

“ANÁLISIS DE UN SISTEMA MC-CDMA”

Trabajo Preparatorio N°12
Laboratorio de Comunicación Digital-GR9

1st Melanny Dávila

Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
melanny.davila@epn.edu.ec

2nd Ronaldo Almachi

Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
ronaldo.almachi@epn.edu.ec

Abstract—En este documento se revisará la teoría necesaria para comprender e implementar un sistema MC-CDMA, con el objetivo de realizar un análisis correspondiente al rendimiento de MC-CDMA en un canal con ruido AWGN. Previamente en la práctica de laboratorio anterior se implementó un sistema CDMA básico, este dio resultados muy buenos referentes a la tasa de bits errados, en esta práctica se realizará los cambios necesarios para que CDMA funcione con 2 usuarios, para este propósito se utilizarán códigos Hadamard, los cuales ayudan a asignar una secuencia pseudoaleatoria ortogonal a cada usuario.

Index Terms—CDMA, Códigos Hadamard, BPSK, BER, IFFT, FFT.

I. INTRODUCCIÓN

El acceso múltiple por división de código de operador múltiple (MC-CDMA) es un concepto relativamente nuevo. Su desarrollo tuvo como objetivo mejorar el rendimiento sobre enlaces multitrayectos, combina el acceso múltiple por división de código (CDMA) convencional y la multiplexación por división de frecuencia ortogonal (OFDM) [1]. Tiene su fundamento teórico en las técnicas de espectro ensanchado (spread spectrum) [2]. Con esta técnica, la señal ocupa una anchura de banda muy superior a la que sería estrictamente necesaria para su transmisión. Tiene varias variantes que dependen de cómo se haga esta combinación. Existen tres configuraciones de la combinación de multiportadora CDMA [3]:

- CDMA de múltiples portadoras con propagación de frecuencia (MC-CDMA).
- CDMA de secuencia directa de múltiples portadoras (MC-DS-CDMA).
- CDMA multitono (MT-CDMA).

Esta técnica brinda a múltiples usuarios la oportunidad de alcanzar altas velocidades de datos en sistemas de comunicación inalámbricos, puede aprovechar la diversidad de frecuencias utilizando un receptor RAKE. Su implementación es relativamente fácil mediante el uso de la transformada rápida de Fourier y su inversa. Así mismo, la interferencia de acceso múltiple (MAI) puede ser reducida considerablemente mediante la correcta selección de códigos [3].

II. OBJETIVOS

- Profundizar los conocimientos de CDMA Multiportadora.
- Implementar en Matlab un sistema de comunicación MC-CDMA [2].

III. PREGUNTAS

A. Consultar los bloques que conforman un Transmisor y Receptor MC-CDMA. Describir la funcionalidad de cada elemento asociado.

En la figura 1, se presenta el diagrama de bloques del transmisor de un sistema MC-CDMA, este sistema es considerado spread spectrum, por lo cual tiene cierta semejanza con un sistema OFDM, las etapas que tiene ese sistema cumplen una función similar pero con un propósito un poco diferente. [4]

A continuación tenemos una breve descripción de cada etapa y bloque que conforman el transmisor MC-CDMA. [4]

- **Etapas Spread:** En esta etapa se realiza el correspondiente ensanchamiento de la señal original con un código pseudoaleatorio ortogonal, que puede ser generado por una secuencia OVVSF si se desean diferentes longitudes de ensanchamiento y en el caso de múltiples usuarios, se usa códigos Walsh-Hadamard. [4]
- **Etapas IFFT:** Como su nombre lo indica se basa en el funcionamiento de un sistema OFDM, esta sección permite tomar las diferentes señales de datos y pasarlas del dominio de la frecuencia al dominio temporal para hacer más fácil su tratamiento. [4]
- **Etapas de conversión Paralelo a Serie:** En esta sección, los diferentes flujos de datos obtenidos al usar la transformada inversa de Fourier, ya sea de un usuario o de múltiples usuarios, son convertidas de paralelo a serie, para la siguiente etapa. [4]
- **Etapas de filtrado:** Si bien este punto no siempre es presentado en todos los modelos MC-CDMA, es importante mencionarlo pues ayudará a reducir el ISI, en los sistemas spread spectrum. [4]

- **Etapla de modulación:** En este punto se modula con diferentes esquemas de modulación la señal ensanchada obtenida a después de realizar todo el proceso antes mencionado antes de ser transmitida. [4]

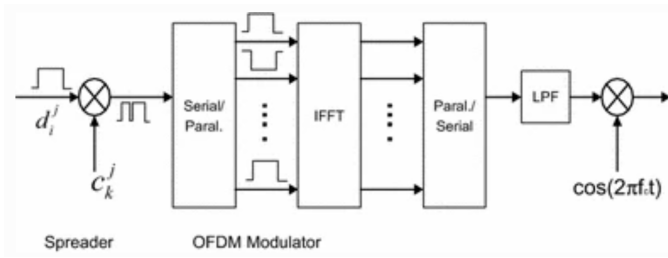


Fig. 1. Diagrama de bloques de un transmisor MC-CDMA [4]

En la figura 2, se muestra un diagrama de bloques alternativo de un transmisor MC-CDMA.

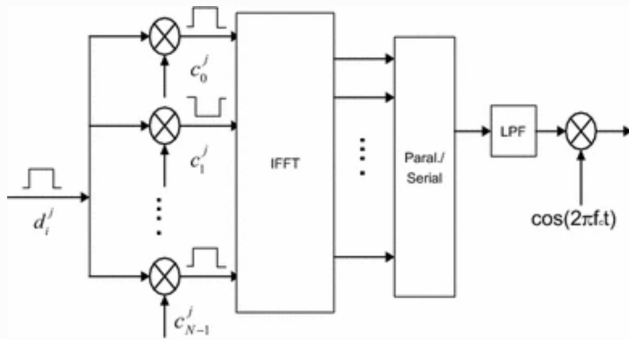


Fig. 2. Diagrama de bloques alternativo de un transmisor MC-CDMA [4]

En la figura 3, se ilustra el diagrama de bloques de un receptor correspondiente a un sistema MC-CDMA, se puede observar que el receptor cuenta con el proceso inverso al del transmisor, pero después de pasar por un canal que aporta cierta cantidad de ruido que afecta directamente a la probabilidad de bits errados. [4]

A continuación se presenta una breve descripción de cada etapa del proceso de recepción.

- **Etapla de demodulación:** En esta sección la información recibida es demodulada con el esquema acorde al que se uso en el transmisor. [4]
- **Etapla de filtrado:** En este punto la señal demodulada es pasada a través de un filtro para eliminar en gran parte el ruido que puede afectar a la recuperación de la señal.
- **Conversor Analógico-Digital:** Si bien se obtuvo la señal demodulada y filtrada, la señal resultante puede tener componentes en las que la amplitud o naturaleza de la señal no corresponden a los niveles esperados, es por que se usa esta etapa como dispositivo de decisión para aproximar el resultado obtenido. [4]
- **Conversión serial a paralelo:** En esta etapa se realiza la conversión del flujo de datos totalen serie a paralelo para la siguiente etapa. [4]

- **Transformada rápida de fourier:** En este punto se pasa del dominio a temporal al dominio de la frecuencia para finalmente recuperar los datos originales. [4]
- **Multiplicación de la señal por el código ortogonal:** Esta sección es una de las más importantes, pues es necesario que el código ortogonal corresponda al que se uso en transmisión, para recuperar la señal lo más parecida a la enviada por el transmisor. [4]
- **Suma de los flujos de datos:** Como última etapa se debe sumar cada flujo para recuperar finalmente la señal de datos que se envió a lo largo del canal, esto para un usuario, cabe recalcar que este proceso en el caso de tener múltiples usuarios es un poco más extenso. [4]

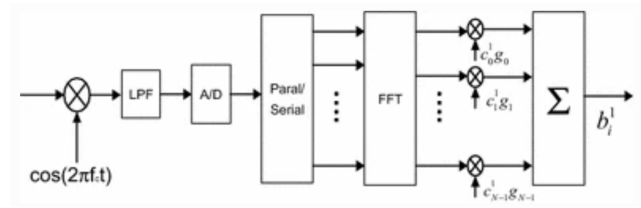


Fig. 3. Diagrama de bloques de un receptor MC-CDMA [4]

B. Consultar tres aplicaciones actuales en las que se utilice como base la técnica de acceso CDMA y describirlas brevemente.

- **2G:** La segunda generación (2G) de redes móviles introdujo dos nuevas tecnologías de acceso TDMA (acceso múltiple por división de tiempo) y CDMA (acceso múltiple por división de código) en la mezcla. Para simplificar, la tecnología de acceso es lo que conecta un teléfono móvil a la red móvil enviando y recibiendo señales a través de la interfaz aérea de forma inalámbrica [5].
- **Sistema universal de telecomunicaciones móviles (Universal Mobile Telecommunications System o UMTS):** Es una de las tecnologías usadas por los móviles de tercera generación, sucesora de GPRS, debido a que la tecnología GPRS (evolución de GSM) propiamente dicha no podía evolucionar para prestar servicios considerados de tercera generación [5]. UMTS utiliza CDMA de banda ancha (WCDMA) como estándar de transmisión de radio. Emplea un ancho de banda de canal de 5 MHz. Con este ancho de banda, tiene la capacidad de transportar más de 100 llamadas de voz simultáneas, o puede transportar datos a velocidades de hasta 2 Mbps en su formato original [6].
- **CDMA2000:** es una familia de estándares de telecomunicaciones móviles de tercera generación (3G) que utilizan CDMA, un esquema de acceso múltiple para redes digitales, para enviar voz, datos, y señalización (como un número telefónico marcado) entre teléfonos celulares y estaciones base [7]. Ésta es la segunda generación de la telefonía celular digital IS-95. CDMA (code division multiple access ó acceso múltiple por división

de código) es una estrategia de multiplexado digital que transmite flujos de bits. Básicamente, CDMA permite que múltiples terminales compartan el mismo canal de frecuencia, identificándose el "canal" de cada usuario mediante (secuencias PN) [9].

C. Consultar las características de la función Hadamard de Matlab. Generar como ejemplo una matriz de orden 8.

La función Hadamard de Matlab permite obtener una matriz de orden n si se utiliza la siguiente sintaxis: $H = \text{hadamard}(n)$, así mismo se puede definir si se desea obtener atributos tipo double o simple se define el atributo 'classname', donde se puede definir tipo: 'double' o 'single' [8].

El siguiente script permite obtener un matrix de orden 8, tal como se muestra en la figura 4.

La matriz de Hadamard de orden 8 generada es:

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
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

Fig. 4. Matriz de Hadamard generada

```

1 clear all
2 close all
3 clc
4
5 n=8; %orden de la matriz
6 b=mod(log2(n),1);%operacion modulo 1
7 %verificacion que el orden de la
  matriz sea potencia de dos
8 if b==0 %si no existe residuo ingresa
  al lazo
9     H=hadamard(n); %uso de la funcion
  hadamard de matlab
10     fprintf('La matriz de Hadamard de
  orden %d generada es: \n',n);
11     disp(H) %impresion de la matriz
12 else
13     disp('El orden debe ser de
  potencia de 2');%impresion
  indicando el error
14 end

```

Script 1. Generación de una matriz Hadamar de orden n

D. Desarrollar un script en Matlab que permita generar un par de secuencias de datos aleatorios de 100 bits para dos usuarios, los cuales se modularán en BPSK. A cada señal de usuario modulada se le asignará los códigos de Hadamard-Walsh de las filas 2 y 6 respectivamente de la matriz del punto C. Realizar la conversión Serie/Paralelo, ensanchamiento y aplicar la IFFT para cada señal de usuario. Previo a la transmisión realizar la conversión Paralelo/Serie y sumar las dos señales de usuario para obtener la señal MC-CDMA.

En la figura 5, se presenta las secuencias de datos, Hadamard y el producto de estas dos secuencias para cada uno de los usuarios presetes en este sistema. Para esto es importante mencionar que las secuencias de Hadamard seleccionadas son ortogonales entre ellas lo que genera buen rendimiento del sistema.

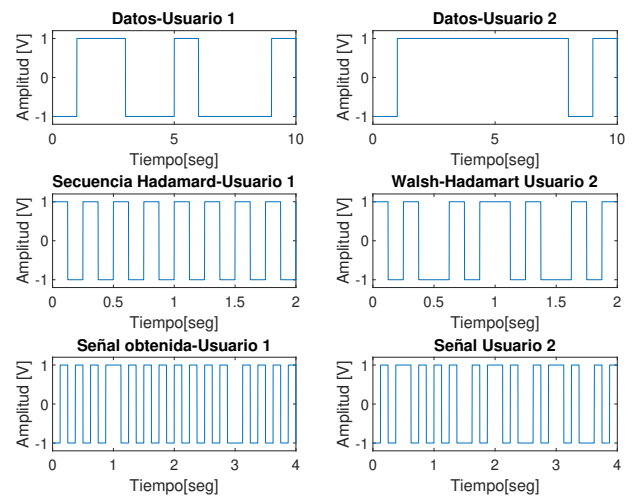


Fig. 5. Secuencias de datos en transmisión

A continuación se presenta la señal de datos total, esta es obtenida mediante la suma de señales de los diferentes usuarios, en este caso son dos señales.

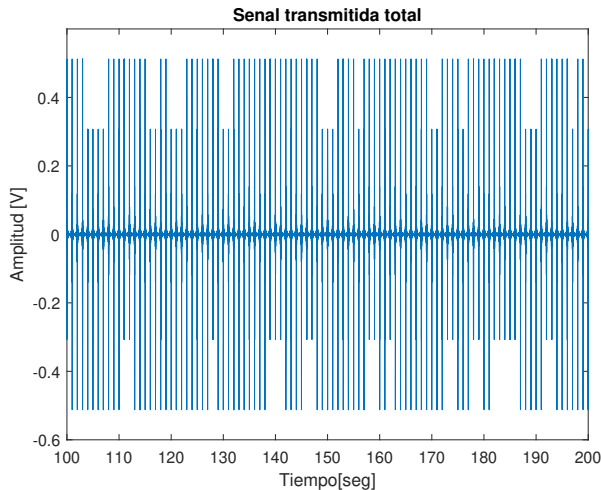


Fig. 6. Señal de datos transmitida

Finalmente, en la figura 7 se presenta una comparación entre las secuencias de datos original, es decir antes de ser transmitida y las secuencias recuperadas en recepción; donde se puede observar que la tasa de bit errado es bastante baja. Todo esto fue obtenido en base al script 2.

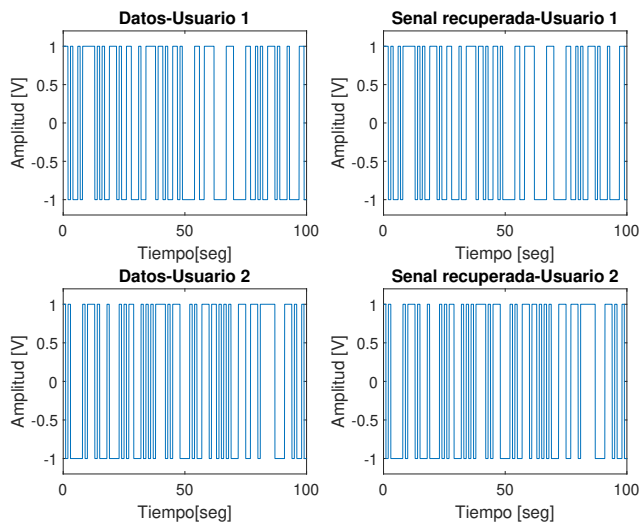


Fig. 7. Secuencias de datos original y obtenida en transmisión

E. Modificar el script del punto D para obtener el espectro en frecuencia de cada señal modulada de los usuarios y el espectro de la señal MC-CDMA.

En la figura 8, se presenta el espectro de las señales de datos en transmisión. Como se puede observar, se presenta la el espectro total, el espectro de la señal del usuario 1 y del usuario 2.

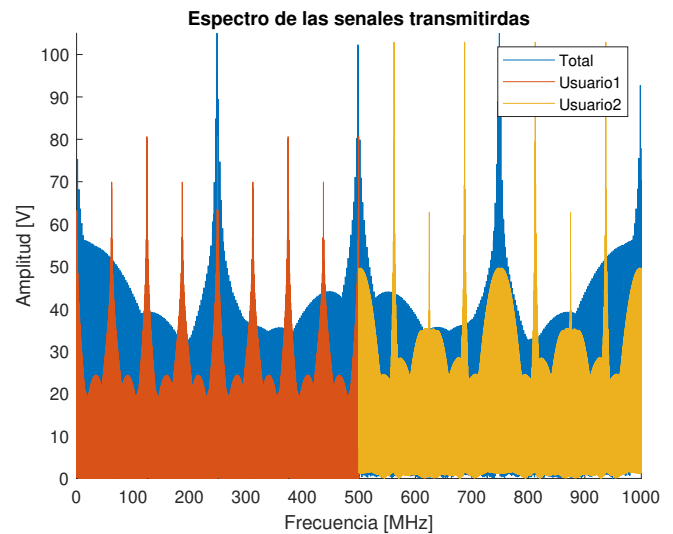


Fig. 8. Espectro de las señales de datos en transmisión

Así mismo, se obtuvo el espectro de las mismas señales de datos luego de atravesar un canal AWGN con un valor de relación señal a ruido de 20 dB.

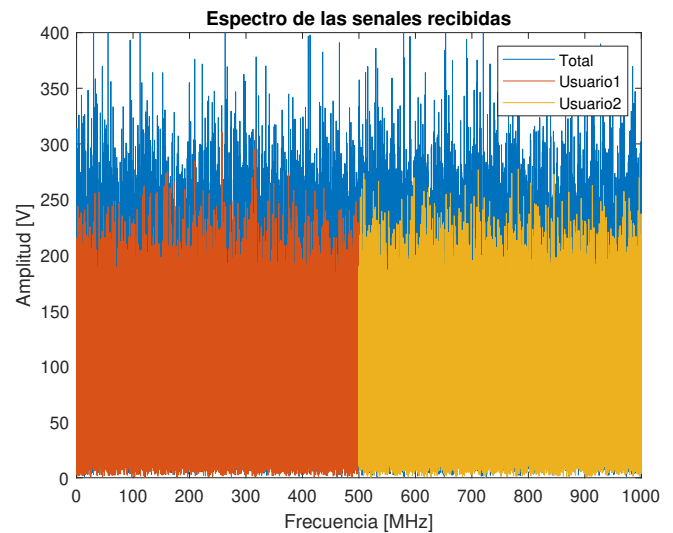


Fig. 9. Espectro de las señales de datos en recepción

El siguiente script permite obtener todos los resultados previamente obtenidos.

```
1 clear all
2 close all
3 clc
4 N=100; %cantidad de datos aleatorios
5 SNR=20; %valor de relaciones enal a
   ruido en dB
6 datos1=randi([0 1],N,1)'; %datos
   aleatorios usuario 1
```

```

7  datos2=randi([0 1],N,1)'; %datos
   aleatorios usuario 2
8  M=2;%cantidad de fases de la
   modulacion M-PSK
9  datosber=[datos1 datos2]; %union de
   los dos datos
10 n=8; %orden de la matriz
11 H=hadamard(n); %uso de la funcion
   hadamard de matlab
12 F2=H(2,:); %Seleccion de la fila 2 de
   la matriz de Hadamard
13 F6=H(6,:); %Seleccion de la fila 6 de
   la matriz de Hadamard
14 tb=1; %tiempo de bit
15 tc=tb/n; %tiempo de chip
16 gd1=[]; %vector vacio
17 for i=1:1:100 %lazo for que recorre
   los datos del usuario 1
18     if datos1(i)==1
19         for j=0:tc/100:tb-tc/100
20             gd1 = [gd1 1]; %creacion de la
               matriz de datos
21         end
22     else
23         for j=0:tc/100:tb-tc/100
24             gd1 = [gd1 -1]; %creacion de
               la matriz de datos
25         end
26     end
27 end
28 subplot(3,2,1)
29 txmod=linspace(0,length(datos1),length
   (gd1));
30 plot(txmod,gd1);
31 axis([0 10 -1.2 1.2])
32 title('Datos-Usuario 1')
33 xlabel('Tiempo[seg]')
34 ylabel('Amplitud [V]')
35
36 sec1=[];%vector vacio
37 for i=1:length(datos1) %lazo for que
   recorre el vector de datos 1
38     sec1=[sec1 F2];%se replica la
   secuencia de Hadamard
39 end
40 gc=[];%vector vacio
41 for i=1:length(sec1) %lazo for que
   recorre el vector obtenido
42     if sec1(i)==-1 %si la secuencia
   tiene el valor de -1 se genera un
   vector
43         for j=0:tc/100:tc-tc/100
44             gc = [gc -1]; %creacion de la
               matriz de datos con valor -1
45         end
46     else

```

```

47         for j=0:tc/100:tc-tc/100 %si
   la secuencia tiene el valor de 1 se
   genera un vector
48             gc = [gc 1]; %creacion de la
               matriz de datos con valor 1
49         end
50     end
51 end
52 subplot(3,2,3)
53 plot(txmod,gc);
54 axis([0 2 -1.2 1.2])
55 title('Secuencia Hadamard-Usuario 1')
56 xlabel('Tiempo[seg]')
57 ylabel('Amplitud [V]')
58
59 res1=gd1.*gc; %Producto entre la
   secuencia de Hadamard y los datos
60 subplot(3,2,5)
61 plot(txmod,res1);
62 axis([0 4 -1.2 1.2])
63 title('Senal obtenida-Usuario 1')
64 xlabel('Tiempo[seg]')
65 ylabel('Amplitud [V]')
66
67 respi=reshape(res1,800,100);%
   conversion de tamaño
68 X1=ifft(respi,800); %uso de la ifft
69 Xaux1=X1(1:4,:); %vector auxiliar
70 X1=[X1;Xaux1]; %registro de los
   valores obtenidos
71
72 gd2=[]; %vector vacio
73 for i=1:1:100 %lazo for que recorre el
   vector de datos 2
74     if datos2(i)==1 %si el dato es 1
   se genera un vector
75         for j=0:tc/100:tb-tc/100
76             gd2 = [gd2 1]; %creacion de la
               matriz de datos cuyo valor es 1
77         end
78     else
79         for j=0:tc/100:tb-tc/100
80             gd2 = [gd2 -1]; %creacion de
               la matriz de datos cuyo valor es -1
81         end
82     end
83 end
84 subplot(3,2,2)
85 txmod2=linspace(0,length(datos2),
   length(gd2));%vector auxiliar de
   tiempo
86 plot(txmod2,gd2);
87 axis([0 10 -1.2 1.2])
88 title('Datos-Usuario 2')
89 xlabel('Tiempo[seg]')
90 ylabel('Amplitud [V]')

```

```

91
92 sec2=[]; %vector vacio
93 for i=1:length(datos2) %lazo for que
    permite la creacion de una matriz
94     sec2=[sec2 F6];%registro de la
        fila de la matriz de hadamard
95 end
96 gc2=[];%vector vacio
97 for i=1:length(sec2) %lazo for que
    recorre el arreglo creado
    anteriormente
98     if sec2(i)==-1
99         for j=0:tc/100:tc-tc/100
100             gc2 = [gc2 -1]; %creacion
                de la matriz de datos
101         end
102     else
103         for j=0:tc/100:tc-tc/100
104             gc2 = [gc2 1]; %creacion
                de la matriz de datos
105         end
106     end
107 end
108 subplot(3,2,4)
109 plot(txmod,gc2);
110 axis([0 2 -1.2 1.2])
111 title('Walsh-Hadamart Usuario 2')
112 xlabel('Tiempo[seg]')
113 ylabel('Amplitud [V]')
114
115 res2=gd2.*gc2; %producto entre los
    datos y la secuencia de Hadamard
116 subplot(3,2,6)
117 plot(txmod,res2);
118 axis([0 4 -1.2 1.2])
119 title('Senal Usuario 2')
120 xlabel('Tiempo[seg]')
121 ylabel('Amplitud [V]')
122
123 respi2=reshape(res2,800,100); %
    conversion de tamano
124 X2=ifft(respi2,800); %uso de la ifft
125 Xaux2=X2(1:4,:);%vector auxiliar
126 X2=[X2;Xaux2]; %registro de los
    valores obtenidos
127
128 XF1=reshape(X1,1,80400); %conversion
    paralelo a serie
129 XF2=reshape(X2,1,80400); %conversion
    paralelo a serie
130 XF=[XF1 XF2]; %union de los usuarios
131
132 figure(5)
133 XT=XF.*cos(2*pi*(1*10)); %modulacion
    BPSK
134 txt=linspace(0,length(datos1)*2,length

```

```

    (XT));
135 plot(txt,real(XF));
136 axis([100 200 -0.6 0.6])
137 title('Senal transmitida total')
138 xlabel('Tiempo[seg]')
139 ylabel('Amplitud [V]')
140
141 figure(2)
142 hold on
143 espT=fft(XT); %uso de la fft
144 espT=fftshift(espT);%obtencion del
    espectro total
145 Fmd=linspace(-1000,1000,length(espT));
    %vector de tiempo auxiliar
146 plot(Fmd,abs(espT))
147 hold on
148 axis([0 1000 0 max(abs(espT))])
149 espT1=fft(XT(:,1:80400));%uso de la
    fft
150 espT1=fftshift(espT1); %obtencion del
    espectro del usuario 1
151 Fmd1=linspace(0,500,length(espT1)); %
    vector de tiempo auxiliar 1
152 plot(Fmd1,abs(espT1));
153 hold on
154 espT2=fft(XT(:,80401:160800));%uso de
    la fft
155 espT2=fftshift(espT2);%obtencion del
    espectro del usuario 2
156 Fmd2=linspace(500,1000,length(espT2));
    %vector de tiempo auxiliar 2
157 plot(Fmd2,abs(espT2));
158 title('Espectro de las senales
    transmitirdas')
159 xlabel('Frecuencia [MHz]')
160 ylabel('Amplitud [V]')
161 legend('Total','Usuario1','Usuario2')
162
163 XR=XT; %renombre de la variable XT
164 XR1=awgn(XT,SNR); %paso a traves de un
    canal awgn
165
166 figure(3)
167 espTr=fft(XR1); %uso de la fft en la
    senal transmitida total
168 espTr=fftshift(espTr); %obtencion del
    espectro
169 Fmdr=linspace(-1000,1000,length(espTr)
    ); %vector de tiempo auxiliar
170 plot(Fmdr,abs(espTr))
171 hold on
172 axis([0 1000 0 max(abs(espTr))])
173 espT1r=fft(XR1(:,1:80400));%uso de la
    fft en la senal transmitida del
    usuario 1
174 espT1r=fftshift(espT1r); %obtencion

```

```

175     del espectro
176 Fmd1r=linspace(0,500,length(espT1r));
177     %vector de tiempo auxiliar
178 plot(Fmd1r,abs(espT1r));
179 hold on
180 espT2r=fft(XR1(:,80401:160800)); %uso
181     de la fft en la senal transmitida
182     del usuario 2
183 espT2r=fftshift(espT2r); %obtencion
184     del espectro
185 Fmd2r=linspace(500,1000,length(espT2r)
186     ); %vector de tiempo auxiliar
187 plot(Fmd2r,abs(espT2r));
188 title('Espectro de las senales
189     recibidas')
190 xlabel('Frecuencia [MHz]')
191 ylabel('Amplitud [V]')
192 legend('Total','Usuario1','Usuario2')
193
194 XRD=XR./ (cos(2*pi*(1*10^9))); %
195     demodulacionn
196 X1R=[XRD(:,1:80400)]; %obtencion de la
197     senal del usuario 1
198 X2R=[XRD(:,80401:160800)]; %obtencion
199     de la senal dle usuario 2
200
201 X1R=reshape(X1R,804,100); %conversion
202     a paralelo
203 X2R=reshape(X2R,804,100); %conversion
204     a paralelo
205 X1RQ=X1R(1:800,:); %creacion de una
206     matriz
207 X2RQ=X2R(1:800,:); %creacion de una
208     matriz
209
210 X1RQF=fft(X1RQ,800); %uso de la fft en
211     la senal 1
212 X2RQF=fft(X2RQ,800); %uso de la fft en
213     la senal 2
214
215 X1RQFF=real(X1RQF); %obtencion de la
216     aprte real de la fft de la senal 1
217 X2RQFF=real(X2RQF); %obtencion de la
218     aprte real de la fft de la senal 2
219
220 X1RQFF=reshape(X1RQFF,1,80000); %
221     conversion de paralelo a serie
222 X2RQFF=reshape(X2RQFF,1,80000); %
223     conversion de paralelo a serie
224
225 datrec1=X1RQFF.*gc; %producto entre la
226     senal 1 obtenida y la secuencia de
227     hadamard
228
229 datrec2=X2RQFF.*gc2; %producto entre
230     la senal 2 obtenida y la secuencia
231     de hadamard
232
233 datrec1=awgn(datrec1,SNR); %paso a
234     traves del canal awgn
235
236 datrec2=awgn(datrec2,SNR); %paso a
237     traves del canal awgn
238
239 j=1; %variable auxiliar
240 for p=400:800:length(datrec1) %vector
241     que recorre la senal 1
242     if datrec1(p)>0.5 %dispositivo de
243         decision
244         dout1(j)=1;
245         j=j+1;%incremento en la variable
246         auxiliar
247     else
248         dout1(j)=0;
249         j=j+1; %incremento en la variable
250         auxiliar
251     end
252 end
253
254 j2=1; %variable auxiliar
255 for p2=400:800:length(datrec2) %vector
256     que recorre la senal 2
257     if datrec2(p2)>0.5 %dispositivo de
258         decision
259         dout2(j2)=1;
260         j2=j2+1;%incremento en la
261         variable auxiliar
262     else
263         dout2(j2)=0;
264         j2=j2+1;%incremento en la
265         variable auxiliar
266     end
267 end
268
269 doutF=[dout1 dout2]; %union de las dos
270     senales obtenidas
271
272 [h BER]=biterr(datosber,doutF); %
273     obtencion del BER
274
275 fprintf('La tasa de bit errado es:
276     %0.3f \n',BER)
277
278
279 gdlr=[]; %vector vacio
280 for i=1:1:100 %lazo for que recorre la
281     senal 1
282     if dout1(i)==1
283         for j=0:tc/100:tb-tc/100
284             gdlr = [gdlr 1]; %creacion de la
285             matriz de datos
286         end
287     else
288         for j=0:tc/100:tb-tc/100
289             gdlr = [gdlr -1]; %creacion de la
290             matriz de datos
291         end
292     end
293 end
294
295 gd2r=[];%vector vacio
296 for i=1:1:100 %lazo for que recorre la
297     senal 2
298     if dout2(i)==1

```



```

246     for j=0:tc/100:tb-tc/100
247         gd2r = [gd2r 1]; %creacion de la
           matriz de datos
248     end
249     else
250         for j=0:tc/100:tb-tc/100
251             gd2r = [gd2r -1]; %creacion de la
           matriz de datos
252         end
253     end
254 end
255 figure(4)
256 subplot(2,2,2)
257 plot(txmod,gdlr)
258 axis([0 100 -1.2 1.2])
259 title('Senal recuperada-Usuario 1')
260 xlabel('Tiempo [seg]')
261 ylabel('Amplitud [V]')
262
263 subplot(2,2,1)
264 plot(txmod,gdl);
265 axis([0 100 -1.2 1.2])
266 title('Datos-Usuario 1')
267 xlabel('Tiempo[seg]')
268 ylabel('Amplitud [V]')
269
270 subplot(2,2,4)
271 plot(txmod,gd2r)
272 axis([0 100 -1.2 1.2])
273 title('Senal recuperada-Usuario 2')
274 xlabel('Tiempo [seg]')
275 ylabel('Amplitud [V]')
276
277 subplot(2,2,3)
278 plot(txmod,gd2);
279 axis([0 100 -1.2 1.2])
280 title('Datos-Usuario 2')
281 xlabel('Tiempo[seg]')
282 ylabel('Amplitud [V]')

```

Script 2. Sistema CDMA

REFERENCES

- [1] A. Persson, T. Ottosson, E. Ström, y E. S. Om, "Time-Frequency Localized CDMA For Downlink Multi-Carrier Systems", en in Proc. of ISSSTA'2002, 2002, pp. 118–122.
- [2] E. Tatayo, "IMPLEMENTACIÓN DE UN SISTEMA CDMA BÁSICO". C.P. COMUNICACIÓN DIGITAL, Accedido: ago. 30, 2020. [En línea].
- [3] H. Sari, "A Review of Multicarrier CDMA", en Multi-Carrier Spread-Spectrum & Related Topics: Third International Workshop, September 26–28, 2001, Oberpfafenhofen, Germany, K. Fazel y S. Kaiser, Eds. Boston, MA: Springer US, 2002, pp. 3-12.
- [4] A. Marczak, "Performance analysis of data transmission in MC-CDMA radio interface with turbo codes", Telecommun Syst, vol. 59, núm. 4, pp. 501–507, ago. 2015, doi: 10.1007/s11235-014-9910-7.
- [5] A. J. Mendez, J. L. Lambert, J. M. Morookian, L. A. Bergman, y R. M. Gagliardi, "Applications of optical code division multiple access (CDMA) to wireless communication", en Wireless Communications, sep. 1995, vol. 2556, pp. 102–106, doi: 10.1117/12.220878.
- [6] "Wayback Machine-CDMA", dic. 10, 2004. <https://web.archive.org/web/20041210061929/whitepaper-pdf/service4.pdf> (accedido sep. 10, 2020).
- [7] "What is 3G UMTS: WCDMA Tutorial Electronics Notes". <https://www.electronics-notes.com/articles/connectivity/3g-umts/what-is-umts-wcdma-tutorial.php> (accedido sep. 10, 2020)
- [8] "Hadamard matrix - MATLAB hadamard - MathWorks América Latina". <https://la.mathworks.com/help/matlab/ref/hadamard.html> (accedido sep. 05, 2020).
- [9] J. Kaiser, M. Goldfarb, y S. LeSage, "A dual-mode CDMA/FM driver amplifier for cellular applications", Microwave Journal, vol. 39, núm. 1, pp. 68–77, ene. 1996.
- [10] A. Ghayas, "What is the difference between GSM, UMTS and LTE?", Commsbrief, sep. 13, 2019. <https://commsbrief.com/difference-between-gsm-umts-lte/> (accedido sep. 10, 2020).