

ANEXO A

En este anexo se recoge la función de MATLAB que se encarga de generar Códigos Gold de 1023 chips y la función de MATLAB encargada de cargar los parámetros de simulación. Para ello, la función tiene como parámetro de entrada un número entero (*PRN_number*), que va desde 1 hasta 37 dando lugar así a los 36 Códigos Gold distintos.

```
function [g, g_amp] = gold(PRN_number)

% [g, g_amp] = gold(PRN_number) devuelve el codigo Gold (1023 bits)
% numero PRN_number. El generador de codigos se compone de
% dos registros de desplazamiento, con 10 bits cada uno.
% Las realimentaciones que se aplican en el registro superior
% son: 10,3; y para el inferior, 2,3,8,10. Ambos casos responden
% a m-secuencias de longitud maxima. La seleccion de los codigos
% depende de tap1 y tap2, que se calculan a partir de PRN-number.
% g_amp: codigo gold ampliado para medios intervalos de chip.

xored_taps = [2 6; 3 7; 4 8; 5 9; 1 9; 2 10; 1 8; 2 9; 3 10; 2 3
              3 4; 5 6; 6 7; 7 8; 8 9; 9 10; 1 4; 2 5; 3 6; 4 7
              5 8; 6 9; 1 3; 4 6; 5 7; 6 8; 7 9; 8 10; 1 6; 2 7
              3 8; 4 9; 5 10; 4 10; 1 7; 2 8; 4 10];
% Para evitar bucles if se prefiere una tabla
% de correspondencias en la memoria.

r1 = [1 1 1 1 1 1 1 1 1 1];
r2 = [1 1 1 1 1 1 1 1 1 1];

tap1 = xored_taps(PRN_number, 1);
tap2 = xored_taps(PRN_number, 2);

for k = 1:1023
    g(k) = xor(r1(10), xor(r2(tap1), r2(tap2)));
    vec_aux1(1) = xor(r1(10), r1(3));
    vec_aux1(2:10) = r1(1:9);
    r1 = vec_aux1;
    vec_aux2(1) = xor(r2(2), xor(r2(3), xor(r2(6), ...
        xor(r2(8), xor(r2(9), r2(10))))));
    vec_aux2(2:10) = r2(1:9);
    r2 = vec_aux2;
end

aux = 2 * g - 1;          % Senalizacion polar para los codigos: +1 / -1.
g_amp(1:2:2045) = aux;
g_amp(2:2:2046) = aux;

%b = 2*g - 1;            % Senalizacion polar para el codigo PRN.
%corr = xcorr(b);
%vec_n = (-0.5 * (1 + length(corr)) + 1:0.5*(1 + length(corr)) - 1);
%plot(vec_n, corr), %title('Funcion de autocorrelacion del codigo PRN'), ...
% Fin de funcion.
```

La función empleada para cargar los parámetros de simulación se llama *empieza.m* y se carga siempre al comienzo de las simulaciones.

```
% Función que recoge los parámetros de la simulación
% Mediante la llamada a la función cargargold se generan
% los códigos GOLD.
clear;
cargargold;
M=2;
Tb=0.02046;
long=length(codgold);
Tc=Tb/(long-1);
Fc=1400e6;
N=16;
Tcs=Tc/N;
%semillas=[12345 56789 43244 5115 26542 87878 76452 3333 6565 8833 4582 7762 9988 ];
SNR=[];
%g=randint(1,12);
%InterSigNum=sum(g);
%cte=0;
```