

MASTER PROFESSIONNEL RAIA

TRAVAUX PRATIQUES

Atelier Systèmes robotiques articulés

TPN°3 Génération d'une trajectoire d'un robot manipulateur par MATLAB

Date : ----- **Classe :** ----- **Durée : 3h**

	<i>Nom & Prénom :</i>	<i>AB/PR</i>	<i>Mot/Part</i>	<i>TP N°</i>	<i>Total</i>
1	-----		/10	/10	/20
2	-----		/10	/10	/20
3	-----		/10	/10	/20

Objectifs du TP :

- ✓ -----
- ✓ -----
- ✓ -----

Conditions de réalisation et moyens :

- ✓ -----
- ✓ -----
- ✓ -----

Objectifs :

- Générer une trajectoire d'un robot via Matlab.
- Maîtriser les outils de base pour la conception d'un robot en utilisant : Robotics System Toolbox et SimScape Toolbox de MATLAB.
-

I- Génération d'une trajectoire pour un robot planaire à 2DDL :

On veut générer une trajectoire d'un parcours en carrée pour un robot planaire à 2DDL figure N°1 et figure N°2. La trajectoire du parcours se trouve dans la zone atteignable du robot.

Les coordonnées des corners du parcours en carré sont indiqués dans le repère 2D (x, y).

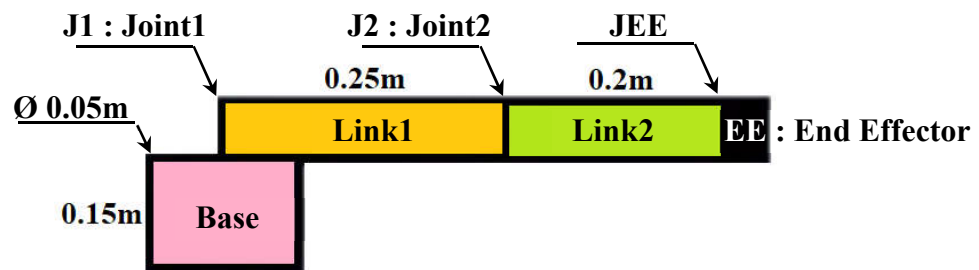
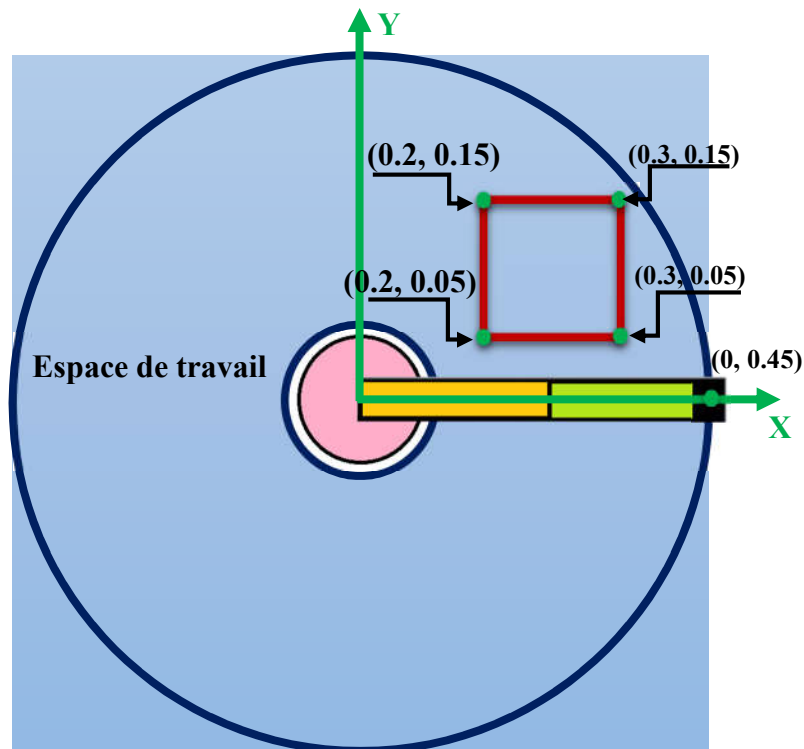


Figure 1 : Robot planaire à 2DDL



Y	X
0	0.45
0.3	0.05
0.3	0.15
0.2	0.15
0.2	0.05
0.3	0.05

Figure 2 : Coordonnées d'un parcours en carré dans l'espace de travail

- ⇒ Dans Matlab, Charger le schéma de la conception du robot (du TPN°1), puis l'exécuter.
- ⇒ Dans Matlab, lancer un nouveau Script, taper le code suivant, l'enregistrer puis l'exécuter.

```

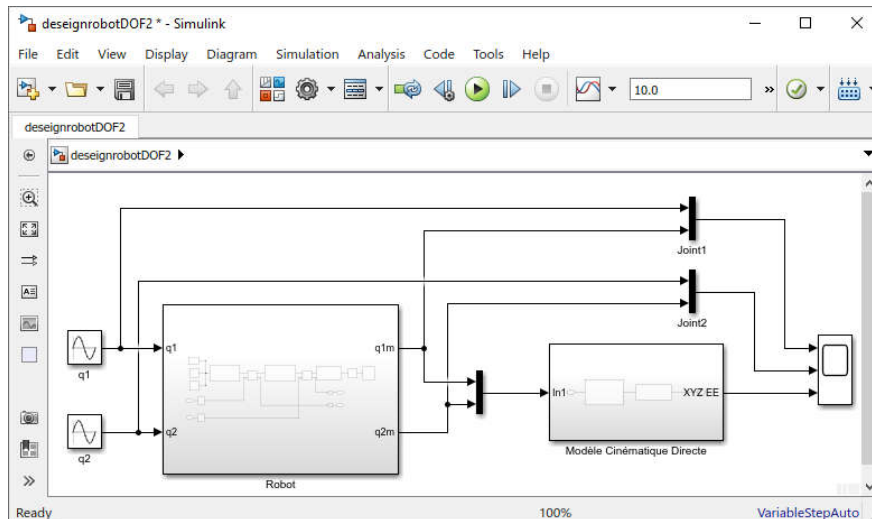
Ts=0.001;%temps d'échantillonnage
%Dans la fonction importrobot écrire le nom de votre fichier du
%robot planaire à 2DDL déjà conçu avec le toolbox SimScape
[DOF2_Arm,ArmInfo]=importrobot('deseignrobot1');

```

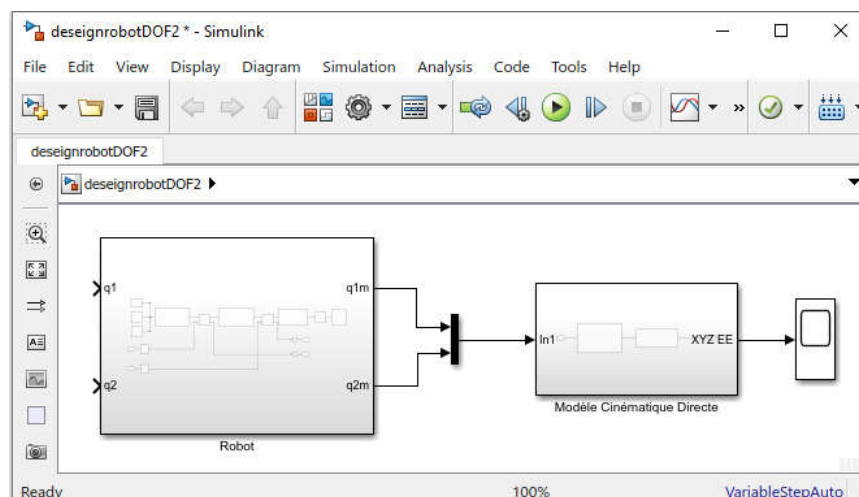
Après compilation, vérifier dans la fenêtre de Workspace que vous avez les informations présentées dans la figure suivante :

Workspace	
Name	Value
ArmInfo	1x1 RigidBodyTreeImportInfo
DOF2_Arm	1x1 RigidBodyTree
Ts	1.0000e-03

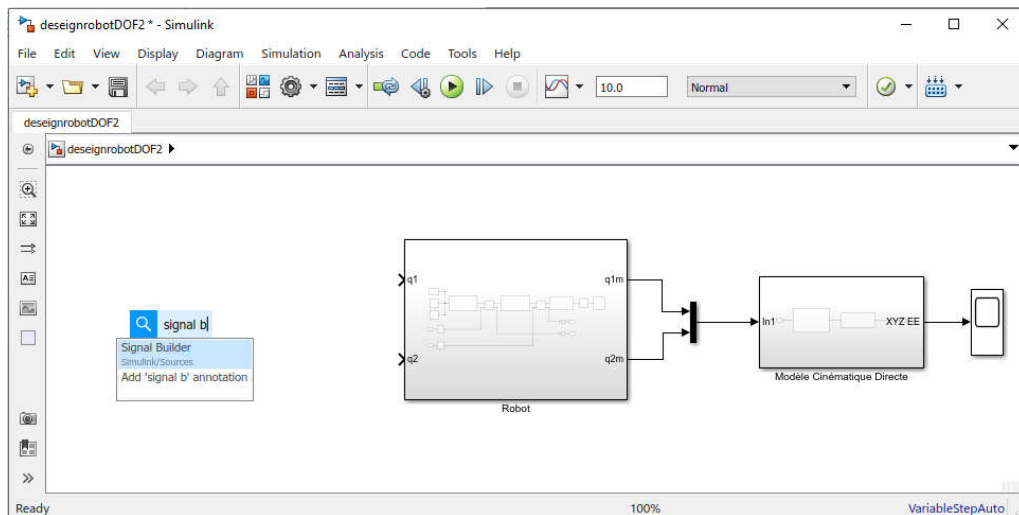
⇒ Charger le schéma de la commande du robot (du TPN°2) comme l'indique la figure suivante :



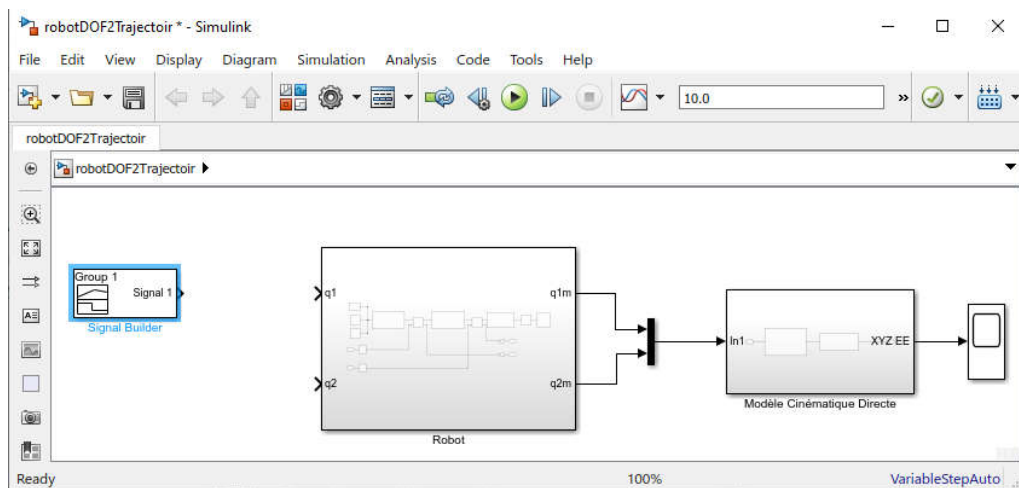
- ⇒ Enregistrer le schéma block de la commande du robot sous un autre nom.
- ⇒ Effacer les blocks de « SinWave » et les deux multiplexeurs « Joint1 » et « Joint2 »
- ⇒ Paramétrer l'oscilloscope pour une seule entrée.



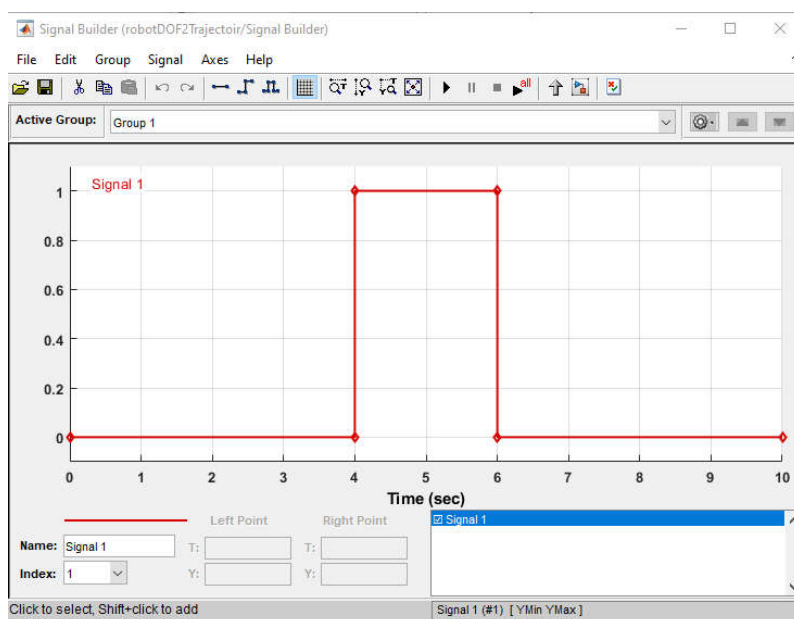
⇒ Charger le Block « Signal Builder » dans la zone de travail de Simulink



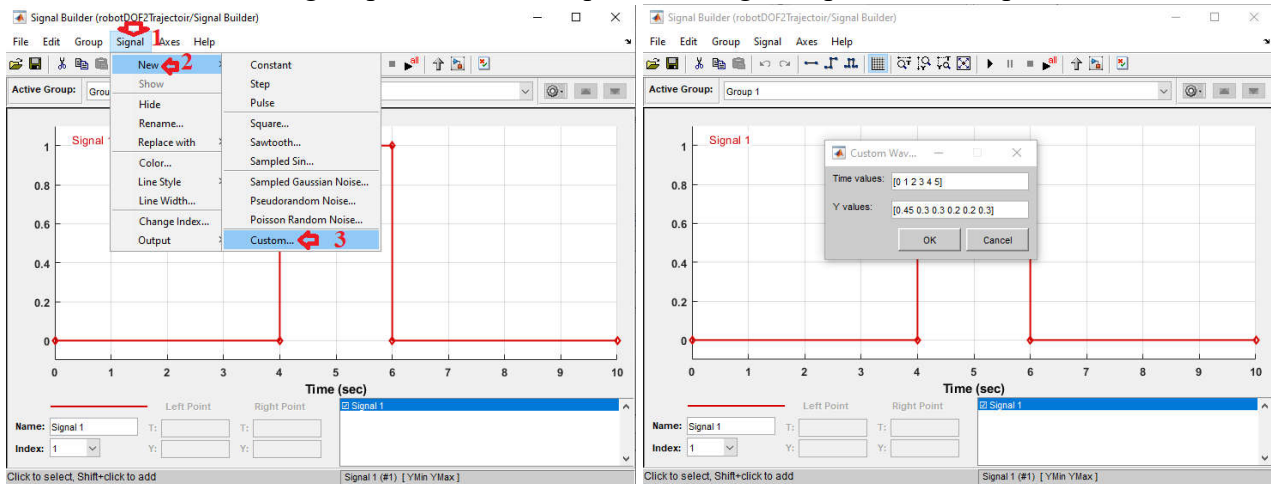
⇒ Cliquez deux fois sur « Signal Builder »



⇒ La fenêtre du « Signal Builder » s'affiche.



⇒ Pour créer un signal personnalisé, cliquer sur « Signal » puis « New » puis « Custom »



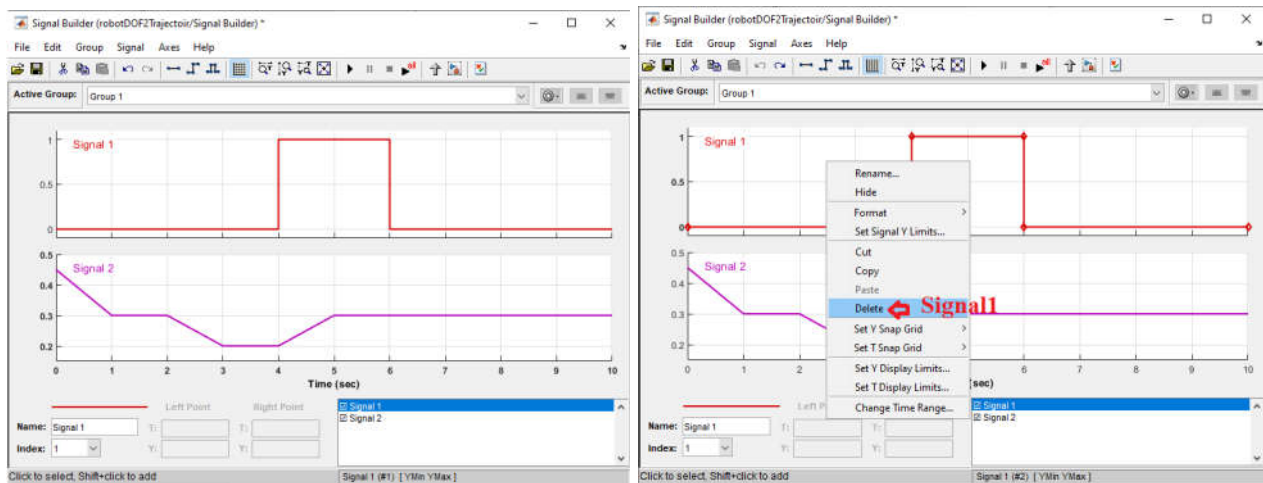
La fenêtre du « Custom Wave » s'affiche.

⇒ Introduire dans « Times Values » les valeurs suivantes : [0 1 2 3 4 5]

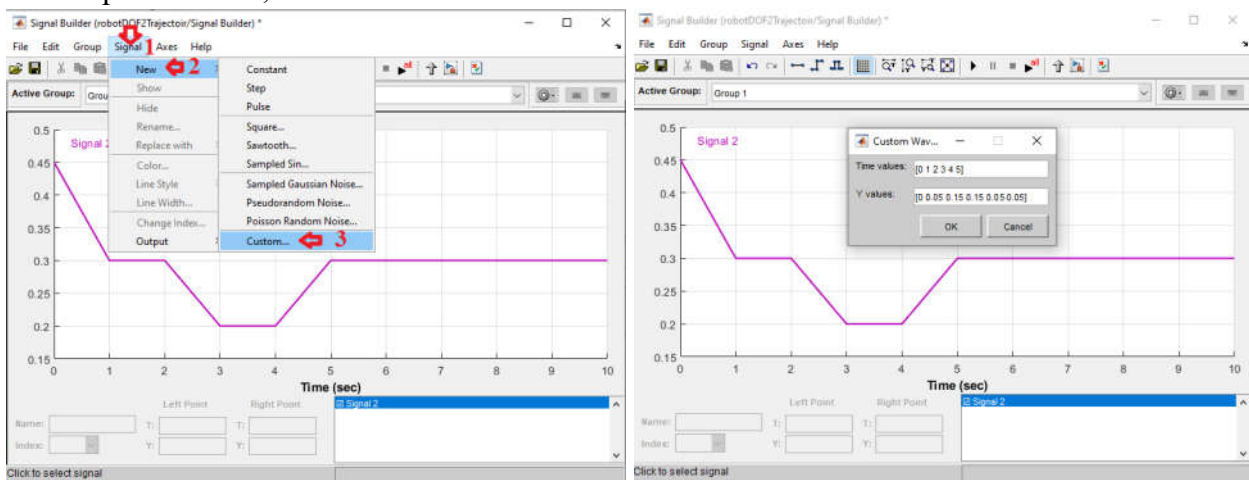
⇒ Introduire dans « Y Values » les valeurs suivantes : [0.45 0.3 0.3 0.2 0.2 0.3]

⇒ *Que représente ses valeurs.*

⇒ Effacer le « signal 1 »

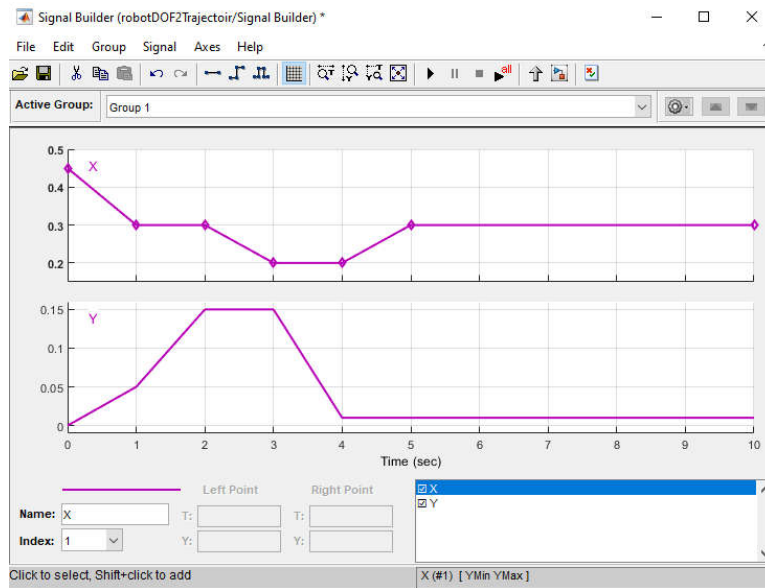


⇒ Cliquer sur « Signal » puis « New » puis « Custom » pour créer le deuxième signal personnalisé,

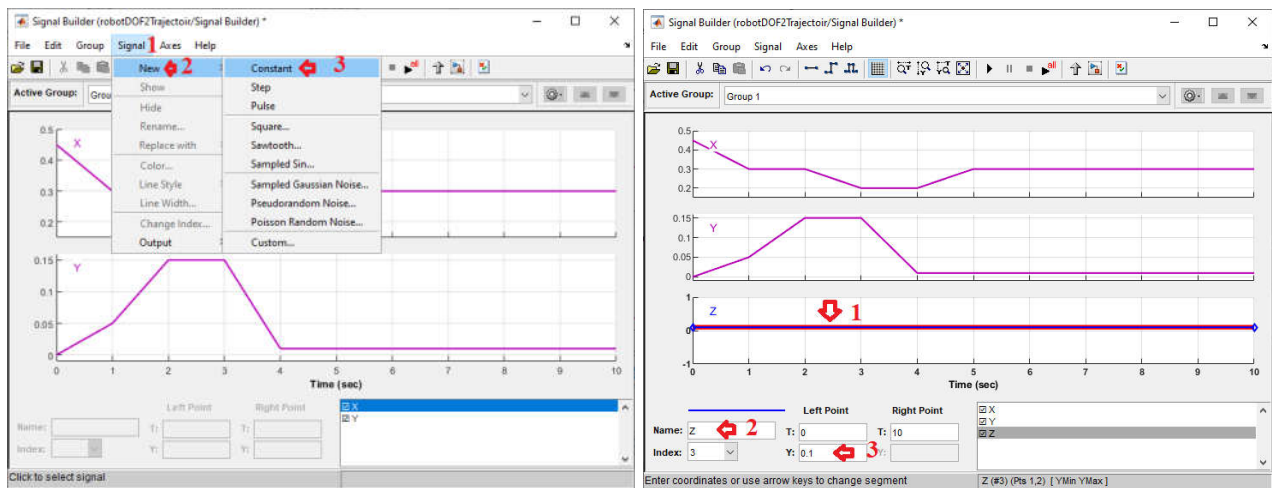


⇒ Introduire dans « Times Values » les valeurs suivantes : [0 1 2 3 4 5]

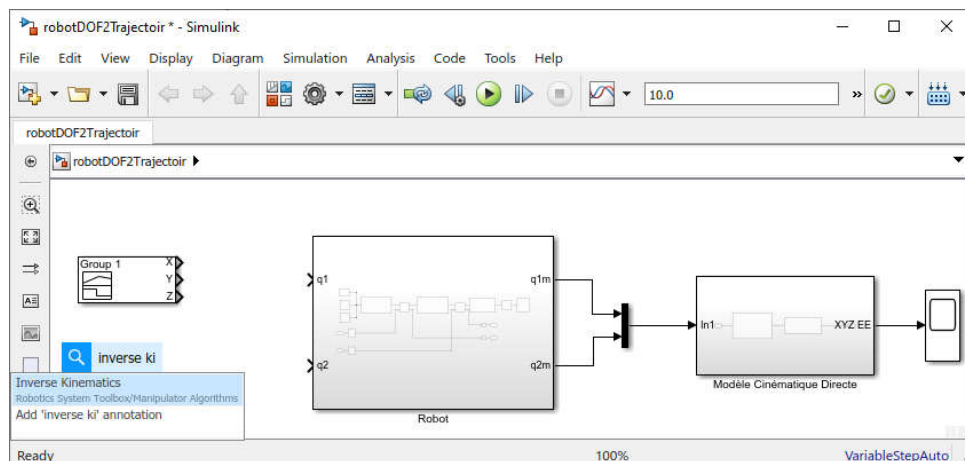
- ⇒ Introduire dans « Y Values » les valeurs suivantes : [0 0.05 0.15 0.15 0.05 0.05]
- ⇒ **Que représente ses valeurs.**
- ⇒ Renommer le « Signal 2 » en X et l'autre signal en Y



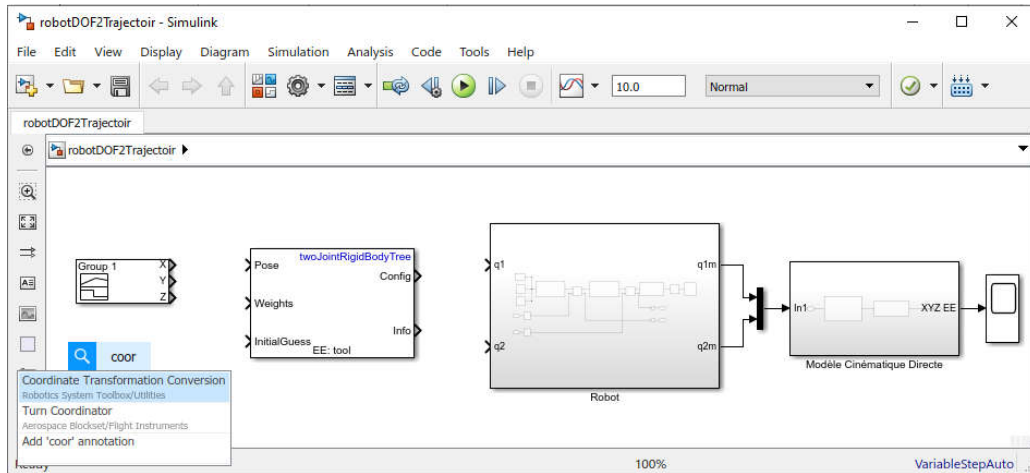
- ⇒ Ajouter le 3^{ème} signal qui représente l'axe Z et il est constant de valeur 0.1



- ⇒ Charger le block « Inverse Kinematics » dans Simulink.

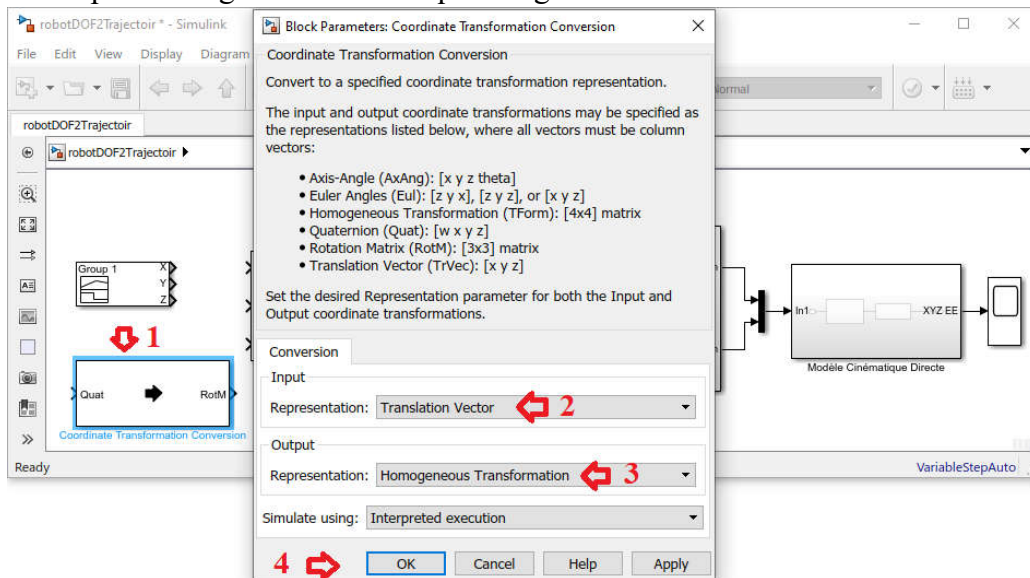


⇒ Mettre le block « Coordinate Transformation Conversion » dans Simulink



⇒ Cliquer deux fois sur le block « Coordinate Transformation Conversion » dans Simulink

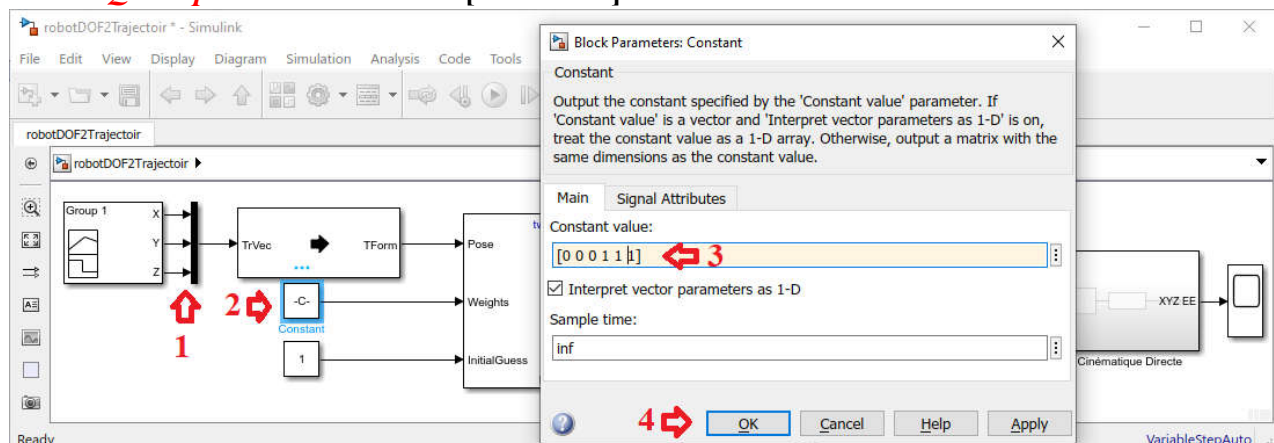
⇒ Faire le paramétrage comme l'indique la figure suivante.



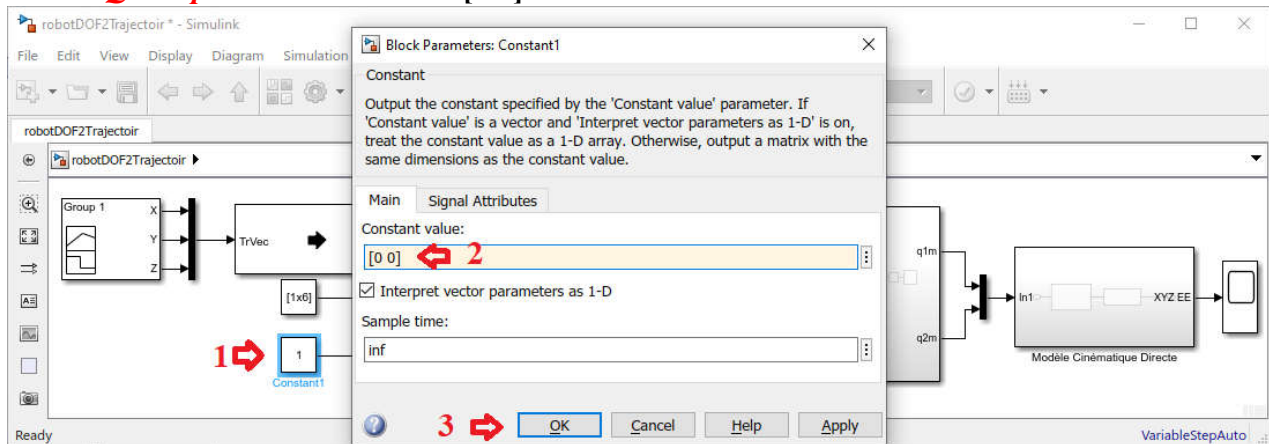
⇒ Ajouter un multiplexeur trois vert un et connecter les deux blocks « Coordinate Transformation Conversion » et « Signal Builder »

⇒ Ajouter deux blocks « Constant » et paramétrer le premier dans « constant Value » à $[0 \ 0 \ 0 \ 1 \ 1 \ 1]$, ce bloc sera connecté à l'entrée « Weights » du bloc « Inverse Kinematics »

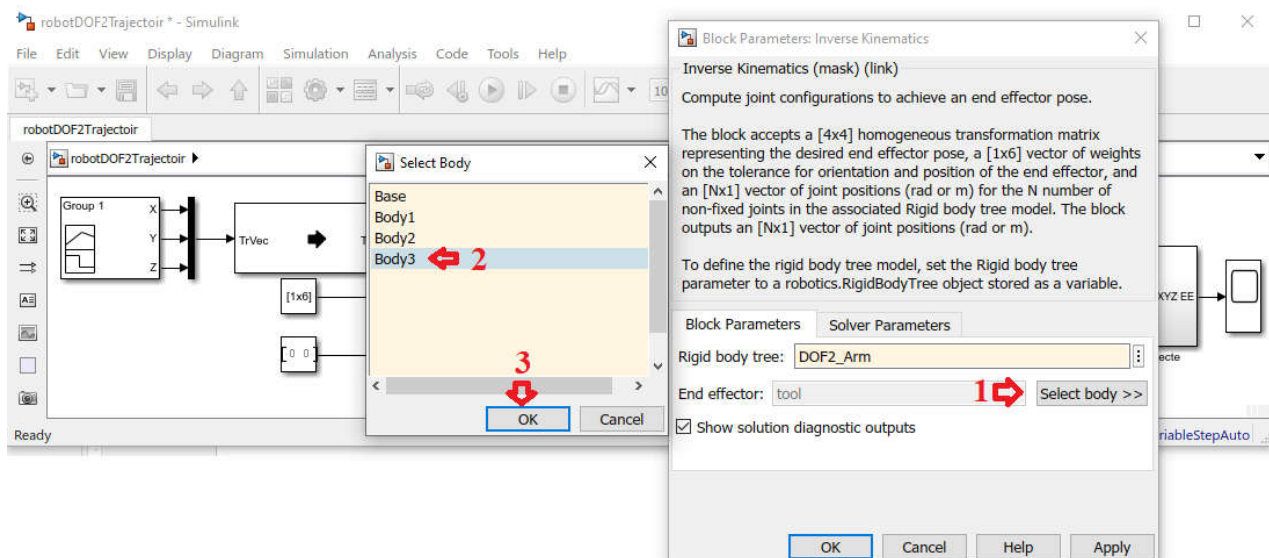
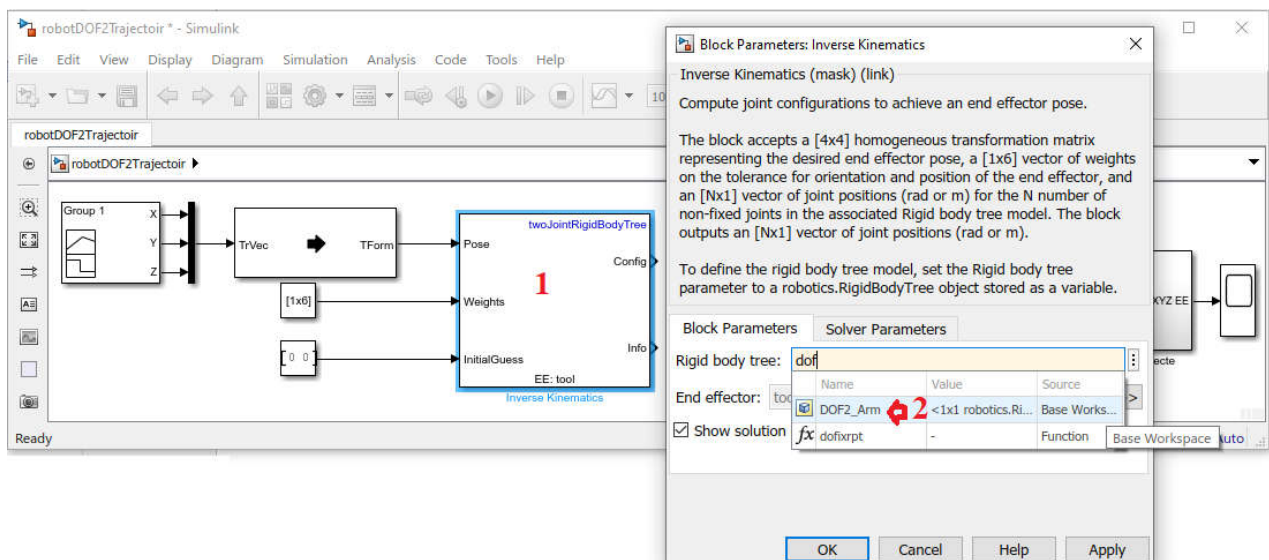
⇒ **Que représente ses valeurs $[0 \ 0 \ 0 \ 1 \ 1 \ 1]$.**

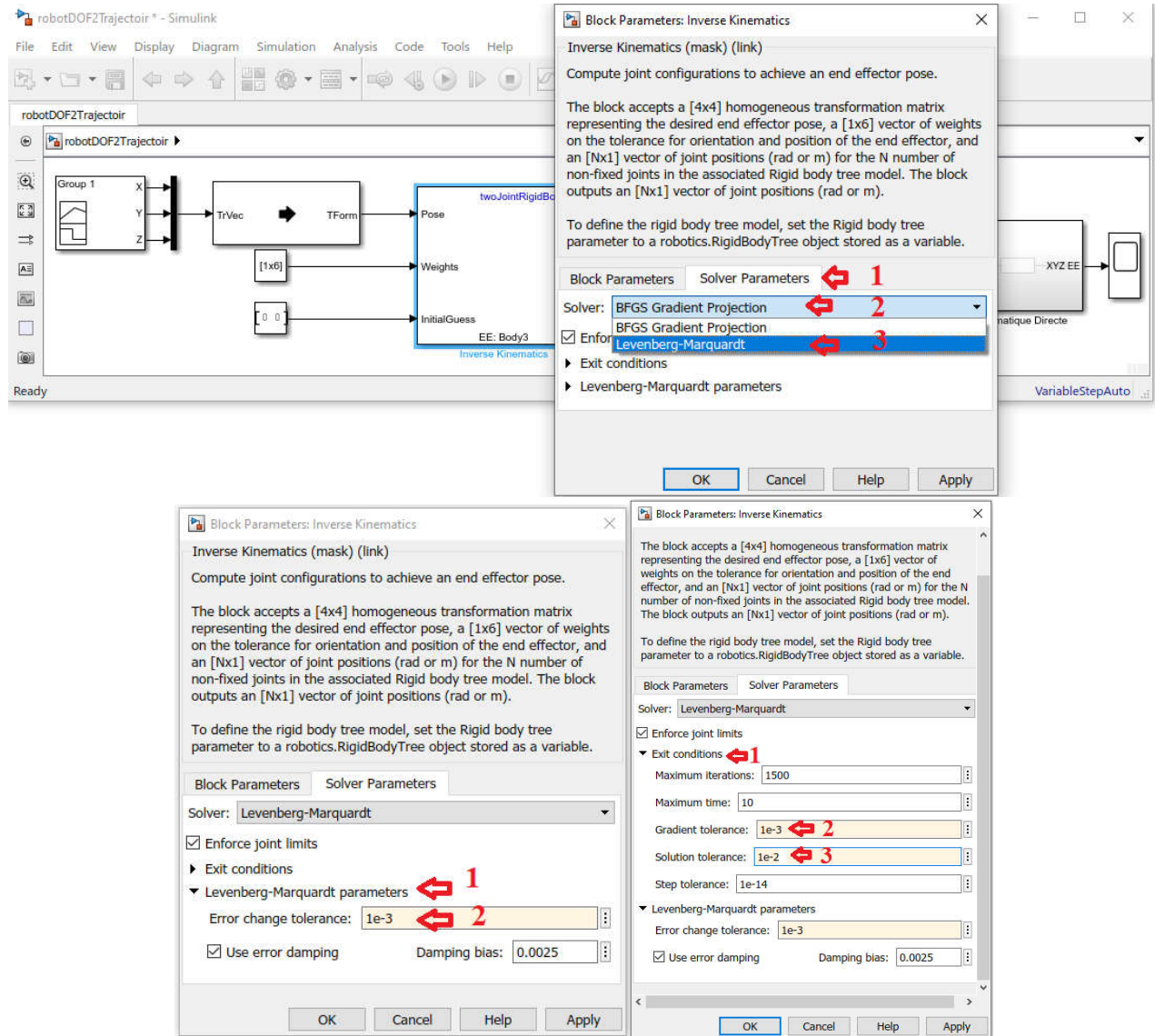


- ⇒ Connecter le deuxième block « Constant » à l'entrée « InitialGuess » du bloc « Inverse Kinematics »
- ⇒ Paramétrer le deuxième block « Constant » dans « constant Value » à [0 0]
- ⇒ **Que représente ses valeurs [0 0].**

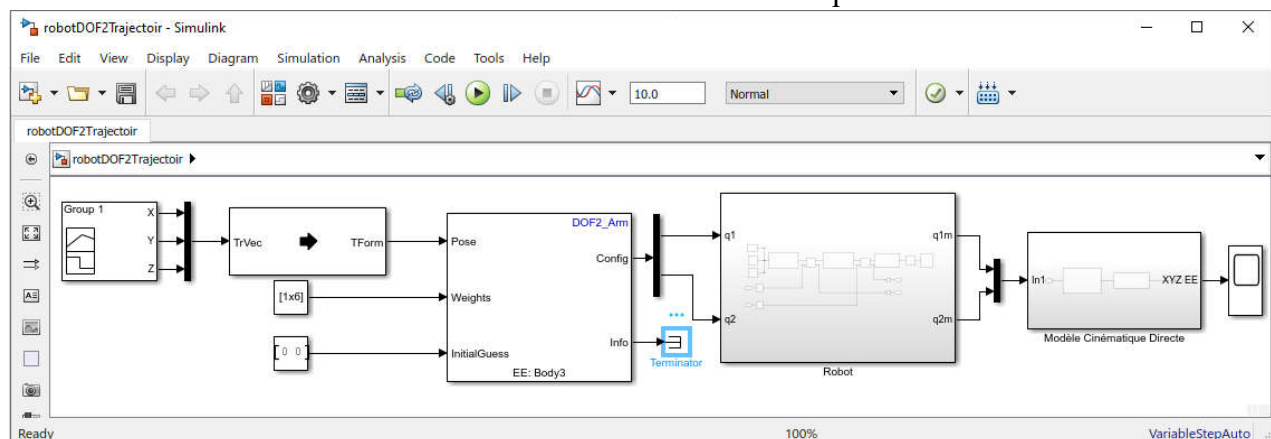


- ⇒ Cliquer deux fois sur bloc « Inverse Kinematics ».
- ⇒ Charger les configurations selon les figures qui suivent :





- ⇒ Ajouter un démultiplexeur un vers deux et connecter les deux blocs « Inverse Kinematics » et « Robot »
- ⇒ Fermer la sortie « Info » du bloc « Inverse Kinematics » par un bloc « Terminator ».



- ⇒ Lancer la simulation
- ⇒ Interpréter et conclure.
- ⇒ Essayer de réaliser une autre trajectoire par un parcours différent.

1. Bibliographie

- [1] «Mathworks,» [En ligne]. Available: <https://fr.mathworks.com/products/simscape.html>. [Accès le 10 10 2022].
- [2] «Mathworks, Inc,» [En ligne]. Available: <https://fr.mathworks.com/products/robotics.html>. [Accès le 10 10 2022].