



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Permission has been granted to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film.

The author (copyright owner) has reserved other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without his/her written permission.

L'autorisation a été accordée à la Bibliothèque nationale du Canada de microfilmer cette thèse et de prêter ou de vendre des exemplaires du film.

L'auteur (titulaire du droit d'auteur) se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation écrite.

ISBN 0-315-56450-4

**METHODE DE DIAGNOSTIC AUTOMATIQUE
POUR LES SYSTEMES D'ASSERVISSEMENT**

Par

JOHANNE SEVIGNY

**THESE PRESENTE A
L'ECOLE DES ETUDES SUPERIEURES
UNIVERSITE D'OTTAWA
EN ACCOMPLISSEMENT PARTIEL
DES EXIGENCES POUR LA
MAITRISE EN SCIENCES APPLIQUEES
AU
DEPARTEMENT DE GENIE MECANIQUE**

OTTAWA, ONTARIO, CANADA.

AOUT 1988

Johanne Sévigny, Ottawa, Canada, 1988



UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

SOMMAIRE

Ce rapport présente une méthode de diagnostic automatique pour un système asservis, s'apparentant au servomécanisme du robot ASEA IRB 6/2. Une méthode d'étude théorique des défaillances est développée et comparée à des résultats expérimentaux. La modélisation du servomécanisme pour une charge fixe, de même que les méthodes de détection et d'identification des anomalies développées sont présentées. L'implémentation de la procédure de diagnostic sur le montage expérimental est discutée et les résultats sont exposés. La conception et la construction du système asservis utilisé sont aussi présentées.

L'étude théorique des anomalies a démontré que l'utilisation d'un modèle linéaire permet de modéliser le fonctionnement normale et en défaillance du système de commande adéquatement. La procédure de diagnostic implémentée sur le montage expérimental s'est avérée apte à détecter et identifier les défaillances correctement pour les deux groupes d'anomalies étudiés.

REMERCIEMENTS

L'auteur désire remercier son directeur de thèse, Dr. Dan S. Necsulescu pour son support financier et scientifique tout au long de ce projet. Merci aussi à Solange Amyot et Mr. Seaman pour leurs conseils judicieux, de même qu'à Dr. A. Fahim pour l'équipement qu'il nous a laissé utiliser. Et un merci tout spécial à mon mari, Mario Tétreault, pour son support scientifique et morale, sa patience et ses encouragements. De plus, je doit remercier le Fonds FCAR de meme que OGS pour leurs supports financier.

TABLE DES MATIERES

	Pages
INTRODUCTION	1
 Chapitre I	
1.0 Les développements antérieurs en fiabilité et diagnostic automatique	5
1.1 Les méthodes classiques d'étude de la fiabilité	
1.1.1 La méthode FMEA	5
1.1.2 L'arbre de vérité	8
1.2 La détection automatique des anomalies	12
1.2.1 La méthode BIT	12
1.2.2 Les systèmes experts	14
1.2.3 Les systèmes d'asservissement tolérant les défaillances	17
1.3 La fiabilité des logiciels	17
1.4 Etude des effets du diagnostic automatique sur une chaîne de montage automatisé	18
1.4.1 La chaîne de montage	19
1.4.2 L'analyse	21
 Chapitre II	
2.0 La modélisation du système de commande	25
2.1 Le modèle de système de commande réel	25
2.1.1 Modèle dans le domaine temporel	27
2.1.2 Modèle dans le domaine Laplace	28
2.1.3 Modèle dans le domaine fréquentiel	30
2.2 L'implémentation du modèle	32
2.2.1 Les algorithmes de transformée de Fourier directe et inverse	33
2.3 Le modèle vs le système de commande réel	36
2.3.1 La procédure de test	36
i) Pour le système de commande réel ..	36
ii) Pour le modèle	37
2.3.2 Comparaisons des résultats de tests	37
i) Pour un échelon	37
ii) Pour une rampe	38
iii) Pour un sinus	38

2.4	Etude théorique et expérimentale des défaillances	39
2.4.1	La modélisation des défaillances	39
2.4.2	Le programme de simulation des défaillances	44
2.4.3	Modélisation des défaillances vs défaillances réels	45
i)	Défaillance de la partie proportionnelle du régulateur	45
ii)	Défaillance de la partie intégrale du régulateur	46
iii)	Défaillance de la partie différentielle du régulateur	47
iv)	Défaillance du module optoelectronique	48
v)	Défaillance du D/A	50
vi)	Défaillance du moteur	51
vii)	Défaillance du comparateur	53
viii)	Conclusion	53
2.5	Le modèle du système de commande réel pour une charge variable	59

Chapitre III

3.0	La détection et l'identification des anomalies	61
3.1	La détection des anomalies: La sélection des traits caractéristiques	62
3.1.1	La moyenne	63
3.1.2	La moyenne et la variance	64
3.1.3	La moyenne et la variance des séquences normalisées	65
3.2	L'identification (classification) des défaillances	66
3.2.1	La connaissance du système	68
3.2.2	Les règles d'identification	72
3.3	Implémentation du processus de diagnostic	73
3.4	Exemple des entrées et valeurs calculées par le programme de diagnostic pour une entrée sinusoïdale	78
3.5	Effet de la défaillance du tachymètre sur la méthode de diagnostic développée	83
3.6	Effet de la défaillance du régulateur sur la méthode de diagnostic développée	84

Chapitre IV

4.0	Conclusions et Recommandations	86
4.1	Conclusions	86
4.2	Recommandations	88
Annexe A	Les circuits utilisés pour le montage expérimental	92
Annexe B	Conception du régulateur PID	109
Annexe C	La programme de commande	128
Annexe D	Le programme de détection automatique des anomalies	154
Annexe E	La construction du montage expérimental	178
	Bibliographie	191

LISTE DES TABLEAUX

	Pages
Table 1.1 Effet de réduire T_d sur E et R_p	24
Table 2.1 Signification des variables sources, X_1 à X_{10} .	41
Table 2.2 Simulation des défaillances	43
Table 3.1 Règles d'identification	74
Table 3.2 Pour le segment 1: Entrée, sortie réelle et sortie du modèle	80
Table 3.3 Segment 1: La moyenne et la variance	80
Table 3.4 Pour le segment 2: Entrée, sortie réelle et sortie du modèle	81
Table 3.5 Segment 2: La moyenne et la variance	81
Table 3.6 Pour le segment 3: Sortie réelle du système .	82
Table A.1 Sortie analogique vs sortie digitale	95
Table A.2 Direction du mouvement	99
Table A.3 Coordination des sorties à la porte 2	100

LISTE DES FIGURES

	Pages
Fig. 1.1 Etapes de la méthode FMEA	6
Fig. 1.2 Circuit d'alimentation pour un moteur	9
Fig. 1.3 L'arbre de vérité pour le système de la fig. 1.2	10
Fig. 1.4 La chaîne de montage	20
Fig. 2.1 Schéma simplifié du processus de diagnostic ...	25
Fig. 2.2 Le système à modéliser	26
Fig. 2.3 Procédure de détection des anomalies dans le domaine temporel	28
Fig. 2.4 Procédure de détection des anomalies avec le modèle dans le domaine Laplace et la comparaison dans le domaine temporel	29
Fig. 2.5 Procédure de détection des anomalies avec le modèle et la comparaison dans le domaine Laplace	29
Fig. 2.6 Processus de diagnostic avec le modèle dans le domaine fréquentiel et la comparaison dans le domaine temporel	31
Fig. 2.7 Processus de diagnostic avec le modèle dans le domaine fréquentiel et la comparaison dans le domaine fréquentiel	32
Fig. 2.8 Séquence de signal échantillonné	33
Fig. 2.9 Transformée de Fourier de la figure 2.8 pour $N = 32$ et $T = 0.5$ sec	34
Fig. 2.10 Modèle vs système réel pour un échelon	37
Fig. 2.11 Modèle vs système réel pour une rampe	38
Fig. 2.12 Modèle vs système réel pour un sinus	38
Fig. 2.13 Le graphe de fluence du système de commande ...	42
Fig. 2.14 Le graphe de fluence modifié	42
Fig. 2.15 Structure du programme de simulation des défaillances	44

Fig. 2.16	Défaillance de la partie proportionnelle du régulateur	46
Fig. 2.17	Défaillance de la partie intégrale du régulateur	46
Fig. 2.18	Echelon utilisé pour la simulation des défaillances	47
Fig. 2.19	Simulation de la défaillance du module optoélectronique	48
Fig. 2.20	Défaillance de lecteur optique	49
Fig. 2.21	Simulation de la défaillance du D/A sur le montage expérimental	50
Fig. 2.22	Défaillance du D/A	51
Fig. 2.23	Simulation de la défaillance du moteur	51
Fig. 2.24	Temps de freinage non-nul	52
Fig. 2.25	Deux groupes de défaillances	54
Fig. 2.26	Défaillances du groupe 1	55
Fig. 2.27	Défaillances du groupe 2	56
Fig. 2.28	Défaillance du groupe 2 avec sortie non-nulle .	58
Fig. 2.29	Schéma fonctionnel du système pour une charge variable	59
Fig. 3.1	Schéma du processus de diagnostic	61
Fig. 3.2	Processus typique de reconnaissance des formes.	62
Fig. 3.3	Deux fonctions sinusoïdales	64
Fig. 3.4	Détection automatique des anomalies	66
Fig. 3.5	La défaillance du lecteur optique	67
Fig. 3.6	Les types d'entrées utilisées	68
Fig. 3.7	La tâche à accomplir	69
Fig. 3.8	La trajectoire dans l'espace cartésien	70
Fig. 3.9	La trajectoire angulaire pour chaque servomécanismes (4) du robot ASEA IRB 6/2	71

Fig. 3.10	Le processus de diagnostic complet	76
Fig. 3.11	Synchronisation du processus de diagnostic	77
Fig. 4.1	Compensateur de vitesse et position	90
Fig. A.1a	Photo du montage experimental.....	93
Fig. A.1b	Schema du montage experimental.....	94
Fig. A.2	Le circuit de conversion digital à analogique .	96
Fig. A.3	L'amplificateur de tension	97
Fig. A.4	L'amplificateur de courant	97
Fig. A.5	Entrées et sorties du Z8	98
Fig. A.6	Module optoélectronique	101
Fig. A.7	Sortie du module optoélectronique	101
Fig. A.8	Amplificateur de la sortie du module optoélectronique	102
Fig. A.9	Le détecteur de passage à zéro	102
Fig. A.10	La formation des ondes carrées	103
Fig. A.11	Conversion au standard TTL	103
Fig. A.12	Lecture et conditionnement du signal de position	104
Fig. A.13	Circuit de direction du mouvement	104
Fig. A.14	Sortie du tachymètre	105
Fig. A.15	Structure du système d'acquisition des données.	107
Fig. A.16	Effet de l'utilisation d'une seule source de tension sur la position obtenue	108
Fig. B.1	Le système à modeler	111
Fig. B.2	Schéma fonctionnel du système	111
Fig. B.3	Schéma fonctionnel du régulateur PID	112
Fig. B.4	La transmittance du D/A	112
Fig. B.5	Transmittance de l'amplificateur	113
Fig. B.6	Transmittance du moteur	113

Fig. B.7	La transmittance de la boucle de retour	114
Fig. B.8	Les subdivisions de $F(s)$	114
Fig. B.9	Graphe de fluence	115
Fig. B.10	Région ou les pôles doivent être situés	118
Fig. B.11	Effet de varier α_1	119
Fig. B.12	Effet de varier α_2	120
Fig. B.13	Effet de varier α_3	121
Fig. B.14	Variation simultanée des paramètres sur le lieu des pôles	122
Fig. B.15	Réponse indicielle	124
Fig. B.16	Equation numérique du régulateur	125
Fig. E.1	Schéma d'assemblage du montage expérimental ...	180
Fig. E.2	Schéma de construction de la base	181
Fig. E.3	Schéma de construction de la base du lecteur optique	182
Fig. E.4	Schéma de construction du support pour le roulement à billes	183
Fig. E.5	Schéma de construction du support du moteur ...	184
Fig. E.6	Schéma de construction du support du disque du lecteur optique	185
Fig. E.7	Schéma de construction du support de la charge	186
Fig. E.8	Schéma de construction du coupleur	187
Fig. E.9	Schéma de construction de l'arbre d'entraînement	188
Fig. E.10	Schéma de construction de la charge	189
Fig. E.11	Schéma de construction du disque du lecteur optique	190

NOMENCLATURE

A	Fonction de transfert de l'amplificateur dans le domaine Laplace
B_0	Coefficient de frottement visqueux
$D(s)$	Fonction de transfert du régulateur PID
D/A	Fonction de transfert du convertisseur D/A dans le domaine Laplace
E	Sortie du comparateur
E_p	Efficacité
EPSIL1	Seuil pour la moyenne
EPSIL2	Seuil pour la variance
ERR1	Erreur entre la moyenne du signal réel et modelé
ERR2	Erreur entre la variance du signal réel et modelé
F	Fréquence des arrêts due aux défaillances
$F(s)$	Fonction de transfert de la boucle de retour
$F(k)$	Transformée Fourier de $f(j)$
$f(j)$	Séquence échantillonnée d'une fonction $f(t)$
$f(t)$	Fonction continue dans le domaine temporel
$G(s)$	Fonction de transfert du modèle dans le domaine Laplace
$G^*(t)$	Séquence correspondant à la transformée de Laplace inverse de $G(s)$
$G_2(s)$	La variation de position causé par les perturbations de torque
H_d	Sortie de la partie différentielle du régulateur
H_i	Sortie de la partie intégrale du régulateur
H_p	Sortie de la partie proportionnelle du régulateur
i	Indique un nombre complexe, $i = \sqrt{-1}$
$I(t)$	Indicateur de mouvement
J_0	Inertie ramenée à l'arbre du moteur
K_a	Constante de l'amplificateur
K_c	Constante du compteur de position
K_d	Constante de la partie différentielle du régulateur
K_e	Coefficient de force contre électromotrice
K_f	Constante du lecteur de position
K_i	Constante de la partie intégrale du régulateur
K_p	Constante de la partie proportionnelle du régulateur
K_t	Coefficient de couple moteur
L_a	auto inductance de l'induit
M	Fonction de transfert du moteur dans le domaine Laplace
M_0	Moyenne échantillonnée
n	Coefficient de réduction de l'engrenage
N_s	Nombre de station dans la chaîne de montage
N	Longueur des séquences
p	Probabilité de défaillance
P_i	Partie imaginaire de la fonction de transfert du système dans le domaine fréquentiel
P_r	Partie réel de la fonction de transfert du système dans le domaine fréquentiel
R_a	Résistance de l'induit
R_c	Taux de production idéal
R_f	Résistance de retour pour l'amplificateur de tension
R_{in}	Résistance d'entrée de l'amplificateur de tension

R_p	Taux de production moyen
s	Opérateur de Laplace
t	temps
T	Période d'échantillonnage
T_c	Cycle de production
T_d	Temps d'arrêt due aux défaillances
$T_d(s)$	Perturbation de torque
T_l	torque total requis
T_m	Torque du moteur
U	Sortie du D/A
U^*	Sortie du régulateur PID
V	Sortie de l'amplificateur
V_a	Variance échantillonnée
V_{in}	Tension d'entrée
V_{out}	Tension de sortie
$X_1 - X_{10}$	Variables sources
Z_c	Impédance de capacité
θ_d	Signal de position désirée
θ_d^*	Signal de position désirée échantillonnée
θ_m^*	Signal de position obtenue mesurée par le lecteur optique
θ_M	Sortie du modèle.
θ_o^*	Signal de position obtenue échantillonnée
θ_o	Signal de position obtenue analogique
w	La fréquence

INTRODUCTION

Pourquoi ?

Pourquoi parler de détection automatique des anomalies? Pourquoi chercher à réduire le temps de réparation lorsqu'il y a une défaillance?

Pour des raisons de productivité, de profit. En effect, l'addition d'une méthode de détection automatique des anomalies à un système existant permet d'en améliorer la FIABILITE la DISPONIBILITE et la SECURITE.

Afin de rester compétitive les industries doivent de plus en plus utiliser des technologies avancées et des équipements sophistiqués (des robots par exemple). Ceux-ci demandent de gros investissements et étant souvent très compliqués, lorsqu'une défaillance survient le temps de réparation est long. Il est aussi parfois nécessaire de faire appel à un spécialiste. De plus on doit penser à la sécurité des travailleurs en contact avec ces équipements. Afin de faciliter la présente analyse on utilisera un système robot comme référence tout au long de ce rapport.

Des études de fiabilité ont été faites pour la firme Unimation. Les résultats prévoaient un MTBF (Mean Time Between Failure) de 500 heures pour les robots de la série 2000. En réalité lors d'expériences en milieu industriel, un MTBF de 145 heures a été obtenu pour les premières utilisations. Cet MTBF augmentait par la suite graduellement avec les années pour atteindre 415 heures [8]. D'après ce même auteur, la solution pour minimiser les temps d'arrêt duent aux défaillances (downtime) dans les usines automatisées est le diagnostic et la surveillance automatisé qui localisera les défaillances presque instantanément.

Lorsqu'un robot entre en défaillance son mouvement peut devenir erratique, mettant ainsi en danger la sécurité de l'opérateur et des équipements adjacents. Les données statistiques officielles suédoises démontrent que le taux moyen d'accident, causant des dommages corporels au personnel, est de, une accident par 45 robots par année [12]. Considérant qu'aux Etats-Unis et au Japon, en 1984 il y avait 77600 robots [12], on peut estimer la moyenne des accidents à 1724 accidents/année. Une usine d'assemblage automobile au Japon rapporte que les accidents les plus fréquentes sont dues à des bruits parasites, des problèmes avec des valves à pression et des servo-valves, des problèmes de capteurs, des problèmes de circuits et des mauvaises procédures d'opération par les opérateurs [12]. Des mesures de sécurité sont en général prises par les industries mais elles ne semblent pas suffir. Comme par exemple, le Rensselear Polytechnique Institute utilise des capteurs, et un système de surveillance informatisé qui vérifie continuellement si certains paramètres ne sont pas violés. Le Bureau National des Standards des Etats-Unis utilise un système informatique de sécurité pour les robots et leurs environnement. Un espace interdit est défini autour de l'espace de travail du robot. Si celui-ci pénètre dans cet espace l'ordinateur de sécurité prendra une action corrective [12]. Contrairement à ces méthodes, une procédure de diagnostic automatique n'attend pas que la situation soit critique pour agir, aussitôt que le robot dévit de sa course le diagnostic se met en marche. Les procédures de diagnostic automatique permettent de prévenir qu'une petite défaillance mineure se propage et cause des dommages majeures au robot et à son environnement.

Comme on vient de le voir, il y a plusieurs raisons pour s'intéresser au diagnostic automatique. Ici on a parlé seulement de robots industriels, mais les mêmes problèmes sont rencontrés avec d'autres pièces d'équipements sophistiqués. L'implémentation d'un processus de détection automatique des anomalies nous permettrait donc de déceler rapidement les défaillances et d'i-

dentifier rapidement le groupe de composantes fonctionnant anormalement. De plus, une action préventive pourrait être prise afin de minimiser les risques de collision avec l'environnement humain et matériel.

Comment ?

Ce rapport a pour but de présenter une méthode de détection automatique des anomalies pour un système de commande s'apparentant avec le servomécanisme du robot ASEA IRB 6/2 pour une charge fixe. Le chapitre premier discute de l'évolution des méthodes classiques d'étude de la fiabilité qui ont amené le développement des premières méthodes de diagnostic automatique. L'avènement des systèmes experts de même que le problème de la fiabilité des logiciels sont brièvement présentés. Pour terminer, une étude des effets de la défaillance d'un robot dans une ligne d'assemblage automatisé est discutée. On y présente l'effet des défaillances sur le taux de production de la chaîne justifiant ainsi le besoin d'un processus de détection préventive des anomalies.

Le chapitre deux présente en détail comment le système de commande a été modélisé et de quelle façon il a été implémenté dans la procédure de diagnostic. On y démontre la validité du modèle en le comparant aux résultats d'expériences sur un montage expérimental. De plus, une méthode d'étude théorique des défaillances est présentée.

La procédure de détection et d'identification des anomalies est discutée au chapitre trois. A partir de cette procédure et des résultats obtenus au chapitre deux le programme de détection automatique des anomalies (annexe D) a été développé et implémenté en parallèle avec le montage expérimental. On peut en observer la performance et discuter de sa pertinence.

Comme conclusion, le chapitre quatre expose les avantages et désavantages de la méthode développée par rapport aux méthodes existantes. Les implications de son implémentation sur un système à grande échelle sont examinées. Une série de recommandations sont formulées.

En parallèle avec le développement de la théorie, le montage expérimental a été construit. Celui-ci représente un système de commande pour un robot à contrôle continu. Tous les détails de la construction des circuits sont donnés dans l'annexe A. La conception du régulateur de type PID (Proportionnel, Intégral et différentiel) est expliquée dans l'annexe B. L'annexe C décrit le programme de commande du moteur, tandis que l'annexe E présente les détails de la construction du montage expérimental.

1.0 LES DEVELOPPEMENTS ANTERIEURS EN FIABILITE ET DIAGNOSTIC AUTOMATIQUE

1.1 Les méthodes classiques d'étude de la fiabilité

Le développement des méthodes classiques d'étude de la fiabilité des systèmes a eus son apogée avec l'avènement des centrales nucléaires, des bases de missiles, des industries de procédés chimiques, etc. Parce que ces entreprises mettaient en danger la vie des gens et de l'environnement, on a cherché à prévoir les défaillances qui pouvaient mener à des conséquences désastreuses. Suite à certains accidents fâcheux, les milieux gouvernementaux ont poussés la recherche dans ce domaine, qui n'a cessé de progresser depuis.

On peut donc définir le but des études de fiabilité et sécurité comme étant la réduction des possibilités de défaillance et des pertes humaines, économiques et environnementales. Ces études au départ devaient aider à la conception et l'analyse des systèmes et par la suite servir de fondement aux méthodes de détection automatique (ou partiellement automatique) des anomalies. Les deux méthodes les plus connues et utilisées seront décrites ici.

1.1.1 La méthode FMEA (Failure Mode and Effects Analysis)

Cette méthode se révèle un outil important, du point de vue de la fiabilité, lors de la conception de système. Elle a été développée au début des années 50 pour la conception et le développement de systèmes de commande pour les avions [7]. Depuis, elle est utilisée en milieu industriel, de recherche et de développement. La méthode se divise en 7 étapes majeures comme représenté à la figure 1.1.

Dans la première étape, le système est défini. Il est décomposé en sous-systèmes, qui ont des fonctions précises, et le lien entre chacun des sous-systèmes est fait.

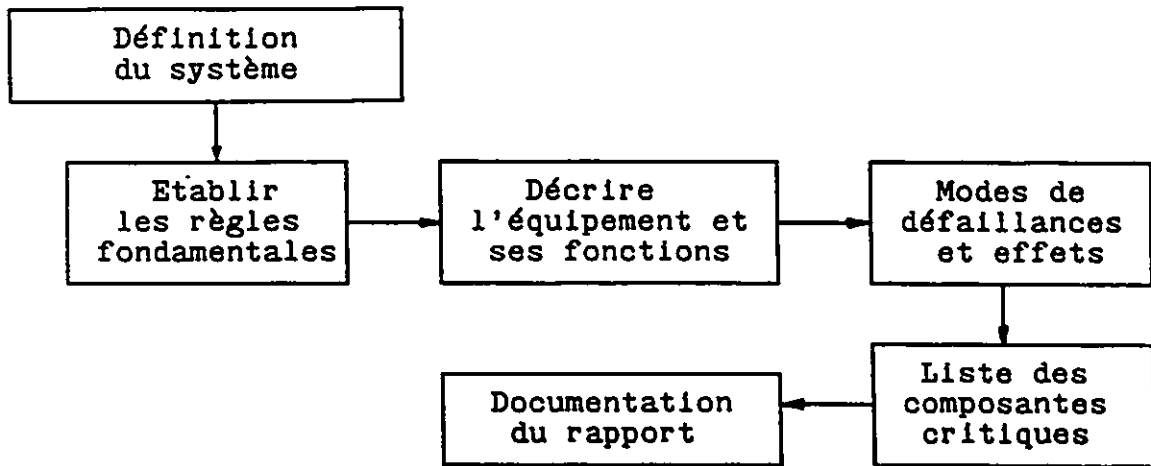


Figure 1.1 Etapes de la méthode FMEA.

L'analyste utilisera ici les dessins, spécifications et plus afin de bien définir le système.

Des règles fondamentales sont posées dans la deuxième étape. Ces règles peuvent comprendre:

- établir la limite des efforts en tension ou compression.
- établir les objectifs primaires et secondaires de l'étude.

La troisième étape est utilisée pour décrire l'équipement et ses fonctions. Elle est généralement divisée en deux parties.

1. La description de l'équipement: On montre graphiquement chaque élément du système à être analysé. Les entrées et les sorties de même que les configurations spécifiques (parallèle ou série) sont aussi représentées sur un schéma fonctionnel.
2. Déclaration des fonctions: La fonction de chaque composante est décrite pour tout les modes d'opérations.

La quatrième étape est directement reliée à l'identification des modes de défaillances et leurs effets. Les résultats de cette étape sont représentés sur un tableau.

On détermine par la suite la liste des composantes critiques qui pourra être utilisée pour le développement d'un processus de détection automatique des anomalies.

La dernière étape est la documentation du rapport.

La méthode décrite jusqu'ici est purement qualitative. Une autre méthode appelée FMECA (Failure Mode, Effect and Criticality analysis) est une combinaison de FMEA et d'une étude quantitative pour donner une valeur aux modes de défaillance considérant leurs probabilité d'arriver.

Cette méthode a été automatisée dans les années 1970. En 1977, Gilbert L. Barbour [18] a présenté une nouvelle formulation de la méthode en utilisant des matrices. Cette modification aura été la base utilisée pour automatiser le processus de façon efficace. La méthode permet de retracer la défaillance d'une composante à travers les sous-systèmes jusqu'au plus bas niveau, soit la pièce brisée. De plus, la nouvelle formulation de la méthode se prête bien à la programmation. La méthode a d'ailleurs été informatisée par Legg [19] en 1978. Le programme de Legg, qui avait certaine limitation du point de vue du nombre de défaillances qui pouvaient être définies a été modifié plus tard.

Goddard et Davis [20] ont amélioré le programme de Legg. Leurs technique d'automation avait pour but de réduire la charge de travail associée avec l'analyse, permettre à plusieurs analystes de travailler sur la même FMEA en provoquant le minimum de conflits et de faciliter l'utilisation des données sortant de la FMEA pour les ingénieurs appelés à les utiliser.

La FMEA a aussi été modifiée pour s'appliquer à des applications particulières comme les diagnostics des systèmes digitaux (Conley [21]). Herrin [22] a pour sa part renversé le processus de la FMEA. La sortie du procédé renversé permet d'identifier les niveaux d'ambiguïté pour l'isolation des défaillances, d'évaluer la qualité des points de test, de développer des tables d'entretiens, de valider les concepts d'entretien, etc.

Les descriptions précédentes sont quelques applications où FMEA a été utilisé, mais elle s'applique aussi dans d'autres domaines.

1.1.2 L'arbre de vérité (Fault tree).

La méthode des arbres de vérité a découlé quelque peu de la FMEA. La différence majeure entre les deux méthodes est que la FMEA est orientée vers les défaillances de composantes particulières tandis que l'arbre de vérité s'oriente vers des événements causant les défaillances. Elle a été développée par H.A. Watson des Laboratoire de Bell Téléphone en 1961-1962. Celui-ci travaillait sur le contrat des "Minuteman Launch Control System" pour le compte des forces armées [6]. La méthode a par la suite été raffinée par la compagnie Boeing.

La première étape dans la construction de l'arbre de vérité est d'identifier les événements indésirables associés avec le système. Ces événements sont appelés "Top Events". Par exemple, prenons un circuit d'alimentation pour un moteur (figure 1.2) [6]. Le "top event" est : le moteur ne démarre pas lorsque l'interrupteur est fermé. Une fois celui-ci identifié, on se demande qu'est ce qui pourrait causer cet événement indésirable. En répondant à cette question, d'autres événements sont générés et connectés par des opérateurs logiques AND, OR, XOR (ET, OU et OU EXCLUSIF).

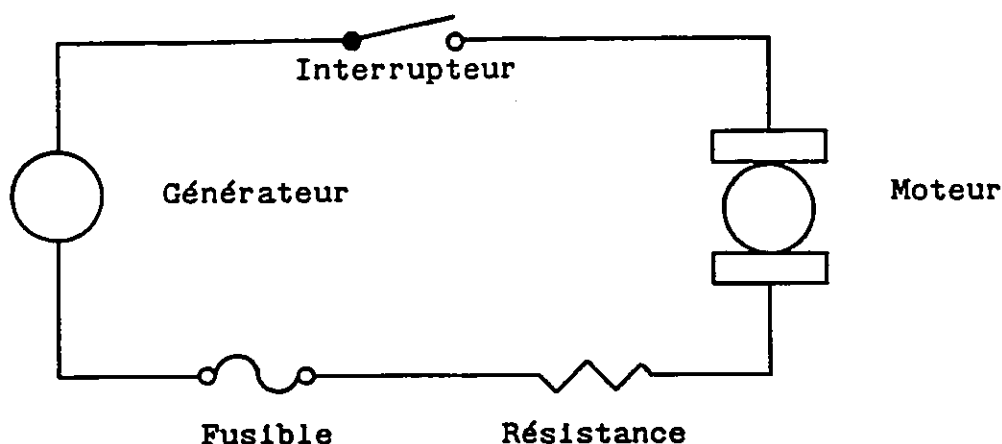


Figure 1.2: Circuit d'alimentation pour un moteur.

Tout les événements successifs sont ainsi développés. L'analyse est terminée quand les événements ne peuvent plus se réduire. A ce niveau on les appellent événements "primaires". La structure logique reliant le "top event" aux événements primaires est l'arbre de vérité. La figure 1.3 nous montre l'arbre de vérité pour le système d'alimentation du moteur.

Une fois construit l'arbre de vérité est utilisé pour trouver les chemins par lequel les événements primaires peuvent se propager à travers le système pour, enfin, causer le "top event". C'est à partir de cette connaissance des chemins (appelé "cut set") qu'une procédure de détection des anomalies peut être développée. Lidia Kiss dans son article [17] propose une méthode pratique d'identification des défaillances à partir des chemins. La méthode est basée sur une approche heuristique qui minimise le nombre de comparaison afin de trouver les défaillances.

La construction de l'arbre de vérité est souvent considérée comme un art plus qu'une science. En effet, deux analystes construiront rarement deux arbres identiques, même si ceux-ci sont fondamentalement équivalents. Par ailleurs, plus les systèmes étudiés sont compliqués et gros, plus la possibilité d'erreurs est élevée. C'est pourquoi plusieurs chercheurs ont voulu automatiser la construction des arbres.

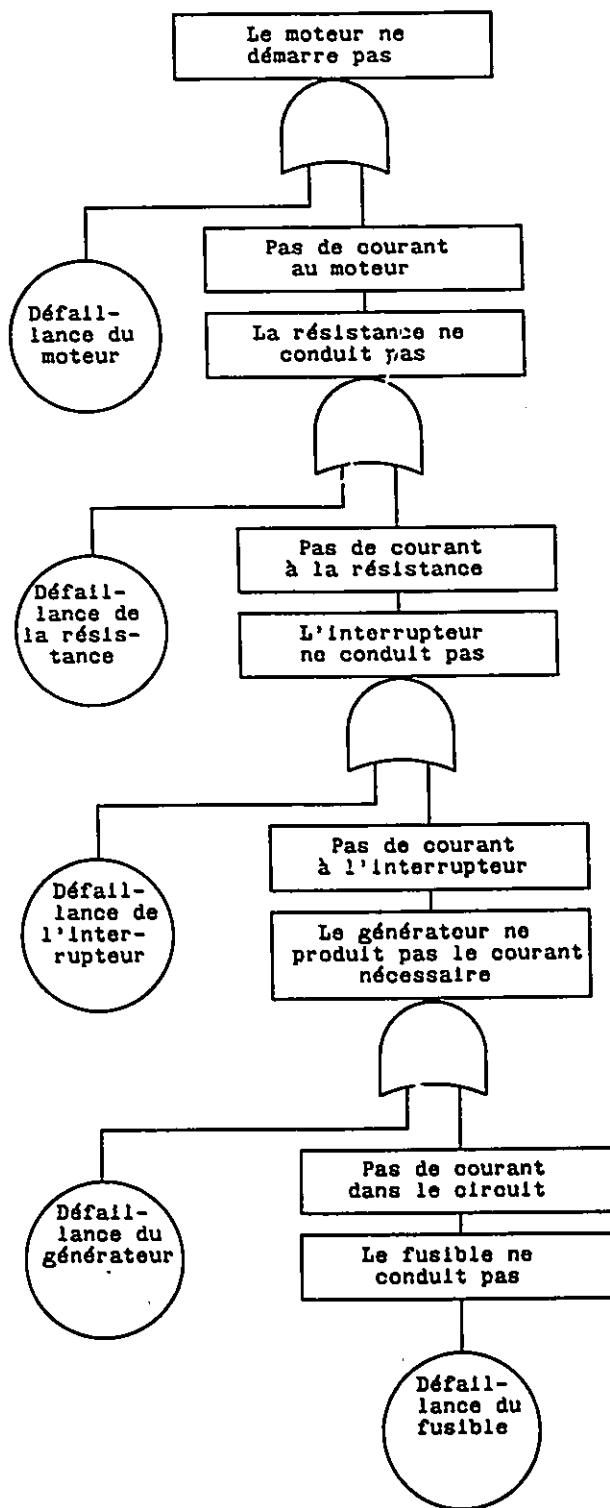


Figure 1.3: Arbre de vérité pour le système de la fig. 1.2

Lapp and Powers [13] ont développé un algorithme de synthèse des arbres de vérité basé sur les graphes dirigés (digraph). Ceux-ci représentent les relations normales, en défaillances et conditionnelles qui existent entre les événements (ou variables) pour un système.

Camarda, Corsi et Trentadue [14] ont conçu un algorithme pour la construction automatique des arbres de vérité fondé sur les graphiques de fiabilités (reliability graph).

Tompkins and Powers [15] ont suggéré une méthode utilisant des modèles d'entrée/sortie pour l'équipement. Ces modèles communiquent l'information relationnelle entre les variables quand le système fonctionne normalement et quand il y a défaillance d'une composante. La construction de l'arbre de vérité commence par l'identification de déviations importantes dans les variables du procédé. On cherche alors la source de ces déviations à partir desquelles l'arbre sera construit.

Les avantages de la méthode de l'arbre de vérité:

- Les défaillances multiples peuvent être étudiées.
- La représentation graphique nous procure un outil visuel important pour les ingénieurs et gestionnaires.
- Des études tant quantitatives que qualitatives sont possibles.
- Les programmes de construction automatique des arbres de vérité rendent l'utilisation des arbres facile, économique et efficace.
- A cause de sa structure d'arbre, la méthode est mieux organisée et facile à modifier que la FMEA.

Les désavantages de la méthode de l'arbre de vérité:

- L'arbre de vérité décrit le système à un temps donné (en régime permanent habituellement), de sorte que les événements séquentiels peuvent être difficilement représentés.

- Galluzzo et Andow [16] ont démontré clairement par de simples expériences en laboratoire sur un système de commande de type PID (Proportionnel, intégral et différentiel) que la méthode de l'arbre de vérité n'est pas appropriée quand on a des actions intégrales ou différentielles mais elle est bonne pour les actions proportionnelles seulement.
- Pour chaque "top event" un arbre de vérité différent est nécessaire.
- Il est difficile de considérer la dégradation des composantes. En effet, elles sont soit en bonne condition ou en défaillance.

1.2 La détection automatique des anomalies

Le diagnostic automatique est la suite logique des études de fiabilité classique. Ces études nous disent que la défaillance d'une composante "i" produira un effet "j" sur le système. Au niveau de la conception on se sert de ceci pour que le moins possible de défaillance provoque un effet désastreux en modifiant le système en conséquence. Au niveau du diagnostic on l'utilise pour savoir que: a)advenant l'effet "j" , quel sous-système ou composante est en défaillance; b)pour prendre une action corrective selon le cas.

Le diagnostic automatique est important car ça permet de réduire le temps moyen des réparations sur les chaînes de montage dans un milieu industriel. De plus, des techniciens moins bien qualifiés peuvent procéder au diagnostic. Différentes méthodes ont été développées afin de procéder à la détection automatique des anomalies. Les plus utilisées seront décrites brièvement ici.

1.2.1 La méthode BIT (Built In Test).

BIT a, au départ, été développée pour des applications militaires contenant principalement des systèmes électroniques, comme par exemple les systèmes de pilotage automatique. BIT est en fait une pièce d'équipement additionnel utilisée pour faire des

tests fonctionnels sur le système. BIT introduit des signaux connus dans le système à tester et vérifie que les signaux de sorties sont corrects. BIT peut nécessiter l'intervention d'un opérateur externe ou peut surveiller continuellement le système à interval régulier.

BIT permet d'augmenter la disponibilité de même que la confiance des utilisateurs dans le système. Mais par contre, BIT rend le système plus couteux et compliqué. Des capteurs additionnels sont nécessaires en plus des circuits de BIT et l'interface avec l'opérateur. Dans le cas de système numérique, BIT peut être presque entièrement implémenter par un logiciel nécessitant de la mémoire additionnelle.

BIT peut aussi provoquer un effet négatif sur la fiabilité d'un système en indiquant inadéquatement que celui-ci est défaillant. Ceci peut être causé par des défaillances dans les circuits de BIT (capteur, connecteur, etc). BIT doit donc être gardée le plus simple possible et être limitée à surveiller les fonctions essentielles du système qui ne peuvent pas être vérifiées autrement.

La façon de construire BIT est aussi très importante. De façon général BIT se base presque uniquement sur le principe de la redondance des composantes comme le montre les références [23] et [25] à [29]. La recherche se situe princialement au niveau de la réduction des informations nécessaires au diagnostic. En effet Peranskii [30] a démontré l'importance de réduire ces informations. Sheskin [27] a utilisé la programmation dynamique afin de générer une séquence de test à faire par BIT qui minimiserait le coût estimé des réparations.

Aly and Elsayedaly [28] cherchaient eux aussi à minimiser le coût de la procédure de diagnostic en réduisant la quantité d'information. Ils ont développé une approche hiérarchique. Ils sont arrivés aux conclusions que leur méthode pouvait être uti-

lisée efficacement pour isoler les défaillances. A cause des principes de dominance et de branchement qu'ils ont utilisés plusieurs solutions non-optimal ont pu être éliminées réduisant ainsi le temps et la quantité des informations à analyser. La limite de leur procédure est la grosseur du système qu'il peuvent étudier. La méthode s'applique uniquement à des modules électroniques. Cohn et Ott [29] ont présenté une méthode similaire.

Lidia Kiss [17] a pour sa part utilisé une approche totalement différente qui en fait ne s'applique pas uniquement à BIT mais à n'importe quel méthode. Elle utilise l'arbre de vérité pour déterminer tout les "cuts set" minimums du système. En étudiant ceux-ci il est possible de voir de façon heuristique qu'il n'est pas nécessaire de tester chaque événement primaire pour détecter une défaillance mais qu'il existe une hiérarchie qui réduit le nombre de tests grandement.

En conclusion, BIT est très utilisée même si son installation demande de l'équipement supplémentaire, donc des coûts en surplus. C'est une méthode efficace et qui se prête surtout bien au systèmes électroniques. La tendance générale est de limiter la quantité d'informations nécessaires au diagnostic afin de simplifier encore plus BIT. On cherche aussi à augmenter la part du logiciel dans le travail de BIT et de diminuer de plus en plus la quantité de pièces d'équipement nécessaires. Ce qui ramène le problème de la fiabilité de BIT à celle du logiciel.

1.2.2 Les systèmes experts.

Dans les dix dernières années, on observe un intérêt accru pour l'application des techniques d'Intelligence Artificiel (ci après appelé IA) au problème de diagnostic. En général un système expert comprendra une base de connaissance (base de donnée) qui résume ce qui est connu du système. De cette connaissance générale le système est simulé et le système expert utilisera le modèle du système analysé pour diagnostiquer les défaillances

avec l'aide de règles d'inférences. On appelle cette approche, basée sur un modèle, la connaissance profonde ([31], [32]). Le système expert emploie aussi la connaissance superficielle pour détecter une défaillance directement associée à un symptôme; cette approche est basée directement sur la connaissance des experts qu'on a recueillie et placée dans la base de connaissance. Plusieurs publications discutent de l'utilisation de IA en médecine, génie nucléaire et chimique, etc. Une idée des applications et de la façon dont IA est appliquée au diagnostic sera brièvement exposée.

Narayanan et Viswanadham [33] étudient le problème de modélisation de la connaissance du comportement d'un système en défaillance et utilisent ce modèle pour détecter les défaillances et leurs sources. Leur solution est basée sur l'arbre de vérité et les "digraphs" auxquels on applique des règles de raisonnement. L'analyse se divise en deux parties, la détection d'une défaillance et la localisation de la source de la défaillance. Un modèle pour chaque partie est développé. Les deux modèles sont par la suite combinés pour former une représentation hiérarchique des connaissances des défaillances. La méthode combine donc les méthodes classiques (fault tree) avec des techniques d'inférence de IA.

Lamb et al. [34] présentent le développement d'un système expert pour la détection automatique des anomalies dans une usine de procédés chimiques. Les auteurs montrent la nécessité de définir les limites du processus de diagnostic. Le système expert qu'ils ont développé combine les informations quantitatives contenues dans le modèle mathématique avec les informations qualitatives pour former des règles heuristiques.

Moore et Kramer [35] présentent un système expert qui supervise le contrôle en temps réel d'un procédé chimique en environnement dynamique. En comparant les approches de connaissance profonde et superficielle pour le processus de diagnostic ils

montrent que les techniques basées sur le raisonnement à partir d'un modèle sont plus appropriées que les méthodes heuristiques. Ils résument les techniques de diagnostic, basées sur un modèle, qui peuvent être utilisées comme la base de la connaissance profonde d'un système expert.

Atwood et Radlinski [36] travaillent au développement d'un système expert à l'intérieur duquel la structure des données et les mécanismes d'inférences sont basés sur l'approche utilisée par les experts. Il discute aussi du fait que le diagnostic de système résistant au défaillance (fault-tolerant) est très difficile et demande plusieurs sources de connaissance. C'est là que l'expertise humaine apporte un modèle pour développer une architecture qui incorpore toute ces sources de connaissance. Ils appliquent leurs principes au diagnostic des défaillances sur un système stéréo.

Thandasseri [37] discute du développement d'un système expert pour la British Telecom en Angleterre. La conception du système expert a commencée par la cueillette de la connaissance auprès des experts et l'introduction de cette connaissance dans la base de la connaissance. Les mécanismes d'inférences ont par la suite été déterminés. Le but du système expert était de pouvoir localiser et compenser pour 70% des défaillances. Le produit finit s'est montré capable de détecter et localiser 85% des bris. De plus, il fait remarqué que le système peu maintenant être opéré par un ingénieur sans expérience.

On peut conclure que l'approche de la connaissance profonde (technique basée sur un modèle) est nécessaire à la description du processus qu'on désire surveiller avec un système expert. La connaissance superficielle doit aussi être présente pour deux raisons majeures. Elle peut simplifier le processus de diagnostic et certaines connaissances peuvent être représentées seulement par des relations empiriques entre les symptômes et les causes.

1.2.3 Les systèmes d'asservissement tolérant les défaillances

Comme il a été mentionné précédemment, lorsqu'un robot entre en défaillance son mouvement peut devenir imprévisible et ainsi provoquer des accidents avec l'environnement. Dans cet ordre d'idées Joshi [61] a développé un système de commande tolérant les défaillances. Il a étudié la stabilité de systèmes de commande de type LQG (Linéaire Quadratique Gaussian) à boucle multiple en présence de défaillance des actionneurs ou des capteurs. Dans le cas de boucles de retour d'état linéaire quadratique, il a été prouvé que la stabilité de la boucle fermée pouvait être maintenue, dans la cas de certaines défaillances de l'actionneur, par l'addition de gains appropriés dans la boucle de commande. Par le principe de dualité, il a aussi appliqué ces résultats à la conception d'estimateurs d'état qui maintiendrait la stabilité lors de la défaillance de capteurs, à la condition que l'état de défaillance soit connu ou détecté. La combinaison de ces deux faits rend un système de commande de type LQG tolérant pour certaine défaillance de l'actionneur ou des capteurs.

1.3 La fiabilité des logiciels.

Avec l'avènement des systèmes experts et des systèmes numériques on doit se poser la question à savoir, " les logiciels sont-ils à l'abri des défaillances?" Le problème majeur dans la question des logiciels est que la complexité des tâches qu'ils doivent accomplir a grandie plus vite que la technologie pour les concevoir, tester et gérer. La recherche en fiabilité des logiciels a commencé au environ de l'année 1968. Plusieurs modèles probabilistiques ont été développés afin d'évaluer le temps moyen entre les erreurs de logiciel (MTBSE) et de trouver des fonctions de fiabilité pour les logiciels ([38] à [45]). La présente discussion est simplement une observation que les systèmes numériques aident à résoudre bien des problèmes d'une façon simple et relativement peu coûteuse à long terme.

Mais le fait de convertir les systèmes traditionnels en leurs contreparties numériques amène aussi d'autres problèmes qui sont souvent négligés. Le domaine de la fiabilité des logiciels est relativement nouveau et les développements futurs risquent d'être très intéressants.

1.4 Etude des effets du diagnostic automatique sur une chaîne de montage.

Afin de rester compétitive les industries doivent fabriquer des produits fiables et à la fine pointe de la technologie. Pour répondre à cette demande et maintenir une croissance constante, les fabricants changent graduellement leurs philosophies de production. Ils doivent être capable de fabriquer des produits de haute qualité à des coûts moindre et avec un cycle de production faible. Cet objectif peut être atteint par l'élimination des incertitudes intrinsèques reliées aux opérations manuelles et par l'automatisation des procédés de fabrication. Ceci est la raison majeure pourquoi les industries vont vers les chaînes de montage automatisées et les cellules flexibles de fabrication.

Les composantes typiques d'une chaîne de montage automatisée sont: les robots pour l'assemblage ou la manutention des pièces, les convoyeurs, véhicules guidés automatiquement, les entrepôts automatisés, les distributeurs de pièces automatique, les palettes. etc, (pour les chaînes de fabrication on aura en plus les machines-outils). Toutes ces composantes sont reliées à un ordinateur principal qui contrôle la production. Qu'arrive t'il lorsque un de ces éléments entre en défaillance? Quels sont les effets sur la production?

Une enquête a été faite, dans le Royaume Unis, à l'intérieur de 3 compagnies comprenant 37 robots pour différentes applications. Dans ces compagnies on a noté que le temps d'arrêt moyen due aux défaillances était de 24.9% du temps de production. C'est-à-dire que le quart du temps de production est perdu en réparation et autres problèmes reliés aux défaillances.

Considérant tout les incidents dans les trois compagnies ils ont pus affirmer que 23.4% des cas de défaillances et que 21.4% des temps d'arrêt dueent aux défaillances étaient directement reliés à des problèmes avec les robots [12]. Afin de voir clairement l'importance des défaillances, un exemple simple est présenté.

1.4.1 La chaîne de montage.

Pour l'exemple, on va utiliser une des chaînes d'assemblage automatique pour une usine d'imprimante japonaise, "Oki Electric Industry Company, Ltd." [46]. La chaîne, tel que montrée à la figure 1.4, se compose de 7 robots d'assemblage, d'un entrepôt de palettes et d'un convoyeur. La chaîne est alimentée par des véhicules guidés qui transportent les palettes qui entrent et qui sortent et qui fournissent chaque stations en pièces.

Afin de simplifier l'analyse, on va assumer que seul les robots peuvent entrer en défaillances. La théorie présentée par Groover [9] chapitre 5 sera utilisée pour l'étude. Les données suivante seront nécessaire à l'analyse.

Données:- Cycle de production (T_C) = 0,5 min.,
(tel que donnée dans [46])

- Probabilité de défaillance (p) = $0,01 \frac{\text{bris}}{\text{station.piece}}$

(toutes les stations ont la même probabilité)

- Temps d'arrêt due (T_d) = 10 min/bris
aux défaillances

- Nombre de station (M) = 7 stations
dans la chaîne

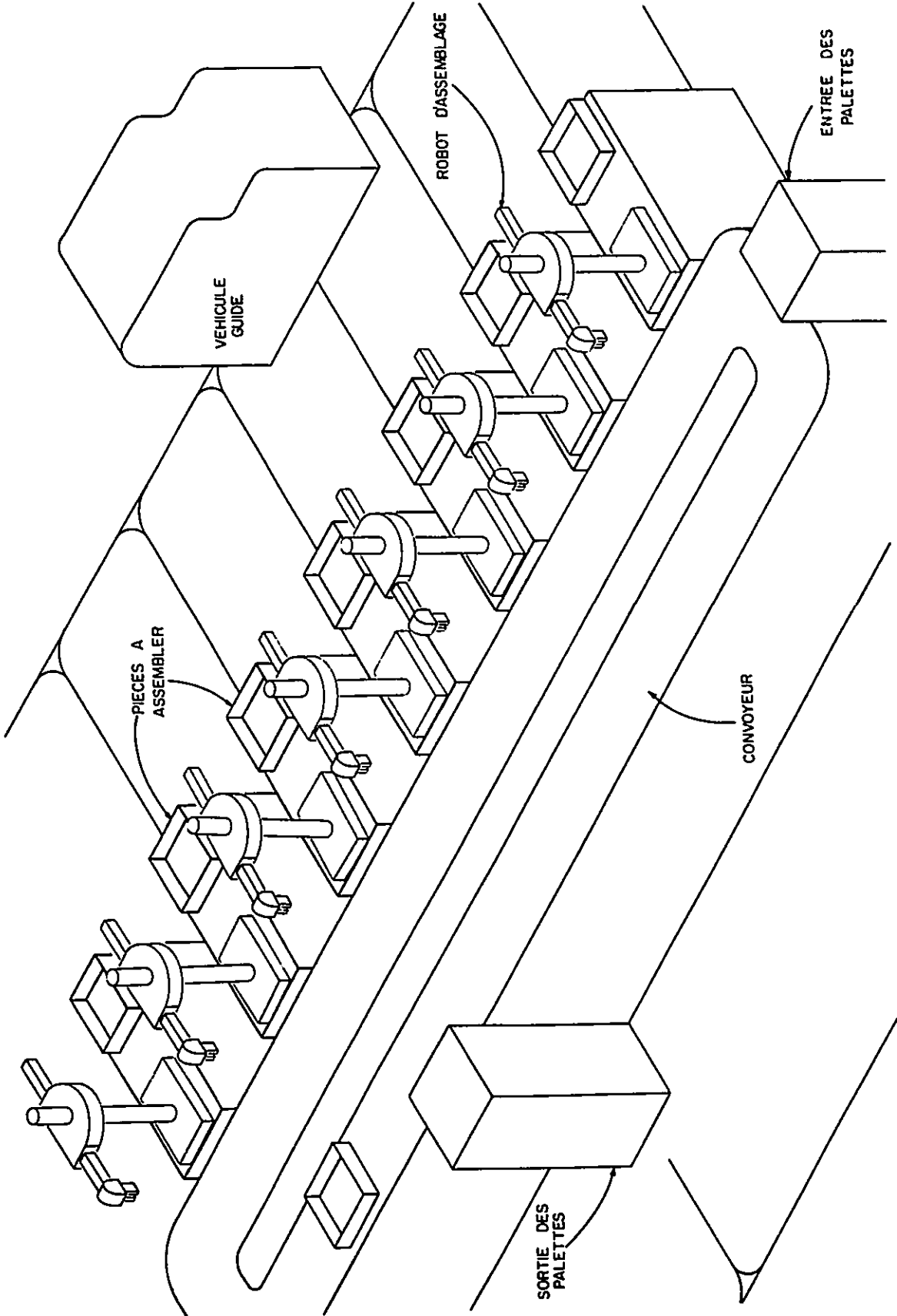


Figure 1.4: La chaîne de montage.

Où " T_C " représente le temps passé par une palette sur la chaîne d'assemblage, " p " est la probabilité que un des robots entre en défaillance pour chaque palette et " T_d " est le temps entre le moment ou le robot entre en défaillance et sa remise en fonction. Cette période inclus le temps nécessaire pour localiser la défaillance et le temps de réparation.

1.4.2 L'Analyse

On cherche: i) La fréquence des arrêts due aux défaillances (F), c'est-à-dire le nombre de bris probable par palette qui passe.

ii) Le taux de production moyen (R_p), c'est-à-dire la quantité de pièce fabriquée par minute, compte tenu des défaillances.

iii) L'efficacité de la chaîne de montage (E), qui représente la proportion du temps ou la chaîne produit.

i) La fréquence des arrêts due aux défaillances (F).

En assumant que lorsqu'il y a défaillance du robot la pièce à travailler ne sera pas abimé (upper bound approach) on obtient l'équation suivante,

$$F = M_p \quad (1.1)$$

$$F = 7 \text{ stations} \times 0,01 \text{ bris/stations.pièce}$$

$F = 0,07 \text{ bris/pièce}$

ii) Le taux de production moyen (R_p).

Une première équation nous donne le temps de production moyen en minute par pièce.

$$T_p = T_c + F \cdot T_d \quad (1.2)$$

$$T_p = 0,5 \text{ min/pièce} + 0,07 \text{ bris/pièce} \times 10 \text{ min/bris}$$

$$T_p = 1,2 \text{ min/pièce}$$

Le taux de production moyen est l'inverse du temps de production.

$$R_p = 1/T_p \quad (1.3)$$

$$R_p = 0,83 \text{ pièce/min.}$$

En assumant que l'usine fonctionne 24 heures/jour et 365 jours par année on aura un taux de production annuel moyen de

$R_p = 438 \text{ 000 pièce/année}$

Si on compare au taux de production idéal, " R_c ", qui représente le nombre de pièces fabriquées par minute s'il n'y a aucune défaillances, alors on obtient;

$$R_c = 1/T_c \quad (1.4)$$

$$R_c = 1 \text{ 051 200 pièce/année}$$

On remarque que le taux moyen R_p est loin de la production idéale.

iii) L'efficacité de la chaîne de montage (E).

$$E = R_p / R_c \quad (1.5)$$

$E = 42\%$

C'est à dire que 58% du temps, la chaîne de montage est arrêtée à cause de la défaillance de l'un des 7 robots.

De quelle façon peut-on améliorer la performance du système? Pour ce faire on doit diminuer le temps moyen de production, T_p . Trois possibilités nous sont offertes si on observe l'équation 1.2 : réduire T_c , F ou T_d . Le cycle de production peut difficilement être réduit, car il dépend de l'équipement utilisé. Il reste donc F et T_d . La fréquence des bris/pièce, F , et le temps d'arrêt due aux défaillances peuvent être diminués par l'implémentation d'un processus de détection automatique des anomalies tel que développé dans le présent rapport. Assumons, pour le moment, qu'on diminue T_d de moitié. C'est à dire,

$$T_d = 5 \text{ min/bris}$$

Si on refait la même analyse on trouvera que

$$R_p = 618\ 353 \text{ pièce/année}$$

$$E = 59\%$$

Les résultats des deux études sont représentés sur le tableau comparatif 1.1. Les effets de la réduction de T_d sont clairs. La productivité d'une chaîne de montage peut être augmentée grandement par l'implémentation d'un processus de détection automatique des anomalies. Dans l'exemple ci-haut seule la défaillance des robots était considérée, mais en réalité il peut aussi y avoir défaillance dans les autres composantes de la chaîne ce qui réduirait encore plus l'efficacité.

Table 1.1 Effet de réduire T_d sur E et R_p .

	Efficacité E (%)	Taux de production R_p (pièce/année)
$T_d = 10$ min/bris	42	438 000
$T_d = 5$ min/bris	59	618 353
Idéalement ($T_d = 0$)	100	1 051 200

Le développement d'une méthode de détection automatique des anomalies est donc justifié. La méthode qui va être développée au chapitre 2 et 3 permettra de diminuer le temps d'arrêt due aux défaillances, T_d , et aussi de réduire le fréquence des bris, F.

2.0 LA MODELISATION DU SYSTEME DE COMMANDE

Le processus de diagnostic se divise en trois parties; la modélisation du système réel, la détection et la localisation des défaillances. Ce processus est illustré de façon simplifiée à la figure 2.1.

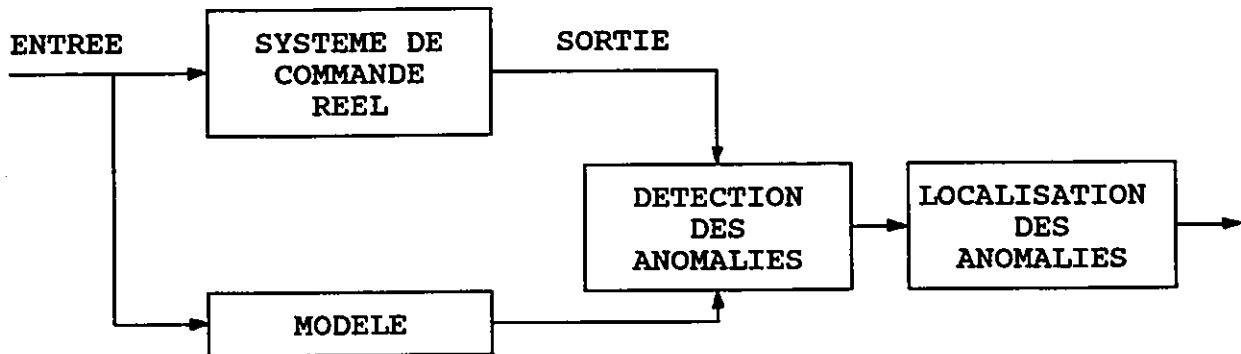


Figure 2.1: Schéma simplifié du processus de diagnostic.

L'entrée du système réel est envoyée à un modèle. La sortie du système de commande et du modèle sont par la suite comparées pour vérifier s'il y a défaillance ou non. Si oui, on passe à la localisation des anomalies.

Ce chapitre présente la méthode de modélisation du système réel et en montre la validité. De plus, une méthode d'étude théorique des défaillances sera présentée et comparée au système de commande lorsqu'il est en défaillance. La façon d'implémenter le modèle sera aussi discutée. La détection et la localisation, des défaillances, pour leurs parts, seront étudiées dans le chapitre trois.

2.1 Le modèle du système de commande réel.

La figure 2.2 nous montre un schéma du système à modéliser. Ce système de commande s'apparentent au servomécanisme du robot ASEA IRB 6/2.

Une charge constante est considérée afin de simplifier l'implémentation; les effets d'avoir une charge variable seront discutés à la section 2.5 du présent chapitre.

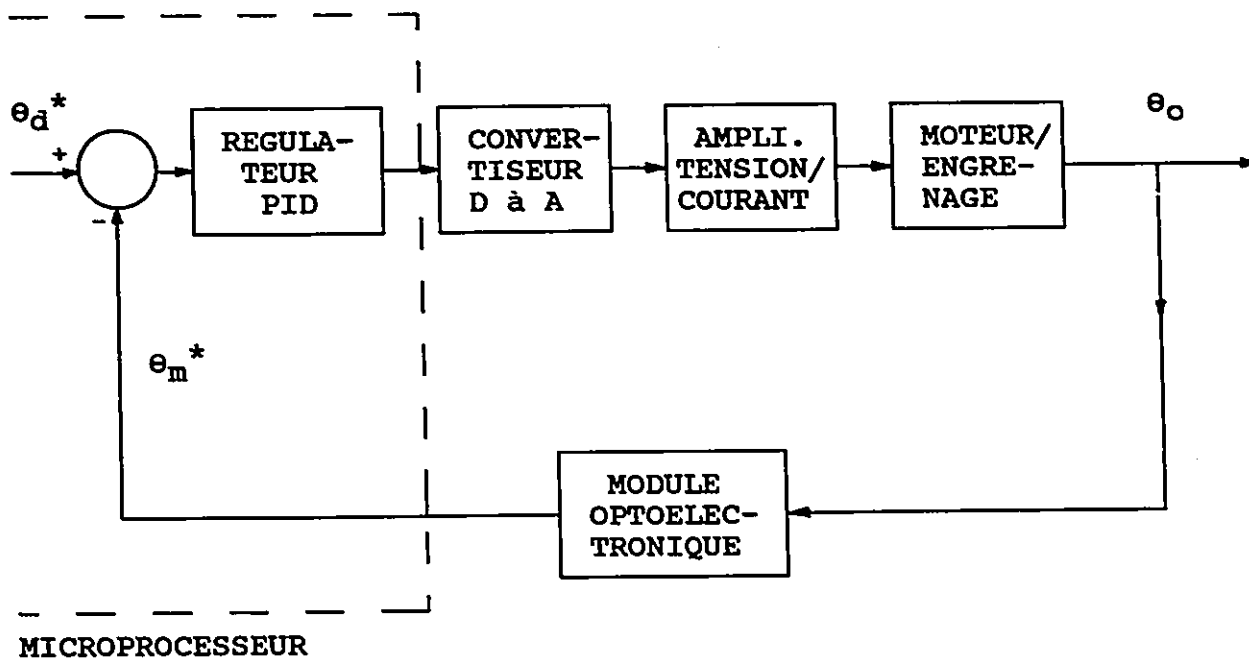


Figure 2.2: Le système à modéliser

Tous les détails de la construction du système et la conception du montage sont disponibles dans l'annexe A, B et E respectivement.

Le modèle, pour le système de la figure 2.2 a été conçu à partir de sa fonction de transfert, dans le domaine de Laplace, tel que dérivée dans l'annexe B, équation B22. On peut exprimer celle-ci comme étant,

$$\frac{\theta_M(s)}{\theta_d(s)} = \frac{22.4 s^2 + 2000 s + 2000}{s^3 + 150 s^2 + 2000 s + 2000} = G(s) \quad (2.1)$$

où "s" est l'opérateur de Laplace
 θ_M est le signal de sortie du modèle
 θ_d est le signal d'entrée du modèle et du système réel

La sortie du modèle, θ_M , s'écrit donc,

$$\theta_M(s) = \theta_d(s) \cdot \left[\frac{22.4 s^2 + 2000 s + 2000}{s^3 + 150 s^2 + 2000 s + 2000} \right] \quad (2.2)$$

ou

$$\theta_M(s) = \theta_d(s) \cdot G(s) \quad (2.3)$$

Pour le système de commande réel, l'entrée et la sortie sont des fonctions non-continues du temps. Ce sont des séquences de longueur fini, "N", chaque valeurs correspondant à la période d'échantillonnage en cours. Afin de pouvoir comparer la sortie du modèle et du système réel, un domaine de référence commun doit être utilisé. Trois façons de procéder ont été étudiées.

2.1.1 Modèle dans le domaine temporel.

En exprimant l'équation (2.3) dans le domaine temporel il est possible de faire toute les opérations dans ce même domaine en utilisant la convolution discrète.

$$\theta_M^*(t) = \sum_{\tau=0}^{N-1} \theta_d^*(t) \cdot G^*(t-\tau) \quad (2.4)$$

où $\theta_M^*(t)$ représente la séquence de sortie du modèle.

$\theta_d^*(t)$ représente la séquence d'entrée du système réel et du modèle.

$G^*(t)$ représente la séquence correspondant à la transformée Laplace inverse de $G(s)$, $G(t)$, évaluée a des intervals de temps équivalents à la période d'échantillonnage.

$$G^*(t) = G(KT), \quad K = i, \dots, i+N$$

N = longueur des séquences

T = période d'échantillonnage

On peut résumer la procédure de détection des anomalies dans le domaine temporel par le schéma de la figure 2.3. Cette procédure est simple et requière peu de calcul.

Le problème majeur est que, pour des systèmes très compliqués, comme les robots, il est pratiquement impossible de calculer la transformée de Laplace inverse de $G(s)$ pour obtenir $G(t)$.

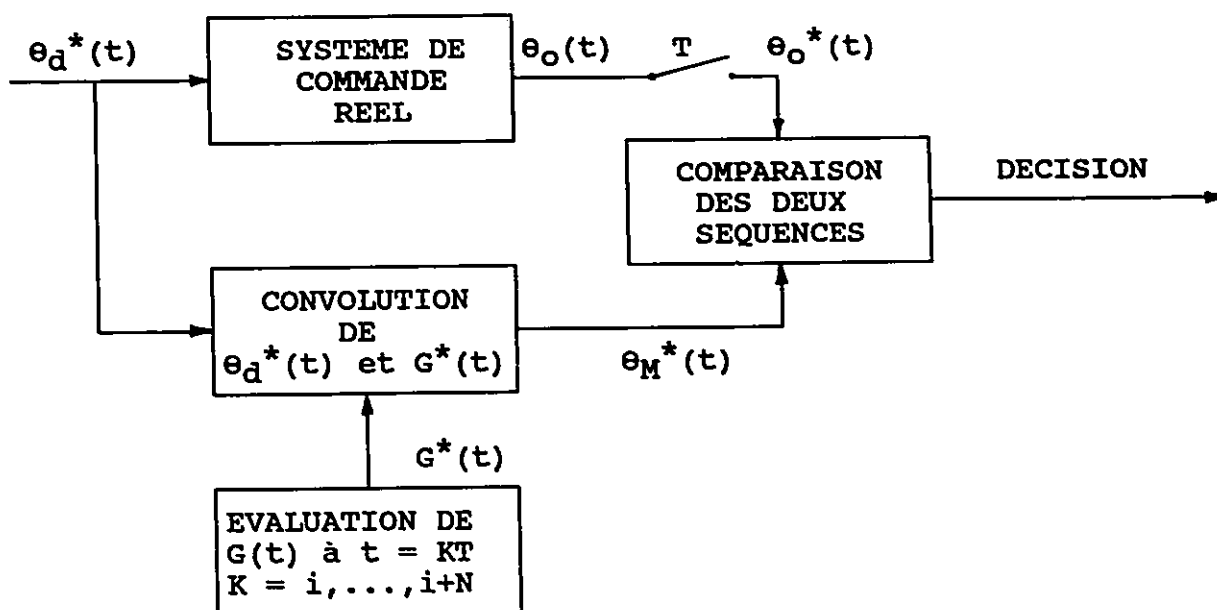


Figure 2.3: Procédure de détection des anomalies dans le domaine temporel.

2.1.2 Modèle dans le domaine Laplace.

La méthode la plus simple pour résoudre le problème mentionné plus haut est de conserver le modèle dans le domaine de Laplace et d'utiliser la transformée de Laplace numérique pour convertir la séquence d'entrée. La figure 2.4 nous montre la procédure qui pourrait être utilisée. La séquence d'entrée est convertie numériquement dans le domaine de Laplace. L'équation du modèle est évaluée dans "s" et la sortie du modèle est retransformée dans le domaine temps par la transformation Laplace inverse. La comparaison de la sortie du modèle et celle du système réel se fait dans le domaine temporel. Il serait aussi possible de faire la comparaison des deux séquences dans le domaine Laplace tel qu'illustré à la figure 2.5.

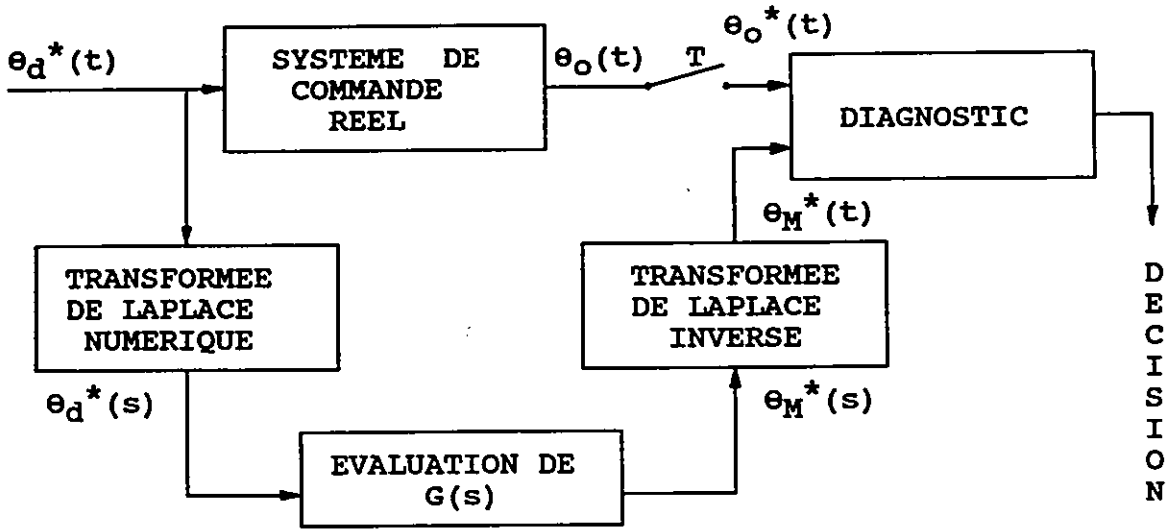


Figure 2.4: Procédure de détection des anomalies avec le modèle dans le domaine Laplace et la comparaison dans le domaine temporel.

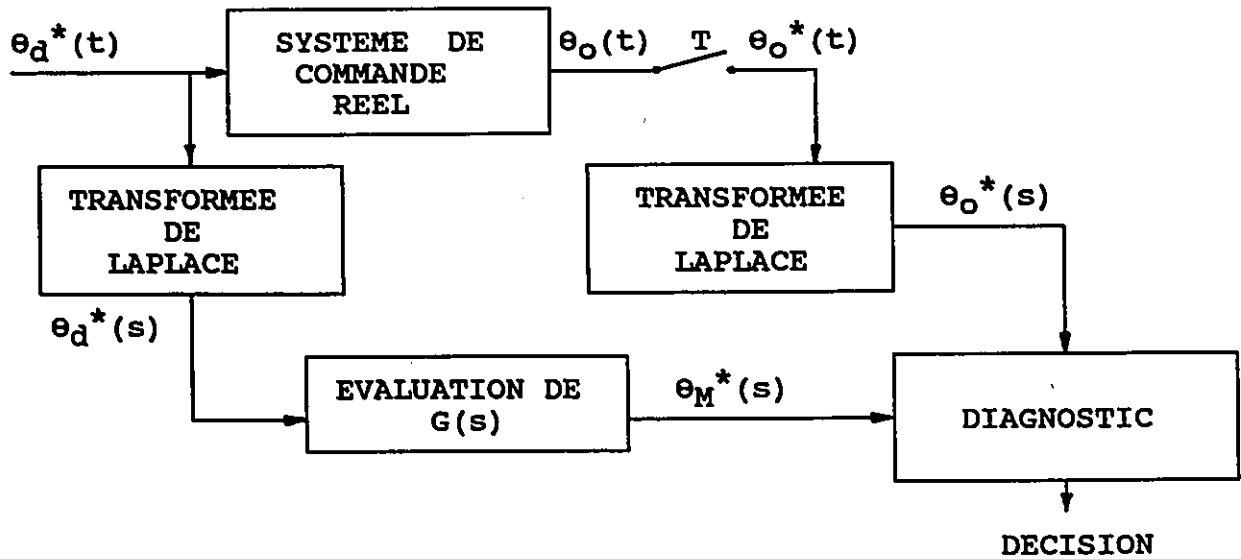


Figure 2.5: Procédure de détection des anomalies avec le modèle et la comparaison dans le domaine Laplace.

2.1.3 Modèle dans le domaine fréquentiel.

La troisième façon de procéder est d'utiliser la transformée de Fourier. Cette méthode est légèrement plus compliquée que celle présentée à la section précédente mais peut offrir certains avantages, comme il sera expliqué plus loin. Le passage du domaine Laplace au domaine fréquentiel requiert des modifications mineures à la fonction de transfert $G(s)$. En effet, il suffit de poser,

$$s = iw$$

où "s" est l'opérateur de Laplace
 i est $\sqrt{-1}$
 w est la fréquence

pour que l'équation 2.2 devienne,

$$\theta_M(iw) = \theta_d(iw) \cdot \left[\frac{22.4(iw)^2 + 2000(iw) + 2000}{(iw)^3 + 150(iw)^2 + 2000(iw) + 2000} \right] \quad (2.5)$$

Cette équation peut être exprimée en deux parties, une partie réelle et une partie imaginaire.

$$\theta_M(iw) = \theta_d(iw) \cdot [P_R + i.P_i] \quad (2.6)$$

où

$$P_R = \frac{4 \times 10^6 + 3,6552 \times 10^6 \cdot w^2 + 1360 \cdot w^4}{4 \times 10^6 + 3,4 \times 10^6 \cdot w^2 - 3977,5 \cdot w^4 + w^6} \quad (2.7)$$

$$P_i = \frac{-22,5 \cdot w^5 - 253200 \cdot w^3}{4 \times 10^6 + 3,4 \times 10^6 \cdot w^2 - 3977,5 \cdot w^4 + w^6} \quad (2.8)$$

En utilisant la fonction de transfert dans le domaine fréquentiel le processus de diagnostic de la figure 2.3 doit être modifié en conséquence.

Les figures 2.6 et 2.7 nous montre deux procédures possible. Rappelons que les signaux d'entrée et de sortie sont des séquences.

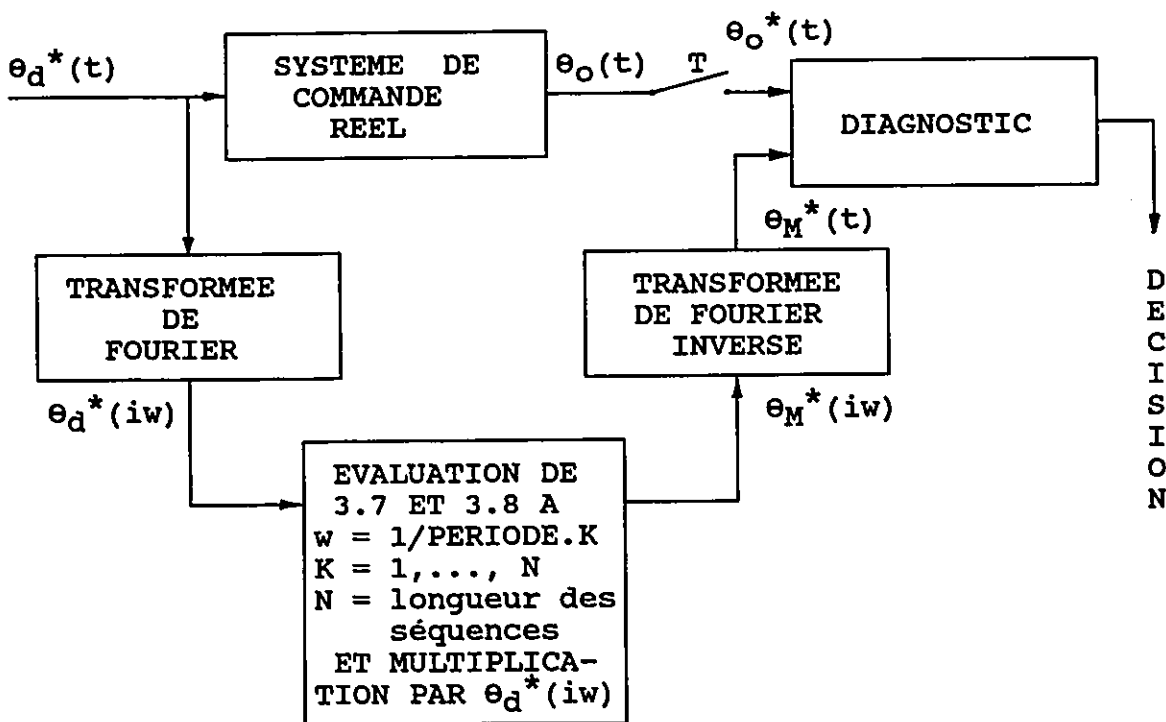


Figure 2.6: Processus de diagnostic avec le modèle dans le domaine fréquentiel et la comparaison dans le domaine temporel

L'étude dans le domaine fréquentiel présente des avantages certains sur celle dans le domaine temporel. Même pour des systèmes très compliqués il est relativement facile d'obtenir la fonction de transfert, $G(iw)$. Comparativement à la méthode dans le domaine Laplace les transformées de Fourier possède certaines propriétés qui pourront être utilisées pour faciliter le diagnostic et l'identification. Les transformées de Fourier sont d'ailleurs fréquemment utilisées en reconnaissance de la parole. Il a d'abord été choisie d'utiliser la méthode de la figure 2.6 pour procéder à l'implémentation parce qu'elle était plus facile à utiliser et les résultats dans le domaine temps sont plus faciles à visualiser que dans le domaine fréquentiel.

Par la suite, il sera probablement nécessaire d'utiliser la méthode de la figure 2.7 qui nous permettrait de traiter de systèmes plus complexes.

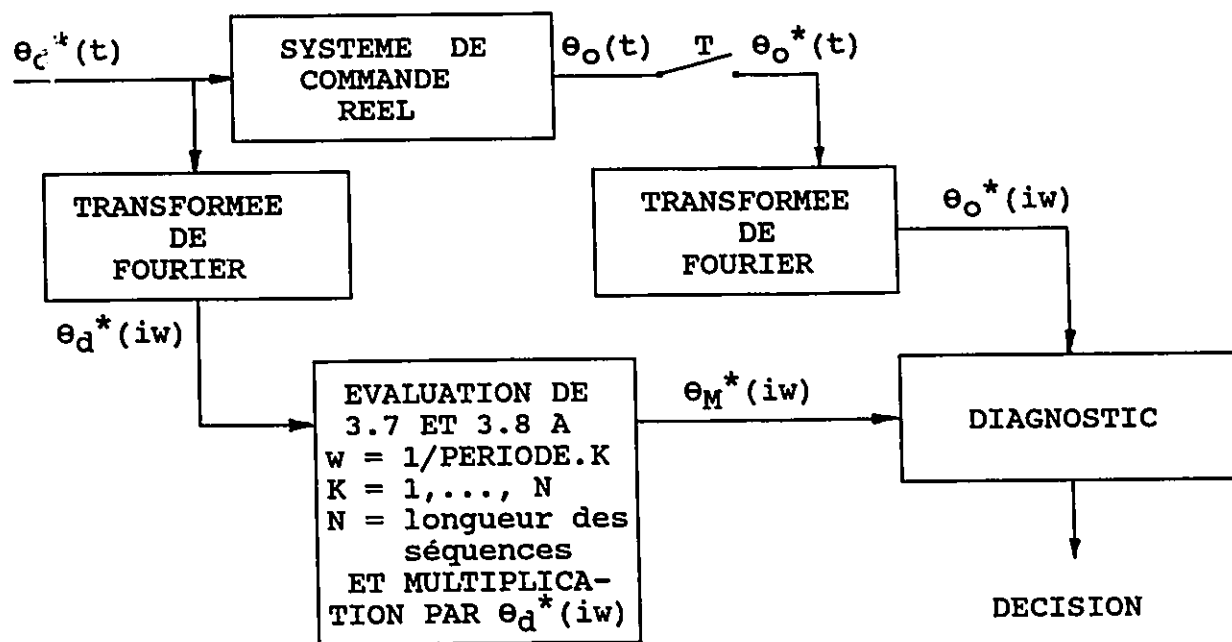


Figure 2.7 Processus de diagnostic avec le modèle dans le domaine fréquentiel et la comparaison dans le domaine fréquentiel.

Comme l'entrée et la sortie du système réel échantillonnée sont des séquences on doit utiliser des algorithmes de transformée de Fourier discrete directe et inverse qui seront décrits dans la section 2.2 du présent chapitre.

2.2 L'implémentation du modèle

Les équations 2.7 et 2.8 seront utilisées pour représenter le modèle du système réel. La sous-routine MODEL dans le programme de détection automatique des anomalies, annexe D, évalue donc ces 2 équations aux fréquences appropriées pour produire la séquence qui sera multipliée par l'entrée.

2.2.1 Les algorithmes de transformée de Fourier directe et inverse

Considérons un signal continu quelconque, $f(t)$, échantillonné à interval régulier, T , et représenté sur la figure 2.8.

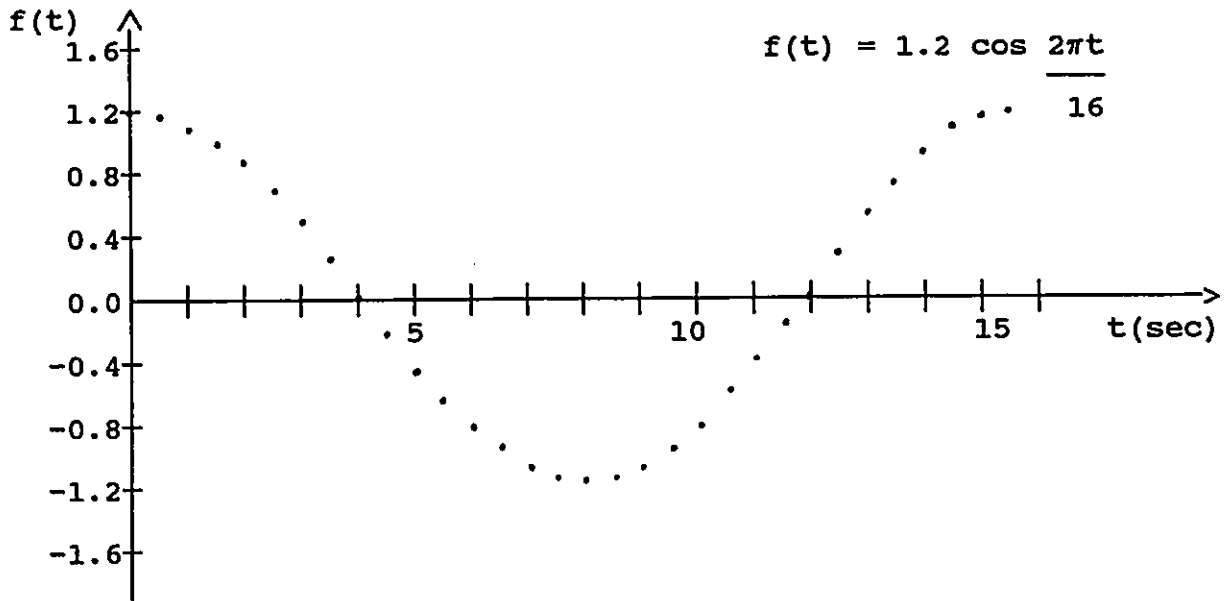


Figure 2.8: Séquence de signal échantillonné.

A cette séquence dans le domaine temporel on peut associer une séquence de N valeurs dans le domaine fréquentiel, définie comme la transformée de Fourier discrète,

$$F \left[\begin{array}{c} k \\ \hline NT \end{array} \right] = T \sum_{j=0}^{N-1} f(jT) e^{-i2\pi kj/N}, \quad k = 0, \dots, N/2 \quad (2.9)$$

où $f(jT)$ est une séquence réel de longueur N .
 i est $\sqrt{-1}$
 T est la période d'échantillonnage.
 N est le nombre d'échantillons utilisés.

Notez que l'indices k varie de 0 à $N/2$ et non de 0 à $N-1$. En effet pour une séquence réelle, il n'est pas nécessaire de calculer la transformée de Fourier pour les indices $N/2+1$ à $N-1$ car ceux-ci sont simplement les valeurs conjuguées des nombres complexes, $F(k/NT)$, calculées de 0 à $N/2$.

Comme le montre la figure 2.8, le signal échantillonné est nul en dehors de la séquence connue. La transformée de Fourier discrète est donc équivalente à la transformée de Fourier continue en (f) échantillonnée en fréquence au pas $\Delta f = 1/NT$. La figure 2.9 nous montre la transformée de Fourier du signal de la figure 2.8 pour $N = 32$ et $T = 0.5$.

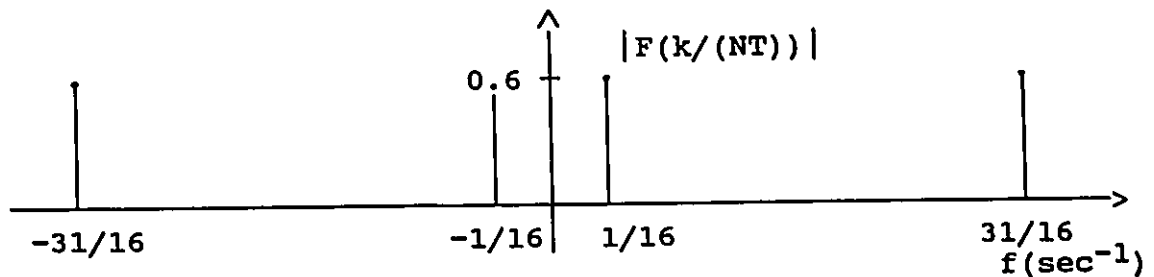


Figure 2.9: Transformée de Fourier de la figure 2.8 pour $N = 32$ et $T = 0.5$ sec

$F(k/NT)$ est une séquence de nombre complexe et sa grandeur peut être exprimée par:

$$|F(k/NT)| = \sqrt{[F(k/NT)_{\text{réel}}]^2 + [F(k/NT)_{\text{imag.}}]^2}$$

pour $k = 0, \dots, N/2$

La transformation inverse est définie par :

$$f(jT) = 1/N \sum_{k=0}^{N/2} F(k/(NT)) e^{i2\pi kj/N}, \quad j = 0, \dots, N-1 \quad (2.10)$$

Pour la transformation Fourier inverse de la figure 2.9, le signal revient à sa forme de la figure 2.8. Pour des signaux nuls en dehors de la séquence connue, comme il est le cas ici, il est possible d'avoir des problèmes de repliement de spectre (aliasing) si la fréquence d'échantillonnage n'est pas assez élevée. Il faut échantillonner le signal au moins au double de la plus haute fréquence significative contenue dans le signal (théorème de Shannon).

L'exemple présenté à la figure 2.8 et 2.9 a été calculé avec les routines qui sont employées dans le programme de détection automatique des anomalies (annexe D). Ces routines de calcul numérique des transformations de Fourier directe et inverse proviennent du logiciel 87FFT de MICROWAY version 2.03, [49]. Ce logiciel utilise l'algorithme de transformée de Fourier Rapide (FFT) "Radix 2" de Cooley-Tukey [47], [48] et [50].

Nous avons maintenant un modèle de notre système réel dans le domaine fréquentiel, éq. (2.7) et (2.8). Un moyen de passer du domaine temporel au domaine fréquentiel est aussi disponible par la transformation DFFT (transformation discrète de Fourier rapide directe) et vice versa par la IDFT (transformation discrète de Fourier rapide inverse). La séquence de sortie du modèle dans le domaine temps peut être comparée à la sortie échantillonnée du système réel afin de vérifier la validité du modèle développé.

2.3 Le modèle vs le système de commande réel

Le modèle développé doit être représentatif de la réalité, s'il sera utilisé pour le diagnostic. La restriction majeure du modèle est qu'il ne tient pas compte des variations de charge sur le moteur. Mais il apparaissait plus logique de procéder par étapes, c'est-à-dire de commencer par étudier un modèle à charge fixe et par la suite modifier celui-ci pour tenir compte des variations de la charge. Les effets d'avoir une charge variable seront étudiés à la section 2.5.

Cette section vise à présenter les résultats des tests effectués sur le montage expérimental, afin de vérifier la validité du modèle. Les tests ont été faits pour différents types de signaux d'entrées, soit: un échelon, une rampe et une entrée sinusoïdale.

2.3.1 La procédure de test

i) Pour le système de commande réel.

1- Modifier la sous-routine POSI dans le programme de commande du moteur (annexe C) pour chaque type d'entrées. La sous-routine POSI donne la position désirée, θ_d , au programme de commande principal.

Echelon : $\theta_d = 70$ degrés

Rampe : $\theta_d = 1/2 \cdot t$, $t = \text{temps}$

Sinus : Le microprocesseur utilisé (Z8671) ne possède pas de fonction trigonométrique, alors un programme a été écrit pour créer un sinus. Le programme est dans la sous-routine POSI de l'annexe C.

2- Actionner le système de commande et enregistrer le signal de sortie échantillonné dans un fichier.

ii) Pour le modèle.

1- Ecrire un programme Fortran pour obtenir la sortie du modèle dans le domaine temps. Les mêmes entrées que pour le système de commande réel doivent être utilisées. Le programme se compose de trois routines.

RFFT pour faire la transformée de Fourier directe du signal d'entrée.

MODEL pour calculer la sortie du modèle dans le domaine fréquentiel.

RIFT pour faire la transformée de Fourier inverse du signal de sortie du modèle et obtenir un signal dans le domaine temporel.

Ces routines sont les mêmes que celles utilisées dans le programme de détection automatique des anomalies de l'annexe D.

2- Activer le programme et enregistrer le signal de sortie du modèle dans un fichier.

2.3.2. Comparaisons des résultats de tests.

i) Pour un échelon, la figure 2.10 nous montre les résultats obtenus pour le modèle (ligne pointillée) et pour le système réel (ligne pleine). Les signaux représentés à la figure 2.10 sont des séquences. Les points ont été reliés par des courbes approximatives afin de faciliter la visualisation.

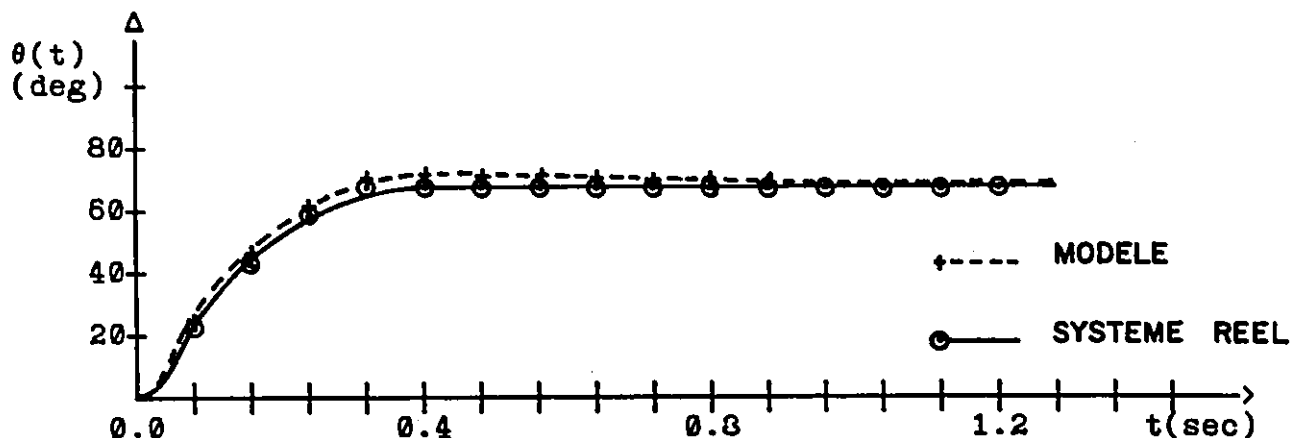


Figure 2.10: Modèle vs système réel pour un échelon.

ii) Pour une rampe, la figure 2.11 nous montre les résultats obtenus.

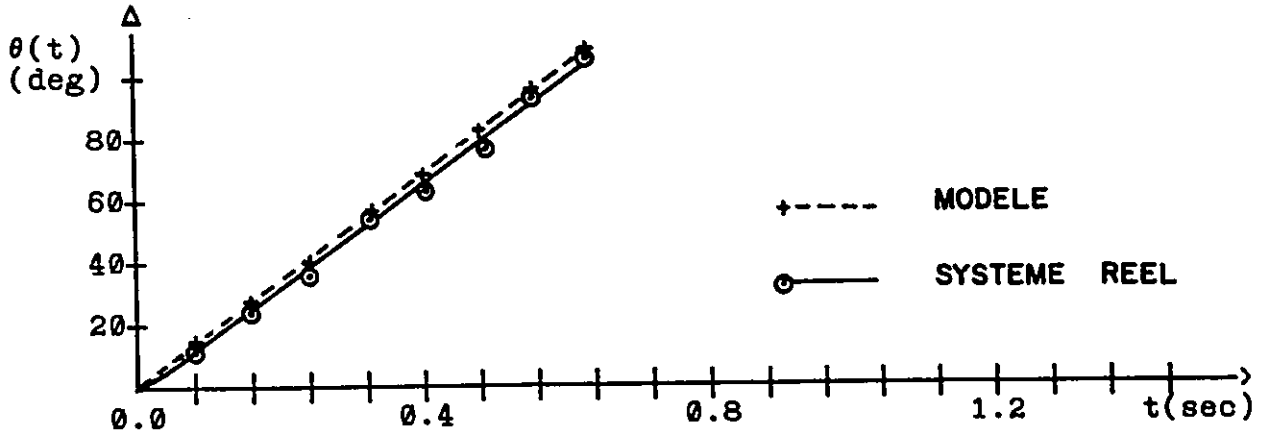


Figure 2.11: Modèle vs système réel pour une rampe.

iii) Pour un sinus, la figure 2.12 nous montre les résultats obtenus.

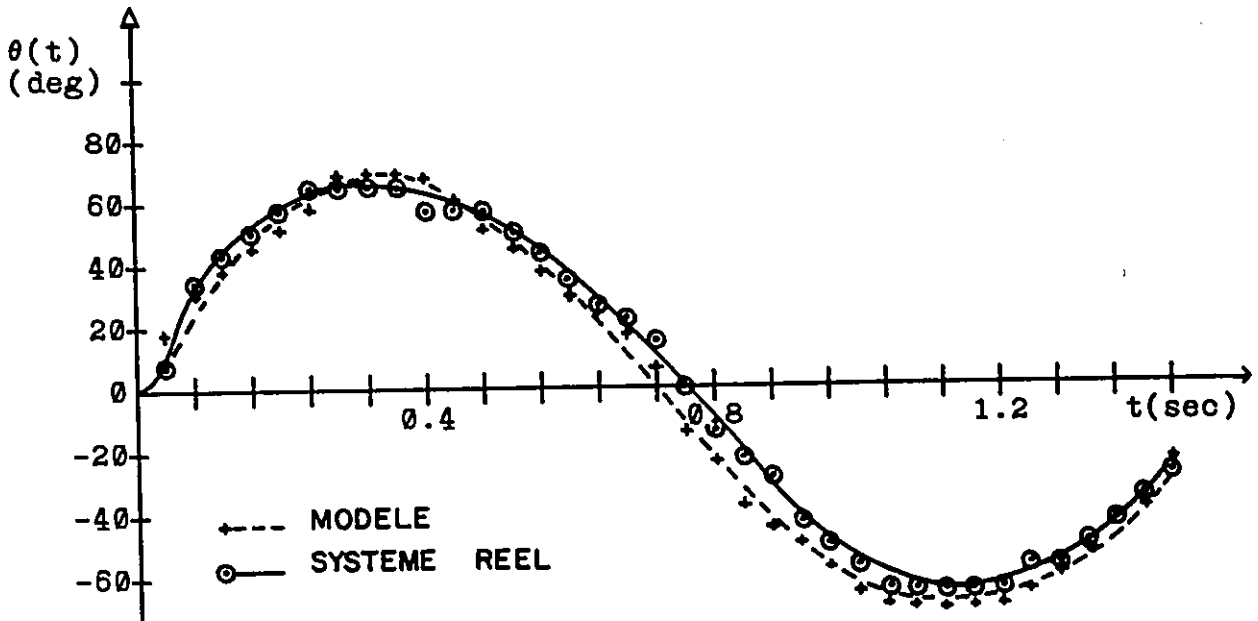


Figure 2.12: Modèle vs système réel pour un sinus.

La comparaison des résultats des figures 2.10 à 2.12 nous permet de dire que le modèle est valide. Celui-ci sera donc utilisé tel que présenté à la section 2.1.3.

La prochaine section présente une étude théorique des défaillances basée sur un modèle similaire à celui présenté plus haut; cette modélisation des défaillances sera aussi comparée au système réel, comme il a été fait pour le modèle du système sans défaillances.

2.4 Etude théorique et expérimentale des défaillances.

Avant de concevoir une méthode de diagnostic pour un système quelconque, il est important de connaître de quelle façon celui-ci réagit aux défaillances. L'introduction d'une méthode de diagnostic peut se faire à deux niveaux: au moment de la conception du système ou sur un système existant et opérationnel. Dans les deux cas, le système n'est pas nécessairement disponible pour effectuer des tests afin de savoir comment le système réagit aux défaillances. Il serait donc souhaitable de pouvoir faire cette étude hors ligne.

Cette section présente une méthode de modélisation des défaillances développée pour étudier les défaillances du système de commande réel. La méthode est basée sur le graphe de fluence du système et permet de visualiser l'effet de défaillances.

2.4.1 La modélisation des défaillances

La méthode présentée ici, a été initialement développée par Kumamoto et Henley [6] pour la construction des arbres de vérité pour des systèmes avec des boucles de contrôle multiples. Elle sera modifiée ici pour modéliser la réponse du système aux diverses défaillances. La première étape est de dessiner le graphe de fluence pour le système à étudier. Celui-ci est représenté à la figure 2.13. Le tachymètre est utilisé ici seulement par le programme de diagnostic comme indicateur de mouvement. Il ne fait pas partie du système de commande donc sa sortie n'influence pas la réponse du système, le programme de diagnostic ne peut donc pas identifier la défaillance du tachymètre. C'est pourquoi

les effets des défaillances du tachymètre sont étudiés séparément à la section 3.5. En fait la décision d'utiliser le tachymètre comme indicateur de mouvement a été prise à la dernière minute, et il était trop tard à ce moment pour rajouter la boucle de retour de vitesse. Si celle-ci était présente, il faudrait obligatoirement refaire toute l'analyse qui suivra, ce qui demande beaucoup de temps et n'était pas possible.

Au graphe de fluence traditionnel on va ajouter 10 variables sources X_1 à X_{11} qui permettront de modéliser les défaillances des composantes. On appelle graphe de fluence modifié, le nouveau graphe de la figure 2.14. La signification des variables sources est donnée à la table 2.1. Pour simplifier l'analyse, seulement les défaillances totales sont considérées, ce qui représente d'après O'Connor [10] en moyenne 30% des cas de défaillances pour des composantes électroniques (transistor, résistance) et de 70% pour les microplaquettes. Ce type de défaillances est équivalent à ouvrir le circuit, c'est à dire que sur le graphe de fluence toute les branches qui entrent dans une composante en défaillances sont considérées nulles. Les autres types de défaillances les plus communes vont produire un court-circuit ou un changement dans la valeur des paramètres pour les composantes en défaillances. Il reste à prouver que la présente méthode est aussi efficace pour ces défaillances. De plus la méthode expérimentale a été développée en premier lieu pour une sortie nulle des composantes en défaillances. Les sorties non-nulles lorsque en défaillance seront toutefois étudiées à la section 2.4.3, avec l'aide du programme de simulation.

En utilisant la règle de Mason, tel que décrite dans l'annexe B, et le principe de superposition il est possible d'obtenir l'équation du système en fonction des variables sources et de l'entrée, θ_a , dans le domaine Laplace, pour le graphe de la figure 2.14, [57], [58] et [59].

$$\theta_o = \frac{X_4 + X_5.M + X_4.A.M + (X_2 + X_3 + X_7 + X_8).D/A.A.M}{1 + D/A.A.M.K_r.K_c.(K_p + K_i/s + K_d s)} \quad (2.11)$$

$$+ \frac{(X_1 + \theta_a^* - X_{10} - X_9.K_c).D/A.A.M.(K_p + K_i/s + K_d)}{1 + D/A.A.M.K_r.K_c.(K_p + K_i/s + K_d s)}$$

A partir de cette équation toute les défaillances peuvent être modélisées. Quoique que nous assumons qu'il y ai une seule défaillance à la fois pour la présente étude, l'équation 2.11 nous permet aussi d'étudier les défaillances multiples.

Table 2.1 Signification des variables sources, X_1 à X_{10} .

VARIABLE	SIGNIFICATION
X_1	Sortie du comparateur lorsque celui-ci est en défaillance.
X_2	Sortie de la partie proportionnelle du regulateur lorsqu'elle est en défaillance
X_3	Sortie du régulateur PID lorsque l'additionneur est en défaillance
X_4	Sortie du convertisseur digital à analogique lorsqu'il est en défaillance
X_5	Sortie de l'amplificateur lorsqu'il est en défaillance
X_6	Sortie du moteur lorsque celui-ci est en défaillance
X_7	Sortie de la partie intégrale du régulateur lorsqu'elle est en défaillance
X_8	Sortie de la partie différentielle du régulateur lorsqu'elle est en défaillance
X_9	Sortie du lecteur optique lorsqu'il est en défaillance
X_{10}	Sortie du compteur lorsqu'il est en défaillance
X_{11}	Sortie du tachymètre en défaillance.

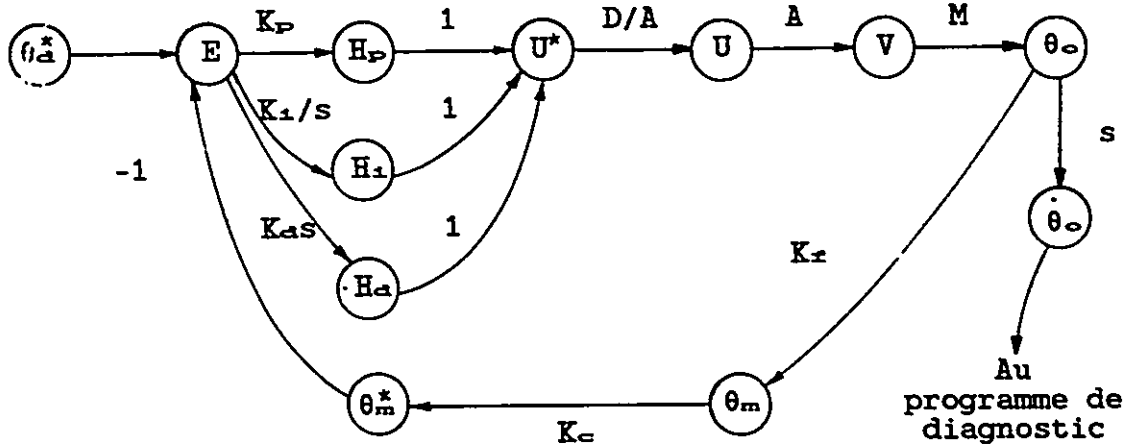


Figure 2.13: Le graphe de fluence du système de commande

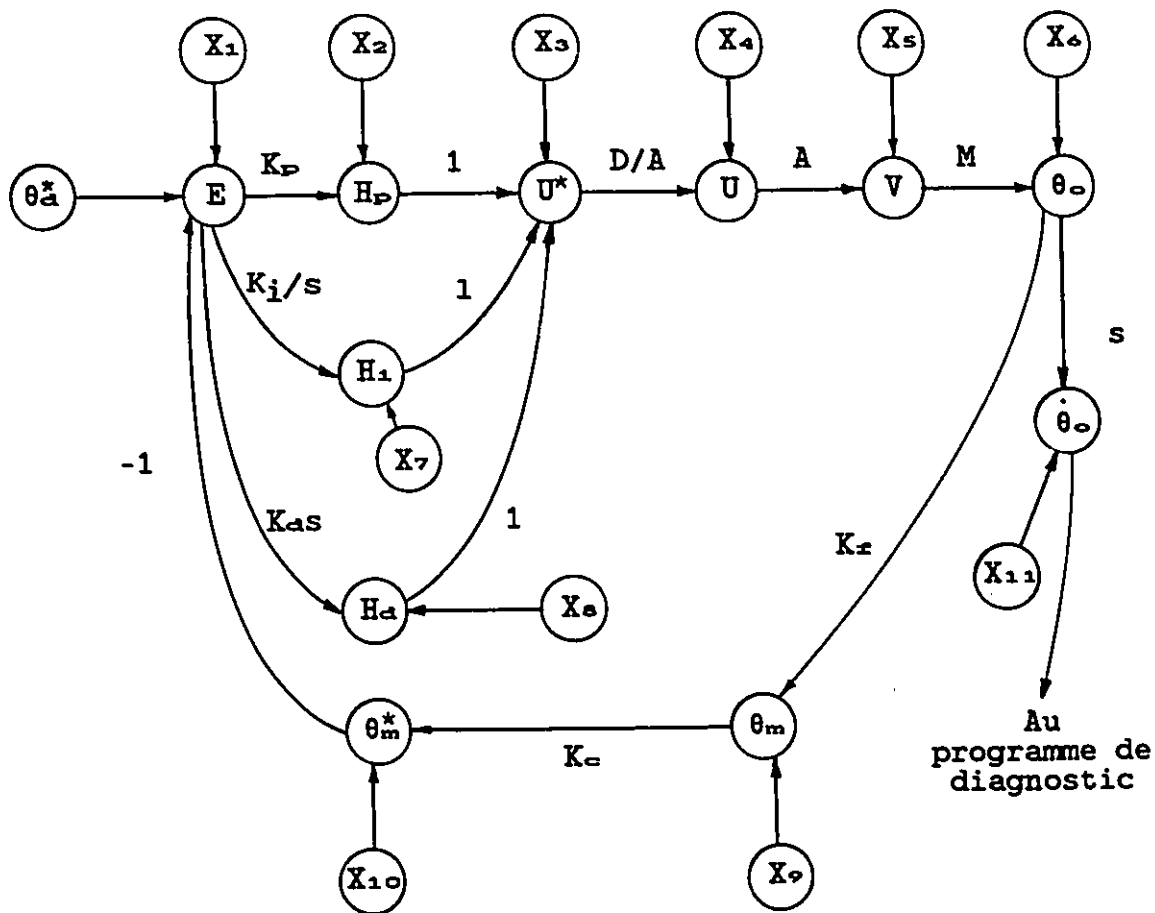


Figure 2.14: Le graphe de fluence modifié

Pour chaque défaillance les variables sources et les transmittances sont soit égales à zéro ou non égales à zéro. La table 2.2 nous montre la convention utilisée. Avec cette convention et l'équation 2.11 il est facile d'écrire un programme de simulation des défaillances.

Table 2.2: Simulation des défaillances

DEFAILLANCE	TRANSMITTANCE	VARIABLE SOURCE
Comparateur, E	$K_c = 0$	$X_1 \neq 0$, tout les autres $X_i = 0$ $\theta_d = 0$
Partie proportionnelle du régulateur H_p	$K_p = 0$	$X_2 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Additionneur, U^*	$K_p = 0$ $K_i = 0$ $K_d = 0$	$X_3 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Convertisseur digital à analogique	$D/A = 0$	$X_4 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Amplificateur	$A = 0$	$X_5 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Moteur	$M = 0$	$X_6 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Partie intégrale du régulateur	$K_i = 0$	$X_7 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Partie différentielle du régulateur	$K_d = 0$	$X_8 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Module optoelectronique	$K_f = 0$	$X_9 \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$
Compteur	$K_c = 0$	$X_{10} \neq 0$, tout les autres $X_i = 0$ $\theta_d \neq 0$

2.4.2 Le programme de simulation des défaillances

Le programme est structuré comme le montre la figure 2.15. Le signal d'entrée échantillonné est lu dans un fichier. Le type de défaillance à simuler est choisi de même que l'instant où la défaillance se produira. La séquence subit par la suite une transformation de Fourier directe discrète (DFFT). La sortie du modèle est calculée et une transformation de Fourier inverse discrète (DIFT) est exécutée pour obtenir un signal échantillonné de sortie représentant la réponse du système. Le graphique de la réponse est alors tracé et peut être comparé à la réponse en fonctionnement normal du système.

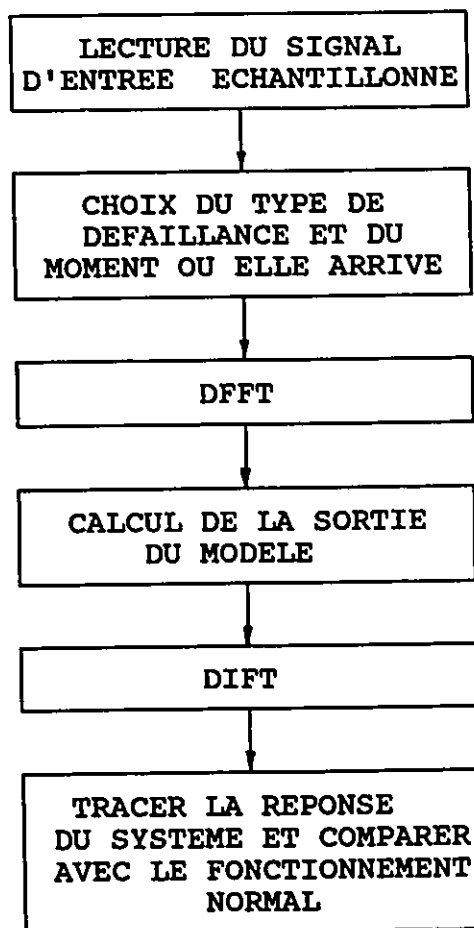


Figure 2.15: Structure du programme de simulation des défaillances.

Ce programme peut être utilisé au niveau de la conception de systèmes de commande. Il permet de visualiser les effets des défaillances sur la réponse du système et par conséquent modifier le système pour le rendre plus résistant aux défaillances. Les résultats obtenus par ce programme sont présentés dans la section suivante, 2.4.3, et ils sont comparés aux résultats obtenus avec le montage expérimental.

2.4.3 Modélisation des défaillances vs défaillances réels

Afin de vérifier la validité des résultats du programme de simulation des défaillances, ceux-ci ont été comparés aux résultats de tests faits sur le montage expérimental. Le sommaire des tests et des comparaisons est présenté dans cette section. Les expériences ont été faites pour un signal nul à la sortie des composantes en défaillances. Les effets d'un signal non-nul ont toutefois été étudié à l'aide du programme de simulation et seront commentés.

i) Défaillance de la partie proportionnelle du régulateur

Il était difficile ici de produire la défaillance de la partie proportionnelle du régulateur sur le montage expérimental car on utilise un régulateur numérique. La défaillance a donc été produite en posant, $K_p = 0$ dans le programme de commande du moteur. Les résultats présentés à la figure 2.16, pour une rampe montrent que le programme de simulation représente assez adéquatement la défaillance du montage expérimental pour une sortie nulle à la défaillance. Pour une sortie non-nulle, l'instabilité croît plus rapidement tel qu'illustré à la figure 2.16.

Lorsque la partie proportionnelle du régulateur est en défaillance le mouvement devient imprévisible, l'allure de la courbe de la figure 2.16 dépend grandement de l'instant où la défaillance se produit. Plus la grandeur du signal est élevée à la défaillance, plus l'amplitude des oscillations augmente vite.

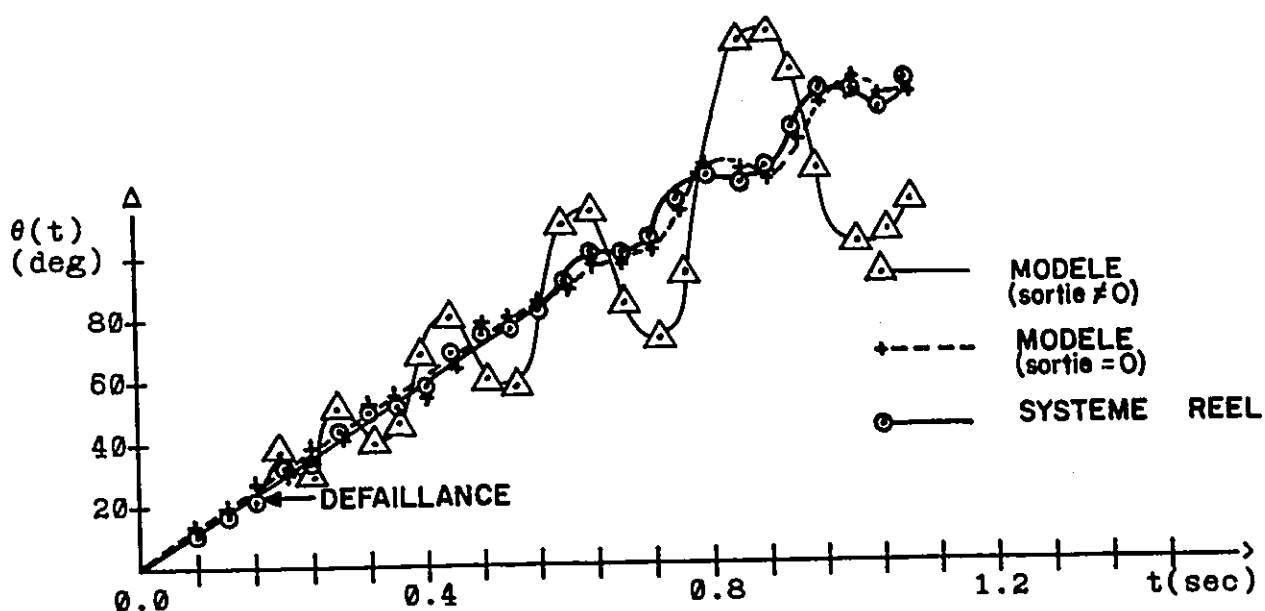


Figure 2.16: Défaillance de la partie proportionnelle du régulateur.

ii) Défaillance de la partie intégrale du régulateur

La défaillance de la partie intégrale du régulateur a été produite, sur le montage expérimental, en posant $K_i = 0$ dans le programme de commande du moteur. Les résultats sont présentés à la figure 2.17.

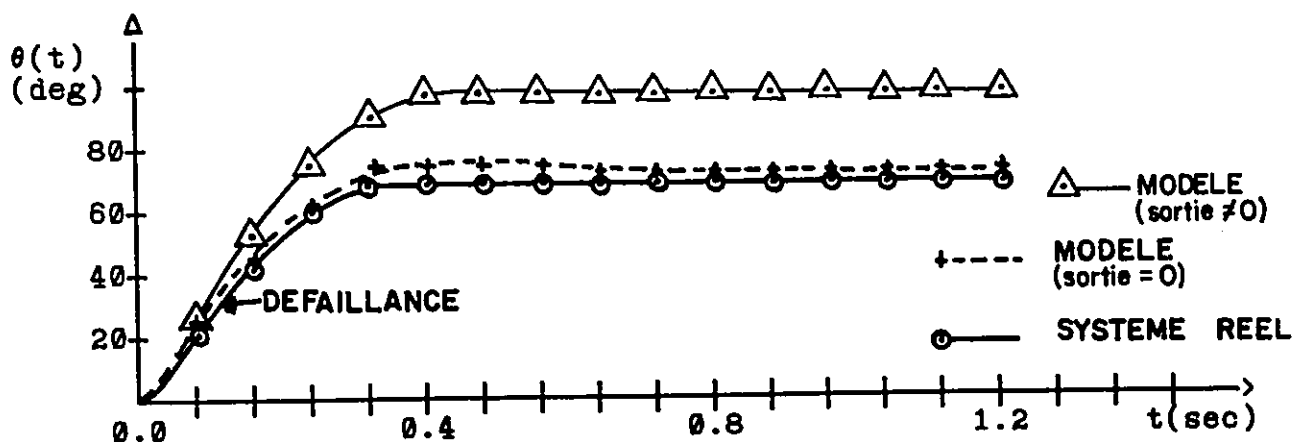


Figure 2.17: Défaillance de la partie intégrale du régulateur

On remarque que la défaillance est à peine perceptible pour une sortie nulle à la défaillance. En général, la partie intégrale est ajoutée à un système pour en réduire l'erreur en régime permanent. Comme on peut le voir sur la figure 2.17, pour un échelon cette défaillance a produit une augmentation du dépassement transitoire et une augmentation de l'erreur comme prévu par la théorie. Plus le signal à la sortie devient grand, plus l'erreur en régime permanent devient grande. Sur le montage expérimental, cette défaillance n'est pas perceptible, pour une sortie nulle parce que le programme de commande du moteur, sur le microprocesseur Z8, fonctionne avec des nombres entiers uniquement et comme l'erreur est faible la sortie du programme n'est pas modifiée. Il était par ailleurs difficile de simuler des valeurs non-nulles à la défaillance sur le montage expérimental sans apporter des modifications importantes à celui-ci.

iii) Défaillance de la partie différentielle du régulateur

Il n'est pas pertinent de simuler la défaillance de la partie différentielle du régulateur avec la procédure employée pour la partie proportionnelle ou intégrale. En général la partie différentielle produit une action plus rapide de la sortie pour une entrée qui varie rapidement et réduit le dépassement transitoire [2]. Les entrées utilisées ici comme la rampe et le sinus ne produisent pas de variation assez rapide du signal et l'échelon tel qu'utilisé et illustré à la figure 2.18 non plus.

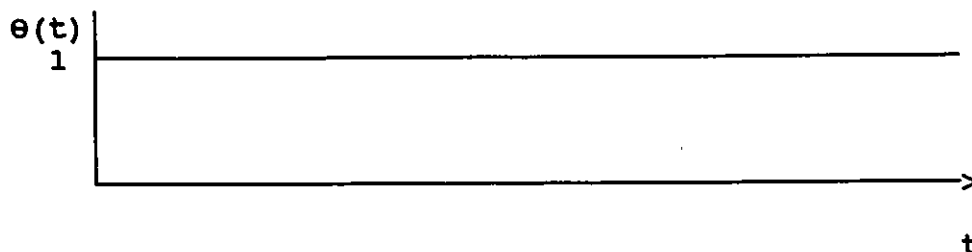


Figure 2.18: Echelon utilisée pour la simulation des défaillances.

Une étude plus détaillée serait donc nécessaire pour avoir des résultats valables.

Ces constatations sont très importantes. En fait, elles signifient que, pour le système, même si la partie différentielle et intégrale sont en défaillance le système préservera sa stabilité et le mouvement sera prévisible. On peut donc affirmer que le régulateur est résistant aux défaillances de ces deux parties, pour des entrées qui ne varient pas trop rapidement, ce qui est fortement désirable. Le modèle représente aussi bien un régulateur digital que analogique car l'équation 2.11 ne tient pas compte du fait que le régulateur est digital. Ceci parce qu'il a été posé que $D/A(s) = 1$ (voir annexe B, section A). Si au lieu d'avoir une entrée digitale, θ_d^* , une entrée analogique était utilisée alors on obtiendrait une sortie analogique. Il reste à vérifier expérimentalement que pour un régulateur analogique la sortie réelle est équivalente à la sortie du modèle.

iv) Défaillance de module optoélectronique

La défaillance du lecteur optique pour une sortie nulle a été produite, sur le montage expérimental, en ouvrant le circuit comme le montre la figure 2.19. Un second lecteur optique a été utilisé pour enregistrer les positions.

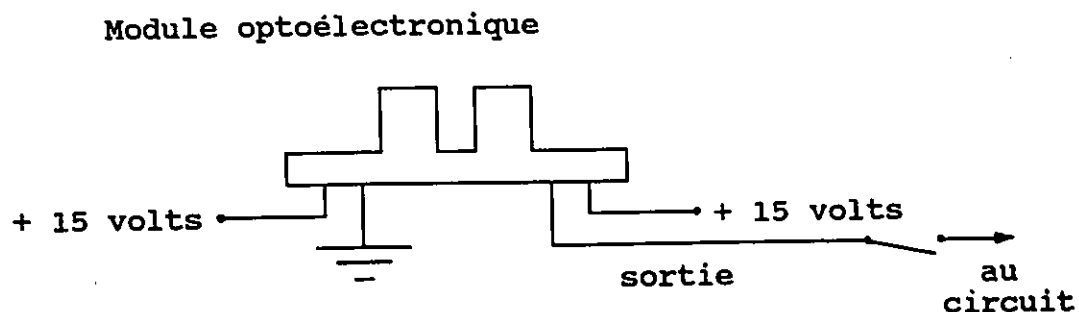


Figure 2.19: Simulation de la défaillance du module optoélectronique

La figure 2.20 nous montre les résultats obtenus, pour une sortie nulle, sur le montage expérimental et par le programme de simulation des défaillances.

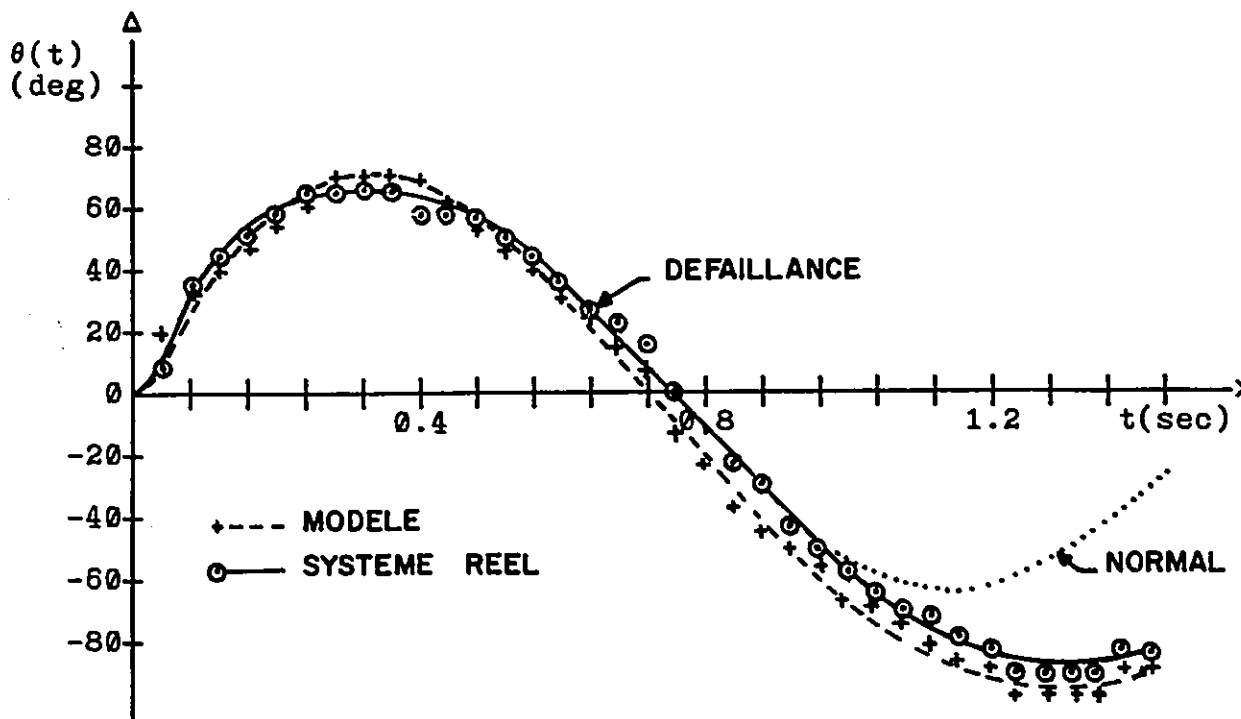


Figure 2.20: Défaillance du lecteur optique.

Pour une sortie non-nulle, l'effet de la variation de la grandeur de la sortie sur la réponse du système est le même que pour une sortie nulle. Ceci est dû au fait que si la valeur de sortie du lecteur est constante et différente de zéro alors le compteur à l'intérieur du Z8 n'incrémente ou ne décrémente plus la position obtenue. Le compteur incrémente seulement lorsque le signal passe d'une valeur nulle à une valeur non-nulle, un signal constant ne peut pas commander le compteur.

Le deuxième module optoélectronique n'est pas connecté au programme de commande du moteur. Il est utilisé uniquement pour mesurer la sortie réelle du système lorsque le lecteur optique fonctionnel est en défaillance. Pour le programme de commande

l'effet d'une défaillance avec une sortie nulle ou non-nulle est le même peu importe le type d'entrée. Comme le programme de commande du moteur ne reçoit plus de signal valide du lecteur optique, il garde en mémoire la dernière valeur lue et considère que le système reste à cette position, quoique le moteur tourne toujours tentant d'atteindre la position désirée.

v) Défaillance du D/A

On a simulé cette défaillance comme suit sur le montage expérimental pour une sortie nulle.

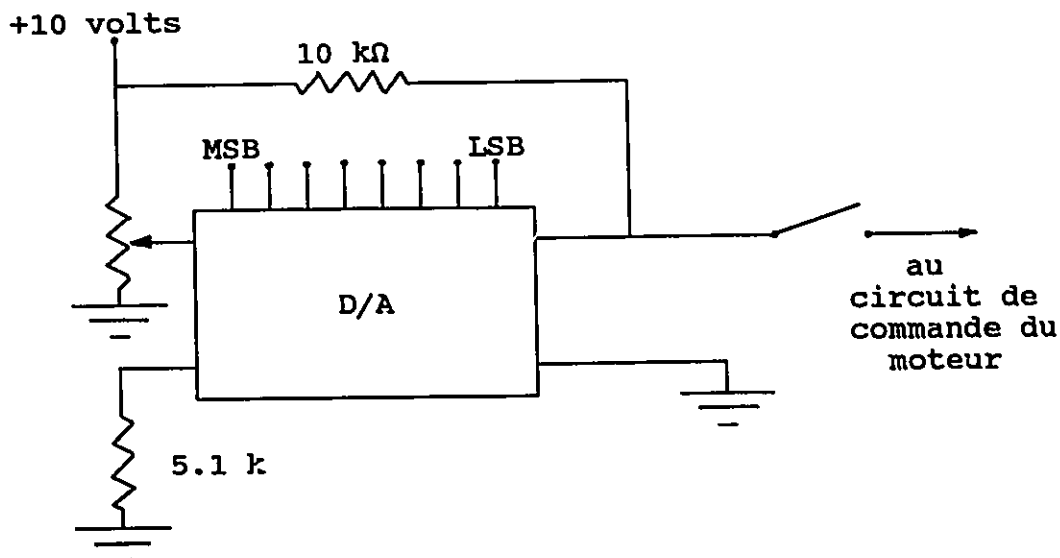


Figure 2.21: Simulation de la défaillance du D/A sur la montage expérimental

Pour le D/A, la réponse est la même peu importe le type d'entrée, (figure 2.22). La défaillance totale du D/A, avec une sortie nulle produit toujours l'arrêt complet du moteur. Pour une sortie non-nulle le moteur tournera à vitesse constante dont la grandeur dépendra de la grandeur du signal de sortie, tel qu'illustré à la figure 2.22.

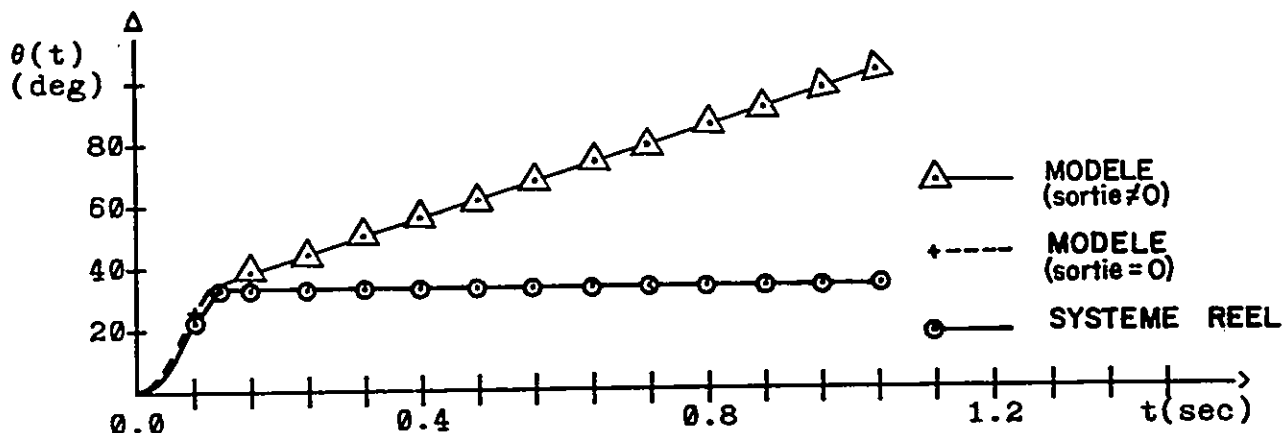


Figure 2.22: Défaillance du D/A.

vi) Défaillance du moteur.

La défaillance du moteur, pour une sortie nulle a été produite en ouvrant le circuit de commande comme le montre la figure 2.24. Ceci représente seulement la défaillance en circuit ouvert du moteur. Les autres types de défaillances du moteur, comme le blocage de l'arbre, ne sont pas considérés.

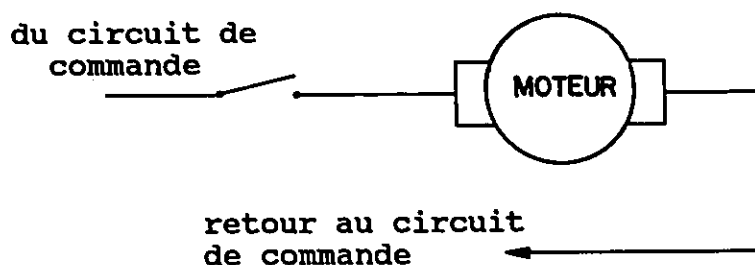


Figure 2.23: simulation de la défaillance du moteur.

L'effet observé est le même que pour le D/A, soit l'arrêt complet du moteur. Il est important de noter que le moteur est arrêté mais est toujours sous tension. Pour un signal non-nulle à la défaillance le moteur tournera à une vitesse constante imposée par le circuit de commande, peu importe le type de signal d'entrée. Il est important de rappeler que nous étudions seule-

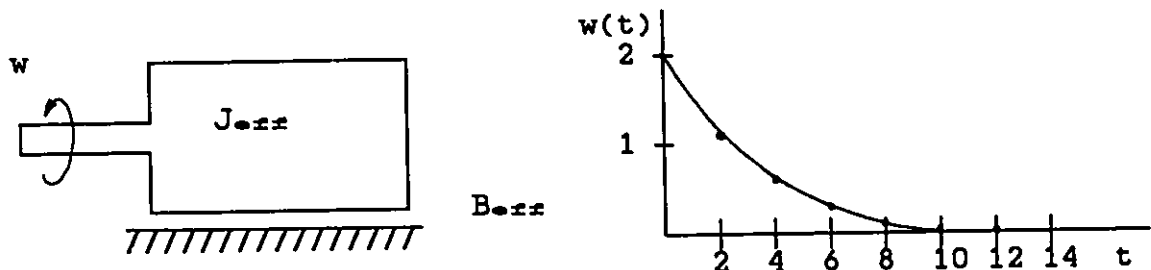
ment les défaillances totales des composantes tel que mentionné à la section 2.4.1. Pour le système expérimental que nous avons étudié, le temps de freinage due à l'inertie était négligeable à cause des engrenages utilisés (facteur de 1/127). Pour des systèmes robots plus complexes l'inertie de la charge est souvent très élevée et les moteurs ne possèdent pas tous le même type d'engrenage. Il est donc important de se rappeler que de tels moteurs demandent un temps d'arrêt non-négligeable et variable en fonction de la charge transportée. Il est possible de calculer approximativement le temps d'arrêt dans de telles situations en schématisant le système comme à la figure 2.24 a) où J_{eff} représente l'inertie efficace et B_{eff} le coefficient d'amortissement efficace à la charge. L'équation de mouvement, dans le domaine Laplace, pour le système de la figure 2.24 a) est

$$J_{eff} \cdot s \cdot w(s) - J_{eff} \cdot w(0)^+ + B_{eff} \cdot w(s) = 0$$

d'où,

$$w(t) = w(0)^+ \cdot e^{-(B_{eff}/J_{eff})t}$$

Par exemple pour le manipulateur Stanford des valeurs d'inertie sont disponibles (62) et vont nous permettre de calculer le temps d'arrêt. Pour le liens 2, avec la charge maximum, J_{eff} égale 1459 oz.in.s²/rad. La valeur totale de B_{eff} n'est pas disponible, alors une valeur minimal de 430 oz.in.s/rad sera utilisée. Cette valeur est calculée à partir de la valeur du coefficient d'amortissement à l'arbre du moteur donnée dans (62).



a) Schéma du système

b) Réponse

Figure 2.24: Temps de freinage non-nul.

De plus une vitesse initiale de 2 rad/s est assumée. La figure 2.24 b) nous montre la réponse obtenue. Pour les valeurs mentionnées le temps d'arrêt est d'environ 11 seconde. Plus le rapport B_{-xx}/J_{-xx} augmente, plus le temps d'arrêt diminuera et vice-versa.

Ceci aura un effet direct sur la méthode de diagnostic. Le processus de détection des erreurs ne sera pas affecté. Par contre le processus d'identification, qui se sert du fait que le moteur arrête lors d'une défaillance, sera perturbé. Supposons une défaillance du moteur 2 du manipulateur Stanford ci-haut mentionné à $t = 0$. Le processus de détection prend de 2.0 à 3.5 sec. pour détecter la défaillance dépendamment du moment où celle-ci survient. Après quoi, on passe à l'identification des anomalies. Donc on peut affirmer que si le temps d'arrêt est inférieur à 2.0 sec. alors l'identification fonctionnera à tout coup. Par contre si le temps d'arrêt est supérieur à 2.00 sec. le programme ne pourra pas, en général, identifier la provenance de la défaillance. Il sera donc important de tenir compte de ce fait lors de l'implémentation sur un système robot où l'inertie est appréciable. Une façon de faire est d'introduire un délai à l'intérieur du processus d'identification correspondant au temps d'arrêt maximum pour le robot concerné.

vii) Défaillance du comparateur.

Comme nous utilisons un comparateur digital on a défini sa défaillance comme des signaux d'entrées étant égaux à zéro. Donc l'erreur entre la position désirée et celle obtenue est toujours égale à zéro. L'effet est donc le même que pour la défaillance du D/A et du moteur.

viii) Conclusion

Le modèle utilisé pour simuler les défaillances est linéaire. Il était important de montrer qu'un tel système, qui peut

être adéquat pour représenter le fonctionnement normal, était acceptable pour modéliser le système en défaillance. Les résultats des comparaisons avec le montage expérimental ont montré que, pour le cas étudié, la modélisation est valable. On peut conclure que le programme de simulation de défaillances est valide et produit des résultats représentatifs de la réalité.

1- Défaillances avec sortie nulle.

Les tests ont révélé que pour le montage expérimental, on peut classer les défaillances avec sortie nulle, en deux groupes.

1. Défaillances des composantes sur le chemin principal.
2. Défaillance des éléments sur la boucle de retour.

On considère seulement les défaillances de composantes physiques. Comme un régulateur numérique est utilisé, ses défaillances ne seront pas considérées par le programme de diagnostic. Les effets de sa défaillance sur le système de diagnostic automatique seront toutefois étudiés dans la section 3.6.

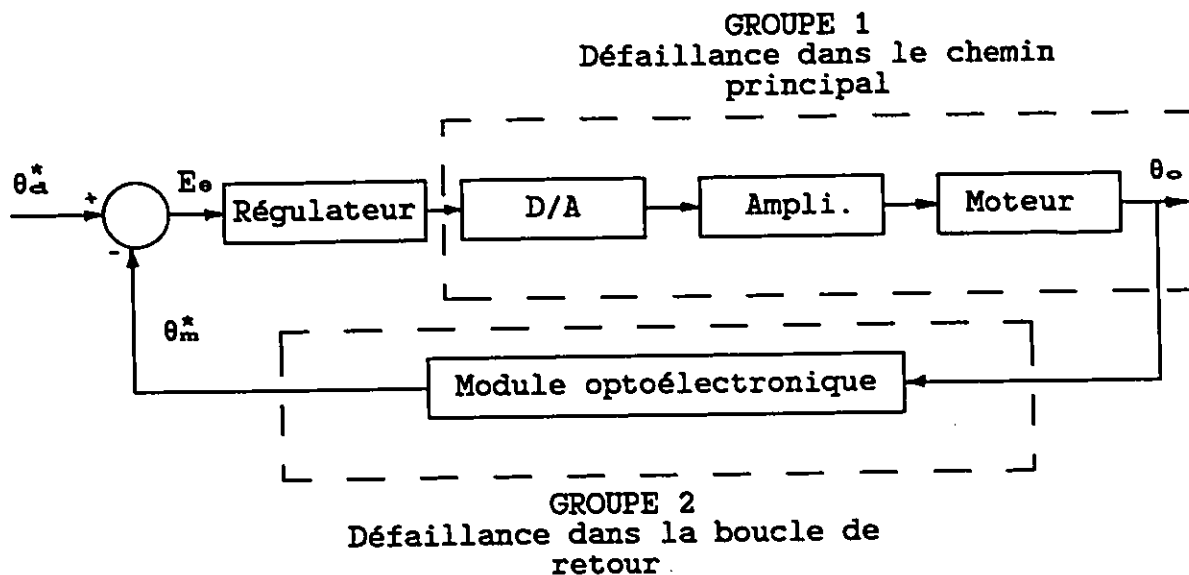


Figure 2.25: Deux groupes de défaillances.

Pour les défaillances du premier groupe, on obtient une réponse indépendante du type d'entrée et qui peut être représentée par la figure 2.26. C'est-à-dire que si une composante du groupe 1 est en défaillance, peu importe le type d'entrée (sinus ou rampe ou autre), le moteur arrête. L'indicateur de mouvement est utilisé pour vérifier que effectivement le moteur est arrêté. Le signal de l'indicateur de mouvement provient d'un tachymètre directement couplé au moteur. Si la sortie du tachymètre est différente de zéro, donc si le moteur tourne à une certaine vitesse, l'indicateur de mouvement devient 1 sinon il est zéro. Cet indicateur est lu par le programme de diagnostic. L'indicateur n'a aucune autre fonction. En général, les systèmes robots sont équipés de tachymètres en plus des lecteurs de position, il n'est donc pas nécessaire de rajouter ceux-ci. Pour la présente étude il a été supposé que le tachymètre est exempt de défaillance. Les effets de sa défaillance sur la méthode de diagnostique automatique seront toutefois étudiés à la section 3.5

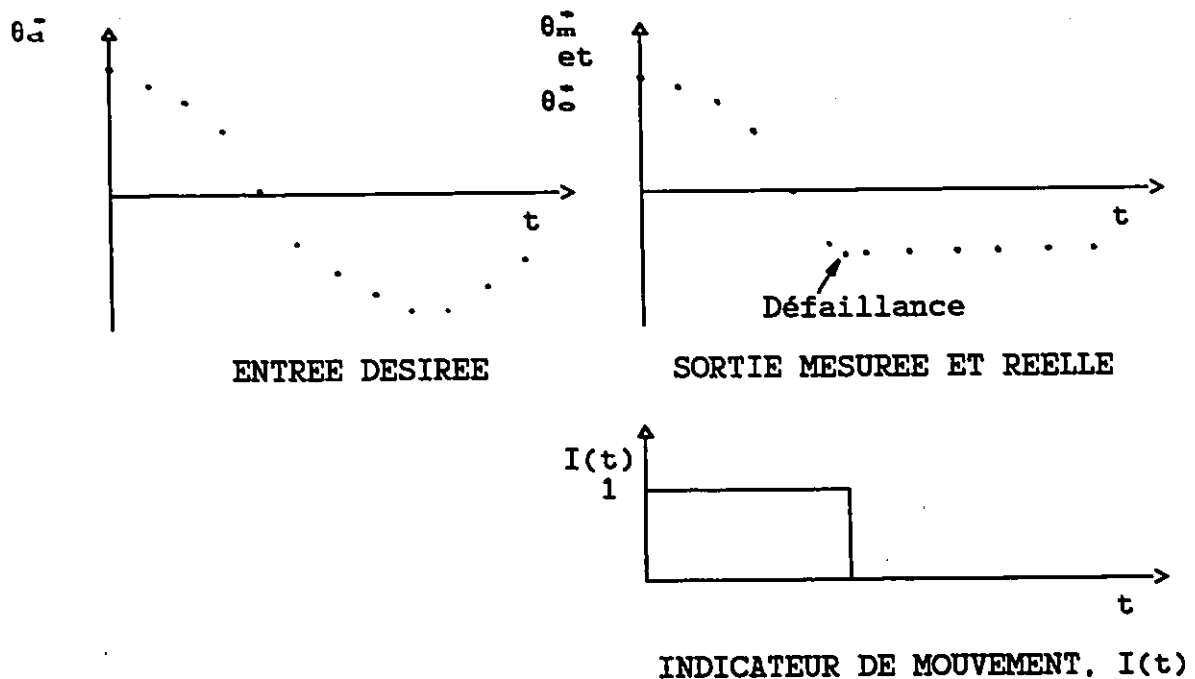


Figure 2.26: Défaillance du groupe 1.

Pour les défaillances du groupe 2, on obtient une réponse dépendante du type d'entrée comme le montre la figure 2.27 a).

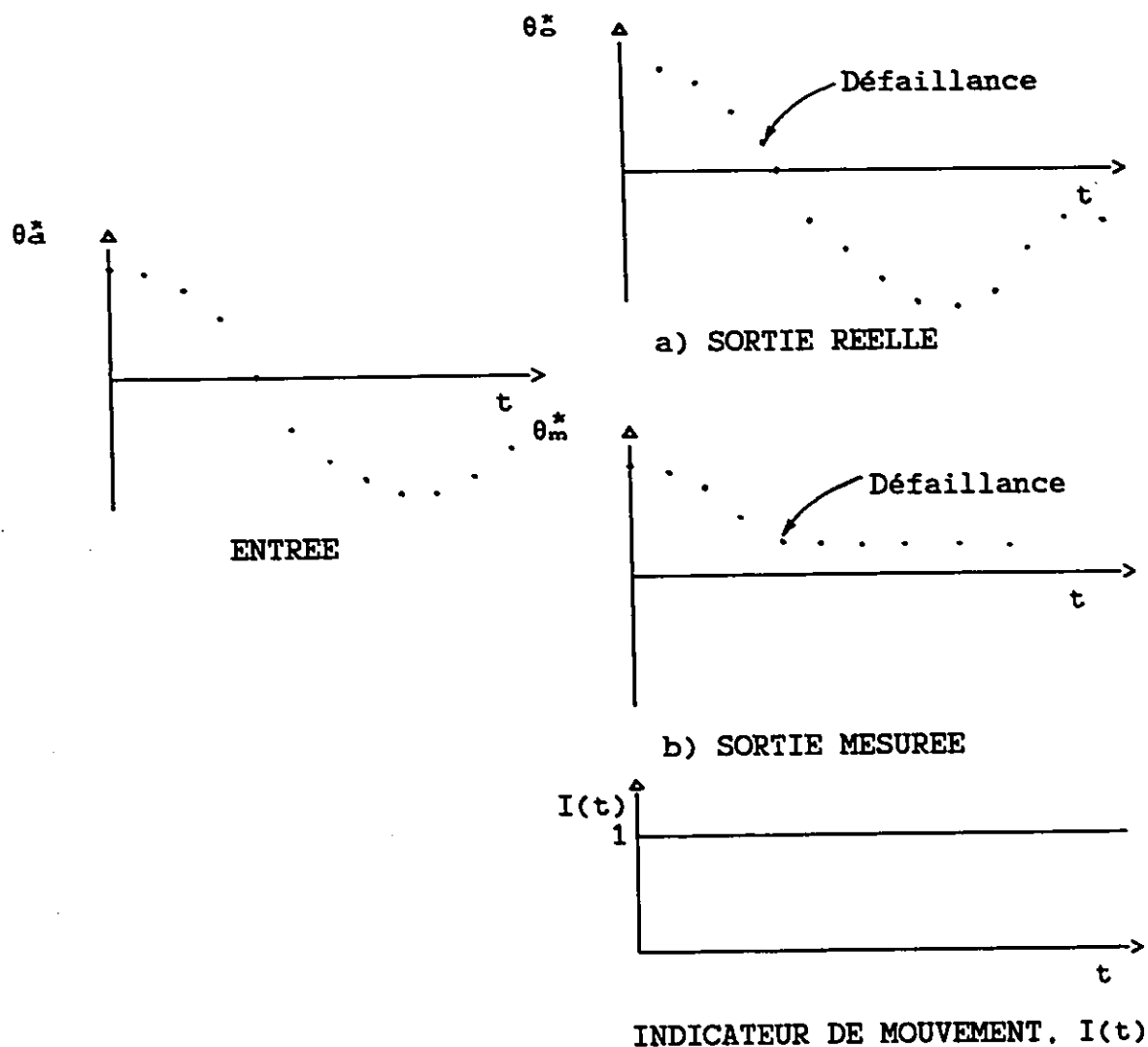


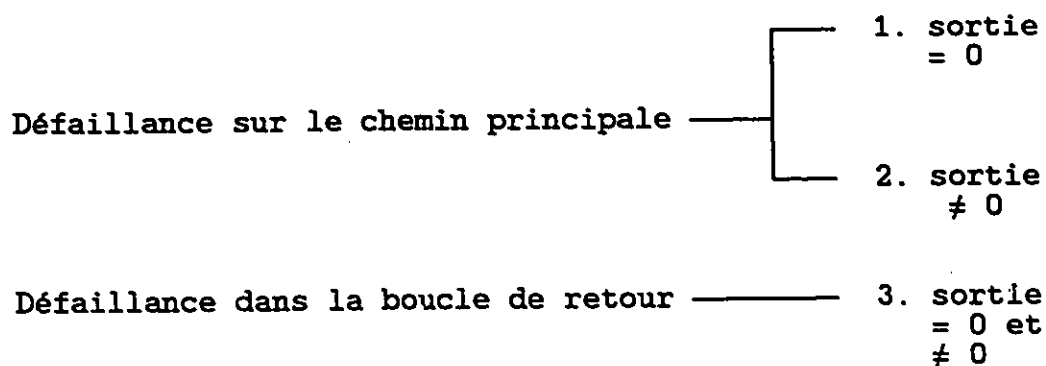
Figure 2.27: Défaillance du groupe 2, a) sortie réelle.
b) sortie mesurée.

La sortie mesurée est du même type que pour les défaillances du groupe 1, c'est-à-dire l'arrêt du moteur, car la sortie du lecteur de position est zéro quand le lecteur est en défaillance. Par contre, l'indicateur de mouvement est non-nulle. Ce qui signifie que le moteur tourne à une certaine vitesse même après la

défaillance, comme le montre d'ailleurs le signal de sortie réelle de la figure 2.27 a). Pour le diagnostic automatique, on procédera à la comparaison du signal d'entrée désirée, θ_a^* , et du signal de sortie mesurée, θ_m^* , pour déterminer s'il y a une défaillance et avec l'aide de l'indicateur de mouvement il sera possible de procéder à l'identification de la défaillance. Il ne serait pas possible dans le cas présent de procéder à la détection et l'identification des anomalies avec uniquement θ_a^* et θ_m^* . Il serait possible de le faire si on utiliserait θ_a^* et θ_s^* , mais ceci demanderait l'addition d'un second lecteur de position au système existant, ce qui n'est pas désirable.

2- Les défaillances avec sortie non-nulle.

Si on veut considérer tant les défaillances avec sortie nulle que non-nulle alors le même type de regroupement peut être utilisé. En fait, trois groupes seraient alors nécessaires, et ils peuvent être décrits comme suit:



Le groupe 1 et 3 étant les mêmes que les groupes 1 et 2 de la section précédente. Pour les défaillances du groupe 2, présenté ici, la réponse est indépendante du type d'entrée et peut être représenté par la figure 2.28. C'est-à-dire que si une composante du groupe 2 est en défaillance le moteur tournera à vitesse constante plus ou moins grande dépendamment de le grandeur du signal de sortie à la défaillance.

Il est à noter que les résultats présentés dans cette section, pour une sortie $\neq 0$, sont basés sur le programme de simulation et aucune expérience n'a été faite à ce moment pour en vérifier la validité. Des tests seraient donc souhaitables pour les défaillances avec sortie non-nulle avant de poursuivre.

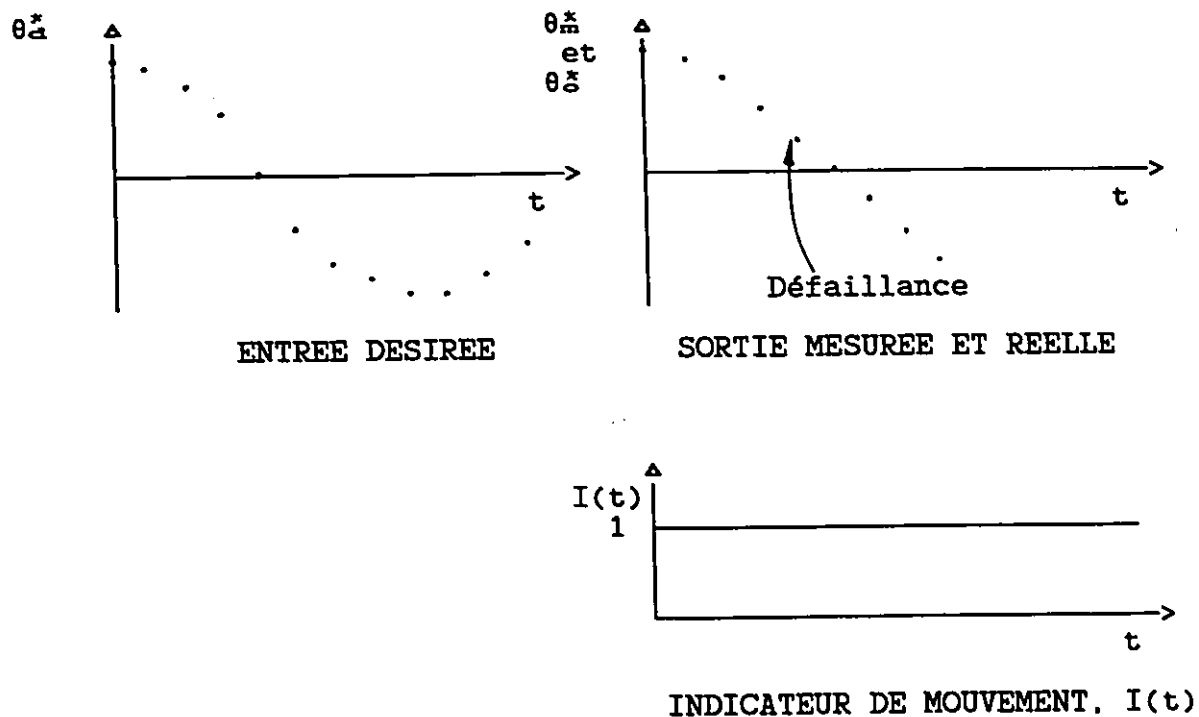


Figure 2.28: Défaillance du groupe 2 avec sortie $\neq 0$.

La considération des sorties non-nulles demande donc l'ajout d'un troisième groupe. Afin de simplifier l'algorithme de diagnostic et comme aucun résultats expérimentaux ne sont disponibles sur les défaillances avec sortie non-nulle, seulement les défaillances avec sortie nulle seront considérées dans un premier temps. Une fois la procédure bien éprouvée pour ce cas il sera plus facile d'implémenter le troisième groupe.

De façon générale, on peut conclure que le programme de simulation est un outil important lors du développement d'une procédure de diagnostic, il permet de recueillir la connaissance

le torque variable, $T_a(s)$, peut être obtenue en appliquant la règle de Mason à la figure 2.29.

$$\theta_o(s) = \frac{- 1/(J_o s + B_o) \cdot T_a(s)}{1 + 1/(L_m s + R_m) \cdot K_t \cdot 1/(J_o s + B_o) [D(s) \cdot D/A(s) \cdot A(s) + K_o]}$$

On peut réécrire cette équation sous la forme

$$\theta_o(s) = G_2(s) \cdot T_a(s) \quad (2.12)$$

La fonction de transfert totale du système peut-être obtenue en appliquant le principe de superposition à l'équation 2.12 et à la fonction de transfert du système pour une charge constante, équation 2.3.

$$\theta_o(s) = G(s) \cdot \theta_a(s) + G_2(s) \cdot T_a(s) \quad (2.13)$$

L'implémentation de l'équation 2.13 requière la connaissance des variations de torques en fonction de la configuration du robot. Un programme de simulation de la dynamique du robot serait nécessaire pour procéder au calcul de ces torques. Si les variations de torques sont connues, l'équation 2.13 s'implémente de la même façon que pour une charge constante.

3. LA DETECTION ET L'IDENTIFICATION DES ANOMALIES.

Ce chapitre présente les dernières étapes du processus de diagnostic, soit le développement de la méthode de détection et de la méthode d'identification des anomalies.

Le processus de diagnostic peut être schématisé par une boîte noire dans laquelle entrent les séquences de sortie du modèle et du système réel et de laquelle sort une décision, à savoir, s'il y a défaillance ou non et où est la défaillance.

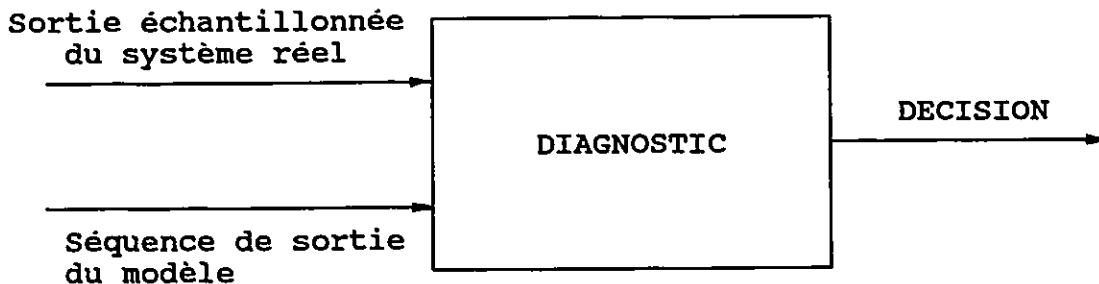


Figure 3.1: Schéma du processus de diagnostic.

De façon générale la comparaison de deux séquences se fait par l'entremise de techniques de reconnaissance des formes. Ces techniques se divisent en deux approches: 1- syntaxique et 2- statistique.

Dans la méthode syntaxique, un patron est divisé en sous-patrons qui sont à leur tour décrits par des patrons plus simple. Les patrons de bases (les plus simples) sont appelés des primitives. Des règles gouvernent la composition de patrons à partir de primitives. On cherche donc dans un patron particulier à retrouver les primitives et par les règles de composition on peut appareiller le patron inconnu à un patron connu.

Cette méthode est principalement utilisée pour la reconnaissance de caractères, empreintes digitales, etc.

L'approche statistique, se base sur un ensemble de mesures caractéristiques appelées traits qui sont tirés du patron inconnu. Ces traits sont généralement moins sensibles aux bruits parasites et distortions des signaux. La méthode se divise en deux sous-problèmes. Le premier sous-problème est de déterminer quels traits devraient être tirés du patron inconnu (comme la moyenne, la variance, autocorrélation, etc). Le deuxième sous-problème est la classification de la défaillance. La figure 3.2 illustre le processus typique de reconnaissance des formes par la méthode statistique.

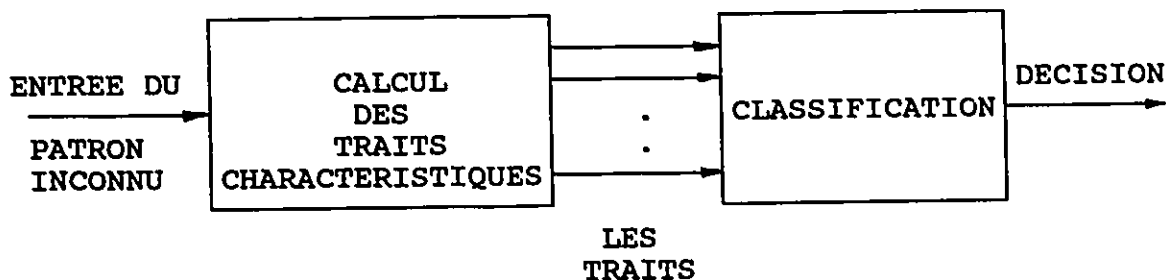


Figure 3.2: Processus typique de reconnaissance des formes.

Pour notre problème particulier, la séquence de sortie du modèle est le patron connu et la sortie échantillonnée du système de commande réel est le patron inconnu. On désire déterminer si le patron inconnu correspond au patron connu. Dans un tel cas la méthode statistique est plus appropriée et sera utilisée pour le diagnostic.

3.1 La détection des anomalies: la sélection des traits caractéristiques.

De façon générale, le nombre de traits doit être le plus bas possible pour faciliter l'implémentation et la classification des patrons.

La sélection des traits dépend en grande partie de l'application. Par exemple, K.S. Fu dans le chapitre 5 de [51] fait une analyse détaillée des différents traits qui peuvent être utilisés pour l'interprétation d'onde sismique. Il a montré que l'autocovariance donnait les meilleurs résultats (93% de reconnaissance parfaite). Le pourcentage de reconnaissance parfaite dépend aussi grandement de la méthode de classification choisie (celles-ci seront étudiées dans la section suivante). Un fait intéressant observé par Fu et par Gonzalez et Tou, [52], est que les meilleurs performances semblent avoir été obtenues avec des méthodes qui fonctionnaient dans le domaine fréquentiel.

Fu [51] conclut en disant que la meilleur méthode pour une application particulière peut être trouvée en testant et comparant les différentes méthodes.

Cela est donc l'approche qui a été suivie pour l'application qui nous intéresse. Un premier trait caractéristique a été sélectionné, soit la moyenne.

3.1.1 La moyenne.

La moyenne d'un signal échantillonné est donnée par l'équation suivante:

$$M_0 = \frac{1}{N} \sum_{i=1}^N X_i \quad (3.1)$$

où N = nombre d'échantillons
 X_i = la valeur de l'échantillon "i"

Des tests ont été faits avec le système de commande réel et le modèle. Sur 20 tests faits pour différentes entrées et différentes défaillances, ce critère a permis de détecter qu'il y avait défaillance dans 16 des 20 cas. La performance du critère de moyenne a été estimée à environ 80%.

En fait il semblait difficile de trouver une valeur seuil de la moyenne qui pouvait détecter les défaillances sans erreur pour différents types de signaux d'entrées, l'exemple de la section 3.4 illustre ce fait. Suite à ces résultats un deuxième critère (trait) a été introduit, soit la variance.

3.1.2 La moyenne et la variance

L'estimation de la variance d'un signal échantillonné est donnée par l'équation suivante:

$$V_0 = \frac{1}{(N-1)} \sum_{i=1}^N (X_i - M)^2 \quad (3.2)$$

La combinaison des deux critères a augmentée la performance à 90%, c'est-à-dire dans 18 des 20 cas testés, la combinaison des deux critères a permis de détecter correctement les défaillances. Mais un problème persistait, en effet, pour des signaux de grandeur relative très différentes, les critères perdent leurs validités. Prenons un exemple simple, deux fonctions sinusoidales discrètes illustrées à la figure 3.3. La moyenne du premier sinus est zéro par contre la moyenne du deuxième est 100.

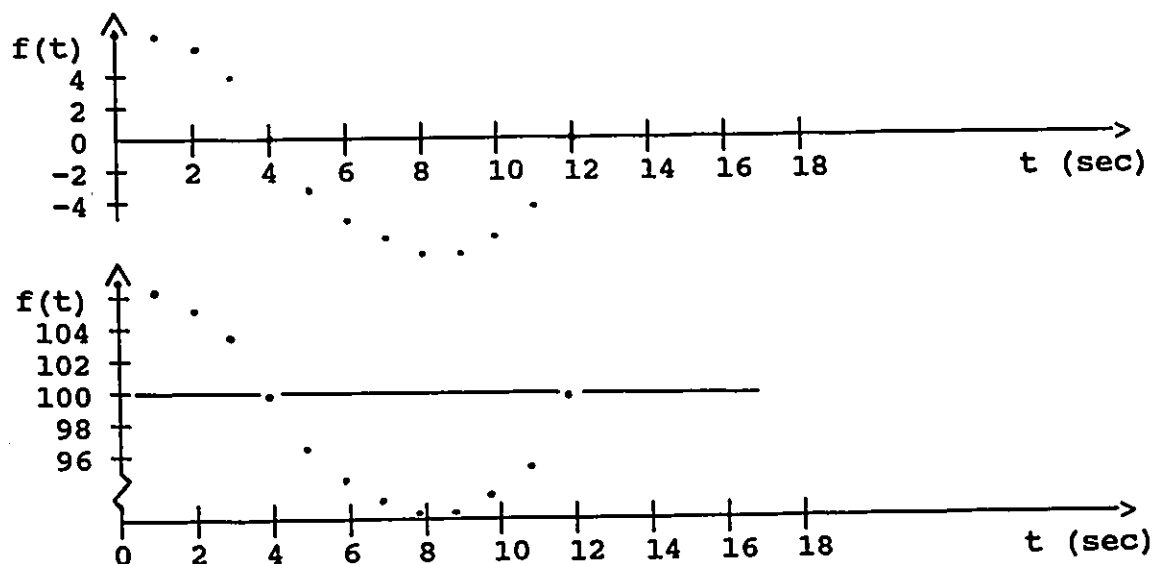


Figure 3.3: Deux fonctions sinusoidales

Afin de palier à ce problème, il a été décidé de normaliser les données avant le calcul de la moyenne et de la variance, réduisant ainsi toutes les valeurs à de même ordre de grandeur.

3.1.3 La moyenne et la variance des séquences normalisées

La première étape de la normalisation est de trouver le maximum pour chaque séquences et la deuxième est de diviser chaque valeurs, X_j , par ce maximum. Par la suite le moyenne et la variance sont calculées. La performance de cette procédure c'est montrée infailible pour tout les types d'entrées testées, soit, un échelon unitaire, une rampe et une entrée sinusoidale.

La procédure qui est très simple et efficace, a été implémentée en parallèle avec le système de commande réel. Plusieurs tests ont été nécessaire afin de déterminer les valeurs seuils à adopter. Pour l'erreur sur la moyenne entre la sortie échantillonnée du système réel et la séquence de sortie du modèle, la valeur seuil choisie est de,

$$\text{EPSIL1} = 0.5 \quad (\text{seuil sur la moyenne})$$

tandis que la valeur seuil choisie pour la variance est donnée par,

$$\text{EPSIL2} = 0.18 \quad (\text{seuil sur la variance})$$

Ces valeurs ont été déterminées expérimentalement pour différents types d'entrées (figure 3.6). Pour chaque types d'entrées, la moyenne et la variance ont été calculées lorsque le système fonctionne normalement à partir de la séquence de sortie du modèle et la sortie échantillonnée du système réel. La même procédure a été suivie pour le système en défaillance. En comparant les valeurs de moyenne et variance normales et en défaillance les valeurs seuils ont été sélectionnées. Elles ne sont pas justifiable autrement.

La figure 3.4 nous montre schématiquement la procédure adoptée pour la première partie du diagnostic: la détection des anomalies.

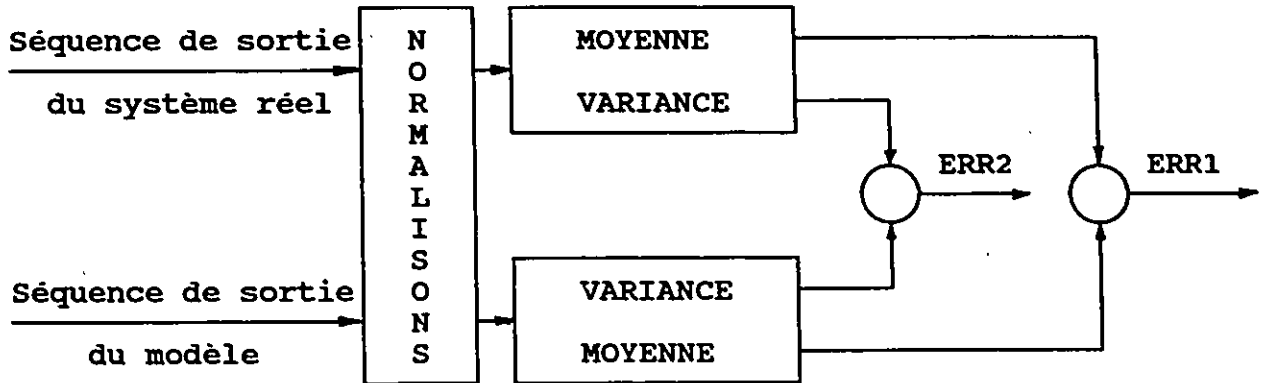


Figure 3.4: Détection automatique des anomalies.

3.2 L'identification (classification) des défaillances.

Beaucoup de documentation est disponible sur les méthodes traditionnelles de classification des patrons. K.S. Fu dans son livre [51] fait un résumé de toutes ces méthodes. Trois approches y sont identifiées sous lesquels se retrouvent les différentes méthodes.

- 1- Classification non-paramétrique.
 - fonction discriminante linéaire
 - classificateur de distance minimum
 - classificateur du plus proche voisin
 - fonction discriminante polynomiale
- 2- Fonction statistique discriminante.
 - Fonction des pertes.
 - Fonction discriminante de Bayes
 - Probabilité maximum
- 3- Analyses des amas (groupes)
 - par apprentissage
 - etc.

L.F. Pau [53] a utilisé la méthode du plus proche voisin avec succès pour le diagnostic des défaillances pour des pièces d'équipements électroniques. Quoique très intéressantes, aucune de ces méthodes traditionnelles n'a été retenue pour la présente application pour la raison suivante. Dans la section 2.4.3 i), il a été conclu que les défaillances avec sortie nulle se divisent en deux groupes. Les défaillances dans le groupe un, entraînent l'arrêt du moteur tandis que celles dans le groupes deux, causent un mouvement imprévisible. Dans ces deux cas, il serait très difficile d'associer la sortie échantillonnée du système réel à une défaillance particulière sans savoir à quel moment la défaillance se produit. Le programme de simulation des défaillances nous permet de connaître la sortie du système en défaillance, mais on doit préciser le moment ou la défaillance se produit. Parce que le moment ou la défaillance se produit est un inconnu dans le cas présent, il était impossible d'utiliser les méthodes décrites ci-haut. La figure 3.5 nous donne deux exemples pour la défaillance du lecteur optique. L'allure de la réponse du système dépend grandement de moment ou la défaillance se produit.

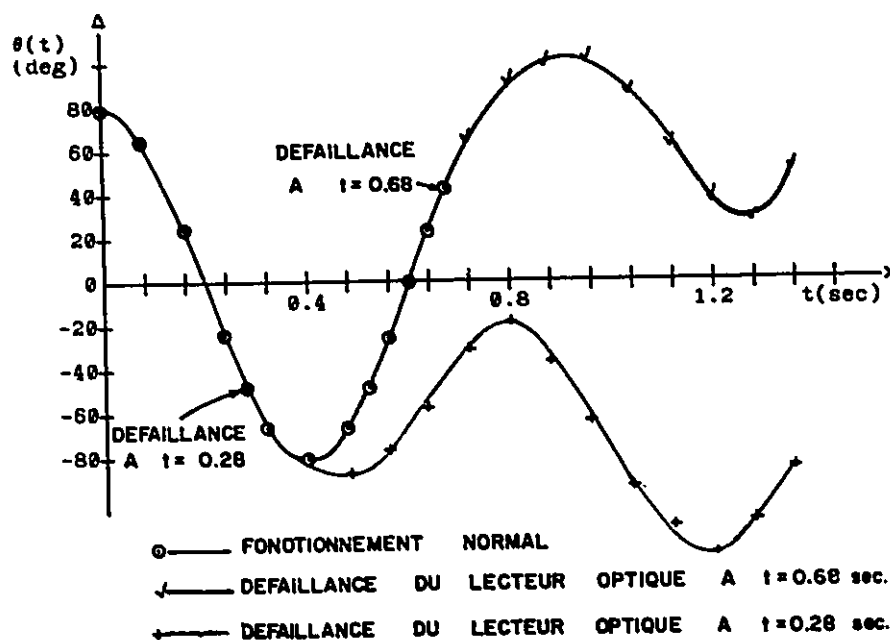


Figure 3.5: La défaillance de lecteur optique

Pour l'identification des défaillances, notre approche s'apparente plutôt à celle utilisée dans les systèmes experts. La procédure d'identification se divise en trois étapes; 1) la connaissance de système, 2) les règles régissant l'identification et 3) l'implémentation. Chacune de ces étapes seront présentées en détail. Il est important de rappeler que la méthode a été développée pour les défaillances en circuit ouvert avec sortie nulle.

3.2.1 La connaissance du système.

La connaissance du système est basée sur des tests sur le système de commande réel. Le fonctionnement normal du système a été étudié pour les entrées suivantes, représentées à la figure 3.6.

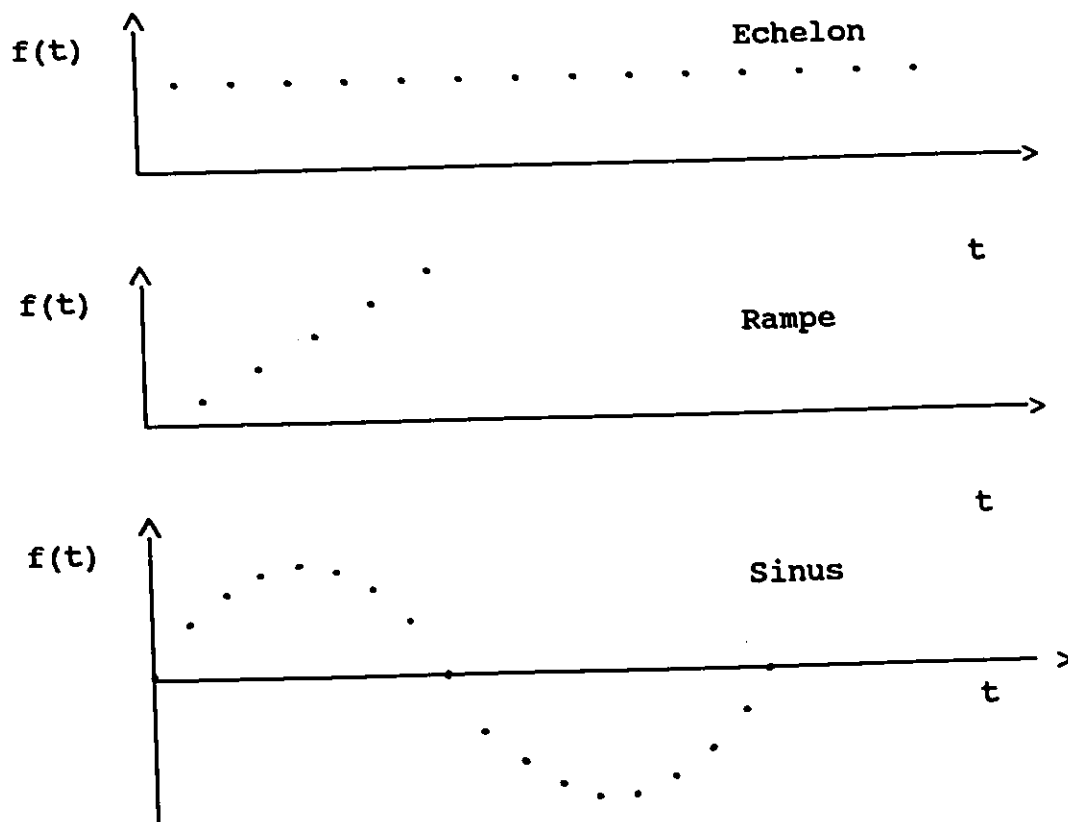


Figure 3.6: les types d'entrées utilisées.

Il n'a pas été jugé nécessaire d'utiliser d'autres types d'entrées car le servomécanisme d'un robot suit habituellement une trajectoire qui se compose de parties sinusoïdale, rampe et échelon. Afin de vérifier ceci, un programme de modélisation de trajectoire, développé par Mr. Mario Tétreault, a été utilisé. La méthode de modélisation utilisée est l'interpolation linéaire tel que dérivée par R.P. Paul [60]. Cette méthode est simple à implémenter et largement utilisée en milieu industriel. Une tâche a été spécifiée tel que montré à la figure 3.7. La tâche a été modélisée en spécifiant les points P_1 à P_4 , et la trajectoire du robot dans l'espace obtenue par l'algorithme de modélisation est montrée à la figure 3.8. A cette trajectoire dans l'espace cartésien correspond, pour chaque servomécanisme, une trajectoire dans l'espace angulaire. Pour le robot ASEA IRB 6/2 la trajectoire de chaque servomécanismes (il y en a 4) est montrée à la figure 3.9.

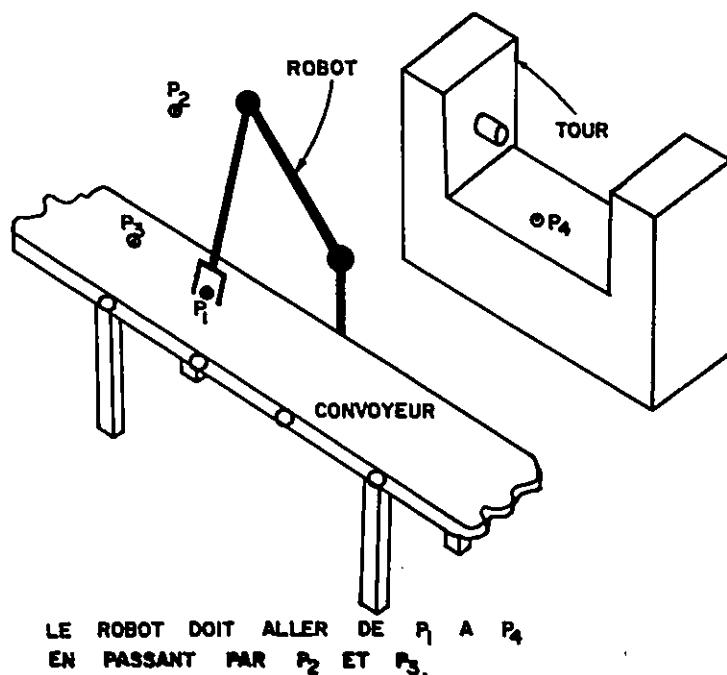


Figure 3.7: La tâche à accomplir.

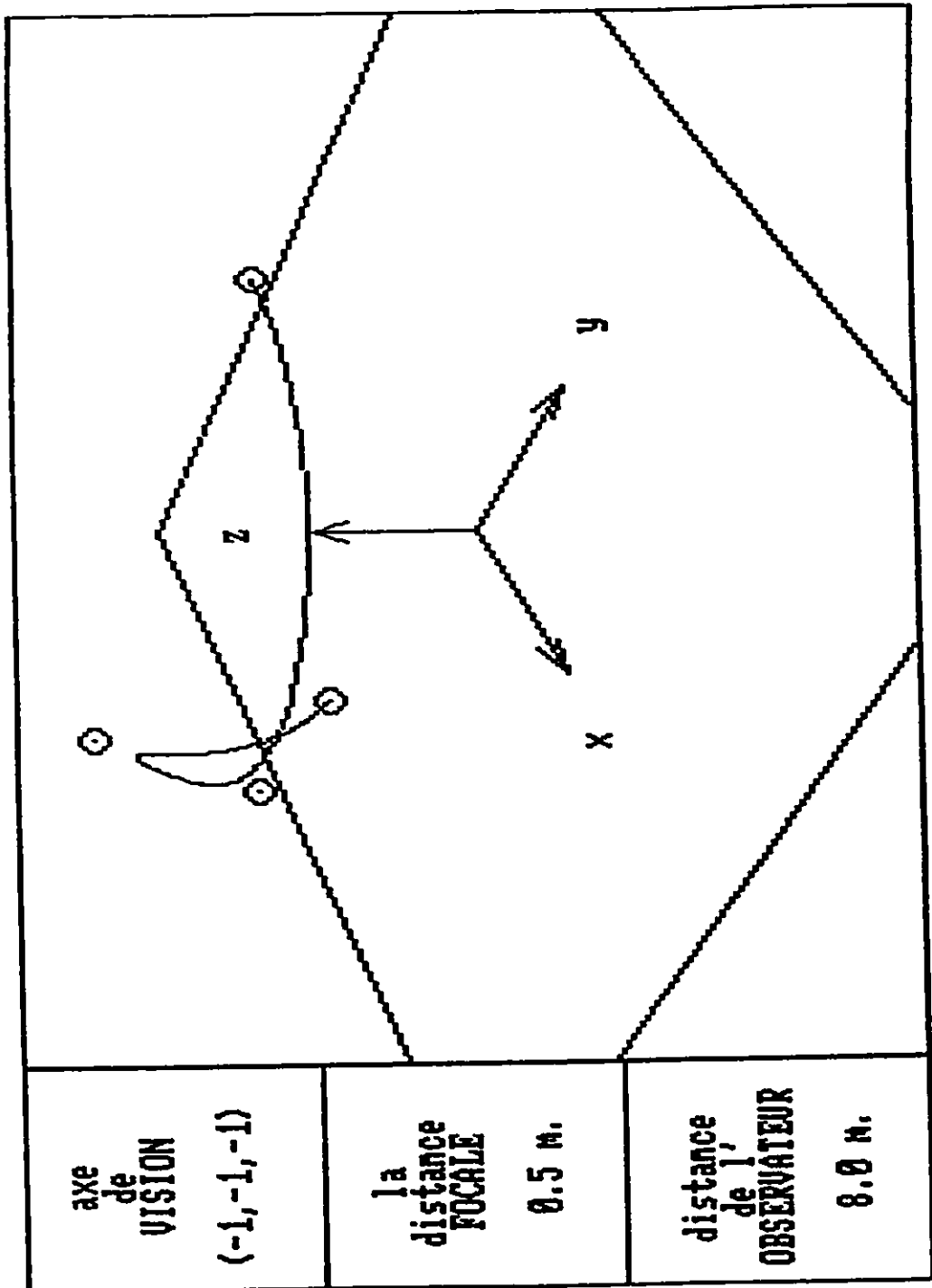


Figure 3.8: La trajectoire dans l'espace cartésien.

Cette trajectoire représente le déplacement de l'organe terminal dans l'espace cartésien.

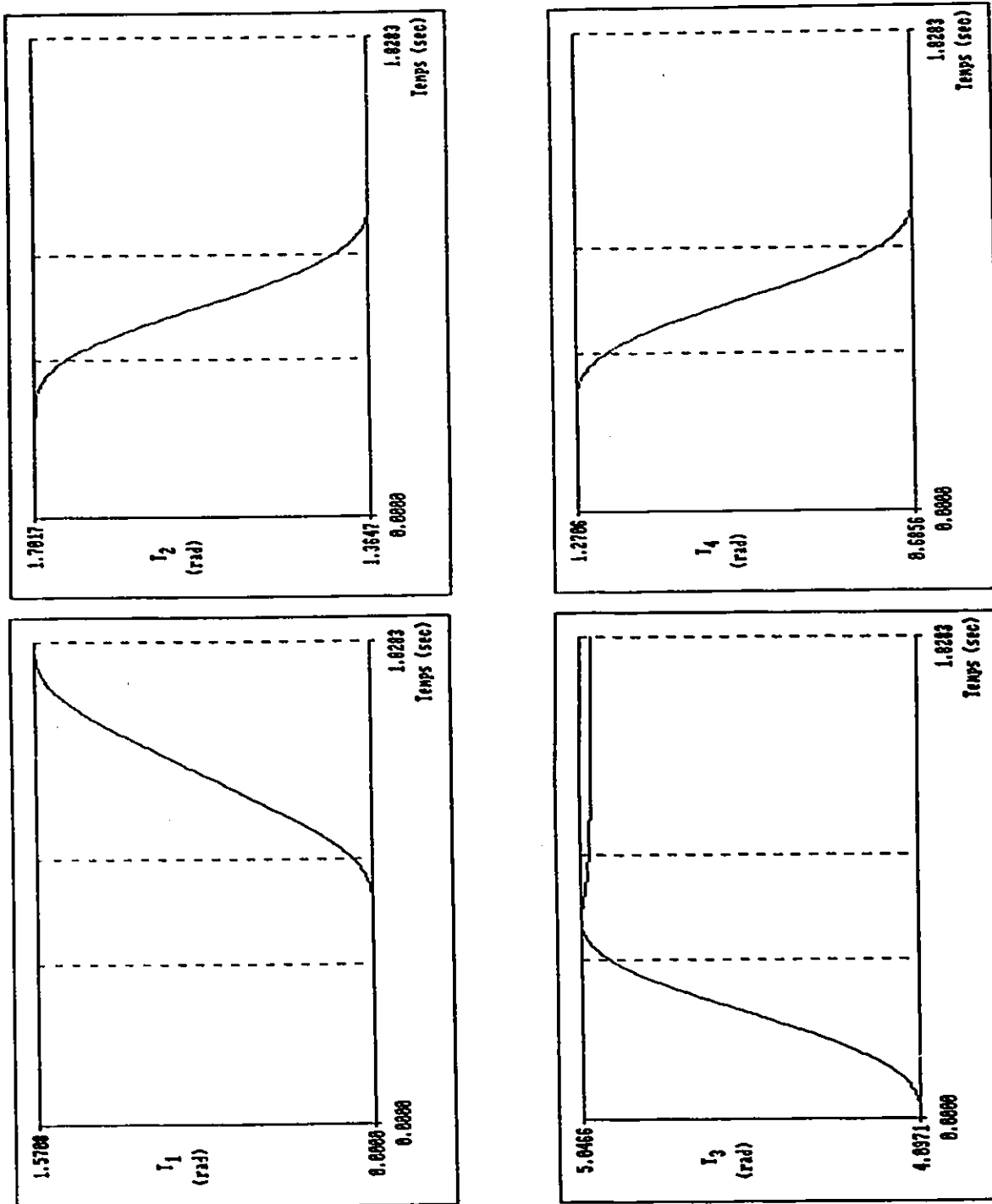


Figure 3.9: La trajectoire angulaire pour chaque servomécanismes du robot ASEA IRB 6/2.

A partir de la figure 3.9 on peut conclure que les trajectoires dans l'espace angulaire sont des portions de sinus, rampe et échelon.

Le chapitre deux nous a présenté les résultats des tests de simulation de défaillances sur le système de commande réel. Les défaillances de composantes physiques seulement seront considérées pour l'identification des défaillances. C'est à dire qu'on ne considère pas les défaillances du régulateur numérique et du comparateur. On va considérer uniquement les deux groupes de défaillances identifiées au chapitre deux pour une sortie nulle, soit les défaillances sur le chemin principale et dans la boucle de retour.

Pour ces deux groupes on peut exprimer la connaissance acquise comme suit:

1. Défaillance sur le chemin principale.
(défaillance du moteur, de l'amplificateur ou du D/A)
 - Effet: arrêt du moteur, le programme de commande du moteur conserve la dernière valeur enregistrée comme position du système.

2. Défaillance dans la boucle de retour.
 - Effet: le mouvement du système devient erratique, la sortie du lecteur de position devient nulle, comme cette sortie est nulle le programme de commande du moteur conserve la dernière valeur enregistrée comme la position du système.

C'est à partir de cette connaissance que les règles régissant l'identification seront établies.

3.2.2 Les règles d'identification.

Le programme de diagnostic reçoit l'information suivante du programme de commande du système réel.

1. La séquence d'entrée du système réel, θ_d^* .
2. La sortie mesurée du système réel, θ_m^* .
3. Indicateur de mouvement (1 si le moteur tourne et 0 s'il ne tourne pas)

L'indicateur de mouvement est nécessaire car lorsque le module optoélectronique est en défaillance la sortie mesurée du système réel est constante quoique le moteur tourne toujours. Ceci est due au fait que la sortie du module optoélectronique, qui est envoyée au programme de commande du moteur, devient nulle lorsqu'il est en défaillance. Donc pour le programme de commande le moteur ne bouge plus et est à la dernière position lue. Si cette position n'est pas la position désirée, alors le programme commande au moteur de tourner.

A partir de ces informations et de la connaissance que nous avons sur les défaillances du système, énoncée à la section précédente, on doit pouvoir identifier à quel groupe les défaillances appartiennent. Notez qu'on passe à l'étape d'identification des défaillances seulement si une défaillance a été détectée au préalable.

La procédure utilise la séquence de sortie mesurée du système réel et l'indicateur de mouvement. Pour la sortie mesurée du système réel on calcule la pente. Les règles sont montrées à la table 3.1.

3.3 Implémentation du processus de diagnostic.

Le processus de diagnostic a été implémenté pour fonctionner en parallèle avec le montage expérimental. Le programme est disponible dans l'annexe D et est schématisé à la figure 3.10.

Le processus se divise au départ en deux; A) La transformation du signal d'entrée et B) la transformation du signal de sortie échantillonnée.

Table 3.1: Règles d'identification

PENTE DE LA SORTIE REELLE MESUREE	MOUVEMENT DU MOTEUR	DEFAILLANCE
0	NON	La défaillance est dans le chemin principale.
0	OUI	La défaillance est dans la boucle de retour.
non 0	OUI ou NON	On ne peut pas identifier où est la défaillance

La partie A comprend quatre phases. Dans la première phase, A1, on lit 64 échantillons du signal d'entrée. La phase B1 est faites simultanément à la phase A1. Le nombre d'échantillons luent peut varier, mais doit être calculé par 2^n , "n" étant un entier, ce qui est nécessaire pour l'algorithme de FFT. Ici, 64 ($n = 6$) a été choisi parce qu'il donne une précision acceptable en un temps de calcul minimum. Plus le nombre d'échantillons est grand plus le temps de calcul de la transformée de Fourier directe et inverse sera long. Sur la figure 3.11, la phase A1 et B1 correspondent au segment i.

Dans la phase A2, la transformée de Fourier discrete directe est calculée, pour la séquence d'entrée, θ_d^* , par la sous-routine RFFT dans le programme de l'annexe D. La sortie du modèle est calculée en A3. La transformée de Fourier inverse discrète est ensuite effectuée en A4, pour donner la séquence de sortie du modèle dans le domaine temps. Pour cette séquence, la moyenne et la variance échantillonnée seront calculées en A5. Parallèlement aux phases A2 à A5, pour la phase B2 on calcul la moyenne et la variance de la sortie mesurée du système réel, θ_m^* .

Dans la partie C, on procède au calcul des erreurs sur la moyenne et la variance et un test est effectué pour déterminer s'il y a défaillance ou non. S'il y a défaillance, on passe à la

partie D, sinon on revient à A et B. Les phases A2 à A5 en parallèle avec la phase B2, de même que la partie C correspondent à la période de calcul i , sur la figure 4.11. Après la partie C, le processus de détection des anomalies est terminé et on peut passer à l'identification, s'il y a lieu.

Si une défaillance a été détectée en C alors on passe à la partie D, qui se divise en quatre phases. Dans la phase D1 on va lire, de nouveau, 64 échantillons du signal de sortie du système réel et on lit l'indicateur de mouvement. Si le moteur tourne cet indicateur prend une valeur de 1 sinon on a 0. La phase A1 correspond au segment $i+1$ sur la figure 3.11.

Dans la phase D2 on calcule la pente du signal lu en D1 et en D3 on applique les règles de classifications afin de déterminer la provenance de la défaillance. En D4, un message est envoyé à l'utilisateur pour lui signifier qu'il y a une défaillance dans le groupe 1 ou 2 et le programme arrête. Le programme ne prend aucune action préventive, cette section reste à développer. Les phases D2 à D4 correspondent à la période de calcul $i+1$ sur la figure 3.11.

Il n'est pas possible dans le processus de diagnostic d'utiliser une seule séquence d'échantillons pour la détection et l'identification des défaillances car, comme il a été expliqué à la section 3.2, l'instant où la défaillance se produit n'est pas connu. C'est pour cette raison que lorsqu'une défaillance est détectée dans la partie C (figure 3.10), on doit aller lire, en D1, une nouvelle séquence de sortie du système réel.

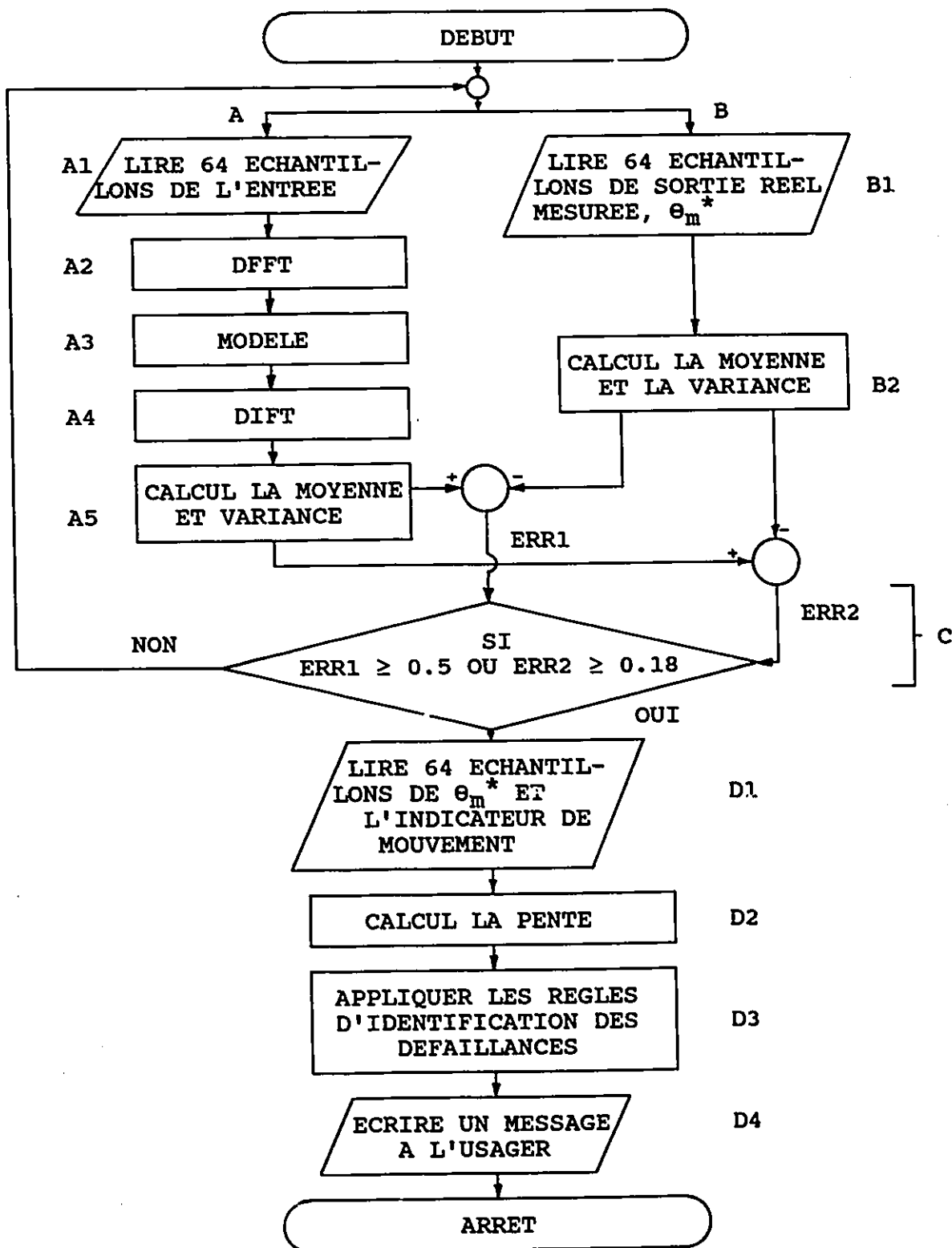


Figure 3.10 : Le processus de diagnostic complet.

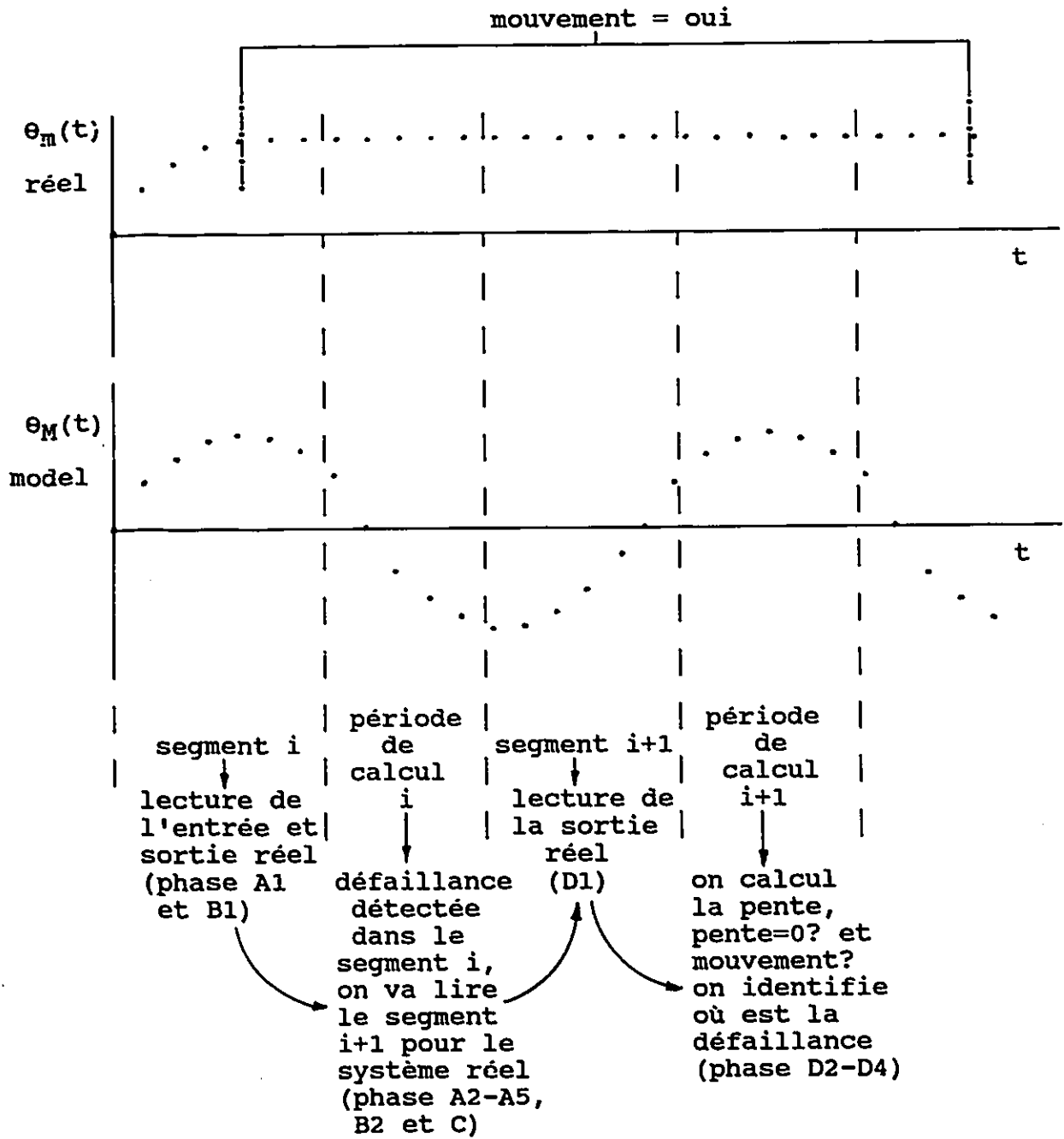


Figure 3.11. Synchronisation du processus de diagnostic.

Le programme de diagnostic s'est avéré très efficace pour détecter et identifier les défaillances appartenant au groupe 1 et 2 pour une sortie nulle. Dans tout les cas étudiés, le programme a identifié adéquatement la défaillance. Le temps entre le moment où la défaillance arrive et l'affichage à l'écran d'un message identifiant l'erreur s'est avéré toujours ≤ 5 secondes même si le programme n'a pas été écrit dans le but de minimiser le temps de calcul nécessaire. La section suivante présente un exemple des entrées et des valeurs calculées par le programme de diagnostic pour une entrée sinusoidale.

3.4 Exemple des entrées et valeurs calculées par le programme de diagnostic pour une entrée sinusoidale.

Pour cet exemple, la défaillance du lecteur optique a été simulée dans le segment 2. C'est-à-dire que pour le premier segment le système fonctionnait normalement et la défaillance a été causée dans le second segment. Une entrée de type sinusoidale a été utilisée. Pour le segment 1 la table 3.2 nous donne la séquence d'entrée, la sortie échantillonnée du système réel de même que la séquence de sortie du modèle. La table 3.3 nous montre la moyenne et la variance calculée pour la sortie réelle et celle du modèle.

Pour le segment 2, la table 3.4 présente l'entrée, la sortie réelle et la sortie du modèle. On voit bien l'instant où la défaillance se produit. La table 3.5 nous montre la moyenne et la variance pour la sortie réelle et la séquence de sortie du modèle pour le segment 2. Cet exemple est intéressant car il montre l'utilité d'avoir deux critères différents. Pour le cas présenté l'erreur sur la moyenne est plus petite que la valeur seuil de 0.5 alors ce critère seulement ne permettrait pas de détecter la défaillance. C'est la raison pour laquelle un critère de variance est aussi utilisé.

Pour le segment trois, seulement la sortie du système réel est lue par le programme et montré à la table 3.6 avec l'indicateur de mouvement. Comme l'indicateur a la valeur de 1 alors le moteur tourne. Pour les valeurs de la table 3.6 la pente de la sortie mesurée, θ_m^* , sera calculée et les règles d'identification de la table 3.1 seront appliquées. Le message indiquant la défaillance de la boucle de retour sera alors affiché à l'écran.

TABLE 3.2: POUR LE SEGMENT 1: ENTREE, SORTIE REELLE ET SORTIE DU MODELE.

ENTREE $\theta_d(t)$ (deg)	SORTIE REELLE MESUREE $\theta_m(t)$ (deg)	SORTIE DU MODELE $\theta_M(t)$ (deg)
36	36	33
43	43	39
50	50	45
58	58	57
65	58	68
72	65	66
72	65	67
72	65	67
72	65	59
72	68	59
65	58	52
58	58	45
50	58	37
43	50	31
35	36	16
22	22	9
14	14	9
0	0	-16
-14	-14	-23
-22	-22	-37
-35	-29	-45
-43	-43	-41
-50	-50	-59
-58	-58	-66
-65	-65	-71
-72	-65	-72
-72	-65	-72
-72	-65	-72
-72	-65	-72
-72	-58	-68
-65	-58	-59
-58	-50	-52
-50	-43	-44
-43	-35	-38
-35	-29	-23
-22	-22	-15
-14	-14	0
0	0	8
14	7	18
22	35	31
35	43	38
43	50	48
50	58	52
58	58	58
65	65	67
72	65	66
72	65	67
72	65	67
72	58	60
72	58	52
65	58	45
58	50	37
50	43	31
43	35	25
35	29	16
22	22	9
14	14	9
0	0	-16
-14	-22	-23
-22	-29	-37
-35	-43	-45
-43	-50	-55
-50	-58	-64
-58	-65	-68

TABLE 3.3: SEGMENT 1: LA MOYENNE ET LA VARIANCE

	SORTIE REELLE MESUREE NORMALISEE	SORTIE DU MODELE NORMALISEE	ERREUR	SEUIL
MOYENNE	0.15	0.06	0.09	0.5
VARIANCE	0.63	0.46	0.07	0.18

TABLE 3.4: POUR LE SEGMENT 2: ENTREE, SORTIE REELLE ET SORTIE DU MODELE.

ENTREE $\theta_d(t)$ (deg)	SORTIE REELLE $\theta_o(t)$ (deg)	SORTIE DU MODELE $\theta_m(t)$ (deg)
36	36	33
43	43	39
50	50	45
58	58	67
65	58	68
72	65	68
72	65	67
72	65	67
72	72	59
72	72	59
65	65	52
58	58	45
50	58	37
43	50	31
35	43	16
22	36	9
14	8	0
0	-7	-16
-14	-7	-23
-22	-7	-37
-35	-7	-45
-43	-7	-41
-50	-7	-69
-58	-7	-66
-65	-7	-71
-72	-7	-72
-72	-7	-72
-72	-7	-72
-72	-7	-72
-72	-7	-66
-65	-7	-59
-58	-7	-62
-50	-7	-44
-43	-7	-38
-35	-7	-23
-22	-7	-15
-14	-7	0
0	-7	8
14	-7	16
22	-7	31
35	-7	38
43	-7	45
50	-7	62
58	-7	69
65	-7	67
72	-7	66
72	-7	67
72	-7	67
72	-7	60
72	-7	52
65	-7	45
58	-7	37
50	-7	31
43	-7	25
35	-7	16
22	-7	9
14	-7	0
0	-7	-16
-14	-7	-23
-22	-7	-37
-35	-7	-45
-43	-7	-66
-50	-7	-64
-58	-7	-66

TABLE 3.5: SEGMENT 2: LA MOYENNE ET LA VARIANCE

	SORTIE REELLE MESUREE NORMALISEE	SORTIE DU MODELE NORMALISEE	ERREUR	REUIL
MOYENNE	0.12	0.06	0.06	0.5
VARIANCE	0.18	0.46	0.31	0.18

TABLE 3.6: POUR LE SEGMENT 3: SORTIE REELLE DU SYSTEME

SORTIE REELLE $\theta_0(t)$ (deg)	INDICATEUR DE MOUVEMENT
-7	1
-7	
-7	
-7	
-7	PENTE CALCULEE
-7	0
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	
-7	

3.5 Effet de la défaillance du tachymètre sur la méthode de diagnostic développée.

Pour la méthode développée dans ce chapitre il a été supposé que le tachymètre utilisé comme indicateur de position ne pouvait entrer en défaillance. Cette hypothèse est logique étant donné que nous avons mentionné à la section 2.4.1 (page 40) que nous n'étudions pas les défaillances multiples. Il est toutefois intéressant de se demander quel serait l'effet de la défaillance du tachymètre sur la méthode développée.

Le tachymètre est utilisé comme indicateur de mouvement afin de savoir si le moteur tourne ou non. L'état du tachymètre n'affecte pas le processus de détection des anomalies, car l'indicateur de mouvement n'est pas utilisé dans cette étape.

Par contre la procédure d'identification des anomalies est affectée par l'état du tachymètre. Deux situations peuvent être observées; 1) la défaillance en circuit ouvert avec sortie nulle et 2) la défaillance en circuit ouvert avec sortie non-nulle.

Dans le premier cas la sortie du tachymètre est toujours zéro, c'est-à-dire que l'indicateur de mouvement aura la valeur de 0 en tout temps. Toutes les défaillances, quelles soient du groupe 1 ou 2 en réalité, seront donc identifiées comme faisant partie du groupe 1. Cette conclusion est facile à obtenir en regardant les règles d'identification énumérées à la table 3.1.

Dans le deuxième cas on aura la situation inverse, toutes les défaillances seront identifiées comme faisant partie du groupe 2.

Afin de palier à ce problème, un critère supplémentaire serait nécessaire pour que les défaillances du tachymètre n'affectent pas le processus d'identification des anomalies.

3.6 Effets des défaillances du régulateur sur la méthode développée.

Pour le présent projet, les défaillances du régulateur PID ne sont pas considérées dans la méthode de diagnostic. Ces défaillances s'avèraient en effet très difficile à simuler en temps réel parce que le régulateur utilisé était digital. Il était nécessaire de modifier le programme de commande ce qui était impossible lorsque le système était en opération. L'utilisation d'un régulateur analogique permettrait une plus grande flexibilité de ce côté là. Toutefois à partir de l'étude théorique des défaillances faites au chapitre précédent il est possible de prévoir la réaction du système aux défaillances du régulateur.

Les défaillances de la partie proportionnelle du régulateur causent, à plus ou moins long terme, un mouvement erratique du moteur. Ceci un peu comme pour les défaillances du lecteur optique. Une telle défaillance se classerait donc dans le groupe 2, mais il serait impossible de l'identifier avec la procédure telle qu'utilisée pour le lecteur optoélectronique. En effet, pour cette défaillance la sortie mesurée du système n'est pas en accord avec la figure 2.28, car si on suppose des défaillances uniques le lecteur optique fonctionnerait, donc la sortie mesurée ne pourrait être constante comme à la figure 2.28.

Les défaillances de la partie intégrale causent une augmentation de l'erreur en régime permanent. Pour une sortie nulle ou près de zéro lors de la défaillance, celle-ci ne pourrait être identifiée car le système n'est pas assez sensible. La défaillance passerait donc inaperçue. Pour une sortie significativement plus grande que zéro alors on pourrait classer la défaillance dans le groupe deux, sauf que, comme pour la partie proportionnelle il serait impossible de l'identifier avec la méthode telle quelle.

Il est difficile de faire des commentaires sur les effets de la défaillance de la partie différentiel car les résultats des tests effectués ne nous ont pas permis de tirer des conclusions valables, à cause des types d'entrée utilisée.

Afin de tenir compte des défaillances du régulateur dans la méthode de diagnostic des modifications majeures seraient nécessaires dans la procédure d'identification des anomalies. La procédure de détection des anomalies serait toutefois valide telle qu'utilisée pour tout les types de défaillances.

4. CONCLUSIONS ET RECOMMANDATIONS

4.1 Conclusions

Le diagnostic de défaillances pour des systèmes complexes est présentement fait par des méthodes du genre de BIT (Built In Test). Ces méthodes sont principalement efficace pour des circuits électroniques digitaux basés sur un microprocesseur. Leurs désavantages majeures est la nécessité d'arrêter le processus pour l'introduction de signaux externes qui permettent de faire le diagnostic. De plus, elles requièrent des équipements supplémentaires coûteux. La tendance actuelle montre un vif intérêt pour les systèmes dit experts qui permettraient de faire le diagnostic en temps réel et sans interférer avec le fonctionnement normal du système. Suivant cette tendance, l'auteur a développé un processus de diagnostic automatique des anomalies qui combine les méthodes classiques et les systèmes experts. Le diagnostic se fait en parallèle avec le système à surveiller et n'interfère en aucun temps avec celui-ci. De plus ce processus ne requière qu'un minimum d'équipements supplémentaires, un microprocesseur.

Dans un milieu industriel automatisé, de telles méthodes permettent d'augmenter la productivité à son maximum. Une étude faite pour une chaîne de montage simple a en effet montrée que le diagnostic automatique permettait de diminuer les temps de réparations et d'augmenter significativement l'efficacité de la chaîne. De plus, l'action préventive qui pourrait être prise permettrait de diminuer le fréquence des bris.

Le programme de diagnostic développé se divise en deux parties: la détection automatique d'une anomalie dans le système et l'identification de la composante ou du groupe de composantes qui l'a produite.

La première partie utilise un modèle linéaire, d'entrée-sortie pour représenter le système de commande à surveiller. Ce modèle est basé sur le graphe de fluence modifié du système. La validité du modèle a été montrée par la comparaison avec les résultats d'un montage expérimental construit à cet effet. Ces résultats nous montre qu'un modèle linéaire intégral-différentiel peut être utilisé pour un système en défaillance. Cette conclusion est très importante car Andow et Galluzzo [16] avait montré que la représentation algébrique linéaire de l'arbre de vérité n'était pas valide pour des systèmes contenant des composantes intégrales et différentielles

La méthode d'identification des anomalies est basée sur la connaissance du système acquise par des tests de défaillances sur le montage expérimental. Cette approche utilise le principe des systèmes experts, sans toutefois en utiliser la sémantique et le style de programmation propre à l'intelligence artificiel.

Le programme de diagnostic a été implémenté en parallèle avec le montage expérimental. La performance du programme s'est avérée parfaite pour les cas étudiés, soit des entrées de types échelon, rampe, sinus. Mais des tests plus avancés dans un milieu réel (sur un vrai robot) seraient souhaitable avant de pouvoir affirmer généralement une performance parfaite.

Les limites de la méthode développée peuvent être résumées comme suit:

- 1- Charge fixe.
- 2- Défaillances en circuit ouvert avec un signal nul à la sortie de la composante en défaillance.
- 3- Une seule défaillance à la fois.
- 4- On ne considère pas les défaillances du tachymètre. (à cause de 3).
- 5- On ne considère pas les défaillances à l'intérieur de partie digitale du système.

Malgré ces limites, la présente méthode de diagnostic nous démontre bien la faisabilité d'une telle procédure et nous donne la base nécessaire au développement d'une méthode plus complexe et complète. Comme très peu d'informations expérimentales étaient disponible sur le sujet, ce projet était une étape essentielle à la compréhension du problème de détection automatique des anomalies pour les systèmes de commande.

En complément au programme de diagnostic un programme de simulation des défaillances a été développé. Ce programme permet, au niveau de la conception d'un système de commande, de vérifier le comportement de celui-ci face à certaines défaillances. Les résultats obtenus peuvent aider à concevoir des systèmes qui seront plus résistants aux défaillances, donc mieux adaptés à leurs environnements.

Les résultats du programme de simulation ont été comparés aux résultats de tests sur le montage expérimental qui justifient ainsi la validité du programme de simulation. Ce programme permet aussi d'étudier les défaillances de systèmes opérationnels sans toutefois avoir à arrêter ceux-ci.

4.2 Recommandations

La présente recherche a permis de recueillir toutes les informations pertinentes nécessaire à la poursuite du projet. C'est une étape essentielle à l'implémentation d'un tel processus sur une plus grande échelle. Les résultats obtenus nous permettent d'entrevoir des possibilités fort intéressantes, possibilités qui seront présentées dans les paragraphes qui suivent.

La prochaine étape consiste à modifier le modèle et le montage expérimental pour tenir compte de charges variables, tel que présenté à la section 3.5. A la limite, le but est d'implémenter la méthode de diagnostic sur un système-robot industriel.

Dans la même ligne d'idée, il faudrait ajouter une boucle de retour de vitesse au système de commande déjà en place. Avec cette modification, le système serait presque identique au servomécanisme du robot ASEA IRB 6/2. L'ajout de cette boucle supplémentaire offre des possibilités intéressantes. Il serait possible d'utiliser la boucle de retour de vitesse pour compenser lors d'une défaillance de la boucle de retour de position et vice-versa. Ceci est possible, car les deux organes de mesure sont indépendants sur la plupart des systèmes-robots. Comme pour le programme de diagnostic développé ici, dans un premier temps, on vérifie s'il y a défaillance, si oui et si la défaillance est identifiée comme étant dans la boucle de retour de position alors on passe en mode de compensation. Le signal de vitesse serait intégré et substitué au signal défaillant du lecteur de position comme le montre la figure 4.1.

L'addition d'un tel compensateur serait très importante pour la sécurité du personnel et de l'environnement. Les tests effectués sur le montage expérimental ont en effet démontrés que lors de la défaillance du lecteur optique, le mouvement devient erratique pouvant ainsi causer des collisions avec l'environnement. L'ajout du compensateur pourrait prévenir de tels accidents et le robot pourrait continuer sa tâche jusqu'à ce que les personnels qualifiés puissent entreprendre les réparations nécessaires.

Il serait aussi possible de concevoir un système de commande qui se compenserait par lui même, qui deviendrait un système adaptatif lorsqu'une défaillance serait présente. Ceci pourrait être fait en variant les valeurs des constantes du régulateur en fonction du type de défaillance détectée [61].

En vue d'une implémentation sur une grande échelle, il serait aussi intéressant d'investiguer des méthodes de détection des anomalies plus complexe.

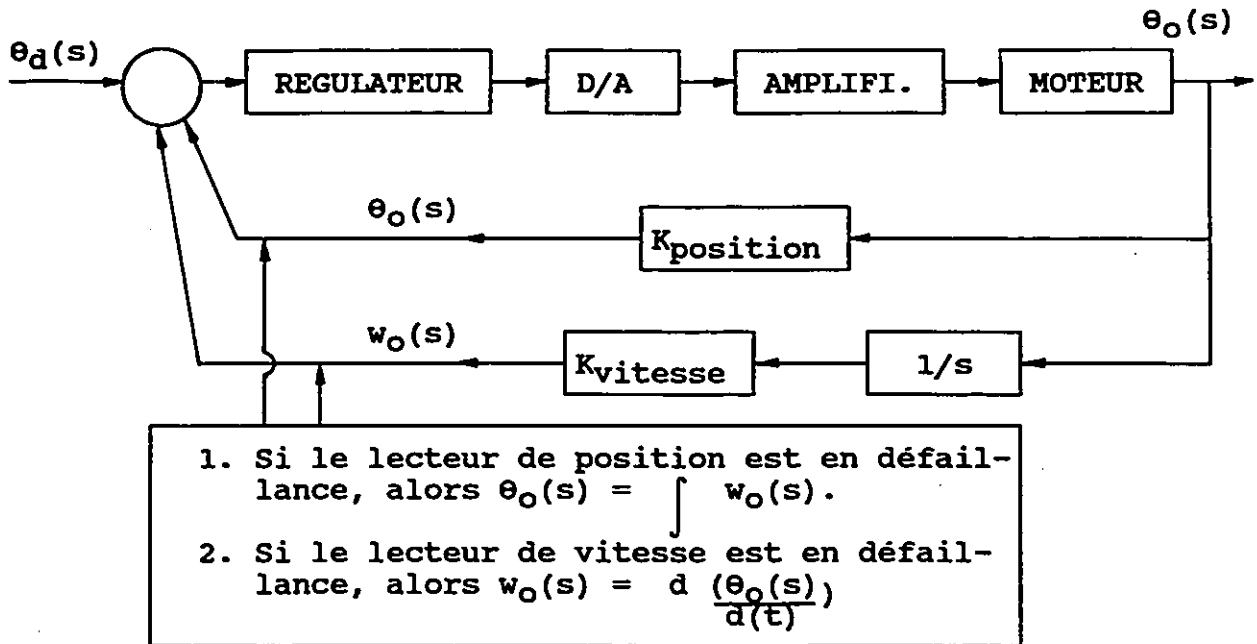


Figure 4.1 : Compensateur de vitesse et position.

Les critères de moyenne et de variance utilisés ici ne seront peut être pas suffisant à l'échelle d'un système-robot. Des critères dans le domaine de Fourier apporterait des informations supplémentaires sur l'état du système. Comme par exemple la densité spectral, autocovariance, autocorrelation, etc.

Un fait intéressant a été observé en faisant l'analyse théorique du système et serait peut-être utilisable pour la détection et l'identification de certaine défaillance. En fonctionnement normale, le système a trois pôles et deux zéros. Par contre, lorsqu'il y a défaillance du D/A, de la partie intégrale ou différentielle du régulateur le nombre de pôles et de zéros est différents. Dans le cas du D/A obtient deux pôles et pas de zéros. Pour la partie intégrale on a deux pôles et un zéros et pour la partie différentielle on a 3 pôles et un zéro. Ceci a été observé à partir de l'analyse théorique des défaillances. S'il était possible que ces faits se reflètent sur le système réel et qu'ils pourraient être identifiés alors un tel critère serait intéressant.

En reconnaissance de la parole des procédures similaires sont utilisées pour identifier les voyelles ou consonnes avec les spectres de Fourier.

La dernière étape de ce projet serait l'implémentation sur le système-robot ASEA IRB 6/2 du processus de diagnostic pour en juger la performance sur une grande échelle et dans un milieu intégré à un environnement. Le projet présenté ici, forme la base de cet implémentation et permet de juger de sa faisabilité.

ANNEXE A
DESCRIPTION GENERALE
DU
MONTAGE EXPERIMENTAL

DESCRIPTION GENERALE DU MONTAGE EXPERIMENTAL

Le système, à être utilisé, devait satisfaire deux critères spécifiques.

1. Tout les paramètres du système devaient être disponible et facilement accessible.
2. Le comportement du système devait être entièrement prévisible, i.e. en accord avec la théorie.

Aucun système répondant à ces deux conditions n'étaient disponible sur le marché. Un système de commande simple a donc été construit. La figure A.1a montre une photo du montage utilisé. Le schéma du montage, montré à la figure A.1b, comprend; un organe actionneur et transmetteur, un circuit de commande pour l'organe actionneur, un microprocesseur, un système de lecture et de conditionnement des données, un système d'acquisition des données, et le système d'alimentation. L'annexe E présente les schémas de construction du montage.

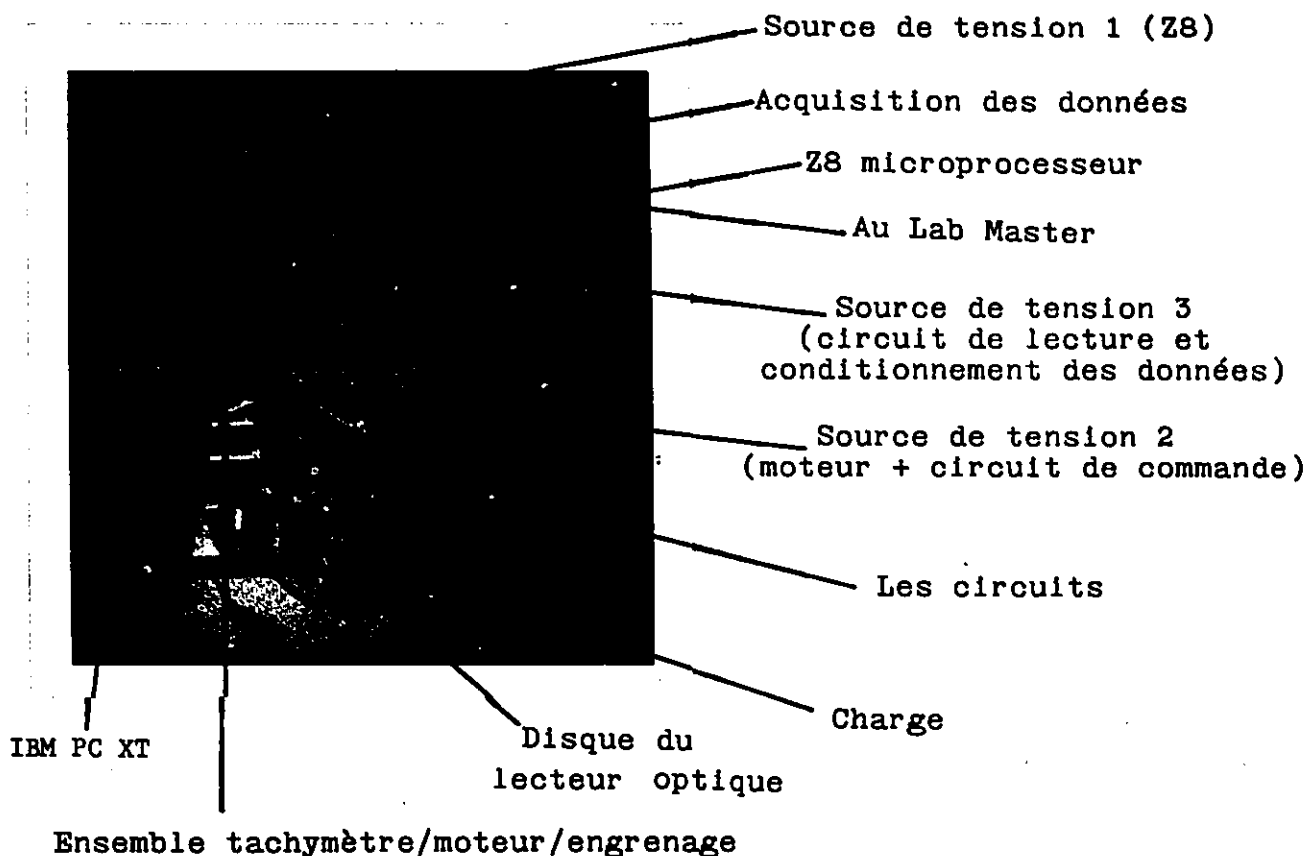


Figure A.1a: Photo du montage expérimental.

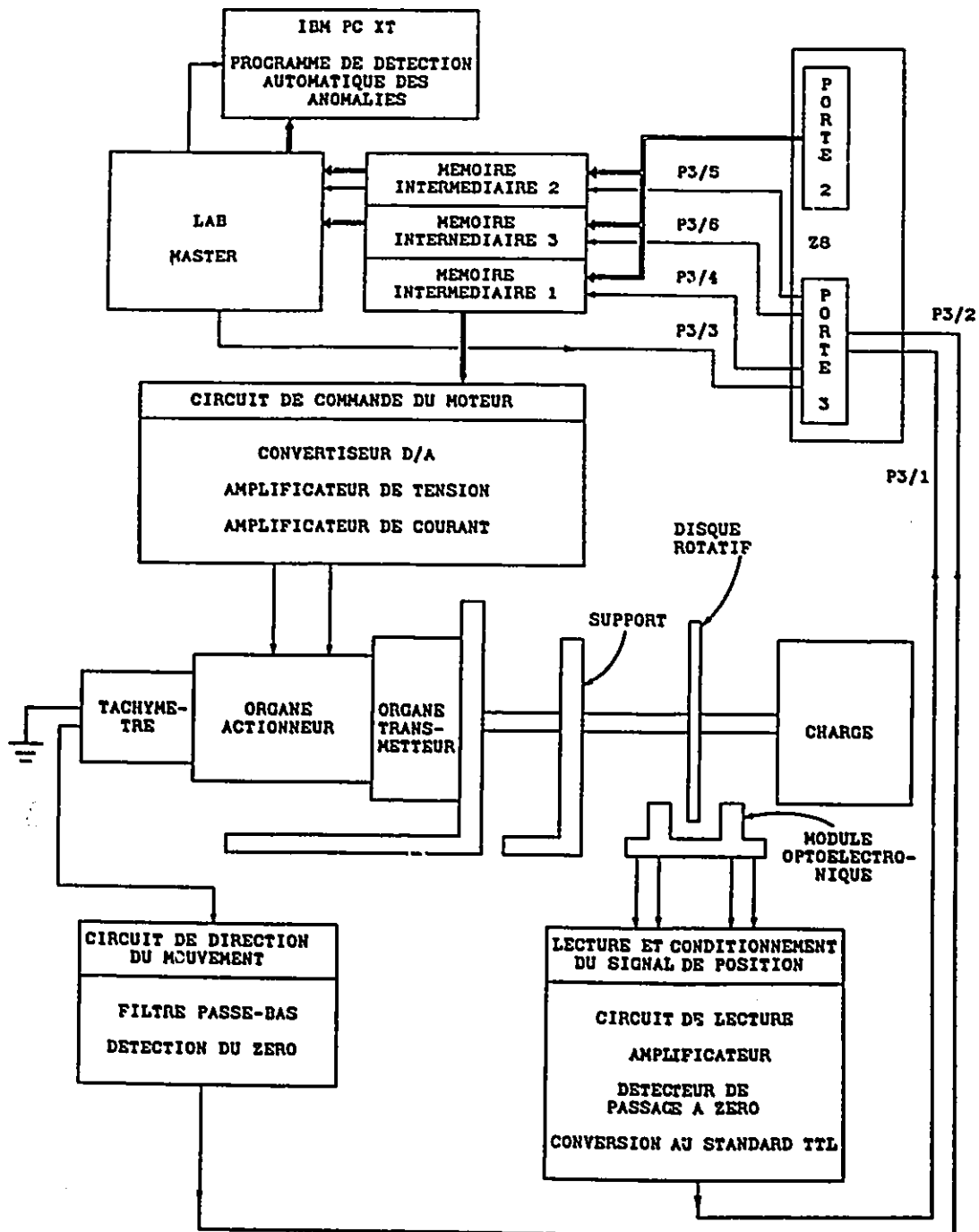


Figure A.1b: Le schéma du montage expérimental

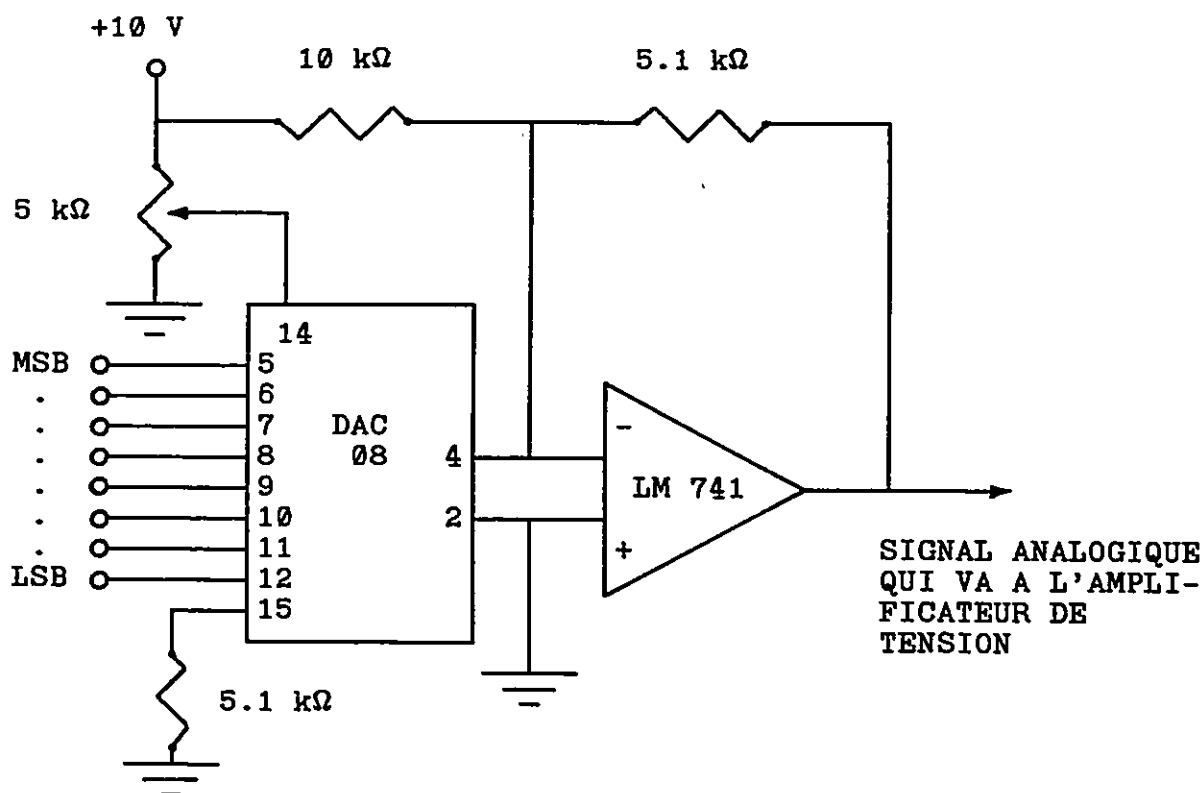


Figure A.2: Le circuit de conversion digital à analogique

2. L'amplificateur de tension.

A la sortie du circuit du L.A on a un voltage entre -5 et 5 volts. Un amplificateur operationnel avec un gain de 2.4 est utilisé pour avoir la tension de ± 12 volts nécessaire pour commander le moteur (figure A.3). Le facteur d'amplification est donné par la formule suivante,

$$V_{out} = V_{in} \cdot R_f / R_{in}$$

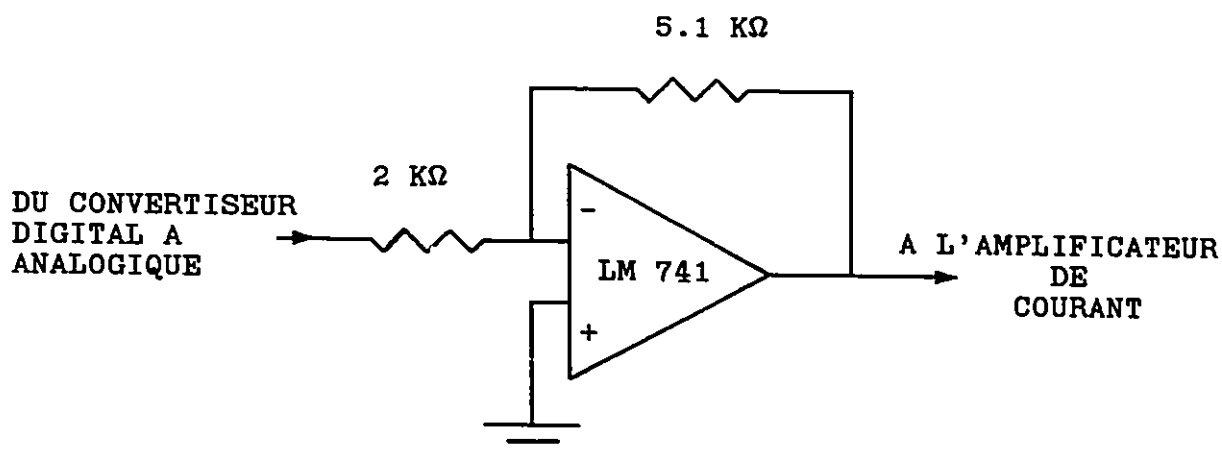


Figure A.3: L'amplificateur de tension.

3. L'amplification de courant.

A la sortie de l'amplificateur de tension on a ± 12 volts mais seulement quelques milliampères tandis que, lorsque calé, le moteur draine 4.75 amp. Afin de fournir le courant nécessaire, deux transistors Darlington sont utilisés, un NPN (2N6388) et son complément un PNP (2N6668) ayant une capacité de 60 volts, 10 ampères. Pour permettre l'opération en mode bipolaire deux diodes sont utilisées en parallèle avec les deux transistors (Figure A.4). De plus deux drains de chaleurs ont été installés sur les transistors afin de prévenir le surchauffement et le claquage thermique qui s'en suit.

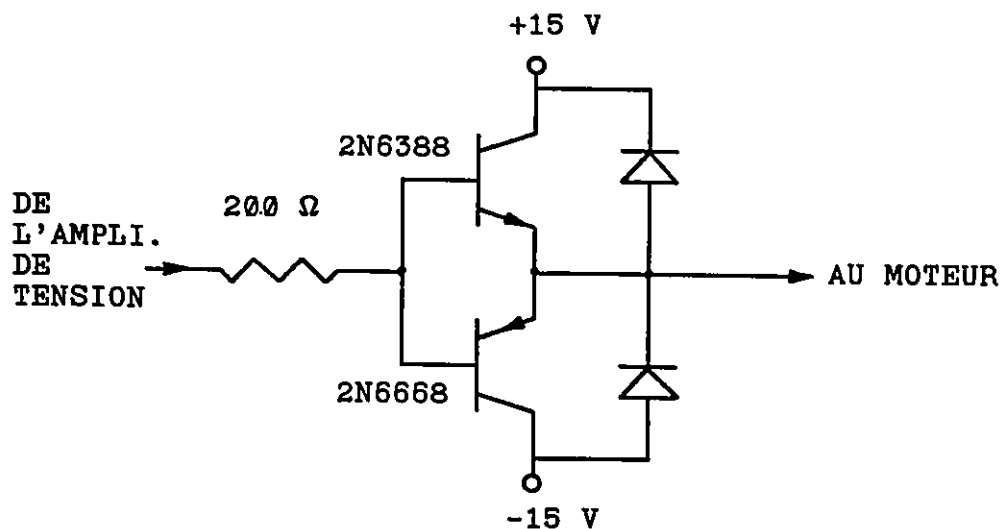


Figure A.4: L'amplificateur de courant.

C. LE MICROPROCESSEUR

Comme on peut le voir à la figure A.1, le microprocesseur est une pièce importante du système d'asservissement. Il reçoit toute les données de position à partir desquels il va décider de l'action à entreprendre.

Un micro-ordinateur à plaquette unique basé sur le microprocesseur Z8671 de Zilog est utilisé. Ce système possède un interpréteur BASIC/DEBUG de 2K ROM ("Read Only Memory"), 144 octets de RAM ("Random Access Memory"). Il comprend aussi 2 portes parallèles, et une porte série RS232C.

1. Les entrées et sorties.

La figure A.5 nous montre la structure des entrées et sorties.

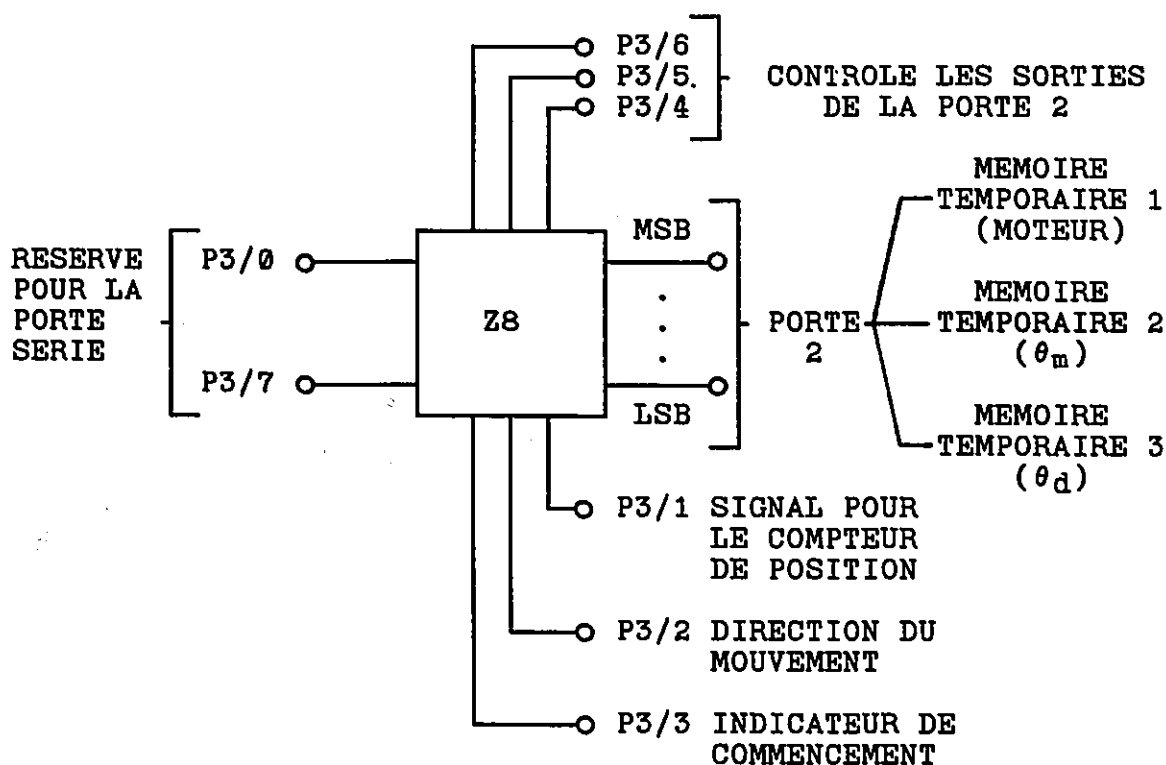


Figure A.5: Entrées et sorties du Z8.

La porte 2 est utilisée alternativement pour le signal de commande du moteur (au D/A) et pour communiquer la position présente et la position désirée à la mémoire temporaire.

Pour la porte 3, deux bits sont réservés pour la porte série, trois bits sont utilisés comme entrée et les trois autres comme sortie.

La position est lue à P3/1 (porte 3 bit 1) qui est relié à un compteur de 8 bits. C'est à dire qu'à chaque fois que le signal de position devient haut (quand le module optoélectronique détecte la présence d'un trou), le compteur est augmenté automatiquement. C'est un compteur de 8 bits parce qu'il peut compter jusqu'à 255 seulement, si on lui demande d'aller plus haut le compteur repart à zéro, le programme doit donc tenir compte de ce fait.

La direction du mouvement est lue à P3/2. La table A.2 nous montre la convention utilisée.

Table A.2: Direction du mouvement

DIRECTION DU MOUVEMENT	VOLTAGE A P3/2 (volts)
horaire	5.0
antihoraire	0.24

P3/3 est utilisé comme indicateur de commencement. En effet tant que P3/3 est nul le programme d'asservissement, dans le Z8 est en attente. C'est à partir du program Fortran de détection des anomalies, sur le IBM PC que P3/3 est posé haut, ce qui fera débiter le programme de commande.

P3/4, P3/5 et P3/6 sont utilisés, comme sorties, pour coordonner les sorties à la porte 2. La table A.3 nous montre la structure utilisée. La notation binaire est utilisée ici, i.e. un 1 équivaut à 5 volt, un zero à zero volt.

Dans le programme de commande, sur le Z8, les valeurs de P3/4, P3/5 et P3/6 sont variés en fonction de l'effet désiré sur la porte 2.

P3/0 et P3/7 sont réservés pour la porte série.

2. Le programme de commande du moteur.

Le programme de commande a été écrit en assembleur pour le Z8671. L'annexe C contient l'organigramme de même que le programme complet.

Table A.3: Coordination des sorties à la porte 2.

PORTE	VALEUR	EFFET
P3/4	0	La mémoire intermédiaire (1) garde la valeur précédente de vitesse pour le moteur.
	1	La mémoire enregistre une nouvelle valeur de vitesse pour le moteur.
P3/5	0	La mémoire intermédiaire (2) garde la valeur de position réelle du système.
	1	La mémoire enregistre une nouvelle valeur de position réelle.
P3/6	0	La mémoire intermédiaire (3) garde la valeur de position désirée.
	1	La mémoire enregistre une nouvelle valeur de position désirée.

D. SYSTEME DE LECTURE ET DE CONDITIONNEMENT DES DONNEES.

Comme on peut le voir à la figure A.1b on peut diviser la lecture et le conditionnement des données en deux sous-systèmes: la position et la direction du mouvement.

1. La position.

1) L'appareil de mesure: On utilise ici un module optoélectronique de type TIL138. Utilisé avec un masque rotatif de 50 trous on obtient une précision de ± 3.6 degrés. La figure A.6 montre l'appareillage utilisé.

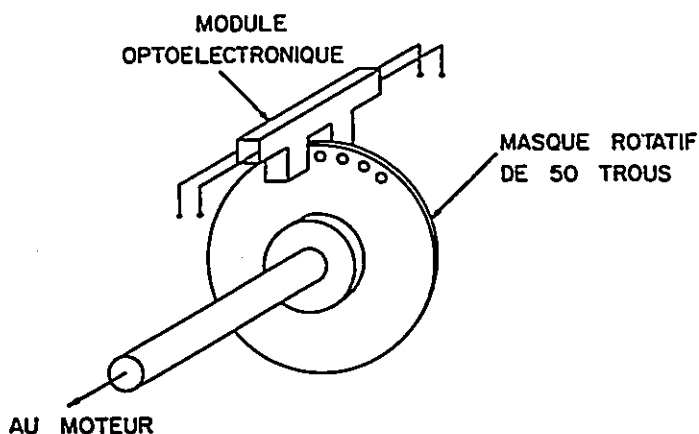


Figure A.6: Module optoelectronique.

ii) Le circuit de conditionnement: Du module optoélectronique un signal de type sinusoïdal d'une amplitude de 150 mVolt est obtenue (figure A.7). Afin d'utiliser ce signal on doit le rendre carré et compatible au standard TTL.

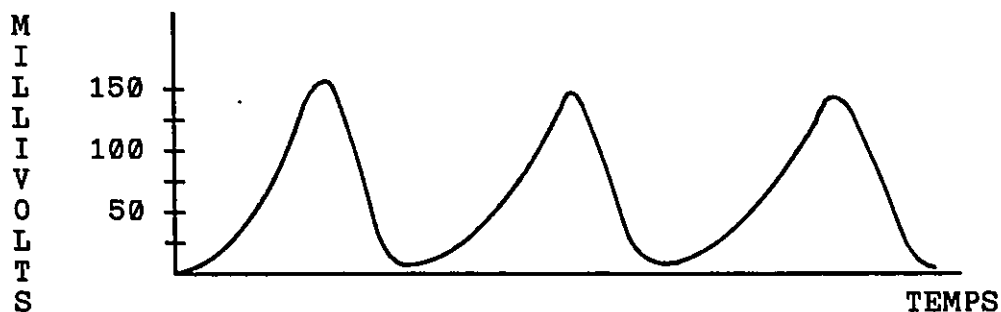


Figure A.7: Sortie du module optoélectronique.

La première phase du procédé de conditionnement est l'amplification du signal. Pour ce faire, on utilise un amplificateur opérationnel (LM741) avec un gain de 30.3 (figure A.8).

La deuxième phase consiste à rendre la signal carré. A cette fin on va utiliser un détecteur de passage à zéro (figure A.9).

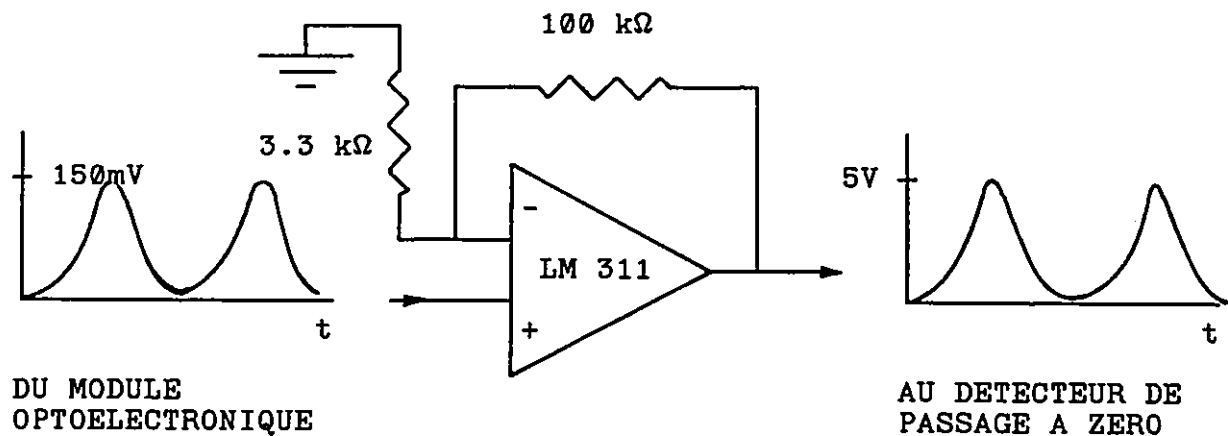


Figure A.8: Amplification de la sortie du module optoélectronique.

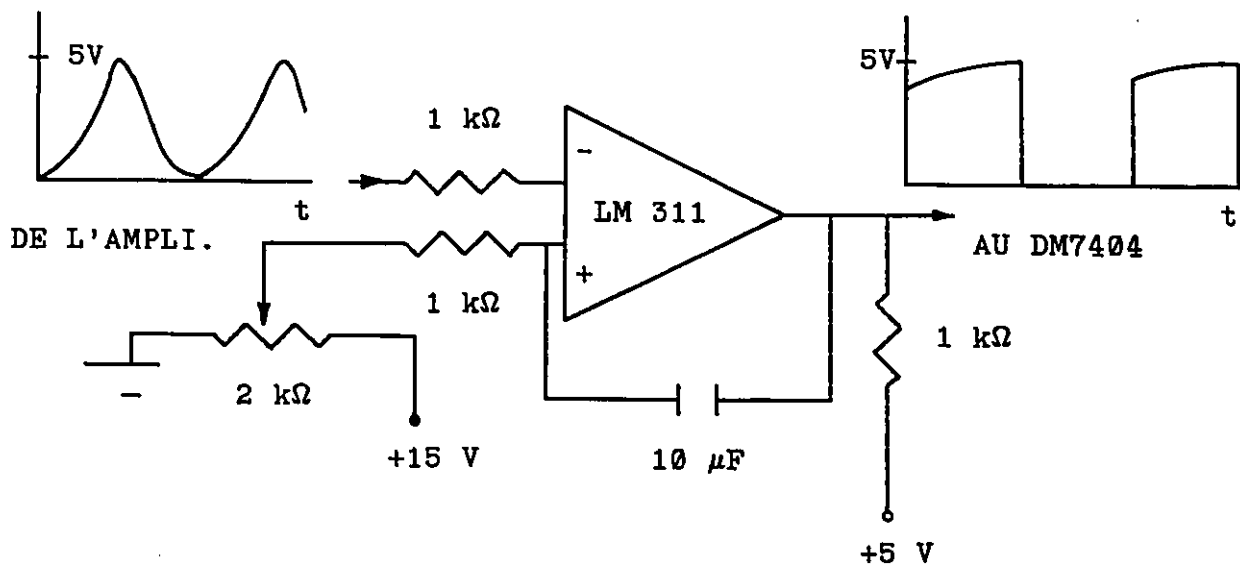


Figure A.9: Le détecteur de passage à zéro.

Le détecteur est constitué d'un comparateur (LM311) qui lorsque le signal à l'entrée est nulle donne une sortie basse, si le signal devient plus grand qu'une valeur seuil alors la sortie devient haute et restera haute (environ 5 volts) tant que le signal sera plus grand cette valeur seuil (figure A10).

Le potentiomètre, de la figure A.9, est utilisé pour fixer la valeur seuil. Plus le seuil est élevé plus l'onde carrée sera étroite.

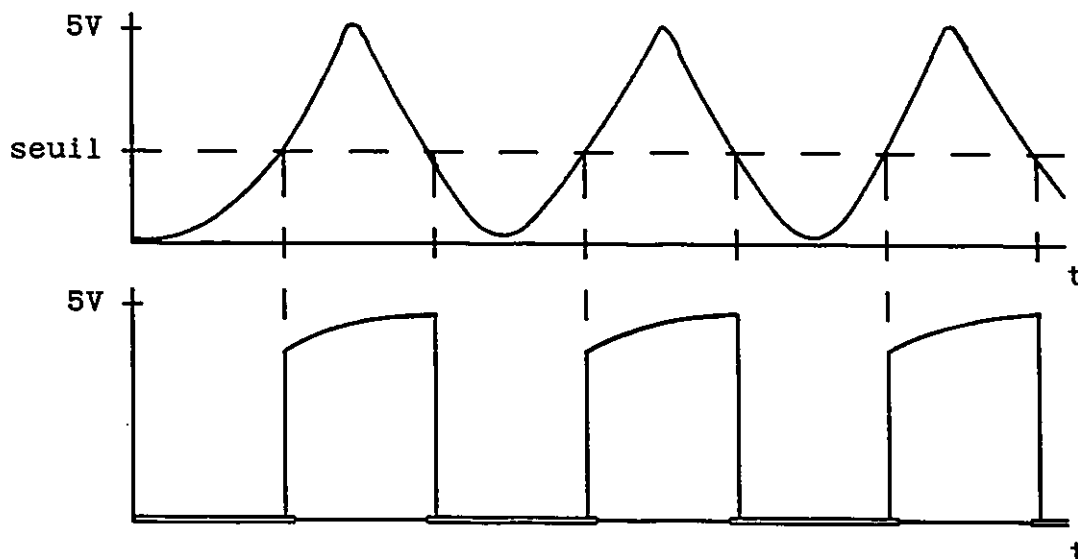


Figure A.10: La formation des ondes carrées.

La dernière phase du conditionnement consiste à rendre le signal compatible au standard TTL. Comme on peut le voir sur la figure A.9 le signal n'est pas tout à fait carré, la partie supérieure du signal étant quelque peu déformée. Un inverseur hexadécimal de type DM7404 est utilisé comme le montre la figure A.11.

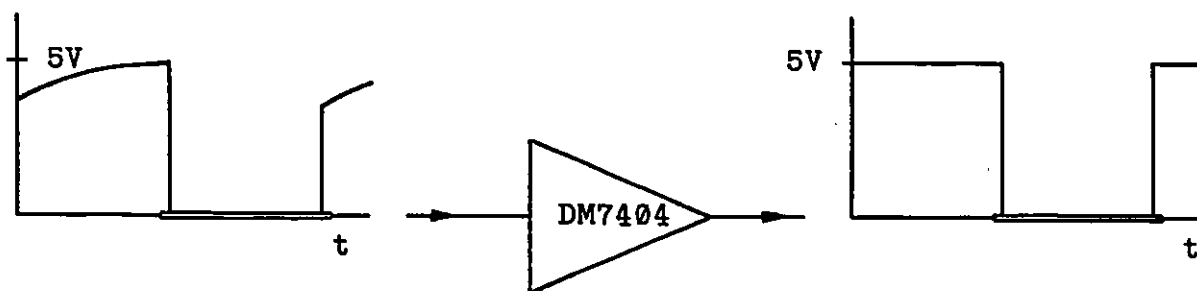


Figure A.11: Conversion au standard TTL.

La figure A.12 nous présente le circuit de lecture et conditionnement du signal de position complet.

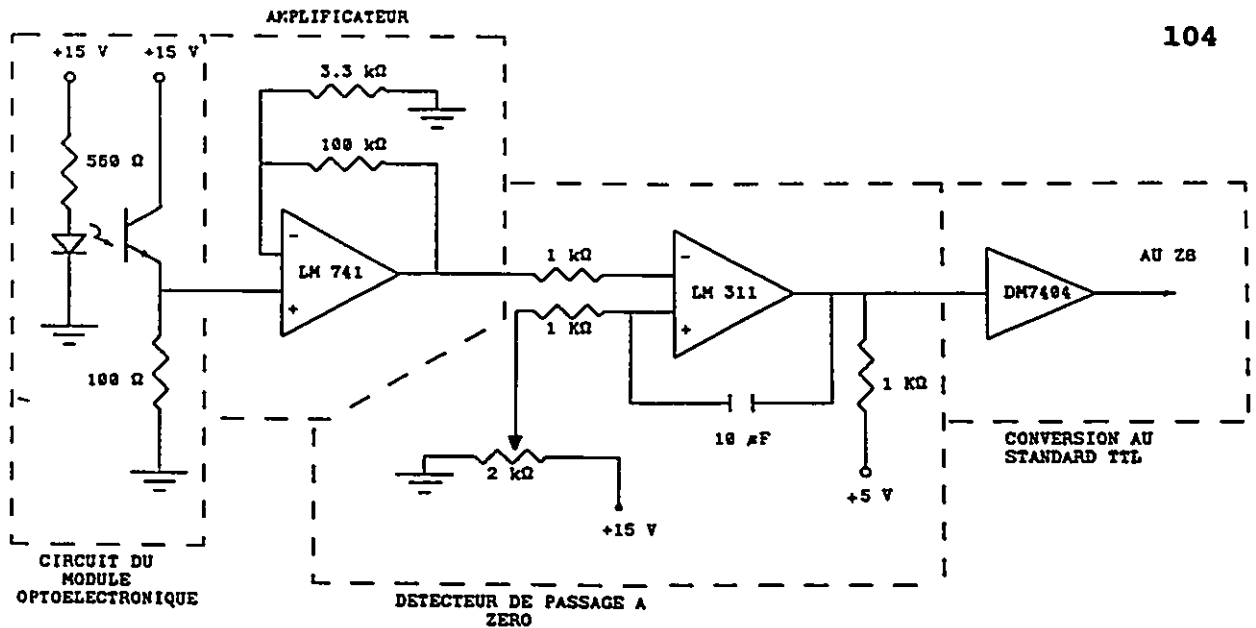


Figure A.12: Lecture et conditionnement du signal de position.

2. La direction du mouvement.

Un circuit de mesure de la direction du mouvement est nécessaire afin d'indiquer au Z8 s'il doit augmenter ou diminuer le compteur de position lorsque celui reçoit le signal. La figure A.13 nous montre le circuit utilisé.

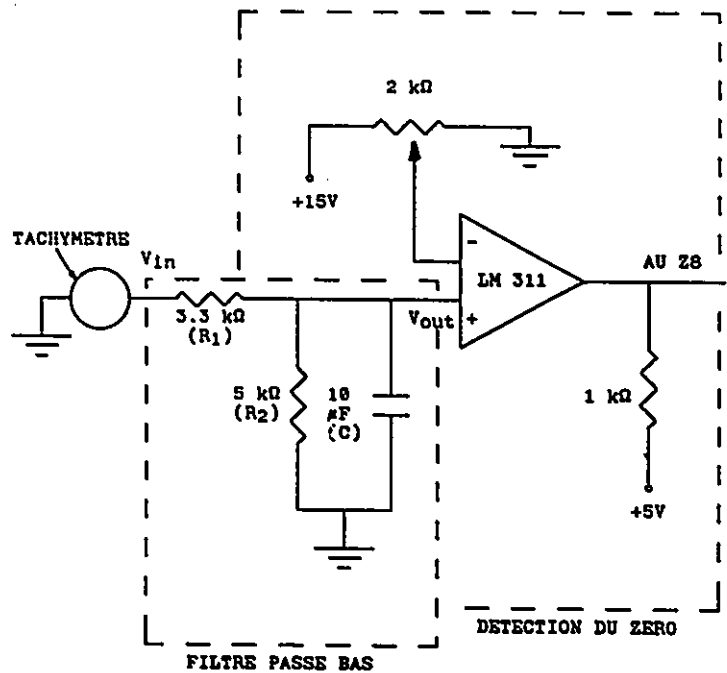


Figure A.13: Circuit de direction du mouvement.

Le circuit se divise en deux parties; le filtrage et la détection du passage à zéro. Pour la première partie, à la sortie du tachymètre on a un signal de ± 10 volts avec une ondulation de 1 volt (figure A.14).

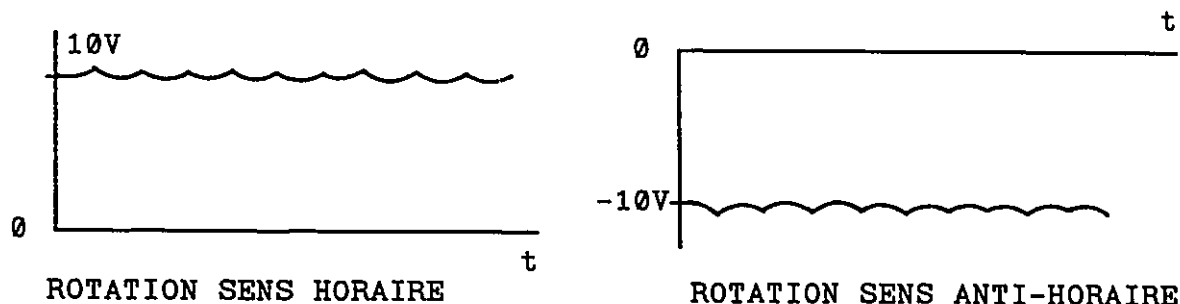


Figure A.14: Sortie du tachymètre.

On doit donc utiliser un filtre passe bas afin d'éliminer les ondulations. Un filtre passif très simple est utilisé et nous donne une sortie très clair. Il a la fonction de transfert suivante:

$$\frac{V_{out}}{V_{in}} = \frac{Z_C R_2}{(Z_C R_2 + R_1 R_2 + R_1 Z_C)}$$

ou

$$Z_C = 1/j\omega C$$

En grandeur on aura,

$$\left| \frac{V_{out}}{V_{in}} \right| = \left[\frac{1.722 \times 10^{15} + 6.808 \times 10^{11} \omega^2}{4.746 \times 10^{15} + 3.751 \times 10^{12} \omega^2 + 7.412 \times 10^8 \omega^4} \right]^{1/2}$$

Ce qui nous donne une fréquence de coupure d'environ 3Hz et une atténuation de 0.604.

Pour la deuxième partie du circuit un comparateur de voltage (LM311) est utilisé. Lorsque que le signal d'entrée dépasse une valeur seuil fixée par le potentiomètre P1 (0.64 volt) alors la sortie est haute sinon elle reste basse.

La valeur seuil ne doit pas être trop grande afin de détecter la direction du mouvement même à basse vitesse. De plus, comme tout comparateur, il nécessite une résistance extérieure ($R_3 = 1k\Omega$) comme charge.

La table A.2, présentée dans une section précédente, nous montre la convention utilisée pour le signal de direction du mouvement, une fois le conditionnement terminé.

E. ACQUISITION DES DONNEES.

Le programme de détection automatique des anomalies, qui est un programme Fortran fonctionnant sur IBM PC XT, requiert comme entrée la position désirée et la position mesurée par le système. Ces données sont lues par le programme Fortran en même temps que le système fonctionne. Comme mentionné précédemment une seule porte de sortie sur le Z8 est disponible pour commander le moteur et fournir les positions. On doit donc utiliser une mémoire intermédiaire (tampon).

De plus, une procédure est nécessaire afin de prendre les données en mémoire intermédiaire et de les transférer dans la mémoire du IBM PC XT, ce qui est fait par le LAB MASTER. La structure du système d'acquisition des données est montrée à la figure A.15.

Pour chaque sortie, une paire de mémoire tampon de type "DM7475 QUAD LATCH" est utilisée. Les paires de tampons sont contrôlées par les bits P3/4, P3/5 et P3/6 de la porte 3 comme indiqué à la table A3.

Une fois les valeurs en mémoire intermédiaire (ou tampon) le contrôle est passé au programme Fortran qui utilise le bit à P3/5 comme indicateur de lecture. Chaque fois que cet bit devient haut il est temps pour le programme Fortran de lire les valeurs à travers le LAB MASTER. La lecture se fait à tout les 0.04 seconde, c'est-à-dire à tout les dix fois la période d'échantillonnage du programme de commande à l'intérieur du Z8. Pour des valeurs plus petites que 0.04, le programme Fortran n'avait pas le temps de lire adéquatement tout les paramètres demandés

Le LAB MASTER est utilisé uniquement pour le transfert des données entre la mémoire tampon et le IBM. Celui-ci possède aussi des convertisseurs analogique à digitale ou vice versa, 5 compteurs indépendants et 24 lignes parallèles d'entrée et sortie programmable.

F. LE SYSTEME D'ALIMENTATION.

Trois sources indépendantes de tension sont nécessaire pour opérer le système. La première, une source de tension fixe de ± 12 volts et +5 volts fournie le puissance requise par le Z8 et la plaquette de circuit des mémoires tampons.

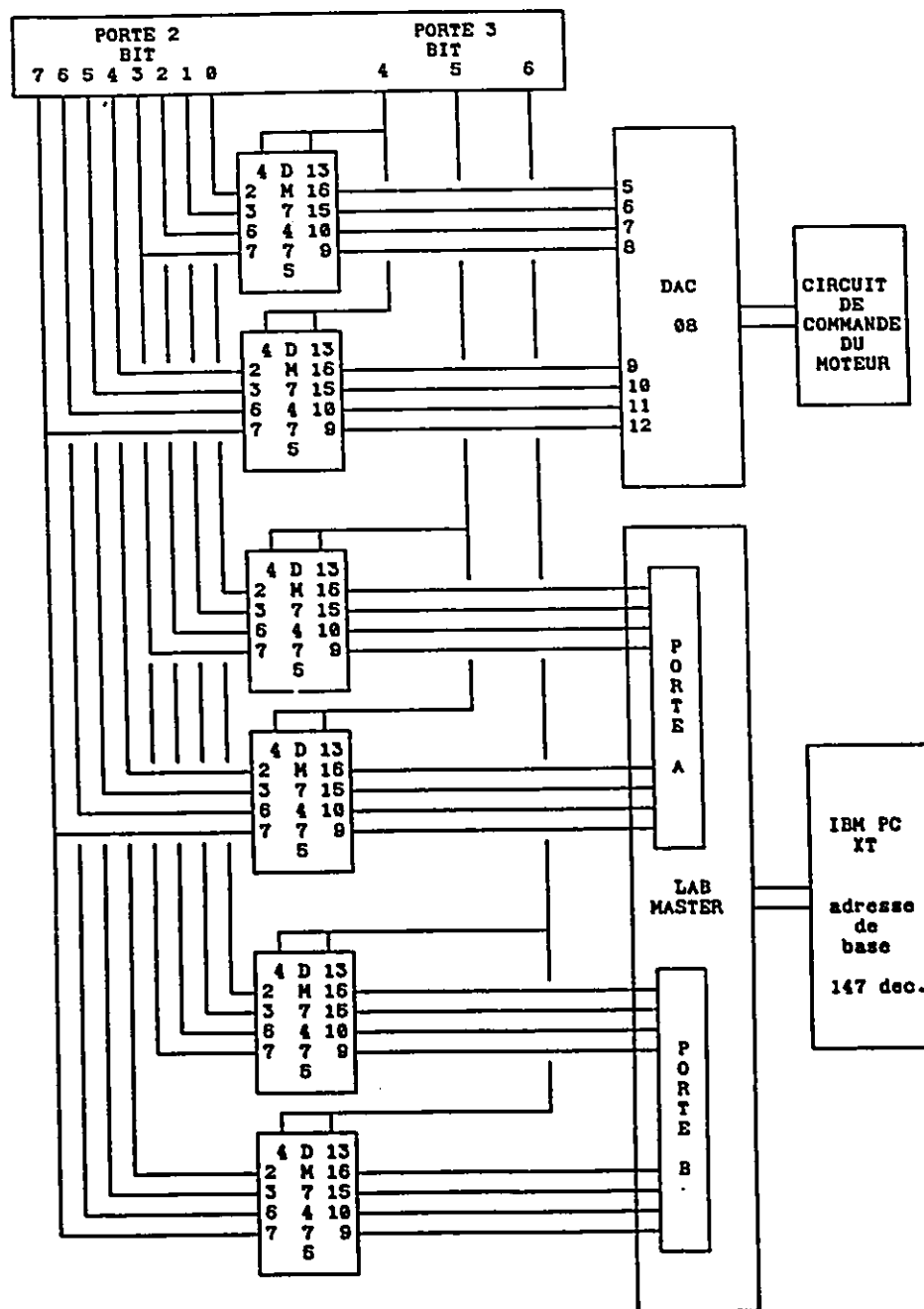
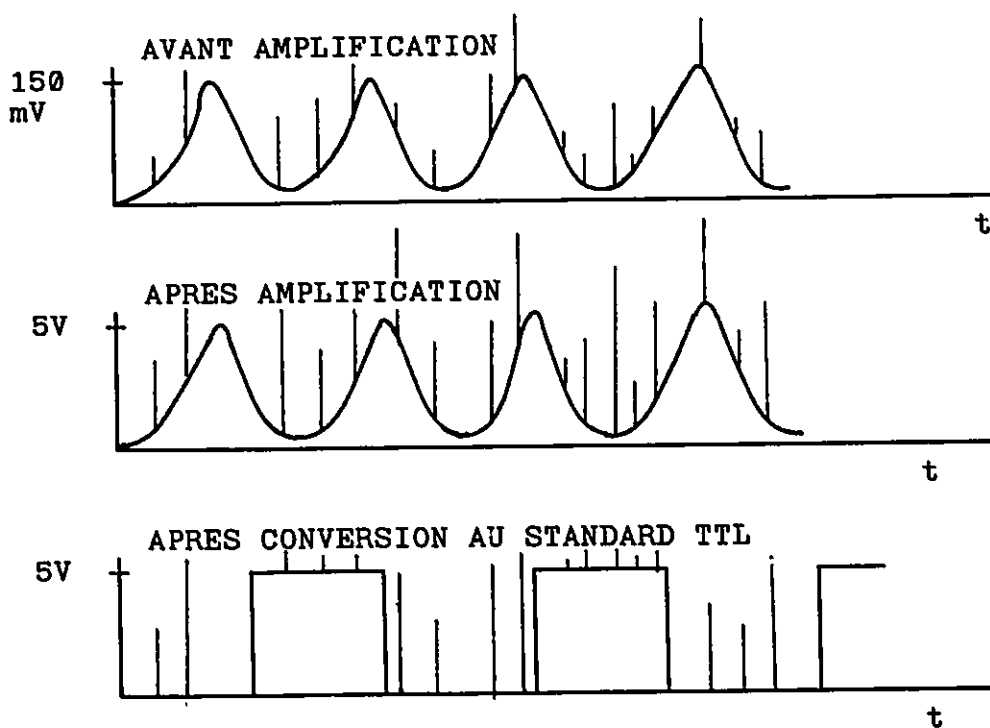


Figure A.15: Structure du système d'acquisition des données.

La deuxième source de tension fournit la puissance nécessaire au moteur et au circuit de commande du moteur. La tension est fixée à ± 15 volts et une prise de terre.

La troisième source de tension fournit les circuits de lecture et de conditionnement des données. La tension nécessaire est de ± 15 volts, 5 volts et une prise de terre. Les mêmes tensions que pour le moteur sont nécessaires mais la même source ne pouvait être utilisée. En effet lors de l'utilisation d'une source conjointe de tension il a été observé que lorsque le moteur fonctionnait il introduisait un faible bruit sur la sortie du module optoélectronique (figure A16). Ce bruit, bien que insignifiant à ce stade, était par la suite amplifié avec le signal de sortie. L'effet sur le signal de position envoyé au Z8 est démontré à la figure A.16. Le bruit était presque invisible sur un oscilloscope due à la grandeur des impulsions. Afin d'éliminer ce bruit deux sources de tensions séparées ont dû être utilisées.



LE COMPTEUR DE POSITION AUGMENTE CHAQUE QUE CE DERNIER SIGNAL DEVIENT HAUT (+5V)

Figure A.16: Effet de l'utilisation d'une seule source de tension sur la position obtenue.

ANNEXE B
CONCEPTION
DU
REGULATEUR PID

CONCEPTION DU REGULATEUR PID.

Les systèmes asservis échantillonnés sont de plus en plus utilisés aujourd'hui. Ceci est principalement due à la grande flexibilité que l'ordinateur apporte dans le processus d'asservissement. Il a donc été choisi d'utiliser un système asservis échantillonné à action PID (Proportionnelle, Intégrale et Différentielle), Le système à action PID étant un des plus utilisé en système asservis continus. La partie intégrale du système a pour effet de réduire l'erreur stationnaire tandis que la partie différentielle apporte un élément d'anticipation qui réduit le dépassement transitoire.

Plusieurs méthode peuvent être utilisées pour analyser et concevoir des systèmes échantillonnés: la méthode des transformées en "z" (qui est l'équivalent discret de la méthode utilisée pour les systèmes continus) et la méthode des équations d'états. De plus, il est aussi possible de faire l'analyse comme pour un système continu et de tenir compte des effets de l'échantillonnage séparément. En fait, cette dernière procédure est sûrement la plus utilisée, encore aujourd'hui même avec le développement des méthodes dans le domaine "z". Cette procédure sera d'ailleurs utilisée pour la conception du système asservis qui nous concerne.

La littérature est très abondante sur l'analyse et la conception de système de commande continus. La présente analyse se base principalement sur la référence [1] et partiellement sur les références [2], [3]. La méthode du lieux des pôles, tel que décrite dans Dorf [1], a donc été choisie car elle nous permet de visualiser les effets des variations des paramètres de commande (K_p , K_i et K_d) sur la stabilité relative du système. La conception se divise en trois partie: A- La modélisation du système; B- La recherche des paramètres du régulateur PID; C- L'étude de la stabilité relative du système.

A.LA MODELISATION DU SYSTEME

Le système que nous désirons modeler peut être représenté par le schéma de la figure B.1. Une entrée désirée est comparée avec la sortie lue par le module optoélectronique et accumulée par le compteur. Une erreur E^* est envoyée au régulateur à action PID, qui décidera de l'action à prendre et produira un signal de commande échantillonné, U^* . Comme un régulateur échantillonné et un moteur à courant continu sont utilisés, U^* doit être converti en signal analogique avant d'être envoyé à l'amplificateur. Le courant et la tension sont amplifiés et transmis à l'organe actionneur qui effectuera une action conséquente. Pour les détails sur les circuits et appareils utilisés voir annexe A.

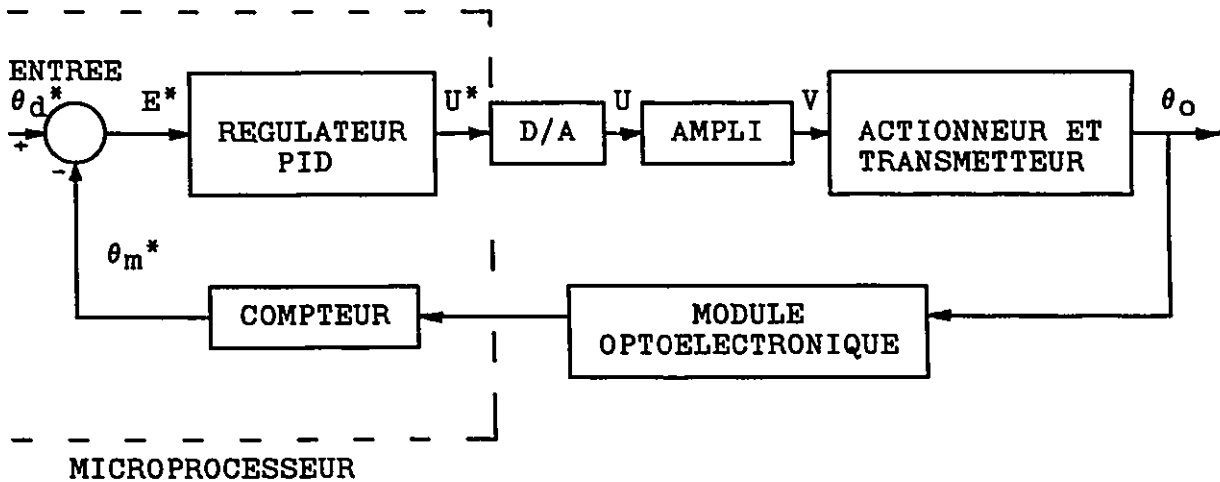


Figure B.1: Le système à modeler.

1) Le schéma fonctionnel correspondant au système.

Toute la modélisation et conception du système se fait dans le domaine de Laplace. La variable "s" est utilisée comme opérateur de Laplace. La figure B.2 nous montre le schéma fonctionnel du système pour une charge constante. Comme pour la figure B.1 toutes les variables sont exprimées dans le domaine Laplace

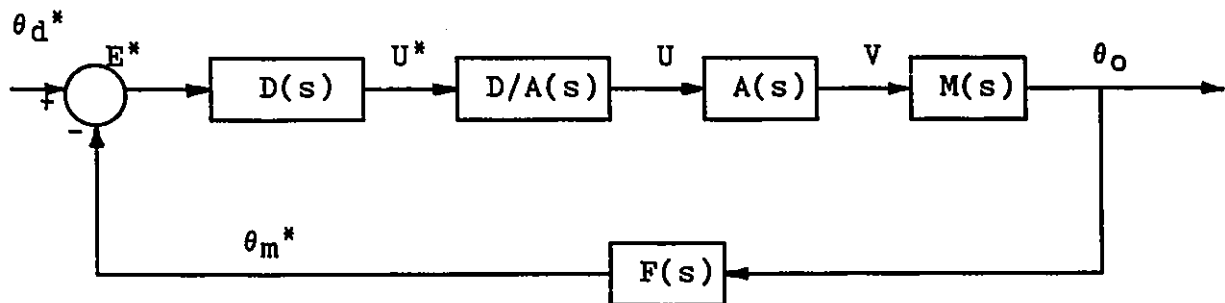


Figure B.2: Schéma fonctionnel du système.

i) La transmittance du régulateur PID, $D(s)$.

Le régulateur de type PID peut être représenté par le schéma fonctionnel suivant dans le domaine Laplace:

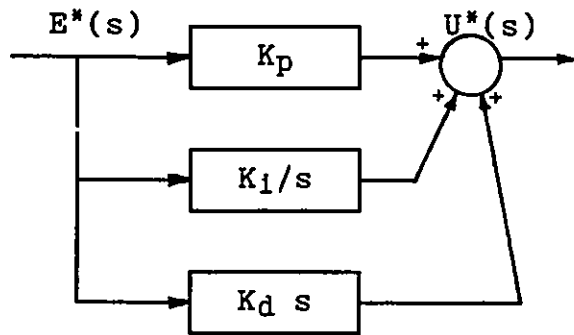


Figure B.3: Schéma fonctionnel du régulateur PID.

L'équation de la transmittance s'écrit,

$$D(s) = U(s)/ E(s) = K_p + K_i/s + K_d s \quad (B1)$$

ii) La transmittance du convertisseur digital à analogique, D/A(s)

En général

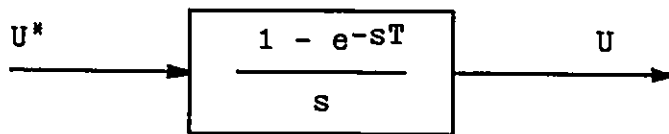


Figure B.4: La transmittance du D/A.

ou

$T =$ la période d'échantillonnage.

donc

$$D/A(s) = \frac{1 - e^{-sT}}{s} \quad (B2)$$

Mais si la période d'échantillonnage est relativement petite le convertisseur peut être considéré comme un délai pure ([2] page 455) c'est-à-dire que,

$$D/A(s) = 1$$

et ce délai est équivalent à $T/2$ dans le domaine temps.

iii) La transmittance de l'amplificateur, $A(s)$.

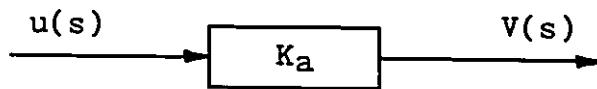


Figure B.5: Transmittance de l'amplificateur.

D'où

$$A(s) = V(s)/u(s) = K_a \quad (\text{B3})$$

et $K_a = 2,4$ (voir page 86, section B.2)

iv) La transmittance du moteur, $M(s)$.

La fonction de transfert d'un moteur CC accompagné d'un système d'engrenage (réducteur) est bien connu et il n'est donc pas nécessaire de refaire l'analyse au complet. On utilisera donc l'équation dérivée dans le livre de Ogata [4] pour une charge constante, B_0 et J_0 .

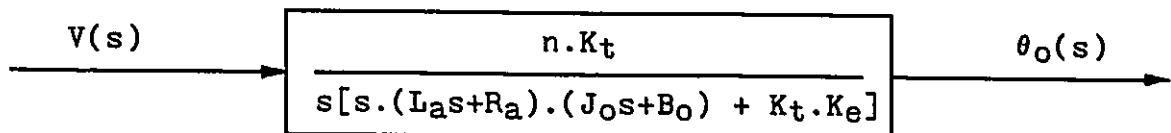


Figure B.6: Transmittance du moteur.

ou

- $\theta_o(s)$ = position angulaire de l'arbre à la sortie du réducteur.
- $V(s)$ = erreur sur la position désirée amplifiée.
- K_t = coefficient de couple moteur.
(0,0194 Nm/Amp.)
- K_e = coefficient de force contre électromotrice.
(0,0194 Nm/Amp)
- L_a = auto inductance de l'induit.
(1,49 mH)
- R_a = résistance de l'induit.
(2,04 Ω)
- J_0 = inertie du moteur et de la charge ramenée à l'arbre du moteur (2,753 x 10⁻⁶ Nm).
- B_0 = coefficient de frottement visqueux (on néglige le frottement pour la charge) (0,000184 Nm/(rad/sec)).

n = coefficient de réduction de l'engrenage.
(1/127)

Note: Toutes les valeurs viennent de la fiche technique du moteur sauf pour l'inertie de la charge.

Sous forme d'équation nous aurons;

$$M(s) = \frac{\theta_o(s)}{V(s)} = \frac{n \cdot K_t}{s[s \cdot (L_a s + R_a) \cdot (J_o s + B_o) + K_t \cdot K_e]} \quad (B4)$$

On peut substituer les valeurs pour obtenir l'équation finale, dans laquelle on néglige le terme en s^3 , qui en fait est très petit ($L_a s \rightarrow 0$).

$$M(s) = \frac{25.94}{s [s + 127.6]} \quad (B5)$$

v) La transmittance de la boucle de retour, $F(s)$.

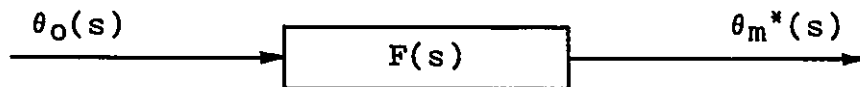


Figure B.7: La transmittance de la boucle de retour.

ou $F(S)$ peut être subdivisé comme suit:

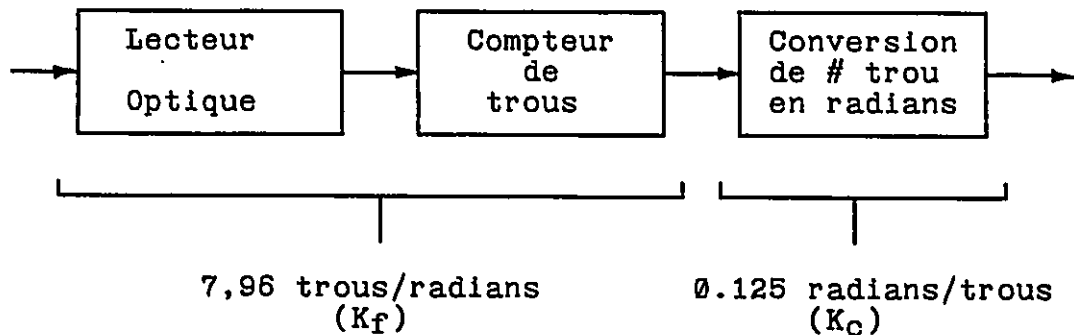


Figure B.8: Les subdivisions de $F(s)$.

Ce qui revient à dire que $F(s)$ est égale à 1.

2. Le graphe de fluence.

Afin d'écrire la fonction de transfert du système de la figure B.2, la méthode du graphe de fluence est utilisée tel que d'écrite par Mason dans [11]. Notez que l'indice "(s)" sera omis à partir de maintenant afin de simplifier la présentation.

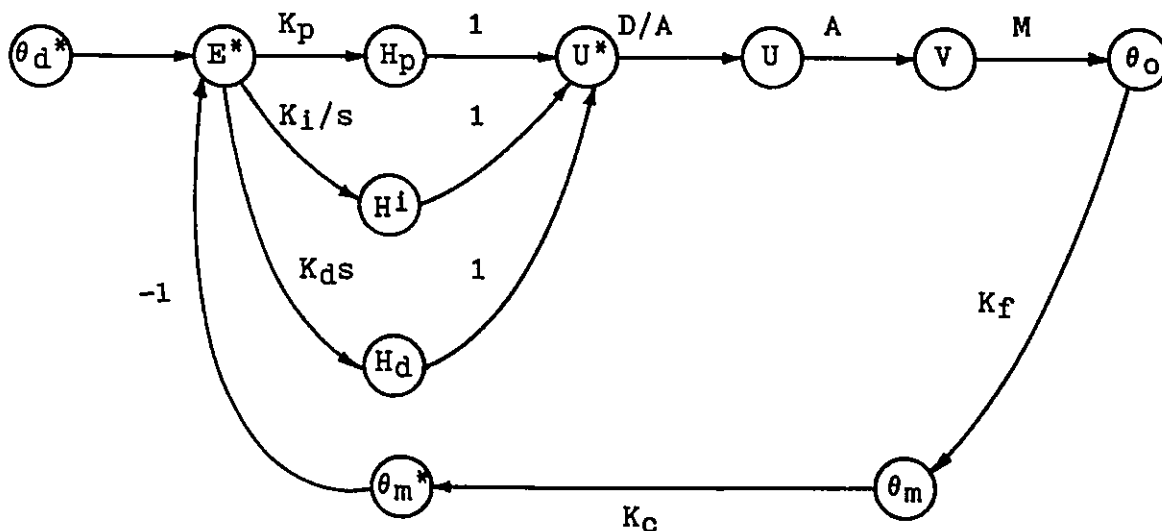


Figure B.9: Graphe de fluence.

3. La fonction de transfert du système.

En appliquant la règle de Mason au graphe de fluence de la figure B.9, on obtiendra la fonction de transfert.

i) les boucles:

$$\text{boucle 1: } E^* - H_p - U^* - U - V - \theta_o - \theta_m - \theta_m^* - E^*$$

$$\text{boucle 2: } E^* - H_i - U^* - U - V - \theta_o - \theta_m - \theta_m^* - E^*$$

$$\text{boucle 3: } E^* - H_d - U^* - U - V - \theta_o - \theta_m - \theta_m^* - E^*$$

ii) La transmittance, P_i , pour la boucle i:

$$P_1 = K_p \cdot D/A \cdot A \cdot M \cdot K_f \cdot K_c \cdot (-1) \quad (B6)$$

$$P_2 = K_i/s \cdot D/A \cdot A \cdot M \cdot K_f \cdot K_c \cdot (-1) \quad (B7)$$

$$P_3 = K_d s \cdot D/A \cdot A \cdot M \cdot K_f \cdot K_c \cdot (-1) \quad (B8)$$

iii) Le déterminant, D.

En général,

$$D = 1 - \sum P_i + \sum P_k P_j - \sum P_m P_n P_p + \dots$$

ou

$\sum P_i$ = somme des transmittances P_i de toutes les boucles.

$\sum P_k P_j$ = somme 2 à 2 des transmittances des boucles n'ayant ni noeud ni branche en commun.

$\sum P_m P_n P_p$ = somme 3 à 3 des transmittances des boucles n'ayant ni noeud ni branche en commun.

Dans la cas présent, les trois boucles ne sont pas indépendantes car elles partagent au moins un noeud ou une branche. Le déterminant sera donc,

$$D = 1 - P_1 - P_2 - P_3 \quad (B9)$$

En substituant les équations B6, B7 et B8 dans B9 on aura,

$$D = 1 + D/A.A.M. (K_p + K_i/s + K_{ds}) \quad (B10)$$

iv) Transmittances des chemins directs.

On a trois chemins directes;

$$C_1 = K_p \cdot D/A.A.M$$

$$C_2 = K_i/s \cdot D/A.A.M$$

$$C_3 = K_{ds} \cdot D/A.A.M$$

v) La règle de Mason.

$$\frac{\theta_o}{\theta_d} = \frac{\sum_{i=1}^3 C_i (1 - \sum_1 P_{i1} + \sum_{2 \text{ à } 2}^{\pi} P_{i1} - \sum_{3 \text{ à } 3}^{\pi} P_{i1} + \dots)}{D}$$

P_{il} = transmittances de p boucles ($l = 1$ à p) n'ayant ni noeud ni branche commun avec le chemin C_l .

$\sum \pi P_{il}$ = somme des produits deux à deux des boucles n'ayant ni branche ni noeud commun.

Pour notre cas, toutes les boucles touchent au chemin direct, alors P_{il} est toujours nul. La fonction de transfert pour une charge constante sera donc,

$$\frac{\theta_o}{\theta_d} = \frac{D/A.A.M.(K_p + K_i/s + K_d)}{1 + D/A.A.M.(K_p + K_i/s + K_d)} \quad (B11)$$

Substituons les valeurs connues.

$$\frac{\theta_o}{\theta_d} = \frac{62,25 (K_p + K_i/s + K_d s)}{s (s + 127,6) + 62,25 (K_p + K_i/s + K_d s)} \quad (B12)$$

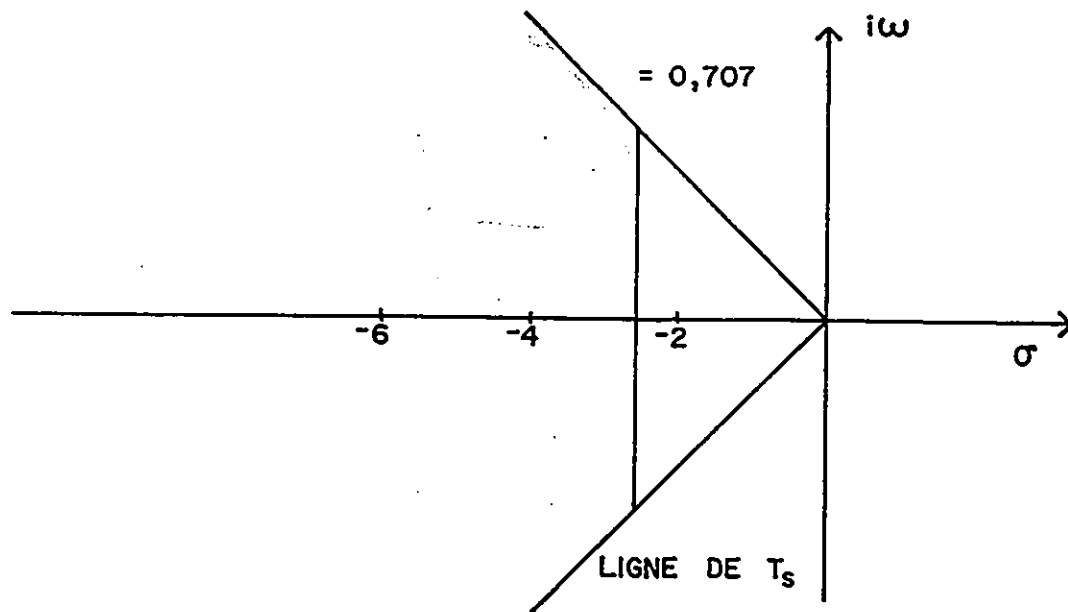
Les paramètres du régulateur PID seront sélectionnés afin de produire une réponse satisfaisant des critères de conception spécifiques.

B. LA RECHERCHE DES PARAMETRES DU REGULATEUR PID

Comme mentionné précédemment, la méthode du lieu des pôles (Root Locus [1]) sera utilisée pour la conception du régulateur. Les critères de conception sont les suivants:

- 1 - Dépassement transitoire $\leq 5\%$
- 2 - Temps pour atteindre le régime permanent avec une erreur de $2\% \leq 1,5$ sec.
- 3 - Facteur d'amortissement $\geq 0,707$

Le critère du facteur d'amortissement requière donc que les racines du système soit sous la ligne du 45° dans le plan gauche du lieu des pôles. De plus le critère de temps pour atteindre le régime permanent avec une erreur de 2% demande que les racines soit à gauche de la ligne $\sigma = -2.6$. Donc, pour satisfaire les critères de conception, toutes les racines doivent être situées à l'intérieure de la section ombragée sur la figure B.10.



T_s = TEMPS POUR ATTEINDRE LE REGIME PERMANENT

Figure B.10: Région où les poles doivent être situés.

i) L'équation caractéristique.

Le dénominateur de la fonction de transfert (B12) nous donne l'équation caractéristique du système à partir de laquelle la conception sera faite. Posons donc celle-ci égale à zéro.

$$s^3 + (127,6 + 62,25 K_d) s^2 + 62,25 K_p s + 62,25 K_i = 0$$

Afin de simplifier l'analyse, des paramètres intermédiaires sont définis.

$$\alpha_1 = 127,6 + 62,25 K_d \quad (B13)$$

$$\alpha_2 = 62,25 K_p \quad (B14)$$

$$\alpha_3 = 62,25 K_i \quad (B15)$$

On peut réécrire l'équation caractéristique,

$$s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3 = 0 \quad (\text{B17})$$

Avec les équations B13 à B17 on pourra déterminer les paramètres du système. On déterminera d'abord chaque paramètre en étudiant l'effet de varier le dit paramètre sur le lieu des pôles. Chaque paramètres seront considérés séparément. Par la suite l'effet de la variation simultanée des paramètres sera étudié.

ii) L'effet de varier α_1 .

Considérons l'effet de varier α_1 sur le système quand α_2 et α_3 sont égaux à zéro. B17 devient alors,

$$s^3 + \alpha_1 s^2 = 0$$

qu'on peut aussi exprimer comme suit;

$$1 + \frac{\alpha_1}{s} = 0 \quad (\text{B18})$$

La figure B.11 nous montre le lieu des pôles quand α_1 varie. Comme nous désirons avoir un K_d positif, la valeur de α_1 doit être plus grande que 127,6 (voir eq. B13). La valeur de K_d a été choisie pour $\alpha_1 = 150$, ce qui nous donne $K_d = 0,3598$.

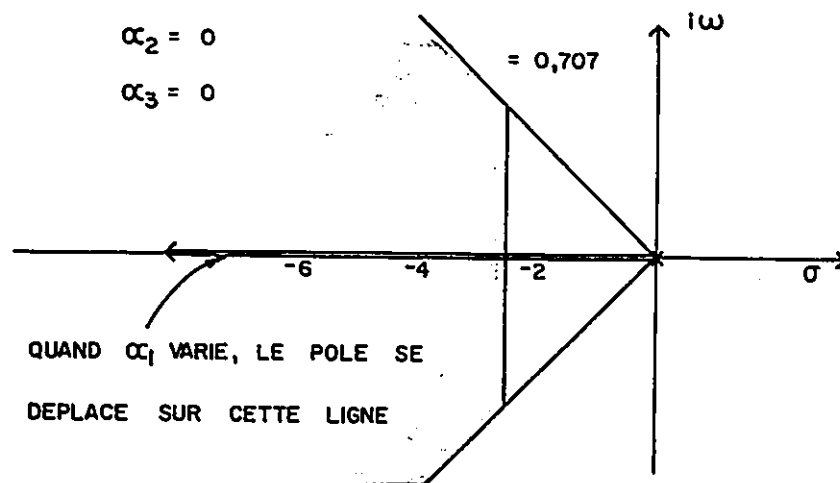


Figure B.11: Effet de varier α_1 .

iii) L'effet de varier α_2 .

Considérons l'effet de varier α_2 sur la système quand $\alpha_1 = 150$ et $\alpha_3 = 0$. B17 devient alors,

$$s^3 + 150 s^2 + \alpha_2 s = 0$$

qu'on peut aussi exprimer comme;

$$1 + \frac{\alpha_2}{s^2 + 150 s} = 0 \quad (\text{B19})$$

La figure B.12 nous montre le lieu des pôles quand α_2 varie. On doit donc choisir la valeur de α_2 de sorte que les pôles de l'équation B19 soient à l'intérieur de la région choisie. La valeur de 2000 pour α_2 à été choisie, ce qui nous donne K_p égal à 32,13. Pour cette valeur les pôles sont représentés en rouge sur la figure B.12

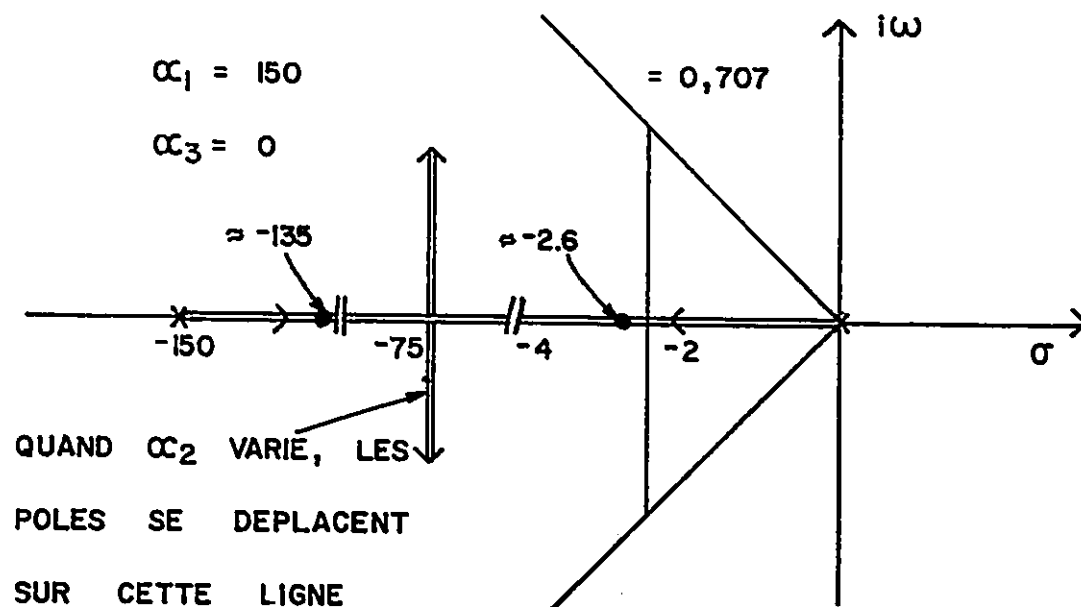


Figure B.12: Effet de varier α_2 .

iv) L'effet de varier α_3 .

Considérons l'effet de varier α_3 sur le système lorsque $\alpha_1 = 150$ et $\alpha_2 = 2000$. B17 devient alors;

$$s^3 + 150 s^2 + 2000 s + \alpha_3 = 0 \quad (\text{B20})$$

qu'on peut aussi écrire comme suit;

$$1 + \frac{\alpha_3}{s^3 + 150 s^2 + 2000 s} = 0 \quad (\text{B21})$$

La figure B.13 nous montre le lieu des pôles lorsqu'on varie α_3 . On cherche un α_3 de sorte que les racines de B20 soient dans la région ombragée. Pour respecter le critère de dépassement transitoire α_3 doit être égal à 2000, ce qui fixe K_i à 32,13.

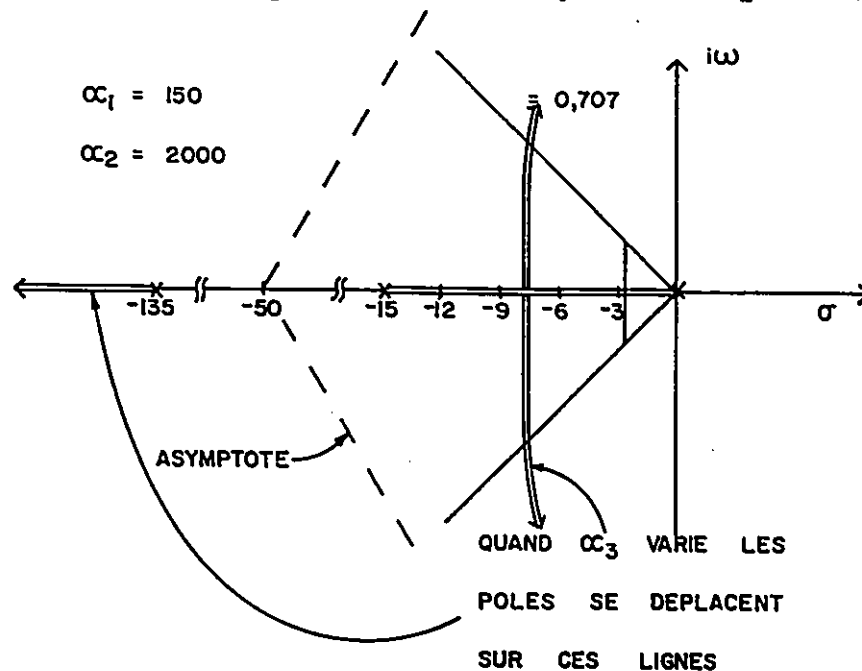


Figure B.13: Effet de varier α_3 .

Les trois paramètres du régulateur sont maintenant fixés à,

$$K_p = 32,13$$

$$K_d = 0,3598$$

$$K_i = 32,13$$

Il est important, à ce point, de vérifier l'effet de la variation des paramètres simultanément sur le lieu des pôles. On voit sur la figure B.14 ces effets pour différentes situations. La valeur de α_1 est posée à 150 et on fait varier α_2 et α_3 . Pour $\alpha_2 = 2000$, les pôles du système se situent sur la courbe en rouge dépendamment de la valeur de α_3 . Pour $\alpha_2 = 5000$, les pôles se situent sur la courbe en bleu dépendamment de la valeur de α_3 . La courbe en vert est pour $\alpha_2 = 5650$ et celle en noire pour $\alpha_2 = 5700$. On remarque que plus la valeur de α_2 augmente plus le système sera stable pour un grand interval de valeur de α_3 . La grandeur de α_2 est limitée dans notre cas par le microprocesseur.

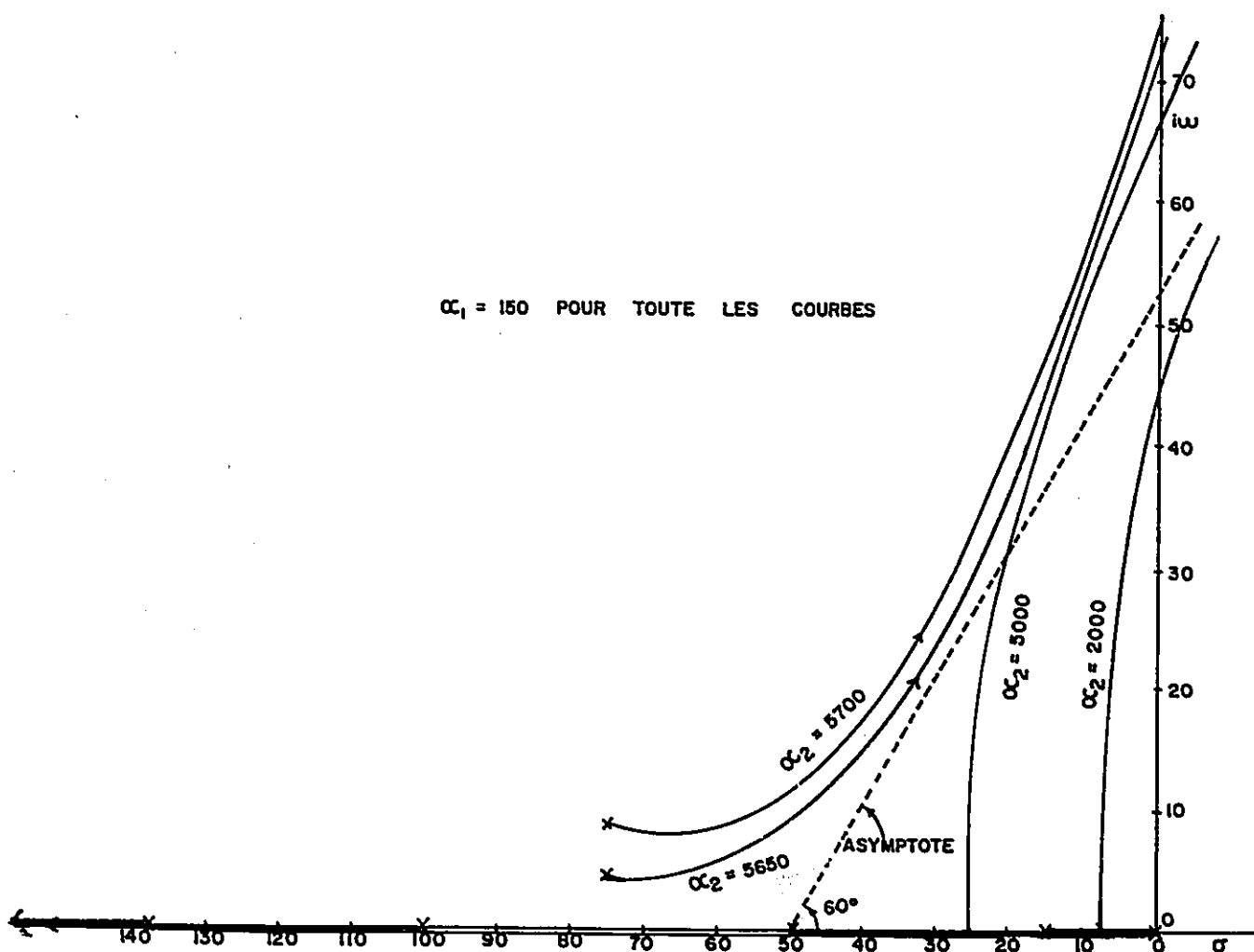


Figure B.14: Variation Simultané des paramètres sur le lieu des pôles.

C. ETUDE DE LA STABILITE DU SYSTEME.

La fonction de transfert du système pour une charge constante donnée par l'équation B12 peut être réécrite,

$$\frac{\theta_o}{\theta_d} = \frac{22,4 s^2 + 2000 s + 2000}{s^3 + 150 s^2 + 2000 s + 2000} \quad (B22)$$

Pour être stable, le système doit avoir tout ces pôles dans le plan gauche du lieu des pôles. Les pôles de B22 soit situés à $s = -2,688$, $s = -13,58$ et à $s = -135,3$. A partir de la réponse indicielle du système on pourrait aussi vérifier que le dépassement transitoire respecte la limite imposée et que le temps pour atteindre le régime permanent est aussi respecté. A partir de l'équation B22 on peut écrire la réponse indicielle dans le domaine de Laplace.

$$\theta_o(s) = \frac{1}{s} \cdot \frac{22,4 s^2 + 2000 s + 2000}{s^3 + 150 s^2 + 2000 s + 2000} \quad (B23)$$

En faisant la transformé Laplace inverse de B23 on obtiendra la réponse indicielle dans le domaine temps.

$$\theta_o(t) = 1 + 0,0821 e^{-1,09t} - 0,0640 e^{-135t} - 1,02 e^{-13,5t}$$

La figure B.5 nous montre le graphique de la réponse indicielle. On remarque que le dépassement transitoire est plus petit que 5% et que le temps avant d'atteindre le régime permanent est d'environ 1,2 secondes. On considère le régime permanent atteint lorsque le dépassement transitoire demeure au dessous de 2%. Le système a été étudié comme un système linéaire. Des expériences sur le montage expérimentale seront donc nécessaire afin de valider la présente analyse.

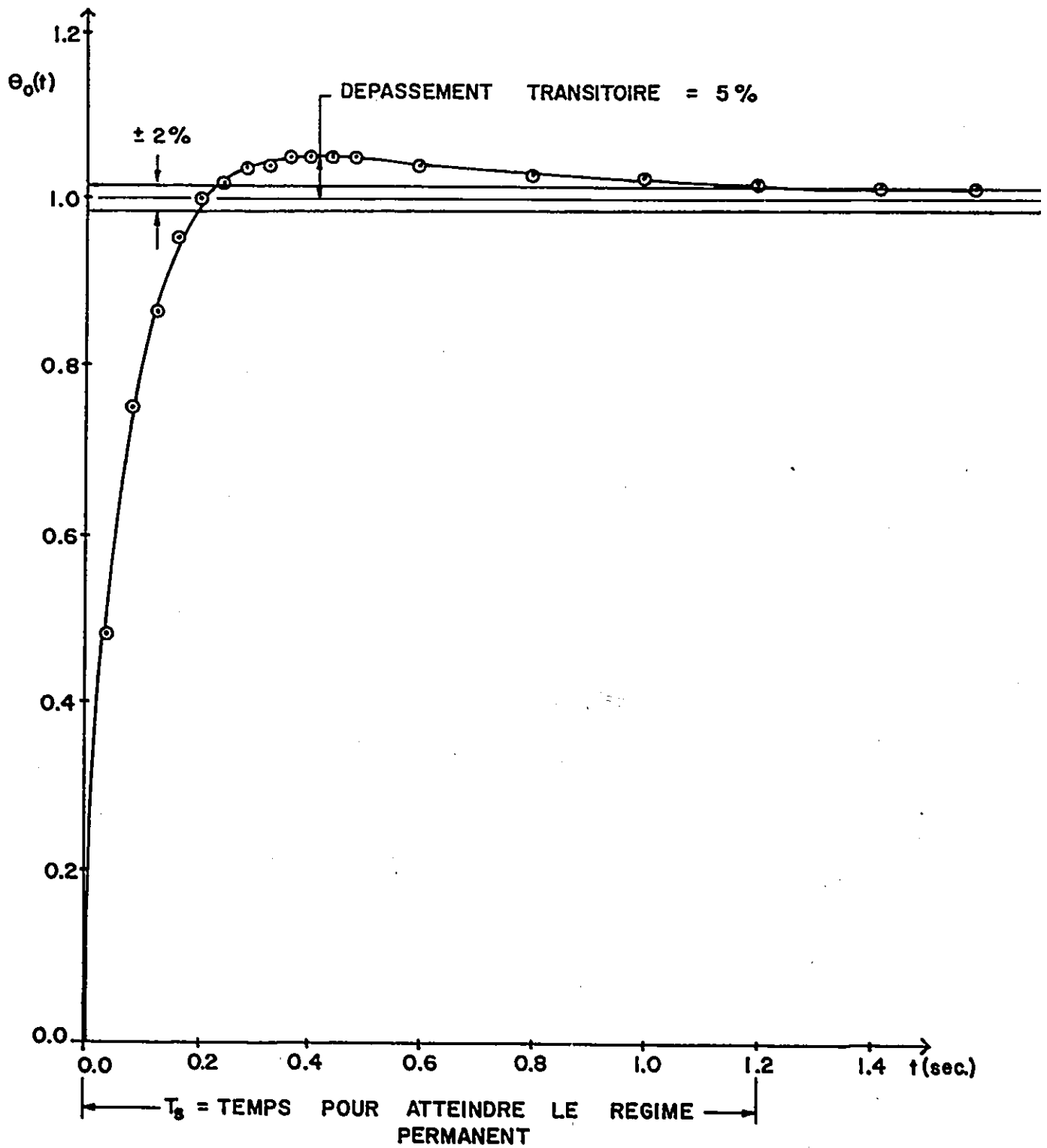


Figure B.15: La réponse indicielle.

D. L'EQUATION DU REGULATEUR.

L'équation du régulateur est maintenant connue dans le domaine de Laplace (Eq. B24). Comme un régulateur numérique sera implémenté, il est nécessaire d'avoir une équation de différences finies pour représenter celui-ci. La première étape dans l'obtention d'une telle équation est de transformer la présente équation,

$$D(s) = K_p + K_i/s + K_d s \quad (B24)$$

dans le domaine "z". Pour le régulateur digital, la partie proportionnelle est implémentée par une constante de proportionnalité, K_p . Par contre il y a diverses façons d'implémenter les parties intégrale et différentielle. Il a été choisi d'utiliser la méthode présentée par Kuo [2]. La partie intégrale, K_i/s est approximée par une intégration polygonale (eq. B25), tandis que la partie différentielle est représentée par l'équation B26.

$$\frac{K_i}{s} = \frac{K_i \cdot T \cdot (z+1)}{2 \cdot (z-1)} \quad (B25)$$

$$K_d s = \frac{K_d \cdot (z-1)}{T \cdot z} \quad (B26)$$

On peut illustrer le système numérique par la figure B.16.

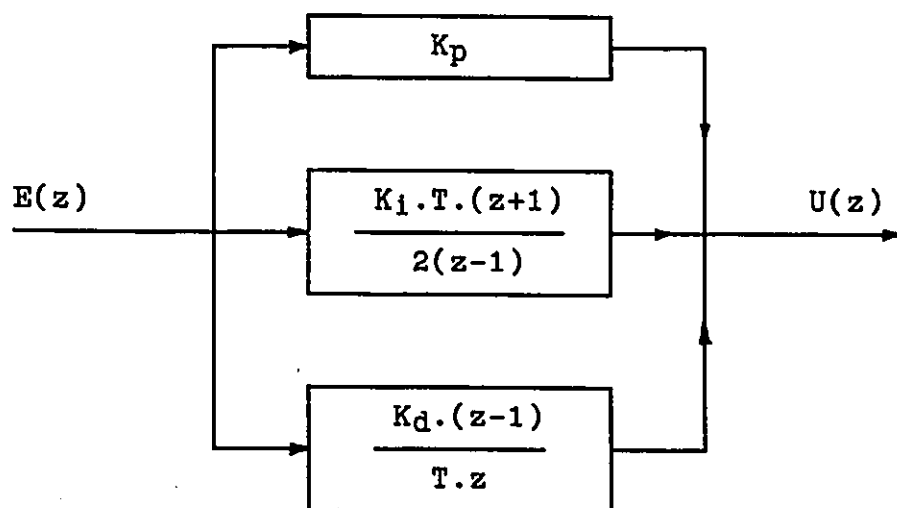


Figure B.16: Equation numérique du régulateur.

$$D(z) = \frac{(K_i T^2 + 2K_p T + 2K_d)z^2 + (K_i T^2 - 2K_p T - 4K_d)z + 2K_d}{2T.z.(z-1)}$$

En substituant les valeurs de K_p , K_i et K_d , de même que la période d'échantillonnage T qui est égale à 0,004 sec (la section suivante justifiera le choix de 0,004 comme période) on obtiendra l'équation suivante;

$$D(z) = \frac{U(z)}{E(z)} = \frac{122 z^2 - 212 z + 90}{z(z-1)} \quad (B27)$$

De B27 l'équation pour du régulateur PID numérique est

$$U(kT) = 122 E(kT) - 212 E(kT-1) + 90 E(kT-2) + U(kT-1) \quad (B28)$$

La référence [2] section 10.4 explique comment obtenir B28 de B27. Cette équation sera programmée dans le programme de commande du microprocesseur.

E. LA PERIODE D'ECHANTILLONNAGE

La sélection de la période d'échantillonnage est très importante car celle-ci peut avoir des effets négatifs sur la réponse du system. Ogata [5] dans son livre explique bien ces effets, qui se résume à ceci.

- 1 - Plus grande est la période, le moins stable sera le système.
- 2 - Du coté opposé, plus la période est petite plus la valeur critique du gain pour que le système soit stable devient grande. Le système se comportera alors de plus en plus comme un système continu. Par contre, comme un système numérique est utilisé, la grandeur de K est limité par la capacité du microprocesseur (1 ou 2 octets pour le Z8 utilisé).

- 3 - Si la période est trop grande le dépassement transitoire maximum, pour la réponse indicielle, sera beaucoup plus grand que celui prédit par le facteur d'amortissement.

De plus la période doit respecter le théorème de Shannon qui dit que: "il est nécessaire que la fréquence d'échantillonnage soit au moins égale à deux fois la plus grande fréquence contenue dans le spectre du signal $F(t)$." Ce théorème ne nous limitait pas pour la présente application, car nous avons le choix des signaux d'entrées. Par contre, dans le montage expérimental nous sommes limités par la précision du lecteur optique. A vitesse maximum l'ensemble disque rotatif et module optoélectronique produit un signal avec une période de 0.03sec. La période d'échantillonnage du système de commande doit donc être plus petite que 0.03sec.

$$T \leq 0.03 \text{ sec.}$$

De plus la période doit être assez grande pour permettre au programme de commande de faire tout les calculs nécessaires dans la période kT avant de passer à la période $kT + 1$. Il a été estimé, à partir du programme en langage assembler (annexe C), que un maximum de 18980 cycles de calcul sont nécessaire par période. Etant donné la fréquence du microprocesseur à 7,3728 MHz, le temps requis pour les calculs est de 0,00257 sec/période. La période d'échantillonnage doit donc être plus grande que 0.00257 sec.

$$T \geq 0,00257 \text{ sec.}$$

Une période de 0,004 sec. a été choisie.

$$T = 0,004 \text{ sec.}$$

ANNEXE C

LE PROGRAMME DE COMMANDE DU MOTEUR

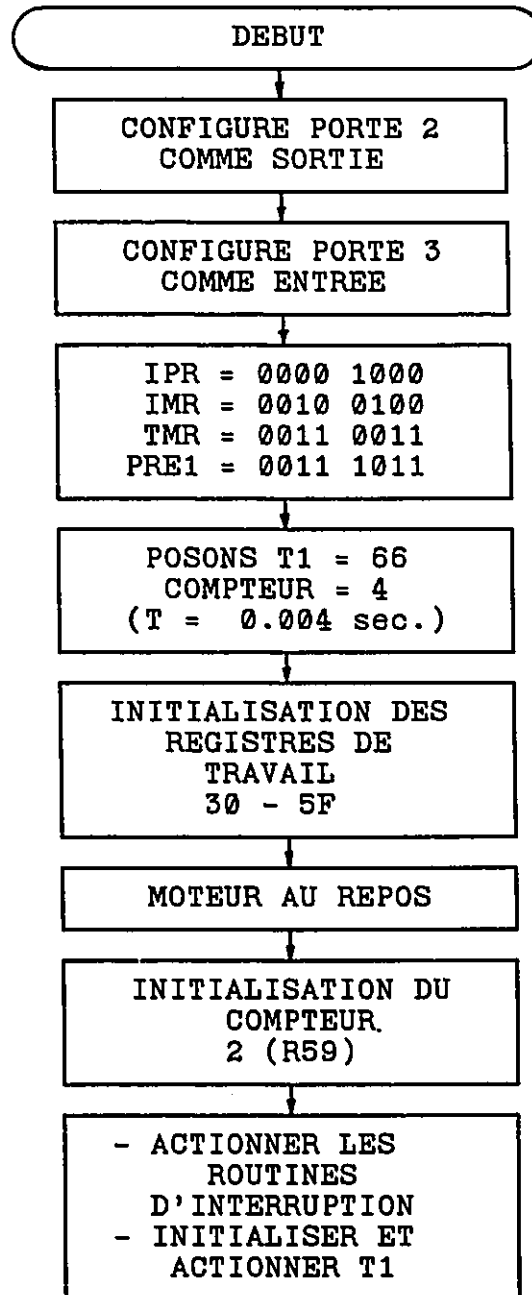
LE PROGRAMME DE COMMANDE

La fonction du programme de commande est de lire le signal de position, incrémenter le compteur de position, calculer l'erreur entre la position désirée et celle obtenue, décider de l'action à prendre suivant l'équation du régulateur dérivée à l'annexe B et envoyer le signal de commande au circuit. Le programme contrôle aussi l'acquisition des données. Le programme a été écrit en langage assembleur pour le Z8671. Cette annexe comprend l'organigramme, la structure de la mémoire et le programme complet.

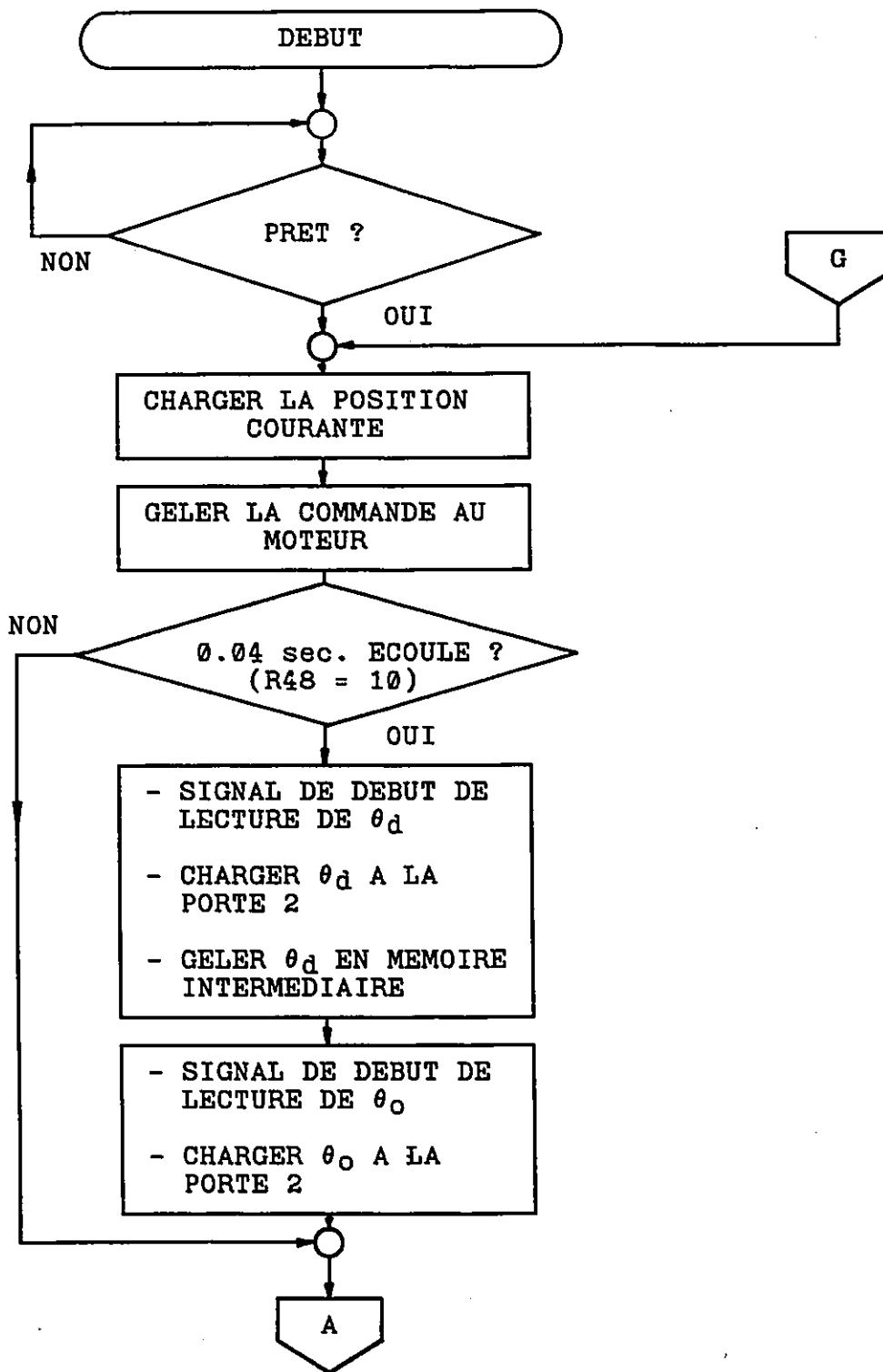
STRUCTURE DE LA MEMOIRE

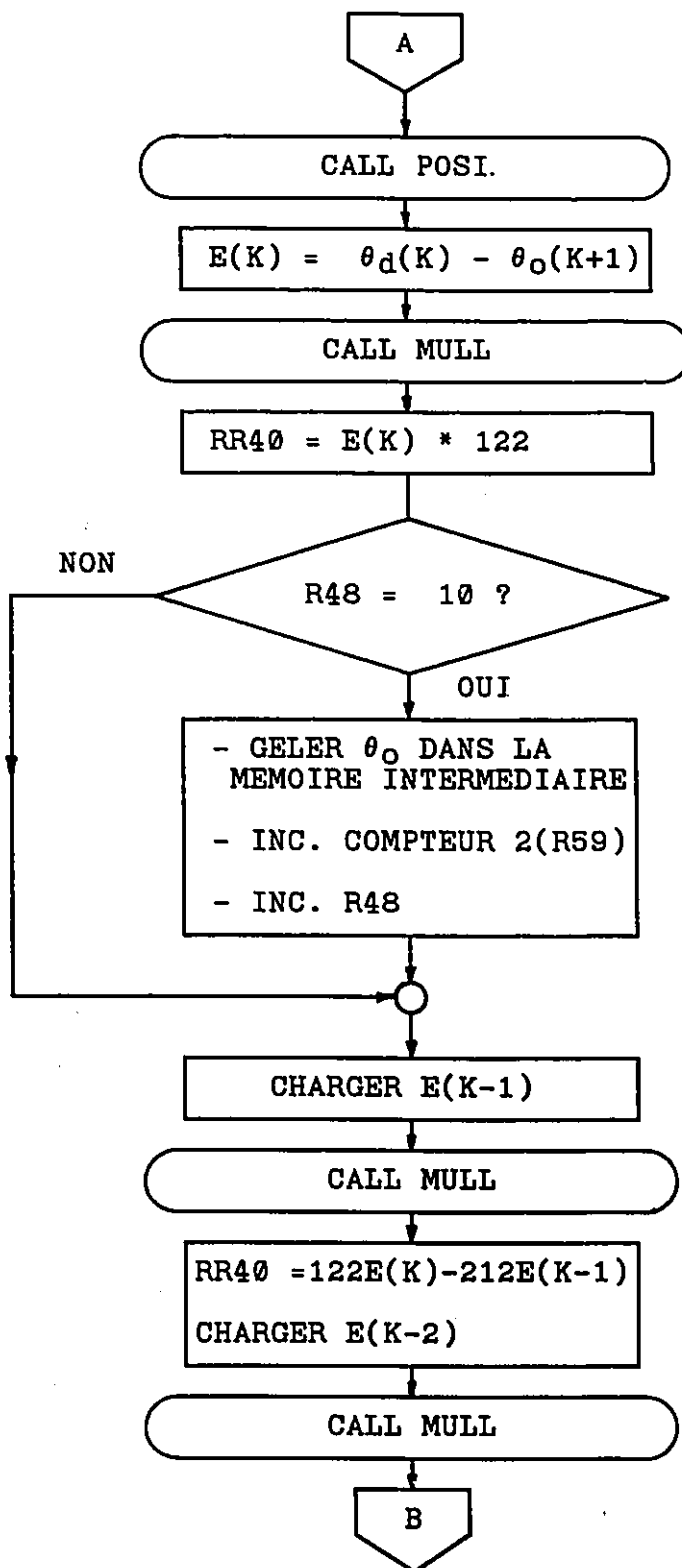
ADRESSE HEXADECIMALE	VARIABLE
Mémoire externe (D/A) 8001	Indicateur de mouvement du moteur
Registres de travail 30-31 32-33 34-35 36-37 38-39 3A-3B 3C-3D 3E-3F 40-41 42 43 44-45 46-47 48 4A 4B 4C-4D 52-53 59	 θ_0 E(k) E(k-1) U(k) θ_d θ_0 E(k-2) U(k-1) multiplicande diviseur multiplicateur utilisé pour l'addition et la soustraction non-utilisé compteur 1 compteur de période d'échantillonnage indicateur de fin période d'échantillonnage utilisé par la routine de multiplication données de la mémoire externe compteur 2

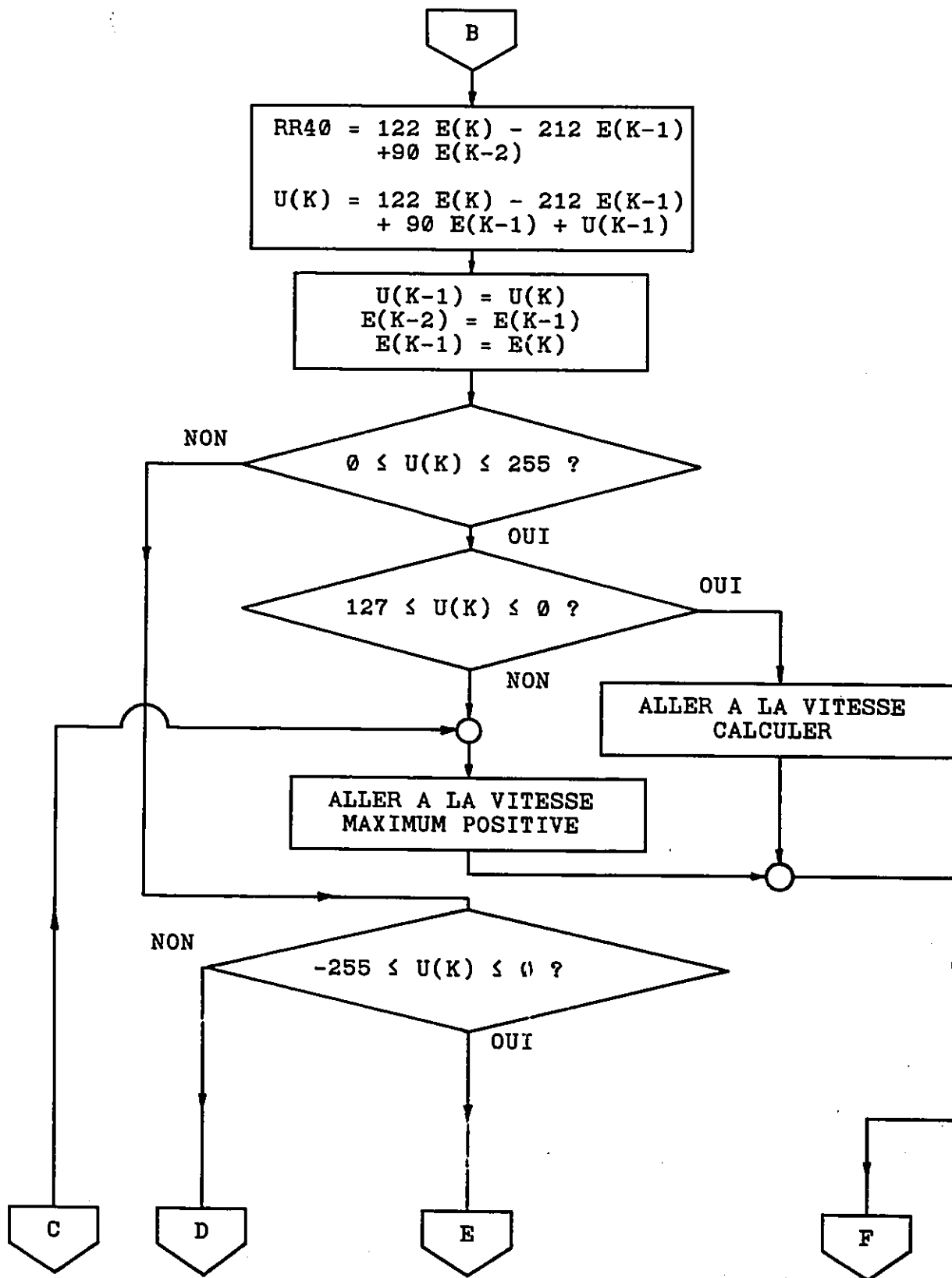
INITIALISATION

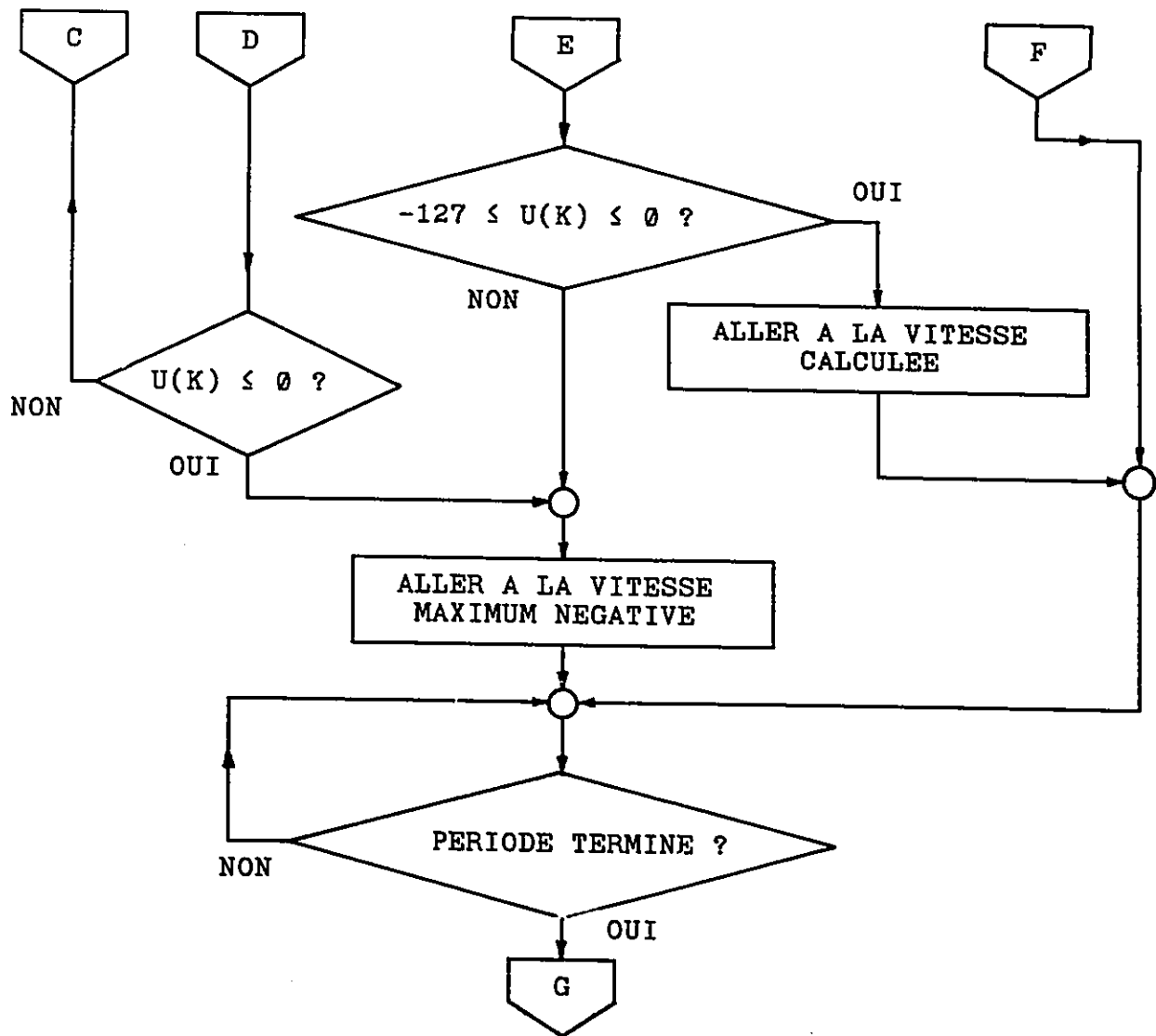


ROUTINE DE COMMANDE

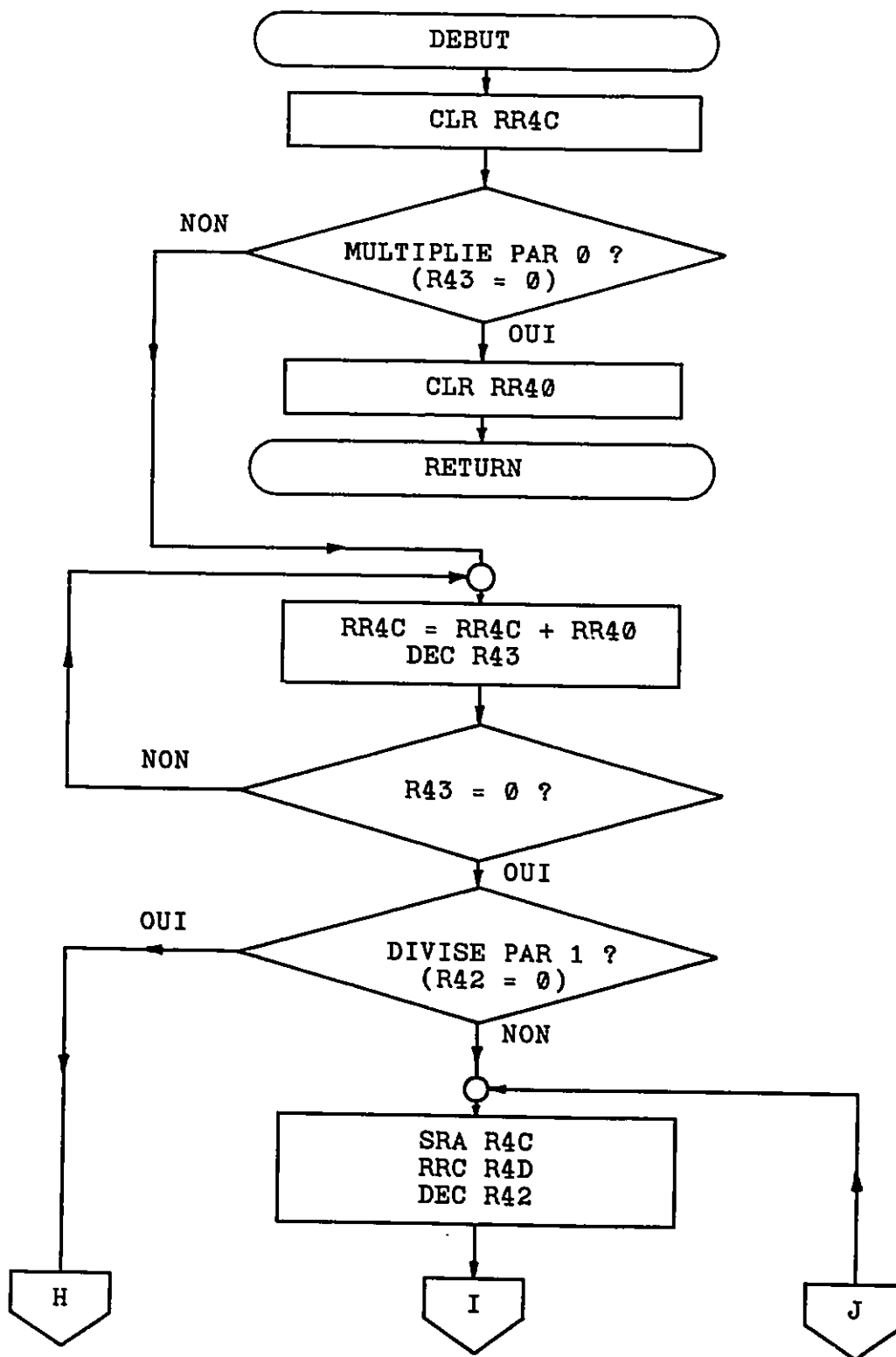


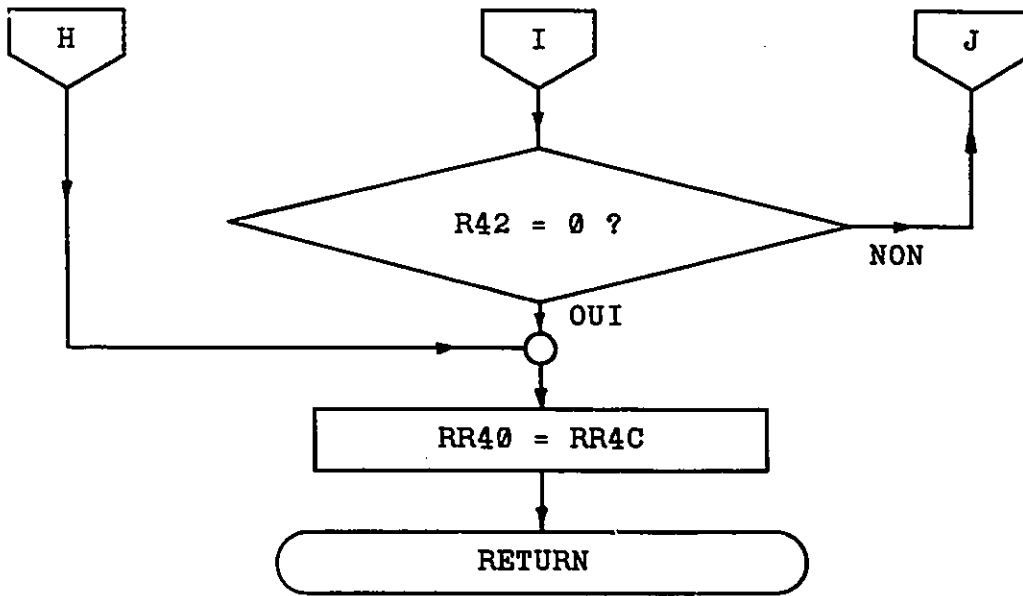






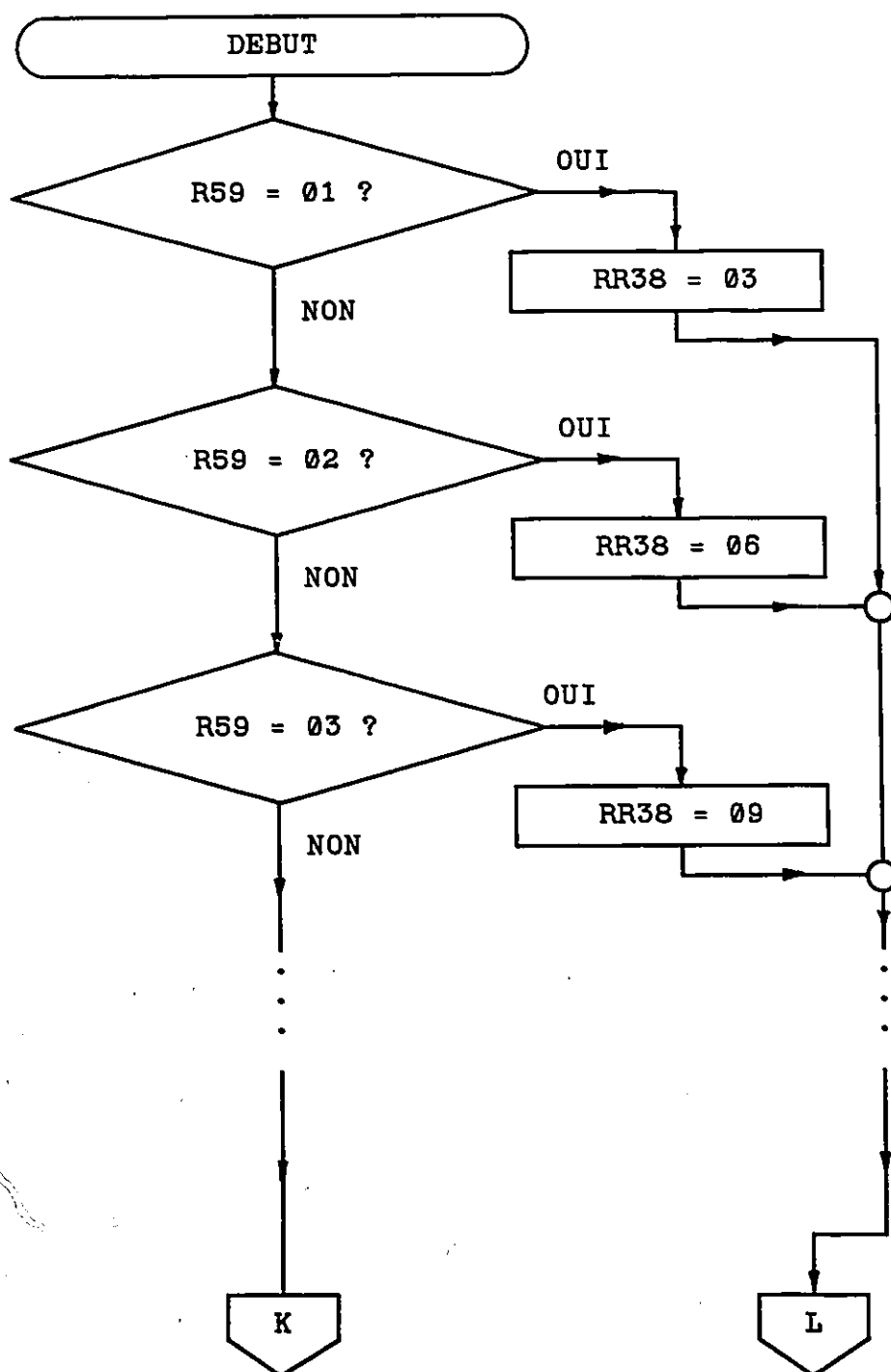
SOUS-ROUTINE MULL

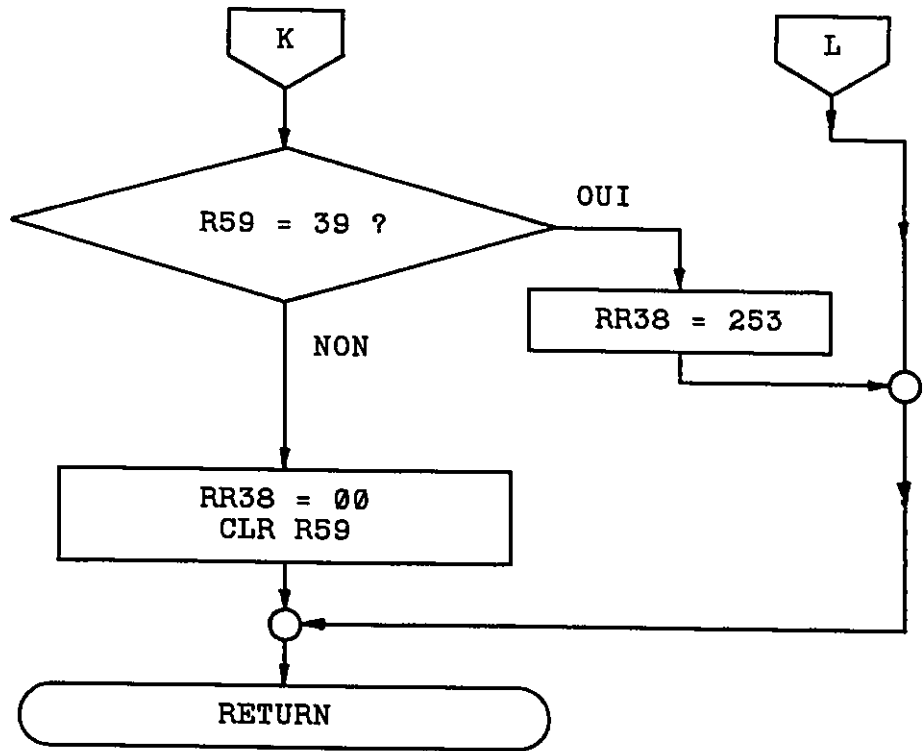




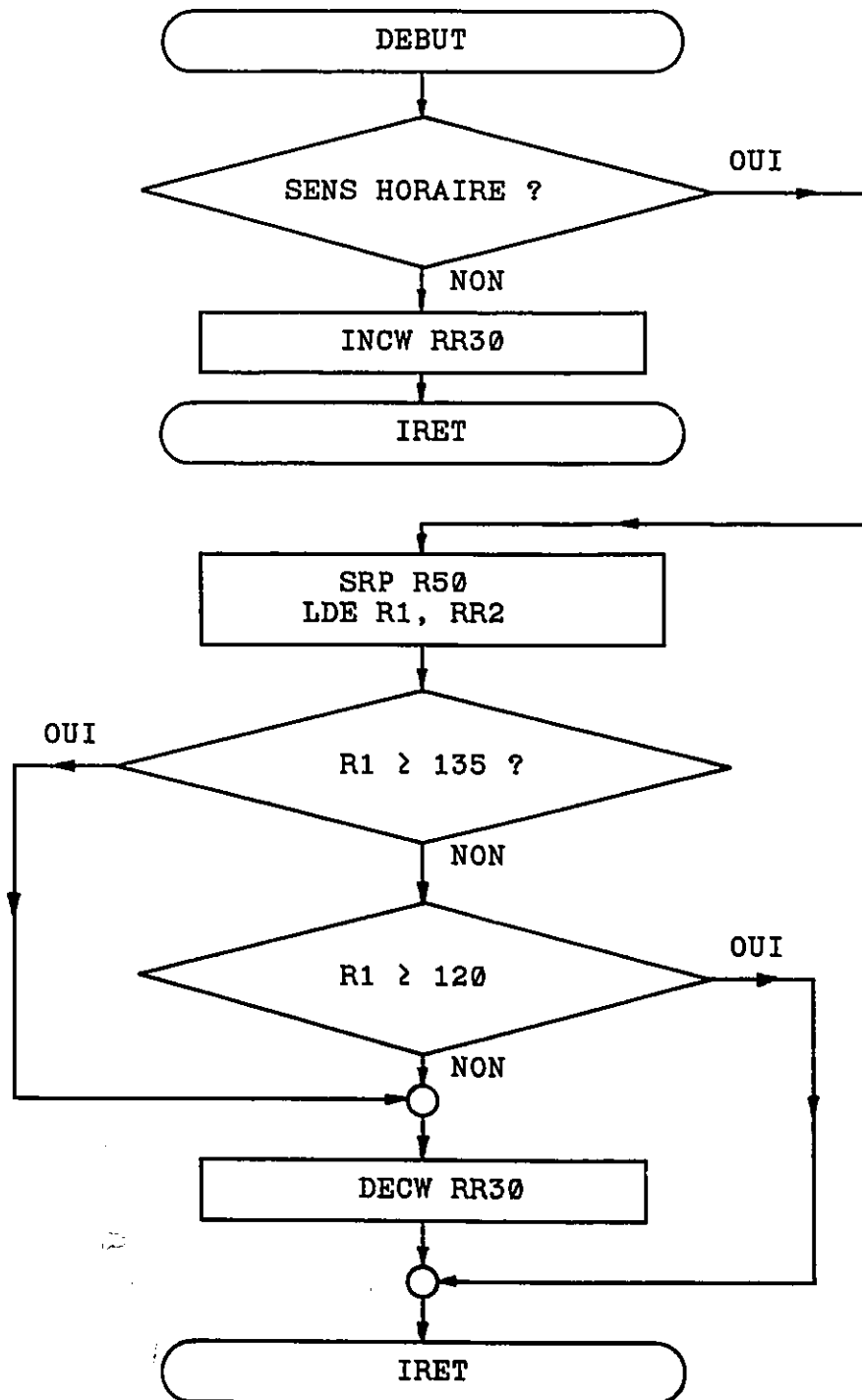
SOUS-ROUTINE POSI

Cette sous-routine modélise une entrée sinusoïdale. Une organigramme simplifié est présenté ici. Le processus étant répétitif il serait inutile de le présenter au complet.

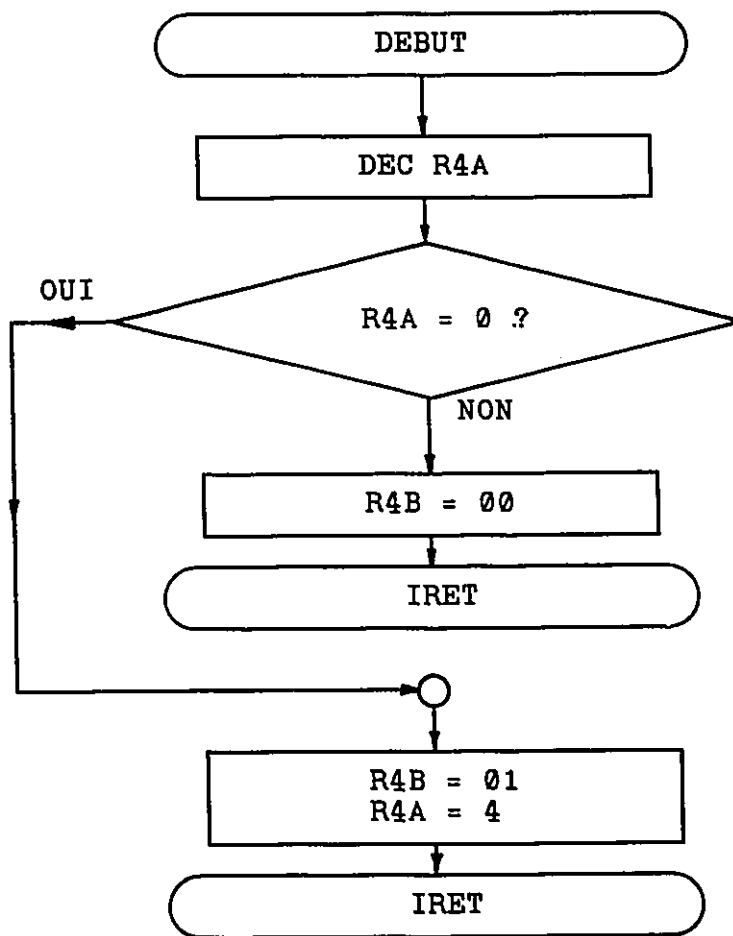




SOUS-ROUTINE D'INTERRUPTION IQ2



SOUS-ROUTINE D'INTERRUPTION IQ5



```

1020          ORG 1020H
1020 FF      NOP
1021 FF      NOP
*****
*
*              INITIALISATION
*
*****

1022 E6F600      LD P2M, #00H      ; Configure porte 2
                                   ; comme sortie
1025 E6F741      LD P3M, #41H      ; Configure porte 3
                                   ; comme entrée
1028 E6F908      LD IPR, #08H      ; IPR = 0000 1000
102B E6FB24      LD IMR, #24H      ; IMR = 0010 0100
102E E6F133      LD TMR, #33H      ; TMR = 0011 0011
1031 E6F33B      LD PRE1, #3BH     ; PRE1 = 0011 1011
1034 E6F242      LD T1, #66        ; T1 = 66
                                   ; (t=0.001 sec)
1037 E62F30      LD %2F, #30H      ; Initialisation
103A B12F        PLUS CLR 02FH      ; des registres de
103C 202F        INC 2FH          ; travail
103E A62F60      CP 2FH, #60H      ; 30 - 5F
1041 ED103A      JP NZ, PLUS      ;
1044 E64A04      LD 4AH, #4        ; Compteur = 4
                                   ; (T = 0.004sec)
1047 E60300      LD 03, #00        ; Initialisation de
                                   ; la porte 3
104A E60280      LD 02, #128       ; Moteur au repos
104D 060310      ADD 03, #10H      ; Envoyer la com-
                                   ; mande au moteur
1050 E65280      LD 52H, #80H      ; 04E = 80
                                   ; (4E-4F = 8001)
1053 E65301      LD 53H, #01H      ;
1056 E65901      LD 59H, #01H      ; Initialisation du
                                   ; compteur 2
1059 9F          EI              ; Actionner la rou-
                                   ; tine d'interrup-
                                   ; tion.
105A 46F10C      OR TMR, #0CH      ; Initialiser et
                                   ; actionner T1

```

```

*****
*
*           ROUTINE DE COMMANDE
*
*****

```

```

105D 760308   DEBUT      TM 03, #08H           ; Attendre le
1060 6D105D           JP Z, DEBUT           ; signal de départ
                                           ; du programme
                                           ; fortran

1063 E43040   SERVO      LD 40H, 30H           ; Charger position
1066 E43141           LD 41H, 31H           ; courante
1069 E4403A           LD 3AH, 40H
106C E4413B           LD 3BH, 41H
106F 260310           SUB 03, #10H         ; Geler la commande
                                           ; au moteur
1072 A6480A           CP 48H, #10          ; Est-ce que 0.04
                                           ; seconde sont
                                           ; passée
1075 5D108A           JP MI, S1            ; Si oui, continuer
1078 060320           ADD 03, #20H         ; Envoyer le signal
                                           ; de début de lec-
                                           ; ture de position
                                           ; désirée
107B E43902           LD 02, 39H          ; Charger la posi-
                                           ; tion uésirée
                                           ; a la porte 2

107E FF         NOP
107F FF         NOP
1080 FF         NOP
1081 260320           SUB 03, #20H         ; Geler la position
                                           ; désirée
                                           ; dans la mémoire
                                           ; intermédiaire
1084 060340           ADD 03, #40H         ; Envoyer la signal
                                           ; de début de lec-
                                           ; ture de position
                                           ; obtenue
1087 E43B02           LD 02, 3BH          ; Charger la posi-
                                           ; tion obtenue
                                           ; à la porte 2
108A D61199   S1         CALL POSI           ; Aller lire la
                                           ; nouvelle comman-
                                           ; de de position
                                           ; désirée

108D E43844           LD 44H, 38H
1090 E43945           LD 45H, 39H
1093 243B45           SUB 45H, 3BH         ; Trouver l'erreur
                                           ; entre la position
                                           ; désirée et
1096 343A44           SBC 44H, 3AH        ; obtenue E(k)

```

1099 E44432	LD 32H, 44H	; Charger E(k)
109C E44533	LD 33H, 45H	
109F E44541	LD 41H, 45H	
10A2 E44440	LD 40H, 44H	
10A5 E64200	LD 42H, #00H	; Diviser par 1
10A8 E6437A	LD 43H, #122	; Mult. par 122
10AB A6480A	CP 48H, #10	; Si 0.04 sec. sont
		; passée
10AE 5D10B8	JP MI, S3	
10B1 260340	SUB 03, #40H	; Geler le signal
		; de position
		; obtenue dans la
		; mémoire
		; intermédiaire
10B4 2059	INC 59H	; Incrémenter le
		; compteur 2
10B6 B048	CLR 48H	; Réinitialiser
		; compteur 1
		; 122 E(k)
10B8 D61168	S3 CALL MULL	
10BB E44145	LD 45H, 41H	
10BE E44044	LD 44H, 40H	
10C1 E43541	LD 41H, 35H	; Charger E(k-1)
10C4 E43440	LD 40H, 34H	
10C7 E64200	LD 42H, #00H	; Diviser par 1
10CA E643D4	LD 43H, #212	; Mult. par 212
10CD D61168	CALL MULL	; 212E(k-1)
10D0 244145	SUB 45H, 41H	; 122E(k)-212E(k-1)
10D3 344044	SBC 44H, 40H	
10D6 E43D41	LD 41H, 3DH	; Charger E(k-2)
10D9 E43C40	LD 40H, 3CH	
10DC E64200	LD 42H, #00H	; Diviser par 1
10DF E6435A	LD 43H, #90	; Mult. par 90
10E2 D61168	CALL MULL	; 90 E(k-2)
10E5 044145	ADD 45H, 41H	; 122E(k)-212E(k-1)
10E8 144044	ADC 44H, 40H	; + 90 E(k-2)
10EB 043F45	ADD 45H, 3FH	; U(K) = 122E(K)
		; -212E(K-1)
		; +90E(K-2)+U(K-1)
10EE 143E44	ADC 44H, 3EH	; Charger U(K) dans
10F1 E44436	LD 36H, 44H	; son registre
10F4 E44537	LD 37H, 45H	; E(K-2) = E(K-1)
10F7 E4343C	LD 3CH, 34H	
10FA E4353D	LD 3DH, 35H	
10FD E43234	LD 34H, 32H	; E(K-1) = E(K)
1100 E43335	LD 35H, 33H	
1103 E4363E	LD 3EH, 36H	; U(K-1) = U(K)
1106 E4373F	LD 3FH, 37H	
1109 E43741	LD 41H, 37H	
110C E43640	LD 40H, 36H	
110F E64200	LD 42H, #0H	; Diviser par 1
1112 E64301	LD 43H, #1	; Multiplier par 1

1115	D61168		CALL MULL	; U(K) X 1
1118	A64000		CP 40H, #00H	; Est-ce que
111B	6D114E		JP Z, D	; $0 < U(K) < 255 ?$
111E	A640FF		CP 40H, #0FFH	; Si oui aller à D
1121	6D113C		JP Z, B2	; Est-ce que
1124	764080		TM 40H, #128	; $-255 < U(K) < 0 ?$
1127	6D1133		JP Z, B	; Si oui aller à B2
112A	060310	A	ADD 03, #10H	; Est-ce que
112D	E60200		LD 02, #00H	; U(K) < 0?
1130	8D115D		JP E	; Si non aller à B
1133	060310	B	ADD 03, #10H	; Envoyer commande
				; au moteur
1136	E602FF		LD 02, #0FFH	; Aller a vitesse
1139	8D115D		JP E	; maximum +
113C	764180	B2	TM 41H, #128	; Terminer
113F	6D112A		JP Z, A	; Est-ce que
1142	264180		SUB 41H, #128	; $-127 < U(K) < 0 ?$
1145	060310		ADD 03, #10H	; Si non aller à A
1148	E44102		LD 02, 41H	; Ajuster l'échelle
114B	8D115D		JP E	; pour DAC
114E	764180	D	TM 41H, #128	; Envoyer commande
1151	ED1133		JP NZ, B	; au moteur
1154	064180		ADD 41H, #128	; Aller à la vites-
1157	060310		ADD 03, #10H	; se calculée
115A	E44102		LD 02, 41H	; Terminer
115D	A64B01	E	CP 4BH, #01H	; Est-ce que
				; $127 < U(K) < 0 ?$
1160	ED115D		JP NZ, E	; Si non aller a B
1163	2048		INC 48H	; Ajuster l'échelle
1165	8D1063		JP SERVO	; pour DAC
				; Envoyer commande
				; au moteur
				; Aller à la vites-
				; se calculée
				; Est-ce que la
				; période d'échan-
				; tillonnage est
				; terminée
				; Si non, attendre
				; Incrémenter
				; compteur 1
				; Si période termi-
				; née aller au
				; début.

```

*****
*
* Sous-routine MULL: multiplie un nombre de 16 *
* bit contenue dans les registres (40-41) par *
* un coefficient de la forme n/2d, ou n et d *
* sont des entiers. *
*
*      0 < n < 255 and 0 < d < 9 *
*
* n est contenue dans le registre 43 et d dans *
* le registre 42. Le resultat est retourné *
* dans les registres (40-41). *
*
*****

```

1168 B04C	MULL	CLR 4CH	
116A B04D		CLR 4DH	
116C A64300		CP 43H, #00H	; Est-ce qu'on
			; multiplie par 0
116F ED1177		JP NZ, MLOOP	; Si non aller à
			; MLOOP
1172 B041		CLR 41H	; Si oui, mettre 0
			; dans 40 et 41 et
1174 B040		CLR 40H	; retourner
1176 AF		RET	
1177 04414D	MLOOP	ADD 4DH, 41H	
117A 14404C		ADC 4CH, 40H	; Boucle de
117D 0043		DEC 43H	; Multiplication
117F ED1177		JP NZ, MLOOP	
1182 FF		NOP	
1183 A64200		CP 42H, #0H	; Est-ce qu'on
			; divise par 1?
1186 6D1192		JP Z, STOR	; Si oui, aller à
			; STOR
1189 D04C	DLOOP	SRA 4CH	
118B C04D		RRC 4DH	; Boucle de
118D 0042		DEC 42H	; Division
118F ED1189		JP NZ, DLOOP	
1192 E44D41	STOR	LD 41H, 4DH	; Mettre les résul
1195 E44C40		LD 40H, 4CH	; tats dans (40-41)
1198 AF		RET	

```

*****
*
* Sous-routine POSI: Cette routine calcule la *
* position désirée à chaque interval de temps *
* k. Le resultat est mis dans le registre *
* (38-39). On a ici une entrée sinusoidale *
*
*****

```

```

1199 A65901      POSI      CP 59H, #01
119C ED11A8      JP NZ, F1
119F E63800      LD 38H, #00
11A2 E63903      LD 39H, #3
11A5 8D137B      JP F40
11A8 A65902      F1        CP 59H, #02
11AB ED11B4      JP NZ, F2
11AE E63906      LD 39H, #6
11B1 8D137B      JP F40
11B4 A65903      F2        CP 59H, #03
11B7 ED11C0      JP NZ, F3
11BA E63909      LD 39H, #9
11BD 8D137B      JP F40
11C0 A65904      F3        CP 59H, #04
11C3 ED11CC      JP NZ, F4
11C6 E6390C      LD 39H, #12
11C9 8D137B      JP F40
11CC A65905      F4        CP 59H, #05
11CF ED11D8      JP NZ, F5
11D2 E6390E      LD 39H, #14
11D5 8D137B      JP F40
11D8 A65906      F5        CP 59H, #06
11DB ED11E4      JP NZ, F6
11DE E63910      LD 39H, #16
11E1 8D137B      JP F40
11E4 A65907      F6        CP 59H, #07
11E7 ED11F0      JP NZ, F7
11EA E63912      LD 39H, #18
11ED 8D137B      JP F40
11F0 A65908      F7        CP 59H, #08
11F3 ED11FC      JP NZ, F8
11F6 E63913      LD 39H, #19
11F9 8D137B      JP F40
11FC A65909      F8        CP 59H, #09
11FF ED1208      JP NZ, F9
1202 E63914      LD 39H, #20
1205 8D137B      JP F40
1208 A6590A      F9        CP 59H, #10
120B ED1214      JP NZ, F10
120E E63914      LD 39H, #20
1211 8D137B      JP F40
1214 A6590B      F10       CP 59H, #11
1217 ED1220      JP NZ, F11
121A E63914      LD 39H, #20

```

121D	8D137B		JP F40
1220	A6590C	F11	CP 59H, #12
1223	ED122C		JP NZ, F12
1226	E63913		LD 39H, #19
1229	8D137B		JP F40
122C	A6590D	F12	CP 59H, #13
122F	ED1238		JP NZ, F13
1232	E63912		LD 39H, #18
1235	8D137B		JP F40
1238	A6590E	F13	CP 59H, #14
123B	ED1244		JP NZ, F14
123E	E63910		LD 39H, #16
1241	8D137B		JP F40
1244	A6590F	F14	CP 59H, #15
1247	ED1250		JP NZ, F15
124A	E6390E		LD 39H, #14
124D	8D137B		JP F40
1250	A65910	F15	CP 59H, #16
1253	ED125C		JP NZ, F16
1256	E6390C		LD 39H, #12
1259	8D137B		JP F40
125C	A65911	F16	CP 59H, #17
125F	ED1268		JP NZ, F17
1262	E63909		LD 39H, #9
1265	8D137B		JP F40
1268	A65912	F17	CP 59H, #18
126B	ED1274		JP NZ, F18
126E	E63906		LD 39H, #6
1271	8D137B		JP F40
1274	A65913	F18	CP 59H, #19
1277	ED1280		JP NZ, F19
127A	E63903		LD 39H, #3
127D	8D137B		JP F40
1280	A65914	F19	CP 59H, #20
1283	ED128C		JP NZ, F20
1286	E63900		LD 39H, #0
1289	8D137B		JP F40
128C	A65915	F20	CP 59H, #21
128F	ED129B		JP NZ, F21
1292	E638FF		LD 38H, #255
1295	E639FD		LD 39H, #253
1298	8D137B		JP F40
129B	A65916	F21	CP 59H, #22
129E	ED12A7		JP NZ, F22
12A1	E639FA		LD 39H, #250
12A4	8D137B		JP F40
12A7	A65917	F22	CP 59H, #23
12AA	ED12B3		JP NZ, F23
12AD	E639F7		LD 39H, #247
12B0	8D137B		JP F40
12B3	A65918	F23	CP 59H, #24
12B6	ED12BF		JP NZ, F24
12B9	E639F4		LD 39H, #244
12BC	8D137B		JP F40

12BF A65919	F24	CP 59H, #25
12C2 ED12CB		JP NZ, F25
12C5 E639F2		LD 39H, #242
12C8 8D137B		JP F40
12CB A6591A	F25	CP 59H, #26
12CE ED12D7		JP NZ, F26
12D1 E639F0		LD 39H, #240
12D4 8D137B		JP F40
12D7 A6591B	F26	CP 59H, #27
12DA ED12E3		JP NZ, F27
12DD E639EE		LD 39H, #238
12E0 8D137B		JP F40
12E3 A6591C	F27	CP 59H, #28
12E6 ED12EF		JP NZ, F28
12E9 E639ED		LD 39H, #237
12EC 8D137B		JP F40
12EF A6591D	F28	CP 59H, #29
12F2 ED12FB		JP NZ, F29
12F5 E639EC		LD 39H, #236
12F8 8D137B		JP F40
12FB A6591E	F29	CP 59H, #30
12FE ED1307		JP NZ, F30
1301 E639EC		LD 39H, #236
1304 8D137B		JP F40
1307 A6591F	F30	CP 59H, #31
130A ED1313		JP NZ, F31
130D E639EC		LD 39H, #236
1310 8D137B		JP F40
1313 A65920	F31	CP 59H, #32
1316 ED131F		JP NZ, F32
1319 E639ED		LD 39H, #237
131C 8D137B		JP F40
131F A65921	F32	CP 59H, #33
1322 ED132B		JP NZ, F33
1325 E639EE		LD 39H, #238
1328 8D137B		JP F40
132B A65922	F33	CP 59H, #34
132E ED1337		JP NZ, F34
1331 E639F0		LD 39H, #240
1334 8D137B		JP F40
1337 A65923	F34	CP 59H, #35
133A ED1343		JP NZ, F35
133D E639F2		LD 39H, #242
1340 8D137B		JP F40
1343 A65924	F35	CP 59H, #36
1346 ED134F		JP NZ, F36
1349 E639F4		LD 39H, #244
134C 8D137B		JP F40
134F A65925	F36	CP 59H, #37
1352 ED135B		JP NZ, F37
1355 E639F7		LD 39H, #247
1358 8D137B		JP F40

135B A65926	F37	CP 59H, #38
135E ED1367		JP NZ, F38
1361 E639FA		LD 39H, #250
1364 8D137B		JP F40
1367 A65927	F38	CP 59H, #39
136A ED1373		JP NZ, F39
136D E639FD		LD 39H, #253
1370 8D137B		JP F40
1373 E63900	F39	LD 39H, #00
1376 B058		CLR 58H
1378 E65900		LD 59H, #00
137B AF	F40	RET

```

*****
*
*   Routine d'interuption IQ2: incrémentation *
*   de la position                             *
*
*****

```

```

137C 760304   IQ2      TM 03, #04H           ; Vérifier la
                                           ; direction du
137F ED1385   JP NZ, MOINS          ; mouvement
1382 A030     INCW 30H           ; Direction sens
                                           ; anti-horaire

1384 BF       IRET
1365 3150     MOINS      SRP #50H          ; Ignorer l'inte-
                                           ; ruption si le
1387 8212     LDE R1, @RR2       ; moteur n'est pas
1389 A6E187   CP R1, #135        ; en mouvement
138C DD1395   JP PL, SOUS
138F A6E178   CP R1, #120
1392 DD1397   JP PL, FIN
1395 8030     SOUS      DECW 30H          ; Direction sens
                                           ; horaire

1397 BF       FIN       IRET

```

```

*****
*
* Routine d'interuption IQ5: Période d'échan- *
* tillonnage *
*
*****

```

```

1398 004A      IQ5      DEC 4AH          ; Décrémenter le
                ; compteur de la
                ; période d'échan-
139A 6D13A1          JP Z, SETT      ; tillonnage
                ; Si 0, alors le
                ; période est
139D E64B00          LD 4BH, #00H      ; terminée
                ; Si non 0, alors
13A0 BF          IRET          ; continuer
13A1 E64B01      SETT      LD 4BH, #01H      ; Charger l'indi-
                ; cateur de fin
13A4 E64A06          LD 4AH, #4          ; de période
                ; Réinitialiser le
                ; compteur
13A7 BF          IRET          ; de periode

```

```
*****  
*  
*           ADRESSES D'INTERUPTION           *  
*  
*****
```

```
100F          ORG 100FH          ; Location de IQ5  
100F 8D1398          JP IQ5
```

```
1006          ORG 1006H         ; Location de IQ2  
1006 8D137C          JP IQ2
```

ANNEXE D

**LE PROGRAMME DE DETECTION
AUTOMATIQUE DES ANOMALIES**

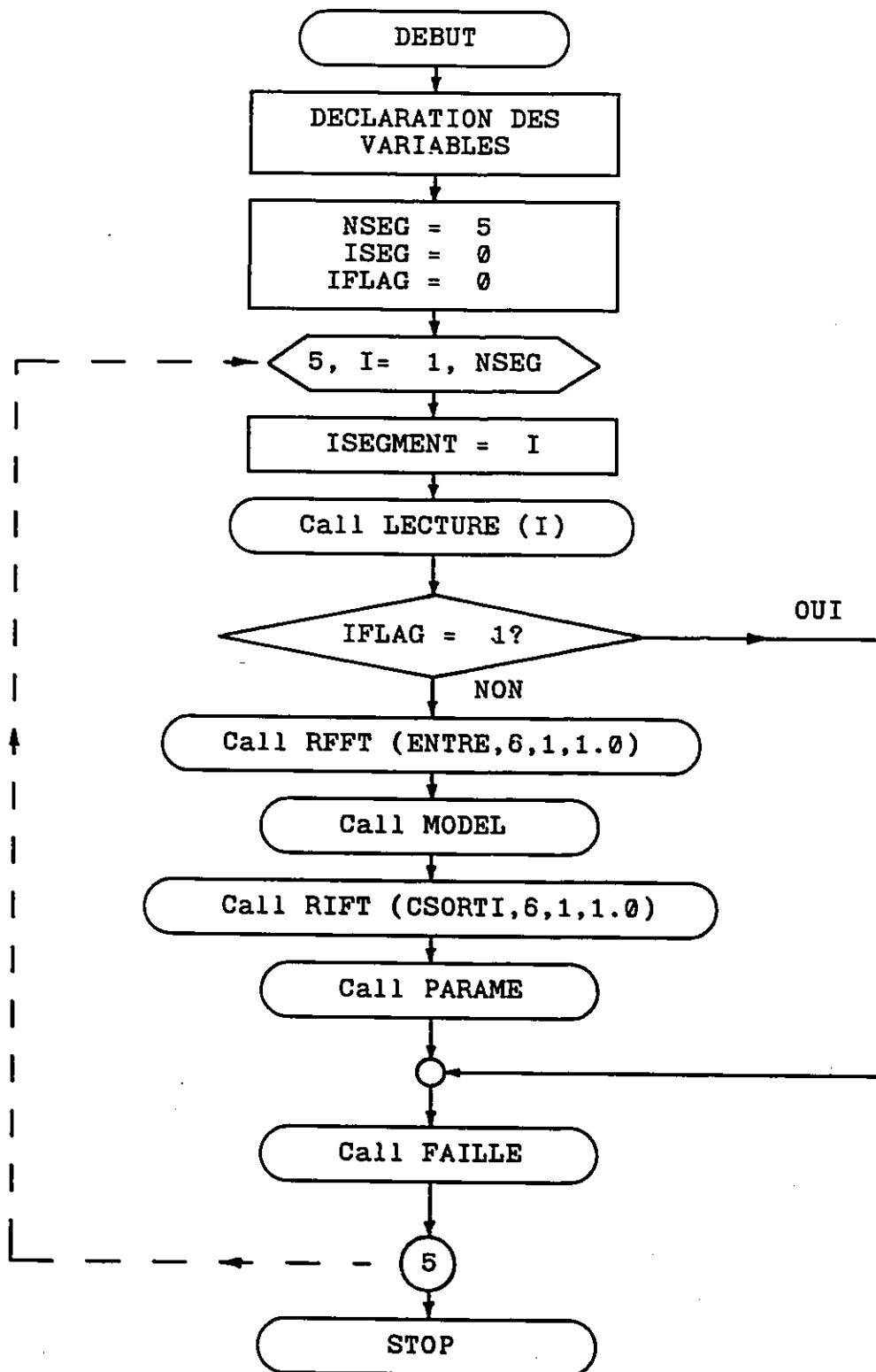
LE PROGRAMME DE DETECTION AUTOMATIQUE DES ANOMALIES

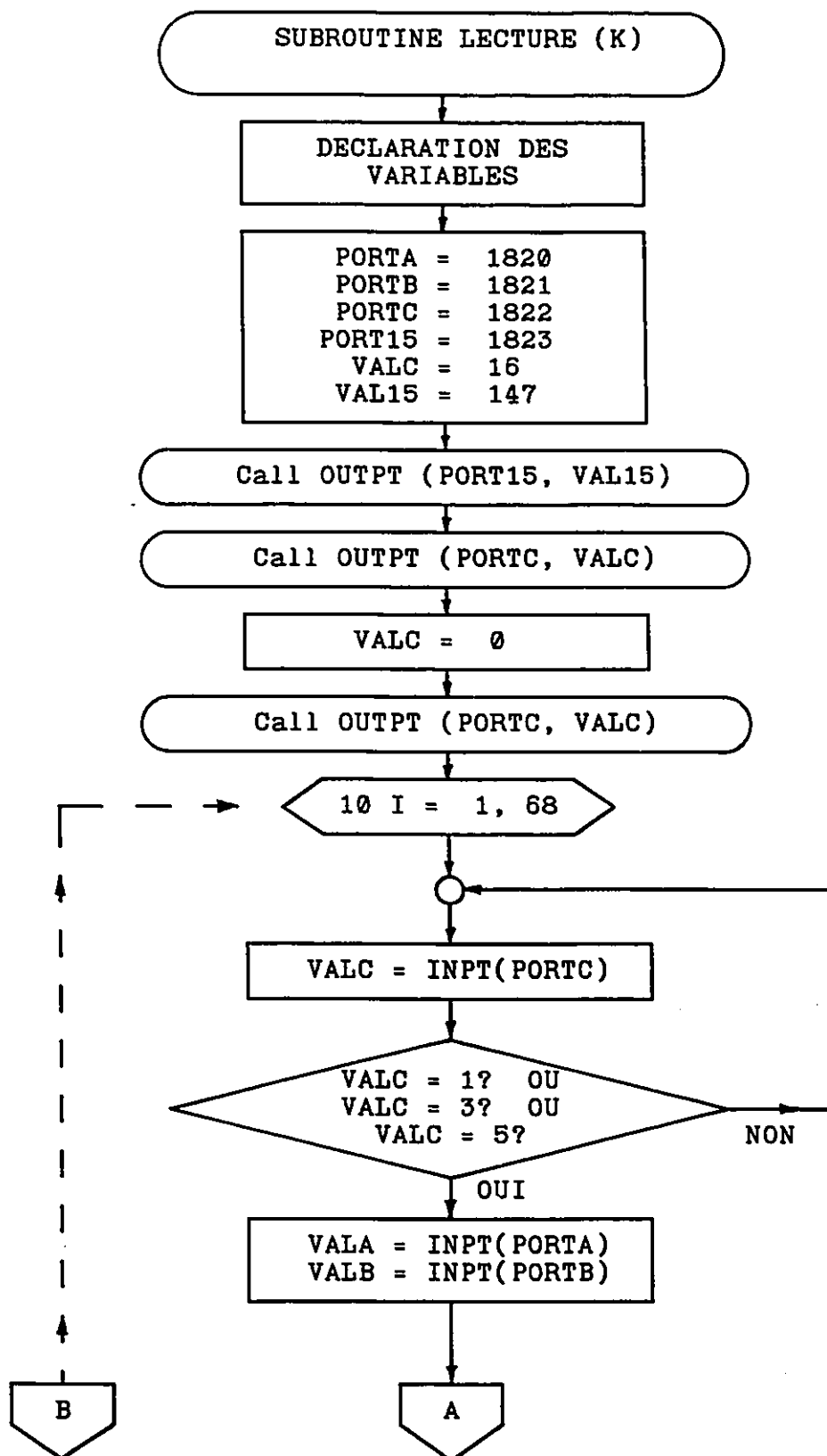
Ce programme a été écrit pour le Fortran de Microsoft version 3.2. Il fonctionne en temps réel, en parallèle avec le montage expérimental afin de détecter la présence d'anomalies dans le système réel. Dans un premier temps, le programme lit les données directement du montage expérimental. Puis il teste si une défaillance est présente. Si oui, alors le programme essaiera de déterminer dans quel groupe la défaillance se trouve. Cette annexe comprend l'organigramme et le programme au complet.

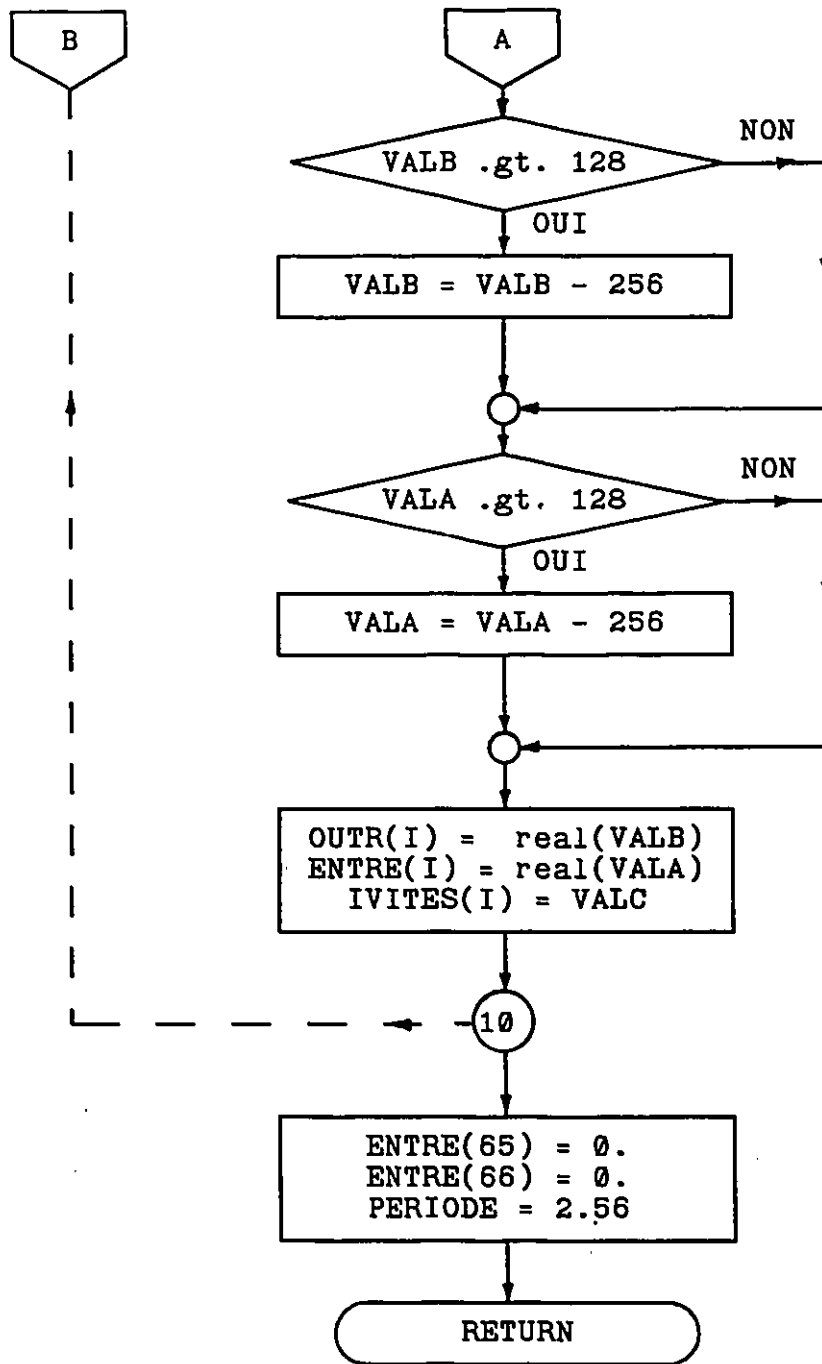
Note: Les sous-routines RFFT et IFFT viennent du logiciel de 87FFT de la compagnie MICROWAY, donc l'organigramme et les programmes ne sont pas disponibles pour ces routines.

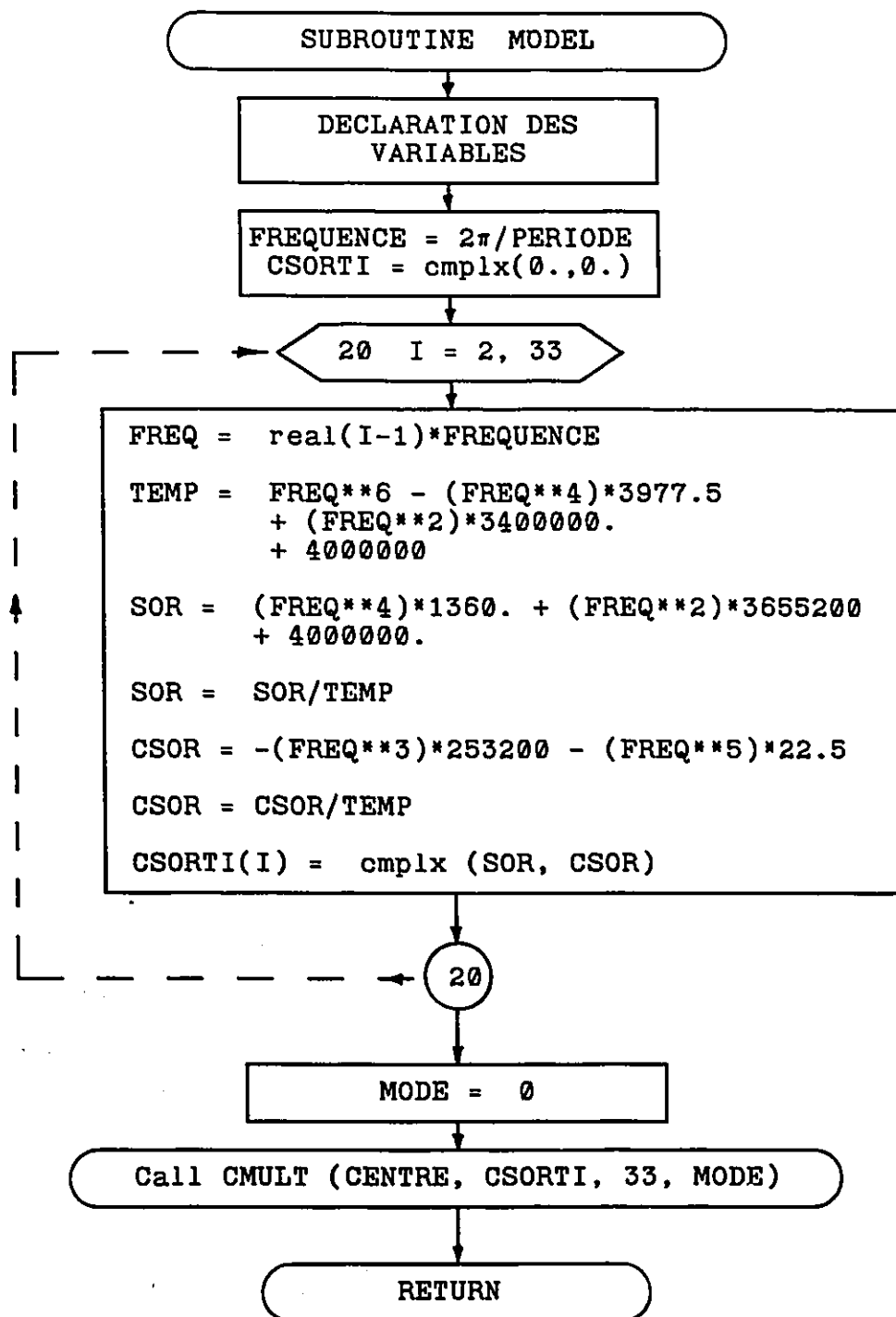
Les sous-routines OUTPT et INPT sont écrites en langage Assembleur et non en Fortran.

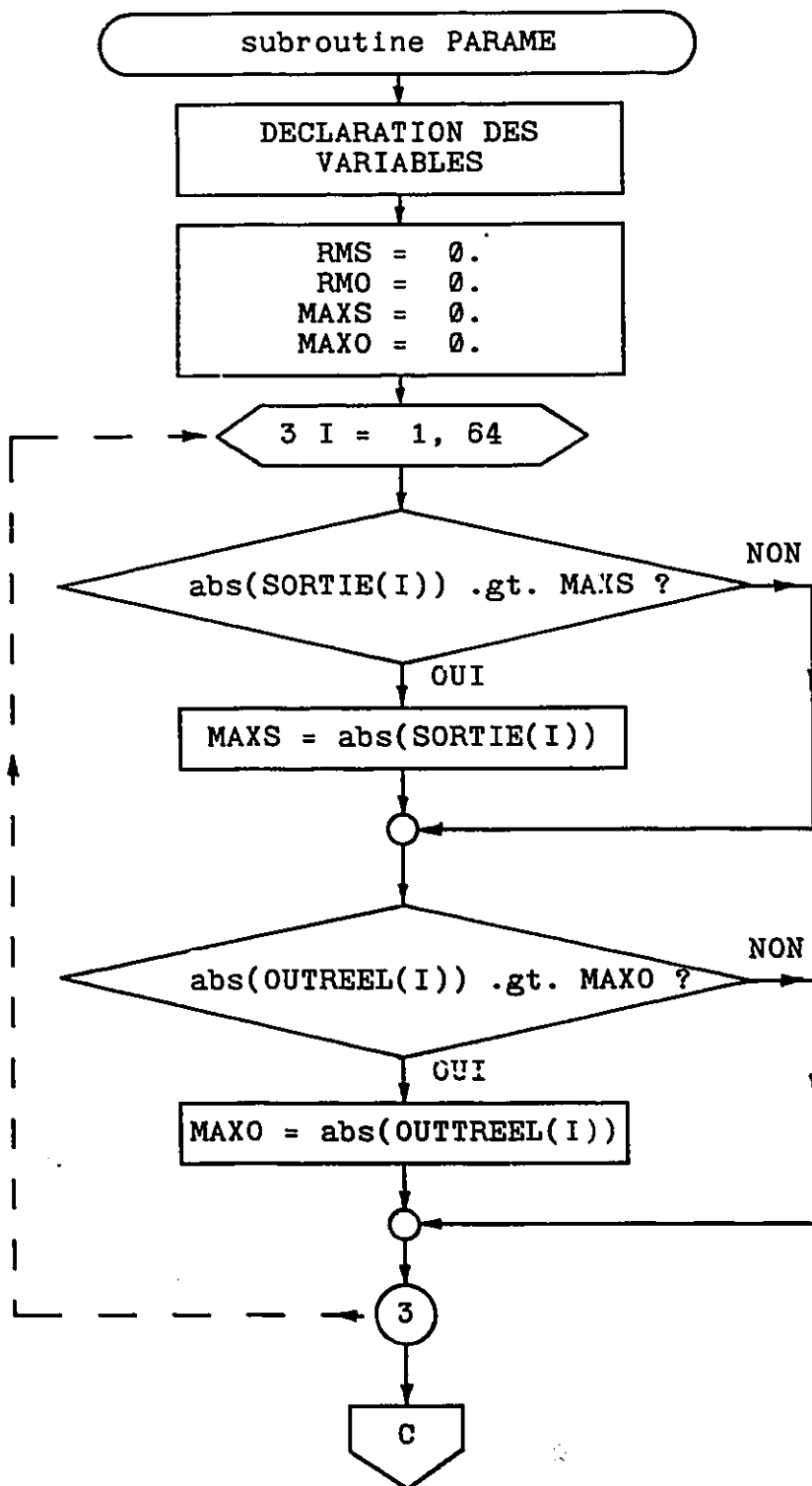
LE PROGRAMME DE DETECTION AUTOMATIQUE DES ANOMALIES

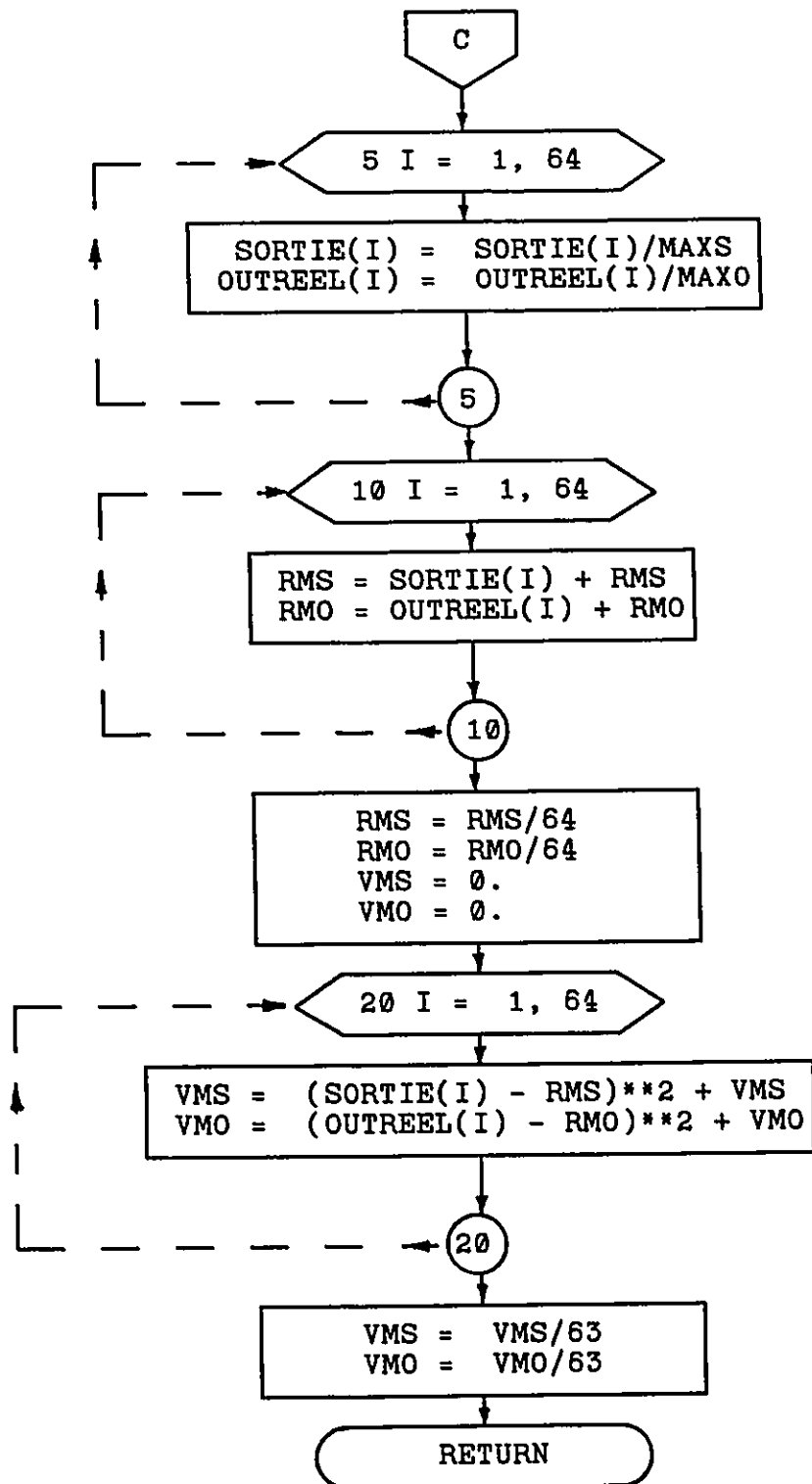


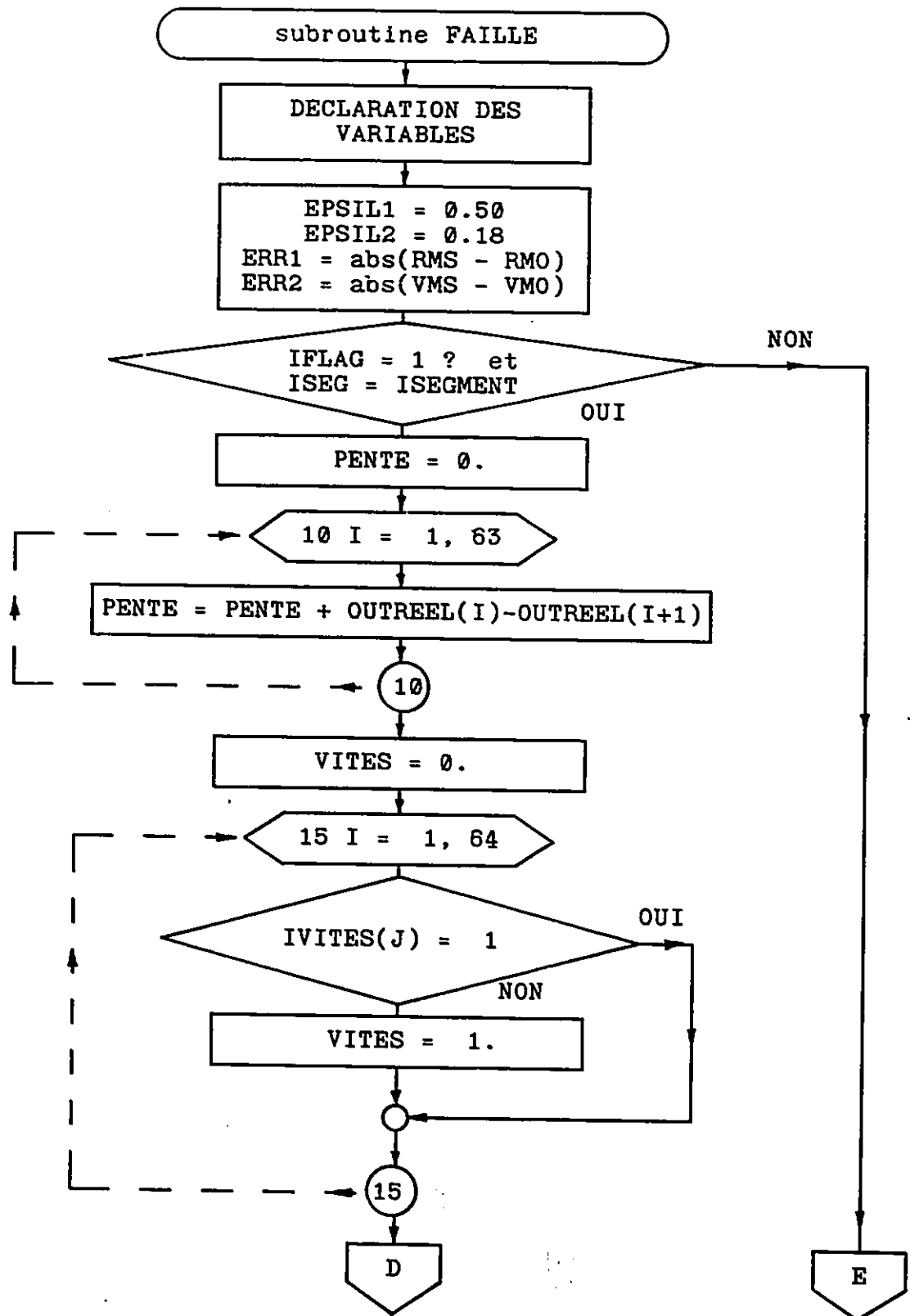


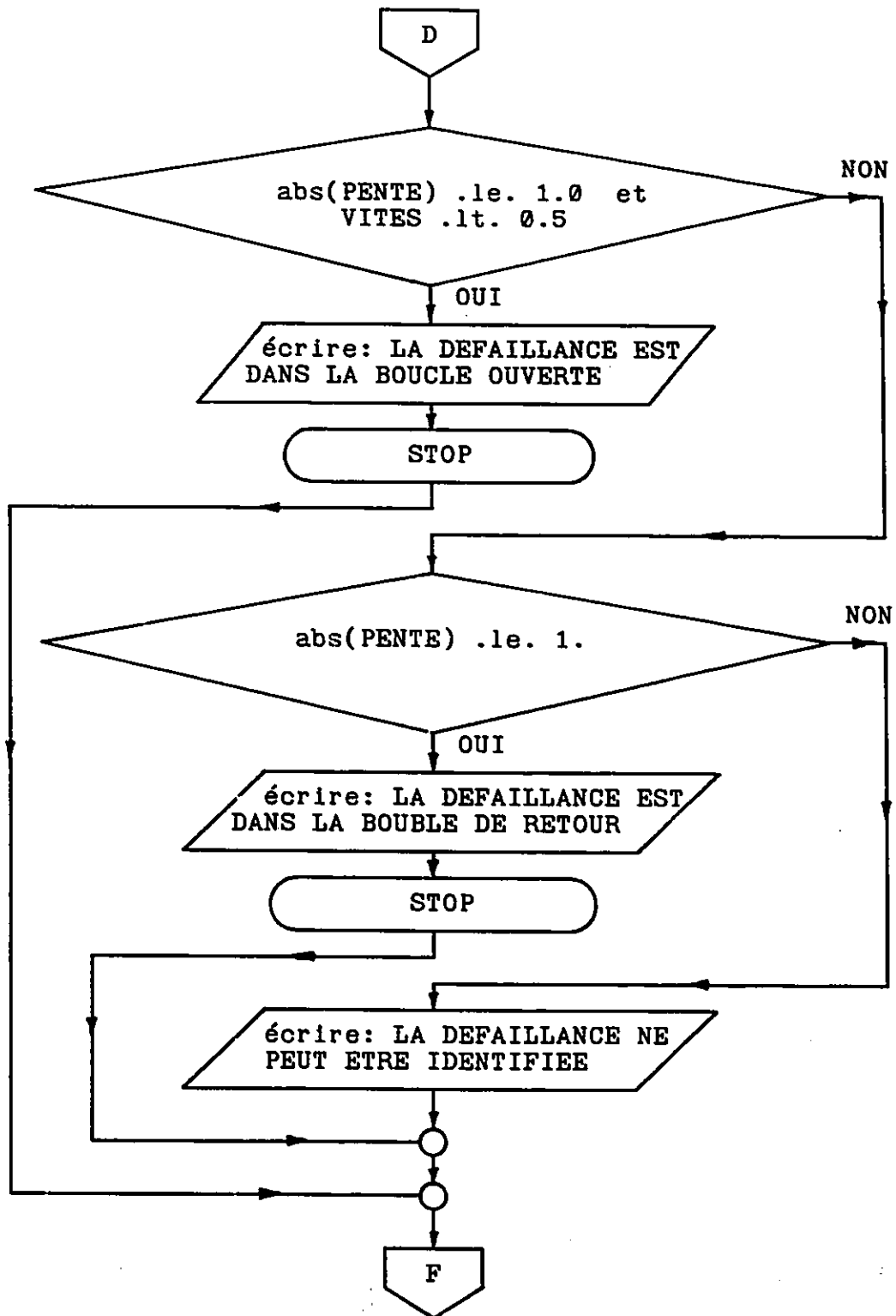


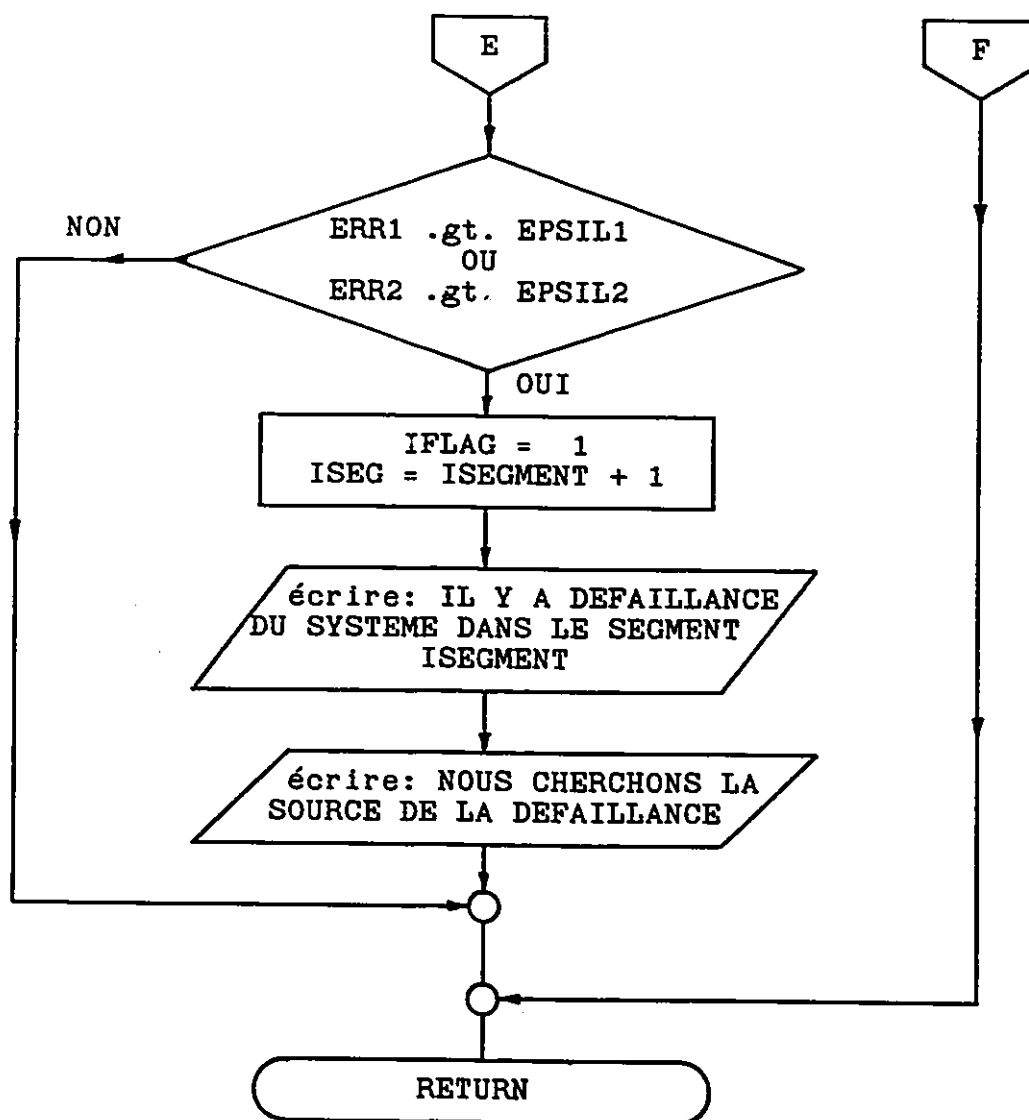












C - Le programme principal.

```
integer*2 IVITES(64)
real*4 ENTRE(66), SORTIE(66), OUTREEL(66)
complex*8 CENTRE(33), CSORTI(33)
equivalence (CENTRE,ENTRE)
equivalence (CSORTI,SORTIE)
common /OUTPUT/ SORTIE
common /INPUT/ ENTRE, PERIODE
common /REELE/ OUTREEL, ISEGMENT, VITES, IFLAG, ISEG, IVITES
```

```
NSEQ = 5
ISEG = 0
IFLAG = 0
```

C - Pour chaque segment vérifions s'il y a défaillance.

```
do 5 I = 1, NSEG
  ISEGMENT = I
  call LECTURE (I)
```

C - Si une défaillance a été détecté dans le segment précédent,
 C on peut aller directement à la sous-routine FAILLE pour
 C déterminer dans quel groupe la défaillance a lieu.

```
if(IFLAG .ne. 1) then
  call RFFT (ENTRE,6,1,1.0)
  call MODEL
  call RIFT (CSORTI,6,1,1.0)
  call PARAME
endif
call FAILLE
```

5 continue

```
stop
end
```

```
*****
```

```
subroutine LECTURE (K)
```

```
*****
```

```
*
*-----*
* Cette sous-routine lit les valeurs d'entrées et de sorties *
* du système de commande aux adresses appropriées. Pour un *
* segment, 64 valeurs sont lues et seront plus tard analysées. *
*-----*
*
```

```
integer*2 PORTA,VALA,PORTB,PORTC,PORT15,VALB,VALC,VAL15
integer*2 IVITES(64)
real*4 ENTRE(66), OUTREEL(66), OUTR(68)
complex*8 CENTRE(33)
equivalence (CENTRE,ENTRE)
common /INPUT/ ENTRE, PERIODE
common /REELE/ OUTREEL,ISEGMENT,VITES,IFLAG,ISEG,IVITES
```

```
PORTA = 1808 + 12
PORTB = 1808 + 13
PORTC = 1808 + 14
PORT15 = 1808 + 15
VALC = 16
VAL15 = 147
```

C - Envoyons le signal pour partir le moteur

```
call OUTPT(PORT15,VAL15)
call OUTPT(PORTC,VALC)
VALC = 0
call OUTPT(PORTC,VALC)
```

C - Lisons le signal d'entrée, de sortie et si le moteur tourne.

```
do 10 I = 1, 64
20 VALC = INPT(PORTC)
   if ((VALC .eq. 1).or.(VALC .eq. 3).or.(VALC .eq. 5))then
       VALA = INPT(PORTA)
       VALB = INPT(PORTB)
       if (VALB .gt. 128) VALB = VALB - 256
       if (VALA .gt. 128) VALA = VALA - 256
       OUTR(I) = real (VALB)
       ENTRE(I) = real (VALA)
       IVITES (I) = VALC
   else
       goto 20
   endif
10 continue
```

```
ENTRE(65) = 0.  
ENTRE(66) = 0.  
PERIODE = 2.45
```

```
return  
end
```

```
*****
      subroutine MODEL
*****
```

```
*
*-----*
*
* - Etant donnée la fonction de transfert du système dans le
*   domaine fréquence, cette sous-routine calcule la sortie
*   du modèle dans le domaine fréquence pour obtenir le vecteur
*   "SORTIE"
*
*-----*
```

```

      real*4 ENTRE(66), SORTIE(66)
      complex*8 CENTRE(33), CSORTI(33)
      equivalence (CENTRE,ENTRE)
      equivalence (CSORTI,SORTIE)
      common /OUTPUT/ SORTIE
      common /INPUT/ ENTRE, PERIODE

      FREQUENCE = 2*3.1416/PERIODE
      CSORTI (1) = CMPLX(0.,0.)
      do 20 I=2, 33
          1  FREQ = real(I-1)*FREQUENCE
             TEMP =  FREQ**6-((FREQ**4)*3977.5)+((FREQ**2)*3400000.)
                + 4000000.
             SOR =  ((FREQ**4)*1360.)+(FREQ**2)*3655200.+4000000.
             SOR =  SOR / TEMP
             CSOR =  - ((FREQ**3)*253200.) - ((FREQ**5)*22.5)
             CSOR =  CSOR / TEMP
             CSORTI (I) = CMPLX(SOR,CSOR)
20      continue

      MODE = 0
      call CMULT(CENTRE,CSORTI,33,MODE)

      return
      end
```

```
*****
```

```
subroutine PARAME
```

```
*****
```

```

*
*-----*
*
* Cette sous-routine calcule les paramètres de comparaison
* entre la sortie réel du système et la sortie modèle du
* system. Ces paramètres sont:
*
*
*           1. La moyenne échantillonnée.
*
*                 $M = 1/N \sum_{i=1, \dots, N} X_i$ 
*
*           2. La variance échantillonnée.
*
*                 $V = 1/N \sum (X_i - M)^2$ 
*
*-----*
*

```

```

integer*2 IVITES(64)
real*4 SORTIE(66), OUTREEL(66)
complex*8 CSORTI(33), COUTREEL(33)
equivalence (CSORTI,SORTIE)
equivalence (COUTREEL,OUTREEL)
common /OUTPUT/ SORTIE
common /REELE/ OUTREEL, ISEGMENT, VITES, IFLAG, ISEG, IVITES
common /VALEUR/ RMS, RMO, VMS, VMO

```

```
RMS = 0.
```

```
RMO = 0.
```

```
C 1 - Trouvons le maximum.
```

```
MAXS = 0.
```

```
MAXO = 0.
```

```
do 3 I = 1, 64
```

```
  if ( abs(SORTIE(I)) .gt. MAXS ) then
```

```
    MAXS = abs(SORTIE(I))
```

```
  endif
```

```
  if ( abs(OUTREEL(I)) .gt. MAXO ) then
```

```
    MAXO = abs(OUTREEL(I))
```

```
  endif
```

```
3 continue
```

C 2 - Normalisons

```
do 5 I = 1, 64
  SORTIE(I) = SORTIE(I)/MAXS
  OUTREEL(I) = OUTREEL(I)/MAXO
5 continue
```

C 3 - Calculons la moyenne échantillonnée.

```
do 10 I = 1, 64
  RMS = SORTIE (I) + RMS
  RMO = OUTREEL (I) + RMO
10 continue
RMS = RMS/64.
RMO = RMO/64.
```

C 4 - Calculons la variance échantillonnée.

```
VMS = 0.
VMO = 0.
do 20 I = 1, 64
  VMS = (SORTIE (I) - RMS)**2 + VMS
  VMO = (OUTREEL (I) - RMO)**2 + VMO
20 continue
VMS = VMS/63
VMO = VMO/63

return
end
```

```
*****
      subroutine FAILLE
*****
```

```
*
*-----*
* Cette sous-routine détermine dans un premier temps, pour le *
* segment "i", s'il y a défaillance. Si une défaillance est *
* détectée dans le segment "i", on va lire le segment "i+1" *
* qu'on utilisera afin de trouver dans quel groupes la *
* défaillance a lieu. *
* *
*-----*
```

```
integer*2 IVITES(64)
real*4 OUTREEL(66)
complex*8 COUTREEL(33)
equivalence (COUTREEL,OUTREEL)
common /REELE/ OUTREEL,ISEGMENT,VITES,IFLAG,ISEG, IVITES
common /VALEUR/ RMS, RMO, VMS, VMO
```

```
EPSIL1 = 0.50
EPSIL2 = 0.18
```

C 1 - Calculons l'erreur sur la moyenne et la variance.

```
ERR1 = ABS(RMS - RMO)
ERR2 = ABS(VMS - VMO)
```

```
if ((IFLAG .EQ. 1) .AND. (ISEG .EQ. ISEGMENT)) THEN
```

C 2 - Calculons la pente à partir du signal de sortie du système

```
      PENTE = 0.
      do 10 I = 1, 63
10      PENTE = PENTE + (OUTREEL(I) - OUTREEL(I+1))
      continue
```

C 3 - Vérifions si le moteur tourne.

```
      VITES = 0.
      do 15 J = 1, 64
          if (IVITES(J) .ne. 1) then
              VITES = 1.
          endif
15      continue
```

C 4 - Si la pente est zéro et le moteur ne tourne pas alors la
 C défaillance est dans la boucle ouverte.
 C Si la pente est zero et le moteur tourne toujours alors la
 C défaillance est dans la boucle de retour.
 C Si la pente n'est pas zero et le moteur tourne alors on ne
 C peut identifier d'ou la défaillance provient.

```

      if ((ABS(PENTE) .LE. 1.0) .and. (VITES .LT. 0.5)) then
        write (*,3000)
        STOP
      else
        if (ABS(PENTE) .LE. 1.) then
          write (*,4000)
          STOP
        else
          write (*,5000)
        endif
      endif
    else
      if ((ERR1 .GT. EPSIL1) .OR. (ERR2 .GT. EPSIL2)) THEN
        IFLAG = 1
        ISEG = ISEGMENT+1
        write(*,1000) ISEGMENT
        write(*,2000)
      endif
    endif

1000 format(//,13x,'< Il y a défaillance du système dans le ',
1      'segment', I2,' >',//)
2000 format(//,16x,'Nous cherchons la source de la défaillance')
3000 format(//,15x,'--> La défaillance est dans la boucle ',
2      'ouverte <--')
4000 format(//,13x,'--> La défaillance est dans la boucle de',
3      ' retour <--')
5000 format(//,16x,'--> La défaillance ne peut etre identifié',
4      ' <--')

      return
      end

```

```

*
*-----*
*                Sous-routine INPT                *
*-----*
* Cette sous-routine en langage assembleur nous permet, dans *
* un programme en fortran (Microsoft v3.2), de lire des valeurs *
* aux portes de sortie parallèles du IBM PC XT. La routine *
* peut être invoquée comme suit:                  *
*-----*
* EX:  RESULTAT = INPT (PORT#)                   *
*-----*
* Dans cet exemple on lit la valeur à la porte numéro PORT# *
* et on retourne cette valeur dans la variable RESULTAT *
*-----*

```

```

PUBLIC  INPT                                ; Permet à d'autre programme
                                           ; d'avoir accès a la routine
                                           ; INPT

A_TEXT SEGMENT
ASSUME CS:A_TEXT
INPT   PROC FAR
                                           ; On doit avoir une procédu-
                                           ; re far pour faire un
                                           ; RET far à la fin.
        PUSH  BP                            ; On veut préserver la
                                           ; valeur originale de BP
        MOV   BP,S                          ; Pointer BP au haut de la
                                           ; pile (stack)
        PUSH  DS                            ; On veut préserver les
                                           ; registres des segments
        LDS   BX,DWORD PTR [BP]+6          ; Lire le paramètre sur la
                                           ; pile
        MOV   DX, [BX]
        IN    AL, DX                        ; Effectuer l'opération
                                           ; d'entrée
        MOV   AH, 00                        ; Laisser la valeur dans AX
        POP   DS
        POP   BP
        RET   4
INPT   ENDP
A_TEXT ENDS
END

```

```

*
*-----*
*           Sous-routine OUTPT
*
* Cette sous-routine en langage assembler nous permet, dans
* un programme en fortran (Microsoft v3.2), d'envoyer des
* valeurs au porte de sortie parallele du IBM PC XT. La
* routine peut etre appellee par un CALL ordinaire.
*
* EX:  call  OUTPT (PORT#, VALEUR)
*
* Dans cet exemple la variable VALEUR est envoye a la porte
* numero PORT#.
*
*-----*

```

```

PUBLIC  OUTPT                ; Permet a d'autre programme
                               ; d'avoir acces a la routine
                               ; OUTPT

A_TEXT  SEGMENT 'CODE'
ASSUME  CS:A_TEXT
OUTPT   PROC  FAR            ; On doit avoir une procedu-
                               ; re FAR pour faire un RET
                               ; far a la fin
                               ; on veut preserver la
                               ; valeur originale de BP
    PUSH BP                  ; Pointe BP au haut de la
                               ; pile (stack)
    MOV BP, SP              ; On veut preserver les
                               ; registres des segments
    PUSH DS                  ; Lire l'adresse du premier
                               ; parametre
    LDS BX, DWORD PTR [BP]+0AH ; Lire le parametre a
                               ; l'adresse indiquee.
    MOV DX, [BX]            ; Lire l'adresse du second
                               ; parametre
    LDS BX, DWORD PTR [BP]+6  ; Lire le second parametre a
                               ; l'adresse indique
    MOV AX, [BX]            ; Effectuer la sortie
    OUT DX, AL
    POP DS
    POP BP
    RET 8                    ; Effacer les parametres sur
                               ; la pile (stack) et retour

OUTPT   ENDP
A_TEXT  ENDS
END

```

ANNEXE E
SCHEMAS DE CONSTRUCTION
DU MONTAGE EXPERIMENTAL

Description des pièces (Fig. E.1).

No.	DESCRIPTION	MATERIEL	QUANTITE
1	Base	Aluminium	1
2	Base du lecteur optique	Aluminium	1
3	Support pour le roulement à billes	Aluminium	1
4	Support du moteur	Aluminium	1
5	Support du disque du lecteur optique	Aluminium	1
6	Support de la charge	Acier	1
7	Coupleur	Acier	1
8	Arbre d'entraînement	Acier	1
9	Charge	Aluminium	1
10	Disque du lecteur optique	Aluminium	1
11	Lecteur optique Type TIL138		1
12	Ensemble Moteur, Engrenage et Tachy- mètre de type Pittman GMT 9413-2 (12 VDC)		1

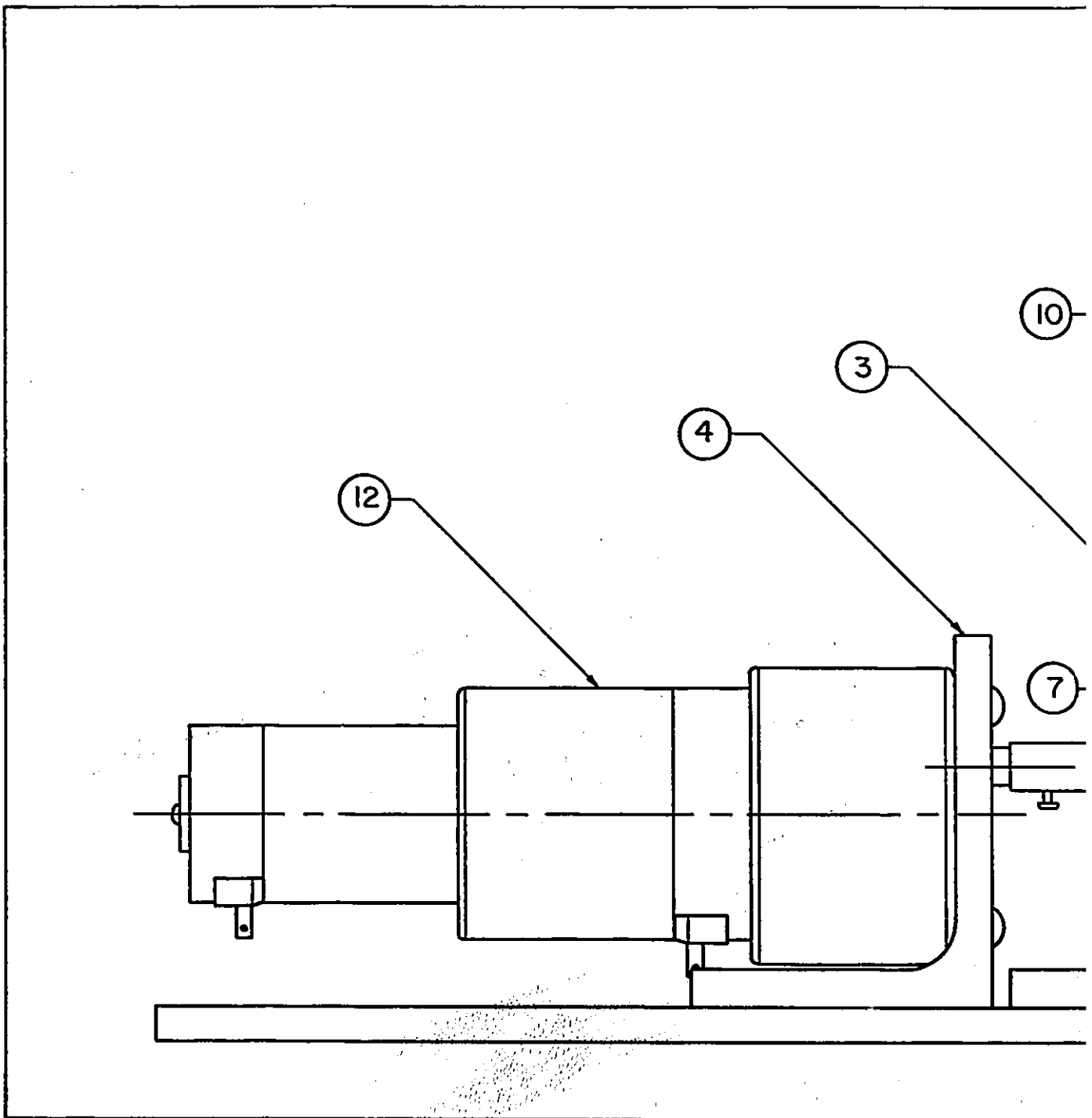


Figure E.1: Schéma d'assemblage du montage expérimental.

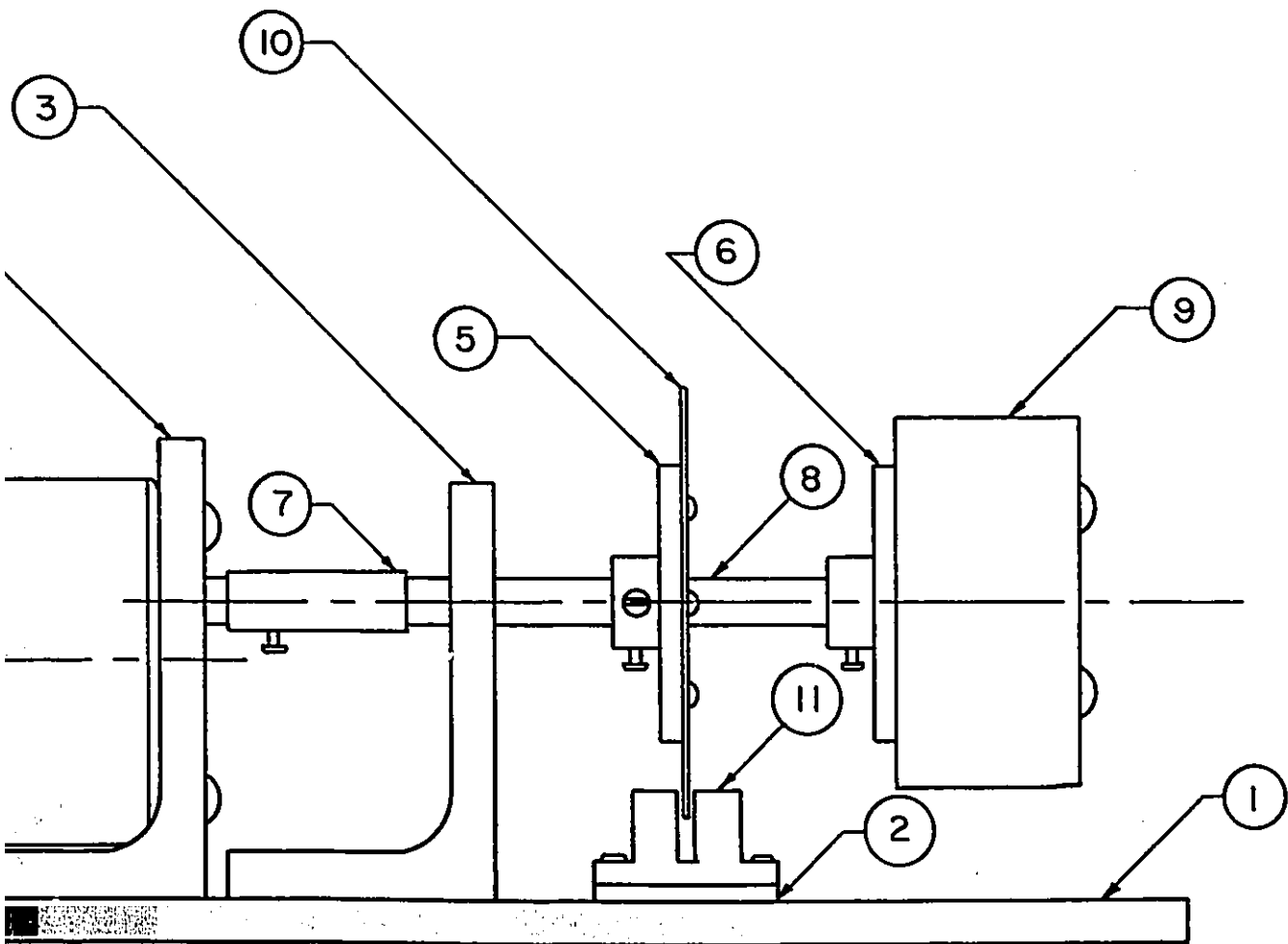
MONTAGE EXPERIMENTAL

JOHANNE SEVIGNY

27/04/87

DIMENSIONS
Pouces

ECHELLE: 1:1



ntal.

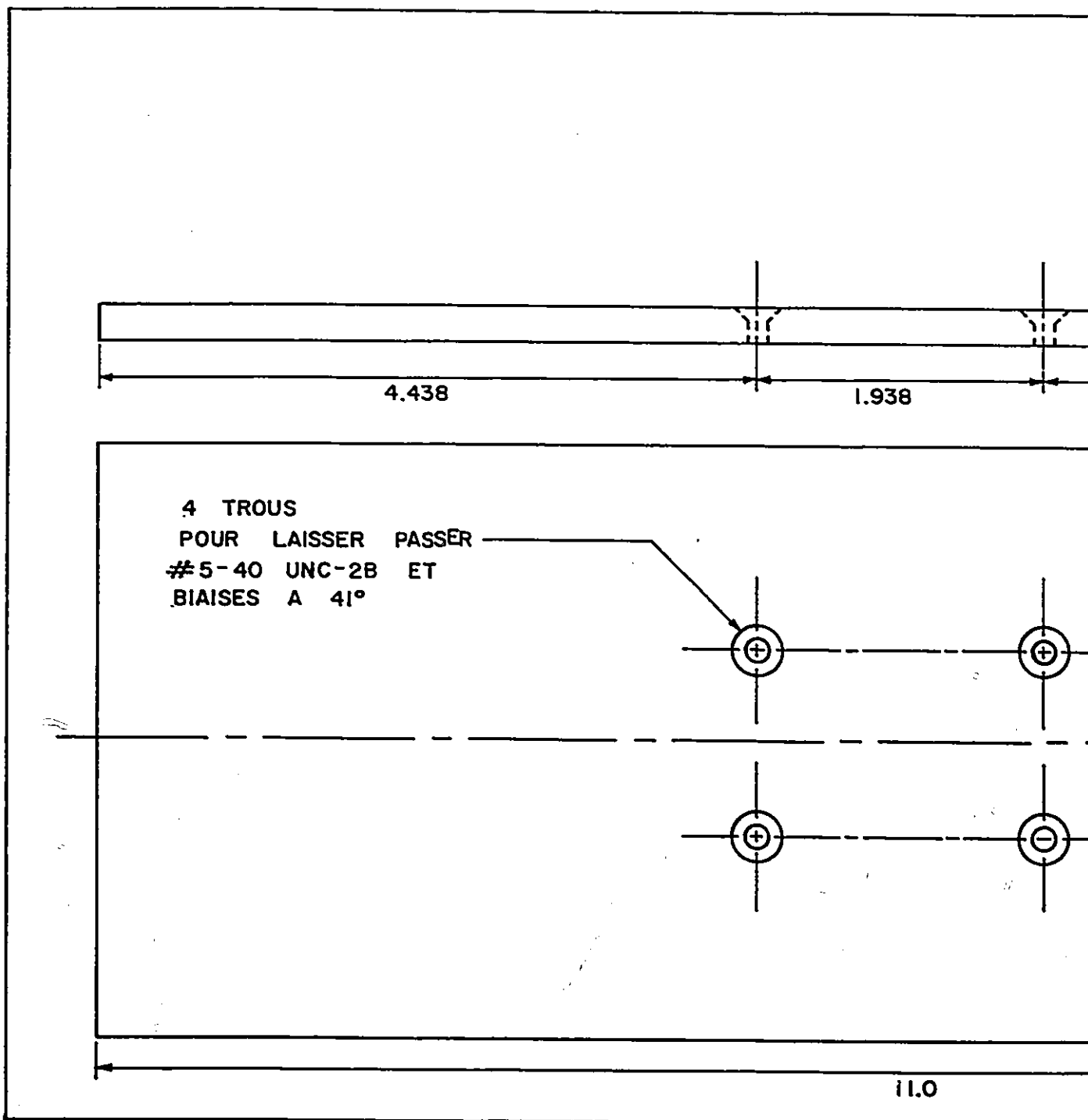
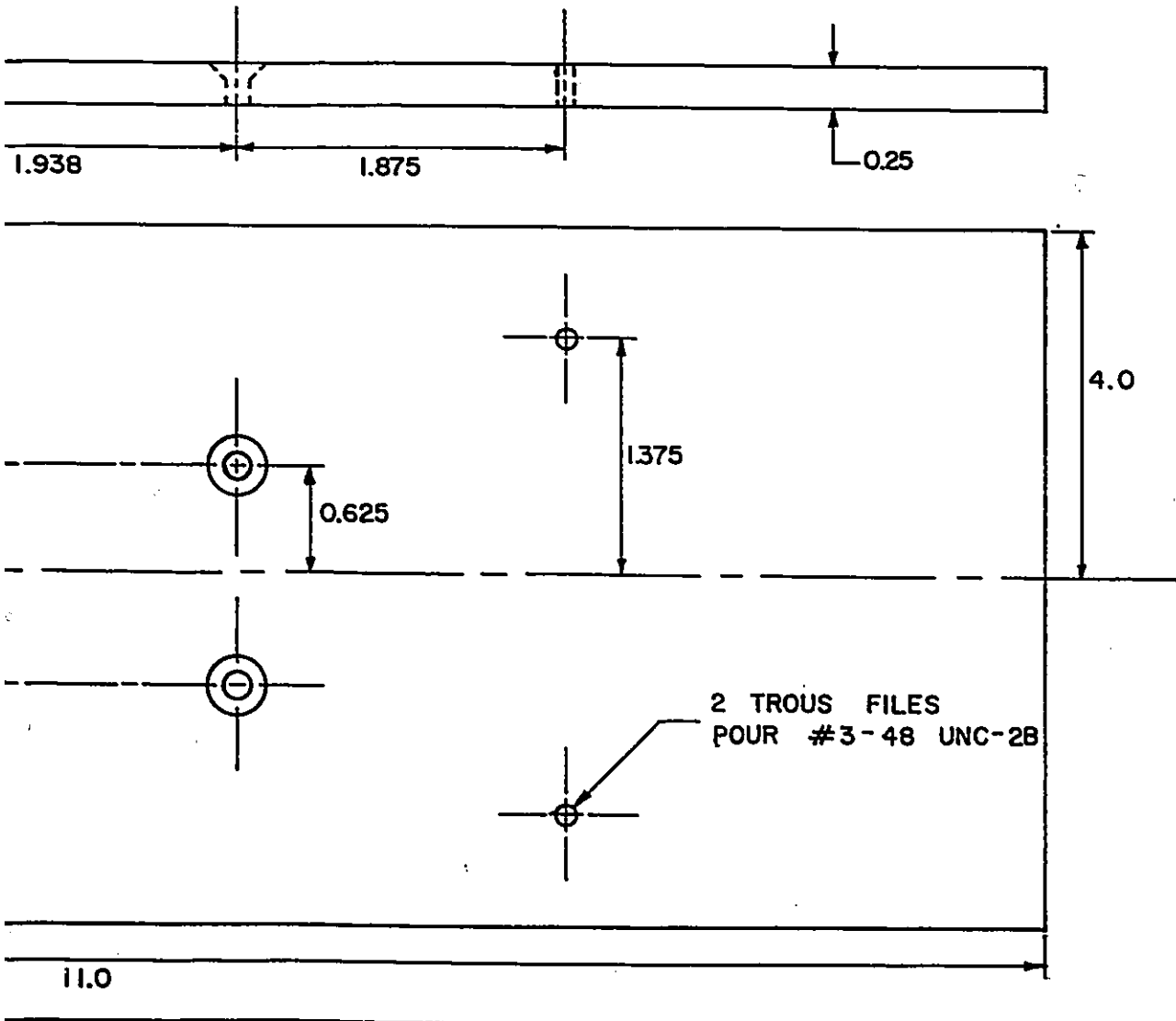


Figure E.2: Schéma de construction de la base.

TOLERANCE 0.1 sur XX.X 0.01 sur XX.XX 0.001 sur XX.XXX	(1) BASE	
	JOHANNE SEVIGNY	27/04/87
	DIMENSIONS Pouces	ECHELLE: 1:1



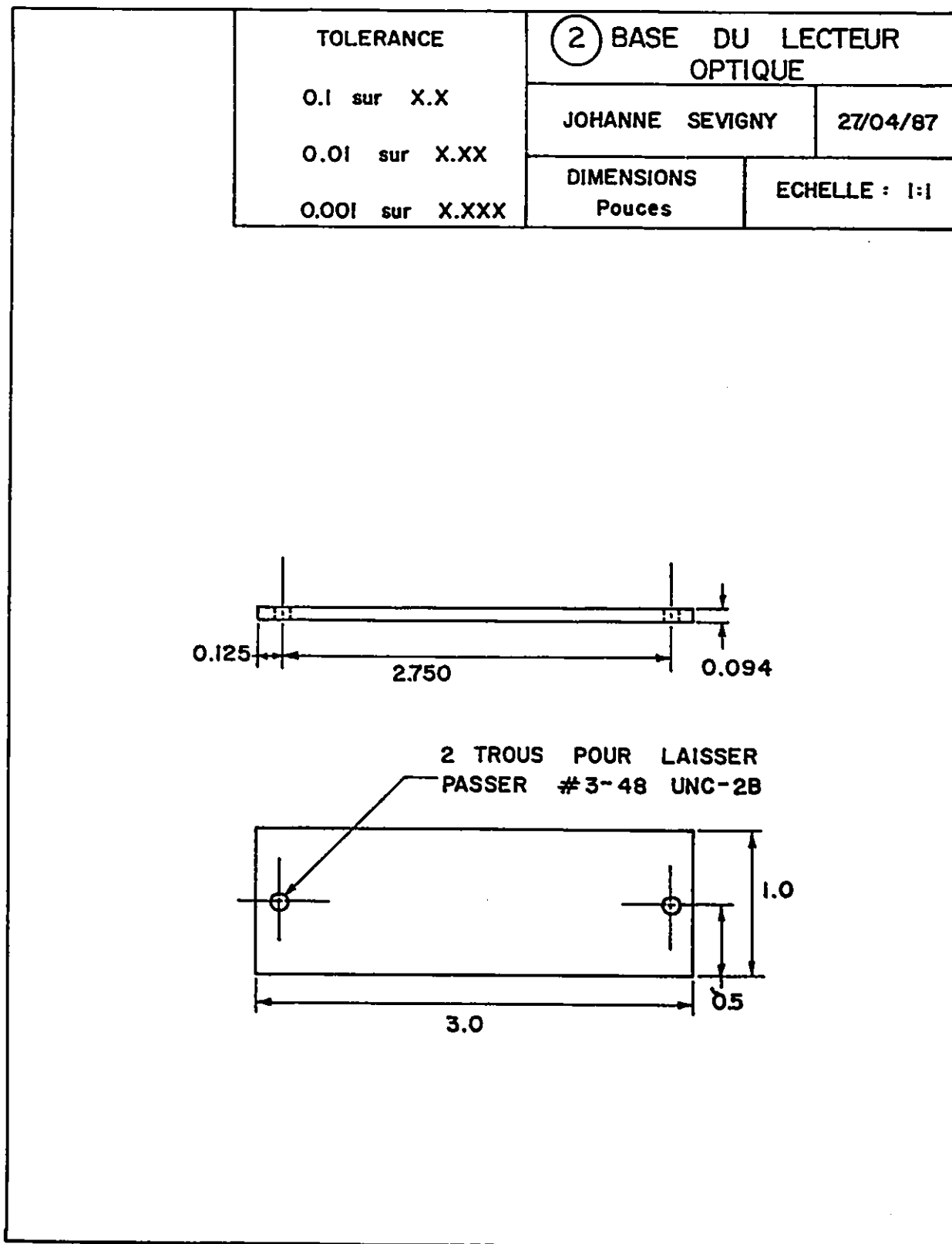


Figure E.3: Schéma de construction de la base du lecteur optique.

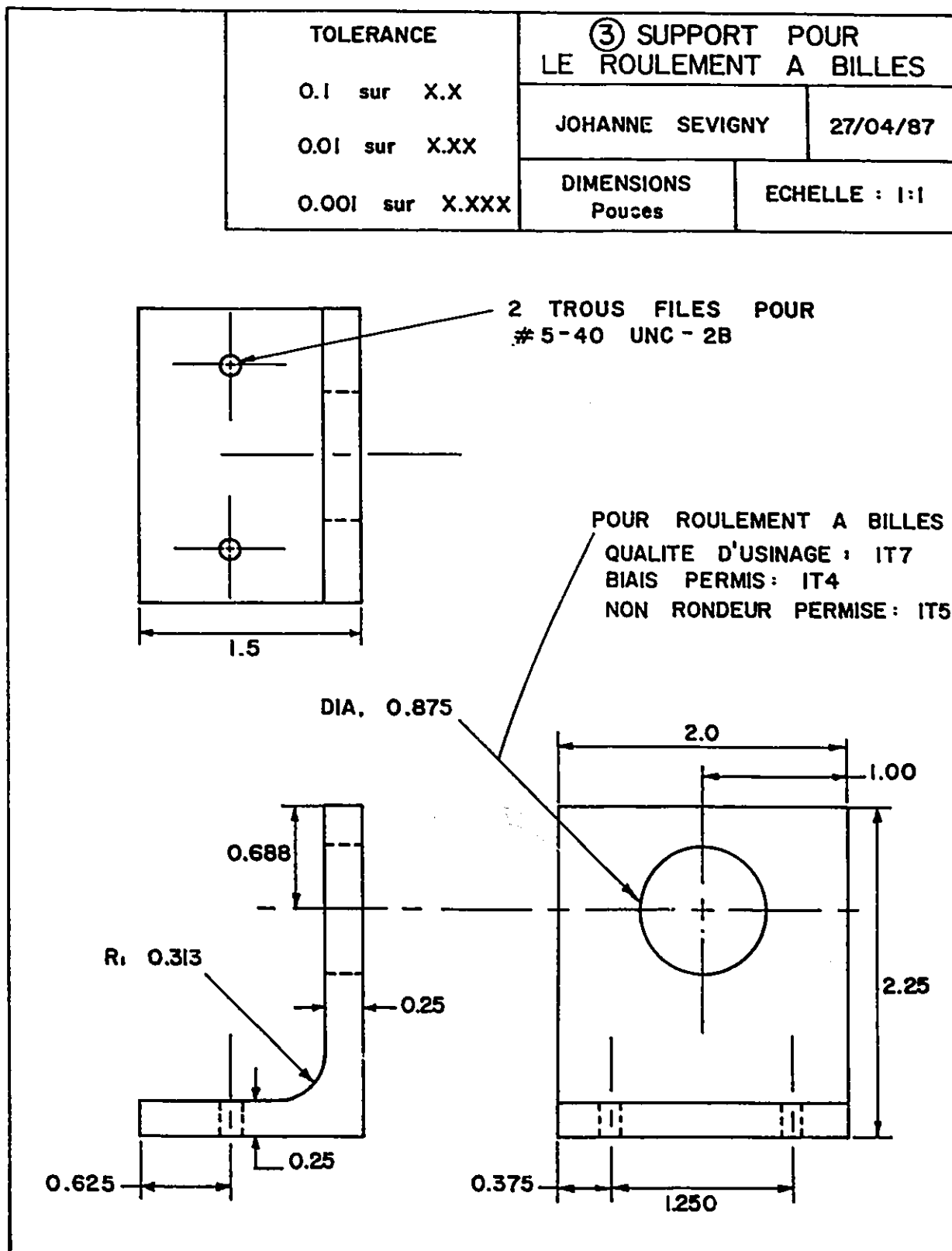


Figure E.4: Schéma de construction du support pour le roulement à billes.

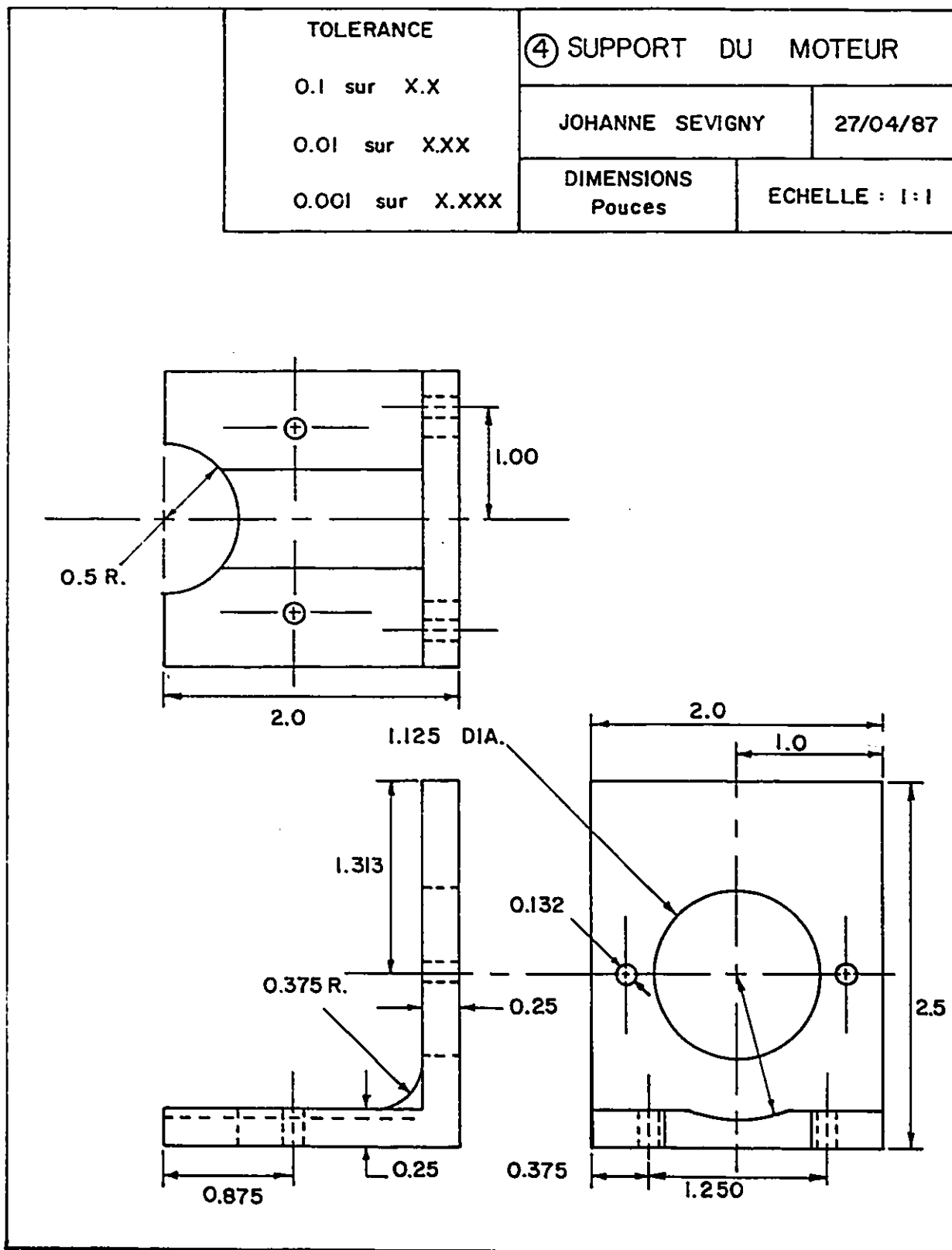


Figure E.5: Schéma de construction du support du moteur.

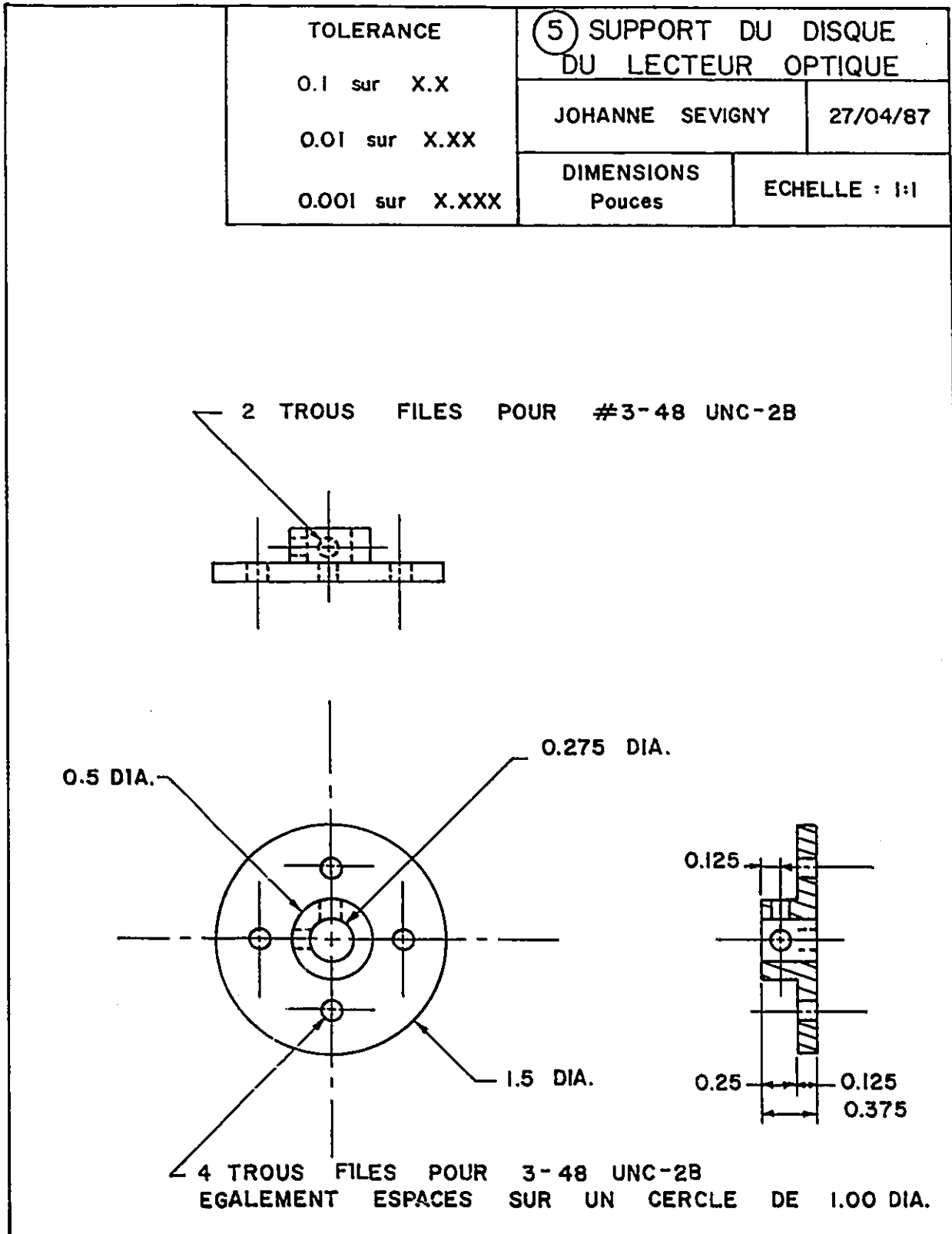


Figure E.6: Schéma de construction du support du disque du lecteur optique.

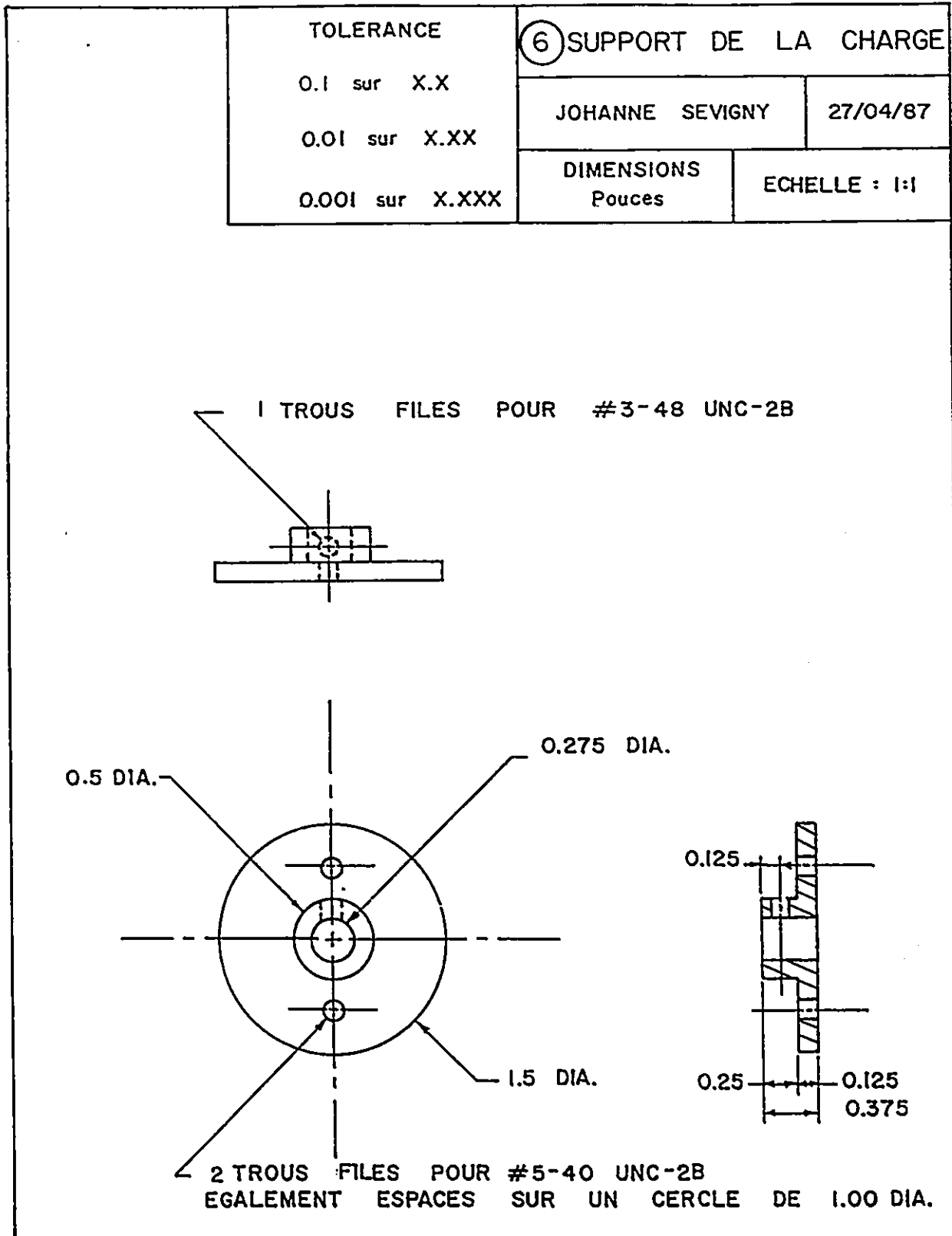


Figure E.7: Schéma de construction du support de la charge.

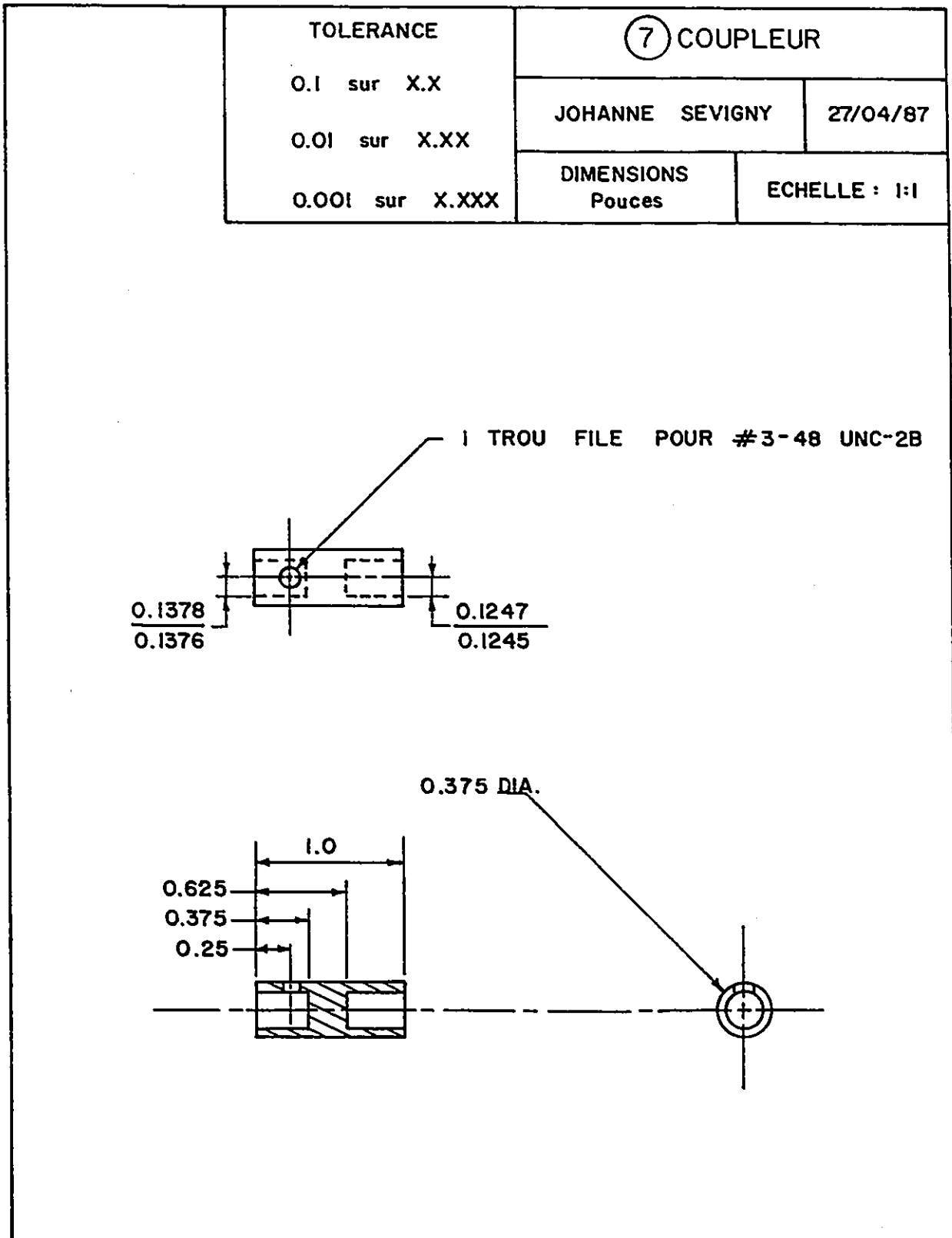


Figure E.8: Schéma de construction du coupleur.

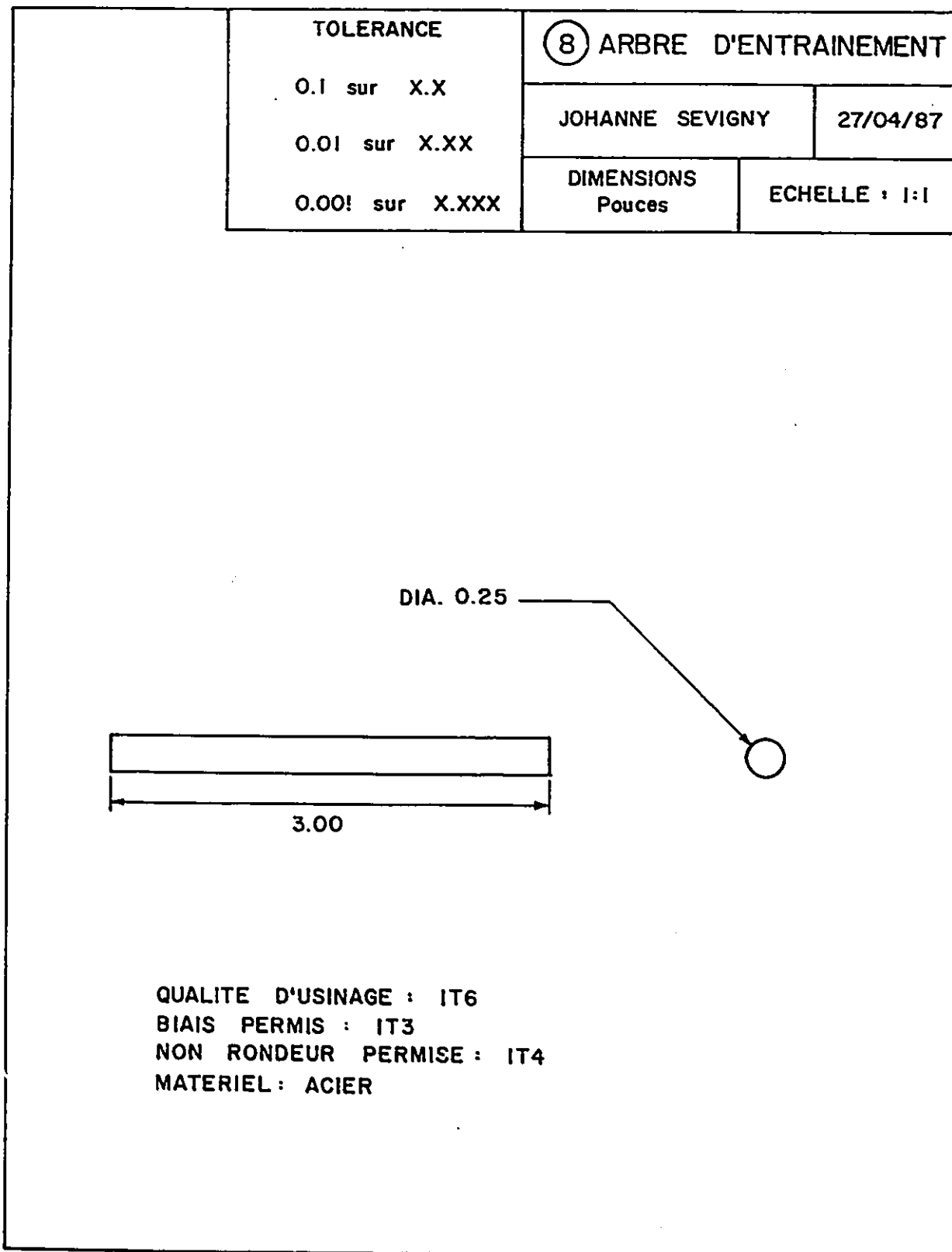


Figure E.9: Schéma de construction de l'arbre d'entraînement.

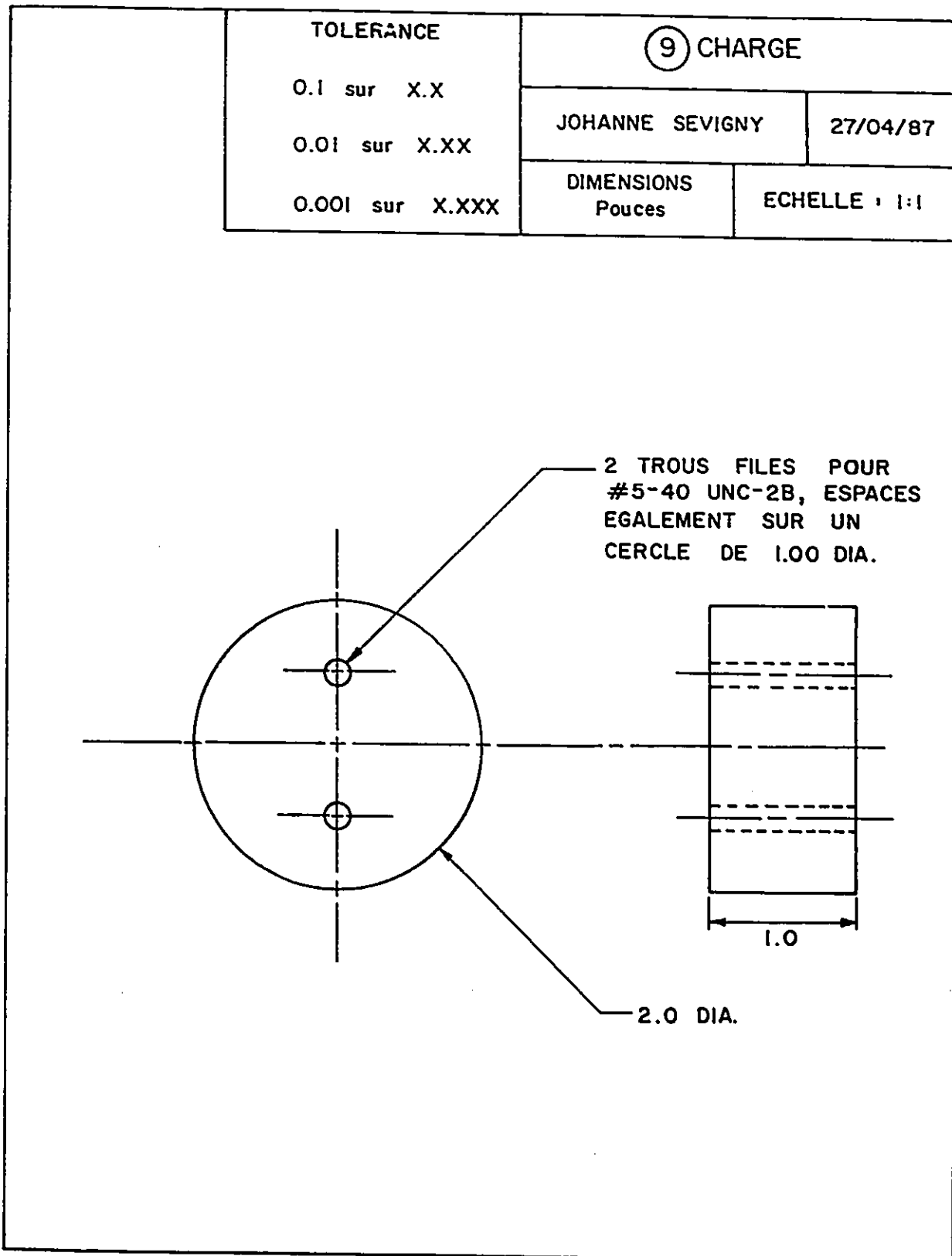


Figure E.10: Schéma de construction de la charge.

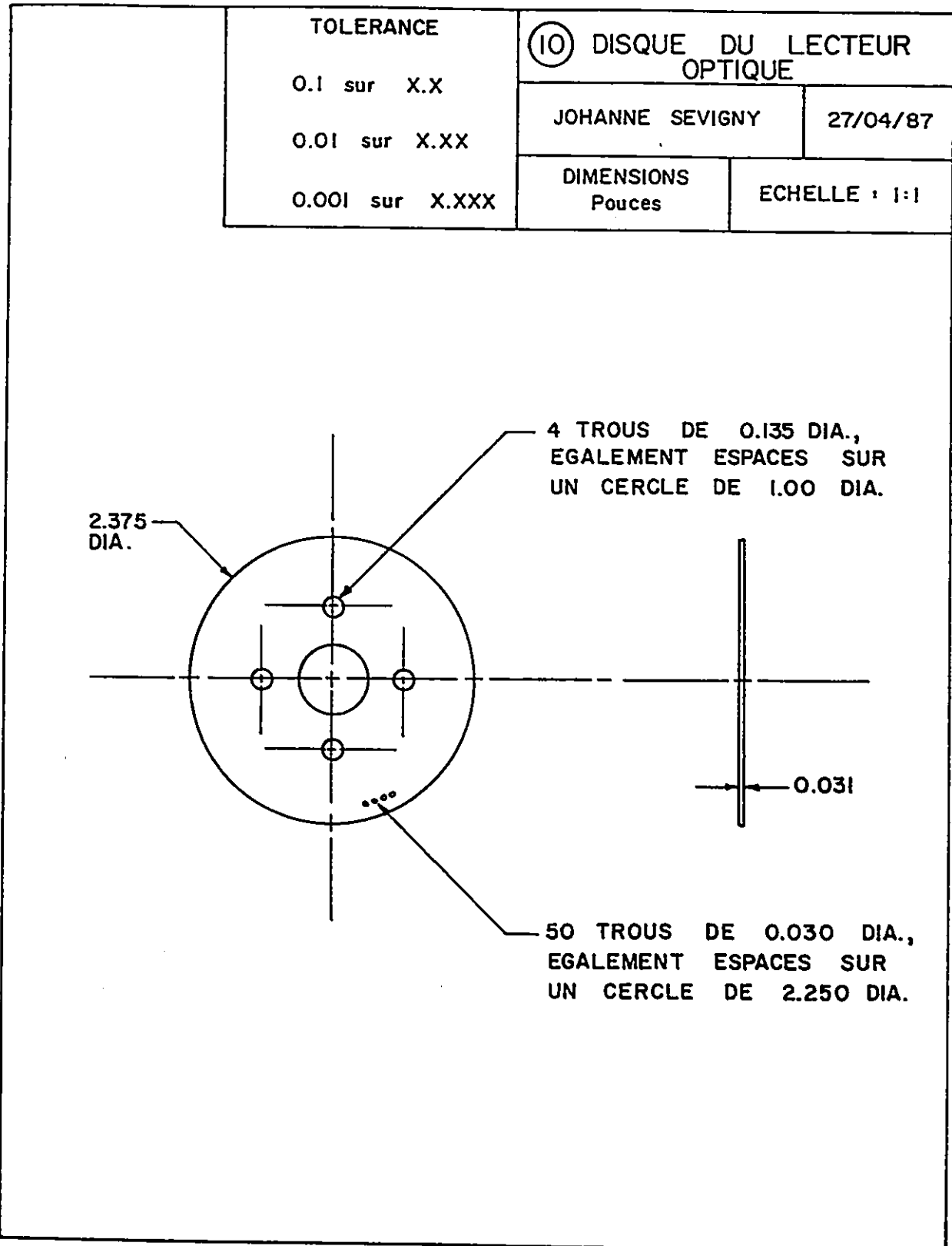


Figure E.11: Schéma de construction du disque du lecteur optique

BIBLIOGRAPHIE

LIVRES GENERALS

- [1] DORF, Richard C. Modern Control Systems, 3^e éd., Addison-Wesly, 1983.
- [2] KUO, Benjamin C. Digital Control Systems, New-York, Holt, Rinehart and Winston, Inc., 1980.
- [3] PRUDHOMME, R. Automatique, Paris, Masson et Cie, 1974, tome 2.
- [4] OGATA, Katsuhiko. Modern Control Engineering, Englewood Cliffs, Prentice-Hall, Inc., 1970.
- [5] OGATA, Katsuhiko. Discrete-Time Control Systems, Englewood Cliffs, Prentice-Hall, Inc., 1987.
- [6] HENLEY, Ernest J., et Hiromitsu KUMAMOTO. Reliability Engineering And Risk Assessment, Englewood Cliffs, Prentice-Hall, Inc., 1981.
- [7] DHILLON, Balbir S. Systems Reliability, Maintainability and Managament, New-York, Petrocelli Books, 1983.
- [8] ENGELBERGER, Joseph F. Robotics in Practice: Management and Applications of Industrials Robots, Londre, Kogan Page, 1980.
- [9] GROOVER, Mikell P. Automation, Production Systems and Computer Integrated Manufacturing, Englewood Cliffs, Prentice-Hall, Inc., 1987.
- [10] O'CONNOR, Patrick D.T. Practical Reliability Engineering, deuxième édition, John Wiley & Sons, 1985.

PUBLICATIONS GENERALES

- [11] MASON, Samuel J. "Feedback Theory: Some Properties of Signal Flow Graphs", Proceedings of the I.R.E., sept. 1953, p. 1144-1156.
- [12] BONNEY, Maurice C. et Y.F. YONG. Robot Safety: International Trends in Manufacturing Technology, IFS Ltd, Springer-Verlag, 1985, 295 pages.

ARBRE DE VERITE

- [13] LAPP, Steven A. et Gary J. POWERS. "Computer-Aided Synthesis of fault-trees", IEEE Transactions on Reliability, Avril 1977, p. 2-12.
- [14] CAMARDA, P. et al. "An Efficient Algorithm for Fault Tree Automatic Synthesis from the Reliability Graph", IEEE Transactions on Reliability, vol. R-27, no. 3, août 1978, p. 215-221.
- [15] POWERS, G.J. et F.C. TOMPKINS. "Fault-Tree Synthesis for Chemical processes", AICHE Journal, vol. 20, mars 1974, p. 376-387.
- [16] GALLUZZO, M. et P.K. ANDOW. "Failures in Control Systems", Reliability Engineering, vol. 7, 1984, p. 193-211.
- [17] KISS, Lidia. "From Fault-Tree to Fault-Identification", IEEE Transactions on Reliability, vol. R32, no. 5, decembre 1983, p. 422-425.

METHODE DE LA FMEA

- [18] BARBOUR, Gilbert L. "Failure Modes and Effects Analysis by Matrix Method" Proceedings of the Annual Reliability and Maintainability Symposium, 1977, p. 114-119.
- [19] LEGG, John M. "Computerized Approach for Matrix-Form FMEA", IEEE Transactions on Reliability, vol. R-27, octobre 1978, p. 254-257.

- [20] GODDARD, Peter L. et Richard W. DAVIS. "The Automated Advanced Matrix FMEA Technique", Proceedings of the Annual Reliability and Maintainability Symposium, 1985, p. 77-81.
- [21] CONLEY, Gregory A. "Digital System Diagnostic - Design/Evaluation", Proceedings of the Annual Reliability and Maintainability Symposium, 1980, p. 38-42.
- [22] HERRIN, Stéphanie A. "Maintainability Applications Using the Matrix FMEA Technique", IEEE Transactions on Reliability, vol. R-30, no. 3, août 1981, p. 212-217.

LA DETECTION AUTOMATIQUE DES ANOMALIES

- [23] KREUZE, Loyd J. "BIT Analysis and Design Reliability", Proceedings of the Annual Reliability and Maintainability Symposium, 1983, p. 328-332.
- [24] JOHNSON, Seldon A. "Practical Preventive Reliability Using Matrix Forms", IEEE Transactions on Reliability, vol. R-32, no. 5, décembre 1983, p. 417-421.
- [25] AKSENOVA, G.P. et al. "Diagnostic Systems for Multiprocessor Microcomputers", Traduit de Avtomatika i Telemekhanika, Moscou, no. 10, octobre 1984, p.150-157.
- [26] ZRELOVA, T.I. "Self-Diagnosis of Digital Systems Containing Units with Self-Checkable Built-in Testing Circuits", Traduit de Avtomatika i Telemekhanika, Moscou, no. 2, février 1984, p. 123-132.
- [27] SHESKIN, Theodore J. "Sequencing of Diagnostic Tests for Fault Isolation by Dynamic Programming", IEEE Transactions on Reliability, vol. R-27, no. 5, décembre 1978, p. 353-359.

- [28] ALY, Adel A. et Nael A. ELSAYEDALY. "An Efficient Algorithm for Optimal Design of Diagnostics", IEEE Transactions on Reliability, vol. R-32, no. 5, décembre 1983, p 426-432.
- [29] COHN, Martin et Gene OTT. "Design of Adaptive Procedures for Fault Detection and Isolation", IEEE Transactions on Reliability, vol. R-20, no. 1, février 1971, p 7-10.
- [30] PERANSKII, D.V. "Approach to the Problem of Reducing Diagnostic Information", traduit de Avtomatika i Telemekhanika, no. 3, mars 1984, p. 151-160.

LES SYSTEMES EXPERTS

- [31] CHANDRASEKARAN, P. et S. MITTAL. "Deep Versus Compiled Knowledge Approaches to Diagnostic Problem-Solving" International Journal of Man-Machines Studies, vol. 19, novembre 1983, p. 425-436.
- [32] FINK, P.K. et al. "A General Expert System Design for Diagnostic Problem Solving", IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-7, sept. 1985, p. 553-560.
- [33] NARAYANAN, Hari N. et N. VISWANADHAM. "A Methodology for Knowledge Acquisition and Reasoning in Failure Analysis of Systems", IEEE Trans. on Systems, Man and Cybernetics, vol. SMC-17, no. 2, mars/avril 1987, p. 274-288.
- [34] LAMB, D.E. et al. "Development of an Expert System for Fault Identification in a Commercial Scale Chemical Process", Proceedings of the 6th International Workshop on Expert System & Their Applications, Avignon, avril 1986, p. 1371-1382.
- [35] MOORE, R.L. et M.A. Kramer. "Expert Systems in On-Line Process Control", Proceedings of the Third International Conference on Chemical Process Control, janvier 1986, p. 12-17.
- [36] ATWOOD, M.E. et E.R. RADLINSKI. "Diagnostic System Architecture", Electrical Communication, vol. 60, no. 2, 1986, p. 174-179.

- [37] THANDASSERI, M. "Expert Systems Application for TXE4A Exchanges", Electrical Communication, vol. 60, no.2, 1986, p. 154-161.

LA FIABILITE DES LOGICIELS

- [38] SHOOMAN, Martin L. "Software Reliability: A Historical Perspective", IEEE Transactions on Reliability, vol. R-33, no. 1, avril 1984, p. 48-55.
- [39] TRACHTENBERG, Martin. "The Linear Reliability Model and Uniform Testing", IEEE Trans. on Reliability, vol. R-34, no. 1, avril 1985, p. 8-16.
- [40] ALDEMIR, Tunc. "Computer-Assisted Markov Failure Modeling of Process Control Systems", IEEE Trans. on Reliab., vol. R-36, no. 1, avril 1987, p. 133-144.
- [41] YAMADA, Shigeru et al. "s-Shaped Software Reliability Growth Models and Their Applications", IEEE Trans. on Reliab., vol. R-33, no. 4, octobre 1984, p. 289-292.
- [42] CROW, Larry H. et Nozer D. SINGPURWALLA. "An Empirically Developed Fourier Series Model for Describing Software Failures", IEEE Trans. on Reliab., vol. R-33, no. 2, juin 1984, p. 176-183.
- [43] SPREIJ, Peter. "Parameter Estimation for a Specific Software Reliability Model", IEEE Trans. on Reliab., vol. R-34, no. 4, octobre 1985, p. 323-328.
- [44] JOE, H. et N. REID. "On the Software Reliability Models of Jelinski-Moranda and Littlewood", IEEE Trans. on Reliab., vol. R-34, no. 3, août 1985, p. 216-218.
- [45] LITTLEWOOD, Bev. "Rational for a Modified Duane Model", IEEE Trans. on Reliab., vol. R-33, no. 2, juin 1984, p. 157-159.

AUTOMATION, CELLULE FLEXIBLE DE FABRICATION

- [46] TANIMOTO, Hiromu. "Factory Automation: An Automatic Assembly Line for the Manufacture of Printers", Computer, déc. 1984, p. 50-68.

TRAITEMENT DU SIGNAL NUMERIQUE

- [47] MAX, J. Méthodes et Techniques de Traitement du signal et Applications aux mesures physiques, 4ième édition, Paris, Masson, 1985.
- [48] OPPENHEIM, Alan V. et Ronald W. SCHAFER. Digital Signal Processing, Englewood Cliffs, Prentice-Hall, Inc., 1975.
- [49] RIFE, Douglas D. Fast Fourier Transform Utility Package, Reference Manual, Kingston, MICROWAY Inc., 1985.
- [50] BRIGHAM, Oran E. The Fast Fourier Transform, Englewood Cliffs, Prentice-Hall, Inc., 1974.

RECONNAISSANCE DES FORMES

- [51] FU, K.S. Applications of Pattern recognition, CRC Press Inc., 1982.
- [52] GONZALEZ, R.C. et J.T. TOU. Pattern Recognition Principles, Addison- Wesley, 1974.
- [53] PAU, L.F. "Diagnosis of equipment Failures by Pattern Recognition", IEEE Transactions on Reliability, vol. R-23, no. 5, Août 1974, p. 202-207.

PROGRAMMATION ASSEMBLEUR

- [54] MORGAN, C.L. et Mitchell WAITE. 8086/8088 16 Bite Microprocesseur Primer, BYTE/McGraw-Hill, 1982.

- [55] NORTON, Peter et John SOCHA, Peter Norton's Assembly Language Book for the IBM PC, New-York, Brady Book, Prentice-Hall, 1986.
- [56] Z8 Microcomputer Technical Manual, Zilog, 1984

LES PUBLICATIONS DE L'AUTEUR

- [57] SEVIGNY, Johanne et Dan NECSULESCU. "Méthode de Diagnostic des Robots Industriels: analyse portant sur la détection préventive des anomalies" 55^e Congrès de l'ACFAS, Ottawa, mai 1987.
- [58] D.S. NECSULESCU, A. FAHIM et Jchanne SEVIGNY. "Failure Analysis and Reliability Modelling of Industrial Robots", Proceedings of the IASTED International Conference 87, Reliability and Quality Control, Paris, France, Juin 1987.
- [59] SEVIGNY, Johanne et Dan NECSULESCU. "Diagnosis Method for Robot System", Proceedings of the IEEE MONTECH (Compint) Conference 87, Montréal, Novembre 1987.

PLANIFICATION DE TRAJECTOIRES

- [60] PAUL, R.P.. Robot Manipulators: Mathematics Programming and Control, MIT Press, Cambridge, MA, 1981.

SYSTEME DE COMMANDE RESISTANT AUX DEFAILLANCES

- [61] JOSHI, S.M.. "Design of Failure-Accommodating Multiloop LQG-Type Controllers", IEEE Trans. on Automatic Control, vol. AC-32, No. 8, Août 1987, p.740-741.

SYSTEME DE COMMANDE

- [62] LUH, j.y.s.. "Conventional Controller Design for Industrial Robots - A Tutorial", IEEE Trans. on Systems, Man and Cybernetics, vol. SMC-13, no. 3, Mai/Juin 1983.