1:2011]

3.7

Giá trị khởi tạo (initialization value)

Giá trị sử dụng trong việc xác định điểm khởi đầu của quá trình mã hóa

3.8

Khóa (key)

Dãy các kí hiệu điều khiển sự vận hành của các phép biến đổi mật mã (ví dụ, phép mã hóa, giải mã, tính toán hàm kiểm tra mật mã, tạo chữ ký số hoặc xác thực chữ ký số)

[ISO/IEC 11770-1:2010]

3.9

Hàm khóa dòng (keystream function)

Hàm nhận đầu vào là trạng thái hiện tại của bộ tạo khóa dòng và (tùy chọn) một phần của bản mã được tạo ra trước đó và cho đầu ra là phần tiếp theo của khóa dòng.

3.10

Bộ tạo khóa dòng (keystream generator)

Quá trình dựa trên trạng thái (nghĩa là máy trạng thái hữu hạn) nhận đầu vào là một khóa, một véc tơ khởi tạo và bản mã nếu cần thiết, và đưa đầu ra là một khóa dòng (nghĩa là dãy tuần tự các bit hoặc các khối bit) có độ dài tùy ý.

3.11

Mã khối n bit (n-bit block cipher)

Mã khối với tính chất là các khối của bản rõ và bản mã đều có độ dài n bit

[ISO/IEC 10116:2006]

3.12

Hàm chuyển trạng thái tiếp theo (next-state function)

Hàm nhận đầu vào là trạng thái hiện tại của bộ tạo khóa dòng và (tùy chọn) một phần của bản mã được tạo trước đó, và đưa đầu ra là một trạng thái mới của bộ tạo khóa dòng.

3.13

Hàm đầu ra (output function)

Hàm kết hợp khóa dòng mã bản rõ để tạo bản mã

CHÚ THÍCH Hàm này thực hiện phép XOR từng bit.

3.14

Đệm (padding)

Bit mở rộng đính kèm cho xâu dữ liệu

[ISO/IEC 10118-1:2000]

3.15

Bản rõ (plaintext)

Thông tin chưa được mã hóa

[ISO/IEC 9797-1:2011]

3.16

Khóa bí mật (secret key)

Khóa sử dụng cho kỹ thuật mật mã đối xứng và được dùng bởi một tập thực thể xác định

[ISO/IEC 11770-3:2008]

3.17

Trạng thái (state)

Trạng thái bên trong hiện tại của bộ tạo khóa dòng

4 Ký hiệu và chữ viết tắt

4.1 Ký hiệu

$0x$	Tiền tố cho các giá trị thập lục phân.
$0^{(n)}$	Biến n-bit mà 0 được gán cho mỗi bit.
AND	Phép toán logic AND từng bit.
$Am^{(i)}[Y]$	Bit thứ Y của thanh ghi $Am^{(i)}$ trong KCipher-2 (K2).
a_i	Các biến trong trạng thái trong của bộ tạo khóa dòng.
b_i	Các biến trong trạng thái trong của bộ tạo khóa dòng.
CFB	Chế độ phản hồi mã khối
CTR	Chế độ bộ đếm của mã khối.
C_i	Khối bản mã.
D_i	Hằng số 64-bit được sử dụng cho MUGI.
e_K	Hàm mã hóa mã khối đối xứng sử dụng khóa bí mật K.
F	Hàm con được sử dụng cho MUGI.
FSM	Hàm con được sử dụng cho SNOW 2.0.
$GF(2^n)$	Trường hữu hạn gồm có 2^n phần tử.
$GF(2^n)[x]$	Vành đa thức trên trường hữu hạn $GF(2^n)$
$Init$	Hàm tạo trạng thái khởi tạo trong của bộ tạo khóa dòng.
IV	Véc tơ khởi tạo.
IK	Khóa trong được sử dụng cho KCipher-2 (K2).

K	Khóa
M	Hàm con được sử dụng cho MUGI.
$Next$	Hàm chuyển trạng thái tiếp theo của bộ tạo khóa dòng.
NLF	Hàm phi tuyến sử dụng cho KCipher-2 (K2).
n	Độ dài khối.
OFB	Chế độ phản hồi đầu ra mã khối.
OR	Phép toán logic OR từng bit.
Out	Hàm đầu ra kết hợp khóa dòng và bản rõ để tạo bản mã.
P	Bản rõ.
P_t	Khối bản rõ.
R	Đầu vào bổ sung cho biến Out.
S_R	Hàm con được sử dụng cho MUGI.
$Strm$	Hàm khóa dòng của bộ tạo khóa dòng.
SUB	Bảng tra cứu sử dụng cho MUGI và SNOW 2.0.
Sub_{K2}	Hàm con được sử dụng cho KCipher-2 (K_2).
S_i	Trạng thái trong của bộ tạo khóa dòng.
T	Hàm con được sử dụng cho SNOW 2.0.
Z	Khóa dòng.
Z_t	Khối khóa dòng.
α_{MUL}	Bảng tra cứu được sử dụng cho SNOW 2.0.
α_{MUL0}	Bảng tra cứu với chỉ số 0 được sử dụng cho KCipher-2 (K2).
α_{MUL1}	Bảng tra cứu với chỉ số 1 được sử dụng cho KCipher-2 (K2).
α_{MUL2}	Bảng tra cứu với chỉ số 2 được sử dụng cho KCipher-2 (K2).
α_{MUL3}	Bảng tra cứu với chỉ số 3 được sử dụng cho KCipher-2 (K2).
α_{inv_MUL}	Bảng tra cứu nghịch đảo được sử dụng cho SNOW 2.0.
ρ_t	Hàm con được sử dụng cho MUGI.

λ_I	Hàm con được sử dụng cho MUGI.
$\lceil x \rceil$	Các số nguyên nhỏ nhất lớn hơn hoặc bằng số thực.
$\neg_x$	Phép toán bù từng bit.
$\cdot$	Phép nhân đa thức.
$\|\|$	Phép ghép các xâu bit.
$+_m$	Phép cộng số nguyên modulo 2^m.
$\oplus$	Phép toán XOR (OR loại trừ) từng bit.
$\otimes$	Phép nhân các phần tử trong trường hữu hạn $GF(2^n)$.
$\boxplus$	Phép cộng modulo
$\lll_n t$	Phép dịch trái t-bit trong thanh ghi n-bit.
$\ggg_n t$	Phép dịch phải t-bit trong thanh ghi n-bit.
$\lll\!\!\lll_n t$	Phép dịch vòng sang trái t- bit trong thanh ghi n-bit.
$\ggg\!\!\ggg_n t$	Phép dịch vòng sang phải t- bit trong thanh ghi n-bit.

4.2 Các hàm

4.2.1 Hàm cắt trái các bit

Phép toán lựa chọn j bit bên trái của mảng $A = (a_0, a_1, \dots, a_{m-1})$ để bạo ra một mảng j-bit và được viết

$$(j{\sim}A) = (a_0, a_1, \dots, a_{j-1})$$

Phép toán xác định với $1 \leq j \leq m$.

Xem trong ISO/IEC 10116:2006.

4.2.2 Phép toán dịch

Phép toán dịch được xác định như sau: Cho một biến n-bit X và biến k-bit V trong đó $1 \leq k \leq n$, tác dụng của hàm dịch để tạo ra biến n-bit

$$Shift_k(X|V) = (x_k, x_{k+1}, \dots, x_{n-1}, v_0, v_1, \dots, v_k) \quad (k < n)$$

$$Shift_k(X|V) = (v_0, v_1, \dots, v_{k-1}) \quad (k = n)$$

Kết quả sự dịch chuyển các bit của mảng X sang trái k vị trí, bỏ đi $x_0, x_1, \dots, x_{k-1}$ và đưa vào bên phải nhất mảng V k vị trí của X. Khi $k = n$ thì hoàn toàn thay thế X bởi V.

Xem ISO/IEC 10116:2006.

4.2.3 Biến *I(k)*

Biến $l(k)$ là một biến k-bit mà mỗi bit được gán giá trị 1.

5 Khung cho mã dòng

Điều này bao gồm mô tả mức cao nhất của khung cho mã dòng được quy định trong phần này của bộ TCVN 11367 (ISO/IEC 18033). Mô tả chi tiết của mô hình tổng quát cho mã dòng được quy định tại điều 6. Mã dòng được quy định trong tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) được xác định bởi các đặc tả của các quy trình sau:

a) Bộ tạo khóa dòng, có thể là:

- Bộ tạo khóa dòng đồng bộ, hoặc

- Bộ tạo khóa dòng tự đồng bộ.

CHÚ THÍCH 1 Các Chế độ hoạt động của mã khối là phương pháp mà mã khối có thể được sử dụng để xây dựng một bộ tạo khóa dòng. Các chế độ này được chuẩn hóa trong tiêu chuẩn ISO/IEC 10116 và ý nghĩa của các hàm được sử dụng trong đặc tả được xác định trong 6.2.1 và 6.2.2.

CHÚ THÍCH 2 Mã khối được định nghĩa trong tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033).

b) Hàm đầu ra, có thể là:

- Hàm đầu ra cộng nhị phân, hoặc

- Hàm đầu ra MULTI-S01.

6 Mô hình tổng quát của mã dòng

6.1 Các bộ tạo khóa dòng

6.1.1 Bộ tạo khóa dòng đồng bộ

Bộ tạo khóa dòng đồng bộ là máy trạng thái hữu hạn được định nghĩa như sau:

a) Một hàm khởi tạo, $Init$, nhận đầu vào khóa K và véc tơ khởi tạo IV và cho đầu ra một trạng thái khởi tạo S_0 cho bộ tạo khóa dòng. Véc tơ khởi tạo cần được lựa chọn để không bao giờ có hai thông báo được mã hóa sử dụng cùng khóa và cùng véc tơ khởi tạo IV.

b) Hàm chuyển trạng thái tiếp theo, $Next$, nhận đầu vào là trạng thái hiện tại của bộ tạo khóa dòng S_i, và đưa ra trạng thái tiếp theo của bộ tạo khóa dòng S_{i+1}.

c) Hàm khóa dòng, $Strm$, nhận đầu vào là trạng thái của bộ tạo khóa dòng S_i, và đưa ra khối khóa dòng Z_i.

Khi bộ tạo khóa dòng đồng bộ lần đầu khởi tạo, nó sẽ nhập vào một trạng thái khởi tạo S_0 được xác định bởi:

$$S_0 = Init(IV, K)$$

Theo nhu cầu bộ tạo khóa dòng đồng bộ sẽ, với $i = 0, 1, \ldots$

a) Đưa ra khối khóa dòng $Z_i = Strm(S_i, K)$.

b) Cập nhật trạng thái máy $S_{i+1} = Next(S_i, K)$.

Vì vậy, để xác định bộ tạo khóa dòng đồng bộ chỉ cần xác định hàm $Init$, $Next$ và $Strm$, bao gồm cả độ dài và bảng chữ cái của khóa, véc tơ khởi tạo, trạng thái và khối đầu ra.

6.1.2 Bộ tạo khóa dòng tự đồng bộ

Việc tạo ra khóa dòng cho mã dòng tự đồng bộ chỉ phụ thuộc vào bản mã trước, khóa và véc tơ khởi tạo. Mô hình tổng quát cho bộ tạo khóa dòng cho mã dòng tự đồng bộ được xác định:

a) Hàm khởi tạo, $Init$, nhận đầu vào là khóa K và véc tơ khởi tạo IV và đưa ra đầu vào trong cho bộ tạo khóa dòng S và khối bản mã giả r $C_{-1}, C_{-2}, \ldots, C_{-r}$.

b) Hàm khóa dòng, $Strm$, nhận đầu vào S và khối bản mã r $C_{-1}, C_{-2}, \ldots, C_{-r}$ và đưa ra khối khóa dòng Z_i.

Để xác định bộ tạo khóa dòng tự đồng bộ chỉ cần thiết xác định số lượng các khối phản hồi r và hàm $Init$ và hàm $Strm$.

CHÚ THÍCH Mã dòng tự đồng bộ khác với mã dòng đồng bộ ở chỗ khóa dòng chỉ phụ thuộc vào bản mã trước, véc tơ khởi tạo và khóa, nghĩa là bộ tạo khóa dòng hoạt động trong kiểu không trạng thái. Kết quả là, sự giải mã cho mật mã như vậy có thể khôi phục từ sự mất mát của đồng bộ hóa sau khi nhận được đầy đủ các khối bản mã. Điều này cũng có nghĩa là các phương pháp tạo khóa dòng phụ thuộc vào hàm đầu ra được lựa chọn Out, mà điển hình là phép toán XOR từng bit.

6.2 Các hàm đầu ra

6.2.1 Mô hình tổng quát của hàm đầu ra

Điều 6.2 chỉ rõ hai hàm đầu ra mã dòng, nghĩa là các kỹ thuật được sử dụng trong mã dòng để kết hợp khóa dòng với bản rõ để nhận được bản mã.

Hàm đầu ra cho mã dòng đồng bộ hoặc tự đồng bộ là hàm Out kết hợp khối bản rõ P_i, khối khóa dòng Z_i và một số đầu vào khác R nếu cần thiết để đưa ra khối bản mã $C_i (i \geq 0)$. Mô hình tổng quát của hàm đầu ra mã dòng được xác định:

Mã hóa khối bản rõ P_i bằng khối khóa dòng Z_i xác định như sau:

$$C_i = Out(P_i, Z_i, R)$$

Và giải mã khối bản mã C_i bằng khối khóa dòng Z_i xác định như sau:

$$P_i = Out^{-1}(P_i, Z_i, R).$$

Hàm đầu ra phải đáp ứng cho bất kỳ khối khóa dòng Z_i, khối bản rõ P_i và đầu vào khác R,

$$P_i = Out^{-1}(Out(P_i, Z_i, R), Z_i, R).$$

6.2.2 Hàm đầu ra cộng nhị phân

Mã dòng cộng nhị phân là mã dòng trong đó khối khóa dòng, khối bản rõ và khối bản mã là các xâu các số nhị phân và phép toán để kết hợp bản rõ với khóa dòng là phép toán XOR từng bit. Phép toán Out có hai đầu vào và không sử dụng bất kỳ thông tin bổ sung R để tính toán. Cho n là độ dài bit của P_i. Hàm này xác định như sau:

$$Out(P_i, Z_i, R) = P_i \oplus Z_i$$

Phép toán Out^{-1} xác định như sau:

$Out^{-1}(C_i, Z_i, R) = C_i \oplus Z_i$

CHÚ THÍCH Hệ mã dòng cộng nhị phân không cung cấp bất kỳ tính bảo vệ tính toàn vẹn cho dữ liệu được mã hóa. Nếu có yêu cầu tính toàn vẹn cho dữ liệu thì hoặc là sử dụng hàm đầu ra MULTI-S01 hoặc có cơ chế toàn vẹn riêng biệt, chẳng hạn MAC, nghĩa là Mã xác thực thông báo (cơ chế này được quy định trong tiêu chuẩn ISO/IEC 9797).

6.2.3 Hàm đầu ra MULTI-S01

a) Mô hình tổng quát của MUTIL-S01

MULTI-S01 là hàm đầu ra cho mã dòng đồng bộ hỗ trợ cả tính toàn vẹn và tính bí mật của dữ liệu. Phép toán mã hóa MULTI-S01 phù hợp để sử dụng trong môi trường trực tuyến. Tuy nhiên, phép toán giải mã của MULTI-S01 có thể chỉ thực hiện trong tình huống ngoại tuyến, như kiểm tra tính toàn vẹn chỉ được thực hiện sau khi nhận được tất cả các khối bản mã. MULTI-S01 có một tham số an toàn n. Việc tính toán đầu ra phụ thuộc vào sự lựa chọn trường $GF(2^n)$, tức là phụ thuộc vào lựa chọn đa thức bất khả quy bậc n trên trường $GF(2^n)$. Hàm MULTI-S01 chỉ chấp nhận thông báo có chiều dài là bội số của n. Để mã hóa thông điệp có chiều dài không phải là bội của n, yêu cầu sử dụng thêm cơ chế đệm $Pad(M)$.

CHÚ THÍCH Việc dư thừa R được tạo ra theo cách mà người gửi và người nhận chia sẻ nó. R có thể là một giá trị công khai cố định như 0x00...0.

b) Hàm mã hóa $Out(P, R, Z)$

Đầu vào: bản rõ P $n.u$- bit , khóa dòng $Z = (Z_0, Z_1, \dots)$. Trong đó Z_i là các khối n-bit, dưa thừa R n-bit.

Đầu ra: Bản mã C.

1) Lấy t là giá trị thấp nhất của i $(i \geq 0)$ sao cho $Z_i \neq 0^{(n)}$.

2) Lấy $(P_0, P_1, \dots P_{u-1}) = P$, trong đó P_i là khối n-bit

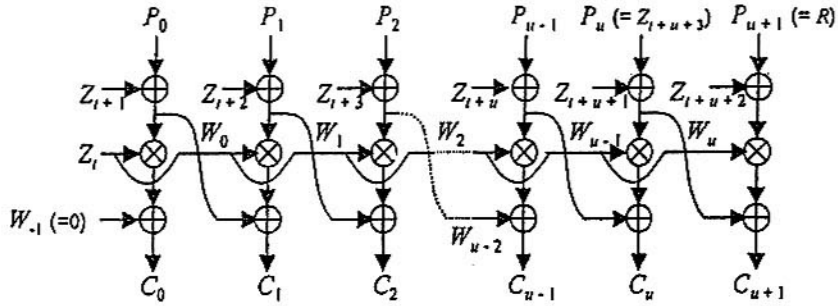
3) Đặt $P_u = Z_{t+u+3}$.

4) Đặt $P_{u+1} = R$.

5) Đối với mỗi P_i, thực hiện các phép tính sau (*với i = 0,1,...,u+1*):

- Lấy $W_i = P_i \oplus Z_{t+i+1}$.
- Lấy $X_i = Z_t \otimes W_i$ (trong trường $GF(2^n)$).
- Lấy $C_i = X_i \oplus W_{i-1}$ trong đó W_{i-1} là giá trị W của khối *i-1* trước đó và $W_{-1} = 0^{(n)}$.
- Đặt $C = C_0 \| C_1 \| \cdots \| C_{u+1}$.
- Đưa ra C.

Hình 1 mô tả sơ đồ khối của hàm Out



Hình 1 – Hàm *Out* của chế độ MUTIL-S01

CHÚ THÍCH 2 Đa thức bất khả quy được sử dụng để xác định phép nhân trong trường phụ thuộc vào *n*. Ví dụ, trong trường hợp $n = 64$ và 128, có thể sử dụng đa thức bất khả quy $x^{64} + x^4 + x^3 + x + 1$ và $x^{128} + x^7 + x^2 + x + 1$.

c) Hàm giải mã $Out^{-1}(P, Z, R)$

Đầu vào: bản mã C độ dài $n.v$-bit, khóa dòng Z, dư thừa R độ dài n-bit.

Đầu ra: bản rõ P hoặc "*từ chối*".

1) Lấy t là giá trị thấp nhất của i ($i \geq 0$) sao cho $Z_i \neq 0^{(n)}$.

2) Lấy $(C_0, C_1, \ldots C_{v-1}) = C$, trong đó C_i là khối n-bit

3) Đối với mỗi C_i, thực hiện các phép tính sau (với *i=0,1,...,v-1*):

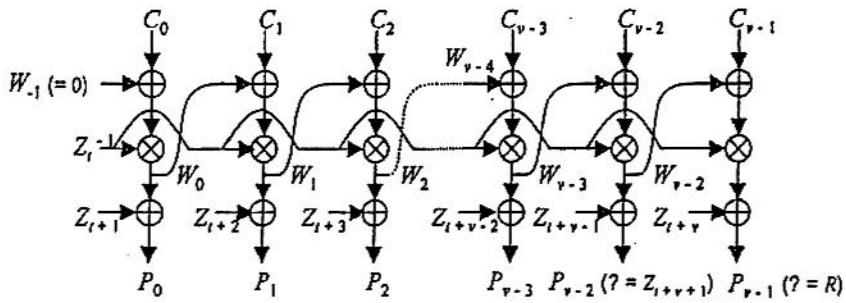
- Lấy $X_i = C_i \oplus W_{i-1}$, trong đó $W_{-1} = 0^{(n)}$.

- Lấy $W_i = Z_t^{-1} \otimes X_i$ (trong trường $GF(2^n)$).

- Lấy $P_i = W_i \oplus Z_{t+i+1}$.

4) Nếu $P_{v-2} = Z_{t+v+1}$ và $P_{v-1} = R$, đưa ra $P = P_0 \| P_1 \| \cdots \| P_{v-3}$ là bản rõ. Nếu không, đưa ra biểu tượng đặc biệt nghĩa là "*từ chối*" mà không có bất kỳ văn bản nào.

Hình 2 mô tả sơ đồ khối của hàm Out^{-1}



Hình 2 – Hàm Out^{-1} của chế độ MULTI-S01

d) Cơ chế đệm *Pad(M)*

Chỉ khi độ dài của thông báo đầu vào không phải là bội số của n, cơ chế đệm $Pad(M)$ sau đây được thực thi:

Đầu vào: Xâu M độ dài $(nv+c)$-bit, trong đó v là số nguyên không âm và $0 \leq c < n$.

Đầu ra: bản rõ P được đệm

1) Đệm xâu bit *"1"* vào cuối thông báo.

2) Đệm xâu $0^{(n-c-1)}$ độ dài $(n-c-1)$-bit vào xâu được tạo bởi bước a).

3) Đưa ra toàn bộ xâu có độ dài $(nv+n)$-bit.

CHÚ THÍCH 3 Nếu độ dài của thông báo là bội số của n, trong môi trường độ dài không phải là nhất định như vậy, cơ chế đệm này được khuyến khích.

CHÚ THÍCH 4 Để bỏ đệm của thông báo, loại bỏ liên tiếp bit *0* ở phần cuối của dữ liệu và loại bỏ các bit *"1"*.

7 Xây dựng bộ tạo khóa dòng từ mã khối

7.1 Các chế độ mã khối cho bộ tạo khóa dòng đồng bộ

7.1.1 Chế độ OFB (Đầu ra phản hồi) và CTR (Bộ đếm)

Điều 7.1 quy định hai chế độ mã khối n-bit cho bộ tạo khóa dòng đồng bộ. Đó là, chế độ OFB (Đầu ra phản hồi) và chế độ CTR (Bộ đếm) của mã khối e_K n-bit.

7.1.2 Chế độ OFB

Chế độ OFB được xác định bởi một tham số r, $1 \leq r \leq n$, là kích thước của khối bản rõ và bản mã.

Véc tơ khởi tạo IV là xâu n-bit. IV sẽ được tạo ra khác nhau cho hai quá trình mã hóa với cùng một khóa K. Hàm $Init$, $Next$ và $Strm$ được quy định như sau:

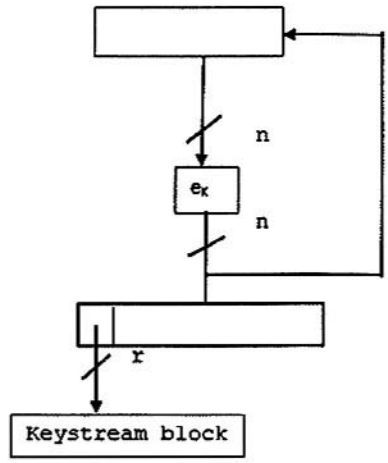
- $Init(IV,K)=IV$.

- $Next(S_i, K) = e_K(S_i)$.

- $Strm(S_i) = (r \sim S_i)$.

CHÚ THÍCH $Init(IV,K) = IV$ tương đương với $S_0 = IV$.

Trong trường hợp của chế độ OFB, hàm đầu ra cộng nhị phân được xác định trong 6.2.2 được sử dụng. Hình 3 mô tả sơ đồ khối của bộ tạo khóa dòng dựa trên chế độ CFB.



Hình 3 – Tạo khóa dòng dựa trên chế độ OFB

7.1.3 Chế độ CTR

Chế độ CTR được xác định bằng một tham số r, $1 \leq r \leq n$, là kích thước của khối bản rõ và bản mã.

Véc tơ khởi tạo IV là xâu n-bit cần được đảm bảo rằng $S_i \neq S_j'$ cho hai khóa dòng $S_0, S_1, S_2, \ldots$ và $S_0', S_1', S_2', \ldots$ được tạo ra với cùng một khóa K. Hàm $Init$, $Next$ và $Strm$ được xác định như sau:

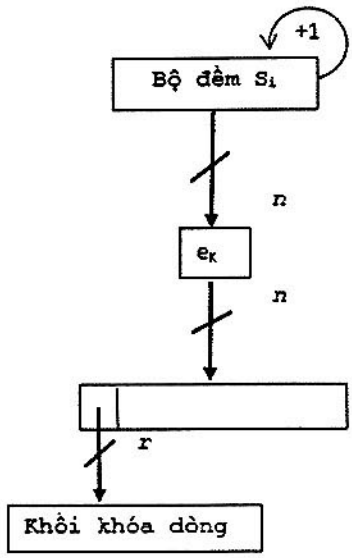
- $Init(IV, K) = IV$.

- $Next(S_i, K) = S_i + 1 \bmod 2^n$.

- $Strm(S_i, K) = (r \sim e_K(S_i))$.

CHÚ THÍCH $Init(IV, K) = IV$ tương đương với $S_0 = IV$.

Trong trường hợp của chế độ CTR, hàm đầu ra cộng nhị phân được xác định trong 6.2.2 được sử dụng. Hình 4 mô tả sơ đồ khối của bộ tạo khóa dòng dựa trên chế độ CFB.



Hình 4 – Tạo khóa dòng dựa trên chế độ CTR

7.2 Chế độ mã khối cho bộ tạo khóa dòng tự đồng bộ

7.2.1 Giới thiệu chế độ CFB

Chế độ CFB của mã khối n-bit là mã dòng tự đồng bộ.

7.2.2 Chế độ CFB

Chế độ CFB (Phản hồi mã) được xác định với 3 tham số, tức là kích thước j của bộ đệm phản hồi S_i, trong đó $n \leq j \leq 1024n$, kích thước biến phản hồi b, trong đó $1 \leq b \leq n$ và khối đầu ra có kích thước r, trong đó $1 \leq r \leq b$.

CHÚ THÍCH 1 Giá trị b-r cần nhỏ hơn b

Véc tơ khởi tạo IV cần được tạo ngẫu nhiên xâu j-bit và cũng cần được tạo ra khác nhau cho hai mã hóa với cùng một khóa K. Hàm $Init$, $Next$ và $Strm$ được xác định như sau:

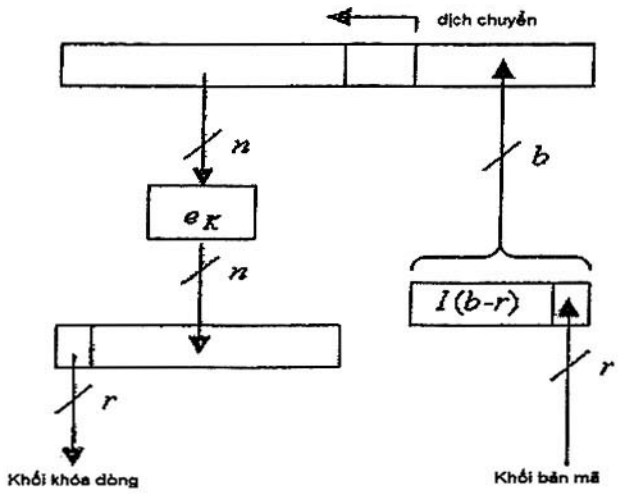
- $Init(IV,K)=IV$.

- $Next(S) = Shift_b(S|Shift_r(I(b)|C_i))$.

- $Strm(S,K) = (r \sim e_K((n \sim S)))$.

CHÚ THÍCH 2 $Init(IV,K) = IV$ tương đương với $S_0 = IV$.

Trong trường hợp của chế độ CFB, hàm đầu ra cộng nhị phân được xác định trong 6.2.2 được sử dụng. Hình 5 mô tả sơ đồ khối của bộ tạo khóa dòng dựa trên chế độ CFP.



Hình 5 – Tạo khóa dòng dựa trên chế độ CFB

8 Bộ tạo khóa dòng chuyên dụng

8.1 Bộ tạo khóa dòng MUGI

8.1.1 Giới thiệu MUGI

MUGI là bộ tạo khóa dòng sử dụng khóa bí mật K 128-bit, véc tơ khởi tạo IV 128-bit và biến trạng thái S_i ($i \geq 0$) bao gồm 19 khối 64-bit (lưu ý rằng mỗi khối được sử dụng thông qua đặc tả của MUGI cho mỗi khối 64-bit) và đưa ra khối khóa dòng Z_i tại mỗi lần lặp của hàm *Strm*.

CHÚ THÍCH Bộ tạo khóa dòng này ban đầu được đề xuất trong [17].

Biến trạng thái được chia nhỏ thành kết hợp của biến 3-khối:

$$a^{(i)} = (a_0^{(i)}, a_1^{(i)}, a_2^{(i)}),$$

Trong đó $a_j^{(i)}$ là khối (với j= 0, 1, 2) và biến 16-khối

$$b^{(i)} = (b_0^{(i)}, b_1^{(i)}, \ldots, b_{15}^{(i)}),$$

Trong đó $b_j^{(i)}$ là khối (với j= 0, 1,..., 15)

Hàm *Init*, được xác định chi tiết trong 8.1.2, nhận đầu vào khóa K độ dài 128-bit và véc tơ khởi tạo IV độ dài 128-bit và tạo giá trị ban đầu của biến trạng thái $S_0 = (a^{(0)}, b^{(0)})$.

Hàm *Next*, được xác định chi tiết trong 8.1.3, nhận đầu vào biến trạng thái 19-khối $S_i = (a^{(i)}, b^{(i)})$ và đầu ra là giá trị tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$.

Hàm *Strm*, được xác định chi tiết trong 8.1.4, nhận đầu vào biến trang thái 19-khối $S_i = (a^{(i)}, b^{(i)})$ và đầu ra là khối khóa dòng Z_i.

Lưu ý rằng hàm *Next* được xác định trong các số hạng của ρ_1 và λ_1 được xác định tương ứng trong 8.1.5 và 8.1.6. Hàm ρ_1 được xác định trong số hạng của hàm F được xác định trong 8.1.7

Có 3 hằng số được sử dụng trong MUGI, D_0 trong hàm khởi tạo *Init*, và D_1, D_2 trong ρ_1. Chúng xác định bởi:

D_0 = 0x6A09E667F3BCC908,

D_1 = 0xBB67AE8584CAA73B,

D_2 = 0x3C6EF372FE94F82B.

8.1.2 Hàm khởi tạo *Init*

Việc khởi tạo MUGI được chia ra thành 8 bước. Các khối nửa trái và nửa phải của K được biểu diễn tương ứng bằng K_0 và K_1. IV_0 và IV_1 được xác định theo cách tương tự. Hàm khởi tạo *Init* như sau:

Đầu vào: Khóa K 128-bit, véc tơ khởi tạo IV độ dài 128-bit.

Đầu ra: Giá trị khởi tạo của biến trạng thái $S_0 = (a^{(0)}, b^{(0)})$.

a) Đặt khóa K vào thành phần của biến trạng thái $a^{(-49)}$ như sau:

- Đặt $(K_0, K_1) = K$, trong đó K_i là 64 bit với i = 0, 1
- Đặt $a_0^{(-49)} = K_0$.
- Đặt $a_1^{(-49)} = K_1$.
- Đặt $a_2^{(-49)} = (K_0 \lll_{64} 7) \oplus = (K_1 \ggg_{64} 7) \oplus D_0$.

D_0 trong biểu thức trên là hằng số (xem 8.1).

b) Với i = -49, -48,..., -34 đặt $a^{(i+1)} = \rho_1(a^{(i)}, 0^{(64)}, 0^{(64)})$. Mô tả của ρ xem 8.1.5.

c) Với i = 0, 1,..., 15 đặt $b_{15-i}^{(-16)} = a_0^{(i-48)}$

d) Thêm véc tơ khởi tạo IV vào trạng thái như sau:

- Đặt $IV_0 || IV_1 = IV$, trong đó IV_i là khối
- Đặt $a_0^{(-32)} = a_0^{(-33)} \oplus IV_0$.
- Đặt $a_0^{(-32)} = a_0^{(-33)} \oplus IV_1$.
- Đặt $a_2^{(-32)} = a_2^{(-33)} \oplus (IV_0 \lll_{64} 7) \oplus (IV_1 \ggg_{64} 7) \oplus D_0$.

e) Với i = -32, -31,..., -17, đặt $a^{(i+1)} = \rho_1(a^{(i)}, 0^{(64)}, 0^{(64)})$

f) Đặt $S_{-16} = (a^{(-16)}, b^{(-16)})$

g) Lặp hàm cập nhật *Next* 16 lần:

Đặt $S_0 = Next^{16}(S_{-16})$

Trong đó $Next^{16}$ đại diện cho 16 lần lặp của hàm chuyển trạng thái theo *Next*.

h) Đưa ra S_0.

8.1.3 Hàm chuyển trạng thái theo *Next*

Hàm chuyển trạng thái theo của MUGI được xác định là sự kết hợp của ρ_1 và λ_1. Hàm chuyển trạng thái theo của MUGI như sau:

Đầu vào: Biến trạng thái $S_i = (a^{(i)}, b^{(i)})$.

Đầu ra: Trạng thái tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$.

- Đặt $a^{(i+1)} = \rho_1(a^i, b_4^{(i)}, b_{10}^{(i)})$. Mô tả chi tiết của hàm ρ_1 được đưa ra trong 8.1.5.

- Đặt $b^{(i+1)} = \lambda_1(b^i, a_0^{(i)})$. Mô tả chi tiết của hàm λ_1 được đưa ra trong 8.1.6.

- Đặt $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$

- Đưa ra S_{i+1}.

8.1.4 Hàm khóa dòng *Strm*

Hàm khóa dòng *Strm* như sau:

Đầu vào: Biến trạng thái S_i.

Đầu ra: Khối khóa dòng Z_i.

- Đặt $Z_i = a_2^{(i)}$

- Đưa ra Z_i.

8.1.5 Hàm ρ_1

Hàm ρ_1 như sau:

Đầu vào: Biến trạng thái $a^{(i)}$, hai tham số độ dài 64-bit w_1, w_2.

Đầu ra: Giá trị tiếp theo của biến trạng thái $a^{(i+1)}$

- Đặt $a_0^{(i+1)} = a_1^{(i)}$.

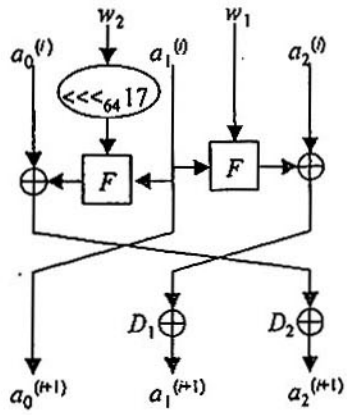
- Đặt $a_1^{(i+1)} = a_2^i \oplus F(a_1^i, w_1) \oplus D_1$.

- Đặt $a_2^{(i+1)} = a_0^{(i)} \oplus F\left(a_1^{(i)}, (w_2 \lll_{64} 17)\right) \oplus D_2$

- Đưa ra $a^{(i+1)}$.

D_1, D_2 là hằng số (xem thêm chi tiết tại 8.1).

Hình 6 mô tả sơ đồ khối của hàm ρ_1. Mô tả chi tiết của hàm F được đưa ra trong 8.1.7.



Hình 6 - Hàn ρ_1 của MUGI

8.1.6 Hàm λ_1

Hàm λ_1 như sau:

Đầu vào: Biến trạng thái $b^{(i)}$, tham số a' độ dài 64-bit.

Đầu ra: Giá trị tiếp theo của biến trạng thái $b^{(i+1)}$.

- Đặt $b_j^{(i+1)} = b_{j-1}^{i}$ với $j \neq 0, 4, 10$.
- Đặt $b_j^{(i+1)} = b_{15}^{(i)} \oplus a'$.
- Đặt $b_4^{(i+1)} = b_3^{(i)} \oplus b_7^{(i)}$
- Đặt $a_{10}^{(i+1)} = b_9^{(i)} \oplus \left(b_{13}^{(i)} \lll_{64} 32\right)$

Đưa ra $b^{(i+1)}$.

8.1.7 Hàm *F*

Hàm F dùng phép toán trên trường hữu hạn *GF(2⁸)*. Trong biểu diễn đa thức, *GF(2⁸)* được thực hiện như $GF(2)[x]/f(x)$, trong đó $f(x)$ là đa thức bất khả quy bậc 8 được xác định trên trường *GF(2)*. Bộ tạo khóa dòng MUGI sử dụng đa thức bất khả quy sau:

$f(x) = x^8 + x^4 + x^3 + x + 1$.

Hàm F là kết hợp của phép cộng khóa (việc bổ sung dữ liệu từ một phần của biến trạng thái b), phép biến đổi phi tuyến sử dụng hàm S_R, biến đổi tuyến tính sử dụng ma trận M và phép xáo trộng byte (xem Hình 7).

Chúng ta biểu diễn đầu vào và đầu ra cho hàm F tương ứng là X và Y. Khi đó, hàm F xác định như sau:

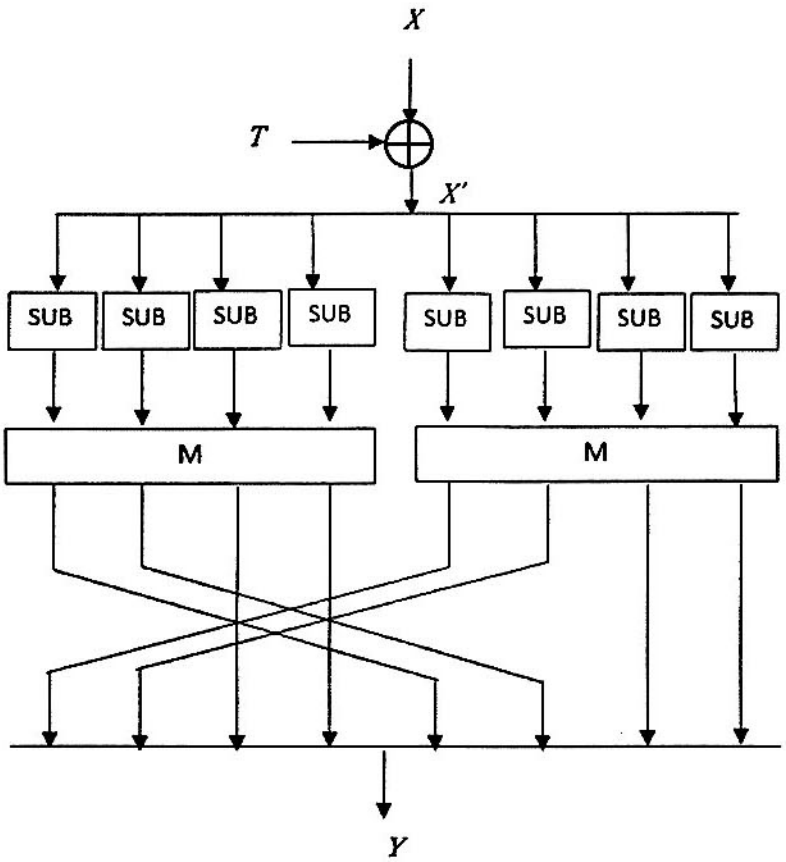
Đầu vào: hai xâu X và T độ dài 64-bit.

Đầu ra: xâu Y độ dài 64-bit.

- $X' = X \oplus T$

- Đặt $(X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7) = X'$ trong đó X_i là xâu có độ dài 8-bit
- Đặt $P_i = S_R(X_i)$ với $i = 0, 1, \ldots, 7$.
- Đặt $P_L = P_0 \| P_1 \| P_2 \| P_3$.
- Đặt $P_R = P_4 \| P_5 \| P_6 \| P_7$,
- Đặt $Q_L = M(P_L)$.
- Đặt $Q_R = M(P_R)$.
- Đặt $(Q_0, Q_1, Q_2, Q_3) = Q_L$.
- Đặt $(Q_4, Q_5, Q_6, Q_7) = Q_R$.,
- Đặt $Y = Q_4 \| Q_5 \| Q_2 \| Q_3 \| Q_0 \| Q_1 \| Q_6 \| Q_7$.
- Đưa ra Y.

Hình 7 mô tả sơ đồ khối của hàm F



Hình 7 – Hàm F của MUGI

8.1.8 Hàm S_R

Hàm S_R là hàm nội tại của hàm F. Hàm S_R có thể được mô tả bằng cách sử dụng bảng thay thế. Trong trường hợp này, hàm S_R như sau:

Đầu vào: xâu x độ dài 8-bit

Đầu ra: xâu y độ dài 8-bit.

- Đặt $y = SUB[x]$

- Đưa ra y.

SUB sử dụng trong hàm S_R là thay thế như sau:

```
SUB[256] = {
0x63,0x7c,0x77,0x7b,0xf2,0x6b,0x6f,0xc5,0x30,0x01,0x67,0x2b,0xfe,0xd7,0xab,0x76,
0xca,0x82,0xc9,0x7d,0xfa,0x59,0x47,0xf0,0xad,0xd4,0xa2,0xaf,0x9c,0xa4,0x72,0xc0,
0xb7,0xfd,0x93,0x26,0x36,0x3f,0xf7,0xcc,0x34,0xa5,0xe5,0xf1,0x71,0xd8,0x31,0x15,
0x04,0xc7,0x23,0xc3,0x18,0x96,0x05,0x9a,0x07,0x12,0x80,0xe2,0xeb,0x27,0xb2,0x75,
0x09,0x83,0x2c,0x1a,0x1b,0x6e,0x5a,0xa0,0x52,0x3b,0xd6,0xb3,0x29,0xe3,0x2f,0x84,
0x53,0xd1,0x00,0xed,0x20,0xfc,0xb1,0x5b,0x6a,0xcb,0xbe,0x39,0x4a,0x4c,0x58,0xcf,
0xd0,0xef,0xaa,0xfb,0x43,0x4d,0x33,0x85,0x45,0xf9,0x02,0x7f,0x50,0x3c,0x9f,0xa8,
0x51,0xa3,0x40,0x8f,0x92,0x9d,0x38,0xf5,0xbc,0xb6,0xda,0x21,0x10,0xff,0xf3,0xd2,
0xcd,0x0c,0x13,0xec,0x5f,0x97,0x44,0x17,0xc4,0xa7,0x7e,0x3d,0x64,0x5d,0x19,0x73,
0x60,0x81,0x4f,0xdc,0x22,0x2a,0x90,0x88,0x46,0xee,0xb8,0x14,0xde,0x5e,0x0b,0xab,
0xe0,0x32,0x3a,0x0a,0x49,0x06,0x24,0x5c,0xc2,0xd3,0xac,0x62,0x91,0x95,0xe4,0x79,
0xe7,0xc8,0x37,0x6d,0x8d,0xd5,0x4e,0xa9,0x6c,0x56,0xf4,0xea,0x65,0x7a,0xae,0x08,
0xba,0x78,0x25,0x2e,0x1c,0xa6,0xb4,0xc6,0xe8,0xdd,0x74,0x1f,0x4b,0xbd,0x8b,0x8a,
0x70,0x3e,0xb5,0x66,0x48,0x03,0xf6,0x0e,0x61,0x35,0x57,0xb9,0x86,0xc1,0x1d,0x9e,
0xe1,0xf8,0x98,0x11,0x69,0xd9,0x8e,0x94,0x9b,0x1e,0x87,0xe9,0xce,0x55,0x28,0xdf,
0x8c,0xa1,0x89,0x0d,0xbf,0xe6,0x42,0x68,0x41,0x99,0x2d,0x0f,0xb0,0x54,0xbb,0x16}
```

8.1.9 Hàm M

Hàm M là hàm nội tại của hàm F. Hàm M như sau:

Đầu vào: xâu X có độ dài 32-bit.

Đầu ra: xâu Y có độ dài 32-bit

- Đặt $(x_0, x_1, x_2, x_3) = X$, trong đó x_t là xâu có độ dài 8-bit và là phần tử của $GF(2^8)$.

- Đặt

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0x02 & 0x03 & 0x01 & 0x01 \\ 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Trong đó 0x01, 0x02 và 0x03 là biểu diễn thập lục phân của các phần tử của trường $GF(2^8)$.

- Đặt $Y = y_0 \| y_1 \| y_2 \| y_3$.

- Đưa ra Y.

8.2 Bộ tạo khóa dòng SNOW 2.0

8.2.1 Giới thiệu SNOW 2.0

SNOW 2.0, trong phần tiếp theo chỉ đơn giản ký hiệu là SNOW, là bộ tạo khóa dòng sử dụng như là đầu vào khóa bí mật K có độ dài 128 hoặc 256-bit và một véc tơ khởi tạo IV có độ dài 128-bit. Chúng được sử dụng để khởi tạo biến trạng thái S_i ($i \geq 0$) bao gồm 18 khối n = 32 bit. Thứ tự bit/byte là big-endian, tức là nếu khóa và véc tơ khởi tạo được cho dưới dạng một dãy các bit/byte, bit/byte phần đầu/tận cùng bên trái có trọng số cao nhất của dữ liệu tương ứng. Đối với mỗi lần lặp hàm $Strm$, 32-bit khóa dòng Z_i được tạo coi như đầu ra.

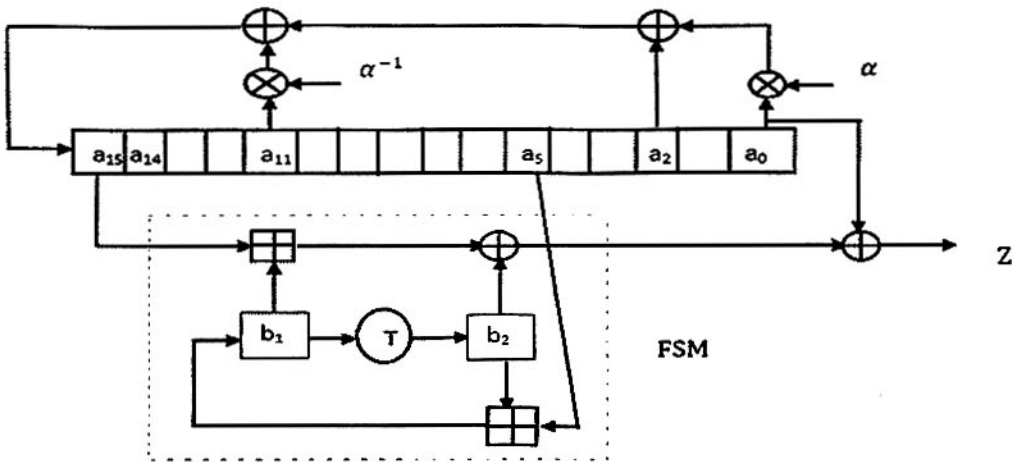
Biến trạng thái SNOW S_i bao gồm hai phần. Phần thứ nhất, 16 biến có độ dài 32-bit:

$$a^{(i)} = (a_{15}^{(i)}, a_{14}^{(i)}, \ldots, a_0^{(i)})$$

thực hiện một thanh ghi dịch phản hồi tuyến tính (LFSR). Phần thứ hai, 2 biến có độ dài 32-bit:

$$b^{(i)} = (b_2^{(i)}, b_1^{(i)})$$

duy trì trạng thái của máy trạng thái hữu hạn (FSM). SNOW được hiểu là tốt nhất với tham chiếu trong Hình 8, trong đó cho thấy một ảnh chụp, tại thời điểm i, bỏ qua biến phụ thuộc thời gian (i).



Hình 8 – Biểu đồ của SNOW

Phép toán SNOW được xác định:

Hàm *Init*, xác định chi tiết trong 8.2.2, nhận đầu vào khóa K có độ dài 128 hoặc 256-bit và véc tơ khởi tạo IV có độ dài 128-bit và tạo giá trị khởi tạo của biến trạng thái $S_0 = (a^{(0)}, b^{(0)})$.

Hàm $Next$, được xác định chi tiết trong 8.2.3, nhận đầu vào 18 biến trạng thái $S_i = (a^{(i)}, b^{(i)})$ có độ dài 32-bit và tạo đầu ra là giá trị tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$. Hàm $Next$ chạy trong 2 chế độ, tùy thuộc vào việc thực hiện lặp là một phần của việc khởi tạo, hoặc, trong chế độ bình thường của tạo đầu ra, xem dưới đây.

Hàm $Strm$, được xác định chi tiết trong 8.2.4 nhận đầu vào là 18 biến trang thái có độ dài 32-bit $S_i = (a^{(i)}, b^{(i)})$ và tạo ra như là đầu ra khóa dòng Z_i có độ dài 32-bit.

CHÚ THÍCH 1 Đối với SNOW, khuyến nghị số lượng tối đa khóa dòng được tạo ra từ (*K,IV*) là 2^{50} khóa có độ dài 32-bit. Giới hạn này đã được lựa chọn để cung cấp biên an toàn tốt nhất đối với việc thám mã và ngụ ý không có giới hạn thực tế áp dụng các thuật toán.

CHÚ THÍCH 2 Bài viết [10] được tham chiếu cho lý thuyết nền tảng về lý do căn bản thiết kế cho SNOW.

8.2.2 Hàm khởi tạo *Init*

Hàm khởi tạo *Init* như sau.

Đầu vào: Khóa K có độ dài 128 hoặc 256-bit, Véc tơ khởi tạo IV có độ dài 128-bit.

Đầu ra: Giá trị khởi tạo của biến trạng thái $S_0 = (a^{(0)}, b^{(0)})$.

a) Khởi tạo thanh ghi bằng thông tin khóa.

- Đối với khóa có độ dài 128-bit, Đặt $(K_3, K_2, K_1, K_0) = K, a_{15-j}^{(-34)} = a_{15-j-8}^{(-34)} = K_{3-j}$ và $a_{15-j-4}^{(-34)} = a_{15-j-12}^{(-34)} = \neg(K_{3-j})$ với $j = 0, 1, 2, 3$.
- Đối với khóa có độ dài 256-bit, Đặt $(K_7, K_6, \dots, K_0) = K, a_{15-j}^{(-34)} = K_{7-j}$ và $a_{15-j-8}^{(-34)} = \neg(K_{7-j})$ với $j = 0, 1, \dots, 7$.

b) Đặt $S_{-33} = (a^{(-33)}, b^{(-33)})$ bằng

- Đặt $(IV_3, IV_2, IV_1, IV_0) = IV$.
- Đặt $a_i^{(-33)} = a_i^{(-34)}$ với i = 0, 1, 2, 3, 4, 5, 6, 7, 8, 11, 13, 14.
- Đặt $a_{15}^{(-33)} = a_{15}^{(-34)} \oplus IV_0$; $a_{12}^{(-33)} = a_{12}^{(-34)} \oplus IV_1$; $a_{10}^{(-33)} = a_{10}^{(-34)} \oplus IV_2$; $a_9^{(-33)} = a_9^{(-34)} \oplus IV_3$.
- Đặt $b_1^{(-33)} = b_2^{(-33)} = 0^{(32)}$.

c) Đặt $S_{-1} = Next^{32}(S_{-33}, INIT)$, trong đó $Next^{32}$ biểu diễn cho 32 lần lặp của hàm $Next$

d) $S_0 = Next(S_{-1})$.

e) Đưa ra S_0.

8.2.3 Hàm chuyển trạng thái theo *Next*

SNOW có hai chế độ với hàm $Next$

Đầu vào: Biến trạng thái $S_i = (a^{(i)}, b^{(i)})$, $mode = \{INIT, null\}$.

Đầu ra: Giá trị tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$.

a) Đặt $b_2^{(i+1)} = T\left(b_1^{(i)}\right)$.

b) Đặt $b_1^{(i+1)} = b_2^{(i)} +_{32} a_5^{(i)}$

c) Với $j = 0, 1, \ldots, 14$ đặt $a_j^{(i+1)} = a_{j+1}^{(i)}$

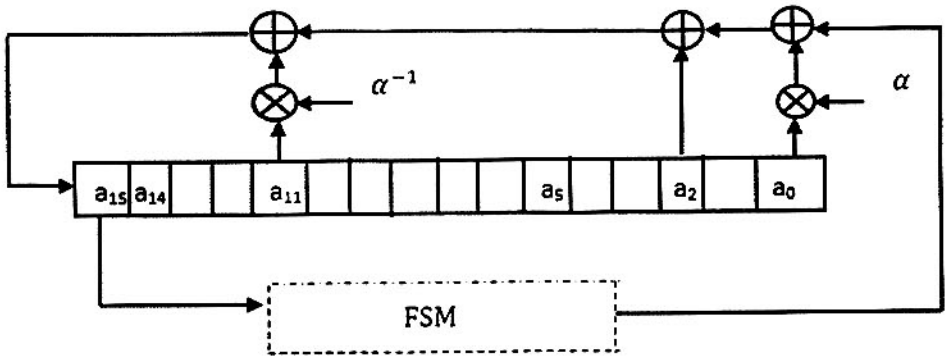
d) Nếu chế độ *INIT*, đặt $a_{15}^{(i+1)} = \left(a_0^{(i)} \otimes \alpha\right) \oplus a_2^{(i)} \oplus \left(a_{11}^{(i)} \otimes \alpha^{-1}\right) \oplus FSM(a_{15}^{(i)}, b_1^{(i)}, b_2^{(i)})$. Nếu không thì, đặt $a_{15}^{(i+1)} = \left(a_0^{(i)} \otimes \alpha\right) \oplus a_2^{(i)} \oplus \left(a_{11}^{(i)} \otimes \alpha^{-1}\right)$.

e) $S_{i+1} = (a^{(i+1)}, b^{(i+1)})$

f) Đưa ra S_{i+1}.

Mô tả của hàm T và số học trường hữu hạn bao gồm thành phần cố định α đề cập tương ứng trong điều 8.2.5 và 8.2.6.

CHÚ THÍCH Hình 9 mô tả sơ đồ khối của chế độ INIT của hàm *Next*.



Hình 9 – Chế độ INIT của hàm *Next*

Định nghĩa của hàn *FSM* đề cập trong điều 8.2.8.

8.2.4 Hàm khóa dòng *Strm*

Hàm khóa dòng *Strm* như sau:

Đầu vào: Biến trạng thái S_i

Đầu ra: Khóa dòng Z_i có độ dài 32-bit

a) Đặt $Z_i = FSM\left(a_{15}^{(i)}, b_1^{(i)}, b_2^{(i)}\right) \oplus a_0^{(i)}$.

b) Đưa ra Z_i.

8.2.5 Hàm *T*

Hàm T là hàm thay thế, đặc biệt là hoán vị trên trường $GF(2^{32})$, dựa trên các thành phần từ Chuẩn mã hóa tiên tiến (AES), TCVN 11367-3:2016 (ISO/IEC 18033-3). Cuối cùng trường hữu hạn $GF(2^8)$ được sử dụng như là trường $GF(2)[x]$ modulo với đa thức bất khả quy.

$$f(x) = x^8 + x^4 + x^3 + x + 1.$$

Và vành đa thức $GF(2^8)[y]$ modulo $(y^4 + 1)$.

Đầu vào: xâu w có độ dài 32-bit

Đầu ra: xâu $q = T(w)$ có độ dài 32-bit

a) Đặt $(w_3, w_2, w_1, w_0 = w)$ trong đó mỗi w_j có độ dài 8 bit.

b) Với $j = 0, 1, 2, 3$ đặt $t_j = SUB[w_j]$.

c) Lấy $t(y)$ là đa thức $t(y) = t_3y^3 + t_2y^2 + t_1y + t_0$ trên trường $GF(2^8)[y]$, trong đó t_j được hiểu như là một phần tử của $GF(2^8)$ theo cách tự nhiên $t_j = t_{j,7}x^7 + \cdots + t_{j,1}x + t_{j,0}$, $t_{j,k}$ trong trường $GF(2)$

d) Đặt $q(y) = c(y).t(y)\ modulo\ (y^4 + 1)$, trong đó $c(y) = (x + 1)y^3 + y^2 + y + x$ trên trường $GF(2^8)[y]$.

e) Kết hợp xâu $q = (q_3, q_2, q_1, q_0)$ 32-bit với kết quả trên, $q(y) = q_3y^3 + q_2y^2 + q_1y + q_0$

f) Đưa ra q.

Lưu ý rằng trong bước c), hai đa thức được nhân với nhau trong đó các phép toán hệ số-nhân-hệ số thực hiện trong trường $GF(2^8)$ được xác định bởi hàm $f(x)$ ở trên. Sau đó kết quả được rút gọn theo modulo $y^4 + 1$.

CHÚ THÍCH 1 Hộp thế *S-box* của AES được tìm thấy trong 8.1.8

CHÚ THÍCH 2 Tham khảo chi tiết tối ưu hóa này và một số vấn đề khác tại bài báo [10].

8.2.6 Phép nhân α trong số học trường hữu hạn

Đầu vào: Xâu w có độ dài 32-bit, đại diện cho phần tử của $GF(2^{32})$.

Đầu ra: Xâu w' có độ dài 32-bit, đại diện cho $\propto \otimes w$ trong trường $GF(2^{32})$.

a) Đặt $w' = (w \ll_{32} 8) \oplus \alpha_{MUL}[w \gg_{32} 24]$

b) Đưa ra w'

Hàm α_{MUL} được xác định như sau:

$\alpha_{MUL}[256] = \{$

```
0x00000000,0xE19FCF13,0x68973726,0x8A03F835,0xD6876E4C,0x3718A15F,0xBD10596A,0x5C8F9679,
0x05A7DC98,0x2438138B,0x6E30E3BE,0x8FAF24AD,0xD320B2D4,0x32BF7DC7,0xB8B785F2,0x59284AE1,
0x0AE71199,0xEB78DE8A,0x617026BF,0x80EFE9AC,0xDC607FD5,0x3DFFB0C6,0xB7F748F3,0x566887E0,
0x0F40CD01,0xEEDF0212,0x64D7FA27,0x85483534,0xD9C7A34D,0x38586C5E,0xB2509463,0x53CF5B78,
0x14672293,0xF5F8ED88,0x7FF015BD,0x9E6FDAAE,0xC2ECf4CD7,0x237F83C4,0xA97773F1,0x48E834E2,
0x11C0FE03,0xF05F3110,0x7A57C925,0x9BC80636,0xC747904F,0x26D85F5C,0xACD0A769,0x4D4F687A,
0x1E803302,0xFF1FFC11,0x75170424,0x9488CB37,0xC8075D4E,0x29989250,0xA3906A68,0x420FA57B,
```

0x1B27EF9A, 0xFAB82089, 0x70B0D83C, 0x912F17AF, 0xCDA081D6, 0x2C3F4EC5, 0xA637B6F0, 0x47A879E3,
0x28CE449F, 0xC951838C, 0x43597339, 0xA2C63CAA, 0xFE492AD3, 0x1FD6E5C0, 0x95DE1DF5, 0x7441D2E6,
0x20699807, 0xCCF65714, 0x46FEAF21, 0xA7616032, 0xFBEEF643, 0x1A713958, 0x9079C16D, 0x71E60E7E,
0x22295506, 0xC3369A15, 0x49BE6220, 0xA821AD33, 0xF4AE3B4A, 0x1531F459, 0x9F390C6C, 0x7EA6C37F,
0x278E899E, 0xC611468D, 0x4C19BEB8, 0xAD8671AB, 0xF109E7D2, 0x109628C1, 0x9A9ED0F4, 0x7B011FE7,
0x3CA96604, 0xDD36A917, 0x573E5122, 0xB6A19E31, 0xEA2E0848, 0x0BB1C75B, 0x81B93F6E, 0x6026F07D,
0x390EBA9C, 0xD891758F, 0x52998DBA, 0xB30642A9, 0xEF89D4C0, 0x0E161BC3, 0x841EE3F6, 0x65812CE5,
0x364E779D, 0xD7D1B88E, 0x5DD940BB, 0xBC468FA8, 0xE0C919D1, 0x0156D6C2, 0x8B5E2EF7, 0x6AC1E1E4,
0x33E9AB05, 0xD2766416, 0x587E9C23, 0xB9E15330, 0xE56EC549, 0x04F10A5A, 0x8EF9F26F, 0x6F663D7C,
0x50358897, 0XB1AA4784, 0x3BA2BF31, 0xDA3D70A2, 0x86B2E6DB, 0x672D29C8, 0XED25D1FD, 0x0CBA1EEE,
0x5592540F, 0xB40D931C, 0x3E056329, 0xDF9AAC3A, 0x83153A43, 0x628AF550, 0xE8820D65, 0x091DC276,
0x5AD2990S, 0xBB4D561D, 0x3145AE28, 0xD0DA613B, 0x8C55F742, 0x6DCA3851, 0xE7C2C064, 0x065D0F77,
0x5F754596, 0xBSEA8A85, 0x34E27230, 0xD57D3DA3, 0x89F22BDA, 0x686DE4C9, 0xE2651CFC, 0x03FAD3EF,
0x4452AA0C, 0xA5CD651F, 0x2FC59D2A, 0xCE5A5239, 0x92D5C440, 0x734A0B53, 0xF942F366, 0x18DD3C75,
0x41F57694, 0xA06AB987, 0x2A6241B2, 0xCBFD8EA1, 0x977218D8, 0x76EDD7C3, 0xFCE52FFE, 0x1D7AE0ED,
0x4EB5BB95, 0xAF2A7486, 0x25228CB3, 0xC4BD43A0, 0x9832D5D9, 0x79AD1ACA, 0xF3A5E2FF, 0xI23A2DEC,
0x4B12670D, 0xAA8DA81E, 0x20855023, 0xC11A9F38, 0x9D950941, 0x7C0AC652, 0xF6023E67, 0x179DF174,
0x78FBCC08, 0x996403IB, 0x136CFB2E, 0xF2F3343D, 0xAE7CA244, 0x4FE36D57, 0xC5EB9562, 0x24745A71,
0x7D5C1090, 0x9CC3DF83, 0x16CB27B6, 0xF754E8A5, 0xABDB7EDC, 0x4A44B1CF, 0xC04C49FA, 0x21D386E9,
0x721CDD91, 0x93831282, 0x198BEA37, 0xF81425A4, 0xA49BB3DD, 0x45047CCE, 0xCF0C84FB, 0x2E934BE8,
0x77BB0109, 0x9624CE1A, 0x1C2C362F, 0xFD33F93C, 0xA13C6F45, 0x40A3A056, 0xCAAB5863, 0x2B349770,
0x6C9CEE93, 0x8D032180, 0x070BD9B5, 0xE69416A6, 0xBA1B80DF, 0x5B844FCC, 0xD18CB7F9, 0x301378EA,
0x693B320B, 0x88A4FD18, 0x02AC052D, 0xE333CA3E, 0xBFBC5C47, 0x5E239354, 0xD4236B61, 0x35B4A472,
0x6673FF0A, 0x87E43019, 0x0DECC82C, 0xEC73073F, 0xB0FC9146, 0x51635E55, 0xDB63A660, 0x3AF46973,
0x63DC2392, 0x8243EC81, 0x084B14B4, 0xE9D4DBA7, 0xB55B4DDE, 0x54C482CD, 0xDECC7AF8, 0x3F53B5E3};

8.2.7 Phép nhân α^{-1} trong số học trường hữu hạn

Đầu vào: Xâu y có độ dài 32-bit, đại diện cho phần tử của $GF(2^{32})$.

Đầu ra: Xâu y có độ dài 32-bit, đại diện cho $\alpha^{-1} \otimes y$ trong trường $GF(2^{32})$.

a) Đặt $y' = (y \gg_{32} 8) \otimes \alpha_{inv\_MUL}[y \bmod 256]$

b) Đưa ra y'

Hàm α_{inv_MUL} được xác định như sau:

α_{inv_MUL} [256] = {

0x00000000, 0x180F40CD, 0x301E6033, 0x2811C0FE, 0x603CA966, 0x7833E9A3, 0x50222955, 0x482D6998,
0xC078FBCC, 0xD877BB01, 0xF0667BFF, 0xE8693332, 0xA04452AA, 0xB84B1267, 0x905AD299, 0x88559254,
0x29F05F31, 0x31FF1FFC, 0x19EEDF02, 0x01E19FCF, 0x49CCF657, 0x51C3B69A, 0x79D27664, 0x61DD36A9,
0xE988A4FD, 0xF187E430, 0xD99624CE, 0xd996403, 0x89B40D9B, 0x91BB4D56, 0xB9AA8DA8, 0xA1A5CD65,
0x5249BE62, 0x4A46FEAF, 0x62573E51, 0x7A587E9C, 0x32751704, 0x2A7A57C9, 0x026B9737, 0x1A64D7FA,

0x923145AE, 0x8A3E0563, 0xA22FC59D, 0xBA208550, 0xF20DECC8, 0xEA02AC05, 0xC2136CFB, 0xDA1C2C36,
0x7BB9E153, 0x63B6A19E, 0x4BA76160, 0x53A821AD, 0x1B854835, 0x038A08F8, 0x2B9BC806, 0x339488CE,
0xBBC11A9F, 0xA3CE5A52, 0x8BDF9AAC, 0x93D0DA61, 0xDBFDB3F9, 0xC3F2F334, 0xE3E333CA, 0xF3EC7307,
0xA492D5C4, 0xBC9D9509, 0x948C55F7, 0x8C83153A, 0xC4AE7CA2, 0xDCA13C6F, 0xF4B0FC91, 0xECBFBC5C,
0x64EA2E08, 0x7CE56EC5, 0x54F4AE3B, 0x4CF3EEF6, 0x04D6876E, 0x1CD9C7A3, 0x34C8075D, 0x2CC74790,
0x8D628AF5, 0x956DCA38, 0xBD7C0AC6, 0xA5734A0B, 0xED5E2393, 0xF551635E, 0xDD40A3A0, 0xC54FE36D,
0x4D1A7139, 0x551531F4, 0x7D04F10A, 0x650BB1C7, 0x2D26D85F, 0x35299892, 0x1D38586C, 0x053718A1,
0xF6D36BA6, 0xEED42B6B, 0xC6C5E395, 0xDECAAB58, 0x96E7C2C0, 0x8EE8820D, 0xA6F942F3, 0x3EF6023E,
0x36A3906A, 0x2EACD0A7, 0x06BD1059, 0x1EB25094, 0x569F390C, 0x4E9079C1, 0x6681B93F, 0x7E8EF9F2,
0xDF2B3497, 0xC724745A, 0xEF35B4A4, 0xF73AF469, 0xBF179DF1, 0xA718DD3C, 0x8F091DC2, 0x97065D0F,
0x1F53CF5B, 0x075C8F96, 0x2F4D4F68, 0x37420FA5, 0x7F6F663D, 0x676026F0, 0x4F71E60E, 0x577EA6C3,
0xE18D0321, 0xF98243EC, 0xD1938312, 0xC99CC3DF, 0x81B1AA47, 0x99BEEA8A, 0xB1AF2A74, 0xA9A06AB9,
0x21F5F8SD, 0x39FAB820, 0x11EB78DE, 0x09E43813, 0x41C9518B, 0x59C61146, 0x71D7D1B8, 0x69D89175,
0xC87D5C10, 0xD0721CDD, 0xF863DC23, 0xE06C9CEE, 0xA841F576, 0xB04EB5B3, 0x985F7545, 0x80503588,
0x0805A7DC, 0x100AE711, 0x381B27EF, 0x20146722, 0x68390EEA, 0x70364E77, 0x58278E89, 0x4028CE44,
0xB3C4BD43, 0xABCBFDSE, 0x83DA3D70, 0x93D57DBD, 0xD3F81425, 0xCBF754E8, 0xE3E69416, 0xFBE9D4DB,
0x733C468F, 0x6BB30642, 0x43A2C6BC, 0x5B'\D8671, 0x1380EFE9, 0x0B8FAF24, 0x239E6FDA, 0x3B912F17,
0x9A34E272, 0x823BA2BF, 0xAA2A6241, 0xB225223C, 0xFA084314, 0xE2070BD9, 0xCA16CB27, 0xD2198BEA,
0x5A4C19BE, 0x42435973, 0x6A52998D, 0x725DD940, 0x3A70E0D8, 0x227FF015, 0x0A6E30EB, 0x12617026,
0x451FD6E5, 0x5D109628, 0x750156D6, 0x6D0E161B, 0x25237F83, 0x3D2C3F4E, 0x153DFFB0, 0x0D323F7D,
0x85672D2S, 0x9D686DE4, 0xB579AD1A, 0xAD76EDD7, 0xE55B844F, 0xFD54C482, 0xD545047C, 0xCD4A44B1,
0x6CEF89D4, 0x74E0C919, 0x5CF109E7, 0x44FE492A, 0x0CD320B2, 0x14DC607F, 0x3CCDA081, 0x24C2E04C,
0xAC977218, 0x349832D5, 0x9C89F22B, 0x8486B2E6, 0xCCABDB7E, 0xD4A493B3, 0xFCB5534D, 0xE4BA1B80,
0x17566887, 0x0F59284A, 0x2748E8B4, 0x3F47A879, 0x776AC1E1, 0x6F65812C, 0x477441D2, 0x5F7B011F,
0xD72E934B, 0xCF21D380, 0xE7301378, 0xFF3F5335, 0xB7123A2D, 0xAF1D7AE0, 0x870CBA1E, 0x9F03FAD3,
0x3EA637B6, 0x26A9777B, 0x0EB8B785, 0x16B7F748, 0x5E9A9ED0, 0x4695DE1D, 0x6E841EE3, 0x76835E2E,
0xFEDECC7A, 0xE6D18CB7, 0xCEC04C49, 0xD6CF0C84, 0x9EE2651C, 0x86ED25D1, 0xAEFCE52F, 0xB6F3A5E2);

8.2.8. Hàm $FSM(x, y, z)$

Đầu vào: 3 xâu có độ dài 32-bit, x, y và z

Đầu ra: xâu q có độ dài 32-bit

a) Đặt $q = (x +_{32} y) \oplus z$.

b) Đưa ra q.

8.3 Bộ tạo khóa dòng Rabbit

8.3.1 Tổng quan bộ tạo khóa dòng Rabbit

Rabbit là bộ tạo khóa dòng sử dụng khóa bí mật K có độ dài 128-bit, véc tơ khởi tạo IV có độ dài 64-bit và biến trạng thái trong S_i ($i \geq 0$) có độ dài 513-bit. Rabbit đưa ra khối khóa dòng Z_i có độ dài 128-bit tại mỗi lần lặp hàm $Strm$.

513-bit của trạng thái trong S_i được chia giữa 8 biến trạng thái có độ dài 32-bit $X_0^{(i)}, \ldots, X_7^{(i)}$, 8 biến đếm $C_0^{(i)}, \ldots, C_7^{(i)}$, và 1 bit nhớ bộ đếm $b^{(i)}$.

Mô tả sử dụng các ký hiệu đưa ra tại Điều 4 của tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033). Ngoài ra, ký hiệu đặc biệt cho mảng bit được sử dụng để tăng cường khả năng đọc: Khi ghi nhãn các bit của một biến A, bit có trọng số thấp nhất được ký hiệu là $A^{(0)}$. Ký hiệu $A^{[h..g]}$ mô tả các bit từ h cho đến g của biến A, trong đó bit có vị trí h là bit có trọng số cao hơn bit ở vị trí g.

CHÚ THÍCH 1 Đối với bộ sinh khóa dòng Rabbit, khuyến nghị số lượng khóa dòng tối đa được tạo ra từ một khóa K cho trước là 2^{64} khối khóa dòng. Giới hạn này đã được lựa chọn để cung cấp biên an toàn tốt nhất đối với việc phân tích mã và đồng thời ngụ ý không có giới hạn thực tế về khả năng áp dụng các thuật toán.

CHÚ THÍCH 2 Bài báo [8] được tham chiếu cho đề xuất ban đầu của mật mã và bài báo [9] được tham chiếu đến tổng quan về an toàn mật mã của nó.

8.3.2 Các biến bổ sung và ký hiệu

Đối với bộ tạo khóa dòng Rabbit, các ký hiệu sau đây được sử dụng thêm:

A	Hằng số cho Rabbit
b	Bit nhớ cho Rabbit
C	Biến đếm cho Rabbit
g	Hàm con được sử dụng cho Rabbit
X	Biến trạng thái trong cho Rabbit

Ngoài ra, một số ký hiệu được sử dụng cho các biến phụ cục bộ trong mô tả các thuật toán. Những biểu tượng này chỉ xảy ra trong đặc tả hàm đưa ra và không có nghĩa toàn cục. Các biến này được mô tả trong phần khai báo của hàm.

8.3.3 Hàm khởi tạo *Init*

Trong phần tiếp theo, hàm khởi tạo $Init$ cho Rabbit được quy định.

Đầu vào: Khóa K có độ dài 128-bit, véc tơ khởi tạo IV có độ đài 64-bit.

Đầu ra: Giá trị khởi tạo của biến trạng thái $S_0 = (b_0, X_0^{(0)}, \ldots, X_7^{(0)}, C_0^{(0)}, \ldots, C_7^{(0)})$.

Biến cục bộ: biến đếm i, j

a) Lấy $K_0 = K^{[15..0]}, K_1 = K^{[31..16]}, \ldots$, và $K_7 = K^{[127..112]}$

b) Đặt S_{-9} như sau:

1) Đặt $b^{(-9)} = 0$.

2) Với $j = 0, 1, \ldots, 7$:

- Nếu j là chẵn, đặt $X_j^{(-9)} = K_{(j+1 \bmod 8)} \,||\, K_j$ và $C_j^{(-9)} = K_{(j+4 \bmod 8)} \,||\, K_{(j+5 \bmod 8)}$.

- Ngược lại j lẻ đặt $X_j^{(-9)} = K_{(j+5 \bmod 8)} \,||\, K_{(j+4 \bmod 8)}$ và $C_j^{(-9)} = K_j \,||\, K_{(j+1 \bmod 8)}$.

c) Lặp lại các hàm trạng thái tiếp theo $Next$ 4 lần: đặt $S_i = Next(S_{i-1})$ với $i = -8, -7, -6, -5$

d) Đặt S_{-4} như sau:

1) Thay đổi các biến đếm như sau:

$$C_0^{(-4)} = C_0^{(-5)} \oplus X_4^{(-5)} \oplus IV^{[31\ldots0]} \qquad C_1^{(-4)} = C_1^{(-5)} \oplus X_5^{(-5)} \oplus (IV^{[63\ldots48]} \,||\, IV^{[31\ldots16]})$$

$$C_2^{(-4)} = C_2^{(-5)} \oplus X_6^{(-5)} \oplus IV^{[63\ldots32]} \qquad C_3^{(-4)} = C_3^{(-5)} \oplus X_7^{(-5)} \oplus (IV^{[47\ldots32]} \,||\, IV^{[15\ldots0]})$$

$$C_4^{(-4)} = C_4^{(-5)} \oplus X_0^{(-5)} \oplus IV^{[31\ldots0]} \qquad C_5^{(-4)} = C_5^{(-5)} \oplus X_1^{(-5)} \oplus (IV^{[63\ldots48]} \,||\, IV^{[31\ldots16]})$$

$$C_6^{(-4)} = C_6^{(-5)} \oplus X_2^{(-5)} \oplus IV^{[63\ldots32]} \qquad C_7^{(-4)} = C_7^{(-5)} \oplus X_3^{(-5)} \oplus (IV^{[47\ldots32]} \,||\, IV^{[15\ldots0]})$$

2) $X_0^{(-4)} = X_0^{(-5)}, \ldots, X_7^{(-4)} = X_7^{(-5)}, b^{(-4)} = b^{(-5)}$

e) Lặp lại hàm chuyển trạng thái theo $Next$ 4 lần: đặt $S_i = Next(S_{i-1})$ với $i = -3, -2, -1, 0$

f) Đưa ra $S_0 = (b_0, X_0^{(0)}, \ldots, X_7^{(0)}, C_0^{(0)}, \ldots, C_7^{(0)})$

CHÚ THÍCH Véc tơ khởi tạo IV được trộn vào trạng thái trong ở bước d) và e) của thuật toán. Nếu các ứng dụng yêu cầu tái khởi tạo thường xuyên với cùng một khóa, điều này có nghĩa để lưu trữ trạng thái trong sau bước c) như trạng thái chủ và để thực thi chỉ bước d) tới bước f) cho tái khởi tạo.

8.3.4 Hàm chuyển trạng thái theo *Next*

Hàm chuyển trạng thái theo $Next$ cho Rabbit được quy định như sau:

Đầu vào: Biến trạng thái $S_i = (b^{(i)}, X_0^{(i)}, \ldots, X_7^{(i)}, C_0^{(i)}, \ldots, C_7^{(i)})$.

Đầu ra: Biến trạng thái $S_{i+1} = (b^{(i+1)}, X_0^{(i+1)}, \ldots, X_7^{(i+1)}, C_0^{(i+1)}, \ldots, C_7^{(i+1)})$.

Biến cục bộ: biến đếm j, biến tạm là số nguyên dương có độ dài 32-bit

a) Đặt hằng số $A_0, \ldots A_7$ như sau:

A_0= 0x4D34D34D A_1= 0xD34D34D3 A_2= 0x34D34D34 A_3= 0x4D34D34D

A_4= 0xD34D34D3 A_5= 0x34D34D34 A_6= 0x4D34D34D A_7= 0xD34D34D3

b) Lấy $b_0^{(i+1)} = b^i$

c) Với $j = 0, 1, 2, \ldots, 7$:

- Lấy $temp = C_j^{(i)} + A_i + b_j^{(i+1)}$; kết quả này là giá trị có độ dài 33-bit

- Lấy $b_{j+1}^{(i+1)} = temp^{[32]}$

- Lấy $C_j^{(i+1)} = temp^{[31...0]}$

d) Lấy $b^{(i+1)} = b_8^{(i+1)}$

e) Với $j = 0, 1, ..., 7$, lấy $G_j = g(X_j^{(i)}, C_j^{(i+1)})$, trong đó hàm g được đưa ra trong 8.3.6.

f) Thay đổi trạng thái trong như sau:

$X_0^{(i+1)} = G_0 +_{32} (G_7 \lll_{32} 16) +_{32} (G_6 \lll_{32} 16)$

$X_1^{(i+1)} = G_1 +_{32} (G_0 \lll_{32} 8) +_{32} G_7$

$X_2^{(i+1)} = G_2 +_{32} (G_1 \lll_{32} 16) +_{32} (G_0 \lll_{32} 16)$

$X_3^{(i+1)} = G_3 +_{32} (G_2 \lll_{32} 8) +_{32} G_1$

$X_4^{(i+1)} = G_4 +_{32} (G_3 \lll_{32} 16) +_{32} (G_2 \lll_{32} 16)$

$X_5^{(i+1)} = G_5 +_{32} (G_4 \lll_{32} 8) +_{32} G_3$

$X_6^{(i+1)} = G_6 +_{32} (G_5 \lll_{32} 16) +_{32} (G_4 \lll_{32} 16)$

$X_7^{(i+1)} = G_7 +_{32} (G_6 \lll_{32} 8) +_{32} G_5$

g) Đưa ra $S_{i+1} = (b^{(i+1)}, X_0^{(i+1)}, ..., X_7^{(i+1)}, C_0^{(i+1)}, ..., C_7^{(i+1)})$

8.3.5 Hàm khóa dòng *Strm*

Hàm khóa dòng *Strm* cho Rabbit được quy định như sau:

Đầu vào: Biến trạng thái $S_i = (b^{(i)}, X_0^{(i)}, ..., X_7^{(i)}, C_0^{(i)}, ..., C_7^{(i)})$.

Đầu ra: Khối khóa dòng Z_i.

a) Đặt Z_i như sau:

$$Z_i[15..0] = X_0^{(i)}[15..0] \oplus X_5^{(i)}[31..16]$$

$$Z_i[31..16] = X_0^{(i)}[31..16] \oplus X_3^{(i)}[15..0]$$

$$Z_i[47..32] = X_2^{(i)}[15..0] \oplus X_7^{(i)}[31..16]$$

$$Z_i[63..48] = X_2^{(i)}[31..16] \oplus X_5^{(i)}[15..0]$$

$$Z_i[79..64] = X_4^{(i)}[15..0] \oplus X_1^{(i)}[31..16]$$

$$Z_i[95..80] = X_4^{(i)}[31..16] \oplus X_7^{(i)}[15..0]$$

$$Z_i[111..96] = X_6^{(i)}[15..0] \oplus X_3^{(i)}[31..16]$$

$$Z_i[127..112] = X_5^{(i)}[31..16] \oplus X_1^{(i)}[15..0]$$

b) Đưa ra Z_i.

8.3.6 Hàm g

Hàm g được quy định như sau:

Đầu vào: 2 tham số u và v có độ dài 32-bit

Đầu ra: kết quả hàm $g(u,v)$ có độ dài 32-bit

Biến cục bộ: $temp$ số nguyên dương có độ dài 32-bit

a) Lấy $temp = (u +_{32} v)^2$; kết quả là giá trị có độ dài 64-bit.

b) Lấy $g(u,v) = temp^{[31..0]} \oplus temp^{[63..32]}$

c) Đưa ra $g(u,v)$

8.4 Bộ tạo khóa dòng $Decim^{v2}$

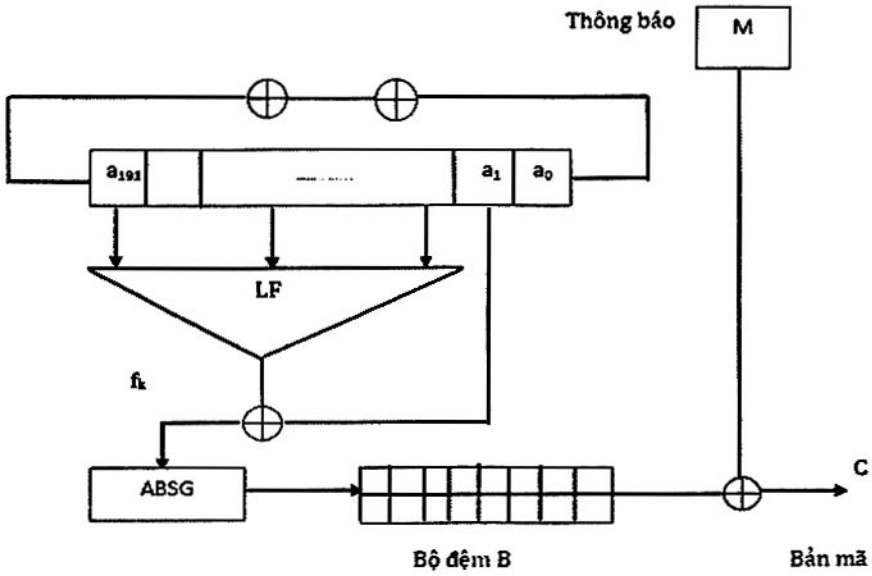
8.4.1 Giới thiệu bộ tạo khóa dòng $Decim^{v2}$

$DECIM^{v2}$ là bộ tạo khóa dòng trong đó sử dụng khóa bí mật K có độ dài 80-bit và véc tơ khởi tạo IV có độ dài 64-bit. $DECIM^{v2}$ bao gồm thanh ghi dịch phản hồi tuyến tính A có độ dài tối đa 192-bit, được lọc bởi hàm Boolean LF 14-biến. Trong chế độ tạo khóa dòng, đầu ra của LF được sử dụng để nuôi một khối nén là hàm được gọi là $ABSG$, mà đầu cuối đi qua một bộ đệm B dài 32-bit để điều chỉnh tốc độ đầu ra khóa dòng.

$DECIM^{v2}$ được mô tả trong Hình 10, trong đó mô tả ảnh chụp, đúng thời điểm, bỏ qua biến phụ thuộc thời gian (i) từ các ký hiệu.

CHÚ THÍCH 1 Bài báo [6] được tham chiếu cho lý thuyết nền tảng về lý do căn bản thiết kế của $DECIM^{v2}$.

Biến trạng thái S_i của $DECIM^{v2}$ bao gồm giá trị độ dài 192-bit của thanh ghi $a^i = (a_0^{(i)}, a_1^{(i)}, \dots, a_{191}^{(i)})$ và biến T^i độ dài 3-bit tương ứng với trạng thái của hàm nén $ABSG$, 32-bit $b^i = (b_0^{(i)}, b_1^{(i)}, \dots, b_{31}^{(i)})$ trong bộ đệm B, và số lượng $I^{(i)}$ bit trong bộ đệm B đã sẵn sàng đưa ra.



Hình 10 – Sơ đồ của $DECIM^{v2}$

Hàm $Init$, được xác định chi tiết trong 8.4.3, nhận đầu vào là khóa K độ dài 80-bit và véc tơ khởi tạo IV độ dài 64-bit và tạo ra giá trị khởi tạo của biến trạng thái $S_0 = (a^{(0)}, T^{(0)}, b^{(0)}, I^{(00)})$.

Hàm $Next$, được xác định chi tiết trong 8.4.5, nhận đầu vào là giá trị các biến trạng thái $S_i = (a^{(i)}, T^{(i)}, b^{(i)}, I^{(i)})$ và tạo ra đầu ra là giá trị tiếp của biến trạng thái $S_{i+1} = (a^{(i+1)}, T^{(i+1)}, b^{(i+1)}, I^{(i+1)})$. Hàm $Next$ chạy trong 3 chế độ, tùy thuộc vào việc thực hiện lặp lại là một phần của khởi tạo thanh ghi, khởi tạo của bộ đệm hoặc bộ tạo khóa dòng tuần tự.

Hàm $Strm$, được xác định chi tiết trong 8.4.6 nhận đầu vào là giá trị của các biến trạng thái $S_i = (a^{(i)}, T^{(i)}, b^{(i)}, I^{(i)})$ và tạo ra đầu ra là bit khóa dòng Z_i.

CHÚ THÍC 2: Tốc độ đầu ra chuẩn của $DECIM^{v2}$ là ¼. Vì vậy, để đồng bộ biến trạng thái và đầu ra khóa dòng, hàm $Next$ thực hiện 4 vòng lặp tiêu chuẩn của $DECIM^{v2}$ được quy định trong [6].

CHÚ THÍCH 3 Hàm nén của $DECIM^{v2}$ có tốc độ đầu ra thay đổi, bằng 1/3 mức trung bình. Vì vậy, một cơ chế đệm được sử dụng để đảm bảo tốc độ đầu ra cố định. Sự khác biệt giữa tốc độ đầu ra bộ đệm và tốc độ đầu ra hàm nén, cũng như độ dài bộ đệm, đã được lựa chọn để đảm bảo rằng các bộ đệm luôn luôn có các chức năng như mong muốn với xác suất đa số, như mô tả trong 8.4.3.

CHÚ THÍCH 4 $DECIM^{v2}$ kháng các tấn công như mô tả trong [18].

8.4.2 Các biến bổ sung và ký hiệu

Đối với bộ tạo khóa dòng $Decim^{v2}$, các ký hiệu sau đây được sử dụng thêm:

a Biến trạng thái trong cho $Decim^{v2}$

$ABSG$	Hàm nén được sử dụng cho $Decim^{v2}$
b, b'	Biến trạng thái trong cho $Decim^{v2}$
B	Hàm đệm được sử dụng cho $Decim^{v2}$
F	Hàm phản hồi tuyến tính cho $Decim^{v2}$
I, I'	Biến trạng thái trong cho $Decim^{v2}$
LF	Hàm lọc được sử dụng cho $Decim^{v2}$
T, T'	Biến trạng thái trong cho $Decim^{v2}$
Y	Hàm Boolean được sử dụng cho $Decim^{v2}$

Ngoài ra, một số ký hiệu được sử dụng cho các biến phụ cục bộ trong mô tả các thuật toán. Những biểu tượng này chỉ xảy ra trong đặc tả hàm đưa ra và không có nghĩa toàn cục. Các biến này được mô tả trong phần khai báo của hàm.

8.4.3 Hàm khởi tạo *Init*

Hàm khởi tạo *Init* được xác định như sau:

Đầu vào: Khóa K độ dài 80-bit và véc tơ khởi tạo IV độ dài 64-bit.

Đầu ra: Giá trị khởi tạo của biến trạng thái $S_0 = (a^{(0)}, T^{(0)}, b^{(0)}, I^{(0)})$.

Biến cục bộ: biến đếm i, j

a) Khởi tạo thanh ghi với khóa K và véc tơ khởi tạo IV.

- Đặt $a_j^{(-256)} = K_j$ với j= 0, 1, 2,..., 79

- Đặt $a_j^{(-256)} = K_{j-80} \oplus IV_{j-80}$ với j= 80, 81,..., 143

- Đặt $a_j^{(-256)} = K_{j-80} \oplus IV_{j-144} \oplus IV_{j-128} \oplus IV_{j-112} \oplus IV_{j-196}$ với j= 144, 145, ...,159

- Đặt $a_j^{(-256)} = K_{j-160} \oplus IV_{j-128} \oplus 1$ với j = 160, 161, ..., 191

b) Khởi tạo bộ đệm và hàm nén:

- Đặt $T^{(-256)} = 000$

- Đặt $b_j^{(-256)} = 0$ với j= 0, 1, 2,...,31

- Đặt $I^{(-256)} = 0$

c) Đặt $S_{-64} = InitNext^{192}(S_{-256}, LFSR)$

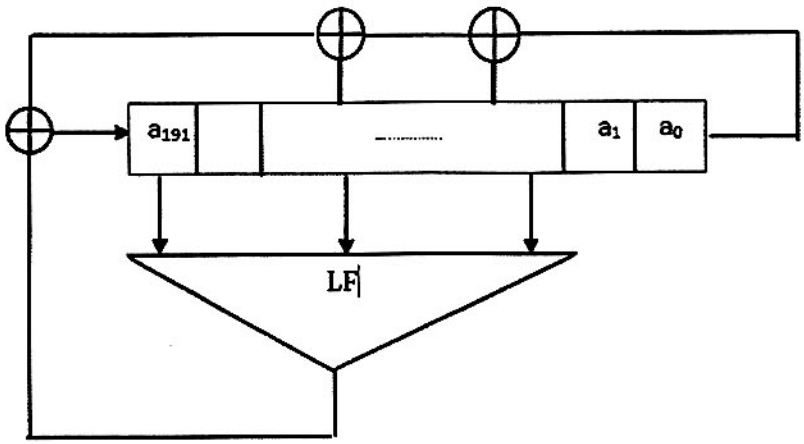
d) Đặt $i = -64$

e) Trong khi $I^{(i)} < 32$ và $i < 0$: Đặt $S_{i+1} = InitNext(S_i, BUFF)$ và $i = i + 1$. Nếu số bước trong hàm *Init* cần cho thực thi được cố định thì có thể việc kiểm tra $I^{(i)} < 32$ có thể được loại bỏ.

f) Đặt $S_0 = S_i$.

g) Đưa ra S_0.

CHÚ THÍCH Các bước d), e) và f) của khởi tạo $DECIM^{v2}$ liên quan đến việc làm đầy bộ đệm trước khi bắt đầu đưa ra khóa dòng. Khi tốc độ đầu ra của hàm nén thay đổi, số lượng các bước yêu cầu để làm đầy bộ đệm có thể thay đổi. Trong bước e) hàm $InitNext(S_i, BUFF)$ được lặp lại nhiều nhất 64 lần, đảm bảo rằng bộ đệm đầy với xác suất lớn hơn $1 - 2^{-97}$. Tính trung bình, bộ đệm đầy sau 24 lần lặp lại.



Hình 11 – Chế độ LFSR của khởi tạo hàm chuyển trạng thái theo *InitNext*

8.4.4 Khởi tạo hàm chuyển trạng thái theo *InitNext*

$Decim^{v2}$ có hai chế độ hoạt động cho hàm *InitNext:* một chế độ được sử dụng trong quá trình khởi tạo thanh ghi A và chế độ thứ 2 trong quá trình khởi tạo làm đầy bộ đệm.

Đầu vào: Biến trạng thái $S_i = (a^{(i)}, T^{(i)}, b^{(i)}, I^{(i)})$, $mode \in \{LFSR, BUFF\}$

Đầu ra: Giá trị tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, T^{(i+1)}, b^{(i+1)}, I^{(i+1)})$

Biến cục bộ: biến đếm j, k, các bộ đệm f_k, r, c, các bộ đệm trạng thái $\alpha^{(0)}, \dots, \alpha^{(4)}, \tau^{(0)}, \dots, \tau^{(4)}, \beta^{(0)}, \dots, \beta^{(4)}, \iota^{(0)}, \dots, \iota^{(4)}$.

Chế độ LFSR (thực thi nếu mode = LFSR):

a) Cập nhật trạng thái của thanh ghi A với các bước sau:

1) Đặt $\alpha^{(0)} = a^{(i)}$

2) Với $k = 0,1,2,3$:

- Đặt $f_k = LF(\alpha^{(k)})$ và $r = L(\alpha^{(k)}) \oplus f_k$.

- Với $j = 0,1,\ldots,190$ đặt $a_j^{(k+1)} = \alpha_{j+1}^{(k)}$

- Đặt $\alpha_{191}^{(k+1)} = r$

3) Đặt $a^{(i+1)} = \alpha^{(4)}$

Hình 11 mô tả sơ đồ khối của chế độ LFSR của hàm *InitNext*.

Chế độ BUFF (thực thi nếu mode = BUFF):

a) Cập nhật trạng thái của thanh ghi A với các bước sau:

4) Đặt $\alpha^{(0)} = a^{(i)}$

5) Với $k = 0,1,2,3$:

- Đặt $f_k = \alpha^{(k)} \oplus LF(\alpha^{(k)})$ và $r = L(\alpha^{(k)})$

- Với $j = 0,1,\ldots,190$ đặt $a_j^{(k+1)} = \alpha_{j+1}^{(k)}$

- Đặt $\alpha_{191}^{(k+1)} = r$

6) Đặt $a^{(i+1)} = \alpha^{(4)}$

b) Đặt $\tau^{(0)} = T^{(i)}, \beta^0 = b^i, \iota^{(0)} = I^{(i)}$

c) Với $k = 0,1,2,3$:

1) Cập nhật trạng thái của khối nén với các bước sau:

- Đặt $c = f_k \oplus \tau_2^{(k)}$

- Đặt $\tau^{(k+1)} = ABSG(\tau^{(k)}, f_k)$

- Nếu $\tau_0^{(k+1)} = 0$, đặt *output* = TRUE, ngược lại đặt *output* = FALSE

2) Cập nhật trạng thái của bộ đệm bởi $(\beta^{(k+1)}, \iota^{(k+1)}) = B(\beta^{(k)}, \iota^{(k)}, \text{đầu ra}, c)$

d) Đặt $T^{(i+1)} = \tau^{(4)}$

e) Đặt $b^{(i+1)} = \beta^{(4)}$ và $I^{(i+1)} = \iota^{(4)}$

8.4.5 Hàm chuyển trạng thái theo *Next*

Đầu vào: Biến trạng thái $S_i = (a^{(i)}, T^{(i)}, b^{(i)}, I^{(i)})$.

Đầu ra: Giá trị tiếp theo của biến trạng thái $S_{i+1} = (a^{(i+1)}, T^{(i+1)}, b^{(i+1)}, I^{(i+1)})$.

Biến cục bộ: biến đếm j, k, các bộ đệm f_k, r, c, các bộ đệm trạng thái $\alpha^{(0)}, \ldots, \alpha^{(4)}, \tau^{(0)}, \ldots, \tau^{(4)}, \beta^{(0)}, \ldots, \beta^{(4)}, \iota^{(0)}, \ldots, \iota^{(4)}$.

a) Cập nhật trạng thái của thanh ghi A với các bước sau:

7) Đặt $a^{(0)} = a^{(i)}$

8) Với $k = 0,1,2,3$:

- Đặt $f_k = a_1^{(k)} \oplus LF(a^{(k)})$ và $r = L(a^{(k)})$

- Với $j = 0,1,\ldots,190$ đặt $a_j^{(k+1)} = a_{j+1}^{(k)}$

- Đặt $a_{191}^{(k+1)} = r$

9) Đặt $a^{(i+1)} = a^{(4)}$

b) Đặt $\tau^{(0)} = T^{(i)}, \beta^0 = b^i, \iota^{(0)} = I^{(i)} - 1$

c) Với $j = 0,1,\ldots,\tau^{(0)} - 1$, đặt $\beta_j^{(0)} = b_{j+1}^{(i)}$

d) Với $k = 0,1,2,3$:

1) nếu $\iota^{(0)} = 0$, đặt $\tau^{(k+1)} = \tau^{(k)}$, *output* = TRUE và $c = f_k$, ngược lại cập nhật trạng thái của khối nén với các bước sau:

- Đặt $c = f_k \oplus \tau_2^{(k)}$

- Đặt $\tau^{(k+1)} = ABSG(\tau^{(k)}, f_k)$

- Nếu $\tau_0^{(k+1)} = 0$, đặt *output* = TRUE, ngược lại đặt *output* = FALSE.

2) Cập nhật trạng thái của bộ đệm bởi $(\beta^{(k+1)}, \iota^{(k+1)}) = B(\beta^{(k)}, \iota^{(k)}, \text{đầu ra}, c)$

e) Đặt $T^{(i+1)} = \tau^{(4)}$, $b^{(i+1)} = \beta^{(4)}$ và $I^{(i+1)} = \iota^{(4)}$

CHÚ THÍCH 1 Điều kiện $\tau^{(0)} = 0$ trong bước 1) của bước d) không bao giờ thỏa mãn: Nếu điều kiện thỏa mãn, điều này có nghĩa bộ đệm trở nên rỗng trong quá trình tạo khóa dòng. Điều này xảy ra với xác suất nhỏ hơn 2^{-80} tại mỗi trạng thái cập nhật, xem chi tiết hơn trong [8]. Ngoài ra, xác suất này cao hơn nếu bộ đệm là không đầy sau hàm khởi tạo *Init*, nhưng, như đã đề cập trong 8.4.3 (CHÚ THÍCH), điều này cũng xảy ra với xác suất không đáng kể.

CHÚ THÍCH 2 Hàm *InitNext* và hàm *Next* cùng chia sẻ nhiều bước tính toán. Thật vậy, chế độ LFSR của hàm *InitNext* chủ yếu bao gồm cập nhật LFSR trong chế độ BUFF và của hàm *Next*, sự khác biệt duy nhất là đầu ra hàm Boolean được thêm bit phản hồi. Chế độ BUFF của hàm *InitNext* và hàm *Next* chỉ khác nhau ở bộ đệm B chỉ được dịch chuyển muộn hơn.

8.4.6 Hàm khóa dòng *Strm*

Đầu vào: Biến trạng thái $S_i = (a^{(i)}, T^{(i)}, b^{(i)}, I^{(i)})$.

Đầu ra: Bit khóa dòng Z_i.

a) Đặt $Z_i = b_0^{(i)}$

b) Đưa ra Z_i

8.4.7 Hàm phản hồi tuyến tính L

Đầu vào: xâu $w = (w_0, w_1, \ldots, w_{191})$ có độ dài 192-bit

Đầu ra: Bit $q = L(w)$

Đặt $q = w_0 \oplus w_3 \oplus w_4 \oplus w_{23} \oplus w_{36} \oplus w_{37} \oplus w_{60} \oplus w_{61} \oplus w_{98} \oplus w_{115} \oplus w_{146} \oplus w_{175} \oplus w_{176} \oplus w_{187}$

8.4.8 Hàm trích lọc LF

Đầu vào: xâu $w = (w_0, w_1, \ldots, w_{191})$ có độ dài 192-bit

Đầu ra: Bit $q = LF(w)$

Đặt $q = L(w)$. Đặt $q = Y((w_{13}, w_{28}, w_{45}, w_{54}, w_{65}, w_{104}, w_{111}, w_{144}, w_{162}, w_{172}, w_{178}, w_{186}, w_{191}))$

8.4.9 Hàm Boolean Y

Đầu vào: xâu $w = (w_0, w_1, \ldots, w_{12})$ có độ dài 13-bit

Đầu ra: Bit $q = Y(w)$

Đặt $q = (\oplus_{0 \le j \le 12} w_j) \oplus (\oplus_{0 \le j \le 12} w_j w_k)$

CHÚ THÍCH Tương ứng, q được cho bởi $q = 0$ nếu $X = 0$ hoặc $X = 3$ và $q = 1$ trong trường hợp ngược lại với $X = w_0 + w_1 + \cdots + w_{12} \ mod \ 4$

8.4.10 Hàm nén $ABSG$

Đầu vào: 3-bit trạng thái T, đưa vào bit c

Đầu ra: 3-bit trạng thái $T' = ABSG(T, c)$

a) Nếu $T_0 = 1$, đặt $T_1 = T_1'$, ngược lại $T_1' = c$.

b) Đặt $T_2' = T_0$ VÀ $(T_1 \oplus c)$

c) Đặt $T_0' = (T_0 \oplus 1)$ $HOẶC$ T_2'

8.4.11 Hàm bộ đệm B

Đầu vào: xâu $b = (b_0, b_1, \ldots, b_{31})$ có độ dài 32-bit, chỉ số I, Boolean $output$, nhập vào bit c.

Đầu ra: xâu $b' = (b_0', b_1', \ldots, b_{31}')$ có độ dài 32-bit, chỉ số I'.

a) Đặt $I' = I$, $b' = b$.

b) Nếu $output = TRUE$, và $I' < 32$, thực hiện như sau:

- Đặt $b'I' = c$.

- Đặt $I' = I' + 1$.

c) Đầu ra $B(b, I, output, c) = (b', I')$

8.5 bộ tạo khóa dòng KCipher-2 (K2)

8.5.1 Giới thiệu KCipher-2 (K2)

KCipher-2 (K2) là bộ tạo khóa dòng trong đó sử dụng đầu vào là khóa bí mật K độ dài 128-bit và véc tơ khởi tạo IV độ dài 128-bit. Khóa bí mật và Véc tơ khởi tạo được sử dụng để khởi tạo biến trạng thái S_i $(i \geq 0)$ bao gồm 20 khối 32-bit, trong đó S_i biểu diễn trạng thái trong của K2 tại chu kỳ i. Thứ tự bit/byte là big-endian, tức là nếu khóa và véc tơ khởi tạo được cho dưới dạng dãy các bit/byte, Thứ nhất/tận cùng bên trái của bit/byte có trọng số cao nhất của dữ liệu tương ứng. Đối với mỗi lần lặp của hàm $Strm$, 64-bit khóa dòng Z_i được tạo coi như đầu ra.

Biến trạng thái S_i của K2 bao gồm 3 thành phần. Thành phần đầu tiên $A^{(i)}$ bao gồm 5 biến 32-bit tuần tự:

$A^{(i)} = (A_4^{(i)}, A_3^{(i)}, A_2^{(i)}, A_1^{(i)}, A_0^{(i)})$ ($A_m^{(i)}$ trên trường $GF(2^{32}), m \geq 0$)

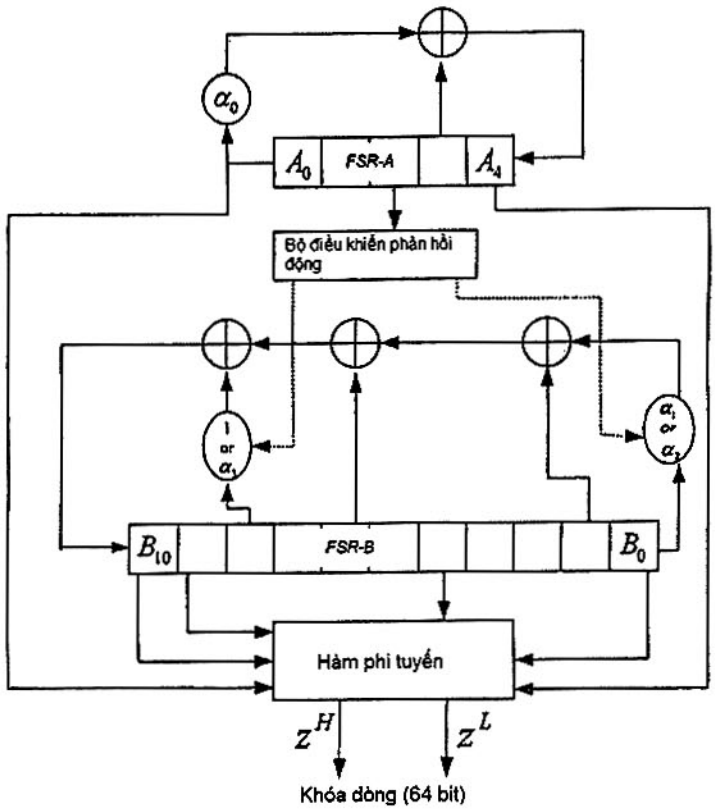
Dạng trạng thái của thanh ghi dịch phản hồi *(FSR) A*. Thành phần thứ hai $B^{(i)}$ bao gồm 11 biến 32-bit tuần tự:

$B^{(i)} = (B_{10}^{(i)}, B_9^{(i)}, \ldots, B_0^{(i)})$ ($B_m^{(i)}$ trên trường $GF(2^{32}), m \geq 0$)

Dạng trạng thái cho thanh ghi dịch phản hồi *(FSR) B*. Thành phần thứ ba bao gồm bộ 4 biến 32-bit:

$R1^{(i)}, L1^{(i)}, R2^{(i)}, L2^{(i)}$, trên trường $GF(2^{32})$

duy trì trạng thái của hàm phi tuyến. Phép toán của K2 được tóm tắt trong Hình 12 và 13, trong đó mô tả một bản chụp các phép toán, tại thời điểm I, bỏ qua biến phụ thuộc thời gian *(i)*.



Hình 12 – Biểu đồ của K2

Hoạt động của K2 được xác định bởi ba hàm sau đây:

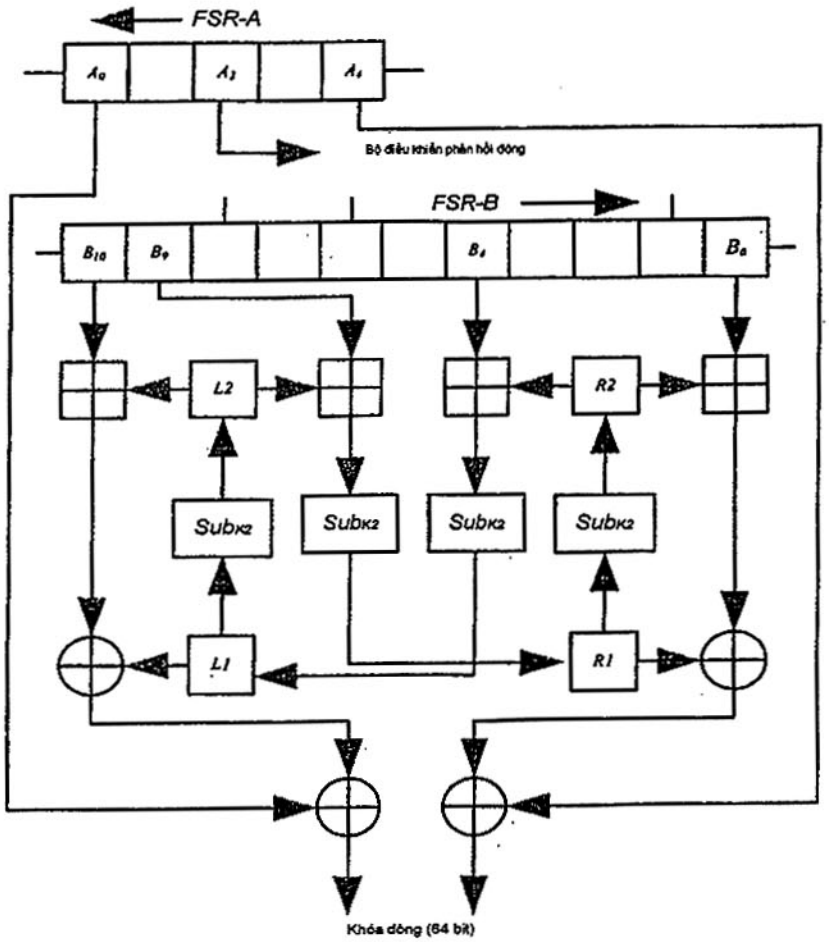
Hàm *Init*, được xác định trong 8.5.2, nhận giá trị đầu vào là khóa K độ dài 128-bit và véc tơ khởi tạo IV độ dài 128-bit để tạo ra trạng thái khởi tạo $S_0 = (A^{(0)}, B^{(0)}, R1^{(0)}, L1^{(0)}, R2^{(0)}, L2^{(0)})$.

Hàm *Next*, được xác định trong 8.5.3, nhận giá trị đầu vào là trạng thái trong $S_i = (A^{(i)}, B^{(i)}, R1^{(i)}, L1^{(i)}, R2^{(i)}, L2^{(i)})$ và tạo như đầu ra giá trị tiếp theo của biến trạng thái $S_{i+1} = (A^{(i+1)}, B^{(i+1)}, R1^{(i+1)}, L1^{(i+1)}, R2^{(i+1)}, L2^{(i+1)})$. Hàm *Next* chạy trong hai chế độ, tùy vào phép lặp thực thi là một phần của khởi tạo hoặc trong chế độ tạo đầu ra thông thường.

Hàm *Strm*, được xác định trong 8.5.4, nhận giá trị đầu vào là trạng thái trong, $S_i = (A^{(i)}, B^{(i)}, R1^{(i)}, L1^{(i)}, R2^{(i)}, L2^{(i)})$ và tạo như đầu ra khóa dòng $Z_i = (Z_i^H + z_i^L)$.

CHÚ THÍCH 1 Số lượng khóa dòng tối đa được khuyến nghị của bit khóa dòng không có hoặc tạo khóa mới hoặc tái khởi tạo với véc tơ khởi tạo IV mới là 2^{64} bit.

CHÚ THÍCH 2 Đối với những cơ sở thiết kế cho K2, tham khảo [12], [13], [14].



Hình 13 – Hàm phi tuyến của K2

8.5.2 Hàm khởi tạo *Init*

Hàm khởi tạo *Init* hoạt động như sau:

Đầu vào: Khóa K độ dài 128-bit và véc tơ khởi tạo IV độ dài 128-bit.

Đầu ra: Giá trị khởi tạo của biến trạng thái $S_0 = (A^{(0)}, B^{(0)}, R1^{(0)}, L1^{(0)}, R2^{(0)}, L2^{(0)})$.

Biến cục bộ: biến đếm m

a) Mở rộng khóa $K = (K_0, K_1, K_2, K_3)$ độ dài 128-bit thành khóa trong $IK = (IK_0, IK_1, \ldots, IK_{11})$ độ dài 384-bit như sau:

1) Với m = 0, 1, 2, 3 đặt $IK_m = K_m$.

2) Với m = 4, 5,...11,

- Nếu $m \neq 4$ hay 8, đặt $IK_m = IK_{m-4} \oplus IK_{m-1}$.

- Nếu $m = 4$, đặt $Rcon[0] = (0x01, 0x00, 0x00, 0x00)$ và $IK_m = IK_{m-4} \oplus Sub_{K2}(IK_{m-1} \lll_{32} 8) \oplus (IK_{m-1} \ggg_{32} 24)) \oplus Rcon[\frac{m}{4} - 1]$. Hàm Sub_{K2} được đề cập đến trong 8.5.5

- Nếu $m = 8$, đặt $Rcon[1] = (0x02, 0x00, 0x00, 0x00)$ và $IK_m = IK_{m-4} \oplus Sub_{K2}(IK_{m-1} \lll_{32} 8) \oplus (IK_{m-1} \ggg_{32} 24)) \oplus Rcon[\frac{m}{4} - 1]$. Hàm Sub_{K2} được đề cập đến trong 8.5.5

b) Khởi tạo các thanh ghi với khóa trong IK và $IV = (IV_0, IV_1, IV_2, IV_3)$.

- Với $m = 0, 1, 2, 3, 4$ đặt $A_m^{(-24)} = IK_{4-m}$.

- Đặt thanh ghi trong *FSR-B* như sau.

$$B_0^{(-24)} = IK_{10},\ B_1^{(-24)} = IK_{11},\ B_2^{(-24)} = IV_0,\ B_3^{(-24)} = IV_1,\ B_4^{(-24)} = IK_8,$$

$$B_5^{(-24)} = IK_9,\ B_6^{(-24)} = IV_2,\ B_7^{(-24)} = IV_3,\ B_8^{(-24)} = IK_7,\ B_9^{(-24)} = IK_5,\ B_{10}^{(-24)} = IK_6$$

- Đặt thanh ghi trong hàm phi tuyến như sau.

$R1^{(-24)}$ =0x00000000, $L1^{(-24)}$ =0x00000000, $R2^{(-24)}$ =0x00000000, $L2^{(-24)}$ =0x00000000

c) Đặt $S_0 = Next^{24}(S_{-24}, INIT)$, trong đó $Next^{24}$ biểu diễn 24 lần lặp của hàm *Next*.

d) Đưa ra S_0

Chung ta tham khảo 8.5.5 cho mô tả hàm Sub_{K2}.

8.5.3 Hàm chuyển trạng thái theo *Next*

K2 có 2 chế độ cho hàm *Next*.

Đầu vào: Biến trạng thái $S_t = (A^{(i)}, B^{(i)}, R1^{(i)}, L1^{(i)}, R2^{(i)}, L2^{(i)})$, mode = {INIT, null}.

Đầu ra: Giá trị tiếp theo của biến trạng thái $S_{i+1} = (A^{(i+1)}, B^{(i+1)}, R1^{(i+1)}, L1^{(i+1)}, R2^{(i+1)}, L2^{(i+1)})$.

Biến cục bộ: biến đếm m

a) Đặt biến trong hàm phi tuyến như sau.

$R1^{(i+1)} = Sub_{K2}\left(L2^{(i)} \oplus_{32} B_9^{(i)}\right)$,

$L1^{(i+1)} = Sub_{K2}\left(R2^{(i)} \oplus_{32} B_4^{(i)}\right)$

$R2^{(i+1)} = Sub_{K2}\left(R1^{(i)}\right)$

$L2^{(i+1)} = Sub_{K2}\left(L1^{(i)}\right)$

b) Với m = 0, 1, 2, 3, đặt $A_m^{(j+1)} = A_{m+1}^{(i)}$.

c) Với $m = 0, 1, 2, \ldots, 9$, đặt $B_m^{(j+1)} = B_{m+1}^{(i)}$.

d) Với chế độ *INIT*, đặt $A_4^{(i+1)} = \left(\alpha_0 \otimes A_0^{(i)}\right) \oplus A_3^{(i)} \oplus NFL(B_0^{(i)}, R2^{(i)}, R1^{(i)}, A_4^{(i)})$

với chế độ null, đặt đặt $A_4^{(i+1)} = \left(\alpha_0 \otimes A_0^{(i)}\right) \oplus A_3^{(i)}$

e) Với chế độ *INIT*, đặt

$$B_{10}^{(i+1)} = \left(\left(\alpha_1^{A_2^{(i)}[30]} + \alpha_2^{1-A_2^{(i)}[30]} - 1\right) \otimes B_0^{(i)}\right) \oplus B_1^{(i)} \oplus B_6^{(i)} \oplus (\alpha_{13}^{A_2^{(i)}[31]} \otimes B_8^{(i)}) \oplus NLF(B_{10}^{(i)}, L2^{(i)}, L1^{(i)}, A_0^{(i)})$$

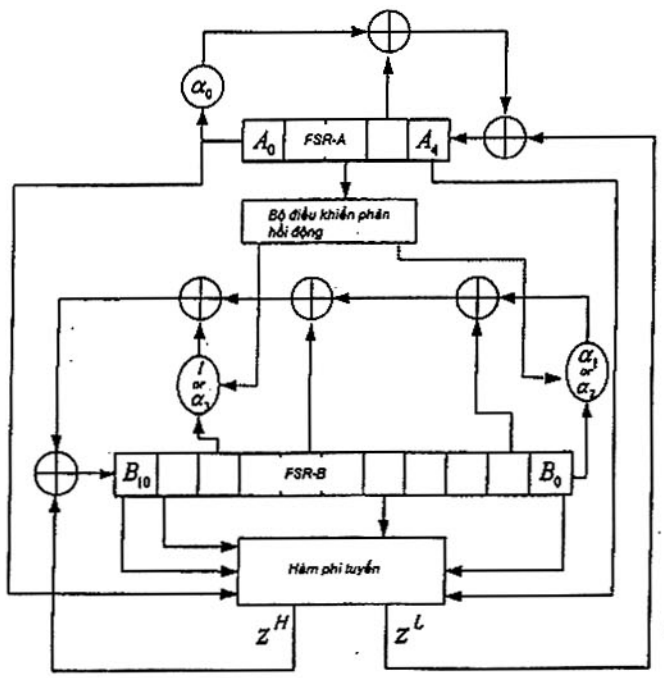
với chế độ null, đặt $B_{10}^{(i+1)} = \left(\left(\alpha_1^{A_2^{(i)}[30]} + \alpha_2^{1-A_2^{(i)}[30]} - 1\right) \otimes B_0^{(i)}\right) \oplus B_1^{(i)} \oplus B_6^{(i)} \oplus (\alpha_{13}^{A_2^{(i)}[31]} \otimes B_8^{(i)})$

f) Đặt $S_{i+1} = (A^{(i+1)}, B^{(i+1)}, R1^{(i+1)}, L1^{(i+1)}, R2^{(i+1)}, L2^{(i+1)})$.

g) Đưa ra S_{i+1}.

$A_m^{(i)}[Y]$ trong {0,1} biểu thị bit thứ Y của thanh ghi $A_m^{(i)}$, trong đó $A_m^{(i)}[31]$ là bit có trọng số cao nhất của $A_m^{(i)}$. Mô tả của hàm Sub_{K2} và số học trường hữu hạn liên quan đến các phần tử cố định $\alpha_0, \alpha_1, \alpha_2$ và α_3 được đề cập tương ứng đến trong 8.5.5, 8.5.6, 8.5.7, 8.5.8 và 8.5.9. Ngoài ra, định nghĩa của hàm *NLF* được đề cập đến trong 8.5.10.

Hình 14 là sơ đồ khối của chế độ *INIT* của hàm *Next*.



Hình 14 – Chế độ *INIT* của hàm *Next*

8.5.4 Hàm khóa dòng *Strm*

Hàm khóa dòng *Strm* hoạt động như sau.

Đầu vào: Biến trạng thái $S_i = (A^{(i)}, B^{(i)}, R1^{(i)}, L1^{(i)}, R2^{(i)}, L2^{(i)})$.

Đầu ra: Khóa dòng $Z_i = (Z_i^H, Z_i^L)$ độ dài 64-bit.

a) Đặt $Z_i^H = NLF(B_{10}^{(i)}, L2^{(i)}, L1^{(i)}, A_0^{(i)})$

b) Đặt $Z_i^L = NLF(B_{10}^{(i)}, R2^{(i)}, R1^{(i)}, A_4^{(i)})$

c) Đặt $Z_i = (Z_i^H, Z_i^L)$

d) Đưa ra Z_i.

Hàm *NLF* được định nghĩa trong 8.5.10.

8.5.5 Hàm Sub_{K2}

Hàm Sub_{K2} là một hoán vị trong trường $GF(2^{32})$, dựa trên các thành phần từ chuẩn mã hóa tiên tiến (AES) [TCVN 11367-3:2016 ISO/IEC 18033-3]. Trong hàm Sub_{K2}, giá trị đầu vào 32-bit được chia thành 4 xâu 1-byte và một hoán vị phi tuyến được áp dụng cho mỗi byte sử dụng hàm thay thế 8x8 bit (*SBox*) tiếp sau một hoán vị tuyến tính 32x32 bit. Các hàm *SBox* là giống như *SBox* của AES, và hoán vị giống như phép toán *Mix Column* của AES.

CHÚ THÍCH 1 Hàm *SBox* AES, *SBox*, có thể được tìm thấy trong 8.1.8 là hàm *SUB*.

CHÚ THÍCH 2 Hàm Sub_{K2} tạo ra đầu ra tương tự như hàm T của 8.2.5.

Đầu vào: Một giá trị w độ dài 32-bit biểu diễn cho một phần tử trên trường $GF(2^{32})$.

Đầu ra: Một xâu $q = Sub_{K2}(w)$ độ dài 32-bit.

Biến cục bộ: biến đếm m.

a) Đặt $w = (w_3, w_2, w_1, w_0)$, trong đó mỗi w_m có độ dài 8 bit.

b) Với mỗi $m = 0, 1, 2, 3$, đặt $t_m = SBox(w_m)$.

c) Đặt $q = (q_3, q_2, q_1, q_0)$ như sau

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \end{pmatrix}$$

Phép nhân các phần tử t_i được thực hiện trong trường $GF(2^8)$ dùng đa thức bất khả quy $f(x) = x^8 + x^4 + x^3 + x + 1$.

d) Đưa ra q.

8.5.6 Phép nhân của α_0 trong trường $GF(2^{32})$

Đầu vào: Một giá trị w độ dài 32-bit, biểu diễn cho 1 phần tử trên trường $GF(2^{32})$.

Đầu ra: Một xâu w' độ dài 32-bit, biểu diễn cho $\alpha_0 \otimes w$ trên trường $GF(2^{32})$.

a) Đặt $w' = (w \ll_{32} 8) \oplus \alpha_{MUL0}[w \gg_{32} 24]$.

b) Đưa ra w'.

Hàm α_{MUL0} được định nghĩa như sau:

$\alpha_{MUL0}[256] = \{$
0x00000000, 0xB6086D1A, 0xAF10DA34, 0x1918E72E, 0x9D207768, 0x2B281A72, 0x3230AD5C,
0x8438C046, 0xF940EED0, 0x4F4883CA, 0x565034E4, 0xE05859FE, 0x646099B8, 0xD268F4A2,
0xCB70438C, 0x7D782E96, 0x31801F63, 0x87887279, 0x9E90C557, 0x2898A84D, 0XACA0680B,
0x1AA30511, 0x03B0B23F, 0xB5E8DF25, 0xC8C0F1B3, 0x7EC89CA9, 0x67D02B87, 0xD1D8469D,
0x55E086DB, 0xE3E8EBC1, 0xFAF05CEF, 0x4CF831F5, 0x62C33EC6, 0xD4CB53DC, 0xCDD3E4F2,
0x7BDB89E8, 0xFFE349AE, 0x49EB24B4, 0x50F3939A, 0xE6FBFE80, 0x9B83D016, 0x2D3BBD0C,
0x34930A22, 0x829B6738, 0x06A3A77E, 0xB0ABCA64, 0xA9B37D4A, 0x1FBB1050, 0x534321A5,
0xE54B4CBF, 0xFC53FB91, 0x4A5B968B, 0xCE6356CD, 0x786B3BD7, 0x61738CF9, 0xD77BE1E3,
0xAA03CF75, 0x1C0BA26F, 0x05131541, 0xB31B785B, 0x3723B81D, 0x812BD507, 0x98336229,
0x2E3B0F33, 0xC4457C4F, 0x724D1155, 0x6B55A67B, 0xDD5DCB61, 0x59650827, 0xEF6D663D,
0xF675D113, 0x407DBC09, 0x3D05929F, 0x8B0DFF85, 0x921548AB, 0x241D25B1, 0xA025E5F7,
0x162D88ED, 0x0F353FC3, 0xB93D52D9, 0xF5C5632C, 0x43CD0E36, 0x5AD5B918, 0xECDDD402,
0x68E51444, 0xDEED795E, 0xC7F5CE70, 0x71FDA36A, 0x0C858DFC, 0xBA8DE0E6, 0xA39557C8,
0x159D3AD2, 0X91A5FA94, 0x27AD978E, 0x3EB520A0, 0x88BD4DBA, 0xA6864289, 0x108E2F93,
0x099698BD, 0xBF9EF5A7, 0x3BA635E1, 0x8DAE58FB, 0x94B6EFD5, 0x22BE82CF, 0x5FC6AC59,
0xE9CEC143, 0xF0D6766D, 0x46DE1B77, 0xC2E6DB31, 0x74EEB62B, 0x6DF60105, 0xDBFE6C1F,
0x97065DEA, 0x210E30F0, 0x381687DE, 0x8E1EEAC4, 0x0A262A82, 0xBC2E4798, 0xA536F0B6,
0x133E9DAC, 0x6E46B33A, 0xD84EDE20, 0xC150690E, 0x775E0414, 0xF366C452, 0x456EA948,
0x5C761E66, 0xEA7E737C, 0x4B8AF89E, 0xFD829584, 0xE49A22AA, 0x52924FB0, 0xD6AA8FF6,
0x60A2E2EC, 0x79BA55C2, 0xCFB238D8, 0xB2CA164E, 0x04C27B54, 0x1DDACC7A, 0xABD2A16D,
0x2FEA6126, 0x99E20C3C, 0x80FABB12, 0x36F2D608, 0x7A0AE7FD, 0xCC028AE7, 0xD51A3DC9,
0x631250D3, 0xE72A9095, 0x5122FD8F, 0x483A4AA1, 0xFE3227BB, 0x834A092D, 0x35426437,
0x2C5AD319, 0x9A52BE03, 0x1E6A7E45, 0xA862135F, 0xB17AA471, 0x0772C96B, 0x2949C658,
0x9F41AB42, 0x86591C6C, 0x30517176, 0xB469B130, 0x0261DC2A, 0x1B796B04, 0xAD71061E,
0xD0092888, 0x66014592, 0x7F19F2BC, 0xC9119FA6, 0x4D295FE0, 0xFB2132FA, 0xE23985D4,
0x5431E8CE, 0x18C9D93B, 0xAEC1B421, 0xB7D9030F, 0x01D16E15, 0x85E9AE53, 0x33E1C349,
0x2AF97467, 0x9CF1197D, 0xE18937EB, 0x57815AF1, 0x4E99EDDF, 0xF89180C5, 0x7CA94083,
0xCAA12D99, 0xD3B99AB7, 0x65B1F7AD, 0x8FCF84D1, 0x39C7E9CB, 0x20DF5EE5, 0x96D733FF,
0x12EFF3B9, 0xA4E79EA3, 0xBDFF298D, 0x0BF74497, 0x768F6A01, 0xC087071B, 0xD99FB035,
0X6F97DD2F, 0xEBAF1D69, 0x5DA77073, 0x44BFC75D, 0xF2B7AA47, 0xBE4F9BB2, 0x0847F6A8,
0x115F4186, 0xA7572C9C, 0x236FECDA, 0x95678IC0, 0x8C7F36EE, 0x3A775BF4, 0x470F7562,
0xF1071878, 0xE81FAF56, 0x5E17C24C, 0xDA2F020A, 0x6C276F10, 0x753FD83E, 0xC337B524,
0xED0CBA17, 0x5B04D70D, 0x421C6023, 0xF4140D39, 0x702CCD7F, 0xC624A065, 0xDF3C174B,
0x69347A51, 0x144C54C7, 0xA24439DD, 0xBB5C8EF3, 0x0D54E3E9, 0x896C23AF, 0x3F644EB5,
0x267CF99B, 0x90749481, 0xDC8CA574, 0x6A84C80E, 0x739C7F40, 0xC594125A, 0x41ACD21C,
0xF7A4EF06, 0xEEBC0828, 0x58346532, 0x25CC4BA4, 0x93C426BE, 0x8ADC9190, 0x3CD4FC8A,
0xB8EC3CCC, 0x0EE451D6, 0x17FCE6F8, 0xA1F48EE2};

8.5.7 Phép nhân của α_1 trong trường $GF(2^{32})$

Đầu vào: Một giá trị w độ dài 32-bit, biểu diễn cho 1 phần tử trên trường $GF(2^{32})$.

Đầu ra: Một xâu w' độ dài 32-bit, biểu diễn $\alpha_1 \otimes w$ trên trường $GF(2^{32})$.

a) Đặt $w' = (w \ll_{32} 8) \oplus \alpha_{MUL1}[w \gg_{32} 24]$.

b) Đưa ra w'.

Hàm α_{MUL1} được định nghĩa như sau:

α_{MUL1} [256] = {

0x00000000, 0xA0F5FC2E, 0x6DC7D55C, 0xCD322972, 0xDAA387B8, 0x7A567B96, 0xB76452E4,
0x1791AECA, 0x996B235D, 0x399EDF73, 0xF4ACF601, 0x54590A2F, 0x43C8A4E5, 0xE33D58CB,
0x2E0F71B9, 0x8EFA8D97, 0x1FD646BA, 0xBF23BA94, 0x721193E6, 0xD2E46FC8, 0xC575C102,
0x65803D2C, 0xA8B2145E, 0x0847E870, 0x86BD65E7, 0x264899C9, 0xEB7AB03B, 0x4B8F4C95,
0x5C1EE25F, 0xFCEB1E71, 0x31D93703, 0x912CCB2D, 0x3E818C59, 0x9E747077, 0x53465905,
0xF3B3A52B, 0xE4220BE1, 0x44D7F7CF, 0x89E5DEBD, 0x29102293, 0xA7EAAF04, 0x071F532A,
0xCA2D7A58, 0x6AD88676, 0x7D4928BC, 0xDDBCD492, 0x10SEFDE0, 0xB07B01CE, 0x2157CAE3,
0x81A236CD, 0x4C901FBF, 0xEC65E391, 0xFBF44D5B, 0x5B01B175, 0x96339807, 0x36C66429,
0xB83CE9BE, 0x18C91590, 0xD5FB3CE2, 0x750EC0CC, 0x629F6E06, 0xC26A9228, 0x0F58BB5A,
0xAFAD4774, 0x7C2F35B2, 0xDCDAC99C, 0x11E8E0EE, 0xB11D1CC0, 0xA68CB20A, 0x06794E24,
0xCB4B6756, 0x6BBE9B78, 0xE54416EF, 0x45B1EAC1, 0x8883C3B3, 0x28763F9D, 0x3FE79157,
0x9F126D79, 05220440B, 0xF2D5B825, 0x63F97308, 0xC30C8F26, 0x0E3EA654, 0xAECB5A7A,
0xB95AF4B0, 0x19AF089E, 0xD49D21EC, 0x7468DDC2, 0xFA925055, 0x5A67AC7B, 0x97558509,
0x37A07927, 0x2031D7ED, 0x80C42BC3, 0x4DF602B1, 0xED03FE9F, 0x42AEB9EB, 0xE25B45C5,
0x2F696CB7, 0x8F9C9099, 0x980D3E53, 0x38F8C27D, 0xF5CAEB0F, 0x553F1721, 0xDBC59AB6,
0x7B306698, 0xB6024FEA, 0x16F7B3C4, 0x01661D0E, 0xA193E120, 0x6CA1C852r0xCC54347C,
0x5D78FF51, 0xFD8D037F, 0x30BF2A0D, 0x9C4AD623, 0x87DB78E9, 0x272E84C7, 0xEA1CADB5,
0x4AE9519B, 0xC413DC0C, 0x64E62022, 0xA9D40950, 0x0921F57E, 0x1EB05BB4, 0xBE45A79A,
0x73778EE8, 0xD38272C6, 0xF85E6A49, 0x58AB9667, 0x9599BF15, 0x356C433B, 0x22FDEDF1,
0x820811DF, 0x4F3A38AD, 0xEFCFC483, 0x61354914, 0xC1C0B53A, 0x0CF29C48, 0xAC076066,
0xBB96CEAC, 0x1B633282, 0xD6511BF0, 0x76A4E7DE, 0xE7882CF3, 0x477DD0DD, 0x8A4FF9AF,
0x2ABA0581, 0x3D2BAB4B, 0x9DDE5765, 0x50EC7E17, 0xF0198239, 0x7EE30FAE, 0xDE16F330,
0x1324DAF2, 0xB3D126DC, 0xA4408816, 0x04B57438, 0xC9875D4A, 0x6972A164, 0xC6DFE610,
0x662A1A3E, 0xAB18334C, 0x0BEDCF62, 0x1C7C61A8, 0xBC899D86, 0x71BBB4F4, 0xD14E48DA,
0x5FS4C54D, 0xFF413963, 0x32731011, 0x9286EC3F, 0x851742F5, 0x25E2BEDB, 0xE8D097A9,
0x48256B87, 0xD909A0AA, 0x79FC5C84, 0xB4CE75F6, 0x143B89D8, 0x03AA2712, 0xA35FDB3C,
0x6E6DF24E, 0xCE980E60, 0x406283F7, 0xE0977FD9, 0x2DA556AB, 0x8D50AA85, 0x9AC1044F,
0x3A34F861, 0xF706D113, 0x57F32D3D, 0x84715FFB, 0x2484A3D5, 0xE9B68AA7, 0x49437689,
0x5ED2D843, 0xFE27246D, 0x33150DIF, 0x93E0F131, 0x1D1A7CA6, 0xBDEF8088, 0x70DDA9FA,
0xD02855D4, 0xC7B9FB1E, 0x674C0730, 0xAA7E2E42, 0x0A8BD26C, 0x9BA71941, 0x3B52E56F,
0XF660CCID, 0x56953033, 0x41049EF9, 0xE1F162D7, 0x2CC34BA5, 0x8C36B78B, 0x02CC3A1C,
0xA239C632, 0x6F0BEF40, 0xCFFE136E, 0xD86FBDA4, 0x789A418A, 0xB5A868F8, 0x155D94D6,
0xBAF0D3A2, 0x1A052F8C, 0xD73706FE, 0x77C2FAD0, 0x6053541A, 0xC0A6A834, 0x0D948146,
0xAD617D68, 0x239BF0FF, 0x836E0CD1, 0x4E5C25A3, 0xEEA9D98D, 0xF9387747, 0x59CD8B69,
0x94FFA2IB, 0x340A5E35, 0xA5269518, 0x05036936, 0xC8E14044, 0x6814BC6A, 0x7F8512A0,
0xDF70EE8E, 0x1242C7FC, 0xB2B73BD2, 0x3C4DB645, 0x9CB84A6B, 0x518A6319, 0xF17F9F37,

0xE6EE31FD, 0x461BCDD3, 0x8B29E4A1, 0x2BDC188F}

8.5.8 Phép nhân của α_2 trong trường $GF(2^{32})$

Đầu vào: Một giá trị w độ dài 32-bit, biểu diễn cho 1 phần tử trên trường $GF(2^{32})$.

Đầu ra: Một xâu w' độ dài 32-bit, biểu diễn cho $\alpha_2 \otimes w$ trên trường $GF(2^{32})$.

a) Đặt $w' = (w \ll_{32} 8) \oplus \alpha_{MUL2}[w \gg_{32} 24]$.

b) Đưa ra w'.

Hàm α_{MUL2} được định nghĩa như sau:

α_{MUL2} [256] = {

0x00000000, 0x5BF87F93, 0xB6BDFS6B, 0xED4581F8, 0x2137B1D6, 0x7ACFCE45, 0x978A4FBD,
0XCC72302E, 0x426E2FE1, 0x19965072, 0xF4D3D18A, 0xAF2BAE19, 0x63599E37, 0x38A1E1A4,
0xD5E4605C, 0x8E1C1FCF, 0x84DC5E8F, 0xDF24211C, 0x3261A0E4, 0x6999DF77, 0xA5EBEF59,
0xFE1390CA, 0x13561132, 0x48AE6EA1, 0xC6B2716E, 0x9D4A0EFD, 0x700F8F05, 0x2BF7F096,
0xE785C0B8, 0xBC7DBF2B, 0x51383ED3, 0x0AC04140, 0x45F5BC53, 0x1E0DC3C0, 0xF3484238,
0xA8B03DAB, 0x64C20D85, 0x3F3A7216, 0xD27FF3EE, 0x89878C7D, 0x079B93B2, 0x5C63EC21,
0xB1266DD9, 0xEADE124A, 0x26AC2264, 0x7D545DF7, 0x9011DC0F, 0xCBE9A39C, 0xC129E2DC,
0x9AD19D4F, 0X77941CB7, 0x2C6C6324, 0xE01E530A, 0xBBE62C99, 0x56A3AD61, 0x0D5BD2F2,
0x8347CD3D, 0xD8BFB2AE, 0x35FA3356, 0x6E024CC5, 0xA2707CEB, 0xF9880378, 0x14CD8280,
0x4F35FD13, 0x8AA735A6, 0xD15F4A35, 0x3C1ACBCD, 0x67E2B45E, 0xAB908470, 0xF068FBE3,
0x1D2D7A1B, 0x46D50588, 0xC8C91A47, 0x933165D4, 0x7E74E42C, 0x258C9BBF, 0xE9FEAB91,
0xB206D402, 0x5F4355FA, 0x04BB2A69, 0x0E7B6B29, 0x558314BA, 0xB8C69542, 0xE33EEAD1,
0x2F4CDAFF, 0x74B4A56C, 0x99F12494, 0xC2095B07, 0x4C1544C8, 0x17ED3B5B, 0xF^1A8BAA,
0xA150C530, 0x6D22F51E, 0x36DA8A8D, 0xDB9F0B75, 0x806774E6, 0xCF5289F5, 0x94AAF666,
0x79EF779E, 0x2217080D, 0xEE653823, 0xB59D47B0, 0x58D8C648, 0x0320B9DB, 0x8D3CA614,
0xD6C4D987, 0x3B81587F, 0x607927EC, 0xAC0B17C2, 0xF7F36851, 0x1AB6E9A9, 0x414E963A,
0x4B8ED77A, 0x1076A8E9, 0xFD332911, 0xA6CB5682, 0x6AB966AC, 0x3141193F, 0xDC0498C7,
0x87FCE754, 0x09E0F89B, 0x52188708, 0xBF5D06F0, 0xE4A57963, 0x28D7494D, 0x732F36DE,
0x9E6AB726, 0xC592C8B5, 0x59036A01, 0x02FB1592, 0xEFBE946A, 0xB446EBF9, 0x7834DBD7,
0x23CCA444, 0xCE8925BC, 0x95715A2F, 0x1B6D45E0, 0x40953A73, 0xADD0BB8B, 0xF628C418,
0x3A5AF436, 0x61A28BA5, 0x8CE70A5D, 0xD71F75CE, 0xDDDF348E, 0x86274B1D, 0x6B62CAE5,
0x309AB576, 0xFCE88558, 0xA710FACB, 0x4A557B33, 0x11AD04A0, 0x9FB11B6F, 0xC44964FC,
0x290CE504, 0x72F49A97, 0xBE86AAB9, 0xE57ED52A, 0x083B54D2, 0x53C32B41, 0x1CF6D652,
0x470EA9C1, 0xAA4B2839, 0xF1B357AA, 0x3DC16784, 0x66391817, 0x8B7C99EF, 0xD084E67C,
0x5E98F9B3, 0x05608620, 0xE82507D8, 0xB3DD784B, 0x7FAF4865, 0x245737F6, 0xC912B60E,
0x92EAC99D, 0x982A88DD, 0xC3D2F74E, 0x2E9776B6, 0x756F0925, 0xB91D390B, 0xE2E54698,
0x0FA0C760, 0x54S8B8F3, 0xDA44A73C, 0x81BCD8AF, 0x6CF95957, 0x370126C4, 0xFB7316EA,
0xA08B6979, 0x4DCEE881, 0x16369712, 0xD3A45FA7, 0x885C2034, 0x6519A1CC, 0x3EE1DE5F,
0XF293EE71, 0xA96B91E2, 0x442E101A, 0x1FD66F89, 0x91CA7046, 0xCA320FD5, 0x27778E2D,
0x7C8FF1BE, 0xB0FDC190, 0xEB05BE03, 0x06403FFB, 0x5DB84068, 0x57780128, 0x0C807EBB,
0xE1C5FF43, CxBA3D80D0, 0x764FB0FE, 0x2DB7CF6D, 0xC0F24E95, 0x960A3106, 0x15162EC9,

0x4EEE515A, 0xA3ABD0A2, 0xF853AF31, 0x34219F1F, 0x6FD9E08C, 0x829C6174, 0xD9641EE7,
0x9651E3F4, 0xCDA99C67, 0x20EC1D9F, 0x7B14620C, 0xB7665222, 0XEC9E2DB1, 0x01DBAC49,
0x5A23D3DA, 0xD43FCC15, 0x8FC7B386, 0x6282327E, 0x397A4DED, 0xF5087DC3, 0xAEF00250,
0x433583A8, 0x184DFC3B, 0x128DBD7B, 0x4975C2E8, 0xA4304310, 0xFFC83C83, 0x33BA0CAD,
0x6842733E, 0x8507F2C6, 0xDEFF8D55, 0x50E3929A, 0x0B1BED09, 0xE65E6CF1, 0xBDA61362,
0x71D4234C, 0x2A2C5CDF, 0xC769DD27, 0x9C91A2B4};

8.5.9 Phép nhân của α_3 trong trường $GF(2^{32})$

Đầu vào: Một giá trị w độ dài 32-bit, biểu diễn cho 1 phần tử trên trường $GF(2^{32})$.

Đầu ra: Một xâu w' độ dài 32-bit, biểu diễn cho $\alpha_3 \otimes w$ trên trường $GF(2^{32})$.

a) Đặt $w' = (w \ll_{32} 8) \oplus \alpha_{MUL3}[w \gg_{32} 24]$.

b) Đưa ra w'.

Hàm α_{MUL3} được định nghĩa như sau:

α_{MUL3} [256] = {

0x00000000, 0X4559568B, 0x8AB2AC73, 0xCFEBFAF8, 0x71013DE6, 0x34586B6D, 0xFBB39195,
0xBEEAC7IE, 0xE2027AA9, 0xA75B2C22, 0x68B0D6DA, 0x2DE98Q51, 0x9303474F, 0xD65A11C4,
0x19B1EB3C, 0x5CE8BDB7, 0xA104F437, 0xE45DA2BC, 0x2BB65844, 0x6EEF0ECF, 0xD005C9D1,
0x955C9F5A, 0x5AB765A2, 0x1FEE3329, 0x43068E9E, 0x065FD815, 0xC9B422ED, 0x8CED7466,
0x32073378, 0x775EE5F3, 0xB8B51F0B, 0xFDEC4980, 0x27088D6E, 0x6251DBE5, 0xADBA211D,
0xE8E37796, 0x5609B088, 0x135CE603, 0xDCBB1CFB, 0x99E24A70, 0xC50AF7C7, 0x8053A14C,
0x4FB85BB4, 0x0AE10D3F, 0xB40BCA21, 0xF1529CAA, 0x3EB96652, 0x7BE030D9, 0x860C7959,
0xC3552FD2, 0x0CBED52A, 0x49E783A1, 0xF70D44BF, 0xB2541234, 0x7DBFE8CC, 0x38E6BE47,
0x640E03F0, 0x2157557B, 0xEEBCAF83, 0xABE5F908, 0x150F3E16, 0x5056689D, 0x9FBD9265,
0xDAE4C4EE, 0x4E107FDC, 0x0B492957, 0xC4A2D3AF, 0x81FB8524, 0x3F11423A, 0x7A4814B1,
0xB5A3EE49, 0xF0FAB8C2, 0xAC120575, 0xE94B53FE, 0x26A0A906, 0x63F9FF8D, 0xDD133893,
0x984A6E18, 0x57A194E0, 0x12F8C26B, 0xEF148BEB, 0xAA4DDD60, 0x65A62798, 0x20FF7113,
0x9E15B60D, 0xDB4CE086, 0x14A71A7E, 0x51FE4CF5, 0x0D16F142, 0x484FA7C9, 0x87A45D31,
0xC2FD0BBA, 0x7C17CCA4, 0x394E9A2F, 0xF6A560D7, 0xB3FC365C, 0x6918F2B2, 0x2C41A439,
0XE3AA5EC1, 0xA6F3084A, 0x1819CF54, 0x5D4099DF, 0x92A36327, 0xD7F235AC, 0x8B1A881B,
0xCE43DE90, 0x01A82468, 0x44F172E3, 0xFA1BB5FD, 0xBF42E376, 0x70A9198E, 0x35F04F05,
0XC81C0685, 0x8D45500E, 0x42AEAAF6, 0x07F7FC7D, 0xB91D3B63, 0xFC446DE8, 0x33AF9710,
0x76F6C19B, 0x2A1E7C2C, 0x6F472AA7, 0xA0ACD0SF, 0xE5F586D4, 0x.5B1F41CA, 0x1E46174,
0xD1ADEDB9, 0x94F4BB32, 0x9C20FEDD, 0xD979A856, 0x169252AE, 0x53CB0425, 0xED21C33B,
0XA87895B0, 0x67936F48, 0x22CA39C3, 0x7E228474, 0x3B7BD2FF, 0xF4902807, 0xB1C97E8C,
0x0F23B992, 0x4A7AEF19, 0x859115E1, 0xC0C8436A, 0x3D240AEA, 0x787D5C61, 0xB796A699,
0xF2CFF012, 0x4C25370C, 0x097C6187, 0xC6979B7F, 0x83CECDF4, 0xDF267043, 0x9A7F26C8,
0x5594DC30, 0x10CD8ABB, 0xAE274DA5, 0xEB7E1B2E, 0x2495E1D6, 0x61CCB75D, 0xBB2873B3,
0xFE712538, 0x319ADFC0, 0x74C3894B, 0xCA294E55, 0x8F7018DE, 0x409BE226, 0x05C2B4AD,

```
0x592A091A,0x1C735F91,0xD398A569,0x96C1F3E2,0x282B34FC,0x6D726277,0xA299988F,
0xE7C0CE04,0x1A2C8784,0x5F75D10F,0x909E2BF7,0xD5C77D7C,0x6B2DBA62,0x2E74ECE9,
0XE19F1611,0xA4C6409A,0xF82EFD2D,0xBD77ABA6,0x729C515E,0x37C507D5,0x892FC0CB,
0xCC769640,0x039D6CB8,0x46C43A33,0xD2308101,0x9769D78A,0x58822D72,0x1DDB7BF9,
0xA331BCE7,0xE668EA6C,0x29831094,0x6CDA461F,0x3032FBA8,0x756BAD23,0xBA8057DB,
0xFFD90150,0x4133C64E,0x046A90C5,0xCB816A3D,0x8ED83CB6,0x73347536,0x366D23BD,
0xF986D945,0xBCDF8FCE,0x023548D0,0x476C1E5B,0x8887E4A3,0xCDDEB228,0x91360F9F,
0xD46F5914,0x1B84A3EC,0x5EDDF567,0xE0373279,0xA56E64F2,0x6A859E0A,0x2FDCC881,
0xF5380C6F,0xB0615AE4,0x7F8AA01C,0x3AD3F697,0x84393189,0xC1606702,0x0E8B9DFA,
0x4BD2CB71,0x173A76C6,0x5263204D,0x9D88DAB5,0xD8D18C3E,0x663B4B20,0x23621DAB,
0XEC89E753,0xA9D0B1D8,0x543CF858,0x1165AED3,0xDE8E542B,0x9BD702A0,0x253DC5BE,
0x60649335,0xAF8F69CD,0xEAD63F46,0xB63E82F1,0xF367D47A,0x3C8C2E82,0x79D57809,
0xC73FBF17,0x3266E99C,0x4D8D1364,0x08D445EF};
```

8.5.10 Hàm $NLF(a, b, c, d)$

Đầu vào: 4 giá trị độ dài 32-bit a, b, c và d.

Đầu ra: Một xâu q độ dài 32-bit

a) Đặt $q = (a +_{32} b) \oplus c \oplus d)$.

b) Đưa ra q.

Phụ lục A

(Quy định)

Định danh đối tượng

Phụ lục này liệt kê các định danh đối tượng gán cho các thuật toán được đặc tả trong bộ TCVN 11367 (ISO/IEC 18033) và xác định các cấu trúc tham số thuật toán. Vui lòng tham khảo tiêu chuẩn TCVN 11367-3:2016 (ISO/IEC 18033-3) về các ID của đối tượng cho các chế độ hoạt động của mã khối.

```
EncryptionAlgorithms-4 {
   iso(1) standard(0) encryption-algorithms(18033) part(4) asnl-
      rnodule(0) algorithm-object-identifiers(0) }
   DEFINITIONS EXPLICIT TAGS ::= BEGIN

-- EXPORTS All/ --

-- IMPORTS None; --

01D OBJECT IDENTIFIER -- Alias

-- Synonyms --

isl8033-4 OID ::= { iso(1) standard(0) isl8033(18033) part4(4) }
id-kg OID ::= { isl8033-4 keystream-generator(1) } id-
scmode OID ::= { isl8033-4 stream-cipher-mode(2) }

-- Assignments --

id-kg-mugi OID ::= { id-kg mugi(1) }
id-kg-snow OID ::= { id-kg snow(2) }
id-kg-rabbit OID ::= { id-kg rabbit(3)
}
id-kg-decim2 OID ::= { id-kg decim2(4) }
id-kg-k2 0ID'::= { id-kg k2(5) }

id-scmode-additive OID ::= { id-scmode additive(1) }
id-scmode-multisOl OID ::= { id-scmode multisOl(2) }

-- Algorithms and parameters --
StreamCipher ::= Algorithmldentifier {{ StreamCipherAlgorithms }}

StreamCipherAlgorithms ALGORITHM ::= {
   aaditiveStreamCipher 1
   multiSOlStreamCipher,

   ... -- Expect additional algorithms --
}
additiveStreamCipher ALGORITHM {
   OID id-scmode-additive PARMS AdditiveStreamCipherParameters
}

AdditiveStreamCipherParameters ::= KeyGenerator

multiSOlScreamCipher ALGORITHM {
   OID id-scmode-multisOl PARMS MultiSOlStreamCipherParameters


      iMultiSOlStreamCipherParameters ::= SEQUENCE { keyGenerator
         KeyGenerator, securityParameter INTEGER DEFAULT 64,
         irreduciblePolynoial BIT STRING, redandancy BIT STRING,
         publicParameterR BIT STRING
            -  length determined by securityParameter -- for full interoperability
            multisOl parameters should -- include the padding method but they do not
```

```
            have object -- identifiers, for the time being they will have to be --
            negotiated in an application-dependent way

        }

KeyGenerator ALGORITHM ::= { mugiKeyGenerator
|  snowKeyGenerator  |  rabbitKeyGenerator  |
decim2KeyGenerator | k2KeyGenerator,

... -- Expect additional algorithms --

}

mugiKeyGenerator ALGORITHM ::= {
OID id-kg-mugi PARMS NullParameters
}

snowKeyGenerator ALGORITHM {
OID id-kg-snow PARMS NullParameters
}

rabbitKeyGenerator ALGORITHM ::={
OID id-kg-rabbit PARMS NullParameters

}

decim2KeyGenerator ALGORITHM ::={
OID id-kg-decim2 PARMS NullParameters

}

k2KeyGenerator ALGORITHM ::={
OID id-kg-k2 PARMS NullParameters
}


NullParameters ::= NULL

-- Cryptographic algorithm identification --

ALGORITHM ::= CLASS {
&id OBJECT IDENTIFIER UNIQUE,
&Type OPTIONAL

}
WITH SYNTAX { OID &id [PARMS &Type] }

AlgorithmIdentifier { ALGORITHM:IOSet } ::= SEQUENCE { algorithm
ALGORITHM.&id( {IOSet} ),
   parameters ALGORITHM.&Type( {IOSet}{Salgorithm} ) OPTIONAL

}

END -- EncryptionAlgorithms-4 --
```

Phụ lục B

(Tham khảo)

Các phép toán trên trường hữu hạn $GF(2^n)$

Đối với bất kỳ số nguyên dương n tồn tại một trường hữu hạn gồm có 2^n phần tử. Trường hữu hạn này duy nhất trên phép đẳng cấu và trong tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033) nó được gọi là trường hữu hạn $GF(2^n)$.

Trong mô tả đa thức, mỗi phần tử của $GF(2^n)$ được mô tả bởi một đa thức nhị phân có bậc nhỏ hơn n. Một cách rõ ràng hơn, xâu bit $a = a_{n-1} \dots, a_2 a_1 a_0$ được thực hiện để đại diện cho đa thức nhị phân $a(x) = a_{n-1}x^{n-1} + \cdots + a_2x^2 + a_1x + a_0$. Cơ sở của đa thức là tập hợp $B = (x^{n-1}, \dots, x^2, x, 1)$. Đối với 2 xâu bit $a = a_{n-1} \dots, a_2 a_1 a_0$ và $b = b_{n-1} \dots b_2 b_1 b_0$, tổng là $c = a \oplus b = c_{n-1} \dots c_2 c_1 c_0$, trong đó $c_i = a_i \oplus b_i$.

Phép nhân trong trường hữu hạn, được viết là $a \otimes b$ tương ứng với phép nhân hai đa thức $a(x)b(x)$ modulo với một đa thức bất khả quy nhị phân $p(x)$ bậc n. Một đa thức là bất khả quy nếu nó không có ước số tầm thường.

$GF(2^n)\backslash\{0\}$ ký hiệu là $GF(2^n)^*$ là một nhóm abel đối với phép nhân và phần tử đơn vị là 1. Đối với bất kỳ đa thức nhị phân khác không $b(x)$ bậc nhỏ hơn n, các nghịch đảo của $b(x)$ ký hiệu là $b^{-1}(x)$, có thể được tính như sau: sử dụng thuật toán Euclid mở rộng để tính toán đa thức $a(x)$ và $c(x)$ sao cho $b(x).a(x) + p(x).c(x) = 1$. Do đó, $a(x).b(x) \; mod \; p(x) = 1$, có nghĩa là $b^{-1}(x) = a(x) \; mod \; p(x)$. Thuật toán Euclid mở rộng được mô tả trong [15].

Phụ Lục C

(Tham khảo)

Các ví dụ

C.1 Ví dụ cho MUGI

C.1.1 Khóa, véc tơ khởi tạo và bộ ba khóa dòng

K = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

z = c7 6e 14 e7 08 36 e6 b6 cb 0e 9c 5a 0b f0 3e 1e 0a cf 9a f4 9e be 6d 67 d5 72 6e 37 4b 13 97 ac.

K = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 34 61 69 88 51 81 21 39 01 55 00 a5 3b 7e 59 87

z = 2a a1 c5 c7 20 73 b1 b3 a9 d1 0d c6 85 50 66 10 28 30 56 0d 9a 24 65 c9 9c 29 1c 13 81 4e 08 8d

K = 51 34 00 b1 04 a0 59 91 30 ad 00 fc 48 d7 59 e0

IV= 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

z = bd df ad 5f 04 b8 86 25 c3 ad ac e1 56 d1 c1 99 36 ff a4 e9 a7 fd f7 5a aa b8 29 13 42 85 aa 4b

K = 69 e1 06 ee 52 95 37 2c 75 13 01 47 30 23 79 93

IV= 2a 00 45 c8 49 27 49 d5 3a 9b 16 4a 25 e4 49 15

z = e3 cc 67 a0 25 5b 0f 28 2d 9a 5b 1b bd f7 f2 df 84 eb 46 f6 07 d6 e6 dd 32 86 13 43 94 dd 95 fb.

C.1.2 Ví dụ các trạng thái trong

K = 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f

IV= f0 e0 d0 c0 b0 a0 90 80 70 60 50 40 30 20 10 00

z = be 62 43 06 14 b7 9b 71 71 a6 66 81 c3 55 42 de 7a ba 5b 4f b8 0e 82 d7 0b 96 98 28 90 b6 e1 43

Intermediate values of the internal state

rho function 0

a: 0001020304050607 08090a0b0c0d0e0f 7498f5f1e727d094

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000

rho function 1

a: 08090a0b0c0d0e0f 9724d9144c5d8926 64b47311d52100a5

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 08090a0b0c0d0e0f

rho function 2

a: 9724d9144c5d8926 09671cfbcfaa95fb e2J38166cd8c441

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 3

a: 09671cfbcfaa95fb 9c0c2097edb20067 6ef29c62b7691210

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 4

a: 9c0c2097edb20067 c08ee4dcb2d08591 201239b2b04d5d6a

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 06090a0b0c0d0e0f

rho function 5

a: c08ee4dcb2d08591 738177859f3210f6 48S63357b89312eb

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 0000000000000000 0000000000000000 c08ee4dcb2d08591
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 6

a: 738177859f3210f6 b30b4d944f5d04cb bc7ac7e83f40ccal

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 738177859f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 03090a0b0c0d0e0f

rho function 7

a: b36b4d944f5d04cb 2d13c00221057d8d 65e12d98fb29feca

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 b36b4d944f5d04cb 738177B59f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 8

a: 2d13c00221057d8d 20ead0479e63cdc3 7169edbc504968d2

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 0000000000000000 0000000000000000

2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 9

a: 20ead0479e63cdc3 591a6857e3112cee 8269181ee80366a1

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 0000000000000000 20ead0479e63cdc3

2d13c00221057c8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 10

a: 591a6857e3112cee dfbbb88c02c9c80a fa312d220ef73c78

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000

0000000000000000 0000000000000000 591a6857e3112cee 20ead0479e63cdc3

2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 11

a: dfbbb88c02c9c80a 5cc4835080bc5321 78e69bd217041ca7

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
0000000000000000 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 9c0c2097edb20067
09671cfbcfaa95fb 9724d9144c5d8926 c08se4dcb2d08591 08090a0b0c0d0e0f

rho function 12

a: 5cc4835080bc5321 fd5755df9cc0ceb9 dd032b76f3534504

b: 0000000000000000 0000000000000000 0000000000000000 0000000000000000
5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 13

a: fd5755df9cc0ceb9 c905d08f50fa71db cfcb255e594b38ee

b: 0000000000000000 0000000000000000 0000000000000000 fd5755df9cc0ceb9
5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 14

a: c905d08f50fa71db bfe2485ac2696cc7 0a77652c7dbcc580

b: 0000000000000000 0000000000000000 c905d08f50fa71db fd5755df9cc0ceb9
5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 15

a: bfe2485ac2696cc7 7dea261cb61d4fea 3991ce48e105a4a1

b: 0000000000000000 bfe2485ac2696cc7 c905d08f50fa71db fd5755df9cc0ceb9
5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591
9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

```
buffer init
a: 7dea261cb61d4fea eafb528479bb687d eb8189612089ff0b
b: 7dea261cb61d4fea bfe2485ac2696cc7 c905d08f50fa71db fd5755df9cc0ceb9
   5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3
   2d13c00221057d8d b36b4d944f5d04cb 738177859f3210f6 c08ee4dcb2d08591
   9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

rho function 0
a: 8d0af6dc06bddf6a 9a9b02c4499b787d f100cffe031d365b

rho function 1
a: 9a9b02c4499b787d 435407f3bbc2c760 b8576326c43c7141

rho function 2
a: 435407f3bbc2c760 b5117172dcf5e507 10d44d672b0cb32b

rho function 3
a: b5117172dcf5e507 9157292760b2892f 45de3e448a22a274

rho function 4
a: 9157292760b2892f aee0542493e7889e d92646e5bf6e90fd

rho function 5
a: aee0542493e7889e a9f2f7fac6cff1ff 668ac5cf634db73d

rho function 6
a: a9f2f7fac6cff1ff 9cb8969f9fc84dc6 d3db5a83153c2d75

rho function 7
a: 9cb8969f9fc84dc6 b1260b2ec980a340 4c06fba0602d20da
```

rho function 8

a: b1260b2ec980a340 152a6fd877969848 4e9d5f1Of22daa44

rho function 9

a: 192a6fd877969348 bbac287d38601209 c31e21b47993441d

rho function 10

a: bbac237d38601209 d58486545129be34 88b995cf25723d71

rho function 11

a: d58486545129be34 c8af8f1422e98119 7cb36f5145a5f171

rho function 12

a: c8af8f1422e98119 00bba312081aa445 2e8517e066c8b410

rho function 13

a: 00bba312081aa445 2f3864a9c27Sal4c 4elbalaafcO6cb55

rho function 14

a: 2f3664a9c279a14c 6551f5e9cbcle0d7 acf8aaa64583d0d7

rho function 15

a: 6551f5e9cbcle0d7 4e466dffcb92db48 4a8ffe073636f5c3

state init

a: 4e466dffcb92db48 f5eb67b928359d8b 5d3c31a0af9cd78f

b: 7dea261cb61d4fea bfe2485ac2696cc7 c905d08f50fa71db fd5755df9cc0ceb9

5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee 20ead0479e63cdc3

2d13c00221057d8d b36b4d944f5d04cb 738177659f3210f6 c08ee4dcb2d08591

9c0c2097edb20067 09671cfbcfaa95fb 9724d9144c5d8926 08090a0b0c0d0e0f

update 1

a: f5eb67b928 359d8b ace6a90bde0af786 529108c358fa4ada

```
b: 464f67f4c75fd547 7dea261cb61d4fea bfe2485ac2696cc7 c905d08f50fa71db
   ddbd859802a3037a 5cc4835080bc5321 dfbbb88c02c9c80a 591a6857e3112cee
   20ead0479e63cdc3 2d13c00221057d8d 7ccld86f463al830 738177859f3210f6
   c08ee4dcb2d08591 9c0c2097edb2C067 09671cfbcfaa95fb 9724d9144c5d8926

update 2
a: ace6a90bde0af786 9fa7al5367ael667 5f241cf311a0bfa7
b: 62cfbead646814ad 464f67f4c79fd547 7dea261cb61d4fea bfe2485ac2696cc7
   901fb8d8b3eb5d35 ddbd859802a3037a 5cc4835080bc5321 dfbbb88c02c9c80a
   591a6857e3112cee 20ead0479e63cdc3 c0alc065bd095dla 7ccld86f463al830
   738177859f3210f6 c08ee4dcb2d08591 9c0c2097edb20067 09671cfbcfaa95fb

update 3
a: 9fa7al5367ael667 75195c2e249e4399 8bd43dd671ad8b05
b: a581b5f011a0627d 62cfbead646814ad 464f67f4c79fd547 7dea261cb61d4fea
   5cc4835080bc5321 dfbbb88c02c9c80a c62878al90905b6b 923a55d65eed291f
   c0alc065bd095dla 7ccld86f463al830 738177859f3210f6 c08ee4dcb2d08591

update 5
a: 9b2239a28cc5a4e3 80cld0bcl8b29e62 4b4f363e4a322d0e
b: b597b8f2964ec608 03ab81c48alcl600 a581b5f011a0627d 62cfbead646814ad
   9bf2e26cc53cd63d 212ea54c36allccb 6059f0d6c0a0a4cd 901fb8d8b3eb5d35
   ddbd859802a3037a 5cc4835080bc5321 998Ia0bc7e081065 c62878al90905b6b
   923a55d65eed291f c0alc065bd095dla 7ccld86f463al830 738177859f3210f6

update 6
a: 80cld0bcl8b29e62 dfe2225186dfbca3 1277ae469887f627
b: e8a34e2713f7b415 b597b8f2964ec608 03ab81c48alcl600 a581b5f01ia0627d
   f2d00675d7834998 9bf2e26cc53cd63d 212ea54c36allccb 6059f0d6c0a0a4cd
   901fb8d8b3eb5d35 ddbd859802a3037a elcdde4a401d9344 998Ia0bc7e081065
   c62878al90905b6b 923a55d65eed291f c0alc065ba095dla 7ccld86f463al830

update 7
```

a: dfe2225186dfbca3 ca86cdc9d2bf2007 7a5c92de7be1811f

b: fc0008d35e888652 e8a34e2713f7b415 b597b8f2964ec608 03ab81c48a1c1600
c5d84526d100c6b0 f2d00675d7834998 9bf2e26cc53cd63d 212ea54c3oa11ccb
6059f0d6c0a0a4cd 901fb8d8b3eb5d35 8350ac87909956ac e1cdde4a401d9344
9981a0bc7e081065 c62878a190905b6b 923a55d65eed291f c0a1c065bd095d1a

update 8

a: ca86cdc9d2bf2007 0870fbf8e065b266 067f3c2be88481d4

b: 1f43e2343bd6e1b9 fc0008d35e888652 e8a34e2713f7b415 b597b8f2964ec608
22852488bcbd0acb c5d84526c100c6b0 f2d00675d7834998 9bf2e26cc53cd63d
212ea54c36a11ccb 6059f0d6c0a0a4cd 008fe3b375c32594 8350ac87909956ac
e1cdde4a401d9344 9981a0bc7e081065 c62878a190905b6b 923a55d65eed291f

update 9

a: 0870fbf8e065b266 73b145404394710d d756724ed3994273

b: 58bc981f8c520918 1f43e2343bd6e1b9 fc0008d35e888652 e8a34e2713f7b415
2e655a9e53721035 22852488bcbd0acb c5d84526d100c6b0 f2d0C675d7834998
9bf2e26cc53cd63d 212ea54c36a11ccb Ie51e0b359210471 008fe3b375c32594
8350ac8790S956ac e1cdde4a401d9344 9981a0bc7e081065 c62878a190905b6b

update 10

a: 73b145404394710d 82d164adcac96d62 0607785b7d152b8b

b: ce58835970f5e90d 58bc98If8c520918 1f43e2343bd6e1b9 fc0008d35e888652
1a734852c474fd8d 2e655a9e53721035 22852488bcbd0acb c5d84526d100c6b0
f2d00675d7834998 9bf2e26cc53cd63d 61333608d76cc281 Ie51e0b359210471
008fe3b375c32594 8350ac879C9956ac e1cdde4a401dS344 9981a0bc7e081065

update 11

a: 82d164adcac96d62 c14072735c68e7e9 f61c61bbde49ed28

b: ea30e5fc3d9c6168 ce58835970f5e90d 58bc981f8c520918 1f43e2343bd6e1b9
39d84df58f8840e2 1a734852c474fd8d 2e655a9e53721035 22852488bcbd0acb
c5d84526d100c6b0 f2d00675d7834998 0b6bb4c0466c7aba 61333608d76cc281
1e51e0b359210471 008fe3b375c32594 8350ac87909956ac e1cdde4a401d9344

update 12

a: c14072735c68e7e9 ce0bee4623950852 af052447a7444e65

b: 631cbae78ad4fe26 ea30e5fc3d9c6168 ce58835970f5e90d 58bc98If8c520918
3dc6c6bc876beb72 39d84df58f8840e2 1a734852c474fd8d 2e655a9e53721035
22852488bcbd0acb c5d64526d100c6b0 871323e1d70caa2b 0b6bb4c0466c7aba
61333608d76cc281 1e51e0b359210471 008fe3b375c32594 8350ac87909956ac

update 13

a: ce0bee4623950852 f6c22506fc93fb5a 9eb296971244bcb3

b: 4210def4ccf1b145 631cbae78ad4fe26 ea30e5fc3d9c6168 ce58335970f5e90d
76d9c281df20192d 3dc6c6bc876beb72 39d84df58f8840e2 1a734852c474fd8d
2e655a9e53721035 22852488bcbd0acb 9cf94157cf512603 871323e1d70caa2b
0b6bb4c0466c7aba 61333608d76cc281 1e51e0b359210471 008fe3b375c32594

update 14

a: f6c22506fc93fb5a b36f504b7eb67fe6 a66ba7dd058722d3

b: ce840df556562dc6 4210def4ccf1b145 631cbae78ad4fe26 ea30e5fc3d9c6168
d42bcb0bb4911480 76d9c281df20192d 3dc6c6bc876beb72 39d84df58f8340e2
1a734852c474fd8d 2e655a9e53721035 f5e9e609dd8e3cc3 9cf94157cf512603
871323e1d70caa2b 0b6bb4c0466c7aba 61333608d76cc281 1e51e0b359210471

update 15

a: b36f504b7eb57fe5 0ce5a4d1a0cbc0f7 bd0c30563f8ee4f7

b: e893c5b5a5b2ff2b ce840df556562dc6 4210def4ccf1b145 631cbae78ad4fe26
d3e8a809b214218a d42bcb0bb4811480 76d9c281df20192d 3dc6c6bc876beb72
39d84df58f8840e2 Ia734852c474fd8d 680920245819a4f5 f5e9e609dd8e3cc3
9cf94157cf512603 871323e1d70caa2b 0b6bb4c0466c7aba 61333608d76cc281

update 16

a: 0ce5a4d1a0cbc0f7 316S93816117e50f bc62430614b79b71

b: d25c6£43a9dabd67 e893c5b5a5b2ff2b ce840df556562dc6 4210def4ccf1b145
5eda7c5b0dbf1554 d3e8a809b214218a d42bcb0bb4811480 76d9c281df20192d

3dc6c6bc876beb72 39d84df58f8840e2 cd7fe2794367de6c 680920245819a4f5

f5e9e609dd8e3cc3 9cf94157cf512603 871323eld70caa2b 0b6bb4c0466c7aba

update 1

a: 316993816117e50f 4f7c747ce422e686 71a66681c35542de

b: 078el011e6a7ba4d d25c6643a9dabd67 e893c5b5a5b2ff2b ce840df556562dc6

34c91c7513dla868 5eda7c5b0dbf1554 d3e8a809b214218a d42bcb0bb4811480

76d9c281df20192d 3dc6c6bc876beb72 f6896bf6137101b5 cd7fe2794367de6c

680920245819a4f5 f5e9e609dd8e3cc3 9cf94157cf512603 871323eld70caa2b

update 2

a: 4f7c747ce422e686 0aeabSf525cla62f 7aba5b4fb80e82d7

b: b67ab060b61b4f24 078el011e6a7ba4d d25c6643a9dabd67 e893c5b5a5b2ff2b

1aafc6fee2d73946 34c91c7513dla868 5eda7c5b0dbf1554 d3e8a809b214218a

d42bcb0bb4811480 76d9c281df20192d e048fa7f72820d7b f6896bf6137101b5

cd7fe2794367de6c 680920245819a4f5 f5e9e609ddee3cc3 9cf94157cf512603

update 3

a: 0aeab5f525cla62f bdla2938a57319c8 0b96982890b6el43

b: d385352b2b73c085 b67ab060b61b4f24 078el011e6a7ba4d d25c6643a9dabd67

3b7b6dbcl7a6deal 1aafc6fee2d73946 34c91c7513dla868 5eda7c5b0dbf1554

d3e8a809b214218a d42bcb0bb4eil480 2ec06674b7293909 e048fa7f72820d7b

f6896bf6137101b5 cd7fe2794367de6c 680920245819a4f5 f5e9e609dd8e3cc3

update 4

a: bdla2938a57319c8 e4684a2bf28ff50d 4930b5d033157f46

b: ff0353fcf84f9aec d385352b2b73c085 b67ab060b61b4f24 078el011e6a7ba4d

8c861al8a465a833 3b7b6dbcl7a6deal 1aafc6fee2d73946 34c91c7513dla868

5eda7c5b0dbf1554 d3e8a809b214218a 974cl56779fef6f9 2ec06674b7293909

e048fa7f72820d7b f6896bf6137101b5 cd7fe2794367de6c 680920245819a4f5

C.2 Ví dụ khóa 128 bit cho SNOW 2.0

C.2.1 Khóa, véc tơ khởi tạo và bộ ba khóa dòng

$(IV_3, IV_2, IV_1, IV_0) = (0, 0, 0, 0)$, key = 80000000000000000000000000000000

Keystream output: 8D590AE9A74A7D056DC9CA74B72D1A4599B0A083FB45D13FCF9411BD9A503783 .

(IV_3, IV_2, IV_1, IV_0) = (0, 0, 0, 0), key = AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Keystream output: E00982F525F02054214992D8706F2B20DA585ESB85E2746D09F22681B2749407.

(IV_3, IV_2, IV_1, IV_0) = (4,3,2,1), key = 60000000000000000000000000000000

Keystream output: D6403358E035 4A6957F43FCE44B4B13FF78E24C246618A0767AC83C10BFC45F0.

(IV_3, IV_2, IV_1, IV_0) = (4,3,2,1), key = AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Keystream output: C355385DB31D6CBDF774AF5366C2E8774DEADAC7DC7229DFED171D7B.

C.2.2 Ví dụ các trạng thái trong

K = 80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 00 00 00 04 00 00 00 03 00 00 00 02 00 00 00 01

z = d6 40 33 58 e0 35 4a 69 57 f4 3f ce 44 b4 b1 3f f7 8e 24 c2 46 61 8a 07 67 ac 83 c1 0b fc 45 f0

Snow 2.0 Internal state at time -34

15:80000000 14:00000000 13:00000000 12:00000000 11:7fffffff 10:ffffffff 09:ffffffff 08:ffffffff
07:80000000 06:00000000 05:00000000 04:00000000 03:7fffffff 02:ffffffff 01:ffffffff 00:ffffffff

Snow 2.0 Internal state at time -33

15:80000001 14:00000000 13:00000000 12:00000002 11:7fffffff 10:fffffffc 09:fffffffb 08:ffffffff
07:80000000 06:00000000 05:00000000 04:00000000 03:7fffffff 02:ffffffff 01:ffffffff 00:ffffffff
R1:00000000 R2:00000000

Snow 2.0 Internal state at time -32

15:09dfef08 14:80000001 13:00000000 12:00000000 11:00000002 10:7fffffff 09:fffffffc 08:fffffffb
07:ffffffff 06:80000000 05:00000000 04:00000000 03:00000000 02:7fffffff 01:ffffffff 00:ffffffff
R1:00000000 R2:63636363

Snow 2.0 Internal state at time -31

15:e5f1b94c 14:09dfef08 13:80000001 12:00000000 11:00000000 10:00000002 09:7fffffff 08:fffffffc

07:fffffffb 06:ffffffff 05:80000000 04:00000000 03:00000000 02:00000000 01:7fffffff 00:ffffffff

R1:63636363 R2:63636363

Snow 2.0 Internal state at time -30

15:ea9a3527 14:e5f1b94c 13:09dfef08 12:80000001 11:00000000 10:00000000 09:00000002 08:7fffffff

07:fffffffc 06:fffffffb 05:ffffffff 04:80000000 03:00000000 02:00000000 01:00000000 00:7fffffff

R1:e3636363 R2:fbfbfbfb

Snow 2.0 Internal state at time -29

15:a69fa10d 14:ea9a3527 13:e5f1b94c 12:09dfef08 11:80000001 10:00000000 09:00000000 08:00000002

07:7fffffff 06:fffffffc 05:fffffffb 04:ffffffff 03:80000000 02:00000000 01:00000000 00:00000000

R1:fbfbfbfa R2:34dellll

Snow 2.0 Internal state at time -28

15:8ecaccdb 14:a69fa10d 13:ea9a3527 12:e5f1b94c 11:09dfef08 10:80000001 09:00000000 08:00000000

07:00000002 06:7fffffff 05:fffffffc 04:fffffffb 03:ffffffff 02:80000000 01:00000000 00:00000000

R1:34dell0c R2:692d2d4b

```
Snow 2.0 Internal state at time -27

15:eaf4d48f 14:8ecaccdb 13:a69fal0d 12:ea9a3527 11:e5flb94c 10:09dfef08 09:80000001 08:00000000

07:00000000 06:00000002 05:7fffffff 04:fffffffc 03:fffffffb 02:ffffffff 01:80000000 00:00000000

Rl:692d2d47 R2:b66ede7f

Snow 2.0 Internal state at time -26

15:19305681 14:eaf4d48f 13:8ecaccdb 12:a69fal0d 11:ea9a3527 10:e5flb94c 09:09dfef08 08:80000001
07:00000000 06:00000000 05:00000002 04:7fffffff 03:fffffffc 02:fffffffb 01:ffffffff 00:80000000
Rl:366ede7e R2:12c38109

Snow 2.0 Internal state at time -25

15:e8688f55 14:19805681 13:eaf4d48f 12:8ecaccdb 11:a69fal0d 10:ea9a3527 09:e5flb94c 08:09dfef08
07:80000001 06:00000000 05:00000000 04:00000002 03:7fffffff 02:fffffffc 01:fffffffb 00:ffffffff

Rl:12c3810b R2:86c47640

Snow 2.0 Internal state at time -24

15:fa565efl 14:e8688f55 13:19805681 12:eaf4d48f 11:8ecaccdb 10:a69fal0d 09:ea9a3527 08:e5flb94c
07:09dfef08 06:80000001 05:00000000 04:00000000 03:00000002 02:7fffffff 01:fffffffc 00:fffffffb
Rl:86c47640 R2:d63b88a5

Snow 2.0 Internal state at time -23

15:6c7a94aa 14:fa565efl 13:e8688f55 12:19805681 11:eaf4d48f 10:8ecaccdb 09:a69fal0d 08:ea9a3527
07;e5flb94c 06:09dfef08 05:80000001 04:00000000 03:00000000 02:00000002 01:7fffffff 00:fffffffc
Rl:d63b88a5 R2:b7c51902

Snow 2.0 Internal state at time -22

15:5ced2805 14:6c7a94aa 13:fa565efl 12:e8688f55 11:19805681 10:eaf4d48f 09:8ecaccdb 08:a69fal0d
07:ea9a3527 06:e5flb94c 05:09dfef08 04:80000001 03:00000000 02:00000000 01:00000002 00:7fffffff
Rl:37c51903 R2:db1c5e4f

Snow 2.0 Internal state at time -21

15:26acle81 14:5ced2805 13:6c7a94aa 12:fa565efl 11:e8688f55 10:19805681 09:eaf4d48f 08:8ecaccdb
07:a69fal0d 06:ea9a3527 05:e5flb94c 04:09dfef08 03:80000001 02:00000000 01:00000000 00:00000002
Rl:e4fc4d57 R2:d04da3ad

Snow 2.0 Internal state at time -20

15:2e5ccla4 14:26acle81 13:5ced2805 12:6c7a94aa 11:fa565efl 10:e8688f55 09:19805681 08:eaf4d48f
07:8ecaccdb 06:a69fal0d 05:ea9a3527 04:e5flb94c 03:09dfef08 02:80000001 01:00000000 00:00000000
Rl:b63f5cf9 R2:6c782451

Snow 2.0 Internal state at time -19

15:2eb71be8 14:2e5ccla4 13:26acle31 12:5ced2805 11:6c7a94aa 10:fa565efl 09:e8688f55 08:19805681
07:eaf4d48f 06:8ecaccdb 05:a69fal0d 04:ea9a3527 03:e5flb94c 02:09dfef08 01:80000001 00:00000000
Rl:57125978 R2:13ebdccc

Snow 2.0 Internal state at time -18

15:dc33fa8c 14:2eb71be8 13:2e5ccla4 12:26acle81 11:5ced2805 10:6c7a94aa 09:fa565efl 08:e8683f55

07:19805681 06:eaf4d48f 05:8ecaccdb 04:a69fal0d 03:ea9a3527 02:e5flb94c 01:09dfef08 00:80000001

Rl:ba8b7dd9 R2:6b132ab7

Snow 2.0 Internal state at time -17

15:3007668a 14:dc33fa8c 13:2eb71be8 12:2e5ccla4 11:26acle81 10:5ced2805 09:6c7a94aa 08:fa565efl

07:e8688f55 06:19805681 05:eaf4d48f 04:8ecaccdb 03:a69fal0d 02:ea9a3527 01:e5flb94c 00:09dfef08

Rl:f9ddf792 R2:6eb763b9

Snow 2.0 Internal state at time -16
```

15:6fbb£cfb 14:3007668a 13:dc33fa8c 12:2eb71be8 11:2e5ccla4 10:26acle81 09:5ced2805 08:6c7a94aa
07:fa565efl 06:e8688f55 05:19805681 04:eaf4d48f 03:8ecaccdb 02:a69fal0d 01:ea9a3527 00:e5flb94c
Rl:59ac3848 R2:510e5e7e

Snow 2.0 Internal state at time -15

15:47128118 14:6fbbfcfb 13:3007668a 12:dc33fa8c 11:2eb71be8 10:2e5ccla4 09:26acle81 08:5ced2805

07:6c7a94aa 06:fa565efl 05:e8688f55 04:19805681 03:eaf4d48f 02:8ecaccdb 01:a69fal0d 00:ea9a3527
Rl:6a8eb4ff R2:ed2a3ff7

Snow 2.0 Internal state at time -14

15:9dd8c346 14:47128118 13:6fbbfcfb 12:3007668a 11:dc33fa8c 10:2eb71be8 09:2e5ccla4 08:26acle81
07:5ced2805 06:6c7a94aa 05:fa565efl 04:e8688f55 03:19805681 02:eaf4a48f 01:8ecaccdb 00:a69fal0d
Rl:d592cf4c R2:aaaf3ebb

Snow 2.0 Internal state at time -13

15:14c6e4bl 14:9dd8c346 13:47128118 12:6fbbfcfb 11:3007668a 10:dc33fa8c 09:2eb71be8 08:2e5ccla4
07:26acle81 06:5ced2805 05:6c7a94aa 04:fa565efl 03:e8688f55 02:19805681 01:eaf4d48f 00:8ecaccdb
Rl:a5059dac R2:b838f49b

Snow 2.0 Internal state at time -12

15:2bel899a 14:14c6e4bl 13:9dd8c346 12:47128118 11:6fbbfcfb 10:3007668a 09:dc33fa8c 08:2eb71be8
07:2e5ccla4 06:26acle81 05:5ced2805 04:6c7a94aa 03:fa565efl 02:e8688f55 01:19805681 00:eaf4d48f
Rl:24b38945 R2:911396b6

Snow 2.0 Internal state at time -11

15:09363669 14:2bel899a 13:14c6e4bl 12:9dd8c346 11:47128115 10:6fbbfcfb 09:3007668a 08:dc33fa8c
07:2eb71be8 06:2e5ccla4 05:26acle81 04:5ced2805 03:6c7a94aa 02:fa565efl 01:e8688f55 00:19805681
Rl:ee00bebb R2:1449ba75

Snow 2.0 Internal state at time -10

15:9e6f24ce 14:09363669 13:2bel899a 12:14c6e4bl 11:9dd8c346 10:47128118 09:6fbbfcfb 08:3007668a
07:dc33fa8c 06:2eb71be8 05:2e5ccla4 04:26acle81 03:5ced2805 02:6c7a94aa 01:fa565efl 00:e8686f55
Rl:3af5d8f6 R2:b8fa206d

Snow 2.0 Internal state at time -9

15:f87d0a5a 14:9e6f24ce 13:09363669 12:2bel899a 11:14c6e4bl 10:9dd8c346 09:47123118 08:6fbbfcfb
07:3007668a 06:dc33fa8c 05:2eb71be3 04:2e5ccla4 03:26acle81 02:5ced2B05 01:6c7a94aa 00:fa565efl
Rl:e756e211 R2:5a6f3141

Snow 2.0 Internal state at time -8

15:056b74c0 14:f87d0a5a 13:9e6f24ce 12:09363669 11:2bel899a 10:14c6e4bl 09:9da8c346 08:47128118
07:6fbbfcfb 06:3007668a 05:dc33fa8c 04:2eb71be8 03:2e5ccla4 02:26acle81 01:5ced2805 00:6c7a94aa
Rl:89264d29 R2:87c4f589

Snow 2.0 Internal state at time -7

15:62d49257 14:056b74c0 13:f87d0a5a 12:9e6f24ce 11:09363669 10:2bel899a 09:14c6e4bl 08:9dd8c346
07:47128118 06:6fbbfcfb 05:3007668a 04:dc33fa8c 03:2eb71be8 02:2e5ccla4 01:26acle81 00:5ced2805
Rl:63f8f015 R2:b541dd3f

```
Snow 2.0 Internal state at time -6

15:4f549ab4 14:82d49257 13:056b74c0 12:f87d0a5a 11:9e6f24ce 10:09363669 09:2be1899a 08:14c6e4b1
07:9dd8c346 06:47128118 05:6fbbfcfb 04:3007668a 03:dc33fa8c 02:2eb71be8 01:2e5cc1a4 00:26ac1e81
R1:e54943c9 R2:cb416287


Snow 2.0 Internal state at time -5

15:01d936bb 14:4f549ab4 13:82d49257 12:056b74c0 11:f87d0a5a 10:9e6f24ce 09:09363669 08:2be1899a
07:14c6e4b1 06:9dd8c346 05:47128118 04:6fbbfcfb 03:3007668a 02:dc33fa8c 01:2eb71be8 00:2e5cc1a4
R1:3afd5f82 R2:f4c17d6d


Snow 2.0 Internal state at time -4

15:29c99eb5 14:01d936bb 13:4f549ab4 12:82d49257 11:056b74c0 10:f87d0a5a 09:9e6f24ce 08:09363669
07:2be1899a 06:14c6e4b1 05:9dd8c346 04:47128118 03:6fbbfcfb 02:3007668a 01:dc33fa8c 00:2eb71be8
R1:3bd3fe85 R2:b5efeab8


Snow 2.0 Internal state at time -3

15:0ea4e3f0 14:99c99eb5 13:01d936bb 12:4f549ab4 11:82d49257 10:056b74c0 09:f87d0a5a 08:9e6f24ce

07:09363669 06:2be1899a 05:14c0e4b1 04:9dd8c346 03:47128118 02:0fbbfcfb 01:3007668a 00:dc33fa8c

R1:53c8adfe R2:a0ddb267


Snow 2.0 Internal state at time -2

15:fa000bc8 14:0ea4e3f0 13:99c99eb5 12:01d936bb 11:4f549ab4 10:82d49257 09:056b74c0 08:f87d0a5a

07:9e6f24ce 06:09363669 05:2be1899a 04:14c6e4b1 03:9dd8c346 02:47128118 01:6fbbfcfb 00:3007668a

R1:b5a49718 R2:6ac944cc


Snow 2.0 Internal state at time -1

15:61dec1b8 14:fa000bc8 13:0ea4e3f0 12:99c99eb5 11:01d936bb 10:4f549ab4 09:82d49257 08:056b74c0
07:f87d0a5a 06:9e6f24ce 05:09363669 04:2be1899a 03:14c6e4b1 02:9dd8c346 01:47128118 00:6fbbfcfb
R1:96aace66 R2:9cd3a85e


Snow 2.0 Internal state at time 0

15:31f914d5 14:61dec1b8 13:fa000bc8 12:0ea4e3f0 11:99c99eb5 10:01d936bb 09:4f549ab4 08:82d49257
07:056b74c0 06:f87d0a5a 05:9e6f24ce 04:09363669 03:2be1899a 02:14c6e4b1 01:9dd8c346 00:47128118
R1:a609dec7 R2:495041dc


Snow 2.0 Internal state at time 1

15:9098ec10 14:31f914d5 13:61dec1b8 12:fa000bc8 11:0ea4e3f0 10:99c99eb5 09:01d936bb 08:4f549ab4
07:82d49257 06:056b74c0 05:f87d0a5a 04:9e6f24ce 03:09363669 02:2be1899a 01:14c6e4b1 00:9dd8c346
R1:e7bf66aa R2:05b5db95


Snow 2.0 Internal state at time 2

15:a5e7b806 14:9098ec10 13:31f914d5 12:61dec1b8 11:fa000bc8 10:0ea4e3f0 09:99c99eb5 08:01dS36bb
07:4f545ab4 06:82d49257 05:056b74c0 04:f87d0a5a 03:9e6f24ce 02:09363669 01:2be1899a 00:14c6s4b1
R1:fe32e5ef R2:e728468a
```

Snow 2.0 Internal state at time 3

15:962fd59e 14:a5e7b800 13:9098ec10 12:31f914d5 11:61dec1b8 10:fa000bc8 09:0ea4e3f0 08:99cS9eb5

07:01d936bb 06:4f549ab4 05:82d49257 04:056b74c0 03:f87d0a5a 02:9e6f24ce 01:09363669 00:2be1899a

R1:ec93bb4a R2:ed96a84d

Snow 2.0 Internal state at time 4

15:be037f87 14:962fd59e 13:a5e7b806 12:9098ec10 11:31f914d5 10:61dec1b8 09:fa000bc8 08:0ea4e3f0

07:99c99eb5 06:01d936bb 05:4f549ab4 04:82d49257 03:056b74c0 02:f87d0a5a 01:9e6f24ce 00:09363669

R1:706b3aa4 R2:d0d6a880

Snow 2.0 Internal state at time 5

15:3e9ee9ba 14:be037f37 13:962fd59e 12:a5e7b806 11:9098ec10 10:31f914d5 09:61dec1b8 08:fa000bc8
07:0ea4e3f0 06:99c99eb5 05:01d936bb 04:4f549ab4 03:82d49257 02:056b74c0 01:f87d0a5a 00:9e6f24ce
R1:202b4334 R2:86c48227

Snow 2.0 Internal state at time 6

15:a14a61e1 14:3e9ee9ba 13:be037f87 12:962fd59e 11:a5e7b806 10:9098ec10 09:31f914d5 08:61dec1b8
07:fa000bc8 06:0ea4e3f0 05:99c99eb5 04:01d936bb 03:4f549ab4 02:82d49257 01:056b74c0 00:f87d0a5a
R1:889db8e2 R2:b6399358

Snow 2.0 Internal state at time 7

15:cc852528 14:a14a61e1 13:3e9ee9ba 12:be037f87 11:962fd59e 10:a5e7b806 09:9098ec10 08:31f914d5
07:61dec1b8 06:fa000bc8 05:0ea4e3f0 04:99c99eb5 03:01d936bb 02:4f549ab4 01:82d49257 00:056b74c0
R1:5003320d R2:121f6605

Snow 2.0 Internal state at time 8

15:4b895ab7 14:cc852528 13:a14a61e1 12:3e9ee9ba 11:be037f87 10:962fd59e 09:a5e7b806 08:9098ec10
07:31f914d5 06:61dec1b8 05:fa000bc8 04:0ea4e3f0 03:99c99eb5 02:01d936bb 01:4f549ab4 00:82d49257
R1:20c449f5 R2:9cf74ff8

C.3 Ví dụ khóa 256 bit cho SNOW 2.0

C.3.1 Khóa, véc tơ khởi tạo và bộ ba khóa dòng

$(IV_3, IV_2, IV_1, IV_0) = (0,0,0,0)$,

key = 80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00

Keystream output: 0B5BCCE20323E28E0FC203809C66AB73CA35A680F2A5DD197E0C5C02287BE822.

$(IV_3, IV_2, IV_1, IV_0) = (0,0,0,0)$,

key = AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Keystream output: D9CC22FD861492D0AE6F43FB0F072012078C5AEEE479DE8CF0E555F458EED858.

(IV_3, IV_2, IV_1, IV_0) = (4,3,2,1) ,

key = 80000000000000000000000000000000000000000000000000000000000000000000000000000000

Keystream output: 7861080D5755E90B736F10916ED519B12C1A3A4255297FC2246A37FA6C089526.

(IV_3, IV_2, IV_1, IV_0) = (4,3,2,1) ,

key = AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Keystream output: 29261FCE5ED038201D6AFAF8B87E74FED49ECB10197EAC025D024EB45E0C7655.

C.3.2 Ví dụ các trạng thái trong

K = 80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 00 00 00 04 00 00 00 03 00 00 00 02 00 00 00 01

Z = 78 61 08 0d 57 55 e9 0b 73 6f 10 91 6e d5 19 b1 2c 1a 3a 42 55 29 7f c2 24 6a b7 fa 6c 08 95 26

Snow 2.0 Internal state at time -34

15:80000000 14:00000000 13:00000000 12:00000000 11:00000000 10:00000000 09:00000000 08:00000000
07:7fffffff 06:ffffffff 05:ffffffff 04:ffffffff 03:ffffffff 02:ffffffff 01:ffffffff 00:ffffffff
R1:0804a9cc R2:00000001

Snow 2.0 Internal state at time -33

15:80000001 14:00000000 13:00000000 12:00000002 11:00000000 10:00000003 09:00000004 08:00000000

07:7fffffff 06:ffffffff 05:ffffffff 04:ffffffff 03:ffffffff 02:ffffffff 01:ffffffff 00:ffffffff

R1:00000000 R2:00000000

Snow 2.0 Internal state at time -32

15:bf53b515 14:80000001 13:00000000 12:00000000 11:00000002 10:00000000 09:00000003 08:00000004

07:00000000 06:7fffffff 05:ffffffff 04:ffffffff 03:ffffffff 02:ffffffff 01:ffffffff 00:ffffffff

R1:ffffffff R2:63636363

Snow 2.0 Internal state at time -31

15:d37de350 14:bf53b515 13:80000001 12:00000000 11:00000000 10:00000002 09:00000000 08:00000003

07:00000004 06:00000000 05:7fffffff 04:ffffffff 03:ffffffff 02:ffffffff 01:ffffffff 00:ffffffff

R1:63636362 R2:16161616

Snow 2.0 Internal state at time -30

15:1fa4e5b0 14:d37de350 13:bf53b515 12:80000001 11:00000000 10:00000000 09:00000002 08:00000000

07:00000003 06:00000004 05:00000000 04:7fffffff 03:ffffffff 02:ffffffff 01:ffffffff 00:ffffffff

R1:96161615 R2:08aaaa59

```
Snow 2.0 Internal state at time -29

15:8243e488 14:1fa4e5b0 13:d37de350 12:bf53b515 11:80000001 10:00000000 09:00000000 08:00000002

07:00000000 06:00000003 05:00000004 04:00000000 03:7fffffff 02:ffffffff 01:ffffffff 00:ffffffff

R1:08aaaa59 R2:d03b8eac


Snow 2.0 Internal state at time -28

15:7d09f594 14:8243e488 13:1fa4e5b0 12:d37de350 11:bf53b515 10:80000001 09:00000000 08:00000000

07:00000002 06:00000000 05:00000003 04:00000004 03:00000000 02:7fffffff 01:ffffffff 00:ffffffff

R1:d03b8eb0 R2:267457fe


Snow 2.0 In ernal state at time -27

15:851e8381 14:7d09f594 13:8243e488 12:1fa4e5b0 11:d37de350 10:bf53b515 09:80000001 08:00000000

07:00000000 06:00000000 05:00000003 04:00000004 03:00000004 02:7fffffff 01:7fffffff 00:ffffffff

R1:26745801 R2:29b1986c


Snow 2.0 Internal state at time -26

15:cf3efe13 14:851e8381 13:7d09f594 12:8243e488 11:1fa4e5b0 10:d37de350 09:bf53b515 08:80000001
07:00000000 06:00000000 05:00000002 04:00000000 03:00000003 02:00000004 01:00000000 00:7fffffff
R1:29b1986c R2:892bf223


Snow 2.0 Internal state at time -25

15:7b69e0b3 14:cf3efe13 13:851e8381 12:7d09f594 11:8243e488 10:1fa4e5b0 09:d37de350 08:bf53b515
07:80000001 05:00000000 05:00000000 04:00000002 03:00000000 02:00000003 01:00000004 00:00000000
R1:892bf225 R2:2f693a07


Snow 2.0 Internal state at time -24

15:0a8b53d5 14:7b69e0b3 13:cf3efe13 12:851e8381 11:7d09f594 10:8243e488 09:1fa4e5b0 08:d37de350
07:bf53b515 06:80000001 05:00000000 04:00000000 03:00000002 02:00000000 01:00000003 00:00000004
R1:2f693a07 R2:6cbd99a8


Snow 2.0 Internal state at time -23

15:fd75ecf7 14:0a8b53d5 13:7b69e0b3 12:cf3efe13 11:851e8381 10:7d09f594 09:8243e488 08:1fa4e5b0

07:d37de350 06:bf53b515 05:80000001 04:00000000 03:00000000 02:00000002 01:00000000 00:00000003

R1:6cbd99a8 R2:0793dbe5


Snow 2.0 Internal state at time -22

15:94a70314 14:fd75ecf7 13:0a8b53d5 12:7b69e0b3 11:cf3efe13 10:851e8381 09:7d09f594 08:8243e488
```

07:1fa4e5b0 06:d37de350 05:bf53b515 04:80000001 03:00000000 02:00000000 01:00000002 00:00000000

R1:8793dbe7 R2:6928db9c

Snow 2.0 Internal state at time -21

15:743ca456 14:94a70314 13:fd75ecf7 12:0a8b53d5 11:7b69e0b3 10:cf3efe13 09:851e8381 08:7d09f594
07:8243e488 06:1fa4e5b0 05:d37de350 04:bf53b515 03:80000001 02:00000000 01:00000000 00:00000002
R1:287c90b1 R2:ecb79528

Snow 2.0 Internal state at time -20

15:c250e943 14:743ca456 13:94a70314 12:fd75ecf7 11:0a8b53d5 10:7b69e0b3 09:cf3efe13 08:851e8381
07:7d09f594 06:8243e488 05:1fa4e5b0 04:d37de350 03:bf53b515 02:80000001 01:00000000 00:00000000
R1:c0357878 R2:5bd40c0f

Snow 2.0 Internal state at time -19

15:4d848699 14:c250e943 13:743ca456 12:94a70314 11:fd75ecf7 10:0a8b53d5 09:7b69e0b3 08:cf3efe13
07:851e8381 06:7d09f594 05:8243e488 04:1fa4e5b0 03:d37de350 02:bf53b515 01:80000001 00:00000000
R1:7b78f1bf R2:9ae2c490

Snow 2.0 Internal state at time -18

15:9b3a221f 14:4d848699 13:c250e943 12:743ca456 11:94a70314 10:fd75ecf7 09:0a8b53d5 08:7b69e0b3

07:cf3efe13 06:851e8381 05:7d09f594 04:8243e488 03:8243e488 02:1fa4e5b0 01:bf53b515 00:80000001

R1:1d26a918 R2:47a9af75

Snow 2.0 Internal state at time -17

15:35d95fd1 14:9b3a221f 13:c250e943 12:743ca456 11:743ca456 10:94a70314 09:fd75ecf7 08:0a8b53d5

07:7b69e0b3 06:cf3efe13 05:851e8381 04:7d09f594 03:1fa4e5b0 02:d37de350 01:d37de350 00:bf53b515

R1:c4b3a509 R2:9b7cb67c

Snow 2.0 Internal state at time -16

15:e7492c66 14:35d95fd1 13:9b3a221f 12:4d848699 11:c250e943 10:743ca456 09:94a70314 08:fd75ecf7

07:0a8b53d5 06:7b69e0b3 05:cf3efe13 04:851e8381 03:7a09f594 02:8243e488 01: Ifa4e5b0 00:d37de350

R1:209b39fd R2:50f9a679

Snow 2.0 Internal state at time -15

15:dce814e5 14:e7492c66 13:35d95fd1 12:9b3a221f 11:4d848699 10:c250e943 09:743ca456 08:94a70314

07:fd75ecf7 06:0a8b53d5 05:7b69e0b3 04:cf3efe13 03:851e8331 02:7d09f594 01:8243e483 00:1fa4e5b0

R1:2038a48c R2:8facfb3d

Snow 2.0 Internal state at time -14

15:e8e83f37 14:dce814e5 13:35d95fdl 12:9b3a221f 11:9b3a221f 10:4d848699 09:c250e943 08:743ca456

07:94a70314 06:fd75ecf7 05:7b69e0b3 04:cf3efel3 03:cf3efel3 02:851e8381 01:7d09f594 00:8243e488

Rl:0bl6dbf0 R2:97el48a3

Snow 2.0 Internal state at time -13

15:387810f3 14:e8e83f37 13:e7492c66 12:35d95fdl 11:35d95fdl 10:9b3a221f 09:4d848699 08:c250e943

07:743ca456 06:94a70314 05:0a8b53d5 04:7b69e0b3 03:7b69e0b3 02:cf3efel3 01:851e8381 00:7d09f594

Rl:a26c9c78 R2:27c607bf

Snow 2.0 Internal state at time -12

15:4bcddadb 14:387810f3 13:dce814e5 12:e7492c66 11:e7492c66 10:35d95fdl 09:9b3a221f 08:4d848699

07:c250e943 06:743ca456 05:fd75ecf7 04:0a8b53d5 03:0a8b53d5 02:7b69e0b3 01:cf3efel3 00:851e8381

Rl:253bf4b6 R2:258cdl70

Snow 2.0 Internal state at time -11

15:f05c5d45 14:4bcddadb 13:e8e83f37 12:dce814e5 11:ace814e5 10:e7492c66 09:35d95fdl 08:9b3a221f

07:4d848699 06:c250e943 05:94a70314 04:fd75ecf7 03:fd75ecf7 02:0a8b53d5 01:7b69e0b3 00:cf3efel3

Rl:ba33d484 R2:f16f299b

Snow 2.0 Internal state at time -10

15:21e0b756 14:f05c5d45 13:387810f3 12:e8e83f37 11: e8e83f37 10:dce814e5 09:e7492c66 08:35d95fdl

07:9b3a221f 06:4d848699 05:743ca456 04:94a70314 03: 94a70314 02:fd75ecf7 01:0a8b53d5 00:7b69e0b3

Rl:65abcdf1 R2:998d6551

Snow 2.0 Internal state at time -9

15:00098c25 14:21e0b756 13:4bcddadb 12:387810f3 11:387810f3 10:e8e83f37 09:dce814e5 08:e7492c66

07:35d95fdl 06:9b3a221f 05:c250e943 04:743ca456 03:743ca456 02:94a70314 01:fd75ecf7 00:0a8b53d5

Rl:5bde4e94 R2:bd0f2baa

Snow 2.0 Internal state at time -8

15:81a343el 14:00098c25 13:f05c5d45 12:4bcddadb 11:4bcddadb 10:387810f3 09:e8e83f37 08:dce814e5

07:e7492c66 06:35d95fdl 05:4d848699 04:c250e943 03:c250e943 02:743ca456 01: 94a70314 00:fd75ecf7

Rl:0a93b243 R2:267c6211

```
Snow 2.0 Internal state at time -7

15:7b933a92 14:81a343e1 13:21e0b756 12:f05c5d45 11:f05c5d45 10:4bcddadb 09:387810f3 08:e8e83f37

07:dce814e5 06:e7492c66 05:9b3a221f 04:4d848699 03:4d848699 02:c250e943 01:743ca456 00:94a70314

R1:c1b68430 R2:0b276cd6


Snow 2.0 Internal state at time -6

15:0339b827 14:7b933a92 13:00098c25 12:21e0b756 11:21e0b756 10:f05c5d45 09:4bcddadb 08:387810f3

07:e8e83f37 06:dce814e5 05:35d95fd1 C4:9b3a221f 03:9b3a221f 02:4d848699 01:c250e943 00:743ca456

R1:4100cca7 R2:ed4f10df


Snow 2.0 Internal state at time -5

15:e5fd1e4e 14:0339b827 13:81a343e1 12:00098c25 11:00098c25 10:21e0b756 09:f05c5d45 08:4bcddadb

07:387810f3 06:e8e83f37 05:e7492c66 04:35d95fd1 03:35d95fd1 02:9b3a221f 01:4d848699 00:c250e943

R1:d4983d45 R2:d14fec85


Snow 2.0 Internal state at time -4

15:991f7362 14:e5fd1e4e 13:0339b827 12:7b933a92 11:81a343e1 10:00098c25 09:21e0b756 08:f05c5d45
07:4bcddadb 06:387810f3 05:e8e83f37 04:dce814e5 03:e7492c66 02:35d95fd1 01:9b3a221f 00:4d848699
R1:ae38016a R2:431da2bb


Snow 2.0 Internal state at time -3

15:a0bca2f7 14:991f7362 13:e5fd1e4e 12:0339b827 11:7b933a92 10:81a343e1 09:00098c25 08:21e0b756
07:f05c5d45 06:4bcddadb 05:387810f3 04:e8e83f37 03:dce814e5 02:e7492c66 01:35d95fd1 00:9b3a221f
R1:2c05e1f2 R2:ae471763


Snow 2.0 Internal state at time -2

15:928b5256 14:a0bca2f7 13:991f7362 12:e5fd1e4e 11:0339b827 10:7b933a92 09:81a343e1 08:00098c25
07:21e0b756 06:f05c5d45 05:4bcddadb 04:387810f3 03:e8e83f37 02:dce814e5 01:e7492c66 00:35d95fd1
R1:e6bf2856 R2:f134ae00


Snow 2.0 Internal state at time -1

15:be366d56 14:928b5256 13:a0bca2f7 12:991f7362 11:e5fd1e4e 10:0339b827 09:7b933a92 08:31a343e1
07:000S8c25 06:21e0b756 05:f05c5d45 04:4bcddadb 03:387810f3 02:e8e83f37 01:dce814e5 00:e7492c66
R1:3d0288db R2:f31c4fa3


Snow 2.0 Internal state at time 0

15:a5fadb9e 14:be366d56 13:928b5256 12:a0bca2f7 11:991f7362 10:e5fd1e4e 09:0339b827 08:7b933a92
07:81a343e1 06:00098c25 05:21e0b756 04:f05c5d45 03:4bcddadb 02:387810f3 01:e8e83f37 00:dce814e5
R1:e378ace8 R2:2dfa946e
```

```
Snow 2.0 Internal state at time 1

15:b70c6e50 14:a5fadb9e 13:be366d56 12:928b5256 11:a0bca2f7 10:991f7362 09:e5fdle4e 08:0339b827
07:7b933a92 06:81a343el 05:00098c25 04:21e0b756 03:f05c5d45 02:4bcddadb 01:387810f3 00:e8e83f37
Rl:4fdb4bc4 R2:b95a6c28



Snow 2.0 Internal state at time 2

15:bce23d5c 14:b70c6e50 13:a5fadb9e 12:be366d56 11:928b5256 10:a0bca2f7 09:991f7362 08:e5fdle4e
07:0339b327 06:7b933a92 05:81a343el 04:00098c25 03:21e0b756 02:f05c5d45 01:4bcddadb 00:387810f3
Rl:b963f84d R2:3d5135cb



Snow 2.0 Internal state at time 3

15:4eb9692d 14:bce23d5c 13:b70c6e50 12:a5fadb9e 11:be366d56 10:928b5256 09:a0bca2f7 08:991f7362
07:e5fdle4e 06:0339b827 05:7b933a92 04:81a343el 03:00098c25 02:21e0b756 01:f05c5d45 00:4bcddadb
Rl:bef479ac R2:28b521b3



Snow 2.0 Internal state at time 4

15:96a599a9 14:4eb9692d 13:bce23d5c 12:b70c6e50 11:a5fadb9e 10:be366d56 09:928b5256 08:a0bca2f7
07:991f7362 06:e5fdle4e 05:0339b827 04:7b933a92 03:81a343el 02:00098c25 01:21e0b756 00:f05c5d45
Rl:a4485c45 R2:e6ab92e9



Snow 2.0 Internal state at time 5

15:62ad427d 14:96a599a9 13:4eb9692d 12:bce23d5c Il:b70c6e50 10:a5fadb9e 09:be366d56 08:928b5256
07:a0bca2f7 06:991f7362 05:e5fdle4e 04:0339b827 03:7b933a92 02:81a343el 01:00098c25 00:21e0b756
Rl:e9e54b10 R2:385b4519

Snow 2.0 Internal state at time 6

15:19397ef2 14:62ad427d 13:96a599a9 12:4eb9692d 11:bce23d5c 10:b70c6e50 09:a5fadb9e 08:be366d56

07:928b5256 06:a0bca2f7 05:991f7362 04:e5fdle4e 03:0339b827 02:7b933a92 01:81a343el 00:00098c25

Rl:1e586367 R2:13f2d986



Snow 2.0 Internal state at time 7

15:5f8525f0 14:19397ef2 13:62ad427d 12:96a599a9 11:4eb9692d 10:bce23d5c 09:b70c6e50 08:a5fadb9e

07:be366d56 06:928b5256 05:a0bca2f7 04:991f7362 03:e5fdle4e 02:0339b827 01:7b933a92 00:81a343el

Rl:adl24ce8 R2:el3ca41f



Snow 2.0 Internal state at time 8

15:fb9c0bcf 14:5f8525f0 13:19397ef2 12:62ad427d 11:96a599a9 10:4eb9692d 09:bce23d5c 08:b70coe50
07:a5fadb9e 06:be366d56 05:928b5256 04:a0bca2f7 03:991f7362 02:e5fdle4e 01:0339b827 00:7b933a92
Rl:81f94716 R2:679flc0a
```

C.4 Ví dụ cho Rabbit

C.4.1 Giới thiệu

Tất cả các véc tơ kiểm tra cho Rabbit được đưa ra trong ký hiệu little-endian, tức là cho số nhiều byte, các byte có trọng số cao nhất được lưu trữ tại các địa chỉ bộ nhớ cao nhất.

C.4.2 Khóa, véc tơ kiểm tra và bộ ba khóa dòng

K = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 00 00 00 00 00 00 00 00

Z = ED B7 05 67 37 5D CD 7C D8 95 54 F8 5E 27 A7 C6 D8 4A DC 70 32 29 8F 7B D4 EF F5 04 AC A6 29 5F

66 8F BF 47 8A DB 2B E5 1E 6C DE 29 2B 82 DE 2A C6 56 59 72 22 0E C9 09 A7 E7 57 60 98

K = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV= 00 01 02 03 04 05 06 07

Z = 98 71 C7 BA 4E A3 08 07 CD AA 49 64 66 39 2D 2F 4A FF 43 55 EF 90 69 56 10 9B 96 65 97 8D AC ED

9B 7C 6F 7F C8 2C 67 D2 73 22 CB DE 9D 30 16 45 8C 38 2C 9C 7D 30 44 E6 52 03 B9 2A 13 53 C0 FF

K = 00 01 02 03 04 05 06 07 08 09 0A 03 0C 0D 0E OF

IV= 00 00 00 00 00 00 00 00

Z = A8 F7 E6 9B 69 40 A7 8D 13 6A 5C 15 4A 15 79 52 A6 E4 23 58 59 E3 02 20 EA 68 64 36 BB 38 EF 53

9C 29 40 55 6B 09 EC D7 FE A2 B0 AC 83 07 FI 69 62 65 A3 D6 44 28 1C 39 C9 CD 5E IE 2F 9B E4 DO

K = 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E OF

IV= 00 01 02 03 04 05 06 07

Z = F2 89 19 DD A1 28 F8 F9 0A 30 34 6E 97 94 D2 B7 4C 69 A2 D9 91 37 27 BC 5A 30 18 E6 33 2A F7 F3

3E 3A C3 EF 33 68 F4 3A 4C B8 58 67 B8 1C 91 F9 24 29 0C 81 6E 8B 57 83 98 C5 7F B4 C0 BA 05 BD

C.4.3 Ví dụ các trạng thái trong

K = 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

IV= 00 01 02 03 04 05 06 07

Z = f2 89 19 DD A1 28 F8 F9 0A 30 34 6E 97 94 D2 B7 4C 69 A2 D9 91 37 27 BC 5A 30 18 E6 33 2A F7 F3

BE 3A C3 EF B3 68 F4 3A 4C B8 58 67 B8 1C 91 F9 24 29 0C 81 6B 8B 57 88 98 C5 7F B4 C0 BA 05 BD

After key expansion (Internal state S(-9))

x0:03020100 x1:0D0C0B0A x2:07060504 x3:01000F0E x4:0B0A0908 x5:05040302 x6:0F0E0D0C x7:09080706

c0:09080B0A c1:03020504 c2:0D0C0F0E c3:07060908 c4:01000302 c5:0B0A0D0C c6:05040706 c7:0F0E0100

carry:0

After key setup iteration 1 (Internal state S (-8))

x0:05783933 x1:162113C0 x2:B38F168E x3:F08A919E x4:7F2CDA94 x5:ACBEB878 x6:0D5257A9 x7:4FF46B46

c0:563CDE57 c1:D64F39D7 c2:41DF5C42 c3:543ADC55 c4:D44D37D5 c5:3FDD5A40 c6:5238DA53 c7:E25B35D3

carry:0

After key setup iteration 2 (Internal state S (-7))

x0:798C2CEC x1:CC05FFD4 x2:50D68324 x3:2C306745 x4:AD519559 x5:81595E7A x6:29A589E2 x7:15212B97

c0:A371B1A4 c1:A99C6EAA c2:76B2A977 c3:A16FAFA2 c4:A79A6CA8 c5:74B0A775 c6:9F6DADA0 c7:B5A86AA6

carry:1

After key setup iteration 3 (Internal state S (-6))

x0:CD328957 x1:66D5Ab1F x2:0D115824 x3:FCCEB784 x4:12E900D7 x5:36A46997 x6:9F40C5BC X7:AB1C8A08

c0:F0A684F2 c1:7CE9A37D c2:AB85F6AC c3:EEA482EF c4:7AE7A17B c5:A983F4AA c6:ECA280ED c7:88F59F79

carry:1

After key setup iteration 4 (Internal state S (-5))

x0:A31515F8 x1:5DFD3AC6 x2:33CD6AD2 x3:4BD778E5 x4:89708269 x5:D93095C1 x6:5E495F60 x7:C197863A

c0:3DDB5840 c1:5036D851 c2:E05943E1 c3:3BD9563C c4:4E34D64F c5:DE5741DF c6:39D7543A c7:5C42D44D

carry:1

After counter modification / IV setup (Internal state S(-4))

x0:A31515F8 x1:5DFD3AC6 x2:33CD6A02 x3:4BD778E5 x4:89708269 x5:D9309SC1 x6:5E495F60 x7:C197863A

c0:B7A9DB29 c1:8E004E92 c2:B9161985 c3:FF4AD106 c4:EE23C2B7 c5:84AC781B c6:0D1C3BEC c7:1291ADA8

carry:1

After IV setup iteration 1 (Internal state S(-3))

x0:054A3F2F x1:BE444CDE x2:573425A4 x3:9347FAD1 x4:29036A2F x5:DD3C6B50 x6:12CC3803 x7:6F7847C0

c0:04DEAE77 c1:614D8366 c2:EDE966BA c3:4C7FA453 c4:C170F78B c5:B97FC550 c6:5A510F39 c7:E5DEE27B

carry:0

After IV setup iteration 2 (Internal state S(-2))

x0:0rDB9A3A x1:334807E8 x2:E663CC98 x3:0FDA371C x4:9C3E3036 x5:7774E657 x6:C6FCB34C x7:A3D1AC4F

c0:521381C4 c1:349AB839 c2:22BCB3EF c3:99B477A1 c4:94BE2C5E c5:EE531285 c6:A785E286 c7:B92C174E

Carry:1

After IV setup iteration 3 (Internal state S (-1))

x0:1A2EF77E x1:FDEEE287 x2:A918F5A1 x3:D6414F76 x4:4848D473 x5:BCE9ED30 x6:3E524094 x7:16242C51

```
c0:9F485512 c1:07E7ED0C c2:57900124 c3:E6E94AEE c4:680B6131 c5:23265FBA c6:F4BAB5D4 c7:8C794C21
carry:1

After IV setup iteration 4 (Internal state S(0))
x0:987651C2 x1:FF5F0007 x2:5C48C79E x3:661B3E75 x4:49247B9A X5:3C7AA744 x6:4AEF3F40 x7:D117584E
c0:EC7D2860 c1:D33521DF c2:8C634E58 c3:341E1E3B c4:3B589605 c5:57F9ACEF c6:41EF8921 c7:5FC680F5
carry:1

After keystream iteration 1 (Internal state S(1))
x0:2A158BE4 x1:D93EC5A4 x2:298B7C1f x3:01F4F70C x4:E241E934 x5:0216D073 x6:72769563 x7:54BA8C75
c0:39B1FBAE c1:AE8256B3 c2:C1369B8D c3:8152F188 c4:0EA5CAD8 c5:8CCCFA24 c6:8F245C6E c7:3313B5C8
carry:1
output F2 89 19 DD A1 28 F8 F9 0A 30 34 6E 97 94 D2 B7

After keystream iteration 2 (Internal state S(2))
x0:46EC0492 x1:A4B5D46E x2:7B374C9E x3:93249F4E x4:E93894EF x5:6DDEC710 x6:2799B917 x7:7B0F0F20
c0:86E6CEFC c1:81CF8B86 c2:F609E8C2 c3:CE87C4D5 c4:E1F2FFAB c5:C1A04758 c6:DC5S2FBB c7:0660EA9B
carry:1
output 4C 69 A2 D9 91 37 27 BC 5A 30 18 E6 33 2A F7 F3

After keystream iteration 3 (Internal state S(3))
x0:98C27422 x1:0D5B5EC2 x2:FEEC9F8D x3:423F7701 x4:E22AB517 x5:4E9CC418 x6:A7535E87 x7:F73E8572
c0:D41EA24A c1:551CC059 c2:2ADD35F7 c3:1BBC9823 c4:B540347F c5:F673948D c6:298E0308 c7:D9AE1F6F
carry:0
output BE 3A C3 EF B3 58 F4 3A 4C B8 58 67 B8 1C 91 F9

After keystream iteration 4 (Internal state S(4))
x0:3B844C36 x1:AF5CD78B x2:2619A0AC x3:774FBA88 x4:D16C6AC4 x5:6512AE4E x6:6A8ECD8F x7:2BC76513
c0:21507597 c1:2869F52D c2:5FB0832C c3:68F16B70 c4:888D6952 c5:2B46E1C2 c6:76C2D656 c7:ACFB5442
carry:1
```

C.5 Ví dụ cho *Decim*^{v2}

C.5.1 Giới thiệu ví dụ *Decim*^{v2}

Các giá trị byte và phân tích nhị phân của các byte theo ký hiệu big-endian, tức là với số nhiều byte, byte có trọng số cao nhất được lưu trữ tại địa chỉ bộ nhớ thấp nhất. Đặc biệt, lưu trữ khóa, IV, khóa dòng, thanh ghi và bộ đệm byte và giá trị nhị phân đưa ra dưới đây.

$K = K_{79} \dots K_0$

$IV = IV_{63} \dots IV_0$

$Z = Z_n \dots Z_0$

$a = a_{191} \dots a_0$

$b = b_{31} \dots b_0$

$T = T_2 T_1 T_0$

và, ví dụ đưa ra khóa

$K = de\ aa\ 00\ 40\ 30\ 00\ 0f\ 08\ 80$

Mỗi thanh ghi được xác định:

$[K_{79} \dots K_{72}] = \text{de}, [K_{71} \dots K_{64}] = \text{aa} \dots [K_7 \dots K_0] = 80$

Với phân tích bit như sau:

$K_{79} \dots K_0 =$ 11011110 10101010 00000000 01000000 00000000

00110000 00000000 00001111 00001000 10000000

C.5.2 Khóa, véc tơ khởi tạo và bộ ba khóa dòng

```
K = 00 00 00 00 00 00 00 00 00 80
IV= 00 00 00 00 00 00 00 00
Z = 76 e3 89 be 1b fb ad d5 3c ce a0 fe 43 b3 c8 fb d3 92 b8 0b 52 94 60 00

K = 00 00 00 00 00 00 00 00 00 00
IV= 00 00 00 00 00 00 00 80
Z = 4c ec bd b3 0e cd c9 c0 8b 41 8f 7f 28 ff 83 48 75 40 ff c5 cb 0a 33 da

K = 09 09 09 09 09 09 09 09 09 09
IV= 00 00 00 00 00 00 00 00
Z = 43 9b ba f8 a1 84 dc f9 e6 d2 90 1d 12 4d 43 09 22 33 f2 47 60 19 70 53

K = 09 08 07 06 05 04 03 02 01 00
IV= 00 00 00 00 00 00 00 00
Z = 52 b1 73 10 01 2a cd 3a d2 20 4f e2 b2 2a 5d 21 64 41 f6 3d d3 b4 43 6a

K = eb 98 45 f2 9f 4c f9 a6 53 00
IV= de 77 10 a9 42 db 74 0d
```

z = 62 ff c9 cc 21 0e 07 ea 6e 50 f0 fb 1b 60 36 1f 88 a6 a5 27 9b 18 cb b8

K = fa a7 54 01 ae 5b 08 b5 62 0f

IV= f9 92 2b c4 5d f6 8f 28

Z = f0 af 66 52 2a 23 8b 29 63 37 8b 18 ec 1f 4c a8 27 91 3d 2c f0 ad 94 d9

C.5.3 Ví dụ trạng thái trong

Các tương đương nhị phân của trạng thái trong cho các giai đoạn khóa, cụ thể được cung cấp tại thời điểm -256, thời điểm -64, thời điểm 0 và thời điểm 193.

K = 00 00 00 00 00 00 00 00 00 80

IV= 00 00 00 00 00 00 00 00

z = 76 e3 89 be 1b fb ad d5 3c ce a0 fe 43 b8 c8 fb d3 92 b8 0b 52 94 60 f8

Từ -256 đến -64 (thực thi của *InitNext(S,LRSR)*, các biến trạng thái trong *T, b* và I có giá trị sau:

T: 000

B: 00 00 00 00

I: 0

Decim v2 Internal State at time -256

a: ff ff ff ff 00 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00 00 00 00 80

Decim v2 Binary Internal State at time -256 (Binary notation)

a: 11111111 11111111 11111111 11111111 00000000 00000000 00000000 00000000
00000000 10000000 00000000 00000000 00000000 00000000 00000000 10000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

Executions of InitNext(S,LFSR)

Decim v2 Internal State at time -255

a: 2f ff ff ff fC 00 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00 00 00 00

Decim v2 Internal State at time -254

a: 42 ff ff ff ff 00 00 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00 00 00

Decim v2 Internal State at time -253

a: 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00 00

Decim v2 Internal State at time -252

a: 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00 00

Decim v2 Internal State at time -251

a: 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00

Decim v2 Internal State at time -250

a: 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 80 00 00 00 00 00 00

Decim v2 Internal State at time -249

a: e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 80 00 00 00 00 00 00

Decim v2 Internal State at time -248

a: 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 80 00 00 00 00 00

Decim v2 Internal State at time -247

a: 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 80 00 00 00 00 00

Decim v2 Internal State at time -246

a: 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 80 00 00 00 00

Decim v2 Internal State at time -245

a: d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 80 00 00 00 00

Decim v2 Internal State at time -244

a: 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 80 00 00 00

Decim v2 Internal State at time -243

a: 16 d0 04 e9 99 64 2f ff ff ff f0 00 00 00 00 00 00 00 80 00 00 00

Decim v2 Internal State at time -242

a: d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 00 00 80 00 00

Decim v2 Internal State at time -241

a: fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 00 80 00 00

Decim v2 Internal State at time -240

a: df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 00 80 00 00

Decim v2 Internal State at time -239

a: ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 08 00 00

Decim v2 Internal State at time -238

a: fe df d1.6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 00 80 00

Decim v2 Internal State at time -237

a: ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 08 00

Decim v2 Internal State at time -236

a: ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 00 80

Decim v2 Internal State at time -235

a: ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00 08

Decim v2 Internal State at time -234

a: 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00 00

Decim v2 Internal State at time -233

a: 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00 00

Decim v2 Internal State at time -232

a: 06 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00 00

Decim v2 Internal State at time -231

a: b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00 00

Decim v2 Internal State at time -230

a: 5b 06 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00 00

Decim v2 Internal State at time -229

a: c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00 00

Decim v2 Internal State at time -228

a: be 5b 06 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00 00

Decim v2 Internal State at time -227

a: 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00 00

Decim v2 Internal State at time -226

a: d6 be 5b 06 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff ff 00 00 00 00 00

Decim v2 Internal State at time -225

a: bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00 00

Decim v2 Internal State at time -224

a: 8b d6 c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff 00 00 00 00 00

Decim v2 Internal State at time -223

a: 38 bd 6b c5 b0 60 ce fe ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00 00

Decim v2 Internal State at time -222

a: b3 8b d6 c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff ff f0 00 00

Decim v2 Internal State at time -221

a: fb 38 bd 6b c5 b0 60 ce fe ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00 00

Decim v2 Internal State at time -220

a: 4f b3 8b d6 c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff ff 00 00

Decim v2 Internal State at time -219

a: 84 fb 38 bd 6b c5 b0 60 ce fe ed fd 16 d0 04 e8 99 84 2f ff ff ff f0 00

Decim v2 Internal State at time -218

a: 18 4f b3 8b d6 c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff ff 00

Decim v2 Internal State at time -217

a: 21 84 fb 38 bd 6b c5 b0 60 ce fe ed fd 16 d0 04 e8 99 84 2f ff ff ff f0

Decim v2 Internal State at time -216

a: f2 84 fb 38 bd 6b c5 b0 60 ce fe ed fd 16 d0 04 e8 99 84 2f ff ff ff ff

Decira v2 Internal State at time -215

a: 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff ff

Decira v2 Internal State at time -214

a: e9 f2 18 4f b3 8b d0 be 5b 06 0c ee fe df d1 6d 00 4e 89 98 42 ff ff ff

Decira v2 Internal State at time -213

a: 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff ff

Decira v2 Internal State at time -212

a: f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c ee fe df d1 6d 00 4e 99 98 42 ff ff

Decira v2 Internal State at time -211

a: 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f ff

Decira v2 Internal State at time -210

a: 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c ee fe df d1 6d 00 4e 89 98 42 ff

Decira v2 Internal State at time -209

a: 04 9f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84 2f

Decira v2 Internal State at time -208

a: 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c ee fe df d1 6d 00 4e 89 98 42

Decira v2 Internal State at time -207

a: d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99 84

Decira v2 Internal State at time -206

a: 1d 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c ee fe df d1 6d 00 4e 89 98

Decira v2 Internal State at time 205

a: 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8 99

Decira v2 Internal State at time -204

a: 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 bc 5b 06 0c ee fe df d1 6d 00 4e 89

Decira v2 Internal State at time -203

a: 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed fd 16 d0 04 e8

Decira v2 Internal State at time -202

a: 88 13 1d 10 45 f6 e9 f2 18 4f b3 38 d6 be 5b 06 0c ee fe df d1 6d 00 4e

Decira v2 Internal State at time -201

a: c3 81 31 d1 04 5f 6e 9f 21 84 fb b3 bd 6b c5 b0 60 ce ef ed fd 16 d0 04

Decira v2 Internal State at time -200

a: 7c 88 13 1d 10 45 f6 e9 f2 18 4f fb 8b d6 be 5b 06 0c ee fe df d1 6d 00

Decira v2 Internal State at time -199

a: 77 c8 31 31 d1 04 5f 6e 9f 21 84 4f 38 bd 6b c5 b0 60 ce ef ed fd 16 d0

Decira v2 Internal State at time -198

a: 17 7c 88 13 1d 10 45 f6 e9 f2 18 84 b3 8b d6 be 5b 06 0c ee fe df d1 6d

Decira v2 Internal State at time -197

a: 91 77 c8 81 31 d1 04 5f 6e 9f 21 18 fb 38 bd 6b c5 b0 60 ce ef ed fd 16

Decira v2 Internal State at time -196

a: 49 17 7c 88 13 1d 10 45 f6 e9 f2 21 4f b3 8b d6 be 5b 06 0c ee fe df d1

Decira v2 Internal State at time -195

a: 44 91 77 c8 81 31 d1 04 5f 6e 9f f2 34 fb 38 bd 6b c5 b0 60 ce ef ed fd

Decira v2 Internal State at time -194

a: c4 49 17 7c 88 13 1d 10 45 f6 e9 9f 18 4f b3 8b d6 be 5b 06 0c ee fe df

Decira v2 Internal State at time -193

a: bc 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef ed

Decira v2 Internal State at time -192

a: 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d0 be 5b 06 0c ee fe

Decira v2 Internal State at time -191

a: cb be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce ef

Decim v2 Internal State at time -190

a: 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c ee

Decim v2 Internal State at time -189

a: 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60 ce

Decim v2 Internal State at time -188

a: 92 5c5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06 0c

Decim v2 Internal State at time -187

a: a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0 60

Decim v2 Internal State at time -186

a: ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b 06

Decim v2 Internal State at time -185

a: fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5 b0

Decim v2 Internal State at time -184

a: 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 be 5b

Decim v2 Internal State at time -183

a: 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd 6b c5

Decim v2 Internal State at time -182

a: f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6 bc

Decim v2 Internal State at time -181

a: af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6c 9f 21 84 fb 38 bd 6d

Decim v2 Internal State at time -180

a: 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3 8b d6

Decim v2 Internal State at time -179

a: 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38 bd

Decim v2 Internal State at time -178

a: b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 68 13 1d 10 45 f6 e9 f2 18 4f b3 8b

Decim v2 Internal State at time -177

a: 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb 38

Decim v2 Internal State at time -17 6

a: 76 b4 6a. f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f b3

Decim v2 Internal State at time -175

a: e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84 fb

Decim v2 Internal State at time -174

a: 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18 4f

Decim v2 Internal State at time -173

a: d2 e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21 84

Decim v2 Internal State at time -172

a: 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2 18

Decim v2 Internal State at time -171

a: f8 d2 e1 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e 9f 21

Decim v2 Internal State at time -170

a: 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9 f2

Decim v2 Internal State at time -169

a: d6 f8 d2 e1 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6 9fe

Decim v2 Internal State at time -168

a: cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6 e9

Decim v2 Internal State at time -167

a: dc d6 f8 d2 e1 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 d1 04 5f 6e

Decim v2 Internal State at time -166

a: ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45 f6

Decim v2 Internal State at time -165

a: fe dc do f8 d2 e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 dl 04 5f

Decim v2 Internal State at time -164

a: 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10 45

Decim v2 Internal State at time -163

a: 21 fe dc d6 f8 d2 e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 dl 04

Decim v2 Internal State at time -162

a: 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d 10

Decim v2 Internal State at time -161

a: 62 21 fe dc d6 f8 d2 e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31 dl

Decim v2 Internal State at time -160

a: 3d 22 If ed cd of 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13 1d

Decim v2 Internal State at time -159

a: 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81 31

Decim v2 Internal State at time -158

a: 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c 88 13

Decim v2 Internal State at time -157

a: d4 53 d2 21 fe dc d6 f8 d2 e7 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8 81

Decim v2 Internal State at time -156

a: ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a £8 9f ca 92 5c 5b c4 49 17 7c 88

Decim v2 Internal State at time -155

a: 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c5 be 44 91 77 c8

Decim v2 Internal State at time -154

a: 23 ed 45 3d 22 1f ed cd 6f 3d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17 7c

Decim v2 Internal State at time -153

a: b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c5 be 44 91 77

Decim v2 Internal State at time -152

a: 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49 17

Decim v2 Internal State at time -151

a: 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c9 be 44 91

Decim v2 Internal State at time -150

a: e9 1b 23 ed 45 3d 22 If ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4 49

Decim v2 Internal State at time -149

a: fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c5 be 44

Decim v2 Internal State at time -148

a: cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b c4

Decim v2 Internal State at time -147

a: 4c fe 91 b2 3e d4 53 d2 21 fe dc do f8 d2 el 6b 46 af 89 fc a9 25 c5 be

Decim v2 Internal State at time -146

a: 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c 5b

Decim v2 Internal State at time -145

a: 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89 fc a9 25 c5

Decim v2 Internal State at time -144

a: 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92 5c

Decim v2 Internal State at time -143

a: d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e1 6b 46 af 89 fc a9 25

Decim v2 Internal State at time -142

a: 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca 92

Decim v2 Internal State at time -141

a: b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e1 6b 46 af 89 fc a9

Decim v2 Internal State at time -140

a: 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8 9f ca

Decim v2 Internal State at time -139

a: f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e7 6b 46 af 89 fc

Decim v2 Internal State at time -138

a: 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 If ed cd 6f 8d 2e 76 b4 6a f8 9f

Decim v2 Internal State at time -137

a: 51 f9 bo d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af 89

Decim v2 Internal State at time -136

a: 85 If 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a f8

Decim v2 Internal State at time -135

a: 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 el 6b 46 af

Decim v2 Internal State at time -134

a: el 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4 6a

Decim v2 Internal State at time -133

a: 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e7 6b 46

Decim v2 Internal State at time -132

a: f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e 76 b4

Decim v2 Internal State at time -131

a: 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e1 6b

Decim v2 Internal State at time -130

a: 39 f6 e1 85 If 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 If ed cd 6f 8d 2e 76

Decim v2 Internal State at time -129

a: c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2 e1

Decim v2 Internal State at time -128

a: ec 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d 2e

Decim v2 Internal State at time -127

a: 38 c3 9f 6e 18 51 f9 b0 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8 d2

Decim v2 Internal State at time -126

a: 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f 8d

Decim v2 Internal State at time -125

a: 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6 f8

Decim v2 Internal State at time -124

a: 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd 6f

Decim v2 Internal State at time -123

a: 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc d6

Decim v2 Internal State at time -122

a: 07 75 53 8c 39 f6 e 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed cd

Decim v2 Internal State at time -121

a: 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe dc

Decim v2 Internal State at time -120

a: f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f ed

Decim v2 Internal State at time -119

a: 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21 fe

Decim v2 Internal State at time -118

a: d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22 1f

Decim v2 Internal State at time -117

a: 9d 6f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53 d2 21

Decim v2 Internal State at time -116

a: 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d 22

Decim v2 Internal State at time -115

a: 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f5 b6 d9 46 4c fe 91 b2 3e d4 53 d2

Decim v2 Internal State at time -114

a: 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45 3d

Decim v2 Internal State at time -113

a: 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4 53

Decim v2 Internal State at time -112

a: 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed 45

Decim v2 Internal State at time -111

a: 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e d4

Decim v2 Internal State at time -110

a: b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23 ed

Decim v2 Internal State at time -109

a: 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2 3e

Decim v2 Internal State at time -108

a: 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b 23

Decim v2 Internal State at time -107

a: 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91 b2

Decim v2 Internal State at time -106

a: a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9 1b

Decim v2 Internal State at time -105

a: ba 47 1b 60 64 66 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe 91

Decim v2 Internal State at time -104

a: bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf e9

Decim v2 Internal State at time -103

a: 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c fe

Decim v2 Internal State at time -102

a: 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64 cf

Decim v2 Internal State at time -101

a: 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46 4c

Decim v2 Internal State at time -100

a: a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94 64

Decim v2 Internal State at time -99

a: 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9 46

Decim v2 Internal State at time -93

a: 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d 94

Decim v2 Internal State at time -97

a: 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6 d9

Decim v2 Internal State at time -96

a: d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f 9b 6d

Decim v2 Internal State at time -95

a: 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51 f9 b6

Decim v2 Internal State at time -94

a: 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f0 e1 85 1f 9b

Decim v2 Internal State at time -93

a: 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 33 c3 9f 6e 13 51 f9

Decim v2 Internal State at time -92

a: 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85 1f

Decim v2 Internal State at time -91

a: 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18 51

Decim v2 Internal State at time -90

a: a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1 85

Decim v2 Internal State at time -89

a: 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e 18

Decim v2 Internal State at time -88

a: a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6 e1

Decim v2 Internal State at time -87

a: da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f 6e

Decim v2 Internal State at time -86

a: 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39 f6

Decim v2 Internal State at time -85

a: d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38 c3 9f

Decim v2 Internal State at time -84

a: ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c 39

Decim v2 Internal State at time -83

a: 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 6f 10 77 55 38 c3

Decim v2 Internal State at time -82

a: 57 ad 9d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53 8c

Decim v2 Internal State at time -81

a: 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55 38

Decim v2 Internal State at time -80

a: a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75 53

Decim v2 Internal State at time -7 9

a: 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77 55

Decim v2 Internal State at time -78

a: 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07 75

Decim v2 Internal State at time -77

a: c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10 77

Decim v2 Internal State at time -76

a: 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1 07

Decim v2 Internal State at time -75

a: 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f 10

Decim v2 Internal State at time -74

a: 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8 f1

Decim v2 Internal State at time -73

a: 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d 8f

Decim v2 Internal State at time -72

a: 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69 d8

Decim v2 Internal State at time -71

a: 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86 9d

Decim v2 Internal State at time -70

a: 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48 69

Decim v2 Internal State at time -69

a: 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64 86

Decim v2 Internal State at time -68

a: f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06 48

Decim v2 Internal State at time -67

a: 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60 64

Decim v2 Internal State at time -66

a: 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb a4 71 b6 06

Decim v2 Internal State at time -65

```
a: 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b 60


Decim v2 Internal State at time -64

a: 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 al 49 69 dl 58 a3 55 bb a4 71 b6


Decim v2 Internal State at time -64 (Binary notation)
a: 00111000 00100011 11111001 01010100 10010100 01100011 01011000 11010001
   01101001 01010111 10101101 10001101 10100101 10100001 01001001 10100111
   10000111 01001100 10100011 01010101 10111011 10100100 01110001 10110110
T: 000  b: 00000000 00000000 00000000 00000000  I: 0
Executions of InitNext(S,BUFF)
Decim v2 Internal State at time -63
a: 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba 47 1b
T: 111 b: 00 00 00 00 I: 1

Decim v2 Internal State at time -62
a: a2 38 23 f9 54 94 63 4c 87 al 57 ad 8d a5 a1 49 69 dl 58 a3 55 bb a4 71
T: 111 b: 00 00 00 00 I: 2

Decim v2 Internal State at time -61
a: 7a 23 82 3f 95 49 46 34 c8 7a 75 7s d8 da 5a 14 96 9d 15 8a 35 5b ba 47
T: 101  b: 00 00 00 00 I: 3

Decim v2 Internal State at time -60
a: 07 a2 38 23 f9 54 9463 4c 87 al 57 ad 8d a5 a1 49 69 dl 58 a3 55 bb a4
T: 101 b: 00 00 00 08 I: 4

Decim v2 Internal State at time -59
a: b0 1a 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b ba
T: 101 b: 00 00 00 18 I: 5

Decim v2 Internal State at time -58
a: 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55 bb
T: 001 b: 00 00 00 38 I: 6

Decim v2 Internal State at time -57
a: 33 b0 7a 23 823f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35 5b
T: 101 b: 00 00 00 78 I: 7

Decim v2 Internal State at time -56
a: 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58 a3 55
T: 001 b: 00 00 00 f8 I: 9

Decim v2 Internal State at time -55
a: 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d 15 8a 35
T: 001 b: 00 00 00 f8 I: 11

Decim v2 Internal State at time -54
a: 20 53 3b 07 a2 38 23 f9 54 94 63 4c 67 a1 57 ad 8d a5 a1 49 69 d1 58 a3
T: 010 b: 00 00 00 f8 I: 13

Decim v2 Internal State at time -53
a: a2 05 33 b0 1a 23 82 3f 95 49 46 34 c8 7a 75 1a d8 da 5a 14 96 9d 15 8a
T: 010 b: 00 00 60 f8 I: 15

Decim v2 Internal State at time -52
a: 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1 58
T: 011 b: 00 00 60 f8 I: 16

Decim v2 Internal State at time -51
a: 93 a2 05 33 b0 1a 23 82 3f 95 49 46 34 c8 1a 75 1a d8 da 5a 14 96 9d 15
T: Oil b: 00 00 60 f8 I: 17

Decim v2 Internal State at time -50
a: d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49 69 d1
```

T: 010 b: 00 00 60 f8 I: 18

Decim v2 Internal State at time -49
a: 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14 96 9d
T: 001 b: 00 04 60 f8 I: 19

Decim v2 Internal State at time -48
a: 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a7 49 69
T: 001 b: 00 14 60 f8 I: 21

Decim v2 Internal State at time -47
a: 64 2d 93 a2 05 33 b0 7a 23 82 3f 55 49 46 34 c8 7a 75 7a d8 da 5a 14 96
T: 001 b: 00 14 60 f8 I: 23

Decim v2 Internal State at time -46
a: 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1 49
T: 000 b: 00 94 60 f8 I: 25

Decim v2 Internal State at time -45
a: 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da 5a 14
T: 000 b: 02 94 60 f8 I: 27

Decim v2 Internal State at time -44
a: e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5 a1
T: 010 b: 12 94 60 f8 I: 29

Decim v2 Internal Stateat time -43
a: 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 4946 34 c8 7a 75 7a d8 da 5a
T: 010 b: 12 94 60 f8 I: 30

Decim v2 Internal State at time -4 2
a: 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5
T: 000 b: 52 94 60 f8 I: 32

Decim v2 Internal State at time 0
a: 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d a5
T: 000 b: 52 94 60 f8 I: 32

Decim v2 Internal State at time 0 (3inary notation)
a: 00000111 11100100 01010110 01000010 11011001 00111010 01001100 01100011
10010100 00000111 10100010 00111000 00100011 11111001 01010100 00111011
01010011 00100000 10000111 10100111 01010111 10101101 10001101 10100101
T: 000 b: 01010010 10010100 01100000 11111000 I: 32

Executions of Next(S)
Decim v2 Internal State at time 1
a: a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a d8 da
T: 101 b: a9 4a 30 7c I: 32

Decim v2 Internal State at time 2
a: 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad 8d
T: 111 b: d4 a5 18 3e I: 32

Decim v2 Internal State at time 3
a: b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a d8
T: 010 b: 6a 52 8c If I: 32

Decim v2 Internal State at time 4
a: bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57 ad
T: 111 b: b5 29 46 0f I: 32

Decim v2 Internal State at time 5
a: eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75 7a
T: 111 b: 5a 94 a3 07 I: 32

Decim v2 Internal State at time 6
a: ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7 57
T: 111 b: 2d 4a 51 83 I: 31

```
Decim v2 Internal State at time 7
a: fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a 75
T: 110 b: 16 a5 28 cl I: 32

Decim v2 Internal State at time 8
a: 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87 a7
T: 111 b: 0b 52 94 60 I: 32

Decim v2 Internal State at time 9
a: f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8 7a
T: 010 b: 05 a9 4a 30 I: 32

Decim v2 Internal State at time 10
a: 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c 87
T: 010 b: 02 d4 a5 18 I: 32

Decim v2 Internal State at time 11
a: f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34 c8
T: 101 b: 01 6a 52 8c I: 32

Decim v2 Internal State at time 12
a: 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63 4c
T: 100 b: 80 b5 29 46 I: 32

Decim v2 Internal State at time 13
a: 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46 34
T: 101 b: c0 5a 94 a3 I: 31

Decim v2 Internal State at time 14
a: 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54 94 63
T: 110 b: e0 2d 4a 51 I: 31

Decim v2 Internal State at time 15
a: 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49 46
T: 110 b: 70 16 a5 28 I: 32

Decim v2 Internal State at time 16
a: f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 3e 23 f9 54 94
T: 000 b: b8 Ob 52 94 I: 32

Decim v2 Internal State at time 17
a: 3f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95 49
T: 111 b: dc 05 a9 4a I: 31

Decim v2 Internal State at time 18
a: 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23 f9 54
T: 100 b: ae 02 d4 a5 I: 32

Decim v2 Internal State at time 19
a: 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f 95
T: 010 b: 57 01 6a 52 I: 32

Decim v2 Internal State at time 20
a: b7 48 f5 37 6f 3f 3f ce bb a0 7e 45 64 2d 93 a 2 05 3 3b0 7a 38 23 f9
T: 101 b: 2b 80 b5 29 I: 32

Decim v2 Internal State at time 21
a: eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82 3f
T: 101 b: 15 c0 5a 94 I: 31

Decim v2 Internal State at time 22
a: be b7 43 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38 23
T: 110 b: 4a e0 2d 4a I: 31

Decim v2 Internal State at time 23
a: 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23 82
T: 000 b: 25 70 16 a5 I: 32

Decim v2 Internal State at time 24
```

a: 39 be b7 43 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2 38
T: 100 b: 92 b8 0b 52 I: 32

Decim v2 Internal State at time 25
a: 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a 23
T: 101 b: c9 5c 05 a9 I: 32

Decim v2 Internal State at time 26
a: 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07 a2
T: 010 b: e4 ae 02 d4 I: 32

Decim v2 Internal State at time 27
a: 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0 7a
T: 111 b: 72 57 01 6a I: 32

Decim v2 Internal State at time 28
a: b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3b 07
T: 100 b: 39 2b 80 b5 I: 32

Decim v2 Internal State at time 29
a: 6b 5243 9b eb74 8f5376 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33 b0
T: 101 b: 1c 95 c0 5a I: 31

Decim v2 Internal State at time 30
a: d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53 3
T: 000 b: 4e 4a e0 2d I: 32

Decim v2 Internal State at time 31
a: bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05 33
T: 010 b: a7 25 70 16 I: 32

Decim v2 Internal State at time 32
a: 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20 53
T: 110 b: d3 92 b8 0b I: 32

Decim v2 Internal State at time 33
a: c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2 05
T: 110 b: e9 c9 5c 05 I: 32

Decim v2 Internal State at time 34
a: 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a 20
T: 111 b: f4 e4 ae 02 I: 32

Decim v2 Internal State at time 35
a: c3 c4 bd 6b 52 43 9b eb 74 Bf 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93 a2
T: 101 b: 7a 72 57 01 I: 32

Decim v2 Internal State at time 36
a: dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9 3a
T: 110 b: bd 39 2b 80 I: 32

Decim v2 Internal State at time 37
a: fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d 93
T: 110 b: de 9c 95 c0 I: 32

Decim v2 Internal State at time 38
a: af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56 42 d9
T: 110 b: ef 4e 4a e0 I: 32

Decim v2 Internal State at time 39
a: 2a fd c3 c4 bd 6b 52 43 9b eb 74 Bf 53 76 f3 f3 fc eb b0 a0 7e 45 64 2d
T: 010 b: f7 a7 25 70 I: 32

Decim v2 Internal State at time 40
a: e2 af dc 3c 4b d6 b5 24 39 be b7 4B f5 37 6f 3f 3f ce bb 0a 07 e4 56 42
T: 000 b: fb d3 92 b8 I: 32

Decim v2 Internal State at time 41
a: fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45 64
T: 111 b: 7d e9 c9 5c I: 32

Decim v2 Internal State at time 42
a: 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4 56
T: 000 b: 3e f4 e4 ae I: 32

```
Decim v2 Internal State at time 43
a: 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e 45
T: 010 b: If 7a 72 57 I: 32

Decim v2 Internal State at time 44
a: 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07 e4
T: 000 b: 8f bd 39 2b I: 32

Decim v2 Internal State at time 45
a: d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0 7e
T: 111 b: 47 de 9c 95 I: 32

Decim v2 Internal State at time 46
a: Id 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a 07
T: 000 b: 23 ef 4e 4a I: 32

Decim v2 Internal State at time 47
a: 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0 a0

T: 101 b: 11 f7 a7 25 I: 31

Decim v2 Internal State at time 48
a: f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb 0a

T: 101 b: 48 fb d3 92 I: 31

Decim v2 Internal State at time 49

a: 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb b0

T: 110 b: 64 7d e9 c9 I: 32
Decim v2 Internal State at time 50

a: 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce bb

T: 111 b: 32 3e f4 e4 I: 32
Decim v2 Internal State at time 51

a: e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc eb

T: 111 b: 19 If 7a 72 I: 31
Decim v2 Internal State at time 52

a: fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f 3f ce

T: 101 b: 0c 8f bd 39 I: 31
Decim v2 Internal State at time 53

a: 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3 f3 fc

T: 000 b: 46 47 de 9c I: 31
Decim v2 Internal State at time 54

a: 70 fe 86 f8 Id 35 9f e2 af dc 3c d6 b5 24 39 be b7 48 f5 37 6f 3f 3f

T: 100 b: 63 23 ef 4e I: 31
Decim v2 Internal State at time 55

a: 27 0f e8 6f 81 d3 59 fe 2a fd c3 bd 6b 52 43 9b eb 74 Bf 53 76 f3 f3

T: 111 b: 71 91 f7 a1 I: 31
Decim v2 Internal State at time 56

a: 32 70 fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f 3f
```

T: 100 b: b8 c8 fb d3 I: 32

Decim v2 Internal State at time 57
a: 73 27 0f e3 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76 f3

T: 100 b: dc 64 7d e9 I: 32

Decim v2 Internal State at time 58
a: 47 32 70 fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37 6f

T: 111 b: ee 32 3e f4 I: 32

Decim v2 Internal State at time 59
a: b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53 76

T: 000 b: 77 19 1f 7a I: 32

Decim v2 Internal State at time 60

a: 0b 47 32 70 fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5 37

T: 010 b: 3b 8c 8f bd I: 32

Decim v2 Internal State at time 61

a: 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f 53

T: 100 b: Id c6 47 de I: 32

Decim v2 Internal State at time 62

a: f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48 f5

T: 101 b: 0e e3 23 ef I: 32

Decim v2 Internal State at time 63

a: af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74 8f

T: 111 b: 87 71 91 f7 I: 32

Decim v2 Internal State at time 64

a: ba f5 0b 47 32 70 fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7 48

T: 111 b: 43 b8 cS fb I: 32

Deciia v2 Internal State at time 65
a: bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb 74

T: 010 b: 21 dc 64 7d I: 32

Decim v2 Internal State at time 66
a: eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b d6 b5 24 39 be b7

T: 000 b: 90 ee 32 3e I: 32

Decim v2 Internal State at time 67
a: 2e bb af 50 b4 73 27 0f e8 f6 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b eb

T: 110 b: c8 77 19 If I: 32

Decim v2 Internal State at time 68
a: f2 eb ba f5 0b 47 32 70 fe 86 f8 1d 35 9f e2 af dc 3c 4b d6 b5 24 39 be
T: 100 b: e4 3b 8c 8f I: 32

Decim v2 Internal State at time 69
a: cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43 9b
T: 101 b: f2 1d c6 47 I: 32

Decim v2 Internal State at time 70
a: be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b d6 b5 24 39

```
T: 000 b: f9 0e e3 23 I: 32

Decim v2 Internal State at time 71
a: ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52 43
T: 000 b: fc 87 71 91 I: 32

Decim v2 Internal State at time 72
a: ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b d6 b5 24
T: 101 b: fe 43 b8 c8 I: 31

Decim v2 Internal State at time 73
a: ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b 52
T: 010 b: 7f 21 dc 64 I: 32

Decim v2 Internal State at time 74
a: 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b d6 b5
T: 010 b: 3f 90 ee 32 I: 32

Decim v2 Internal State at time 75
a: 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd 6b
T: 110 b: If c8 77 19 I: 32

Decim v2 Internal State at time 76
a: 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c 4b a6
T: 110 b: 0f e4 3b 8c I: 32

Decim v2 Internal State at time 77
a: b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4 bd
T: 000 b: 07 f2 Id c6 I: 32

Decim v2 Internal mState at time 7 8
a: 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 r8 Id 35 9f e2 af dc 3c 4b
T: 100 b: 83 f 9 0e e3 I: 32

Decim v2 Internal State at time 79
a: a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3 c4
T: 101 b: 41 fc 87 71 I: 32

Decim v2 Internal State at time 80
a: aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc 3c
T: 010 b: a0 fe 43 b8 I: 32

Decim v2 Internal State at time 81
a: 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd c3
T: 111 b: d0 7f 21 dc I: 31

Decim v2 Internal State at time 82
a: 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2 af dc
T: 010 b: a8 3f 90 ee I: 31

Decim v2 Internal State at time 83
a: f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a fd
T: 110 b: d4 1f c8 77 I: 31

Decim v2 Internal State at time 84
a: df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 1d 35 9f e2 af
T: 110 b: ea 0f e4 3b I: 32

Decim v2 Internal State at time 85
a: bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe 2a
T: 111 b: 75 07 f2 Id I: 32

Decim v2 Internal State at time 86
a: 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 Id 35 9f e2
T: 110b: 3a 83 f9 0e I: 32

Decim v2 Internal State at time 87
a: e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59 fe
T: 101 b: 9d 41 fc 87 I: 32

Decim v2 Internal State at time 88
a: 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 Ob 47 32 70 fe 86 f8 Id 35 9f
T: 000 b: ce a0 fe 43 I: 32

Decim v2 Internal State at time 89
```

a: 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3 59
T: 010 b: 67 50 7f 21 I: 32

Decim v2 Internal State at time 90
a: 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 1d 35
T: 110 b: 33 a8 3f 90 I: 32

Decim v2 Internal State at time 91
a: 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81 d3
T: 000 b: 99 d4 1f c8 I: 32

Decim v2 Internal State at time 92
a: 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8 1d
T: 000 b: cc ea 0f e4 I: 32

Decim v2 Internal State at time 93
a: f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f 81
T: 101 b: e6 75 07 f2 I: 32

Decim v2 Internal State at time 94
a: df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86 f8
T: 000 b: f3 3a 83 f9 I: 32

Decim v2 Internal State at time 95
a: 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f e8 6f
T: 111 b: fS 9d 41 fc I: 31

Decim v2 Internal State at time 96
a: 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe 86
T: 111 b: be ce a0 fe I: 31

Decim v2 Internal State at time 97
a: 27 7d f9 20 92 e8 bd f6 8a a3 b0 36 ce ab cf 2e bb af 50 b4 73 27 0f e8
T: 010 b: 9e 67 50 7f I: 32

Decim v2 Internal State at time 98
a: c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70 fe
T: 110 b: 4f 33 a8 3f I: 32

Decim v2 Internal State at time 99
a: 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27 0f
T: 110 b: a7 99 d4 If I:32

Decim v2 Internal State at time 100
a: c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32 70
T: 000 b: 53 cc ea 0f I: 32

Decim v2 Internal State at time 101
a: 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73 27
T: 000 b: a9 e6 75 07 I: 32

Decim v2 Internal State at time 102
a: e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47 32
T: 010 b: 54 f3 3a 83 I: 32

Decim v2 Internal State at time 103
a: 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50 b4 73
T: 010 b: aa 79 9d 41 I: 32

Decim v2 Internal mState at time 104
a: 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b 47
T: 000 b: d5 3c ce a0 I: 32

Decim v2 Internal State at time 105
a: 53 4e 0c 7c 27 7d f9 20 92 e8 bd fb 8a a3 b6 36 ce ab cf 2e bb af 50 b4
T: 110 b: ea 9e 67 50 I: 32

Decim v2 Internal State at time 106
a: 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5 0b
T: 010 b: 75 4f 33 a8 I: 32

Decim v2 Internal State at time 107
a: a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af 50
T: 010 b: ba a7 99 d4 I: 32

Decim v2 Internal State at time 108
a: 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba f5

T: 100 b: dd 53 ce ea I: 32

Decim v2 Internal State at time 109
a: 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb af
T: 010 b: 6e a9 e6 75 I: 32

Decim v2 Internal State at time 110
a: 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb ba
T: 110 b: b7 54 f3 3a I: 32

Decim v2 Internal State at time 111
a: 36 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e bb
T: 010 b: 5b aa 79 9d I: 32

Decim v2 Internal State at time 112
a: 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2 eb
T: 101 b: ad d5 3c ce I: 32

Decim v2 Internal State at time 113
a: 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf 2e
T: 000 b: d6 ea 9e 67 I: 32

Decim v2 Internal State at time 114
a: 49 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be f2
T: 100 b: eb 75 4f 33 I: 32

Decim v2 Internal State at time 115
a: 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab cf
T: 100 b: 75 ba a7 99 I: 32

Decim v2 Internal State at time 116
a: c0 49 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c ea be
T: 010 b: ba dd 53 ce I: 32

Decim v2 Internal State at time 117
a: ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce ab
T: 100 b: dd 6e a9 e6 I: 32

Decim v2 Internal State at time 118
a: ba c0 49 13 83 6a 85 34 e0 c1 c2 11 df 92 09 2e 8b df 68 aa 3b 63 6c ea
T: 010 b: ee b7 54 f3 I: 32

Decim v2 Internal State at time 119
a: 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36 ce
T: 000 b: f7 5b aa 79 I: 32

Decim v2 Internal State at time 120
a: 14 ba c0 49 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa 3b 63 6c
T: 010 b: fb ad d5 3c I: 32

Decim v2 Internal State at time 121
a: 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6 36
T: 000 b: fd d6 ea 9e I: 32

Decim v2 Internal State at time 122
a: 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 77 df 92 09 2e 8b df 68 aa 3b 63
T: 010 b: fe eb 75 4f I: 32

Decim v2 Internal State at time 123
a: b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3 b6
T: 100 b: 7f 75 ba a7 I: 32

Decim v2 Internal State at time 124
a: bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 77 df 92 09 2e 8b df 68 aa 3b
T: 000 b: bf ba dd 53 I: 32

Decim v2 Internal State at time
125

a: 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a a3

T: 111 b: df d 6e a9 I: 32

Decim v2 Internal State at time 126

a: 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68 aa
T: 100 b: 6f ee b7 54 I: 32

Decim v2 Internal State at time 127
a: 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6 8a
T: 110 b: 37 f7 5b aa I: 32

Decim v2 Internal State at time 128
a: 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c7 c2 77 df 92 09 2e 8b df 68
T: 111 b: 1b fb ad d5 I: 32

Decim v2 Internal State at time 129
a: b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8 bd f6
T: 101 b: 0d fd d0 ea I: 32

Decim v2 Internal State at time 130
a: 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 11 df 92 09 2e 8b df
T: 100 b: 86 fe eb 75 I: 32

Decim v2 Internal State at time 131
a: e4 b2 22 9b b7 41 4b ac 04 91 38 36 aS 53 4e 0c 1c 21 7d f9 20 92 e8 bd
T: 100 b: c3 7f 75 ba I: 32

Decim v2 Internal State at time 132
a: 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 11 df 92 09 2e 8b
T: 101 b: el bf ba dd I: 31

Decim v2 Internal State at time 133
a: 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92 e8
T: 110 b: f0 df dd 6e I: 32

Decim v2 Internal State at time 134
a: 15 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 77 df 92 09 2e
T: 010 b: f8 6f ee b7 I: 32

Decim v2 Internal State at time 135
a: 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20 92
T: 101 b: 1c 37 f7 5b I: 32

Decim v2 Internal State at time 136
a: 89 15 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 11 df 92 09
T: 101 b: be 1b fb ad I: 32

Decim v2 Internal State at time 137
a: e3 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9 20
T: 111 b: df 0d fd d6 I: 32

Decim v2 Internal State at time 138
a: 5e 89 15 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 77 df 92
T: 010 b: 6f 86 fe eb I: 32

Decim v2 Internal State at time 139
a: 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27 7d f9
T: 101 b: 37 c3 7f 75 I: 32

Decim v2 Internal State at time 140
a: al 5e 89 15 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c7 c2 77 df
T: 000 b: 9b el bf ba I: 32

Decim v2 Internal State at time 141
a: 7a 15 e8 91 51 e4 b2 22 5b b7 41 4b ac 04 91 33 36 a8 53 4e 0c 7c 27 7d
T: 010 b: 4ci f0 df dd I: 32

Decim v2 Internal State at time 142
a: d7 al 5e 89 15 1e 4b 22 25 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c1 c2 11
T: 111 b: 26 f8 6f ee I: 32

Decim v2 Internal State at time 143
a: dd la 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c 27
T: 110 b: 13 7c 37 fl I: 32

Decira v2 Internal State at time 144
a: 2d d7 al 5e b9 15 1e 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c7 c2
T: 100 b: 89 be 1b fb I: 32

```
Decim v2 Internal State at time 145
a: a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c 7c
T: 110 b: c4 df 0d fd I: 32

Decim v2 Internal State at time 14 6
a: ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0 c7
T: 110 b: e2 6f 86 fe I: 32

Decim v2 Internal State at time 147
a: le a2 dd la 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e 0c
T: 111 b: 71 37 c3 7f I: 32

Decim v2 Internal State at time 148
a: 11 ea 2d dl al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34 e0
T: 000 b: 38 9b el bf I: 32

Decim v2 Internal State at time 149
a: 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53 4e
T: 010 b: 1c 4d f0 df I: 32

Decim v2 Internal State at time 150
a: 31 11 ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85 34
T: 101 b: 0e 26 f8 6f I: 31

Decim v2 Internal State at time 151
a: 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8 53
T: 100 b: c7 13 7c 37 I: 32

Decim v2 Internal State at time 152
a: a7 31 11 ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83 6a 85
T: 010 b: e3 89 be lb I: 32

Decim v2 Internal State at time 153
a: fa 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38 36 a8
T: 100 b: 71 c4 df 0d I: 32

Decim v2 Internal State at time 154
a: df a7 31 11 ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83 6a
T: 100 b: b8 e2 6f 86 I: 32

Decim v2 Internal State at time 155
a: ad fa 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b hi 41 4b ac 04 91 38 36
T: 000 b: dc 71 37 c3 I: 32

Decim v2 Internal State at time 156
a: la df a7 31 11 ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13 83
T: 100 b: 6e 38 9b el I: 32

Decim v2 Internal State at time 157
a: 41 ad fa 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91 38
T: 101 b: b7 1c 4d f0 I: 32

Decim v2 Internal State at time 158
a: 84 la df al 31 11 ea 2d dl al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49 13
T: 100 b: db 8e 26 f8 I: 32

Decim v2 Internal State at time 159
a: a8 41 ad fa 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04 91
T: 111 b: ed cl 13 7c I: 32

Decim v2 Internal State at time 160
a: 6a 84 la df a7 31 11 ea 2d d7 al 5e 89 15 le 4b 22 29 bb 74 14 ba c0 49
T: 101 b: 76 e3 89 be I: 32

Decim v2 Internal State at time 161
a: a6 a3 41 ad fa 73 11 le a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac 04
T: 000 b: bb 71 c4 df I: 32


Decim v2 Internal State at time 162
a: 0a 6a 84 1a df a7 31 11 ea 2d d7 al 5e 89 15 1e 4b 22 29 bb 74 14 ba c0
T: 111 b: dd bS e2 6f I: 31


Decim v2 Internal State at time 163
```

a: 60 a6 a8 41 ad fa 73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b ac

T: 010 b: ae dc 71 37 I: 31

Decim v2 Internal State at time 164

a: 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a7 5e 89 15 7e 4b 22 29 bb 74 14 ba

T: 100 b: 97 6e 38 9b I: 31

Decim v2 Internal State at time 165
a: f2 60 a6 a8 41 ad fa73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41 4b
T: 110 b: cb b7 1c 4d I: 31

Decim v2 Internal State at time 166
a: df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a7 5e 89 15 7e 4b 22 29 bb 74 14
T: 101 b: a5 db 8e 26 I: 31

Decim v2 Internal State at time 167
a: 8d f2 60 a6 a8 41 ad fa 73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7 41
T: 100 b: d2 ed c7 13 I: 31

Decim v2 Internal State at time 168
a: 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a7 5e 89 15 7e 4b 22 29 bb 74
T: 010 b: 29 76 e3 89 I: 32

Decim v2 Internal State at time 169
a: 26 8d f2 60 a6 a8 41 ad fa 73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22 9b b7
T: 101 b: 14 bb 71 c4 I: 31

Decim v2 Internal State at time 170
a: f2 68 df 26 0a 6a 84 7a df a7 31 11 ea 2d d7 a7 5e 89 15 7e 4b 22 29 bb
T: 010 b: 4a 5d b8 e2 I: 32

Decim v2 Internal State at time 171
a: af 26 8d f2 60 a6 a8 41 ad fa 73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22 9b
T: 101 b: 25 2e dc 71 I: 32

Decim v2 Internal State at time 172
a: aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a7 5e 89 15 7e 4b 22 29
T: 000 b: 92 97 6e 38 I: 32

Decim v2 Internal State at time 173
a: aa af 26 3d f2 60 a6 a8 41 ad fa 73 11 7e a2 dd 7a 15 e8 91 51 e4 b2 22
T: 010 b: 49 4b b7 1c I: 32

Decim v2 Internal State at time 174
a: 3a aa f2 68 df 26 0a 6a 84 7a df a7 31 11 ea 2d d7 a1 5e 89 15 7e 4b 22
T: 111 b: a4 a5 db 8e I: 32

Decim v2 Internal State at time 175
a; a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15 e8 91 51 e4 b2
T: 111 b: 52 52 ed c7 I: 32

Decim v2 Internal State at time 176
a: 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a7 5e 89 15 1e 4b
T: 010 b: 29 29 76 e3 I: 32

Decim v2 Internal State at time 177
a: a5 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15 e8 91 51 e4
T: 010 b: 94 94 bb 71 I: 32

Decim v2 Internal State at time 178
a: 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a1 5e 89 15 1e
T: 100 b: ca 4a 5d b8 I: 32

Decim v2 Internal State at time 179
a: 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15 e8 91 51
T: 010 b: e5 25 2e dc I: 32

Decim v2 Internal State at time 180
a: 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a1 5e 89 15
T: 101 b: f2 92 97 6e I: 31

```
Decim v2 Internal State at time 181
a: 44 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15 e8 91
 T: 000 b: f9 49 4b b7 I: 31
Decim v2 Internal State at time 182
a: 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a1 5e 89
T: 100 b: f ca 4 a 5 d b 1 : 3 1

Decim v2 Internal State at time 183
a: 94 44 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15 e8
T: 100 b: 3e 52 52 ed I: 32

Decim v2 Internal State at time 184
a: 99 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a1 5e
T: 010 b: If 29 29 76 I: 32

Decim v2 Internal State at time 185
a: 59 94 44 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a 15
T: 010 b: 8f 94 94 bb I: 32

Decim v2 Internal State at time 186
a: c5 99 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7 a1
T: 111 b: c7 ca 4a 5d I: 32

Decim v2 Internal State at time 187
a: 4c 59 94 44 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd 7a
T: 111 b: 63 e5 25 2e I: 32

Decim v2 Internal State at time 188
a: 34 c5 99 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d d7
T: 101 b: 31 f2 92 97 I: 32

Decim v2 Internal State at time 189
a: b3 4c 59 94 44 14 a6 a3 aa af 26 8d f2 60 a6 a8 41 ad fa 73 11 1e a2 dd
T: 100 b: 98 f9 49 4b I: 32

Decim v2 Internal State at time 190
a: ab 34 c5 99 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea 2d
T: 101 b: 4c 7c a4 a5 I: 32

Decim v2 Internal State at time 191
a: 4a b3 4c 59 94 44 14 a6 a3 aa af 26 ed f2 60 a6 a8 41 ad fa 73 11 1e a2
T: 101 b: a6 3e 52 52 I: 32

Decim v2 Internal State at time 192
a: 44 ab 34 c5 99 44 41 4a 6a 3a aa f2 68 df 26 0a 6a 84 1a df a7 31 11 ea
T: 010 b: d3 1f 29 29 I: 32

Decim v2 Internal State at time 192 (Binary notation)

a:01000100 10101011 00110100 11000101 10011001 01000100 01000001 01001010

  01101010 00111010 10101010 11110010 01101000 11011111 00100110 00001010

  01101010 10000100 00011010 11011111 10100111 00110001 00010001 11101010

T: 010 b: 11010011 00011111 00101001 00101001 I: 3
```

C.6 Các ví dụ cho Kcipher-2(K2)

C.6.1 Khóa, véc tơ khởi tạo và bộ ba khóa dòng

```
K = (K0,K1,K2,K3)=(0x00000000, 0x00000000, 0x00000000, 0x00000000)

IV = (IV0, IV1, IV2, IV3) = (0x00000000, 0x00000000, 0x00000000, 0x00000000)

Keystream[0] = 0xF871EBEF945B7272

Keystream[1] = 0xE40C04941DFF0537

Keystream[2] = 0x0B981A59FBC8AC57
```

Keystream[3] = 0x566D3B02C179DB34

Keystream[4] = 0x3B46F1FO33554C72

Keystream[5] = 0x5DE68BCC9872858F

Keystream[6] = 0x575496024062F0E9

Keystream[7] = 0xF932C998226DB63A

K = (KO,K1,K2,K3)=(0x0F1E2D3C, 0x4B5A6978, 0x8796A5B4, 0xC3D2E1F0)

IV = (IV0, IV1, IV2, IV3) = (0xF0E0D0C0, 0xB0A09Q80, 0x70605040, 0x30201000)

Keystream[0] = 0x9FB6B580A6A5E7AF

Keystream[1] = 0xD1989DC6A77D5E28

Keystream[2] = 0x4EFCC8CB7BCFB32B

Keystream[3] = 0xF69297F5DD974CE8

Keystream[4] = 0xFBD9139C7A71F41A

Keystream[5] = 0x61382C76D3D2F6CA

Keystream[6] = 0xD5265037659CF838

Keystream[7] = 0x774121C26F6474F3

K = (K0, K1, K2, K3) =(0xAC2F75C0, 0x43FBC367, 0x09D315F2, 0x245746D8)

IV = (IV0, IV1, IV2, IV3)=(0xF6B29A58, 0x45CCCD8C, 0x6229393A, 0x7A4842C1)

Keystream[0] = 0xDA38138B32864E05

Keystream[1] = 0x24B8B90944E5117A

Keystream[2] = 0xC3E883DCFA22C458

Keystream[3] = 0x1F2C9DDFE98DC5DE

Keystream[4] = 0x33B2FC05064C6FEF

Keystream[5] = 0xA9A3D3ED31060DFF

Keystream[6] = 0xF7DE1857E224E70F

Keystream[7] = 0x4EFE5C36CEB974AC

C.6.2 Ví dụ các trạng thái trong

K = (K0,K1,K2,K3)=(0x80000000, 0x00000000, 0x00000000, 0x00000000)

IV = (IV0, IV1, IV2, IV3)=(0x00000004, 0x00000003, 0x00000002, 0x00000001)

Keystream = (0x9B753FAA, 0x404A0EF5, 0x52919406, 0x18177FDD, 0xA419D11E, 0x47481D1B, 0x2DD49337

0x640BDEC91)

K2 Internal state at time -24

(A0, A1, A2, A3, A4) = (0xE2636363, 0x00000000, 0x00000000, 0x00000000, 0x80000000)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xBEFBFB5E, 0x5C98983D, 0x00000004, 0x00000003, 0xBEFBF35E, 0x5C98983D, 0x00000002, 0x00000001,0xE2636363, 0xE2636363, 0xE2636363)

(R1, L1, R2, L2) = (0x00000000, 0x00000000, 0x00000000, 0x00000000)

K2 internal state at time -23

(A0, A1, A2, A3, A4) = (0x00000000, 0x00000000, 0x00000000, 0x80000000, 0x1F84F87D)

(B0, B1, B2, E3, B4, B5, B6, B7, B8, B9, B10) =

(0x5C98 98 3D, 0x00000004, 0x00000003, 0xBEFBFB5E, 0x5C98983D, 0x00000002, 0x00000001, 0xE2636363,0xE2636363, 0xE2636363, 0x08CE4DDD)

(R1, L1, R2, L2) = (0x3D5E9898, QxAFAOF900, 0x63636363, 0x63636363)

K2 internal state at time -22

(A0, A1, A2, A3, A4) = (0x00000000, 0x00000000, 0x30000000, 0x1F84F87D, 0x1D219B45)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x00000004, 0x00000003, 0xBEFBFB5E, 0x5CS8983D, 0x00000002, 0x00000001, 0xE2636363, 0xE2636363,0xE2036363, 0x08CE4DDD, 0xD448E338)

(R1, L1, R2, L2) = (0x1BC16E6E, 0x2BE9E7CD, 0x9AD90539, 0x2EAA08EF)

K2 internal state at time -21

(A0, A1, A2, A3, A4) = (0x00000000, 0x80000000, 0x1F84F87D, 0x1D219B45, 0x83BD086B)

(B0, B1, B2, B3, B4, B5, B6, 87, B8, B9, B10) =

(0x00000003, 0xBEFBFB5E, 0x5C93983D, 0x00000002, 0x00000001, 0xE2636363, 0xE2636363, 0xE2636363,0x08CE4 DDD, 0xD448E338, 0xD791AF96)

(R1, L1, R2, L2) = (0xFF2C7778, 0x0830D3EF, 0x181A9D48, 0xAF1D5D29)

K2 internal state at time -20

(A0, A1, A2, A3, A4) = (0x80000000, 0x1F84F87D, 0x1D219B45, 0x83BD086B, 0x79AA791D)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xBEFBFB5E, 0x5C98983D, 0x00000002, 0x00000001, 0xE2636363, 0xE2636363, 0xE2636363, 0x08CE4DDD,0xD448E338, 0xD791AF96, 0xDAC909BO)

(R1, L1, R2, L2) = (0xD3AF4 401, 0xF0D6D793, 0x7791C800, 0x78E12F3B)

K2 internal state at time -19

(A0, A1, A2, A3, A4) = (0x1F84F87D, 0x1D219B45, 0x83BD086B, 0x79AA791D, 0x54BF0EB7)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x5C98983D, 0x00000002, 0x00000001, 0xE2636363, 0xE2636363, 0xE2636363, 0xO8CE4DDD, 0xD44BE338,0xD791AF96, 0xDAC909B0, 0xFFFA4197)

(R1, L1, R2, L2) = (0xB9569748, 0x8C9BF8C8, 0x2A3FA7CA, 0xC7628540)

K2 internal state at time -18

(A0, A1, A2, A3, A4) = (0x1D219B45, 0x83BD086B, 0x79AA791D, 0x54BF0EB7, 0xDA9BEC10)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x00000002, 0x00000001, 0xE2636363, 0xE2636363, 0xE2636363, 0x08CE4DDD, 0xD448E338, 0xD791AF96, 0xDAC909B0, 0xFFFA4197, 0xF546S78B)

(R1, L1, R2, L2) = (0x13228CE3, 0xB5FE6E22, 0x6359C7C0, 0xBE2D3278)

K2 internal state at time -17

(A0, A1, A2, A3, A4) = (0x83BD086B, 0x79AA791D, 0x54BF0EB7, 0xDA9BEC10, 0x3C2C0747)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x00000001, 0xE2636363, 0xE2636363, 0xE2636363, 0x08CS4DDD, 0xD448E338, 0xD791AF96, 0xDAC909B0,0xFFFA4197, 0xF546678B, 0x152D87ED)

(R1, L1, R2, L2) = (0x838EA823, 0x3D913FSO, 0x3ECF0A60, 0x3B05B5E9)

K2 internal state at time -16

(A0, A1, A2, A3, A4) = (0x79AA791D, 0x54BF0EB7, 0xDA9BEC10, 0x3C2C0747, 0xB46C6DA5)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xE2636363, 0xE2636363, 0xE2636363, 0x08CE'4DDD, 0xD44eE338, 0xD791AF96, 0xDAC909B0, 0xFFFA4197,0xF546678B, 0x152D87ED, 0x2417868F)

(R1, L1, R2, L2) = (0x28683EE5, 0x060AB10E, 0x72F97EE4, 0x4F560o9F)

K2 internal state at time -15

(A0, A1, A2, A3, A4) = (0x54BF0EB7, 0xDA9Bed0, 0x3C2C0747, 0xB46C6DA5, 0x8743F560)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xE2636363, 0xE2636363, 0x08CE4DDD, 0xD448E338, 0xD791AF96, 0xDAC90930, 0xFFFA4197, 0xF5466783,0x152D87ED, 0x2417868F, 0xA6305225)

(R1, L1, R2, L2) = (0xDF733A4C, 0x6295074A, 0xF3A16531, 0x971CE606)

K2 internal state at time -14

(A0, A1, A2, A3, A4) = (0xDA9Bed0, 0x3C2C0747, 0x346C6DA5, 0x8743F560, 0xEDBBC959)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xE2636363, 0xO8CE4DOD, 0xD448E338, 0xD791AF96, 0xDAC90930, 0xFFFA4197, 0xF546678B, 0x152D87ED,0x2417868F, 0xA6305225, 0x405CEA50)

(R1, L1, R2, L2) = (0xF96F4856, 0x5680685E, 0xFD52CF76, 0xC1A29363)

K2 internal state at time -13

(A0, A1, A2, A3, A4) = (0x30200747, 0xB46C6DA5, 0x8743F560, 0xEDBBC959, 0x3FD2F9E0)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x08CE4DDD, 0xD448E338, 0xD791AF96, 0xDAC909B0, 0xFFFA4197, 0xF546678B, 0x152D87ED, 0x2417868F,0xA6305225, 0x405CEA50, 0x726C92BC)

(R1, L1, R2, L2) = (0x2AF82CD3, 0x5A241664, 0x1B1BSFBE, 0x19542F03)

K2 internal state at time -12

(A0, A1, A2, A3, A4) = (0xB46C6DA5, 0x8743F560, 0xEDBBC959, 0x3FD2F9E0, 0xC0535EEC)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xD448E338, 0xD791AF96, 0xDAC909B0, 0xFFFA4197, 0xF546678B, 0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C923C, 0xD047818E)

(R1, L1, R2, L2) = (0xBDF 4 D6A5, 0x5B5798C6, 0x4BA1A2FB, 0xD3B129C7)

K2 internal state at time -11

(A0, A1, A2, A3, A4) = (0x8743F560, 0xEDBBC959, 0x3FD2F9E0, 0xC0535EEC, 0xB41BFCC9)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xD791AF96, 0xDAC909B0, 0xFFFA4197, 0xF546678B, 0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A)

(R1, L1, R2, L2) = (0x5A348B92, 0x221535B3, 0xB71B51C8 0xA80FECDB)

K2 internal state at time -10

(A0, A1, A2, A3, A4) = (0xEDBBC959, 0x3FD2F9E0, 0xC0535EEC, 0xB41BFCC9, 0x2C967031)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xDAC909B0, 0xFFFA4197, 0xF546678B, 0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC

0xD047818E, 0xA458FE0A, 0xED0E4419)

(R1, L1, R2, L2) = (0xB70F8DB4, 0x959902F7, 0x939BA37F, 0x45E722B1)

K2 internal state at time -9

(A0, A1, A2, A3, A4) = (0x3FD2F9E0, 0xC0535EEC, 0xB41BFCC9, 0x2c967031, 0xC54B3BD6)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xFFFA4197, 0xF546678B, 0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5)

(R1, L1, R2, L2) = (0x8EDD0383, 0x277464BD, 0xEEDC0439, 0x75A6858D)

K2 internal state at time -8

(A0, A1, A2, A3, A4) = (0xC0535EEC, 0xB41BFCC9, 0x2C967031, 0xC54B3BD6, 0x75146287)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xF546678B, 0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A 0xED0E4 419, 0x67C6A2D5, 0xB6E631E9)

(R1, L1, R2, L2) = (0x731DA313, 0xF37012AC, 0xA7255B96, 0xDC499D6F)

K2 internal state at time -7

(A0, A1, A2, A3, A4) = (0xB41BFCC9, 0x2C967031, 0xC54BBBD6, 0x75146237, 0x83B8D1B2)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x152D87ED, 0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419 0x67C6A2D5, 0xB6E531E9, 0xA64D395C)

(R1, L1, R2, L2) = (0xD8B057FC, 0xC761F332, 0x2CAE11CF, 0x2AEDE625)

K2 internal state at time -6

(A0, A1, A2, A3, A4) = (0x2C967031, 0xC54BBBD6, 0x75146287, 0x83B8D1B2, 0xF1D21A26)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x2417868F, 0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5 0xB6E631E9, 0xA64D395C, 0x7E59587F)

(R1, L1, R2, L2) = (0xE885CA11, 0x015D89D8, 0xB59D5510, 0x10BAD578)

K2 internal state at time -5

(A0, A1, A2, A3, A4) = (0xC54B3BD6, 0x75146287, 0x83B8D1B2, 0xF1D21A26, 0xD3884C64)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xA6305225, 0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5, 0xB6E631E9

0xA64D395C, 0x7B59587F, 0xAFA3772A)

(R1, L1, R2, L2) = (0x0A63FCEF, 0xB78DD0E9, 0x5375538F, 0xB0DA9C00)

K2 internal scate at time -4

(A0, A1, A2, A3, A4) = (0x75146287, 0x83B8D1B2, 0xF1D21A26, 0xD3884C64, 0x3EC047BA)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x405CEA50, 0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5, 0xB6E631E9, 0xA64D395C 0x7B59587F, 0xAFA3772A, 0x83AE3259)

(R1, L1, R2, L2) = (0xF35398D7, 0xA38AB94C, 0xFF2BD5F2, 0x4634B058)

K2 internal state at time -3

(A0, A1, A2, A3, A4) = (0>;83B8D1B2, 0xF1D21A25, 0xD3884C64, 0x3EC047BA, 0x89EF93D3)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x726C92BC, 0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5, 0xB6E631E9, 0xA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, 0x479E0649)

(R1, L1, R2, L2) = (0x4F2CD5EE, 0xD8DB21E5, 0xB5B29920, 0x479D0DDC)

K2 internal state at time -2

(A0, A1, A2, A3, A4) = (0xF1D21A26, 0xD3884C64, 0x3EC047BA, 0x89EF93D3, 0x3A5FD7EB)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xD047818E, 0xA458FE0A, 0xED0E4419, 0x67C6A2D5, 0x36E631E9, 0xA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, 0x479E0649, 0xBAC82EA9)

(R1, L1, R2, L2) = (0x25C35580, 0x36845CF8, 0x195E39A0, 0xF6EE896D)

K2 internal state at time -1

(A0, A1, A2, A3, A4) = (0xD3884C64, 0x3EC0473A, 0x89EF93D3, 0x3A5FD7EB, 0xC74844F8)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xA458FE0A, 0xEDOE4419, 0x67C6A2D5, 0xB6E631E9, 0xA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, 0x479E0649, 0x3AC82EA9, 0x3AB6E5CC)

(R1, L1, R2, L2) = (0xBA38B6DF, 0x767E04BF, 0xE02C638F, 0xDCBA3106)

K2 internal state at time 0

(A0, A1, A2, A3, A4) = (0x3SCC)47BA, 0x89EF93D3, 0x3A5FD7EB, 0xC74844F8, 0xECB10CC9)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xED0E4419, 0x67C6A2D5, 0x36E631E9, 0xA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, Cx479E0649, 0XBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C)

(R1, L1, R2, L2) = (0x5C134123, OxCOSCCB42, 0x03D9FF06, 0x694FC1D6)

K2 internal state at time 1

(A0, A1, A2, A3, A4) = (0x89EF93D3, 0x3A5FD7EB, 0xC74844F8, 0xECB10CC9, 0x9F3C9CD1)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x67C6A2D5, 0xB6E631E9, 0xA64D395C, 0x7359587F, 0xAFA3772A, 0x83AE3259, 0x479EG649, 0xBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575)

(R1, L1, R2, L2) = (0xEF637AB6, 0x853CADC9, Ox0031F6E5, 0x602E04A7)

K2 internal state at time 2

(AO, A1, A2, A3, A4) = (0x3A5FD7EB, 0xC74844F8, OxECB1OCC9, 0x9F3C9CD1, OxO7E0A49D)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xB6E631E9, 0xA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, Ox479EOS49, 0xBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575, 0x3ABC9602)

(R1, L1, R2, L2) = (0x4C41E3 30, 0xEFA5F58E, 0x560328CD, 0x37275D79)

K2 internal state at time 3

(A0, A1, A2, A3, A4) = (0xC74844F8, OxECB10CC9, 0x9F3C9CD1, 0x07E0A49D, 0xC5F86290)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(OxA64D395C, 0x7B59587F, 0xAFA3772A, 0x83AE3259, 0x479EO649, 0xBAC82EA9, 0x3A36E5CC, 0xF3E9F17C, 0xFE14A575, 0x3ABC9602, 0x04FD08B2)

(R1, L1, R2, L2) = (OxD3 33 7 6B4, 0x991AF343, 0xCC739191, 0x6E891BDA)

K2 internal state at time 4

(A0, A1, A2, A3, A4) = (0xECB10CC9, 0x9F3C9CD1, 0x07E0A49D, 0xC5F86290, 0x4453180A)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x7B59587F, 0xAFA3772A, 0x83AE3259, 0x479E0649, 0xBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575, 0x3ABC9602, 0x04FD08B2, OxDE821E38)

(R1, L1, R2, L2) = (0x8AFC6CE9, 0x1CD53B55, 0xBB82C5EC, 0x4661136F)

K2 internal state at time 5

(A0, A1, A2, A3, A4) = (0x9F3C9CD1, Ox07E0A49D, 0 ^5F86290, 0x4453180A, 0xFD98883F)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0xAFA3772A, 0x83AE3259, 0x479E0649, 0xBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575, 0x3ABC9602,0x04FD08B2, 0xDE821E38, 0xD30B1637)

(R1, L1, R2, L2) = (0xA51F85B5, Qx676A1660, 0x3EB70BQ2, 0xDDA7BA41)

K2 internal state at time 6

(A0, A1, A2, A3, A4) = (0x07E0A49D, 0xC5F86290, 0x4453180A, 0xFD98883F, 0x7FBD0061)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x83AE3259, 0x479EO649, 0xBAC82EAS, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575, 0x3ABC9602, 0x04FD08B2,0xDE821E38, OxD3QB1637, 0x1AE1594D)

(R1, L1, R2, L2) = (0xCF29E514, 0x41BE7A08, 0x3FD3BDD5, 0x3F07DDF5)

K2 internal state at time 7

(A0, A1, A2, A3, A4) = (0xC5F86290, 0x4453180A, 0xFD98883F, 0x7FBD0061, 0x99O4D579)

(B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10) =

(0x479EO649, 0xBAC82EA9, 0x3AB6E5CC, 0xFBE9F17C, 0xFE14A575, 0x3ABC9602, 0x04FD08B2, 0xDE821E39,0xD30B1637, 0x1AE1594D, 0x38FE9448)

(R1, L1, R2, L2) = (0x6C07 9D38, 0xB4614FAA, 0x66F72DB0, 0x3933F538)

Phụ lục D

(Tham khảo)

Thông tin an toàn

D.1 Mức độ an toàn của mã dòng

Phụ lục này liệt kê mức độ an toàn của mã dòng được mô tả trong tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033).

Bảng D.1-Mức độ an toàn của các chế độ mã dòng

Hàm đầu ra	Bí mật dữ liệu	Toàn vẹn dữ liệu
Chế độ cộng nhị phân	Một cơ chế mã hóa dựa trên các hàm đầu ra là an toàn miễn là bộ tạo khóa dòng an toàn. Điều này được chứng minh bằng toán học.	Chế độ này không cung cấp an toàn liên quan đến toàn vẹn dữ liệu.
Chế độ MULTI-S01		Việc an toàn liên quan đến toàn vẹn dữ liệu thông thường an toàn như bộ tạo khóa dòng. Tuy nhiên mức độ an toàn luôn luôn bị chặn trên bởi chiều dài của thông báo giả mạo. Một thông báo giả mạo có chiều dài là un bit có thể được chấp nhận với xác suất thành công $(u-2)2^{-n}$, trong đó n là tham số an toàn của chế độ này.

Bảng D.2-Mức độ an toàn của bộ tạo khóa dòng chuyên dụng

Bộ tạo khóa dòng	Tuyên bố an toàn	Độ dài khóa	Độ phức tạp tính toán của tấn công tốt nhất được biết đến
Chế độ CFB, OFB, CTR với mã khối n-bit với độ dài khóa k-bit	Giả sử rằng các mã khối về cơ bản là an toàn, nói chung không thể phân biệt chế độ CFB, OFB và CTR từ một chuỗi ngẫu nhiên. Tuy nhiên, nếu số lượng các khối được xử lý vượt quá khoảng $2^{n/2}$ thì xác suất của phân biệt giữa khóa dòng và chuỗi ngẫu nhiên trở nên đáng kể. Sự an toàn của khóa dòng được chặn trên bởi an toàn của mã khối, chẳng hạn độ dài khóa và mức độ mật mã có thể được phân tích mã.	k- bit	$2^{n/2}$
MUGI	Không có một tấn công phân tích mã nào được biết đến đối với MUGI nhanh hơn so với tấn	128-bit	2^{128}

	công vét cạn khóa. MUGI có độ dài khóa là 128-bit.		
SNOW 2.0	Đối với chế độ 256-bit, các thuật toán phân biệt đã được phân biệt đầu ra của SNOW 2.0 từ một chuỗi ngẫu nhiên. Cách tốt nhất để ước tính và cần biết khoảng 2^{174}-bit khóa dòng. Điều này hầu như không vi phạm mức độ an toàn trên lý thuyết, có thể được suy ra từ độ dài khóa, 256-bit. Tuy nhiên, bất kỳ tấn công nào liên quan đến phân biệt đã biết sẽ không có mối đe dọa nào với sử dụng thực tế.	128 hoặc 256-bit	2^{174}
Rabbit	Không có tấn công phân biệt nào được biết đến đối với với Rabbit nhanh hơn so với tấn công vét cạn khóa, miễn là không quá 2^{64} khối khóa dòng được tạo ra sử dụng một khóa.	128 bit	2^{128}
$Decim^{v2}$	Không có một tấn công phân tích mã nào được biết đến đối với $Decim^{v2}$ nhanh hơn so với tấn công vét cạn khóa.	80 bit	2^{80}
KCipher-2 (K2)	Không có một tấn công phân tích mã nào được biết đến đối với K2 nhanh hơn so với tấn công vét cạn khóa. K2 có độ dài khóa là 128-bit	128 bit	2^{128}

D.2 An toàn hiệu quả đánh đổi trong MULTI-S01

Cho n là kích thước khối của hàm đầu ra MULTI-S01. Với thông báo $n.u$-bit, hàm đầu ra MULTI-S01 lặp xử lý khối $u+2$ lần. Trong cả hai phần mềm và phần cứng thực hiện, việc tính toán chi phối là các phép nhân trong trường hữu hạn $GF(2^n)$.

Nếu hai thực thi, giữa kích thước khối n khác nhau, được so sánh, việc thực thi với n nhỏ hơn nói chung là nhanh hơn khối khác – mặc dù việc thực hiện với giá trị n nhỏ hơn lặp nhiều lần hơn, mỗi lần nhân có thể được tính trong thời gian ít hơn và sử dụng không gian nhỏ hơn. Các yếu tố mà việc tính toán tăng tốc độ lên phụ thuộc vào thuật toán và nền tảng được sử dụng để thực thi nó.

Bảng D.3 – cho thấy một số kết quả thực nghiệm của việc thực thi MULTI-S01 với $n = 64$.

Bảng D.3 – Kết quả thực nghiệm của việc thực thi chế độ MULTI-S01 không có bộ tạo khóa dòng ($n = 64$)

Chọn lựa	Tốc độ/Kích thước	Đặc tả nền tảng
ASIC tốc độ cao	5.1 Gbps@80MHz, 25.7 K Cổng	ASIC sử dụng thư viện ngăn Hitachi HG73C (0.35 μm)
ASIC kích thước nhỏ	2.0 K Cổng, 100Mbps@100MHz	
Phần mềm	10 chu kỳ/byte (hoặc tương đương 520 bps@650 MHz)	Intel Pentium III 650 MHz (Coppermine), Windows98 SE, RAM 64MB, Visual C++

		Ver.6.0 Service Pack 3.

D.3 Hướng dẫn về mã dòng

Phụ lục này liệt kê các tính năng của mã dòng được mô tả trong tiêu chuẩn này của bộ TCVN 11367 (ISO/IEC 18033).

Bảng D.4 – Đặc điểm của mã dòng

Bộ tạo khóa dòng	Phát biểu thuộc tính
Chế độ CFB, OFB, CTR với mã khối n-bit với độ dài khóa k-bit	Thuận lợi của 3 chế độ này là chúng chia sẻ các thành phần với các chế độ khác của mã khối. Các chế độ này có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một vài kiến trúc, và OFB và CTR là hai chế độ phù hợp cho việc mã hóa và giải mã bản rõ độ dài nx-bit. CFB có thể khôi phục lỗi đồng bộ và hiệu năng của nó phụ thuộc vào độ dài khoa dòng. CTR chấp nhận truy cập ngẫu nhiên đến bản mã.
MUGI	MUGI sử dụng véc tơ khởi tạo độ dài 128-bit. Nó bao gồm các phép toán 64-bit và tạo ra 64-bit khóa dòng cho mỗi lần thực hiện. Mật mã có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một số kiến trúc, và nó hiệu quả hơn cho kiến trúc CPU 64-bit và mã hóa/giải mã bản rõ độ dài $64x$.
SNOW 2.0	SNOW 2.0 sử dụng véc tơ khởi tạo độ dài 128-bit. Nó bao gồm các phép toán 32-bit và tạo ra 32-bit khóa dòng cho mỗi lần thực hiện. Mật mã có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một số kiến trúc và nó hiệu quả hơn cho kiến trúc CPU 32-bit và mã hóa/giải mã bản rõ độ dài $32x$.
Rabbit	Rabbit sử dụng véc tơ khởi tạo độ dài 64-bit. Nó bao gồm các phép toán 32-bit và tạo ra 128-bit khóa dòng cho mỗi lần thực hiện. Mật mã có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một số kiến trúc và nó hiệu quả hơn cho kiến trúc CPU 32-bit và mã hóa/giải mã bản rõ độ dài $128x$.
$Decim^{v2}$	$Decim^{v2}$ sử dụng véc tơ khởi tạo độ dài 64-bit. Nó bao gồm các phép toán từng bit và tạo ra 1-bit khóa dòng cho mỗi lần thực hiện. Mật mã có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một số kiến trúc và nó hiệu quả hơn cho thực hiện phần cứng trên các thiết bị có tài nguyên hạn chế.
KCipher-2 (K2)	KCipher-2 sử dụng véc tơ khởi tạo độ dài 128-bit. Nó bao gồm các phép toán 32-bit và tạo ra 64-bit khóa dòng cho mỗi lần thực hiện. Mật mã có thể được sử dụng cho bản rõ có độ dài bất kỳ trên một số kiến trúc, và nó hiệu quả hơn cho kiến trúc CPU 32-bit và mã hóa/giải mã bản rõ độ dài $64x$.

Thư mục tài liệu tham khảo

[1] ISO/IEC 9797-1:2011, *Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher*

[2] ISO/IEC 10116:2006, *Information technology — Security techniques — Modes of operation for an n-bit block cipher*

[3] ISO/IEC 10118-1:2000, *Information technology — Security techniques — Hash-functions — Part 1: General*

[4] ISO/IEC 11770-1:2010, *Information technology — Security techniques — Key management — Part 1: Framework*

[5] ISO/IEC 11770-3:2008, *Information technology — Security techniques — Key management — Part 3: Mechanisms using asymmetric techniques*

[6] Berbain, C., Billet, O., Canteaut, A., Courtois, N., Debraize, B., Gilbert, H., Goubin, L., Gouget, A., Granboulan, L., Lauradoux, C., Minier, M., Pornin, T. and Sibert, H., "DECIMv2, a compact hardware- oriented stream cipher", SASC 2006 - Stream Ciphers revisited Workshop, Leuven, Belgium, 2006

[7] Biryukov, A. and Shamir, A., "Cryptanalytic Time/Memory/Data Tradeoffs for Stream Ciphers", Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 2000, Proceedings, ed. Okamoto, T., Lecture Notes in Computer Science vol. 1976, Springer-Verlag, pp.1-13, 2000

[8] Boesgaard, M., Vesterager, M., Pedersen, T., Christiansen, J., and Scavenius, O., "Rabbit: A new high- performance stream cipher". In T. Johansson, editor, Proc. Fast Software Encryption 2003, Lecture Notes in Computer Science vol.2887, Springer-Verlag, pp.307-329, 2003

[9] Boesgaard, M.(Vesterager, M., Christensen, T., Zenner, E., "The Rabbit stream cipher - design and security analysis". Available from <http://www.cryptico.com/files/filer/rabbit sasc final.pdf>

[10] Ekdahl, P. and Johansson, T., "A new version of the stream cipher SNOW", Selected Areas in Cryptography, 9th Annual Workshop, SAC 2002, St. John's, Newfoundland, Canada, Aug. 2002, Revised Papers, eds. Nyberg, K. and Heys, H., Lecture Notes in Computer Science vol. 2595, Springer- Verlag, pp.47-61, 2002

[11] Furuya, S., Watanabe, D., Seto, Y., and Takaragi, K., "Integrity-Aware Mode of Stream Cipher," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science vol. E85-A No.1, pp.58-65, 2002

[12] Kiyomoto, S., Tanaka, T., and Sakurai, K., "A Word-Oriented Stream Cipher Using Clock Control", In SASC 2007 Workshop Record, pp.260-274, January, 2007

[13] Kiyomoto, S., Tanaka, T., and Sakurai, K., "K2: A Stream Cipher Algorithm Using Dynamic Feedback Control", In Proc. of SECRYPT 2007, pp.204-213, July, 2007

[14] Kiyomoto, S., Tanaka, T., and Sakurai, K., "K2 Stream Cipher", Communications in Computer and Information Science, E-business and Telecommunications, 4th International Conference, ICETE 2007, Barcelona. Spain, July 28-31, 2007, Revised Selected Papers, pp.14-226

[15] Menezes, A.J., van Oorschot, P.C., and Vanstone, S.A., Handbook of Applied Cryptography, CRC Press, 1996

[16] Nyberg, K., Wallen, J., "Improved Linear Distinguishers for SNOW 2.0", FSE 2006, Lecture Notes in Computer Science vol.4047, Springer-Verlag, pp.44-162, 2006

[17] Watanabe, D., Furuya, S., Yoshida, H., Takaragi, K., and Preneel, B.t "A New Key Stream Generator MUGI," Fast Software Encryption, 9th International Workshop, FSE 2002, Leuven, Belgium, February 4-6, 2002, Revised Papers, eds. Daemen, J. and Rijmen, V., Lecture Notes in Computer Science vol.2365, Springer-Verlag, pp.179-194, 2002

[18] Wu, H. and Preneel, B., "Cryptanalysis of the Stream Cipher DECIM", Proc. FSE 2006, Lecture Notes in Computer Science vol. 4047, Springer-Verlag, pp.30-40, 2006

[19] ISO/IEC 9834 (all parts), Information technology — Open Systems Interconnection — Procedures for the operation of OS I Registration Authorities