

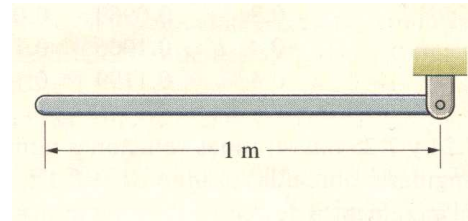
**UNIVERSIDAD DE IBAGUÉ – CORUNIVERSITARIA**  
**TUTORIAL – WORKING MODEL**

**Diego Andrés Montoya Chacón      Cod: 2120041007**

En este tutorial haremos el ejercicio de una barra esbelta apoyándonos en el uso de un simulador y lo vamos a comprobar de manera analítica, para así poder confirmar la respuesta que nos arrojará la simulación.

### EJERCICIO DE UNA BARRA ESBELTA

La barra esbelta de 10 Kg se libera desde el reposo en la posición horizontal mostrada. Suponga que el soporte de pasador de la barra contiene un dispositivo amortiguador que ejerce un par resistente sobre la barra de magnitud



$c\omega$  (N-m), donde  $\omega$  es la velocidad angular en rad/s. Usando unas diferencias de tiempo de 0.001 seg., dibuje las gráficas de la velocidad angular de la barra como función del tiempo de  $t=0$  a  $t=0.8$  seg. para los casos en que  $c=0$ ,  $c=2$ ,  $c=4$  y  $c=8$ .

Desarrollo

#### 1. Configure su espacio de trabajo:

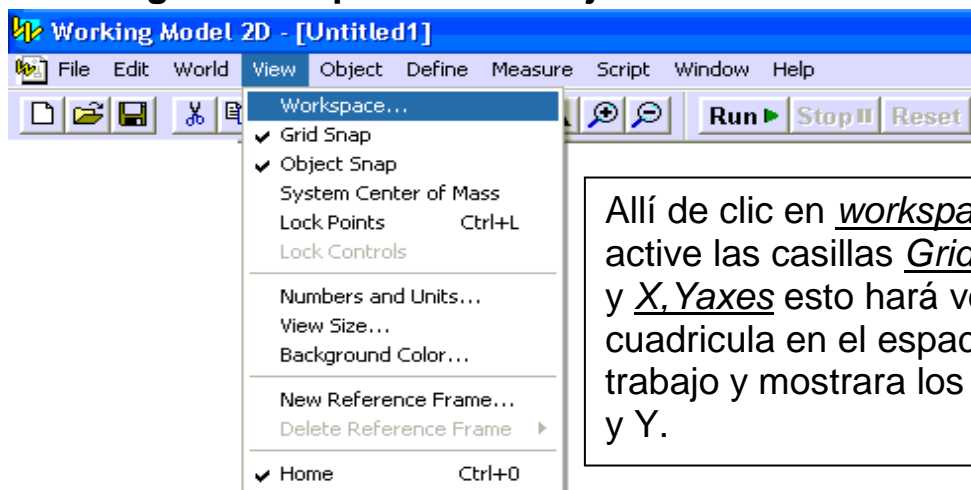


Figura 1.

## 2. Seleccione las unidades a trabajar:

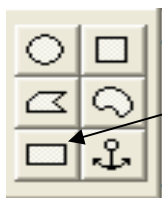
Esto es muy importante determinarlo desde el principio porque sino el simulador me puede generar errores en los cálculos. Analizando el problema vemos que podemos trabajar todas las unidades en el sistema internacional la única variación que debemos hacer es cambiar los grados por radianes debido a que la velocidad angular el ejercicio nos la exige en rad/seg.

Para realizar lo anteriormente dicho nos vamos a la opción “Numbers and Units” que se ve en la figura 1; luego colocamos las unidades en SI y damos clic en “more choices” para cambiar los grados a radianes.

## 3. Dibuje el sistema:

Como vemos este sistema es muy simple ya que consta de un solo sólido el cual es la barra esbelta.

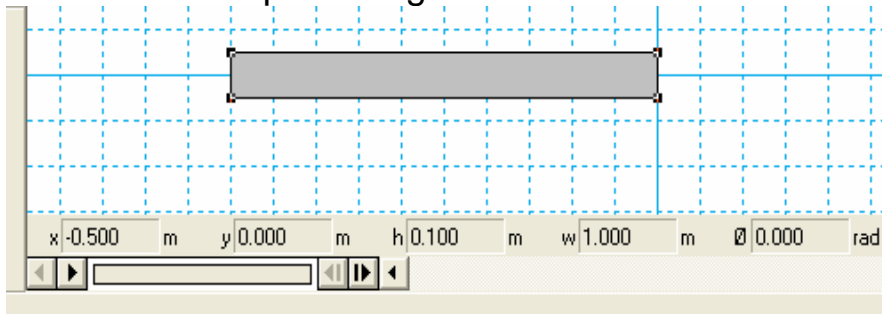
Para dibujarla usamos las opciones para generar sólidos, debido a que es una barra usaremos la opción “rectangle” y la damos las medidas 1m de ancho y un espesor pequeño teniendo en cuenta que el problema nos dice que es una barra esbelta que puede ser de 0.1m. Trate que el sólido este cerca de la coordenada (0,0).



### OPCIONES PARA GENERAR SOLIDOS

Rectangle

La barra debe quedar algo así:



En la parte baja de la grafica notamos las dimensiones del cuerpo  $w=1$  (ancho) y el  $h=0.1$  (alto).

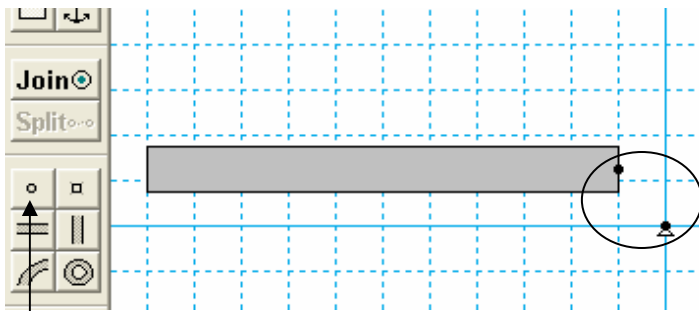
Debemos recordar que la barra tiene una masa de 10 Kg y esta se le asigna haciendo doble clic sobre el objeto y sale un recuadro con opciones del objeto entre estas la masa.

#### 4. Coloque la unión:

Como vemos en el dibujo del ejemplo notamos que la barra esta conectada a un punto fijo sobre el cual rota.

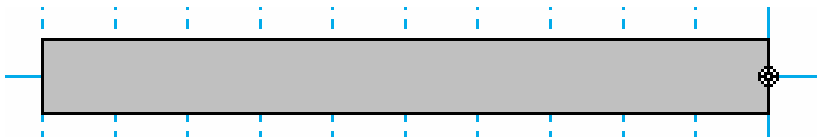
En el programa esta opción es posible por medio de un "point element" el cual se le coloca al punto donde queremos que este fijo el objeto (para nuestra comodidad lo colocaremos en el centro del sistema) y de la parte de la cual queremos que el objeto se encuentre sujeto (por la configuración del problema lo colocaremos en la mitad de la parte angosta de la barra para que su rotación sea balanceada).

Luego de colocar los "point element" los uniremos por medio del siguiente proceso escogemos uno de los point y con la tecla shif oprimida seleccionamos el otro point, con esto se nos activara una casilla que tiene la palabra "join" la cual unirá los dos "point join" y así podremos generar el movimiento de rotación.



Los dos point element estan seleccionados y se activa la opción join

(o)Point element

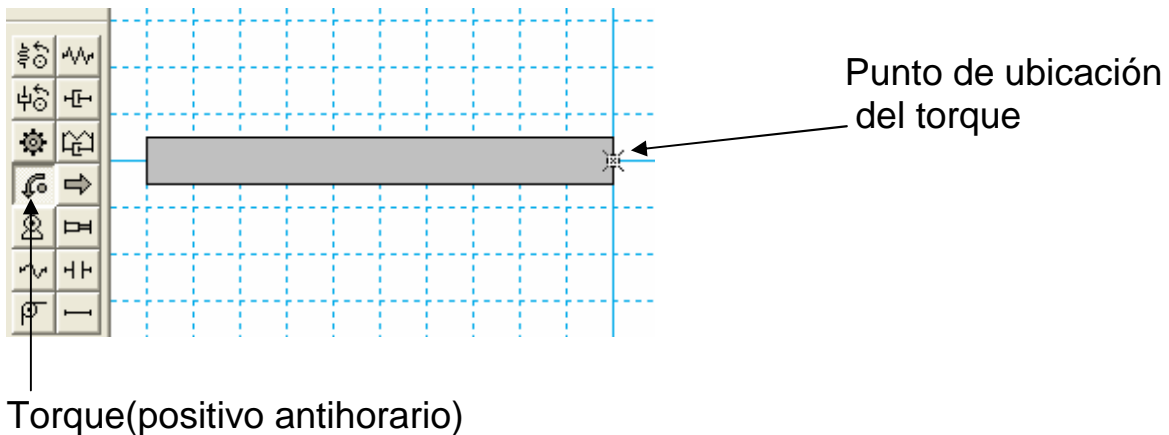


Así queda la barra después de ser unida a un punto para que genere el movimiento de rotación pura.

#### 5. Colocar el momento de amortiguamiento:

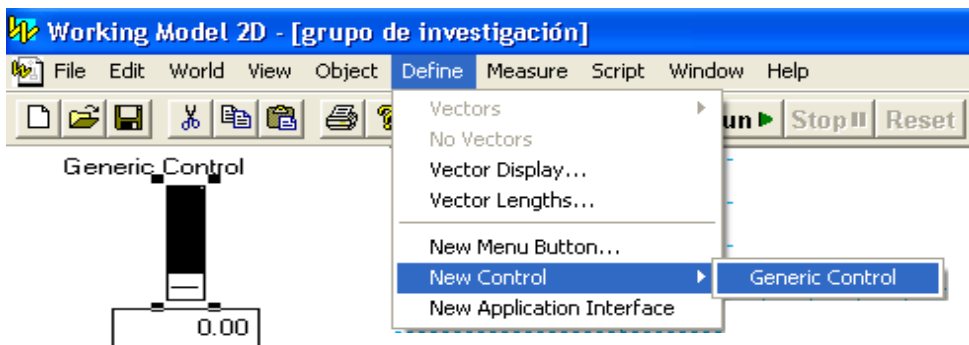
Cuando en el problema nos hacen referencia de esto debemos tener en cuenta que este momento se opondrá al movimiento y hará que el sistema se mueva más lento.

Un momento o troqué se coloca por medio de la opción "Torque" que se encuentra en la barra de herramientas. Se da clic a la opción y la vamos a colocar a la parte fija de la barra. La magnitud se la vamos a dar cuando coloquemos un control para variar la  $e$  que nos dan en el problema debido a que esta es variable.



## 6. Control de valores para $C$ .

Debido a que los valores varían en el problema desde cero hasta ocho de dos en dos debemos colocar un dispositivo para variar este valor.



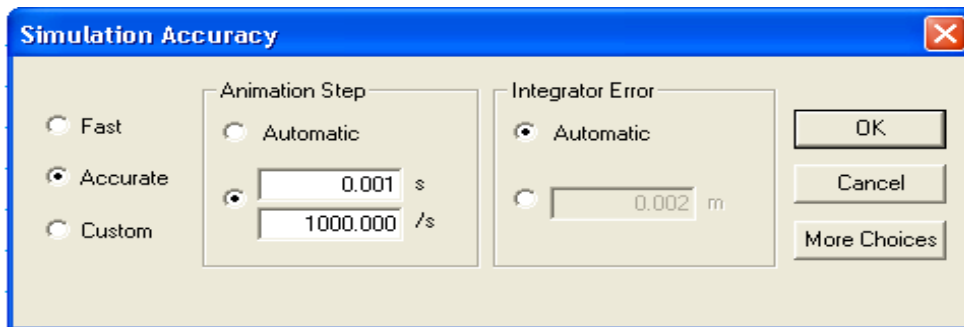
Esta es la ruta para crear un "Generic Control" que como ya lo habíamos dicho antes es para variar el valor de  $C$  para nuestro caso. Para variar los valores máximos y mínimos del "Generic control" lo hacemos desde sus propiedades que se muestran se le damos doble clic sobre el generis control para nuestra caso el máximo será 8 y el mínimo será 0.

## 7. Darle los valores al momento de amortiguamiento:

Debido a que el momento de amortiguamiento se le va a dar diferentes valores ya que su magnitud va ser igual  $C\omega$  debemos dejarlo en relación a estos valores. Se debe tener en cuenta que  $C$  es un valor constante para cada experimento el cual se lo damos a través del “Generic control”; pero  $\omega$  es variable con el tiempo y esto lo podemos medir desde el rectángulo y la línea de código para hacerlo es **Body[1].v.r**. Para colocar la magnitud del momento de amortiguamiento daremos doble clic sobre la convención de momento que colocamos previamente en la barra, allí se desplegara un recuadro donde habrá una casilla que dice **Value** y escribiremos en ese espacio **-Input[7]\*Body[1].v.r\*** se le coloca el menos ya que es para que se oponga al movimiento debido a que la barra hace el movimiento en sentido anti-horario y en ese sentido el programa toma el movimiento como positivo y así poder amortiguarlo.

## 8. Diferencias de tiempo para los cálculos:

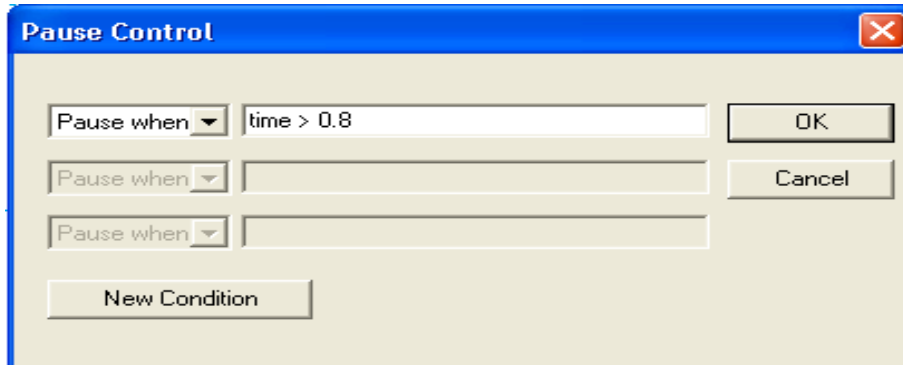
Como en el problema nos dice que usemos diferencias de tiempo de 0.001 seg. esta condición se le da al programa por medio de la siguiente ruta: *despliegue World, y escoja accuracy* allí saldrá un recuadro y en *animation step* coloque el valor 0.001 seg como se ve en la figura.



\* (El número se le asigna a los objetos en el programa por orden de aparición en caso tal que no haya seguido el tutorial en el orden puede ver que número le corresponde a la barra haciendo doble clic sobre esta y aparecerá un recuadro que en la parte superior mostrara el nombre del objeto y su número de aparición “Body[?]-Rectangle” lo otro que le agregamos a la palabra *body* es para que nos de la magnitud de la velocidad angular del cuerpo por medio de *.v.r* y la palabra “Input[7]” es la que el programa le da al “*generic control*” recuerde que el número que acompaña al *input* es por el orden de aparición).

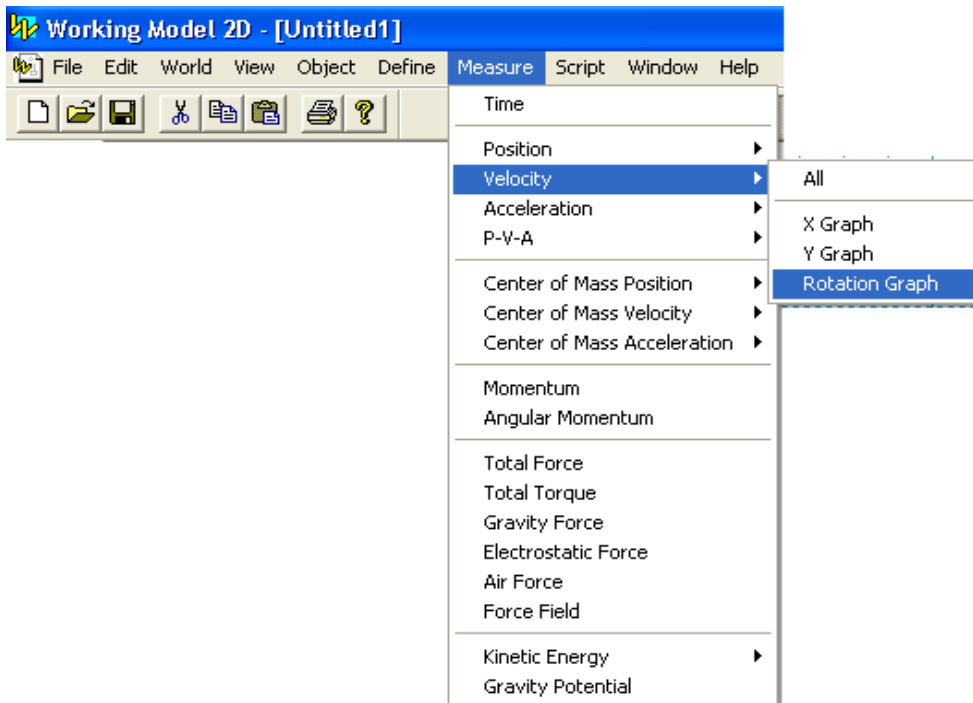
## 9. Limite de tiempo:

Como el problema nos dice que analicemos el movimiento de 0 a 0.8seg debemos colocar un control de pausa para restringir el tiempo de la simulación. Esto lo hacemos siguiendo esta ruta: *World*, y escoja *pause control*, allí le aparecerá un recuadro y darán clic en la opción *New condition* y colocaran la opción que se ve en la figura.



## 10. Obtención de la grafica:

Par obtener la grafica que nos piden que es velocidad angular( $\omega$ ) contra el tiempo, siendo el tiempo la propiedad independiente, seleccionamos a lo que le queremos sacar la grafica; para nuestro caso será la barra y seguimos la siguiente ruta:



Luego de hacer esta ruta pulse la opción “RUN” y listo obtiene su grafica; pero si usted quiere ver todas las graficas en una misma imagen para poder compararlas tendrá que sacar los datos para cada caso y verlos en MATLAB.



### 11. Extraer los datos de la simulación:

Podemos sacar los datos de la simulación para cada valor de  $c$  Este proceso es sencillo seguimos la siguiente ruta: *File, escoja Export*, Luego de hacer esto guarde los datos con cualquier nombre, nosotros lo llamaremos TABLA1 para  $c=0$ , TABLA2 para  $c = 2$ , TABLA3 para  $c= 4$  y TABLA4 para  $c = 8$ .

Luego de tener los datos convertimos los archivos a *bloc de notas* es decir con una extensión de .txt y los modificamos para que MATLAB los pueda leer; para nuestro caso le colocaremos el símbolo de porcentaje(%) a las líneas que tengan letras. Y colocamos el siguiente código para que nos de las graficas de *t Vs.  $\omega$* .

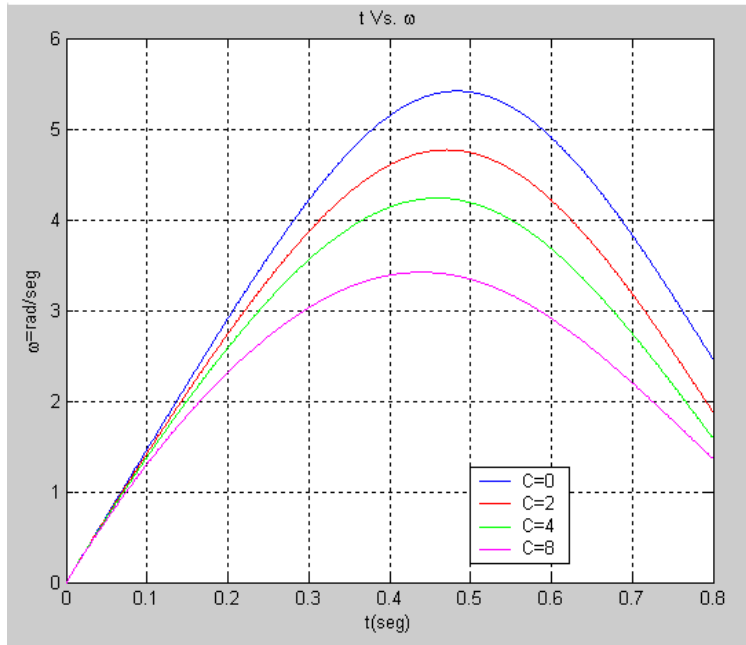
Primero en MATLAB la cargamos teniendo en cuenta que el archivo se encuentre en el directorio del programa :

```
-load TABLA1.txt  
y así para las otras tres tablas
```

Ahora en el scrip escriba la siguiente línea de código:

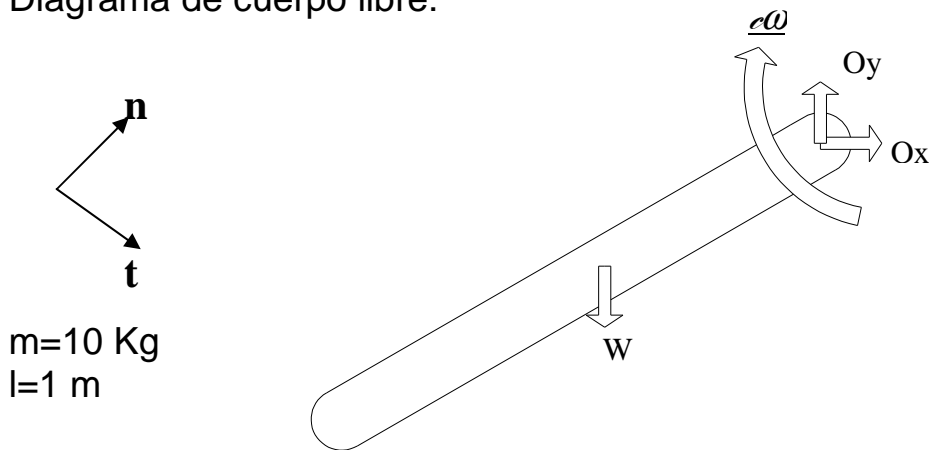
```
clc  
plot(TABLA1(:,1),TABLA1(:,2))  
title('t Vs. \omega')  
xlabel('t(seg)')  
ylabel('\omega=rad/seg')  
hold on  
plot(TABLA2(:,1),TABLA2(:,2),'r')  
plot(TABLA3(:,1),TABLA3(:,2),'g')  
plot(TABLA4(:,1),TABLA4(:,2),'m')  
legend('C=0','C=2','C=4','C=8',0)  
grid
```

Debe aparecerle la siguiente grafica:



### SOLUCION POR EL METODO ANALITICO

Diagrama de cuerpo libre:



La sumatoria de momentos respecto a O con sentido positivo antihorario

$$\Sigma M_o = I_o \alpha$$

$$-c\dot{\theta} + (0.5)mg\cos\theta = \frac{1}{3}ml^2\alpha$$

$$\frac{1}{3}ml^2\ddot{\theta} + c\dot{\theta} - (0.5)mg\cos\theta = 0$$

$$\ddot{\theta} + \frac{3c}{ml^2}\dot{\theta} - \frac{3(0.5)mg\cos\theta}{ml^2} = 0$$

$$\ddot{\theta} + \frac{3c}{10*1^2}\dot{\theta} - \frac{3*(0.5)*10*9.81\cos\theta}{10*1^2} = 0$$

$$\omega = \frac{d\theta}{dt} = \dot{\theta}$$

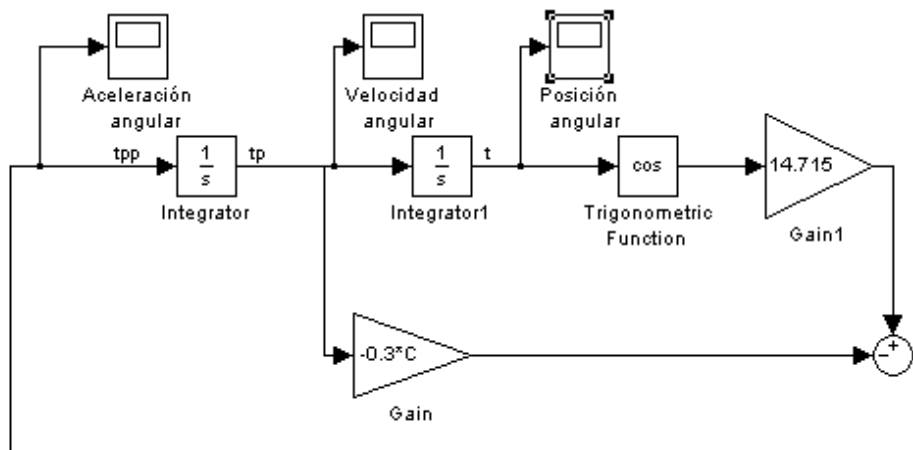
$$\alpha = \frac{d\omega}{dt} = \frac{d^2\theta}{dt^2} = \ddot{\theta}$$

$$\ddot{\theta} + 0.3c\dot{\theta} - 14.715\cos\theta = 0$$

Ecuación diferencial que describe el movimiento de la barra.

Ahora usando simulink\* generaremos la grafica que nos describira el movimiento que realiza la barra para un de t=0 hasta t=0.8.

Pero para introducirla al programa debemos despejar  $\ddot{\theta}$  y con esta ecuación colocarla en diagrama de bloques así:



\***Simulink**: Es un toolbox de MATLAB, el cual permite la simulación de ecuaciones diferenciales a partir de un método grafico usando bloques para formar la ecuación.

Como podemos ver en este diagrama también podemos obtener la grafica de aceleración angular y posición angular, lo cual es muy beneficioso para analizar diferentes tipos de situaciones.

Para obtener las graficas para cada caso guardaremos el archivo de bloques con el nombre de barra, y la grafica que nos genera el diagrama de bloques lo denominaremos velocidadangular este se debe guardar como un array(matriz) y ahora si colocamos en el scrip de MATLAB las siguientes lineas de codigo:

```
C=0 % Defincion de la constante
sim('barra') % Simulacion del diagrama
plot(velocidadangular(:,1),velocidadangular(:,2)) % Obtencion de la grafica
hold on

C=2 % Defincion de la constante
sim('barra') % Simulacion del diagrama
plot(velocidadangular(:,1),velocidadangular(:,2),'r') % Obtencion de la grafica

C=4 % Defincion de la constante
sim('barra') % Simulacion del diagrama
plot(velocidadangular(:,1),velocidadangular(:,2),'g') % Obtencion de la grafica

C=8 % Defincion de la constante
sim('barra') % Simulacion del diagrama
plot(velocidadangular(:,1),velocidadangular(:,2),'m') % Obtencion de la grafica

title('t Vs. \omega')
xlabel('t(seg)')
ylabel('\omega=rad/seg')
legend('C=0','C=2','C=4','C=8',0)
grid
```

Ahora comparando las graficas dadas por medio de los dos métodos utilizados, nos damos cuenta que el ejercicio realizado analíticamente quedo bien, siendo este comprobado por medio de la simulación.