

Circuitos Electricos II

Roberto Sanchez Figueroa
brrsanchezfi@unal.edu.co

Monitorias Circuitos II

Soluciones propuestas para los ejercicios del taller 5

Table of Contents

Circuitos Electricos II.....	1
Soluciones propuestas para los ejercicios del taller 5.....	1
Problema 1, solucion analitica y grafica.....	1
Solucion analitica.....	2
Solucion grafica.....	3
Metodo 1 'manual grafico'.....	4
Simulacion.....	6
Problema 2, Analisis	7
Simulacion.....	10
Problema 3.....	11
Simulacion.....	14

Problema 1, solucion analitica y grafica

Problema 1

- 1 Hallar la impedancia de entrada $Z_i(s)$;
- 2 Calcular la frecuencia de resonancia ω_0 .

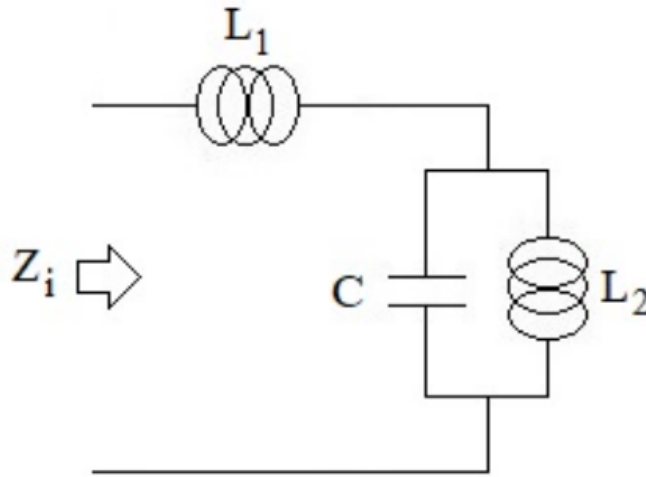


Figure: Circuito LC, propaga valores para los parámetros $\{L_1, L_2, C\}$.

Solucion analitica

```
%declaramos variables
syms L_1 L_2 C s

%asignamos valores, uso valores grandes por conveniencia en el ejercicio 2,
%sin embargo puede usar el codigo para asignar valores cualquiera
C1 =2;
L1 =3;
L2 =5;

%escribimos la impedancia total Z_i ,la funcion paralelo suma las impedancias paralelas.
Z_i = simplify(L_1*s + paralelo((L_2*s), (1/(C*s))))
```

$$Z_i = \frac{s (C L_1 L_2 s^2 + L_1 + L_2)}{C L_2 s^2 + 1}$$

Recordemos que $s = \omega j$

por tanto si operamos solo la parte imaginaria podemos encontrar la frecuencias de corte y por tanto la frecuencia de resonancia.

Apartir de la Impedancia total hallada anteriormente procedemos a despejar s, "si no queda muy claro este punto puede dirigirse al sadiku version 3 pag 630 ecuaciones 14,22 14,23 14,24 que pueden dar una idea mas clara de este punto "

```
%Despejamos s la impedancia total, note que para este punto omega es igual
%a s, recuerde ver la pagina del libro.
w_0 = solve(Z_i,s)
```

Warning: Solutions are only valid under certain conditions. To include parameters and conditions in the solution, specify the 'ReturnConditions' value as 'true'.

$$w_0 = \begin{pmatrix} 0 \\ \frac{\sqrt{-L_1 - L_2}}{\sqrt{C} \sqrt{L_1} \sqrt{L_2}} \\ -\frac{\sqrt{-L_1 - L_2}}{\sqrt{C} \sqrt{L_1} \sqrt{L_2}} \end{pmatrix}$$

```
%sustituimos los valores numericos en las variables simbolicas
w_0 = subs(w_0,[L_1 L_2 C],[L1 L2 C1])
```

$$w_0 = \begin{pmatrix} 0 \\ \frac{\sqrt{2} \sqrt{3} \sqrt{5} \sqrt{8} i}{30} \\ -\frac{\sqrt{2} \sqrt{3} \sqrt{5} \sqrt{8} i}{30} \end{pmatrix}$$

```
%note que tenemos 3 respuestas, sin embargo debemos escoger la respuesta
%coherente a la naturaleza de lo que estudiamos, por tanto escogemos la
%frecuencia \omega positiva.
w_0 = imag(double(w_0))
```

```
w_0 = 3x1
      0
      0.5164
     -0.5164
```

```
W_0 = w_0(2)
```

```
W_0 = 0.5164
```

Solucion grafica

por otra parte , una solucion igual de valida es el analisis grafico del diagrama de bode, sin embargo, es importante reconocer el metodo anterior.

Para este punto empezamos calculando la ecuacion de transferencia del sistema, no sin antes tener presente que dado la simplisidad del sistema propuesto podemos indentificar un divisor de voltaje por el cual "suponemos nuestra senal de salida como el tal divisor", aclaro esto por que nunca se nos da una senal de salida.

```
%A continuacion procedemos a calcular la ecuacion de tranferencia del
%anterior problema.
```

```
H_s = simplify(paralelo((L_2*s),(1/(C*s))) / (paralelo((L_2*s),(1/(C*s))) + L_1*s))
```

H_s =

$$\frac{L_2}{C L_1 L_2 s^2 + L_1 + L_2}$$

```
%sustituimos los valores numericos en las variables simbolicas
```

```
H_s = simplify(subs(H_s, [L_1 L_2 C], [L1 L2 C1]))
```

H_s =

$$\frac{5}{30 s^2 + 8}$$

```
T_s_2 = H_s;
```

```
%convertimos de simbolico a tf
```

```
H_s = sym2tf(H_s, 0)
```

H_s =

$$\frac{5}{30 s^2 + 8}$$

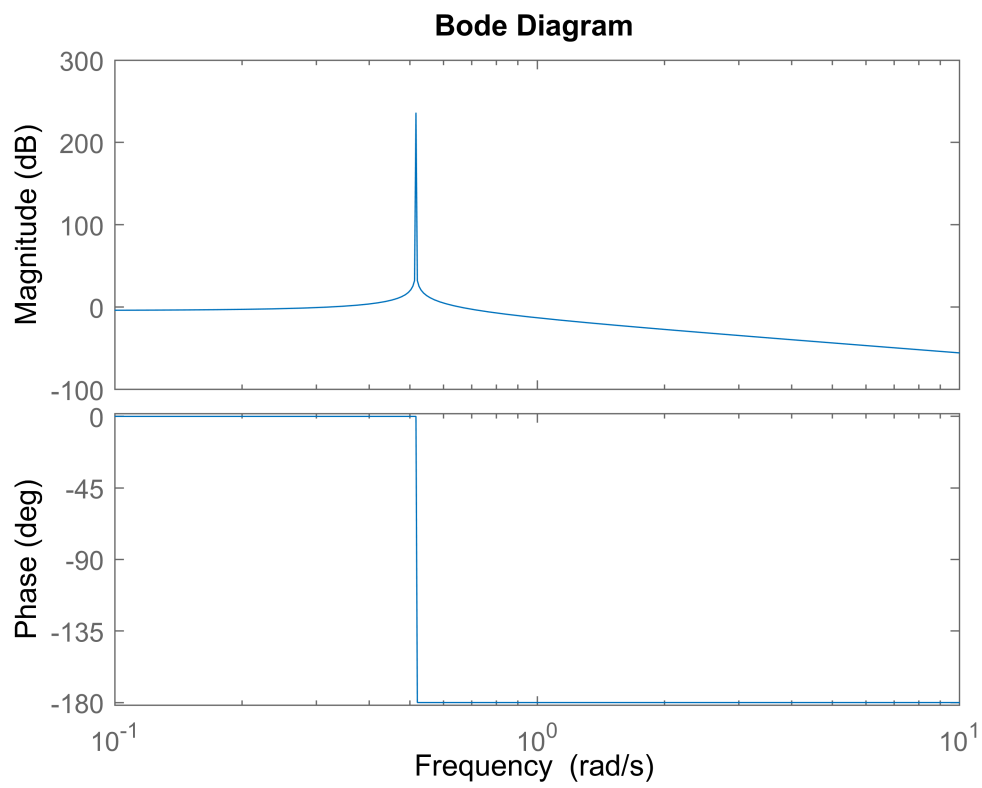
Continuous-time transfer function.

```
%hacemos el posterior analisis grafico por dos metodos distintos
```

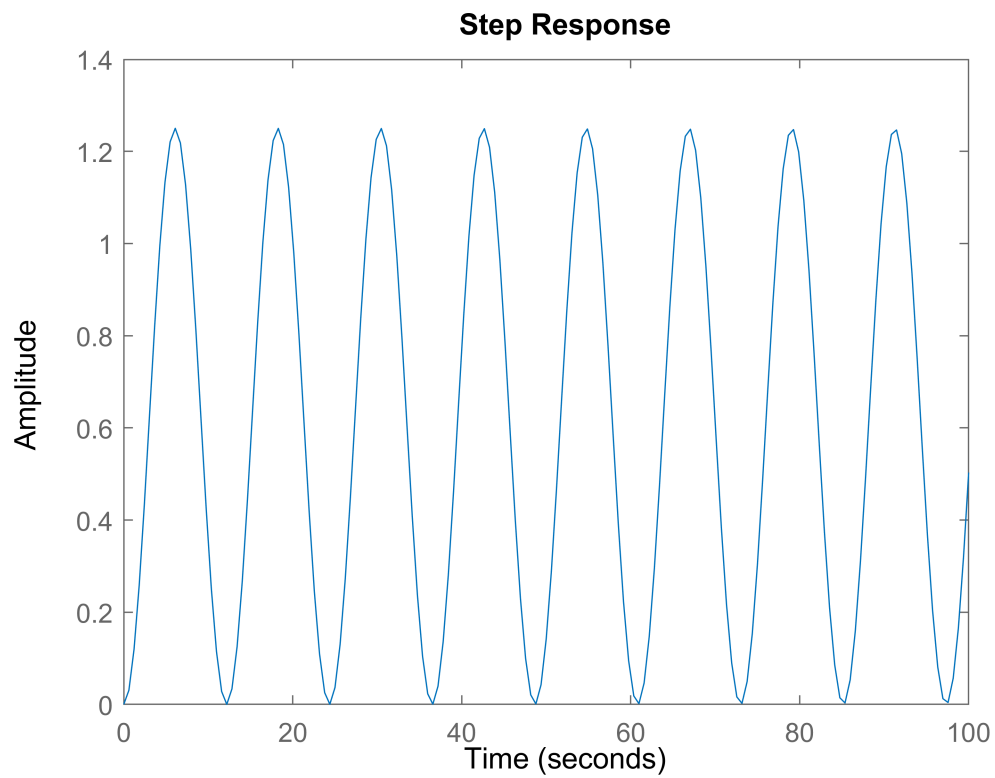
Metodo 1 'manual grafico'

Identifical el pico de resonancia y posterior mente su frecuencia

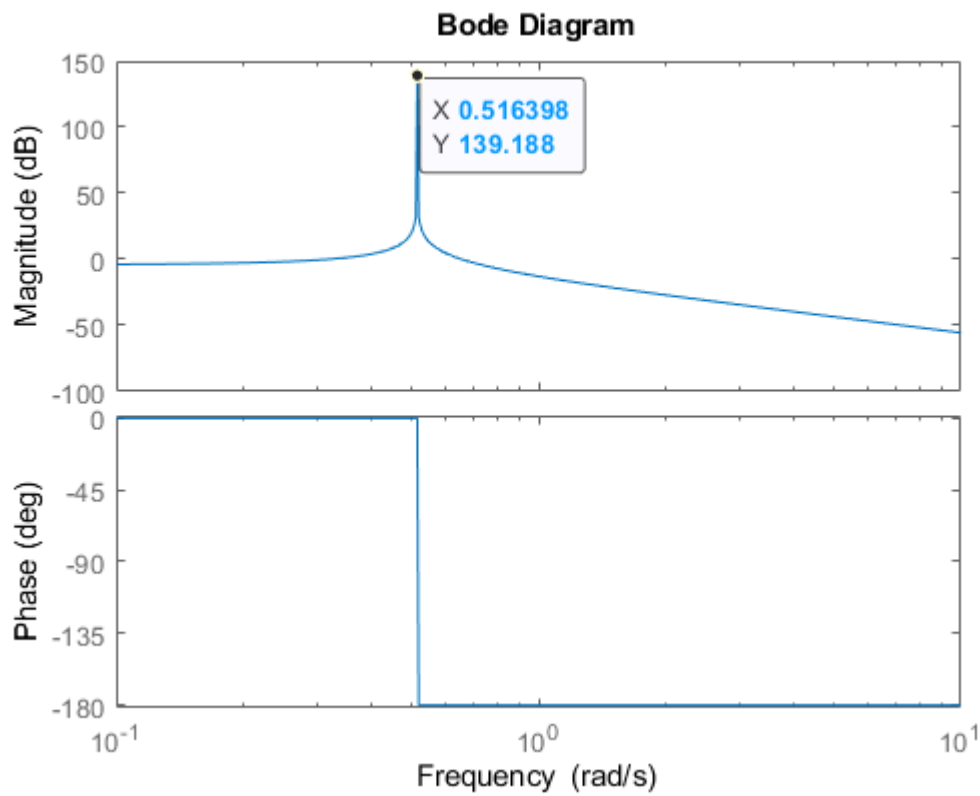
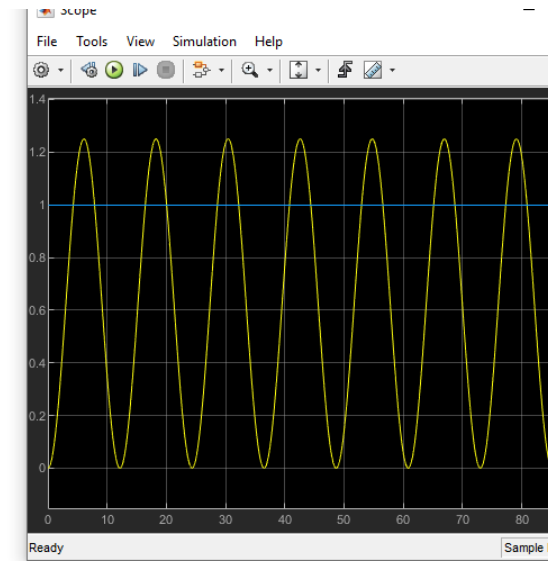
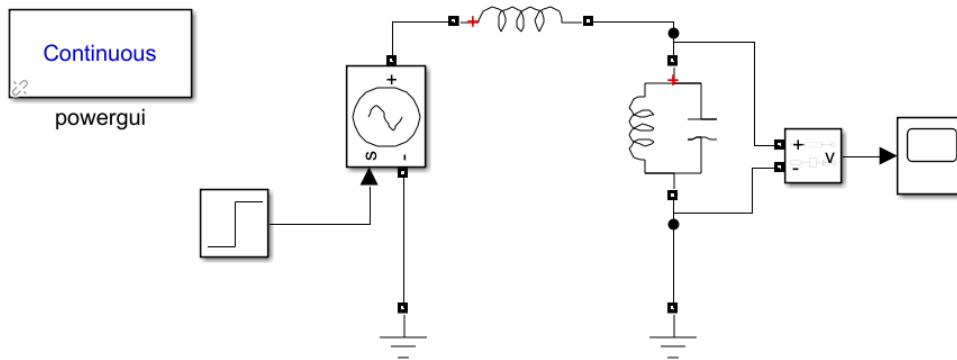
```
bode(H_s);
```



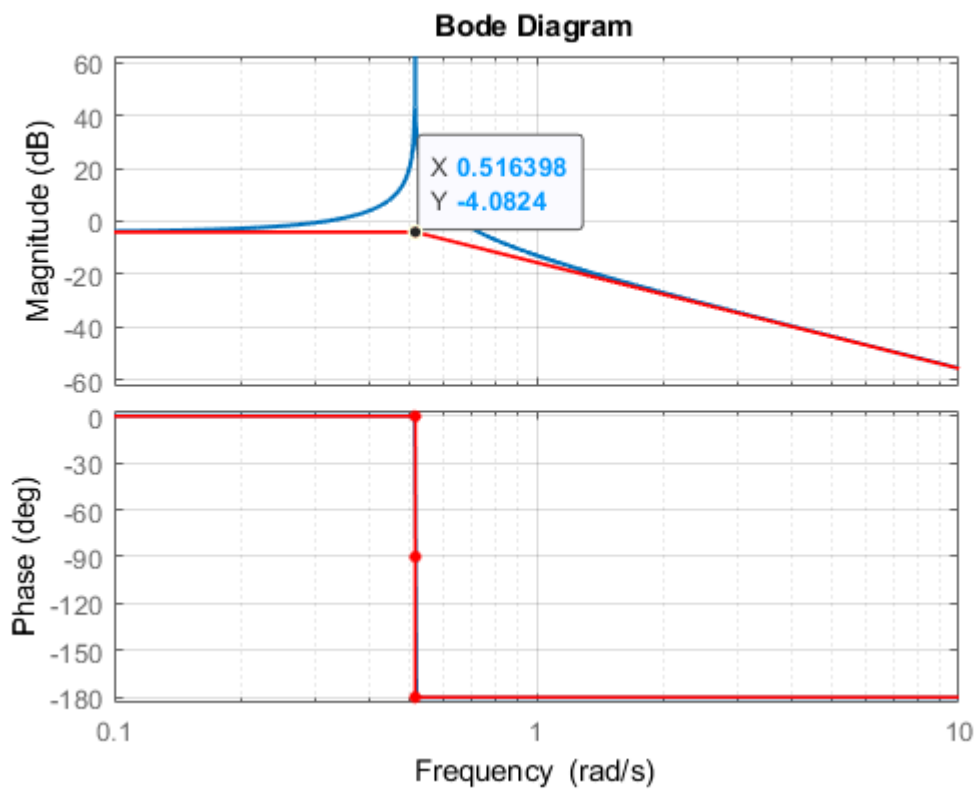
```
step(H_s,100)
```



Simulacion



Este bode corresponde a los valores de



Problema 2, Analisis

Problema 2

Dibujar el voltaje en los distintos nodos cuando la entrada es un impulso unitario, utilice los valores del problema 1.

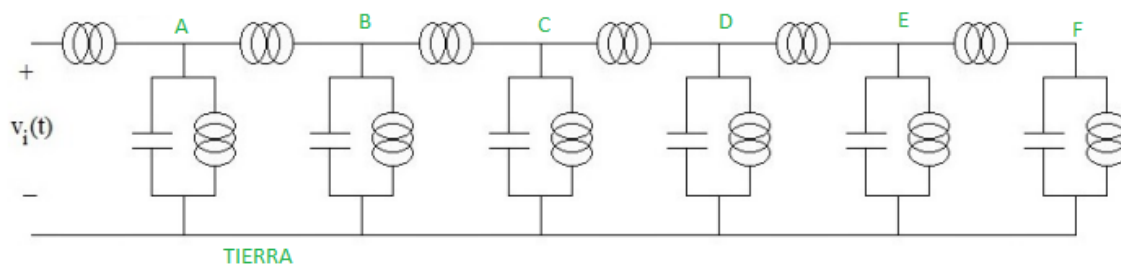


Figure: Línea de transmisión.

Cuál es la frecuencia de resonancia para todo el circuito?

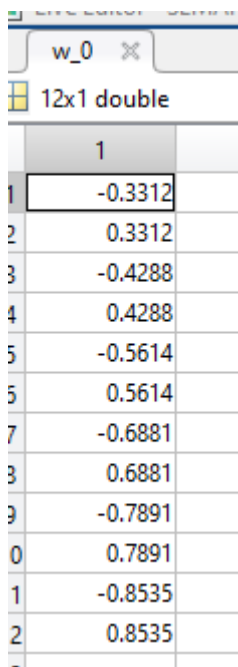
simplificar el circuito calculando el paralelo del capacitor y el inductor 2

```
syms C L_1 L_2

%calculamos las impedancias
C_L2_p = paralelo((1/(C*s)),(L2*s));%f

impedancia_N_E = paralelo(C_L2_p,((L_1*s)+C_L2_p));
impedancia_N_D = paralelo(C_L2_p,((L_1*s)+impedancia_N_E));
impedancia_N_C = paralelo(C_L2_p,((L_1*s)+impedancia_N_D));
impedancia_N_B = paralelo(C_L2_p,((L_1*s)+impedancia_N_C));
impedancia_N_A = paralelo(C_L2_p,((L_1*s)+impedancia_N_B));
impedancia_total = L1*s + impedancia_N_A;
w_0 = subs(impedancia_total,[L_1 L_2 C],[L1 L2 C1]);
w_0 = solve(w_0==0,s);
w_0 = imag(double(w_0))
```

```
w_0 = 12x1
    -0.3312
     0.3312
    -0.4288
     0.4288
    -0.5614
     0.5614
    -0.6881
     0.6881
    -0.7891
     0.7891
     ⋮
```



	1
1	-0.3312
2	0.3312
3	-0.4288
4	0.4288
5	-0.5614
6	0.5614
7	-0.6881
8	0.6881
9	-0.7891
10	0.7891
11	-0.8535
12	0.8535

```
T_s = (impedancia_total);
```



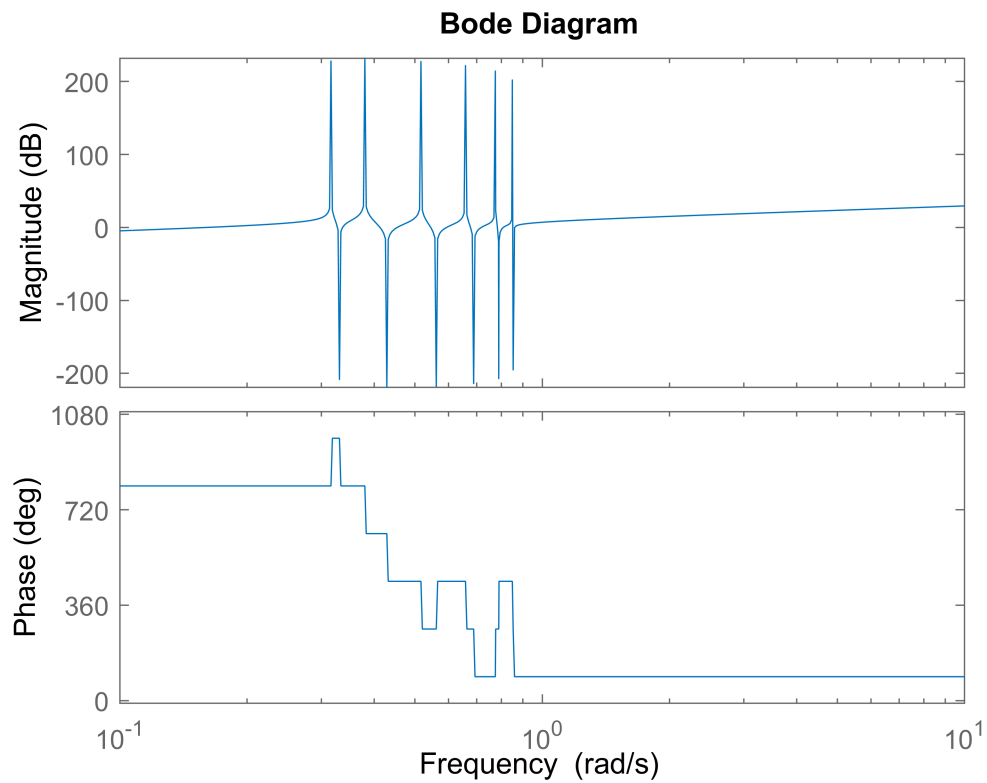
```
T_s = subs(T_s,[L_1 L_2 C],[L1 L2 C1]);
T_s = sym2tf(T_s)
```

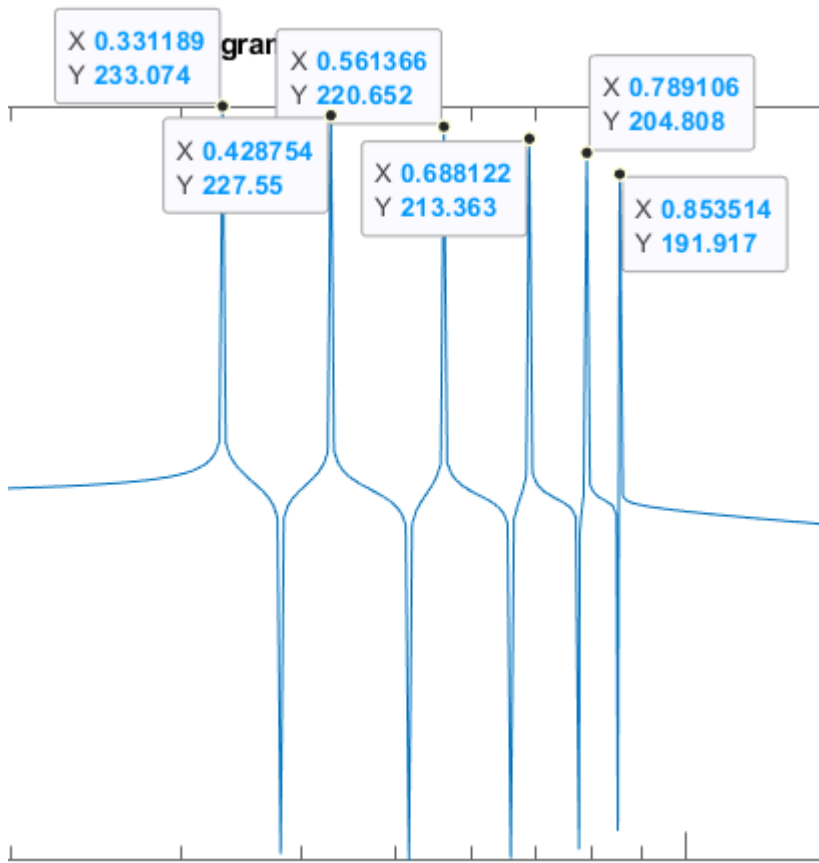
T_s =

$$\frac{7.29e08 s^{13} + 1.774e09 s^{11} + 1.689e09 s^9 + 7.962e08 s^7 + 1.936e08 s^5 + 2.271e07 s^3 + 994969 s}{2.43e08 s^{12} + 5.508e08 s^{10} + 4.82e08 s^8 + 2.056e08 s^6 + 4.446e07 s^4 + 4.579e06 s^2 + 175968}$$

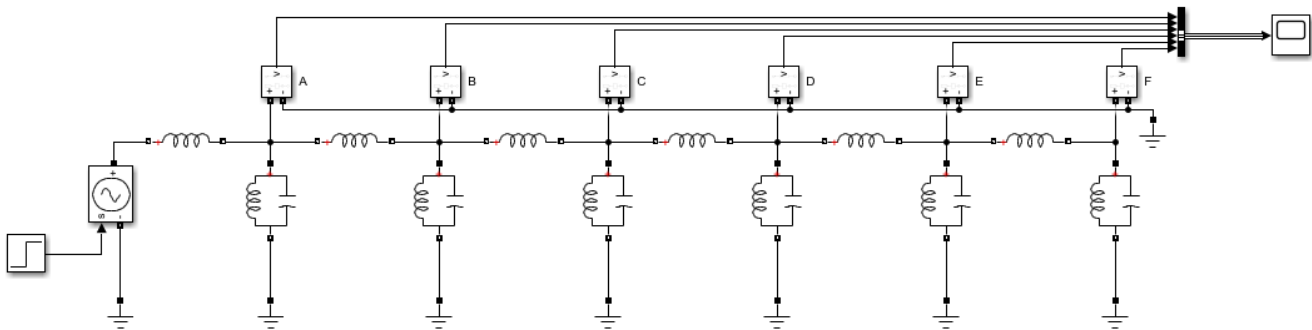
Continuous-time transfer function.

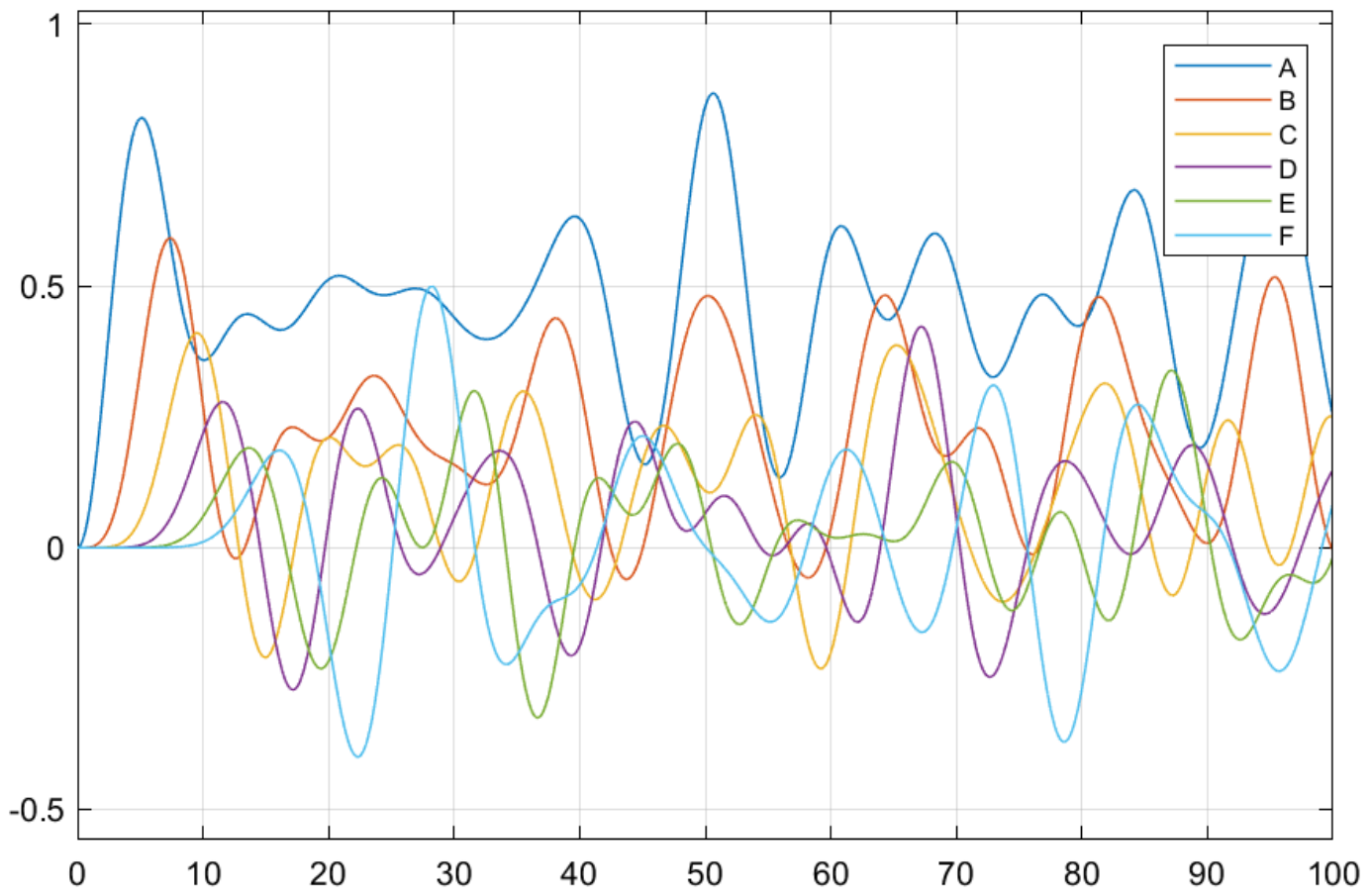
```
bode(T_s)
```





Simulacion





Offset=0

Problema 3

Problema 2

Dibujar el voltaje en los distintos nodos cuando la entrada es un impulso unitario, utilice los valores del problema 1.

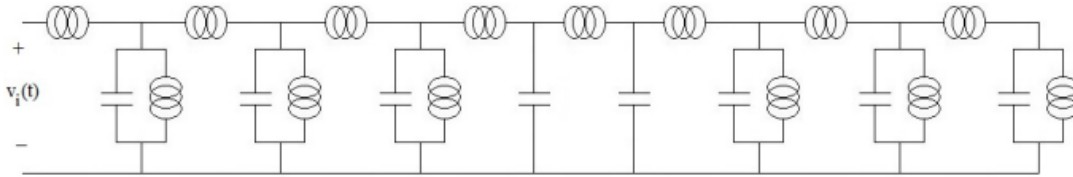
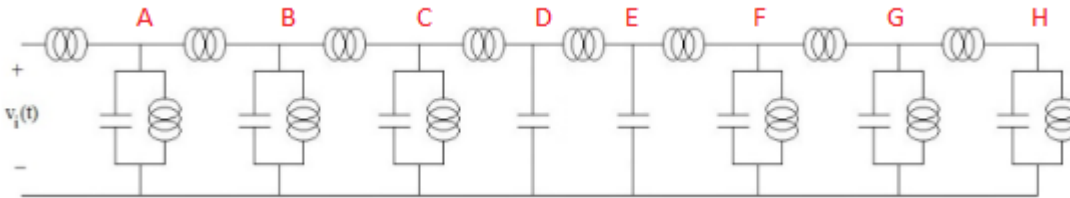


Figure: Línea de transmisión.

Cuál es la frecuencia de resonancia para todo el circuito?

como en el anterior ejercicio simplifíco en paralelo en un equivalente



```
syms C L_1 L_2 s
```

```
C1 =2;
```

```
L1 =3;
```

```
L2 =5;
```

```
C_L2_p = paralelo((1/(C*s)),(L_2*s));
```

```
%IMPEDANCIA EQUIVALENTE DEL SISTEMA
```

```
impedancia_H = C_L2_p;
```

```
impedancia_N_G = paralelo(C_L2_p,((L_1*s)+C_L2_p));
```

```
impedancia_N_F = paralelo(C_L2_p,((L_1*s)+impedancia_N_G));
```

```
impedancia_N_E = paralelo((1/(C*s)),((L_1*s)+impedancia_N_F));
```

```
impedancia_N_D = paralelo((1/(C*s)),((L_1*s)+impedancia_N_E));
```

```
impedancia_N_C = paralelo(C_L2_p,((L_1*s)+impedancia_N_D));
```

```
impedancia_N_B = paralelo(C_L2_p,((L_1*s)+impedancia_N_C));
```

```
impedancia_N_A = paralelo(C_L2_p,((L_1*s)+impedancia_N_B));
```

```
impedancia_total_3 = (L_1*s)+impedancia_N_A;
```

```
w_0 = subs(impedancia_total_3,[L_1 L_2 C],[L1 L2 C1]);
```

```
w_0 = solve(w_0==0,s);
```

```
w_0 = imag(double(w_0))
```

```
w_0 = 16×1  
-0.2487  
 0.2487  
-0.3688  
 0.3688  
-0.4569  
 0.4569  
-0.5656  
 0.5656  
-0.6669  
 0.6669  
⋮  
⋮
```

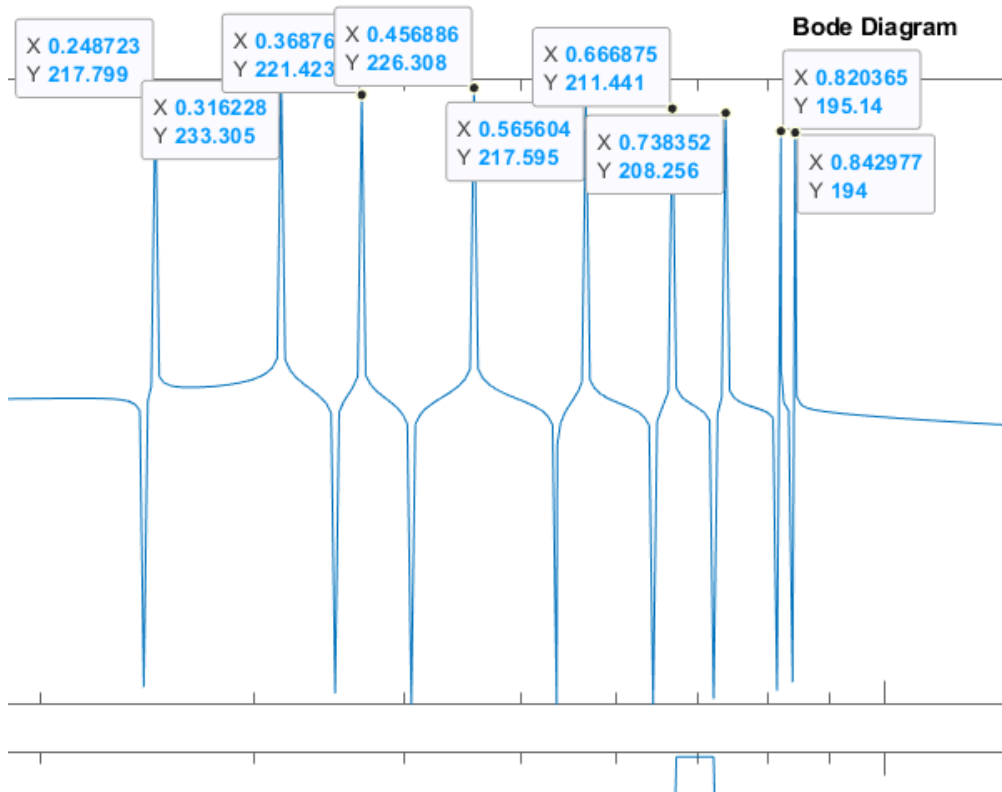
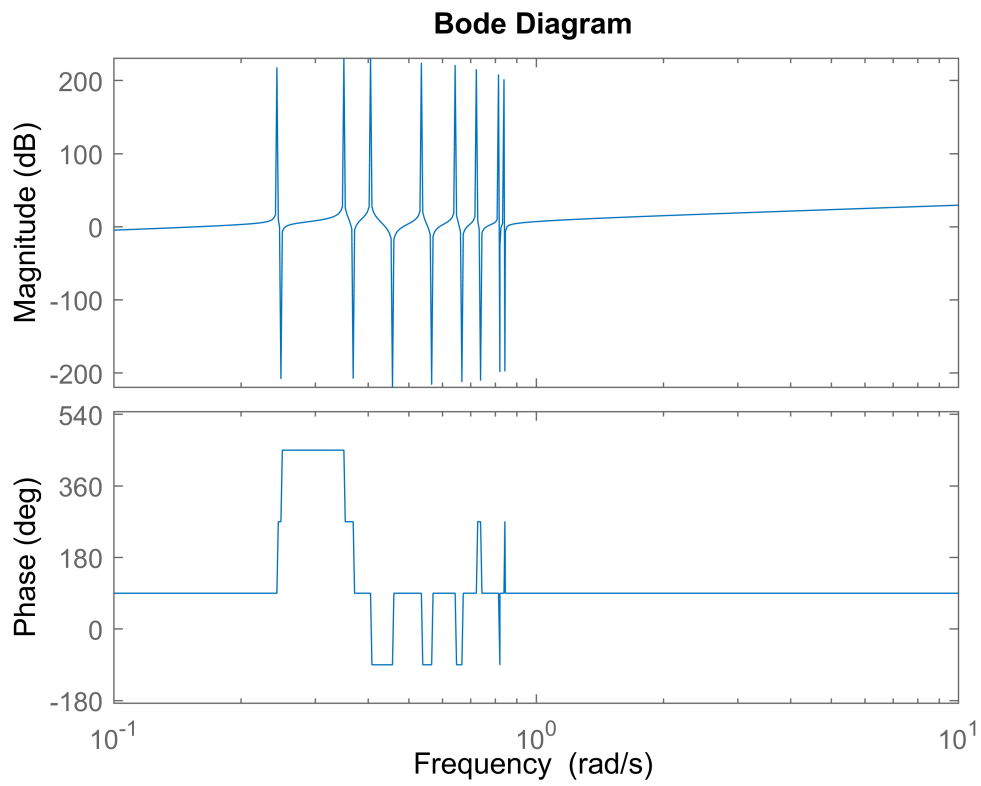
```
T_s = (impedancia_total_3);  
T_s = subs(T_s,[L_1 L_2 C],[L1 L2 C1]);  
T_s = sym2tf(T_s)
```

```
T_s =
```

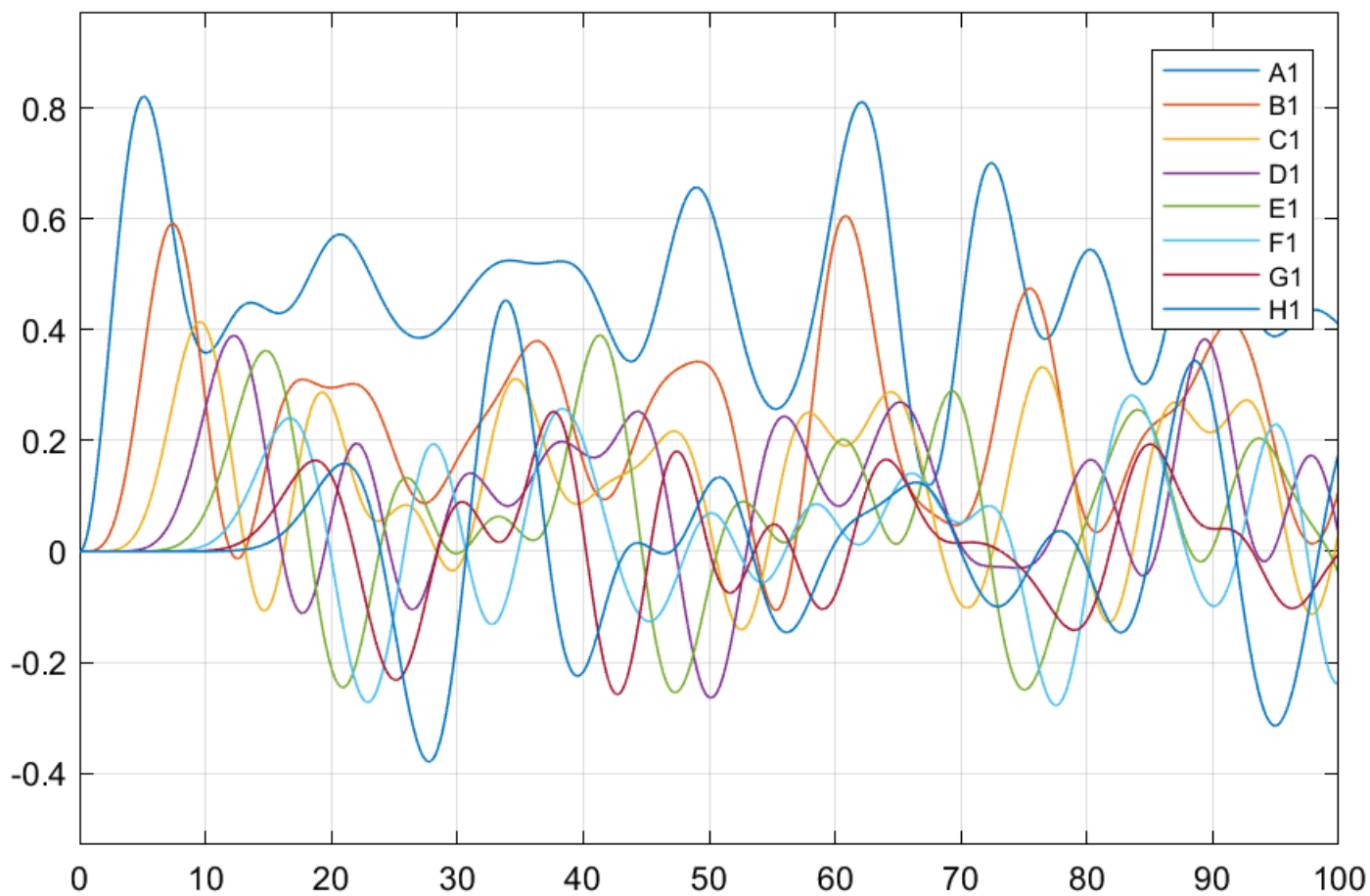
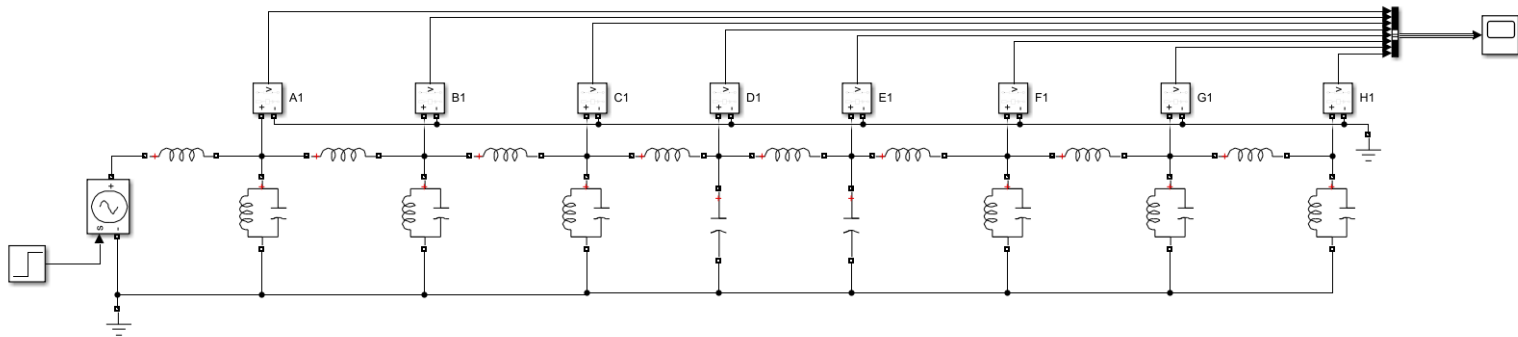
```
2.624e10 s^17 + 8.136e10 s^15 + 1.048e11 s^13 + 7.274e10 s^11 + 2.944e10 s^9 + 7.031e09 s^7 + 9.528e08 s^5  
+ 6.566e07 s^3 + 1.709e06 s  
-----  
8.748e09 s^16 + 2.566e10 s^14 + 3.106e10 s^12 + 2.007e10 s^10 + 7.493e09 s^8 + 1.636e09 s^6 + 2.012e08 s^4  
+ 1.258e07 s^2 + 300384
```

Continuous-time transfer function.

```
bode(T_s)
```



Simulacion



```
function x = paralelo(n1,n2)
    x = (n1*n2)/(n1+n2);
end
```

```

function t_sym = tf2sym(H)
    [num,den] = tfdata(H);
    syms s;
    t_sym = simplify(poly2sym(cell2mat(num),s)/poly2sym(cell2mat(den),s));
end

```

```

function [ tfobj ] = sym2tf( symobj, Ts) %pasa de sym a tf    Ts es el samplin, para continuas
    % SYM2TF convert symbolic math rationals to transfer function

    if isnumeric(symobj)
        tfobj=symobj;
        return;
    end

    [n,d]=numden(symobj);
    num=sym2poly(n);
    den=sym2poly(d);

    if nargin==1
        tfobj=tf(num,den);
    else
        tfobj=tf(num,den,Ts);
    end
end

```