wave_gen

Fonction permettant de générer une forme d'onde à partir d'une séquence binaire.

wave_gen(*binary_sequence*,'*line_code_name*',R_b)

binary_sequence = vecteur de 1 et de 0

line_code_name = type de mise en forme.
ex : 'polar_nrz' , 'unipolar_rz' , 'bipolar_rz' , 'manchester'

R_b = taux des données binaires, en bit/seconde ou Hz.

Pour afficher une onde : **waveplot**

EXEMPLE :

>> b = [1 0 1 0 1 1] ;

>> x = wave_gen(b , 'unipolar_nrz' , 1000) ;

>> waveplot(x)

binary

Fonction permettant de générer un vecteur aléatoire de 1 et de 0

binary(N)

N = nombre de bits composant le vecteur

EXEMPLE :

>> b = binary(10) ;

acf

Fonction permettant de générer et d'afficher la fonction d'autocorrélation d'une séquence de données ou d'une onde.

acf(x)

x = vecteur de nombres ou onde créée par la fonction *wave_gen*

EXEMPLE :

>> acf(wave_gen(binary(1000),'polar_nrz',1000)) ;

psd

Fonction permettant de générer et d'afficher la densité spectrale de puissance d'une séquence de données ou d'une onde.

psd(x)

x = vecteur de nombres ou onde créée par la fonction *wave_gen*

EXEMPLE :

>> psd (wave_gen(binary(1000),'manchester',1000))

channel

Fonction simulant le comportement d'un canal de transmission.

channel(***input*** **,** ***gain*** **,** ***noise_power*** **,** ***bandwidth*** **)**

input = signal d'entrée (onde créée par *wave_gen*)

gain = gain du canal (nombre sans dimension)

noise_power = puissance du bruit (Watt)

bandwidth = bande passante du canal (Hz)

Cette fonction renvoie une autre onde qui peut être affichée par *waveplot*.

EXEMPLE :

```
>> b = binary(10) ;

>> x = wave_gen( b , 'polar_nrz' , 1000 ) ;

>> y = channel ( x , 1 , 0.01 , 4900 ) ;

>> waveplot(y)
```

eye_diag

Fonction permettant de générer et d'afficher le diagramme de l'oeil d'une onde créée par *wave_gen* ou *channel*

eye_diag(x , -1)

x = onde dont on veut déterminer le diagramme de l'oeil.

-1 = paramètre non nécessaire si on désire afficher le diagramme complet directement. Si ce paramètre est présent, le diagramme s'affiche, bit après bit, à chaque [RETURN]

EXEMPLE :

>> b = [1 0 0 1 0 1 1] ;

>> x = wave_gen(b,'polar_nrz',1000) ;

>> **eye_diag**(x) ;

match

Fonction permettant de simuler l'effet du filtre adapté. Attention : 2 utilisations possibles.

match('*line_code_name*') génère et affiche la réponse impulsionnelle du filtre adapté correspondant à *line_code_name*.

match('*line_code_name*' , *x*)

x = onde créée par wave_gen (ou channel) et qui se trouve à l'entrée du filtre adapté

line_code_name = permet d'adapter le filtre au code choisi

Dans ce 2ème cas, la fonction match crée une autre onde qui peut être visualisée grâce à la fonction *waveplot*.

EXEMPLE :

>> x = wave_gen([1 0 1] , 'polar_nrz' , 1000) ;

>> **match**('polar_nrz') ;

>> waveplot(**match**('polar_nrz' , x)) ;

Fonctions graphiques

1. clf

vide la fenêtre graphique en cours

2. subplot

permet d'afficher plusieurs graphiques dans la même fenêtre. Peut être utilisée avec les fonctions *axis* pour faire coïncider les axes des graphiques. Voir également la fonction *hold* pour afficher deux ondes dans les mêmes axes.

EXEMPLE :

```
>> clf

>> b=binary(10) ;

>> x=wave_gen(b,'polar_nrz',1000) ;

>> subplot(211),waveplot(x), a=axis ;

>> subplot(212),waveplot(channel(x,1,0.01,4900)) ;

>> axis(a) ;
```