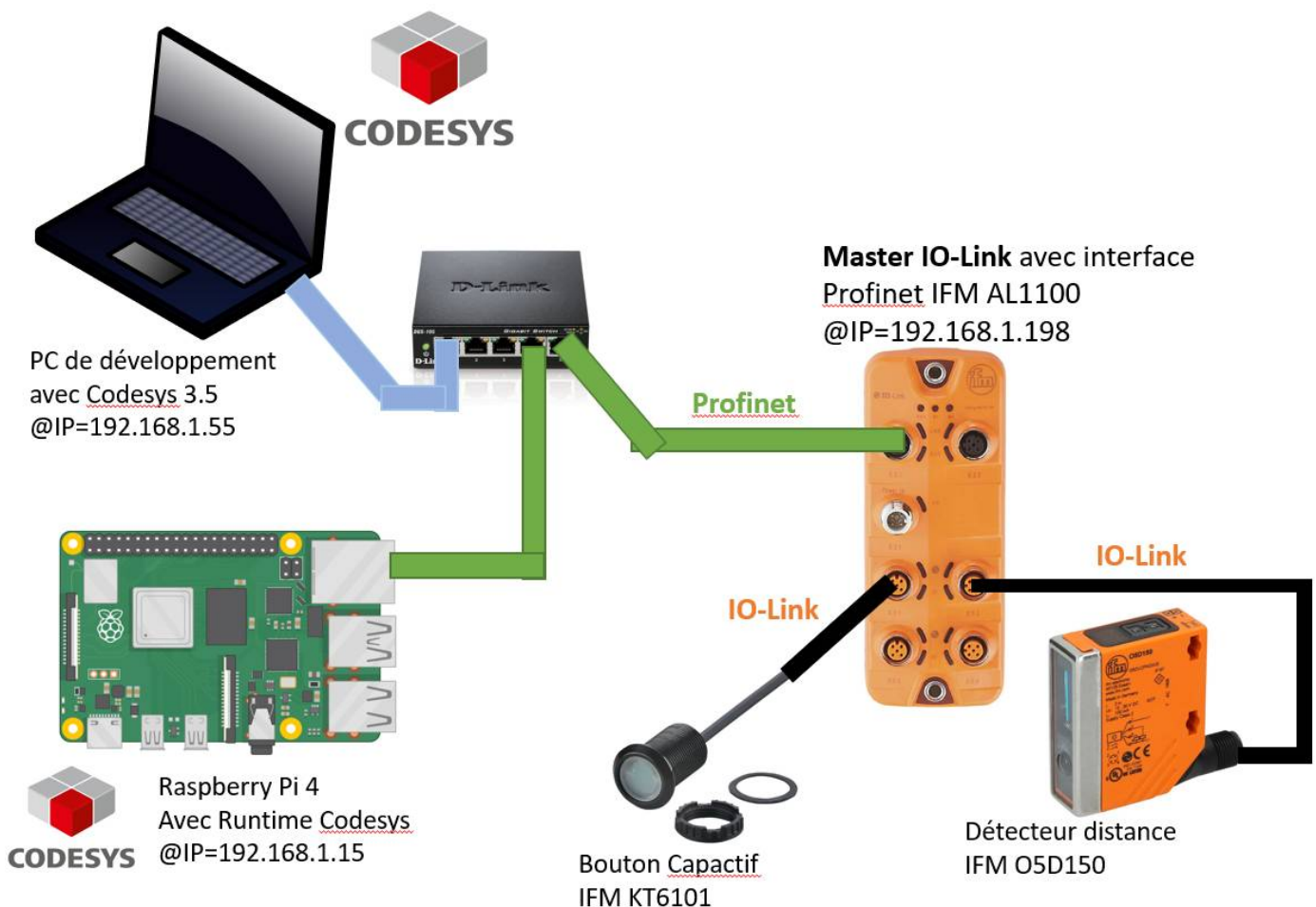


# Master IO-Link sur Raspberry Pi

## Master IO-Link Raspberry Pi

Cet article fait suite à la communication entre TwinCAT 3 et un Master IO-Link Profinet. Il est également possible de connecter un Master IO-Link Profinet sur un Raspberry Pi 4 avec le Runtime Codesys. Le schéma du montage utilisé est représenté ci-dessous :



- Le Raspberry Pi va constituer notre PLC. Le Runtime Codesys est installé dessus. Il va gérer la communication Profinet avec le Master IO-Link Profinet AL1100 d'IFM.
  - @IP du Raspberry Pi est 192. 168. 1. 15
- Le Master IO-Link Profinet AL1100 d'IFM va être associé à deux capteurs IO-Link

- Détecteur de distance O5D150
- Bouton capacitif KT6101
- @IP du Master IO-Link est `192. 168. 1. 198`
- Le PC de développement va permettre de générer le code PLC et le transférer sur le Raspberry Pi. Contrairement à TwinCAT où le Runtime est exécuté sur le PC, dans cette application, le Runtime est sur le Raspberry Pi.
  - @IP du PC de dev : `192. 168. 1. 55`

### Prérequis :

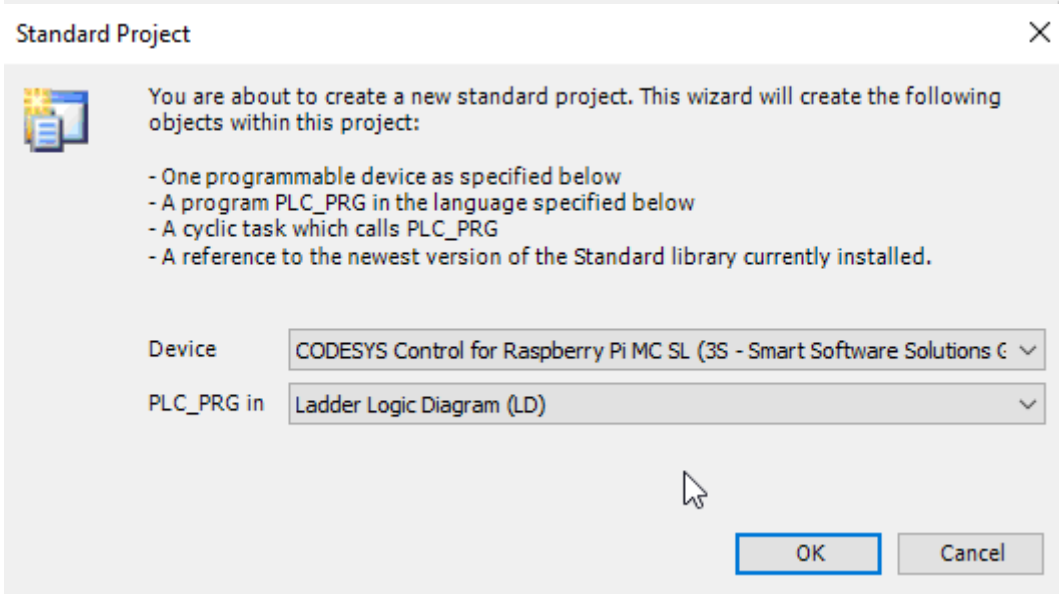
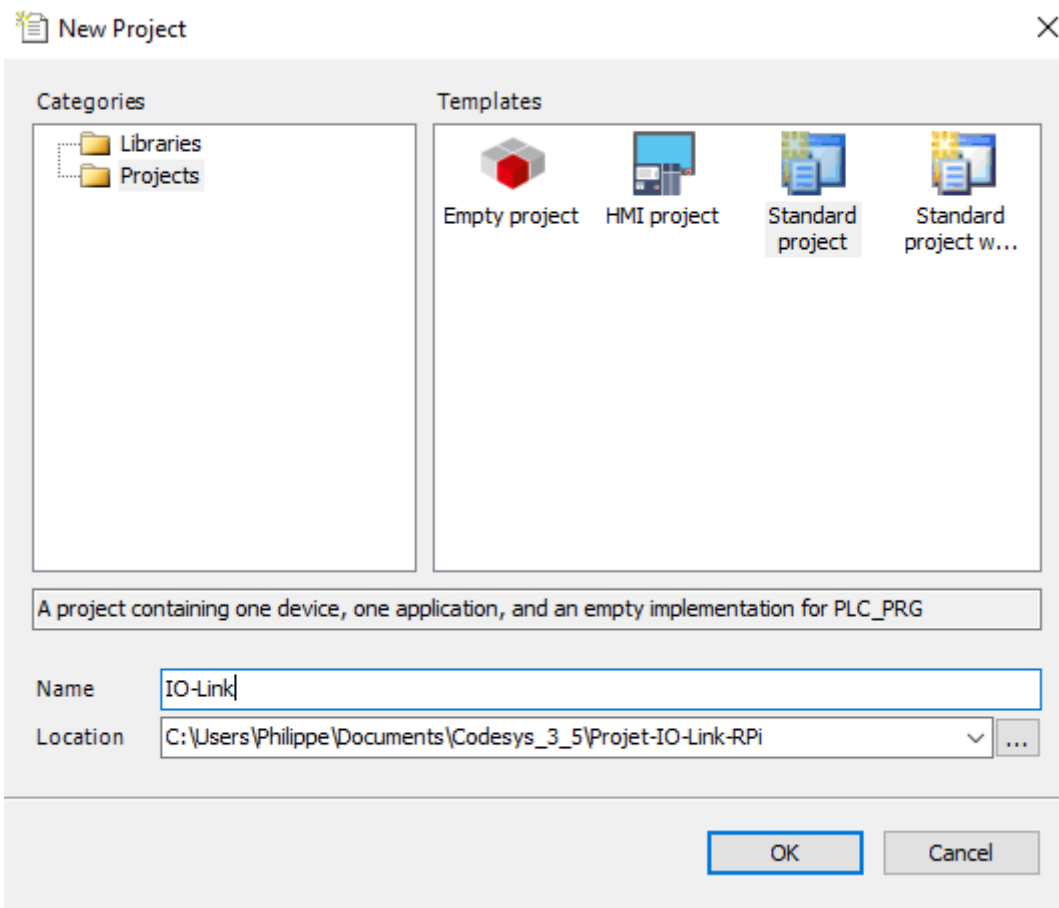
- Codesys 3.5 installé (et fonctionnel) sur un PC de développement -> cf article
- Runtime Codesys installé sur un Raspberry Pi -> cf article

Il est également nécessaire d'avoir lu l'article sur la communication entre le Master IO-Link Profinet et TwinCAT car plusieurs aspects vont se retrouver ici :

- les fonctions en langage ST pour le décodage des capteurs
- l'association des variables

# Projet Codesys

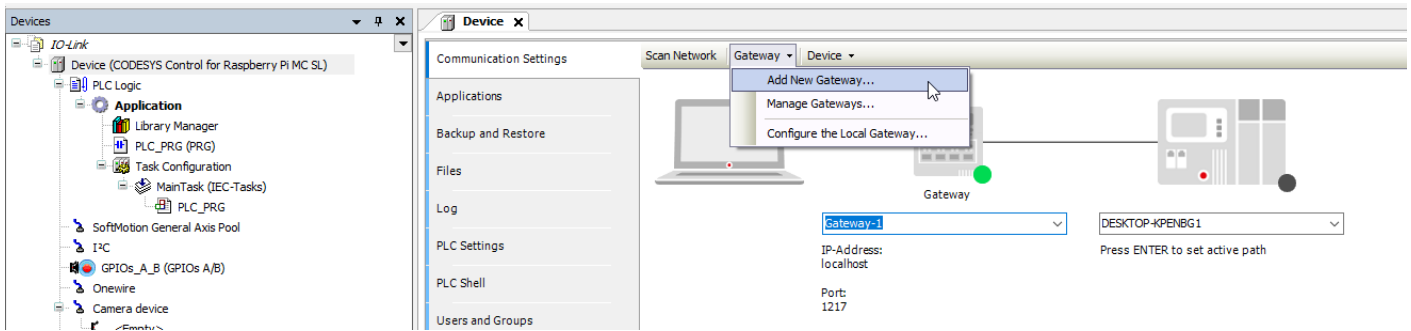
- Créer un Standard New Project
- Choisir pour le Device Codesys Control for Raspberry PI MC SL
- Choisir Ladder Logic pour PLC\_PRG



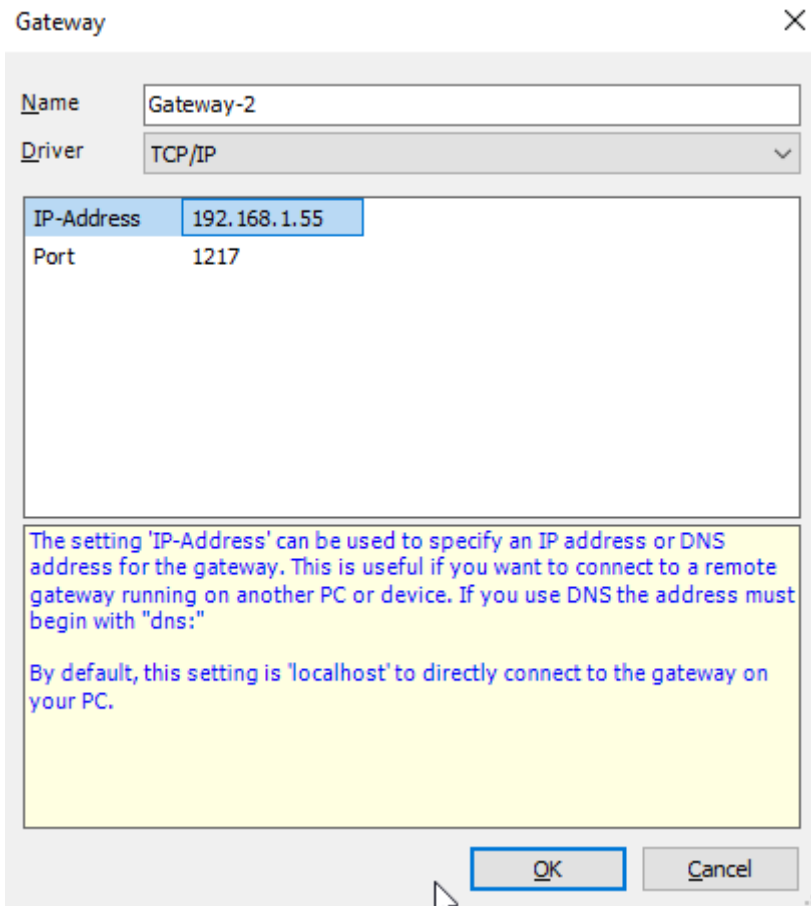
## Gateway et Raspberry Pi

En double cliquant sur Device, dans les Communications Settings:

- Gateway -> Add New Gateway

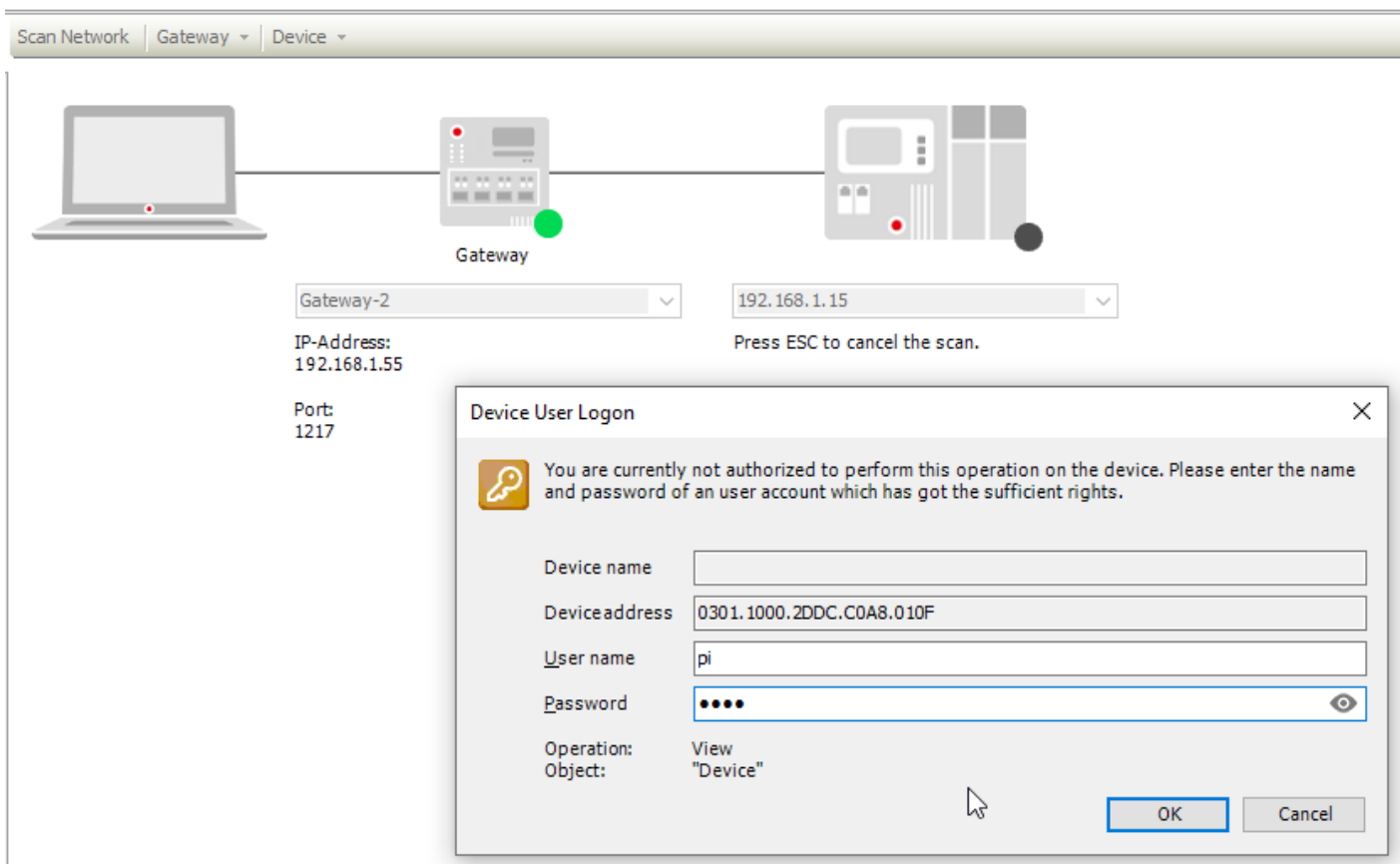


- Placer l'adresse IP du PC développement (@IP=192.168.1.55 dans mon cas)

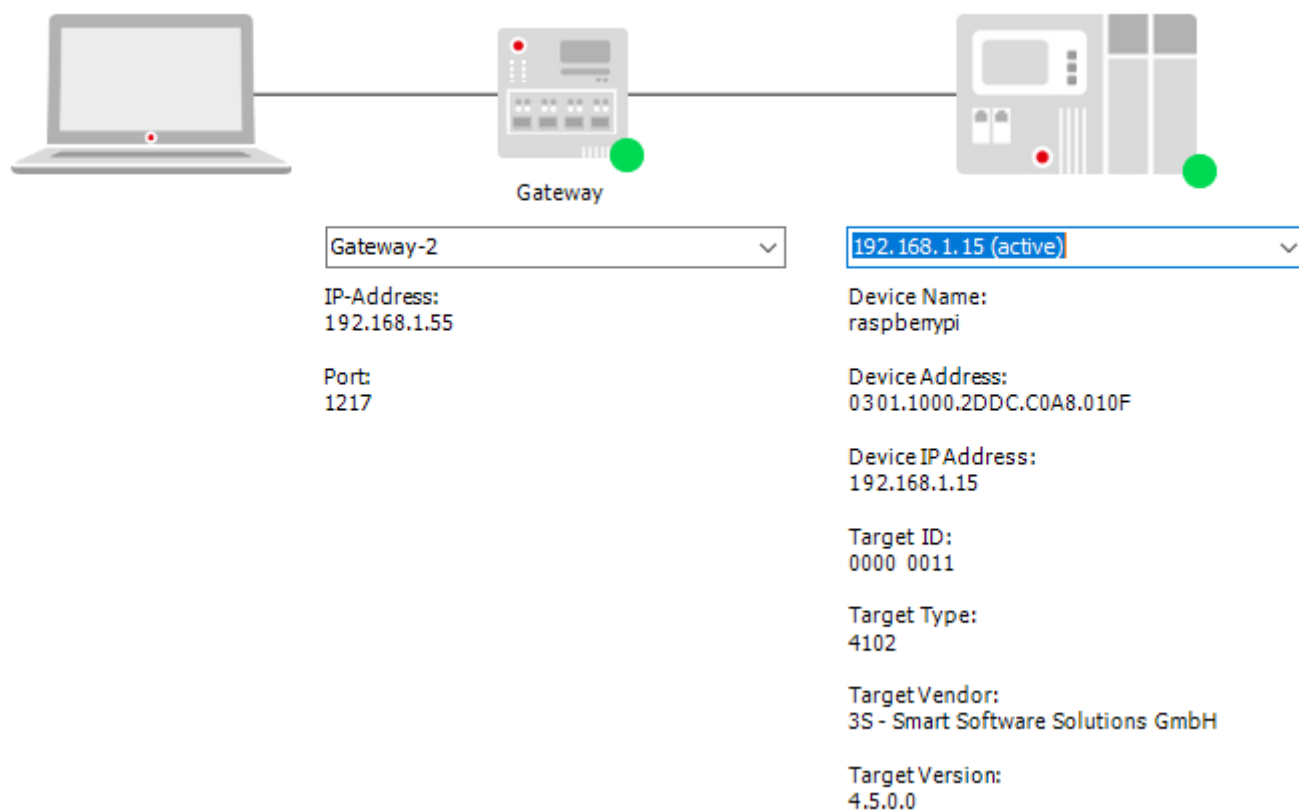


À droite de la Gateway,

- placer l'adresse IP du Raspberry Pi
- en appuyant sur Entrée, un fenêtre de Device User Logon apparaît :
  - mettre le User Name du Raspberry pi -> pi
  - mettre le Password -> 3.14 dans mon cas



La liaison avec le Raspberry Pi doit passer au vert et indiquer (active)

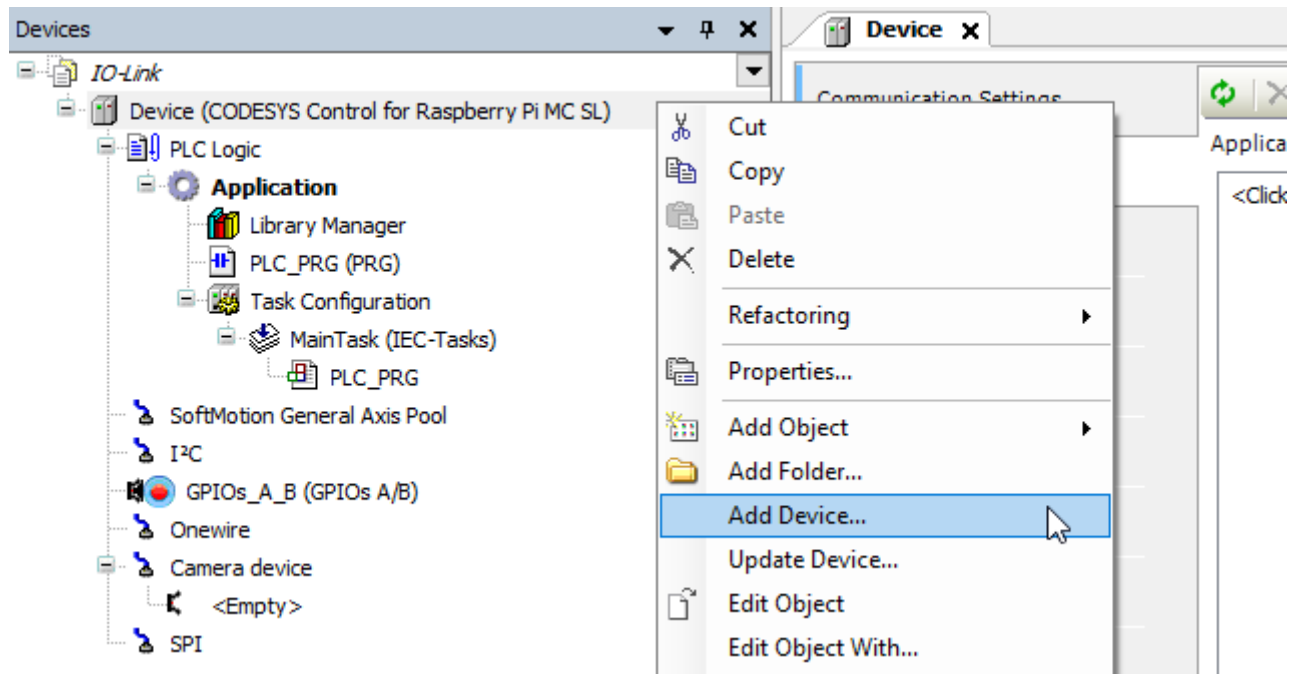


Si tout est au vert, on peut passer à la liaison Profinet

# Configuration de la liaison Profinet

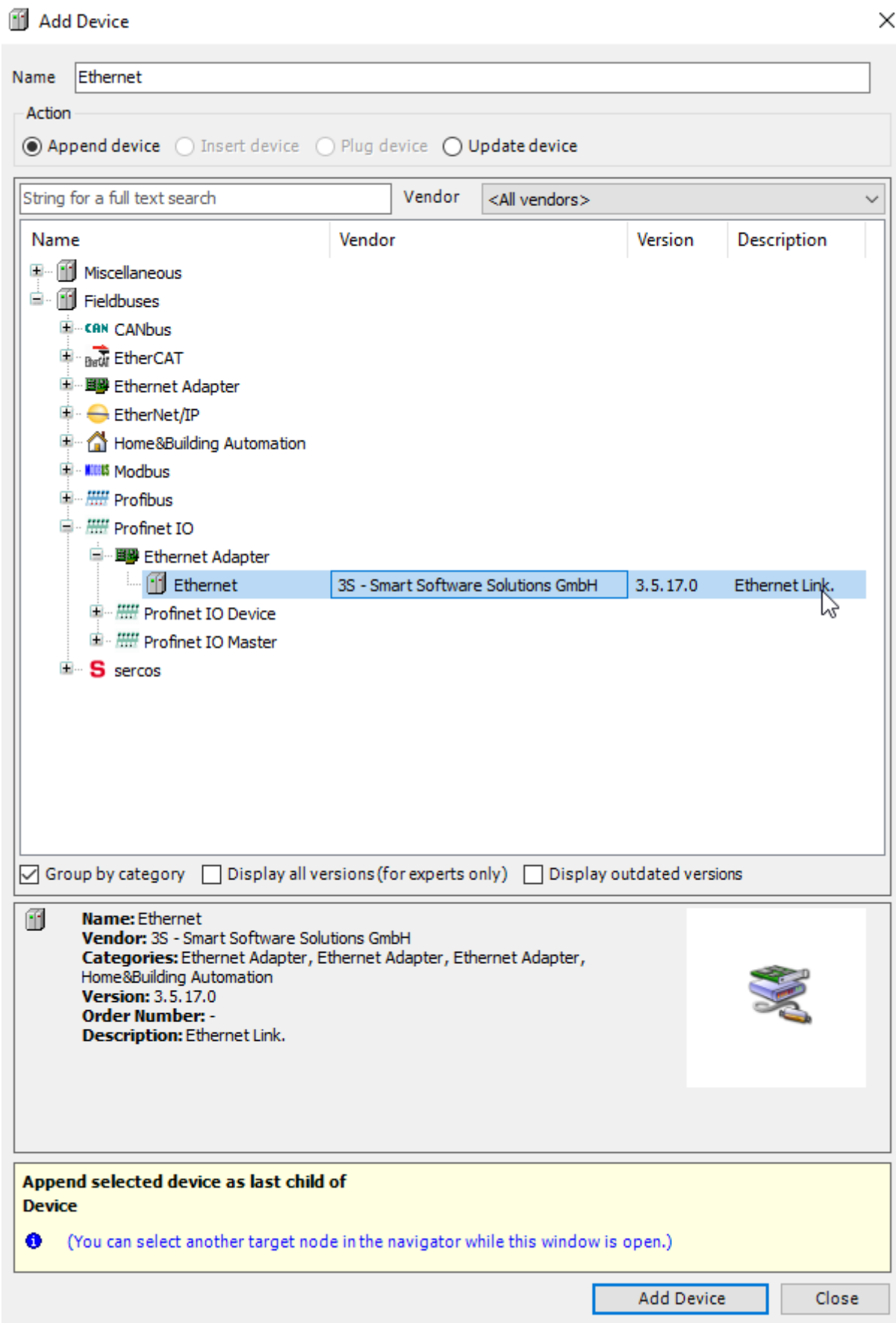
Faire un clic droit sur Device :

- Add Device



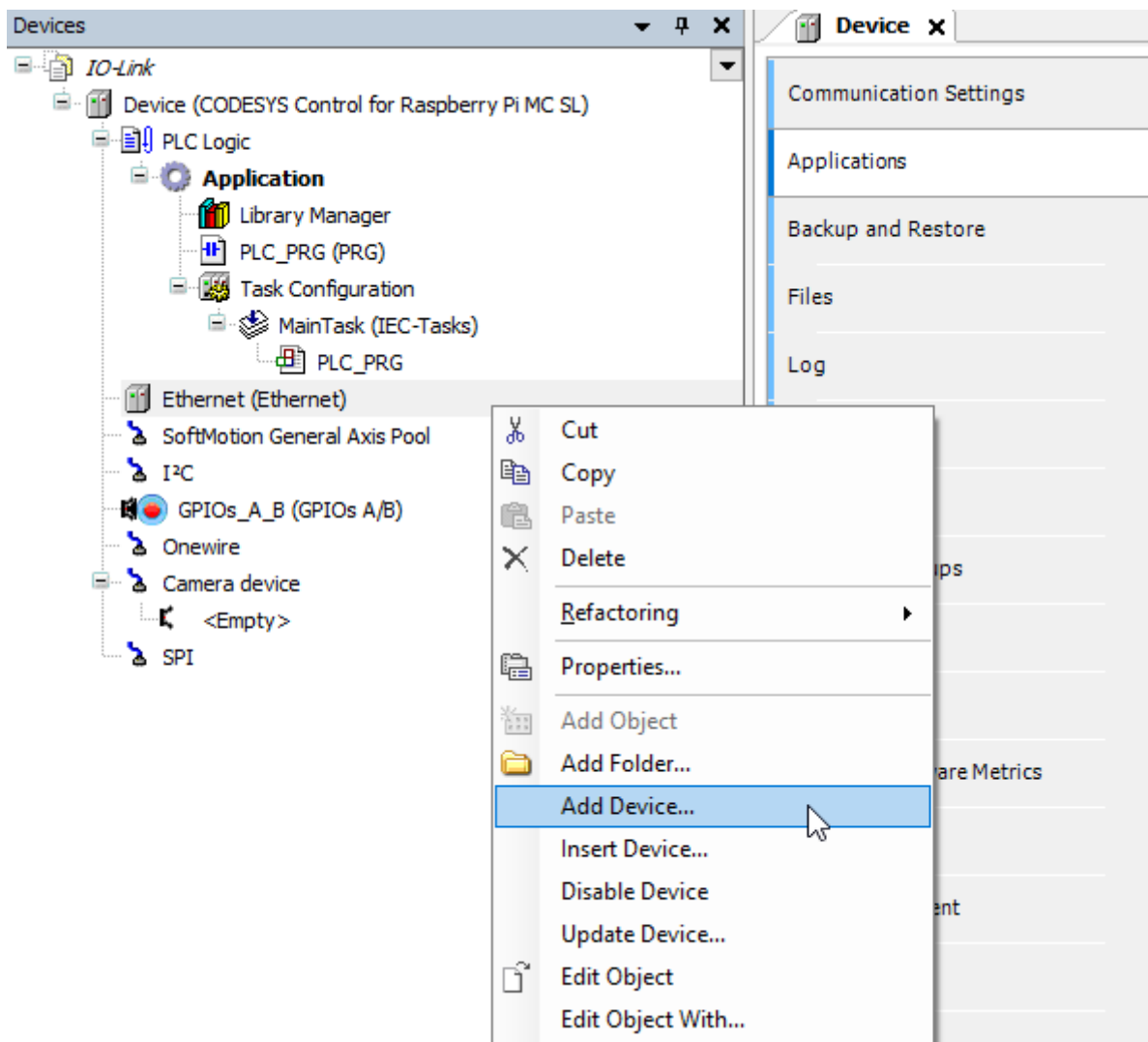
Dans l'arborescence Profinet IO :

- Choisir Ethernet
- et faire Add Device



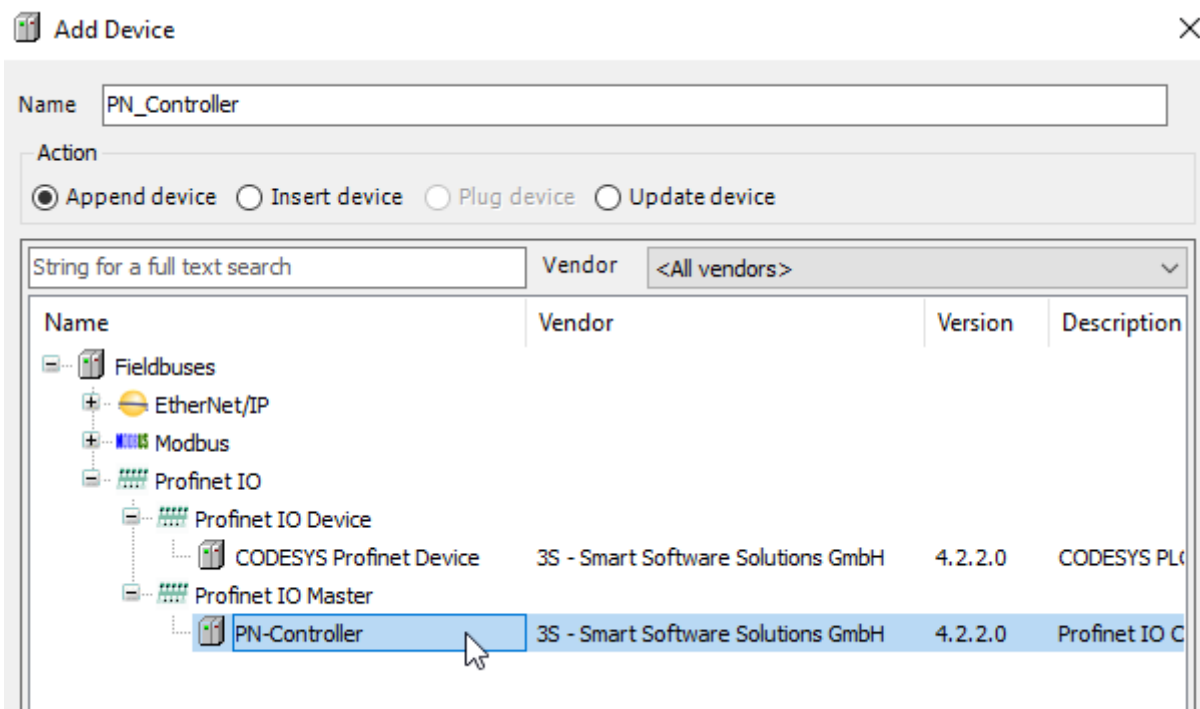
Sur Ethernet(Ethernet), faire un clic droit :

- Add Device



Dans l'arborescence Profinet IO Master :

- Choisir PN-Controller

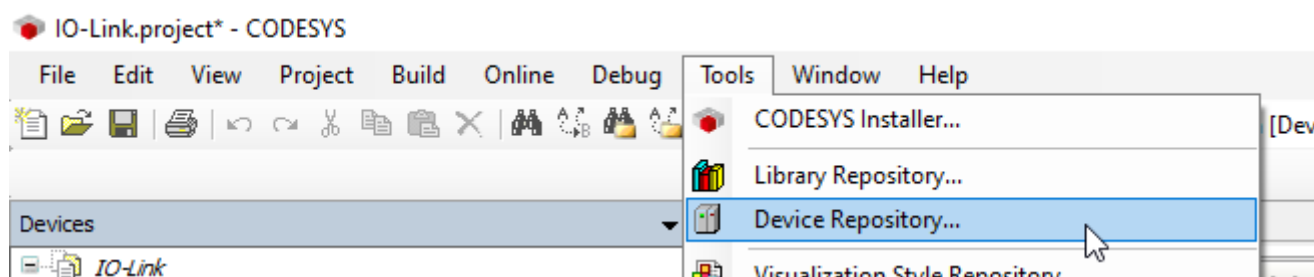


## GSDML du Master IO-Link Profinet d'IFM

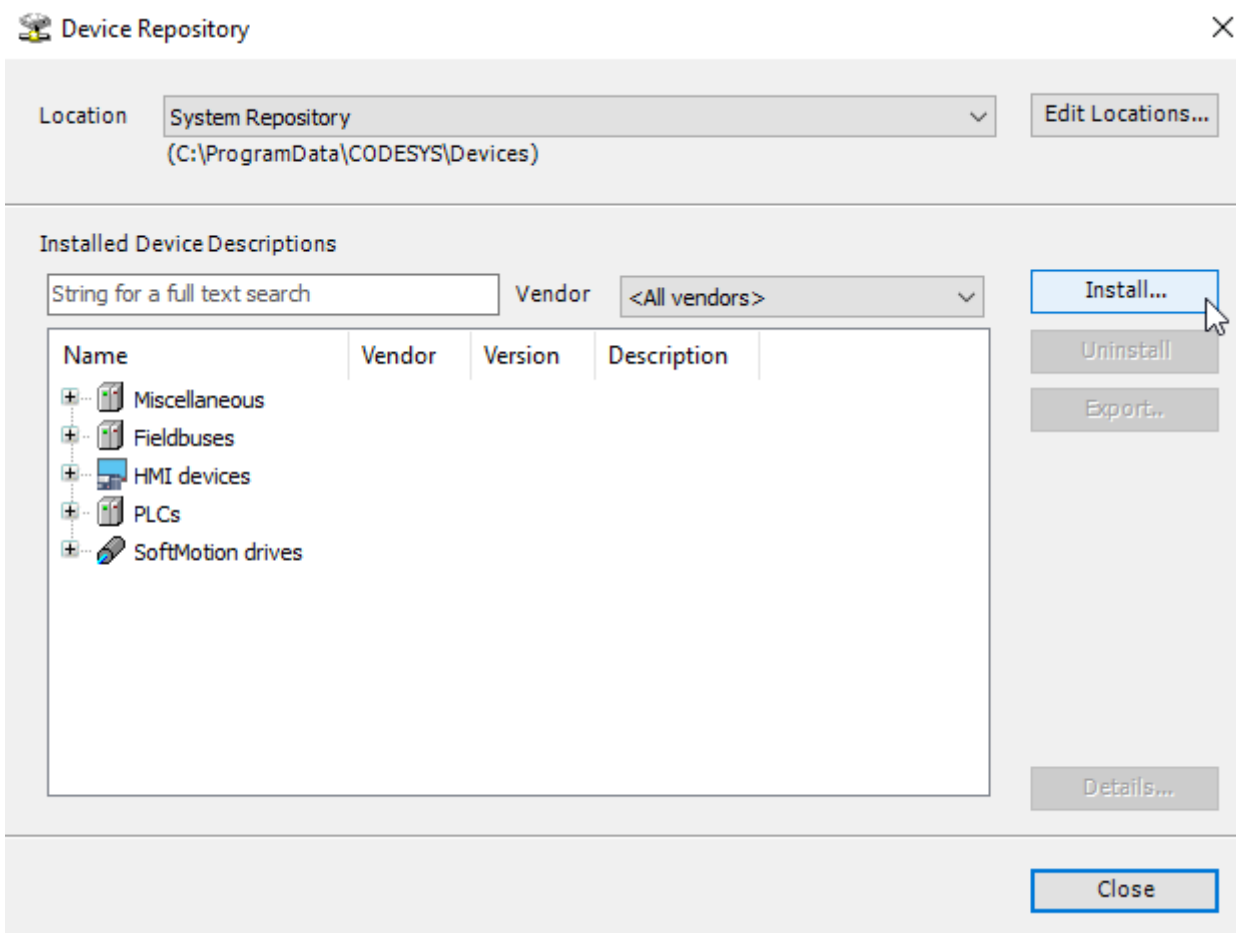
Comme pour TIA Portal, le Master IO-Link Profinet n'est pas installée de base dans Codesys. Il est nécessaire d'ajouter le fichier GSDML correspondant dans le Device Repository.

Dans l'onglet Tools, faire :

- Device Repository

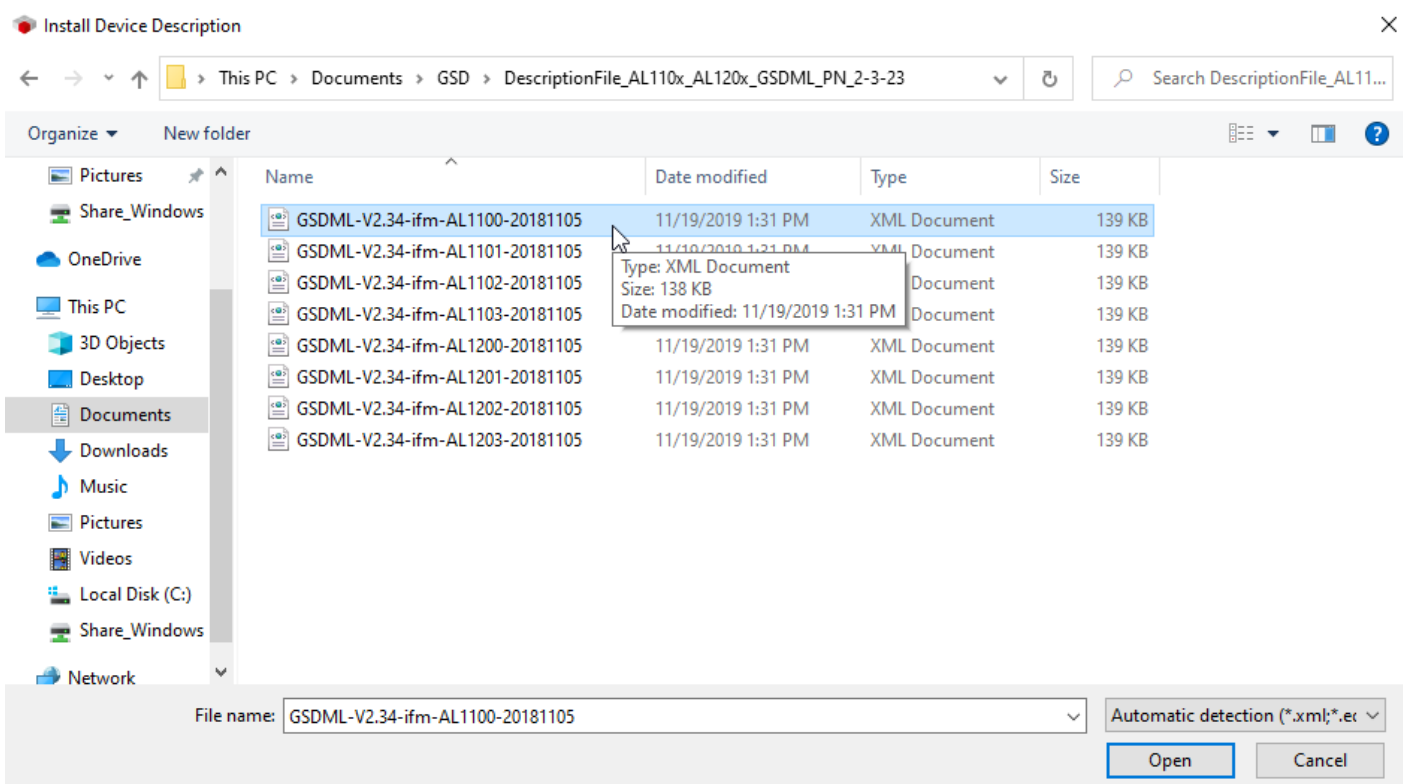


Cliquer sur Install



et sélectionner le GSDML correspondant à votre Master IO-Link. Pour rappel, le fichier GSDML du Master IO-Link Profinet AL1100 se télécharge directement chez IFM.

- après avoir sélectionné le fichier, faire open.



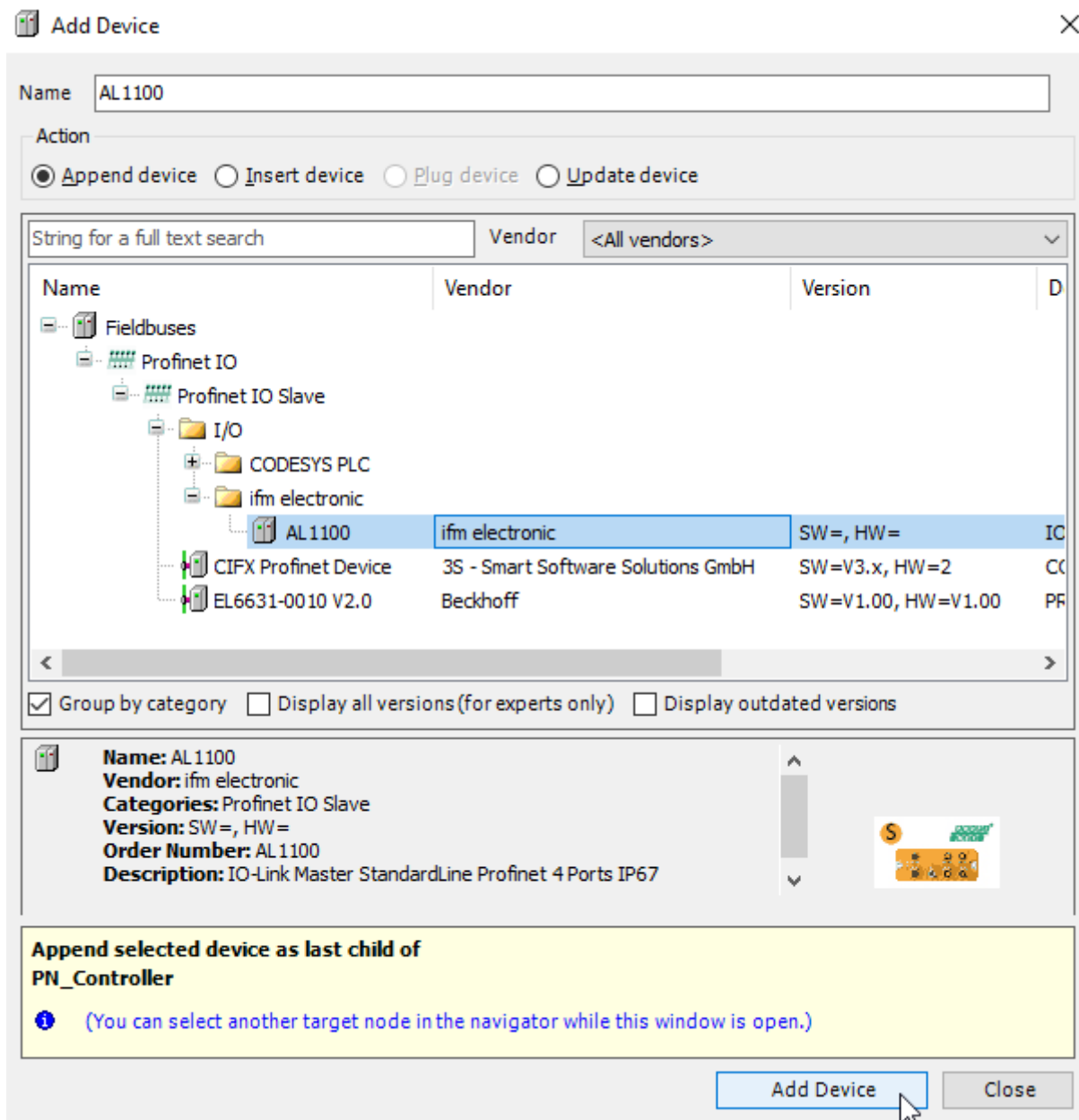
# Ajout du Master IO-Link

En cliquant sur PN\_Controller (PN-Controller), faire

- Add Device

Dans l'arborescence Profinet IO -> IO -> ifm electronic, choisir

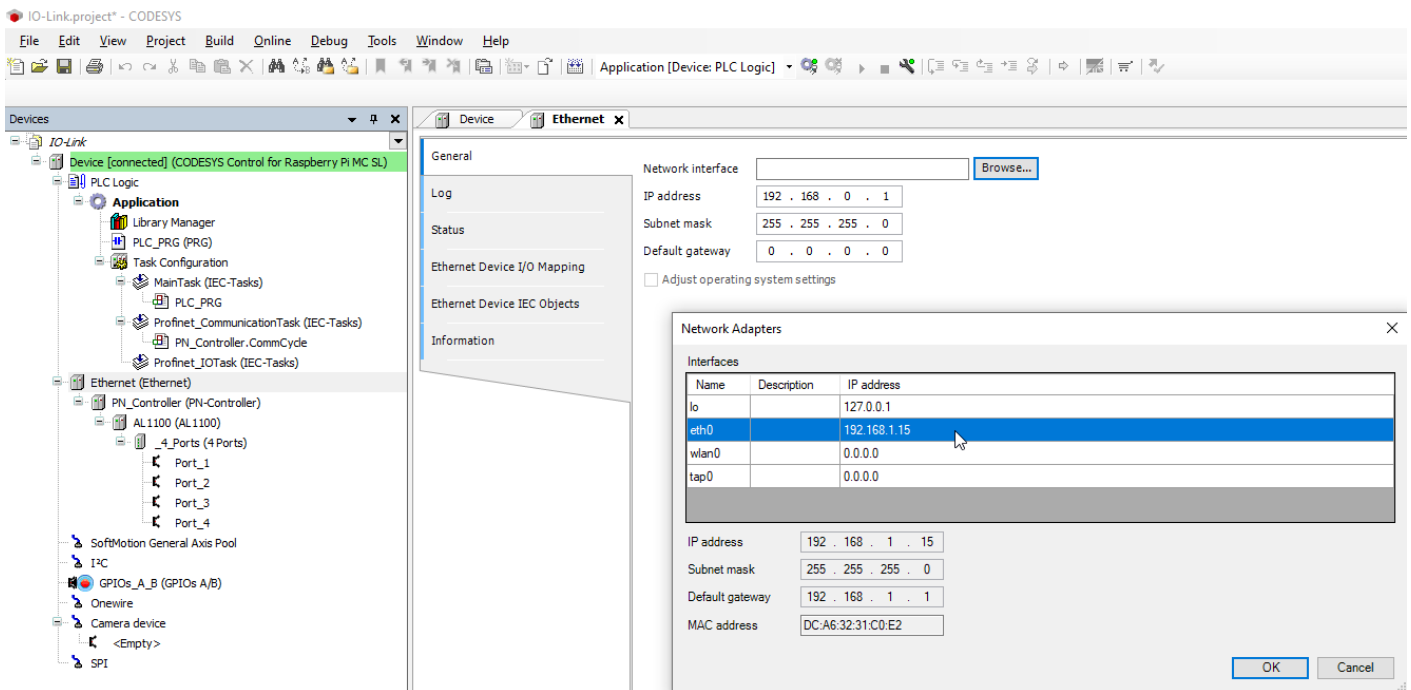
- AL1100, ifm electronic
- faire Add Device



## Configuration Profinet et IO-Link

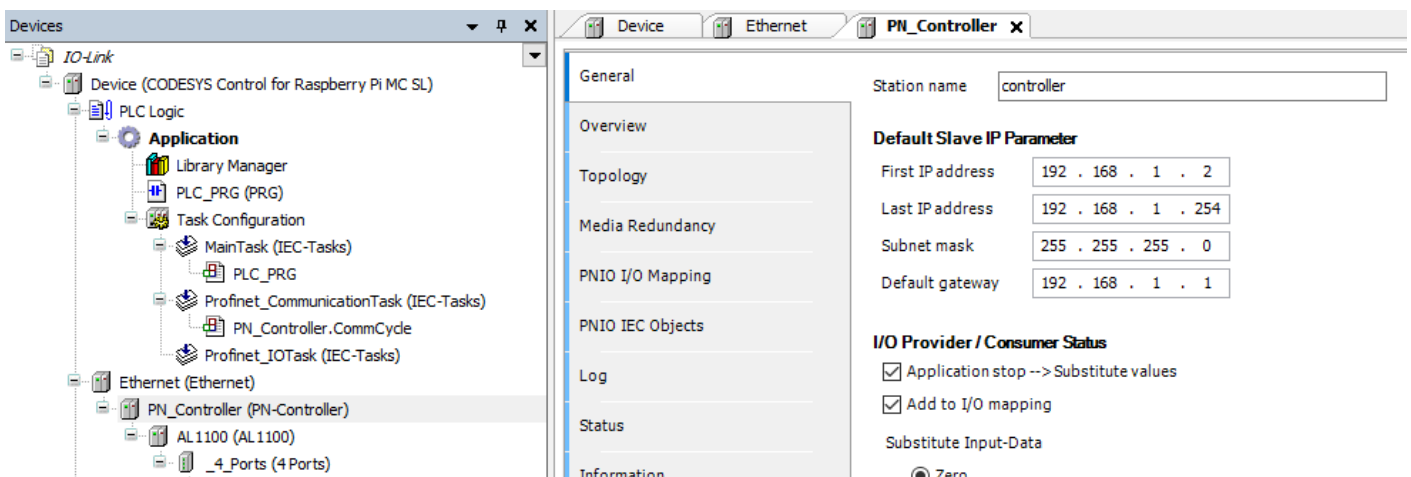
Double cliquer sur Ethernet (Ethernet)

- Dans Network interface, faire Browse
- une fenêtre Network Adapters apparaît
  - choisir l'interface `eth0` qui correspond à l'interface ethernet du Raspberry Pi



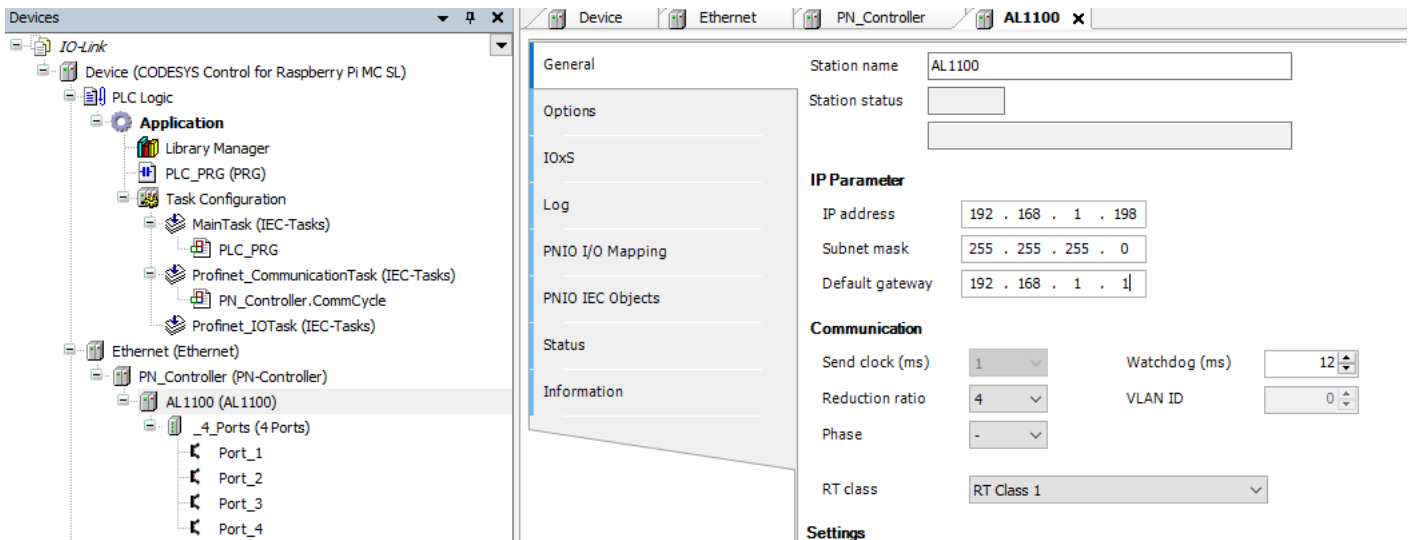
Double cliquer sur PN\_Controller et modifier la plage d'adresses pour les Slave

- First IP Address : 192.168.1.2
- Last IP Address : 192.168.1.254
- Subnet Mask : 255.255.255.0
- Default Gateway : 192.168.1.1



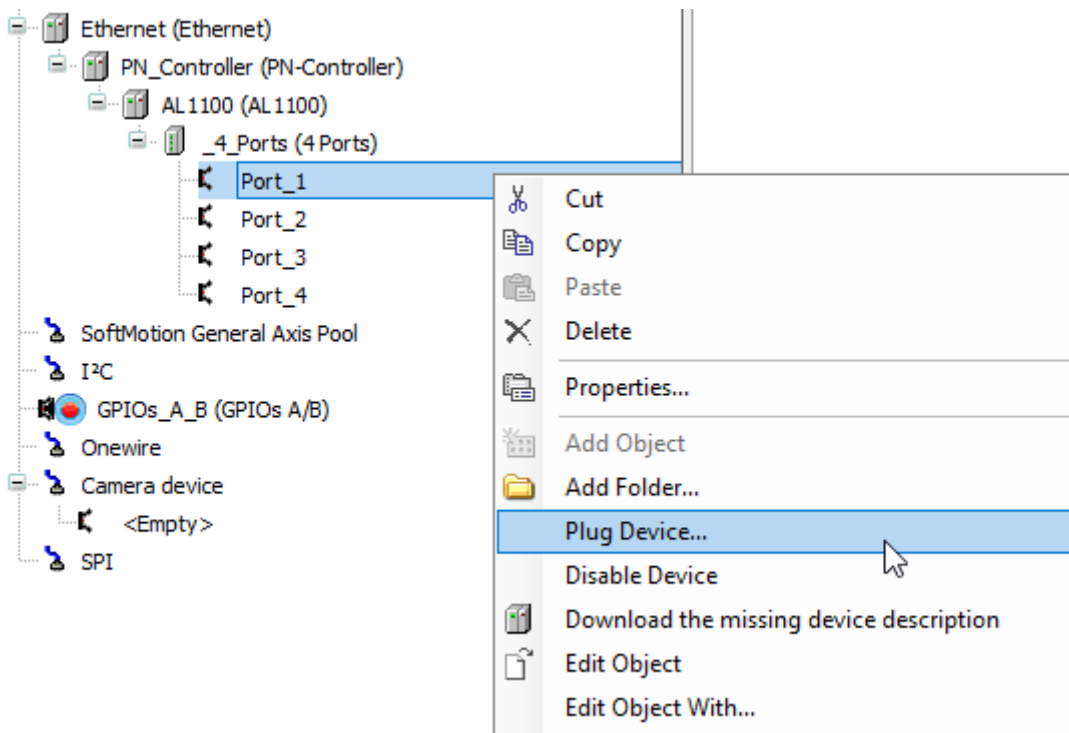
Double cliquer sur AL1100 (AL1100) et modifier les IP Parameter

- IP Address : 192.168.1.198
- Subnet Mask : 255.255.255.0
- Default Gateway : 192.168.1.1



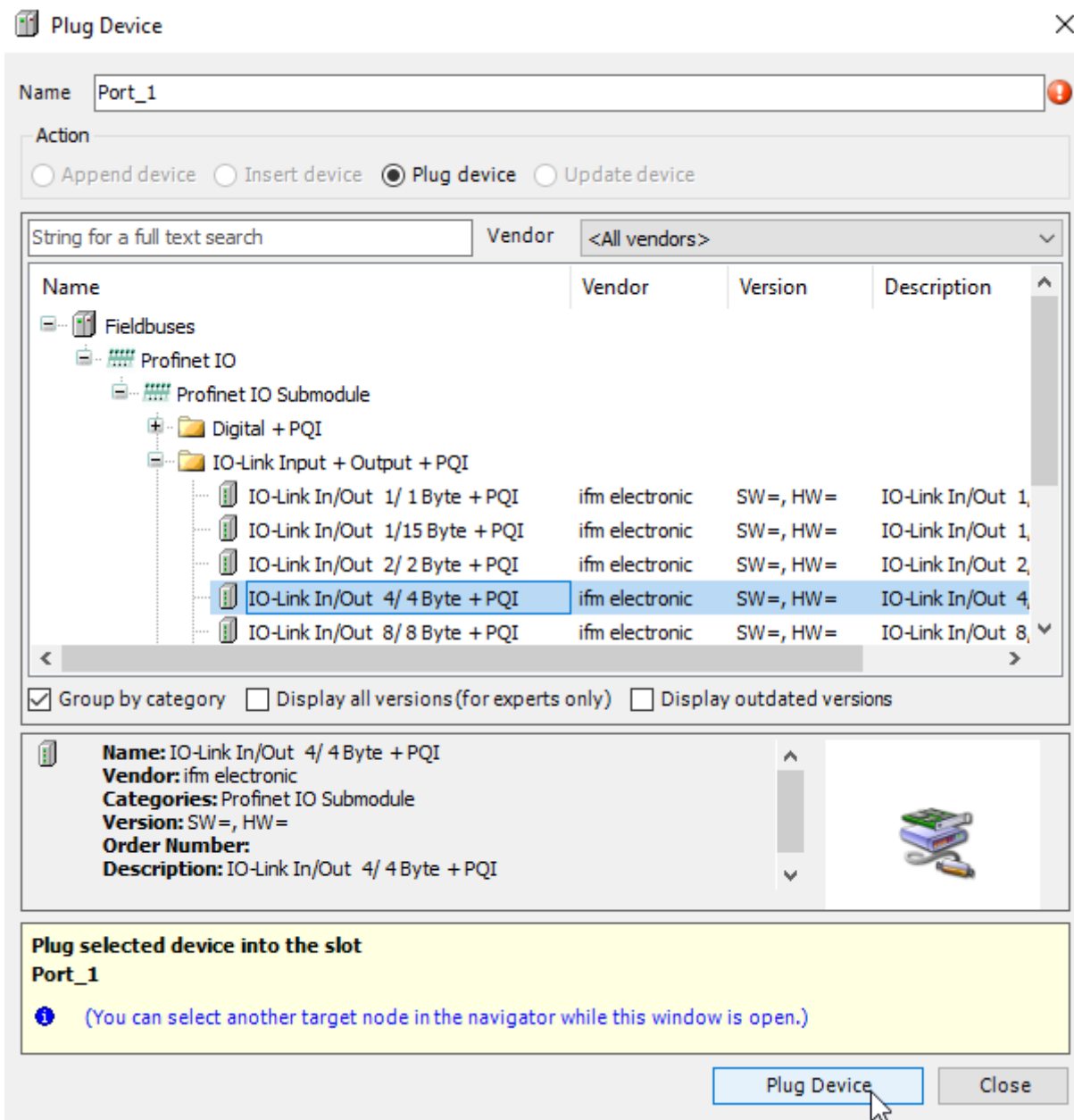
Dans \_4\_Ports (4 Ports) , sur le Port\_1, faire

- Plug Device



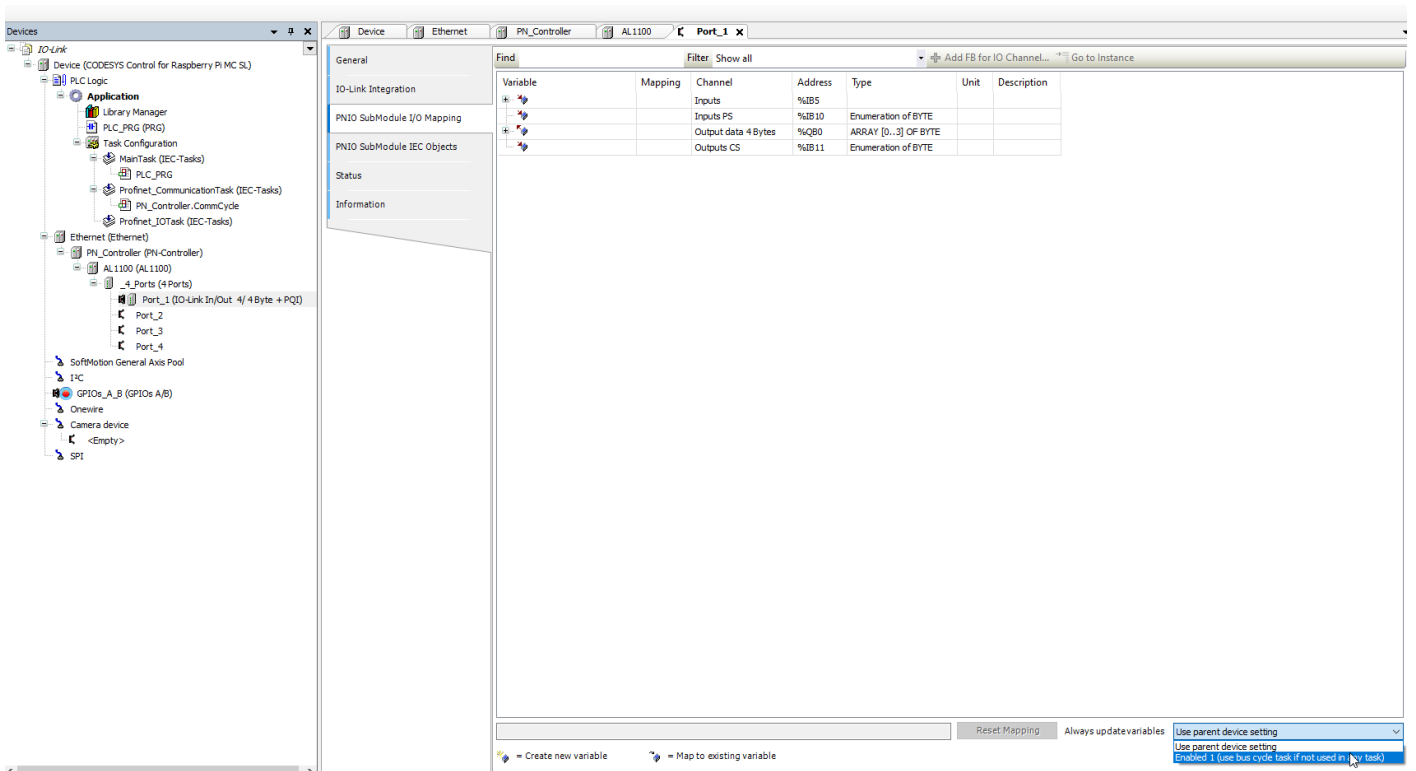
Comme nous avons branché le bouton capacitif KT6101 sur le port 1, il faut lui associer la taille de IO-Link Input + Output adapté au message. Dans notre cas, choisir :

- IO-Link In/Out 4/4 Byte + PQI
- et faire Plug Device



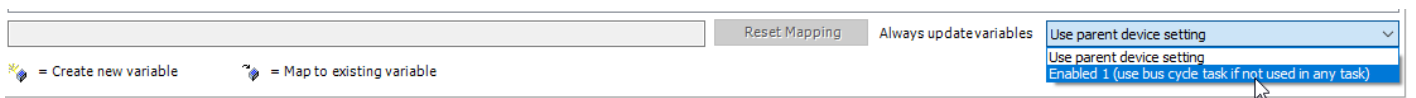
Avant d'aller plus loin, nous allons tout de suite configurer le rafraichissement automatique des données :

En double cliquant sur Port 1, en bas à droite, dans le champs Always Update Variables :



Faire :

- Enabled 1 (uses bus cycle task if not used in any task)

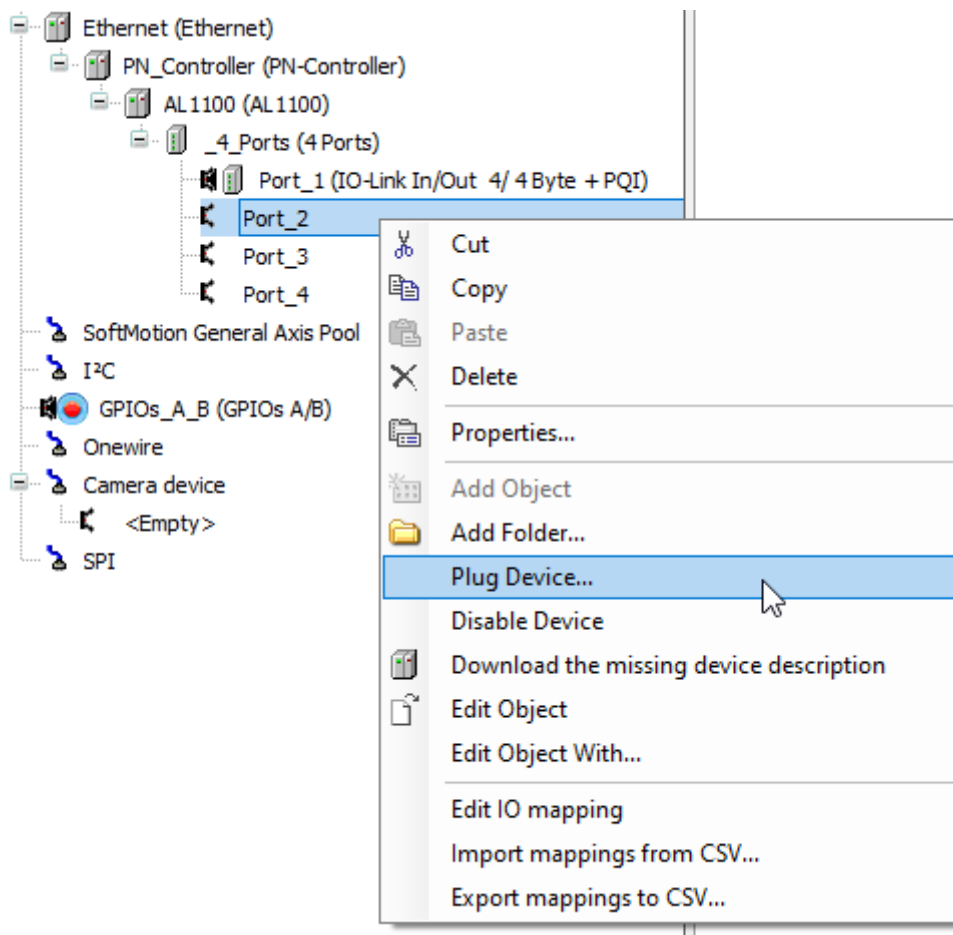


Vérifier que cela soit bien pris en compte sinon les valeurs des capteurs n'apparaîtrons pas !



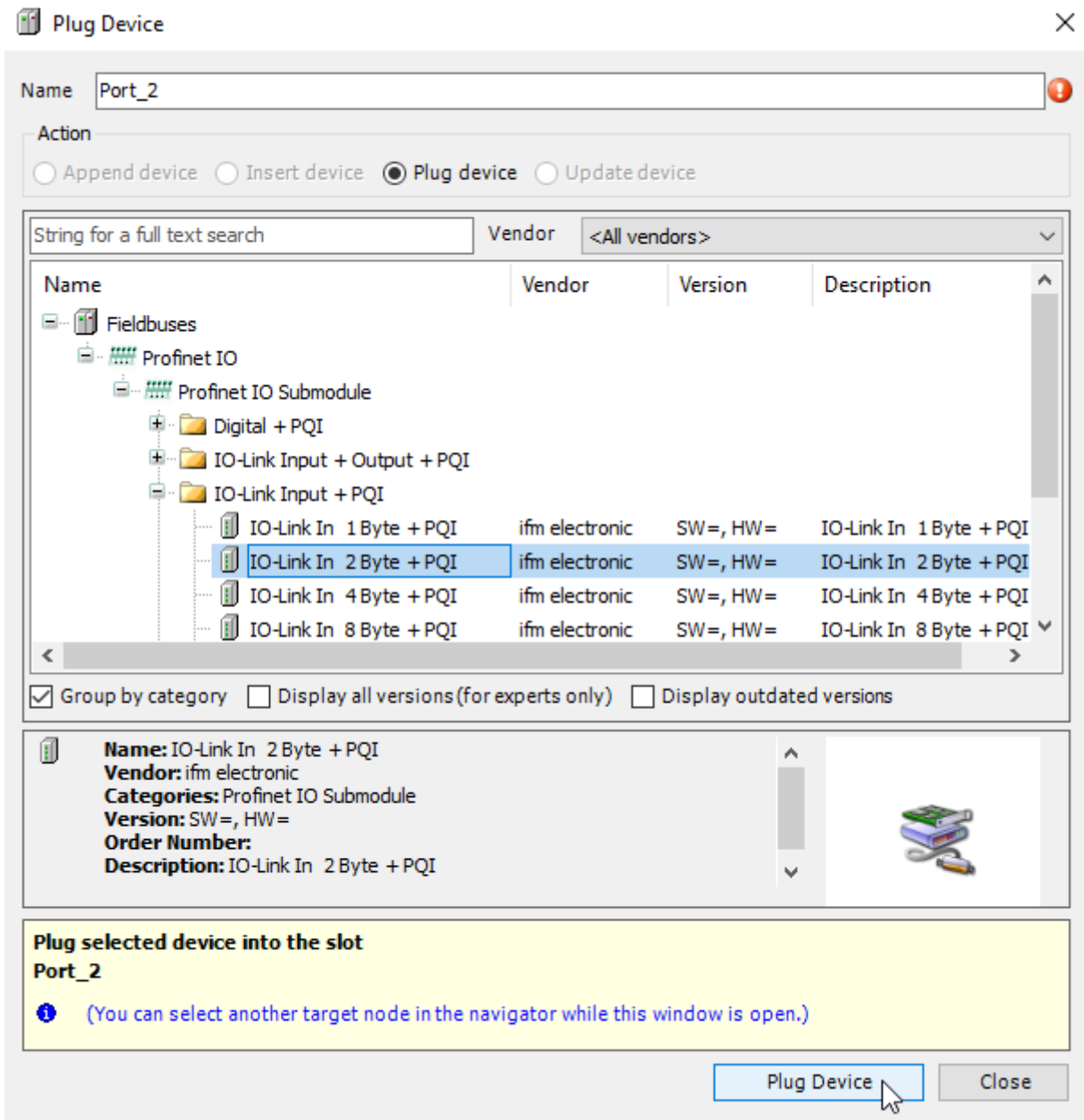
On va procéder de manière analogue pour Port\_2 :

- on clic sur Plug Device



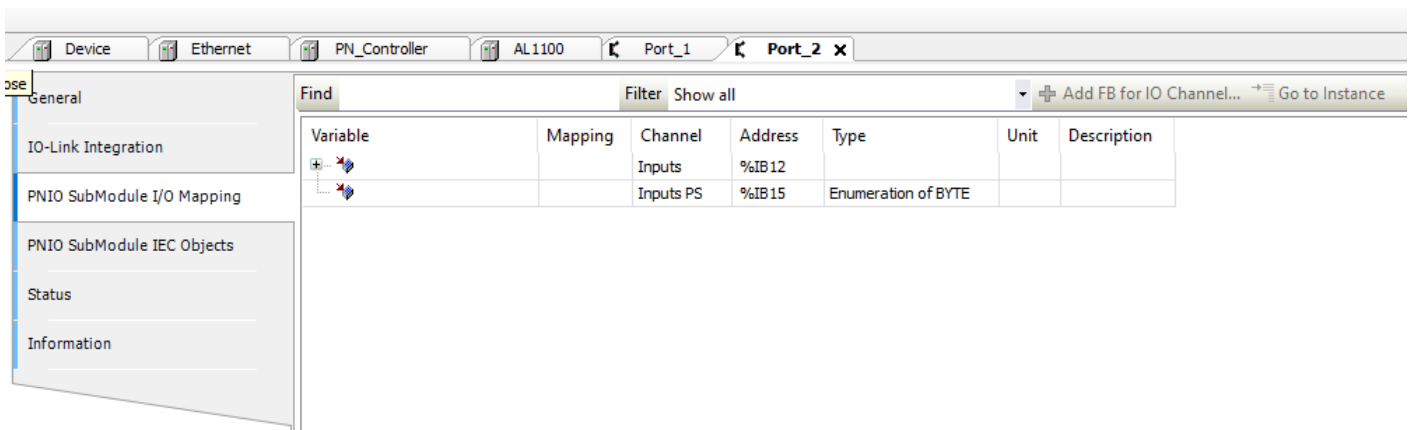
Il s'agit du détecteur de distance O5D150 qui est branché sur Port\_2, celui ci nécessite 2 Bytes d'Input pour le message :

- IO-Link 2 Bytes + PQI
- et Plug Device



Pareil que précédemment, on fait attention à autoriser Always update variables avec l'option :

- Enabled 1 (use bus cycle task if not used in any task)



Reset Mapping
Always update variables
Enabled 1 (use bus cycle task if not used in any task) ▾

🔍 = Create new variable
🔗 = Map to existing variable

Pour les Port\_3 et Port\_4, comme aucun capteur n'est branché dessus, nous placerons Disabled

Plug Device
✕

Name  !

Action
☐ Append device
☐ Insert device
☒ Plug device
☐ Update device

String for a full text search 
Vendor <All vendors> ▾

Name	Vendor	Version	Description
Fieldbuses			
Profinet IO			
Profinet IO Submodule			
Digital + PQI			
IO-Link Input + Output + PQI			
IO-Link Input + PQI			
IO-Link Output + PQI			
Disabled	ifm electronic	SW =, HW =	Disabled

☒ Group by category
☐ Display all versions (for experts only)
☐ Display outdated versions

**Name:** Disabled  
**Vendor:** ifm electronic  
**Categories:** Profinet IO Submodule  
**Version:** SW =, HW =  
**Order Number:**  
**Description:** Disabled

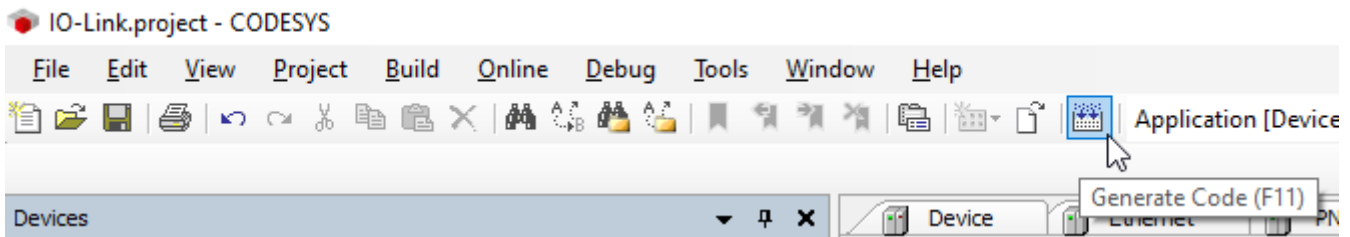
**Plug selected device into the slot**  
Port\_3  
! (You can select another target node in the navigator while this window is open.)

Plug Device Close

# Test rapide du fonctionnement du Master IO-Link

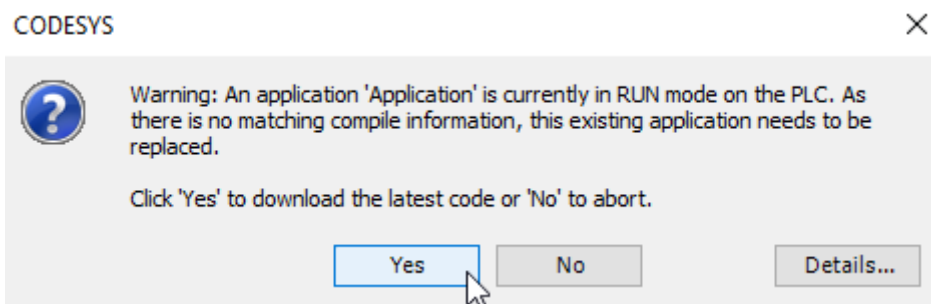
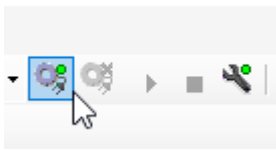
Sans oublier de faire Save, cliquer sur

- Generate Code (F11)



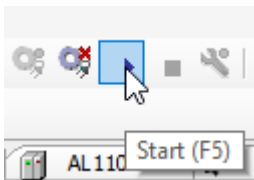
Pour se connecter sur le Raspberry Pi et y transférer le code Automate, cliquer sur la petite prise de courant (Login)

- Codesys peut indiquer ce message en indiquant qu'une application est déjà en Run sur le Raspberry Pi, confirmer avec Yes le transfert.

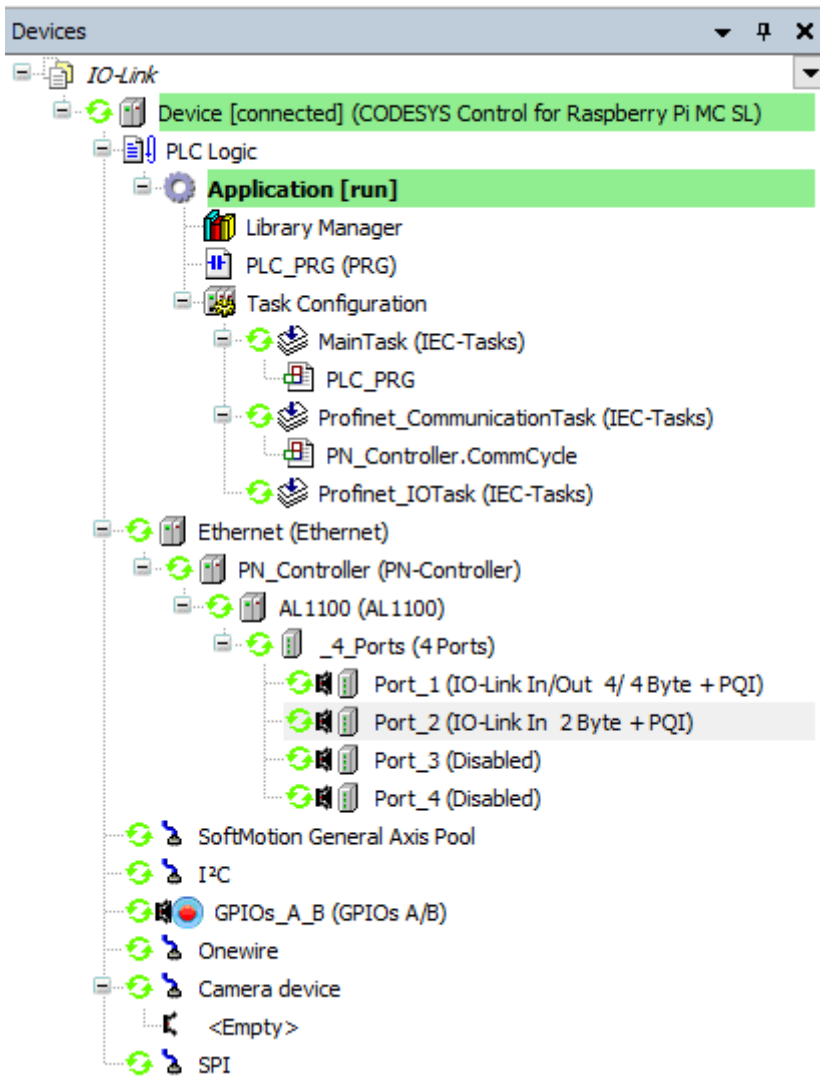


On clique sur :

- Start (F5)



Normalement, tous les éléments passent au vert.



En cliquant sur le Port\_2, dans

- PN/IO SubModule I/O Mapping, en développant l'arborescence, on peut voir les données de distance dans :
  - Input data 2 Bytes[0] %IB12
  - Input data 2 Bytes[0] %IB13

Ces données varient avec la distance mesurée. Comme pour TwinCAT, il faudra mapper ces données aux variables du programme Automate.

Variable	Mapping	Channel	Address	Type	Current Value
		Inputs	%IB12	ARRAY [0..1] OF BYTE	Only subelements up...
		Input data 2 Bytes	%IB12	BYTE	1
		Input data 2 Bytes[0]	%IB12	BYTE	97
		Input data 2 Bytes[1]	%IB13	BYTE	160
		Port Status	%IB14	USINT	GOOD
		Inputs PS	%IB15	Enumeration of BYTE	

En cliquant sur le Port\_1, dans

- PN/IO SubModule I/O Mapping, en développant l'arborescence, on peut voir les données de distance dans :
  - Input datat 4 Bytes[0] %IB5
  - Input datat 4 Bytes[1] %IB6
  - ... Ces données varient avec l'appui sur le bouton capacitif. Pareillement, nous devrons mapper ces variables capteur au programme automate.

Variable	Mapping	Channel	Address	Type	Current Value
Inputs			%IB5		Only subelements up...
Input data 4 Bytes			%IB5	ARRAY [0..3] OF BYTE	Only subelements up...
Input data 4 Bytes[0]			%IB5	BYTE	11
Input data 4 Bytes[1]			%IB6	BYTE	184
Input data 4 Bytes[2]			%IB7	BYTE	0
Input data 4 Bytes[3]			%IB8	BYTE	1
Port Status			%IB9	USINT	160
Inputs PS			%IB10	Enumeration of BYTE	GOOD
Output data 4 Bytes			%QB0	ARRAY [0..3] OF BYTE	Only subelements up...
Output data 4 Bytes[0]			%QB0	BYTE	0
Output data 4 Bytes[1]			%QB1	BYTE	0
Output data 4 Bytes[2]			%QB2	BYTE	0
Output data 4 Bytes[3]			%QB3	BYTE	0
Outputs CS			%IB11	Enumeration of BYTE	GOOD

Pour se déconnecter, on clique sur Stop pour Logout.

## Bilan

Cette première étape nous a permis de valider la bonne communication avec le Master IO-Link. Les prochaines étapes seront:

- création des fonctions de décodage pour les capteurs
- mapping des variables automate avec les variables IO-Link
- création d'une IHM permettant d'afficher ces résultats
- test de l'IHM sur une tablette

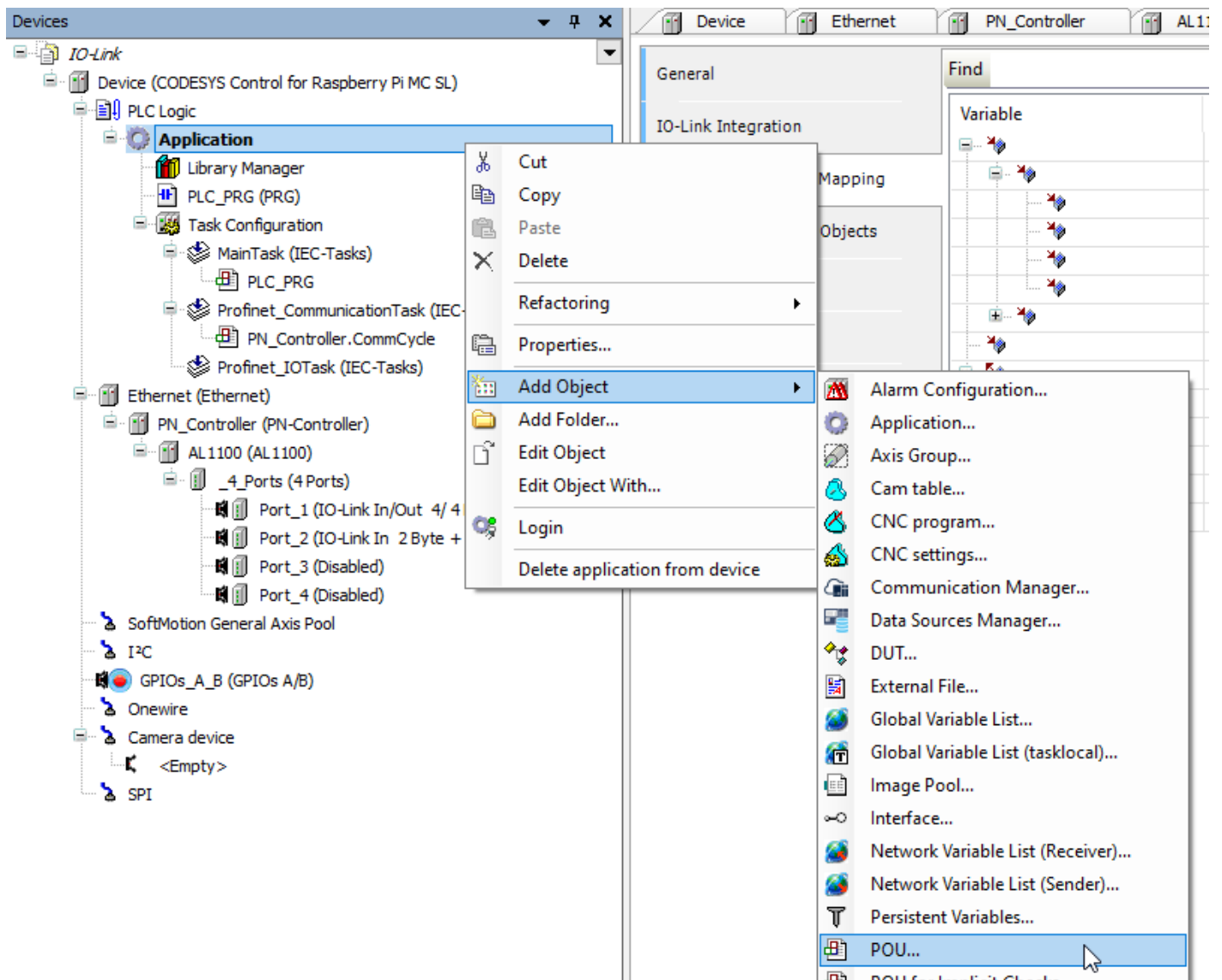
# Programme Automate et fonctions capteurs

Les différentes étapes présentées ci-dessous seront très similaires à celles présentées pour l'association d'un Master IO-Link à TwinCAT.

## Fonction capteur O5D150

Dans Application faire :

- Add Object -> POU...



Dans Add POU :

- Name : FB\_O5D150
- Type : Function block
- Implementation Language : ST



Create a new POU (Program Organization Unit)

Name  
FB\_O5D150

Type

☐ Program

☒ **Function block**

☐ Extends  ...

☐ Implements  ...

☐ Final ☐ Abstract

Access specifier  
 ▾

Method implementation language  
Continuous Function Chart (CFC) ▾

☐ **Function**

Return type  ...

Implementation language  
Structured Text (ST) ▾

Add Cancel

Le programme de la fonction FB\_O5D150 est identique à celui présenté dans TwinCAT :

The screenshot shows the CODESYS IDE interface. On the left, the 'Devices' tree is expanded to 'IO-Link', showing the hierarchy: Device (CODESYS Control for Raspberry Pi MC SL) > PLC Logic > Application > Library Manager > FB\_O5D150 (FB). The main editor displays the ladder logic for the 'FUNCTION\_BLOCK FB\_O5D150'. The code is as follows:

```

1 FUNCTION_BLOCK FB_O5D150
2   VAR_INPUT
3   END_VAR
4   VAR_OUTPUT
5       nCurrentDistance : INT;
6       bSwitchState     : BOOL;
7   END_VAR
8   VAR
9       aInputBytes      : ARRAY[0..1] OF BYTE;
10      nWord0            : WORD;
11      nWord_temp0       : WORD;
12      nWord_temp1       : WORD;
13   END_VAR
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

De même pour l'instanciation et l'appel de fonction, seule variante, le PLC\_PRG est dans notre cas écrit en Ladder, ce qui signifie qu'il est nécessaire d'ajouter un bloc à nommer fbO5D150 pour réaliser l'appel de fonction.

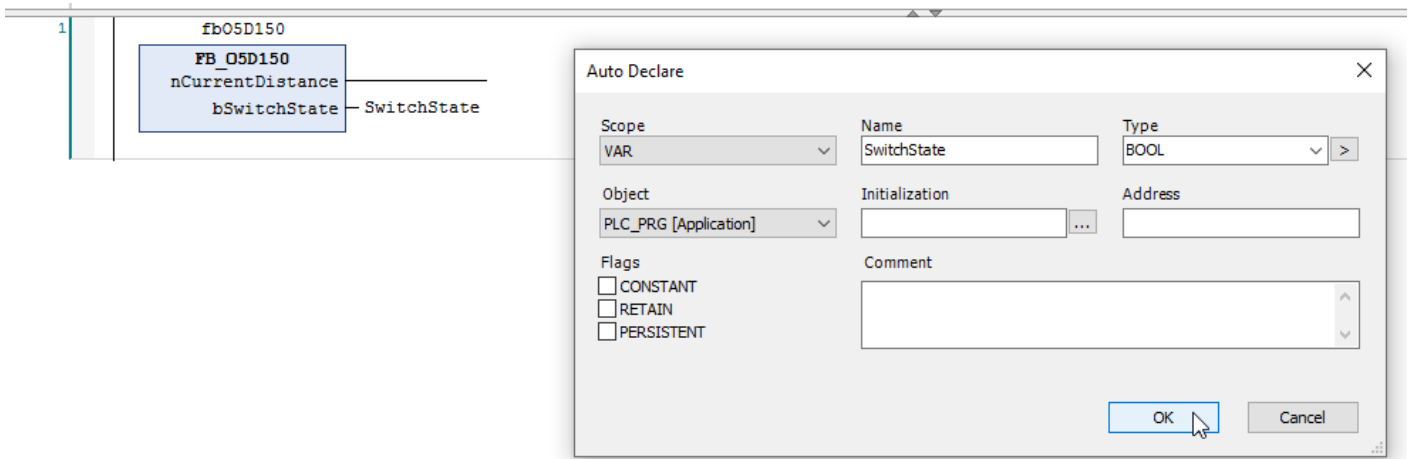
The screenshot shows the CODESYS IDE interface. On the left, the 'Devices' tree is expanded to 'IO-Link', showing the hierarchy: Device (CODESYS Control for Raspberry Pi MC SL) > PLC Logic > Application > Library Manager > PLC\_PRG (PRG). The main editor displays the ladder logic for the 'PROGRAM PLC\_PRG'. The code is as follows:

```

1 PROGRAM PLC_PRG
2   VAR
3       fbO5D150 : FB_O5D150;
4   END_VAR
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

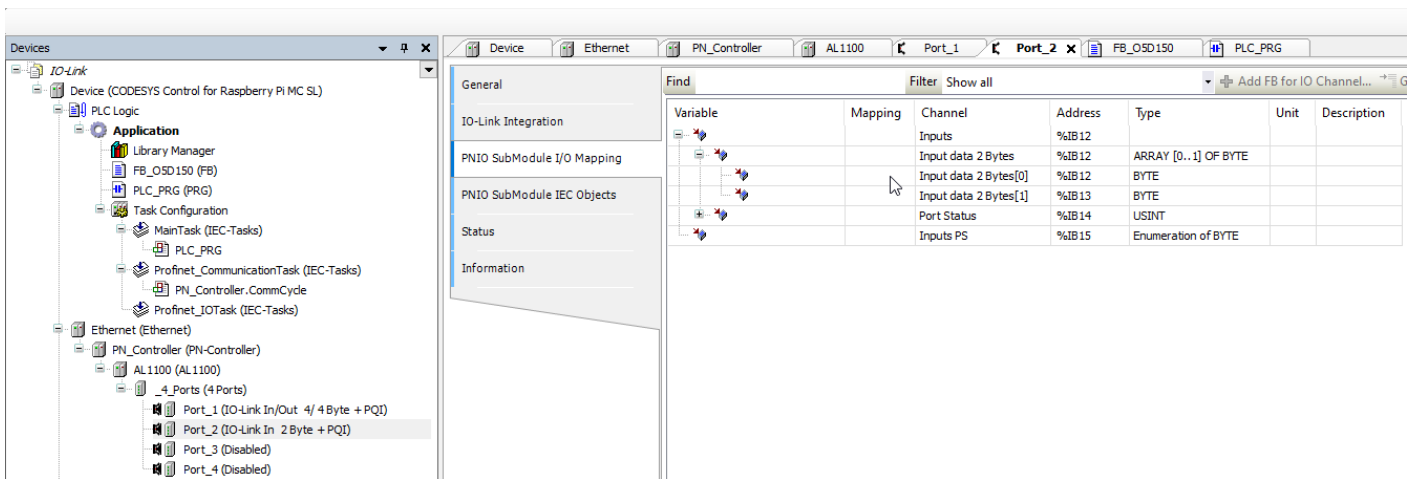
```

On peut ajouter une variable SwitchState de type dans le PLC\_PRG simplement en écrivant SwitchState en sortie du bloc.



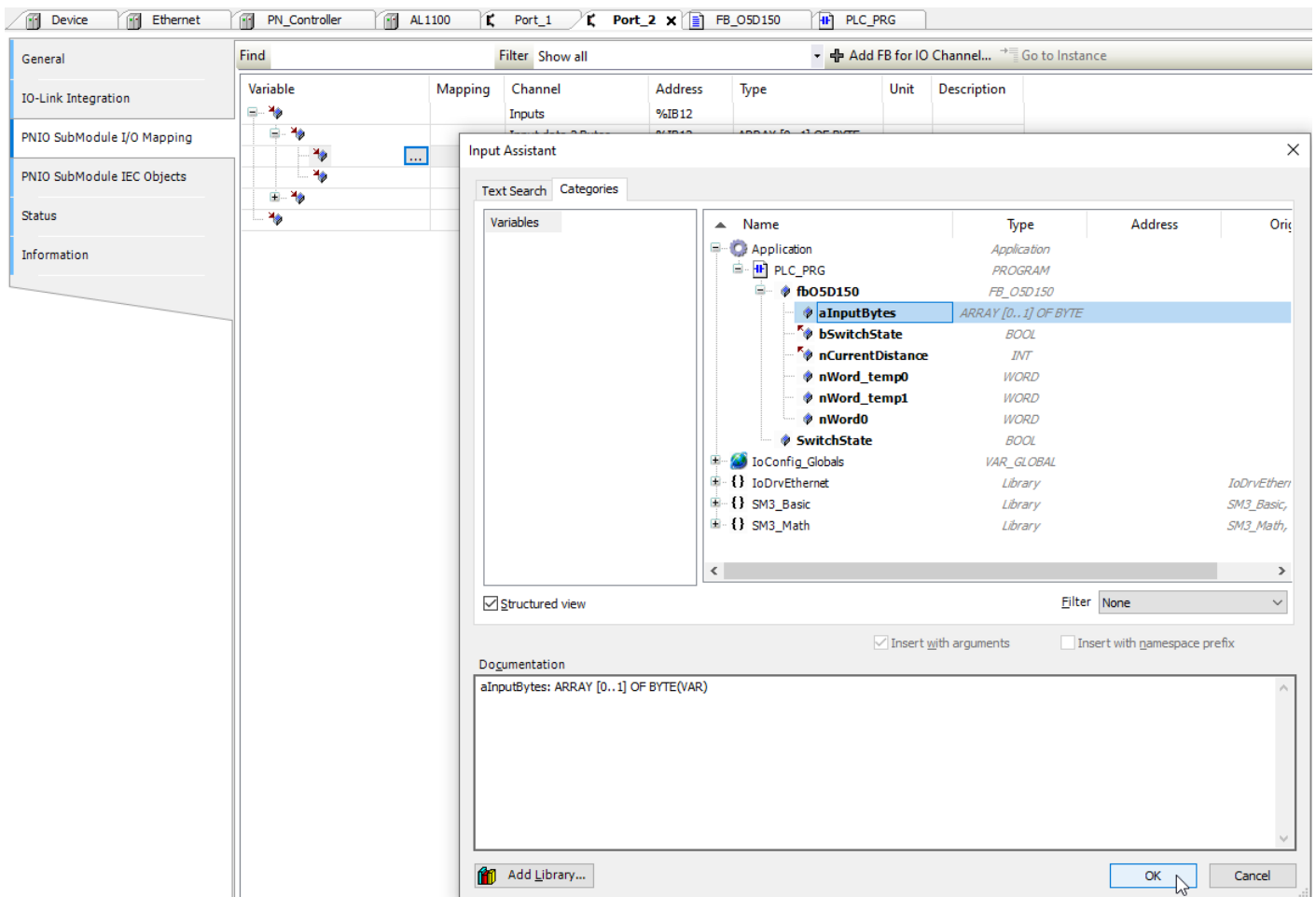
Nous allons mapper les variables de la fonction O5D150 aux variables IO-Link :

- dans Port\_2 -> PN I/O SubModule Mapping
- cliquer à côté de la petite boîte bleue dans variable au niveau de la ligne %IB12 BYTE

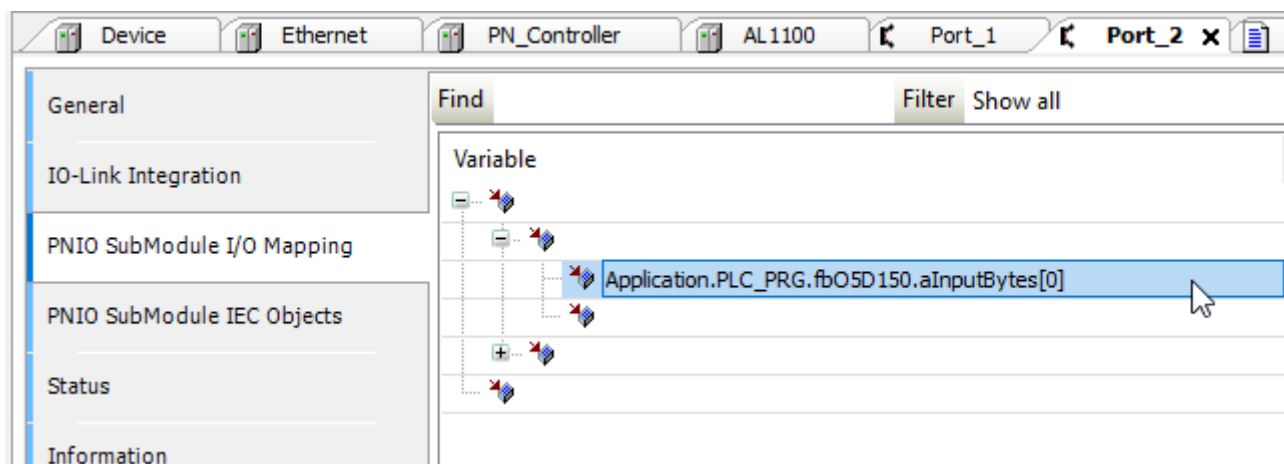


Trois petits points doivent apparaître et lancer l'Input Assistant:

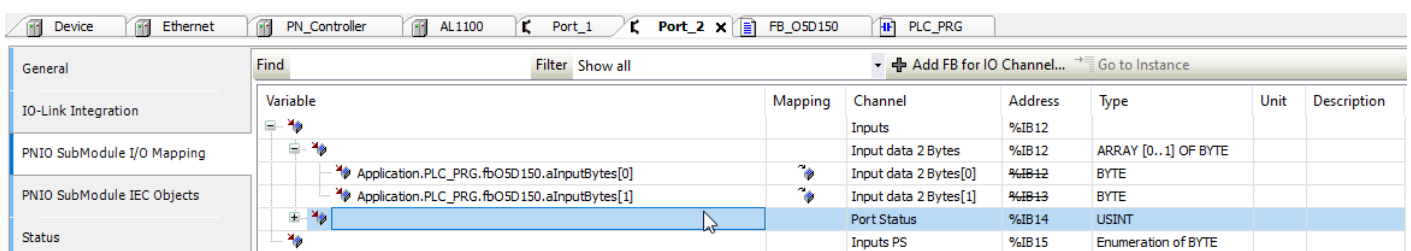
- choisir dan PLC\_PRG -> fbO5D150 -> alnputBytes



Placer à la fin de la variable [0] pour indiquer qu'il s'agit de l'élément 0 de l'array of Byte.

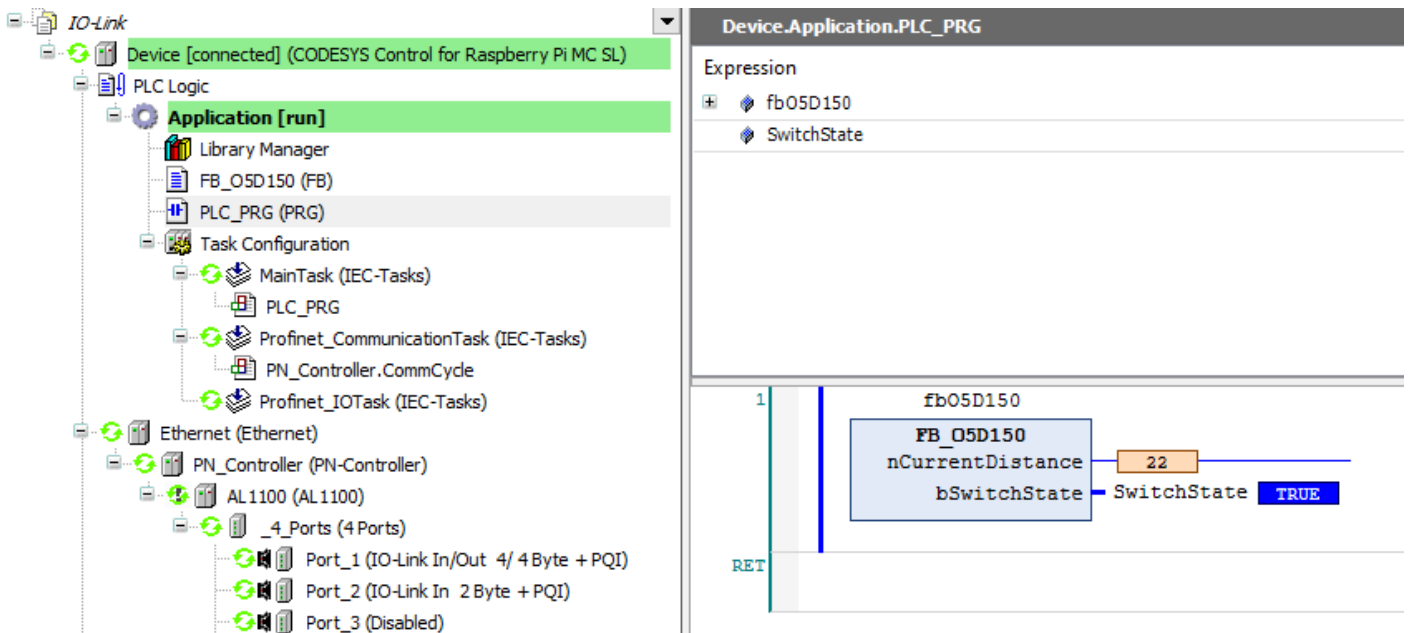


On fait de même pour la ligne %IB13. N'oubliez pas de placer à la fin de la variable [1] pour indiquer qu'il s'agit de l'élément 1 de l'array of Byte.



On effectue un Save, suivi d'un Generate, Login et Start

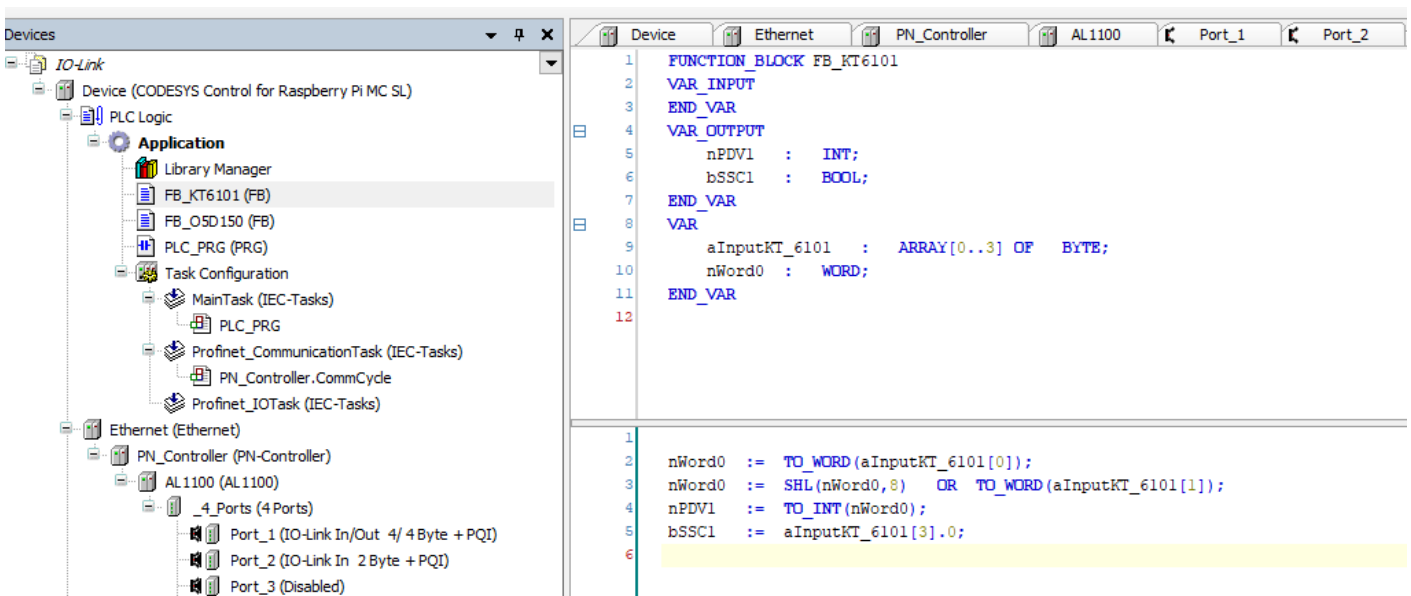
- et normalement le PLC\_PRG indique la valeur de distance mesurée par le capteur O5D150.



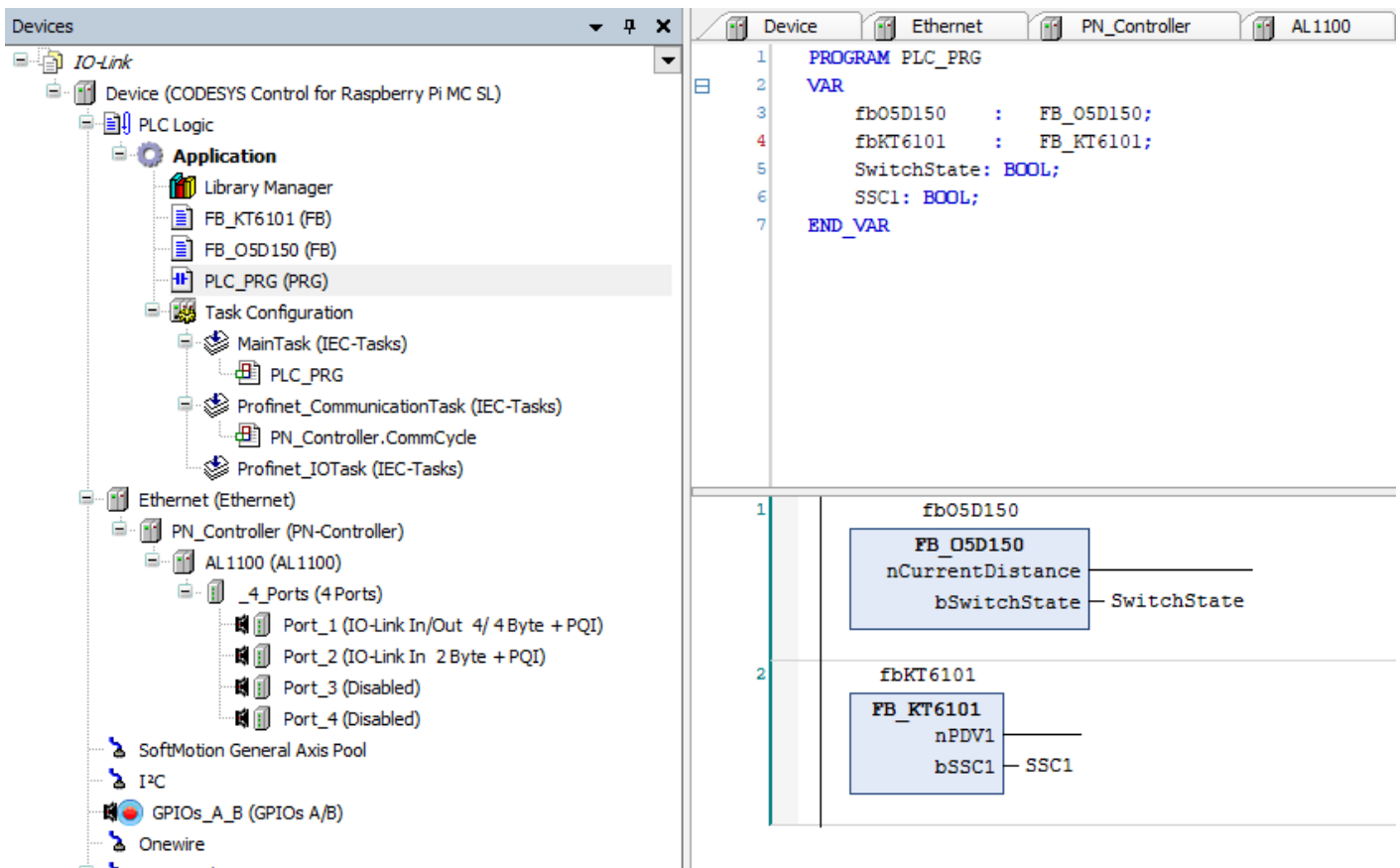
On n'oublie pas de faire Stop et Logout pour faire des modification du programme.

## Fonction bouton capacitif KT6101

Comme pour le capteur O5D150, le code pour le bouton capacitif KT6101 est identique à celui développé pour TwinCAT.



On réalise l'instanciation et l'appel de la fonction fbKT6101 dans PLC\_PRG



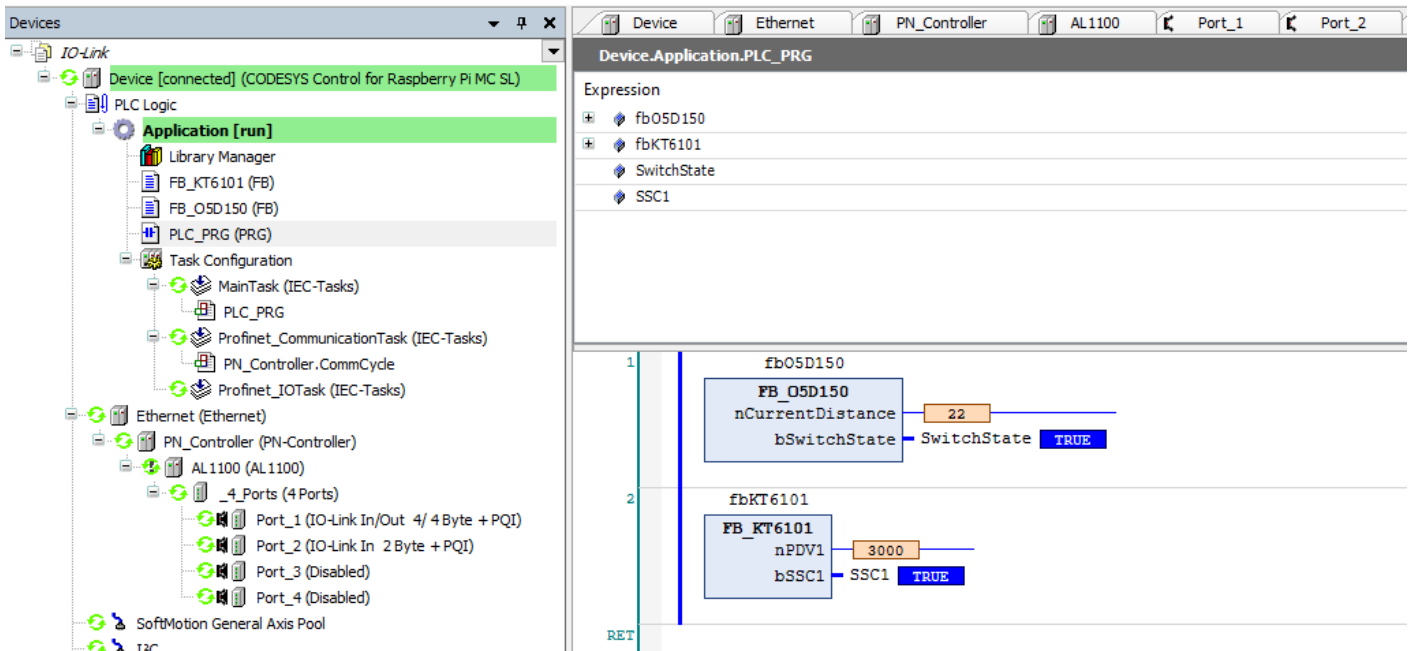
Pour le mapping des variables IO-Link, on procède de la même manière que le capteur O5D150

The screenshot shows the 'I/O Mapping' window for 'Port\_1'. The 'Find' tab is active, showing a list of variables and their mappings. The 'Filter' is set to 'Show all'. The 'Add FB for IO Channel...' button is visible. The 'Go to Instance' button is also visible.

Variable	Mapping	Channel	Address	Type
Application.PLC_PRG.fbKT6101.aInputKT_6101[0]		Inputs	%IB5	ARRAY [0..3] OF BYTE
Application.PLC_PRG.fbKT6101.aInputKT_6101[1]		Input data 4 Bytes[0]	%IB5	BYTE
Application.PLC_PRG.fbKT6101.aInputKT_6101[2]		Input data 4 Bytes[1]	%IB6	BYTE
Application.PLC_PRG.fbKT6101.aInputKT_6101[3]		Input data 4 Bytes[2]	%IB7	BYTE
		Input data 4 Bytes[3]	%IB8	BYTE
		Port Status	%IB9	USINT
		Inputs PS	%IB 10	Enumeration of BYTE
		Output data 4 Bytes	%QB0	ARRAY [0..3] OF BYTE
		Output data 4 Bytes[0]	%QB0	BYTE
		Output data 4 Bytes[1]	%QB1	BYTE
		Output data 4 Bytes[2]	%QB2	BYTE
		Output data 4 Bytes[3]	%QB3	BYTE
		Outputs CS	%IB 11	Enumeration of BYTE

On fait Generate, Login et Start :

- Le capteur de distance est toujours fonctionnel
- Le bouton capacitif réagit en fonction de l'appui

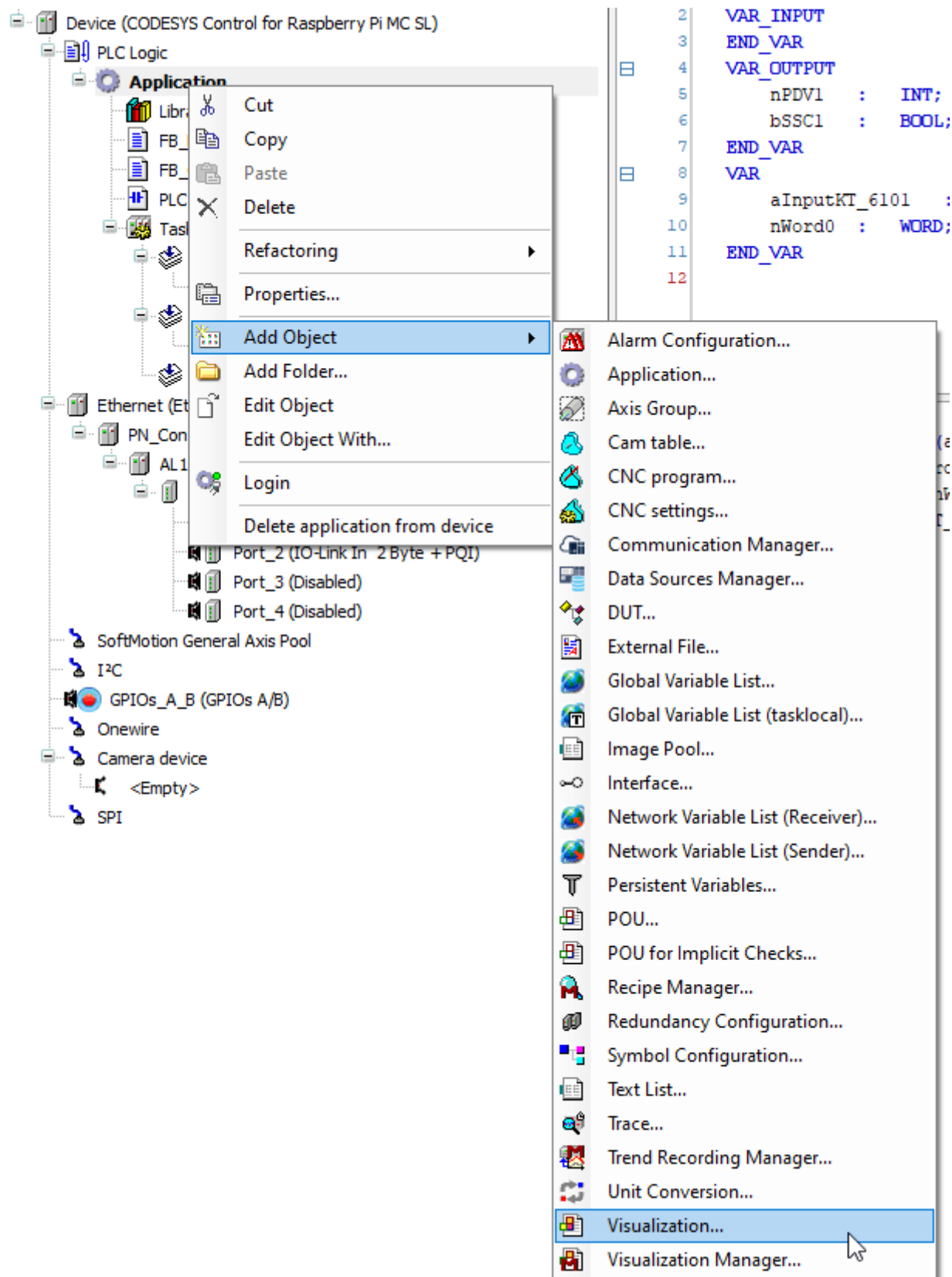


Pour finaliser cette démonstration, nous allons intégrer une IHM et réaliser une WebVisu sur tablette.

# Visualisation IHM et WebVisu

Pour intégrer une visualisation :

- Clic droit sur Application -> Add Object -> Visualization...
- Dans la fenêtre Add Visualization, laisser le Name : Visualization et faire Add



Pour simuler les deux couleurs lumineuses Bleue et Verte, j'utilise une astuce sur la visualisation :

- Je crée un voyant Bleu que j'associe à la variable PLC\_PRG.fbKT6101.bSSC1 que j'inverse avec NOT().
- ce voyant passera au premier plan avec la même variable dans Bring to foreground
- -> quand le voyant bleu est allumé, il passe au premier plan



Property	Value
Element name	GenElemInst_1
Type of element	Lamp
Position	
X	519
Y	133
Width	70
Height	70
Variable	NOT(PLC_PRG.fbKT6101.bSSC1)
Image settings	
Transparent	<input type="checkbox"/>
Transparent color	Black
Isotropic type	Isotropic
Horizontal align...	Left
Vertical alignment	Top
Texts	
Tooltip	
State variables	
Invisible	
Center	
X	554
Y	168
Absolute movement	
Movement	
X	
Y	
Rotation	
Scaling	
Interior rotation	
Animation duration	0
Bring to foreground	NOT(PLC_PRG.fbKT6101.bSSC1)
Background	
Image	Blue

Pour le voyant vert, je procède de la même manière sans inverser la variable PLC\_PRG.fbKT6101.bSSC1



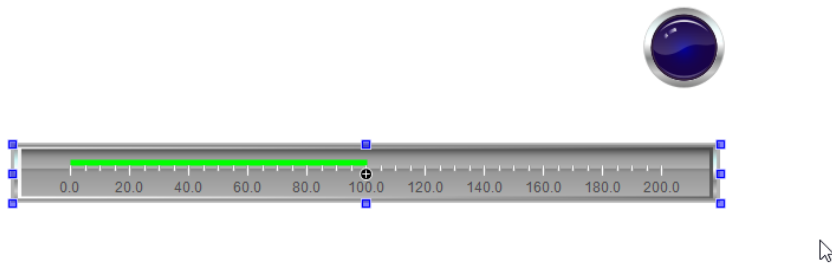
Element name	GenElemInst_2
Type of element	Lamp
Position	
X	614
Y	132
Width	70
Height	70
Variable	PLC_PRG.fbKT6101.bSSC1
Image settings	
Transparent	
Transparent color	Black
Isotropic type	Isotropic
Horizontal align...	Left
Vertical alignment	Top
Texts	
Tooltip	
State variables	
Invisible	
Center	
X	649
Y	167
Absolute movement	
Movement	
X	
Y	
Rotation	
Scaling	
Interior rotation	
Animation duration	0
Bring to foreground	PLC_PRG.fbKT6101.bSSC1
Background	
Image	Green

Je superpose les deux voyants sur l'IHM :

- si le voyant Bleu est actif, il passera au premier plan
- si le voyant Vert est actif, il passera au premier plan

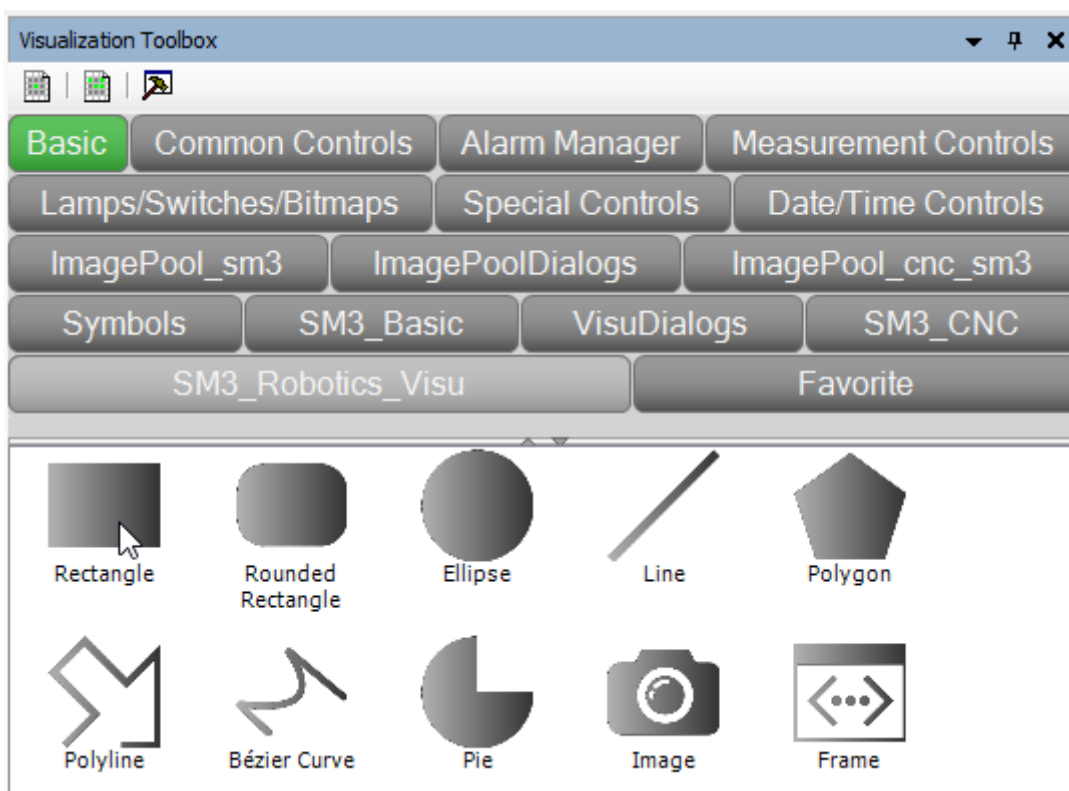
Une jauge de distance est également placée :

- la variable associée à la jauge est `PLC_PRG.fb05D150.nCurrentDistance`
- Scale End est configuré à 200 pour la distance max du capteur en cm
- Main Scale est placée à 20



Property	Value
Element name	GenElemInst_3
Type of element	Bar Display
Value	PLC_PRG.fbOSD150.nCurrentDistance
Center	
X	463
Y	274
Absolute movement	
Movement	
X	
Y	
Rotation	
Scaling	
Interior rotation	
Animation duration	0
Bring to foreground	
Position	
X	164
Y	249
Width	599
Height	50
Background	
Image color	Gray
Own image	
Bar	
Diagram type	Scale besides bar
Orientation	Horizontal
Running direction	Left to right
Optimum size f...	<input type="checkbox"/>
Scale	
Scale start	0
Scale end	200
Main scale	20
Subscale	5
Scale line width	1
Scale color	<input type="text" value="Scale color barelement"/>
Scale in 3D	<input checked="" type="checkbox"/>
Element frame	<input type="checkbox"/>

On souhaite également afficher la distance dans un champ, nous placerons ainsi un rectangle qui servira de champ de visualisation :

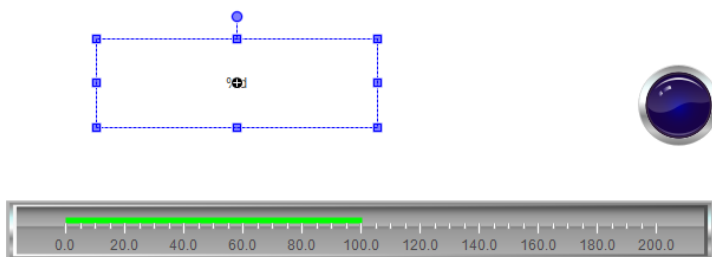


Dans le champ Text, nous mettrons :

- %d pour la valeur à afficher en Int

Dans le Champ Text variable, nous mettrons :

- PLC\_PRG. fb05D150. nCurrentDistance

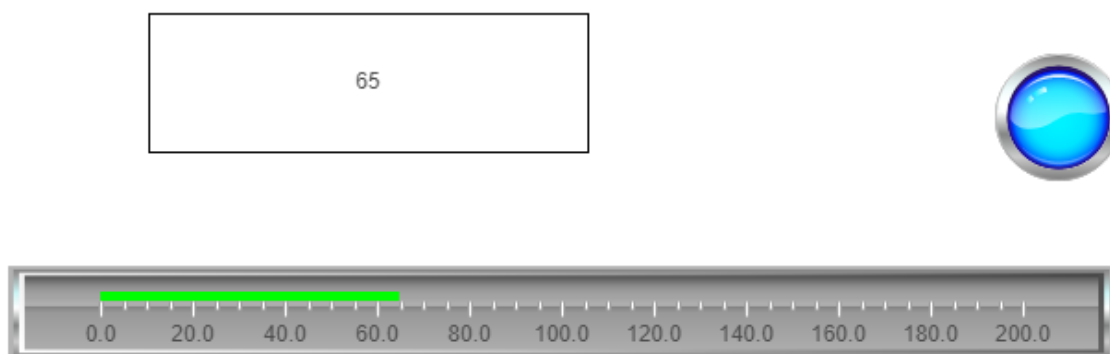


Property	Value
Element name	GenElemInst_4
Text ID	569
Type of element	Rectangle
Position	
X	239
Y	111
Width	238
Height	75
Angle	0
Center	
X	358
Y	148
Colors	
Use gradient color	<input type="checkbox"/>
Gradient setting	linear, Black, White
Appearance	
Texts	
Text	%d
Tooltip	
Text properties	
Absolute movement	
Movement	
X	
Y	
Rotation	
Scaling	
Interior rotation	
Use REAL values	<input type="checkbox"/>
Relative movement	
Text variables	
Text variable	PLC_PRG.fb05D150.nCurrentDistance
Tooltip variable	

Il reste à faire la connexion au Raspberry Pi et lancer le Runtime :

- Generate, Login, Start

Et l'IHM s'anime en fonction des mesures et appuis sur les capteurs.



# WebVisu

Il est possible de récupérer cette IHM sur une page Web à travers la WebVisu.

- On place l'adresse IP du Raspberry Pi dans le navigateur de la tablette suivi du numéro de port 8080 et l'on charge la page webvisu.htm
- 192.168.1.15:8080/webvisu.htm
- On peut visualiser la mesure de distance faite par le capteur ainsi que les appuis sur le bouton capacitif

---

Revision #2

Created 4 July 2023 09:28:01 by Philippe Celka

Updated 4 July 2023 09:51:21 by Philippe Celka