3GPP TSG RAN WG1 meeting #7
Hanover, Germany
August 30th – September 3rd, 1999

Tdoc R1-(99)C21

Agenda Item: AH4
Source: Silicon Automation Systems Ltd., India.
Title: Combined 1st interleaving and Radio frame segmentation
Document for: Decision

Introduction

This document proposes to combine 1st interleaving and Radio frame segmentation. In the last meeting (WG1#6), a rule for Radio frame segmentation was included in the TS 25.212 document. We propose that this functionality can be included in the 1st interleaving itself, there by avoiding extra processing.

Working assumption

1st Interleaving

Here data of size K_1 is written row wise into a rectangular matrix of size $R_1 \times C_1$ where R_1 is chosen such that $K_1 <= R_1 \times C_1$.
After column interleaving, data is read column wise deleting non-existence bits.

Radio frame segmentation

Here data is divided into (TTI/10) segments each of 10ms duration. If the radio frame size is not a multiple of (TTI/10), filler bits will be added.

Notational Equivalence

Since the aim is to merge radio frame segmentation into 1st interleaving, first step is to list out parameters, which mean the same in both. Following is the list of parameters.

Parameter	1st interleaving	Radio frame segmentation
Radio frame size	Input block length, K_1	Size of transport channel data, L_I
TTI(msec) / 10 (msec)	Number of columns, C_1	T_I
Segment size	Number of rows, R_1	$(L_i + r_i)/ T_i = R_I$ or K_i
Number of filler bits	$l_1 = R_1 \times C_1 – K_1$	$r_i = T_i - (L_i \bmod T_i)$

Thus segmentation does two things, division into segments & inserting filler bits. This can be incorporated in the interleaver itself since data is already being divided into columns.

Proposed Changes

1st Interleaving

<u>**First Stage**</u>**:**

(1) Select column number C_1 = TTI/10.
(2) Determine a row number R_1 by finding minimum integer R_1 such that,

$$K_1 <= R_1 \times C_1.$$

(3) The input sequence of the 1st interleaving is written into the $R_1 \times C_1$ rectangular matrix row by row.
(4) **Initialize the remaining l_1 bits in the last row to filler bits, where $l_1 = R_1 \times C_1 - K_1$.**

Second Stage:

(1) Perform the inter-column permutation based on the pattern $\{P_1(j)\}$ (j=0,1, ..., C-1), where $P_1(j)$ is the original column position of the j-th permuted column.
(2) The output of the 1st interleaving is the sequence read out column by column from the inter-column permuted $R_1 \times C_1$ matrix, **where each column corresponds to a 10ms radio frame segment**.

Conclusion

Processing can be reduced by merging 1st interleaving and radio frame segmentation, without change in performance. Since this involves minor editorial changes, we propose that our text proposal be accepted.

Text Proposal

4.2 Transport channel coding/multiplexing

Figure 1: Remove Radio frame segmentation block.
Figure 2: Remove Radio frame segmentation block.

4.2.4 1st interleaving and Radio frame segmentation

The 1st interleaving of channel interleaving consists of two stage operations. In first stage, the input sequence is written into rectangular matrix row by row. The second stage is inter-column permutation. The two-stage operations are described as follows, the input block length is assumed to be K_1.

First Stage:

(1) Select a column number C_1 from Table 3.
(2) Determine a row number R_1 by finding minimum integer R_1 such that,

$$K_1 <= R_1 \times C_1.$$

(3) The input sequence of the 1st interleaving is written into the $R_1 \times C_1$ rectangular matrix row by row.

(4) Initialize the remaining l_1 bits in the last row to filler bits, where $l_1 = R_1 \times C_1 - K_1$

Second Stage:

(1) Perform the inter-column permutation based on the pattern $\{P_1(j)\}$ (j=0,1, ..., C-1) that is shown in Table 3, where $P_1(j)$ is the original column position of the j-th permuted column.

(2) The output of the 1st interleaving is the sequence read out column by column from the inter-column permuted $R_1 \times C_1$ matrix, where each column corresponds to a 10ms radio frame segment.

~~and the output is pruned by deleting the non-existence bits in the input sequence, where the deleting bits number l_1 is defined as:~~

~~$l_1 = R_1 \times C_1 - K_1$.~~

Table 3

Interleaving span	Column number C_1	Inter-column permutation patterns
10 ms	1	{0}
20 ms	2	{0,1}
40 ms	4	{0,2,1,3}
80 ms	8	{0,4,2,6,1,5,3,7}

~~4.2.5 Radio frame segmentation~~

~~Each transport channel with transmission time interval 10, 20, 40, or 80 msec is segmented into 10 msec equi-sized data blocks. Those segmented 1, 2, 4, or 8 blocks, depending on transmission time interval, are output to rate matching for uplink and 2nd multiplexing for downlink in block-wise order at every 10 msec. Figure B-1 and B-2 illustrate data flow from 1st interleaver down to 2nd interleaver in both uplink and downlink channel coding and multiplexing chains. In the figures, it is assumed that there are N different channel coding and multiplexing chains. The following subsections describe input-output relationship of~~

radio frame segmentation in bit-wise manner, referring to the notations in Figure B-1 and B-2, where the notations in each data block, for examples L_i, R_i, K_i, P/M, etc., indicate number of bits of the data block. Define some notations:

L_i = Size of i^{th} transport channel data in bits to radio frame segmentation

T_i = Transmission Time Interval of i^{th} channel coding and multiplexing chain (msec) / 10 (msec)

So, T_i Î {1, 2, 4, 8} for $i = 0, 1, 2, \ldots, N$

4.2.5.1 Radio frame size equalization

i^{th} transport channel data of size L_i is segmented into radio frames of size L_i/T_i. Since the size of radio frame, L_i/T_i is not necessarily an integer, some of T_i the radio frames will contain one bit less than others. For systematic process of the proceeding functional blocks, the radio frame sizes are equalized to be one finite size by considering the number of proper filler bits. Note that maximum possible filler bits are 7 for transmission time interval of 80 msec. These filler bits are evenly distributed over the one-bit short radio frames. Following is the algorithm of radio frame size equalization.

t = radio frame index (1, 2, 3, …, T_i) for a given i^{th} channel coding and multiplexing chain

$r_i = T_i - (L_i \bmod T_i)$ Î {0, 1, 2, …, T_i-1}   // number of filler bits

$(L_i + r_i)/T_i = R_i$ // Target radio frame size for uplink

$(L_i + r_i)/T_i = K_i$ // Target radio frame size for downlink

If ri ¹ 0 then

For each t (³ T_i-r_i+1)

Add one filler bit to the end of t^{th} radio frame

End

End If

4.2.5.2 Radio frame segmentation rule

Parameter r_i for segmentation are determined in radio frame size equalization.

The bits before radio frame segmentation for i^{th} channel coding and multiplexing chain are denoted by:

$b_{i1}, b_{i2}, \ldots b_{iL_i}$

Bits after radio frame segmentation block are 10 msec-based and denoted by:

$c_{i1}, \ldots c_{i,(L_i+r_i)/T_i}$

and related to the input bits to radio frame segmentation as follows.

Bits after radio frame segmentation in the first 10 msec time interval: *(t=1)*

$c_{ij} = b_{ij}$ $j=1,2,\ldots,(L_i+r_i)/T_i$

($(L_i + r_i)/T_i$ equals to R_i and K_i for uplink and downlink, respectively.)

Bits after radio frame segmentation in the second 10 msec time interval: *(t=2)*

$c_{ij} = b_{i,(j+(L_i+r_i)/T_i)}$ $j=1,2,\ldots,(L_i+r_i)/T_i$

…

Bits after radio frame segmentation in the $(T_i - r_i)^{th}$ 10 msec time interval: $(t=T_i - r_i)$

$c_{ij} = b_{i,(j+(T_i-r_i-1)(L_i+r_i)/T_i)}$ $j=1,2,\ldots,(L_i+r_i)/T_i$

Bits after radio frame segmentation in the $(T_i - r_i+1)^{th}$ 10 msec time interval: $(t=T_i - r_i+1)$

$c_{ij} = b_{i,(j+(T_i-r_i)(L_i+r_i)/T_i)}$ $j=1,2,\ldots,(L_i+r_i)/T_i-1$

$c_{ij} = filler\_bit(0/1)$ $j=(L_i+r_i)/T_i$ *(filler bit)*

...

Bits after radio frame segmentation in the T_i^{th} 10 msec time interval: *(t=T_i)*

$c_{ij} = b_{i,(j+(T_i-1)(L_i+r_i)/T_i)}$ $j=1,2,\ldots,(L_i+r_i)/T_i-1$

$c_{ij} = filler\_bit(0/1)$ $j=(L_i+r_i)/T_i$ *(filler bit)*

Annex B (informative):

Remove this annex