TSG-RAN Working Group1 meeting #7 *TSGR1#7(99)D83*
Hannover, Germany,
August 30 – September 3, 1999
Agenda item: Ad Hoc 10 Report and Text proposal

Source: Samsung

Title: **Text proposal regarding Multiple Scrambling Codes (rev. of R1-99d51)**

Document for: Approval

Abstract

Samsung proposed about the multiple scrambling code generating in [1]. This text proposal descries the text change of the multiple scrambling code section to the contribution.

=============== Start of Proposed Text ===============

5.2.2 Scrambling code

There are a total 512*~~512~~16 = ~~262,144~~8,192 scrambling codes, numbered 0…~~262,144~~ 8,191. The scrambling codes are divided into 512 sets each of a primary scrambling code and ~~511~~ 15 secondary scrambling codes.
The primary scrambling codes consist of scrambling codes n=16*i, where i=0…511. The i:th set of secondary scrambling codes consists of scrambling codes ~~i+k*512, where k=1...511~~16*i+k, where k=1...15.
There is a one-to-one mapping between each primary scrambling code and ~~511~~ 15 secondary scrambling codes in a set such that i:th primary scrambling code corresponds to i:th set of scrambling codes.
The set of primary scrambling codes is further divided into 32 scrambling code groups, each consisting of 16 primary scrambling codes. The j:th scrambling code group consists of primary scrambling codes 16*16*j+16*k, where j=0...31, and k=0...15~~j*16,..., j*16+15,where j=0,...,31.~~.
Each cell is allocated one and only one primary scrambling code. The primary CCPCH is always transmitted using the primary scrambling code. The other downlink physical channels can be transmitted with either the primary scrambling code or a secondary scrambling code from the set associated with the primary scrambling code of the cell.

~~*<Editor's note: There may be a need to limit the actual number of codes used in each set of secondary scrambling codes, in order to limit the signalling requirements.>*~~
<Editor's note: it is not standardised how many scrambling codes a UE must decode in parallel.>
The scrambling code sequences are constructed by combining two real sequences into a complex sequence. Each of the two real sequences are constructed as the position wise modulo 2 sum of [38400 chip segments of] two binary *m*-sequences generated by means of two generator polynomials of degree 18. The resulting sequences thus constitute segments of a set of Gold sequences. The scrambling codes are repeated for every 10 ms radio frame. Let *x* and *y* be the two sequences respectively. The *x* sequence is constructed using the primitive (over GF(2)) polynomial $1+X^7+X^{18}$. The y sequence is constructed using the polynomial $1+X^5+X^7+X^{10}+X^{18}$.
<Editor's note: [ ] is due to the fact that only 3.84Mcps is an agreement. 0.96, 7.68, and 15.36Mcps are ffs.>
~~Let n_{17} ... n_0 be the binary representation of the scrambling code number n (decimal) with n_0 being the least significant bit.~~
The *x* sequence depends on the chosen scrambling code number *n* and is denoted x_n, in the sequel. Furthermore, let $x_n(i)$ and $y(i)$ denote the *i*:th symbol of the sequence x_n and *y*, respectively.
The *m*-sequences x_n and *y* are constructed as:

Initial conditions:
x_0 is constructed with $x_0(0)= x_0 (1)=...= x_0 (16)=0, x_0 (17)=1$

~~$x_n(0)=n_0 , x_n(1)= n_1 , \ldots =x_n(16)= n_{16} , x_n(17)= n_{17}$~~
$y(0)=y(1)= \ldots =y(16)= y(17)=1$

Recursive definition of subsequent symbols:
$x_n(i+18) =x_n(i+7) + x_n(i)$ *modulo* 2, $i=0,\ldots,2^{18}-20$,
$y(i+18) = y(i+10)+y(i+7)+y(i+5)+y(i)$ *modulo* 2, $i=0,\ldots, 2^{18}-20$.
x_n is constructed with the following equation.
$x_n(i) =x_0(i+n), \quad i=0,...,2^{18}-2$
The n:th Gold code sequence z_n is then defined as
$z_n(i) = x_n(i) + y(i)$ *modulo* 2, $i=0,\ldots, 2^{18}-2$.
These binary code words are converted to real valued sequences by the transformation ‘0’ -> ‘+1’, ‘1’ ->

Finally, the n:th complex scrambling code sequence C_{scramb} is defined as (the lowest index corresponding to the chip scrambled first in each radio frame): (where N is the period in chips and M is 131,072)

$C_{scramb}(i) = z_n(i) + j\, z_n(i+M), i=0,1,\ldots,N-1.$
<Editor’s note: the values 38400 is based on an assumption of a chip rate of 3.84 Mcps.>
Note that the pattern from phase 0 up to the phase of 38399 is repeated.

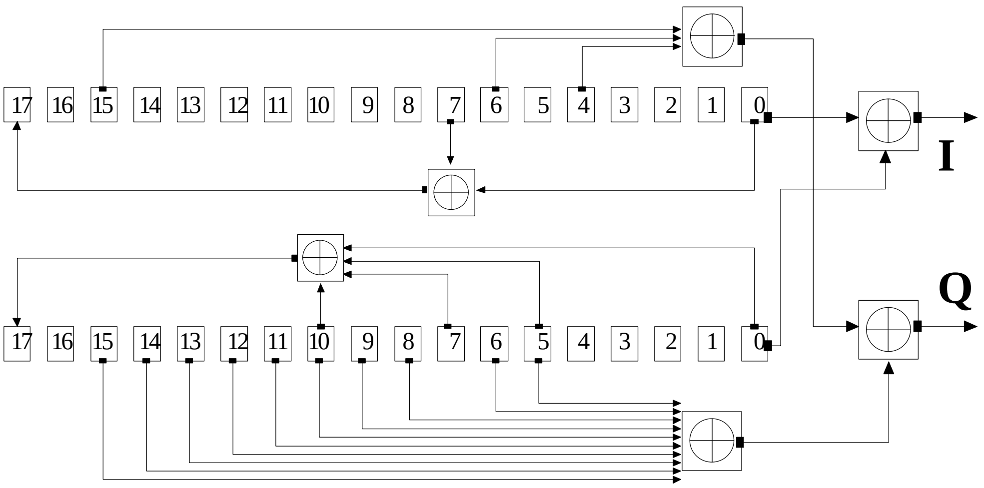

Figure 14. Configuration of downlink scrambling code generator
================== End of Proposed Text ==================

References

[1] 3GPP TSGR1#6 (99)924, ‘Multiple scrambling code’, Samsung
[2] 3GPP TSGR1#7 (99)a66, ‘TS 25.2.1.3 v2.1.2 (R1-99a66.doc), Spreading and modulation (FDD)’
[3] 3GPP TSGR1#7 (99)b59, ‘Text proposal regarding multiple scrambling codes’, Samsung
[4] 3GPP TSGR1#7 (99)d31, ‘Text proposal regarding multiple scrambling codes (rev.)’, Samsung