

Rapport Olympiades FANUC BUT 3



IUT GEII de l'école d'Evry-Courcouronnes (91)

Etudiant :

- Aubin TOURAIS
- Matthieu MOCHET
- Tom AUGER
- Nicolas FELLAH

Professeur référent :

- Sofiane Ahmed-Ali

SOMMAIRE

INTRODUCTION.....	3
MISSIONS À RÉALISER.....	3
MISSION 1 : QUESTIONS À RÉPONDRE.....	4
● Question 1 :.....	4
● Question 2 :.....	10
● Question 3 :.....	10
● Question 4 :.....	11
● Question 5 :.....	12
● Question 6 :.....	12
MISSION 2 : FANUC PICTURE.....	13
● Page 1 : OPÉRATEUR.....	14
● Page 2 : MAGASIN.....	23
● Page 3 : MAINTENANCE.....	25
● Page 4 : SIMULATEUR CAPTEURS (BONUS POUR NOUS).....	27
MISSION 3 : FANUC LADDER (PMC).....	28
● GRAFCET : OLYMPIADES 2024 - 2025.....	28
● ATTRIBUTION DES ENTRÉES :.....	29
● BOUTONS FANUC Picture : (TEST Simulation des capteurs sous picture).....	30
● TIMER :.....	32
Transitions des grafcets.....	36
- Sous-Programme - IUT EVRY BUT3.....	36
- Avant - Etape 0 :.....	36
- Après - Etape 0 :.....	36
- Après - Etape 1a :.....	38
- Après - Etape 2a :.....	39
- Après - Etape 3a :.....	40
- Après - Etape 4ai :.....	42
- Pour Etape 4aii et 1b - 1c :.....	43
- Après - Etape 2c :.....	45
- Après - Etape 3c :.....	46
- Après - Etape 4c :.....	47
- Après - Etape 4aii et 1b :.....	50
● Boutons Reset :.....	52
MISSION 4 : PROGRAMMATION ISO.....	56
MISSION 5 : GESTION DES ALARMES.....	68
Alarmes sur FANUC PICTURE : IHM.....	68
Alarmes sur FANUC LADDER III : PMC.....	70
CONCLUSION.....	72
Grafcet des coordonnées.....	73

INTRODUCTION

Notre équipe d'étudiants de l'IUT GEII d'Évry-Courcouronnes, engagée dans les olympiades FANUC BUT3, s'est efforcée de concevoir, programmer et optimiser des systèmes qui intègrent la technologie FANUC.

Ce document expose en détail les diverses tâches effectuées, depuis la programmation ISO jusqu'à la mise en place d'une interface utilisateur sur FANUC Picture, sans oublier la gestion des alertes et la simulation des capteurs.

Non seulement nous avions pour but de satisfaire les critères techniques fixés par le cahier des charges, mais aussi d'illustrer notre aptitude à innover et à offrir des solutions fiables et performantes.

MISSIONS À RÉALISER

Vous pourrez avant tout, retrouver ci-dessous les missions qui nous sont attribués pour les **Olympiades FANUC BUT3 CN**.

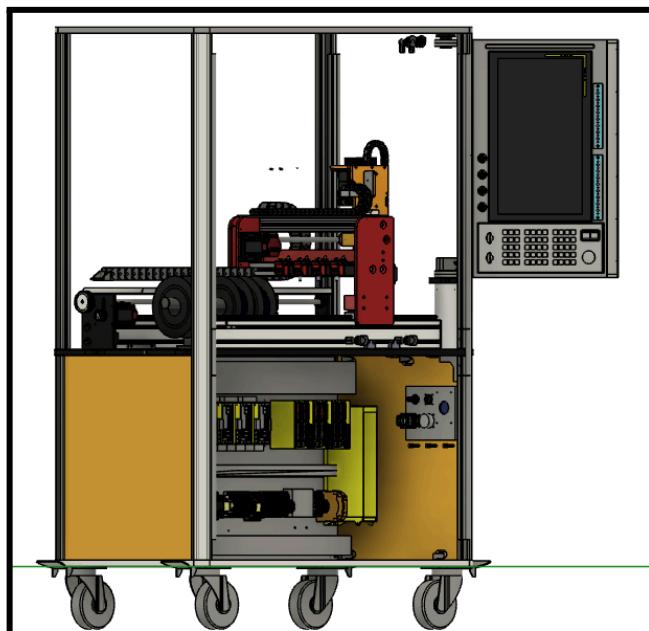
1	Mission n°1 : Répondre aux questions préalables, soyez le plus précis possible.
2	Mission n° 2 : Programmer à l'aide du logiciel FANUC Picture les différentes pages demandées.
3	Mission n°3 : En lien avec la Mission n°2, adaptez le programme automate (PMC) sans supprimer d'éléments déjà existants. Gérez les différents capteurs, les séquences de changement de buses, de matériaux et consignes...
4	Mission n°4 : Si besoin vous pouvez modifier un programme ISO (éventuellement fourni en annexe) en y ajoutant des macro variables (#), des codes M, en ajoutant des numéros de lignes (N1, N2, N3...) ... si votre développement vous amène à le faire. S'agissant d'un vrai programme, vous pouvez tronquer bon nombre de lignes pour réaliser vos essais.
5	Mission n°5 : Développer les alarmes, et messages comme demandé
6	Mission n°6 : Rédigez un rapport (les détails et la description de votre démarche sera valorisée) et transmettez à FANUC France la cellule complète CNC GUIDE, les sources FANUC PICTURE, le projet automate et le backup complet de la cellule.

MISSION 1 : QUESTIONS À RÉPONDRE

Pour la première mission, nous devons répondre au question qui nous a été demandée dans le cahier des charges, que l'on répondra au fur et à mesure.

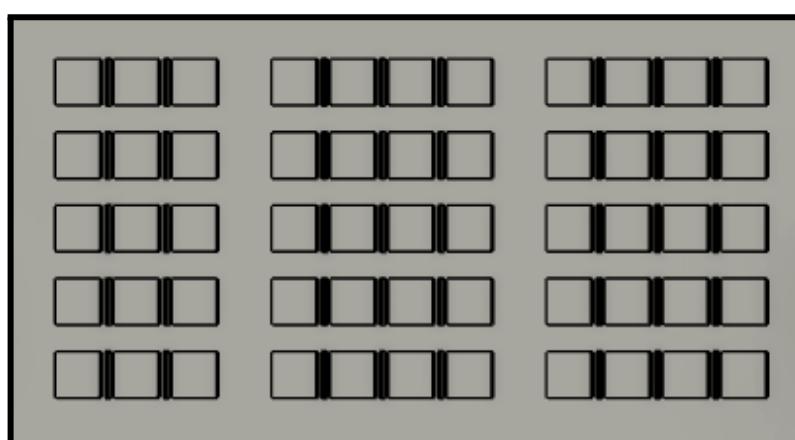
• Question 1 :

A partir des images (ou du fichier .STEP), identifiez les éléments suivants de la machine :



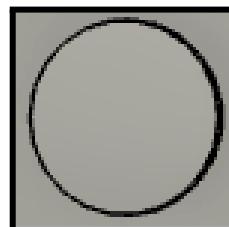
- Clavier Opérateur (ou MOP)

Le clavier Opérateur permet d'interagir avec les modes d'utilisation de la machine, mode de fonctionnement automatique, mode de fonctionnement manuel pour bouger les axes, démarrer la machine, lancer un programme, simuler des capteurs avec des boutons. Celui-ci se trouve en dessous de l'écran de l'ordinateur.



- L'arrêt d'urgence

L'arrêt d'urgence permet d'arrêter la machine en cas d'urgence en arrêtant le programme en cours et en coupant l'alimentation des moteurs. Celui-ci se trouve à côté du clavier MOP. (vu ci-dessus)



- Plateau d'impression

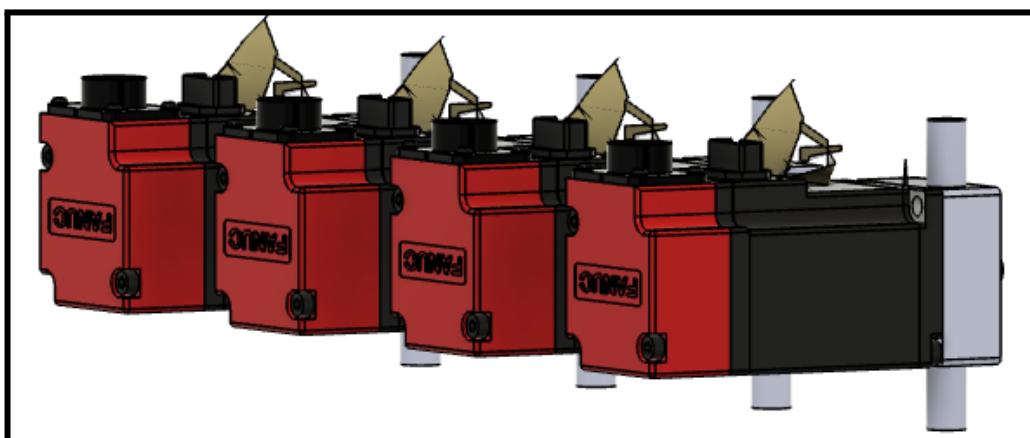
Le plateau d'impression permet d'accueillir l'impression 3D en cours d'impression: Celui-ci se trouve à l'intérieur de la machine (dans la partie vitrée)



- Moteur Extrusion

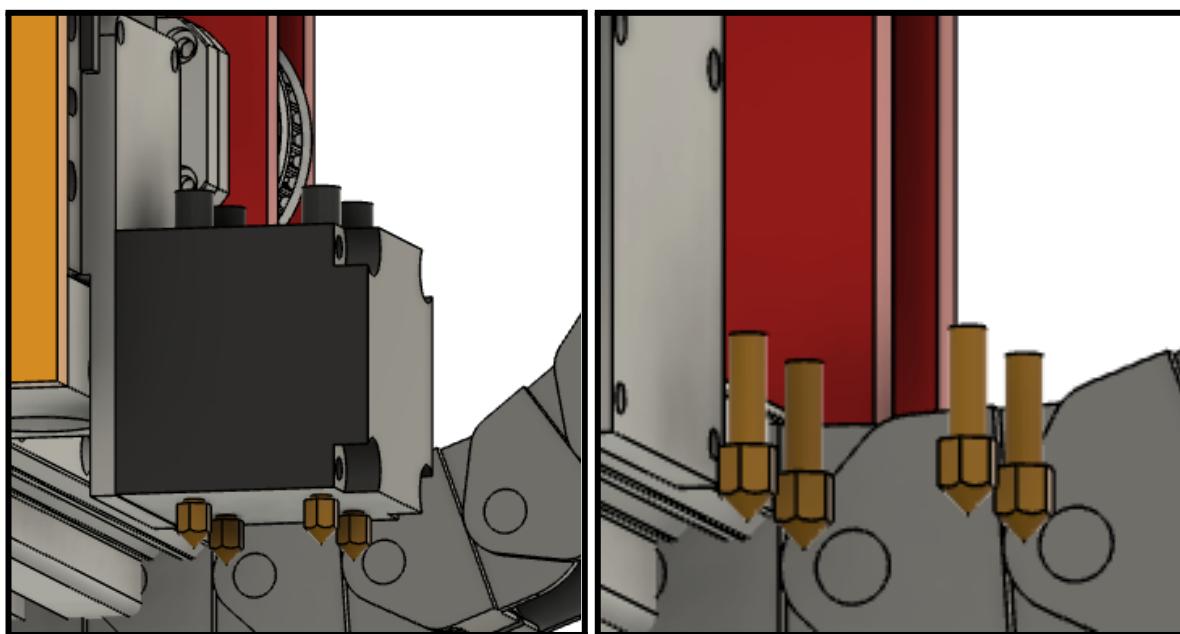
Les moteurs à extrusion permettent de pousser le filament en plastique de manière contrôlée dans l'extrudeur. Celui-ci se trouve dans la zone de travail, la partie vitrée.

(à voir)



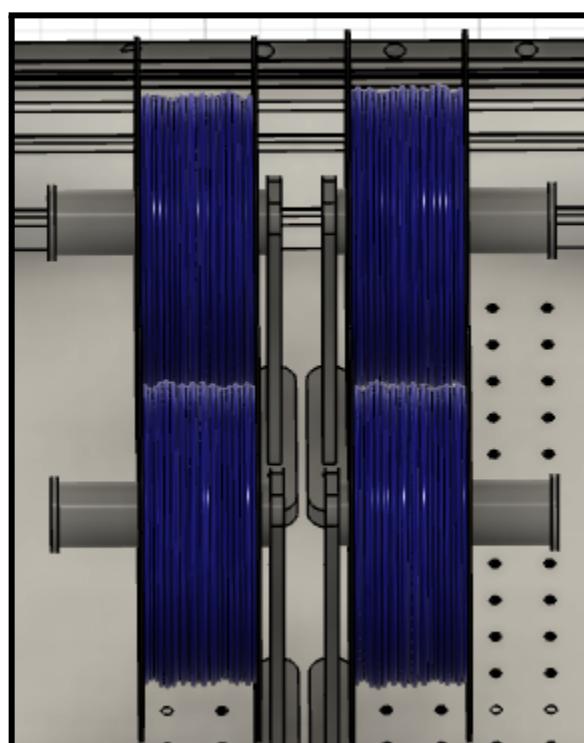
- Les 4 buses

Les buses permettent de faire fondre le plastique à un certain débit précis et de déposer le matériau. Ils se trouvent dans le porte buses que l'on peut trouver dans la photo de gauche.



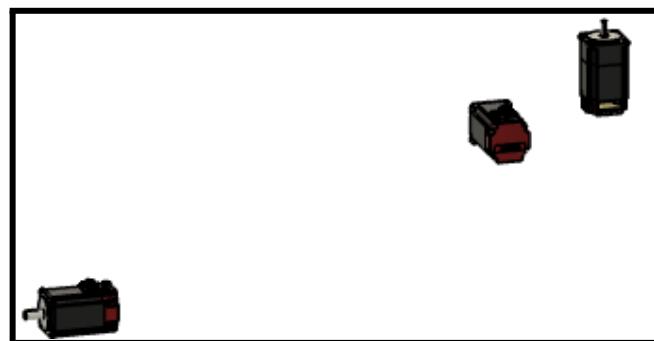
- Le magasin de bobines

Le magasin de bobines permet de stocker les bobines à l'intérieur de la machine et de maintenir le filament en place. Celui-ci se trouve à l'intérieur de la machine dans la partie vitré supérieure (au niveau de la plaque)



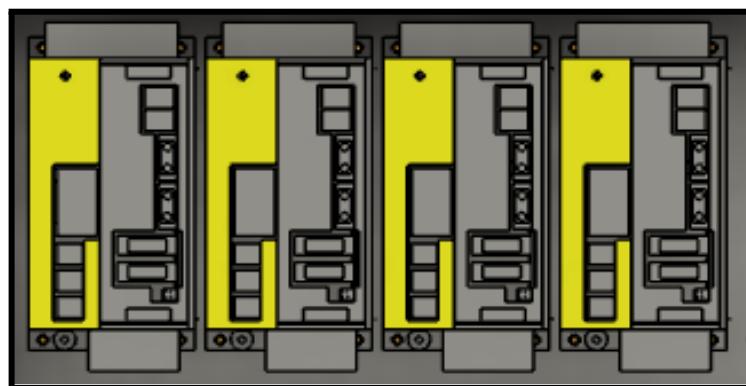
- Les moteurs X, Y, Z et E

Les moteurs permettent de déplacer les axes, X sur la longueur, Y sur la largeur, Z sur la hauteur et E sont les moteurs à extrusion. Celui-ci se trouve dans la partie vitrée supérieure.



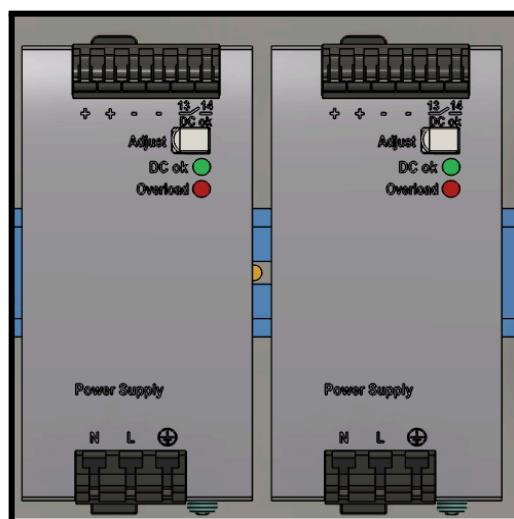
- Les amplificateurs d'axe

Les amplificateurs permettent d'amener le courant aux moteurs.
Celui-ci se trouve dans l'armoire électrique (partie puissance)



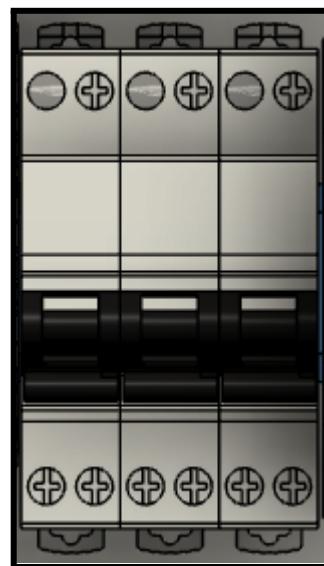
- Au moins une alimentation 24VDC

L'alimentation 24VDC permet d'alimenter les moteurs.
Celui-ci se trouve dans l'armoire électrique. (partie puissance)



- **Au moins un disjoncteur**

Les disjoncteurs permettent de couper le courant d'un certain élément de la machine.
Celui-ci se trouve dans l'armoire électrique. (partie puissance)



- **La CNC (il s'agit d'une CNC dites stand-alone, car elle se trouve dans l'armoire)**

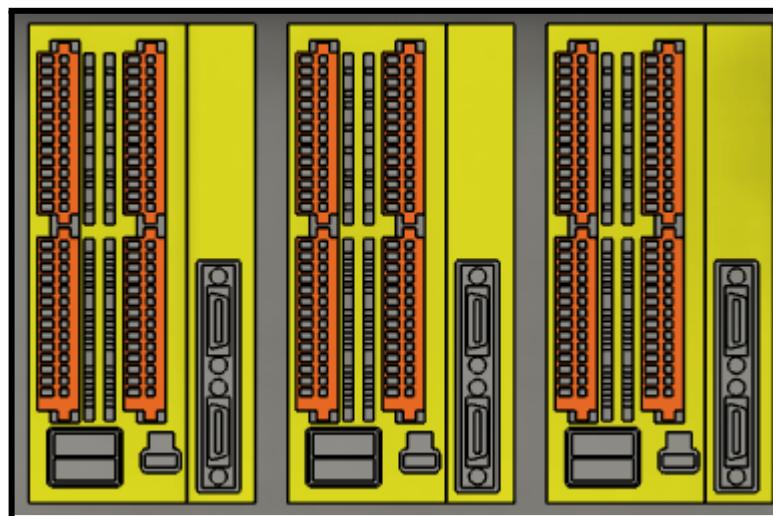
La CNC est la tour de communication, elle permet de gérer l'ISO pour renvoyer les comptes rendus F. Celui-ci se trouve dans l'armoire électrique (partie commande)



- Les cartes d'entrées/sorties

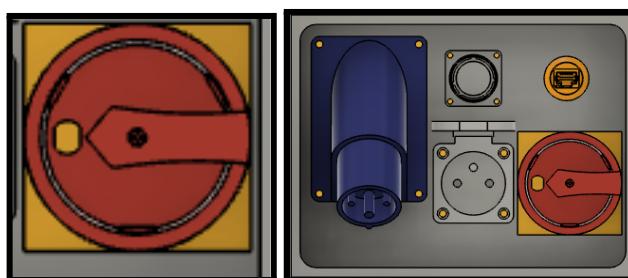
Les cartes entrées/sorties permettent de recevoir les informations des capteurs pour les traiter et d'envoyer les commandes aux actionneurs.

Celui-ci se trouve sur dans la partie commande de l'armoire électrique.



- Le sectionneur

Le sectionneur permet d'éteindre la machine. Celui-ci se trouve à côté de l'alimentation du secteur.



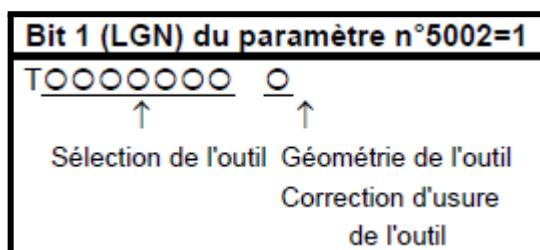
● Question 2 :

A quoi servent les codes M ? Les codes T ?

Les codes M nous permettent de contrôler des actions spécifiques qui ne sont pas directement liées à la machine, mais qui gèrent les actions comme l'activation ou la désactivation des dispositifs, l'arrêt du programme ou pour finir la gestion du cycle de travail.

Exemple	
Une commande M dans un mode bloc à bloc	Plusieurs codes M dans un même bloc
M40; M50 ; M60; M70; M80; G28G91X0Y0Z0; : : :	M40M50M60M70M80; G28G91X0Y0Z0; : : : :

Tandis que les codes T permettent de transmettre un signal d'échantillonnage, un signal de code à la machine-outil en spécifiant une valeur numérique à 8 chiffres.



● Question 3 :

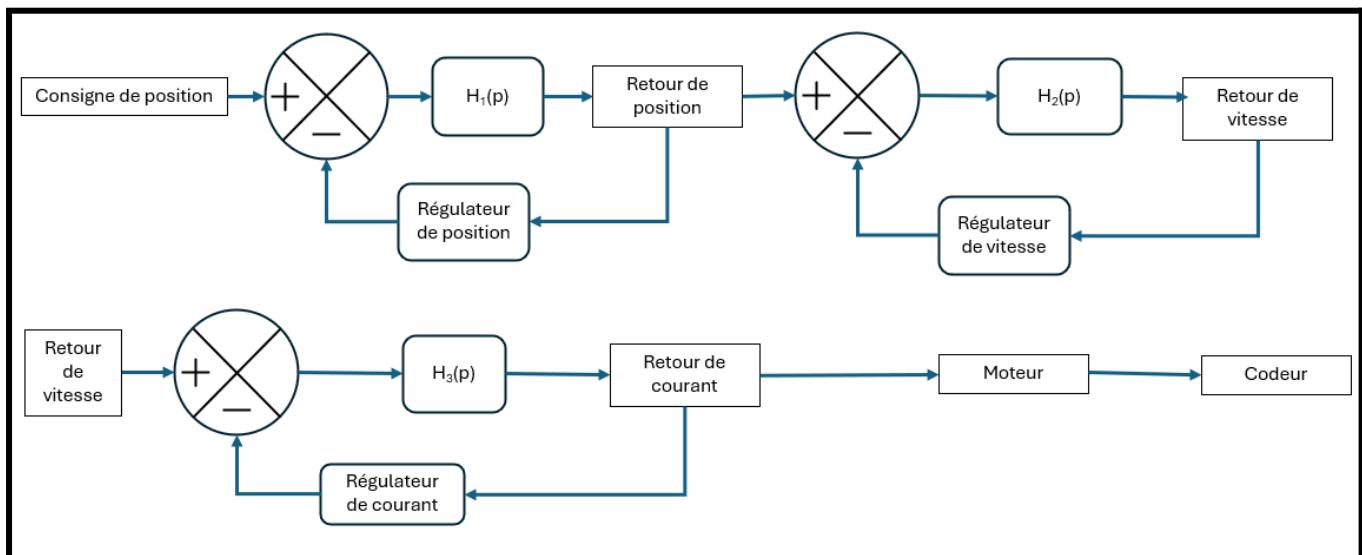
Pour chaque couche (layer) d'impression, quels sont les axes interpolés ?

Les axes qui sont interpolés pour chaque couche d'impression sont :

- L'axe X et Y permettent le dessin des différentes couches sur le plan horizontal.
- L'axe Z permet la progression des couches en superposant les couches les unes sur les autres.

● **Question 4 :**

Représentez schématiquement la boucle d'asservissement d'un axe CN en représentant les blocs vitesse, courant et position, le moteur, ainsi que le codeur.



Les axes X, Y et Z ont des courroies. Cet élément de la chaîne cinématique peut se détériorer avec le temps, amener de l'imprécision où encore casser. Quel matériel dois-je intégrer à la chaîne cinématique pour prévenir cela ?

Ajoutez un capteur de tension de la courroie, si elle est endommagée, sa tension est sous-optimale, donc le capteur le déetectera.

Ajoutez un système de compensation de la tension, ce système s'activera pour augmenter la tension dans la courroie pour compenser la tension optimale.

- **Question 5 :**

Comment s'appelle le phénomène ci-contre ?



Le phénomène ci-contre est le warping ou gauchissement est causé par une contraction du matériau en refroidissant trop rapidement, ce qui entraîne une tension dans la pièce et provoque le soulèvement des coins. Le Warping est plus fréquent pour les pièces de grande taille ou pour les matériaux qui ont une forte contraction lorsqu'ils refroidissent.

- **Question 6 :**

Comment s'appelle en impression 3D le logiciel qui transforme en fichier 3D en programme prêt à exécuter ? (exemple : Ultimaker Cura, OctoPrint...)

Le logiciel permettant de transformer un fichier 3D en programme prêt à exécuter s'appelle un slicer.

MISSION 2 : FANUC PICTURE

A NOTER: Le nom du fichier principal Picture s'appelle FANUC 23-24 au lieu de FANUC 2024-2025

Rapport Client

Les pages de l'IHM à présenter au client doivent contenir les éléments suivants :

Page : Opérateur

- Position des axes
- Affichage de consigne et possibilité d'ajuster les températures du plateau et des buses
- Ajustement de la position de l'axe Z si besoin [ajouter +/- 0.1mm]
- Ajustement de la vitesse programmée
- Sélection et visualisation des modes de fonctionnement [Auto/Edit/MDI/Manuel].
- Départ cycle / Stop cycle avec retour information
- Validation retrait pièce fabriquée
- Proposer une solution pour afficher le temps écoulé depuis le départ cycle
- Proposer un affichage de la progression en % en fonction de la ligne du programme ISO

Page : Magasin

Les bobines étant liées par un code d'identification : **AABB**

Ainsi le code 1001 correspond à du PLA noir, 1202 à du PETG rouge...

On vous demande alors d'intégrer à l'IHM la simulation des bobines dans le magasin [vide ou non vide en fonction du poids, en passant par la simulation] et d'intégrer la séquence pour renseigner un type de bobine.

- Possibilité de sélection manuellement une buse [dans le mode adéquat]

Bonus :

- Animation et esthétisme du magasin

Page : Maintenance

- Affichage des alarmes et messages
- Créer au moins un message opérateur - exemple : Valider le plateau vide, plateau en chauffe, buse en chauffe, pièce finie...)
- Créer au moins une alarme - exemple : Bobine vide démarrage impossible, buse HS [si consigne non atteinte au bout de xxx temps...]
- Une aide [popup, page contextuelle... au choix] pour résoudre un défaut [au moins un]

Code couleur :

AA représente la matière :

- 10 : PLA
- 11 : ABS
- 12 : PETG
- 13 : Nylon

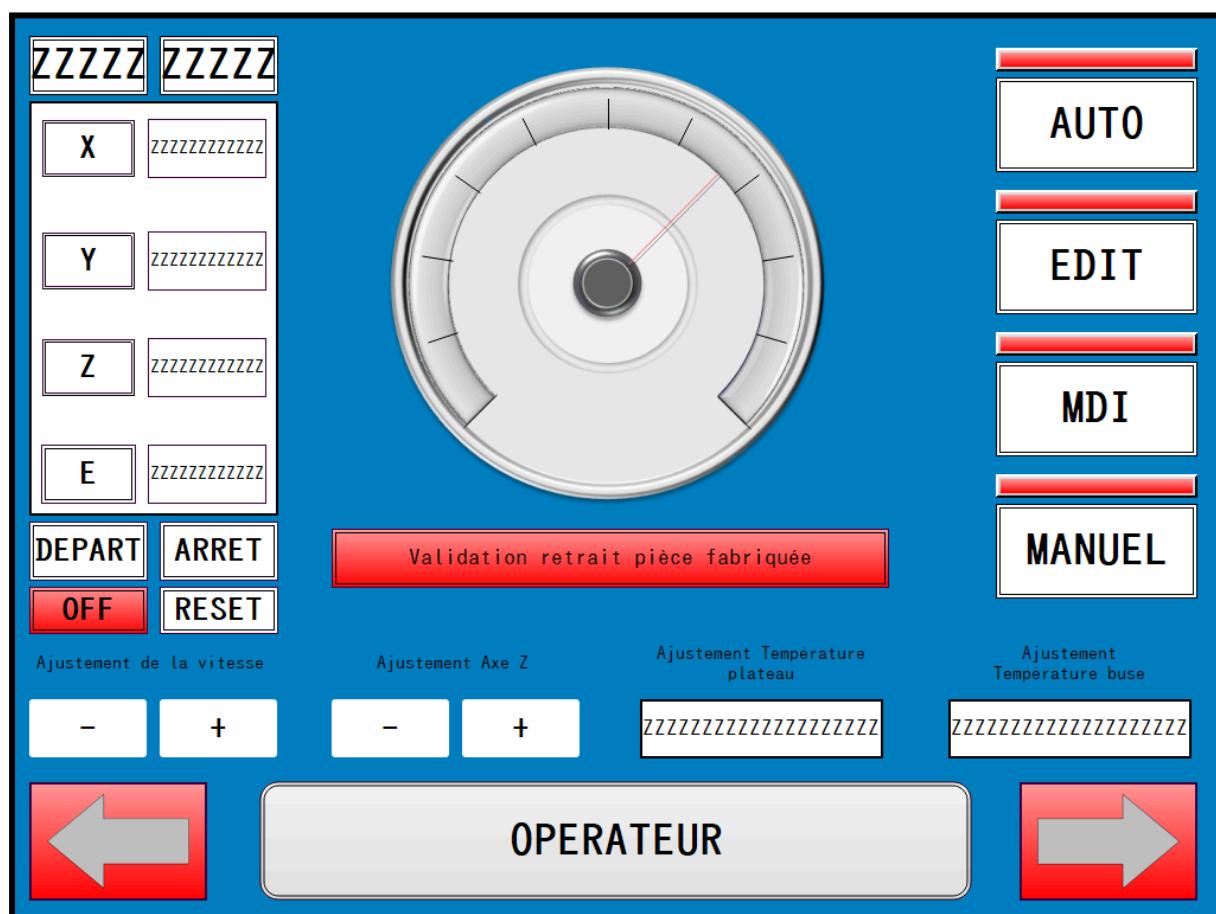
BB représente la couleur :

- 00 : blanc
- 01 : noir
- 02 : rouge
- 03 : bleu
- 04 : gris

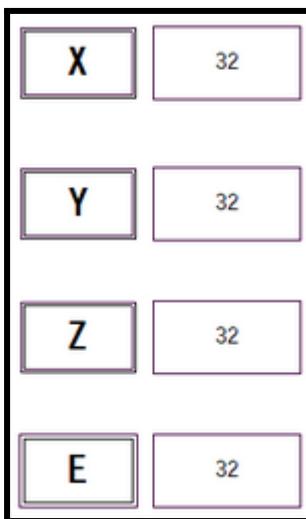
• **Page 1 : OPÉRATEUR**

Page : Opérateur

- Position des axes
 - Affichage de consigne et possibilité d'ajuster les températures du plateau et des buses
 - Ajustement de la position de l'axe Z si besoin [ajouter +/- 0.1mm]
 - Ajustement de la vitesse programmée
 - Sélection et visualisation des modes de fonctionnement [Auto/Edit/MDI/Manuel].
 - Départ cycle / Stop cycle avec retour information
 - Validation retrait pièce fabriquée
 - Proposer une solution pour afficher le temps écoulé depuis le départ cycle
 - Proposer un affichage de la progression en % en fonction de la ligne du programme ISO



Position des axes :



Voici la solution que nous proposons pour afficher la position des axes. On peut y voir les différents axes : X, Y et Z et E représentant la hauteur de la buse.

Les valeurs affichées sont les valeurs de la CNC :

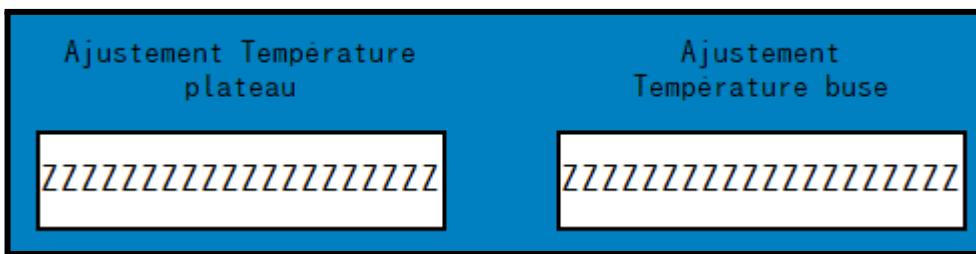
G00	ABSOLUE	DISTANCE
X	0.000	0.000
Y	0.000	0.000
Z	0.000	0.000
E	0.000	0.000

Nous utilisons la fonction absolute en paramètre de lecture, c'est celle-ci qui nous permet de pouvoir lire les valeurs directement dans la page IHM, pour le **X** on a **absolute(1)**, pour **Y absolute(2)** et ainsi de suite.

The dialog box contains the following settings:

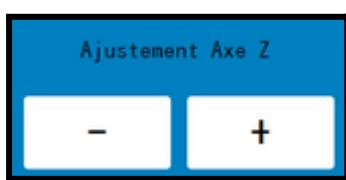
- NoAction :
- In/Out Target : Function
- PMC Path : 1st PMC
- PMC Area : R
- Address : 0
- Symbol :
- Project variable :
- ReadFunction : absolute(1)
- WriteFunction :
- CheckFunction :
- Data Type : Decimal
- In/Out Format : Decimal
- Set max. and min. values : Max. value : 127 Min. value : -128
- Use Decimal Point : Value : 3
- Digits / Bytes specification : Value : 1
- Non zero suppress :
- Left Align :
- Identification number : Value : 0
- Use focus index for the identification number :
- Popup Screen Name :

Affichage de consigne et ajustement température plateau et des buses :



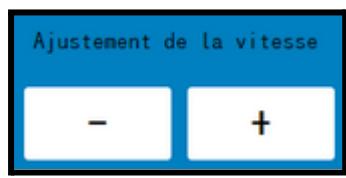
Pour régler la température de du plateau ainsi que la température des buses, nous avons mis des input, ainsi, l'utilisateur rentre la température nécessaire à son utilisation.

Ajustement de l'axe Z :

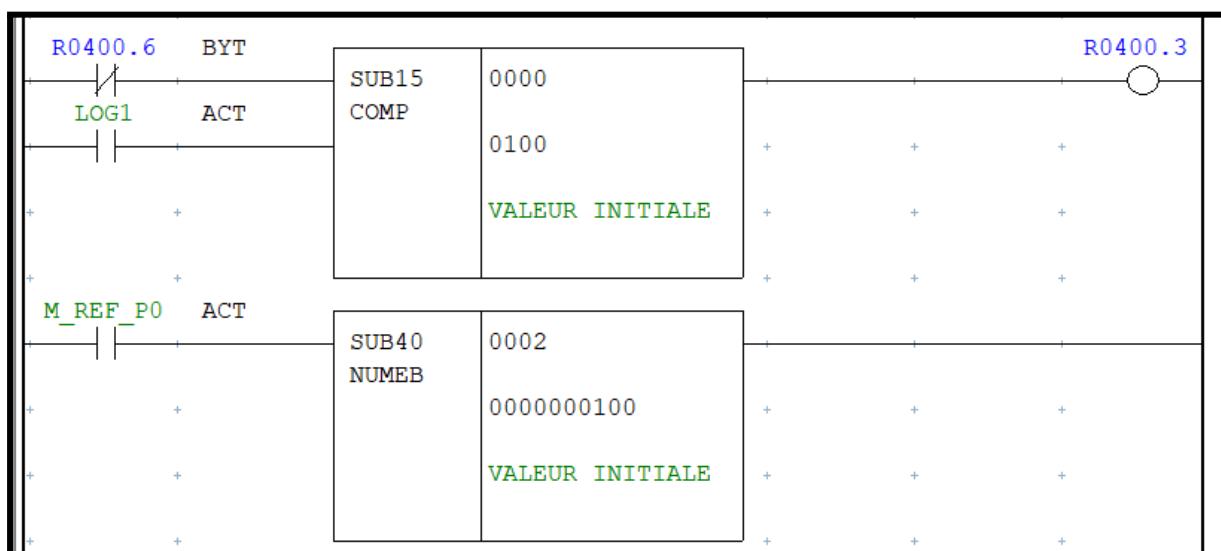
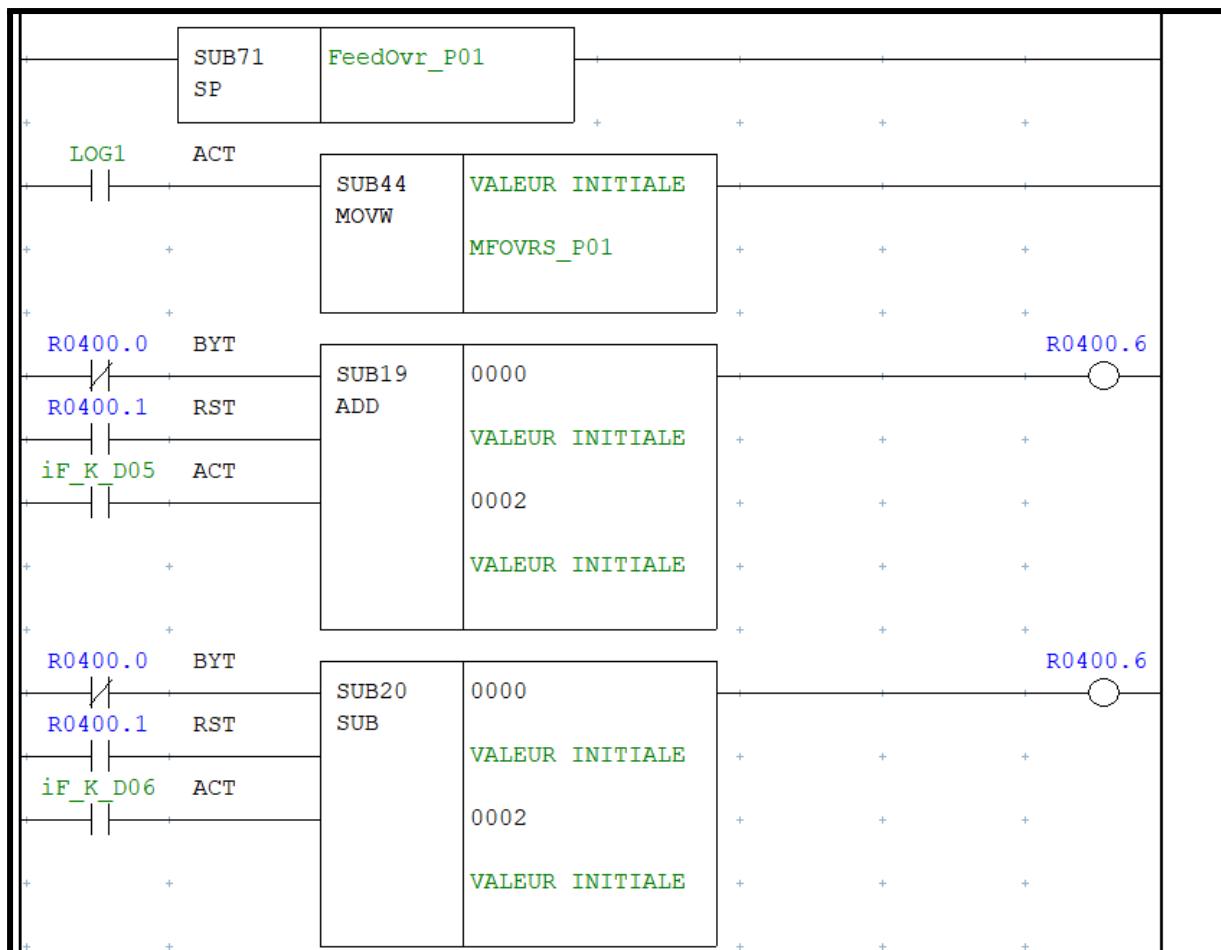


L'ajustement de l'axe Z se fait grâce au sous programme que nous avons réalisé dans le ladder. Il y a deux boutons classiques reliés au programme ladder qui incrémentent ou décrémentent une valeur stockée dans la datatable qui augmente de 0.1

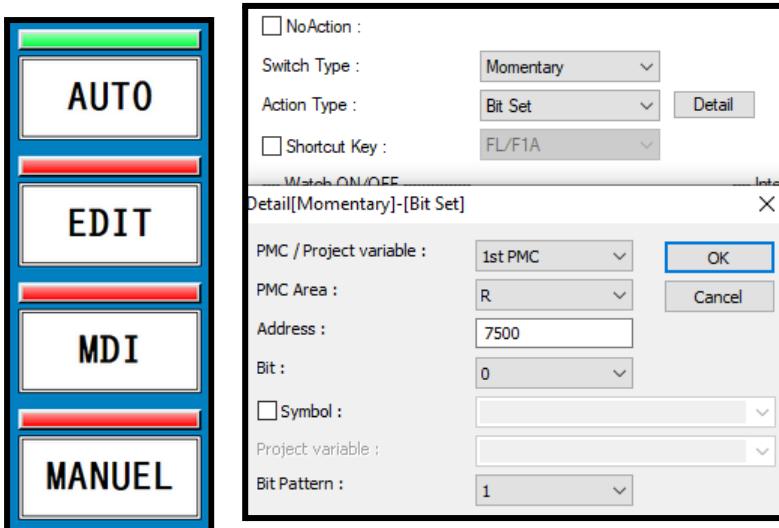
Ajustement de la vitesse :



Pour l'ajustement de la vitesse, nous devons remplacer la molette permettant de régler la vitesse en pourcentage. Pour cela, nous avons essayé de modifier le sous programme ladder **feedback_P01**. Ce que nous avons fait n'est pas encore fonctionnel.



Sélection et visualisation de des modes de fonctionnement :



Pour sélectionner les différents modes de fonctionnement nous avons utilisé des boutons momentanés qui sélectionnent un mode de fonctionnement souhaité. Nous lui avons affecté la variable R7500.0

Départ cycle stop cycle avec retour de l'information :



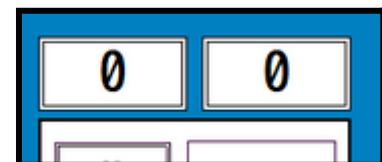
Les boutons ci-présents sont des boutons classiques, ils ont été configurés de sorte à faire référence au clavier MOP, le bouton départ correspond au bouton start cycle du clavier, le bouton arrêt au bouton stop, et le bouton reset au bouton reset.

Validation retrait pièce :



Il s'agit simplement d'une led qui, à chaque re-bouclage permet de confirmer que la pièce fabriquée a bien été retirée

Temps écoulé en fonction du départ :

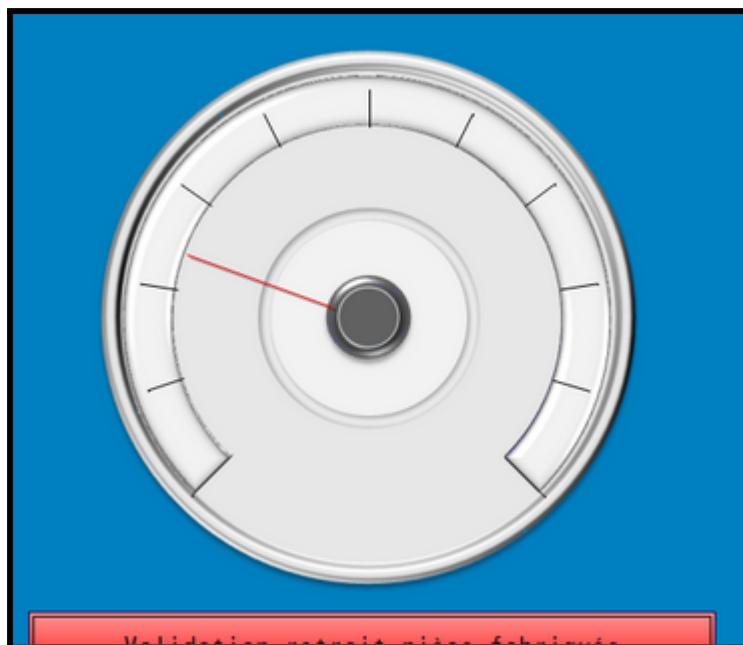


Input

General	Character	Action	Focus	Interlock	Image	Option	NoAction	Image
<input type="checkbox"/> NoAction :								
In/Out Target :	PMC							
PMC Path :	1st PMC							
PMC Area :	D							
Address :	49							
<input type="checkbox"/> Symbol :								
Project variable :								

Pour afficher le temps écoulé depuis le départ du cycle, nous avons utilisé des timers dans un sous programme du ladder, les données de ces timers sont stockées dans la datatable, on met donc dans le paramètre PMC Area D qui correspond à la datatable, et l'adresse dans laquelle nous retrouvons les données dans notre cas l'adresse 49.

Affichage de la progression en % en fonction de la ligne du programme ISO :



L'afficheur d'avancement du programme à été configuré de sorte à lire la macro 801.

Pour faire fonctionner l'afficheur d'avancement du programme ISO, nous l'avons dans un premier temps testé via un petit programme ISO qui incrémente une valeur et la sauvegarde dans une variable qui est lue en temps réel par l'afficheur.

```
<TUTO> G56 G90 G94 ;
GOTO 20 ;
N10 ;
#700 =#700+0.001 ;
IF[#700 EQ 100] GOTO 20 ;
GOTO 10 ;
;
;
;
N20 M40 ;
#700 =0 ;
GOTO 10 ;
%
```

700	43.2200
-----	---------

En fonction de l'étape dans laquelle se trouve le ladder, l'ISO affiche les valeurs suivantes :

1a → 20%

2a → 40%

3a → 60%

4ai → 80%

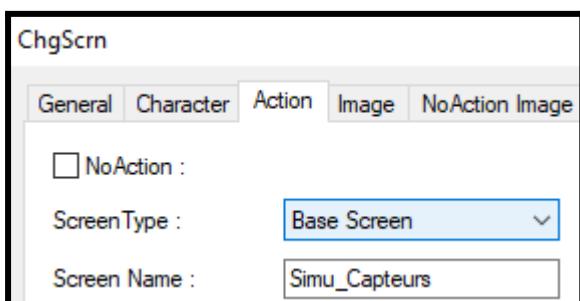
1c → 65%

2c → 70%

3c → 75%

4c → 80%

Changement de page :



Pour changer de page, nous avons utilisé des blocs ChgScrn, il suffit simplement de renseigner le nom de la fenêtre sur laquelle nous souhaitons nous rendre dans le paramètre Screen Name.

• Page 2 : MAGASIN

Page : Magasin

Les bobines étant liées par un code d'identification : **AABB**

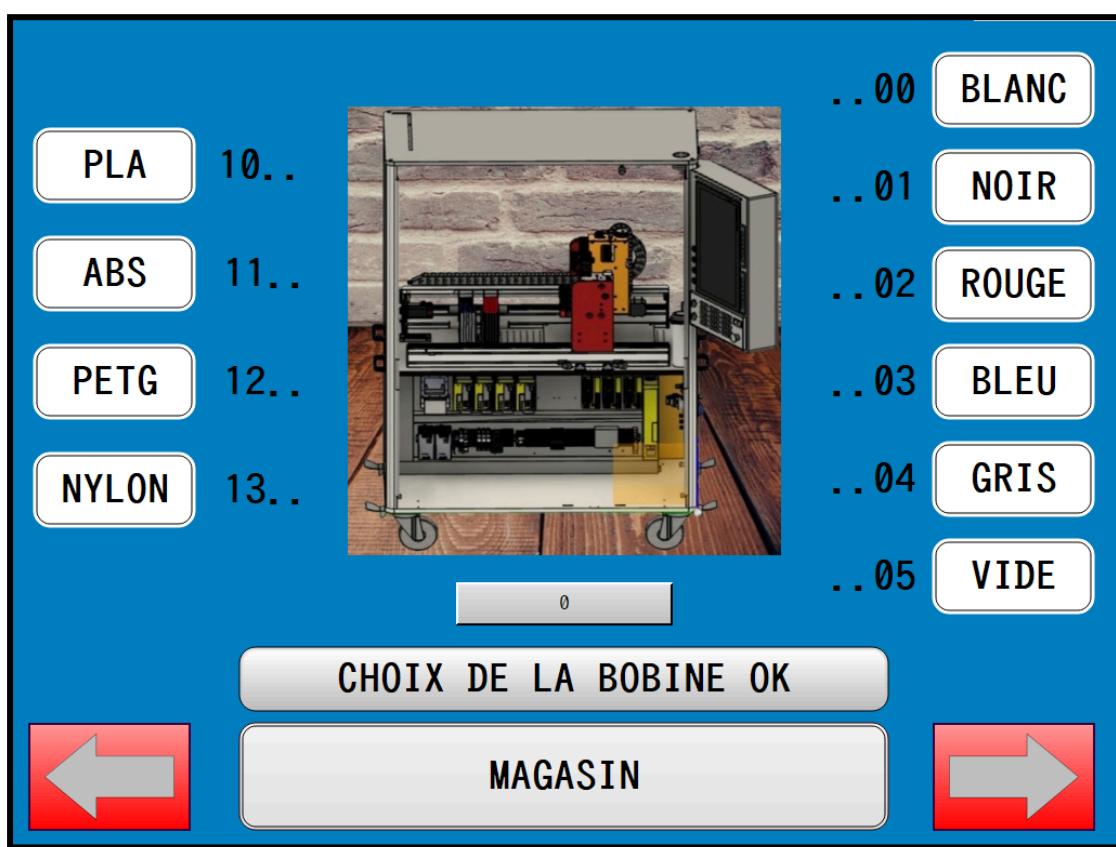
Ainsi le code 1001 correspond à du PLA noir, 1202 à du PETG rouge...

On vous demande alors d'intégrer à l'IHM la simulation des bobines dans le magasin [vide ou non vide en fonction du poids, en passant par la simulation] et d'intégrer la séquence pour renseigner un type de bobine.

- Possibilité de sélection manuellement une buse [dans le mode adéquat]

Bonus :

- Animation et esthétisme du magasin



Code couleur :

AA représente la matière :

- 10 : PLA
- 11 : ABS
- 12 : PETG
- 13 : Nylon

BB représente la couleur :

- 00 : blanc
- 01 : noir
- 02 : rouge
- 03 : bleu
- 04 : gris

Il nous a été demandé de pouvoir sélectionner la matière avec le code qui lui a été attribué et pareillement pour la couleur. Pour y arriver nous proposons la solution suivante :

Pour sélectionner le matériaux ainsi que la couleur, nous avons décidé de laisser l'utilisateur rentrer le code correspondant dans l'espace suivant :



Ten-key Title :	<input type="text" value="Macro Variable #802"/>
ReadFunction :	<input type="text" value="rdmacro(802,99,0)"/>
WriteFunction :	<input type="text" value="wmacro(802,0,0,99,0)"/>
CheckFunction :	<input type="text"/>

Onglet : Tinput

Cette page concerne les paramètres du tenkey, nous écrivons et lisons la macro 802.

Il faut, une fois que le code correspondant au matériaux et à la couleur souhaitée à été entré, que l'utilisateur confirme que la bobine est bien celle qu'il a choisie en appuyant sur le bouton choix de la bobine ok.

• **Page 3 : MAINTENANCE**

Page : Maintenance

- Affichage des alarmes et messages
- Créer au moins un message opérateur - exemple : Valider le plateau vide, plateau en chauffe, buse en chauffe, pièce finie...)
- Créer au moins une alarme - exemple : Bobine vide démarrage impossible, buse HS [si consigne non atteinte au bout de xxx temps...]
- Une aide (popup, page contextuelle... au choix) pour résoudre un défaut (au moins un)



Pour afficher les alarmes et les messages d'erreur, nous avons utilisé des **Ndisp**.

Messages :

Function Name :	<input checked="" type="checkbox"/> Display :
<input type="text" value="rdopmsg2(1,-1,99,0)"/>	

Alarmes :

Function Name :	<input checked="" type="checkbox"/> Display :
<input type="text" value="rdalminfo(1,99,10,99,0)"/>	

PENSEZ A METTRE LES TEMPERATURES DE PLATEAU ET DE BUSE CORRESPONDANT AU MATERIAU CHOISI

Matériaux:	Température		Coordonnées Bobines		
	Plateau (#804)	Buse (#805)	X	Y	Z
PLA	60°C	200°C	400	200	100
ABS	90°C	240°C	400	200	100
PETG	75°C	240°C	400	100	100
NYLON	80°C	255°C	500	100	100

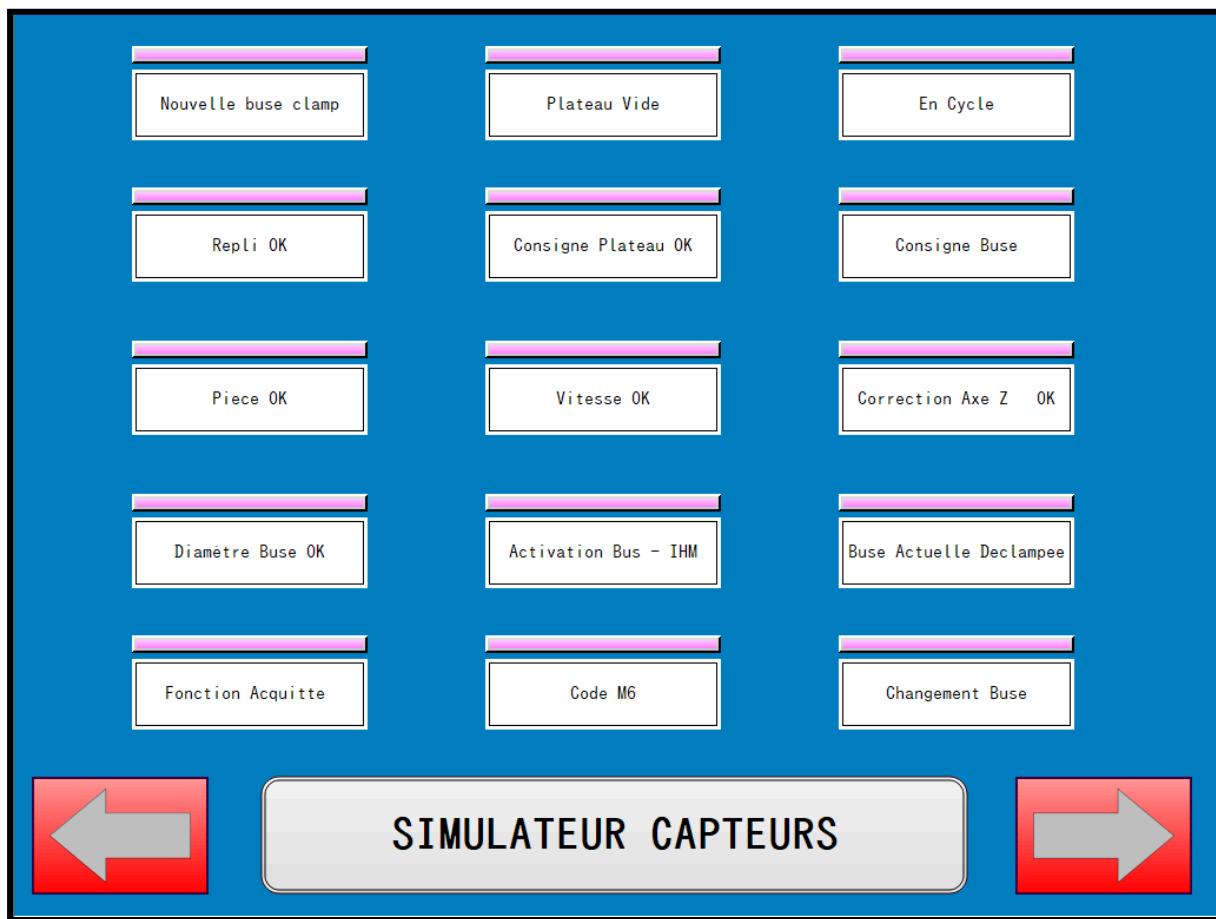
AIDE



Il nous a également été demandé de créer une page d'aide.

Nous avons donc fait une pop up qui comprend les températures du plateau, des buses et des coordonnées des bobines. Pour configurer une page en pop up, il suffit simplement de cocher la case pop up dans les paramètres du screen.

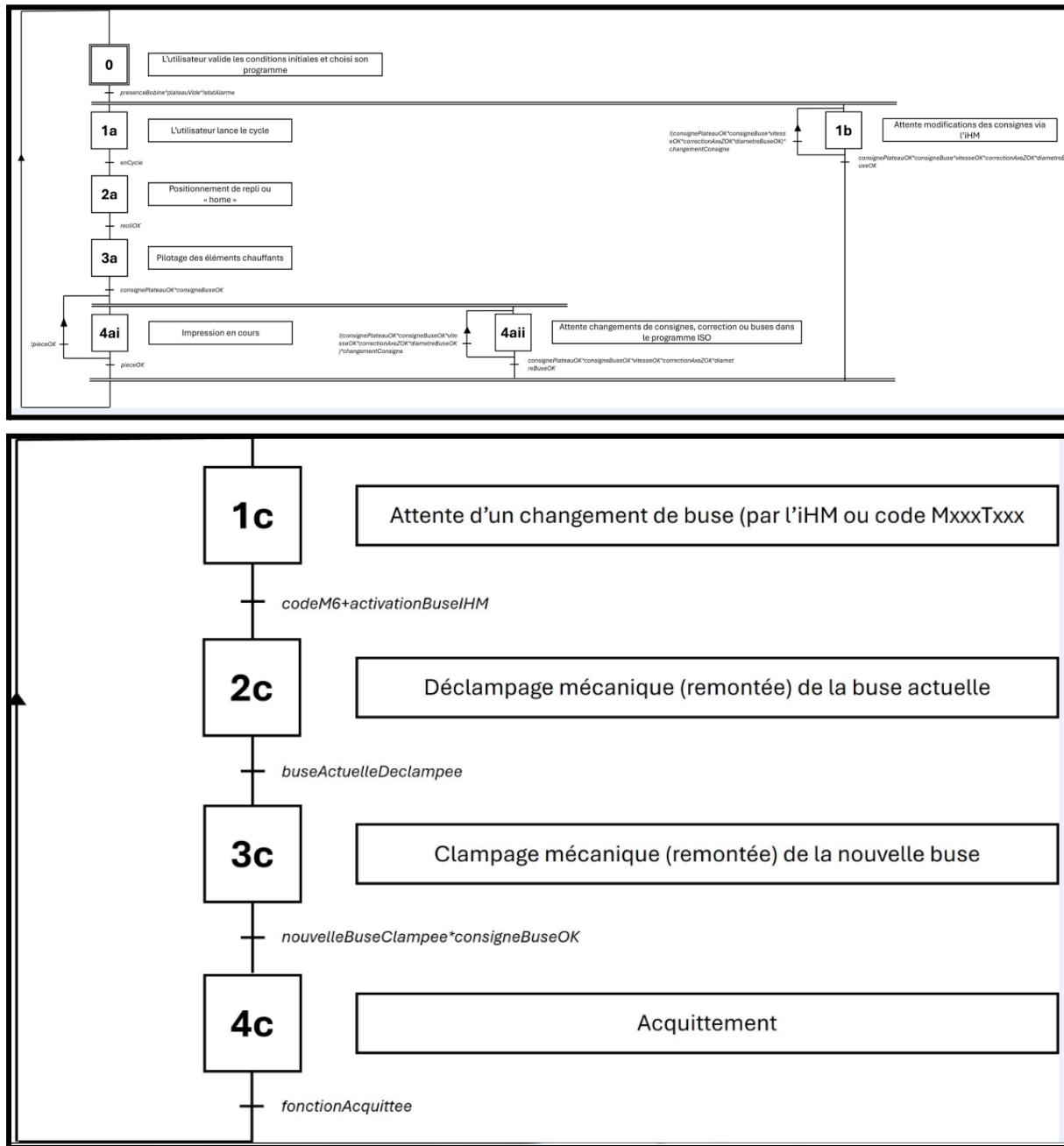
- **Page 4 : SIMULATEUR CAPTEURS (BONUS POUR NOUS)**



Pour pouvoir simuler et tester le bon fonctionnement de nos programmes ainsi que le picture, nous avons créé une page Simulateur Capteurs avec des boutons nous permettant de simuler les capteurs, ce qui nous permet d'avancer dans les étapes du grafct.

MISSION 3 : FANUC LADDER (PMC)

- **GRAFCET : OLYMPIADES 2024 - 2025**

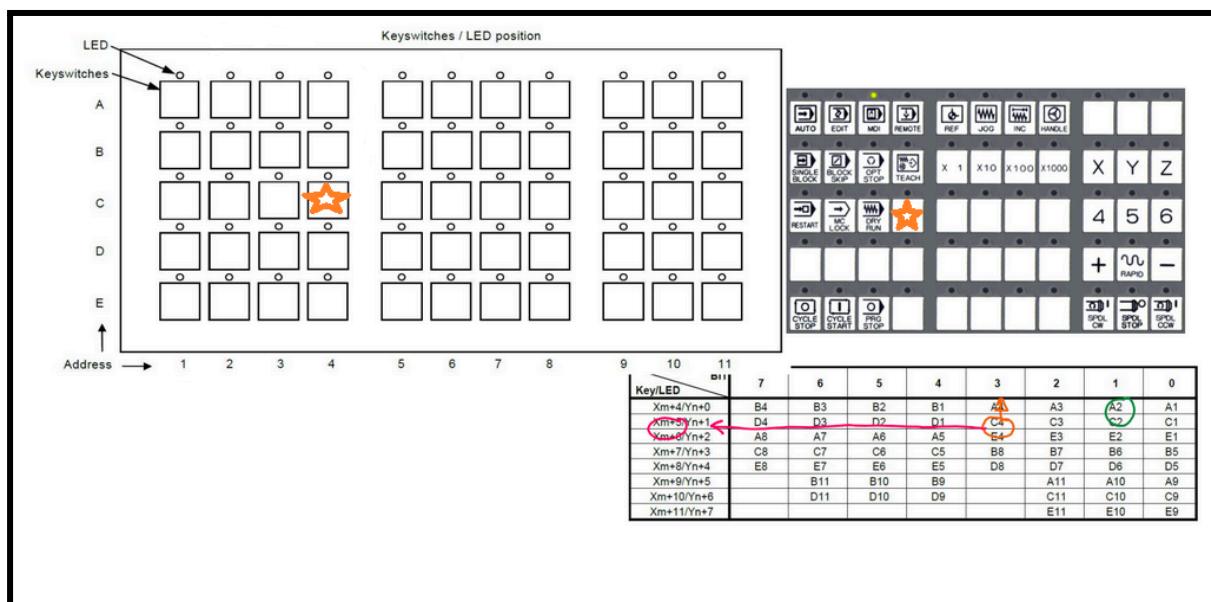


A NOTER :

Tout au long du projet, nous avions pensé qu'au début il y avait une divergence en ET et à la fin une Convergence en OU car pour nous on ne peut pas imprimer et en même temps changer la buse ça ne nous semblait pas logique.

Lors de la compilation du grafset, il y a 2 erreurs dues au non appel des sous programmes (speed axe Z Evry et vitesse BUT3 Evry)

- ATTRIBUTION DES ENTRÉES :**



Voici le clavier qu'on voulait faire avec toutes les transitions marquées dessus pour que ça soit plus lisible.

Si on devait se qualifier pour la finale on aurait bien aimé savoir comment faire cela sur la journée de préparation de la finale, afin de pouvoir savoir comment implémenter une image sous virtuelle.

*31	X0105.3	iF_K_C04	ON	op55: key C04
*32	X0105.4	Bouton Reset	Bouton RESET Clavier	op55: key D01
*33	X0105.5	BTN_TA_EVRY	TEST ALARME EVRY	op55: key D02
*34	X0105.6	iF_K_D03	ChangementConsigne	op55: key D03
*35	X0105.7	iF_K_D04	OFF	op55: key D04
*36	X0106.0	iF_K_E01	Bouton STOP	op55: key E01
*37	X0106.1	iF_K_E02	Bouton START	op55: key E02
<hr/>				
*47	X0107.4	iF_K_C05	Présence Bobine	op55: key C05
*48	X0107.5	iF_K_C06	Plateau Vide	op55: key C06
*49	X0107.6	iF_K_C07	En_Cycle	op55: key C07
*50	X0107.7	iF_K_C08	Repli_OK	op55: key C08
*51	X0108.0	iF_K_D05	ConsignePlateau_OK	op55: key D05
*52	X0108.1	iF_K_D06	ConsigneBuse_OK	op55: key D06
*53	X0108.2	iF_K_D07	Pièce_OK	op55: key D07
*54	X0108.3	iF_K_D08	Vitesse_OK	op55: key D08
*55	X0108.4	iF_K_E05	CorrectionAxeZ_OK	op55: key E05
*56	X0108.5	iF_K_E06	DiamètreBuse_OK	op55: key E06
*57	X0108.6	iF_K_E07	ActivationBuselHM	op55: key E07
*58	X0108.7	iF_K_E08	BuseActuelleDeclampee	op55: key E08
*59	X0109.0	iF_K_A09	NouvelleBuseClampée	op55: key A09
*60	X0109.1	iF_K_A10	FonctionAcquittee	op55: key A10
*61	X0109.2	iF_K_A11	CodeM6	op55: key A11

Boutons pour notre grafset :

PrésenceBobine = PB	=> C05	ActivationBuselHM = ABI	=> E07
PlateauVide = PV	=> C06	nouvelleBuseClampée = NBC	=> A09
enCycle = EC	=> C07	fonctionAcquittee = FA	=> A10
repliOK = RO	=> C08	changementConsigne = CC	=> D03
ConsignePlateauOK = CPO	=> D05	CodeM6 = CM6	=> A11
ConsigneBuseOK = CBO	=> D06	diametreBuseOK = DBO	=> E06
VitesseOK = VO	=> D08	buseActuelleDeclampee = BAD	=> E08
CorrectionAxeZOK = CAZO	=> E05	PièceOK = PO	=> D07
TestAlarme = TA	=> D02		

- **BUTTONS FANUC Picture : (TEST Simulation des capteurs sous picture)**

*1251	R0600.0	BPicture_C05	Présence Bobine
*1252	R0600.1	BPicture_C06	Plateau Vide
*1253	R0600.2	BPicture_C07	En_Cycle
*1254	R0600.3	BPicture_C08	Repli_OK
*1255	R0600.4	BPicture_D05	ConsignePlateau_OK
*1256	R0600.5	BPicture_D06	ConsigneBuse_OK
*1257	R0600.6	BPicture_D07	Pièce_OK
*1258	R0600.7	BPicture_D08	Vitesse_OK
*1259	R0601.0	BPicture_E05	CorrectionAxeZ_OK
*1260	R0601.1	BPicture_E06	DiamètreBuse_OK
*1261	R0601.2	BPicture_E07	ActivationBuselHM
*1262	R0601.3	BPicture_E08	BuseActuelleDeclampee
*1263	R0601.4	BPicture_A09	NouvelleBuseClampée
*1264	R0601.5	BPicture_A10	FonctionAcquitte
*1265	R0601.6	BPicture_A11	M6_Picture
*1266	R0601.7	BPicture_D03	Chang_Conigne
*1267	R0602.0	Verifie debut Code M6	MemM6_Bis

*522	G0054.1	E_1a	Etape 1a - G54.1
*523	G0054.2	E_1b	Etape 1bi et 1c - G54.2
*524	G0054.3	E_2a	Etape 2a - G54.3
*525	G0054.4	E_3a	Etape 3a - G54.4
*526	G0054.5	E_4ai	Etape 4ai - G54.5
*527	G0054.6	E_4aII	Etape 4aII et 1c - G54.6

*530	G0055.2	E_2c	Etape 2c - G55.2
*531	G0055.3	E_3c	Etape 3c - G55.3
*532	G0055.4	E_4c	Etape 4c - G55.4
*533	G0055.5	Fin Programme	Fin Programme - RESET TOUT

*528	G0055.0	Départ Programme	Etape 0
------	---------	------------------	---------

Voici tous les G qu'on a utilisés dans le cadre du projet Fanuc Picture Ladder qui nous ont permis de l'envoyer à l'ISO.

*174	F0054.1	CR_1a	Compte Rendu Etape 1a
*175	F0054.2	CR_1b - 1c	Compte Rendu Etape 1b et 1c
*176	F0054.3	CR_2a	Compte Rendu Etape 2a
*177	F0054.4	CR_3a	Compte Rendu Etape 3a
*178	F0054.5	CR_4ai	Compte Rendu Etape 4ai
*179	F0054.6	CR_4aII - 1c	

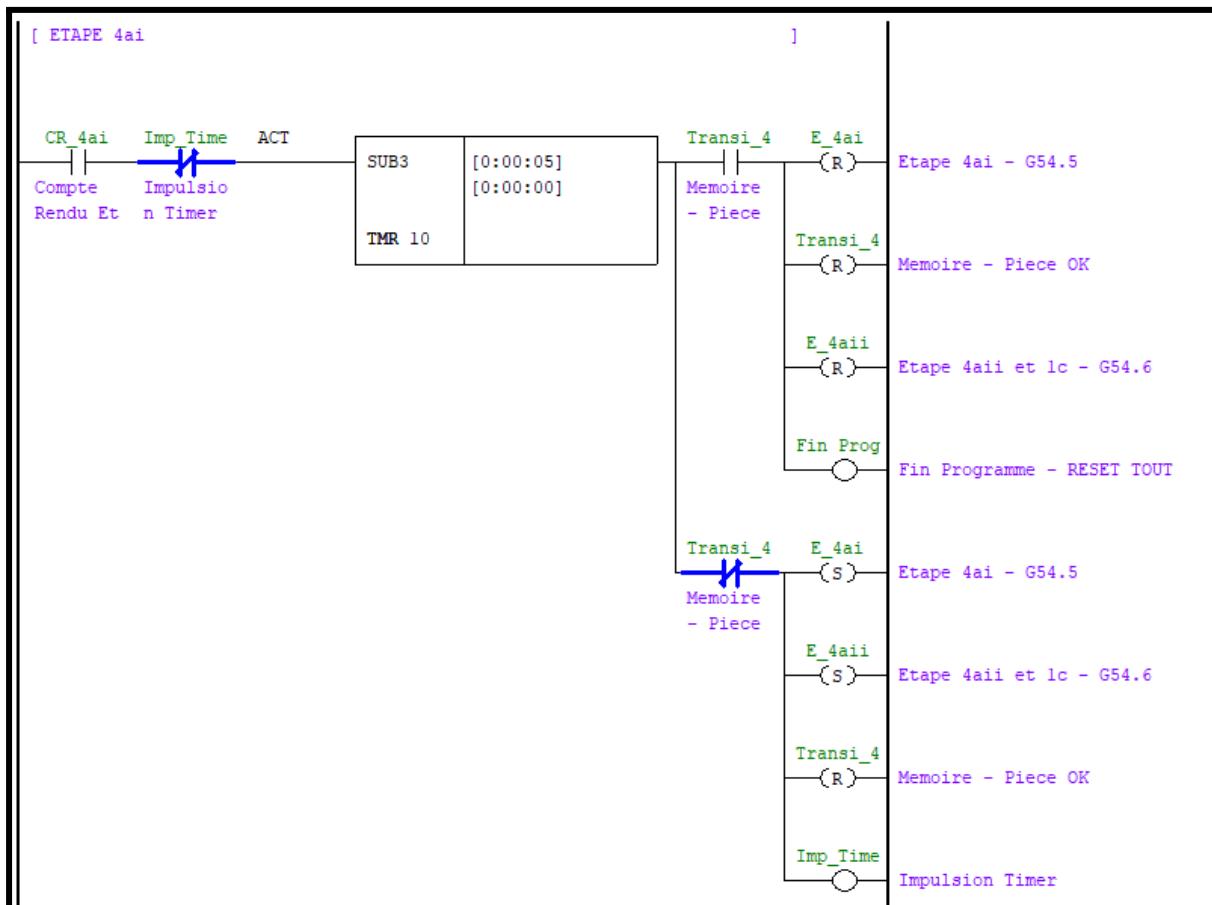
*180	F0055.2	CR_2c	Compte Rendu 2c
*181	F0055.3	CR_3c	Compte Rendu 3c
*182	F0055.4	CR_4c	Compte Rendu 4c

Voici tous les F qu'on a utilisés dans le cadre du projet Fanuc Picture Ladder qui nous permis de les récupérer de l'ISO.

- **TIMER :**

10	T18	00H_00M_05S	1SEC
11	T20	00H_00M_05S	1SEC
12	T22	00H_00M_05S	1SEC

Exemple moment d'utilisation :



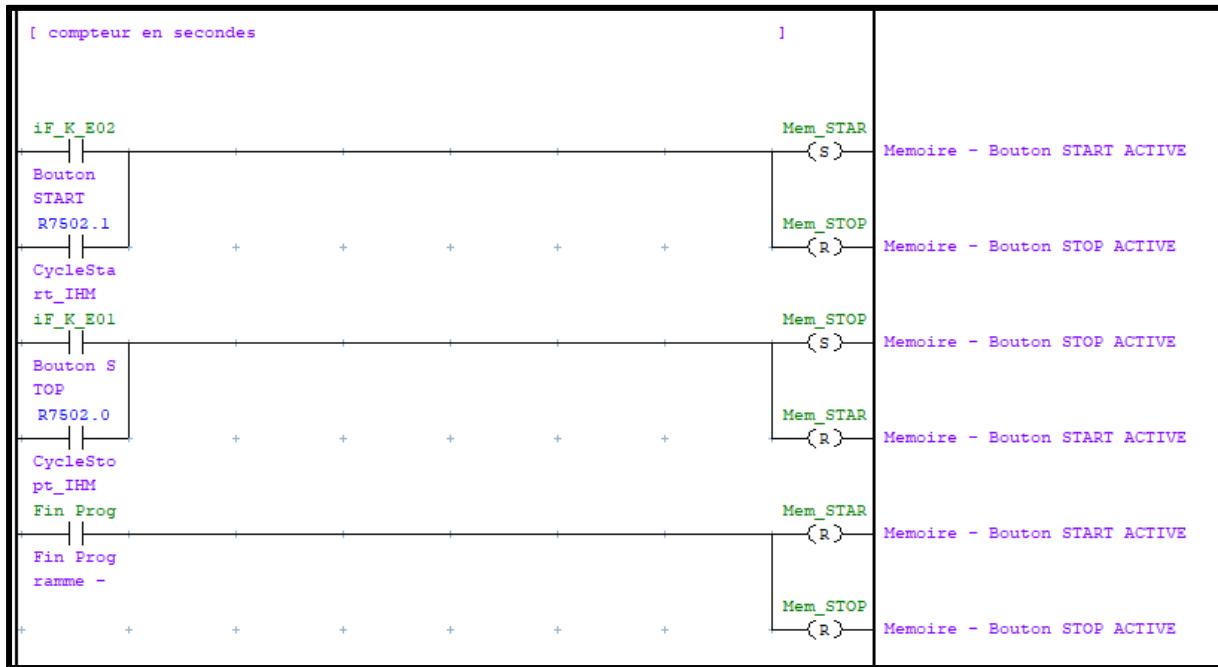
Utiliser pour les étapes **4ai - 4aii_4c** et **1b_4c**.

*1960	R7502.0	CycleStop_IHM
*1961	R7502.1	CycleStart_IHM

*36	X0106.0	iF_K_E01	Bouton STOP	op55: key E01
*37	X0106.1	iF_K_E02	Bouton START	op55: key E02

Ces timers nous permettent de pouvoir nous laisser le temps de pouvoir appuyer où non sur les boutons afin de simuler les capteurs, finissant ainsi le programme ou de recommencer la boucle.

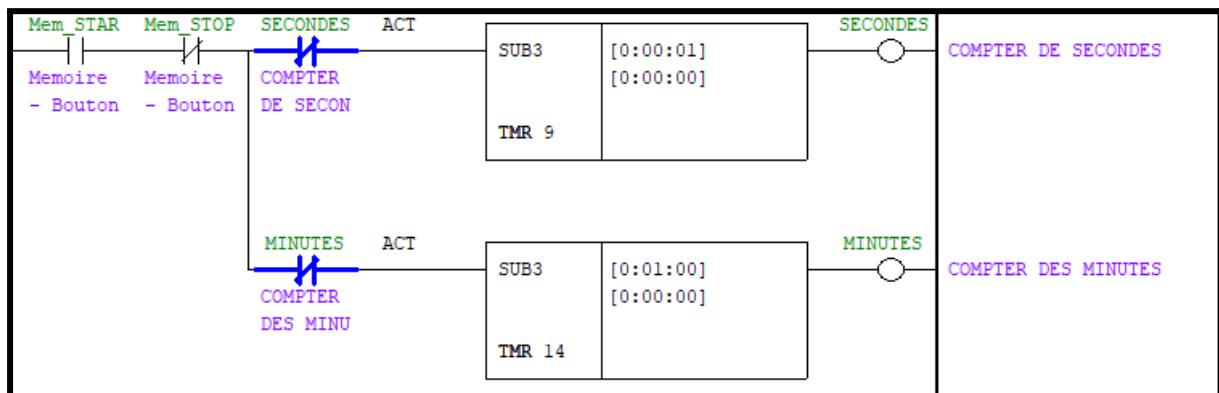
- Timer pour départ Programme :



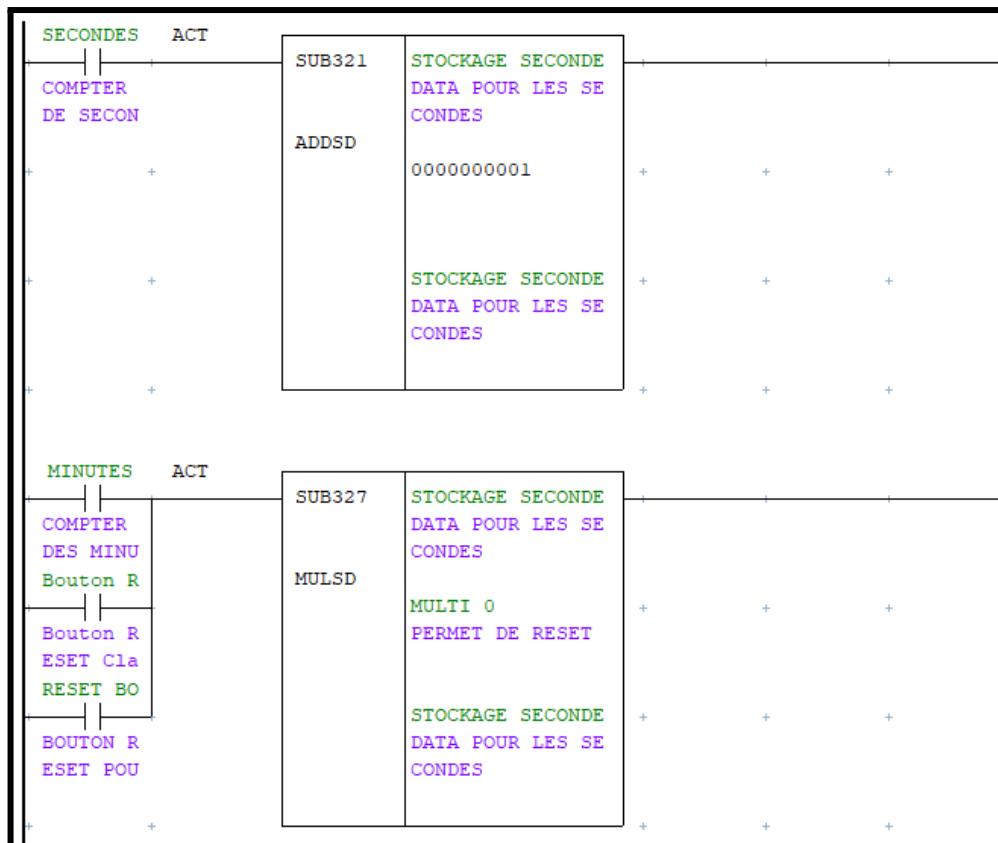
Pour commencer le programme, soit un appuie sur le bouton Start **E02** sur le clavier soit dans Fanuc Picture en envoyant un **R7502.1** pour lancer le cycle de l'IHM.

Une fois exécuté, ça va **Set** la mémoire Bouton Start Active et donc **reset** la mémoire du Bouton Stop Active.

Puis, pour arrêter le programme il faudra appuyer sur le bouton **E01** Stop ou dans Fanuc Picture le **R7502.0** qui va **Set** la mémoire Bouton Stop Active et reset la mémoire Bouton Start Active.

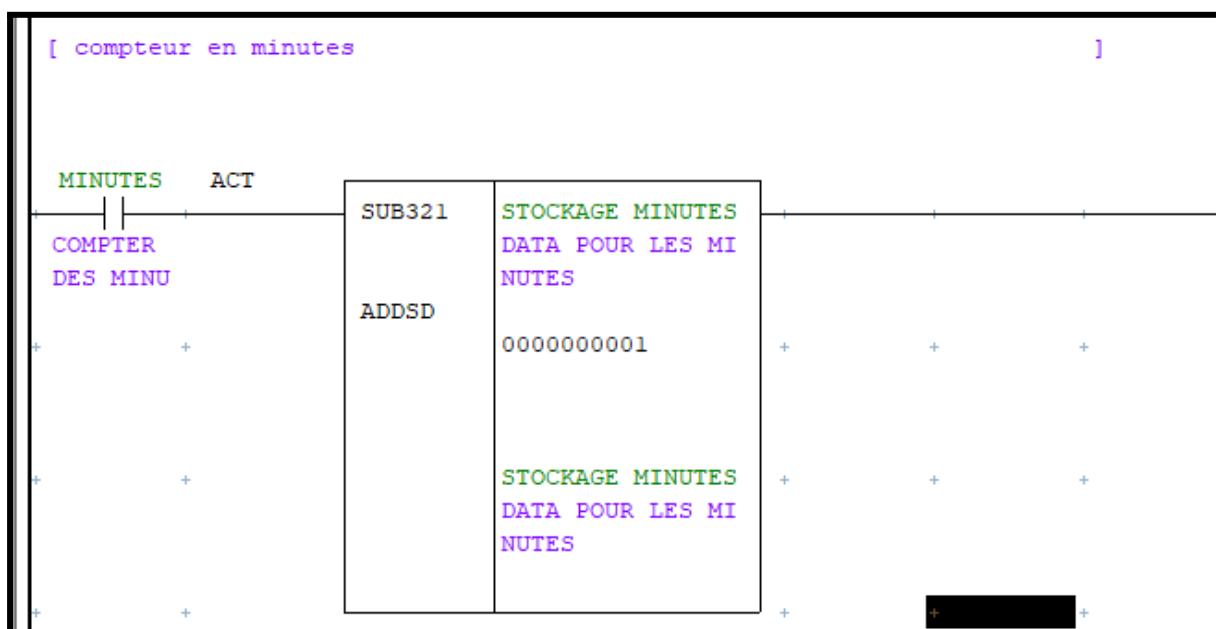


Nous sommes dans le compteur au début du programme, quand le bouton Start a été activé et que le bouton Stop n'a pas été déclenché, ça va déclencher un timer qui va compter les secondes et les minutes.

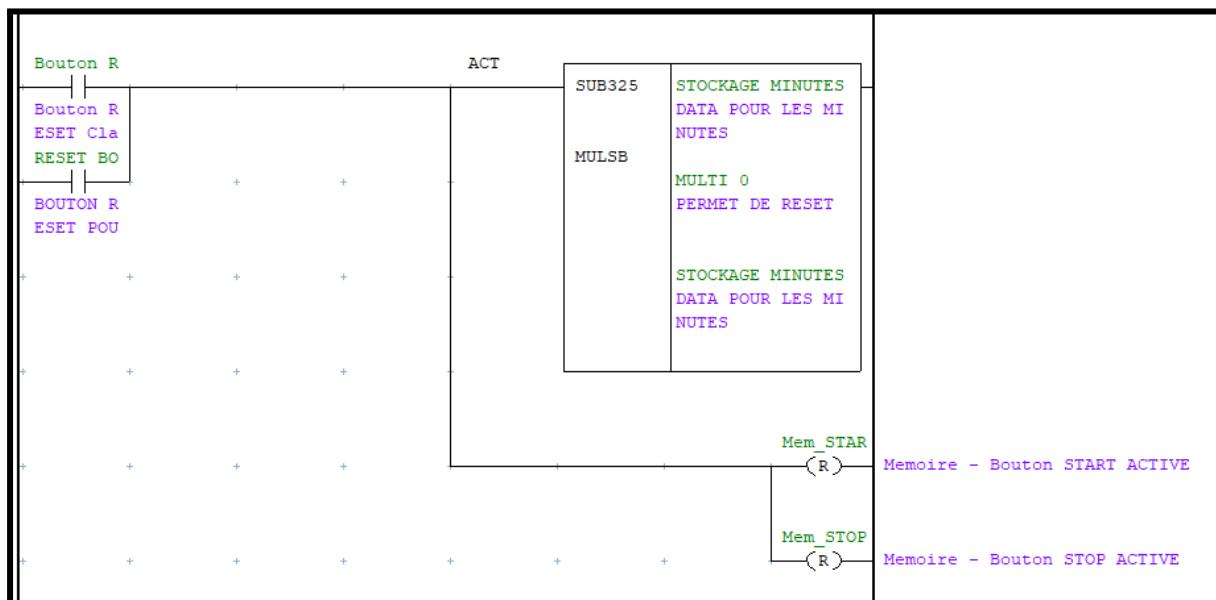


Nous voici dans le timer, donc pour faire fonctionner le timer des secondes on va utiliser le bloc **SUB321** pour ajouter, ici de 1, les secondes qui seront stockées .

Maintenant si on veut tout reset (les minutes et les secondes) on va appuyer sur le bouton **D01**, donc pour effectuer ce reset on va prendre les secondes qui sont stockées et on va les multiplier par 0 pour que le résultat sera tout le temps égal à 0 peu importe le nombre de de secondes. Ça se passe dans le bloc "**SUB 327**".



Dans le compteur pour les minutes ça se passe pour que le timer des secondes atteint les 60s, ça va ajouter 1 au compteur minute avec le bloc Add **SUB321**.



Quand on appuie soit sur le bouton **Reset** du clavier sur **D01** ou sur le Fanuc Picture pour l'IHM ca va reset le compteur des minutes en prenant les minutes stockées dans la base de données le multiplier à 0 et on va re stocker la valeur 0 pour reset les minutes. Puis, cela va ensuite en même temps, reset la mémoire du Bouton Start Active et le Bouton Stop Actif.

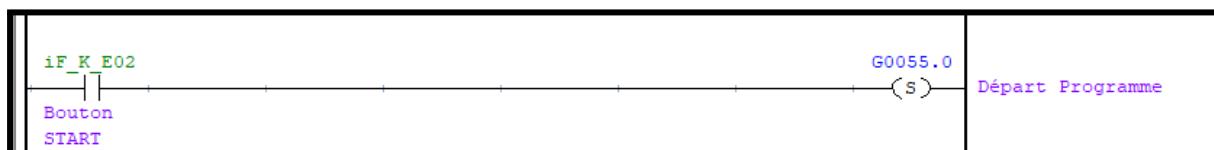
Transitions des grafcets

- Sous-Programme - IUT EVRY BUT3



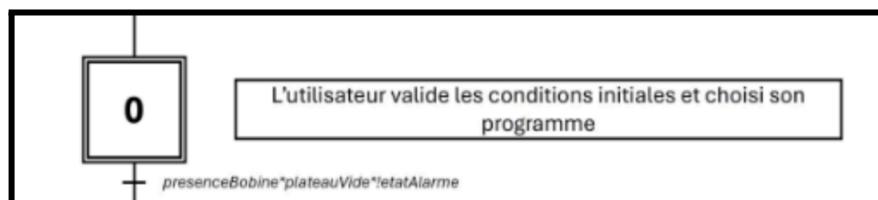
Voici le sous programme qu'on a créé pour réaliser le Ladder du concours FANUC.

- Avant - Etape 0 :

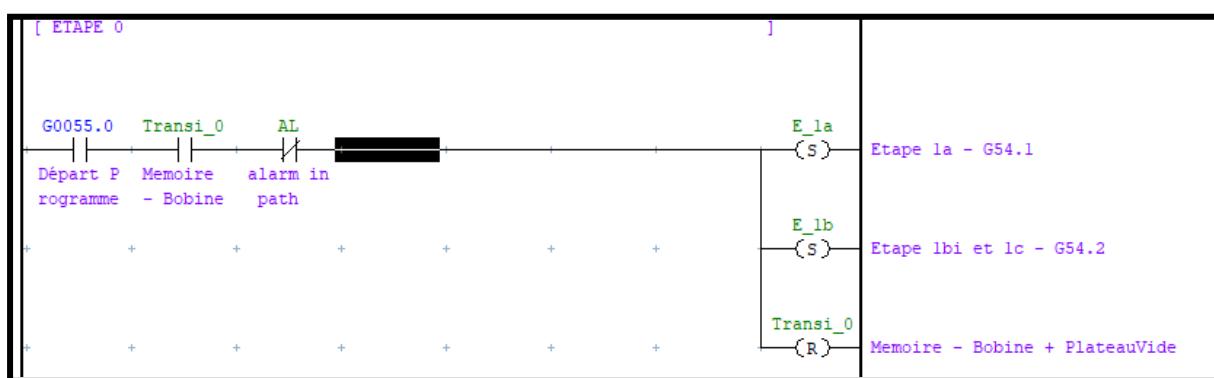


Le premier bouton du programme va permettre de commencer le programme et cela va envoyer le **G0055.0** à l'ISO.

- Après - Etape 0 :



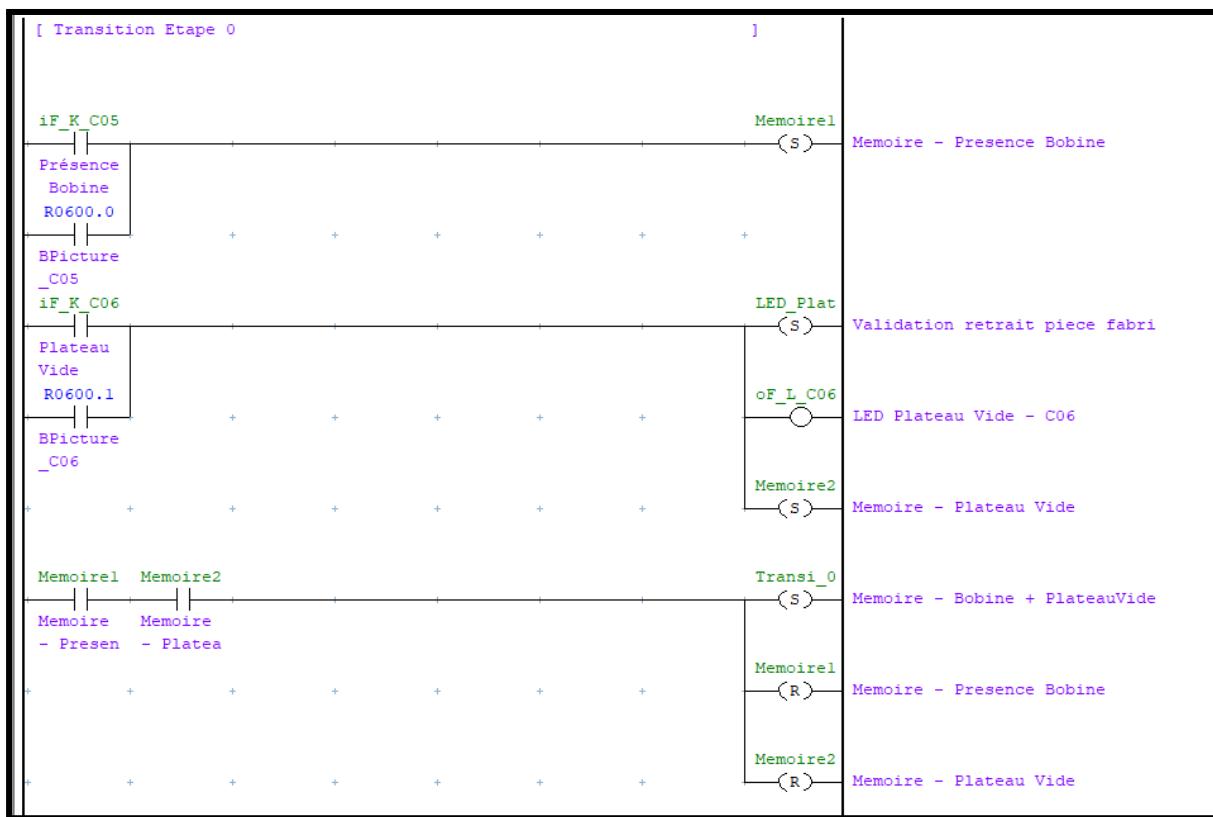
Grafcet :



Quand le programme est lancé, le G de start est récupéré et on va tester les transitions qui suivent.

Une fois que ça a été testé on va passer à l'étape 1 donc le envoyer le **G54.1** à l'ISO, envoyer un **G54.2** pour l'étape 1b/4aii et pour la 1c. Puis, nous allons reset la mémoire Bobine + Plateau Vide pour l'arrêter.

Mémoires :

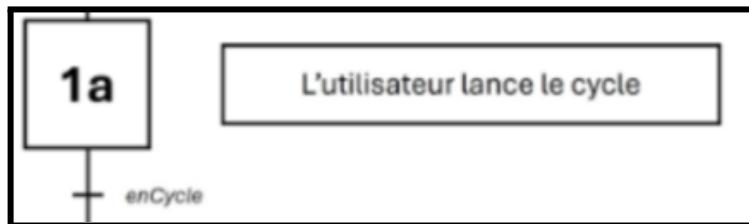


Voici ce qui se passe dans la transition de l'étape 0, pour commencer on envoie un R au picture pour dire qu'il y a une présence bobine dans la machine. Ensuite, soit on passe par Fanuc Picture en envoyant un R pour savoir si le plateau est vide ou alors on appuie sur le clavier bouton **C06**.

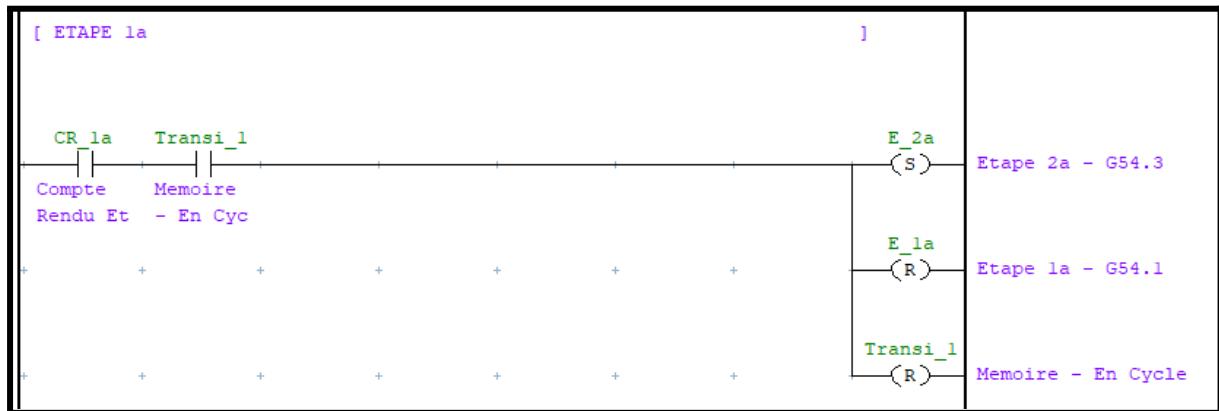
Si le plateau est vide, la led du bouton **C06** va s'allumer pour prévenir et on Set Mémoire-plateau vide pour poursuivre le programme. S'il y a un bien une bobine et que le plateau est bien vide, on va Set la Mémoire Bobine+Plateau Vide passer à l'étape 1.

Puis, pour finir on va Reset les deux mémoires Bobine et plateau vide pour ne laisser que la mémoire Bobine+Plateau Vide.

- Après - Etape 1a :



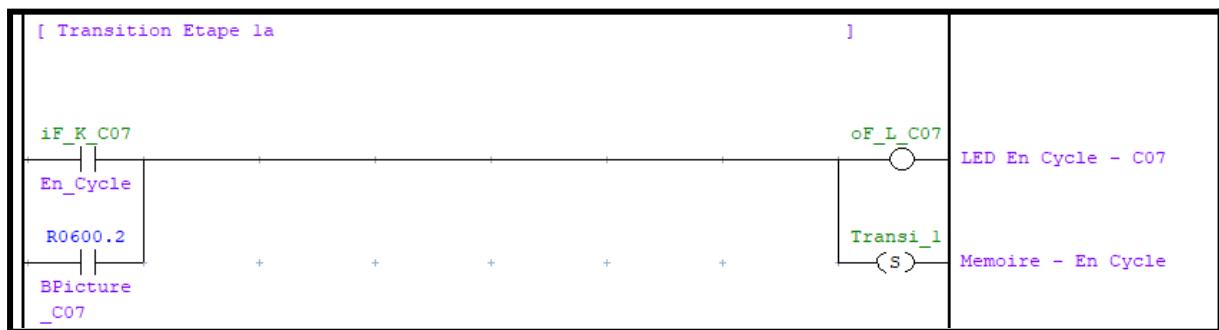
Grafcet :



Pour finir l'étape 0 on va vérifier le si on a bien récupéré le F, si on l'a bien reçu on passe à l'étape suivante donc commencer l'étape 1a en mettant une entrée mémoire pour la transition.

Quand toute l'étape de la transition à été faite on envoie le **G54.3** à l'ISO et on va reset le **G53.1** et la transition mémoire machine En cycle.

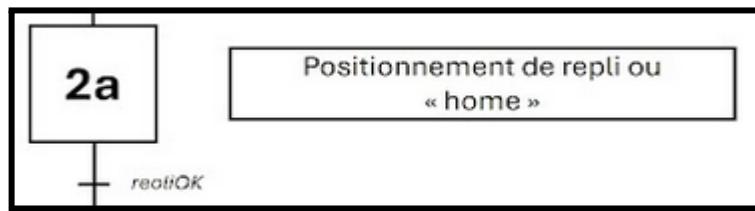
Mémoires :



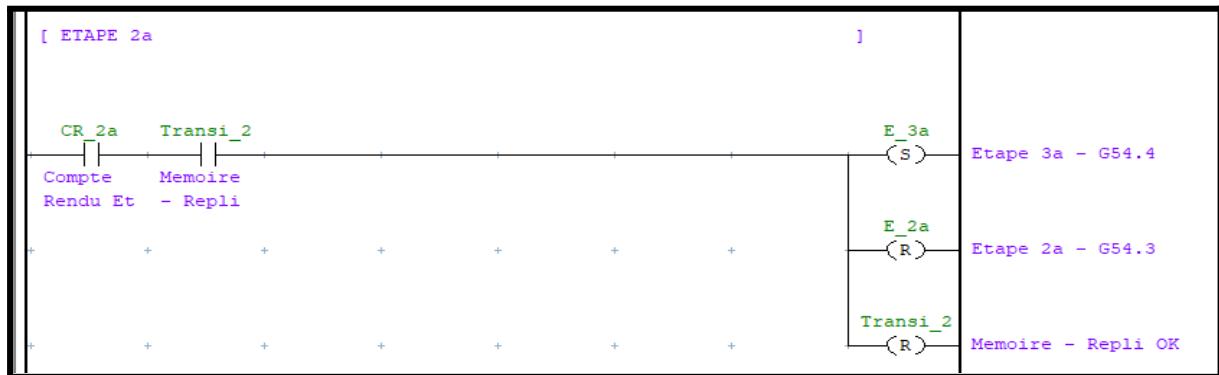
Nous sommes dans la partie mémoire de la transition donc pour valider l'étape on passe soit par Fanuc Picture en envoyant un R pour savoir si c'est en cycle ou alors on appuie sur le clavier bouton **C07**.

Si c'est en cycle, la led du bouton **C07** va s'allumer pour prévenir que la machine est en cycle et on va Set le mémoire pour passer à la prochaine étape.

- Après - Etape 2a :



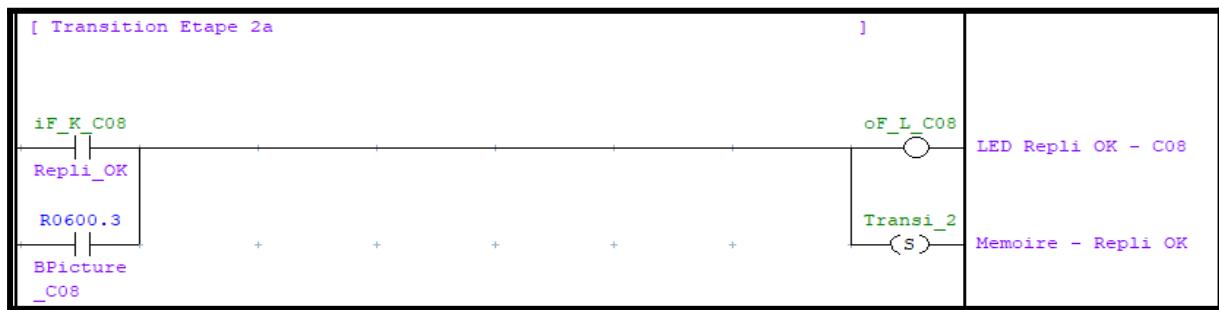
Grafcet :



Pour finir l'étape 1a on va vérifier le si on a bien récupéré le F, si on l'a bien reçu on passe à l'étape suivante donc commencer l'étape 2a en mettant une entrée mémoire Repli pour la transition.

Quand toute l'étape de la transition à été faite on envoie le **G54.4** à l'ISO et on va reset le G54.3 et la transition mémoire Repli OK.

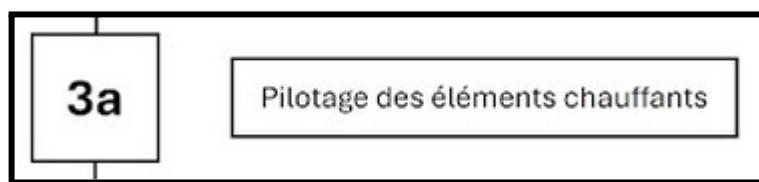
Mémoires :



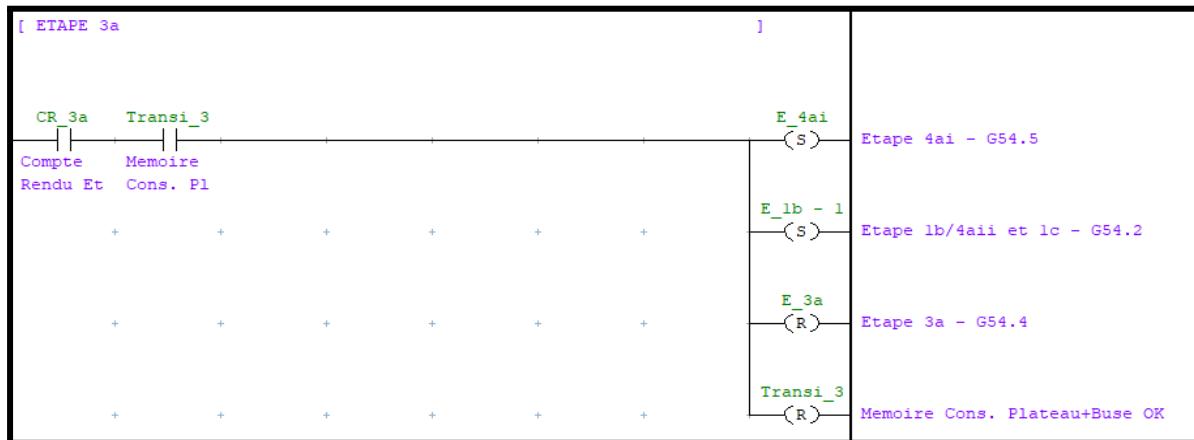
Nous sommes dans la partie mémoire de la transition Repli OK, pour valider l'étape on passe soit par Fanuc Picture en envoyant le **R600.3** pour savoir si ce repli est bien OK ou alors on appuie sur le clavier bouton **C08**.

Si le repli s'est bien effectué, la led du bouton **C08** va s'allumer pour prévenir que le repli a bien été fait, et on va Set la mémoire pour passer à la prochaine étape qui va être d'envoyer le G.

- Après - Etape 3a :



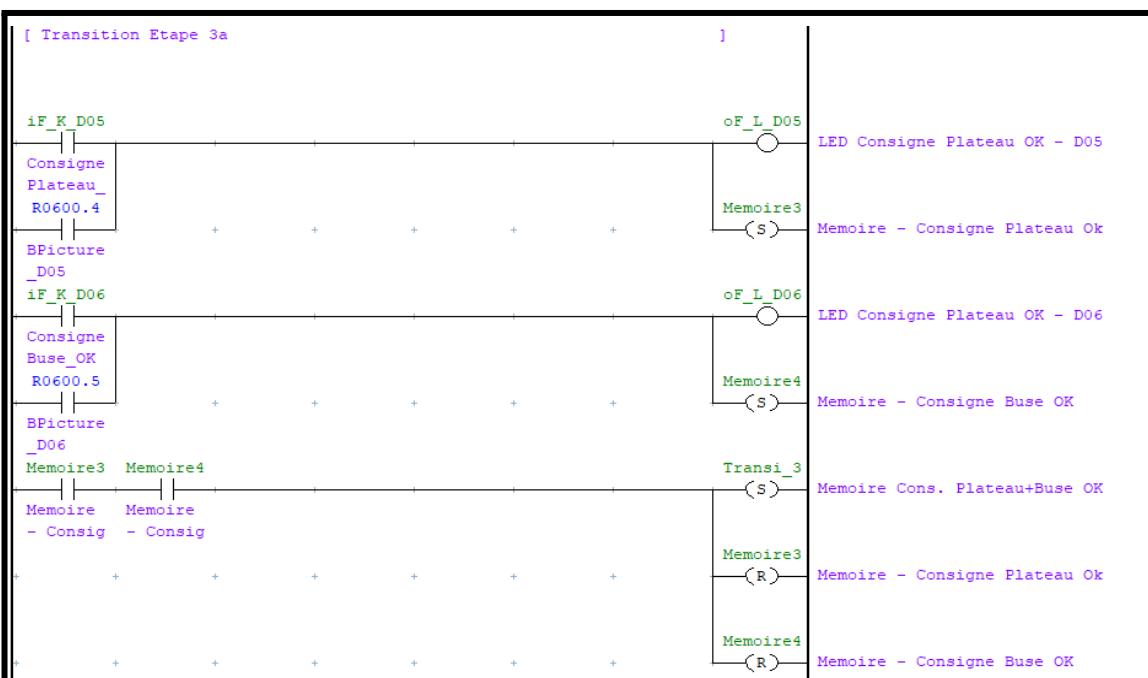
Grafcet :



Pour finir l'étape 3a on va vérifier si on a bien récupéré le F, si on l'a bien reçu on passe à l'étape suivante donc commencer l'étape 3a en mettant une entrée mémoire Cons. P1 pour la transition.

Quand toute l'étape de la transition à été faite on envoie le **G54.5** à l'ISO pour l'étape 4ai et le **G54.2** pour l'étape 1b/4aii et 1c, et on va reset le **G54.4** et la transition mémoire Cons. Plateau OK.

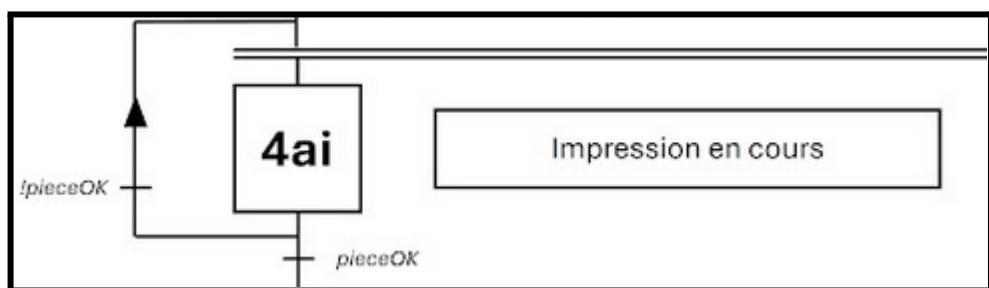
Mémoires :



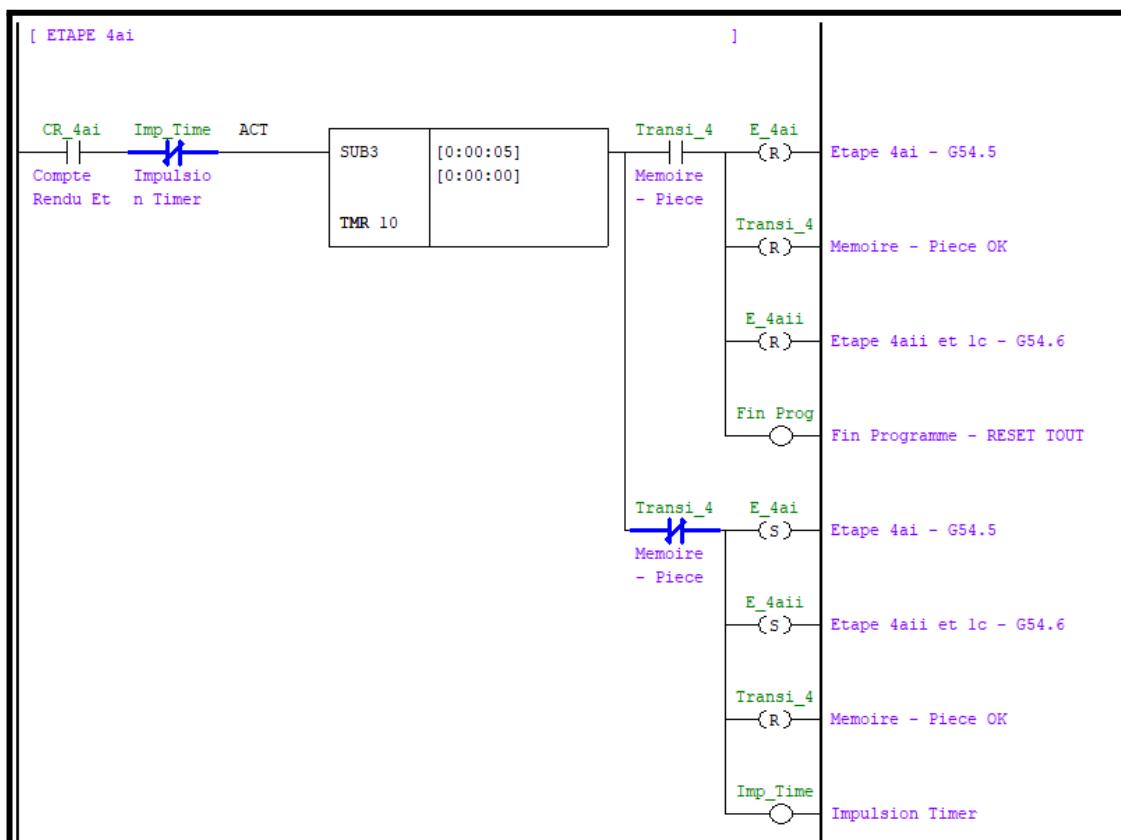
Nous sommes dans la partie mémoire de la transition Cons. Plateau OK, pour valider l'étape on passe soit par Fanuc Picture en envoyant le **R600.4** pour savoir si la Consigne et le Plateau est bien OK ou alors on appuie sur le clavier bouton **D06**.

Si la Consigne et le plateau sont OK, la led du bouton **D05** va s'allumer pour nous prévenir., et on va Set la mémoire pour passer à la prochaine étape qui va être d'envoyer le G.

- Après - Etape 4ai :



Grafcet :

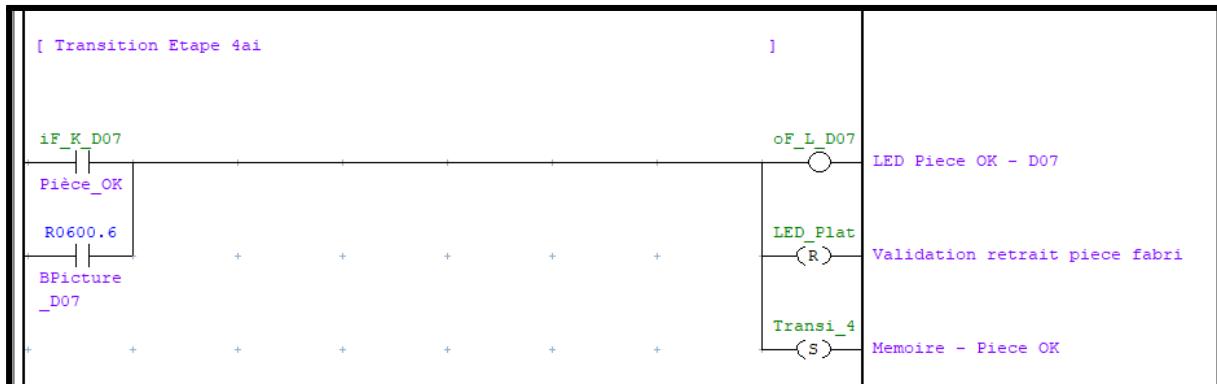


Pour valider l'étape, il faut vérifier qu'on a bien reçu le R, une fois que c'est fait, on a mis un timer de 5 secondes pour laisser le temps d'appuyer sur le bouton, une fois la durée écoulée ça va déclencher la mémoire Pièce OK.

Si la pièce a fini d'être imprimée ça voudra dire que la condition de la mémoire Pièce OK a été remplie et qu'on va réinitialiser l'étape 4ai, la mémoire Pièce OK, l'étape 4aïi et 1c et ça annonce la fin du programme donc de tout réinitialiser.

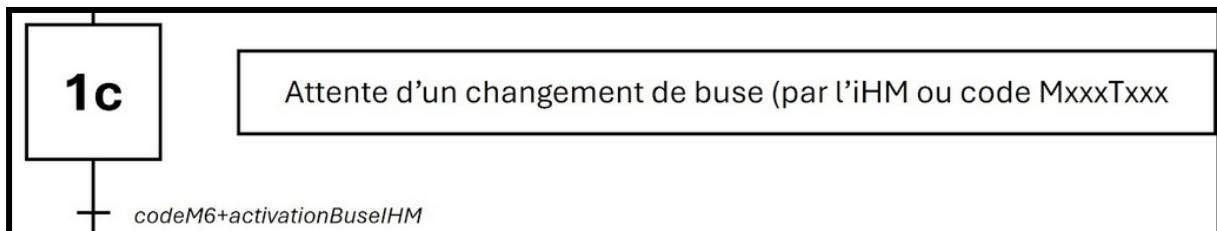
Si la pièce n'a pas fini d'être imprimée, ça voudra dire que la condition de la mémoire Pièce OK n'a pas été remplie et qu'on va activer l'étape 4ai en envoyant un G54.5 à l'ISO, l'étape 4aïi et 1c en envoyant un G54.6 à l'ISO. Pour finir ça réinitialise le mémoire pièce OK pour supprimer l'étape d'avant pour y revenir après tant que la pièce n'a pas fini d'être imprimée.

Mémoires :

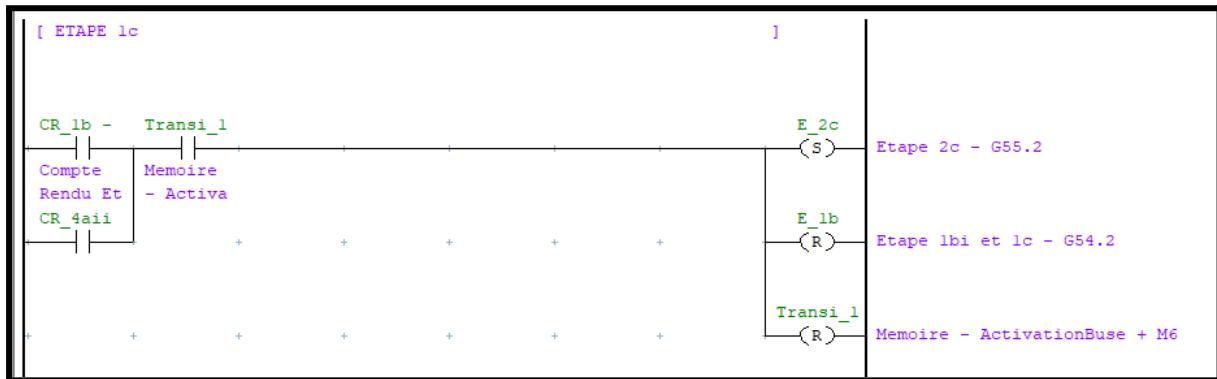


Dans la transition de l'étape 4ai, si on appuie soit sur le bouton D07 du clavier ou en envoyant un R600.6 dans le fanuc picture ça va allumer la led du bouton, réinitialiser la validation du retrait pièce fabriquée et d'initialiser la mémoire pièce OK.

- Pour Etape 4aii et 1b - 1c :



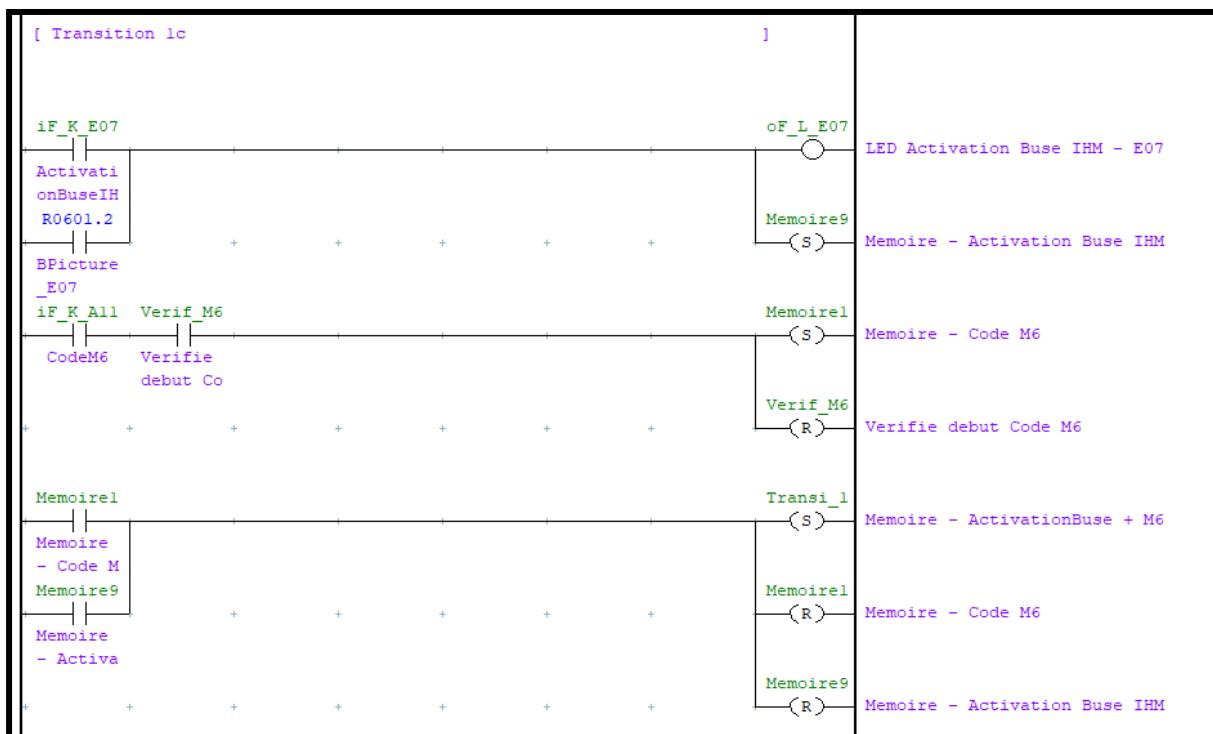
Grafcet :



Pour finir l'étape 1b, on vérifie si on a bien récupéré le R. Si c'est fait, on passe à l'étape suivante qui est la transition mémoire Activation Buse IHM + Code M6.

Une fois que c'est terminée, on va donner le G55.2 à l'ISO et reset l'étape précédente 1b/4aai et 1c, et la mémoire de la transition Activation Buse IHM + Code M6.

Mémoires :

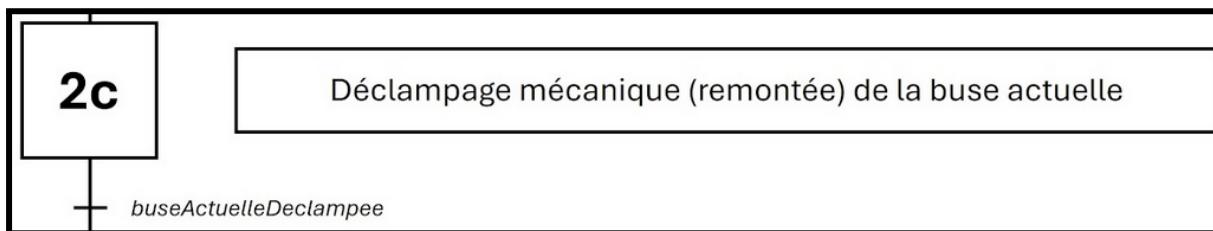


Nous sommes dans la partie mémoire de la transition 1c qui est divisé en 3 parties, pour valider la première partie de l'étape, on va passer soit par Fanuc Picture en envoyant le **R601.2** pour savoir si ça à bien activé la Buse IHM ou alors on appuie sur le clavier bouton **E07**.

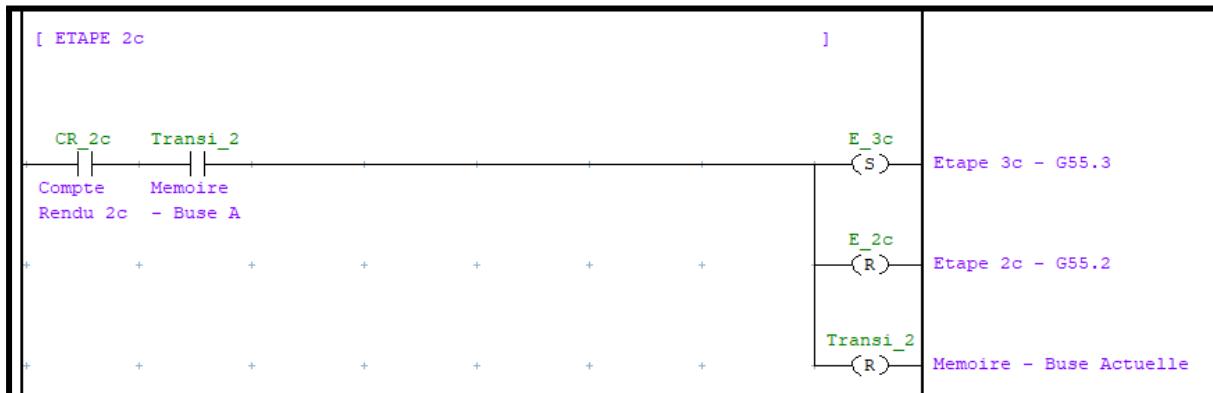
Si la Buse IHM est activée, la led du bouton **D07** va s'allumer pour nous prévenir, et ça va la mettre en mémoire pour passer à la deuxième étape qui va être d'appuyer sur le bouton A11 pour dire que le sous-programme code M6 est terminé et ensuite ça va vérifié si le sous programme du Code M6 est bien terminé pour qu'on puisse la mettre en mémoire pour la prochaine étape et qu'on reset Début Code M6.

Et pour finir, on met les 2 mémoires en forme de OU pour dire que si l'un des deux est exécuté ça va créer une mémoire ActivationBuse+M6 ça va reset les deux mémoires précédentes pour passer à la prochaine transition.

- Après - Etape 2c :



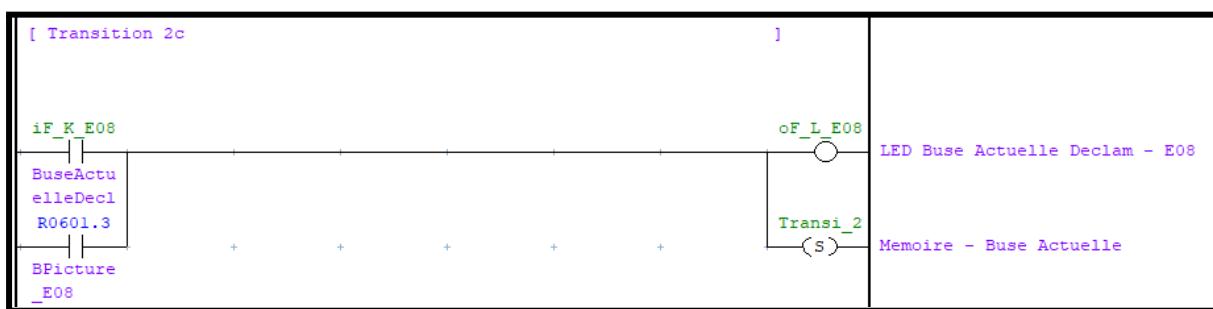
Grafcet :



Pour finir l'étape 1c, on vérifie si on a bien récupéré le R. Si c'est fait, on passe à l'étape suivante qui est la transition mémoire Buse Actuelle.

Une fois que c'est terminé, on va donner le **G55.3** à l'ISO et reset l'étape précédente 1c, et la mémoire de la transition Buse Actuelle.

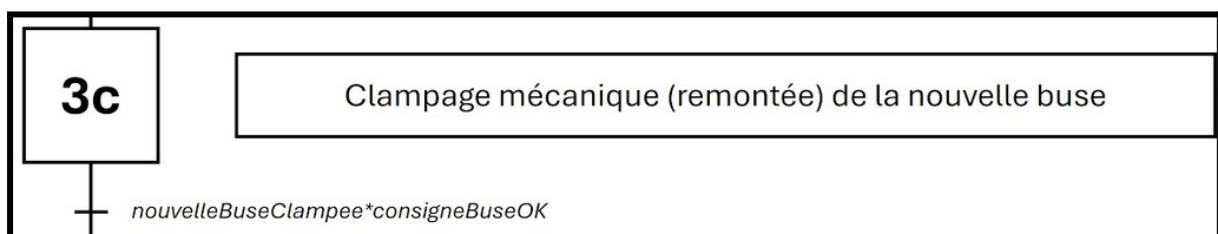
Mémoires :



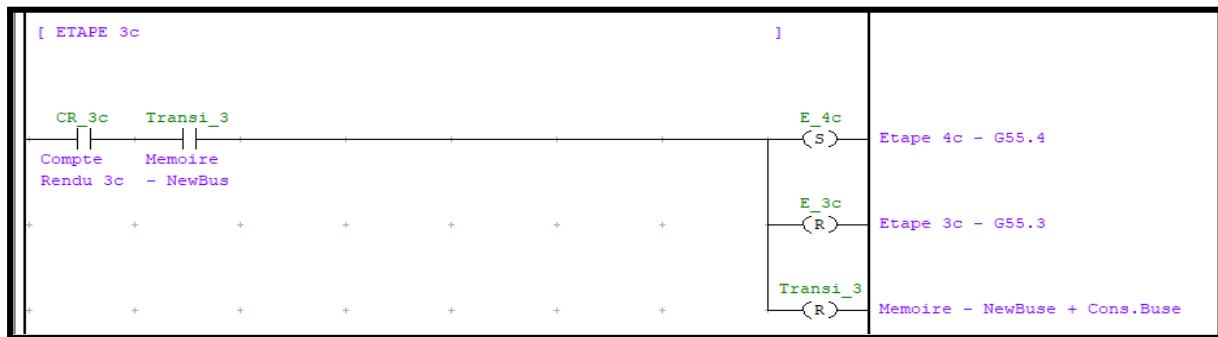
Nous sommes dans la transition 2c, quand la buse actuelle est déclamée, soit on envoie un **R601.3** sur le Fanuc Picture soit on clique sur le bouton **E08** du clavier virtuel.

Si la buse actuelle est déclamée, la Led du bouton va s'allumer pour nous prévenir et on va en même temps créer la mémoire Buse Actuelle.

- Après - Etape 3c :

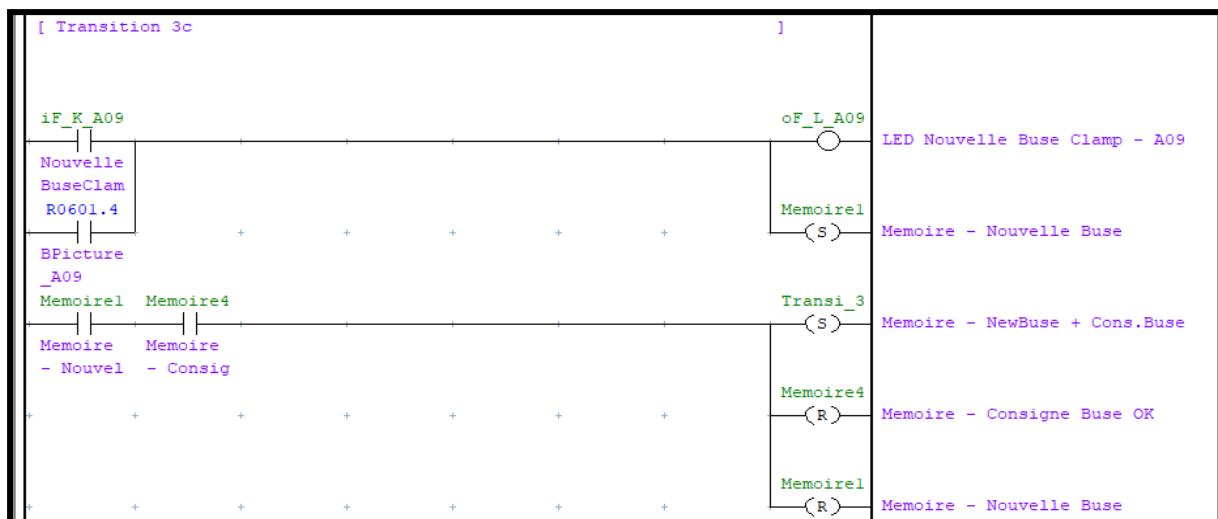


Grafcet :



Pour finir l'étape 2c, on vérifie si on a bien récupéré le R. Si c'est fait, on passe à l'étape suivante qui est la transition mémoire NewBus. Une fois que c'est terminé, on va donner le **G55.4** à l'ISO et reset l'étape précédente 2c, et la mémoire de la transition NewBus.

Mémoires :

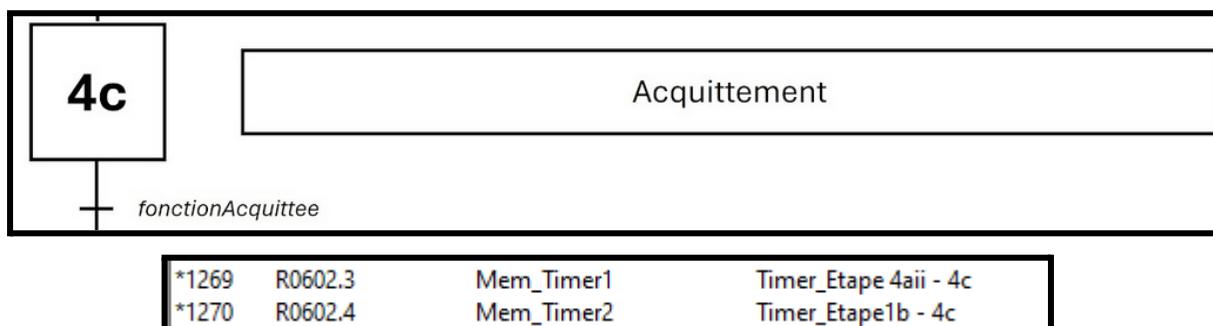


Nous sommes dans la partie mémoire de la transition 3c qui est divisé en 2 parties, pour valider la première partie de l'étape, on va passer soit par Fanuc Picture en envoyant le R601.4 pour savoir si la nouvelle Buse a bien été Clampée ou alors on appuie sur le clavier bouton A09.

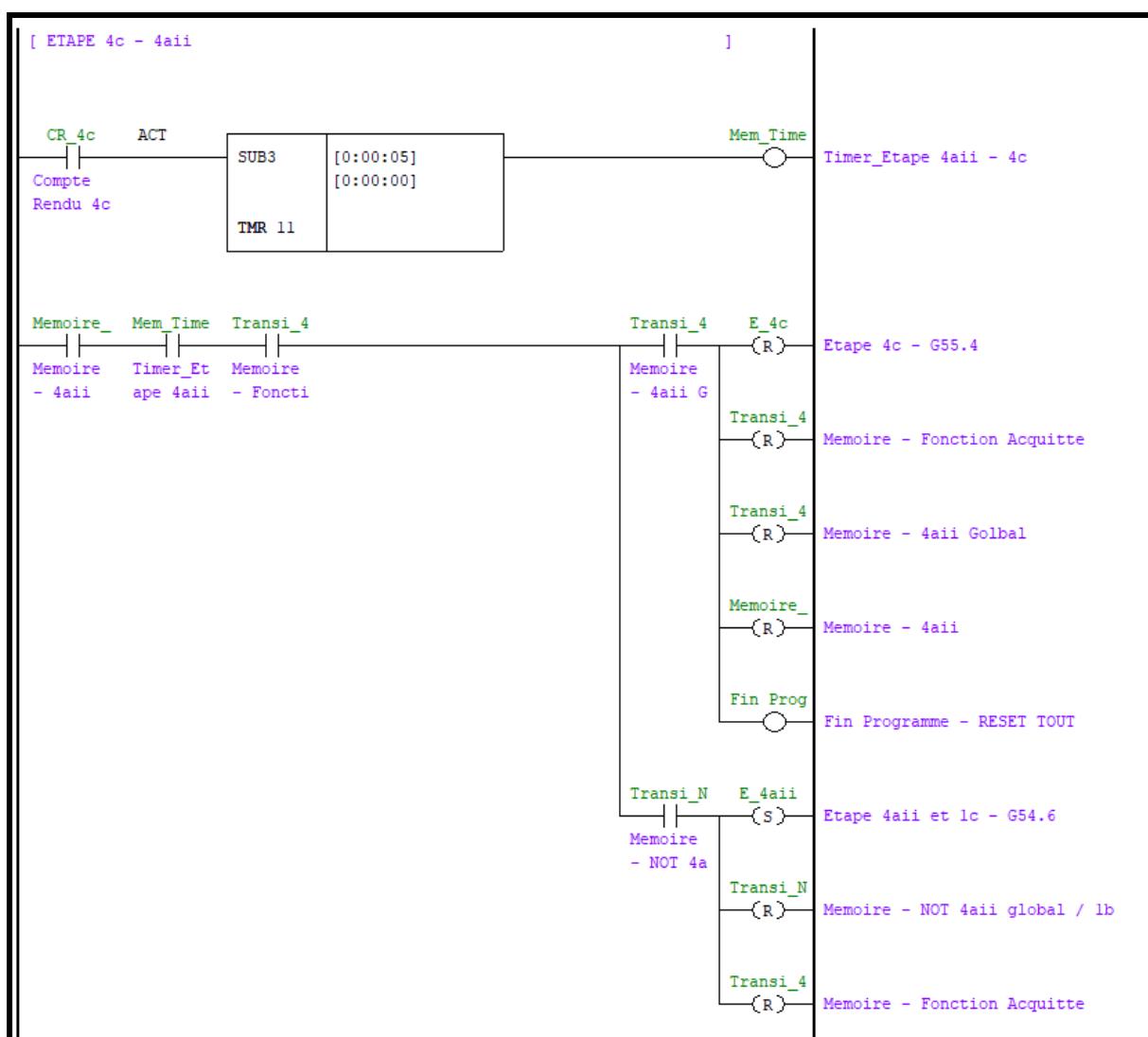
Si la nouvelle Buse est clampée , la led du bouton A09 va s'allumer pour nous prévenir et on va créer la mémoire Nouvelle Buse.

Ensuite on va réunir la mémoire qu'on a créé et ajouter la mémoire qu'on avait créé plus tôt dans le Ladder qui est la mémoire Consigne Buse OK, si les deux sont exécutées ça va créer une mémoire NewBuse + Cons.Buse et on reset les deux mémoires précédentes pour passer à la prochaine transition.

- Après - Etape 4c :



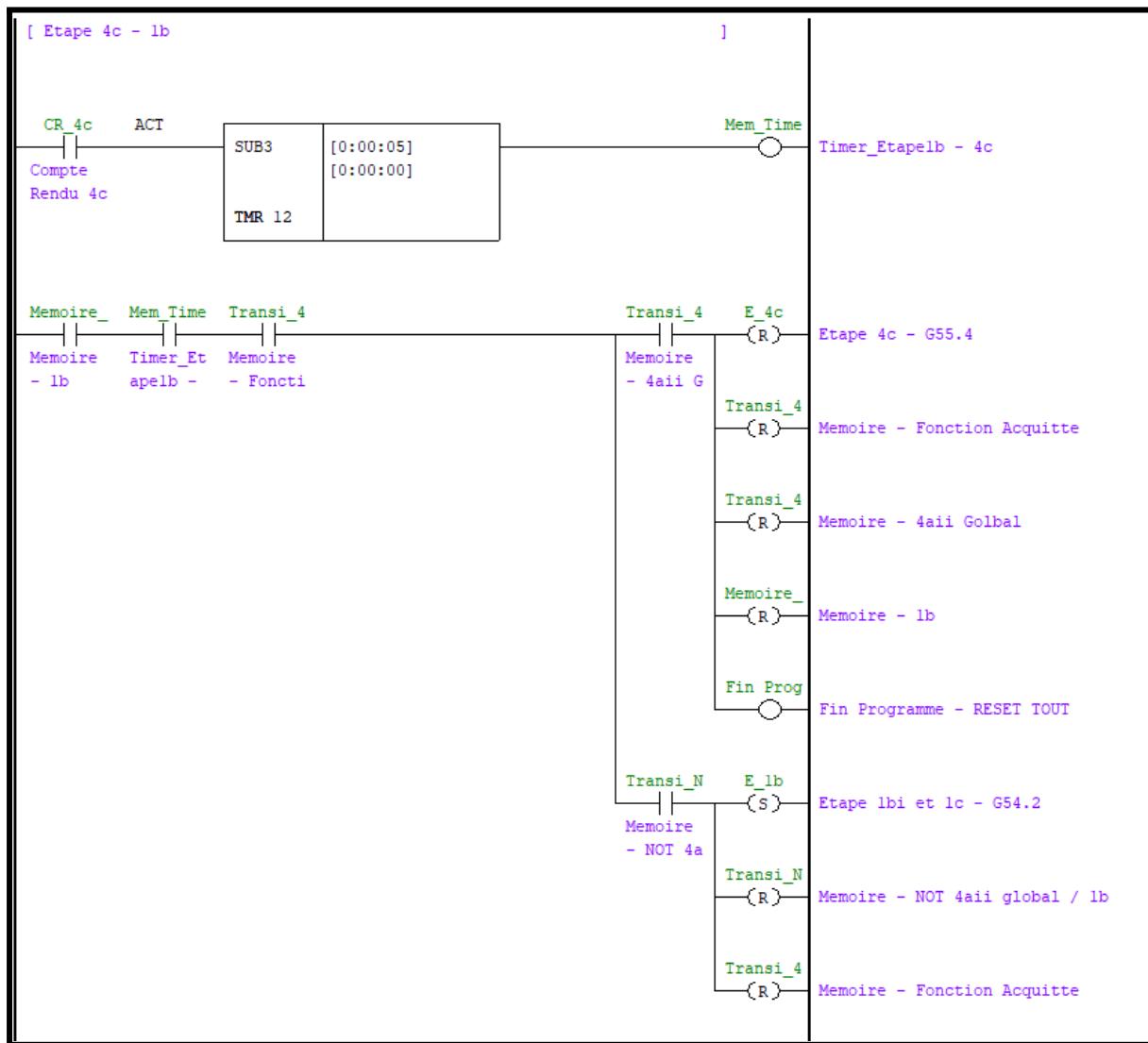
Grafcet :



Pour valider l'étape, il faut vérifier qu'on a bien reçu le R, une fois que c'est fait, on a mis un timer de 5 secondes pour laisser le temps d'appuyer sur le bouton, une fois la durée écoulée ça va déclencher la mémoire Timer_Etape 4aii-4c.

Ensuite si la mémoire 4-aii et le timer_etape 4aii sont activées, cela va activer la transition Mémoire-Fonction Acquitté. Quand les conditions de la transition Mémoire 4aii G sont remplies, ça va réinitialiser l'étape 4c, mémoire Fonction Acquitté, mémoire 4aii Global, mémoire 4aii et ça va mettre fin au programme.

Si les conditions de la transition Mémoire Not 4a sont remplies, ca va activer l'étape 4aii et 1c en envoyant un **G54.6** à l'ISO, et réinitialiser les deux mémoires (Not 4aii global/1b et Fonction Acquitte).

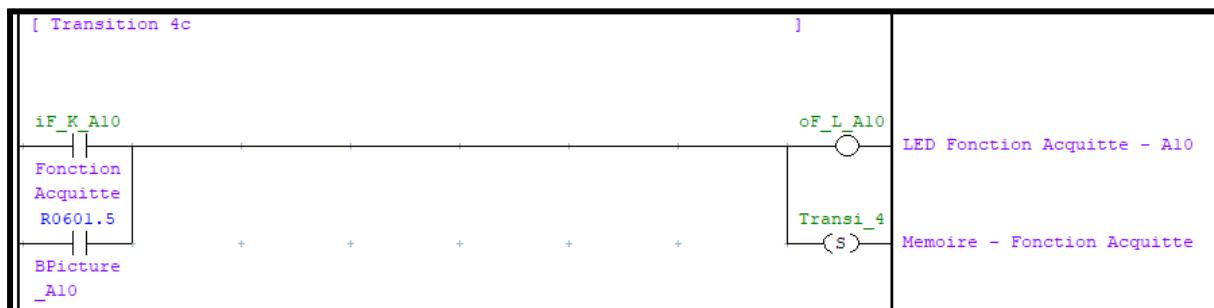


Pour valider l'étape, il faut vérifier qu'on a bien reçu le R, une fois que c'est fait, on a mis un timer de 5 secondes pour laisser le temps d'appuyer sur le bouton, une fois la durée écoulée ça va déclencher la mémoire Timer_Etape1b-4c. Ensuite, si la mémoire-1b et le Timer_etape 1b sont activées, cela va activer la transition Mémoire-Fonction Acquitée.

Quand les conditions de la transition Mémoire 4aii G sont remplies, ça va réinitialiser l'étape 4c, mémoire Fonction Acquitté, mémoire 4aii Global, mémoire 1b et ça va mettre fin au programme.

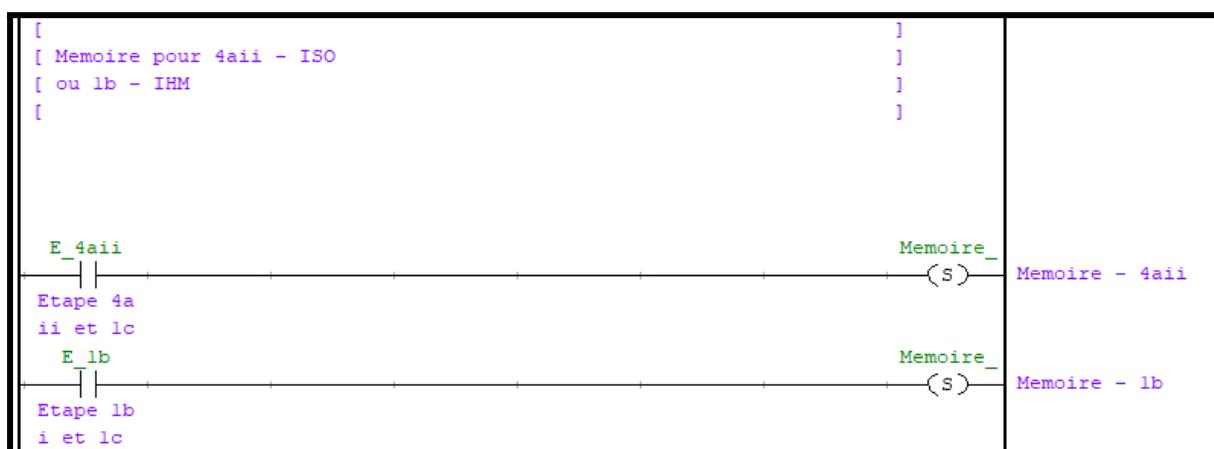
Si les conditions de la transition Mémoire Not 4a sont remplies, ça va activer l'étape 1bi et 1c en envoyant un **G54.2** à l'ISO, et réinitialiser les deux mémoires (Not 4aii global/1b et Fonction Acquitte).

Mémoires :



Nous sommes dans la transition 4c, pour activer cette fonction on va appuyer soit sur le bouton du clavier A10 ou dans le Fanuc Picture en envoyant un R601.5.

Une fois que c'est exécuté, ça va allumer une led au-dessus du bouton du clavier pour prévenir que la Fonction Acquittée à bien été remplie et on va activer la mémoire Fonction Acquittée.



Nous sommes dans la fonction Mémoire 4aii ISO/1b-IHM, pour activer la mémoire 4aii, il faut que l'étape(4aii et 1c) soit validé, et pour que la mémoire 1b soit activer il faut que l'étape (1bi et 1c) soit validée.

- Après - Etape 4aii et 1b :



Nous sommes dans la transition 4aii et 1b, donc pour activer la Mémoire Vitesse OK, il faut qu'on appuie soit sur le bouton D08 du clavier soit en envoyant le **R600.7** dans l'ISO, une fois appuyé, ca va allumer une led au dessus du bouton pour prévenir que l'étape a bien été effectuée.

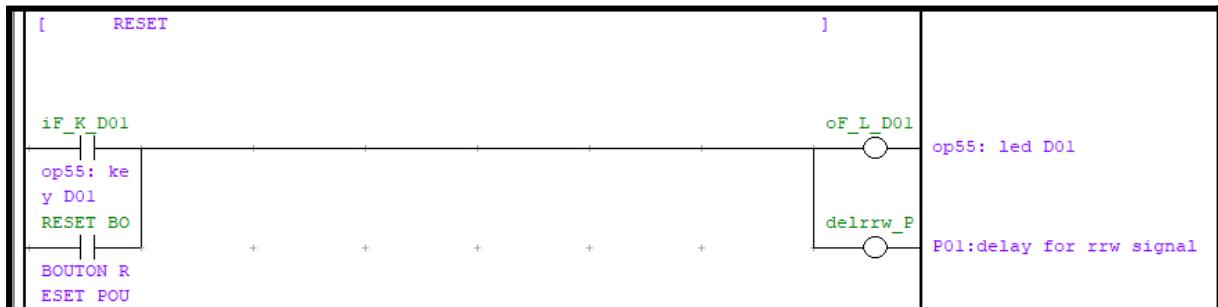
Ensuite, pour activer la Mémoire Correction Axe Z OK , il faut qu'on appuie soit sur le bouton E05 du clavier soit en envoyant le **R601.0** dans l'ISO, une fois appuyé, ca va allumer une led au dessus du bouton pour prévenir que l'étape a bien été effectuée.

Dans la troisième partie, pour activer la Mémoire Diamètre Buse OK , il faut qu'on appuie soit sur le bouton E06 du clavier soit en envoyant le **R601.1** dans l'ISO, une fois appuyé, ca va allumer une led au dessus du bouton pour prévenir que l'étape a bien été effectuée. Ensuite pour activer la Mémoire Changement Consigne, il faut qu'on appuie soit sur le bouton D03 du clavier soit en envoyant le **R601.7** dans l'ISO.

Si les conditions des mémoires Cons. P1, Vitesse, Correction Axe Z OK et Diamètre Buse OK ne sont pas remplies mais que celle de Changement Consigne est remplie cela va activer la Mémoire Not 4aii Global et réinitialiser la mémoire Changement Consigne.

Maintenant si les conditions des mémoires Cons. P1, Vitesse, Correction Axe Z OK et Diamètre Buse OK sont complètement remplies, ça va pouvoir activer la mémoire 4aii Global et de réinitialiser les mémoires suivantes: Cons. Plateau + Buse OK, Vitesse OK, Correction Axe Z OK et Diamètre Buse OK

- Boutons Reset :**



Si on veut reset tout le programme en supprimant tous les G, les F et les R on appuie sur le bouton D01 du clavier ou sur l'ISO pour allumer la led.

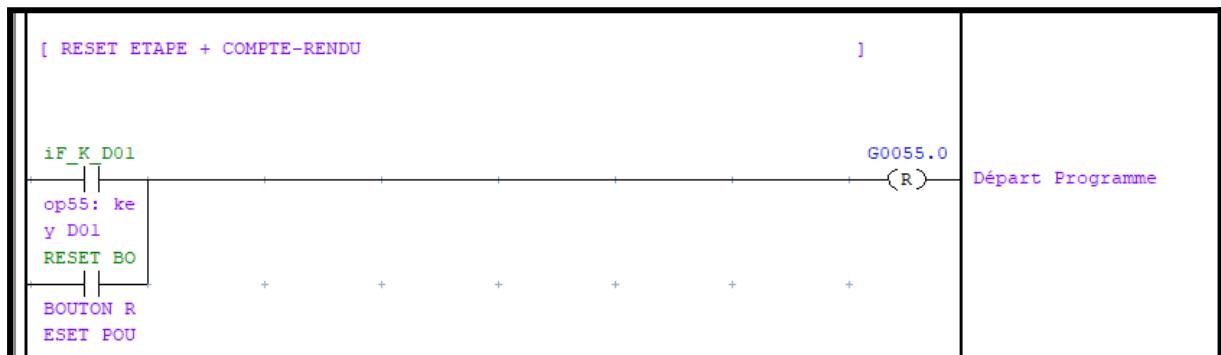


+	+	+	+	+	+		Transi_4 (R)	Memoire - Piece OK
+	+	+	+	+	+		Memoire5 (R)	Memoire - Vitesse OK
+	+	+	+	+	+		Memoire6 (R)	Memoire - Correction Axe Z OK
+	+	+	+	+	+	+	Memoire7 (R)	Memoire - Diametre Buse OK
+	+	+	+	+	+	+	Memoire8 (R)	Memoire - Changement Consigne
+	+	+	+	+	+	+	Transi_N (R)	Memoire - NOT 4aii global / lb
+	+	+	+	+	+	+	Memoire9 (R)	Memoire - Activation Buse IHM
+	+	+	+	+	+	+	Memoire1 (R)	Memoire - Code M6
+	+	+	+	+	+	+	Transi_1 (R)	Memoire - ActivationBuse + M6
+	+	+	+	+	+	+	Transi_2 (R)	Memoire - Buse Actuelle

+	+	+	+	+	+	+	Memoire_1 (R)	Mémoire - Nouvelle Buse
+	+	+	+	+	+	+	Transi_3 (R)	Mémoire - NewBuse + Cons.Buse
+	+	+	+	+	+	+	Transi_4 (R)	Mémoire - Fonction Acquitte
+	+	+	+	+	+	+	Transi_4 (R)	Mémoire - 4aii Golbal
+	+	+	+	+	+	+	Verif_M6 (R)	Verifie debut Code M6
+	+	+	+	+	+	+	Memoire_ (R)	Mémoire - 4aii
+	+	+	+	+	+	+	Memoire_ (R)	Mémoire - 1b
+	+	+	+	+	+	+	E_la (R)	Etape la - G54.1
+	+	+	+	+	+	+	E_lb (R)	Etape lbi et lc - G54.2
+	+	+	+	+	+	+	E_2a (R)	Etape 2a - G54.3

+	+	+	+	+	+	+	E_3a (R)	Etape 3a - G54.4
+	+	+	+	+	+	+	E_4ai (R)	Etape 4ai - G54.5
+	+	+	+	+	+	+	E_4aai (R)	Etape 4aai et lc - G54.6
+	+	+	+	+	+	+	G0055.0 (R)	Départ Programme
+	+	+	+	+	+	+	E_2c (R)	Etape 2c - G55.2
+	+	+	+	+	+	+	E_4c (R)	Etape 4c - G55.4
+	+	+	+	+	+	+	E_3c (R)	Etape 3c - G55.3

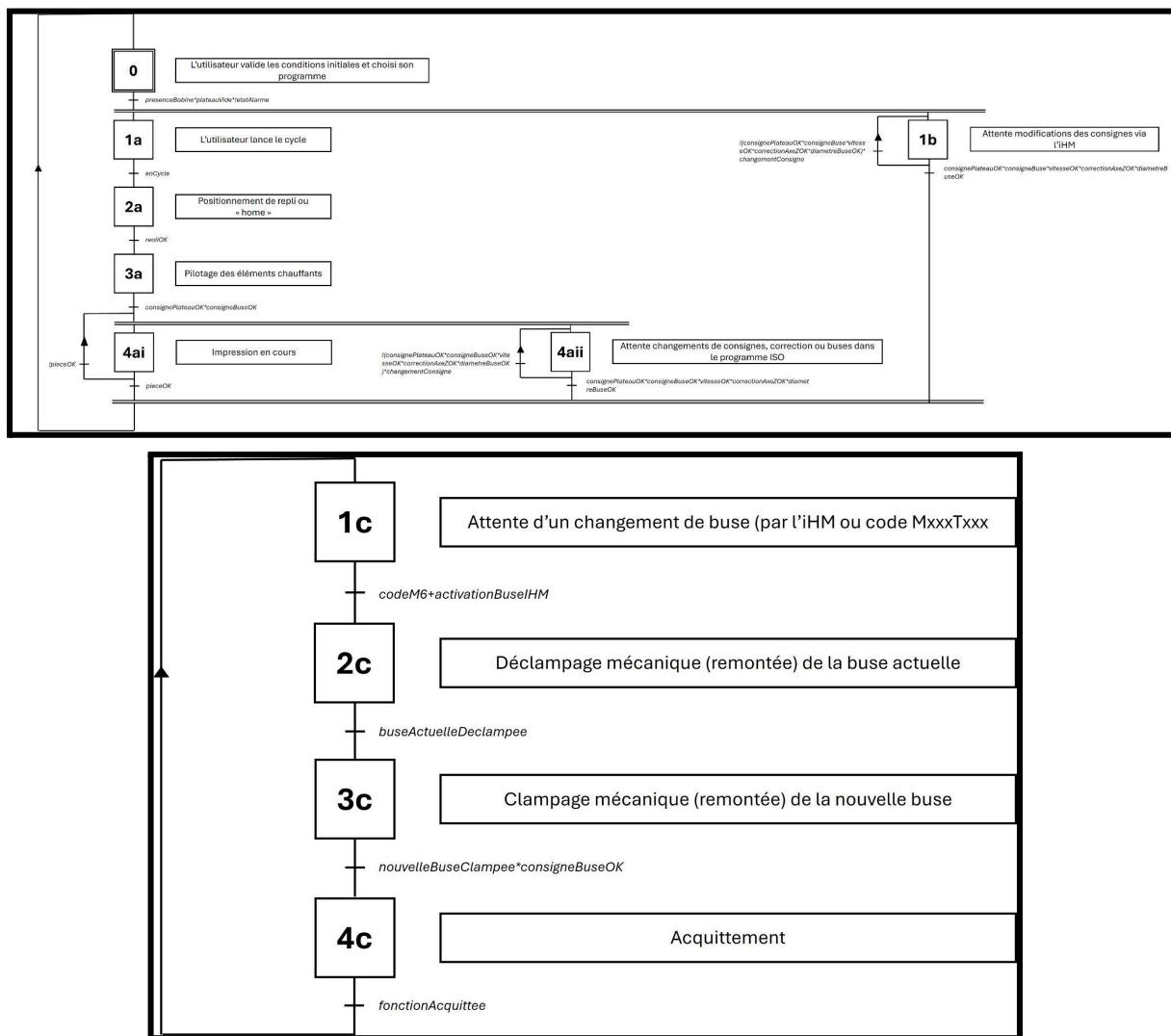
Si on appuie le bouton reset du clavier, ou sur l'ISO ou que le programme est fini, on va reset tout le programme entier en enlevant tous les G, les F et les R pour éviter que le programme se bloque dans une étape.



Et pour finir pour débuter le programme il faut appuyer sur le bouton D01 du clavier ou alors on appuie sur le bouton reset.

MISSION 4 : PROGRAMMATION ISO

Dans cette partie, nous allons mettre en place le programme permettant le fonctionnement du mode AUTO décrit par les grafcets suivant:



```
<Olympiades_FANUC_BUT3_24-25>
;
G54 G90 G94;
;
(Valeurs Limites ABSOLU - Plaque Metal)
#501 = 0; (Min X)
#502 = 401; (Max X)
#504 = 0; (Min Y)
#505 = 280; (Max Y)
#507 = 0; (Min Z)
#508 = 120; (Max Z)
;
(VARIABLES COORDONNEES AXES)
#500 = 0; (Co de X)
#503 = 0; (Co de Y)
#506 = 0; (Co de Z)
;
```

G54 : Décalage d'origine de pièce

G90 : cotation absolue

G94 : Avance par minute

On commence par initialiser des variables qui contiennent les limites d'axes que nous avons décidés, nous avons décidé de tracer les coordonnées pour faire en sorte qu'elles représentent la plaque en métal, ces coordonnées sont arbitraires et peuvent être modifiées si elles ne correspondent pas en fonction de la machine.

Les variables utilisées pour stocker les coordonnées des axes sont toujours les mêmes et sont fixées sur

#500, #503, #506.

```
(Variables de conditions)
#700 = 1; (code M6)
#800 = 0; (Capteurs)
#801 = 0; (Pourcentage avancement)
#802 = 0; (Type bobine matériaux couleur)
#803 = 1; (Variable 1b)
#804 = 0; (Température plateau)
#805 = 0; (Température buse)
;
```

Nous avons ensuite Set des variables nous permettant plusieurs choses.

Tout d'abord, la variable **#800** est une variable que l'on incrémente à chaque étape, cette variable en s'incrémentant permet au programme ISO en la vérifiant dans nos conditions de ne pas aller dans la mauvaise étape. Tout en sachant où on en est à tout moment.

Ensuite, la variable **#801** est une variable qui se modifie à chaque étape de l'ISO et permet de stocker le pourcentage d'avancement de l'ISO.

Après cela, la variable **#802**, lorsque dans picture on choisit un matériau de bobine et une couleur, cette valeur est envoyée dans la variable **#802** et permet de choisir le bon mode de chauffe en fonction du matériau.

La variable **#700** est liée au code M permettant la permission du changement de buse et elle est de base à 1, grâce à cette variable, le programme ne restera donc pas bloqué dans la boucle.

Les variables **#804** et **#805** sont des variables dont la valeur est modifiable via l'IHM, pour pouvoir choisir les températures de chauffe du plateau et de la buse en fonction du matériau utilisé pour l'impression.

La variable **#803** est de base à 1 et sert l'étape 1b, elle sert d'équivalent à notre **#800** mais juste pour l'étape 1b

```
(TEST MESSAGE ERREUR COOR. X-Y-Z)
IF[[#500LT#501] OR [#500GT#502] OR [#503LT#504] OR [#503GT#505] OR [#506LT#507] OR [#506GT#508]] THEN #3000 = 12; (ATTENTION LIMITES AXES DEPASSEES)
;
```

Le **THEN #3000 = 12**, avec le message entre parenthèses après permet de déclencher une alarme bloquante, indiquant que les axes vont être en surcourses.

```
(MESSAGE TEMPERATURES FAUSSES)
IF[[#804 NE 60] OR [#804 NE 90] OR [#804 NE 75] OR [#804 NE 80] OR [#805 NE 200] OR [#805 NE 240] OR [#805 NE 255]] THEN #3000 = 12; (ATTENTION TEMPERATURES INCORRECTES, ALLEZ SUR LA PAGE AIDE)
;
```

Dans cette condition, on test toutes les valeurs de température, puis via le **THEN #3000 = 12**, j'envoie une alarme bloquante indiquant que les températures sont incorrectes et qu'il faut aller se référer à la page Aide, où il y a un tableau indiquant les températures en fonction du matériau choisi. Cette fonction n'est pas encore intégrée car, on l'initialise de base à 0 et donc notre programme déclenche directement l'alarme.

Matériaux:	Température		Coordonnées Bobines		
	Plateau (#804)	Buse (#805)	X	Y	Z
PLA	60°C	200°C	400	200	100
ABS	90°C	240°C	400	200	100
PETG	75°C	240°C	400	100	100
NYLON	80°C	255°C	500	100	100

```
(VARIABLES COORDONNEES BOBINES)
#518 = 10 (Z DE TOUTE LES BOBINES)
(BLANC NOIR)
#510 = 400; (X)
#511 = 200; (Y)
;
(ROUGE)
#512 = 400; (X)
#513 = 100; (Y)
;
(BLEU)
#514 = 300; (X)
#515 = 100; (Y)
;
(GRIS)
#516 = 300; (X)
#517 = 200; (Y)
;
```

Nous avons décidé de créer des variables permettant de stocker la position des bobines, pour pouvoir s'y déplacer lors du programme, la coordonnée Z est commune à toutes les bobines car c'est la coordonnée de hauteur, et les coordonnées X et Y sont spécifiques à chaque bobine (sauf blanche et noire qui sont communes). Ces valeurs sont modifiables au besoin.

```
(#Av0 - CONDITION INIT ET PROG)
IF[#800EQ0] GOTO 999;
;
```

```
(CONDITION INIT ET PROG)
N999 M40
#800 = 1; (Incrémentation de la valeur d'avancement)
#801=9; (Incrémentation du pourcentage d'avancement)
GOTO 10; (Vers condition 1A)
;
```

L'étape 0 pour les conditions initiales du programme, on vérifie juste que le **#800** vaut 0 puis on va à l'étape 0 via un **GOTO 999**.

Dans cette étape, on passe le **#800** à 1 pour qu'il ne retourne pas dans l'étape 0, on passe **#801** à 9 pour le pourcentage d'avancement de l'ISO, via le **GOTO 10** permet d'aller à la condition de l'étape 1A.

```
(#Av1A - LANCEMENT CYCLE)
N10 IF[[#1001EQ1] AND [#800EQ1]] GOTO 998;
;
```

```
(LANCEMENT CYCLE)
N998 M40
#800 = 2; (Incrémentation de la valeur d'avancement)
#801=20; (Incrémentation du pourcentage d'avancement)
#1101 =1; (Set F54.1)
GOTO 20; (Vers condition 2A)
;
```

L'étape 1A pour le lancement de cycle, on vérifie que le **#800** vaut 1 et on vérifie que **#1001** vaut 1, la variable **#1001** est liée à la commande **G54.1**. Si les conditions sont remplies, on va à l'étape 1A via **GOTO 998**. Dans cette étape, on fait comme la précédente, on incrémente la variable **#800** à 2, on augmente le pourcentage d'avancement de l'ISO et on passe le **F54.1 (#1101)** à 1 pour dire que l'étape est faite, via le **GOTO 20** permet d'aller à la condition de l'étape 2A.

```
(#AV2A - POSITIONNEMENT REPLI)
N20 IF[[#1003EQ1] AND [#800EQ2]] GOTO 997;
;
```

```
(POSITIONNEMENT REPLI)
N997 M40
#1101 = 0; (Reset F54.1)
G01 Z#506 F1000; (Déplace Axe Z)
G01 X#500 E#500 F1000; (Déplace Axe X et E)
G01 Y#503 F1000; (Déplace Axe Y)
#800 = 3; (Incrémentation de la valeur d'avancement)
#801=20;(Incrémentation du pourcentage d'avancement)
#1103 = 1; (Set F54.3)
GOTO 30; (Vers condition 3A)
;
```

A partir de cette étape, toutes les étapes suivantes ont à leur début **M40**, code M pour que le ladder ne passe pas cette étape.

L'étape 2A pour le positionnement en repli, on vérifie que **#800** vaut 2 et que **#1003** vaut 1, la variable **#1003** est liée à la commande **G54.3**. Si les conditions sont remplies, on va à l'étape 2A via **GOTO 997**. Dans cette étape, on reset le **F54.1** en passant la variable **#1101** à 0, ensuite on déplace les axes à la position 0, 0, 0 définie précédemment, on les fait déplacer à la vitesse de travail choisie avec **G01** et on a décidé de le mettre à la vitesse 1000 via **F1000**.

Une fois le déplacement d'axe fait, on incrémente **#800**, on augmente le pourcentage d'avancement de l'ISO (**#801**) et on envoie le compte rendu **F54.3** en passant la variable **#1103** à 1.

Enfin, via le **GOTO 30** permet d'aller à la condition de l'étape 3A.

```

(#Av3A - Pilotage des Elements chauffants PLA) ;
N30 IF[[#1004 EQ 1]AND[#800 EQ 3]AND[[#802 EQ 1000]OR[#802 EQ 1001]OR[#802 EQ 1002]OR[#802 EQ
1003]OR[#802 EQ 1004]] AND [#804 = 60] AND [#805 = 200]] GOTO 9960 ;
;
(#Av3A - Pilotage des Elements chauffants ABS) ;
N30 IF[[#1004 EQ 1]AND[#800 EQ 3]AND[[#802 EQ 1100]OR[#802 EQ 1101]OR[#802 EQ 1102]OR[#802 EQ
1103]OR[#802 EQ 1104]] AND [#804 = 90] AND [#805 = 240]] GOTO 9961 ;
;
(#Av3A - Pilotage des Elements chauffants PETG) ;
N30 IF[[#1004 EQ 1]AND[#800 EQ 3]AND[[#802 EQ 1200]OR[#802 EQ 1201]OR[#802 EQ 1202]OR[#802 EQ
1203]OR[#802 EQ 1204]] AND [#804 = 75] AND [#805 = 240]] GOTO 9962 ;
;
(#Av3A - Pilotage des Elements chauffants NYLON) ;
N30 IF[[#1004 EQ 1]AND[#800 EQ 3]AND[[#802 EQ 1300]OR[#802 EQ 1301]OR[#802 EQ 1302]OR[#802 EQ
1303]OR[#802 EQ 1304]] AND [#804 = 80] AND [#805 = 255]] GOTO 9963 ;
;
(#Av3A - Pilotage des Elements chauffants BOBINE VIDE) ;
N30 IF[[#1004 EQ 1]AND[#800 EQ 3]AND[[#802 EQ 1005]OR[#802 EQ 1105]OR[#802 EQ 1205]OR[#802 EQ
1305]]] THEN #3000 =12 ; (ATTENTION BOBINES VIDÉS) ;
;
```

L'étape 3A est découpée en **5 possibilités**, 4 pour le matériau choisi et 1 pour le cas où les bobines sont vides. Pour les conditions, on vérifie d'abord la commande **G54.4** ainsi que la valeur de notre variable à **#800**. Ensuite, on vérifie la valeur de notre variable **#802** qui permet de stocker le matériau et la couleur. (les 2 premiers chiffres pour le matériau et les 2 suivants pour la couleur, conformément au sujet, **PLA:10, ABS:11, PETG:12, NYLON: 13**). Enfin, on vérifie la température de chauffe de la buse et du plateau qui a été entré (conformément aux températures moyennes de chauffe du plateau et de la buse en fonction du matériau sélectionné).

PLA: plateau: 60°C, buse: 200°C

ABS: plateau: 90°C, buse: 240°C

PETG:, plateau: 75°C, buse: 240°C

NYLON: plateau: 80°C, buse: 255°C

Pour ce qui est du cas de la bobine vide, le vide est considéré comme la "couleur" 05, si ce cas apparaît, on envoie un **THEN #3000 = 12**, avec le message entre parenthèses après permet de déclencher une alarme bloquante, indiquant que les bobines sont vides.

Pour ce qui est des GOTO, vu que sur le GRAFCET ils décrivent la même étape, nous avons décidé de décrémenter la valeur (de 997 à 996, tout en ajoutant un chiffre derrière afin de distinguer tout en gardant la continuité, 9960, 9961 etc).

```

(PILOTAGE ELEMENT CHAUFFANT PLA)
N9960 M40 ;
#1103 =0 ; (Reset F54.3)
G01 X#510 Y#511 Z#518 F1000 (Axes X Y Z Vers la bobine Blanche) ;
#800 =4 ; (Incrémentation de la valeur d'avancement)
#801 =40 ; (Incrémentation du pourcentage d'avancement)
#1104 =1 ; (Set F54.4)
#1106 = 1; (Set F54.6)
GOTO 40 ; (Vers conditions 4AI et 4AII)
;
(Pilotage Elements Chauffants ABS) ;
N9961 M40 ;
#1103 =0 ;(Reset F54.3)
G01 X#510 Y#511 Z#518 F1000 (Axes X Y Z Vers la bobine Noire) ;
#800 =4 ; (Incrémentation de la valeur d'avancement)
#801 =40 ; (Incrémentation du pourcentage d'avancement)
#1104 =1 ; (Set F54.4)
#1106 = 1; (Set F54.6)
GOTO 40 ; (Vers conditions 4AI et 4AII)
;
(Pilotage Elements Chauffants PETG) ;
N9962 M40 ;
#1103 =0 ; (Reset F54.3)
G01 X#512 Y#513 Z#518 F1000 (Axes X Y Z Vers la bobine Rouge) ;
#800 =4 ; (Incrémentation de la valeur d'avancement)
#801 =40 ;(Incrémentation du pourcentage d'avancement)
#1104 =1 ;(Set F54.4)
#1106 = 1;(Set F54.6)
GOTO 40 ; (Vers conditions 4AI et 4AII)
;
(Pilotage Elements Chauffants NYLON) ;
N9963 M40 ;
#1103 =0 ; (Reset F54.3)
G01 X#514 Y#515 Z#518 F1000 (Axes X Y Z Vers la bobine Bleue) ;
#800 =4 ; (Incrémentation de la valeur d'avancement)
#801 =40 ; (Incrémentation du pourcentage d'avancement)
#1104 =1 ; (Set F54.4)
#1106 = 1; (Set F54.6)
GOTO 40 ; (Vers conditions 4AI et 4AII)
;
```

Dans les étapes 3A, on reset tout d'abord le **F54.3** en passant **#1103** à 0, on déplace ensuite nos axes vers une bobine choisie arbitrairement, on incrémente la variable **#800**, on augmente le pourcentage d'avancement en modifiant la valeur de **#801**, on set les compte rendus **F54.4** et **F54.6** et via le **GOTO 40**, on va vers la condition pour l'étape 4AI et 4AII.

```
(#Av4AI - Impression en Cours) ;
N40 IF[[#1005 EQ 1]AND[[#800 EQ 4]OR[#800 EQ 5]]] GOTO 995 ;
;
```

```
(IMPRESSION EN COURS)
N995 M40
#1104 = 0; (Reset F54.4)
(APPEL SOUS PROGRAMME o2025)
#800 = 5; (Incrémentation de la valeur d'avancement)
#801=80; (Incrémentation du pourcentage d'avancement)
#1105 = 1; (Set F54.5)
GOTO 90; (Vers début programme pour RESET)
;
```

L'étape 4AI pour l'impression en cours, on vérifie que la commande **G54.5** est à 1 via la variable **#1005**, on vérifie l'avancement de **#800** et on va à l'étape 4AI via **GOTO 995**.

Dans l'étape 4AI, on passe le compte rendu de l'étape 3AI à 0, on incrémente le #800, on augmente la variable de pourcentage d'avancement de l'ISO, on set le F54.5 à 1 et via le **GOTO 90**, on va vers la condition pour l'étape 4AI et 4AII.

```
(#Av1C et 4AII - ATTENTE CHANGEMENT BUSE)
N40 IF[[#1006EQ1] AND [[#800EQ5]OR[#800 EQ 9]]] GOTO 9940;
;
(#Av1C et 1B - ATTENTE CHANGEMENT BUSE)
N10 IF[[#1002EQ1] AND [[#803EQ1]OR[#800 EQ 9]]] GOTO 9941;
;
```

L'étape 4AII pour l'attente de changement de buse, on vérifie **G54.6** si on de l'étape 3A et **G54.2** si on vient de l'étape 0, on vérifie l'avancement de **#800** de l'étape 4AI, ou 3A, ou de l'étape 4C pour 4AII. Pour l'étape 1B, on vérifie seulement que la variable **#803** est à 1 car on a pas encore augmenté le **#800** à ce moment-ci du programme, vu que sur le GRAFCET ils décrivent la même étape, nous avons décidé de décrémenter la valeur (de 995 à 994, tout en ajoutant un chiffre derrière afin de distinguer tout en gardant la continuité, 9940 et 9941).

```
(IMPRESSION EN COURS)
N995 M40
#1104 = 0; (Reset F54.4)
(APPEL SOUS PROGRAMME o2025)
#800 = 5; (Incrémentation de la valeur d'avancement)
#801=80; (Incrémentation du pourcentage d'avancement)
#1105 = 1; (Set F54.5)
GOTO 90; (Vers début programme pour RESET)
;
(ATTENTE CHANGEMENT BUSE 1C 4AII)
N9940 M40
#1104 = 0; (Reset F54.4)
#1112 = 0; (Reset F55.4)
M6;(Code M)
#700=0; (Reset variable code M)
#800 = 6; (Incrémentation de la valeur d'avancement)
#801=65; (Incrémentation du pourcentage d'avancement)
#1106 = 1;(Set F 54.6, pour rebouclage)
GOTO 50; (Vers condition 2C)
;
```

Dans cette étape, pour le cas 4All, on passe **F54.4 et F54.5 à 0**, on active le code **M6**, on passe la variable du code M, **#700 à 0**, on incrémente **#800 et #801**, on Set **F54.6**.

Dans cette étape, pour le cas 1B, on reset **#803**, pour qu'il ne retourne pas dans cette étape, on incrémente **#801** et on Set **F54.2** pour l'envoyer au ladder.

```
(#Av2C - DECLAMPAGE)
N50 IF[[#1010EQ1] AND [[#800EQ6]OR [#803EQ0]]] GOTO 993;
;
```

```
(DECLAMPAGE)
N993 M40
#1102 = 0; (Reset F54.2)
#800 = 7; (Incrémentation de la valeur d'avancement)
#801=70; (Incrémentation du pourcentage d'avancement)
#1110 = 1; (Set F55.2)
GOTO 60; (Vers condition 3C)
;
```

L'étape 2C pour le déclampage, on vérifie que **G55.2** est à 1 via **#1010**, si on vient de l'étape 1C de la partie 4All, on vérifie **#800 = 6**, si on vient de 1C de la partie 1B, on vérifie que **#803 = 0** (l'étape précédente est finie). Puis on va à l'étape de déclampage via **GOTO 993**.

Dans cette étape, on reset **F54.2** en passant **#1102 à 0**, on incrémente **#800** en le passant à 7, on augmente le pourcentage d'avancement de l'ISO et on envoie le compte rendu **F55.2** au ladder en passant **#1110 à 1** et via le **GOTO 60**, on va vers la condition pour l'étape 3C.

```
(#Av3C - CLAMPAGE)
N60 IF[ [#1011EQ1] AND [#800EQ7]] GOTO 992;
;
```

```
(CLAMPAGE)
N992 M40
#1110 = 0; (Reset F55.2)
#800 = 8; (Incrémentation de la valeur d'avancement)
#801=75; (Incrémentation du pourcentage d'avancement)
#1111 = 1; (Set F55.3)
GOTO 70; (Vers condition 4C)
;
```

L'étape 3C pour le clampage, on vérifie que **G55.3** est à 1 via **#1011**, on vérifie **#800 = 7**. Puis on va à l'étape de déclampage via **GOTO 992**.

Dans cette étape, on reset **F55.2** en passant **#1110** à **0**, on incrémente **#800** en le passant à **8**, on augmente le pourcentage d'avancement de l'ISO et on envoie le compte rendu **F55.3** au ladder en passant **#1111** à **1** et via le **GOTO 70**, on va vers la condition pour l'étape 4C.

```
(#Av4C - ACQUITTEMENT)
N70 IF[ [#1012EQ1] AND [#800EQ8]] GOTO 991;
;
```

```
(ACQUITTEMENT)
N991 M40
#1111 = 0; (Reset F55.3)
#800 = 9; (Incrémentation de la valeur d'avancement)
#801=80; (Incrémentation du pourcentage d'avancement)
#1112 = 1; (Set F55.4)
GOTO 90; (Vers début programme pour RESET)
;
```

L'étape 4C pour l'acquittement, on vérifie que **G55.4** est à 1 via **#1012**, on vérifie **#800 = 8**. Puis on va à l'étape de déclampage via **GOTO 991**.

Dans cette étape, on reset **F55.3** en passant **#1111** à **0**, on incrémente **#800** en le passant à **9**, on augmente le pourcentage d'avancement de l'ISO et on envoie le compte rendu **F55.4** au ladder en passant **#1112** à **1** et via le **GOTO 90**, on va vers la condition pour le reset.

```
(RESET)
IF[#1013EQ1] GOTO 990;
;
```

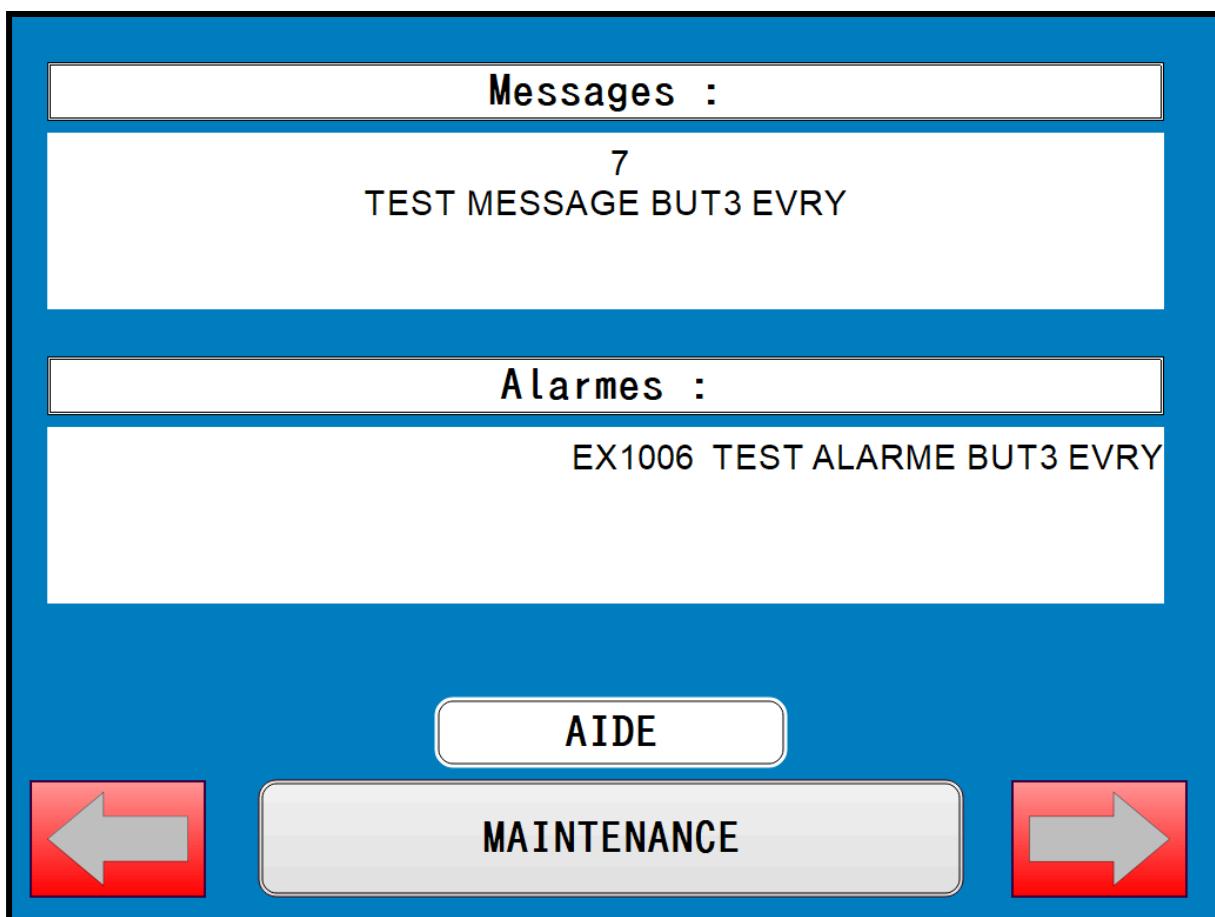
```
(RESET)
N990 M40
G00 X0; (RESET POS)
G00 Y0 E0 Z0; (RESET POS)
#1101=0; (F54.1, pour G54.1)
#1102=0; (F54.2, pour G54.2)
#1103=0; (F54.3, pour G54.3)
#1104=0; (F54.4, pour G54.4)
#1105=0; (F54.5, pour G54.5)
#1106=0; (F54.6, pour G54.6)
#1110=0; (F55.2, pour G55.2)
#1111=0; (F55.3, pour G55.3)
#1112=0; (F55.4, pour G55.4)
#1113=0; (F55.5, pour G55.5)
#1114=0; (F55.6, pour G55.6)
#700 = 1; (code M6)
#800 = 0; (Capteurs)
#801 = 0; (Pourcentage avancement)
#802 = 0; (Type bobine matériaux couleur)
#803 = 1; (Variable 1b)
#804 = 0; (Température plateau)
#805 = 0; (Température buse)
GOTO 120;
%
```

Pour le reset, on vérifie que **#800** vaut 9, puis on va à l'étape de RESET avec **GOTO 990**.
 Dans l'étape de RESET, on remet nos axes à la position 0, 0, 0, on RESET tous les F en les passants à 0, et on passe toutes les variables utilisées dans le programme à 0.

MISSION 5 : GESTION DES ALARMES

Alarmes sur FANUC PICTURE : IHM

Sur cette page du FANUC Picture, nous faisons l'affichage des messages ainsi que des alarmes. Afin de pouvoir effectuer cela, nous utilisons le LADDER afin de remonter des messages simples, ou alors des alarmes bloquantes.



De plus, sur cette page, nous pourrons trouver une pop up “AIDE” permettant au client, de pouvoir connaître les températures qui sont associées à chaque matériaux.

Il pourra ainsi ajuster la température dans la page opérateur en y entrant celle qui correspond au matériau qu'il a choisi au départ du programme.

A noter, que nos températures sont des valeurs moyenne trouvées sur internet, afin d'avoir une approximation.

PENSEZ A METTRE LES TEMPERATURES DE PLATEAU ET DE BUSE CORRESPONDANT AU MATERIAU CHOISI

Matériaux:	Température		Coordonnées Bobines		
	Plateau (#804)	Buse (#805)	X	Y	Z
PLA	60°C	200°C	400	200	100
ABS	90°C	240°C	400	200	100
PETG	75°C	240°C	400	100	100
NYLON	80°C	255°C	500	100	100

AIDE



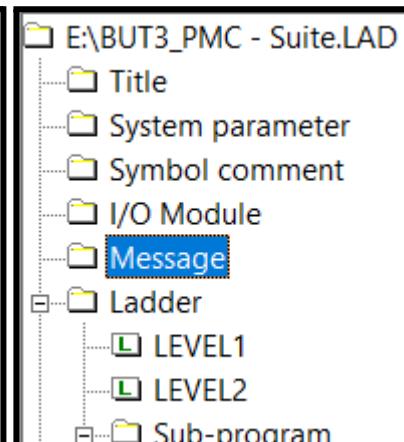
Pour les alarmes, nous avons effectué un bouton test, qui est en position **D02**, sous le nom, **TA**, du clavier afin de pouvoir y avoir une avant première.



Alarmes sur FANUC LADDER III : PMC

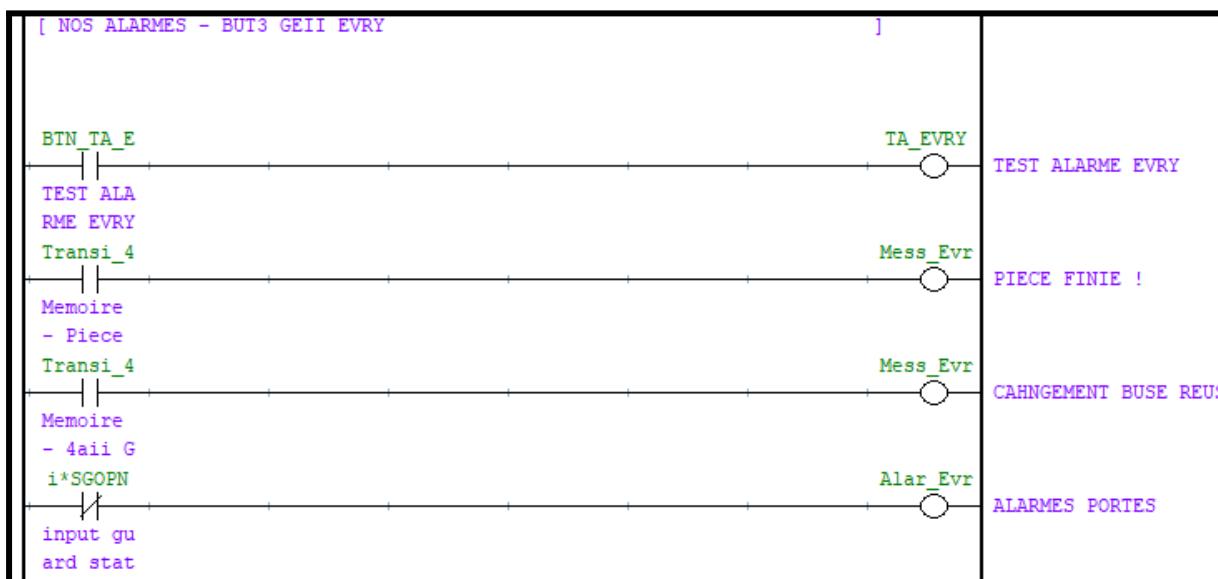
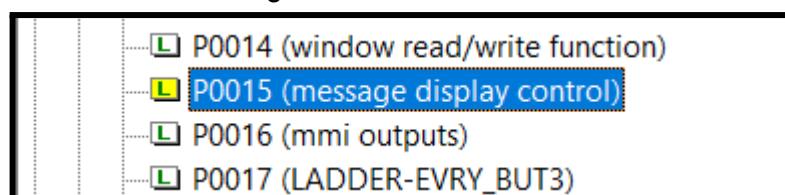
Voici comment nous avons paramétré nos alarmes et messages depuis le LADDER :

1	A0.0	AL1+000=EMERGENCY STOP ACTIVE
2	A0.1	AL1+001=SPINDLE ALARM
3	A0.2	AL1+002=ERROR CENTRAL LUBRICATION
4	A0.3	OP1+004=ABNORMAL TOOL DETECTION
5	A0.4	AL1+004=TIMEOUT IPENDANT
6	A0.5	OP1+005=Z AXIS NOT SAFE
7	A0.6	
8	A0.7	AL1+007=PORTE (S) OUVERTE (S)
9	A1.0	OP1+008=PIECE FINIE !
10	A1.1	OP1+009=CHANGEMENT BUSE REUSSI !
11	A1.2	OPL+010=TEST EVRY BUT 3



AL1+00x avec x qui est le bit de l'adresse, permet de faire une alarme bloquante.

OP1+00x permet de faire un message.

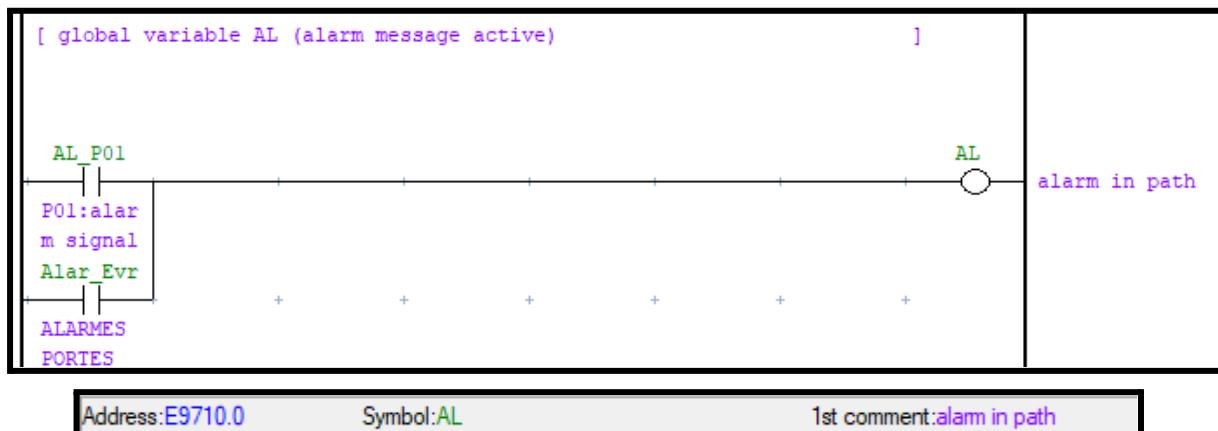


L'entrée "input guard stat" est censée fonctionner pour déclencher une alarme lorsque la porte de la CN est ouverte mais étant sur machine virtuelle, cela ne fonctionne pas.

Nous avons aussi trouver la sortie qui fait alarme bloquante, "alarm in path" nous permettant de répondre au début du grafset, où nous avions la demande suivant :

presenceBobine*plateauVide!*etatAlarme

Nous avons ainsi, récupéré en entrée, l'adresse **E9710.0** qui correspond à l'état d' alarme, si nous avons une alarme alors nous ne pouvons pas lancer le programme.



CONCLUSION

Ce projet nous a permis d'appliquer nos compétences et d'approfondir notre connaissance des outils FANUC.

Chaque tâche effectuée a amélioré notre savoir-faire, que ce soit dans les domaines de la programmation et de l'automatisation des systèmes industriels.

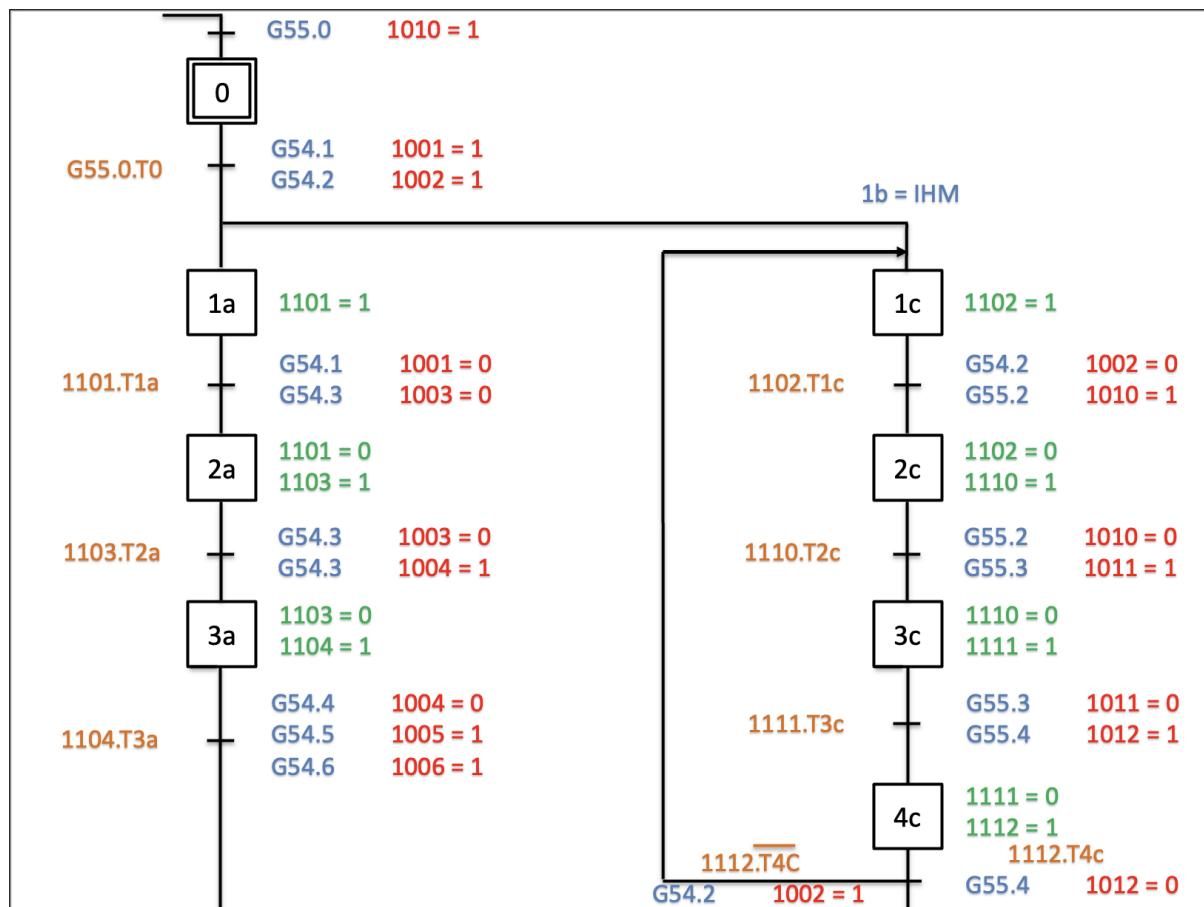
Les obstacles auxquels nous avons été confrontés ont constitué des défis à surmonter, et nous avons pu apprendre davantage grâce à ce concours.

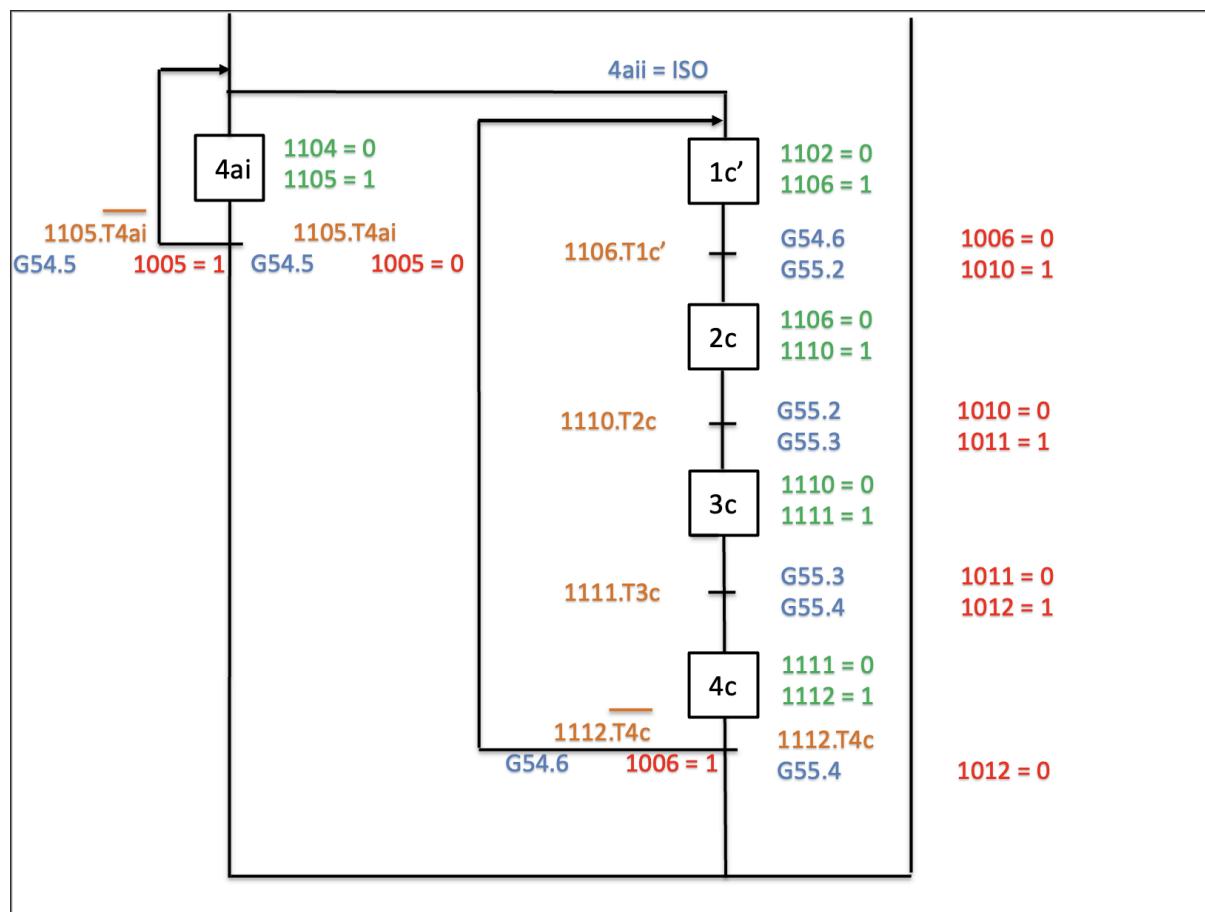
Ce concours nous a permis d'acquérir des aptitudes solides, comme la gestion de projet et le travail d'équipe, qui seront inestimables pour notre future carrière.

- **Améliorations possibles :**

- Led sur clavier MOP
- La vitesse de travail
- Ajout alarmes / messages supplémentaires
- Synoptique animée
- Ajustement température avec un bouton + / -
- Ajouter des LED d'alertes pour nos alarmes dans nos pages
- Réussir l'implémentation de la fonction d'alarme de la température
- Remettre le timer à 0 lorsque le dernier G est activé

Grafcet des coordonnées





Voici le grafcet qui permet de voir toutes données en détails avec les différentes couleurs pour bien visualiser clairement et précisément les données.

- => Les F récupérées de l'ISO
- => Les G envoyées à l'ISO
- => Transitions entre les actions
- => Actions