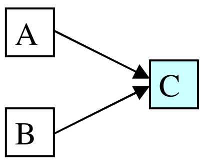
# TELE303 Lab 2 Tutorial

*Direct Sequence Spread Spectrum (DSSS) and CDMA*

Today we have a study of *Spread Spectrum* in a digital form (i.e., with signalling stripped away). The so-called ‘direct sequence’ technique, involves with chipping (i.e. multiplying) a binary data bit with a longer chip sequence whenever a sender transmits. On the receiver’s end, the data bit can be restored using the same chip sequence. Based on this encoding and decoding scheme, CDMA employs specially designed, different chips for different users, so that there is little confusion between users’ transmissions that happen at the same time.

We look at a simple scenario, shown as follows. User A uses a PN chip of [1 1 0 1 1 0], User B uses [1 1 0 0 1 1]. User C uses the PN chip of either A or B to decode the data sent by A or B respectively.




First, we need to translate the chips into bipolar representation (1/-1) so as to get ready for DSSS calculation. For consistency, let us use 1→1, 0→-1. For decoding (i.e., a reverse translation), a positive sum will mean a “1” bit, and a negative sum stands for “0”. If the sum happens to be zero, simply put an “X” (meaning unknown or to be ignored).

Here is an example case, which shows that C will recover the ‘~~0~~1’ bit from A using Chip A. What is the result if C tries to decode the received signal using Chip B?

Data encoded using a chip

Scenario 1: A sends 1; B does not send

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Chip A ($c_a$) | 1 | 1 | -1 | 1 | 1 | -1 | Sum Across / Decoding |
| A sends “1” | 1 | 1 | -1 | 1 | 1 | -1 | |
| Received ($s$) | 1 | 1 | -1 | 1 | 1 | -1 | |
| $s * c_a$ | 1 | 1 | 1 | 1 | 1 | 1 | 6→ “1” |
| Chip B ($c_b$) | 1 | 1 | -1 | -1 | 1 | 1 | |
| $s * c_b$ | | | | | | | |

Received encoded data from ALL transmitters

Multiply column-by-column Received and the Chip

Sum across the ‘green’ row and then decode the sum

In the following scenarios given in the data tables, complete the encoding and decoding processes and work out the decoded data bits. Are the data correctly recovered?

Scenario 2: A does not send; B sends 0

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Chip B ($c_b$) | 1 | 1 | -1 | -1 | 1 | 1 | Sum Across / Decoding |
| B sends "0" | | | | | | | |
| Received ($s$) | | | | | | | |
| $s * c_b$ | | | | | | | |

Scenario 3: A sends 0; B sends 1

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Chip A | | | | | | | Sum Across / Decoding |
| Chip B | | | | | | | |
| A sends "0" | | | | | | | |
| B sends "1" | | | | | | | |
| $s$ | | | | | | | |
| $s * c_a$ | | | | | | | |
| $s * c_b$ | | | | | | | |

Scenario 4: A sends 0 in double power; B sends 1

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Chip A | | | | | | | Sum Across / Decoding |
| Chip B | | | | | | | |
| A sends "0" | | | | | | | |
| B sends "1" | | | | | | | |
| $s$ | | | | | | | |
| $s * c_a$ | | | | | | | |
| $s * c_b$ | | | | | | | |

Scenario 5: A sends 0; B sends 1 in triple power

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Chip A | | | | | | | Sum Across / Decoding |
| Chip B | | | | | | | |
| A sends "0" | | | | | | | |
| B sends "1" | | | | | | | |
| $s$ | | | | | | | |
| $s * c_a$ | | | | | | | |
| $s * c_b$ | | | | | | | |