

“IMPLEMENTACIÓN DE UN SISTEMA CDMA BÁSICO”

Trabajo Preparatorio N°11
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 presentará la información teórica necesaria para la implementación de un sistema CDMA básico. Así, se revisarán conceptos referentes a la generación de secuencias no ortogonales como Gold y Kasami y la generación de secuencias ortogonales OSVF y Hadamard-Walsh, para complementar esta información se tendrá en cuenta las definiciones de Correlación y Autocorrelación de las secuencias PN. Una vez recopilada toda la información necesaria se procederá a realizar una implementación de un sistema CDMA con la ayuda de Matlab.

Index Terms—CDMA, Secuencias PN, SF, Códigos OVSF

I. INTRODUCCIÓN

La técnica de acceso múltiple por división de código, su principal ventaja consiste en la capacidad de tolerar un gran cantidad de señales interferentes. Es así como un número de usuarios comparte un canal sin mecanismos de sincronización externos. Con esta técnica, la señal ocupa un ancho de banda muy superior a lo que sería estrictamente necesaria para su transmisión [1].

El problema del multitrayecto presente en los sistemas de comunicación inalámbrica se soluciona gracias a que, al utilizar señales de gran ancho de banda, se tiene mucha resolución en el dominio temporal, de manera que se pueden separar ecos de la señal muy próximos en el tiempo y combinarlos de forma coherente si se utiliza el receptor Rake.

La tecnología CDMA, se utiliza como tecnología en aplicaciones militares, donde el rechazo a la interferencia y la seguridad son unos de los factores más importantes. Así como para el rechazo al multitrayecto en entornos de radio móvil sobre tierra [1].

II. OBJETIVOS

- Analizar la generación de secuencias pseudoaleatorias (PN) utilizadas en sistemas CDMA.
- Implementar en Matlab un sistema de comunicación CDMA básico [2].

III. PREGUNTAS

A. Consultar sobre la generación de secuencias no ortogonales Gold y Kasami, describir las propiedades más relevantes de las mismas.

- **Códigos Gold:** Son los más adecuados para sistemas multi usuarios CDMA, debido a que su función de correlación ofrece mejores propiedades en comparación a las secuencias PN [3].

La compartición del espectro radioeléctrico entre muchos usuarios provoca que la señal de datos que llega a recepción sea una suma de varias señales, esto es debido a que la interferencia entre usuarios es grande. Por lo que cuando se trabaja con correlación cruzada, el valor debe ser mínimo con el fin de poder manejar mayor cantidad de usuarios en el sistema [4].

Las secuencias Gold, se generan mediante la combinación de dos secuencias concretas PN de misma longitud $L = 2^m - 1$ chis [3]. Son elegidas de tal forma que dan como resultado una baja correlación cruzada con 3 posibles valores: $(-1, -t(m), t(m) - 2)$ donde:

$$t(m) = \begin{cases} 2^{(m+1)/2} & \text{si } m = \text{impar} \\ 2^{(m+2)/2} & \text{si } m = \text{par} \end{cases} \quad (1)$$

En la figura 1, se presenta el proceso de formación de secuencias Gold.

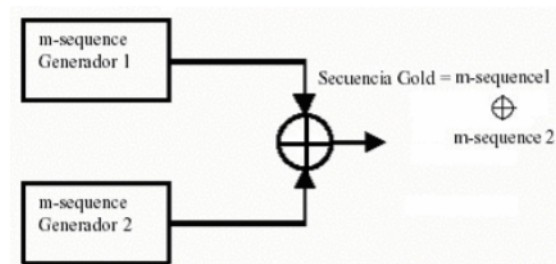


Fig. 1. Generación de una secuencia Gold [5]

Donde la secuencia obtenida también tiene una longitud de L chis. Además el número de secuencias posibles es $2^m + 1$, las mismas que son obtenidas de la suma en módulo 2 de una de las secuencias PN del par deseado con las versiones desplazadas de otra secuencia PN [5].

- **Códigos Kasami:** El conjunto de secuencias Kasami es uno de los más importantes tipos de secuencia binarias debido a su muy bajo valor de correlación cruzada. Existen dos tipos diferentes de secuencias Kasami [6]:

- **Pequeñas secuencias Kasami:** Son generadas de una manera similar a las secuencias Gold, de esta forma se pueden generar $M = 2^{m/2}$ secuencias binarias de período $L = 2^m - 1$ chips, donde m es par.

Para su generación, se parte de una secuencia PN, a , para generar una secuencia binaria b , como resultado de un diezmo de la secuencia a de $2^{m/2} + 1$ chips. Es así como se puede comprobar que la secuencia binaria b resultante es periódica en $2^{m/2} - 1$ [5].

Finalmente, el conjunto de secuencias de Kasami se obtiene tomando $L = 2^m - 1$ chips de la secuencia a y sumándolos en módulo 2 con la secuencia binaria b y todos los desplazamientos cíclicos de la misma. Así, se obtienen $2^{m/2}$ secuencias de Kasami de longitud $L = 2^m - 1$ chips [5].

- **Grandes secuencias Kasami:** Consisten en secuencias de período $N = 2^n - 1$, con n par. Están formadas por secuencias Gold y secuencias pequeñas de Kasami. Dada una secuencia PN, se obtienen por muestreo las secuencias a' y a'' con $2^{n/2} + 1$ y $2^{(n+2)/2} + 1$ puntos y luego, tomando todas las secuencias formadas por la secuencia PN, a' y a'' con diferentes desplazamientos de a' y a'' se obtienen estas secuencias de Kasami [7].

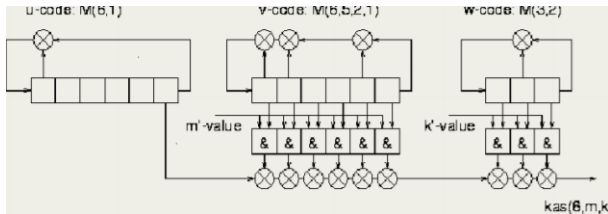


Fig. 2. Generación de una secuencia Kasami

B. Consultar sobre la generación de secuencias ortogonales OVSF (Ortogonal Variable Spreading Factor) y Hadamard-Walsh, describir las propiedades más relevantes de estas secuencias.

- **OVSF:** Es otro método para generar códigos ortogonales es utilizando estructuras de árbol con un factor de ensanchamiento variable [8].

La generación de esta estructura de árbol, tal y como se muestra en la figura 6 se realiza mediante el siguiente proceso recursivo:

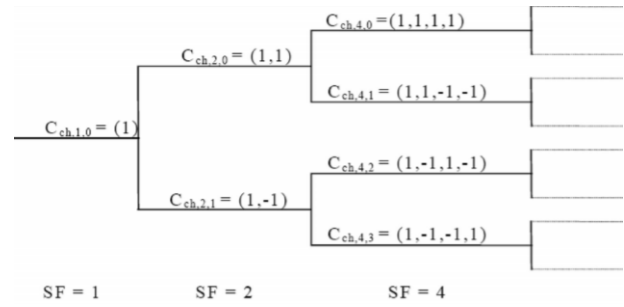


Fig. 3. Generación de códigos ortogonales OVSF mediante estructuras de árbol

$$C_{2n} = \begin{bmatrix} C_{2n,1} \\ C_{2n,2} \\ \dots \\ C_{2n,2n} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} C_{n,1} & C_{n,1} \\ C_{n,1} & C_{n,1} \end{bmatrix} \\ \dots \\ \begin{bmatrix} C_{n,n} & C_{n,n} \\ C_{n,n} & C_{n,n} \end{bmatrix} \end{bmatrix} \quad (2)$$

Donde C_{2n} es un conjunto de códigos ortogonales de tamaño '2n' y SF es el "spreading factor" o factor de ensanchamiento. El valor más a la izquierda en cada código es el transmitido en primer lugar [9].

Las propiedades de ortogonalidad de estos códigos son similares a las de los códigos Walsh. El orden de las funciones de la matriz no es el mismo que el de la matriz de Hadamard, pero las funciones en sí son las mismas [8]. Las secuencias pertenecientes a la misma rama forman un conjunto de códigos ortogonales, es más, dos secuencias cualesquiera de diferentes ramas son ortogonales excepto si una secuencia es la madre de la otra [10].

En estos códigos ortogonales, si dos secuencias llegan al mismo tiempo al receptor, la interferencia que se causarían entre dos usuarios sería nula. Lo que ocurre en el enlace ascendente, es que es muy difícil que los terminales emitan a la vez y los códigos estén alineados en el tiempo. Si existiera algún desplazamiento en el tiempo, las propiedades de correlación cruzada empeorarían mucho, por lo que resultaría poco aconsejable su utilización en el enlace ascendente [10].

- **Hadamard-Walsh:** Los códigos Walsh son unos de los códigos ortogonales más comunes usados en aplicaciones CDMA [8]. Estos códigos corresponden a las columnas de una matriz cuadrada especial conocida como la matriz de Hadamard.

Para un conjunto de códigos Walsh de longitud 'n', se tiene una matriz cuadrada de 'n' códigos Walsh de forma que el orden dicha matriz es 'n x n'. La primera columna de esta matriz contiene una serie de 1's y las siguientes columnas contienen una serie de -1's y 1's combinados [10]. Cada columna es ortogonal entre sí y tienen la misma aparición de los bits binarios. Esta matriz se define recursivamente de la siguiente manera:

$$W_1 = \begin{bmatrix} 1 \end{bmatrix} \quad (3)$$

$$W_{2n} = \begin{bmatrix} W_n & W_n \\ W_n & \overline{W_n} \end{bmatrix} \quad (4)$$

En la ecuación (4), 'n' es una potencia de 2 que indica la dimensión de la matriz y $\overline{W_n}$ hace referencia al operador lógico NOT en todos los bits de esa matriz [10].

Se puede ver con facilidad que todas las filas y columnas son mutuamente ortogonales. Las siguientes propiedades se pueden derivar si se define la secuencia de Walsh W_i como la i-ésima fila o columna de una matriz de Hadamard [8]:

- Las secuencias de Walsh son secuencias binarias con valores de +1 y -1.
 - La longitud de las secuencias de Walsh son siempre potencia de 2.
 - Siempre hay 'L' secuencias diferentes de longitud 'L' [9].
 - Las secuencias de Walsh son mutuamente ortogonales si están sincronizadas, es decir, $\phi_{xy}(l=0) = 0$.
 - Si dos secuencias de Walsh tienen desplazamientos en el tiempo, la función de correlación cruzada puede tomar valores mayores que el pico de la función de autocorrelación, el cual es igual a la longitud 'L' de la secuencia [9].
- Aunque también es posible que la función de correlación cruzada tome un valor de cero incluso cuando existe cualquier desplazamiento en el tiempo.
- Todas las secuencias de Walsh empiezan por +1 [10].

C. Consultar las características de Correlación y Autocorrelación de las secuencias PN.

Correlación en secuencias PN

La correlación cruzada como normalmente se encuentra definida como una medida de similitud entre dos señales. Este análisis se lo hará en base a la suposición de tener una señal $x_1[n]$ y una $x_2[n]$ de igual longitud N, la correlación cruzada C_{12} , se puede determinar con la suma de los productos de ambas señales para un determinado tamaño, tal y como se puede apreciar en la ecuación 5. [11] [12] [4]

$$C_{12} = \sum_{n=0}^{N-1} x_1[n] \cdot x_2[n] \quad (5)$$

Una correlación cruzada negativa, indica que un incremento en una variable está asociado con el decremento de otra variable. Mientras que una correlación de cero indicaría que una señal no influye en otra, pero sería sólo un caso ideal. [4]

En las secuencias PN usadas en sistemas CDMA, estas deben ser ortogonales para cada usuario, de tal manera que la correlación cruzada de estas debe ser cero, pero esto podría cumplirse sólo en situaciones en las que la longitud de la señal sea pequeña. Por otro lado, cuando estas señales tienden

a tener un mayor tamaño se presenta una correlación distinta a cero, que si bien no es posible de eliminar es su totalidad se busca reducir al máximo dicha característica. [4]

Autocorrelación en secuencias PN

La autocorrelación es un término que se usa para definir la correlación cruzada de una señal consigo misma. Eso sirve para determinar patrones repetitivos de una señal, como la periodicidad de una señal enmascarada bajo el ruido o para identificar la frecuencia fundamental de una señal que no contiene dicha componente, pero aparecen numerosas frecuencias armónicas de esta. [4] [13]

La función de autocorrelación viene definida de la siguiente manera.

$$C(k) = \sum_{n=1}^N a_n \cdot a_{n-k} \begin{cases} N & k = 0, N, 2N \\ -1 & \text{el resto} \end{cases} \quad (6)$$

En una secuencia PN, la autocorrelación tiene el mismo comportamiento que el ruido, tal y como se ilustra en la figura 4.

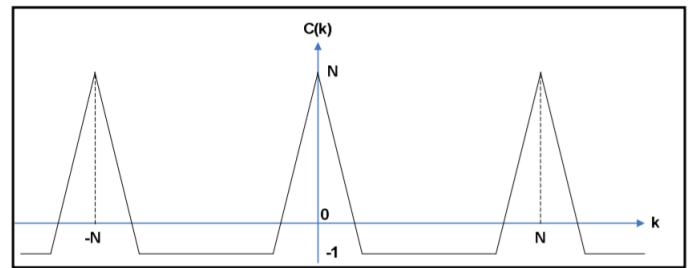


Fig. 4. Autocorrelación de una M-secuencia [4]

D. Desarrollar un script en Matlab que permita generar un código OVFS de acuerdo con los siguientes factores de ensanchamiento (SF):

- GR2: $SF = 4$, índice del código: 2
- GR5: $SF = 8$, índice del código: 5
- GR9: $SF = 16$, índice del código: 8

Según la distribución presentada previamente, el código que se debe implementar es aquel cuyo factor de ensanchamiento es de 16 bits y el índice del código es el octavo. El script 1, permite obtener dicha secuencia de código donde el factor de ensanchamiento y el índice del código pueden ser cambiado.

```
1 clear all
2 close all
3 clc
4
5 SF=16; %variable que describe el
    factor de ensanchamiento
6 In=8; %índice del código deseado
7 M=SF; %numero de muestras por trama
```

```

8 hOVSF = comm.OVSFCode('SpreadingFactor', SF, 'Index', In, 'SamplesPerFrame', M);
9 seq = step(hOVSF)'; %secuencia de c\
    odigo
10
11 Seq=strjoin(cellstr(num2str(seq')),','); %Construccion del vector de
    secuencia con comas entre elementos
12 fprintf('El codigo OVSF es: [ %s ]\n', Seq); %impresion de la secuencia
    OVSF obtenido

```

Script 1. Obtención de un señal OFDM con 64 subportadoras

La siguiente figura, presenta el resultado del código obtenido:

El codigo OVSF es: [1,-1, 1,-1, 1,-1, 1,-1, 1,-1, 1,-1, 1,-1, 1,-1]

Fig. 5. Código OVSF obtenido

E. Desarrollar un script en Matlab que permita implementar un transmisor y receptor CDMA como se indica en la figura 6 empleando un generador de códigos Gold. Pasar la señal modulada a través de un canal AWGN. Mostrar en un solo gráfico de 4x2: la secuencia de bits de entrada, el código de Gold generado, la secuencia expandida, la señal modulada, la señal recibida, la señal demodulada, la secuencia expandida recibida y la secuencia de bits recuperada. Emplear la siguiente asignación de técnicas de modulación:

- GR2: QPSK, SNR: 15 dB
- GR5: 8-PSK, SNR: 20 dB
- GR9: 4-QAM, SNR: 30 dB

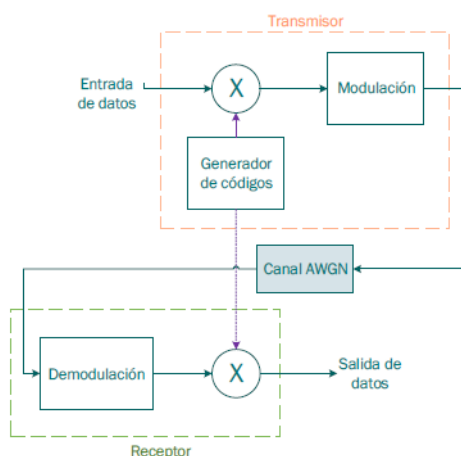


Fig. 6. Sistema CDMA

En la figura 7, se presenta el transmisor y receptor respectivamente de un sistema CDMA básico.

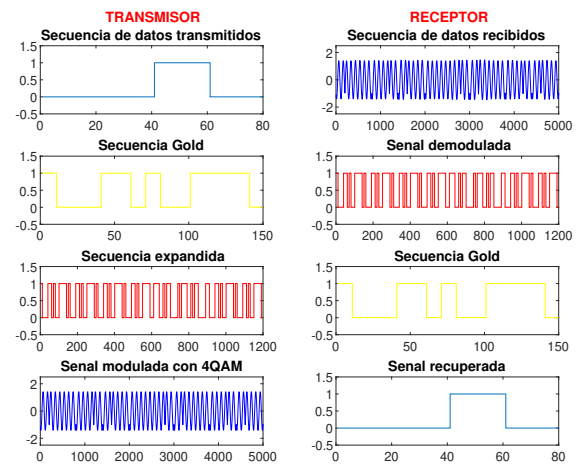


Fig. 7. Transmisor y receptor CDMA

Sin embargo, para obtener una mejor visualización, se presentan las imágenes de los datos obtenidos en transmisión y recepción por separado. La figura 8 muestra la implementación de un transmisor CDMA, mientras que la figura 9 muestra el proceso de recepción de los mismos datos

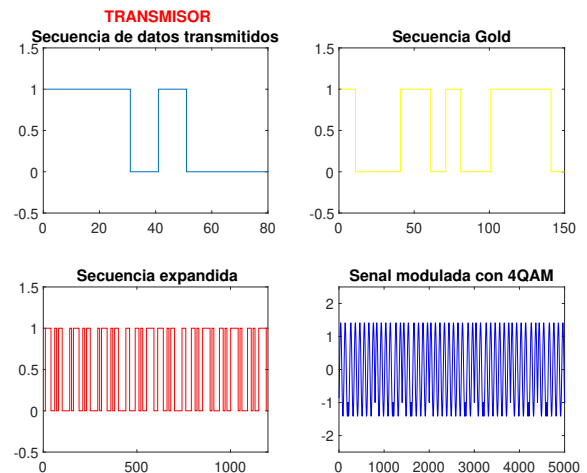


Fig. 8. Transmisor CDMA

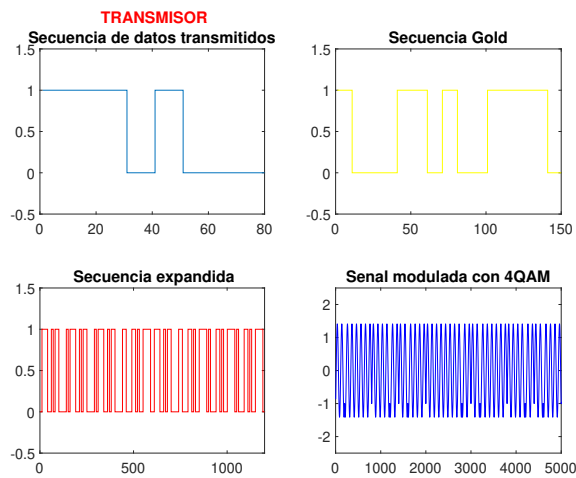


Fig. 9. Receptor CDMA

Estos resultados se obtuvieron en base al script 2.

```

1 %% SISTEMA CDMA BASICO
2 close all;
3 clear all;
4 clc;
5 %% TRANSMISOR
6 %GENERACION DE LA SECUENCIA DE DATOS Y
  DE LA SECUENCIA EXPANDIDA
7 Senal_extendida=[];
8 N1=8; %longitud del vector de datos
9 datos=randi([0 1],N1,1)'; %vector de
  datos aleatorio
10 N=15; %longitud del codigo gold
11 I=2; %indice del codigo gold
12 H = comm.GoldSequence('SamplesPerFrame
  ',N,'Index',I); %objeto que permite
  la creacion del codigo gold
13 secGold=H()'; %secuencia del codigo
  gold obtenida
14 k=1; %variable auxiliar para recorrer
  el lazo for
15
16 %GENERACION DE LA SECUENCIA EXPANDIDA
17 for i=1:N1 %lazo for anidado para
  realizar la expansion
18     for j=1:N
19         Senal_extendida(1,k)=xor(datos
          (1,i),secGold(1,j)); %uso de la
          funcion xor para el calculo del
          resultado entre la secuencia gold y
          los datos
20         k=k+1; %incremento en la
          variable auxiliar
21     end
22 end
23
24 %GRAFICA DE LAS SECUENCIAS DE DATOS
25 figure
26 Tb=10; %variable que describe el
  tiempo de bit
27 Dato_extendido = rectpulse(datos(1,:)
  ,10); %se extienden los datos para
  la grafica
28 subplot(4,2,1);
29 stairs(Dato_extendido); %grafica de
  los datos extendidos
30 axis([0 length(Dato_extendido) -0.5
  1.5]) %ejes
31 title({'\color{red} TRANSMISOR';'
  Secuencia de datos transmitidos'});
  %titulo
32
33 %GRAFICA DE LA SECUENCIA GOLD
34 Gold_extendida = rectpulse(secGold
  (1,:),10); %extension de la
  secuencia Gold
35 subplot(4,2,3);
36 stairs(Gold_extendida,'y'); %grafico
  de la secuencia Gold
37 axis([0 length(Gold_extendida) -0.5
  1.5])
38 title(' Secuencia Gold '); %titulo
39
40 %GRAFICA DE LA SECUENCIA EXPANDIDA
41 Senal1 = rectpulse(Senal_extendida
  (1,:),10); %tension de la secuencia
  expandida
42 subplot(4,2,5);
43 stairs(Senal1,'r'); %grafico de la
  secuencia expandida
44 axis([0 length(Senal1) -0.5 1.5]) %
  ejes
45 title(' Secuencia expandida') %titulo
46
47 %ETAPA DE MODULACION M-QAM
48 bp = 100; %tiempo de bit
49 M=4; %Numero de fase de modulacion
50 semod = qammod(Senal_extendida,M); %
  modulacion de la senal con M fases
51
52 %OBTENCION DE LOS DATOS PARA GRAFICAR
  4QAM
53 RealMod=real(semod); %parte real de la
  modulacion
54 ImagMod=imag(semod); %parte imaginaria
  de la modulacion
55 sp=bp*2; %tiempo de spread (2 veces el
  tiempo de bit)
56 sr=1/sp; %velocidad de spread
57 f=sr*2;%frecuencia
58 t=sp/100:sp/100:sp; %vector de tiempo
59 ss=length(t); %longitud del vector del

```

```

60     tiempo
    m=[]; %vector vacio para registrar las
        componentes reales e imaginarias
        de la modulacion
61
62 %CREACION DE LA SENAL MODULADA
63
64 for(k=1:length(RealMod))
65     yr=RealMod(k)*cos(2*pi*f*t); %
        producto entre la portadora Q y la
        parte real
66     yim=ImagMod(k)*sin(2*pi*f*t); %
        producto entre la portadora I y la
        parte imaginaria
67     y=yr+yim; %suma de ambas
        componentes
68     m=[m y]; %registro en el vector
        creado anteriormente
69 end
70 Tiempo=sp/100:sp/100:sp*length(RealMod
    ); %vector de tiempo
71 subplot(4,2,7);
72 plot(Tiempo,m,'b'); %grafica de la
        senal modulada
73 axis([0 5000 -2.5 2.5]) %ejes
74 title('Senal modulada con 4QAM'); %
        titulo
75
76 %% CANAL CON RUIDO AWGN
77
78 SNR = 30; %variable que describe la
        relacion senal a ruido en dB
79 ruido = awgn(semod,SNR); %paso a
        traves de un canal AWGN
80
81 bp = 100; %tiempo de bit
82 RealMod=real(ruido); %obtencion de la
        parte real de la senal con ruido
83 ImagMod=imag(ruido); %obtencion de la
        parte imaginaria de la senal con
        ruido
84 sp=bp*2; %tiempo de spread
85 sr=1/sp; %velocidad de spread
86 f=sr*2; %frecuencia
87 t=sp/100:sp/100:sp; %vector de tiempo
88 ss=length(t); %longitud del vector de
        tiempo
89 m=[]; %vector vacio que permite
        guardar en memoria los datos
        obtenidos
90 for(k=1:length(RealMod)) %lazo for
        que permite realizar la suma de la
        parte real e imaginaria de la senal
91     yr=RealMod(k)*cos(2*pi*f*t); %canal
        Q
92     yim=ImagMod(k)*sin(2*pi*f*t); %

```

```

        canal I
93     y=yr+yim; %suma del canal Q con el
        canal I
94     m=[m y]; %registro de los valores
        obtenidos
95 end
96 Tiempo=sp/100:sp/100:sp*length(RealMod
    ); %vector de tiempo para graficar
97 subplot(4,2,2);
98 plot(Tiempo,m,'b'); %grafica de la
        senal obtenida
99 axis([0 5000 -2.5 2.5]) %eje
100 title({'\color{red} RECEPTOR';'
        Secuencia de datos recibidos'}); %
        titulo
101
102 %% RECEPCION
103 %DEMULACION DE LA SENAL RECIBIDA
104 sedemod = qamdemod(ruido,M); %
        demulacion de la senal de datos
        con M fases
105 sedemod_pulso = rectpulse(sedemod(1,:),
    10); %extension de la senal
        demodulada
106 subplot(4,2,4);
107 stairs(sedemod_pulso,'r'); %grafica
        de la senal demodulada
108 axis([0 length(sedemod_pulso) -0.5
    1.5]) %ejes
109 title(' Senal demodulada') %titulo
110
111 %RECUPERACION DE LA SENAL ORIGINAL
112 i=1; %variable auxiliar para el lazo
        for
113 k=1; %variable auxiliar para el bucle
        while
114 los= length(sedemod); %longitud de los
        datos demodulados
115 while k < los %bucle while permite
        recurrir hasta la longitud maxima
        del vector
116     s=0; %variable auxiliar
117 for j=1:N %lazo for que permite
        realizar la multiplicacion de los
        datos recibidos con la secuencia
        Gold
118     SenalRecibida(1,j) = xor(sedemod
        (1,k),secGold(1,j)); %producto xor
        entre la senal demodulada y la
        secuencia Gold
119     k=k+1; %incremento de la variable
        k
120     s=s+SenalRecibida(1,j); % se
        guardan los valores obtenidos del
        producto
121 end

```



```

122     %la siguiente condicion permite
    implementar el dispositivo de
    decision
123     if(s==0) %si s es cero, se designa
        el valor de 0
124         b2(1,i) = 0;
125     else %caso contrario el valor de 1
126         b2(1,i) = 1;
127     end
128     i=i+1; %se incrementa la
        variable i
129 end
130 Senal_no_ensanchada= b2; %obtencion de
        la senal no ensanchada
131
132 p=[]; %vector vacio que permite
        registrar valores
133 for k=1:N1 %lazo for que recorre la
        senal no ensanchada para expandirla
134     if b2(1,k)==0 %si el dato k es 0,
        se asignan 10 ceros
135         senal1=zeros(1,10);
136     else %caso contrario, se asignan
        10 unos
137         senal1=ones(1,10);
138     end
139     p=[p senal1]; %registro de los
        valores asignados
140 end
141 %GRAFICA DE LAS SENALES RECUPERADAS
142 Gold_extendida = rectpulse(secGold
        (1,:),10); %extension de la
        secuencia Gold extendida
143 subplot(4,2,6);
144 stairs(Gold_extendida,'y'); %grafica
        de la secuencia Gold
145 axis([0 length(Gold_extendida) -0.5
        1.5]) %ejes
146 title('Secuencia Gold'); %titulo
147
148 subplot(4,2,8);
149 stairs(p); %grafica de la senal
        recuperada extendida
150 axis([0 length(p) -0.5 1.5]) %ejes
151 title('Senal recuperada') %titulo

```

Script 2. Código para transmisión y recepción 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] "(PDF) On the Effectivity of Different Pseudo-Noise and Orthogonal Sequences for Speech Encryption from Correlation Properties", ResearchGate. <https://www.researchgate.net/publication/255580061On-the-Effectivity-of-Different-Pseudo-Noise-and-Orthogonal-Sequences-for-Speech-Encryption-from-Correlation-Properties/figures> (accedido ago. 30, 2020).
- [4] A. A. Christian y R. U. Jonnathan, "MATERIAL DIDÁCTICO PARA EL ESTUDIO Y SIMULACIÓN DE CDMA: APLICACIÓN A COMUNICACIONES MÓVILES", p. 10.
- [5] "Acceso múltiple por división de código". <http://bibing.us.es/proyectos/abreproy/10930/fichero/Memoria%252FIV-CDMA.pdf> (accedido ago. 30, 2020).
- [6] "Sistemas CDMA". <http://bibing.us.es/proyectos/abreproy/11479/fichero/3-Sistemas+CDMA.pdf> (accedido ago. 30, 2020).
- [7] C. Pérez, "Generación y correlación eficiente de códigos binarios derivados de conjuntos de secuencias complementarias para sistemas ultrasónicos", Universidad de Alcalá.
- [8] J Andrew Richardson, "WCDMA Design Handbook" 2005.
- [9] J The 3rd Generation Partnership Project: Technical Specification Group Radio Access Network; Spreading and Modulation (FDD); <http://www.3gpp.org/>
- [10] I. Mingote, "Estudio de esquemas de diversidad cooperativa en sistemas CDMA", UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITÉCNICA SUPERIOR.
- [11] "correlación cruzada - español - definición, gramática, pronunciación, sinónimos y ejemplos — Glosbe". <https://es.glosbe.com/es/es/correlaci%C3%B3n%20cruzada> (consultado sep. 03, 2020).
- [12] "CORRELACIÓN CRUZADA Y AUTOCORRELACIÓN." <http://www.ehu.eus/Procesadodesenales/tema8/corre1.html> (consultado sep. 03, 2020).
- [13] "Cross-correlation - MATLAB xcorr - MathWorks América Latina". <https://la.mathworks.com/help/matlab/ref/xcorr.html> (accedido sep. 03, 2020).