

CLASSWORK #2

REDES I

S1-2022

Daniel Felipe Martínez
dmartin40973@universidadean.edu.co

CLASSWORK 2

Activity: Individual

Subject: Toolboxes in MATLAB

Deliverable: Word-File (LastnameName_Classwork2.doc) + M-Files (LastnameName_Classwork2.zip)

Deadline: Mon-28-March (2:00 pm)

According to the Labs done in class, make a detailed documentation for the following:

- 1.[30/100] AppDesigner: Transmitters with Line Coding (BONUS: +5 Bipolar NRZ) (BONUS: +5 MANCHESTER)
- 2.[30/100] Simulink: Modulation & Coding (BONUS +5 NRZ)
- 3.[40/100] AntennaDesigner: Parameters & Pattern

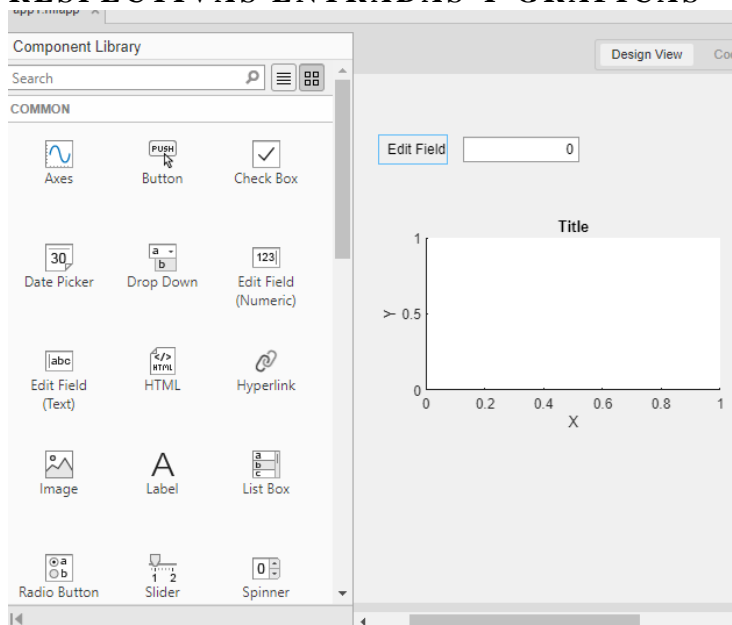
SOLUCIÓN

AppDesigner: Transmitters with Line Coding

PASO 1: ABRIMOS APPS DESIGNER



PASO 2: CREAMOS UN NUEVO PROYECTO Y AGREGAMOS SUS RESPECTIVAS ENTRADAS Y GRAFICAS



TRANSMITTERS

BITS

TB

REGLA UP NRZ CERO

AMPLITUD

REGLA BIT ALTO

REGLA UN NRZ CERO

REGLA POLAR NRZ ALTO

DATOS (bits)

Coding: UN NRZ

Coding: UP NRZ

Coding: POLAR NRZ

PASO 3: UNA VEZ CREADA LA INTERFAZ ASGNAMOS CALLBACK A EL BOTON Y RELACIONAMOS LOS INPUTS Y SUS VARIABLES

```

% CALLBACKS: click, mouse, component events
methods (Access = private)

% Button pushed function: CODIFICARButton
function CODIFICARButtonPushed(app, event)
    clc; format compact
    %----- INPUT -----
    %datos de entrada DATOS (bits)
    bits = app.BITSEditField.Value;
    tb = app.TBEditField.Value;
    regla_bit_alto = app.REGLABITALTOEditField.Value; % valores: 1 v 0
    %REGLAS de codificación
    A = app.AMPLITUDEEditField.Value; %Volts
    regla_UP_NRZ_cero = app.REGLAUPNRZCEROEditField.Value; % valores: 1 v 0
    regla_UN_NRZ_cero = app.REGLAUNNRZCEROEditField.Value; % valores: 1 v 0
    regla_Polar_NRZ_alto = app.REGLAPOLARNRZALTOEditField.Value; % valores: 1 v 0
    %----- PROCESS -----
    %construir cadena bits
    uno = ones(1,tb);
    cero = zeros(1,tb);
    cadena = [];
    cadenaInversa = [];
    polar = [];
    polarInversa = [];
    for n=1:length(bits)
        if(bits(n)=='1')
            cadena = [cadena uno];
            cadenaInversa = [cadenaInversa cero];
            polar = [polar uno];
            polarInversa = [polarInversa -uno];
        else
            cadena = [cadena cero];
            cadenaInversa = [cadenaInversa uno];
            polar = [polar -uno];
            polarInversa = [polarInversa uno];
        end
    end
    %Construcción de los BITS
    if (regla_bit_alto==1)
        bitstream = 5*cadena;
    else
        bitstream = 5*cadenaInversa;
    end
    %codificación Unipolar Positiva NRZ
    if (regla_UP_NRZ_cero == 0)
        UP_NRZ = A*cadena;
    end

```

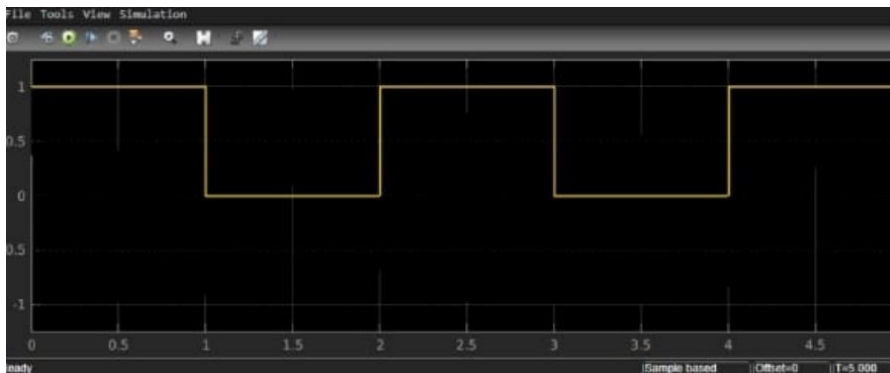
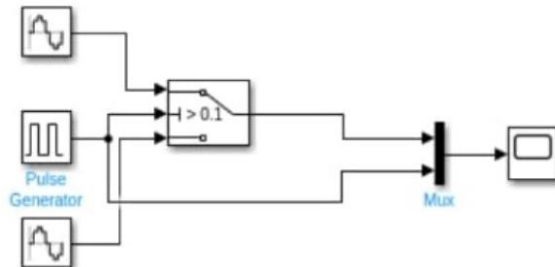
PASO 4: CORREMOS EL PROGRAMA, LLENAMOS LOS CAMPOS CORRESPONDIENTES Y OBSERVAMOS LAS GRAFICAS



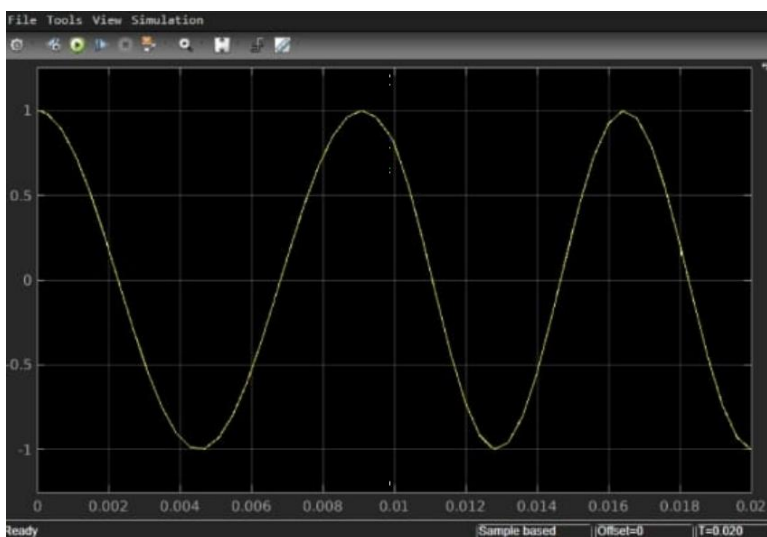
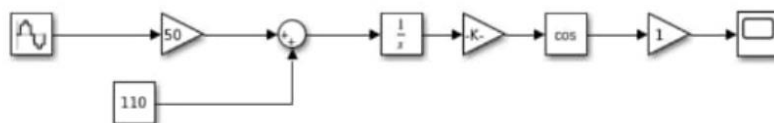
Simulink: Modulation & Coding

SIMULACIÓN EN SIMULINK

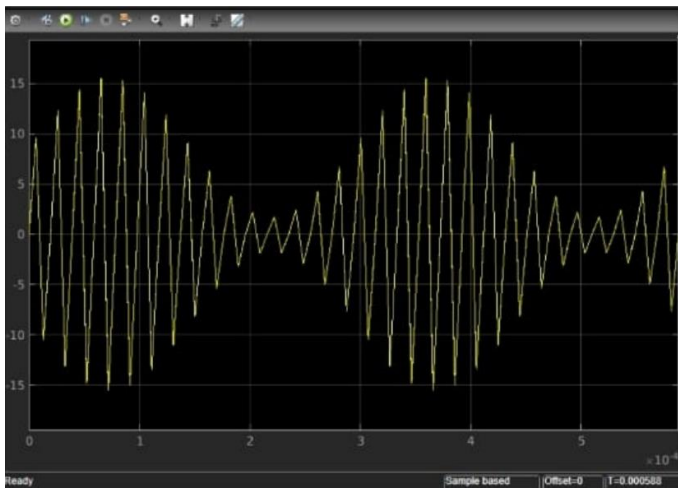
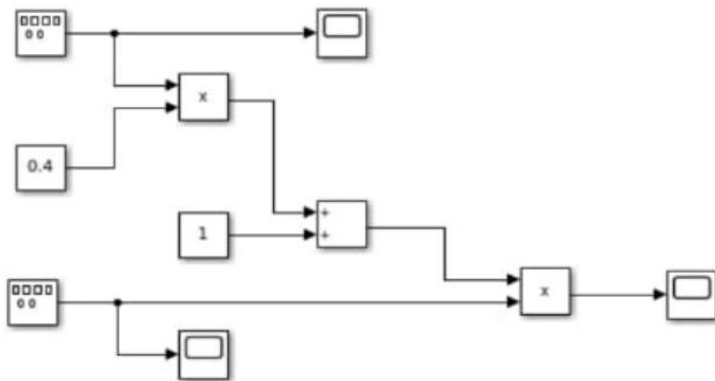
1. MODULACIÓN FSK:



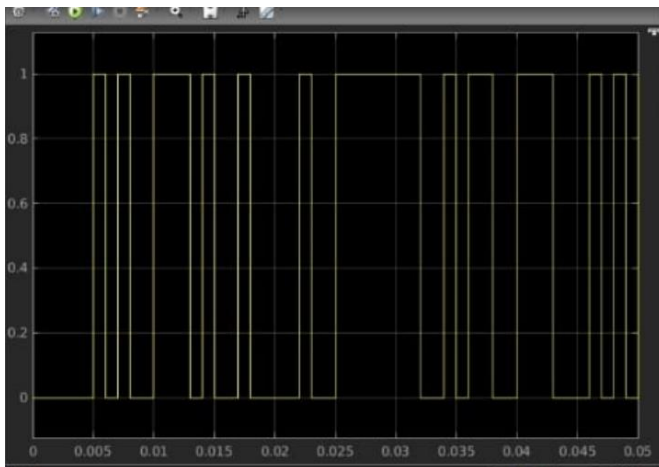
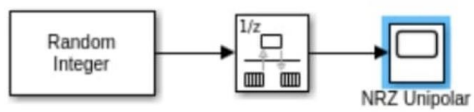
2. MODULACIÓN FM



3. MODULACIÓN AM

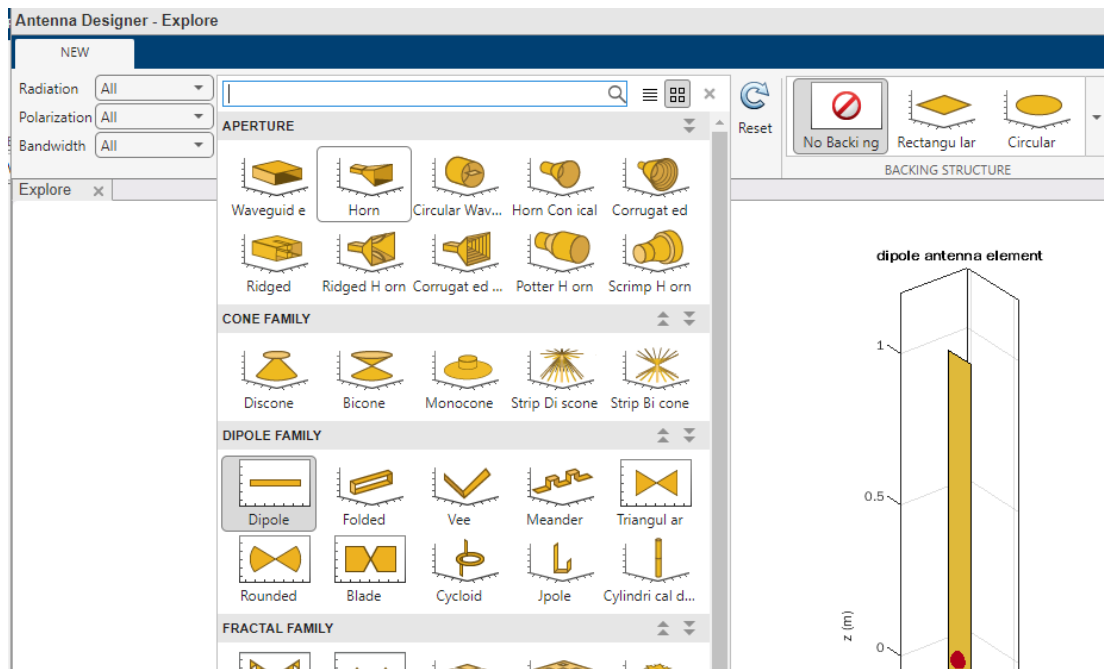


4. CODIFICACIÓN NRZ

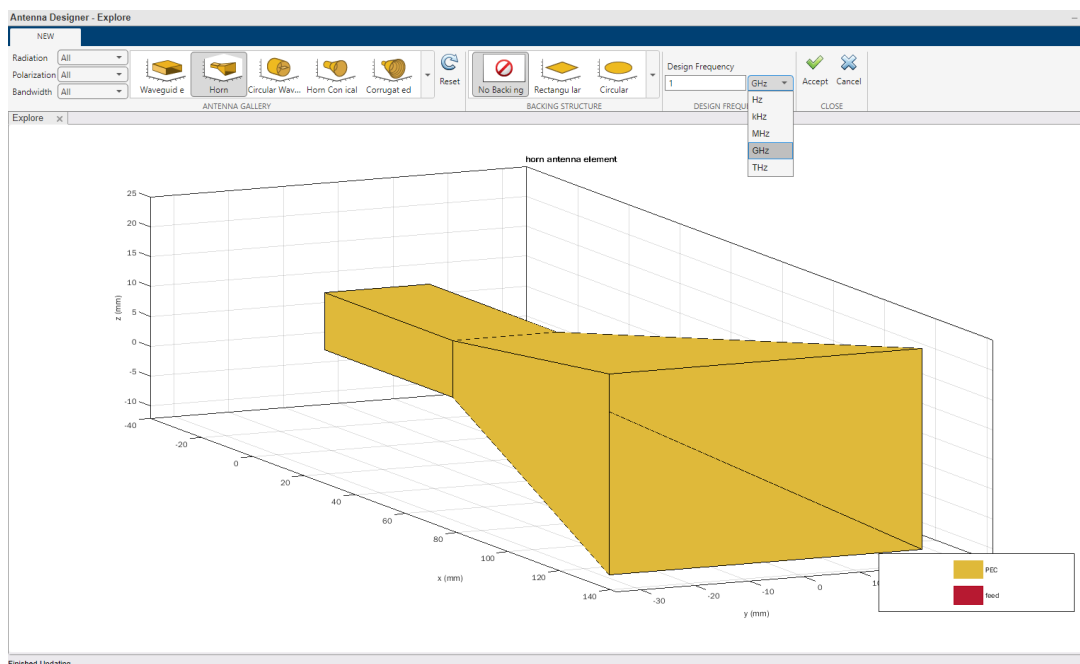


3. AntennaDesigner: Parameters & Pattern

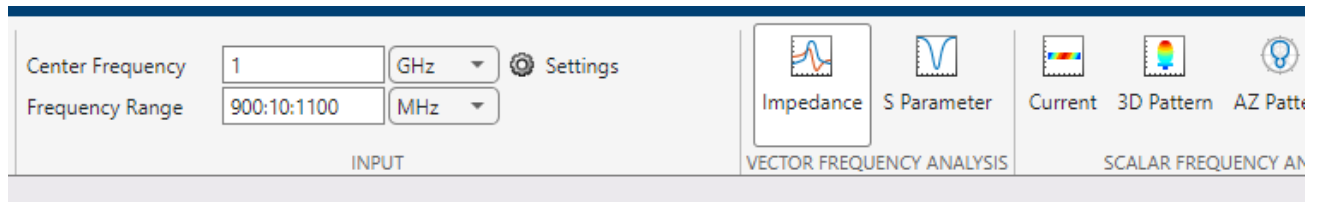
PASO 1: ABRIMOS ANTENNA DESIGNER Y SELECCIONAMOS LA ANTENA “HORN”



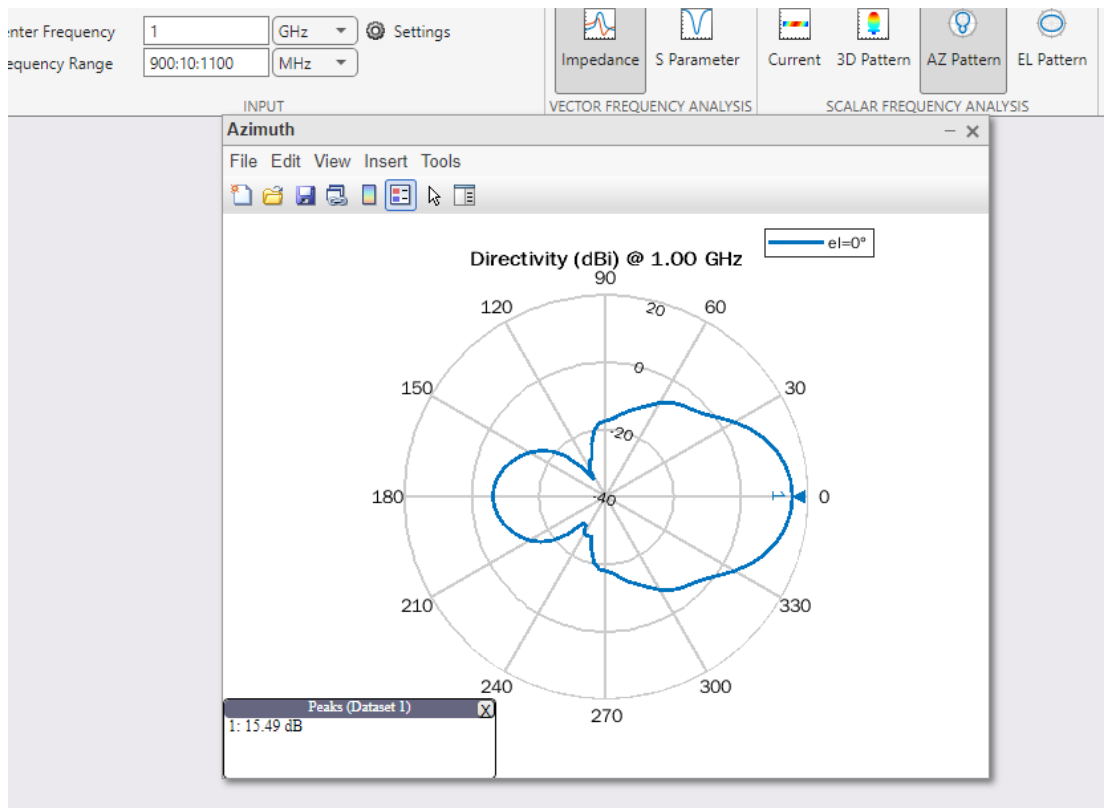
PASO 2: CAMBIAMOS LA FRECUENCIA A 1GHZ Y ACEPTAMOS LOS AJUSTES



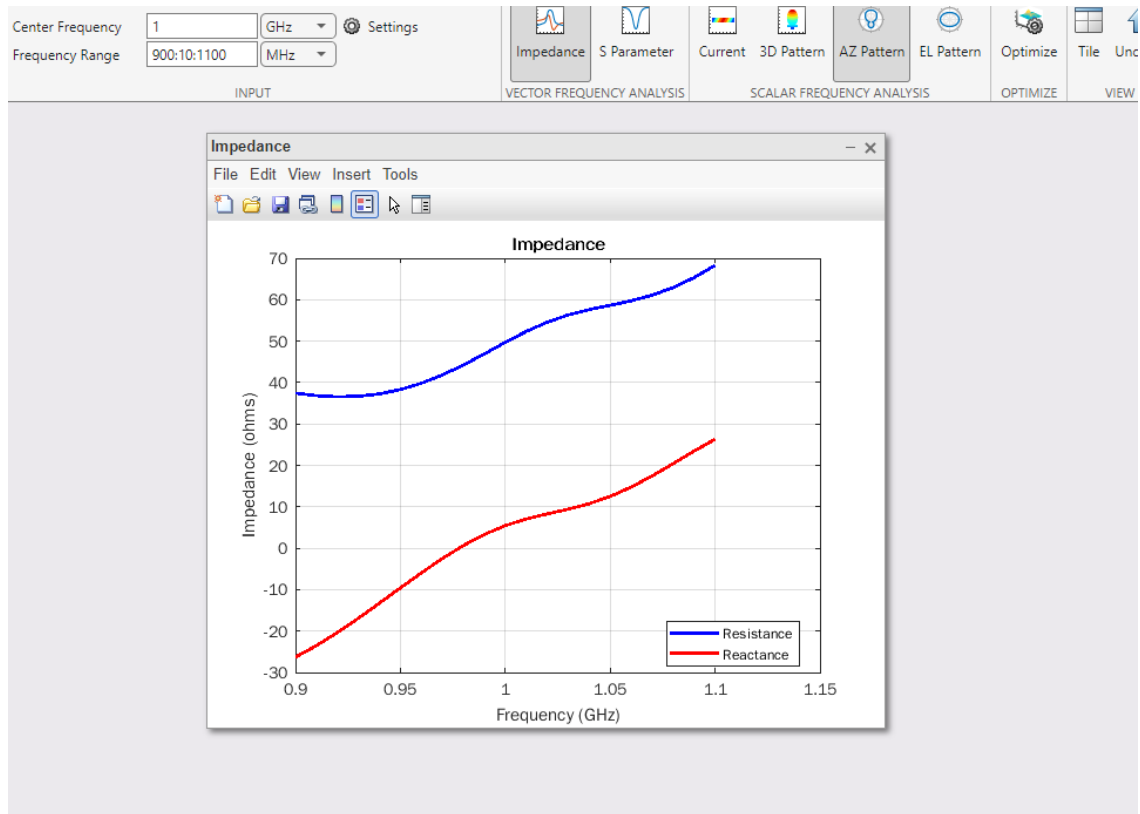
PASO 3: SELECCIONAMOS IMPEDANCIA Y AZ PATTERN



PASO 4: LE DAMOS A UNDOCK PARA QUE NOS MUESTRE EL DIAGRAMA DE RADIACION



PASO CINCO: SELECCIONAMOS SOLO LA IMPEDANCIA Y DE DAMOS EN UNDOCK PARA QUE LA GRAFIQUE



PASO CINCO: SELECCIONAMOS IMPEDANCE Y EL PATTERN PARA QUE NOS MUESTRE LOS NIVELES DE ELEVACIÓN

