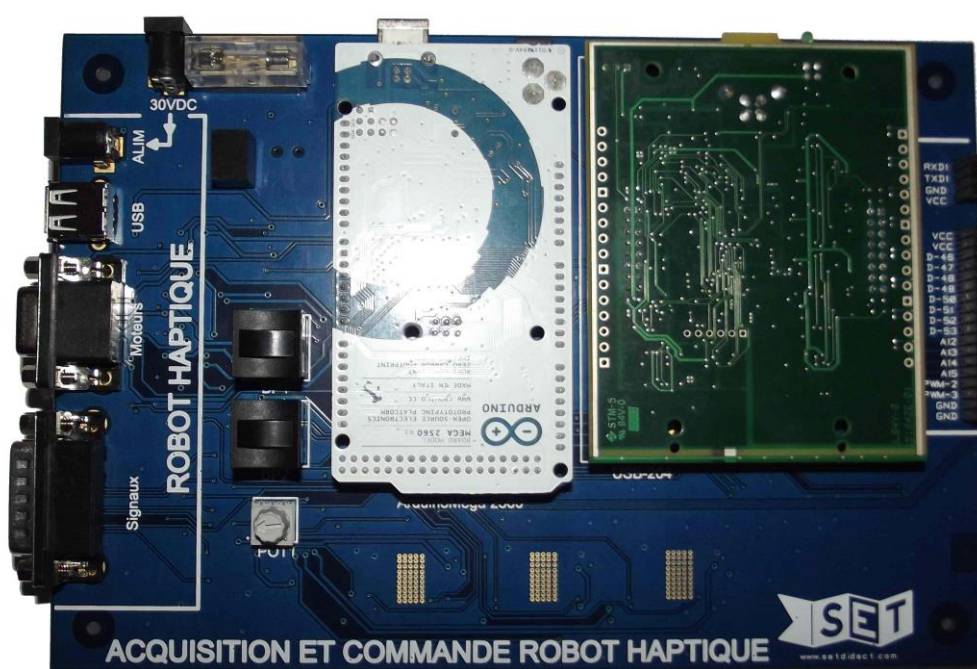


# Robot Haptique

## Manuel de mise en service



## Table des matières

<b>1 Le matériel.....</b>	<b>1</b>
<b>1.1 Contenu de l'ensemble livré .....</b>	<b>1</b>
<b>1.2 Descriptions.....</b>	<b>1</b>
1.2.1 Robot haptique didactisé : .....	1
1.2.2 Carte électronique programmable.....	3
<b>2 Connexion du matériel .....</b>	<b>9</b>
<b>2.1 Robot seul .....</b>	<b>9</b>
<b>2.2 Robot avec carte d'acquisition et de commande .....</b>	<b>10</b>
<b>3 Logiciels .....</b>	<b>11</b>
<b>3.1 Installation du driver du Robot et du logiciel de test .....</b>	<b>11</b>
<b>3.2 Installation des logiciels du fabricant.....</b>	<b>14</b>
3.2.1 Exemple d'application : Falcon Tutorial.....	14
3.2.2 Développements d'applications.....	16
3.2.2.1 : Intégration de la DLL dans LABVIEW.....	18
3.2.2.2 : Développement de l'environnement 3D avec CHAI3D.....	25
<b>3.3 Installation des drivers et paramétrage de la carte d'acquisition .....</b>	<b>25</b>
<b>3.4 Installation de la carte ARDUINO dans le logiciel MATLAB.....</b>	<b>32</b>

---



Système Etude Technique

Robot-Haptique

Mise en service



Ce document vous aide à découvrir le produit, étape par étape, du déballage à la première mise en service.

## 1 Le matériel

### 1.1 Contenu de l'ensemble livré

- 2 robots haptique didactisés,
  - 2 cartes électroniques de commande et d'acquisition des signaux programmables ArduinoMega(compatible Matlab, LabView et Flowcode),
  - 1 carte d'acquisition USB-204 à plugger sur carte de commande et d'acquisition (500 Ke/s),
  - 2 blocs d'alimentations continues 30 volts / 1 ampère,
  - 5 cordons USB A vers B,
  - 2 cordons d'alimentation jacks femelles à chaque extrémité,
  - 2 câbles 15 points haute densité et prise DIN mâle 12points,
  - 2 câbles 15 points mâle/femelle,
  - 2 connecteurs mâles shunt,
- 
- Un DVD pédagogique contenant le dossier technique, les activités pédagogiques, les logiciels.

**Il est recommandé de faire un contrôle du matériel reçu lors du déballage.**

### 1.2 Descriptions

#### 1.2.1 Robot haptique didactisé :

Le Robot Haptique est un produit pédagogique basé sur une interface Haptique de type robot delta, à structure parallèle avec 3 chaînes cinématiques fermées identiques, reliant la base à l'organe terminal. Il offre 3 degrés de liberté en translation. Chaque chaîne cinématique compose d'un moteur courant continu et un codeur incrémental : le système est à la fois capteur et actionneur d'où son nom commercial de Robot Haptique.

Une boule amovible se monte sur la base du robot. Elle permet de manipuler le robot et d'avoir à disposition 4 boutons poussoir. Pour insérer la boule glisser la dans la base jusqu'à entendre un clic, pour retirer la boule serrer la pince située à gauche du plateau de base et retirer la boule.

#### **Modifications électrique :**

Le robot a été modifié de façon à pouvoir récupérer les informations :

- ✓ Des codeurs : 6 signaux TOR,
- ✓ Des détecteurs de butée : 3 signaux TOR,
- ✓ Communication série entre la manette et la carte processeur du robot : 1 signal série TOR,

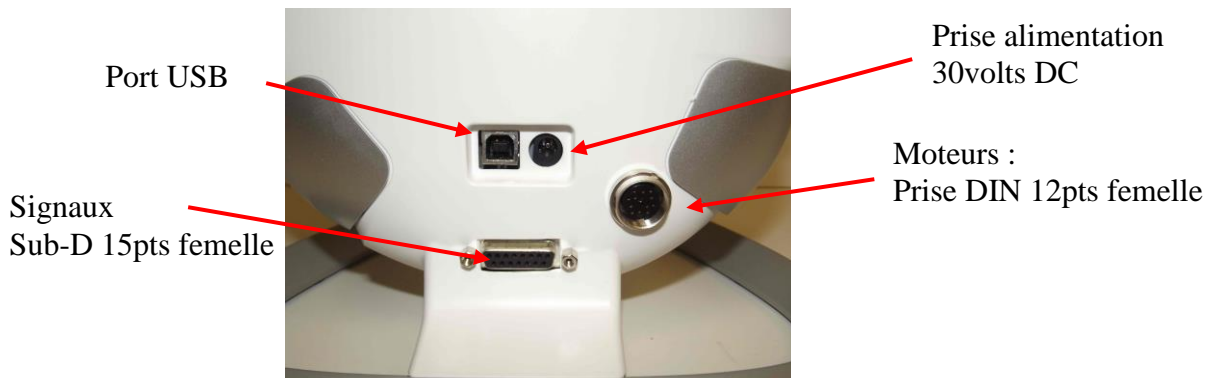
Ces signaux sont accessibles depuis la sub-D 15 points femelle située à l'arrière du robot.

La commande des trois moteurs a également été ouverte électriquement pour donner la possibilité de commander chacun des moteurs du robot depuis la carte électronique programmable par la prise DIN ronde 12 points femelle située également à l'arrière du robot.

Ces modifications sont visibles sur l'arrière du robot.



**Vue arrière :**



Le robot peut toujours être utilisé dans ses fonctions d'origine. Il suffit pour cela de câbler le connecteur Shunt sur la prise circulaire. Si ce dernier n'est pas monté, les moteurs ne sont pas commandés.

Numérotation des moteurs.

Par convention on adoptera la numérotation des moteurs suivante : lorsque le système est vue de face le moteur 1 se trouvent en haut, le moteur deux en bas à droite et le moteur 3 en bas à gauche.

Affectations des signaux de sortie du robot : Sub-D 15 points femelle

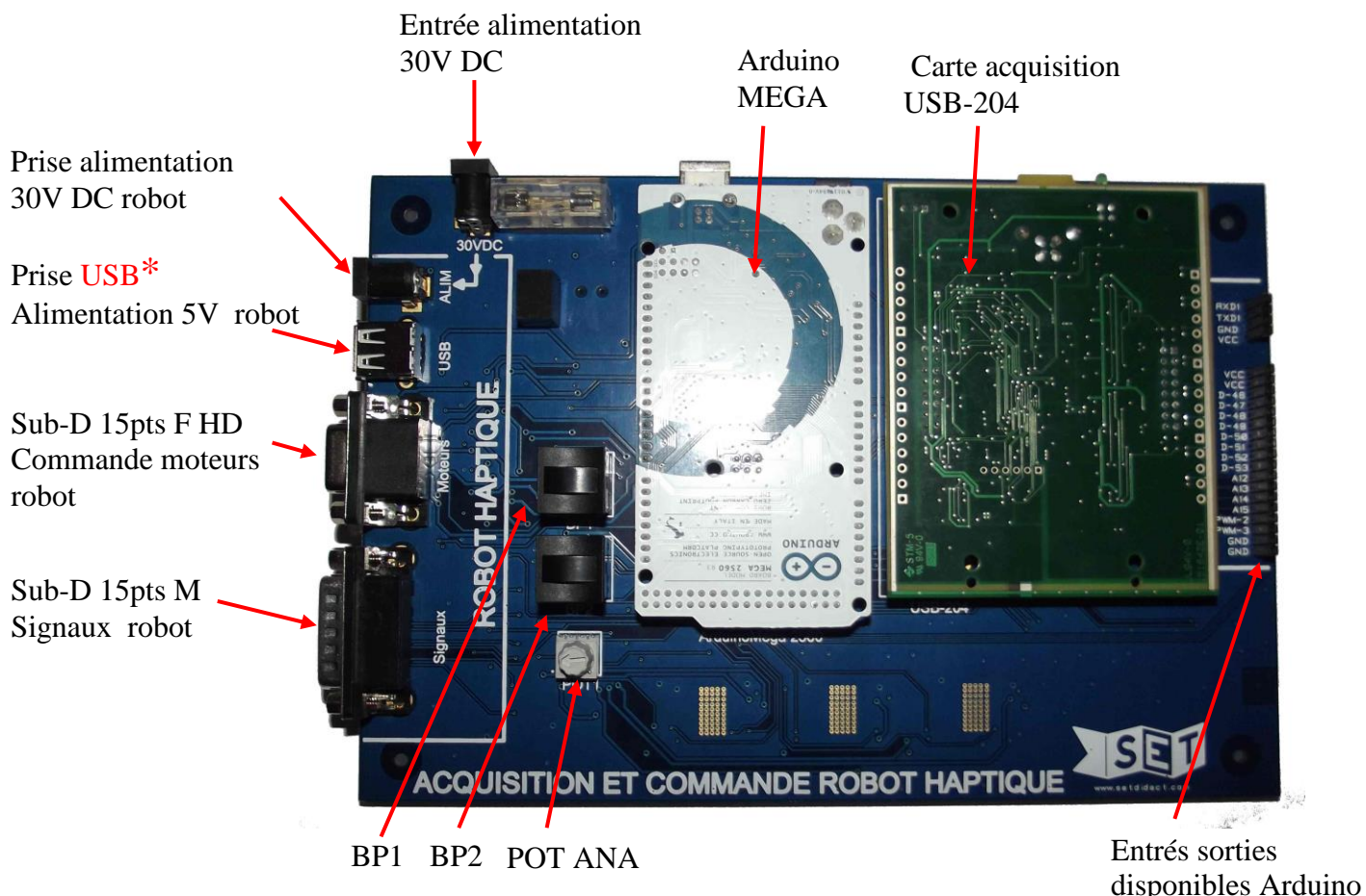
Désignation	Nom du signal	Type de signal	Sub-D 15pts femelle
Phototransistor A du codeur moteur 1	Y1A	TOR 0 - 4 V DC	4
Phototransistor B du codeur moteur 1	Y1B	TOR 0 - 4 V DC	12
Détecteur de butée bras 1	P4	TOR 0 - 3,1 V DC	1
Phototransistor A du codeur moteur 2	Y5A	TOR 0 - 4 V DC	7
Phototransistor B du codeur moteur 2	Y5B	TOR 0 - 4 V DC	15
Détecteur de butée bras 2	P5	TOR 0 - 3,1 V DC	10
Phototransistor A du codeur moteur 3	Y3A	TOR 0 - 4 V DC	13
Phototransistor B du codeur moteur 3	Y3B	TOR 0 - 4 V DC	6
Détecteur de butée bras 2	P1	TOR 0 - 3,1 V DC	9
Manette	MANP2	Liaison série TTL	3
0 volt	GND	/	2, 5, 8, 11, 14

\* : Tout ou rien.

Affectations des commandes des moteurs du robot : Connecteur DIN 12 points femelle

Désignation	Noms du signal	DIN 12pts femelle
Commande - moteur 1	MT1-	B
Pôle - moteur 1	MR1-	A
Commande + moteur 1	MT1+	L
Pôle + moteur 1	MR1+	C
Commande - moteur 2	MT2-	H
Pôle - moteur 2	MR2-	M
Commande + moteur 2	MT2+	J
Pôle + moteur 2	MR2+	K
Commande - moteur 3	MT3-	E
Pôle - moteur 3	MR3-	D
Commande + moteur 3	MT3+	F
Pôle + moteur 3	MR3+	G

### 1.2.2 Carte électronique programmable



**USB\*** : Cette prise USB sert uniquement à alimenter le robot en 5 volts. Il n'y a pas de communication entre le robot et la carte.

Affectation subD 15 points HD femelle :

Désignation	Noms du signal	DIN 12pts femelle
Commande – moteur 1	MT1-	1
Pôle + moteur 1	MR1+	2
Pôle – moteur 2	MR2-	3
0 volt	GND	4
Pôle – moteur 3	MR3-	5
Pôle – moteur 1	MR1-	6
0 volt	GND	7
Commande + moteur 2	MT2+	8
0 volt	GND	9
Commande + moteur 3	MT3+	10
Commande + moteur 1	MT1+	11
Commande – moteur 2	MT2-	12
Pôle + moteur 2	MR2+	13
Commande – moteur 3	MT3-	14
Pôle + moteur 3	MR3+	15





Affectation sub-D 15 points mâle la connexion entre le robot et la carte est droite. L'affectation est identique à celle [située à l'arrière du robot](#).

Affectation prise USB

Désignation	Noms du signal	Prise USB
+ Alimentation 5volts	+5v	1
Non connecté	/	2
Non connecté	/	3
Masse alimentation	GND	4

Affectation prise alimentation 30V DC

Désignation	Noms du signal	Prise ALIM
+ Alimentation 30 volts	+30v	1 (point milieu)
Masse alimentation	GND	2

L'alimentation du robot est protégée par un fusible 1A temporisé présent sur la carte.

L'alimentation de la carte se fait par bloc secteur 30 VDC relié sur la prise "30VDC" située en haut à gauche sur la carte.

**Attention :**

**Ne jamais alimenter la carte "Arduino MEGA" avec le bloc secteur 30VDC.**



**Carte Arduino MEGA :**

La carte didactisée est munie d'une carte microcontrôleur "Arduino MEGA".

**Affectation des E/S Robot Haptique – Arduino MEGA**

Sorties PWM :

$\mu$ C ATMega2560	ARDUINO	Noms du signal	Désignation	Commentaires
PE4	4	PWM-2	Option utilisation sortie PWM	Disponible sur Connecteur de droite
PE5	5	PWM-3	Option utilisation sortie PWM	Disponible sur Connecteur de droite
PH3	6	IN1M1	Commande IN1 pour moteur 1	Commande sens du moteur 1
PH4	7	IN1M2	Commande IN1 pour moteur 2	Commande sens du moteur 2
PH5	8	IN1M3	Commande IN1 pour moteur 3	Commande sens du moteur 3
PH6	9	IN2M1	Commande IN2 pour moteur 1	Commande vitesse moteur 1
PB4	10	IN2M2	Commande IN2 pour moteur 2	Commande vitesse moteur 2
PB5	11	IN2M3	Commande IN2 pour moteur 3	Commande vitesse moteur 3

Communication :

$\mu$ C ATMega2560	ARDUINO	Noms du signal	Désignation
PH0(RXD2)	12	RXD	Liaison série Externe
PH1(TXD2)	13	TXD	
PJ1(RXD3)	15	MAN	Mesure du signal liaison série dialogue Manette-carte processeur





## Brochage des entrées-sorties Digitales :

$\mu$ C ATMega2560	ARDUINO	Noms du signal	Entrée Sortie	Désignation	Commentaires
PH6 PA0	18, 22	Y1A	Entrée	Phototransistor A du codeur moteur 1	TOR 0 – 4V DC compatible TTL
PA1	23	Y1B	Entrée	Phototransistor B du codeur moteur 1	TOR 0 – 4V DC compatible TTL
PB0, PA2	19, 24	Y5A	Entrée	Phototransistor A du codeur moteur 2	TOR 0 – 4V DC compatible TTL
PA3	25	Y5B	Entrée	Phototransistor B du codeur moteur 2	TOR 0 – 4V DC compatible TTL
PB1, PA4	20, 26	Y3A	Entrée	Phototransistor A du codeur moteur 3	TOR 0 – 4V DC compatible TTL
PA5	27	Y3B	Entrée	Phototransistor B du codeur moteur 3	TOR 0 – 4V DC compatible TTL
PA6	28	SENS1	Entrée	Sens bras du moteur 1	TOR 0 – 5 VDC
PA7	29	SENS2	Entrée	Sens bras du moteur 2	TOR 0 – 5 VDC
PC7	30	SENS3	Entrée	Sens bras du moteur 3	TOR 0 – 5 VDC
PC6	31	ENM1	Sortie	"Enable" du driver moteur 1	0 = OFF / 1 = ON
PC5	32	ENM2	Sortie	"Enable" du driver moteur 2	0 = OFF / 1 = ON
PC4	33	ENM3	Sortie	"Enable" du driver moteur 3	0 = OFF / 1 = ON
PE0, PC3	2, 34	CAPM1	Entrée	Phototransistor d'indexation moteur 1	TOR 0 – 5 VDC
PE1, PC2	3	CAPM2	Entrée	Phototransistor d'indexation moteur 2	TOR 0 – 5 VDC
PB2, PC1	21	CAPM3	Entrée	Phototransistor d'indexation moteur 3	TOR 0 – 5 VDC
PC0	37	FSM1	Entrée	"Fault Status" driver moteur 1	0 si défaut Driver
PD7	38	FSM2	Entrée	"Fault Status" driver moteur 2	0 si défaut Driver
PG2	39	FSM3	Entrée	"Fault Status" driver moteur 3	0 si défaut Driver
PG1	40	RCMOT	Sortie	Sélection de la commande Moteur	0 = PC / 1 = Arduino
PG0	41	BP1	Entrée	Bouton poussoir BP1	0 si appuie sur BP
PL7	42	BP2	Entrée	Bouton poussoir BP2	0 si appuie sur BP
PL6	43	CM1	E	Communication avec module d'acquisition	/
PL5	44	CM2	E	Communication avec module d'acquisition	/
PL4	45	CM3	E	Communication avec module d'acquisition	/
PL3	46	D-46	E/S	Option E/S digital	Connecteur droite
PL2	47	D-47	E/S	Option E/S digital	Connecteur droite
PL1	48	D-48	E/S	Option E/S digital	Connecteur droite
PL0	49	D-49	E/S	Option E/S digital	Connecteur droite
PB3	50	D-50	E/S	Option E/S digital	Connecteur droite
PB2	51	D-51	E/S	Option E/S digital	Connecteur droite
PB1	52	D-52	E/S	Option E/S digital	Connecteur droite
PB0	53	D-53	E/S	Option E/S digital	Connecteur droite


Brochage des entrées-sorties Analogique :

Arduino	Sérigraphie sur ARDUINO	Noms du signal	Désignation	Commentaires
ADC0	A0	UALIM	Tension image de la tension générale	Indique la valeur de la tension générale (VM) divisée par 10,1
ADC1**	A1	IALIM	Tension image du courant général	Courant général (A) = $U / 2$
ADC2	A2	IMOT1ARD	Tension image du courant moteur 1	Courant moteur 1 (A) = $(U-2.5)/2$
ADC3	A3	VITMOT1	Tension image de la vitesse moteur 1	Vitesse moteur 1 (tr/s) = $U \times 10$
ADC4	A4	IMOT2ARD	Tension image du courant moteur 2	Courant moteur 2 (A) = $(U-2.5)/2$
ADC5	A5	VITMOT2	Tension image de la vitesse moteur 2	Vitesse moteur 2 (tr/s) = $U \times 10$
ADC6	A6	IMOT3ARD	Tension image du courant moteur 3	Courant moteur 3 (A) = $(U-2.5)/2$
ADC7	A7	VITMOT3	Tension image de la vitesse moteur 3	Vitesse moteur 3 (tr/s) = $U \times 10$
ADC8	A8	UMOT1ARD	Tension image de la tension du moteur 1	Tension moteur 1 = $(U - 2,5) \times 20$
ADC9	A9	UMOT2ARD	Tension image de la tension du moteur 2	Tension moteur 2 = $(U - 2,5) \times 20$
ADC10	A10	UMOT3ARD	Tension image de la tension du moteur 3	Tension moteur 3 = $(U - 2,5) \times 20$
ADC11	A11	POTANA	Tension variable 0-5 volts	Tension 0-5volts
ADC12	A12	A12	Option utilisation entrée ANA	Connecteur femelle
ADC13	A13	A13	Option utilisation entrée ANA	Connecteur femelle
ADC14	A14	A14	Option utilisation entrée ANA	Connecteur femelle
ADC15	A15	A15	Option utilisation entrée ANA	Connecteur femelle

\*\* :Sur la version 1.0 du cuivre : IALIM =  $U - 2,5$ .



**Carte d'acquisition et mesure :**

La carte USB-204 associé au logiciel d'exploitation fourni est une carte dédiée à l'acquisition, la visualisation et le stockage des signaux mesurés.

Les caractéristiques :

- 8 entrées analogiques (12 bits),
- Fréquence d'échantillonnage : 500 Ke / s,
- 8 E / S numériques,
- Un compteur 32 bits.

Brochage des entrées analogiques:

Désignation	Noms du signal	Entrée analogique	Commentaires
Tension général d'alimentation	UALIM	CH0	Indique la valeur de la tension générale (VM) divisée par 10
Courant général d'alimentation	IALIM	CH1**	Courant général (A) = U / 2
Tension moteur 1	UMOT1	CH2**	Tension Moteur1 = U x 10
Courant moteur 1	IMOT1	CH3**	Courant moteur 1 (A) = U / 2
Vitesse moteur 1	VITMOT1	CH4	Vitesse moteur 1 (tr/s) = U*10
Capteur Y1 sur roue codeuse moteur 1	Y1A	CH5	TOR 0 – 4V DC
Capteur Y1 sur roue codeuse moteur 1	Y1B	CH6	TOR 0 – 4V DC
Sens de rotation moteur 1	SENS1	CH7	TOR 0 – 5V DC

Brochage des entrées-sorties numériques :

Le brochage est ici utilisé pour la communication avec la carte "Arduino Méga"

Noms du signal	E/S USB204	Arduino Méga
CM1	DIO0	DIO43
CM2	DIO1	DIO44
CM3	DIO2	DIO45

**Fonctions complémentaires :**

- Deux boutons poussoirs bistables (BP1 et BP2) sont disponibles sur la carte didactique. Ils sont respectivement reliés à DIO 41 et DIO 42. En position enfoncée, le bouton réalise un contact à la masse (via une résistance de tirage de 1 Kohms).
- Un potentiomètre variant entre 0 et 5 VDC est relié à AI11.
- Les entrées et sorties libres de la carte Arduino ainsi que les alimentations sont disponibles sur deux barrettes femelles situées sur l'extrémité droite de la carte J5 et J16. Le brochage de J5 est prévu pour la mise en place d'un module Bluetooth.

\*\* : Sur la version 1.0 du cuivre :

- Le signal UMot1 est inversé,
- Le signal IAlim = U - 2,5,
- IMOT1 = U/4.

## 2 Connexion du matériel

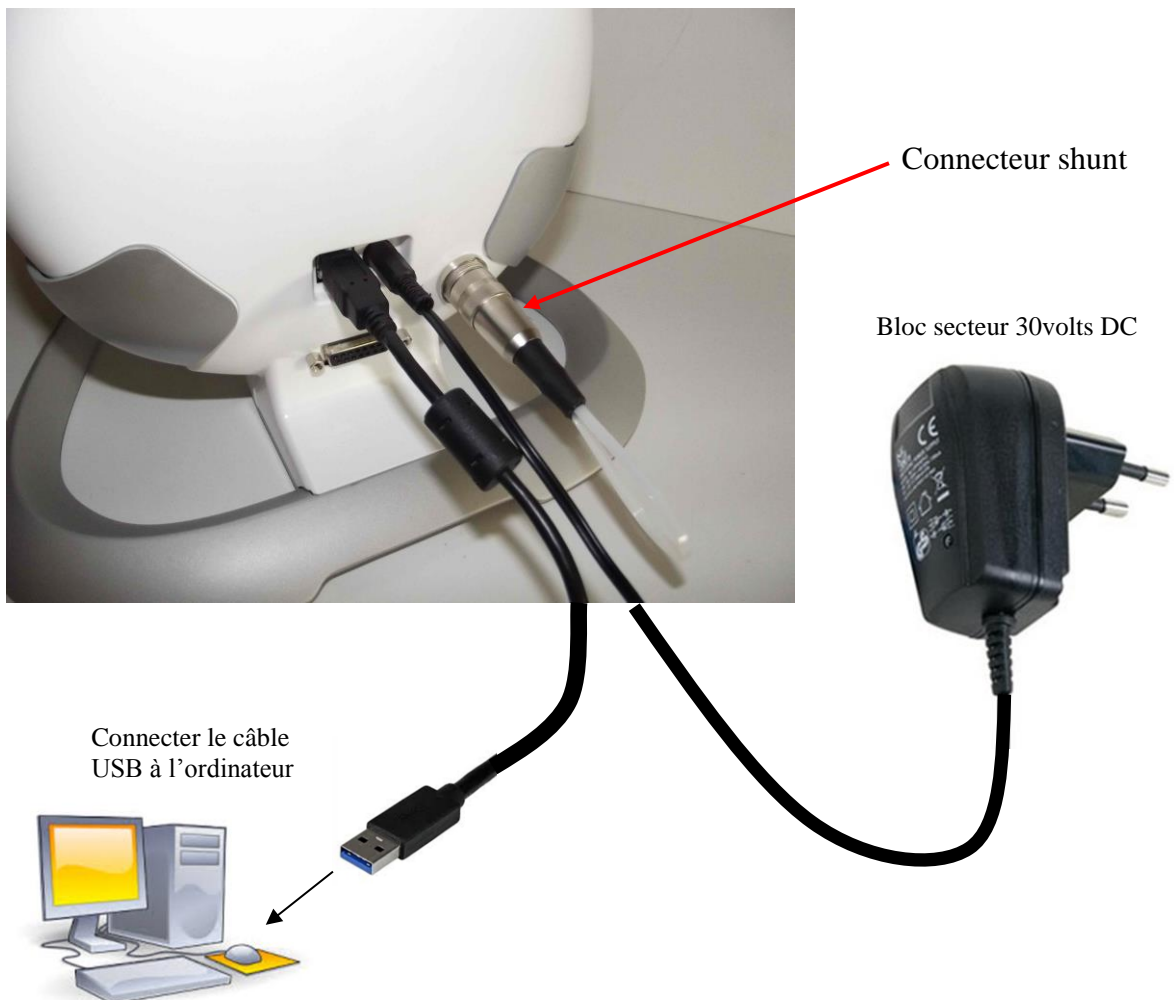
### 2.1 Robot seul

Le robot seul peut fonctionner avec les différents logiciels proposés par le fabricant (exemple : Falcon Tutorial) ou issus du monde libre (exemple : CHAI3D).

Se munir du robot avec le matériel suivant :

- ✓ L'alimentation 30 volts DC,
- ✓ Un câble USB A vers B,
- ✓ Une prise shunt.

Pour pouvoir piloter les moteurs du robot, la prise mâle shunt doit être connectée derrière celui-ci.



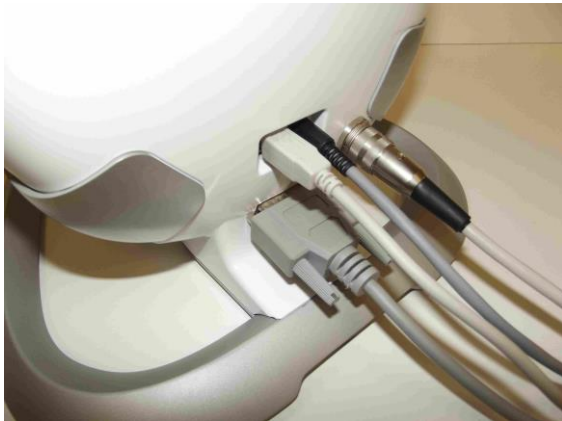


## 2.2 Robot avec carte d'acquisition et de commande

Se munir du robot avec le matériel suivant :

- 1 alimentation 30volts DC,
- 3 câbles USB A vers B,
- 1 cordon d'alimentation,
- 1 câble 15 points Haute densité mâle avec prise DIN mâle 12points,
- 1 câble SubD 15 points mâle/femelle.

Bloc secteur 30volts DC



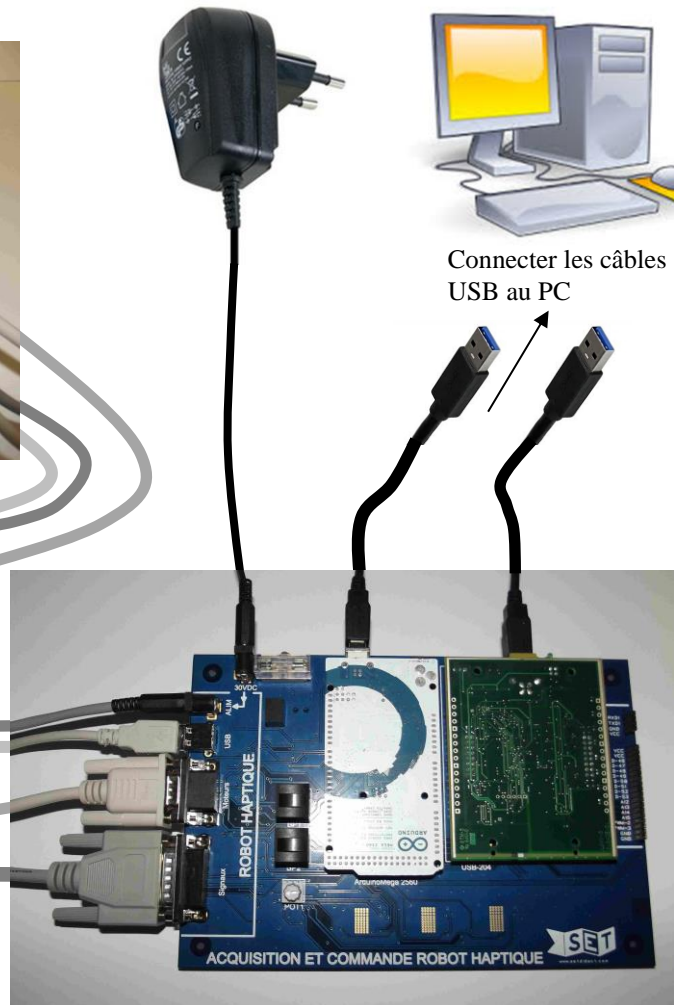
Connecter les câbles USB au PC

Cordon d'alimentation

Câble USB A vers B

Câble SubD 15pts  
HD / DIN 15pts

Câble SubD 15pts M/F



Connecter comme suit :

- Le câble sub-D 15 pts mâle-femelle permet la liaison des signaux entre la carte et le robot.
- Le câble 15 pts HD-DIN vers prise cylindrique 12 points permet la connexion des moteurs.
- Le cordon jack-jack d'alimentation entre le robot et la carte. Le bloc d'alimentation 30V sur le haut de la carte.
- Le câble USB A vers B sur la prise USB du robot et la prise USB de la carte.

**Ne jamais alimenter la carte "Arduino MEGA" directement avec le bloc secteur 30VDC.**

N.B : Le robot peut aussi être commandé par un logiciel tout en réalisant des acquisitions avec la carte Arduino et/ou la carte d'acquisition. Pour cela il faut veiller aux points suivants :

- Les entrées signaux reliées à la carte Arduino doivent bien être configurées en entrée pour ne pas perturber le fonctionnement (cf. [Brochage des entrées-sorties Digitales](#) )
- Le relais RCMOT ne doit pas être commandé pour que les moteurs soient commandés par la commande interne au robot,
- La liaison USB doit se faire entre le robot et le PC et non entre la carte Arduino et le PC.



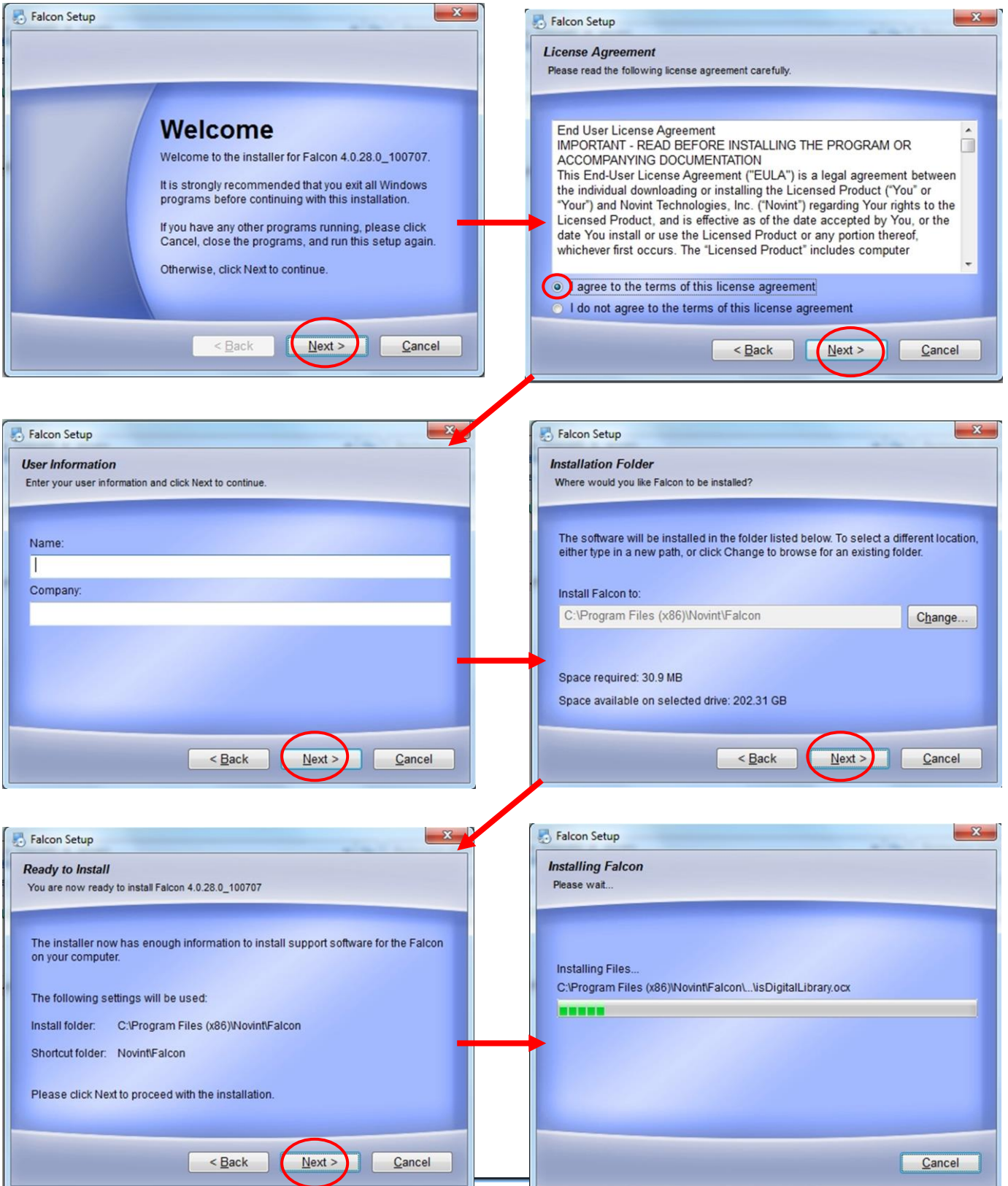


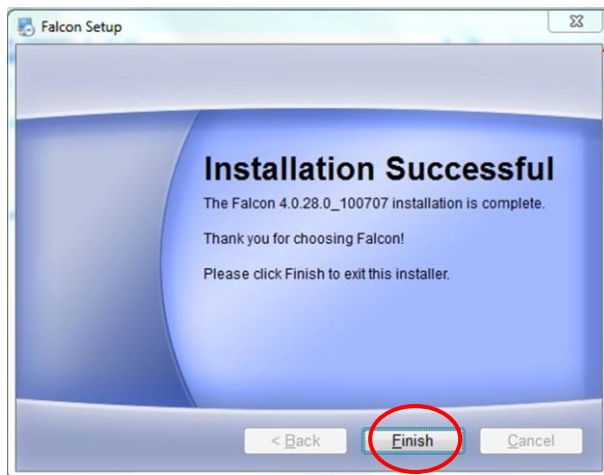
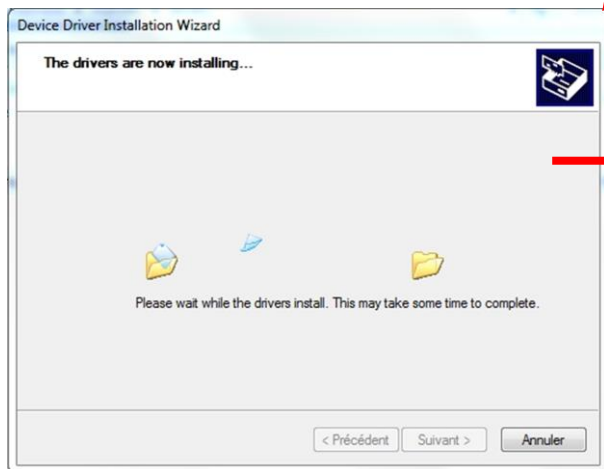
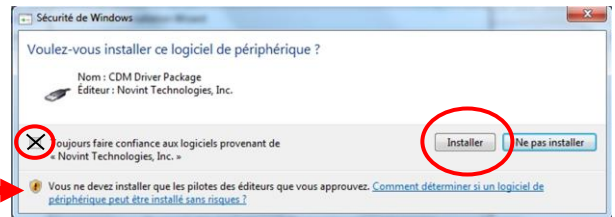
### 3 Logiciels

#### 3.1 Installation du driver du Robot et du logiciel de test

Pour installer le driver, il ne faut pas connecter le robot au PC.

Sur le CD-Rom exécuter le programme "setup.Falcon.v4.0.28.0\_100707.exe" et suivre les instructions d'installation.





En plus du driver, un logiciel de test a été installé.

Lors de l'installation du logiciel, il n'y a pas de raccourci ou de lien créé vers le programme de test. Nous vous conseillons de le constituer.

Pour accéder à l'exe du programme de test, ouvrir le dossier :  
Program File(x86)/Novint/Falcon/TestUtilities/FalconTest/FalconTest.exe

Générer un raccourci de « FalconTest.exe »





**Utilisation du programme de test :**

Ce programme a pour but de tester :

- ✓ Le fonctionnement des 3 moteurs,
- ✓ L'appui des 4 boutons poussoirs de la manette,
- ✓ L'allumage des 3 voyants se trouvant derrière le logo SET,
- ✓ La détection des butées des bras,
- ✓ Le retour des 3 codeurs.

Le programme indique également le nombre d'erreurs qu'il pourrait y avoir sur :

- ✓ La communication entre le PC et le robot,
- ✓ L'acquisition des codeurs moteurs,
- ✓ La commande,
- ✓ Le fonctionnement des moteurs.

Réaliser le câblage du chapitre 2.1 robot seul et alimenter le robot.  
Lancer le programme FalconTest

The screenshot shows the FalconTest V 1.0 software interface. Red arrows point to the following elements:

- Départ du test:** Points to the 'Open' button in the USB Communication section.
- Test des voyants en face avant robot:** Points to the 'LED 1', 'LED 2', and 'LED 3' indicators in the Front LEDs section.
- Test BP manette:** Points to the joystick button layout in the Buttons section.
- Test des butées des bras:** Points to the 'Home' button in the Homing section.
- Codeurs moteurs:** Points to the Encoder 1, Encoder 2, and Encoder 3 values in the Position Sensors section.
- Commande des moteurs:** Points to the Motor 1, Motor 2, Motor 3, and All sliders in the Motors section.



## 3.2 Installation des logiciels du constructeur

### 3.2.1 Exemple d'application : Falcon Tutorial

Le logiciel Falcon Tutorial permet de découvrir la réalité virtuelle. Il propose 4 activités :

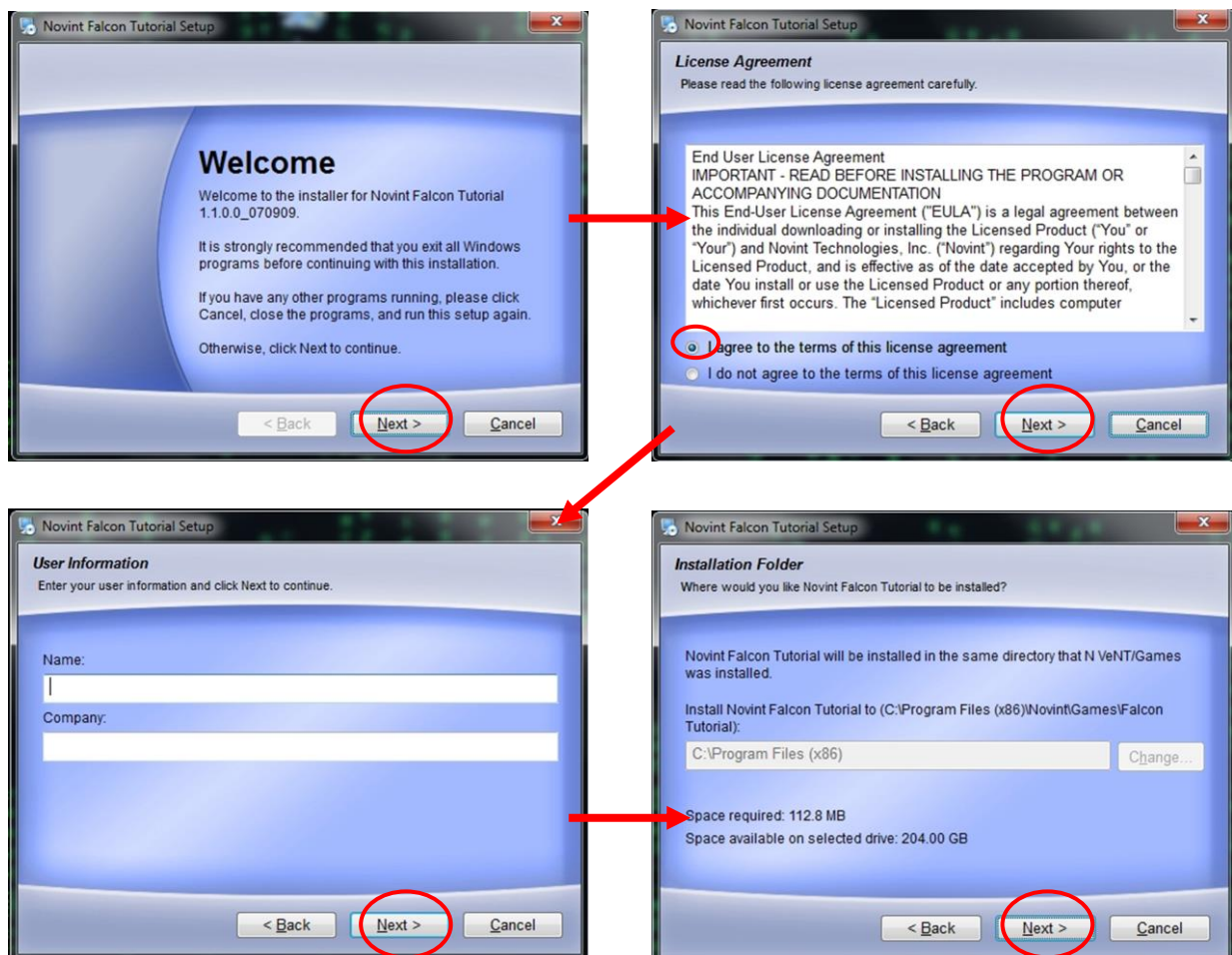
- ✓ Simulation de matière, sous la forme d'une boule qui peut être de 9 textures différentes,
- ✓ Le maniement d'une balle suspendue,
- ✓ Un jeu de rattrapage de balle,
- ✓ Un jeu de lance pierre,

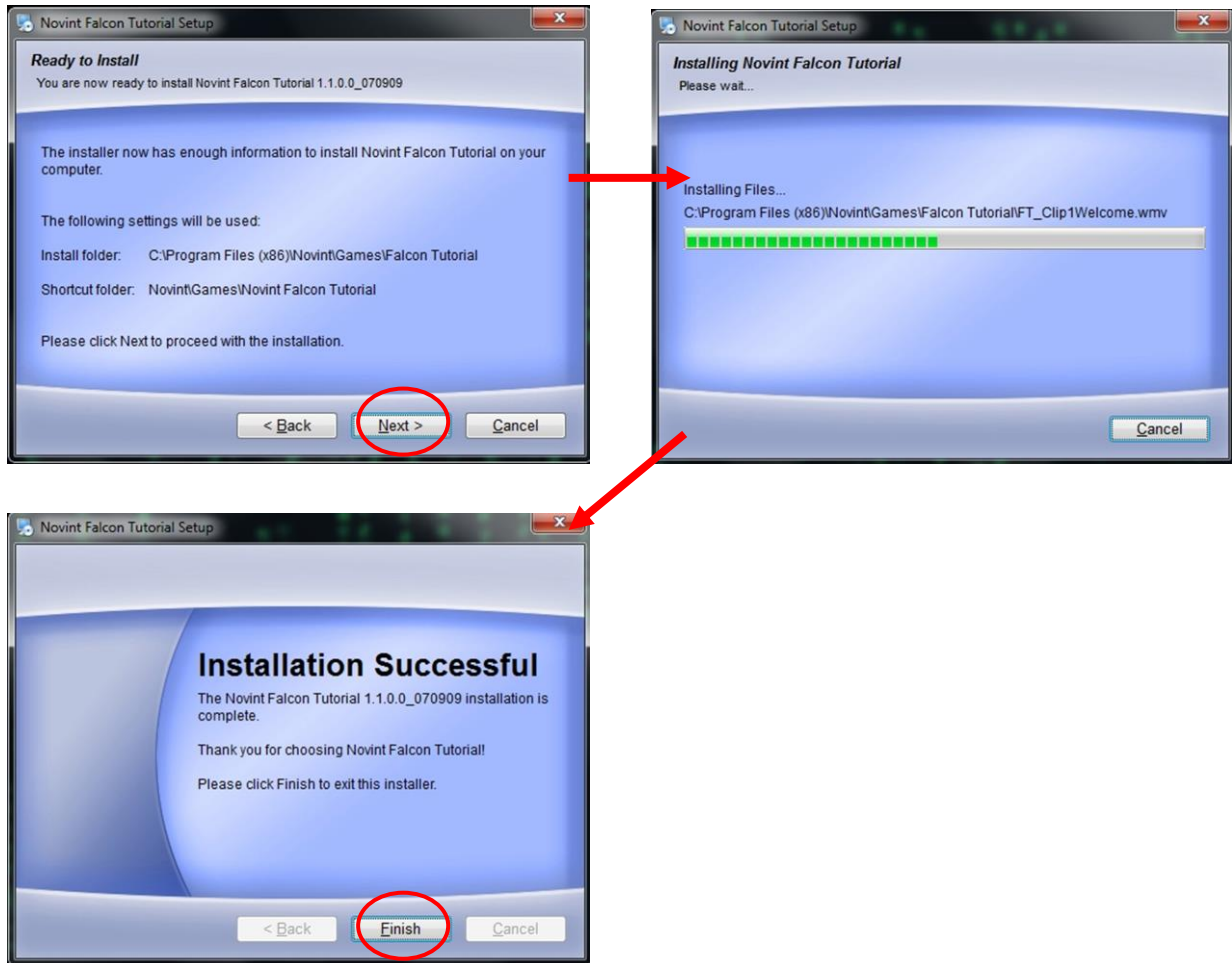
Ces activités permettent de ressentir : le poids, la forme, la texture, la dynamique et la force liés à des objets évoluant dans l'univers virtuel.

Des vidéos de démonstration sont aussi présentes.

#### **Installation :**

Sur le CD-Rom exécuter le programme "setup.FalconTutorial\_v1.1.0.0\_070909.exe" et suivre les instructions d'installation.





Lors de l'installation du logiciel, il n'y a pas de raccourci ou de lien créé vers le programme Falcon tutorial. Nous vous conseillons de le constituer.

Pour accéder au programme, ouvrir le dossier :

Program File(x86)/Novint/Games/Falcon Tutorial/ Falcon Tutorial.exe

Nous vous recommandons de créer un raccourci de l'application « Falcon Tutorial.exe »

#### **Utilisation du programme Facon Tutorial :**

Réaliser le câblage du chapitre 2.1 robot seul et alimenter le robot.

Lancer le programme Falcon Tutorial

Le programme a été développé pour le marché américain. Le clavier utilisé est de type QWERTY. Dans l'interface de navigation du logiciel, lorsque vous voulez passer les vidéos de démonstration, appuyer sur la touche A de votre clavier (et non Q comme indiqué dans le menu).

#### **Choix du mode :**

Il existe trois modes de fonctionnement pour le logiciel de démonstration. Pour modifier le mode il faut modifier le fichier "FT\_config.txt". Ce dernier se situe dans "C:\ProgramData\Novint\Games\Falcon Tutorial" il suffit de modifier le mode indiqué après define :

KIOSK\_MODE : il s'agit du mode par défaut avec des vidéos d'introduction en anglais et un temps d'utilisation limité à 11 minutes et 36 secondes.

TUTORIAL\_MODE : identique au KIOSK\_MODE mais sans limite de temps.

DEMO\_MODE : Utilisation directe sans vidéos d'introduction.

### **Problème d'installation Falcon Tutorial :**

Lors de l'installation il se peut que lien entre l'application et la dll ne soit pas réalisé (variable d'environnement).

Ce problème est lié aux droits et peut arriver en mode administrateur sur n'importe quel système d'exploitation (XP, Vista, 7, 8 ou 8.1).

Lors de l'exécution de Falcon Tutorial vous aurez alors un message d'erreur indiquant un problème de dll.

Pour résoudre simplement ce problème **copier** le fichier hdl.dll situé dans "C:\Program Files (x86)\Novint\Falcon\HDAL\bin" et coller le dans les répertoires "C:\Windows\System32" et "C:\Windows\SysWOW64" puis redémarrer votre PC.

Attention il s'agit bien de faire un copier-coller et non un couper-coller, le fichier hdl.dll doit toujours être présent dans "C:\Program Files (x86)\Novint\Falcon\HDAL\bin".

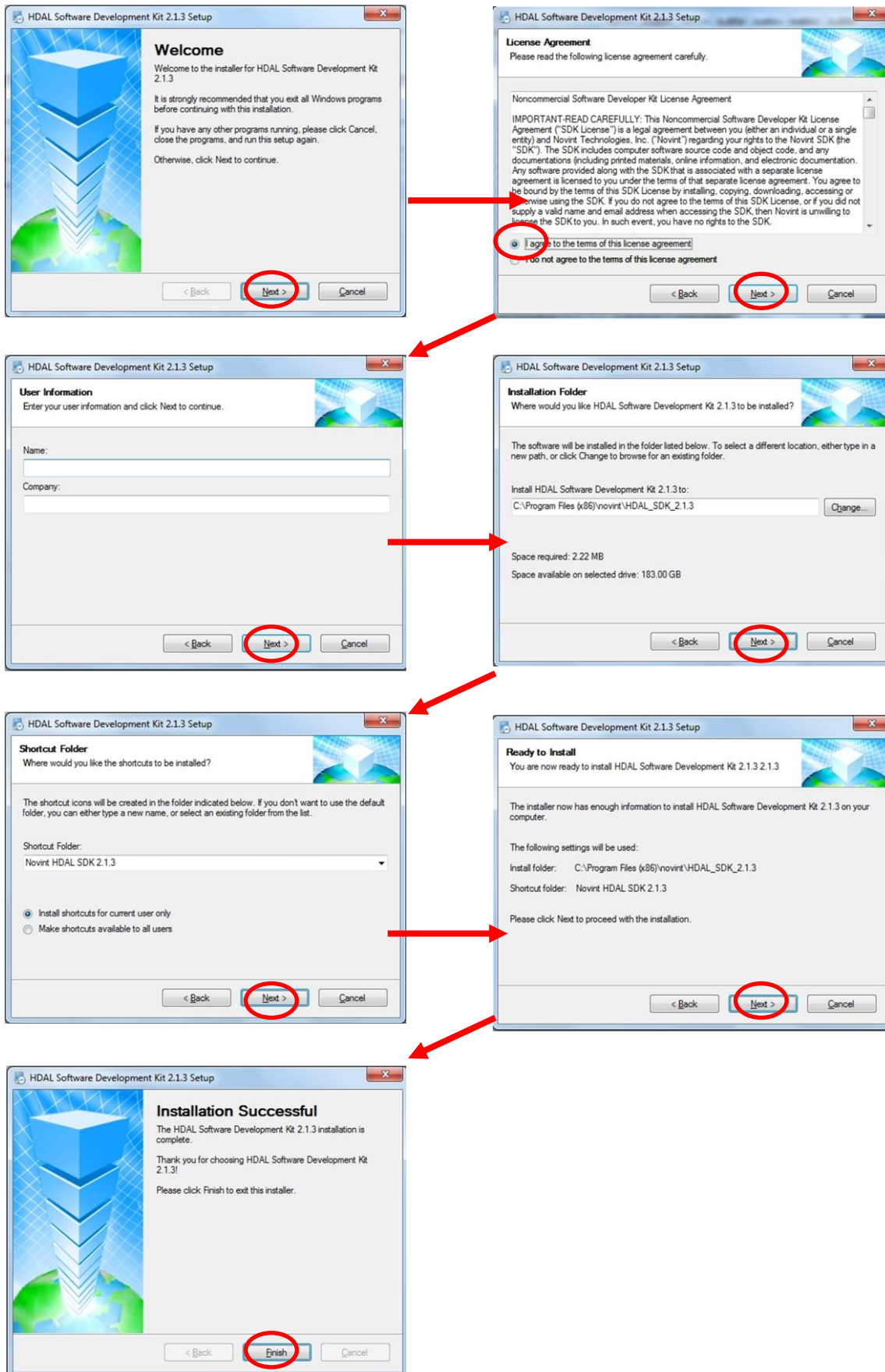
### **3.2.2 Développement d'applications**

Le SDK (Software Development Kit) Novint est une plate-forme C++ sous Windows, il est livré avec de la documentation et des exemples de code. Ce kit de développement logiciel est utilisé pour créer vos propres applications et interfaces personnalisées pour le périphérique Novint Falcon.





Sur le CD-Rom exécuter le programme " HDAL\_SDK\_2.1.3\_setup.exe" et suivre les instructions d'installation.





### 3.2.2.1 : Intégration de la DLL dans LABVIEW

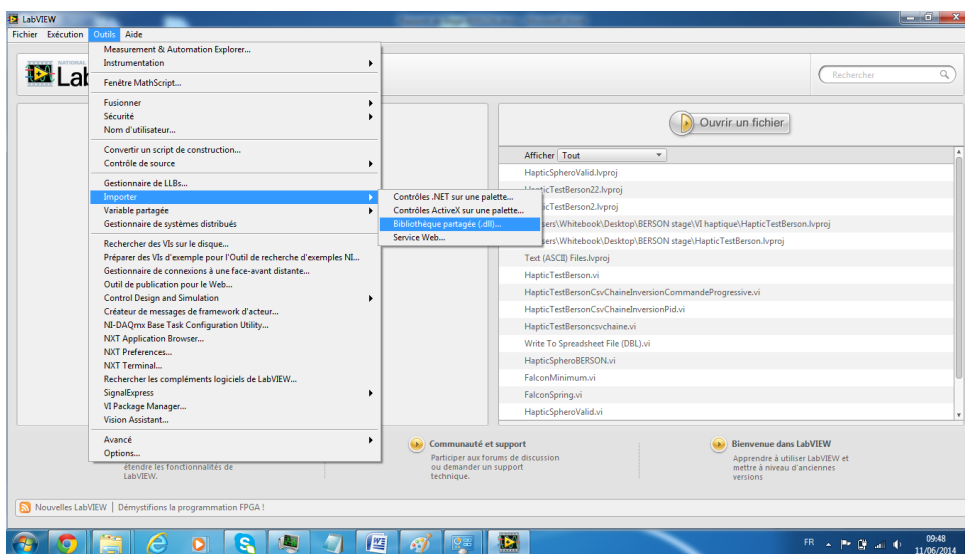
Afin de pouvoir contrôler le Robot Haptique sous Labview, il est nécessaire de créer une librairie Labview issue du programme de base de Novint.

Au préalable il faut avoir installé les drivers du robot et le programme SDK (Software Développement Kit )

Ouvrir LABVIEW et suivre les étapes suivantes :

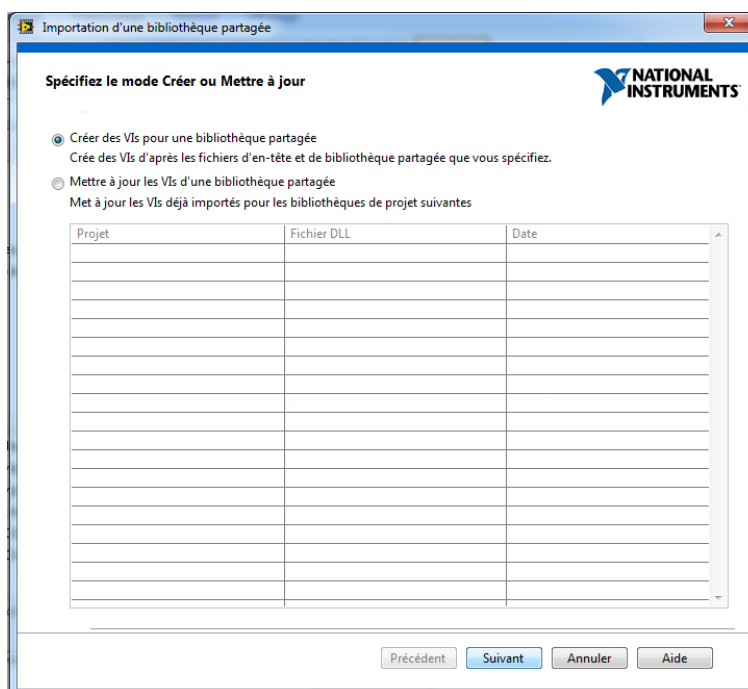
#### **Etape 1 : Importer la bibliothèque partagée (.dll) :**

Cliquer sur « outils », puis sur « importer » et finalement sur « Bibliothèque partagée (.dll) »,



#### **Etape 2 : Importer la librairie :**

Sélectionner « Créer des VIs pour une bibliothèque partagée », puis cliquer sur suivant,





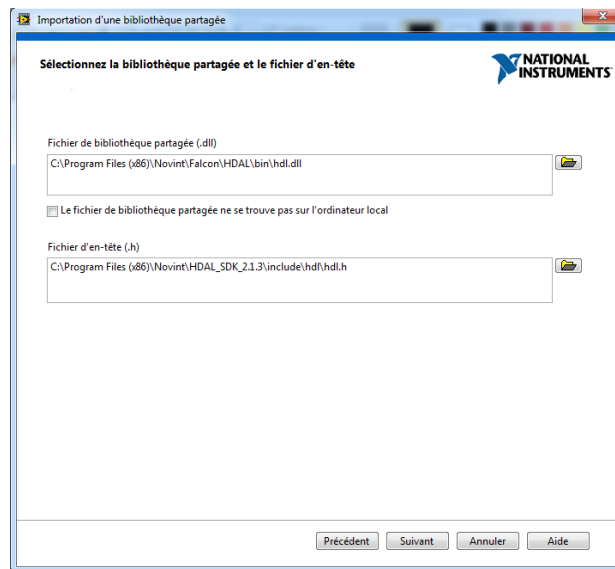
**Etape 3 : Indiquer les emplacements de la librairie et de son fichier d'en tête :**

Le fichier .dll, se situe ici : C:\Program Files (x86)\Novint\Falcon\HDAL\bin\hdl.dll

**La dll issue du SDK ne fonctionne pas très bien selon les versions de Labview utilisées, nous vous conseillons donc de ne pas prendre la librairie partagée issue du SDK mais celle utilisée pour les programmes Novint.**

Prendre le fichier « header » dans le SDK :

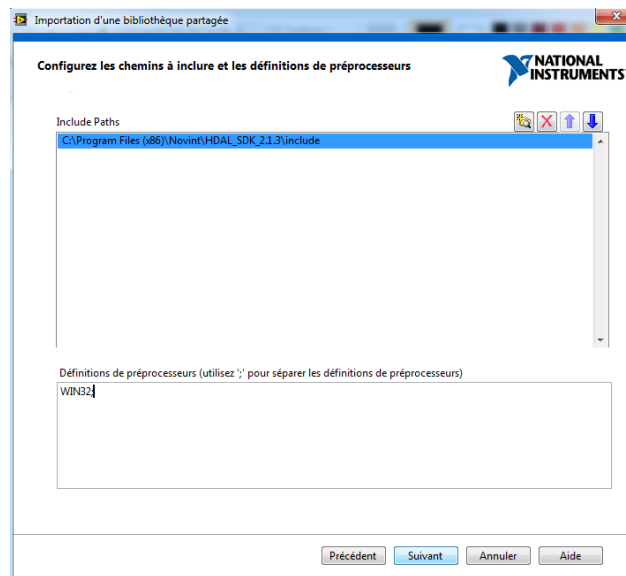
C:\Program Files (x86)\Novint\HDAL\_SDK\_2.1.3\include\hdl\hdl.h



Cliquer ensuite sur suivant.

**Etape 4 : Spécifier la location du dossier « include » :**

C:\Program Files (x86)\Novint\HDAL\_SDK\_2.1.3\include, puis il faut définir le préprocesseur « WIN32 ».



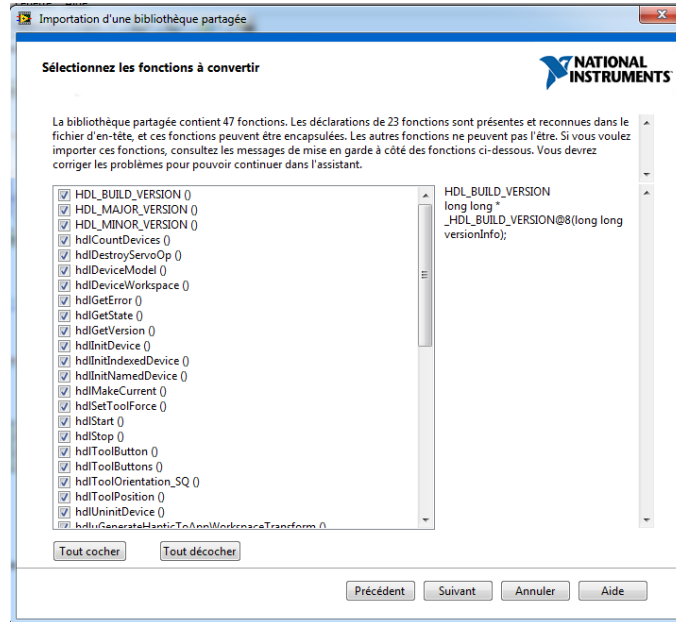
Cliquer sur suivant pour valider.



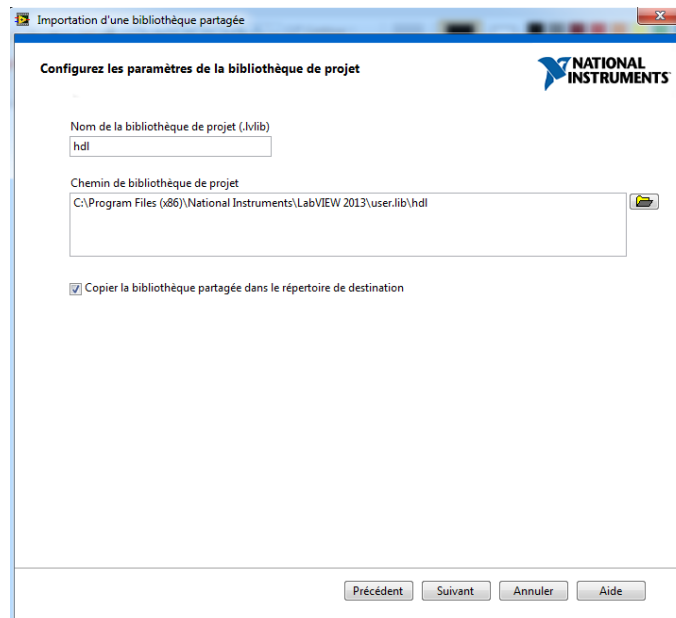


### Etape 5: Sélection des fonctions à convertir

23 fonctions sur 47 ont été générées. En effet, plusieurs fonctions ne peuvent pas être extraites car nous avons au préalable choisi la dll qui ne correspond pas au Header. Sélectionner « Tout cocher » et cliquer sur suivant.



### Etape 6: Nommer la librairie et indiquer son emplacement

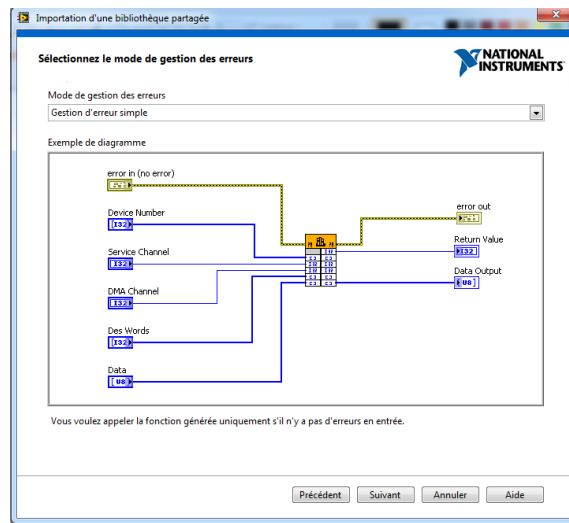


Cliquer sur suivant



### Etape 7: Sélectionner la gestion d'erreurs

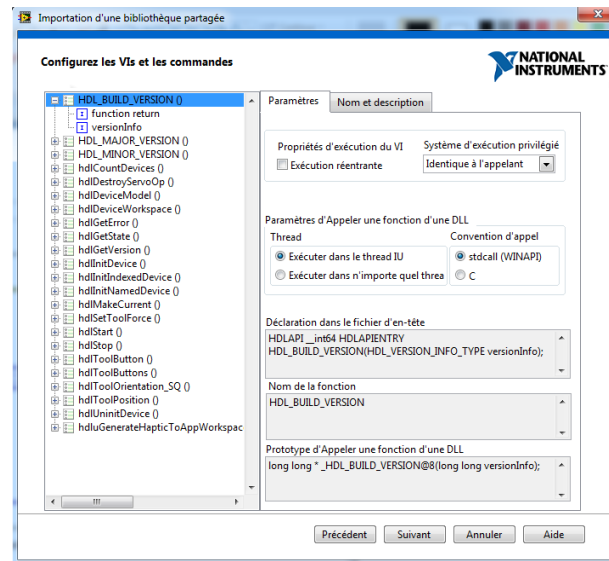
Nous préconisons la gestion d'erreurs simples, afin de pouvoir déboguer le programme.



Cliquer sur suivant.

### Etape 8:

Vérifier que la convention d'appel des fonctions est bien « stdcall (WINAPI) », puis cliquer sur suivant.

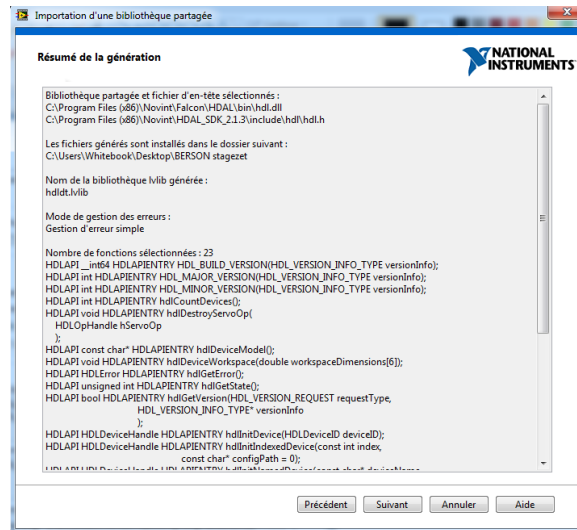


N.B : Labview ne peut pas savoir lorsqu'il importe des fonctions si les paramètres passés en argument dans la fonction sont des paramètres d'entrée, de sortie ou les deux. Par défaut il mettra systématiquement les deux. Vous pouvez néanmoins indiquer pour chaque fonction si le paramètre est un paramètre d'entrée (Ex : `hdlSetToolForce` qui prend en entrée la force à appliquer selon les axes X, Y et Z) ou un paramètre de sortie (Ex : `hdlToolPosition` qui retourne la position du robot en X, Y et Z en mètre). Pour connaître les fonctionnalités des fonctions importées reportez-vous à la doc : « `\Program Files (x86)\Novint\HDAL_SDK_2.1.3\doc\HDAL_API.pdf` ». Les fonctions principales sont expliquées peu après dans le présent document.

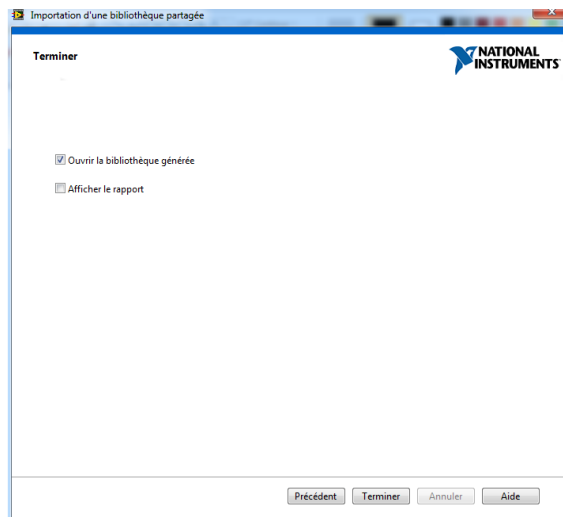


**Etape 9 :**

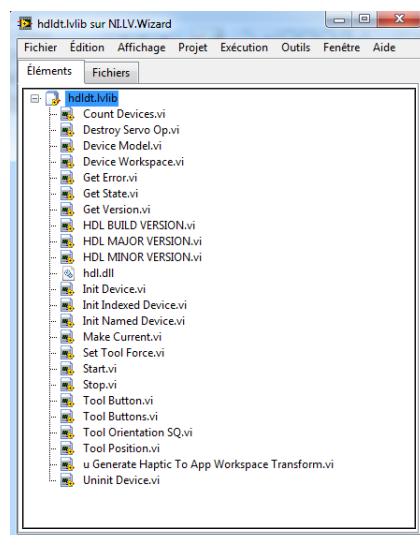
La librairie est générée, cliquer sur suivant pour commencer à programmer.



**Etape 10 :**

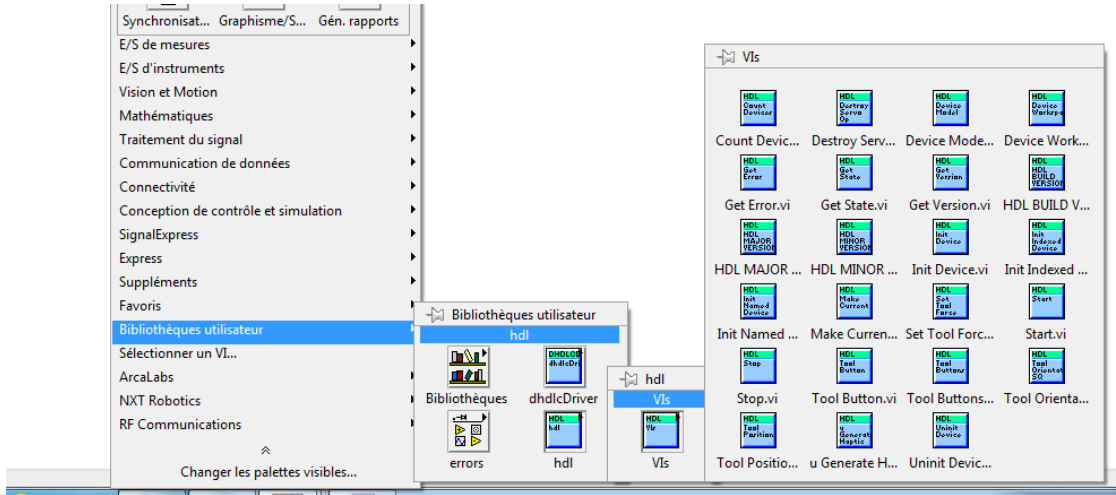


**Etape 11 : Début de la programmation.**





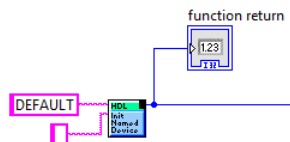
La librairie devrait se situer dans la palette d'outils de Labview (clic droit sur le diagramme, bibliothèques utilisateur, hdl, Vis).





Description des fonctions utilisées :

- « Count device » : Renvoie en sortie le nombre d'appareils Novint détectés (type entier).
- « Device Model » : Renvoie en sortie le type d'appareil Novint connectés (Falcon, Default...) (type chaîne de caractères).
- « Get Version » : Renvoie la version de l'appareil détecté (de type Réel).
- « Init Named Device » : Initialise le robot connecté et détecté en fonction de son nom, comme ceci :

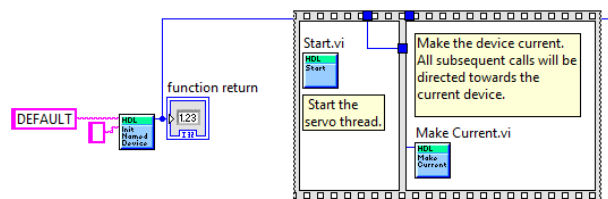


(Paramètre 1 de type chaînes de caractères : le nom par défaut de notre robot est « DEFAULT », le paramètre 2 (« config path ») de type chaîne de caractère, aucun emplacement n'est spécifié ici.)

« Make Current » : Envoi le courant dans les moteurs, ceux-ci doivent-être « souples ». Il faut que le paramètre d'entrée soit issu de la fonction « Init Named Device » (celui-ci indique si le robot a été détecté ou non).

« Start » : On va pouvoir dès l'appel de cette fonction terminer l'initialisation du robot.

Il est fortement conseillé d'utiliser une **séquence déroulée** pour assurer l'initialisation étape par étape :



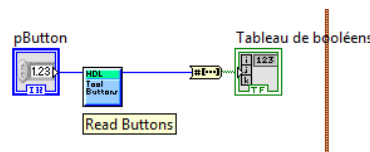
Les fonctions qui suivent doivent être employées dans une boucle de période minimale 1ms. Sans cette temporisation, le robot ne pourra pas répondre aux commandes envoyées.

« Set Tool Force » : Génère une force dans les différentes directions (X,Y,Z). Il y a donc trois paramètres pour les trois moteurs, de type réel.

« Stop » : Arrête l'envoi de courant dans les moteurs. Cette fonction doit précéder la fonction « Uninit Device » afin de ne pas déconnecter brusquement l'appareil.

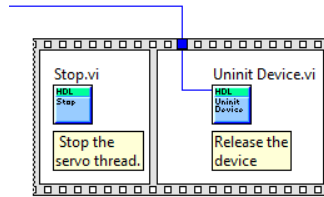
« Tool Button » : Indique l'état des boutons ; si l'on appuie sur un des 4 boutons (voir en fonction de l'accessoire connecté) renvoie un booléen à l'état haut, 0 dans le cas contraire.

« Tools Buttons » : Indique dans un tableau d'entiers (à convertir dans un tableau de booléens) l'état de chaque bouton :



« Tool Position » : Renvoie un cluster contenant les informations de position selon chaque axe (X,Y,Z). Il est à vérifier si l'on peut ou non envoyer la position désirée en entrée.

**« Uninit Device » : Etape finale et capitale de l'utilisation du robot : IL EST IMPERATIF D'APPELER CETTE FONCTION AVANT LA FIN DU PROGRAMME, DANS LE CAS CONTRAIRE LE PROGRAMME LABVIEW PLANTERA JUSQU'A CE QUE LE ROBOT SOIT REINITIALISE.** Il faut la paramétrer de la même façon que la fonction « make current ».



- Sources : - <http://gomezdegabriel.com/wordpress/2014/02/creating-a-library-for-novints-falcon-haptic-programming-with-labview/>
- <http://gomezdegabriel.com/wordpress/2014/02/start-programming-a-falcon-haptic-with-labview/>
  - <http://wenku.baidu.com/view/8a489537ee06eff9aef8078a.html> .

### 3.2.2.2 : Développement de l'environnement 3D avec CHAI3D

CHAI 3D propose un ensemble d'applications, de code source C++ open-source pour le développement d'environnements 3D. Il peut prendre des interfaces haptiques comme contrôle dont notre robot haptique.

Nous vous invitons à explorer les divers exemples qui ont été soigneusement conçus et documentés. Les informations détaillées au sujet des différentes classes et méthodes peuvent également être trouvées dans les documentations en ligne du site internet : [www.chai3d.org/](http://www.chai3d.org/)

Pour avoir accès aux projets ouvrir le dossier "chai3d-2.0.0-win32.zip".

## 3.3 Installation des drivers et paramétrage de la carte d'acquisition

Utiliser le programme d'installation "mccdaq.exe" fourni dans le CD-Rom. Le programme qui va installer les applications suivantes :

**Logiciel InstaCal 6.32** : logiciel d'installation, de calibrage, et de test,

**TracerDAQ 2.3** : Logiciel d'application virtuel d'instrument avec graphique à barres, Oscilloscope, Générateur de fonction, et générateur de taux,

**Universel 6.32** : Bibliothèques de programmes et composants pour développer les applications 32/64-bit, utilisant des langages de programmation de Windows,

**UL for Android™ 1.2.0** : Bibliothèque de programmes et exemples pour développer des applications pour l'usage avec des dispositifs Android, y compris les tablettes et les smartphones. Peut être utilisé avec Windows, Linux, ou Mac pour le développement de projet Android. Communique avec les dispositifs soutenus de MCC DAQ à partir de la version d'Android 3,1.

**ULx for NI LabVIEW™ 2.04** : Bibliothèque des instruments virtuels (force) et des programmes d'exemple employés avec les instruments nationaux LabVIEW pour développer l'instrumentation, l'acquisition, et les applications de contrôle. Compatible avec la version 8,5 en 32/64-bit LabVIEW jusqu'en 2013.

**UL for NI LabVIEW 7.11a** : Bibliothèque de legs des instruments virtuels (force) et des programmes d'exemple employés avec les instruments nationaux LabVIEW pour développer l'instrumentation, l'acquisition, et les applications de contrôle. Compatible avec la version à 32 bits 6.0-8.2.1 de LabVIEW.

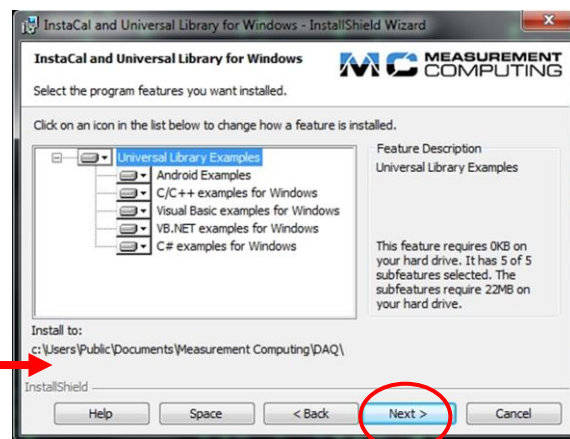
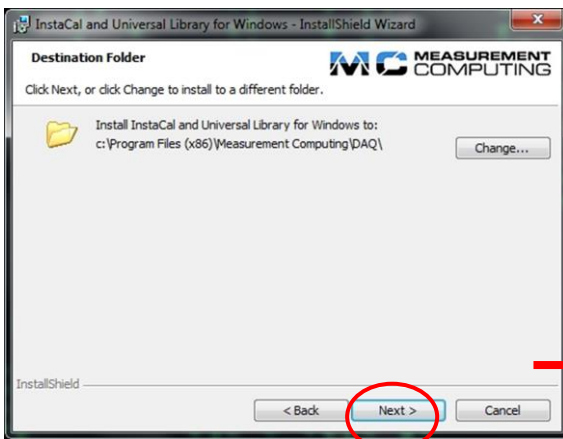
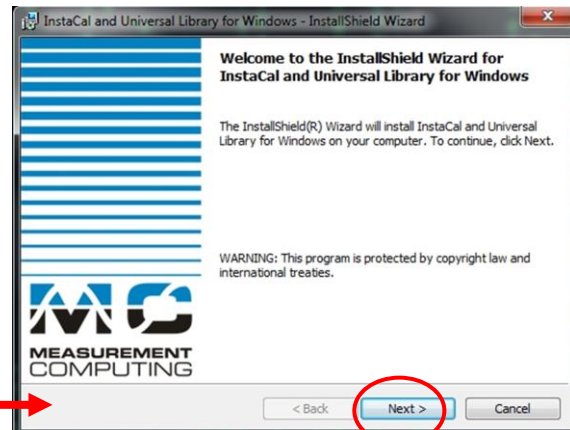
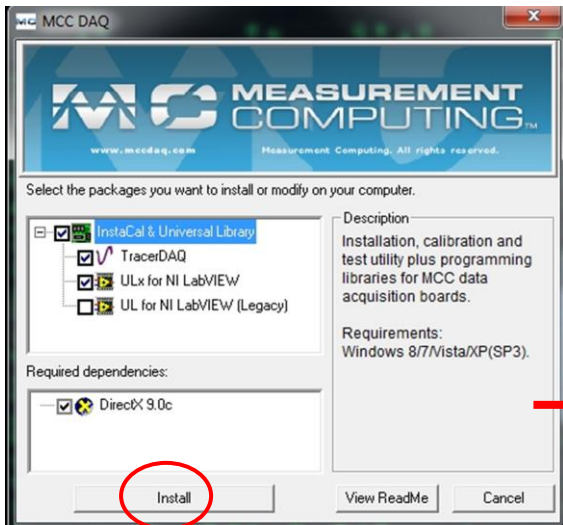
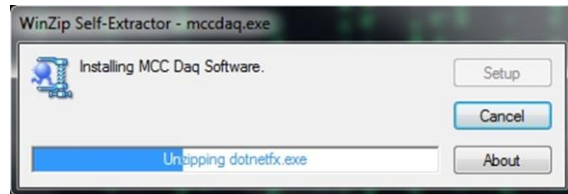
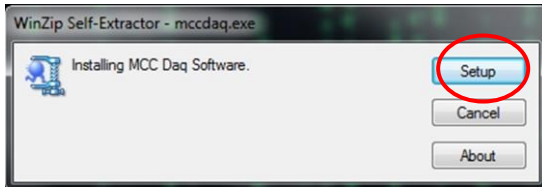
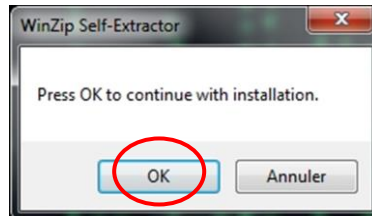


DirectX sera également installé.

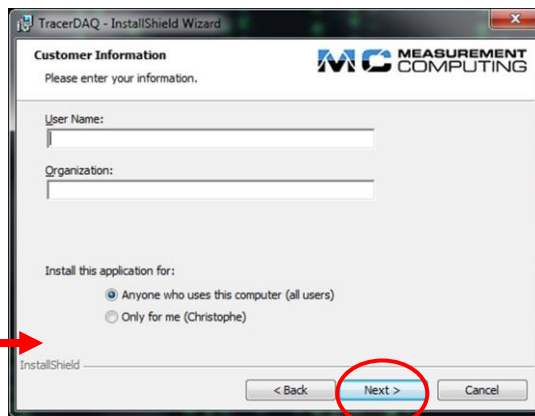
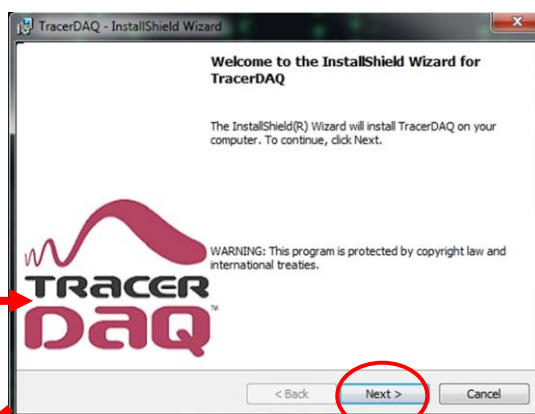
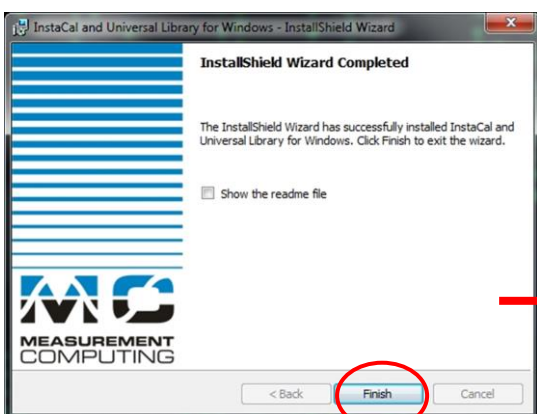
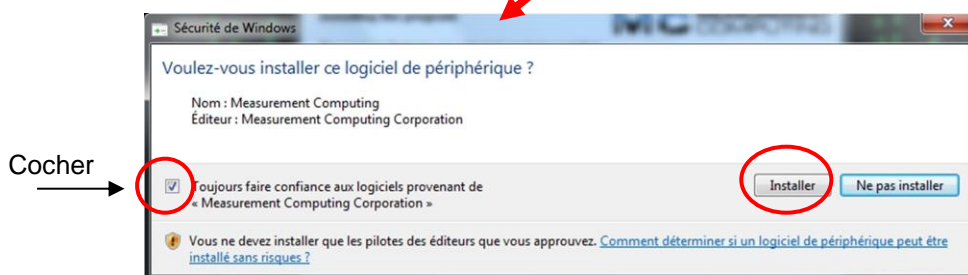
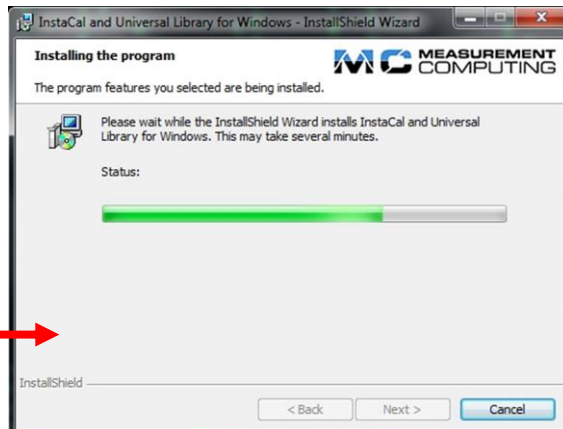
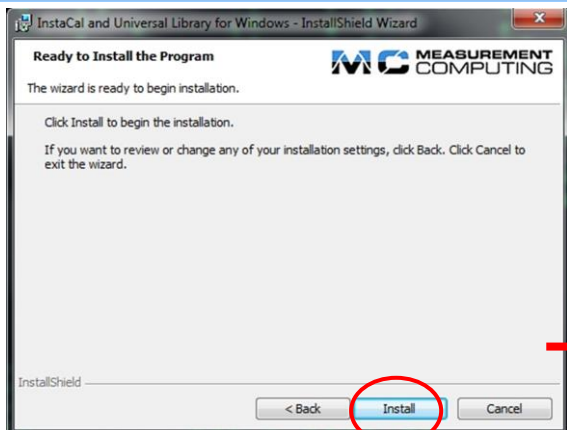
**Installation :**

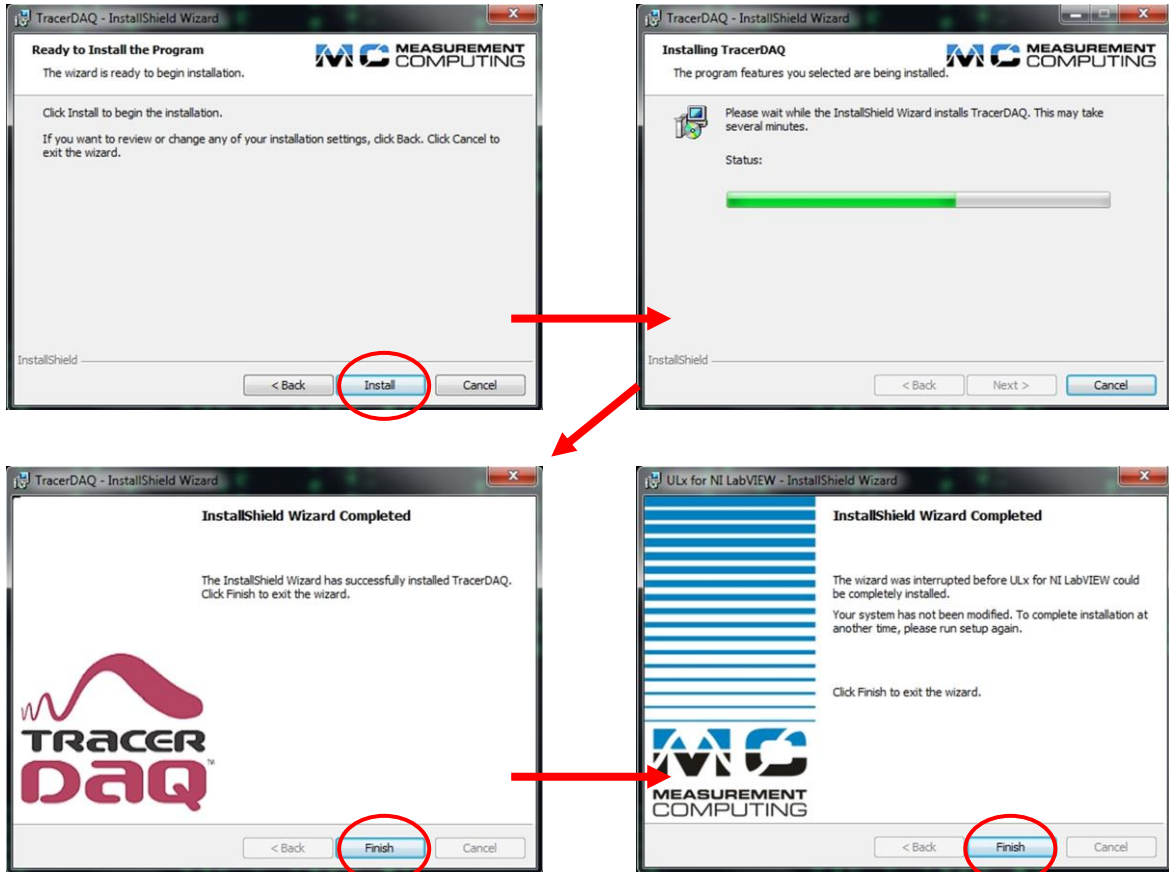
Ne pas connecter la carte d'acquisition sur l'ordinateur

Sur le CD-Rom exécuter le programme " mccdaq.exe" et suivre les instructions d'installation.

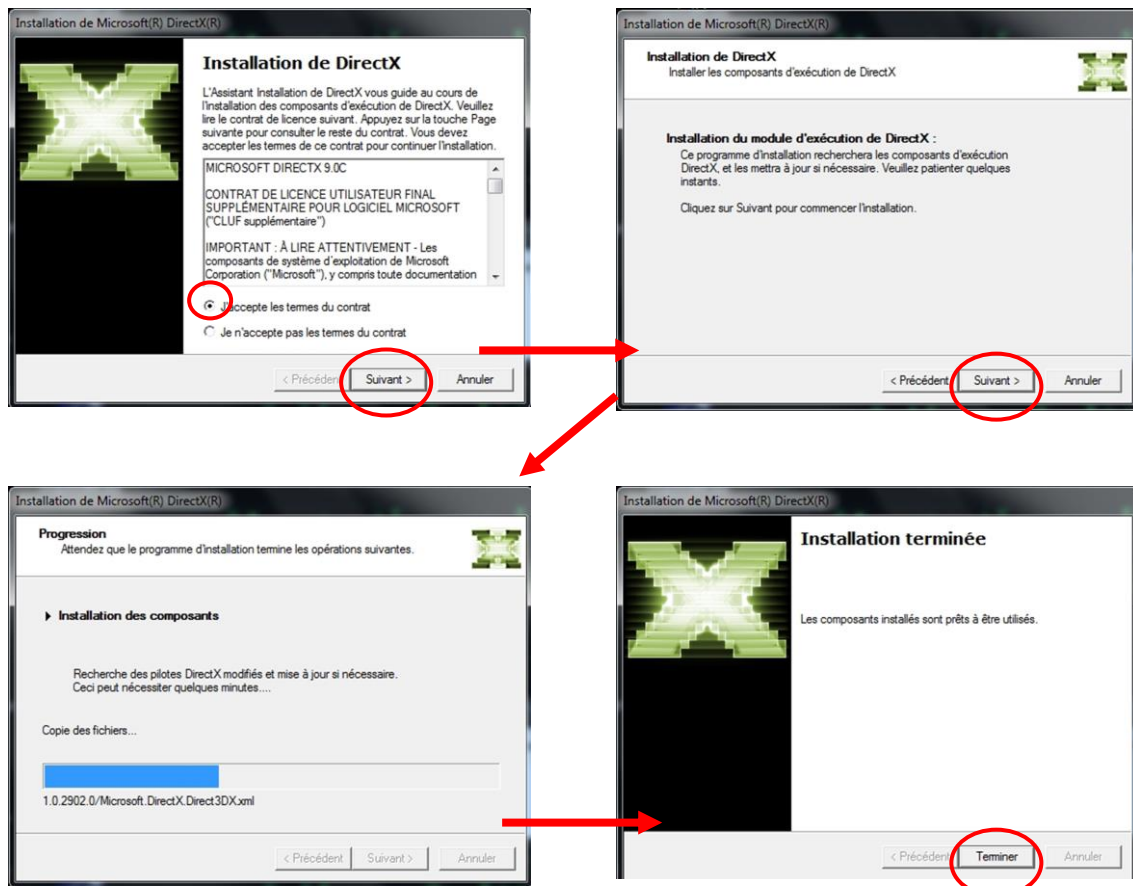


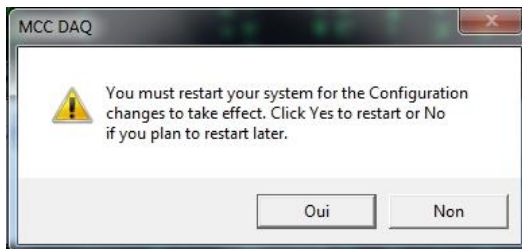






A la suite de l'installation des logiciels, le programme DirectX s'installe :





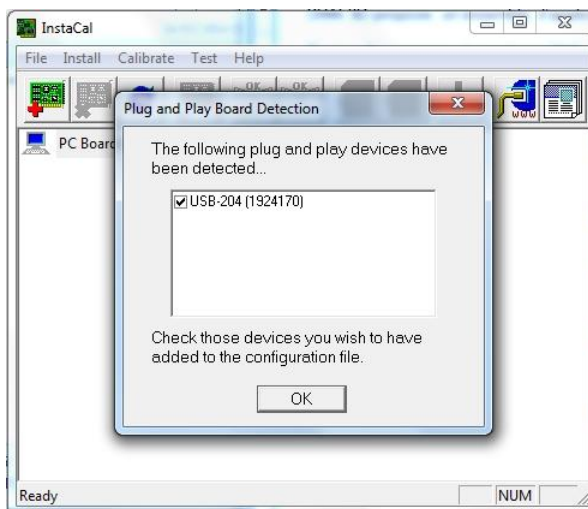
Redémarrer votre ordinateur et connecter la carte d'acquisition à l'ordinateur.

Votre ordinateur détecte le périphérique :



### Paramétrage du logiciel InstaCal :

Dans le menu démarrer de l'ordinateur, exécuter le logiciel "Instacal"



Votre ordinateur détecte la présence de la carte et finit l'installation.

### Paramétrage du logiciel Matlab :

**Remarque :** Pour utiliser la Data Acquisition Toolbox ou Sofstscope, vous devez disposer d'une version 32 bits de Matlab. En effet la toolbox ne fonctionne actuellement pas sur les versions 64 bits de Matlab. Une installation 32 bits et une 64 bits peuvent coexister sur un même PC avec la même licence.

Pour faire fonctionner la carte acquisition sur le logiciel Matlab, il est nécessaire de modifier deux fichiers "mwmcc.ini" pour les versions antérieure à Matlab 2014.

Le premier est placé dans matlab\toolbox\daq\daq\private  
Le second est placé dans matlab\toolbox\daq\daq\src

Nous vous conseillons de faire une copie de ces deux fichiers avant de les modifier.



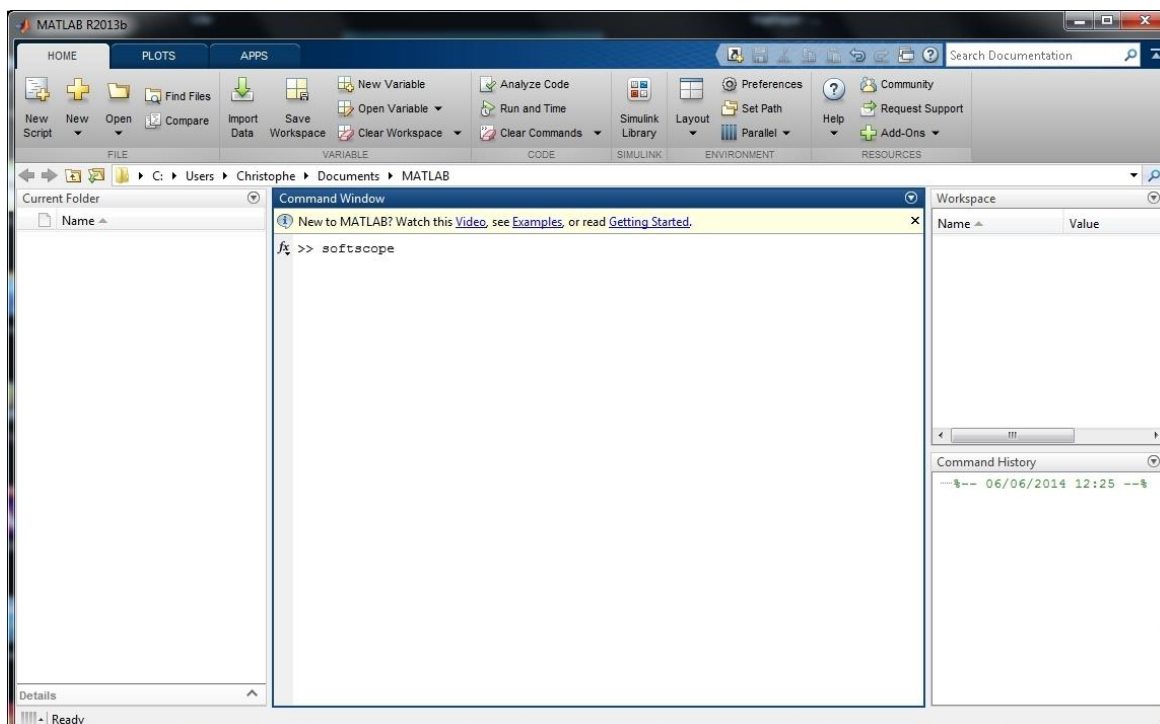
Ouvrir les fichiers et rajouter à la suite :

```
; Device listed twice, under old and new name  
[USB-204]  
ID = 0x0114  
Bus = USB  
AIResolution = 12  
AIClockingType = 0  
AIMaxSR = 500000  
AIMaxAggregateContSR = 500000  
AIMaxSingleChanContSR = 625000  
AIMinSR = 0.016  
AIBurstMode = 0  
PacketSize = 32  
SettleTime = 125  
Scanning = 1  
AnalogTrig = 0  
DigitalTrig = 0  
//AISupportedTriggerConditions = 0 ; (11000000000000)  
ChanGainQueue = 8  
ChansContigAscending = 1  
AIFIFO = 61440  
SEInputRanges = 1  
DIOConfigurable = 1  
DPORTS = 1  
DPORT0LINES = 8  
DPORT0TYPE = 1  
DPORT0CONFIG = 3
```

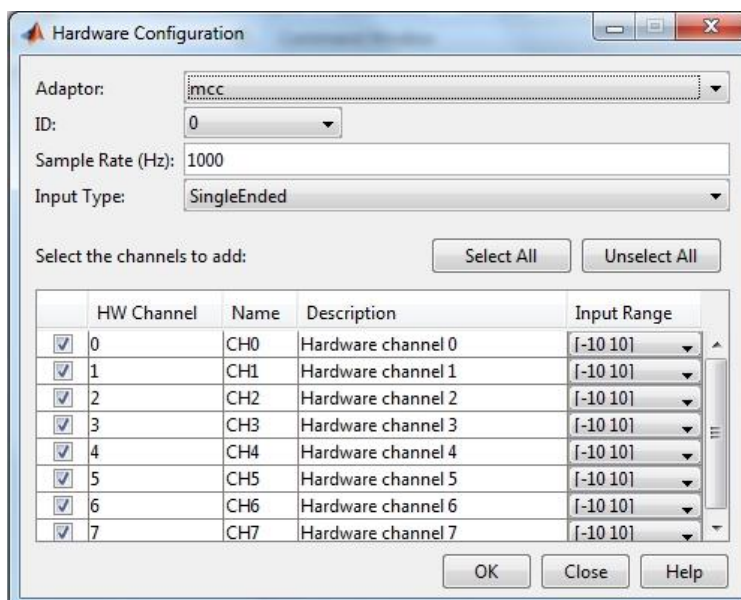
Enregistrer les deux fichiers modifiés et lancer le logiciel Matlab.



Dans f(x), taper la commande "softscope"



La fenêtre suivante s'ouvre :



La configuration de MATLAB est terminée.

La carte est bien reconnue sous MATLAB, vous pouvez utiliser la carte dans Matlab notamment grâce à la Data Acquisition Toolbox.

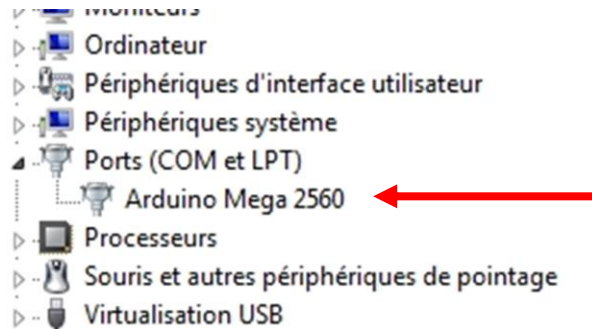




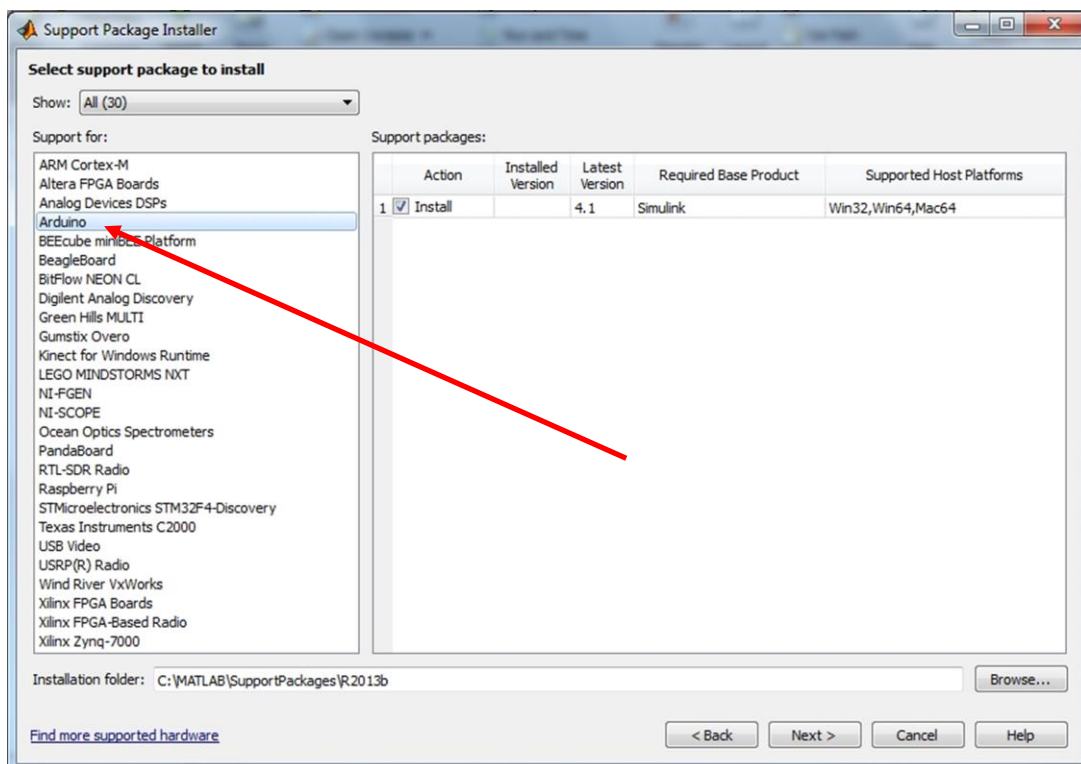
### 3.4 Installation de la carte ARDUINO dans le logiciel MATLAB (R2012 et >)

Si cela n'est pas déjà le cas, veuillez installer le logiciel "Arduino" sur le site [www.arduino.cc/fr/](http://www.arduino.cc/fr/)

Connecter la carte ArduinoMega sur votre ordinateur et vérifier que la carte a bien été reconnue.



Ouvrir "Matlab",  
 Dans f(x), taper la commande "supportPackageInstaller",  
 Sélectionner installer à partir d'internet.



Dans l'onglet de gauche choisir Arduino comme paquet à installer.

Pour continuer vous devez vous connecter avec votre identifiant et votre mot de passe.

Suivre les instructions de téléchargement. Lorsque ce dernier est terminé le logiciel Matlab est prêt.