

“ Progress So Far”

Kwame Ackah Bohulu

May 13, 2020

0.1 Notation

1. RTZ (Return-To-Zero) input :- A RTZ input is a binary input which causes a RSC encoder's final state to be return to zero after it has exited the zero state.
2. N :- Interleaver length. We assume that $\tau|N$
3. τ :- cycle length of the RSC encoder. For the 5/7 RSC encoder $\tau = 3$
4. \mathbf{x} :- input vector to the interleaver. It contains the indices of the input bit sequence $\mathbf{x} = \{0, 1, \dots, N - 1\}$
5. \mathbf{y} :- output from the interleaver. It contains the elements of \mathbf{x} rearranged according to the interleaving rule.
6. \mathbf{c}^k :- Coset vector containing elements of \mathbf{x} such that $\mathbf{x} \bmod \tau = k$, $k = 0, 1, \dots, \tau - 1$

0.2 Cosets and RTZ inputs

Let $\mathbf{x} = \{0, 1, \dots, N - 1\}$ be the vector containing the indices of an input bit sequence of length N . Assuming that $\tau|N$, we proceed to form 3 cosets vectors from \mathbf{x} which we will represent by $\mathbf{c}^0, \mathbf{c}^1$ and \mathbf{c}^2 . Having done this, we show how RTZ inputs may be related to the idea of cosets.

For every RSC encoder, there are base RTZ inputs and compound RTZ inputs. Compound RTZ inputs are formed from the combination of one or more base RTZ inputs. These RTZ inputs are important in the sense that they tend to generate low weight codewords which in turn affects the error correcting performance of the RSC code. In the case of the turbo code, we can design an interleaver to transform RTZ inputs in two ways so as to improve the error correcting capability of the turbo code by increasing its minimum distance. These 2 ways include either transforming the RTZ input into a non-RTZ input or transforming the RTZ input into another RTZ input which produces a parity bit sequence with a larger weight than the original. We refer to the latter kind of RTZ input as a heavy RTZ input.

As shown in the previous section, the base RTZ inputs for the 5/7 RSC have weights 2 and 3 respectively and any RTZ input with weight greater than 3 is a compound RTZ input. From a coset perspective, the conditions required for weight 2 or 3 inputs to be RTZ inputs are listed below.

1. Let $P(x) = x^a + x^b$ be the polynomial representation of a weight 2 input sequence. $a, b \in \mathcal{W}$ and represent the index of the "1" bits. $P(x)$ is a RTZ input iff $a \bmod 3 = b \bmod 3$
2. Let $Q(x) = x^a + x^b + x^c$ be the polynomial representation of a weight 3 input sequence. $a, b, c \in \mathcal{W}$ and represent the index of the "1" bits. $Q(x)$ is a RTZ input iff $\{a \bmod 3, b \bmod 3, c \bmod 3\} = (0, 1, 2)$

In other words, weight 2 RTZ inputs have the index of their "1" bits occur in the same coset, whiles weight 3 RTZ inputs have the index of each "1" bits occur in different cosets. For example the weight $1 + x^3$ input is RTZ because $0 \bmod 3 = 3 \bmod 3 = 0$ whiles $1 + x^2$ is not RTZ because $0 \bmod 3 \neq 2 \bmod 3$. Also $x + x^5 + x^9$ is a weight 3 RTZ input since $1 \bmod 3, 5 \bmod 3, 9 \bmod 3$ give the sorted set $(0, 1, 2)$ whiles $1 + x^3 + x^6$ is not a weight 3 RTZ input.

With this insight, we can easily identify weight 2 and weight 3 RTZ inputs and come up with a strategy to transform weight 2 and weight 3 RTZ inputs into non-RTZ inputs. Before

we move on to the next section, we introduce the notion of the coset interleaving pattern and coset interleaving array with an example.

Assuming we have a linear interleaver $\Pi_L(x_i) = \alpha x_i \bmod N$ with $N = 9$ and angular coefficient $\alpha = 5$. We have $\mathbf{x} = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ and $\mathbf{y} = \{0, 5, 1, 6, 2, 7, 3, 8, 4\}$. To show the mapping relation between \mathbf{x} and \mathbf{y} we may use the set we use the notation $\mathbf{x} \mapsto \mathbf{y} = \{0 \mapsto 0, 1 \mapsto 5, \dots, 8 \mapsto 4\}$. A more visually pleasing representation will be in $2 \times N$ matrix representation where the mapping relationship between elements in \mathbf{x} and \mathbf{y} are read column wise as shown below

$$\begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 0 & 5 & 1 & 6 & 2 & 7 & 3 & 8 & 4 \end{bmatrix}$$

The coset interleaver pattern is simply replacing the elements of $\mathbf{x} \mapsto \mathbf{y}$ with the cosets they belong to, in other words finding the value of each element in $\mathbf{x}, \mathbf{y} \bmod 3$

We then have $\mathbf{x} \mapsto \mathbf{y}$ becomes $\{0 \mapsto 0, 1 \mapsto 2, \dots, 2 \mapsto 1\} = (\mathbf{x} \mapsto \mathbf{y}) \bmod 3$ and the matrix representation becomes

$$\begin{bmatrix} 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 0 & 2 & 1 & 0 & 2 & 1 & 0 & 2 & 1 \end{bmatrix}$$

The coset interleaving array is $\mathbf{y} \bmod 3$ and the coset interleaving pattern is $(\mathbf{x} \mapsto \mathbf{y}) \bmod 3$. The coset interleaving pattern shows the input and output relation of the interleaver in terms of its cosets.

0.3 Coset Interleaver Design For Weight-2 RTZ inputs

From the definition of Weight-2 RTZ inputs in the previous section, we know that the index of the “1” bits are in the same coset. Let the indices of the “1” bits before interleaving be represented by i and $i + m\tau$, $m = 1, 2, \dots, \lfloor \frac{N-1}{3} \rfloor$, $i = 0, 1, \dots, N-1$. Let the indices of the “1” bits after interleaving be represented by $\Pi(i)$ and $\Pi(i + m\tau)$. Also let $t = (\Pi(i + m\tau) - \Pi(i)) \bmod 3$ and $s = \Pi(i + m\tau) - \Pi(i)$ be the pre-interleaving separation and post interleaving separation respectively.

To convert a weight-2 RTZ input into a non-RTZ input the following condition should be met.

$$\begin{aligned} \Pi(i) \bmod 3 &\neq \Pi(i + m\tau) \bmod 3, \forall i, m \\ \text{OR} \\ t \bmod 3 &\neq s \bmod 3 \end{aligned} \tag{0-1}$$

Next, we consider the following question.

1. Is there a cut-off interleaver length (N_c) beyond which the condition in (0-1) cannot be met.
2. What is the best course of action to take when the condition in (0-1) cannot be met

To answer the first question, we will do a little thought exercise.

For interleavers with length $0 > N \leq 9$, the number of elements in each coset is at most equal to the number of coset and it is possible map elements in the same coset to elements in another coset. However, when $N > 9$, it is not possible to prevent elements in the same coset from being

mapped to the same coset since the number of elements in each coset is greater than the number of cosets. Therefore, the cut-off interleaver length is $N_c = 9 = \tau^2$.

Table 1 shows all unique coset interleaving arrays of length N_c that convert weight-2 RTZ inputs to non-RTZ inputs. They are labeled from A to X . A coset interleaving array is unique if a shift of the elements in the array does not produce another another coset interleaving array.

| | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|
| A | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 2 |
| B | 0 | 0 | 0 | 1 | 1 | 2 | 2 | 2 | 1 |
| C | 0 | 0 | 0 | 1 | 2 | 1 | 2 | 1 | 2 |
| D | 0 | 0 | 0 | 1 | 2 | 2 | 2 | 1 | 1 |
| E | 0 | 0 | 0 | 2 | 1 | 1 | 1 | 2 | 2 |
| F | 0 | 0 | 0 | 2 | 1 | 2 | 1 | 2 | 1 |
| F | 0 | 0 | 0 | 2 | 2 | 1 | 1 | 1 | 2 |
| H | 0 | 0 | 0 | 2 | 2 | 2 | 1 | 1 | 1 |
| I | 0 | 0 | 1 | 1 | 1 | 0 | 2 | 2 | 2 |
| J | 0 | 0 | 1 | 1 | 2 | 0 | 2 | 1 | 2 |
| K | 0 | 0 | 1 | 2 | 1 | 0 | 1 | 2 | 2 |
| L | 0 | 0 | 1 | 2 | 2 | 0 | 1 | 1 | 2 |
| M | 0 | 0 | 2 | 1 | 1 | 0 | 2 | 2 | 1 |
| N | 0 | 0 | 2 | 1 | 2 | 0 | 2 | 1 | 1 |
| O | 0 | 0 | 2 | 2 | 1 | 0 | 1 | 2 | 1 |
| P | 0 | 0 | 2 | 2 | 2 | 0 | 1 | 1 | 1 |
| Q | 0 | 1 | 0 | 1 | 0 | 1 | 2 | 2 | 2 |
| R | 0 | 1 | 0 | 1 | 0 | 2 | 2 | 2 | 1 |
| S | 0 | 1 | 0 | 1 | 2 | 1 | 2 | 0 | 2 |
| T | 0 | 1 | 0 | 2 | 0 | 1 | 1 | 2 | 2 |
| U | 0 | 1 | 0 | 2 | 0 | 2 | 1 | 2 | 1 |
| V | 0 | 1 | 0 | 2 | 2 | 1 | 1 | 0 | 2 |
| W | 0 | 1 | 1 | 1 | 2 | 0 | 2 | 0 | 2 |
| X | 0 | 2 | 0 | 2 | 0 | 2 | 1 | 1 | 1 |

Table 1: All unique coset interleaving arrays of length $N_c = 9$ for weight-2 RTZ inputs

The interleaver length used in turbo coding are way greater than N_c and it is not possible to transform weight-2 RTZ inputs into non-RTZ inputs for all values of i . All is not lost however, since not all weight-2 RTZ inputs produce low-weight codewords. The formula for calculating the Hamming weight (w_H) of the Turbo codeword produced by a weight-2' RTZ input occurring in both component codes is given by[SunTakeshita]

$$\begin{aligned}
 w_H &= 2 + (2 + \frac{t}{3})w_0 + (2 + \frac{s}{3})w_0 \\
 &= 6 + \left(\frac{t+s}{3}\right)w_0, \quad w_0 = 2
 \end{aligned} \tag{0-2}$$

This means that by altering the value of t and s , we can increase the weight of the codeword produced. With this knowledge, all we need to do is to make sure that the if the input to the interleaver is a weight-2 RTZ inputs with a small value of t , it is converted to a weight-2 RTZ with a large value of s

In summary, an interleaver designed to deal with weight-2 RTZ inputs when $N > N_c$ should

1. Convert weight-2 RTZ inputs to non-RTZ inputs
2. Convert weight-2 RTZ inputs to a weight-2 RTZ inputs with a large value of s when condition 1 isn't possible.

An interleaver design method which makes use of the above points is as follows. Given an interleaver with length N , we break it up into $\frac{N}{N_c}$ blocks each of length N_c . At the beginning of the n th block, the coset interleaving pattern is repeated until the last block. By applying this method, we make sure that when the weight-2 RTZ occurs within a block, condition 1 is met and condition 2 is met when the weight-2 RTZ occurs in 2 consecutive blocks i.e. when $t = N_c$.

This method only works when $N_c | N$ and we will delay the details of what to do when $N_c \nmid N$.

0.4 Coset Interleaver Design For Weight-3 RTZ inputs

As mentioned earlier, a weight-3 RTZ input is formed when the indices of the “1” bits each occur in different cosets. It goes without saying that the simplest way to convert a weight-3 RTZ input into a non-RTZ input is to make sure that at least two of indices of the “1” bits occur within the same coset after interleaving.

With regards to the cut-off interleaver length (N_c) within which this is possible, the simple answer would be that as long as $N > 3n$, $n > 1$ then $N_c = N$. For reasons that will be explained in a later section, we set $N_c = 9$.

Next, we explore coset interleaver patterns for interleaver length $N = N_c$. In the case of weight-2 RTZ inputs and an interleaver length of $N = N_c$, the only RTZ inputs that exist are $1 + x^3$ and $1 + x^6$. With the coset interleaver patterns listed in Table 1, we manage to get rid of all of them. For the case of the weight-3 RTZ however, there are 9 of such RTZ inputs for interleaver length $N = N_c$. Getting rid of all of them might not be possible, which means we might have to alter the conditions for choosing coset interleaver patterns for weight-3 RTZ inputs, such as prioritizing coset interleaver patterns that contain weight-3 RTZ inputs which produce parity bits beyond a certain weight. To calculate the parity weight and by extension, the Hamming weight of a turbo codeword produced by a weight-3 RTZ input, we introduce the idea of layer distance l .

We proceed to form a $N/\tau \times \tau$ matrix \mathbf{X} from \mathbf{x} . We assign indices to each row and refer to each row of \mathbf{X} as a layer. Elements of \mathbf{X} that are on the same row are said to be in the same layer and elements of \mathbf{X} that are in the same column are obviously in the same coset. The layer distance l is the difference between two rows indices of \mathbf{X} . $\mathcal{L}(x)$ represents the layer of x and can be calculated as $\mathcal{L}(x) = \lfloor \frac{x}{3} \rfloor$

Assume we are given a weight-3 RTZ input of the form $x^a + x^b + x^c$, $a < b < c$. Let $l_1 = \mathcal{L}(b) - \mathcal{L}(a)$, $l_2 = \mathcal{L}(c) - \mathcal{L}(a)$ be the pre-interleaving layer distance between b, a and c, a respectively. The parity weight w_p for the given weight-3 RTZ is $w_p = 2(\max(l_1, l_2)) + 2$. Again assuming after interleaving, $x^a + x^b + x^c$ is transformed into another weight-3 RTZ input of the form $x^{a'} + x^{b'} + x^{c'}$, $a' < b' < c'$. Also let $l'_1 = \mathcal{L}(b') - \mathcal{L}(a')$, $l'_2 = \mathcal{L}(c') - \mathcal{L}(a')$ be the post-interleaving layer distance between b', a' and c', a' respectively. And the Hamming weight of the turbo codeword will be

$$\begin{aligned} w_H &= 2(\max(l_1, l_2)) + 2 + 2(\max(l'_1, l'_2)) + 2 + 3 \\ &= 7 + 2(\max(l_1, l_2) + \max(l'_1, l'_2)) \end{aligned} \tag{0-3}$$

For example the weight-3 RTZ input $1 + x + x^2$, $l_1 = l_2 = 0$ and $w_p = 2$. If the output after interleaving is still $1 + x + x^2$ or any shifted version, the hamming weight of the turbo codeword

$w_H = 7$ It should be mentioned that if $a \bmod 3 = 2$ or $a' \bmod 3 = 2$ the value of the layer distance reduces by 1. With this in mind, for interleaver length $N = N_c$, we select coset interleaver arrays which produce a parity weight $w_p > 2$, i.e. those which get rid of the weight-3 RTZ $1 + x + x^2$. Unique coset interleaving arrays which meet this criteria are shown in Table 2 and they are labeled from A to L

| | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|
| A | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 2 |
| B | 0 | 0 | 0 | 1 | 1 | 2 | 1 | 2 | 2 |
| C | 0 | 0 | 0 | 1 | 1 | 2 | 2 | 1 | 2 |
| D | 0 | 0 | 0 | 1 | 1 | 2 | 2 | 2 | 1 |
| E | 0 | 0 | 0 | 2 | 2 | 1 | 1 | 1 | 2 |
| F | 0 | 0 | 0 | 2 | 2 | 1 | 1 | 2 | 1 |
| G | 0 | 0 | 0 | 2 | 2 | 1 | 2 | 1 | 1 |
| H | 0 | 0 | 1 | 0 | 1 | 1 | 2 | 2 | 2 |
| I | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 0 | 2 |
| J | 0 | 0 | 2 | 0 | 2 | 2 | 1 | 1 | 1 |
| K | 0 | 0 | 2 | 2 | 2 | 1 | 1 | 0 | 1 |
| L | 0 | 1 | 0 | 1 | 1 | 2 | 2 | 0 | 2 |

Table 2: All unique permutation matrices of length $N_c = 9$ for weight-3 RTZ inputs

As mentioned earlier, there are 9 weight-3 RTZ inputs for $N = N_c$. The weight-3 RTZ inputs that remain depends on the coset interleaver pattern that is chosen.

0.5 Coset Interleaver Design For Weight-2 and Weight-3 RTZ Inputs

In this section we attempt to design an interleaver while considering both weight-2 and weight-3 RTZ inputs.

We begin by determining coset interleaver patterns which fulfil the conditions for both weight-2 and weight-3 inputs when $N = N_c$. To recap this coset interleaver pattern should fulfil the following conditions

1. Should convert all weight-2 RTZ inputs into non-RTZ inputs
2. Should convert all weight-3 RTZ inputs into non-RTZ inputs (best case) or RTZ inputs which produce high weight parity bits.

There are 4 coset interleaving arrays of length N_c and these are shown in Table 3 labelled A to D

| | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|
| A | 0 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 2 |
| B | 0 | 0 | 0 | 1 | 1 | 2 | 2 | 2 | 1 |
| C | 0 | 0 | 0 | 2 | 2 | 1 | 1 | 1 | 2 |
| D | 0 | 0 | 0 | 2 | 2 | 2 | 1 | 1 | 1 |

Table 3: All unique coset interleaving arrays of length $N_c = 9$ for weight-2 and weight-3 RTZ inputs

For the case where $N > N_c$, we divide N into N/N_c blocks of length N_c and at the n th block, the coset interleaver pattern is repeated until the last block. Regardless of which coset interleaver pattern is chosen, the following things should be confirmed

Weight-2 RTZ

1. The weight-2 RTZ is mapped to another weight-2 RTZ iff the pre-interleaving distance t is a multiple of N_c

Weight-3 RTZ

1. The weight-3 RTZ is never mapped to the weight-3 RTZ $1 + x + x^2$

With this in mind, the only coset interleaver patterns which can do this are A and D of Table 3

0.6 Implementation of Coset Interleaver

Now that we have used the coset interleaver pattern to place a limit of the kinds of RTZ inputs that occur after interleaving, we present an algorithm that not only implements the coset interleaver pattern, but also determines the best input output RZT pairing to increase the minimum codeword weight for weight-2 and weight-3 RTZ inputs.

0.6.1 Implementation Algorithm

Let N be the interleaver length, $L = N/3$ be the coset length and $\mathbf{x} = \{0, 1, \dots, N-1\}$ be the input to the interleaver. Also let \mathbf{c}^k , $k = 0, 1, 2$ contain elements in \mathbf{x} such that $x_i \bmod 3 = k$. The algorithm is as follows

Algorithm 1 Coset Interleaver Algorithm 1

Require: $i = 0$, $k = 0$, $j = 0$, $L \bmod 3 = 0$

```

for  $i = 0$  to  $N - 1$  do
   $y[i] = \mathbf{c}^k[j]$ 
   $j = j + D \bmod L$ 
  if  $i + 1 \bmod 3 = 0$  then
     $k = k + 1 \bmod 3$ 
  end if
end for

```

Algorithm 1 only works for cases where $L \bmod 3 = 0$. If $L \bmod 3 = 1$ or $L \bmod 3 = 2$ The change to the algorithm is as follows

For the case of Algorithm 2, when $L \bmod 3 = 0$, the coset interleaving pattern is always $\{0 \mapsto 0, 1 \mapsto 0, 2 \mapsto 0, 0 \mapsto 1, 1 \mapsto 1, 2 \mapsto 1, 0 \mapsto 2, 1 \mapsto 2, 2 \mapsto 2\}$ for every n th block of length $N_c = 9$. When $L \bmod 3 = 1$ or $L \bmod 3 = 2$, there is a slight change in the coset interleaver pattern. The coset interleaver pattern that is generated when $N = 15$ ($L = 5$ and $5 \bmod 3 = 2$) and $N = 12$ ($L = 4$ and $4 \bmod 3 = 1$) is show in Table 4 and Table 5 respectively.

For the coset interleaver pattern in Table 4 and Table 5, the minimum pre-interleaving separation t at which a weight 2 RTZ input is mapped to another weight-2 RTZ input is $t = 6$. Also,

Algorithm 2 Coset Interleaver Algorithm 2

Require: $i = 0, k = 0, j = 0, r = 0$

```

for  $i = 0$  to  $N - 1$  do
   $y[i] = \mathbf{c}^k[j]$ 
   $j = j + D \bmod L$ 
  if  $i + 1 \bmod 3 = r$  then
     $k = k + 1 \bmod 3$ 
  else if  $i + 1 \bmod L = 0$  then
     $k = k + 1 \bmod 3$ 
     $r = i + 1 \bmod 3$ 
  end if
end for

```

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 |
| 0 | 0 | 0 | 1 | 1 | 2 | 2 | 2 | 0 | 0 | 1 | 1 | 1 | 2 | 2 |

Table 4: Coset Interleaver pattern when $L \bmod 3 = 2$

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 | 0 | 1 | 2 |
| 0 | 0 | 0 | 1 | 2 | 2 | 2 | 0 | 1 | 1 | 1 | 2 |

Table 5: Coset Interleaver pattern when $L \bmod 3 = 1$

for the coset interleaver pattern in Table 5, there is a point in the coset interleaver pattern where the weight-3 RTZ input $1 + x + x^2$ is mapped to itself.

Fortunately the multiplicity of these irregularities is few and by carefully choosing the value of D these irregularities pose no problem to the effectiveness of the interleaver.

Unfortunately, when $\lfloor \frac{L}{3} \rfloor \bmod 3 = 0$, Algorithm 2 is unable to produce an interleaver and requires further adjustment(*Still thinking about it)

0.6.2 Choosing D for the coset interleaver

As seen from the algorithms above, there is a factor D which needs to be chosen. This factor is important as it determines the values of s , which is the post-interleaving separation for weight-2 RTZ inputs as well as l'_1 and l'_2 which are the post-interleaving layer distances for weight-3 RTZ inputs. This in turn influences the minimum weight codeword of the turbo code.

For the case of weight-2 RTZ inputs we know that the minimum pre-interleaver separation t within which a RTZ input occurs is $t = 9$ for the case when $L \bmod 3 = 0$ and $t = 6$ for the case when $L \bmod 3 = 1$ or $L \bmod 3 = 2$. For the different values of t we need to make sure that the post-interleaving distance s is close to $\lfloor \frac{L}{2} \rfloor$

For the case when $L \bmod 3 = 1$ or $L \bmod 3 = 2$, finding D is as simple as solving the congruence equation below

$$\begin{aligned}
 tD &\equiv \lfloor \frac{L}{2} \rfloor \bmod L \\
 6D &\equiv \lfloor \frac{L}{2} \rfloor \bmod L
 \end{aligned} \tag{0-4}$$

However, when $L \bmod 3 = 0$, finding the best D is a bit more complicated. Let \mathbf{d} be a

vector containing integer values from the vector $\mathbf{x}' = \{2, \dots, L-1\}$ which meet the condition $\gcd(x'_i, L) = 1$. The process for finding the best D is shown below.

Algorithm 3 Algorithm to find D when $L \bmod 3 = 0$

```

for  $i = 0$  to  $\|\mathbf{d}\|$  do
   $f(d_i) = \lfloor \frac{L}{2} \rfloor - (9d_i \bmod L)$ 
end for
 $D = \arg \min f(d_i)$ 

```

From Algorithm 3, the value of D can be arbitrarily chosen from \mathbf{D} .

Now that we have decided upon the value of D , can calculate the value of s and return calculate the corresponding w_H for the codeword produced using (0-2).

$$s = \min \left\{ 3 \left(tD \bmod L \right), 3 \left(t(L - D) \bmod L \right) \right\} \quad (0-5)$$

(0-2) then becomes,

$$\begin{aligned}
 w_H &= 6 + \left(\frac{t+s}{3} \right) w_0 \\
 &= \min \left\{ 6 + \left(\frac{t + 3 \left(tD \bmod L \right)}{3} \right) w_0, 6 + \left(\frac{t + 3 \left(t(L - D) \bmod L \right)}{3} \right) w_0 \right\} \quad (0-6)
 \end{aligned}$$

Finding the value of D which maximizes the value of post interleaving pattern distance for weight-3 RTZ is a bit more difficult. Figure shows the coset interleaver array for interleaver on length N divided into blocks of length N_c with the focus on the first 2 blocks. We refer to RTZ inputs that occur within a block as intra-block RTZ inputs and those that occur between 2 blocks as inter-block RTZ inputs. Let $L \bmod 3 = 0$. In the case of weight-2 RTZ inputs, there are no inter-block RTZ inputs and the only inter-block input is $1 + x^9$ and its shifted versions. For the case of weight-3 RTZ inputs, there are multiple intra-block RTZ inputs as well as inter-block RTZ inputs. These are listed in Table

In choosing the value of D to maximize the post-interleaving layer distances, the algorithm below is used.

Algorithm 4 Weight-3 RTZ input max D Algorithm

Require: $i = 0, k = 0, j = 0, r = 0$

```

for  $i = 0$  to  $N - 1$  do
   $y[i] = \mathbf{c}^k[j]$ 
   $j = j + D \bmod L$ 
  if  $i + 1 \bmod 3 = r$  then
     $k = k + 1 \bmod 3$ 
  else if  $i + 1 \bmod L = 0$  then
     $k = k + 1 \bmod 3$ 
     $r = i + 1 \bmod 3$ 
  end if
end for

```

Having chosen the value of D using Algorithm 4, we go ahead and calculate the minimum Hamming distance of the codeword produced by weight-3 RTZ inputs. From the above algorithm,

we have the weight-3 RTZ input that produces the codeword with the largest minimum distance. The pre-interleaving layer distance with respect to that input is

$$l_1 = \left\lfloor \frac{c-a}{3} \right\rfloor \quad (0-7)$$

and the post-interleaving layer distance is

$$l'_1 = \min \left\{ \left\lfloor \frac{D(c-a)}{3} \right\rfloor, \left\lfloor \frac{(L-D)(c-a)}{3} \right\rfloor \right\} \quad (0-8)$$

Substituting into (eq4) gives

$$\begin{aligned} w_H &= 2(\max(l_1, l_2)) + 2 + 2(\max(l'_1, l'_2)) + 2 + 3 \\ &= 7 + 2 \left(\max(l_1, l_2) + \max(l'_1, l'_2) \right) \\ &= \min \left\{ 7 + 2 \left(\left\lfloor \frac{c-a}{3} \right\rfloor + \left\lfloor \frac{D(c-a)}{3} \right\rfloor \right), 7 + 2 \left(\left\lfloor \frac{c-a}{3} \right\rfloor + \left\lfloor \frac{(L-D)(c-a)}{3} \right\rfloor \right) \right\} \end{aligned} \quad (0-9)$$