ＦＥＡＬ－ＮＸ仕様書

１　序

１．１　ＦＥＡＬ－ＮＸ暗号の概要

ＦＥＡＬ（the Fast Data Encipherment Algorithm，高速データ暗号アルゴリズム）は，６４ビットブロック暗号アルゴリズムであり，６４ビットの平文を６４ビットの暗号文に暗号化し，また，６４ビットの暗号文を６４ビットの平文に復号化する．

ＦＥＡＬは3つのオプション，鍵の長さ，回転数，鍵パリティ，を持つ．鍵の長さは，６４ビットか，１２８ビットであり，回転数（N）は，データランダム化のための内部回転数を指定し，鍵パリティオプションは，鍵ブロックにおけるパリティビットを使うか，使わないかを選択する．ここではＦＥＡＬ－ＮＸ暗号をＦＥＡＬ暗号の鍵パリティビットなし１２８ビット鍵N回転限定の版と位置付ける．

１．２　ＦＥＡＬ－ＮＸオプション

ＦＥＡＬ－ＮＸオプションは次のように定義される．

（１）回転数N
　回転数Nは，ＦＥＡＬデータランダム化部の回転数(N)を指定する．Nは，N≧３２，偶数である．

１．３　定義と記法

１．３．１　定義

（１）鍵ブロック：１２８ビットブロックである.

（２）鍵：暗号化／復号化に用いる鍵情報

（３）回転数（N）：ＦＥＡＬデータランダム化部の内部回転数

（４）拡大鍵：１６ビットブロック，K_i，ｉ＝0, 1, 2, ‥,(N+7). 拡大鍵は，鍵が，ＦＥＡＬ鍵処理部によりランダム化されて拡大され生成される.

１．３．２　記法

（１）A，A_r，…：　ブロック（長さは、各所で定義される）

（２）（A，B，……）：　この順序の連結（concatenation）

（３）A⊕B：　ブロックＡとＢのビット対応の排他的論理和

（４）φ　：　３２ビット長の零ブロック

（５）等号 ＝ ：　右辺を左辺に代入する

（６）ビット位置：　ブロックの最左端ビット（MSB）から右方向に１，２，３，‥，と数える.

２　暗号化手続き

２．１　計算ステージ

この暗号化手続きの中で使われる拡大鍵K_iは，４節で記載される鍵処理により生成される．同様に，ここで使われる関数ｆは，５節で定義される．計算ステージは次の通りであり，詳細は２．２節から２．４節により詳細に記載される（図１も参照)．

ａ）前処理（２．２節）

ｂ）反復計算（２．３節）

ｃ）後処理（２．４節）

２．２　前処理

平文ブロックＰは, 同じ長さ（３２ビット）のブロック，L_0, R_0に分けられる．即ち，（L_0，R_0）＝Ｐ．

初めに,

$$（L_0，R_0）＝（L_0，R_0）\oplus（K_N，K_{N+1}，K_{N+2}，K_{N+3}）$$

次に,

$$（L_0，R_0）＝（L_0，R_0）\oplus（\phi，L_0）$$

２．３　反復計算

（L_0，R_0）を入力し，次の計算を，ｒ＝１～Ｎについて反復して計算する．

$$R_r = L_{r-1} \oplus f（R_{r-1}，K_{r-1}）$$
$$L_r = R_{r-1}$$

最終ラウンドの出力は,（L_N，R_N）である．

２．４　後処理

反復計算の最終出力（L_N，R_N）の左右を交換して,（R_N，L_N）とする．

次に,

$$(R_N,\ L_N) = (R_N,\ L_N) \oplus (\phi,\ R_N)$$

最後に,

$$(R_N,\ L_N) = (R_N,\ L_N) \oplus (K_{N+4},\ K_{N+5},\ K_{N+6},\ K_{N+7})$$

暗号文ブロックは（R_N, L_N）で得られる.

3　復号化手続き

3．1　計算ステージ

この復号化手続きの中で使われる拡大鍵K_iは，4節で記載される鍵処理により生成される．この手続きで使われる関数fは，5節で定義される．計算ステージは次の通りであり（図1も参照），詳細は3．2節から3．4節により詳細に記載される.

a）前処理（3．2節）
b）反復計算（3．3節）
c）後処理（3．4節）

3．2　前処理

暗号文ブロック（R_N, L_N）は，同じ長さ（32ビット）のブロック，R_N, L_N に分けられる.

初めに,

$$(R_N,\ L_N) = (R_N,\ L_N) \oplus (K_{N+4},\ K_{N+5},\ K_{N+6},\ K_{N+7})$$

次に,

$$(R_N,\ L_N) = (R_N,\ L_N) \oplus (\phi,\ R_N)$$

3．3　反復計算

（R_N，L_N）を入力し，次の計算を，ｒ＝Ｎ～１について反復して計算する．

$$L_{r-1} = R_r \oplus f(L_r, K_{r-1})$$
$$R_{r-1} = L_r$$

最終ラウンドの出力は，（R_0，L_0）である．

３．４　後処理

反復計算の最終出力（R_0，L_0）の左右を交換して，（L_0，R_0）とする．

次に，

$$(L_0, R_0) = (L_0, R_0) \oplus (\phi, L_0)$$

最後に，

$$(L_0, R_0) = (L_0, R_0) \oplus (K_N, K_{N+1}, K_{N+2}, K_{N+3})$$

平文ブロックは（L_0，R_0）として得られる．

４　鍵処理

４．１　ＦＥＡＬ－ＮＸの鍵処理

ＦＥＡＬ－ＮＸの鍵処理を記述する（図２を参照）．ここで利用する関数は，５節で定義する．鍵処理では，１２８ビットの鍵から，拡大鍵K_i（ｉ＝ 0, 1, 2, 3, ‥，Ｎ＋7）を生成する．

４．１．１　左鍵K_Lと右鍵K_Rの定義

入力した１２８ビットの鍵は，同じ長さの６４ビットずつに分ける，K_Lが，６４ビットの左鍵であり，K_Rが６４ビットの右鍵である．即ち，（K_L，K_R）が入力した１２８ビット鍵である．

４．１．２　反復計算

（１）右鍵K_Rの処理

右鍵K_Rを３２ビットずつ２等分し，その左３２ビットをK_{R1}，その右３２ビットをK_{R2}とおき（即ち，（K_{R1}，K_{R2}）＝K_R），一時変数Q_rを定義する．

$Q_r = K_{R1} \oplus K_{R2}$　　for r = 1, 4, 7...,　　(r = 3i+1; i = 0, 1, ...)
$Q_r = K_{R1}$　　for r = 2, 5, 8...,　　(r = 3i+2; i = 0, 1, ...)
$Q_r = K_{R2}$　　for r = 3, 6, 9...,　　(r = 3i+3; i = 0, 1, ...)

但し，1≦ r ≦（N／2）＋４，　（N≧３２，N：偶数）

（２）左鍵K_Lの処理

左鍵K_Lの左半分をA_0，右半分B_0とする．即ち，（A_0，B_0）＝K_L.
最初に，D_0 ＝ φとする．続いて，ｒ＝１～（N／２）＋４について，K_i（ｉ＝０～（N＋７））を定める．

$D_r = A_{r-1}$
$A_r = B_{r-1}$
$B_r = f_K(\alpha, \beta)$
$= f_K(A_{r-1}, (B_{r-1} \oplus D_{r-1}) \oplus Q_r)$
$K_{2(r-1)} = (B_{r0}, B_{r1})$
$K_{2(r-1)+1} = (B_{r2}, B_{r3})$

ここで，A_r，B_r，D_r，Q_rは補助変数である．
但し，（B_{r0}，B_{r1}，B_{r2}，B_{r3}）＝B_r，B_{rj}（ｊ＝０～３）は１バイト長ブロック．

鍵処理段数は,（N／２）＋４で与えられる．A_r，B_r，D_rは，補助変数である

５　関数

２，３，４節で使われる関数を定義する.

５．１　関数ｆ（図３参照）

ｆ（α，β）をｆ，αを（α_0，α_1，α_2，α_3），βを（β_0，β_1）と表す.

ｆ（つまりf_0，f_1，f_2，f_3）は，次の順序により計算する．ここで，α_i（ｉ＝０～３），β_i（ｉ＝０～１）は，１バイト長ブロックである．関数S_0とS_1は５．３節に定義される.

$f_1 = \alpha_1 \oplus \beta_0$
$f_2 = \alpha_2 \oplus \beta_1$
$f_1 = f_1 \oplus \alpha_0$
$f_2 = f_2 \oplus \alpha_3$
$f_1 = S_1(f_1, f_2)$
$f_2 = S_0(f_2, f_1)$
$f_0 = S_0(\alpha_0, f_1)$
$f_3 = S_1(\alpha_3, f_2)$

例：入力：α＝ 00 FF FF 00,　β＝ FF FF,　出力：ｆ＝ 10 04 10 44

５．２　関数f_K（図４参照）

f_K（α，β）をf_K，αを（α_0，α_1，α_2，α_3），βを（β_0，β_1，β_2，β_3）と表す．ここで，α_i（ｉ＝０～３），β_i（ｉ＝０～１）は，１バイト長ブロックである.

f_K（つまりf_{K0}，f_{K1}，f_{K2}，f_{K3}）は，次の順序により計算する．関数S_0とS_1は５．３節に定義される.

$f_{K1} = \alpha_1 \oplus \alpha_0$

$f_{K2} = \alpha_2 \oplus \alpha_3$

$f_{K1} = S_1(f_{K1}, (f_{K2} \oplus \beta_0))$

$f_{K2} = S_0(f_{K2}, (f_{K1} \oplus \beta_1))$

$f_{K0} = S_0(\alpha_0, (f_{K1} \oplus \beta_2))$

$f_{K3} = S_1(\alpha_3, (f_{K2} \oplus \beta_3))$

例：入力：α = 00 00 00 00, β = 00 00 00 00, 出力：f_K = 10 04 10 44

５．３　関数Ｓ

S_0とS_1は以下の様に定義される.

$S_0(X_1, X_2) = \text{Rot2}((X_1 + X_2) \bmod 256)$

$S_1(X_1, X_2) = \text{Rot2}((X_1 + X_2 + 1) \bmod 256)$

X_1, X_2は，１バイトブロックである. Rot2（Ｔ）は，１バイトブロックＴを，２ビット左循環シフト（左２ビット回転）して得られる１バイトブロックである.

例：　X_1 = 00010011, X_2 = 11110010　のとき,

$T = (X_1 + X_2 + 1) \bmod 256$ = 00000110

$S_1(X_1, X_2)$ = Rot2（Ｔ）= 00011000

６．　動作例

動作データはビット系列と１６進数により示される.

６．１　ＦＥＡＬ－ＮＸのオプション（１．２節参照）

この例では，次のＦＥＡＬのオプションが選択される.

（１）回転数　Ｎ＝３２

６．２　入力データ

入力ブロックは，鍵ブロックと平文ブロックである．

鍵ブロックKは：

K＝ 0000 0001 0010 0011 0100 0101 0110 0111
1000 1001 1010 1011 1100 1101 1110 1111
0000 0001 0010 0011 0100 0101 0110 0111
1000 1001 1010 1011 1100 1101 1110 1111（ビット系列）

K＝ 01 23 45 67 89 AB CD EF
01 23 45 67 89 AB CD EF （１６進数）

平文Pは：

P＝ 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000（ビット系列）

P＝ 00 00 00 00 00 00 00 00 （１６進数）

６．３　鍵処理（４節参照）

まず，拡大鍵，K_0，K_1，K_2，‥‥，K_{39}の生成を考える．それぞれは，鍵ブロックKから生成され，１６ビットである．

６．３．１　反復計算（４．１．２節参照）

A_0をK_Lの左半分，B_0をK_Lの右半分とする．即ち，

K_L=(A_0，B_0)　D_0＝φ．　従って，

A_0＝ 0000 0001 0010 0011 0100 0101 0110 0111 （ビット系列）
＝ 01 23 45 67 （１６進数）

B_0＝ 1000 1001 1010 1011 1100 1101 1110 1111 （ビット系列）
＝ 89 AB CD EF （１６進数）

D_0＝ 0000 0000 0000 0000 0000 0000 0000 0000 （ビット系列）
＝ 00 00 00 00 （１６進数）

D_1，A_1，B_1，K_0，K_1を次のように計算する．

$D_1 = A_0 =$ 0000 0001 0010 0011 0100 0101 0110 0111
（ビット系列）
$=$ 01 23 45 67 （１６進数）
$A_1 = B_0 =$ 1000 1001 1010 1011 1100 1101 1110 1111
（ビット系列）
$=$ 89 AB CD EF （１６進数）
$B_1 = f_K(A_0, B_0 \oplus D_0)$
$=$ 0111 0101 0001 1001 0111 0001 1111 1001
（ビット系列）
$=$ 75 19 71 F9 （１６進数）
$K_0 =$ 0111 0101 0001 1001 （ビット系列）
$=$ 75 19 （１６進数）
$K_1 =$ 0111 0001 1111 1001 （ビット系列）
$=$ 71 F9 （１６進数）

この手続きを続け，拡大鍵K_iが１６進数で次のように得られる．

$K_0 =$ 75 19	$K_1 =$ 71 F9	$K_2 =$ 84 E9	$K_3 =$ 48 86
$K_4 =$ 88 E5	$K_5 =$ 52 3B	$K_6 =$ 4E A4	$K_7 =$ 7A DE
$K_8 =$ FE 40	$K_9 =$ 5E 76	$K_{10} =$ 98 19	$K_{11} =$ EE AC
$K_{12} =$ 1B D4	$K_{13} =$ 24 55	$K_{14} =$ DC A0	$K_{15} =$ 65 3B
$K_{16} =$ 3E 32	$K_{17} =$ 46 52	$K_{18} =$ 1C C1	$K_{19} =$ 34 DF
$K_{20} =$ 77 8B	$K_{21} =$ 77 1D	$K_{22} =$ D3 24	$K_{23} =$ 84 10
$K_{24} =$ 1C A8	$K_{25} =$ BC 64	$K_{26} =$ A0 DB	$K_{27} =$ BD D2
$K_{28} =$ 1F 5F	$K_{29} =$ 8F 1C	$K_{30} =$ 6B 81	$K_{31} =$ B5 60
$K_{32} =$ 19 6A	$K_{33} =$ 9A B1	$K_{34} =$ E0 15	$K_{35} =$ 81 90
$K_{36} =$ 9F 72	$K_{37} =$ 66 43	$K_{38} =$ AD 32	$K_{39} =$ 68 3A

６．４　暗号化アルゴリズム（２節参照）

６．４．１　前処理（２．２節参照）

Ｐ＝ 0000 0000 0000 0000 0000 0000 0000 0000
　　0000 0000 0000 0000 0000 0000 0000 0000　（ビット系列）
Ｐ＝ 00 00 00 00 00 00 00 00　（１６進数）

Ｐは，３２ビット長の，L_0とR_0とに分けられる．
（L_0，R_0）＝（L_0，R_0）⊕（K_{32}，K_{33}，K_{34}，K_{35}）
　　＝ 0001 1001 0110 1010 1001 1010 1011 0001
　　1110 0000 0001 0101 1000 0001 1001 0000
　　（ビット系列）
　　＝ 19 6A 9A B1 E0 15 81 90　（１６進数）
　次に，
（L_0，R_0）＝（L_0，R_0）⊕（φ，L_0）
　　＝ 0001 1001 0110 1010 1001 1010 1011 0001
　　1111 1001 0111 1111 0001 1011 0010 0001
　　（ビット系列）
　　＝ 19 6A 9A B1 F9 7F 1B 21　（１６進数）

L_0 ＝ 0001 1001 0110 1010 1001 1010 1011 0001　（ビット系列）
　＝ 19 6A 9A B1　（１６進数）
R_0 ＝ 1111 1001 0111 1111 0001 1011 0010 0001　（ビット系列）
　＝ F9 7F 1B 21　（１６進数）

６．４．２　反復計算（２．３節参照）

６．４．２．１　第１ステージにおけるR_0とL_0の計算
　最初に，ｆ（R_0，K_0）を計算する．
　ｆ（R_0，K_0）
　＝ 0101 0101 0101 1100 1111 1101 0111 1100　（ビット系列）
　＝ 55 5C FD 7C　（１６進数）
ここで詳細は，６．４．２．２で記述される．
　L_0⊕ｆ（R_0，K_0）＝ 4C 36 67 CD　（１６進数）

反復計算のステージの出力は,

$L_1 = R_0$ = 1111 1001 0111 1111 0001 1011 0010 0001
（ビット系列）
= F9 7F 1B 21 （１６進数）

R_1= 0100 1100 0011 0110 0110 0111 1100 1101 （ビット系列）
= 4C 36 67 CD （１６進数）

６．４．２．２ 第１ステージの計算

次のｆ（R_0, K_0）の計算において，ｆ（R_0, K_0）は短縮してｆとし，αとβは，次のように定義する.

$\alpha = (\alpha_0, \alpha_1, \alpha_2, \alpha_3) = R_0$
= 1111 1001 0111 1111 0001 1011 0010 0001 （ビット系列）
= F9 7F 1B 21 （16進数）

$\beta = (\beta_0, \beta_1) = K_0$
= 0111 0101 0001 1001 （ビット系列）
= 75 19 （１６進数）

α_0=1111 1001= F9 （１６進数), α_1=0111 1111= 7F （１６進数）
α_2=0001 1011= 1B （１６進数), α_3=0010 0001= 21 （１６進数）
β_0=0111 0101= 75 （１６進数), β_1=0001 1001= 19 （１６進数）

ｆ（つまりf_0, f_1, f_2, f_3）は逐次計算する.

$f_1 = \alpha_1 \oplus \beta_0$= 0000 1010 = 0A （１６進数）
$f_2 = \alpha_2 \oplus \beta_1$= 0000 0010 = 02 （１６進数）
$f_1 = f_1 \oplus \alpha_0$= 1111 0011 = F3 （１６進数）
$f_2 = f_2 \oplus \alpha_3$= 0010 0011 = 23 （１６進数）
$f_1 = S_1(f_1, f_2)$ = 0101 1100 = 5C （１６進数）
$f_2 = S_0(f_2, f_1)$ = 1111 1101 = FD （１６進数）
$f_0 = S_0(\alpha_0, f_1)$ = 0101 0101 = 55 （１６進数）
$f_3 = S_1(\alpha_3, f_2)$ = 0111 1100 = 7C （１６進数）

６．４．２．３　続く計算

上記の計算を継続すると，L_iとR_iが次のように与えられる（１６進数表示、ただしｉは１０進数表示）.

処理ステージ

i	L_i	R_i	K_{i-1}	$f(R_{i-1}, K_{i-1})$
0	19 6A 9A B1	F9 7F 1B 21		
1	F9 7F 1B 21	4C 36 67 CD	75 19	55 5C FD 7C
2	4C 36 67 CD	DE 02 58 65	71 F9	27 7D 43 44
3	DE 02 58 65	06 82 45 EF	84 E9	4A B4 22 22
4	06 82 45 EF	69 E5 14 95	48 86	B7 E7 4C F0
5	69 E5 14 95	3E 27 61 05	88 E5	38 A5 24 EA
6	3E 27 61 05	DA 4B 20 7D	52 3B	B3 AE 34 E8
7	DA 4B 20 7D	3B 40 E0 FA	4E A4	05 67 81 FF
8	3B 40 E0 FA	83 50 5F 94	7A DE	59 1B 7F E9
9	83 50 5F 94	9E A6 25 93	FE 40	A5 E6 C5 69
10	9E A6 25 93	6B CC 2E 80	5E 76	E8 9C 71 14
11	6B CC 2E 80	B7 79 7F FC	98 19	29 DF 5A 6F
12	B7 79 7F FC	88 8D EF 7A	EE AC	E3 41 C1 FA
13	88 8D EF 7A	93 F8 74 E6	1B D4	24 81 0B 1A
14	93 F8 74 E6	37 D1 63 B7	24 55	BF 5C 8C CD
15	37 D1 63 B7	44 46 BC E4	DC A0	D7 BE C8 02
16	44 46 BC E4	FA FE 29 0B	65 3B	CD 2F 4A BC
17	FA FE 29 0B	D8 6B 48 E4	3E 32	9C 2D F4 00
18	D8 6B 48 E4	54 2D 6E BB	46 52	AE D3 47 B0
19	54 2D 6E BB	2C 82 BF 2A	1C C1	F4 E9 F7 CE
20	2C 82 BF 2A	5B BA E9 71	34 DF	0F 97 87 CA
21	5B BA E9 71	38 28 49 8B	77 8B	14 AA F6 A1
22	38 28 49 8B	0E A7 1A 8C	77 1D	55 1D F3 FD

23	0E A7 1A 8C	33 9C D0 13	D3 24	0B B4 99 98
24	33 9C D0 13	C6 58 51 F1	84 10	C8 FF 4B 7D
25	C6 58 51 F1	E0 B2 08 38	1C A8	D3 2E D8 2B
26	E0 B2 08 38	71 55 D4 0B	BC 64	D7 0D 85 FA
27	71 55 D4 0B	BE 94 A0 EA	A0 DB	5E 26 A8 D2
28	BE 94 A0 EA	88 95 B5 3A	BD D2	F9 C0 61 31
29	88 95 B5 3A	E1 DB DC 34	1F 5F	5F 4F 7C DE
30	E1 DB DC 34	A6 3F CF 84	8F 1C	2E AA 7A BE
31	A6 3F CF 84	93 2D DF 16	6B 81	72 F6 03 22
32	93 2D DF 16	03 E9 32 D4	B5 60	A5 D6 FD 50

６．４．３　後処理（２．４節参照）

最初に，L_{32}とR_{32}を交換する．

(R_{32}, L_{32}) ＝ 0000 0011 1110 1001 0011 0010 1101 0100
1001 0011 0010 1101 1101 1111 0001 0110
（ビット系列）
＝ 03 E9 32 D4 93 2D DF 16 （１６進数）

次に，

(R_{32}, L_{32}) ＝ $(R_{32}, L_{32}) \oplus (\phi, R_{32})$

(R_{32}, L_{32}) ＝ 0000 0011 1110 1001 0011 0010 1101 0100
1001 0000 1100 0100 1110 1101 1100 0010
（ビット系列）
＝ 03 E9 32 D4 90 C4 ED C2 （１６進数）

最後に，

(R_{32}, L_{32}) ＝ $(R_{32}, L_{32}) \oplus (K_{36}, K_{37}, K_{38}, K_{39})$
＝ 1001 1100 1001 1011 0101 0100 1001 0111
0011 1101 1111 0110 1000 0101 1111 1000

（ビット系列）

＝ 9C 9B 54 97 3D F6 85 F8　（１６進数）

暗号文ブロックは（R_{32}, L_{32}）で得られる.

最終結果（暗号文ブロック）は：

C ＝ 1001 1100 1001 1011 0101 0100 1001 0111
0011 1101 1111 0110 1000 0101 1111 1000

（ビット系列）

＝ 9C 9B 54 97 3D F6 85 F8　（１６進数）

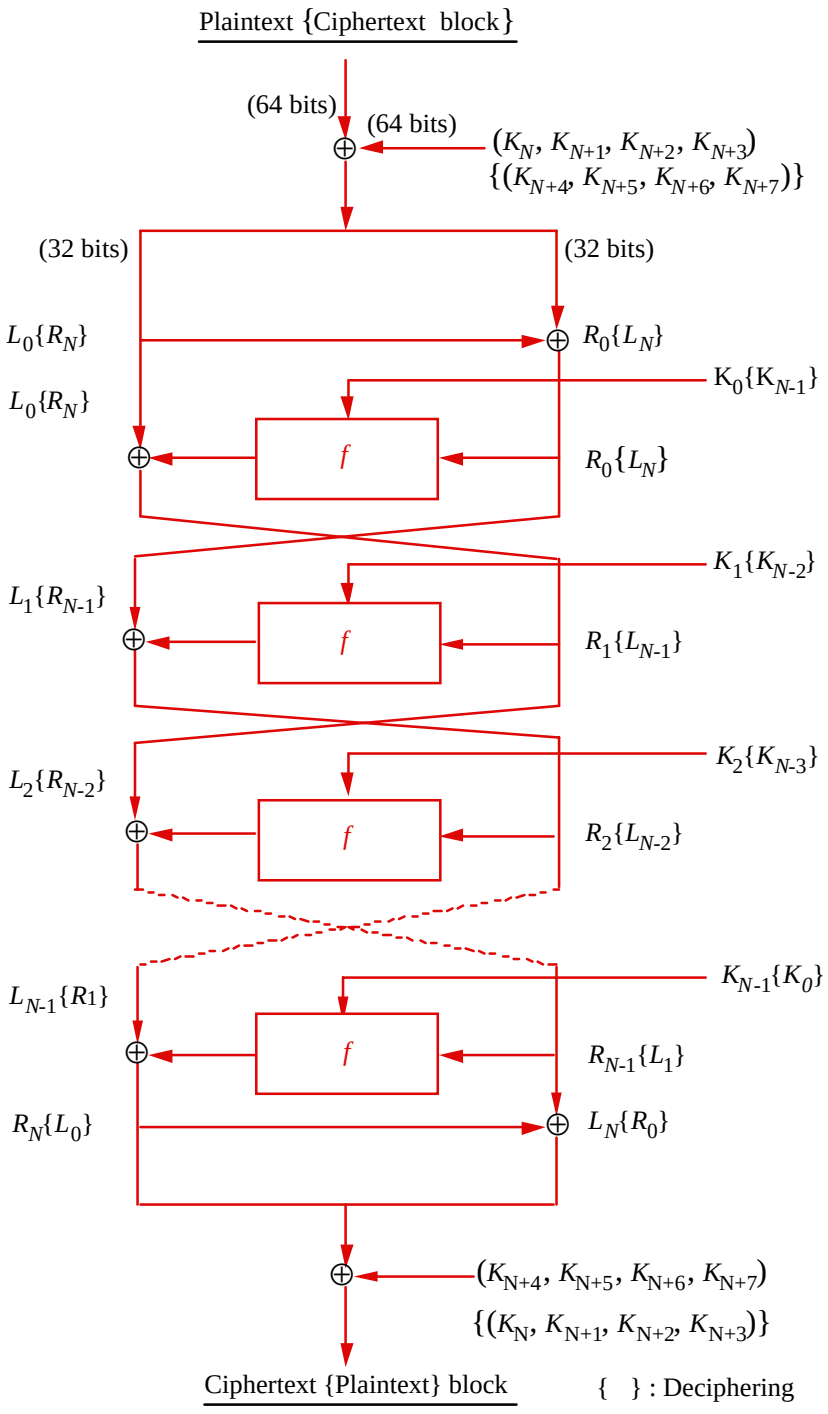
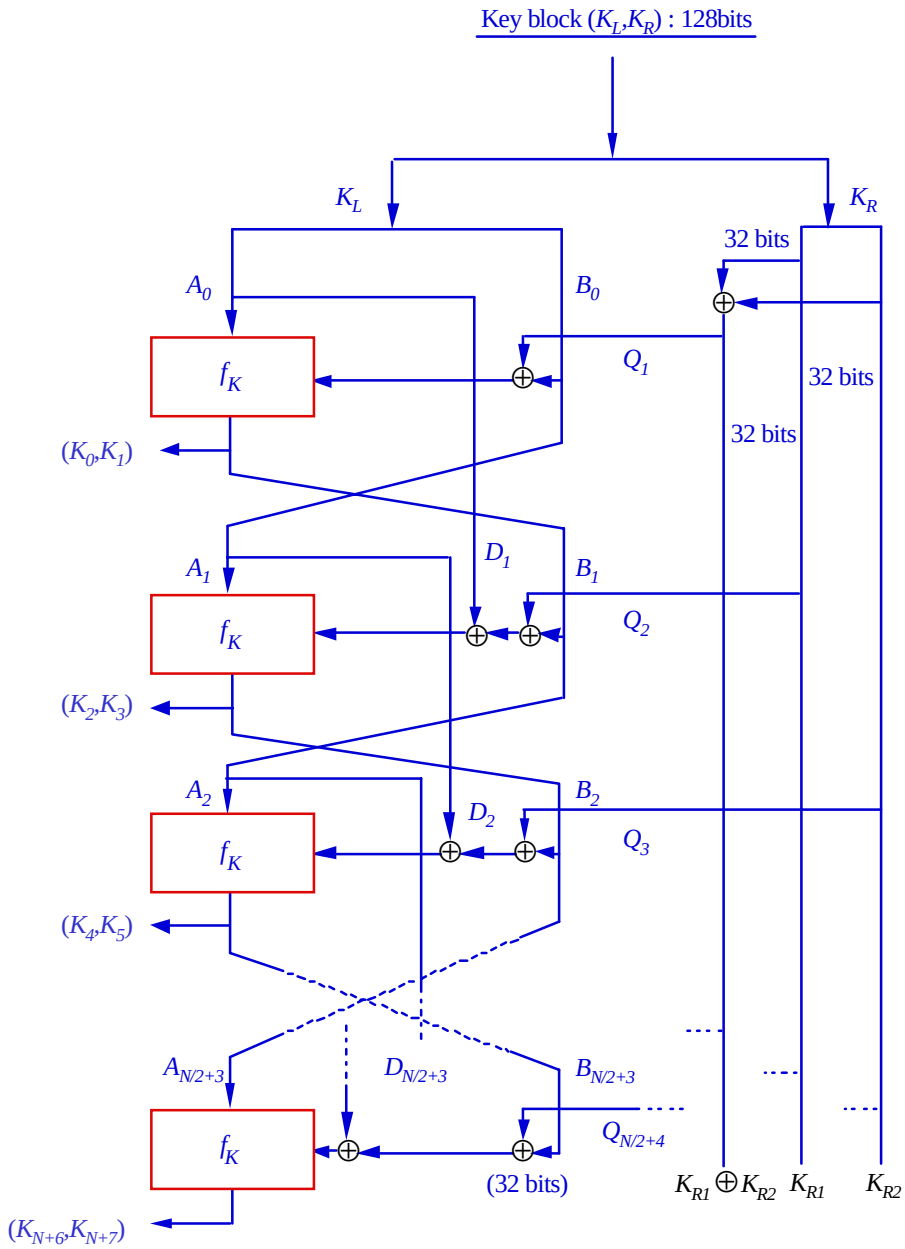


図 1　FEAL-NX のデータランダム(暗号化／復号化アルゴリズム)



$K_{2(r-1)}$: Left half of B_r (16bits)

$K_{2(r-1)+1}$: Right half of B_r (16bits)

Number of iterations is (N/2)+4.

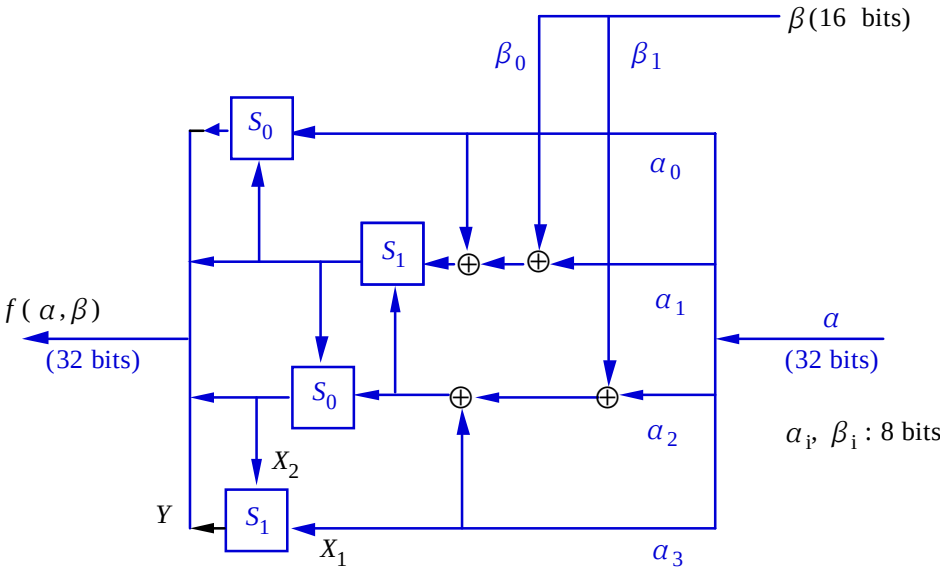
$Q_r = K_{R1} \oplus K_{R2}$ r = 1,4,7,…

$Q_r = K_{R1}$ r = 2,5,8,…

$Q_r = K_{R2}$ r = 3,6,9,…

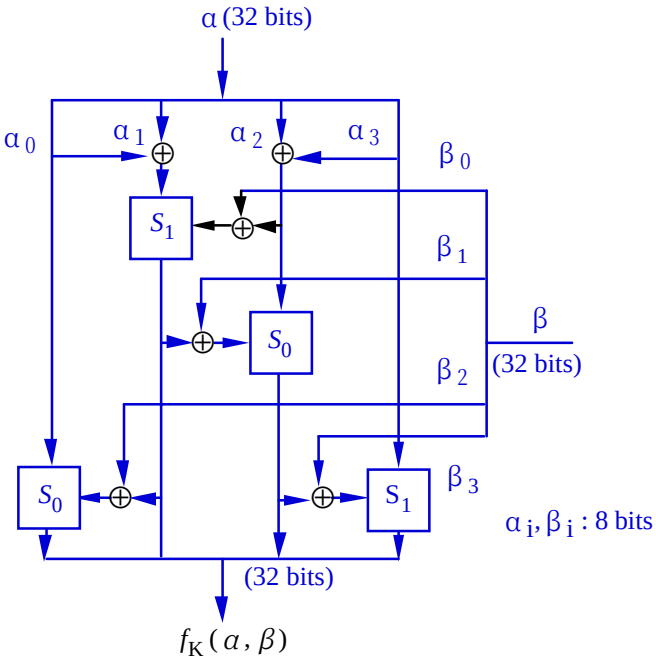
$K_R = (K_{R1}, K_{R2})$

図 2　FEAL-NX の鍵処理



$Y = S_0(X_1, X_2) = \mathrm{Rot2}((X_1 + X_2) \bmod 256)$
$Y = S_1(X_1, X_2) = \mathrm{Rot2}((X_1 + X_2 + 1) \bmod 256)$
Y : output (8 bits), X_1/ X_2 : inputs (8 bits),
Rot2 (Y) : a 2-bit left rotation on 8-bit data Y

図 3　FEAL-NX の関数 f



Note : S_0/S_1 are the same as S_0/S_1 in f-function.

図 4　FEAL-NX の関数 f_K