```
function varargout = LTEcalc(varargin)
% LTECALC MATLAB code for LTEcalc.fig
%      LTECALC, by itself, creates a new LTECALC or raises the
existing
%      singleton*.
%
%      H = LTECALC returns the handle to a new LTECALC or the
handle to
%      the existing singleton*.
%
%      LTECALC('CALLBACK',hObject,eventData,handles,...) calls the
local
%      function named CALLBACK in LTECALC.M with the given input
arguments.
%
%      LTECALC('Property','Value',...) creates a new LTECALC or
raises the
%      existing singleton*.  Starting from the left, property
value pairs are
%      applied to the GUI before LTEcalc_OpeningFcn gets called.
An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to LTEcalc_OpeningFcn via
varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows
only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help LTEcalc

% Last Modified by GUIDE v2.5 26-Mar-2016 18:19:26

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @LTEcalc_OpeningFcn, ...
                   'gui_OutputFcn',  @LTEcalc_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
```

```
end
% End initialization code - DO NOT EDIT


% --- Executes just before LTEcalc is made visible.
function LTEcalc_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to LTEcalc (see VARARGIN)




                        % set(handles.text6,'string','DL:UL');
                        %
set(handles.text7,'string','DwPTS/GP/UpPTS');
                        % set(handles.text3,'string','El número
total de REs disponibles para datos en DL es:');
                        % set(handles.text8,'string','El número
total de REs disponibles para datos en UL es:');
                        % set(handles.text12,'string','El th para
DL sin code rate es de:');
                        % set(handles.text14,'string','El th para
UL sin code rate es de:');


        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);



% Para figura de circulos concentricos de distancia para rango de
señal

%                     axes (handles.axes4);
%                     t=0:0.001:1;%Initializing time samples
%                     %Transfer chara of sine and cosi.e sin Vs
Cos will give circle
```

```
%                       %Here we are plotting group of sine Vscos
with differnt amplitudes
%                       s15=15*sin(2*pi*t);
%                       c15=15*cos(2*pi*t);
%                       s14=14*sin(2*pi*t);
%                       c14=14*cos(2*pi*t);
%                       s13=13*sin(2*pi*t);
%                       c13=13*cos(2*pi*t);
%                       s12=12*sin(2*pi*t);
%                       c12=12*cos(2*pi*t);
%                       s11=11*sin(2*pi*t);
%                       c11=11*cos(2*pi*t);
%                       s10=10*sin(2*pi*t);
%                       c10=10*cos(2*pi*t);
%                       s9=9*sin(2*pi*t);
%                       c9=9*cos(2*pi*t);
%                       s8=8*sin(2*pi*t);
%                       c8=8*cos(2*pi*t);
%                       s7=7*sin(2*pi*t);
%                       c7=7*cos(2*pi*t);
%                       s6=6*sin(2*pi*t);
%                       c6=6*cos(2*pi*t);
%                       s5=5*sin(2*pi*t);%Sine wave with
amplitude=3unit
%                       c5=5*cos(2*pi*t);%Cosine wave with
amplitude=3unit
%                       s4=4*sin(2*pi*t);
%                       c4=4*cos(2*pi*t);
%                       s3=3*sin(2*pi*t);
%                       c3=3*cos(2*pi*t);
%                       s2=2*sin(2*pi*t);
%                       c2=2*cos(2*pi*t);
%                       s1=1*sin(2*pi*t);
%                       c1=1*cos(2*pi*t);
%                       plot(s1,c1,'-y',s2,c2,'-y',s3,c3,'-
y',s4,c4,'-y',s5,c5,'-y',s6,c6,'-g',s7,c7,'-g',s8,c8,'-g',s9,c9,'-
b',s10,c10,'-b',s11,c11,'-b',s12,c12,'-b',s13,c13,'-b',s14,c14,'-
b',s15,c15,'-b');%Plotting sin Vs Cos
%                       %grid on;%Enable grid lines
%                       axis equal;%Equal width of X and Y axis
%                       axis off



%                               set(gca,'ytick',[]);
%                               set(gca,'yticklabel',[])
%                               set(gca,'xtick',[]);
%                               set(gca,'xticklabel',[])


% Choose default command line output for LTEcalc
handles.output = hObject;
```

```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
axes(handles.axes2);
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
imshow('FI image.jpg');
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
axes(handles.axes3);
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
imshow('UNAM.jpg');


% Update handles structure
guidata(hObject, handles);




% UIWAIT makes LTEcalc wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = LTEcalc_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton11_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

x = get(handles.popupmenu16,'Value');
y = get(handles.popupmenu15,'Value');


% BW = get(handles.popupmenu21,'Value');
%
%             if (BW==1)
```

```
%                 msgbox({'Seleccione un BW de canal'},
'Error','error');
%                 PRB = 0;
%             elseif (BW==2)
%                 set(handles.text102,'String', '6');
%             elseif (BW==3)
%                 set(handles.text102,'String', '15');
%             elseif (BW==4)
%                 set(handles.text102,'String', '25');
%             elseif (BW==5)
%                 set(handles.text102,'String', '50');
%             elseif (BW==6)
%                 set(handles.text102,'String', '75');
%             elseif (BW==7)
%                 set(handles.text102,'String', '100');
%             end
%

PRB = str2num(get(handles.text102,'string'));

tdd = x-1;
ss = y-1;
a = 12*14*PRB;          %OFDM Symbols %%%16800 para 20 MHz;   %a =
subcarriers*ofdm symbols *physical resource blocks = 12*14*100; Se
multiplicara por el Num de subtramas para UL y DL

%Cálculo de Resource Elements de acuerdo
%a PRBs para canal PUCCH

if (PRB==6)
        pucch_res = 12*14*2; %12 subcarr*14 OFDM Sym*2 PRBs
asignados a canal PUCCH

elseif (PRB==15)
        pucch_res = 12*14*4;

elseif (PRB==25)
        pucch_res = 12*14*8;

elseif (PRB==50)
        pucch_res = 12*14*16;

elseif (PRB==75)
        pucch_res = 12*14*24;

elseif (PRB==100)
        pucch_res = 12*14*32;

end

%%%%%%%Se obtiene el índice de modulación por parte del
usuario%%%%
```

```
% % % % % %         %%%%%%      Asociación de Code Rate con índice
de modulación         %%%%%%
cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000 0.4700
0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500 0.5900
0.3900 0.4200 0.4600 0.5100 0.5500 0.6000 0.6400 0.6900 0.7400
0.7800 0.8100 0.8100];

modulationindex = get(handles.popupmenu17, 'value');


            if (modulationindex==1)
                                set(handles.popupmenu18, 'String',
'Seleccione índice de modulación...');
                                m = msgbox({'Indice de modulación
inválido' '' 'Seleccione un índice de modulación'},
'Error','error');

            elseif (modulationindex==2)
                                m = 2;
                                elements = cr(1:10);
                                set(handles.popupmenu18, 'String',
elements);

            elseif (modulationindex==3)
                                m = 4;
                                elements = cr(11:17);
                                set(handles.popupmenu18, 'String',
elements);

            elseif (modulationindex==4)
                                m = 6;
                                elements = cr(18:28);
                                set(handles.popupmenu18, 'String',
elements);
            else

                                m = 0;


            end



% Línea usada para cuando el valor del índice de modulación es
introducido por el usuario en la caja con tag: 'edit1'

%     m = str2num(get(handles.edit6, 'string'));
```

```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % %
if (m~=0)


if ((tdd==0)&&(ss==0))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '3/10/1');
    dwpts=3;
    gp=10;
    uppts=1;
    subul=6; %subtramas uplink
    subdl=2; %subtramas downlink
    ss=2; %special subframes in the frame

%DOWNLINK

    totresdl=a*subdl; %a = 12*14*PRB;
%     pdcchresdl=12*4*subdl; %12*4*subdl  --4 símbolos (max) para
cada subportadora en 12 subportadoras-- %360 OFDM symbols in tdd=0
ss=0
%     rsdl=6*6*subdl;  %reference signal DL -- 1188 para
100PRBs=20MHz
%     sschres=62*2; %secondary syncrhonization channel resources
%     pbchres=12*4*subdl; %physical broadcast channel resources
%     pdsch=totresdl-pdcchresdl-rsdl-sschres-pbchres;

    pdsch = 206*PRB; %206 REs para PDSCH


    set(handles.text4,'string',pdsch);

%UPLINK

%     totresul=(a*subul)+(24*PRB); %total resources uplink =
12*14*subul+(12 REs para UL en cada subtrama especial - son dos
subtramas especiales en el conf index)
%     pucchres=pucch_res*subul; %2688 para 100 PRB = 20MHz
%     dmrsres=11664; %
%     srsres=4032;
%     prachres=3024; %72 subcarriers * 14 OFDM symbols * subtramas
%     pusch=totresul-pucchres-dmrsres-srsres-prachres;

    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB; %24 REs por cada subtrama de UL y por
cada PRB
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text9,'string',pusch);
```

```
    thdl=pdsch*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==0)&&(ss==1))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '9/4/1');

    dwpts=9;
    gp=4;
    uppts=1;
    subul=6;
    subdl=2;
    ss=2;

%     b=24780;%REs por subtrama DL
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss; % 12 *100
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss; %resources special subframe
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%%

%DL
    pdsch=342*PRB;
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',pdsch);
    set(handles.text9,'string',pusch);


    thdl=pdsch*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==0)&&(ss==2))
    set(handles.text1,'String', '2:6');
```

```
    set(handles.text2,'String', '10/3/1');

    dwpts=10;
    gp=3;
    uppts=1;
    subul=6;
    subdl=2;
    ss=2;

%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss;
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;

%DL
    d=366*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==0)&&(ss==3))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '11/2/1');

    dwpts=11;
    gp=2;
    uppts=1;
    subul=6;
    subdl=2;
```

```
    ss=2;
%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss;
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;

%DL
    d=390*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==0)&&(ss==4))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '12/1/1');

    dwpts=12;
    gp=1;
    uppts=1;
    subul=6;
    subdl=2;
    ss=2;
    %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=6*100*ss;
%     c=totresss-pdcchss-psch-rsss;
```

```
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;

%DL
    d=410*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==0)&&(ss==5))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '3/9/2');

    dwpts=3;
    gp=9;
    uppts=2;
    subul=6;
    subdl=2;
    ss=2;
%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     %%%No hay aportación al DL de las subtramas especiales,
ambas se
%     %%%desperdician
%
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;

%DL
    d=206*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
```

```
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==0)&&(ss==6))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '9/3/2');

    dwpts=9;
    gp=3;
    uppts=2;
    subul=6;
    subdl=2;
    ss=2;
%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss;
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;


%DL
    d=342*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
```

```
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==0)&&(ss==7))
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '10/2/2');

    dwpts=10;
    gp=2;
    uppts=2;
    subul=6;
    subdl=2;
    ss=2;
%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss;
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%%
%     pusch=65952;


%DL
    d=366*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==0)&&(ss==8))
```

```
    set(handles.text1,'String', '2:6');
    set(handles.text2,'String', '11/1/2');

    dwpts=11;
    gp=1;
    uppts=2;
    subul=6;
    subdl=2;
    ss=2;
%     %REs en las subtramas de DL es constante
%     b=24780;
%     %%%%%%%%%Cuenta de REs de las subtramas especiales%%%%%%
%     totresss=1200*dwpts*ss;
%     pdcchss=2400*ss;
%     psch=72*ss;
%     rsss=4*100*ss;
%     c=totresss-pdcchss-psch-rsss;
%     %%%%%%%Cuenta total para DL
%     d=b+c;
%     %%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%
%     pusch=65952;

%DL
    d=390*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);




% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % %



elseif ((tdd==1)&&(ss==0))
    set(handles.text1,'String', '4:4');
    set(handles.text2,'String', '3/10/1');
```

```
        dwpts=3;
        gp=10;
        uppts=1;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=482*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==1)&&(ss==1))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
```

```
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=618*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==1)&&(ss==2))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
```

```
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=642*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==1)&&(ss==3))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=666*PRB; %pdsch
```

```
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==1)&&(ss==4))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;


%DL
    d=686*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;
```

```
    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==1)&&(ss==5))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '3/9/2');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=482*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
```

```
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==1)&&(ss==6))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '9/3/2');

       dwpts=9;
        gp=3;
        uppts=2;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=618*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


 elseif ((tdd==1)&&(ss==7))
       set(handles.text1,'String', '4:4');
```

```
       set(handles.text2,'String', '10/2/2');

       dwpts=10;
        gp=2;
        uppts=2;
        subul=4;
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=43776;


%DL
    d=642*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==1)&&(ss==8))
       set(handles.text1,'String', '4:4');
       set(handles.text2,'String', '11/1/2');

       dwpts=11;
        gp=1;
        uppts=2;
        subul=4;
```

```
        subdl=4;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=49980;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%
%         pusch=43776;

%DL
    d=666*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % %


elseif ((tdd==2)&&(ss==0))
    set(handles.text1,'String', '6:2');
    set(handles.text2,'String', '3/10/1');

        dwpts=3;
        gp=10;
        uppts=1;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
```

```
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=758*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==2)&&(ss==1))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
```

```
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=894*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);



    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==2)&&(ss==2))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=918*PRB; %pdsch
```

```
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==2)&&(ss==3))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=942*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;
```

```
    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==2)&&(ss==4))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=962*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
```

```
    set(handles.text15,'string',thul);


elseif ((tdd==2)&&(ss==5))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '3/9/2');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=758*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==2)&&(ss==6))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '9/3/2');
```

```
       dwpts=9;
        gp=3;
        uppts=2;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=894*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


 elseif ((tdd==2)&&(ss==7))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '10/2/2');

       dwpts=10;
        gp=2;
        uppts=2;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
```

```
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=918*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==2)&&(ss==8))
       set(handles.text1,'String', '6:2');
       set(handles.text2,'String', '11/1/2');

       dwpts=11;
        gp=1;
        uppts=2;
        subul=2;
        subdl=6;
        ss=2;
%         %REs en las subtramas de DL es constante
%          b=75180;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
```

```
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=942*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*2*PRB);
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % %


elseif ((tdd==3)&&(ss==0))
    set(handles.text1,'String', '6:3');
    set(handles.text2,'String', '3/10/1');

        dwpts=3;
        gp=10;
        uppts=1;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%          b=76308;
%          %%%%%%%%%%Cuenta de REs de las subtramas 
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%
```

```
%         pusch=31680;

%DL
    d=746*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolo y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==3)&&(ss==1))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=814*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolo y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
```

```
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==3)&&(ss==2))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=826*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolo y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
```

```
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==3)&&(ss==3))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=838*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolo y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);
```

```
elseif ((tdd==3)&&(ss==4))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;


%DL
    d=848*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolo y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==3)&&(ss==5))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '3/9/2');
```

```
        dwpts=3;
        gp=9;
        uppts=2;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=12000*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=746*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==3)&&(ss==6))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '9/3/2');

       dwpts=9;
        gp=3;
        uppts=2;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
```

```
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=814*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


 elseif ((tdd==3)&&(ss==7))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '10/2/2');

       dwpts=10;
        gp=2;
        uppts=2;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=76308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
```

```
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;

%DL
    d=826*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==3)&&(ss==8))
       set(handles.text1,'String', '6:3');
       set(handles.text2,'String', '11/1/2');

       dwpts=11;
        gp=1;
        uppts=2;
        subul=3;
        subdl=6;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=31680;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=31680;
```

```
%DL
    d=838*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);




% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %



elseif ((tdd==4)&&(ss==0))
    set(handles.text1,'String', '7:2');
    set(handles.text2,'String', '3/10/1');

        dwpts=3;
        gp=10;
        uppts=1;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss; %1200 para BW=20MHz--> 100PRBs -
-> 100 *12
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;
```

```
%DL
    d=884*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==4)&&(ss==1))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=952*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
```

```
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;



    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==4)&&(ss==2))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=964*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;
```

```
    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==4)&&(ss==3))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=976*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
```

```
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==4)&&(ss==4))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=986*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;



    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);
```

```
elseif ((tdd==4)&&(ss==5))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '3/9/2');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=884*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==4)&&(ss==6))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '9/3/2');

       dwpts=9;
        gp=3;
```

```
        uppts=2;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=952*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


 elseif ((tdd==4)&&(ss==7))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '10/2/2');

       dwpts=10;
        gp=2;
        uppts=2;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
```

```
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=964*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==4)&&(ss==8))
       set(handles.text1,'String', '7:2');
       set(handles.text2,'String', '11/1/2');

       dwpts=11;
        gp=1;
        uppts=2;
        subul=2;
        subdl=7;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=88908;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
```

```
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%
%         pusch=20448;

%DL
    d=976*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);




% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % %




elseif ((tdd==5)&&(ss==0))
    set(handles.text1,'String', '8:1');
    set(handles.text2,'String', '3/10/1');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=1;
        subdl=8;
        ss=1;
        %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
```

```
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;

%DL
    d=1022*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==5)&&(ss==1))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=16068;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;
```

```
%DL
    d=1090*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==5)&&(ss==2))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;

%DL
    d=1102*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
```

```
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==5)&&(ss==3))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;


%DL
    d=1114*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
```

```
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==5)&&(ss==4))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;

%DL
    d=1124*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*PRB); %12 REs (1 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;



    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;
```

```
    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==5)&&(ss==5))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '3/9/2');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;

%DL
    d=1022*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==5)&&(ss==6))
```

```
        set(handles.text1,'String', '8:1');
        set(handles.text2,'String', '9/3/2');

        dwpts=9;
         gp=3;
         uppts=2;
         subul=1;
         subdl=8;
         ss=1;
%          %REs en las subtramas de DL es constante
%          b=101508;
%          %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%          totresss=1200*dwpts*ss;
%          pdcchss=2400*ss;
%          psch=72*ss;
%          rsss=4*100*ss;
%          c=totresss-pdcchss-psch-rsss;
%          %%%%%%%%Cuenta total para DL
%          d=b+c;
%          %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%          pusch=10224;


%DL
    d=1090*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


 elseif ((tdd==5)&&(ss==7))
        set(handles.text1,'String', '8:1');
        set(handles.text2,'String', '10/2/2');

        dwpts=10;
         gp=2;
         uppts=2;
```

```
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=10224;


%DL
    d=1102*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==5)&&(ss==8))
       set(handles.text1,'String', '8:1');
       set(handles.text2,'String', '11/1/2');

        dwpts=11;
        gp=1;
        uppts=2;
        subul=1;
        subdl=8;
        ss=1;
%         %REs en las subtramas de DL es constante
%         b=101508;
```

```
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%
%         pusch=10224;

%DL
    d=1124*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % %





elseif ((tdd==6)&&(ss==0))
    set(handles.text1,'String', '3:5');
    set(handles.text2,'String', '3/10/1');

        dwpts=3;
        gp=10;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
```

```
%         %REs en las subtramas de DL es constante
%         b=37380;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;


%DL
    d=344*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==6)&&(ss==1))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '9/4/1');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37380;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
```

```
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;

%DL
    d=480*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==6)&&(ss==2))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '10/3/1');

        dwpts=10;
        gp=3;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
```

```
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;

%DL
    d=504*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);

elseif ((tdd==6)&&(ss==3))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '11/2/1');

        dwpts=11;
        gp=2;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;

%DL
    d=528*PRB; %pdsch
%UL
```

```
    totresul=(a*subul)+(12*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==6)&&(ss==4))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '12/1/1');

       dwpts=12;
        gp=1;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=6*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;


%DL
    d=548*PRB; %pdsch
%UL
    totresul=(a*subul)+(12*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
```

```
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==6)&&(ss==5))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '3/9/2');

        dwpts=3;
        gp=9;
        uppts=2;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=12000*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;


%DL
    d=344*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);
```

```
    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);


elseif ((tdd==6)&&(ss==6))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '9/3/2');

       dwpts=9;
        gp=4;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;


%DL
    d=480*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);
```

```
 elseif ((tdd==6)&&(ss==7))
       set(handles.text1,'String', '3:5');
       set(handles.text2,'String', '10/2/2');

       dwpts=10;
        gp=2;
        uppts=2;
        subul=5;
        subdl=3;
        ss=2;
%         %REs en las subtramas de DL es constante
%         b=37308;
%         %%%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%         totresss=1200*dwpts*ss;
%         pdcchss=2400*ss;
%         psch=72*ss;
%         rsss=4*100*ss;
%         c=totresss-pdcchss-psch-rsss;
%         %%%%%%%%Cuenta total para DL
%         d=b+c;
%         %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%%
%         pusch=53856;


%DL
    d=504*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;


    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



elseif ((tdd==6)&&(ss==8))
       set(handles.text1,'String', '3:5');
```

```
       set(handles.text2,'String', '11/1/2');

        dwpts=9;
        gp=4;
        uppts=1;
        subul=5;
        subdl=3;
        ss=2;
%          %REs en las subtramas de DL es constante
%          b=37308;
%          %%%%%%%%%Cuenta de REs de las subtramas
especiales%%%%%%%
%          totresss=1200*dwpts*ss;
%          pdcchss=2400*ss;
%          psch=72*ss;
%          rsss=4*100*ss;
%          c=totresss-pdcchss-psch-rsss;
%          %%%%%%%%Cuenta total para DL
%          d=b+c;
%          %%%%%%%%%Cuenta de REs para UL%%%%%%%%%%%%%%%%%%%
%          pusch=53856;


%DL
    d=528*PRB; %pdsch
%UL
    totresul=(a*subul)+(24*ss*PRB); %24 REs (2 símbolos y 12
subportadoras en un PRB; sólo hay una Subtrama especial - UpPTS)
    pucch = pucch_res*subul;
    dmrsres = 24*subul*PRB;
    srs = 12*subul*PRB;
    pusch = totresul-pucch-dmrsres-srs;

    set(handles.text4,'string',d);
    set(handles.text9,'string',pusch);


    thdl=d*m*100;
    thul=pusch*m*100;

    set(handles.text13,'string',thdl);
    set(handles.text15,'string',thul);



% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% % % % % % % % % % % % % % % % % % % % % % % % % % %
```

```
end


end


% --- Executes on selection change in tddindex.
function tddindex_Callback(hObject, eventdata, handles)
% hObject    handle to tddindex (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
tddindex contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
tddindex


% --- Executes during object creation, after setting all
properties.
function tddindex_CreateFcn(hObject, eventdata, handles)
% hObject    handle to tddindex (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in ssindex.
function ssindex_Callback(hObject, eventdata, handles)
% hObject    handle to ssindex (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns ssindex
contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
ssindex
```

```
% --- Executes during object creation, after setting all
properties.
function ssindex_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ssindex (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of
edit1 as a double


% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% % --- Executes on button press in pushbutton5.
% function pushbutton5_Callback(hObject, eventdata, handles)
% % hObject    handle to pushbutton5 (see GCBO)
% % eventdata  reserved - to be defined in a future version of
MATLAB
```

```
% % handles    structure with handles and user data (see GUIDATA)


% --- Executes on selection change in popupmenu14.
function popupmenu14_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu14 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

codec = get(hObject, 'value');


thdl = str2num(get(handles.text13,'string'));
thul = str2num(get(handles.text15,'string'));

switch codec


    case 3

        set(handles.text68,'string','96 Kbps'); % 96 kbps =
(80+40[Bytes])*8[bits]*100[pkt/s]/1000[ms]
        set(handles.text70,'string','105.6 Kbps'); %105.6 kbps
=(80+52[Bytes])*8[bits]*100[pkt/s]/1000[ms]

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2   0.22;4 2
0.28;5 2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4
0.34;12 4 0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

%         coderate = str2num(get(handles.edit5, 'string'));
%
%         %%%%%Cálculo del th con el code rate%%%%%%
%
%         thulcr=(thul*coderate)/1000000;
```

```
%         set(handles.text74,'string',thulcr);
%
%         thdlcr=(thdl*coderate)/1000000;
%         set(handles.text76,'string',thdlcr);
%
%
%         set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 105.6 Kbps por llamada');
%
%
%
%
%        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%
%
%         voip1=thulcr/0.1056;
%         calls1=floor(voip1);
%
%         voip2=thdlcr/0.1056;
%         calls2=floor(voip2);
%
%         disp('El número máximo de llamadas admitidas para UL con
G711 es:  ');
%         disp(calls1);
%         disp('El número máximo de llamadas admitidas para DL con
G711 es:  ');
%         disp(calls2);
%         ague=1;
%



    case 4

        set(handles.text68,'string','40 Kbps');
        set(handles.text70,'string','49.6 Kbps');


%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];
```

```
        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);




    case 5

        set(handles.text68,'string','10.37 Kbps');
        set(handles.text70,'string','12.3 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 6

        set(handles.text68,'string','16 Kbps');
        set(handles.text70,'string','19.2 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];
```

```
        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 7

        set(handles.text68,'string','104 Kbps');
        set(handles.text70,'string','123.2 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 8

        set(handles.text68,'string','88 Kbps');
        set(handles.text70,'string','107.2 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];
```

```
        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 9

        set(handles.text68,'string','80 Kbps');
        set(handles.text70,'string','99.2 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 10

        set(handles.text68,'string','28.2 Kbps');
        set(handles.text70,'string','33 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];
```

```
        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 11

        set(handles.text68,'string','44.85 Kbps');
        set(handles.text70,'string','54.5 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);



    case 14

        set(handles.text68,'string','20.8 Kbps');
        set(handles.text70,'string','25.6 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];
```

```
        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 15

        set(handles.text68,'string','21.2 Kbps');
        set(handles.text70,'string','26 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 16

        set(handles.text68,'string','21.9 Kbps');
        set(handles.text70,'string','26.7 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
```

```
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 17

        set(handles.text68,'string','22.7 Kbps');
        set(handles.text70,'string','27.5 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 18

        set(handles.text68,'string','23.4 Kbps');
        set(handles.text70,'string','28.2 Kbps');
```

```
%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 19

        set(handles.text68,'string','24 Kbps');
        set(handles.text70,'string','28.8 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 20

        set(handles.text68,'string','26.2 Kbps');
        set(handles.text70,'string','31 Kbps');
```

```
%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 21

        set(handles.text68,'string','28.2 Kbps');
        set(handles.text70,'string','33 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 24

        set(handles.text68,'string','22.6 Kbps');
```

```
        set(handles.text70,'string','27.4 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);



    case 25

        set(handles.text68,'string','24.9 Kbps');
        set(handles.text70,'string','29.7 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);
```

```
    case 26

        set(handles.text68,'string','28.7 Kbps');
        set(handles.text70,'string','33.5 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 27

        set(handles.text68,'string','30.3 Kbps');
        set(handles.text70,'string','35.1 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);
```

```
    case 28

        set(handles.text68,'string','31.9 Kbps');
        set(handles.text70,'string','36.7 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 29

        set(handles.text68,'string','35.9 Kbps');
        set(handles.text70,'string','40.7 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];
```

```
        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    case 30

        set(handles.text68,'string','39.1 Kbps');
        set(handles.text70,'string','43.9 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];

        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);

    case 31

        set(handles.text68,'string','39.9 Kbps');
        set(handles.text70,'string','44.7 Kbps');

%         codrate=[0 2 0.11;1 2 0.14;2 2 0.17;3 2 0.22;4 2 0.28;5
2 0.34; 6 2 0.4;7 2 0.47;8 2 0.54;9 2 0.61;10 4 0.3;11 4 0.34;12 4
0.39;13 4 0.44;14 4 0.5;15 4 0.55;
%         16 4 0.59;17 6 0.39;18 6 0.42;19 6 0.46;20 6 0.51;21 6
0.55;22 6 0.6;23 6 0.64;24 6 0.69;25 6 0.74;26 6 0.78;27 6 0.81;28
6 0.81];

        MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];

        m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6 6
6 6];

        cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];
```

```
        tabla = [MCS' m' cr'];
        set(handles.uitable1, 'data', tabla);


    otherwise


        set(handles.text68,'string','');
        set(handles.text70,'string','');

end



% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu14 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu14


% --- Executes during object creation, after setting all
properties.
function popupmenu14_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu14 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit5 as text
%        str2double(get(hObject,'String')) returns contents of
edit5 as a double
```

```
% --- Executes during object creation, after setting all
properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton8 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
thdl = str2num(get(handles.text13,'string'));
thul = str2num(get(handles.text15,'string'));

codec = get(handles.popupmenu14, 'value');
coderate = str2num(get(handles.text89, 'string'));




% ifexist = exist('coderate','var');

%

% if (ifexist==0)

%     msgbox('Invalid Value', 'Error','error');

%

% elseif (ifexist==1)



    if (codec==3)


     datrate=0.1056;

        %%%%%Cálculo del throughput con el code rate%%%%%%
```

```
        thulcr=(thul*coderate)/1000000;  %Throughput UL con
CodeRate entre 1x10^6 (Mb)
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;     %Throughput DL con
CodeRate
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 105.6 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/0.1056; %105.6 kbps data rate PHY ([105.6
kbps/1x10^6] = .1056 Mbps)
        calls1=floor(voip1);

        voip2=thdlcr/0.1056;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G711:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G711:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1; %Aggregate UE


         %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+20
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
%              x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+20
            for q=p:ague:calls2

                 y=datrate*p;
```

```
                %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+20;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=0.1056*calls1;
        vec0=d*c;
        vec1=0.1056:0.1056*ague:0.1056*calls1;
        vec2=1:ague:calls1+20;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%       plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+20;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=0.1056*calls2;
        vec02=d2*c2;
        vec12=0.1056:0.1056*ague:0.1056*calls2;
        vec22=1:ague:calls2+20;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ','Linewidth', 10);

        graph = plot(vec22,vec32,'-b',vec2,vec3,'-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G711)', 'Th UpLink (G711)');
```

```
        return



elseif(codec==4)


      datrate=0.0496;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 49.6 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G729
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G729 (8Kbps):');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G729 (8Kbps):');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;


        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
```

```
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %disp(['Trhoughput Mbbps: ',num2str(x)]);

    end

    for p=1:ague:calls2+8
        for q=p:ague:calls2

            y=datrate*p;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        end
        %disp(['Numero de UEs:',num2str(p)] );
        y=y*1;
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %disp(['Trhoughput bps: ',num2str(y)]);

    end

    %%%GRAFICA PARA UL %%%
    a=calls1+8;
    b=(a-calls1)/ague;
    c=ones(1,b);
    d=datrate*calls1;
    vec0=d*c;
    vec1=datrate:datrate*ague:datrate*calls1;
    vec2=1:ague:calls1+8;
    vec3=[vec1 vec0];

    axes(handles.axes1);

%     plot(vec2,vec3,':r')
    grid on;
    hold on

    %%%GRAFICA PARA EL DOWNLINK %%%

    a2=calls2+8;
    b2=(a2-calls2)/ague;
    c2=ones(1,b2);
    d2=datrate*calls2;
    vec02=d2*c2;
    vec12=datrate:datrate*ague:datrate*calls2;
    vec22=1:ague:calls2+8;
    vec32=[vec12 vec02];

    axes(handles.axes1);

%     plot(vec22,vec32,':b')
%     grid on;
%     xlabel('UE');
%     ylabel('Throughput Mbps');
%     title('Throughput teórico vs UE ');
```

```
%         legend('Th UpLink (G729 - 8kbps)','Th DownLink (G729 -
8kbps)');


        graph = plot(vec22,vec32,':b',vec2,vec3,':r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G729 - 8kbps)', 'Th UpLink
(G729 - 8kbps)');

        return



elseif(codec==5)

    datrate=0.02226667;


        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 22.26 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G729
%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G723.1 (6.4Kbps):');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G723.1 (6.4Kbps):');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);
```

```
        ague=1;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+8
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+8;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+8;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-.r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+8;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
```

```
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+8;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-.b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink (G723.1 - 6.4Kbps)','Th DownLink
(G723.1 - 6.4Kbps)');

        graph = plot(vec22,vec32,'-.b',vec2,vec3,'-.r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G723.1 - 6.4Kbps)', 'Th UpLink
(G723.1 - 6.4Kbps)');


        return


elseif(codec==6)

    datrate=0.0192;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 19.2 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);
```

```
        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G723.1 (5.3Kbps):');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G723.1 (5.3Kbps):');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+8
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+8;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+8;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'--r')
        grid on;
```

```
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+8;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+8;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'--b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink (G723.1 - 5.3Kbps)','Th DownLink
(G723.1 - 5.3Kbps)');


        graph = plot(vec22,vec32,'--b',vec2,vec3,'--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G723.1 - 5.3Kbps)', 'Th UpLink
(G723.1 - 5.3Kbps)');

        return



elseif(codec==7)

    datrate=0.1152;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 115.2 Kbps por llamada');
```

```
        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G726:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G726:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;


        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+8
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+8;
        b=(a-calls1)/ague;
```

```
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+8;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'x-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+8;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+8;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'x-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink (G726)','Th DownLink (G726)');

        graph = plot(vec22,vec32,'x-b',vec2,vec3,'x-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G726)', 'Th UpLink (G726)');

        return


elseif(codec==8)

    datrate=0.1072;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
```

```
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 107.2 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G729 (24Kbps):');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G729 (24Kbps):');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;


        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+8
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+8;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+8;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'x:r')
        grid on;
        hold on

        %%%GRAFICZA PARA EL DOWNLINK %%%

        a2=calls2+8;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+8;
        vec32=[vec12 vec02];


        axes(handles.axes1);

%         plot(vec22,vec32,'x:b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink (G729-24Kbps)','Th DownLink (G729-
24Kbps)');


        graph = plot(vec22,vec32,'x:b',vec2,vec3,'x:r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G729-24Kbps)', 'Th UpLink
(G729-24Kbps)');
```

```
        return


elseif(codec==9)

    datrate=0.0992;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 99.2 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G728
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G728:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G728:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;


        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);
```

```
end

for p=1:ague:calls2+8
    for q=p:ague:calls2

         y=datrate*p;
         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    end
    %disp(['Numero de UEs:',num2str(p)] );
    y=y*1;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %disp(['Trhoughput bps: ',num2str(y)]);

end

%%%GRAFICA PARA UL %%%
a=calls1+8;
b=(a-calls1)/ague;
c=ones(1,b);
d=datrate*calls1;
vec0=d*c;
vec1=datrate:datrate*ague:datrate*calls1;
vec2=1:ague:calls1+8;
vec3=[vec1 vec0];

axes(handles.axes1);

%plot(vec2,vec3,'+-r');
grid on;
hold on




%%%GRAFICZA PARA EL DOWNLINK %%%

a2=calls2+8;
b2=(a2-calls2)/ague;
c2=ones(1,b2);
d2=datrate*calls2;
vec02=d2*c2;
vec12=datrate:datrate*ague:datrate*calls2;
vec22=1:ague:calls2+8;
vec32=[vec12 vec02];

axes(handles.axes1);

graph = plot(vec22,vec32,'+-b',vec2,vec3,'+-r');
grid on;


xlabel('UE');
```

```
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G728)', 'Th UpLink (G728)');



        return


elseif(codec==10)

    datrate=0.0332;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 33.2 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=1;


        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+8
               for j=i:ague:calls1
                   x=datrate*i;
               end
```

```
            %disp(['Numero de UEs:',num2str(i)] );
            x=x*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+8
            for q=p:ague:calls2

                y=datrate*p;
                %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+8;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+8;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'*-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+8;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+8;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'*-b')
%         grid on;
%         xlabel('UE');
```

```
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink (AMR)','Th DownLink (AMR)');


        graph = plot(vec22,vec32,'*-b',vec2,vec3,'*-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR)', 'Th UpLink (AMR)');

        return


elseif(codec==11)

      datrate=0.0545;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 54.5 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con G722.2:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con G722.2:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;
```

```
        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
```

```
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'s-b',vec2,vec3,'s-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (G722.2)', 'Th UpLink
(G722.2)');

        return

elseif(codec==14)

      datrate=0.0256;

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 25.6 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 4.75:');
```

```
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 4.75:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%
```

```
        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'o--b',vec2,vec3,'o--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 4.75)', 'Th UpLink (AMR
4.75)');

        return

elseif(codec==15)

      datrate=0.026; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 26 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);
```

```
        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 5.15:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 5.15:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];
```

```
        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'d-b',vec2,vec3,'d-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 5.15)', 'Th UpLink (AMR
5.15)');

        return

elseif(codec==16)

      datrate=0.0267; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);
```

```
        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 26.7 Kbps por llamada');

        %%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 5.90:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 5.90:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
```

```
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');



        graph = plot(vec22,vec32,'d--b',vec2,vec3,'d--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 5.90)', 'Th UpLink (AMR
5.90)');

        return

elseif(codec==17)

      datrate=0.0275; %valor de Data rate en capa PHY
```

```
        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 27.5 Kbps por llamada');

        %%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 6.70:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 6.70:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
```

```
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'v-b',vec2,vec3,'v-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 6.70)', 'Th UpLink (AMR
6.70)');
```

```
        return

elseif(codec==18)

      datrate=0.0282; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 28.2 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 7.40:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 7.40:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end
```

```
for p=1:ague:calls2+80
    for q=p:ague:calls2

         y=datrate*p;
         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    end
    %disp(['Numero de UEs:',num2str(p)] );
    y=y*1;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %disp(['Trhoughput bps: ',num2str(y)]);

end

%%%GRAFICA PARA UL %%%
a=calls1+80;
b=(a-calls1)/ague;
c=ones(1,b);
d=datrate*calls1;
vec0=d*c;
vec1=datrate:datrate*ague:datrate*calls1;
vec2=1:ague:calls1+80;
vec3=[vec1 vec0];

axes(handles.axes1);

%   plot(vec2,vec3,'-r')
grid on;
hold on

%%%GRAFICA PARA EL DOWNLINK %%%

a2=calls2+80;
b2=(a2-calls2)/ague;
c2=ones(1,b2);
d2=datrate*calls2;
vec02=d2*c2;
vec12=datrate:datrate*ague:datrate*calls2;
vec22=1:ague:calls2+80;
vec32=[vec12 vec02];

axes(handles.axes1);

%   plot(vec22,vec32,'-b')
%   grid on;
%   xlabel('UE');
%   ylabel('Throughput Mbps');
%   title('Throughput teórico vs UE ');
%   legend('Th UpLink','Th DownLink');


graph = plot(vec22,vec32,'v--b',vec2,vec3,'v--r');
grid on;
```

```
        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 7.40)', 'Th UpLink (AMR
7.40)');

        return

elseif(codec==19)

      datrate=0.0288; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 28.8 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 7.95:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 7.95:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
```

```
     %disp(['Numero de UEs:',num2str(i)] );
     x=x*1;
     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
     %disp(['Trhoughput Mbbps: ',num2str(x)]);

end

for p=1:ague:calls2+80
    for q=p:ague:calls2

         y=datrate*p;
         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    end
    %disp(['Numero de UEs:',num2str(p)] );
    y=y*1;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %disp(['Trhoughput bps: ',num2str(y)]);

end

%%%GRAFICA PARA UL %%%
a=calls1+80;
b=(a-calls1)/ague;
c=ones(1,b);
d=datrate*calls1;
vec0=d*c;
vec1=datrate:datrate*ague:datrate*calls1;
vec2=1:ague:calls1+80;
vec3=[vec1 vec0];

axes(handles.axes1);

plot(vec2,vec3,'-r')
grid on;
hold on

%%%GRAFICA PARA EL DOWNLINK %%%

a2=calls2+80;
b2=(a2-calls2)/ague;
c2=ones(1,b2);
d2=datrate*calls2;
vec02=d2*c2;
vec12=datrate:datrate*ague:datrate*calls2;
vec22=1:ague:calls2+80;
vec32=[vec12 vec02];

axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
```

```
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'v:b',vec2,vec3,'v:r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 7.95)', 'Th UpLink (AMR
7.95)');

        return

elseif(codec==20)

      datrate=0.031; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 31 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 10.20:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 10.20:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;
```

```
        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
```

```
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'^-b',vec2,vec3,'^-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 10.20)', 'Th UpLink (AMR
10.20)');


        return

elseif(codec==21)

      datrate=0.033; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 33 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 12.20:');
```

```
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 12.20:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%
```

```
        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');

        graph = plot(vec22,vec32,'^--b',vec2,vec3,'^--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 12.20)', 'Th UpLink (AMR
12.20)');


        return

elseif(codec==24)

      datrate=0.0274; %valor de Data rate en capa PHY %%% text70

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 27.4 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);
```

```
        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 6.60:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 6.60:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];
```

```
        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'<-b',vec2,vec3,'<-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 6.60)', 'Th UpLink (AMR
6.60)');

        return

elseif(codec==25)

      datrate=0.0297; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);
```

```
        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 29.7 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 8.85:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 8.85:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
```

```
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');



        graph = plot(vec22,vec32,'<--b',vec2,vec3,'<--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 8.85)', 'Th UpLink (AMR
8.85)');


        return


elseif(codec==26)
```

```
      datrate=0.0335; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 33.5 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 12.65:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 12.65:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'>-b',vec2,vec3,'>-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
```

```
        legend(graph,'Th DownLink (AMR 12.65)', 'Th UpLink (AMR
12.65)');

        return

elseif(codec==27)

      datrate=0.0351; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 35.1 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 14.25:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 14.25:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);
```

```
        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');

        graph = plot(vec22,vec32,'>--b',vec2,vec3,'>--r');
```

```
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 14.25)', 'Th UpLink (AMR
14.25)');


        return

elseif(codec==28)

      datrate=0.0367; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 36.7 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 15.85:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 15.85:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
```

```
            x=datrate*i;
          end
         %disp(['Numero de UEs:',num2str(i)] );
         x=x*1;
         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
         %disp(['Trhoughput Mbbps: ',num2str(x)]);

    end

    for p=1:ague:calls2+80
        for q=p:ague:calls2

             y=datrate*p;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        end
        %disp(['Numero de UEs:',num2str(p)] );
        y=y*1;
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %disp(['Trhoughput bps: ',num2str(y)]);

    end

    %%%GRAFICA PARA UL %%%
    a=calls1+80;
    b=(a-calls1)/ague;
    c=ones(1,b);
    d=datrate*calls1;
    vec0=d*c;
    vec1=datrate:datrate*ague:datrate*calls1;
    vec2=1:ague:calls1+80;
    vec3=[vec1 vec0];

    axes(handles.axes1);

%     plot(vec2,vec3,'-r')
    grid on;
    hold on

    %%%GRAFICA PARA EL DOWNLINK %%%

    a2=calls2+80;
    b2=(a2-calls2)/ague;
    c2=ones(1,b2);
    d2=datrate*calls2;
    vec02=d2*c2;
    vec12=datrate:datrate*ague:datrate*calls2;
    vec22=1:ague:calls2+80;
    vec32=[vec12 vec02];

    axes(handles.axes1);

%     plot(vec22,vec32,'-b')
```

```
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');

        graph = plot(vec22,vec32,'p-b',vec2,vec3,'p-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 15.85)', 'Th UpLink (AMR
15.85)');

        return


elseif(codec==29)

      datrate=0.0407; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 40.7 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 19.85:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 19.85:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);
```

```
        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
```

```
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');



        graph = plot(vec22,vec32,'p--b',vec2,vec3,'p--r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 19.85)', 'Th UpLink (AMR
19.85)');

        return


elseif(codec==30)

      datrate=0.0439; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 43.9 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%

        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);
```

```
        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 23.05:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 23.05:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
```

```
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');

        graph = plot(vec22,vec32,'p:b',vec2,vec3,'p:r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 23.05)', 'Th UpLink (AMR
23.05)');


        return

elseif(codec==31)

      datrate=0.0447; %valor de Data rate en capa PHY

        %%%%%Cálculo del th con el code rate%%%%%%

        thulcr=(thul*coderate)/1000000;
        set(handles.text74,'string',thulcr);

        thdlcr=(thdl*coderate)/1000000;
        set(handles.text76,'string',thdlcr);

        set(handles.text77,'string', 'Cada UE genera en promedio
un tráfico de 44.7 Kbps por llamada');

        %%%%%%% Cálculo del numero de llamadas admitidas con G711
%%%%%%%%%%%%%%%%%
```

```
        voip1=thulcr/datrate;
        calls1=floor(voip1);

        voip2=thdlcr/datrate;
        calls2=floor(voip2);


        set(handles.text78,'string', 'No. máximo de llamadas
admitidas para UL con AMR 23.85:');
        set(handles.text80,'string', 'No. máximo de llamadas
admitidas para DL con AMR 23.85:');

        set(handles.text79,'string', calls1);
        set(handles.text81,'string', calls2);

        ague=2;

        %%%%%%%%%%%%%%%%%% GENERACION DE LAS GRAFICAS UL-DL
%%%%%%%%%%%%%%%%%%

        for i=1:ague:calls1+80
               for j=i:ague:calls1
                   x=datrate*i;
               end
             %disp(['Numero de UEs:',num2str(i)] );
             x=x*1;
             %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
             %disp(['Trhoughput Mbbps: ',num2str(x)]);

        end

        for p=1:ague:calls2+80
            for q=p:ague:calls2

                 y=datrate*p;
                 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            end
            %disp(['Numero de UEs:',num2str(p)] );
            y=y*1;
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %disp(['Trhoughput bps: ',num2str(y)]);

        end

        %%%GRAFICA PARA UL %%%
        a=calls1+80;
        b=(a-calls1)/ague;
        c=ones(1,b);
        d=datrate*calls1;
        vec0=d*c;
        vec1=datrate:datrate*ague:datrate*calls1;
```

```
        vec2=1:ague:calls1+80;
        vec3=[vec1 vec0];

        axes(handles.axes1);

%         plot(vec2,vec3,'-r')
        grid on;
        hold on

        %%%GRAFICA PARA EL DOWNLINK %%%

        a2=calls2+80;
        b2=(a2-calls2)/ague;
        c2=ones(1,b2);
        d2=datrate*calls2;
        vec02=d2*c2;
        vec12=datrate:datrate*ague:datrate*calls2;
        vec22=1:ague:calls2+80;
        vec32=[vec12 vec02];

        axes(handles.axes1);

%         plot(vec22,vec32,'-b')
%         grid on;
%         xlabel('UE');
%         ylabel('Throughput Mbps');
%         title('Throughput teórico vs UE ');
%         legend('Th UpLink','Th DownLink');


        graph = plot(vec22,vec32,'h-b',vec2,vec3,'h-r');
        grid on;


        xlabel('UE');
        ylabel('Throughput Mbps');
        title('Throughput teórico vs UE ');
        legend(graph,'Th DownLink (AMR 23.85)', 'Th UpLink (AMR
23.85)');

        return


    end




function edit6_Callback(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
```

```
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit6 as text
%        str2double(get(hObject,'String')) returns contents of
edit6 as a double


% --- Executes during object creation, after setting all
properties.
function edit6_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in popupmenu16.
function popupmenu16_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu16 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu16 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu16


% --- Executes during object creation, after setting all
properties.
function popupmenu16_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu16 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
```

```
end


% --- Executes on selection change in popupmenu15.
function popupmenu15_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu15 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu15 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu15


% --- Executes during object creation, after setting all
properties.
function popupmenu15_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu15 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in popupmenu17.
function popupmenu17_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu17 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)



cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000 0.4700
0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500 0.5900
0.3900 0.4200 0.4600 0.5100 0.5500 0.6000 0.6400 0.6900 0.7400
0.7800 0.8100];

modindex = get(handles.popupmenu17, 'value');
```

```
            if (modindex==1)
                                set(handles.popupmenu18, 'String',
'Code Rate');

            elseif (modindex==2)
                                m = 2;
                                elements = cr(1:10);
                                set(handles.popupmenu18, 'String',
elements);

            elseif (modindex==3)
                                m = 4;
                                elements = cr(11:17);
                                set(handles.popupmenu18, 'String',
elements);

            elseif (modindex==4)
                                m = 6;
                                elements = cr(18:28);
                                set(handles.popupmenu18, 'String',
elements);


            end



% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu17 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu17


% --- Executes during object creation, after setting all
properties.
function popupmenu17_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu17 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
% --- Executes on selection change in listbox2.
function listbox2_Callback(hObject, eventdata, handles)
% hObject    handle to listbox2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
listbox2 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
listbox2


% --- Executes during object creation, after setting all
properties.
function listbox2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: listbox controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in popupmenu18.
function popupmenu18_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu18 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


set(handles.popupmenu18, 'String', '');

cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000 0.4700
0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500 0.5900
0.3900 0.4200 0.4600 0.5100 0.5500 0.6000 0.6400 0.6900 0.7400
0.7800 0.8100];

mod_index = get(handles.popupmenu17, 'value');


            if (mod_index==1)
                                    set(handles.popupmenu18, 'String',
'Seleccione un índice de modulación...');
```

```
            elseif (mod_index==2)
                                m = 2;
                                elements = cr(1:10);
                                set(handles.popupmenu18, 'String',
elements);

                                coderate =
get(handles.popupmenu18,'value');


                                    if (coderate==1)
                                        set(handles.text89,
'String', cr(1));
                                        set(handles.text117,
'String', cr(1));

                                    elseif (coderate==2)
                                        set(handles.text89,
'String', cr(2));
                                        set(handles.text117,
'String', cr(2));

                                    elseif (coderate==3)
                                        set(handles.text89,
'String', cr(3));
                                        set(handles.text117,
'String', cr(3));

                                    elseif (coderate==4)
                                        set(handles.text89,
'String', cr(4));
                                        set(handles.text117,
'String', cr(4));

                                    elseif (coderate==5)
                                        set(handles.text89,
'String', cr(5));
                                        set(handles.text117,
'String', cr(5));

                                    elseif (coderate==6)
                                        set(handles.text89,
'String', cr(6));
                                        set(handles.text117,
'String', cr(6));

                                    elseif (coderate==7)
                                        set(handles.text89,
'String', cr(7));
                                        set(handles.text117,
'String', cr(7));
```

```
                                    elseif (coderate==8)
                                        set(handles.text89,
'String', cr(8));
                                        set(handles.text117,
'String', cr(8));

                                    elseif (coderate==9)
                                        set(handles.text89,
'String', cr(9));
                                        set(handles.text117,
'String', cr(9));

                                    elseif (coderate==10)
                                        set(handles.text89,
'String', cr(10));
                                        set(handles.text117,
'String', cr(10));

                                    end

            elseif (mod_index==3)
                                m = 4;
                                elements = cr(11:17);
                                set(handles.popupmenu18, 'String',
elements);

                                coderate =
get(handles.popupmenu18,'value');

                                    if (coderate==1)
                                        set(handles.text89,
'String', cr(11));
                                        set(handles.text117,
'String', cr(11));

                                    elseif (coderate==2)
                                        set(handles.text89,
'String', cr(12));
                                        set(handles.text117,
'String', cr(12));

                                    elseif (coderate==3)
                                        set(handles.text89,
'String', cr(13));
                                        set(handles.text117,
'String', cr(13));

                                    elseif (coderate==4)
                                        set(handles.text89,
'String', cr(14));
```

```
                                        set(handles.text117,
'String', cr(14));

                                    elseif (coderate==5)
                                        set(handles.text89,
'String', cr(15));
                                        set(handles.text117,
'String', cr(15));

                                    elseif (coderate==6)
                                        set(handles.text89,
'String', cr(16));
                                        set(handles.text117,
'String', cr(16));

                                    elseif (coderate==7)
                                        set(handles.text89,
'String', cr(17));
                                        set(handles.text117,
'String', cr(17));


                                    end


            elseif (mod_index==4)
                                m = 6;
                                elements = cr(18:28);
                                set(handles.popupmenu18, 'String',
elements);

                                coderate =
get(handles.popupmenu18,'value');

                                    if (coderate==1)
                                        set(handles.text89,
'String', cr(18));
                                        set(handles.text117,
'String', cr(18));

                                    elseif (coderate==2)
                                        set(handles.text89,
'String', cr(19));
                                        set(handles.text117,
'String', cr(19));

                                    elseif (coderate==3)
                                        set(handles.text89,
'String', cr(20));
                                        set(handles.text117,
'String', cr(20));
```

```
'String', cr(21));

'String', cr(21));


'String', cr(22));

'String', cr(22));


'String', cr(23));

'String', cr(23));


'String', cr(24));

'String', cr(24));


'String', cr(25));

'String', cr(25));


'String', cr(26));

'String', cr(26));


'String', cr(27));

'String', cr(27));


'String', cr(28));

'String', cr(28));

%
%
'String', cr(29));
```

```
elseif (coderate==4)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==5)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==6)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==7)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==8)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==9)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==10)
    set(handles.text89,

    set(handles.text117,


elseif (coderate==11)
    set(handles.text89,

    set(handles.text117,


  elseif (coderate==12)
      set(handles.text89,
```

```
%                                       set(handles.text117,
'String', cr(29));

                                    end


            end


% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu18 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu18


% --- Executes during object creation, after setting all
properties.
function popupmenu18_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu18 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton14.
function pushbutton14_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton14 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


helpdlg({'DwPTS / GP/ UpPTS' '(Downlink Pilot Signal / Guard
Period / Uplink Pilot Signal):' '' 'Los valores desplegados para
cada uno de los tres campos de la subtrama especial están en
función de símbolos OFDM'},...
        'Ayuda');


% --- Executes on button press in pushbutton15.
function pushbutton15_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton15 (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'DL:UL (Downlink:Uplink):' 'Subtramas dedicadas al enlace
descendente y al enlace ascendente respectivamente'},...
        'Ayuda');



% --- Executes on button press in pushbutton17.
function pushbutton17_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton17 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg('Time Division Duplex Configuration Index: El índice
determina el número de subtramas dedicadas al enlace ascendente y
al enlace descendente',...
        'Ayuda');



% --- Executes on button press in pushbutton16.
function pushbutton16_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton16 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


helpdlg('Special Subframe Configuration Index: Cada valor
representa un tipo de configuración para el número de símbolos
OFDM que se le asignarán a los campos DwPTS, GP, y UpPTS',...
        'Ayuda');


% --- Executes when figure1 is resized.
function figure1_ResizeFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --- Executes on button press in pushbutton18.
function pushbutton18_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton18 (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg('Indica el número total de Resource Elements (de canal
PDSCH) disponibles para datos en enlace de Downlink',...
        'Ayuda');


% --- Executes on button press in pushbutton19.
function pushbutton19_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton19 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg('Indica el número total de Resource Elements (de canal
PUSCH) disponibles para datos en enlace de Uplink',...
        'Ayuda');


% --- Executes on button press in pushbutton20.
function pushbutton20_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton20 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Throughput [bps] en enlace de Downlink sin uso de bits
de redundancia para la codificación del canal' '' 'Throughput =
(REs de PDSCH)*(índice de modulación)*(100 - Número de tramas
transmitidas en un segundo)' ''
'Throughput=[symbol]*[bits/symbol]*[s] = [bps]'},...
        'Ayuda');

% --- Executes on button press in pushbutton21.
function pushbutton21_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton21 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Throughput [bps] en enlace de Uplink sin uso de bits de
redundancia para la codificación del canal' '' 'Throughput = (REs
de PUSCH)*(índice de modulación)*(100 - Número de tramas
transmitidas en un segundo)' ''
'Throughput=[symbol]*[bits/symbol]*[s] = [bps]'},...
        'Ayuda');

% --- Executes on button press in pushbutton22.
function pushbutton22_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton22 (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'MCS: Modulation and Coding Scheme.' '' 'm: bits por
símbolo usados para la modulación' '' '      m=2 --> QPSK' '' '
m=4 --> 16 QAM' '' '      m=6 --> 64 QAM' '' '' 'Code Rate: Es
definido como la tasa de datos a transmitir que se pueden asignar
a una subtrama en relación a la tasa de datos a transmitir máxima
que puede ser asignada en la subtrama. Un CR menor indica que se
incluyen más bits de redundancia en el proceso de codificación del
canal, mientras un CR mayor refiere el uso de menos bits de
redundancia.'},...
        'Ayuda');


% --- Executes on button press in pushbutton23.
function pushbutton23_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton23 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


cla %clear axes


% --- Executes on button press in pushbutton24.
function pushbutton24_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton24 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Tasa de transmisión de datos a nivel de capa 3' '' 'Sólo
se consideran 40 Bytes de encabezado' '' '' '12+8+20 Bytes =
RTP+UDP+IP'},...
        'Ayuda');

% --- Executes on button press in pushbutton25.
function pushbutton25_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton25 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


helpdlg({'Tasa de transmisión calculada para la capa física
considerando la información de capas superiores (Encabezados de
aplicación, de transporte, y de enlace de datos)' '' 'Se
consideran 52 Bytes de encabezado' '' '' '(12+8+20)+(12) Bytes =
(RTP+UDP+IP)+(PDCP,RLC,MAC)'},...
```

```
        'Ayuda');


% --- Executes on button press in pushbutton29.
function pushbutton29_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton29 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'El throughput considerado en esta sección hace
referencia al throughput de capa física multiplicado por el code
rate elegido'},...
        'Ayuda');


% --- Executes on selection change in popupmenu21.
function popupmenu21_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu21 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu21 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu21

BW = get(handles.popupmenu21,'Value');


            if (BW==1)
                msgbox({'Seleccione un BW de canal'},
'Achtung','warn');
                PRB = 0;
                set(handles.text102,'String', '');
            elseif (BW==2)
                set(handles.text102,'String', '6');
            elseif (BW==3)
                set(handles.text102,'String', '15');
            elseif (BW==4)
                set(handles.text102,'String', '25');
            elseif (BW==5)
                set(handles.text102,'String', '50');
            elseif (BW==6)
                set(handles.text102,'String', '75');
            elseif (BW==7)
                set(handles.text102,'String', '100');
            end
```

```
% --- Executes during object creation, after setting all
properties.
function popupmenu21_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu21 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton30.
function pushbutton30_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton30 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Physical Resource Blocks' '' 'Número de bloques de
recursos físicos de acuerdo al ancho de banda del canal'},...
        'Ayuda');


% --- Executes on key press with focus on popupmenu21 and none of
its controls.
function popupmenu21_KeyPressFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu21 (see GCBO)
% eventdata  structure with the following fields (see UICONTROL)
%    Key: name of the key that was pressed, in lower case
%    Character: character interpretation of the key(s) that was
pressed
%    Modifier: name(s) of the modifier key(s) (i.e., control,
shift) pressed
% handles    structure with handles and user data (see GUIDATA)


% --- Executes on button press in pushbutton31.
function pushbutton31_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton31 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
helpdlg({'Para hacer uso completo de herramientas sobre la
gráfica:' '' ' - Haga click derecho sobre el cuadro de
acotaciones*' ' - Haga uso del toolbox ubicado en la parte
superior izquierda de la interfaz' '' '' '' '  * Para ver el
cuadro completo de acotaciones seleccione la opción -Refresh-
'},...
        'Ayuda');


% --- Executes on button press in pushbutton32.
function pushbutton32_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton32 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


helpdlg({'La capacidad de llamadas está en referencia a las
llamadas que se pueden llevar a cabo en una célula de red' ''
'Considerese el enlace de UL como las llamadas salientes de la
célula (UE hacia eNodeB)' 'Considerese el enlace de DL como las
llamadas entrantes a la célula (eNodeB hacia UE)'},...
        'Info');


% --- Executes on button press in pushbutton33.
function pushbutton33_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton33 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Realiza el cálculo del throughput en la célula de LTE
para enlaces de DL y UL en referencia a la elección de indice de
configuración TDD, indice de configuración de Subtrama especial,
índice de modulación, code rate, y tipo de codec'},...
        'Ayuda');


% --- Executes on button press in pushbutton34.
function pushbutton34_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton34 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Realiza el cálculo de Resource Elements en el canal para
enlaces DL y UL desplegando en el menú lateral derecho del
apartado de capa física el tipo de configuración elegida de
acuerdo a la elección de índices de configuración TDD y SS e
índice de modulación'},...
        'Ayuda');
```

```
% helpdlg({''},...
%         'Ayuda');


% --- Executes on button press in pushbutton35.
function pushbutton35_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton35 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


[s,v] = listdlg('PromptString','Seleccione un modelo de
propagación:',...
'SelectionMode','single',...
'ListString',{'LTE Modelo vehicular','Clase WiMAX Cat A B C
LTE'},...
'ListSize',[200 300]);


        if (v==1)

                if(s==1)

                    Do = 100;    %[m]
                    ht= 1.5;     %[m] altura terminal
                    hr= 25;      %[m] altura radiobase
                    f_c    = 1700090000;   %[Hz]
                    lambda = 0.176461246;
                    A = 77.05121043;  %[dB] =
20*LOG10((4*3.14159264945*Do/lamda))
                    Cf     = -0.423348497; %[dB] =
6*LOG10(f_c/2000000000)
                    Ch_a   = 1.336844482;  %[dB]
                    Ch_b   = 1.336844482;  %[dB]
                    Ch_c   = 2.498774732;  %[dB]
                    n_a    = 4.615    ;
                    n_b    = 4.1675   ;
                    n_c    = 3.9;
                    s_a    = 10.6;     %[dB]
                    s_b    = 9.6;      %[dB]
                    s_c    = 8.2;      %[dB]
                    eNB_Nf = 2.3; %dB
                    ue_Nf = 4; %dB
                    BW = 180000; %Hz
                    N_eNB  = -119.1276523; %dBm
                    N_ue = -117.4276523; %dBm
                    Backoff      = 0; %dB
                    eNB_G  = 18; %dBi
                    Ptx_eNB      = 26.9897; %dBm
                    EIRP_eNB = 44.9897; %dBm
```

```
                    ue_G = -1; %dBi
                    Ptx_Ue = 23.01029996; %dBm
                    EIRP_ue = 22.01029996; %dBm
                    s = 0; %dB


                    allparameters =
[Do;ht;hr;f_c;lambda;A;Cf;eNB_Nf;ue_Nf;BW;N_eNB;N_ue;Backoff;eNB_G
;ue_G;Ptx_eNB;Ptx_Ue;EIRP_ue;EIRP_eNB;s];

                    set(handles.uitable3, 'data', allparameters);

                    set(handles.text114, 'string','LTE Modelo
vehicular');

%                     get(handles.text114,'LTE Modelo vehicular');


                elseif(s==2)

                    Do = 100;    %[m]
                    ht= 1.5;     %[m]
                    hr= 40;      %[m]
                    f_c    = 2111350000;   %[Hz]
                    lambda = 0.142089185;
                    A = 78.93297682;  %[dB]
                    Cf     = 0.141181421;  %[dB]
                    Ch_a   = 1.336844482;  %[dB]
                    Ch_b   = 1.336844482;  %[dB]
                    Ch_c   = 2.498774732;  %[dB]
                    n_a    = 4.615    ;
                    n_b    = 4.1675   ;
                    n_c    = 3.9 ;
                    s_a    = 10.6;    %[dB]
                    s_b    = 9.6;     %[dB]
                    s_c    = 8.2;     %[dB]

                    eNB_Nf = 4; %dB
                    ue_Nf = 4; %dB
                    BW = 2699999; %Hz
                    N_eNB  = -147.4276523; %dBm
                    N_ue = -105.6667413; %dBm
                    Backoff     = 0; %dB
                    eNB_G  = 15; %dBi
                    Ptx_eNB     = 26.9897; %dBm
                    EIRP_eNB = 41.9897; %dBm
                    ue_G = -1; %dBi
                    Ptx_Ue = 26.98970004; %dBm
                    EIRP_ue = 25.98970004; %dBm
                    s = 10; %dB
```

```
                    allparameters =
[Do;ht;hr;f_c;lambda;A;Cf;eNB_Nf;ue_Nf;BW;N_eNB;N_ue;Backoff;eNB_G
;ue_G;Ptx_eNB;Ptx_Ue;EIRP_ue;EIRP_eNB;s];

                    set(handles.uitable3, 'data', allparameters);

                    set(handles.text114, 'string','Clase WiMAX
CatA,B,C LTE');


                elseif(s==3)



                elseif(s==4)



                end

        elseif (v==0)

            msgbox({'Selección no válida' '' 'Seleccione un modelo
de propagación para la obtención de parámetros y gráficas
correspondientes'}, 'Warning','warn');

        end




% --- Executes on button press in pushbutton36.
function pushbutton36_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton36 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% model = get(s);

%SNR vehículo vs Distance

SNR = [53.46096225
42.62388241
36.28459708
31.78680256
25.44751724
20.94972272
17.46096225
14.61043739
12.20035297
```

10.11264288
8.271152068
6.623882408
5.133745742
3.77335755
2.521921725
1.363273123
0.284597082
-0.724436968
-1.672278762
-2.565927776
-3.411247227
-4.213197436
-4.976012203
-5.703334102
-6.398319689
-7.063722294
-7.701957905
-8.315158119
-8.905213102
-9.473806721
-10.02244552
-10.55248276
-11.06513857
-11.56151681
-12.04261943
-12.50935861
-12.96256719
-13.40300762
-13.83137966
-14.24832707
-14.65444345
-15.05027728
-15.43633643
-15.81309205
-16.18098199
-16.54041395
-16.89176809
-17.23539953
-17.57164048
-17.90080214
-18.22317647
-19.74229295
-21.12679001
-22.39859666
-23.57468486
-24.6684595
-25.69069288
-26.65017189
-27.55415783
-28.40872032
-29.2189846

```
-29.98931781
-30.72347152
-31.42469224
-32.0958082
-32.73929826
-33.35734716
-33.95189033
-34.5246504
-35.07716731
-35.61082326
-36.12686354
-36.62641407
-37.11049612
-37.58003896
-38.03589053
-38.47882671
-38.90955922
-39.32874254
-39.73697987
-40.13482834
-40.52280362
-40.90138389
-41.27101343
-41.63210573
-41.98504625
-42.33019496
-42.66788846
-42.99844203
-43.32215134
-43.63929409];


% SNR Line of Sight= [84.36614192
% 78.34554201
% 74.82371683
% 72.32494209
% 68.80311691
% 66.30434218
% 64.36614192
% 62.782517
% 61.44358121
% 60.28374227
% 59.26069182
% 58.34554201
% 57.5176883
% 56.76191709
% 56.06667496
% 55.42298129
% 54.82371683
% 54.26314235
% 53.73656358
% 53.24009191
```

% 52.77046999
% 52.32494209
% 51.90115611
% 51.49708839
% 51.11098529
% 50.74131717
% 50.38674183
% 50.04607505
% 49.71826672
% 49.40238138
% 49.09758205
% 48.80311691
% 48.51830813
% 48.24254244
% 47.97526321
% 47.71596367
% 47.46418112
% 47.21949199
% 46.98150753
% 46.74987007
% 46.52424987
% 46.30434218
% 46.08986487
% 45.8805562
% 45.6761729
% 45.47648848
% 45.28129173
% 45.09038537
% 44.90358485
% 44.72071726
% 44.54162041
% 43.69766681
% 42.92850177
% 42.22194253
% 41.56856019
% 40.96090761
% 40.39300018
% 39.85995629
% 39.35774187
% 38.88298494
% 38.43283812
% 38.00487522
% 37.59701205
% 37.20744498
% 36.83460278
% 36.4771083
% 36.1337478
% 35.80344604
% 35.485246
% 35.17829217
% 34.88181664
% 34.59512759

```
% 34.31759952
% 34.04866505
% 33.78780791
% 33.53455704
% 33.28848139
% 33.04918555
% 32.81630592
% 32.58950741
% 32.36848048
% 32.15293866
% 31.94261628
% 31.73726654
% 31.53665971
% 31.34058164
% 31.14883236
% 30.96122486
% 30.77758399
% 30.59774548
% 30.42155507];

d = [100
200
300
400
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600
2800
3000
3200
3400
3600
3800
4000
4200
4400
4600
4800
5000
5200
5400
5600
5800
6000
```

6200
6400
6600
6800
7000
7200
7400
7600
7800
8000
8200
8400
8600
8800
9000
9200
9400
9600
9800
10800
11800
12800
13800
14800
15800
16800
17800
18800
19800
20800
21800
22800
23800
24800
25800
26800
27800
28800
29800
30800
31800
32800
33800
34800
35800
36800
37800
38800
39800
40800
41800
42800

```
43800
44800
45800
46800
47800
48800
49800];

        figure

        plot(d',SNR','-','Color',[0 0 1]);

        xlabel('Distancia [m]');
        ylabel('SNR [dB]');
        title('Signal to Noise Ratio vs Distancia');
%         legend(SNRvsd, 'SNR');
        grid on;

pause(0.00001);
frame_h = get(handle(gcf),'JavaFrame');
set(frame_h,'Maximized',1);


% --- Executes on button press in pushbutton37.
function pushbutton37_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton37 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)



%Distancia radial

% axes (handles.axes4);
figure
                    t=0:0.001:1;%Initializing time samples
                    %Transfer chara of sine and cosi.e sin Vs Cos
will give circle
                    %Here we are plotting group of sine Vscos with
differnt amplitudes
                    s15=15*sin(2*pi*t);
                    c15=15*cos(2*pi*t);
                    s14=14*sin(2*pi*t);
                    c14=14*cos(2*pi*t);
                    s13=13*sin(2*pi*t);
                    c13=13*cos(2*pi*t);
                    s12=12*sin(2*pi*t);
                    c12=12*cos(2*pi*t);
                    s11=11*sin(2*pi*t);
                    c11=11*cos(2*pi*t);
                    s10=10*sin(2*pi*t);
```

```
                    c10=10*cos(2*pi*t);
                    s9=9*sin(2*pi*t);
                    c9=9*cos(2*pi*t);
                    s8=8*sin(2*pi*t);
                    c8=8*cos(2*pi*t);
                    s7=7*sin(2*pi*t);
                    c7=7*cos(2*pi*t);
                    s6=6*sin(2*pi*t);
                    c6=6*cos(2*pi*t);
                    s5=5*sin(2*pi*t);%Sine wave with
amplitude=3unit
                    c5=5*cos(2*pi*t);%Cosine wave with
amplitude=3unit
                    s4=4*sin(2*pi*t);
                    c4=4*cos(2*pi*t);
                    s3=3*sin(2*pi*t);
                    c3=3*cos(2*pi*t);
                    s2=2*sin(2*pi*t);
                    c2=2*cos(2*pi*t);
                    s1=1*sin(2*pi*t);
                    c1=1*cos(2*pi*t);


%                   %propagation =
plot(s1,c1,s2,c2,s3,c3,s4,c4,s5,c5,s6,c6,s7,c7,s8,c8,s9,c9,s10,c10
,s11,c11,s12,c12,s13,c13,s14,c14,s15,c15);
                    %propagation =
plot(s1,c1,'Color',[1,1,0],s2,c2,'Color',[0.8,0.8,0],s3,c3,'Color'
,[0.6,0.6,0],s4,c4,'Color',[0.4,0.4,0],s5,c5,'Color',[0.2,0.2,0],s
6,c6,'Color',[0,1,0],s7,c7,'Color',[0,0.6,0],s8,c8,'Color',[0,0.3,
0],s9,c9,'Color',[0,0,1],s10,c10,'Color',[0,0,0.85],s11,c11,'Color
',[0,0,0.7],s12,c12,'Color',[0,0,0.55],s13,c13,'Color',[0,0,0.4],s
14,c14,'Color',[0,0,0.25],s15,c15,'Color',[0,0,0.2]);%Plotting sin
Vs Cos

                    propagation = plot(s1,c1,'-y',s2,c2,'-
y',s3,c3,'-y',s4,c4,'-y',s5,c5,'-y',s6,c6,'-g',s7,c7,'-g',s8,c8,'-
g',s9,c9,'-b',s10,c10,'-b',s11,c11,'-b',s12,c12,'-b',s13,c13,'-
b',s14,c14,'-b',s15,c15,'-b');%Plotting sin Vs Cos

                    title('Distancia Radial');

                    legend(propagation,'CR = 0.81 - SNR = 22.8
[dB]','CR = 0.78 - SNR = 21.3 [dB]','CR = 0.64 - SNR = 19.2
[dB]','CR = 0.55 - SNR = 17.4 [dB]','CR = 0.42 - SNR = 16.2
[dB]','CR = 0.59 - SNR = 13.6 [dB]','CR = 0.44 - SNR = 11.9
[dB]','CR = 0.39 - SNR = 10 [dB]','CR = 0.61 - SNR = 8.2 [dB]','CR
= 0.47 - SNR = 6.8 [dB]','CR = 0.34 - SNR = 5 [dB]','CR = 0.22 -
SNR = 4.4 [dB]','CR = 0.14 - SNR = 3.2 [dB]','CR = 0.11 - SNR = 3
[dB]','CR = 0.11 - SNR = 3 [dB]');
```

```
%                       graph1 = plot(s1,c1);
%                       legend(graph1,'CR = 0.81');
%
%                       graph2 = plot(s2,c2);
%                       legend(graph2,'CR = 0.78');
%
%                       graph3 = plot(s3,c3);
%                       legend(graph3,'CR = 0.64');
%
%                       graph4 = plot(s4,c4);
%                       legend(graph4,'CR = 0.55');
%
%                       graph5 = plot(s5,c5);
%                       legend(graph5,'CR = 0.42');
%
%                       graph6 = plot(s6,c6);
%                       legend(graph6,'CR = 0.59');
%
%                       graph7 = plot(s7,c7);
%                       legend(graph7,'CR = 0.44');
%
%                       graph8 = plot(s8,c8);
%                       legend(graph8,'CR = 0.39');
%
%                       graph9 = plot(s9,c9);
%                       legend(graph9,'CR = 0.61');
%
%                       graph10 = plot(s10,c10);
%                       legend(graph10,'CR = 0.47');
%
%                       graph11 = plot(s11,c11);
%                       legend(graph11,'CR = 0.31');
%
%                       graph12 = plot(s12,c12);
%                       legend(graph12,'CR = 0.22');
%
%                       graph13 = plot(s13,c13);
%                       legend(graph13,'CR = 0.14');
%
%                       graph14 = plot(s14,c14);
%                       legend(graph14,'CR = 0.11');
%
%                       %graph15 = plot(s15,c15);
%                       plot(s15,c15);
%                       legend(graph15,'');



%                       graph = plot(vec22,vec32,'v-b',vec2,vec3,'v-
r');
%                       grid on;
```

```
%
%
%                       xlabel('UE');
%                       ylabel('Throughput Mbps');
%                       title('Throughput teórico vs UE ');
%                       legend(graph,'Th DownLink (AMR 6.70)', 'Th
UpLink (AMR 6.70)');

                    %grid on;%Enable grid lines
                    axis equal;%Equal width of X and Y axis
                    axis off

pause(0.00001);
frame_h = get(handle(gcf),'JavaFrame');
set(frame_h,'Maximized',1);


% --- Executes on button press in pushbutton38.
function pushbutton38_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton38 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


%Prx vs Distance

PrxLoss = [-33.06151039
-39.0821103
-42.60393548
-45.10271021
-48.6245354
-51.12331013
-53.06151039
-54.64513531
-55.9840711
-57.14391004
-58.16696049
-59.0821103
-59.909964
-60.66573522
-61.36097735
-62.00467101
-62.60393548
-63.16450995
-63.69108873
-64.1875604
-64.65718232
-65.10271021
-65.5264962
-65.93056392
-66.31666702
```

-66.68633514
-67.04091047
-67.38157726
-67.70938558
-68.02527093
-68.33007026
-68.6245354
-68.90934418
-69.18510987
-69.4523891
-69.71168864
-69.96347119
-70.20816032
-70.44614478
-70.67778223
-70.90340244
-71.12331013
-71.33778744
-71.54709611
-71.75147941
-71.95116383
-72.14636058
-72.33726693
-72.52406746
-72.70693505
-72.8860319
-73.7299855
-74.49915053
-75.20570978
-75.85909212
-76.4667447
-77.03465213
-77.56769602
-78.06991043
-78.54466737
-78.99481419
-79.42277709
-79.83064026
-80.22020733
-80.59304953
-80.950544
-81.29390451
-81.62420627
-81.94240631
-82.24936014
-82.54583567
-82.83252472
-83.11005279
-83.37898726
-83.63984439
-83.89309527
-84.13917092

```
-84.37846676
-84.61134638
-84.8381449
-85.05917183
-85.27471365
-85.48503602
-85.69038577
-85.8909926
-86.08707067
-86.27881995
-86.46642745
-86.65006832
-86.82990683
-87.00609724];


d = [100
200
300
400
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600
2800
3000
3200
3400
3600
3800
4000
4200
4400
4600
4800
5000
5200
5400
5600
5800
6000
6200
6400
6600
6800
```

7000
7200
7400
7600
7800
8000
8200
8400
8600
8800
9000
9200
9400
9600
9800
10800
11800
12800
13800
14800
15800
16800
17800
18800
19800
20800
21800
22800
23800
24800
25800
26800
27800
28800
29800
30800
31800
32800
33800
34800
35800
36800
37800
38800
39800
40800
41800
42800
43800
44800
45800
46800

```
47800
48800
49800];

figure

        plot(d',PrxLoss','-','Color',[0 0.5 0]);

        xlabel('Distancia [m]');
        ylabel('Prx Loss [dBm]');
        title('Pérdida de potencia vs Distancia');
%         legend(SNRvsd, 'SNR');
        grid on;


pause(0.00001);
frame_h = get(handle(gcf),'JavaFrame');
set(frame_h,'Maximized',1);



%Pushbutton44 era para el despliegue de menu cuya informacion
llena el
%usuario

% --- Executes on button press in pushbutton44.
function pushbutton44_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton44 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

x = inputdlg({'Free Space Loss (Lf)' 'Frecuencia central (fc)'
'Distancia (d)' 'Altura a la que se encuentra la antena de UE(ht)'
'Altura de antena eNB (hr)'},...
             'Introduzca los valores solicitados', [1 70]);


Do = x{3};
lambda = (300000000/(x{2}));   %lambda = c/fc

Amu = 20*log10((4*3.14159264945*Do/lambda)); %necesitas el segundo
y tercer elemento de la tabla
%allparameters = {x;Amu};


%                     eNB_Nf    = 2.3; %dB
%                     ue_Nf = 4; %dB
%                     BW = 180000; %Hz
%                     N_eNB     = -119.1276523; %dBm
%                     N_ue = -117.4276523; %dBm
%                     Backoff   = 0; %dB
```

```
%                       eNB_G     = 18; %dBi
%                       Ptx_eNB   = 26.9897; %dBm
%                       EIRP_eNB = 44.9897; %dBm
%                       ue_G = -1; %dBi
%                       Ptx_Ue = 23.01029996; %dBm
%                       EIRP_ue = 22.01029996; %dBm
%                       s = 0; %dB


allparameters = [x;Amu];

set(handles.uitable3, 'data', allparameters);



%         MCS = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28];
%
%         m = [2 2 2 2 2 2 2 2 2 2 4 4 4 4 4 4 4 6 6 6 6 6 6 6 6 6
6 6 6];
%
%         cr = [0.1100 0.1400 0.1700 0.2200 0.2800 0.3400 0.4000
0.4700 0.5400 0.6100 0.3000 0.3400 0.3900 0.4400 0.5000 0.5500
0.5900 0.3900 0.4200 0.4600 0.5100 0.5500 0.6000  0.6400  0.6900
0.7400 0.7800 0.8100 0.8100];
%
%         tabla = [MCS' m' cr'];
%         set(handles.uitable1, 'data', tabla);





% --- Executes on button press in pushbutton45.

%%%Botón de ayuda para
% cuadro modelo de propagación

function pushbutton45_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton45 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


%allparameters =
[Do;ht;hr;f_c;lambda;A;Cf;eNB_Nf;ue_Nf;BW;N_eNB;N_ue;Backoff;eNB_G
;ue_G;Ptx_eNB;Ptx_Ue;EIRP_ue;EIRP_eNB;s];

helpdlg({'Do: Distancia inicial' '' 'ht: altura de la terminal' ''
'hr: altura de la radiobase' '' 'fc: frecuencia central' ''
```

```
'lambda: c/fc   ; donde "c" es la velocidad de la luz' '' 'Amu:
Atenuación promedio. Está en función de Do y de lambda' '' 'Cf: '
'' 'eNB Nf: Noise Figure de eNB' '' 'ue_Nf: Noise Figure de eNB'
'' 'BW: Ancho de banda del canal' '' 'N eNB: Ruido en eNB' '' 'N
ue: Ruido en ue' '' 'Backoff: ' '' 'eNB G: Ganancia en eNB' '' 'ue
G: Ganancia en UE' '' 'Ptx eNB: Potencia de transmisión de eNB' ''
'Ptx Ue: Potencia de transmisión de UE' '' 'EIRP ue: Potencia
Isotrópica Radiada Equivalente de UE' '' 'EIRP eNB: Potencia
Isotrópica Radiada Equivalente de UE' '' 's: ' '' '' ''},...
        'Ayuda');




% --- Executes on button press in pushbutton47.
function pushbutton47_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton47 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

helpdlg({'Configuración de MIMO (Multiple Input Multiple Output)
como Multiplexado Espacial seleccionando el número de antenas
involucradas en el proceso de transmisión-recepción'},...
        'Ayuda');


% --- Executes on selection change in popupmenu25.
function popupmenu25_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu25 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu25 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from
popupmenu25

thdl = str2num(get(handles.text76,'string'));
thul = str2num(get(handles.text74,'string'));

MIMO = get(handles.popupmenu25, 'value');



    if (MIMO==1)

        msgbox({'Configuración de MIMO' '' 'Seleccione el tipo de
configuración MIMO'}, 'Error','warn');
```

```
    elseif(MIMO==2) %mimo 2x2

        thdlmimo = thdl*2;
        set(handles.text124, 'string',thdlmimo);

    elseif(MIMO==3) %mimo 4x4

        thdlmimo = thdl*4;
        set(handles.text124, 'string',thdlmimo);

    elseif(MIMO==4) %mimo 8x8

        thdlmimo = thdl*8;
        set(handles.text124, 'string',thdlmimo);

    end


% --- Executes during object creation, after setting all
properties.
function popupmenu25_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu25 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton49.
function pushbutton49_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton49 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)


helpdlg({'Selección de codec con el que se codificarán las
llamadas a establecer' '' 'La información deplegada en este menú
es:' '' ' -- Nombre de codec (Bit Rate de codec) -- ' ''},...
        'Ayuda');
```