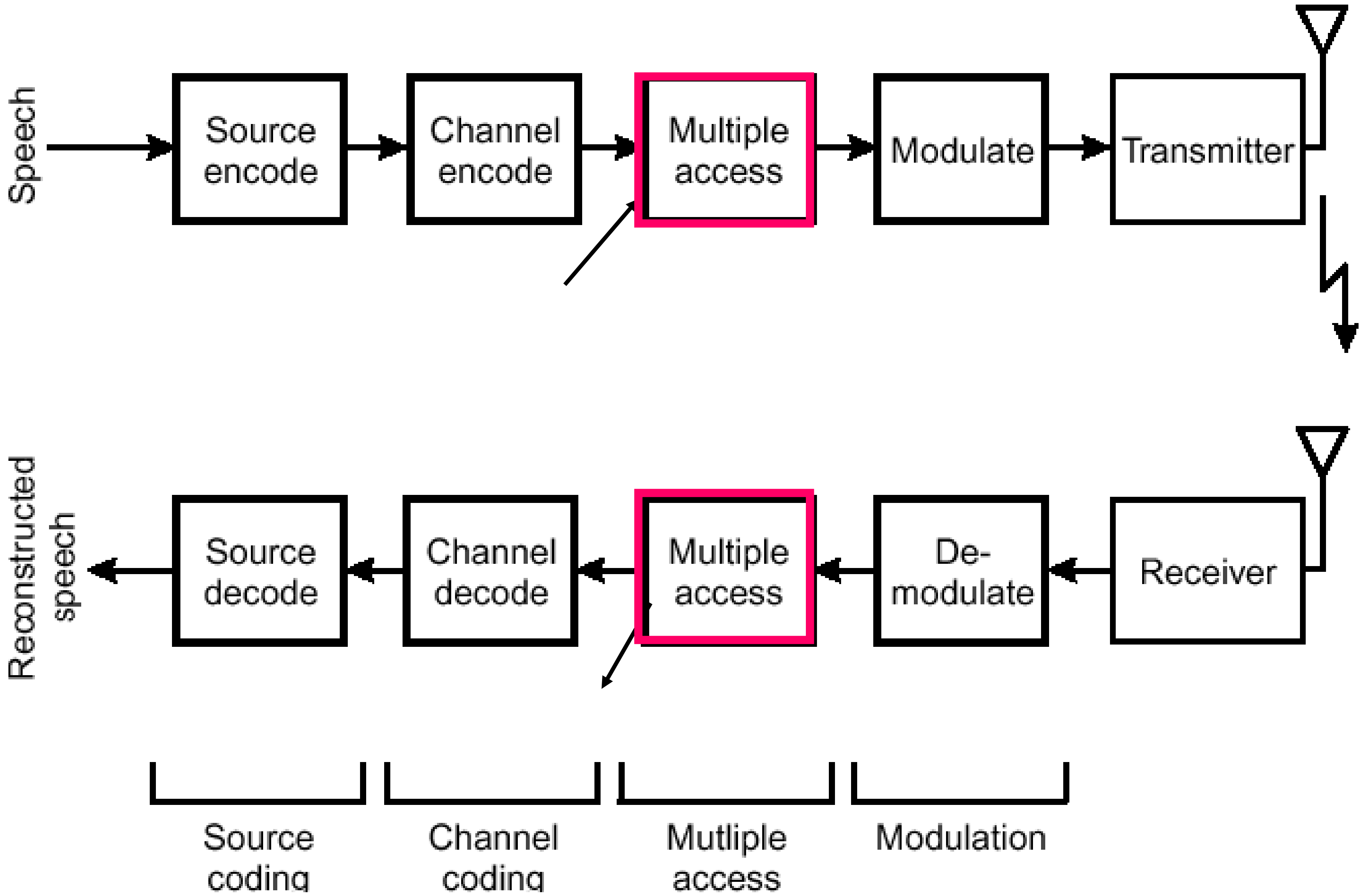
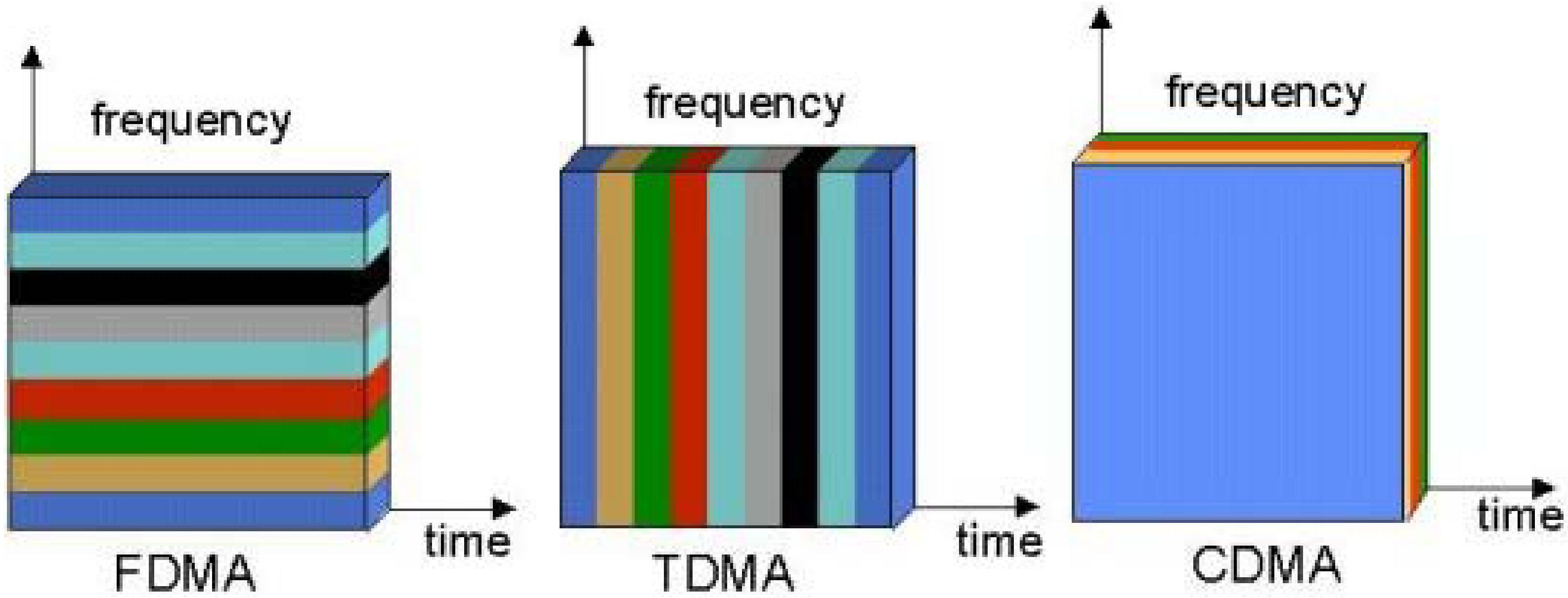
Spread spectrum and CDMA

Welcome to the World of CDMA


Speech
Source encode
Channel encode
Multiple access
Modulate
Transmitter
Reconstructed speech
Source decode
Channel decode
Multiple access
De-modulate
Receiver
Source coding
Channel coding
Mutliple access
Modulation


frequency
time
FDMA
frequency
time
TDMA
frequency
time
CDMA

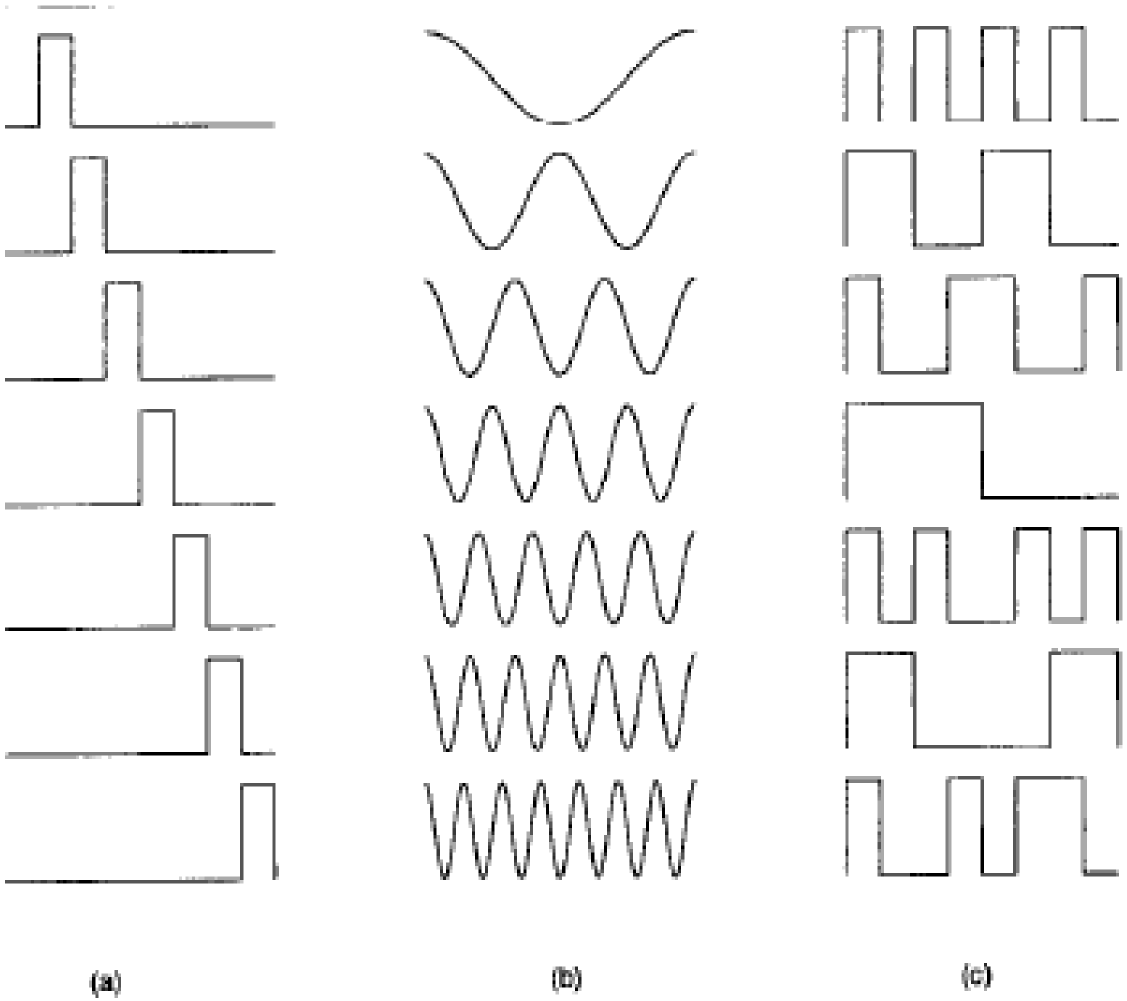


Figure 1.7 TDMA and FDMA as special cases of generic CDMA.

Spread Spectrum Techniques

By far the most popular spreading techniques are

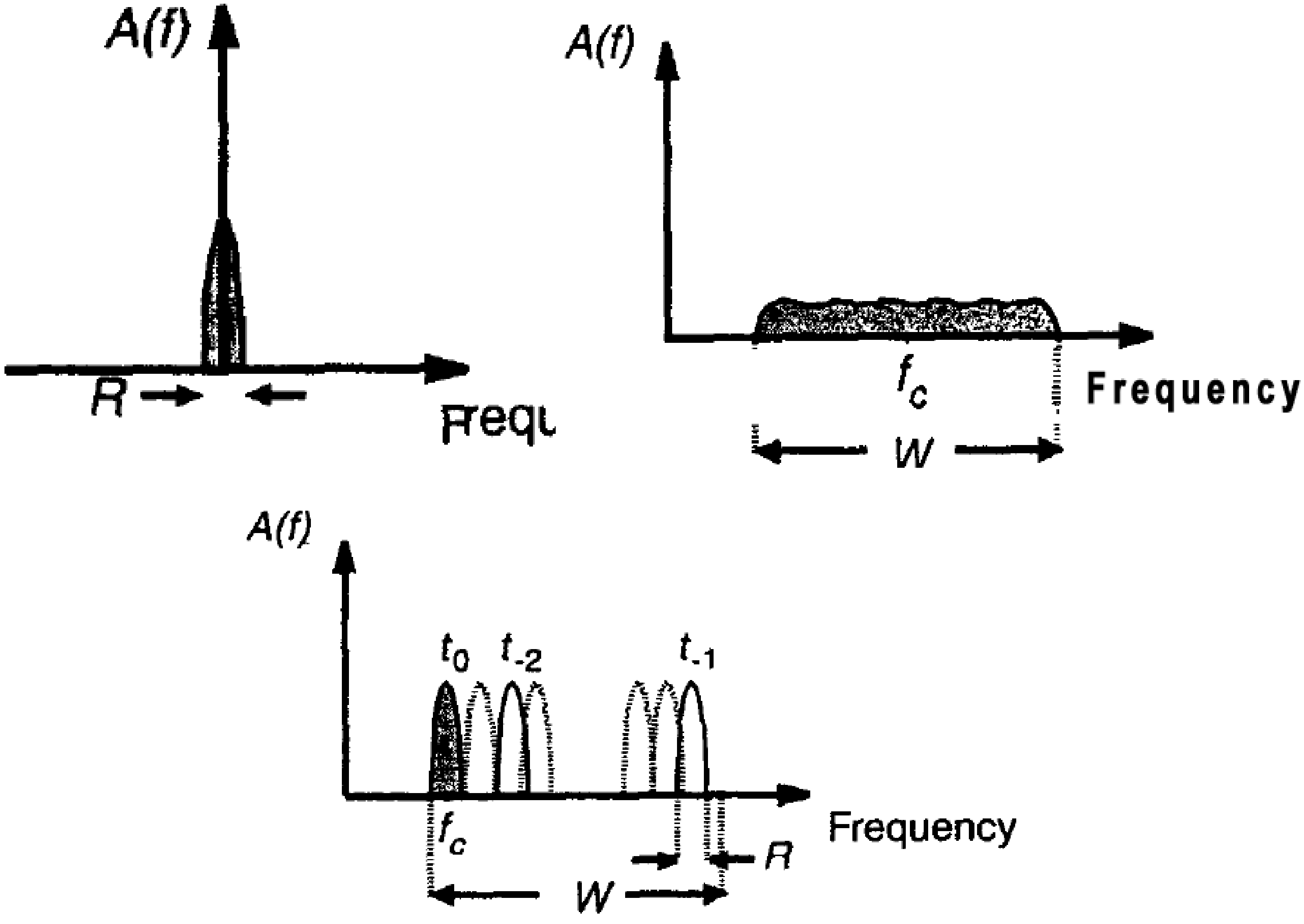
1) Direct sequence (DS) modulation.
2) Frequency hopping (FH) modulation.
3) Time hopping.
4) Hybrid (S.S) :

   *DS/FH

   *DS/TH

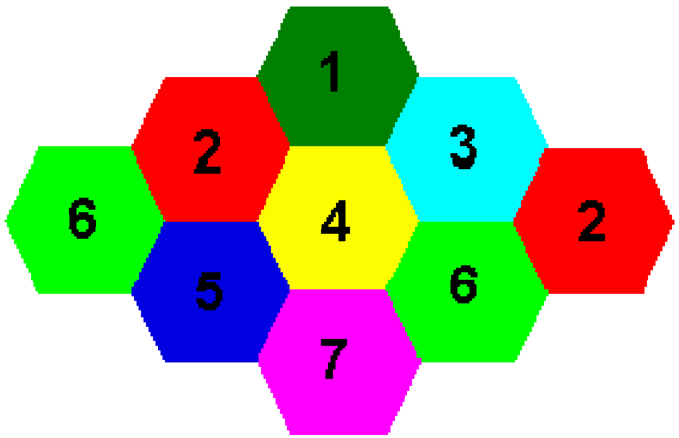
   *FH/TH

   *DS/FH/TH

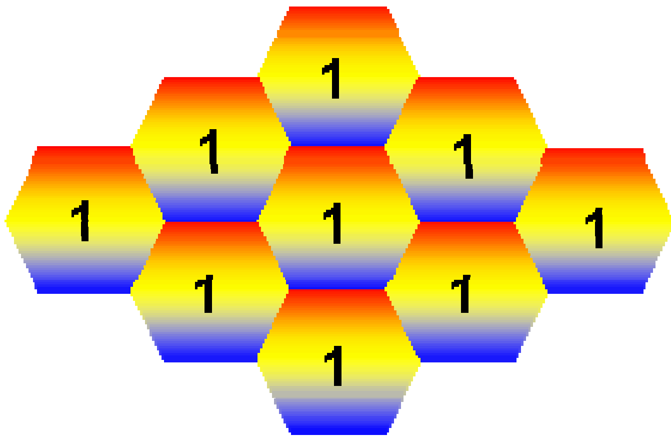


(c)

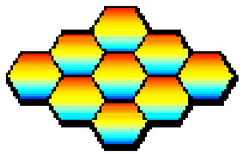
Cellular Frequency Reuse Patterns



FDMA Reuse



CDMA Reuse



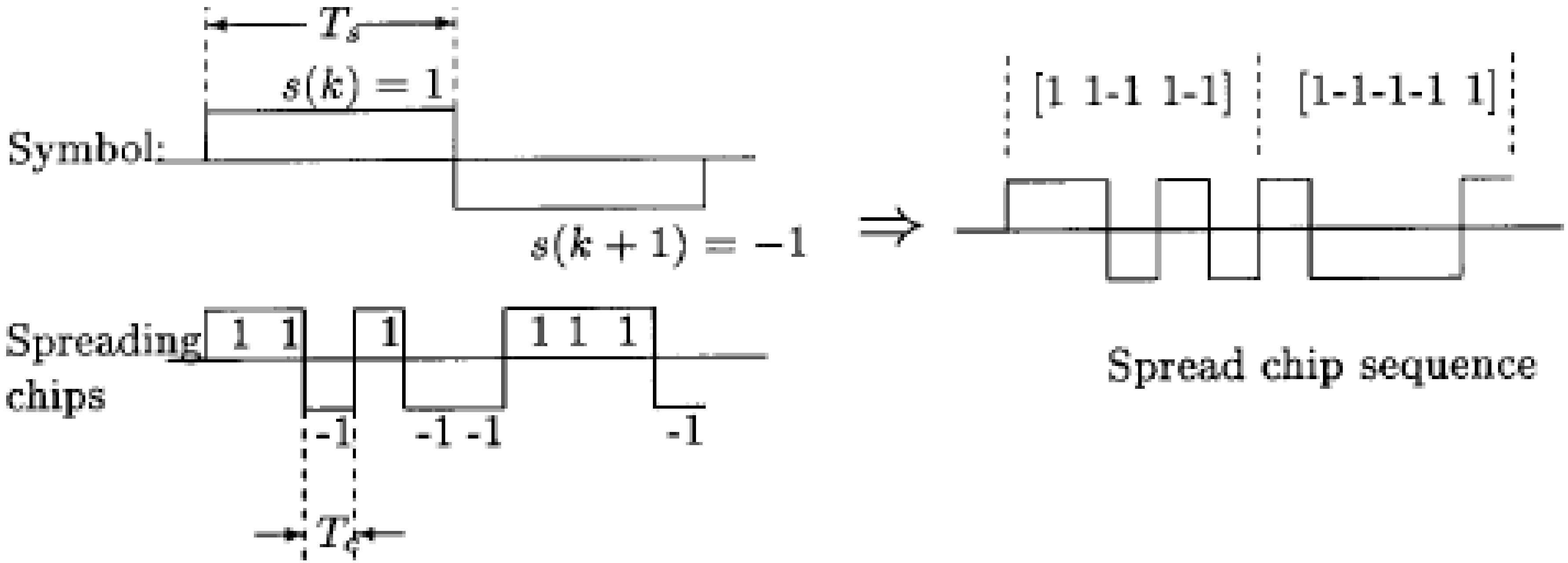


Figure 1.2 Spreading in SS communications.

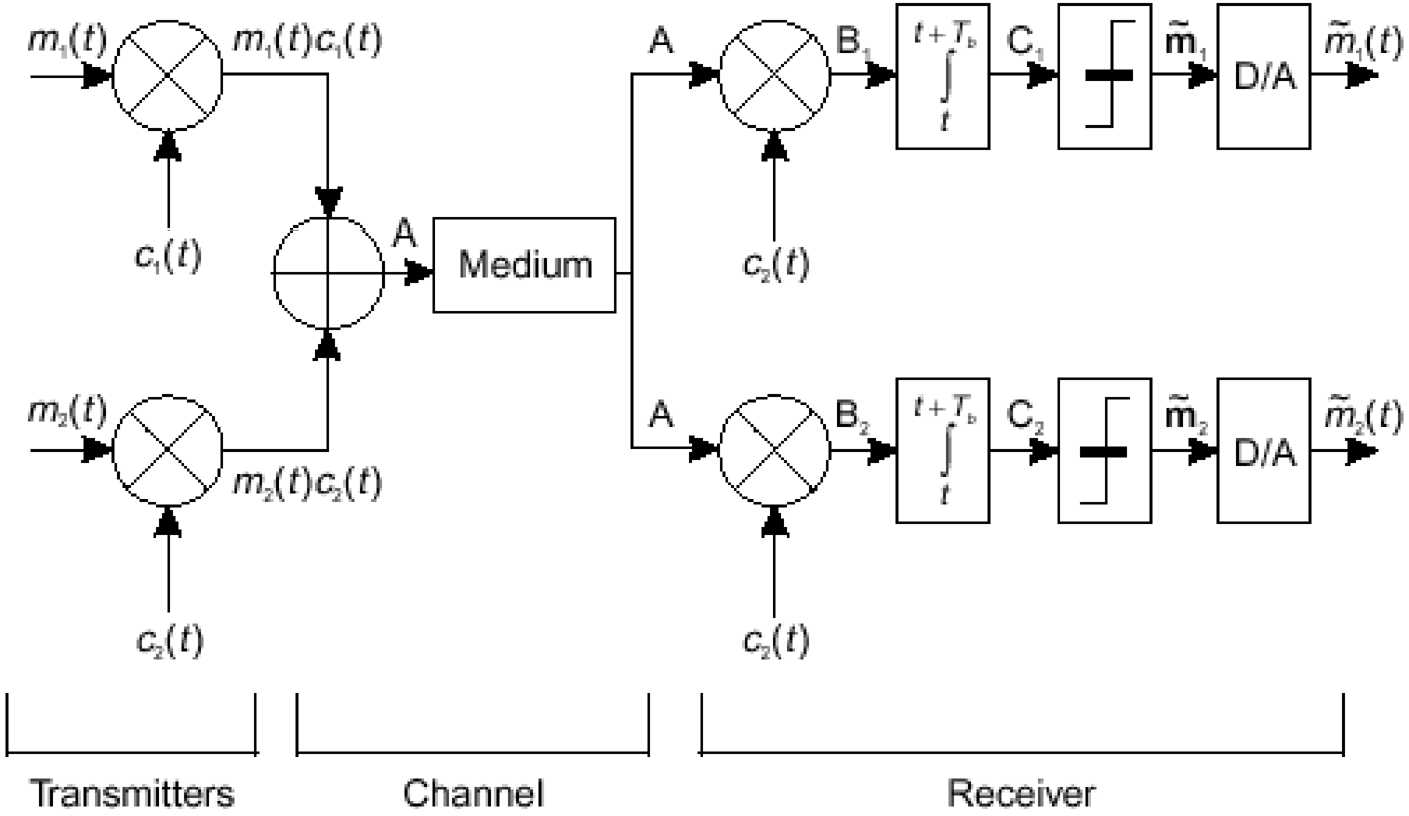


Figure 1.1 An example showing the operating principle of DS-SS multiple access. Two users are sending two separate messages, $m_1(t)$ and $m_2(t)$, simultaneously through the same channel in the same frequency band and at the same time. Through the use of orthogonal codes $c_1(t)$ and $c_2(t)$, the receiver recovers the

By: Dr.Mohab Mangoud

Multiple access using spread spectrum

Traditional ways of separating signals in time (i.e., *time division multiple access*, (TDMA)), or in frequency (i.e., FDMA) are relatively simple ways of making sure that the signals are orthogonal and noninterfering. However, in CDMA, different users occupy the same bandwidth at the same time, but are separated from each other via the use of a set of orthogonal waveforms, sequences, or codes. Two real-valued waveforms x and y are said to be orthogonal if their *cross-correlation* $R_{xy}(0)$ over T is zero, where

$$R_{xy}(0) = \int_0^T x(t)y(t)dt \qquad (1.1)$$

In discrete time, the two sequences **x** and **y** are orthogonal if their *cross-product* $R_{xy}(0)$ is zero. The cross product is defined as

$$R_{xy}(0) = \mathbf{x}^T\mathbf{y} = \sum_{i=1}^{I} x_i y_i \qquad (1.2)$$

where

$$\mathbf{x}^T = [x_1 \quad x_2 \quad \cdots \quad x_I]$$

By: Dr.Mohab Mangoud

Orthogonal code properties

Note that T denotes the transpose of the column vector, which is another representation of a sequence of numbers. For example, the following two sequences or codes, **x** and **y**, are orthogonal:

$$\mathbf{x}^T = \begin{bmatrix} -1 & -1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{y}^T = \begin{bmatrix} -1 & 1 & 1 & -1 \end{bmatrix}$$

because their cross-correlation is zero; that is,

$$R_{xy}(0) = \mathbf{x}^T \mathbf{y} = (-1)(-1) + (-1)(1) + (1)(1) + (1)(-1) = 0$$

Orthogonal code properties

1. The cross-correlation should be zero or very small.
2. Each sequence in the set has an equal number of 1s and –1s, or the number of 1s differs from the number of –1s by at most 1.
3. The scaled dot product of each code should be equal to 1.

- ***Correlation Property***
- ***Balance Property***
- ***Run Property***

Codes for CDMA

Codes are classified into two families:

- **Orthogonal Codes.**
- **Pseudorandom Noise (PN) Codes.**

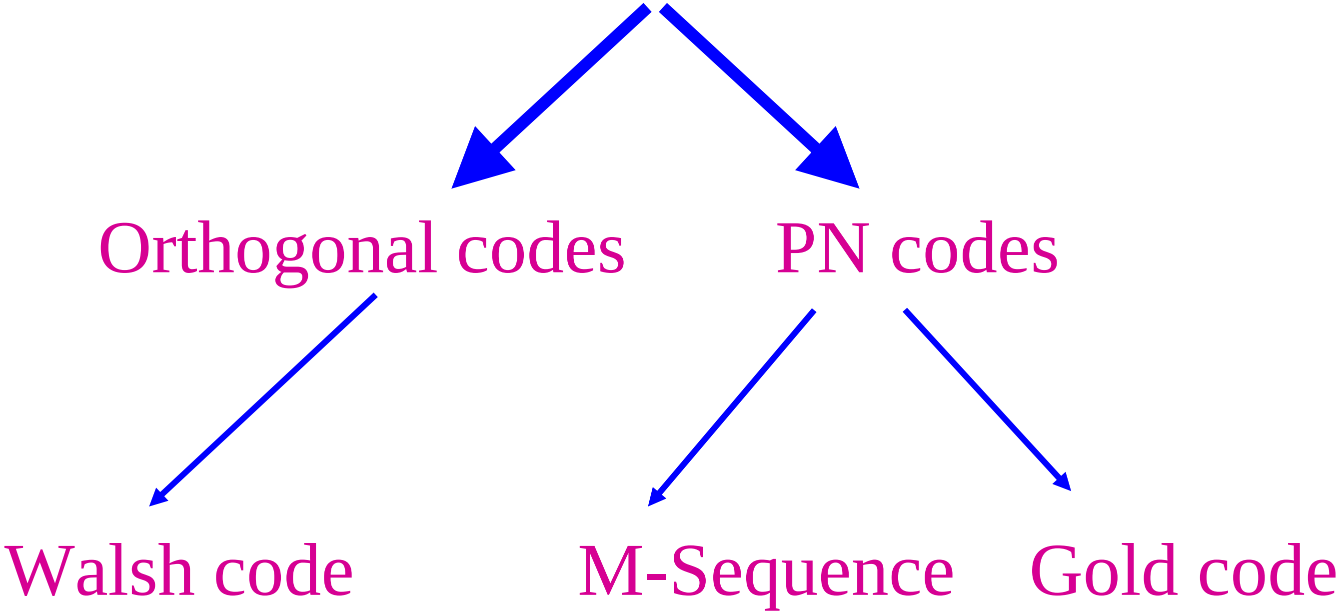
Kinds of Orthogonal Codes:

1-) Walsh Codes
2-) Orthogonal Gold Codes
3-) Multi-rate Orthogonal Gold Codes

Kinds of Pseudorandom Noise (PN) Codes:

1-) Maximal Length Sequences
2-) Gold Codes
3-) Kasami Sequences
4-) Barker Codes.

Codes in CDMA



Orthogonal Codes

Orthogonal functions have zero cross-correlation. Two binary sequences are orthogonal if the process of "XORing" them results in an equal number of 1's and 0's

Example:

```
 0000
 0101
 0101
```

The disadvantage here that they have a large cross-correlation value with different offsets, much larger than PN codes.

So Orthogonal Codes have an application in perfectly synchronized environments such as in the forward link of IS-95.

The best known technique to generate orthogonal codes is the Hadamard transform. Sometimes a modified Hadamard transform is applied.

3.5.1 Walsh Codes

3.5.1.1 Generation of Walsh Codes

Figure 3.9 shows that in a CDMA system, all the users are transmitted in the same RF band. In order to avoid mutual interference on the forward link, Walsh codes are used to separate individual users while they simultaneously occupy the same RF band. Walsh codes as used in IS-95 are a set of 64 binary orthogonal sequences. These sequences are orthogonal to each other, and they are generated by using the Hadamard matrix. Recursion is used to generate higher order matrices from lower order ones; that is,

$$\mathbf{H}_{2N} = \begin{bmatrix} \mathbf{H}_N & \mathbf{H}_N \\ \mathbf{H}_N & \overline{\mathbf{H}}_N \end{bmatrix} \tag{3.6}$$

where $\overline{\mathbf{H}}_N$ contains the same but inverted elements of $\mathbf{H}_N$. The seed matrix is

$$\mathbf{H}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{3.7}$$

Therefore, to derive a set of four orthogonal Walsh sequences $\mathbf{w}_0$, $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$, we only need to generate a Hadamard matrix of order 4, or

By: Dr.Mohab Mangoud

$$H_4 = \begin{bmatrix} H_2 & H_2 \\ H_2 & \overline{H}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

The four orthogonal sequences in this Walsh code set are taken from the rows of the matrix H_4; that is,

$$\mathbf{w}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\mathbf{w}_1 = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}$$
$$\mathbf{w}_2 = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}$$
$$\mathbf{w}_3 = \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}$$

For DS-SS multiple access, Section 1.2 specifies three conditions that must be met by a set of orthogonal sequences. The three conditions are

1. The cross-correlation should be zero or very small.
2. Each sequence in the set has an equal number of 1s and −1s, or the number of 1s differs from the number of −1s by at most one.
3. The scaled dot product of each code should equal to 1.

By changing the 0s to −1s in each of the four sequences above, that is,

$$\mathbf{w}_0 = [-1 \quad -1 \quad -1 \quad -1]$$
$$\mathbf{w}_1 = [-1 \quad +1 \quad -1 \quad +1]$$
$$\mathbf{w}_2 = [-1 \quad -1 \quad +1 \quad +1]$$
$$\mathbf{w}_3 = [-1 \quad +1 \quad +1 \quad -1]$$

By: Dr.Mohab Mangoud

we can facilitate the calculation of cross products and dot products. The readers can easily verify that all of the above sequences except $\mathbf{w}_0$ satisfy the conditions. In general, the 0th Walsh sequence consists of all −1s and thus cannot be used for channelization. In the IS-95 CDMA system, $\mathbf{w}_0$ is not used to transmit any baseband information.

Equation (3.6) can be recursively used to generate Hadamard matrices of higher orders in order to obtain larger sets of orthogonal sequences. For example, 8 orthogonal sequences, each of length 8, can be obtained by generating $\mathbf{H}_8$; 16 orthogonal sequences, each of length 16, can be obtained by generating $\mathbf{H}_{16}$. The IS-95 forward link uses a set of 64 orthogonal Walsh sequences, thus the physical limitation on the number of channels on the forward link is 63 because in an IS-95 system, $\mathbf{w}_0$ is not used to transmit any baseband information.

:xample 3.1

Equation (3.6) can be used to generate H_8, which is

$$\mathbf{H}_8 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

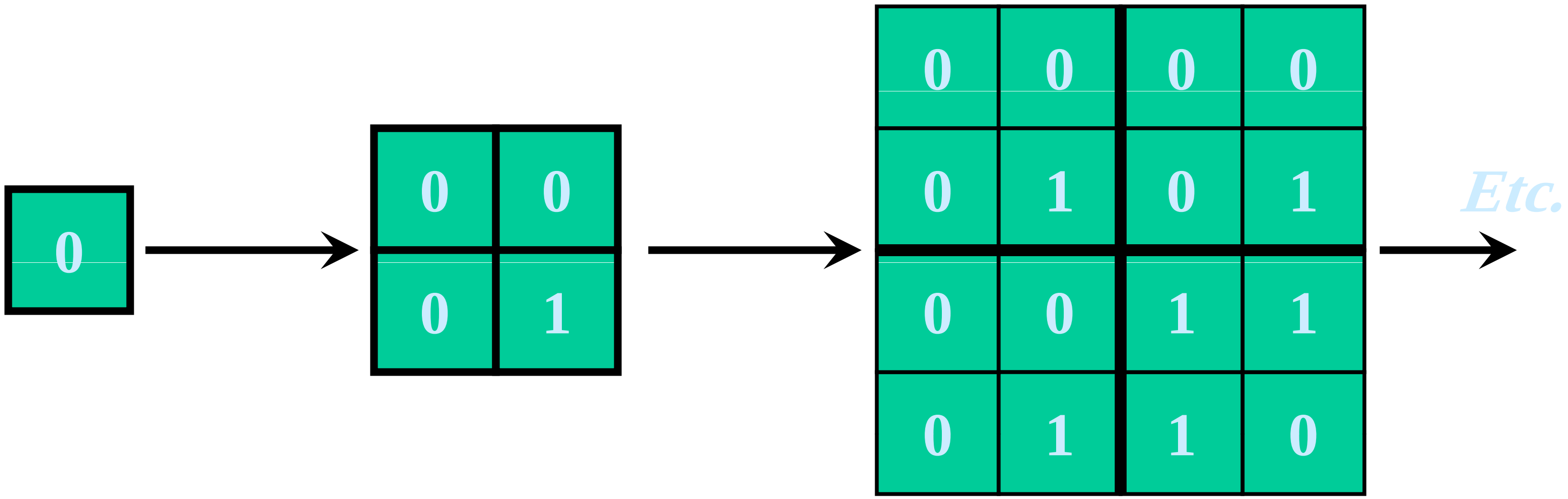
The eight resulting orthogonal Walsh codes are

$$\mathbf{w}_0 = [-1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1]$$
$$\mathbf{w}_1 = [-1 \quad +1 \quad -1 \quad +1 \quad -1 \quad +1 \quad -1 \quad +1]$$
$$\mathbf{w}_2 = [-1 \quad -1 \quad +1 \quad +1 \quad -1 \quad -1 \quad +1 \quad +1]$$
$$\mathbf{w}_3 = [-1 \quad +1 \quad +1 \quad -1 \quad -1 \quad +1 \quad +1 \quad -1]$$
$$\mathbf{w}_4 = [-1 \quad -1 \quad -1 \quad -1 \quad +1 \quad +1 \quad +1 \quad +1]$$
$$\mathbf{w}_5 = [-1 \quad +1 \quad -1 \quad +1 \quad +1 \quad -1 \quad +1 \quad -1]$$
$$\mathbf{w}_6 = [-1 \quad -1 \quad +1 \quad +1 \quad +1 \quad +1 \quad -1 \quad -1]$$
$$\mathbf{w}_7 = [-1 \quad +1 \quad +1 \quad -1 \quad +1 \quad -1 \quad -1 \quad +1]$$

By: Dr.Mohab Mangoud

Walsh Codes

$$H_{2n} = \begin{bmatrix} H_n & H_n \\ H_n & \overline{H}_n \end{bmatrix}$$



By: Dr.Mohab Mangoud

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
02	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
03	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
04	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
05	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
06	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
07	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
08	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
09	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
10	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
11	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
12	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
13	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
14	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
15	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
17	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
18	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
19	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
20	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
21	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1
22	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1
23	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0
24	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
25	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1
26	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1
27	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0
28	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
29	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0
30	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0
31	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
33	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
34	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
35	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
36	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
37	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1
38	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1
39	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0
40	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
41	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1
42	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1
43	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0
44	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
45	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0
46	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0
47	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1
48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
49	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
50	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
51	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
52	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
53	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
54	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
55	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
56	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
57	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
58	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0
59	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
60	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
61	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1
62	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	1	1
63	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

3.5.1.2 Channelization Using Walsh Codes

The following example illustrates how Walsh codes can be used for DS-SS multiple access. Suppose that there are three different users, and each user wishes to send a separate message. The separate messages are

$$\mathbf{m}_1 = [+1 \quad -1 \quad +1] \qquad \mathbf{m}_2 = [+1 \quad +1 \quad -1] \qquad \mathbf{m}_3 [-1 \quad +1 \quad +1]$$

Each of the three users is assigned a Walsh code, respectively:

$$\mathbf{w}_1 = \begin{bmatrix} -1 & +1 & -1 & +1 \end{bmatrix}$$

$$\mathbf{w}_2 = \begin{bmatrix} -1 & -1 & +1 & +1 \end{bmatrix}$$

$$\mathbf{w}_3 = \begin{bmatrix} -1 & +1 & +1 & -1 \end{bmatrix}$$

Each message is spread by its assigned Walsh code. Note that the chip rate of the Walsh code is four times the bit rate of the message, contributing to a processing gain of 4. For message one:

$m_1(t)$	1				−1				1			
$m_1(t)$	1	1	1	1	−1	−1	−1	−1	1	1	1	1
$w_1(t)$	−1	1	−1	1	−1	1	−1	1	−1	1	−1	1
$m_1(t)w_1(t)$	−1	1	−1	1	1	−1	1	−1	−1	1	−1	1

Note that $m_1(t)w_1(t)$ is the spread-spectrum signal of the first message.

$m_2(t)$	1				1				−1			
$m_2(t)$	1	1	1	1	1	1	1	1	−1	−1	−1	−1
$w_2(t)$	−1	−1	1	1	−1	−1	1	1	−1	−1	1	1
$m_2(t)w_2(t)$	−1	−1	1	1	−1	−1	1	1	1	1	−1	−1

For message three:

$m_3(t)$	−1				1				1			
$m_3(t)$	−1	−1	−1	−1	1	1	1	1	1	1	1	1
$w_3(t)$	−1	1	1	−1	−1	1	1	−1	−1	1	1	−1
$m_3(t)w_3(t)$	1	−1	−1	1	−1	1	1	−1	−1	1	1	−1

The spread-spectrum signals for all three messages, $m_1(t)w_1(t)$, $m_2(t)w_2(t)$, and $m_3(t)w_3(t)$, are combined to form a composite signal $C(t)$; that is,

$$C(t) = m_1(t)w_1(t) + m_2(t)w_2(t) + m_3(t)w_3(t)$$

The resulting $C(t)$ is

$C(t)$	−1	−1	−1	3	−1	−1	3	−1	−1	3	−1	−1

$C(t)$ is the composite signal that is transmitted in the single RF band. If there are negligible errors during the transmission process, the receiver intercepts $C(t)$. In order to separate out the original messages $m_1(t)$, $m_2(t)$, and $m_3(t)$ from the composite signal $C(t)$, the receiver multiplies $C(t)$ by the assigned Walsh code for each message:

$C(t)w_1(t)$	1	−1	1	3	1	−1	−3	−1	1	3	1	−1
$C(t)w_2(t)$	1	1	−1	3	1	1	3	−1	1	−3	−1	−1
$C(t)w_3(t)$	1	−1	−1	−3	1	−1	3	1	1	3	−1	1

Then the receiver integrates, or adds up, all the values over each bit period. The functions $M_1(t)$, $M_2(t)$, and $M_3(t)$ are the results:

$C(t)w_1(t)$	1	−1	1	3	1	−1	−3	−1	1	3	1	−1
$M_1(t)$				4				−4				4

$C(t)w_2(t)$	1	1	−1	3	1	1	3	−1	1	−3	−1	−1
$M_2(t)$				4				4				−4

$C(t)w_3(t)$	1	−1	−1	−3	1	−1	3	1	1	3	−1	1
$M_3(t)$				−4				4				4

A “decision threshold” looks at the integrated functions $M_1(t$

By: Dr.Mohab Mangoud

$C(t)w_3(t)$	1	−1	−1	−3	1	−1	3	1	1	3	−1	1
$M_3(t)$				−4				4				4

A "decision threshold" looks at the integrated functions $M_1(t)$, $M_2(t)$, and $M_3(t)$. The decision rules used are

$$\tilde{m}(t) = 1 \quad \text{if } M(t) > 0$$
$$\tilde{m}(t) = -1 \quad \text{if } M(t) < 0$$

After applying the above decision rules, we obtain the results:

$\tilde{m}_1(t)$	1	−1	1
$\tilde{m}_2(t)$	1	1	−1
$\tilde{m}_3(t)$	−1	1	1

3.5.1.3 Concluding Remarks

We have just illustrated how orthogonal Walsh codes can be used to provide channelization of different users. However, the ability to channelize depends heavily on the orthogonality of the code sequences during *all* stages of the transmission. For example, if due to multipath delay one of the users' codes is delayed by one chip, then the delayed code is no longer orthogonal to the other (nondelayed) codes in the code set. For example, the two Walsh codes

$$\mathbf{w}_2 = [-1 \quad -1 \quad +1 \quad +1]$$
$$\mathbf{w}_3 = [-1 \quad +1 \quad +1 \quad -1]$$

are orthogonal. However, if $\mathbf{w}_3$ is delayed by one chip, that is,

$$\mathbf{w}'_3 = [-1 \quad -1 \quad +1 \quad +1]$$

then the reader can easily verify that $\mathbf{w}_2$ and $\mathbf{w}'_3$ are no longer orthogonal. Therefore, synchronization is essential for using Walsh codes for DS-SS multiple access. In practice, the IS-95 CDMA system uses a pilot channel and a sync channel to synchronize the forward link and to ensure that the link is coherent.

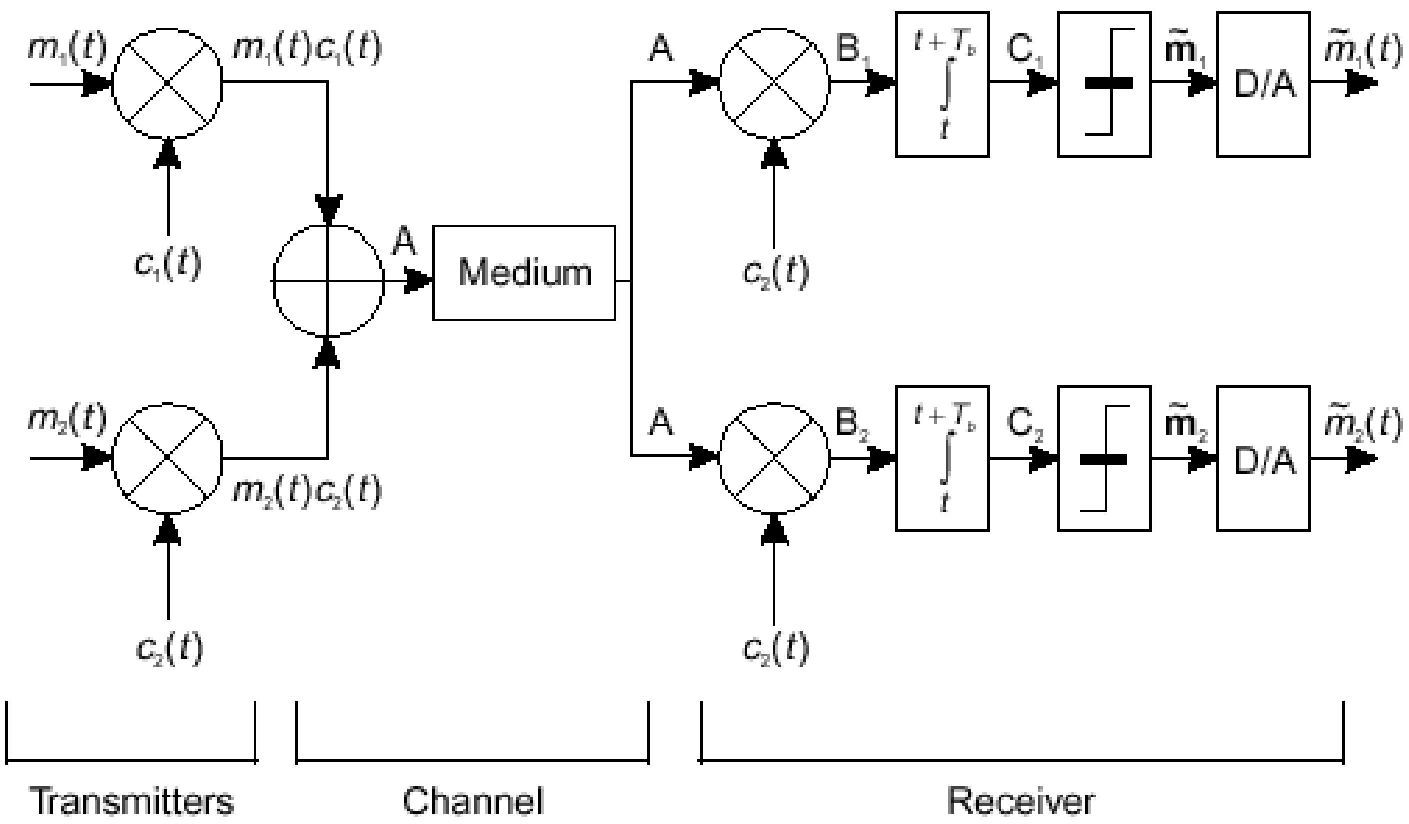


Figure 1.1 An example showing the operating principle of DS-SS multiple access. Two users are sending two separate messages, $m_1(t)$ and $m_2(t)$, simultaneously through the same channel in the same frequency band and at the same time. Through the use of orthogonal codes $c_1(t)$ and $c_2(t)$, the receiver recovers the

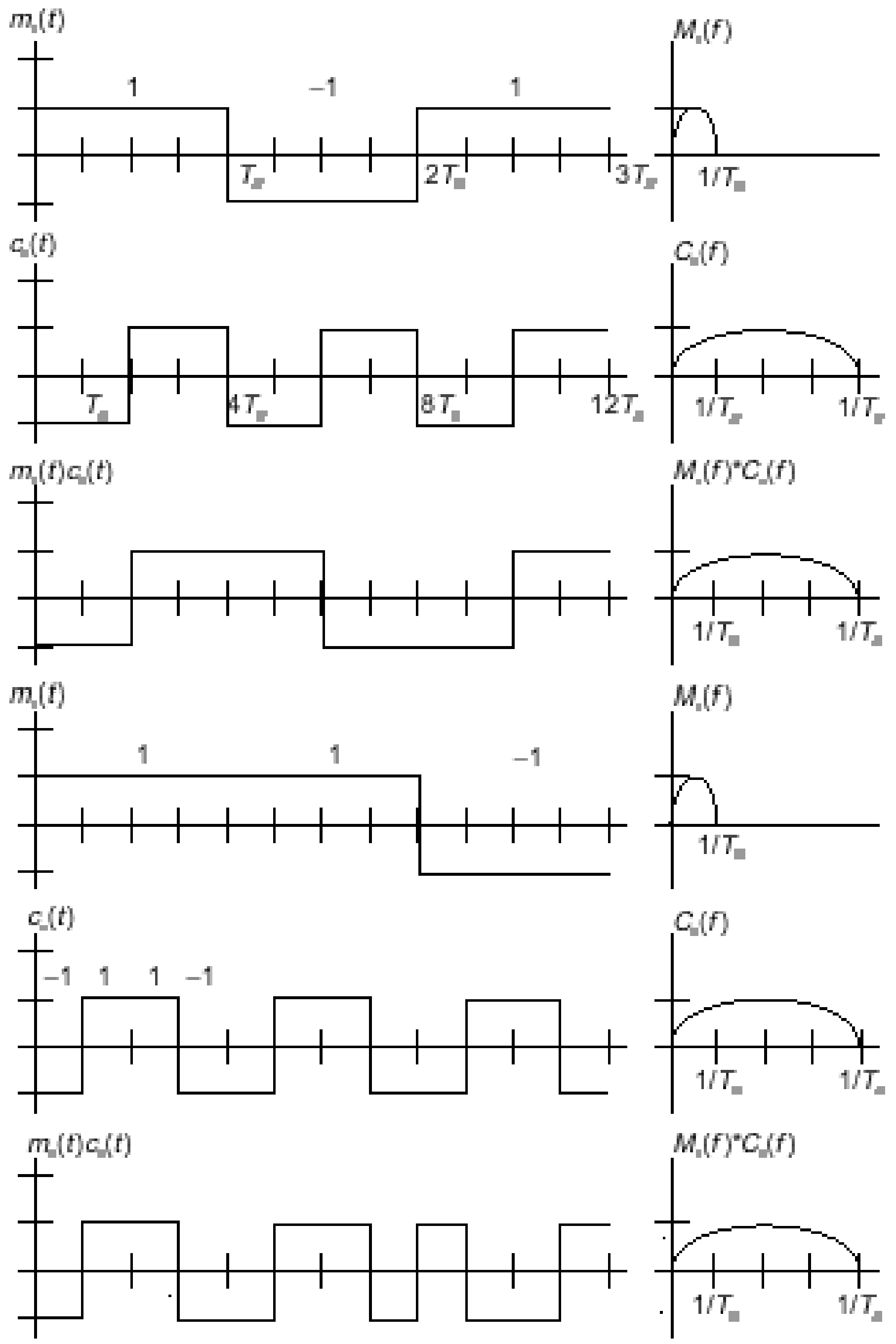


Figure 1.2 Time waveforms and frequency spectra for the baseband messages $m_i(t)$ and

By: Dr.Mohab Mangoud

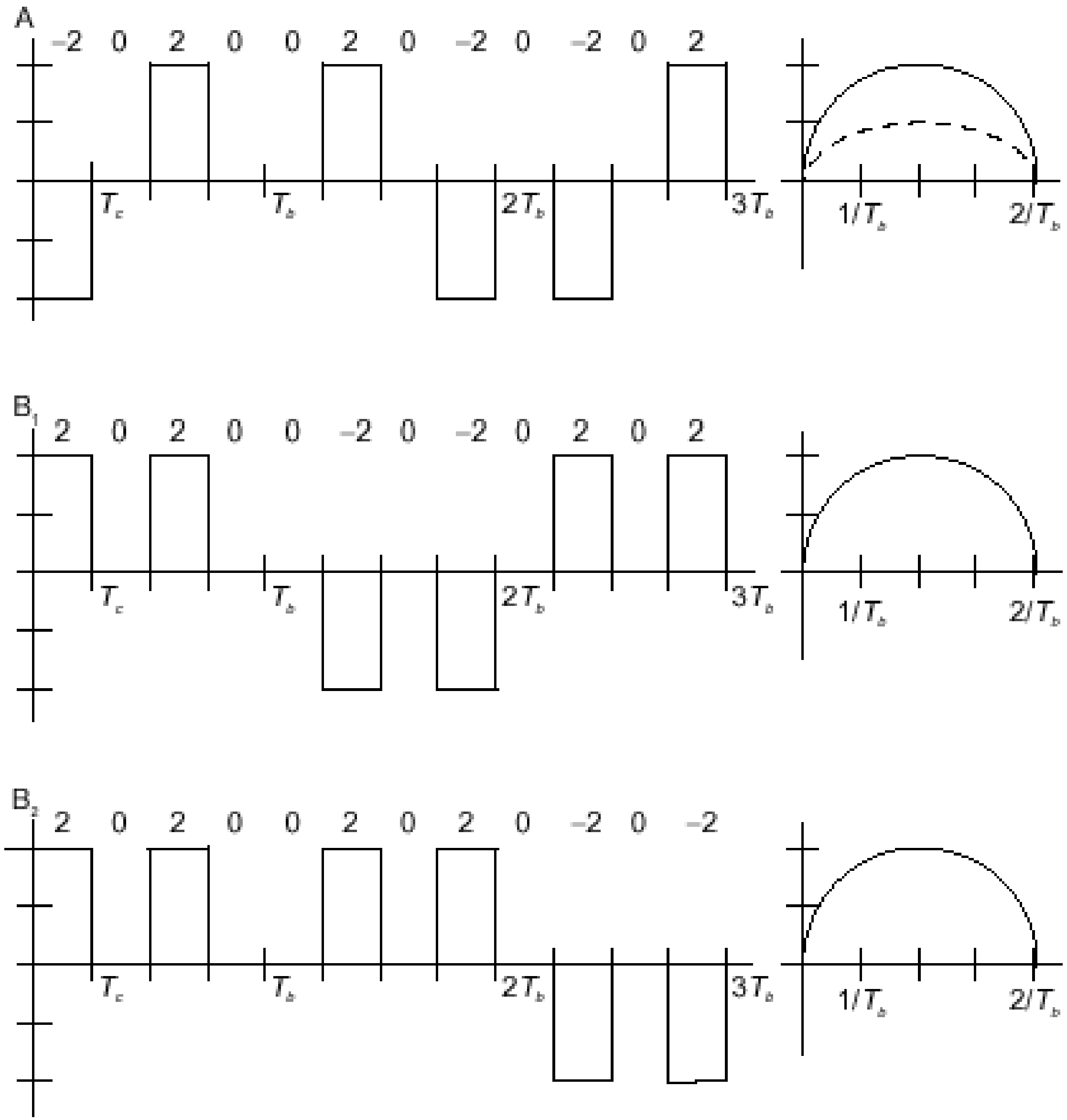


Figure 1.3 Time waveforms and frequency spectra for the signals at different points of the receiver.

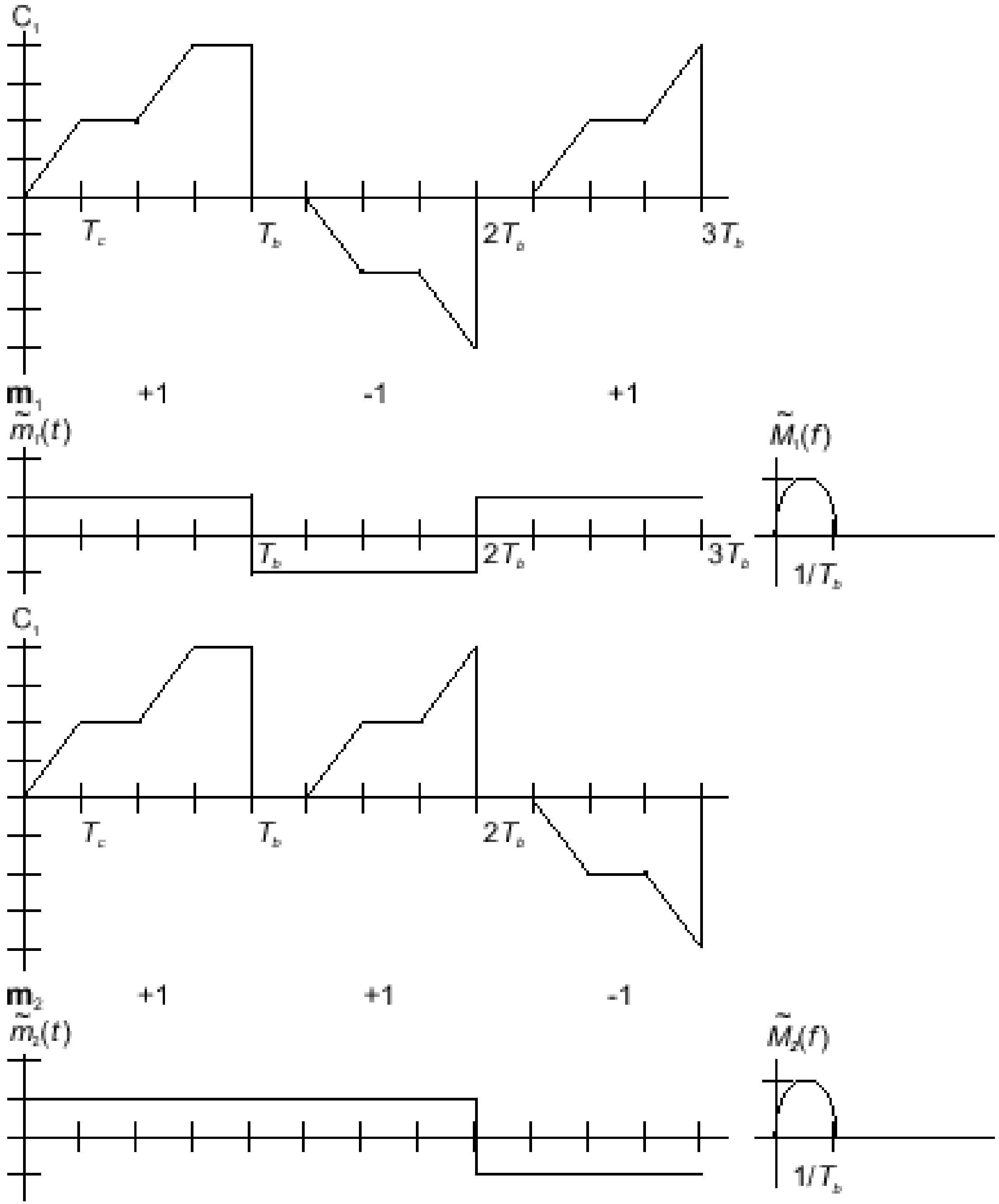
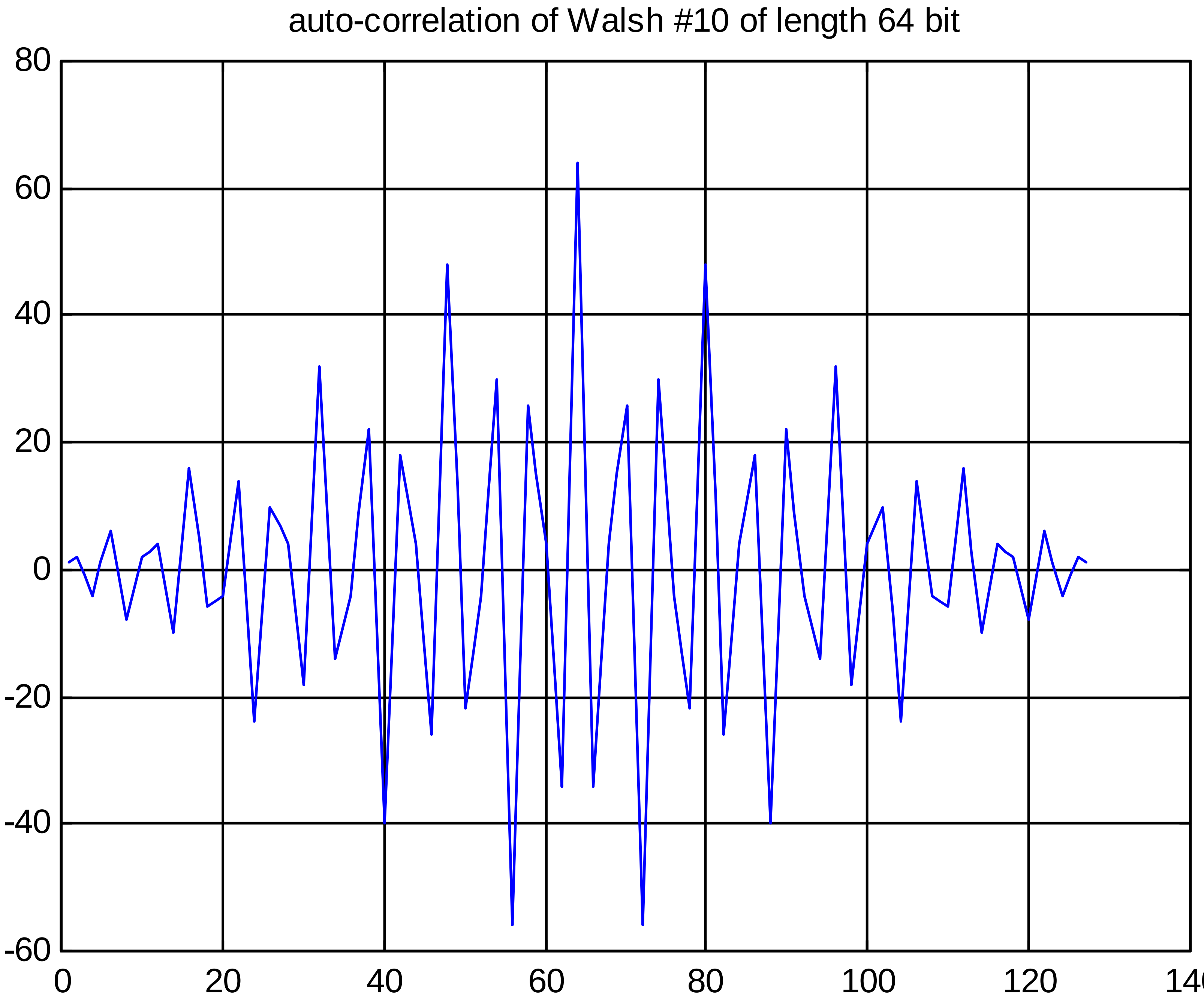
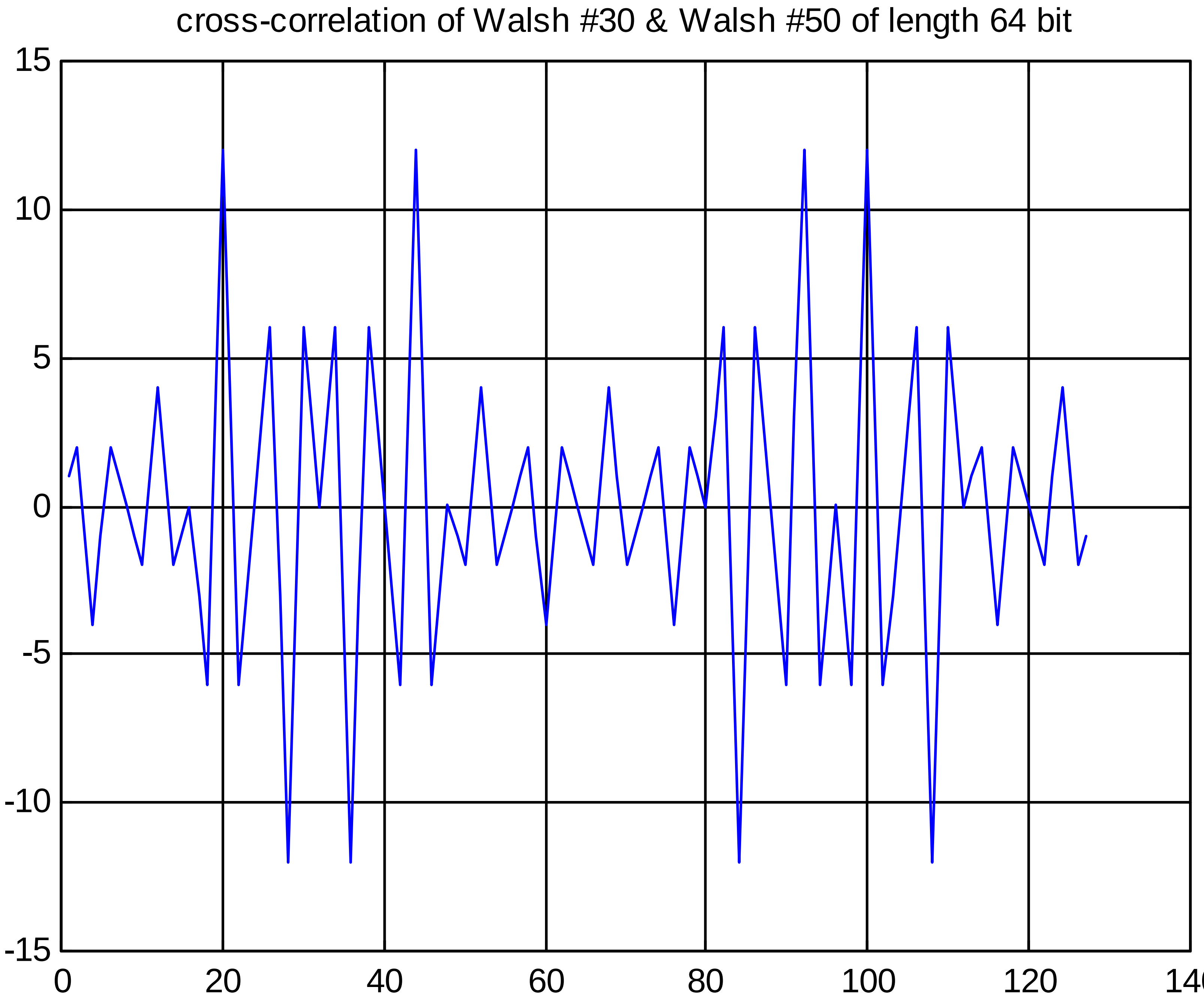


Figure 1.4 Time waveforms at the output of the integrators and decision threshold.

By: Dr.Mohab Mangoud


auto-correlation of Walsh #10 of length 64 bit
80
60
40
20
0
-20
-40
-60
0
20
40
60
80
100
120
140


cross-correlation of Walsh #30 & Walsh #50 of length 64 bit
15
10
5
0
-5
-10
-15
0
20
40
60
80
100
120
140

Capacity

Processing gain=
bandwidth expansion factor

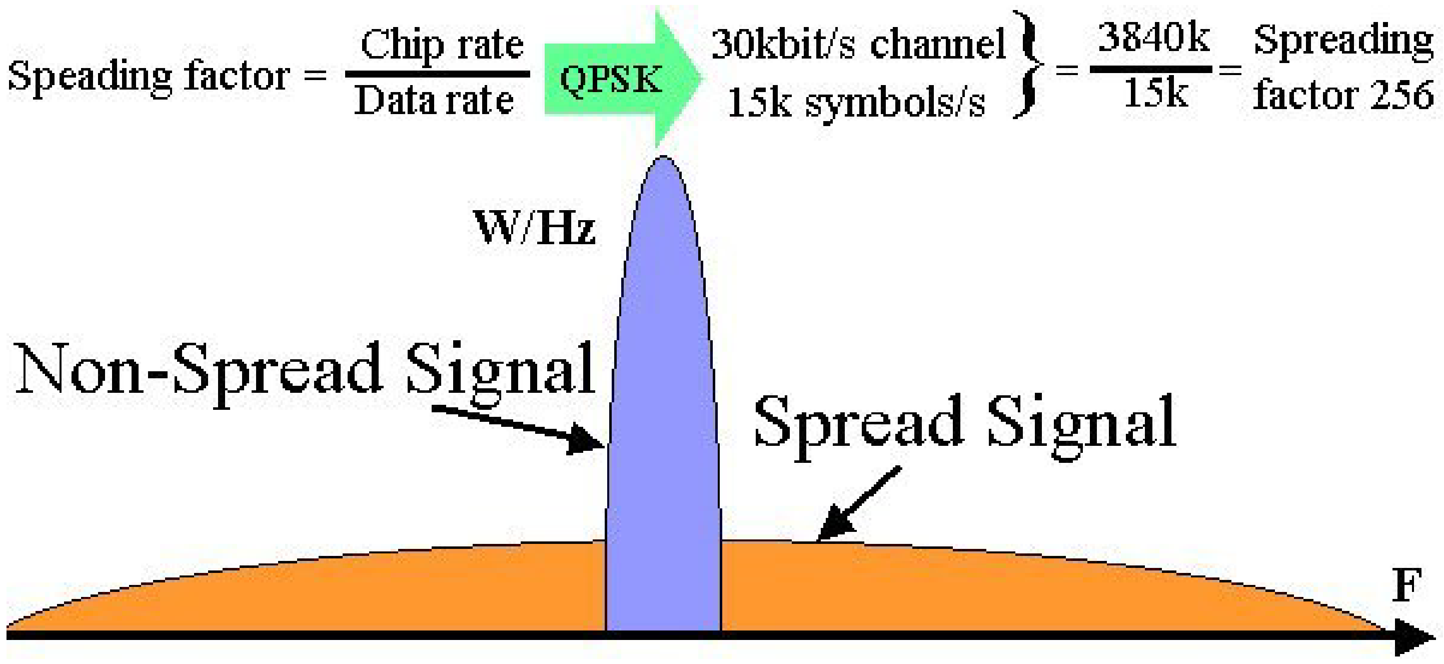
Capacity

$$M \approx \frac{(W/R)}{(E_b/N_0)}$$

Signal –noise ratio

Example: WCDMA: THE SPREADING PROCESS

WCDMA uses Direct Sequence spreading, where spreading process is done by directly combining the baseband information to high chip rate binary code. The Spreading Factor is the ratio of the chips (UMTS = 3.84Mchips/s) to baseband information rate. Spreading factors vary from 4 to 512 in FDD UMTS. Spreading process gain can in expressed in dBs (Spreading factor 128 = 21dB gain).



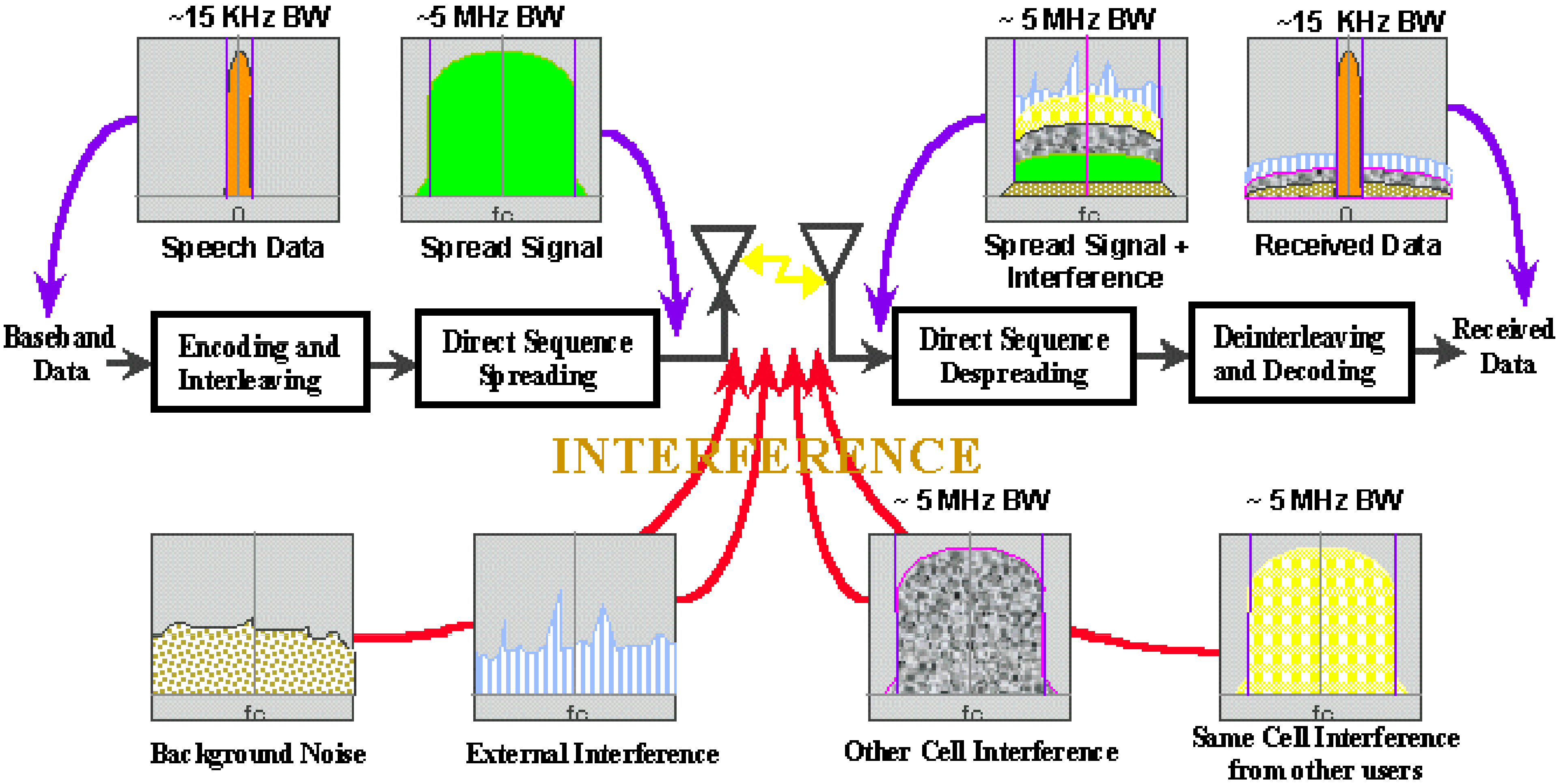
WCDMA Spreading

TDD WCDMA uses spreading factors 4 - 512 to spread the base band data over ~5MHz band. Spreading factor in dBs indicates the process gain. Spreading factor 128 = 21 dB process gain). Interference margin is calculated from that:

Interference Margin = Process Gain - (Required SNR + System Losses)

Required Signal to Noise Ration is typically about 5 dB

•System losses are defined as losses in receiver path. System losses are typically 4 - 6 dBs



Advantages of CDMA

- Improving the voice quality & eliminating the audible effects of multi-path fading.
- Enhancing privacy and security through the spreading of voice signals.
- Reducing average transmitted power
- Also reducing interference to other electronic devices.

Thank you for attending this course

- **End of the course**

Feedback survey

– **Tell me what you think about this course**

– **STOP**

– **Continue**

– **I suggest**

– **Please email me your opinion and suggestions on:**

mangoud@ eng.uob.bh

mangoud@vt.edu