Laboratorio

de

Introducción al

Procesamiento de

Señales

Año 2015

Alumno: Luciano Molinero 385/9

Ejercicio 1

La señal hA[n] retornada por la función hcanald.p representa la respuesta impulsional de un sistema SLID FIR que modela el comportamiento de un canal digital con dos ecos.

1. Grafique esta respuesta al impulso e interprete en qué consiste el modelo mencionado. Escriba la ecuación en diferencias que representa el sistema. ¿Qué puede decir acerca de la estabilidad

del sistema?

2. Escriba en MATLAB una función que implemente un sistema con esta ecuación en diferencias.

Verifique que este sistema tiene la misma respuesta impulsional que la obtenida en 1.

Por otra parte, el comando1

[x, fs] = audioread(’audio.wav’);

permite cargar un archivo de audio (en este caso, el provisto entre los archivos de la presente practica)

en la variable x. La variable **fs** indica la frecuencia de muestreo de la señal, concepto que estudiaremos más adelante. Como cualquier otra variable, se puede graficar utilizando el comando plot(x). Es posible escuchar la señal mediante el comando:

sound (x,fs);

3. Usando la función escrita en 2. halle la salida del sistema (que llamaremos y) cuando se aplica a

su entrada la señal de audio x. ¿Cómo se escucha esta señal? Grafique la señal de entrada y de

salida en el tiempo.

1. Con el siguiente código en matlab realizo un vector de enteros “n”(el eje x), y luego creo la función delta, que es la señal impulso unitario o Delta de Kronecker :

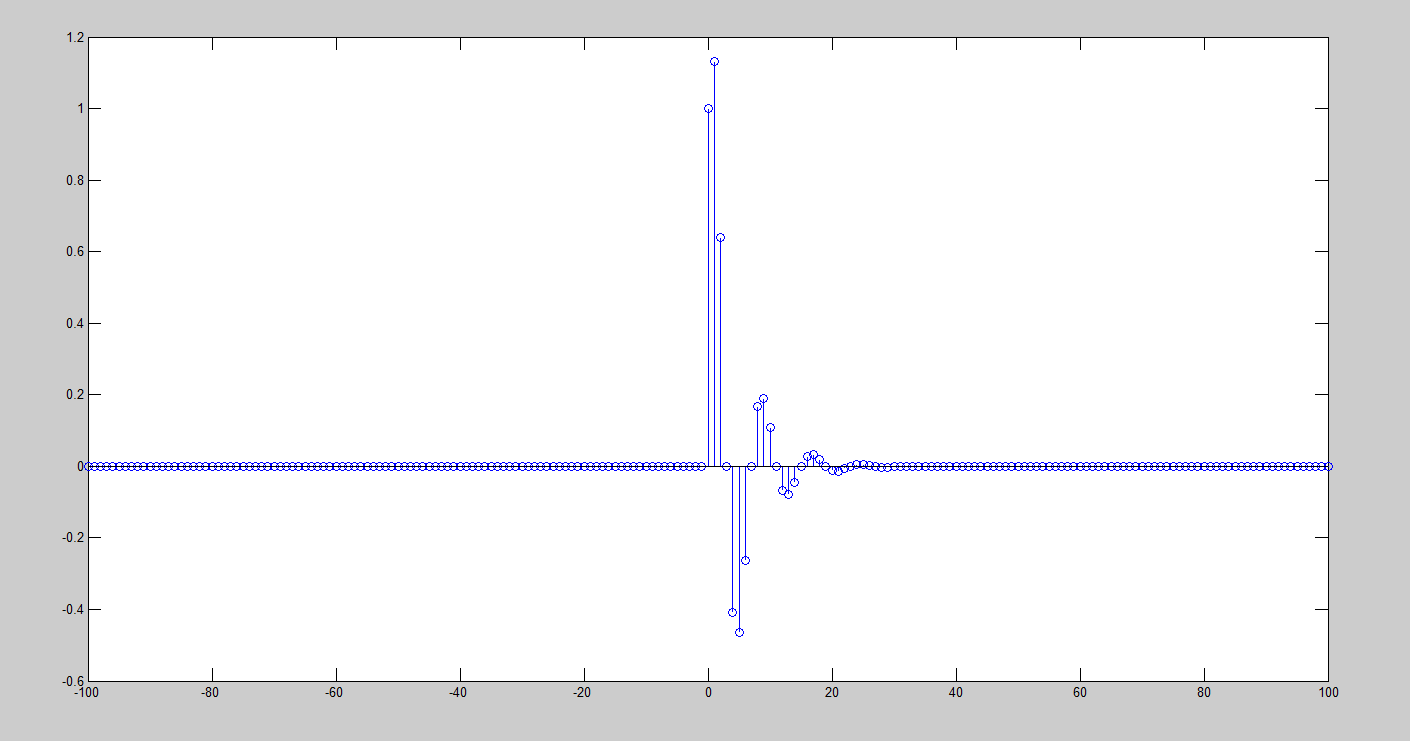
N = 100;

n = [-N:N];

Delta = [zeros (1,N) 1 zeros(1,N)];

Luego aplico la señal a la entrada del sistema usando la guía del SistemaA:

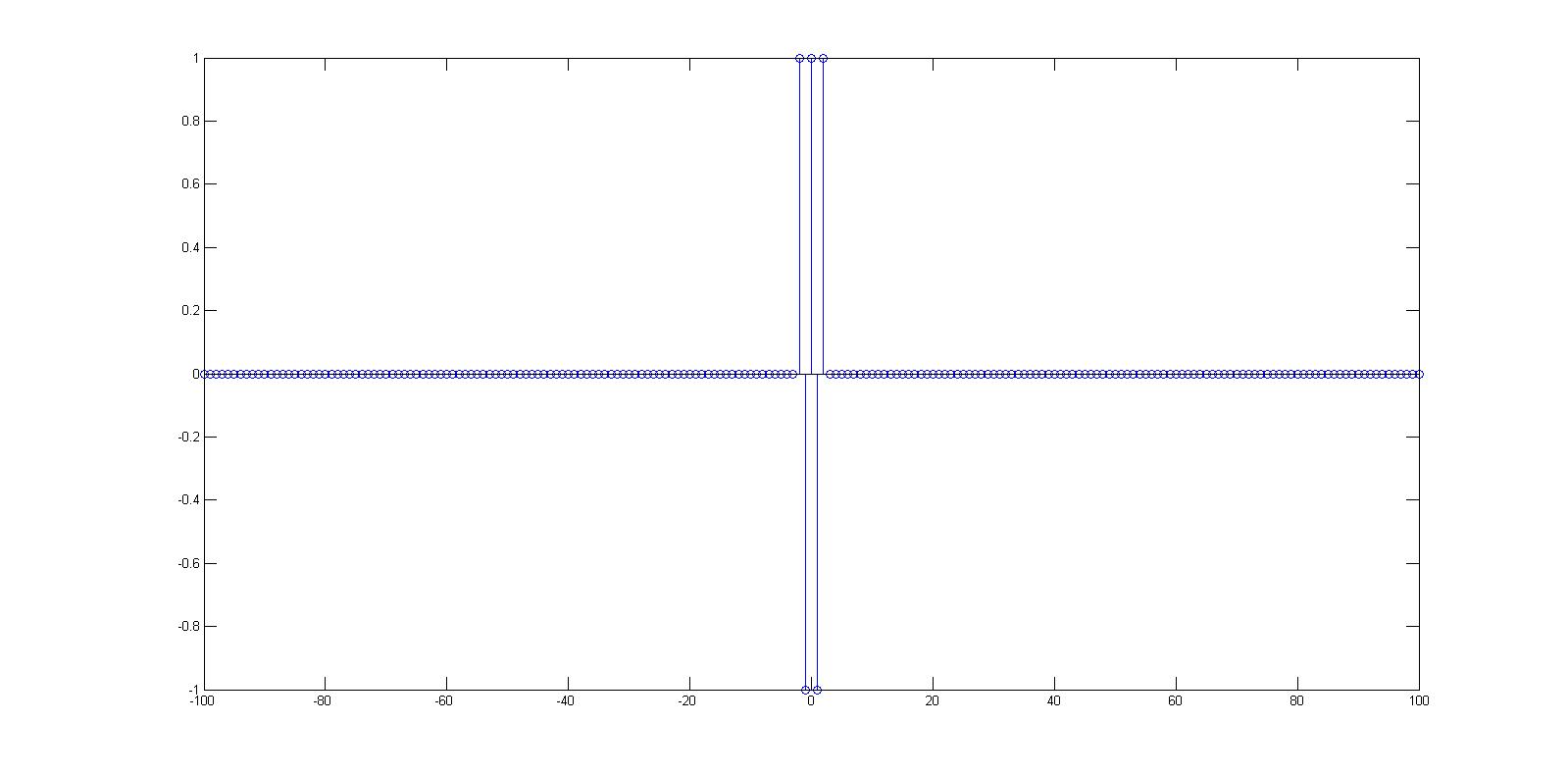
y = sistemaA(n,delta,385)

stem(n, y)

Se puede decir que el sistema es estable (también se podría notar más aumentando el eje de las abscisas, o seas el rango que toma las “n” )

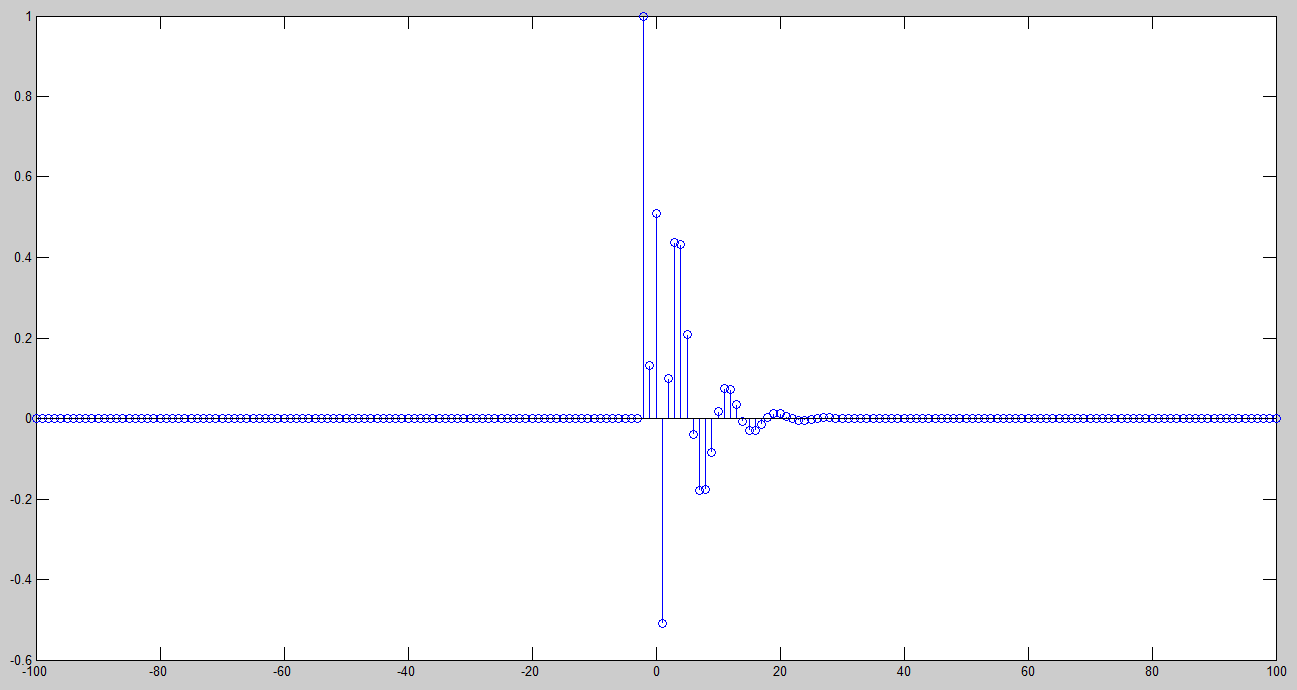
1. Para la señal cajón= x1[n] = (−1)n⊓5[n] , utilizo la misma señal “delta”, pero modificada, para que tome los valores en el eje “Y” = 1 , -1 , 1 , -1 , 1 ;

desde el valor de la abscisa x=n= -2 … 2

Gráficamente se puede interpretar de la siguiente manera:

y = sistemaA(n,cajon,385); stem(n,cajon);

La salida del sistema, como respuesta de la señal cajón x1[n] es la siguiente:

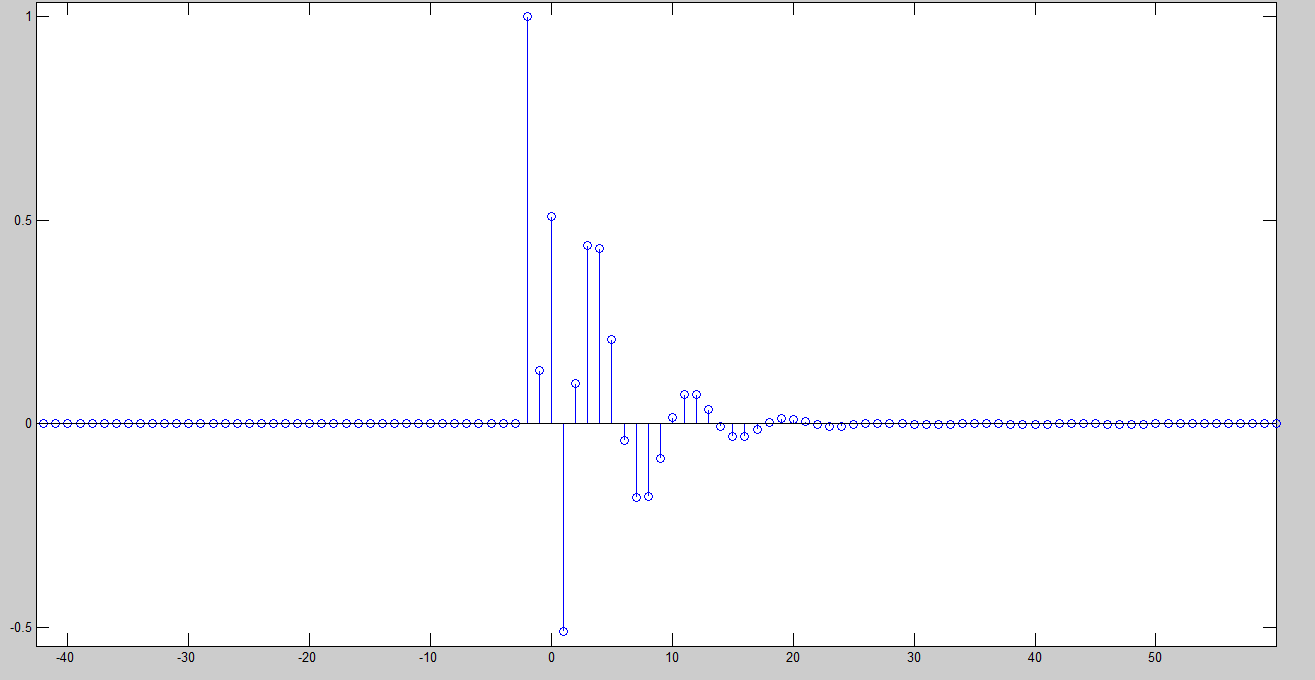


1. Luego para la convolución entre la respuesta impulsional, y la señal cajón discreto, vuelvo a cargar a “y” con la delta,para después usar el comando “conv”, y este es el script usado:

y = sistemaA(n,delta,385)

conv(caj,y)

stem(m,ans)//Con “m” como variable:m =[2\*n(1):2\*n(end)];



Concluyendo (aumentando con la lupa la figura) queda demostrado que es un sistema “IIR”,pero no se puede determinar completamente porque no se puede evaluar en el infinito con Matlab.

*Ejercicio 2*

*Dada la señal hB[n], respuesta impulsional de un sistema SLID FIR, retornada por la función respuestaB.p .*

*1. Escriba la ecuación en diferencias que describe al sistema.*

*2. Escriba en MATLAB una función que implemente el sistema.*

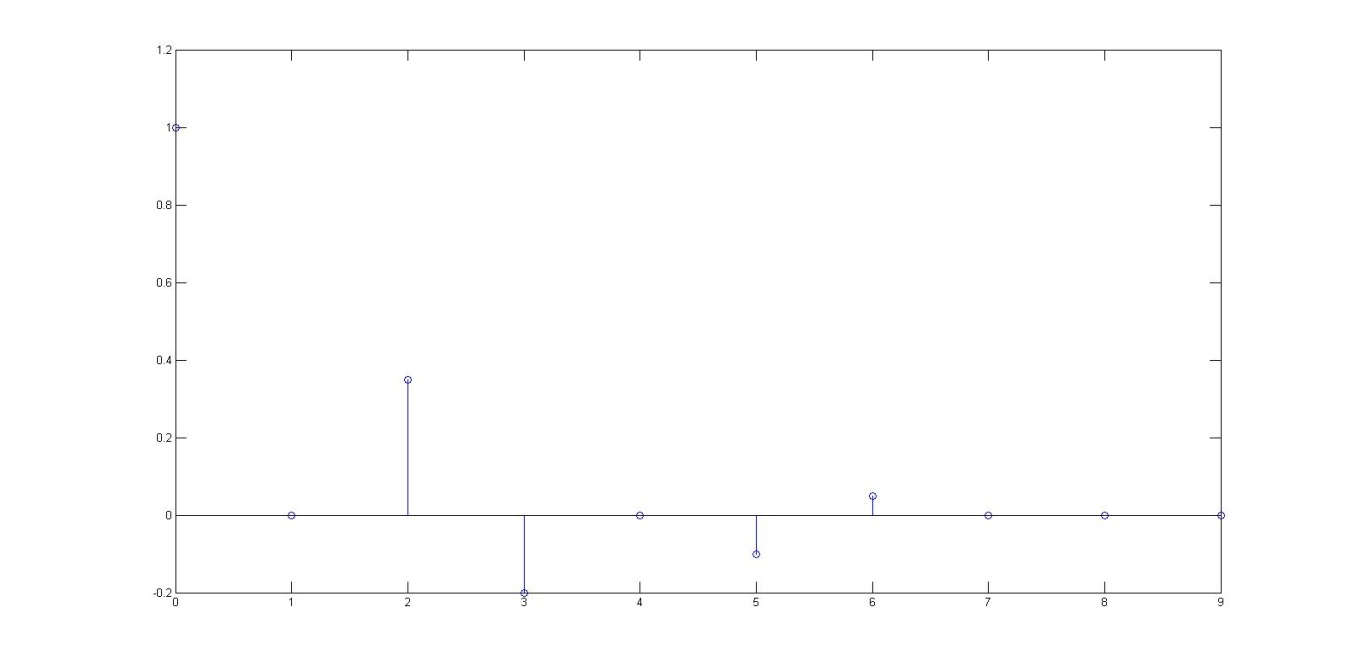
*3. Utilizando su implementación, halle la respuesta del sistema cuando se aplica a su entrada la señal x2[n] = Triángulo5[n].*

*4. Calcule la convolución de las señales hB[n] y x2[n]. Compárela con el resultado del inciso anterior.*

1. Utilizando los comandos en Matlab de ayuda, averiguo la figura de la cual se trata:

[n,h] = respuestaB(385)

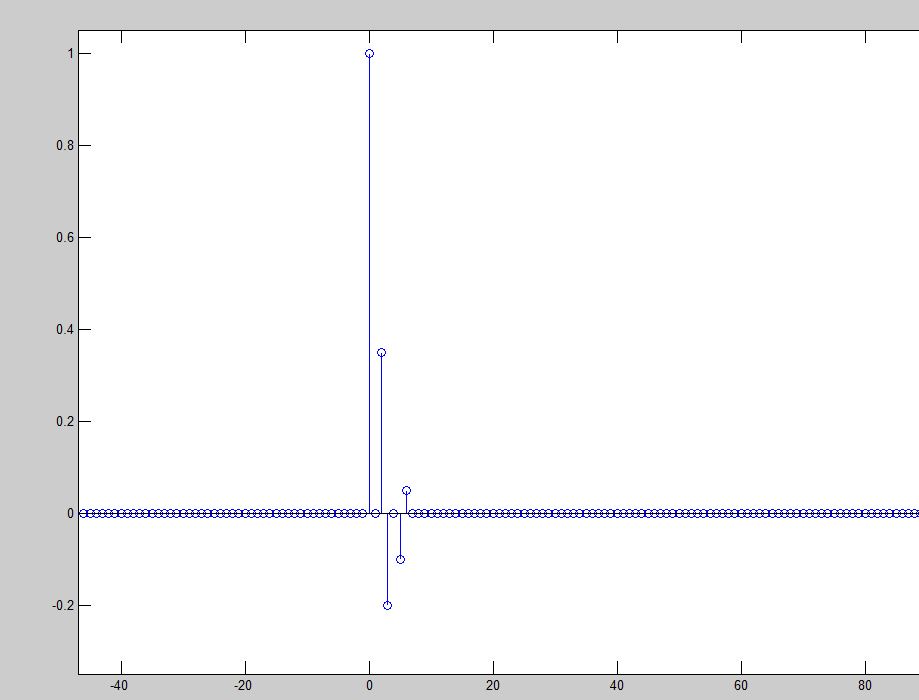
stem(n,h)

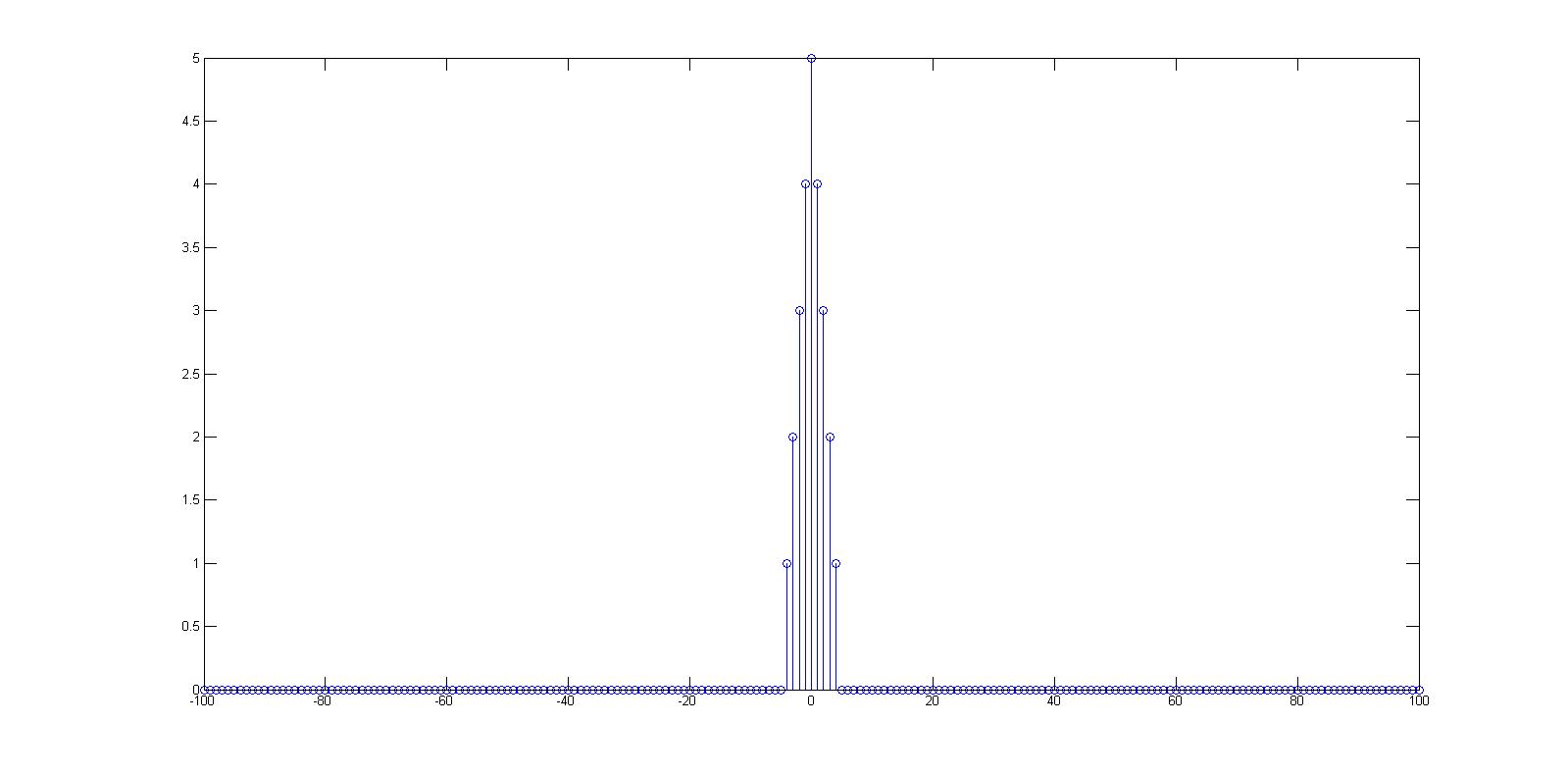


La ecuación que describe al sistema es :

Y[n]= X(n) + 0.35\*X(n-2)+ (-0,2)\*X(n-3)+(-0.1)\*X(n-5)+(0.05)\*X(n-6)

1. La implementación de este sistema en Matlab es la siguiente:
2. N=100;
3. n = -N:N;
4. x = 1\*(n==0);
5. figure, stem(n,x);
6. y = zeros(size(x));
7. for k = 1:length(x)
8. if (k == 1)
9. y(k)=1\*x(k);
10. else
11. if k == 3
12. y(k)= 1\*x(k)+(0.35)\*x(k-2);
13. else
14. if k == 4
15. y(k)= 1\*x(k)+(0.35)\*x(k-2)+(-0.2)\*x(k-3);
16. else
17. if k==6
18. y(k)= 1\*x(k)+(0.35)\*x(k-2)+(-0.2)\*x(k-3)+(-0.1)\*x(k-5);
19. else
20. if k>=7
21. y(k)= 1\*x(k)+(0.35)\*x(k-2)+(-0.2)\*x(k-3)+(-0.1)\*x(k-5)+(0.05)\*x(k-6);
22. end
23. end
24. end
25. end
26. end
27. end
28. stem(n,y)

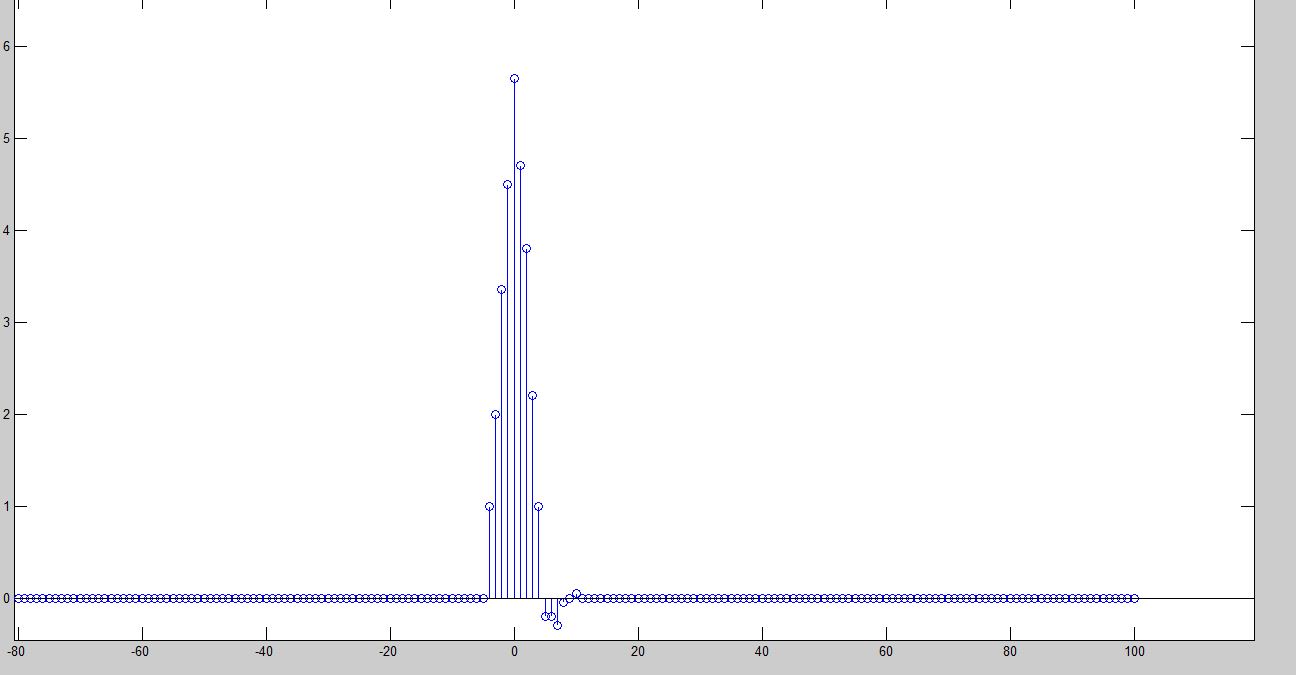


3.Para la señal x2[n] = Triángulo 5 [n], utilizo la señal “delta” modificada para obtener la figura:

Tri = (1 , 2 , 3 , 4 , 5 , 4 , 3, 2, 1 ) (En el workspace, en la variable “Tri” que creo, doble click y modifico los valores).

Luego igualoa la variable “tri” a “x”, con el siguiente script: **x= tri**;

Y ejecuto el primer script de la implementación del sistema, y obtengo la siguiente figura:



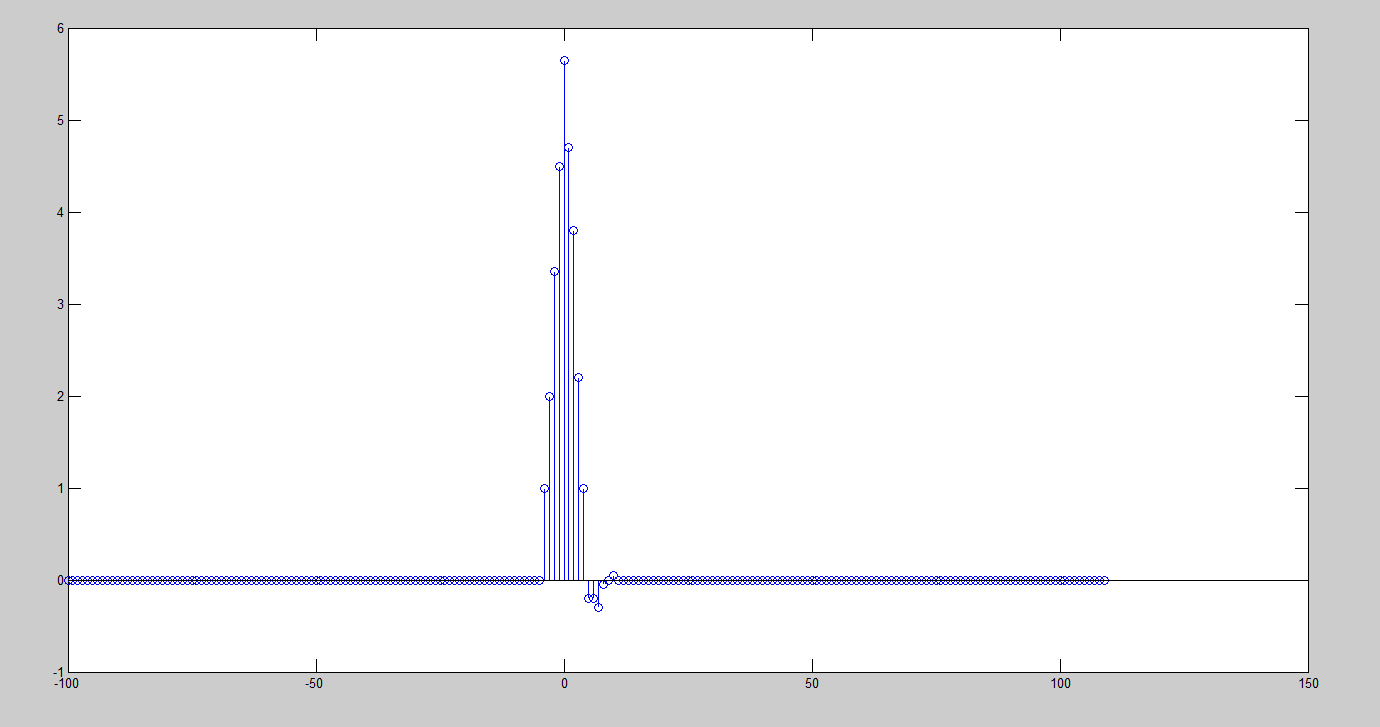
1. Para la convolución de las dos señales, utilizo el siguiente script:

conv(tri,h)

Creo una variable auxiliar “m”, y le escribo el rango desde –N:N+9, para que coincida con el de la respuesta a la convolución almacenada en la variable “ans”

Luego escribo para graficarla

stem(m, ans) ; Y obtengo la siguiente figura:



Comparándola con la señal anterior, queda demostrado que es la misma señal