

# “OFDM SOBRE UN CANAL AWGN”

Trabajo Preparatorio N°10  
Laboratorio de Comunicación Digital-GR9

1<sup>st</sup> Melanny Dávila  
Ingeniería en Telecomunicaciones  
Facultad de Eléctrica y Electrónica  
Quito, Ecuador  
melanny.davila@epn.edu.ec

2<sup>nd</sup> Ronaldo Almachi  
Ingeniería en Telecomunicaciones  
Facultad de Eléctrica y Electrónica  
Quito, Ecuador  
ronaldo.almachi@epn.edu.ec

**Abstract**—En este documento se va a presentar un script generado en Matlab, el mismo que implementa la técnica de transmisión OFDM en sistema de comunicaciones que usa una técnica de modulación digital QPKS. Así mismo, pasa a través de un canal con ruido AWGN con un valor específico de SNR. Para la comprobación del funcionamiento del sistema de una manera adecuada se va transmitir y recibir una imagen de 680x680 pixeles con formato BMP. El resultado del programa mostrará un gráfico en donde se tiene la imagen original y la imagen recuperada en el receptor, con su respectivo diagrama de constelación.

**Index Terms**—OFDM, prefijo cíclico, SNR, M-PSK, M-QAM.

## I. INTRODUCCIÓN

Orthogonal Frequency Division Multiplexing (OFDM), es una técnica de transmisión multiportadora de banda ancha. Es una tecnología que se ha usado desde la década de los 70, y que a lo largo del tiempo ha evolucionando del tal manera que aún sigue vigente su uso en sistemas de comunicaciones inalámbricos. OFDM es ampliamente usado pues permite usar diferentes técnicas de modulación como M-PSK o M-QAM. Sin embargo, para implementar esta tecnología se debe seguir los estándares que rigen actualmente al desarrollo de las telecomunicaciones como lo son, para WLAN (IEEE 802.11 a, g), o redes de mayor cobertura WiMAX (IEEE 802.16), sistemas de radio difusión digital DAB y sistemas de televisión digital DVB. [1]

Existen dos tipos de OFDM: continuo y discreto en tiempo. OFDM continuo en tiempo es un sistema que no empleaba modulación digital y por esto tiene capacidad limitada; mientras que OFDM discreto en tiempo, es uno de los que se usa actualmente debido a la relativa facilidad de implementación mediante la transformada de Fourier y su inversa, en donde si se usa técnicas de modulación, con todas las ventajas que ofrece un sistema OFDM como lo es la eficiencia espectral [1].

Como en todo sistema de comunicaciones se debe tener precaución al implementar OFDM, pues se debe tener en consideración todos los factores que influyen para la generación de errores durante la transmisión. Es por eso que se utiliza herramientas como, Matlab, CST, ect. para tener una idea

aproximada de como el sistema se va a comportar en una situación real [1].

## II. OBJETIVOS

- Implementar en Matlab un sistema de comunicación OFDM.
- Evaluar el desempeño de los distintos esquemas de modulación convencionales en conjunto con OFDM a través de un canal AWGN. [2].

## III. PREGUNTAS

A. Desarrollar un script en Matlab que permita implementar el sistema de comunicación presentado en la Figura 1. El programa debe tomar como entrada de datos una imagen en formato BMP o PNG de resolución de 680 pixeles de alto x 680 pixeles de ancho, para luego ser transmitida sobre un canal AWGN y finalmente ser recuperada en el receptor. Mostrar en un mismo gráfico (subplot 2x2) la imagen original transmitida, la imagen recibida, el diagrama de constelación de la señal OFDM transmitida y el diagrama de constelación de la señal OFDM recibida. Implementar el sistema de acuerdo con los parámetros asignados a cada grupo indicados en la Tabla 1.

Grupo	Tamaño FFT/IFFT	Long. Prefijo Cíclico	Esquema de modulación	SNR
GR2	128	8	8-PSK	20 dB
GR5	512	4	16-QAM	30 dB
GR9	256	16	QPSK	25 dB

TABLE I  
DISTRIBUCIÓN DE LAS CARACTERÍSTICAS

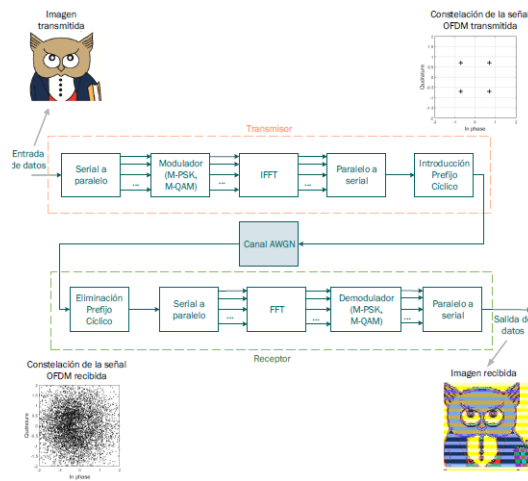


Fig. 1. Sistema de comunicación OFDM

En la figura 2, se presenta el resultado de un sistema de comunicaciones con OFDM programado en Matlab con las características solicitadas en la tabla 1 del trabajo preparatorio.

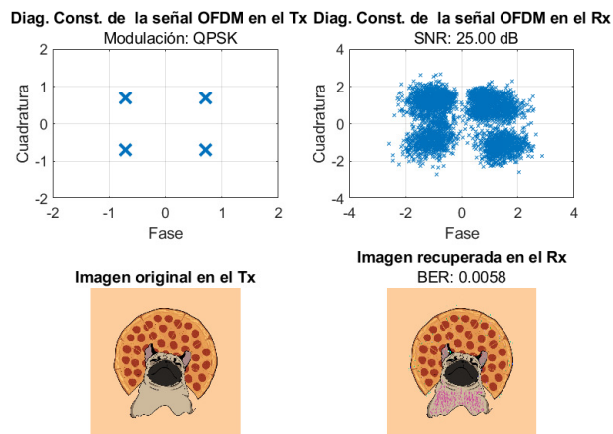


Fig. 2. Resultados obtenidos

El resultado que se muestra en la figura 2, se obtiene con el script 1.

```

1 clear all
2 close all
3 clc
4
5 modulacion='QPSK'; %modulacion: BPSK,
   QPSK, 8PSK, 16QAM, 32QAM, 64QAM
6
7 fft_long=256; %tamano de ifft/fft
8
9 PC=16; %tamano del prefijo ciclico
10
11 SNR=25; %relacion senal a ruido en dB

```

```

12
13 num_taps=8; %numero de taps de canal
   (1=no canal)
14
15 estim_canal='LS'; %metodo de
   estimacion de canal: ninguno
16
17 %calculo del orden de la modulacion
   desde el metodo de modulacion
18 metodos_mod={'BPSK','QPSK','8PSK','16
   QAM','32QAM','64QAM'};
19 orden_mod=find(ismember(metodos_mod,
   modulacion));
20
21 %cambio de los datos a un conjunto de
   bits
22 imagen=imread('cartoon.bmp'); %lectura
   de la imagen
23 imagen_bin=dec2bin(imagen(:)); %
   imagen en bits
24 imagen_bin=imagen_bin(:); %vector
   imagen
25
26 %cambio de bits a simbolos, los pads
   hacen que la senal tenga una
   longitud
27 %apropiada
28 simb_rem=mod(orden_mod-mod(length(
   imagen_bin),orden_mod),orden_mod);
   %modulacion de los bits
29 padding= repmat('0',simb_rem,1); %
   completacion de la longitud de la
   senal
30 imag_bin_padded=[imagen_bin;padding];
   %imagen suavizada
31 constelacion_datos=reshape(
   imag_bin_padded,orden_mod,length(
   imag_bin_padded)/orden_mod); %
   creacion de la constelacion de los
   datos
32 constelacion_simb=bin2dec(
   constelacion_datos); %creacion de
   la constelacion de simbolos
33
34 %simbolos de la modulacion
35 %BPSK
36 if orden_mod==1 %si se utiliza un bit
37     mod_ind=2^(orden_mod-1);
38     n=0:pi/mod_ind:2*pi-pi/mod_ind;
39     in_phase=cos(n); %portadora en
   fase
40     quadrature=sin(n); %portadora en
   cuadratura
41     symbol_book=(in_phase+quadrature*1
   i)';
42 end

```

```

43
44 %desplazamiento de fase alrededor de
    un circuito de radio 1
45 if orden_mod==2 || orden_mod==3 %si se
    utiliza 2 o 3 bits
46     mod_ind=2^(orden_mod-1);
47     n=0:pi/mod_ind:2*pi-pi/mod_ind;
48     in_phase=cos(n*pi/4); %portadora
    en fase
49     quadrature=sin(n*pi/4); %portadora
    en cuadratura
50     symbol_book=(in_phase+quadrature*1
        i)';
51 end
52
53 %Modulaciones 16QAM y 64QAM
54 if orden_mod==4||orden_mod==6 %si se
    utilizan 4 o 6 bits
55     mod_ind=sqrt(2^orden_mod);
56     in_phase=repmat(linspace(-1,1,
        mod_ind),mod_ind,1); %portadora en
    fase
57     quadrature=repmat(linspace(-1,1,
        mod_ind)',1,mod_ind); %portadora en
    cuadratura
58     symbol_book=in_phase(:)+quadrature
        (:)*1i;
59 end
60
61
62 %Modulacion 32QAM
63 if orden_mod==5 %si utilizan 5 bits
64     mod_ind=6;
65     in_phase=repmat(linspace(-1,1,
        mod_ind),mod_ind,1); %portadora en
    fase
66     quadrature=repmat(linspace(-1,1,
        mod_ind)',1,mod_ind); %portadora en
    cuadratura
67     symbol_book=in_phase(:)+quadrature
        (:)*1i;
68     symbol_book=symbol_book([2:5 7:30
        32:35]);
69 end
70
71 %modulacion de los datos de acuerdo al
    libro de simbolos
72 X=symbol_book(constelacion_simb+1);
73
74 %Uso de la IFFT para el dominio de
    tiempo
75 fft_tiempo=mod(fft_long-mod(length(X),
    fft_long),fft_long);
76 Senal_padded=[X;zeros(fft_tiempo,1)];
77 Senal_bloques=reshape(Senal_padded,
    fft_long,length(Senal_padded)/
        fft_long);
78 x=ifft(Senal_bloques);
79
80 %Adicion del prefijo ciclico y cambio
    de paralelo a serie
81 Senal_PC=[x(end-PC+1:end,:);x];
82 Senal_TX=Senal_PC(:); %senal lista
    para ser transmitida
83
84 %Paso a traves de un canal AWGN
85 Potencia_datos=mean(abs(Senal_TX.^2));
    %calculo de la potencia de los
    datos
86
87 %Adicion de ruido al canal
88 Potencia_ruido=Potencia_datos/10^(SNR
    /10); %calculo de la potencia del
    ruido
89 Ruido=normrnd(0,sqrt(Potencia_ruido/2)
    ,size(Senal_TX))+normrnd(0,sqrt(
    Potencia_ruido/2),size(Senal_TX))*1
    i;
90 Senal_ruido=Senal_TX+Ruido;
91
92 %medicion del SNR
93 Med_SNR=10*log10(mean(abs(Senal_TX.^2)
    )/mean(abs(Ruido.^2)));
94
95 %multitrayecto (aplicacion de fading)
96 M=exp(-(0:num_taps-1));
97 M=M/norm(M);
98 Senal_ruido_fading=conv(Senal_ruido,M,
    'same'); %senal con ruido suavizada
99
100 %Uso de la FFT para pasar al dominio
    de frecuencia
101
102 %Eliminacion del prefijo ciclico y
    paso de serial a paralelo
103 Senal_sinCP=reshape(Senal_ruido_fading
    ,fft_long+PC,length(
    Senal_ruido_fading)/(fft_long+PC));
    %senal sin prefijo ciclico
104 Senal_Completa_sinCP=Senal_sinCP(PC+1:
    end,:);
105
106 Senal_RX_bloques=fft(
    Senal_Completa_sinCP);%cambio al
    dominio de la frecuencia
107
108 %Estimacion del canal
109 if num_taps>1
110     switch (estim_canal)
111         case 'none'
112         case 'LS'
113             G=Senal_RX_bloques(:,1)./

```

```

114     Senal_bloques(:,1);
        Senal_RX_bloques=
        Senal_RX_bloques./ repmat(G,1,size(
        Senal_RX_bloques,2));
115     end
116 end
117
118 %Demodulacion de los simbolos
119 %remover el padding de fft
120 X_hat=Senal_RX_bloques(:);
121 X_hat=X_hat(1:end-fft_tiempo);
122
123 %Recuperacion de los datos de los
        simbolos demodulados
124 rec_syms=knnsearch([real(symbol_book)
        imag(symbol_book)],[real(X_hat)
        imag(X_hat)])-1;
125
126 %Cambio de secuencia de bits y
        eliminacion del simbolo de padding
127 rec_syms_cons=dec2bin(rec_syms);
128 rec_im_bin=reshape(rec_syms_cons',
        numel(rec_syms_cons),1);
129 rec_im_bin=rec_im_bin(1:end-simb_rem);
130 ber=sum(abs(rec_im_bin-imagen_bin))/
        length(imagen_bin);
131
132 %%Recuperacion de la imagen
133 rec_im=reshape(rec_im_bin,8,numel(
        rec_im_bin)/8);
134 rec_im=uint8(bin2dec(rec_im));
135 rec_im=reshape(rec_im,size(imagen));
136
137 %Generacion de las gr\'aficas
138
139 %constelacion de de la senal
        transmitida
140 subplot(2,2,1);
141 plot(X,'x','linewidth',2,'markersize'
        ,10);
142 xlim([-2 2]);
143 ylim([-2 2]);
144 xlabel('Fase');
145 ylabel('Cuadratura');
146
147 if num_taps>1
148     title(sprintf('\bDiag. Const. de
        la senal OFDM en el Tx \n \rm
        Modulacion: %s',modulacion));
149 else
150     title(sprintf('\bDiag. Const. de
        la senal OFDM en el Tx \n \rm
        Modulacion: %s ',modulacion));
151 end
152 grid on;
153

```

```

154 %Constelacion recuperada
155 subplot(2,2,2);
156 plot(X_hat(1:500:end),'x','markersize'
        ,3);
157 xlim([-4 4]);
158 ylim([-4 4]);
159 xlabel('Fase');
160 ylabel('Cuadratura');
161 if num_taps>1
162     title(sprintf('\bDiag. Const. de
        la senal OFDM en el Rx\n \rm SNR
        : %.2f dB', Med_SNR));
163 else
164     title(sprintf('\bDiag. Const. de
        la senal OFDM en el Rx\n]]rm SNR:
        %.2f dB',Med_SNR));
165 end
166 grid on;
167
168 %Imagen original
169 subplot(2,2,3);
170 imshow(imagen);
171 title('\bImagen original en el Tx');
172
173 %Imagen recuperada
174 subplot(2,2,4);
175 imshow(rec_im);
176 title(sprintf('\bImagen recuperada
        en el Rx\n\rmBER: %.2g',ber));
177
178 %Impresion del BER
179 disp(['Probabilidad de bit errado (BER
        ): ', num2str(ber) ])

```

Script 1. Código usado para un sistema de comunicaciones con una técnica OFDM

## REFERENCES

- [1] William Stallings, Wireless Communications and Networks, Second Edition, Pearson Prentice Hall, Upper Saddle River, NJ, 2005. ISBN 0-13-191835-4.
- [2] E. Tatayo, "OFDM SOBRE UN CANAL AWGN". C.P. COMUNICACIÓN DIGITAL, Accedido: ago. 25, 2020. [En línea].
- [3] Jordan Street, "OFDM Simulation in Matlab" abril 2015. [En línea]. Disponible: <https://www.youtube.com/watch?v=SyKJrrNhPO8>. [Último acceso: agosto 2020].