

# Manuel de l'utilisateur V2.3

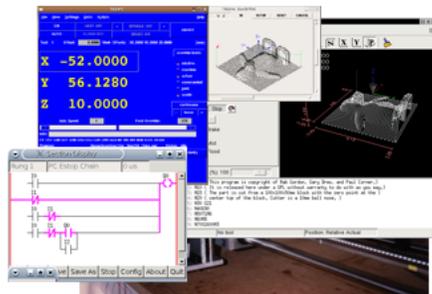
The EMC Team

22 mars 2009



# EMC<sup>2</sup>

## The Enhanced Machine Controller



[www.linuxcnc.org](http://www.linuxcnc.org)

This handbook is a work in progress. If you are able to help with writing, editing, or graphic preparation please contact any member of the writing team or join and send an email to [emc-users@lists.sourceforge.net](mailto:emc-users@lists.sourceforge.net).

Copyright (c) 2000-9 LinuxCNC.org

---

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and one Back-Cover Text : "This EMC Handbook is the product of several authors writing for linuxCNC.org. As you find it to be of value in your work, we invite you to contribute to its revision and growth." A copy of the license is included in the section entitled "GNU Free Documentation License". If you do not find the license you may order a copy from Free Software Foundation, Inc. 59 Temple Place, Suite 330 Boston, MA 02111-1307

# Table des matières

<b>Couverture</b>	<b>i</b>
<b>Table des matières</b>	<b>1</b>
<b>Avant-propos</b>	<b>3</b>
<b>I Les concepts</b>	<b>3</b>
<b>1 Le concept d'EMC</b>	<b>4</b>
1.1 Introduction	4
1.2 Vue d'ensemble de la CNC	4
1.3 Système d'exploitation de l'ordinateur	5
1.4 Comment marche EMC2	5
1.5 Interfaces utilisateur graphiques	6
1.6 Penser comme un opérateur sur CNC	6
1.7 Modes opératoires	9
1.8 Informations affichées	9
1.9 Unités	10
<b>2 Concepts importants</b>	<b>11</b>
2.1 Contrôle de trajectoire	11
2.1.1 La planification de trajectoire	11
2.1.2 Le suivi du parcours	11
2.1.3 La programmation du planificateur	12
2.1.4 Planification des mouvements	12
2.2 G-code	13
2.2.1 Vitesse d'avance	13
2.2.2 Compensation de rayon d'outil	13
2.3 Prise d'origine machine	13
2.4 Changement d'outil	13

<b>II</b>	<b>Les interfaces utilisateur</b>	<b>14</b>
<b>3</b>	<b>L'interface graphique AXIS</b>	<b>15</b>
3.1	Introduction	15
3.2	Pour commencer	15
3.2.1	Une session typique avec AXIS	16
3.3	Éléments de la fenêtre d'AXIS	16
3.3.1	Menu Items	16
3.3.2	Boutons de la barre d'outils	17
3.3.3	Zones d'affichage graphique du programme	17
3.3.3.1	Affichage des coordonnées	17
3.3.3.2	Vue du parcours d'outil	18
3.3.3.3	Etendues du programme	18
3.3.3.4	Le cône d'outil	18
3.3.3.5	Tracé d'outil	19
3.3.3.6	Interaction avec l'affichage	19
3.3.4	Zone texte d'affichage du programme	19
3.3.5	Contrôle manuel	19
3.3.5.1	Le groupe « Axes »	19
3.3.5.2	Le groupe « Broche »	21
3.3.5.3	Le groupe « Arrosage »	21
3.3.6	Données manuelles	21
3.3.6.1	Historique	22
3.3.6.2	Commande MDI	22
3.3.6.3	G-Codes actifs	22
3.3.7	Correcteurs de vitesse	22
3.3.8	Correcteur de vitesse de broche	22
3.3.9	Vitesse de Jog	23
3.4	Raccourcis clavier	23
3.5	emctop : Affichage de l'état d'EMC	23
3.6	mdi : Entrée de données en texte (MDI)	24
3.7	axis-remote : Commande à distance de l'interface graphique d'AXIS	25
3.8	hal_manualtoolchange : Dialogue de changement manuel d'outil	25
3.9	Modules en Python	25
3.10	Utiliser AXIS pour contrôler un tour CNC	25
3.11	Configuration avancée d'AXIS	26
3.11.1	Filtres de programme	26
3.11.2	La base de données des ressources X	27
3.11.3	Manivelle de jog	28
3.11.4	~/.axisrc	28
3.11.5	Editeur externe	28
3.11.6	Panneau de contrôle virtuel	28

<b>4</b>	<b>L'interface TkEMC</b>	<b>29</b>
4.1	Introduction	29
4.2	Utiliser TkEMC	29
4.2.1	Une session typique avec TkEMC	29
4.3	Éléments affichés par TkEMC	30
4.3.1	Boutons principaux	31
4.3.2	Barre de status des différents offsets	31
4.3.3	Zône d'affichage des coordonnées	31
4.3.3.1	Parcours d'outil	31
4.3.4	Contrôle en automatique	32
4.3.4.1	Boutons de contrôle	32
4.3.4.2	Zône texte d'affichage du programme	32
4.3.5	Contrôle en manuel	32
4.3.5.1	Touches implicites	32
4.3.5.2	Le groupe de boutons « Broche »	32
4.3.5.3	Le groupe de boutons « Arrosage »	33
4.3.6	Entrée manuelle de G-code (MDI)	33
4.3.6.1	MDI :	33
4.3.6.2	G-Codes actifs	33
4.3.7	Vitesse de Jog	33
4.3.8	Correcteur de vitesse d'avance travail	34
4.3.9	Correcteur de vitesse de broche	34
4.4	Raccourcis clavier	34
<b>5</b>	<b>L'interface MINI</b>	<b>35</b>
5.1	Introduction	35
5.2	L'écran	36
5.3	Barre de menu	36
5.4	Barre de boutons de contrôle	38
5.4.1	MANUEL	38
5.4.2	AUTO	39
5.4.3	MDI	40
5.4.4	« VITESSE-0 » – « CONTINUER »	40
5.4.5	« ABANDON »	41
5.4.6	« A/U »	41
5.5	Colonne de gauche	41
5.5.1	Afficheurs de position des axes	41
5.5.2	Correcteur de vitesse travail	42
5.5.3	Messages	42

5.6	Colonne de droite	42
5.6.1	Editeur de texte	42
5.6.2	Parcours d'outil	43
5.6.3	Page des outils	44
5.6.4	Page des offsets	44
5.7	Raccourcis clavier	45
5.7.1	Touches courantes	45
5.7.2	Mode Manuel	45
5.7.3	Mode Auto	46
5.8	Divers	46
<b>6</b>	<b>L'interface KEYSTICK</b>	<b>47</b>
6.1	Introduction	47
6.2	Installation	47
6.3	Utilisation	48
<b>III</b>	<b>L'utilisation d'EMC2</b>	<b>49</b>
<b>7</b>	<b>Vue d'ensemble d'un centre d'usinage</b>	<b>50</b>
7.1	Composants mécaniques	50
7.1.1	Axes	50
7.1.1.1	Axes linéaires primaires	50
7.1.1.2	Axes linéaires secondaires	51
7.1.1.3	Axes rotatifs	51
7.1.2	Broche	51
7.1.3	Arrosages	51
7.1.4	Chargeur de pièces	51
7.1.5	Carrousel d'outils	51
7.1.6	Changeur d'outil	51
7.1.7	Visu des messages	51
7.1.8	Correcteurs de vitesse d'avance et de broche	51
7.1.9	Bouton d'effacement de block	52
7.1.10	Bouton d'arrêt optionnel du programme	52
7.2	Composants de contrôle et de données	52
7.2.1	Axes linéaires	52
7.2.2	Axes rotatifs	52
7.2.3	Point contrôlé	52
7.2.4	Mouvement linéaire coordonné	52
7.2.5	Vitesse	53

7.2.6	Arrosage	53
7.2.7	Temporisation	53
7.2.8	Unités	53
7.2.9	Position courante	53
7.2.10	Choix du plan de travail	53
7.2.11	carrousel d'outils	54
7.2.12	Changeur d'outil	54
7.2.13	Chargeur de pièce	54
7.2.14	Boutons des correcteurs de vitesses	54
7.2.15	Modes de contrôle de trajectoire	54
7.3	Interaction de l'interpréteur avec les boutons	54
7.3.1	Boutons de correction de vitesses	54
7.3.2	Bouton d'effacement de block Switch	54
7.3.3	Bouton d'arrêt optionnel du programme	55
7.4	Fichier d'outils	55
7.4.1	Fichier d'outils au format fraiseuse	55
7.4.2	Fichier d'outils au format tour	56
7.5	Paramètres	57
7.6	Systèmes de coordonnées	58
<b>8</b>	<b>Vue d'ensemble du langage</b>	<b>60</b>
8.1	Format d'une ligne	60
8.2	Numéro de ligne	61
8.3	Les mots	61
8.3.1	Nombres	61
8.3.2	Paramètres numérotés	61
8.3.3	Paramètres nommés	62
8.3.4	Expressions	63
8.3.5	Opérateurs binaires	63
8.3.6	Fonctions	63
8.4	Commentaires	64
8.4.1	Messages	64
8.4.2	Enregistrement des mesures	64
8.4.3	Log général	65
	8.4.3.1 (LOGOPEN,filename)	65
	8.4.3.2 (LOGCLOSE)	65
	8.4.3.3 (LOG,...)	65
8.4.4	Messages de déboguage	65
8.4.5	Paramètres dans les commentaires	65
8.5	Répétitions d'items	65
8.6	Ordre des items	66
8.7	Commandes et modes machine	66
8.8	Groupes modaux	66

<b>9</b>	<b>Ordre d'exécution</b>	<b>68</b>
<b>10</b>	<b>G-Code : Bonnes pratiques</b>	<b>69</b>
10.1	Utiliser un nombre de décimales approprié	69
10.2	Utiliser les espaces de façon cohérente	69
10.3	Préférer le "format centre" pour les arcs	69
10.4	Placer les codes modaux importants au début des programmes	69
10.5	Ne pas mettre trop de choses sur une ligne	70
10.6	Ne pas utiliser les numéros de ligne	70
10.7	Lorsque plusieurs systèmes de coordonnées sont déplacés, envisager le mode vitesse inverse du temps	70
<b>11</b>	<b>Fichier d'outils et compensations</b>	<b>71</b>
11.1	Fichier d'outils	71
11.2	Compensation d'outil	71
11.3	Compensation de longueur d'outil	71
11.4	Compensation de rayon d'outil	72
11.4.1	Compensation de rayon d'outil, détails	72
<b>12</b>	<b>G Codes</b>	<b>82</b>
12.1	Table résumée du G-code	83
12.2	G0 : Interpolation linéaire en vitesse rapide	84
12.3	G1 : Interpolation linéaire en vitesse travail	84
12.4	G2, G3 : Interpolation circulaire en vitesse travail	84
12.4.1	Arc au format centre (format recommandé)	85
12.4.2	Arcs au format rayon (format non recommandé)	85
12.5	G4 : Tempo	86
12.6	G7 : Mode diamètre	86
12.7	G8 : Mode rayon	86
12.8	G10 L1 : Entrée dans la table d'outils	86
12.9	G10 L2 : Établissement du système de coordonnées pièce	86
12.10	G17, G18, G19, G17.1, G18.1, G19.1 : Choix du plan de travail	87
12.11	G20, G21 : Choix des unités machine	87
12.12	G28, G28.1 : Aller à une position prédéfinie	87
12.13	G30, G30.1 : Aller à une position prédéfinie	88
12.14	G33, G33.1 : Mouvement avec broche synchronisée	88
12.15	G38.x : Mesure au palpeur	89
12.16	G40, G41, G42, G41.1, G42.1 : Compensation de rayon d'outil	89
12.16.1	Compensation de rayon d'outil depuis une table d'outils	90
12.16.2	Compensation dynamique de rayon d'outil	90

12.17 G43, G43.1, G49 : Compensation de longueur d'outil . . . . .	90
12.17.1 G43, G43.1 : Activation de la compensation de longueur d'outil . . . . .	90
12.17.1.1 G43 : Compensation de longueur d'outil depuis une table d'outils . . . . .	90
12.17.1.2 G43.1 : Compensation dynamique de longueur d'outil . . . . .	91
12.17.2 G49 : Annulation de la compensation de longueur d'outil . . . . .	91
12.18 G53 : Mouvement en coordonnées absolues . . . . .	91
12.19 G54 à G59.3 : Choix du système de coordonnées . . . . .	91
12.20 G61, G61.1, G64 : Types de contrôle de trajectoire . . . . .	91
12.21 G73 : Cycle de perçage avec brise copeaux . . . . .	92
12.22 G76 : Cycle de filetage préprogrammé . . . . .	92
12.23 G80 : Révocation des codes modaux . . . . .	93
12.24 Les cycles préprogrammés . . . . .	93
12.24.1 Mots communs . . . . .	94
12.24.2 Mots « sticky » . . . . .	94
12.24.3 Répétition de cycle . . . . .	94
12.24.4 Mode de retrait . . . . .	94
12.24.5 Erreurs des cycles préprogrammés . . . . .	94
12.24.6 Mouvement préliminaire et Intermédiaire . . . . .	95
12.24.7 G81 : Cycle de perçage . . . . .	95
12.24.8 G82 : Cycle de perçage avec temporisation . . . . .	96
12.24.9 G83 : Cycle de perçage avec débouillage . . . . .	96
12.24.10 G84 : Cycle de taraudage à droite . . . . .	97
12.24.11 G85 : Cycle d'alésage, sans temporisation, retrait en vitesse travail . . . . .	97
12.24.12 G86 : Cycle d'alésage, arrêt de broche, retrait en vitesse rapide . . . . .	97
12.24.13 G87 : Alésage inverse . . . . .	97
12.24.14 G88 : Alésage, arrêt de broche, retrait en manuel . . . . .	97
12.24.15 G89 : Cycle d'alésage, temporisation, retrait en vitesse travail . . . . .	98
12.25 G90, G91 : Modes de déplacement . . . . .	98
12.26 G90.1, G91.1 : Mode distance (I, J et K) . . . . .	98
12.27 G92, G92.1, G92.2, G92.3 : Décalage d'origine des systèmes de coordonnées . . . . .	98
12.28 G93, G94, G95 : Choix des modes de vitesse . . . . .	99
12.29 G96, G97 : Modes de contrôle de la broche . . . . .	99
12.30 G98, G99 : Options du plan de retrait . . . . .	100

<b>13 M Codes</b>	<b>101</b>
13.1 M0, M1, M2, M30, M60 : Arrêts de programme	101
13.2 M3, M4, M5 : Contrôle de broche	101
13.3 M6 : Appel d'outil	102
13.4 M7, M8, M9 : Contrôle d'arrosage	102
13.5 M48, M49 : Contrôle des correcteurs de vitesse	102
13.6 M50 : Contrôle du correcteur de vitesse travail	103
13.7 M51 : Contrôle du correcteur de vitesse broche	103
13.8 M52 : Contrôle de vitesse adaptative	103
13.9 M53 : Contrôle de coupure de vitesse	103
13.10 M61 : Ajuste le numéro de l'outil courant	103
13.11 M62 à M65 : Contrôle de sorties numériques	103
13.12 M66 : Contrôle d'entrée digitale	104
13.13 M100 à M199 : Commandes définies par l'utilisateur	104
<b>14 O Codes</b>	<b>106</b>
14.1 Sous-programmes : « sub », « endsub », « return », « call »	106
14.2 Boucles : « do », « while », « endwhile », « break », « continue »	107
14.3 Conditionnel : « if », « else », « endif »	107
14.4 Répétition : « Repeat »	107
14.5 Indirection	107
14.6 Calcul des valeurs dans les mots O	108
14.7 Appel de fichiers	108
<b>15 Autres Codes</b>	<b>109</b>
15.1 F : Réglage de la vitesse travail	109
15.2 S : Réglage de la vitesse de rotation de la broche	109
15.3 T : Choix de l'outil	109
<b>16 Différences entre le g-code d'EMC2 et le RS274NGC</b>	<b>110</b>
16.1 Différences changeant la signification d'un programme correctement écrit en RS274NGC	110
16.1.1 Position après un changement d'outil	110
16.1.2 Les décalages sont dans les unités du fichier ini	110
16.1.3 Les longueurs/diamètres de la table d'outils sont dans les unités du fichier ini	110
16.1.4 G84, G87 non implémentés	110
16.1.5 G28, G30 avec des mots d'axes	111
16.1.6 M62, M63 non implémentés	111
16.1.7 G0 réduit les longs mouvements angulaires pour les rendre inférieurs à un tour	111
16.2 Différences ne changeant pas la signification des programmes bien écrits en RS274NGC	111
16.2.1 G33, G76 codes de taraudage	111

16.2.2	G38.2 . . . . .	111
16.2.3	G38.3...G38.5 . . . . .	111
16.2.4	O-codes . . . . .	111
16.2.5	M50. . .M53 survitesses . . . . .	111
16.2.6	M61. . .M66 . . . . .	111
16.2.7	G43, G43.1 . . . . .	112
	16.2.7.1 Longueurs d'outil négatives . . . . .	112
	16.2.7.2 Outils de tour . . . . .	112
	16.2.7.3 Longueurs d'outil dynamiques . . . . .	112
16.2.8	G41.1, G42.1 . . . . .	112
16.2.9	G43 sans mot H . . . . .	112
16.2.10	Axes U, V et W . . . . .	112
<b>17 Fichier d'outils et compensations</b>		<b>113</b>
17.1	Fichier d'outils . . . . .	113
17.2	Compensation d'outil . . . . .	113
17.3	Compensation de longueur d'outil . . . . .	113
17.4	Compensation de rayon d'outil . . . . .	114
	17.4.1 Compensation de rayon d'outil, détails . . . . .	114
<b>18 Systèmes de coordonnées et décalage G92</b>		<b>124</b>
18.1	Introduction . . . . .	124
18.2	Commande en coordonnées machine : G53 . . . . .	124
18.3	Décalages pièce (G54 à G59.3 ) . . . . .	124
	18.3.1 Système de coordonnées par défaut . . . . .	126
	18.3.2 Réglage des décalages avec G10 . . . . .	126
18.4	G92 Décalages d'axes . . . . .	126
	18.4.1 Les commandes G92 . . . . .	127
	18.4.2 Réglage des valeurs de G92 . . . . .	127
	18.4.3 Prudence avec G92 . . . . .	128
18.5	Exemple de programme utilisant les décalages d'axes . . . . .	129
<b>19 Spécificités des tours</b>		<b>131</b>
19.1	EMC configuré pour un tour . . . . .	131
19.2	Fichier d'outils . . . . .	131
19.3	Tangenter et corriger l'outil . . . . .	133
19.4	Filetage . . . . .	133

<b>IV Quelques exemples</b>	<b>134</b>
<b>20 Exemples de G-Code</b>	<b>135</b>
20.1 Exemples pour une fraiseuse	135
20.1.1 Fraisage hélicoïdal d'un orifice	135
20.1.2 Rainurage	135
20.1.3 Palpage d'une grille rectangulaire de points	135
20.1.4 Amélioration du palpage d'une grille rectangulaire de points	135
20.1.5 Mesure de longueur d'outil	136
20.1.6 Mesure d'un alésage au palpeur	136
20.1.7 Compensation de rayon d'outil	136
20.2 Exemples pour un tour	136
20.2.1 Filetage	136
<b>21 Image-to-gcode : Usiner un "depth maps"</b>	<b>137</b>
21.1 Qu'est-ce qu'un "depth map"?	137
21.2 Intégrer image-to-gcode dans l'interface utilisateur d'AXIS	137
21.3 Utilisation d'image-to-gcode	138
21.4 Les différentes options	138
21.4.1 Unités	138
21.4.2 Invert Image	138
21.4.3 Normalize Image	138
21.4.4 Expand Image Border	138
21.4.5 Tolerance (unités)	138
21.4.6 Pixel Size (unités)	139
21.4.7 Plunge Feed Rate (unités par minute)	139
21.4.8 Feed Rate (unités par minute)	139
21.4.9 Spindle Speed (RPM)	139
21.4.10 Scan Pattern	139
21.4.11 Scan Direction	139
21.4.12 Depth (unités)	139
21.4.13 Step Over (pixels)	139
21.4.14 Tool Diameter	140
21.4.15 Safety Height	140
21.4.16 Tool Type	140
21.4.17 Lace bounding	140
21.4.18 Contact angle	140
21.4.19 Offset d'ébauche et profondeur par passe d'ébauche	140

<b>V Diagnostic</b>	<b>142</b>
<b>22 Moteurs pas à pas</b>	<b>143</b>
22.1 Problèmes communs . . . . .	143
22.1.1 Le moteur n'avance que d'un pas . . . . .	143
22.1.2 Le moteur ne bouge pas . . . . .	143
22.1.3 Distance incorrecte . . . . .	143
22.2 Messages d'erreur . . . . .	143
22.2.1 Erreur de suivi . . . . .	143
22.2.2 Erreur de RTAPI . . . . .	144
22.3 Tester . . . . .	144
22.3.1 Tester le timing des pas . . . . .	144
<b>VI Annexes</b>	<b>146</b>
<b>23 Legal Section</b>	<b>147</b>
23.1 Copyright Terms . . . . .	147
23.2 GNU Free Documentation License . . . . .	147
<b>VII Index de l'ouvrage</b>	<b>151</b>

# Avant-propos

EMC2 est souple et modulaire. Ces attributs l'ont fait apparaître à certains comme un brouillon de petits morceaux confus, ils se sont demandé pourquoi il en était ainsi. Cette page tente de répondre à cette question avant que vous lecteurs, ne plongiez dedans pour vous faire votre propre idée.

EMC a débuté à l'institut national des standards et des technologies des Etats Unis, le NIST. Il a mûri comme un logiciel fonctionnant sur le système d'exploitation Unix. Unix le rendait différent. Très tôt des développeurs Unix ont apporté une série d'idées concernant l'écriture du code, c'est devenu une écriture selon « la tradition d'Unix ». Les premiers auteurs d'EMC ont suivi cette voie.

Eric S. Raymond, dans son livre *The Art of Unix Programming*, résume la philosophie Unix par la philosophie largement utilisée en ingénierie, le principe KISS Keep it Simple, Stupid Reste Simple, Crétin ou Sois Simple et Concis. Puis il décrit sa vision selon laquelle cette philosophie globale s'applique en tant que norme culturelle Unix, bien qu'on trouve sans surprise de graves entorses à la plupart des règles Unix suivantes :

- \* Règle de modularité : Ecrire des éléments simples reliés par de bonnes interfaces.
- \* Règle de clarté : La Clarté vaut mieux que l'ingéniosité.
- \* Règle de composition : Concevoir des programmes qui peuvent être reliés à d'autres programmes.
- \* Règle de séparation : Séparer les règles du fonctionnement ; Séparer les interfaces du mécanisme.<sup>1</sup>

M. Raymond offre d'autres règles mais ces quatre décrivent les caractéristiques essentielles du système de contrôle de mouvement EMC2.

La règle de **Modularité** est critique. Tout au long de ces manuels, vous trouverez des discussions à propos de l'interpréteur ou à propos des planificateurs de tâche ou de mouvement ou encore à propos de HAL. Chacun d'eux est un module ou un ensemble de modules. Cette modularité vous permettra de ne connecter entre elles que les parties dont vous avez besoin pour le bon fonctionnement de votre machine.

La règle de **clarté** est essentielle. EMC2 est en perpétuelle évolution, il n'est pas terminé et ne le sera jamais. Il est assez complet pour piloter toutes les machines que nous avons voulu qu'il pilote. Une bonne partie de cette évolution est atteinte parce que les utilisateurs et les développeurs peuvent voir le travail des autres et construire sur ce qui est déjà fait.

La règle de **composition** nous permet de concevoir et de construire un contrôleur à partir des nombreux modules existants, en les rendant connectables entre eux. Nous obtenons cette connectivité en appliquant une interface standard à tous les modules et en suivant ce standard.

La règle de **séparation** exige que chaque petite chose soit faite par une partie distincte. En séparant les fonctions, le dépannage est rendu plus aisé, le remplacement de modules par d'autres peut être fait à l'intérieur du système et la comparaison s'effectue facilement.

Qu'apporte la fameuse « tradition d'Unix » à vous, utilisateurs d'EMC2. Elle signifie que vous pourrez faire des choix sur la façon d'utiliser le système. Beaucoup de ces choix affecteront les parties

---

<sup>1</sup>Trouvé sur [http://fr.wikipedia.org/wiki/Philosophie\\_d%27Unix](http://fr.wikipedia.org/wiki/Philosophie_d%27Unix) le 09/09/2008

intégrées à la machine, mais beaucoup également affecteront la manière dont vous utiliserez votre machine. Au cours de votre lecture, vous trouverez différents endroits où vous pourrez faire des comparaisons. Finalement vous pourrez dire « J'utiliserai cette interface plutôt que telle autre » ou, « J'écrirai cette nouvelle partie de telle manière plutôt que de telle autre. » Tout au long de ces manuels nous décrirons l'étendue des possibilités d'EMC2 actuellement disponibles.

Puisque vous commencez votre voyage dans l'utilisation d'EMC2 nous vous proposons ces deux citations :<sup>2</sup>

- "Unix n'a pas été conçu pour empêcher ses utilisateurs de commettre des actes stupides, car cela les empêcherait aussi de réaliser des actes ingénieux." Doug Gwyn
- "Unix est convivial. Cependant Unix ne précise pas vraiment avec qui." Steven King

---

<sup>2</sup>Trouvé sur [http://fr.wikipedia.org/wiki/Philosophie\\_d%27Unix](http://fr.wikipedia.org/wiki/Philosophie_d%27Unix) le 06/07/2008

**Première partie**

**Les concepts**

# Chapitre 1

## Le concept d'EMC

### 1.1 Introduction

Ce document est centré sur l'utilisation d'EMC2, il est plutôt destiné aux lecteurs l'ayant déjà installé et configuré. Quelques informations sur l'installation sont données dans les chapitres suivants. La documentation complète sur l'installation et la configuration se trouve dans le manuel de l'intégrateur.

### 1.2 Vue d'ensemble de la CNC

Le terme CNC a pris beaucoup de sens différents au cours des années. Dans les débuts, la CNC a consisté à remplacer les mains de l'opérateur par des moteurs, lesquels suivaient des commandes reproduisant au mieux les mouvements d'un opérateur tournant les manivelles. De ces premières machines, un langage de contrôle de l'outil est né, puis a grandi. Ce langage est appelé RS274 dont différentes variantes ont été proposées. Des ajouts lui ont été apportés par les fabricants de machines-outils et de contrôleurs pour répondre aux besoins spécifiques de leurs machines. Pour qu'une machine puisse changer d'outil pendant l'exécution d'un programme il est nécessaire de disposer de commandes de changement d'outil. Si un chargeur de pièces doit installer un nouveau brut, il est également nécessaire d'avoir les commandes permettant de gérer ce type de matériel. Comme tous les langages, RS274 a évolué au fil du temps. Actuellement plusieurs dialectes existent. En général, chaque fabricant de machines-outils essaye de rester cohérent avec ses lignes de produits mais il arrive que des commandes produisent des effets complètement différents d'une machine à une autre.

Très récemment le langage de CNC a été masqué derrière différents symboles de programmation se référant à un "Langage <sup>1</sup> de programmation conversationnel." Une caractéristique commune de ces types de programmation est la sélection d'une forme ou d'une géométrie et l'ajout de valeurs pour les coins, les limites, ou les caractéristiques de cette géométrie.

L'usage du Dessin Assisté par Ordinateur a également eue un impact sur les langages de programmation CNC. Les dessins réalisés en DAO étant sauvegardés sous forme de listes de données avec les variables associées à chaque géométrie, ces listes sont facilement interprétées dans le G-Code. Les interpréteurs qui réalisent ce travail sont appelés des programmes de CAM (Computer Aided Machining).

Ce sont des convertisseurs de DAO, l'ensemble des logiciels de dessin, comme Corel<sup>TM</sup> et presque la totalité des programmes graphiques, fournissent des fichiers pouvant être convertis. Ces convertisseurs ont été écrits pour lire un fichier d'une image bitmap, en trames ou vectorielle et la transformer en un fichier G-Code pouvant être lancé sur une machine CNC.

<sup>1</sup>Un fabricant de machines, Hurco, clame avoir seul les droits de l'usage de ces symboles en programmation ainsi que de l'usage du terme conversationnel utilisé dans ce contexte.

Vous vous demandez peut être, "Pourquoi ai-je besoin de savoir celà?" La réponse est que dans sa forme actuelle EMC2 ne peut pas utiliser directement, pour l'usinage, un fichier non converti issu d'une DAO, ni aucune image. EMC2 utilise une variante du premier langage CNC nommée RS274NGC. (pour Next Generation Controller). Toutes les commandes passées à EMC2 doivent l'être dans une forme reconnue par son interpréteur RS274NGC. Ce qui signifie que si vous voulez usiner une pièce dessinée dans un programme graphique ou de dessin, vous devez également utiliser un programme capable de convertir le format du fichier de sortie de ce programme en un autre fichier dans le format reconnu et accepté par l'interpréteur d'EMC2. Il existe plusieurs programmes commerciaux effectuant cette tâche. Au moins un convertisseur (Ace) a été mis sous un type de copyright le rendant disponible au public.

Il y a eue de récentes discussions à propos de l'écriture d'une interface « conversationnelle » ou géométrique, permettant à l'opérateur d'entrer les programmes plus ou moins de la même façon que sur certains contrôleurs propriétaires modernes, mais c'est encore loin.

### 1.3 Système d'exploitation de l'ordinateur

Le code d'EMC2 peut être compilé sur la plupart des distributions GNU-Linux (sous réserve que son noyau ait été patché avec une extension temps réel). En plus du code source, certaines distributions binaires sont disponibles. Le dernier paquet a été créé autour d'une distribution Ubuntu qui est une distribution GNU-Linux. Ubuntu est la distribution la plus appréciée par les utilisateurs de Linux novices, elle est considérée comme étant très facile à utiliser. En plus de celà, il existe pour elle un grand nombre de sites d'entraide dans le monde entier. Installer EMC2 sur cette distribution est extrêmement simple, voir la section ??

EMC2 ne fonctionne pas sur les systèmes d'exploitation Microsoft (TM). La raison en est qu'EMC2 requiert un environnement temps réel pour son planificateur de mouvements et ses générateurs de pas. Il bénéficie aussi de l'excellente stabilité et des performances nettement meilleures du système d'exploitation Linux.

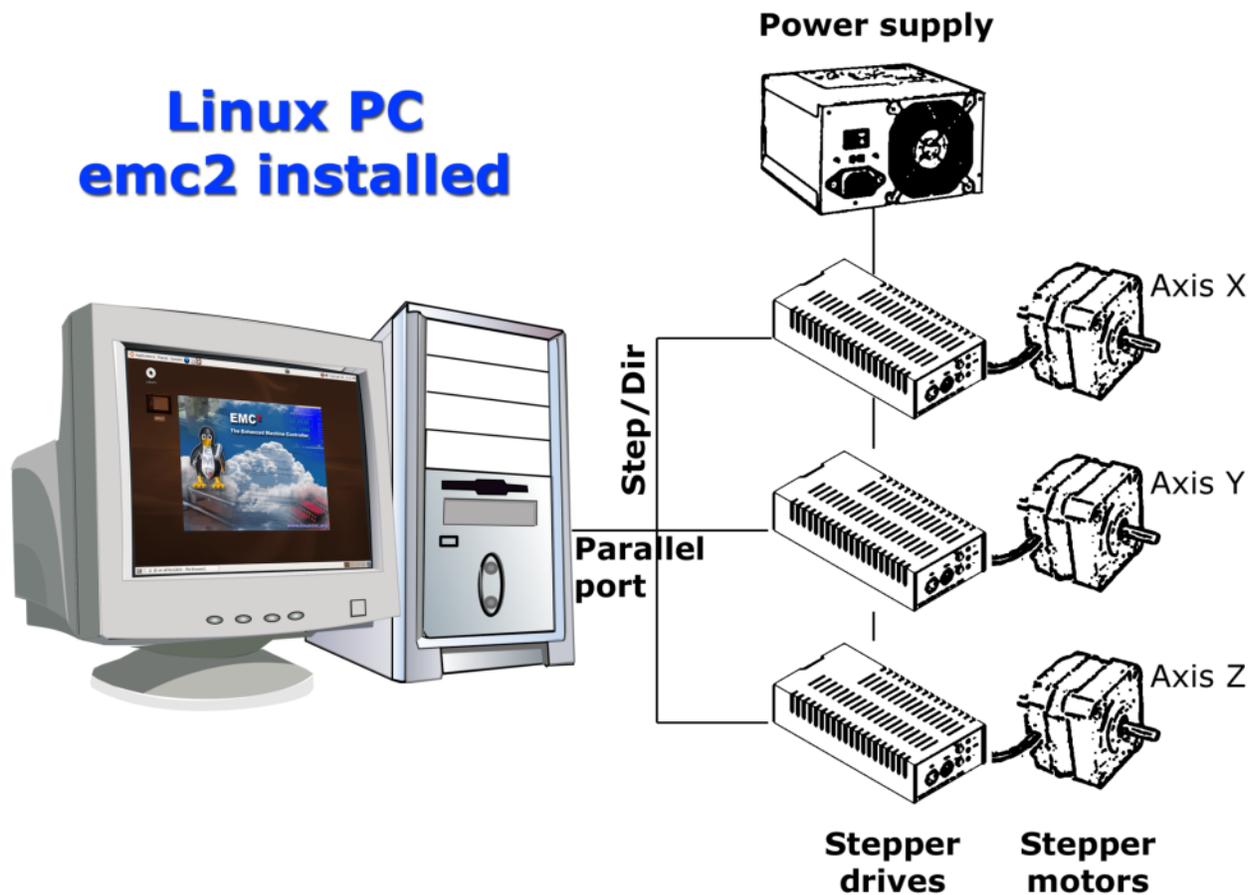
### 1.4 Comment marche EMC2

EMC2 est un peu plus que juste un autre programme de fraiseuse CNC . Il est capable de contrôler des machines-outils, des robots ou d'autres automatismes. Il est capable de contrôler des servomoteurs, des moteurs pas à pas, des relais ainsi que d'autres mécanismes relatifs aux machines-outils. Dans ce manuel nous nous concentrerons seulement sur de petites machines mais aux possibilités impressionnantes, les mini fraiseuses.

La figure 1.1 montre un simple diagramme bloc représentant une machine 3 axes typique comme EMC2 les aime. Cette figure montre un système basé sur des moteurs pas à pas. Le PC, tournant sous Linux contrôle les interfaces de puissance des moteurs pas à pas en leur envoyant des signaux au travers du port parallèle. Ces signaux (impulsionnels) font que la puissance adéquate est fournie aux moteurs. EMC2 peut également contrôler des servomoteurs via une interface de puissance pour servomoteurs ou utiliser le port parallèle étendu connecté à une carte de contrôle externe. Quand nous examinerons chacun des composants qui forment un système EMC2, nous nous référerons à cette machine typique.

Il y a quatre composants dans le logiciel EMC2 : un contrôleur de mouvement (EMCMOT), un contrôleur d'entrées/sorties discrètes (EMCIO), un coordinateur d'exécution des tâches (EMCTASK) et une bibliothèque d'interfaces utilisateur, tant graphiques que textuelles. EMC2, pour prendre le contrôle complet d'une fraiseuse, il doit lancer chacun de ces quatre composants. Chaque composant sera brièvement décrit ci-dessous. En plus, il existe une couche appelée HAL (la couche d'abstraction du matériel), qui permet la configuration simple d'EMC2 sans qu'il soit besoin de le recompiler.

FIG. 1.1 – Simple machine contrôlée par EMC2



## 1.5 Interfaces utilisateur graphiques

L'interface graphique est la partie d'EMC2 qui interagit avec l'opérateur de la machine-outil. EMC2 est fourni avec plusieurs interfaces utilisateurs graphiques :

- Un écran graphique de type caractères nommé keystick 1.3
- Un programme X Windows nommé xemc ??
- Deux GUI basées sur Tcl/Tk nommées tkemc 1.5 et mini 1.4.
- Une GUI basée sur OpenGL, avec prévisualisation du G-Code, appelée AXIS 1.2

Tkemc et Mini peuvent tourner sous Linux, Mac et Microsoft Windows si le langage de programmation Tcl/Tk a été préalablement installé. Les versions Mac et Microsoft Windows peuvent être connectées à EMC2 tournant sur une machine Linux temps réel via une connection réseau, permettant de visualiser la machine depuis un emplacement distant. Les instructions pour installer et configurer une telle connection entre un MAC ou une machine Microsoft et un PC sur lequel tourne EMC2 se trouvent dans le manuel de l'intégrateur.

## 1.6 Penser comme un opérateur sur CNC

Ce manuel ne prétend pas vous apprendre à utiliser un tour ou une fraiseuse. Devenir un opérateur expérimenté prends beaucoup de temps et demande beaucoup de travail. Un auteur a dit une fois, « Nous apprenons de l'expérience, si on la possède toute ». Les outils cassés, les étaux attachés

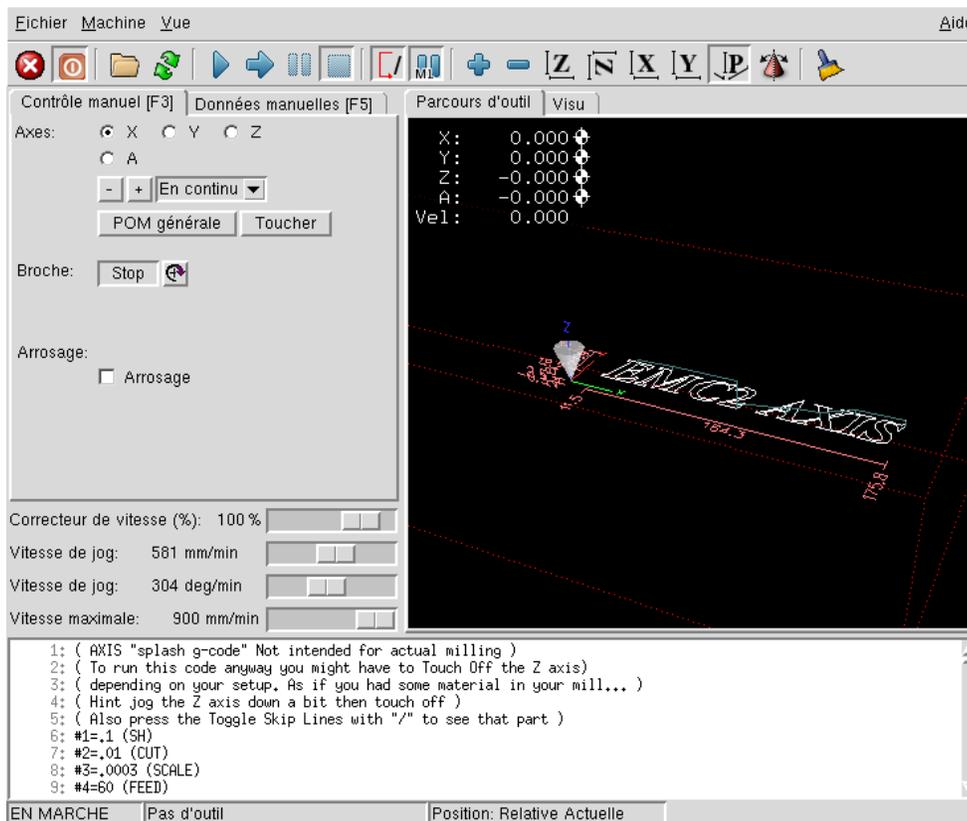


FIG. 1.2 – L'interface graphique AXIS

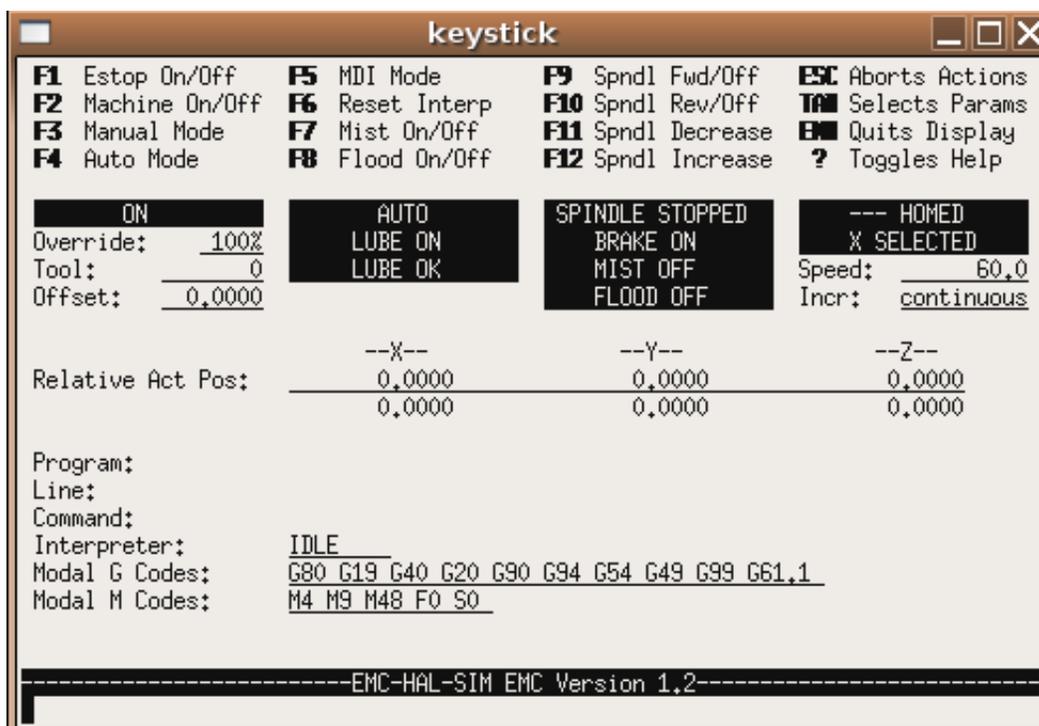


FIG. 1.3 – L'interface graphique Keystick

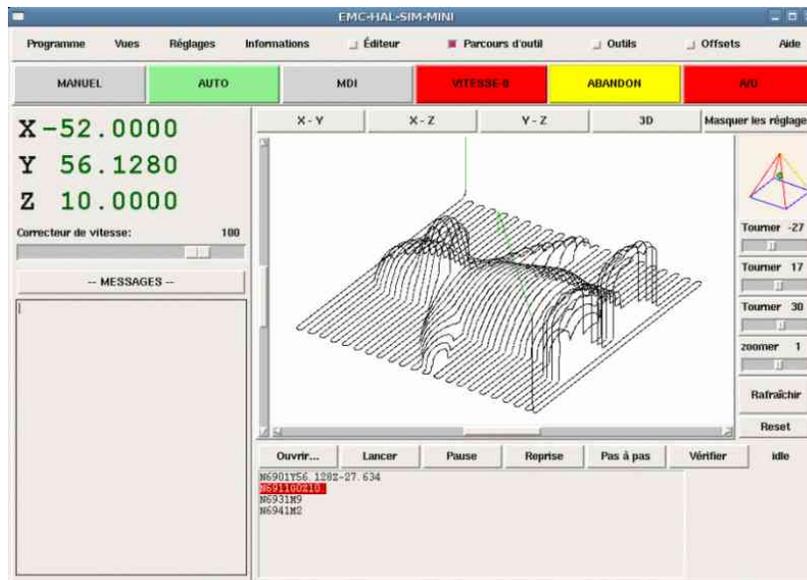


FIG. 1.4 – L'interface graphique Mini

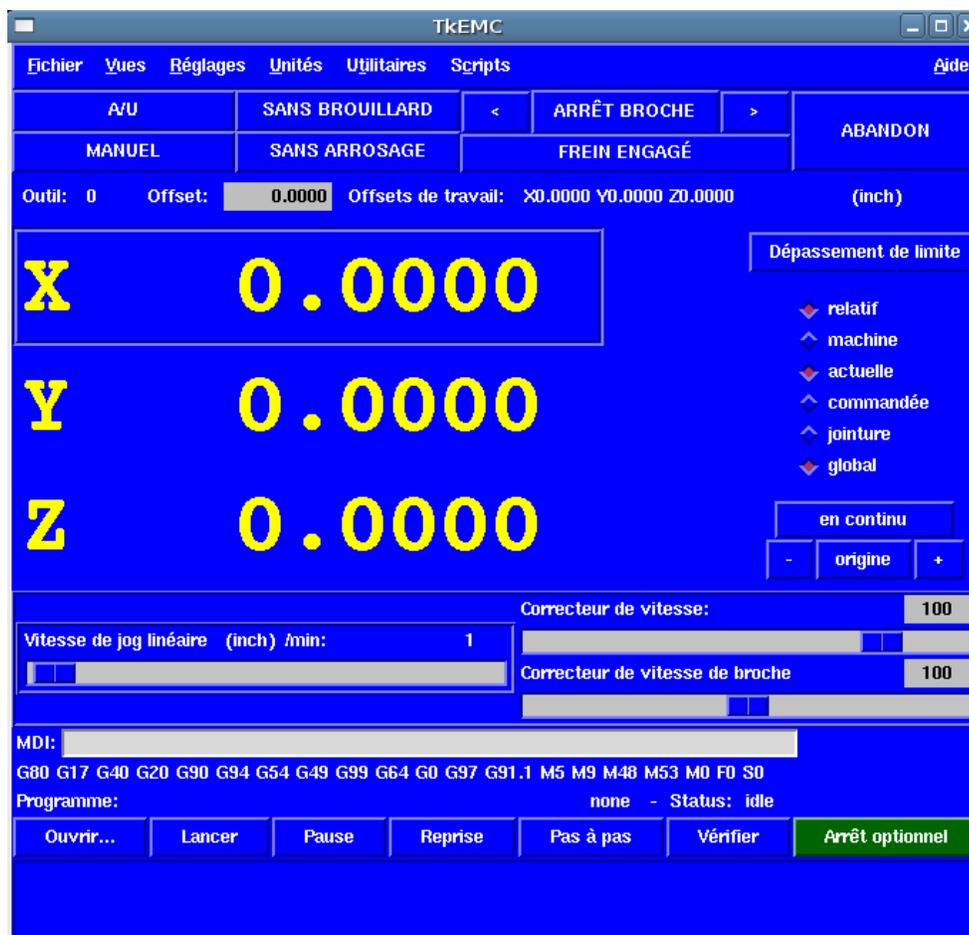


FIG. 1.5 – L'interface graphique TkEmc

et les cicatrices sont les preuves des leçons apprises. Une belle finition, des tolérances serrées et la prudence pendant le travail sont les preuves des leçons retenues. Aucune machine, aucun

programme ne peut remplacer l'expérience humaine.

Maintenant que vous commencez à travailler avec le programme EMC2, vous devez vous placer dans la peau d'un opérateur. Vous devez être dans le rôle de quelqu'un qui a la charge d'une machine. C'est une machine qui attendra vos commandes puis qui exécutera les ordres que vous lui donnerez. Dans ces pages nous donnerons les explications qui vous aideront à devenir un bon opérateur de fraiseuse avec EMC2.

## 1.7 Modes opératoires

Quand EMC2 fonctionne, il existe trois différents modes majeurs pour entrer des commandes. Les modes Manuel, Auto et MDI. Passer d'un mode à un autre marque une grande différence dans le comportement d'EMC2. Des choses spécifiques à un mode ne peuvent pas être faites dans un autre. L'opérateur peut faire une prise d'origine sur un axe en mode manuel mais pas en mode auto ou MDI. L'opérateur peut lancer l'exécution complète d'un programme de G-codes en mode auto mais pas en mode manuel ni en MDI.

En mode manuel, chaque commande est entrée séparément. En termes humains une commande manuelle pourrait être « active l'arrosage » ou « jog l'axe X à 250 millimètres par minute. » C'est en gros équivalent à basculer un interrupteur ou à tourner la manivelle d'un axe. Ces commandes sont normalement contrôlées en pressant un bouton de l'interface graphique avec la souris ou en maintenant appuyée une touche du clavier. En mode auto, un bouton similaire ou l'appui d'une touche peuvent être utilisés pour charger ou lancer l'exécution complète d'un programme de G-codes stocké dans un fichier. En mode d'entrée de données manuelles (MDI) l'opérateur peut saisir un bloc de codes est dire à la machine de l'exécuter en pressant la touche <Return> ou <Entrée> du clavier.

Certaines commandes de mouvement sont disponibles et produisent les mêmes effets dans tous les modes. Il s'agit des commandes ABORT, ESTOP et FEED RATE OVERRIDE. Ces commandes se dispensent d'explications.

L'interface utilisateur graphique AXIS supprime certaines distinctions entre Auto et les autres modes en rendant automatique la disponibilité des commandes, la plupart du temps. Il rend également floue la distinction entre Manuel et MDI parce que certaines commandes manuelles comme Toucher, sont également implémentées en envoyant une commande MDI.

## 1.8 Informations affichées

Quand EMC2 fonctionne, chacun de ses modules maintient la conversation avec les autres et avec l'interface graphique. C'est à la charge de l'affichage de sélectionner quel flux d'information l'opérateur doit voir et de l'arranger sur son écran pour lui rendre facile à comprendre. Le point le plus important de l'affichage est peut être la possibilité de voir le fonctionnement interne d'EMC2. Vous devrez garder un oeil sur le mode d'affichage.

L'affichage du mode actif permet de savoir si il y a cohérence entre l'affichage et la position de chaque axe. La plupart des interfaces permettent à l'opérateur de lire la position actuelle ou commandée ainsi que la position machine ou relative.

**Machine** C'est la position d'un axe par rapport à l'origine machine, après que la POM a été faite.

**Relative** C'est la position relative d'un axe après qu'un ou plusieurs décalages aient été appliqués (pièce, outil, etc).

**Actuelle** C'est la position réelle d'un axe en coordonnées machine ou relatives.

**Commandée** C'est la position commandée sur laquelle l'axe doit aller.

Elles sont toutes exactement les mêmes si aucun décalage n'a été appliqué et qu'aucune bande morte n'a été paramétrée dans le fichier INI. La bande morte (Deadband) est une toute petite distance, supposée très proche – peut être une impulsion de pas ou un point de codeur.

Il est également important de voir tous les messages ou les codes d'erreur envoyés par EMC2. Ceux-ci sont utilisés pour demander que l'opérateur change l'outil, pour décrire les problèmes dans les programmes G-code ou indiquer pourquoi la machine s'est arrêtée.

Vous modifierez votre façon de travailler avec ces petits textes, vous apprendrez, petit à petit, comment ajuster et faire fonctionner votre machine avec le logiciel EMC2. Pendant que vous apprendrez à affiner les réglages, que vous travaillerez avec votre mini fraiseuse, vous penserez à d'autres applications et à d'autres possibilités. Ce seront les sujets d'autres manuels de linuxcnc.org.

## 1.9 Unités

Les unités peuvent prêter à confusion. Vous pouvez vous demander, "Vaut-il mieux travailler en pouces, en pieds, en centimètres, millimètres ou autre?" Plusieurs réponses sont possibles à cette question mais la meilleure est que ça marchera mieux dans l'unité avec laquelle vous avez choisi de configurer la machine.

Au niveau de la machine, chacun des axes est réglé avec une unité fixée dans une variable du fichier INI. Cette variable ressemble à ceci :

```
UNITS = inch
```

ou

```
UNITS = mm
```

Après avoir décidé de l'unité utilisée sur les axes, il faut indiquer à EMC2 combien d'impulsions moteur, ou de codeur, il doit envoyer ou lire pour produire un déplacement pour chaque unité de distance. Après cela, EMC2 sait compter les unités de distance. Cependant il est très important de comprendre que ce calcul de distance pour les axes est différent de la distance commandée dans le programme. Vous pouvez commander une distance en millimètres ou en pouces sans même penser aux unités car il existe des G-code permettant de basculer d'une unité métrique à une unité impériale.

# Chapitre 2

## Concepts importants

Ce chapitre couvre les concepts importants qui doivent être assimilés avant d'essayer de démarrer un fichier de G-code.

### 2.1 Contrôle de trajectoire

#### 2.1.1 La planification de trajectoire

La planification de trajectoire est en général, le moyen qui permet à EMC2 de suivre le chemin spécifié par le programme G-code, tout en restant dans les limites permises par la machine.

Un programme en G-code ne peut jamais être totalement suivi. Par exemple imaginez que vous spécifiez dans une ligne du programme les mouvements suivants :

```
G1 X10 F100 (G1 un mouvement linéaire, X10 la destination, F100 la vitesse)
```

En réalité, la totalité du mouvement ne peut pas être effectuée à F100, puisque la machine commence le mouvement à une vitesse nulle, elle doit accélérer pour se déplacer vers X=10, puis décélérer pour revenir à une vitesse nulle en fin de mouvement. Parfois une portion du mouvement se fera bien à F100, mais pour beaucoup de mouvements, spécialement les petits mouvements, la vitesse spécifiée ne sera jamais atteinte.

Les accélérations et décélérations de base décrite ici ne sont pas complexes et ne nécessite pas de compromis. Les contraintes des axes de la machine sont placés dans le fichier INI, comme la vitesse maximum de l'axe et l'accélération ne devant pas être dépassées par le planificateur de trajectoire.

#### 2.1.2 Le suivi du parcours

Un problème plus compliqué est posé par le suivi du parcours. Quand vous programmez un angle droit en G-code, le planificateur de trajectoire peut suivre différents parcours, tous sont bons dans certains cas; il peut décélérer et s'arrêter exactement sur les coordonnées du sommet de l'angle, puis accélérer dans la direction perpendiculaire. Il peut également faire ce qui est appelé le mode « trajectoire continue », qui consiste à maintenir la vitesse d'avance en passant vers le sommet de l'angle, ce qui nécessite d'arrondir l'angle de façon à respecter les contraintes machine. Vous pouvez remarquer qu'il y a dans ce cas un compromis : vous pouvez ralentir pour avoir un meilleur suivi du parcours, ou conserver une vitesse d'avance élevée au détriment de la finesse des angles, du fait d'un moins bon suivi du parcours. Selon les particularités de l'usinage, du matériau, de l'outillage, etc., le programmeur devra décider du bon compromis.

### 2.1.3 La programmation du planificateur

Les commandes de contrôle de trajectoire sont les suivantes :

- G61** (mode arrêt exact) G61 indique au planificateur de s'arrêter exactement à la fin de chaque segment. Le parcours sera suivi avec exactitude mais les arrêts complets de l'avance peuvent se révéler destructeurs pour la pièce ou l'outillage, selon les particularités de l'usinage.
- G64** (mode trajectoire continue sans tolérance) G64 est juste une trajectoire continue, le « naive cam detector » n'est pas activé. G64 et G64 P0 indiquent au planificateur de sacrifier la précision de suivi du parcours pour conserver une vitesse d'avance élevée. Ce mode est nécessaire pour certains types de matériaux ou d'outillages pour lesquels l'arrêt exact est dangereux. Il peut très bien fonctionner tant que le programmeur garde à l'esprit que le parcours d'outil pourra être plus arrondi que celui indiqué par le programme.
- G64 Px.xxx** (mode trajectoire continue avec tolérance) Ce mode active le « naive cam detector » et active le mode trajectoire continue avec tolérance. Si vous utilisez le millimètre comme unité et programmez G64 P1.27, vous dites au planificateur que vous souhaitez une vitesse d'avance continue, mais qu'aux coins programmés vous voulez un ralentissement suffisant pour que le parcours de l'outil puisse rester à moins de 1.27mm du parcours programmé. L'amplitude exacte du ralentissement dépend de la géométrie de l'angle programmé et des contraintes machine, mais la seule chose dont le programmeur ait à se soucier est la tolérance, ce qui lui donne le contrôle complet des compromis du suivi de parcours. La tolérance de ce mode peut être modifiée tout au long du programme si nécessaire. Attention : spécifier un G64 P0 aura le même effet qu'un G64 seul (voir ci-dessus), c'est rendu nécessaire pour conserver la compatibilité ascendante avec les anciens programmes G-code.

**Trajectoire continue sans tolérance** Le point contrôlé touchera chaque mouvement spécifié à au moins un point. La machine ne pourra jamais se déplacer à une vitesse d'avance telle qu'elle ne puisse pas s'arrêter avec précision à la fin du mouvement en cours (ou du prochain mouvement, si vous mettez en pause lorsque la trajectoire est déjà commencée). La distance avec le point final du mouvement est aussi grande que nécessaire pour maintenir la meilleure vitesse d'avance possible pendant le parcours.

**« Naive Cam Detector »** Les mouvements successifs en G1, concernant uniquement les axes XYZ, dont la déviation par rapport à une ligne droite est inférieur à P, sont fusionnés en une seule ligne droite. Ce mouvement fusionné remplace les mouvements individuels en G1 pour obtenir une nouvelle trajectoire avec tolérance. Entre les mouvements successifs, le point contrôlé ne passera jamais à plus de P- du point final du mouvement en cours. Le point contrôlé touchera au moins un point de chacun des mouvements. La machine ne pourra jamais se déplacer à une vitesse ne lui permettant pas de venir s'arrêter exactement à la fin du mouvement actuel (ou du prochain mouvement, si vous mettez en pause lorsque la trajectoire est déjà commencée).

### 2.1.4 Planification des mouvements

Assurez-vous que les mouvements soient « assez longs » pour convenir à votre machine/matériel. Principalement en raison de la règle selon laquelle « la machine ne pourra jamais se déplacer à une vitesse ne lui permettant pas de venir s'arrêter complètement à la fin du mouvement actuel », il y a une longueur minimale de déplacement permettant à la machine d'atteindre la vitesse demandée avec un réglage d'accélération donné.

Les phases d'accélération et de décélération utilisent chacune la moitié de la variable MAX\_ACCELERATION du fichier .ini. Avec une trajectoire continue c'est exactement inversé, ce qui fait que l'accélération totale de l'axe est égal à la variable MAX\_ACCELERATION. Dans d'autres cas, l'accélération actuelle de la machine est un peu inférieure à celle du fichier ini.

Pour maintenir la vitesse d'avance, le mouvement doit être plus long que la distance qui lui est nécessaire pour accélérer de zéro à la vitesse souhaitée, puis de décélérer pour s'arrêter. En utilisant A comme étant 1/2 de la variable MAX\_ACCELERATION et F comme étant la vitesse d'avance

\*en unités par seconde\*, le temps d'accélération sera  $t_a = F/A$  et la distance d'accélération sera  $d_a = (1/2) * F * t_a$ . Les temps et distance de décélération sont les mêmes, ce qui fait que la distance critique  $d_c = d_a + d_d = 2 * d_a = F^2 / A$ .

Par exemple, pour une vitesse d'avance de 25mm par seconde et une accélération de 250mm/sec<sup>2</sup>, la distance critique sera de  $10^2 / 100 = 1$ mm. Pour une vitesse d'avance de 5mm par seconde, la distance critique ne serait que de  $5^2 / 100 = .25$ mm.

## 2.2 G-code

### 2.2.1 Vitesse d'avance

Si vous avez un tour ou un axe rotatif, pour savoir comment la vitesse d'avance s'applique selon que l'axe est linéaire ou rotatif, lire et comprendre la section « vitesse d'avance » (7.2.5) du manuel de l'utilisateur.

### 2.2.2 Compensation de rayon d'outil

La compensation de rayon d'outil (G41/42) nécessite que l'outil puisse toucher tout au long de la trajectoire programmée sans interférer avec les mouvements d'entrée ou de sortie. Si c'est impossible avec le diamètre de l'outil courant, une erreur est signalée. Un diamètre d'outil inférieur est peut être utilisable sans erreur pour le même parcours. Ce qui signifie que quand ce type de problème se présente, il est possible de programmer un outil plus petit pour usiner le même parcours sans erreur. Voir la section compensation de rayon d'outil (17.4) pour plus d'informations.

## 2.3 Prise d'origine machine

Après le démarrage d'EMC2 chaque axe doit être référencé sur son point d'origine machine avant tout mouvement ou commande MDI.

Pour déroger à ce comportement par défaut, ou pour utiliser l'interface Mini, il est possible d'ajuster l'option `NO_FORCE_HOMING = 1` dans la section [TRAJ] du fichier ini.

## 2.4 Changement d'outil

Il existe plusieurs options pour effectuer un changement d'outil. Voir la section [EMCIO] dans le manuel de l'intégrateur pour les informations sur la configuration de ces options. Voir également les sections G28 12.12 et G30 12.13 du manuel de l'utilisateur.

## **Deuxième partie**

# **Les interfaces utilisateur**

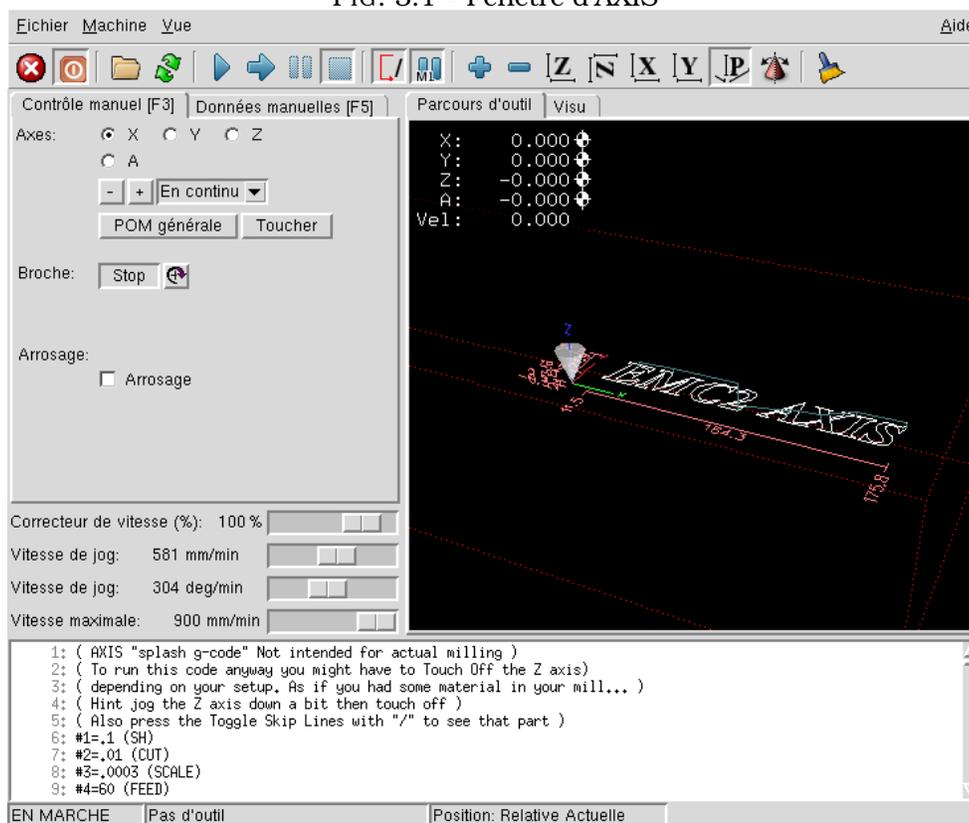
# Chapitre 3

## L'interface graphique AXIS

### 3.1 Introduction

AXIS est une interface utilisateur graphique pour emc2, il offre un aperçu permanent du parcours de l'outil. Il est écrit en Python, utilise Tk et OpenGL pour l'affichage de l'interface graphique.

FIG. 3.1 – Fenêtre d'AXIS



### 3.2 Pour commencer

Pour choisir AXIS comme interface graphique d'emc2, éditez le fichier .ini et dans la section [DISPLAY] changez la ligne DISPLAY comme ceci :

```
DISPLAY = axis
```

Puis, lancez emc2 et choisissez le fichier ini. La configuration simplifiée `sim/axis.ini` est déjà configurée pour utiliser AXIS comme interface.

Quand vous démarrez AXIS, une fenêtre telle que celle de la figure 3.1 s'ouvre.

### 3.2.1 Une session typique avec AXIS

1. Lancez emc et sélectionnez un fichier de configuration.
2. Relâchez le bouton d'arrêt d'urgence « A/U » et mettez la machine en marche.
3. Faites les prises d'origine machine « POM » pour chacun des axes.
4. Chargez un fichier d'usinage.
5. Utilisez l'affichage du parcours d'outil pour vérifier que le programme est correct.
6. Bridez le brut à usiner sur la table.
7. Faites les prises d'origine pièce « POP » de chacun des axes avec le jog et en utilisant le bouton « Toucher ».
8. Lancez le programme.
9. Pour usiner le même fichier une nouvelle fois, retournez à l'étape 6. Pour usiner un fichier différent, retournez à l'étape 4. Quand vous avez fini, quittez AXIS.

## 3.3 Éléments de la fenêtre d'AXIS

La fenêtre d'AXIS contient les éléments suivants :

- Un espace d'affichage qui montre une pré-visualisation du fichier chargé (dans ce cas, « axis.ngc »), ainsi que la position courante du point programmé par la machine CNC. Plus tard, cette zone affichera le parcours de l'outil déplacé par la machine CNC, cette zone est appelée le parcours d'outil (« backplot »)
- Une barre de menus, une barre d'outils, des curseurs et des onglets permettent d'effectuer différentes actions.
- L'onglet « Contrôle manuel », qui permet de faire des mouvements d'axe, de mettre la broche en rotation ou de l'arrêter, de mettre l'arrosage en marche ou de l'arrêter.
- L'onglet « Données manuelles » (aussi appelé MDI), où les blocs de programme G-code peuvent être entrés à la main, une ligne à la fois.
- Les curseurs « Correcteurs de vitesse », qui permettent d'augmenter ou de diminuer la vitesse de la fonction concernée.
- Une zone de textes qui affiche le G-code du fichier chargé.
- Une barre d'état qui affiche l'état de la machine. Dans cette capture d'écran, la machine est en marche, aucun outil n'est monté, la position affichée est « relative » à l'origine machine (par opposition à une position « absolue ») et « actuelle » (par opposition à une position « commandée »)

### 3.3.1 Menu Items

SousSection en cours de rédaction - To Do List

finish menu descriptions

### 3.3.2 Boutons de la barre d'outils

Signification des boutons, de gauche à droite :

1.  « Arrêt d'urgence » (A/U) en Anglais (E-Stop)
2.  Marche/Arrêt puissance machine
3.  Ouvrir un fichier
4.  Recharger le fichier courant
5.  Départ cycle
6.  Cycle en pas à pas
7.  Pause/Reprise
8.  Stopper l'exécution du programme
9.  Sauter ou non les lignes commençant par "/"
10.  Avec ou sans pause optionnelle
11.  Zoom plus
12.  Zoom moins
13.  Vue prédéfinie « Z » (vue de dessus)
14.  Vue prédéfinie « Z basculée »
15.  Vue prédéfinie « X » (vue de côté)
16.  Vue prédéfinie « Y » (vue de face)
17.  Vue prédéfinie « P » (vue en perspective)
18.  Orienter la vue avec le bouton gauche de la souris
19.  Rafraîchir le parcours d'outil

### 3.3.3 Zones d'affichage graphique du programme

#### 3.3.3.1 Affichage des coordonnées

L'affichage des coordonnées est situé en haut à gauche de l'écran graphique. Il montre les positions de la machine. A gauche du nom de l'axe, un symbole d'origine (⊕) est visible si la prise d'origine de

l'axe a été faite. A droite du nom de l'axe, un symbole de limite (←) est visible si l'axe est sur un de ses capteurs de limite.

Pour interpréter correctement ces valeurs, référez vous à l'indicateur « Position : » de la barre d'état. Si la position est « Absolue », alors les valeurs affichées sont exprimées en coordonnées machine. Si la position est « Relative », alors les valeurs affichées sont exprimées en coordonnées relatives à la pièce. Quand les coordonnées affichées sont relatives, une marque d'origine de couleur cyan est visible pour représenter « l'origine machine » (⊗). Si la position est « Commandée », alors il s'agit de la position à atteindre – par exemple, les coordonnées passées dans une commande G0. Si la position est « Actuelle », alors il s'agit de la position à laquelle la machine vient de se déplacer. Ces valeurs peuvent varier pour certaines raisons : erreur de suivi, bande morte, résolution d'encodeur ou taille de pas. Par exemple, si vous demandez un mouvement à X 0.08 à votre fraiseuse, mais un pas du moteur fait 0.03, alors la position « Commandée » sera de 0.08 mais la position « Actuelle » sera de 0.06 (2 pas) ou 0.09 (3 pas).

### 3.3.3.2 Vue du parcours d'outil

Quand un fichier est chargé, une vue du parcours d'outil qu'il produira est visible dans la zone graphique. Les mouvements en vitesse rapide (tels ceux produits par une commande G0) sont affichés en lignes pointillées vertes. Les déplacements en vitesse programmée (tels ceux produits par une commande G1) sont affichés en lignes continues blanches. Les arrêts temporisés (tels ceux produits par la commande G4) sont représentés par une petite marque « X ».

### 3.3.3.3 Etendues du programme

Les « étendues » du programme sont visibles pour chacun des axes. Aux extrémités de chacune, la plus petite et la plus grande valeur des coordonnées sont indiquées. Au milieu, la différence entre ces valeurs de coordonnées est visible. Sur la figure 3.1, l'étendue des X du fichier va de -1.95 à 1.88 pouces, un total de 3.83 pouces.

Quand une coordonnée excède une « soft limits » établie dans le fichier .ini, la dimension correspondante est affichée dans une couleur différente. Ici, la limite maximale est dépassée sur l'axe X :



### 3.3.3.4 Le cône d'outil

L'emplacement de la pointe de l'outil est indiqué par le « cône d'outil ». Le cône d'outil ne donne pas d'indication sur la forme, la longueur ou le rayon de l'outil.

Quand un outil est chargé, (par exemple dans le MDI, avec la commande T1 M6), le cône passe de cône à cylindrique, il indique alors le diamètre de l'outil lu dans le fichier de la table d'outil.

### 3.3.3.5 Tracé d'outil

Quand la machine se déplace, elle laisse une trace appelée le tracé d'outil. La couleur des lignes indique le type de mouvement : Jaune pour les jogs, vert clair pour les mouvements en vitesse rapide, rouge pour les mouvements en vitesse d'avance programmée et magenta pour les mouvements circulaires en vitesse d'avance programmée.

### 3.3.3.6 Interaction avec l'affichage

Par un clic gauche sur une portion du parcours d'outil, la ligne sous la souris passe en surbrillance à la fois dans le graphique et dans le texte. Un clic droit dans une zone vide enlève la surbrillance.

En déplaçant la souris avec son bouton gauche appuyé, la vue est glissée sur l'écran.

En déplaçant la souris avec le bouton « Maj » enfoncé, ou en glissant avec la molette de la souris appuyée, la vue est tournée. Si une ligne du tracé est en surbrillance, elle devient le centre de rotation de la vue. Autrement, le centre de rotation est le milieu du fichier dans son ensemble.

En tournant la molette de la souris ou en glissant la souris avec son bouton droit enfoncé ou encore en glissant la souris avec son bouton gauche enfoncé et la touche « Ctrl » appuyée, le tracé sera zoomé en plus ou en moins.

En cliquant sur une des icônes de vue pré-définie de la barre d'outils, ou en pressant la touche « V », cette vue est sélectionnée.

### 3.3.4 Zone texte d'affichage du programme

Un clic gauche sur une ligne du programme passe la ligne en surbrillance à la fois dans la zone texte et dans la zone graphique.

Quand le programme est lancé, la ligne en cours d'exécution est en surbrillance rouge. Si aucune ligne n'est sélectionnée par l'utilisateur, le texte défile automatiquement pour toujours laisser la ligne courante visible.

### 3.3.5 Contrôle manuel

Quand la machine est en marche mais qu'aucun programme n'est exécuté, les éléments graphiques de l'onglet « Contrôle manuel » peuvent être utilisés pour actionner le centre d'usinage ou mettre en marche et arrêter les différents organes de la machine.

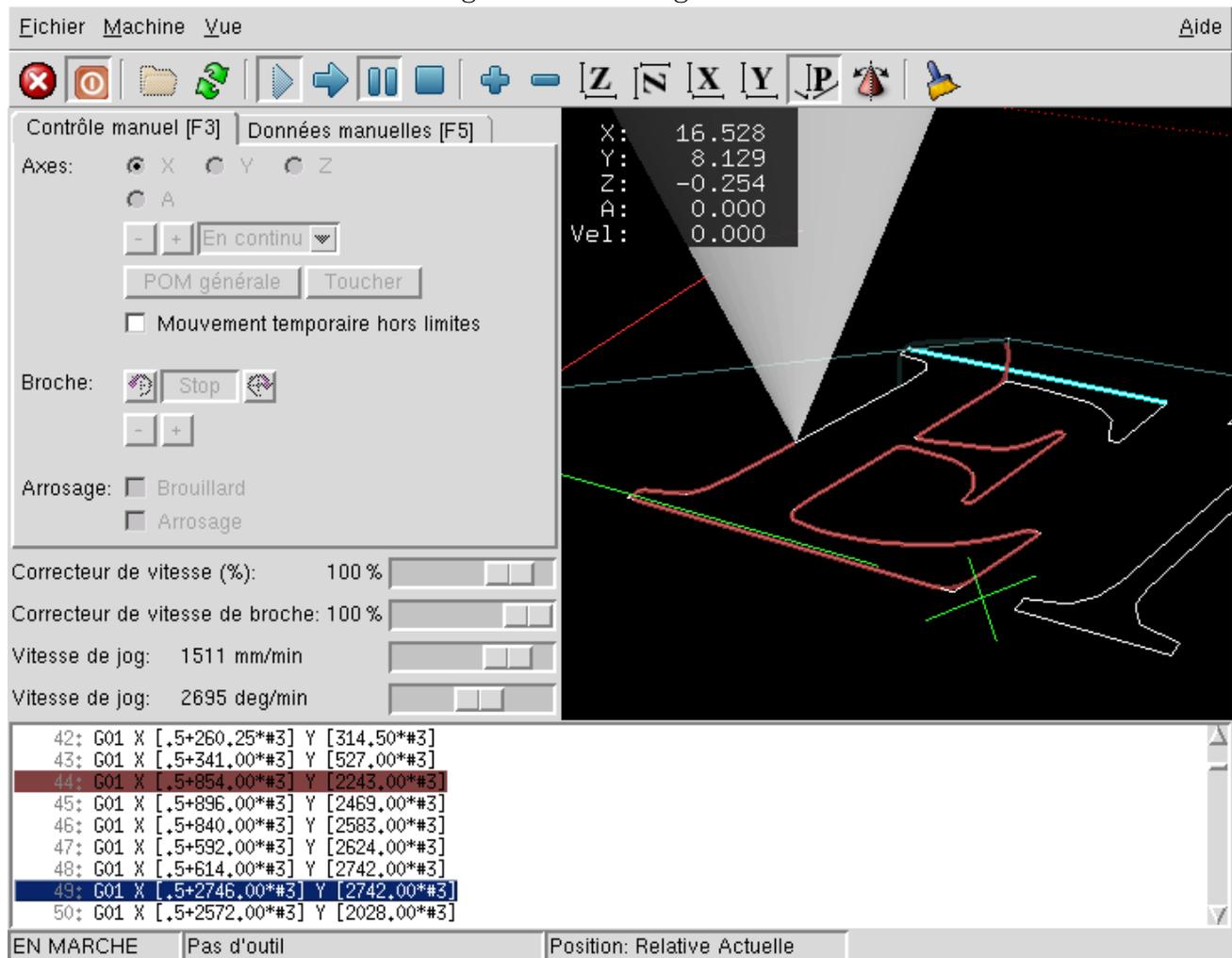
Quand la machine n'est pas en marche ou quand un programme est en cours d'exécution, le contrôle manuel est indisponible.

Certains des éléments décrits plus bas ne sont pas disponibles sur toutes les machines. Quand AXIS détecte qu'une « pin » particulière n'est pas connectée dans le fichier HAL, l'élément correspondant de l'onglet Contrôle manuel est supprimé. Par exemple, si la « pin HAL » `motion.spindle-brake` n'est pas connectée, alors le bouton « Frein de broche » n'apparaîtra pas sur l'écran. Si la variable d'environnement `AXIS_NO_AUTOCONFIGURE` est mise à 1, ce comportement est désactivé et tous les boutons sont visibles.

#### 3.3.5.1 Le groupe « Axes »

Les cases à cocher du groupe « Axes » permettent de choisir l'axe de la machine à actionner manuellement. Cette action s'appelle le « jog ». Premièrement sélectionnez l'axe à actionner en cochant sa case. Puis cliquez le bouton « + » ou « - » selon le sens de déplacement souhaité. Les quatre premiers axes peuvent aussi être déplacés avec les touches fléchées pour (X et Y), Page précédente et Page suivante pour (Z) et les touches [ et ] pour (A).

FIG. 3.2 – Ligne courante et ligne en surbrillance



Si « En continu » est sélectionné, le mouvement continuera tant que la touche ou le bouton resteront appuyés. Si une autre valeur est sélectionnée, la machine se déplacera juste de la distance affichée à chaque fois que la touche ou le bouton seront appuyés. Par défaut, les valeurs disponibles sont :

0.1000 0.0100 0.0010 0.0001

Le réglage du fichier .ini [DISPLAY] INCREMENTS peut être utilisé pour remplacer ces valeurs par défaut. Ces valeurs doivent contenir des nombres décimaux (ex. 0.1000) ou des nombres fractionnaires (ex. 1/16), éventuellement suivis par une unité (une parmi 'cm', 'mm', 'um', 'inch', 'in', ou 'mil'). Si aucune unité n'est spécifiée, les unités natives de la machine seront utilisées. Pour les utilisateurs préférant les unités métriques, un bon réglage pourrait être :

INCREMENTS = 10 um, 50 um, 0.1mm, 0.5mm, 1mm, 5mm, 10 mm

Pour les utilisateurs préférant les unités « impériales », un bon réglage pourrait être :

INCREMENTS = 1/4 in, 1/16 in, 1/32 in, 1/64 in, 1 mil, .1 mil

ou

INCREMENTS = .5 in, .1 in, 50 mil, 10 mil, 5 mil, 1 mil, .1 mil

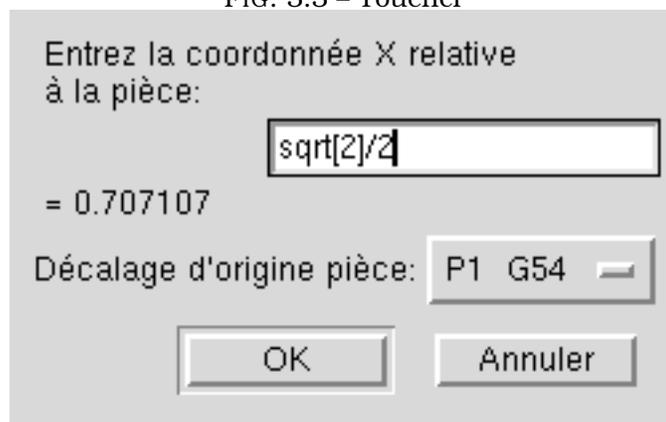
Le mélange entre métrique et impérial est possible :

INCREMENTS = 1 inch, 1 mil, 1 cm, 1 mm, 1 um

Si votre machine a une séquence de prise d'origines machine définie, le bouton « POM générale » ou les touches Ctrl+origine lanceront cette séquence. Autrement, le bouton sera lu « Origine machine », et enverra l'axe courant à sa position d'origine. L'appui de la touche origine seule enverra l'axe courant à sa position d'origine même si une séquence de prise d'origines machine est définie. Selon votre configuration, la POM peut ajuster la valeur de la position de l'axe comme étant la position absolue 0.0, ou elle peut faire déplacer la machine vers une position d'origine spécifiée et avec l'usage de « capteurs d'origine ».

Si le bouton « Toucher » ou la touche Fin sont appuyés, le décalage d'origine pièce « G54 » de l'axe actif prendra la valeur spécifiée dans le champ de la boîte de dialogue. Les expressions peuvent être entrées en suivant les règles de programmation rs274ngc, sauf les variables qui ne peuvent pas être utilisées. La valeur résultante sera affichée sous le champ.

FIG. 3.3 – Toucher



La case à cocher « Mouvement temporaire hors limites » permet un déplacement temporaire de la machine hors des limites physiques. Par exemple, pour ramener dans les limites, un axe engagé par erreur sur un fin de course.

### 3.3.5.2 Le groupe « Broche »

Les boutons de la première rangée permettent de sélectionner la direction de rotation de la broche : Sens anti-horaire, Arrêt, Sens horaire. Les boutons de la rangée suivante augmentent ou diminuent la fréquence de rotation. La case à cocher de la troisième rangée permet d'engager ou de relâcher le frein de broche. Selon la configuration de votre machine, ces éléments n'apparaîtront peut être pas tous.

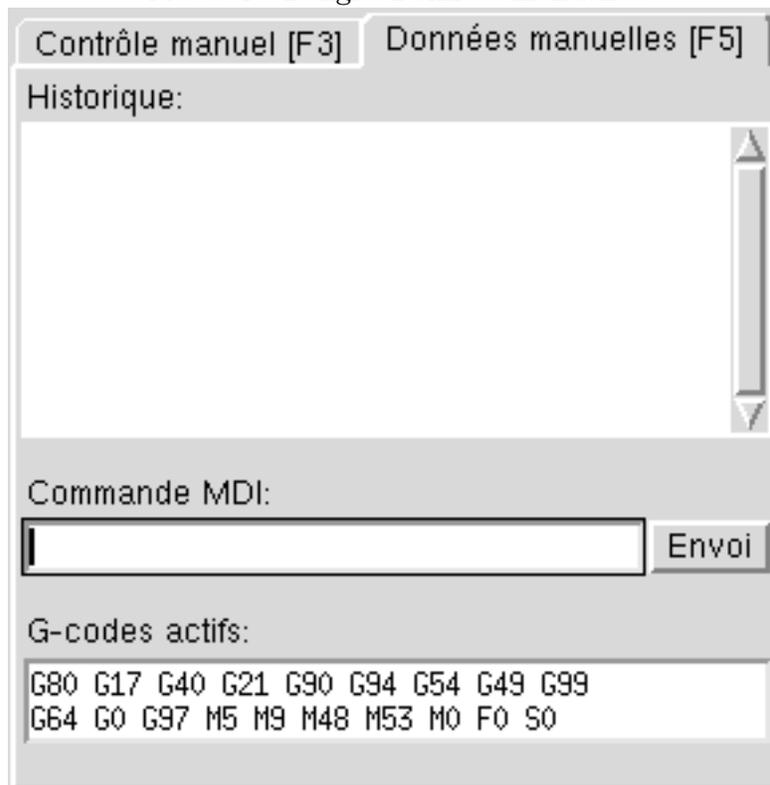
### 3.3.5.3 Le groupe « Arrosage »

Ces deux boutons permettent d'activer le « Brouillard » et l'« Arrosage » ou de les désactiver. Selon la configuration de votre machine, ces boutons n'apparaîtront peut être pas tous.

### 3.3.6 Données manuelles

L'onglet d'entrée de données manuelles (encore appelé MDI), permet d'entrer manuellement et une par une, des lignes de programme en G-code. Quand la machine n'est pas en marche, ou quand un programme est en cours d'exécution, cet onglet n'est pas opérationnel.

FIG. 3.4 – L'onglet Données manuelles



### 3.3.6.1 Historique

Affiche les commandes précédemment tapées au cours de cette session.

### 3.3.6.2 Commande MDI

Ce champ permet la saisie d'une ligne de commande à exécuter. La commande sera exécutée par l'appui de la touche « Entrée » ou un clic du bouton « Envoi ».

### 3.3.6.3 G-Codes actifs

Ce champ affiche les « codes modaux » actuellement actifs dans l'interpréteur. Par exemple, « G54 » indique que le décalage d'origine « G54 » sera appliqué à toutes les coordonnées qui seront entrées.

## 3.3.7 Correcteurs de vitesse

En déplaçant le curseur, la vitesse de déplacement programmée peut être modifiée. Par exemple, si un programme requiert une vitesse à F60 et que le curseur est placé sur 120%, alors la vitesse résultante sera de 72.

## 3.3.8 Correcteur de vitesse de broche

En déplaçant ce curseur, la vitesse programmée de la broche peut être modifiée. Par exemple, si un programme requiert une vitesse à F8000 et que le curseur est placé sur 80%, alors la fréquence de rotation résultante sera de 6400. Cet élément n'apparaît que si la « HAL pin » `motion.spindle-speed-out` est connectée dans `.ini`.

### 3.3.9 Vitesse de Jog

En déplaçant ce curseur, la vitesse de jog peut être modifiée. Par exemple, si ce curseur est placé sur 100 mm/mn, alors un jog de 1 mm durera .6 secondes, ou 1/100 de minute. Du côté gauche du curseur (jog lent) l'espacement des valeurs est petit alors que du côté droit (jog rapide) l'espacement des valeurs est plus grand, cela permet une large étendue de vitesses de jog avec un contrôle plus fin du curseur dans les zones les plus importantes.

Sur les machines avec axes rotatifs, un second curseur de vitesse est présent. Il permet d'ajuster la vitesse de rotation des axes rotatifs (A, B et C).

## 3.4 Raccourcis clavier

La plupart des actions d'AXIS sont accessibles depuis le clavier. La liste complète des raccourcis clavier est disponible dans l'aide rapide d'AXIS qui s'affiche en cliquant sur AIDE > AIDE RAPIDE. Beaucoup de ces raccourcis sont inaccessible en mode Entrées manuelles.

Les raccourcis clavier les plus fréquents sont visibles dans la table 3.1.

TAB. 3.1 – Raccourcis clavier usuels

Touches	Actions produites
F1	Bascule l'arrêt d'urgence
F2	Bascule le marche/arrêt machine
, 1 .. 9, 0	Correcteurs de vitesse de 0% à 100%
X, `	Active le premier axe
Y, 1	Active le deuxième axe
Z, 2	Active le troisième axe
A, 3	Active le quatrième axe
I	Sélection d'incrément du jog
C	jog en mode continu
Ctrl+origine	Lance une séquence de POM
Fin	Toucher : valide l'offset G54 de l'axe actif
Gauche, Droite	Jog du premier axe
Up, Down	Jog du deuxième axe
Pg Up, Pg Dn	Jog du troisième axe
[, ]	Jog du quatrième axe
O	Ouvrir un fichier
Ctrl+R	Recharger le fichier courant
R	Exécuter le programme
P	Pause dans l'exécution du programme
S	Reprise de l'exécution du programme
ESC	Stopper l'exécution
Ctrl+K	Raffraîchi le tracé d'outil
V	Défilement cyclique des vues prédéfinies

## 3.5 emctop : Affichage de l'état d'EMC

AXIS inclut un programme appelé « emctop » qui affiche en détail l'état d'emc. Ce programme est accessible dans le menu MACHINE > FENÊTRE D'ÉTAT D'EMC2

Le nom de chaque entrée est affiché dans la colonne de gauche. La valeur courante de chaque entrée s'affiche dans la colonne de droite. Si la valeur a changé récemment, elle s'affiche en surbrillance rouge.

FIG. 3.5 – Fenêtre d'état d'EMC

```

acceleration      20,0
actual_position   0,0 3,91499996185 2,66483998299 0,0 0,0 0,0
angular_units     1,0
axes              3
command           n3510 M2
current_line      57
cycle_time        0,01
debug             0
echo_serial_number 11
enabled           1
estop             0
exec_state        5
feedrate          9,0
file              /home/jepler/src/emc2/nc_files/cds.ngc
flood             0
gcodes            G1 G17 G40 G20 G90 G94 G54 G49 G99 G64
homed             0 0 0 0 0 0
id                18
inpos             0
interp_state      waiting
interpreter_errcode 0
kinematics_type   1
..

```

### 3.6 mdi : Entrée de données en texte (MDI)

AXIS inclut un programme appelé « mdi », il permet d'envoyer des commandes à la session d'EMC2 en cours, sous forme de lignes de texte. Vous pouvez lancer ce programme en ouvrant une console et en tapant :

```
mdi /path/to/emc.nml
```

En cours d'exécution il affiche le prompt : MDI>. Quand une ligne vide est entrée, la position courante de la machine est affichée. Quand une commande est entrée, elle est passée à emc qui l'exécute. Une courte session MDI est visible sur la figure 3.6.

FIG. 3.6 – Session MDI

```

$ mdi ~/emc2/configs/sim/emc.nml
MDI>
(0.0, 0.0, 0.0, 0.0, 0.0, 0.0)
MDI> G1 F5 X1
MDI>
(0.5928500000000374, 0.0, 0.0, 0.0, 0.0, 0.0)
MDI>
(1.0000000000000639, 0.0, 0.0, 0.0, 0.0, 0.0)

```

### 3.7 axis-remote : Commande à distance de l'interface graphique d'AXIS

AXIS inclut un programme appelé « axis-remote » qui permet d'envoyer certaines commandes vers l'application AXIS fonctionnant à distance. Les commandes disponibles sont visibles en faisant : `axis-remote --help` pour vérifier qu'AXIS est en marche, inclure : (`--ping`), charger un fichier, recharger le fichier courant avec : (`--reload`) et quitter le programme AXIS avec : (`--quit`).

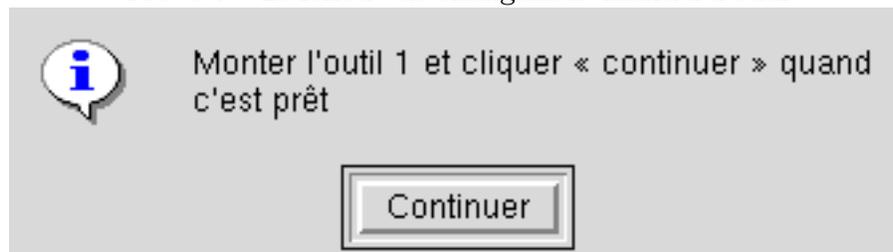
### 3.8 hal\_manualtoolchange : Dialogue de changement manuel d'outil

AXIS inclut un composant Hall appelé « hal\_manualtoolchange », il ouvre une fenêtre d'appel d'outil (Figure 3.7) quand la commande M6 est invoquée. Dès que le bouton Continuer est pressé, l'exécution du programme reprend.

Le fichier de configuration `.hal : configs/sim/axis_manualtoolchange.hal` montre les commandes HAL nécessaires pour l'utilisation de ce composant.

hal\_manualtoolchange peut être utilisé même si l'interface graphique AXIS n'est pas en service.

FIG. 3.7 – La fenêtre de changement manuel d'outil



### 3.9 Modules en Python

AXIS inclut plusieurs modules en Python qui peuvent être très utiles. Pour des informations complètes sur ces modules, faites : « `pydoc <nom du module>` » ou lisez son code source. Modules inclus :

- emc fournit l'accès aux commandes d'emc, à son état et aux chaînes d'erreur
- gcode fournit l'accès à l'interpréteur rs274ngc
- rs274 fournit des outils supplémentaires pour travailler sur les fichiers rs274ngc
- hal permet la création par l'utilisateur de composants de HAL écrits en Python
- \_togl fournit des éléments OpenGL utilisables dans les applications Tkinter
- minigl fournit l'accès aux sous-ensembles d'OpenGL utilisés par AXIS

Pour utiliser ces modules dans vos propres scripts, assurez-vous que le répertoire où ils se trouvent est dans le chemin d'accès des modules Python. Avec une version installée d'EMC2, ça se fera automatiquement. Avec une version installée en « in-place », ça peut être fait avec l'aide de : `scripts/emc-environme`

### 3.10 Utiliser AXIS pour contrôler un tour CNC

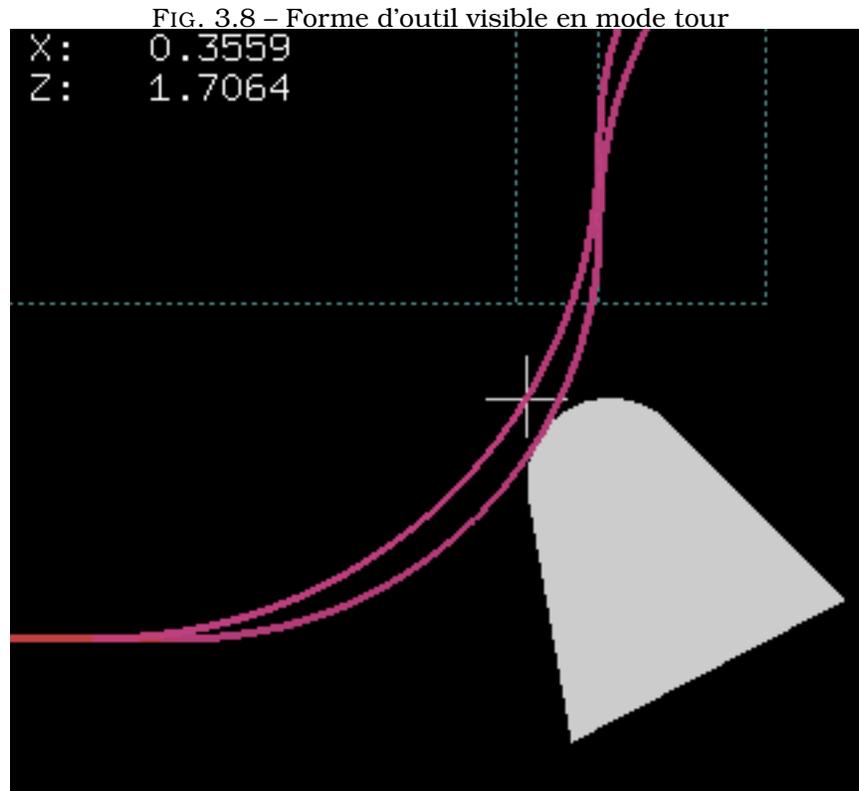
En incluant cette ligne dans le fichier ini le mode tour (lathe) sera sélectionné :

```
[DISPLAY]
LATHE = 1
```

L'axe « Y » ne sera pas visible dans l'affichage des coordonnées, la vue sera modifiée pour placer l'axe Z dans le sens gauche/droite et l'axe X dans le sens avant/arrière et différents éléments (tels que les icônes des vues prédéfinies) seront supprimés.

La touche « V » agit alors sur le zoom pour afficher le tracé complet du fichier chargé.

En mode tour (lathe), la forme de l'outil chargé est visible.



## 3.11 Configuration avancée d'AXIS

### 3.11.1 Filtres de programme

AXIS a la capacité d'envoyer des fichiers chargés à travers un « filtre de programme ». Ce filtre peut faire diverses tâches : Simple, comme s'assurer que le programme se termine bien par un : 'M2' ou complexe, comme détecter que l'entrée est une image et générer le g-code qui permettra d'usiner sa forme.

La section [FILTER] du fichier ini définit comment les filtres doivent agir. Premièrement, pour chaque type de fichier, écrire une ligne : PROGRAM\_EXTENSION puis, spécifier le programme à exécuter pour chaque type de fichier. Ce programme reçoit comme argument le nom du fichier d'entrée, il doit produire le code selon le standard rs274ngc, en sortie. Les lignes de cette sortie s'affichent alors dans la zone texte, le parcours d'outil résultant est visible dans la zone graphique, enfin il sera exécuté quand emc recevra la commande « Exécuter le programme ». Les lignes suivantes fournissent la possibilité d'utiliser « image-to-gcode », le convertisseur d'images fourni avec EMC2 :

```
[FILTER]
```

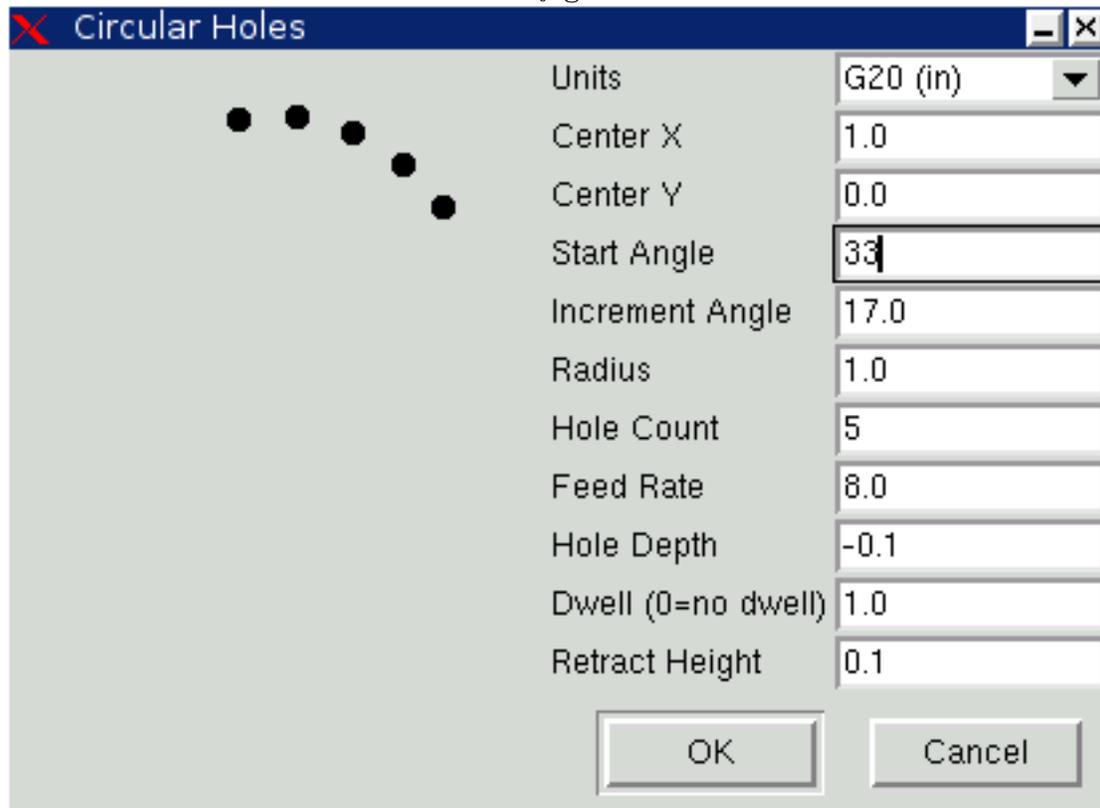
```
PROGRAM_EXTENSION = .png, .gif Greyscale Depth Image
png = image-to-gcode
gif = image-to-gcode
```

Il est également possible de spécifier un interpréteur :

```
PROGRAM_EXTENSION = .py Python Script
py = python
```

De cette manière, n'importe quel script Python pourra être ouvert et sa sortie traitée comme du g-code. Un autre exemple est disponible dans : `nc_files/holecircle.py`. Ce script crée le g-code pour percer une série de trous suivant un arc de cercle.

FIG. 3.9 – Perçages circulaires



Si la variable d'environnement : `AXIS_PROGRESS_BAR` est active, alors les lignes seront écrites sur `stderr` de la forme :

```
FILTER_PROGRESS=%d
```

AXIS fixera la barre de progression selon le pourcentage donné. Cette fonction devrait être utilisée pour un filtre qui fonctionne suffisamment longtemps.

### 3.11.2 La base de données des ressources X

Les couleurs de la plupart des éléments de l'interface utilisateur d'AXIS peuvent être personnalisées grâce à la base de données X. Le fichier : `axis_light_background` modifie les couleurs de la fenêtre du parcours d'outil sur le modèle « lignes noires et fond blanc », il sert aussi de référence des éléments configurables dans l'écran graphique.

Pour plus d'informations au sujet des éléments configurables dans les applications Tk, référez vous aux manuels de Tk.

Les bureaux graphiques modernes effectuent certains réglages dans la base de données des ressources X ces réglages peuvent affecter ceux d'AXIS, par défaut ces réglages sont ignorés. Pour que les éléments des ressources X écrasent ceux par défaut dans AXIS, il faut inclure cette ligne dans vos ressources X :

```
*Axis*optionLevel : widgetDefault
```

ce qui entraînera la construction des options au niveau « widgetDefault », de sorte que les ressources X (qui sont elles, au niveau « userDefault ») puissent l'emporter.

### 3.11.3 Manivelle de jog

Pour accroître l'interaction d'AXIS avec une manivelle de jog physique, l'axe actif courant sélectionné dans l'interface graphique est aussi reporté sur une « pin Hal » avec un nom comme `axisui.jog.x`. Excepté pendant un court instant après que l'axe courant ait changé, une seule de ces pins à la fois est TRUE, les autres restent FALSE.

Après qu'AXIS ait créé ces « HAL pins », il exécute le fichier hal déclaré avec : `[HAL]POSTGUI_HALFILE`. Ce qui diffère de `[HAL]HALFILE`, qui lui ne s'utilise qu'une seule fois.

### 3.11.4 ~/.axisrc

Si il existe, le contenu de : `~/.axisrc` est exécuté comme un code source Python juste avant l'ouverture de l'interface graphique d'AXIS. Les détails de ce qui peut être écrit dans `.axisrc` sont sujets à changement durant le cycle de développement.

Les lignes visibles sur la figure 3.10 ajoutent un `Ctrl+Q` comme raccourci clavier pour `Q`uitter et activer l'option "Distance restante" par défaut.

FIG. 3.10 – Exemple de fichier `.axisrc`

```
root_window.bind("<Ctrl+q>", "destroy .")
help2.append(("Ctrl+Q", "Quitter"))
vars.show_distance_to_go.set(1)
```

### 3.11.5 Editeur externe

En définissant : `[DISPLAY]EDITOR`, les options de menu : Fichier > Editer ainsi que Fichier > Editer la table d'outils, deviennent accessibles. Deux valeurs qui marchent bien : `EDITOR=gedit` et `EDITOR=gnome-terminal -e vim`.

### 3.11.6 Panneau de contrôle virtuel

AXIS peut afficher le fichier xml `pyVCP` dans le panneau droit de l'interface.

# Chapitre 4

## L'interface TkEMC

### 4.1 Introduction

TkEMC est l'interface utilisateur graphique la plus populaire après Axis, c'est l'interface traditionnelle d'EMC. Elle est écrite en Tcl et utilise le toolkit Tk pour l'affichage. Le fait d'être écrite en TCL la rend vraiment très portable (elle fonctionne sur une multitude de plateformes).

### 4.2 Utiliser TkEMC

Pour sélectionner l'interface graphique TkEMC avec EMC2, éditer le fichier .ini et dans la section [DISPLAY] modifier l'affichage comme ci-dessous :

```
DISPLAY = tkemc
```

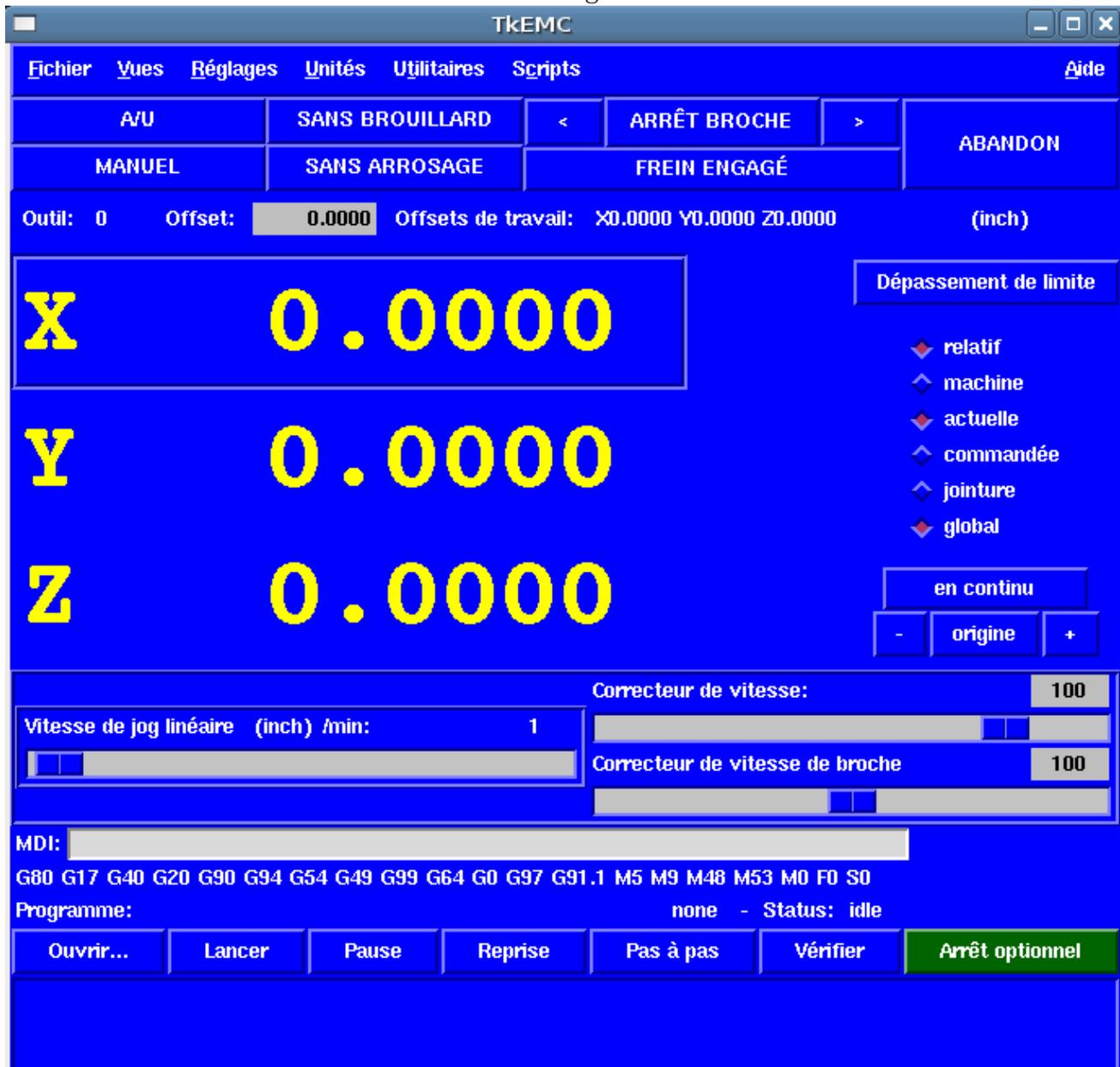
Puis, lancer EMC2 et choisir ce fichier ini. La configuration qui se trouve dans `sim/tkemc.ini` est déjà configurée pour utiliser TkEMC comme interface utilisateur.

Quand EMC2 est lancé avec TkEMC, une fenêtre comme celle de la figure 4.1 s'affiche.

#### 4.2.1 Une session typique avec TkEMC

1. Lancer EMC2 et sélectionner un fichier de configuration.
2. Libérer l'« Arrêt d'Urgence » et mettre la machine en marche (en pressant F1 puis F2).
3. Faire l'« Origine Machine » de chacun des axes.
4. Charger un fichier d'usinage.
5. Brider le brut à usiner sur la table.
6. Faire l'« Origine Pièce » de chacun des axes, à l'aide du jog ou en introduisant une valeur de décalage d'origine après un clic droit sur le nom d'un axe.
7. Lancer le programme.
8. Pour refaire une autre pièce identique, reprendre à l'étape 6. Pour usiner une pièce différente, reprendre à l'étape 4. Quand c'est terminé, quitter EMC2.

FIG. 4.1 – L'affichage de TkEMC



### 4.3 Éléments affichés par TkEMC

La fenêtre TkEMC contient les éléments suivants :

- Une barre de menu permettant diverses actions ;
- Un jeu de boutons permettant d'agir sur le mode de travail, Marche/Arrêt de la broche et autres éléments ;
- Une barre de status pour l'affichage des différents offsets ;
- Une zone d'affichage des coordonnées ;
- Un jeu de curseurs pour contrôler la « vitesse de jog », le « Correcteur de vitesse d'avance » et le « Correcteur de vitesse broche » qui permettent d'augmenter ou de diminuer ces vitesses ;
- Une boîte d'entrée de données manuelles ;
- Une barre de status affichant le bloc de programme actif, G-codes, M-codes, mots F et S ;
- Les boutons relatifs à l'interpréteur ;

– Une zone d'affichage de texte montrant le G-code du programme chargé.

### 4.3.1 Boutons principaux

Dans la première ligne de la gauche vers la droite et cycliquement :

1. Marche Machine : « Arrêt d'Urgence » / « Arrêt d'Urgence relâché » / « Marche »
2. Bascule brouillard
3. Broche moins vite
4. Direction de rotation de la broche « Arrêt broche » / « Broche sens horaire » / « Broche sens anti-horaire »
5. Broche plus vite
6. Annuler

puis dans la deuxième ligne :

1. Mode de marche : "MANUEL" / "MDI" / "AUTO"
2. Bascule d'arrosage
3. Bascule du contrôle frein de broche

### 4.3.2 Barre de status des différents offsets

Elle affiche, l'offset de rayon de l'outil courant (sélectionné avec Txx M6), l'offset éventuel de longueur d'outil si il est actif et les offsets de travail (ajustables par un clic droit sur les coordonnées).

### 4.3.3 Zone d'affichage des coordonnées

La partie principale affiche la position courante de l'outil. La couleur varie selon l'état de l'axe. Si l'axe n'est pas référencé il est affiché en caractères jaunes. Si il est référencé il s'affiche en vert. Si il est en erreur, TkEMC l'affiche en rouge pour montrer un défaut. (par exemple si un contact de fin de course est activé).

Pour interpréter correctement les différentes valeurs, se référer aux boutons de droite. Si la position est « Machine », alors la valeur affichée est en coordonnées machine. Si elle est « Relative », la valeur affichée est en coordonnées pièce. Deux autres en dessous indiquent « actuel » ou « commandé ». Actuel fait référence aux valeurs retournées par les codeurs (si la machine est équipée de servomoteurs) et « commandé » fait référence à la position à atteindre envoyée aux moteurs. Ces valeurs peuvent différer pour certaines raisons : Erreur de suivi, bande morte, résolution d'encodeur ou taille de pas. Par exemple, si un mouvement est commandé vers X0.08 sur une fraiseuse, mais qu'un pas moteur fait 0.03, alors la position « Commandée » sera 0.03 mais la position « Actuelle » sera soit 0.06 (2 pas) soit 0.09 (3 pas).

Deux autres boutons permettent de choisir entre la vue « Jointure » et la vue « Global ». Cela a peu de sens avec les machines de type normal (cinématiques triviales), mais se révèle très utile sur les machines avec des cinématiques non triviales telles que les robots ou plateforme de Stewart. (Des informations plus complètes se trouvent dans le manuel de l'intégrateur).

#### 4.3.3.1 Parcours d'outil

Quand la machine se déplace, elle laisse un tracé appelé parcours d'outil. La fenêtre d'affichage du parcours d'outil s'active via le menu Vues->Parcours d'outil.

FIG. 4.2 – Interpréteur de TkEMC



## 4.3.4 Contrôle en automatique

### 4.3.4.1 Boutons de contrôle

Les boutons de contrôle de la partie inférieure de TkEMC, voir la figure 4.2, sont utilisés pour l'exécution du programme : « Ouvrir » pour charger un fichier, « Vérifier » pour vérifier si il contient des erreurs, « Lancer » pour commencer l'usinage, « Pause » pour stopper temporairement l'usinage, « Reprise » pour reprendre un programme mis en pause, « Pas à pas » pour avancer d'une seule ligne de programme et « Arrêt optionnel » pour basculer l'arrêt optionnel, si ce bouton est vert l'exécution du programme est stoppée quand un code M1 est rencontré.

### 4.3.4.2 Zone texte d'affichage du programme

Quand un programme est lancé, la ligne courante est affichée en surbrillance blanche. L'affichage du texte défile automatiquement pour montrer la ligne courante.

## 4.3.5 Contrôle en manuel

### 4.3.5.1 Touches implicites

TkEMC permet les déplacements manuels de la machine. Cette action s'appelle le « jogging ». Premièrement, sélectionner l'axe à déplacer en cliquant dessus. Puis, cliquer et maintenir les boutons « + » ou « - » selon la direction du mouvement souhaité. Les quatres premiers axes peuvent aussi être déplacés à l'aide des touches fléchées pour les axes X et Y, Pg.préc et Pg.suiv pour l'axe Z et les touches [ et ] pour l'axe A.

Si « Continu » est activé, le mouvement sera continu tant que la touche sera pressée, si une valeur d'incrément est sélectionnée, le mobile se déplacera exactement de cette valeur à chaque appui sur la touche ou à chaque clic. Les valeurs disponibles sont :

1.0000 0.1000 0.0100 0.0010 0.0001

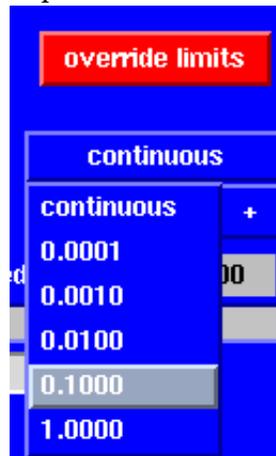
En cliquant le bouton « Origine » ou en pressant la touche Origine, l'axe actif est référencé sur son origine machine. Selon la configuration, la valeur de l'axe peut être simplement mise à la position absolue 0.0, ou la machine peut se déplacer vers un point spécifique matérialisé par le « contact d'origine ». Voir le manuel de l'intégrateur pour plus de détails sur les prises d'origine.

En cliquant le bouton « Dépassement de limite », la machine permet un jog temporaire pour même si l'axe à franchi une limite d'axe fixée dans le fichier .ini. Noter que si « Dépassement de limite » est activé il s'affiche en rouge.

### 4.3.5.2 Le groupe de boutons « Broche »

Le bouton central du dessus sélectionne le sens de rotation de la broche : Anti-horaire, Arrêt, Horaire. Les boutons fléchés augmentent ou diminuent la vitesse de rotation. Le bouton central du dessous permet d'engager ou de relâcher le frein de broche. Selon la configuration de la machine, les items de ce groupe ne sont peut être pas tous visibles.

FIG. 4.3 – Exemple de dépassement de limite et incréments de jog



#### 4.3.5.3 Le groupe de boutons « Arrosage »

Ces deux boutons permettent d'activer ou non les lubrifiants « brouillard » et « Arrosage ». Selon la configuration de la machine, les items de ce groupe ne sont peut être pas tous visibles.

#### 4.3.6 Entrée manuelle de G-code (MDI)

L'entrée manuelle de données (aussi appelée MDI), permet d'entrer et d'exécuter des lignes de G-code, une à la fois. Quand la machine n'est pas en marche ni mise en mode MDI, l'entrée de code n'est pas possible.

FIG. 4.4 – Le champ de saisie des entrées manuelles



##### 4.3.6.1 MDI :

Le mode MDI permet d'exécuter une commande en G-code en pressant la touche « Entrée ».

##### 4.3.6.2 G-Codes actifs

Ce champs montre les « codes modaux » actuellement actifs dans l'interpréteur. Par exemple, « G54 » indique que le système de coordonnées courant est celui de G54 et qu'il s'applique à toutes les coordonnées entrées.

#### 4.3.7 Vitesse de Jog

En déplaçant ce curseur, la vitesse de jog peut être modifiée. Le nombre indique une vitesse en unités par minute. Le champs de texte est cliquable. Un clic ouvre un dialogue permettant d'entrer un nombre.

### 4.3.8 Correcteur de vitesse d'avance travail

En déplaçant ce curseur, la vitesse d'avance travail peut être modifiée. Par exemple, si la vitesse d'avance travail du programme est F600 et que le curseur est placé sur 120%, alors la vitesse d'avance travail sera de 720. Le champs de texte est cliquable. Un clic ouvre un dialogue permettant d'entrer un nombre.

### 4.3.9 Correcteur de vitesse de broche

Le fonctionnement de ce curseur est le même que celui de la vitesse d'avance, mais il contrôle la vitesse de rotation de la broche. Si le programme demande S500 (broche à 500 tr/mn) et que le curseur est placé sur 80%, alors la vitesse de broche résultante sera de 400 tr/mn. Le minimum et le maximum pour ce curseur sont définis dans le fichier ini. Par défaut le curseur est placé sur 100%. Le champs de texte est cliquable. Un clic ouvre un dialogue permettant d'entrer un nombre.

## 4.4 Raccourcis clavier

La plupart des actions de TkEMC peuvent être accomplies au clavier. Beaucoup des raccourcis clavier ne sont pas accessibles en mode MDI.

Les raccourcis clavier les plus fréquemment utilisés sont montrés dans la table 4.1.

TAB. 4.1 – Les raccourcis clavier les plus utilisés

Touche	Action
F1	Bascule de l'Arrêt d'Urgence
F2	Marche/Arrêt machine
‘, 1 .. 9, 0	Correcteur vitesse d'avance 0% à 100%
X, ‘	Active le premier axe
Y, 1	Active le deuxième axe
Z, 2	Active le troisième axe
A, 3	Active le quatrième axe
Origine	POM de l'axe actif
Gauche, Droite	Jog du premier axe
Haut, Bas	Jog du deuxième axe
Pg.prec, Pg.suiv	Jog du troisième axe
[, ]	Jog du quatrième ax
Echap	Arrête l'exécution

# Chapitre 5

## L'interface MINI

### 5.1 Introduction<sup>1</sup>

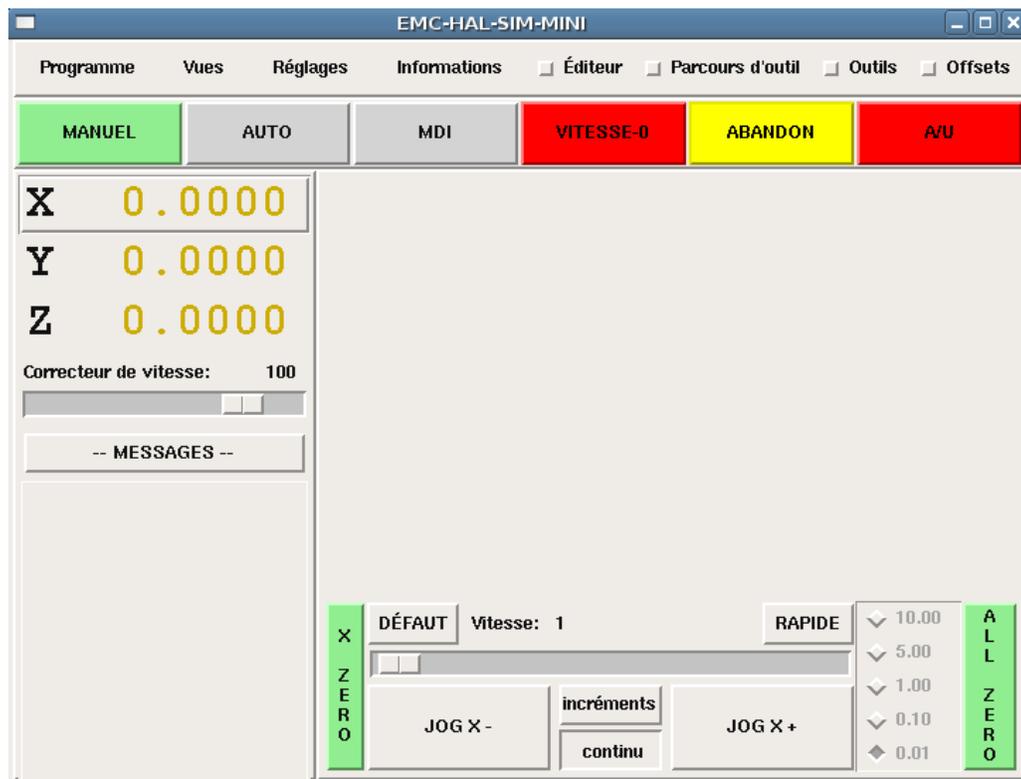


FIG. 5.1 – L'interface graphique Mini

Mini a été prévu pour être une interface graphique plein écran. Il a été écrit initialement pour Sherline CNC mais est disponible pour ceux qui veulent l'utiliser, le copier le distribuer selon les termes de la GPL.

Au lieu d'ouvrir une nouvelle fenêtre pour chaque chose que l'opérateur veut faire, Mini permet de les afficher toutes dans la même fenêtre. Certaines parties de ce chapitre sont des copies des instructions écrites par Joe Martin et Ray Henry.

<sup>1</sup>Much of this chapter quotes from a chapter of the Sherline CNC operators manual.

## 5.2 L'écran

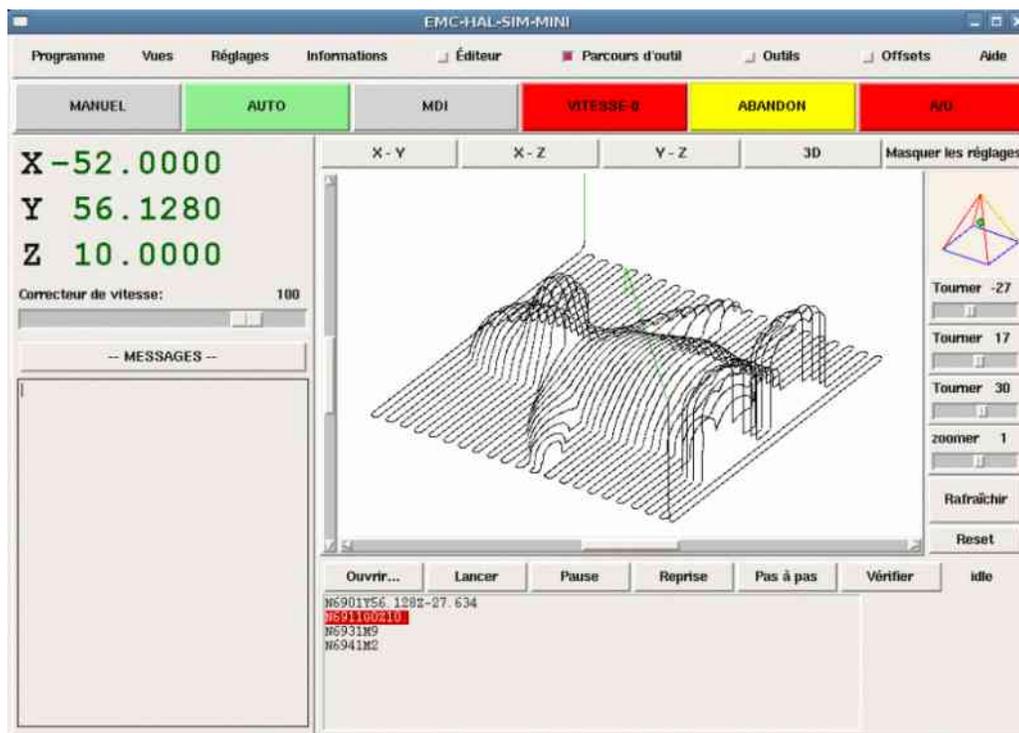


FIG. 5.2 – Affichage de Mini lors du fonctionnement d'EMC

L'écran de Mini est visible dans plusieurs sections, voir la figure 5.1. Il inclut une barre de menu, un jeu de boutons de contrôle juste en dessous et deux larges colonnes d'information montrant l'état de la machine et permettant à l'opérateur d'entrer des commandes ou des programmes.

Quand on compare entre la figure 5.1 et la figure 5.2 des différences apparaissent. Dans la seconde figure :

- Chaque axe a été référencé sur son origine – L'affichage est vert sombre
- EMC est en mode auto – le bouton est vert clair
- Le parcours d'outil a été activé – le parcours est visible dans la fenêtre
- Le bloc de G-code en cours est en surbrillance blanche sur fond rouge parmi les lignes défilantes du programme.

Dès le début de l'utilisation de Mini on découvre combien il est facile de visualiser le fonctionnement d'EMC et d'agir sur lui.

## 5.3 Barre de menu

La première ligne est occupée par la barre de menu. Elle permet de configurer les informations optionnelles que l'écran doit afficher. Certains des items du menu sont très différents de ce qu'il est coutume de rencontrer avec d'autres programmes. Il faut prendre quelques minutes pour regarder chaque item du menu et se familiariser avec les possibilités qu'il offre.

Le menu inclut chacune des sections et sous-sections suivantes :

**Programme** ce menu inclut les deux fonctions « reset » et « quitter ». Reset renvoie EMC dans l'état dans lequel il était quand il a démarré. Certains items de conditions de démarrage comme les unités du programme peuvent être spécifiés dans le fichier ini.

**Vues** ce menu inclus plusieurs éléments d'affichage pouvant être ajoutés pendant qu'un programme s'exécute, pour donner des informations supplémentaires. Ce qui inclus :

**Type\_de\_position** cet item ajoute une ligne, au dessus de l'afficheur principal de position, qui indique si la position affichée est en coordonnées machine (absolues) ou en coordonnées pièce (relatives), si la position affichée est actuelle ou commandée et si l'unité est le pouce ou le millimètre. Ce qui peut être modifié en agissant sur le menu « Réglages » décrit plus loin.

**Informations\_outil** cet item ajoute une ligne immédiatement sous l'afficheur principal, elle affiche quel outil est sélectionné et la valeur de l'offset appliqué.

**Informations\_offset** ajoute une ligne immédiatement sous celle des informations outil, elle affiche les offsets de travail actuellement appliqués. C'est la distance totale pour chaque axe depuis son point d'origine machine.

**Afficher\_Reprise** ajoute, en mode Auto, un bloc de boutons en bas à droite de l'affichage du programme. Ces boutons permettent à l'opérateur de relancer un programme après un A/U ou un Abandon. Le programme est figé lorsqu'un A/U ou un Abandon est engagé, mais l'opérateur peut ici choisir le bloc sur lequel redémarrer dans le mode Auto, en activant cet item de menu.

**Masquer\_Reprise** enlève le bloc de boutons contrôlant la reprise d'un programme stoppé par Abandon ou Arrêt d'Urgence.

**Diviser\_la\_vue\_droite** modifie la nature de l'affichage de la colonne de droite pour qu'il montre plus d'informations.

**Afficher\_en\_mode\_complet** modifie l'affichage de la colonne de droite, soit en mode boutons et la colonne divisée, soit affichage sur la colonne entière. En mode manuel et en utilisant ce mode entier, les boutons de la broche et du lubrifiant sont visibles en plus des boutons de mouvement.

**Afficher\_dans\_toute\_la\_fenêtre** modifie l'affichage dans la colonne de droite pour qu'elle soit remplie entièrement.

**Réglages** Les items de ce menu permettent à l'opérateur de contrôler certains paramètres en cours d'usinage.

**Position\_Actuelle** règle l'afficheur principal sur des valeurs de position actuelle (basées sur la machine).

**Position\_Commandée** règle l'afficheur principal sur des valeurs de position commandée (positions cibles).

**Position\_Machine** règle l'afficheur principal sur des valeurs de position absolue, distance depuis le point d'origine machine.

**Position\_Relative** règle l'affichage principal pour qu'il affiche les valeurs de la position courante en incluant tous les offsets comme les origines pièces qui seraient actives. Pour plus de détails voir le chapitre sur les « Systèmes de coordonnées » .

**Informations** Indique diverses activités en affichant des valeurs dans la zone MESSAGES.

**Fichier\_programme** Indique le nom du fichier programme actif.

**Editeur\_de\_Fichiers** Indique le nom de l'éditeur de texte lancé quand un fichier est choisi pour être édité.

**Fichier\_de\_paramètres** Indique le nom du fichier devant être utilisé pour enregistrer les paramètres du programme. Il est possible d'avoir plus d'informations dans le chapitre sur les offsets et l'utilisation des paramètres dans les programmes.

**Fichiers\_d'outils** Indique le nom du fichier d'outils utilisé pour l'usinage en cours.

**G-Codes\_actifs** Indique la liste des G-codes modaux actifs au moment où cet item a été sélectionné. Pour plus d'informations sur les codes modaux, se reporter aux chapitres sur la programmation.

**Aide** Ouvre une fenêtre de texte contenant un fichier d'aide.

Noter que entre le menu informations et le menu d'aide il y a un jeu de quatre cases à cocher. Elles sont appelées cases sélectionnables car elles passent au rouge si elles sont sélectionnées. Ces quatre cases, Editeur, Parcours d'outil, Outils et Offsets activent les différents écrans. Si plus d'une de ces cases est validée (case en rouge) il est possible de passer de l'un à l'autre des écrans surgissant par un clic droit de la souris.

## 5.4 Barre de boutons de contrôle

Sous la ligne de menu se trouve une ligne de boutons de contrôle. Ce sont les boutons principaux de l'interface. En utilisant ces boutons il est possible de changer de mode cycliquement entre « MANUEL » puis « AUTO » puis « MDI » (Manual Data Input). Ces boutons ont un fond vert clair quand le mode correspondant est actif.

Il est également possible d'utiliser les boutons « VITESSE-0 », « ABANDON » et « A/U » pour contrôler les mouvements du programme.

### 5.4.1 MANUEL

Ce bouton, ou la touche « F3 » placent EMC en mode Manuel et affiche un jeu condensé de boutons que l'opérateur peut utiliser pour effectuer des commandes de mouvements manuels. Les labels des boutons de jog varient pour refléter l'axe actif. Quand en plus de ce mode manuel, la vue en mode complet est activée, les boutons de la broche et du lubrifiant sont visibles en plus de ceux de mouvement. La touche « i » ou « I » basculera entre jog en continu et jog par incréments. Presser cette touche plusieurs fois provoquera le changement cyclique de la taille de l'incrément de jog.

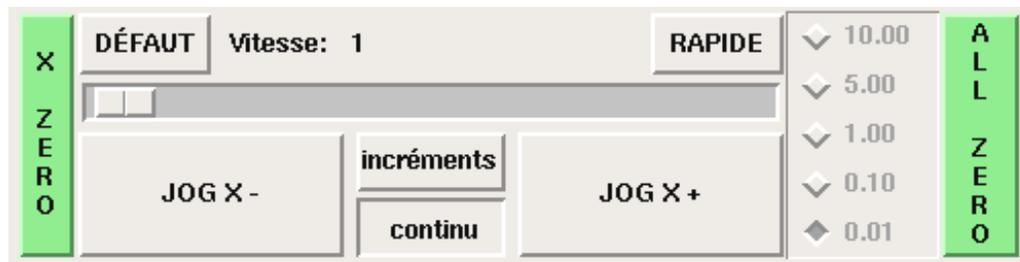


FIG. 5.3 – Boutons du mode Manuel

Un bouton a été ajouté pour désigner la position actuelle comme étant la position d'origine. Une machine simple telle que la Sherline 5400 serait facile à utiliser sans avoir de position d'origine. Ce bouton met les offsets de tous les axes à zéro et place l'origine de tous les axes au point courant.

Ce concentrer sur les axes est important ici. Noter que sur la capture 5.1 qui est en mode manuel, on voit un cadre autour de l'axe X pour mettre son affichage en évidence. Ce cadre (le focus) indique que l'axe X est l'axe actif. Il est l'axe cible pour les mouvements de jog faits par appui sur les boutons *plus* ou *moins*. Il est possible de changer l'axe actif en cliquant sur l'affichage d'un autre axe. C'est également possible en mode manuel en pressant la touche de son nom sur le clavier. La casse n'a pas d'importance « Y » ou « y » donneront le focus à l'axe Y. « A » ou « a » le donneront à l'axe A. Pour aider à se rappeler quel est l'axe actif avant de faire un jog, son nom est indiqué sur les boutons de jog.

EMC peut faire un jog (mouvement d'un axe particulier) aussi longtemps que le bouton est maintenu pressé, si le jog est réglé sur *continu*, ou il peut se déplacer d'une valeur prédéfinie, quand il est réglé sur *incréments*. Il est aussi possible de faire un jog de l'axe actif en pressant les touches « + » ou moins « - » du clavier. De nouveau, la casse n'a pas d'importance pour les jogs au clavier. Les deux petits boutons situés entre les deux gros

du jog, permettent de choisir quel type de jog est souhaité. Quand on est en mode incréments, les boutons de taille d'incrément sont accessibles. La sélection d'une taille d'incrément se fait par clic sur la case à cocher avec la souris ou cycliquement en pressant la touche « i » ou « I » sur le clavier. Le jog par incréments présente quelques effets intéressants autant qu'inattendus. Si vous pressez le bouton de jog alors qu'un mouvement de jog est déjà en cours, la distance à laquelle il était lorsqu'est arrivée la commande du second jog, sera ajoutée à la position. Deux pressions successives rapides de 10mm d'incrément ne donneront pas 20mm de mouvement. Vous devez attendre que le premier soit terminé avant d'envoyer le second.

La vitesse de jog est affichée au dessus du curseur. Il est possible de régler le curseur en cliquant dans la glissière du curseur, du côté où vous voulez son déplacement, ou en cliquant sur les boutons « Défaut » ou « Rapide ». Ces réglages n'affectent que les mouvements de jog en mode manuel. Tant qu'un mouvement de jog est en cours, un changement de la vitesse de jog est sans effet sur le jog. Par exemple, disons que vous avez réglé le jog par incréments de 10mm. Même si vous pressez le bouton « Jog », le déplacement de dix millimètres se finira à la vitesse à laquelle il a commencé.

### 5.4.2 AUTO

Quand le bouton Auto est pressé, ou « F4 » sur le clavier, EMC passe dans ce mode, un jeu de boutons traditionnels des opérations en auto est affiché et une petite fenêtre textuelle s'ouvre, montrant une partie du programme. Durant un usinage, la ligne active est affichée en surbrillance blanche sur fond rouge.

Dans le mode auto, beaucoup de touches sont liées aux contrôles. Par exemple, les touches numérotées au dessus du clavier sont liées aux réglages du correcteur de vitesse d'avance travail. Le « 0 » l'ajuste à 100%, le « 9 » l'ajuste à 90% etc. D'autres touches fonctionnent de la même manière qu'avec l'interface TkEMC.

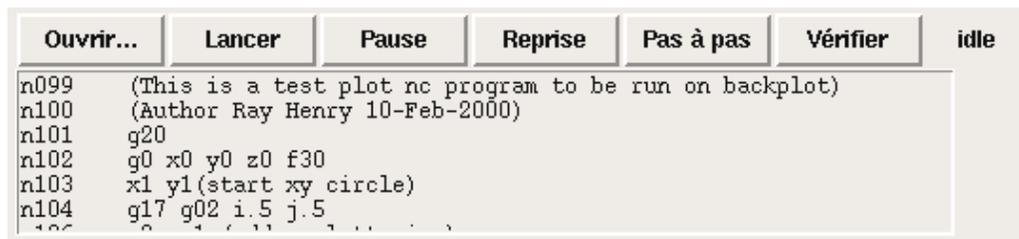


FIG. 5.4 – Mode Auto

Le mode Auto n'affiche normalement pas les G-codes actifs ou modaux. Si l'opérateur veut les voir, il doit utiliser le menu « Informations -> G-codes actifs » et afficher ainsi la liste des codes modaux dans la zone de texte MESSAGES.

Si un Arrêt d'Urgence ou un Abandon est pressé pendant un usinage, un jeu de boutons s'affiche en bas et à droite, ils permettent à l'opérateur de décaler la ligne de reprise vers l'avant ou vers l'arrière. Si la ligne de reprise n'est pas la dernière ligne active, elle sera mise en surbrillance blanche sur fond bleu. ATTENTION, une vitesse très faible et un doigt collé sur le bouton de pause est prudent pendant toute reprise de programme !

Ce qui est le mieux avec une machine CNC, c'est le mode auto. Le mode auto affiche les fonctions typiques que tout le monde espère utiliser avec EMC. Au dessus de la fenêtre un jeu de boutons qui contrôlent ce qui se passe en mode auto. En dessous, la fenêtre montrant la partie du programme en cours d'exécution. Quand le programme est lancé, la ligne active s'affiche en surbrillance blanche sur fond rouge. Les trois premiers boutons, « Ouvrir », « Lancer » et « Pause » font ce que leurs noms indiquent. « Pause » stoppe le programme en cours d'exécution là où il est. Le bouton suivant, « Reprise », reprend le

mouvement. Le résultat est le même si le bouton « Vitesse-0 » était pressé au lieu du bouton « Pause », le mouvement est stoppé, « Pas à pas » reprend aussi le mouvement, mais il ne continue que jusqu'à la fin du bloc courant. Presser une nouvelle fois « Pas à pas » exécutera le mouvement du bloc suivant. Presser « Reprise » à ce moment là et l'interpréteur reviendra en arrière lire et relancer le programme. La combinaison entre « Pause » et « Pas à pas » marche un peu comme un seul bloc dans plusieurs interpréteurs. Avec la différence que « Pause » ne laisse pas le mouvement se poursuivre jusqu'à la fin du bloc courant. Le correcteur de vitesse d'avance travail peut se révéler très pratique pour s'approcher de la matière pour un premier usinage. Le placer sur 100% pour les déplacements rapides, le régler sur 10% et basculez entre « Vitesse-0 » et 10% en utilisant le bouton « Pause ». Quand vous êtes satisfait et que les choses se présentent bien, pressez le zéro à la droite du neuf en haut du clavier, et c'est parti.

Le bouton « Vérifier » passe le code dans l'interpréteur sans production de mouvements. Si Vérifier trouve un problème il stoppe la lecture juste derrière le bloc posant problème et affiche un message. La plupart du temps vous serez en mesure de régler le problème avec votre programme par la lecture du message et en vérifiant la ligne de code en surbrillance dans la fenêtre du programme. Certains messages, toutefois, ne sont pas d'un grand secours. Parfois vous devrez lire une ou deux lignes en avant de celle en surbrillance pour voir le problème. Occasionnellement le message fait référence à quelque chose très en avant de la ligne en surbrillance. Le plus souvent ça se produit si vous oubliez de terminer votre programme par un code correct comme %, M2, M30, ou M60.

### 5.4.3 MDI

Le bouton MDI ou la touche « F5 » activent le mode d'entrée de données en manuel (Manual Data Input). Ce mode affiche un simple champ de saisie d'une ligne de texte et montre les codes modaux actuellement actifs dans l'interpréteur.

Le mode MDI vous permet d'entrer de simples blocs et de les faire exécuter par l'interpréteur comme si ils étaient une partie d'un programme (Une sorte de programme d'une seule ligne). Vous pouvez exécuter des cercles, des arcs, des lignes et autres. Vous pouvez aussi mettre au point une ligne de programme en entrant cette ligne comme un seul bloc, attendre que le mouvement se termine et entrer le bloc suivant. Sous la fenêtre d'entrée, se trouve une liste des codes modaux courants. Cette liste peut être très pratique. J'oublie parfois d'entrer un G0 avant une commande de mouvement. Si rien ne se passe je regarde dans la liste si G80 est actif. G80 stoppe tous les mouvements. Si il y est je me rappelle de mettre un bloc comme G00 X0 Y0 Z0. Dans le MDI, vous entrez du texte depuis le clavier, ainsi toutes les touches principales ne fonctionnent pas comme raccourcis clavier pour les commandes machine. « F1 » engage l'Arrêt d'Urgence.

Puisque les touches du clavier sont nécessaires à la saisie du texte, beaucoup des raccourcis clavier disponibles en mode auto ne le sont pas ici.

### 5.4.4 « VITESSE-0 » – « CONTINUER »

Vitesse-0 est une bascule. Quand EMC est prêt pour exécuter, ou qu'il exécute une commande de mouvement, ce bouton affiche son label « VITESSE-0 » sur fond rouge. Si Vitesse-0 a été pressé il affiche le label « CONTINUER ». Utiliser ce bouton pour faire une pause dans un mouvement présente l'avantage d'être capable de relancer le programme d'où il a été stoppé. Vitesse-0 bascule simplement entre vitesse zéro et la vitesse d'avance travail avec l'éventuel correcteur qui était actif au moment de l'arrêt. Ce bouton et la fonction qu'il active sont également liés à la touche pause de la plupart des claviers.

### 5.4.5 « ABANDON »

Le bouton Abandon stoppe tous les mouvements quand il est pressé. Il désactive aussi la commande de marche d'EMC. Plus aucun mouvement ne survient après l'appui sur ce bouton. Si vous êtes en mode auto, ce bouton enlève le reste du programme du planificateur de mouvements. Il enregistre aussi le numéro de la ligne qui s'exécutait quand il a été pressé. Vous pouvez vous servir de ce numéro de ligne pour redémarrer le programme après avoir supprimé la raison de l'appui...

### 5.4.6 « A/U »

Le bouton d'Arrêt d'Urgence est une bascule mais il a trois fonctionnements possibles.

- Au démarrage de Mini c'est un bouton avec le texte « A/U » écrit en noir sur fond rouge. C'est état de la machine est correct pour charger un programme ou faire un jog sur un axe. L'Arrêt d'Urgence est libéré quand il s'affiche dans cet état.
- Si vous pressez sur le bouton d'Arrêt d'Urgence pendant qu'un mouvement est exécuté, le texte sur le bouton devient « A/U Engagé » sur fond gris et le bouton s'enfonce. Plus aucun mouvement n'est possible et plus rien ne réagit sur l'interface Mini tant que l'Arrêt d'Urgence est dans cet état. Le presser à nouveau à la souris le fera repasser en conditions normales de fonctionnement.
- Un troisième état est encore possible. Un bouton enfoncé portant le texte « A/U Libéré » sur fond vert signifie que l'A/U a bien été libéré mais que la machine n'a pas été mise en marche. Normalement cet état apparaît quand l'A/U était libéré mais que la touche Marche Machine « F2 » a été pressée.

Joe Martin disait, "Quand tout le reste a échoué, pressez un « Arrêt d'Urgence » software." Si vous avez un circuit externe qui gère l'A/U en surveillant une broche du port parallèle ou celle d'une carte d'entrées/sorties, un arrêt d'urgence software pourra couper la puissance sur les moteurs.

La plupart du temps, quand un Abandon ou un Arrêt d'Urgence est engagé c'est parce que quelque chose se passe mal. Peut être un outil cassé et qui doit être changé. On passe alors en mode manuel et on arrête la broche, on change l'outil et en supposant que sa longueur reste la même, on est prêt pour relancer le programme. Si on renvoie l'outil à la même place ou il était avant, EMC va fonctionner parfaitement. Il est aussi possible de se déplacer sur la ligne suivante ou précédente de celle où c'est produit l'abandon. Si vous pressez le bouton « Arrière » ou « Avant » vous voyez une ligne en surbrillance bleue montrant l'écart entre la ligne sur laquelle l'abandon s'est produit (restée en surbrillance rouge) et la ligne à laquelle EMC va redémarrer. En réfléchissant à ce qui va se produire au moment de la reprise vous serez en mesure de placer l'arête de l'outil là où la reprise pourra se faire de manière acceptable. Vous aurez peut être à solutionner certaines difficultés comme celles créées par les compensations de rayon d'outil le long d'une ligne diagonale et vous devrez être sûr de vous avant de presser sur le bouton « Reprise ».

## 5.5 Colonne de gauche

Il y a deux colonnes sous la barre de contrôle. La colonne de gauche affiche les informations intéressant l'opérateur. Il y a seulement deux boutons dans cette zone.

### 5.5.1 Afficheurs de position des axes

Ces afficheurs se comportent exactement comme ceux de TkeEMC. La couleur des afficheurs est importante.

- Rouge, elle indique que la machine est en appui sur un contact de fin de course ou que la polarité d'une limite est mal positionnée dans le fichier ini.

- Jaune, elle indique que la machine est prête pour faire ses prises d'origine.
- Verte, elle indique que la machine a bien été référencée sur ses points d'origine.

Le type de position affichée peut varier, selon les options choisies dans le menu « Réglages ». Les réglages par défaut, ou de démarrage, peuvent être changés dans le fichier ini pour correspondre à vos besoins.

### 5.5.2 Correcteur de vitesse travail

Immédiatement sous les afficheurs de position on trouve un curseur, c'est le correcteur de vitesse travail. Vous pouvez agir sur le correcteur de vitesse et sur le bouton Vitesse-0 dans tous les modes de marche. Le correcteur agit sur la vitesse de jog et sur la vitesse d'avance travail dans les modes manuel ou MDI. Il est possible de modifier la position du curseur en le déplaçant à la souris le long de sa glissière. Il est également possible de modifier le correcteur de 1% à chaque fois qu'un clic de souris est fait dans la glissière du curseur. En mode Manuel il est possible d'ajuster le correcteur par incréments de 10% avec les touches chiffrées du haut du clavier. Le curseur est une référence visuelle très pratique pour estimer la correction appliquée sur la vitesse d'avance programmée.

### 5.5.3 Messages

L'affichage des messages situé sous le curseur du correcteur de vitesse est une sorte de bloc-notes pour EMC. Si un problème survient, il est reporté sur ce bloc-notes. Si vous essayez de déplacer un axe alors que l'Arrêt d'Urgence est engagé, vous recevez un message qui dit quelques choses à propos des conditions de marche empêchant EMC de répondre à la commande de mouvement. Si un axe est en défaut, par exemple un dépassement de limite, le message affiché sur le bloc-notes indiquera ce qui se passe. Pour demander à l'opérateur de changer d'outil par exemple, vous pouvez aussi ajouter une ligne de code dans le programme qui s'affichera sur l'écran, dans la boîte de messages. Un exemple pourrait être : (msg, Montez l'outil N°3 puis pressez Reprise). Cette ligne de code, incluse dans un programme, va afficher « Montez l'outil N°3 puis pressez Reprise » dans la boîte de messages. Le mot msg, (avec la virgule) est la commande qui fait apparaître le commentaire, sans msg, le message ne serait pas affiché.

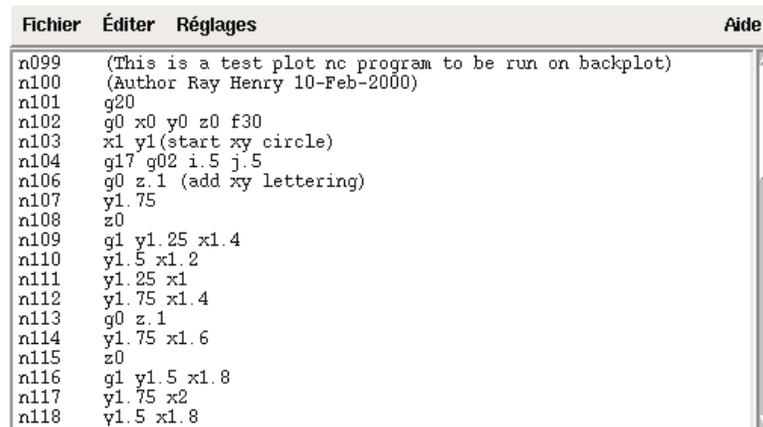
Pour effacer les messages cliquer simplement sur le bandeau -MESSAGES- au dessus du bloc-notes ou au clavier, presser « m » tout en maintenant la touche « Alt » appuyée.

## 5.6 Colonne de droite

La colonne de droite est l'emplacement sur lequel seront affichés les différents éléments résultants des choix de l'utilisateur. C'est ici que seront visibles les différentes entrées de texte, les différents affichages et les boutons du mode manuel. C'est ici que le parcours d'outil sera affiché pendant l'exécution d'un programme. C'est également ici que l'éditeur de texte s'ouvrira pour permettre l'édition des programmes, l'édition des tables d'outils ou d'offsets. L'écran du mode manuel a déjà été décrit précédemment. Chaque écran surgissant va être décrit en détail ci-dessous.

### 5.6.1 Editeur de texte

L'éditeur de texte de Mini peut sembler un peu limité par rapport aux éditeurs de texte modernes. Il a été inclus parce qu'il comporte l'excellente possibilité de numéroter et renuméroter un programme de la même manière que l'interpréteur le fait avec un fichier. Il permet également le couper/coller d'une partie vers une autre du fichier. En plus, il permet d'enregistrer les changements faits dans le programme et de soumettre celui-ci à l'interpréteur d'EMC depuis le même menu. Il est possible de travailler sur un fichier ouvert dans cet éditeur puis de l'enregistrer et de le recharger si EMC est en mode Auto. Si vous avez lancé un fichier et que vous avez besoin de l'éditer, ce fichier sera placé dans l'éditeur quand vous cliquez sur le bouton « Editeur » du menu.



```

Fichier  Éditer  Réglages  Aide
n099  (This is a test plot nc program to be run on backplot)
n100  (Author Ray Henry 10-Feb-2000)
n101  g20
n102  g0 x0 y0 z0 f30
n103  x1 y1(start xy circle)
n104  g17 g02 i.5 j.5
n106  g0 z.1 (add xy lettering)
n107  y1.75
n108  z0
n109  g1 y1.25 x1.4
n110  y1.5 x1.2
n111  y1.25 x1
n112  y1.75 x1.4
n113  g0 z.1
n114  y1.75 x1.6
n115  z0
n116  g1 y1.5 x1.8
n117  y1.75 x2
n118  y1.5 x1.8

```

FIG. 5.5 – Editeur de texte de Mini

### 5.6.2 Parcours d'outil

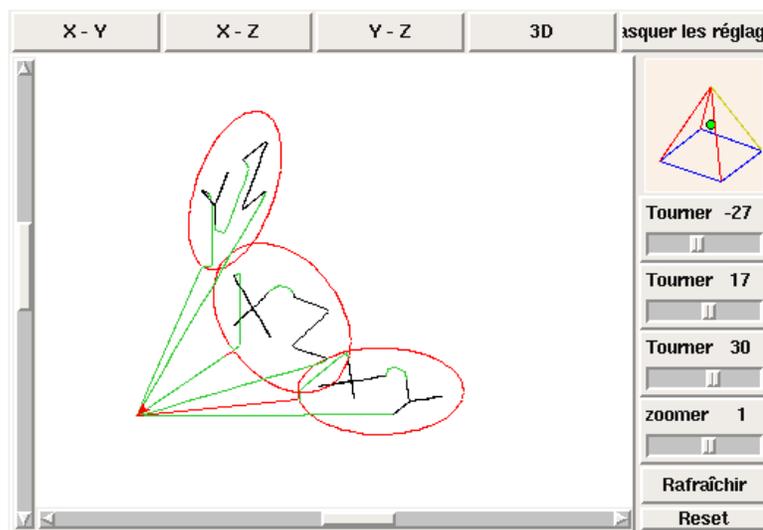


FIG. 5.6 – Parcours d'outil de Mini

Le bouton « Parcours d'outil » du menu, affiche un tracé produit par le cheminement de l'outil pendant l'usinage. Cette trace est affichable selon plusieurs plans et en « 3-D » qui est le mode par défaut. Les choix sont affichés au dessus et à droite de cette fenêtre. Si l'usinage est déjà en cours depuis un certain temps quand cette fenêtre est activée, il lui faudra un peu de temps pour recalculer la vue du parcours.

Sur le côté droit de la fenêtre une petite pyramide graphique montre l'angle depuis lequel l'opérateur voit le tracé du parcours d'outil. En dessous, une série de curseurs permettent de modifier les angles et la taille de la vue. Il est possible de contrôler les modifications en regardant l'attitude de la petite pyramide. Les modifications ne prendront effet qu'après appui sur le bouton « Rafraîchir ». Le bouton « Reset » efface toutes les traces du parcours affiché et prépare la fenêtre pour un nouveau lancement du programme, il conserve toutefois les orientations et le niveau de zoom actuellement définis.

Si un parcours d'outil est lancé avant qu'un programme ne soit démarré, il va afficher quelques lignes de couleur pour indiquer le type de tracé qui sera utilisé. Une ligne verte représente un mouvement en vitesse d'avance rapide. Une ligne noire, un mouvement en vitesse d'avance travail. Une ligne bleue et une ligne rouge indiquent respectivement un arc en sens anti-horaire et un arc en sens horaire.

Le tracé du parcours d'outil de Mini permet de zoomer et d'orienter les vues du programme en cours d'exécution mais il n'est pas conçu pour enregistrer un chemin d'outil sur une longue période de temps.

### 5.6.3 Page des outils

La page des outils permet d'ajuster la longueur et le diamètre des outils, ces valeurs deviennent effectives dès l'appui sur la touche « Entrée ». Il est nécessaire de définir les paramètres des outils avant de lancer le programme. Les offsets d'outils ne peuvent pas être modifiés une fois le programme démarré ni quand il est en pause.

RÉGLAGE OUTIL			
Cliquer ou « tab » pour éditer. Presser « Entrée » pour retourner au contrôle de la machine depuis le clavier.			
NUMÉRO D'OUTIL	LONGUEUR	DIAMÈTRE	COMMENTAIRE
2	0.0	1.0	forêt de 1
3	0.0	2.0	forêt de 2
4	0.0	4.0	fraise 2 dents

Ajouter un outil supplémentaire      Enlever le dernier outil

FIG. 5.7 – Affichage de la table d'outils de Mini

Les boutons « Ajouter un outil » et « Enlever le dernier outil » portent sur la fin de la liste des outils de sorte que l'ajout d'outils s'effectuent vers le bas. Quand un nouvel outil est ajouté, il est utilisable dans le programme avec la commande G-code habituelle. Le nombre maximum d'outils est de 32 dans le fichier de configuration d'EMC mais vous serez arrivé au bout des possibilités d'affichage de Mini bien avant cela. A la place vous pouvez utiliser le « menu » -> « Vues » -> « Afficher dans toute la fenêtre » pour voir plus d'outils si nécessaire.

### 5.6.4 Page des offsets

La page des offsets peut être utilisée pour afficher et ajuster les décalages d'origine des différents systèmes de coordonnées. Le système de coordonnées est choisi dans la colonne de gauche, par sélection d'une case à cocher. Quand un système de coordonnées est choisi il est possible d'entrer directement les valeurs ou de déplacer l'axe à une position d'apprentissage. Il est également possible

RÉGLAGE COORDONNÉES SYSTÈME		
Cliquer la valeur à éditer au clavier. Presser « entrée » pour revenir au contrôle de la machine depuis le clavier.		
<input type="checkbox"/> G54	Axe	Valeur
<input checked="" type="checkbox"/> G55	X	0.000000    Apprentissage
<input type="checkbox"/> G56	Y	0.000000    Apprentissage
<input type="checkbox"/> G57	Z	0.000000    Apprentissage
<input type="checkbox"/> G58	Offset par rayon	0.000000
<input type="checkbox"/> G59	Offset par longueur	0.000000
<input type="checkbox"/> G59.1	<input type="checkbox"/> Soustraire	<input checked="" type="checkbox"/> Ajouter
<input type="checkbox"/> G59.2	<input type="button" value="Écrire et charger fichier"/>	
<input type="checkbox"/> G59.3	<input type="button" value="Écrire et charger fichier"/>	

FIG. 5.8 – Mini Offset Display

d'utiliser une pinnule puis d'ajouter le rayon et la longueur par le bouton « Apprentissage ». Pour cela, il peut être nécessaire d'ajouter ou de soustraire le rayon de la pinnule selon la surface qui est touchée. C'est la case à cocher « Soustraire » ou « Ajouter » qui permet ce choix.

Le bouton « Tous les Gxx à zéro » enlève tous les décalages du système courant visibles sur l'écran mais ne modifie pas les variables dans le fichier tant que le bouton « écrire et recharger » n'est pas

pressé. Le bouton « écrire et recharger » est à presser pour valider et recharger tous les changements effectués sur les valeurs des systèmes de coordonnées.

## 5.7 Raccourcis clavier

Un certain nombre de raccourcis clavier utilisés avec TkEMC ont été conservés avec Mini. Quelques uns ont changé pour étendre leurs fonctions ou pour faciliter les opérations sur une machine utilisant cette interface. Certaines touches opèrent de la même manière quelque soit le mode. D'autres change en fonction du mode de travail d'EMC.

### 5.7.1 Touches courantes

**Pause** Bascule du correcteur de vitesse travail

**Echap** Abandonne le mouvement

**F1** Bascule l'état de l'Arrêt d'Urgence

**F2** Marche/Arrêt machine

**F3** Mode manuel

**F4** Mode Auto

**F5** Mode MDI

**F6** Réinitialise l'interpréteur

Ces touches fonctionnent seulement pour une machine utilisant des entrées sorties auxiliaires

**F7** Marche/Arrêt du brouillard

**F8** Marche/Arrêt arrosage

**F9** Marche/Arrêt broche sens horaire

**F10** Marche/Arrêt broche sens anti-horaire

**F11** Diminue la vitesse de broche

**F12** Augmente la vitesse de broche

### 5.7.2 Mode Manuel

**1-9 0** Positionne le correcteur de vitesse d'avance travail de 10% en 10%, de 0 à 100%

~ Positionne le correcteur de vitesse à 0

**x** Sélectionne l'axe X

**y** Sélectionne l'axe Y

**z** Sélectionne l'axe Z

**a** Sélectionne l'axe A

**b** Sélectionne l'axe B

**c** Sélectionne l'axe C

**Gauche Droite** jog de l'axe X

**Haut Bas** jog de l'axe Y

**Pg.préc Pg.suiv** jog de l'axe Z

- \_ jog de l'axe actif dans la direction moins

+ = jog l'axe actif dans la direction plus

**Origine** POM de l'axe actif

**i I** Bascule cyclique des incréments de jog

Les touches suivantes ne fonctionnent que sur les machines utilisant des entrées sorties auxiliaires

**b** Relâcher le frein de broche

**Alt-b** Engager le frein de broche

### 5.7.3 Mode Auto

**1-9,0** Positionne le correcteur de vitesse d'avance travail de 10% en 10%, de 0 à 100%

~ Positionne le correcteur de vitesse à 0

**o/O** Ouvre un programme

**r/R** Lance le programme ouvert

**p/P** Met le programme en pause

**s/S** Reprise d'un programme en pause

**a/A** Avance d'une ligne dans un programme en pause

## 5.8 Divers

Une des possibilités de Mini est d'afficher n'importe quel axe, au delà de 2, comme étant un axe rotatif, il affiche alors des degrés pour celui-ci. Il converti également les unités en degrés quand un axe rotatif a le focus.

# Chapitre 6

## L'interface KEYSTICK

### 6.1 Introduction

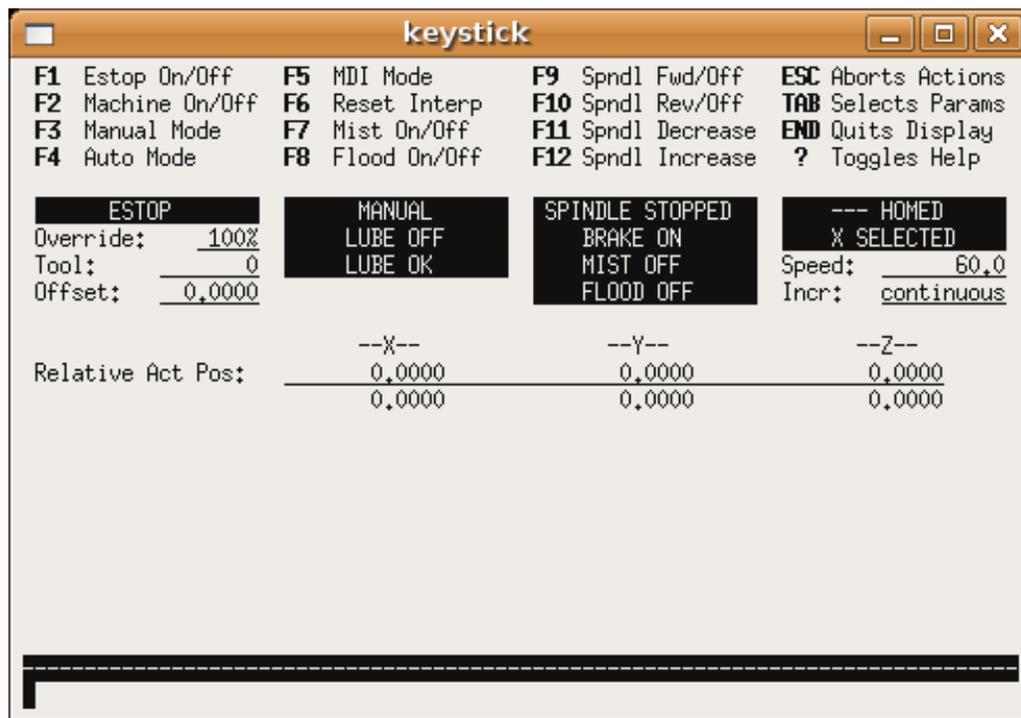


FIG. 6.1 – L'interface utilisateur minimale

Keystick est une interface minimaliste textuelle.

### 6.2 Installation

Pour utiliser « keystick » modifier la section DISPLAY du fichier ini de la manière suivante :

[DISPLAY]

DISPLAY = keystick

## 6.3 Utilisation

Keystick est extrêmement simple à utiliser. Dans le mode MDI vous pouvez directement taper le G-code et il sera visible dans la partie supérieur. La touche « ? » bascule avec ou sans aide.

1

---

<sup>1</sup>ndt : keystick a été conçu pour être minimaliste, par conséquent son interface n'est pas traductible.

**Troisième partie**

**L'utilisation d'EMC2**

# Chapitre 7

## Vue d'ensemble d'un centre d'usinage

Cette section donne une brève description des interactions entre les organes d'un centre d'usinage et son interpréteur de commandes. Le lecteur est sensé être déjà familier avec les centres d'usinage.

Le langage RS274/NGC et les fonctions d'usinage standards qui en découlent permettent d'envisager un centre d'usinage sous deux points de vue :

- (1) les composants mécaniques de la machine.
- (2) les composants de contrôle et les données utilisées pour le contrôle de la machine.

### 7.1 Composants mécaniques

Un centre d'usinage dispose de beaucoup de composants mécaniques pouvant être contrôlés ou qui peuvent avoir une incidence sur la façon dont le contrôle de la machine s'effectue. Cette section décrit les composants qui interagissent avec l'interpréteur. Les composants mécaniques, qui n'interagissent pas avec l'interpréteur, comme les boutons du jog, ne seront pas décrits ici, même si ils affectent le contrôle.

#### 7.1.1 Axes

Tout centre d'usinage dispose d'un ou de plusieurs axes. Les différents types de machines ont différentes combinaisons d'axes. Par exemple, une fraiseuse 4 axes peut avoir la combinaison d'axes XYZA ou XYZB. Un tour classique aura les axes XZ. Une machine de découpe à fil chaud aura les axes XYUZ.<sup>12</sup>

##### 7.1.1.1 Axes linéaires primaires

Les axes X, Y et Z produisent des mouvements linéaires dans trois directions, mutuellement orthogonales.

---

<sup>1</sup>Si le mouvement des composants mécaniques n'est pas indépendant, comme sur une machine hexapode, le langage RS274/NGC et les fonctions standards seront quand même utilisables, tant que le contrôle de bas niveau sait comment contrôler les mécanismes actuels pour produire le mouvement relatif de l'outil et de la pièce qui auraient été produits par des axes indépendants. C'est appelé, la cinématique.

<sup>2</sup>Avec EMC, le cas de la machine à portique XYYZ avec deux moteurs pour un axe est mieux traité par la cinématique que par un axe linéaire supplémentaire.

### **7.1.1.2 Axes linéaires secondaires**

Les axes U, V et W produisent des mouvements linéaires dans trois directions mutuellement orthogonales. Habituellement, X et U sont parallèles, Y et V sont parallèles et Z et W sont parallèles.

### **7.1.1.3 Axes rotatifs**

Les axes A, B et C produisent des mouvements angulaires (rotations). Habituellement, l'axe de rotation de A est parallèle à X, l'axe de rotation de B est parallèle à Y et l'axe de rotation de C est parallèle à Z.

## **7.1.2 Broche**

Un centre d'usinage est équipé d'une broche qui maintient un outil coupant, un palpeur ou d'autres outils. La broche peut tourner dans les deux sens. Elle peut être conçue pour tourner à vitesse constante mais réglable. Excepté sur les machines dont la broche est montée sur un axe rotatif, l'axe de la broche est maintenu parallèle à l'axe Z et il est coïncident avec l'axe Z quand X et Y sont à zéro. La broche peut être stoppée sur une position fixée ou non.

## **7.1.3 Arrosages**

Un centre d'usinage peut être équipé d'un composant fournissant l'arrosage fluide ou en brouillard.

## **7.1.4 Chargeur de pièces**

Un centre d'usinage peut être équipé d'un système de chargement des pièces. Le système se compose de deux porte-pièces sur lesquels sont fixés les bruts des pièces à usiner. Un seul porte-pièce à la fois est en position d'usinage.

## **7.1.5 Carrousel d'outils**

Un centre d'usinage peut être équipé d'un carrousel d'outils dans lequel sont rangés les outils déjà montés dans leurs porte-outils.

## **7.1.6 Changeur d'outil**

Un centre d'usinage peut être équipé d'un mécanisme de changement d'outils (fixés dans les porte-outils) entre le carrousel d'outils et la broche.

## **7.1.7 Visu des messages**

Un centre d'usinage est équipé d'une visualisation capable d'afficher les messages pour l'opérateur.

## **7.1.8 Correcteurs de vitesse d'avance et de broche**

Un centre d'usinage est équipé de boutons de réglage de la vitesse d'avance et de la vitesse de rotation de la broche, ils laissent l'opérateur corriger les vitesses nécessaires pour la broche et l'avance travail, il peut ainsi augmenter ou réduire les vitesses programmées.

### 7.1.9 Bouton d'effacement de block

Un centre d'usinage peut être équipé d'un bouton d'effacement de block. Voir la section 7.3.2.

### 7.1.10 Bouton d'arrêt optionnel du programme

Un centre d'usinage peut être équipé d'un bouton d'arrêt du programme. Voir la section 7.3.3.

## 7.2 Composants de contrôle et de données

### 7.2.1 Axes linéaires

Les axes X, Y et Z forment un système de coordonnées orthogonales standard. La position d'un axe s'exprime en utilisant ses coordonnées.

Les axes U, V et W forment également un système de coordonnées standard. X et U sont parallèles, Y et V sont parallèles enfin Z et W sont parallèles.

### 7.2.2 Axes rotatifs

Les axes rotatifs se mesurent en degrés. Leur sens de rotation positif est le sens anti-horaire quand l'observateur est placé face à l'axe.<sup>3</sup>

### 7.2.3 Point contrôlé

Le point contrôlé est le point dont la position et la vitesse de déplacement sont contrôlés. Quand la compensation de longueur d'outil est zéro (valeur par défaut), c'est un point situé sur l'axe de la broche et proche de la fin de celle-ci. Cette position peut être déplacée le long de l'axe de la broche en spécifiant une compensation de longueur d'outil. Cette compensation correspond généralement à la longueur de l'outil coupant courant. Ainsi, le point contrôlé est à la pointe de l'outil. Sur un tour, les correcteurs d'outil peuvent être spécifiés pour les axes X et Z, le point contrôlé est à la pointe de l'outil ou (correction du rayon de bec) légèrement en retrait du point d'intersection des droites perpendiculaires formées par l'axe des points de tangence à la pièce, de face et sur le côté de l'outil.

### 7.2.4 Mouvement linéaire coordonné

Pour mener un outil sur une trajectoire spécifiée, un centre d'usinage doit coordonner les mouvements de plusieurs axes. Nous utilisons le terme "mouvement linéaire coordonné" pour décrire une situation dans laquelle, nominalement, chacun des axes se déplace à vitesse constante et tous les axes se déplacent de leur point de départ à leur point d'arrivée en même temps. Si deux des axes X, Y, Z (ou les trois) se déplacent, ceci produit un mouvement en ligne droite, d'où le mot "linéaire" dans le terme. Dans les véritables mouvements, ce n'est souvent pas possible de maintenir la vitesse constante à cause des accélérations et décélérations nécessaires en début et fin de mouvement. C'est faisable, cependant, de contrôler les axes ainsi, chaque axe doit en permanence faire la même fraction du mouvement requis que les autres axes. Ceci déplace l'outil le long du même parcours et nous appelons aussi ce genre de mouvement, mouvement linéaire coordonné.

Un mouvement linéaire coordonné peut être exécuté soit en vitesse travail, soit en vitesse rapide, ou il peut être synchronisé à la rotation de la broche. Si les limites physique de l'axe rendent le déplacement impossible, tous les axes seront ralentis pour maintenir le chemin prévu.

<sup>3</sup>Si les parallélismes sont particuliers, le constructeur du système devra indiquer à quels sens de rotation correspondent horaire et anti-horaire.

### 7.2.5 Vitesse

La vitesse à laquelle le point contrôlé se déplace est ajustable par l'opérateur. Sauf cas particulier, (vitesse inverse du temps, vitesse par tour, voir la section [12.28](#)) , dans l'interpréteur, l'interprétation des vitesses est la suivante :

1. Si le déplacement concerne un des axes XYZ, F est en unités machine par minute dans le système Cartésien XYZ et les mouvements des autres axes (UVWABC) sont également dans un même mode de coordonnées.
2. Autrement, si le déplacement concerne un des axes UVW, F est en unités machine par minute dans le système Cartésien UVW, tous les autres axes (ABC) se déplacent dans un même mode de coordonnées.
3. Autrement, le mouvement est purement rotatif et le mot F est en unités de rotation dans le système pseudo-Cartésien ABC.

### 7.2.6 Arrosage

Arrosage fluide ou brouillard (gouttelettes) peuvent être activés séparément. Le langage RS274/NGC les arrête ensemble (voir la section [13.4](#)).

### 7.2.7 Temporisation

Une temporisation peut être commandée (ex : pour immobiliser tous les axes) pendant une durée spécifique. La broche n'est pas arrêtée pendant une temporisation! Sans s'occuper du mode de contrôle de trajectoire (voir la section [7.2.15](#)) la machine s'arrêtera exactement à la fin du dernier mouvement avant la temporisation.

### 7.2.8 Unités

Les unités utilisées pour les distances le long des axes X, Y et Z peuvent être les pouces ou les millimètres. La vitesse de rotation de la broche est en tours par minute. Les positions des axes rotatifs sont exprimées en degrés. Les vitesses d'avance sont exprimées en unités machine par minute ou en degrés par minute ou en unités de longueur par tour de broche, comme décrit dans la section [7.2.5](#).

### 7.2.9 Position courante

Le point contrôlé est toujours à un emplacement appelé la "position courante," et le contrôleur sait toujours où est cette position. Les valeurs représentant la position courante doivent être ajustées en l'absence de tout mouvement des axes si un de ces événements a lieu :

1. Les unités de longueur ont changé.
2. La compensation de longueur d'outil a changé.
3. Le décalage d'origine a changé.

### 7.2.10 Choix du plan de travail

Il y a toujours un plan sélectionné, qui doit être le plan XY, le plan YZ, ou le plan XZ de la machine. L'axe Z est, bien sûr, perpendiculaire au plan XY, l'axe X perpendiculaire au plan YZ et l'axe Y perpendiculaire au plan XZ.

### **7.2.11 Carrousel d'outils**

Aucun ou un outil est assigné à chaque emplacement dans le carrousel.

### **7.2.12 Changeur d'outil**

Un centre d'usinage peut commander un changeur d'outils.

### **7.2.13 Chargeur de pièce**

Les deux porte-pièces peuvent être intervertis par commande.

### **7.2.14 Boutons des correcteurs de vitesses**

Les boutons des correcteurs de vitesses peuvent être activés (ils fonctionnent normalement) ou rendus inopérents (Ils n'ont plus aucun effet). Le langage RS274/NGC dispose d'une commande qui active tous les boutons et une autre qui les désactive (voir la section 13.5). Voir la section 7.3.1 pour d'autres détails.

### **7.2.15 Modes de contrôle de trajectoire**

La machine peut être placée dans un de ces trois modes de contrôle de trajectoire : (1) mode arrêt exact, (2) mode trajectoire exacte ou (3) mode trajectoire continue avec tolérance optionnelle. En mode arrêt exact, le mobile s'arrête brièvement à la fin de chaque mouvement programmé. En mode trajectoire exacte, le mobile suit la trajectoire programmée aussi précisément que possible, ralentissant ou s'arrêtant si nécessaire aux angles vifs du parcours. En mode trajectoire continue, les angles vifs du parcours peuvent être légèrement arrondis pour que la vitesse soit maintenue (sans dépasser la tolérance, si elle est spécifiée). Voir la section 12.20.

## **7.3 Interaction de l'interpréteur avec les boutons**

L'interpréteur interagit avec plusieurs boutons de commande. Cette section décrit ces interactions plus en détail. En aucun cas l'interpréteur ne connaît ce que sont les réglages de ces boutons.

### **7.3.1 Boutons de correction de vitesses**

L'interpréteur de commande RS274/NGC autorise (M48) ou interdit (M49) l'action des boutons d'ajustement des vitesses. Pour certains mouvements, tels que la sortie de filet à la fin d'un cycle de filetage, les boutons sont neutralisés automatiquement.

EMC2 réagit aux réglages de ces boutons seulement quand ils sont autorisés.

### **7.3.2 Bouton d'effacement de block Switch**

Si le bouton "Effacement de block" est actif, les lignes de code RS274/NGC commençant par le caractère barre de fraction (caractère d'effacement de block) ne sont pas interprétées. Si le bouton est désactivé, ces mêmes lignes sont interprétées. Normalement le bouton d'effacement de block doit être positionné avant de lancer le programme NGC.

### 7.3.3 Bouton d'arrêt optionnel du programme

Si ce bouton est actif et qu'un code M1 est rencontré, le programme est mis en pause.

## 7.4 Fichier d'outils

Un fichier d'outils est requis par l'interpréteur. Le fichier indique dans quels emplacements du carrousel sont placés les outils, la longueur et le diamètre de chacun des outils.

Le fichier est composé d'un certain nombre de lignes d'en-tête, suivies par une ligne vide, suivie d'un nombre quelconque de lignes de données. Les lignes d'en-tête sont ignorées par l'interpréteur. Il est important qu'il y ait une ligne vide (sans espace ni tabulation), avant les données. La ligne d'en-tête montrée dans ce tableau 7.1, décrit les colonnes de données, il est donc proposé (mais pas obligatoire) que cette ligne soit toujours présente.

Chaque ligne de données du fichier contient les données d'un outil. La ligne peut contenir 4 ou 5 éléments ("format fraiseuse") ou 8 ou 9 éléments ("format tour").

Les unités utilisées pour la longueur et le diamètre sont en unités machine.

Les lignes n'ont pas à être dans un ordre particulier. Permuter l'ordre des lignes est sans effet, sauf si le même numéro d'emplacement est utilisé sur deux ou plusieurs lignes, ce qui ne devrait normalement pas être fait, dans ce cas, seules les données de la dernière de ces lignes seront utilisées.

Dans emc2, l'emplacement du fichier d'outils est spécifié dans le fichier ini. Voir la section [EMCIO] du Manuel de l'intégrateur pour plus de détails.

Un fichier d'outils peut être un mélange de lignes au "format fraiseuse" et au "format tour", bien que généralement les lignes du style "format tour" soient seulement requises pour les outils de tour.

### 7.4.1 Fichier d'outils au format fraiseuse

Le "format fraiseuse" d'un fichier d'outils est visible dans le tableau 7.1.

TAB. 7.1 – Exemple de fichier d'outils (format fraiseuse)

Slot	FMS	TLO	Diamètre	Commentaire
1	1	2.0	1.0	
2	2	1.0	0.2	
5	5	1.5	0.25	coupe en bout
10	10	2.4	-0.3	à tester

Chaque ligne comporte 5 éléments. Les quatre premiers éléments sont obligatoires. Le commentaire, placé en cinquième est facultatif. La lecture est rendue plus facile si les éléments sont disposés en colonnes, comme dans le tableau ci-dessus, mais la seule exigence de forme, c'est qu'il y ait au moins un espace ou une tabulation après chacun des trois premiers éléments d'une ligne, et une tabulation ou un saut de ligne à la fin des cinq éléments. La signification des colonnes et le type de données à mettre dans chacune sont les suivants :

La colonne "Slot" contient un entier non signé représentant le numéro d'emplacement dans le carrousel d'outils (numéro de slot) dans lequel cet outil est placé. Les éléments de cette colonne doivent être tous différents.

La colonne "FMS" contient un entier non signé représentant le numéro de code de l'outil. L'utilisateur peut y placer n'importe quel code pour tous les outils, tant que les codes sont des entiers non signés. C'est en général le même que celui de l'emplacement.

La colonne "TLO" contient un nombre réel représentant l'offset de longueur d'outil. Cette valeur sera utilisée si la compensation de longueur d'outil est activée et que ce numéro d'emplacement est choisi. C'est normalement un nombre réel positif, mais il peut être mis à zéro ou toute autre valeur si il n'a pas à être utilisé.

La colonne "Diamètre" contient un nombre réel. Cette valeur est utilisée seulement si la compensation de rayon d'outil est activée et que ce numéro d'emplacement est choisi. Si la trajectoire programmée avec la compensation est le bord du matériau à usiner, ce devrait être un nombre réel positif représentant le diamètre mesuré de l'outil. Si la trajectoire programmée pendant la compensation est la trajectoire d'un outil dont le diamètre est nominal, ce nombre devrait être petit (positif, négatif ou nul) et représenter la différence entre le diamètre mesuré de l'outil et le diamètre nominal. Si la compensation de rayon d'outil n'est pas utilisée avec un outil, le contenu de cette colonne est sans effet.

La colonne "Commentaire" est optionnelle, elle peut être utilisée pour décrire l'outil. Tout type de description convient. Cette colonne améliore la lisibilité du fichier.

### 7.4.2 Fichier d'outils au format tour

Le "format tour" d'un fichier d'outils est visible dans le tableau 7.2.

TAB. 7.2 – Exemple de fichier d'outils (format tour)

Slot	FMS	ZOFFSET	XOFFSET	DIA	FRONTANGLE	BACKANGLE	ORIENTATION	Commentaire
1	1	0.0	0.0	0.1	95.0	155.0	1	
2	2	0.5	0.5	0.1	120	60	6	

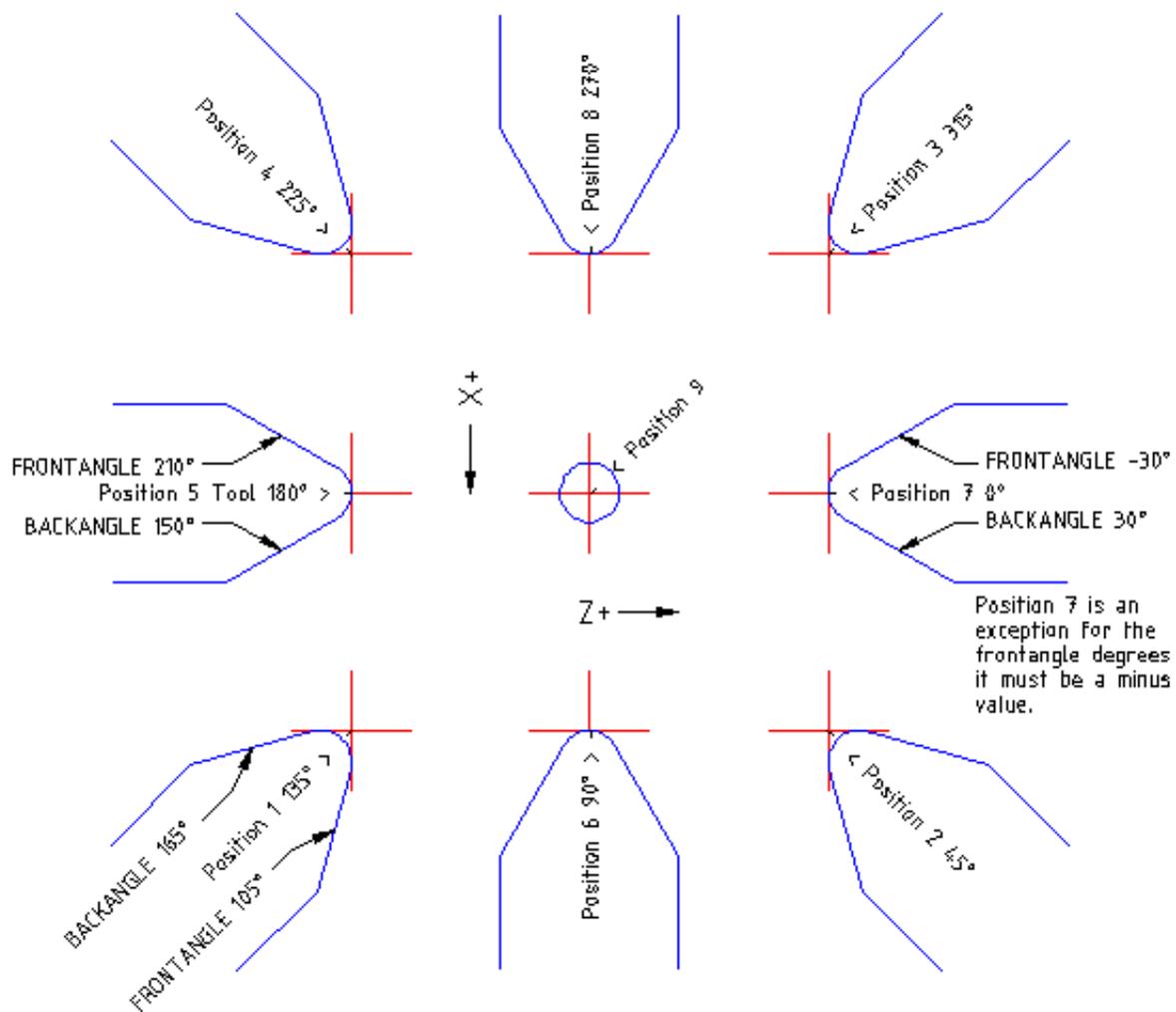
Les champs Slot, FMS, DIA et Commentaire sont les mêmes que dans un fichier d'outils de fraiseuse. La colonne ZOFFSET est la même que la colonne TLO d'un fichier d'outils de fraiseuse; offset de longueur d'outil.

La colonne XOFFSET fournit l'offset pour la coordonnée X quand la compensation de longueur d'outil est active.

La colonne ORIENTATION donne l'orientation de l'outil de tour, comme illustré sur 7.1. La croix rouge correspond au point contrôlé. Voir 7.2.3.

Les colonnes FRONTANGLE et BACKANGLE sont utilisées par certaines interfaces utilisateur pour afficher une représentation de l'outil de tour.

FIG. 7.1 – Orientations des outils de tour



## 7.5 Paramètres

Dans le langage RS274/NGC, le centre d'usinage maintient un tableau de 5400 paramètres numériques. La plupart d'entre eux ont un usage spécifique. Le tableau de paramètres est persistant, même quand la machine est mise hors tension. EMC2 utilise un fichier de paramètres et assure sa persistance, il donne à l'interpréteur la responsabilité d'actualiser le fichier. L'interpréteur lit le fichier quand il démarre et l'écrit juste avant de s'arrêter.

Le format d'un fichier de paramètres est visible dans la table 7.4. Le fichier est composé d'un certain nombre de lignes d'en-tête, suivie par une ligne vide, suivie d'un nombre quelconque de lignes de données. Les lignes d'en-tête sont ignorées par l'interpréteur. Il est important qu'il y ait une ligne vide (sans espace ni tabulation), avant les données. La ligne d'en-tête montrée dans ce tableau 7.4, décrit les colonnes de données, il est donc proposé (mais pas obligatoire) que cette ligne soit toujours présente.

TAB. 7.3 – Paramètres utilisés par l'interpréteur RS274NGC

Numéros de paramètre	Signification
5061-5070	Résultats des mesures "G38.2"
5161-5169	"G28" Origines machine
5181-5189	"G30" Origines pièce
5211-5219	"G92" Décalages d'origines
5220	Numéro du système de coordonnées
5221-5229	Système de coordonnées 1
5241-5249	Système de coordonnées 2
5261-5269	Système de coordonnées 3
5281-5289	Système de coordonnées 4
5301-5309	Système de coordonnées 5
5321-5329	Système de coordonnées 6
5341-5349	Système de coordonnées 7
5361-5369	Système de coordonnées 8
5381-5389	Système de coordonnées 9
5399	Résultat de M66 - Surveillance entrée

L'interpréteur lit seulement les deux premières colonnes du tableau. Il ignore la troisième colonne, Commentaire.

Chaque ligne du fichier contient le numéro d'index d'un paramètre dans la première colonne et la valeur attribuée à ce paramètre, dans la deuxième colonne. La valeur est représentée par un nombre flottant en double précision à l'intérieur de l'interpréteur, mais le point décimal n'est pas exigé dans le fichier. Le format des paramètres visible dans le tableau 7.4 est obligatoire et doit être utilisé pour tous les fichiers de paramètres, à l'exception des paramètres représentant une valeur sur un axe rotatif inutilisé, qui peuvent être omis. Une erreur sera signalée si un paramètre requis est absent. Un fichier de paramètres peut inclure tout autre paramètre, tant que son numéro est compris dans une fourchette de 1 à 5400. Les numéros de paramètre doivent être disposés dans l'ordre croissant. Sinon, une erreur sera signalée. Le fichier original est copié comme fichier de sauvegarde lorsque le nouveau fichier est écrit. Les commentaires ne sont pas conservés lorsque le fichier est écrit.

TAB. 7.4 – Format d'un fichier de paramètres

Numéro d'index	Valeur	Commentaire
5161	0.0	G28 pom X
5162	0.0	G28 pom Y

## 7.6 Systèmes de coordonnées

Dans le langage RS274/NGC, un centre d'usinage dispose d'un système de coordonnées machine (absolues) et de 9 systèmes de coordonnées programme (relatives).

Vous pouvez définir le décalage d'origine de neuf systèmes de coordonnées programme en utilisant G10 L2 Pn (n étant le numéro du système de coordonnées), avec des valeurs pour les axes en coordonnées absolues. Voir la section 12.9.

Vous pouvez choisir un des neuf systèmes en utilisant G54, G55, G56, G57, G58, G59, G59.1, G59.2, ou G59.3 (Voir la section 12.19). Il n'est pas possible de choisir directement le système de coordonnées machine.

Vous pouvez décaler l'origine du système de coordonnées actuel en utilisant G92 ou G92.3. Ce décalage s'appliquera alors à l'ensemble des neuf systèmes de coordonnées programme. Ce décalage peut être annulé avec G92.1 ou G92.2. Voir la section 12.27.

Vous pouvez faire des mouvements dans le système de coordonnées machine absolues en utilisant G53 avec G0 ou G1. Voir la section [12.18](#).

Les données des systèmes de coordonnées sont enregistrées dans les paramètres.

Lors de l'initialisation, le système de coordonnées choisi est celui spécifié par le paramètre 5220. Une valeur de 1 signifie le premier système de coordonnées (celui qu'active G54), une valeur de 2 signifie le deuxième système de coordonnées (celui qu'active G55), et ainsi de suite. Il y a erreur si la valeur du paramètre 5220 n'est pas un nombre entier compris entre 1 et 9.

# Chapitre 8

## Vue d'ensemble du langage

Le langage RS274/NGC est basé sur des lignes de code. Chaque ligne (également appelée un "block") peut inclure des commandes pour faire produire diverses actions au centre d'usinage. Plusieurs lignes de code peuvent être regroupées dans un fichier pour créer un programme.

Une ligne de code typique commence par un numéro de ligne optionnel suivi par un ou plusieurs "mots". Un mot commence par une lettre suivie d'un nombre (ou quelque chose qui permet d'évaluer un nombre). Un mot peut, soit donner une commande, soit fournir un argument à une commande. Par exemple, "G1 X3" est une ligne de code valide avec deux mots. "G1" est une commande qui signifie "déplaces toi en ligne droite à la vitesse programmée" et "X3" fournit la valeur d'argument (la valeur de X doit être 3 à la fin du mouvement). La plupart des commandes RS274/NGC commencent avec G ou M (G pour Général et M pour Miscellaneous (auxiliaire)). Les termes pour ces commandes sont "G codes" et "M codes."

Le langage RS274/NGC n'a pas d'indicateur de début et de fin de programme. L'interpréteur cependant traite les fichiers. Un programme simple peut être en un seul fichier, mais il peut aussi être partagé sur plusieurs fichiers. Un fichier peut être délimité par le signe pourcent de la manière suivante. La première ligne non vide d'un fichier peut contenir un signe "%" seul, éventuellement encadré d'espaces blancs, ensuite, à la fin du fichier on doit trouver une ligne similaire. Délimiter un fichier avec des % est facultatif si le fichier comporte un M2 ou un M30, mais est requis sinon. Une erreur sera signalée si un fichier a une ligne pourcent au début, mais pas à la fin. Le contenu utile d'un fichier délimité par pourcent s'arrête après la seconde ligne pourcent. Tout le reste est ignoré.

Le langage RS274/NGC prévoit les deux commandes (M2 ou M30) pour finir un programme. Le programme peut se terminer avant la fin du fichier. Les lignes placées après la fin d'un programme ne seront pas exécutées. L'interpréteur ne les lit même pas.

### 8.1 Format d'une ligne

Une ligne de code permise par la norme RS274/NGC est construite de la façon suivante, dans l'ordre avec la restriction à un maximum de 256 caractères sur la même ligne.

1. Un block optionnel contenant le caractère d'effacement de ligne, barre de fraction "/".
2. Un numéro de ligne optionnel.
3. N'importe quel nombre de mots, valeurs de paramètres et commentaires.
4. Un caractère de fin de ligne (retour chariot ou saut de ligne ou les deux).

Toute entrée non explicitement permise est illégale, elle provoquera un message d'erreur de l'interpréteur.

Les espaces sont permis ainsi que les tabulations dans une ligne de code dont ils ne changent pas la signification, excepté dans les commentaires. Ceci peut donner d'étranges lignes, mais elles sont autorisées. La ligne "g0x +0. 12 34y 7" est équivalente à "g0 x+0.1234 y7", par exemple.

Les lignes vides sont permises, elles seront ignorées.

La casse des caractères est ignorée, excepté dans les commentaires. Toutes les lettres en dehors des commentaires peuvent être, indifféremment des majuscules ou des minuscules sans changer la signification de la ligne.

## 8.2 Numéro de ligne

Un numéro de ligne commence par la lettre N suivie d'un nombre entier non signé compris entre 0 et 99999 écrit avec moins de six caractères (par exemple : 000009 est interdit). Les numéros de ligne peuvent se suivre ou être dans le désordre, bien qu'une pratique normale évite ce genre d'usage. Les numéros de ligne peuvent être sautés, c'est une pratique normale. L'utilisation d'un numéro de ligne n'est pas obligatoire, mais si il est utilisé, il doit être à sa place.

## 8.3 Les mots

Un mot est une lettre, autre que N, suivie d'un nombre réel.

Les mots peuvent commencer avec l'une ou l'autre des lettres indiquées dans le tableau 8.1. Le tableau inclus N pour être complet, même si, comme défini précédemment, les numéros de lignes ne sont pas des mots. Plusieurs lettres (I, J, K, L, P, R) peuvent avoir différentes significations dans des contextes différents. Les lettres qui se réfèrent aux noms d'axes ne sont pas valides sur une machine n'ayant pas les axes correspondants.

### 8.3.1 Nombres

Les règles suivantes sont employées pour des nombres (explicites). Dans ces règles un chiffre est un caractère simple entre 0 et 9.

- Un nombre commence par : (1) un signe plus ou un signe moins optionnel, suivi par (2) de zéro à plusieurs chiffres, peut être suivis par, (3) un point décimal, suivi par (4) de zéro à plusieurs chiffres, il doit au moins y avoir un chiffre.
- Il existe deux types de nombres : les entiers et les décimaux. Un entier n'a pas de point décimal ; un décimal en a un.
- Les nombres peuvent avoir n'importe quel nombre de chiffres, sous réserve de la limitation de longueur d'une ligne. Seulement environ dix-sept chiffres significatifs seront retenus, c'est toutefois suffisant pour toutes les applications connues.
- Un nombre non nul sans autre signe que le premier caractère est considéré positif.

Les zéros non significatifs, ne sont pas nécessaires.

Si un nombre utilisé dans le langage RS274/NGC est proche d'une valeur entière à moins de quatre décimales, il est considéré comme entier, par exemple 0.9999.

### 8.3.2 Paramètres numérotés

Un paramètre numéroté commence par le caractère # suivi par un entier compris entre 1 et 5399. Le paramètre est référencé par cet entier, sa valeur est la valeur stockée dans le paramètre.

Une valeur est stockée dans un paramètre avec l'opérateur = par exemple "#3 = 15" signifie que la valeur 15 est stockée dans le paramètre numéro 3.

TAB. 8.1 – Les mots et leur signification

Lettre	Signification
A	Axe A de la machine
B	Axe B de la machine
C	Axe C de la machine
D	Valeur de la compensation de rayon d'outil
F	Vitesse d'avance travail
G	Fonction Générale (voir la table 5)
H	Index d'offset de longueur d'outil
I	Décalage en X pour les arcs et dans les cycles préprogrammés G87
J	Décalage en Y pour les arcs et dans les cycles préprogrammés G87
K	Décalage en Z pour les arcs et dans les cycles préprogrammés G87 Distance de déplacement par tour de broche avec G33
M	Fonction auxiliaire (voir la table 7)
N	Numéro de ligne
P	Temporisation utilisée dans les cycles préprogrammés et avec G4. Mot clé utilisé avec G10.
Q	Incrément Delta en Z dans un cycle préprogrammé G73, G83
R	Rayon d'arc ou plan de retrait dans un cycle préprogrammé
S	Vitesse de rotation de la broche
T	Numéro d'outil
U	Axe U de la machine
V	Axe V de la machine
W	Axe W de la machine
X	Axe X de la machine
Y	Axe Y de la machine
Z	Axe Z de la machine

Le caractère # a une précedence supérieure à celle des autres opérations, ainsi par exemple, “#1+2” signifie la valeur trouvée en ajoutant 2 à la valeur contenue dans le paramètre 1 et non la valeur trouvée dans le paramètre 3. Bien sûr, #[1+2] signifie la valeur trouvée dans le paramètre 3. Le caractère # peut être répété, par exemple ##2 signifie le paramètre dont le numéro est égal à la valeur entière trouvée dans le paramètre 2.

### 8.3.3 Paramètres nommés

Les paramètres nommés fonctionnent comme les paramètres numérotés mais sont plus faciles à lire. Les paramètres nommés sont convertis en minuscules, les espaces et tabulations sont supprimés. Les paramètres nommés doivent être encadrés des signes < et >.

#<Un paramètre nommé> est un paramètre nommé local. Par défaut, un paramètre nommé est local à l'étendue dans laquelle il est assigné. L'accès à un paramètre local, en dehors de son sous-programme est impossible, de sorte que deux sous-programmes puissent utiliser le même nom de paramètre sans craindre qu'un des deux n'écrase la valeur de l'autre.

#<\_un paramètre global> est un paramètre nommé global. Ils sont accessibles depuis des sous-programmes appelés et peuvent placer des valeurs dans tous les sous-programmes accessibles à l'appelant. En ce qui concerne la portée, ils agissent comme des paramètres numérotés. Ils ne sont pas enregistrés dans des fichiers.

Exemples :

– Déclaration d'une variable nommée globale

```
#<_trois dents_dia> = 10.00
```

- Référence à la variable globale précédemment déclarée

```
#<_troisdents_rayon> = [#<_troisdents_dia>/2.0]
```

- Mélange de paramètres nommés et de valeurs littérales

```
o100 call [0.0] [0.0] [#<_interieur_decoupe>-#<_troisdents_dia>] [#<_Zprofondeur>]
[#<_vitesse>]
```

Notes :

Les paramètres globaux `_a`, `_b`, `_c`, ... `_z` sont réservés pour une utilisation spéciale. Dans le futur, ils pourront fournir l'accès aux derniers `Aword`, `Bword`, `Cword`, ... `Zword` etc.

### 8.3.4 Expressions

Une expression est un groupe de caractères commençant avec le crochet gauche `[` et se terminant avec le crochet droit `]`. Entre les crochets, on trouve des nombres, des valeurs de paramètre, des opérations mathématiques et d'autres expressions. Une expression est évaluée pour produire un nombre. Les expressions sur une ligne sont évaluées quand la ligne est lue et avant que quoi que ce soit ne soit exécuté sur cette ligne. Un exemple d'expression : `[1 + acos[0] - [#3 ** [4.0/2]]]`.

### 8.3.5 Opérateurs binaires

Les opérateurs binaires ne se rencontrent que dans les expressions. Il y a quatre opérateurs mathématiques de base : addition (+), soustraction (-), multiplication (\*) et division (/). Il y a trois opérateurs logiques : ou (OR), ou exclusif (XOR) et logique (AND). Le huitième opérateur est le modulo (MOD). Le neuvième opérateur est l'élevation à la puissance (\*\*) qui élève le nombre situé à sa gauche à la puissance du nombre situé à sa droite. Les opérateurs de relation sont : égalité (EQ), non égalité (NE), strictement supérieur (GT), supérieur ou égal (GE), strictement inférieur (LT) et inférieur ou égal (LE).

Les opérations binaires sont divisées en plusieurs groupes selon leur précedence. (voir la table 8.2) Si dans une opération se trouvent différents groupes de précedence (par exemple dans l'expression `[2.0 / 3 * 1.5 - 5.5 / 11.0]`), les opérations du groupe supérieur seront effectuées avant celles des groupes inférieurs. Si une expression contient plusieurs opérations du même groupe (comme les premiers / et \* dans l'exemple), l'opération de gauche est effectuée en premier. Notre exemple est équivalent à : `[[[2.0 / 3] * 1.5] - [5.5 / 11.0]]`, qui est équivalent à `[1.0 - 0.5]`, le résultat est : `0.5`.

Les opérations logiques et le modulo sont exécutés sur des nombres réels et non pas seulement sur des entiers. Le zéro est équivalent à un état logique faux (FALSE), tout nombre différent de zéro est équivalent à un état logique vrai (TRUE).

TAB. 8.2 – Précedence des opérateurs

Opérateurs	Précedence
**	<i>haute</i>
* / MOD	
+ -	
EQ NE GT GE LT LE	
AND OR XOR	<i>basse</i>

### 8.3.6 Fonctions

Une fonction commence par son nom, ex : "ATAN" suivi par une expression divisée par une autre expression (par exemple "ATAN[2] / [1+3]") ou tout autre nom de fonction suivi par une expression

(par exemple "SIN[90]"). Les fonctions disponibles sont visibles dans le tableau 8.3. Les arguments pour les opérations unaires sur des angles (COS, SIN et TAN) sont en degrés. Les valeurs retournées par les opérations sur les angles (ACOS, ASIN et ATAN) sont également en degrés.

L'opérateur FIX arrondi vers la gauche (moins positif ou plus négatif), par exemple : FIX[2.8] = 2 et FIX[-2.8] = -3 . L'opérateur FUP arrondi vers la droite (plus positif ou moins négatif), par exemple : FUP[2.8] = 3 et FUP[-2.8] = -2 .

TAB. 8.3 – Fonctions

Nom de fonction	Fonction
ATAN[Y]/[X]	Tangente quatre quadrants
ABS[arg]	Valeur absolue
ACOS[arg]	Arc cosinus
ASIN[arg]	Arc sinus
COS[arg]	Cosinus
EXP[arg]	Exposant
FIX[arg]	Arrondi à l'entier immédiatement inférieur
FUP[arg]	Arrondi à l'entier immédiatement supérieur
ROUND[arg]	Arrondi à l'entier le plus proche
LN[arg]	Logarithme Néperien
SIN[arg]	Sinus
SQRT[arg]	Racine carrée
TAN[arg]	Tangente

## 8.4 Commentaires

Un ensemble de caractères et espaces blancs entre parenthèses est un commentaire. Une parenthèse ouvrante débute toujours un commentaire. Le commentaire se termine à la première parenthèse fermante trouvée. Si une parenthèse ouvrante est trouvée sur une ligne, une parenthèse fermante doit être également rencontrée avant la fin de la ligne. Les commentaires ne peuvent pas être imbriqués, une erreur sera signalée si une parenthèse ouvrante est rencontrée après le début d'un commentaire et avant la fin d'un commentaire. Voici un exemple de ligne de commentaire : "G80 M5 (arrêt du mouvement)". Les commentaires sont seulement informatifs, ils n'ont aucune influence sur la machine.

### 8.4.1 Messages

Un commentaire contient un message si "MSG" apparaît après la parenthèse ouvrante, et avant tout autre caractère. Les variantes de "MSG" qui incluent un espace blanc et des minuscules sont autorisées. Le reste du texte avant la parenthèse fermante est considéré comme le message. Les messages sont affichés sur la visu de l'interface utilisateur. Les commentaires ne contenant pas de message ne sont pas affichés.

### 8.4.2 Enregistrement des mesures

Un commentaire peut aussi être utilisé pour spécifier le fichier de log des résultats des mesures faites au palpeur avec G38.x . Voir la section 12.15.

### 8.4.3 Log général

#### 8.4.3.1 (LOGOPEN,filename)

Ouvre le fichier de log "filename". Si le fichier existe déjà, il sera tronqué.

#### 8.4.3.2 (LOGCLOSE)

Si le fichier est ouvert, il sera fermé.

#### 8.4.3.3 (LOG,...)

Le message "..." est étendu comme décrit plus loin, il est écrit dans le fichier de log si il est ouvert.

### 8.4.4 Messages de déboguage

Les commentaires comme : (debug, reste du commentaire) sont traités de la même façon que ceux avec (msg, reste du commentaire) avec l'ajout de possibilités spéciales pour les paramètres.

Les commentaires comme : (print, reste du commentaire) vont directement sur la sortie stderr avec des possibilités spéciales pour les paramètres.

### 8.4.5 Paramètres dans les commentaires

Dans les commentaires avec DEBUG, PRINT et LOG, les valeurs des paramètres dans le message sont étendues.

Par exemple : pour afficher une variable nommée globale sur la sortie stderr (la fenêtre de la console par défaut) ajouter une ligne au g-code comme :

```
(print,diamètre fraise 3 dents = #<troisdents_dia>)
```

À l'intérieur de ces types de commentaires, les séquences comme #123 sont remplacées par la valeur du paramètre 123. Les séquences comme #<paramètre nommé> sont remplacées par la valeur du paramètre nommé. Rappelez vous que les espaces dans les noms des paramètres nommés sont supprimés, #<parametre nomme> est équivalent à #<parametrenomme>.

## 8.5 Répétitions d'items

Une ligne peut contenir autant de mots G que voulu, mais seulement deux mots G du même groupe modal peuvent apparaître sur la même ligne. (voir la section 8.8)

Une ligne peut avoir de zéro à quatre mots M. Mais pas deux mots M du même groupe modal.

Pour toutes les autres lettres légales, un seul mot commençant par cette lettre peut se trouver sur la même ligne.

Si plusieurs valeurs de paramètre se répètent sur la même ligne, par exemple : "#3=15 #3=6", seule la dernière valeur prendra effet. Il est absurde, mais pas illégal, de fixer le même paramètre deux fois sur la même ligne.

Si plus d'un commentaire apparaît sur la même ligne, seul le dernier sera utilisé, chacun des autres sera lu et son format vérifié, mais il sera ignoré. Placer plusieurs commentaires sur la même ligne est très rare.

## 8.6 Ordre des items

Les trois types d'item dont la commande peut varier sur une ligne (comme indiqué au début de cette section) sont les mots, les paramètres et les commentaires. Imaginez que ces trois types d'éléments sont divisés en trois groupes selon leur type.

Dans le premier groupe les mots, peuvent être arrangés dans n'importe quel ordre sans changer la signification de la ligne.

Dans le second groupe les valeurs de paramètre, quelque soit leur arrangement, il n'y aura pas de changement dans la signification de la ligne sauf si le même paramètre est présent plusieurs fois. Dans ce cas, seule la valeur du dernier paramètre prendra effet. Par exemple, quand la ligne "#3=15 #3=6" aura été interprétée, la valeur du paramètre 3 vaudra 6. Si l'ordre est inversé, "#3=6 #3=15" après interprétation, la valeur du paramètre 3 vaudra 15.

Enfin dans le troisième groupe les commentaires, si plusieurs commentaires sont présents sur une ligne, seul le dernier commentaire sera utilisé.

Si chaque groupe est laissé, ou réordonné, dans l'ordre recommandé, la signification de la ligne ne changera pas, alors les trois groupes peuvent être entrecroisés n'importe comment sans changer la signification de la ligne. Par exemple, la ligne "g40 g1 #3=15 (foo) #4=-7.0" à cinq items est signifiera exactement la même chose dans les 120 ordres d'arrangement possibles des cinq items comme "#4=-7.0 g1 #3=15 g40 (foo)".

## 8.7 Commandes et modes machine

En RS274/NGC, de nombreuses commandes produisent, d'un mode à un autre, quelque chose de différent au niveau de la machine, le mode reste actif jusqu'à ce qu'une autre commande ne le révoque, implicitement ou explicitement. Ces commandes sont appelées "modales". Par exemple, si l'arrosage est mis en marche, il y reste jusqu'à ce qu'il soit explicitement arrêté. Les G-codes pour les mouvements sont également modaux. Si, par exemple, une commande G1 (déplacement linéaire) se trouve sur une ligne, elle peut être utilisée sur la ligne suivante avec seulement un mot d'axe, tant qu'une commande explicite est donnée sur la ligne suivante en utilisant des axes ou un arrêt de mouvement.

Les codes "non modaux" n'ont d'effet que sur la ligne ou ils se présentent. Par exemple, G4 (tempo) est non modale.

## 8.8 Groupes modaux

Les commandes modales sont arrangées par lots appelés "groupes modaux", à tout moment, un seul membre d'un groupe modal peut être actif. En général, un groupe modal contient des commandes pour lesquelles il est logiquement impossible que deux membres soient actifs simultanément, comme les unités en pouces et les unités en millimètres. Un centre d'usinage peut être dans plusieurs modes simultanément, si seulement un mode pour chaque groupe est actif. Les groupes modaux sont visibles dans le tableau 8.4.

Pour plusieurs groupes modaux, quand la machine est prête à accepter des commandes, un membre du groupe doit être en vigueur. Il y a des paramètres par défaut pour ces groupes modaux. Lorsque la machine est mise en marche ou ré-initialisées, les valeurs par défaut sont automatiquement actives.

Groupe 1, le premier groupe du tableau, est un groupe de G-codes pour les mouvements. À tout moment, un seul d'entre eux est actif. Il est appelé le mode de mouvement courant.

C'est une erreur que de mettre un G-code du groupe 1 et un G-code du groupe 0 sur la même ligne si les deux utilisent les mêmes axes. Si un mot d'axe utilisant un G-codé du groupe 1 est

TAB. 8.4 – Groupes modaux

Signification du groupe modal	Mots des membres
Mouvements ("Groupe 1")	G0 G1 G2 G3 G33 G38.x G73 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89
Choix du plan de travail	G17 G18 G19
Mode diamètre/rayon sur les tours	G7, G8
Mode de déplacements	G90, G91
Mode de vitesses	G93, G94
Unités machine	G20, G21
Correcteurs de rayon d'outil	G40, G41, G42, G41.1, G42.1
Correcteurs de longueur d'outil	G43, G43.1, G49
Options plan de retrait	G98, G99
Systèmes de coordonnées	G54, G55, G56, G57, G58, G59, G59.1, G59.2, G59.3
Types d'arrêt de programme	M0, M1, M2, M30, M60
Appel d'outil	M6
Types de rotation de la broche	M3, M4, M5
Arrosages	M7, M8, M9. Cas spéciaux : M7, M8 peuvent être actifs simultanément
Boutons de correction de vitesse	M48, M49
Contrôle de flux	O-
Codes non modaux ("Groupe 0")	G4, G10, G28, G28.1, G30, G30.1, G53 G92, G92.1, G92.2, G92.3 M100 à M199

implicitement actif sur la ligne (en ayant été activé sur une ancienne ligne) et qu'un G-code du groupe 0 utilisant des mots d'axes apparaît sur la même ligne, l'activité du G-code du groupe 1 est révoquée pour le reste de la ligne. Les mots d'axes utilisant des G-codes du groupe 0 sont G10, G28, G30 et G92.

C'est une erreur d'inclure des mots sans rapport sur une ligne avec le contrôle de flux O.

# Chapitre 9

## Ordre d'exécution

L'ordre d'exécution des éléments d'une ligne est essentiel à la sécurité et l'efficacité d'une machine. Les éléments sont exécutés dans l'ordre indiqué ci-dessous, si ils se trouvent sur la même ligne.

1. Commentaire (message inclu)
2. Positionnement du mode de vitesses (G93, G94).
3. Réglage de la vitesse travail (F).
4. Réglage de la vitesse de rotation de la broche (S).
5. Sélection de l'outil (T).
6. Appel d'outil (M6).
7. Marche/Arrêt broche (M3, M4, M5).
8. Marche/Arrêt arrosages (M7, M8, M9).
9. Activation/Inhibition correcteurs de vitesse (M48, M49).
10. Temporisation (G4).
11. Choix du plan de travail (G17, G18, G19).
12. Choix de l'unité de longueur (G20, G21).
13. Activation/Désactivation de la compensation de rayon d'outil (G40, G41, G42)
14. Activation/Désactivation de la compensation de longueur d'outil (G43, G49)
15. Sélection du système de coordonnées (G54, G55, G56, G57, G58, G59, G59.1, G59.2, G59.3).
16. Réglage du mode de trajectoire (G61, G61.1, G64)
17. Réglage du mode de déplacement (G90, G91).
18. Réglage du type de retrait (G98, G99).
19. Prise d'origine machine (G28, G30) ou établissement du système de coordonnées (G10) ou encore, réglage des décalages d'axes (G92, G92.1, G92.2, G94).
20. Effectuer les mouvements (G0 à G3, G33, G80 à G89), tels que modifiés (éventuellement) par G53.
21. Arrêt (M0, M1, M2, M30, M60).

# Chapitre 10

## G-Code : Bonnes pratiques

### 10.1 Utiliser un nombre de décimales approprié

Utilisez au plus 3 chiffres après la virgule pour l'usinage en millimètres et au plus 4 chiffres après la virgule pour l'usinage en pouces. En particulier, les contrôles de tolérance des arcs sont faits pour .001 et .0001 selon les unités actives.

### 10.2 Utiliser les espaces de façon cohérente

Le G-code est plus lisible quand au moins un espace apparaît avant les mots. S'il est permis d'insérer des espaces blancs au milieu des chiffres, il faut éviter de le faire.

### 10.3 Préférer le "format centre" pour les arcs

Les arcs en format centre (qui utilisent I- J- K- au lieu de R-) se comportent de façon plus précise que ceux en format rayon, particulièrement pour des angles proche de 180 et 360 degrés.

### 10.4 Placer les codes modaux importants au début des programmes

Lorsque l'exécution correcte de votre programme dépend de paramètres modaux, n'oubliez pas de les mettre au début du programme. Des modes incorrects peuvent provenir d'un programme précédent ou depuis des entrées manuelles.

Une bonne mesure préventive consiste à placer la ligne suivante au début de tous les programmes :

```
G17 G21 G40 G49 G54 G80 G90 G94
```

(plan XY, mode mm, annulation de la compensation de rayon, et de longueur, système de coordonnées numéro 1, arrêt des mouvements, déplacements absolus, mode vitesse/minute)

Peut-être que le code modal le plus important est le réglage des unités machine. Si les codes G20 ou G21, ne sont pas inclus, selon les machines l'échelle d'usinage sera différente. D'autres valeurs comme le plan de retrait des cycles préprogrammés peuvent être importantes.

## **10.5 Ne pas mettre trop de choses sur une ligne**

Ignorer le contenu de la section 9 et ne pas écrire de ligne de code qui laisse la moindre ambiguïté. De même, ne pas régler un paramètre et l'utiliser sur la même ligne, même si la sémantique est bien définie. (Exception : Mise à jour d'une variable à une nouvelle valeur, comme #1=[#1+#2])

## **10.6 Ne pas utiliser les numéros de ligne**

Les numéros de ligne n'apportent rien. Quand des numéros de ligne sont rapportés dans les messages d'erreur, ces numéros font référence aux numéros de lignes à l'intérieur du programme, pas aux valeurs des mots N.

## **10.7 Lorsque plusieurs systèmes de coordonnées sont déplacés, envisager le mode vitesse inverse du temps**

Parce que la signification d'un mot F en mètres par minute varie selon les axes à déplacer et parce que la quantité de matière enlevée ne dépend pas que de la vitesse travail, il peut être plus simple d'utiliser G93, vitesse inverse du temps pour atteindre l'enlèvement de matière souhaité.

# Chapitre 11

## Fichier d'outils et compensations

### 11.1 Fichier d'outils

Les longueurs et diamètres d'outils peuvent être lus dans un fichier d'outils (voir la section 7.4) ou provenir d'un mot spécifié pour activer la compensation d'outil.

### 11.2 Compensation d'outil

La compensation d'outil peut causer beaucoup de problèmes aux meilleurs programmeurs de code nc. Mais elle peut aussi être une aide puissante quand elle est utilisée pour aider un opérateur à obtenir une pièce à la cote. En réglant la longueur et le diamètre des outils dans une table d'outils, les décalages peuvent être utilisés pendant un cycle d'usinage qui tient compte des variations de taille de l'outil, ou pour des déviations mineures des trajectoires programmées. Et ces changements peuvent être faits sans que l'opérateur n'ait à changer quoi que ce soit dans le programme.

Tout au long de ce module, vous trouverez occasionnellement des références à des fonctions canoniques, là où il est nécessaire pour le lecteur de comprendre comment fonctionne une compensation d'outil dans une situation spécifique. Ces références ont pour but de donner au lecteur une idée de la séquences plutôt que d'exiger qu'il comprenne la façon dont les fonctions canoniques elles-mêmes fonctionnent dans EMC.

### 11.3 Compensation de longueur d'outil

Les compensations de longueur d'outil sont données comme des nombres positifs dans la table d'outils. Une compensation d'outil est programmée en utilisant G43 Hn, où n est le numéro d'index de l'outil souhaité dans la table d'outil. Il est prévu que toutes les entrées dans la table d'outils soit positives. La valeur de H est vérifiée, elle doit être un entier non négatif quand elle est lue. L'interpréteur se comporte comme suit :

1. Si G43 Hn est programmé, un appel à la fonction `USE_TOOL_LENGTH_OFFSET(longueur)` est fait (où longueur est l'écart de longueur, lu dans la table d'outils, de l'outil indexé n), `tool_length_offset` est repositionné dans le modèle de réglages de la machine et la valeur de `current_z` dans le modèle est ajustée. Noter que n n'a pas besoin d'être le même que le numéro de slot de l'outil actuellement dans la broche.

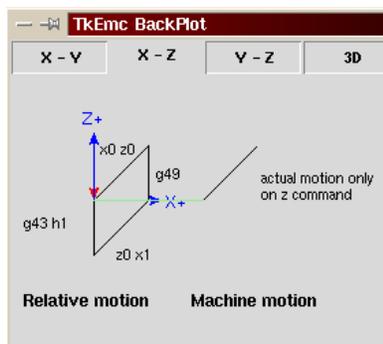
2. Si G49 est programmé, `USE_TOOL_LENGTH_OFFSET(0.0)` est appelée, `tool_length_offset` est remis à 0.0 dans le modèle de réglages de la machine et la valeur courante de `current_z` dans le modèle est ajustée. L'effet de la compensation de longueur d'outil est illustrée dans la capture ci-dessous.

Noter que la longueur de l'outil est soustraite de Z pour que le point contrôlé programmé corresponde à la pointe de l'outil. Il faut également noter que l'effet de la compensation de longueur est immédiat quand on voit la position de Z comme une coordonnée relative mais il est sans effet sur la position actuelle de la machine jusqu'à ce qu'un mouvement en Z soit programmé.

Programme de test de longueur d'outil.

Tool #1 is 1 inch long.

```
N01 G1 F15 X0 Y0 Z0
N02 G43 H1 Z0 X1
N03 G49 X0 Z0
N04 G0 X2
N05 G1 G43 H1 G4 P10 Z0 X3
N06 G49 X2 Z0
N07 G0 X0
```



Avec ce programme, dans la plupart des cas, la machine va appliquer le décalage sous forme d'une rampe pendant le mouvement en xyz suivant le mot G43.

## 11.4 Compensation de rayon d'outil

La compensation de rayon d'outil (également appelée compensation de diamètre d'outil) a été ajoutée aux spécifications RS-274D à la demande d'utilisateurs, car elle est extrêmement utile, mais son implémentation a été assez mal pensée. L'objectif de cette fonctionnalité est de permettre aux programmeurs de "virtualiser" la trajectoire de l'outil, de sorte que la machine puisse pendant toute l'exécution, déterminer le bon décalage à apporter à la position de l'outil pour respecter les cotes, en s'appuyant sur les diamètres d'outils existants. Si un outil est réaffûté, son diamètre sera légèrement plus petit que celui d'origine, il faudra également en tenir compte.

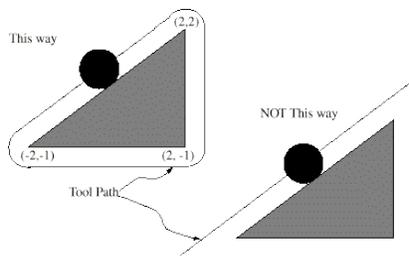
Le problème est pour donner à la machine la trajectoire exacte où l'outil doit usiner, sur le côté intérieur d'un parcours imaginaire, ou sur le côté extérieur. Comme ces trajectoires ne sont pas nécessairement fermées (même si elles peuvent l'être), il est quasiment impossible à la machine de connaître de quel côté du profil elle doit compenser l'outil. Il a été décidé qu'il n'y aurait que deux choix possibles, outil à "gauche" du profil à usiner et outil à "droite" du profil à usiner. Ce qui doit être interprété à gauche ou à droite du profil à usiner en suivant l'outil le long du profil.

### 11.4.1 Compensation de rayon d'outil, détails

Les possibilités de compensation de rayon d'outil de l'interpréteur, autorise le programmeur à spécifier si l'outil doit passer à gauche ou à droite du profil à usiner. Ce profil peut être un contour ouvert ou fermé, dans le plan XY composé de segments en arcs de cercles et en lignes droites. Le

contour peut être le pourtour d'une pièce brute ou, il peut être une trajectoire exacte suivie par un outil mesuré avec précision. La figure ci-dessous, montre deux exemples de trajectoires d'usinage d'une pièce triangulaire, utilisant la compensation de rayon d'outil.

Dans les deux exemples, le triangle gris représente le matériau restant après usinage et la ligne extérieure représente le parcours suivi par le centre de l'outil. Dans les deux cas le triangle gris est conservé. Le parcours de gauche (avec les coins arrondis) est le parcours généralement interprété. Dans la méthode de droite (celle marquée Not this way), l'outil ne reste pas en contact avec les angles vifs du triangle gris.



Des mouvements sur l'axe Z peuvent avoir lieu pendant que le contour est suivi dans le plan XY. Des portions du contour peuvent être franchies avec l'axe Z en retrait au dessus de la pièce pendant la poursuite du parcours et jusqu'au point où l'usinage doit reprendre, l'axe Z plongera de nouveau en position. Ces dégagements de zones non usinées peuvent être faits en vitesse travail (G1), en rapide (G0), en vitesse inverse du temps (G93) ou en avance en unités par minute (G94) toutes peuvent être utilisées avec la compensation de rayon d'outil. Sous G94, la vitesse sera appliquée à la pointe de l'outil coupant, non au contour programmé.

### Instructions de programmation

- Pour activer la compensation de rayon d'outil, programmer soit, G41 (pour maintenir l'outil à gauche du profil à usiner) soit, G42 (pour maintenir l'outil à droite du profil). Dans la figure 17.4.1, par exemple, si G41 était programmé, l'outil devrait tourner en sens horaire autour du triangle et dans le sens contraire si G42 était programmé.
- Pour désactiver la compensation de rayon d'outil, programmer G40.
- Si un G40, G41, ou G42 est programmé dans la même ligne qu'un mouvement d'axe, la compensation de rayon sera activée ou désactivée avant que le mouvement ne soit fait. Pour que le mouvement s'effectue en premier, il doit être programmé séparément et avant.

### La valeur de D

L'interpréteur actuel requiert une valeur D sur chaque ligne contenant un mot G41 ou G42. Le nombre D doit être un entier positif. Il représente le numéro de slot de l'outil, dont le rayon (la moitié du diamètre d'outil indiqué dans la table d'outils) sera compensé. Il peut aussi être égal à zéro, dans ce cas, la valeur du rayon sera aussi égale à zéro. Tous les slots de la table d'outils peuvent être sélectionnés de cette façon. Le nombre D n'est pas nécessairement le même que le numéro de slot de l'outil monté dans la broche.

### Table d'outils

La compensation de rayon d'outil utilise les données fournies par la table d'outils de la machine. Pour chaque slot d'outil dans le carrousel, la table d'outils contient le diamètre de l'outil rangé à cet emplacement (ou la différence entre le diamètre nominal de l'outil placé dans ce slot et son diamètre actuel). La table d'outils est indexée par les numéros de slot. Reportez vous à la page des "Fichiers d'outils" pour savoir comment remplir ces fichiers.

### Deux types de contour

L'interpréteur contrôle la compensation pour deux types de contour :

- 1) Le contour donné dans le code NC correspond au bord extérieur du matériau après usinage. Nous l'appellerons "contour sur le profil du matériau".
- 2) Le contour donné dans le code NC correspond à la trajectoire suivie par le centre d'un outil de rayon nominal. Nous l'appellerons "contour sur le parcours d'outil".

L'interpréteur ne dispose d'aucun paramètre pour déterminer quel type de contour est utilisé, mais la description des contours est différente (pour la même géométrie de pièce) entre les deux types, les valeurs des diamètres dans la table d'outils seront également différentes pour les deux types.

### Contour sur le profil du matériau

Lorsque le contour est sur le profil du matériau, c'est ce tracé qui est décrit dans le programme NC. Pour un contour sur le profil du matériau, la valeur du diamètre dans la table d'outils correspond au diamètre réel de l'outil courant. Cette valeur dans la table doit être positive. Le code NC pour ce type de contour reste toujours le même à l'exception du diamètre de l'outil (actuel ou nominal).

Exemple 1 :

Voici un programme NC qui usine le matériau en suivant le profil d'un des triangles de la figure précédente. Dans cet exemple, la compensation de rayon est celle du rayon actuel de l'outil, soit 0.5". La valeur pour le diamètre dans la table d'outil est de 1.0" .

```
N0010 G41 G1 X2 Y2 (active la compensation et fait le mouvement d'entrée)
```

```
N0020 Y-1 (suit la face droite du triangle)
```

```
N0030 X-2 (suit la base du triangle)
```

```
N0040 X2 Y2 (suit l'hypoténuse du triangle)
```

```
N0050 G40 (désactive la compensation)
```

Avec ce programme, l'outil suit cette trajectoire : un mouvement d'entrée, puis la trajectoire montrée dans la partie gauche de la figure, avec un déplacement de l'outil en sens horaire autour du triangle. Noter que les coordonnées du triangle de matériau apparaissent dans le code NC. Noter aussi que la trajectoire inclus trois arcs qui ne sont pas explicitement programmés, ils sont générés automatiquement.

### Contour sur le parcours d'outil

Lorsque le contour est sur le parcours d'outil, la trajectoire décrite dans le programme correspond au parcours que devra suivre le centre de l'outil. Le bord de l'outil, à un rayon de là, (excepté durant les mouvements d'entrée) suivra la géométrie de la pièce. La trajectoire peut être créée manuellement ou par un post-processeur, selon la pièce qui doit être réalisée. Pour l'interpréteur, le parcours d'outil doit être tel que le bord de l'outil reste en contact avec la géométrie de la pièce, comme montré à gauche de la figure 7. Si une trajectoire du genre de celle montrée sur la droite de la figure 7 est utilisée, dans laquelle l'outil ne reste pas en permanence au contact avec la géométrie

de la pièce, l'interpréteur ne pourra pas compenser correctement si un outil en dessous de son diamètre nominal est utilisé.

Pour un contour sur le parcours d'outil, la valeur pour le diamètre de l'outil dans la table d'outils devra être un petit nombre positif si l'outil sélectionné est légèrement surdimensionné. La valeur du diamètre sera un petit nombre négatif si l'outil est légèrement sous-dimensionné. Si un diamètre d'outil est négatif, l'interpréteur compense de l'autre côté du contour programmé et utilise la valeur absolue donnée au diamètre. Si l'outil courant est à son diamètre nominal, la valeur dans la table d'outil doit être à zéro.

Exemple de contour sur le parcours d'outil

Supposons que le diamètre de l'outil courant monté dans la broche est de 0.97 et le diamètre utilisé en générant le parcours d'outil a été de 1.0 . Alors la valeur de diamètre dans la table d'outils pour cet outil est de -0.03. Voici un programme NC qui va usiner l'extérieur d'un triangle de la figure 7.

```
N0010 G1 X1 Y4.5 (mouvement d'alignement)
N0020 G41 G1 Y3.5 (active la compensation et premier mouvement d'entrée)
N0030 G3 X2 Y2.5 I1 (deuxième mouvement d'entrée)
N0040 G2 X2.5 Y2 J-0.5 (usinage le long de l'arc en haut du parcours d'outil)
N0050 G1 Y-1 (usinage le long du côté droit du parcours d'outil)
N0060 G2 X2 Y-1.5 I-0.5 (usinage de l'arc en bas à droite)
N0070 G1 X-2 (usinage de la base du parcours d'outil)
N0080 G2 X-2.3 Y-0.6 J0.5 (usinage de l'arc en bas à gauche)
N0090 G1 X1.7 Y2.4 (usinage de l'hypoténuse)
N0100 G2 X2 Y2.5 I0.3 J-0.4 (usinage de l'arc en haut à droite)
N0110 G40 (désactive la compensation)
```

Avec ce programme, l'outil suit cette trajectoire : un mouvement d'alignement, deux mouvements d'entrée, puis il suit une trajectoire légèrement intérieure au parcours d'outil montré sur la figure 7 en tournant en sens horaire autour de la pièce. Cette compensation est à droite de la trajectoire programmée, même si c'est G41 qui est programmé, en raison de la valeur négative du diamètre.

### Erreurs de programmation et limitations

Les messages en rapport avec la compensation de rayon d'outil, délivrés par l'interpréteur sont les suivants :

1. Impossible de changer les décalages d'axes avec la compensation de rayon d'outil
2. Impossible de changer d'unité avec la compensation de rayon d'outil
3. Impossible d'activer la compensation de rayon d'outil en dehors du plan XY
4. Action impossible, la compensation de rayon d'outil est déjà active
5. Impossible d'utiliser G28 ou G30 avec la compensation de rayon d'outil
6. Impossible d'utiliser G53 avec la compensation de rayon d'outil
7. Impossible d'utiliser le plan XZ avec la compensation de rayon d'outil
8. Impossible d'utiliser le plan YZ avec la compensation de rayon d'outil
9. Coin concave avec la compensation de rayon d'outil
10. Interférence de l'outil avec une partie finie de la pièce avec la compensation de rayon d'outil<sup>1</sup>
11. Mot D sur une ligne sans mot de commande G41 ni G42
12. Index de rayon d'outil trop grand
13. Le rayon de l'outil n'est pas inférieur au rayon de l'arc avec la compensation de rayon
14. Deux G-codes du même groupe modal sont utilisés.

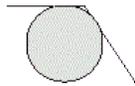
<sup>1</sup>Le terme anglais "gouging" indique une interférence entre l'outil et une partie finie de la pièce ou la paroi d'une cavité. Par extension, le terme est parfois repris pour une interférence entre le porte-outil ou la broche et la pièce.

Pour certains de ces messages, des explications sont données plus loin.

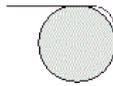
Changer d'outil alors que la compensation de rayon d'outil est active n'est pas considéré comme une erreur, mais il est peu probable que cela soit fait intentionnellement. Le rayon d'outil utilisé lors de l'établissement de la compensation continuera à être utilisé jusqu'à la désactivation de la compensation, même si un nouvel outil est effectivement utilisé.

Quand la compensation de rayon d'outil est active, il est physiquement possible de faire un cercle, dont le rayon est la moitié du diamètre de l'outil donné dans la table d'outils, il sera tangent avec l'outil en tout point de son contour.

concave corner - tool does not fit



concave arc too small - tool does not fit



En particulier, l'interpréteur traite les coins concaves et les arcs concaves plus petits que l'outil, comme des erreurs, le cercle ne peut pas être maintenu tangent avec le contour dans ces situations. Cette détection de défaut, ne limite pas les formes qui peuvent être usinées, mais elle requiert que le programmeur précise la forme exacte à usiner (ou le parcours d'outil qui doit être suivi) et non une approximation. A cet égard, l'interpréteur utilisé par EMC diffère des interpréteurs utilisés dans beaucoup d'autres contrôleurs, qui passent ces erreurs sous silence et laissent l'outil interférer avec la partie finie de la pièce (gouging) ou arrondissent des angles qui devraient être vifs. Il n'est pas nécessaire, de déplacer l'outil entre la désactivation de la compensation et sa réactivation, mais le premier mouvement suivant la réactivation sera considéré comme premier mouvement, comme déjà décrit plus tôt.

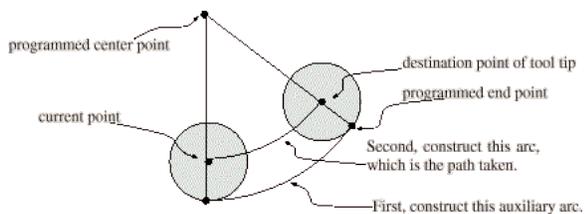
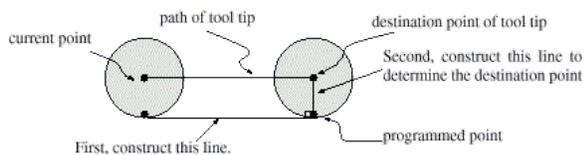
Il n'est pas possible de passer d'un index de rayon d'outil à un autre alors que la compensation est active. Il est également impossible de basculer la compensation d'un côté à l'autre avec la compensation active. Si le prochain point de destination XY est déjà dans le périmètre d'action de l'outil quand la compensation est activée, le message indiquant une interférence outil/surface finie, s'affichera quand la ligne du programme qui donne cette destination sera atteinte. Dans ce cas, l'outil a déjà usiné dans le matériau, là où il n'aurait pas dû...

Si le numéro de slot programmé par le mot D est supérieur au nombre d'emplacements disponibles dans le carrousel, un message d'erreur sera affiché. Dans l'implémentation actuelle, le nombre d'emplacements maximum est de 68.

Le message d'erreur "Deux G-codes du même groupe modal sont utilisés" est un message générique utilisé pour plusieurs jeux de G-codes. Il s'applique à la compensation de rayon d'outil, il signifie que plus d'un code G40, G41 ou G42 apparaît sur la même ligne de programme NC, ce qui n'est pas permis.

## Premier mouvement

L'algorithme utilisé lors du premier déplacement, quand c'est une ligne droite, consiste à tracer une droite, depuis le point d'arrivée, tangente à un cercle dont le centre est le point actuel, et le rayon, celui de l'outil. Le point de destination de la pointe de l'outil se trouve alors au centre d'un cercle de même rayon, tangent à la ligne droite tracée précédemment. C'est montré sur la figure 9. Si le point programmé est situé à l'intérieur de la première section d'outil (le cercle de gauche), une erreur sera signalée.



Si le premier mouvement après que la compensation de rayon d'outil a été activée est un arc, l'arc qui sera généré est dérivé d'un arc auxiliaire, qui a son centre identique à celui du point central programmé, passe par le point final de l'arc programmé et, est tangent à l'outil à son emplacement courant. Si l'arc auxiliaire ne peut pas être construit, une erreur sera signalée. L'arc généré déplacera l'outil pour qu'il reste tangent à l'arc auxiliaire pendant tout le mouvement. C'est ce que montre sur la figure 10.

Indépendamment du fait que le premier déplacement est une droite ou un arc, l'axe Z peut aussi se déplacer en même temps. Il se déplacera linéairement, comme c'est le cas quand la compensation de rayon n'est pas utilisée. Les mouvements des axes rotatifs (A, B et C) sont autorisés avec la compensation de rayon d'outil, mais leur utilisation serait vraiment très inhabituelle.

Après les mouvements d'entrée en compensation de rayon d'outil, l'interpréteur maintiendra l'outil tangent au contour programmé et du côté approprié. Si un angle aigu se trouve dans le parcours, un arc est inséré pour tourner autour de l'angle. Le rayon de cet arc sera de la moitié du diamètre de l'outil donné dans la table d'outils.

Quand la compensation de rayon est désactivée, aucun mouvement de sortie particulier n'est fait. Le mouvement suivant sera ce qu'il aurait été si la compensation n'avait jamais été activée et que le mouvement précédent ait placé l'outil à sa position actuelle.

### Programmation des mouvements d'entrée

En général, un mouvement d'alignement et deux mouvements d'entrée sont demandés pour commencer la compensation correctement. Cependant, si le contour programmé comporte des pointes et des angles aigus, un seul mouvement d'entrée (plus, éventuellement, un mouvement de pré-entrée) est demandé. La méthode générale, qui fonctionne dans toutes les situations, est décrite en premier. Elle suppose que le programmeur connaît déjà le contour et son but est d'ajouter le mouvement d'entrée.

#### Méthode générale

La méthode générale de programmation comprend un mouvement d'alignement et deux mouvements d'entrée. Les mouvements d'entrée expliqués ci-dessus, seront repris comme exemple. Voici le code correspondant :

```
N0010 G1 X1 Y4.5 (mouvement d'alignement vers le point C)
N0020 G41 G1 Y3.5 (active la compensation et fait le premier mouvement d'entrée vers le point B)
N0030 G3 X2 Y2.5 I1 (fait le second mouvement d'entrée vers le point A)
```

Voir la figure 11. La figure montre les deux mouvements d'entrée mais pas le mouvement d'alignement.

En premier, choisir un point A sur le contour où il convient d'attacher un arc d'entrée. Spécifier un arc à l'extérieur du contour qui commence au point B et s'achève au point A, tangent au contour (et aller dans la même direction que celle prévue pour tourner autour du contour). Le rayon doit être supérieur à la moitié du diamètre donné dans la table d'outils. Ensuite, tirer une ligne tangente à l'arc, du point B au point C, placé de telle sorte que la ligne BC fasse plus d'un rayon de long.

Après que la construction soit terminée, le code est écrit dans l'ordre inverse de celui de la construction. La compensation de rayon d'outil est activée après le mouvement d'alignement et avant le premier mouvement d'entrée. Dans le code précédent, la ligne N0010 fait le mouvement d'alignement, la ligne N0020 active la compensation et fait le premier mouvement d'entrée et la ligne N0030 fait le second mouvement d'entrée.

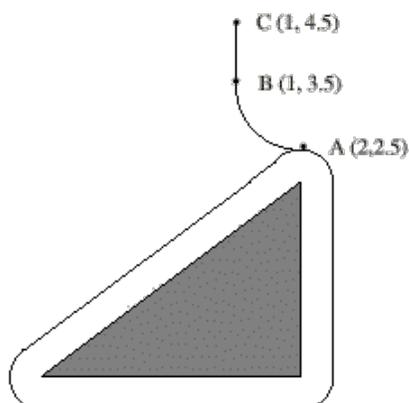


Figure 11. Cutter Radius Compensation Entry Moves

Dans cet exemple, l'arc AB et la ligne BC sont très larges, ce n'est pas nécessaire. Pour un contour sur parcours d'outil, le rayon de l'arc AB demande juste à être légèrement plus grand que la variation maximale du rayon de l'outil par rapport à son rayon nominal. Egalement, pour un contour sur parcours d'outil, le côté choisi pour la compensation doit être celui utilisé si l'outil est surdimensionné. Comme mentionné précédemment, si l'outil est sousdimensionné, l'interpréteur basculera de l'autre côté.

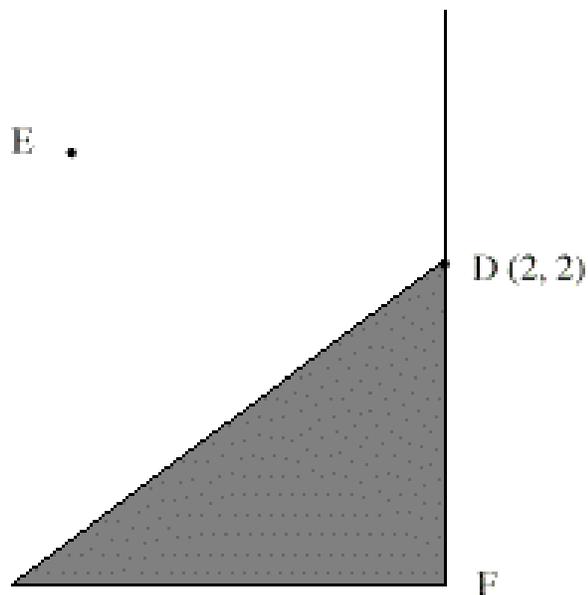
### Méthode simple

Si le contour est sur le profil du matériau et qu'il comprends des angles aigus quelque part sur le contour, une méthode simple pour faire l'entrée est possible. Voir la figure 12.

Premièrement, choisir un angle aigu, par exemple D. Ensuite, décider comment on va tourner autour du matériau depuis le point D. Dans notre exemple nous maintiendrons l'outil à gauche du profil et nous avancerons vers F. Prolonger la ligne FD (si le segment suivant du contour est un arc, prolonger la tangente à l'arc FD depuis D) pour diviser la surface extérieure au contour proche de D en deux parties. S'assurer que le centre de l'outil est actuellement dans la partie du même côté de la ligne prolongée que le matériau. Sinon, déplacer l'outil dans cette partie. Par exemple, le point E représente la position courante du centre de l'outil. Comme il est du même côté de la ligne FD prolongée que le triangle gris du matériau, aucun mouvement supplémentaire n'est nécessaire. Maintenant écrire la ligne de code NC qui active la compensation et faire le mouvement vers le point D

```
N0010 G41 G1 X2 Y2 (active la
compensation et fait le mouvement
d'entrée)
```

Cette méthode fonctionnera également avec un angle aigu sur un contour sur parcours d'outil, si l'outil est surdimensionné, mais elle échouera si il est sousdimensionné.



### **Autres points où est exécutée la compensation de rayon d'outil**

Le jeu complet de fonctions canoniques comprend des fonctions qui activent et désactivent la compensation de rayon d'outil, de sorte qu'elle puisse être activée quand le contrôleur exécute une de ces fonctions. Dans l'interpréteur cependant, ces commandes ne sont pas utilisées. La compensation est assurée par l'interpréteur et reflétée dans les sorties des commandes, c'est l'interpréteur qui continuera à diriger les mouvements du centre de l'outil. Cela simplifie le travail du contrôleur de mouvement tout en rendant le travail de l'interpréteur un peu plus difficile.

### **Algorithmes pour compensation de rayon d'outil**

L'interpréteur permet que les mouvements d'entrée et de sortie soient des arcs. Le comportement pour les mouvements intermédiaires est le même, excepté que certaines situations sont traitées comme des erreurs par l'interpréteur alors qu'elles ne le sont pas sur d'autres contrôleurs de machine.

Données relatives à la compensation de rayon d'outil :

L'interpréteur conserve trois données pour la compensation de rayon d'outil : Le réglage lui même (gauche, droite ou arrêt), `program_x` et `program_y`. Les deux dernières représentent les positions en X et en Y données dans le code NC quand la compensation est active. Quand elle est désactivée, les deux entrées sont fixées à de très petites valeurs ( $10 \times 10^{-20}$ ) dont la valeur symbolique (dans un `#define`) est "unknown". L'interpréteur utilise, les items `current_x` et `current_y` qui représentent, le centre de la pointe de l'outil (dans le système de coordonnées courant), à tout moment.

### **Exemples de Jon Elson**

Toutes les informations spécifiques au système se réfèrent au programme EMC du NIST, mais doit aussi s'appliquer aux plus modernes contrôleurs CNC. Ma méthode de vérification de ces programmes est d'abord de sélectionner l'outil zéro, de sorte que les commandes de compensation soient ignorées. Ensuite, je colle une feuille de papier sur une plaque tenue de niveau dans l'étau, une sorte de platine. J'installe une recharge de stylo à ressort dans la broche. C'est une recharge standard de stylo à bille en métal avec un ressort, dans un corps de 12mm de diamètre. Elle a un ressort pour la faire rentrer dans le corps du stylo, et un 'collet' à l'arrière qui permet à la pointe de se rétracter malgré le ressort, mais qui la laisse centrée à quelques dixièmes près. Je charge le programme avec l'outil zéro sélectionné, et il trace une ligne à l'extérieur de la pièce. (voir la figure suivante) Alors, je sélectionne un outil avec le diamètre de l'outil que j'envisage d'utiliser et je lance le programme une nouvelle fois. (Noter que la coordonnée Z dans le programme ne doit pas être changée pour éviter de plonger le stylo au travers du plateau;-) Maintenant, je dois voir si la compensation G41 ou G42 que je spécifie passe sur le côté voulu de la pièce. Sinon, je modifie avec la compensation du côté opposé, et j'édite la compensation opposée dans le programme, puis j'essaye à nouveau. Maintenant, avec l'outil sur le côté correct de la pièce, je peut vérifier si quelque part sur le parcours l'outil est 'trop gros' pour usiner les surfaces concaves. Ma vieille Allen-Bradley 7320 était très indulgente sur ce point, mais EMC ne tolère rien. Si vous avez la moindre concavité où deux lignes se rencontrent à moins de 180 degrés avec un outil de taille définies, EMC va s'arrêter là, avec un message d'erreur. Même si le gougeage est de .001mm de profondeur. Alors, je fais toujours l'approche sur le mouvement d'entrée et le mouvement de sortie juste sur un coin de la pièce, en fournissant un angle de plus de 180 degrés, afin qu'EMC ne hurle pas. Cela exige une grande attention lors de l'ajustement des points de départ et de sortie, qui ne sont pas compensés par le rayon d'outil, mais ils doivent être choisis avec un rayon approximatif bien réfléchi.

Les commandes sont :

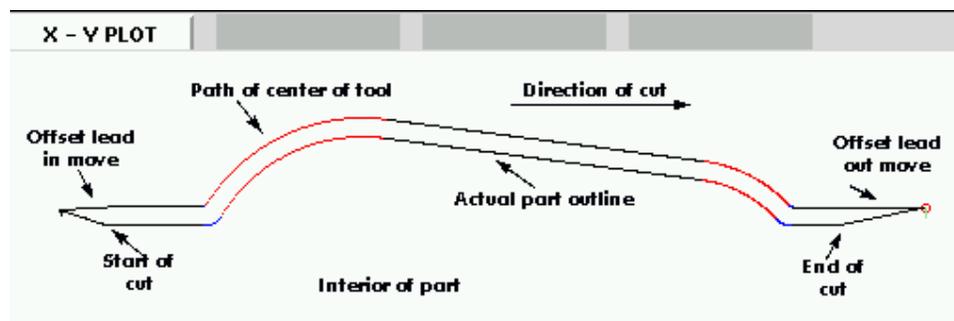
- G40 - Annuler la compensation de rayon d'outil
- G41 - Activer la compensation, outil à gauche du profil
- G42 - Activer la compensation, outil à droite du profil

Voici un petit fichier qui usine le côté d'une pièce avec de multiples arcs convexes et concaves et plusieurs lignes droites. La plupart de ces commandes ont été tracées depuis Bobcad/CAM, mais les lignes N15 et N110 ont été ajoutées par moi et certaines coordonnées dans ce contour ont été bricolées un peu par moi.

```
N10 G01 G40 X-1.3531 Y3.4
N15 F10 G17 G41 D4 X-0.7 Y3.1875 (ligne d'entrée)
N20 X0. Y3.1875
N40 X0.5667 F10
N50 G03 X0.8225 Y3.3307 R0.3
N60 G02 X2.9728 Y4.3563 R2.1875
N70 G01 X7.212 Y3.7986
N80 G02 X8.1985 Y3.2849 R1.625
N90 G03 X8.4197 Y3.1875 R0.3
N100 G01 X9.
N110 G40 X10.1972 Y3.432 (ligne de sortie)
N220 M02
```

La ligne 15 contient G41 D4, qui signifie que le diamètre de l'outil est celui de l'outil #4 dans la table d'outils, il sera utilisé pour décaler la broche de 1/2 diamètre, qui est, bien sûr, le rayon d'outil. Noter que la ligne avec la commande G41 contient le point final du mouvement dans lequel la compensation de rayon est interpolée. Cela signifie qu'au début de ce mouvement, il n'y a aucun effet de compensation et à la fin, l'outil est décalé de 100% du rayon de l'outil sélectionné. Immédiatement après le G41 il y a D4, signifiant que le décalage sera le rayon de l'outil N°4 dans la table d'outils. Noter que les DIAMETRES d'outil sont entrés dans la table d'outils. (le diamètre de l'outil de Jon est de 0.4890)

Mais, noter qu'à la ligne 110, où il y a la commande G40, l'interpolation de la compensation d'outil est en dehors de ce mouvement. La manière d'obtenir ce réglage, les mouvements des lignes 15 et 110 sont presque exactement parallèles à l'axe X et la différence dans les coordonnées Y est à la ligne où l'outil est appelé, en dehors de la compensation d'outil.



Certaines autres choses sont à noter, le programme commence avec G40, pour désactiver les compensations éventuellement actives. Cela évite un tas d'ennuis quand le programme s'arrête à cause d'une erreur de concavité, mais laisse la compensation désactivée. Noter aussi, en ligne 15, G17 est utilisé pour spécifier le plan de travail XY pour les interpolations circulaires. J'ai utilisé le format rayon pour les spécifications des arcs plutôt que la forme I, J. EMC est très pointilleux au sujet des rayons qu'il calcule à partir du format des coordonnées I, J et il doit trouver le début et la fin du mouvement avec  $10^{-11}$  unités internes, de sorte qu'il y a beaucoup de problèmes avec des arcs arbitraires. Normalement, si vous avez un arc de 90 degrés, centré sur (1.0,1.0) avec un rayon de 1", tout ira bien, mais si le rayon ne peut pas être exprimé exactement et avec juste le nombre de chiffres significatifs, ou si l'arc a un nombre étrange de degrés, alors les problèmes commencent avec EMC. Le mot R supprime tous ce désordre et il est beaucoup plus facile de travailler avec lui, de toute façon. Si l'arc est de plus de 180 degrés, R doit être négatif.

# Chapitre 12

## G Codes

Les G-codes du langage RS274/NGC sont décrits ci-dessous.

Dans une commande type, le tiret (-) signifie une valeur réelle. Comme décrite précédemment, une valeur réelle peut être (1) un nombre explicite, 4 par exemple, (2) une expression, [2+2] par exemple, (3) une valeur de paramètre, #88 par exemple, ou (4) une fonction unaire de la valeur  $\cos [0]$ , par exemple.

Dans la plupart des cas, si des mots d'axes sont donnés, parmi X, Y, Z, A, B, C, U, V, W, ils spécifient le point de destination. Les axes sont donnés dans le système de coordonnées courant, à moins qu'explicitement décrit comme étant dans le système de coordonnées absolues (machine). Où les axes sont optionnels, tout axe omis gardera sa valeur courante. Tout item dans une commande non explicitement décrit comme optionnel sera requis. Une erreur sera signalée si un item requis est omis.

Dans les commandes, les valeurs suivant les lettres sont souvent données comme des nombres explicites. Sauf indication contraire, les nombres explicites peuvent être des valeurs réelles. Par exemple, G10 L2 pourrait aussi bien être écrite G [2 \* 5] L [1 +1]. Si la valeur du paramètre 100 étaient 2, G10 L#100 signifierait également la même chose. L'utilisation de valeurs réelles qui ne sont pas des nombres explicites, comme indiqué dans les exemples sont rarement utiles.

Si L- est écrit dans une commande le "-" fera référence à "L nombre". De la même manière, le "-" dans H- peut être appelé le "H nombre" et ainsi de suite pour les autres lettres.

## 12.1 Table résumée du G-code

Codes	Descriptions	Paramètres	Sections
G0	Interpolation linéaire en vitesse rapide	n/a	12.2
G1	Interpolation linéaire en vitesse travail	n/a	12.3
G2, G3	Interpolation circulaire sens horaire, sens anti-horaire	I J K or R	12.4
G7, G8	Mode diamètre, rayon sur les tours	n/a	12.6
G10 L1	Entre longu., rayon, orientation outil dans table d'outils	P R X Z	12.8
G10 L2	Etablissement du système de coord. pièce (1 à 9)	P X Y Z A B C	12.9
G17, G18, G19	Choix du plan de travail	n/a	12.10
G20, G21	Unité machine	n/a	12.11
G28, G28.1	Aller à une position prédéfinie	n/a	12.12
G30, G30.1	Aller à une position prédéfinie	n/a	12.13
G33	Mouvement synchronisé avec la broche	K	12.14
G33.1	Taraudage rigide	K	12.14
G38.2-G38.5	Mesures au palpeur	n/a	12.15
G40	Annulation de la compensation de longueur d'outil	n/a	12.16
G41, G42	Compensation de rayon d'outil	D	12.16
G41.1, G42.1	Comp. dynamique de rayon d'outil à gauche, à droite	D L	12.16
G43	Comp. de longueur d'outil depuis la table d'outils	H	12.17
G49	Annulation de la compensation de longueur d'outil	I K	12.17
G53	Déplacements en coordonnées machine	n/a	12.18
G54-G59	Choix du système de coordonnées (1 a 9)	n/a	12.19
G59.1-G59.3	Choix du système de coordonnées	n/a	12.19
G61, G61.1	Mode trajectoire exacte, mode arrêt exact	n/a	12.20
G64	Mode trajectoire continue avec tolérance optionnelle	P	12.20
G73	Cycle de perçage avec brise copeau	R L Q	12.21
G76	Cycle de filetage multipasses préprogrammé (tour)	P Z I J R K Q H L E	12.1
G80	Révocation des codes modaux	n/a	12.23
G81	Cycle de perçage	R L P	12.24
G82-G89	Autres cycles preprogrammes	R L P Q	12.24.8
G90, G91	Types de déplacements	n/a	12.25
G92	Décalages d'origines avec mise à jour des paramètres	X Y Z A B C	12.27
G92.1, G92.2	Annulation des décalages d'origine	n/a	12.27
G92.3	Appliquer contenu des paramètres aux déc. d'origine	n/a	12.27
G93, G94, G95	Modes de vitesses	n/a	12.28
G96	Vitesse de coupe constante (IPM ou m/mn)	D S	12.29
G97	Vitesse en tours par minute	n/a	12.29
G98, G99	Options de retrait des cycles preprogrammes	n/a	12.30
F, S	F=réglage vitesse travail, S=réglage vitesse broche	n/a	15.115.2
M0, M1, M2	Contrôle du programme	n/a	13.1
M3, M4, M5	Contrôle de la broche	S	13.2
M6	Appel d'outil Tn n=numéro d'outil	T	13.3
M7, M8, M9	Contrôle arrosage	n/a	13.4
M30, M60	Pallet Shuttle	n/a	13.1
M50	Correcteur de vitesse d'avance travail	P	13.6
M51	Correcteur de vitesse de broche	n/a	13.7
M52	Correcteur dynamique de vitesse d'avance	P	13.8
M53	Vitesse constante	P	12.18
M61	Numéro de l'outil courant	Q	13.10
M62 à M65	Output control	P	13.11
M66	Input control	P, E, L, Q	13.12
M100-M199	M-Codes définis par l'utilisateur	P Q	13.13
O	O Codes	n/a	14

## 12.2 G0 : Interpolation linéaire en vitesse rapide

Pour un mouvement linéaire en vitesse rapide, programmer G0 *axes*, au moins un mot d'axe doit être présent, les autres sont optionnels. Le G0 est optionnel si le mode mouvement courant est déjà G0. Cela produira un mouvement linéaire vers le point de destination à la vitesse rapide courante (ou moins vite si la machine n'atteint pas cette vitesse). Il n'est pas prévu d'usiner la matière quand une commande G0 est exécutée.

C'est une erreur si :

- tous les axes sont omis.

Si la compensation de rayon d'outil est active, le mouvement sera différent de celui décrit ci-dessus, voir la section 17.4. Si G53 est programmé sur la même ligne, le mouvement sera également différent, voir la section 12.18.

Si un mouvement G0 déplace seulement des axes rotatifs et que la position de la cible pour ces axes est dans une échelle de -360 à 360 degrés, le mouvement sera organisé pour que chaque axe rotatif fasse moins d'un tour complet. Exemple :

```
N1 G0 X1 A[20*360]
N2 G0 A0
```

Après la ligne N1, la position de l'axe A sera 7200 degrés. 7200 degrés est égal à zéro degré sur un axe rotatif, de sorte que le mouvement linéaire rapide spécifié en ligne N2 ne produira aucun mouvement.

## 12.3 G1 : Interpolation linéaire en vitesse travail

Pour un mouvement linéaire en vitesse travail, programmer G1 *axes*, au moins un mot d'axe doit être présent, les autres sont optionnels. Le G1 est optionnel si le mode mouvement courant est déjà G1. Cela produira un mouvement linéaire vers le point de destination à la vitesse de travail courante (ou moins vite si la machine n'atteint pas cette vitesse).

C'est une erreur si :

- tous les axes sont omis.

Si la compensation de rayon d'outil est active, le mouvement sera différent de celui décrit ci-dessus, voir la section 17.4. Si G53 est programmé sur la même ligne, le mouvement sera également différent, voir la section 12.18.

## 12.4 G2, G3 : Interpolation circulaire en vitesse travail

Un mouvement circulaire ou hélicoïdal est spécifié en sens horaire avec G2 ou en sens anti-horaire avec G3. Les axes du cercle ou de l'hélicoïde, doivent être parallèles aux axes X, Y, ou Z du système de coordonnées machine. Les axes (ou, leurs équivalents, les plans perpendiculaires aux axes) sont sélectionnés avec G17 (axe Z, plan XY), G18 (axe Y, plan XZ), ou G19 (axe X, plan YZ). Si l'arc est circulaire, il se trouve dans un plan parallèle au plan sélectionné.

Si une ligne de code RS274/NGC forme un arc et inclut le mouvement d'un axe rotatif, l'axe rotatif tournera à vitesse constante, de sorte que le mouvement de l'axe rotatif commence et se termine en même temps que les autres axes XYZ. De telles lignes ne sont pratiquement jamais programmées.

Si la compensation de rayon d'outil est active, le mouvement sera différent de celui décrit ci-dessus, voir la section 12.16.

Deux formats sont autorisés pour spécifier un arc : le format centre et le format rayon.

### 12.4.1 Arc au format centre (format recommandé)

Dans le format centre, les coordonnées du point final de l'arc, dans le plan choisi, sont spécifiées par décalage du centre de l'arc depuis le point courant. Dans ce format, il est permis d'avoir le point final de l'arc identique au point courant. C'est une erreur si :

- Lorsque l'arc est projeté sur le plan choisi, la distance entre le point courant et le centre différent de la distance entre le point d'arrivée et le centre de plus de 0,0002 pouce (si les pouces sont utilisés), soit 0,002 millimètre (si les millimètres sont utilisés).

Lorsque le plan XY est sélectionné, programmer G2 Axe I- J- (ou utiliser G3 au lieu de G2). Les mots d'axes sont tous facultatifs sauf au moins un entre X et Y qui doit être utilisé. I et J sont les décalages du centre de l'arc, par rapport au point actuel (dans le sens X et Y, respectivement). I et J sont facultatifs, sauf qu'au moins un des deux doit être utilisé. C'est une erreur si :

- X et Y sont tous les deux omis.
- I et J sont tous les deux omis .

Lorsque le plan XZ est sélectionné, programmer G2 axes I- K- (ou utiliser G3 au lieu de G2). Les mots d'axes sont tous facultatifs sauf au moins un entre X et Z qui doit être utilisé. I et K sont les décalages du centre de l'arc, par rapport au point actuel (dans le sens X et Z , respectivement). I et K sont facultatifs, sauf qu'au moins un des deux doit être utilisé. C'est une erreur si :

- X et Z sont tous les deux omis.
- I et K sont tous les deux omis .

Lorsque le plan YZ est sélectionné, programmer G2 axes J- K- (ou utiliser G3 au lieu de G2). Les mots d'axes sont tous facultatifs sauf au moins un entre Y et Z qui doit être utilisé. J et K sont les décalages du centre de l'arc, par rapport au point actuel (dans le sens Y et Z , respectivement). J et K sont facultatifs, sauf qu'au moins un des deux doit être utilisé. C'est une erreur si :

- Y et Z sont tous les deux omis.
- J et K sont tous les deux omis .

Voici un exemple de commande pour usiner un arc au format centre : G17 G2 x10 y16 i3 j4 z9.

Cela signifie faire un mouvement en arc ou hélicoïdal en sens horaire (vu du côté positif de l'axe Z), se terminant en X=10, Y=16 et Z=9, avec son centre décalé du point actuel de 3 unités en X et de 4 unités en Y. Si la position courante est X=7, Y=7 au final, le centre sera en X=10, Y=11. Si la position de départ de l'axe Z est 9, ce sera un arc circulaire, autrement ce sera un arc hélicoïdal. Le rayon de l'arc sera de 5.

Dans le format centre, le rayon de l'arc n'est pas spécifié, mais il peut facilement être trouvé comme étant la distance entre le centre de l'arc et soit, son point d'arrivée soit, le point courant.

### 12.4.2 Arcs au format rayon (format non recommandé)

Dans le format rayon, les coordonnées du point final de l'arc, dans le plan choisi, sont spécifiées en même temps que le rayon de l'arc. Programmer G2 axes R- (ou utiliser G3 au lieu de G2). R est le rayon. Les mots d'axes sont facultatifs sauf au moins un des deux du plan choisi, qui doit être utilisé. Un rayon positif indique que l'arc fait moins de 180 degrés, alors qu'un rayon négatif indique un arc supérieur à 180 degrés. Si l'arc est hélicoïdal, la valeur du point d'arrivée de l'arc dans les coordonnées de l'axe perpendiculaire au plan choisi sera également spécifiée.

C'est une erreur si :

- Les deux mots d'axes pour le plan choisi sont omis.
- Le point d'arrivée de l'arc est identique au point courant.

Ce n'est pas une bonne pratique de programmer au format rayon, des arcs qui sont presque des cercles entiers ou des demi-cercles, car un changement minime dans l'emplacement du point d'arrivée va produire un changement beaucoup plus grand dans l'emplacement du centre du cercle (et

donc, du milieu de l'arc). L'effet de grossissement est tellement important, qu'une erreur d'arrondi peut facilement produire un usinage hors tolérance. Par exemple, 1% de déplacement de l'extrémité d'un arc de 180 degrés produit 7% de déplacement du point situé à 90 degrés le long de l'arc. Les cercles presque complets sont encore pires. Autrement l'usinage d'arcs dans une gamme de petits à 165 degrés ou de 195 à 345 degrés sera bon.

Voici un exemple de commande pour usiner un arc au format `rayon : G17 G2 x 10 y 15 r 20 z 5`.

Cela signifie faire un mouvement en arc ou hélicoïdal en sens horaire (vu du côté positif de l'axe Z), se terminant en X=10, Y=15 et Z=5, avec un rayon de 20. Si la valeur de départ de Z est 5, Ce sera un arc de cercle parallèle au plan XY sinon, ce sera un arc hélicoïdal.

## 12.5 G4 : Tempo

Pour une tempo, programmer G4 P- . Les axes s'immobiliseront pour une durée de P secondes. C'est une erreur si :

- Le nombre P est négatif.

## 12.6 G7 : Mode diamètre

Sur un tour, programmer G7 pour passer l'axe X en mode diamètre. En mode diamètre, les mouvements de l'axe X font la moitié de la cote programmée. Par exemple, X10 placera l'outil à 5 unités du centre, ce qui produira bien une pièce d'un diamètre de 10 unités.

## 12.7 G8 : Mode rayon

Sur un tour, programmer G8 pour passer l'axe X en mode rayon. En mode rayon, les mouvements de l'axe X sont égaux à la cote programmée. Ce qui signifie que X10 placera l'outil à 10 unités du centre et aura pour résultat une pièce d'un diamètre de 20 unités. G8 est le mode par défaut à la mise sous tension.

## 12.8 G10 L1 : Entrée dans la table d'outils

Programmer un G10 L1 pour ajuster une entrée dans la table d'outils depuis un programme ou depuis la fenêtre d'entrées manuelles.

G10 recharge la table d'outils.

Format de la commande G10 L1 :

G10 L1 P[Numéro d'outil] R[rayon] X[correcteur en X] Z[correcteur en Z] Q[orientation]

Pour connaître l'orientation des outils, se référer à la figure 7.1

## 12.9 G10 L2 : Établissement du système de coordonnées pièce

Le langage RS274/NGC utilise les systèmes de coordonnées décrits à la section 7.6.

Pour définir les coordonnées de l'origine d'un système de coordonnées pièce, programmer G10 L2 P- axes, où l'entier P, compris entre 1 et 9 (correspondance de G54 à G59.3) et tous les mots d'axes,

sont optionnels. L'origine du système de coordonnées spécifié par le nombre P, est remplacée par les valeurs indiquées (en termes de coordonnées machine non décalées). Seules, les coordonnées pour lesquelles un mot d'axe est inclu sur la ligne, seront remplacées. Etre en mode incrémental (G91), est sans effet sur G10 L2.

C'est une erreur si :

- Le nombre P n'est pas un entier compris entre 1 et 9.

Si un décalage d'origine créé avec G92 est actif avant la commande G10 L2, il reste actif après.

Le système de coordonnées dont l'origine est définie par la commande G10 peut être actif ou non au moment de l'exécution de G10. Si il est actif à ce moment là, les nouvelles coordonnées prennent effet immédiatement.

Exemple : G10 L2 P1 X3.5 Y17.2 place l'origine du premier système de coordonnées (celui sélectionné par G54) au point où X=3.5 et Y=17.2 (en coordonnées absolues). La coordonnée Z de l'origine, ainsi que les coordonnées de tous les autres axes, restent inchangées puisque seuls X et Y étaient spécifiés.

## 12.10 G17, G18, G19, G17.1, G18.1, G19.1 : Choix du plan de travail

Ces codes sélectionnent le plan courant comme ci-dessous :

G17	XY (par défaut)
G18	ZX
G19	YZ
G17.1	UV
G18.1	WU
G19.1	VW

Les différents effets provoqués selon la sélection du plan de travail, sont expliqués dans les sections [12.4](#) et [7.2.10](#)

## 12.11 G20, G21 : Choix des unités machine

Programmer G20 pour utiliser le pouce comme unité de longueur. Programmer G21 pour utiliser le millimètre.

C'est toujours une bonne pratique de programmer soit G20, soit G21, au début d'un programme, avant tout mouvement et de ne plus en changer ailleurs dans le programme. C'est la responsabilité de l'opérateur d'être sûr que toutes les longueurs sont appropriées pour l'utilisation des unités actuelles.

## 12.12 G28, G28.1 : Aller à une position prédéfinie

La commande G28 utilise les valeurs stockées dans les paramètres 5161 à 5166 comme valeurs absolues pour effectuer un mouvement en vitesse rapide de la position courante à la position définie dans les paramètres. Les valeurs stockées dans les paramètres font référence au système de coordonnées absolues qui est le système de coordonnées machine.

La commande G28 *axes* effectue un déplacement en vitesse rapide depuis la position courante jusqu'à la position spécifiée par *axes*, suivi d'un mouvement rapide à la position prédéfinie dans les paramètres 5161 à 5166.

G28.1 enregistre la position absolue courante dans les paramètres 5161 à 5166.

C'est une erreur si :

- La compensation de rayon d'outil est active.

## 12.13 G30, G30.1 : Aller à une position prédéfinie

La commande G30 utilise les valeurs stockées dans les paramètres 5181 à 5186 comme valeurs absolues pour effectuer un mouvement en vitesse rapide de la position courante à la position définie dans les paramètres. Les valeurs stockées dans les paramètres font référence au système de coordonnées absolues qui est le système de coordonnées machine.

La commande G30 *axes* effectue un déplacement en vitesse rapide depuis la position courante jusqu'à la position spécifiée par *axes*, suivi d'un mouvement rapide à la position prédéfinie dans les paramètres 5181-5186.

G30.1 enregistre la position absolue courante dans les paramètres 5181-5186.

Les paramètres de G30 peuvent être utilisés pour déplacer l'outil quand un M6 est programmé et que la variable [TOOL\_CHANGE\_AT\_G30]=1 dans la section [EMCIO] du fichier ini.

C'est une erreur si :

- La compensation de rayon d'outil est active.

## 12.14 G33, G33.1 : Mouvement avec broche synchronisée

Pour un mouvement avec broche synchronisée dans une direction, programmer G33 X- Y- Z- K- où K donne la longueur du mouvement en XYZ pour chaque tour de broche. Par exemple, il commence à Z=0, G33 Z-1 K.0625 produira un mouvement d'un pouce de long en Z, et en même temps 16 tours de broche. Cette commande pourrait être la base d'un programme pour faire un filetage de 16 filets par pouce. Un autre exemple en métrique, G33 Z-15 K1.5 produira un mouvement de 15mm de long pendant que la broche fera 10 tours soit un pas de 1.5mm.

Pour un taraudage rigide avec broche synchronisée et mouvement de retour, programmer G33.1 X- Y- Z- K- où K- donne la longueur du mouvement pour chaque tour de broche. Un mouvement de taraudage rigide suit cette séquence :

- Un mouvement aux coordonnées spécifiées, synchronisé avec la rotation de la broche, avec un ratio donné et débutant sur l'impulsion d'index du codeur de la broche.
- Quand le point final est atteint, la commande inverse le sens de rotation de la broche (ex : de 300 tours/mn en sens horaire à 300 tours/mn en sens anti-horaire)
- Le mouvement reste synchronisé en continu avec la broche même **au delà** de la coordonnée du point final spécifiée pendant l'arrêt de la broche et son inversion.
- Le mouvement synchronisé se poursuit pour revenir aux coordonnées initiales.
- Quand les coordonnées d'origines sont atteintes, la commande inverse la broche une seconde fois (ex : de 300tr/mn sens anti-horaire à 300tr/mn sens horaire)
- Le mouvement reste synchronisé même **au delà** des coordonnées d'origine pendant que la broche s'arrête et s'inverse.
- Un mouvement **non synchronisé** ramène le mobile en arrière, aux coordonnées d'origine.

Tous les mouvements avec broche synchronisée ont besoin d'un index de broche, pour conserver la trajectoire prévue. Un mouvement avec G33 se termine au point final programmé, un mouvement avec G33.1 se termine aux coordonnées initiales.

Les mots d'axes sont facultatifs, sauf au moins un qui doit être utilisé.

C'est une erreur si :

- Tous les axes sont omis.
- La broche ne tourne pas quand cette commande est exécutée.
- Le mouvement linéaire requis excède les limites de vitesse machine en raison de la vitesse de broche.

## 12.15 G38.x : Mesure au palpeur

Programmer G38.2 axes, G38.3 axes, G38.4 axes, ou G38.5 axes pour effectuer une mesure au palpeur. Au moins un mot d'axe est obligatoire, les autres sont optionnels. L'outil dans la broche doit être un palpeur.

C'est une erreur si :

- Le point programmé est le même que le point courant.
- Aucun mot d'axe n'est utilisé.
- La compensation de rayon d'outil est active.
- La vitesse travail est à zéro.
- Le palpeur est déjà au contact de la cible.

En réponse à cette commande, la machine déplace le point contrôlé (qui est la pointe ou la boule du palpeur), en ligne droite, à la vitesse actuelle, vers le point programmé, en mode vitesse inverse du temps; la vitesse est telle, que le mouvement depuis le point courant jusqu'au point programmé, prendra le temps spécifié. Le mouvement s'arrête lorsque le point programmé est atteint, ou quand l'entrée du palpeur bascule dans l'état attendu<sup>1</sup>. Le tableau 12.1 montre la signification des différents codes de mesure.

TAB. 12.1 – Codes de mesure

Code	Etat ciblé	Direction	Signal d'erreur
G38.2	Contact	Vers la pièce	Oui
G38.3	Contact	Vers la pièce	Non
G38.4	Pas de contact	Depuis la pièce	Oui
G38.5	Pas de contact	Depuis la pièce	Non

Après une mesure réussie, les paramètres 5061 à 5069 contiendront les coordonnées de l'emplacement du point contrôlé lors du changement d'état du palpeur. Après une mesure manquée, ils contiendront les coordonnées du point programmé. Le paramètre 5070 est mis à 1 si la mesure est réussie et à 0 si elle est manquée. Si la mesure n'a pas réussi, G38.2 et G38.4 signaleront une erreur.

Un commentaire de la forme (PROBEOPEN *filename.txt*) ouvrira le fichier *filename.txt* et y enregistrera les 9 coordonnées de chaque mesure réussie. Le fichier doit être fermé avec le commentaire (PROBECLOSE).

## 12.16 G40, G41, G42, G41.1, G42.1 : Compensation de rayon d'outil

Pour désactiver la compensation de rayon d'outil, programmer G40. Le mouvement suivant doit être une droite. Ce n'est pas une erreur de désactiver la compensation quand elle est déjà inactive. C'est une erreur de programmer un mouvement en arc avec G2 ou G3 derrière un G40.

Pour activer la compensation de rayon d'outil à gauche du profil, programmer G41.

Pour activer la compensation de rayon d'outil à droite du profil, programmer G42.

<sup>1</sup>En réalité, le mouvement continue légèrement après que l'état du palpeur ait basculé, à cause des contraintes d'accélération/décélération de la machine.

Le mouvement d'entrée doit être au moins aussi long que le rayon de l'outil. Le mouvement d'entrée peut être effectué en vitesse rapide.

La compensation de rayon d'outil ne peut être activée que si le plan de travail courant est le plan XY.

Le comportement du centre d'usinage quand la compensation de rayon d'outil est active est décrit dans la section [17.4](#)

### 12.16.1 Compensation de rayon d'outil depuis une table d'outils

Pour activer la compensation de rayon d'outil à gauche du profil, c'est à dire, la fraise reste à gauche de la trajectoire programmée quand le rayon d'outil est positif, programmer G41 D-. Pour activer la compensation de rayon d'outil à droite du profil, c'est à dire, la fraise reste à droite de la trajectoire programmée quand le rayon d'outil est positif, programmer G42 D-. Le mot D est optionnel, si il n'y a pas de mot D, le rayon de l'outil monté actuellement dans la broche est utilisé. Si D est utilisé, sa valeur devrait être le numéro de l'outil monté dans la broche, cependant, ce n'est pas obligatoire. Ce n'est pas une erreur d'avoir D à zéro, le rayon utilisé sera de zéro.

C'est une erreur si :

- La valeur de D n'est pas un entier, il est négatif ou supérieur au nombre d'emplacements dans le carrousel.
- Le plan de travail n'est pas YZ.
- La compensation de rayon d'outil est activée alors qu'elle est déjà active.

### 12.16.2 Compensation dynamique de rayon d'outil

Pour activer la compensation dynamique de rayon d'outil à gauche du profil, programmer G41.1 D-L-. Pour activer la compensation dynamique de rayon d'outil à droite du profil, programmer G42.1 D-L-. Le mot D spécifie le diamètre de l'outil. Le mot L spécifie l'orientation des arêtes de l'outil, et par défaut 0 si non spécifié.

C'est une erreur si :

- Le plan yz est le plan de travail actif.
- La valeur de L n'est pas comprise entre 0 et 9 inclus.
- La compensation de rayon d'outil est activée alors qu'elle est déjà active.

## 12.17 G43, G43.1, G49 : Compensation de longueur d'outil

### 12.17.1 G43, G43.1 : Activation de la compensation de longueur d'outil

G43 et G43.1 modifient les mouvements ultérieurs en décalant, de la longueur de l'outil, les coordonnées de Z et/ou de X. G43 et G43.1 ne provoquent aucun mouvement. L'effet de la compensation ne se produira qu'au prochain mouvement de l'axe compensé.

#### 12.17.1.1 G43 : Compensation de longueur d'outil depuis une table d'outils

Pour utiliser la compensation de longueur d'outil depuis une table d'outils, programmer G43 H-, où la valeur de H est l'index souhaité dans la table d'outils. La valeur de H n'est pas obligatoirement la même que celle du numéro d'outil monté dans la broche. Ce n'est pas une erreur d'avoir la valeur de H à zéro, une compensation de zéro sera utilisée.

C'est une erreur si :

- La valeur de H n'est pas un entier, il est négatif, ou il est supérieur au nombre d'outils dans le carrousel.

### 12.17.1.2 G43.1 : Compensation dynamique de longueur d'outil

Pour utiliser la compensation dynamique de longueur d'outil depuis un programme, utiliser G43.1 I- K-, où I- donne la compensation de longueur d'outil en X (pour les tours) et K- donne la compensation de longueur en Z (pour les tours et les fraiseuses).

C'est une erreur si :

- Une commande de mouvement est sur la même ligne que G43.1

### 12.17.2 G49 : Annulation de la compensation de longueur d'outil

Pour annuler la compensation de longueur d'outil, programmer G49.

Ce n'est pas une erreur de programmer une compensation qui est déjà utilisée. Ce n'est pas non plus une erreur d'annuler la compensation de longueur d'outil alors qu'aucune n'est couramment utilisée.

## 12.18 G53 : Mouvement en coordonnées absolues

Pour un déplacement à un point exprimé en coordonnées absolues, programmer G1 G53 X- Y- Z- A- B- C- (ou utiliser G0 à la place de G1), au moins un mot d'axe est obligatoire, les autres sont optionnels. Le G0 ou le G1 est optionnel si il est déjà le mode de mouvement courant. G53 n'est pas modal, il doit être programmé sur chaque ligne où il doit être actif. Il produit un mouvement linéaire coordonné au point programmé. Si G1 est actif, la vitesse travail courante est utilisée si la machine est assez rapide. Si G0 est actif, la vitesse rapide courante sera utilisée si la machine est assez rapide.

C'est une erreur si :

- G53 est utilisé sans que G0 ou G1 ne soit actif.
- G53 est utilisé alors que la compensation de rayon d'outil est active.

Voir la section 7.6 pour une vue complète des systèmes de coordonnées.

## 12.19 G54 à G59.3 : Choix du système de coordonnées

Le code G54 est apparié avec le système de coordonnées pièce N°1, pour le choisir programmer G54 et ainsi de suite pour les autres systèmes. Les systèmes de coordonnées appariés à un G-code sont les suivants : (1 avec G54), (2 avec G55), (3 avec G56), (4 avec G57), (5 avec G58), (6 avec G59), (7 avec G59.1), (8 avec G59.2) et (9 avec G59.3).

C'est une erreur si :

- Un de ces G-codes est utilisé alors que la compensation de rayon d'outil est active.

Voir la section 7.6 pour une vue complète des systèmes de coordonnées.

## 12.20 G61, G61.1, G64 : Types de contrôle de trajectoire

Programmer G61 pour passer la machine en mode de trajectoire exacte, G61.1 pour la passer en mode arrêt exact, ou G64 P- pour le mode trajectoire continue avec tolérance optionnelle. Ce n'est pas une erreur de programmer un mode déjà actif. Voir la section 7.2.15 pour une discussion de ces modes.

## 12.21 G73 : Cycle de perçage avec brise copeaux

Le cycle G73 est destiné au perçage profond ou au fraisage avec brise-copeaux. Les retraits, au cours de ce cycle, fragmentent les copeaux longs (qui sont fréquents lors du perçage dans l'aluminium). Ce cycle utilise la valeur Q qui représente un incrément "delta" le long de l'axe Z. Programmer G73 X- Y- Z- A- B- C- R- L- Q-

donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un mouvement de l'axe Z seul, en vitesse travail, sur la position la moins profonde entre, un incrément delta, ou la position de Z programmée.
3. Une petite remontée en vitesse rapide.
4. Répétition des étapes 2 et 3 jusqu'à ce que la position programmée de Z soit atteinte à l'étape 2.
5. Un mouvement de l'axe Z en vitesse rapide vers le plan de retrait.

C'est une erreur si :

- La valeur de Q est négative ou égale à zéro.

## 12.22 G76 : Cycle de filetage préprogrammé

Programmer G76 P- Z- I- J- R- K- Q- H- E- L- pour produire un cycle de filetage multi-passes. C'est une erreur si :

- Le plan de travail actif n'est pas ZX.
- D'autres mots d'axes comme X- ou Y-, sont spécifiés.
- La dégressivité R- est inférieure à 1.0.
- Certains mots requis ne sont pas spécifiés.
- P-, J-, K- ou H- est négatif.
- E- est supérieure à la moitié de la longueur de la ligne pilote.

La ligne pilote (drive line) est une ligne imaginaire, parallèle à l'axe de la broche, située en sécurité à l'extérieur du matériau à fileter. La ligne pilote va du point initial en Z jusqu'à la fin du filetage donnée par la valeur de Z- dans la commande.

Le pas du filet, ou la longueur du déplacement par tour est donné par la valeur de P-.

La crête du filet est donnée par la valeur de I-, c'est une cote entre la ligne pilote et la surface de la pièce. Une valeur négative de I-, indique un filetage externe et une valeur positive, indique un filetage interne. C'est généralement à ce diamètre nominal que le matériau est cylindré avant de commencer le cycle G76.

La profondeur de la passe initiale est donnée par la valeur de J-. La première passe sera à J- au delà de la crête du filet. J- est positif, même quand I- est négatif.

La profondeur du filet est donnée par la valeur de K-. La dernière passe du filetage sera à K- au delà de la crête du filet. K- est positif, même quand I- est négatif.

La dégressivité de la profondeur de passe est donnée par la valeur de R-. R1.0 sélectionne une profondeur de passe constante pour les passes successives du filetage. R2.0 sélectionne une surface constante. Les valeurs comprises entre 1.0 et 2.0 sélectionnent une profondeur décroissante mais une surface croissante. Enfin, les valeurs supérieures à 2.0 sélectionnent une surface décroissante. Attention : les valeurs inutilement hautes de dégression, demanderont un nombre inutilement important de passes.

L'angle de pénétration oblique est donné par la valeur de Q- c'est l'angle (en degrés) décrivant de combien les passes successives doivent être décalées le long de l'axe. C'est utilisé pour faire enlever

plus de matériau d'un côté de l'outil que de l'autre. Une valeur positive de  $Q$  fait couper d'avantage le bord principal de l'outil. Typiquement, les valeurs sont 29, 29.5 ou 30 degrés.

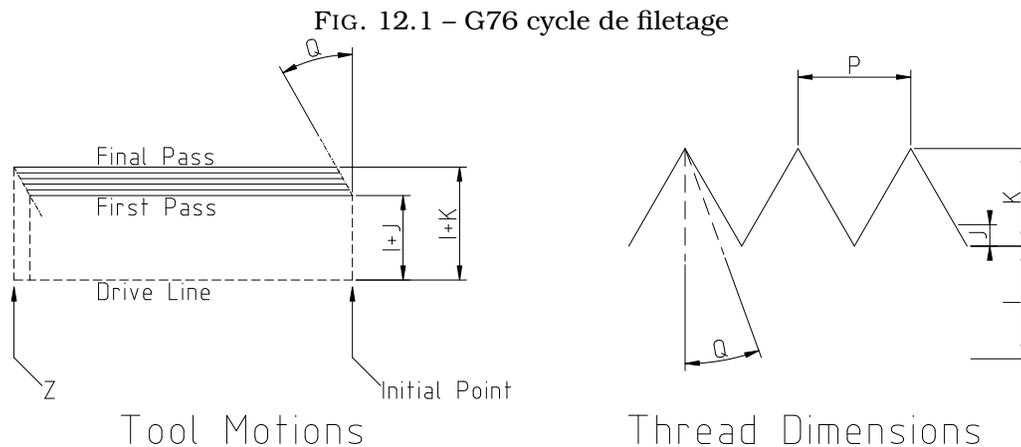
Le nombre de passes de finition est donné par la valeur de  $H-$ . Les passes de finition sont des passes additionnelles en fond de filet. Pour ne pas faire de passe de finition, programmer  $H0$ .

Les entrées et sorties de filetage peuvent être programmées côniques avec les valeurs de  $E-$  et  $L-$ .  $E-$  donne la longueur de la partie cônique le long de l'axe.  $E0.2$  donnera un cône pour le premier et le dernier filet, de 0.2 unités dans le sens du filetage.  $L-$  est utilisé pour spécifier quelles extrémités du filetage doivent être côniques. Programmer  $L0$  pour aucune (par défaut),  $L1$  pour une entrée cônique,  $L2$  pour une sortie cônique, ou  $L3$  pour l'entrée et la sortie côniques.

L'outil fera une brève pause pour la synchronisation avant chaque passe de filetage, ainsi une gorge de dégagement sera requise à l'entrée, à moins que le début du filetage ne soit après l'extrémité de la pièce ou qu'un cône d'entrée soit utilisé.

À moins d'utiliser un cône de sortie, le mouvement de sortie (retour rapide sur X d'origine) n'est pas synchronisé sur la vitesse de broche. Avec une broche lente, la sortie pourrait se faire sur une petite fraction de tour. Si la vitesse de broche est augmentée après qu'un certain nombre de passes ne soient déjà faites, la sortie va prendre une plus grande fraction de tour, il en résultera un usinage très brutal pendant ce nouveau mouvement de sortie. Ceci peut être évité en prévoyant une gorge de sortie, ou en ne changeant pas la vitesse de broche pendant le filetage.

Un exemple de programme, `g76.ngc` montre l'utilisation d'un cycle de filetage G76, il peut être visualisé et exécuté sur n'importe quelle machine utilisant la configuration `sim/lathe.ini`.



## 12.23 G80 : Révocation des codes modaux

Programmer  $G80$  pour s'assurer qu'aucun mouvement d'axe ne surviendra sans G-code modal. C'est une erreur si :

- Des mots d'axes sont programmés quand  $G80$  est actif, sans qu'un G-code modal du groupe 0 ne soit programmé avec les mots d'axes.

## 12.24 Les cycles préprogrammés

Les cycles préprogrammés de  $G81$  à  $G89$  sont décrits dans cette section. Deux exemples sont donnés plus bas avec la description de  $G81$ .

Tous les cycles préprogrammés sont effectués dans le respect du plan de travail courant. N'importe lequel des six plans de travail peut être choisi. Dans cette section, la plupart des descriptions

supposeront que le plan de travail XY est le plan courant. Le comportement reste analogue pour les autres plans de travail et les mots corrects doivent être utilisés.

Les mots d'axes rotatifs sont autorisés dans les cycles préprogrammés, mais il est préférable de les omettre. Si les mots d'axes rotatifs sont utilisés, leurs valeurs doivent rester les mêmes que celles de la position courante, de sorte qu'ils ne tournent pas.

### 12.24.1 Mots communs

Tous les cycles préprogrammés utilisent les mots X-, Y-, R-, et Z-. La position de R- (signifiant rétraction) est perpendiculaire au plan de travail courant (axe Z pour le plan XY, axe X pour le plan YZ, axe Y pour le plan XZ, etc.). Quelques cycles préprogrammés utilisent des arguments supplémentaires.

### 12.24.2 Mots « sticky »

Dans les cycles préprogrammés, un nombre est appelé “sticky” (collant) si, quand le même cycle est répété sur plusieurs lignes de code en colonne, le nombre doit être indiqué la première fois, mais qu'il devient optionnel pour le reste des lignes suivantes. Les nombres “sticky” conservent leur valeur tant qu'ils ne sont pas explicitement programmés avec une nouvelle valeur. La valeur de R est toujours “sticky”.

En mode de déplacements incrémentaux (G91), les valeurs X-, Y-, et R- sont traitées comme des incréments depuis la position courante, Z est un incrément depuis la position de l'axe Z avant le mouvement impliquant l'axe Z. En mode de déplacements absolus, les valeurs de X-, Y-, R-, et Z- sont des positions absolues dans le système de coordonnées courant.

La valeur L- est optionnelle, elle représente le nombre de répétitions.  $L = 0$  n'est pas permis. Si les répétitions sont utilisées, elles le sont normalement en mode incrémental, de sorte que la même séquence de mouvements puisse être répétée à plusieurs endroits, également espacés, le long d'une ligne droite. En mode de déplacements absolus,  $L > 1$  signifie “faire le même cycle au même endroit, plusieurs fois”. L'omission du mot L revient à spécifier  $L = 1$ . La valeur de L n'est pas “sticky”.

### 12.24.3 Répétition de cycle

Quand  $L > 1$  en mode incrémental et XY comme plan courant, les positions X et Y sont déterminées en ajoutant les valeurs X et Y de la commande à celles de la position courante, pour le premier trajet ou ensuite, à celles de la position finale du précédent trajet, pour les répétitions. Ainsi, si vous programmez L10, vous obtiendrez 10 cycles. Le premier cycle sera la distance X, Y depuis la position d'origine. Les positions de R- et Z- ne changent pas durant toutes les répétitions.

### 12.24.4 Mode de retrait

La hauteur du mouvement de retrait à la fin de chaque répétition (appelée “plan de retrait” dans les descriptions suivantes) est déterminée par le mode de retrait : retrait sur la position initiale de Z, si elle est au dessus de la valeur de R- et que le mode de retrait est G98, OLD\_Z, sinon, à la position de R-. Voir la section [12.30](#)

### 12.24.5 Erreurs des cycles préprogrammés

C'est une erreur si :

- Tous les mots X, Y et Z sont manquants durant un cycle préprogrammé.

- Un nombre P est requis mais un nombre P négatif est utilisé.
- Un nombre L est utilisé mais n'est pas un entier positif.
- Un mouvement d'axe rotatif est utilisé durant un cycle préprogrammé.
- Une vitesse inverse du temps est activée durant un cycle préprogrammé.
- La correction de rayon d'outil est activée durant un cycle préprogrammé.

Quand le plan XY est actif, la valeur de Z est "sticky", et c'est une erreur si :

- La valeur de Z est manquante alors qu'un même cycle préprogrammé n'a pas encore été activé.
- La valeur de R est inférieure à celle de Z.

Si un autre plan est actif, les conditions d'erreur sont analogues à celles du plan XY décrites ci-dessus.

### 12.24.6 Mouvement préliminaire et Intermédiaire

Tout au début de l'exécution d'un cycle préprogrammé, avec le plan courant XY, si la position actuelle de Z est en dessous de la position de retrait R, l'axe Z va à la position R. Ceci n'arrive qu'une fois, sans tenir compte de la valeur de L.

En plus, au début du premier cycle et à chaque répétition, un ou deux des mouvements suivants sont faits :

1. Un déplacement en ligne droite, parallèle au plan XY, vers le position programmée.
2. Un déplacement en ligne droite, de l'axe Z seul vers la position de retrait R, si il n'est pas déjà à cette position R.

Si un autre plan est actif, le mouvement préliminaire et intermédiaire est analogue.

### 12.24.7 G81 : Cycle de perçage

Le cycle G81 est destiné au perçage. Programmer G81 X- Y- Z- A- B- C- R- L- donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un déplacement de l'axe Z seul à la vitesse programmée, vers la position Z programmée.
3. Retrait de l'axe Z en vitesse rapide jusqu'au plan de retrait R.

**Exemple 1.** Supposons que la position courante soit, X1, Y2, Z3 dans le plan XY, la ligne de codes suivante est interprétée :

```
G90 G81 G98 X4 Y5 Z1.5 R2.8
```

Le mode de déplacements absolus est appelé (G90), le plan de retrait est positionné sur OLD\_Z (G98), l'appel du cycle de perçage G81 va lancer ce cycle une fois. La position X deviendra celle demandée, X4. La position de Y deviendra celle demandée, Y5. La position de Z deviendra celle demandée, Z1.5. La valeur de R fixe le plan de retrait de Z à 2.8. La valeur de OLD\_Z est 3. Les mouvements suivants vont se produire.

1. Un mouvement en vitesse rapide, parallèle au plan XY vers X4, Y5, Z3
2. Un mouvement en vitesse rapide, parallèle à l'axe Z vers X4, Y5, Z2.8
3. Un mouvement en vitesse travail, parallèle à l'axe Z vers X4, Y5, Z1.5
4. Un mouvement en vitesse rapide, parallèle à l'axe Z vers X4, Y5, Z3

**Exemple 2.** Supposons que la position courante soit, X1, Y2, Z3 dans le plan XY, la ligne de codes suivante est interprétée :

G91 G81 G98 X4 Y5 Z-0.6 R1.8 L3

Le mode de déplacements incrémentaux est appelé (G91), le plan de retrait est positionné sur OLD\_Z (G98), l'appel du cycle de perçage G81 demande 3 répétitions du cycle. La valeur demandée de X est 4, la valeur demandée de Y est 5, la valeur demandée de Z est -0.6 et le retrait R est à 1.8. La position initiale de X sera 5 (1+4), la position initiale de Y sera 7 (2+5), le plan de retrait sera positionné sur 4.8 (1.8+3) et Z positionné sur 4.2 (4.8-0.6). OLD\_Z est à 3.

Le premier mouvement en vitesse rapide le long de l'axe Z vers X1, Y2, Z4.8), puisque OLD\_Z est inférieur au plan de retrait.

La première répétition produira 3 mouvements.

1. Un déplacement en vitesse rapide, parallèle au plan XY vers X5, Y7, Z4.8
2. Un déplacement en vitesse travail, parallèle à l'axe Z vers X5, Y7, Z4.2
3. Un déplacement en vitesse rapide, parallèle à l'axe Z vers X5, Y7, Z4.8

La deuxième répétition produira 3 mouvements. La position de X est augmentée de 4 et passe à 9, la position de Y est augmentée de 5 et passe à 12.

1. Un déplacement en vitesse rapide, parallèle au plan XY vers X9, Y12, Z4.8
2. Un déplacement en vitesse travail, parallèle à l'axe Z vers X9, Y12, Z4.2
3. Un déplacement en vitesse rapide, parallèle à l'axe Z vers X9, Y12, Z4.8

La troisième répétition produira 3 mouvements. La position de X est augmentée de 4 et passe à 13, la position de Y est augmentée de 5 et passe à 17.

1. Un déplacement en vitesse rapide, parallèle au plan XY vers X13, Y17, Z4.8
2. Un déplacement en vitesse travail, parallèle à l'axe Z vers X13, Y17, Z4.2
3. Un déplacement en vitesse rapide, parallèle à l'axe Z vers X13, Y17, Z4.8

### 12.24.8 G82 : Cycle de perçage avec temporisation

Le cycle G82 est destiné au perçage. Programmer G82 X- Y- Z- A- B- C- R- L- P- donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un déplacement de l'axe Z seul en vitesse programmée, vers la position Z programmée.
3. Une temporisation de P secondes.
4. Retrait de l'axe Z en vitesse rapide jusqu'au plan de retrait R.

### 12.24.9 G83 : Cycle de perçage avec déburrage

Le cycle G83 est destiné au perçage profond ou au fraisage avec brise-copeaux. Les retraits, au cours de ce cycle, dégagent les copeaux du trou et fragmentent les copeaux longs (qui sont fréquents lors du perçage dans l'aluminium). Ce cycle utilise la valeur Q qui représente un incrément "delta" le long de l'axe Z. Programmer G83 X- Y- Z- A- B- C- R- L- Q-

donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un mouvement de l'axe Z seul, en vitesse travail, sur la position la moins profonde entre, un incrément delta, ou la position de Z programmée.

3. Un mouvement en vitesse rapide au plan de retrait.
4. Une plongée en vitesse rapide dans le même trou, presque jusqu'au fond.
5. Répétition des étapes 2, 3 et 4 jusqu'à ce que la position programmée de Z soit atteinte à l'étape 2.
6. Un mouvement de l'axe Z en vitesse rapide vers le plan de retrait.

C'est une erreur si :

- La valeur de Q est négative ou égale à zéro.

### **12.24.10 G84 : Cycle de taraudage à droite**

Ce code n'est pas encore implémenté dans EMC2. Il est accepté mais son comportement n'est pas défini. Voir G33.1

### **12.24.11 G85 : Cycle d'alésage, sans temporisation, retrait en vitesse travail**

Le cycle G85 est destiné à l'alésage, mais peut être utilisé pour le perçage ou le fraisage. Programmer G85 X- Y- Z- A- B- C- R- L- donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un déplacement de l'axe Z seul en vitesse travail, vers la position Z programmée.
3. Retrait de l'axe Z en vitesse travail vers le plan de retrait.

### **12.24.12 G86 : Cycle d'alésage, arrêt de broche, retrait en vitesse rapide**

Le cycle G86 est destiné à l'alésage. Ce cycle utilise la valeur P pour une temporisation en secondes. Programmer G86 X- Y- Z- A- B- C- R- L- P-

donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un déplacement de l'axe Z seul en vitesse travail, vers la position Z programmée.
3. Une temporisation de P secondes.
4. L'arrêt de rotation de la broche.
5. Retrait de l'axe Z en vitesse rapide vers le plan de retrait.
6. Reprise de la rotation de la broche dans la même direction que précédemment.

La broche doit tourner avant le lancement de ce cycle. C'est une erreur si :

- La broche ne tourne pas avant que ce cycle ne soit exécuté.

### **12.24.13 G87 : Alésage inverse**

Ce code n'est pas encore implémenté dans EMC2. Il est accepté mais son comportement n'est pas défini.

### **12.24.14 G88 : Alésage, arrêt de broche, retrait en manuel**

Ce code n'est pas encore implémenté dans EMC2. Il est accepté mais son comportement n'est pas défini.

### 12.24.15 G89 : Cycle d'alésage, temporisation, retrait en vitesse travail

Le cycle G89 est destiné à l'alésage. Il utilise la valeur de P pour une temporisation en secondes. Programmer G89 X- Y- Z- A- B- C- R- L- P-

donnera :

1. Un mouvement préliminaire. Comme décrit ici [12.24.6](#)
2. Un déplacement de l'axe Z seul en vitesse travail, vers la position Z programmée.
3. Temporisation de P secondes.
4. Retrait de l'axe Z en vitesse travail vers le plan de retrait.

## 12.25 G90, G91 : Modes de déplacement

L'interprétation du code RS274/NGC peut se faire dans deux modes de déplacements : absolu ou incrémental.

Pour se déplacer en mode absolu, programmer G90. En mode absolu, les valeurs d'axes X, Y, Z, A, B, C, U, V, W représentent les positions dans le système de coordonnées courant. Les exceptions à cette règle sont décrites dans la section [12.24](#).

Pour se déplacer en mode incrémental, programmer G91. En mode incrémental, les valeurs d'axes représentent un incrément depuis la position courante.

### 12.26 G90.1, G91.1 : Mode distance (I, J et K)

Note : Cette fonctionnalité est en cours de développement, elle n'est peut être pas totalement implémentée.

G90.1 Mode distance absolue pour les offsets I, J et K.

– I et J doivent tous les deux être spécifiés ou une erreur se produira.

G91.1 Mode distance incrémentale pour les offsets I, J et K.

– Renvoie I, J et K à leur fonctionnement normal.

### 12.27 G92, G92.1, G92.2, G92.3 : Décalage d'origine des systèmes de coordonnées

Voir la section [7.6](#) pour une vision générale des systèmes de coordonnées.

Pour donner au point actuel de nouvelles valeurs de coordonnées (sans faire de mouvement), programmer G92 X- Y- Z- A- B- C- U-, V-, W-, où les mots d'axes contiennent les valeurs souhaitées. Au moins un mot d'axe est obligatoire, les autres sont optionnels. Si il n'y a pas de mot d'axe pour un axe donné, les coordonnées de cet axe resteront inchangées. C'est une erreur si :

1. Tous les mots d'axes sont omis.

Quand G92 est exécuté, les origines de tout les systèmes de coordonnées sont déplacées. Elles sont déplacées de la même valeur que le point contrôlé courant l'est, dans le système de coordonnées courant, pour prendre les valeurs spécifiées dans la ligne du G92. Les origines de tous les systèmes de coordonnées sont décalées de la même valeur.

Un exemple : supposons que le point courant soit à  $X=4$  et qu'aucun décalage G92 ne soit actif. La ligne G92 X7 est programmée, toutes les origines seront décalées de -3 en X, ce qui fera que le point courant deviendra  $X=7$ . Ce -3 est sauvegardé dans le paramètre 5211.

Le fait d'être en mode de déplacement incrémental est sans effet sur l'action de G92.

Des décalages G92 peuvent déjà être actifs quand G92 est appelé. Si c'est le cas, ils sont remplacés par le nouveau décalage, de sorte que le point courant prenne la valeur spécifiée.

Pour repasser un décalage d'axe à zéro, programmer G92.1 ou G92.2. G92.1 positionne les paramètres 5211 à 5219 à zéro, tandis que G92.2 conserve leurs valeurs courantes inchangées.

Pour positionner des valeurs de décalage d'axes aux valeurs données dans les paramètres 5211 à 5219, programmer G92.3.

Vous pouvez positionner les décalages d'axes dans un programme et ré-utiliser les mêmes dans un autre programme. Programmer G92 dans le premier programme, ce qui positionnera les paramètres 5211 à 5219. Ne pas utiliser G92.1 dans la suite du premier programme. Les valeurs des paramètres seront enregistrées lors de la sortie du premier programme et rétablies au chargement du second programme. Utiliser G92.3 vers le début du deuxième programme, ce qui restaurera les décalages d'axes enregistrés par le premier.

EMC2 enregistre les décalages G92 et les réutilise au lancement du programme suivant. Il y a deux moyens de contourner cela, il est possible de programmer G92.1 pour effacer les décalages, ou de programmer G92.2 pour que les décalages enregistrés ne soient pas appliqués.

## 12.28 G93, G94, G95 : Choix des modes de vitesse

Trois modes de vitesse sont reconnus : unités par minute, inverse du temps et unités par tour. Programmer G94 pour passer en mode unités par minute. Programmer G93 pour passer en mode inverse du temps. Programmer G95 pour passer en mode unités par tour.

Dans le mode vitesse en unités par minute, le mot F est interprété pour indiquer que le point contrôlé doit se déplacer à un certain nombre de pouces par minute, de millimètres par minute, ou de degrés par minute, selon l'unité de longueur choisie pour les axes et quels types d'axes doivent se déplacer.

Dans le mode vitesse en unités par tour, le mot F est interprété pour indiquer que le point contrôlé doit se déplacer à un certain nombre de pouces par tour de broche, de millimètres par tour, selon l'unité de longueur utilisée et quels axes doivent être déplacés.

Dans le mode vitesse inverse du temps, le mot F signifie que le mouvement doit être terminé en [1 divisé par la valeur de F] minutes. Par exemple, si la valeur de F est 2.0, les mouvements doivent être terminés en 1/2 minute.

Quand le mode vitesse inverse du temps est actif, le mot F doit apparaître sur chaque ligne contenant un mouvement G1, G2, ou G3. Les mots F qui sont sur des lignes sans G1, G2, ou G3 sont ignorés. Être en mode vitesse inverse du temps est sans effet sur les mouvements G0 (vitesse rapide). C'est une erreur si :

- Le mode vitesse inverse du temps est actif et qu'une ligne avec G1, G2, ou G3 (explicitement ou implicitement) n'a pas de mot F.
- Une nouvelle vitesse n'a pas été spécifiée après un passage en G94 ou G95.

## 12.29 G96, G97 : Modes de contrôle de la broche

Deux modes de contrôle de la broche sont reconnus : tours par minute, et vitesse de coupe constante. Programmer G96 D- S- pour valider une vitesse de coupe constante de S pieds par minute si G20

est actif, ou mètres par minute si G21 est actif. La vitesse de rotation maximale est indiquée par la valeur de D- en tours par minute.

Programmer G97 pour activer le mode vitesse en tours par minute.

C'est une erreur si :

- S n'est pas spécifié avec G96.
- Une vitesse est spécifiée en mode G96 et la broche ne tourne pas.

### **12.30 G98, G99 : Options du plan de retrait**

Quand la broche se rétracte pendant les cycles préprogrammés, il existe deux options pour indiquer comment elle se rétracte : (1) Retrait perpendiculaire au plan de travail courant jusqu'à la position indiquée par le mot R, ou (2) Retrait perpendiculaire au plan de travail courant jusqu'à la position qui était celle de cet axe juste avant le début du cycle préprogrammé (à moins que cette position ne soit inférieure à celle indiquée par le mot R, auquel cas, c'est cette dernière qui serait utilisée.

Pour utiliser l'option (1), programmer G99. Pour utiliser l'option (2), programmer G98. Ne pas oublier que le mot R a différentes significations en mode de déplacement absolu et en mode de déplacement incrémental.

# Chapitre 13

## M Codes

### 13.1 M0, M1, M2, M30, M60 : Arrêts de programme

Pour stopper temporairement un programme en cours (quelle que soit la position du bouton d'arrêt facultatif), programmer M0.

Pour stopper temporairement un programme en cours (mais seulement si le bouton d'arrêt optionnel est activé), programmer M1.

Il est permis de programmer M0 et M1 en mode données manuelles (MDI), mais l'effet ne sera probablement pas perceptible, puisque le comportement normal en mode MDI est de s'arrêter, de toute façon, à la fin de chaque ligne.

Pour procéder à l'échange de porte-pièce avec le chargeur de pièces et stopper temporairement un programme en cours (quel que soit le réglage du bouton d'arrêt facultatif), programmer M60.

Si un programme est stoppé par M0, M1, ou M60, en appuyant sur le bouton de départ cycle, le programme reprend à la ligne suivante.

Pour finir un programme, programmer M2. Pour changer le porte-pièce du chargeur et finir un programme, programmer M30. Ces deux commandes produisent les effets suivants :

1. Les décalages d'axes sont mis à zéro (comme avec G92.2) et les décalages d'origine sont mis aux valeurs par défaut (comme avec G54).
2. Le plan de travail actif devient XY (comme avec G17).
3. Le mode de déplacement devient absolu (comme avec G90).
4. La vitesse travail passe en unités par minute (comme avec G94).
5. Les correcteurs de vitesse sont activés (comme avec M48).
6. Les compensations d'outil sont désactivées (comme avec G40).
7. La broche est arrêtée (comme avec M5).
8. Le mode mouvement courant devient G1 (comme avec G1).
9. L'arrosage est arrêté (comme avec M9).

Plus aucune ligne de code RS274/NGC ne sera exécutée après exécution de la commande M2 ou M30. Presser le départ cycle relance le programme au début du fichier.

### 13.2 M3, M4, M5 : Contrôle de broche

Pour démarrer la rotation de la broche en sens horaire à la vitesse programmée courante, programmer M3.

Pour démarrer la rotation de la broche en sens anti-horaire à la vitesse programmée courante, programmer M4.

Pour arrêter la rotation de la broche, programmer M5.

Il est permis d'utiliser M3 ou M4 si la vitesse de broche est à zéro. Si cela est fait (ou si le bouton de correction de vitesse est activé mais mis à zéro), la broche ne tournera pas, si, plus tard la vitesse de broche est augmentée (ou que le correcteur de vitesse est augmenté), la broche va se mettre en rotation. Il est permis d'utiliser M3 ou M4 quand la broche est déjà en rotation ou d'utiliser M5 quand la broche est déjà arrêtée.

### 13.3 M6 : Appel d'outil

Pour changer l'outil actuellement dans la broche par un autre, nouvellement sélectionné en utilisant le mot T, voir la section 15.3), programmer M6. Un changement d'outil complet donnera :

- La rotation de la broche est arrêtée.
- L'outil qui a été sélectionné (par le mot T sur la même ligne ou sur n'importe quelle ligne après le changement d'outil précédent), sera placé dans la broche. Le mot T est un nombre entier indiquant le numéro du slot d'outil dans le carrousel (non son index).
- Si l'outil sélectionné n'est pas déjà dans la broche avant le changement d'outil, l'outil qui était dans la broche (s'il y en avait un) va être remplacé dans son emplacement dans le chargeur.
- Les coordonnées des axes seront arrêtées dans les mêmes positions absolues qu'elles avaient avant le changement d'outil (mais la broche devra peut-être être réorientée).
- Aucune autre modification ne sera apportée. Par exemple, l'arrosage continue à couler durant le changement d'outil à moins qu'il ne soit arrêté par M9.

Le changement d'outil peut inclure des mouvements d'axes pendant son exécution. Il est permis, mais pas utile, de programmer un changement d'outil avec le même outil que celui qui est déjà dans la broche. Il est permis également, si il n'y a pas d'outil dans le slot sélectionné, dans ce cas, la broche sera vide après le changement d'outil. Si le slot zéro a été le dernier sélectionné, il n'y aura pas d'outil dans la broche après le changement.

### 13.4 M7, M8, M9 : Contrôle d'arrosage

Pour activer l'arrosage par brouillard (gouttelettes), programmer M7.

Pour activer l'arrosage fluide, programmer M8.

Pour arrêter tous les arrosages, programmer M9.

Il est toujours permis d'utiliser une de ces commandes, que les arrosages soient arrêtés ou non.

### 13.5 M48, M49 : Contrôle des correcteurs de vitesse

Pour autoriser les potentiomètres de corrections de vitesses de broche et celui de vitesse travail, programmer M48. Pour les inhiber tous les deux, programmer M49. Voir la section 7.3.1 pour plus de détails. Il est permis d'autoriser ou d'inhiber ces potentiomètres quand ils sont déjà autorisés ou inhibés. Ces potentiomètres peuvent aussi être activés individuellement en utilisant les commandes M50 et M51 comme décrit dans les sections 13.6 et 13.7.

## 13.6 M50 : Contrôle du correcteur de vitesse travail

Pour autoriser le potentiomètre de correction de vitesse travail, programmer M50 ou M50 P1. Pour inhiber ce potentiomètre, programmer M50 P0. Quand il est inhibé, le potentiomètre de correction de vitesse n'a plus aucune influence et les mouvements seront exécutés à la vitesse de travail programmée. (à moins que ne soit actif un correcteur de vitesse adaptative).

## 13.7 M51 : Contrôle du correcteur de vitesse broche

Pour autoriser le potentiomètre de correction de vitesse de la broche, programmer M51 ou M51 P1. Pour inhiber ce potentiomètre programmer M51 P0. Quand il est inhibé, le potentiomètre de correction de vitesse de broche n'a plus aucune influence et la broche tournera à la vitesse programmée, en utilisant le mot S comme décrit dans la section 15.2.

## 13.8 M52 : Contrôle de vitesse adaptative

Pour utiliser une vitesse adaptative, programmer M52 ou M52 P1. Pour stopper l'utilisation d'une vitesse adaptative, programmer M52 P0. Quand la vitesse adaptative est utilisée, certaines valeurs externes sont utilisées avec les correcteurs de vitesse de l'interface utilisateur et les vitesses programmées pour obtenir la vitesse travail. Dans EMC2, la HAL pin `motion.adaptive-feed` est utilisée dans ce but. Les valeurs de `motion.adaptive-feed` doivent être dans une échelle comprise entre 0 (pas de vitesse) et 1 (pleine vitesse).

## 13.9 M53 : Contrôle de coupure de vitesse

Pour autoriser le bouton de coupure de vitesse, programmer M53 ou M53 P1. Pour inhiber ce bouton programmer M53 P0. Autoriser la coupure de vitesse permet d'interrompre les mouvements par le biais d'une coupure de vitesse. Dans EMC2, la HAL pin `motion.feed-hold` est utilisée pour cette fonctionnalité. Une valeur de 1 provoque un arrêt des mouvements (si M53 est actif).

## 13.10 M61 : Ajuste le numéro de l'outil courant

Pour corriger le numéro de l'outil courant, en mode MDI ou après un changement manuel d'outil programmer M61 Qxx dans la fenêtre de données manuelles. Au démarrage d'EMC2 avec un outil dans la broche, il est possible ainsi d'ajuster le numéro de l'outil courant sans faire de changement d'outil.

C'est une erreur si :

- Qxx n'est pas égal ou supérieur à 0

## 13.11 M62 à M65 : Contrôle de sorties numériques

Pour contrôler un bit de sortie digitale, programmer M- P-, où le mot M doit être compris entre 62 et 65, et le mot P compris entre 0 et un maximum défini selon l'implémentation.

**M62** Activer la sortie digitale synchronisée avec un mouvement.

**M63** Désactiver la sortie digitale synchronisée avec un mouvement.

**M64** Activer immédiatement la sortie digitale.

**M65** Désactiver immédiatement la sortie digitale.

## 13.12 M66 : Contrôle d'entrée digitale

Pour contrôler un bit d'entrée digitale, programmer `M66 P- E- L- Q-` , où le mot P et le mot E doivent être compris entre 0 et un maximum défini selon l'implémentation. Un seul des mots P ou E doit être présent. C'est une erreur si ils sont présents tous les deux.

**M66** Attente d'une entrée

- Le mot P spécifie le numéro d'une entrée digitale.
- Le mot E spécifie le numéro d'une entrée analogique.
- Le mot L spécifie le type d'attente :

**0** - WAIT\_MODE\_IMMEDIATE - pas d'attente, retour immédiat. La valeur courante de l'entrée est stockée dans le paramètre #5399

**1** - WAIT\_MODE\_RISE - attente d'un front montant sur l'entrée.

**2** - WAIT\_MODE\_FALL - attente d'un front descendant sur l'entrée.

**3** - WAIT\_MODE\_HIGH - attente d'un état logique HIGH sur l'entrée.

**4** - WAIT\_MODE\_LOW - attente d'un état logique LOW sur l'entrée.

- Le mot Q spécifie le timeout pour l'attente. Si le timeout est dépassé, l'attente est interrompue et la variable #5399 positionnée à -1.
- Le type 0 est le seul autorisé pour une entrée analogique.

M66 attends un nouvel événement sur l'entrée ou l'arrêt de l'exécution du programme, jusqu'à ce que l'événement sélectionné (ou le timeout programmé) ne survienne. C'est une erreur de programmer une valeur de timeout à 0 dans tous les types, sauf le type 0.

C'est également une erreur de programmer M66 avec les deux mots, un mot P et un mot E (ce qui reviendrait à sélectionner à la fois une entrée analogique et une digitale).

## 13.13 M100 à M199 : Commandes définies par l'utilisateur

Pour invoquer une commande définie par l'utilisateur, programmer `M1xx P- Q-` où P et Q sont facultatifs. Le programme externe " M1nn", qui doit se trouver dans le répertoire pointé par la variable [DISPLAY] PROGRAM\_PREFIX du fichier ini, est exécuté avec les valeurs P et Q comme étant ses deux arguments. L'exécution du fichier G-code courant passe en pause jusqu'à ce que le programme invoqué soit terminé.

Le message d'erreur « M code inconnu » signifie que :

- La commande spécifiée n'existe pas.
- Le fichier n'est pas exécutable.

Exemple : dans un programme g-code, on doit ouvrir et fermer un mandrin automatique via une broche du port parallèle, on appellera respectivement M101 pour ouvrir le mandrin et M102 pour le fermer. Les deux scripts bash correspondants, appelés M101 et M102 seront créés avant le lancement d'EMC2 puis rendus exécutables, par exemple par un click droit puis propriétés -> permissions -> Exécution.

M101

```
#!/bin/bash
# ce fichier met la broche 14 du port // à 1 pour ouvrir le mandrin automatique
halcmd setp parport.0.pin-14-out True
exit 0
```

M102

```
#!/bin/bash  
# ce fichier met la broche 14 du port // à 0 pour fermer le mandrin automatique  
halcmd setp parport.0.pin-14-out False  
exit 0
```

# Chapitre 14

## O Codes

Les O-codes permettent le contrôle de flux dans les programmes NC. Chaque block est associé à une adresse, qui est la valeur utilisée après le O. Il faut prendre soin de bien faire correspondre les adresses des codes O.

Le comportement est indéfini si :

- D'autres mots sont utilisés sur une ligne contenant un mot O.
- Un commentaire est utilisé sur une ligne contenant un code O.

### 14.1 Sous-programmes : « **sub** », « **endsub** », « **return** », « **call** »

Les sous-programmes s'étendent d'un O- **sub** à un O- **endsub**. Les lignes, à l'intérieur du sous-programme (le corps du sous-programme), ne sont pas exécutées dans l'ordre, mais elles sont exécutées à chaque fois que le sous-programme est appelé avec un O- **call**.

```
O100 sub (sous-programme pour aller à l'origine programme)
GO X0 Y0 Z0
O100 endsub
... (plusieurs lignes)
O100 call
```

À l'intérieur d'un sous-programme, O- **return** peut être exécuté, pour retourner immédiatement au code appelant, comme si O- **endsub** avait été rencontré.

O- **call** peut prendre jusqu'à 30 arguments optionnels, qui sont passés au sous-programme comme #1, #2, ..., #N. Les paramètres de #N+1 à #30 ont la même valeur dans le contexte de l'appel. Au retour du sous-programme, les valeurs des paramètres #1 jusqu'à #30 (quel que soit le nombre d'arguments) sont restaurés aux valeurs qu'ils avaient avant l'appel.

Parce que " 1 2 3" est analysé comme le nombre 123, les paramètres doivent être placés entre crochets. L'appel de sous-programme suivant, s'effectue avec 3 arguments :

```
O200 call [1] [2] [3]
```

Les corps de sous-programme ne peuvent pas être imbriqués. Ils ne peuvent être appelés qu'après avoir été définis. Ils peuvent être appelés depuis d'autres fonctions et peuvent s'appeler eux même récursivement, s'il est judicieux de le faire. Le niveau maximum d'imbrication des sous-programmes est de 10.

Les sous-programmes n'ont pas de "valeur de retour", mais ils peuvent changer la valeur des paramètres au dessus de #30 et ces changements sont visibles depuis le code appelant. Les sous-programmes peuvent aussi changer la valeur des paramètres nommés globaux.

## 14.2 Boucles : « do », « while », « endwhile », « break », « continue »

La boucle **while** a deux structures possibles : **while/endwhile** et **do/while**. Dans chaque cas, la boucle est quittée quand la condition du “while” devient fausse.

```
(dessine la forme d'une dent de scie)
F100
#1 = 0
O101 while [#1 lt 10]
G1 X0
G1 Y[#1/10] X1
#1 = [#1+1]
O101 endwhile
```

À l'intérieur d'une boucle **while**, **O- break**, quitte immédiatement la boucle et **O- continue**, saute immédiatement à la prochaine évaluation de la condition du **while**. Si elle est vraie, la boucle recommence au début. Si elle est fausse, la boucle est quittée.

## 14.3 Conditionnel : « if », « else », « endif »

Le **if** conditionnel exécute un groupe d'instructions si sa condition est vraie et un autre groupe si elle est fausse.

```
(Ajuste la vitesse travail en fonction d'une variable)
O102 if [#2 GT 5]
F100
O102 else
F200
O102 endif
```

## 14.4 Répétition : « Repeat »

La répétition « **repeat** », exécutera les blocs contenus entre « **repeat** » et « **endrepeat** » le nombre de fois spécifié entre crochets. L'exemple suivant montre comment usiner une série de 5 formes diagonales commençant à la position courante.

```
(Usine 5 formes diagonales)
G91 (Mode incrémental)
O103 repeat [5]
... (insérer le code à usiner ici)
G0 X1 Y1 (mouvement en diagonale à la position suivante)
O103 endrepeat
G90 (Mode absolu)
```

## 14.5 Indirection

L'adresse de **O-** peut être donnée par un paramètre ou un calcul.

```
O[#101+2] call
```

## 14.6 Calcul des valeurs dans les mots O

Dans les mots O-, les paramètres (section 8.3.2), les expressions (section 8.3.4), les opérateurs binaires (section 8.3.5) et les fonctions (tableau 8.3), sont particulièrement intéressants.

## 14.7 Appel de fichiers

Pour appeler un fichier par son nom, celui-ci doit contenir un « sub » et un « endsub ». Le fichier appelé doit se trouver dans le répertoire pointé par la variable PROGRAM\_PREFIX du fichier ini.

```
o<monfichier> call (fichier nommé)
```

ou

```
o123 call (fichier numéroté)
```

Dans le fichier appelé doit se trouver le « sub » et le « endsub » correspondant à l'appel. Le fichier doit être un fichier valide.

Exemple :

```
monfichier.ngc
o<monfichier> sub
...
o<monfichier> endsub
M2
```

# Chapitre 15

## Autres Codes

### 15.1 F : Réglage de la vitesse travail

Pour régler la vitesse d'avance, programmer F-. L'application de la vitesse est telle que décrite dans la section 7.2.5, à moins que le mode vitesse inverse du temps ne soit actif, dans ce cas, la vitesse est telle que décrite dans la section 12.28.

### 15.2 S : Réglage de la vitesse de rotation de la broche

Pour régler la vitesse en tours par minute (tr/mn) de la broche, programmer S-. La broche va tourner à cette vitesse quand elle sera programmée pour tourner. Il est permis de programmer un mot S que la broche tourne ou non. Si le potentiomètre de correction de vitesse broche est autorisé et n'est pas positionné sur 100%, la vitesse de broche sera différente de celle programmée. Il est permis de programmer S0, la broche ne tournera pas. C'est une erreur si :

- La valeur de S est négative.

Comme décrit dans la section 12.24.10, si un cycle préprogrammé G84 (taraudage) est actif et que les potentiomètres de vitesse et d'avance sont autorisés, celui qui a le réglage le plus bas sera utilisé. La vitesse de rotation et d'avance resteront synchronisées. Dans ce cas, la vitesse peut différer de celle programmée, même si le potentiomètre de correction de vitesse travail est sur 100%.

### 15.3 T : Choix de l'outil

Pour sélectionner un outil, programmer T-, où la valeur de T correspond au numéro du slot d'outil dans le carrousel. L'outil ne sera appelé et changé que quand un M6 sera programmé (voir la section 13.3). Le mot T peut apparaître sur la même ligne que le M6 ou sur une ligne précédente. Il est permis, mais normalement inutile, qu'un mot T apparaisse à plus de deux lignes avant, sans changement d'outil. Le carrousel peut bouger, seulement le plus récent mot T ne prendra effet qu'au prochain changement d'outil. Il est permis de programmer T0, aucun outil ne sera sélectionné. C'est utile pour avoir la broche vide. C'est une erreur si :

- Une valeur négative de T est utilisée.
- Une valeur de T supérieure au nombre de slot d'outils dans le carrousel est utilisée.

Sur certaines machines, le carrousel se déplace lorsque le mot T est programmé, avec l'usinage en cours. Sur ces machines, programmer T plusieurs lignes de texte avant le changement d'outil permet de gagner du temps. Une pratique de programmation courante pour ces types de machines, consiste à placer le mot T pour le prochain outil sur la ligne suivant le changement d'outil. Cela laisse au carrousel tout le temps pour se positionner.

## Chapitre 16

# Différences entre le g-code d'EMC2 et le RS274NGC

### 16.1 Différences changeant la signification d'un programme correctement écrit en RS274NGC

#### 16.1.1 Position après un changement d'outil

Dans EMC2, la machine ne revient pas à sa position initiale après un changement d'outil. Ce choix a été fait parce que le nouvel outil pourrait être plus long que l'ancien, rendant le retour de la machine à sa position initiale, impossible sans heurter la pièce.

#### 16.1.2 Les décalages sont dans les unités du fichier ini

Dans EMC2, les valeurs stockées dans les paramètres des points d'origine G28 et G30, des systèmes de coordonnées P1...P9 et du décalage G92 sont dans les unités définies dans le fichier ini. Cette différence existe parce que, sinon, la signification d'un emplacement changeait selon que G20 ou G21 était actif lorsque G28, G30, G10, L2, ou G92.3 était programmé.

#### 16.1.3 Les longueurs/diamètres de la table d'outils sont dans les unités du fichier ini

Dans EMC2, les longueurs (compensation) et diamètres de la table d'outils sont seulement dans les unités du fichier ini. Cette différence existe parce que autrement, la longueur d'un outil et son diamètre aurait changé selon que G20 ou G21 aurait été actif dans les modes G43, G41, G42. Ce qui aurait rendu impossible, sans modification de la table d'outils, l'exécution de g-code avec la machine en unités non natives, même pour un g-code bien écrit (ligne en début avec G20 ou G21 et mêmes unités tout au long du programme).

#### 16.1.4 G84, G87 non implémentés

G84 et G87 ne sont pas actuellement implémentés, mais seront peut être ajoutés à de futures versions d'EMC2.

### 16.1.5 G28, G30 avec des mots d'axes

Quand G28 ou G30 est programmé avec un mot d'axe présent, EMC2 déplace cet axe seul. Il est commun sur les autres contrôles machine. Pour déplacer certains axes vers un point intermédiaire puis les déplacer tous vers un point pré-défini, écrire deux lignes de gcode :

```
G0 X- Y- (Les axes se déplacent vers le point intermédiaire)
G28 (Les axes se déplacent vers leurs origines)
```

### 16.1.6 M62, M63 non implémentés

M62 et M63 ne sont pas encore implémentés, mais seront peut être ajoutés à de futures versions d'EMC2.

### 16.1.7 G0 réduit les longs mouvements angulaires pour les rendre inférieurs à un tour

Comme décrit précédemment, G0 réduit les mouvements angulaires à moins d'un tour. Ce qui peut produire des trajectoires différentes de celles produites avec RS274NGC.

## 16.2 Différences ne changeant pas la signification des programmes bien écrits en RS274NGC

### 16.2.1 G33, G76 codes de taraudage

Ces codes ne sont pas définis dans RS274NGC.

### 16.2.2 G38.2

La pointe du palpeur n'est pas rétractée après un mouvement G38.2. Ce mouvement de retrait sera peut être ajouté dans de futures versions d'EMC2.

### 16.2.3 G38.3...G38.5

Ces codes ne sont pas définis dans RS274NGC

### 16.2.4 O-codes

Ces codes ne sont pas définis dans RS274NGC

### 16.2.5 M50...M53 survitesses

Ces codes ne sont pas définis dans RS274NGC

### 16.2.6 M61...M66

Ces codes ne sont pas définis dans RS274NGC

## **16.2.7 G43, G43.1**

### **16.2.7.1 Longueurs d'outil négatives**

Les spécifications du RS274NGC disent "il est prévu que" toutes les longueurs d'outils soient positives. Toutefois, G43 fonctionne aussi avec des longueurs d'outils négatives.

### **16.2.7.2 Outils de tour**

La compensation de longueur d'outil G43 peut décaler l'outil dans les deux dimensions X et Z. Cette fonctionnalité est très utile sur les tours.

### **16.2.7.3 Longueurs d'outil dynamiques**

EMC2 permet la spécification d'une longueur d'outil calculée par `G43.1 I K`.

## **16.2.8 G41.1, G42.1**

EMC2 permet la spécification dans le gcode d'un diamètre d'outil et sur un tour, de son orientation. Le format est `G41.1/G42.1 D- L-`, où D est le diamètre et L (si spécifié) est l'orientation de l'outil de tour.

## **16.2.9 G43 sans mot H**

Dans `ngc`, cela n'est pas permis. Dans EMC2, il règle la cote de longueur du décalage pour l'outil courant. Si aucun outil n'est chargé actuellement, c'est une erreur. Cette différence a été apportée pour que l'utilisateur n'ait pas à spécifier le numéro d'outil à deux endroits pour chaque changement d'outil et parce que c'est plus cohérent avec la manière de travailler de G41/G42 quand le mot D n'est pas spécifié.

## **16.2.10 Axes U, V et W**

EMC2 gère des machines avec un maximum de 9 axes en définissant un jeu supplémentaire de 3 axes linéaires nommés U, V et W.

# Chapitre 17

## Fichier d'outils et compensations

### 17.1 Fichier d'outils

Les longueurs et diamètres d'outils peuvent être lus dans un fichier d'outils (voir la section 7.4) ou provenir d'un mot spécifié pour activer la compensation d'outil.

### 17.2 Compensation d'outil

La compensation d'outil peut causer beaucoup de problèmes aux meilleurs programmeurs de code nc. Mais elle peut aussi être une aide puissante quand elle est utilisée pour aider un opérateur à obtenir une pièce à la cote. En réglant la longueur et le diamètre des outils dans une table d'outils, les décalages peuvent être utilisés pendant un cycle d'usinage qui tient compte des variations de taille de l'outil, ou pour des déviations mineures des trajectoires programmées. Et ces changements peuvent être faits sans que l'opérateur n'ait à changer quoi que ce soit dans le programme.

Tout au long de ce module, vous trouverez occasionnellement des références à des fonctions canoniques, là où il est nécessaire pour le lecteur de comprendre comment fonctionne une compensation d'outil dans une situation spécifique. Ces références ont pour but de donner au lecteur une idée de la séquences plutôt que d'exiger qu'il comprenne la façon dont les fonctions canoniques elles-mêmes fonctionnent dans EMC.

### 17.3 Compensation de longueur d'outil

Les compensations de longueur d'outil sont données comme des nombres positifs dans la table d'outils. Une compensation d'outil est programmée en utilisant G43 Hn, où n est le numéro d'index de l'outil souhaité dans la table d'outil. Il est prévu que toutes les entrées dans la table d'outils soit positives. La valeur de H est vérifiée, elle doit être un entier non négatif quand elle est lue. L'interpréteur se comporte comme suit :

1. Si G43 Hn est programmé, un appel à la fonction `USE_TOOL_LENGTH_OFFSET(longueur)` est fait (où longueur est l'écart de longueur, lu dans la table d'outils, de l'outil indexé n), `tool_length_offset` est repositionné dans le modèle de réglages de la machine et la valeur de `current_z` dans le modèle est ajustée. Noter que n n'a pas besoin d'être le même que le numéro de slot de l'outil actuellement dans la broche.

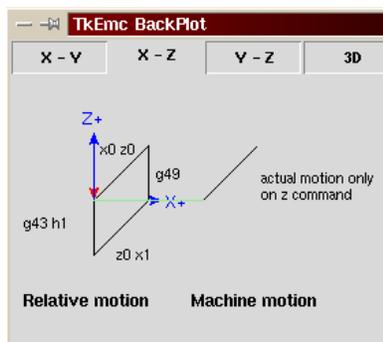
2. Si G49 est programmé, `USE_TOOL_LENGTH_OFFSET(0.0)` est appelée, `tool_length_offset` est remis à 0.0 dans le modèle de réglages de la machine et la valeur courante de `current_z` dans le modèle est ajustée. L'effet de la compensation de longueur d'outil est illustrée dans la capture ci-dessous.

Noter que la longueur de l'outil est soustraite de Z pour que le point contrôlé programmé corresponde à la pointe de l'outil. Il faut également noter que l'effet de la compensation de longueur est immédiat quand on voit la position de Z comme une coordonnée relative mais il est sans effet sur la position actuelle de la machine jusqu'à ce qu'un mouvement en Z soit programmé.

Programme de test de longueur d'outil.

Tool #1 is 1 inch long.

```
N01 G1 F15 X0 Y0 Z0
N02 G43 H1 Z0 X1
N03 G49 X0 Z0
N04 G0 X2
N05 G1 G43 H1 G4 P10 Z0 X3
N06 G49 X2 Z0
N07 G0 X0
```



Avec ce programme, dans la plupart des cas, la machine va appliquer le décalage sous forme d'une rampe pendant le mouvement en xyz suivant le mot G43.

## 17.4 Compensation de rayon d'outil

La compensation de rayon d'outil (également appelée compensation de diamètre d'outil) a été ajoutée aux spécifications RS-274D à la demande d'utilisateurs, car elle est extrêmement utile, mais son implémentation a été assez mal pensée. L'objectif de cette fonctionnalité est de permettre aux programmeurs de "virtualiser" la trajectoire de l'outil, de sorte que la machine puisse pendant toute l'exécution, déterminer le bon décalage à apporter à la position de l'outil pour respecter les cotes, en s'appuyant sur les diamètres d'outils existants. Si un outil est réaffûté, son diamètre sera légèrement plus petit que celui d'origine, il faudra également en tenir compte.

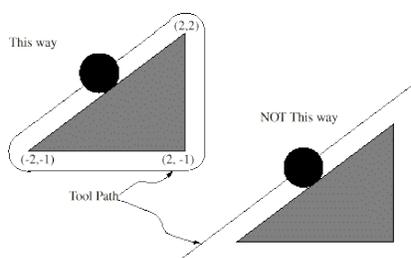
Le problème est pour donner à la machine la trajectoire exacte où l'outil doit usiner, sur le côté intérieur d'un parcours imaginaire, ou sur le côté extérieur. Comme ces trajectoires ne sont pas nécessairement fermées (même si elles peuvent l'être), il est quasiment impossible à la machine de connaître de quel côté du profil elle doit compenser l'outil. Il a été décidé qu'il n'y aurait que deux choix possibles, outil à "gauche" du profil à usiner et outil à "droite" du profil à usiner. Ce qui doit être interprété à gauche ou à droite du profil à usiner en suivant l'outil le long du profil.

### 17.4.1 Compensation de rayon d'outil, détails

Les possibilités de compensation de rayon d'outil de l'interpréteur, autorise le programmeur à spécifier si l'outil doit passer à gauche ou à droite du profil à usiner. Ce profil peut être un contour ouvert ou fermé, dans le plan XY composé de segments en arcs de cercles et en lignes droites. Le

contour peut être le pourtour d'une pièce brute ou, il peut être une trajectoire exacte suivie par un outil mesuré avec précision. La figure ci-dessous, montre deux exemples de trajectoires d'usinage d'une pièce triangulaire, utilisant la compensation de rayon d'outil.

Dans les deux exemples, le triangle gris représente le matériau restant après usinage et la ligne extérieure représente le parcours suivi par le centre de l'outil. Dans les deux cas le triangle gris est conservé. Le parcours de gauche (avec les coins arrondis) est le parcours généralement interprété. Dans la méthode de droite (celle marquée Not this way), l'outil ne reste pas en contact avec les angles vifs du triangle gris.



Des mouvements sur l'axe Z peuvent avoir lieu pendant que le contour est suivi dans le plan XY. Des portions du contour peuvent être franchies avec l'axe Z en retrait au dessus de la pièce pendant la poursuite du parcours et jusqu'au point où l'usinage doit reprendre, l'axe Z plongera de nouveau en position. Ces dégagements de zones non usinées peuvent être faits en vitesse travail (G1), en rapide (G0), en vitesse inverse du temps (G93) ou en avance en unités par minute (G94) toutes peuvent être utilisées avec la compensation de rayon d'outil. Sous G94, la vitesse sera appliquée à la pointe de l'outil coupant, non au contour programmé.

### Instructions de programmation

- Pour activer la compensation de rayon d'outil, programmer soit, G41 (pour maintenir l'outil à gauche du profil à usiner) soit, G42 (pour maintenir l'outil à droite du profil). Dans la figure 17.4.1, par exemple, si G41 était programmé, l'outil devrait tourner en sens horaire autour du triangle et dans le sens contraire si G42 était programmé.
- Pour désactiver la compensation de rayon d'outil, programmer G40.
- Si un G40, G41, ou G42 est programmé dans la même ligne qu'un mouvement d'axe, la compensation de rayon sera activée ou désactivée avant que le mouvement ne soit fait. Pour que le mouvement s'effectue en premier, il doit être programmé séparément et avant.

### La valeur de D

L'interpréteur actuel requiert une valeur D sur chaque ligne contenant un mot G41 ou G42. Le nombre D doit être un entier positif. Il représente le numéro de slot de l'outil, dont le rayon (la moitié du diamètre d'outil indiqué dans la table d'outils) sera compensé. Il peut aussi être égal à zéro, dans ce cas, la valeur du rayon sera aussi égale à zéro. Tous les slots de la table d'outils peuvent être sélectionnés de cette façon. Le nombre D n'est pas nécessairement le même que le numéro de slot de l'outil monté dans la broche.

### Table d'outils

La compensation de rayon d'outil utilise les données fournies par la table d'outils de la machine. Pour chaque slot d'outil dans le carrousel, la table d'outils contient le diamètre de l'outil rangé à cet emplacement (ou la différence entre le diamètre nominal de l'outil placé dans ce slot et son diamètre actuel). La table d'outils est indexée par les numéros de slot. Reportez vous à la page des "Fichiers d'outils" pour savoir comment remplir ces fichiers.

### Deux types de contour

L'interpréteur contrôle la compensation pour deux types de contour :

- 1) Le contour donné dans le code NC correspond au bord extérieur du matériau après usinage. Nous l'appellerons "contour sur le profil du matériau".
- 2) Le contour donné dans le code NC correspond à la trajectoire suivie par le centre d'un outil de rayon nominal. Nous l'appellerons "contour sur le parcours d'outil".

L'interpréteur ne dispose d'aucun paramètre pour déterminer quel type de contour est utilisé, mais la description des contours est différente (pour la même géométrie de pièce) entre les deux types, les valeurs des diamètres dans la table d'outils seront également différentes pour les deux types.

### Contour sur le profil du matériau

Lorsque le contour est sur le profil du matériau, c'est ce tracé qui est décrit dans le programme NC. Pour un contour sur le profil du matériau, la valeur du diamètre dans la table d'outils correspond au diamètre réel de l'outil courant. Cette valeur dans la table doit être positive. Le code NC pour ce type de contour reste toujours le même à l'exception du diamètre de l'outil (actuel ou nominal).

Exemple 1 :

Voici un programme NC qui usine le matériau en suivant le profil d'un des triangles de la figure précédente. Dans cet exemple, la compensation de rayon est celle du rayon actuel de l'outil, soit 0.5". La valeur pour le diamètre dans la table d'outil est de 1.0" .

```
N0010 G41 G1 X2 Y2 (active la compensation et fait le mouvement d'entrée)
```

```
N0020 Y-1 (suit la face droite du triangle)
```

```
N0030 X-2 (suit la base du triangle)
```

```
N0040 X2 Y2 (suit l'hypoténuse du triangle)
```

```
N0050 G40 (désactive la compensation)
```

Avec ce programme, l'outil suit cette trajectoire : un mouvement d'entrée, puis la trajectoire montrée dans la partie gauche de la figure, avec un déplacement de l'outil en sens horaire autour du triangle. Noter que les coordonnées du triangle de matériau apparaissent dans le code NC. Noter aussi que la trajectoire inclus trois arcs qui ne sont pas explicitement programmés, ils sont générés automatiquement.

### Contour sur le parcours d'outil

Lorsque le contour est sur le parcours d'outil, la trajectoire décrite dans le programme correspond au parcours que devra suivre le centre de l'outil. Le bord de l'outil, à un rayon de là, (excepté durant les mouvements d'entrée) suivra la géométrie de la pièce. La trajectoire peut être créée manuellement ou par un post-processeur, selon la pièce qui doit être réalisée. Pour l'interpréteur, le parcours d'outil doit être tel que le bord de l'outil reste en contact avec la géométrie de la pièce, comme montré à gauche de la figure 7. Si une trajectoire du genre de celle montrée sur la droite de la figure 7 est utilisée, dans laquelle l'outil ne reste pas en permanence au contact avec la géométrie

de la pièce, l'interpréteur ne pourra pas compenser correctement si un outil en dessous de son diamètre nominal est utilisé.

Pour un contour sur le parcours d'outil, la valeur pour le diamètre de l'outil dans la table d'outils devra être un petit nombre positif si l'outil sélectionné est légèrement surdimensionné. La valeur du diamètre sera un petit nombre négatif si l'outil est légèrement sous-dimensionné. Si un diamètre d'outil est négatif, l'interpréteur compense de l'autre côté du contour programmé et utilise la valeur absolue donnée au diamètre. Si l'outil courant est à son diamètre nominal, la valeur dans la table d'outil doit être à zéro.

Exemple de contour sur le parcours d'outil

Supposons que le diamètre de l'outil courant monté dans la broche est de 0.97 et le diamètre utilisé en générant le parcours d'outil a été de 1.0 . Alors la valeur de diamètre dans la table d'outils pour cet outil est de -0.03. Voici un programme NC qui va usiner l'extérieur d'un triangle de la figure 7.

```
N0010 G1 X1 Y4.5 (mouvement d'alignement)
N0020 G41 G1 Y3.5 (active la compensation et premier mouvement d'entrée)
N0030 G3 X2 Y2.5 I1 (deuxième mouvement d'entrée)
N0040 G2 X2.5 Y2 J-0.5 (usinage le long de l'arc en haut du parcours d'outil)
N0050 G1 Y-1 (usinage le long du côté droit du parcours d'outil)
N0060 G2 X2 Y-1.5 I-0.5 (usinage de l'arc en bas à droite)
N0070 G1 X-2 (usinage de la base du parcours d'outil)
N0080 G2 X-2.3 Y-0.6 J0.5 (usinage de l'arc en bas à gauche)
N0090 G1 X1.7 Y2.4 (usinage de l'hypoténuse)
N0100 G2 X2 Y2.5 I0.3 J-0.4 (usinage de l'arc en haut à droite)
N0110 G40 (désactive la compensation)
```

Avec ce programme, l'outil suit cette trajectoire : un mouvement d'alignement, deux mouvements d'entrée, puis il suit une trajectoire légèrement intérieure au parcours d'outil montré sur la figure 7 en tournant en sens horaire autour de la pièce. Cette compensation est à droite de la trajectoire programmée, même si c'est G41 qui est programmé, en raison de la valeur négative du diamètre.

### Erreurs de programmation et limitations

Les messages en rapport avec la compensation de rayon d'outil, délivrés par l'interpréteur sont les suivants :

1. Impossible de changer les décalages d'axes avec la compensation de rayon d'outil
2. Impossible de changer d'unité avec la compensation de rayon d'outil
3. Impossible d'activer la compensation de rayon d'outil en dehors du plan XY
4. Action impossible, la compensation de rayon d'outil est déjà active
5. Impossible d'utiliser G28 ou G30 avec la compensation de rayon d'outil
6. Impossible d'utiliser G53 avec la compensation de rayon d'outil
7. Impossible d'utiliser le plan XZ avec la compensation de rayon d'outil
8. Impossible d'utiliser le plan YZ avec la compensation de rayon d'outil
9. Coin concave avec la compensation de rayon d'outil
10. Interférence de l'outil avec une partie finie de la pièce avec la compensation de rayon d'outil<sup>1</sup>
11. Mot D sur une ligne sans mot de commande G41 ni G42
12. Index de rayon d'outil trop grand
13. Le rayon de l'outil n'est pas inférieur au rayon de l'arc avec la compensation de rayon
14. Deux G-codes du même groupe modal sont utilisés.

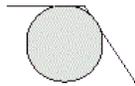
<sup>1</sup>Le terme anglais "gouging" indique une interférence entre l'outil et une partie finie de la pièce ou la paroi d'une cavité. Par extension, le terme est parfois repris pour une interférence entre le porte-outil ou la broche et la pièce.

Pour certains de ces messages, des explications sont données plus loin.

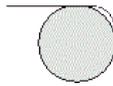
Changer d'outil alors que la compensation de rayon d'outil est active n'est pas considéré comme une erreur, mais il est peu probable que cela soit fait intentionnellement. Le rayon d'outil utilisé lors de l'établissement de la compensation continuera à être utilisé jusqu'à la désactivation de la compensation, même si un nouvel outil est effectivement utilisé.

Quand la compensation de rayon d'outil est active, il est physiquement possible de faire un cercle, dont le rayon est la moitié du diamètre de l'outil donné dans la table d'outils, il sera tangent avec l'outil en tout point de son contour.

concave corner - tool does not fit



concave arc too small - tool does not fit



En particulier, l'interpréteur traite les coins concaves et les arcs concaves plus petits que l'outil, comme des erreurs, le cercle ne peut pas être maintenu tangent avec le contour dans ces situations. Cette détection de défaut, ne limite pas les formes qui peuvent être usinées, mais elle requiert que le programmeur précise la forme exacte à usiner (ou le parcours d'outil qui doit être suivi) et non une approximation. A cet égard, l'interpréteur utilisé par EMC diffère des interpréteurs utilisés dans beaucoup d'autres contrôleurs, qui passent ces erreurs sous silence et laissent l'outil interférer avec la partie finie de la pièce (gouging) ou arrondissent des angles qui devraient être vifs. Il n'est pas nécessaire, de déplacer l'outil entre la désactivation de la compensation et sa réactivation, mais le premier mouvement suivant la réactivation sera considéré comme premier mouvement, comme déjà décrit plus tôt.

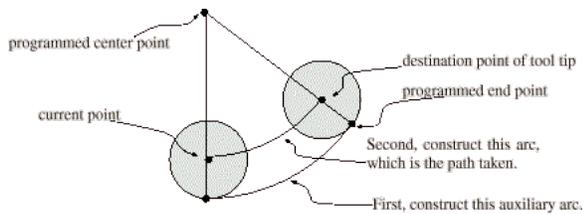
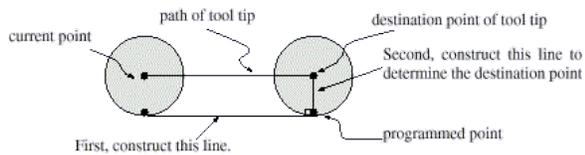
Il n'est pas possible de passer d'un index de rayon d'outil à un autre alors que la compensation est active. Il est également impossible de basculer la compensation d'un côté à l'autre avec la compensation active. Si le prochain point de destination XY est déjà dans le périmètre d'action de l'outil quand la compensation est activée, le message indiquant une interférence outil/surface finie, s'affichera quand la ligne du programme qui donne cette destination sera atteinte. Dans ce cas, l'outil a déjà usiné dans le matériau, là où il n'aurait pas dû...

Si le numéro de slot programmé par le mot D est supérieur au nombre d'emplacements disponibles dans le carrousel, un message d'erreur sera affiché. Dans l'implémentation actuelle, le nombre d'emplacements maximum est de 68.

Le message d'erreur "Deux G-codes du même groupe modal sont utilisés" est un message générique utilisé pour plusieurs jeux de G-codes. Il s'applique à la compensation de rayon d'outil, il signifie que plus d'un code G40, G41 ou G42 apparaît sur la même ligne de programme NC, ce qui n'est pas permis.

## Premier mouvement

L'algorithme utilisé lors du premier déplacement, quand c'est une ligne droite, consiste à tracer une droite, depuis le point d'arrivée, tangente à un cercle dont le centre est le point actuel, et le rayon, celui de l'outil. Le point de destination de la pointe de l'outil se trouve alors au centre d'un cercle de même rayon, tangent à la ligne droite tracée précédemment. C'est montré sur la figure 9. Si le point programmé est situé à l'intérieur de la première section d'outil (le cercle de gauche), une erreur sera signalée.



Si le premier mouvement après que la compensation de rayon d'outil a été activée est un arc, l'arc qui sera généré est dérivé d'un arc auxiliaire, qui a son centre identique à celui du point central programmé, passe par le point final de l'arc programmé et, est tangent à l'outil à son emplacement courant. Si l'arc auxiliaire ne peut pas être construit, une erreur sera signalée. L'arc généré déplacera l'outil pour qu'il reste tangent à l'arc auxiliaire pendant tout le mouvement. C'est ce que montre sur la figure 10.

Indépendamment du fait que le premier déplacement est une droite ou un arc, l'axe Z peut aussi se déplacer en même temps. Il se déplacera linéairement, comme c'est le cas quand la compensation de rayon n'est pas utilisée. Les mouvements des axes rotatifs (A, B et C) sont autorisés avec la compensation de rayon d'outil, mais leur utilisation serait vraiment très inhabituelle.

Après les mouvements d'entrée en compensation de rayon d'outil, l'interpréteur maintiendra l'outil tangent au contour programmé et du côté approprié. Si un angle aigu se trouve dans le parcours, un arc est inséré pour tourner autour de l'angle. Le rayon de cet arc sera de la moitié du diamètre de l'outil donné dans la table d'outils.

Quand la compensation de rayon est désactivée, aucun mouvement de sortie particulier n'est fait. Le mouvement suivant sera ce qu'il aurait été si la compensation n'avait jamais été activée et que le mouvement précédent ait placé l'outil à sa position actuelle.

### Programmation des mouvements d'entrée

En général, un mouvement d'alignement et deux mouvements d'entrée sont demandés pour commencer la compensation correctement. Cependant, si le contour programmé comporte des pointes et des angles aigus, un seul mouvement d'entrée (plus, éventuellement, un mouvement de pré-entrée) est demandé. La méthode générale, qui fonctionne dans toutes les situations, est décrite en premier. Elle suppose que le programmeur connaît déjà le contour et son but est d'ajouter le mouvement d'entrée.

#### Méthode générale

La méthode générale de programmation comprend un mouvement d'alignement et deux mouvements d'entrée. Les mouvements d'entrée expliqués ci-dessus, seront repris comme exemple. Voici le code correspondant :

```
N0010 G1 X1 Y4.5 (mouvement d'alignement vers le point C)
N0020 G41 G1 Y3.5 (active la compensation et fait le premier mouvement d'entrée vers le point B)
N0030 G3 X2 Y2.5 I1 (fait le second mouvement d'entrée vers le point A)
```

Voir la figure 11. La figure montre les deux mouvements d'entrée mais pas le mouvement d'alignement.

En premier, choisir un point A sur le contour où il convient d'attacher un arc d'entrée. Spécifier un arc à l'extérieur du contour qui commence au point B et s'achève au point A, tangent au contour (et aller dans la même direction que celle prévue pour tourner autour du contour). Le rayon doit être supérieur à la moitié du diamètre donné dans la table d'outils. Ensuite, tirer une ligne tangente à l'arc, du point B au point C, placé de telle sorte que la ligne BC fasse plus d'un rayon de long.

Après que la construction soit terminée, le code est écrit dans l'ordre inverse de celui de la construction. La compensation de rayon d'outil est activée après le mouvement d'alignement et avant le premier mouvement d'entrée. Dans le code précédent, la ligne N0010 fait le mouvement d'alignement, la ligne N0020 active la compensation et fait le premier mouvement d'entrée et la ligne N0030 fait le second mouvement d'entrée.

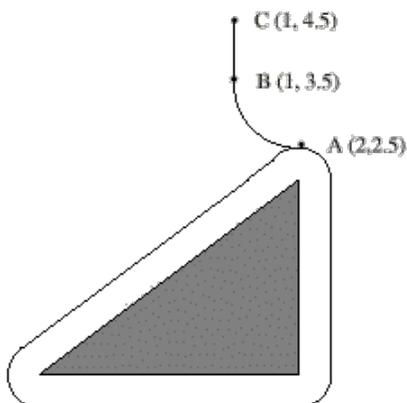


Figure 11. Cutter Radius Compensation Entry Moves

Dans cet exemple, l'arc AB et la ligne BC sont très larges, ce n'est pas nécessaire. Pour un contour sur parcours d'outil, le rayon de l'arc AB demande juste à être légèrement plus grand que la variation maximale du rayon de l'outil par rapport à son rayon nominal. Egalement, pour un contour sur parcours d'outil, le côté choisi pour la compensation doit être celui utilisé si l'outil est surdimensionné. Comme mentionné précédemment, si l'outil est sousdimensionné, l'interpréteur basculera de l'autre côté.

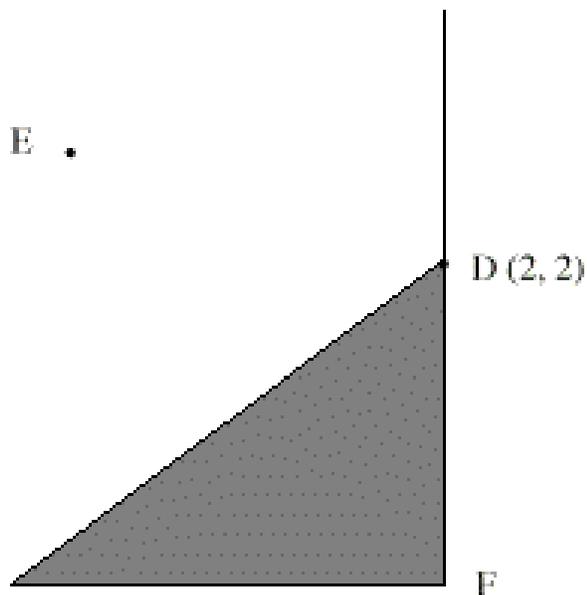
### Méthode simple

Si le contour est sur le profil du matériau et qu'il comprends des angles aigus quelque part sur le contour, une méthode simple pour faire l'entrée est possible. Voir la figure 12.

Premièrement, choisir un angle aigu, par exemple D. Ensuite, décider comment on va tourner autour du matériau depuis le point D. Dans notre exemple nous maintiendrons l'outil à gauche du profil et nous avancerons vers F. Prolonger la ligne FD (si le segment suivant du contour est un arc, prolonger la tangente à l'arc FD depuis D) pour diviser la surface extérieure au contour proche de D en deux parties. S'assurer que le centre de l'outil est actuellement dans la partie du même côté de la ligne prolongée que le matériau. Sinon, déplacer l'outil dans cette partie. Par exemple, le point E représente la position courante du centre de l'outil. Comme il est du même côté de la ligne FD prolongée que le triangle gris du matériau, aucun mouvement supplémentaire n'est nécessaire. Maintenant écrire la ligne de code NC qui active la compensation et faire le mouvement vers le point D

```
N0010 G41 G1 X2 Y2 (active la
compensation et fait le mouvement
d'entrée)
```

Cette méthode fonctionnera également avec un angle aigu sur un contour sur parcours d'outil, si l'outil est surdimensionné, mais elle échouera si il est sousdimensionné.



### **Autres points où est exécutée la compensation de rayon d'outil**

Le jeu complet de fonctions canoniques comprend des fonctions qui activent et désactivent la compensation de rayon d'outil, de sorte qu'elle puisse être activée quand le contrôleur exécute une de ces fonctions. Dans l'interpréteur cependant, ces commandes ne sont pas utilisées. La compensation est assurée par l'interpréteur et reflétée dans les sorties des commandes, c'est l'interpréteur qui continuera à diriger les mouvements du centre de l'outil. Cela simplifie le travail du contrôleur de mouvement tout en rendant le travail de l'interpréteur un peu plus difficile.

### **Algorithmes pour compensation de rayon d'outil**

L'interpréteur permet que les mouvements d'entrée et de sortie soient des arcs. Le comportement pour les mouvements intermédiaires est le même, excepté que certaines situations sont traitées comme des erreurs par l'interpréteur alors qu'elles ne le sont pas sur d'autres contrôleurs de machine.

Données relatives à la compensation de rayon d'outil :

L'interpréteur conserve trois données pour la compensation de rayon d'outil : Le réglage lui même (gauche, droite ou arrêt), `program_x` et `program_y`. Les deux dernières représentent les positions en X et en Y données dans le code NC quand la compensation est active. Quand elle est désactivée, les deux entrées sont fixées à de très petites valeurs ( $10 \times 10^{-20}$ ) dont la valeur symbolique (dans un `#define`) est "unknown". L'interpréteur utilise, les items `current_x` et `current_y` qui représentent, le centre de la pointe de l'outil (dans le système de coordonnées courant), à tout moment.

### **Exemples de Jon Elson**

Toutes les informations spécifiques au système se réfèrent au programme EMC du NIST, mais doit aussi s'appliquer aux plus modernes contrôleurs CNC. Ma méthode de vérification de ces programmes est d'abord de sélectionner l'outil zéro, de sorte que les commandes de compensation soient ignorées. Ensuite, je colle une feuille de papier sur une plaque tenue de niveau dans l'étau, une sorte de platine. J'installe une recharge de stylo à ressort dans la broche. C'est une recharge standard de stylo à bille en métal avec un ressort, dans un corps de 12mm de diamètre. Elle a un ressort pour la faire rentrer dans le corps du stylo, et un 'collet' à l'arrière qui permet à la pointe de se rétracter malgré le ressort, mais qui la laisse centrée à quelques dixièmes près. Je charge le programme avec l'outil zéro sélectionné, et il trace une ligne à l'extérieur de la pièce. (voir la figure suivante) Alors, je sélectionne un outil avec le diamètre de l'outil que j'envisage d'utiliser et je lance le programme une nouvelle fois. (Noter que la coordonnée Z dans le programme ne doit pas être changée pour éviter de plonger le stylo au travers du plateau;-) Maintenant, je dois voir si la compensation G41 ou G42 que je spécifie passe sur le côté voulu de la pièce. Sinon, je modifie avec la compensation du côté opposé, et j'édite la compensation opposée dans le programme, puis j'essaye à nouveau. Maintenant, avec l'outil sur le côté correct de la pièce, je peut vérifier si quelque part sur le parcours l'outil est 'trop gros' pour usiner les surfaces concaves. Ma vieille Allen-Bradley 7320 était très indulgente sur ce point, mais EMC ne tolère rien. Si vous avez la moindre concavité où deux lignes se rencontrent à moins de 180 degrés avec un outil de taille définies, EMC va s'arrêter là, avec un message d'erreur. Même si le gougeage est de .001mm de profondeur. Alors, je fais toujours l'approche sur le mouvement d'entrée et le mouvement de sortie juste sur un coin de la pièce, en fournissant un angle de plus de 180 degrés, afin qu'EMC ne hurle pas. Cela exige une grande attention lors de l'ajustement des points de départ et de sortie, qui ne sont pas compensés par le rayon d'outil, mais ils doivent être choisis avec un rayon approximatif bien réfléchi.

Les commandes sont :

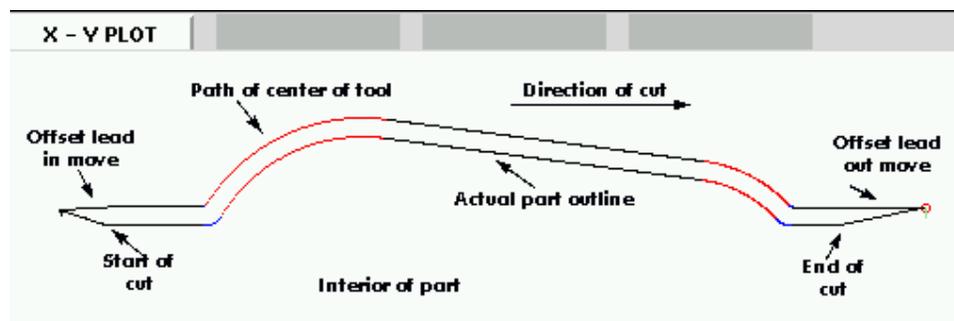
- G40 - Annuler la compensation de rayon d'outil
- G41 - Activer la compensation, outil à gauche du profil
- G42 - Activer la compensation, outil à droite du profil

Voici un petit fichier qui usine le côté d'une pièce avec de multiples arcs convexes et concaves et plusieurs lignes droites. La plupart de ces commandes ont été tracées depuis Bobcad/CAM, mais les lignes N15 et N110 ont été ajoutées par moi et certaines coordonnées dans ce contour ont été bricolées un peu par moi.

```
N10 G01 G40 X-1.3531 Y3.4
N15 F10 G17 G41 D4 X-0.7 Y3.1875 (ligne d'entrée)
N20 X0. Y3.1875
N40 X0.5667 F10
N50 G03 X0.8225 Y3.3307 R0.3
N60 G02 X2.9728 Y4.3563 R2.1875
N70 G01 X7.212 Y3.7986
N80 G02 X8.1985 Y3.2849 R1.625
N90 G03 X8.4197 Y3.1875 R0.3
N100 G01 X9.
N110 G40 X10.1972 Y3.432 (ligne de sortie)
N220 M02
```

La ligne 15 contient G41 D4, qui signifie que le diamètre de l'outil est celui de l'outil #4 dans la table d'outils, il sera utilisé pour décaler la broche de 1/2 diamètre, qui est, bien sûr, le rayon d'outil. Noter que la ligne avec la commande G41 contient le point final du mouvement dans lequel la compensation de rayon est interpolée. Cela signifie qu'au début de ce mouvement, il n'y a aucun effet de compensation et à la fin, l'outil est décalé de 100% du rayon de l'outil sélectionné. Immédiatement après le G41 il y a D4, signifiant que le décalage sera le rayon de l'outil N°4 dans la table d'outils. Noter que les DIAMETRES d'outil sont entrés dans la table d'outils. (le diamètre de l'outil de Jon est de 0.4890)

Mais, noter qu'à la ligne 110, où il y a la commande G40, l'interpolation de la compensation d'outil est en dehors de ce mouvement. La manière d'obtenir ce réglage, les mouvements des lignes 15 et 110 sont presque exactement parallèles à l'axe X et la différence dans les coordonnées Y est à la ligne où l'outil est appelé, en dehors de la compensation d'outil.



Certaines autres choses sont à noter, le programme commence avec G40, pour désactiver les compensations éventuellement actives. Cela évite un tas d'ennuis quand le programme s'arrête à cause d'une erreur de concavité, mais laisse la compensation désactivée. Noter aussi, en ligne 15, G17 est utilisé pour spécifier le plan de travail XY pour les interpolations circulaires. J'ai utilisé le format rayon pour les spécifications des arcs plutôt que la forme I, J. EMC est très pointilleux au sujet des rayons qu'il calcule à partir du format des coordonnées I, J et il doit trouver le début et la fin du mouvement avec  $10^{-11}$  unités internes, de sorte qu'il y a beaucoup de problèmes avec des arcs arbitraires. Normalement, si vous avez un arc de 90 degrés, centré sur (1.0,1.0) avec un rayon de 1", tout ira bien, mais si le rayon ne peut pas être exprimé exactement et avec juste le nombre de chiffres significatifs, ou si l'arc à un nombre étrange de degrés, alors les problèmes commencent avec EMC. Le mot R supprime tous ce désordre et il est beaucoup plus facile de travailler avec lui, de toute façon. Si l'arc est de plus de 180 degrés, R doit être négatif.

# Chapitre 18

## Systemes de coordonnées et décalage G92

### 18.1 Introduction

Vous avez vu comme il est pratique d'utiliser la compensation de longueur d'outil, elle évite au programmeur d'avoir à connaître en tout temps la hauteur de l'outil quand il écrit un programme. De la même manière, il est très intéressant d'utiliser un point de référence sur le brut ou la pièce et de faire travailler le programme à partir de ce point, sans avoir à tenir compte d'où est placée la pièce sur la machine pendant l'usinage.

Ce chapitre va introduire les décalages et comment ils sont utilisés par EMC. Ce qui inclus :

- Les coordonnées machine (G53)
- Les neuf décalages d'origine pièce (G54 à G59.3 )
- Un jeu de décalages globaux (G92).

### 18.2 Commande en coordonnées machine : G53

Indépendamment de tout décalage pouvant être actif, un G53 dans une ligne de code indique à l'interpréteur de se déplacer aux positions réelles des axes (positions absolues), commandées dans la ligne. Par exemple :

```
G53 G0 X0 Y0 Z0
```

déplacera le mobile depuis la position actuelle vers la position où les coordonnées machine des trois axes seront à zéro. Vous pouvez utiliser cette commande si vous avez une position fixe pour le changement d'outil ou si votre machine dispose d'un changeur automatique. Vous pouvez aussi utiliser cette commande pour dégager la zone de travail et accéder à la pièce dans l'étau.

G53 est une commande non modale. Elle doit être utilisée sur chaque ligne où un mouvement basé sur les coordonnées machine est souhaité.

### 18.3 Décalages pièce (G54 à G59.3 )

Le décalage d'origine est utilisé pour séparer le point de référence de la pièce, de l'origine machine, créant ainsi un système de coordonnées (relatif), propre à chaque pièce et séparé du système de coordonnées machine (absolu). Il permet, entre autre, dans le cas de pièces multiples mais semblables, de créer en décalant ses origine, le système de coordonnées de chaque pièce, le programme restant

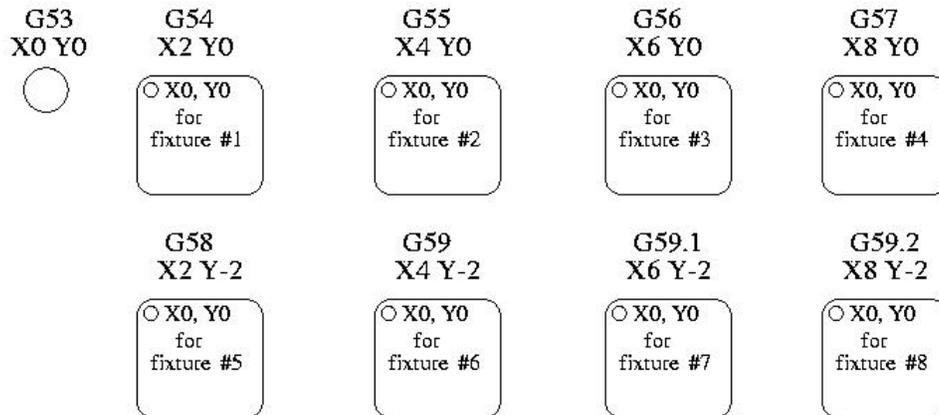


FIG. 18.1 – Décalages pour 8 ilots

le même. Un cas typique d'utilisation de cette fonctionnalité, pour usiner huit ilots identiques sur la même pièce est illustré sur la figure 18.1

Les valeurs des décalages sont enregistrées dans le fichier VAR qui est requis par le fichier INI durant le démarrage d'EMC. Dans l'exemple ci-dessous qui utilise G55 la valeur de chacun des axes de G55 est enregistrée dans une variable numérotée.

```
5241  0.000000
5242  0.000000
5243  0.000000
5244  0.000000
5245  0.000000
5246  0.000000
```

Dans le schéma d'un fichier VAR, la première variable contient la valeur du décalage de l'axe X, la seconde variable le décalage de l'axe Y et ainsi de suite pour les six axes. Il y a une série de variables de ce genre pour chacun des décalages pièce.

Chacune des interfaces graphiques offre un moyen de fixer les valeurs de ces décalages. Vous pouvez également définir ces valeurs en modifiant le fichier VAR lui-même, puis faire un [reset], pour qu'EMC lise les nouvelles valeurs. Pour notre exemple, nous allons éditer directement le fichier pour que G55 prenne les valeurs suivantes :

```
5241  2.000000
5242  1.000000
5243  -2.000000
5244  0.000000
5245  0.000000
5246  0.000000
```

Vous pouvez voir que les positions zéro de G55 sont passées en X=2, Y=1 et Z=-2 éloignées donc de l'origine absolue (machine).

Une fois que les valeurs sont assignées, un appel de G55 dans une ligne de programme décalera le point de référence zéro, l'origine, vers les valeurs enregistrées. La ligne suivante décalera chacun des axes vers sa nouvelle position d'origine. Contrairement à G53, les commandes G54 à G59.3 sont modales. Elles agissent sur toutes les lignes de g-code du programme après qu'une ait été rencontrée. Voyons le programme qui pourrait être écrit à l'aide de la figure 18.1, il suffira d'un seul point de référence pour chacune des pièces pour faire tout le travail. Le code suivant est proposé

comme exemple pour faire un rectangle, il utilisera les décalages G55 que nous avons expliqué ci-dessus.

```
G55 G0 x0 y0 z0
g1 f2 z-0.2000
x1
y1
x0
y0
g0 z0
g54 x0 y0 z0
m2
```

“Mais,” dites vous, “pourquoi y a-t-il un G54 vers la fin ?” C’est une pratique courante de quitter le système de coordonnées G54 avec l’ensemble des valeurs d’axes à zéro afin de laisser un code modal basé sur les positions machine absolues. Nous le faisons avec cette commande qui met la machine à zéro. Il aurait été possible d’utiliser G53 et d’arriver au même endroit, mais la commande n’aurait pas été modale, les commandes suivantes auraient voulu retourner dans le système de coordonnées du G55 toujours actif.

```
G54   utilise les réglages du système de coordonnées 1
G55   utilise les réglages du système de coordonnées 2
G56   utilise les réglages du système de coordonnées 3
G57   utilise les réglages du système de coordonnées 4
G58   utilise les réglages du système de coordonnées 5
G59   utilise les réglages du système de coordonnées 6
G59.1 utilise les réglages du système de coordonnées 7
G59.2 utilise les réglages du système de coordonnées 8
G59.3 utilise les réglages du système de coordonnées 9
```

### 18.3.1 Système de coordonnées par défaut

Une autre variable dans le fichier VAR joue un rôle important dans les décalages, c’est la variable 5220. Dans les fichiers par défaut, sa valeur est fixée à 1,00000. Ce qui signifie que lorsque EMC démarre, il doit utiliser le premier système de coordonnées comme système par défaut. Si vous définissez celui-ci à 9,00000 le neuvième système décalé sera utilisé par défaut au démarrage du système et aux réinitialisations. Toute valeur autre qu’un entier compris entre 1 et 9, ou l’absence de la variable 5220, provoquera au démarrage le retour d’EMC à la valeur par défaut de 1.00000.

### 18.3.2 Réglage des décalages avec G10

La commande G10 L- peut être utilisée pour modifier les valeurs des décalages dans un système de coordonnées. (add here)

## 18.4 G92 Décalages d’axes

G92 est la plus incomprise et la plus maligne des commandes programmables avec EMC. La façon dont elle fonctionne a un peu changé entre les premières versions et l’actuelle. Ces changements ont sans doute déconcerté de nombreux utilisateurs. Elle devrait être vue comme une commande produisant un décalage temporaire, qui s’applique à tous les autres décalages.

En réponse à une critique de cette commande, Ray Henry l'a étudiée en comparant la façon dont les auteurs de l'interpréteur s'attendaient à ce qu'elle fonctionne et la façon dont elle fonctionnait sur sa mini-fraiseuse Grizzly. Dans les paragraphes ci-dessous, entre guillemets, vous trouverez des extraits de son document, disponible sur <http://www.linuxcnc.org>.

### 18.4.1 Les commandes G92

Ce jeu de commandes inclus :

**G92** Cette commande, utilisée avec des mots d'axes, fixe les valeurs des variables de décalage.

**G92.1** Cette commande met à zéro les valeurs des variables de g92.

**G92.2** Cette commande suspend, sans les mettre à zéro, les variables de g92.

**G92.3** Cette commande applique les valeurs de décalage qui ont été suspendues.

Lorsque les commandes sont utilisées comme décrit ci-dessus, elles fonctionneront à peu près comme vous vous y attendiez.

L'utilisateur doit bien comprendre le fonctionnement des valeurs de g92. Elles sont basées sur l'emplacement de chaque axe au moment où la commande g92 est invoquée. Le document du NIST est clair, " pour faire en sorte que le point actuel ait les coordonnées" x0, y0 et z0 nous utiliserons g92 x0 y0 z0. *G92 ne fonctionne pas depuis le système de coordonnées machine absolues. Il fonctionne à partir de l'emplacement actuel.*

G92 travaille également à partir d'un emplacement actuel déjà modifié par tout autre décalage actif au moment où la commande g92 est invoquée. Lors de test des différences entre les décalages de travail et les décalages réels, il a été constaté qu'un décalage g54 pouvait annuler un g92 et ainsi, donner l'apparence qu'aucun décalage n'était actif. Toutefois, le g92 était toujours actif, pour toutes les coordonnées et il a produit les décalages attendus pour tous les autres systèmes de coordonnées.

Il est probable que l'absence de contact d'origine machine et d'une bonne procédure de prise d'origine machine se traduira par de très grandes erreurs dans l'application des valeurs de g92 si elles existent dans le fichier VAR. Beaucoup d'utilisateurs d'EMC n'ont pas de contact d'origine machine sur leurs machines. Pour eux, l'origine machine devrait être trouvée en déplaçant chaque axe à l'emplacement supposé et en utilisant la commande de prise d'origine. Lorsque chaque axe est dans une position connue, la commande de prise d'origine recalculera comment les valeurs de g92 doivent être appliquées pour produire des résultats cohérents. Sans séquence de prise d'origine machine, les valeurs sont appliquées à la position de la machine au démarrage d'EMC.

### 18.4.2 Réglage des valeurs de G92

Il y a au moins deux façons d'établir les valeurs de G92.

- Par un clic droit de la souris sur les afficheurs de position de tkemc, une fenêtre s'ouvre dans laquelle il est possible de saisir une valeur.
- Par la commande G92.

Toutes les deux, fonctionnent depuis l'emplacement courant de l'axe auquel le déplacement doit être appliqué.

Programmer g92 x- y- z- a- b- c- fixe les valeurs des variables de G92 de sorte que chaque axe prenne la valeur associée à son nom. Ces valeurs sont assignées à la position courante des axes. Ces résultats satisfont les paragraphes un et deux du document du NIST.

Les commandes G92 fonctionnent à partir de la position courante de l'axe, à laquelle elles ajoutent ou soustraient des valeurs pour donner à la position courante la valeur assignée par la commande g92. Elles prennent effet même si d'autres décalages sont déjà actifs.

Ainsi, si l'axe X est actuellement en position  $X=2.0000$ , un G92 x0 fixera un décalage de  $-2.0000$ , de sorte que l'emplacement actuel de X devienne  $X=0.0000$ . Un nouveau G92 X5.0000 fixera un décalage de  $3.0000$  et l'affichage indiquera une position courante  $X=5.0000$ .

Maintenant, un G92 X5 fixera un décalage de  $0.0000$  et l'affichage ne changera pas.

### 18.4.3 Prudence avec G92

Parfois, les valeurs de décalage d'un G92 restent bloquées dans le fichier VAR. Quand ça arrive, une ré-initialisation ou un redémarrage peut les rendre de nouveau actifs. Les variables sont numérotées

```
5211  0.000000
5212  0.000000
5213  0.000000
5214  0.000000
5215  0.000000
5216  0.000000
```

où 5211 est le numéro du décalage de l'axe X et ainsi de suite. Si vous voyez des positions inattendues à la suite d'une commande de déplacement, ou même des chiffres inattendus dans l'affichage de la position lorsque vous démarrez, regardez ces variables dans le fichier VAR pour vérifier si elles contiennent des valeurs. Si c'est le cas, les mettre à zéro devrait solutionner le problème.

Avec ces tests, nous pouvons voir qu'après une réinitialisation, G92 retourne aux conditions qu'il avait au démarrage de l'interpréteur. Le lecteur doit noter que nous avons établi ... qu'aucune de ces valeurs n'est écrite durant un démarrage normal si aucun G92 n'a été fixé au démarrage, aucune ne pourra être lue lors d'un redémarrage.

Il est possible que ce soit le coeur du problème que certains ont vécu avec les différences entre l'ancien et le nouvel interpréteur. C'est possible, mais je laisse le soin à d'autres de faire des essais, pour vérifier que l'ancien interpréteur et les programmes écrivent les variables immédiatement et de constater ensuite que ces valeurs sont lues lors d'un redémarrage.

D'autre part, si les valeurs de G92 existent dans le fichier VAR lorsque EMC a démarré

... À partir d'EMC avec les valeurs de g92 dans le fichier de variables, EMC appliquera ces valeurs à partir de l'emplacement actuel de chaque axe. Si la prise d'origine machine a été faite et que l'origine machine correspond bien au zéro des axes, tout sera correct. Une fois que la prise d'origine a été faite en utilisant les contacts d'origine machine ou en déplaçant chaque axe à une position connue et en appliquant la commande de prise d'origine, les commandes g92 et leurs valeurs fonctionnent comme prévu.

Ces tests n'ont pas étudié l'effet produit par la lecture des variables dans le fichier alors qu'il contient des valeurs. Cela pourrait poser des problèmes si les décalages g92 avaient été enlevés avec g92.1 alors que le fichier de variables contenait encore les valeurs précédentes.

C'est cette complexité qui nous amène à dire que les valeurs de G92 doivent être considérées comme provisoires. Elles devraient être utilisées pour définir des décalages globaux et à court terme. Les commandes de décalage d'origine G54 à 59.3 devraient être utilisées chaque fois que des décalages durables et prévisibles sont nécessaires.

## 18.5 Exemple de programme utilisant les décalages d'axes

Cet exemple de projet de gravure fraise un jeu de quatre cercles de rayon .1 pouce dans une forme grossière d'étoile au centre du cercle. Nous pouvons configurer individuellement les formes de la façon suivante :

```
G10 L2 P1 x0 y0 z0 (assure que g54 a mis la machine à zéro)
g0 x-.1 y0 z0
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g0 z0
m2
```

Nous pouvons émettre une série de commandes pour créer des décalages pour les quatre autres cercles comme cela.

```
G10 L2 P2 x0.5 (offsets g55 x value by 0.5 inch)
G10 L2 P3 x-0.5 (offsets g56 x value by -0.5 inch)
G10 L2 P4 y0.5 (offsets g57 y value by 0.5 inch)
G10 L2 P5 y-0.5 (offsets g58 y value by -0.5 inch)
```

Nous mettons ces ensemble dans le programme suivant.

```
(Un programme de fraisage de cinq petits cercles dans un losange)
G10 L2 P1 x0 y0 z0 (assure que g54 a mis la machine à zéro)
G10 L2 P2 x0.5 (offsets g55 x value by 0.5 inch)
G10 L2 P3 x-0.5 (offsets g56 x value by -0.5 inch)
G10 L2 P4 y0.5 (offsets g57 y value by 0.5 inch)
G10 L2 P5 y-0.5 (offsets g58 y value by -0.5 inch)
g54 g0 x-.1 y0 z0 (center du cercle)
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g0 z0
g55 g0 x-.1 y0 z0 (premier décalage circulaire)
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g0 z0
g56 g0 x-.1 y0 z0 (second décalage circulaire)
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g0 z0
g57 g0 x-.1 y0 z0 (troisième décalage circulaire)
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g0 z0
g58 g0 x-.1 y0 z0 (décalage circulaire)
g1 f1 z-.25
g3 x-.1 y0 i.1 j0
g54 g0 x0 y0 z0
m2
```

Maintenant vient le moment où l'on applique une série de décalages G92 à ce programme. Vous verrez qu'il c'est fait dans chaque cas de z0. Si la machine étaient à la position de zéro, un g92 z1.0000 placé en tête de programme le décalera d'un pouce. Vous pouvez également modifier l'ensemble du dessin dans le plan XY en ajoutant quelques décalages x et y avec g92. Si vous faites cela, vous devez ajouter une commande G92.1 juste avant le m2 qui termine le programme. Si vous ne le faites pas, d'autres programmes que vous pourriez lancer après celui-ci, utiliseront également les décalages g92. En outre, cela permettrait d'éviter d'écrire les valeurs de g92 lorsque vous arrêtez EMC et donc, d'éviter de les recharger quand vous démarrez à nouveau le programme.

# Chapitre 19

## Spécificités des tours

Ce chapitre va regrouper les informations spécifiques aux tours, il est en cours de rédaction.

### 19.1 EMC configuré pour un tour

En plus d'utiliser une configuration propre à un tour il est préférable d'utiliser également un fichier d'outils de sorte que AXIS affiche correctement la forme de l'outil. La section sur les fichiers INI du manuel de l'intégrateur indique comment configurer AXIS pour un tour.

### 19.2 Fichier d'outils

Les outils de tour diffèrent de ceux d'une fraiseuse par l'ajout des angles FRONTANGLE (angle avant), BACKANGLE (angle arrière) et ORIENT (orientation). La table suivante montre le format d'un fichier d'outils pour tour.

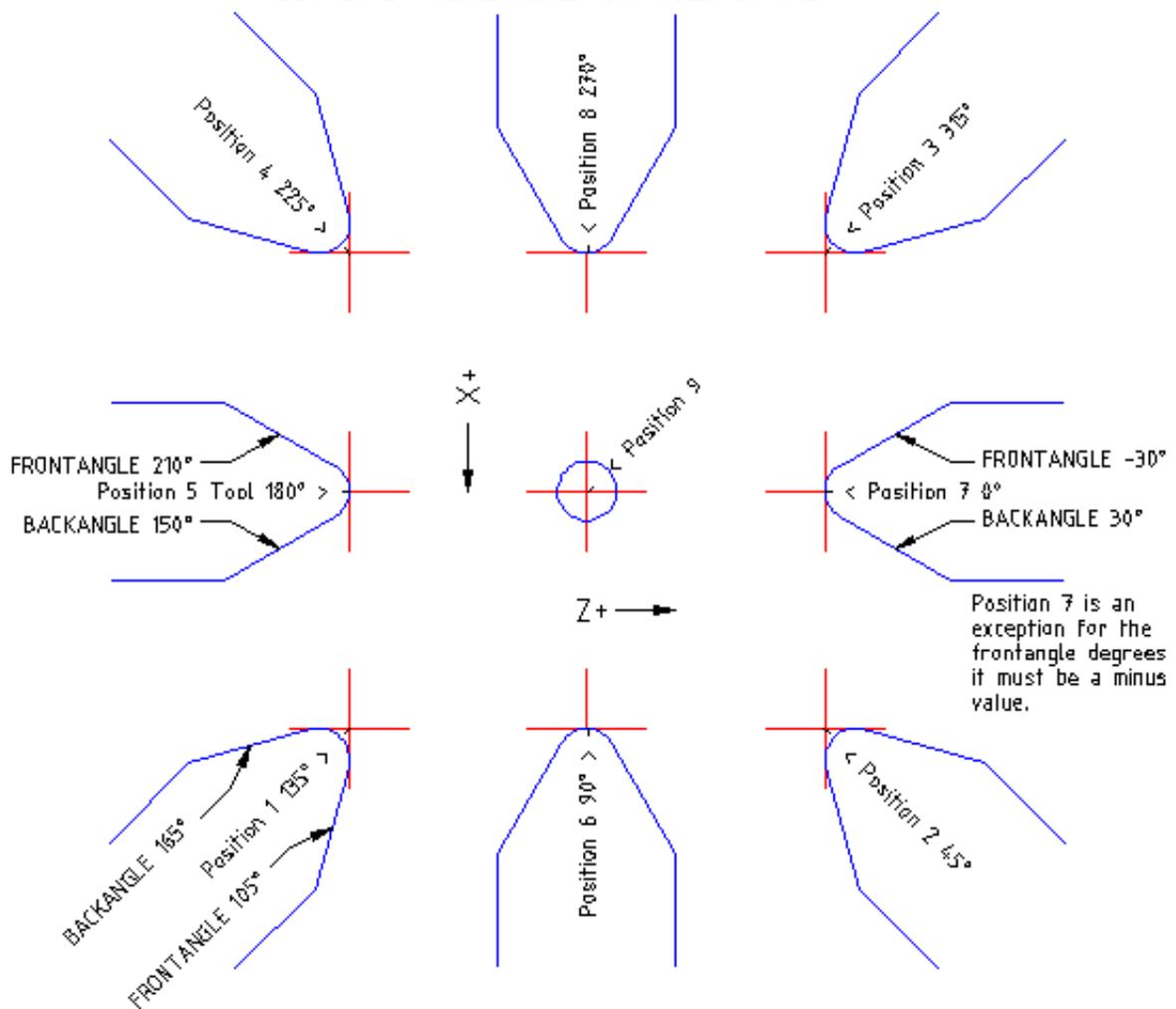
Les valeurs du fichier d'outils peuvent être ajustées en utilisant G10 L1, à l'exception de FRONTANGLE, BACKANGLE et COMMENT qui doivent être, actuellement, renseignés manuellement.

La figure suivante indique les différentes positions conventionnelles ainsi que l'angle de l'axe de l'outil et les informations concernant le FRONTANGLE et le BACKANGLE.

TAB. 19.1 – Fichier d'outils pour tour

Pocket	FMS	ZOFFSET	XOFFSET	DIA	FRONTANGLE	BACKANGLE	ORIENT	COMMENT
1	1	0.000	0.000	0.10	105.0	165.0	1	60 deg
2	2	0.000	0.000	0.05	22.5	67.5	2	45 deg
3	3	0.000	0.000	0.10	277.5	352.5	3	75 deg
4	4	0.000	0.000	0.10	255.0	195.0	4	60 deg
5	5	0.000	0.000	0.10	202.5	157.5	5	45 deg
6	6	0.000	0.000	0.10	90	45	6	45 deg
7	7	0.000	0.000	0.10	-22.5	22.5	7	45 deg
8	8	0.000	0.000	0.10	247.5	292.5	8	45 deg
9	9	0.000	0.000	0.10	0	0	9	

FIG. 19.1 – Orientations des outils de tour



La figure suivante montre comment sont représentées par AXIS les positions d'outil du fichier précédent.

FIG. 19.2 – Tool Positions 1, 2, 3 & 4

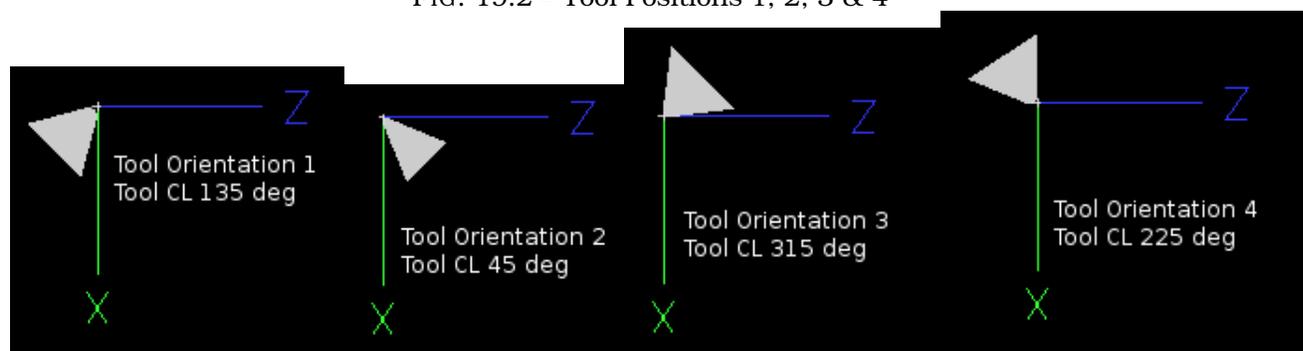
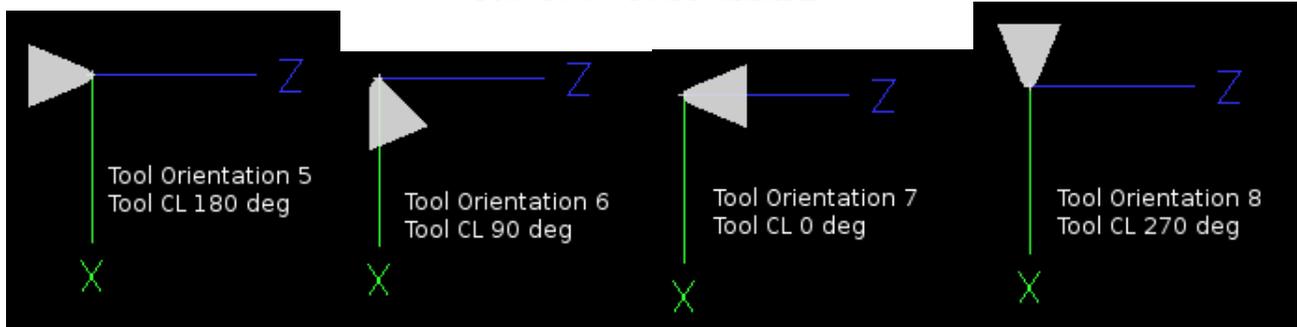


FIG. 19.3 – Tool Position 2



### 19.3 Tangenter et corriger l'outil

Quand AXIS est utilisé sur un tour, il est possible de corriger l'outil sur les deux axes X et Z. Les corrections sont alors introduites dans la table d'outils en utilisant le bouton « Toucher » et sa fenêtre de dialogue.

Voici une séquence typique :

1. Prise d'origine de chacun des axes.
2. Déclarer l'outil avec « Tn M6 » dans lequel "n" est le numéro de l'outil courant.
3. Sélectionner l'axe X dans la fenêtre de l'onglet « Contrôle manuel » (F3).
4. Déplacer l'axe X sur une position connue ou prendre une passe de test et mesurer le diamètre obtenu.
5. Cliquer le bouton « Toucher » et choisir l'option « T Table d'outils », ce qui entrera la correction dans la table.
6. Recommencer la même séquence pour corriger l'axe Z.

Note : Si l'opérateur travaille en mode rayon (G8), la correction introduite dans le fichier d'outils portera sur le rayon, par contre, elle sera relative au diamètre si le mode diamètre (G7) est utilisé.

### 19.4 Filetage

Sur un tour, le filetage nécessite un signal de retour entre la broche et EMC2. Généralement, c'est un codeur de position qui fournit ce retour. Le manuel de l'intégrateur donne davantage d'informations sur le codeur de broche.

Le cycle de filetage préprogrammé G76 est utilisé, tant en filetage intérieur qu'en filetage extérieur, voir G7612.22 dans la section G-code.

**Quatrième partie**

**Quelques exemples**

# Chapitre 20

## Exemples de G-Code

Après l'installation d'EMC2 vous pourrez trouver plusieurs fichiers d'exemple placés dans le répertoire /nc\_files de votre dossier d'installation. Vous ne pouvez utiliser que les fichiers appropriés au type de votre machine.

### 20.1 Exemples pour une fraiseuse

#### 20.1.1 Fraisage hélicoïdal d'un orifice

Nom du fichier : useful-subroutines.ngc

Description : Programme pour usiner une poche avec utilisation des paramètres.

#### 20.1.2 Rainurage

Nom du fichier : useful-subroutines.ngc

Description : Programme pour fraiser une rainure avec utilisation des paramètres.

#### 20.1.3 Palpage d'une grille rectangulaire de points

Nom du fichier : gridprobe.ngc

Description : Relevé d'une grille rectangulaire de points au palpeur.

Ce programme palpe de manière répétitive, en suivant une grille régulière en XY et écrit les points mesurés dans le fichier 'probe-results.txt' situé dans le même répertoire que le fichier .ini.

#### 20.1.4 Amélioration du palpage d'une grille rectangulaire de points

Nom du fichier : smartprobe.ngc

Description : Relevé d'une grille rectangulaire de points au palpeur.

Ce programme est une amélioration du précédent. Il palpe de manière répétitive, en suivant une grille régulière en XY et écrit les points mesurés dans le fichier 'probe-results.txt' situé dans le même répertoire que le fichier .ini.

### **20.1.5 Mesure de longueur d'outil**

Nom du fichier : tool-length-probe.ngc

Description : Mesure automatique de la longueur de l'outil.

Ce programme donne un exemple de la mesure automatique de longueur d'outil en utilisant un contact raccordé à l'entrée sonde. C'est très pratique pour une machine sur laquelle la longueur des outils est différente à chaque montage.

### **20.1.6 Mesure d'un alésage au palpeur**

Nom du fichier : probe-hole.ngc

Description : Mesure le centre et le diamètre d'un alésage.

Ce programme montre comment trouver le centre d'un alésage, comment calculer son diamètre et enregistrer les mesures dans un fichier.

### **20.1.7 Compensation de rayon d'outil**

Nom du fichier : comp-g1.ngc

Description : Mouvements d'entrée et de sortie avec la compensation de rayon d'outil.

Ce programme démontre la particularité du chemin d'outil sans et avec la compensation de rayon d'outil. Le rayon d'outil est pris dans la table d'outils.

## **20.2 Exemples pour un tour**

### **20.2.1 Filetage**

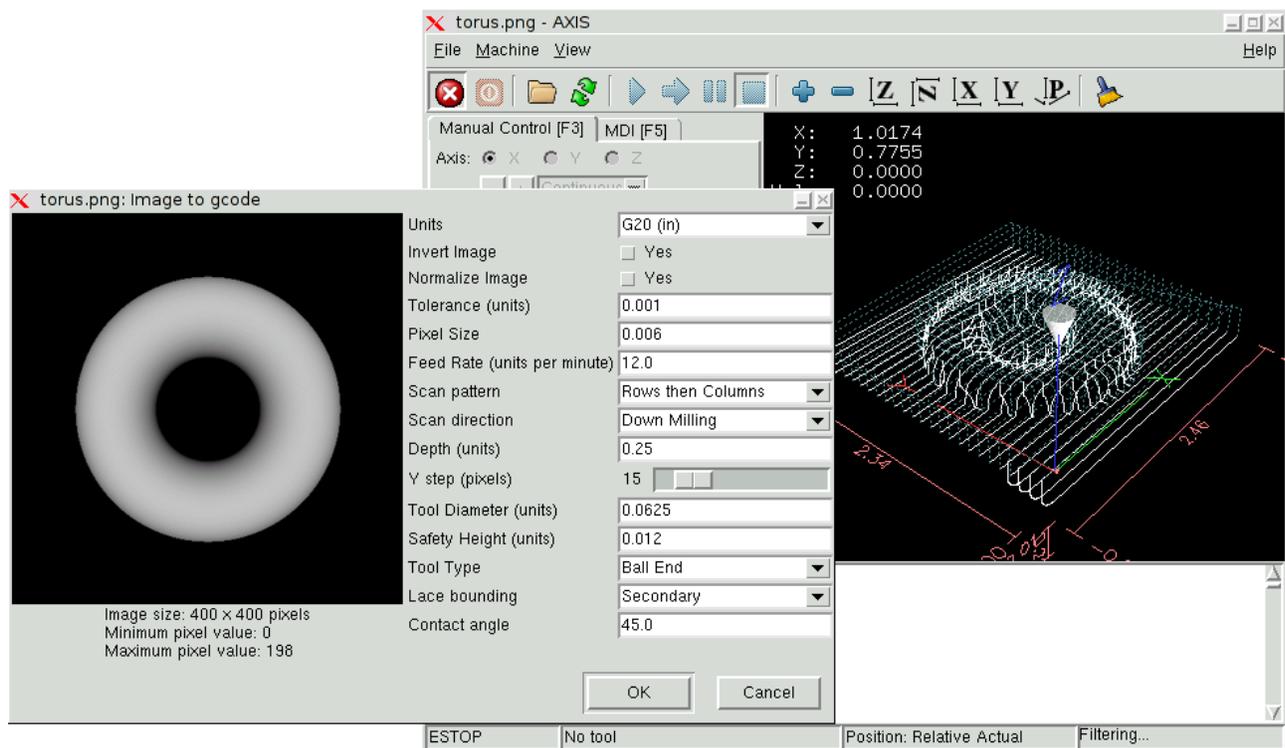
Nom du fichier : lathe-g76.ngc

Description : Dressage, filetage et tronçonnage.

Ce programme donne un exemple de filetage automatique sur un tour avec utilisation des paramètres.

# Chapitre 21

## Image-to-gcode : Usiner un “depth maps”



### 21.1 Qu'est-ce qu'un “depth map” ?

Il s'agit d'une image en échelle de gris dont la luminosité de chaque pixel correspond à la profondeur (ou hauteur) de chaque point de l'objet.

### 21.2 Intégrer image-to-gcode dans l'interface utilisateur d'AXIS

Ajoutez les lignes suivantes dans la section : [FILTER] de votre fichier .ini pour qu'AXIS invoque automatiquement image-to-gcode à l'ouverture d'une image .png, .gif, ou .jpg :

```
PROGRAM_EXTENSION = .png, .gif, .jpg Grayscale Depth Image
png = image-to-gcode
gif = image-to-gcode
jpg = image-to-gcode
```

Le fichier de configuration : `sim/axis.ini` est déjà configuré de cette façon.

## 21.3 Utilisation d'Image-to-gcode

image-to-gcode peut être démarré soit en ouvrant une image dans AXIS, soit en invoquant image-to-gcode dans une console, de la manière suivante :

```
image-to-gcode torus.png > torus.ngc
```

Ajustez les réglages dans la colonne de droite, puis pressez OK pour créer le g-code. Selon la taille de l'image et les options choisies, le traitement peut durer de quelques secondes à quelques minutes. Quand une image est appelée, le gcode sera automatiquement chargé et visualisé dans AXIS une fois le traitement terminé. Dans AXIS, faites "Recharger" pour afficher de nouveau l'écran d'options d' image-to-gcode, vous pourrez ainsi travailler en boucle.

## 21.4 Les différentes options

### 21.4.1 Unités

Spécifie quelle unité sera utilisée dans le g-code généré G20 (pouces) ou G21 (mm), ce sera également l'unité utilisée par toutes les options marquées : **(units)**.

### 21.4.2 Invert Image

Si "no", le pixel noir sera le point le plus bas et le pixel blanc le point le plus haut. Si "yes", le pixel noir sera le point le plus haut et le pixel blanc le point le plus bas.

### 21.4.3 Normalize Image

Si "yes", le pixel le plus sombre est ramené au noir, le pixel le plus lumineux est ramené au blanc.

### 21.4.4 Expand Image Border

Si "None", l'image entrée sera utilisée telle-quelle, les détails les plus aux bords de l'image pourraient être coupés. Si "White" ou "Black", alors une bordure de pixels égale au diamètre de l'outil sera ajoutée sur tout le pourtour pour éviter ce risque.

### 21.4.5 Tolerance (unités)

Quand une série de points est proche d'une ligne droite au point d'être dans la **tolerance**, elle sera traitée comme une ligne droite en sortie. Augmenter la tolérance peut donner de meilleures performances de contourage avec emc, mais peut aussi estomper ou gommer les détails les plus fins de l'image.

### 21.4.6 Pixel Size (unités)

Il y a beaucoup d'unités pour un pixel dans l'image entrée. Habituellement ce nombre est beaucoup plus petit que 1.0. Par exemple, pour usiner un objet de 50x50mm depuis une image de 400x400 pixels, utiliser un "pixel size" de 0.125, parce que  $50 / 400 = 0.125$ .

### 21.4.7 Plunge Feed Rate (unités par minute)

Vitesse du mouvement de plongée initial.

### 21.4.8 Feed Rate (unités par minute)

Vitesse d'avance pour le reste de l'usinage.

### 21.4.9 Spindle Speed (RPM)

Vitesse de rotation de la broche, en tours/mn

### 21.4.10 Scan Pattern

Modèles de balayage possibles :

- Rangées
- Colonnes
- Rangées puis colonnes
- Colonnes puis rangées

### 21.4.11 Scan Direction

Directions de balayage possibles :

- Positive : le fraisage commence à de petites valeurs de X ou Y et se poursuit avec des valeurs croissantes.
- Négative : le fraisage commence à des valeurs élevées de X ou Y et se poursuit avec des valeurs décroissantes.
- Alternative : le fraisage commence aux valeurs de X ou Y où s'est terminé le dernier mouvement. Cela réduit les déplacements "en l'air".
- Up Milling : le fraisage commence en points bas et se poursuit vers les points hauts.
- Down Milling : le fraisage commence en points hauts et se poursuit vers les points bas.

### 21.4.12 Depth (unités)

Le dessus du bloc est toujours à **Z=0**. La profondeur d'usinage dans le matériau est de **Z=-depth**.

### 21.4.13 Step Over (pixels)

Distance entre rangées ou colonnes adjacentes. Pour trouver le nombre en pixels pour une distance donnée en unités, calculez : **distance/pixel size** et arrondissez au nombre le plus proche. Par exemple : si **pixel size=.006** et le pas souhaité sur la **distance=.015**, alors utilisez un Step Over de 2 ou 3 pixels, parce que **.015/.006=2.5**.

#### 21.4.14 Tool Diameter

Le diamètre du taillant de l'outil.

#### 21.4.15 Safety Height

La hauteur à laquelle les mouvements de traversée. image-to-gcode considère toujours le dessus du matériau comme étant : **Z=0**.

#### 21.4.16 Tool Type

La forme du taillant de l'outil. Les formes possibles sont :

- Hémisphérique
- Plate
- Vê à 45 degrés
- Vê à 60 degrés

#### 21.4.17 Lace bounding

Contrôle si les zones relativement plates le long d'une colonne ou d'une rangée peuvent être ignorées. Ces options n'ont de sens que pour un fraisage dans les deux directions. Trois choix sont possibles :

- None : toutes les rangées et les colonnes seront entièrement fraisées.
- Secondary : lors du fraisage dans la deuxième direction, les zones qui ne présentent pas une forte pente dans cette direction seront ignorées.
- Full : lors du fraisage dans la première direction, les zones qui présentent une forte pente dans la deuxième direction seront ignorées. Lors du fraisage dans la deuxième direction, les zones qui ne présentent pas une forte pente dans cette direction seront ignorées.

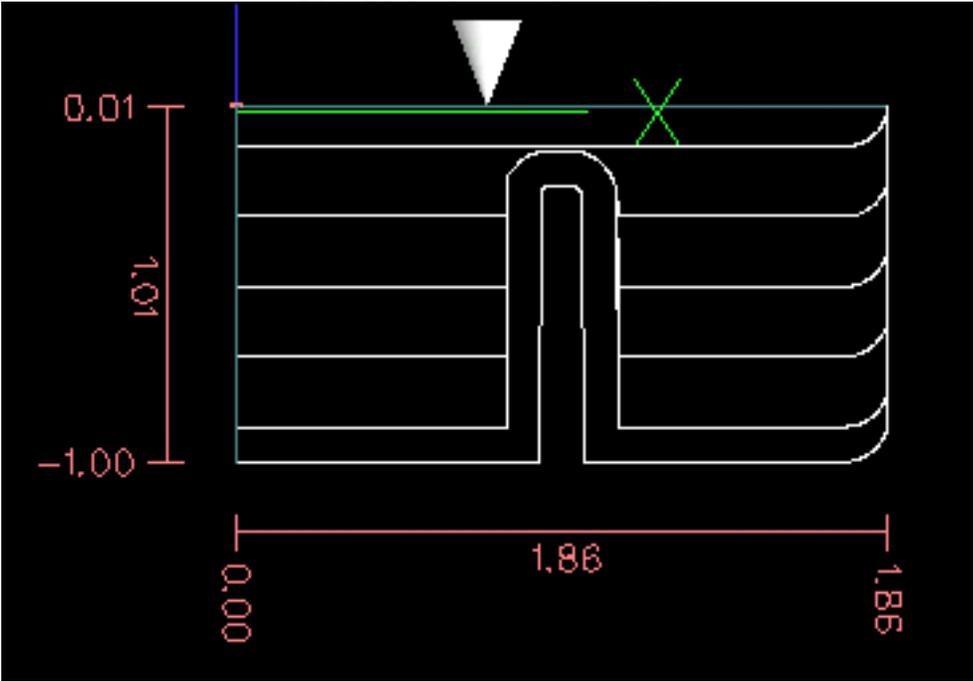
#### 21.4.18 Contact angle

Quand **Lace bounding** n'est pas None, les pentes qui présentent une pente supérieure à **Contact angle** seront considérées comme de "fortes" pentes et celles en dessous de cet angle considérées comme de faible pentes.

#### 21.4.19 Offset d'ébauche et profondeur par passe d'ébauche

Image-to-gcode peut optionnellement produire des passes d'ébauche. La profondeur des passes d'ébauche successives est fixée par "Roughing depth per pass". Par exemple, entrer 0.2 pour une première passe d'ébauche d'une profondeur de 0.2, la seconde passe d'ébauche aura une profondeur de 0.4 et ainsi de suite, jusqu'à ce que la profondeur totale Depth de l'image soit atteinte. Aucune des passes d'ébauche n'usinera plus près de la partie finale que Roughing Offset. La figure 21.1 montre une grande profondeur verticale à usiner. Sur cette image, la profondeur des passes d'ébauche est de 0.2 pouces et Roughing Offset de 0.1 pouces.

FIG. 21.1 – Passes d'ébauche



## **Cinquième partie**

# **Diagnostic**

# Chapitre 22

## Moteurs pas à pas

Si ce que vous obtenez ne correspond pas à ce que vous espérez, la plupart du temps c'est juste un petit manque d'expérience. Accroître son expérience permet souvent une meilleure compréhension globale. Porter un diagnostic sur plusieurs problèmes est toujours plus facile en les prenant séparément, de même qu'une équation dont on a réduit le nombre de variables est toujours plus rapide à résoudre. Dans le monde réel ce n'est pas toujours le cas mais c'est une bonne voie à suivre.

### 22.1 Problèmes communs

#### 22.1.1 Le moteur n'avance que d'un pas

La raison la plus fréquente dans une nouvelle installation pour que le moteur ne bouge pas est l'interversion entre le signal de pas et le signal de direction. Si, quand vous pressez le bouton de jog dans un sens puis dans l'autre, le moteur n'avance que d'un pas à chaque fois et toujours dans la même direction, vous êtes dans ce cas.

#### 22.1.2 Le moteur ne bouge pas

Certaines interfaces de pilotage de moteurs ont une broche d'activation (enable) ou demandent un signal de pompe de charge pour activer leurs sorties.

#### 22.1.3 Distance incorrecte

Si vous commandez une distance de déplacement précise sur un axe et que le déplacement réel ne correspond pas, alors l'échelle de l'axe n'est pas bonne.

### 22.2 Messages d'erreur

#### 22.2.1 Erreur de suivi

Le concept d'erreur de suivi est étrange quand il s'agit de moteurs pas à pas. Etant un système en boucle ouverte, aucune contre réaction ne permet de savoir si le suivi est correct ou non. EMC calcule si il peut maintenir le suivi demandé par une commande, si ce n'est pas possible il stoppe le mouvement et affiche une erreur de suivi. Les erreurs de suivi sur les systèmes pas à pas sont habituellement les suivantes :

- FERROR to small - (FERROR trop petit)
- MIN\_FERROR to small - (MIN\_FERROR trop petit)
- MAX\_VELOCITY to fast - (MAX\_VELOCITY trop rapide)
- MAX\_ACCELERATION to fast - (MAX\_ACCELERATION trop rapide)
- BASE\_PERIOD set to long - (BASE\_PERIOD trop longue)
- Backlash ajouté à un axe (rattrapage de jeu)

Toutes ces erreurs se produisent lorsque l'horloge temps réel n'est pas capable de fournir le nombre de pas nécessaire pour maintenir la vitesse requise par le réglage de la variable BASE\_PERIOD. Ce qui peut se produire, par exemple après un test de latence trop bref pour obtenir un valeur fiable, dans ce cas, revenir à une valeur plus proche de ce qu'elle était et réessayez. C'est également le cas quand les valeurs de vitesse maximum et d'accélération maximum sont trop élevées.

Si un backlash a été ajouté, il est nécessaire d'augmenter STEPGEN\_MAXACCEL aux environs du double de MAX\_ACCELERATION dans la section [AXIS] du fichier INI et ce, pour chacun des axes sur lesquels un backlash a été ajouté. EMC2 utilise une "extra accélération" au moment de l'inversion de sens pour reprendre le jeu. Sans correction du backlash, l'accélération pour le générateur de pas peut être juste un peu plus basse que celle du planificateur de mouvements.

### 22.2.2 Erreur de RTAPI

Quand vous rencontrez cette erreur :

```
RTAPI : ERROR : Unexpected realtime delay on task n
```

C'est généralement que la variable BASE\_PERIOD dans la section [EMCMOT] du fichier ini a une valeur trop petite. Vous devez lancer un « Latency Test » pendant une durée plus longue pour voir si vous n'avez pas un délai excessif quelque part, responsable de ce problème. Si c'est le cas réajuster alors BASE\_PERIOD avec la nouvelle valeur obtenue.

EMC2 vérifie le nombre de cycles du CPU entre les invocations du thread temps réel. Si certains éléments de votre matériel provoquent un délai excessif ou que les threads sont ajustés à des valeurs trop rapides, vous rencontrerez cette erreur.

NOTE : Cette erreur n'est affichée qu'une seule fois par session. En effet, si votre BASE\_PERIOD était trop basse vous pourriez avoir des centaines de milliers de messages d'erreur par seconde si plus d'un était affiché.

## 22.3 Tester

### 22.3.1 Tester le timing des pas

Si un de vos axes vibre, grogne ou fait des petits mouvements dans toutes les directions, c'est révélateur d'un mauvais timing d'impulsions de pas de ce moteur. Les paramètres du pilote matériel sont à vérifier et à ajuster. Il peut aussi y avoir des pertes de pas aux changements de direction. Si le moteur cale complètement, il est aussi possible que les paramètres MAX\_ACCELERATION ou MAX\_VELOCITY aient des valeurs trop élevées.

Le programme suivant vérifie que la configuration de l'axe Z est correcte. Copiez le programme dans le répertoire de votre emc2/nc\_files nommez le « TestZ.ngc » ou similaire. Initialisez votre machine avec Z = 0.000 sur le dessus de la table. Chargez et lancez le programme. Il va effectuer 200 mouvements d'aller et retour entre 10.00 et 30.00mm. Si vous avez un problème de configuration, la position de l'axe Z affichée à la fin du programme, soit 10.00mm, ne correspondra pas à la position mesurée. Pour tester un autre axe remplacez simplement le Z des G0 par le nouvel axe.

```
( Faite Z=0 au dessus de la table avant de démarrer! )
( Ce programme teste les pertes de position en Z )
( msg, test 1 de la configuration de l'axe Z )
G21 #1000=100 ( boucle 100 fois )
( cette boucle comporte un delai après chaque mouvement )
( test des réglages d'accélération et de vitesse )
o100 while [#1000]
  G0 Z30.000
  G4 P0.250
  G0 Z10.000
  G4 P0.250
  #1000 = [#1000 - 1]
o100 endwhile
( msg, test 2 de la configuration de l'axe Z, pressez S pour continuer)
M1 (un arrêt ici)
#1000=100 ( boucle 100 fois )
( Les boucles suivantes n'ont plus de délai en fin de mouvements )
( test des « hold times » de la conf. du pilote et le réglage de max accel )
o101 while [#1000]
  G0 Z30.000 .
  G0 Z10.000
  #1000 = [#1000 - 1]
o101 endwhile
( msg, Fin Z doit être à 10mm au dessus de la table )
M2
```

## **Sixième partie**

### **Annexes**

# Chapter 23

## Legal Section

### 23.1 Copyright Terms

Copyright (c) 2000 LinuxCNC.org

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and one Back-Cover Text: "This EMC Handbook is the product of several authors writing for linuxCNC.org. As you find it to be of value in your work, we invite you to contribute to its revision and growth." A copy of the license is included in the section entitled "GNU Free Documentation License". If you do not find the license you may order a copy from Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307

### 23.2 GNU Free Documentation License

GNU Free Documentation License Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA  
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

#### 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

#### 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format,  $\LaTeX$  input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

## **2. VERBATIM COPYING**

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## **3. COPYING IN QUANTITY**

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

#### 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five). C. State on the Title page the name of the publisher of the Modified Version, as the publisher. D. Preserve all the copyright notices of the Document. E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. H. Include an unaltered copy of this License. I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version. N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

#### 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

## 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

## 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

## 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

## 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

**ADDENDUM:** How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

## **Part VII**

# **Index de l'ouvrage**

# Index

- .axisrc, 28
- A/U, 16, 29, 31
- ABORT, 9
- Appel de fichiers, 108
- Arrêt d'urgence, 17
- arrêt optionnel, 32
- Arrets optionnels, 55
- Arrosage, 21
- arrosage, 33, 51, 53
- arrêt optionnel, 52
- Auto, 9, 31, 36, 39, 46
- axes, 50
- axes lineaires primaires, 50
- axes lineaires secondaires, 51
- Axes rotatifs, 51
- AXIS, 6, 15, 25
- AXIS avec un tour, 25
  
- Bouton effacement de block, 52
- break, 107
- Broche, 21
- broche, 31, 32, 51
- Brouillard, 31
  
- call, 106
- carrousel d'outils, 51
- changeur d'outil, 51
- chargement, 51
- Chemin d'outil, 18
- CNC, 5, 16, 35
- commentaires, 64
- continue, 107
- Contrôle manuel, 16, 19
- correcteur de vitesse, 30
- Correcteur de vitesse broche, 22
- correcteur v. broche, 51
- correcteur vitesse, 42
- correcteur vitesse broche, 30
- Correcteurs de vitesse, 16, 22
- correcteurs vitesse, 51
- Cycles preprogrammes, 93
  
- do, 107
- Données manuelles, 16
  
- effacement de block, 54
- else, 107
  
- endif, 107
- endsub, 106
- endwhile, 107
- ESTOP, 9, 41
- Etendues du programme, 18
  
- feed override, 9
  
- G0 Interpolation lineaire rapide, 84
- G1 Interpolation lineaire travail, 84
- G10 L1 Entree dans la table d'outils, 86
- G10 L1 Etablissement coordonnees piece, 86
- G17 Plan XY, 87
- G18 Plan XZ, 87
- G19 Plan YZ, 87
- G2 Interpolation circulaire sens horaire, 84
- G20 Pouce, 87
- G21 Millimetre, 87
- G28, 87
- G28.1, 87
- G3 Interpolation circulaire anti-horaire, 84
- G30, 88
- G30.1, 88
- G33 MVT avec broche synchronisee, 88
- G33.1 Taraudage rigide, 88
- G38.2 Palpeur, 89
- G38.5 Palpeur, 89
- G4 Temporisation, 86
- G40 Revocation compensation rayon, 89
- G41 Compensation de rayon, 89
- G41.1 Compensation dynamique de rayon, 89
- G42 Compensation de rayon, 89
- G42.1 Compensation dynamique de rayon, 89
- G43 Compensation de longueur, 90
- G43.1 Compensation dynamique de longueur, 90
- G49 Revocation compensation longueur, 90, 91
- G53, 124
- G53 Mouvements en absolu, 91
- G54, 124, 126
- G54 Origine piece 1, 91
- G55, 124–126
- G55 Origine piece 2, 91
- G56, 124, 126
- G56 Origine piece 3, 91
- G57, 124, 126
- G57 Origine piece 4, 91
- G58, 124, 126
- G58 Origine piece 5, 91

- G59, [124](#), [126](#)
- G59 Origine piece 6, [91](#)
- G59.1, [124](#), [126](#)
- G59.1 Origine piece 7, [91](#)
- G59.2, [124](#), [126](#)
- G59.2 Origine piece 8, [91](#)
- G59.3, [124](#), [126](#)
- G59.3 Origine piece 9, [91](#)
- G61 Trajectoire exacte, [91](#)
- G61.1 Arret exact, [91](#)
- G64 Trajectoire continue, [91](#)
- G7 Mode diametre, [86](#)
- G73 Cycle perage brise copeaux, [92](#)
- G76 Cycle filetage multipasses, [92](#)
- G8 Mode rayon, [86](#)
- G80 Revocation codes modaux, [93](#)
- G81 Cycle de perage, [95](#)
- G82 Cycle de perage avec tempo, [96](#)
- G83 Cycle de perage avec debourrage, [96](#)
- G84 Cycle de taraudage, [97](#)
- G85 Cycle d'alesage, [97](#)
- G86 Cycle d'alesage, [97](#)
- G87 Alesage inverse, [97](#)
- G88 Cycle d'alesage, [97](#)
- G89 Cycle d'alesage, [98](#)
- G90 Mode déplacements absolus, [98](#)
- G91 Mode déplacements incrementaux, [98](#)
- G92, [98](#), [124](#), [126](#)
- G92.1, [98](#), [127](#)
- G92.2, [98](#), [127](#)
- G92.3, [98](#), [127](#)
- G93 Mode de vitesse, [99](#)
- G94 Mode de vitesse, [99](#)
- G95 Mode de vitesse, [99](#)
- G96 Vitesse de coupe constante, [99](#)
- G97 Vitesse de coupe en tr/mn, [99](#)
- G98 Retrait sur R, [100](#)
- G99 Retrait position initiale, [100](#)
- GNU-Linux, [5](#)
  
- HAL, [5](#)
  
- if, [107](#)
  
- KEYSTICK, [47](#)
- keystick, [6](#)
  
- lancer, [32](#)
- Linux, [5](#)
  
- M0 Arret de programme, [101](#)
- M1 Arret de programme, [101](#)
- M100..199 Mcodes utilisateur, [104](#)
- M2 Fin de programme, [101](#)
- M3 Broche sens horaire, [101](#)
- M30 Fin de programme, [101](#)
- M4 Broche sens anti-horaire, [101](#)
  
- M48 Controle correcteur de vitesse, [102](#)
- M49, [102](#)
- M5 Arret de broche, [101](#)
- M50 Controle correcteur travail, [103](#)
- M51 Controle correcteur broche, [103](#)
- M52 Controle vitesse adaptative, [103](#)
- M53 Controle coupure vitesse, [103](#)
- M6 Appel d'outil, [102](#)
- M60 Arret de programme, [101](#)
- M61 Ajuste numero outil, [103](#)
- M62, [103](#)
- M62 Controle sorties num, [103](#)
- M63, [103](#)
- M64, [103](#)
- M65, [103](#)
- M66, [104](#)
- M66 Controle entrees num, [104](#)
- M7 Goutelettes, [102](#)
- M8 Arrosage, [102](#)
- M9 Arret d'arrosages, [102](#)
- Manuel, [9](#), [31](#), [38](#), [45](#)
- Marche/Arret, [17](#)
- MDI, [9](#), [21](#), [30](#), [31](#), [40](#)
- mini, [6](#), [35](#)
- mode, [99](#)
  
- O Codes, [106](#)
- OpenGL, [15](#)
- Origine Machine, [29](#)
- Origine Piece, [29](#)
- ouvrir, [32](#)
  
- parametres, [57](#)
- parcours outil, [36](#)
- pas a pas, [32](#)
- pause, [32](#)
- point contrôlé, [52](#)
- Position: Absolue, [16](#)
- Position: Actuelle, [9](#), [16](#)
- Position: Commandee, [9](#)
- Position: Commandée, [16](#)
- Position: Machine, [9](#)
- Position: Relative, [9](#), [16](#)
- Precedence des operateurs, [63](#)
- Python, [15](#), [25](#)
  
- Repeat, [107](#)
- reprise, [32](#)
- return, [106](#)
- RS274NGC, [50](#), [60](#)
  
- Sherline, [35](#)
- sub, [106](#)
- systèmes de coordonnées, [58](#)
  
- Table du G Code, [83](#)
- Tcl, [29](#)

tempo, 53  
Tk, 15, 29  
tkemc, 6, 29  
Toucher, 16  
trace d'outil, 43  
  
Ubuntu, 5  
unites, 10, 53  
  
verifie, 32  
vitesse d'avance, 53  
Vitesse de jog, 23  
vitesse de jog, 30  
  
while, 107  
  
xemc, 6