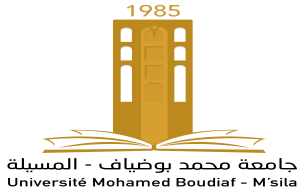
**الجمهوريــة الجزائرية الديمقراطية الشعبيــة**

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

**وزارة التعليـم العالـي و البحـث العلمــي**

**MINISTERE DE L’ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE**

****

***Université Mohamed Boudiaf - M’sila***

***Faculté de Technologie***

***Département d’Electronique***

**جامعة محمد بوضياف بالمســيلة**

**كلية التكنولوجيا**

**قسم الإلكترونيك**

**3ème ANNEE LMD**

**Option : ELECTRONIQUE**

**TP1  : GENERATION ET AFFICHAGE DE SIGNAUX,**

**DANS MATLAB**



**Enseignant :** Mostefa TABBAKH

**GÉNÉRATION DES SIGNAUX**

Ce TP a pour but de familiariser l'étudiant avec l'utilisation du logiciel MATLAB pour la génération des différents types de formes d'ondes.

The Signal Processing Toolbox de MATLAB contient des fonctions pour la génération des formes d'ondes couramment utilisées périodiques et non périodiques, séquences (impulsion, échelon, rampe), train d'impulsions…

**1. Génération des signaux périodiques**

Une autre classe de signaux très utile en traitement du signal et en automatique sont les signaux sinusoïdaux de la forme :

x[n]=A (w n+f ) 0 .cos

Avec MATLAB, ce type de signaux peut être génère en utilisant les operateurs trigonométriques cos et sin.

Le programme P1 est un exemple qui génère un signal sinusoïdal.

% Programme P1

% Generation d'un signal sinusoidal

n = 0:100;

f = 0.1;

phase = 0;

A = 1.5;

arg = 2\*pi\*f\*n - phase;

x = A\*cos(arg);

clf; %efface l’ancienne figure

stem(n,x);

axis([0 100 -2 2]);

grid;

title('Signal Sinusoidal');

xlabel('Temps indexe en n');

ylabel('Amplitude') ;

axis ;

**Application :**

1- Modifier le programme précédent pour générer et tracer un signal sinusoïdal de longueur 50, fréquence 0.08, amplitude 2.5 et phase décalée de 90 degrés.

2- Générer et tracer les séquences définies ci dessus :



En plus des fonctions sin et cos, le toolbox de MATLAB offre d'autres fonctions qui produisent des signaux périodiques comme sawtooth et square.

La fonction sawtooth génère une onde triangulaire avec des maximums en +-1 et une période de 2\*pi.

La fonction square génère une onde carrée avec une période de 2\*pi. Un paramètre optionnel spécifie le cycle, la période en pourcent pour laquelle le signal est positif.

Le programme P1 génère 1.5 secondes d'onde triangulaire (respectivement carre), de fréquence 50Hz et avec une fréquence d'échantillonnage de 10KHz.

%Programme P1

fs = 100;

t = 0:.0001:.0625;

x1 = sawtooth(2\*pi\*50\*t);

x2 = square(2\*pi\*50\*t);

subplot(2,1,1),plot(t,x1), xlabel('Time (sec)'); ylabel('Amplitude'); title('Sawtooth Periodic Wave')

subplot(2,1,2), plot(t,x2), xlabel('Time (sec)'); ylabel('Amplitude'); title('Square Periodic Wave')

**2. Génération des signaux : Impulsion Unité et Echelon Unité**

Deux signaux élémentaires de base sont : l’impulsion Unité et l’échelon Unité. L’impulsion unité u[n] de longueur N peut être générée en utilisant la commande MATLAB suivante :

u = [1 zeros(1,N-1)] ;

De même, l’impulsion unité ud[n] de longueur N et décalées de M échantillons, tel que M<N, peut être générée par la commande MATLAB suivante :

ud = [zeros(1,M) 1 zeros(1,N-M-1)];

L’échelon unité s[n] de longueur N peut être génère en utilisant la commande MATLAB suivante :

s=[ones(1,N)]

La version décalée de l’échelon unité s’obtient de la même manière que dans le cas de l’impulsion unité.

Le programme P3 peut être utilise pour générer et tracer un signal impulsion unité :

%Programme P3

% Génération d'une impulsion unité

% Génération d'un vecteur de -10 a 20

n=-10:20;

% Génération de l'impulsion unité

u = [zeros(1,10) 1 zeros(1,20)];

% Tracer le signal genere

stem(n,u);

xlabel('Temps indexe en n'); ylabel('Amplitude');

title('Impulsion Unite');

axis([-10 20 0 1.2]);

**Application :**

1- Modifier le programme précédent pour générer une impulsion unité décalée de 11 échantillons. Exécuter le programme et tracer le signal génère.

2- Modifier le programme précédent pour générer un échelon unité décale de 20 échantillons.

**3. Génération des signaux complexes**

Des signaux complexes peuvent être génères en utilisant les opérations faites pour les signaux élémentaires.

Par exemple un signal de modulation d’amplitude peut être génère par la modulation d’un signal sinusoïdal de haute fréquence xH[n] = cos(ωHn) avec un signal sinusoïdal de basse fréquence xB[n] = cos(ωBn). Le signal résultant y[n] a la forme suivante :

y[n] = A(1+m xB[n]) xH[n] = A(1 + m cos(ωBn)) cos(ωHn)

Avec m, appelé facteur de modulation, est choisi pour s’assurer que l’expression (1+m xB[n]) est positive pour toutes les valeurs de n.

Le programme P4 est utilise pour générer un signal module en amplitude.

% Programme P4

% Génération d’un signal module en amplitude

clf ;

n = 0 :100 ;

m = 0.4 ; fH = 0.1 ; fL = 0.01 ;

xH = sin(2\*pi\*fH\*n);

xL = sin(2\*pi\*fL\*n);

y = (1+m\*xL).\*xH;

stem(n,y); grid;

xlabel('Temps indexe en n'); ylabel('Amplitude');

**Application :**

Générer et tracer un signal de modulation d’amplitude de fréquence fH=0.08 et fL=0.04 et de facteur de modulation m=0.5. Interpréter.

**4. Génération des signaux aléatoires**

Un signal aléatoire de longueur N avec une distribution uniforme dans l’intervalle [0 1] peut être génère par la commande de MATLAB suivante :

x = rand(1,N) ;

De même, u signal aléatoire x[n] de longueur N avec une distribution normalisée a moyenne nulle et variance unite peu être génère en utilisant la commande suivante de MATLAB :

x = randn (1,N) ;

**Application :** Générer et tracer un signal aléatoire de longueur 100 tels que ces éléments sont uniformément distribues dans l’intervalle [-2,2].