Iota function

The ι mapping is parameterized by the round index, ir, whose values are specified in Step 2 of Algorithm 7 for computing KECCAK-p [b, nr], Within the specification of ι in Algorithm 2 below, this parameter determines l + 1 bits of a lane value called the round constant, denoted by RC. Each of these l + 1 bits is generated by a function that is based on a linear feedback shift register. This function, denoted by rc, is specified in Algorithm 1. The forward mapping function of iota is based on the two algorithms and the algorithm is given by:

Algorithm 1: To find rc(t)

Input:

integer t.

Output:

bit rc(t).

Steps:

1. If t mod 255 = 0, return 1.

2. Let R = 10000000.

3. For i from 1 to t mod 255, let:

a. R = 0 || R;

b. R [0] = R [0] $\oplus$ R [8];

c. R [4] = R [4] $\oplus$ R [8];

d. R [5] = R [5] $\oplus$ R [8];

e. R [6] = R [6] $\oplus$ R [8];

f. R =Trunc8[R].

4. Return R [0]

Algorithm 6: ι (A, ir)

Input:

state array A;

round index ir

Output:

state array A'.

Steps:

1. For all triples (x, y, z) such that $0 \leq x < 5$, $0 \leq y < 5$, and $0 \leq z < w$, let A' [x, y, z] = A [x, y, z].

2. Let RC = 0w.

3. For j from 0 to l, let RC [2j – 1] = rc (j + 7ir). -(1)

4. For all z such that 0 ≤ z < w, let A′ [0, 0, z] = A′ [0, 0, z] ⊕ RC[z]. – (2)

5. Return A′.

The effect of ι is to modify some of the bits of *Lane* (0, 0) in a manner that depends on the round index *ir* The other 24 lanes are not affected by ι.

For round 1:

For ir =1

For j =0, $RC\ [2^j - 1] = RC\ [0] = rc\ [7]$

For j =1, $RC\ [2^j - 1] = RC\ [1] = rc\ [8]$

For j =2, $RC\ [2^j - 1] = RC\ [3] = rc\ [9]$

For j =3, $RC\ [2^j - 1] = RC\ [7] = rc\ [10]$

For round 2:

For ir =2

For j=0, $RC\ [2^j - 1] = RC\ [0] = rc\ [14]$

For j=1, $RC\ [2^j - 1] = RC\ [0] = rc\ [15]$

For j=2, $RC\ [2^j - 1] = RC\ [0] = rc\ [16]$

For j=3, $RC\ [2^j - 1] = RC\ [0] = rc\ [17]$

Now,

t=0, R= 10000000 RC [0] = 1

t=1, R= 01000000 RC [1] = 0

t=2, R= 00100000 RC [2] = 0

t=3, R= 00010000 RC [3] = 0

t=4, R= 00001000 RC [4] = 0

t=5, R= 00000100 RC [5] = 0

t=6, R= 00000010 RC [6] = 0

t=7, R= 00000001 RC [7] = 0

t=8, R= 10001110 RC [8] = 1

t=9, R= 01000111 RC [9] = 0

t=10, R= 10101101 RC [10] = 1

t=11, R= 11011000 RC [11] = 1

t=12, R= 01101100 RC [12] = 0

t=13, R= 00110110 RC [13] = 0

t=14, R= 00011011 RC [14] = 0

t=15, R= 10000011 RC [15] = 1

t=16, R= 11001111 RC [16] = 1

t=17, R= 11101001 RC [17] = 1

Now when we tried the equation that was given in the NIST paper for pre-image also, it was working perfectly fine. Since the second round is dependent on the first round, we used only the round two equation and verified the number by changing the 0^{th}, 1^{st}, 3^{rd} and 7^{th} bit. The equation that we used was $RC[2^j - 1] = rc\ (j + 14)$