Plan du TP N°5

Titre du TP :

GENERATION ET PRODUIT DE CONVOLUTION
DES SIGNAUX DISCRETS

Objectifs :

A la fin de la séance de travaux pratiques l'étudiant doit être capable de :

- Générer des signaux discrets ;
- Réaliser la convolution entre deux signaux discrets.

Pre-requis :

- Notions théoriques de base sur :
  - Signaux discrets ;
  - Le produit de convolution des signaux discrets;
- Manipulation de la Maquette DSP ;
- Matlab.

TP N°5 : GENERATION ET PRODUIT DE CONVOLUTION DES SIGNAUX DISCRETS

OBJECTIF GENERAL :

Générer quelques signaux discrets et Réaliser leurs produits de convolutions.

OBJECTIFS SPECIFIQUES	ELEMENTS DE CONTENU	MOYEN	DUREE
▪ L'étudiant sera capable de : - Charger le programme de fonctionnement du produit de convolution du logiciel Code explorer; - Comprendre le programme de l'application; -Restaurer les résultats de traitement et les comparer par rapport aux résultats théoriques.	1. Procédure de chargement ; 2. Procédure de fonctionnement ; saisi des valeurs des amplitudes des signaux discrets dans les cases mémoires spécifiées par le programme ; 3. Lecture des résultats de convolution à partir des cases mémoires spécifiques et comparaison avec les résultats théoriques.	▪ Maquette de Traitement de signal ; ▪ Ordinateur ; ▪ Logiciel Code Explorer.	120 mn
▪ L'étudiant sera capable de : - Générer les signaux discrets en utilisant le logiciel Matlab.	1. Procédure de chargement de matlab ; 2. Ecriture de programme en Matlab ; 3. Exécution et Test de programme ; 4. Interprétation des résultats.	▪ Ordinateur ; ▪ Logiciel Matlab.	60 mn

Travaux Pratiques N°5 :

GENERATION ET PRODUIT DE CONVOLUTION DES SIGNAUX DISCRETS

Objectif :

L'objectif de ce TP est de générer quelques signaux discrets et de déterminer leur produits de convolutions en utilisant la maquette DSP et MATLAB.

I. Rappel théorique

I.1. Représentation temporelle des signaux discrets

La variable temporelle du signal est maintenant discrète n, il est noté x[n].
Soit l'écriture de la séquence discrète d'une manière générale:

$$\{\ x[n]\} = \begin{cases} \{\ x[N],\ x[N+1],\ ....x[M]\} & ,N \le M \\ longueur: l = M - N + 1 \end{cases}$$

I.2. Produit de convolution numérique

La convolution inclue la réflexion, l'inversion et les opérations de multiplication et de somme.
Les séquences apériodiques sont supposées nulles hors de leur intervalle de définition.
On considère les deux séquences numériques apériodiques non-nul sur les intervalles de durée N_x et N_h .
Soit x(n) séquence non nul pour n ϵ [0, N_{x-1}] et f(n) séquence non nul pour n ϵ [0, N_{h-1}]

$$X(k):\ 0 \le k \le N_{x-1} \ \text{ et } \ h(k):\ 0 \le k \le N_{h-1}$$

La convolution apériodique de x(n) et de f(n) est exprimée par :

$$\mathbf{Y[k] = x[k] * h[k]} = \sum_{tous\ \ les\ p} x[p].\,h[k-p]$$

- **Exemple : Calcul du produit de convolution par la méthode graphique :**

Soient les deux signaux discrets suivant :

$$f_1[k] = 1.\delta[k] + 2.\delta[k-1] + 3.\delta[k-2] + 4.\delta[k-3]$$
$$f_2[k] = 9.\delta[k] + 7.\delta[k-1] + 4.\delta[k-2] + 1.\delta[k-3]$$

Le produit de convolution $f_1[k]* f_2[k]$ est le suivant :

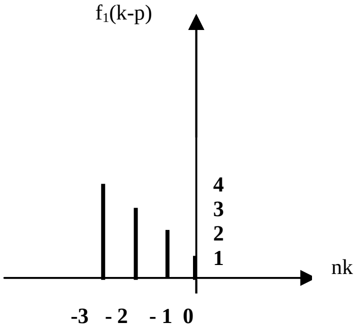
$$Y[k] = f_1[k] * f_2[k] = \sum_{p=0}^{p=3} f_1[p].f_2[k-p]$$

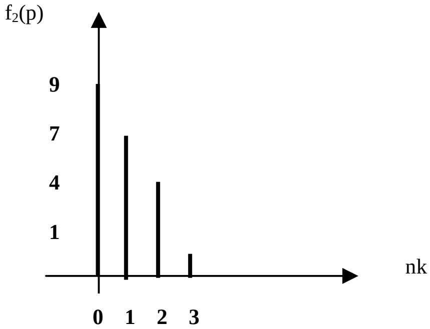
***Calcul des valeurs de k :**

$k \geq 0 + 0 = 0$

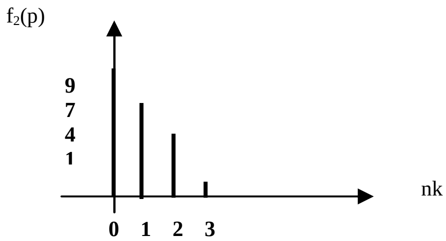
$\Rightarrow \quad 0 \leq k \leq 6$

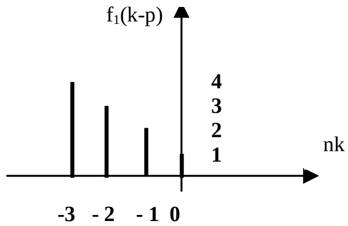
$k \leq 3 + 3 = 6$





<u>1er cas : pour k =0</u>

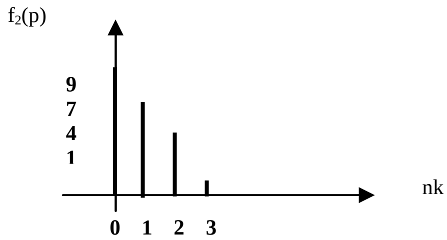


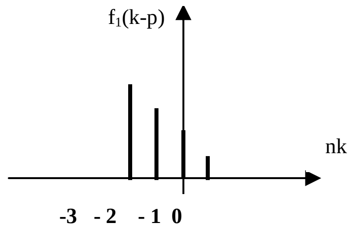


y(k) = 9.1 = 9

D'où **<u>y(0) = 9</u>**

<u>2eme cas : pour k =1</u>

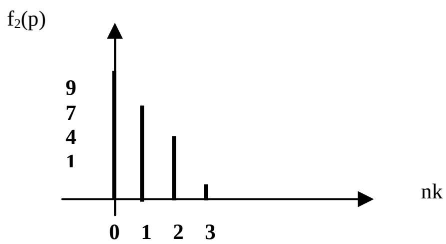


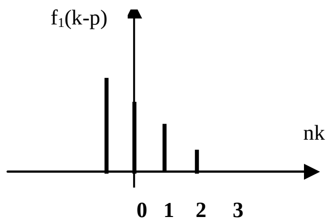


y(1) = 9.2 + 7.1 = 25

<u>D'où y(1) = 25</u>

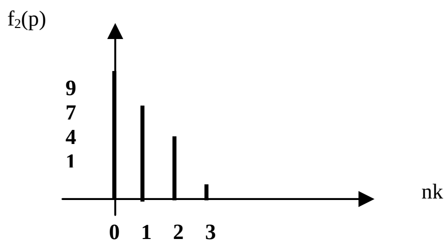
<u>3eme cas : pour k = 2</u>

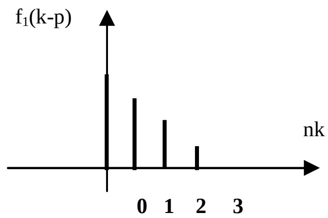




y(k) = 9.3 + 7.2 + 4.1 = 42

<u>4eme cas : pour k = 3</u>



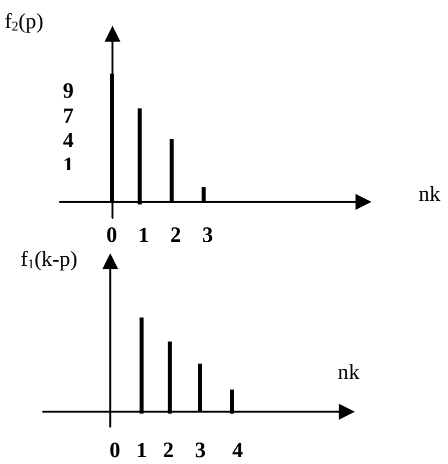


y(k) = 9.4 + 7.3 +4.2+1 = 66

D’où y(2) = 42

D’où y(3) = 66

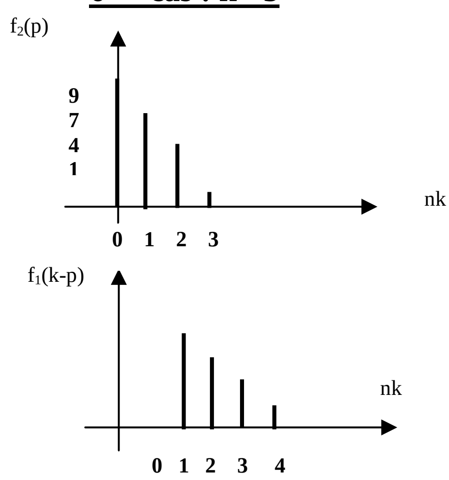
5^{eme} cas :K= 4



y(k) = 7.4 + 4.3 + 1.2 = 45

D’où y(5) = 45

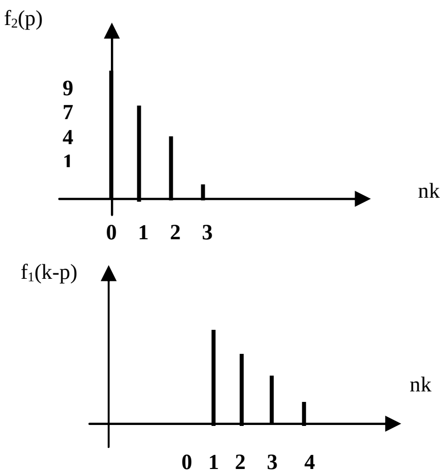
6^{eme} cas : k =5



y(k) = 4.4 + 1.3 = 19

D’où y(6) = 19

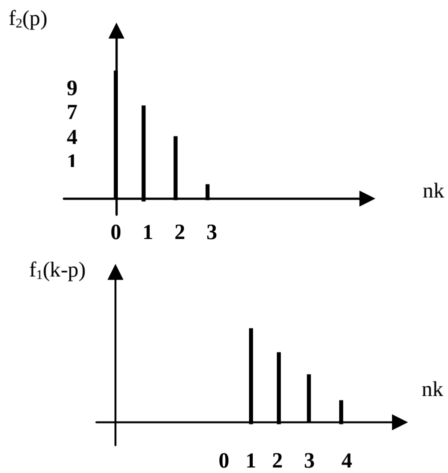
7^{eme} cas :K= 6



y(k) = 4. 1 = 4

D’où y(6) = 4

8^{eme} cas :K= 7

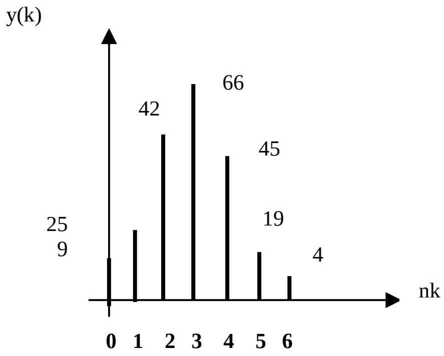


Pas d’intersection entre les deux signaux.

D’où y(7) = 0

Figure (3) : Tous les cas possibles pour avoir le produit de convolution de deux signaux discrets $f_1(k)$ et $f_2(k)$.

La représentation du produit de convolution y(k) est :



Figure(4) : Produit de convolution des deux signaux.

II . Partie pratique

Soit l'exemple traité par le constructeur de la maquette de traitement de signal selon le programme suivant :

Le programme **MATHPS.ASM**, du logiciel « CODE EXPLORER» , représente la convolution de deux signaux de séquences « **4 »** et qui est :[1,2,3,4] *[9,7,4,1] .

La première séquence est stockée dans le registre (Buffer) Y et la deuxième séquence est stockée dans le registre **DATAI**.

Le résultat de la convolution est stocké dans le registre **Z**.

II.1. Matériels utilisés

- Maquette : L'unité CIC-520 ;
- Ordinateur ;
- Câble de communication entre PC et maquette.

II.2. Manipulation

1 . Connecter l'ordinateur à la maquette de traitement de signal « CIC-520 » ;

2 . Charger le programme « mathps.obj » du logiciel **C54x Code Explorer** selon le chemin suivant :

C:\dskplus\demos\mathtst\mathps.obj.

3 . Le produit de deux signaux sera stocké dans la partie mémoire d'adresse « 0X020C » :

- Cliquer sur **View** de la barre d'outils ;
- Choisir **Memory ;**
- Entrer la nouvelle adresse choisie pour stocker le résultat du produit**: 0X020C.**

4 . Entrer les valeurs de chaque signal selon l'ordre donné par le constructeur :

- les valeurs du premier signal seront stockées dans le registre « **Y ».**
- les valeurs du deuxième signal seront stockées dans le registre **« DATAI ».**
- Le résultat du produit de ces deux signaux sera stocké dans le registre **« Z».**

```
DATAI : word   9,7,4,1,0,0,0          ;deuxième séquence
Z     : word  0,0,0,0,0               ;registre des résultats
Y     : word   4,3,2,1                ;la première séquence
```

5 . Exécuter puis arrêter le programme en appuyant sur **RUN** puis sur **HALT**.

Le résultat sera stocké dans la case mémoire d'adresse **0X020C**.

6 . Recharger le programme de nouveau pour effacer les résultats précédents.

7 .Soient les signaux discrets suivants :

a- $f_1[k] = 1.\delta[k] + 2.\delta[k-1] + 4.\delta[k-2] + 1.\delta[k-3]$
$f_2[k] = 5.\delta[k] + 1.\delta[k-1] + 4.\delta[k-2] + 2.\delta[k-3]$

b- $f_3[k] = 2.\delta[k] + 4.\delta[k-1] + 5.\delta[k-2] +6.\delta[k-3]$
$f_4[k] = 1.\delta[k] + 2.\delta[k-1] + 3.\delta[k-2] + 2.\delta[k-3]$

c- $f_5[k] = 1.\delta[k] + 2.\delta[k-1] + 3.\delta[k-2] + 4.\delta[k-3]$
$f_6[k] = 9.\delta[k] + 2.\delta[k-1] + 4.\delta[k-2] + 1.\delta[k-3]$

d- $f_7[k] = 1.\delta[k] + 2.\delta[k-1] + 3.\delta[k-2] + 4.\delta[k-3]$
$f_8[k] = 1.\delta[k] + 6.\delta[k-1] + 4.\delta[k-2] + 1.\delta[k-3]$

e- $f_1[k] = 1.\delta[k] +1.\delta[k-1] + 1.\delta[k-2] + 1.\delta[k-3]$
$f_2[k] = 2.\delta[k] + 1.\delta[k-1] + 2.\delta[k-2] +1.\delta[k-3]$

7.1. Représenter les signaux discrets pour chaque cas.

7.2. Calculer le produit de convolution pratiquement et graphiquement.

7.3. Interpréter les résultats trouvés.

III . Simulation par MATLAB

Dans cette partie vous allez voir quelques signaux discrets utilisés dans le traitement de signal :

III .1. L'échelon unité

Pour le cas de l'échelon unité, on se contentera d'un nombre fini d'échantillons.

- **Soit le programme suivant :**

```
n=- 20:20 ;
x=[zeros(1,20),ones(1,21)] ;
stem(n,x) ;
axis([-20 20 -0.5 1.5]) ;
title('u[n]');
xlabel('n') ;
ylabel('Amplitude') ;
```

Ecrire et tester ce programme.

III .2. La fonction Sinus

- **Soit le programme suivant :**

```
% La fonction sinus
n=-10:10;
x=sin(0.35*n);
stem(n,x);
axis([-10 10 -1.5 1.5]);
title('sinus');xlabel('n');ylabel('Amplitude');
```

Tester ce programme et déterminer les caractéristiques de ce signal .

III .3. La fonction exponentielle décroissant

- **Soit le programme suivant :**

```
%exponentielle décroissant : x[n] = e^-0.2n .u[n]
n=-10:10;
u=[zeros(1,10),ones(1,11)];
x=exp(-0.2*n).*u;
stem(n,x);axis([-10 10 -1.5 1.5]);
title('Exponentielle retardée');
xlabel('n');ylabel('Amplitude');
```

Tester ce programme et utiliser ce programme pour déterminer la fonction **:**
$y[n] = 2.\ e^{-0.1n}\ .u[n]$.

Remarque : On utilise ici l'opérateur de multiplication termes à termes (.*) qui permet d'effectuer la multiplication terme à terme de deux matrices.

III .4. Décalage des signaux numériques

Les signaux numériques sont souvent exprimés comme des combinaisons d'autres signaux élémentaires décalés.

❖ **Soit le programme suivant :**

```
t=-10:10;
u=[zeros(1,10),ones(1,11)];
subplot(3,1,1);
stem(t,u);
axis([-10 10 -1.5 1.5]);
title('u[n]');
xlabel('n');
ylabel('Amplitude');
subplot(3,1,2);
deltam2=[zeros(1,2),u(1:length(d
elta)-2)];
stem(t,deltam2);
axis([-10 10 -1.5 1.5]);
title('u[n-2]');
xlabel('n');
ylabel('Amplitude');
subplot(3,1,3);
deltap2=[u(3:length(delta)),zero
s(1,2)];
stem(t,deltap2);
axis([-10 10 -1.5 1.5]);
title('u[n+2]');
xlabel('n');
ylabel('Amplitude');
```

Ecrire, tester et déterminer les caractéristiques des différentes fonctions de ce programme.