

Annexes : Programmation ST/LD/SFC sous OpenPLC

Culture Sciences
de l'Ingénieur

Jean-Philippe ILARY

Édité le
15/03/2021

école
normale
supérieure
paris-saclay

*Ces annexes sont issues d'une publication du numéro 103 de La Revue 3EI de janvier 2021, et accompagnent la ressource « Programmation ST/LD/SFC sous OpenPLC ».
Jean-Philippe Ilary est enseignant au département GEII de l'IUT Ville d'Avray.*

Contenu du document :

Annexe 1 : Syntaxe des adresses I/O TOR sous OpenPLC.....	2
Annexe 2 : Syntaxe des adresses O/I Analogiques sous OpenPLC.....	3
Annexe 3 : Syntaxe des adresses mémoires interne (Registres) sous OpenPLC.....	4
Annexe 3 : Temporisation (LADDER).....	5
Annexe 3 : Temporisation (SFC).....	6
Annexe 4 : Compteurs (LADDER).....	7
Annexe 4' : Compteurs (SFC).....	8

ANNEXE 1

Syntaxe des adresses I/O TOR sous OpenPLC

%IX & %QX

<https://www.openplcproject.com/reference/plc-addressing/>

Entrées TOR :

L'adressage des entrées TOR se fait de façon classique. La plage va de %IX0.0 à %IX99.7 (soit 100×8 entrées TOR possibles) :

Description de la syntaxe :

- %IX0.1 : Input
- %IX0.1 : « Carte » n°0, il y a 100 « cartes » possible à 8 entrées
- %IX0.1 : Entrée n°1 de la carte n°0, il y a 8 entrées/carte

Sorties TOR :

L'adressage des sorties TOR se fait aussi de façon classique. La plage va de %QX0.0 à %QX99.7 (soit 100×8 sorties TOR possibles) :

Description de la syntaxe :

- %QX0.2 : Sortie
- %QX0.2 : « Carte » n°0, il y a 100 « cartes » possible à 8 sorties
- %QX0.2 : Sortie n°2 de la carte n°0, il y a 8 sorties/carte

#	Nom	Classe	Type	Adresse
1	Entree_TOR	Locale	BOOL	%IX0.1
2	Sortie_TOR	Locale	BOOL	%QX0.2

Extrait du lien ci-dessus :

<data-size>	Common Name	Number of Bits	Elementary Data Types
X	Bit	1	BOOL
B	Byte	8	BYTE, , SINTUSINT
W	Word	16	WORD, , INTUINT
D	Double word	32	DWORD, , DINTUDINTFLOAT
L	Long word	64	LWORD, , LINTULINTDOUBLE

À titre de comparaison Sous l'environnement d'Unity Pro

Entrées TOR : (%IRack.Module.Voie.[rang])

L'adressage des entrées TOR se fait de façon classique :

Description de la syntaxe :

- %I0.3.1.0 : Input
- %I0.3.1.0 : Rack n°0,
- %I0.3.1.0 : Carte n°3,
- %I0.3.1.0 : Bit n°1 de la carte n°3,
- %I0.3.1.0 : Rang =0 pas de propriété pour cette entrée.

Sorties TOR : (%QRack.Module.Voie.[rang])

L'adressage des sorties TOR se fait aussi de façon classique :

Description de la syntaxe :

- %Q0.2.1.0 : **Sortie**
- %Q0.2.1.0 : Rack n°0,
- %Q0.2.1.0 : Carte n°2,
- %Q0.2.1.0 : Bit n°1 de la carte n°2,
- %Q0.2.1.0 : Rang =0 pas de propriété pour cette entrée.

ANNEXE 2

Syntaxe des adresses I/O Analogiques sous OpenPLC

%IW & %QW

Entrées ANALOGIQUES :

L'adressage des entrées ANALOGIQUE suit la représentation suivante et la plage va de %IW0 à %IW99 :

Description de la syntaxe :

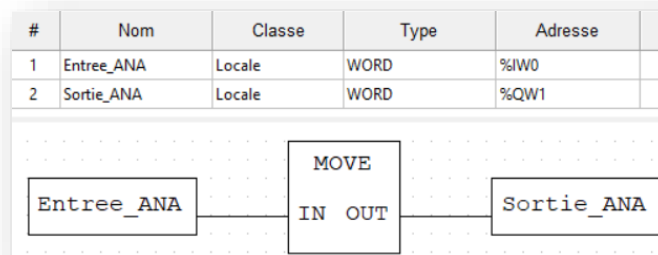
- %IW0 : Input,
- %IW0 : Mot de 16 bits (donc de 0 à 65535),
- %IW0 : Entrée analogique n°0.

Sorties ANALOGIQUES :

L'adressage des sorties ANALOGIQUE suit la représentation suivante et la plage va de %QW0 à %QW99 :

Description de la syntaxe :

- %QW1 : Sortie,
- %QW1 : Mot de 16 bits (donc de 0 à 65535),
- %QW1 : Sortie analogique n°1.



À titre de comparaison Sous l'environnement d'Unity Pro

Entrées ANA : (%IWRack.Module.Voie.[rang])

L'adressage des entrées ANA se fait de façon classique :

Description de la syntaxe :

- %IW0.3.1.0 : Input
- %IW0.3.1.0 : Rack n°0,
- %IW0.3.1.0 : Carte n°3,
- %IW0.3.1.0 : mot n°1 de la carte n°3,
- %IW0.3.1.0 : Rang =0 pas de propriété pour cette entrée (on peut s'en passer)

Sorties ANA : (%QWRack.Module.Voie.[rang])

L'adressage des sorties ANA se fait aussi de façon normalisée :

Description de la syntaxe :

- %QW0.2.1 : **Sortie**
- %QW0.2.1 : Rack n°0,
- %QW0.2.1 : Carte n°2,
- %QW0.2.1 : Voie n°1 de la carte n°2.

ANNEXE 3

Syntaxe des adresses mémoires interne (Registres) sous OpenPLC

%M

Registres sur 8 bits :

La plage va de %MB0 à %MB1023 :

Description de la syntaxe :

- **%MB1** : **M**émoire - **R**egistre
- **%MB1** : Byte de 8 bits pouvant être rangé dans ce registre
- **%MB1** : Registre n° 1

Registres sur 16 bits :

La plage va de %MW0 à %MW1023 :

Description de la syntaxe :

- **%MW1** : **M**émoire - **R**egistre
- **%MW1** : Mot de 16 bits pouvant être rangé dans ce registre
- **%MW1** : Registre n° 1

Registres sur 32 bits :

La plage va de %MD0 à %MD1023 :

Description de la syntaxe :

- **%MD2** : **M**émoire - **R**egistre
- **%MD2** : Double Mot de 32 bits pouvant être rangé dans ce registre
- **%MD2** : Registre n° 2

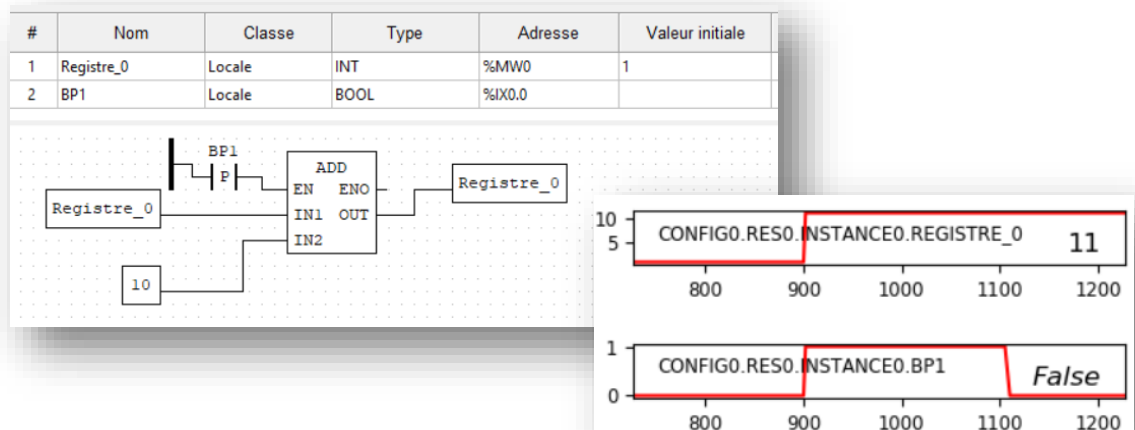
Registres sur 64 bits :

La plage va de %ML0 à %ML1023 :

Description de la syntaxe :

- **%ML3** : **M**émoire - **R**egistre
- **%ML3** : Long Mot de 64 bits pouvant être rangé dans ce registre
- **%ML3** : Registre n° 3

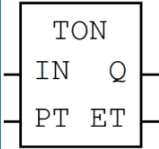
Registre_0=Registre_0+10 Sur front montant de BP1



ANNEXE 3

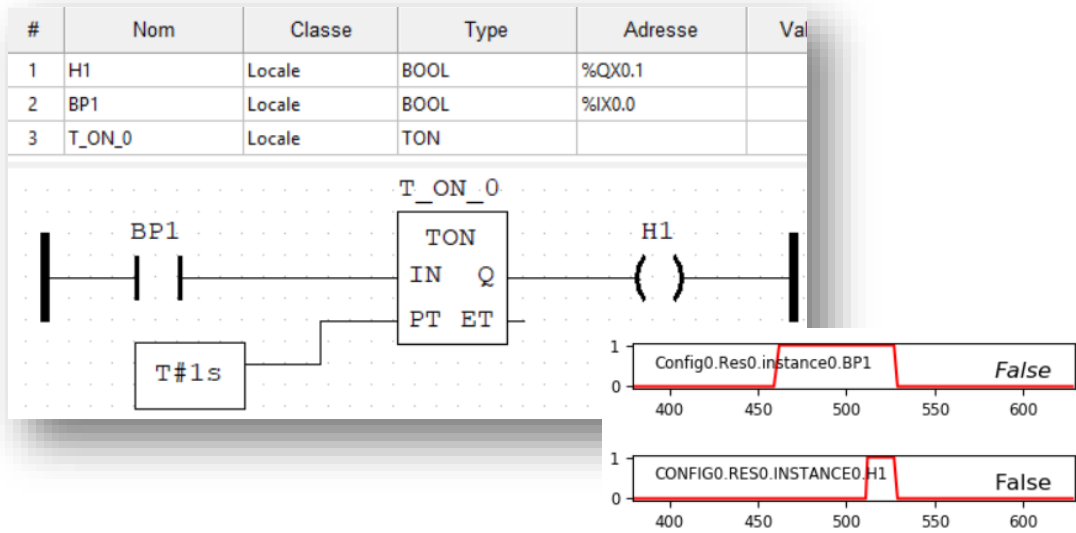
Temporisations (LADDER)

T_ON_0

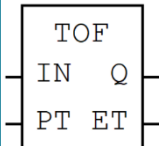


La temporisation à retard peut être utilisée pour retarder un front montant d'une durée donnée.

Dans l'exemple ci-dessous, la temporisation au travail est mise à 1s

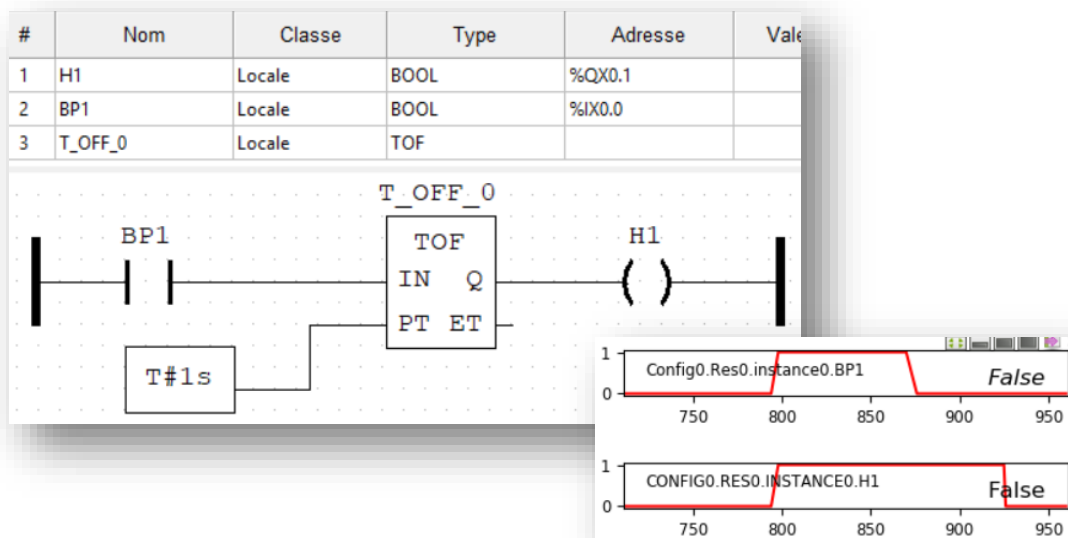


T_OFF_0



La temporisation de retard peut être utilisée pour retarder un front descendant d'une durée donnée.

Dans l'exemple ci-dessous, la temporisation est au repos est mise à 1s



ANNEXE 3

Temporisations (SFC)

Les temporisations en SFC sont possibles grâce aux Qualificatifs suivants :

Qualificatif

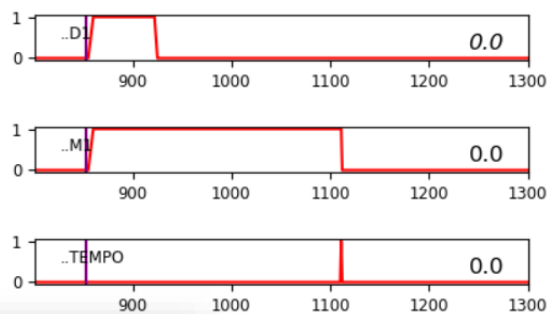
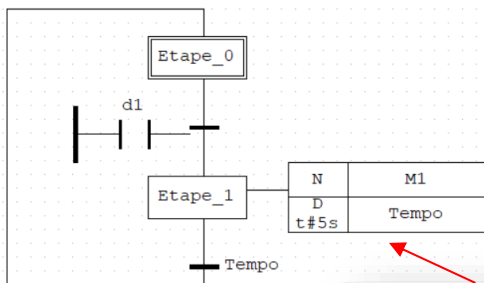
Pour associer des actions à des étapes CEI, vous disposez des qualificatifs (qualifier) suivant:

N	Non-stored (non mémorisé)	l'action demeure active aussi longtemps que l'étape demeure active
R	overriding Reset (remise à zéro prioritaire)	désactivation de l'action
S	Set (Stored) (positionné (mémorisé))	activation de l'action, qui demeure ensuite active jusqu'au prochain reset
L	time Limited (limite dans le temps)	activation de l'action pendant une durée déterminée
D	time Delayed (temporisé)	activation de l'action après un certain temps, pour autant que l'étape demeure active
P	Pulse (impulsion)	l'action est exécutée exactement une fois, lorsque l'étape est activée
SD	Stored and time Delayed (mémorisé et temporisé)	activation de l'action après un certain temps; l'action demeure ensuite active jusqu'au prochain reset
DS	Delayed and Stored (temporisé et mémorisé)	activation de l'action après un certain temps, pour autant que l'étape demeure active; l'action demeure alors active jusqu'au prochain reset
SL	Stored and time Limited (mémorisé et limité dans le temps)	activation de l'action pendant une durée déterminée

Les qualificatifs L, D, SD, DS et SL doivent être accompagnés d'une spécification de temps sous format de constante TIME, p.ex. L T#5s.

D : Time Delayed

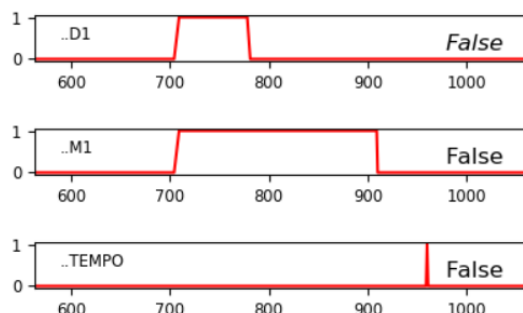
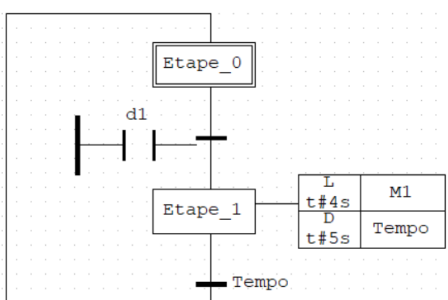
La variable Tempo passe à 1 après 5s.



Qualificatif	Durée	Type	Valeur
N		Variable	M1
D	t#5s	Variable	Tempo

L : Time limited

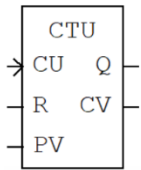
L'action M1 est réalisé pendant 4s dès que l'étape_1 est active. L'étape_1 sera restée active 5s grâce à tempo.



ANNEXE 4

Compteurs (LADDER)

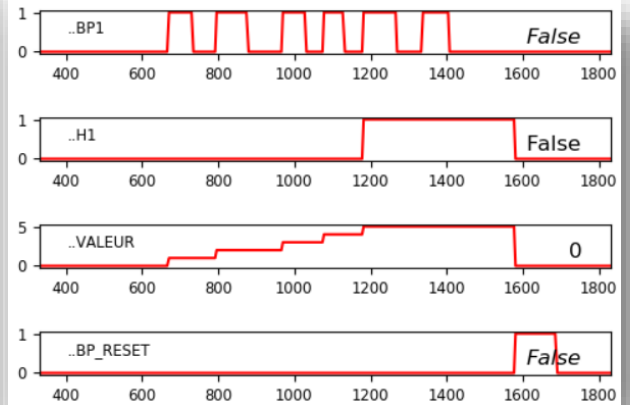
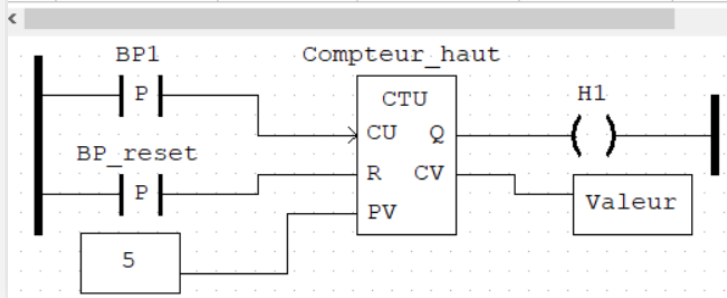
Compteur_Haut



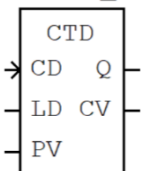
Le compteur peut être utilisé pour signaler qu'un compte a atteint une valeur maximale.

Dans l'exemple ci-dessous, le compte (PV) est mis à 5 et chaque front montant de BP1 incrémente de 1 le compteur. La sortie Q passe à '1' lorsque PV est atteint. L'appui sur le BP_reset remet tout à 0.

#	Nom	Classe	Type	Adresse	Val
1	BP_reset	Locale	BOOL	%IX0.1	
2	BP1	Locale	BOOL	%IX0.0	
3	H1	Locale	BOOL	%QX0.1	
4	Compteur_haut	Locale	CTU		
5	Valeur	Locale	INT	%MW0	



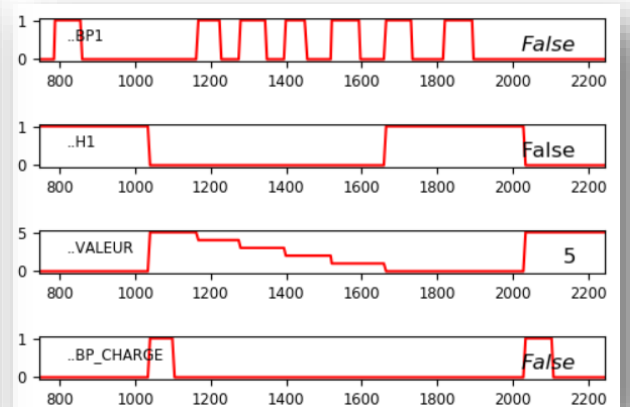
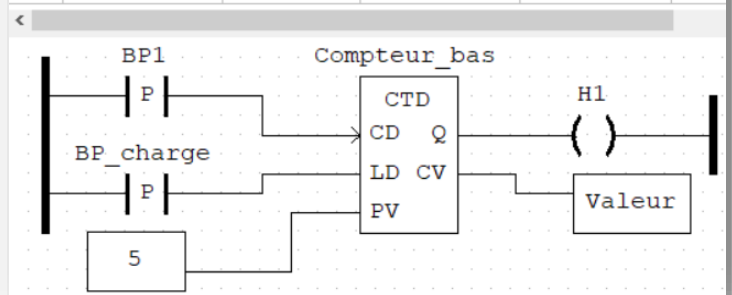
Compteur_bas



Le décompteur peut être utilisé pour signaler qu'un compte a atteint zéro, lors du décompte d'une valeur prédéfinie.

Lorsque LD passe à 1, le compteur est initialisé à la valeur présente sur PV (ici 5). Ensuite, à chaque impulsion sur CD, cette valeur est décrétement jusqu'à 0.

#	Nom	Classe	Type	Adresse	Val
1	BP_charge	Locale	BOOL	%IX0.1	
2	BP1	Locale	BOOL	%IX0.0	
3	H1	Locale	BOOL	%QX0.1	
4	Compteur_bas	Locale	CTD		
5	Valeur	Locale	INT	%MW0	



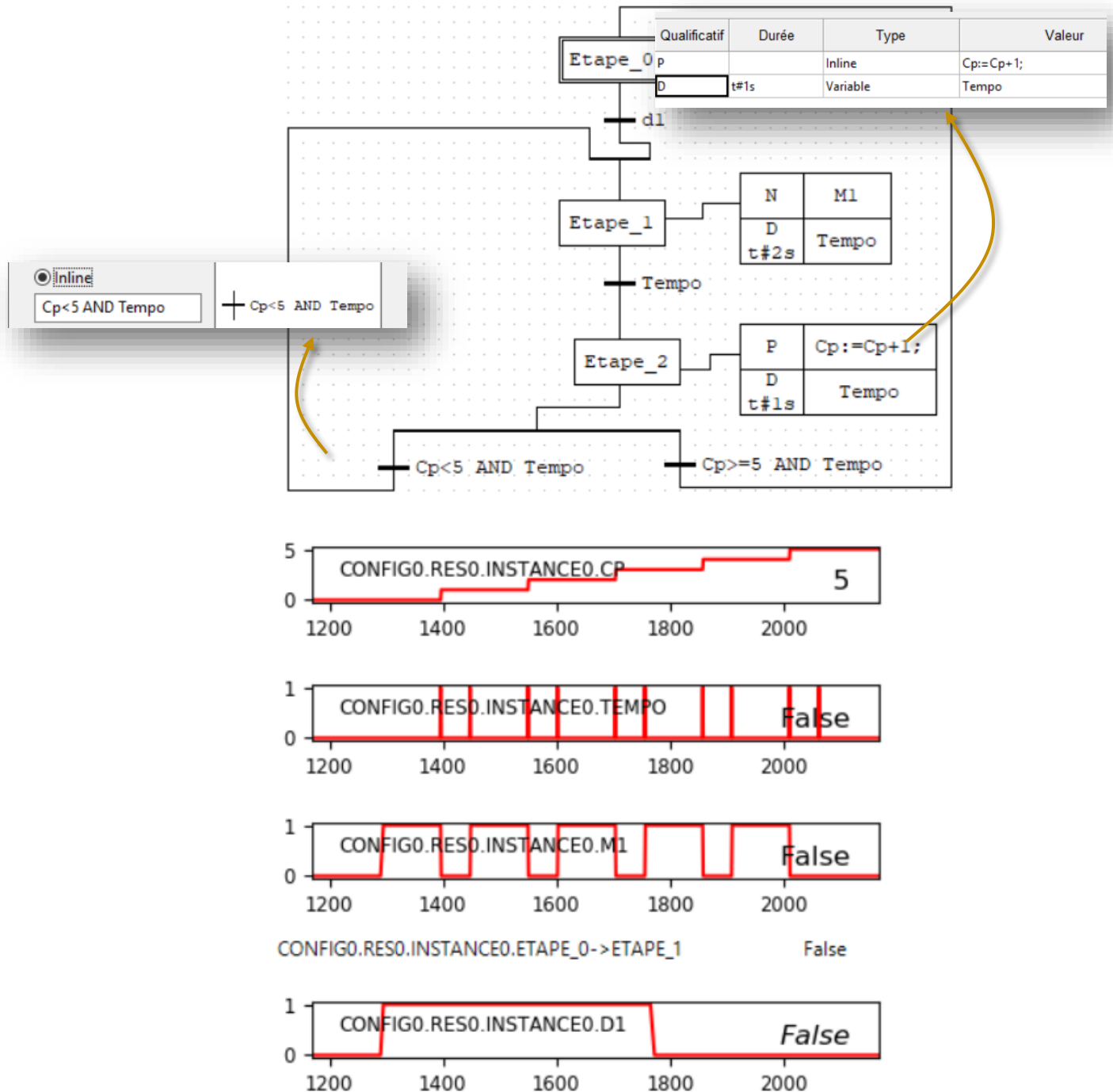
ANNEXE 4'

Compteurs (SFC)

Un exemple pour présenter cette fonctionnalité. On souhaite que le moteur M1 soit soumis cinq fois de suite au cycle suivant dès l'activation de d1 :

- activé 2s,
- en pause 1s.

Après ce cycle de 5 répétitions, le système s'arrête



Ressource publiée sur Culture Sciences de l'Ingénieur : <https://eduscol.education.fr/sti/si-ens-paris-saclay>