

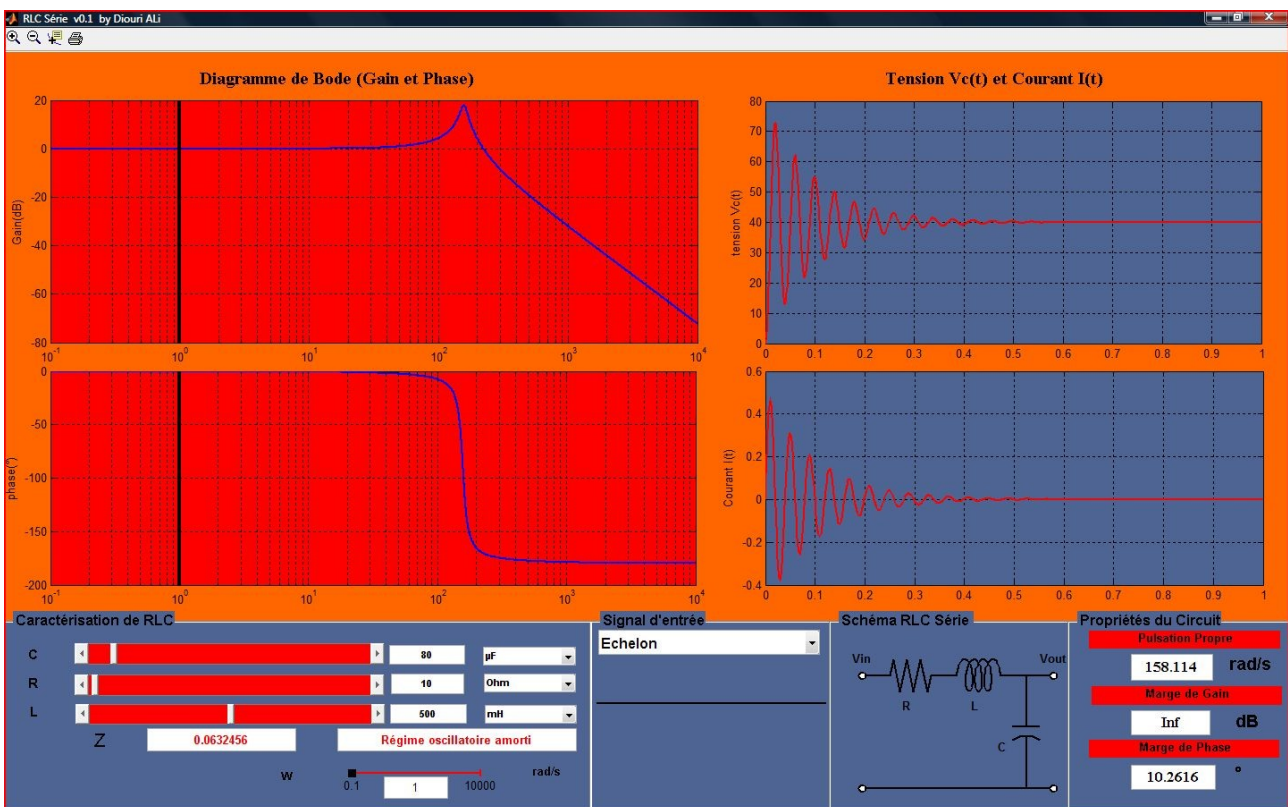
Gui Matlab

RLC Série

V 0.1

By Diouri Ali

sous licence GPL

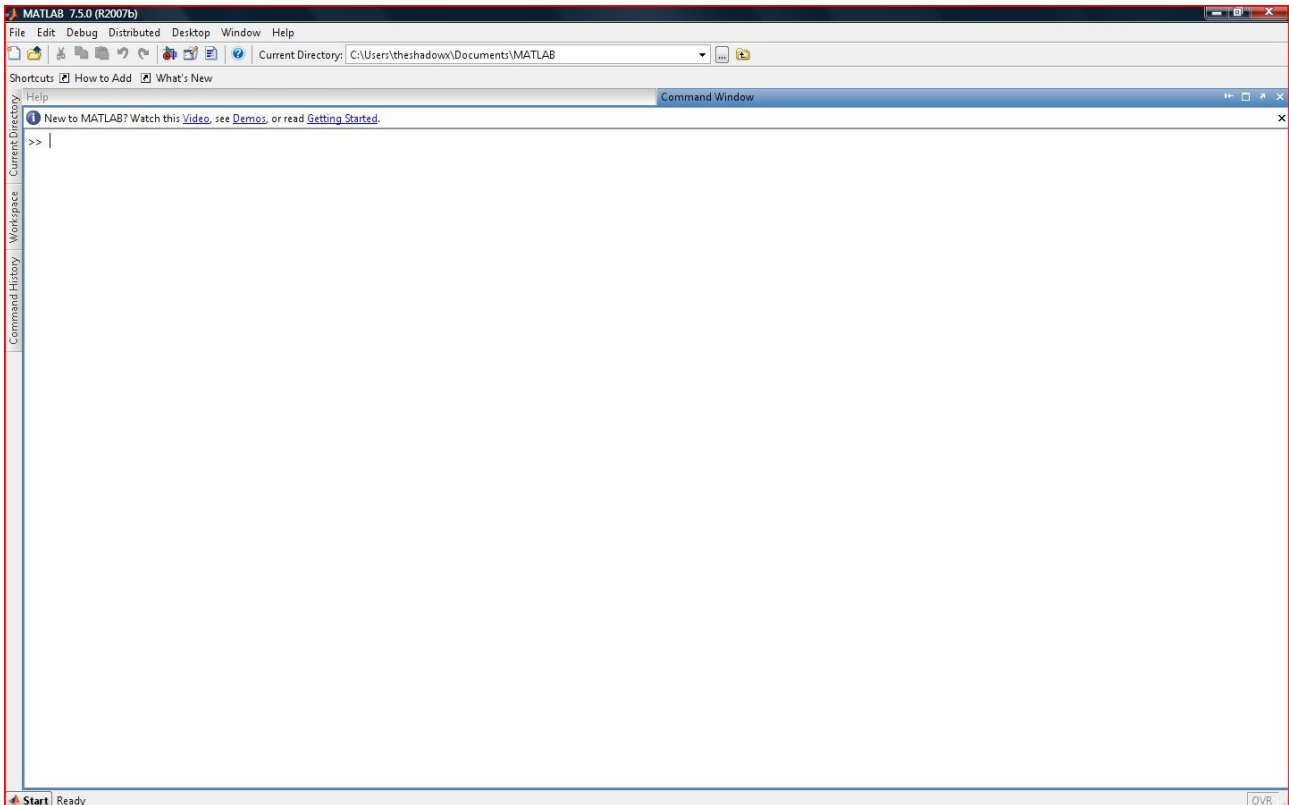


Introduction

Le circuit RLC est l'un des circuits basics, mais ses applications sont plusieurs (filtrage,...), ce qui nous mène à l'étudier pour savoir son comportement pour différentes valeurs de R (résistance), L (inductance) et C (capacité) d'une part, et d'autre part pour les différents types de signaux d'entrées (échelon, sinusoïdal, triangulaire). Matlab nous donne l'opportunité pour construire une interface qui va nous aider à étudier ce circuit et cela par le biais de Guide.

Construction de l'interface

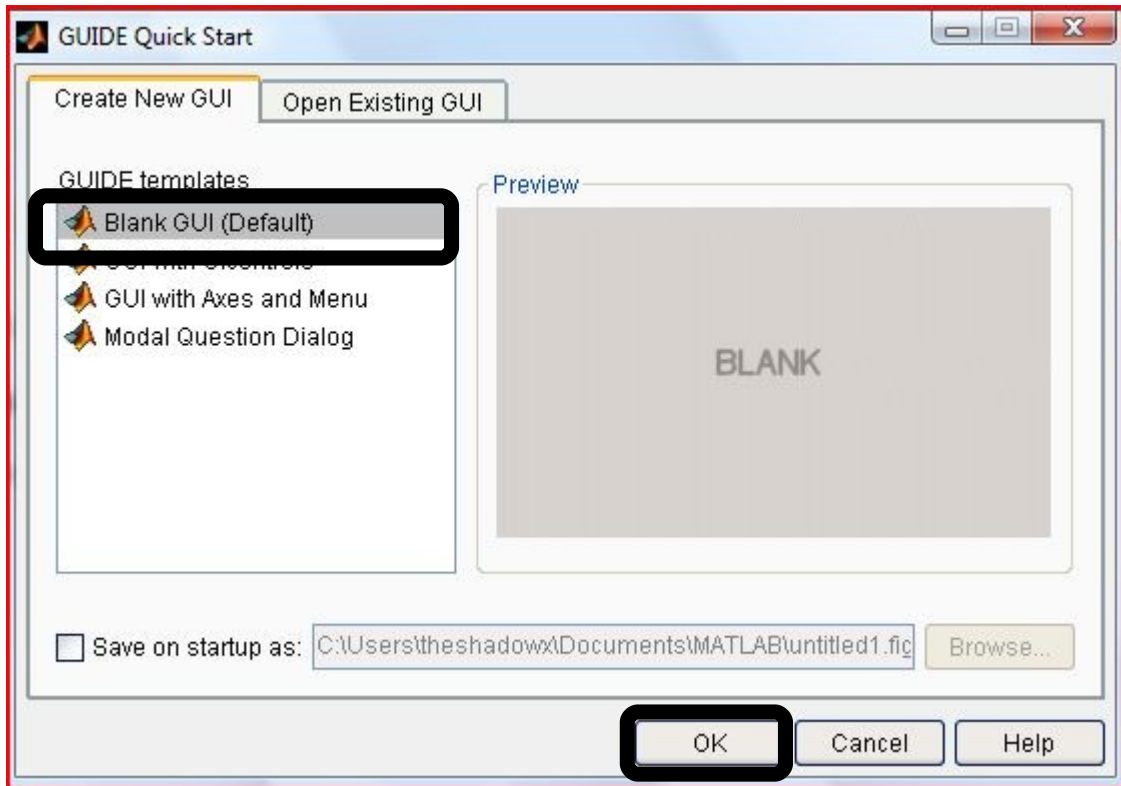
En ouvrant Matlab on obtient:



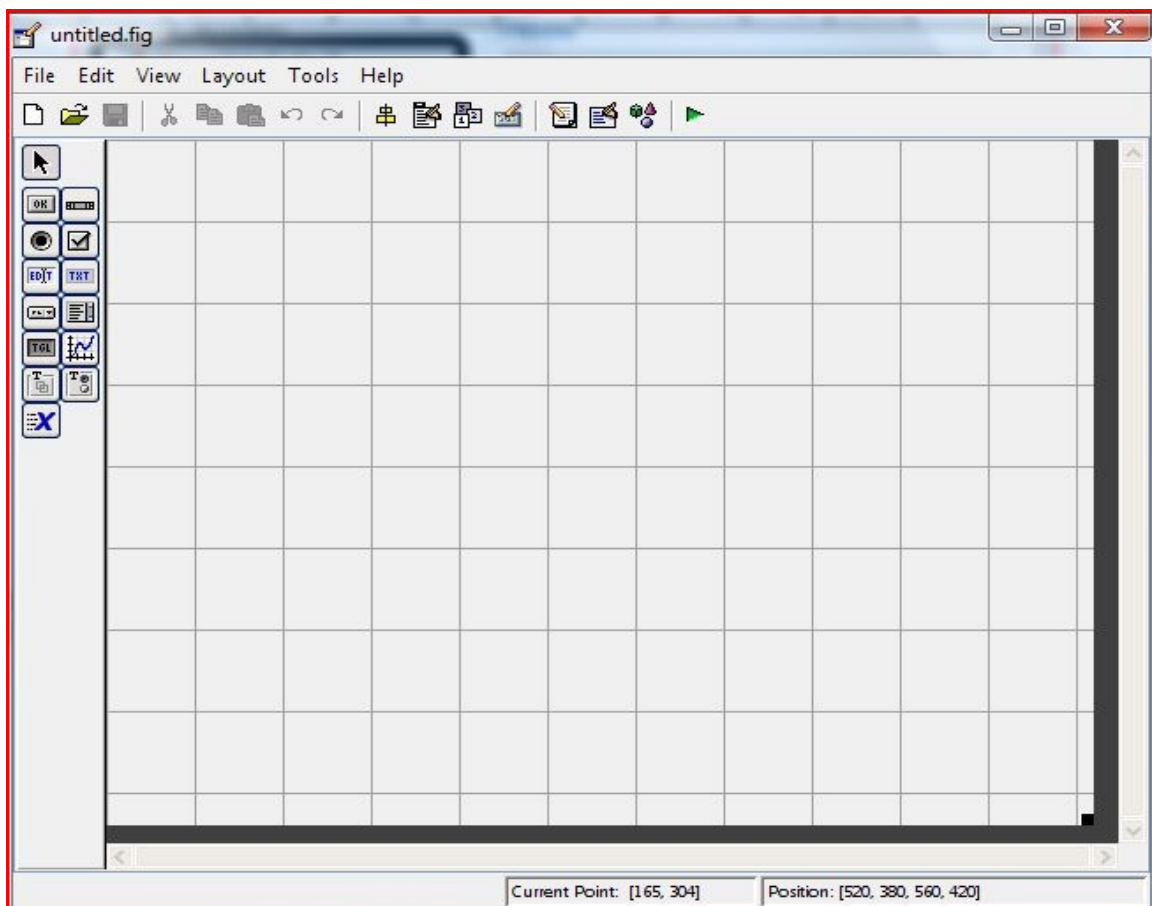
Dans la fenêtre de commande on écrit:

>> guide

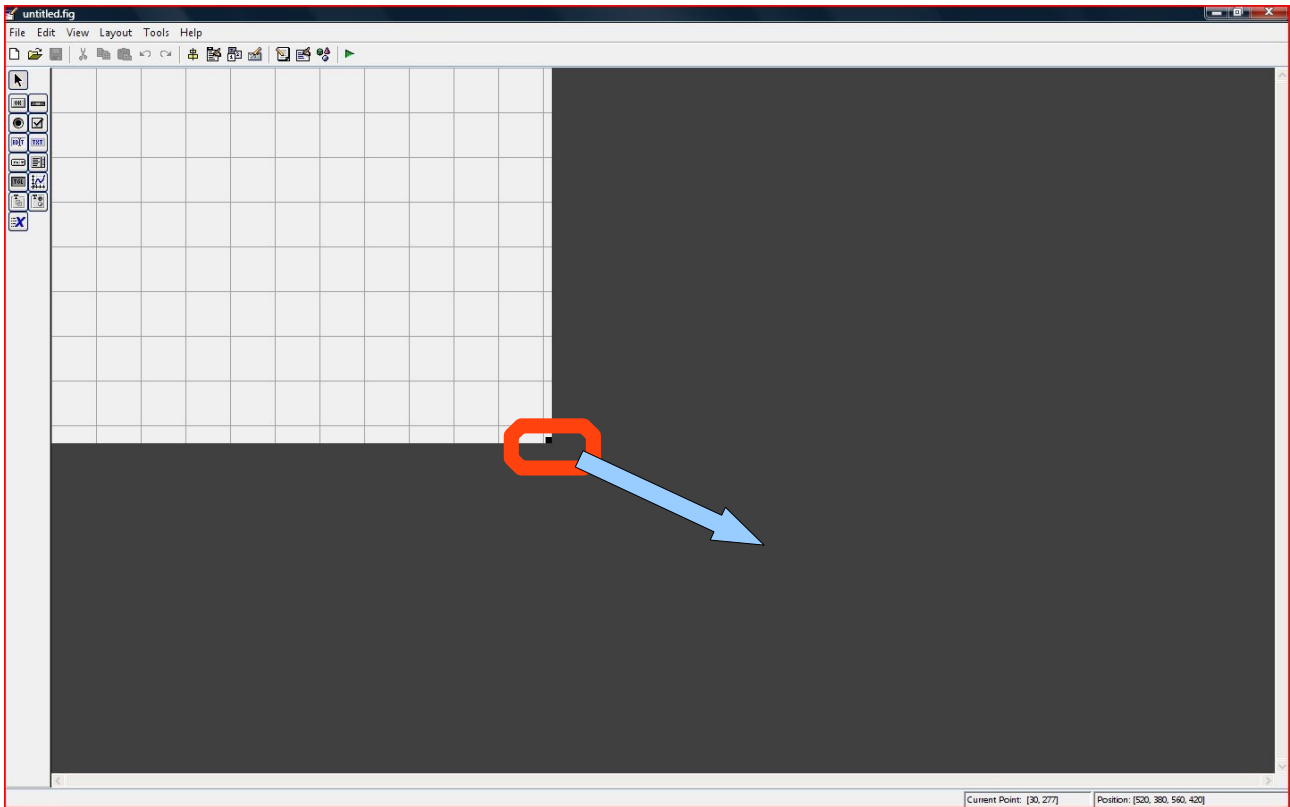
Cela va afficher une fenêtre :



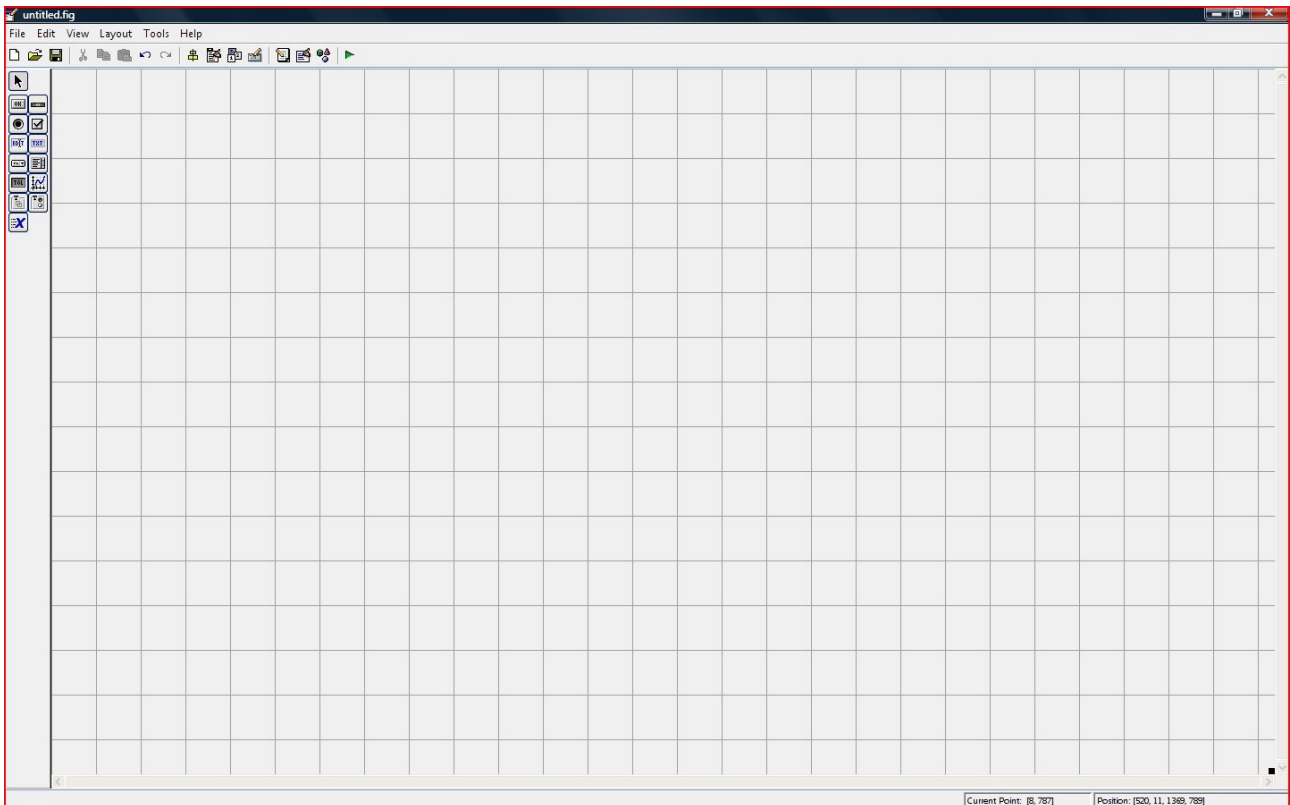
On choisit la première option « Blank GUI » et on click sur « OK »
Ce qui nous va ramener à la fenêtre suivantes:



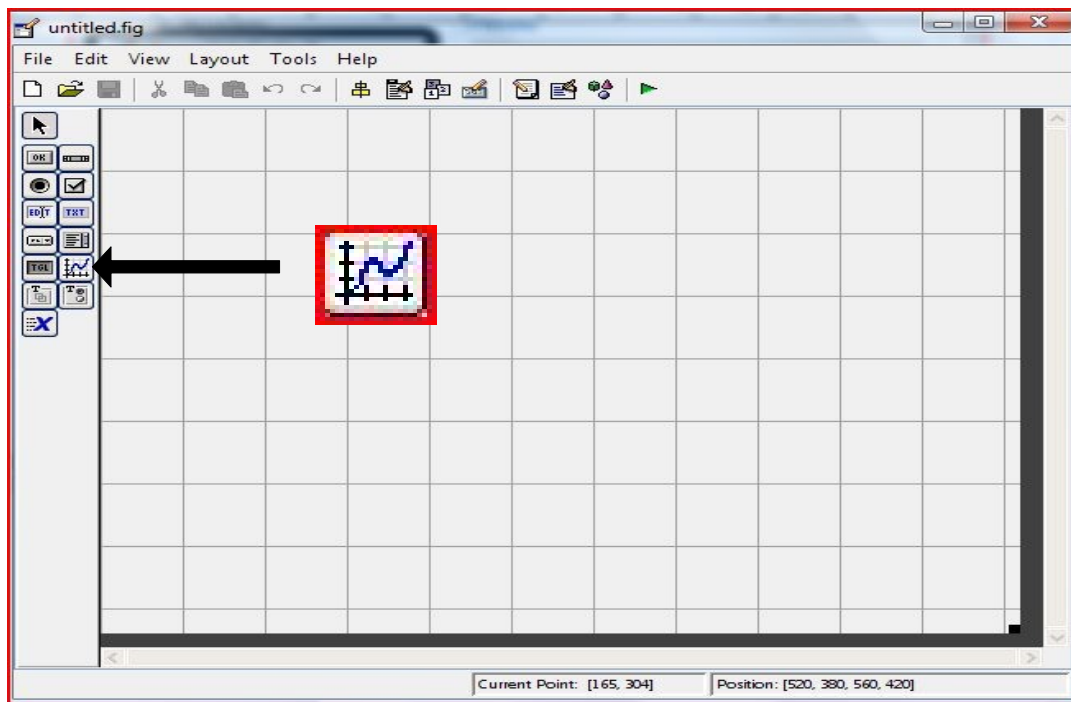
Pour que l'interface soit grande , il faut tout d'abord agrandir la fenêtre
Après on va vers le coin mentionné et on l'attire jusqu'à l'extrême/



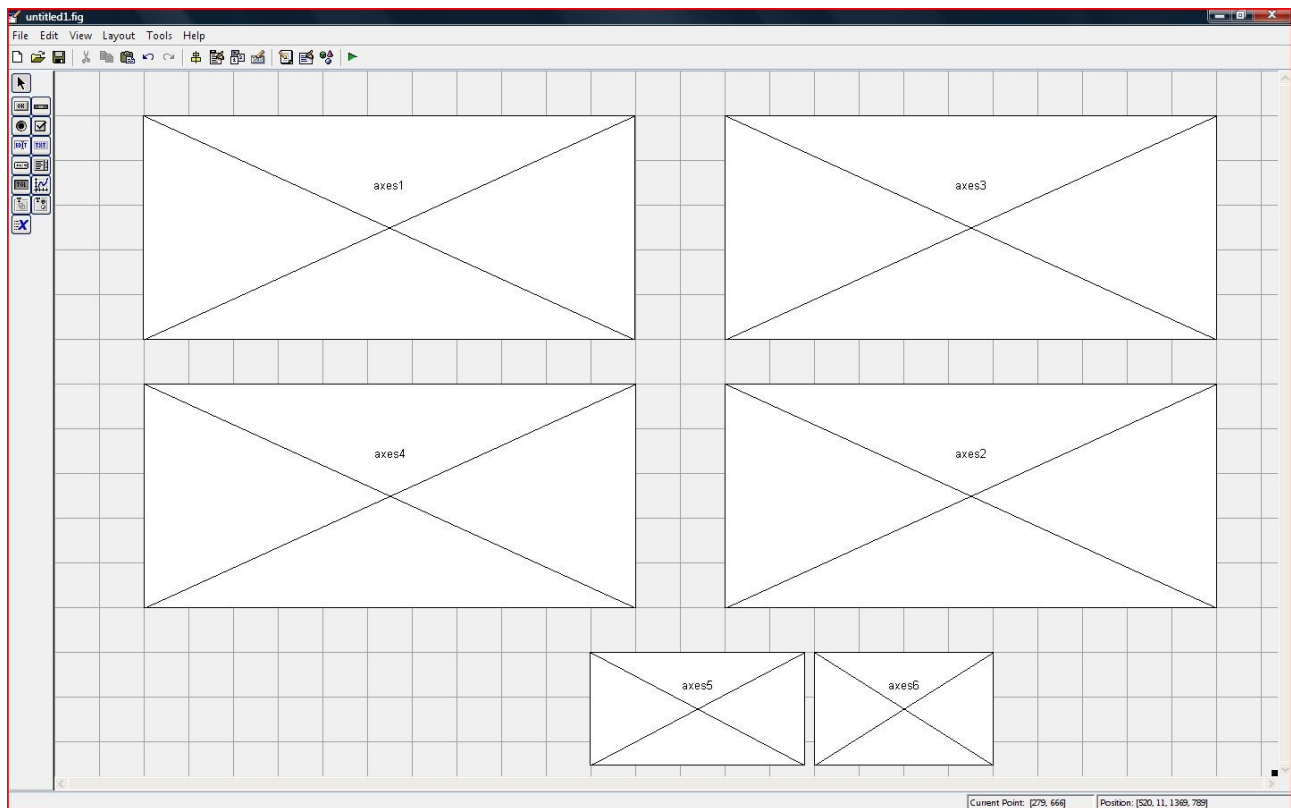
Pour obtenir ça:



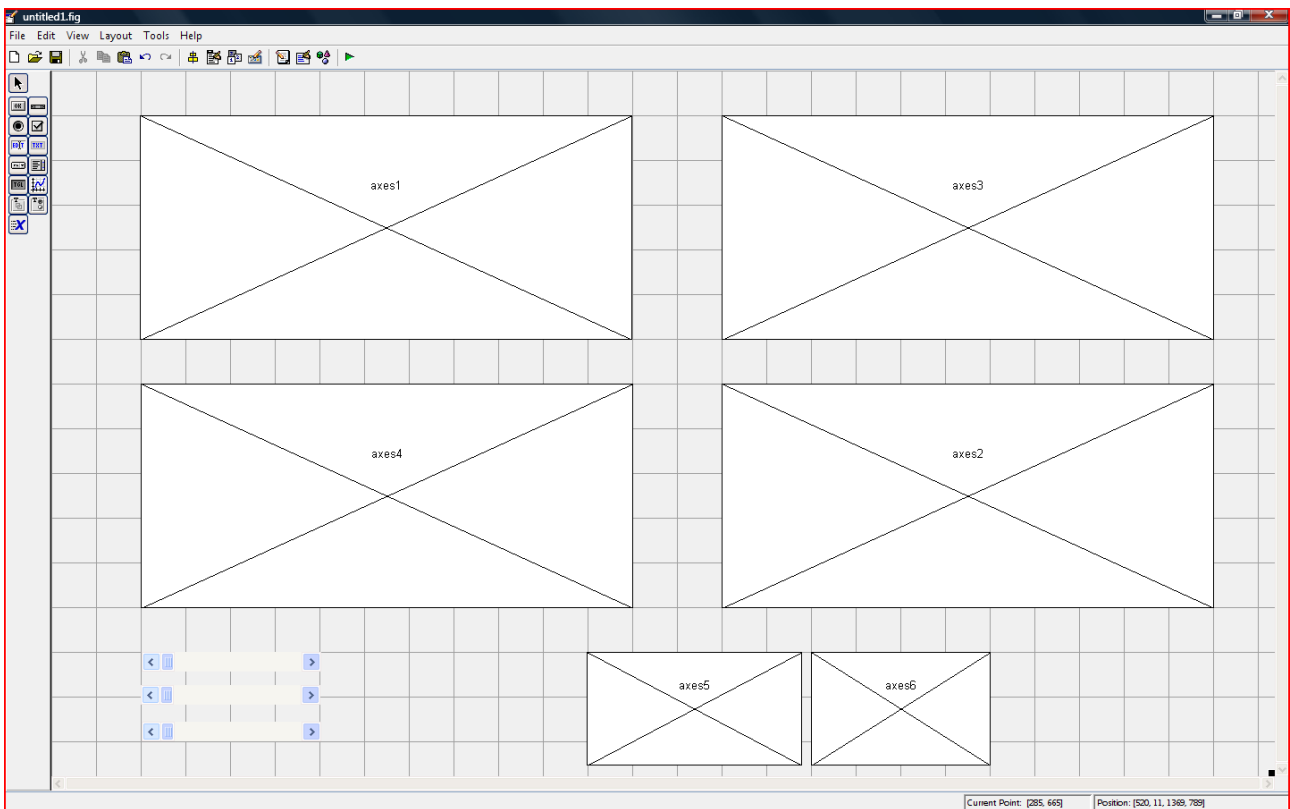
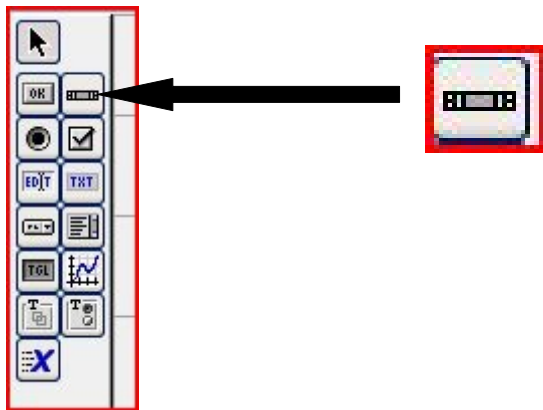
Pour ce qui concerne notre interface, on a besoin des graphes . On clic sur l'icône mentionnée:



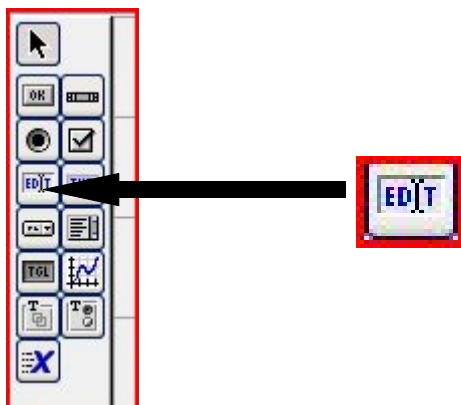
et on dimensionne nos graphes

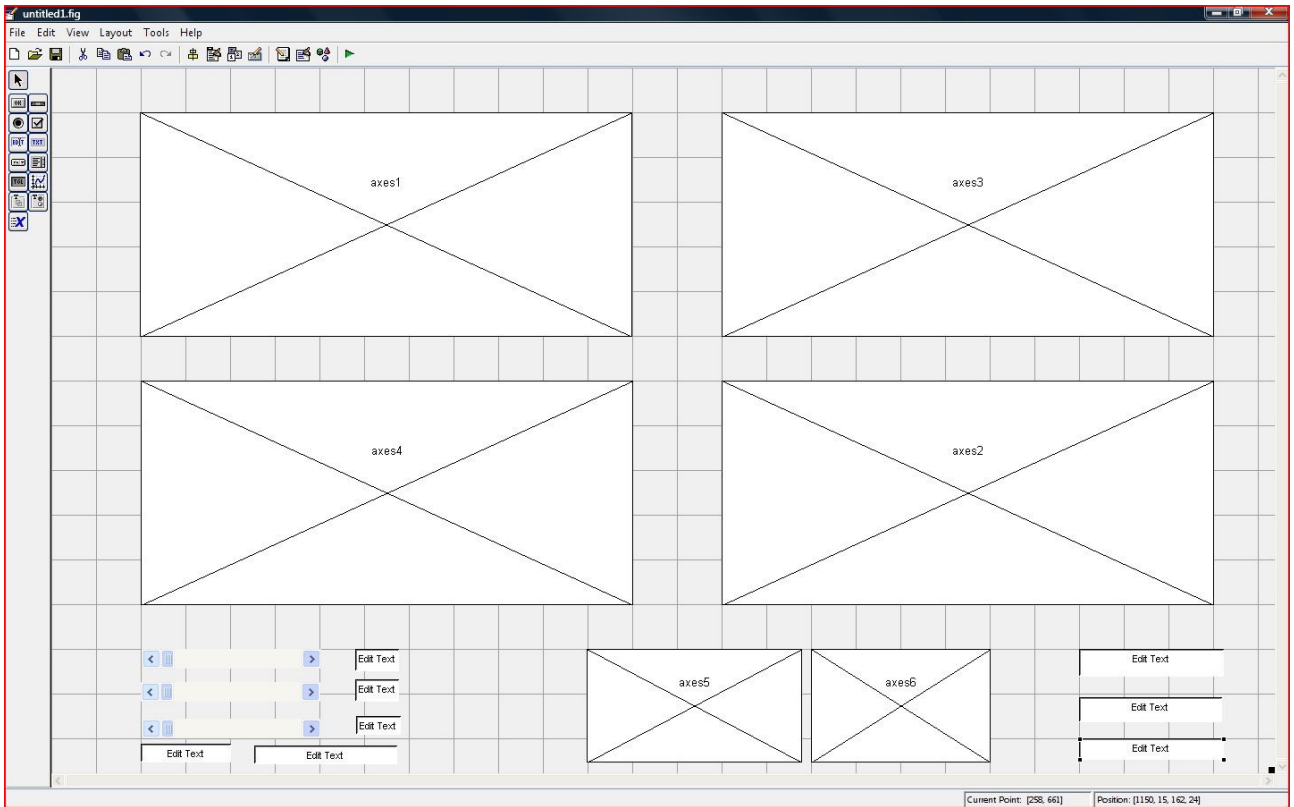


Après on a besoin des 'sliders' pour faire changer les valeurs de R L et C.

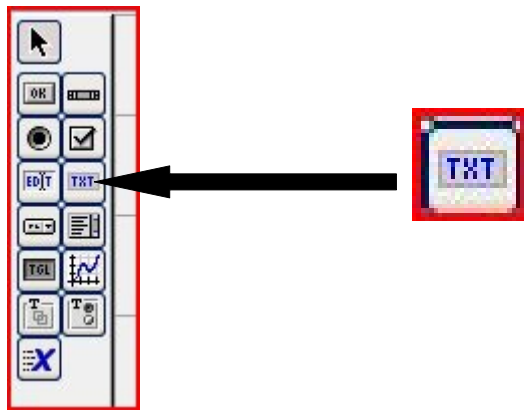


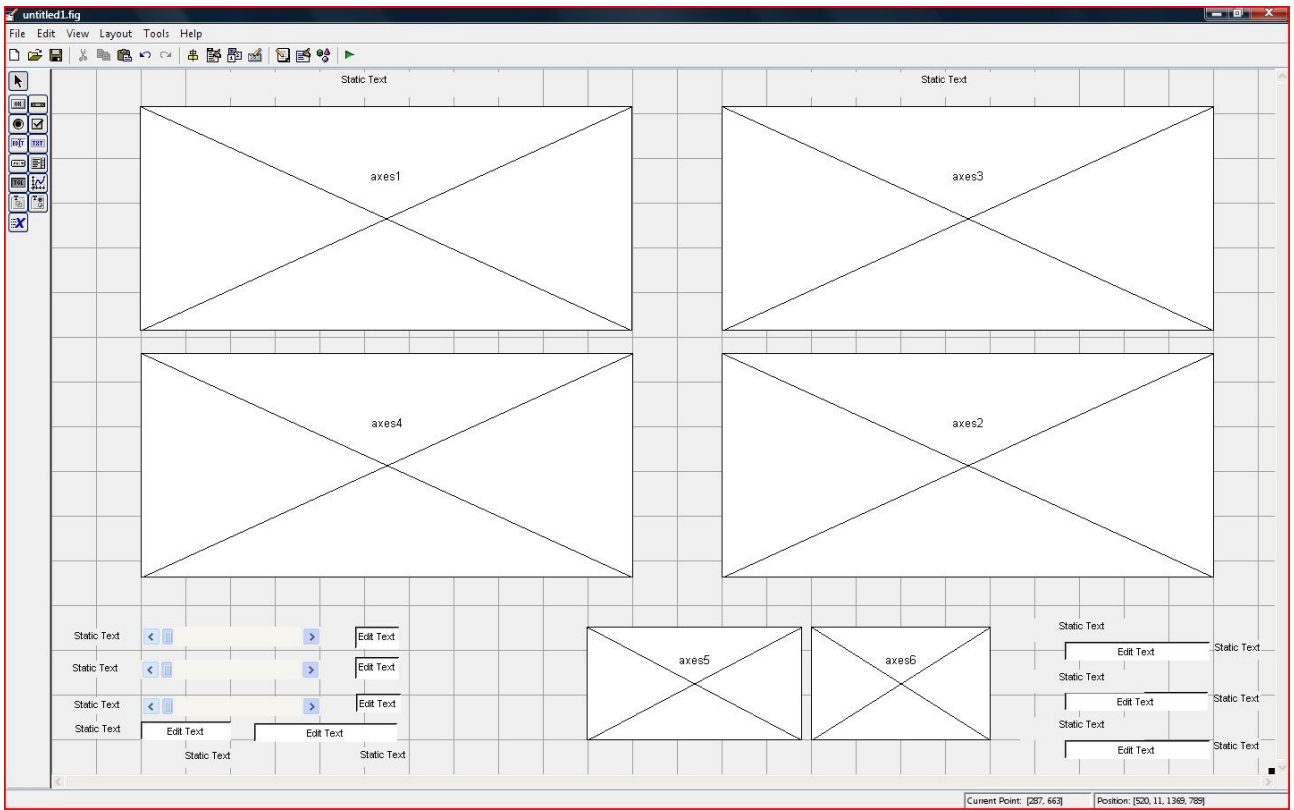
Maintenant, il faut mettre les zones ou les valeurs des sliders s'affichent



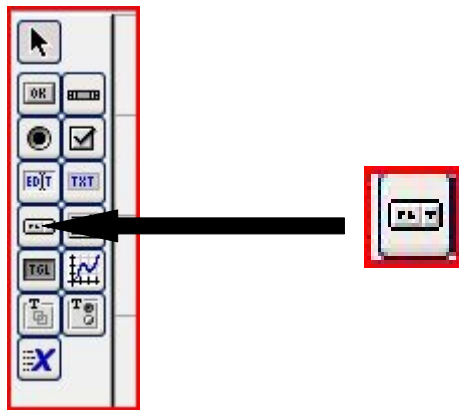


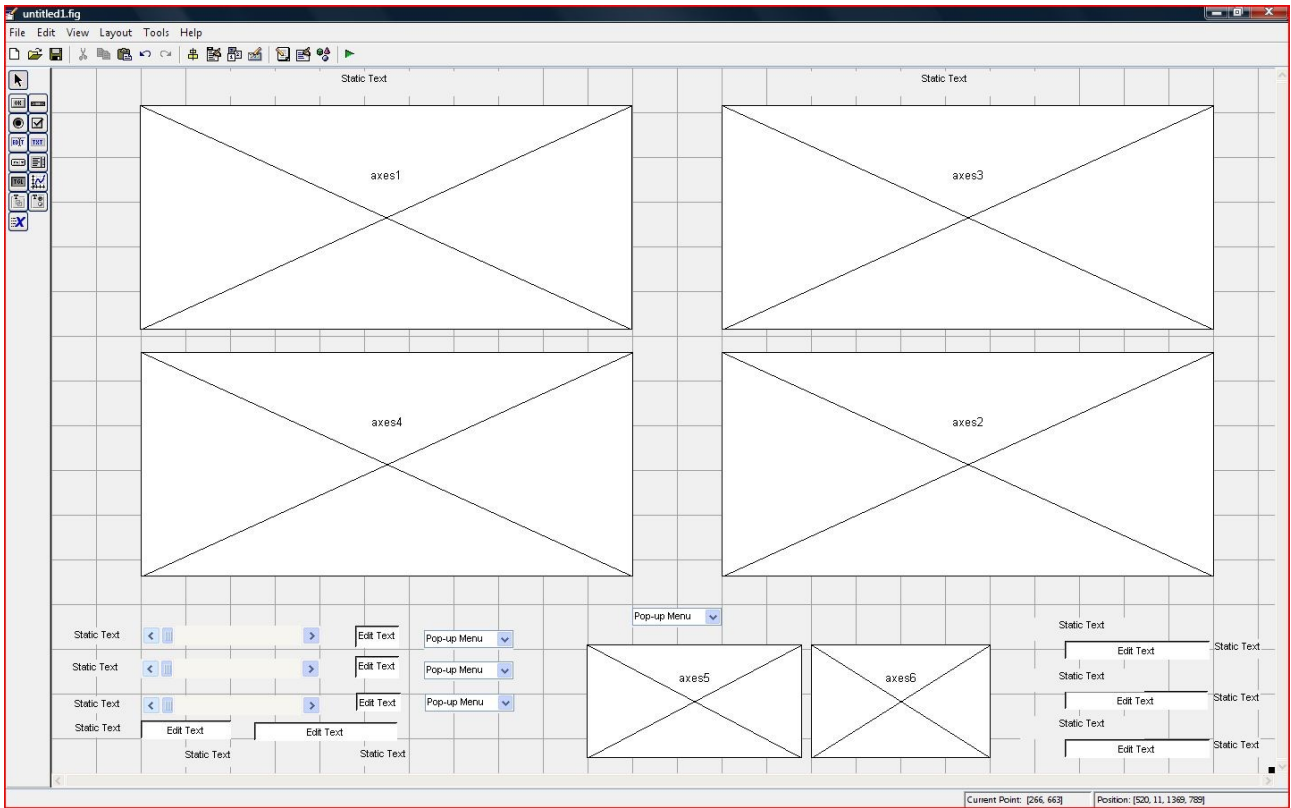
Pour donner les noms à chaque objet on utilise 'static text'



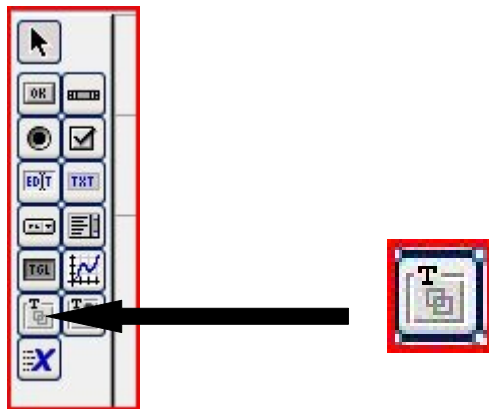


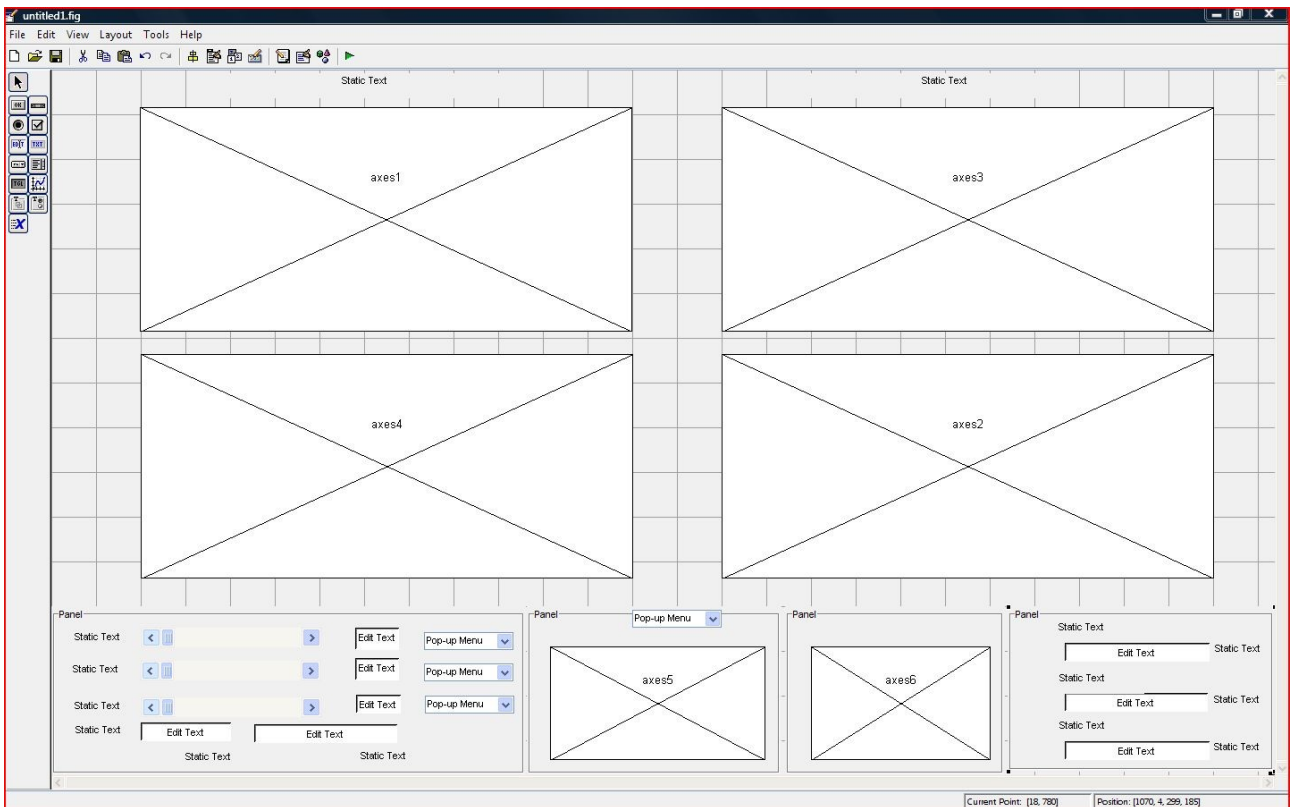
On ajoute les bars à choix 'popupmenu' pour les unités et le type du signal d'entrée





On donne à chaque groupe une catégorie en utilisant 'panel'





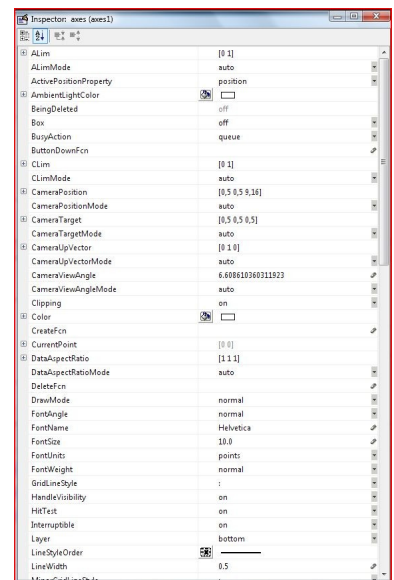
Remarque:

Quand on fait panel sur le graphe , ce dernier disparaît. Pour l'afficher on sélectionne le panel et avec un clic droit on choisit ' send to back'

Maintenant, il faut changer les propriétés de chaque chose.

On commence par les graphes: Pour réduire l'utilisation de la mémoire au démarrage de l'interface on va configurer les graphes à ce qu'ils ne soient pas visibles. On clic deux fois sur le graphe en question .

On cherche l'option 'visible' et on met 'off' au lieu de 'on'



Pour afficher les noms dans 'static text' , de même on clic deux fois sur l'objet et on va au ' String' et la bas on met le nom adéquat.

Pour donner les noms à afficher dans le 'popupmenu' On clic deux fois sur l'objet et dans la case 'String' on met la liste.

Pour ce qui concerne 'edit text' on peut donner nous même les valeurs initiales que l'interface va démarrer avec .

Pour changer les couleurs : On clic deux fois sur l'objet et dans la case 'color' on met la couleur qu'on veut.

Les sliders

il faut définir la valeur max et min et le pas et la valeur initial qu'elle doit être comprise entre min et max .

Property	Value
BackgroundColor	[Color Selector]
BeingDeleted	off
BusyAction	queue
ButtonDownFcn	
CData	[0x0 double array]
Callback	%automatic
Clipping	on
CreateFcn	%automatic
DeleteFcn	
Enable	on
Extent	[0 0 6,2 1,385]
FontAngle	normal
FontName	MS Sans Serif
FontSize	8.0
FontUnits	points
FontWeight	normal
ForegroundColor	[Color Selector]
HandleVisibility	on
HitTest	on
HorizontalAlignment	center
Interruptible	on
KeyPressFcn	
ListboxTop	1.0
Max	1.0
Min	0.0
Position	[20 4,923 40,2 1,692]
SelectionHighlight	on
SliderStep	[0,01 0,1]
String	Slider
Style	slider
Tag	slider9
TooltipString	
UIContextMenu	<None>
Units	characters
UserData	[0x0 double array]
Value	[0.0]
Visible	on

Max →

Min →

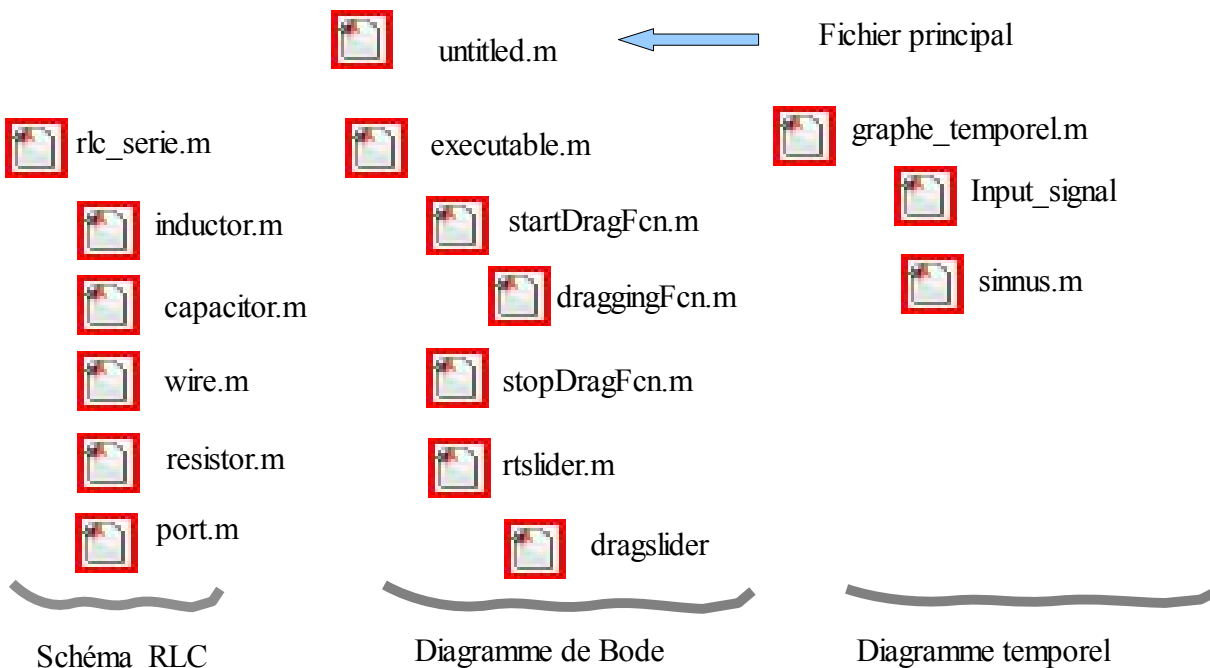
Le pas →

Valeur initiale →

Maintenant qu'on à finit avec la forme graphique on passe au codage mais avant tout il faut sauvegarder. Après le programme va générer un fichier qui portera le même nom mais avec une autre extension '.m' alors que la figure est sauvegardée sous '.fig'.

Dans le fichier créé on trouvera les fonctions Callback de chaque objet .leurs rôle est qu'à chaque modification qu'on fait il y a une réaction de la part de la fonction qui génère une action . Par exemple on fait bouger le slider est les graphes changent à chaque que je met une poussoir dans un nouvelle position .

Pour ce qui me concerne voici comment mon codage fonctionne dans l'ensemble:

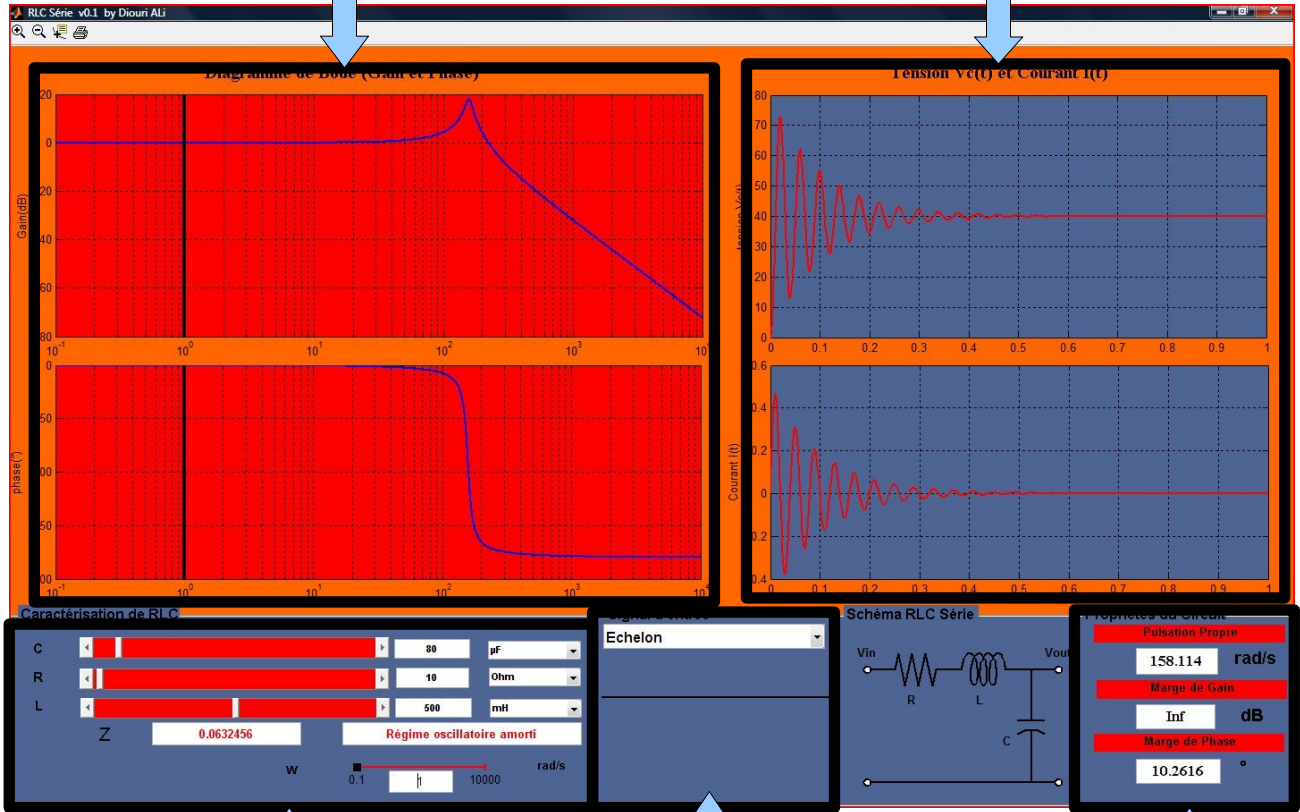


Vous trouverez des commentaires dans chaque fichier de codage.

Manipulation

Diagramme de Bode

Diagramme temporel



Caractérisation du circuit

Signal d'entrée

Propriétés du circuit

Caractérisation du circuits RLC

Type de régime établi

Caractérisation de RLC

C 80 µF

R 10 Ohm

L 500 mH

Z 0.0632456

Régime oscillatoire amorti

w 0.1 1 10000 rad/s

Coefficient d'ammortissement

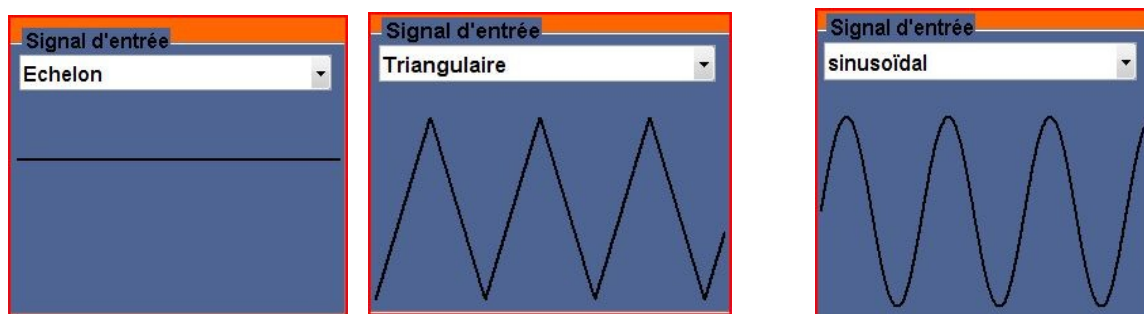
Pulsation

Pour ce qui concerne les valeurs de R, L et C

$$1 < C < 999$$
$$1 < R < 999$$
$$1 < L < 999$$

Pour modifier les valeurs de R, L et C on peut soit utiliser les sliders soit donner les valeurs dans les champs adéquats. D'autre part on peut changer l'unité des composants.

Signal d'entrée



Comme on le remarque, il y a 3 types de signaux :Échelon, Triangulaire et Sinusoïdal.

Propriétés du circuit



Ce champs donne les propriétés du circuit en montrant

- . la pulsation propre ou de coupure,
- . la marge de gain (elle est toujours infinie car on a un circuit de 2ème ordre)
- . la marge de phase

Remarque

Durant la manipulation ne faites pas varier R L et C trop vite, à chaque variation attendez que tous les graphes soient établis . Car sinon tout va se bloquer. Je suis désolé pour cet inconvénient.

Tout remarque ou commentaire est le bienvenu, merci