TSG-RAN Working Group 1 meeting #6
Helsinki, Filand
July 13-16, 1999

TSGR1#6(99)916

Agenda item:

Source: Samsung

Title: Text proposal for Multiple scrambling codes

Document for: Decision

Introduction

This paper includes a text proposal for multiple scrambling codes in document 25.213. Some additional descriptions are also proposed. We propose that 512 of the scrambling codes are primary scrambling codes. For each primary scrambling code, there are then 4 additional secondary scrambling codes defined. The secondary scrambling codes are generated by masking operation to minimize the H/W complexity of the various scrambling code generation.

--- Text proposal for 25.213 ---

5 Downlink spreading and modulation

5.2.2 Scrambling code

~~There are a~~A total of 2^{18} = 262144 scrambling codes can be generated, but ~~Then, we can use~~ only 2560 codes, numbered 0, …, 2560 are to be used ~~,~~ as scrambling codes. Code 0 is a sum of two different m-sequences $x_0(t)$ and y (t) initialized by $x_0(0)= x_0(1)=, \ldots =x_0(16)= 0$, $x_0(17)=1$, and y(0)= y(1)=, … =y(16)= 0 , y(17)=1. Code N can be defined by a sum of the N shifted version $x_N(t)$ of $x_0(t)$ and y(t). The scrambling codes are divided into a set of *primary scrambling codes*, consisting of scrambling code 5*n, n=0, …, 511, and 512 sets of *secondary scrambling codes*, where the i:th set of secondary scrambling codes consists of scrambling codes i*5+1,...,i*5+4, ~~i*511+512, …, i*511+512+510,~~ where i=0,…,511. There is a one-to-one mapping between each primary scrambling code and a set of secondary scrambling codes such that i:th primary scrambling code corresponds to i:th set of secondary scrambling codes.

The set of primary scrambling codes is further divided into 32 scrambling code groups, each consisting of 16 primary scrambling codes. The i:th scrambling code group consists of scrambling codes 512*(i*16 + t)~~i*16, …, i*16+15,~~ where i=0, …, 31, t=0,...,15.

Each cell is allocated a single ~~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 requriements. >

~~The total number of available scrambling codes is 512, divided into 32 code groups with 16 codes in each group.~~

~~[In order to avoid code limitation in some cases, e.g. when increasing the capacity using adaptive antennas, the possibility to associate several scrambling codes with one cell (BCH area) has been identified as one solution. The exact implementation of such a scheme is still to be determined.]~~

<Ericsson's note: There should obviously be some restrictions on how many different scrambling codes the UE must be able to descramble in parallel>

<Editor's note: Use of multiple downlink scrambling codes to aid adaptive antennas are ffs.>

The primary 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~~40960~~ 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. As we see in the Figure1, the secondary scrambling code sequences are constructed by combining one mask operation value of one shift register and summing with the other sequences into a complex sequence. 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 4.096Mcps is a working assumptions. 1.024, 8.196, and 16.384Mcps are ffs.>

Let $n_{17} \ldots n_0$ 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:

$x_n(0)=n_0$, $x_n(1)= n_1$, … $=x_n(16)= n_{16}$, $x_n(17)= n_{17}$

$y(0)=y(1)= \ldots =y(16)= y(17)=1$

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, $i=0,\ldots, 2^{18}-20$.

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' -> '-1'.

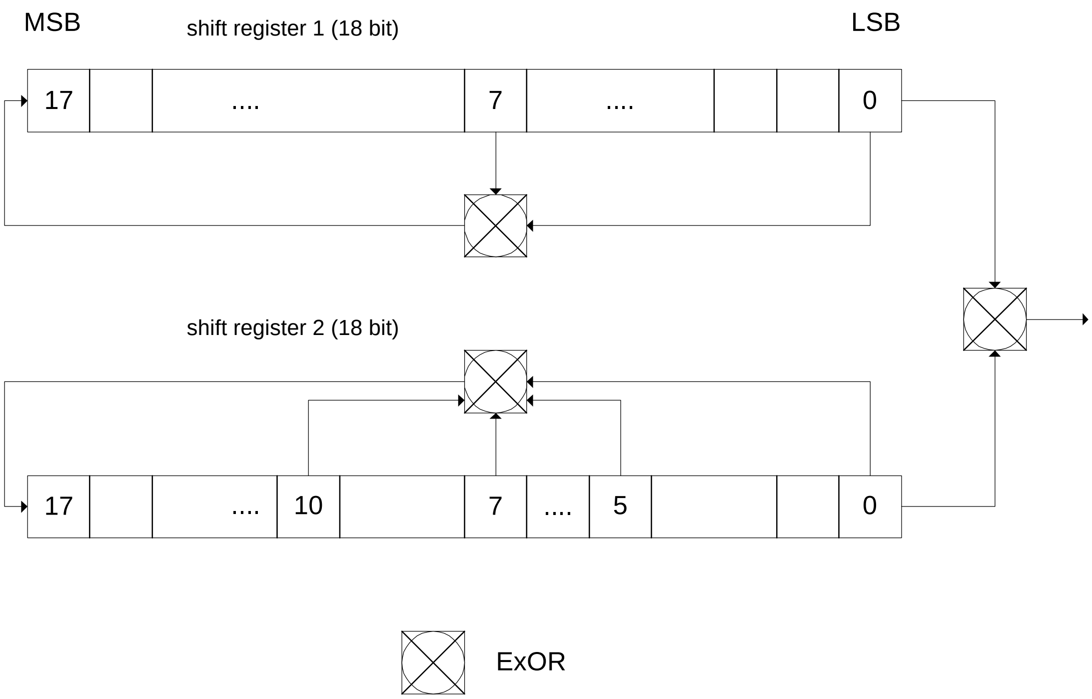
Finally, the n:th complex scrambling code sequence C_{scramb} is defined as (the lowest index corresponding to the chip scrambled first in each radio frame): (see Table 1 for definition of N and M)

$C_{scramb}(i) = z'_n(i) + j\, z'_n(i+M)$, $i=0,1,\ldots,N-1$.

<Editor's note: the values 3584 and 40960 are based on an assumption of a chip rate of 4.096 Mcps.>

Note that the pattern from phase 0 up to the phase of 10 msec is repeated.

~~The index n runs from 0 to 511 giving 512 distinct 40960 chip sequences.~~



~~**Figure 1. Configuration of downlink scrambling code generator**~~

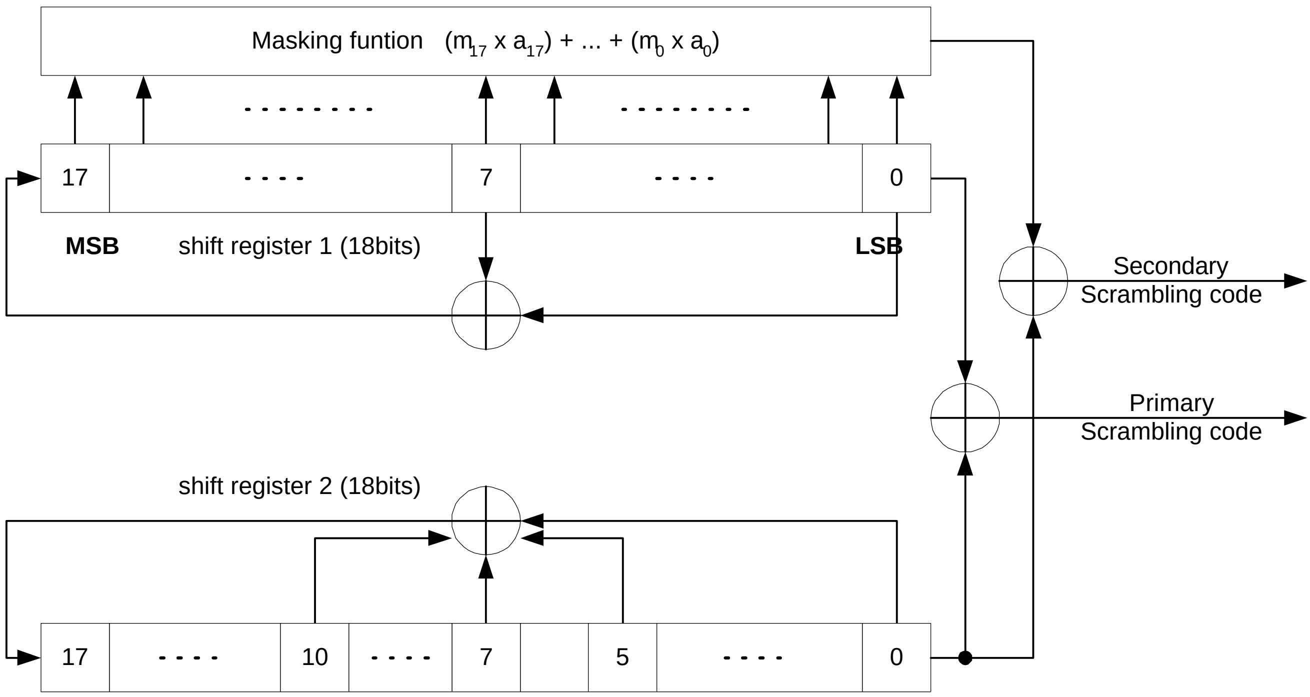


Figure 1. Configuration of downlink scrambling code generator