

MANUEL D'UTILISATION
MODULE USB
8 ENTrees + 6 SORTIES + CODEUR INCREMENTAL
TC/USB-16IO



Electronique
Informatique
Industrielle



Révision document : 22a
Révision module: 01

3C Concept S.A.S – 17 rue d'Ahuy, 21121 HAUTEVILLE LES DIJON
Tel : 0380.550.130 - Fax : 0380.556.449
www.3cconcept.fr

Sommaire

DESCRIPTION.....	3
SPECIFICATIONS	3
CARACTERISTIQUES ELECTRIQUES	4
DISPOSITION DES ENTREES-SORTIES	6
SCHEMA DE PRINCIPE DES ENTREES ET DES SORTIES	8
PROCEDURE D'INSTALLATION SOUS WINDOWS XP	9
PROCEDURE D'INSTALLATION SOUS WINDOWS 2000	13
PROCEDURE D'INSTALLATION SOUS WINDOWS 7	17
PROCEDURE D'INSTALLATION SOUS WINDOWS 8	23
PROCEDURE D'INSTALLATION SOUS WINDOWS 10	28
INSTALLATION SOUS WINDOWS 10 POUR LES MODULES ACQUIS A PARTIR DE FEVRIER 2019 OU MIS A NIVEAU VIA TCUSB_SIGN.EXE.....	33
EN CAS DE PROBLEME	37
PROGRAMME DE DEMONSTRATION	39
UTILISATION DE LA DLL 32BITS TCUSB16IO.DLL.....	41
UTILISATION DE LA DLL 64BITS TCUSB16IOX64.DLL	42
DESCRIPTION DES FONCTIONS DE LA DLL.....	43
F.A.Q	48

Description

Le module TC/USB-16IO est un module d'interface pour PC, comportant 8 entrées logiques isolées, 2 sorties sur contacts secs par relais NO, 4 sorties sur opto-coupleurs et un compteur pour codeur incrémental. Auto-alimenté par le port USB, ce module permet d'obtenir une grande variété d'entrées-sorties pour un coût réduit et une grande facilité d'installation. Un chien de garde débrayable assure la coupure de la commande des sorties si une séquence particulière n'est pas exécutée toutes les 2 secondes. Le module se présente dans un boîtier ABS de 100 x 120mm, il se fixe sur un rail DIN 35 mm et ses connecteurs sont débrochables améliorant ainsi la facilité des opérations de maintenance.

Spécifications

- 8 entrées ToR opto-isolées, 12 à 24 Vdc 10 mA Max avec compteurs d'évènements 8 bits sur chaque entrée et possibilité de latch du compteur de position sur les deux dernières.
- Filtrage passe-bas 100 Hz sur les entrées
- 2 sorties ToR par relais sur contact sec NO (48V 1A max)
- 4 sorties ToR par opto-coupleurs (48V 30mA max)
- 3 entrées A/B/Z pour codeur incrémental en quadrature (configurables par cavaliers en 5V ou 12/24V), elles peuvent être utilisées en entrées Tor ordinaires.
- Compteur de position sur 32 bits.
- 2 latches du compteur de position sur 32 bits.
- Chien de garde débrayable
- Isolation galvanique entre l'USB et les entrées / sorties : 1500Vrms
- Compatible USB 1.1 et 2.0 (12 Mbits/s)
- Auto-alimentation par l'USB (150 mA max)
- Montage sur rail DIN 35mm
- Dimensions : (L) 100mm x (l) 23mm x (P) 120mm

Caractéristiques électriques

Caractéristiques générales :

Nom	Description	Min	Typ	Max	Unités
Iusb	Courant sur la ligne USB	40		150	mA

Entrées 24 V :

Nom	Description	Min	Typ	Max	Unités
Rin	Impédance de charge		2200		ohms
Vmax	Tension limite admissible sur les entrées	-30		30	volts
Ie	Courant d'entrée pour un 1 logique	3	5	15	mA
Vih	Tension d'entrée pour 1 logique	10		30	volts
Vil	Tension d'entrée pour 0 logique		0	2	volts

Entrées Codeur 5V :

Nom	Description	Min	Typ	Max	Unités
Rin	Impédance de charge		470		ohms
Vmax	Tension limite admissible sur les entrées	-7		12	volts
Ie	Courant d'entrée pour un 1 logique	3		15	mA
Vih	Tension différentielle pour 1 logique	3	5	12	volts
Vil	Tension différentielle pour 0 logique		0	1	volts

Sorties relais :

Nom	Description	Min	Typ	Max	Unités
Vmax	Tension maximale admissible			48	volts
Ismax	Courant maximum admissible			1000	mA

Sorties opto-coupleurs :

Nom	Description	Min	Typ	Max	Unités
Vmax	Tension maximale admissible			48	volts
Ismax	Courant maximum admissible			30	mA

Caractéristiques dynamiques :

Nom	Description	Min	Typ	Max	Unités
Tre	Temps de réponse des entrées		1	10	msec
Trsr	Temps de réponse des sorties relais		6	12	msec
Nbcy	Nombre de manœuvres des sorties relais*		10 ⁶		Cycles
Trso	Temps de réponse des sorties opto-coupleurs		100	300	µsec
Twdg	Temps de réponse du chien de garde		2		Sec

* : sur charges résistives

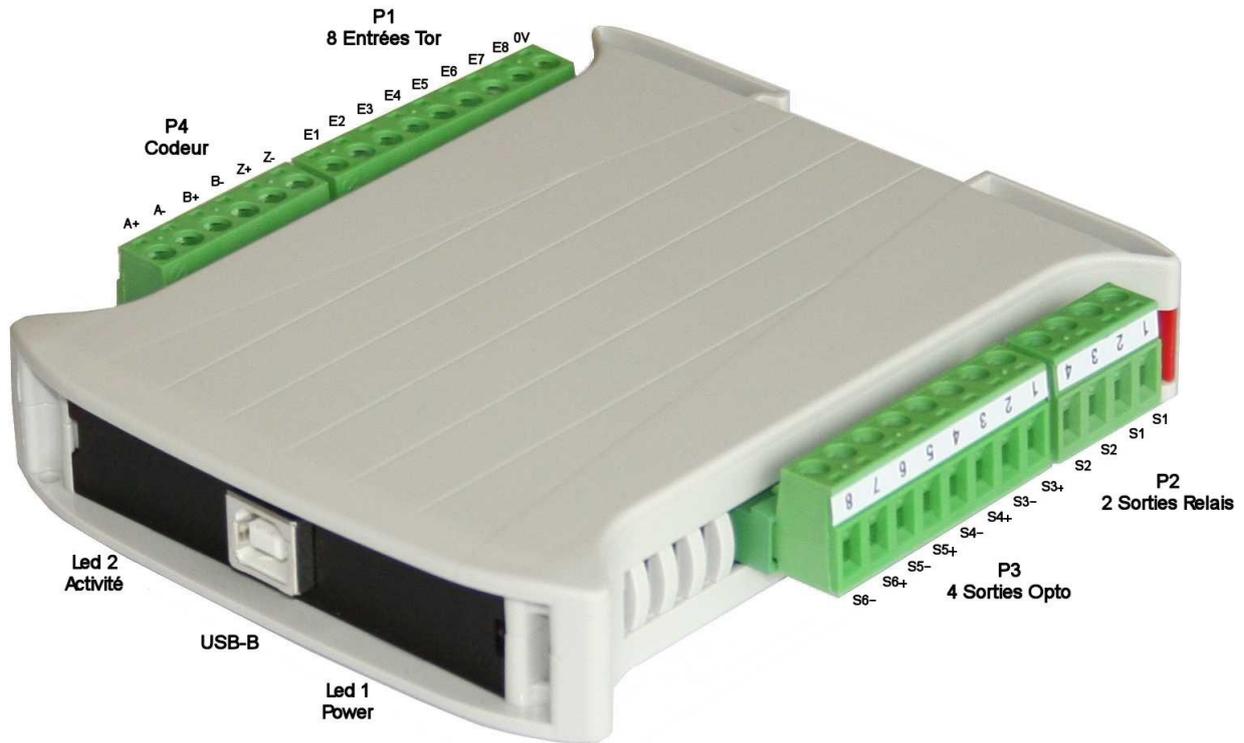
Spécifications USB :

- Nombre maxi de périphériques = 127
- Distance maxi entre périphériques = 5 m
- Possibilité d'utiliser des hubs USB
- Possibilité d'étendre les distances (exemple extremeUSB(R) www.icron.com)
- Hub industriel USB (4 ou 7 ports) voir hubport sur www.digi.com

Environnement :

- Température d'utilisation : 0°C à 55°C
- Température de stockage : -25°C à 70°C
- Humidité : de 5% à 90%, sans condensation
- Degré de pollution maximum : 2 (selon IEC 60664)
- Altitude maximale d'utilisation : 2000m
- Usage intérieur exclusivement

Disposition des entrées-sorties



Description des Leds :

Led 1 : Allumée fixe, témoin d'alimentation par la ligne USB

Led 2 : Clignotante, Indique le bon fonctionnement du processeur interne du module.

P1 : Connecteur 9 bornes - 8 Entrées Tor

Borne	Signal
1	E1
2	E2
3	E3
4	E4
5	E5
6	E6
7	E7 / Latch 1
8	E8 / Latch 2
9	0V

P2 : Connecteur 4 bornes - 2 sorties relais

Borne	Signal
1	REL S1
2	REL S1
3	REL S2
4	REL S2

P3 : Connecteur 8 bornes - 4 sorties opto-coupleurs

Borne	Signal
1	OPTO S3+
2	OPTO S3-
3	OPTO S4+
4	OPTO S4-
5	OPTO S5+
6	OPTO S5-
7	OPTO S6+
8	OPTO S6-

ⓘ Attention : Ces sorties sont polarisées, respecter le sens de câblage (voir page suivante).

P4 : Connecteur 6 bornes - Entrées codeur

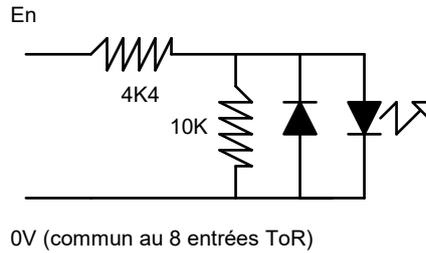
Borne	Signal
1	A+
2	A-
3	B+
4	B-
5	Z+
6	Z-

Configuration des entrées Codeurs :

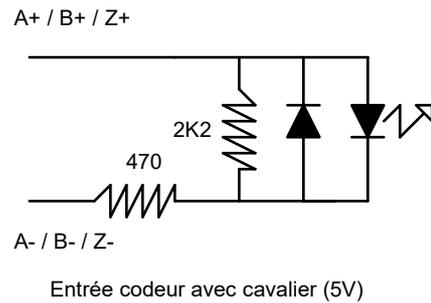
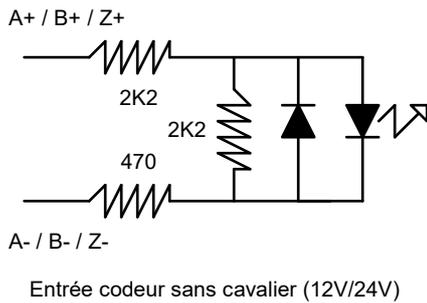
Les 3 cavaliers présents à côté du connecteur P4 permettent de configurer les entrées codeurs en 5V (cavaliers présents) ou en 12/24V (cavaliers retirés).

Schéma de principe des entrées et des sorties

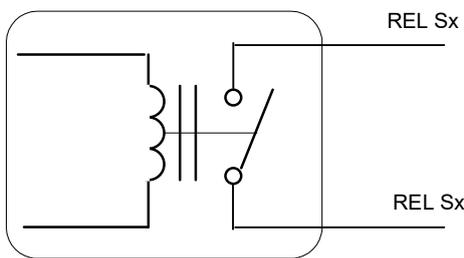
Entrées ToR :



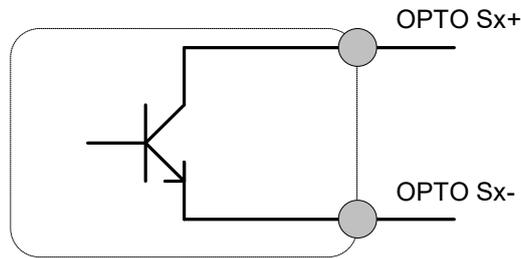
Entrées codeur :



Sorties relais :



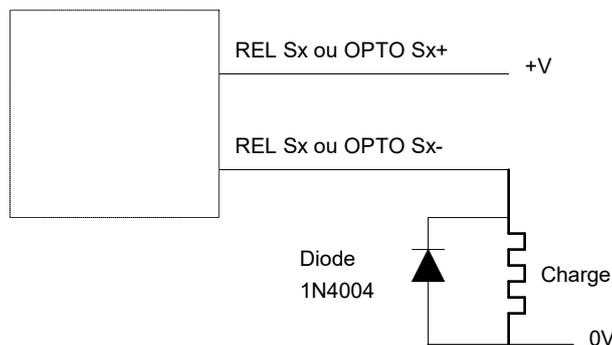
Sorties opto-coupleur :



:

Protection des sorties relais et opto-coupleur :

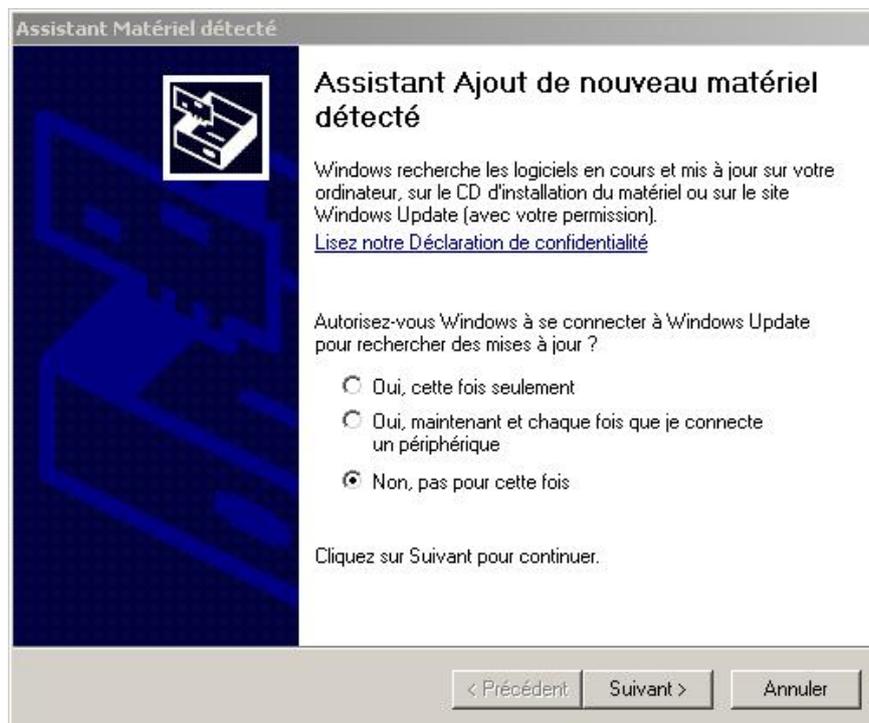
Dans le cas d'utilisations de charge inductive (relais, électrovanne...) il conviendra de placer au borne de celle-ci une diode de roue libre afin d'éviter la détérioration de la sortie du module.



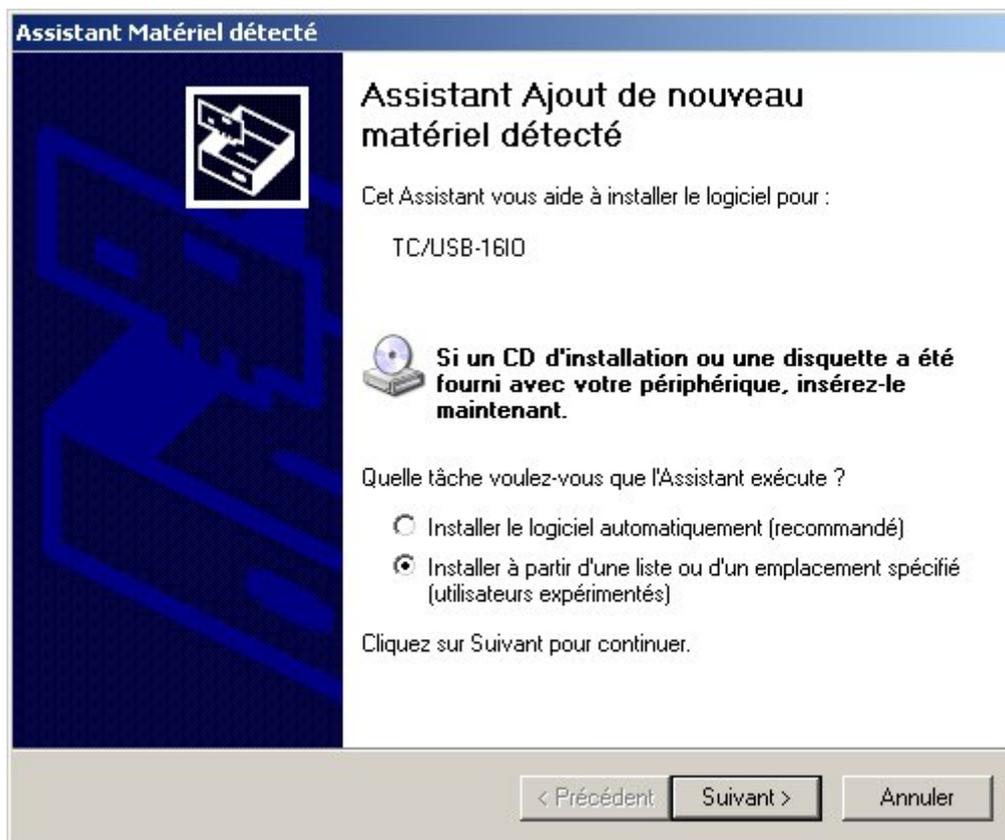
Procédure d'installation sous Windows XP

Le contenu du CD-ROM rassemble les drivers, le programme de démonstration, ainsi qu'une DLL (32 ou 64bits) et les fichiers d'entête pour concevoir ses propres programmes.

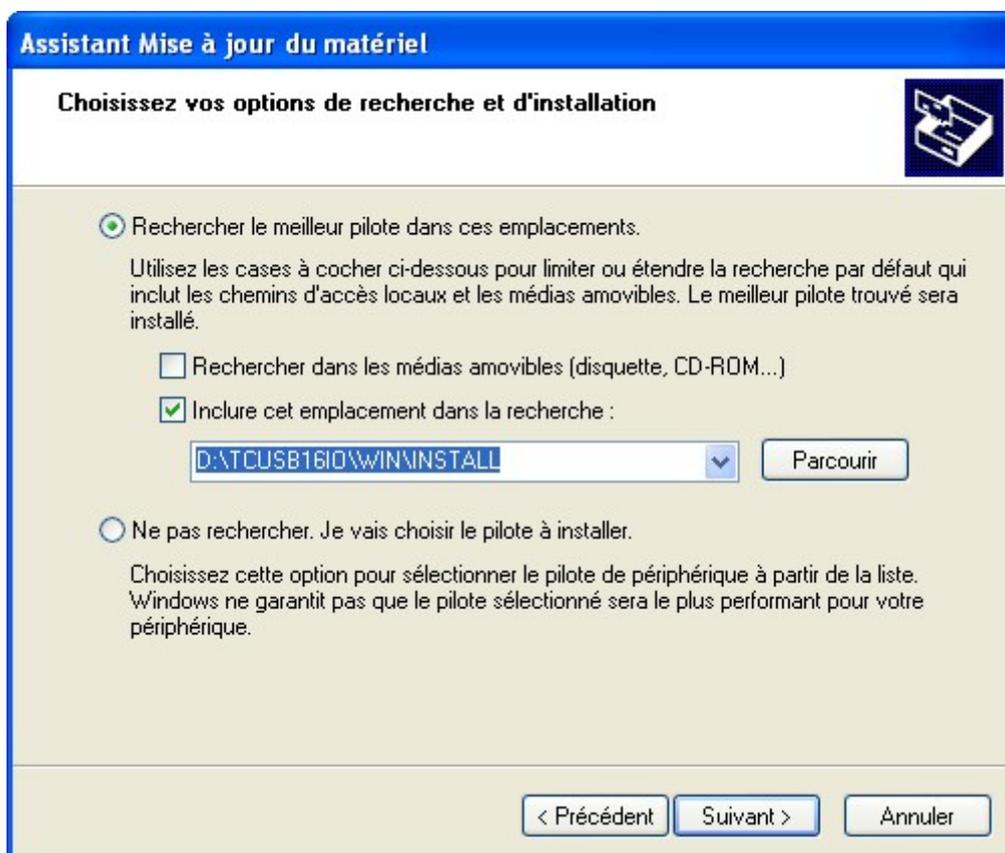
- Branchez le module au PC, Windows doit signaler la présence d'un nouveau périphérique :



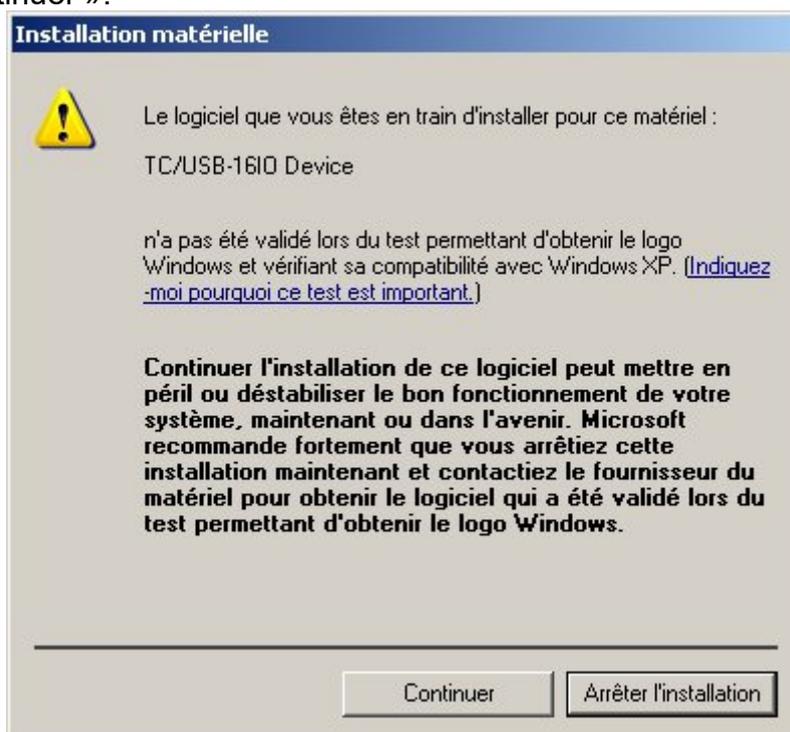
- Windows vous propose éventuellement de se connecter au site Windows Update, Sélectionner « non pas pour cette fois »
- Dans la fenêtre suivante, sélectionner « installer à partir... » , cliquez sur Suivant .



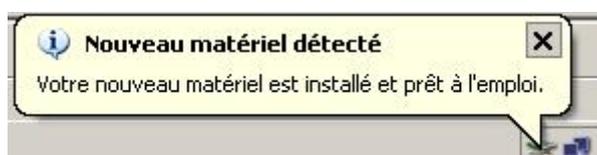
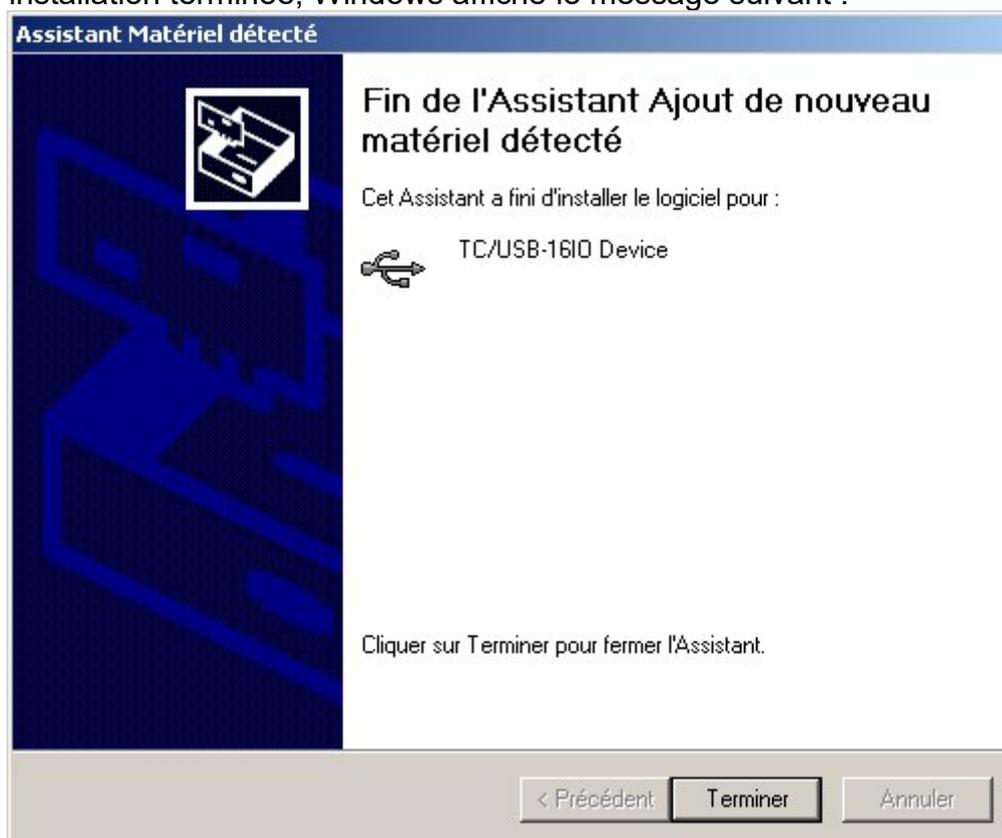
- Dans la fenêtre suivante, cliquez sur « parcourir », cherchez sur le CDROM fournit le répertoire **TCUSB16IO\WINxx\INSTALL**, cliquez sur Ok.



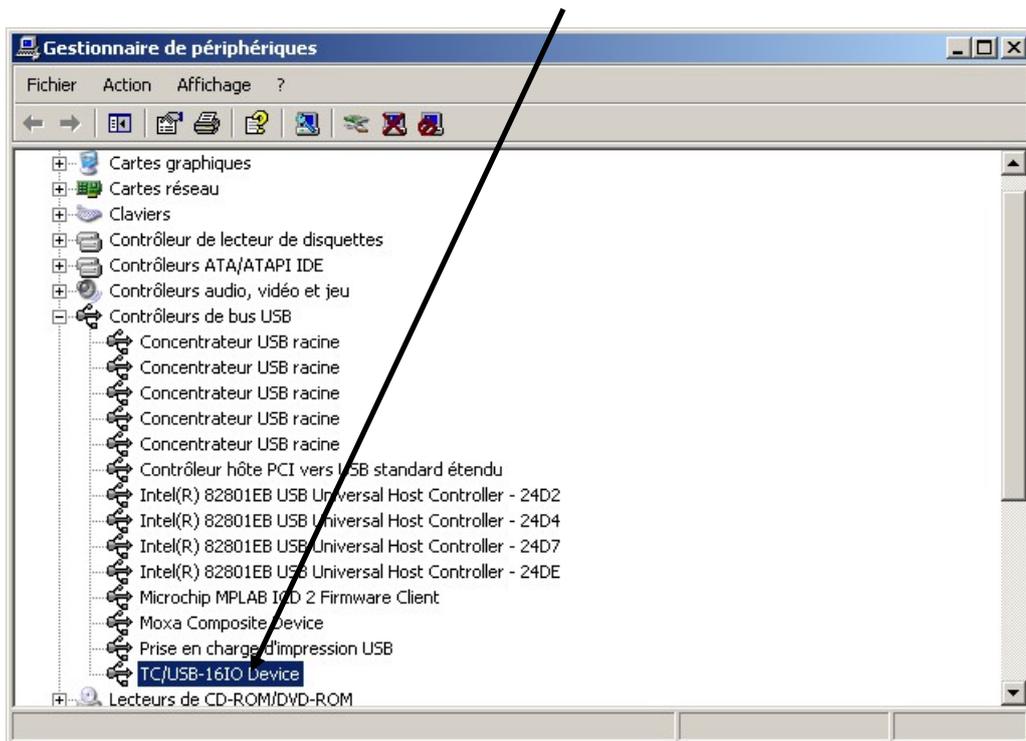
Lors de l'installation et le transfert de fichiers, WindowsXP indiquera que le pilote en cours d'installation n'est pas signé WHQL (ie :validé par Microsoft), c'est normal, veuillez cliquer sur « continuer ».



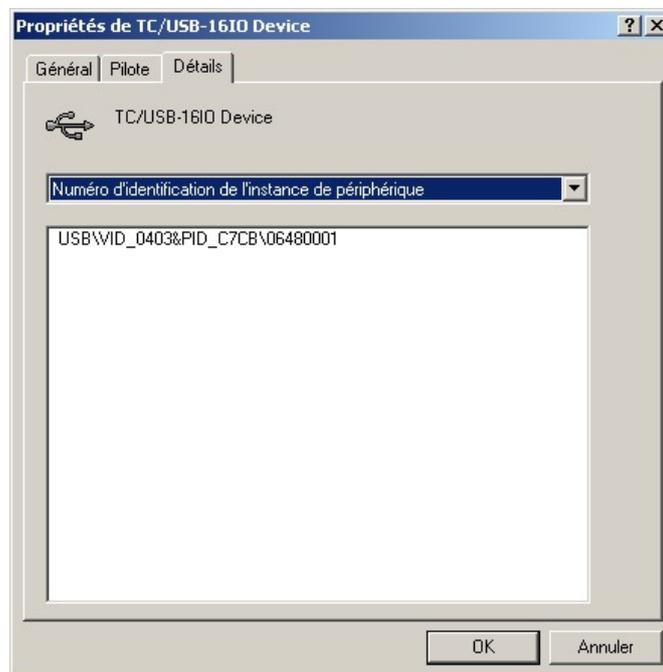
Une fois l'installation terminée, Windows affiche le message suivant :



A titre de vérification vous pouvez aller dans le panneau de configuration , système, gestionnaire de périphérique et vérifier ce qui suit :



De plus, vous pouvez retrouver le numéro de série du module en cliquant sur « propriétés » de TC/USB-16IO, onglet Détails



En dernier lieu, lancez le programme **TCUSB16IO\WINxx\DEMO\DemoTCUSB16IO.EXE** présent sur le CD-ROM pour vérifier que l'installation s'est bien déroulée et que le module est accessible. (voir partie Programme de démonstration)

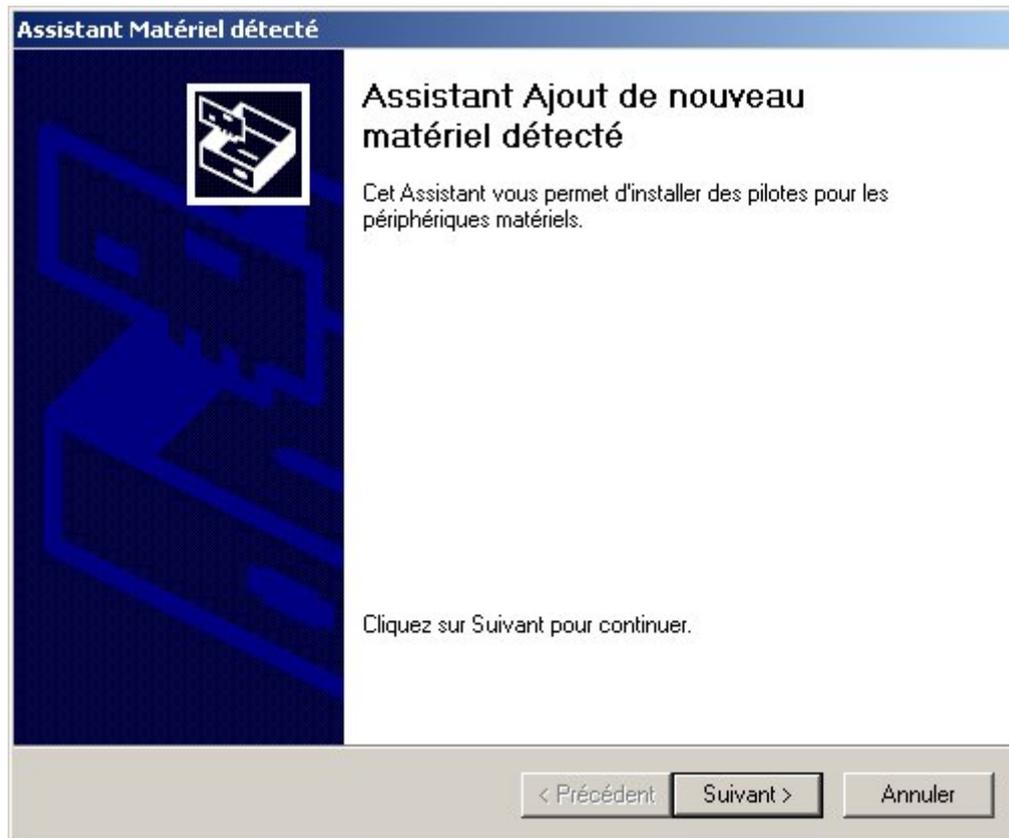
Procédure d'installation sous Windows 2000

Le contenu du CD-ROM rassemble les drivers, le programme de démonstration, ainsi qu'une DLL et les fichiers d'entête pour concevoir ses propres programmes.

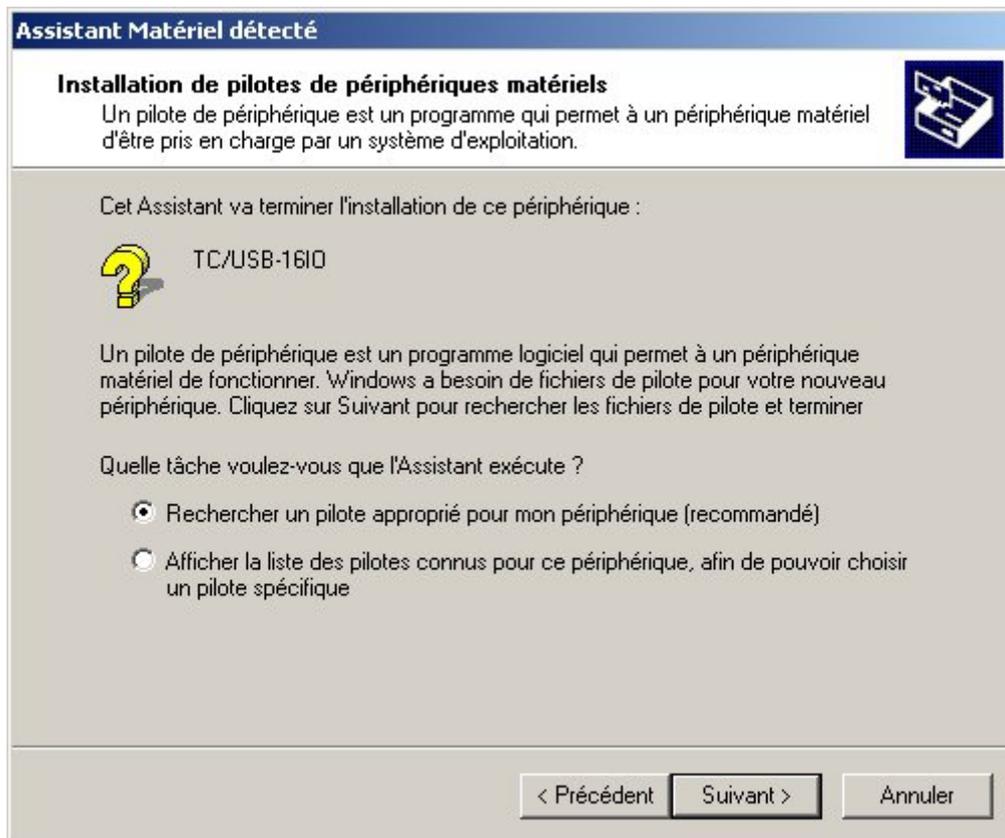
- Brancher le module au PC: Windows détecte alors l'apparition d'un nouveau périphérique :



- Dans la fenêtre suivante, cliquez sur Suivant .



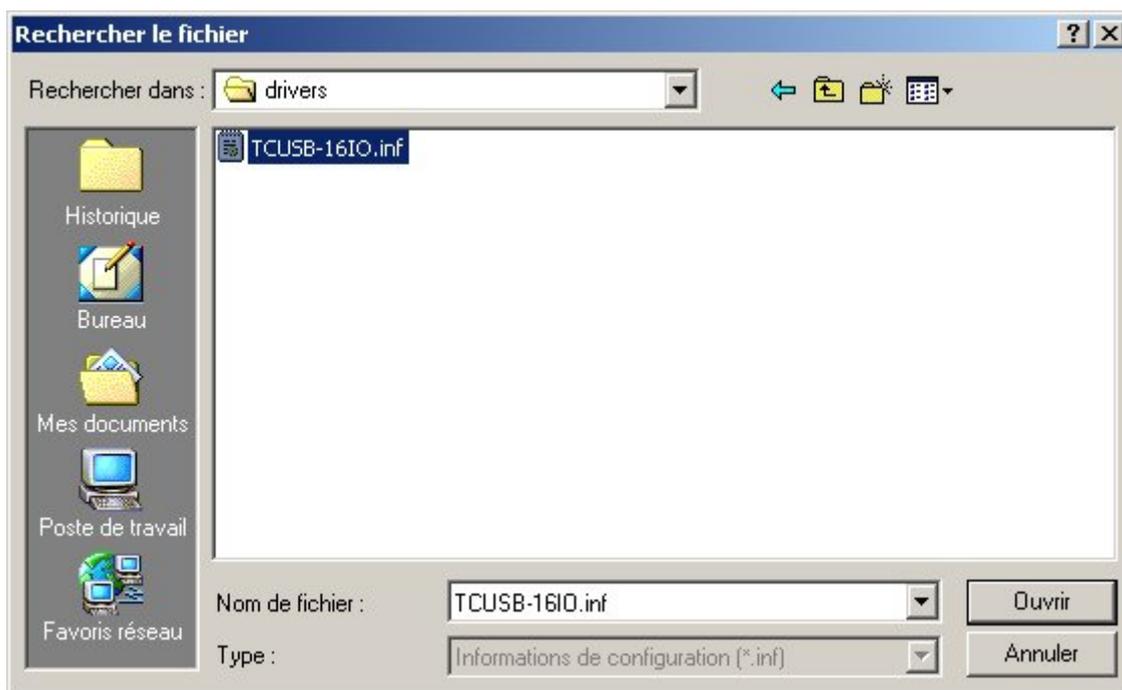
- Dans la fenêtre suivante, Cochez « Rechercher un pilote ... » , cliquez sur Suivant



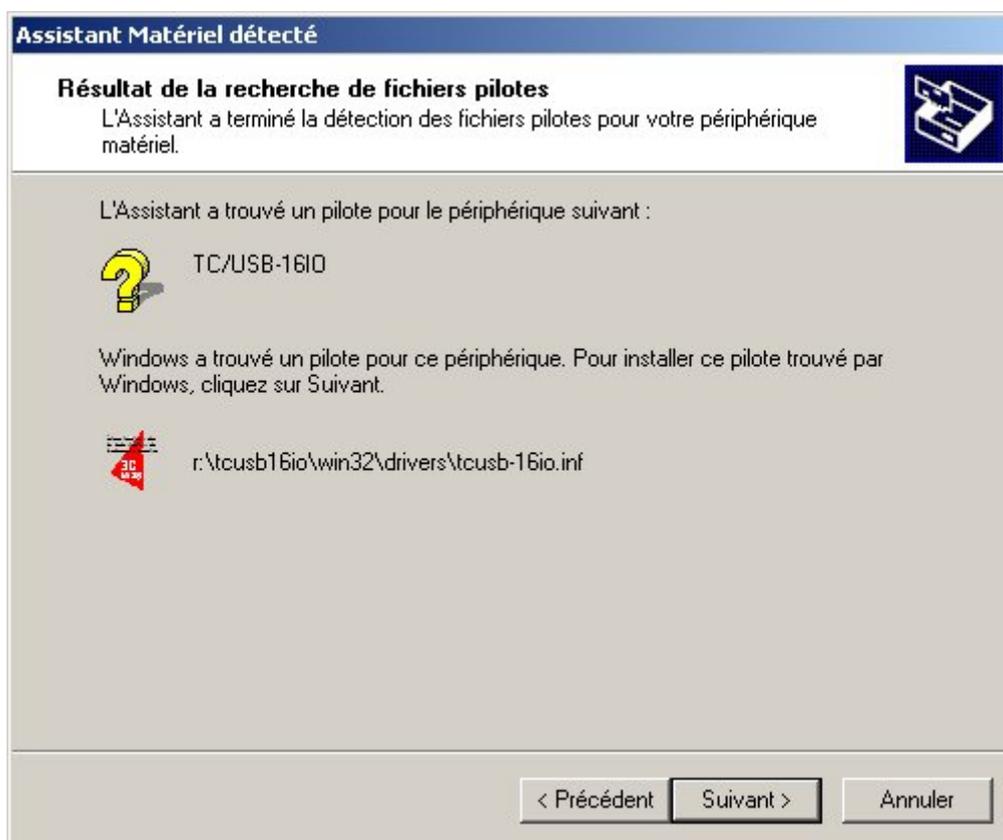
- Windows vous propose différents emplacement de recherche, veuillez sélectionner emplacement spécifique et cliquez sur Suivant



- Dans la fenêtre suivante, cliquez sur Parcourir, cherchez sur le CDROM fournit le répertoire **TCUSB16IO\WINxx\INSTALL\TCUSB.INF**, cliquez sur Ouvrir.



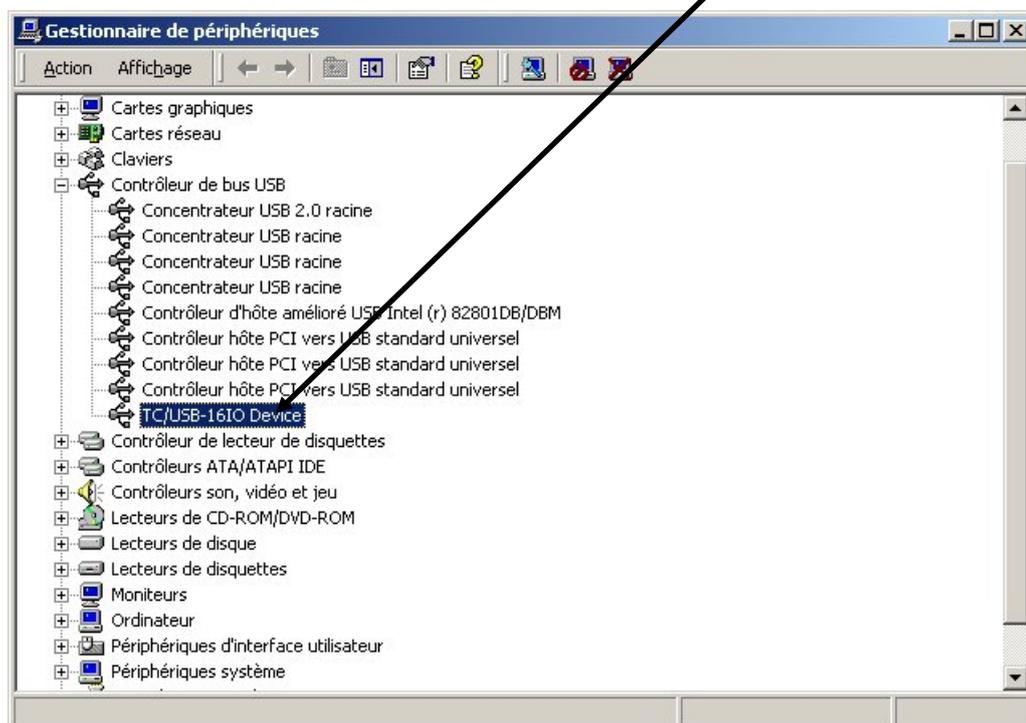
- Windows affiche un résumé des informations récoltées, cliquez sur Suivant



- Lorsque l'installation du driver est terminée, windows affiche l'écran suivant :



A titre de vérification vous pouvez aller dans le panneau de configuration , système, onglet Matériel, Gestionnaire de périphérique et vérifier ce qui suit :

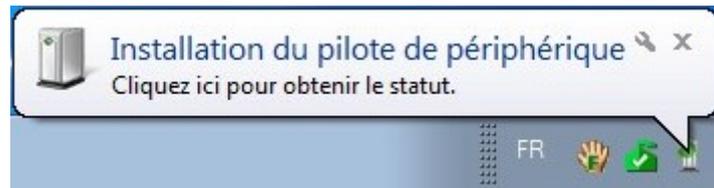


En dernier lieu, lancez le programme **TCUSB16IO\WINxx\DEMO\DemoTCUSB16IO.EXE** présent sur le CD-ROM pour vérifier que l'installation s'est bien déroulée et que le module est accessible. (voir partie Programme de démonstration)

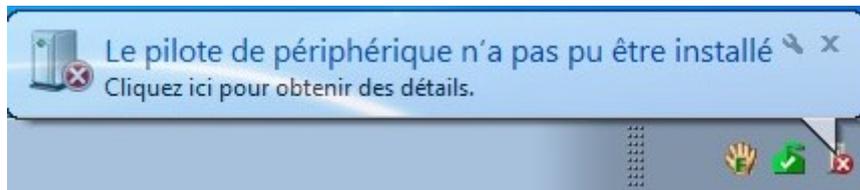
Procédure d'installation sous Windows 7

Le contenu du CD-ROM rassemble les drivers, le programme de démonstration, ainsi qu'une DLL (32 ou 64bits) et les fichiers d'entête pour concevoir ses propres programmes.

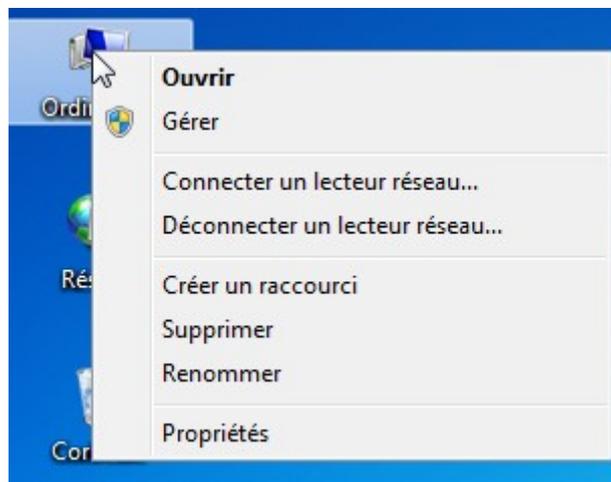
- Branchez le module au PC, Windows doit signaler la présence d'un nouveau périphérique :



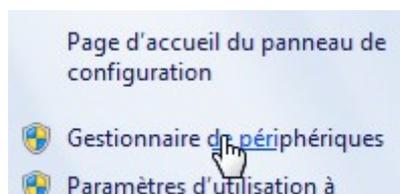
- Le driver n'étant pas installé automatiquement par Windows, celui-ci signale une erreur d'installation.



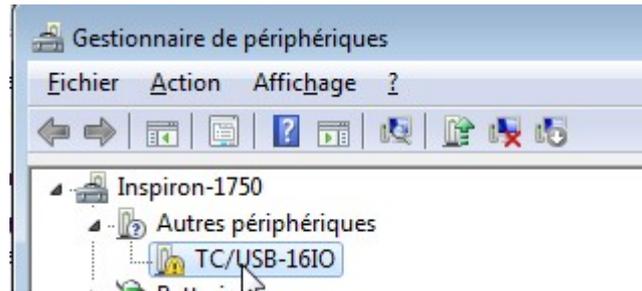
- Faites un clic droit sur Ordinateur et cliquez sur propriétés dans la fenêtre qui s'ouvre



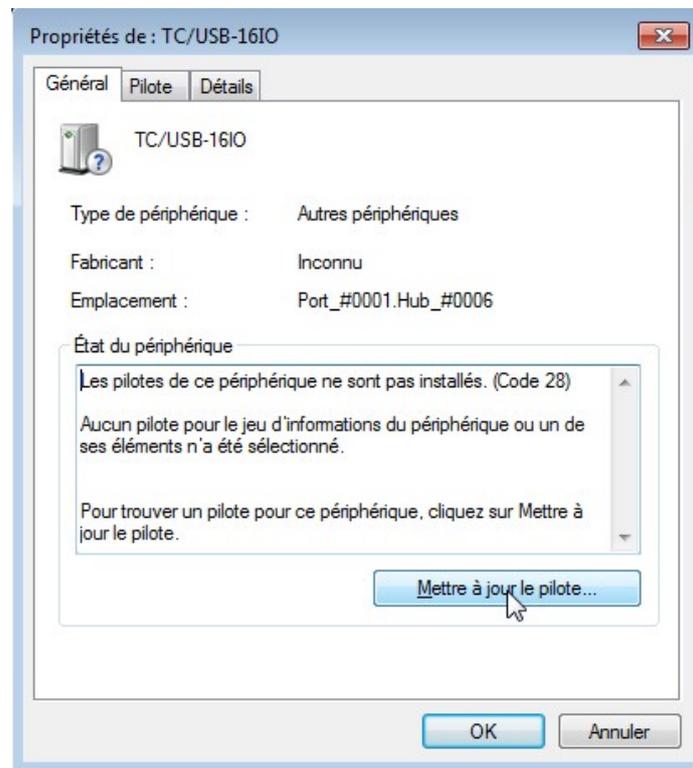
- Windows affiche alors le panneau de configuration système.
- Sur la gauche de la fenêtre cliquez sur Gestionnaire de périphériques



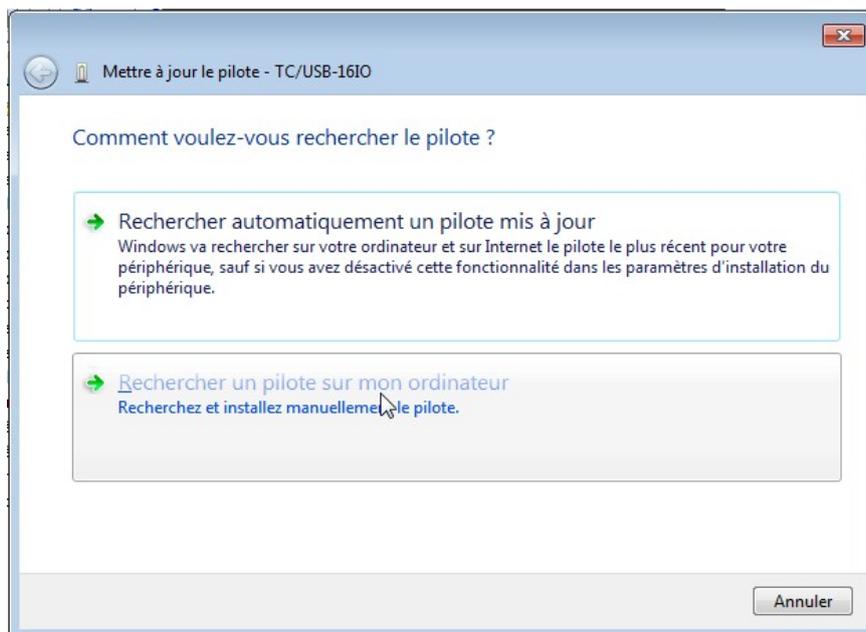
- La fenêtre suivante doit alors apparaître.



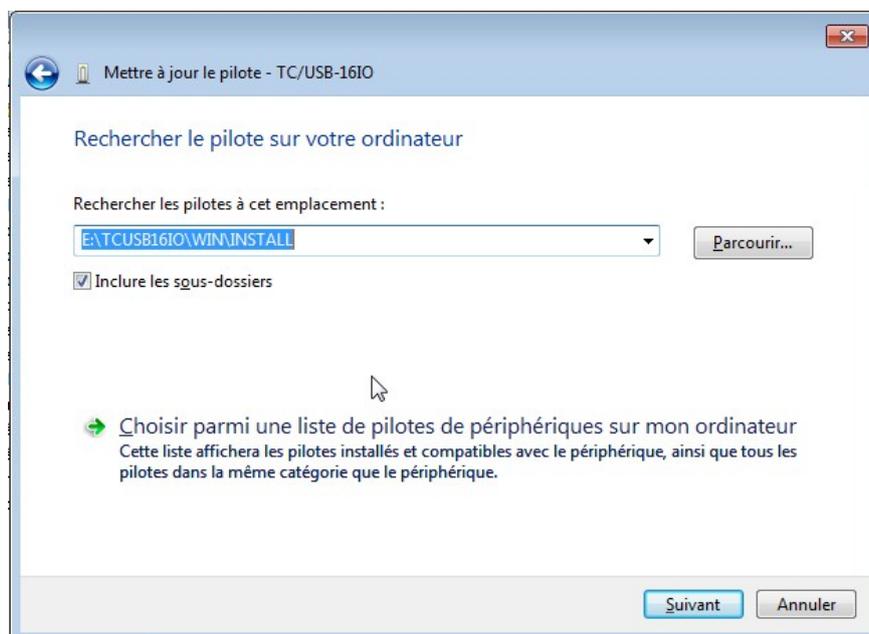
- Un point d'exclamation jaune indique que le module TCUSB1610 a été détecté mais que son driver n'est pas correctement installé ou est inexistant.
- Cliquer deux fois sur ce dernier



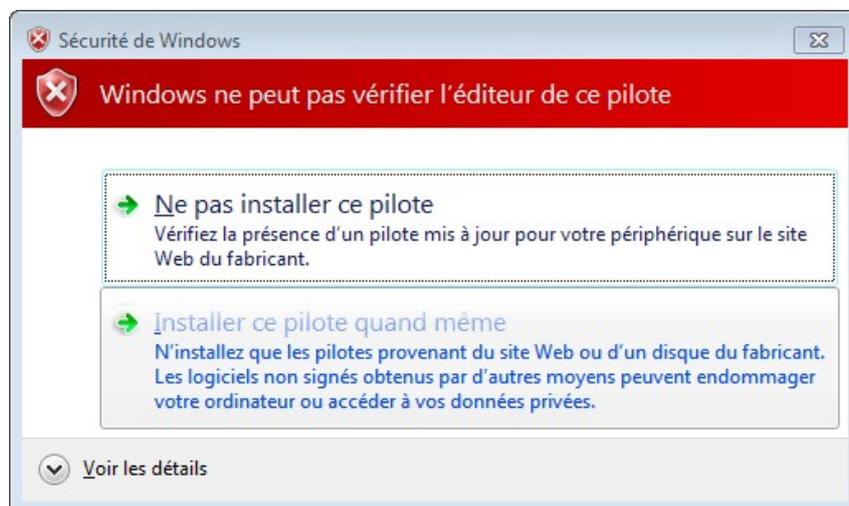
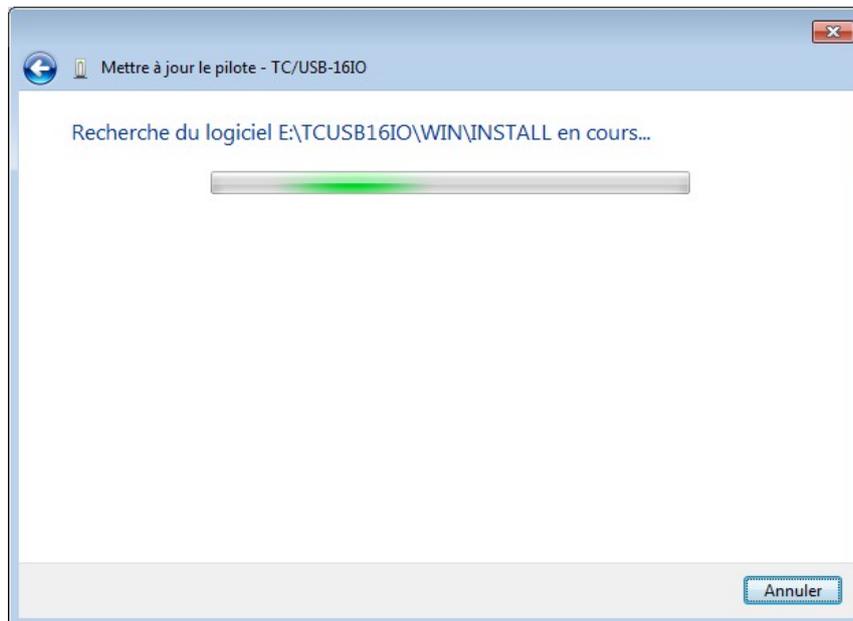
- Cliquer sur mettre à jour le pilote



- Cliquer sur recherchez un pilote sur mon ordinateur

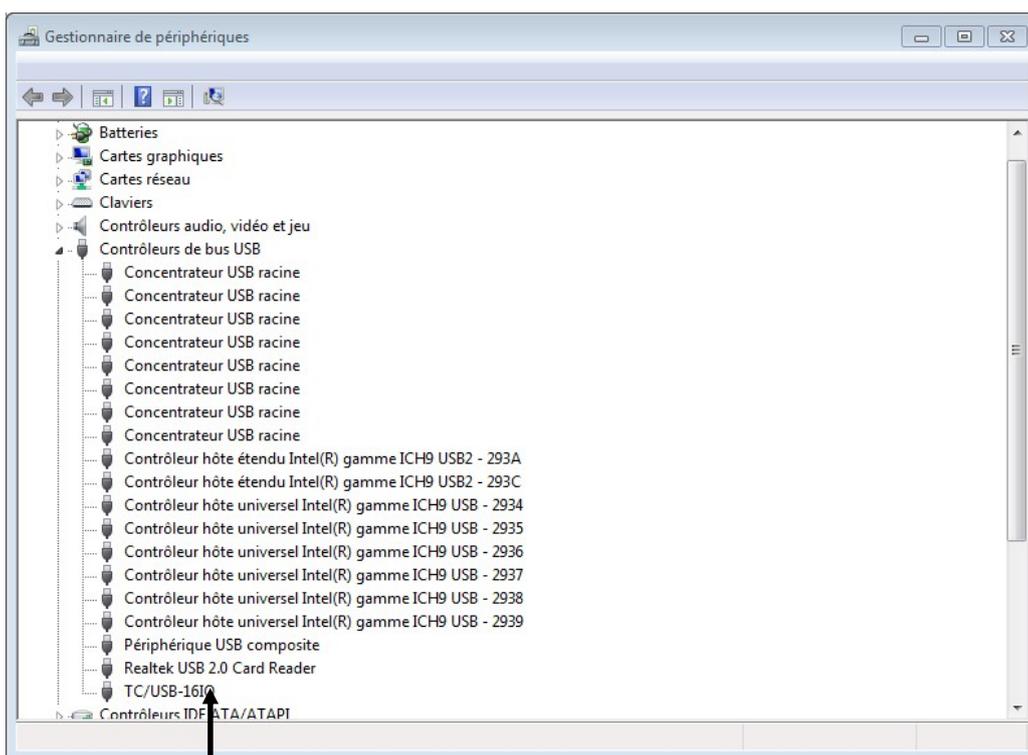
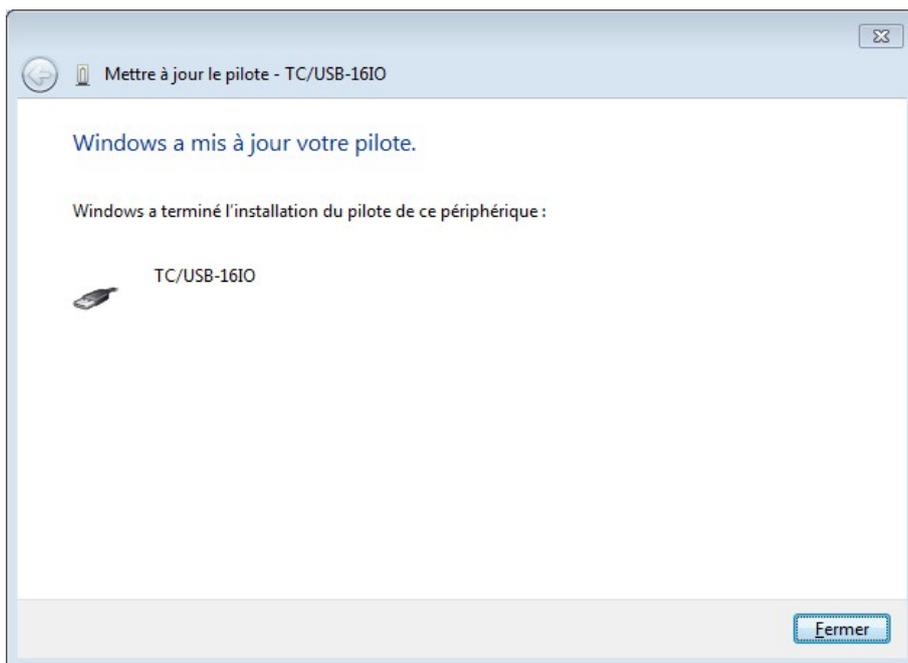


- Indiquer le chemin « CDROM:\TCUSB1610\WINxx\INSTALL » et cliquer sur suivant

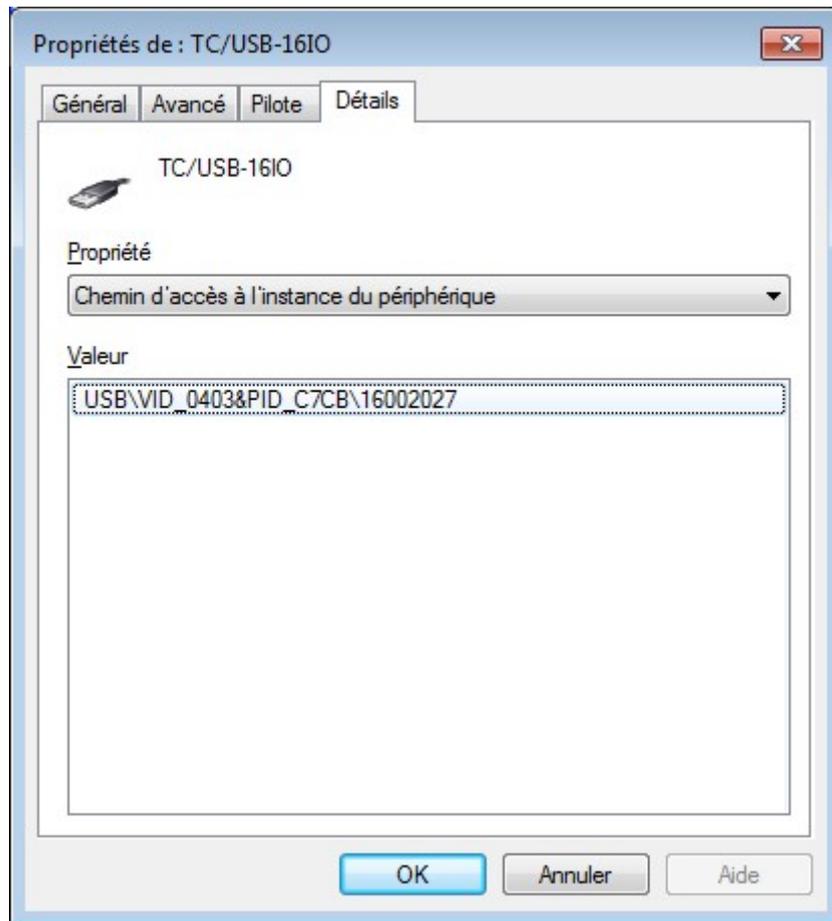


- Cliquer sur « Installer ce pilote quand même »

- L'écran suivant indique que l'installation est terminée avec succès.



- De plus, vous pouvez retrouver le numéro de série du module en cliquant sur « propriétés » de TC/USB-16IO, onglet Détails

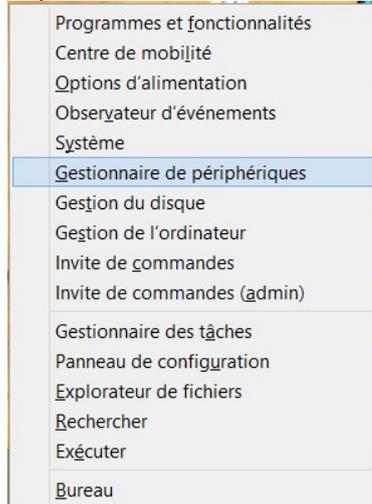


- En dernier lieu, lancez le programme ***TCUSB16IO\WINxx\DEMO\DemoTCUSB16IO.EXE*** présent sur le CD-ROM pour vérifier que l'installation s'est bien déroulée et que le module est accessible. (voir partie Programme de démonstration)

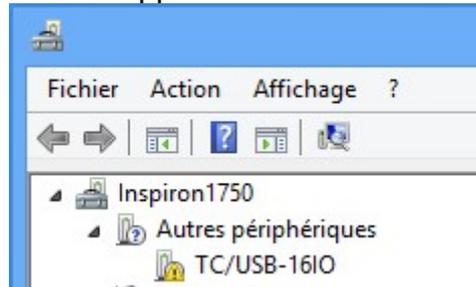
Procédure d'installation sous Windows 8

Le contenu du CD-ROM rassemble les drivers, le programme de démonstration, ainsi qu'une DLL 32bits et les fichiers d'entête pour concevoir ses propres programmes.

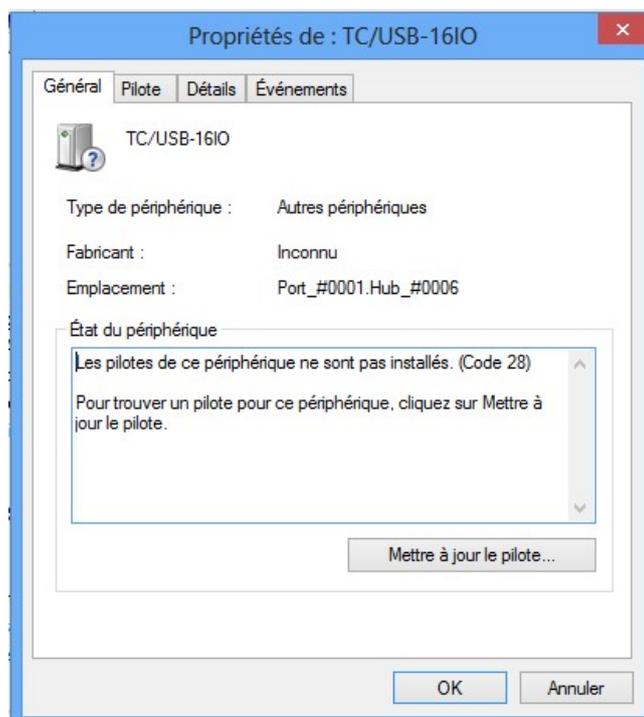
- Branchez le module au PC.
- Appuyez simultanément sur les touches Windows + x du clavier pour faire apparaître le menu suivant puis cliquez sur « **Gestionnaires de périphériques** »



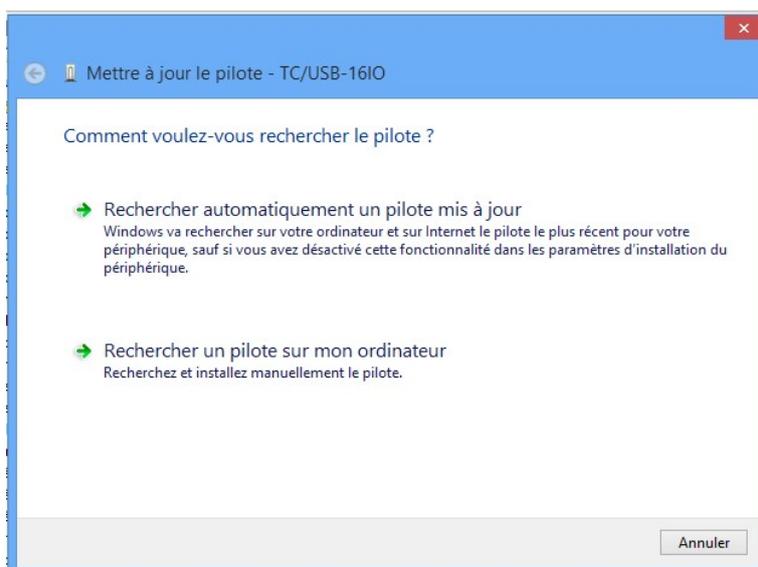
- La fenêtre suivante doit alors apparaître.



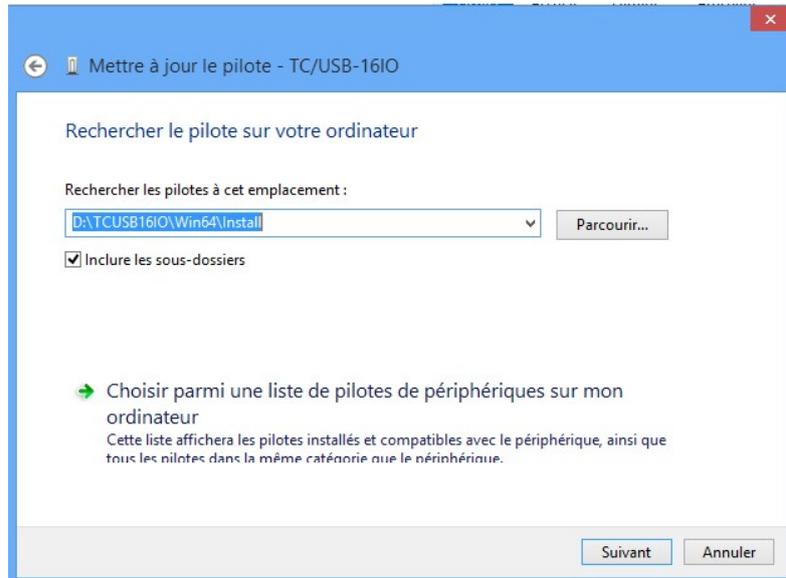
- Un point d'exclamation jaune indique que le module TCUSB16IO a été détecté mais que son driver n'est pas correctement installé ou est inexistant. Cliquer deux fois sur ce dernier



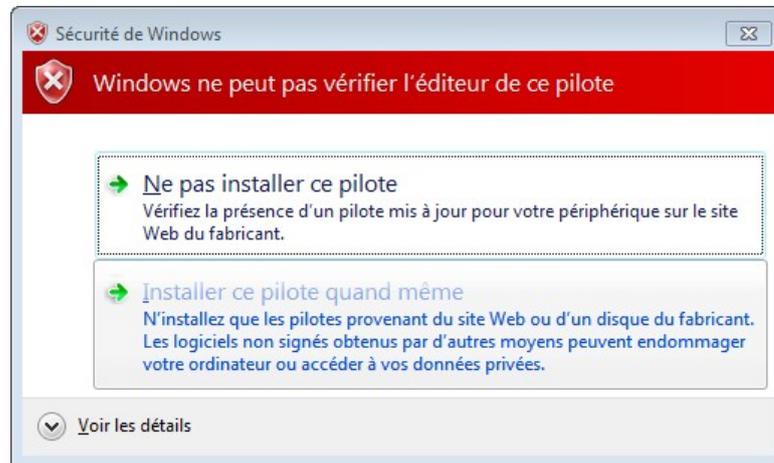
- Cliquer sur mettre à jour le pilote



- Cliquer sur recherchez un pilote sur mon ordinateur

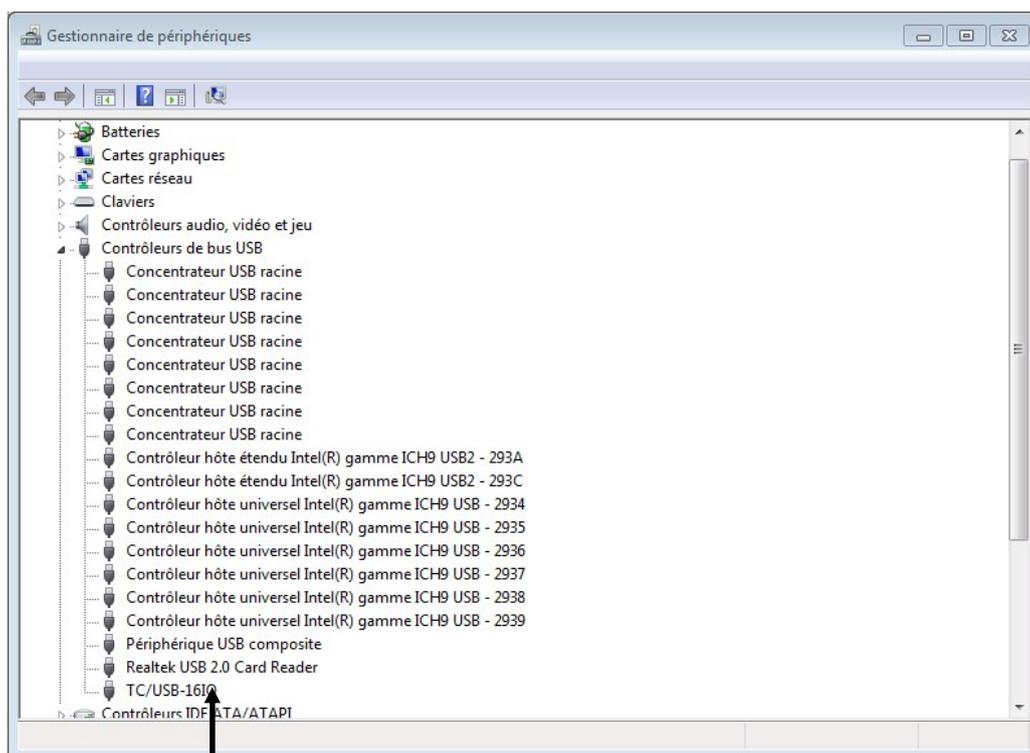
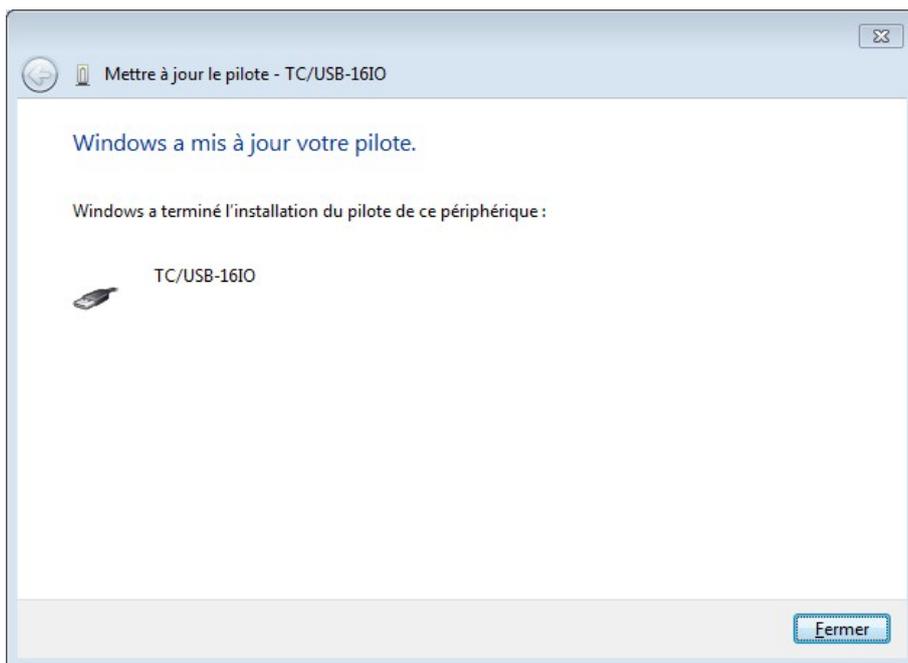


- Indiquer le chemin « CDROM:\TCUSB16IO\WIN32\INSTALL» et cliquer sur suivant

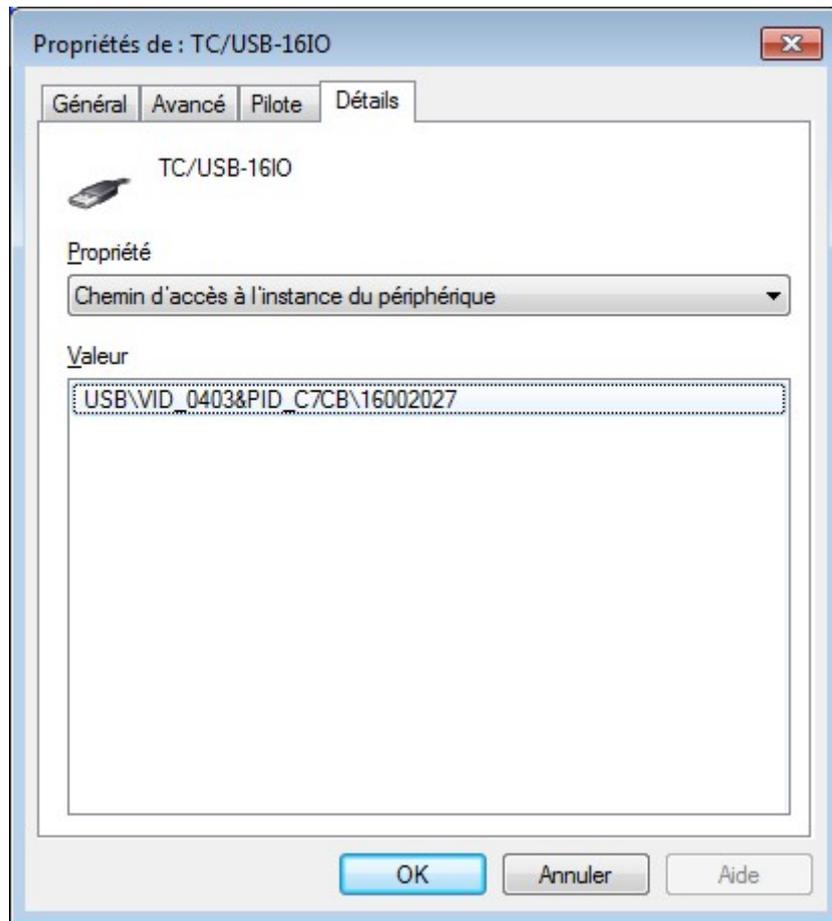


- Cliquer sur « Installer ce pilote quand même »

L'écran suivant indique que l'installation est terminée avec succès.



- De plus, vous pouvez retrouver le numéro de série du module en cliquant sur « propriétés » de TC/USB-16IO, onglet Détails



- En dernier lieu, lancez le programme ***TCUSB16IO\WIN32\DEMO\DemoTCUSB16IO.EXE*** présent sur le CD-ROM pour vérifier que l'installation s'est bien déroulée et que le module est accessible. (voir partie Programme de démonstration)

Procédure d'installation sous Windows 10

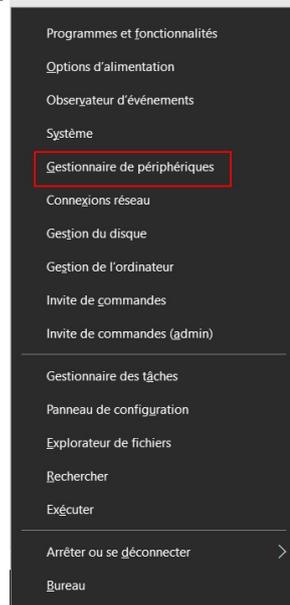
*** Installation sous Windows 10 64 bits ***

Se référer au manuel « Installation_Pilotes_non_signes_windows_10.pdf » présent dans le fichier zip contenant les drivers ou sur notre site Internet.

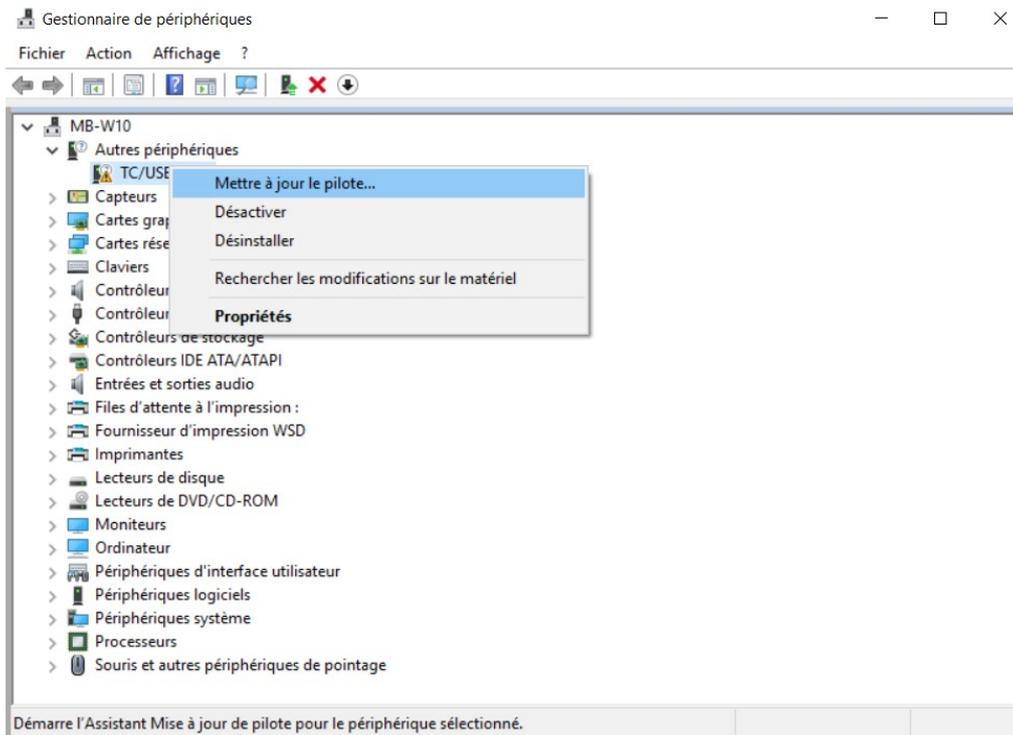
*** Installation sous Windows 10 32 bits ***

Le contenu du CD-ROM rassemble les drivers, le programme de démonstration, ainsi qu'une DLL 32bits et les fichiers d'entête pour concevoir ses propres programmes.

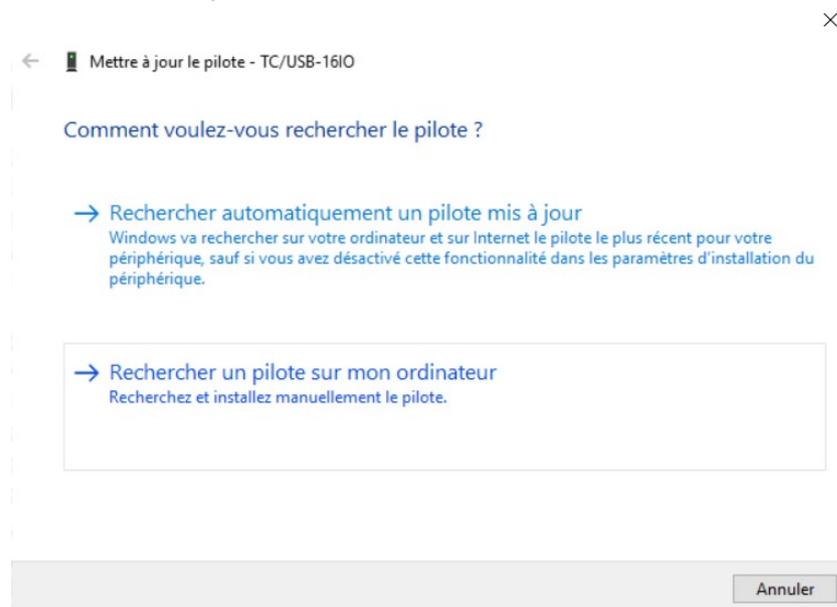
- Branchez le module au PC.
- Appuyez simultanément sur les touches Windows + x du clavier pour faire apparaître le menu suivant puis cliquez sur « **Gestionnaires de périphériques** »



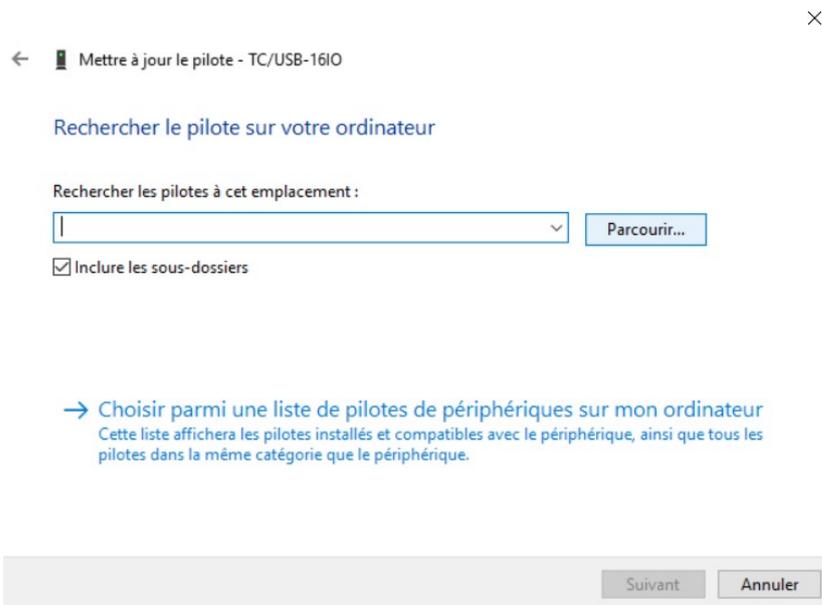
- Un point d'exclamation jaune indique que le module TCUSB16IO a été détecté mais que son driver n'est pas correctement installé ou est inexistant. Faites un clic droit, puis sélectionnez « Mettre à jour le pilote ... »



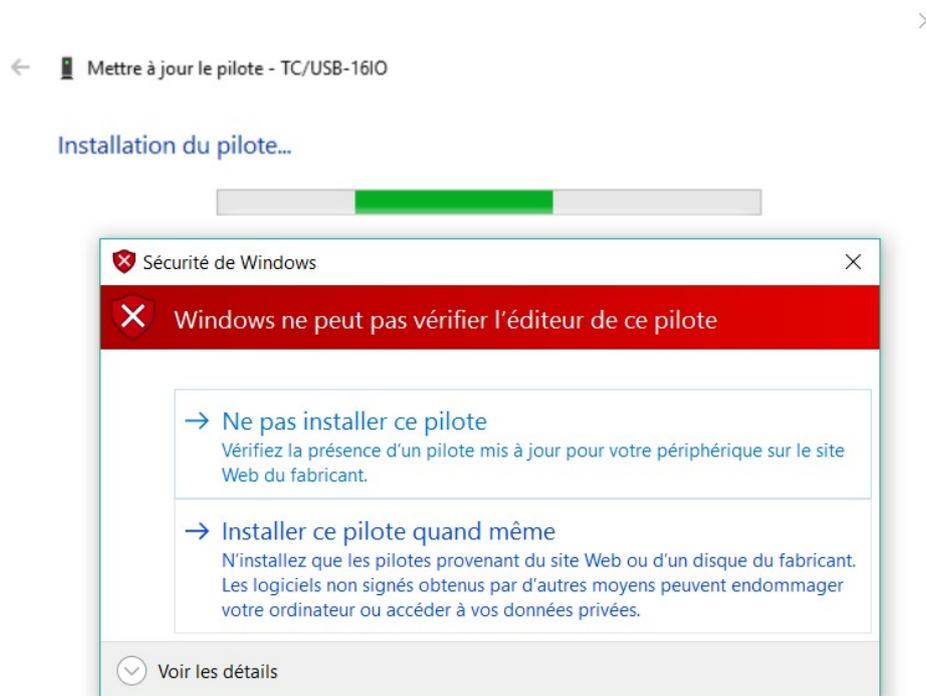
- Cliquer sur recherchez un pilote sur mon ordinateur

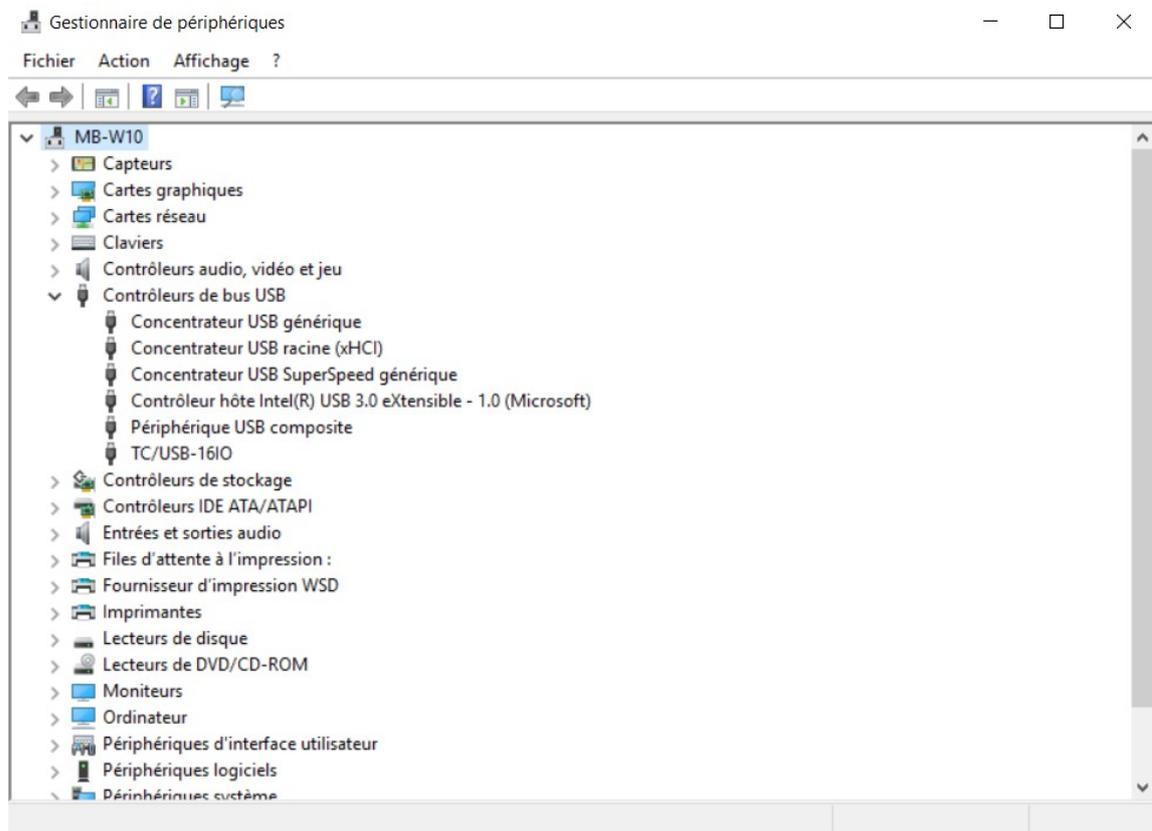


- Indiquer le chemin « CDROM:\TCUSB16IO\WIN32\INSTALL» et cliquer sur suivant

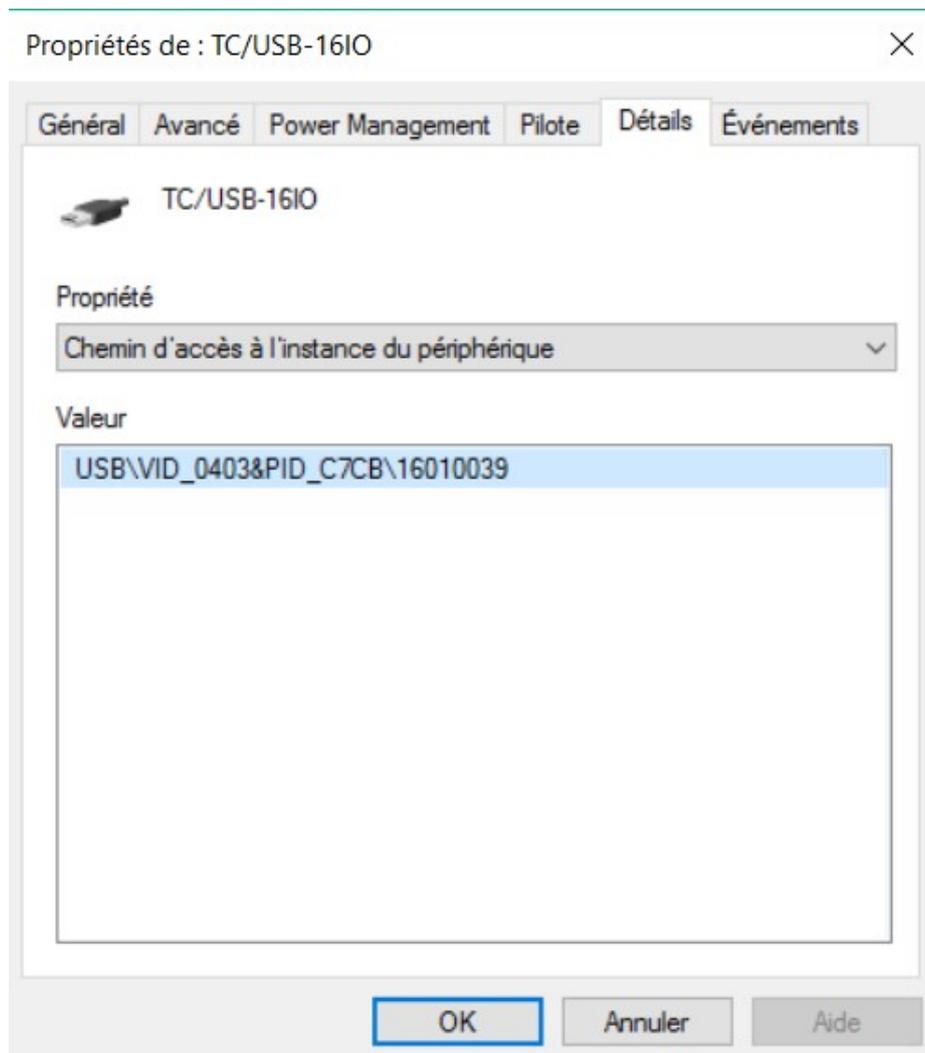


- Cliquer sur « Installer ce pilote quand même »





- De plus, vous pouvez retrouver le numéro de série du module en cliquant sur « propriétés » de TC/USB-16IO, onglet Détails



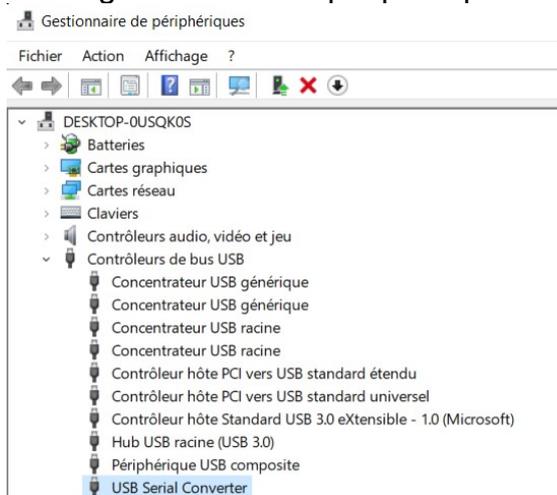
- En dernier lieu, lancez le programme **TCUSB16IO\WIN32\DEMO\DemoTCUSB16IO.EXE** présent sur le CD-ROM pour vérifier que l'installation s'est bien déroulée et que le module est accessible. (voir partie Programme de démonstration)

Installation sous Windows 10 pour les modules acquis à partir de Février 2019 ou mis à niveau via TCUSB_SIGN.exe

1. PC sous Windows 10 connecté à Internet :

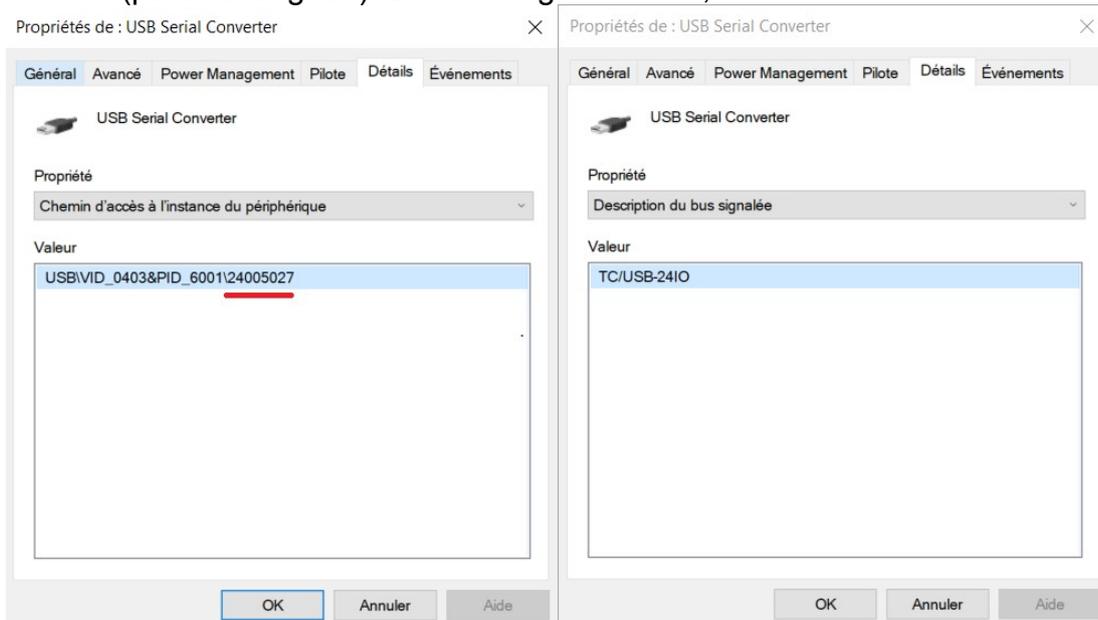
Windows va automatiquement installer les pilotes adéquats pour le module TCUSB concerné, via la fonctionnalité « Windows Update ». Tout sera transparent pour l'utilisateur et ne requiert aucune action particulière de sa part.

Au final, le module sera détecté comme un « USB Serial Converter » dans le menu « Contrôleurs de bus USB » du gestionnaire de périphériques.



On peut vérifier dans les « Propriétés » du périphérique que l'installation s'est bien déroulée :

Onglet « Détails », sélectionner dans le menu déroulant la ligne « Description du bus signalée » et vérifier qu'elle corresponde bien au module connecté au PC. De même, la ligne « Chemin d'accès à l'instance du périphérique » contient le numéro de série du module (partie soulignée). Dans l'image suivante, le module est un module TC/USB-24IO :



Attention, il se peut qu'il existe plusieurs « USB Serial Converter » dans la liste du gestionnaire de périphériques. Si vous avez des produits réalisant cette fonction (ex :

câble émulation de port série), il faudra chercher le bon produit parmi la liste ou déconnecter temporairement les autres produits.

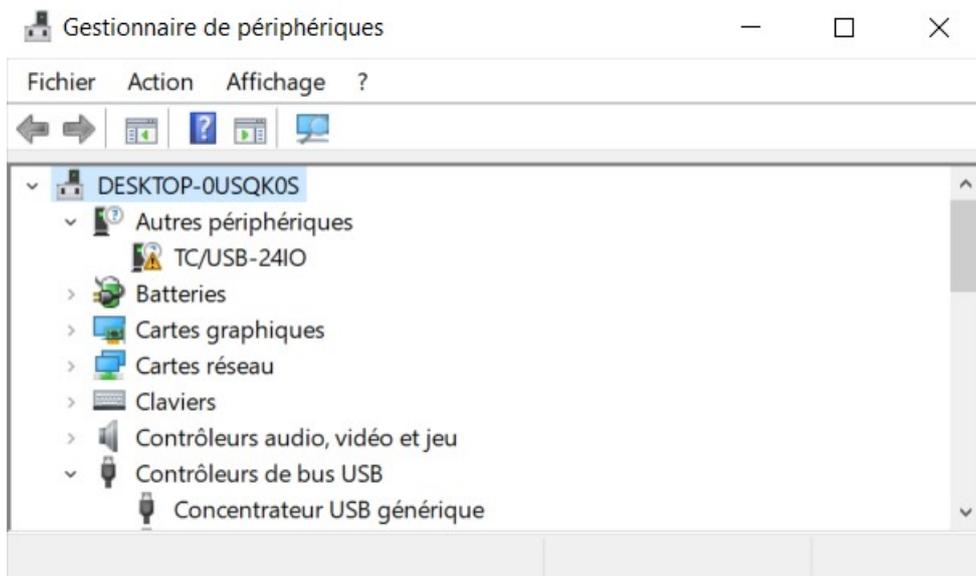
2. PC sous Windows 10 sans connexion à Internet :

Dans ce cas, Windows va détecter un nouveau périphérique, mais ne signalera PAS à l'utilisateur qu'il ne peut l'installer correctement. Il faut aller dans le gestionnaire de périphériques, le module TCUSB apparaît avec un triangle de signalisation jaune.

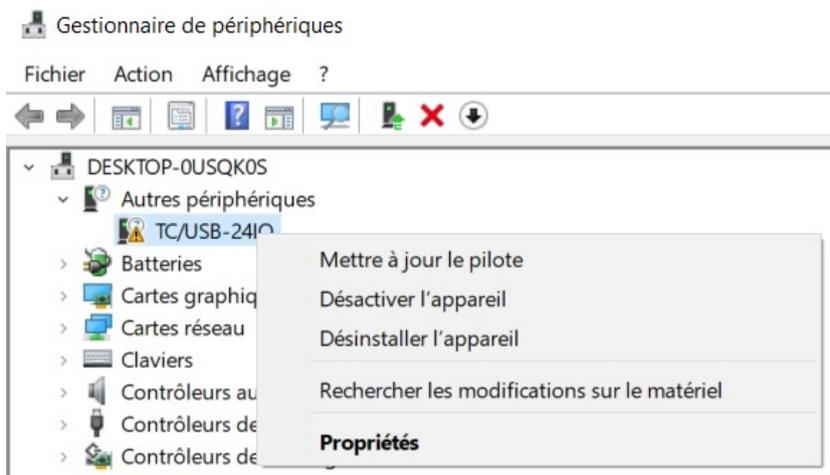
Nb : Les captures d'écrans suivantes montrent en exemple l'installation d'un module TC/USB-24IO. La procédure est la même quelle que soit le module TCUSB concerné. En premier lieu, il faut récupérer une archive disponible sur notre site Internet :

www.3conconcept.fr/tcusb

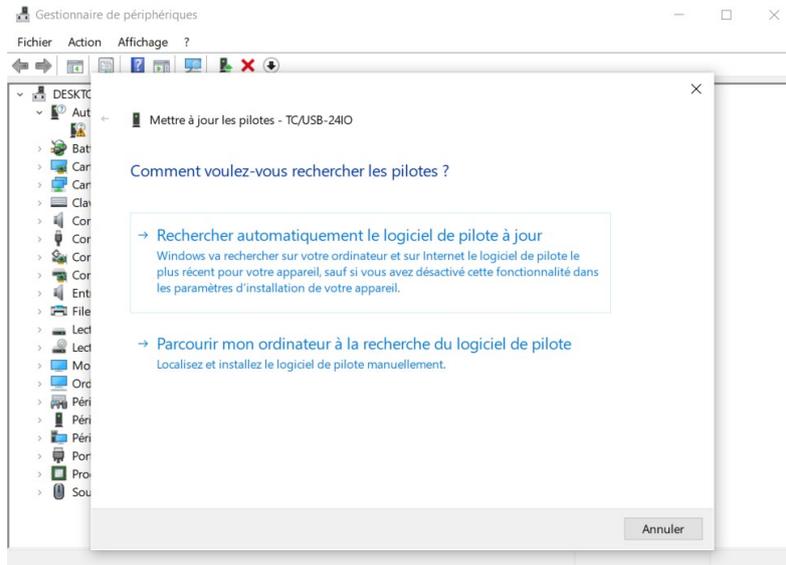
Décompresser l'archive du module concerné sur votre machine, puis suivez la procédure décrite ci-dessous.



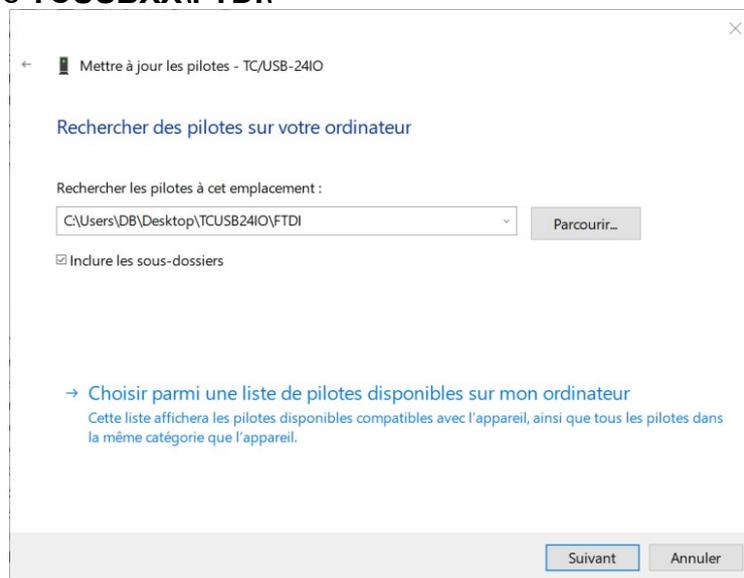
Effectuer un clic droit sur le module et sélectionner « mettre à jour le pilote » :



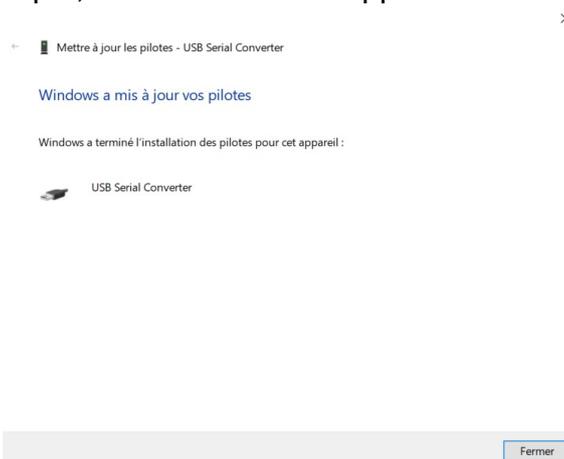
Puis dans la fenêtre qui s'ouvre, sélectionner « Parcourir mon ordinateur... »



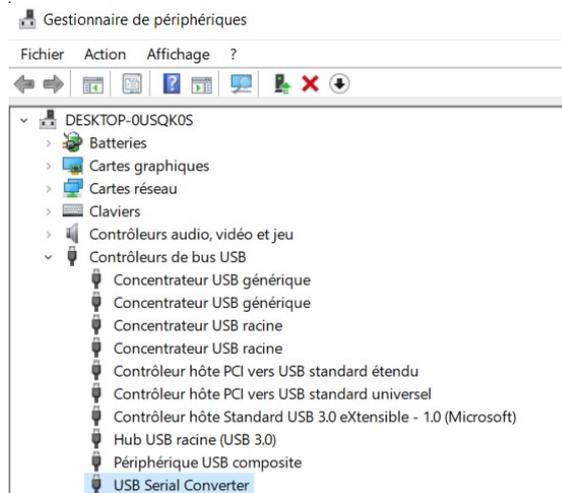
Pointer le répertoire **TCUSBXX\FTDI**



Le processus est automatique, et l'écran suivant apparaît :

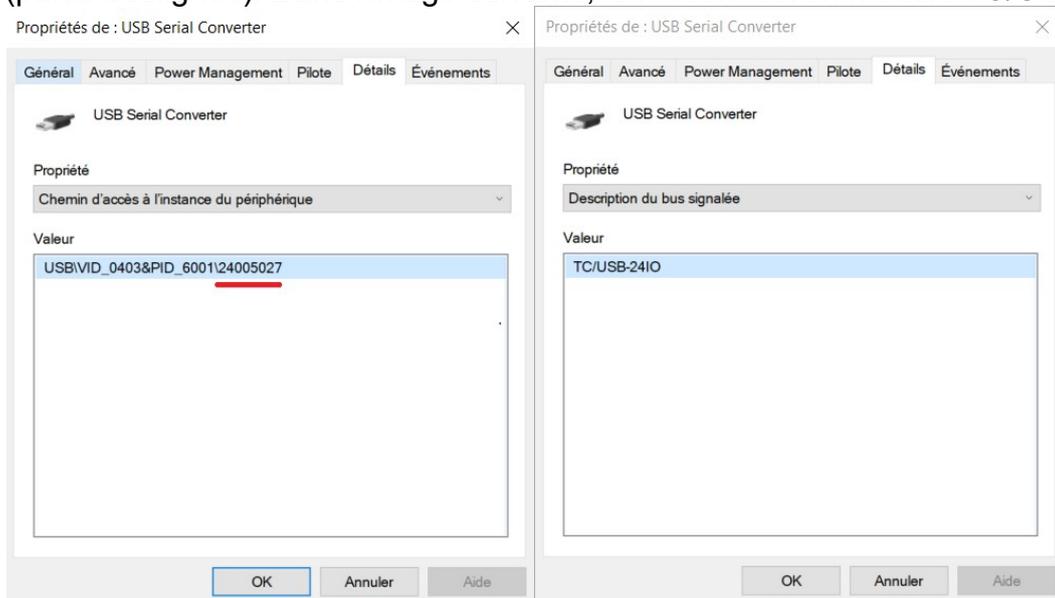


Au final, le module sera détecté comme un « USB Serial Converter » dans le menu « Contrôleurs de bus USB » du gestionnaire de périphériques.



On peut vérifier dans les « Propriétés » du périphérique que l'installation s'est bien déroulée :

Onglet « Détails », sélectionner dans le menu déroulant la ligne « Description du bus signalée » et vérifier qu'elle corresponde bien au module connecté au PC. De même, la ligne « Chemin d'accès à l'instance du périphérique » contient le numéro de série du module (partie soulignée). Dans l'image suivante, le module est un module TC/USB-24IO :



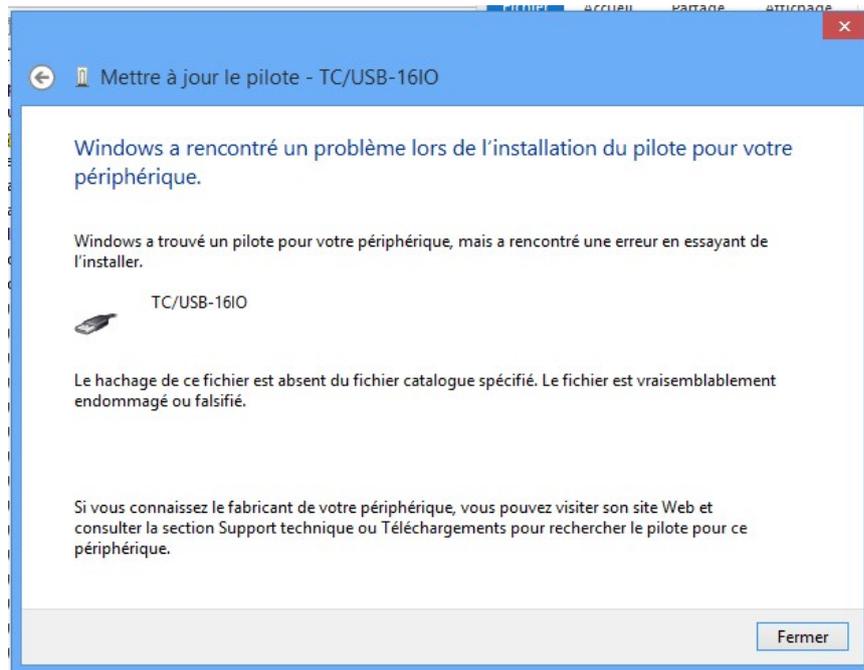
Attention, il se peut qu'il existe plusieurs « USB Serial Converter » dans la liste du gestionnaire de périphériques. Si vous avez des produits réalisant cette fonction (ex : câble émulation de port série), il faudra chercher le bon produit parmi la liste ou déconnecter temporairement les autres produits.

En dernier lieu, lancer le programme **DemoTCUSBXX.exe** fourni avec l'archive ou récupéré sur notre site, et vérifier le bon fonctionnement du module. (XX désigne un module TCUSB générique, on remplace les lettres XX par le produit concerné, Ex : TC/USB-24IO).

- Point d'exclamation sur système/gestion de périphérique :

Un point d'exclamation jaune dans le gestionnaire de périphérique de Windows indique qu'un matériel a été détecté mais que son driver n'est pas correctement installé ou est inexistant. Pour résoudre ce problème il suffit de sélectionner les propriétés du périphérique, onglet pilote, choisir « mettre à jour le pilote » et recommencer les étapes de l'installation décrites auparavant.

- Windows a rencontré un problème lors de l'installation du pilote pour votre périphérique :



Si vous obtenez cet écran lors de l'installation sous Windows 8 / Windows 10, cela signifie que vous êtes en version 64 bits.

Consultez le document « Marche à suivre pour installer des pilotes non signés sous Windows 8 / Windows 10 » disponible sur le CD ou sur notre site.

- Installation interrompue :

Si au cours de l'installation un problème informatique survient, veuillez recommencer toutes les étapes de l'installation, et se référer aux autres cas de résolution de problèmes pour trouver une issue.

Une autre solution consiste à redémarrer le PC sans aucun module branché, puis de le relancer après avoir branché le module : Ce cycle de redémarrage force Windows à re-détecter les nouveaux périphériques, permettant à l'utilisateur de ressaisir l'emplacement du pilote.

- Désinstallation du driver :

Si vous tenez à supprimer le driver, il suffit de lancer le fichier « Uninstall Driver TCUSB16IO.bat » présent sur le CD fournit avec le module, dans le répertoire « Win32/Install ».

Il faut cependant **au préalable** copier le fichier « **Uninstall Driver TCUSB16IO.bat** » et « **CDMUninstaller.exe** » sur le disque dur du PC hôte. (un fichier Log est créé à l'issu de la désinstallation)

Programme de démonstration

Le programme de démonstration est présent sur le CD-ROM livré avec le module USB, il se nomme « TCUSB16IO\WIN32\Demo\DemoTCUSB16IO.exe ».



Lors du lancement du programme de démonstration, celui-ci demande le numéro de série du module TC/USB-16IO. Si vous n'utilisez qu'un seul module, vous n'êtes pas obligé de rentrer son numéro de série, il suffit de cliquer sur OK et le programme détecte le premier module branché.

NB : le numéro de série est affiché au dos du module

Etats des 8 entrées ToR

Compteur d'événement des 8 entrées ToR

Etats des 6 sorties ToR

Etats des entrées codeurs

Compteur du codeur

Toutes les sorties sont pilotables manuellement en cliquant sur la zone verte de la sortie correspondante.

Bouton " Sequencieur ON/OFF " : Lance/ arrête le défilement automatique des sorties

Bouton "WatchDogOn" : Mise en marche du chien de garde

Bouton " WatchdogOff" : Arret du chien de garde

Bouton " WatchDogRearm" : Réarmement (manuel , toutes les 2secondes) du chien de garde

Bouton "StopAutoRefresh" : Permet d'accéder aux fonctionnalités du chien de garde, et stoppe la scrutation continue des entrées / sorties./ codeur

Bouton "Reset Position" : Permet la RAZ de la position du codeur

Utilisation de la DLL 32bits TCUSB16IO.DLL

Une DLL destinée à l'utilisation du module dans un environnement Windows 32 bits (2000/XP) est fournie avec le module. Cette DLL permet à l'utilisateur d'appeler les procédures et fonctions de gestion du module depuis un langage quelconque supportant l'appel aux procédures externes.

Pour un compilateur PASCAL (DELPHI):

Utilisez le fichier TCUSB16IO.INC et la DLL TCUSB16IO.DLL

Pour un compilateur C :

Utilisez le fichier TCUSB16IO.H et la DLL TCUSB16IO.DLL

Pour le langage Basic:

Utilisez le fichier TCUSB16IO.BAS et la DLL TCUSB16IO.DLL

Pour le langage C# :

Utilisez le fichier TCUSB16IO.CS et la DLL TCUSB16IO.DLL

Shared object sous Linux:

Voir dans le répertoire « linux » du disque fourni, un fichier texte explique la démarche à suivre.(TCUSB16IO\Linux\)

ⓘ Attention : Toutes les fonctions ci-dessous travaillent sur une image mémoire actualisée par la fonction TCUSB16IO_Refresh. Cette fonction doit être appelée en tête de toute fonction utilisant les entrées et à la fin de toute fonction actualisant les sorties. Il convient de ne pas actualiser les données après chaque action en raison du temps de transfert sur le bus USB.

Une fenêtre d'information peut apparaître de façon fugitive (3 sec environ) si le driver utilisé n'est plus à jour. Dans ce cas, il est conseillé de télécharger la nouvelle version en cliquant sur le lien présent sur la fenêtre d'information.

Utilisation de la DLL 64bits TCUSB16IOx64.DLL

Une autre version de la DLL est destinée à l'utilisation du module avec un compilateur 64 bits. Cette DLL permet à l'utilisateur d'appeler les procédures et fonctions de gestion du module depuis un langage quelconque supportant l'appel aux procédures externes.

Pour un compilateur C++ :

Utilisez le fichier TCUSB16IOx64.H, TCUSB16IOx64.LIB et la DLL TCUSB16IOx64.DLL

Il existe un programme d'exemple mettant en œuvre cette DLL, il a été développé avec le programme suivant:

Microsoft Visual Studio 2010 C++

**Il faut noter que la DLL 64bits n'effectue plus de contrôle de version sur le web (pour une éventuelle mise à jour de celle-ci).
De plus, seule la fonction TCUSB16IO_OpenN a été implantée. La version TCUSB16IO_Open est devenue obsolète.
En dernier lieu, la procédure d'appel des fonctions est de type CDECL dans le cas d'utilisation de cette DLL.**

Description des fonctions de la DLL

TCUSB16IO_OpenN(Nmodule, NumSerie)

Paramètre : Nmodule: entier signé sur 32 bits.

Numserie : Pointeur sur un Dword non signé sur 32 bits contenant le numéro de série.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Initialise le driver, affecte le numéro de série du module au numéro de module spécifié (1 à 8) et remet à 0 les sorties. Si le numéro de série=0 alors elle ouvre le premier module détecté et renvoi son numéro de série.

A utiliser avant tout autre appel à une fonction TCUSB16IO_xxx.

Fermer la session par TCUSB16IO_Close .

TCUSB16IO_Open(Nmodule, NumSerie)

Paramètre : Nmodule: entier signé sur 32 bits.

Numserie : Pointeur sur une chaîne de caractère contenant le numéro de série.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Initialise le driver, affecte le numéro de série du module au numéro de module spécifié (1 à 8) et remet à 0 les sorties. Si le numéro de série est vide alors elle ouvre le premier module détecté et renvoi son numéro de série.

A utiliser avant tout autre appel à une fonction TCUSB16IO_xxx.

Fermer la session par TCUSB16IO_Close .

L'utilisation des pointeurs sur les chaînes de caractères n'étant pas toujours aisée à mettre en œuvre en fonction du système de développement, il est préférable d'utiliser la fonction TCUSB16IO_OpenN

TCUSB16IO_Close(NModule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Termine la session et ferme le driver (indispensable à la fin du programme)

TCUSB16IO_Refresh(NModule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Actualise les données (Entrées + Sorties + Compteurs+ Codeur) du module indiqué. A utiliser après toute fonction sur les sorties et avant toute fonction sur les entrées et compteurs

TCUSB16IO_Input(Nmodule , Data) (1)

Paramètre : Nmodule: entier signé sur 32 bits.

Data : Pointeur sur un Dword non signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Lecture de l'état des 8 entrées ToR + 3 entrées Codeurs du module indiqué, le retour se fait par paramètre dans la variable Data (32 bits), un code erreur correspond à la valeur retournée par la fonction. Les entrées codeur (A,B,Z) correspondent respectivement au bit (8,9,10).ie : Z,B,A,b7,b6,b5,b4,b3,b2,b1,b0.

Variable Input	Bit10	Bit9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Etat de :	Z	B	A	E8	E7	E6	E5	E4	E3	E2	E1

TCUSB16IO_Output(Nmodule, Data) (1)

Paramètre : Nmodule: entier signé sur 32 bits.

Data : Byte non signé sur 8 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Envoi sur les sorties du module indiqué la valeur contenue dans la variable Data (8 bits), un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_BitInput(Nmodule , Entree) (1)

Paramètre : Nmodule: entier signé sur 32 bits.

Entree : entier signé sur 32 bits.

Retour : la fonction renvoi l'état (0 ou 1) de l'entrée spécifiée par un entier signé sur 32 bits.

Lecture de l'état d'une entrée (1 à 11) du module spécifié.

Variable entrée	1	2	3	4	5	6	7	8	9	10	11
Etat de :	E1	E2	E3	E4	E5	E6	E7	E8	A	B	Z

TCUSB16IO_BitOutput(NModule,Sortie,etat) (1)

Paramètre : Nmodule : entier signé sur 32 bits.

Sortie : entier signé sur 32 bits.

etat : entier signé sur 32 bits.

Place la sortie (1 à 6) du module spécifié dans l'état indiqué (0 ou 1)

TCUSB16IO_BitReadBack(NModule,Sortie) (1)

Paramètre : Nmodule: entier signé sur 32 bits.

Sortie : entier signé sur 32 bits.

Retour : la fonction renvoi l'état (0 ou 1) de la sortie spécifiée par un entier signé sur 32 bits.

Relecture de l'état d'une sortie (1 à 6) du module spécifié

TCUSB16IO_Counter(NModule,Entree,Data) (1)

Paramètre : Nmodule : entier signé sur 32 bits.

Entree : entier signé sur 32 bits.

Data : Pointeur sur un Byte non signé sur 8 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Lecture du compteur de l'entrée spécifiée du module indiqué, le retour se fait par paramètre dans la variable Data (8 bits), un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_Config(Nmodule,Data) (1)

Paramètre : Nmodule : entier signé sur 32 bits.
Data : Byte non signé sur 8 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Configure sur le module spécifié :

- 1) sur les bits 1 à 3 de data, la valeur du filtre numérique à appliquer sur les entrées A, B, Z (de 0 à 6 avec 0:24Mhz, 1:12Mhz, 2 :6Mhz, 3 :1.5Mhz, 4 :750Khz, 5 :375Khz, 6 :187.5Khz)
- 2) sur le bit 4, la validité du Top Zéro pour remettre à zéro le compteur du codeur un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_Pos(Nmodule,Data) (1)

Paramètre : Nmodule: entier signé sur 32 bits.
Data : Pointeur sur un entier signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Lecture de la valeur du compteur du codeur du module spécifié. Le retour se fait par paramètre dans la variable Data (32 bits signés).
un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_ResetPos(Nmodule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Remet à zéro la valeur du compteur du codeur et des deux latches du module spécifié.
un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_ConfigLatch(Nmodule,Data)

Paramètre : Nmodule : entier signé sur 32 bits.
Data : Byte non signé sur 8 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Configure sur le module spécifié :

- 1) sur les bits 1 à 3 de data, le mode de latch 1 du codeur possible sur l'entrée 7 (de 0 à 3 avec 0:Pas de latch, 1 : Latch sur front montant de l'entrée 7, 2 : Latch sur front descendant de l'entrée 7, 3 : Latch sur changement d'état de l'entrée 7)
- 2) sur les bits 5 à 7 de data, le mode de latch 2 du codeur possible sur l'entrée 8 (de 0 à 3 avec 0:Pas de latch, 1 : Latch sur front montant de l'entrée 8, 2 : Latch sur front descendant de l'entrée 8, 3 : Latch sur changement d'état de l'entrée 8)

un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_Latch(Nmodule,Latch,Data) (1)

Paramètre : Nmodule: entier signé sur 32 bits.

Latch : Byte non signé sur 8 bits

Data : Pointeur sur un entier signé sur 32 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Lecture de la valeur du latch spécifié (de 1 à 2) du codeur du module spécifié. Le retour se fait par paramètre dans la variable Data (32 bits signés).

un code erreur correspond à la valeur retournée par la fonction

TCUSB16IO_WdgRun(Nmodule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : Aucun

Arme la surveillance du chien de garde sur le module spécifié

TCUSB16IO_WdgStop(Nmodule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : Aucun

Désactive la surveillance du chien de garde sur le module spécifié

TCUSB16IO_WdgRearm(NModule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : Aucun

Réarmement du chien de garde sur le module spécifié

TCUSB16IO_ResetPort(NModule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : Aucun

Réinitialise le port USB du module : En cas de perte de communication avec le module, cette commande tente de reprendre le dialogue. (Cette fonction n'est pas équivalente au fait de dé-brancher/re-brancher le module.)

TCUSB16IO_CyclePort(NModule)

Paramètre : Nmodule: entier signé sur 32 bits.

Retour : Aucun

Déconnecte et reconnecte le port USB du module : Cette commande est équivalente au fait de dé-brancher et re-brancher physiquement le module. Cette commande peut être utilisée suite à une erreur sérieuse , pour tenter de rétablir la communication module/PC.

TCUSB16IO_Version(NModule,Data1,Data2) (1)

Paramètre : Nmodule : entier signé sur 32 bits.

Data1 : Pointeur sur un Byte non signé sur 8 bits.

Data2 : Pointeur sur un Byte non signé sur 8 bits.

Retour : la fonction renvoi un entier signé sur 32 bits correspondant au code du défaut (voir liste des défauts ci-dessous).

Lecture de la version du firmware du module indiqué et de la DLL, le retour se fait par paramètre dans les variables Data1 (module) et Data2 (DLL) (8 bits non signé), un code erreur correspond à la valeur retournée par la fonction

(1) Voir la mise en garde concernant l'usage de la fonction *TCUSB16IO_Refresh* ci-dessus

Liste des défauts:

N°défaut	Désignation
0	Pas d'erreur
1	Numéro de série incorrect
2	N° de module incorrect
3	Pas de module TC/USB-16IO détecté
4	Module non initialisé
5	Erreur écriture
6	Erreur lecture
7	Erreur à la fermeture

Problème sur l'ouverture du module avec la fonction TCUSB16IO_Open

L'appel à la fonction TCUSB16IO_Open se fait en passant comme paramètre un numéro de module (de 1 à 8) et un numéro de série,
Le numéro de série est passé en tant que pointeur sur la chaîne de caractère de type zéro terminal contenant le numéro de série, Cette chaîne peut être vide lors de l'appel à TCUSB16IO_Open auquel cas, le premier module trouvé sera ouvert et la fonction retournera le numéro de série du module trouvé.

Sous ATEasy, lors de la déclaration de la procédure TCUSB16IO_Open, la chaîne contenant le numéro de série doit être de type String parameter Var avec une dimension de 1.

Lors de l'appel à cette procédure dans le programme la variable doit être de type String parameter None avec une dimension de 1.