Codesys et Master EtherCAT sur Raspberry PI

Introduction

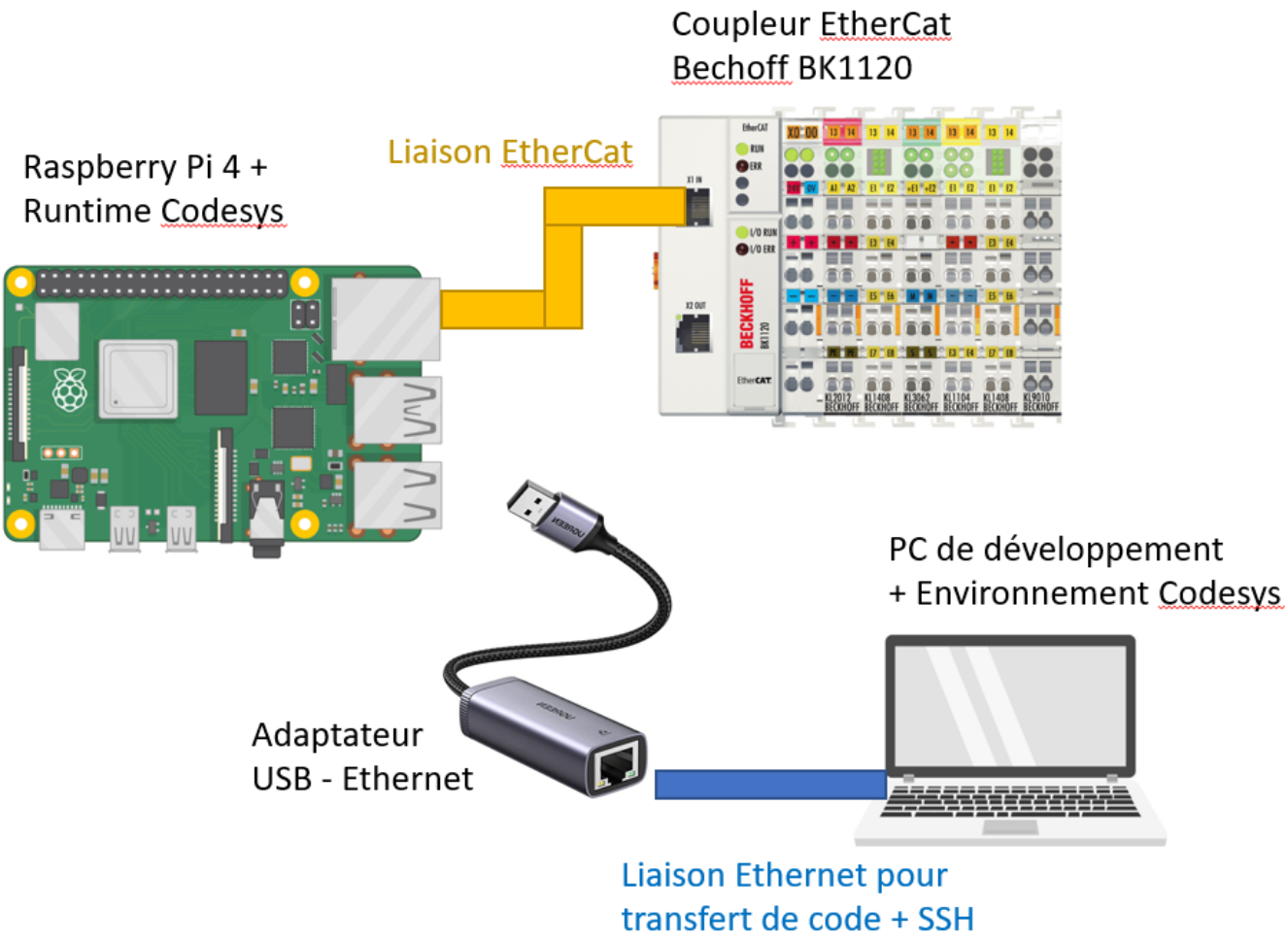
Le Runtime de Codesys permet de générer un Master Ethercat. Dans cet article, nous allons utiliser un Raspberry Pi avec le Runtime CodeSys comme Master EtherCAT et un coupleur EtherCAT BK1120 de Beckhoff commme I/O déporté (Slave).

Branchement

Le branchement du Raspberry Pi sera modifié dans le contexte d’un réseau Ethercat.

- On utilisera le connecteur Ethernet du Rpi pour la liaison Ethercat avec le coupleur Beckhoff.
- Pour connecter le RPi au PC de développement, on ajoutera un second port Ethernet avec un adapteur USB-Ethernet.

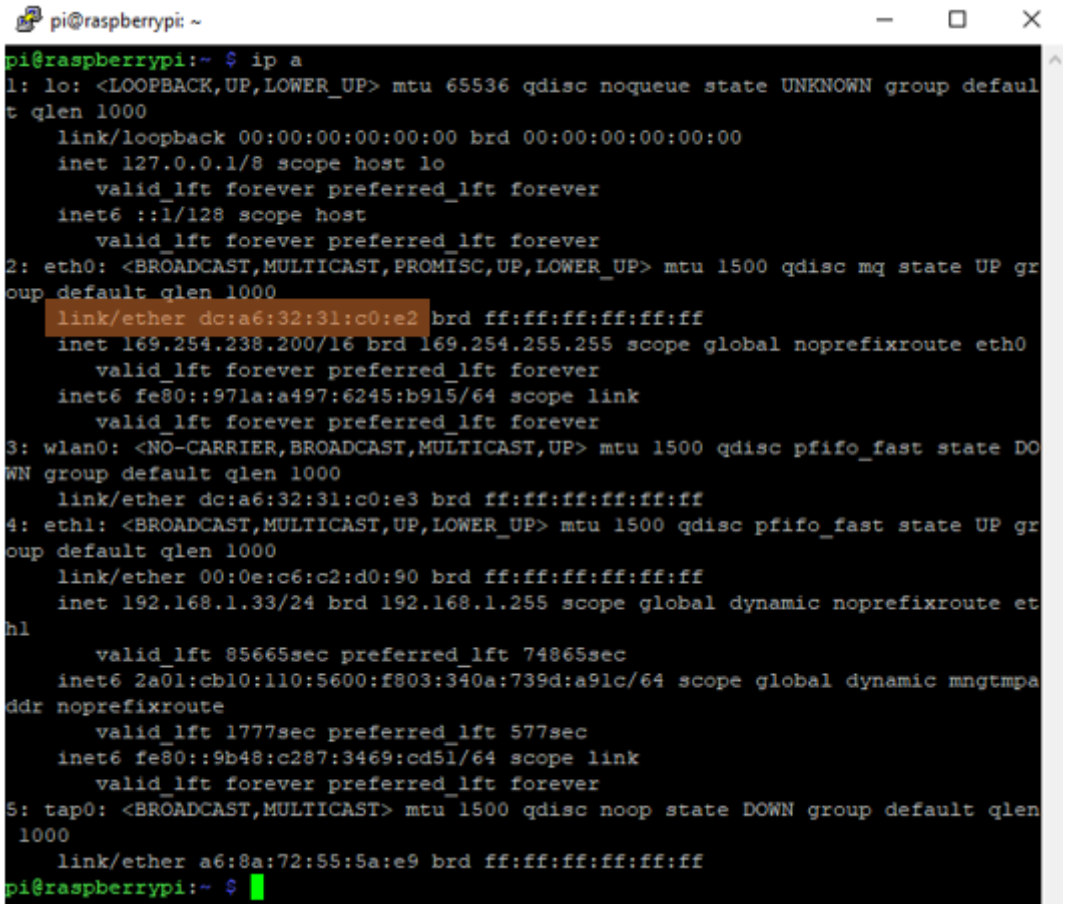
Dans ce cas d’une communication Ethercat, il ne faut pas utiliser de switch, un branchement direct entre le Master et le Slave est nécessaire.



Configuration réseau : IP + EtherCAT

Deux configurations réseau seront à effectuer sur le Raspberry Pi:

- affecter une adresse IP pour l'adaptateur USB-Ethernet que nous utiliserons pour les liaisons SSH, VNC et Codesys.
- pour la liaison Ethercat, il est nécessaire de récupérer l'adresse MAC (physique) du port Ethernet physique du RPi. On peut utiliser SSH + la commande `ip a` pour récupérer cette information.



Le port Ethernet physique du RPi est associé à eth0. L'adresse MAC ou physique, aussi nommée ether, est donc DC-A6-32-31-C0-E2. Cette adresse devra être écrite en Majuscule séparée par des tirets dans Codesys.

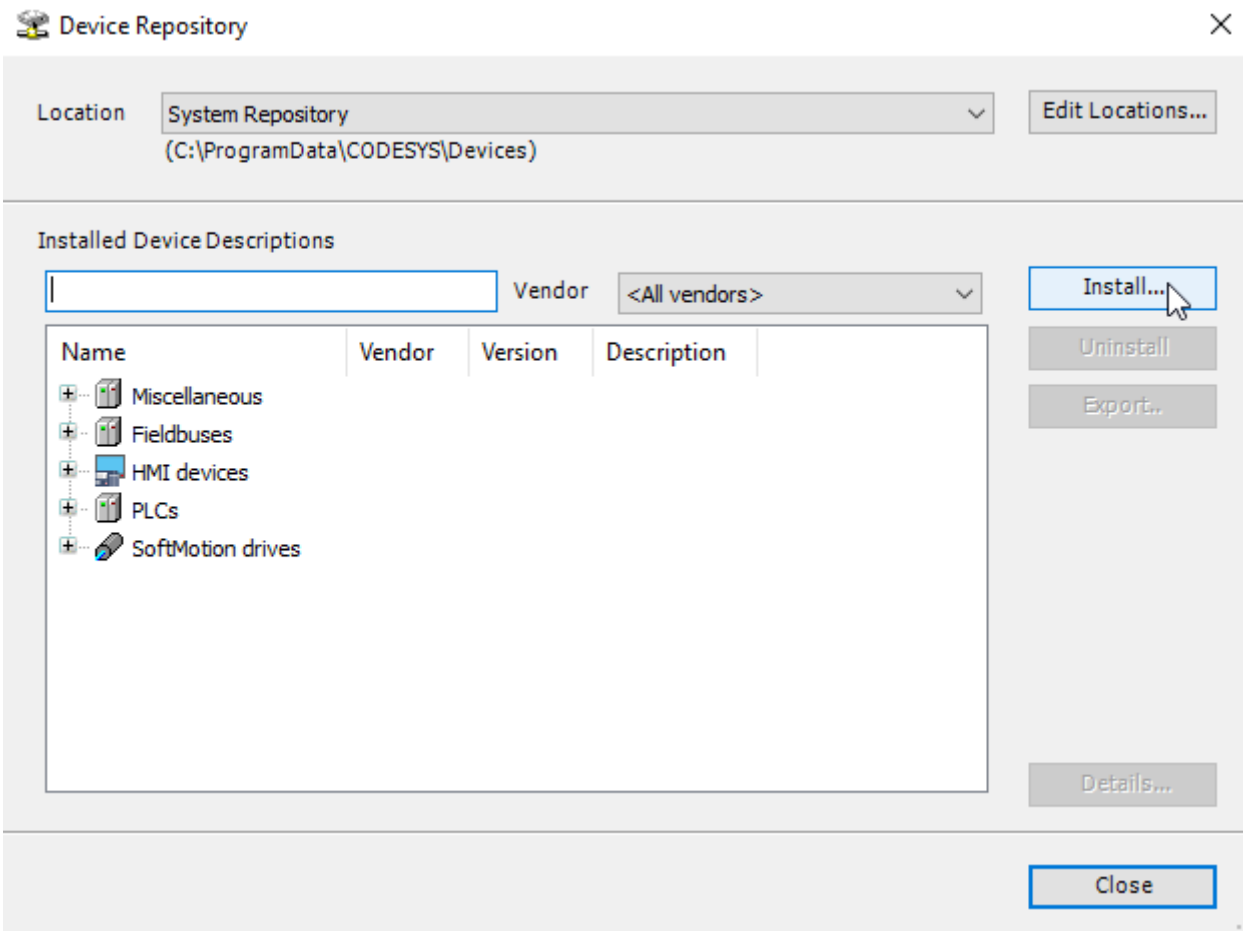
Fichiers de description Beckhoff

Avant de pouvoir connecter le coupleur Beckhoff, nous devons nous assurer de disposer des fichiers XML de description du BK1120 et de ses modules. Par analogie avec l'environnement Siemens, les fichiers XML Beckhoff correspondent aux fichiers GSDML de Siemens.

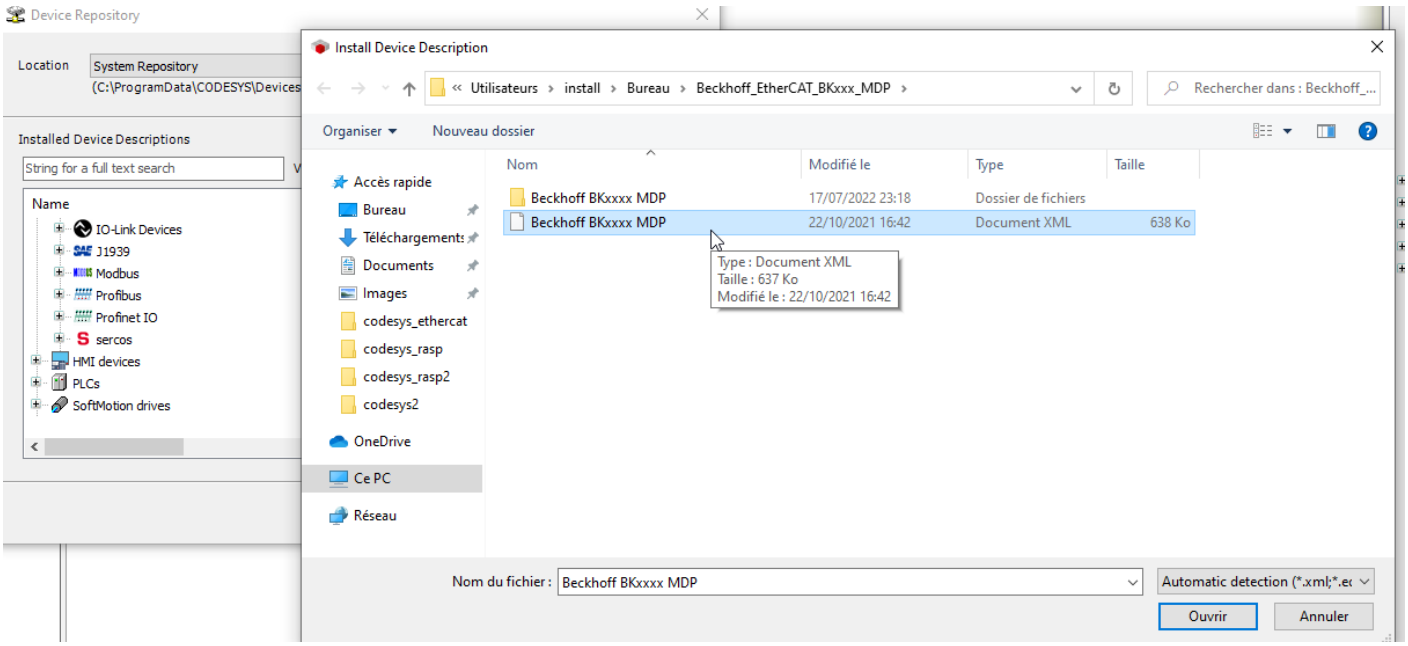
- Téléchargez et décompressez le fichier Beckhoff_EtherCAT_BKxxx_MDP.zip dans un endroit sûr.
- Ouvrez CODESYS, dans la barre d'outils, sélectionnez Tools -> Device Repository ...



- Cliquez sur "Install..."



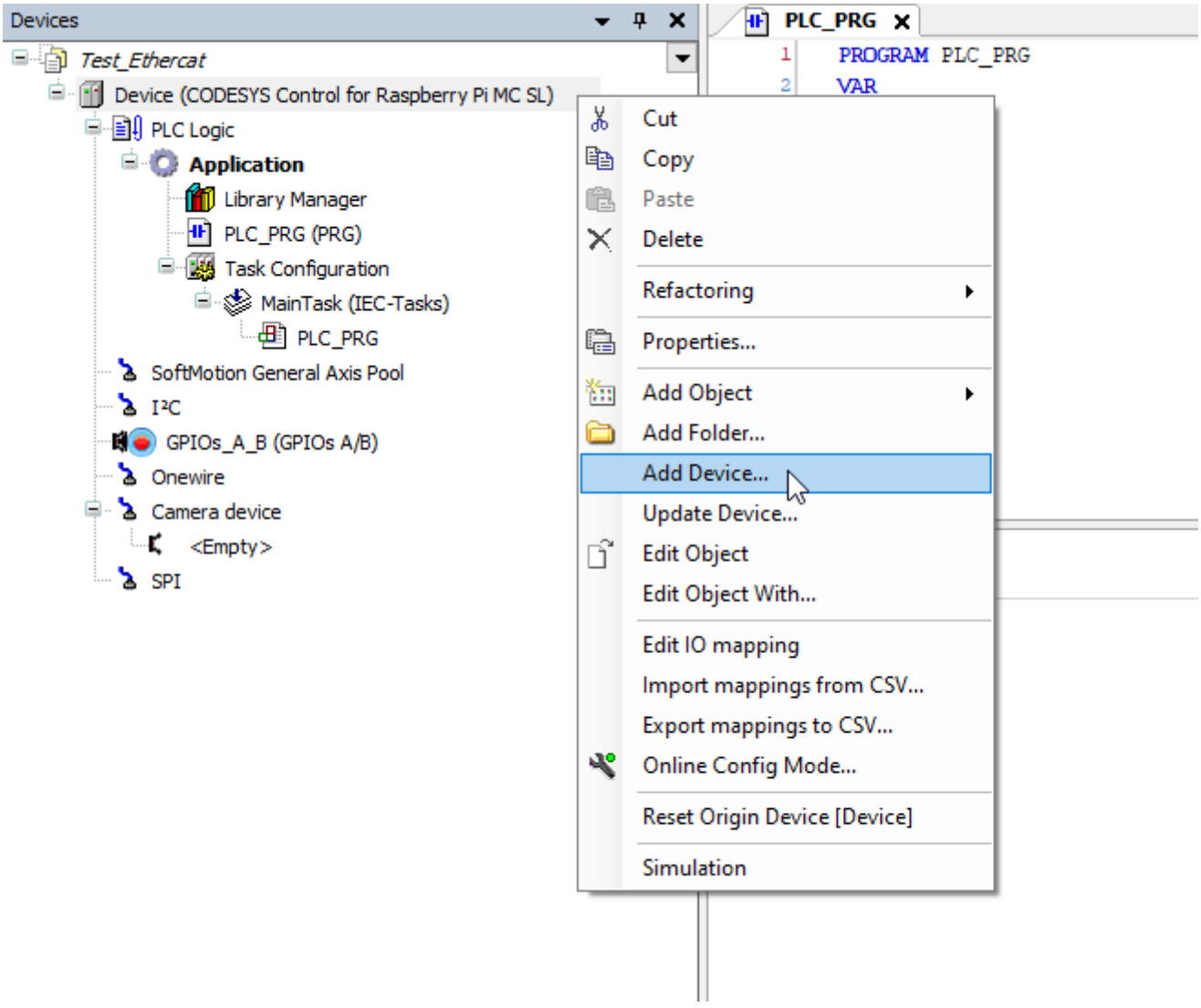
- Accédez au dossier que vous venez de décompresser, sélectionnez le fichier XML "BKxxxx" et cliquez sur Ouvrir, CODESYS installera automatiquement l'appareil.



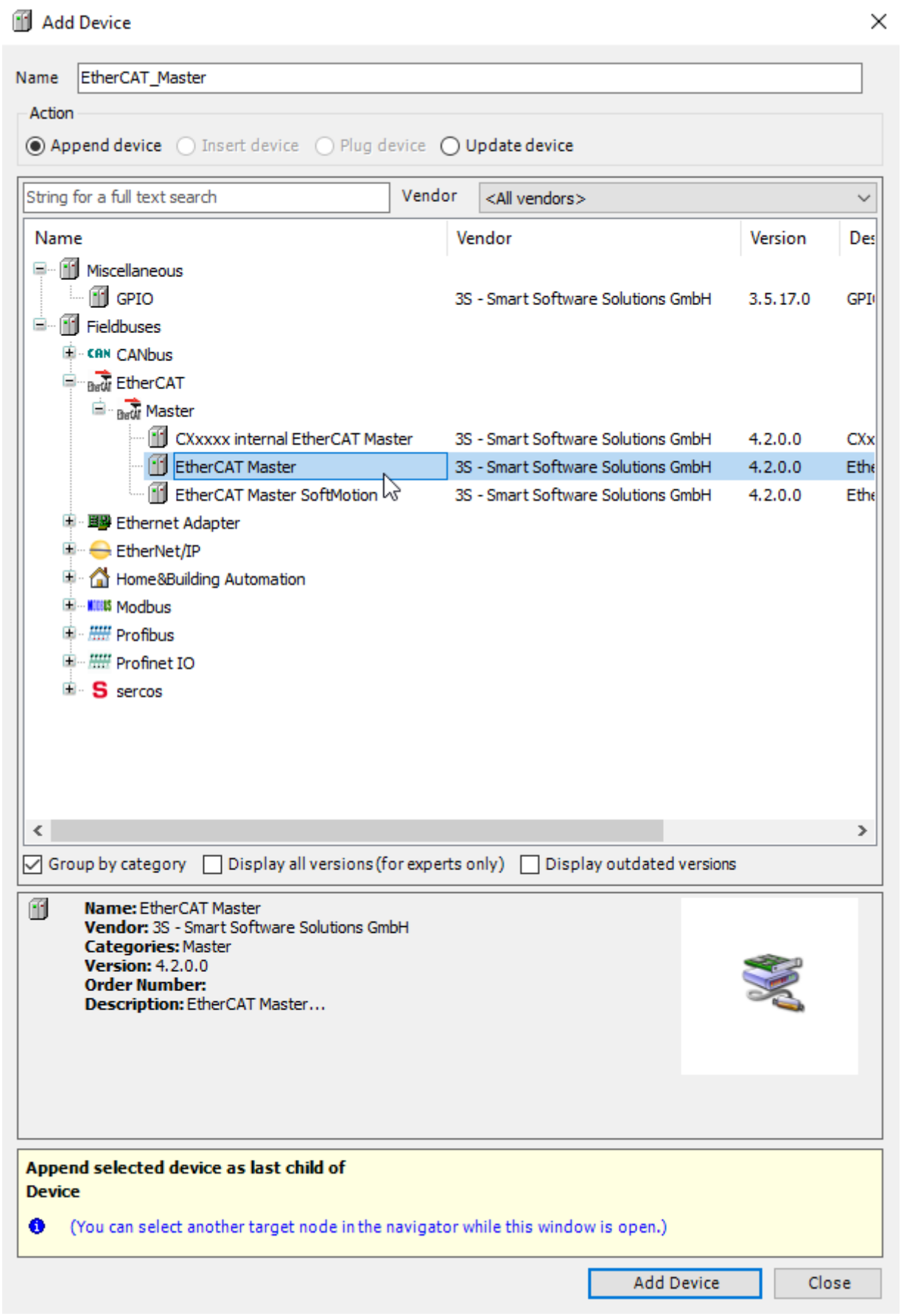
Configuration du Master EtherCAT

Pour créer la connexion entre le Master EtherCAT sur le Raspberry Pi et le coupleur EtherCAT Beckhoff BK1120 à l'aide de CODESYS :

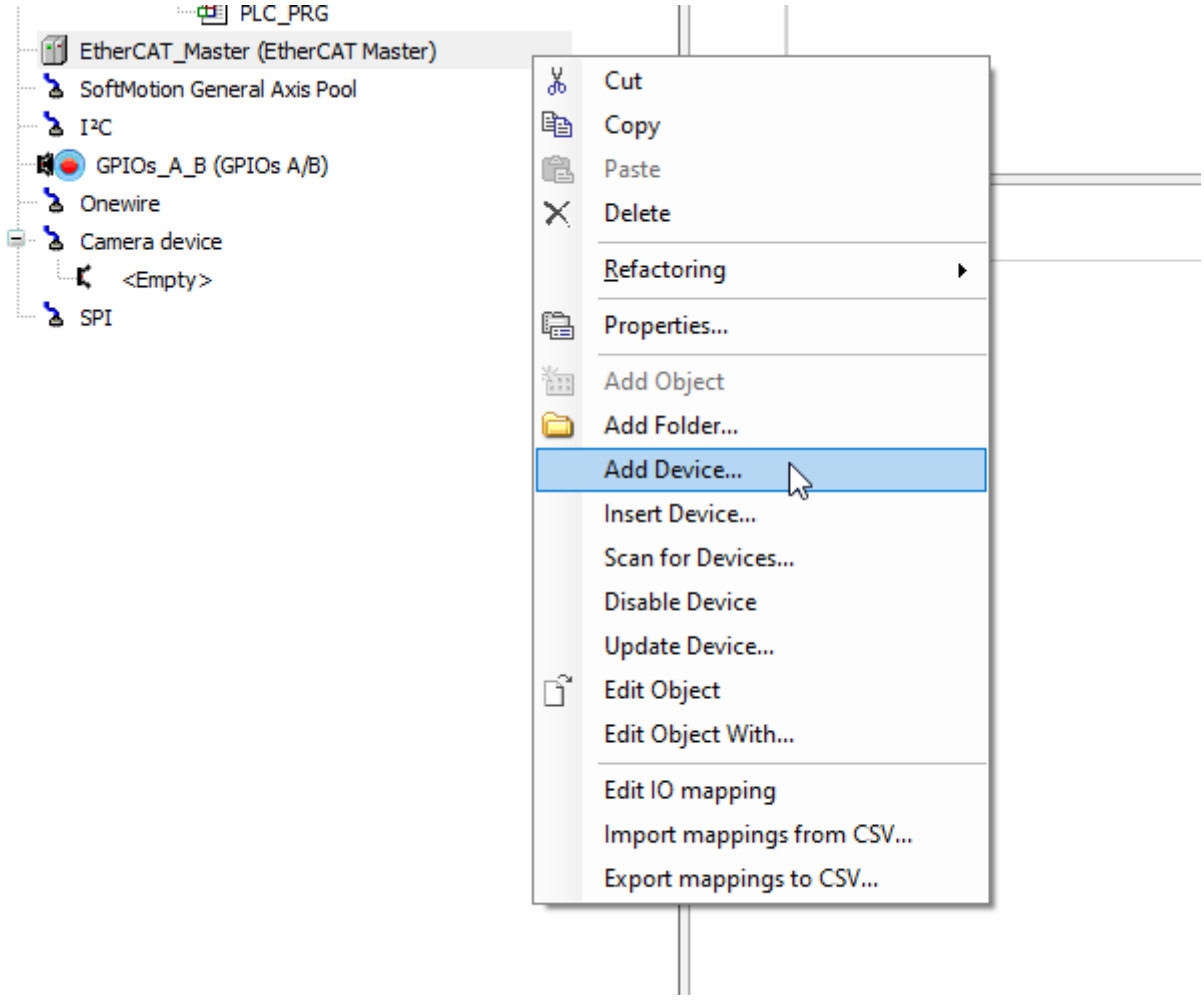
- Démarrez un nouveau projet dans CODESYS et connectez-vous au Raspberry Pi (cf article sur le Runtime CODESYS sur Raspberry Pi)
- Faites un clic droit sur "Device" dans votre arborescence d'appareils et sélectionnez "Add Device...".



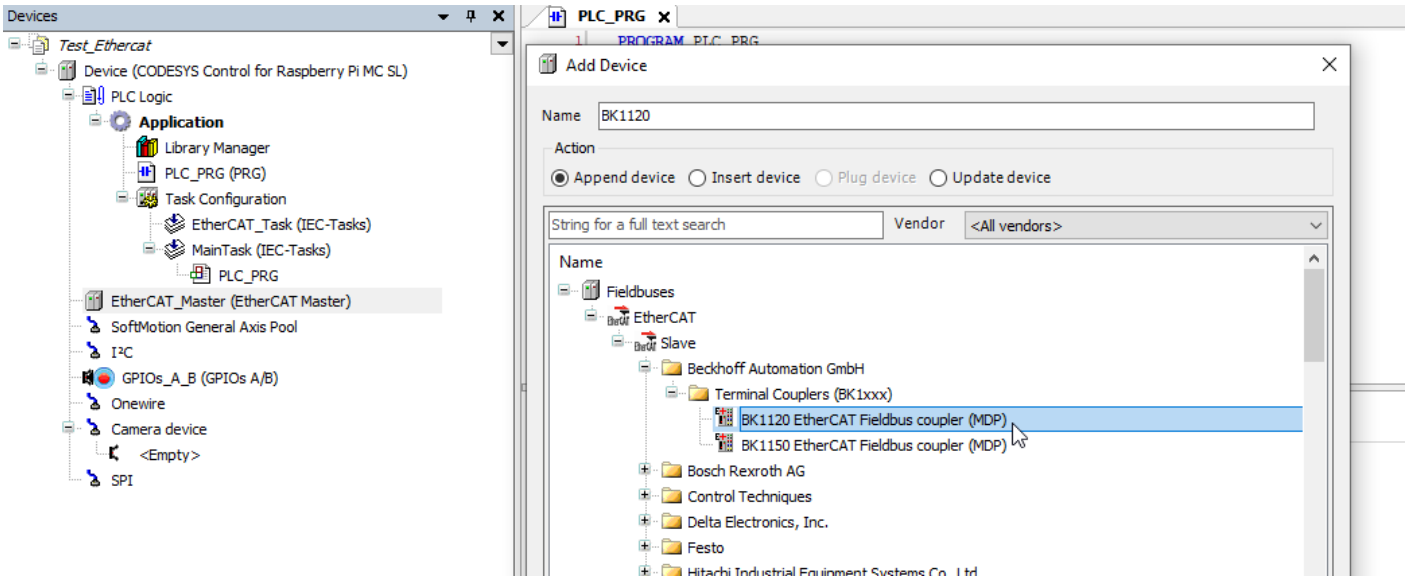
- Sélectionnez "EtherCAT Master" et cliquez sur "Ajouter un appareil".



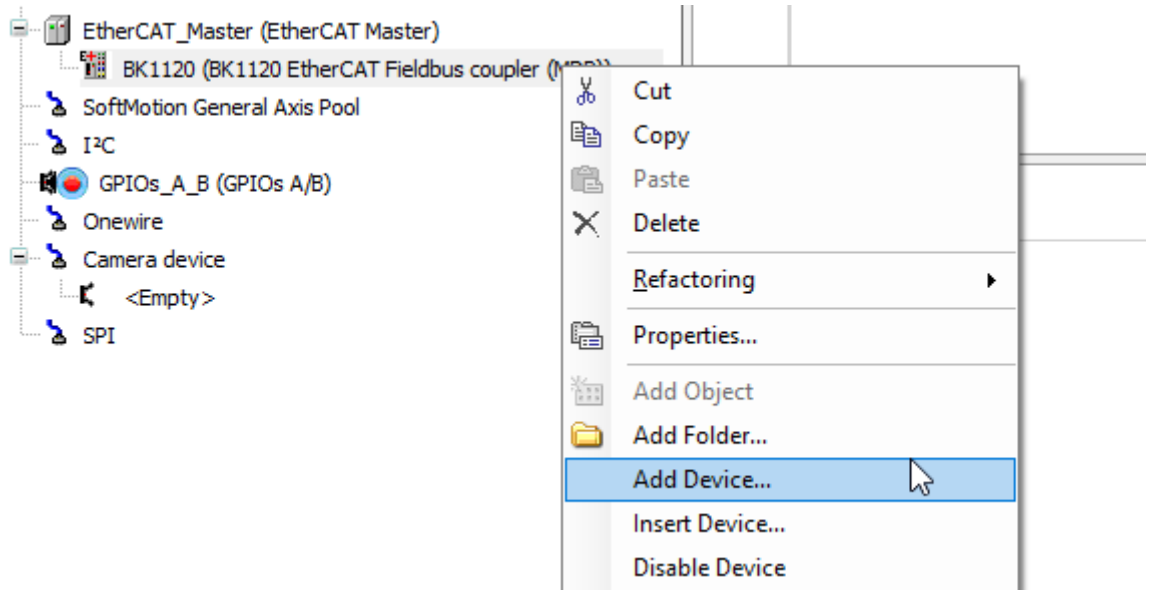
- Fermez la fenêtre et faites un clic droit sur "EtherCat_Master" dans l'arborescence de vos appareils et sélectionnez "Add Device...".



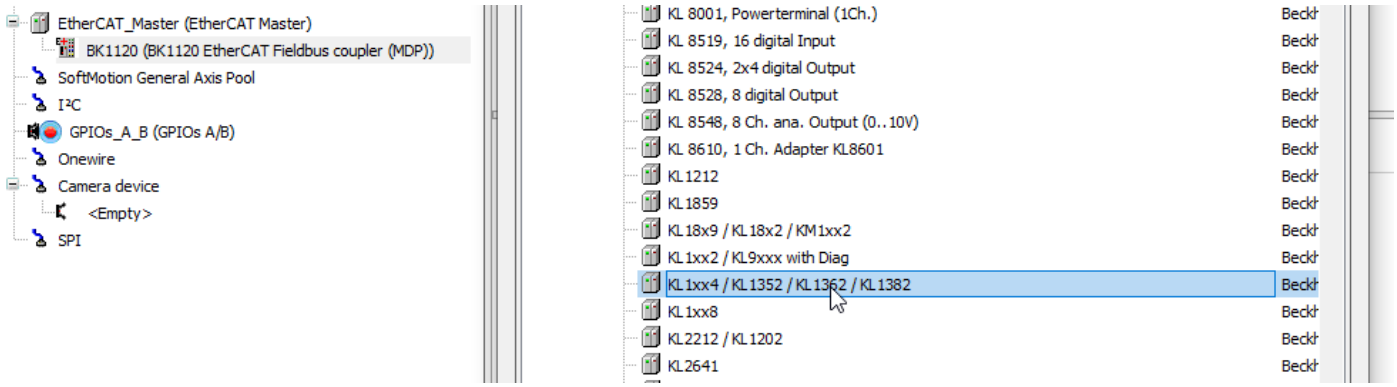
- Sélectionnez "BK1120" dans EtherCat > Slave > Beckhoff Automation System Couplers et cliquez sur "Add Device...".



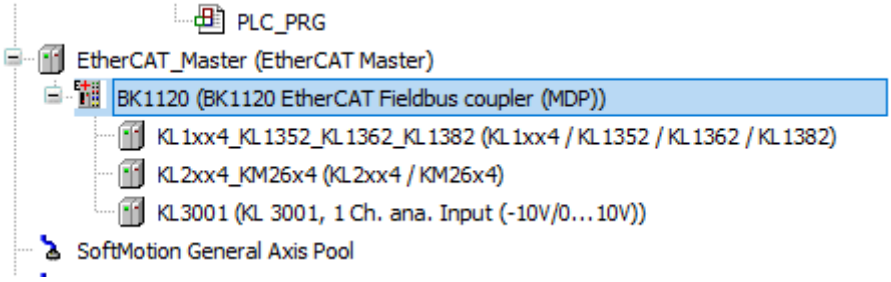
- Fermez la fenêtre et faites un clic droit sur "BK1120" dans l'arborescence de votre appareil et sélectionnez "Ajouter un appareil...".



- Sélectionnez le premier de vos modules IO, dans mon cas le "KL1114" depuis EtherCat > Slave > Beckhoff Automation Digital Input Terminals et cliquez sur "Add Device...".

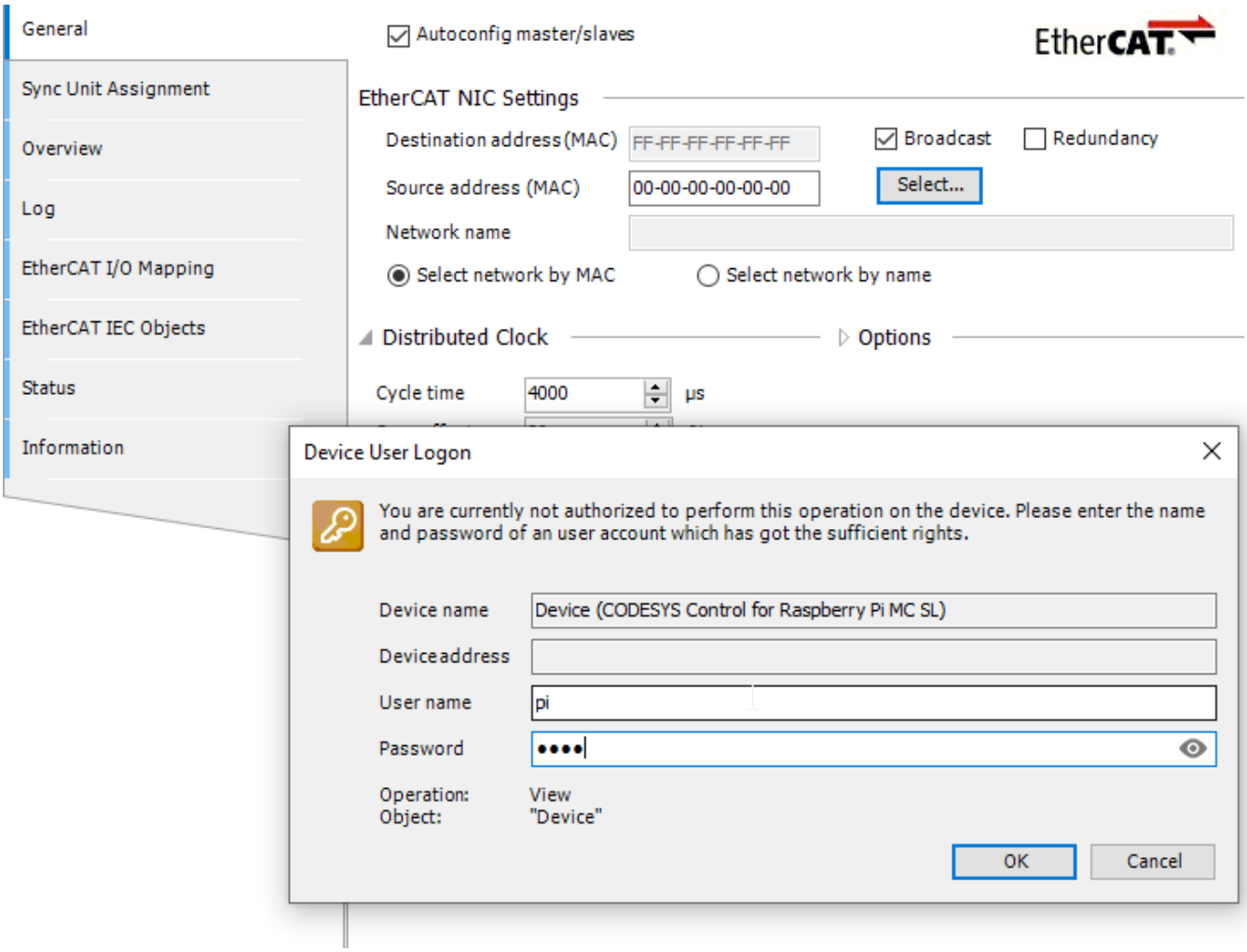


- Répétez l'étape ci-dessus pour toutes les E/S supplémentaires dans l'ordre de connexion au coupleur système.

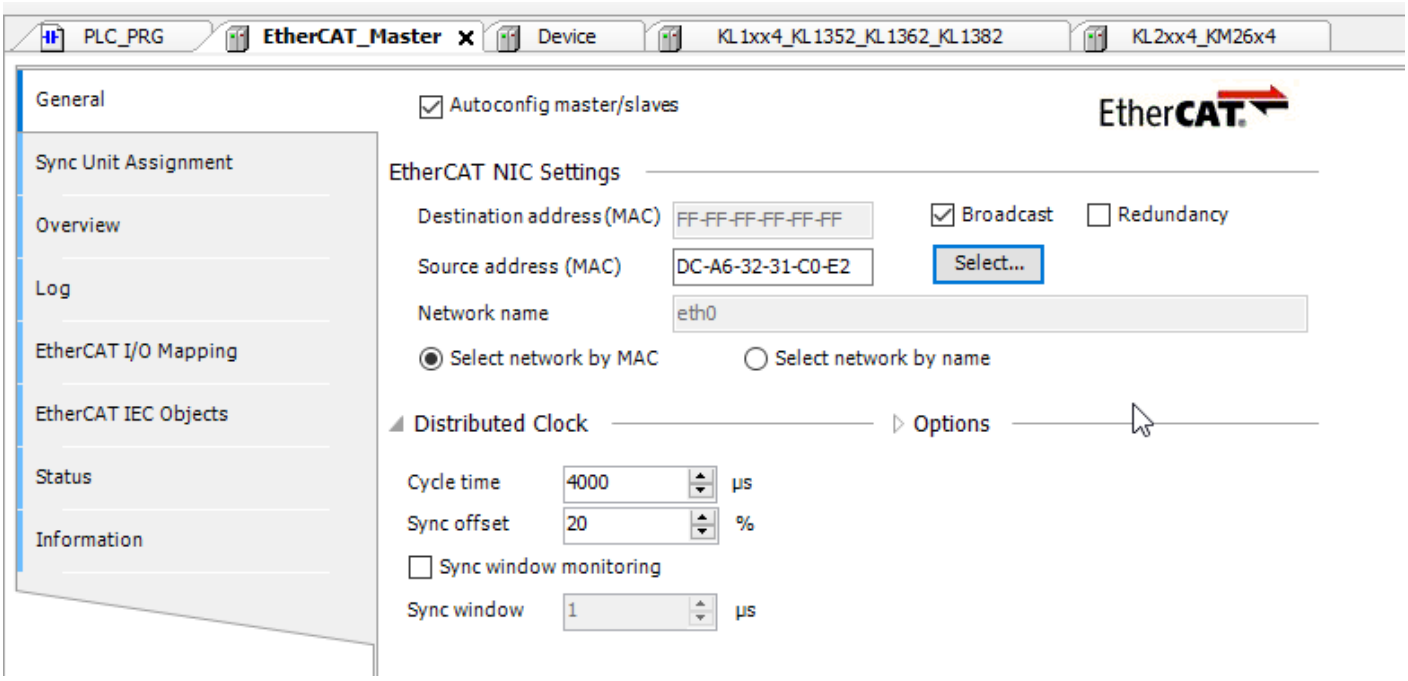


Nous devrons maintenant saisir l'adresse matérielle (MAC) du Raspberry Pi (relevée précemment) dans CODESYS :

- Depuis l'arborescence des appareils, double-cliquez sur "EtherCAT_Master".



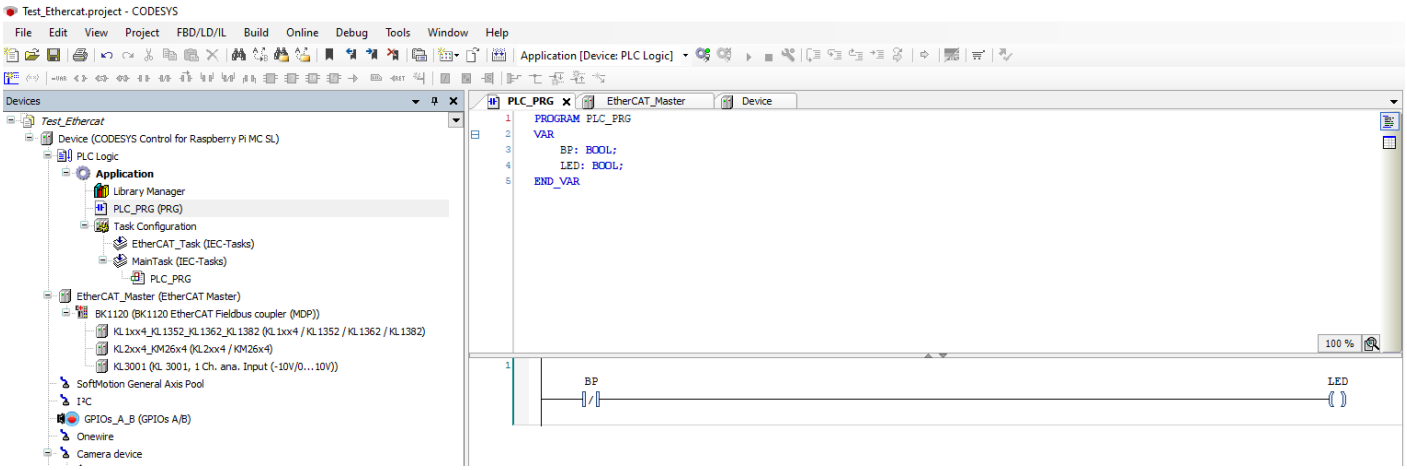
- Entrez l'adresse matérielle précédente dans le champ de saisie de texte "Adresse source (MAC)". (L'adresse MAC peut également être détectée directement sur le réseau.) Tout le matériel est maintenant configuré et nous sommes maintenant prêts à créer notre premier programme pour contrôler les E/S industrielles.



Programme MAIN sur le Master

Double-cliquez sur "PLC_PRG (PRG)" dans l'arborescence des équipements. Comme vu dans les articles précédents, nous ferons notre programme en Ladder.

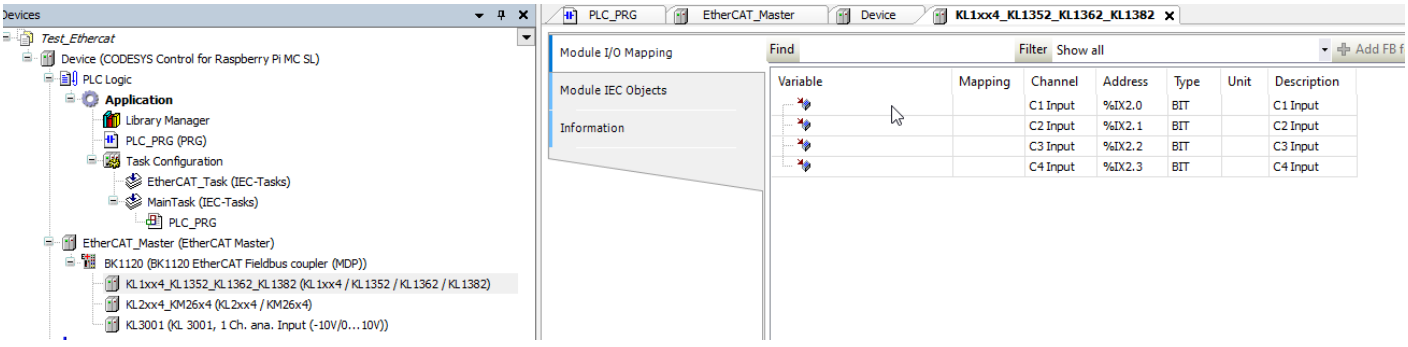
- Dans la partie haute de la fenêtre de programmation, nous déclarons les variables
- Dans la partie basse, il s'agit du programme en Ladder. Dans cet exemple, nous inversons l'état du contact BP pour activer ou non la sortie LED.



Affection des variables

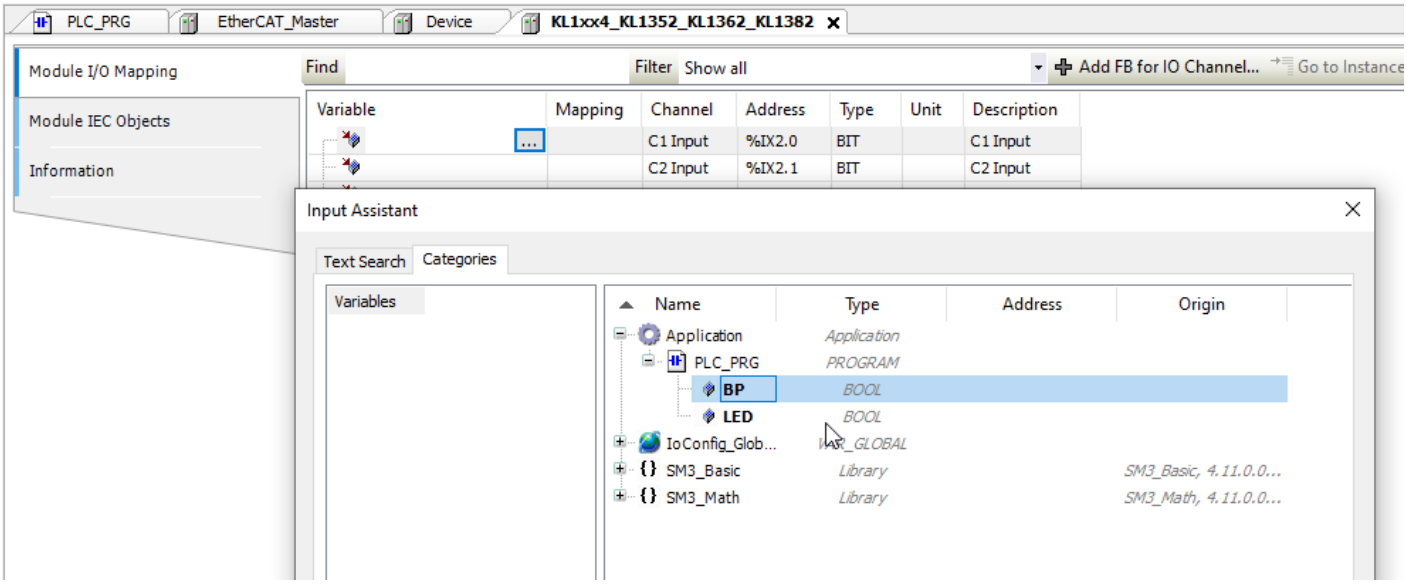
Nous allons maintenant lier les variables du programme MAIN aux entrées sorties du coupleur BK1120.

- Cliquer sur KL1xx4

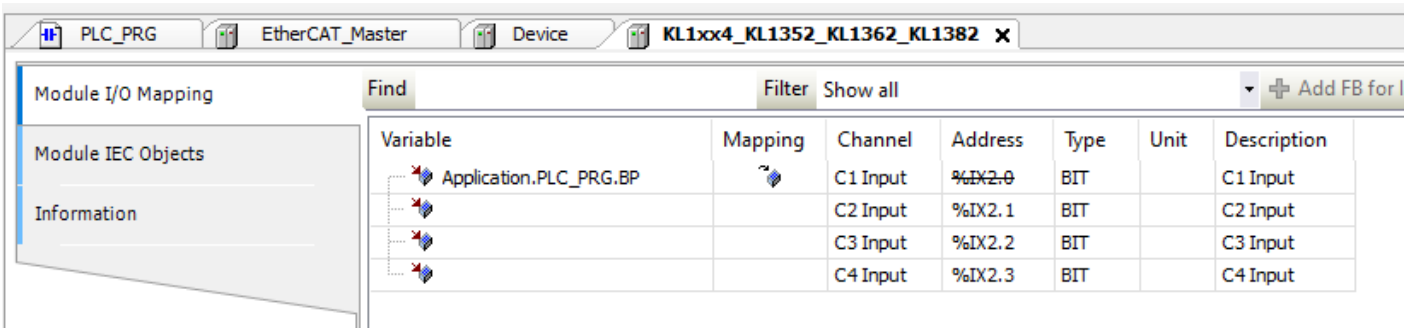


Nous devons maintenant lier notre Variable BP à "C1 Input" du Beckhoff IO

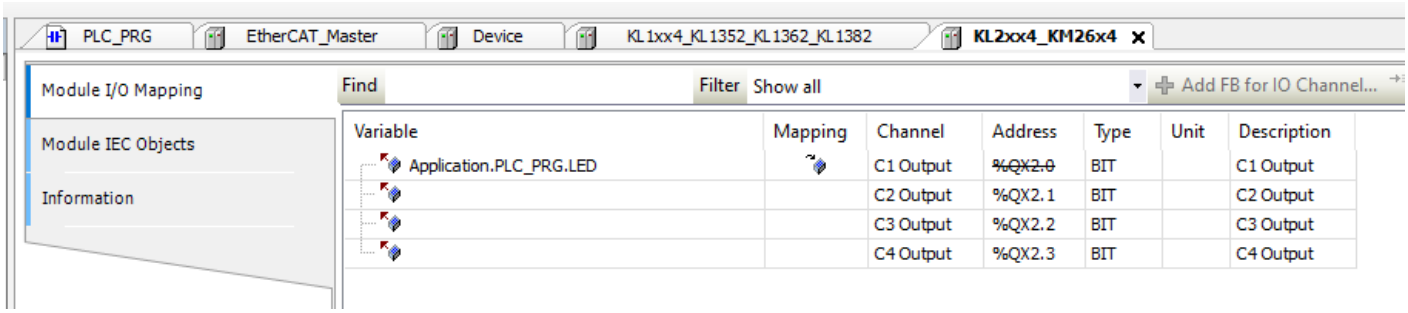
- Allez dans Module IO Mapping
- Cliquez sur les ...
- et cherchez la variable BP sous PLC_PRG



- Vérifiez que l'association soit bien prise en compte



- Nous allons maintenant associer la variable LED à la sortie C1 Ouptut du coupleur Beckhoff.



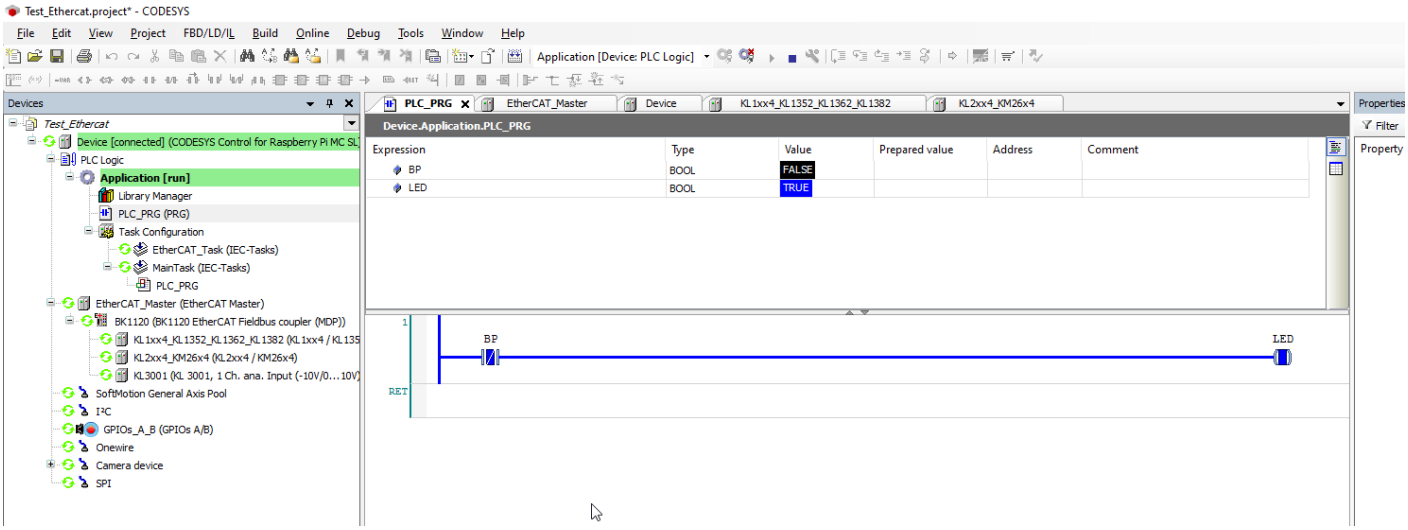
Test du programme

Pour exécuter le programme :

- Appuyez sur "F11" sur votre clavier, cela construit/compile le code. Également trouvé sous "Construire" dans la barre d'outils.
- Une fois le code compilé, appuyez sur "Alt + F8" sur votre clavier pour vous connecter au Pi (le PLC puisqu'il va assurer ce rôle). Cette commande se retrouve également sous "En

ligne" dans la barre d'outils.

- Cliquez sur "Oui" dans la fenêtre contextuelle pour télécharger le programme.
- Une fois le programme téléchargé, appuyez sur "F5" pour lancer le programme, fonctionnalité que l'on retrouve sous "Debug" dans la barre d'outils.
- Vous devriez maintenant voir la LED de sortie 1 sur le terminal Beckhoff s'allumer ou s'éteindre en fonction de l'appui sur le l'interrupteur BP.



- Pour arrêter le code appuyez sur " Maj + F8" sur votre clavier, fonctionnalité que l'on retrouve également sous "Debug" dans la barre d'outils.

Conclusion

Nous pouvons maintenant contrôler des coupleurs d'E/S industriels à l'aide d'un Raspberry Pi en tant que Master EtherCAT. Néanmoins, il est illusoire de penser qu'il est possible d'atteindre les perfomances temps réel avec le Runtime Codesys comme sur un IPC Beckhoff qui est prévu pour. Pour des applications classiques d'automatisme sans besoin de synchroniser des axes moteurs, les performances offertes par ce type de dispositif sont équivalents à celles offertes par une communication Profinet I/O.

Revision #2
Created 29 June 2023 16:53:53 by Philippe Celka
Updated 4 July 2023 09:18:43 by Philippe Celka