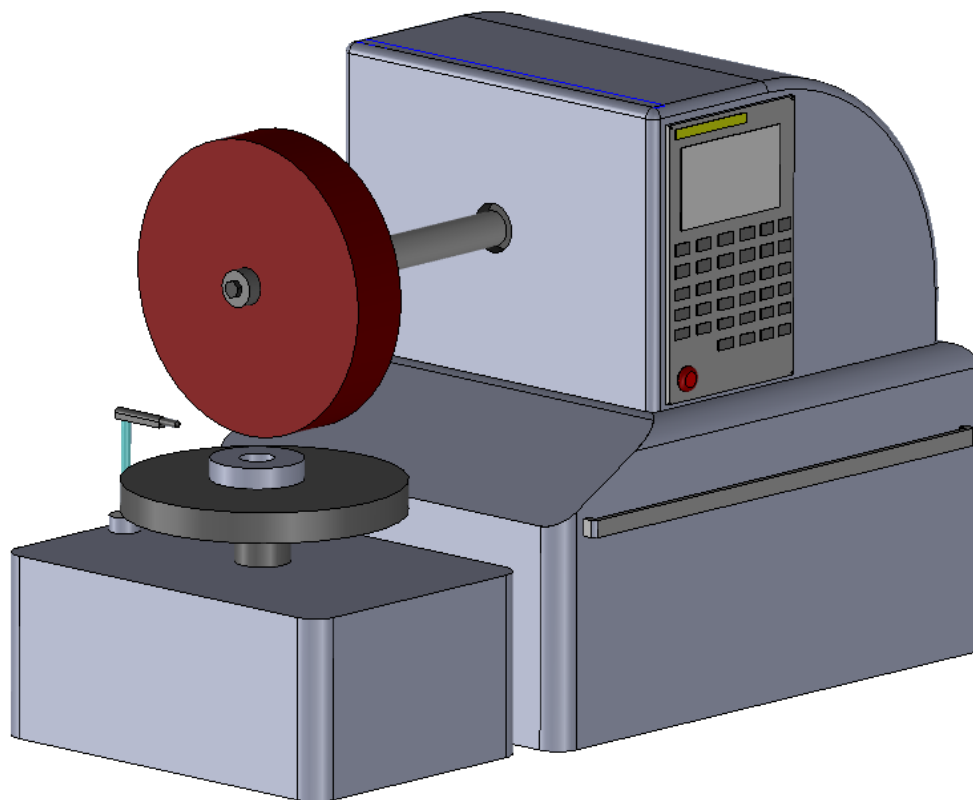


Concours FANUC



D4R4 – Rapport de fin de projet

ROB1 – Groupe 304



Victor BILLAUD

Promotion 2019-2021

Année 2020-2021

Concours FANUC

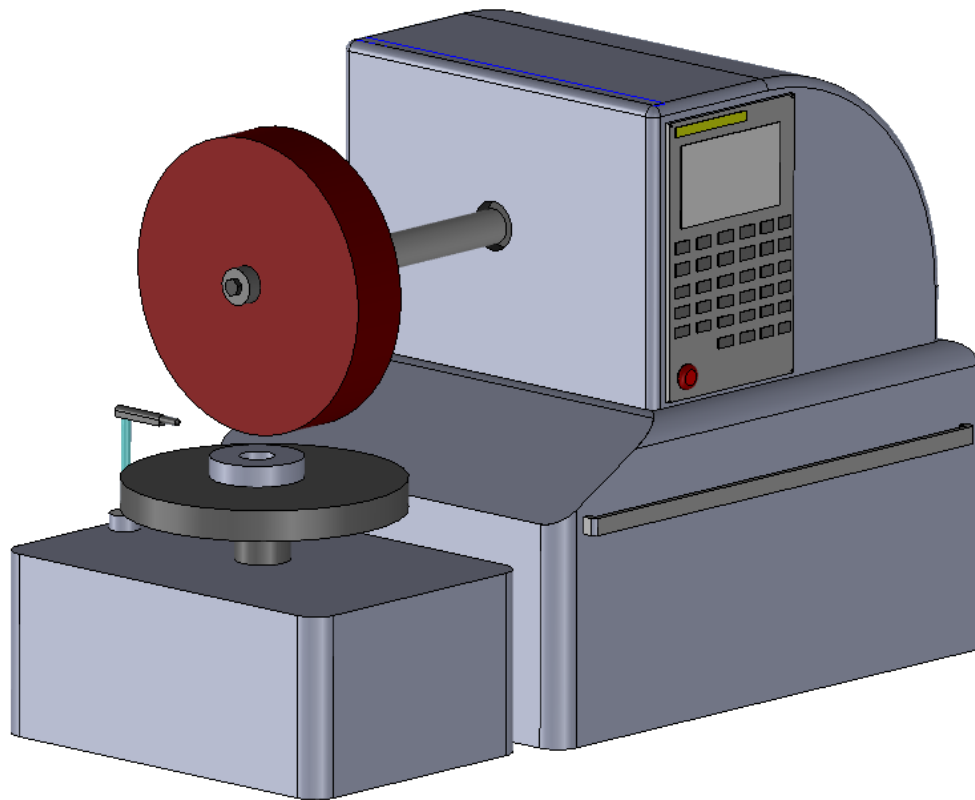


Figure 1 : Image de couverture : modèle 3D d'une rectifieuse, réalisé par nos soins

D4R4 – Rapport de fin de projet

ROB1 – Groupe 304



Victor BILLAUD

Promotion 2019-2021

Année 2020-2021

Sommaire

Introduction	5
Développement	6
1) Présentation du projet et de son organisation	6
1.1 Le cahier des charges du projet	6
1.2 Evolution de la carte mentale	8
1.3 Evolution du planning prévisionnel	10
1.4 Répartition des tâches	12
1.5 Evolution des coûts, besoins financiers et matériels.....	13
2) Analyse du concours.....	14
2.1 Explications générales et composition CN	14
2.2 Analyse de l'avant-projet.....	18
2.3 Attendus - Interface Homme Machine (IHM)	20
2.4 Attendus - <i>Program Machine Controller</i> (PMC) et sécurité.....	20
2.5 Attendus – Programme pièce (ou programme ISO)	21
3) Avancement du concours	22
3.1 L'interface Homme-Machine (IHM).....	22
3.2 Le programme PMC.....	37
3.3 Le programme pièce.....	42
3.4 La gestion des alarmes.....	47
3.5 Contenu de la journée de formation.....	50
3.6 Problèmes rencontrés et avenir du concours	50
3.7 Bilan technique	51
Conclusion.....	52
Résumé.....	55
Abstract.....	56
Bibliographie.....	57
Index des mots clés.....	58
Table d'illustration	59

Introduction

Dans le cadre de notre projet SUPER POUET, nous avons eu l'occasion de participer à la 3^{ème} édition du concours « Intégration Commande Numérique » (ICN) organisé par l'entreprise FANUC. Ce projet est supervisé par [REDACTED], nous sommes quatre participants ; [REDACTED] et Victor BILLAUD.

FANUC est l'entreprise leader dans le domaine de l'automatisation et organise depuis 2014 différents concours ayant pour but de former des étudiants dans les domaines de la robotique industrielle et de l'intégration de la commande numérique. Le concours auquel nous participons consiste à réaliser le programme de commande d'une machine-outil et son interface homme-machine tactile.

C'est la première année que l'IUT de Tours participe à ce concours. De plus, à l'issue de celui-ci, il nous serait possible d'obtenir une certification ICN reconnue par l'entreprise FANUC.

Chronologiquement, le concours se découpe en quatre temps. D'abord, il nous est demandé de travailler sur un avant-projet grâce aux logiciels fournis par l'entreprise ; celui-ci consiste en la réalisation d'un programme respectant un cahier des charges fourni et permet à l'entreprise de sélectionner les finalistes. Ensuite, les équipes sélectionnées participent à une journée de formation sur le site de l'entreprise dans laquelle elles apprennent à manipuler les commandes numériques réelles. A l'issue de cette journée, les équipes finalistes repartent avec leur commande numérique afin de s'entraîner jusqu'à la finale. Enfin, celles-ci sont conviées dans les locaux de l'entreprise une seconde fois et s'affrontent durant une journée d'épreuves afin de décrocher la victoire du concours d'Intégration Commande Numérique.

Nous allons ainsi chercher à **gagner le concours tout en représentant notre IUT à l'échelle nationale**. Pour cela, nous nous intéresserons dans un premier temps au déroulement général du concours et de son organisation, puis ensuite aux attendus techniques de l'avant-projet de cette édition. Il nous faudra également nous renseigner sur le fonctionnement physique de la commande numérique afin d'être opérationnel pour la finale.

Par ailleurs, à l'aide des logiciels FANUC, nous essayerons de comprendre comment programmer un cycle industriel complet d'après un cahier des charges.

Nous allons, par la suite, traiter du cahier des charges du concours et de l'avant-projet puis des compétences techniques qui en découlent avant de conclure sur l'avancement actuel du projet et son avenir.

Développement

1) Présentation du projet et de son organisation

Dans cette première partie, nous allons introduire les fondements du projet et analyser ce qui nous est demandé. Ainsi, nous traiterons du cahier des charges, de la traduction de celui-ci en carte mentale, de l'organisation temporelle du projet ainsi que de la répartition des tâches et concluons sur le budget prévisionnel que solliciterait ce projet.

1.1 Le cahier des charges du projet

Dans un premier temps, évoquons le cahier des charges du projet (cf. annexe 1). Nous identifierons, dans cette sous-partie, les acteurs du projet, détaillerons ses différentes étapes et mettrons en relief les objectifs et les attendus de celui-ci.

Avant tout, il nous est nécessaire de préciser que lorsque nous évoquerons le « concours FANUC », nous ferons seulement référence au concours auquel nous participons. En effet, les Olympiades FANUC désignent l'ensemble des différents concours organisés par l'entreprise et leur niveau d'étude correspondant. Dans notre cas, nous participons au concours d'Intégration Commande Numérique (ICN) à un niveau BAC+2.

Acteurs du concours

Le concours ICN est organisé par l'entreprise FANUC et plus particulièrement par la branche « FANUC France Education ». Il est dirigé par [REDACTED] et regroupe cette année quinze équipes de 3 à 4 étudiants.

Au sein de l'IUT, ce projet « concours » est dirigé par [REDACTED] et est subventionné par le ministère de l'Education Nationale.

Organisation du concours

Le cahier des charges nous permet de décomposer le concours en **quatre phases**.

La première consiste en la réalisation d'un **avant-projet** suivant un cahier des charges fourni ; le programme réalisé de septembre à fin novembre nous permet d'être ensuite sélectionné pour la journée de formation. Durant cette phase, nous travaillons uniquement en simulation.

Ensuite, les équipes ayant été sélectionnées seront conviées pour une **journée de formation** à FANUC France afin de prendre en main les Commandes Numériques (CN) et d'apprendre les rudiments.

Enfin, durant la troisième phase de ce concours, une **amélioration de l'avant-projet** est souhaitée dans le but de le présenter aux intervenants extérieurs (entreprises, public etc...) lors de la finale.

En dernier lieu, la **finale du concours** devrait se dérouler au siège de l'entreprise pendant 2 jours (une nuit et une journée d'épreuves). Elle serait constituée de 4 à 6 épreuves allant de 30 minutes à une heure, d'une présentation de l'avant-projet et potentiellement d'un quiz portant sur l'entreprise.

Objectifs et attendus du concours

De cette organisation, nous pouvons en déduire les deux objectifs majeurs de ce projet. Il nous faudra d'abord chercher à être sélectionnés pour la journée de formation puis ensuite remporter la finale du concours ICN 2021.

De plus, l'obtention de la certification ICN par FANUC peut s'avérer être un objectif annexe car il témoignerait du travail fourni.

Les attendus, quant à eux, se résument à prendre en mains les logiciels fournis : FANUC PICTURE, FANUC LADDER III et CNC GUIDE (voir leur environnement respectif annexe 9) et à rendre un projet complet comportant une notice explicative. En d'autres termes, nous devons réaliser une Interface Homme-Machine (IHM), un programme PMC¹ ainsi qu'un programme pièce², le tout accompagné d'un dossier résumant notre approche.

Remarque : Les attendus de chaque sous-programme seront détaillés dans leurs sous-parties correspondantes.

De ces points, nous pouvons établir une carte mentale, un planning prévisionnel ainsi qu'une estimation du budget sollicité par ce projet.

¹ *Program Machine Controller* : Il s'agit de l'automate programmable de la CN. Un programme PMC est réalisé sur FANUC LADDER III et permet le contrôle de ce dernier.

² Il s'agit du code indiquant les mouvements demandés à la machine. Il sera également souvent appelé « code G » ou « code ISO ».

1.2 Evolution de la carte mentale

Deuxièmement, évoquons l'évolution de la carte mentale. Lors du début de notre projet, nous avons réalisé la carte mentale suivante :

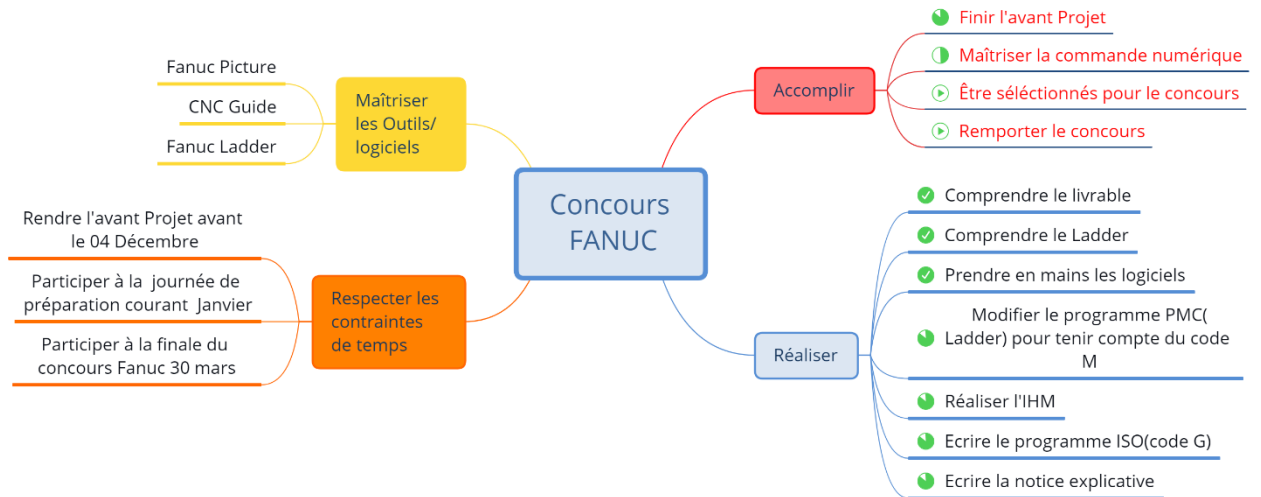


Figure 2 : Carte mentale du concours FANUC – Ancienne version

Néanmoins, celle-ci présente le défaut de ne pas faire apparaître la notion de continuité de l'avant-projet, de son prolongement dans le temps et l'apparition de la CN physique dans nos entraînements.

La carte mentale repensée devient alors :

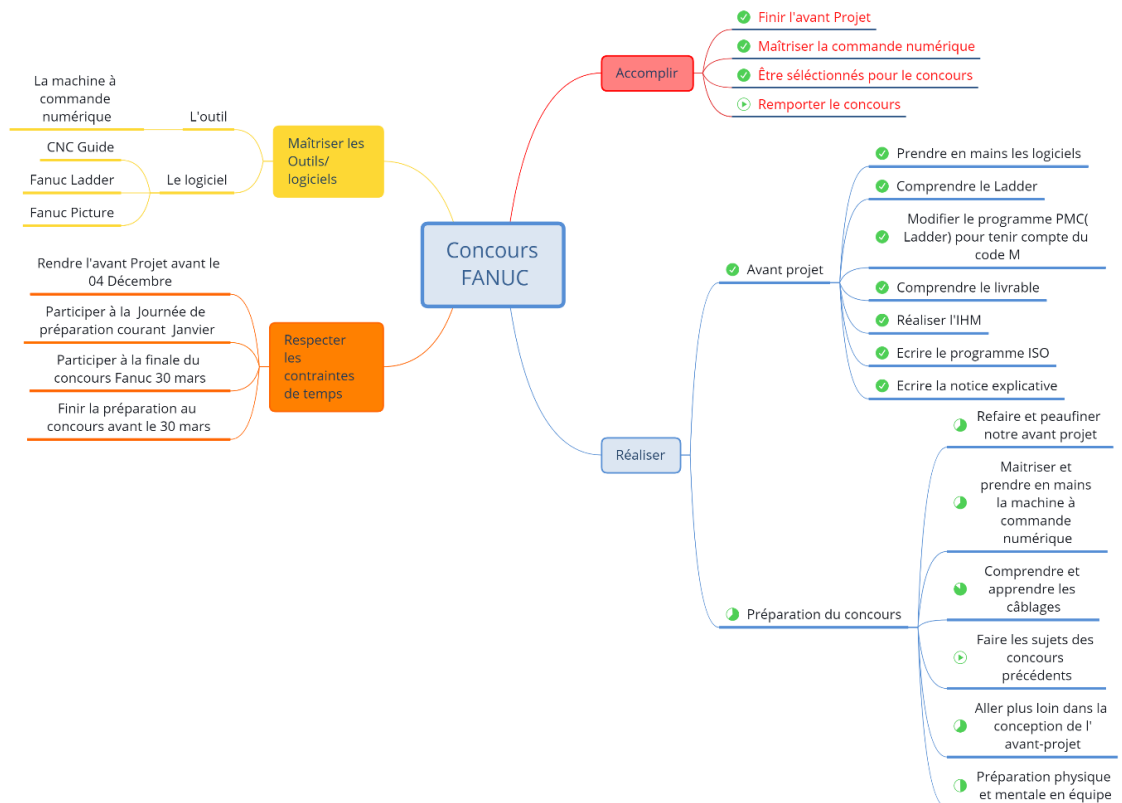


Figure 3 : Carte mentale du concours FANUC – Nouvelle version

Comme nous pouvons le remarquer, le projet repose sur les quatre axes majeurs que sont la **maîtrise** des outils et des logiciels fournis, le **respect de la contrainte de temps**, l'**accomplissement** des différentes phases du concours ainsi que la **réalisation** de l'avant-projet et de la préparation au concours.

- La maîtrise des outils et des logiciels fournis consiste en la prise en main, d'abord, des logiciels FANUC, puis de l'outil fourni : la commande numérique.
- Le respect du temps, et, plus généralement l'organisation de celui-ci, est une notion de grande importance dans le concours : bien maîtrisé, il peut nous permettre de nous qualifier. Les dates clés sont donc à anticiper dans notre réalisation afin de permettre l'accomplissement de nos objectifs.
- La réalisation présente les attendus globaux de l'avant-projet, nous devons rendre un avant-projet composé d'une Interface Homme-Machine, d'un programme PMC, d'un programme pièce et d'une notice explicative. De plus, elle présente également les événements annexes à la préparation du concours tels que les entraînements, l'apprentissage des câblages de la CN ou encore le renforcement de l'esprit d'équipe.
- Enfin, les objectifs du projet se basent sur les différentes phases : nous devons être sélectionnés parmi les meilleurs, puis apprendre à maîtriser la CN au mieux afin de gagner la finale.

De cette organisation, nous pouvons déduire le planning que nous devons suivre afin de mener à bien le projet.

1.3 Evolution du planning prévisionnel

Troisièmement, évoquons le planning réel découlant du projet. Nous traiterons d'abord du planning prévisionnel d'avant-projet puis du planning réel tel qu'il est aujourd'hui et concluons sur la répartition des tâches au sein de notre équipe.

Le planning établi au début du projet était le suivant :

	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
Choix projet															
	1h30	3h30	3h30 + 3h30	3h30 + 3h30	0h	3h30	7h00	0h	0h	7h + 3h	0h + 2h	3h30	0h + 2h	?	?
	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13
	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Figure 4 : Capture du planning prévisionnel, réalisé par nos soins

En revanche, celui-ci ne faisait ni apparaître la journée de formation, ni les dates de finales du concours. De plus, nous ne prévoyons pas de retravailler sur l'avant-projet.

Nous avons, à ce jour, respecté le planning suivant :

	Pas de séance
	Réaliser l'IHM
	Travail en équipe
	Code machine (G et ISO)
	Ladder et Codes M

Figure 5 : Légende des plannings, réalisée par nos soins

	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
Choix du projet															
	1h30	3h30	3h0 + 3h30	3h30 + 3h30	0h	3h30	7h00	0h	0h	7h + 3h	0h + 2h	3h30	0h	10h	4h00
	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13
	4h15 + 2h	0h	0h	5h + 3h	3h30 + 1h30	5h	3h30	5h + 2h	4h15	3h30	1h30	5h	3h30	3h	

Figure 6 : Capture du planning réel, réalisé par nos soins

Ainsi, le planning que nous avons esquissé nous a permis de travailler efficacement sur l'avant-projet tout en gardant un rythme soutenu, et cela malgré les périodes de distanciel.

Notons également que la contrainte temps a, pour des raisons sanitaires, quelque peu évolué. En effet, l'avant-projet devait initialement être rendu lors de la semaine 47 mais sa date de remise a été repoussée de 2 semaines, nous laissant ainsi le temps de le perfectionner.

Nous observons donc bien distinctement les **quatre phases** de notre projet :

- Lors de la première phase, nous apprenons en équipe les bases du fonctionnement de CN FANUC et des différents programmes qui nous sont demandés puis nous répartissons en duo afin de rendre l'avant-projet avant la semaine 49.
- Ensuite, ayant été sélectionnés, nous participons à la formation puis repartons avec une CN et un tas de nouvelles connaissances et pistes à appliquer dans notre avant-projet.
- Avant la finale, nous reprenons alors notre avant-projet de manière plus détaillé sans la contrainte temps afin de le perfectionner au mieux.
- Enfin, la finale se déroulerait lors de la 13^{ème} semaine le 30 mars à Evry.

Finalement, nous pouvons remarquer l'ébauche d'une répartition des tâches. Dans la partie suivante, nous verrons avec précision cette dernière en abordant la répartition de chaque partie technique.

1.4 Répartition des tâches

Avant tout, il est important de souligner le caractère exceptionnel de notre projet. Un concours implique une toute autre manière de pensée et une toute autre organisation. De ce fait, nous n'avons à ce jour, que peu travaillé de façon individuelle. En effet, l'esprit d'équipe est plus que primordial car il influe largement sur la suite des événements.

Ainsi, nous avons d'abord travaillé en groupe entier afin d'apprendre les rudiments des langages et logiciels mis à notre disposition, puis nous nous sommes, selon affinités, répartis en duos afin de nous focaliser sur l'IHM d'une part et le code machine (PMC et Programme ISO) d'autre part. Par la suite, selon les objectifs de la semaine, chacun se réparti le travail à effectuer au sein du duo. Finalement, le rôle de chacun pourrait se résumer à :

- [REDACTED] était chargée du développement de l'IHM et de la gestion des alarmes (partie IHM),
- [REDACTED] développait le programme ISO et assistait les modifications du programme PMC,
- [REDACTED] modifiait le programme PMC et assistait le développement du programme pièce,
- Victor était chargé du développement de l'IHM et de la rédaction de la notice explicative.

Passons désormais à la détermination du budget nécessaire au fonctionnement du projet.

1.5 Evolution des coûts, besoins financiers et matériels

Finalement, nous pouvons estimer le budget que nécessite ce projet.

Lors des prémisses du projet, nous prévoyions dans le budget la location d'un véhicule et d'un logement à Evry (sur le site de l'entreprise) en cas de sélection. Nous avions alors prévu :

- **Location d'une chambre d'hôtel** : ~ 65€ par personne (B&B Evry).
- **Location d'une voiture** : ~ 90€ par jour (SIXT Tours).

Soit, en comptant deux jours de location de voiture et une nuit à l'hôtel pour 5 personnes, un peu plus de **645€**.

Néanmoins, celui-ci a été amené à évoluer. Nous ne serions plus 5 mais 6 personnes à faire le voyage et celui-ci se ferait dans le mini-bus du SUAPS. Cela nous permettrait ainsi d'économiser les frais de location d'un véhicule supplémentaire.

En conclusion, nous estimons les coûts à **385 €** environ (en rajoutant une estimation de 60€ de frais de carburant).

Finalement, à travers cette partie, nous avons dressé les grandes lignes de notre projet et son organisation. Cependant, nous n'avons pas traité des détails techniques de l'avant-projet. Pour cela, analysons son cahier des charges puis déduisons-en les attendus techniques de chaque aspect de celui-ci.

2) Analyse du concours

Dans cette deuxième partie, nous traiterons de la composition interne d'une CN et décortiquerons, puis analyserons les attendus de chaque partie technique.

2.1 Explications générales et composition CN

Explications générales

Le concours, présente donc deux aspects : un aspect de longue durée : **l'avant-projet**, et un aspect de courte durée : **la finale**. De ce fait, nous traiterons de l'avant-projet comme s'il était le cœur du projet ; la finale ne servant qu'à mesurer le travail et les compétences acquises. Dans la suite de ce rapport, nous n'évoquerons donc que le travail effectué dans le cadre de cet avant-projet. Les bénéfices de la journée de formation et l'avenir de la finale ne viendront que plus tardivement à la lecture et seront organisés dans des parties dédiées.

Architecture d'une CN

Dans un premier temps, avant d'évoquer les fondements de la façon de programmer une CN, il est nécessaire de présenter son architecture machine.

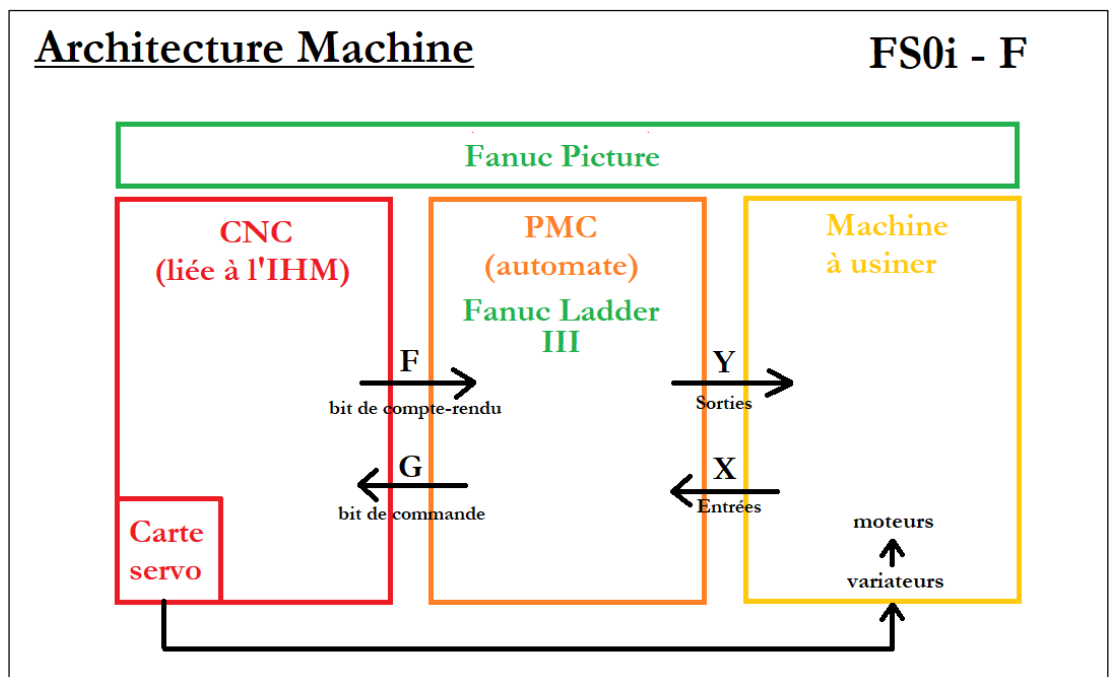


Figure 7 : Schéma présentant l'architecture machine d'une CN, réalisé par nos soins

Comme nous pouvons le remarquer, une CN peut se décomposer en trois différents blocs :

Le **premier bloc** correspond, pour simplifier, à « l'ordinateur » de la machine-outil à commande numérique. Sur le schéma, ce bloc s'intitule CNC, signifiant *Computer Numerical Control* ou plus simplement commande numérique par calculateur.

Le **deuxième bloc** se nomme le « PMC » (*Program Machine Controller*) . Il s'agit de l'automate programmable se situant physiquement derrière la CN ; il permet de relier les actions du clavier matriciel de la CN à des actions spécifiques de notre programme.

Enfin, le **dernier bloc** représente la machine-outil. Durant cette troisième édition, nous avons en charge une rectifieuse ; une machine de fraisage permettant d'enlever de la matière.

Ensuite, notons que chaque bloc peut communiquer avec son voisin ; la CNC avec le PMC à l'aide de bit de commande (G) et de compte rendu (F) et le PMC avec la machine à usiner grâce à ses entrées (X) et à ses sorties (Y).

Finalement, une commande numérique n'est qu'un simple outil que l'on rattache à une machine-outil nous permettant de connaître l'état et la position des éléments physiques la constituant. Généralement, il est d'usage de désigner par le terme CN ou CNC l'ensemble de la machine-outil à commande numérique (et ses différents éléments).

Remarque : La mention de « FS0i-F » sur le schéma n'est pas négligeable, c'est le modèle de la gamme sur lequel nous allons travailler.

Logiciels permettant la simulation de la CN

Comme expliqué plus tôt, lorsque nous travaillerons sur l'avant-projet durant la première phase, le travail se fera exclusivement par simulation. Introduisons alors les logiciels et leur rôle. Par ailleurs, la présence des deux logiciels FANUC PICTURE et FANUC LADDER III sur le schéma figure 7 nous renseigne sur leur domaine d'action.

Premièrement, **FANUC LADDER III** va permettre le contrôle de l'automate PMC par la création d'un programme LADDER. Celui-ci influera majoritairement sur les bits mémoires « R » de la CN, ses bits « M » (relatifs aux fonctions M³) ou encore sur les entrées et sorties de la machine à usiner.

³ Ces fonctions seront abordées plus en détail dans la partie relative aux attendus de l'avant-projet.

Néanmoins, notons qu'il existe de nombreux autres paramètres sur lesquels le logiciel peut influencer (voir figure 8).

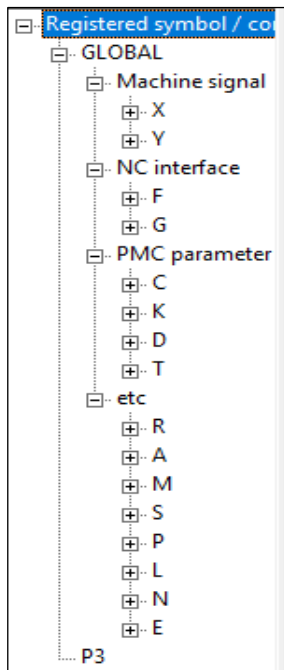


Figure 8 : Capture d'écran provenant du logiciel FANUC LADDER III

Deuxièmement, la mention du logiciel **FANUC PICTURE** nous informe que ce dernier peut jouer sur un grand nombre des éléments de la CN. En effet, celui-ci nous sera utile lors de la conception de l'interface homme machine : seul logiciel permettant le dialogue entre l'opérateur et la machine-outil ; il se doit d'être complet de manière à pouvoir dialoguer avec la CNC, le PMC ou encore directement la machine.

Enfin, le logiciel **CNC GUIDE** permet de simuler l'écran et le clavier matriciel de la CN (voir sous-partie suivante pour illustration). Nous avons choisi de ne pas le représenter sur le schéma car lorsqu'il y a présence d'une vraie CN, contrairement aux deux logiciels vu précédemment, il devient obsolète ; nous réalisons alors les manipulations sur la CN physique.

Remarque : Malgré son « inutilité » dans la troisième phase du concours (car ayant une CN physique à manipuler), nous avons continué de l'utiliser lors de l'amélioration de l'avant-projet car la simplicité qu'il offre nous faisait parfois gagner du temps.

Les différences entre CN simulée et CN réelle

Enfin, avant de traiter du sujet de l'avant-projet de cette édition, abordons les différences entre une CN simulée (CNC GUIDE) et une CN réelle.

▪ L'environnement.

L'environnement entre les deux CN reste le même visuellement parlant : CNC GUIDE reproduit à la perfection la fonction d'une CN. En revanche, l'avantage de ce dernier réside dans sa « mobilité » : il n'a pas besoin d'être branché et, lorsqu'il est allumé, les éléments le composant peuvent être déplacés selon leur utilité.

Vous trouverez figure 9 une capture d'écran du logiciel CNC GUIDE illustrant cet aspect.

Enfin, malgré tout, la CN physique offre la possibilité d'être connectée à une machine à usiner afin de voir en direct l'avancement du cycle. Dans notre cas, lors de la réception de la CN, nous avons une valise à disposition permettant de visualiser un mouvement.

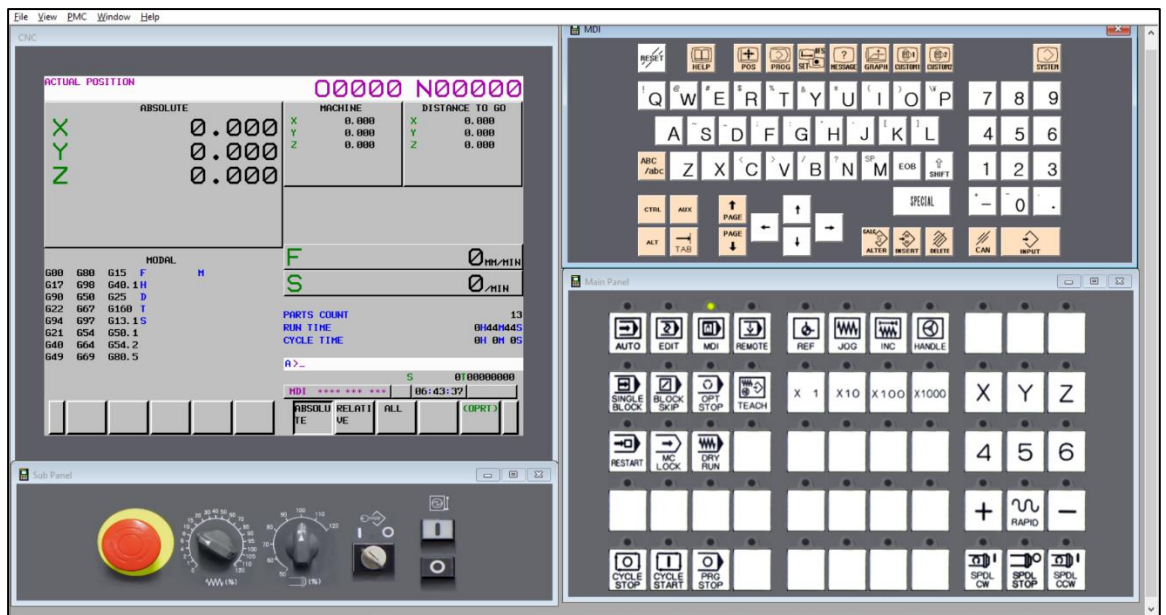


Figure 9 : Environnement CNC GUIDE, modulable grâce aux fenêtres

■ Le transfert de fichier.

La méthode de transfert des fichiers (on entend par « fichiers » le programme pièce, IHM ou encore le programme PMC) diffère elle aussi selon les deux options.

En simulation, il nous faut trouver le répertoire caché dans lequel CNC GUIDE est enregistré, créer une « machine virtuelle » (avec les bons paramètres) puis transférer les fichiers dans ce répertoire. Cela présente un avantage pour l'IHM car une fois le répertoire renseigné sous PICTURE, le projet s'exportera automatiquement dans celui-ci. En revanche, cela reste infiniment moins intuitif pour le transfert du programme pièce et PMC.

Sur la CN physique, il nous suffit de déposer tous les fichiers sur la carte mémoire PCMCIA⁴ puis de redémarrer la CN avec une combinaison de touche afin de les téléverser dans cette dernière.

Remarque : Ainsi, lorsqu'un test de l'IHM est à effectuer, nous préférons utiliser CNC GUIDE.

Passons désormais à l'analyse du sujet de l'avant-projet fourni.

⁴ Type de carte mémoire, PCMCIA signifiant *Personal Computer Memory Card International Association*.

2.2 Analyse de l'avant-projet

Traisons enfin du sujet de l'« avant-projet » de cette troisième édition. Celui-ci nous permettra par la suite de décomposer ses attendus en plusieurs thèmes majeurs présentant les points techniques importants : l'IHM, le programme PMC et le programme pièce (pour sujet complet, voir annexe 6).

Contexte de l'avant-projet

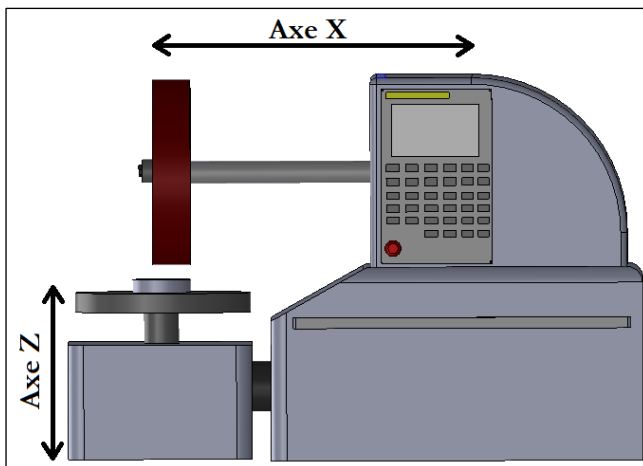


Figure 10 : Schéma fonctionnel d'une rectifieuse et de ces axes, réalisé par nos soins

Le sujet portait autour d'une rectifieuse (voir figure 10), une machine de finition permettant l'enlèvement de matière. Elle permettrait de rectifier des rondelles de réglage pour l'industrie aéronautique.

Elle présente donc 2 axes, X et Z, et quatre mouvements réalisables, l'avancement de l'axe X ou Z et la rotation de l'axe X ou Z.

Notons que l'axe X est relié à la meule et l'axe Z au plateau de la pièce. Ainsi, lorsque l'on évoquera un avancement de la meule ou une montée du plateau tournant, nous ferons référence respectivement aux mouvements d'avancement de l'axe X et Z.

Détails du cycle demandé

Le cycle de fonctionnement est détaillé dans la deuxième diapositive (page 2 de l'annexe 6) et est composé majoritairement de quatre phases.

- Durant la phase « zéro », l'opérateur réalise un **paramétrage manuel** des battements maximaux et minimaux de l'axe oscillant (axe X) via les écrans de contrôles. Ainsi, ceux-là sont mémorisés dans la CN afin d'être réutilisés par la suite.
- Après avoir validé ces positions, la machine-outil entame la phase d'**ébauche** et procède à un enlèvement de matière important. Pour cela elle réalisera l'avancement, le recul et la rotation de l'axe X à vitesse élevée

grâce aux positions préalablement enregistrées. Parallèlement, l'axe Z sera lui aussi en rotation et après chaque battement de la meule, le plateau s'élèvera.

- Ensuite, lors de la phase de **finition** , nous chercherons à réaliser un enlèvement de matière plus précis. La vitesse ainsi que la prise de passe seront donc plus faibles.
- Finalement, après avoir enlevé la majorité de la matière souhaitée, nous stopperons l'ascension de l'axe Z et poursuivrons la rotation et l'avance de la meule afin d'aplanir le surface de la pièce. Nous réaliserons ainsi l'ultime phase de **planage** .

Remarque : Le cycle sera détaillé par la suite plus précisément lorsqu'il sera question de le traduire en programme pièce (ou code G).

Attendus globaux : ce que nous devons rendre

Il nous sera demandé, entre autres, de réaliser les trois écrans constituant l'IHM ; d'écrire les programmes ISO paramétrés nécessaires au cycle de rectification ; de modifier le programme PMC pour prendre en compte les codes M et enfin, de gérer les alarmes relatives au cycle et aux erreurs de paramétrage.

Ainsi, comme résumé dans la diaporama 7 du sujet, nous devons rendre le programme NC GUIDE (autre appellation du logiciel CNC GUIDE) ; le programme FANUC PICTURE ; le programme Automate (ou PMC) ; une sauvegarde des paramètres machines et une notice explicative.

Notons par ailleurs que lors de la réception de l'avant-projet, il nous a été fourni la base du programme PMC que nous devons modifier. Nous devons donc développer entièrement l'Interface Homme-Machine ainsi que le programme pièce du projet.

Après cette succincte présentation du sujet, la suite sera décomposée suivant les attendus techniques de chaque partie constituant l'avant-projet.

2.3 Attendus - Interface Homme Machine (IHM)

Traitons désormais des attendus de chaque point technique. Concernant l'IHM, trois pages sont demandées :

- Une **page de suivi des opérations** contenant les informations nécessaires à la phase du cycle en cours, aux positions des axes, aux caractéristiques de la meule et le mode de fonctionnement de la CN.
- Une **page de paramétrage** dans laquelle nous pourrons entrer les différentes valeurs des vitesses voulues en fonction du mode de fonctionnement mais aussi le niveau de suppression de matière souhaité (tableau de la diapositive 5).
- Enfin, une **page de réglage des axes**, utilisée durant le réglage manuel des positions maximales et minimales des axes X et Z de la rectifieuse.

De plus, d'autres facteurs sont imposés : le diamètre de la meule et son usure par exemple. Il faudra donc veiller à les prendre en considération lors de la réalisation des différents écrans.

2.4 Attendus - Program Machine Controller (PMC) et sécurité

Pour ce qui est du programme PMC, nous avons, comme évoqué précédemment, la base de celui-ci. Il nous est simplement demandé de le modifier afin de prendre en compte les codes M impliqués dans le programme pièce.

Il nous faudra donc programmer les fonctions M permettant le pilotage ainsi que l'arrêt de la rotation (horaire et antihoraire) de la pièce et de la meule, l'aimantation (ou non) de la pièce et la mise en marche (ou à l'arrêt) de l'arrosage.

Concernant la sécurité, nous devons prendre en charge dans le code global la gestion des différentes erreurs de paramétrage ou encore l'appui sur le bouton d'arrêt d'urgence. Par exemple, nous devons veiller à ce que la prise de passe et les vitesses ne soient pas nulles ou encore que celles de la phase d'ébauche soient bien strictement supérieures à celles de la phase de finition.

2.5 Attendus – Programme pièce (ou programme ISO)

Le programme-pièce, quant à lui, correspond à l'écriture de chaque action à réaliser lors du cycle. Il est à écrire au complet et permet la réalisation entière du cycle, de la première phase à la dernière (nous omettons la phase « zéro »). Il servira au pilotage des différents axes et des différentes actions et pourra solliciter des sous-programmes.

Remarque : A l'origine, un programme pièce était écrit en code G. C'est depuis la normalisation de son code par la norme « ISO 6983-1:1982 » qu'il est souvent nommé « code ISO ».

De ces attendus, nous pouvons désormais évoquer l'état actuel du concours de chaque aspect technique, l'avenir de celui-ci ainsi que les problèmes rencontrés au cours de sa réalisation.

3) Avancement du concours

Enfin, traitons de l'état de chaque partie technique : l'IHM, le programme pièce, le programme PMC et la gestion des alarmes. Nous mettrons en évidence l'état final de chaque partie de l'avant-projet puis les comparerons aux parties rendues lors de la sélection si nécessaire. Les apports de la journée de formation et les modifications effectuées seront ainsi mis en évidence. Nous finirons par traiter des problèmes rencontrés et de l'avenir du concours.

A ce jour, nous sommes dans l'avant-dernière phase ; la phase de préparation de la finale. Nous avons ainsi été sélectionnés à travers l'épreuve de l'avant-projet et avons réalisé la formation de l'entreprise. Durant cette partie, nous détaillerons donc l'avancement du concours de notre point de vue.

3.1 L'interface Homme-Machine (IHM)

Dans cette sous-partie, nous allons évoquer, et brièvement présenter, l'état de l'IHM à ce stade du concours. Nous mettrons ainsi les apports de la journée de formation et les modifications apportées à l'IHM en avant par rapport à sa version antérieure.

3.1.1 Page de suivi des opérations

La première page de cette IHM se trouve être la page de suivi des opérations. Dans celle-ci, nous détaillons la phase actuelle du programme (Ebauche, Finition ou Planage) ainsi que les positions des axes X et Z. Il nous est également demandé de faire apparaître les informations concernant la meule ; sa vitesse de rotation et son diamètre. Les modes de fonctionnements sélectionnés sur la CN doivent aussi apparaître.

De ces attendus, nous avons dressé la page suivante (figure 11) dans laquelle les informations détaillées ci-dessus apparaissent bien. Détaillons désormais la réalisation de chaque élément. Il sera alors question d'indiquer les éléments utilisés sur le logiciel.

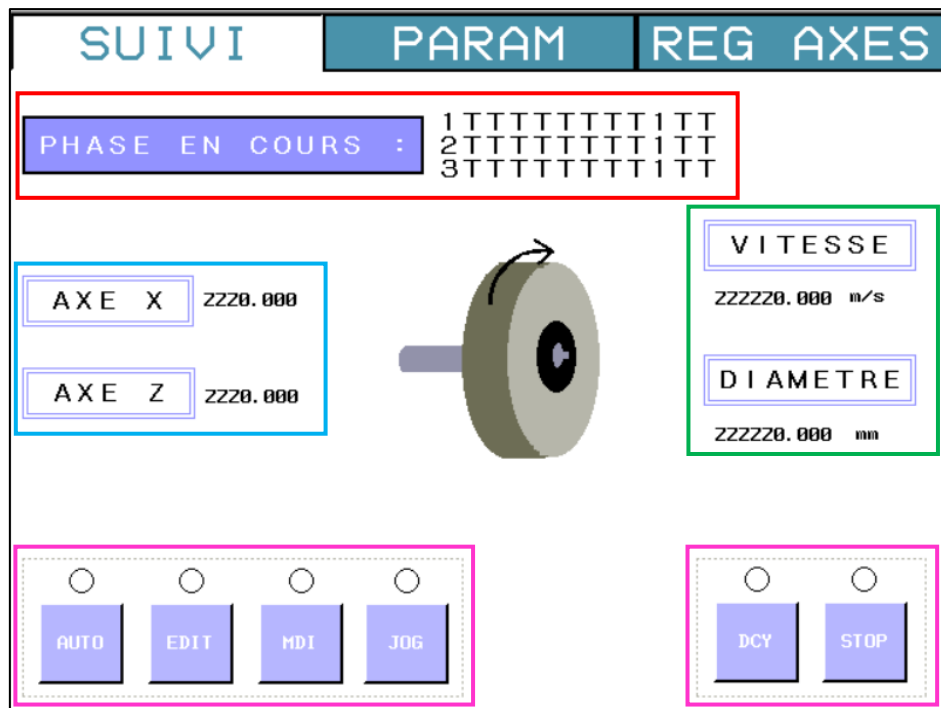


Figure 11 : Capture de la page 1 de l'IHM : Suivi des opérations

Remarques :

- Les « onglets » de page en haut de l'écran sont communs à toutes les pages ; ils permettent de se déplacer dans le projet d'IHM.
- Les rectangles de couleurs ne sont pas présents dans l'interface finale mais nous serviront à traiter des éléments qu'ils regroupent.

Abordons désormais la construction de chaque élément technique et leur technologie : ce sur quoi ils agissent.

Les onglets de haut de page

D'abord, lors de la création d'une nouvelle page sur FANUC PICTURE, il est nécessaire d'associer à cette dernière l'élément « PScrn » qui permet de la définir en tant qu'écran. C'est en quelque sorte la structure de page qui va nous permettre de l'associer à un nom et à un type (écran, écran « pop-up », etc...).

Ensuite, afin de pouvoir choisir l'écran souhaité, nous avons fait le choix de les représenter l'arborescence sous forme d'onglet : « SUIVI », « PARAM » et « REG AXES ». Pour cela, nous avons simplement utilisé des objets « ChgScrn » dans lesquels nous avons mis le nom de la page correspondante.

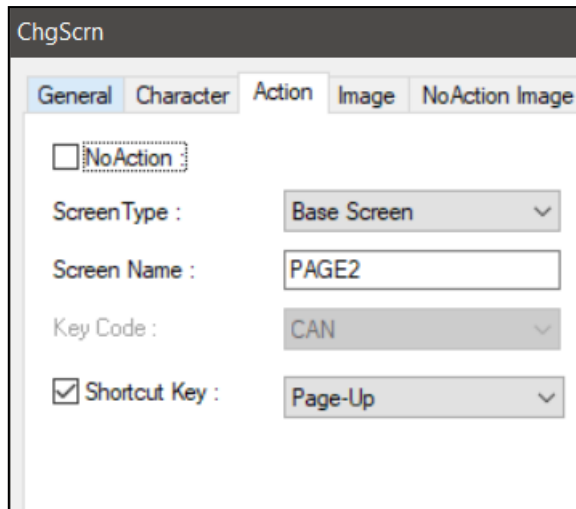


Figure 12 : Extrait du paramétrage de l'action d'un « ChgScr »

Par exemple, vous trouverez ci-contre une capture d'écran indiquant l'action que réalise le « ChgScr ». Ici, celui-ci permet d'aller à la page 2, soit la page de paramétrage. Il s'agit donc de l'action de l'onglet « PARAM ».

Remarque : Nous avons aussi associé cette action à une touche du clavier de la CN d'où la présence de « Shortcut Key » coché et le choix de « Page-Up ».

Cela nous permettra de nous déplacer d'une page à une autre par le biais du clavier et non pas seulement de l'écran tactile.

Par ailleurs, lors de la réalisation de l'onglet de la page active (par exemple l'onglet « SUIVI » de la figure 1), il nous faut cocher la case « no action » afin d'éviter un quelconque conflit.

Les phases du cycle (rectangle rouge)

Concernant les différentes phases du cycle, nous avons simplement utilisé une zone de texte (soit l'objet « Label ») et un « TxtMsg ». Ce dernier permet, grâce à une lecture de paramètre, de donner la phase actuelle de la CN : lorsque le PMC envoie l'ordre de la phase actuelle, le message texte changera.

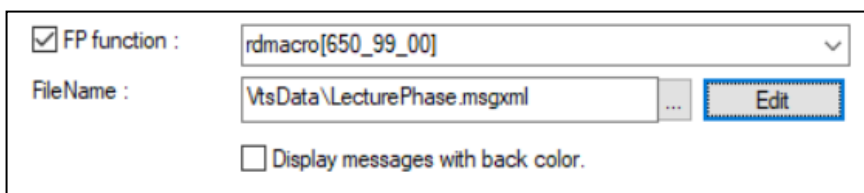


Figure 13 : Extrait de la lecture de l'adresse #650 pour lecture de phase

Ici, nous lisons l'adresse #650 ; et suivant le texte changera selon le programme « LecturePhase »

	Number	Bit set	Char color	
1	1			EBAUCHE
2	2			FINITION
3	3			PLANAGE
4	4			CYCLE TERMINE

Figure 14 : Changement affichage TxtMsg en fonction des valeurs

Ensuite, suivant la valeur de l'adresse #650, lue précédemment, nous affichons le message correspondant.

Par exemple, lorsque #650 = 1, FPTxtMsg1 affichera « EBAUCHE ».

Il sera donc nécessaire de faire évoluer ces variables au cours du cycle⁵.

Remarque : Lorsque nous parlerons de « variable mémoire de la CN », nous ferons référence à une variable de type #XXX (où les X sont des chiffres), généralement entre 500 et 900 (celles qui sont disponibles à la modification). De même, lorsque nous parlerons de « variable mémoire du PMC », nous sous-entendrons les variables (ou codes) de la forme RXXX. Leur existence est indépendante ; la PMC et la CN ne peuvent, comme vu plus haut, communiquer entre elles que par des bits de commande et de compte rendus, c'est l'IHM qui pourra mettre en lien les spécificités de chacune.

Mode de fonctionnement de la CN (rectangles rose)

Chaque ensemble est composé d'un bouton en mode « Momentary » (il n'est actif que lorsque qu'il est appuyé) et d'une lampe indiquant le mode enclenché.

Le détail technique des éléments de cette partie reste globalement simple : nous allons aller lire dans les lampes des bits mémoires, R511.0, R511.1 et R511.2 (ici, on parle de bit car nous spécifions l'adresse voulue : le mot R511 au bit 0 par exemple pour R511.0), et allumer la lampe correspondant au bit à l'état haut.

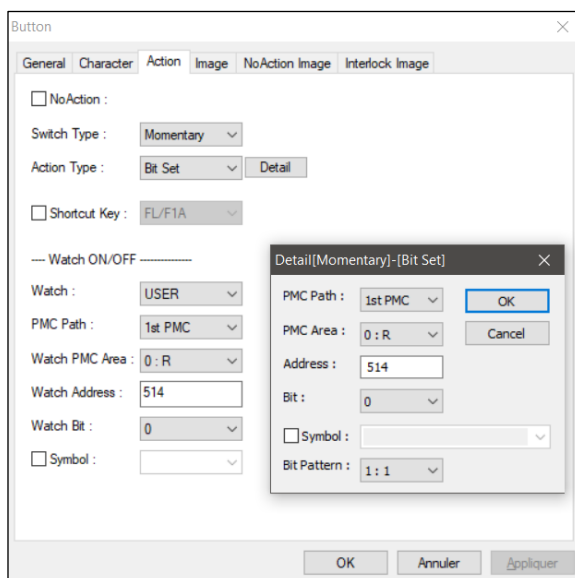


Figure 15 : Capture de la programmation d'un bouton, ici le bouton départ cycle

Si l'opérateur appuie sur les boutons ; on va également activer le mode de la CN correspondant grâce à l'action du bouton « Action Type », ici « Bit Set » pour donner une valeur à un bit, et au détail de l'action.

Ici, lorsque le bouton est actif, le bit R514.0 est mis à l'état haut.

Ces adresses mémoires seront ensuite traitées dans le programme PMC afin de correspondre à l'appui du mode sur le clavier de la CN.

⁵ Vous pourrez retrouver cela dans l'annexe concernant le code G commenté (numéro 8).

De plus, nous avons ajouté les boutons « JOG », « départ cycle » (autrement appelé DCY) et « arrêt cycle » (surnommé STOP) afin de permettre à l'opérateur de gérer le cycle entier grâce à l'IHM.

Affichage des coordonnées X et Z (rectangle bleu)

Ensuite, l'affichage des coordonnées en temps réel de X et Z était également demandé. Pour ce faire, il nous a fallu poser deux objets « Value » permettant l'affichage d'un nombre obtenu. Leur nom est « FPIInput1 » et « FPIInput2 »⁶.

Le paramétrage de ceux-ci repose sur trois principales caractéristiques : l'**action** permettant de lire la donnée, son **type** et son **affichage** (nombre de chiffre après la virgule dans notre cas). Voici le paramétrage final de l'axe X:

- Une fonction de lecture et d'écriture « absolute[1] » permettant de recopier la valeur de l'axe X dans l'afficheur en temps réel.

- Un type « double » correspondant au format des coordonnées.

- Un nombre de décimal après la virgule de 3 ce qui permet d'afficher les coordonnées précises (car les coordonnées sont de la forme 0.000).

Figure 16 : Programmation d'un affichage d'axe, ici axe X

Nous avons par la suite réalisé la même chose pour l'axe Z à l'aide de la fonction « absolute[3] » et des mêmes paramétrages de type et d'affichage.

Affichage informations meule (rectangle vert)

Afin d'afficher les informations concernant la meule, nous devons utiliser des objets « Numeral/String » ou « NDisp » permettant d'afficher le contenu de l'adresse mémoire #502 (vitesse meule) et #503 (diamètre meule).

Pour la vitesse, il faudra aller chercher la valeur dans le tableau de paramètre en fonction de la phase du cycle en cours.

⁶ Nous ne pouvons pas changer les noms sur le logiciel, d'où l'aspect brutal de certains objets.

Le diamètre, quant à lui, suit une loi affine : à chaque fin de cycle, son diamètre diminue de 0.01mm. Cet aspect ne sera détaillé mais est présent lors de la programmation de l'adresse #503 en annexe 8.

Ainsi, comme nous venons de le voir (et comme nous le verrons encore), chaque élément de l'IHM se doit de dialoguer avec les autres parties du programmes. Il nous a ainsi fallu répertorier chaque adresse mémoire (de type #XXX et RXXX) afin de les assigner plus tard dans les programmes. La majorité des adresses utilisées dans nos programmes sont répertoriées dans l'annexe 5.

3.1.2 Page de paramétrage

La deuxième page de cette IHM est la page de paramétrage ; c'est la page dans laquelle nous pourrions rentrer les paramètres relatifs aux vitesses, à la prise de passe voulue et à la surépaisseur dans les modes ébauche ou finition.

Nous obtenons après réalisation cette page :

The screenshot shows the 'PARAM' tab of the IHM interface. It contains two tables for parameter settings and two input fields.

PARAMETRE EBAUCHE			PARAMETRE FINITION		
Prise de passe	2220.0	mm	Prise de passe	2220.0	mm
Vitesse axe vertical	2220	mm/mn	Vitesse axe vertical	2220	mm/mn
Vitesse bras oscillant	2220	mm/mn	Vitesse bras oscillant	2220	mm/mn
Surépaisseur finition	2220.0	mm	Tempo de planage	2220	s
Vitesse rtn meule	2220	m/s	Vitesse rtn meule	2220	m/s
Vitesse rtn plateau	2220	tr/mn	Vitesse rtn plateau	2220	tr/m

Below the tables, there are two input fields:

- Diamètre meule :** 2222220.0 mm
- Vitesse bras oscillant :** 2222220.0 mm/mn

On the right side, there are two buttons:

- Ouvrir le gestionnaire d'erreur
- Réinitialiser le diamètre de la meule

Figure 17 : Capture de la page 2 de l'IHM : Page de paramétrage

A l'image de la page 1, présentons les « objets » de FANUC PICTURE utilisés lors de la réalisation cette page.

Le tableau de paramètres à remplir (rectangle orange)

Nous avons choisi de présenter la saisie des paramètres dans un « tableau » (qui n'a d'un tableau que le nom) comme sur la dispositive 5 du sujet. Ainsi, les lignes et les intitulés ne sont que des illustrations et des zones de texte et seul les paramètres sont des objets à modifier.

Les objets utilisés pour renseigner les valeurs de chaque paramètre sont des « TInput » qui sont des « claviers virtuels ». Ainsi, lorsque nous voudrions changer la valeur d'un paramètre, nous pourrions le faire directement sur l'IHM par le biais d'un clavier numérique qui apparaîtra.

Techniquement parlant, chaque « TInput » utilise les fonctions de lecture et d'écriture d'une variable. Ainsi, dans la colonne des paramètres d'ébauche nous retrouverons les variables #600 à #605 et dans la colonne de finition celles allant de #700 à #705

ReadFunction :	<input type="text" value="rdmacro[601_99_0]"/>
WriteFunction :	<input type="text" value="wmacro[601_0_0_99_0]"/>
CheckFunction :	<input type="text"/>

Figure 18 : Exemple d'une fonction de lecture et d'écriture

L'action de lecture est possible grâce à la fonction « rdmacro » (*read macro*) et celle d'écriture grâce à « wmacro » (*write macro*). La valeur modifiée est ici celle de la variable #601 qui correspond ici à la valeur de la vitesse de l'axe vertical lors de la phase d'ébauche.

Les paramètres supplémentaires (rectangles bleu)

Pour la praticité du programme, nous avons décidé de rajouter les paramètres de vitesse de bras de planage, diamètre de la meule et la possibilité de réinitialiser celui-ci.

Ainsi, le diamètre initial de la meule est à renseigner dans le paramètre ajouté puis à valider grâce au bouton de réinitialisation. Cela permet donc de dissocier le diamètre de la meule de son usure après chaque cycle. En d'autres termes, le diamètre de la meule de la page 1 (contenu dans l'adresse #503) est le **diamètre en temps réel** et celui de la page 2 (contenu dans #707) le **diamètre initial**. Cette modification nous sera grandement utile lors du développement du code G et de la prise en charge de l'usure de la meule à la fin de chaque cycle (en effet, nous n'utiliserons que la variable #503).

Techniquement, les paramètres sont eux aussi des « TInput » et le bouton de réinitialisation est un « Button » couplé à une fonction de « FPScript »⁷ permettant de copier les valeurs des différentes variables des meules (la valeur de #707 copiée dans #503).

Remarque : Le « gestionnaire d’erreur » sera abordé dans la partie relative à la sécurité et aux alarmes car permettant de vérifier les erreurs de paramétrage.

3.1.3 Page de réglage des axes

Cette dernière page concerne le paramétrage manuel des axes. Dans celle-ci, nous devons observer les coordonnées des axes X et Z et, lorsque l’opérateur a déterminé les positions maximales de X et Z et minimales de X, sauvegarder ces valeurs dans les adresses mémoires #800 (Xmin), #801 (Xmax) et #802 (Zmax). Une représentation schématique du bras oscillant est également demandée.

La page réalisée sur FANUC Picture est la suivante :

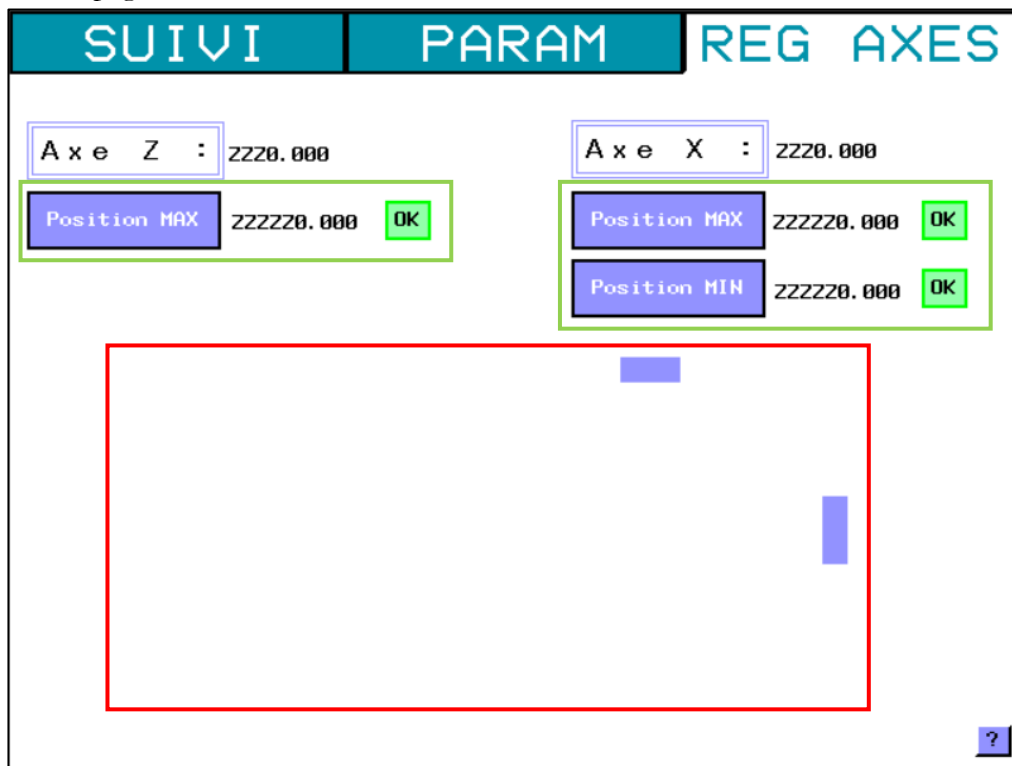


Figure 19 : Capture de la page 3 de l'IHM : Page de réglage des axes oscillants

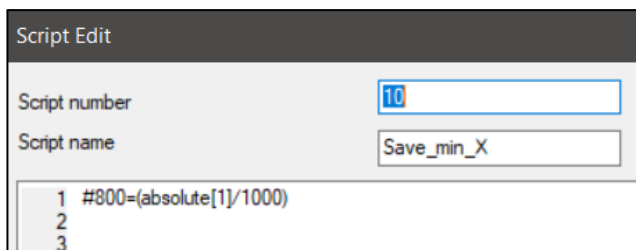
⁷ Nous les aborderons plus en détail ces fonctions dans la partie de la gestion des alarmes.

Dans cette dernière page, nous affichons les axes Z et X en temps réel grâce à la méthode vue plus haut (page 1) et, lorsque l'on appuie sur un bouton de validation, sauvegardons la valeur actuelle de l'axe dans une variable mémoire.

Sauvegarde des axes (rectangles verts)

Afin de sauvegarder les valeurs, nous avons trois objets : un label nous indiquant ce que représente la valeur, un objet « Value » qui va lire et afficher les variables mémoires de #800 à #802 et un bouton de validation qui va nous permettre de stocker la coordonnée de l'axe correspondant dans la variable souhaitée.

En fait, ici, rien de nouveau. Les objets « Value » utilisent la fonction « rdmacro » et observe l'adresse souhaitée et le bouton lance une fonction « FPScript » permettant de copier dans l'adresse la valeur de l'axe (voir figure 20).



Lorsque l'utilisateur appuie sur le bouton vert « OK » de saisie de Xmin, on lance le script numéro 10 intitulé Save_min_X qui va copier dans la variable #800 la valeur de X.

Figure 20 : Exemple du stockage de la valeur de Xmin dans #800

Remarque : la division par 1000 de la valeur finale sert à faciliter l'affichage.

Représentation schématique du système (rectangle rouge)

Enfin, nous devons faire apparaître un schéma simple de rappel du système. Avant de traiter du schéma, abordons les « rectangles violet » composant la sélection.

En effet, ceux-ci sont des « jauges » ajoutées à chaque axe afin de voir les mouvements. Par exemple, lorsque l'axe Z monte, la jauge « Y » se remplit, et lorsque l'axe X recule la jauge « X » se vide.

Concernant le schéma, bien qu'il ne soit pas visible sur la capture de la page, il apparaît bien lors du démarrage de la CN.

	Number	Bit set	Back color
1	1		PhotoData\PAGE3_1.bmp
2	2		PhotoData\PAGE3_3.bmp

L'objet utilité est une « ImgLamp », ou « Image Lampe » : elle va fonctionner comme les « TxtMsg » vu précédemment ; suivant la valeur d'une variable, elle affichera une image correspondante.

Figure 21 : Capture d'un paramétrage d'objet "ImgLamp", ici pour la page 3

Voici les images que celle-ci peut afficher :

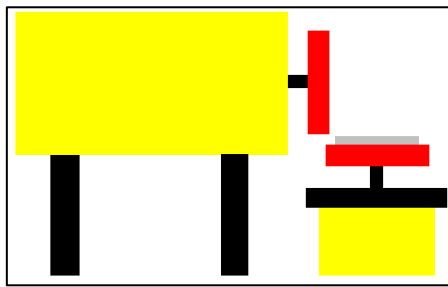


Figure 22 : Image "PAGE3_1.bmp" utilisée dans la page 3, réalisée par nos soins

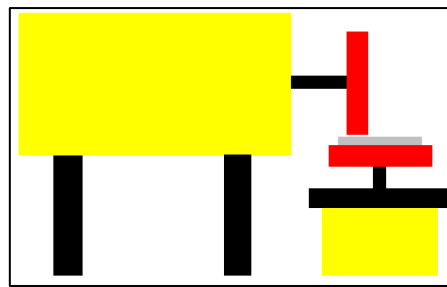


Figure 23 : Image "PAGE3_3.bmp" utilisée dans la page 3, réalisée par nos soins

Ainsi, lorsque la valeur à l'adresse #515 sera égale à 1, l'ImgLamp affichera l'image PAGE3_1.bmp et lorsqu'elle sera égale à 2, l'image PAGE3_3.bmp.

Cela nous permettra de voir les positions minimales et maximales de l'axe X.

Remarque : Nous verrons par la suite que même si cet ajout visuel ne semble pas très « poussé » (seulement deux images d'une piètre qualité visuelle), c'est le mieux que l'on puisse faire en termes de praticité.

3.1.4 Les écrans « bonus »

Comme indiqué dans le sujet, nous devons dresser les trois pages demandées. En revanche, nous avons aussi pris la liberté de rajouter des écrans « bonus ».

La page d'accueil

Dans un premier temps, nous avons décidé de rédiger une page supplémentaire d'accueil afin de faire apparaître les logos de l'entreprise, de l'IUT ainsi que nos noms et le libellé du concours. Cette page se lancera à chaque premier allumage de la CN. Pour cela, lors du renseignement de sa place, nous l'avons déclaré comme page de « démarrage » (start-up screen sur figure 24).

En revanche, afin de ne pas afficher celle-ci lors de la manœuvre du cycle sur l'IHM, nous n'en n'avons pas fait la page « principale ».

Ainsi, lorsque la CN se lance, nous arrivons bel et bien sur la page d'accueil, mais, une fois passée, nous n'y retournons pas.

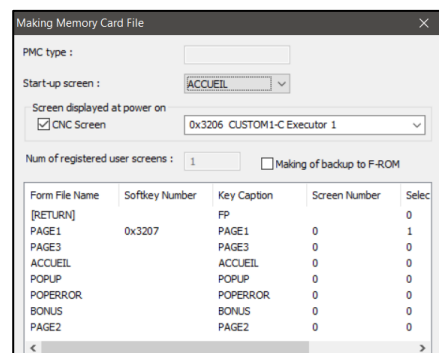


Figure 24 : Capture de l'onglet de la gestion des pages dans FANUC PICTURE

La page réalisée est la suivante :

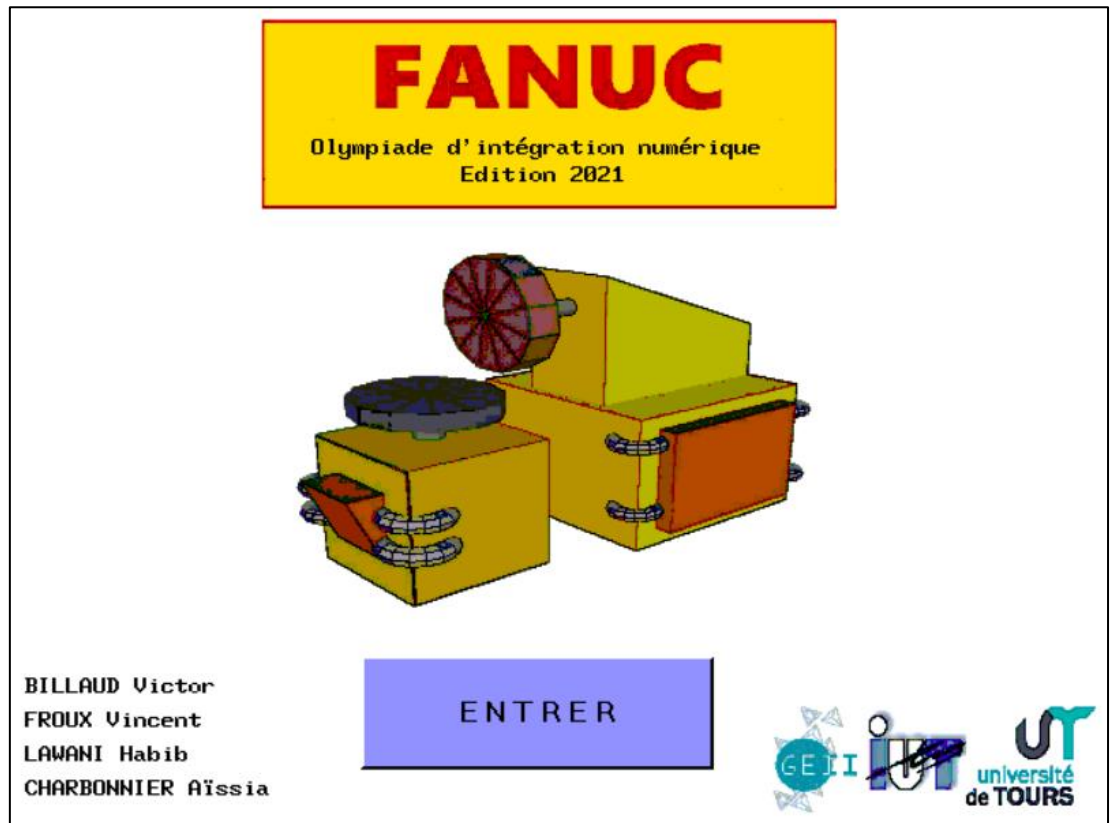


Figure 25 : Capture de la page d'accueil de l'IHM

Techniquement, cette page ne comporte que des labels de texte, des images et un bouton « ENTRER » nous dirigeant vers la page principale, la page de suivi des opérations.

Les différents « pop-ups »

Deuxièmement, nous avons décidé de créer des fenêtres de « pop-up » qui seraient sollicitées à différents moments.

Nous avons notamment la fenêtre de gestion des erreurs de paramétrage évoquée plus haut (voir figure 27) qui permet de vérifier, par un procédé que nous évoquerons plus tard, les conflits entre chaque paramètre ainsi que la fenêtre de détection d'arrêt d'urgence (voir figure 28) permettant d'indiquer l'appui de ce dernier.

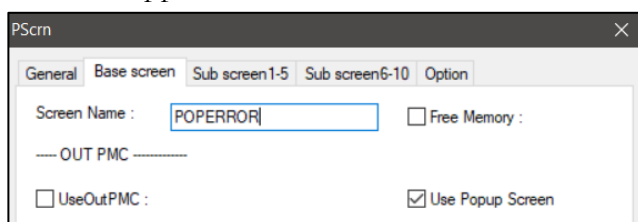


Figure 26 : Déclaration d'un écran "pop-up" sur FANUC PICTURE

Afin de créer un écran « pop-up », il suffit de déclarer dans sa structure (élément permettant de convertir sur FANUC PICTURE une simple page en un écran) la mention « Use Pop Screen ».

Les pop-ups réalisés sont les suivants :



Figure 27 : Capture d'une page pop-up de l'IHM : Détection arrêt d'urgence

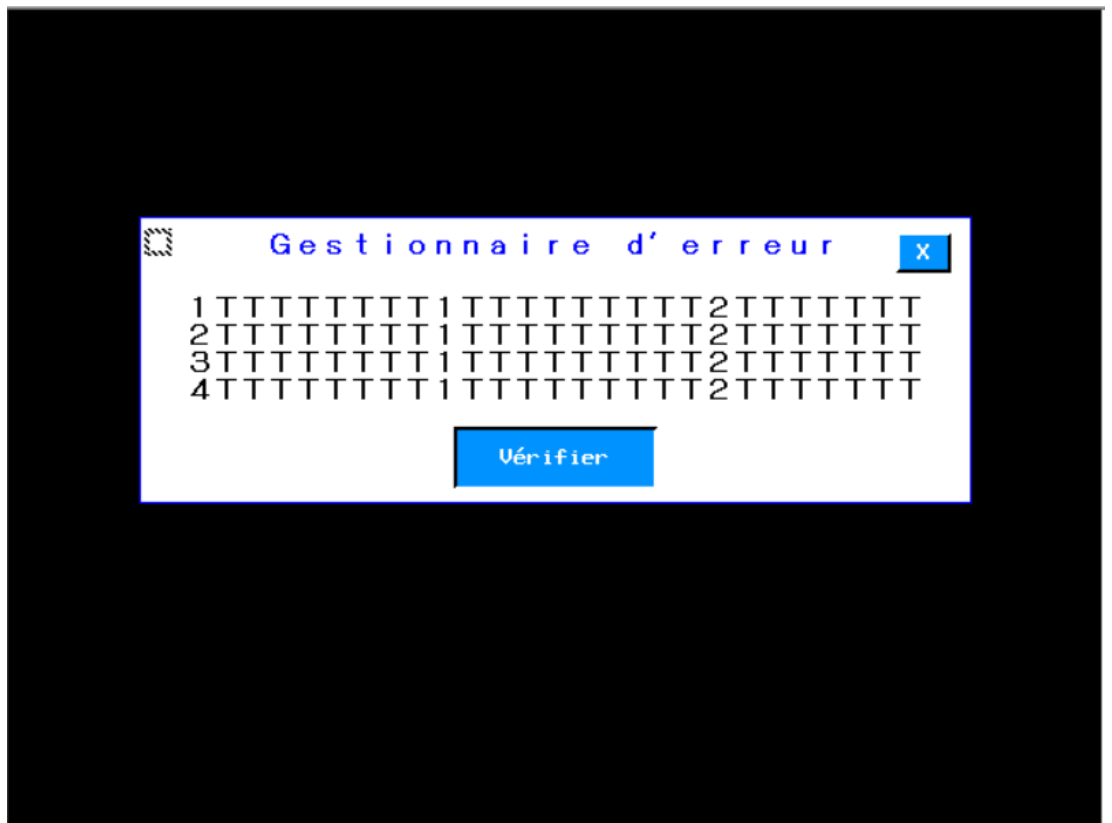


Figure 28 : Capture d'une page pop-up de l'IHM : Gestionnaire d'erreurs

Remarque : Le fond noir derrière un pop-up indique son champ d'action, il se superposera sur l'écran en cours de la même manière que lorsqu'il a été conçu.

Enfin, après avoir traité de l'état actuel de l'IHM, revenons aux bases de celle-ci et mettons en évidence son évolution à travers les différentes phases du projet.

3.1.5 L'évolution de l'IHM

Enfin, relevons l'évolution de l'IHM au fil de l'avancement du concours.

L'IHM à l'état de l'avant-projet

Avant tout, nous devons faire apparaître l'état de l'IHM à l'heure de la remise de l'avant-projet ce 4 décembre 2020.

En outre, les écrans réalisés étaient les suivants :

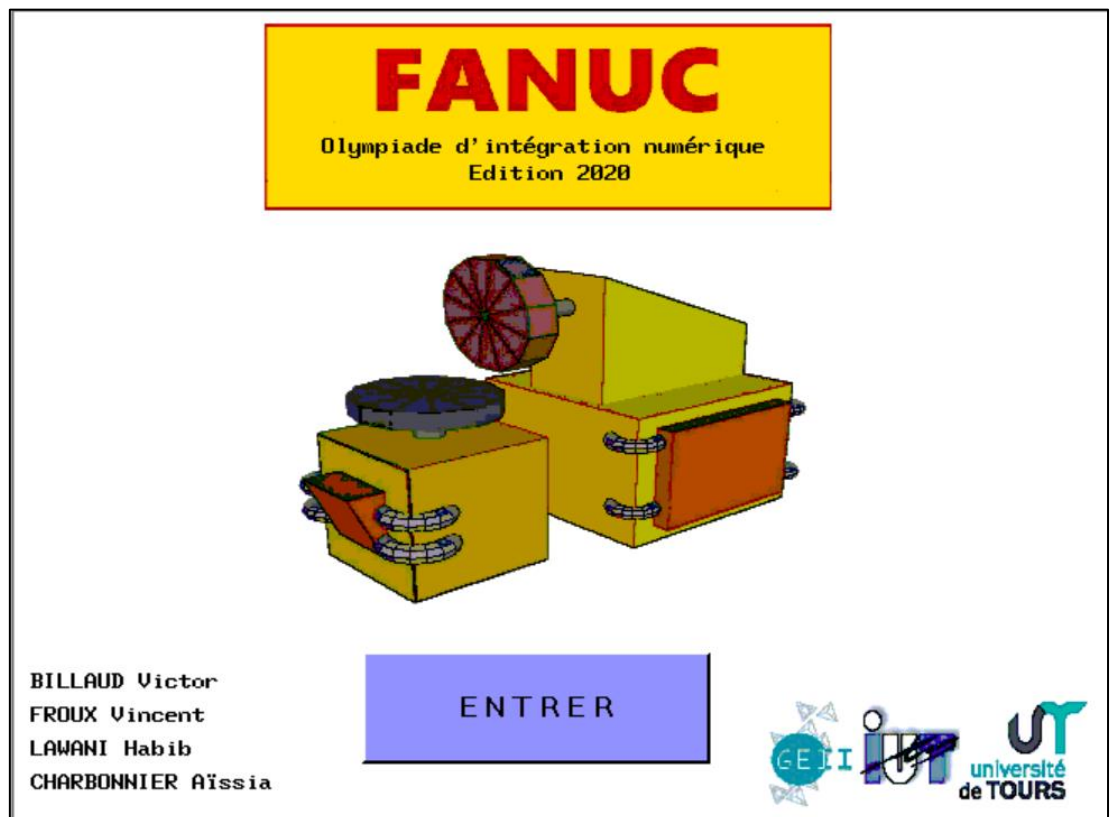


Figure 29 : Capture page accueil IHM - avant sélection

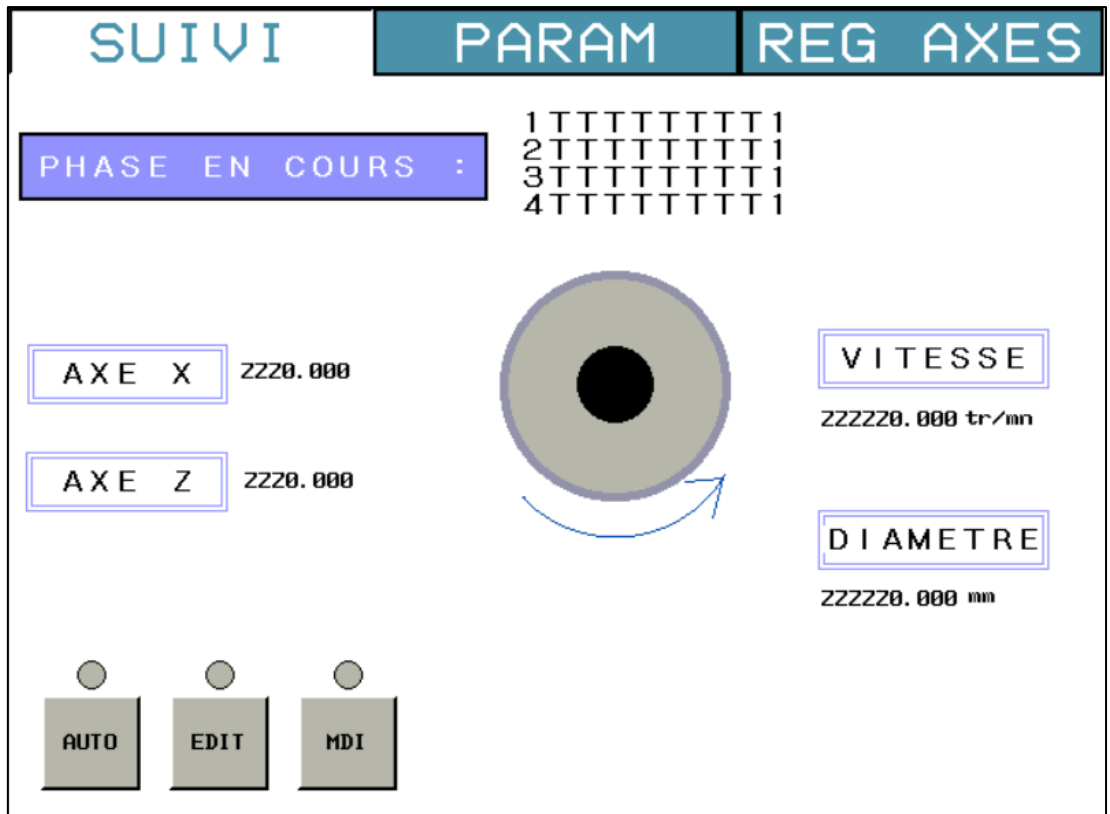


Figure 30 : Capture page 1 IHM - avant sélection

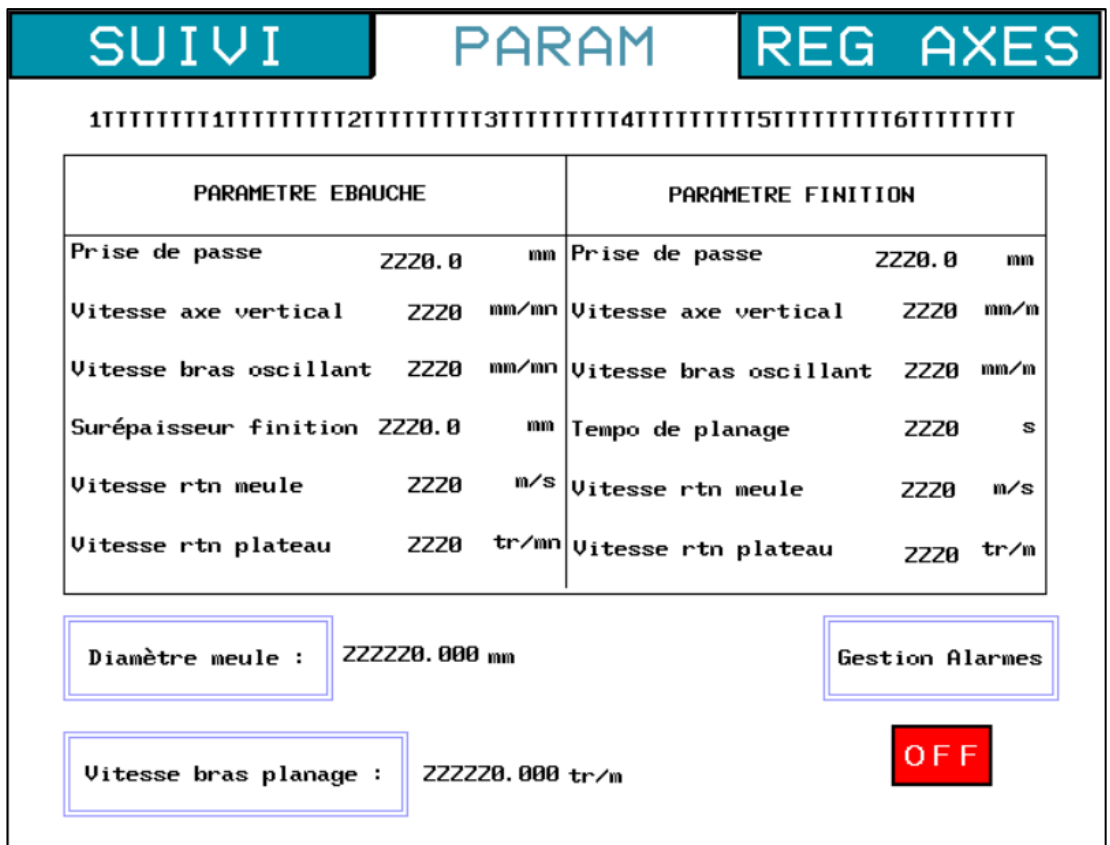


Figure 31 : Capture page 2 IHM - avant sélection

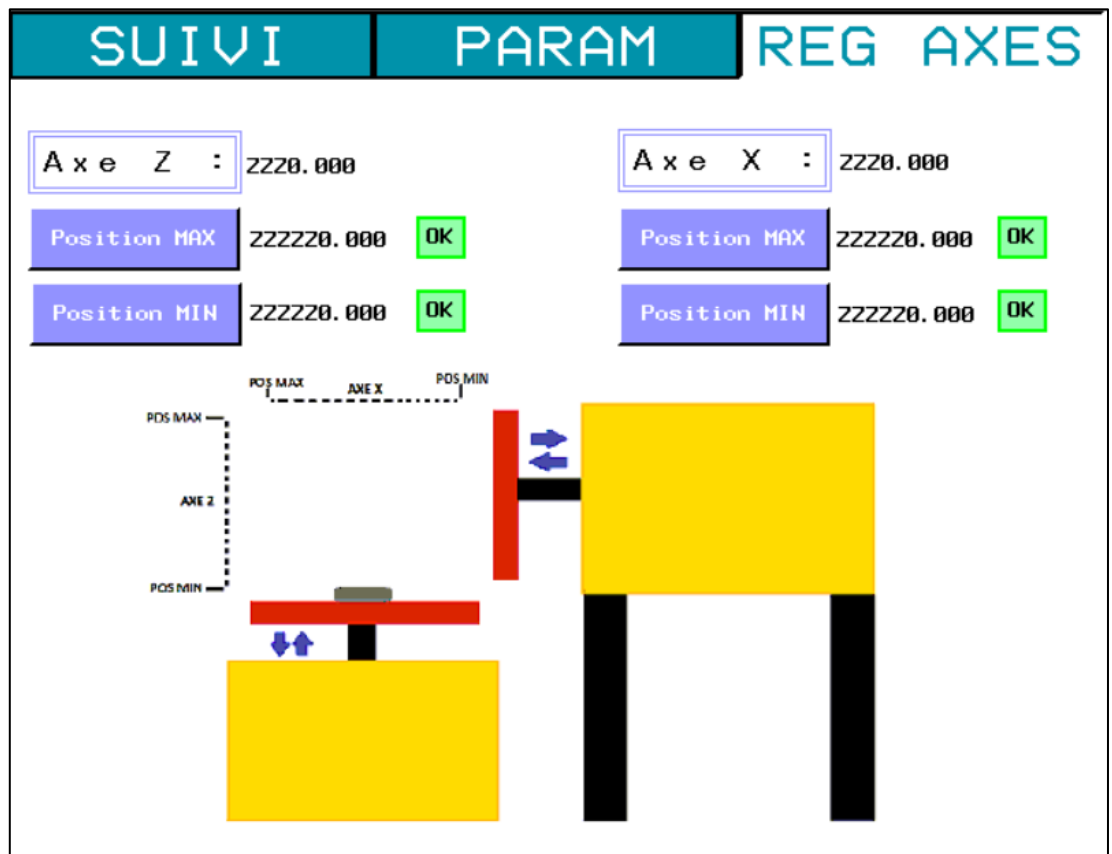


Figure 32 : Capture page 3 IHM - avant sélection

Les apports de la journée de formation - IHM

Evoquons désormais les points relevés concernant l'IHM lors de la formation réalisée ce 13 janvier.

Durant cette journée et avec le recul de notre avant-projet, nous avons soulevé de nombreux problèmes et améliorations à effectuer, par exemple :

- Les boutons de la première page n'étaient pas associés avec les actions de la CN,
- L'apparition de la valeur minimale de l'axe Z ne nous était d'aucune utilité,
- Nous devrions mettre en place un système de pop-up afin de ne pas polluer les écrans de contrôle.

De plus, il a été pour nous important de rajouter un aspect plus professionnel dans notre réalisation. Pour cela, nous avons remodeler une meule en 3D à l'aide des outils du logiciel sur la page 1, ajouter une animation dans la page 3 afin que l'on comprenne l'état actuel des axes X et Z et différents écrans, utiles à notre programme.

Remarque : L'organisation de cette partie technique peut paraître redondante néanmoins c'était, d'après nous, la solution évitant le plus de répétition inutile.

3.2 Le programme PMC

Ensuite, présentons les modifications apportées au programme PMC fourni à l'origine dans le sujet de l'avant-projet. Nous traiterons dans un premier temps du programme ladder rendu ce 4 décembre dernier puis de sa version la plus récente.

3.2.1 Le programme PMC à l'heure de la sélection

Lors de la remise de notre programme PMC celui-ci comportait la gestion des différents codes M (se référer au sujet annexe 6), et les bits de liaison avec l'IHM. Cependant nous avons pu noter après la relecture de celui-ci quelques petits problèmes à corriger comme un léger conflit entre les boutons de la CN et ceux de l'IHM (les bits de communications sont traités par la PMC, voir figure 33) et un problème au niveau du bouton STOP qui ne fonctionnait plus.

Liaison PMC – IHM

La liaison entre PMC et IHM s'est majoritairement faite grâce à la déclaration de nouvelles variables mémoires R. Nous pouvons par exemple citer :

119	BAU	BOOL	R0520.0	1 Bit d'arrêt d'urgence
120	LEDStop	BOOL	R0515.5	1 LED de l'arrêt stop
121	LEDStart	BOOL	R0515.4	1 LED du lancement start
122	LEDJog	BOOL	R0515.3	1 LED du mode JOG
123	LEDMDI	BOOL	R0515.2	1 LED du mode MDI
124	LEDEdit	BOOL	R0515.1	1 LED du mode Edit
125	LEDAuto	BOOL	R0515.0	1 LED du mode auto
126	BStop	BOOL	R0514.1	1 Bouton arrêt cycle IHM
127	BStart	BOOL	R0514.0	1 Bouton lancement cycle IHM
128	BJOG	BOOL	R0512.0	1 Bouton Manuel IHM
129	BMDI	BOOL	R0511.2	1 Bouton MDI IHM
130	BEdit	BOOL	R0511.1	1 Bouton Edit IHM
131	BAuto	BOOL	R0511.0	1 Bouton Auto IHM\n

Figure 33 : Bits de liaisons ajoutés permettant la communication entre PMC et IHM

Les bits de liaison ainsi ajoutés permettent à l'utilisateur de communiquer avec la machine directement depuis l'écran de l'IHM et non les touches de la CN. Ces bits ont ensuite été programmés dans le programme déjà existant afin de les prendre en compte au même ordre qu'un appui sur un bouton physique.

L'acquiescement des fonctions M

La méthode que nous utilisons était la suivante :

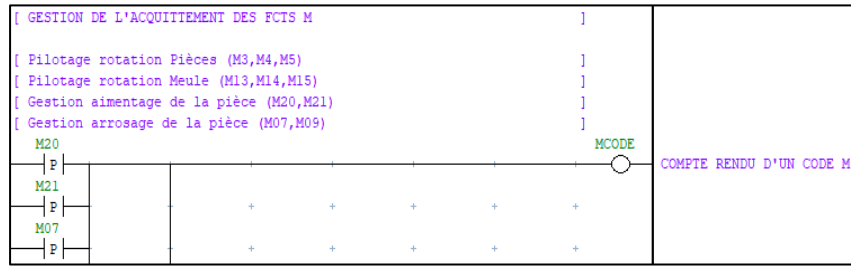


Figure 34 : Exemple d'acquiescement d'une fonction M

Ainsi, lorsqu'un code M était appelé dans le programme ISO, il envoyait un compte rendu (bit F) lors de son activation (« P » = front montant). Il fallait donc prendre en charge toutes les fonctions M utilisées dans notre programme.

Ensuite le compte rendu servait à l'acquiescement afin de mettre fin au code M.

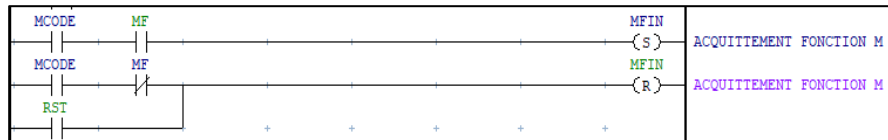


Figure 35 : Exemple d'un set-reset de l'acquiescement d'une fonction M

L'assignation des fonctions M

L'assignation des fonctions M à des commandes ou à des indicateurs était réalisée de façon simple : nous relierons les fonctions M aux fonctions de la CN. Par exemple, le code ISO utilise la fonction M07 relative à l'arrosage et durant celle-ci, la lampe du bouton 34 de la CN restera allumée. Cela permet un affichage personnalisable indiquant les différentes actions du cycle.

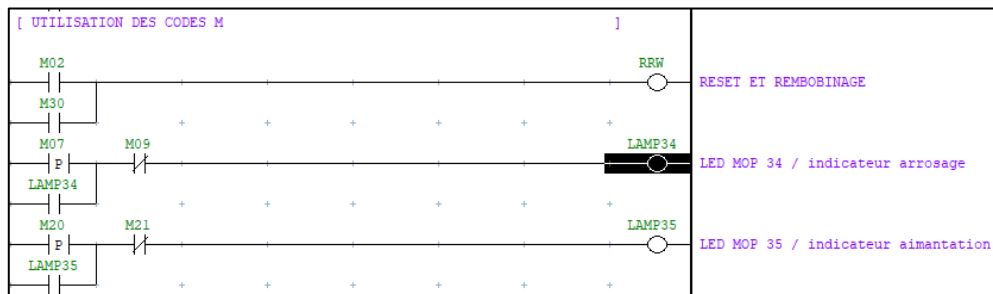


Figure 36 : Exemple d'association d'un indicateur à une fonction M

Nous utilisons ainsi 2 indicateurs lumineux de la CN lorsque les fonctions d'arrosage et d'aimantation sont actives.

3.2.2 Les apports de la journée de formation - PMC

Lors de cette journée, le majeur changement a été pour nous la remise de documentation technique. En effet, notre formateur M. Gautier nous a remis de nombreux documents sous format PDF tel que le manuel de maintenance, traitant en majorité de la communication du PMC avec la CN et l'IHM, ainsi qu'un PDF des différentes fonctions réalisables par notre machine où un chapitre est consacré à celle de la CN.

De plus, à l'image des points relevés dans la partie IHM, nous avons également relevé des problèmes et des améliorations potentielles, notamment :

- Améliorer l'acquittement des codes M,
- Allumer les indicateurs des touches « + » et « - » de la CN lors des déplacements de l'axe X,
- Modifier la programmation LADDER du Bouton STOP,
- Trouver et régler le conflit des boutons entre l'IHM et la CN (page 1).

De ces nouvelles pistes, le programme PMC a connu une évolution et est à ce jour fonctionnel et plus optimisé qu'avant.

3.2.3 Les améliorations effectuées

Dans un premier temps, persistait un problème dans l'attribution de l'action du bouton « STOP » de l'IHM. Leur ancien code est rappelé ci-dessous :

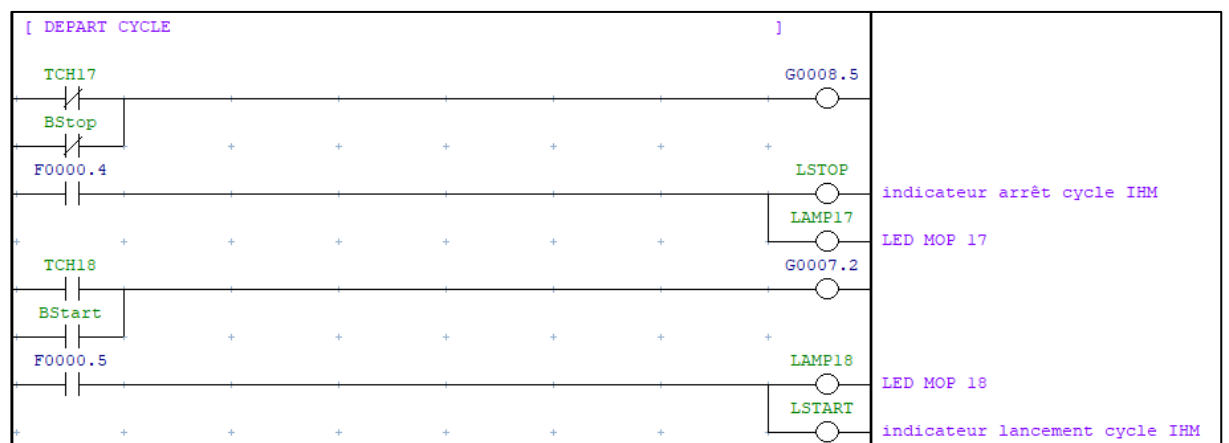


Figure 37 : Problème programmation LADDER bouton stop

En outre, la mise en parallèle du bouton « STOP » de la CN (TCH17) et du bouton de l'IHM causait le dysfonctionnement du programme.

Ainsi, nous obtenons alors :

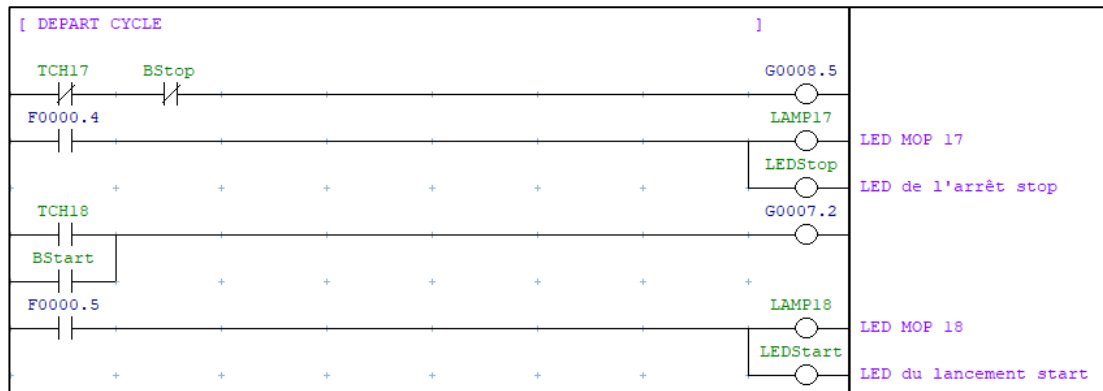


Figure 38 : Solution de la programmation LADDER du bouton stop

Ici, le changement opéré est fondamental : lorsque les deux boutons étaient en parallèle, ils formaient une structure « OU » et il nous était impossible de stopper le cycle de la machine-outil. En effet, si l'on appuie sur un des deux boutons l'autre fait barrage et cela implique la continuité du cycle.

La mise en série de la détection des deux boutons implique ainsi une structure « ET » et permet ainsi de désactiver le bit de commande G0008.5 dès lors qu'un bouton est appuyé.

Dans un deuxième temps, nous avons appris lors de la journée de formation l'existence d'un bit F, le F0001.3, qui avait pour rôle d'être au niveau haut à chaque acquittement d'une fonction M.

De ce fait, l'acquittement des fonctions M s'est simplifié :

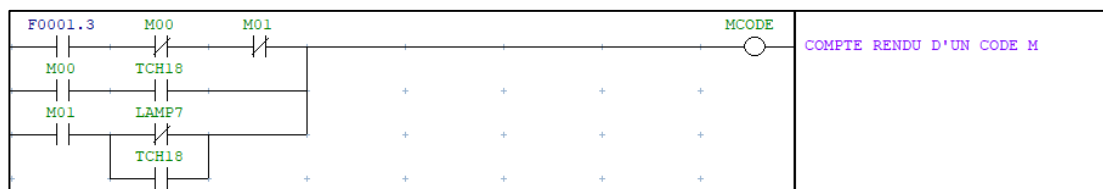


Figure 39 : Nouvelle méthode d'acquittement des fonctions M

Ainsi, grâce à cette amélioration, le programme LADDER se réduit considérablement et nous ne sommes pas obligés d'acquitter manuellement chaque fonction M.

Dans un dernier temps, nous avons décidé d'afficher l'avance et de recul de l'axe X, et donc de la meule, grâce aux touches « + » et « - ». Pour cela, nous utilisons les bits de compte rendu F correspondant à l'avance ou au recul de l'axe.

Cela donne alors :

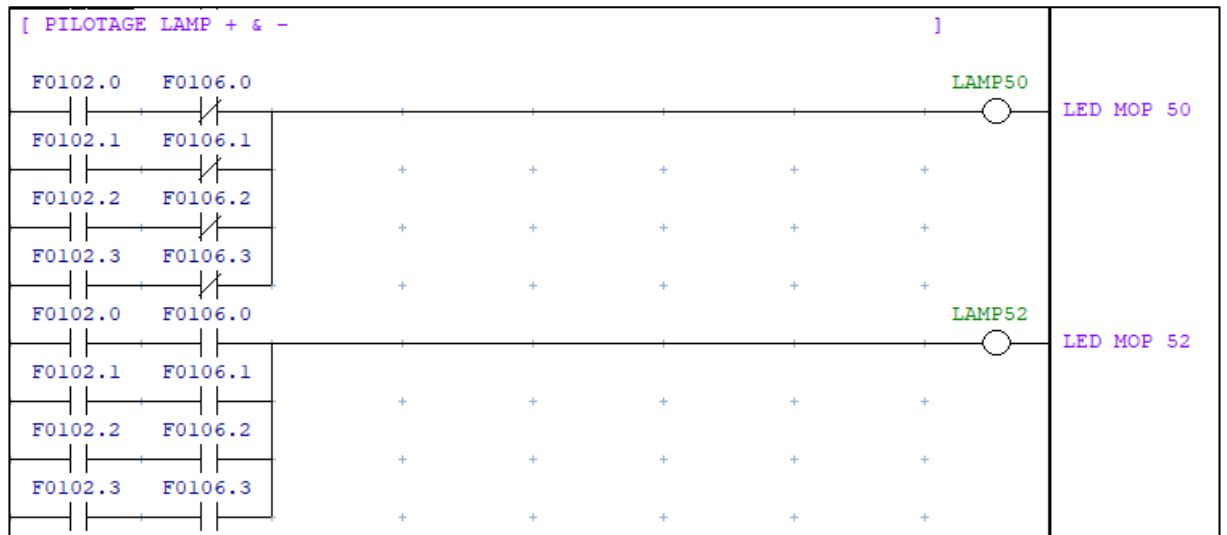


Figure 40 : Programmation des touches "+" et "-" de la CN

Notons que les indicateurs « LAMP50 » et « LAMP52 » correspondent respectivement aux LED des touches 50 et 52 de la CN : les touches « + » et « - ».

Enfin, notons que le programme PMC a été également modifié dans une optique de sécurité, cet aspect sera abordé dans la partie dédiée aux alarmes.

3.3 Le programme pièce

Dans cette troisième sous-partie, nous allons présenter le fonctionnement du programme pièce réalisé. N'ayant été que peu modifié, nous ne le comparerons pas à sa version antérieure. En outre, nous indiquerons les seuls problèmes rencontrés et mettrons en avant les solutions trouvées.

3.3.1 Compréhension « physique » du cycle

Lorsque le cycle débute, l'opérateur procède à un réglage manuel des axes grâce au mode « JOG ». Durant ce réglage, nous enregistrons la position minimale de X et les positions maximales de X et Z.

Par la suite, lorsque l'opérateur envoie la consigne de début de cycle, ce dernier rentre dans de la phase **d'ébauche**. Nous lançons l'aimantation et l'arrosage de la pièce, puis débutons la rotation antihoraire de l'axe Z (axe de la pièce). Puis, en fonction d'un pas calculé précédemment, nous faisons monter l'axe Z afin d'enlever cette quantité de matière par une rotation horaire et un avancement de l'axe X (axe de la meule) à sa position maximale. Ensuite, nous faisons de nouveau monter l'axe Z en fonction du pas calculé puis nous ramenons la meule à sa position de départ (Xmin) sur l'axe X sans stopper la rotation de la meule. Nous obtenons ainsi un cycle dit « carré »⁸. Nous réalisons cela ensuite à 10 reprises afin d'enlever la prise de passe nécessaire. Cette première phase participe à l'enlèvement de la majorité de la matière.

Dans un deuxième temps, le cycle passe dans la phase de **finition** , dans laquelle les axes X et Z réalisent la série de mouvements décrite précédemment de façon identique à vitesse réduite. Durant ce cycle, l'enlèvement de matière sera donc de plus faible valeur.

Finalement, nous procéderons à la phase de **planage** consistant en une rotation et une oscillation de la meule entre Xmax et Xmin à vitesse faible afin de « lisser » la pièce. Une fois cette ultime phase terminée, nous stoppons l'aimantation et l'arrosage de la pièce puis l'indiquons à l'opérateur.

Remarque : Durant l'entièreté du cycle, nous devons prendre en charge l'affichage du diamètre, des vitesses de rotation de la meule ainsi que des différentes phases en cours.

⁸ La nature « carrée » du cycle est évoquée dans le sujet de l'avant-projet. D'après notre compréhension, la montée de la pièce, l'avance de la meule, la montée la pièce et le recul de la meule présente la seule possibilité de ce type de cycle.

3.3.2 La traduction du cycle en organigramme

Du cycle précédemment décrit, nous en avons dressé des organigrammes. Par ailleurs, nous avons décidé de fonctionner à l'aide de **sous-programme** afin de pouvoir utiliser le même sous-programme durant l'ébauche et la finition, et ainsi ne pas surcharger le programme complet.

Nous nous retrouvons alors avec 3 programmes pièce : le programme principal O1000, le sous-programme servant à l'ébauche et à la finition O1001 et le sous-programme servant au planage O1002⁹.

Les organigramme décrivant leur déroulement sont les suivants :

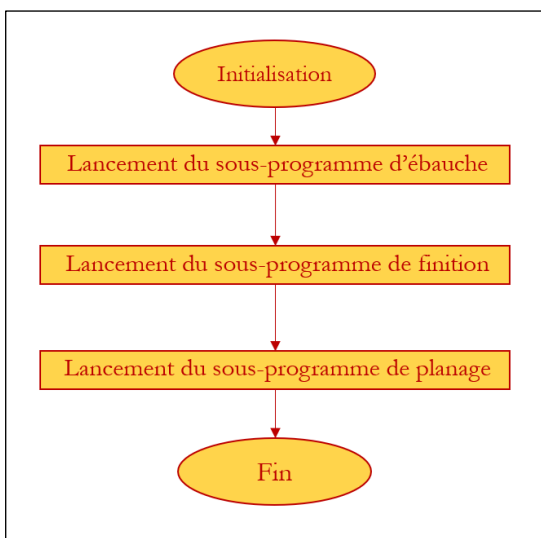


Figure 42 : Organigramme du programme principal O1000

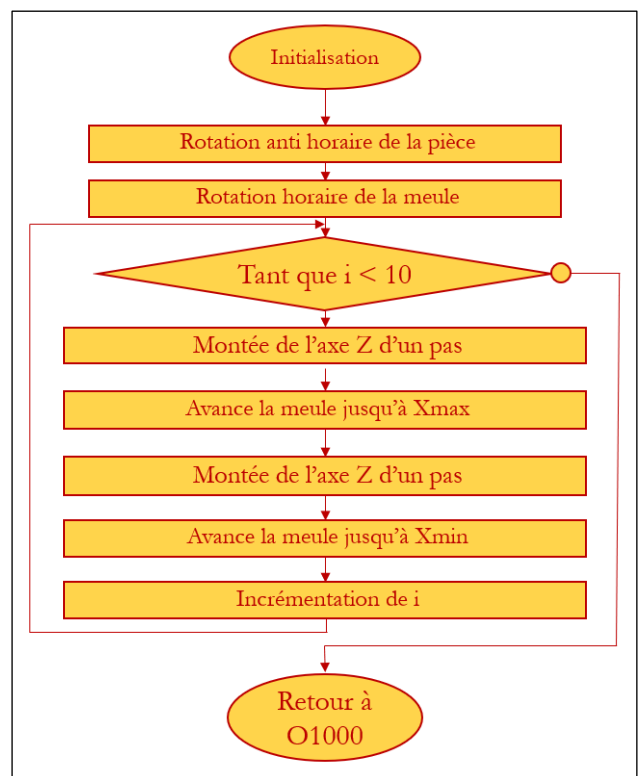


Figure 43 : Organigramme du sous-programme d'ébauche et de finition O1001

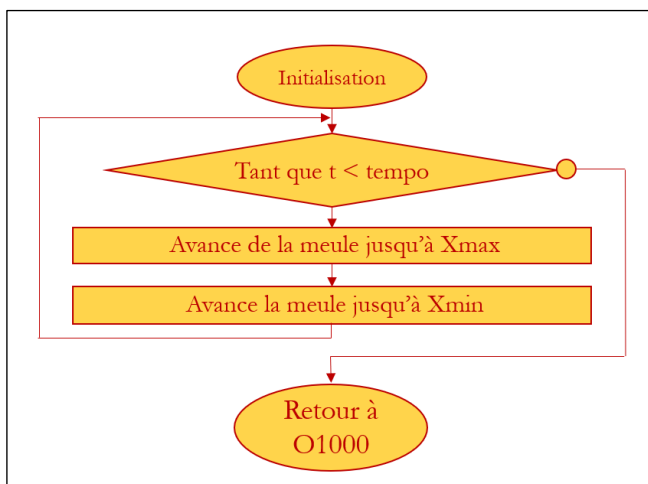


Figure 41 : Organigramme du sous-programme de planage O1002

Nous remarquons ainsi que le programme aura pour objectif d'initialiser les variables et les états des fonctions M mais surtout de lancer les différents sous-programmes au fur et à mesure de l'avancement du cycle.

⁹ Ces appellations « barbares » sont normalisées, nous n'avons pu faire autrement.

3.3.3 Quelques remarques sur le programme

A la suite de cette « traduction », nous avons rédigé notre programme pièce dans les grandes lignes puis ajouté les éléments lui permettant de dialoguer avec l'IHM tels que l'affichage du diamètre ou encore de la vitesse de la meule évoqués plus haut. Le programme complet est disponible commenté en annexe 8.

Dans cette sous-partie, nous allons traiter de certains points qu'il nous semble important d'évoquer tel que la gestion d'appel de sous-programme, la notion de boucle « tant que » en code G ou encore tout simplement l'ordre traduisant un déplacement.

Les déplacements (rectilignes)

Lors de la conception de notre programme, il nous a fallu apprendre l'utilisation et la structure du code G. Afin d'illustrer ce dernier, les codes G00 et G01 étaient les plus souvent évoqués ; il s'agit de codes d'avance (ou de recul).

G00 est un code G indiquant un déplacement à vitesse maximale. Ainsi, lors de son utilisation, nous devons lui préciser seulement les coordonnées qu'il doit atteindre. Dans le programme principal, nous l'utilisons comme suit :

G00 X#800;

Ici, nous souhaitons aller en X à la valeur contenue dans l'adresse #800. La coordonnée en Z n'étant pas précisée, la machine-outil ne se mouvra qu'en X.

G01 est quant à lui un code d'avance dans lequel nous devons spécifier sa vitesse en plus des coordonnées. D'après son utilisation dans le programme de planage :

G01 X#801 F#706;

Ici, nous souhaitons, comme précédemment, nous déplacer à la coordonnée #801 en X, mais cette fois-ci à la vitesse contenue dans l'adresse #706.

Ainsi, tout code G présente une action différente et a une structure qui lui est propre.

Appel de sous-programme

Lors du déroulement de notre cycle, le programme principal devra faire appel aux sous-programmes O1001 et O1002. Pour cela, nous devons utiliser un code G spécial.

Lors de l'appel du sous-programme O1001, nous utilisons le code suivant :

G65 P1001 A#601. B#602. I#604. J#605.; (par exemple)

- Le code G65 permet de lancer le sous-programme nommé après « P », ici 1001, et de lui passer des paramètres.
- Les mentions A, B, I et J permettent l'ajout de paramètre. Ici nous lui passons les adresses 601, 602, 604 et 605.

Remarque : Il s'agit ici du code lançant la phase d'ébauche ; nous prenons les adresses des variables mémoires de la première colonne du tableau.

Afin d'appeler le sous-programme de planage, nous utilisons la ligne suivante :

M98 P1002 L1;

- La fonction M98 permet, quant à elle, de lancer le sous-programme le 1002 sans passer de paramètre. N'ayant pas passé de paramètre, le programme utilisera les variables globales de la CN.
- La mention L1 permet d'indiquer que nous n'appelons qu'une fois seulement le sous-programme O1002.

Boucle « tant-que »

Concernant les deux boucles « while » de sous-programme, celles-ci ont la structure suivante :

WHILE[#900 LT #606]DO1

...

END1

Avec le séparateur « LT » signifiant « LESS THAN » (inférieur à) entre la variable à comparer et la valeur voulue.

Remarque :

- Lors du sous-programme O1001, nous la comparerons à une variable valant 10 afin de réaliser les 10 cycles.
- Grâce à un calcul de vitesse prenant en compte la distance entre Xmin et Xmax réalisé lors de l'initialisation et la durée de temporisation demandée. Ainsi, quel que soit le nombre d'aller-retour que l'on décide de faire nous pourrions respecter la temporisation en faisant varier la vitesse.

Enfin, concernant le développement du programme, celui-ci n'a pas de logiciel dédié. Il faut seulement que les fichiers contenant les programmes ISO n'aient pas d'extension quelconque. Sur Windows, ils seront appelés « fichier » et n'ont pas d'icônes.

O1000	03/03/2021 10:49	Fichier	1 Ko
O1001	03/03/2021 12:35	Fichier	1 Ko
O1002	03/03/2021 12:35	Fichier	1 Ko

Figure 44 : Exemple des programmes ISO sous Windows

Afin de les rédiger, nous avons utilisé le logiciel NOTEPAD++ mais nous aurions très bien pu utiliser la CN (via CNC GUIDE ou en vrai).

Lorsque ces programmes sont importés dans la CN, ils sont visualisables grâce à l'onglet « PROG » (voir figure 45).

```
PROGRAM<WORD>                                00000 N00000
//CNC_MEM/USER/PATH1/
O1000 (BG:READ ONLY)
O1000 M01 ;
G21 ;
#650=0 ;
#810=#801 ;
#503=#707 ;
M7 ;
M20 ;
G90 G00 X#800 Z#801 ;
;
#650=1 ;
G65 P1001 A#601. B#602. C#603.
I#604. J#605. ;
;
#650=2 ;
G65 P1001 A#701. B#702. C#703.
I#704. J#705. ;
;
#650=3 ;
M98 P1002 L1 ;
#803=#810 ;
#820=[#820+1] ;
#803=[#803+0.005] ;
#503=[#503-0.005] ;
M05 ;
M15 ;
M9 ;
M21 ;
M30 ;
%
```

Figure 45 : Visualisation d'un programme pièce (ici O1000) sur la CN

3.4 La gestion des alarmes

Finalement, abordons l'ultime partie technique : la gestion des alarmes. Nous traiterons dans un premier temps des modifications apportées à l'IHM pour prendre en charge les différentes erreurs possibles puis les modifications du PMC résultantes de celles-ci. Nous ne ferons pas apparaître d'évolution à proprement parler car cette partie, à l'image du programme pièce, n'a pas subi de modification durant la troisième phase.

3.4.1 L'Interface Homme Machine - sécurité

Concernant les modifications apportées à l'IHM, nous avons simplement ajouté l'affichage d'un arrêt d'urgence et un gestionnaire de vérification d'erreur.

L'affichage de l'arrêt d'urgence

Dans un premier temps, nous souhaitions afficher l'état d'arrêt d'urgence à l'aide d'une fenêtre « pop-up » mais, tout en nous rendant compte que cela était impossible, nous avons préféré prioriser un affichage simple.

Lorsque la CN est en arrêt d'urgence, la page 1 affiche alors :

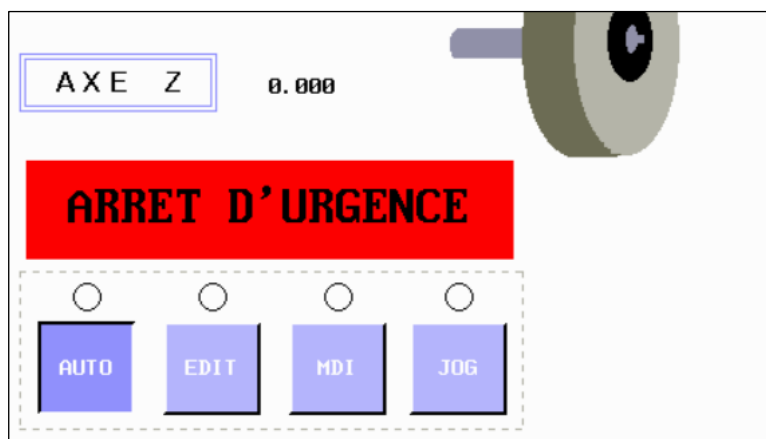


Figure 46 : Présence de l'arrêt d'urgence de la CN

Techniquement parlant, l'élément utilisé est une « Lamp ». Afin que cette dernière « s'allume » lors d'un arrêt d'urgence, nous lui faisons seulement lire le bit mémoire R520.0 que le programme PMC nous fournit lors d'un ordre d'arrêt d'urgence.

Le gestionnaire d'erreur

Dans un second temps, nous avons décidé de créer une page « pop-up » gestionnaire d'erreur qui nous permet de vérifier les éventuelles erreurs de paramétrage.

Son fonctionnement est plutôt basique : lors de l'appui du bouton « ouvrir le gestionnaire des erreurs » (en réalité, il s'agit d'un ChgScrn), la fenêtre pop-up s'ouvre et nous pouvons ainsi vérifier (grâce au bouton « vérifier ») les erreurs commises. Le bouton a pour fonction de lancer un script (ou FPScript) qui va chercher les éventuelles erreurs.

Un FPScript va jouer le rôle, informatiquement parlant, de fonction : lorsqu'il sera appelé, il se déroulera puis pourra renvoyer (ou non) une ou plusieurs valeurs.

Le FPScript réalisé est le suivant :



```
Script Edit
Script number 4
Script name GestionAlames

1 if(#600 < #700){
2   #680 = 2;}
3 else{
4   if(#603 < #700){#680 = 3;}
5
6   else {
7     if(#707 < 50){#680 = 8;}
8
9     else {
10      if(#503 < 50){#680 = 9;}
11
12      else {
13        #681 = 2;
14        if(#601|#602|#604|#605|#701|#702|#704|#705 < 0){#680 = 6;}
15
16        else {
17          if(#703 < 0){#680 = 5;}
18
19          else{#680=1;}
20
21        }
22      }
23    }
24  }
25 }
26 }
27 }
28
```

Figure 47 : FPScript permettant de débuser les erreurs de paramétrage

Dans celui-ci, nous retrouvons les quelques erreurs que nous avons paramétrées telles que la relation entre les prises de passe (prise de passe d'ébauche doit être plus important que prise de passe de finition), le diamètre minimale de la meule (ne doit pas dépasser 50mm) ou encore le fait que chaque valeur doit être supérieure à zéro.

Ensuite, nous afficherons dans le TxtMsg (élément traité durant la page 1 de l'IHM) une phrase résumant l'erreur suivant la valeur contenue dans la variable #680.

Voici l'annonce des différentes erreurs et un exemple du rendu sur CNC GUIDE :

	Number	Bit set	Char color	Message
1	1			Pas d'erreur de parametrage
2	2			ERREUR : Verifier prise de passe
3	3			ERREUR: Verifier surepaisseur
4	4			ERREUR: Diametre meule insuffisant
5	5			ERREUR: Augmenter tempo de planage
6	6			ERREUR: Verifier vitesse
7	7			ERREUR: Diminuer vitesse
8	8			ERREUR: Le diametre initial de la meule est trop faible
9	9			ERREUR: Le diametre actuel de la meule est trop faible
10	10			

Figure 48 : Capture du paramétrage du TxtMsg de gestion d'erreur

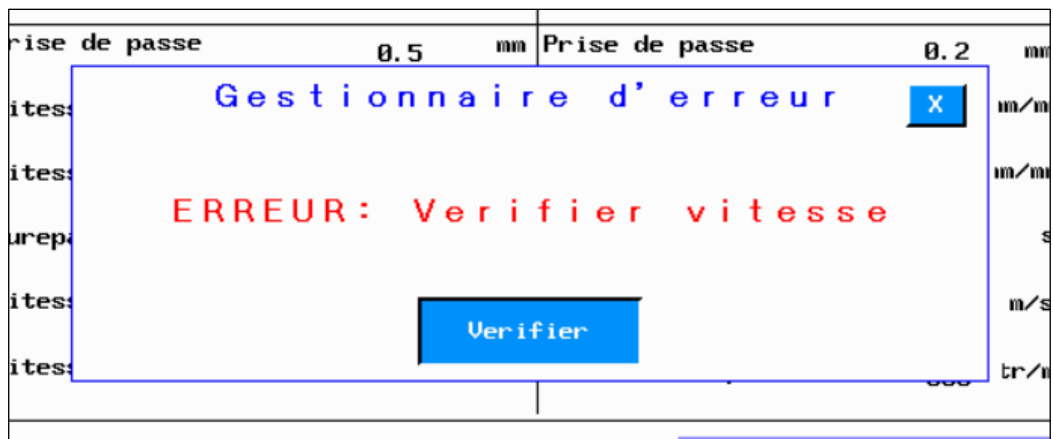


Figure 49 : Exemple d'un cas de vérification d'erreur – vitesse négative

Notons qu'après avoir changé la vitesse concernée, l'erreur disparaît et est remplacée par la mention « Pas d'erreur de paramétrage ».

Modification apportée dans le programme PMC

Finalement, la majeure partie des « alarmes » a été prise en compte par l'IHM et proviendrait d'erreur de paramétrage. La modification apportée dans le programme PMC semble ainsi ridicule : nous avons seulement programmé le bit mémoire R520.0 évoqué plus haut afin de transmettre l'état d'arrêt d'urgence à l'IHM.

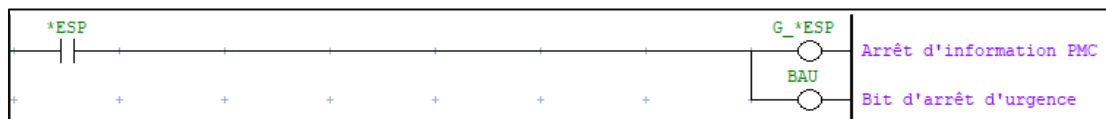


Figure 50 : Programmation du bit R520.0 - arrêt d'urgence

En conclusion, l'évolution des parties techniques se traduit par un changement de fonctionnement général de l'IHM, une simplification du programme PMC et une version plus aboutie du programme pièce.

3.5 Contenu de la journée de formation

Avant de traiter des problèmes rencontrés durant le projet, nous avons décidé d'évoquer la journée de formation effectuée.

Durant cette journée de formation, nous avons rencontré M. Gautier puis avons réalisé des manipulations de re-paramétrage (majoritairement) sur la CN afin de savoir nous débrouiller. Nous avons également appris à téléverser les programmes via la carte mémoire fournie.

A l'issue de cette journée, nous sommes donc repartis accompagnés d'une CN FANUC ainsi que d'une valise nous permettant de visualiser un axe. Les photos des éléments constituant la CN réelle sont répertoriées dans l'annexe 4.

Remarque : Par la suite, lors de la redéfinition des axes de la CN, nous avons paramétré le moteur afin qu'il nous indique l'oscillation de l'axe X (ce qui sera notamment intéressant lors des différentes phases afin de visualiser l'avancement positif et négatif de la meule).

3.6 Problèmes rencontrés et avenir du concours

Enfin, évoquons les problèmes rencontrés et l'avenir du concours.

Problèmes rencontrés

Dans un premier temps, nous n'avions aucune notion de mécanique et connaissions très peu l'industrie. Le sujet de l'avant-projet fût donc, pour nous, compliquer à concevoir. Nous avons dû assimiler toutes les connaissances nécessaires (code G, fonctions M, comprendre le programme PMC etc...) et apprendre un nouveau type de vocabulaire auquel nous faisons face. Il nous a fallu plusieurs semaines avant de comprendre le fonctionnement de la rectifieuse, les enjeux du cycle ou encore la traduction du cycle en code G.

Ensuite, nous avons eu plusieurs problèmes lors de l'amélioration de l'IHM. A ce jour, la plupart ont été réglés mais persiste le problème évoqué plus haut quant aux fenêtres pop-up. En effet, étant impossible de les appeler avec un bit « R » provenant du PMC, nous ne pouvons pas gérer un affichage pop-up de l'arrêt d'urgence. De plus, la limite de performance de la CN nous freine dans le

développement d'animation (cf. Page de réglage des axes) car les images sont plutôt lentes à charger (d'où la diminution de leur qualité afin de gagner des performances).

Finalement, nous n'avons pas eu de problème majeur mais plutôt des blocages passagers que nous résolvions la semaine suivante ou dans la séance.

Avenir et état du concours

Concernant l'état actuel du concours, après avoir été sélectionnés le 4 décembre dernier nous avons participé à la journée de formation de l'entreprise et serons conviés le 30 mars 2021 à finale.

Néanmoins, malgré tous les efforts investis dans l'avant-projet, nous ne pouvons assurer une première place à l'avance. L'avant-projet n'avait pour but que de nous « affûter » pour la finale et constituait simplement un entraînement.

Finalement, même si cette expérience arrive à sa fin, nous restons motivés comme au premier jour afin de prouver notre investissement dans le concours et représenter, fièrement, l'IUT GEII de Tours.

3.7 Bilan technique

Nous avons réalisé une Interface Homme-Machine complète dialoguant avec le PMC et la CN en accord avec le cahier des charges. Nous l'avons par ailleurs approfondi à l'aide d'écrans bonus ou d'une amélioration de l'ergonomie. De plus, nous avons essayé de la rendre le plus agréable visuellement afin de nous démarquer.

Ensuite, nous avons modifié le programme PMC afin de prendre en compte les fonctions M que nous utilisons dans le programme pièce. Il nous a également fallu paramétrer les bits mémoires que le PMC partage avec l'IHM à travers l'environnement LADDER proposé.

Enfin, nous avons développé l'entièreté du programme pièce relatif au cycle demandé et l'avons regroupé en un programme principal et deux sous-programmes intelligemment utilisés. Celui-ci prend en compte l'affichage de l'IHM et permet un fonctionnement sans conflit.

Finalement, nous pensons que si nous connaissons si bien l'environnement de la CN, c'est avant tout grâce aux erreurs que nous avons commises et aux moyens employés pour les corriger.

A travers cette dernière partie, nous avons traité de l'avancement du projet en détail et de l'avenir qui nous attend. L'étape finale se profile et nous avons, à ce jour, hâte de pouvoir nous mesurer aux autres équipes.

Conclusion

Pour conclure, notre projet se base avant tout sur une organisation solide et un esprit d'équipe certain. A travers celui-ci, nous allons mettre un pied dans le domaine de la robotique industrielle et nous former aux compétences de l'Intégration de Commande Numérique.

En termes de connaissances, nous avons trouvé ce projet plus qu'enrichissant car nous n'avons jamais arrêté d'apprendre : d'abord ce fût le vocabulaire de l'industrie, puis les différents programmes et leur spécificités et enfin l'apprentissage tiré de nos erreurs.

Finalement, afin d'approfondir l'avant-projet nous avons essayé de nous démarquer sous plusieurs aspects tels que le graphisme de l'IHM, la tripartition du programme pièce en sous-programme ou encore la simplification du programme PMC. La notice explicative est également censée exprimer la rigueur et le travail fournis par le groupe. Ainsi, nous espérons donner la meilleure image de l'IUT de Tours à l'échelle nationale.

A ce jour, nous avons abattu énormément de travail personnel et sommes fiers d'avoir pu donner le meilleur de nous-même dans la réalisation de cet avant-projet. Nous pensons avoir assimilé les connaissances de base en ICN et certaines notions de façon avancées. Dans l'attente de la finale, nous ne relâcherons rien et resterons concentrés sur notre objectif principal : gagner le concours.

Mots du développement : 10365.



Figure 51 : L'équipe de choc représentant le DUT GEII de Tours !

Avis personnels :

« Les Olympiades FANUC ont été pour moi un véritable défi, en effet la crainte d'un climat tendu et maussade a vite été remplacée par la découverte d'une ambiance ludique et entraînante. Le concours m'a permis de développer mon sens de cohésion et d'acquérir des compétences dans le domaine de l'intégration numérique. Tout en suivant un cahier des charges exigeant et précis, j'ai pu affûter mon esprit analytique et fragmenter mon travail en différentes tâches plus accessibles. En travaillant ainsi j'ai pu acquérir des compétences certaines dans la programmation d'interface de communication entre homme et machine. Ainsi les logiciels de création d'IHM sont pour moi bien plus simples à aborder. Je suis fier d'avoir apporté ma contribution à l'aboutissement de ce projet et, pourquoi pas, une possible victoire ? »

« Dans l'ensemble j'ai bien aimé travailler sur ce projet. J'avais un peu de mal au début car l'aspect concours m'intimidait : je pensais que les attendus seraient irréalisables. Mais, avec le suivi de [REDACTED] et le soutien de mes camarades, j'ai pu pleinement profiter de tout ce qu'on a appris durant la réalisation de ce projet. En plus, nous avons passé de bons moments en groupe ; je ressens vraiment l'aspect d'équipe. Ce projet m'a également permis de prendre en main un nouveau langage informatique qui était assez différent des langages que je connaissais. J'ai donc eu l'opportunité de découvrir les particularités d'un programme « très proche » de la machine et ai pu, tant bien que mal, visualiser les déplacements dans l'espace. Cela ne m'a pas empêché d'y réinvestir mes connaissances apprises en C, par exemple, à travers des structures ou des variables. J'espère sincèrement qu'on ira le plus loin possible. »

« Mon premier ressenti de ce projet se confirme : j'ai me suis prouvé que j'étais capable de travailler en équipe et de persévérer même sur un sujet que je ne maîtrise pas. Le travail tournant autour du PMC dont j'étais en charge m'a permis d'acquérir rigueur et organisation de par la compréhension dont j'ai dû faire part afin de le comprendre comme des modifications que j'ai pu lui apporter. A ce jour, nous avons réussi à nous qualifier mais nous sommes loin d'avoir gagné la finale. L'obstacle à franchir maintenant est de continuer à se former en autonomie et se préparer mentalement à la finale. J'apprécie fortement ce projet, l'ambiance de groupe est toujours radieuse : c'est notre moteur, notre pilier pour avancer et réussir à aller jusqu'au bout de ce concours ! »

« Pour moi, ce concours est plus qu'un simple projet. C'est l'opportunité de participer, avec mes camarades, à une expérience inoubliable. De plus, c'est aussi l'occasion de découvrir le monde industriel auprès d'une entreprise leader dans le domaine de l'ICN. C'est un premier pas dans le monde du travail et l'aspect compétitif permet de voir les choses d'une manière différente. A titre personnel, la gestion de l'IHM m'a permis de prendre du recul sous plusieurs aspects et de développer ma créativité. La gestion de projets m'a, quant à elle, réconforté avec l'esprit d'équipe et la confiance mutuelle. Je suis très fier de la charge de travail accomplie par notre équipe et suis prêt à poursuivre mes efforts jusqu'à la finale ! »

Victor Billaud

Résumé

Dans ce dossier, nous avons abordé les fondements de notre projet de deuxième année : la participation au concours d'Intégration de Commande-Numérique (ICN) organisé par FANUC. Celui-ci est organisé en plusieurs phases : une phase d'avant-projet qui nous a permis d'être sélectionné, une phase de formation, une phase d'amélioration de l'avant-projet et une phase finale : la finale nationale du concours. Les attendus de celui-ci sont la remise d'une Interface Homme-Machine (IHM), d'un programme pièce, d'un programme PMC et d'une notice explicative.

D'abord, l'IHM a été traitée dans sa globalité : page de suivi des opérations, page de paramétrage et page de réglage manuel des axes ; mais également améliorée à l'aide d'un écran d'accueil et d'un gestionnaire d'erreur interactif.

Ensuite, le programme PMC a été modifié afin de prendre en compte les fonctions M utilisées et cela de la façon la plus optimisée. Celui-ci joue également un rôle dans la sécurité du système car il permet d'indiquer à l'IHM lors d'un arrêt d'urgence. De plus, il présente l'avantage d'utiliser les lampes du clavier matriciel lors des mouvements effectués afin de prévenir visuellement l'opérateur des actions en cours.

Pour finir, le programme ISO présente la traduction du cycle demandé en langage machine et incrémente correctement les valeurs des adresses mémoires utilisées dans l'IHM. Il est composé d'un programme principal et de deux sous-programmes.

Finalement, nous sommes prêts à représenter l'IUT à l'échelle nationale lors de la finale et à donner le meilleur de nous-même afin d'espérer gagner cette troisième édition.

Mots du résumé : 250.

Abstract

In this document, we talked about the basis of our second-year project: our involvement in the Numerical Control Integration contest, hosted by the FANUC company. This project is divided into few stages: a selection based on a pre-project, a day of formation and the national final. To complete the pre-project, we had to create a Human Machine Interface (HMI), a piece program, a PMC program and finally an instruction to explain them.

Firstly, the HMI has been fully completed, we have done the three different screens. Furthermore, we also add bonus screens and perfected them all.

Secondly, the PMC program has been optimized and is now able to read M functions. This program manages the safety system as well, like the emergency stop on the HIM or using LED lamps to warn the user about the state of the cycle.

Then, the ISO program, which is divided into three programs (a main one and two subprogrammes), transcribes the asked cycle in parameter readable by the machine and manages the memory addresses used in the HIM.

Finally, we are ready to represent nationally the IUT and give the best of us to win the final of the third edition of the FANUC contest.

Number of words: 202.

Bibliographie

France, Fanuc. « Le concours ». *Les Olympiades Fanuc*, 21 juillet 2020, olympiades-fanuc.com/le-concours/

« Fanuc G-Code List ». *Helman CNC*, 1 janvier 2014, www.helmacnc.com/fanuc-g-code-list/#Fanuc_G-Code_List_Mill.

Appert, Loïc. « Codes G de programmation ISO (pour machine HAAS/FANUC) – Technicien d’Usinage ». *Technicien d’Usinage*, 2004, tu-quincy.fr/codes-g-de-programmation-iso-pour-machine-haas-fanuc/#G00-Positionnement_en_deplacement_rapide

« Les M-codes ». *LinuxCNC*, 2020, linuxcnc.org/docs/html/gcode/-code_fr.html#cha:M-codes

« HOW TO WRITE FANUC LADDER PLC FOR M CODE FUNCTION ». *YouTube*, téléchargé par FANUC PLC and CNC Tutorials, 30 août 2017, www.youtube.com/watch?v=y-VMNhYiqIs

Wikipedia contributors. *Machine-outil à commande numérique*, 10 novembre 2020 (8 décembre 2020). Wikipédia. https://fr.wikipedia.org/wiki/Machine-outil_%C3%A0_commande_num%C3%A9rique

Lab4Sys. *Que ce qu’une machine-outil à commande numérique (CNC) ?* 4 février 2021 (5 avril 2020). Lab4Sys.com. <https://lab4sys.com/fr/que-ce-quune-machine-outil-a-commande-numerique-cnc/?cn-reloaded=1>

ISO 6983-1:1982. (Septembre 1982). ISO. <https://www.iso.org/fr/standard/13540.html>

Index des mots clés

A

automatisation · 5
avant-projet · 9, 10, 13, 14, 51, 55

B

budget · 6, 7, 12, 13

C

carte mentale · 6, 7, 8
CN · 15, 20, 38
CNC GUIDE · 7, 16, 17, 19, 46, 49
code ISO · 38
commande numérique · 5, 9, 15
concours · 5, 6, 7, 9, 10, 12, 14, 16, 21, 22, 31, 34,
50, 51, 52, 53, 54, 55, 60
cycle · 5, 19, 20, 21, 42

F

FANUC · 1, 3, 5, 6, 7, 9, 11, 15, 16, 19, 23, 28, 29,
32, 50, 53, 55, 60
FANUC LADDER III · 7, 15
FANUC PICTURE · 7, 15, 16, 19, 23, 28, 32

I

ICN · 5, 6, 7, 52, 54, 55
IHM · 12, 19, 20
intégration · 5, 53
Interface Homme-Machine · 7, 9, 19, 51, 55
ISO · 7, 12, 19, 21, 38, 46, 55, 60

O

Olympiades · 6, 53, 60

P

PMC · 15, 19, 20
programme pièce · 7, 9, 12, 17, 18, 19, 20, 21, 22,
42, 44, 46, 47, 50, 51, 52, 55
programme PMC · 7, 9, 12, 17, 18, 19, 20, 22, 25,
37, 39, 41, 47, 49, 50, 51, 52, 55
programme-pièce · 21
projet · 1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19, 22, 23, 34, 36, 37, 42, 50, 51, 52, 53,
54, 55

R

rectifieuse · 20
robotique · 5, 52

Table d'illustration

Figure 1 : Image de couverture : modèle 3D d'une rectifieuse, réalisé par nos soins.....	3
Figure 2 : Carte mentale du concours FANUC – Ancienne version	8
Figure 3 : Carte mentale du concours FANUC – Nouvelle version.....	8
Figure 4 : Capture du planning prévisionnel, réalisé par nos soins	10
Figure 5 : Légende des plannings, réalisée par nos soins	10
Figure 6 : Capture du planning réel, réalisé par nos soins.....	10
Figure 7 : Schéma présentant l'architecture machine d'une CN, réalisé par nos soins....	14
Figure 8 : Capture d'écran provenant du logiciel FANUC LADDER III	16
Figure 9 : Environnement CNC GUIDE, modulable grâce aux fenêtres.....	17
Figure 10 : Schéma fonctionnel d'une rectifieuse et de ces axes, réalisé par nos soins....	18
Figure 11 : Capture de la page 1 de l'IHM : Suivi des opérations.....	23
Figure 12 : Extrait du paramétrage de l'action d'un « ChgScrn »	24
Figure 13 : Extrait de la lecture de l'adresse #650 pour lecture de phase	24
Figure 14 : Changement affichage TxtMsg en fonction des valeurs	24
Figure 15 : Capture de la programmation d'un bouton, ici le bouton départ cycle	25
Figure 16 : Programmation d'un affichage d'axe, ici axe X.....	26
Figure 17 : Capture de la page 2 de l'IHM : Page de paramétrage.....	27
Figure 18 : Exemple d'une fonction de lecture et d'écriture.....	28
Figure 19 : Capture de la page 3 de l'IHM : Page de réglage des axes oscillants.....	29
Figure 20 : Exemple du stockage de la valeur de Xmin dans #800	30
Figure 21 : Capture d'un paramétrage d'objet "ImgLamp", ici pour la page 3	30
Figure 22 : Image "PAGE3_1.bmp" utilisée dans la page 3, réalisée par nos soins	31
Figure 23 : Image "PAGE3_3.bmp" utilisée dans la page 3, réalisée par nos soins	31
Figure 24 : Capture de l'onglet de la gestion des pages dans FANUC PICTURE	31
Figure 25 : Capture de la page d'accueil de l'IHM.....	32
Figure 26 : Déclaration d'un écran "pop-up" sur FANUC PICTURE.....	32
Figure 27 : Capture d'une page pop-up de l'IHM : Détection arrêt d'urgence.....	33
Figure 28 : Capture d'une page pop-up de l'IHM : Gestionnaire d'erreurs.....	33
Figure 29 : Capture page accueil IHM - avant sélection.....	34
Figure 30 : Capture page 1 IHM - avant sélection.....	35
Figure 31 : Capture page 2 IHM - avant sélection.....	35
Figure 32 : Capture page 3 IHM - avant sélection.....	36
Figure 33 : Bits de liaisons ajoutés permettant la communication entre PMC et IHM ...	37
Figure 34 : Exemple d'acquiescement d'une fonction M.....	38
Figure 35 : Exemple d'un set-reset de l'acquiescement d'une fonction M.....	38
Figure 36 : Exemple d'association d'un indicateur à une fonction M.....	38
Figure 37 : Problème programmation LADDER bouton stop	39
Figure 38 : Solution de la programmation LADDER du bouton stop.....	40
Figure 39 : Nouvelle méthode d'acquiescement des fonctions M.....	40
Figure 40 : Programmation des touches "+" et "-" de la CN.....	41

Figure 41 : Organigramme du sous-programme de planage O1002.....	43
Figure 42 : Organigramme du programme principal O1000.....	43
Figure 43 : Organigramme du sous-programme d'ébauche et de finition O1001	43
Figure 44 : Exemple des programmes ISO sous Windows	46
Figure 45 : Visualisation d'un programme pièce (ici O1000) sur la CN	46
Figure 46 : Présence de l'arrêt d'urgence de la CN.....	47
Figure 47 : FPScript permettant de débuser les erreurs de paramétrage.....	48
Figure 48 : Capture du paramétrage du TxtMsg de gestion d'erreur.....	49
Figure 49 : Exemple d'un cas de vérification d'erreur – vitesse négative	49
Figure 50 : Programmation du bit R520.0 - arrêt d'urgence.....	49
Figure 51 : L'équipe de choc représentant le DUT GEII de Tours !	52

Table des matières

D4R4 – Rapport de fin de projet.....	1
D4R4 – Rapport de fin de projet.....	3
Sommaire.....	4
Introduction.....	5
Développement.....	6
1) Présentation du projet et de son organisation.....	6
1.1 Le cahier des charges du projet.....	6
1.2 Evolution de la carte mentale.....	8
1.3 Evolution du planning prévisionnel.....	10
1.4 Répartition des tâches.....	12
1.5 Evolution des coûts, besoins financiers et matériels.....	13
2) Analyse du concours.....	14
2.1 Explications générales et composition CN.....	14
2.2 Analyse de l'avant-projet.....	18
2.3 Attendus - Interface Homme Machine (IHM).....	20
2.4 Attendus - <i>Program Machine Controller</i> (PMC) et sécurité.....	20
2.5 Attendus – Programme pièce (ou programme ISO).....	21
3) Avancement du concours.....	22
3.1 L'interface Homme-Machine (IHM).....	22
3.1.1 Page de suivi des opérations.....	22
3.1.2 Page de paramétrage.....	27
3.1.3 Page de réglage des axes.....	29
3.1.4 Les écrans « bonus ».....	31
3.1.5 L'évolution de l'IHM.....	34
3.2 Le programme PMC.....	37
3.2.1 Le programme PMC à l'heure de la sélection.....	37
3.2.2 Les apports de la journée de formation - PMC.....	39
3.2.3 Les améliorations effectuées.....	39
3.3 Le programme pièce.....	42
3.3.1 Compréhension « physique » du cycle.....	42

3.3.2	La traduction du cycle en organigramme	43
3.3.3	Quelques remarques sur le programme	44
3.4	La gestion des alarmes.....	47
3.4.1	L'Interface Homme Machine - sécurité.....	47
3.5	Contenu de la journée de formation.....	50
3.6	Problèmes rencontrés et avenir du concours	50
3.7	Bilan technique.....	51
	Conclusion.....	52
	Résumé.....	55
	Abstract.....	56
	Bibliographie.....	57
	Index des mots clés.....	58
	Table d'illustration	59
	Table des matières.....	61