Universidad Nacional Autónoma de México

FES Cuautitlán
Campo IV

"Tutorial para la solución de ejercicios de Teoría de control en Matlab"

Ingeniería Mecánica Eléctrica

Elaborado por:
David Alejandro Lobato López

Con el seguimiento del Ingeniero David Tinoco Varela

INDICE No. De Página

“Tutorial para la solución de ejercicios de Teoría de control en Matlab”

El propósito de este trabajo es dar un seguimiento sobre cómo resolver diferentes tipos de problemas correspondientes a la materia de Teoría de Control y Robótica por medio del uso de software. En este caso se usará “Matlab”. La intención es redactar paso a paso el procedimiento de solución para cada uno de los problemas planteados, es decir, se dirá para qué sirve cada comando utilizado dentro del código de Matlab, así como el por qué de su uso. Cabe mencionar que las soluciones realizadas en este trabajo no serán la única manera de llegar al resultado para los problemas seleccionados, pues cada alumno elige la forma que le resulte más sencilla y fácil de explicar.

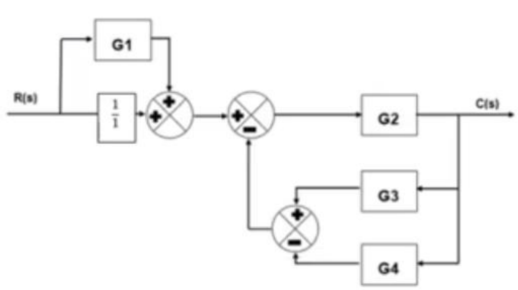
Los temas que se llevarán a cabo son los siguientes:

- Diagramas de bloques.
- Modelo de sistemas en Matlab simulink.
- Transformada y anti transformada de Laplace.
- Fracciones parciales.
- Sistemas de primer orden.
- Sistemas de segundo orden.
- Respuesta en frecuencia.
- Sistemas de Control

Todos estos temas ya han sido vistos previamente en clase, así que ya se sabe cómo solucionarlos de manera teórica. La implementación del uso de software nos será de gran ayuda, ya que la mayoría de las soluciones de estos problemas resultan ser algo complicadas y llegar a ellas nos llevan demasiado tiempo. Con Matlab, la solución es inmediata y sin errores, solamente es necesario saber cómo manipularlo de una forma adecuada para que no haya problemas en la realización del código solución para nuestro problema.

Tema 1: Diagramas de Bloques

Ejercicio 1: Resuelve el siguiente diagrama de bloques:



Donde:

$$G1 = \frac{s+6}{s+3} \qquad G3 = \frac{s+9}{s+4}$$

$$G2 = \frac{s+8}{s^2+6s+2} \qquad\qquad G4 = \frac{9}{s^2+s+8}$$

Primeramente, para introducir los valores de G1, G2, G3 y G4 en Matlab, debemos solicitar al usuario que indique los dos valores de cada una de estas variables haciendo uso de del comando input, uno para el numerador (usando el comando num) y otro para el denominador (usando el comando den). Y dentro del código nos quedaría de la siguiente manera:

>>num1=input ('Introduzca el valor del numerador de G1');
>> den1= input ('Introduzca el valor del numerador de G1');

Suponiendo que num1 es la variable de nuestro numerador de G1 y den1 es la varaible designada para nuestro denominador de G1.

Haciendo este procedimiento para todas las variables puede resultar largo, es por eso que para éste caso, no usaremos variables, sino constantes. Esto quiere decir que nosotros (el programador) vamos a designar el valor directamente de cada constante sin solicitar esos datos al usuario. Esto para que resulte más sencillo y rápido. Entonces nos quedaría de la siguiente manera:

Para ingresar $G1 = \frac{s+6}{s+3}$

Como se da por entendido que, en este tipo de ejercicios los valores de las G están compuestos por fracciones formadas por polinomios donde la variable es la letra "s", dentro de Matlab no será necesario poner la "s", basta con poner el coeficiente de ésta. Todas estas expresiones deben tener un término independiente el cual debe introducirse y si no lo hay, se debe de indicar con un 0.

Para ingresar esto a Matlab se deberá hacer de la siguiente manera:

Supongamos que se quiere ingresar el numerador de G1:

$$s+6$$

Donde los coeficientes se muestran marcados en azul:

$$\mathbf{1}s+\mathbf{6}$$

En Matlab deberíamos ingresar solamente los coeficientes entre corchetes, en ese mismo orden y separados uno de otro por medio de un espacio. Es decir:

[1 6]

Al final de cada renglón, se debe de poner un ; excepto cuando el renglón trata alguna operación a realizar, en ese caso no es necesario poner un ; al final de éste.

Entonces, para $G1 = \frac{s+6}{s+3}$ sabemos que el numerador es $s+6$ y el denominador es $s+3$, lo cual se va a escribir de la siguiente manera:

>>num1= [1 6]; → Indicando que es el numerador de G1 .
>>den1=[1 3]; → Indicando que es el denominador de G1.

Sabiendo esto, ya podemos ingresar los valores de G1, G2, G3 y G4, de la siguiente manera:

Este mismo procedimiento se realizara en los siguientes valores.

Para $G2 = \frac{s+8}{s^2+6s+2}$:

>>num2= [1 8];   → Indicando que es el numerador de G2.
>>den2=[1 6 2];   → Indicando que es el denominador de G2.

Para $G3 = \frac{s+9}{s+4}$:

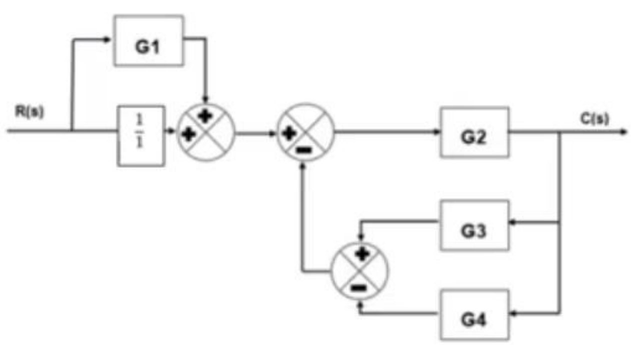
>>num2= [1 9];   → Indicando que es el numerador de G3.
>>den2=[1 4];   → Indicando que es el denominador de G3.

Para $G4 = \frac{9}{s^2+s+8}$:

>>num2= [-9];   → Indicando que es el numerador de G4.
>>den2=[1 1 8];   → Indicando que es el denominador de G4.

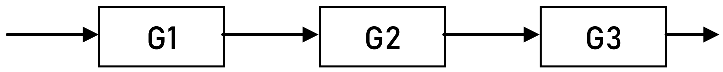
Nota: El numerador de G4 se puso con un signo negativo debido a que, si observamos el diagrama de bloques, podemos ver que entra a una resta, entonces, es por eso que hacemos negativa la expresión de G4.

Sabemos que para ir resolviendo estos diagramas, es necesario ir simplificando poco a poco. En Matlab sucede lo mismo, debemos ir indicando que pasos son necesarios hacer para reducir el diagrama a su forma más simple.



Se puede empezar de cualquier punto, esto no va a afectar el proceso. Recordemos que se busca que al final solo quede el "camino" que une la entrada con la salida, así que lo que nos interesa simplificar son todos los "caminos" que se derivan de este "camino" principal. Supongamos que lo primero que identificamos es que podemos reducir G3 y G4 por la posición en donde se encuentra. De manera teórica deberíamos recurrir al formulario y realizar las operaciones señaladas. En Matlab es mucho más fácil y antes de explicar cómo, es necesario que sepamos los comandos de algunos diagramas básicos:

Cuando tengamos el siguiente comportamiento:



Usaremos el comando series y también tendremos que usar al menos dos variables más para este caso (pues este comando va simplificando pares de bloques y en el ejemplo tenemos tres bloques) éstas comúnmente son llamadas H. Es decir, H1 será asignado para la operación entre G1 y G2 y después de obtener el resultado, se usará H2 para la operación entre H1 y G3, para de esta manera llegar a la solución de éste diagrama.

Dentro de Matlab, éste comando para este ejemplo se usaría de la siguiente manera (considerado que los valores de los numeradores y denominadores ya fueron declarados).

```
>> [num4,den4]=series(num1,den1,num2,den2);
>> H1=tf(num4,den4)
```

Aquí, Matlab nos arrojaría el valor calculado de H1.

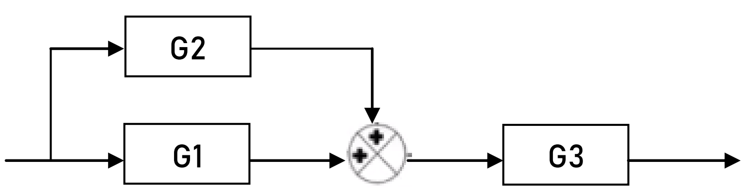
```
>> [num5,den5]=series(num3,den3,num4,den4);
>> H2=tf(num5,den5)
```

Aquí, Matlab nos arrojaría el valor calculado de H2, que es la solución para este ejemplo.

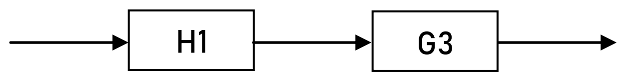
Como se puede observar, para el numerador y denominador de H1, se utilizaron las variables num4 y den4. La razón es la siguiente: como estamos suponiendo que los valores de las constantes de los numeradores y denominadores de G1, G2 y G3 ya fueron designadas con num1, den1, num2, den2, num3, den3 respectivamente, lo más recomendado es seguir un orden para evitarnos errores nuestros al momento de introducir los datos. Esto quiere decir que el numerador y denominador de H1 podrían ser num7 y den7 ó num46 y den 46, no importa qué número utilicemos, solo debemos ser cuidadosos en que esa variable no se haya usado antes. Esto mismo aplica para H2.

También se puede ver que se utiliza otro comando: tf, que es el responsable de obtener la función de transferencia entre los bloques indicados.

También se pueden tener otros comportamientos, como por ejemplo:



Para este caso usaremos el comando parallel para resolver la operación que se encuentra entre G1 y G2. Resolviendo esta parte del diagrama, nos quedaría algo asi:



Para lo que sería necesario hacer uso del comando series para poder llegar a la solución del diagrama.

Dentro de Matlab, nos quedaría de la siguiente manera:

>> [num4,den4]=parallel(num1,den1,num2,den2);
>> H1=tf(num4,den4)

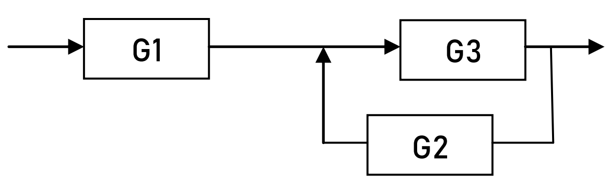
Aquí, Matlab nos arrojaría el valor calculado de H1.

>> [num5,den5]=series(num3,den3,num4,den4);
>> H2=tf(num5,den5)

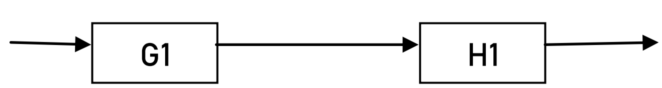
Aquí, Matlab nos arrojaría el valor calculado de H2, que es la solución para este ejemplo.

Las variables H1 y H2, como en el ejercicio anterior, también son propuestas, pero realmente podemos usar cualquier variable de nuestra elección sin que se repitan.

Y el último comportamiento que se nos puede presentar es el siguiente:



En este caso, para simplificar la parte de los bloques G2 y G3, se usa el comando feedback, el cuál realizara la operación entre G2 y G3. Haciendo esto nos quedaría el siguiente diagrama:



En donde usaríamos el comando series para llegar a la solución del diagrama.

Dentro de Matlab, nos quedaría de la siguiente manera:

>> [num4,den4]=feedback(num2,den2,num3,den3);
>> H1=tf(num4,den4)

Aquí, Matlab nos arrojaría el valor calculado de H1.

```
>> [num5,den5]=series(num3,den3,num4,den4);
>> H2=tf(num5,den5)
```

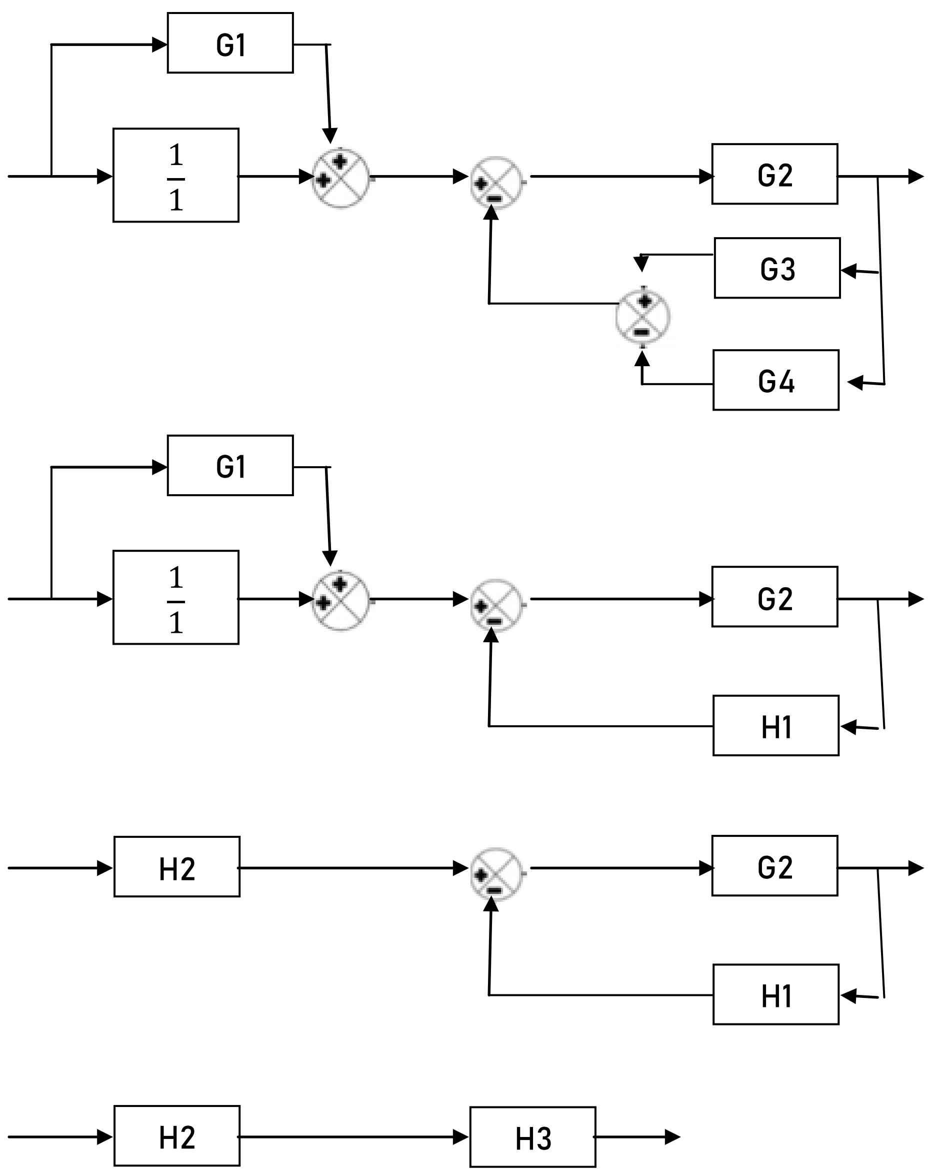
Aquí, Matlab nos arrojaría el valor calculado de H2, que es la solución para este ejemplo.

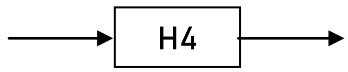
Las variables H1 y H2, como en el ejercicio anterior, también son propuestas, pero realmente podemos usar cualquier variable de nuestra elección sin que se repitan.

Dicho esto, ya podemos resolver el primer ejercicio.

Primero, se mostrara la manera utilizada para simplificar este problema y después los pasos para solucionarlo dentro de Matlab.

Simplificación del diagrama:





Donde:

$$G1 = \frac{s+6}{s+3} \qquad G3 = \frac{s+9}{s+4}$$

$$G2 = \frac{s+8}{s^2+6s+2} \qquad G4 = \frac{9}{s^2+s+8}$$

En Matlab se llevaría a cabo el proceso haciendo usos de los comandos anteriormente mencionados. El código debería ser así:

```
>>num1=[1 6];
>> den1=[1 3];
>> num11=[1];
>> den11=[1];
>> num2=[1 8];
>> den2=[1 6 2];
>> num3=[1 9];
>> den3=[1 4];
>> num4=[-9];
>> den4=[1 1 8];
>> [num5,den5]=parallel(num3,den3,num4,den4);
>> H1=tf(num5,den5)
```

Aquí, Matlab nos arrojaría el valor calculado de H1.

```
>> [num6,den6]=parallel(num1,den1,num11,den11);
>> H2=tf(num6,den6)
```

Aquí, Matlab nos arrojaría el valor calculado de H2.

```
>> [num7,den7]=feedback(num2,den2,num5,den5);
>> H3=tf(num7,den7)
```

Aquí, Matlab nos arrojaría el valor calculado de H3.

```
>> [num8,den8]=series(num6,den6,num7,den7);
>> H4=tf(num8,den8)
```

Aquí, Matlab nos arrojaría el valor calculado de H4.

Siendo el resultado del diagrama de bloques el siguiente:

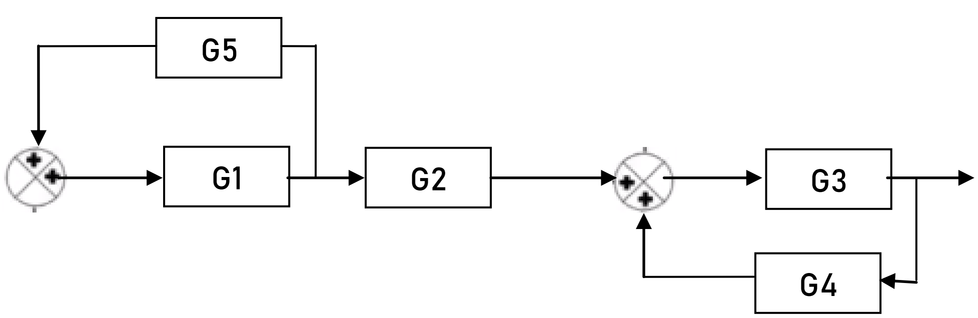
$$\frac{2s^5 + 35\,s^4 + 221s^3 + 724s^2 + 1664s + 2304}{s^6 + 15s^5 + 98s^4 + 388s^3 + 922s^2 + 1300s + 1056}$$

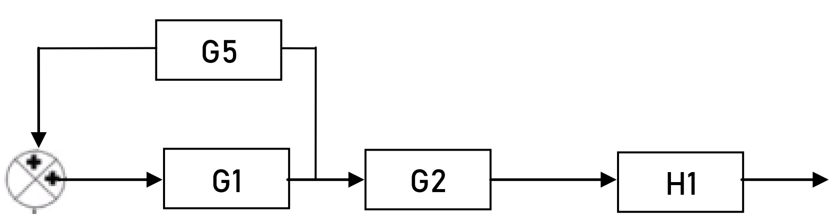
Dentro de Matlab, tendríamos la siguiente vista:

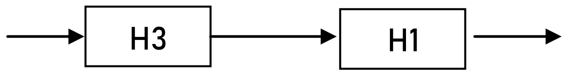
```
Command Window
New to MATLAB? See resources for Getting Started.

>> num1=[1 6];
>> den1=[1 3];
>> num11=[1];
>> den11=[1];
>> num2=[1 8];
>> den2=[1 6 2];
>> num3=[1 9];
>> den3=[1 4];
>> num4=[-9];
>> den4=[1 1 8];
>> [num5,den5]=parallel(num3,den3,num4,den4);
>> H1=tf(num5,den5)

H1 =

  s^3 + 10 s^2 + 8 s + 36
  -----------------------
  s^3 + 5 s^2 + 12 s + 32

Continuous-time transfer function.

>> [num6,den6]=parallel(num1,den1,num11,den11);
>> H2=tf(num6,den6)
```

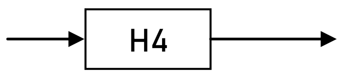
```
H2 =

  2 s + 9
  -------
   s + 3

Continuous-time transfer function.

>> [num7,den7]=feedback(num2,den2,num5,den5);
>> H3=tf(num7,den7)

H3 =

       s^4 + 13 s^3 + 52 s^2 + 128 s + 256
  --------------------------------------------
  s^5 + 12 s^4 + 62 s^3 + 202 s^2 + 316 s + 352

Continuous-time transfer function.

>> [num8,den8]=series(num6,den6,num7,den7);
>> H4=tf(num8,den8)

H4 =

     2 s^5 + 35 s^4 + 221 s^3 + 724 s^2 + 1664 s + 2304
  ----------------------------------------------------------
  s^6 + 15 s^5 + 98 s^4 + 388 s^3 + 922 s^2 + 1300 s + 1056

Continuous-time transfer function.
```

El segundo ejemplo es el siguiente:









Donde:

$$G1 = \frac{s+2}{s+1} \qquad G3 = \frac{s+6}{s^2+3s+1}$$

$$G2 = \frac{s+1}{s+2} \qquad G4 = \frac{9}{2s+1}$$

$$G5 = \frac{s+7}{s^2+s+9}$$

En Matlab se llevaría a cabo el proceso haciendo usos de los comandos anteriormente mencionados. El código debería ser así:

```
>>num1=[1 2];
>> den1=[1 1];
>> num2=[1 1];
>> den2=[1 2];
>> num3=[1 6];
>> den3=[1 3 1];
>> num4=[9];
>> den4=[2 1];
>> num5=[1 7];
>> den5=[1 1 9];
>> [num6,den6]=feedback(num3,den3,num4,den4);
>> H1=tf(num6,den6)
```

Aquí, Matlab nos arrojaría el valor calculado de H1.

```
>> [num7,den7]=feedback(num1,den1,num5,den5);
>> H2=tf(num7,den7)
```

Aquí, Matlab nos arrojaría el valor calculado de H2.

```
>> [num8,den8]=series(num2,den2,num7,den7);
>> H3=tf(num8,den8)
```

Aquí, Matlab nos arrojaría el valor calculado de H3.

```
>> [num9,den9]=series(num6,den6,num8,den8);
>> H4=tf(num9,den9)
```

Aquí, Matlab nos arrojaría el valor calculado de H4.

Siendo el resultado del diagrama de bloques el siguiente:

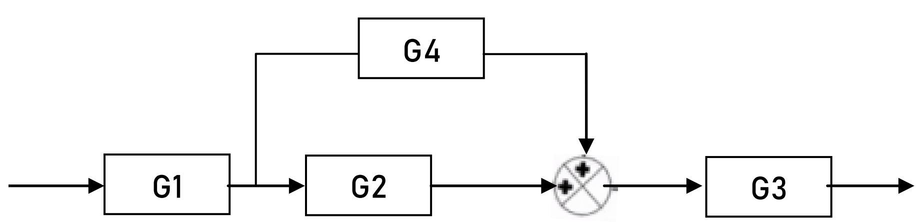
$$\frac{2s^6+21s^5+36\,s^4+264s^3+497s^2+408s+108}{2s^7+17\,s^6+99s^5+422s^4+1144s^3+2551s^2+3999s+2530}$$

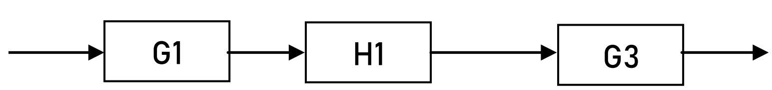
Dentro de Matlab, tendríamos la siguiente vista:

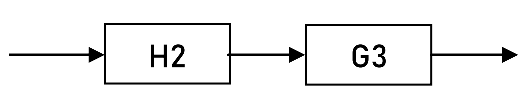
```
Command Window
New to MATLAB? See resources for Getting Started.

>> num1=[1 2];
>> den1=[1 1];
>> num2=[1 1];
>> den2=[1 2];
>> num3=[1 6];
>> den3=[1 3 1];
>> num4=[9];
>> den4=[2 1];
>> num5=[1 7];
>> den5=[1 1 9];
>> [num6,den6]=feedback(num3,den3,num4,den4);
>> H1=tf(num6,den6)

H1 =

      2 s^2 + 13 s + 6
  -------------------------
  2 s^3 + 7 s^2 + 14 s + 55

Continuous-time transfer function.

>> [num7,den7]=feedback(num1,den1,num5,den5);
>> H2=tf(num7,den7)
```

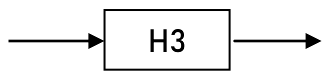
```
H2 =

  s^3 + 3 s^2 + 11 s + 18
  -----------------------
  s^3 + 3 s^2 + 19 s + 23

Continuous-time transfer function.

>> [num8,den8]=series(num2,den2,num7,den7);
>> H3=tf(num8,den8)

H3 =

  s^4 + 4 s^3 + 14 s^2 + 29 s + 18
  --------------------------------
  s^4 + 5 s^3 + 25 s^2 + 61 s + 46

Continuous-time transfer function.

>> [num9,den9]=series(num8,den8,num6,den6);
>> H4=tf(num9,den9)

H4 =

        2 s^6 + 21 s^5 + 86 s^4 + 264 s^3 + 497 s^2 + 408 s + 108
  ---------------------------------------------------------------------
  2 s^7 + 17 s^6 + 99 s^5 + 422 s^4 + 1144 s^3 + 2551 s^2 + 3999 s + 2530

Continuous-time transfer function.
```

El tercer ejemplo es el siguiente:









Donde:

$$G1 = \frac{s+1}{s^2+3s+1} \qquad G3 = \frac{s-3}{s^2+6s+8}$$

$$G2 = \frac{s+4}{s+2} \qquad G4 = \frac{2}{s+5}$$

En Matlab se llevaría a cabo el proceso haciendo usos de los comandos anteriormente mencionados. El código debería ser así:

```
>>num1=[1 1];
>> den1=[1 3 1];
>> num2=[1 4];
>> den2=[1 2];
>> num3=[1 -3];
>> den3=[1 6 8];
>> num4=[2];
>> den4=[1 5];
>> [num5,den5]=parallel(num2,den2,num4,den4);
>> H1=tf(num5,den5)
```

Aquí, Matlab nos arrojaría el valor calculado de H1.

```
>> [num6,den6]=series(num1,den1,num5,den5);
>> H2=tf(num6,den6)
```

Aquí, Matlab nos arrojaría el valor calculado de H2.

```
>> [num7,den7]=series(num3,den3,num6,den6);
>> H3=tf(num7,den7)
```

Aquí, Matlab nos arrojaría el valor calculado de H3.

Siendo el resultado del diagrama de bloques el siguiente:

$$\frac{\boldsymbol{s^4 + 9s^3 - s^2 - 81s - 72}}{\boldsymbol{s^6 + 16s^5 + 100s^4 + 309s^3 + 488s^2 + 356s + 80}}$$

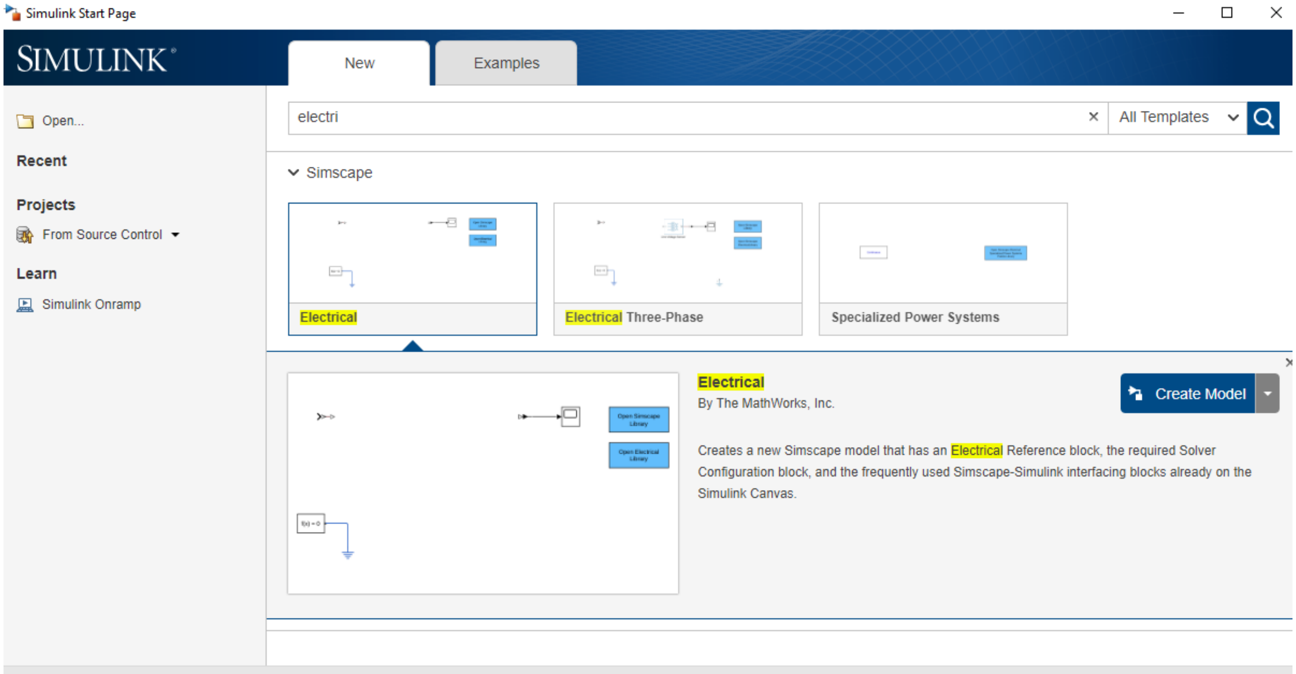
Dentro de Matlab, tendríamos la siguiente vista:

```
Command Window
New to MATLAB? See resources for Getting Started.
>> num1=[1 1];
>> den1=[1 3 1];
>> num2=[1 4];
>> den2=[1 2];
>> num3=[1 -3];
>> den3=[1 6 8];
>> num4=[2];
>> den4=[1 5];
>> [num5,den5]=parallel(num2,den2,num4,den4);
>> H1=tf(num5,den5)

H1 =

  s^2 + 11 s + 24
  ---------------
  s^2 + 7 s + 10

Continuous-time transfer function.
>> [num6,den6]=series(num1,den1,num5,den5);
>> H2=tf(num6,den6)

H2 =

       s^3 + 12 s^2 + 35 s + 24
  ---------------------------------
  s^4 + 10 s^3 + 32 s^2 + 37 s + 10

Continuous-time transfer function.

>> [num7,den7]=series(num6,den6,num3,den3);
>> H3=tf(num7,den7)

H3 =

                s^4 + 9 s^3 - s^2 - 81 s - 72
  --------------------------------------------------------
  s^6 + 16 s^5 + 100 s^4 + 309 s^3 + 488 s^2 + 356 s + 80

Continuous-time transfer function.
```

Tema 2: Modelo de sistemas en Matlab-Simulink

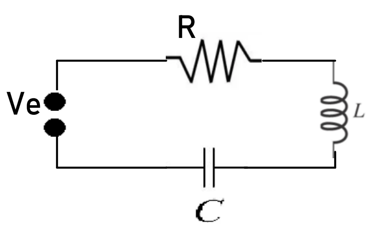
Simulink nos ofrece una gran cantidad para simular sistemas de todo tipo, en este caso, solamente usaremos el simulador para circuitos eléctricos. Podremos obtener el modelo del comportamiento de cada uno de los circuitos, así como realizar mediciones en diferentes puntos del circuito con posibilidad de graficar.

Para ingresar a esta modalidad de Matlab, primero debemos hacer click en el ícono se Simulink y se abrirá una ventana llamada “Simulink Start Page”. Dentro de ésta ventana, para encontrar el modelo de sistema que vamos a utilizar es necesaria utilizar el buscador. En éste escribiremos la palabra “electrical” y seleccionamos la primer opción.



Posteriormente, se abrirán dos ventanas nuevas, una donde podremos armar los circuitos eléctricos y otra que es la librería, de donde vamos a obtener los diferentes elementos a utilizar en la construcción del circuito.

Para ejemplificar este tema, se armaran tres circuitos diferentes. El primero es un circuito RLC:



En donde:
Frecuencia= 60 Hz
R= 5Ω
L=3.4 H
C=1 x 10^{-3} F
Ve= 12V

Lo primero que se tiene que hacer es definir la ecuación modelo, este caso no resulta tan difícil, pues es solamente una malla, para ello se debe hacer el siguiente análisis:

$$Vi(t) = Ri(t) + L\frac{di}{dt} + \frac{q}{C} \text{..... (1)}$$

Sabemos que:

$$i = \frac{dq}{dt} \text{ .... (2)}$$

Entonces:

$$\frac{di}{dt} = \frac{d^2q}{dt^2} \text{ ..... (3)}$$

Sistituyendo (2) y (3) en (1):

$$Vi(t) = R\,\frac{dq}{dt} + L\,\frac{d^2q}{dt^2} + \frac{q}{C}$$

Despejamos $\frac{d^2q}{dt^2}$:

$$\frac{d^2q}{dt^2} = \left(Vi(t) - R\frac{dq}{dt} - \frac{q}{C}\right)\frac{1}{L}$$

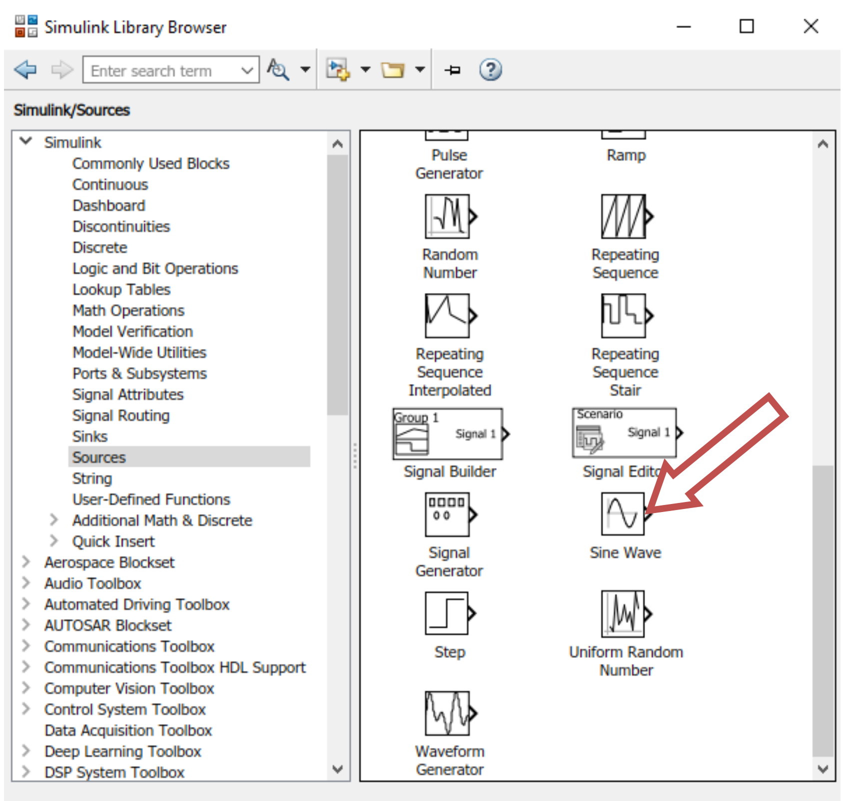
$$\frac{d^2q}{dt^2} = \left(\frac{Vi(t)}{L} - \frac{R}{L}\frac{dq}{dt} - \frac{q}{CL}\right) \quad \dots (4)$$

Sustituyendo los valores en (4):

$$\frac{d^2q}{dt^2} = \left(\frac{12V}{3.5\,H} - \frac{5\Omega}{3.5\,H}\frac{dq}{dt} - \frac{q}{(1x10^{-3}F)(3.5\,H)}\right)$$

Esta última ecuación es con la que se va a trabajar dentro de Simulink. Recordemos que todos los componentes los vamos a obtener de la librería del mismo programa.

Entonces, primeramente vamos a buscar la fuente de voltaje, que en este caso es de corriente alterna de 12V. En la librería la encontraremos dentro de "Sources" con el nombre de "Sine Wave" ya que, al ser alterna, nos arroja una señal senoidal



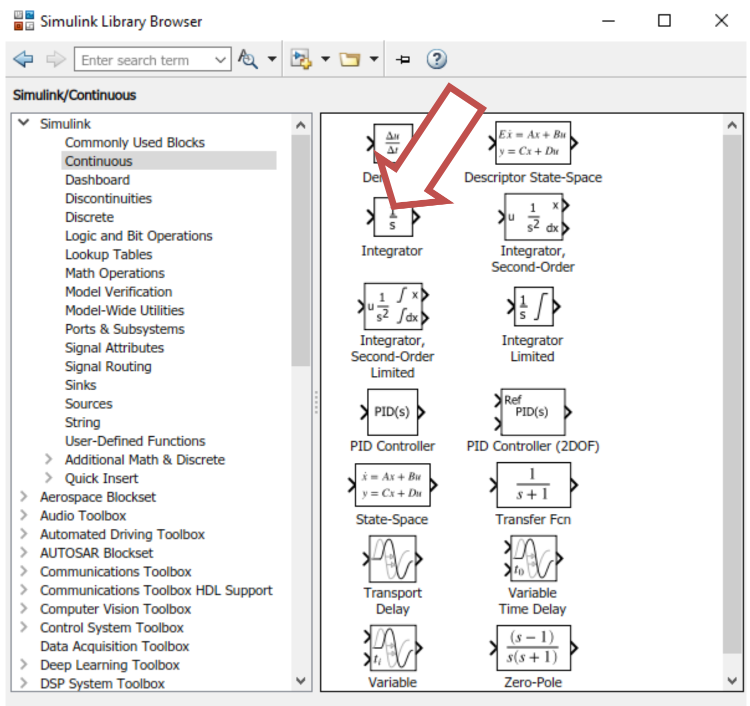
Arrastramos el elemento a la ventana de creación del circuito y daremos doble click para ajustar el valor del voltaje que vamos a utilizar (12/3.5) en el apartado de “Amplitude”.

Podemos ver que dentro de la expresión que vamos a usar se realizan una suma y dos restas:

$$\frac{d^2q}{dt^2} = \left(+ \frac{12V}{3.5\ H} - \frac{5\Omega}{3.5\ H}\frac{dq}{dt} - \frac{q}{(1x10^{-3}F)(3.5\ H)} \right)$$

Para ello, vamos a buscar un sumador ubicado en “Commonly Used Blocks” con el nombre de “Sum”. Una vez encontrado lo vamos a arrastrar hasta la ventana de creación del circuito y le daremos doble click. Se nos abrirá su menú y en el apartado de “List of signs” colocaremos los signos que necesitamos, en este caso son dos restas y una suma, entonces lo colocaremos de la siguiente manera: |+| --

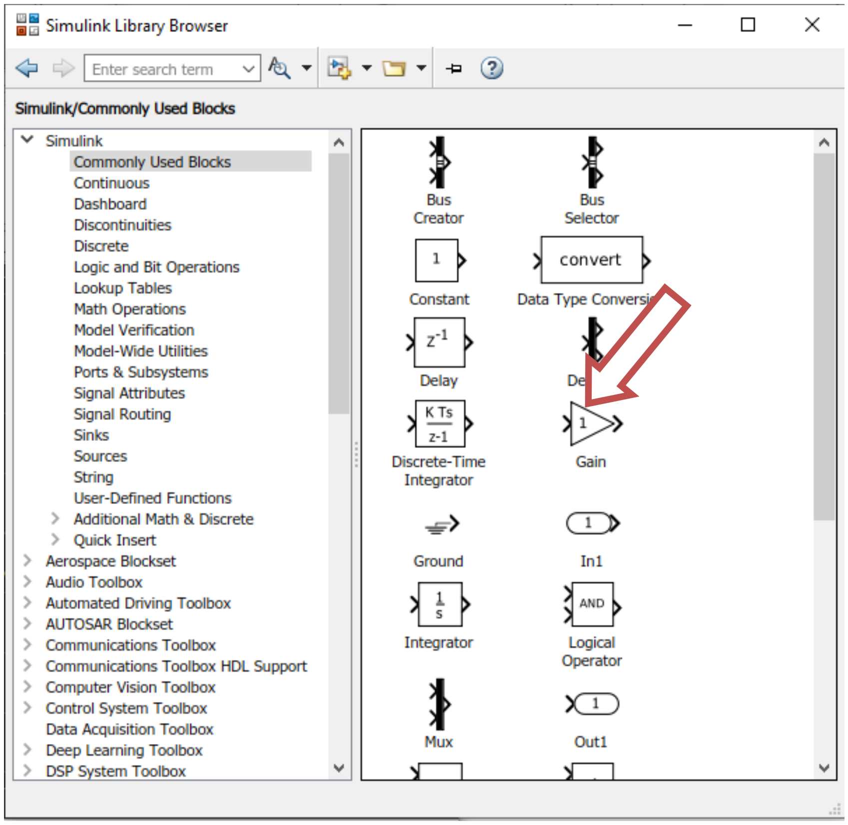
También será necesario colocar un integrador para obtener $\frac{dq}{dt}$ a partir de $\frac{d^2q}{dt^2}$
Éste lo podemos encontrar dentro de la librería, seleccionando”Continuous” y posteriormente “integrator” el cual arrastraremos hasta la ventana de creación del circuito. En la entrada de el integrador debe ir la segunda derivada $\frac{d^2q}{dt^2}$, para que a la salida de éste tengamos la primer derivada $\frac{dq}{dt}$.



Hasta este momento, si partimos de la ecuación del circuito, esto es lo que ya hemos incluido en nuestro circuito dentro de Simulink (marcado con rojo):

$$\frac{d^2q}{dt^2} = \left(+ \frac{12V}{3.5\ H} - \frac{5\Omega}{3.5\ H}\frac{dq}{dt} - \frac{q}{(1x10^{-3}F)(3.5\ H)} \right)$$

Para colocar los datos faltantes, los vamos a incluir con un elemento llamado “Gain” dento de la librería en el apartado de “Commonly Used Blocks”.



En este elemento se va a incluir el dato $\frac{5\Omega}{3.5\,H}$, esto se realiza haciendo doble click sobre el elemento y en el apartado de "Gain" colocar la operación correspondiente: 5/3.5.

Observemos en que lugar de la ecuación encontramos este dato:

$$\frac{d^2q}{dt^2} = \left(+ \frac{12V}{3.5\,H} - \frac{5\Omega}{3.5\,H}\frac{dq}{dt} - \frac{q}{(1x10^{-3}F)(3.5\,H)} \right)$$

Podemos ver qué está restando y se encuentra multiplicando a la primer derivada. Entonces este elemento se va a conectar a una entrada negativa del sumador y a la salida del integrador. Hasta este momento llevaríamos esto (en rojo):

$$\frac{d^2q}{dt^2} = \left(+ \frac{12V}{3.5\,H} - \frac{5\Omega}{3.5\,H}\frac{dq}{dt} - \frac{q}{(1x10^{-3}F)(3.5\,H)} \right)$$

Para obtener esta expresión $\frac{q}{(1x10^{-3})(3.5\,H)}$ Es necesario colocar otro integrador a la salida del integrador que ya habíamos colocado anteriormente, pues al integrar $\frac{dq}{dt}$ el resultado es q y es justamente lo que necesitamos para esa expresión. El integrados lo podemos copiar y pegar o bien, buscarlo de nuevo en la librería y arrastrarlo hasta la ventana de creación del circuito.

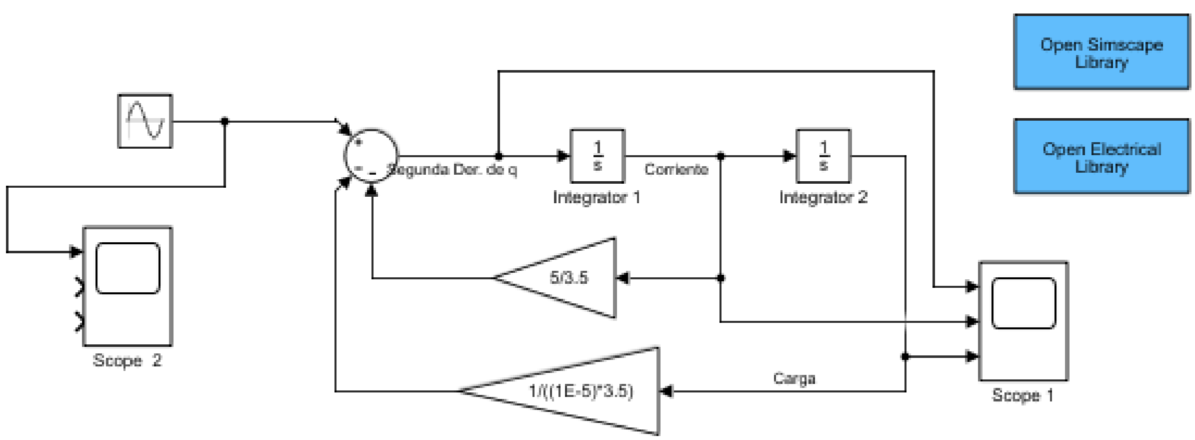
De la salida de este segundo integrador, se va a añadir un segundo elemento “Gain” en donde, se colocara la expresión deseada: 1/((1E-3)*3.5). De la salida de este elemento “Gain” se conectará a la otra resta del sumador, pues en la ecuación se encuentra restando.

Ya tenemos todos los elementos, solamente haría falta conectar la entrada de voltaje que colocamos en el primer paso, a la entrada positiva del sumador.

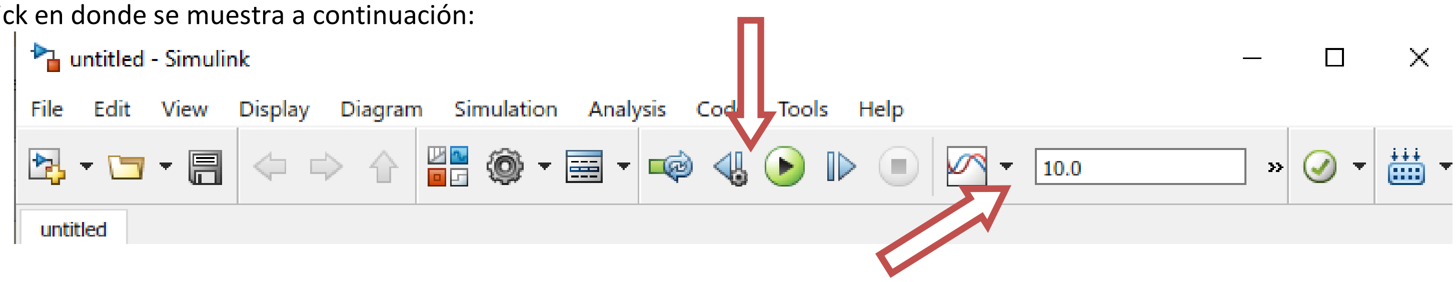
El objetivo de este armado es de poder obtener las gráficas características de cada elemento, en este caso se van a obtener la gráfica del término $\frac{d^2q}{dt^2}$ principalmente, pero también vamos a obtener la gráfica de la corriente y de la carga. Esro será posible colocando un Osciloscopio, seleccionándolo en la librería de Simulink “Commonly Used Blocks” con el nombre de “Scope”. Haciendo doble click en él se abrirá una ventana con sus parámetros. En esa ventana vamos a editar el apartado de “Nuber of axes=1” a “Nuber of axes=3” porque necesitamos obtener tres diferentes graficas, como se meniocó anteriormente.

Tambien se podría inlcuir un segundo osciloscopio conectado a la entrada, es decir, después del elemento “Sine Wave”, con la finalidad de revisar la señal de entrada que está alimentando al circuito.

Nuestro circuito deberá verse de la siguiente manera:

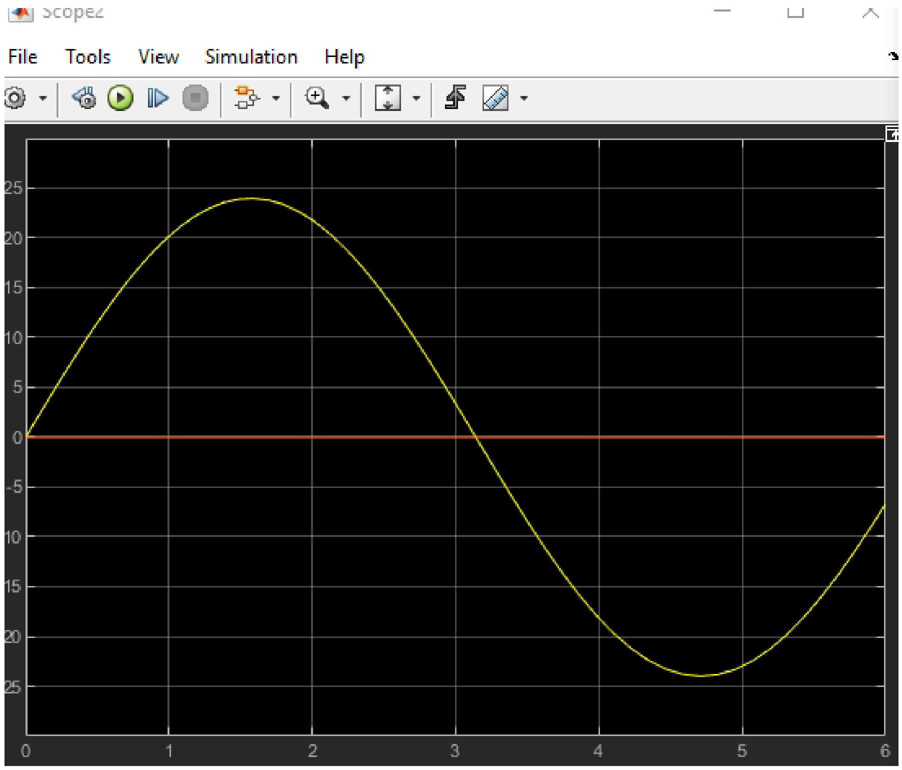


No le hemos dado un valor definido a la variable t que corresponde al tiempo, este valor podemos elegirlo nosotros haciendo click en donde se muestra a continuación:

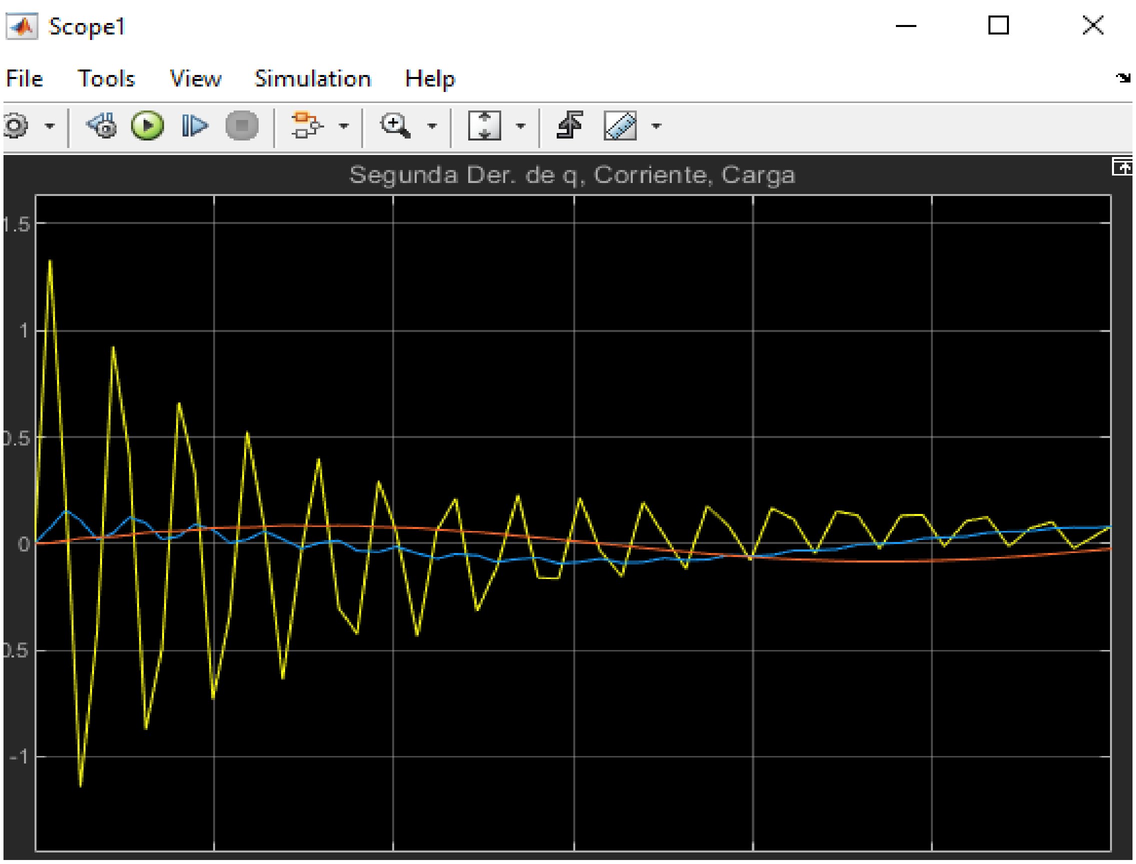


Por último damos al botón de inicio y el programa nos mostrará las graficas de nuestro circuito.

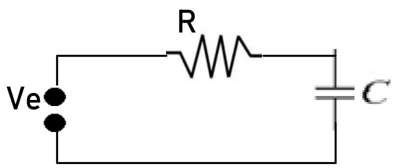
Señal de entrada:



Señal de salida:



Siguiendo los mismos pasos anteriormente mencionados, realizaremos otro ejemplo:



En donde:
Ve= 4 V
R=200 Ω
C= 200 µF
F= 60 Hz

De igual manera, para poder ingresar este circuito a Simulink, primero debemos de determinar la ecuación que rige el comportamiento de este tipo de circuitos como se muestra a continuación:

$$Ve(t) = Ri(t) + \frac{1}{C}\int i(t)dt$$

Si:

$$i = \frac{dq}{dt}$$

Entonces:

$$Ve(t) = R\frac{dq}{dt} + \frac{q}{C}$$

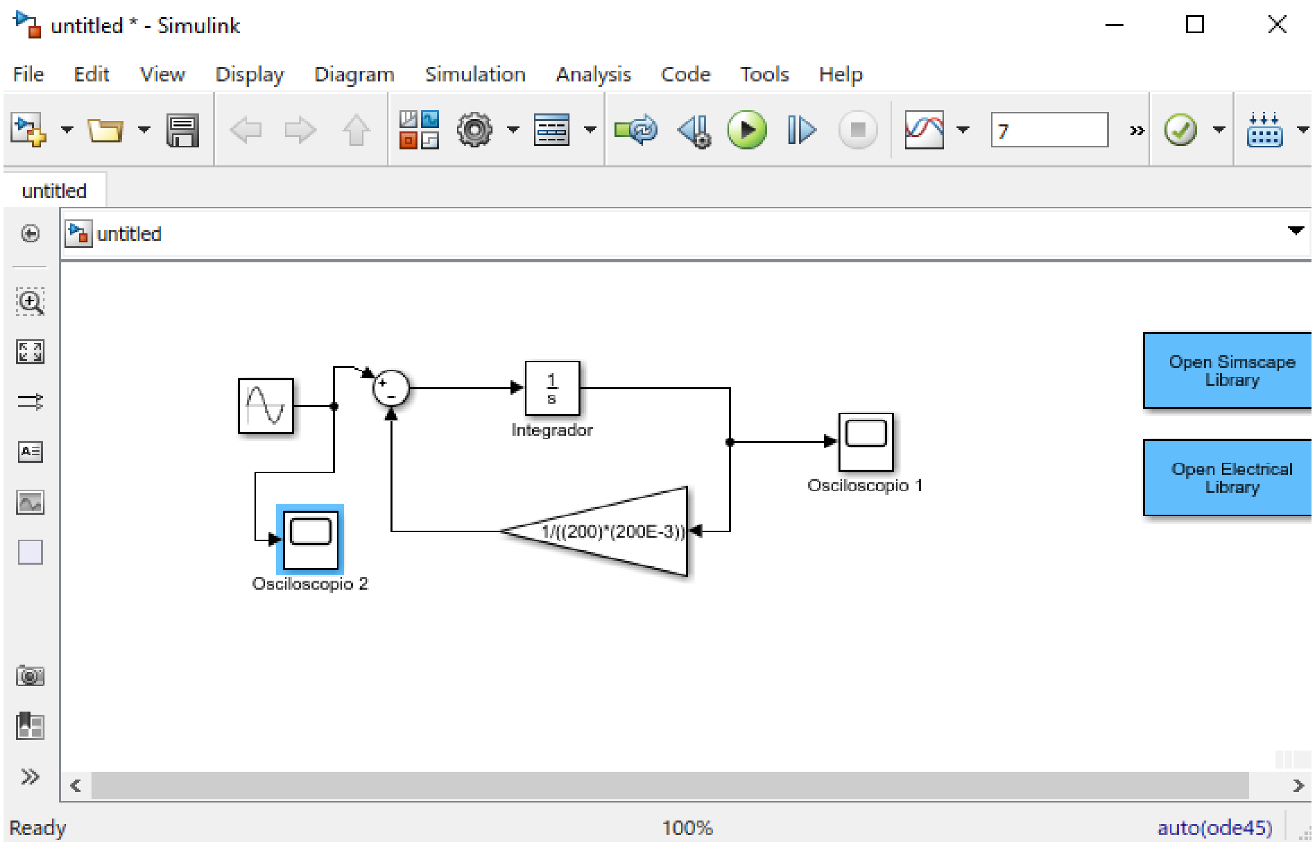
Despejando $\frac{dq}{dt}$:

$$\frac{dq}{dt} = \frac{Ve}{R} - \frac{q}{RC}$$

Sustituyendo los valores iniciales:

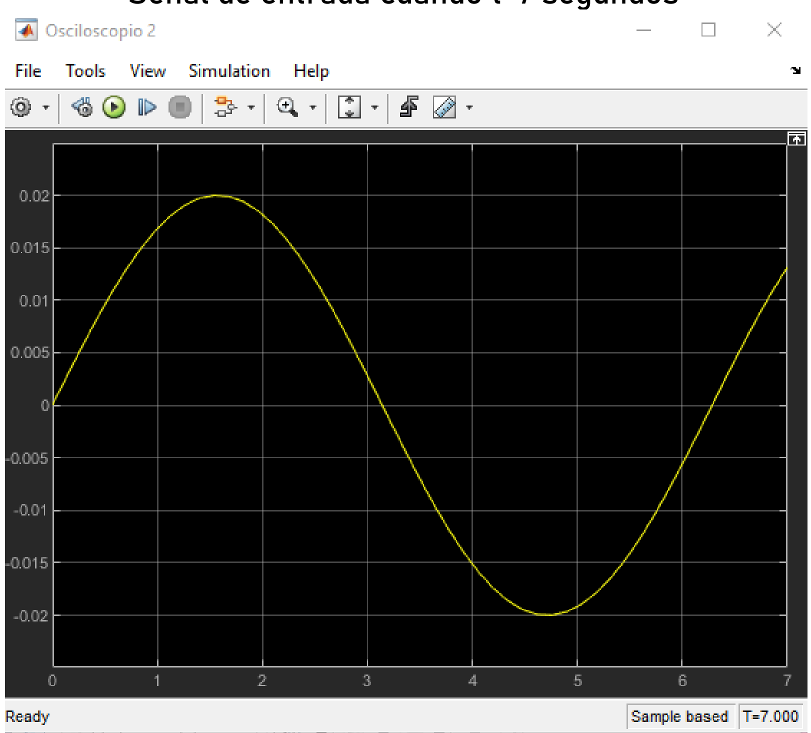
$$\frac{dq}{dt} = \frac{4V}{200\Omega} - \frac{1}{(200\Omega)(200x10^{-3}F)}$$

Ésta será la ecuación que diseñaremos en Simulink. (Los elementos a usar ya se mencionaron en el ejemplo anterior.)

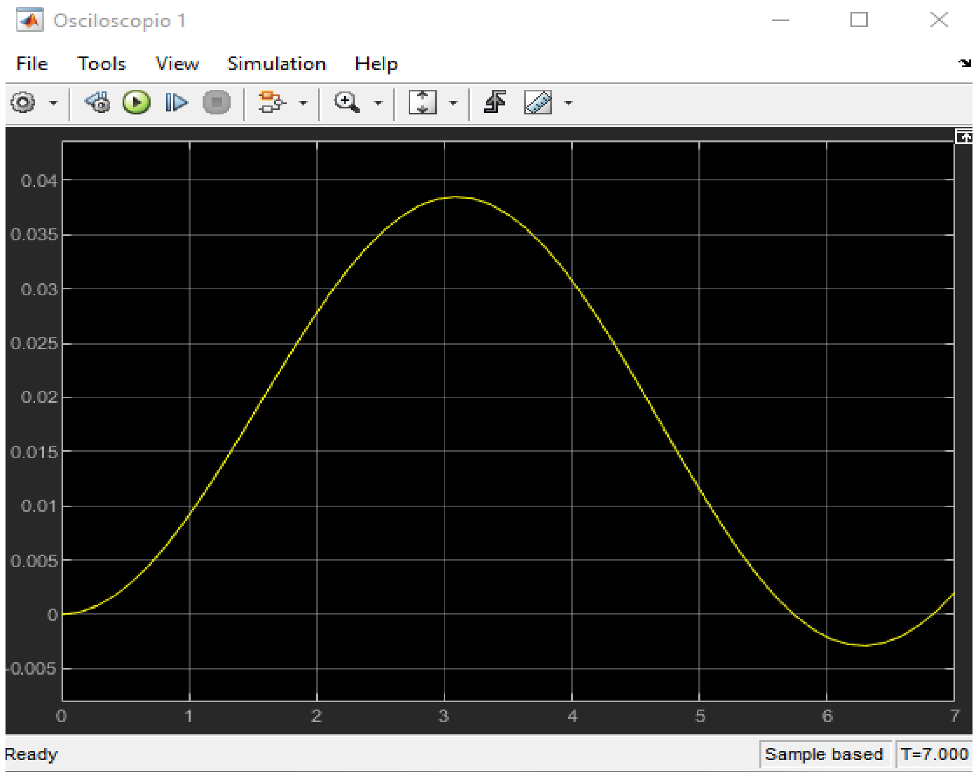


Como en las condiciones iniciales no especifican en que tiempo evaluar la ecuación modelo, entonces vamos a probar con 7 segundos y con 3 segundos para poder observar las diferencias que existen por medio de las gráficas.

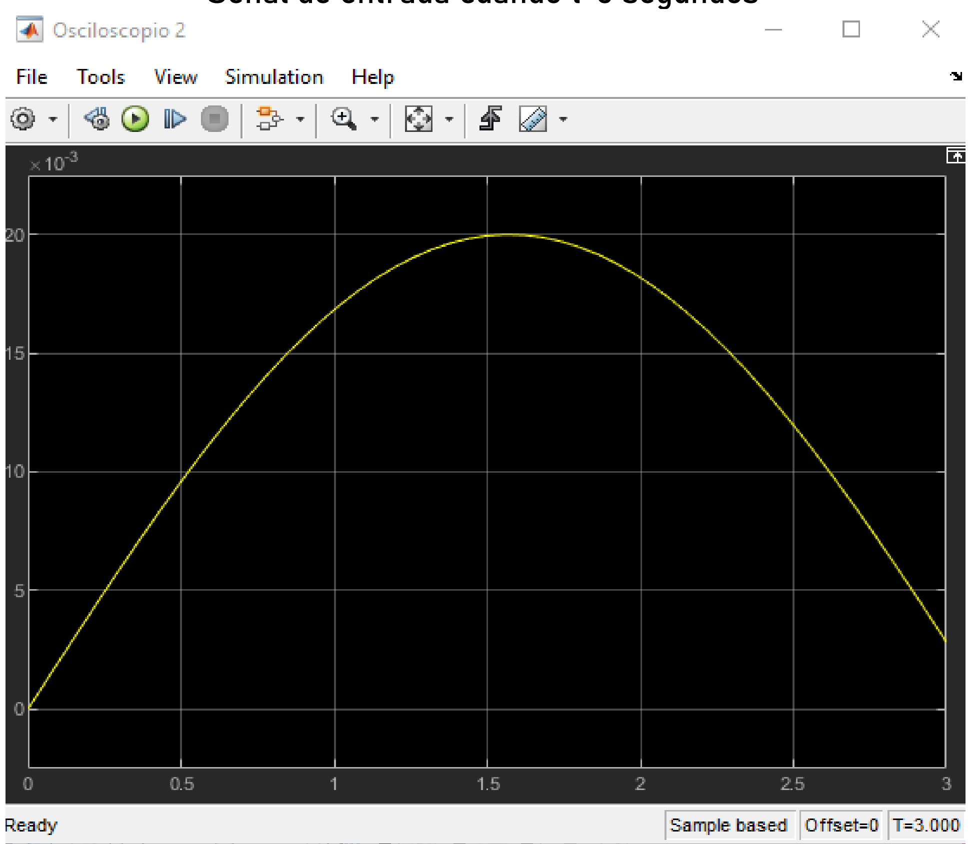
Señal de entrada cuando t=7 segundos



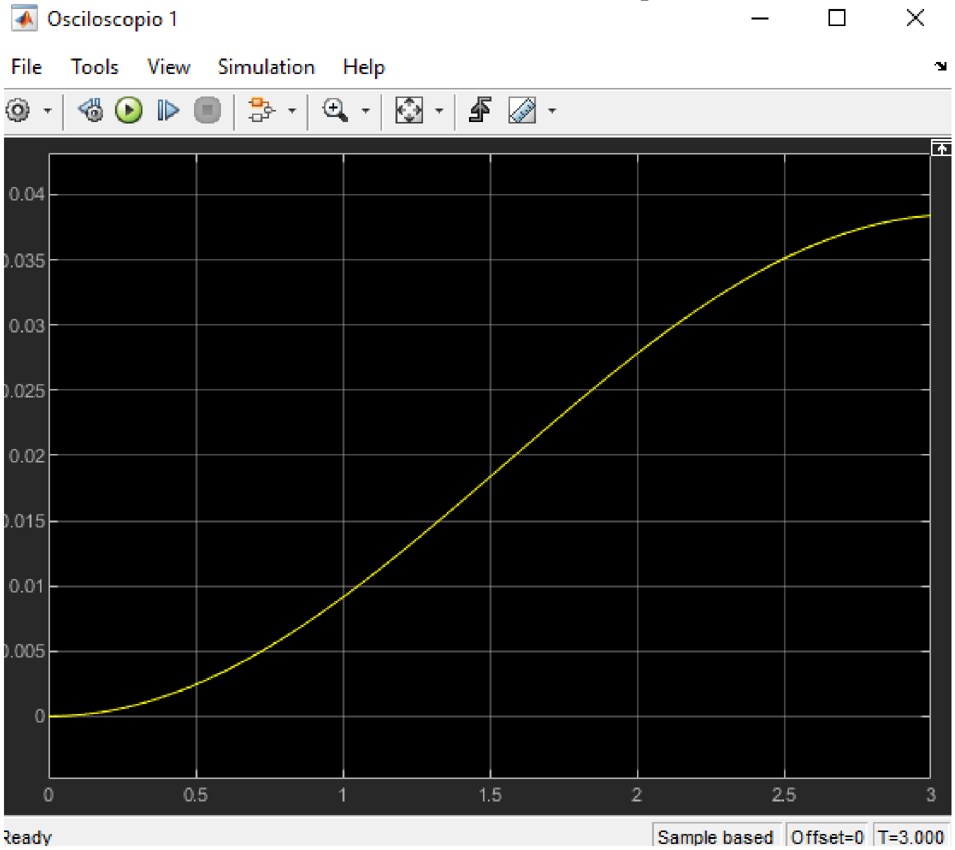
Señal de Salida cuando t=7 segundos



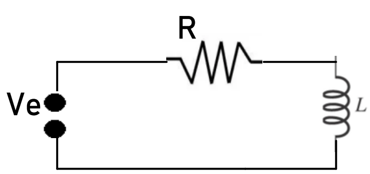
Señal de entrada cuando t=3 segundos



Señal de Salida cuando t=3 segundos



A continuación se realizara un ejemplo más:



En donde:
Ve= 15 V
R=100 Ω
L= 6 H
F= 60 Hz
t= 10 segundos

Se obtiene el modelo:

$$Ve(t) = L\frac{di}{dt} + \; Ri\ (t)$$

Si:

$$i = \frac{dq}{dt} \; \rightarrow \; \frac{d^2q}{dt^2}$$

Entonces:

$$Ve(t) = L\frac{d^2q}{dt^2} + \; R\frac{dq}{dt}$$

Despejando $\frac{d^2q}{dt^2}$:

$$\frac{d^2q}{dt^2} = \left(Ve - R\frac{dq}{dt}\right)\frac{1}{L}$$

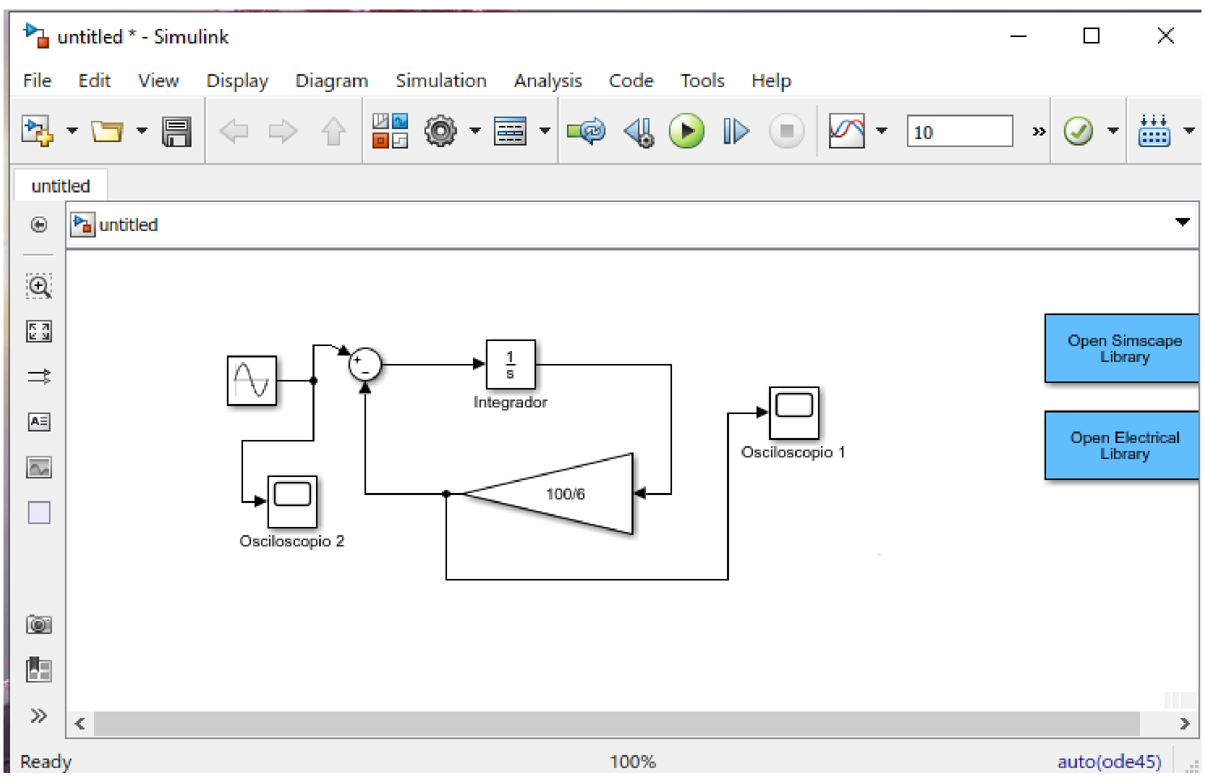
$$\frac{d^2q}{dt^2} = \left(\frac{Ve}{L} - \frac{R}{L}\frac{dq}{dt}\right)$$

Sustituyendo los valores iniciales:

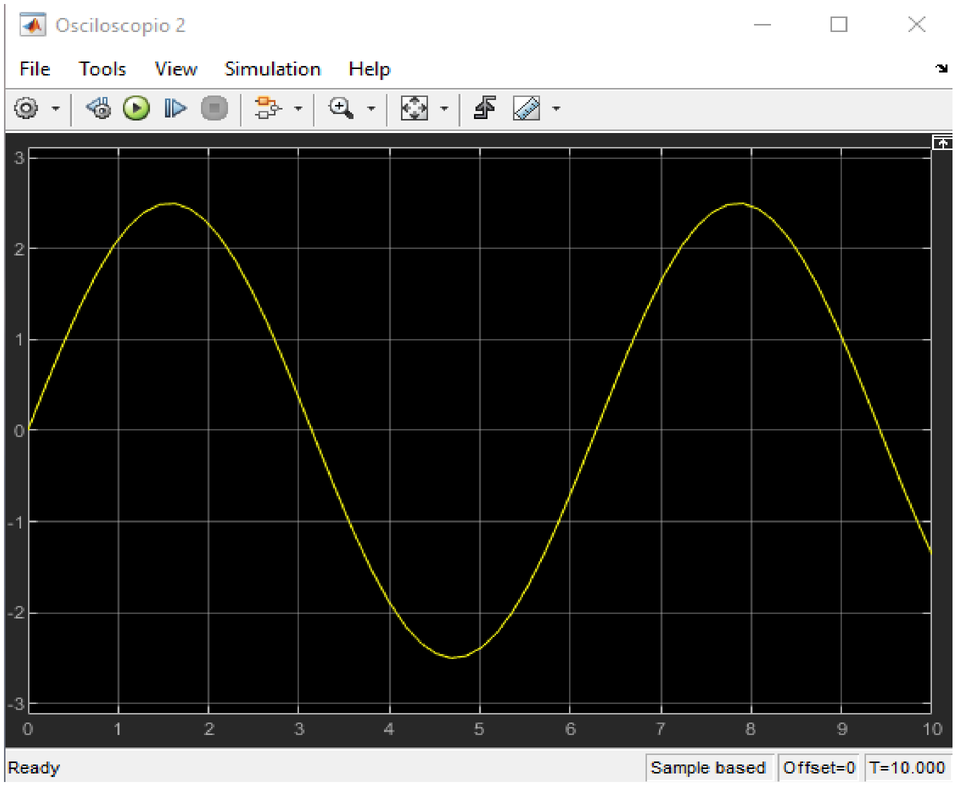
$$\frac{d^2q}{dt^2} = \left(\frac{15V}{6H} - \frac{100\Omega}{6H}\frac{dq}{dt}\right)$$

Esta es la ecuación con la que vamos a trabajar para realizar el circuito en Simulink. El procedimiento es el mismo que el de los circuitos anteriores, es decir, se usan elementos que ya se usaron anteriormente. Las librerías y elementos están explicados en el primer ejercicio de éste tema.

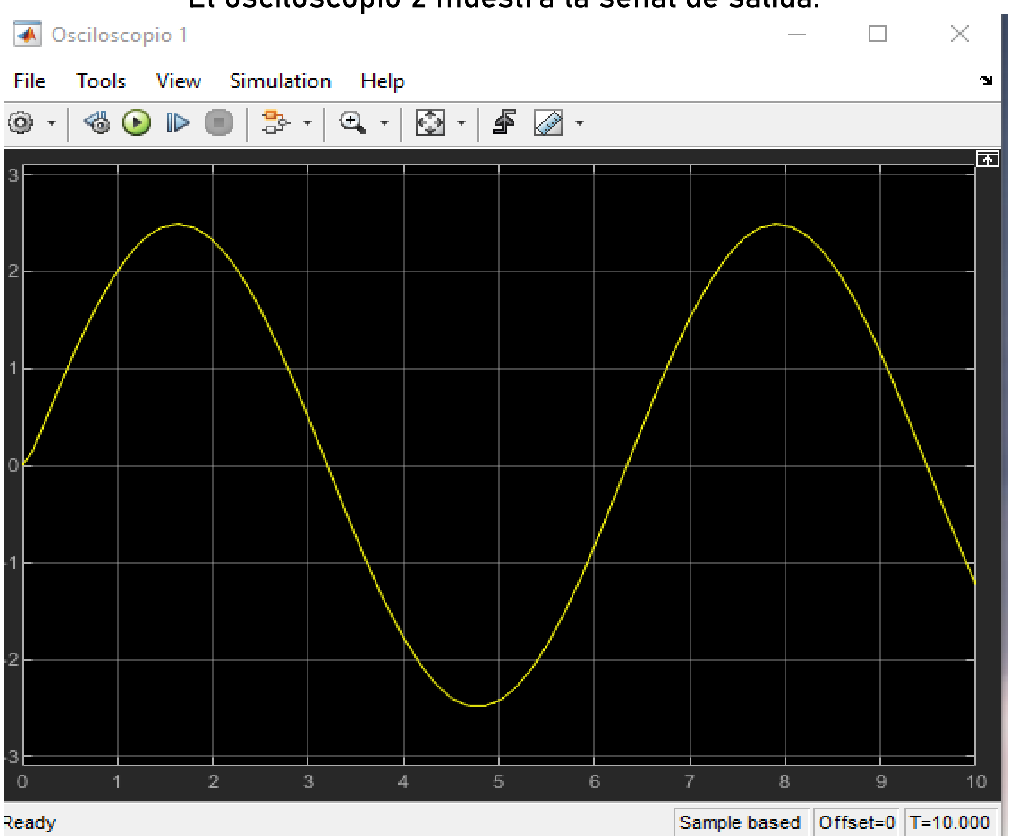
El circuito en Matlab nos quedaría de la siguiente manera:



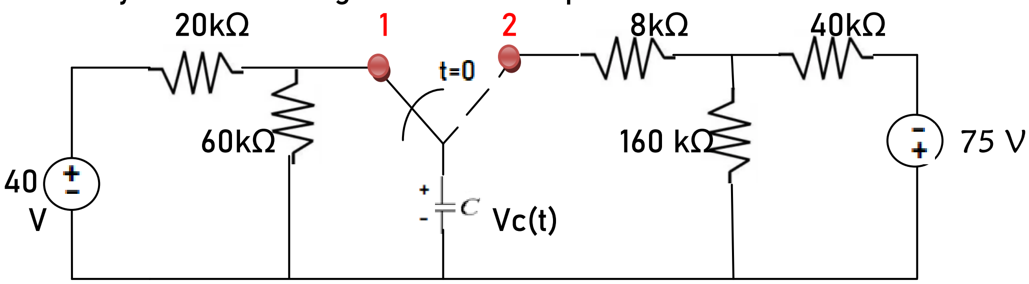
El osciloscopio 2 es el responsable de mostrar la señal de entrada:



El osciloscopio 2 muestra la señal de salida:

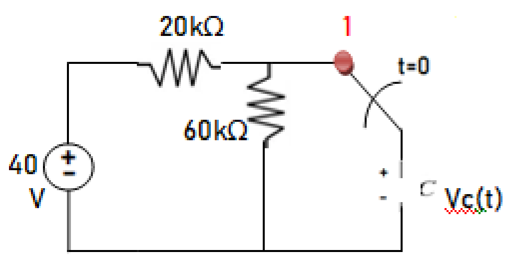


Analiza y resuelve el siguiente circuito por medio de la transformada de Laplace:



El switch pasa de la posición 1 a la posición 2 cuando t=0. Hallar Vc(t) cuando t>0 con C= $^1/_4$μF

Cuando t<0 entonces tenemos el siguiente circuito:



En DC, el capacitor se comporta como un circuito abierto.

Analizando este circuito por ley de Kirchhoff

$$\frac{Vc(0^-) - 40}{20\ k\Omega} + \frac{Vc(0^-) - 0}{60\ k\Omega} = 0$$

$$Vc(0^-)\left(\frac{1}{20\ k\Omega} + \frac{1}{60\ k\Omega}\right) = \frac{40}{20\ k\Omega}$$

$$Vc(0^-)\left(\frac{1}{15000}\right) = \frac{40}{20000\Omega}$$

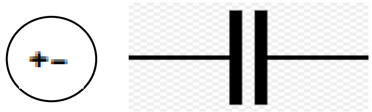
$$Vc(0^-) = 30V$$

Entonces cuando t<0

$$Vc(0^-) = Vc(0^+)$$

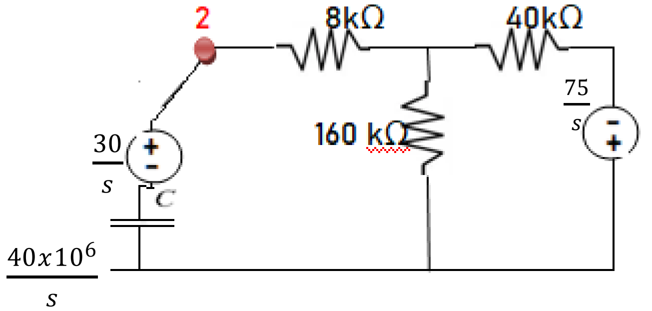
Pasando el capacitor a Dominio de Laplace:

Vc(s)



$\frac{Vc(0^-)}{s}$ $\frac{1}{sc}$

Entonces:



Malla 1 (I1) Malla 2 (2)

Para malla 1:

$$-Vc\,(s) + \,(8000)I1(s) + \,\big(I1\,(s) - \,\,I2(s)\big)(160000) = 0\,\,\,\ldots..(1\,)$$

$$Vc(s) + \left(\frac{4000000}{s}\right) I1(s) + \frac{30}{s} \ldots.la\ sustituimos\ en\ (\,1\,)$$

$$-\frac{4000000}{s}\,I1(s) - \frac{30}{s} + \,8000I1(s) + \,\big(I1(s) - I2(s)\big)(160000) = 0$$

$$\left(-\frac{4000000}{s} + 168000\right)\,I1(s) - \,160000I2(s) - \frac{30}{s} = \,0\,\,\ldots\ldots\ldots..(\,2\,)$$

Malla 2:

$$-\frac{75}{s} + 160000\,(\,I2(s) - \,I1(s)\,) + \,40000\,I2(s) = 0$$

$$-160000I1(s) + \,200000\,I2(s) - \frac{75}{s} = \,0\,\,\ldots\ldots\ldots..(\,3\,)$$

Para hallar los valores de I1(s), I2(s) recurrimos al uso de Matlab:

Primero ingresamos la ecuación que llamaremos eqn1:

$$\left(-\frac{4000000}{s} + 168000\right)\,I1(s) - \,160000I2(s) - \frac{30}{s} = \,0$$

En Matlab, para meter una ecuación, se escribe de la siguiente manera:

eqn1=('((-4000000/s)+168000)*I1-(160000)*I2-(30/s)=0')

Posterior mente ingresamos la segunda ecuación que llamaremos eqn2:

$$-160000I1(s) + 200000\, I2(s) - \frac{75}{s} = 0$$

En Matlab se vería asi:

eqn2=('((-160000)*I1)+(200000*I2)-(75/s)=0')

Después damos la instrucción a Matlab de que resuelva ese sistema de operaciones para obtener los resultados de I1 e I2, para esto se designa una variable más que es la que contendrá dicho resultado. En seguida, se usa el comando solve y entre paréntesis se pondrán primero las variables de ambas ecuaciones y después las variables a despejar dentro de esas dos ecuaciones (situándolas entre comillas), todo esto separado por comas. De la siguiente manera:

solution=solve(eqn1,eqn2,'I1','I2')

Dependiendo de la versión de Matlab, es probable que el programa no te arroje el resultado de cada variable después de realizar el punto anterior, por lo cuál es necesario realizar otro paso y es el siguiente:

I1=solution.I1

De esta manera, Matlab nos arrojará el valor de I1 dentro del dominio de Laplace. Posteriormente debemos declarar una variable simbólica por medio del comando syms, en este caso será la variable "s" por estar en el dominio de Laplace:

syms s;

Ahora se necesita una expresión para encontrar el voltaje del capacitor, le vamos a otorgar la variable "v", pero para saber cual es esa expresión debemos analizar el circuito.

EL voltaje del capacitor sería la impedancia $\left(\frac{4000000}{s}\right)$ multiplicada por la corriente I1 más la condición inicial. Una vez analizado, se tendría lo siguiente:

$$V = \frac{4000000}{s}\, I1 + \frac{30}{s}$$

Y en Matlab lo escribiríamos de la siguiente manera:

v=(4000000/s)*I1+(30/s)

Después, Matlab nos arrojaría el siguiente resultado:

V=30/s + 9000/(s*s(-100.0))

O bien:

$$V = \frac{30}{s} + \frac{9000}{s(s-100)}$$

Notamos que sigue estando en el dominio de Laplace y para obtener el resultado que nosotros necesitamos es necesario realizar la transformada inversa con ayuda del comando ilaplace y la variable a la que le vamos a realizar la transformada inversa. En Matlab se vería así:

ilaplace(V)

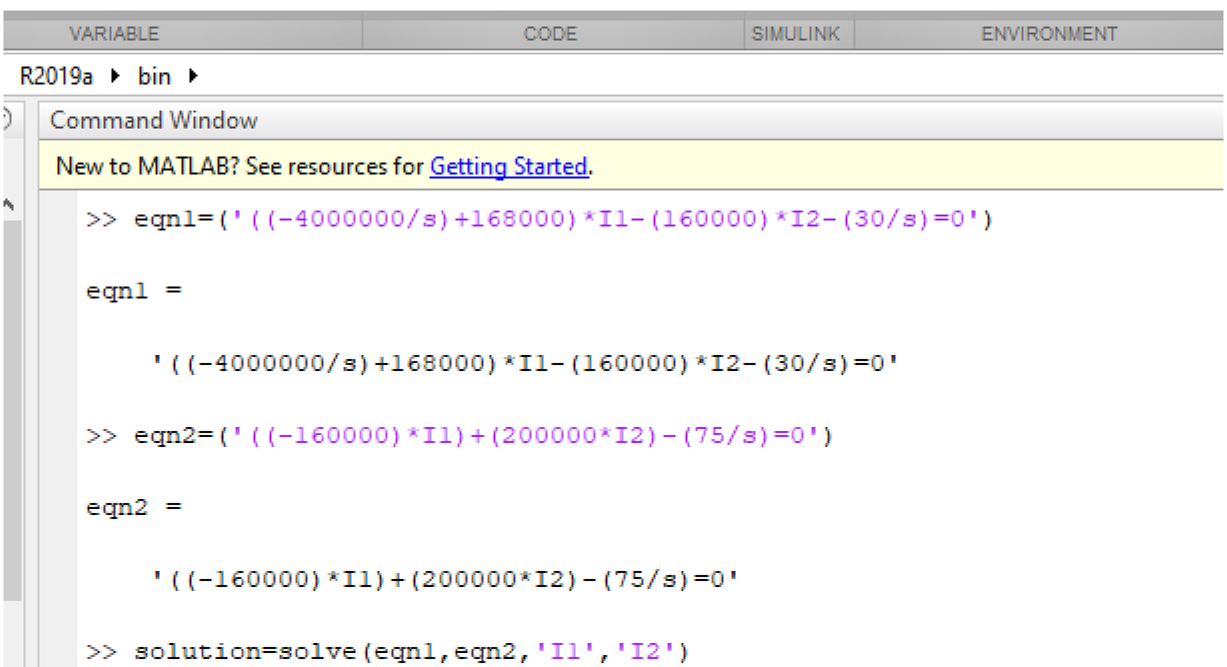
Mostrandonos como resultado:

$$V = 90^{100t} - \ 60$$

Es decir que el voltaje en el capacitor para un t>0 es:

$$\boldsymbol{V = 90^{100t} - \ 60}$$

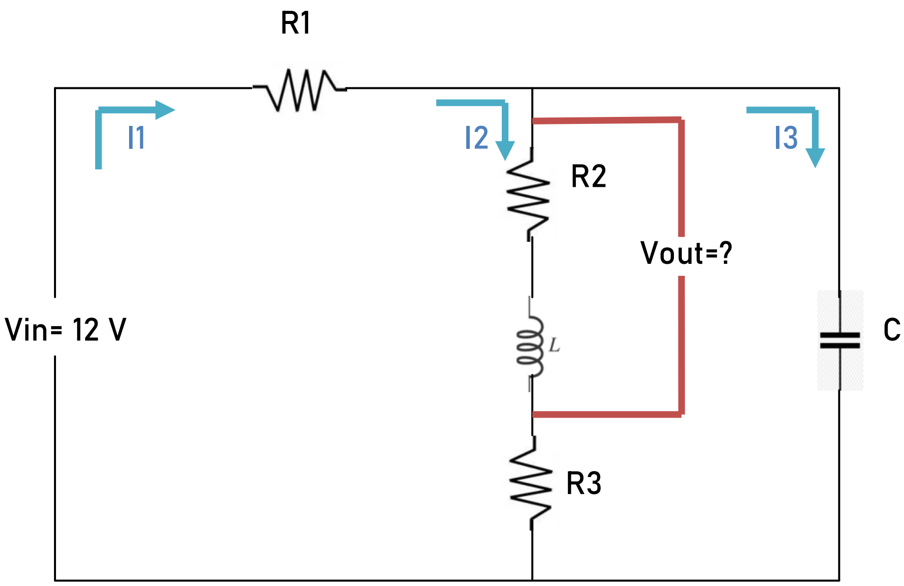
Todo el proceso en Matlab sería de la siguiente manera:



```
solution =

    I1: [1x1 sym]
    I2: [1x1 sym]

>> I1=solution.I1
```

```
I1 =

0.00225/(s - 100.0)

>> v=(4000000/s)*I1+(30/s)

V =

30/s + 9000.0/(s*(s - 100.0))

>> ilaplace(V)

ans =

90.0*exp(100.0*t) - 60.0
```

A continuación, se realizara otro ejemplo sobre la transformada de Laplace aplicada a circuitos eléctricos para obtener el voltaje de salida de dicho circuito:



En donde:
R1= 100Ω
R2=78Ω
R3=550Ω
L=0.48 H
C= 620 µF
Vin=12 V

$$V_{out} = (R_2 I_2(t)) + L\frac{dI(t)}{dt}$$

Al aplicar Laplace:

$$V_{out} = (R_2 I_2(s)) + SLI_2(s) \ldots\ldots\ldots (1)$$

$$V_{out} = I_2(s)\,(R_2 + SL) \ldots\ldots\ldots (2)$$

Despejamos $I_2(s)$

$$I_2(s) = \frac{Vout}{R2 + SL} \ldots\ldots\ldots (3)$$

Por medio de la ilistración del circuito sabemos que:

$$I_1 = I_2 + I_3$$

Entonces:

$$R_2 I_2(t) + L\frac{dI(t)}{dt} + R_3 I_2(t) = C\frac{di(t)}{dt}$$

Al aplicar Laplace:

$$(R_2 I_2(s)) + SLI_2(s) + R_3 I_2(t) = \frac{I_{3(s)}}{SC}$$

Y sabemos que la ec. (1) :

$$V_{out} = (R_2 I_2(s)) + SLI_2(s)$$

Entonces:

$$Vout + R_3 I_2(t) = \frac{I_{3(s)}}{SC} \ldots\ldots\ldots (4)$$

Despejamos $I_3(s)$

$$I3(s) = SC\left(V_{out} + R_3 I_2(t)\right) \ldots\ldots\ldots (5)$$

También en el circuito se puede determinar que en la malla 1 (izquierda):

$$Vin(t) = R_1 I_1(t) + R_2 I_2(t) + L\frac{dI_2(t)}{dt} + R_3 I_2(t)$$

Aplicando Laplace:

$$Vin(s) = R_1 I_1(s) + R_2 I_2(s) + SLI_2(s) + R_3 I_2(s)$$

$$V_{out} = (R_2 I_2(s)) + SLI_2(s)$$

$$Vin(s) = R_1\left(\left(\frac{Vout}{R_2 + SL}\right) + SC\ (Vout + R_3 I_2(s)\ )\right) + Vout + \frac{R_3 Vout}{R_2 + SL}$$

$$Vin(s) = R_1\left(\frac{Vout + SCVout\ (R_2 + SL) + SC\ R_3 Vout}{R_2 + SL}\right) + Vout + \frac{R_3 Vout}{R_2 + SL}$$

$$Vin(s) = Vout\ \left(\left(\frac{R_1\ (1 + SC\ (R_2 + SL) + SC\ R_3)}{R_2 + SL}\right) + \frac{R_2 + SL + R_3}{R_2 + SL}\right)$$

$$Vin(s) = Vout\ \left(\frac{R_1\ (1 + SC\ (R_2 + SL) + SC\ R_3) + R_2 + SL + R_3}{R_2 + SL}\right)$$

$$Vin(s) = Vout\ \left(\frac{R_1 + R_1 SCR_2 + R_1 S^2 LC + R_1 R_3 SC + R_2 + SL + R_3}{R_2 + SL}\right)$$

Despejando $Vout$:

$$Vout = \frac{R_2 + SL}{R_1 S^2 LC + S\ (L + R_1 R_3 C + R_1 R_2 C + R_1 + R_2 + R_3\ )} \ldots\ldots\ldots (\ 6\ )$$

La ecuación (6) es la que nos va a servir para determinar el Voltaje de salida $Vout$, por eso es la que vamos a ocupar para realizar la operación usando Matlab

Antes de introducir la ecuación debemos determinar los valores de los elementos del circuito como se hizo en el primer ejemplo de éste tema. No importa el orden en que introduzcamos estos valores, pero si debemos asegurarnos que los introducimos todos y de manera correcta.

Los datos a introducir son:

R1= 273Ω
R2=78Ω
R3=550Ω
L=0.48 F
C= 620 μF

$$Vout = \frac{R_2 + SL}{R_1 S^2 LC + S\ (L + R_1 R_3 C + R_1 R_2 C + R_1 + R_2 + R_3\ )}$$

En Matlab se vería asi:
Primero introducimos el valor de las variables:

```
>> R1=273;
>> R2=99;
>> R3=618;
>> L=0.66;
>> C=0.0062;
```

Después, como aplicaremos Laplace inversa, antes es necesario usar el comando syms s t; el que define que estamos utilizando una ecuación en el dominio de laplace. Si no introducimos este comando, Matlab no realizara la operación requerida.

```
>> syms s t;
```

Luego, introducimos la ecuación a resolver, que en este caso llamaremos eqn6, pero podría llevar cualquier otro nombre:

```
>> eqn6=(R2+(s*L))/((R1*L*C*s*s)+(s*(L+(R1*R3*C)+(R1*R2*C)+R1+R2+R3)));
```

Usamos el comando ilaplace poniendo entre paréntesis la ecuación a la que le queremos aplicar Laplace inversa, en este caso la llamamos eqn6;

```
>> ilaplace(eqn6)
```

Matlab nos arrojala el resultado (no simplificado) de esa Laplace inversa, que es el resultado del voltaje de salida solicitado en el ejercicio. Esta ecuación resultante nos sirve para determinar el voltaje de salida en cualquier instante de tiempo.

```
ans =

(3328110901109300143222885932O7296*exp(-
(9927078393749850112*t)/5031043201329619))/609662600853776098069044437940275 +
217703302299648/4847206246948169
```

Para que el resultado fuese un poco más gráfico, se realizó una gráfica. Para ello es necesario seguir los siguientes pasos:

Primero definimos nuestro numerados de la ecuación, es decir, de la eqn6 usando el comando num como se muestra a continuación:

```
>> num= ([L R2])
```

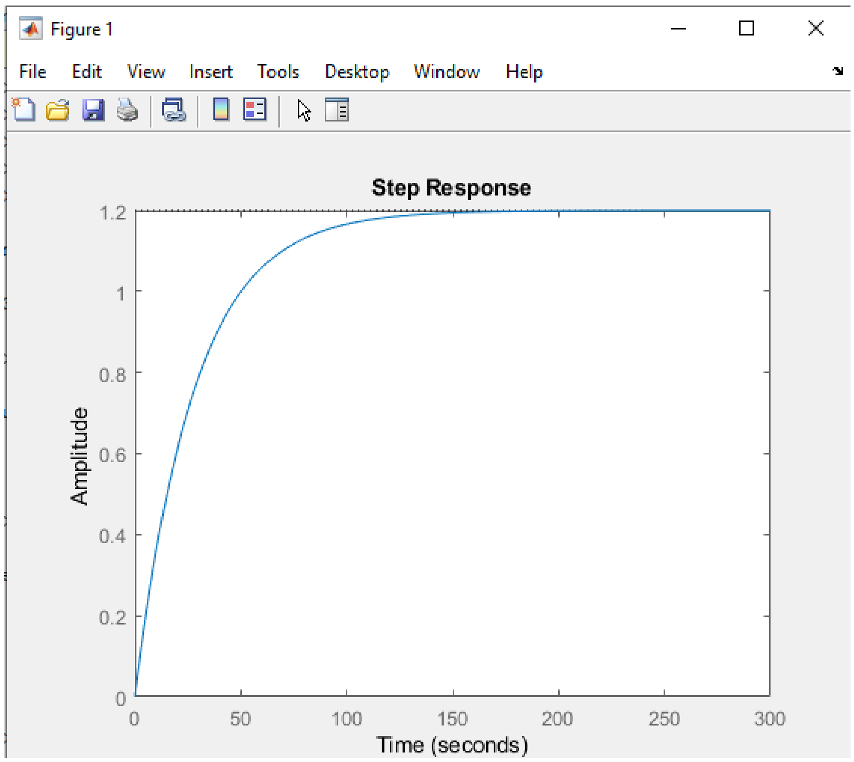
Despues hacemos lo mismo pero ingresando el denominador con el comando den:

```
>> den=[(R1*L*C) (L*R1*R3*C+R1*R2) (R1+R2+R3)]
```

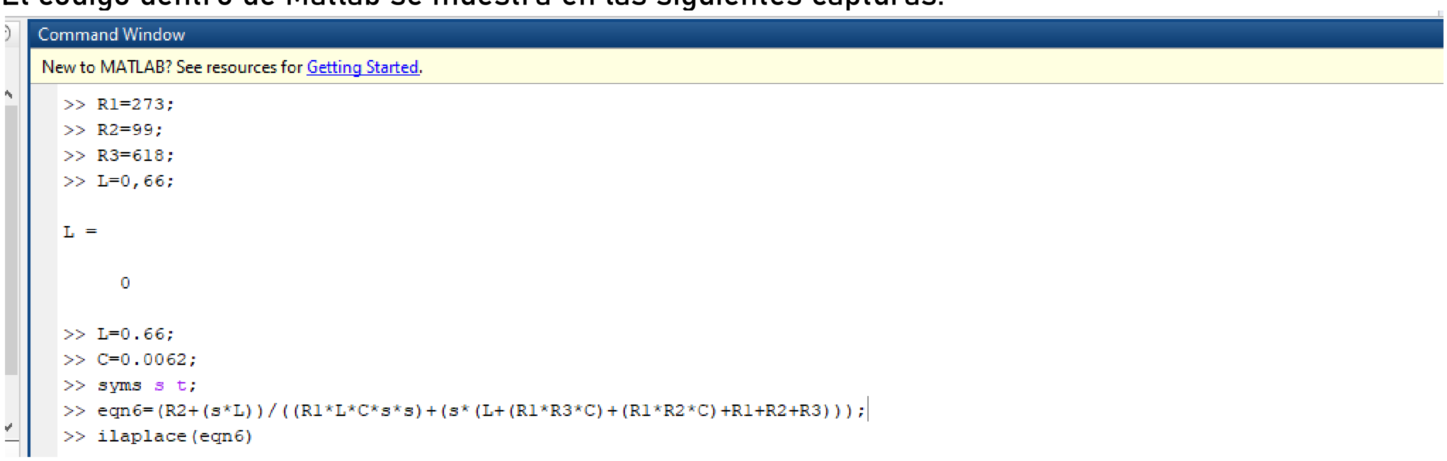
Por último usamos el comando step el cual nos va a realizar la gráfica solicitada. Hay que recordar que hay un voltaje de entrada el cual es Vin=12 V, asi que multiplicamos los 12V por el numerador, esto se ingresa en Matlab de la siguiente manera:

```
>> step(12*num,den)
```

Y al dar "Enter" se nos abrirá una ventana con la gráfica solicitada, la cual nos muestra cómo se va estabilizando el voltaje de salida conforme el tiempo va avanzando.

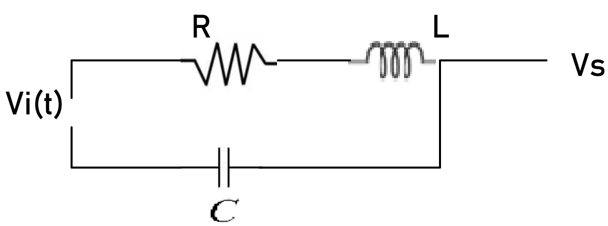


El código dentro de Matlab se muestra en las siguientes capturas:



```
ans =

(33281109011093001432228859320 7296*exp(-(992707839374985 0112*t)/5031043201329619))/60966260085377609806904443794 0275 + 217703302299648/4847206246948

>> num= ([L R2])

num =

    0.6600   99.0000

>> den=[(R1*L*C) (L*R1*R3*C+R1*R2) (R1+R2+R3)]

den =

   1.0e+04 *

    0.0001    2.7717    0.0990

>> step(12*num,den)
```

Por último se realizará un ejemplo más, en donde se deberá determinar el voltaje de salida Vs:



En donde:
R=136Ω
L=0.15 H
C=0.00531 F
Vi=18 V

Haremos el análisis del circuito:

$$Vi(t) = L\frac{di(s)}{dt} + \ Ri(t) + \frac{1}{C}\int i(t)dt$$

$$Vs(t) = \frac{1}{C}\int i(t)dt$$

Aplicamos la transformada de Laplace a ambas ecuaciones:

$$Vi(s) = SL(I(s)) + \ RI(s) + \frac{1}{SC}I(s)$$

$$Vs(s) = \frac{1}{SC}I(s)$$

Agrupamos y Simplificamos:

$$\frac{Vs}{Vi} = \frac{\frac{1}{SC} I(s)}{SL(I(s)) + RI(s) + \frac{1}{SC} I(s)}$$

$$\frac{Vs}{Vi} = \frac{\frac{1}{SC} I(s)}{I(s)\left(SL + R + \frac{1}{SC}\right)}$$

$$\frac{Vs}{Vi} = \frac{\frac{1}{SC}}{\left(SL + R + \frac{1}{SC}\right)}$$

$$\frac{Vs}{Vi} = \frac{1}{SC\left(SL + R + \frac{1}{SC}\right)}$$

$$\frac{Vs}{Vi} = \frac{1}{S^2CL + SCR + 1}$$

$$Vs = Vi\left(\frac{1}{S^2CL + SCR + 1}\right) \ldots \ldots \ldots (1)$$

La ecuación (1) es la que nos va a dar la solución de cuál es nuestro voltaje de salida. Pero observamos que está en el dominio de Laplace, por ello, recurrimos a Matlab.

Deberemos escribir lo siguiente dentro de Matlab:
Nota: todos los comandos utilizados en este ejercicio se explicaron en ejercicios anteriores

```
>> R=136;
>> L=0.15;
>> C=0.00531;
>> Vi=18;
>> syms s t;
>> Vs=((Vi)/((s^2*C*L)+(s*C*R)+1));
>> ilaplace(Vs)

ans =

(50000*177^(1/2)*203362^(1/2)*sinh((40*177^(1/2)*203362^(1/2)*t)/531)*exp(-(1360*t)/3))/5999179
```

Las capturas del código dentro de Matlab son las siguientes:

```
▸ C: ▸ Program Files ▸ Polyspace ▸ R2019a ▸ bin ▸
Command Window
New to MATLAB? See resources for Getting Started.
>> R=136;
>> L=0.15;
>> C=0.00531;
>> Vi=18;
>> syms s t;
Did you mean:
>> Vs=((Vi)/((s^2*C*L)+(s*C*R)+1));
>> ilaplace(Vs)

ans =

(50000*177^(1/2)*203362^(1/2)*sinh((40*177^(1/2)*203362^(1/2)*t)/531)*exp(-(1360*t)/3))/5999179
```

Tema 4: Fracciones Parciales.

Usando Matlab, resuelve la siguiente fracción parcial y aplica Laplace inversa al resultado.

$$F(s) = \frac{s+3}{s^2+3s+2}$$

Como en el denominador, el término mayor es s^2 entonces sabemos que de dicha eciación se obtendrán al menos dos ecuaciones parciales, una con un numerador A y otra con un numerador B. Para ello, debemos factorizar el denominador y se puede hacer con ayuda de Matlab como se muestra a continuación:

```
>> f=((s+3)/(s^2+(3*s)+2))

>> c=factor(s^2 + 3*s + 2)

c =

[ s + 2, s + 1]
```

Primero se introdujo la ecuación llamándola “f”. Para factorizar el denominador basta con elegir una variable en la cual se guardará esta factorización, en este caso se eligió la variable “c”. El comando factor es el que se va a encargar de factorizar dicho denominador.

Una vez obtenido ahora tendremos algo así:

$$F(s) = \frac{s+3}{(s+2)(s+1)}$$

Y al momento de descomponer la ecuación en fracciones parciales nos quedaría así:

$$F(s) = \frac{s+3}{(s+2)(s+1)} = \frac{A}{s+2} + \frac{B}{s+1}$$

Entonces, en Matlab debemos declarar las variables A y B por medio del comando syms:

```
>> syms A B
```

Como al resultado final le vamos a sacar la Laplace inversa, entonces declaramos también las variables necesarias para este procedimiento (este paso se puede realizar más adelante, pero se recomienda hacerlo al inicio para evitar confusiones).

```
>> syms s t;
```

Y después igualamos la ecuación F(s) (que en Matlab guardamos con la variable f) a la multiplicación de A (s+1) más la multiplicación de B(s+2), de la siguiente manera:

```
>> f=[(A*(s+1))+(B*(s+2))]

f =

A*(s + 1) + B*(s + 2)
```

Esta parte sería igual a esto:

$$s + 3 = A(s+1) + B(s+2)$$

En donde:
(s+3) → numerador de F(s)

Se simplifica esta ecuación:

$$s + 3 = As + A + Bs + 2B$$

Se agrupan términos:

$$s + 3 = \underbrace{s(A+B)}_{\text{PARTE 1}} + \underbrace{(A+2B)}_{\text{PARTE 2}}$$

Como vemos, del lado izquierdo de la igualación, el coeficiente de s es 1 así que en la primer parte s=1. De término independiente tenemos al 3 así que la segunda parte se iguala a 3, nos quedaría el siguiente sistema:

A + B =1
A + 2B =3

Esto se hace con la intención de determinar los valores de A y B. En mat lab, este sistema de ecuaciones de debe de meter a manera de matriz. La primer matriz seria la del lado izquierdo de ambas ecuaciones y la segunda matriz sería la del lado derecho del signo "=" de ambas ecuaciones.

A la primera matriz le vamos a otorgar la variable "g" y a la segunda matriz le vamos a otorgar la variabe "h". Se ingresan de la siguiente manera

```
>> g=[1 1;1 2]

g =

     1     1
     1     2

>> h=[1;3]

h =

     1
     3
```

Para resolver este sistema en Matlab, es necesario sacar la matriz inversa de la matriz "g" y la llamaremos "i". Para esto se usa el comando inv:

```
>> i=inv(g)

i =

     2    -1
    -1     1
```

Hecho esto, se resuelve la matriz, multiplicando la matriz inversa "i" por la matriz "h". Al resultado de ésta operación le llamaremos "j".

```
>> j=i*h

j =

    -1
     2
```

Estos dos valores obtenidos corresponden a A y a B, siendo:

A=-1

B=2

Entonces, si reemplazamos los valores de A y B en la ecuación F(s) tendríamos:

$$F(s) = \frac{s+3}{(s+2)(s+1)} = \frac{A}{s+2} + \frac{B}{s+1} = -\frac{1}{s+2} + \frac{2}{s+1}$$

$$F(s) = -\frac{1}{s+2} + \frac{2}{s+1}$$

Hay otra forma de calcular las fracciones parciales, que consiste en solamente meter a Matlab el numerador y denominador. Este método resulta mas sencillo y rápido, pues al usar el comando [r,p,k]=residue(num,den), Matlab nos arroja directamente las fracciones parciales y lo hace de la siguiente manera:

Primero introducimos los coeficientes del numerador num y denominador den como ya lo hemos estado haciendo en este trabajo. Nos queda de la siguiente manera:

```
>> num=[1,3]

num =

     1     3

>> den=[1,3,2]

den =

     1     3     2
```

Ahora aplicamos el comando [r,p,k]=residue(num,den). Este comando se puede explicar de la siguiente manera:

En los resultados de r, estarán los coeficientes de los numeradores de las fracciones parciales.

En los resultados de p, estarán los polos de los denominadores, por ejemplo:

p=-2

Significa que el polo o raíz del denominador es -2, entonces tendríamos algo así:

s=-2
s+2=0
s+2 → es el denominador

Y en los resultados de k serán los valores cuando con la fracción que estemos trabajando sea impropia. En caso de haber un resultado en k, se lo deberemos de sumar a las fracciones parciales resultantes.

En Matlab se llevaría a cabo de la siguiente manera:

```
>> [r,p,k]=residue(num,den)

r =

    -1
     2


p =

    -2
    -1


k =

     []
```

COEFICIENTE DEL NUMERADOR DE FRACCIÓN 1

COEFICIENTE DEL NUMERADOR DE FRACCIÓN 2

POLO DEL DENOMINADOR DE FRACCION 1

POLO DEL DENOMINADOR DE FRACCIÓN 2

ES UNA FRACCION PROPIA, POR LO QUE K=0 (NO HAY QUE AGREGARLE NINGUN VALOR A LAS FRACCIONES PARCIALES)

Esto se traduce a:
A=-1
Denominador de A = s+2
B=2
Denominador de B=s+1

Viéndolo de otra manera, sería así:

$$F(s) = -\frac{1}{s+2} + \frac{2}{s+1}$$

A esta ecuación de transferencia obtenida por medio de fracciones parciales, le vamos a calcular la Laplace inversa para que nos dé un resultado que podamos interpretar.

Para ello, ingresamos en Matlab la ecuación $F(s) = -\frac{1}{s+2} + \frac{2}{s+1}$ nombrandola "ec" y posteriormente usamos el comando ilaplace para calcular la inversa.

```
>> ec=((-1)/(s+2))+((2)/(s+1));
>> ilaplace(ec)

ans =

2*exp(-t) - exp(-2*t)
```

Siendo el resultado:

$$F(t) = 2e^{-t} - e^{-2t}$$

El código completo dentro de Matlab sería el siguiente:

```
>> syms s t;
>> f=((s+3)/(s^2+(3*s)+2))

f =

(s + 3)/(s^2 + 3*s + 2)

>> syms A B
>> c=factor(s^2 + 3*s + 2)

c =

[ s + 2, s + 1]
```

```
>> f=[(A*(s+1))+(B*(s+2))]

f =

A*(s + 1) + B*(s + 2)

>> g=[1 1;1 2]

g =

     1     1
     1     2

>> h=[1;3]

h =

     1
     3
```

```
>> i=inv(g)

i =

     2    -1
    -1     1

>> j=i*h

j =

    -1
     2
```

```
>> ec=((-1)/(s+2))+((2)/(s+1));
>> ilaplace(ec)

ans =

2*exp(-t) - exp(-2*t)
```

Segundo ejemplo:

$$F(s) = \frac{8s^2 - 7s + 6}{s^2(s-2)}$$

Podríamos descomponer la fracción en:

$$F(s) = \frac{8s^2 - 7s + 6}{s^2(s-2)} = \frac{A}{s-2} + \frac{B}{s} + \frac{C}{s^2}$$

Y para que nos sea más fácil ingresar el denominador dentro de Matlab lo vamos a simplificar:

$$s^2(s-2) = s^3 - 2s^2$$

Pasamos directamente a Matlab a ingresar el numerador y denominador como se hizo en el ejercicio anterior, solamente ingresando los coeficientes de éstos entre corchetes y separados por comas, es decir:

Para el numerador $8s^2 - 7s + 6$:
num=[8,-7,6]

Para el denominador $s^3 - 2s^2$:
den=[1,-2,0,0]

Y usando el comando [r,p,k]=residue(num,den) obtenemos las fracciones parciales:

```
>> num=[8,-7,6]

num =

     8    -7     6

>> den=[1,-2,0,0]

den =

     1    -2     0     0

>> [r,p,k]=residue(num,den)
```

```
r =

     6
     2
    -3


p =

     2
     0
     0


k =

    []
```

En este caso, las fracciones parciales que nos indica Matlab son:

A=6, B=2 , C=-3

$$\frac{6}{s-2} \quad , \quad 2 \quad y \quad -3$$

Pero recordemos como lo habíamos ordenado inicialmente:

$$F(s) = \frac{8s^2 - 7s + 6}{s^2(s-2)} = \frac{A}{s-2} + \frac{B}{s} + \frac{C}{s^2}$$

Sustituyendo los datos nos quedaría de la siguiente manera:

$$F(s) = \frac{8s^2 - 7s + 6}{s^2(s-2)} = \frac{6}{s-2} + \frac{2}{s} - \frac{3}{s^2}$$

Para determinar la inversa de Laplace usamos el comando syms s t; e ingresamos la ecuación de las fracciones parciales dándole una variable al azar, en este caso elegimos la variable “eqn”:

```
>> syms s t;
```

```
>> eqn=[(6/(s-2))+(2/s)-(3/s^2)]

eqn =

6/(s - 2) + 2/s - 3/s^2

>> ilaplace(eqn)

ans =

6*exp(2*t) - 3*t + 2
```

Y el resultado es el que Matlab nos muestra:

$$6e^{2t} - 3t + 2$$

Ejemplo 3: Encuentra las fracciones parciales para:

$$F(s) = \frac{s^4 + 3s^3 + 6s^2 + 2s + 3}{s^2 + 6s + 8}$$

Primero se debe de realizar una división sintética para disminuir el grado del numerador:

$$\begin{array}{r}
s^2 - 3s + 16 \\
\hline
s^2 + 6s + 8 \mid s^4 + 3s^3 + 6s^2 + 2s + 3 \\
-s^4 - 6s^3 - 8s^2 \\
\hline
0 - 3s^3 - 2s^2 + 2s \\
3s^3 + 18s^2 + 24s \\
\hline
0 + 16s^2 + 26s + 3 \\
-16s^2 - 96s - 128 \\
\hline
-70s - 12
\end{array}$$

Entonces:

$$(s^2 - 3s + 16) + \frac{-70s - 125}{s^2 + 6s + 8}$$

Ecuación 1 Ecuación 2

Trabajaremos con la ecuación 2, de la cuál vamos a hallar las fracciones parciales por medio de Matlab:

$$\frac{-70s - 125}{s^2 + 6s + 8}$$

```
>> num=[-70,-125]

num =

   -70  -125

>> den=[1,6,8]

den =

     1     6     8

>> [r,p,k]=residue(num,den)

r =

  -77.5000
    7.5000


p =

    -4
    -2


k =

     []
```

La primera fracción parcial corresponde a A=.77.5 o bien A=$\frac{-155}{2}$ con un polo de s=-4. Es decir:

$$-\frac{155/_2}{s+4}$$

Y la segunda fracción parcial corresponde a B=7.5 o bien B= $\frac{15}{2}$ con un polo de s=-2. Es decir:

$$+\frac{15/_2}{s+2}$$

A estas dos fracciones parciales le sumamos la ecuación 1 y se obtiene el resultado:

$$F(s) = \frac{s^4 + 3s^3 + 6s^2 + 2s + 3}{s^2 + 6s + 8} = (s^2 - 3s + 16) - \frac{155/2}{s+4} + \frac{15/2}{s+2}$$

Tema 5: Sistemas de Primer Orden

Partiendo del sistema de primer orden:

$$C(s) = \frac{1}{s+1}$$

Determina la función de transferencia a partir de una entrada escalón $\frac{1}{s}$

Analizando el sistema de primer orden, podemos notar que tiene la siguiente forma:

$$C(s) = \frac{k}{\alpha s + 1}$$

En donde:
k= ganancia= 1
α=cte. de tiempo= 1

Multiplicamos la ecuación del sistema por la entrada escalón $\frac{1}{s}$:

$$C(s) = \frac{1}{s+1} \left(\frac{1}{s}\right)$$

$$C(s) = \frac{1}{s(s+1)}$$

Esta última expresión podemos descomponerla en fracciones parciales. Para ello utilizamos Matlab, como lo hemos hecho anteriormente usando el siguiente numerador y denominador:

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

```
>> syms s t;
>> syms A B;
>> num=[1]

num =

     1

>> den=[1 1 0]

den =

     1     1     0

>> [r,p,k]=residue(num,den)
```

```
r =

    -1
     1


p =

    -1
     0


k =

     []
```

Esto quiere decir que nuestras fracciones parciales quedan de la siguiente manera:

$$C(s) = \frac{1}{s(s+1)} = \frac{1}{s} - \frac{1}{s+1}$$

Ahora determinamos la Laplace Inversa de la ecuación resultante por medio de Matlab:

```
>> eqn=[((1/s)-(1/(s+1)))]

eqn =

1/s - 1/(s + 1)

>> ilaplace(eqn)

ans =

1 - exp(-t)
```

Entonces, la salida al aplicar una entrada escalón $\frac{1}{s}$ al sistema $C(s) = \frac{1}{s+1}$ es:

$$C(t) = 1 - e^{-t}$$

Para tener un resultado gráfico, se hicieron los siguientes pasos en Matlab:

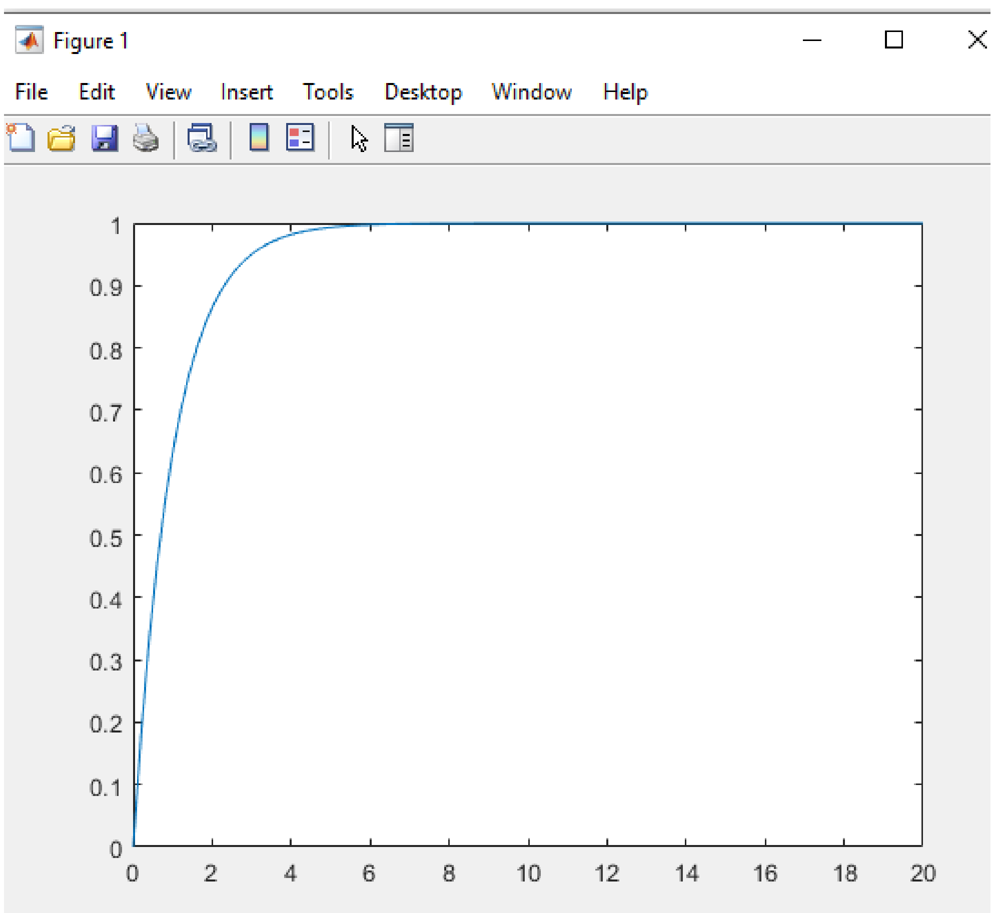
Primero definimos nuestra variable t (tiempo) con el comando linspace, lo que hace es determinar en cuantas partes dividir el eje en donde se situará t (en este caso será en el eje de las “x”).

Después, en el eje de las “y” se va a introducir la función de transferencia ya obtenida y la vamos a nombrar “y”.

Por último usamos el comando plot() que es para realizar la gráfica y entre paréntesis primero escribimos la variable que irá en el eje de las “x” y después, separado por una coma la variable del eje de las “y”. De la siguiente manera:

```
>> t=linspace(0,20);
>> y=(1-exp(-t));
>> plot(t,y);
```

Y la gráfica es la siguiente:



Que nos muestra en qué momento aproximado el sistema se va a estabilizar.

¿Qué sucederá si la ganancia de nuestro sistema fuera de 6?

$$C(s) = \frac{6}{s+1}$$

La solución en Matlab sería:

```
>> syms s t;
>> syms A B;
>> num=[6]

num =

     6

>> den=[1 1 0]

den =

     1     1     0

>> [r,p,k]=residue(num,den)

r =

    -6
     6


p =

    -1
     0


k =

     []

>> eqn2=[(-6/(s+1))+(6/s)]
```

Las fracciones parciales serían:

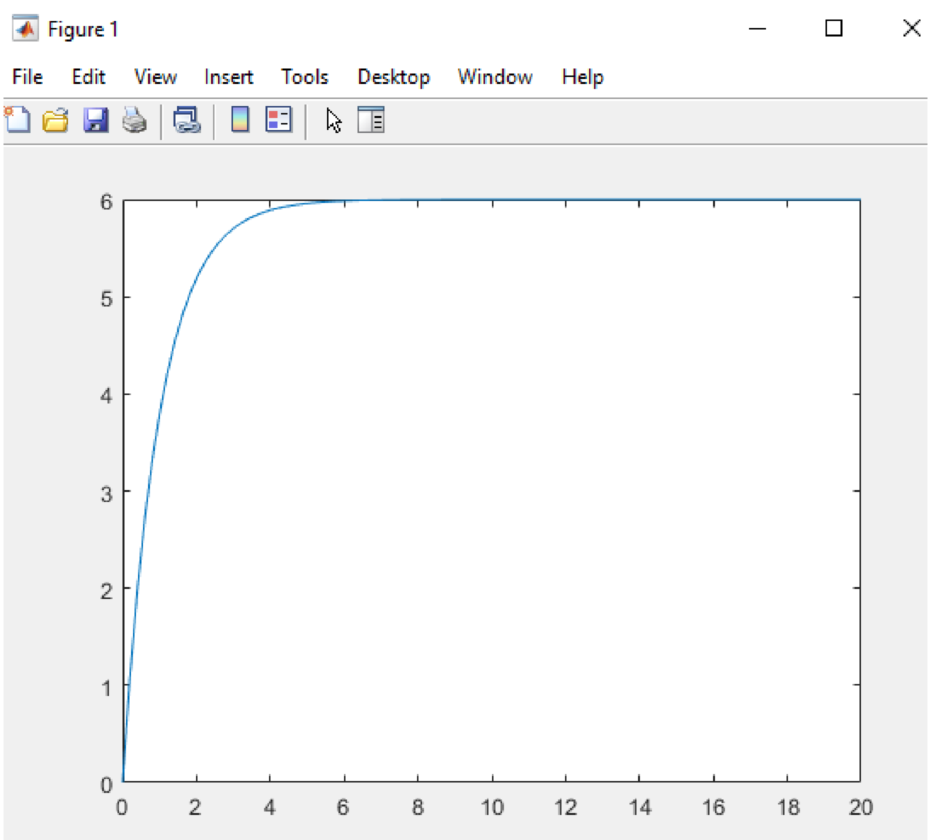
$$C(s) = \frac{6}{s+1} = -\frac{6}{s+1} + \frac{6}{s}$$

```
eqn2 =

6/s - 6/(s + 1)

>> ilaplace(eqn2)

ans =

6 - 6*exp(-t)

>> t=linspace(0,20);
>> y=(6-6*exp(-t));
>> plot(t,y);
```

La ecuación de transferencia sería:

$$C(s) = 6 - 6e^{-t}$$

Y su gráfica sería la siguiente:



Ejemplo 2:

Analiza el mismo sistema de primer orden, pero ahora con una señal rampa $\frac{1}{s^2}$ para una ganancia de K=1 y de K=8.

$$C(s) = \frac{1}{s+1}$$

Primero multiplicamos por la señal rampa:

$$C(s) = \frac{1}{s+1} \quad \left(\frac{1}{s^2}\right)$$

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

Esta última expresión podemos descomponerla en fracciones parciales. Para ello utilizamos Matlab, como lo hemos hecho anteriormente usando el siguiente numerador y denominador:

$$C(s) = \frac{1}{s^3 + s^2}$$

Metemos los datos a Matlab y solicitamos los valores de las fracciones parciales por medio del comando [r,p,k]=residue(num,den)

```
>> syms s t;
>> syms A B C;
>> num=[1];
>> den=[1 1 0 0];
>> [r,p,k]=residue(num,den)

r =

     1
    -1
     1


p =

    -1
     0
     0


k =

     []
```

Es decir, el resultado seria asi:

$$C(s) = \frac{1}{s^3 + s^2} = \frac{1}{s^2(s+1)} = \frac{A}{s+1} + \frac{B}{s} + \frac{C}{s^2}$$

$$C(s) = \frac{1}{s+1} - \frac{1}{s} + \frac{1}{s^2}$$

El siguiente paso es determinar la Laplace Inversa de la ecuación resultante por medio de Matlab:

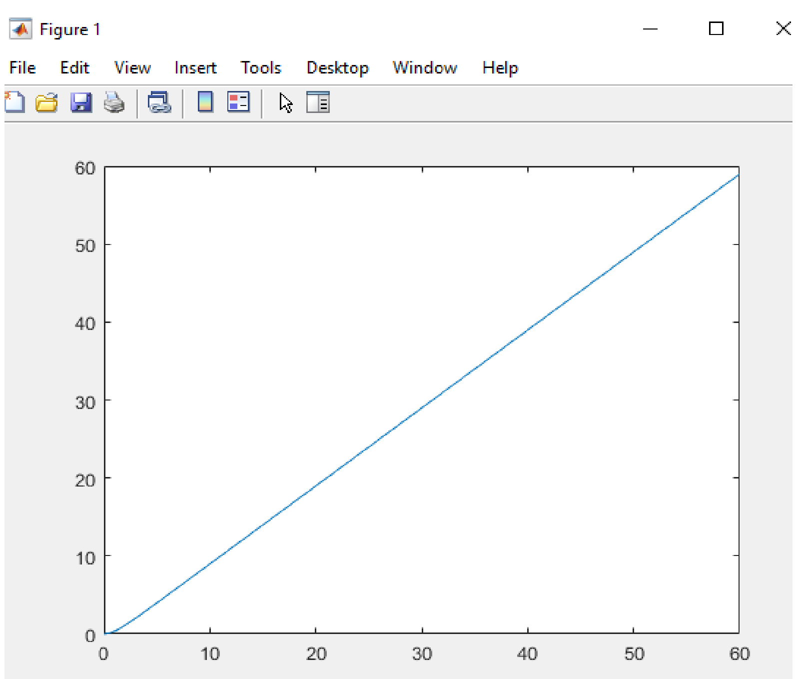
```
>> ec=[(1/(s+1)-(1/s)+(1/s^2))]

ec =

1/(s + 1) - 1/s + 1/s^2

>> ilaplace(ec)

ans =

t + exp(-t) - 1
```

Entonces, la salida al aplicar una entrada rampa $\frac{1}{s^2}$ al sistema $C(s) = \frac{1}{s+1}$ es:

$$C(t) = t + e^{-t} - 1$$

Y graficamos el resultado:

```
>> t=linspace(0,60);
>> y=(t + exp(-t) - 1);
>> plot(t,y);
```



En entradas rampa (c(r) =t) es posible calcular el error de la siguiente manera:

Error= c(r) – c(t)

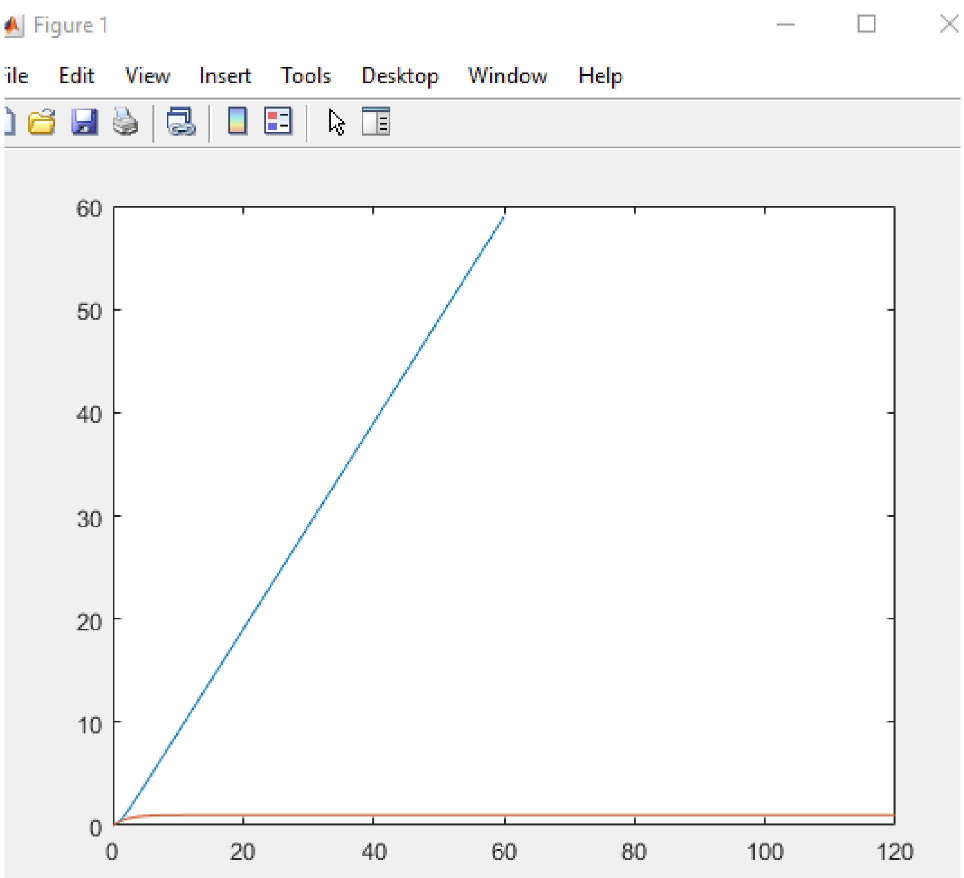
$$Error = \ t - (t + \ e^{-t} - 1)$$
$$Error = \ t - t - \ e^{-t} + 1$$
$$Error = \ - e^{-t} + 1$$

Estos valores resultantes de error se introducen en una nueva variable llamada "y3".

Con ayuda del comando hold on, lograremos que la primer gráfica ya realizada no se borre y podamos colocar la segunda gráfica en los mismos ejes coordenados.

Este es el proceso en Matlab:

```
>> syms s t;
>> t=linspace(0,60);
>> y=(t + exp(-t)-1);
>> plot(t,y);
>> hold on
>> t3=linspace(0,120);
>> y2=(-exp(-t)+1);
>> y3=(-exp(-t)+1);
```



Ahora, si la ganancia fiera k=8

$$C(s) = \frac{8}{s+1} \left(\frac{1}{s^2}\right)$$

$$C(s) = \frac{8}{s^2(s+1)}$$

```
>> syms t s;
>> syms A B C;
>> num=[8];
>> den=[1 1 0 0];
>> [r,p,k]=residue(num,den)

r =


     8
    -8
     8


p =


    -1
     0
     0


k =


     []
```

Es decir, el resultado seria asi:

$$C(s) = \frac{8}{s^3+s^2} = \frac{8}{s^2(s+1)} = \frac{A}{s+1} + \frac{B}{s} + \frac{C}{s^2}$$

$$C(s) = \frac{8}{s+1} - \frac{8}{s} + \frac{8}{s^2}$$

Aplicamos Laplace Inversa:

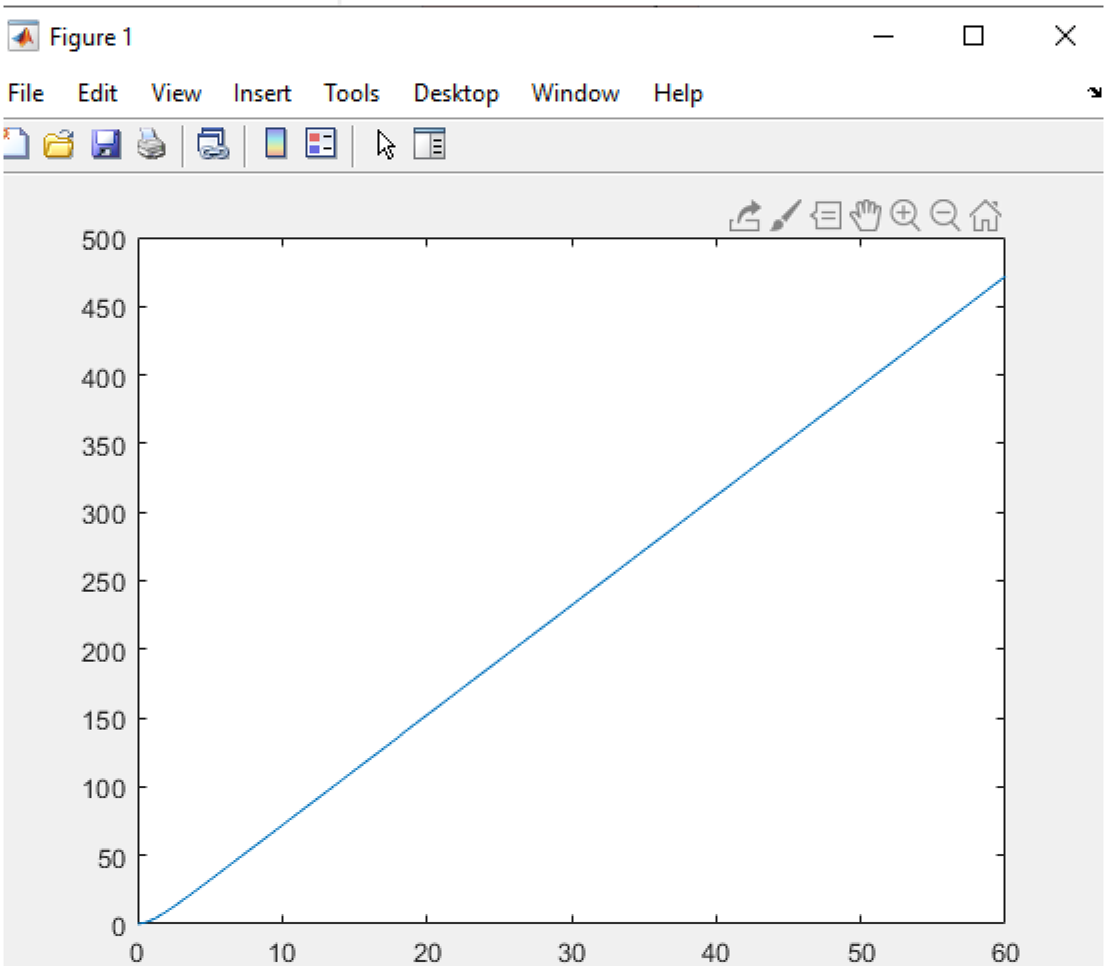
```
>> eq=[(8/(s+1)-(8/s)+(8/s^2))]

eq =

8/(s + 1) - 8/s + 8/s^2

>> ilaplace(eq)

ans =

8*t + 8*exp(-t) - 8
```

Entonces, la salida al aplicar una entrada rampa $\frac{1}{s^2}$ al sistema $C(s) = \frac{8}{s+1}$ es:

$$C(t) = 8t + 8e^{-t} - 8$$

Y graficamos el resultado:

```
>> t=linspace(0,60);
>> y=(8*t+8*exp(-t)-8);
>> plot(t,y);
>>
```

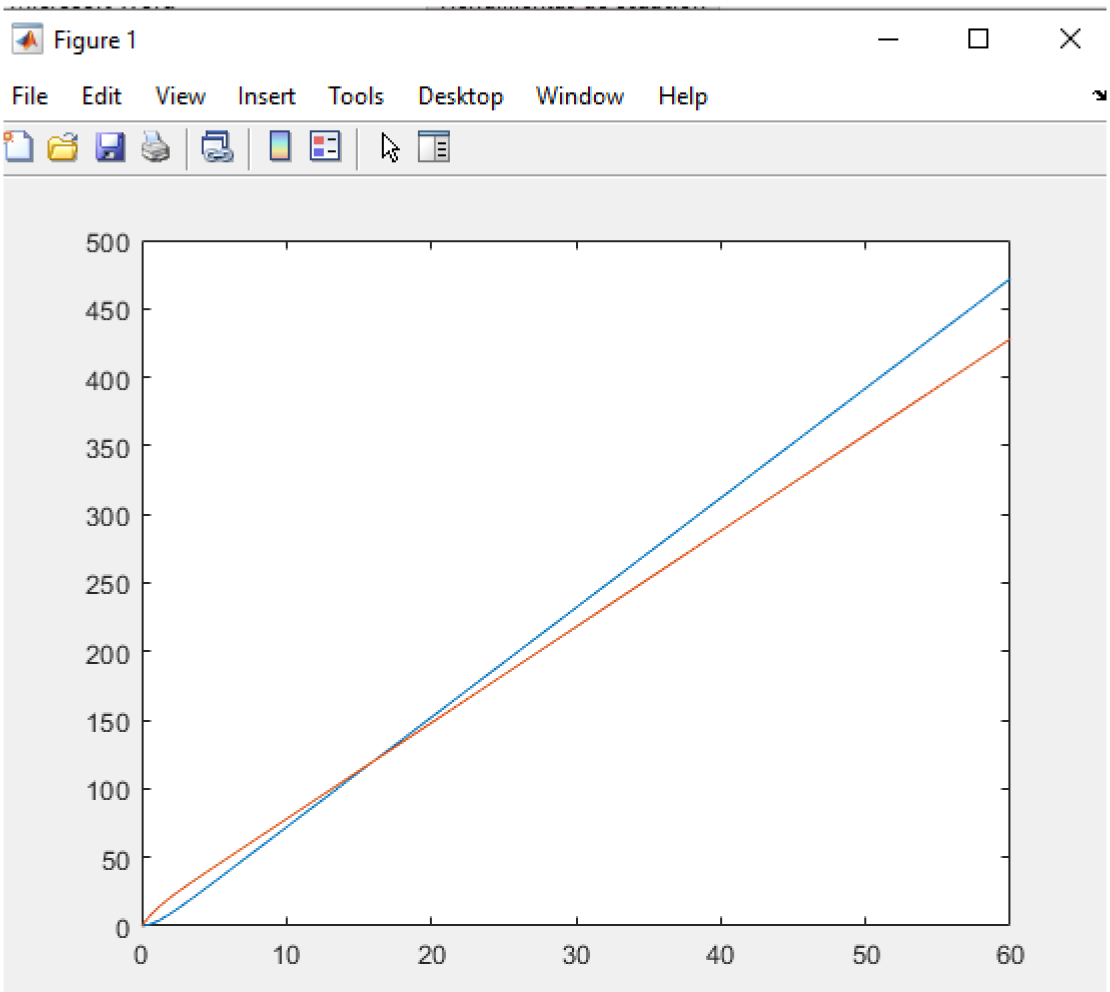


Calculamos el error:

$$\text{Error= c(r) – c(t)}$$
$$Error = \ t - (8t + \ 8e^{-t} - 8)$$
$$Error = \ t - 8t - 8\,e^{-t} + 8$$
$$Error = \ 7t - 8\,e^{-t} + 8$$

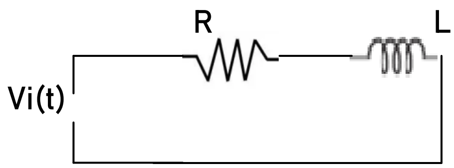
E implementamos esta parte para la graficar:

```
>> t=linspace(0,60);
>> y=(8*t+8*exp(-t)-8);
>> plot(t,y);
>> hold on
>> t2=linspace(0,60);
>> y2=(7*t-8*exp(-t)+8);
>> plot(t2,y2);
```



Comparando estas gráficas donde K=8 con las gráficas de K=1, podemos observar que al aumentar la ganancia del sistema de primer orden, el error disminuye considerablemente.

Por último, haremos un ejercicio más, el cuál solicita la corriente que pasa por el inductor en un tiempo t.



En donde:
Vi(t)= 15V
R=1 kΩ
iL(0)= 4mA
iL(t)=?

Primero vamos a determinar la ecuación modelo que rige al sistema de primer orden:

$$Vi(t) = Ri_L(t) + L\frac{di_L(t)}{dt}$$

Aplicamos Laplace para resolverla:

$$\mathcal{L}\{Vi(t)\} = \mathrm{R}\,\mathcal{L}\{i_L(t)\} + \mathrm{L}\,\mathcal{L}\left\{\frac{di_L(t)}{dt}\right\}$$

$$\frac{Vi(S)}{S} = RI_L(S) + L\left(SI_L(S) - I_L(0)\right)$$

$$\frac{Vi(S)}{S} = (R + LS)\,I_L(S) - LI_L(0) \dots\dots\dots (1)$$

De (1) despejamos a $I_L(S)$:

$$I_L(S) = \frac{\frac{VI(S)}{S} + LI_L(0)}{R + LS}$$

Le damos valores:

$$I_L(S) = \frac{\frac{15}{S} + (13x10^{-3})(4x10^{-3})}{(1x10^3) + (13x10^{-3})S}$$

$$I_L(S) = \frac{\frac{15V + S(13x10^{-3}H)(4x10^{-3}A)}{S}}{(1x10^3\Omega) + (13x10^{-3}H)S}$$

$$I_L(S) = \frac{15V + S(13x10^{-3}H)(4x10^{-3}A)}{S(\,(1x10^3\Omega) + (13x10^{-3}H)S)}$$

$$I_L(S) = \frac{15}{S(\,(1000\Omega) +\ 13x10^{-3}HS)} + \frac{S(5.2x10^{-3})}{S(\,(1000\Omega) + 13x10^{-3}HS)}$$

$$I_L(S) = \frac{15}{S(\,(1000\Omega) + 1\,3x10^{-3}HS)} + \frac{(5.2x10^{-3})}{(\,(1000\Omega) +\ 13x10^{-3}HS)}$$

Si descomponemos en fracciones parciales, tendríamos:

$$I_L(S) = \frac{15}{S(1000 +\ 13x10^{-3}S)} + \frac{(5.2x10^{-3})}{(1000 +\ 13x10^{-3}S)}$$

$$I_L(S) = \frac{15}{13x10^{-3}S^2 + 1000S} + \frac{(5.2x10^{-3})}{(1000 +\ 13x10^{-3}S)}$$

$$I_L(S) = \frac{A}{S} + \frac{B}{13x10^{-3}S + 1000} + \frac{(5.2x10^{-3})}{1000 +\ 13x10^{-3}S}$$

Encontramos las fracciones parciales por medio de Matlab:

```
>> syms s t;
>> syms A B;
>> num=[15];
>> den=[0.013 1000 0];
>> [r,p,k]=residue(num,den)

r =

   -0.0150
    0.0150


p =

   1.0e+04 *

   -7.6923
         0


k =

     []
```

Y a ese resultado debemos sumarle la fracción que quedó fuera de las fracciones parciales, entonces nos quedaría asi:

$$I_L(S) = \frac{0.015}{S} - \frac{1.95x10^{-4}}{13x10^{-3}S + 1000} + \frac{(5.2x10^{-3})}{1000 +\ 13x10^{-3}S}$$

Como podemos ver, las fracciones parciales obtenidas en Matlab no fueron las obtenidas en el punto anterior. Esto es porque Matlab intenta reducir la expresión. Si lo hacemos nosotros mismos sería dividir el segundo y tercer término entre $13x10^{-3}$:

$$I_L(S) = \frac{0.015}{S} - \frac{0.015}{S + 76.923x10^3} + \frac{0.004}{S + 76.923x10^3}$$

Y este sí es el resultado de las fracciones parciales dadas por Matlab

Después aplicamos Laplace inversa con ayuda de Matlab:

```
>> ec=[(0.015/s)-(0.015/(s+73923.077))+(0.004/(s+73923.077))]

ec =

3/(200*s) - 11/(1000*(s + 5079955170155037/68719476736))
```

```
>> ilaplace(ec)

ans =

3/200 - (11*exp(-(5079955170155037*t)/68719476736))/1000
```

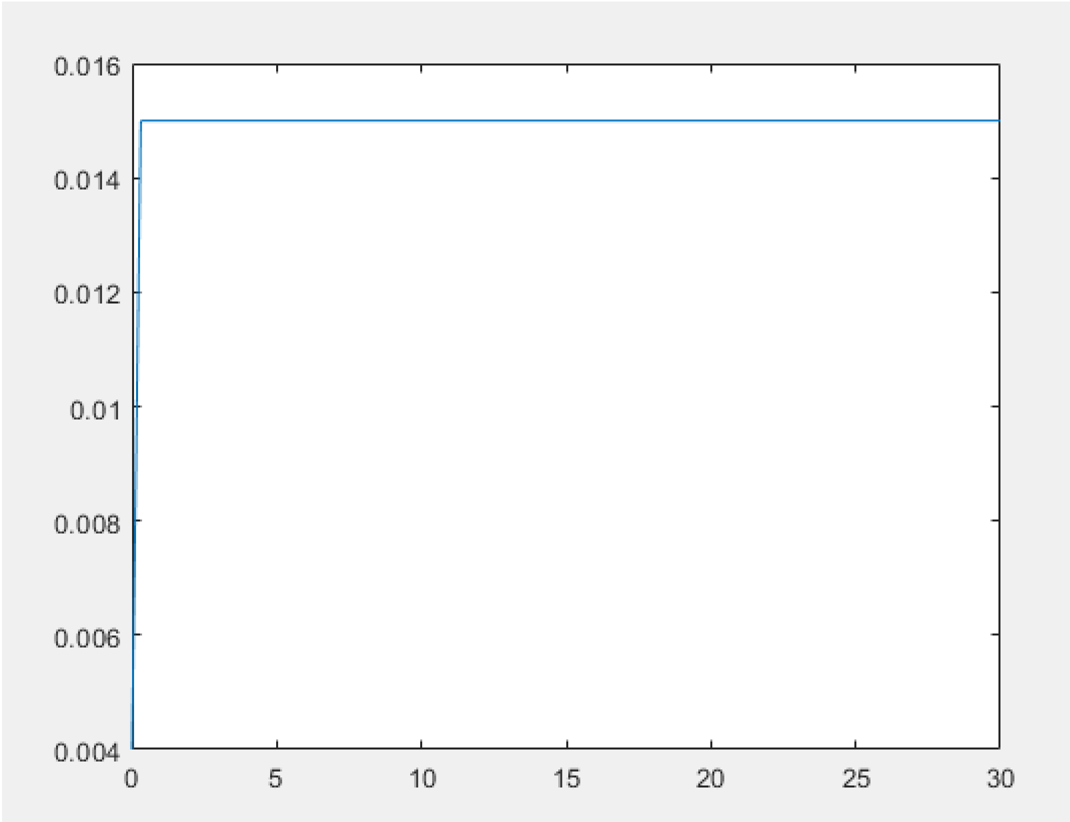
Simplificando la expresión obtenida en Matlab:

$$I_L(t) = \frac{3}{200} - \frac{11e^{-73923.077t}}{1000}$$

$$I_L(t) = 0.015 - 0.0\,11e^{-73923.077t}$$

Esta ecuación es la solicitada, la cual determina la corriente que pasa por L en cualquier instante de tiempo. Para tener un resultado más gráfico, recurrimos a Matlab una vez más:

```
>> t=linspace(0,30);
>> y=(3/200 - (11*exp(-(5079955170155037*t)/68719476736))/1000);
>> plot(t,y);
```



Tema 5: Sistemas de Segundo Orden

La forma general de los sistemas de segundo orden es la siguiente:

$$\tau^2 \frac{d^2y(t)}{dt} + 2\xi\tau \frac{dy(t)}{dt} + y(t) = ku(t)$$

En donde:

$$\tau^2 = \sqrt{\frac{a_1}{a_0}} \qquad 2\xi\tau = \frac{a_1}{a_0} \qquad k = \frac{b_0}{a_0}$$

Y reduciendo:

$$\frac{y(s)}{u(s)} = \frac{k}{\tau^2 s^2 + 2\xi\tau S + 1}$$

En este tipo de sistemas vamos a obtener diferentes tipos de comportamientos regidos por el factor de amortiguamiento. Éste depende de los polos o raíces del mismo sistema y serán nombrados de la siguiente manera:

Comportamiento Sub-amortiguado (0<ξ<1)
Este comportamiento se distingue cuando resultan 2 polos complejos conjugados.

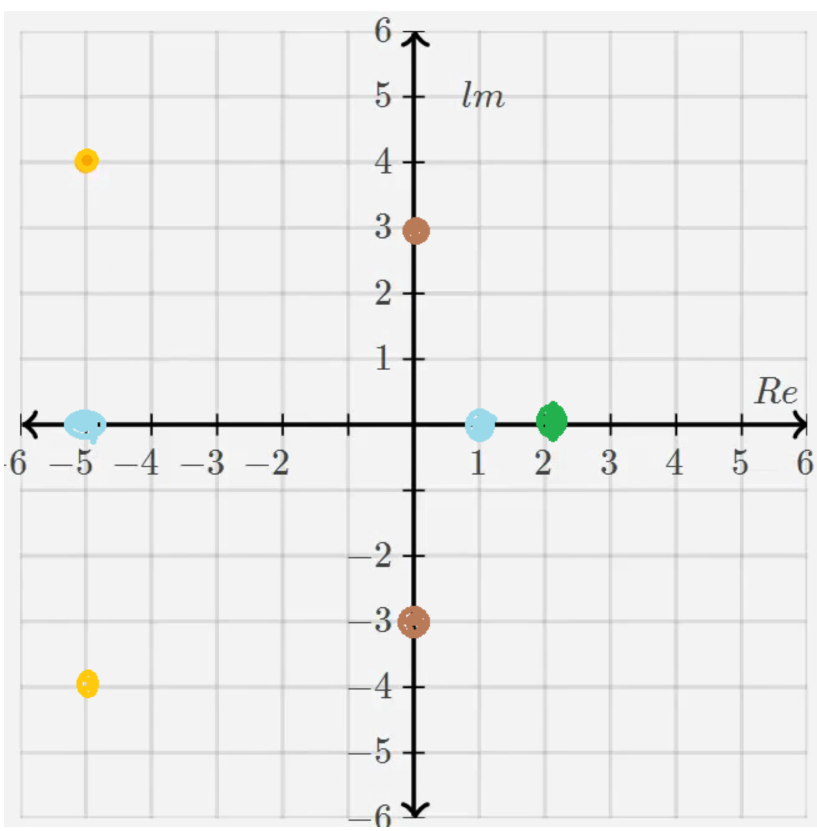
Comportamiento Críticamente- amortiguado (ξ=1)
Este comportamiento se distingue cuando resultan 2 polos reales iguales.

Comportamiento Sobre-amortiguado (ξ>1)
Este comportamiento se distingue cuando resultan 2 polos reales distintos.

Comportamiento No amortiguado (ξ=0)
Este comportamiento se distingue cuando resultan 2 polos complejos.

Los polos de estos comportamientos se ubicarían de la siguiente manera (el color con que están escritos los comportamientos corresponden al color del punto dentro del plano complejo):

Comportamiento Sub-amortiguado (0<ξ<1)
Comportamiento Críticamente- amortiguado (ξ=1)
Comportamiento Sobre-amortiguado (ξ>1)
Comportamiento No amortiguado (ξ=0)



Por esta razón es importante determinar los polos de este tipo de sistemas, pues nos dirán qué tipo de amortiguamiento se está presentando.

Ejemplo 1:

Determina el comportamiento del siguiente sistema y obtén la gráfica de la función de transferencia.

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

Todo el proceso se realizó por medio de Matlab. Los comandos utilizados ya se usaron anteriormente, pero hay algunos nuevos que se van a explicar a continuación.

El comando pole(ecuación) nos sirve para determinar los polos ó raíces de nuestra ecuación de transferencia, Esto resulta útil para saber de qué tipo de amortiguamiento se trata.

El comando pzmap(ecuación) hace gráfico el comando pole, es decir nos muestra los polos en una gráfica para que sea posible observar en donde se ubican exactamente éstos polos.

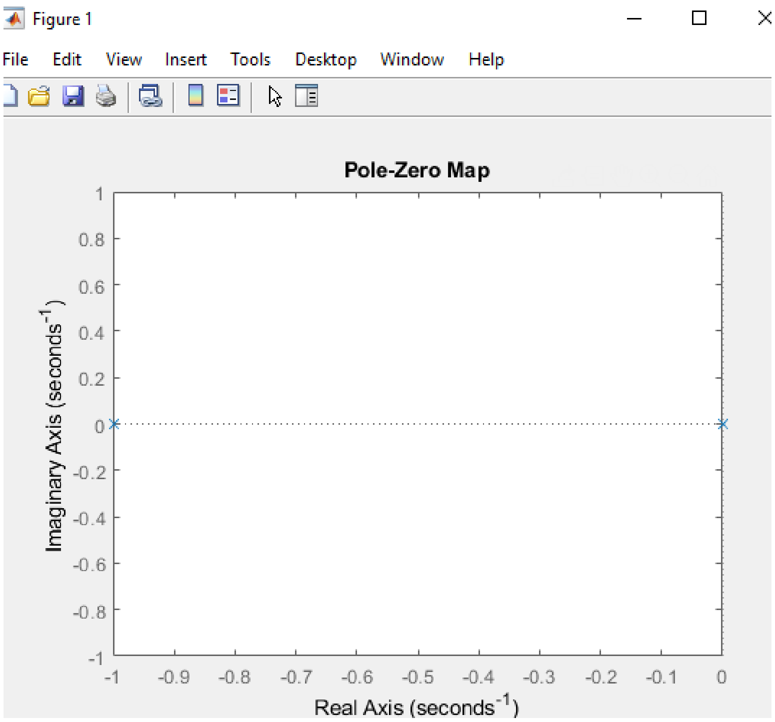
Dicho esto, ingresamos el numerador y denominador a Matlab como ya se ha hecho anteriormente en otros ejercicios. Indicamos que la ecuación que tenemos es una función de transferencia con el comando tf.

Después usamos el comando pole y pzmap para que nos dé los valores de los polos.

Y por último escribimos el comando step, que es el que nos dará la gráfica del comportamiento.

En Matlab se vería de la siguiente manera:

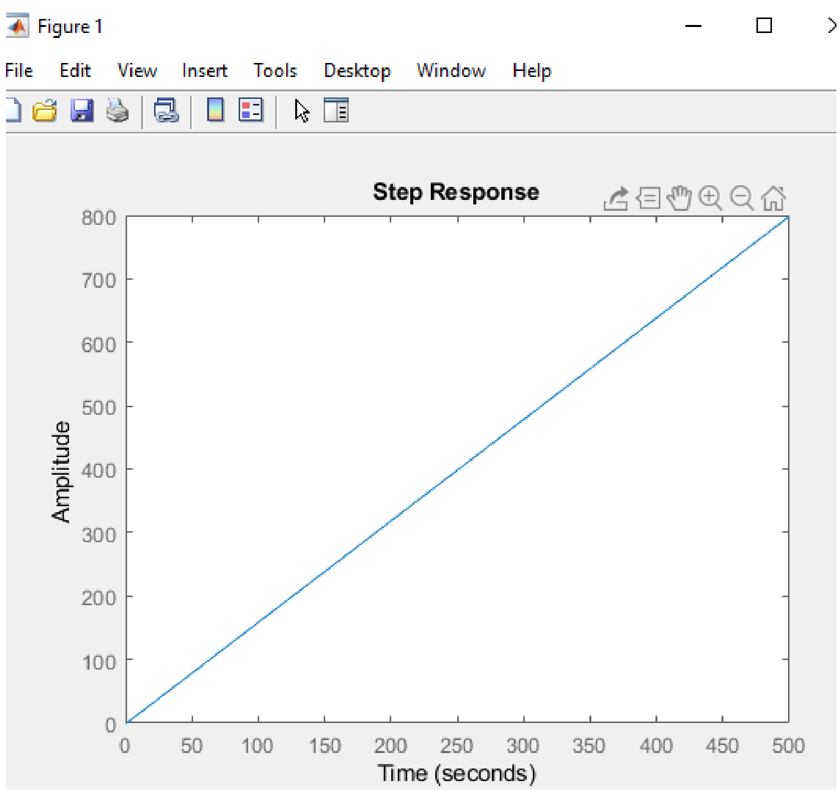
```
>> num=[8];
>> den=[5 5 0];
>> eq=tf(num,den);
>> pole(eq)

ans =


     0
    -1


>> pzmap(eq)
>> step(eq)
```



Polos:
$x_1 = 0$
$x_2 = -1$

Podemos observar que los polos de ésta función de transferencia de segundo orden son reales.

Esto significa que la función de transferencia tendrá un comportamiento sobre amortiguado, es decir, que el movimiento oscilatorio no ocurrirá puesto que el amortiguamiento es fuerte.



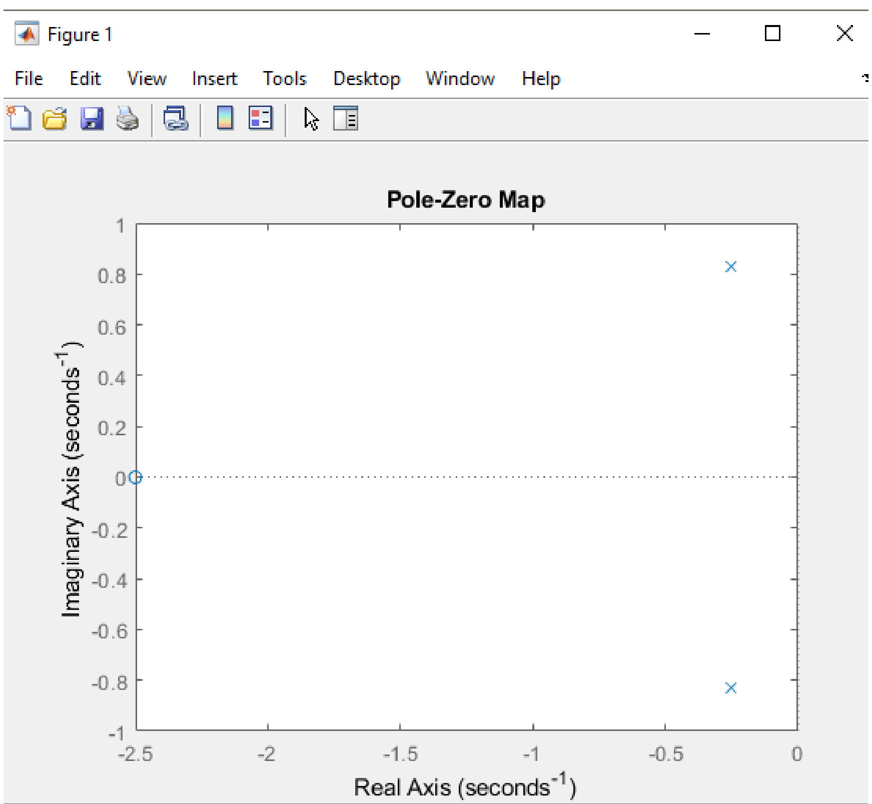
Ejemplo 2:

$$\frac{2s+5}{4s^2+2s+3}$$

Determina el comportamiento del siguiente sistema y obtén la gráfica de la función de transferencia.

De igual manera que se resolvió el problema anterior, se resolverá éste. Los comandos a utilizar son los mismos.

```
>>  num2=[2 5];
>> den2=[4 2 3];
>> eq2=tf(num2,den2);
>> step(eq2)
>> pole(eq2)

ans =

  -0.2500 + 0.8292i
  -0.2500 - 0.8292i
```



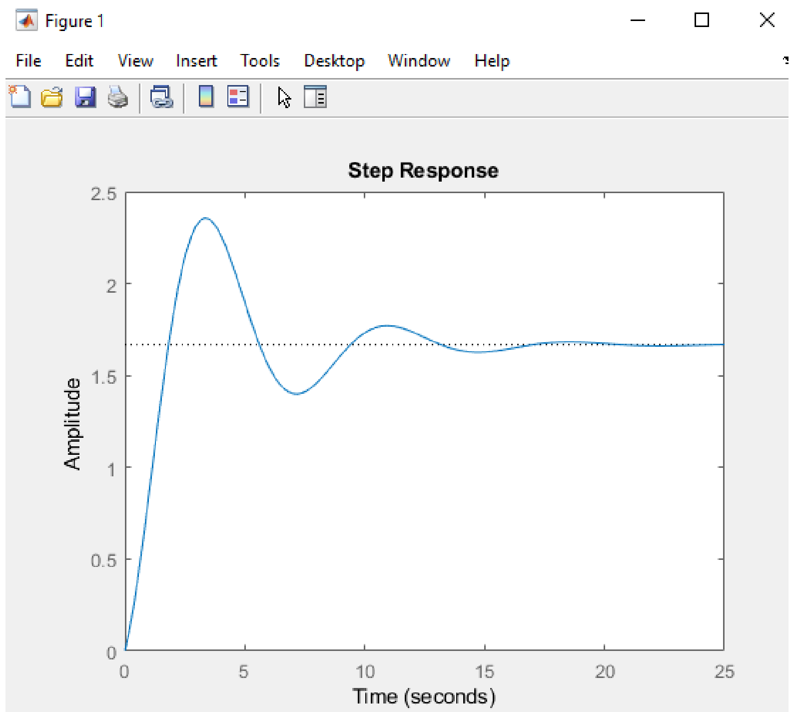
Polos:

$x_1 = -0.25 + 0.8292\ i$

$x_2 = -0.25 - 0.8292\ i$

Podemos observar que los polos de ésta función de transferencia de segundo orden son complejos conjugados.

Esto significa que la función de transferencia tendrá un comportamiento sub amortiguado, es decir, que el sistema tendrá un movimiento oscilatorio hasta que regresé a su posición de equilibrio.



Ejemplo 3:

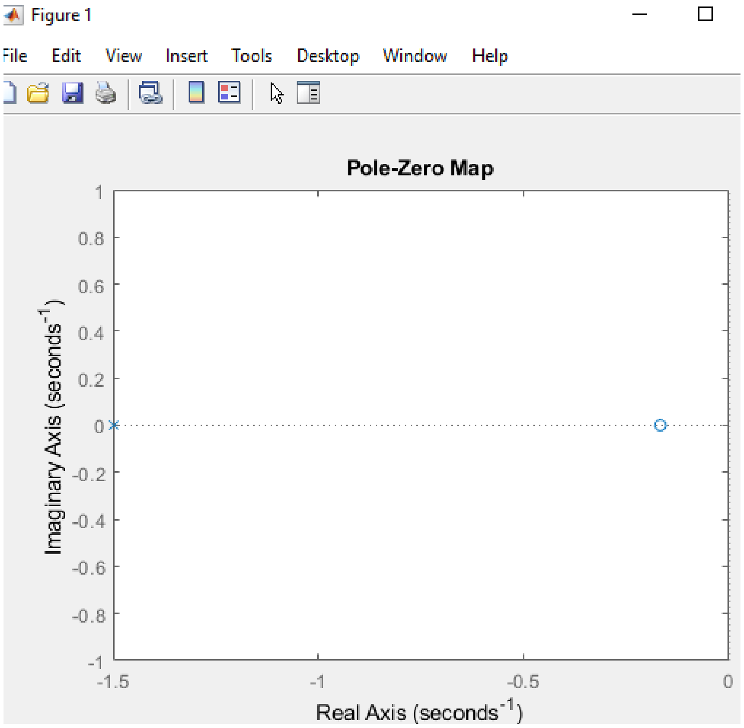
$$\frac{6s + 1}{4s^2 + 12s + 9}$$

Determina el comportamiento del siguiente sistema y obtén la gráfica de la función de transferencia.

De igual manera que se resolvió el problema anterior, se resolverá éste. Los comandos a utilizar son los mismos.

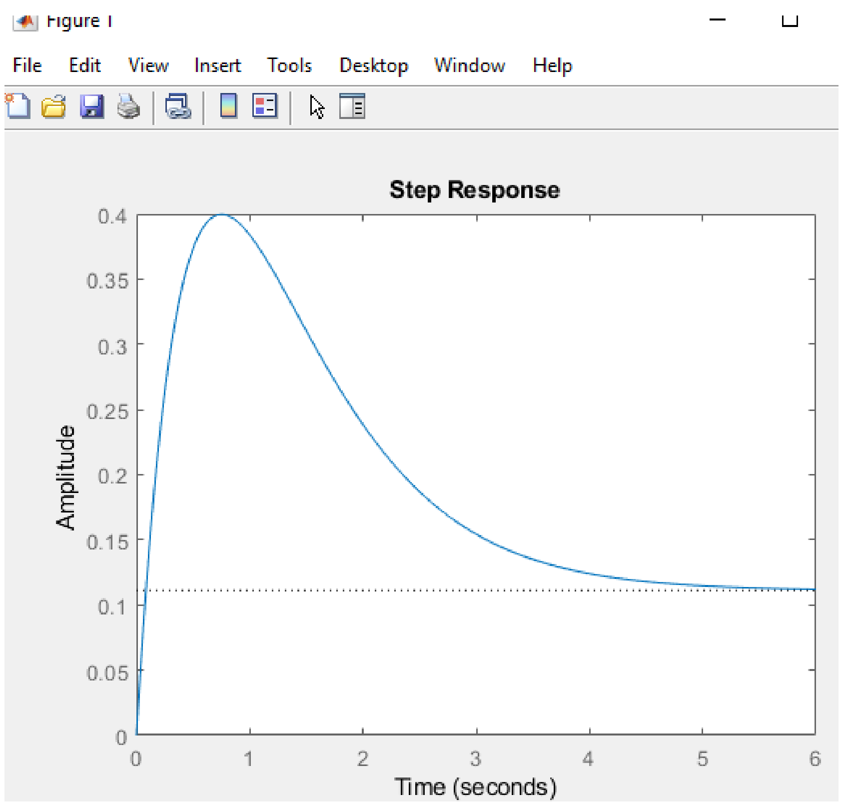
```
>> num3=[6 1];
>> den3=[4 12 9];
>> eq3=tf(num3,den3);
```

```
>> pole(eq3)

ans =

   -1.5000
   -1.5000

>> pzmap(eq3)
>> step(eq3)
```



Polos:
$x_1 = -1.5$
$x_2 = -1.5$

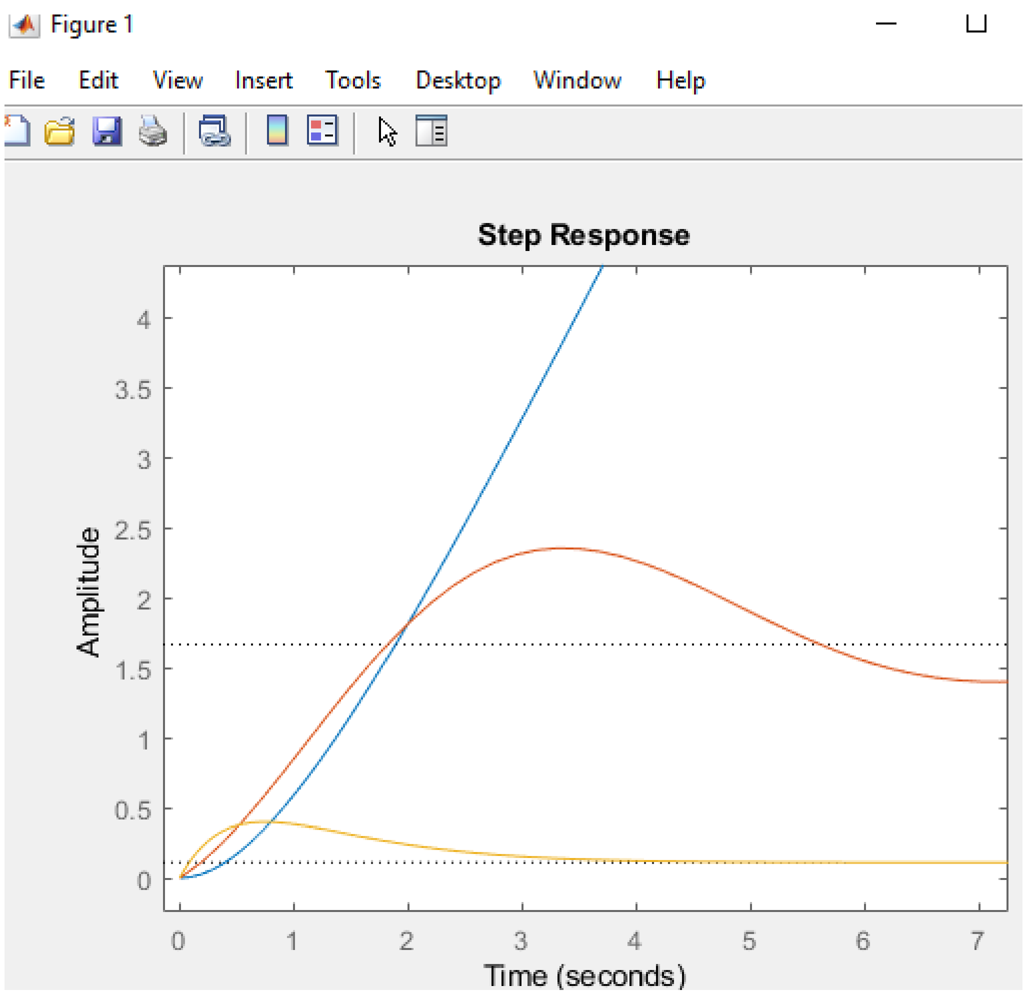
Podemos observar que los polos de ésta función de transferencia de segundo orden son reales y además son iguales.



Esto significa que la función de transferencia tendrá un comportamiento críticamente amortiguado, es decir, el sistema se encuentra en un estado estático, que cualquier variación en la fuerza de amortiguamiento el sistema pasaría a ser sobre-amortiguado (aumento), o sub-amortiguado (disminución); esto indica que al liberar la masa esta regresará a su posición de equilibrio estático sin ningún tipo de oscilación.

Usando el comando hold on, vamos a permitir que las tres gráficas de los tres sistemas realizados se mantengan en una sola, para ver las diferencias entre una y otra:

```
>> step(eq)
>> hold on
>> step(eq2)
>> hold on
>> step(eq3)
```



$$\frac{8}{5s^2 + 5s} \quad \textit{Sobre amortiguado}$$

$$\frac{2s+5}{4s^2+2s+3} \; Sub\ Amortiguado$$

$$\frac{6s+1}{4s^2+\ 12s+9} \; Críticamente\ Amortiguado$$

Tema 7: Respuesta en Frecuencia: Diagrama de Bode

Un diagrama de Bode es una representación gráfica que sirve para caracterizar la respuesta en frecuencia de un sistema. Normalmente consta de dos gráficas separadas, una que corresponde con la magnitud de dicha función y otra que corresponde con la fase.

La respuesta en amplitud y en fase de los diagramas de Bode no pueden por lo general cambiarse de forma independiente: cambiar la ganancia implica cambiar también desfase y viceversa.

Se conoce por respuesta en frecuencia, a la respuesta de un sistema, en régimen permanente, cuando se utiliza como señal de entrada una excitación senoidal de amplitud constante y de frecuencia variable desde cero hasta infinito. Tal cual se va a demostrar, la respuesta de un sistema LTI ante este tipo de excitación, es otra senoidal de la misma frecuencia que la entrada, pero que difiere en amplitud y fase.

Si la función de transferencia es una función racional, entonces el diagrama de Bode se puede aproximar con segmentos rectilíneos. Estas representaciones asintóticas son útiles porque se pueden dibujar a mano siguiendo una serie de sencillas reglas (y en algunos casos se pueden predecir incluso sin dibujar la gráfica). Esta aproximación se puede hacer más precisa corrigiendo el valor de las frecuencias de corte ("diagrama de Bode corregido").

Ejercicio 1: Obtén el diagrama de bode por medio de Matlab.

$$G(s)=\frac{(s+3)}{(s+6)(2s^2+\ 870s+1000)}$$

Primero explicaremos el análisis de manera teórica, ya que el procedimiento en Matlab es bastante sencillo y es necesario explicar algunos puntos de éste tipo de ejercicios. Iniciaremos por descomponer el término cuadrado del denominador $2s^2+\ 870s+1000$, esto se hace obteniendo sus polos y ceros, en Matlab lo podemos realizar como ya lo hemos hecho antes en otros ejercicios usando los comandos `polos=roots()` y `zeros=roots()` :

```
>> sist=tf([1 3],[2 882 6220 6000])

sist =

               s + 3
  -------------------------------
  2 s^3 + 882 s^2 + 6220 s + 6000

Continuous-time transfer function.

>> polos=roots(sist.den{1})

polos =

 -433.8475
   -6.0000
   -1.1525

>> zeros=roots(sist.num{1})

zeros =

    -3
```

Los polos son los que utilizaremos para descomponer el denominador. El cero, pertenece al numerador pero no lo usaremos por ahora, ya que la expresión del numerador ya se encuentra simplificada. Entonces, los polos serían los siguientes:

$$s_1 = -6 \qquad s_2 = -433.84 \qquad s_3 = -1.15$$

Nos resultaría la siguiente expresión:

$$G(s) = \frac{(s+3)}{(s+6)(s+1.15)(s+433.84)}$$

Para determinar la ganancia estática, deducimos que $s = 0$:

$$G(s) = \frac{(3)}{(6)(1.15)(433.34)} = 0.001$$

La ganancia entonces sería:

$$k = 0.001$$

Para llevar la ganancia k a decibeles, se utiliza $20\log(k)$:

$$20\log(0.001) = -60\ dB$$

Pasamos la expresión anterior al dominio de la frecuencia, es decir:

$$s = j\omega$$

Nos quedaría de la siguiente manera:

$$G(j\omega) = \frac{(j\omega+3)}{(j\omega+6)(j\omega+1.15)(j\omega+433.84)}$$

Esto nos sirve para poder calcular el margen de ganancia, de la siguiente manera:

$$|G(j\omega)| = \frac{\sqrt{((j\omega)^2 + (3)^2)}}{\sqrt{((j\omega)^2 + (6)^2)}\ \sqrt{((j\omega)^2 + (1.15)^2)}\ \ \sqrt{((j\omega)^2 + (433.84)^2)}}$$

Recordemos que un número imaginario al cuadrado es igual a uno: $j^2 = 1$. Entonces:

$$|G(j\omega)| = \frac{\sqrt{\omega^2 + 9}}{\sqrt{(\omega^2 + 36)}\ \sqrt{((\omega^2 + (1.15)^2)}\ \ \sqrt{(\omega^2 + (433.84)^2)}}$$

Para obtener la fase será por medio de arco-tangentes. Los arco-tangentes de todos los ceros (numerador) sumados y restando a los arco-tangentes de todos los polos (denominador) partiendo de la ecuación que ya teníamos anteriormente:

$$G(j\omega) = \frac{(\omega + 3)}{(\omega + 6)(\omega + 1.15)(\omega + 433.84)}$$

Nos quedaría de la siguiente manera:

$$\varphi\big(G(j\omega)\big) = \arctan\left(\frac{\omega}{3}\right) - \arctan\left(\frac{\omega}{6}\right) - \arctan\left(\frac{\omega}{1.15}\right) - \arctan\left(\frac{\omega}{433.84}\right)$$

Ahora, indicaremos los ceros y los polos en términos de la frecuencia, esto es:

$$\begin{aligned} s_0 &= -3 & &\rightarrow & \omega_0 &= 3 \\ s_1 &= -6 & &\rightarrow & \omega_1 &= 6 \\ s_2 &= -1.15 & &\rightarrow & \omega_2 &= 1.15 \\ s_3 &= -433.84 & &\rightarrow & \omega_3 &= 433.84 \end{aligned}$$

Esto es importante, porque al ubicar cada uno de los ceros y polos dentro del diagrama de Bode, observamos que en esos puntos hay un cambio en las gráficas.

Dentro del diagrama de Bode, mientras no haya alguna intersección de las señales con alguno de los polos o ceros, la señal será cien por ciento la ganancia para dicho sistema. Para este caso k=-60dB.

Por la parte del diagrama de Magnitud (dB), cada intersección con un polo, la señal se verá afectada cayendo 20dB por década o bien:

$$20 \log\left(\frac{\omega_{n-1}}{\omega_n}\right)$$

Siendo ω_n el polo con el que se está interceptando la señal y ω_{n-1} el polo más cercano a ω_n.

Cada intersección con un cero, se verá afectada aumentando 20dB por década.

En el caso del diagrama de la frecuencia, se encuentra en grados y el funcionamiento es el siguiente: Cada que la señal se encuentra con un cero, ésta aumentará 90° y cada que se encuentre con un polo, disminuirá 90°.

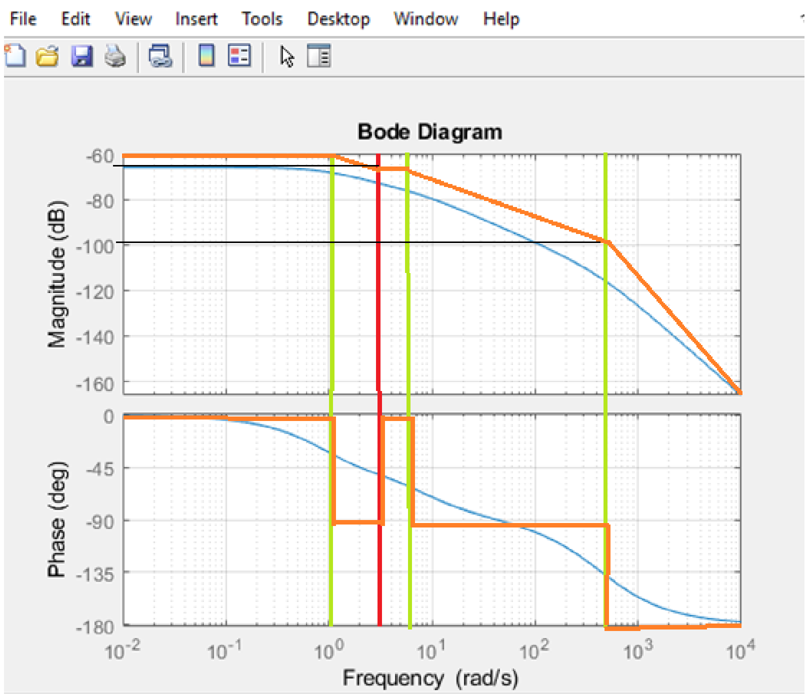
Cabe mencionar que la explicación anterior es usada para definir solamente las asíntotas de las señales resultantes y no es como se verán realmente. Es solo un procedimiento que nos dirá como será el comportamiento aproximado de éstas señales resultantes.

Procedemos a graficar el diagrama de Bode en Matlab de la siguiente manera:

Declaramos nuestro sistema de control con cualquier variable, especificando que es una ecuación de transferencia con el comando ft como ya lo hemos trabajado antes. Y usaremos también el comando bode() el cual nos va a arrojar el diagrama de bode representante de ése sistema.

```
>> sist=tf([1 3],conv([1 6],[2 870 1000]));
>> bode(sist),grid
```

Obtendremos el diagrama de Bode. Pero para su explicación por el método teórico se realizaron más trazos (asíntotas) para un mejor entendimiento que se explica a continuación.



Las líneas verticales verdes representan la ubicación aproximada de los polos:

$$\omega_1 = 6$$
$$\omega_2 = 1.15$$
$$\omega_3 = 433.84$$

La línea vertical roja representa la ubicación aproximada del cero:

$$\omega_0 = 3$$

Las líneas naranjas representan el trayecto aproximado de cada una de las señales.

La línea azul representa la señal real resultante.

Iniciando por el diagrama de Magnitud[dB} (también llamado atenuación ó amplificación), la señal permanecerá en -60dB, pues recordemos que ésa es la ganancia $20\log(0.001) = -60\ dB$. Será de ésta manera hasta que nos encontramos con el primer polo $\omega_2 = 1.15$ donde habrá una caída de decibeles:

$$20\log\left(\frac{\omega_0}{\omega_2}\right) = 20\log\left(\frac{3}{1.15}\right) = 8.32\ dB \rightarrow (-60db) - (8.32dB) = -68.32\ dB$$

Es por eso que en éste punto la señal cae a $-68.32\ dB$.Posteriormente nos encontramos con el cero $\omega_0 = 3$ donde aumentaremos los decibeles. Como ya los habíamos disminuido, entonces el trayecto que sigue es una línea recta hasta el siguiente polo.

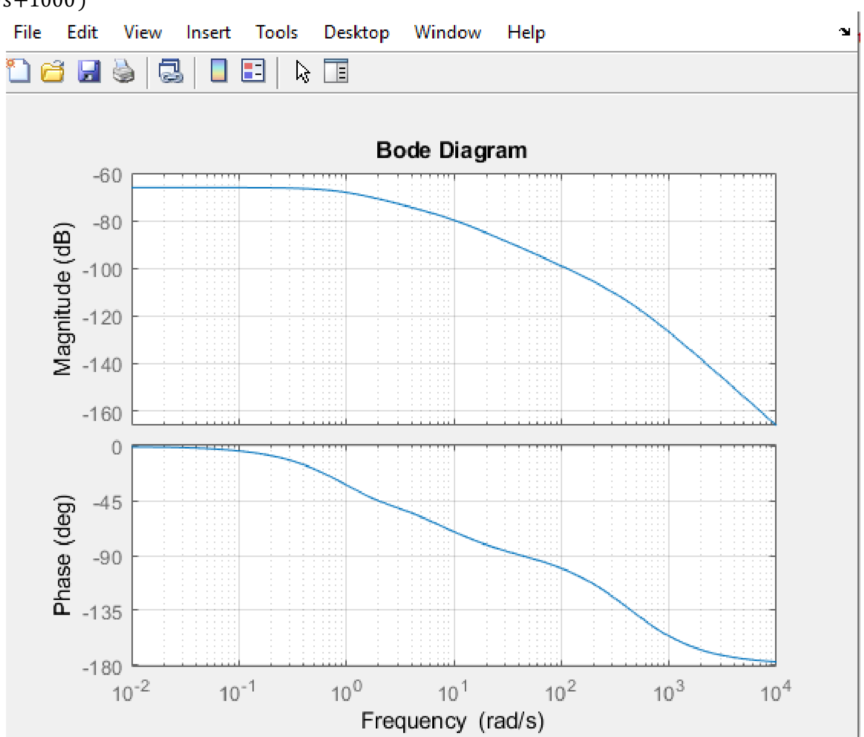
El polo que sigue corresponde a $\omega_1 = 6$ el cual indica que vuelve a haber una caída de decibeles:

$$20\log\left(\frac{\omega_3}{\omega_1}\right) = 20\log\left(\frac{433.84}{6}\right) = 37.82\ dB \rightarrow (-60db) - (37.82dB) = -97.82\ dB$$

Y como ahora estamos ubicados en el polo $\omega_3 = 433.84$ y ya no hay ningún otro, ahí termina el trayecto de los decibeles.

Por la parte del diagrama de fase, iniciamos en 0° y esto cambiará hasta que encontremos el primer polo $\omega_2 = 1.15$ el cuál hará disminuir 90°. En este momento nos encontramos en -90° y posteriormente está el cero $\omega_0 = 3$ el cuál aumenta 90°, es decir, volvemos a la posición 0°. Luego nos encontramos con otro polo: $\omega_1 = 6$ el cual nos hace disminuir de nuevo 90°, entonces estaríamos en la posición -90°. Por último está el último polo $\omega_3 = 433.84$ con el cual disminuiremos de nuevo 90°. Nos encontramos en -180° y ahí se mantiene hasta el final porque ya no hay otro cero o polo.

Recordemos que estas líneas marcadas solamente sirven de guía muy aproximada a las señales resultantes reales. El diagrama de Bode corregido (Matlab nos da el diagrama de Bode corrigiendo las frecuencias de corte) para $G(s) = \frac{(s+3)}{(s+6)(2s^2+870s+1000)}$ es:



Ejercicio 2: Obtén el diagrama de bode por medio de Matlab

$$H(s) = \frac{s(s+2)}{(s+4)}$$

Lo primero que haremos es encontrar los polos y ceros de ésta función. Como es muy simple, puede que sean muy visibles a simple vista, pero aun así usaremos Matlab para determinarlos de la mima manera que se hizo con el ejercicio anterior:

```
>> polos=roots(eq.den{1})

polos =

    -4

>> zeros=roots(eq.num{1})

zeros =

     0
    -2
```

Podemos descomponer la función de transferencia de la siguiente manera:

$$H(s) = \frac{s(s+2)}{(s+4)}$$

En este caso, para obtener la ganancia no podemos suponer que s=0 porque nos quedaría $\frac{0}{4}$ y no lo podríamos resolver. Lo que se hace en éste caso, es seguir descomponiendo la función de transferencia buscando que el numerador y denominador tengan la forma $\frac{S}{N}+1$ (en donde N=valor numérico):

$$H(s) = \frac{s(s+2)}{(s+4)} = \frac{s(s\frac{2}{2}+2)}{\left(s\frac{4}{4}+4\right)} = \frac{2S\left(\frac{S}{2}+\ 1\right)}{4\left(\frac{S}{4}+1\right)}$$

Como podemos ver, con esto se modifica la función, pero sigue siendo la misma, solo expresada de una manera que resulte más sencilla la localización de K. Aun podemos simplificar más:

$$H(s) = \frac{2S\left(\frac{S}{2}+\ 1\right)}{4\left(\frac{S}{4}+1\right)} = \frac{0.5s\left(\frac{S}{2}+\ 1\right)}{\frac{S}{4}+1}$$

Y lo ordenamos, siendo positivos los ceros y negativos los polos, así como el ejercicio anterior:

$$20\log H(s) = 20\log(0.5) +\ 20\log(s) +\ 20\log(\frac{S}{2}+1) - 20\log(\frac{S}{4}+1)$$

Podemos observar que el término 0.5 es, realmente, nuestra constante K:

$$H(s) = \frac{0.5s\left(\frac{s}{2} + 1\right)}{\frac{s}{4} + 1}$$

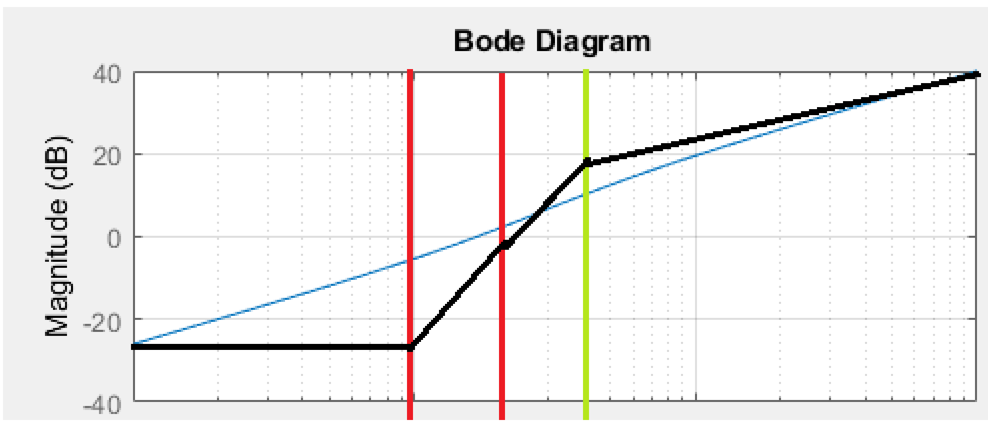
Entonces ya podemos conocer la ganancia de nuestra función:

$$20\log(K) \rightarrow 20\log(0.5) = -6.02\ dB$$

Entonces hasta éste momento ya tenemos los polos, los ceros y la ganancia. Estos datos son los necesarios para realizar nuestro diagrama de Bode de la misma manera que lo explicamos en el ejercicio anterior. Estos datos nos servirán para marcar de manera asintótica las señales resultantes en el diagrama. Así que recurrimos a Matlab, en donde realizaremos el diagrama de Bode con los comandos ya usados anteriormente:

```
>> num=[1 2 0];
>> den=[1 4];
>> eq=tf(num,den);
>> bode(eq), grid
```

Iniciamos con el diagrama de Magnitud dB o ganancia y atenuación:



Las líneas rojas corresponden a la ubicación de los ceros $\omega_2 = 0\ y\ \omega_3 = 2$.

La línea verde corresponde al polo $\omega_1 = 4$.

$$H(s) = \frac{0.5s\left(\frac{s}{2} + 1\right)}{\frac{s}{4} + 1}$$

Como nuestra ganancia se consiguió por medio de la constante 0.5:

$$20\log(0.5) = -6.02\ dB$$

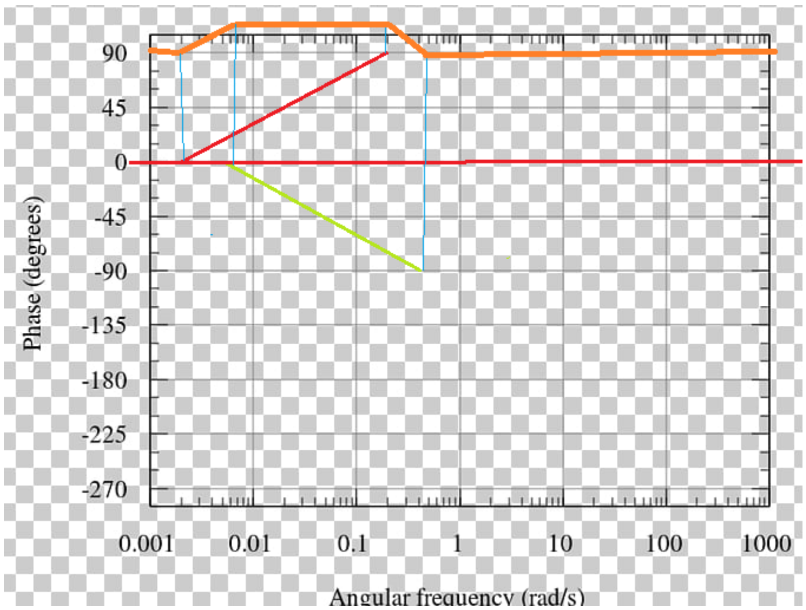
Esto nos indica que nuestra señal asíntota iniciará en -20dB, pero si le sumamos la ganancia, entonces nuestra señal comienza realmente en -26.02dB. En este caso nos encontramos otro cero $\omega_2 = 0$ el cual se obtuvo de la constante:

$$0.5s = 0 \rightarrow s = 0$$

Cuando esto pasa, no habrá ganancia en la señal. Es decir desde el punto de inicio (-26.02dB) hasta el primer cero $\omega_2 = 0$, la señal asíntota será una línea recta horizontal. Posteriormente nos encontramos con otro cero, el cual se obtuvo de la forma $\frac{s}{2}+\ 1$ la cual corresponde a $\frac{s}{\omega}+1$ y esto indica que habrá una ganancia de 20dB hasta el próximo polo o cero. Entonces aquí la línea asíntota incrementa 20 dB desde el cero $\omega_2 = 0$ hasta el cero $\omega_3 = 2$. Como en $\omega_3 = 2$ es el mismo caso, entonces habrá otra ganancia de 20dB hasta el próximo polo o cero.

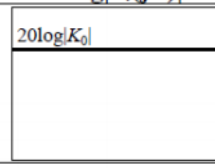
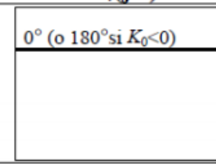
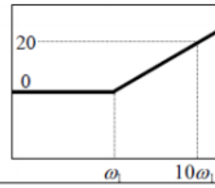
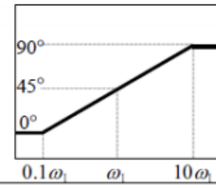
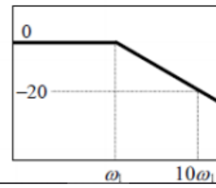
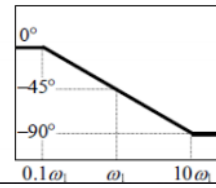
Por último nos encontramos con un polo $\omega_1 = 4$ obtenido de la forma $\frac{s}{4}+1$. Aquí vuelve a haber un incremento de 20dB y este incremento será de este punto en adelante, pues ya no existen mas polos o ceros para esta función de transferencia.

Para el diagrama de fase, se tiene lo siguiente:



La línea verde indica el polo $\omega_1 = 4$.
Las líneas rojas corresponden a la ubicación de los ceros $\omega_2 = 0\ y\ \omega_3 = 2$.
La línea naranja es la línea asíntota de señal resultante.

Las líneas azules solo se usan para señalar hasta dónde llegará nuestro comportamiento de la señal asíntota indicando el inicio y fin de los polos y ceros.

Término	$20\log\lvert T_i(j\omega)\rvert$	$\angle T_i(j\omega)$
Constante K_0		
Cero $\frac{s}{\omega_1}+1$		
Polo $\frac{1}{\frac{s}{\omega_1}+1}$		

Recordemos que la ganancia la obtuvimos a partir de una constante, esto nos indica que nuestro punto de inicio serán 90° sumados a los 0°, es decir, 90° + 0° = 90°. La línea permanecerá a 0° (puesto que $\omega_2 = 0$) hasta llegar a nuestro próximo cero. Nos encontramos con $\omega_3 = 2$ y esto nos indica que nuestra señal crecerá 45° (porque se puede apreciar que la línea de $\omega_3 = 2$ esta ascendiendo, por eso el valor es de +45°) hasta el próximo polo o cero.

Después encontramos el inicio de polo $\omega_1 = 4$ y esto nos indica que va a decrecer 45° (pues la línea de $\omega_1 = 4$ está en descenso y por eso el valor de -45°)y como la señal ya estaba creciendo 45°, entonces +45° - 45° = 0°. Lo que india que nuestra señal en este punto será una línea recta horizontal hasta el próximo inicio o fin de polos o ceros.

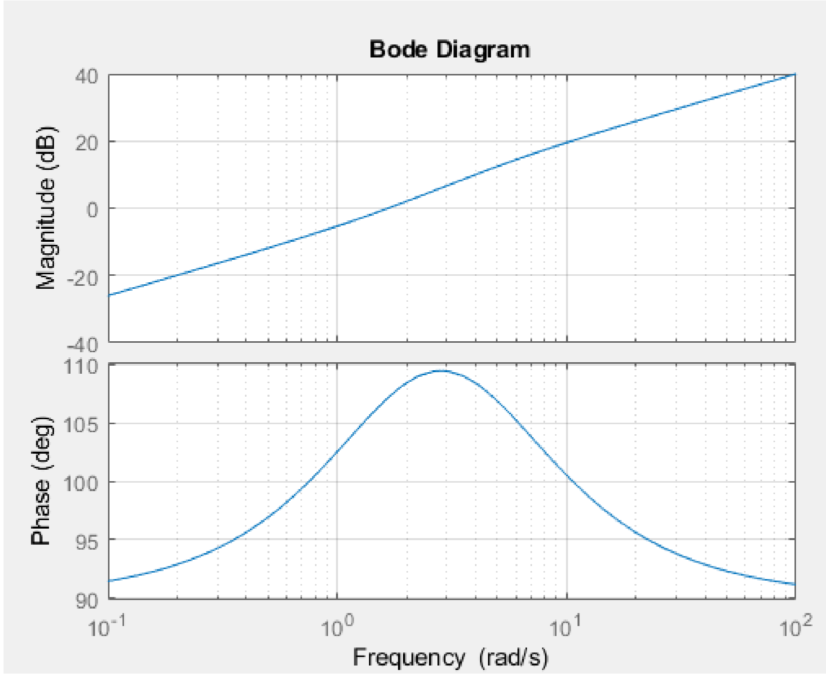
Después vemos que la línea del cero $\omega_3 = 2$ detiene su ascenso, esto provoca que nuestra señal sufra un cambio de -45°. Ahora la señal se desvía a 0° - 45° = -45°.

Por último podemos observar que la línea del polo $\omega_1 = 4$ detiene su descenso, esto quiere decir que nos otorga +45°. Entonces la señal sufre un cambio de – 45° que ya teníamos + 45°, es decir, -45° + 45° = 0°.

Entonces nuestra señal asíntota continúa en una dirección de 0° hasta encontrar algún otro polo o cero. Pero como ya son todos los polos y ceros para esta función de transferencia continúa en 0° hasta el final.

En Matlab, el diagrama de Bode nos quedaría de la siguiente manera:

```
>> num=[1 2 0];
>> den=[1 4];
>> eq=tf(num,den);
>> bode(eq), grid
```



Ejemplo 3: Encuentra el diagrama de bode que representa la siguiente función de transferencia:

$$I_{(s)} = \frac{36}{9 + 4s}$$

Lo primero que haremos es encontrar los polos y los celos de ésta función de transferencia. Es evidente que la función no tendrá ceros, pues no hay ningún término s en el numerador. En este caso solamente obtendremos un polo y podríamos deducirlo fácilmente haciendo un despeje. Pero, como éste tutorial es sobre la aplicación de estos problemas en Matlab, lo usaremos de nuevo para obtener el polo. Se ingresan los datos a Matlab de la siguiente manera:

```
>> Is=tf([36],[4 9])

Is =

    36
  -------
  4 s + 9

Continuous-time transfer function.
```

```
>> polos=roots(Is.den{1})

polos =

   -2.2500

>> zeros=roots(Is.num{1})

zeros =

  0×1 empty double column vector
```

Lo que nos indica que no hay ceros en ésta función y solamente un polo: $\omega = 2{,}25$

Después debemos de obtener la ganancia de nuestro sistema de transferencia, para ello haremos s=0 sobre la función:

$$I_{(s)} = \frac{36}{9+4s} = \frac{36}{9+4(0)} = \frac{36}{9} = 4$$

$$k = 4$$

Para llevar la ganancia k a decibeles, se utiliza $20\log(k)$:

$$20\log(4) = 12.04 \ dB$$

Pasamos la expresión anterior al dominio de la frecuencia, es decir:

$$s = j\omega$$

Nos quedaría de la siguiente manera:

$$I(j\omega) = \frac{(36)}{(4j\omega + 9)}$$

Esto nos sirve para poder calcular el margen de ganancia, de la siguiente manera:

$$|I(j\omega)| = \frac{\sqrt{(36)^2}}{\sqrt{((4j\omega)^2 + (9)^2)}}$$

Recordemos que un número imaginario al cuadrado es igual a uno: $j^2 = 1$. Entonces:

$$|I(j\omega)| = \frac{\sqrt{1296}}{\sqrt{(16\omega^2 + 81)}}$$

Para obtener la fase será por medio de arco-tangentes. Los arco-tangentes de todos los ceros (numerador) sumados y restando a los arco-tangentes de todos los polos (denominador) partiendo de la ecuación que ya teníamos anteriormente:

$$|I(j\omega)| = \frac{36}{4\omega + 9}$$

Nos quedaría de la siguiente manera:

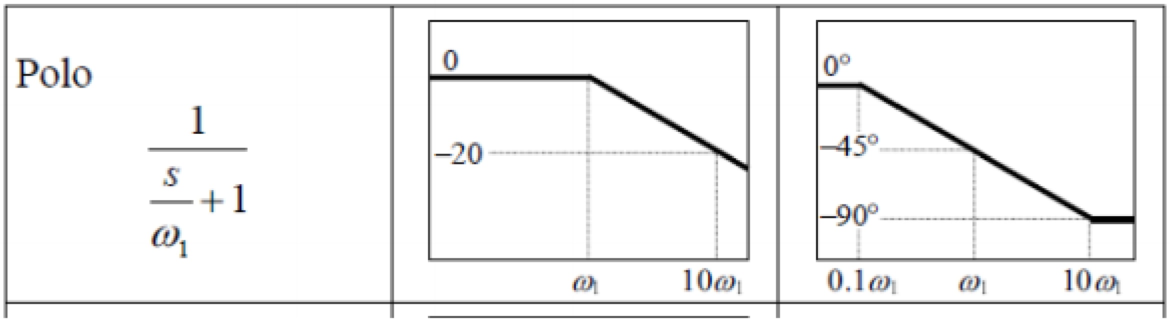
$$\varphi\big(G(j\omega)\big) = \arctan - \arctan\left(\frac{\omega}{6}\right) - \arctan\left(\frac{\omega}{1.15}\right) - \arctan\left(\frac{\omega}{433.84}\right)$$

$$\varphi\big(I(j\omega)\big) = -4\arctan\left(\frac{\omega}{9}\right)$$

Antes de obtener el diagrama de Bode en Matlab, haremos el trazo de las asíntotas, para ver mas o menos el comportamiento de ésta señal. Para ello debemos revisar que forma tiene nuestros polos y ceros y compararlos con las tablas. En este caso solamente tenemos un polo y en la función de transferencia original no parece tener ninguna forma parecida a como viene en las tablas, asi que haremos un poco de algebra para hacer que la expresión del polo se parezca a alguna expresión de as tablas utilizadas. Se hace lo siguiente:

$$I_{(s)} = \frac{36}{9+4s} = \frac{36}{9s\frac{4}{4}+4} = \frac{36}{4(\frac{9s}{4}+1)} = \frac{9}{(\frac{9s}{4}+1)}$$

Ahora vemo que nuestro polo ya tiene la forma:



Lo cual indica que, al momento que nuestra señal pase por dicho polo, en la parte de la ganancia se va a ver afectada, disminuyendo 20dB y en el diagrama de fase, la señal también se verá afectada disminuyendo 90°.

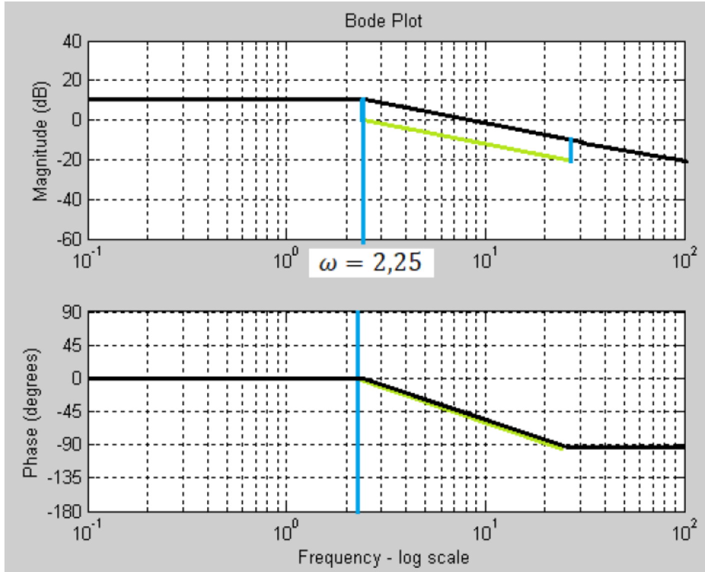
Entonces, en el diagrama de Magnitud (dB) nuestra señal comenzaría en 12.04 dB Porque:

$$20\log(k) = 20\log(4) = 12.04 \ dB$$

Al ser una constante, la señal (línea asíntota negra) permanecerá en esa dirección hasta encontrarse con el inicio del polo. Recordemos que el polo inicia en $\omega = 2.25$ y, por su forma, cae 20 dB y su longitud es de 10 décadas (observar línea verde en diagrama). Esto quiere decir, que al momento que la señal pase por el polo, va a decaer 20 dB y así lo hará hasta el final porque no hay ningún otro polo o cero que cambie su comportamiento.

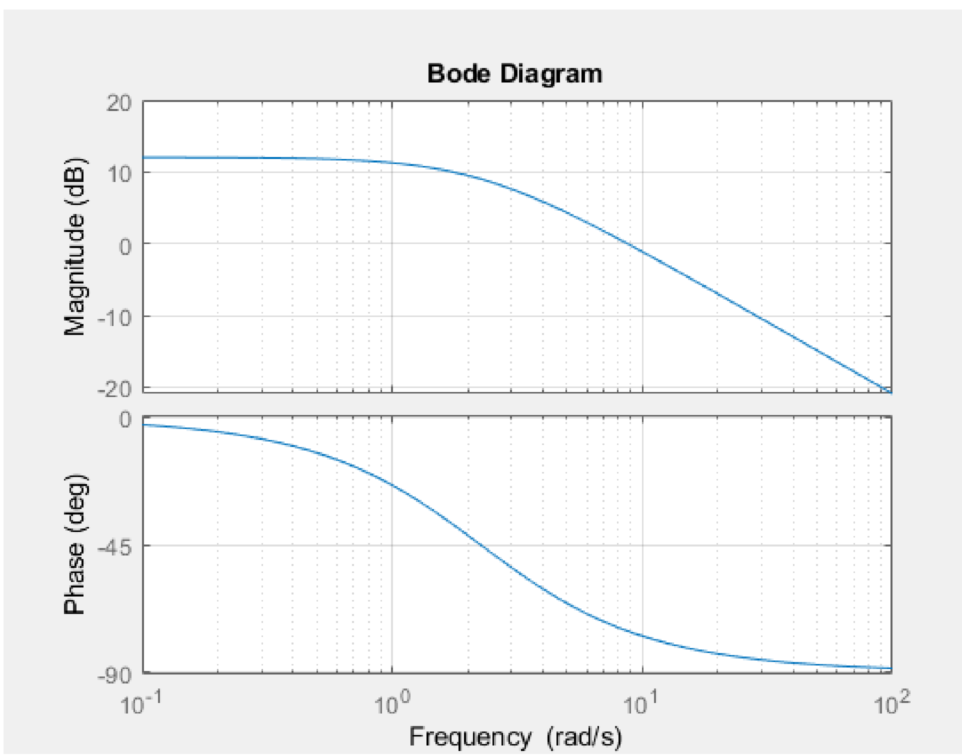
En el caso del diagrama de fase, la forma del polo $\omega = 2.25$ indica que la señal comenzará en 0°. Este polo comienza en 0° y decae a 90° en su trayectoria de 10 décadas (línea verde en diagrama de fase). Entonces, la señal asíntota iniciará en 0°, al encontrarse con el polo, ésta igual va a caer 90° (0° - 90°= -90°). Al terminar el polo, la señal se recupera (-90° + 90° = 0°) y sigue su trayecto en un ángulo de 0° hasta el final, pues no hay ningún otro polo o cero que altere su comportamiento.

Nos quedaría un diagrama de la siguiente manera:



Para comprobar esto en Matlab, declaramos nuestro sistema de control, especificando que es una ecuación de transferencia con el comando ft como ya lo hemos trabajado antes. Y usaremos también el comando bode() el cual nos va a arrojar el diagrama de bode representante de ése sistema.

```
>> num=[36];
>> den=[4 9];
>> Is=tf(num,den);
>> bode(Is), grid
```



Controlador Proporcional (P)

En estos controladores la señal de accionamiento es proporcional a la señal de error del sistema. La señal de error es la obtenida en la salida del comparador entre la señal de referencia y la señal realimentada. Es el más sencillo de los distintos tipos de control y consiste en amplificar la señal de error antes de aplicarla a la planta o proceso. La función de transferencia de este tipo de reguladores es una variable real, denominada Kp (constante de proporcionalidad) que determinará el grado de amplificación del elemento de control. Si y(t) es la señal de salida (salida del controlador) y e(t) la señal de error (entrada al controlador), en un sistema de control proporcional tendremos:

$$y(t) = K_p \cdot e(t)$$

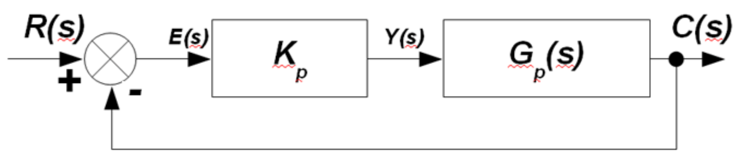
Que en el dominio de Laplace, será:

$$Y(s) = K_p \cdot E(s)$$

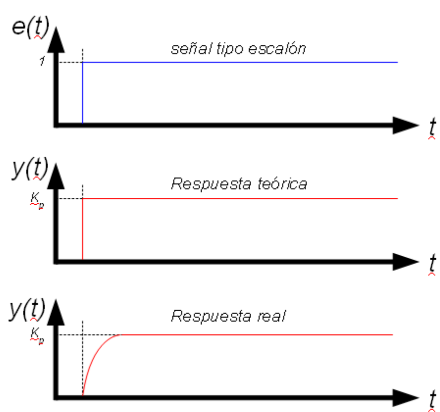
Por lo que su función de transferencia será:

$$G(s) = \frac{Y(s)}{E(s)} = K_p$$

Donde Y(s) es la salida del regulador o controlador, E(s) la señal de error y K_p la ganancia del bloque de control.



Teóricamente, en este tipo de controlador, si la señal de error es cero, también lo será la salida del controlador. La respuesta, en teoría es instantánea, con lo cual el tiempo no intervendría en el control. En la práctica, no ocurre esto, si la variación de la señal de entrada es muy rápida, el controlador no puede seguir dicha variación y presentará una trayectoria exponencial hasta alcanzar la salida deseada.



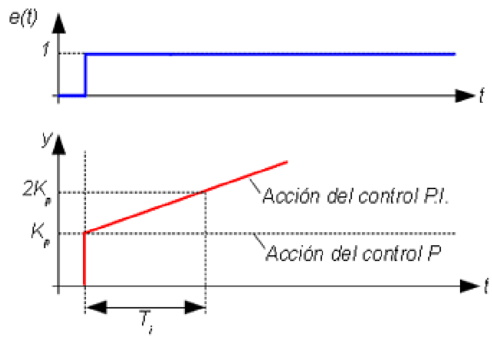
Controlador Proporcional Integral (PI)

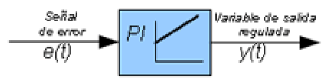
En realidad no existen controladores que actúen únicamente con acción integral, siempre actúan en combinación con reguladores de una acción proporcional, complementándose los dos tipos de reguladores, primero entra en acción el regulador proporcional (instantáneamente) mientras que el integral actúa durante un intervalo de tiempo. (Ti= tiempo integral)
La Función de transferencia del bloque de control PI responde a la ecuación:

$$G(s)=\frac{Y(s)}{E(s)}=K_P\cdot(\frac{1}{T_i\cdot s}+1)$$

Donde Kp y Ti son parámetros que se pueden modificar según las necesidades del sistema. Si Ti es grande la pendiente de la rampa, correspondiente al efecto integral será pequeña y, su efecto será atenuado, y viceversa.

Respuesta temporal de un regulador PI.





Por lo tanto la respuesta de un regulador PI será la suma de las respuestas debidas a un control proporcional P, que será instantánea a detección de la señal de error, y con un cierto retardo entrará en acción el control integral I, que será el encargado de anular totalmente la señal de error.

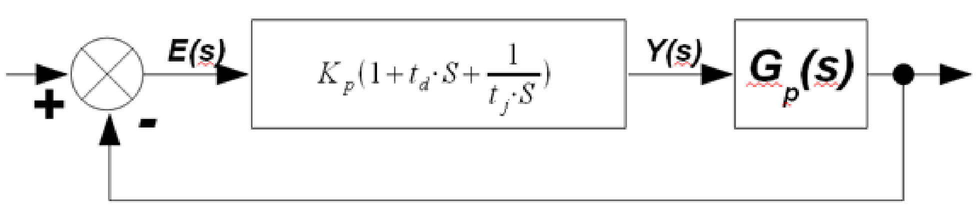
Controlador de acción Proporcional Integral y Derivativa (PID)

Es un sistema de regulación que trata de aprovechar las ventajas de cada uno de los controladores de acciones básicas, de manera, que si la señal de error varía lentamente en el tiempo, predomina la acción proporcional e integral y, mientras que si la señal de error varía rápidamente, predomina la acción derivativa. Tiene la ventaja de ofrecer una respuesta muy rápida y una compensación de la señal de error inmediata en el caso de perturbaciones. Presenta el inconveniente de que este sistema es muy propenso a oscilar y los ajustes de los parámetros son mucho más difíciles de realizar.

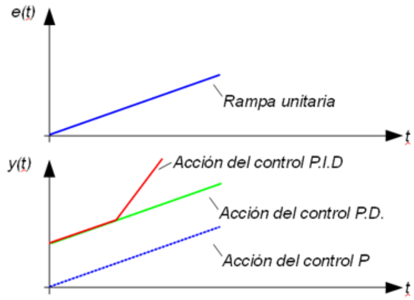
Y por tanto la función de transferencia del bloque de control PID será:

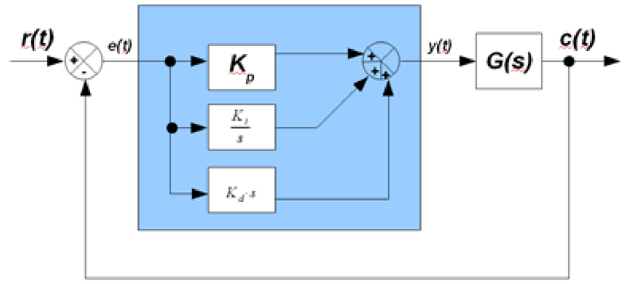
$$G(s)=\frac{Y(s)}{E(s)}=K_P(a+T_d\cdot s+\frac{1}{T_i\cdot s})$$

Donde K_p, T_i y T_d son parámetros ajustables del sistema.



La respuesta temporal de un regulador PID sería la mostrada en la figura siguiente:





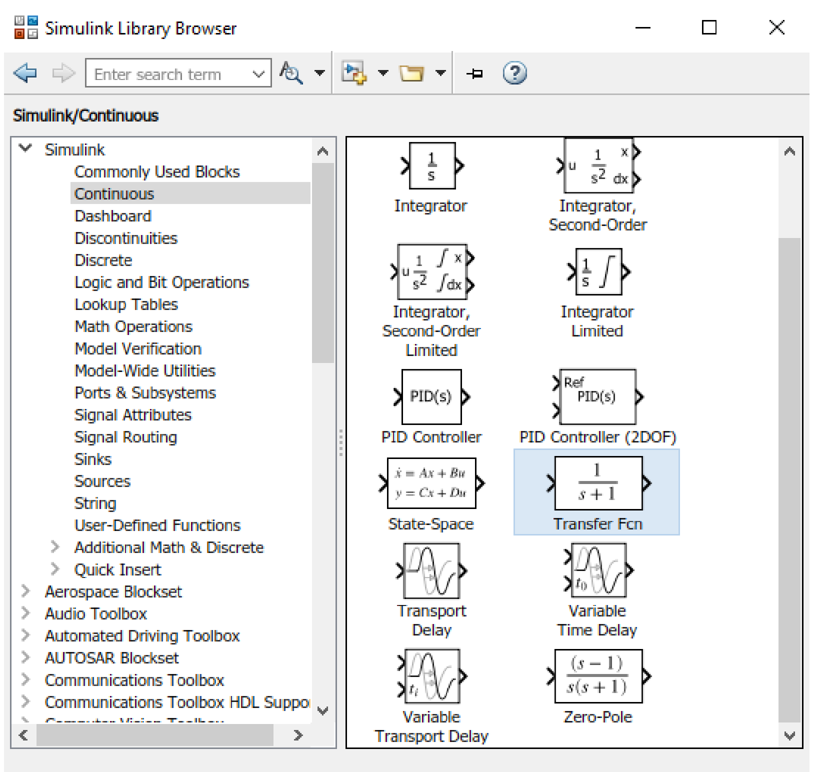
Ejemplo 1:

Diseña un sistema de control empleando un controlador P (a diferentes ganancias) usando también la siguiente función de transferencia:

$$H(s) = \frac{24}{s^2 + 2s + 1}$$

Primero iniciaremos abriendo Simulink dentro de Matlab (como ya se hizo en ejercicios pasados) y seleccionamos una hoja en blanco. Una vez hecho esto, se nos abrirá una ventana en donde nosotros elaboraremos el diagrama de nuestro sistema de control empleando diferentes elementos. Estos elementos los vamos a encontrar en las librerías de Simulink.

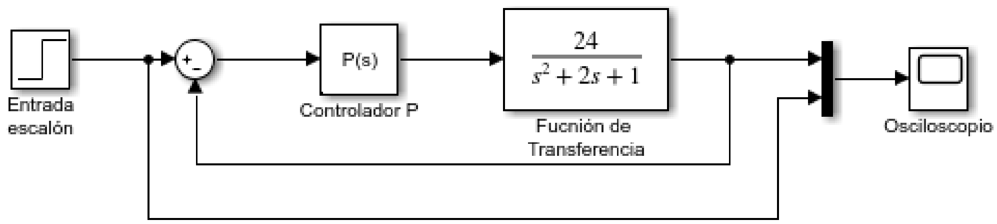
Los elementos que usaremos para estos ejercicios se encuentran con el siguiente nombre:



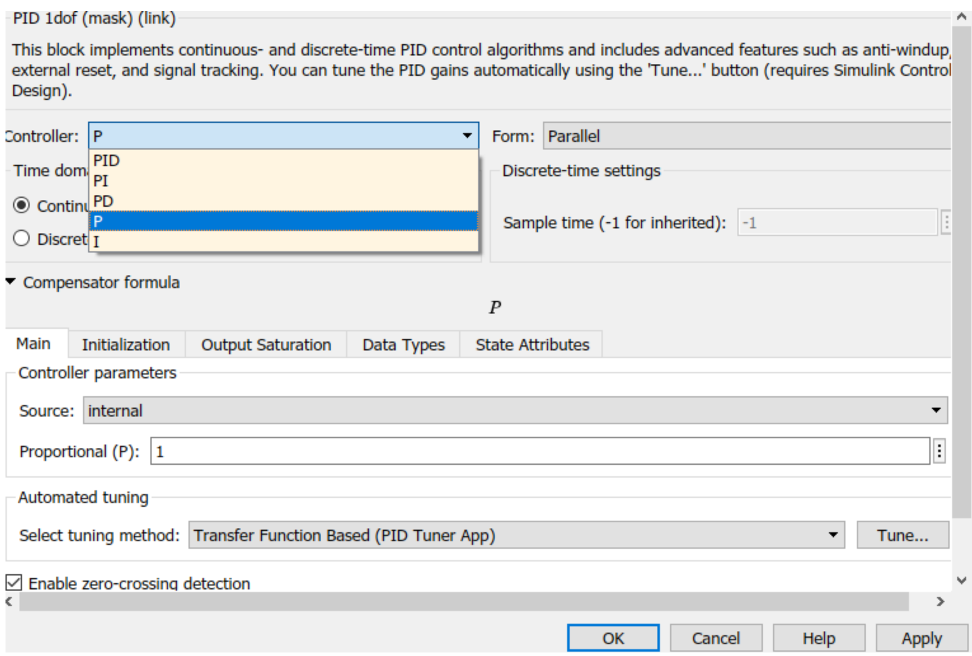
Entrada escalón: Sources → "Step"
Sumador: Math Operations → "Sum"

Controlador: Continuous → "PID Controller"
Función de transferencia: Continuous → "Transfer fnc"
Osciloscopio: Sinks → "Scope"
Conector: Signal Routing → "Mux"

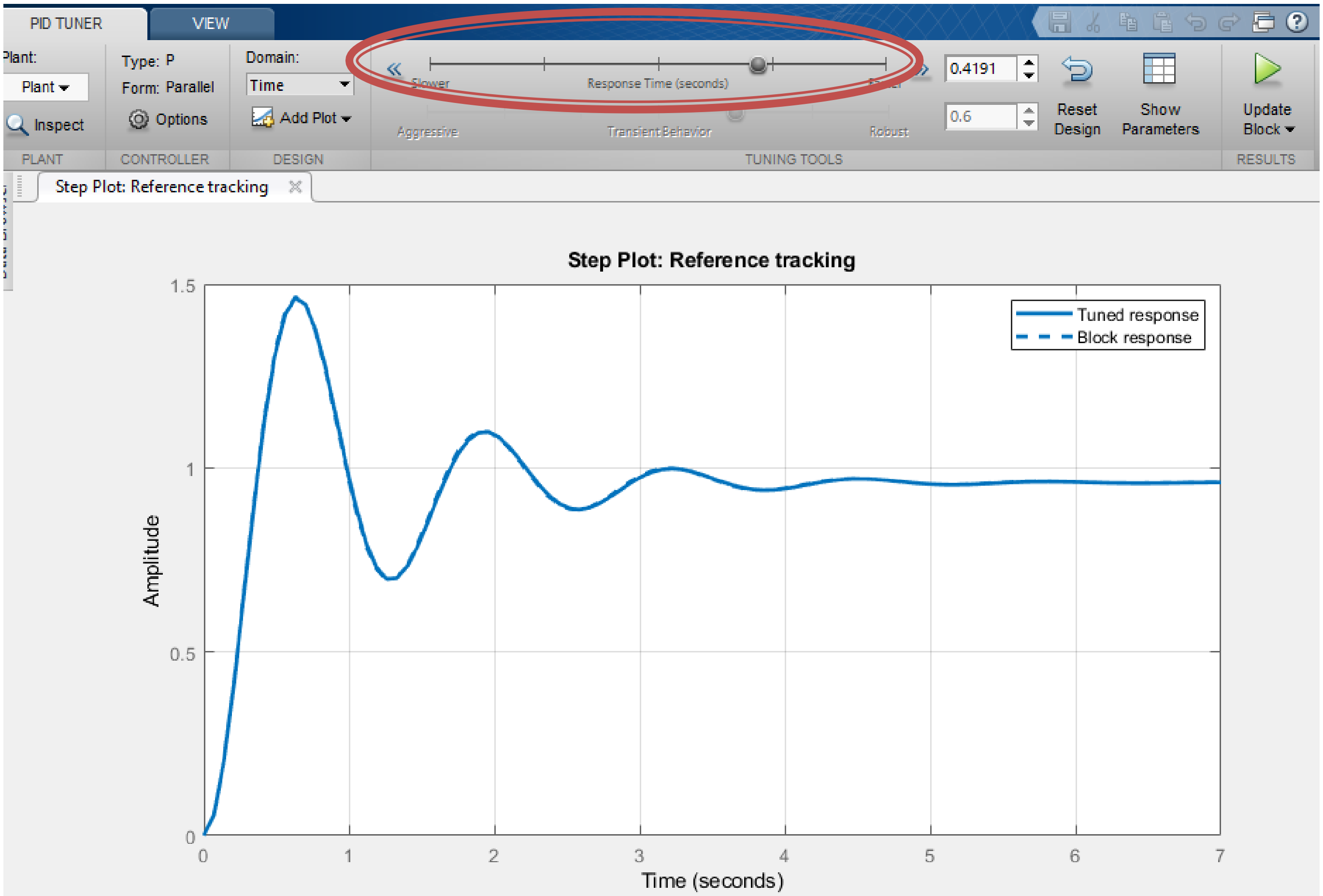
Luego procedemos a realizar nuestro diagrama:



Para éste problema se requiere usar un controlador P, entonces haremos doble click en el bloque "PID Controller" y seleccionamos "Controller P".



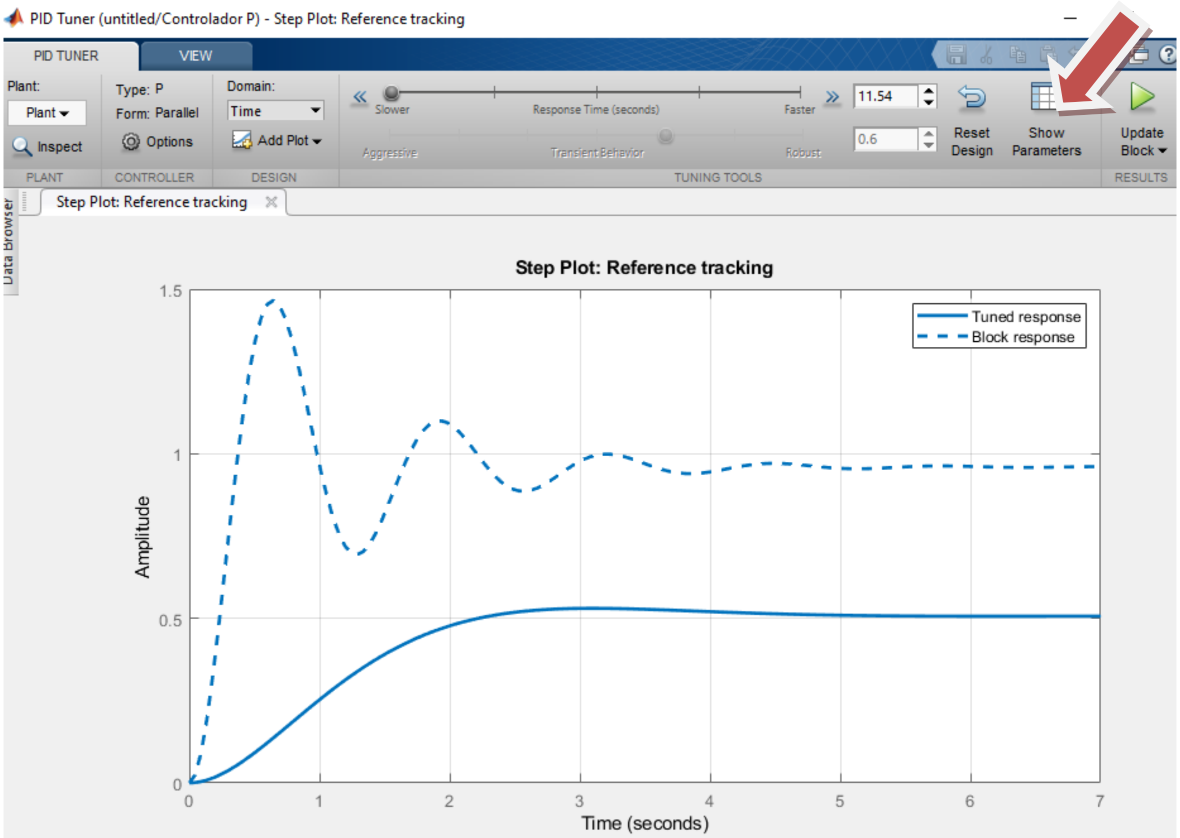
Posteriormente le daremos click en el botón de "Tune" el cuál nos va a permitir jugar con nuestra señal para saber qué valor va a tener el controlador "P" dependiendo de la señal que vayamos a requerir. Esto se hace moviendo de izquierda a derecha el cursor sobre la barra señalada en rojo en la siguiente imagen. También daremos click en "Show Parameters" y se nos abrirá una ventana donde nos dirá como es qué varía el valor de P al estar moviendo la barra.



Controller Parameters

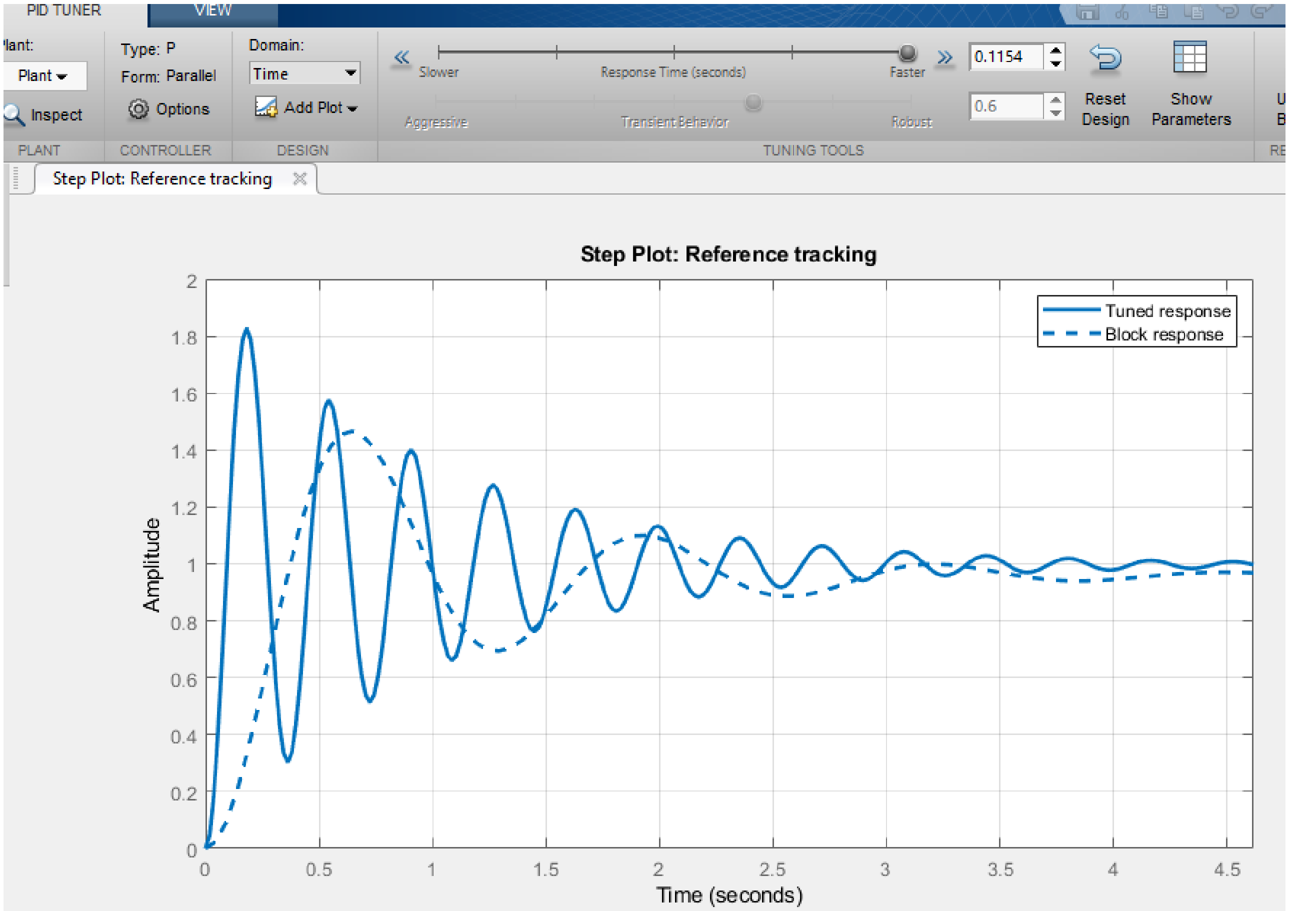
	Tuned	Block
P	0.99046	1
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a

Como al abrir la ventana del controlador, se preselecciona un valor de P=1, entonces la señal punteada es cuando P=1. Como se dijo anteriormente, el valor de P va a “arreglar” nuestra señal dependiendo de que sea lo que estemos buscando, aquí dejaremos dos ejemplos, uno con el valor mínimo y otro con el valor máximo, para ver cómo cambia la señal al pasar de un valor a otro.



Controller Parameters

	Tuned	Block
P	0.042917	1
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a



Controller Parameters

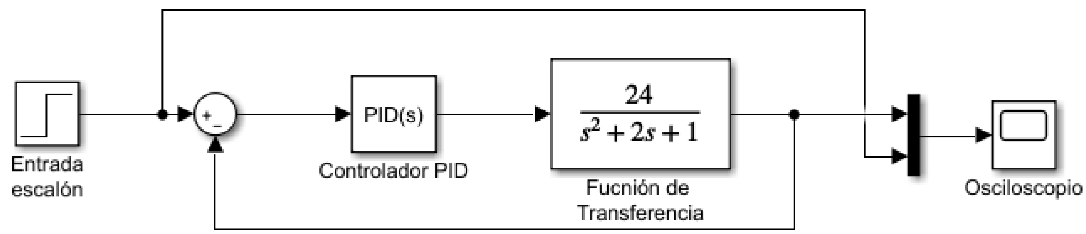
	Tuned	Block
P	12.5492	1
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a

Al ser una función de transferencia de segundo orden, pues obtendremo diferentes comportamientos de amortiguamiento al cambiar el valor de P.

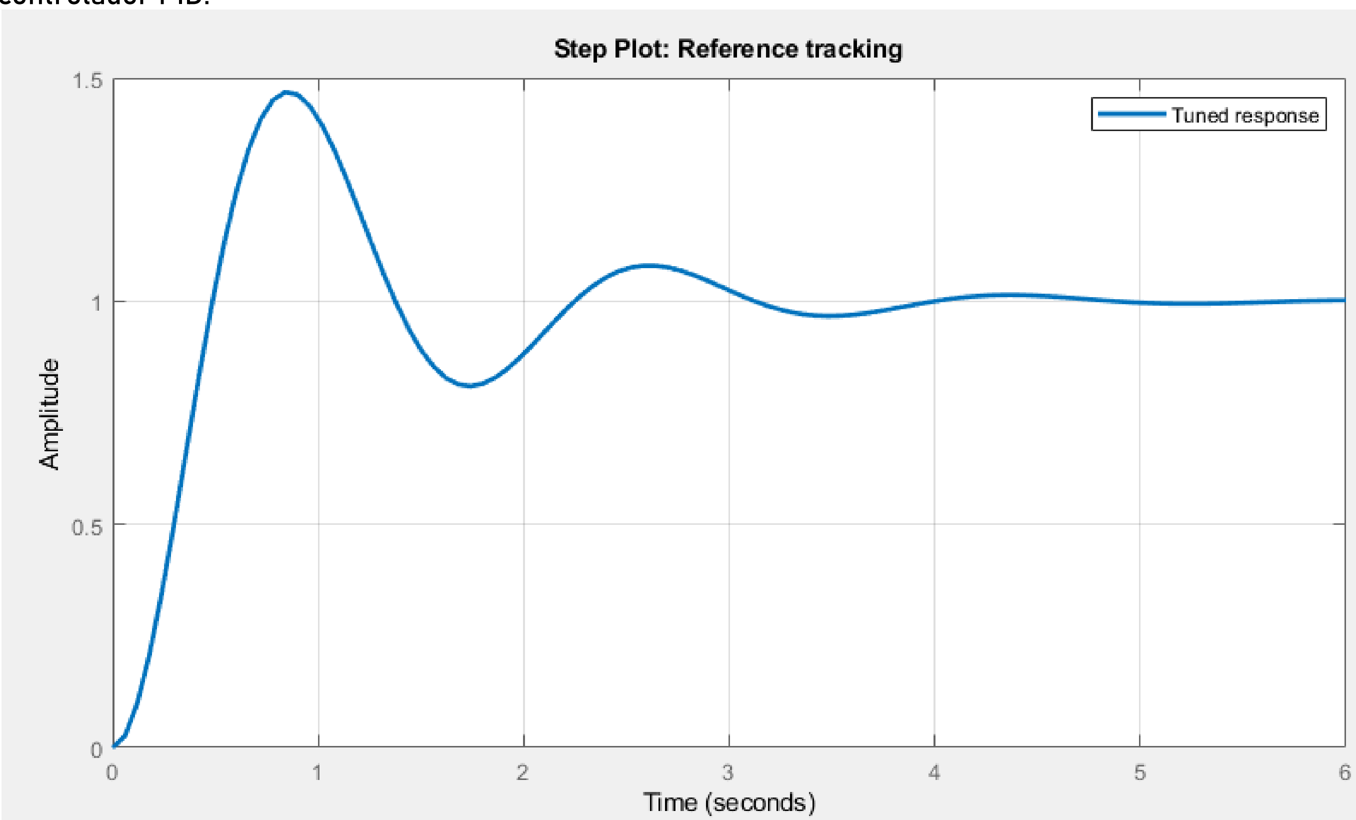
Ejemplo 2: Realiza un sistema de control empleando un controlador PID con la misma función de transferencia del ejercicio 1 y observa que cambios hay en las señales:

$$H(s) = \frac{24}{s^2 + 2s + 1}$$

El diagrama dentro de Simulink nos queda de la siguiente manera:



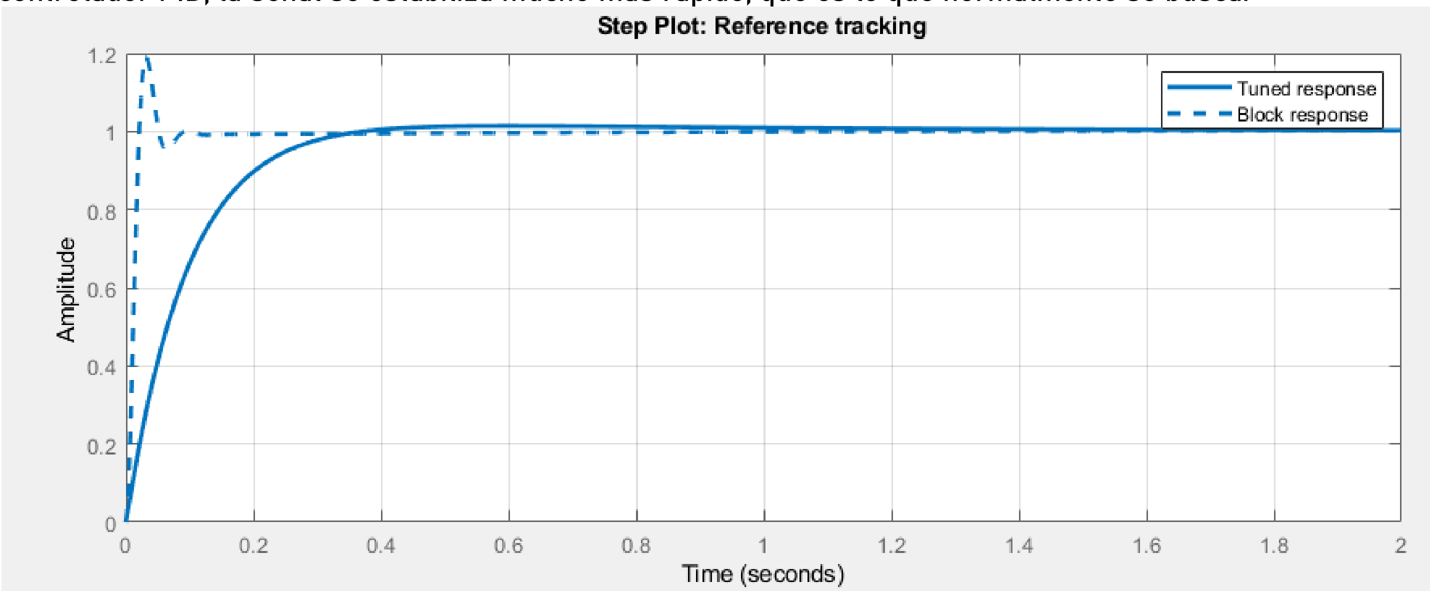
Una señal agresiva es cuando obtiene mucho movimiento ondulatorio al inicio y al final se estabiliza, en donde los valores del integrados, derivador y del proporcional son mínimos, es decir, ésta sería nuestra señal sin usar el controlador PID:



Controller Parameters

	Tuned
P	0.48028
I	0.40671
D	0.045118
N	3.5306

Y ésta sería nuestra señal al usar un controlador PID en óptimas condiciones. Como se puede observar, al usar el controlador PID, la señal se estabiliza mucho mas rápido, que es lo que normalmente se busca:



Controller Parameters

	Tuned	Block
P	1.0106	5
I	0.56994	5
D	0.44223	5
N	1218.5205	100

Entonces los controladores nos van a ayudar para distorsionar las señales que obtenemos tanto como deseemos. Si esto lo llevamos a problemas reales, El controlador PID permite regular la velocidad, temperatura, presión y flujo entre otras variables de un proceso en general. El controlador PID calcula la diferencia entre nuestra variable real contra la variable deseada.

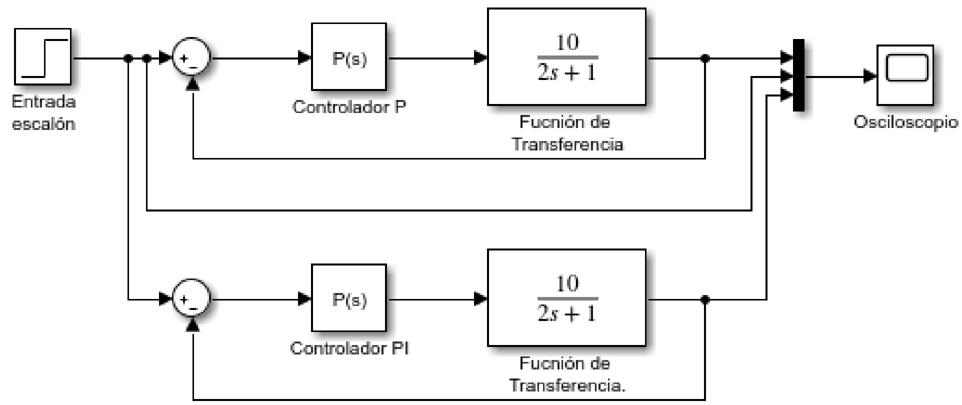
EL control P es una de las acciones de control empleadas en los lazos de regulación automática, es el el tipo de control que utilizan la mayoría de los controladores que regulan la velocidad de un automóvil.

El control PI es la estructura más usual del controlador, pues es la forma más simple de eliminar el error en régimen permanente. También se usa cuando el desfase que introduce el proceso es moderado. También se recomienda la acción PI cuando hay retardos en el proceso, ya que, es este tipo de procesos la acción derivativa no resulta apropiada en este tipo de sistemas. Un tercer caso en el que se debería prescindir de la acción derivativa es cuando el proceso está contaminado con niveles de ruido elevados.

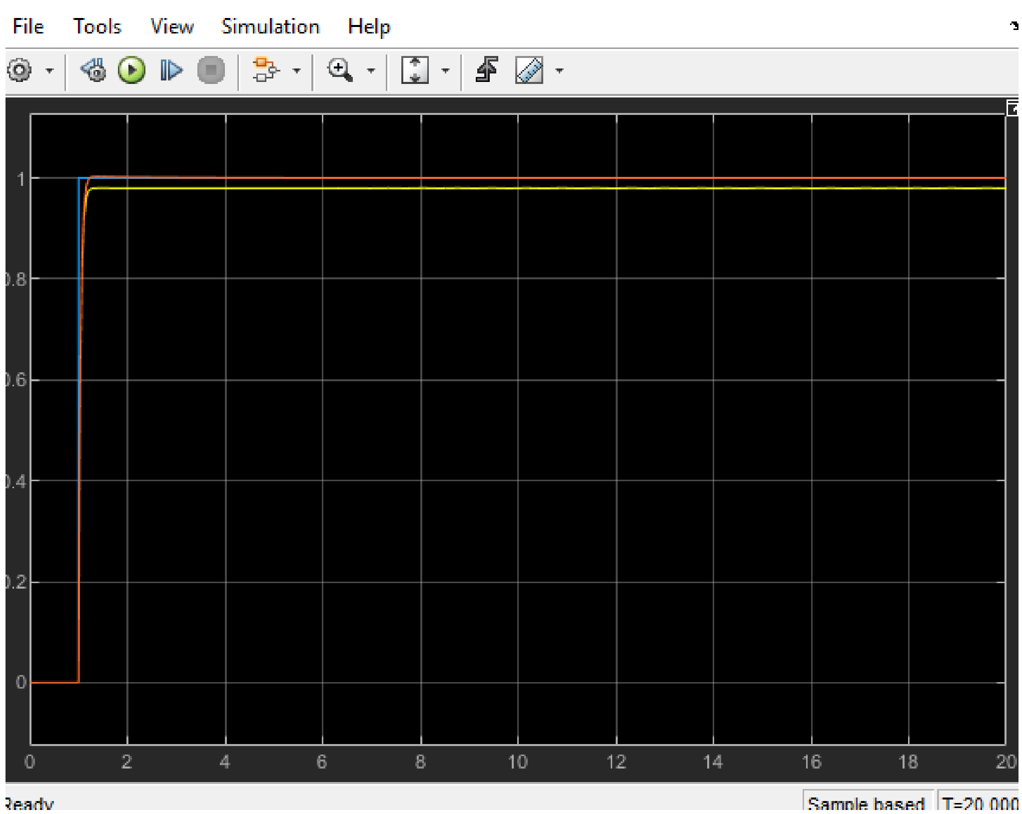
Ejercicio 3: Realiza una comparación entre un controlador P y un controlador PI usando una esntrada de escalón unitario y la siguiente función de transferencia de primer orden:

$$F(s) = \frac{10}{2s + 1}$$

El diagrama en Simulink nos quedaría de la siguiente manera:



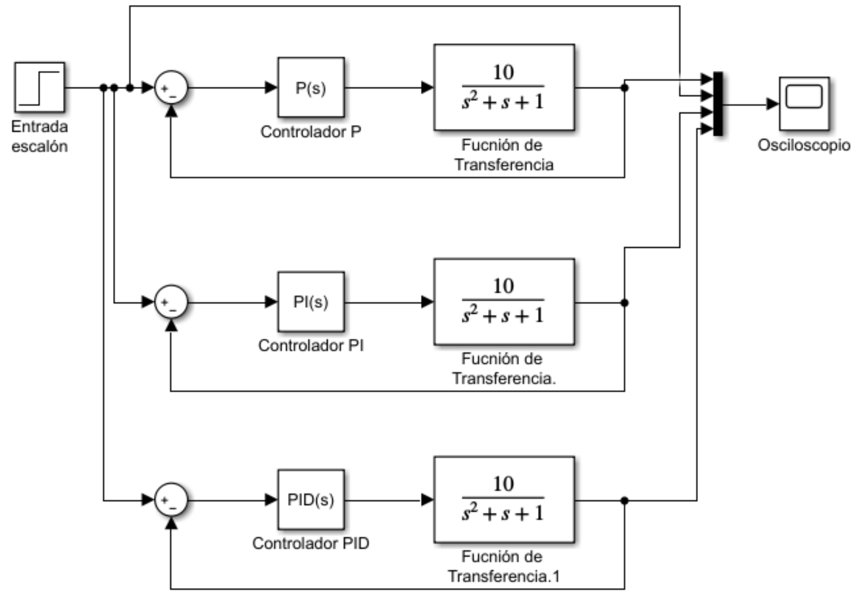
Y la respuesta de ambos controladores es muy parecida, ambos se estabilizan rápidamente pero la señal del controlador P se mantiene con un offset, es decir que no alcanza a llegar al valor del escalón unitario y nunca lo hará, mientras que la señal del controlador PI no tiene ese offset y es muy parecida a la señal de escalón unitario. El error es mínimo en el controlador PI.

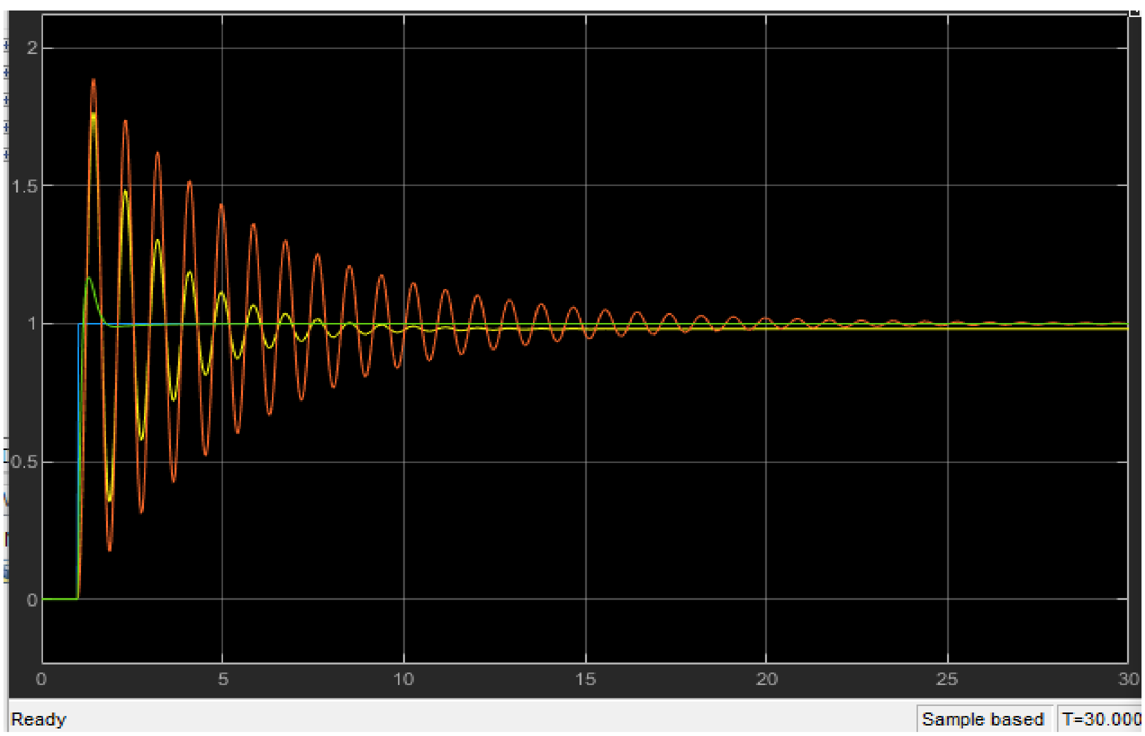


Los colores de las señales corresponden a:

Señal azul: señal de entrada
Señal amarilla: Controlador P (P=5)
Señal naranja: Controlador PI (P=5, I=3)

Ejercicio 4: Realiza una comparación entre los controladores P, Pi y PID con una función de transferencia de segundo orden y obtén las señales correspondientes a cada controlador.

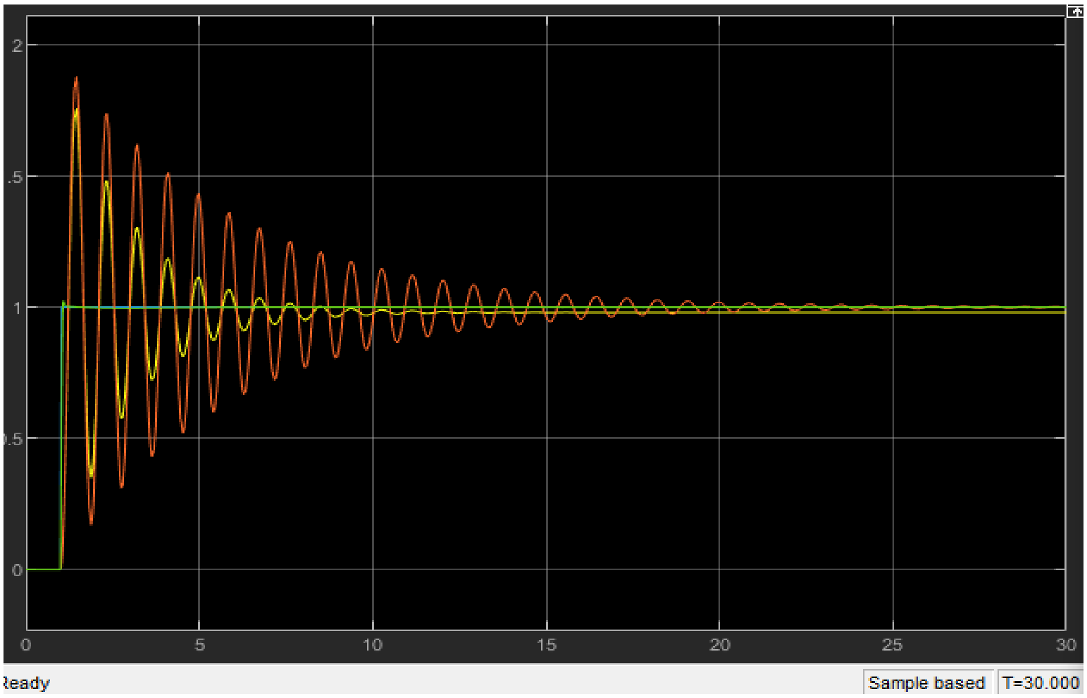




Señal azul: señal de entrada
Señal amarilla: Controlador P (P=5)
Señal naranja: Controlador PI (P=5, I=3)
Señal verde: Controlador PID (P=5, I=3, D=1)

Podemos ver que la señal del controlador PID es la que tiene la menor distorsión y la que alcanza la estabilización más rápido con respecto a la señal escalón unitario, pero aun así hay un "pico" que puede arreglarse y es lo que intentaremos en una segunda gráfica.

Para esto incrementamos el valor del controlador PID en el aspecto derivativo, con esperanzas de mejorar un poco la señal y de tener el mínimo ruido posible, al mismo tiempo de alcanzar una estabilización más rápida:



Señal azul: señal de entrada
Señal amarilla: Controlador P (P=5)
Señal naranja: Controlador PI (P=5, I=3)
Señal verde: Controlador PID (P=5, I=3, D=4)

Tema 9: Lugar Geométrico de las Raíces (LGR)

El lugar geométrico de las raíces es un poderoso método de análisis y diseño para la estabilidad y respuesta transitoria de un sistema de control. Consiste en una representación gráfica de los polos de la función de transferencia a lazo cerrado a medida que varía uno o varios parámetros del sistema.

Los sistemas de control realimentados son difíciles de comprender desde el punto de vista cualitativo, por lo que dicha comprensión depende en gran medida de las matemáticas. El lugar geométrico de las raíces es la técnica gráfica que nos da esa descripción cualitativa sobre el rendimiento del sistema de control que estamos diseñando. Además, también sirve como una poderosa herramienta cuantitativa que produce más información que los métodos ya discutidos, debido a que no sólo puede servir para encontrar la solución de sistemas de primero y segundo orden, sino que también es de utilidad para solucionar sistemas de orden mayor a dos.

Mediante el lugar geométrico de las raíces, se puede observar el comportamiento del sistema (relativo a la respuesta transitoria y la estabilidad) a medida que se varían varios parámetros a la vez, como el sobrepaso, el tiempo de asentamiento y el tiempo pico. Luego, esta información cualitativa puede ser verificada mediante el análisis cuantitativo.

Hay varias reglas que deben de cumplirse el en LGR:

1. Debe ser simétrico con el eje real R.
2. Todos los polos (P) tienen a un cero (Z), (ya sea un cero finito o uno infinito).
3. El número de ceros siempre debe ser menor o igual al número de polos.

Para calcular las asíntotas se hará uso de las siguientes formulas:

Lugar Geométrico de las Raíces:

$$\boldsymbol{\theta_n} = \frac{(\boldsymbol{2k+1})\boldsymbol{180}^{\circ}}{\boldsymbol{\#P - \#Z}}$$

Para: $\boldsymbol{k = 0, 1 \ldots\ldots [(\#P - \#Z) - 1]}$

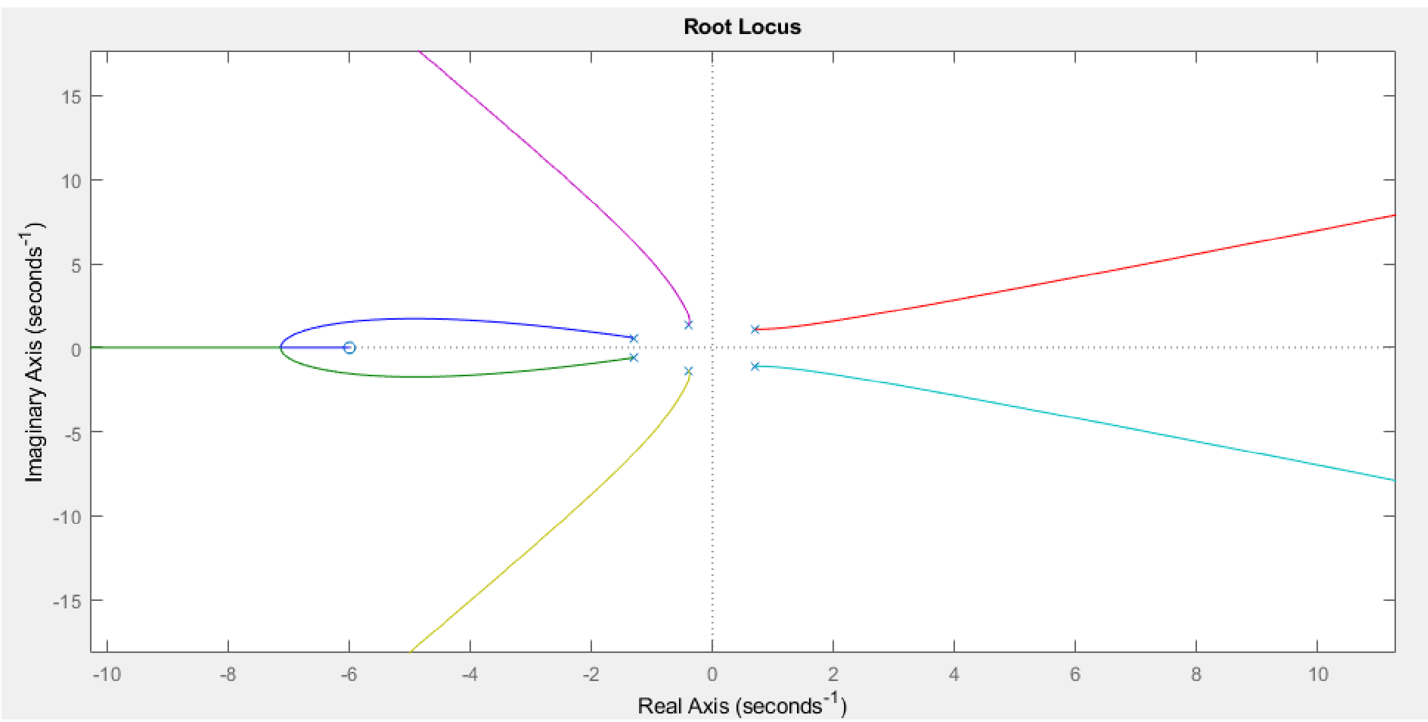
$$\boldsymbol{\gamma} = \frac{\boldsymbol{\Sigma polos - \Sigma zeros}}{\boldsymbol{\#P - \#Z}}$$

Para poder ver esto gráficamente mediante Matlab, es necesario ingresar la función de transferencia con el comando tf como ya lo hemos hecho antes y posteriormente utilizar el comando rlocus() el cual nos arrojará la gráfica del lugar geométrico de las raíces de la función de transferencia.

$$F(s) = \frac{s+6}{s^6 + 2s^5 + 3s^4 + 4s^3 + 5s^2 + 6s + 7}$$

```
f=tf([1 6],[1 2 3 4 5 6 7])


f =

                        s + 6
  ---------------------------------------------
  s^6 + 2 s^5 + 3 s^4 + 4 s^3 + 5 s^2 + 6 s + 7

Continuous-time transfer function.
```

```
>> polos=roots(f.den{1})

polos =

  -1.3079 + 0.5933i
  -1.3079 - 0.5933i
   0.7104 + 1.1068i
   0.7104 - 1.1068i
  -0.4025 + 1.3417i
  -0.4025 - 1.3417i

>> zeros=roots(f.num{1})

zeros =

    -6

>> rlocus(f)
```



$$\theta_n = \frac{(2k+1)180°}{\#P - \#Z}$$

Para: $k = 0, 1 \ldots\ldots [(\#P - \#Z) - 1]$

$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z}$$

$$k = [(\#P - \#Z) - 1] = (6 - 1) - 1 = 4$$

$$\boldsymbol{k = [0, 1, 2, 3]}$$

$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z} = \frac{(-2) - (-6)}{6-1} = \frac{4}{5} = 0.8$$
$$\boldsymbol{\gamma = 0.8}$$

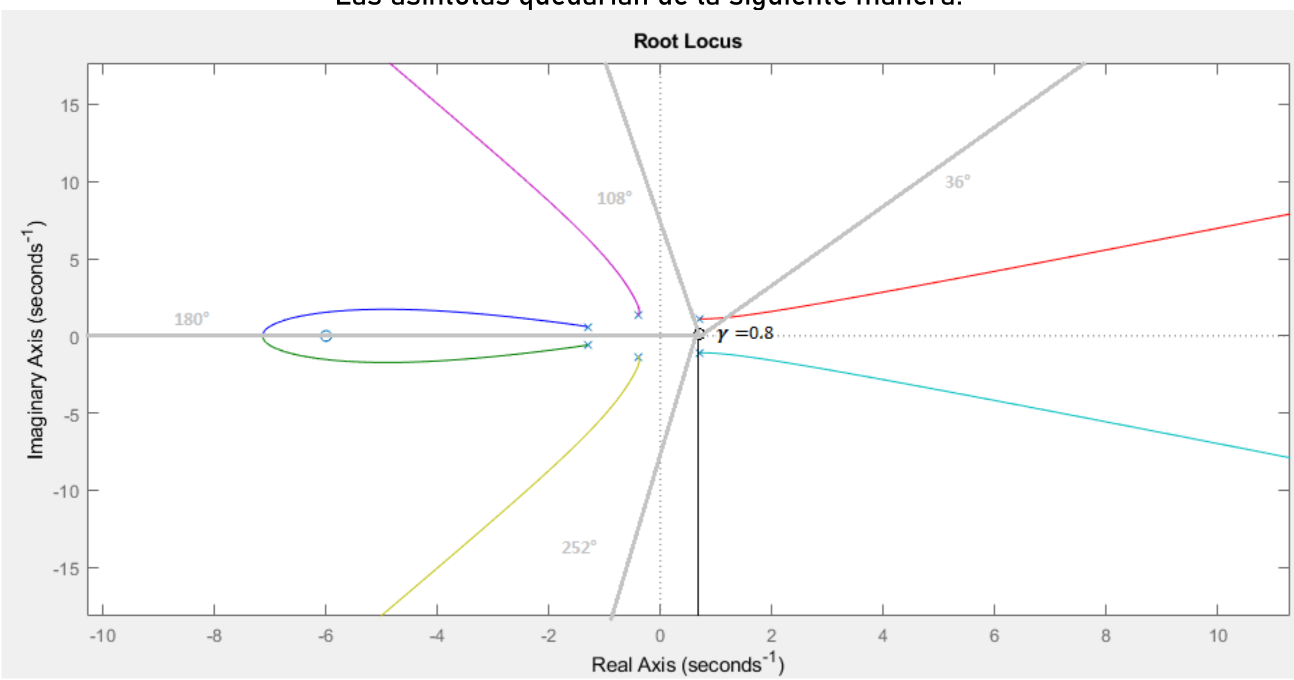
$$\theta_1 = \frac{(2k+1)180^\circ}{\#P - \#Z} = \frac{(2(0)+1)180^\circ}{5} = \frac{180^\circ}{5} = 36^\circ$$
$$\boldsymbol{\theta_1 = 36^\circ}$$

$$\theta_1 = \frac{(2k+1)180^\circ}{\#P - \#Z} = \frac{(2(1)+1)180^\circ}{5} = \frac{540^\circ}{5} = 108^\circ$$
$$\boldsymbol{\theta_1 = 108^\circ}$$

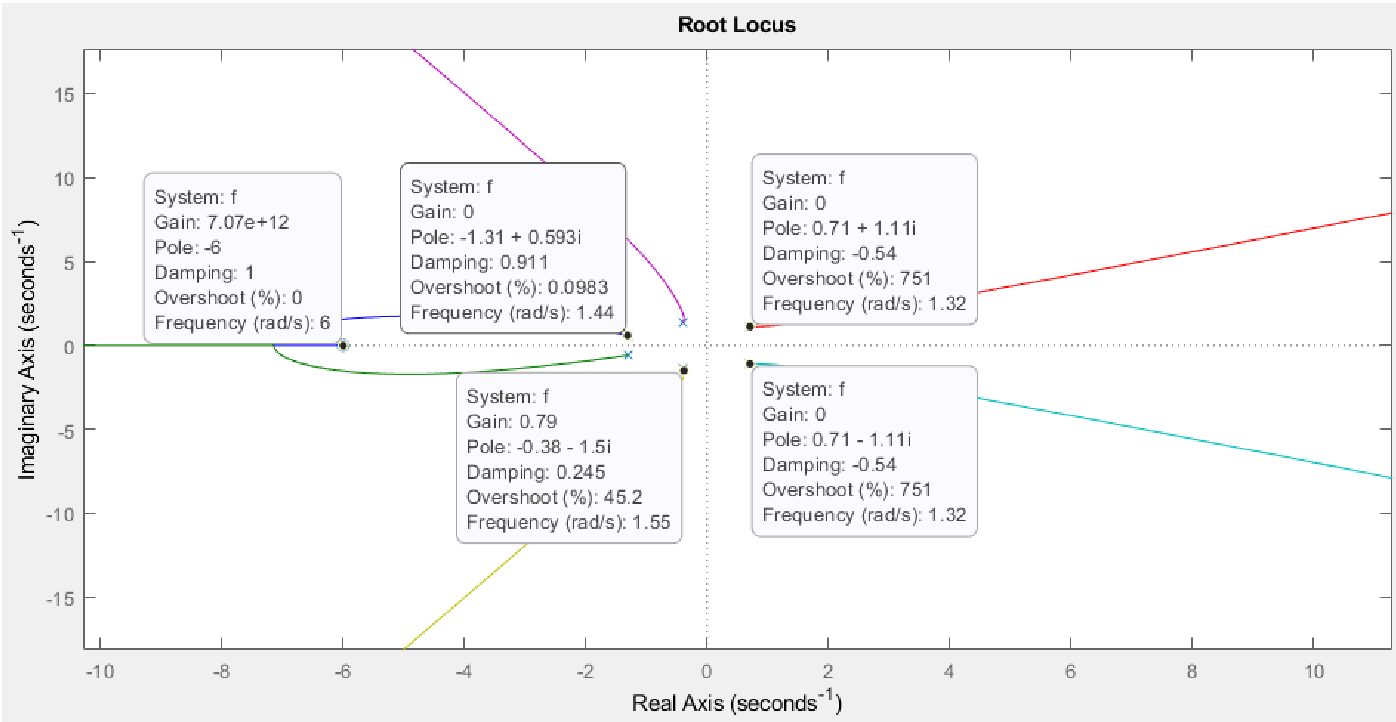
$$\theta_1 = \frac{(2k+1)180^\circ}{\#P - \#Z} = \frac{(2(2)+1)180^\circ}{5} = \frac{900^\circ}{5} = 180^\circ$$
$$\boldsymbol{\theta_1 = 180^\circ}$$

$$\theta_1 = \frac{(2k+1)180^\circ}{\#P - \#Z} = \frac{(2(3)+1)180^\circ}{5} = \frac{1260^\circ}{5} = 252^\circ$$
$$\boldsymbol{\theta_1 = 252^\circ}$$

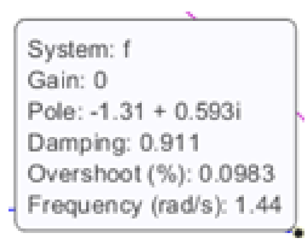
Las asíntotas quedarían de la siguiente manera:



Podemos observar que al mover el cursor sobre la gráfica nos arroja diferentes datos, los cuales se explican a continuación:



Pongamos el siguiente ejemplo:



System

Se refiere al nombre del sistema de la función de transferencia ingresada. En este caso, al sistema lo llamamos "f" dentro de Matlab.

Gain

Hace referencia al valor de la ganancia dependiendo si es un polo o cero.

Pole

Son las coordenadas de cada polo o cero. En este caso se está señalando uno de los polos: -1.31 + 0.593i.

Damping

Es el factor de amortiguamiento en cada polo o cero de la función de transferencia.

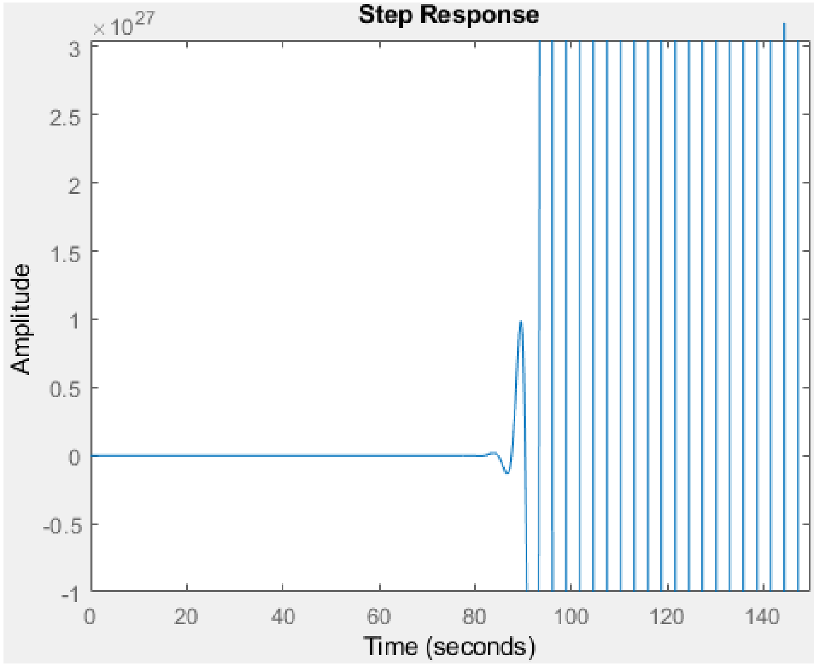
Overshoot

Se refiere a la sobre oscilación que experimenta el sistema.

Frecuency

Frecuencia natural del sistema que corresponde a la distancia del origen al punto señalado en rad/seg.

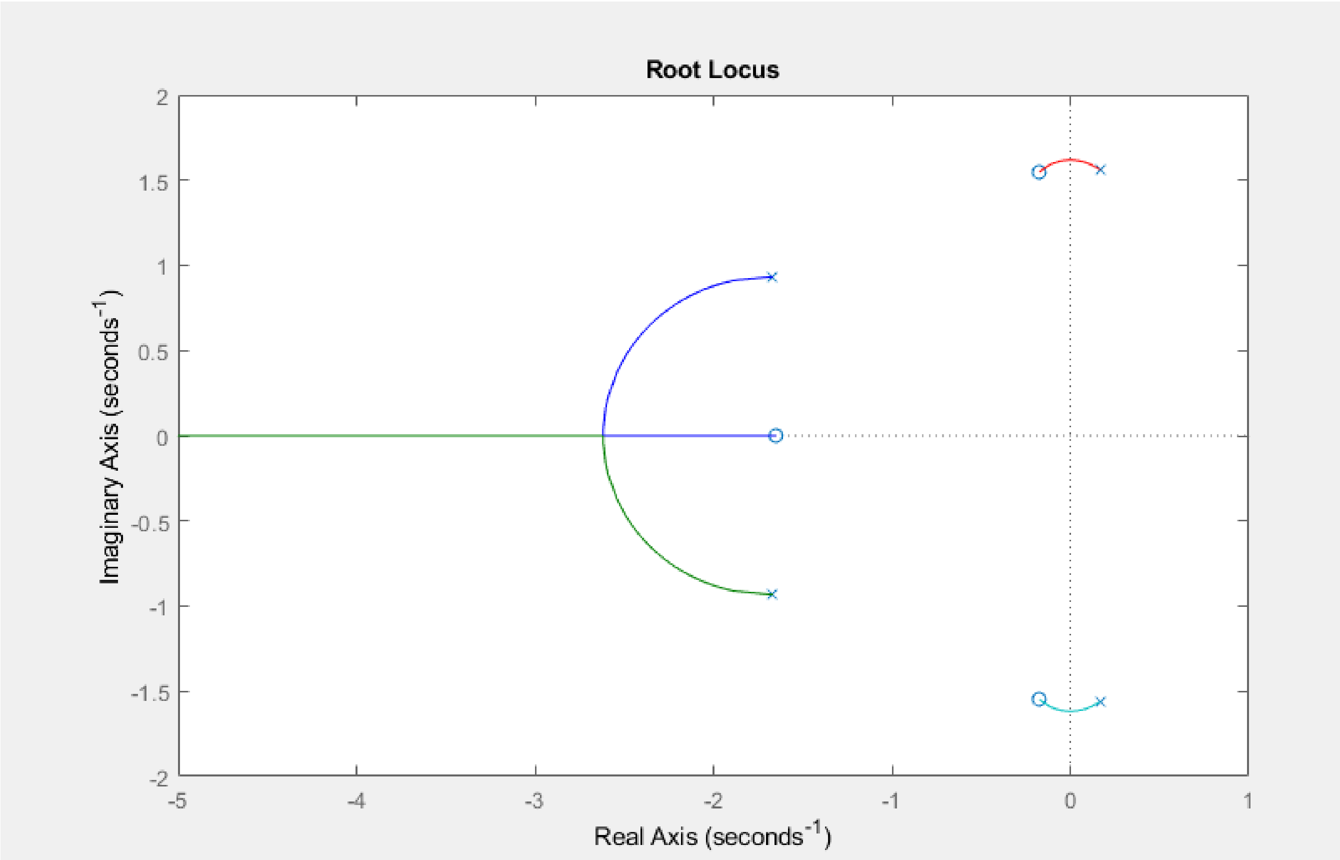
Se puede concluir que, como hay polos y ceros en la parte derecha del eje, entonces se trata de un sistema inestable, pero para hacerlo un poco más visible se aplicará una entrada escalón al sistema. En la siguiente gráfica se puede comprobar que el sistema si es inestable:



Veamos el siguiente ejemplo:

$$G(s) = \frac{s^3 + 2s^2 + 3s + 4}{s^4 + 3s^3 + 5s^2 + 7s + 9}$$

```
>> g=tf([1 2 3 4],[1 3 5 7 9])

g =

      s^3 + 2 s^2 + 3 s + 4
  -----------------------------
  s^4 + 3 s^3 + 5 s^2 + 7 s + 9

Continuous-time transfer function.

>> polos=roots(g.den{1})

polos =

  -1.6673 + 0.9330i
  -1.6673 - 0.9330i
   0.1673 + 1.5613i
   0.1673 - 1.5613i

>> zeros=roots(g.num{1})

zeros =

  -1.6506 + 0.0000i
  -0.1747 + 1.5469i
  -0.1747 - 1.5469i

>> rlocus(g)
```



Los polos están representados por cruces y los ceros están representados por círculos.

$$\boldsymbol{\theta_n = \frac{(2k+1)180^{\circ}}{\#P - \#Z}}$$

Para: $\boldsymbol{k = 0, 1 \ldots\ldots [(\#P - \#Z) - 1]}$

$$\boldsymbol{\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z}}$$

Para: $k = 0{,}1 \ldots\ldots [(\#P - \#Z) - 1]$

$$k = [(\#P - \#Z) - 1] = (4-3) - 1 = 0$$

$$\boldsymbol{k = [0]}$$

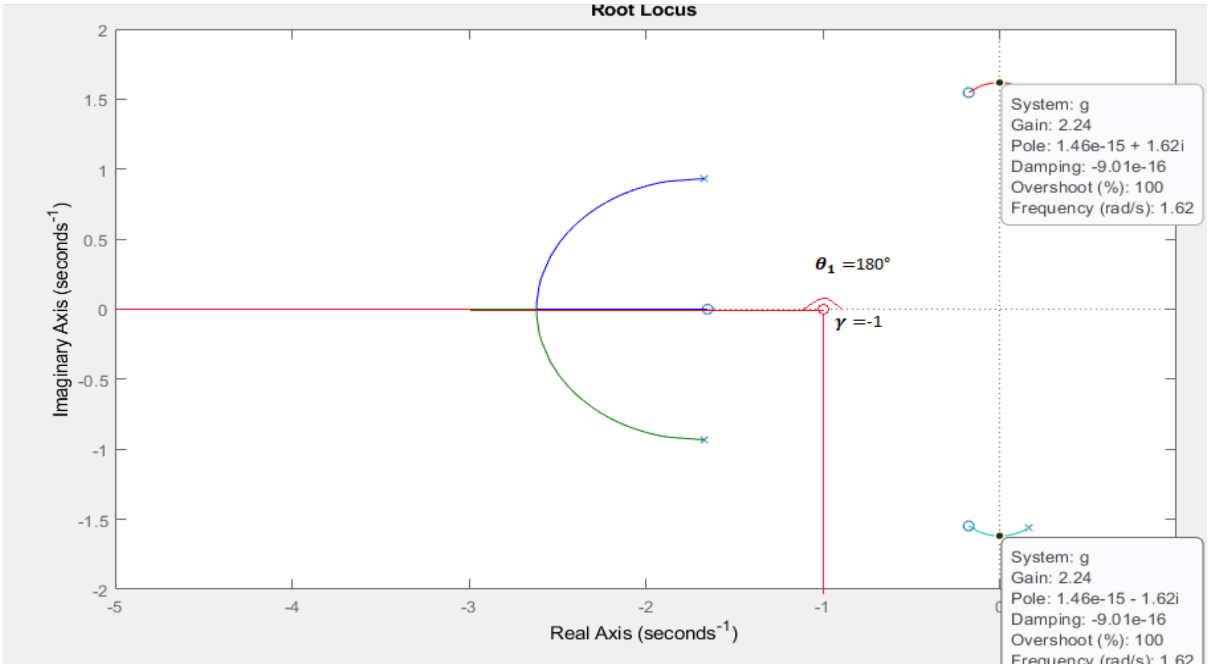
$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z} = \frac{(-3) - (-2)}{4-3} = \frac{-1}{1} = -1$$

$$\boldsymbol{\gamma = -1}$$

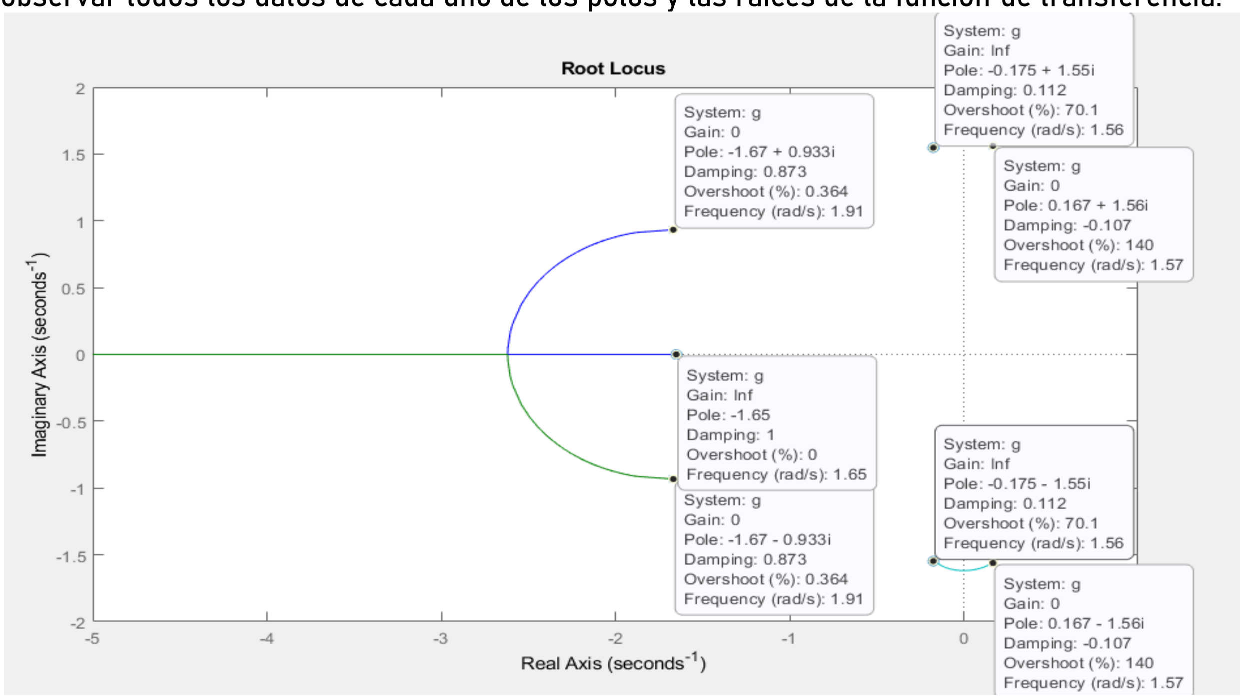
$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(0)+1)180^{\circ}}{4-3} = \frac{180^{\circ}}{1} = 180^{\circ}$$

$$\boldsymbol{\theta_1 = 180^{\circ}}$$

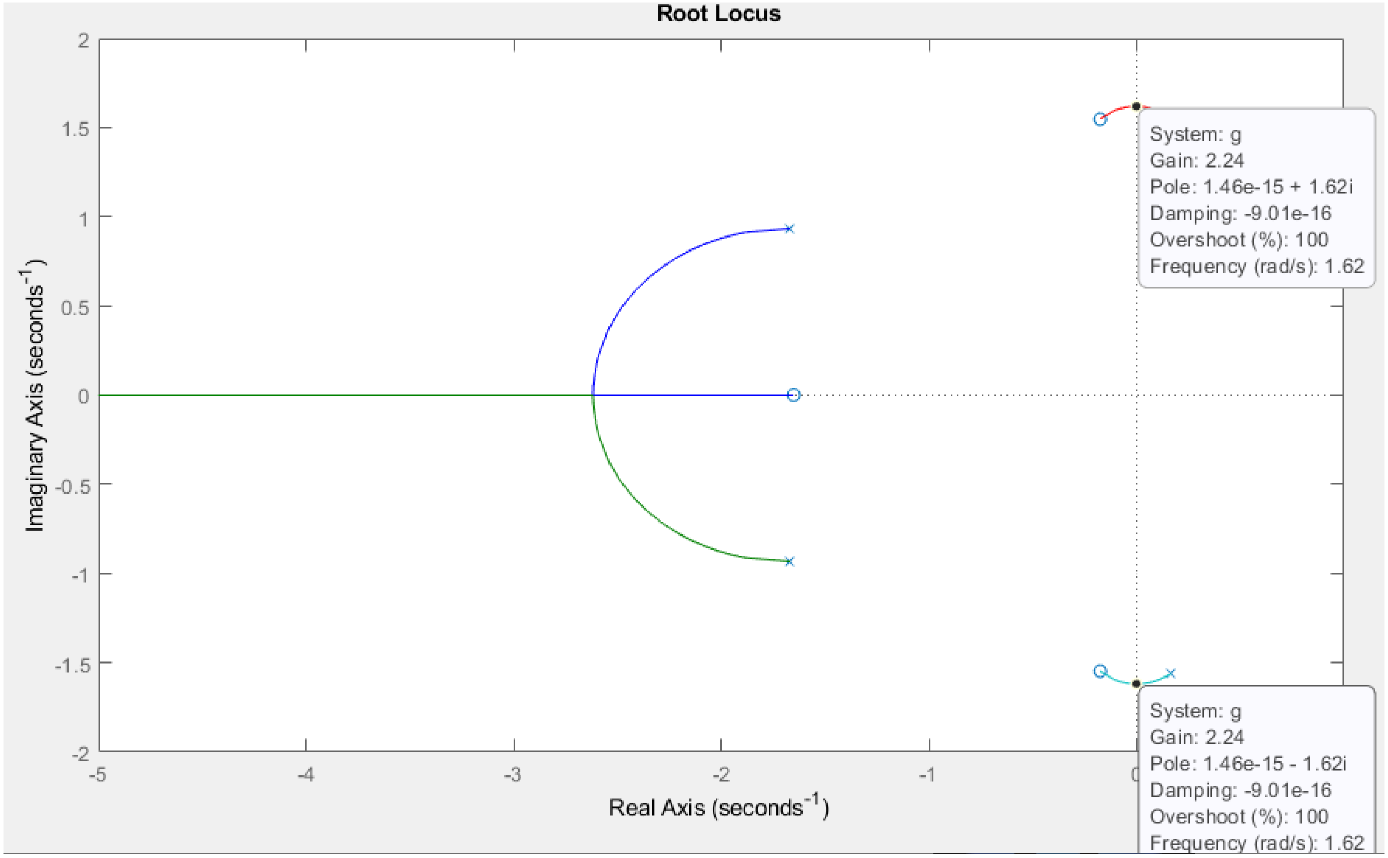
Estos datos nos indican el punto de partida y dirección de nuestras asíntotas, que en este caso, solamente será una. Nos quedaría de la siguiente manera:



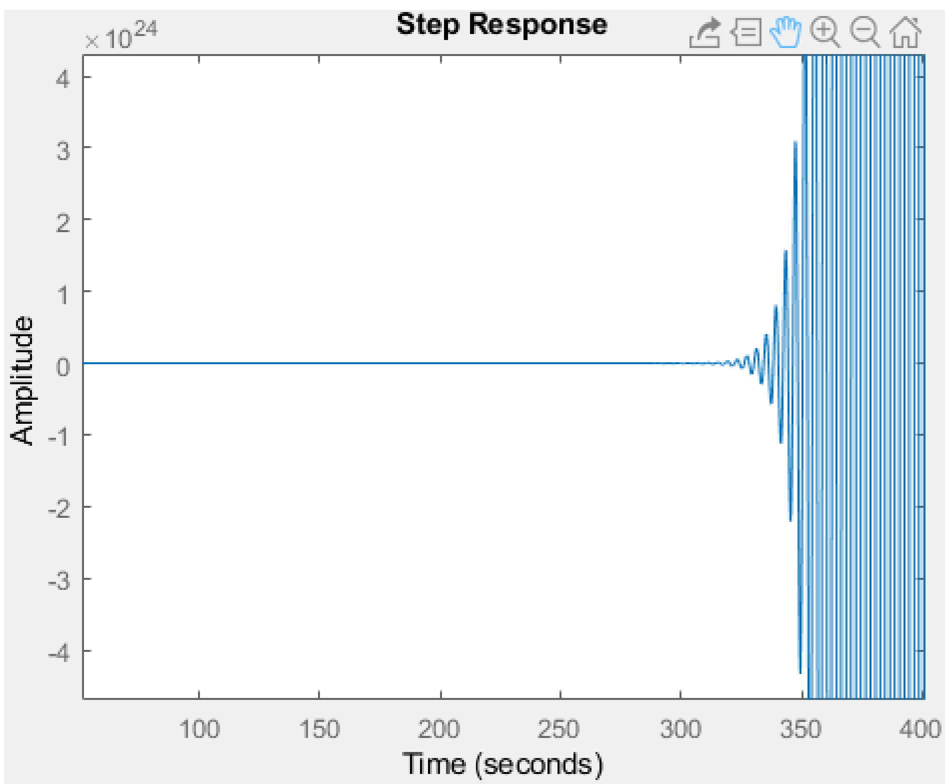
El lugar geométrico de la raíz sería el siguiente:
Podemos observar todos los datos de cada uno de los polos y las raíces de la función de transferencia:



El punto en donde deja de ser un sistema estable es en donde cruza de la parte imaginaria a la parte real, es decir, en estos puntos:



En la siguiente gráfica se puede comprobar que el sistema si es inestable:



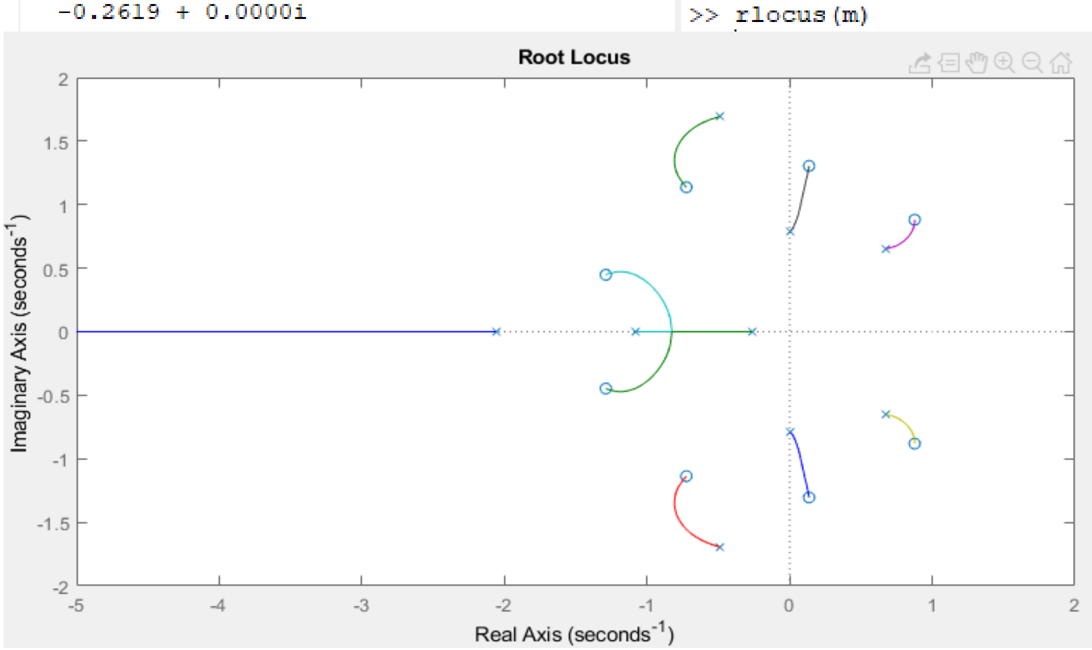
El siguiente ejemplo es:

$$G(s) = \frac{s^8 + 2s^7 + 3s^6 + 4s^5 + 5s^4 + 6s^3 + 7s^2 + 8s + 9}{s^9 + 3s^8 + 5s^7 + 7s^6 + 2s^5 + 4s^4 + 6s^3 + 2s^2 + 4s + 1}$$

```
>> rlocus(c)
>> m=tf([1 2 3 4 5 6 7 8 9],[1 3 5 7 2 4 6 2 4 1])

m =

     s^8 + 2 s^7 + 3 s^6 + 4 s^5 + 5 s^4 + 6 s^3 + 7 s^2 + 8 s + 9
  ---------------------------------------------------------------------
  s^9 + 3 s^8 + 5 s^7 + 7 s^6 + 2 s^5 + 4 s^4 + 6 s^3 + 2 s^2 + 4 s + 1

Continuous-time transfer function.
```

```
>> polos=roots(m.den{1})

polos =

  -2.0581 + 0.0000i
  -0.4850 + 1.6948i
  -0.4850 - 1.6948i
  -1.0769 + 0.0000i
   0.6773 + 0.6553i
   0.6773 - 0.6553i
   0.0061 + 0.7900i
   0.0061 - 0.7900i
  -0.2619 + 0.0000i
```

```
>> zeros=roots(m.num{1})

zeros =

  -1.2888 + 0.4477i
  -1.2888 - 0.4477i
  -0.7244 + 1.1370i
  -0.7244 - 1.1370i
   0.1364 + 1.3050i
   0.1364 - 1.3050i
   0.8767 + 0.8814i
   0.8767 - 0.8814i

>> rlocus(m)
```



$$\theta_n = \frac{(2k+1)180^\circ}{\#P - \#Z}$$

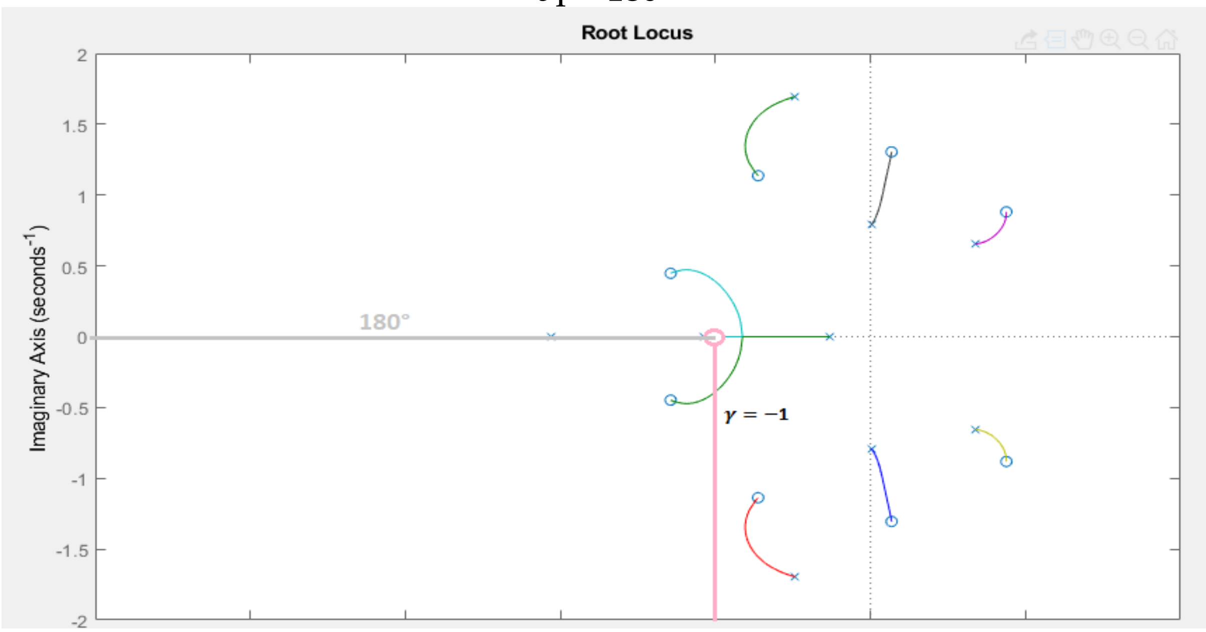
Para: $k = 0, 1 \ldots\ldots [(\#P - \#Z) - 1]$

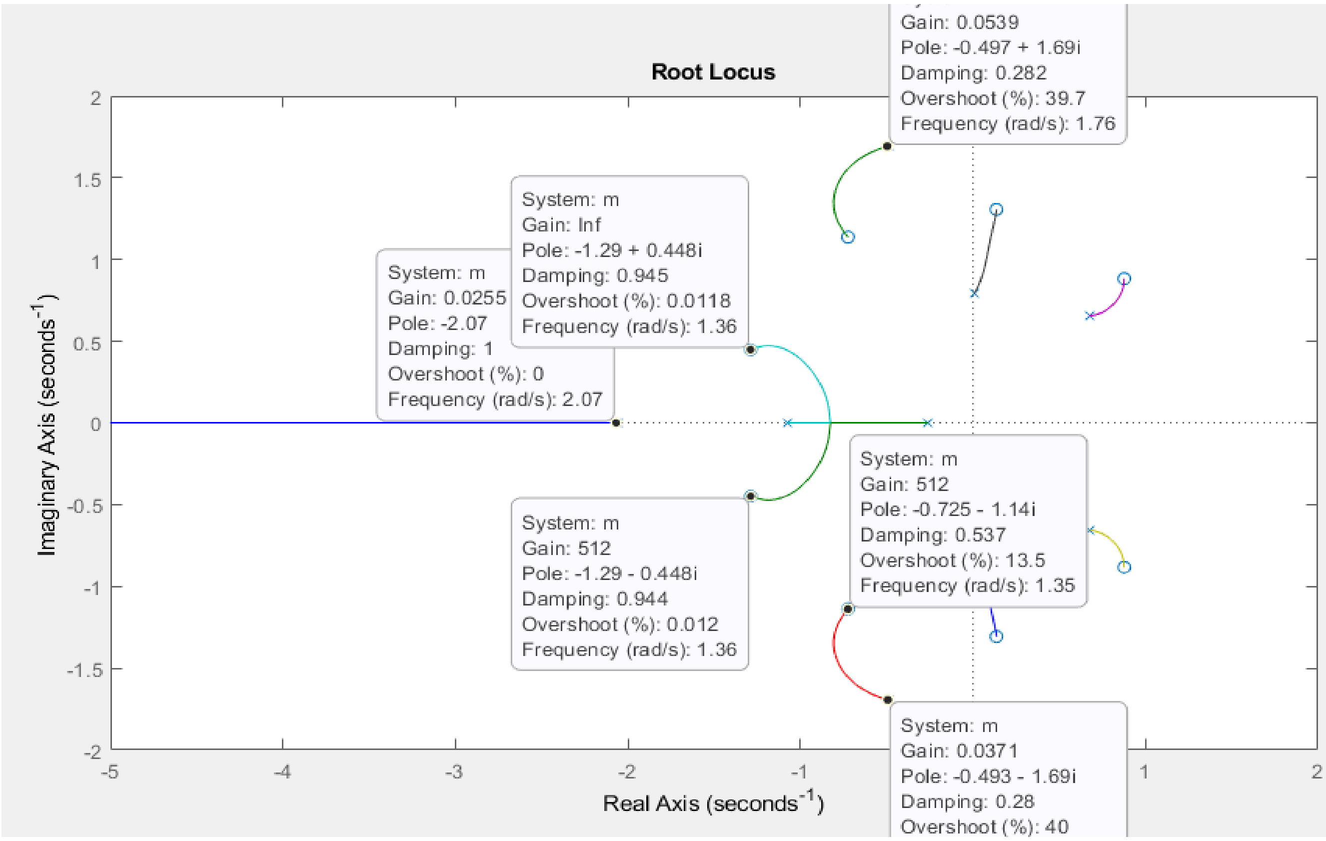
$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z}$$

$$k = [(\#P - \#Z) - 1] = (9-8) - 1 = 0$$
$$\boldsymbol{k = [0]}$$

$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z} = \frac{(-3) - (-2)}{9-8} = \frac{-1}{1} = -1$$
$$\boldsymbol{\gamma = -1}$$

$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(0)+1)180^{\circ}}{9-8} = \frac{180^{\circ}}{1} = 180^{\circ}$$
$$\boldsymbol{\theta_1 = 180^{\circ}}$$





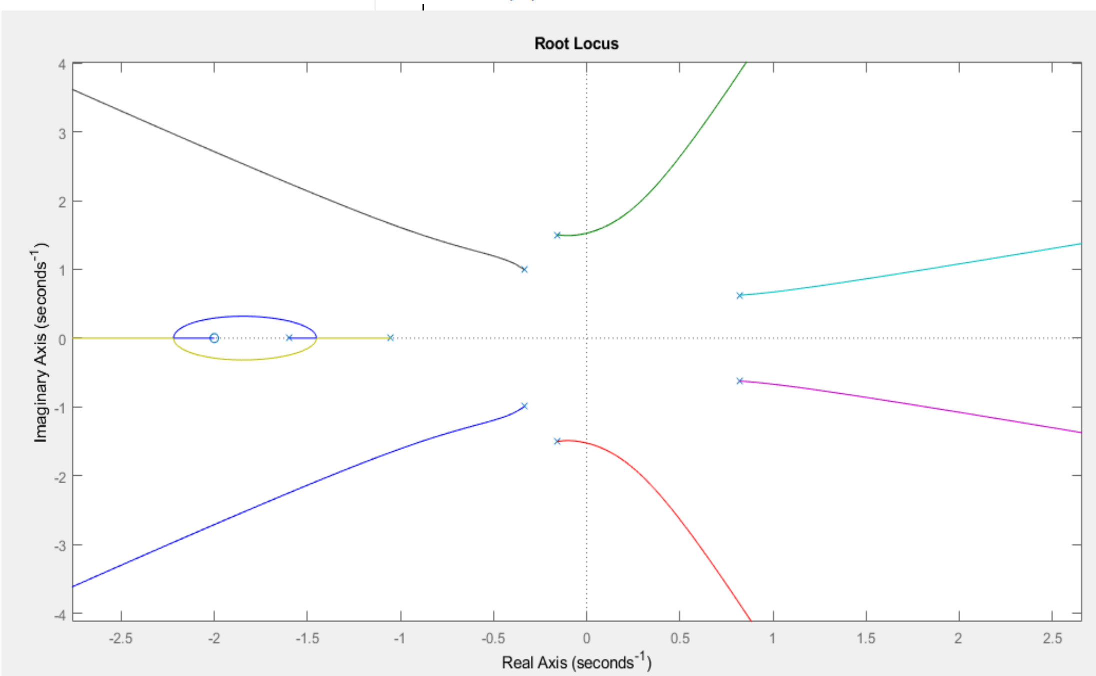
En la siguiente gráfica se puede comprobar que el sistema si es inestable:

Siguiente ejemplo:

$$S(s) = \frac{s+2}{2s^8 + 4s^7 + 6s^6 + 8s^5 + s^4 + 3s^3 + 5s^2 + 7s + 9}$$

```
>> s=tf([1 2],[2 4 6 8 1 3 5 7 9])

s =

                              s + 2
  -------------------------------------------------------------
  2 s^8 + 4 s^7 + 6 s^6 + 8 s^5 + s^4 + 3 s^3 + 5 s^2 + 7 s + 9

Continuous-time transfer function.
```

```
>> polos=roots(s.den{1})

polos =

  -1.5978 + 0.0000i
  -0.1589 + 1.5034i
  -0.1589 - 1.5034i
   0.8195 + 0.6242i
   0.8195 - 0.6242i
  -1.0532 + 0.0000i
  -0.3351 + 0.9951i
  -0.3351 - 0.9951i

>> zeros=roots(s.num{1})

zeros =

    -2

>> rlocus(s)
```



$$\theta_n = \frac{(2k+1)180^\circ}{\#P - \#Z}$$

Para: $k = 0, 1 \ldots\ldots [(\#P - \#Z) - 1]$

$$\boldsymbol{\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z}}$$

$$k = [(\#P - \#Z) - 1] = (8 - 1) - 1 = 6$$
$$\boldsymbol{k = [0, 1, 2, 3, 4, 5]}$$

$$\gamma = \frac{\Sigma polos - \Sigma zeros}{\#P - \#Z} = \frac{(-2) - (-2)}{8 - 1} = \frac{0}{7} = 0$$
$$\boldsymbol{\gamma = 0}$$

$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(0)+1)180^{\circ}}{7} = \frac{180^{\circ}}{7} = 25.71^{\circ}$$
$$\boldsymbol{\theta_1 = 25.71^{\circ}}$$

$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(1)+1)180^{\circ}}{7} = \frac{540^{\circ}}{7} = 77.14^{\circ}$$
$$\boldsymbol{\theta_1 = 77.14^{\circ}}$$

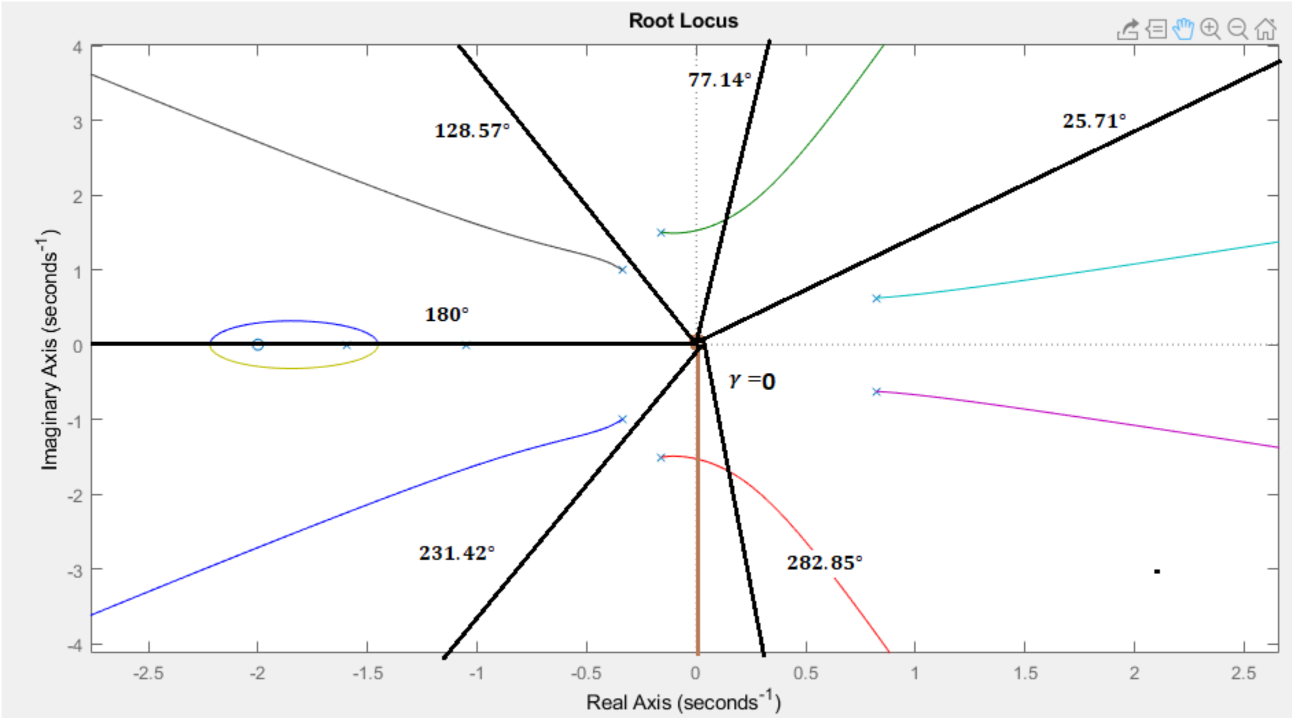
$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(2)+1)180^{\circ}}{7} = \frac{900^{\circ}}{7} = 128.57^{\circ}$$
$$\boldsymbol{\theta_1 = 128.57^{\circ}}$$

$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(3)+1)180^{\circ}}{7} = \frac{1260^{\circ}}{7} = 180^{\circ}$$
$$\boldsymbol{\theta_1 = 180^{\circ}}$$

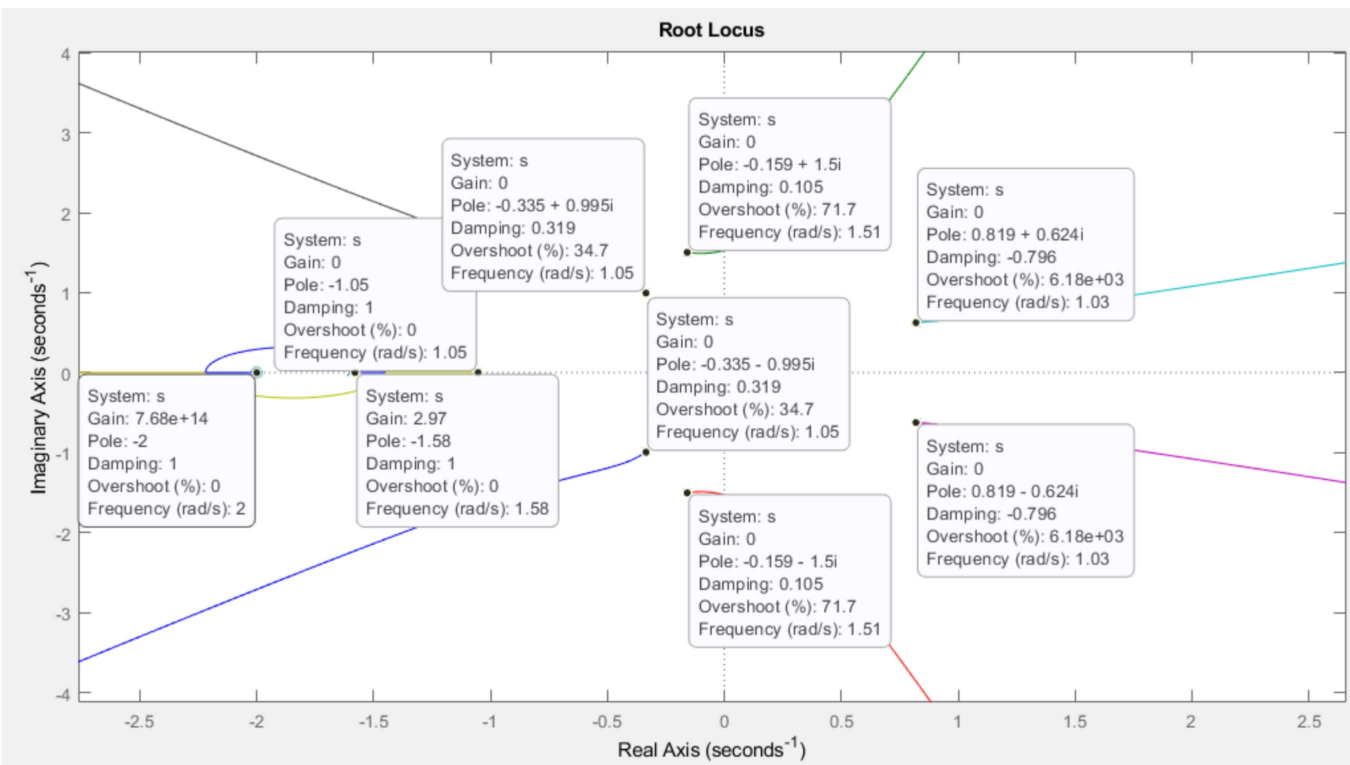
$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(4)+1)180^{\circ}}{7} = \frac{1620^{\circ}}{7} = 231.42^{\circ}$$
$$\boldsymbol{\theta_1 = 231.42^{\circ}}$$

$$\theta_1 = \frac{(2k+1)180^{\circ}}{\#P - \#Z} = \frac{(2(5)+1)180^{\circ}}{7} = \frac{1980^{\circ}}{7} = 282.85^{\circ}$$
$$\boldsymbol{\theta_1 = 282.85^{\circ}}$$

Las asíntotas quedarían de la siguiente manera:

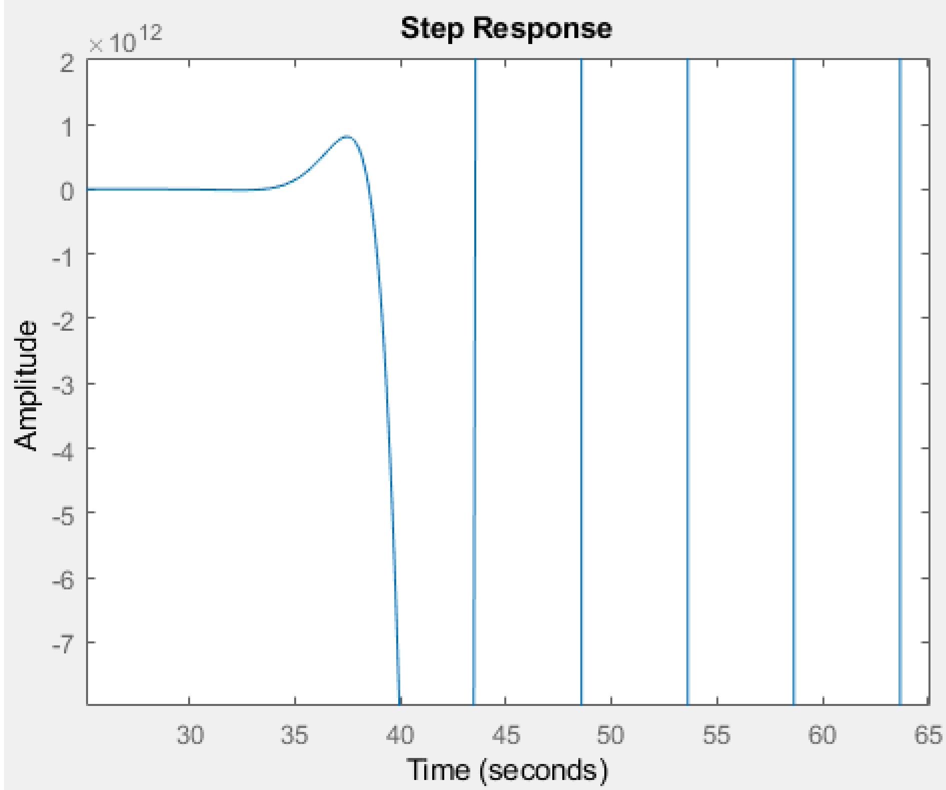


Root Locus
77.14°
128.57°
25.71°
180°
γ =0
231.42°
282.85°
Imaginary Axis (seconds-1)
Real Axis (seconds-1)



Root Locus
System: s
Gain: 0
Pole: -0.159 + 1.5i
Damping: 0.105
Overshoot (%): 71.7
Frequency (rad/s): 1.51
System: s
Gain: 0
Pole: -0.335 + 0.995i
Damping: 0.319
Overshoot (%): 34.7
Frequency (rad/s): 1.05
System: s
Gain: 0
Pole: 0.819 + 0.624i
Damping: -0.796
Overshoot (%): 6.18e+03
Frequency (rad/s): 1.03
System: s
Gain: 0
Pole: -1.05
Damping: 1
Overshoot (%): 0
Frequency (rad/s): 1.05
System: s
Gain: 0
Pole: -0.335 - 0.995i
Damping: 0.319
Overshoot (%): 34.7
Frequency (rad/s): 1.05
System: s
Gain: 7.68e+14
Pole: -2
Damping: 1
Overshoot (%): 0
Frequency (rad/s): 2
System: s
Gain: 2.97
Pole: -1.58
Damping: 1
Overshoot (%): 0
Frequency (rad/s): 1.58
System: s
Gain: 0
Pole: 0.819 - 0.624i
Damping: -0.796
Overshoot (%): 6.18e+03
Frequency (rad/s): 1.03
System: s
Gain: 0
Pole: -0.159 - 1.5i
Damping: 0.105
Overshoot (%): 71.7
Frequency (rad/s): 1.51
Imaginary Axis (seconds-1)
Real Axis (seconds-1)

En la siguiente gráfica se puede comprobar que el sistema si es inestable:



Bibliografía:

Apuntes de la materia "Teoría de Control y Robótica" por el Ingeniero David Tinoco Varela.

Anónimo (2019). "Algebra de Bloques". Disponible en: http://informatica.fquim.unam.mx/~fbarragan/index_archivos/algebra%20de%20Bloques.pdf

Anónimo (2020). "Algebra de Bloques- Diagrama de Bloques". Disponible en: http://www.cartagena99.com/recursos/alumnos/apuntes/TABLA%20DE%20ALGEBRA%20DE%20BLOQUES.pdf

StuDoc (2019). "Simplificación de diagrama de Bloques". Disponible en: https://www.studocu.com/es-mx/document/universidad-autonoma-de-guadalajara/control/apuntes/6-diagramas-a-bloques/3009787/view

"ITCM- Diagrama de bloques"(2013). Disponible en: https://www.youtube.com/watch?v=XchxZwp5ioA

"Función de transferencia en Matlab" (2018). Disponible en: https://www.youtube.com/watch?v=gVZIeicbqPk&t=398s

"Modelo Circuito RLC".(2013). Disponible en: https://www.youtube.com/watch?v=MNMZBnTpmTI&t=519s

"Ecuaciones Dinámicas de circuitos Eléctricos" (2017). Disponible en: https://www.youtube.com/watch?v=91cZjbkEi1g

Anónimo (2017). "Prácticas de Matlab: Transformada de Laplace". Disponible en: https://personales.unican.es/alvareze/CalculoWeb/CalculoII/practicas/PR11_TransformadaLaplace_16_17.pdf

MathWorks (2020). "Laplace". Disponible en: https://www.mathworks.com/help/symbolic/laplace.html

"Transformada directa e Inversa de Laplace en Matlab" (2012). Disponible en: https://www.youtube.com/watch?v=oOy0q7xB0CI

"Matlab- Transformada inversa de Laplace". Disponible en: https://www.youtube.com/watch?v=w8R9BBBczyY

"Transformada de Laplace para circuitos eléctricos" (2017). Disponible en: https://www.youtube.com/watch?v=r0TC4BNx834

StuDocu (2020). "Polinomios en Matlab". Disponible en: https://www.studocu.com/co/document/universidad-de-la-guajira/ingenieria-economica/otros/practica-matlab-2-polinomios-fracciones-parciales-matrices-laplace/5169998/view

Control de Proceso (2011). “Descomposición en fracciones parciales”. Disponible en: http://plantscontrol.blogspot.com/2012/02/5_08.html

“Fracciones Parciales | Matlab” (2016). Disponible en: https://www.youtube.com/watch?v=MY9DEX8soi4

“Fracciones Parciales Matlab” (2018). Disponible en: https://www.youtube.com/watch?v=bASGtqAvbNw

Anónimo (2015). “Sistemas de primer orden”. Disponible en: http://www.cartagena99.com/recursos/alumnos/apuntes/Cap_5_SDx.pdf

Anónimo (2017). “Sistemas de Primer y Segundo Orden”. Disponible en: http://www.udb.edu.sv/udb/archivo/guia/electronica-ingenieria/sistemas-de-control-automatico/2011/ii/guia-3.pdf

“Función de transferencia de Primer Orden” (2017). Disponible en: https://www.youtube.com/watch?v=IMEZ1Sw0cNA

Ma. del Carmen Cornejo Serrano , Eloísa Bernardett Villalobos Oliver , Santiago Molina Reséndiz , Willian Gerardo Arreola Galván. (2016). “Tipos de comportamientos de amrtiguación”. Disponible en: file:///C:/Users/David%20Lobato/Downloads/664-1871-4-PB.pdf

Roberto Cárdenas Dobson. (2018). “Sistemas de segundo orden”. Disponible en: file:///C:/Users/David%20Lobato/Downloads/sistema_de_segundo_orden.pdf

Méndez Pérez Juan Albino. (2017). “Respuestas de sistemas de segundo orden”. Disponible en: https://www.youtube.com/watch?v=QEWXpYiH2to

(2018). “Análisis de funciones de transferencia de segundo orden usando Matlab”. Disponible en: https://www.youtube.com/watch?v=tllf-hDOxZ0

Anónimo (2016). “Diagrama de Bode”. Disponible en: http://anthony-diagrama.blogspot.com/2016/01/diagrama-de-bode.html

Castaño Giraldo Sergio (2015). “Diagrama de Bode asintótico y Matlab”. Disponible en: https://www.youtube.com/watch?v=nynG2aFgVfw&t=761s

Ing, David Tinoco Varela (2020). “Respuestas en frecuencia”. Obtenido de: https://e.edim.co/176399701/GnWXnbNSwTznZYIm.pdf?response-content-disposition=filename%3D%22Respuesta_en_frecuencia.pdf%22%3B%20filename%2A%3DUTF-8%27%27Respuesta%2520en%2520frecuencia.pdf&Expires=1587674099&Signature=N6bDVh738eugy3to9IP24v9sbbAOPAtsJdiDeiwSDXM4WWJhxZPmQp2v5Q~nhYDtN7xi9XFjX007LgRIbbqOm5olxqk82IAl12w0kWciGQ0T26BfyXjDdlV59dDljcpGv~e-VxC3gBA4owFWhuFId01uliZ6k1iIzqdzGcu7LV~hNZZv7ptt2XbynJKjtXxh-~lEdNPvGBbEtwGKYyClZ0iNxUsWGdGivMhLxmHzxs1GChtfHUbCRKHRnVk8dl4QMlh4VlcoUivxAG59wcZ7VYBIno5eZUnawXnfs0fFmB8OMEI2dJOPI6AzHxmSQOG9OSEFKEvrX5F0NnfRt5DHhg__&Key-Pair-Id=APKAJMSU6JYPN6FG5PBQ

Anónimo (2015). “Diagrama de Bode- Trazado Asintótico de una función de transferencia”. Disponible en: https://www.youtube.com/watch?v=R03tJqyKso4&t=423s

“Controlador de acción proporcional, integral y derivativa”. Disponible en: http://e-ducativa.catedu.es/44700165/aula/archivos/repositorio/4750/4926/html/15_controlador_de_accin_proporcional_integral_y_derivativa_pid.html

“Controlador de acción proporcional e integral”. Disponible en: http://e-ducativa.catedu.es/44700165/aula/archivos/repositorio/4750/4926/html/13_controlador_de_accin_proporcional_e_integral_pi.html

“Controlador de acción proporcional”. Disponible en:http://e-ducativa.catedu.es/44700165/aula/archivos/repositorio/4750/4926/html/11_controlador_de_accin_proporcional_p.html

Izquierdo Fernandez Jonatan (2011). “Controladores PI con acción de reset”. Disponible en: https://ddd.uab.cat/pub/trerecpro/2011/hdl_2072_199563/PFC_JonatanIzquierdoFernandez.pdf

Tinoco Varela David (2020). “Lugar Geométrico de las Raíces”. Disponible en: https://e.edim.co/176399701/x6oZX6qjkGUQVAT1.pdf?response-content-disposition=filename%3D%22Lugar_geom_trico_de_la_ra_z.pdf%22%3B%20filename%2A%3DUTF-8%27%27Lugar%2520geom%25C3%25A9trico%2520de%2520la%2520ra%25C3%25ADz.pdf&Expires=1589754434&Signature=EKCFNqXvRjOZaK0Zgoq7kacYOgV2WPz9BVEk7xm2tdUq1cMGRLD3ZwAP6X3XW5J~O8Duw9qwM~5dHTuB~EUaTcDLdp8U8~tAW~HMjjiwHg6XMLcsKcfOK~f-QH4YtW5mgXvRWcMuilo32zqU~HV1nLOi4GHTrnsVbFaqfl766sqaRZEuL0gtUxkELKabDydS6mvC3kvmNkrYTUBB9sLc0Z6dsrTe1ylUK-z6gZTsZnZjDW-tFcMgv2JoxH7qrrPfFNqk5gmXvHnYYEcb02DGiB1XiQBo-FSF8LeWi5ErzcMXJQ4493zqCv8yEZmO84pm-mdp~GFzbMkahSEvlbUY5w__&Key-Pair-Id=APKAJMSU6JYPN6FG5PBQ

Automática (2017). “Lugar de las Raíces con Matlab”. Disponible en: https://www.youtube.com/watch?v=CdKE08LlK3o&fbclid=IwAR1v0VpaaB2dkYZpv_N-19xNY1K8H2VHFaL_gOnRDlqQ04I98dQKz1C3uV4

Anónimo (2018). “Lugar geométrico de las Raices de un sistema de control: Parte 1”.Disponible en: https://dademuch.com/2018/05/16/el-lugar-geometrico-de-las-raices/

Anónimo (2018). “Lugar geométrico de las Raices de un sistema de control: Parte 2”.Disponible en: https://dademuch.com/2018/05/19/el-lugar-geometrico-de-las-raices-de-un-sistema-de-control-2da-parte/