Chapter 8

Designing Signals for Multiple-Access Systems

8.1 Problems

Problem 8.1 (Orthogonal signals) *If $sA1_i$ and $sB1_i$ are orthogonal, show that their complementary signals $sA0_i$ and $sB0_i$ are also orthogonal.*

(ans: If $sA1_i$ and $sB1_i$ are orthogonal, then

$$\sum_{i=0}^{n_x-1} sA1_i \, sB1_i = 0$$

If $sA1_i = sA_i$ and $sA0_i = -sA_i$, then $sA0_i = -sA1_i$ for $0 \leq i \leq n_x - 1$. Similarly, $sB0_i = -sB1_i$ for $0 \leq i \leq n_x - 1$.

Inserting these equalities into the summation above gives

$$\sum_{i=0}^{n_x-1} sA1_i \, sB1_i = \sum_{i=0}^{n_x-1} (-sA0_i)\,(-sB0_i) = \sum_{i=0}^{n_x-1} sA0_i \, sB0_i = 0$$

The last equality demonstrates that $sA0_i$ and $sB0_i$ are orthogonal.
)

Problem 8.2 (Orthogonal signal design in TDMA) *Specify signals sA_i and sB_i for $0 \leq i \leq 7$ that are orthogonal over time. Design matched processors tuned to your signals and demonstrate that they process these signals correctly.*

(ans: TDMA signals occupy separate time sub-intervals. For user A define

$$\begin{aligned} sA_i &= 1 \quad \textit{for} \quad 0 \leq i \leq 3 \\ &= 0 \quad \textit{for} \quad 4 \leq i \leq 7 \end{aligned}$$

with $\mathcal{E}_{sA} = 4$. The complementary signals are $sA1_i = sA_i$ and $sA0_i = -sA_i$.

The matched processor tuned to user A signals has coefficients $c_i = sA_i$ for $0 \le i \le 7$. Note that $c_i = 0$ for $4 \le i \le 7$.

For user B define

$$\begin{aligned} sB_i &= 0 && \textit{for} \quad 0 \le i \le 3 \\ &= 2sin(2\pi(i-4)/4) && \textit{for} \quad 4 \le i \le 7 \end{aligned}$$

The amplitude 2 gives the same signal energy $\mathcal{E}_{sB} = 4$. The complementary signals are $sB1_i = sB_i$ and $sB0_i = -sB_i$.

The matched processor tuned to user B signals has coefficients $c_i = sB_i$ for $0 \le i \le 7$. Note that $c_i = 0$ for $0 \le i \le 3$.

The processor outputs are given by

$$VA_{|X=sA1} = \sum_{i=0}^{7} sA_i\, sA1_i = \sum_{i=0}^{7} sA_i\, sA_i = \sum_{i=0}^{3} 1 = 4$$

$$VA_{|X=sA0} = \sum_{i=0}^{7} sA_i\, sA0_i = \sum_{i=0}^{7} sA_i\, (-sA_i) = -\sum_{i=0}^{3} 1 = -4$$

$$VA_{|X=sB1} = \sum_{i=0}^{7} sA_i\, sB1_i = \sum_{i=0}^{3} sA_i\,(0) + \sum_{i=4}^{7} (0)\, sB1_i = 0$$

$$VA_{|X=sB0} = \sum_{i=0}^{7} sA_i\, sB0_i = \sum_{i=0}^{3} sA_i\,(0) + \sum_{i=4}^{7} (0)\, sB0_i = 0$$

$$VB_{|X=sA1} = \sum_{i=0}^{7} sB_i\, sA1_i = \sum_{i=0}^{3} (0)\,1 + \sum_{i=4}^{7} sB_i\,(0) = 0$$

$$VB_{|X=sA0} = \sum_{i=0}^{7} sB_i\, sA0_i = \sum_{i=0}^{3} (0)\,(-1) + \sum_{i=4}^{7} sB_i\,(0) = 0$$

$$VB_{|X=sB1} = \sum_{i=0}^{7} sB_i\, sB1_i = \sum_{i=4}^{7} sB_i^2 = 4$$

$$VB_{|X=sB0} = \sum_{i=0}^{7} sB_i\, sB0_i = \sum_{i=4}^{7} sB_i\,(-sB_i) = -4$$

The matched processors work as intended.
)

Problem 8.3 (Number of orthogonal FDMA signals) *With $n_x = 8$, compute n_{max}, which is the maximum number of unique orthogonal sinusoidal sequences. Show the $n_{max} + 2$ harmonic has values that are identical to one of the complementary signals in the orthogonal set.*

(ans: With $n_x = 8$, $f_1 = 1/8$

$$n_{max} = \frac{n_x}{2} - 1 = 3$$

The $n_{max}+1$ harmonic $f_5 = 5/8$. Computing its eight values gives

i	0	1	2	3	4	5	6	7
$\sin(2\pi i(5/8))$	0	-0.707	1	-0.707	0	0.707	-1	0.707

Computing the values for f_1, f_2, and f_3 gives

i	0	1	2	3	4	5	6	7
$\sin(2\pi i(1/8))$	0	0.707	1	0.707	0	-0.707	-1	-0.707
$\sin(2\pi i(2/8))$	0	1	0	-1	0	1	0	-1
$\sin(2\pi i(3/8))$	0	0.707	-1	0.707	0	-0.707	1	-0.707

Comparing the sets of values shows $\sin(2\pi i(5/8)) = -\sin(2\pi i(3/8))$.
)

Problem 8.4 (Orthogonal signal design in FDMA) *Specify signals sA_i and sB_i for $0 \le i \le 6$, that are orthogonal in frequency. Let $T_s = 10^{-4}$ s and specify their frequency values in the range $0 \le f \le 5{,}000$ Hz. Design matched processors tuned to your signals and demonstrate that they process these signals correctly.*

(ans: Results are given for $n_x = 7$ and for $n_x = 8$.

For $n_x = 7$, $f_1 = 1/7$

$$n_{max} = \frac{n_x}{2} - 1 = 2.5$$

Hence, only $f_1 = 1/7$ and $f_2 = 2/7$ are the only valid harmonic frequencies.

With $T_s = 10^{-4}$ s. the duration of each signal is $7T_s = 7 \times 10^{-4}$ s. The fundamental analog frequency equals

$$\frac{1}{7T_s} = \frac{1}{7 \times 10^{-4}\, s} = 1.429 \text{ kHz}$$

and its second harmonic equals

$$\frac{2}{7T_s} = \frac{2}{7 \times 10^{-4}\, s} = 2.857 \text{ kHz}$$

User A's signals:

$$sA_i = \sin(2\pi f_1 i) = \sin(2\pi i/7) \quad \text{for } 0 \le i \le 6$$
$$sA1_i = sA_i$$
$$sA0_i = -sA_i$$

User B's signals:

$$sB_i = \sin(2\pi f_2 i) = \sin(4\pi i/7) \quad \text{for } 0 \le i \le 6$$
$$sB1_i = sB_i$$
$$sB0_i = -sB_i$$

Processor outputs: Use the basic signals sA_i and sB_i to demonstrate operation

$$\begin{aligned} VA_{|X=sA} &= \sum_{i=0}^{6} sA_i^2 \\ &= 0 + 0.782^2 + 0.975^2 + 0.434^2 + (-0.434)^2 + (-0.975)^2 + (-0.782)^2 \\ &= 3.5 \quad (= \mathcal{E}_{sA} = n_x/2) \end{aligned}$$

$$
\begin{aligned}
VB_{|X=sB} &= \sum_{i=0}^{6} sB_i^2 \\
&= 0 + 0.975^2 + (-0.434)^2 + (-0.782)^2 + 0.782^2 + 0.434^2 + (-0.975)^2 \\
&= 3.5 \quad (= \mathcal{E}_{sB} = n_x/2)
\end{aligned}
$$

$$
\begin{aligned}
VA_{|X=sB} &= \sum_{i=0}^{6} sA_i\, sB_i \\
&= 0 + 0.762 - 0.423 - 0.339 - 0.339 - 0.423 + 0.762 = 0
\end{aligned}
$$

For $n_x = 8$, $f_1 = 1/8$

$$n_{max} = \frac{n_x}{2} - 1 = 3$$

Hence, $f_1 = 1/8$, $f_2 = 1/4$ *and* $f_3 = 3/8$ *are the only valid harmonic frequencies.*

With $T_s = 10^{-4}$ *s. the duration of each signal is* $8T_s = 8 \times 10^{-4}$ *s. Hence, the fundamental analog frequency equals*

$$\frac{1}{8T_s} = \frac{1}{8 \times 10^{-4}\, s} = 1.25 \ \textit{kHz}$$

The second harmonic equals

$$\frac{2}{8T_s} = \frac{2}{8 \times 10^{-4}\, s} = 2.5 \ \textit{kHz}$$

The third harmonic equals

$$\frac{3}{8T_s} = \frac{3}{8 \times 10^{-4}\, s} = 3.75 \ \textit{kHz}$$

In this case we can have three users (A, B, and C):

User A's signals:

$$
\begin{gathered}
sA_i = \sin(2\pi f_1 i) = \sin(2\pi i/8) = \sin(\pi i/4) \quad \textit{for } 0 \le i \le 7 \\
sA1_i = sA_i \\
sA0_i = -sA_i
\end{gathered}
$$

User B's signals:

$$
\begin{gathered}
sB_i = \sin(2\pi f_2 i) = \sin(4\pi i/8) = \sin(\pi i/2) \quad \textit{for } 0 \le i \le 7 \\
sB1_i = sB_i \\
sB0_i = -sB_i
\end{gathered}
$$

User C's signals:

$$
\begin{gathered}
sC_i = \sin(2\pi f_3 i) = \sin(6\pi i/8) = \sin(3\pi i/4) \quad \textit{for } 0 \le i \le 7 \\
sC1_i = sC_i \\
sC0_i = -sC_i
\end{gathered}
$$

Processor outputs: Use the basic signals sA_i, sB_i, and sC_i to demonstrate operation

$$
\begin{aligned}
VA_{|X=sA} &= \sum_{i=0}^{7} sA_i^2 \\
&= 0 + 0.707^2 + 1 + 0.707^2 + 0 + (-0.707)^2 + (-1)^2 + (-0.707)^2 \\
&= 4 \quad (= \mathcal{E}_{sA} = n_x/2)
\end{aligned}
$$

$$
\begin{aligned}
VB_{|X=sB} &= \sum_{i=0}^{7} sB_i^2 \\
&= 0 + 1 + 0 + (-1)^2 + 0 + 1 + 0 + (-1)^2 \\
&= 4 \quad (= \mathcal{E}_{sB} = n_x/2)
\end{aligned}
$$

$$
\begin{aligned}
VC_{|X=sC} &= \sum_{i=0}^{7} sC_i^2 \\
&= 0 + 0.707^2 + (-1)^2 + 0.707^2 + 0 + (-0.707)^2 + 1 + (-0.707)^2 \\
&= 4 \quad (= \mathcal{E}_{sC} = n_x/2)
\end{aligned}
$$

$$
\begin{aligned}
VA_{|X=sB} &= \sum_{i=1}^{7} sA_i \, sB_i \\
&= 0 + 0.7070 - 0.707 + 0 - 0.707 + 0 + 0.707 = 0
\end{aligned}
$$

$$
\begin{aligned}
VA_{|X=sC} &= \sum_{i=0}^{7} sA_i \, sC_i \\
&= 0 + 0.5 - 1 + 0.5 + 0 + 0.5 - 1 + 0.5 = 0
\end{aligned}
$$

$$
\begin{aligned}
VB_{|X=sC} &= \sum_{i=0}^{7} sB_i \, sC_i \\
&= 0 + 0.707 + 0 - 0.707 + 0 - 0.707 + 0 + 0.707 = 0
\end{aligned}
$$

)

Problem 8.5 (Orthogonal signal design in CDMA) *Specify signals sA_i and sB_i for $0 \le i \le 4$ having values equal to either ± 1 that are approximately orthogonal in code. First, flip a coin (tail $\rightarrow -1$, and head $\rightarrow +1$) to specify the values for sA_i. Then do the same repeatedly to generate sB_i until*

$$
\epsilon = \left| \sum_{i=0}^{4} sA_i \, sB_i \right| \le 1
$$

Design matched processors tuned to your signals and demonstrate that they process these signals correctly.

(ans: Initial set of flips:

$$T, H, H, T, T \quad \rightarrow \quad sA_i = -1, +1, +1, -1, -1$$

First flips for sB_i:

$$T, H, H, H, H \quad \rightarrow \quad sB_i = -1, +1, +1, +1, +1$$

$$\epsilon = |(-1)(-1) + (+1)(+1) + (+1)(+1) + (-1)(+1) + (-1)(+1)| = 1$$

Hence, sA_i and sB_i are approximately orthogonal.
User A's signals:

$$sA1_i = sA_i$$
$$sA0_i = -sA_i$$

User B's signals:

$$sB1_i = sB_i$$
$$sB0_i = -sB_i$$

Processor outputs:

$$VA_{|X=sA1} = \sum_{i=0}^{4} sA_i^2 = 1+1+1+1+1 = 5 \quad (= \mathcal{E}_{sA} = n_x)$$

$$VA_{|X=sA0} = -\sum_{i=0}^{4} sA_i^2 = -1-1-1-1-1 = -5 \quad (= -\mathcal{E}_{sA} = -n_x)$$

$$VB_{|X=sB1} = \sum_{i=0}^{4} sB_i^2 = 1+1+1+1+1 \quad (= \mathcal{E}_{sB} = n_x)$$

$$VA_{|X=sB1} = \sum_{i=0}^{4} sA_i\, sB_i = 1+1+1-1-1 = -1$$

)

Problem 8.6 (Processing superimposed CDMA signals from users A and B) *Using the codes determined in the previous problem, let X_i equal the sum of the codes that would occur for the four possibilities of transmitted data (00, 01, 10, 11). Design matched processors tuned to your signals and demonstrate that they process these signals correctly.*

(ans: Using

$$sA_i = -1, +1, +1, -1, -1$$

$$sB_i = -1, +1, +1, +1, +1$$

For $D_{AB} = 00$

$$X_i = sA0_i + sB0_i = -sA_i - sB_i = +2, -2, -2, 0, 0$$

$$VA_X = \sum_{i=0}^{4} sA_i\, X_i = -1(2) + 1(-2) + 1(-2) - 1(0) - 1(0) = -6 \quad \rightarrow D_A = 0$$

$$VB_X = \sum_{i=0}^{4} sB_i\, X_i = -1(2) + 1(-2) + 1(-2) + 1(0) + 1(0) = -6 \quad \rightarrow D_B = 0$$

For $D_{AB} = 01$

$$X_i = sA0_i + sB1_i = -sA_i + sB_i = 0, 0, 0, +2, +2$$

$$VA_X = \sum_{i=0}^{4} sA_i\, X_i = -1(0) + 1(0) + 1(0) - 1(2) - 1(2) = -4 \quad \rightarrow D_A = 0$$

$$VB_X = \sum_{i=0}^{4} sB_i\, X_i = -1(0) + 1(0) + 1(0) + 1(2) + 1(2) = 4 \quad \rightarrow D_B = 1$$

For $D_{AB} = 10$

$$X_i = sA1_i + sB0_i = sA_i - sB_i = 0, 0, 0, -2, -2$$

$$VA_X = \sum_{i=0}^{4} sA_i\, X_i = -1(0) + 1(0) + 1(0) - 1(-2) - 1(-2) = +4 \quad \rightarrow D_A = 1$$

$$VB_X = \sum_{i=0}^{4} sB_i\, X_i = -1(0) + 1(0) + 1(0) + 1(-2) + 1(-2) = -4 \quad \rightarrow D_B = 1$$

For $D_{AB} = 11$

$$X_i = sA0_i + sB0_i = sA_i + sB_i = -2, +2, +2, 0, 0$$

$$VA_X = \sum_{i=0}^{4} sA_i\, X_i = -1(-2) + 1(2) + 1(2) - 1(0) - 1(0) = 6 \quad \rightarrow D_A = 1$$

$$VB_X = \sum_{i=0}^{4} sB_i\, X_i = -1(-2) + 1(2) + 1(2) + 1(0) + 1(0) = 6 \quad \rightarrow D_B = 1$$

)

8.2 Excel Projects

The projects set all signals to have a maximum amplitude $A_{max} = 10$. The performance as a function of σ_N for signal types starts at the value $\sigma_N = 5$ for easy comparison of signal types.

Project 8.1 (Extending TDMA to 3 simultaneous users) *Extend Example 13.44 to generate TDMA signals with $n_X = 30$ to serve three users simultaneously.*

(ans: The VBA offers an alternative to stopping automatic recalculations by storing the binary values to check for errors. Reading a cell value does not cause recalculation. The "Value" attribute appears to be the default value, so it was omitted in some commands.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	DA_t	DA_r	DB_t	DB_r	DC_t	DC_r				σ_N=		# xmits=	#errors=	P[error]=				
2	1	0	0	0	1	1				640		1002	494	0.4930				START
3																		
4	i	sA_i	sB_i	sC_i	N_i	sAT_i	sBT_i	sCTi		X_i	$sA_i X_i$	$sB_i X_i$	$sC_i X_i$		σ_N	P[error]		
5	0	10	0	0	48.03	10	0	0		58.03	580.30	0.00	0.00		5	0.000		XMIT
6	1	10	0	0	520.56	10	0	0		530.56	5305.57	0.00	0.00		7	0.000		
7	2	10	0	0	-358.71	10	0	0		-348.71	-3487.14	0.00	0.00		10	0.002		
8	3	10	0	0	-46.94	10	0	0		-36.94	-369.43	0.00	0.00		14	0.011		AUTO
9	4	10	0	0	-430.20	10	0	0		-420.20	-4201.97	0.00	0.00		20	0.059		
10	5	10	0	0	1083.20	10	0	0		1093.20	10931.96	0.00	0.00		28	0.160		
11	6	10	0	0	-1090.23	10	0	0		-1080.23	-10802.31	0.00	0.00		40	0.200		
12	7	10	0	0	-1591.05	10	0	0		-1581.05	-15810.47	0.00	0.00		57	0.283		
13	8	10	0	0	-251.83	10	0	0		-241.83	-2418.32	0.00	0.00		80	0.338		
14	9	10	0	0	167.05	10	0	0		177.05	1770.53	0.00	0.00		113	0.383		
15	10	0	-10	0	-294.22	0	10	0		-284.22	0.00	2842.25	0.00		160	0.411		
16	11	0	-10	0	941.70	0	10	0		951.70	0.00	-9517.04	0.00		226	0.465		
17	12	0	-10	0	559.73	0	10	0		569.73	0.00	-5697.26	0.00		320	0.437		
18	13	0	-10	0	1113.03	0	10	0		1123.03	0.00	-11230.28	0.00		453	0.493		
19	14	0	-10	0	-52.07	0	10	0		-42.07	0.00	420.75	0.00					
20	15	0	-10	0	302.40	0	10	0		312.40	0.00	-3123.99	0.00					
21	16	0	-10	0	-442.11	0	10	0		-432.11	0.00	4321.07	0.00					
22	17	0	-10	0	176.32	0	10	0		186.32	0.00	-1863.24	0.00		E_s = 1000, when σ_N = sqrt(1000)= 28, P[error]=0.16 (as expected). When σ_N >>E_s P[error]→ 0.5			
23	18	0	-10	0	-228.50	0	10	0		-218.50	0.00	2184.97	0.00					
24	19	0	-10	0	-311.59	0	10	0		-301.59	0.00	3015.92	0.00					
25	20	0	0	10	-97.20	0	0	10		-87.20	0.00	0.00	-871.96					
26	21	0	0	10	-61.88	0	0	10		-51.88	0.00	0.00	-518.77					
27	22	0	0	10	930.07	0	0	10		940.07	0.00	0.00	9400.72					
28	23	0	0	10	-910.16	0	0	10		-900.16	0.00	0.00	-9001.59					
29	24	0	0	10	-1318.62	0	0	10		-1308.62	0.00	0.00	-13086.21					
30	25	0	0	10	491.24	0	0	10		501.24	0.00	0.00	5012.43					
31	26	0	0	10	930.81	0	0	10		940.81	0.00	0.00	9408.10					
32	27	0	0	10	1934.50	0	0	10		1944.50	0.00	0.00	19444.99					
33	28	0	0	10	1162.68	0	0	10		1172.68	0.00	0.00	11726.80					
34	29	0	0	10	705.24	0	0	10		715.24	0.00	0.00	7152.42					
35																		
36										V=	-18501.26	-18646.86	38666.92					

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	DA_t	DA_r	DB_t	DB_r	DC_t	DC_r				σ_N=		# xmits=	#errors=	P[error]=
2	=IF(RAND()<0.5,0,1)	=IF(K36<0,0,1)	=IF(RAND()<0.5,0,1)	=IF(L36<0,0,1)	=IF(RAND()<0.5,0,1)	=IF(M36<0,0,1)				640		1002	494	=IF(L2>0,M2/L2,0)
3														
4	i	sA_i	sB_i	sC_i	N_i	sAT_i	sBT_i	sCTi		X_i	$sA_i X_i$	$sB_i X_i$	$sC_i X_i$	
5	0	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B5,-B5)	=IF($C$2=1,C5,-C5)	=IF($E$2=1,D5,-D5)		=E5+F5+G5+H5	=B5*J5	=C5*J5	=J5*D5	
6	=1+A5	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B6,-B6)	=IF($C$2=1,C6,-C6)	=IF($E$2=1,D6,-D6)		=E6+F6+G6+H6	=B6*J6	=C6*J6	=J6*D6	
7	=1+A6	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B7,-B7)	=IF($C$2=1,C7,-C7)	=IF($E$2=1,D7,-D7)		=E7+F7+G7+H7	=B7*J7	=C7*J7	=J7*D7	
8	=1+A7	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B8,-B8)	=IF($C$2=1,C8,-C8)	=IF($E$2=1,D8,-D8)		=E8+F8+G8+H8	=B8*J8	=C8*J8	=J8*D8	
9	=1+A8	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B9,-B9)	=IF($C$2=1,C9,-C9)	=IF($E$2=1,D9,-D9)		=E9+F9+G9+H9	=B9*J9	=C9*J9	=J9*D9	
10	=1+A9	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B10,-B10)	=IF($C$2=1,C10,-C10)	=IF($E$2=1,D10,-D10)		=E10+F10+G10+H10	=B10*J10	=C10*J10	=J10*D10	
11	=1+A10	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B11,-B11)	=IF($C$2=1,C11,-C11)	=IF($E$2=1,D11,-D11)		=E11+F11+G11+H11	=B11*J11	=C11*J11	=J11*D11	
12	=1+A11	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B12,-B12)	=IF($C$2=1,C12,-C12)	=IF($E$2=1,D12,-D12)		=E12+F12+G12+H12	=B12*J12	=C12*J12	=J12*D12	
13	=1+A12	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B13,-B13)	=IF($C$2=1,C13,-C13)	=IF($E$2=1,D13,-D13)		=E13+F13+G13+H13	=B13*J13	=C13*J13	=J13*D13	
14	=1+A13	10	0	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B14,-B14)	=IF($C$2=1,C14,-C14)	=IF($E$2=1,D14,-D14)		=E14+F14+G14+H14	=B14*J14	=C14*J14	=J14*D14	
15	=1+A14	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B15,-B15)	=IF($C$2=1,C15,-C15)	=IF($E$2=1,D15,-D15)		=E15+F15+G15+H15	=B15*J15	=C15*J15	=J15*D15	
16	=1+A15	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B16,-B16)	=IF($C$2=1,C16,-C16)	=IF($E$2=1,D16,-D16)		=E16+F16+G16+H16	=B16*J16	=C16*J16	=J16*D16	
17	=1+A16	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B17,-B17)	=IF($C$2=1,C17,-C17)	=IF($E$2=1,D17,-D17)		=E17+F17+G17+H17	=B17*J17	=C17*J17	=J17*D17	
18	=1+A17	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B18,-B18)	=IF($C$2=1,C18,-C18)	=IF($E$2=1,D18,-D18)		=E18+F18+G18+H18	=B18*J18	=C18*J18	=J18*D18	
19	=1+A18	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B19,-B19)	=IF($C$2=1,C19,-C19)	=IF($E$2=1,D19,-D19)		=E19+F19+G19+H19	=B19*J19	=C19*J19	=J19*D19	
20	=1+A19	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B20,-B20)	=IF($C$2=1,C20,-C20)	=IF($E$2=1,D20,-D20)		=E20+F20+G20+H20	=B20*J20	=C20*J20	=J20*D20	
21	=1+A20	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B21,-B21)	=IF($C$2=1,C21,-C21)	=IF($E$2=1,D21,-D21)		=E21+F21+G21+H21	=B21*J21	=C21*J21	=J21*D21	
22	=1+A21	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B22,-B22)	=IF($C$2=1,C22,-C22)	=IF($E$2=1,D22,-D22)		=E22+F22+G22+H22	=B22*J22	=C22*J22	=J22*D22	
23	=1+A22	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B23,-B23)	=IF($C$2=1,C23,-C23)	=IF($E$2=1,D23,-D23)		=E23+F23+G23+H23	=B23*J23	=C23*J23	=J23*D23	
24	=1+A23	0	-10	0	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B24,-B24)	=IF($C$2=1,C24,-C24)	=IF($E$2=1,D24,-D24)		=E24+F24+G24+H24	=B24*J24	=C24*J24	=J24*D24	
25	=1+A24	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B25,-B25)	=IF($C$2=1,C25,-C25)	=IF($E$2=1,D25,-D25)		=E25+F25+G25+H25	=B25*J25	=C25*J25	=J25*D25	
26	=1+A25	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B26,-B26)	=IF($C$2=1,C26,-C26)	=IF($E$2=1,D26,-D26)		=E26+F26+G26+H26	=B26*J26	=C26*J26	=J26*D26	
27	=1+A26	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B27,-B27)	=IF($C$2=1,C27,-C27)	=IF($E$2=1,D27,-D27)		=E27+F27+G27+H27	=B27*J27	=C27*J27	=J27*D27	
28	=1+A27	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B28,-B28)	=IF($C$2=1,C28,-C28)	=IF($E$2=1,D28,-D28)		=E28+F28+G28+H28	=B28*J28	=C28*J28	=J28*D28	
29	=1+A28	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B29,-B29)	=IF($C$2=1,C29,-C29)	=IF($E$2=1,D29,-D29)		=E29+F29+G29+H29	=B29*J29	=C29*J29	=J29*D29	
30	=1+A29	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B30,-B30)	=IF($C$2=1,C30,-C30)	=IF($E$2=1,D30,-D30)		=E30+F30+G30+H30	=B30*J30	=C30*J30	=J30*D30	
31	=1+A30	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B31,-B31)	=IF($C$2=1,C31,-C31)	=IF($E$2=1,D31,-D31)		=E31+F31+G31+H31	=B31*J31	=C31*J31	=J31*D31	
32	=1+A31	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B32,-B32)	=IF($C$2=1,C32,-C32)	=IF($E$2=1,D32,-D32)		=E32+F32+G32+H32	=B32*J32	=C32*J32	=J32*D32	
33	=1+A32	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B33,-B33)	=IF($C$2=1,C33,-C33)	=IF($E$2=1,D33,-D33)		=E33+F33+G33+H33	=B33*J33	=C33*J33	=J33*D33	
34	=1+A33	0	0	10	=$J$2*NORM.S.INV(RAND())	=IF($A$2=1,B34,-B34)	=IF($C$2=1,C34,-C34)	=IF($E$2=1,D34,-D34)		=E34+F34+G34+H34	=B34*J34	=C34*J34	=J34*D34	
35														
36										V=	=SUM(K5:K34)	=SUM(L5:L34)	=SUM(M5:M34)	

The VBA code is the following. The same VBA code is used in Projects 8.2 and 8.3.

```
Sub start()
Range("L2").Value = 0
Range("M2").Value = 0
End Sub
Sub xmit()
Dim DA_t As Integer
Dim DA_r As Integer
Dim DB_t As Integer
Dim DB_r As Integer
Dim DC_t As Integer
Dim DC_r As Integer
Range("L2") = Range("L2") + 3   ' transmit 3 bits. cell change causes recalc
' Store values so that they are not recomputed
' Recall any change produces recalculation
DA_t = Range("A2").Value
DA_r = Range("B2").Value
DB_t = Range("C2").Value
DB_r = Range("D2").Value
DC_t = Range("E2").Value
DC_r = Range("F2").Value
If DA_t <> DA_r Then             ' error in DA
    Range("M2") = Range("M2") + 1
End If
If DB_t <> DB_r Then             ' error in DB
    Range("M2") = Range("M2") + 1
End If
If DC_t <> DC_r Then             ' error in DC
    Range("M2") = Range("M2") + 1
End If
End Sub
'---
Sub auto()
    Dim RowNum As Integer
        RowNum = 5               ' starting row of data compilation
        Range("J2") = 5          ' initial noise SD value
        Do While Range("J2") <= 1000    ' end noise SD value
            start
            Do While Range("L2") < 1000 ' 1000 transmissions
                xmit
            Loop
            Range("O" & RowNum) = Range("J2").Value     ' enter noise SD
            Range("P" & RowNum) = Range("N2").Value     ' enter Prob[err]
            RowNum = RowNum + 1  ' increment row
            Range("J2") = Sqr(2) * Range("J2")          ' increase noise SD
        Loop
End Sub
```

)

Project 8.2 (Extending FDMA to 3 simultaneous users) *Extend Example 13.44 to generate FDMA signals with $n_X = 32$ to serve three users simultaneously. How many unique FDMA are possible with with $n_X = 32$?*

(ans: The frequency values were assigned to users as $f_a = 1/32, f_B = 2/32, f_C = 3/32$. With $n_x = 32$, there are $(n_x/2) - 2$ frequencies are available. ($f = 0$ and $f = 1/2$ will not work because they produce all-zero values, which violates the constant $\mathcal{E}_s$ rule.

The VBA Macro is the same as in Project 8.1.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	DA_t	DA_r	DB_t	DB_r	DC_t	DC_r	f_B=1/16,			σ_N=		# xmits=	#errors=	P[error]=				
2	0	1	1	1	0	0	f_C=3/32,			1280		1002	475	0.4741				START
3																		
4	i	sA_i	sB_i	sC_i	N_i	sAT_i	sBT_i	sCTi		X_i	$sA_i\ X_i$	$sB_i\ X_i$	$sC_i\ X_i$		σ_N	P[error]		
5	0	0.00	0.00	0.00	125.01	0	0	0		125.01	0.00	0.00	0.00		5.0	0.000		XMIT
6	1	1.95	3.83	5.56	1950.10	-2	4	-6		1946.42	3797.27	7448.62	10813.72		7.1	0.000		
7	2	3.83	7.07	9.24	737.29	-4	7	-9		731.29	2798.53	5171.02	6756.26		10.0	0.000		
8	3	5.56	9.24	9.81	292.59	-6	9	-10		286.46	1591.51	2646.59	2809.61		14.1	0.001		AUTO
9	4	7.07	10.00	7.07	-78.95	-7	10	-7		-83.10	-587.58	-830.97	-587.58		20.0	0.024		
10	5	8.31	9.24	1.95	88.93	-8	9	-2		87.90	730.86	812.09	171.48		28.3	0.089		
11	6	9.24	7.07	-3.83	-820.90	-9	7	4		-819.25	-7568.84	-5792.94	3135.12		40.0	0.149		
12	7	9.81	3.83	-8.31	3586.27	-10	4	8		3588.61	35196.51	13733.00	-29838.16		56.6	0.239		
13	8	10.00	0.00	-10.00	546.95	-10	0	10		546.95	5469.48	0.00	-5469.48		80.0	0.314		
14	9	9.81	-3.83	-8.31	438.01	-10	-4	8		432.69	4243.80	-1655.85	-3597.72		113.1	0.356		
15	10	9.24	-7.07	-3.83	1685.16	-9	-7	4		1672.68	15453.52	-11827.61	-6401.06		160.0	0.416		
16	11	8.31	-9.24	1.95	728.86	-8	-9	-2		709.35	5898.04	-6553.55	1383.88		226.3	0.429		
17	12	7.07	-10.00	7.07	-816.42	-7	-10	-7		-840.57	-5943.70	8405.66	-5943.70		320.0	0.445		
18	13	5.56	-9.24	9.81	-1060.22	-6	-9	-10		-1084.82	-6026.94	10022.43	-10639.76		452.5	0.463		
19	14	3.83	-7.07	9.24	-611.28	-4	-7	-9		-631.42	-2416.34	4464.81	-5833.55		640.000	0.460		
20	15	1.95	-3.83	5.56	-3013.88	-2	-4	-6		-3025.22	-5901.90	11577.00	-16807.20		905.097	0.474		
21	16	0.00	0.00	0.00	-3208.24	0	0	0		-3208.24	0.00	0.00	0.00					
22	17	-1.95	3.83	-5.56	-469.47	2	4	6		-458.14	893.78	-1753.22	2545.28		$E_s = A^2 n_x/2$ = 1600. When σ_N=40, P[error] = 0.16 (as expected). When $\sigma_N >> E_s^2$, P[error] →0.5. When n_x=32, there are 14 harmonic frequencies (f=0, and f=1/2 will not work).			
23	18	-3.83	7.07	-9.24	113.81	4	7	9		133.95	-512.60	947.17	-1237.53					
24	19	-5.56	9.24	-9.81	1591.11	6	9	10		1615.71	-8976.40	14927.21	-15846.64					
25	20	-7.07	10.00	-7.07	1703.15	7	10	7		1727.30	-12213.82	17272.95	-12213.82					
26	21	-8.31	9.24	-1.95	-166.94	8	9	2		-147.43	1225.86	-1362.10	287.63					
27	22	-9.24	7.07	3.83	742.15	9	7	-4		754.63	-6971.91	5336.07	2887.86					
28	23	-9.81	3.83	8.31	639.16	10	4	-8		644.48	-6320.94	2466.31	5358.64					
29	24	-10.00	0.00	10.00	-981.36	10	0	-10		-981.36	9813.65	0.00	-9813.65					
30	25	-9.81	-3.83	8.31	2094.71	10	-4	-8		2092.38	-20521.71	-8007.18	17397.47					
31	26	-9.24	-7.07	3.83	-638.98	9	-7	-4		-640.64	5918.75	4530.02	-2451.63					
32	27	-8.31	-9.24	-1.95	-257.85	8	-9	2		-256.82	2135.39	2372.71	501.03					
33	28	-7.07	-10.00	-7.07	-174.38	7	-10	7		-170.24	1203.78	1702.40	1203.78					
34	29	-5.56	-9.24	-9.81	-1221.02	6	-9	10		-1214.90	6749.60	11224.18	11915.52					
35	30	-3.83	-7.07	-9.24	-1474.56	4	-7	9		-1468.56	5619.95	10384.32	13567.76					
36	31	-1.95	-3.83	-5.56	158.62	2	-4	6		162.30	-316.62	-621.08	-901.66					
37																		
38										V=	24460.98	97040.05	-46848.11					

)

Project 8.3 (Extending CDMA to 3 simultaneous users) *Extend Example 13.44 to generate CDMA signals with $n_X = 32$ to serve three users simultaneously. (Hint: you need to compute 3 cross-product sums $(1-2, 1-3, 2-3)$).*

(ans: First, we search for the CDMA codes for three users using the following Excel worksheet. While the codes can be found manually by repeatedly pressing `F9`*, because the number of guesses can vary between 10 and 800, the VBA code makes the search painless. The search stops when all three orthogonality measures equal zero. The three codes in* `E6:G37` *are then copied with one selection and paste/special/values into* `J6:L37`*. This past will cause new (possibly non-orthogonal) codes to appear in* `E6:G37`*, but the paste command has inserted the correct codes.*

	A	B	C	D	E	F	G	H	I	J	K	L
1	A_{max}=	n_x=			Guess#	Find CDMA						
2	10	32			59							
3												
4	CDMA									Final Codes		
5	sA_i	sB_i	sC_i		sA_isB_i	sA_isC_i	sB_isC_i		i	sA_i	sB_i	sC_i
6	-10	10	-10		-100	100	-100		0	-10	-10	-10
7	10	-10	-10		-100	-100	100		1	10	-10	-10
8	-10	-10	10		100	-100	-100		2	-10	10	-10
9	-10	10	10		-100	-100	100		3	10	-10	10
10	10	10	10		100	100	100		4	10	-10	10
11	-10	10	10		-100	-100	100		5	-10	10	10
12	10	-10	-10		-100	-100	100		6	10	10	10
13	-10	10	10		-100	-100	100		7	-10	-10	10
14	-10	-10	-10		100	100	100		8	10	-10	-10
15	10	10	10		100	100	100		9	-10	-10	-10
16	-10	-10	-10		100	100	100		10	-10	-10	-10
17	10	10	-10		100	-100	-100		11	10	10	10
18	-10	-10	10		100	-100	-100		12	10	10	-10
19	-10	-10	-10		100	100	100		13	-10	10	10
20	-10	-10	10		100	-100	-100		14	-10	-10	10
21	10	10	10		100	100	100		15	-10	-10	10
22	10	-10	10		-100	100	-100		16	10	-10	-10
23	-10	-10	-10		100	100	100		17	10	-10	10
24	-10	-10	10		100	-100	-100		18	10	-10	10
25	10	10	-10		100	-100	-100		19	10	-10	-10
26	-10	10	-10		-100	100	-100		20	10	10	10
27	10	-10	10		-100	100	-100		21	-10	10	10
28	-10	10	10		-100	-100	100		22	10	10	-10
29	-10	10	10		-100	-100	100		23	10	-10	10
30	10	10	10		100	100	100		24	10	10	-10
31	10	-10	10		-100	100	-100		25	-10	-10	-10
32	-10	10	-10		-100	100	-100		26	-10	10	-10
33	10	10	-10		100	-100	-100		27	10	-10	10
34	-10	10	-10		-100	100	-100		28	-10	-10	10
35	10	-10	-10		-100	-100	100		29	10	-10	-10
36	10	-10	10		-100	100	-100		30	10	10	10
37	-10	-10	10		100	-100	-100		31	-10	-10	10
38												
39				Sum=	0	0	0					

```
Sub find()
Range("E2").Value = 0 ' reset counter
Do While Range("E39").Value > 1 Or Range("F39").Value > 1 Or Range("G39").Value > 1
    Range("E2").Value = Range("E2").Value + 1
Loop
End Sub
```

The three CDMA codes are copied from `E6:G37` *into the CDMA worksheet (a copy of the same one* *that was used for TDMA and FDMA - along with the VBA Macros). The VBA code is given in Project 8.1.*

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	DA_t	DA_r	DB_t	DB_r	DC_t	DC_r				σ_N=		# xmits=	#errors=	P[error]=				
2	0	0	0	1	1	1				1280		1002	481	0.4800				START
3																		
4	i	sA_i	sB_i	sC_i	N_i	sAT_i	sBT_i	sCTi		X_i	$sA_i\ X_i$	$sB_i\ X_i$	$sC_i\ X_i$		σ_N	P[error]		
5	0	-10.00	-10.00	-10.00	-237.69	10	10	-10		-227.69	2276.92	2276.92	2276.92		5.0	0.000		XMIT
6	1	10.00	-10.00	-10.00	-609.77	-10	10	-10		-619.77	-6197.75	6197.75	6197.75		7.1	0.000		
7	2	-10.00	10.00	-10.00	2017.78	10	-10	-10		2007.78	-20077.82	20077.82	-20077.82		10.0	0.000		
8	3	10.00	-10.00	10.00	1378.23	-10	10	10		1388.23	13882.28	-13882.28	13882.28		14.1	0.000		AUTO
9	4	10.00	-10.00	10.00	-2026.24	-10	10	10		-2016.24	-20162.39	20162.39	-20162.39		20.0	0.003		
10	5	-10.00	10.00	10.00	1823.00	10	-10	10		1833.00	-18330.01	18330.01	18330.01		28.3	0.026		
11	6	10.00	10.00	10.00	-359.85	-10	-10	10		-369.85	-3698.48	-3698.48	-3698.48		40.0	0.088		
12	7	-10.00	-10.00	10.00	527.20	10	10	10		557.20	-5572.03	-5572.03	5572.03		56.6	0.166		
13	8	10.00	-10.00	-10.00	71.07	-10	10	-10		61.07	610.68	-610.68	-610.68		80.0	0.266		
14	9	-10.00	-10.00	-10.00	317.50	10	10	-10		327.50	-3275.02	-3275.02	-3275.02		113.1	0.318		
15	10	-10.00	-10.00	-10.00	-797.73	10	10	-10		-787.73	7877.34	7877.34	7877.34		160.0	0.373		
16	11	10.00	10.00	10.00	-1840.63	-10	-10	10		-1850.63	-18506.27	-18506.27	-18506.27		226.3	0.380		
17	12	10.00	10.00	-10.00	-2598.65	-10	-10	-10		-2628.65	-26286.47	-26286.47	26286.47		320.0	0.434		
18	13	-10.00	10.00	10.00	648.37	10	-10	10		658.37	-6583.68	6583.68	6583.68		452.5	0.456		
19	14	-10.00	-10.00	10.00	-494.69	10	10	10		-464.69	4646.94	4646.94	-4646.94		640.000	0.478		
20	15	-10.00	-10.00	10.00	-2504.22	10	10	10		-2474.22	24742.18	24742.18	-24742.18		905.097	0.480		
21	16	10.00	-10.00	-10.00	-565.64	-10	10	-10		-575.64	-5756.43	5756.43	5756.43					
22	17	10.00	-10.00	10.00	69.52	-10	10	10		79.52	795.16	-795.16	795.16		$E_s = A^2 n_x = 3200$. When σ_N=57, P[error] = 0.16 (as expected). As $\sigma_N >> E_s$, P[Error] → 0.5			
23	18	10.00	-10.00	10.00	-288.16	-10	10	10		-278.16	-2781.60	2781.60	-2781.60					
24	19	10.00	-10.00	-10.00	2901.87	-10	10	-10		2891.87	28918.68	-28918.68	-28918.68					
25	20	10.00	10.00	10.00	1614.45	-10	-10	10		1604.45	16044.55	16044.55	16044.55					
26	21	-10.00	10.00	10.00	1718.14	10	-10	10		1728.14	-17281.39	17281.39	17281.39					
27	22	10.00	10.00	-10.00	1081.32	-10	-10	-10		1051.32	10513.24	10513.24	-10513.24					
28	23	10.00	-10.00	10.00	1096.26	-10	10	10		1106.26	11062.55	-11062.55	11062.55					
29	24	10.00	10.00	-10.00	763.47	-10	-10	-10		733.47	7334.65	7334.65	-7334.65					
30	25	-10.00	-10.00	-10.00	-320.13	10	10	-10		-310.13	3101.27	3101.27	3101.27					
31	26	-10.00	10.00	-10.00	-1076.32	10	-10	-10		-1086.32	10863.19	-10863.19	10863.19					
32	27	10.00	-10.00	10.00	1108.79	-10	10	10		1118.79	11187.89	-11187.89	11187.89					
33	28	-10.00	-10.00	10.00	-186.05	10	10	10		-156.05	1560.53	1560.53	-1560.53					
34	29	10.00	-10.00	-10.00	208.81	-10	10	-10		198.81	1988.14	-1988.14	-1988.14					
35	30	10.00	10.00	10.00	-1702.73	-10	-10	10		-1712.73	-17127.29	-17127.29	-17127.29					
36	31	-10.00	-10.00	10.00	981.12	10	10	10		1011.12	-10111.19	-10111.19	10111.19					
37																		
38										V=	-24341.61	11383.37	7266.20					

)

Project 8.4 (Histogram of orthogonal signals in noise) *Using Example 13.32 as a guide, generate a histogram of one thousand matched processor output values V when processing signals in the presence of noise. Choose your favorite orthogonal complementary signal pair with $n_x = 16$ and a signal-to-noise ratio $\mathcal{E}_s/\sigma_N^2 = 4$. Repeat for $\mathcal{E}_s/\sigma_N^2 = 16$.*

(ans: TDMA, FDMA, and CDMA two-user signals are considered, each with maximum allowed amplitude = 10. The histograms should look similar for all signal types because the overlap in the PDFs is determined solely by the SNR. Recall the signal energy is the same for TDMA and FDMA signals and is equal to half that provided by CDMA. Hence, for the same SNR, the σ_N^2 specified for CDMA systems will be twice the value specified for TDMA and FDMA systems.

The following worksheets show results for SNR=4 (in `E5`) and SNR=16 for TDMA signals, although they are typical for FDMA and CDMA signals. The SNR=4 histograms show significant overlap, while those for SNR=16 show clear separation because bins contain zero counts.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	n_x=	A_{max} =	Es=	σ^2_N=	σ_V=		bin	count		D_t=	1			$X_i = sT_i + N_i$		
2	16	10	800	200	400		-2400	0		i	s_i	sT_i=		c_i	X_i	$c_i X_i$
3							-2133	1		0	10.00	10.00		10.00	6.18	61.82
4	n_T=	i=	bin#=		SNR=		-1866	3		1	10.00	10.00		10.00	-11.21	-112.13
5	1000	1000	11		4		-1599	20		2	10.00	10.00		10.00	-15.60	-155.99
6							-1332	60		3	10.00	10.00		10.00	1.45	14.50
7	min bin =	# bins=	width=				-1065	112		4	10.00	10.00		10.00	16.77	167.69
8	-2400	19	267				-798	117		5	10.00	10.00		10.00	1.91	19.10
9							-531	108		6	10.00	10.00		10.00	-25.23	-252.32
10							-264	59		7	10.00	10.00		10.00	24.44	244.41
11							3	43		8	0.00	0.00		0.00	-4.99	0.00
12		Hist					270	50		9	0.00	0.00		0.00	8.90	0.00
13							537	117		10	0.00	0.00		0.00	3.98	0.00
14							804	126		11	0.00	0.00		0.00	5.35	0.00
15							1071	101		12	0.00	0.00		0.00	-15.39	0.00
16							1338	64		13	0.00	0.00		0.00	18.69	0.00
17							1605	16		14	0.00	0.00		0.00	-0.86	0.00
18							1872	2		15	0.00	0.00		0.00	9.17	0.00
19							2139	1								
20							2406	0							$V\|_{X=sT+N}$ =	-12.94

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	n_x=	A_{max} =	Es=	σ^2_N=	σ_V=		bin	count		D_t=	1			$X_i = sT_i + N_i$		
2	16	10	800	50	200		-1600	0		i	s_i	sT_i=		c_i	X_i	$c_i X_i$
3							-1422	2		0	10.00	10.00		10.00	10.90	108.99
4	n_T=	i=	bin#=		SNR=		-1244	21		1	10.00	10.00		10.00	4.98	49.75
5	1000	1000	14		16		-1066	74		2	10.00	10.00		10.00	-1.71	-17.15
6							-888	159		3	10.00	10.00		10.00	-1.17	-11.69
7	min bin =	# bins=	width=				-710	162		4	10.00	10.00		10.00	13.73	137.28
8	-1600	19	178				-532	79		5	10.00	10.00		10.00	0.45	4.52
9							-354	21		6	10.00	10.00		10.00	7.70	77.04
10							-176	0		7	10.00	10.00		10.00	16.63	166.34
11							2	0		8	0.00	0.00		0.00	-2.90	0.00
12		Hist					180	2		9	0.00	0.00		0.00	1.21	0.00
13							358	16		10	0.00	0.00		0.00	6.18	0.00
14							536	72		11	0.00	0.00		0.00	-3.53	0.00
15							714	166		12	0.00	0.00		0.00	0.02	0.00
16							892	135		13	0.00	0.00		0.00	-11.63	0.00
17							1070	71		14	0.00	0.00		0.00	-0.21	0.00
18							1248	18		15	0.00	0.00		0.00	3.29	0.00
19							1426	2								
20							1604	0							$V\|_{X=sT+N}$ =	515.09

The signal parameters are specified in the worksheet in terms of the signal duration n_x, the maximum allowable amplitude A_{max}, and the noise variance σ_N^2. The signal sequence is specified in `K3:K18`. For the signal sequence we compute the signal energy E_s, the matched processor coefficients *in* `C3:C18`*. The filter output variance*

$$\sigma_V^2 = \sigma_N^2 \sum_{i=0}^{n_x-1} c_i^2$$

is computed and its square root in computed in `E2`*.*

The histogram extends from hist min *to* hist max *(*$=-$ hist min*) and its parameters are computed in the worksheet. The* hist min *value in* `A8` *is computed from* $\mathcal{E}_s$ *and* σ_V *as* hist min $= -(\mathcal{E}_s - 4\sigma_V)$*) to effectively guarantee that it will be less than the minimum* V *value encountered. The number of bins is specified by the user in* `B8`*. The bin width is computed in* `C8` *as* $-2($ *hist min)/(# bins-1). The bin number of an observed value is the row number is the histogram display and is computed as rounded((value minus* hist min*)/bin width) +2 in* `C5` *as*

```
=ROUND((P20-A8)/C8,0)+2
```

This allows the verification of the bin number for each manual observation (produce by `F9`*).*

The VBA program is the same for each orthogonal signal type. The histogram parameters are computed in the worksheet in the same manner for each signal type.

```
Sub hist()
Dim Val As Integer
Range("B5").Value = 0               ' Reset counter i
For Val = 1 To Range("B8").Value    ' Set Hist Bin Vals
    Range("G" & 1 + Val).Value = Range("A8").Value + (Val - 1) * Range("C8").Value
    Range("H" & 1 + Val).Value = 0  ' Initialize counts
Next
Do While Range("B5").Value < Range("A5").Value   ' loop for nT times
    Val = (Range("P20").Value - Range("A8").Value) / Range("C8").Value + 2  'bin value
    If Val > 1 Then                            '  if valid bin value
        Range("H" & Val).Value = Range("H" & Val).Value + 1    ' incr bin count
    End If
    Range("B5").Value = Range("B5").Value + 1   ' increment counter i
Loop
End Sub
```

1. TDMA signal with two users.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	n_x=	A_{max} =	Es=	σ^2_N=	σ_V=		bin	count		D_t=	1			$X_i = sT_i + N_i$		
2	16	10	800	100	283		-1931	0		i	s_i	sT_i=		c_i	X_i	$c_i X_i$
3							-1716	1		0	10.00	10.00		10.00	22.29	222.92
4	n_T=	i=	bin#=		SNR=		-1501	7		1	10.00	10.00		10.00	21.69	216.91
5	1000	1000	17		8		-1286	37		2	10.00	10.00		10.00	18.39	183.86
6							-1071	96		3	10.00	10.00		10.00	20.76	207.65
7	min bin =	# bins=	width=				-856	148		4	10.00	10.00		10.00	19.80	197.99
8	-1931	19	215				-641	113		5	10.00	10.00		10.00	13.81	138.08
9							-426	60		6	10.00	10.00		10.00	4.25	42.50
10							-211	20		7	10.00	10.00		10.00	0.87	8.67
11							4	8		8	0.00	0.00		0.00	18.92	0.00
12	Hist						219	20		9	0.00	0.00		0.00	14.90	0.00
13							434	68		10	0.00	0.00		0.00	11.44	0.00
14							649	154		11	0.00	0.00		0.00	3.65	0.00
15							864	145		12	0.00	0.00		0.00	17.12	0.00
16							1079	85		13	0.00	0.00		0.00	-4.35	0.00
17							1294	30		14	0.00	0.00		0.00	-5.01	0.00
18							1509	8		15	0.00	0.00		0.00	0.49	0.00
19							1724	0								
20							1939	0							$V\|_{X=sT+N}$ =	1218.58

	A	B	C	D	E
1	n_x=	A_{max} =	Es=	σ^2_N=	σ_V=
2	16	10	=SUMSQ(K3:K18)	100	=SQRT(D2*SUMSQ(N3:N18))
3					
4	n_T=	i=	bin#=		SNR=
5	1000	1000	=ROUND((P20-A8)/C8,0)+2		=C2/D2
6					
7	min bin =	# bins=	width=		
8	= -ROUND(C2+4*E2,0)	19	=-ROUND(2*A8/(B8-1),0)		

	J	K	L	M	N	O	P
1	D_t=	=ROUND(RAND(),0)			$X_i = sT_i + N_i$		
2	i	s_i	sT_i=		c_i	X_i	$c_i X_i$
3	0	=$B$2	=IF($K$1=1,K3,-K3)		=K3	=L3+SQRT($D$2)*NORM.S.INV(RAND())	=N3*O3
4	=1+J3	=$B$2	=IF($K$1=1,K4,-K4)		=K4	=L4+SQRT($D$2)*NORM.S.INV(RAND())	=N4*O4
5	=1+J4	=$B$2	=IF($K$1=1,K5,-K5)		=K5	=L5+SQRT($D$2)*NORM.S.INV(RAND())	=N5*O5
6	=1+J5	=$B$2	=IF($K$1=1,K6,-K6)		=K6	=L6+SQRT($D$2)*NORM.S.INV(RAND())	=N6*O6
7	=1+J6	=$B$2	=IF($K$1=1,K7,-K7)		=K7	=L7+SQRT($D$2)*NORM.S.INV(RAND())	=N7*O7
8	=1+J7	=$B$2	=IF($K$1=1,K8,-K8)		=K8	=L8+SQRT($D$2)*NORM.S.INV(RAND())	=N8*O8
9	=1+J8	=$B$2	=IF($K$1=1,K9,-K9)		=K9	=L9+SQRT($D$2)*NORM.S.INV(RAND())	=N9*O9
10	=1+J9	=$B$2	=IF($K$1=1,K10,-K10)		=K10	=L10+SQRT($D$2)*NORM.S.INV(RAND())	=N10*O10
11	=1+J10	0	=IF($K$1=1,K11,-K11)		=K11	=L11+SQRT($D$2)*NORM.S.INV(RAND())	=N11*O11
12	=1+J11	0	=IF($K$1=1,K12,-K12)		=K12	=L12+SQRT($D$2)*NORM.S.INV(RAND())	=N12*O12
13	=1+J12	0	=IF($K$1=1,K13,-K13)		=K13	=L13+SQRT($D$2)*NORM.S.INV(RAND())	=N13*O13
14	=1+J13	0	=IF($K$1=1,K14,-K14)		=K14	=L14+SQRT($D$2)*NORM.S.INV(RAND())	=N14*O14
15	=1+J14	0	=IF($K$1=1,K15,-K15)		=K15	=L15+SQRT($D$2)*NORM.S.INV(RAND())	=N15*O15
16	=1+J15	0	=IF($K$1=1,K16,-K16)		=K16	=L16+SQRT($D$2)*NORM.S.INV(RAND())	=N16*O16
17	=1+J16	0	=IF($K$1=1,K17,-K17)		=K17	=L17+SQRT($D$2)*NORM.S.INV(RAND())	=N17*O17
18	=1+J17	0	=IF($K$1=1,K18,-K18)		=K18	=L18+SQRT($D$2)*NORM.S.INV(RAND())	=N18*O18
19							
20						$V\|_{X=sT+N}$ =	=SUM(P3:P18)

2. *FDMA signal with two users.*

	A	B	C	D	E	F	G	H	I	J	K	L
1	σ_N=	116.5		D_t=	1		#Transmits=	1000		Start		Xmit
2	Es=	1500		~D=	1		#Errors=	390				
3	SNR=	0.11					P[err]=	0.390		Restore		Auto
4												
5	i	s_i	sT_i	N_i		c_i	X_i	$c_i X_i$		σ_N	SNR	P[err]
6	0	0.00	0.00	52.03		0.00	52.03	0.00		12.2	10.0	0.002
7	1	2.08	2.08	18.39		2.08	20.47	42.55		14.6	7.1	0.000
8	2	4.07	4.07	-3.91		4.07	0.15	0.62		17.3	5.0	0.015
9	3	5.88	5.88	15.63		5.88	21.51	126.44		20.6	3.5	0.034
10	4	7.43	7.43	-223.63		7.43	-216.20	-1606.64		24.5	2.5	0.050
11	5	8.66	8.66	76.27		8.66	84.93	735.55		29.1	1.8	0.084
12	6	9.51	9.51	63.80		9.51	73.31	697.25		34.6	1.3	0.103
13	7	9.95	9.95	52.93		9.95	62.87	625.26		41.2	0.9	0.191
14	8	9.95	9.95	-149.64		9.95	-139.70	-1389.34		49.0	0.6	0.202
15	9	9.51	9.51	55.31		9.51	64.82	616.47		58.3	0.4	0.272
16	10	8.66	8.66	258.69		8.66	267.35	2315.33		69.3	0.3	0.288
17	11	7.43	7.43	14.70		7.43	22.13	164.43		82.4	0.2	0.332
18	12	5.88	5.88	-112.82		5.88	-106.94	-628.59		98.0	0.2	0.348
19	13	4.07	4.07	-24.03		4.07	-19.96	-81.18		116.5	0.1	0.390
20	14	2.08	2.08	165.37		2.08	167.45	348.15				
21	15	0.00	0.00	65.20		0.00	65.20	0.00				
22	16	-2.08	-2.08	36.17		-2.08	34.09	-70.88				
23	17	-4.07	-4.07	54.89		-4.07	50.82	-206.71				
24	18	-5.88	-5.88	53.22		-5.88	47.35	-278.30				
25	19	-7.43	-7.43	-74.09		-7.43	-81.52	605.82				
26	20	-8.66	-8.66	-7.40		-8.66	-16.06	139.07				
27	21	-9.51	-9.51	225.45		-9.51	215.94	-2053.68				
28	22	-9.95	-9.95	-128.91		-9.95	-138.85	1380.92				
29	23	-9.95	-9.95	-42.85		-9.95	-52.80	525.10				
30	24	-9.51	-9.51	48.53		-9.51	39.02	-371.13				
31	25	-8.66	-8.66	-24.74		-8.66	-33.40	289.28				
32	26	-7.43	-7.43	-237.09		-7.43	-244.52	1817.14				
33	27	-5.88	-5.88	17.18		-5.88	11.30	-66.44				
34	28	-4.07	-4.07	75.87		-4.07	71.80	-292.03				
35	29	-2.08	-2.08	-98.63		-2.08	-100.71	209.40				
36							$V\|_{X=sT+N}$=	3593.87				

	J	K	L
1	D_t=	=ROUND(RAND(),0)	
2	i	s_i	sT_i=
3	0	=$B$2*SIN(2*PI()*J3/16)	=IF($K$1=1,K3,-K3)
4	=1+J3	=$B$2*SIN(2*PI()*J4/16)	=IF($K$1=1,K4,-K4)
5	=1+J4	=$B$2*SIN(2*PI()*J5/16)	=IF($K$1=1,K5,-K5)
6	=1+J5	=$B$2*SIN(2*PI()*J6/16)	=IF($K$1=1,K6,-K6)
7	=1+J6	=$B$2*SIN(2*PI()*J7/16)	=IF($K$1=1,K7,-K7)
8	=1+J7	=$B$2*SIN(2*PI()*J8/16)	=IF($K$1=1,K8,-K8)
9	=1+J8	=$B$2*SIN(2*PI()*J9/16)	=IF($K$1=1,K9,-K9)
10	=1+J9	=$B$2*SIN(2*PI()*J10/16)	=IF($K$1=1,K10,-K10)
11	=1+J10	=$B$2*SIN(2*PI()*J11/16)	=IF($K$1=1,K11,-K11)
12	=1+J11	=$B$2*SIN(2*PI()*J12/16)	=IF($K$1=1,K12,-K12)
13	=1+J12	=$B$2*SIN(2*PI()*J13/16)	=IF($K$1=1,K13,-K13)
14	=1+J13	=$B$2*SIN(2*PI()*J14/16)	=IF($K$1=1,K14,-K14)
15	=1+J14	=$B$2*SIN(2*PI()*J15/16)	=IF($K$1=1,K15,-K15)
16	=1+J15	=$B$2*SIN(2*PI()*J16/16)	=IF($K$1=1,K16,-K16)
17	=1+J16	=$B$2*SIN(2*PI()*J17/16)	=IF($K$1=1,K17,-K17)
18	=1+J17	=$B$2*SIN(2*PI()*J18/16)	=IF($K$1=1,K18,-K18)

3. *CDMA signal with two users. The CDMA signal in column* `K` *is copied from Project 8.3 or formed using*

   `=$B$2*(2*ROUND(RAND(),0)-1)`

   in `K3:K18` *and then copied and paste/special/values into the same cells to prevent them from changing during the experiments.*

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	n_x=	A_{max} =	Es=	σ^2_N=	σ_V=		bin	count		D_t=	1			$X_i = sT_i + N_i$		
2	16	10	1600	100	400		-3200	0		i	s_i	sT_i=		c_i	X_i	$c_i X_i$
3							-2844	1		0	10.00	10.00		10.00	-4.30	-42.98
4	n_T=	i=	bin#=		SNR=		-2488	12		1	-10.00	-10.00		-10.00	14.43	-144.30
5	1000	1000	14		16		-2132	73		2	10.00	10.00		10.00	10.26	102.59
6							-1776	156		3	-10.00	-10.00		-10.00	-24.65	246.51
7	min bin =	# bins=	width=				-1420	165		4	10.00	10.00		10.00	-11.43	-114.32
8	-3200	19	356				-1064	58		5	-10.00	-10.00		-10.00	-6.50	64.99
9							-708	20		6	10.00	10.00		10.00	6.60	66.01
10							-352	2		7	-10.00	-10.00		-10.00	-9.86	98.63
11							4	0		8	-10.00	-10.00		-10.00	-24.48	244.80
12	Hist						360	2		9	-10.00	-10.00		-10.00	-21.47	214.67
13							716	15		10	-10.00	-10.00		-10.00	-4.42	44.16
14							1072	81		11	-10.00	-10.00		-10.00	-19.96	199.64
15							1428	153		12	-10.00	-10.00		-10.00	3.38	-33.76
16							1784	174		13	-10.00	-10.00		-10.00	-10.43	104.28
17							2140	72		14	10.00	10.00		10.00	0.73	7.26
18							2496	16		15	10.00	10.00		10.00	11.68	116.76
19							2852	0								
20							3208	0							$V\vert_{X=sT+N}$ =	1174.96

)

Project 8.5 (Estimating probability of orthogonal signals) *Modify Examples 13.45 to compute the probability of error as σ_N^2 increases between $0.1\mathcal{E}_s$ and $10\mathcal{E}_s$ by factor $\sqrt{2}$, using one of the orthogonal signal with $n_X = 30$ assigned to one of the two users for s_i in the worksheet.*

(ans:TDMA, FDMA, and CDMA two-user signals are considered, each with maximum allowed amplitude = 10.

The example gives σ_N (rather than σ_N^2) as the parameter. The σ_N factor is then ($2^{1/4}$) to obtain the $\sqrt{2}$ factor in the σ_N^2 specified in the problem.

The orthogonal signals have maximum amplitude of 10.

One clue as to whether students are doing the project correctly is that when $SNR \approx 1$ the Prob[err] $\approx=$ 0.16.

The following VBA program is the same for each orthogonal signal type.

```
Sub start()     ' resets counters to zero
Application.Calculation = xlCalculationManual       ' Stop Auto Calculations
Range("H1").Value = 0 ' Reset # transmits
Range("H2").Value = 0 ' Reset # errors
End Sub
'---
Sub xmit()       ' transmits another data signal
Calculate        ' Force one recalculation, New D
Range("H1").Value = Range("H1").Value + 1           ' Increment # transmits
If Range("E1").Value <> Range("E2").Value Then      ' ~D not equal D -> error
    Range("H2") = Range("H2") + 1                   ' Increment # errors
End If
End Sub
'---
Sub auto()
Dim SigmaN As Double                                ' noise SD
Dim RowNum As Integer                               ' row number in table
SigmaN = Sqr(Range("B2").Value / 10)                ' initial value of noise SD
RowNum = 6                                          ' first row of table
Do While SigmaN * SigmaN <= 10 * Range("B2").Value
    Range("B1").Value = SigmaN                      ' set noise SD in worksheet
    Range("J" & RowNum).Value = SigmaN
    start
    Do While Range("H1").Value < 1000               ' 1000 transmissions
        xmit
    Loop
    Range("L" & RowNum).Value = Range("H3").Value   ' P[err]
    Range("K" & RowNum).Value = Range("B3").Value   ' SNR
    SigmaN = SigmaN * Sqr(Sqr(2))                   ' increases var by Sqr(2)
    RowNum = RowNum + 1                             ' go to next row in table
Loop
Application.Calculation = xlCalculationAutomatic    ' restore Auto Calculations
End Sub
'---
Sub restore()
Application.Calculation = xlCalculationAutomatic    ' restore Auto Calculations
End Sub
```

1. *TDMA signal with two users.*

	A	B	C	D	E	F	G	H	I	J	K	L
1	σ_N=	116.5		D_t=	1		#Transmits=	1000		Start		Xmit
2	Es=	1500		~D=	0		#Errors=	333				
3	SNR=	0.11					P[err]=	0.333		Restore		Auto
4												
5	i	s_i	sT_i	N_i		c_i	X_i	$c_i X_i$		σ_N	SNR	P[err]
6	0	10.00	10.00	-45.37		10.00	-35.37	-353.74		12.2	10.0	0.000
7	1	10.00	10.00	26.09		10.00	36.09	360.93		14.6	7.1	0.001
8	2	10.00	10.00	44.92		10.00	54.92	549.24		17.3	5.0	0.013
9	3	10.00	10.00	-41.61		10.00	-31.61	-316.07		20.6	3.5	0.026
10	4	10.00	10.00	-113.80		10.00	-103.80	-1038.01		24.5	2.5	0.057
11	5	10.00	10.00	8.54		10.00	18.54	185.45		29.1	1.8	0.089
12	6	10.00	10.00	212.77		10.00	222.77	2227.68		34.6	1.3	0.140
13	7	10.00	10.00	-156.75		10.00	-146.75	-1467.54		41.2	0.9	0.172
14	8	10.00	10.00	37.05		10.00	47.05	470.52		49.0	0.6	0.201
15	9	10.00	10.00	10.58		10.00	20.58	205.81		58.3	0.4	0.255
16	10	10.00	10.00	43.74		10.00	53.74	537.40		69.3	0.3	0.299
17	11	10.00	10.00	-102.66		10.00	-92.66	-926.64		82.4	0.2	0.334
18	12	10.00	10.00	-121.81		10.00	-111.81	-1118.09		98.0	0.2	0.323
19	13	10.00	10.00	-37.81		10.00	-27.81	-278.15		116.5	0.1	0.333
20	14	10.00	10.00	16.53		10.00	26.53	265.34				
21	15	0.00	0.00	-99.06		0.00	-99.06	0.00				
22	16	0.00	0.00	-65.36		0.00	-65.36	0.00				
23	17	0.00	0.00	-184.06		0.00	-184.06	0.00				
24	18	0.00	0.00	-26.01		0.00	-26.01	0.00				
25	19	0.00	0.00	-202.69		0.00	-202.69	0.00				
26	20	0.00	0.00	-62.14		0.00	-62.14	0.00				
27	21	0.00	0.00	-179.02		0.00	-179.02	0.00				
28	22	0.00	0.00	-36.68		0.00	-36.68	0.00				
29	23	0.00	0.00	-128.10		0.00	-128.10	0.00				
30	24	0.00	0.00	-97.64		0.00	-97.64	0.00				
31	25	0.00	0.00	153.31		0.00	153.31	0.00				
32	26	0.00	0.00	65.46		0.00	65.46	0.00				
33	27	0.00	0.00	0.29		0.00	0.29	0.00				
34	28	0.00	0.00	-10.98		0.00	-10.98	0.00				
35	29	0.00	0.00	153.92		0.00	153.92	0.00				
36							$V\vert_{X=sT+N}$=	-695.85				

	A	B	C	D	E
1	σ_N=	175.271218401653		D_t=	=IF(RAND()<0.5,0,1)
2	Es=	=SUMSQ(B6:B35)		~D=	=IF(H36<0,0,1)
3	SNR=	=B2/B1^2			
4					
5	i	s_i	sT_i	N_i	
6	0	10	=IF($E$1=1,B6,-B6)	= $B$1*NORM.S.INV(RAND())	
7	=1+A6	10	=IF($E$1=1,B7,-B7)	= $B$1*NORM.S.INV(RAND())	
8	=1+A7	10	=IF($E$1=1,B8,-B8)	= $B$1*NORM.S.INV(RAND())	
9	=1+A8	10	=IF($E$1=1,B9,-B9)	= $B$1*NORM.S.INV(RAND())	
10	=1+A9	10	=IF($E$1=1,B10,-B10)	= $B$1*NORM.S.INV(RAND())	
11	=1+A10	10	=IF($E$1=1,B11,-B11)	= $B$1*NORM.S.INV(RAND())	
12	=1+A11	10	=IF($E$1=1,B12,-B12)	= $B$1*NORM.S.INV(RAND())	

	F	G	H
1		#Transmits=	1000
2		#Errors=	439
3		P[err]=	=IF(H1>0,H2/H1,0)
4			
5	c_i	X_i	c_i X_i
6	=B6	=C6+D6	=F6*G6
7	=B7	=C7+D7	=F7*G7
8	=B8	=C8+D8	=F8*G8
9	=B9	=C9+D9	=F9*G9
10	=B10	=C10+D10	=F10*G10
11	=B11	=C11+D11	=F11*G11
12	=B12	=C12+D12	=F12*G12
13	=B13	=C13+D13	=F13*G13
14	=B14	=C14+D14	=F14*G14
15	=B15	=C15+D15	=F15*G15
16	=B16	=C16+D16	=F16*G16
17	=B17	=C17+D17	=F17*G17
18	=B18	=C18+D18	=F18*G18
19	=B19	=C19+D19	=F19*G19
20	=B20	=C20+D20	=F20*G20
21	=B21	=C21+D21	=F21*G21
22	=B22	=C22+D22	=F22*G22
23	=B23	=C23+D23	=F23*G23
24	=B24	=C24+D24	=F24*G24
25	=B25	=C25+D25	=F25*G25
26	=B26	=C26+D26	=F26*G26
27	=B27	=C27+D27	=F27*G27
28	=B28	=C28+D28	=F28*G28
29	=B29	=C29+D29	=F29*G29
30	=B30	=C30+D30	=F30*G30
31	=B31	=C31+D31	=F31*G31
32	=B32	=C32+D32	=F32*G32
33	=B33	=C33+D33	=F33*G33
34	=B34	=C34+D34	=F34*G34
35	=B35	=C35+D35	=F35*G35
36		$V\|_{X=sT+N}$=	=SUM(H6:H35)

This last code is the same for all signal types. The sum in `H36` *must be over the entire* n_x *values (This is a common mistake).*

2. *FDMA signal with two users.*

	A	B	C	D	E	F	G	H	I	J	K	L
1	σ_N=	116.5		D_t=	0		#Transmits=	1000		Start		Xmit
2	Es=	1500		~D=	1		#Errors=	390				
3	SNR=	0.11					P[err]=	0.390		Restore		Auto
4												
5	i	s_i	sT_i	N_i		c_i	X_i	$c_i\ X_i$		σ_N	SNR	P[err]
6	0	0.00	0.00	88.16		0.00	88.16	0.00		12.2	10.0	0.002
7	1	2.08	-2.08	-6.41		2.08	-8.49	-17.65		14.6	7.1	0.000
8	2	4.07	-4.07	-122.18		4.07	-126.25	-513.51		17.3	5.0	0.015
9	3	5.88	-5.88	-19.60		5.88	-25.48	-149.74		20.6	3.5	0.034
10	4	7.43	-7.43	23.62		7.43	16.19	120.34		24.5	2.5	0.050
11	5	8.66	-8.66	4.57		8.66	-4.09	-35.39		29.1	1.8	0.084
12	6	9.51	-9.51	260.51		9.51	251.00	2387.19		34.6	1.3	0.103
13	7	9.95	-9.95	134.27		9.95	124.32	1236.43		41.2	0.9	0.191
14	8	9.95	-9.95	-341.65		9.95	-351.59	-3496.66		49.0	0.6	0.202
15	9	9.51	-9.51	25.56		9.51	16.05	152.67		58.3	0.4	0.272
16	10	8.66	-8.66	258.76		8.66	250.10	2165.93		69.3	0.3	0.288
17	11	7.43	-7.43	-85.68		7.43	-93.11	-691.92		82.4	0.2	0.332
18	12	5.88	-5.88	-65.37		5.88	-71.25	-418.77		98.0	0.2	0.348
19	13	4.07	-4.07	122.61		4.07	118.54	482.15		116.5	0.1	0.390
20	14	2.08	-2.08	-47.99		2.08	-50.07	-104.10				
21	15	0.00	0.00	-56.43		0.00	-56.43	0.00				
22	16	-2.08	2.08	37.30		-2.08	39.38	-81.87				
23	17	-4.07	4.07	130.03		-4.07	134.10	-545.43				
24	18	-5.88	5.88	-121.24		-5.88	-115.36	678.06				
25	19	-7.43	7.43	105.69		-7.43	113.12	-840.63				
26	20	-8.66	8.66	46.59		-8.66	55.25	-478.45				
27	21	-9.51	9.51	-24.70		-9.51	-15.19	144.47				
28	22	-9.95	9.95	-207.42		-9.95	-197.48	1963.97				
29	23	-9.95	9.95	-40.15		-9.95	-30.21	300.42				
30	24	-9.51	9.51	98.32		-9.51	107.83	-1025.55				
31	25	-8.66	8.66	-132.53		-8.66	-123.87	1072.77				
32	26	-7.43	7.43	80.17		-7.43	87.60	-651.01				
33	27	-5.88	5.88	-103.86		-5.88	-97.99	575.95				
34	28	-4.07	4.07	48.75		-4.07	52.82	-214.83				
35	29	-2.08	2.08	102.99		-2.08	105.07	-218.45				
36							$V\|_{X=sT+N}$=	1796.39				

	A	B	C	D	E
1	σ_N=	175.271218401653		D_t=	=IF(RAND()<0.5,0,1)
2	Es=	=SUMSQ(B6:B35)		~D=	=IF(H36<0,0,1)
3	SNR=	=B2/B1^2			
4					
5	i	s_i	sT_i	N_i	
6	0	=10*SIN(2*PI()*A6/30)	=IF($E$1=1,B6,-B6)	= $B$1*NORM.S.INV(RAND())	
7	=1+A6	=10*SIN(2*PI()*A7/30)	=IF($E$1=1,B7,-B7)	= $B$1*NORM.S.INV(RAND())	
8	=1+A7	=10*SIN(2*PI()*A8/30)	=IF($E$1=1,B8,-B8)	= $B$1*NORM.S.INV(RAND())	
9	=1+A8	=10*SIN(2*PI()*A9/30)	=IF($E$1=1,B9,-B9)	= $B$1*NORM.S.INV(RAND())	
10	=1+A9	=10*SIN(2*PI()*A10/30)	=IF($E$1=1,B10,-B10)	= $B$1*NORM.S.INV(RAND())	
11	=1+A10	=10*SIN(2*PI()*A11/30)	=IF($E$1=1,B11,-B11)	= $B$1*NORM.S.INV(RAND())	
12	=1+A11	=10*SIN(2*PI()*A12/30)	=IF($E$1=1,B12,-B12)	= $B$1*NORM.S.INV(RAND())	

3. *CDMA signal with two users. The CDMA signal in* `B6:B35` *is formed using*

`=10*(2*ROUND(RAND(),0)-1)`

in `B6:B35`. *It is then copied and paste/special/values into the same cells to prevent them from changing during the experiments.*

	A	B	C	D	E	F	G	H	I	J	K	L
1	σ_N=	247.9		D_t=	0		#Transmits=	1000		Start		Xmit
2	Es=	3000		~D=	0		#Errors=	426				
3	SNR=	0.05					P[err]=	0.426		Restore		Auto
4												
5	i	s_i	sT_i	N_i		c_i	X_i	$c_i X_i$		σ_N	SNR	P[err]
6	0	-10.00	10.00	-232.65		-10.00	-222.65	2226.55		11.0	25.00	0.000
7	1	10.00	-10.00	315.35		10.00	305.35	3053.48		13.0	17.68	0.000
8	2	-10.00	10.00	-287.30		-10.00	-277.30	2773.02		15.5	12.50	0.000
9	3	10.00	-10.00	118.26		10.00	108.26	1082.64		18.4	8.84	0.001
10	4	-10.00	10.00	358.02		-10.00	368.02	-3680.23		21.9	6.25	0.003
11	5	10.00	-10.00	-382.42		10.00	-392.42	-3924.19		26.1	4.42	0.016
12	6	10.00	-10.00	-414.19		10.00	-424.19	-4241.88		31.0	3.13	0.037
13	7	-10.00	10.00	-130.41		-10.00	-120.41	1204.15		36.8	2.21	0.082
14	8	-10.00	10.00	20.81		-10.00	30.81	-308.13		43.8	1.56	0.099
15	9	10.00	-10.00	246.03		10.00	236.03	2360.33		52.1	1.10	0.147
16	10	10.00	-10.00	-27.09		10.00	-37.09	-370.89		62.0	0.78	0.174
17	11	10.00	-10.00	53.92		10.00	43.92	439.24		73.7	0.55	0.259
18	12	-10.00	10.00	165.49		-10.00	175.49	-1754.91		87.6	0.39	0.281
19	13	-10.00	10.00	71.57		-10.00	81.57	-815.69		104.2	0.28	0.300
20	14	10.00	-10.00	32.82		10.00	22.82	228.23		123.9	0.20	0.345
21	15	10.00	-10.00	-430.15		10.00	-440.15	-4401.52		147.4	0.14	0.334
22	16	-10.00	10.00	-117.61		-10.00	-107.61	1076.12		175.3	0.10	0.361
23	17	10.00	-10.00	270.91		10.00	260.91	2609.10		208.4	0.07	0.384
24	18	10.00	-10.00	9.56		10.00	-0.44	-4.39		247.9	0.05	0.426
25	19	10.00	-10.00	94.83		10.00	84.83	848.29				
26	20	-10.00	10.00	127.84		-10.00	137.84	-1378.36				
27	21	-10.00	10.00	164.78		-10.00	174.78	-1747.75				
28	22	-10.00	10.00	213.56		-10.00	223.56	-2235.55				
29	23	-10.00	10.00	-128.41		-10.00	-118.41	1184.06				
30	24	-10.00	10.00	-256.06		-10.00	-246.06	2460.59				
31	25	-10.00	10.00	-127.51		-10.00	-117.51	1175.15				
32	26	-10.00	10.00	-4.65		-10.00	5.35	-53.47				
33	27	10.00	-10.00	136.36		10.00	126.36	1263.63				
34	28	-10.00	10.00	-302.62		-10.00	-292.62	2926.22				
35	29	-10.00	10.00	706.16		-10.00	716.16	-7161.61				
36							$V\vert_{X=sT+N}$=	-5167.78				

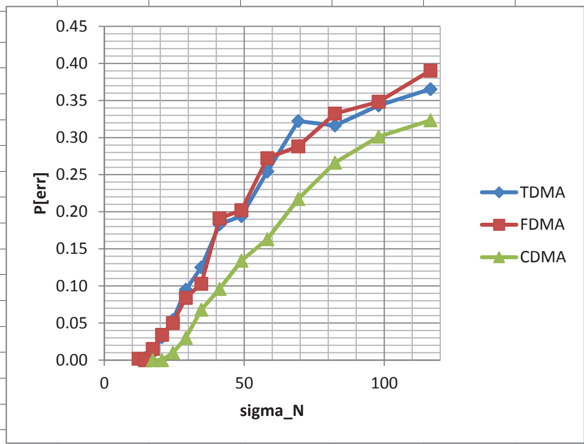
)

Project 8.6 (Comparing probability of orthogonal signals) *Modify Examples 13.45 to compare the probability of error as σ_N increases between $0.1\mathcal{E}_s$ (using the TDMA signal energy) and $10\mathcal{E}_s$ by factor $\sqrt{2}$ for the TDMA, FDMA, and CDMA signals with maximum amplitude $|s_i| \leq 10$ with $n_X = 30$ assigned to one of the users for s_i in the worksheet. Which signal type performs best, and which is the worst? Explain why.*

(ans: If this problem is done by applying Problem 8.5 to the three orthogonal signal types (it is simple matter to form the change the signal type in column `B` *- the only change necessary), it is important to note that CDMA signals have twice the $\mathcal{E}_s$ as TDMA and FDMA. Hence, their initial variance will be twice that of the TDMA nd FDMA signals. It is important that their* `Prob[err]` *must start at the correct σ_N value 9 (as done below). Then this project is accomplished by copying the* `P[err]` *values and paste/special/values into the worksheet and performing a scatter plot.*

Otherwise, a constant value for $\mathcal{E}_s$ (use either the TDMA or FDMA value) should be inserted in the VBA code for all signal types.

	A	B	C	D
1			P[err]	
2	σ_N	TDMA	FDMA	CDMA
3	12.2	0.001	0.002	
4	14.6	0.003	0.000	
5	17.3	0.010	0.015	0.000
6	20.6	0.031	0.034	0.000
7	24.5	0.054	0.050	0.010
8	29.1	0.095	0.084	0.030
9	34.6	0.125	0.103	0.068
10	41.2	0.183	0.191	0.096
11	49.0	0.194	0.202	0.134
12	58.3	0.254	0.272	0.163
13	69.3	0.322	0.288	0.217
14	82.4	0.316	0.332	0.266
15	98.0	0.343	0.348	0.301
16	116.5	0.365	0.390	0.323
17				0.382
18				0.352



For the same sigma_N value CDMA signals have a lower P[err] that TDMA and FDMA because the CDMA signal has twice the energy of TDMA and FDMA signals. The performance of TDMA and FDMA signals is comparable because they have the same E_s.

)