

PARTIE I : COMMANDES AVANCEES :

I.1 REPRESENTAION D'UN POLYNÔME :

Dans MATLAB, un polynôme $F(p)$ de degré n est représenté sous la forme codée d'un vecteur ligne F de $n+1$ termes. Ceux-ci sont les coefficients des puissances de p ordonnées par valeurs décroissantes.

EXEMPLE 1:

$$F(p) = 9p^5 + 6p^2 + 2$$

$$F=[9 \ 0 \ 0 \ 6 \ 0 \ 2]$$

$$F =$$

$$9 \quad 0 \quad 0 \quad 6 \quad 0 \quad 2$$

Remarque : Les puissances absentes sont représentées par le terme 0.

Lorsqu'une instruction Matlab est suivie d'un 'point virgule' le résultat n'est pas visualisé à l'exécution.

I.2 OPERATIONS SUR LES POLYNÔMES :

I.2.1 CALCUL DES RACINES D'UN POLYNÔME :

roots() : calcule les racines du polynôme.

EXEMPLE 2 :

$$F(p) = 9p^5 + 6p^2 + 2$$

$$F=[9 \ 0 \ 0 \ 6 \ 0 \ 2]$$

$$F =$$

$$9 \quad 0 \quad 0 \quad 6 \quad 0 \quad 2$$

$$\text{roots}(F)$$

$$\text{ans} =$$

-0.9670 + 0.0000i

0.5425 + 0.6834i

0.5425 - 0.6834i

-0.0590 + 0.5462i

-0.0590 - 0.5462i

I.2.2 DERIVATION DE POLYNÔMES

Polyder() : réalise la dérivation du polynôme

EXEMPLE 3 :

$$F(p) = 9p^5 + 6p^2 + 2$$

F=[9 0 0 6 0 2];

polyder(F)

ans =

$$45 \quad 0 \quad 0 \quad 12 \quad 0 \rightarrow 45p^4 + 0p^3 + 0p^2 + 12p^1 + 0p^0$$

I.2.3 EVALUATION D'UN POLYNÔME POUR UNE VALEUR DONNEE :

Polyval(f,val) : donne la valeur numérique que prend le polynôme lorsqu'on lui applique la valeur numérique val.

EXEMPLE 4 :

$$F(p) = 9p^5 + 6p^2 + 2$$

F=[9 0 0 6 0 2];

polyval(F,2)

ans =

314

I.2.4 MULTIPLICATION DE POLYNÔMES :

z=conv(x,y) : crée le polynôme $z = x \times y$ sous sa forme développée. Le degré du polynôme z est la somme des degrés des polynômes x et y.

EXEMPLE 5 :

$$F(p) = 9p^5 + 6p^2 + 2$$

$$F=[9 \ 0 \ 0 \ 6 \ 0 \ 2]$$

$$F =$$

$$9 \quad 0 \quad 0 \quad 6 \quad 0 \quad 2$$

$$g=[1 \ 2 \ 5] \rightarrow 1p^2 + 2p^1 + 5p^0$$

$$g =$$

$$1 \quad 2 \quad 5$$

$$z=\text{conv}(F,g)$$

$$z =$$

$$9 \quad 18 \quad 45 \quad 6 \quad 12 \quad 32 \quad 4 \quad 10 \rightarrow 9p^7 + 18p^6 + 45p^5 + 6p^4 + 12p^3 + 32p^2 + 4p^1 + 10p^0$$

I.2.5 LES ZEROS ET LES PÔLES D'UNE FONCTION :

zero(sys) : donne les racines du numérateur.

pole(sys) : donne les racines du dénominateur.

EXEMPLE 6:

On calcule les zéros et les pôles de la fonction : $\text{sys} = \frac{p^2+2p+5}{4p^2+5p+6}$

$$\text{num}=[1 \ 2 \ 5]$$

$$\text{num} =$$

$$1 \quad 2 \quad 5$$

$$\text{den}=[4 \ 5 \ 6]$$

$$\text{den} =$$

$$4 \quad 5 \quad 6$$

$$\text{sys}=\text{tf}(\text{num},\text{den})$$

```
sys =
```

```
s^2 + 2 s + 5
```

```
-----
```

```
4 s^2 + 5 s + 6
```

Continuous-time transfer function.

```
zero(sys)
```

```
ans =
```

```
-1.0000 + 2.0000i
```

```
-1.0000 - 2.0000i
```

```
pole(sys)
```

```
ans =
```

```
-0.6250 + 1.0533i
```

```
-0.6250 - 1.0533i
```

I.2.6 LA TRANSFORMEE DE LAPLACE:

Matlab permet de calculer les transformées de Laplace et les transformées inverses de Laplace.

Syms : définit les symboles "t" et "s".....

laplace : calcule la transformée de Laplace de l'expression donnée.

EXEMPLE 7 :

```
syms t
```

```
f=sin(t);
```

```
F=laplace(f)
```

F =

$1/(s^2 + 1)$

Remarque :

Matlab utilise (s) qui est la variable (p) de la TL.

I.2.7 LA TRANSFORMEE INVERSE DE LAPLACE

ilaplace : calcule la transformée inverse de Laplace de l'expression donnée.

EXEMPLE 8:

```
syms s
```

```
ilaplace(1/(s^2 + 1))
```

```
ans =
```

```
sin(t)
```

PARTIE II : CONTROL SYSTEM TOOLBOX ET SIMULINK

A. CONTROL SYSTEM TOOLBOX

II.1 DEFINITION D'UN SYSTEME LINEAIRE:

II.1.1 DEFINITION D'UN SYSTEME LINEAIRE PAR SA FONCTION DE TRANSFERT:

Sys=tf(num, den)

num : c'est le polynôme numérateur.

den : c'est le polynôme dénominateur.

tf : définit une fonction de transfert

Exemples :

EXEMPLE 9 :

```
num=5;
```

```
den=[1 2];  
sys1=tf(num,den)  
  
sys1 =  
      5  
-----  
      s + 2  
Continuous-time transfer function.
```

EXEMPLE 10 :

```
num=[7 5];  
den=[1 6 0];  
sys2=tf(num,den)  
sys2 =  
      7 s + 5  
-----  
      s^2 + 6 s  
Continuous-time transfer function.
```

EXEMPLE 11 :

```
num=[1 2 4];  
den=[4 5 6];  
sys3=tf(num,den)  
sys3 =  
      s^2 + 2 s + 4  
-----  
      4 s^2 + 5 s + 6  
Continuous-time transfer function.
```

II.1.2 DEFINITION D'UN SYSTEME LINEAIRE PAR SES POLES ET ZEROS:

Sys= zpk (zer,pol,gain)

zer: vecteur ligne qui donne la liste des zéros. (Les zéros sont les racines du numérateur)

pol: vecteur ligne qui donne la liste des pôles. (Les pôles sont les racines du dénominateur)

gain: est un scalaire, c'est le gain global que l'on peut mettre en facteur de l'ensemble.

Exemples :

EXEMPLE 12 :

```
zer=[ ];  
pol=[-2];  
gain=[1];  
sys1=zpk(zer,pol,gain)  
sys1 =  
    1  
-----  
    (s+2)  
Continuous-time zero/pole/gain model.
```

EXEMPLE 13 :

```
zer=[1 2];  
pol=[2 5];  
gain=[5];  
sys2=zpk(zer,pol,gain)  
  
sys2 =  
    5 (s-1) (s-2)  
-----  
    (s-2) (s-5)  
Continuous-time zero/pole/gain model.
```

Remarque :

La fonction roots() permet de calculer les racines d'un polynôme et de déterminer les zéros et les pôles d'une fonction de transfert .

II.1.3 DEFINITION D'UN SYSTEME LINEAIRE EN INTRODUISANT SA FONCTION DE TRANSFERT DIRECTEMENT :

EXEMPLE 14:

```
syms s
s=tf('s');
sys1=(1/(s+2)^2)
```

sys1 =

$$\frac{1}{s^2 + 4s + 4}$$

Continuous-time transfer function.

B. SIMULINK :

II.1 PRESENTATION

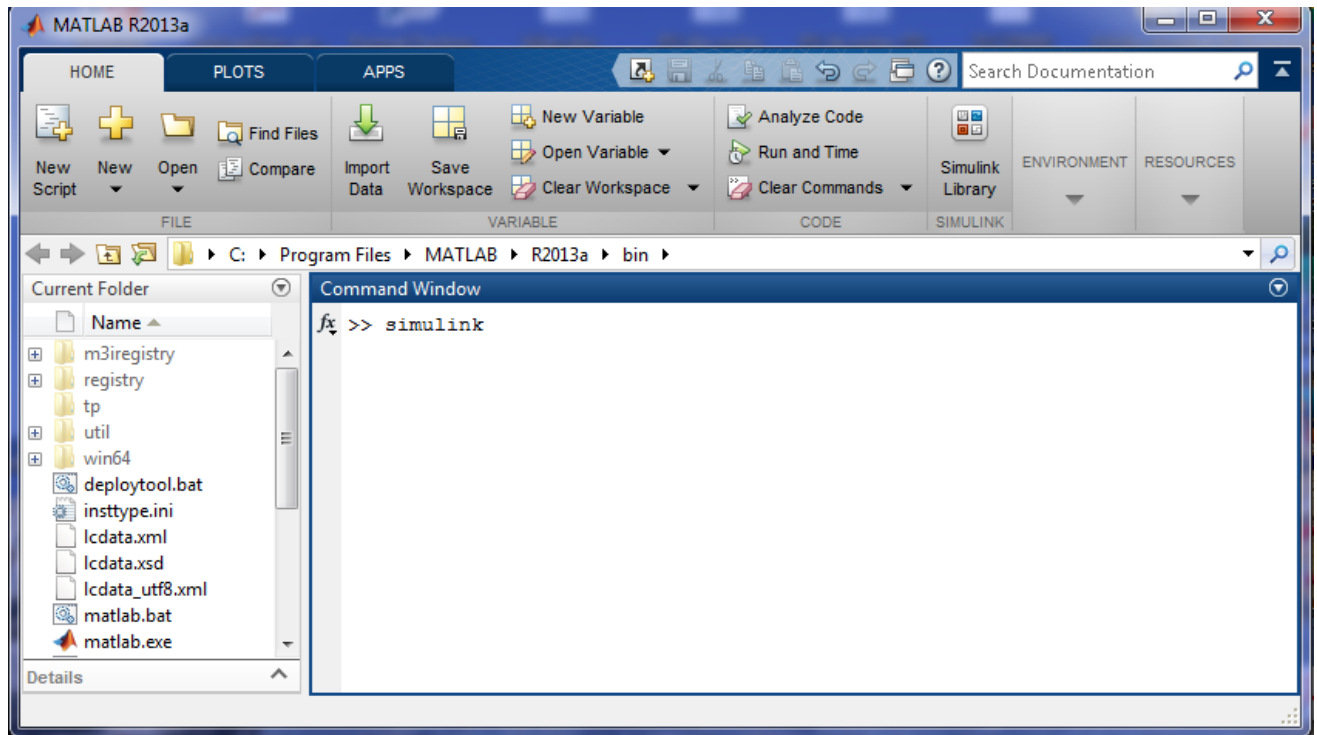
Simulink est une interface graphique qui facilite l'analyse des systèmes dans le domaine temporel.

Les systèmes ne sont pas décrits par des lignes de code Matlab mais simplement définis par des blocs dont tous les éléments sont prédéfinis dans des bibliothèques qu'il suffit d'assembler.

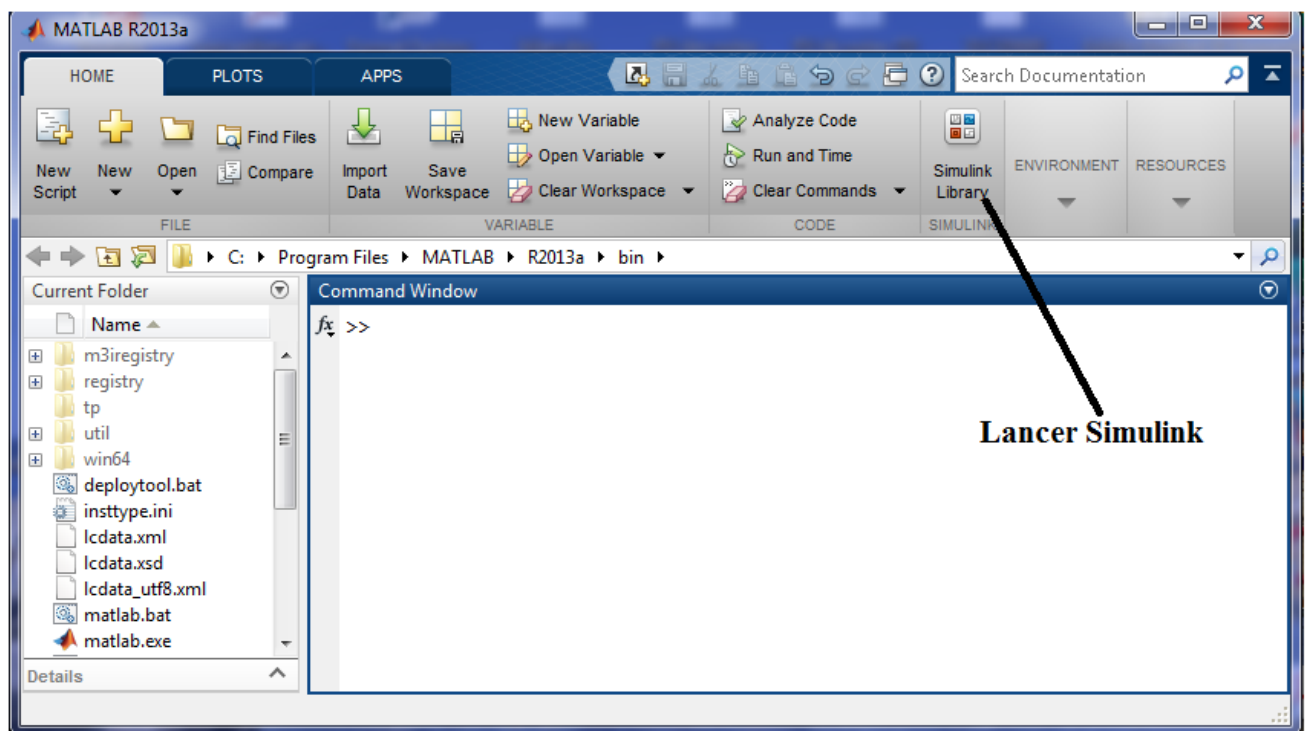
Lorsque le schéma bloc du système que l'on étudie est représenté sous Simulink, il est possible d'analyser sa réponse temporelle pour des entrées diverses (échelon, rampe.....).

Pour lancer Simulink, on procède à partir de la fenêtre de commande de Matlab :

*Soit on tape à la suite du prompt Matlab (>>) tout simplement la commande Simulink.

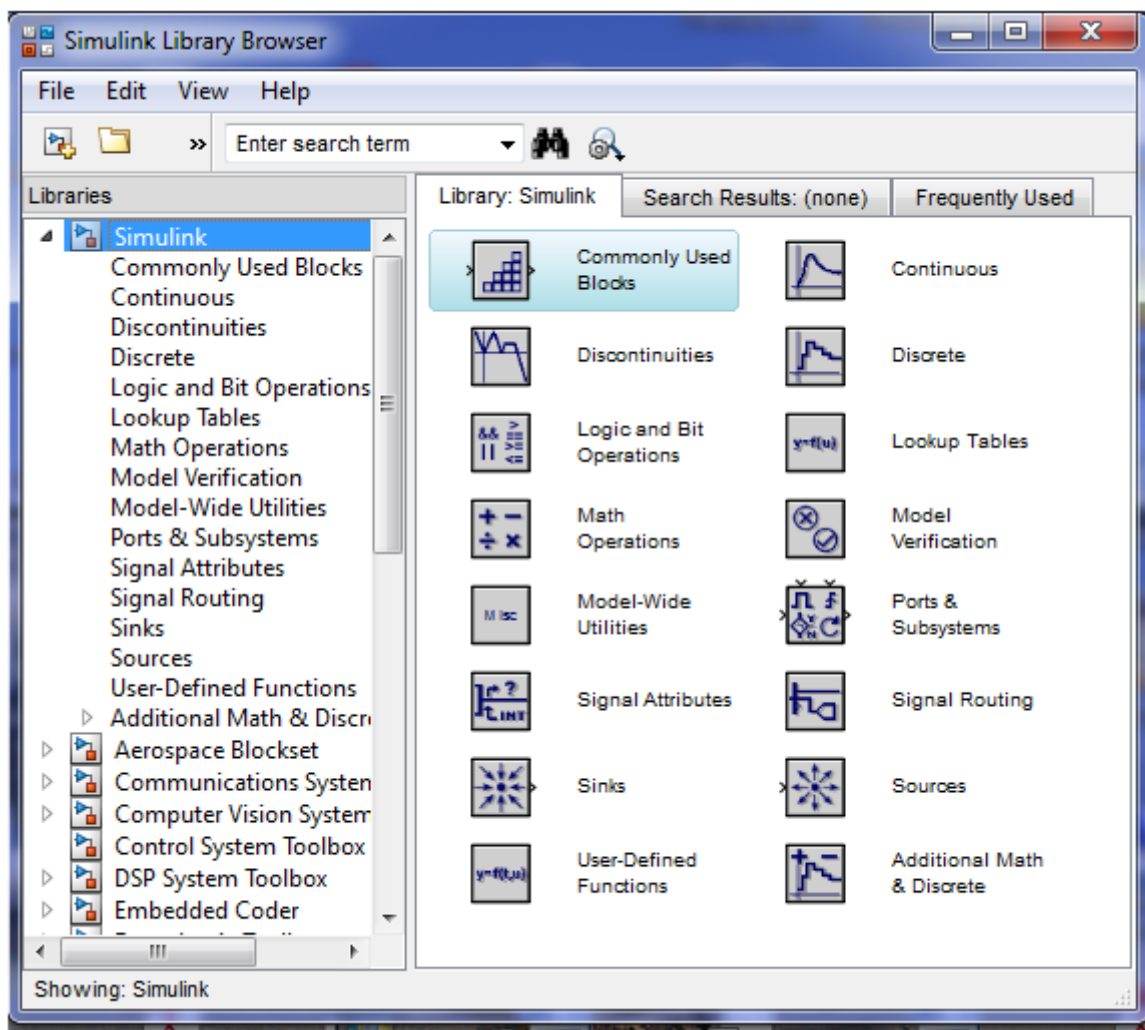


*Soit depuis la barre d'outils de Matlab, on clique sur l'icône dédiée à Matlab.



Ces deux actions ouvrent la fenêtre Simulink Library Browser qui permet l'accès à la Bibliothèque Simulink ainsi qu'à d'autres bibliothèques (control system, par exemple). Simulink, comme chaque bibliothèque importante, comporte des compartiments : Continuous, Discrete, Source.... qui contiennent les blocs.

II.2 QUELQUES BIBLIOTHEQUES :



II.2.1 Bibliothèque Source:

Les blocs sont utilisés pour générer des signaux, ils possèdent une ou plusieurs sorties et aucune entrée.

- ✓ **Step** : génère un échelon d'amplitude réglable.

- ✓ **Ramp** : génère une rampe de pente réglable.
- ✓ **Sin Wave** : génère une sinusoïde d'amplitude, pulsation et déphasage réglables.
- ✓ **Constant** : délivre un signal constant dans le temps et de niveau réglable.

III.2.2 Bibliothèque Sinks :

Les blocs sont utilisés pour l'affichage des signaux, ils possèdent une ou plusieurs entrées.

- ✓ **Scope** : permet l'affichage des signaux générés par une simulation dans une fenêtre spécifique différente des fenêtres Matlab .On peut changer les paramètres tels que l'échelle des temps, des ordonnées, le nombre de points à afficher par courbe. On peut zoomer, sauvegarder, imprimer.
- ✓ **XY Graph** : permet le tracé de deux signaux en mode XY(deux entrées de type scalaire).

III.2.3 Bibliothèque Continuous :

- ✓ **Transfer Fcn** : Simule la fonction de transfert d'un système à temps continu. Le numérateur et le dénominateur sont entrés par l'utilisateur au niveau de la boîte de dialogue du bloc (entrés dans l'ordre décroissant des puissances de la variable de Laplace).
- ✓ **State-Space** : Simule le comportement d'un système à temps continu représenté dans l'espace d'état.
- ✓ **Integrator** : Simule la fonction de transfert d'un intégrateur pur.
- ✓ **Derivative** : Simule la fonction de transfert d'un dérivateur pur.
- ✓ **Transport Delay** : Simule un retard pur.

III.2.4 Bibliothèque Signal et Systems :

- ✓ **Mux** : permet de passer de plusieurs entrées (scalaires ou vectorielles) à une sortie unique vectorielle.
- ✓ **Demux** : réalise l'opération inverse d'un Mux, sépare un vecteur en différents sous-secteurs ou même en scalaires.
- ✓ **In1** : insert un port d'entrée.

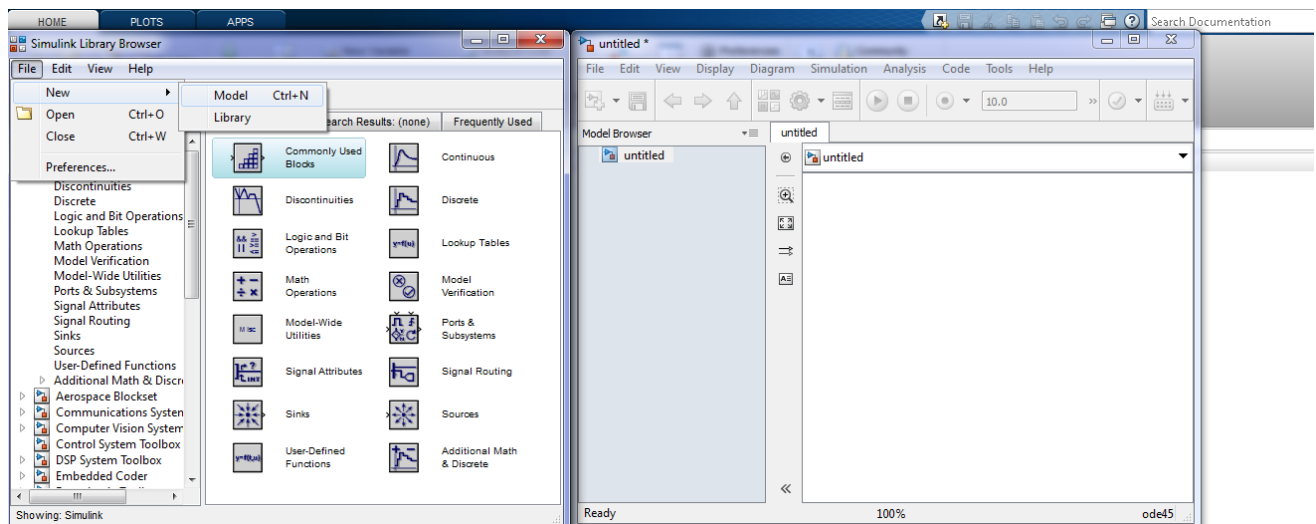
- ✓ **Out1** : insert un port de sortie.

II.2.5 Bibliothèque Math Opérations :

Les blocs réalisent une fonction mathématique appliquée aux signaux entrants.

- ✓ **Abs** : la sortie de ce bloc est la valeur absolue de l'entrée.
- ✓ **Gain** : la sortie est un signal d'entrée multiplié par un gain (scalaire ou vectoriel) entré par l'utilisateur.
- ✓ **Sum** : la sortie est la somme des entrées associées à un signe + à laquelle on soustrait les entrées associées à un signe - .
- ✓ **Sign** : donne le signe de l'entrée : si 1 \rightarrow entrée > 0
si 0 \rightarrow entrée = 0
si -1 \rightarrow entrée < 0

En cliquant sur File New Model, on ouvre une fenêtre de travail Untitled (sans titre) pour composer le nouveau schéma.



Pour copier un bloc d'une bibliothèque Simulink dans la fenêtre de travail, on le sélectionne en cliquant dessus avec le bouton gauche de la souris. Tout en maintenant le bouton gauche enfoncé, on se déplace avec la souris jusqu'à la fenêtre de travail Simulink. On relâche le

bouton de la souris à l'endroit où l'on souhaite positionner son bloc. Le bloc se retrouve ainsi dupliqué dans la fenêtre de travail.

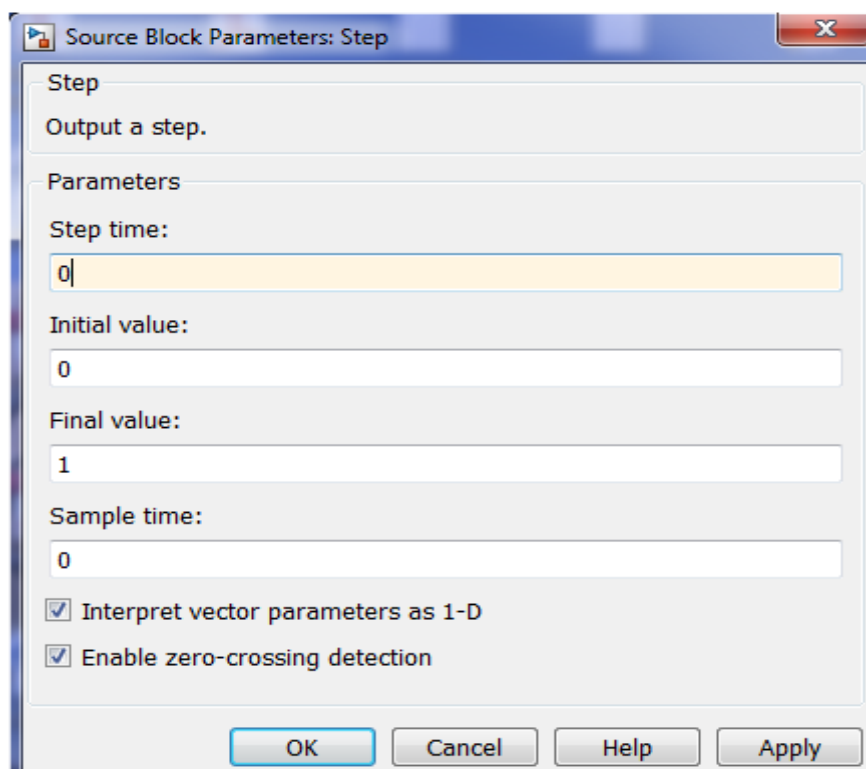
EXEMPLE 15 :

On souhaite analyser la réponse indicielle d'un système du premier ordre : $F(p) = \frac{k}{1+\tau p}$ avec $k = 5$ et $\tau = 2s$.

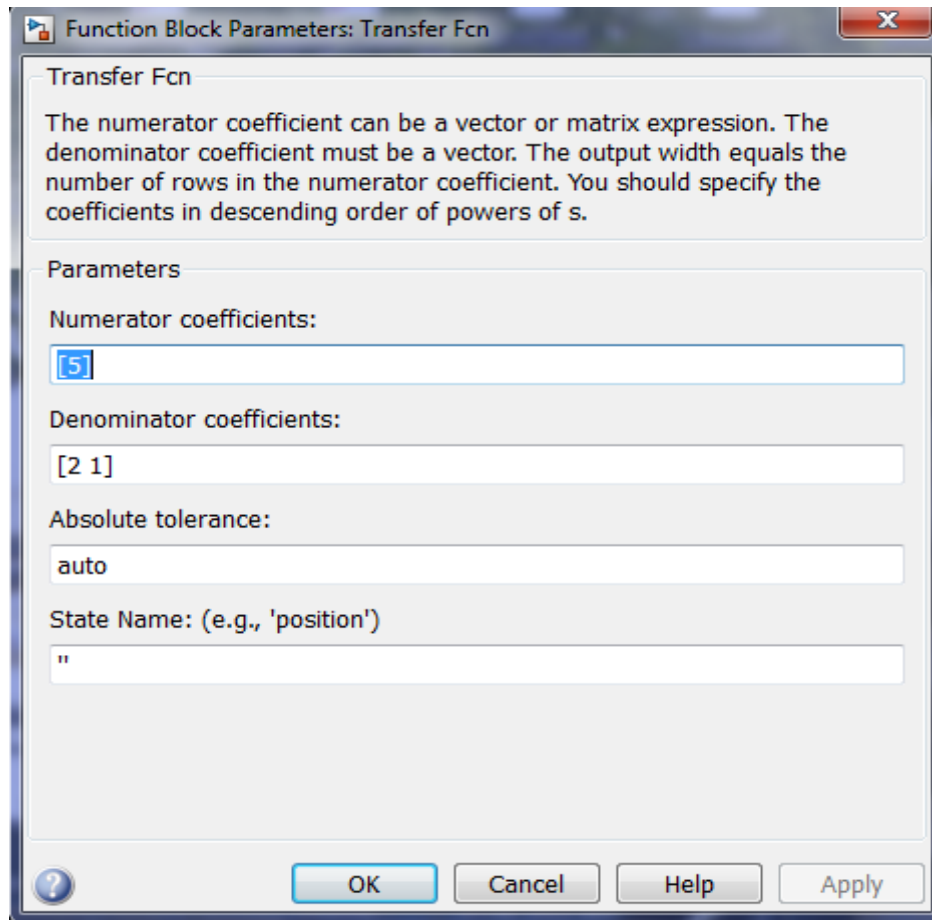
Pour paramétrer un bloc Simulink, on procède toujours de la même façon : on ouvre la boîte de dialogue du bloc en double cliquant sur le bloc concerné puis on renseigne les différents champs de la boîte de dialogue.

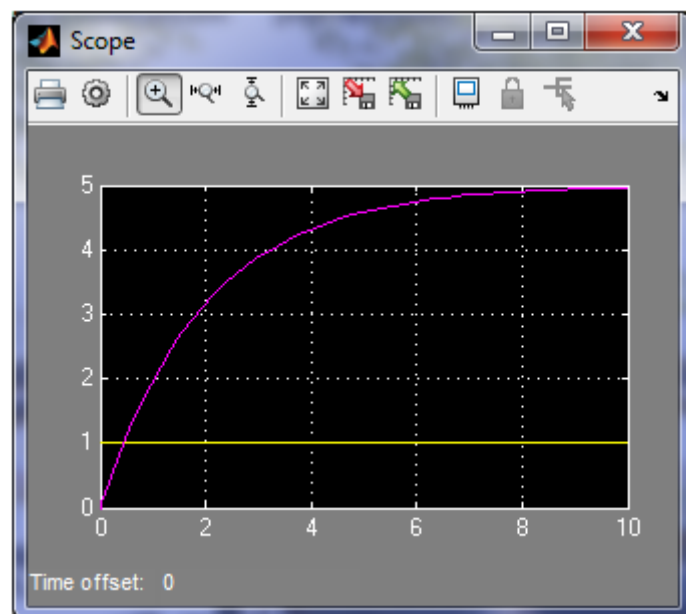
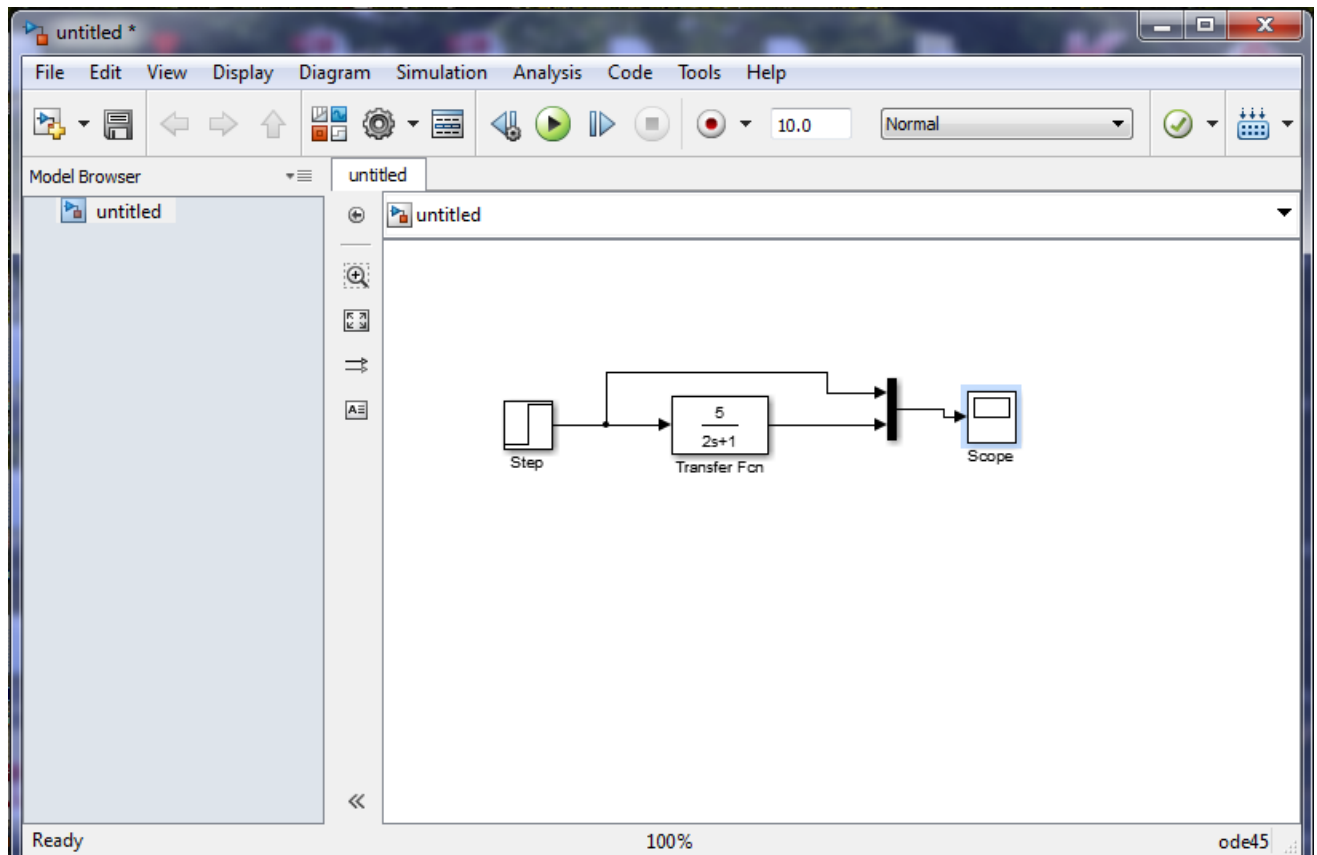
Le bloc Step : il y a quatre lignes à modifier.

Pour un échelon unité continu à partir de l'origine des temps, on aura :



Le bloc Trans Func : on déclare le numérateur et dénominateur.





CC2

TP 1 Mise à Niveau pour l'exploitation des boîtes à outils de Matlab

1. **Objectif :** L'objectif de ce TP est de familiariser les étudiants à l'utilisation du logiciel Matlab-Simulink et la réalisation de programmes Matlab pour la simulation des systèmes asservis.
2. **Introduction :** Comme entrevu, le logiciel **Matlab**: Matrix Laboratory est un langage de calcul mathématique basé sur la manipulation de variables matricielles.

Simulink est un outil additionnel qui permet de faire la programmation graphique en faisant appel à des bibliothèques classées par catégories.

Prise en main du logiciel :

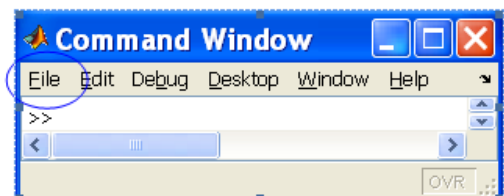
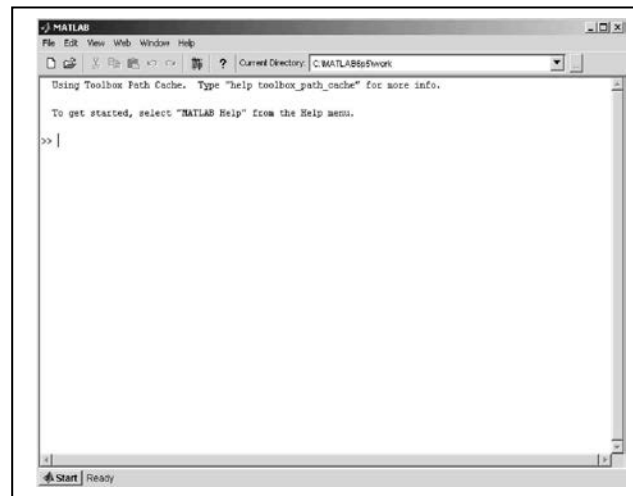
Cliquer deux fois sur l'icône MATLAB sur le bureau. Quand MATLAB s'ouvre, une fenêtre apparaît, c'est la fenêtre de commande.

L'environnement MATLAB se présente sous la forme d'un espace de travail: tout mot, tapé à la suite de l'invite (`>>`) dans la fenêtre de commande, et terminé par l'appui sur touche **ENTREE** (validation), est interprété comme étant une commande MATLAB. Si sa syntaxe est valide, la commande sera immédiatement exécutée, sinon un message d'erreur sera délivré.

Création de programme dans un fichier :

Cliquer sur « File », ensuite dans le menu déroulant cliquer sur « New » « M-File ». Ceci fait apparaître une nouvelle fenêtre ('Editor') dans laquelle on éditera le texte du programme.

*La commande **help** permet d'obtenir de l'aide sur un sujet, une instruction ou une fonction. On tape help suivi par le sujet.



Exemple : Essayer les instructions suivantes :

help sin

help plot

*La commande **clc** permet d'effacer (nettoyer) l'écran.

Traçage de courbes : On utilise l'instruction **plot** pour tracer un graphique 2D. **plot(x,y) :** permet de tracer y en fonction de x : $y=f(x)$ avec x et y deux vecteurs de même longueur.

On désire par exemple représenter la fonction $f(t)=t$, pour $0 \leq t \leq 6\pi$ avec un pas de $\pi/10$.

Introduire alors la commande suivante :

```
t=[0 :pi/10: 6*pi] ; f = t ; plot(t , f)
```

On peut ajouter à plot un troisième paramètre pour indiquer la couleur et le type de tracé (continu ou discontinu) de la courbe (Voir tableau 1).

	Couleur	Marqueur		Style du tracé	
y	jaune	.	point	- (*)	ligne continue
m	magenta	o	cercle	:	ligne pointillée
c	cyan	x	x	- -	tirets
r	red	+	plus	-.	tiret point
g	vert	*	étoiles		
b	bleu	s	carrés		
w	blanc	d	losanges		
k	noir (*)	<, >	triangles		

Tableau 1

REMARQUE : Les valeurs notées (*) sont les valeurs par défaut.

On peut ajouter à la courbe obtenue les identifiants suivants :

- Le titre de la figure : `title(' ')` ;
- Le titre de l'axe des abscisses : `xlabel('x')` ;
- Le titre de l'axe des ordonnées : `ylabel('y')` ;

Un quadrillage (grille), réalisé par la commande **grid**, donne plus de lisibilité au graphique.

La commande **grid off** supprime le quadrillage du graphique courant.

REMARQUE :

- On peut faire appel à l'éditeur MATLAB en suivant le chemin suivant : **File/New/M-file.**
- On peut également accéder à un fichier déjà enregistré pour le modifier, en suivant : **File/Open/Nom du fichier.**

Exercice 1 :

Soit la fonction suivante :

$y = \sin(t)$, t : le temps.

Ecrire un programme sous Matlab pour tracer cette fonction pour $0 \leq t \leq 2\pi$ avec un pas de $\pi/100$ et en indiquant sur la figure : le titre (fonction sinusoïdale), (temps en s) sur l'axe «X » et (amplitude) sur l'axe « Y ».

Mettre la grille sur la figure.

Exercice 2 :

Ecrire l'équation suivante sous Matlab :

$y(t) = 5t^4 + 5t^2 + 8t - 1$; t : étant le temps

Calculer les racines de $y(t)$.

Calculer $y(-1), y(1), y(2)$.

Calculer $dy(t)/dt$

Calculer $z(t) = y(t).x(t)$ avec $x(t) = 9t^2 + t + 5$

Exercice 3 :

Ecrire sous Matlab, les fonctions de transfert suivantes selon 2 méthodes :

$$F1(p) = \frac{1}{p + 6};$$

$$F2(p) = \frac{p + 2}{p(p - 5)};$$

$$F3(p) = 7 \frac{(p + 9)}{p^2 + 2p + 5};$$

$$F4(p) = 5 \frac{(p - 7)}{p^2 + 0,99p + 0,4}$$

Exercice 4 :

Ecrire un programme sous Matlab pour trouver les transformées de Laplace des fonctions suivantes :

$$e^{-at}, \quad t^7, \quad \sin wt, \quad \cos wt.$$

Exercice 5 :

Ecrire un programme sous Matlab, pour trouver les transformées inverses de :

$$y1(p) = \frac{3p+1}{p^2+8p+3}, \quad y2(p) = \frac{5}{p(p^2+w^2)}, \quad y3(p) = e^{-\pi p}, \quad y4(p) = \ln\left(5 + \frac{w^2}{p^2}\right).$$

w constante.

✓ TRAVAIL SOUS SIMULINK

1. PRESENTATION DE SIMULINK :

Comme précité, SIMULINK est une extension du logiciel Matlab. Simulink est l'interface graphique de MATLAB qui permet de s'affranchir du code et de la syntaxe pour la saisie des lignes de commandes MATLAB. C'est un logiciel de simulation de systèmes variables dynamiques muni d'une interface graphique qui facilitera la saisie du modèle, la simulation du modèle.

Afin de voir comment utiliser SIMULINK, il faut lancer en premier lieu, le logiciel MATLAB. Taper la commande **simulink** dans la fenêtre de commande MATLAB sera la prochaine étape.

La fenêtre principale va être affichée. On sélectionne **New...Model** dans le menu **File**. Une fenêtre s'ouvre avec un choix de schémas-blocs.

Selon notre conception voulue, on ramène les blocs désirés à partir de la fenêtre principale de Simulink. Elle comporte les différentes bibliothèques.

Les blocs choisis sont insérés dans une fenêtre de travail par « copier-coller » ou en glissant le bloc d'une fenêtre à une autre, grâce à la souris.

Afin de connecter deux blocs, on pointe avec la souris sur la pointe du bloc et on enfonce le bouton droit de la souris. Un trait se verra. On pointe ensuite sur la face gauche du second bloc au niveau d'une flèche, et on relâche le bouton de la souris. Un trait entre les deux blocs apparaîtra montrant la réussite de la connexion.

2. Déroulement de la simulation

On définit les paramètres de la simulation qui porte sur l'instant du début de simulation, l'instant de fin de simulation, la période de simulation, la méthode d'intégration numérique

utilisée... On va aller dans le menu « **Simulation** » puis sur « **parameters** » et spécifier les paramètres relatifs à la simulation. Ce choix est important car les résultats vont en dépendre.

Dans le menu « **Simulation** » on sélectionne «**start**». Pour l'utilisation des blocs de visualisation (graph, scope, ...), on paramètre ceux-ci en fonction des paramètres de la simulation.

Pour interrompre la simulation en cours, on va dans le menu « **Simulation** » et on clique sur « **stop** ».

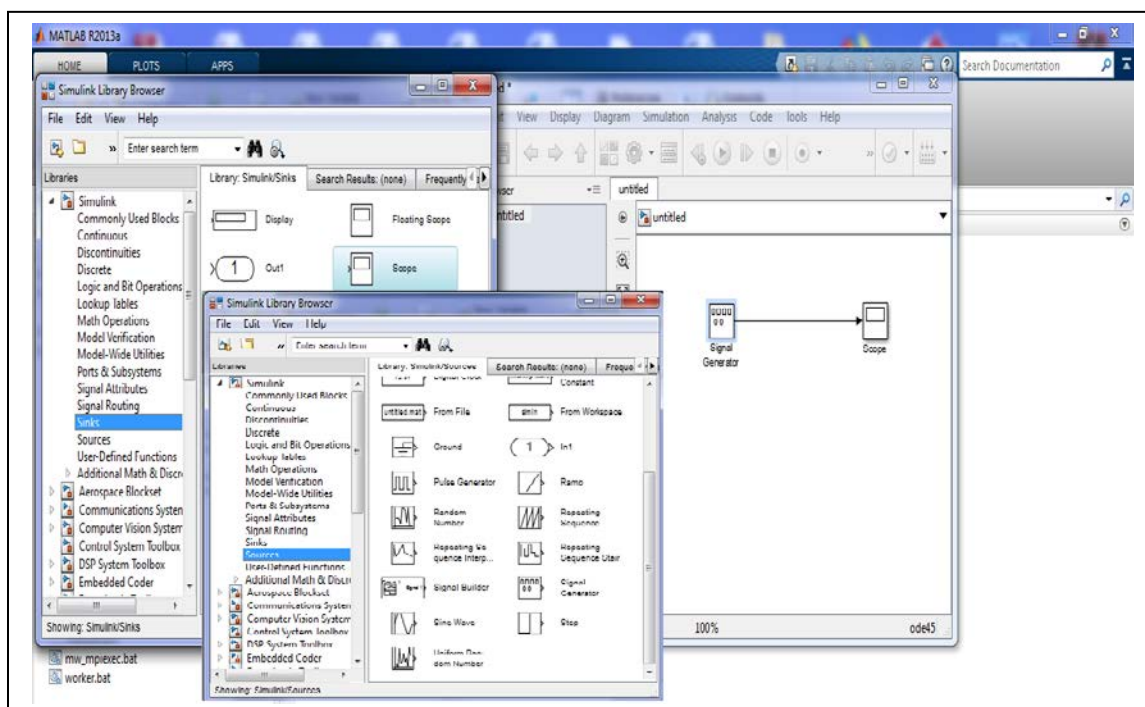
Exemple:

Ici, c'est la SIMULATION D'UN SIGNAL SINUSOIDAL

On vise sa variation temporelle.

Donc, on lance Simulink. On sélectionne **New...Model** dans le menu **File**. Une fenêtre de travail va s'ouvrir. On y insérera les schémas-blocs.

On ouvre la bibliothèque **Sources**, on sélectionne l'icône «**Signal generator**» en «cliquant» une fois dessus. On fait glisser ceci dans la fenêtre de travail. On ouvre **Sinks** et on sélectionne l'**oscilloscope**. On fait glisser ce dernier dans la fenêtre de travail. A l'aide de la souris, on relie la sortie du bloc générateur de signal, à l'entrée de l'oscilloscope. L'oscilloscope permet de visualiser une partie du signal à l'écran.



On souhaite générer une sinusoïde d'amplitude 5 V, de fréquence 2 Hz et de phase nulle à l'origine.

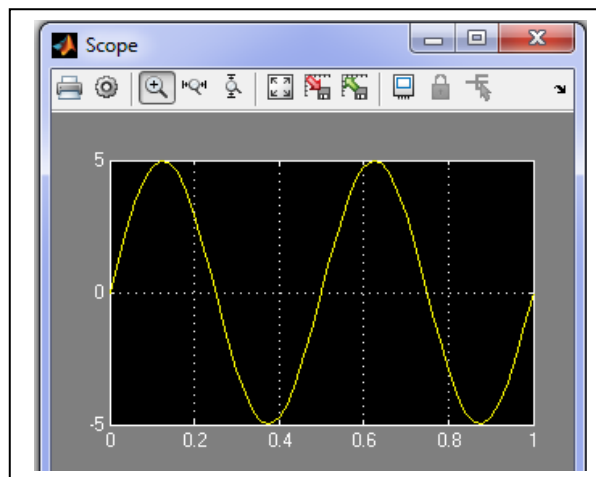
On configure les différents blocs en cliquant deux fois sur chacun d'eux:

a) le bloc « signal generator » permet de définir la pulsation du signal en rad/s ou en Hz pour la fréquence.

b) On configure l'oscilloscope.

c) On configure les paramètres de simulation, en particulier la date de début (souvent 0) et la date de fin (1 sec par exemple), dans la case blanche en dessous du menu Help.

La phase de paramétrage est donc achevée. On lance la simulation à l'aide de la commande **Start** du menu **Simulation**. On peut cliquer sur le bouton **Run**. Le signal est obtenu sur l'oscilloscope (les axes peuvent être vérifiés). Voilà ce qu'on obtient :

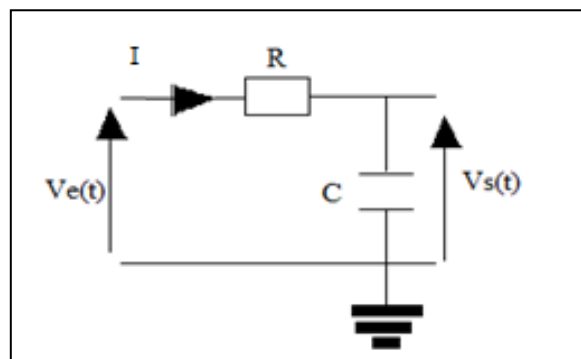


Exercice 6:

En utilisant Simulink construisez un modèle pour avoir la réponse à un échelon en boucle ouverte et en boucle fermée d'un système du premier ordre : $\frac{k}{1+\tau p}$, ayant un gain $k=5$ et une constante de temps $\tau=30s$.

Exercice 7:

Soit un circuit RC attaqué par un échelon d'amplitude 24V, avec $R=50\Omega$, $C=100\mu F$.



Le condensateur est initialement déchargé.

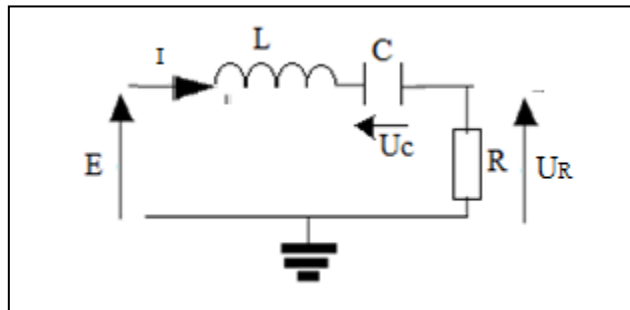
1-Etablissez le modèle simulink de ce montage.

2- Visualisez les courbes en fonction du temps ,de la tension et du courant obtenus au niveau du condensateur.

3-Etablissez l'étude théorique et interprétez les résultats obtenus.

Exercice 8:

Soit le circuit suivant, on donne : $E=24V$, $R=1.5\text{ K}\Omega$, $C=10\text{nF}$,
 $L=10^{-4}\text{H}$.



Le condensateur est initialement déchargé.

1-Etablissez le modèle Simulink du circuit.

2-Visualisez les tensions E , U_C , U_R selon le temps.

3-Faire l'étude théorique et établissez les diverses équations.