

MATHIEU GOULET

**HEXAPODE : DÉVELOPPEMENT MÉCATRONIQUE
D'UN ROBOT MARCHEUR**

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de Maîtrise en génie mécanique
pour l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES ET DE GÉNIE
UNIVERSITÉ LAVAL
QUÉBEC

JUIN 2006

Résumé

Ce mémoire traite du développement et de la réalisation d'un robot mobile marcheur à six pattes : Hexapode. Tant la conception mécanique que celle du contrôleur utilise des principes biomimétiques, s'inspirant de la fourmi. Le développement de ce robot permet l'étude théorique et la concrétisation expérimentale de concepts menant à des démarches stables et efficaces. La notion de polygone de support et de retour permet de gérer adéquatement les paramètres de stabilité. Des critères de performance tels que la mobilité, l'autonomie et l'adaptabilité sont utilisés pour parvenir à générer les idées et les concepts derrière cette plate-forme. Les étapes préalables à la fabrication du robot ainsi que le prototype final, Hexapode, qui a été construit sont décrits. L'équilibrage statique du robot a été effectué afin de diminuer la consommation énergétique. La programmation hiérarchique qui se décompose en couche d'interface matérielle, de couche de coordination et de couche décisionnelle permet la gestion des déplacements, des séquences de fonctionnement et la prise de décision pour permettre au robot de mieux faire face à l'environnement dans lequel il évolue. Les applications où Hexapode se démarquerait sont celles qui requièrent la dextérité propre à l'exploration de terrain, ceci pourrait inclure des tâches spatiales ou terrestres.

Avant-propos

Ces travaux sont dédiés à l'avancement des technologies. Que ce soit dans les domaines de la mécanique, de l'électronique ou de la programmation. Les travaux sur le robot mobile marcheur Hexapode ont été réalisés dans l'optique d'améliorer les ponts entre les disciplines de la mécatronique. Il est important de faire un lien entre les robots et les vivants, le biomimétisme a été à la base de la réalisation conceptuelle et opérationnelle d'Hexapode.

Mes premiers remerciements s'adressent au professeur Clément Gosselin qui m'a permis de vivre cette expérience extraordinaire. Il m'a partagé sa passion pour la robotique, ce que je lui serai éternellement reconnaissant.

Je remercie aussi les professionnels de recherche du laboratoire Thierry Laliberté, Simon Foucault et Boris Mayer Saint-Onge pour leur support technique hors pair.

Je remercie aussi Yves Jean, Michel Dominique et André Chamberland pour leur aide au niveau matériel, algorithmique et électronique. Je remercie Lionel Birglen qui m'a fait réfléchir et m'a apporté de la lumière sur le sujet de la mécatronique.

Je remercie mon père Michel Goulet pour ses précieux conseils et ses corrections. Je remercie aussi ma conjointe Tanis Keiner ainsi que les autres membres du laboratoire de robotique de l'université Laval pour leur support moral.

Table des matières

Résumé	i
Avant-propos	ii
Table des matières	iii
Liste des tableaux	vii
Liste des figures	viii
Introduction	1
1 Route menant à Hexapode	5
1.1 Biomimétisme	6
1.2 La marche	7
1.2.1 Morphologie du marcheur	8
1.2.2 Topologie variable	10
1.2.3 Points de contact	10
1.2.4 Polygones	12
1.2.5 Stabilité	13
1.2.6 Type de marche	13
1.2.6.1 Marche dynamique	14
1.2.6.2 Marche quasi-statique	14
2 Évaluation des morphologies	15
2.1 Animaux	15
2.1.1 Vertébrés	16

2.1.1.1	Quadrupède	18
2.1.1.2	Bipèdes	20
2.1.2	Invertébrés	21
2.1.2.1	Arachnides	22
2.1.2.2	Insectes	22
2.1.2.3	Polynéoptères	23
2.1.2.4	Phasmes	23
2.1.2.5	Fourmis	24
2.2	Historique de la robotique des insectes	24
2.2.1	D.M. Wilson	24
2.2.2	K. Pearson	26
2.2.3	H. Cruse	26
2.2.4	MoBot Lab du MIT	27
2.2.5	Université de Bremen	27
2.2.6	Jet Propulsion Lab (Nasa)	28
2.2.7	Case Western Reserve University	28
2.2.8	Hexapode	29
3	Les premiers modèles	31
3.1	Description des critères de performance	31
3.1.1	Mobilité	32
3.1.2	Autonomie	33
3.1.3	Adaptabilité	34
3.2	La fourmi en détail	34
3.3	Première représentation robotisée d'Hexapode	35
3.3.1	Évaluation selon les critères de performance	38
3.4	Prototype à compensation utilisant un mécanisme déployable	39
3.4.1	Problème géométrique inverse d'une patte	42
3.4.2	Évaluation selon les critères de performance	46
4	Hexapode le robot	48
4.1	Prototypes	49
4.1.1	Hexapode	49
4.1.2	Conception d'une patte de validation	50
4.2	Mobilité	52
4.2.1	Architecture à trois degrés de liberté par patte	52
4.2.1.1	Résolution du problème géométrique inverse	55

4.2.2	Type d'actionneurs	59
4.3	Design mécanique	64
4.3.1	Patte de validation	64
4.3.2	Hexapode	65
4.3.2.1	Fabrication des membres	66
4.3.2.2	Fabrication du corps et assemblage	67
4.4	Autonomie	71
4.4.1	Les dimensions des membres	71
4.4.2	Les pertes énergétiques du système de motorisation	72
4.4.3	Les effets de la variation des angles ϕ et ψ	72
4.4.4	Équilibrage statique	73
4.4.4.1	Étude de chargement	76
4.5	Adaptabilité	85
4.5.1	Système de capteur	86
4.5.1.1	Orientation des membres des pattes	86
4.5.1.2	Chargement de l'extrémité de la patte	89
4.5.1.3	Autres capteurs	91
4.5.2	Microcontrôleur	91
4.5.3	Programmation hiérarchique	95
4.5.4	Couche d'interface matérielle	99
4.5.4.1	Générateur d'échelon PWM	99
4.5.4.2	Utilisation des compteurs	100
4.5.4.3	Problème géométrique inverse	100
4.5.4.4	Librairie math.h	101
4.5.4.5	Correction des consignes	101
4.5.5	Couche de coordination	103
4.5.5.1	Stabilité	104
4.5.5.2	Lois de Wilson	104
4.5.5.3	Paramètres de la marche	108
4.5.5.4	Trajectoire des pattes	109
4.5.6	Couche décisionnelle	111
4.5.6.1	Limitations	113
4.5.6.2	Remplacement de la couche décisionnelle.	114
5	Conclusion	118
5.1	Retour sur les éléments essentiels du mémoire	118
5.2	Application	122

5.2.1	Domaine spatial	123
5.2.2	Domaine terrestre	124
5.3	La fin ou le début des recherches?	125
Bibliographie		129
A Animaux		132
B CD-rom d'accompagnement		134
B.1	Contenu du disque d'accompagnement	134
C Programmation		137
C.1	Calcul du chargement	137
C.2	Architecture de la structure d'Hexapode	150
D Pièces		153
D.1	Composantes mécaniques	153
D.2	Composantes électroniques	167

Liste des tableaux

4.1	Limites articulaires	55
4.2	Paramètres HD	55
4.3	Masse des membres	76

Liste des figures

1.1	Modélisation du contact	11
2.1	Marcheur dynamique de Cornell	21
2.2	Asimo de Honda	21
2.3	Wendy de l'université Waseda	21
2.4	Araignée	25
2.5	Sauterelle	25
2.6	Phasme	25
2.7	Fourmi	25
2.8	Hannibal	30
2.9	Aramies	30
2.10	Lemur	30
2.11	Bill-Ant	30
3.1	Anatomie d'une fourmi	36
3.2	Schéma d'une patte	37
3.3	Première représentation robotisée d'Hexapode	38
3.4	Prototype utilisant la compensation avec des mécanismes déployables	39
3.5	Liaisons entre les membres	40
3.6	Représentation vectorielle d'une patte utilisant la compensation	41
3.7	Repère 1 (Plan X-Y)	42
3.8	Repère 1 (Plan X-Z)	42
4.1	Modèle CAO d'Hexapode	49

4.2	Prototype Final	50
4.3	Prototype de patte en Lego	51
4.4	Forme schématique d'une patte	56
4.5	Servomoteurs	61
4.6	Contrôle des servomoteurs RC	61
4.7	Présentation des composantes d'une patte.	68
4.8	Paliers lisses pour les pivots	69
4.9	Ressort	69
4.10	Description des composantes du corps.	70
4.11	Représentation des poses des cas de chargement	82
4.12	Mécanisme de mesure de la pression à l'intérieur d'une patte.	90
4.13	FSR	90
4.14	Valeurs de la conductance	90
4.15	Servopod avec microcontrôleur DSP56F807.	92
4.16	Diagramme des périphériques du microcontrôleur.	93
4.17	La programmation hiérarchique.	97
4.18	Légende des différents type de boîtes	98
4.19	Représentation visuelle des séquences.	105
4.20	Trajectoire de l'extrémité d'une patte.	110
4.21	Console de téléopération.	116
5.1	Hexapode, version finale.	123

Introduction

Rouler ou marcher ?

Il y a beaucoup de roues, mais bien peu de jambes dans le domaine des moyens de transport non vivants. La marche présente l'avantage indéniable de pouvoir produire des déplacements dans toutes directions où des points de contact le permettent. La succession de ces points permet de produire la trajectoire désirée. Cette façon de se déplacer repose sur des critères d'équilibre et de stabilité.

En contrepartie, le domaine de la mécanique emploie des roues pour produire des mouvements. Celles-ci n'ont aucun équivalent biologique, elles procurent des bénéfices qui ne peuvent être mis de côté, comme permettre de franchir de grandes distances avec une faible quantité d'énergie à fournir. Cependant, il y a des situations où la roue est moins pratique, notamment lorsqu'il y a un obstacle à franchir. En fait, il a été démontré qu'il est difficile de franchir un élément de relief ayant une hauteur de plus du quart du diamètre de la roue. Par contre, une patte le peut.

En comparant la mobilité d'un marcheur et celle d'une plate-forme comportant des roues cylindriques (une voiture par exemple), on s'aperçoit que la marche (sur un plan) permet d'effectuer deux translations latérales (selon les axes X et Y du plan) en même temps qu'une orientation (rotation autour de l'axe Z), tandis que les roues ne peuvent

effectuer qu'une seule translation (selon l'axe X) et qu'une orientation (rotation autour de l'axe Z). Ceci est un inconvénient de taille pour circuler dans un environnement encombré ou ayant un fort relief. C'est en observant les caractéristiques avantageuses de la marche que l'idée de développer un robot mobile marcheur a émergé.

Objectifs du projet

L'objectif principal du présent projet est d'explorer le domaine de la robotique mobile des marcheurs. Il s'agit du premier projet de recherche et de développement de ce genre dans le laboratoire de robotique de l'Université Laval.

Le développement d'un robot de ce type est un projet d'envergure, il a mené à la conception et la mise au point du robot «Hexapode». Les différentes étapes de ce projet ont inclus :

- la recherche sur les robots existants,
- l'élaboration et la validation de concepts,
- la conception du design mécanique et électronique du robot,
- la programmation du contrôleur embarqué,
- la mise au point du robot Hexapode,
- la vérification de l'atteinte des objectifs,
- l'élaboration de pistes de solution afin de laisser une référence pour continuer l'aventure du développement d'un robot mobile marcheur.

L'idée est de construire une plate-forme mobile permettant de procéder à l'étude théorique et expérimentale des concepts qu'implique une démarche stable, efficace et performante. Le robot idéal doit être le plus autonome, mobile et adaptable possible. La poursuite de ces objectifs a permis d'effectuer des recherches dans la direction de la mécatronique, soit l'ingénierie de combiner des notions de mécanique, d'électronique et d'informatique. Hexapode se veut une réplique, à un facteur d'échelle plus grande, d'un insecte commun, la fourmi, ce qui sera expliqué dans les prochaines sections et chapitres.

La création d'un robot de ce type équivaut à tenter de donner la vie à un objet. Pour

que la créature puisse vivre, il lui faut un squelette, des muscles, des sens, des nerfs, un système de coordination, un cerveau et une intelligence artificielle assez complexe pour générer des intentions. Dans un monde idéal, cette créature serait une parfaite symbiose d'éléments inanimés qui, mis ensemble, formeraient un tout suffisamment complet pour pouvoir simplement exister. L'utilisation du biomimétisme, qui sera expliqué plus en détail à la section 1.1, est une façon de s'inspirer du vivant pour parvenir à recréer le vivant.

Une telle oeuvre est toujours perfectible et son créateur la trouvera toujours inachevée. Tout au long des travaux pour le développement d'Hexapode ainsi que durant la rédaction du présent mémoire, de nouvelles idées et de nouvelles possibilités ont émergé. Leur développement et leur mise au point seront fascinants pour qui en relèvera le défi.

Optimalement, ce type de robot pourra s'acquitter de tâches comme l'exploration de surfaces accidentées ou de naviguer dans un environnement encombré. Des applications spatiales peuvent être imaginées, le robot devra donc pouvoir fonctionner avec des ressources énergétiques réduites, être de petite dimension et comporter des systèmes mécaniques, électroniques et informatiques fiables. Le prototype actuel n'est pas celui qui sera envoyé sur la Lune ou sur Mars, mais celui-là pourrait être un de ses descendants. Des applications de recherche terrestre sont aussi possibles. Il pourrait aussi être utilisé comme robot domestique, son apparence doit donc être soignée pour le rendre visuellement attractif. Les présentes recherches ont été menées dans la direction de développer une plate-forme de robot mobile marcheur qui sera plus tard adaptée plus spécifiquement à la tâche qui lui sera attribuée.

Organisation du mémoire

Le présent mémoire est déposé en guise de conclusion des travaux de recherche effectués dans le cadre d'une maîtrise en génie mécanique à l'Université Laval. Le mémoire est constitué de quatre chapitres. Le premier est consacré aux notions préalables au projet. Ce chapitre présente les définitions qui sont utilisées lors des étapes de développement mécanique, électronique et algorithmique du projet.

Le second chapitre présente une analyse des animaux marcheurs en mettant l'emphasis sur leur morphologie. Grâce à l'approche biomimétique, des pistes de solution ont été dégagées pour la construction d'un robot marcheur. La deuxième partie de ce chapitre présente une revue de la littérature relative aux autres robots ayant été développés sur cette base.

Le troisième chapitre définit d'abord les critères qui serviront à évaluer les performances des divers modèles présentés. Ensuite, la morphologie des fourmis est étudiée plus en détail. Une architecture de patte est présentée et servira pour les modèles suivants. Ensuite, la première représentation robotisée est dévoilée. Suivi du modèle de patte utilisant les mécanismes déployables.

Le quatrième chapitre présente la réalisation concrète et l'analyse de la performance du prototype final de robot marcheur : «Hexapode». Ce chapitre est divisé en trois sections principales, c'est-à-dire la présentation de la performance du robot par rapport aux critères. La section sur la mobilité présente l'architecture à trois degrés de liberté qui est utilisée ainsi que le problème géométrique inverse y étant relié. La section sur l'autonomie énonce les méthodes étudiées afin de réduire la consommation énergétique requise lors de la navigation du robot. La section sur l'adaptabilité présente les explications sur la programmation hiérarchique. Trois couches de programmation sont présentées, soient la couche d'interface matérielle, la couche de coordination et la couche décisionnelle.

Le mémoire se termine par une conclusion et une présentation des applications potentielles où pourrait être utilisé Hexapode. La dernière section est une ouverture qui se veut une réflexion sur l'avenir et l'évolution des robots.

Enfin, les annexes présentent des informations complémentaires. L'annexe A contient un arbre évolutif des animaux. L'annexe B renferme les informations sur les composantes matérielles ayant été utilisées pour le prototype final d'Hexapode. L'annexe C contient certains programmes utilisés dans le cadre du projet Hexapode. Les autres programmes sont disponibles sur un CD-Rom accompagnant ce mémoire.

Chapitre 1

Route menant à Hexapode

Le présent chapitre traite du cheminement qui a conduit à la naissance d'Hexapode. Les objectifs relatifs au projet y sont énoncés. Une description des concepts clefs sur lesquels repose la recherche pour atteindre ces objectifs y est présentée.

Les points saillants du chapitre sont une définition du biomimétisme, une explication cinématique et dynamique de la marche et des diverses façons qu'ont les vivants de se déplacer. La marche requiert une architecture à topologie variable, des explications sont fournies en ce sens. La notion des polygones de contact et de support est également présentée, elle permet de mieux gérer le positionnement de l'extrémité des pattes. Enfin, les critères qui ont servi comme base au développement du prototype du robot Hexapode sont énoncés.

1.1 Biomimétisme

Selon l'encyclopédie Web, Wikipedia [1] : « Le biomimétisme est une discipline qui étudie les modèles de la nature et qui cherche à les imiter, ou à s'en inspirer pour la conception ou la mise en place de processus et d'outils destinés à résoudre les problèmes. [...] Il s'agit donc d'une nouvelle approche dans la façon de voir et d'estimer la valeur de la nature. Il est axé non pas sur l'utilisation des ressources naturelles, mais sur ce que l'on peut apprendre du fonctionnement de la biosphère. »

Le biomimétisme est une approche qui s'inspire de solutions développées par les vivants afin de résoudre des problèmes. Au fil de son évolution, la nature a fait un nombre incalculable d'itérations de design dans le développement des vivants durant les millions d'années qui se sont écoulées depuis l'apparition des organismes unicellulaires. Chacune des générations d'individus a eu sa chance pour mieux s'adapter à son environnement, les moins forts ont été retirés de la chaîne. La reproduction des individus a conduit à une diversité génétique qui s'est avérée utile et même nécessaire à l'évolution des espèces. Si la théorie de l'évolution est exacte, certaines circonstances (hasard) ainsi la qualité des individus ont fait que les vivants actuels sont le plus récent prototype du grand laboratoire de la nature. Il s'agit donc maintenant d'analyser les solutions de la nature pour les réutiliser dans des contextes artificiels.

La difficulté du biomimétisme est qu'elle implique une compréhension de la déconstruction (ingénierie inverse) de la solution naturelle, suivie d'une reconstruction dans des conditions différentes. Or l'oeil du chercheur ne voit pas toujours facilement les caractéristiques de l'organisation naturelle. Il faut qu'il se base sur les réalisations de chercheurs ayant déjà travaillé à d'autres méthodes traditionnelles pour s'attaquer au problème de façon biomimétique. La reconstruction reste toujours difficile, car elle suppose une maîtrise des processus du vivant dont la plupart restent inconnus.

Certains des principes intéressants émergeant de cette approche.

- La nature évolue au fil des générations d'individus grâce à la sélection naturelle.
- La nature minimise l'énergie dont elle a besoin pour survivre en conditions difficiles, ce qui la rend plus efficace.
- La nature ne fait pas d'excès, sinon un mécanisme inverse viendra faire l'équilibre.
- La nature est créative et s'adapte aux conditions limites qui lui sont imposées.

- La nature adopte une forme requise à la fonction.

Depuis plusieurs années, une des directions utilisée dans le domaine de la robotique mobile est de s'orienter vers des systèmes de locomotion s'inspirant des solutions biologiques qu'offre la nature. Historiquement, cette tendance fut d'observer les animaux et d'en dégager les principes locomoteurs mis en oeuvre. Ces principes sont alors implantés dans des systèmes qui tentent de les reproduire. Bien entendu, ces principes sont quelque peu modifiés et adaptés pour les rendre mécaniquement réalisables.

La marche est utilisée par les vivants pour répondre au besoin de se déplacer. Il est donc possible de s'inspirer de ce moyen de transport et de le reproduire dans la robotique. C'est ce qui a été développé dans le projet Hexapode. La morphologie du corps du robot ainsi que son contrôleur sont directement inspirés de la nature. L'approche biomimétique n'est pas la seule qui permet au robot mobile de se déplacer, par contre elle mérite de s'y attarder.

1.2 La marche

La marche est le moyen qu'ont la plupart des vivants pour se déplacer sur le sol ou au fond de l'eau. La marche est le résultat des déplacements successifs des jambes ou des pattes de l'animal. À chaque pas, les extrémités des pattes cherchent de nouveaux points d'appui. De préférence, ces points d'appui doivent être sur une surface solide et permettant l'adhérence de l'extrémité des pattes. Dans certains cas, ces conditions ne peuvent être obtenues et des compensations doivent être appliquées. Le déplacement du corps se fait donc dans la direction du prochain point d'appui. La vitesse, la stabilité ainsi que la consommation d'énergie résultant de la marche sont beaucoup fonction de la morphologie du marcheur et de la façon qu'est exécutée la marche. Deux types distincts de marche sont susceptibles d'être retrouvés : la marche quasi statique et la marche dynamique. Chacune présente des avantages et des inconvénients.

Il est possible d'analyser la mécanique de la marche en utilisant la cinématique et la dynamique. Dans le cas de la cinématique, il faut regarder la configuration des divers membres pour obtenir la position du corps par rapport au sol. La dynamique intervient au niveau des forces et des couples en jeu dans les articulations lors de la marche. Il est possible d'utiliser les principes de Newton-Euler. Ils stipulent que l'ensemble des forces

appliquées sur un corps sont égale à la masse multiplié par l'accélération de ce corps. De même, l'ensemble des couples appliqués sur un corps est égale à l'inertie du corps multiplié par l'accélération angulaire du que subit le corps. Les équations suivantes présentent ces principes sous forme vectorielles.

$$\sum \mathbf{F} = \sum m\mathbf{a} \quad (1.1)$$

$$\sum \mathbf{C} = \sum (I\boldsymbol{\alpha} + \boldsymbol{\omega} \times I\boldsymbol{\omega}) \quad (1.2)$$

Où « \mathbf{F} » est la force, « m » la masse, « \mathbf{a} » l'accélération, « \mathbf{C} » le couple, « I » l'inertie, « $\boldsymbol{\alpha}$ » l'accélération angulaire et $\boldsymbol{\omega}$ est la vitesse angulaire.

Pour qu'un objet puisse être statiquement positionné et orienté dans un espace tridimensionnel, les termes d'accélération doivent être nuls, les sommes des forces et des moments par rapport à chacun des axes doivent donc être égales à zéro. Il en résulte six équations scalaires. Un corps dans l'espace a six degrés de liberté.

Dans le cas de la marche, les contraintes résultantes des points de contact avec le sol permettent l'équilibre. Ces contacts peuvent être modélisés de diverses manières, bloquant un certain nombre de degrés de liberté. Dans le cas où des forces ou des couples sont appliqués sans que des contraintes soient bloquées ou que les contraintes ne soient pas assez fortes pour soutenir les forces ou couples en jeu, il en résulte un déplacement. Ce déplacement peut être désiré par le marcheur (cela dépend de sa façon de marcher), c'est ce qui est expliqué dans les prochaines sections.

1.2.1 Morphologie du marcheur

La forme du corps, des membres et des articulations ainsi que le nombre de pattes influencent la façon de marcher. Le positionnement de l'extrémité des pattes doit se faire de façon à avoir un appui fiable pour permettre de déplacer le corps. Afin d'assurer plus de stabilité, l'extrémité des pattes de certains vivants se compose d'une cheville et d'un pied ou du moins d'un organe terminal donnant une plus grande adhérence à la patte.

Pour accomplir un positionnement dans l'espace, un minimum de trois degrés de liberté sont nécessaires. Tandis que pour permettre également d'orienter l'extrémité de la patte, un minimum de six degrés de liberté sont nécessaires. Cependant, ces degrés de liberté sont parfois compliants ou sous-actionnés.

Les articulations des vivants peuvent être modélisées mécaniquement par des liaisons rotoïdes, de cardan ou sphériques et dans de rares cas les liaisons sont prismatiques ou cylindriques. La modélisation des membres des vivants permet une meilleure compréhension des mouvements menant au déplacement. Chacun de ces types d'articulation permet un certain nombre de degrés de liberté :

- rotoïde = un degré de liberté en rotation,
- cardan = deux degrés de liberté en rotation,
- sphérique = trois degrés de liberté en rotation,
- prismatique = un degré de liberté en translation,
- cylindrique = un degré de liberté en translation et un degré de liberté en rotation.

Les types d'articulation permettant uniquement des rotations sont répandues. Ils permettent d'orienter les membres pour permettre un nouveau positionnement de l'extrémité de la jambe ou de la patte. Ces types d'articulation sont répandus puisqu'ils impliquent le contact des membres en tout temps sans que des glissements soient nécessaires. Le système moteur des vivants est habituellement composé de muscles qui sont fixés sur le squelette ou la carapace par des ligaments. Les types d'articulation permettant des translations sont très peu fréquents et se rencontrent dans des membres télescopiques, la langue de certains insectes et crustacés fonctionne par exemple en s'allongeant grâce à la pression interne du corps de l'animal. Les débattements articulaires des vivants sont évidemment limités. Les membres entre les articulations doivent être liés pour permettre l'apport de nourriture aux cellules. Les membres peuvent habituellement être modélisés comme des corps rigides, bien que dans certaines circonstances ce ne soit pas le cas (membres déformables compliants).

Le nombre de pattes des vivants est aussi un critère qui influence la façon de marcher. Les grandes branches des vivants ont pour la plupart quatre ou six pattes. Parfois, comme dans le cas des bipèdes ou des oiseaux, une paire de pattes s'est adaptée à la préhension d'objets avec les mains, ou s'est transformée en ailes pour pouvoir voler, la marche résultant de ces configurations utilise cependant les mêmes principes de base.

Ceci sera mis en évidence dans le chapitre 2.

1.2.2 Topologie variable

Une modélisation des marcheurs peut donc être faite avec un corps, des membres et des articulations. Durant la marche, la configuration du modèle change. Ceci se produit lors du déplacement des pattes qui sont en contact ou non avec le sol (déplacement volontaire) ou lorsqu'il se produit un glissement (déplacement involontaire).

Par définition, un mécanisme à topologie variable est «un système pour lequel la représentation par graphe varie selon la réalisation d'une tâche»[4]. Dans le cas de la marche, le nombre de chaînes cinématiques partant du corps jusqu'au sol est variable selon la quantité et la qualité des points de contact au sol. Pour produire un déplacement, les pattes fournissent une alternance de chaînes cinématiques ouvertes et fermées entre le corps et le sol. Un marcheur peut donc être considéré comme un mécanisme à topologie variable. Il est donc utile de vérifier la qualité des points de contact pour connaître la stabilité de la marche.

Pour représenter la marche, il est possible de la comparer à des déplacements de robot parallèle dont la base serait déformable successivement à chaque pas, et donc à topologie variable.

1.2.3 Points de contact

La marche est une succession de points de contact pour chacune des extrémités des pattes. Ces points de contact avec le sol peuvent être modélisés de plusieurs manières. La figure 1.1 illustre les deux représentations des types les plus communs.

Lorsque deux surfaces sont fixées en contact, six contraintes sont en jeu (trois contraintes de positionnement et trois contraintes d'orientation). Cependant, certaines contraintes sont d'ordre moins important et sont alors moins critiques dans la modélisation du contact. Il devient alors possible de simplifier le modèle du contact sans perte d'informations critiques. Cette simplification engendre l'amélioration des temps

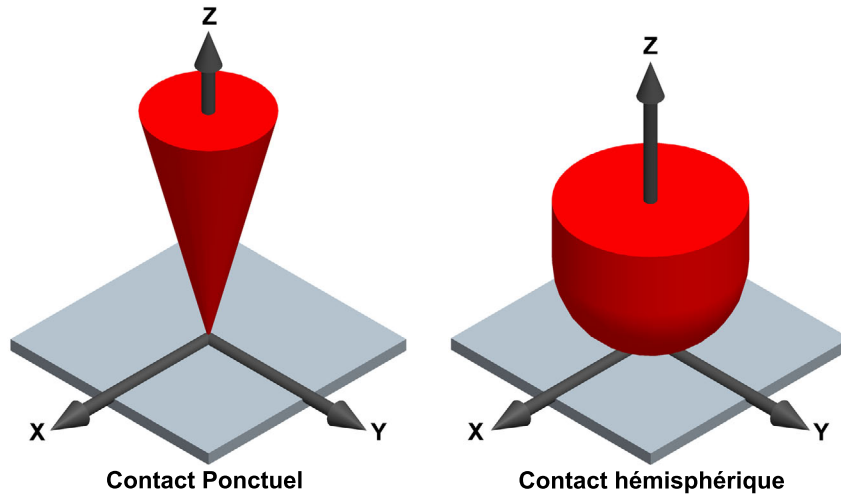


FIG. 1.1 – Modélisation du contact

de calcul dans les simulations des modèles.

Les extrémités des patte du robot Hexapode sont de forme hémisphérique. Le contact qui se fait sur la surface portante pourrait donc être modélisé comme hémisphérique. Dans ce cas, l'orientation du contact ferait aussi varier la position de l'extrémité de la patte. La variation maximale de la position du point de contact est de la grandeur du rayon de l'hémisphère. Le diamètre des pattes devant être petit par rapport au déplacement en jeu durant la marche (ratio approximé à 1 :25), il est possible de négliger le déplacement du point de contact sur l'hémisphère.

Le contact ponctuel est une simplification du précédent contact. Il modélise uniquement les contraintes de positionnement. Dans le cas qui nous intéresse, la précision du positionnement du contact nécessaire ne justifie pas actuellement l'utilisation du modèle de contact hémisphérique, le modèle de contact ponctuel est donc employé. Partant de cette hypothèse, il n'y a pas sur le robot de système prenant la mesure du point de contact réel, une mesure de la pression exercée de façon normale à l'hémisphère est prise (discuté à la section 4.5.1.2). La surface de l'extrémité de la patte est faite d'une gaine en caoutchouc qui est déformable et qui épouse la surface de contact, ceci qui fait encore diminuer la zone d'incertitude de la position du contact.

1.2.4 Polygones

Une des innovations propres au développement du robot Hexapode est la représentation de la marche au moyen de la notion de polygones. Les polygones contiennent les informations provenant des extrémités des pattes. La nature de ces informations dépend de la modélisation des points de contact. Dans notre cas, le contact est modélisé ponctuellement, les positions cartésiennes des extrémités des pattes sont donc ce que renferment les polygones.

La méthode résultant de la notion de polygones gère les caractéristiques des extrémités des pattes en les classant selon leur degré d'appartenance à deux groupes distincts : le polygone de contact et le polygone de retour. La marche est, pour chacune des extrémités des pattes, une succession d'états d'appartenance à ces deux polygones.

Le polygone de support est défini comme l'arrangement géométrique des points de contact au sol. Les caractéristiques de ce polygone sont données par rapport au repère fixé sur le corps du robot. Pour que la marche soit stable, il faut que les caractéristiques de position et d'orientation relative de cette géométrie, ne se déforment pas durant l'avancée du corps. Tous les points composant le polygone de support doivent donc subir, en tout temps, les mêmes transformations, en translation ou en orientation.

Le polygone de retour est défini par l'arrangement géométrique des extrémités des pattes qui ne sont pas en contact avec le sol. Puisque ces pattes forment des chaînes cinématiques ouvertes entre le sol et le corps du robot, les extrémités de ces pattes peuvent se déplacer librement et de façon indépendamment. En fait, leur déplacement doit se faire dans la direction menant au prochain polygone de support le plus favorable. Cette géométrie est donc déformable.

La notion de polygones est conforme à l'analogie entre les marcheurs et les robots parallèles à topologie variable. Ce sont les variations d'orientation et de position pour tout le polygone de support par rapport au corps, qui permettent au marcheur de se déplacer. Il faut donc que tous les points du polygone de support subissent les mêmes transformations en tout temps.

1.2.5 Stabilité

Lorsque l'extrémité d'une patte comprise dans le polygone de support subit un déplacement relatif par rapport à l'extrémité des autres pattes, il y a une diminution locale de la stabilité de la marche du robot. Cette instabilité n'entraîne pas nécessairement la chute, mais fait passer le marcheur dans une nouvelle phase de recherche de stabilité.

La perte de stabilité est le résultat de la perte de contraintes du point de contact. Les effets encourus sont proportionnels à l'importance de la contrainte perdue. Par exemple, lorsque le marcheur s'appuie sur une surface molle, la perte de la contrainte de la force normale au sol résulte habituellement en une chute. Alors que la perte de la contrainte de la rotation autour de l'axe normal au sol n'est habituellement pas critique.

Les études de stabilité passent par les équations du mouvement (1.1). Les contraintes présentes au sol doivent être suffisantes pour ne produire que des accélérations selon les directions et les orientations conformes au trajet désiré. Dans le cas contraire, la marche devient instable et il en résulte une possibilité de chute du marcheur. Lors de la déformation du polygone de support le nouveau polygone généré doit être en mesure de contraindre tout les efforts et les moments venant de la dynamique actuelle du robot. Une nouvelle pose stable doit être recherchée après que le système soit devenu localement incontrôlable.

1.2.6 Type de marche

Comme il y a plusieurs sortes de marcheurs, il y a plus d'une façon de marcher. Chaque animal adapte sa façon de marcher à sa morphologie. La marche implique le mouvement, ce mouvement implique des accélérations plus ou moins élevées. Il est donc possible de distinguer deux types généraux de marche, la marche dynamique et la marche quasi statique. La plupart des animaux peuvent utiliser les deux types, mais leur morphologie est souvent optimisée pour un type en particulier.

1.2.6.1 Marche dynamique

Dans la marche de type dynamique, les effets d'accélération et d'inertie sont dominants. Cette façon de marcher utilise les perturbations d'équilibre pour produire des déplacements dans la direction désirée. Le marcheur positionne son centre de masse en dehors de la plage de stabilité et se sert de la quantité de mouvement déjà acquise pour poursuivre son mouvement. Cette démarche est donc une succession de déstabilisations contrôlées pour produire l'élan nécessaire pour avancer.

On pourrait comparer cette façon de marcher aux oscillations d'un pendule. L'énergie cinétique de la masse est convertie en énergie potentielle qui retourne dans le cycle en énergie cinétique. Cependant, il est impossible d'avoir un environnement réel s'approchant des conditions théoriques. Pour que la marche soit fluide, des corrections doivent donc être faites en tout temps pour permettre une continuité du mouvement. Ce type de marche est adapté pour des déplacements continus sur de grandes distances sans arrêts fréquents. Les modèles utilisant ce type de marche impliquent la résolution de la dynamique du marcheur en temps réel, ce qui rend critique la résolution simultanée de beaucoup de paramètres en même temps et demande plus de puissance de calcul.

1.2.6.2 Marche quasi-statique

La marche quasi statique correspond aux cas où les effets d'accélération et d'inertie sont négligeables. Cette démarche permet des poses stables en tout temps et même des arrêts dans la démarche. Les déplacements qui résultent de la marche quasi statique sont souvent lents. Les termes de droite des équations de mouvements (1.1) deviennent très faibles et donc négligeables. L'énergie impliquée dans ce type de marche est faible.

Une pose est la configuration articulaire que prend un marcheur à un certain instant. Comme chaque pose qui décompose les mouvements de la marche est stable, un équilibre des forces et des moments est atteint en tout temps. Ce type de marche est adapté pour les déplacements sur de courtes distances et accompagnés d'arrêts fréquents. Les modèles utilisant ce mode de marche ne requièrent que la résolution de l'information de la position de l'extrémité des pattes, de la qualité du contact avec le sol et de la minimisation du chargement sur les muscles ou actionneurs. Ceci rend ce type de marche moins gourmande en temps de calcul.

Chapitre 2

Évaluation des morphologies

Ce chapitre fait l'évaluation de la morphologie de certains animaux, vertébrés et in-vertébrés. Le parallèle avec leur utilisation possible dans la robotique est fait en parlant des caractéristiques principales qui constituent leur système de locomotion. Les notions de biomimétisme sont utilisées pour la description des morphologies qui remplisse la tâche de permettre le déplacement. Certains des travaux antérieurs dans le domaine des robots mobiles marcheurs sont aussi présentés pour situer Hexapode dans le domaine.

2.1 Animaux

L'objectif du projet Hexapode est de mettre au point un robot marcheur autonome, il est donc essentiel d'exécuter une étude initiale spécifiquement sur les vivants

se déplaçant par eux-mêmes. Le règne des animaux est intéressant à imiter dans le domaine de la robotique à cause des dimensions de leurs membres et de la capacité de ceux-ci.

En annexe A, la figure montre un arbre de classification du règne animal pluricellulaire. Deux grandes branches sont visibles : les vertébrés possédant un squelette interne (endosquelette) et les invertébrés, qui ont un squelette externe (exosquelette) ou pas de squelette du tout. Ceux qui n'en ont pas sont moins compatibles avec le monde de la robotique puisque leur corps mou est composé de muscles, n'ayant pas d'équivalent direct mécanique. Pour les soins du projet Hexapode, nous nous restreindrons aux marcheurs terrestres et plus spécifiquement à l'analyse des insectes qui ne volent pas de la famille des arthropodes de l'ordre des invertébrés, ainsi que des mammifères et des reptiles de l'ordre des vertébrés.

Certaines espèces animales ont été étudiées pour vérifier leur performance et la faisabilité de s'en inspirer pour le domaine de la robotique. L'analyse des morphologies particulièrement efficaces a été effectuée et a permis de dégager les paramètres de base du design du robot. La présente section fait une évaluation sommaire des divers animaux au sens du biomimétisme.

2.1.1 Vertébrés

Les vertébrés sont caractérisés par la présence d'un axe médian dans leur squelette : la colonne vertébrale. Leur morphologie se présente habituellement en trois ou quatre parties.

- La tête dans laquelle se place l'encéphale, est le siège des principaux organes des sens : vision, audition, touché, olfaction, et goût. Le centre de l'équilibre se trouve habituellement dans la tête.
- Le tronc contient les viscères, les systèmes circulatoires et digestifs. Une queue peut être rencontrée comme le prolongement de la colonne vertébrale. Cette dernière peut favoriser l'équilibre de l'animal lors de ses déplacements.
- Les membres, généralement au nombre de quatre, permettent à l'animal de se déplacer.

Les organes internes assurent au corps un apport en énergie et constitue un système interne de nettoyage de lubrification et de réparation. La peau, qui est aussi un organe, est la frontière entre l'environnement extérieur et l'intérieur du corps. La peau sert de paroi au-dessus des muscles et les autres organes internes. Son rôle est de protéger contre les agressions biologiques (virus, bactérie, etc.) ou physiques (température, humidité, etc.). Le corps des vertébrés implique habituellement une grande quantité de muscles, ce qui nécessite une alimentation en énergie assurée par une plus grande consommation alimentaire.

La structure interne du corps des vertébrés est un endosquelette. Elle est composée de deux types de tissus : les cartilages et les os. La colonne vertébrale sert de support flexible aux membres. Elle permet une grande agilité, la flexion de la colonne vertébrale donne la possibilité de modifier les paramètres dynamiques du corps. Elle permet de se servir des déplacements du corps pour produire des changements de trajectoire même lorsque aucun point de contact n'est disponible, par exemple lorsqu'un chat se tourne dans les airs pour retomber sur ses pattes.

Les articulations des vertébrés sont constituées de muscles, de cartilages et de ligaments permettant des débattements articulaires restreints. L'extrémité des membres est habituellement constituée de pieds ou de sabots pour faciliter la marche et prévenir l'usure des tissus durant de longs trajets.

Les vertébrés utilisent de préférence une démarche dynamique. Leur morphologie les guide à utiliser la continuité du mouvement pour minimiser la dépense en énergie. Les ajustements nécessaires pour rendre stable la démarche sont rendues possibles par la facilité de flexion de la colonne vertébrale. Il en résulte des aptitudes à la course, à des enjambées pour franchir des obstacles ou même à effectuer des sauts ou d'autres acrobaties.

Continuellement pendant la marche le cerveau reçoit de l'information afin d'ajuster la force fournie par les muscles pour permettre le positionnement des membres. La hauteur, l'amplitude et la fréquence des pas mènent à des démarches efficaces pour les déplacements du vertébré qui les effectue.

2.1.1.1 Quadrupède

Les quadrupèdes sont la classe des vertébrés ayant quatre pattes ou membres. Chez certains, tous les membres sont utilisés pour la marche, la fonction première de tous leurs membres est donc de permettre le déplacement. Pour d'autres, les marcheurs bipèdes, seulement deux des membres sont utilisées pour la marche, les pattes avant ont évolué en bras et mains pour la préhension d'objets. Deux branches de vertébrés terrestres sont intéressantes pour l'étude de la marche : les reptiles et les mammifères. Les oiseaux, les poissons et les amphibiens ne sont pas des animaux vivant sur la terre ferme, leur corps est adapté à des utilisations différentes que la marche.

Les reptiles sont des animaux à sang-froid, ce qui implique qu'ils doivent être dans un environnement où la température ambiante est dans leur zone de fonctionnement. Le fait de ne pas réguler leur température interne présente l'avantage d'avoir à fournir moins d'énergie. Cependant, cela amène comme désavantage de les contraindre à un environnement spécifique. Leurs déplacements sont donc habituellement plus courts que ceux des mammifères.

Il existe quatre ordres chez les reptiles, notamment les saurophidiens dont font partie les lézards. Cette classe est redivisée en deux groupes principaux. Les ascalabotes regroupent les lézards de type geckos et iguane. Ils se déplacent sans exécuter de mouvement ondulatoire de reptation (se déplacer en rampant). Les autarchoglosses regroupent les autres lézards qui peuvent se déplacer par ondulations latérales de leur corps, comme les serpents.

La plupart des lézards sont d'une taille comprise entre quinze et quatre-vingts centimètres. Cette taille les rend attirants du point de vue biomimétique puisque cette échelle est adéquate pour obtenir un robot ayant des systèmes électroniques et des actionneurs de dimensions répandues sur le marché.

Les mammifères sont des vertébrés à sang chaud. Il existe 21 ordres chez les mammifères. Ici sont décrit ceux qui habitent principalement sur terre et se déplaçant en marchant et qui serait les plus aptes à être amenés dans la robotique.

- Ongulés : Ordre qui contient les mammifères herbivores caractérisés par la présence de sabots. Leur morphologie leur permet de franchir de grande distance afin de

s'alimenter. La fréquence de la course pour les ongulés a entraîné des modifications de leurs membres. Les pattes sont longues et le rétrécissement de la surface de contact avec le sol améliore les performances des déplacements à la course, dans le cas des ongulés seul le bout de l'ongle touche au sol, les rendant de très bons coureurs. Tant les chevaux que les éléphants font partie de cet ordre.

- Carnivores : Quatre catégories de carnivores existent. Les marcheurs ont des membres non spécialisés et se déplacent habituellement sur la plante des pieds, on y retrouve notamment les ours. Les coureurs marchent sur la pointe des doigts, leurs membres sont longs et bien adaptés à la course. Les fouisseurs ont des membres courts spécialisés dans le creusage, comme le blaireau. Les grimpeurs ont les membres adaptés pour s'agripper et marchent aussi sur les doigts pour les pattes avant. La manière de se déplacer des carnivores est efficace et spécialisée. Cependant, ils ne peuvent faire de déplacement sur de longues périodes puisque cela requiert beaucoup d'énergie, qu'ils doivent reprendre en mangeant ou en se reposant.
- Insectivores : Groupe de petits mammifères mangeant des insectes. Les insectivores sont bien adaptés à des conditions difficiles, par exemple la taupe à des environnements souterrains ou le hérisson à des conditions nocturnes. L'ouïe et l'odorat sont leurs sens les plus développés. Leur marche est stable, mais lente à cause de leurs pattes très courtes.
- Lagomorphes : Groupe de petits mammifères dont font partie les lapins et les lièvres. Leurs pattes arrière sont longues et fortes, permettant à ces mammifères de se déplacer par bonds. Toute la poussée du déplacement vient des pattes arrière, les pattes avant ne servant qu'à préparer le prochain pas.
- Rongeurs : Mammifères herbivores caractérisés par une paire de larges dents qui servent à ronger la végétation. Leur morphologie varie selon les espèces, certains sont adaptés à creuser des terriers et d'autre à grimper aux arbres. Ils ont en général une bonne dextérité avec leurs pattes avant, leur permettant de saisir des objets. Leurs pattes sont courtes ce qui fait d'eux des marcheurs lents. Leur corps souvent trapu, mais flexible leur permet d'accomplir des déplacements avec assez d'agilité.
- Primates : Groupe de mammifères essentiellement arboricoles, comprenant les lémuriens et les singes. Les membres des primates se terminent par des mains et des pieds préhenseurs. Les déplacements des primates sont agiles et efficaces, mais leur morphologie ne leur permet pas de parcourir de grande distance. Bien que quadrupèdes, ils sont aussi habiles à des démarches bipèdes, libérant leur main

pour la manutention d'objets. L'être humain est dans cet ordre et a délaissé sa nature arboricole pour se consacrer uniquement au déplacement bipède.

2.1.1.2 Bipèdes

Les marcheurs bipèdes utilisent la marche dynamique en utilisant diverses combinaisons de points d'appui. Les pieds humains peuvent être considérés comme permettant trois points de contact, un au gros orteil, un au petit orteil et l'autre sous le talon. Pour qu'une démarche quasi statique stable soit rencontrée, il faut que le centre de masse soit au-dessus du polyèdre de contact, ce qui ne laisse pas une grande marge.

Des recherches sur des mécanismes marcheurs bipèdes utilisant la marche dynamique ont été réalisées au Laboratoire de biorobotique de l'Université de Delft [5], au laboratoire de biorobotique et locomotion de l'université Cornell [6] (figure 2.1) et au Laboratoire de biorobotique du MIT. L'approche occidentale est de modéliser mathématiquement les principes de la marche, une démarche puriste. Les résultats obtenus sont impressionnants, cependant le nombre et la complexité des paramètres à considérer est très imposant, ce qui rend ces modèles quelquefois peu efficaces.

L'approche asiatique est à l'inverse, elle est basée sur des résultats empiriques et tente de faire répéter les mêmes séquences d'actions au robot pour qu'il marche. Les résultats sont encore plus extraordinaires. On peut penser au célèbre robot Asimo qui est le dernier prototype de Honda (figure 2.2 et [2]). Un autre robot bipède impressionnant est Qrio de Sony [3].

Le développement des sens pour permettre au robot d'interagir avec les humains est aussi un aspect important. L'Université Waseda est un institut spécialisé uniquement dans le développement de robots humanoïdes et des activités parallèles y étant reliées. Un des robots intéressants est Wendy, ce robot est illustré à la figure 2.3. C'est un robot interactif qui mise sur l'intelligence artificielle pour permettre de réaliser des tâches complexes et de créer des interactions avec les humains.

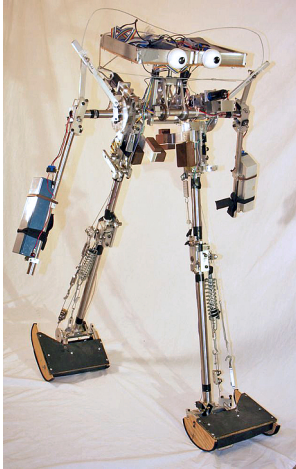


FIG. 2.1 – Marcheur dynamique de Cornell

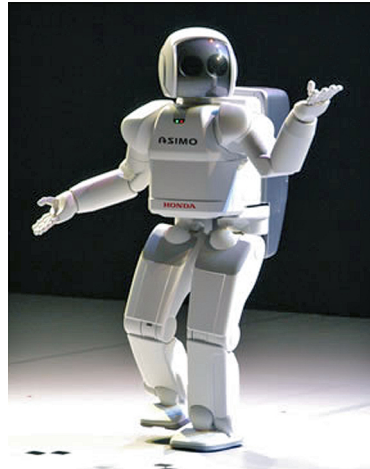


FIG. 2.2 – Asimo de Honda

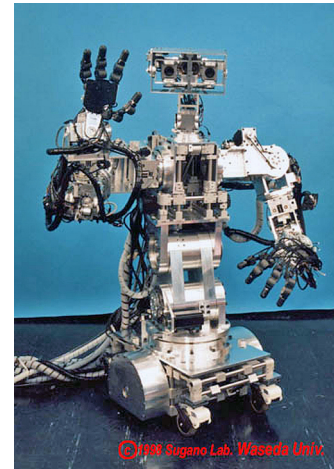


FIG. 2.3 – Wendy de l'université Waseda

2.1.2 Invertébrés

Les arthropodes sont des invertébrés dont le corps est formé par segments et qui sont recouverts d'une carapace rigide. Ils représentent environ quatre-vingt-dix pour cent des espèces animales connues [22]. Ces animaux ont une morphologie moins complexe (moins avancée en terme d'évolution) que celle des vertébrés. Cela ne veut pas dire qu'ils sont moins efficaces, en effet les arthropodes sont en étroite relation avec leur environnement.

On pourrait se faire une représentation biomimétique d'eux comme étant des mécanismes à câbles (fibres musculaires) qui sont introduits dans une enveloppe rigide. Leurs articulations compliantes sont les endroits où leur carapace est moins épaisse et donc plus flexible. Le reste du contenu du corps est la somme des éléments nécessaires pour l'alimentation en énergie. Leurs capteurs sont perfectionnés pour leur permettre d'avoir une représentation de l'environnement nécessaire à leur survie. Partant de cette idée, il devient donc possible d'imaginer des robots fonctionnant de manière similaire.

Dans ce groupe animal, on retrouve les crustacés, les arachnides et les insectes. Les crustacés sont principalement des animaux aquatiques, ils sont d'un intérêt moins grand pour l'étude de la marche terrestre. Les sections suivantes couvriront les arachnides et s'attarderont au insectes.

2.1.2.1 Arachnides

Les arachnides sont essentiellement représentés par les araignées (figure 2.4). Elles sont constituées de deux segments, le céphalothorax et l'abdomen. Elles ont habituellement quatre paires de pattes. Certaines ont des pattes particulièrement agiles, généralement minces et assez courtes. Ceci leur permet de se déplacer dans leur toile sans s'empêtrer dans leur propre piège. D'autres araignées n'utilisent pas de toile pour chasser et ont à se déplacer très rapidement au sol pour capturer leur proie, elles ont des patte plus longue et plus forte.

Le fait de posséder huit pattes avec plus de quatre segment augmente la difficulté de les imiter en robotique. Les araignées sont souvent longtemps immobiles en attente de leur proie. Ces prédateurs ne sont pas faits pour parcourir de grandes distances. De plus, la complexité de leur morphologie et le nombre abondant d'actionneurs pour mettre en mouvement toutes ces pattes génèrent de grandes contraintes opérationnelles de fonctionnement dans la robotique.

2.1.2.2 Insectes

Les insectes sont parmi les êtres vivants qui se sont le mieux adaptés à leur milieu. Ce sont les vivants les plus répandus sur la Terre, avec plus de 750 000 espèces [22]. Ils se retrouvent dans toutes les régions tempérées du globe. Ce sont des arthropodes pourvus d'antennes et de mandibules. Leur corps est constitué de trois segments : la tête, le thorax et l'abdomen. Au thorax est fixé trois paires de pattes et sur certaines espèces deux paires d'ailes.

Leur système nerveux est bien intégré à leur corps et leur cerveau est dorsal, donc dans le thorax. La tête est le siège de la vision, avec des yeux en facettes. L'entrée du tube digestif est précédée de puissantes mandibules et de mâchoires qui peuvent également servir à la manutention d'objet. Sur la tête sont également situées des antennes qui servent d'organe tactile, de communication et d'odorat. L'abdomen renferme le système circulatoire et digestif ainsi que les glandes à venin pour certaines espèces.

Malgré leur petite taille, leurs possibilités sont énormes. Leur physionomie leur permet une intégration parfaite à leur environnement bien que leur complexité physiologique soit réduite. De plus, leur corps est doté d'un exosquelette léger et résistant.

Ceci est d'autant plus évident lorsqu'une comparaison, à l'échelle de leur dimension, de leur masse et de leur force, est faite par rapport à d'autres créatures. Les insectes ont tendance à utiliser la marche quasi statique pour s'assurer de toujours pouvoir être dans une position stable en tout temps.

Toutes ces particularités sont source d'inspirations potentielles pour la robotique. Comme la présente étude porte sur le sujet d'un robot marcheur, seuls les insectes terrestres seront investigués.

2.1.2.3 Polynéoptères

La famille des polynéoptères est un groupe d'insectes habituellement ailés, mais dont les aptitudes à la marche au sol les rendent intéressants à analyser en vue du développement d'un robot marcheur. On y retrouve les grillons, les criquets et les sauterelles (figure 2.5 et [8]). Leurs paires de pattes sont symétriques selon un axe parallèle au corps et très différente entre elles. Celles d'en avant sont agiles pour la recherche de nourriture et pour assurer une reconnaissance du terrain, les pattes centrales servent de soutien au corps et finalement les pattes postérieures sont adaptées pour pouvoir donner de puissantes impulsions en vue d'un saut.

Cette famille inclut aussi les blattes ou coquerelles. La partie inférieure du thorax de celle-ci est élargie en forme de bouclier et les protège de leurs prédateurs. Plusieurs études ont été réalisées sur ces insectes et leur morphologie ainsi que leur façon de marcher sont décrites dans la littérature (section 2.2).

2.1.2.4 Phasmes

Certains des autres membres de la famille des polynéoptères sont les perce-oreilles, les termites et les phasmes (figure 2.6). Leur corps de forme allongée a l'aspect d'une brindille. Leur morphologie a l'aspect de feuilles ou de branches (d'où leur nom anglais : *stick insect*). Les phasmes peuvent atteindre jusqu'à trente centimètres, ce qui est une échelle intéressante dans le domaine de la robotique. Leur taille ainsi que leur lenteur sont d'autant pratiques puisque cela permet d'étudier facilement leurs mouvements durant la marche [9].

2.1.2.5 Fourmis

La fourmi possède un physique hexapode, particulièrement bien constitué (figure 2.7). Petite, agile et forte, elle répond aux critères mécaniques de la robotique pour le présent projet. Les détails obtenus de la documentation ont montré que la fourmi était digne d'une étude plus approfondie. La fourmi est une marcheuse quasi statique à six pattes, un des aspects intéressant est sa facilité à s'adapter lorsque certaines de ses pattes doivent accomplir d'autres tâches, comme la manutention d'objet. Un autre détail intéressant de la fourmi est son aptitude à vivre en société et la spécialisation des morphologies associés au castes (travailleur, soldat, reine). Cette approche pourrait aussi être transposé dans la robotique si des équipes de robot fonctionneraient pour la réalisation de tâches complexes.

Une description plus approfondie de sa morphologie est faite à la section 3.2 du prochain chapitre. Les critères d'évaluation présentés à la section 3.1 permettront de mieux montrer pourquoi la réalisation du robot Hexapode s'est inspiré de cet insecte.

2.2 Historique de la robotique des insectes

En comparant les avantages et les inconvénients des morphologies des insectes, il devient possible de s'en inspirer en robotique. En effet, leur corps est en général simplement constitué et très efficace. La robotique utilise habituellement des exosquelettes où la carapace tient compte de châssis pour protéger et retenir les diverses composantes. Historiquement, plusieurs robots marcheurs ont été inspirés des insectes, il est donc possible de retrouver des références dans la littérature. La présente section résume certains des travaux importants faits dans le domaine des robots marcheurs qui s'inspirent des insectes de façon biomimétique.

2.2.1 D.M. Wilson

Les premières recherches majeures dans le domaine de la marche chez les insectes ont été initiées par D. M. Wilson en 1966 [7]. Ses travaux portaient sur la mise en



FIG. 2.4 – Araignée



FIG. 2.5 – Sauterelle



FIG. 2.6 – Phasme



FIG. 2.7 – Fourmi

place d'un modèle descriptif catégorisant les divers modes de marche rencontrés chez les insectes, plus particulièrement chez les phasmes (figure 2.6) et les scarabées. Grâce à ses observations empiriques, Wilson a énoncé des lois régissant l'ensemble des mouvements généralement rencontrés lors de déplacement des insectes marcheurs à six pattes sur des surfaces horizontales, ces lois seront décrites plus loin dans la section 4.5.5.2. Encore aujourd'hui, les travaux de Wilson sont très actuels puisque l'application de ces lois permet de reproduire des démarches stables et adaptables selon les conditions du terrain.

2.2.2 K. Pearson

Les travaux de Keir Pearson depuis 1976 [8] au Centre de neuroscience de l'Université de l'Alberta ont porté sur la mise en forme des comportements étudiés par Wilson sous forme de réseaux neuronaux. L'extrapolation faite des lois de Wilson sous forme de comportements neuronaux à réflexes servira par la suite de base à l'implantation d'algorithmes sur des machines marchantes. Pearson complète et valide les conclusions de Wilson par une approche neurobiologique des vivants en se servant plus précisément de sauterelles comme cobaye (voir section 2.1.2.3). Cependant vu la puissance de calcul des ordinateurs de cette époque, le modèle de Pearson est resté théorique pendant longtemps, avant d'être enfin validé. Il sera démontré plus tard, par d'autres chercheurs en biorobotique, qu'il est possible d'obtenir des démarches stables grâce à l'implantation de ce modèle sur un robot réel.

2.2.3 H. Cruse

Les travaux de Holk Cruse [9] et [10] sont importants dans le domaine de la cybernétique biologique, et par extension, de la robotique mobile biomimétique. Depuis 1972, ses recherches portent sur l'étude du transfert d'informations constaté à l'intérieur du système nerveux, durant divers mouvements et comportements des êtres vivants. Les travaux de Cruse sont une extension de ceux de Wilson et de Pearson. Il travaille plus précisément à la modélisation des réseaux neuronaux utilisés par les insectes lors de la marche. Les résultats obtenus sont assez simples pour être applicables en temps réel sur des robots et suffisamment élaborés pour donner de bons modèles qui se rapprochent

du comportement des insectes. Les modèles tiennent compte de la coordination des pattes les unes par rapport aux autres et du contrôle individuel des mouvements des pattes. La clef de ses recherches est d'obtenir des résultats expérimentaux sur des sujets vivants et de tenter de reproduire les comportements à partir de modèles neuronaux, puis de valider ces modèles sur des robots dont la morphologie est semblable au cobaye de l'expérience. Les comportements peuvent venir de capteurs ou de données. Ces données peuvent être préprogrammées ou provenir de sources extérieures qui analysent l'environnement.

2.2.4 MoBot Lab du MIT

L'idée de développer un robot marcheur avec une approche biomimétique pour l'exploration de terrain n'est pas nouvelle. Quelques projets ont été réalisés et certains sont en cours de réalisation. Un des premiers robots fonctionnels, nommé Hannibal (figure 2.8), a été construit au MIT en 1991 sous la supervision de Rodney A. Brook. Il comportait, au total, dix-neuf degrés de liberté actionnés, c'est-à-dire trois degrés pour chacune de ses six pattes et un autre pour changer l'orientation du premier axe de toutes les pattes. Ce robot pesait environ trois kilogrammes et était équipé de plus de cent capteurs et de huit ordinateurs de bord. Il était le successeur du robot Genghis [11] développé par Brook et palliait aux carences de mobilité et d'autonomie de Genghis. Une équipe nombreuse s'est répartie la tâche de construire physiquement Hannibal et Attila, son jumeau ([12] et [13]), ainsi que leur contrôleur ([14] et [15]).

2.2.5 Université de Bremen

Le robot Aramies (figure 2.9) a été réalisé par l'université de Bremen pour l'agence spatiale européenne ([16] et [17]). Cette plate-forme est spécifiquement dédiée à des déplacements sur des surfaces terrestres ou martiennes, qui seraient impossibles à réaliser avec des robots à roues.

L'objectif de ce projet est de développer un robot multifonctionnel permettant d'accomplir des tâches autonomes dans des environnements difficiles. Ceci inclut les reliefs très escarpés, les inclinaisons et même l'escalade de falaise. Il est le successeur de

Scorpion. Cette nouvelle version est spécifiquement dédiée à des déplacements sur des surfaces terrestres ou martiennes, qui seraient impossibles de réaliser avec des robots à roues. Des études impliquant la collaboration simultanée de plusieurs robots ont été réalisées. Il est à noter qu'Holk Cruse, cité auparavant, fait partie de l'équipe travaillant sur ces robots.

2.2.6 Jet Propulsion Lab (Nasa)

Le projet Lemur (figure 2.10) du Jet Propulsion Lab (NASA) propose un robot œuvrant aussi dans le domaine spatial où les tâches à accomplir requièrent une grande dextérité [18]. Le robot Lemur est un hexapode destiné à inspecter, assembler et réparer des installations spatiales. En plus de marcher de façon efficace, l'extrémité de ses pattes comporte des outils interchangeables. Chacune de ses pattes est complètement autonome et comporte quatre degrés de liberté actionnés. Il est aussi doté d'une vision stéréoscopique, ce qui lui permet un contact étroit avec son environnement. C'est un robot polyvalent et bien adapté à ses tâches.

2.2.7 Case Western Reserve University

Un autre laboratoire montrant des résultats des plus surprenants est celui du CWRU. Leur recherche principale porte sur une plate-forme biomimétique reproduisant le corps d'une coquerelle. Chacune des trois paires de pattes du robot possède un nombre différent de degrés de liberté, soient respectivement sept pour la section avant, cinq pour la section centrale et trois pour la section arrières. Le plus impressionnant dans leur design est l'utilisation de muscles artificiels dits «muscles de McKibben» : ces muscles artificiels sont actionnés de façon pneumatique ce qui fait que pour l'instant le robot n'est pas autonome quant à son alimentation énergétique [19].

Un autre robot intéressant de ce laboratoire ayant été conçu depuis le début des travaux sur le robot Hexapode, est le robot Bill-Ant (figure 2.11). Ce robot s'inspire de la morphologie des fourmis et intègre un contrôleur du type de ceux développés par Cruse. Il comporte des capteurs de contact pour recueillir l'information en provenance des pattes en contact avec le sol. Il est également pourvu d'une tête avec des mandibules

pour saisir les objets.

2.2.8 Hexapode

Enfin, le présent robot Hexapode est en développement au Laboratoire de robotique de l'Université Laval depuis mai 2003([20] et [21]). Les chapitres qui suivent le décrivent en détail.



FIG. 2.8 – Hannibal

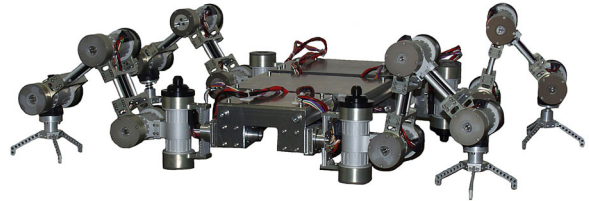


FIG. 2.9 – Aramies

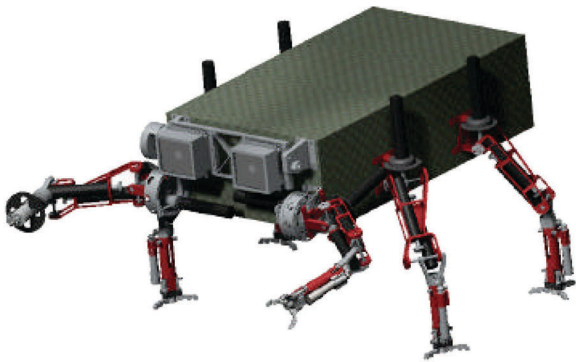


FIG. 2.10 – Lemur

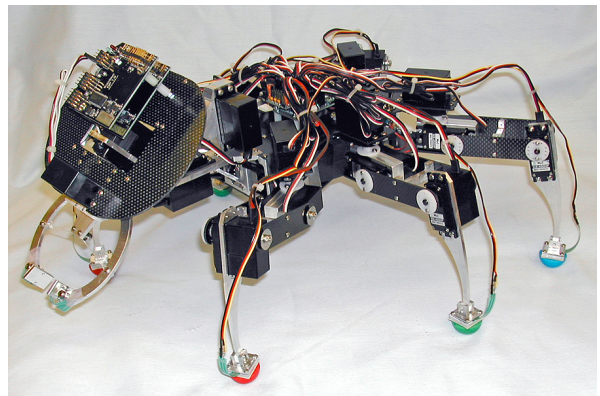


FIG. 2.11 – Bill-Ant

Chapitre 3

Les premiers modèles

Le présent chapitre montre le début des travaux de conception qui ont mené au prototype final du robot Hexapode. Dans ce chapitre sont décrits les critères d'évaluation ayant été utilisés pour le développement du robot. Ensuite, sont présentés les premiers modèles qui ont permis la familiarisation avec les concepts en lien avec le robot Hexapode.

3.1 Description des critères de performance

La conception du robot a été réalisée de façon itérative, c'est-à-dire par des successions de générations d'idées, de conceptions à l'aide de logiciel CAO et d'évaluations des prototypes en comparaison à des critères de performance. Ces critères permettent

de vérifier si les objectifs ont été atteints et conduisent à améliorer les performances du prototype final. Ces critères sont : la mobilité, l'autonomie et l'adaptabilité du robot.

Bien que qualitatifs, ces trois critères ont permis de diriger les itérations du design jusqu'au choix final. Ils sont à la base du développement du prototype du robot Hexapode. Les sections 3.1.1 à 3.1.3 décrivent ces critères et expliquent les résultats atteints ou qui pourraient l'être dans des travaux futurs. Dans la littérature, Alan Bowling s'intéresse à une méthode de quantification de la performance des marcheurs [24]. Cette méthode n'a pas été utilisée dans le cadre du projet puisque son objectif n'était pas d'optimiser les performances du robot mais bien d'explorer et délaborer les concepts menant à la marche.

3.1.1 Mobilité

La mobilité est le critère qui évalue la facilité qu'a le robot de bouger. Il implique la possibilité de se déplacer de façon stable sur diverses surfaces. La démarche doit être la plus fluide possible, en évitant de déstabiliser le robot à chaque pas. Idéalement, le corps du robot doit pouvoir être positionné, orienté et déplacé en tout temps indépendamment des conditions du terrain. La vitesse de déplacement du robot n'est pas un critère de premier plan, cependant elle doit être optimisée selon les capacités de couple et de positionnement des actionneurs, ainsi que le degré de stabilité désiré. L'évaluation de la performance de ce critère passe également par le degré de rugosité de la surface pouvant être franchie. Tant les surfaces très lisses que les surfaces très escarpées présentent des difficultés de nature différentes.

La mobilité implique aussi la possibilité qu'ont les pattes de s'orienter et de se positionner par rapport au corps du robot. Chacune des pattes doit être agile et précise. La redondance de motorisation est aussi à envisager pour permettre de contourner les obstacles et pour atteindre le meilleur point d'appui au prochain pas. L'indépendance des pattes les unes par rapport aux autres est aussi très utile pour permettre de se mouvoir dans diverses configurations. En cas de défaillances partielles, le robot doit pouvoir être le plus opérationnel possible même avec les éléments qui lui manquent.

C'est sur ce critère que repose l'analyse du type de motorisation et de la puis-

sance des actionneurs nécessaires pour permettre au robot de se mouvoir efficacement. L'optimisation du design du châssis et des membres, pour permettre de meilleurs débattements articulaires, est un élément primordial du développement du robot. La sélection des actionneurs et des éléments mécaniques du châssis et des pattes passe donc en partie par ce critère.

3.1.2 Autonomie

Le critère d'autonomie évalue le degré d'aisance du robot à se déplacer par lui-même. Ce critère tient compte de la présence de toutes les composantes nécessaires au déplacement du robot et directement implantées sur celui-ci. Ceci implique la motorisation des pattes, une unité de calcul et de traitement des informations (ordinateur embarqué), des réserves énergétiques et des capteurs permettant au robot d'obtenir les informations nécessaires à la modélisation de son environnement.

Afin que le robot soit autonome, un des éléments primordiaux est la gestion de la consommation énergétique. Le robot doit pouvoir fonctionner avec la meilleure efficacité énergétique possible pour permettre de se déplacer plus longtemps avec une réserve minimale. Les éléments qui consomment le plus d'énergie sont les actionneurs. La force ou le couple à fournir par ceux-ci doit être réduit afin que la puissance nécessaire au robot pour accomplir ses actions soit minimale.

Le design du châssis doit être fait avec attention pour diminuer la masse transportée et pour améliorer les performances de la marche. La géométrie des membres doit également être optimisée. Des mécanismes de compensations doivent être aussi possibles afin de minimiser le couple statique des moteurs lorsque le robot se tient au repos. La nature tend à minimiser la consommation énergétique durant la marche des vivants, il faut donc s'inspirer des morphologies et des mouvements des vivants afin d'atteindre de bonnes performances pour ce critère.

3.1.3 Adaptabilité

Un robot d'exploration doit idéalement être en mesure de se diriger et d'atteindre les objectifs fixés avec un minimum de commandes venant de l'extérieur. Idéalement, le robot devrait être en mesure de se construire une représentation du terrain sur lequel il se trouve. Une forme d'intelligence artificielle pourrait donc être implantée au robot pour lui permettre de prendre les décisions relatives au bien de sa mission. Cette intelligence peut être distribuée sur plusieurs niveaux. Les niveaux supérieurs, impliquant la génération de trajectoire, et la reconnaissance des conditions du terrain sont d'une importance moins primordiale dans la réalisation du projet Hexapode. Les objectifs du présent projet sont de mettre au point une plate-forme de robot marcheur qui aura la possibilité d'être améliorée au fur et à mesure de son développement. Ce qui importe le plus pour le critère de l'adaptabilité est de permettre une paramétrisation du déplacement du robot, soit le contrôle de sa vitesse, de son orientation et de sa stabilité.

L'adaptabilité du robot caractérise sa capacité de modifier sa démarche afin de faire face à des situations jusque alors inconnues. La programmation du contrôleur du robot entre dans ce critère puisque c'est elle qui lui permet de remplir des tâches relativement complexes. Ceci comprend la possibilité pour le robot d'exécuter des démarches s'accordant aux lois de Wilson (décrites en détail à la section 4.5.5.2) et de modifier ses modes de fonctionnement à l'aide de l'intégration de l'intelligence artificielle et du corps du robot. Le robot doit être capable d'ajuster ses comportements aux caprices de l'environnement. L'intelligence artificielle commande ces comportements.

3.2 La fourmi en détail

La fourmi possède une grande adaptabilité, une autonomie prodigieuse et une grande mobilité lui permettant d'affronter des obstacles variés. De plus, la cohabitation sociale avec les autres membres de la fourmilière crée des possibilités hors du commun pour un groupe de fourmis. C'est un insecte évolué, du point de vue morphologique, des sens et de la communication. La fourmi est l'organisme vivant le plus répandu sur Terre, elles représentent plus de dix pour cent de la biomasse [22].

Son corps à exosquelette est composé de trois sections principales (Fig. 3.1) : la tête, le thorax et l'abdomen. Les six pattes sont rattachées sur le thorax et elles sont composées de plusieurs sections. La première, qui fixe la patte sur le thorax, est appelé coxa et serait l'équivalent du bassin si on comparait avec l'être humain. L'articulation entre le coxa et le thorax peut être modélisée par une articulation sphérique, motorisée et solide. Ensuite vient le fémur. Chez l'humain, cet os est le plus gros du corps, la rigidité de cet os est également importante pour les fourmis. La section de membre suivante est le tibia, plus fin et plus mobile. L'articulation entre le fémur et le tibia ressemble à une jonction rotoïde, il faut bien entendu négliger les faibles débattements articulaires dans les autres directions.

Les deux dernières sections de la patte de la fourmi sont le tarse et le prétarse. Ces sections pourraient être comparées au pied et aux orteils chez l'humain. Les articulations reliant le tibia et le tarse et celui-ci avec le prétarse sont de type joint de cardant, cependant leur débattement articulaire est assez faible et d'une nature compliant permettant de s'adapter au terrain sans avoir à fournir trop d'efforts. À l'extrémité de la patte, se trouvent des crochets que la fourmi utilise pour s'accrocher à des objets.

La fourmi est de nature sociale et des différences physiques sont apparentes entre les castes d'individus. On retrouve notamment les ouvrières, les soldats, la reine et les mâles. La vie de chaque fourmi se déroule dans un rôle spécifique, l'équilibre de la colonie dépend donc de tous les individus. Il est possible de parler d'intelligence collective lorsqu'on évalue le potentiel d'une colonie entière de fourmis.

3.3 Première représentation robotisée d'Hexapode

La fourmi vivante remplit à la perfection tous les critères de performance. Elle est la championne de la mobilité, de l'autonomie dans sa démarche et dans son adaptabilité au terrain. Il s'agit donc de construire, avec l'approche biomimétique, un robot qui s'en inspire. C'est ce qui a été fait dans le cadre du développement du robot Hexapode.

La patte d'une fourmi comporte cinq articulations, certaines sont des articulations rotoïdes, d'autres sont des équivalents de joint de cardan ou d'articulations sphériques. Le nombre total de degrés de liberté que contient la patte d'une fourmi est discutable

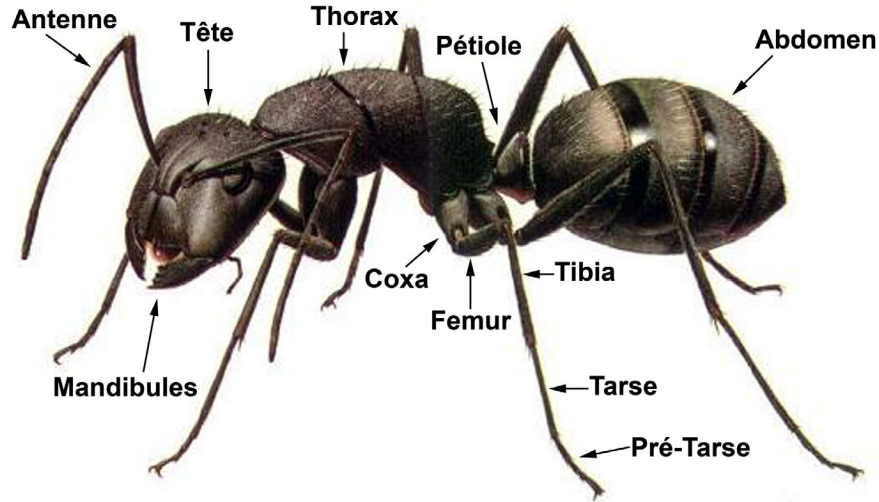


FIG. 3.1 – Anatomie d'une fourmi

puisque les articulations présentent parfois des débattements assez faibles, cependant ce nombre s'approche de neuf degrés de liberté actionnés ou compliants.

Les membres d'une patte peuvent être schématisés plus simplement comme sur la figure 3.2. Il est à remarquer que le tarse et le prétarse n'y sont pas représentés. Sur l'insecte, ils jouent un rôle de stabilisation, de traction et de manutention. Pour le développement du robot, l'hypothèse est que le contact à l'extrémité de la patte est ponctuel et sans glissement (voir la section 1.2.3). Il devient alors possible de ne pas utiliser les sections du tarse et de la prétarse dans le modèle de la patte. Les articulations présentes ont aussi été simplifiées pour mieux permettre de décomposer les mouvements.

Le modèle d'une patte simplifiée est tout de même assez proche du modèle d'une patte réelle et permet l'analogie des mouvements effectués par une fourmi réelle. L'architecture qui résulte de ce premier modèle possède cinq degrés de liberté. L'hypothèse d'un contact ponctuel et sans glissement de l'extrémité de la patte est valide dans des conditions standards de marche pour les fourmis, ce qui a été corroboré par les observations empiriques de Wilson cité plus tôt (section 2.2.1).

L'hypothèse faite permet d'axer ce modèle sur le positionnement de l'extrémité de la patte et de ne pas tenir compte de l'orientation de celle-ci. L'architecture la plus simple pour un mécanisme de positionnement est celle d'un manipulateur en translation à

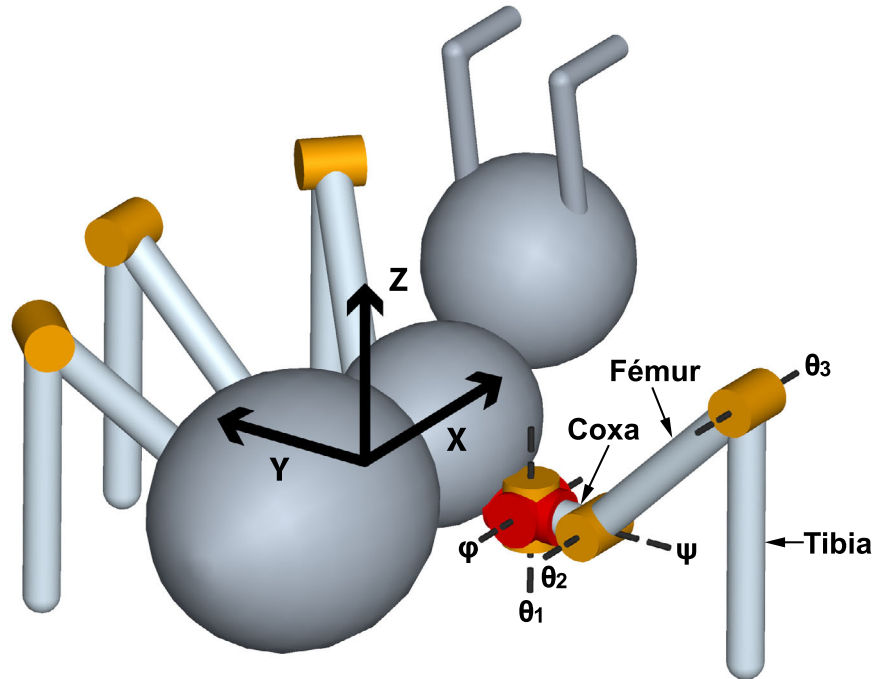


FIG. 3.2 – Schéma d'une patte

trois degrés de liberté, les angles ϕ et ψ sur la figure 3.2 sont alors fixés. Des études sur le chargement des membres et des articulations ont alors été réalisées pour connaître l'impact de la variation de ces angles, plus de détails seront donnés dans le chapitre 4 qui traite du robot final.

La figure 3.3 présente la première ébauche virtuelle d'un modèle envisageable du robot Hexapode. Elle a été réalisée avec le logiciel de CAO Pro-Engineer. Il ne comporte pas d'actionneur et est utilisé pour la visualisation des divers mouvements possibles avec des pattes à cinq degrés de liberté. Il est à noter que les proportions approximatives de la morphologie des fourmis ont été conservées.

Pour marcher, il faut que les extrémités des pattes qui touchent au sol se déplacent selon certaines séquences. Ceci produit la translation du corps. Un premier modèle a été assemblé avec le module « Pro-engineer Mechanism » de façon à ce que les articulations des pattes puissent bouger. Il est possible de déplacer les pattes comme si c'était celles d'un pantin. Une inspection visuelle du mouvement des membres sert à donner l'intuition des mouvements réalisables par les pattes, en vue de déterminer les mouvements menant à la marche.

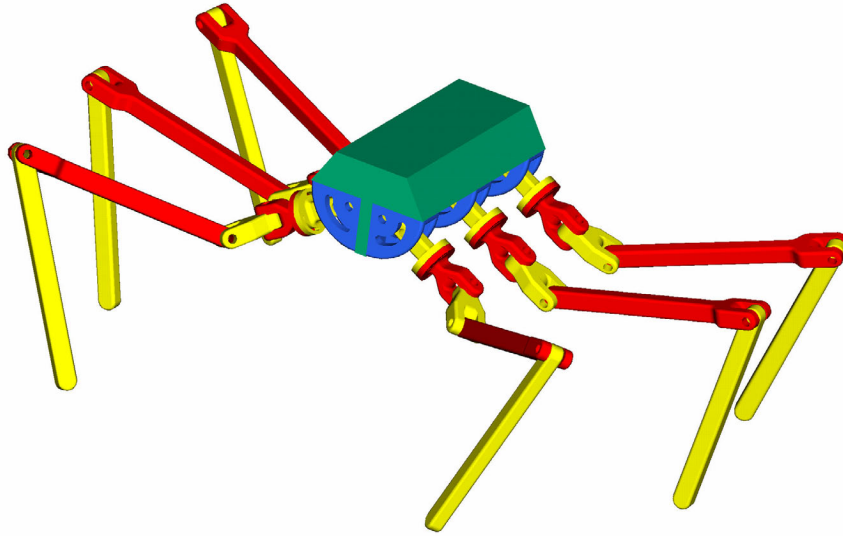


FIG. 3.3 – Première représentation robotisée d'Hexapode

3.3.1 Évaluation selon les critères de performance

Ce premier modèle est utilisé pour explorer le passage de la morphologie de la fourmi dans un contexte robotique. Cette version est davantage utile pour cerner les aspects critiques du design. Elle n'est donc pas apte à répondre de façon efficace aux critères de performance. Cependant grâce à ce modèle, il est possible d'imaginer les mouvements qui sont nécessaires d'imposer aux membres pour déplacer le corps avec une bonne mobilité. Il est également possible d'étudier la possibilité d'interférence entre les pattes ou avec le corps. Du point de vue de l'autonomie, ce modèle cerne le fait qu'une optimisation du châssis et des membres est critique pour la consommation d'énergie. La valeur des angles ϕ et ψ sur la figure 3.2 mène à une répartition des efforts entre les actionneurs, cet aspect sera traité plus en détail à la section 4.4.4. Ce modèle n'étant pas complet, il est donc impossible de le considérer comme pouvant répondre au critère d'adaptabilité puisqu'aucun actionneur, capteur ou contrôleur n'est présent.

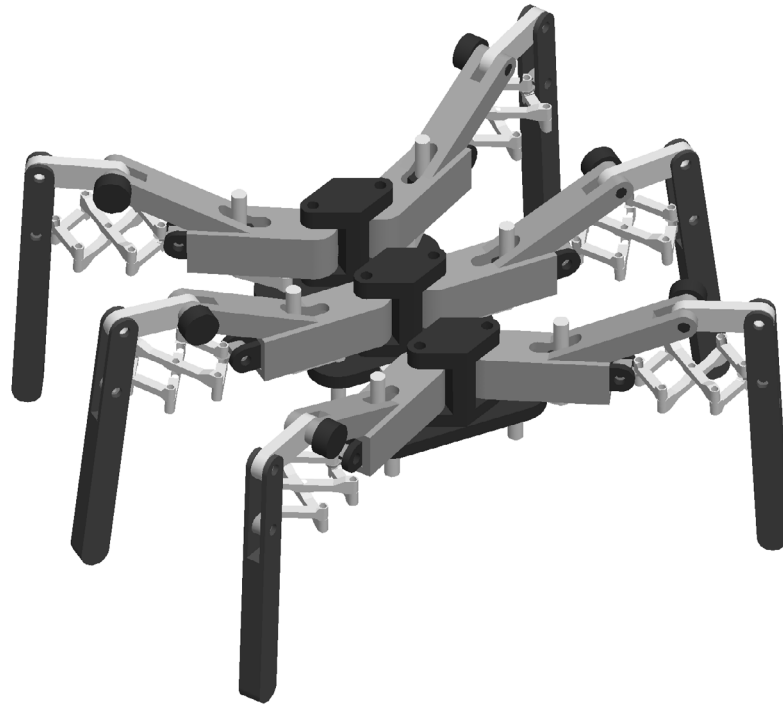


FIG. 3.4 – Prototype utilisant la compensation avec des mécanismes déployables

3.4 Prototype à compensation utilisant un mécanisme déployable

Suite à l'analyse des déplacements des pattes à l'aide du premier modèle, la poursuite des travaux a consisté à vérifier s'il était viable de réduire le nombre de degrés de liberté actionnés présents sur les pattes ou d'en coupler certains. Un nombre moindre de degrés de liberté actionnés par patte signifierait un nombre moindre de moteurs à contrôler et à alimenter en énergie.

Un mécanisme déployable a émergé comme une façon de réduire le nombre de degrés de liberté en permettant l'extension de la patte sans avoir recours à l'articulation θ_3 sur la figure 3.2. Pour que le corps du robot effectue une trajectoire rectiligne et continue, le polygone de support entier doit se déplacer dans une direction et une orientation inverse par rapport à celle désirée du corps. Plus spécifiquement, pour avancer dans

une direction parallèle au corps du robot, la trajectoire à appliquer au polygone de support est une translation parallèle à la direction de la marche. Il faut ensuite relever la patte et rechercher un prochain point de contact qui permettra le support.

L'utilisation d'un mécanisme déployable permettant l'extension de la patte permettrait la correction des arcs de cercle générés par la rotation de l'articulation représentée par l'angle θ_1 (Fig. 3.2). L'idée est de pouvoir générer une trajectoire linéaire en trois dimensions. Ceci implique la gestion de trois degrés de liberté à l'extrémité de la patte avec seulement deux actionneurs.

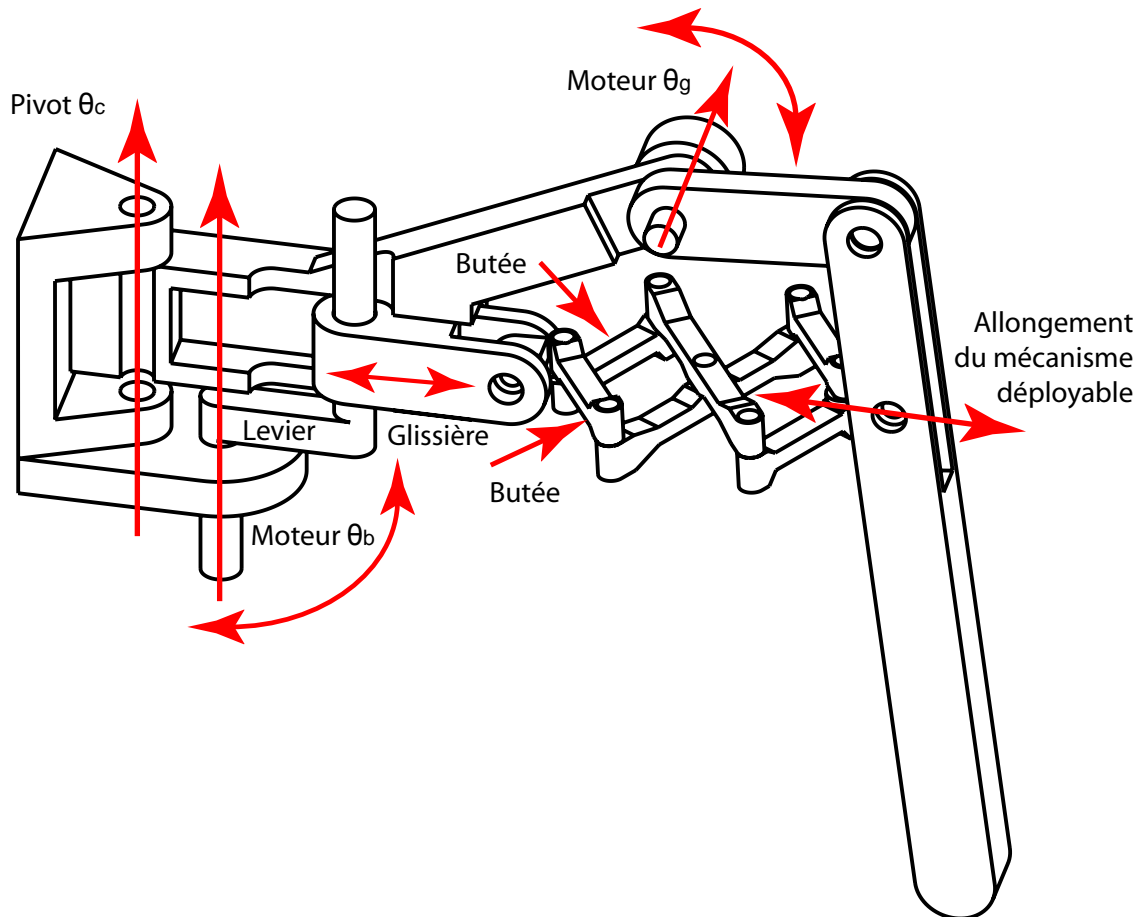


FIG. 3.5 – Liaisons entre les membres

L'extrémité de la patte devrait idéalement pouvoir se déplacer dans un plan qui serait contraint selon l'axe parallèle à la direction de l'avance de la marche et normal au sol. Cette contrainte permettrait à un mécanisme déployable correctement configuré de remplacer un actionneur dans le cas spécifique de la marche dans une seule direction, tout en permettant le changement de la hauteur du point de support. La position et

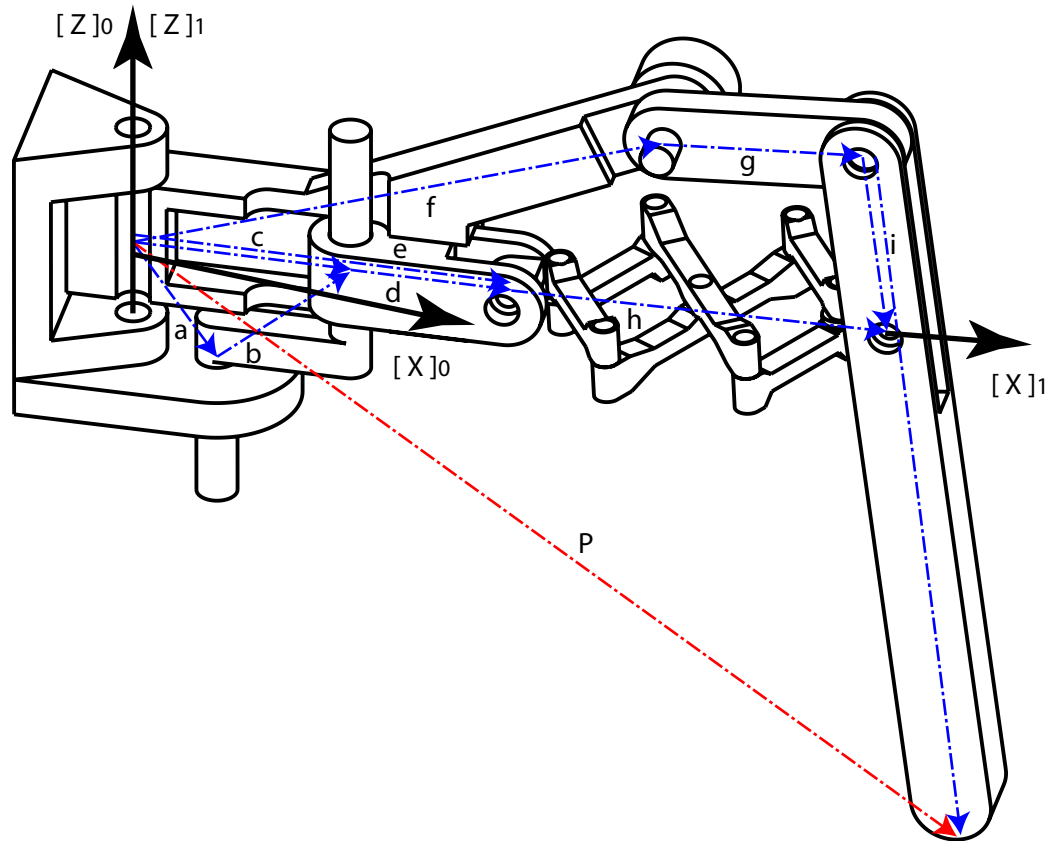


FIG. 3.6 – Représentation vectorielle d'une patte utilisant la compensation

l'orientation du corps seraient donc stables. Un prototype CAO du mécanisme pouvant être utilisé pour parvenir à la réduction du nombre d'actionneurs est présenté à la figure 3.4.

La figure 3.5 montre plus en détail le mécanisme qui a été exploré. L'actionnement horizontal de ce mécanisme se fait par le moteur θ_b qui fait tourner le levier. Ce dernier a deux fonctions, il oriente toute la patte autour du pivot θ_c et pousse la glissière. Ceci a pour effet de changer la longueur du mécanisme déployable puisque ce dernier s'appuie sur les butées et est étiré par des ressorts qui le maintiennent ouvert. Le moteur θ_g est celui qui monte ou descend la patte verticalement.

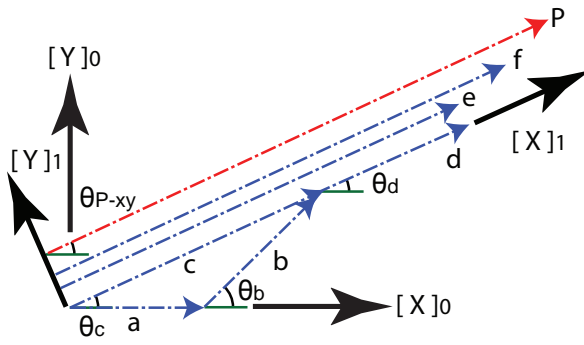


FIG. 3.7 – Repère 1 (Plan X-Y)

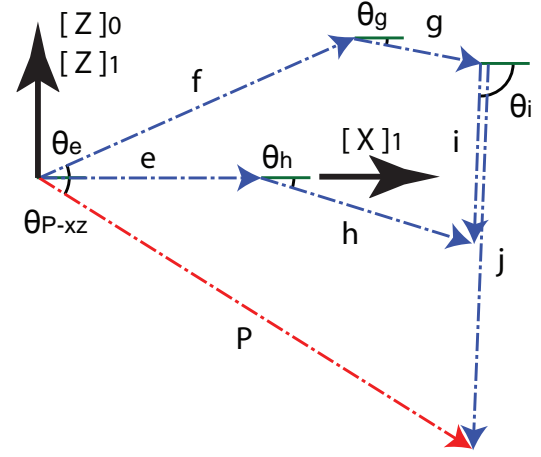


FIG. 3.8 – Repère 1 (Plan X-Z)

3.4.1 Problème géométrique inverse d'une patte

Le problème géométrique inverse, liant les valeurs articulaires motorisées et la longueur du mécanisme déployable en fonction de la position cartésienne de l'extrémité de la patte a été développé. Les figures 3.6, 3.7 et 3.8 montrent la définition des vecteurs qui sont utilisés dans le calcul. Le repère 0 est fixé sur le châssis du robot.

$$\begin{bmatrix} \theta_b & \theta_f & l_h \end{bmatrix} = PGI(\mathbf{P}) \quad (3.1)$$

$$\mathbf{P} = \begin{bmatrix} X & Y & Z \end{bmatrix}^T \quad (3.2)$$

Le vecteur \mathbf{a} part de l'origine et va jusqu'à l'arbre de sortie du premier moteur. Le vecteur \mathbf{a}_{XY} est la projection du vecteur \mathbf{a} dans le plan X-Y. La norme $\|\mathbf{a}\|_{XY}$ est le module de cette projection.

$$\mathbf{a} = \begin{bmatrix} l_a & 0 & -Z_a \end{bmatrix}^T \quad (3.3)$$

$$\|\mathbf{a}\|_{XY} = l_a \quad (3.4)$$

Le vecteur \mathbf{b} est le pivot situé entre l'arbre du moteur et la pièce coulissante qui pousse sur le mécanisme déployable. Le vecteur \mathbf{b}_{XY} est la projection du vecteur \mathbf{b} dans le plan X-Y. La norme $\|\mathbf{b}\|_{XY}$ est le module de cette projection.

$$\mathbf{b} = \begin{bmatrix} l_b \cos \theta_b & l_b \sin \theta_b & Z_a \end{bmatrix}^T \quad (3.5)$$

$$\|\mathbf{b}\|_{XY} = l_b \quad (3.6)$$

Il est d'abord nécessaire de connaître l'angle θ_c afin de définir le vecteur \mathbf{c} qui relie l'origine globale et l'attache de la pièce coulissante. La valeur de cet angle est liée directement à la position de l'extrémité de la patte. En effet, à l'exception du pivot (les vecteurs \mathbf{a} et \mathbf{b}), tous les membres sont dans le plan X-Z du repère mobile 1 qui est normal à l'axe de l'articulation verticale passant par l'origine. Le vecteur \mathbf{c}_{XY} est la projection du vecteur \mathbf{c} dans le plan X-Y. La norme $\|\mathbf{c}\|_{XY}$ est le module de cette projection.

$$\theta_c = \text{atan2} \begin{bmatrix} Y & X \end{bmatrix} \quad (3.7)$$

$$\mathbf{c} = \begin{bmatrix} l_c \cos \theta_c & l_c \sin \theta_c & 0 \end{bmatrix}^T \quad (3.8)$$

$$\|\mathbf{c}\|_{XY} = l_c \quad (3.9)$$

Connaissant cet angle, il devient alors possible de calculer la distance entre le point de pivot et le point d'attache du membre C. On sait que vectoriellement :

$$\mathbf{b} = \mathbf{c} - \mathbf{a} \quad (3.10)$$

En remaniant l'équation précédante il devient alors possible de trouver sous forme scalaire :

$$l_c^2 + (2l_a \cos \theta_c)l_c + (l_a^2 - l_b^2) = 0 \quad (3.11)$$

Cette équation quadratique donne deux solutions pour l_c . On utilise la branche de solution négative puisque on cherche la valeur qui correspond à l'hypothénuse la plus courte : on cherche une valeur angulaire de π lorsqu'on impose une valeur articulaire θ_b de zéro. On cherche alors cette valeur articulaire θ_b . On la trouve en repartant de l'équation vectorielle générale :

$$\mathbf{c} = \mathbf{a} + \mathbf{b} \quad (3.12)$$

En remaniant les équations scalaires émergeant de l'équation vectorielle précédante, on obtient :

$$\cos \theta_b = \frac{l_c^2 - l_a^2 - l_b^2}{2l_a l_b}; \quad \sin \theta_b = \frac{l_c \sin \theta_c}{l_b} \quad (3.13)$$

$$\theta_b = \text{atan2} \begin{bmatrix} \cos \theta_b & \sin \theta_b \end{bmatrix} \quad (3.14)$$

Il est alors possible de définir les vecteurs suivants :

$$\mathbf{d} = \begin{bmatrix} l_d \cos \theta_c & l_d \sin \theta_c & 0 \end{bmatrix}^T; \quad \|\mathbf{d}\|_{XY} = l_d \quad (3.15)$$

$$\mathbf{e} = \mathbf{a} + \mathbf{b} + \mathbf{d} = \begin{bmatrix} l_e \cos(\theta_c) & l_e \sin(\theta_c) & 0 \end{bmatrix}^T; \quad \|\mathbf{e}\|_{XY} = l_e \quad (3.16)$$

$$\mathbf{f} = \begin{bmatrix} l_f \cos(\theta_c) & l_f \sin(\theta_c) & Z_f \end{bmatrix}^T; \quad \|\mathbf{f}\|_{XY} = l_f \quad (3.17)$$

Le repère mobile 2 est lié au plan de la patte. Ce repère permet de découpler les rotations afin de résoudre les équations selon les coordonnées X et Z seulement. Pour travailler dans ce repère il est utile d'y ramener les vecteurs \mathbf{P} , \mathbf{e} et \mathbf{f} .

$$\mathbf{Q}_{\theta_c} = \begin{bmatrix} \cos \theta_c & -\sin \theta_c & 0 \\ \sin \theta_c & \cos \theta_c & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.18)$$

On obtient alors :

$$\mathbf{P}^* = \mathbf{Q}_{\theta_c}^T \mathbf{P} \quad (3.19)$$

$$\mathbf{e}^* = \mathbf{Q}_{\theta_c}^T \mathbf{e} = \begin{bmatrix} l_e & 0 & 0 \end{bmatrix}^T; \quad \|\mathbf{e}^*\| = l_e \quad (3.20)$$

$$\mathbf{f}^* = \mathbf{Q}_{\theta_c}^T \mathbf{f} = \begin{bmatrix} l_f & 0 & Z_f \end{bmatrix}^T \quad (3.21)$$

Les vecteurs qui suivent forment un mécanisme à quatre barres.

$$\mathbf{g} = \begin{bmatrix} l_g \cos \theta_g & l_g \sin \theta_g & 0 \end{bmatrix}^T; \quad \|\mathbf{g}\| = l_g \quad (3.22)$$

$$\mathbf{h} = \begin{bmatrix} l_h \cos \theta_h & l_h \sin \theta_h & 0 \end{bmatrix}^T; \quad \|\mathbf{h}\| = l_h \quad (3.23)$$

$$\mathbf{i} = \begin{bmatrix} l_i \cos \theta_i & l_i \sin \theta_i & 0 \end{bmatrix}^T; \quad \|\mathbf{i}\| = l_i \quad (3.24)$$

$$\mathbf{j} = \begin{bmatrix} l_j \cos \theta_j & l_j \sin \theta_j & 0 \end{bmatrix}^T \quad (3.25)$$

Il est également possible de représenter ce dernier vecteur comme la combinaison des autres.

$$\mathbf{j} = \mathbf{P}^* - \mathbf{f}^* - \mathbf{g}; \quad \|\mathbf{j}\| = l_j \quad (3.26)$$

En remaniant l'équation, il est alors possible d'obtenir :

$$2(\mathbf{P}^* - \mathbf{f}^*)^T \mathbf{g} + l_j^2 - l_g^2 - \|\mathbf{P}^* - \mathbf{f}^*\|^2 = 0 \quad (3.27)$$

Il est possible de ramener l'équation précédente sous la forme :

$$A \cos \theta_g + B \sin \theta_g + C = 0 \quad (3.28)$$

Où

$$A = 2(\mathbf{P}^* - \mathbf{f}^*)^T \begin{bmatrix} 1 \\ 0 \end{bmatrix} l_g \quad (3.29)$$

$$B = 2(\mathbf{P}^* - \mathbf{f}^*)^T \begin{bmatrix} 0 \\ 1 \end{bmatrix} l_g \quad (3.30)$$

$$C = l_j^2 - l_g^2 - \|\mathbf{P}^* - \mathbf{f}^*\|^2 \quad (3.31)$$

Il est alors possible d'obtenir une équation sous la forme quadratique en utilisant les substitutions suivante :

$$\cos \theta_g = \frac{1 - t_g^2}{1 + t_g^2}; \quad \sin \theta_g = \frac{2t_g}{1 + t_g^2}; \quad t_g = \tan\left(\frac{\theta_g}{2}\right) \quad (3.32)$$

$$(C - A)T^2 + (2B)T + (C + A) = 0 \quad (3.33)$$

On peut donc résoudre l'équation pour deux solutions pour T et donc deux solutions pour θ_g . Il faut utiliser la branche positive de l'équation afin d'obtenir un angle θ_i qui soit compris entre $\frac{5}{8}\pi$ et $\frac{7}{8}\pi$. Il faut ensuite trouver l'angle θ_j à partir de l'équation générale :

$$\mathbf{j} = \mathbf{P}^* - \mathbf{f}^* - \mathbf{g} \quad (3.34)$$

Ce qui conduit à une solution unique pour l'angle θ_j :

$$\cos \theta_j = \frac{X - l_f - l_g \cos \theta_g}{l_j}; \quad \sin \theta_j = \frac{Z - Z_f - l_g \sin \theta_g}{l_j}; \quad (3.35)$$

$$\theta_j = \arctan(\sin \theta_j, \cos \theta_j) \quad (3.36)$$

Les vecteurs \mathbf{i} et \mathbf{j} sont sur la même membrure, il est donc possible de poser :

$$\mathbf{i} = \frac{l_i}{l_j} \mathbf{j} \quad (3.37)$$

$$\mathbf{i} - \mathbf{j} = \left(\frac{l_i}{l_j} - 1\right) \mathbf{j} \quad (3.38)$$

Il faut finalement résoudre pour trouver la longueur que devrait avoir la membrure déployable h.

$$\mathbf{h} = \mathbf{P} - \mathbf{e}^* - \left(\frac{l_i}{l_j} - 1\right) \mathbf{j} \quad (3.39)$$

Et donc :

$$l_h = \sqrt{\|\mathbf{P}^* - \mathbf{e}^*\|^2 + \left(1 - \frac{l_i}{l_j}\right)^2 l_j^2 - 2(\mathbf{P}^* - \mathbf{e}^*)^T \left(\frac{l_i}{l_j} - 1\right) \mathbf{j}} \quad (3.40)$$

À la lumière de ce qui précède, il est donc possible de connaître les valeurs articulaires et la longueur du mécanisme nécessaire au positionnement de l'extrémité de la patte. L'équation (3.40) montre que la longueur l_h est fonction de la position de

l'extrémité de la patte, mais aussi de \mathbf{e} et de \mathbf{j} qui sont eux même fonction de la position de l'extrémité de la patte. La façon de changer la longueur du mécanisme déployable l_h passe par le déplacement selon l'articulation à glissière de la membrure intermédiaire représentée par le vecteur \mathbf{d} . En tirant sur le mécanisme déployable, il s'allonge et pousse donc sur le tibia. Cette compensation permet de gérer approximativement les erreurs de positionnement dans le plan XY du repère fixé au corps du robot. La variation de la hauteur selon l'axe Z ferait intervenir un changement progressif de la géométrie de la butée qui agrandirait ou rétrécirait le mécanisme déployable selon les changements de l'angle θ_g , soit l'angle que fait le mécanisme déployable avec le plan XY du repère fixé au corps du robot.

Une géométrie de butée est donc spécifique à chacune des trajectoires désirées du bout de la patte puisque la longueur du mécanisme déployable et la valeur articulaire imposée aux actionneurs déterminent la position finale de l'extrémité de la patte.

3.4.2 Évaluation selon les critères de performance

La réduction du nombre de degrés de liberté actionnés permet de diminuer le nombre d'actionneurs par patte. De façon théorique, il serait donc envisageable d'augmenter l'autonomie du robot puisque sa consommation énergétique diminuerait. Cependant, ceci ne se ferait pas sans impact sur la mobilité et l'adaptabilité du robot puisque les trajectoires pouvant être imposées aux pattes ne pourraient être que celles inscrites dans un plan choisi lors de la construction du robot. Rien n'indique que les environnements que rencontrera le robot lors de ses promenades en terrain inconnu seront compatibles avec ce style de déplacement de l'extrémité des pattes. Ce type de mécanisme de compensation utilisant un mécanisme déployable pour allonger la patte n'est donc pas très performant au niveau des critères d'adaptabilité et de mobilité.

Si l'analyse de ce mécanisme était extrapolée au niveau de la fabrication mécanique, alors les résultats répondraient encore moins bien aux critères de performance. La friction est inévitable entre deux pièces en contact. Pour que le mécanisme de compensation soit avantageux, il faut que les avantages présents au niveau de la diminution de la consommation énergétique et de la simplification des contrôles dépassent ses limites au niveau du déplacement restreint de l'extrémité des pattes. Or, la butée à

géométrie variable qui devrait permettre la modification de la longueur du mécanisme déployable présentera inévitablement de la friction et donc une perte énergétique. Pour que le mécanisme puisse fonctionner, il serait nécessaire d'utiliser des ressorts pour permettre au mécanisme déployable d'être étendu et de voir sa dimension changer avec la pression de la butée. La tension dans les ressorts maintenant le mécanisme déployable de la bonne dimension résulte en des contraintes internes qu'il faut vaincre. Au bout du compte, cette pression augmente la friction et dans toute autre position que celle d'équilibre (orthogonale) qui celle où le ressort est détendu.

De plus, la friction dans le mécanisme déployable lui-même n'est pas à négliger. En plus, celle présente dans la glissière qui tire sur ce mécanisme devrait être également non négligeable. Il faut aussi considérer le fait que la transmission du couple permettant une rotation de toute la patte autour de son axe de fixation n'est pas directe. Il en résulte encore une perte de puissance dans les éléments de transmission.

La diminution de masse résultant de l'utilisation d'un nombre réduit de moteurs, est complètement éclipsée par la masse à ajouter par le mécanisme de compensation lui-même. De plus, le fait de générer des trajectoires linéaires sur un plan est de peu d'intérêt, car il n'est qu'un des cas rencontrés dans la marche sur terrain encombré. Les polygones de contact générés avec cette architecture ne peuvent effectuer que des translations dans une seule direction et ne peuvent effectuer de rotation. Ceci amène une contre-indication au niveau de la mobilité et de l'adaptabilité qui ne peut être négligée.

Après avoir analysé la complexité du mécanisme. Un design à trois degrés de liberté permettant le plein positionnement est préférable. C'est sur la base de cette analyse que le prototype final du robot Hexapode a été conçu. C'est l'objet du chapitre suivant.

Chapitre 4

Hexapode le robot

Dans ce chapitre est présenté le prototype final du robot marcheur : Hexapode. Le prototype de patte de validation utilisé pour vérifier l'architecture de la patte, le choix des actionneurs et le contrôleur implanté dans l'unité de calcul sera aussi présenté. L'organisation du chapitre montre le développement du robot par rapport aux critères de performance, soit la mobilité, l'autonomie et l'adaptabilité. Il sera clairement démontré que le cheminement pour parvenir à la version finale d'Hexapode à permis d'atteindre, avec satisfaction, la majeure partie des objectifs fixés.

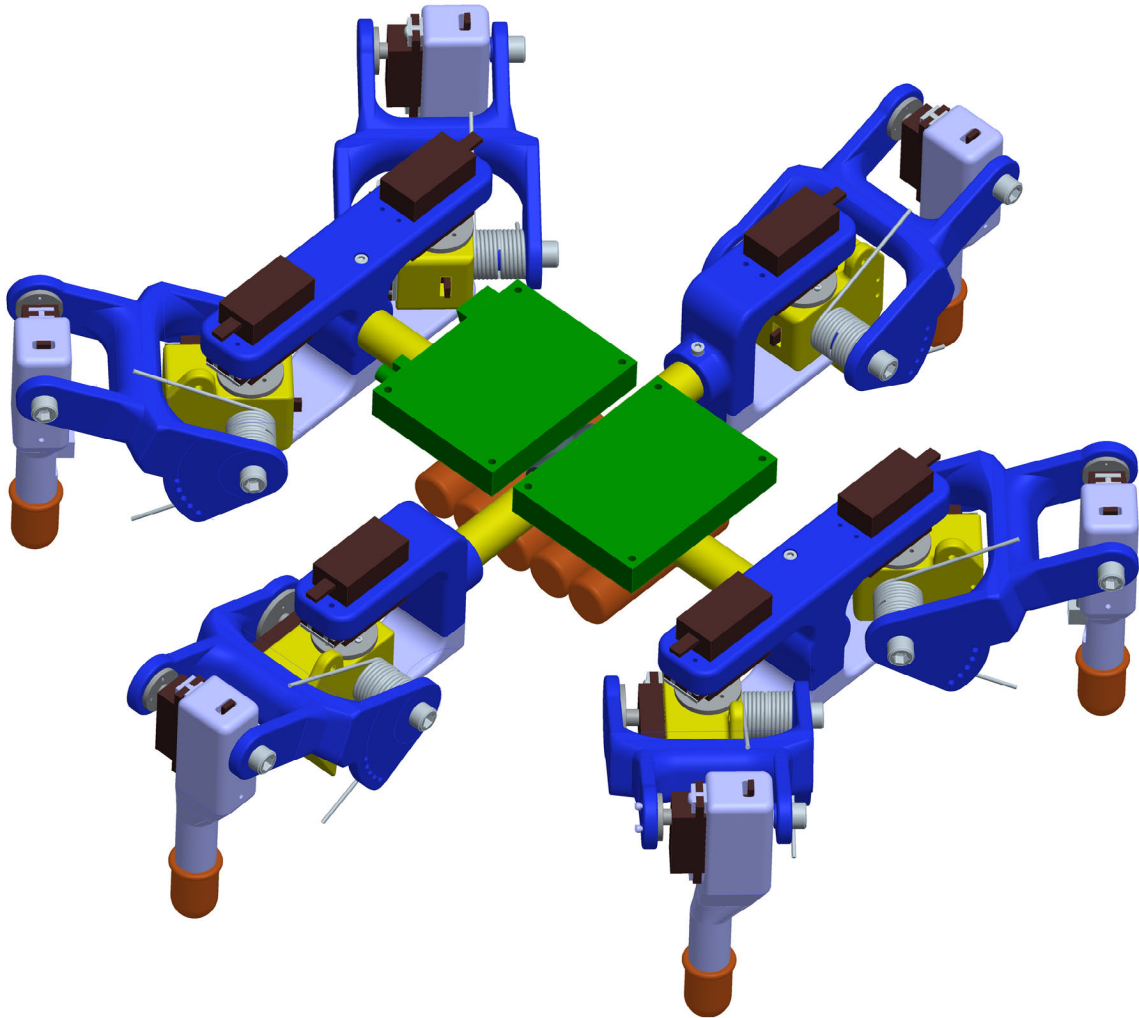


FIG. 4.1 – Modèle CAO d'Hexapode

4.1 Prototypes

4.1.1 Hexapode

Pour situer le lecteur par rapport aux résultats, une image du prototype final sous sa version CAO est disponible à la figure 4.1 et une photo du modèle réel à la figure 4.2. Les explications sur les divers aspects du développement se retrouvent dans les sections suivantes du présent chapitre.

Suite à sa construction et sa programmation, Hexapode est capable de se déplacer. Il

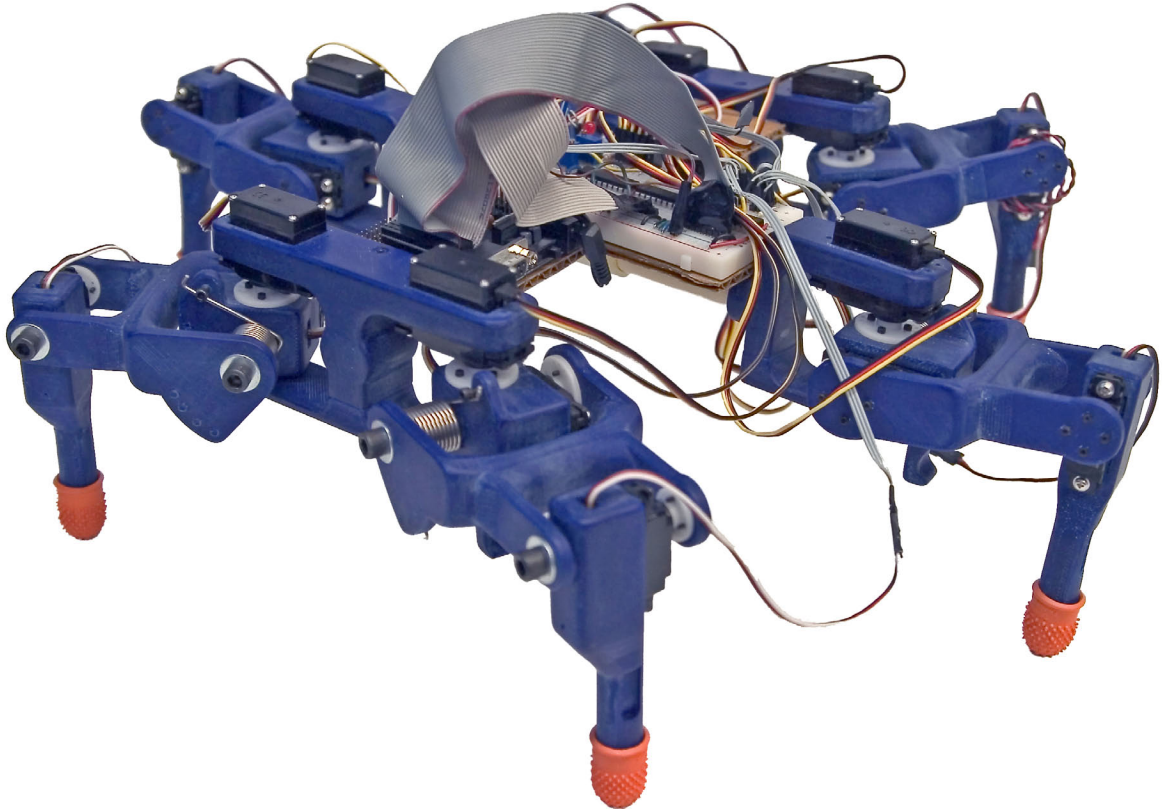


FIG. 4.2 – Prototype Final

a donc été possible dans le projet d’explorer divers aspects de la marche, ce qui remplit donc l’objectif primaire du projet. Comme on verra, il sera démontré qu’Hexapode obtient une bonne évaluation par rapport aux critères de performance.

4.1.2 Conception d’une patte de validation

Puisqu’un robot marcheur est avant tout composé de pattes, la conception de celles-ci est critique. C’est pourquoi il est pertinent de construire d’abord une patte dite de validation. La figure 4.3 montre le prototype de la patte de validation qui a été utilisé pour vérifier les concepts clés avant la construction du prototype final. Cette patte test a servi à valider l’architecture de patte, le type d’actionneur et l’unité de calcul du prototype final. Les conclusions tirées de la patte de validation ont donc mené au raffinement des concepts y étant reliés, en vue de la construction des pattes finales d’Hexapode. Ces éléments ont un impact sur l’ensemble du design d’Hexapode, c’est pourquoi ils ont été spécifiquement explorés à l’aide de la patte de validation.

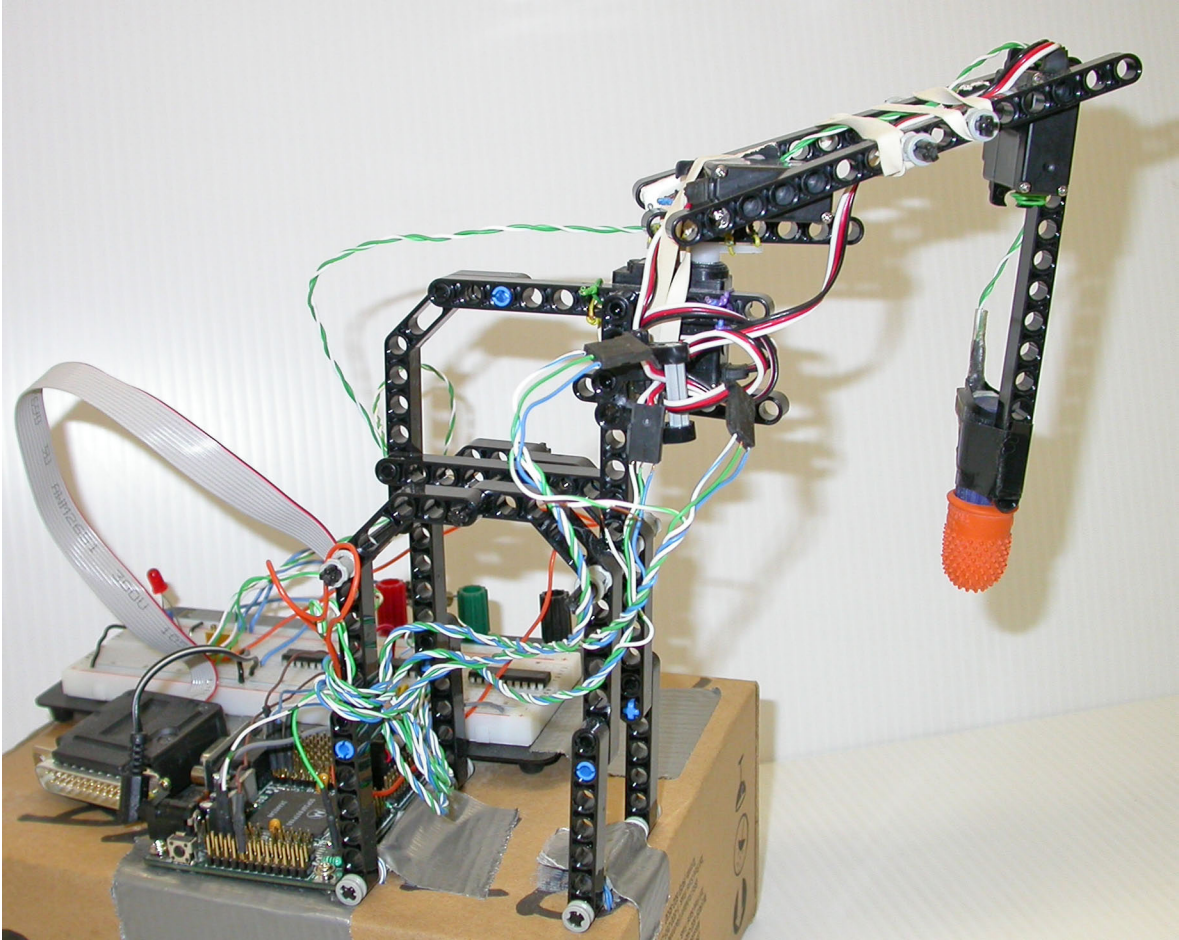


FIG. 4.3 – Prototype de patte en Lego

L'architecture des pattes est celle d'un manipulateur cartésien à trois degrés de liberté. Ce paramètre du design influence au niveau de la cinématique et de la dynamique du robot. L'architecture utilisée pour la patte de validation ainsi que pour le prototype final sera discutée à la section 4.2.1 qui traite de la performance par rapport au critère de mobilité.

Le choix des actionneurs est également une étape critique du développement. Le mode de fonctionnement, ses dimensions, sa puissance ainsi que son rendement énergétique sont des caractéristiques qui ont un impact important sur l'atteinte des objectifs évalués par les critères de performance. Les actionneurs qui permettent la motorisation des deux prototypes sont décrits à la section 4.2.2. Les moteurs seront contrôlés par une unité de calcul.

Le choix de l'unité de calcul pèse également lourd dans la balance puisque ce dernier

apportera des possibilités et des limites au niveau de la possible intelligence artificielle du robot ainsi que sur la coordination des pattes qui sont mues par les actionneurs qui doivent être contrôlés individuellement. L'unité de calcul qui servira de cerveau au prototype est discutée à la section 4.5.2.

4.2 Mobilité

La mobilité est le critère qui tient compte de la facilité et du moyen qu'a le robot de bouger. Il faut que la marche s'effectue de façon stable dans toutes les directions et même sur une surface non uniforme. L'environnement dans lequel naviguera le robot peut comporter des surfaces lisses ou rugueuses, planes ou à relief.

Le critère de mobilité fait le focus sur les aptitudes physiques du robot à marcher dans toutes conditions. En utilisant l'approche biomimétique, la morphologie des pattes est basée sur une version simplifiée de celles qui se retrouvent chez les insectes. Dans un cadre robotique, on parle de l'architecture du mécanisme pour référer à la morphologie. L'architecture retenue comporte trois degrés de liberté par patte. Ces degrés de liberté sont actionnés par des moteurs qui doivent être contrôlés pour réagir à la consigne en fonction de la position désirée. Les membres des pattes ainsi que le corps du robot doivent être développés afin de favoriser la meilleure mobilité. Leur design mécanique doit être fait avec grand soin pour minimiser leur masse, maximiser leur rigidité et éliminer les interférences géométriques dans la plage de débattement articulaire choisie.

4.2.1 Architecture à trois degrés de liberté par patte

Comme traitée dans la section 1.2.3 du premier chapitre, la modélisation des points de contact avec le sol est simplifiée. Les contraintes prises en compte sont celles de position seulement, les valeurs d'orientation sont négligées, c'est le cas de la modélisation des contacts ponctuels qui est utilisé. Pour imposer une position dans un espace cartésien, trois contraintes sont requises.

Au chapitre précédent, des conclusions ont été tirées de l'étude du prototype à

compensation par mécanisme déployable. Il a été jugé que le couplage de degrés de liberté au niveau des actionneurs composant une patte ne menait pas à une solution viable pour le prototype élaboré. Cependant, l'avenue du couplage des degrés de liberté entre les pattes reste à explorer.

La position de l'extrémité des pattes doit donc être contrôlée de façon individuelle et dans toutes les directions. La plate-forme actuelle permettra de faire l'étude des divers mouvements nécessaires pour exécuter des démarches stables. Il sera donc possible dans de futurs travaux, d'imposer de façon virtuelle des couplages entre les moteurs de diverses pattes et de tester l'efficacité résultante. Le prototype développé propose un ensemble de mécanismes où chacune des pattes est autonome par rapport aux autres. C'est donc une première itération dans l'étude de la marche au laboratoire de robotique de l'Université Laval.

En revenant au modèle de patte le plus simple, qui permet des déplacements cartésiens (figure 3.2), il devient alors justifiable de tenter de construire un prototype de patte utilisant une architecture semblable. Trois degrés de liberté actionnés par patte, soit les angles θ_1 , θ_2 et θ_3 sur la figure 3.2, serviront au positionnement de l'extrémité de la patte.

La présence de trois degrés de liberté sur chacune des six pattes est requise pour grouper les extrémités des pattes dans un polygone de support et un polygone de retour. Chacun des points composant le polygone peut être positionner cartésienement. Puisque l'hypothèse de points de contact a été posée (discuté à la section 1.2.3), l'orientation de la patte n'est pas prise en compte. Ce qui est alors important est de pouvoir positionner ces point de façon indépendante.

Il est alors possible de générer des translations et des rotations au polygone de support sans le déformer. Cette condition est nécessaire pour la validation de la théorie de polygones. Les trajectoires des polygones qui en résultent permettent au corps de se déplacer selon la direction et l'orientation requise pour les conditions environnementales extérieures. Ceci permet d'assurer une démarche stable dans tous les cas où les points de contact nécessaires sont à l'intérieur de l'espace de travail des pattes et qu'il n'y a pas de glissement à ce point de contact. Tous les polygones, de support ou de retour, composés d'un nombre variable de points de contact, peuvent alors être déplacés et orientés. Il est possible, en tout temps, d'imposer le contrôle sur six degrés de liberté

au polygone.

Le robot complet comporte six pattes ayant chacune trois degrés de liberté. Au total, dix-huit degrés de liberté actionnés constituent la plate-forme reproduisant la marche. Une démarche tripode (voir à la section 4.5.5.2 pour la définition détaillée) est celle qui assure une stabilité tout en comportant le moins de pattes simultanément en contact avec le sol.

Avec trois pattes au sol, neuf actionneurs, sont utilisés pour permettre les six degrés de liberté en orientation et en translation du corps par rapport au polygone de support. Les contacts ponctuels au sol sont modélisés comme des liaisons à trois degrés de liberté. Neuf contraintes de positionnement des pattes sont présentes dans le cas de la démarche tripode. Il y a donc redondance de trois contraintes. Il serait alors possible de produire le même déplacement avec trois degrés de liberté motorisés de moins.

Le polygone de support ne doit pas être déformé puisqu'il en résulterait une diminution de la stabilité. Dans le cas d'une perte de contact ou de glissement de l'une des pattes, les actionneurs redondants permettent la déformation momentanée du polygone de support pour retrouver la stabilité. Lors d'une déformation, un nouveau polygone de support pouvant supporter toutes les contraintes du corps doit être généré pour éviter la chute. Comme chacune des pattes constituant le polygone de support peut être à l'origine de la perte de contrainte, il importe de pouvoir repositionner n'importe quelle d'entre elle. Ces degrés de libertés redondants sont donc justifiés. Il est aussi utile de pouvoir connaître l'état du contact au sol et d'en maximiser la qualité (discuté à la section 4.5.1.2).

Le fait d'avoir des actionneurs redondants permet aussi d'être plus tolérant aux bris. En effet, dans le cas qu'un des actionneurs se bloque ou devienne incontrôlable, il serait possible de reconfigurer les conditions de base des algorithmes de marche. Il en résulterait des démarches moins stables, mais tout de même capables de produire un déplacement du corps. Ceci est particulièrement utile pour des situations où il serait complexe, voire impossible, d'effectuer les réparations nécessaires pour remettre le robot en plein état de fonctionnement. Ce genre de situation se présenterait dans un contexte d'exploration spatiale ou dans un environnement terrestre où le robot n'est pas accessible ou ne peut faire marche arrière.

Tout comme le positionnement des membres du robot, le contrôle du déplacement de chacune des pattes se fait de façon indépendante. Ceci permet d'augmenter grandement les possibilités de mouvements. L'impact du choix des angles ϕ et ψ de la figure 3.2 sera discuté dans la section 4.4.4. Pour l'instant, il faut considérer que ces valeurs d'angle sont fixées à zéro degré, pour une fixation de la patte possédant le premier axe selon une direction parallèle à l'axe vertical Z du corps du robot. De ce fait, l'architecture des six pattes est donc identique, permettant l'homogénéité du contrôle de toutes les pattes de la plate-forme.

4.2.1.1 Résolution du problème géométrique inverse

Contrairement aux autres robots marcheurs rencontrés dans la littérature, le contrôle des pattes d'Hexapode s'effectue sur la position cartésienne de l'extrémité de la patte. Ceci a pour conséquence de simplifier grandement le contrôle des trajectoires du robot en passant par la gestion des polygones. L'architecture qui a été retenue est celle permettant la plus grande simplification des calculs puisque ceux-ci doivent être résolus en temps réel sur une unité de calcul embarquée sur le robot. Les deux premiers axes sont donc coïncidents (longueur $a_1 = 0$), ceci permet de diminuer le nombre de solutions possibles à la résolution du problème géométrique inverse.

Un résumé des procédures menant à la résolution de la cinématique d'une patte est présenté ci-après. La représentation sous forme de paramètres Hartenberg-Denavit a été utilisée. Ceci a pour but de permettre une solution générique pour les manipulateurs sériels à trois degrés de liberté. Il faut alors trouver les valeurs articulaires associées à une position cartésienne du bout de la patte.

Les débattements articulaires désirés sur les articulations sont de l'ordre de $\frac{\pi}{2}$. Les valeurs articulaires limites et les valeurs des paramètres HD sont présentées dans les tableaux suivants.

	Pattes de gauche	Pattes de droite
θ_1	$[\frac{3\pi}{4}; \frac{5\pi}{4}]$	$[\frac{-\pi}{4}; \frac{\pi}{4}]$
θ_2	$[\frac{-\pi}{4}; \frac{\pi}{4}]$	$[\frac{-\pi}{4}; \frac{\pi}{4}]$
θ_3	$[\frac{-3\pi}{4}; \frac{-\pi}{4}]$	$[\frac{-3\pi}{4}; \frac{-\pi}{4}]$

TAB. 4.1 – Limites articulaires

	a	b	α	θ
Coxa - 1	0	0	$\frac{\pi}{2}$	θ_1
Fémur - 2	a_2	0	0	θ_2
Tibia - 3	a_3	0	0	θ_3

TAB. 4.2 – Paramètres HD

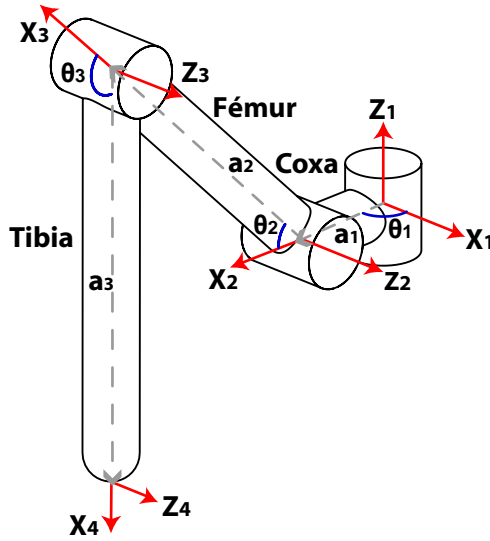


FIG. 4.4 – Forme schématique d'une patte

$$[\theta_1 \ \theta_2 \ \theta_3] = PGI [X, Y, Z] \quad (4.1)$$

Vecteur de position du coxa (voir figure 4.4) :

$$[\mathbf{a}_1]_1 = \mathbf{a}_1 = [0 \ 0 \ 0]^T \quad (4.2)$$

Matrice de rotation du coxa par rapport au corps du robot :

$$[\mathbf{Q}_1]_1 = \mathbf{Q}_1 = \begin{bmatrix} \cos \theta_1 & 0 & \sin \theta_1 \\ \sin \theta_1 & 0 & -\cos \theta_1 \\ 0 & 1 & 0 \end{bmatrix} \quad (4.3)$$

Vecteur de position du fémur (voir figure 4.4) :

$$[\mathbf{a}_2]_2 = \mathbf{a}_2 = [a_2 \cos \theta_2 \ a_2 \sin \theta_2 \ 0]^T \quad (4.4)$$

Matrice de rotation du fémur par rapport au coxa :

$$[\mathbf{Q}_2]_2 = \mathbf{Q}_2 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 \\ \sin \theta_2 & \cos \theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

Vecteur de position du tibia (voir figure 4.4) :

$$[\mathbf{a}_3]_3 = \mathbf{a}_3 = [a_3 \cos \theta_3 \ a_3 \sin \theta_3 \ 0]^T \quad (4.6)$$

Matrice de rotation du le tibia par rapport au fémur :

$$\left[\mathbf{Q}_3 \right]_3 = \mathbf{Q}_3 = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 \\ \sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.7)$$

Le vecteur liant l'extrémité de la patte à l'origine peut donc être écrit sous la forme suivante :

$$\mathbf{P} = \mathbf{a}_1 + \mathbf{Q}_1 \mathbf{a}_2 + \mathbf{Q}_1 \mathbf{Q}_2 \mathbf{a}_3 = \begin{bmatrix} X & Y & Z \end{bmatrix}^T \quad (4.8)$$

En remaniant les termes, l'équation devient :

$$\mathbf{a}_2 + \mathbf{Q}_2 \mathbf{a}_3 = \mathbf{Q}_1^T (\mathbf{P} - \mathbf{a}_1) \quad (4.9)$$

En représentant toutes les composantes des vecteurs :

$$\begin{bmatrix} a_2 \cos \theta_2 + a_3 \cos \theta_2 \cos \theta_3 - a_3 \sin \theta_2 \sin \theta_3 \\ a_2 \sin \theta_2 + a_3 \sin \theta_2 \cos \theta_3 - a_3 \cos \theta_2 \sin \theta_3 \\ 0 \end{bmatrix} = \begin{bmatrix} X \cos \theta_1 + Y \sin \theta_1 \\ Z \\ X \sin \theta_1 + Y \cos \theta_1 \end{bmatrix} \quad (4.10)$$

Il devient alors possible d'isoler θ_2 dans l'équation précédente :

$$\cos \theta_2 = \frac{(a_2 + a_3 \cos \theta_3)(X \cos \theta_1 + Y \sin \theta_1)}{a_2^2 + 2a_2 + a_3 \cos \theta_3 a_3^2} - a_3 Z \sin \theta_3 \quad (4.11)$$

$$\sin \theta_2 = \frac{-a_3 \sin \theta_3 (X \cos \theta_1 + Y \sin \theta_1)}{a_2^2 + 2a_2 + a_3 \cos \theta_3 a_3^2} + (a_2 + a_3 \sin \theta_3) Z \quad (4.12)$$

Ces dernières équations serviront à résoudre pour θ_2 lorsque les autres valeurs d'angle seront trouvées. Il est alors possible de substituer les résultats dans les équations précédentes. En mettant au carré les termes des lignes un et deux de l'équation (4.10) et en les additionnant, on obtient :

$$2a_2 a_3 \cos \theta_3 + a_2^2 + a_3^2 - X^2 - Y^2 - Z^2 = 0 \quad (4.13)$$

Ou plus simplement on trouve :

$$\cos \theta_3 = \frac{-a_2^2 - a_3^2 + X^2 + Y^2 + Z^2}{2a_2 a_3} = 0 \quad (4.14)$$

Deux branches de solutions émergent de cette résolution. Les contraintes mécaniques et les interférences entre les membres du robot permettent de discriminer la branche de solution qui correspond à une hyperextension du tibia. En effet, la valeur de θ_3 est limitée à des angles compris dans l'intervalle $[\frac{-3\pi}{4}; \frac{-\pi}{4}]$. La branche de solution négative

est donc celle qui est sélectionnée pour la poursuite de la procédure. Tout comme pour θ_3 , il est possible d'utiliser les substitutions trigonométriques suivantes :

$$\cos \theta_1 = \frac{1 - t_1^2}{1 + t_1^2}; \quad \sin \theta_1 = \frac{2t_1}{1 + t_1^2}; \quad t_1 = \tan\left(\frac{\theta_1}{2}\right) \quad (4.15)$$

En introduisant les valeurs de θ_3 trouvées précédemment dans l'équation (4.10) et en effectuant les substitutions de θ_1 on obtient l'équation quadratique suivante :

$$Yt_1^2 + 2Xt_1 - Y = 0 \quad (4.16)$$

Cette dernière peut se résoudre pour θ_1 de la façon suivante :

$$t_1 = \frac{-X \pm \sqrt{X^2 + Y^2}}{Y} \quad (4.17)$$

Deux branches de solutions émergent encore une fois de cette résolution. Les contraintes mécaniques et les interférences entre les membres du robot permettent de discriminer une de ces branches de solution. En effet, la valeur de θ_1 est limitée à des angles compris dans l'intervalle $[\frac{3\pi}{4}; \frac{5\pi}{4}]$ pour les pattes du côté gauche et dans l'intervalle $[\frac{-\pi}{4}; \frac{\pi}{4}]$ pour les pattes du côté droit. Une seule des solutions donnera un angle dans cet intervalle, celui des deux qui sera la plus près de π pour les pattes de gauche et la plus près de zéro pour les pattes de droite. Physiquement, une des deux solutions permet l'hyperextension du fémur à l'articulation de la variable articulaire θ_2 . Pour compléter la résolution, il faut donc exécuter le calcul de l'angle θ_2 grâce aux équations (4.11 et 4.12) en utilisant simplement :

$$\theta_2 = \text{atan2}(\sin \theta_2, \cos \theta_2) \quad (4.18)$$

La résolution de θ_2 à l'aide d'arctangente à deux arguments donne une seule solution, il faut cependant s'assurer que cette solution soit dans les limites articulaires et comprise dans l'intervalle $[\frac{-\pi}{4}; \frac{\pi}{4}]$.

Grâce à la résolution du problème géométrique inverse énoncée précédemment, l'orientation qui doit être imposée aux actionneurs peut être trouvée en fonction d'une position cartésienne à imposer à l'extrémité de la patte. Cette procédure doit être effectuée en temps réel par l'unité de calcul embarquée sur le robot et à chacun des incréments du déplacement. À la fin de la procédure, une seule solution peut répondre à la prescription imposée par les limites articulaires des pattes. Dans le cas où aucune des branches de solution ne satisfait ces prescriptions, le point demandé est hors de l'espace de travail et doit être rejeté.

4.2.2 Type d'actionneurs

L'architecture retenue pour les pattes du robot implique des articulations rotoïdes. Le type de motorisation le plus répandu pour ce type d'articulation est d'utiliser des moteurs rotatifs électriques à courant continu. Il est aussi possible d'utiliser des moteurs rotatifs sans brosse (brushless) ou des moteurs rotatifs à ultra-son, cependant ces solutions ont été rejetées puisque ces types de moteur sont dispendieux et nécessitent une unité de contrôle particulière et habituellement encombrante. Les articulations pourraient aussi être motorisées par des actionneurs linéaires. En effet, des vérins, des mécanismes utilisant des vis sans fin, des moteurs linéaires ou même des matériaux à mémoire de forme pourraient également permettre aux pattes de bouger. Cependant, ces types de motorisation impliquent la conversion de mouvement linéaire en mouvement rotatif, ce qui ajoute à la complexité de la patte ainsi qu'à sa masse. Le prototype de robot marcheur qui a été développé s'avère une première exploration des éléments de ce domaine, la simplicité est donc un avantage indéniable. Le choix des actionneurs s'est donc porté sur des servomoteurs à courant continu. Simple et efficace, ce type de moteur est réputé fiable et robuste. Il se présente sous plusieurs modèles. Afin de choisir le modèle le plus apte à répondre à notre besoin, plusieurs paramètres ont été évalués. Les aspects les plus importants qui ont conduit au choix des actionneurs sont :

- les dimensions,
- la masse,
- la puissance et le couple fourni,
- le rendement énergétique,
- la friction interne dans le réducteur ou l'autoblocage du moteur à l'arrêt,
- le type d'unité de contrôle requis,
- le coût.

L'architecture choisie requiert trois actionneurs pour positionner l'extrémité de la patte. Dix-huit moteurs sont donc nécessaires pour le robot au complet. Ce nombre élevé d'actionneurs rend critique choix de ceux-ci. La dimension des moteurs servira de paramètre dimensionnel au design du reste de chacune des pattes. En effet, des moteurs de petites dimensions permettent par le fait même la construction d'un robot de petite dimension. Un tel robot est plus pratique du point de vue expérimental et est tout autant suffisant pour la validation des concepts théoriques.

Une partie non négligeable de la masse du robot vient des moteurs. Les membres du robot actionnés par des moteurs de faibles dimensions sont habituellement eux aussi de

dimensions plus petites, ce qui est avantageux. De petits moteurs ont tendance à être eux aussi de faible masse, permettant à tout l'assemblage d'être plus léger. La masse du robot est un aspect critique au développement, surtout dans le cadre d'une application pour une plate-forme robotisée d'exploration spatiale.

Le couple pouvant être fourni par les moteurs est un aspect critique du choix de ceux-ci. Le calcul menant à une évaluation de la puissance nécessaire pour que le robot puisse marcher sera expliqué en détail dans la section 4.4.4 sur l'équilibrage statique. Il est cependant utile de noter que les moteurs actionnant le coxa (θ_1) et le fémur (θ_2) ont à fournir plus de couples durant des conditions de marche normale que celui actionnant le tibia (θ_3). Le moteur pour le tibia est donc d'un modèle moins puissant et par le fait même, moins volumineux.

Au niveau du contrôle, l'unité de calcul qui sera utilisée doit offrir assez de sorties pour permettre d'envoyer la consigne à tous les moteurs en fonction de l'orientation désirée pour ceux-ci. En utilisant des servomoteurs standard, des potentiomètres donnant la position sous forme d'une valeur de tension analogique sont envisageables. L'autre possibilité pour connaître l'orientation des moteurs est d'utiliser des encodeurs. Cette façon de faire nécessite aussi un module permettant l'envoi de tension et de courant aux bornes du moteur. Un pont en H (H bridge) est souvent utilisé pour accomplir cette mission. La conversion de la consigne en commande à envoyer au moteur se fait à l'intérieur de l'unité de calcul, ce qui augmente la charge des tâches qu'il a à effectuer.

La solution qui a été choisie est d'utiliser des moteurs rotatifs de type RC servo (Remote Control) fabriqués par Hitec. Ils sont habituellement utilisés dans le domaine des véhicules radioguidés pour actionner le système de direction. Les fiches de caractéristique des moteurs utilisés sont présentées en annexe D.1. Ces moteurs peuvent fonctionner entre 4.8 et à 6 volts selon la vitesse et le couple qui sont requis, ils seront utilisés à 6VDC pour utiliser au maximum le potentiel de leurs caractéristiques. Ils sont compacts et peu coûteux, c'est d'ailleurs pourquoi leur utilisation en robotique se multiplie. Leur plage de fonctionnement est seulement de cent-quatre-vingts degrés, ceci n'est cependant pas limitatif puisque les débattements articulaires visés sont de quatre-vingt-dix degrés.

Leur avantage principal est qu'ils comprennent un module interne de gestion de la puissance et de la position qui remplace l'utilisation d'un pont en H externe. La consigne

leur est transmise directement en envoyant un échelon de 5VDC d'une durée comprise entre 0.9 et 2.1 millisecondes et rafraîchie selon une période de vingt millisecondes. La figure 4.5 est une photo des moteurs utilisés.



FIG. 4.5 – Servomoteurs

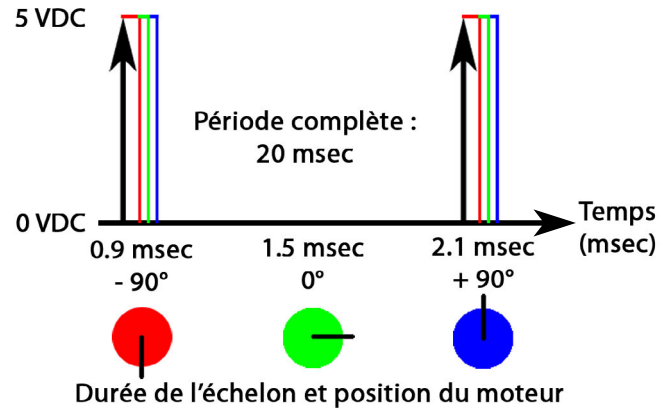


FIG. 4.6 – Contrôle des servomoteurs RC

La figure 4.6 montre l'allure de l'échelon qui est envoyé comme consigne au moteur en fonction de l'orientation désirée. La période totale de vingt millisecondes doit être fidèlement respectée, un nouveau signal doit être envoyé périodiquement sans quoi le moteur interprétera une consigne de zéro milliseconde. La durée de l'échelon est fonction de l'orientation désirée selon les équations suivantes :

$$\theta_1^{PWM} = \frac{\theta_1^{degre} + \theta_1^{\delta cote}}{\pi} \lambda_1 + \Delta_1^{offset} \quad (4.19)$$

$$\theta_2^{PWM} = \frac{signe_2^{cote} \theta_2^{degre}}{\pi} \lambda_2 + \Delta_2^{offset} + \Delta_2^{load} \quad (4.20)$$

$$\theta_3^{PWM} = \frac{signe_3^{cote} (\theta_3 + \pi/2)}{\pi} \lambda_3 + \Delta_3^{offset} \quad (4.21)$$

θ_*^{degre} est la valeur de consigne qui vient du problème géométrique inverse. $\theta_1^{\delta cote}$ est la rotation qu'effectue l'axe des X autour de l'axe des Z durant le changement de repère nécessaire pour que la même fonction de résolution du problème géométrique inverse puisse être utilisée pour toutes les pattes. λ_* est un facteur de correspondance entre la durée de l'impulsion nécessaire pour correspondre à une valeur d'angle. Δ_1^{offset} est le biais qui recentre l'orientation des des moteurs à l'origine. $signe_*^{cote}$ prend la valeur 1 pour les pattes situées du coté droit et -1 pour celles situées du coté gauche.

Le circuit de contrôle interne de ces moteurs est cependant assez primitif. Il n'y a pas de limitation électronique à ce qui est envoyé au moteur, ce qui veut dire que si la consigne d'entrée n'est pas dans la plage de ± 90 degré, le contrôleur essaiera tout de même de faire prendre la position résultante au moteur. Ce dernier ayant des limites physiques à son orientation de sortie en vient à l'autodestruction dans le cas d'une consigne erronée, ce qui est assez désavantageux. Le contrôleur de l'orientation du moteur utilise seulement une compensation proportionnelle à l'erreur d'orientation fournie par un potentiomètre fixé directement sur l'arbre de sortie du réducteur du moteur. Dans le cas où le couple statique à fournir est grand, l'erreur sur la position devient grande, ce qui n'est pas souhaité. L'hypothèse du fonctionnement de la commande en boucle ouverte pour le positionnement des moteurs est donc fautive. Le terme Δ_2^{load} dans les équations précédentes est un biais qui est ajouté à la commande du moteur pour compenser la charge statique du ressort en torsion présente sur cette articulation.

Afin de corriger ce problème et d'obtenir de meilleures performances en orientation avec les moteurs, des modifications peuvent être réalisées. Une des solutions qu'il est possible d'utiliser est de remplacer le contrôleur interne par un contrôleur externe. Pour conserver le module de gestion de la puissance interne au moteur, il est possible de modifier l'électronique interne du moteur afin de tromper le contrôleur interne et qu'il reçoive en permanence la même information, comme si le moteur était en position centrale. La façon de faire est de remplacer, sur la carte de contrôle, l'entrée du potentiomètre donnant la position de l'arbre de sortie du moteur par deux résistances de valeurs équivalentes. Le signal du potentiomètre en question doit alors être lu par ce qui servira de contrôleur externe.

Pour gérer la position, ce contrôleur doit envoyer une commande semblable à une position désirée, cette dernière doit être proportionnelle au couple à fournir, soit pour faire tourner le moteur ou pour maintenir la charge. Plus la position est loin de la valeur centrale, plus le couple fourni est grand. Cependant, cette solution a le désavantage de présenter un retard d'au moins une période par rapport à la consigne. Les signaux analogiques d'orientation doivent être lus par l'unité de calcul, puis la consigne doit être traduite en commande à envoyer aux moteurs. Toutes ces opérations doivent être faites en temps réel, ce qui augmente de façon importante la charge d'opération à réaliser par l'unité de calcul. La lecture d'une valeur analogique nécessite aussi un convertisseur analogique digital par moteur. Les canaux utilisés ne peuvent servir à la mesure de capteurs externes ou d'autres périphériques sur le robot.

Des tests ont tout de même été réalisés sur un moteur pour évaluer les résultats. Il en est ressorti que les limites physiques des moteurs de ce type sont presque atteintes par le contrôleur interne des moteurs. En effet, en prenant des mesures de tension et de courant à la sortie de la carte interne de contrôle, cela permet de comprendre comment elle fonctionne. La durée de l'échelon de consigne à l'entrée est convertie proportionnellement en échelon aux bornes du moteur. Le maximum pouvant être envoyé correspond à la période complète, une alimentation continue. Le couple qui est alors produit est donc maximum si le moteur est sans charge. Dans ce cas, le moteur chauffe rapidement, et ce, au point qu'il entraîne la fonte du châssis du moteur et du réducteur qui est fait de plastique.

Cette solution augmente de beaucoup la complexité électronique et la charge d'opération sur l'unité de calcul tout en mettant en danger la fiabilité des servomoteurs, cette solution a donc été rejetée. La continuité des travaux avec des moteurs de ce type passe donc par l'équilibrage statique du couple du moteur du fémur, représentée par θ_2 .

Les moteurs de type RC servo ont été utilisés pour leur puissance et le couple fourni qui sont certainement très impressionnant par rapport à leur coût, leur simplicité, leur faible dimension et leur faible masse. Leur rendement énergétique est à l'intérieur de la limite acceptable puisque ces moteurs ont été conçus pour fonctionner dans des véhicules télécommandés qui sont généralement alimentés par des piles.

La friction interne dans le réducteur et surtout le frottement dans le collecteur du servomoteur aide le maintien de couple statique, mais réduit le couple lorsque le moteur est en mouvement. Les spécifications du fabricant traitent de couple à l'arrêt, mais ne donnent pas d'indication sur le couple en rotation ce qui peut être trompeur à l'achat de ce type de moteur. L'autoblocage du moteur à l'arrêt provenant de la friction dans le réducteur sera aussi tiré à profit par l'équilibrage statique.

Le type de consigne devant être envoyé au moteur nécessite une sortie PWM (pulse width modulator). Un circuit de protection pour l'unité de calcul a été installé à la sortie des canaux PWM pour éviter tout retour de tension dans celle-ci et pour fournir des échelons de 5VDC. Des puces électroniques qui servent de commutateurs («latches») remplissent ce rôle, la documentation sur ces composantes se retrouve en annexe D.2.

4.3 Design mécanique

4.3.1 Patte de validation

En partant de l'architecture choisie, plusieurs itérations ont été réalisées et ont permis d'obtenir une version s'approchant de la version finale du design. La patte de validation, montrée à la figure 4.3, a permis de s'assurer de l'atteinte des objectifs fixés. La présentation de la programmation de l'unité de calcul pour la gestion de la consigne envoyée aux moteurs sera faite à la section 4.5.2.

Les dimensions des moteurs donnent un repère dimensionnel pour la longueur des membres. Ces longueurs sont les variables a_1 , a_2 et a_3 dans les équations sur la résolution du problème géométrique inverse de la section 4.2.1.1. Pour que la validation de l'architecture soit faite, les articulations et les membres doivent pouvoir se déplacer sans interférence à l'intérieur des limites articulaires fixées. Les longueurs des membres de cette patte sont semblables à celle du prototype final, le modèle donne donc une bonne idée des possibilités réelles qu'aura une des pattes du prototype final.

Lors du choix de l'architecture, il a été déterminé que $a_1 = 0$ pour des fins de simplification des calculs du problème géométrique inverse. Avec les dimensions des moteurs, a_2 a été fixé dans les environs de quatre-vingts millimètres et a_3 dans les environs de cent millimètres. L'impact de ces longueurs sur la marche du robot sera expliqué dans la prochaine section sur l'équilibrage statique (4.4.4).

La fabrication des membres de la patte de validation a été réduite au plus simple en employant des blocs «Lego technics» en guise de structure. Ceux-ci se présentent comme des barres de plastique noir de longueurs variables où des trous ronds sont présents sur un des côtés, ce qui permet de fixer d'autres éléments. Dans le cas de la patte de validation, ce sont les servomoteurs RC qui y sont fixés à l'aide de fil métallique. L'assemblage résultant est assez solide pour soutenir son propre poids lors de déplacements sans appui au sol. L'assemblage est suffisant pour tester l'architecture de la patte, les moteurs et la possibilité de les contrôler. Les possibilités qu'offrent les blocs «Lego technics» pour permettre de générer et tester des prototypes sont impressionnantes. En peu de temps, des structures peuvent être construites et donnent alors un modèle réel des concepts à analyser.

Les résultats obtenus avec cette patte de validation ont montré qu'il est effectivement possible de positionner l'extrémité de la patte. L'architecture choisie ainsi que les débattements articulaires désirés sont alors réalisables avec le type de moteur choisi ainsi que par l'unité de calcul. Lorsque seul le poids des membrures est appliqué comme charge, les erreurs d'orientation sur les servomoteurs sont inférieure à cinq degrés et le changement de pose s'effectue à l'intérieur de cinq secondes, ce qui est très acceptable dans le cadre du projet.

4.3.2 Hexapode

Plusieurs itérations de design ont abouti à une version finale du robot Hexapode. La modélisation d'Hexapode s'est faite par étapes : d'abord les pattes, puis le corps. L'optimisation de tous les éléments a été faite une fois que toutes les parties ont été modélisées. La conception des modèles CAO menant au prototype final a été faite avec le logiciel de CAO Pro-Engineer. À l'aide de ce logiciel, il est possible d'évaluer la possibilité d'interférence entre les diverses parties de l'assemblage.

Les membres et le corps d'Hexapode ont été soigneusement conçus pour comporter le moins de matériel possible tout en conservant le maximum de rigidité. Pour ce faire, le design des membres doit intégrer les moteurs comme élément de la structure. Les moteurs sont donc enveloppés dans des pochettes internes aux membres pour que l'assemblage résultant soit solide et de faible dimension. L'esthétisme qui en résulte est également soigné puisque le design général donne une impression de continuité par les parties qui le composent. L'intégration des autres composantes a aussi été tenue en compte lors de la conception afin d'augmenter le caractère d'intégrité du design du robot final.

Les trois paires de pattes sont identiques pour permettre la pleine omnidirectionnalité du robot. Sur un même segment, les pattes sont symétriques. La similitude des pattes a pour effet de simplifier le design et l'usinage des pièces du robot. Le modèle CAO, ainsi que la version réelle ont été montrées sur la figure 4.1 et la figure 4.2).

Le design du robot a été fait en gardant en tête la facilité d'assemblage. En effet, les étapes menant au montage des différents éléments du robot ont été prévues avec

soin afin que toutes les pièces puissent être installées selon une séquence précise. Les interférences possibles lors de l'assemblage ont dû être planifiées. Les membres, le corps et les autres éléments sont emboîtables sans briser le matériel. Tous les endroits où des vis ou d'autres éléments d'assemblage sont présents, sont accessibles lors d'au moins une étape de l'assemblage. Il est capital que le design des éléments séparés ait été prévu en vue d'être monté en un robot complet.

4.3.2.1 Fabrication des membres

La majeure partie du prototype final est en matière plastique. Les pièces sont fabriquées avec la technologie du prototypage rapide FDM (Fused Deposition Modeling). Cette méthode de fabrication est explicitée dans l'article du Laboratoire de robotique de l'Université Laval [23].

Cette technologie permet de construire un volume à partir de fines couches de plastique ABS. Le Plastique est chauffé puis déposé par une buse selon la forme de la pièce à produire. La position de la buse est contrôlable selon trois degrés de liberté, ce qui donne à la machine la possibilité de former des pièces complexes, voire impossible à fabriquer par des moyens conventionnels. Pour que la pièce soit fabriquée selon les dimensions et la forme voulue, le plastique chaud et donc encore mou doit être soutenu par un matériel de support pour éviter qu'il ne se déforme. Ce matériel est placé par une seconde buse située sur la même tête mobile que la première buse.

La séquence utilisée pour la fabrication de prototypage rapide est la suivante :

- la conception et modélisation des pièces avec le logiciel Pro-engineer,
- la polygonisation des modèles solides en format de fichier STL,
- l'importation des pièces dans le logiciel Insight,
- le choix de l'orientation et des paramètres de la pièce durant la fabrication,
- la génération et validation de la trajectoire de la buse lors de la fabrication,
- l'implantation de la trajectoire dans le contrôleur de la machine Stratasys,
- la fabrication des pièces,
- le nettoyage et sablage des pièces,
- l'assemblage du prototype.

Pour la machine FDM, il est aussi simple de générer toutes sortes de formes, les pièces ne sont pas conçues avec les mêmes contraintes que celles usinées avec des méthodes de fabrication traditionnelle. Presque n'importe quelle courbe ou forme peut

être générée. De plus, puisque le plastique est moins résistant que le métal, le design à tout avantage à présenter des formes rondes que des arrêtes vives. Ces dernières favorisent les concentrations de contraintes et donc augmentent les risques de bris. Les pièces fabriquées peuvent même être presque creuses. Pour ce faire, la machine à prototypage rapide dépose de fins fil de plastique, permettant de soutenir la structure afin qu'elle puisse être refermée. Ceci présente l'avantage indéniable de diminuer la quantité de matériel pour obtenir une pièce creuse possédant une plus grande inertie. Il faut cependant que la géométrie de la pièce soit favorable à l'évidement, soit présentée une section assez grande pour que la paroi puisse être assez épaisse. Une plus faible quantité de matériel présente l'avantage d'avoir une masse plus faible et un coût de fabrication moins élevé.

Puisque la machine dépose des couches de plastique selon une séquence verticale, les propriétés des pièces à fabriquer sont anisotropes. En effet, les caractéristiques mécaniques des pièces sont différentes selon l'orientation des trajectoires qui ont laissé le plastique. Une compression dans le sens vertical et donc dans la direction normale aux couches est moins dommageable que des forces qui viendraient cisailer les couches entre-elles. La traction dans le sens vertical est aussi à éviter puisque cela reviendrait, à la limite, à séparer les couches. Ces propriétés viennent des différences entre les températures des matériaux solidifiés et encore liquides lors de l'usinage. La résistance est accrue lorsque le matériel se solidifie en même temps sur une même couche par rapport au lien qui se crée lorsque le nouveau matériel chaud vient faire fondre localement le plastique qui est sur la couche de dessous.

Les membres du robot Hexapode ont été conçus afin de tirer avantage de la méthode de fabrication. Ils sont de géométrie simple, et présentent des arrondis partout où c'est possible. Les servomoteurs ont été assemblés directement sur les membres à l'aide de quatre vis à six pans creux # 0. Lors de l'assemblage, les moteurs ont été orientés afin de tenir compte des limites articulaires qui doivent être permises aux membres.

4.3.2.2 Fabrication du corps et assemblage

Le corps du robot (figure 4.10) est construit à partir de tuyaux de un demi pouce de diamètre en CPVC standard. Ce matériau a l'avantage d'être solide, léger et peu dispendieux. De plus, il est possible de trouver en magasin des pièces permettant de raccorder les tuyaux entre eux. Le châssis est conçu de manière à ce que les points de

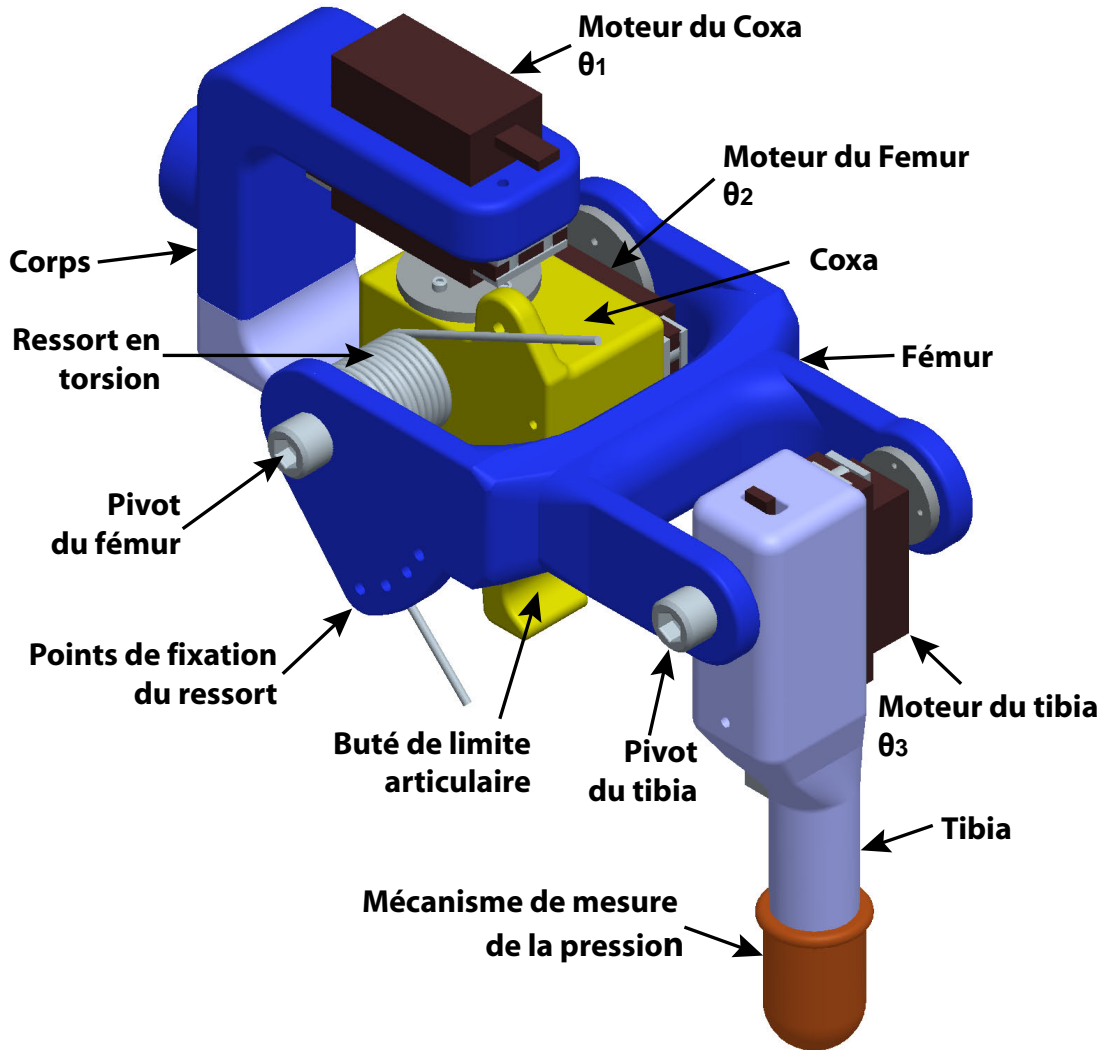


FIG. 4.7 – Présentation des composantes d'une patte.

rencontre des deux premiers axes de rotation des articulations de chacune des pattes se retrouvent sur un cercle de trois cents millimètres de diamètre. Cette géométrie présente trois avantages majeurs.

- Le centre du robot demeure disponible pour recevoir les composantes électroniques.
- La charge soutenue sur chacune des pattes est semblable.
- Les pattes du segment du centre sont dégagées, réduisant ainsi les chances d'interférence avec les autres pattes.

D'autres pièces qui ne sont pas fabriquées par prototypage rapide ou en CPVC viennent compléter le design. Pour éviter que les arbres des moteurs aient à subir trop d'effort radial, des pivots sont ajoutés pour réduire l'usure induite par les charges appliquées. Ces pivots sont des vis dont les extrémités ont été usinées pour permettre

le glissement. La partie près de la tête est encore filetée et sert à visser la vis dans le plastique des membres ou du corps du robot. Afin que ce pivot puisse tourner sans frottement, des paliers lisses sont utilisés à l'endroit où se fera le contact. Ces paliers lisses proviennent de la compagnie Igus, la figure 4.8 montre l'apparence. Leurs caractéristiques détaillées sont présentées à l'annexe D.1.



FIG. 4.8 – Paliers lisses
pour les pivots



FIG. 4.9 – Ressort

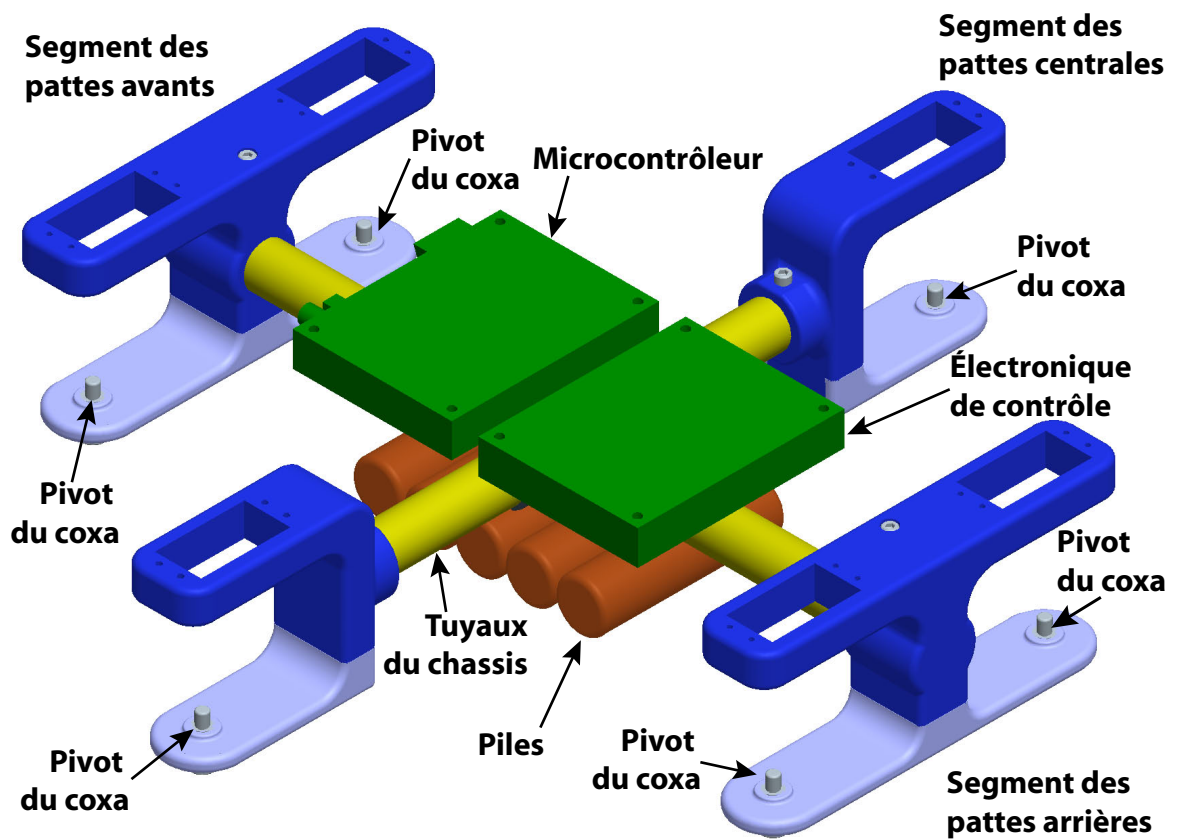


FIG. 4.10 – Description des composantes du corps.

4.4 Autonomie

La clef de l'autonomie est de dépenser le moins d'énergie possible pour se déplacer. Il devient alors possible de naviguer plus longtemps ou d'accomplir davantage d'actions avec les mêmes réserves énergétiques. Les éléments qui consomment le plus d'énergie sont, hors de tout doute, les moteurs. Ces derniers doivent être en mesure de fournir un couple capable soutenir le poids du robot et de le faire se déplacer avec aisance. L'autre composante qui consomme de l'énergie est le microcontrôleur, sa consommation est de 200mA, ce qui est inférieur à la consommation sans charge des HS-85mg (240mA) et à celle des HS-225mg (300mA).

Pour un robot marcheur à six pattes, la majorité du poids vient des pattes, donc des membres et des servomoteurs. Le reste du poids provient principalement du corps et de l'unité de calcul. Il importe donc de minimiser la masse transportée, l'énergie cinétique et potentielle en jeu dans le cas de déplacement d'une petite structure légère est plus petite.

Pour minimiser la consommation d'énergie, les servomoteurs doivent fournir les couples les plus faibles possible. Puisque les déplacements du robot sont quasi statiques, il devient réaliste de calculer le couple statique à chacun des incréments de la position des pattes. Les paramètres qui influencent le couple à fournir sont :

- les dimensions des membres,
- les pertes énergétiques dans le système de motorisation,
- l'architecture du robot,
- la pose du robot,
- la masse et la répartition de la masse des composantes,
- l'orientation du robot et du polygone de support par rapport à la verticale,
- les conditions du terrain.

4.4.1 Les dimensions des membres

La longueur des membres est représentée par les scalaires a et b dans les paramètres HD dans le tableau 4.2.1.1. Des membres longs permettent un espace de travail plus

grand à l'extrémité de la patte. Cependant, le couple provenant de la force appliquée au point de contact au sol est plus important lorsque ce point de contact est loin de l'articulation. Un compromis doit être fait entre la dimension de l'espace de travail venant des dimensions des membres et des couples résultant de tel bras de levier.

4.4.2 Les pertes énergétiques du système de motorisation

Les organes de transmission et les réducteurs ont des rendements qui sont rarement près de l'unité, à cause des frottements internes entre leurs composantes et leur déflexion. Il est donc requis de les utiliser intelligemment. En fixant les membres directement sur l'arbre de sortie des servomoteurs, les organes supplémentaires de transmission intermédiaires sont évités. Les servomoteurs utilisés présentent déjà des réducteurs en interne qui sont optimisés pour permettre de meilleures performances, il est donc utile que les servomoteurs sélectionnés soient en mesure de fournir le couple nécessaire au bon fonctionnement des déplacements du robot sans ajouter d'autres systèmes de réduction. Les pertes de puissance dues au mauvais rendement que présentent des systèmes de réduction en série sont évitées, ce qui permet d'abaisser le couple à fournir par le servomoteur. De plus, il est difficile d'évaluer les pertes dues à la friction qui seront rencontrées sur un modèle réel. Des dégradations apparaissent avec le temps, augmentant le coefficient de friction entre les pièces fabriquées en prototypage rapide. Le fait de sélectionner un servomoteur sans avoir à ajouter de système de réduction externe aide à garder les propriétés mécaniques du robot Hexapode constantes dans le temps.

4.4.3 Les effets de la variation des angles ϕ et ψ

Le couple que les servomoteurs ont à fournir, est fonction de l'architecture du robot. En effet, l'arrangement des articulations et les dimensions des membres ont un impact sur les couples nécessaires pour effectuer les déplacements propres à la marche. Les paramètres ϕ et ψ de l'architecture, montrés sur la figure 3.2, ont un impact pour la répartition des couples sur les servomoteurs d'une patte. En effet, les couples que doivent fournir les servomoteurs résultant de la force normale provenant du support au sol sont sensibles à l'orientation des axes de rotation de leurs articulations et du bras de levier créé par le membre par rapport à la charge. Un servomoteur ressent le

moment axial qui résulte d'une force appliquée sur le bras de levier ainsi que le moment dû à la friction de ses éléments internes. Les moments selon des directions différentes de celle de l'axe du servomoteur ne sont pas subis par le servomoteur et sont repris par les roulements du servomoteur et le pivot.

Le fait de varier l'orientation de l'axe de rotation du coxa (θ_1) a donc pour effet de changer la charge effective appliquée aux servomoteurs. Cependant, ceci a aussi pour effet de modifier les valeurs articulaires nécessaires pour obtenir une position à l'extrémité de la patte. Avec une configuration orthogonale ($\phi = 0$ et $\psi = 0$), il y a découplage partiel des valeurs articulaires. Avec cette configuration, l'articulation du coxa (θ_1) n'a pas d'effet sur la position verticale (axe Z) de l'extrémité de la patte, et n'a d'influence que pour des mouvements dans le plan X-Y. Cette articulation permet d'orienter un plan dans lequel les autres articulations ont leur axe de rotation normal à ce plan. Il est donc possible de répartir la charge que subissent les servomoteurs en faisant varier ces paramètres de l'architecture. Ceci est jugé avantageux puisque cela permet d'isoler la charge due aux effets de la gravité sur les actionneurs représentés par θ_2 et θ_3 .

L'orthogonalité des axes est préférée, car elle permet d'isoler le couple à fournir pour soutenir le poids du corps d'Hexapode principalement sur un seul actionneur. La pose dite standard pour Hexapode est d'orienter les membres selon des valeurs articulaires de $\theta_1 = 0$, $\theta_2 = 0$ et $\theta_3 = \frac{-\pi}{2}$. Dans cette pose, si le polygone de support est horizontal, le couple nécessaire à l'articulation du tibia (θ_3) est nul. La force de contact appliquée sur le membre est normale au sol, il n'y a donc pas de composante de bras de levier venant créer un moment à l'articulation. Le servomoteur qui soutient la majorité de la charge est donc celui qui donne l'orientation de la variable articulaire θ_2 (Fig. 3.2). Il devient alors possible d'agir localement sur cette articulation.

4.4.4 Équilibrage statique

Il est intéressant au niveau énergétique et dynamique de tenter de compenser les effets dus à la gravité par un mécanisme d'équilibrage statique. Cette technique est utilisée pour minimiser le couple que doivent fournir les actionneurs. L'utilisation de contrepoids, de ressorts ou de leviers permet de créer des forces ou des couples inverses

à ceux qui proviennent de l'effet de la gravité. Le résultat est que, pour un équilibrage statique parfait, la plate-forme équilibrée maintient sa position et son orientation sans que les actionneurs ne fournissent de couple. Le concept derrière l'équilibrage statique est que l'énergie potentielle totale d'un système demeure constante en tout point de son espace de travail. Ceci a comme résultat que le système sera à l'équilibre peu importe, sa configuration. La conséquence directe de ce principe est que la masse apparente du système est annulée par rapport aux actionneurs.

La section précédente a montré que le couple nécessaire aux actionneurs pour soutenir le poids d'Hexapode serait concentré à l'articulation θ_2 . Il est donc intéressant de fournir un couple d'équilibrage qui soit compensatoire sur le servomoteur ayant à combattre la force de gravité qui s'applique sur l'Hexapode. Cette technique s'appelle l'équilibrage statique. En théorie, elle permet aux servomoteurs d'appliquer seulement le couple requis pour vaincre l'inertie de l'élément à déplacer, qui est généralement moindre que le couple total à fournir.

Cette méthode a aussi été aussi utilisée dans le laboratoire de robotique de l'Université Laval sur des manipulateurs parallèles [26] et [27]. Dans le cas des simulateurs de vol, la grande masse de l'effecteur implique un apport énergétique énorme venant des actionneurs, et ce, même dans le cas où il faut uniquement maintenir leur position. La grande inertie de cette masse fait diminuer la précision et la vitesse des mouvements de la plate-forme ce qui n'est pas souhaitable.

L'équilibrage statique permet aux actionneurs de ne fournir uniquement ce qui est nécessaire pour mettre le système en mouvement. La consommation énergétique est donc réduite, la taille des servomoteurs est moindre et par le fait même, la masse totale du système aussi. Cependant, ce qui est possible dans le cadre des robots parallèles ne l'est pas exactement dans le cadre des robots marcheurs. Pour les manipulateurs parallèles, les paramètres physiques peuvent être déterminés avec précision pour tout point de l'espace de travail. La base est ancrée au sol et la plate-forme est de masse et de dimension connues, elle ne subit habituellement pas de modification en cours d'utilisation. Dans le cas des robots marcheurs, les paramètres physiques changent en fonction de plusieurs paramètres environnementaux :

- le relief qui impose la géométrie au polygone de support,
- l'orientation du robot par rapport à la verticale,

- l’orientation du polygone de support par rapport à celle du corps,
- les variations de pose pour effectuer le déplacement des polygones de support et de retour.

Les robots marcheurs peuvent donc être considérés comme des mécanismes à topologie variable, puisque les conditions des chaînes cinématiques qui les relient avec les points de contact changent constamment. Lors de la marche, la recherche de la stabilité entraîne des modifications radicales des paramètres physiques du robot. Ceci entraîne par le fait même des modifications sur la possibilité d’équilibrer statiquement le système.

Les conditions du terrain suggèrent des paramètres de navigation spécifiques lors de l’exécution de trajectoire. Ces paramètres de navigation sont la vitesse, la grandeur et la direction des incréments du déplacement des polygones de support et de retour. De par ces paramètres, la topologie, l’orientation du robot par rapport aux effets de la gravité, la qualité des contacts avec le sol, ainsi que le nombre de pattes en contact avec le sol, font que la force locale qui s’applique à l’extrémité de la patte est variable. Cela implique bien entendu des couples variables aux actionneurs.

Pour cette raison, l’équilibrage statique parfait ne peut être réalisé. Il est tout de même possible de réaliser un équilibrage partiel, aidant à diminuer le couple dans une situation d’utilisation standard. Un ressort en torsion est ajouté pour fournir ce couple antagoniste. Toutefois, ce type de composante ne fournit pas un couple continu sur toute sa plage d’utilisation. Ce couple est proportionnel au produit de la fraction d’un tour complet où il est opéré, par le couple maximal pouvant être fourni. L’équation suivante montre la relation sous forme mathématique :

$$C = \frac{\Delta\theta}{\theta_{max}} C_{max} \quad (4.22)$$

Les ressorts en torsion sont habituellement faits de fil d’acier inoxydable d’un certain diamètre enroulé hélicoïdalement pour un nombre de spires d’un diamètre permettant une déformation élastique. Ces diamètres et le nombre de spires sont les paramètres qui influencent le couple maximal pouvant être fourni par le ressort avant de subir des déformations plastiques. Plus les diamètres sont gros plus le ressort peut emmagasiner d’énergie potentielle et donc fournir de couple. Les ressorts ayant été utilisés proviennent

de Associated Spring Raymond. Les tableaux donnant les spécifications des ressorts utilisés se retrouvent dans les tableaux du catalogue de la compagnie, voir l'annexe D.1.

Comme la plage d'utilisation est de $\frac{\pi}{2}$ radian, une variation du quart du couple maximal est à prévoir entre le maximum et le minimum fourni. Le ressort qui est sélectionné doit être celui qui a le couple minimum pour l'application tout en laissant un facteur de sécurité pour éviter les déformations plastiques. Puisque le couple varie en fonction de l'orientation du fémur du robot, il y a une position d'équilibre différente selon la topologie du robot. Bien que désavantageux, cet écart est inévitable avec l'utilisation d'un seul ressort en torsion comme système de compensation. Cependant en pratique, la friction dans les réducteurs des servomoteurs est assez élevée pour permettre de maintenir une plage de position d'équilibre d'environ $\frac{\pi}{8}$ radian près de la pose standard.

4.4.4.1 Étude de chargement

Une étude du chargement statique a été réalisée afin de connaître plus précisément les composantes de couple que subissent les servomoteurs. Le cas où le nombre de pattes qui soutiennent le poids du robot Hexapode est celui où une démarche tripode est effectuée. Seulement trois pattes sont alors en contact avec le sol. Les type de démarches, dont la démarche tripode, sont expliquées à la section 4.5.5.2 du présent chapitre. Après avoir modélisé les membres et le châssis, les masses de ces éléments peuvent être évaluées en connaissant le volume de matière qui est nécessaire à leur fabrication. La masse volumique du plastique ABS utilisé dans la fabrication par prototypage rapide est de $1.07g/cm^3$. Une approximation des paramètres physiques des membres est montrée dans le tableau 4.3.

	Masse (g)	Longueur (mm)	Position CdM (mm)
<i>Coxa</i>	50	[0, 0, 0]	[0, 0, 0]
<i>Femur</i>	70	[80, 0, 0]	[40, 0, 0]
<i>Tibia</i>	45	[100, 0, 0]	[40, 0, 0]
<i>Corps</i>	870	[0, 0, 0]	[0, 0, 0]

TAB. 4.3 – Masse des membres

Le poids des pattes qui forment le polygone de retour et du corps repose sur les pattes du polygone de support. Les équations d'équilibre statique ont été écrites pour chacun

des membres possédant un actionneur soutenant la charge. À l'équilibre statique, la somme des forces est égale à zéros, de même que la somme des moments.

$$\sum \mathbf{F} = 0 \quad \sum \mathbf{M} = 0 \quad (4.23)$$

Où \mathbf{F} est la force et \mathbf{M} le couple.

Les sommes des forces et les sommes des moments calculées pour chacun des membres résultent en un système d'équations. Pour permettre de résoudre ce système, d'autres contraintes sont ajoutées. Puisque les calculs s'effectuent dans le repère global et fixe, la gravité a seulement un effet selon l'axe Z. Les forces résultantes dans les membres sont donc dans cette direction uniquement, les composantes selon l'axe X et l'axe Y sont posées à zéro. Ceci est important et empêche la présence de forces internes parasites dans le polygone de support. Les moments selon l'axe Z sont également imposés à zéro encore une fois puisque la stabilité du polygone de support implique l'absence de force ou de couple interne.

Sous une forme vectorielle, les trois composantes des forces sont représentées en $[X, Y, Z]^T$, la même représentation s'applique pour les moments selon les mêmes axes. La représentation vectorielle simplifie les calculs puisqu'elle permet de traiter plus d'informations en même temps.

Le premier système d'équations qui apparaît ci-après, prend en compte la somme des forces et des moments en considérant tous les éléments du robot en même temps. Toutes les forces résultant des masses des composantes du robot « \mathbf{W} » doivent être égales aux forces normales « \mathbf{N} » qui s'appliquent au polygone de contact. La convention suivante est appliquée à reste du calcul, « \mathbf{R} » définit les forces résultantes internes entre les membres, « \mathbf{L} » est le bras de levier représentant la distance entre les articulations. L'indice « B » est pour le corps, l'indice « C » est pour le coxa, l'indice « F » pour le fémur et l'indice « T » pour le tibia. L'exposant appliqué aux vecteurs indique à quelle patte est lié le vecteur. « Av » est pour exprimer une des pattes du segment avant, « Ce » est pour le segment du centre et « Ar » est pour le segment arrière. Pour compléter la notation, « G » est pour une patte du côté gauche et « D » est pour une patte du côté droit. L'indice « $_{cdm}$ » tient pour une référence de distance par rapport au centre de masse.

Système d'équations du robot complet :

$$\sum \mathbf{F} = \mathbf{W}_B + 6(\mathbf{W}_C + \mathbf{W}_F + \mathbf{W}_T) + \mathbf{N}^{AvG} + \mathbf{N}^{CeD} + \mathbf{N}^{ArG} \quad (4.24)$$

$$\begin{aligned} \sum \mathbf{M} = & (\mathbf{L}_{Bcdm-C}^{AvG} + \mathbf{L}_F^{AvG} + \mathbf{L}_T^{AvG}) \times \mathbf{N}^{AvG} + \dots \\ & (\mathbf{L}_{Bcdm-C}^{CeD} + \mathbf{L}_F^{CeD} + \mathbf{L}_T^{CeD}) \times \mathbf{N}^{CeD} + \dots \\ & (\mathbf{L}_{Bcdm-C}^{ArG} + \mathbf{L}_F^{ArG} + \mathbf{L}_T^{ArG}) \times \mathbf{N}^{ArG} + \dots \\ & \mathbf{L}_{Bcdm-C}^{AvG} \times \mathbf{W}_C + \dots \\ & \mathbf{L}_{Bcdm-C}^{CeD} \times \mathbf{W}_C + \dots \\ & \mathbf{L}_{Bcdm-C}^{ArG} \times \mathbf{W}_C + \dots \\ & (\mathbf{L}_{Bcdm-C}^{AvG} + \mathbf{L}_{C-Fcdm}^{AvG}) \times \mathbf{W}_F + \dots \\ & (\mathbf{L}_{Bcdm-C}^{CeD} + \mathbf{L}_{C-Fcdm}^{CeD}) \times \mathbf{W}_F + \dots \\ & (\mathbf{L}_{Bcdm-C}^{ArG} + \mathbf{L}_{C-Fcdm}^{ArG}) \times \mathbf{W}_F + \dots \\ & (\mathbf{L}_{Bcdm-C}^{AvG} + \mathbf{L}_F^{AvG} + \mathbf{L}_{F-Tcdm}^{AvG}) \times \mathbf{W}_T + \dots \\ & (\mathbf{L}_{Bcdm-C}^{CeD} + \mathbf{L}_F^{CeD} + \mathbf{L}_{F-Tcdm}^{CeD}) \times \mathbf{W}_T + \dots \\ & (\mathbf{L}_{Bcdm-C}^{ArG} + \mathbf{L}_F^{ArG} + \mathbf{L}_{F-Tcdm}^{ArG}) \times \mathbf{W}_T \end{aligned} \quad (4.25)$$

La résolution de ce système donne les informations sur la valeur des forces normales par rapport au poids du robot. Ceci est important puisque ces forces sont ce qui est supporté par chaque patte. Les capteurs de charge qui sont installés à l'extrémité des pattes peuvent servir à faire une comparaison qualitative de cette valeur. La poursuite du calcul de chargement passe par la résolution des contraintes internes à chacun des membres. Cette opération donne un système d'équations à résoudre en tenant compte de toutes les contraintes à la fois. Les équations pour chacun des membres sont décrites selon les relations avec les membres avoisinants.

D'abord, la somme des forces et des moments du corps d'Hexapode est définie. Le centre de masse du corps est au centre géométrique du châssis. Il est à noter que les pattes comprises dans le polygone de retour sont incluses comme faisant partie du corps puisqu'elles ne soutiennent pas de charge et ne font partie que de l'encombrement total du corps du robot.

$$\sum \mathbf{F} = \mathbf{R}_{B-C}^{AvG} + \mathbf{R}_{B-C}^{CeD} + \mathbf{R}_{B-C}^{ArG} + \mathbf{W}_C + 3(\mathbf{W}_C + \mathbf{W}_F + \mathbf{W}_T) \quad (4.26)$$

$$\begin{aligned} \sum \mathbf{M} = & \mathbf{L}_{Bcdm-C}^{AvG} \times \mathbf{R}_{B-C}^{AvG} + \mathbf{L}_{Bcdm-C}^{CeD} \times \mathbf{R}_{B-C}^{CeD} + \dots \\ & \mathbf{L}_{Bcdm-C}^{ArG} \times \mathbf{R}_{B-C}^{ArG} + \mathbf{M}_{B-C}^{AvG} + \mathbf{M}_{B-C}^{CeD} + \mathbf{M}_{B-C}^{ArG} \end{aligned} \quad (4.27)$$

Ensuite, est présentée la somme des forces et des moments pour le coxa de chacune des pattes comprises dans le polygone de support. Dans le cas traité, les trois pattes sont celle en avant à gauche, celle au centre à droite et celle en arrière à gauche. Le même cheminement peut être fait pour des polygones de contact différents.

$$\sum \mathbf{F}_C^{AvG} = -\mathbf{R}_{B-C}^{AvG} + \mathbf{R}_{C-F}^{AvG} + \mathbf{W}_C \quad (4.28)$$

$$\sum \mathbf{F}_C^{CeD} = -\mathbf{R}_{B-C}^{CeD} + \mathbf{R}_{C-F}^{CeD} + \mathbf{W}_C \quad (4.29)$$

$$\sum \mathbf{F}_C^{ArG} = -\mathbf{R}_{B-C}^{ArG} + \mathbf{R}_{C-F}^{ArG} + \mathbf{W}_C \quad (4.30)$$

$$\sum \mathbf{M}_C^{AvG} = -\mathbf{M}_{B-C}^{AvG} + \mathbf{M}_{C-F}^{AvG} \quad (4.31)$$

$$\sum \mathbf{M}_C^{CeD} = -\mathbf{M}_{B-C}^{CeD} + \mathbf{M}_{C-F}^{CeD} \quad (4.32)$$

$$\sum \mathbf{M}_C^{ArG} = -\mathbf{M}_{B-C}^{ArG} + \mathbf{M}_{C-F}^{ArG} \quad (4.33)$$

Puis, viennent les contraintes relatives à la somme des forces et des moments pour le fémur de chacune des pattes. Il est à noter que l'étude de la longueur du fémur a été réalisée en modifiant ce paramètre et en analysant son effet. Un compromis a été fait entre l'espace de travail pouvant être utilisé par le robot et le couple qui résulte d'un tel bras de levier. La longueur entre les articulations du fémur a été fixée à quatre-vingts millimètres.

$$\sum \mathbf{F}_F^{AvG} = -\mathbf{R}_{C-F}^{AvG} + \mathbf{R}_{F-T}^{AvG} + \mathbf{W}_F \quad (4.34)$$

$$\sum \mathbf{F}_F^{CeD} = -\mathbf{R}_{C-F}^{CeD} + \mathbf{R}_{F-T}^{CeD} + \mathbf{W}_F \quad (4.35)$$

$$\sum \mathbf{F}_F^{ArG} = -\mathbf{R}_{C-F}^{ArG} + \mathbf{R}_{F-T}^{ArG} + \mathbf{W}_F \quad (4.36)$$

$$\sum \mathbf{M}_F^{AvG} = -\mathbf{L}_{C-Fcdm}^{AvG} \times -\mathbf{R}_{C-F}^{AvG} + \mathbf{L}_{Fcdm-T}^{AvG} \times \mathbf{R}_{F-T}^{AvG} - \mathbf{M}_{C-F}^{AvG} + \mathbf{M}_{F-T}^{AvG} \quad (4.37)$$

$$\sum \mathbf{M}_F^{CeD} = -\mathbf{L}_{C-Fcdm}^{CeD} \times -\mathbf{R}_{C-F}^{CeD} + \mathbf{L}_{Fcdm-T}^{CeD} \times \mathbf{R}_{F-T}^{CeD} - \mathbf{M}_{C-F}^{CeD} + \mathbf{M}_{F-T}^{CeD} \quad (4.38)$$

$$\sum \mathbf{M}_F^{ArG} = -\mathbf{L}_{C-Fcdm}^{ArG} \times -\mathbf{R}_{C-F}^{ArG} + \mathbf{L}_{Fcdm-T}^{ArG} \times \mathbf{R}_{F-T}^{ArG} - \mathbf{M}_{C-F}^{ArG} + \mathbf{M}_{F-T}^{ArG} \quad (4.39)$$

Finalement sont présentées les contraintes dues à la somme des forces et des moments pour le tibia de chacune des pattes. Puisque le tibia travaille dans une orientation qui est près de la normale avec le sol, sa longueur est d'une importance moins grande que celle du fémur. Cependant lorsque l'axe Z du robot n'est plus selon la verticale avec le terrain, la longueur du fémur prend alors de l'importance. Il est aussi utile pour permettre au robot d'enjamber des obstacles, que cette longueur lui permette une

bonne marge au sol. En mettant en commun toutes ces caractéristiques, la longueur du tibia a été fixée à cent millimètres. Le fait que le tibia soit plus long que le fémur est également conforme aux observations biomimétiques qui montrent que c'est également le cas, chez la fourmi [22], pour augmenter la mobilité de chacune des pattes.

$$\sum \mathbf{F}_T^{AvG} = (-\mathbf{R}_{F-T}^{AvG} + \mathbf{N}^{AvG} + \mathbf{W}_T) \quad (4.40)$$

$$\sum \mathbf{F}_T^{CeD} = (-\mathbf{R}_{F-T}^{CeD} + \mathbf{N}^{CeD} + \mathbf{W}_T) \quad (4.41)$$

$$\sum \mathbf{F}_T^{ArG} = (-\mathbf{R}_{F-T}^{ArG} + \mathbf{N}^{ArG} + \mathbf{W}_T) \quad (4.42)$$

$$\sum \mathbf{M}_T^{AvG} = -\mathbf{L}_{F-Tcdm}^{AvG} \times -\mathbf{R}_{F-T}^{AvG} - \mathbf{M}_{F-T}^{AvG} + \mathbf{L}_{Tcdm-N}^{AvG} \times \mathbf{N}^{AvG} \quad (4.43)$$

$$\sum \mathbf{M}_T^{CeD} = -\mathbf{L}_{F-Tcdm}^{CeD} \times -\mathbf{R}_{F-T}^{CeD} - \mathbf{M}_{F-T}^{CeD} + \mathbf{L}_{Tcdm-N}^{CeD} \times \mathbf{N}^{CeD} \quad (4.44)$$

$$\sum \mathbf{M}_T^{ArG} = -\mathbf{L}_{F-Tcdm}^{ArG} \times -\mathbf{R}_{F-T}^{ArG} - \mathbf{M}_{F-T}^{ArG} + \mathbf{L}_{Tcdm-N}^{ArG} \times \mathbf{N}^{ArG} \quad (4.45)$$

La mise en commun de toutes les équations du corps et des divers membres, donne un système qui est résolvable afin de connaître les chargements des servomoteurs. La résolution est faite par le logiciel Matlab en utilisant les outils propres au mathématiques symboliques et numériques. L'algorithme permettant la résolution est présenté à l'annexe C.1.

Les entrées de l'algorithme sont la position de l'extrémité des pattes du polygone de support ainsi que l'orientation du repère du corps par rapport à l'horizontale. Les sorties de l'algorithme sont les forces et les couples internes des membres. Ces informations sont particulièrement utiles lors du raffinement du design du robot, permettant de voir l'effet des modifications amenées aux géométries des membres et du corps. La mise en évidence des conditions favorables émergeant de la conception d'un robot de faible masse et de petite dimension a permis de produire le design final.

Spécifiquement, le couple maximal devant être fourni par l'articulation du fémur (selon l'axe Z) est le plus élevé. Les approximations venant du modèle d'Hexapode permettent de réaliser l'équilibrage statique de l'axe représenté par la variable articulaire θ_2 , soit l'articulation du fémur. Selon les spécifications des servomoteurs utilisés, ainsi que les ressorts en torsion fournissant l'équilibrage, il devient possible de faire la sélection définitive de ces composantes.

Le couple pouvant être produit par le ressort conjointement au couple fourni par le servomoteur, doit supporter la charge nécessaire pour orienter le fémur. L'équilibrage statique à l'aide d'un ressort en torsion sur cet axe implique qu'en l'absence de contact à l'extrémité de la patte, θ_2 aura tendance à diminuer pour venir s'appuyer sur la butée. En présence d'un poids, la charge du robot met en équilibre le système entier à une certaine valeur de θ_2 . Le robot peut donc maintenir une pose sans que le servomoteur ne fournisse de couple. De plus, la friction dans les réducteurs des servomoteurs permet qu'une certaine plage de pose soient stable. Cette plage est fixé autour de $\theta_2 = 0$.

Pour que la patte puisse être relevée, le servomoteur doit fournir un couple supérieur au couple de maintien de la pose avec une extrémité de la patte au sol. Le couple du servomoteur doit être suffisant pour vaincre celui fourni par le ressort ainsi que la friction présente dans le réducteur. La valeur de couple pouvant être fournie par le ressort ne peut donc être plus grande que celui du servomoteur. Ce qui met une limite au couple total qui peut servir à maintenir une pose pour le robot.

Le cas de chargement extrême, que l'on voit sur les illustrations de la ligne du haut de la figure 4.11, se produit lorsque les pattes sont dans la pose :

$$\theta_1^{AvG} = \frac{-\pi}{4}, \theta_1^{CeD} = 0, \theta_1^{ArG} = \frac{\pi}{4}, \theta_2^* = 0 \text{ et } \theta_3^* = \frac{-\pi}{4}.$$

Dans cette pose, le polygone de support est le plus grand pouvant être réalisé, les points de contact sont le plus loin possible du centre de masse du robot, les bras de levier résultants sont donc les plus longs qui puissent être réalisés. Les couples engendrés sont maximaux pour les paramètres appliqués dans ce cas extrême. Cela mène à des mesures de couple de $796mN.m$ pour les pattes avant gauche et arrière gauche et de $916mN.m$ pour la patte du centre droit.

La raison de cet écart s'explique en raison de la disposition du centre de masse du robot par rapport au centre géométrique du polygone de contact. La direction et l'orientation de cette différence a une influence importante dans la répartition de la charge entre les pattes. En effet, dans le cas traité plus tôt, les forces normales appliquées par le sol sur les pattes avant gauche et arrière gauche sont de $5.8N$ alors que la patte centrale subit une force normale de $6.6N$. La distance entre le centre de masse du robot et le centre du polygone est de 3.1 mm dans la direction de l'axe Y (montré sur la figure 3.2).

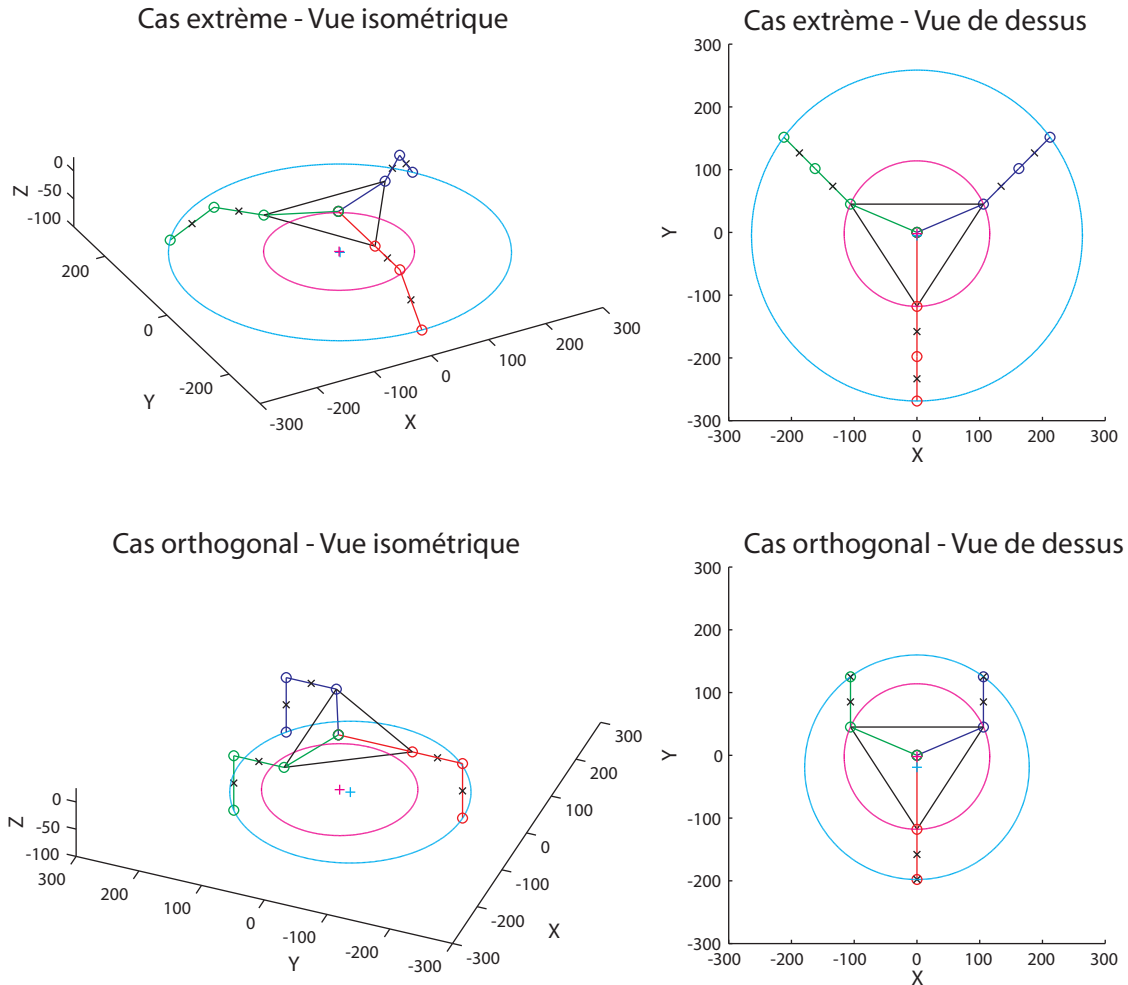


FIG. 4.11 – Représentation des poses des cas de chargement

La pose représentée sur les illustrations de la seconde ligne de la figure 4.11 est celle du cas orthogonal. Les variables articulaires prennent les valeurs suivantes $\theta_1^* = 0$, $\theta_2^* = 0$ et $\theta_3^* = -\frac{\pi}{2}$. Dans ce cas, les couples théoriques à fournir par les servomoteurs sont de 387 mN.m pour les pattes avant gauche et arrière gauche et de 498 mN.m pour la patte du centre droit. Les forces normales appliquées sont de 5.6 N pour les pattes avant gauche et arrière gauche et de 7.0 N pour la patte du centre droit.

L'écart entre les couples des pattes de gauche et les pattes de droite est encore plus marqué dans le cas de la pose orthogonale puisque l'écart entre le centre de masse du robot et le centre géométrique du polygone de support est de 17.1 mm ce qui est assez important si on considère l'échelle des dimensions du robot.

Pour que les couples et les forces normales soient identiques, il aurait fallu que le corps du robot permette de positionner son centre de masse au même endroit que le centre géométrique du polygone de support. Cependant dans la pratique ce n'est pas réalisable, car l'orientation et la position du polygone changent et c'est justement ce déplacement qui fait avancer le robot. La géométrie du corps qui a été choisie est celle qui place les axes de la variable articulaire θ_1 normales à un cercle centré au centre de masse du robot. Ce cercle est celui du plus petit diamètre dans la figure 4.11. Le plus grand cercle est celui qui passe par les points de contact des pattes et qui a comme centre, le centre géométrique du polygone de support.

Pour que le robot soit en mesure de marcher dans toutes les conditions prévues dans l'espace de travail des pattes, il faut que le couple combiné des servomoteurs et les ressorts soient plus grands que celui du cas extrême. Les servomoteurs sélectionnés peuvent fournir un couple de $470mN.m$ et les ressorts les plus adaptés fournissent un couple de $475mN.m$. Un décalage de la plage d'utilisation est alors nécessaire puisque le ressort est plus fort que le servomoteur. Le total théorique de couple pouvant être fourni est donc de $470 + 470 \times \frac{7}{8} = 940mN.m$ pour que le servomoteur atteigne sa valeur maximale de couple lorsque la patte est de $\frac{1}{8} \times C_{max}$ plus tendue, lorsque la valeur de $\theta_2 = \frac{\pi}{8}$. Le cas critique est donc couvert dans la plage de fonctionnement théorique, le couple nécessaire est moindre.

Cependant, dans la pratique, la façon dont les spécifications des servomoteurs sont présentées, est trompeuse (annexe D.1). Par «Stall torque», il n'est pas entendu «couple à l'arrêt» au sens du couple pouvant être fourni pour faire tourner le servomoteur. Il est plutôt entendu le couple à l'arrêt des servomoteurs en incluant le couple de friction. Cette façon de présenter les caractéristiques fait que le couple effectif qui peut servir à mettre le système en mouvement est moins élevé que ce qui était escompté. Pour les autres servomoteurs rencontrés, la façon de présenter les spécifications montre un couple à la sortie de l'arbre du servomoteur et ce couple subit une diminution selon l'efficacité du réducteur à transmettre ce couple.

Dans le cas où le robot supporte une charge sans se déplacer, la friction aide le moteur. Dans le cas où il faut mettre le système en mouvement, la friction nuit au moteur. Dans le cas où seuls les servomoteurs sont utilisés, l'information sur le couple statique est donnée sans donner d'information sur le couple possible pour mettre le système en mouvement. Ce type de servomoteur vient en deux versions, l'une avec

un réducteur composé de roues dentées en métal et l'autre avec des roues dentées en plastique, le premier ayant des spécifications de couple plus grandes. Ce qui est surprenant est que ce couple plus grand vient aussi de la plus grande friction présente dans le servomoteur et non pas uniquement du frottement sec dans les brosses du collecteur et des limites physiques des matériaux dont est constitué le réducteur.

Le résultat est que le couple qui peut être fourni par le ressort pour permettre l'équilibrage statique dans le cas extrême est abaissé. Pour permettre un ajustement du couple, plusieurs points de fixation sont possibles sur le fémur. De façon radiale le point d'attache peut avoir un déplacement de $\frac{\pi}{8}$ radian. Le ressort lui-même a aussi été déformé plastiquement pour s'assurer d'un meilleur ajustement.

Face à ces nouvelles contraintes, le cas orthogonal a été priorisé. Celui-ci nécessite des couples moindres. L'équilibre statique pour ce couple, combiné à la friction dans les réducteurs des servomoteurs, permet au robot de maintenir cette pose avec un nombre de pattes au sol allant de trois à six. Pour que le robot puisse maintenir sa position il faut que la valeur résultante du couple fourni par le ressort et le couple appliqué par la charge venant du robot soit inférieure au couple pour vaincre la friction. Ceci est intéressant du fait que les performances en positionnement et la stabilité de la pose ne sont pas fortement pénalisées par le nombre de pattes au sol.

Les polygones de support qui seront priorisés sont donc semblables à celui de la position orthogonale. Dans le cas où les conditions du terrain obligerait des géométries de polygone de plus grandes dimensions, il deviendrait alors possible de réduire la vitesse d'avance du robot et d'utiliser un nombre plus grand de pattes dans le polygone.

Hexapode est donc en mesure de modifier sa démarche pour lui permettre de franchir toute sorte de terrain. L'autonomie du robot passe par la minimisation de sa consommation énergétique. Dans cette section, il a été présenté, l'équilibrage statique partiel qui remplit cette tâche a été expliqué. Les ressorts en torsion combinée à la friction présente dans les réducteurs des servomoteurs permettent le maintien de poses statiques avec un nombre variable de pattes au sol, c'est ce qui était désiré.

4.5 Adaptabilité

Dans cette section sont présentées les performances d'Hexapode par rapport au critère d'adaptabilité. Au stade de développement atteint lors de la rédaction de ce mémoire, le prototype est en mesure de se diriger et d'atteindre les objectifs fixés en recevant un minimum de consignes externes.

L'intelligence artificielle donnée au robot permet l'adaptabilité. En s'inspirant de principes biomimétiques, cette intelligence se distribue sur plusieurs niveaux. Les niveaux primaires sont comparés au système nerveux de locomotion chez les vivants. C'est la base de ce qui permet les mouvements et leur coordination. Les niveaux supérieurs, impliquent la capacité de gérer des concepts abstraits. Les tâches liées à ces niveaux d'intelligence sont la possibilité de générer des trajectoires après avoir reconnu les conditions de l'environnement. Ce niveau n'a pas été implanté sur Hexapode principalement à cause des limitations de traitement et de mémoire de l'unité de calcul qui joue le rôle de cervelet, la capacité de celui-ci déborde lorsque vient le temps de remplir aussi les tâches nécessitant un cerveau. Les grandes lignes derrière ce niveau ont tout de même été élaborées et seront présentées. Au départ du projet, le niveau d'intelligence abstrait avait cependant été jugé d'une importance moins capitale.

Les objectifs propres au projet Hexapode portent plus spécifiquement sur le développement d'une plate-forme de robot marcheur ayant des capacités mécatroniques permettant l'étude de la marche. Le robot a été conçu de façon modulaire, et a donc la possibilité d'être amélioré par la suite.

Ce qui importe le plus dans le cas du critère d'adaptabilité est l'implantation d'une paramétrisation des déplacements du robot, soit le contrôle de sa vitesse, de son orientation et de sa stabilité. Idéalement, la poursuite des travaux sur cette plate-forme permettrait d'être en mesure de construire une représentation du terrain et d'agir directement sur les paramètres d'entrée de la locomotion.

Les composantes mécaniques et électroniques qui mènent à une forme d'intelligence artificielle du niveau locomoteur sont implantées au robot pour lui permettre de se déplacer en marchant, ce qui sera décrit plus loin dans la présente section. Ces composantes, aidées de la programmation hiérarchique, permettent de prendre les décisions

relatives à l'atteinte des objectifs. L'adaptabilité du robot se caractérise par sa capacité de pouvoir générer plusieurs démarches. Ces démarches ont des caractéristiques spécifiques et seront expliquées en détail plus loin.

La programmation de l'unité de calcul permet de remplir des tâches plus ou moins complexes, allant du positionnement des extrémités des six pattes par la résolution du problème géométrique inverse en temps réel, jusqu'à la coordination des polygones de support et de retour par l'implantation des lois de Wilson. Le tout en permettant de modifier ses modes de fonctionnement à l'aide de l'intégration serrée des paramètres de la navigation et des mouvements du robot.

4.5.1 Système de capteur

Du point de vue biomimétique, un des paramètres de l'intelligence est la capacité de se représenter son environnement. Pour que cela soit possible, l'information venant des sens est essentielle. Un cerveau fonctionnant en vase clos correspondrait à faire du contrôle en boucle ouverte : rien n'indique le résultat de ses actions. Également, il est très difficile d'apprendre si aucun renforcement n'était fait pour montrer que la bonne voie a été choisie.

Comme le projet de robot marcheur est de développer une créature artificielle qui s'approche des créatures vivantes, Hexapode s'est vu implanter des capteurs. Ces sens, actuellement rudimentaires, ont la possibilité d'être augmentés et traités par un système nerveux artificiel, ceci pour lui permettre ultimement de prendre conscience de son environnement.

4.5.1.1 Orientation des membres des pattes

Le positionnement des pattes se fait à l'aide de la résolution du problème géométrique inverse qui génère trois consignes à envoyer aux trois moteurs de chacune des pattes. Cependant, dans ce processus, aucun retour d'information n'est fait à l'unité de calcul, ce qui a déjà été énoncé à la section 4.2.2 sur les actionneurs. Dans un servomoteur, un potentiomètre interne, branché comme diviseur de tension, fournit un signal qui permet de connaître la position angulaire des moteurs. Il serait envisageable de modifier

physiquement les moteurs pour permettre au robot de connaître cette information.

En regardant à l'oscilloscope les signaux du potentiomètre obtenu lors du fonctionnement normal du servomoteur, les résultats s'avèrent différents de ce qu'on pourrait s'attendre. L'alimentation du diviseur de tension qui permet une mesure de sortie n'est pas constante dans le temps, mais elle se fait lorsque le circuit interne de commande a besoin d'information. Souder des fils et faire l'acquisition des signaux du potentiomètre par l'unité de calcul implique le débranchement du potentiomètre du circuit de contrôle interne. Le résultat est que le servomoteur doit alors être modifié pour remplacer le potentiomètre par deux résistances de valeur identique dont la somme est égale à la valeur du potentiomètre. Le servomoteur est alors commandé en envoyant des échelons dont la durée est proportionnelle à l'énergie qui doit être envoyée au moteur. Un processus doit donc rouler dans l'unité de calcul pour déterminer la commande à envoyer en fonction de la consigne et de la valeur lue sur le potentiomètre.

Un contrôleur proportionnel dérivatif intégral (PID) pourrait être implanté.

$$\Delta = \theta_d - \theta_a \quad (4.46)$$

$$P = \lambda_P \times \Delta(t) \quad (4.47)$$

$$I = \lambda_I \times \sum_{i=0}^{i=t} \Delta(i) \quad (4.48)$$

$$D = \lambda_D \times (\Delta(t-1) - \Delta(t)) \quad (4.49)$$

$$Valeur^{PWM} = \sum (P + I + D + K_{calibration}) \quad (4.50)$$

Où θ_d est la valeur d'orientation désirée, θ_a est la valeur d'orientation actuelle, « P » est la valeur liée au terme proportionnel, « I » est la valeur associée au terme intégral, « D » est celle en lien avec le terme dérivatif. Les variables « λ » sont les coefficients respectifs menant aux valeurs précédentes. « $Valeur^{PWM}$ » est lié à la largeur de l'échelon qui sera envoyé comme consigne du servomoteur. « $K_{calibration}$ » est un biais qui est défini lors de la calibration des servomoteurs pour recentrer l'orientation à zéro degré.

Par analogie, la valeur proportionnelle (P) est comme la constante d'un ressort. Plus le ressort est loin de sa position d'équilibre plus la force de rappel est forte. La valeur dérivative (D) peut être comparée à l'amortissement du système. Plus la variation de

l'erreur nécessaire est grande, plus grand sera l'amortissement. Les dépassements de la consigne sont atténués par la diminution de l'erreur d'orientation.

La valeur intégrale (I) est ce qui permet de corriger l'erreur statique en ajoutant volontairement une instabilité dans le système en vue d'obtenir la bonne orientation. Lorsqu'une erreur statique est présente, cette erreur est sommée et lorsque l'impact du terme intégral devient assez grand pour faire bouger le système, ce dernier se déplace à une nouvelle orientation. Cependant, la nouvelle orientation n'est pas nécessairement celle désirée. Le système peut alors osciller entre des positions autour de celle désirée. Dans le cas des systèmes ayant besoin d'un couple important pour démarrer, la variation de commande à envoyer au servomoteur pour permettre le déplacement est élevée. Lorsque ce système commence à se déplacer, si le terme intégral ne devient pas rapidement nul ou que l'amortissement du terme dérivatif n'est pas suffisant pour ralentir l'impulsion du mouvement, un dépassement survient par rapport à la consigne. Dans le pire des cas, le système s'emballe et diverge puisque la somme des erreurs est de plus en plus grande d'une direction à l'autre. Le terme intégral est donc capricieux, mais fort utile. C'est une perturbation contrôlée introduite dans le but de corriger l'erreur statique.

Dans le cas du contrôleur interne des servomoteurs, c'est justement le terme intégral qui est absent. Pour fournir le couple nécessaire au maintien de l'orientation, les termes proportionnels et dérivatifs ont besoin d'une erreur de position. L'absence du terme intégral est donc néfaste sur la qualité de l'orientation finale et du couple pouvant être produit avec les moteurs. Cependant, son absence permet d'éviter les problèmes de divergence. C'est d'autant plus utile puisque le servomoteur ne peut effectuer une rotation de plus de π radian. Des limitations mécaniques sont implantées alors qu'il n'y a pas de limitation électronique. Il est facile d'imaginer qu'un système qui diverge aurait comme conséquence l'autodestruction du servomoteur. Puisque les réglages des coefficients du PID ont été déterminés lors de la conception du servomoteur, il est donc impossible de la modifier directement cette correction pour optimiser la performance du système complet. Ces coefficients ont donc été placés à des valeurs conservatrices pour permettre un fonctionnement dans toutes les applications où ces servomoteurs sont utilisés.

La mesure du potentiomètre à donc été mise de côté pour l'instant puisqu'elle demande une charge de calcul supplémentaire à l'unité de calcul qui doit en tout temps

faire l'acquisition des données du potentiomètre et la résolution de la commande qui sera envoyée aux servomoteurs. De plus, les résultats obtenus sans remplacer le contrôleur interne des servomoteurs sont suffisants pour qu'Hexapode puisse marcher sans montrer de faiblesse ou d'instabilités. La possibilité de modifier les servomoteurs existe, mais elle sera exploitée dans des travaux de raffinement du prototype.

4.5.1.2 Chargement de l'extrémité de la patte

La fermeture de la boucle de commande de la position de l'extrémité de la patte passe par la mesure la pression résultant de la force normale dont la composante est selon la direction de la longueur du tibia. L'information en lien avec le chargement de pression au bout de la patte est aussi nécessaire afin que Hexapode puisse s'adapter aux conditions du terrain. Pour ce faire, un système comprenant un capteur de force à résistance variable est utilisé. Un poussoir en forme de champignon est installé à l'intérieur de la gaine de protection du bout de la patte et vient faire pression sur le capteur FSR lorsqu'il y a contact au sol.

L'extrémité des pattes est recouverte d'une gaine de caoutchouc antidérapant afin de maximiser leur adhérence. Ces gaines sont vendues commercialement pour faciliter la manipulation de feuille de papier avec les doigts. Elles présentent une texture intéressante qui est pourvue de petites pointes qui augmentent leur adhérence à la plupart des surfaces. Comme l'extrémité des pattes est sensiblement de taille semblable à celle d'un doigt, cette gaine est donc tout à fait appropriée.

Le mécanisme de mesure de la pression est composé d'une pièce en forme de champignon renversé qui est introduite dans la gaine. Le champignon appuie par l'entremise d'un disque de caoutchouc sur le capteur à résistance variable qui est introduit dans le tibia. Pour que le système reprenne sa place, un anneau d'étanchéité torique en caoutchouc est installé entre le champignon et le tibia. L'ajustement de la valeur lue par le système se fait en plaçant une charge d'un kilogramme sur le tibia et en sablant le champignon pour que la mesure soit de trois volts l'anneau d'étanchéité est comprimé. La figure 4.12 montre le mécanisme de mesure de la pression dans le détail.

L'élément servant à prendre les mesures est un capteur à résistance variable en fonction de la pression (FSR), dont une image est présentée à la figure 4.13. Il est fait de plastique flexible et de matériaux semiconducteurs. Lorsqu'aucune pression n'est

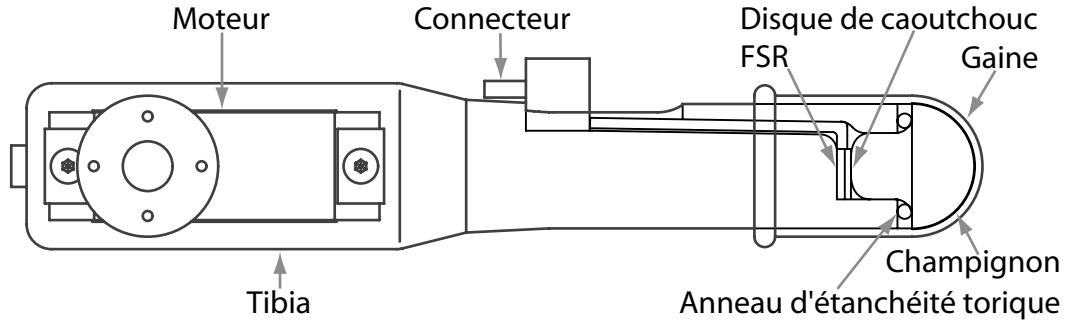


FIG. 4.12 – Mécanisme de mesure de la pression à l'intérieur d'une patte.

appliquée, le capteur montre une résistance d'environ un mégaohm. Lorsqu'une pression est appliquée sur la surface circulaire du bout, la résistance baisse. Les caractéristiques de ce capteur se retrouvent à l'annexe D.2.

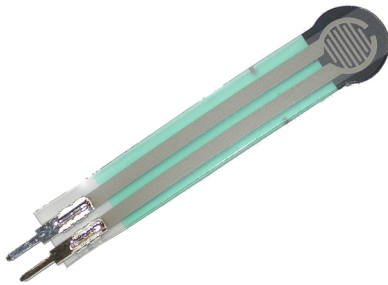


FIG. 4.13 – FSR

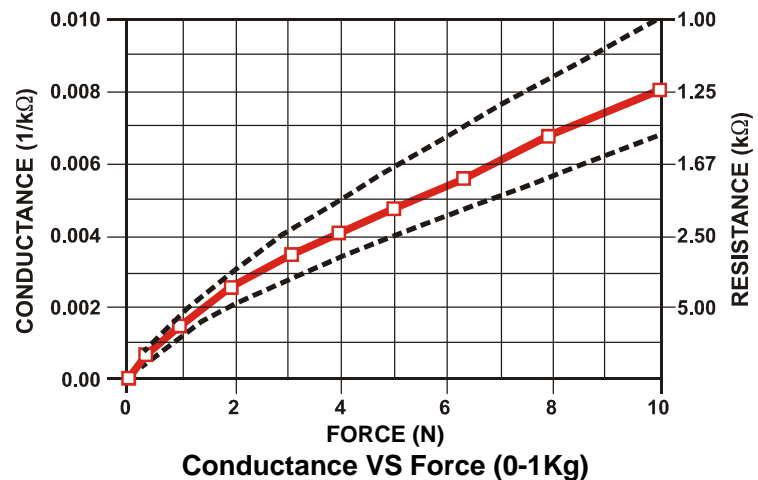


FIG. 4.14 – Valeurs de la conductance

La figure 4.14 montre la courbe approximative de la valeur de la résistance qui est attendue pour des charges allant jusqu'à $10N$, soit un environ un kilogramme. Les pointillés représentent la plage de répétabilité qui s'agrandit proportionnellement à l'augmentation de la charge. Il est à noter que les propriétés de mesure des capteurs de ce type sont plus qualitatives que quantitatives. En effet, les capteurs subissent une dérive dans le temps. Les mesures faites à l'aide de ces capteurs seront utilisées pour une comparaison entre la charge des pattes et pour déterminer s'il y a présence de contact ou non avec le sol ou un objet. Bien que leur précision et la répétabilité des mesures soient limitées, leur utilisation est très serviable dans le cadre du projet en cours. Du

point de vue biomimétique, il est également difficile aux vivants de déterminer avec précision les charges qu'ils transportent. Tout comme Hexapode, les vivants font une analyse qualitative de l'évaluation du poids par le sens du toucher.

4.5.1.3 Autres capteurs

Pour permettre au robot de connaître son environnement, des capteurs de proximité de type phototransistor pourraient être intégrés au robot. Des diodes led infrarouges éclaireraient la scène par impulsion, les phototransistors montés en périphérie du robot capteraient les différents niveaux de lumière réfléchié selon la proximité des obstacles. Ces signaux ne permettent pas une vue à très longue distance, mais sont suffisants pour permettre au robot de voir les obstacles qu'il aura à franchir. Des capteurs d'orientation pourraient également être installés pour donner plus d'information sur les conditions du terrain. Un inclinomètre, c'est-à-dire un accéléromètre qui mesure l'accélération gravitationnelle, permet d'orienter un plan de navigation inertielle. Enfin, un magnétomètre permettrait de situer le nord magnétique, ce qui aiderait le robot à gérer ses déplacements sur de longues distances. Ces systèmes de capteurs pourraient devenir de nouveaux sens pour Hexapode. Cette liste n'est pas limitative, seule l'imagination l'est.

4.5.2 Microcontrôleur

Hexapode doit être en mesure de s'adapter aux situations qu'il aura à faire face. Il faut qu'il puisse prendre des mesures, gérer le déplacement de ses pattes et modifier ses séquences de fonctionnement par rapport aux consignes de haut niveau afin de se donner une démarche permettant de s'adapter à l'environnement.

L'unité de calcul est l'élément qui gère toutes les informations internes au robot. Elle est la composante qui fait le lien entre le robot et l'environnement par l'acquisition des capteurs. Elle gère aussi le contrôle des actionneurs. De plus, elle s'occupe de la coordination entre les pattes. Les paramètres de la coordination des pattes du robot sont fonction de la représentation que se fait le robot de son environnement. Toutes les opérations nécessitant des algorithmes passent par l'unité de calcul. Il est important que le traitement des informations se fasse le plus possible à bord du robot pour que le

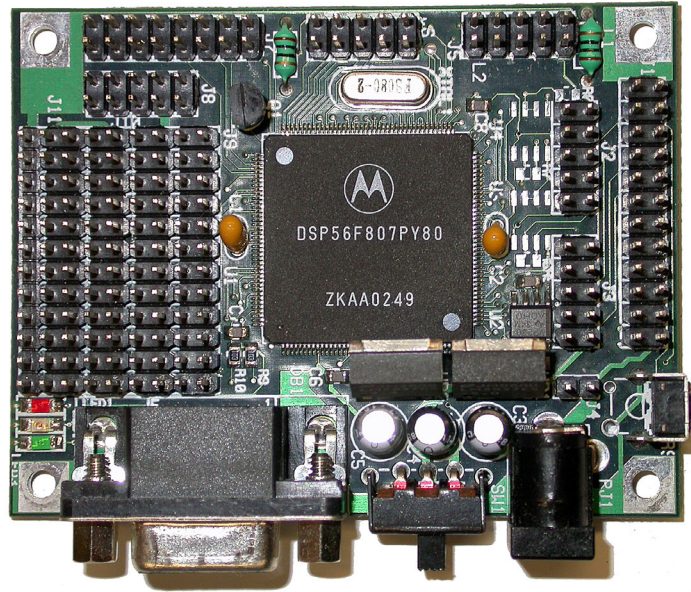


FIG. 4.15 – Servopod avec microcontrôleur DSP56F807.

robot soit le moins dépendant possible des ressources extérieures. Les dimensions et la consommation énergétique de l'unité de calcul sont donc des paramètres importants.

L'unité de calcul qui sert de cerveau et de cervelet à Hexapode est un microcontrôleur de modèle DSP56F807 [25] fabriquée par Freescale (autrefois Motorola). Ce microcontrôleur est spécialisé aussi bien dans le traitement de signal (Digital Signal Processing), que dans le calcul d'opération de commande. La figure 4.15 montre le microcontrôleur qui est monté sur un circuit imprimé. L'ensemble est vendu sous le nom de Servopod par la compagnie New Micros Inc.. Cet assemblage est fort pratique puisque les pattes de la puce électronique sont de très petites dimensions et que la carte fait le pont entre ces dernières et des connecteurs plus gros et plus accessibles. De plus, un connecteur d'interface sérielle RS-232 est présent sur la carte, ce qui permet un branchement facile.

Le processeur fonctionne avec des mots de seize bits à une cadence de quatre-vingts mégahertz et est capable de gérer 40 MIPS. L'environnement de développement utilisé est CodeWarrior de Metrowerk. La programmation s'effectue en langage C, ce qui permet d'obtenir des niveaux de complexité beaucoup plus intéressants qu'avec la programmation directement en langage Assembleur. Les programmes sont donc plus transportables et il est possible de les faire fonctionner sur d'autres plates-formes en ne supposant que des modifications mineures. Le processeur exécute toujours les programmes

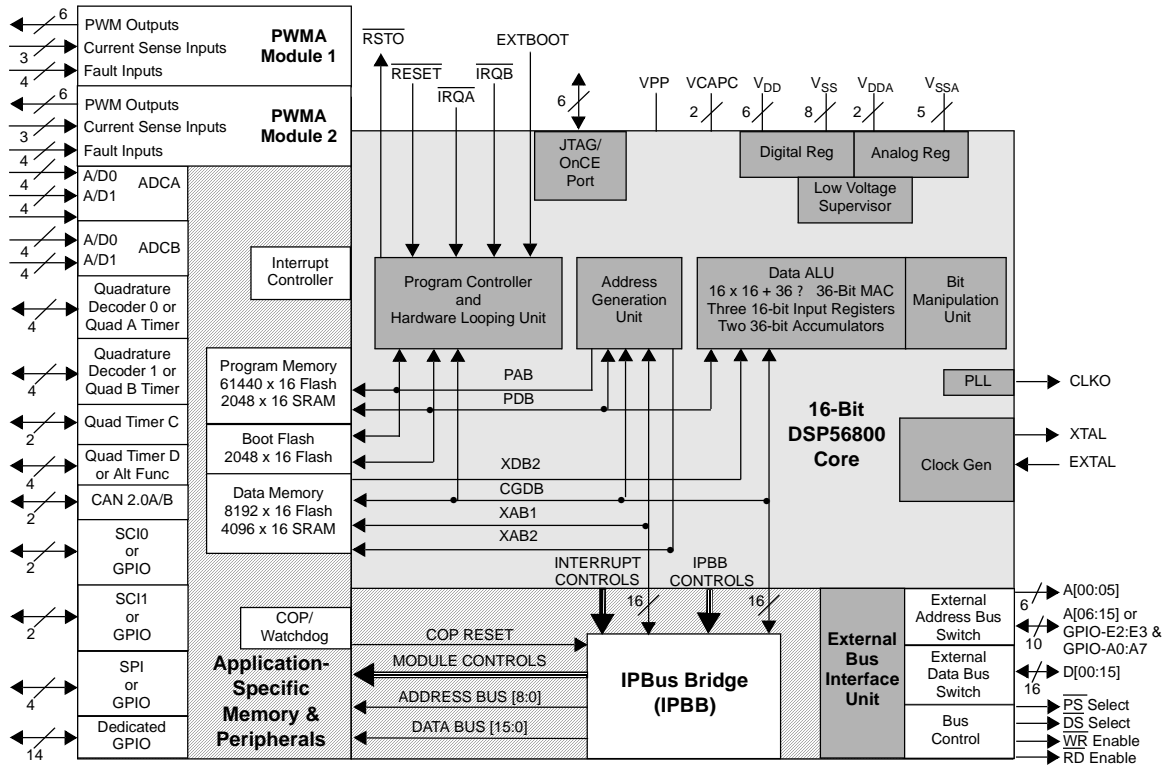


FIG. 4.16 – Diagramme des périphériques du microcontrôleur.

en langage assembleur. La compilation et l'exécution du programme en langage C génère le programme en langage assembleur correspondant. L'avantage du langage C est qu'il facilite la tâche du développeur en permettant l'utilisation d'instruction compacte et performante.

Comme le montre le diagramme de la figure 4.16, plusieurs périphériques sont déjà inclus dans le microcontrôleur. Ceux qui s'avèrent particulièrement utiles servent d'interface entre le corps du robot, son l'environnement et le calculateur, permettent d'exécuter des routines. Plus spécifiquement, les périphériques suivants se retrouvent dans le microcontrôleur.

- Seize convertisseurs transforment la mesure de tension analogique en valeurs digitales. Ils permettent de faire la mesure des capteurs externes par exemple les capteurs de mesure de pression du bout des pattes. Quatre conversions peuvent être faites simultanément.
- Trente-deux entrées et sorties digitales (i/o) sont présentes pour permettre le contrôle et la communication avec des périphériques externes.

- Des générateurs d'échelon sont disponibles pour permettre la commande des servomoteurs. Douze générateurs PWM dédiés sont présents.
- Seize temporisateurs à sortie digitale sont aussi disponibles pour pouvoir contrôler un total de vingt-six servomoteurs RC.
- Un autre périphérique intégré bien utile, est la communication sérielle par protocole RS-232 par 2 canaux indépendants. Cela permet de communiquer, de façon facile et efficace avec un ordinateur externe, bien que ce mode de communication ne soit pas à toute épreuve.
- Deux canaux d'interruption externe sont directement utilisable et d'autres interruptions internes peuvent être gérées.
- Bien qu'il ne soit pas utilisé pour l'instant avec Hexapode, quatre compteurs de quadrature dédiés sont présents pour compter les échelons de sources externes. Il est aussi possible de partager les compteurs pour obtenir deux canaux supplémentaires.

Tous ces périphériques requièrent bien entendu une programmation de bibliothèques afin de gérer l'accès aux adresses-mémoires correspondantes. La programmation de ces bibliothèques et des autres fonctions de plus haut niveau utilisées pour les tâches complexes implique une optimisation en vue de diminuer le temps de calcul requis et de réduire l'espace mémoire nécessaire. Il est à noter que les variables en format à virgule flottante ne sont pas directement supportées par l'unité de calcul. La mémoire disponible est également limitée : 60K en mots de 16 bits dans la mémoire flash EEPROM dédiée au programme et 2K en mots de 16 bits dans la mémoire RAM. La quantité de mémoire peut également être augmentée par l'ajout d'une carte «Smart media» pouvant aller jusqu'à 128 mégaoctets.

Les bibliothèques permettant l'utilisation des périphériques du microcontrôleur ont d'abord été optimisées par rapport à celles ayant été développées et retravaillées dans le cadre du cours de mécatronique du département de génie mécanique de l'Université Laval (automne 2002) mettant en contexte l'intégration de notions d'électronique et de programmation, associées à des notions mécaniques. Ce cours permet notamment d'intégrer et d'appliquer les concepts liés à la commande et à l'algorithmie de système mécanique. Le microcontrôleur utilisé pour ce cours est le Motorola DSP56F803. Il est dans la même famille que celui qui est utilisé dans le présent projet. Le DSP56F803 est semblable au niveau du type de périphériques sauf que leur quantité est moins abondante.

Ce sont les adresses-mémoires des registres des périphériques qui diffèrent dans la programmation du DSP56F803 et le DSP56F807. La traduction des librairies a donc été faite pour qu'elle fonctionne avec le microcontrôleur qui sera utilisé par Hexapode. La validation des librairies s'est avérée un moyen efficace pour les optimiser et les adapter à leur cadre actuel d'utilisation.

4.5.3 Programmation hiérarchique

La clef du développement du système de contrôle d'Hexapode est la programmation hiérarchique. Cette approche consiste à développer les différents niveaux d'algorithmes correspondant à divers niveaux de fonctionnement indépendants. Ceci revient à construire différentes couches indépendantes qui s'imbriquent hiérarchiquement. Chacune des couches est construite pour être autonome et indépendante des autres à l'exception de paramètres qui transigent d'une couche à l'autre.

Le fait que les couches soient empilées présente un avantage majeur lors de la correction des erreurs dans les programmes. Chacune des couches peut être isolée des autres. Ses entrées sont émulées aux valeurs dont il est possible de connaître le résultat. Les sorties de la couche ainsi que les étapes intermédiaires pour les atteindre sont vérifiées dans le détail. Cette façon de programmer présente aussi des avantages au niveau de la prévention des bris pouvant survenir en cours de résolution des erreurs. Puisque les sorties des couches peuvent être analysées de façon individuelle, les éléments matériels peuvent être remplacés par l'affichage des variables de sortie de la couche ou mesurées en temps réel par un oscilloscope. Ensuite, toutes les composantes peuvent être branchées et fonctionner selon les résultats escomptés.

La méthode de programmation hiérarchique s'applique aussi aux sous-éléments des couches. En effet, le développement de chacun des fonctions et des processus utilisés le cadre du projet s'est fait étape par étape. Chaque fois qu'un élément nouveau était créé ou intégré, la correction des erreurs de programmation et de concept était faite avant d'entreprendre d'autres actions.

Puisque l'unité de calcul a une capacité finie de traitement de données et que les tâches doivent s'exécuter en temps réel, il s'avère essentiel que les informations traitées

soient réduites à l'essentiel. L'échelle de la charge des processus devant être exécutés en temps réel par l'unité de calcul a été élaborée. Tous les processus n'ont pas la même importance et la distribution des priorités est un des aspects qui favorise une meilleure utilisation des ressources disponibles.

Les tâches ayant trait au processus matériel sont habituellement sensibles à leur disposition temporelle, leur priorité est donc critique. Tandis que les processus de plus haut niveau liés aux calculs du positionnement et de coordination des pattes le sont moins. Dans le pire des cas, une des itérations sera perdue et la nouvelle consigne ne sera donnée qu'une itération plus tard. Les processus d'interface avec l'utilisateur sont les moins critiques. Dans le cas d'informations plus pressantes à traiter, le temps de calcul utilisé habituellement pour l'affichage des données de navigation sera redirigé. La gestion des priorités s'est réalisée avec succès, les processus nécessaires pour effectuer la marche du robot ont pu être gérés convenablement sur le prototype.

La programmation hiérarchique s'inscrit sous le critère adaptabilité en permettant à de nombreuses situations d'être gérées. L'algorithmie implantée dans Hexapode doit être robuste puisque les actions se passent en temps réel et que les mauvais fonctionnements pourraient être fatals pour les systèmes physiques. Tous les processus doivent être en état de fonctionner sans erreur. Les situations d'urgence sont à éviter et lorsqu'elles se produisent, il doit y avoir une façon de les arrêter rapidement. En cas de problème majeur, l'alimentation des servomoteurs est coupée pour limiter les dégâts. Après vérification de l'état des opérations, le système est redémarré. Si le problème persiste, le programme principal est retéléchargé dans le microcontrôleur.

Chacune des couches de la programmation hiérarchique crée un lien étroit entre des choses distinctes. Pour atteindre les objectifs, les problèmes sont décomposés en plusieurs étapes au lieu de tenter de tout régler en un seul coup. Cette approche minimise les risques d'erreur et mène à la réutilisation de sections du programme pour résoudre des situations semblables. L'atteinte des objectifs est constituée de plusieurs règles simples menant à la génération de comportements découlant des processus. La complexification se fait progressivement durant l'atteinte des niveaux de compétence, une plus grande stabilité de chacune des couches est donc obtunue.

La figure 4.17 montre l'organigramme de l'algorithmie utilisée pour Hexapode. Trois couches distinctes sont représentées :

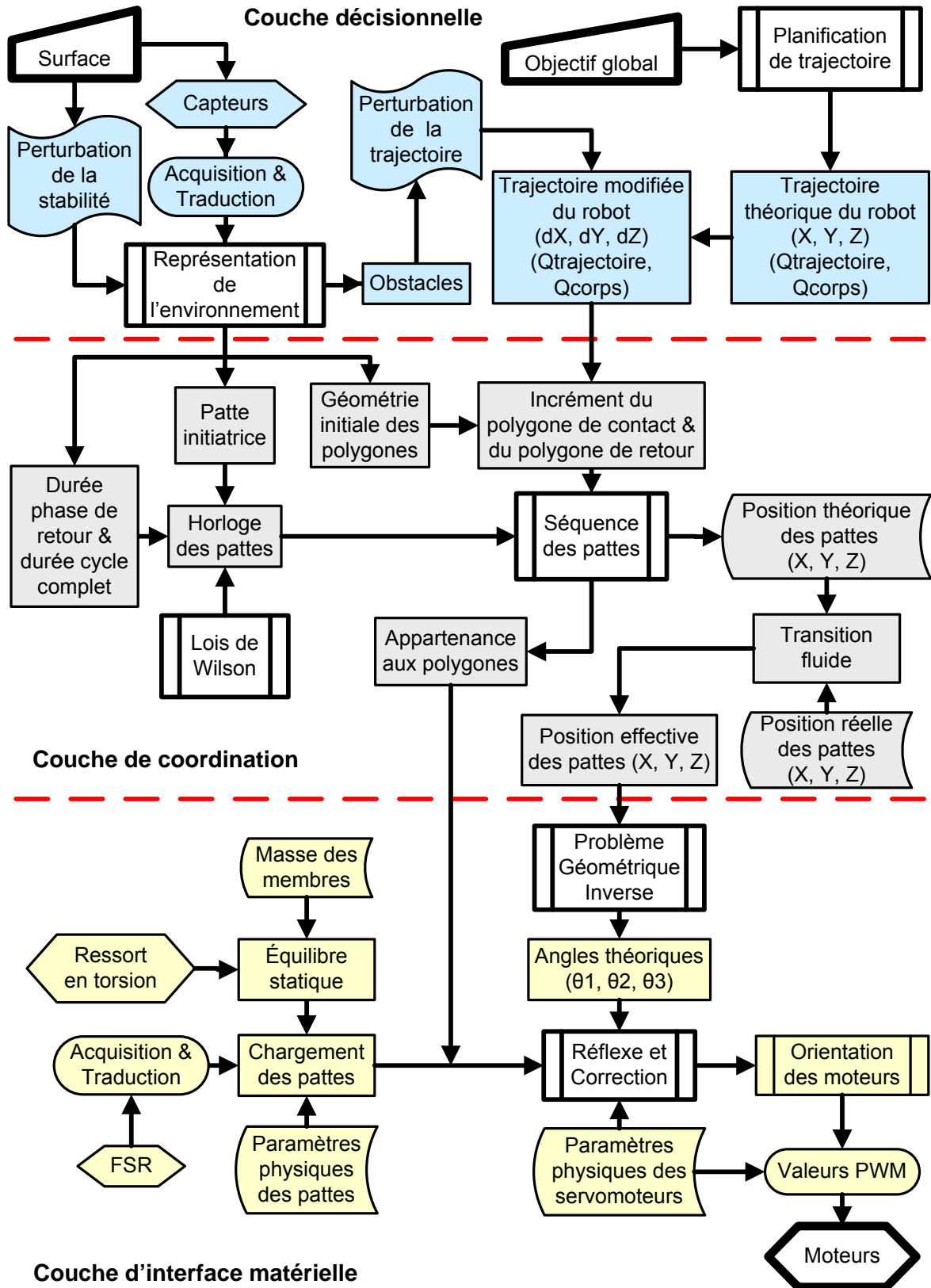


FIG. 4.17 – La programmation hiérarchique.

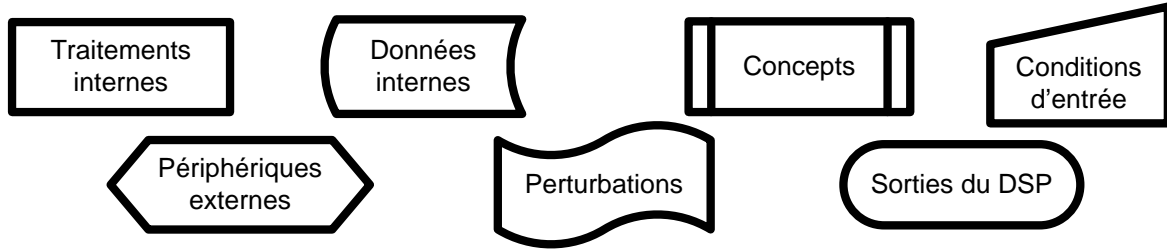


FIG. 4.18 – Légende des différents type de boîtes

- L'étage du bas de la figure 4.17 présente la couche d'interface matérielle. Cette couche est la base par laquelle l'informatique discute avec le matériel. Le fonctionnement de cette couche est critique, sans elle aucune action réelle ne peut être accomplie, elle est le système nerveux. Cette couche sera présentée dans le détail à la sous-section 4.5.4.
- L'étage du centre de la figure 4.17 est la couche de coordination. Cette couche gère les polygones de retour et de support. Selon les paramètres de navigation, elle détermine quelles pattes font partie des différents polygones. Finalement, elle génère les positions cartésiennes nécessaires aux extrémités des pattes. Cette couche est comme le cervelet chez les vivants. La couche de coordination sera expliquée en profondeur à la sous-section 4.5.5.
- L'étage du haut de la figure 4.17 montre la couche décisionnelle. Cette dernière dirige le comportement d'Hexapode en définissant les paramètres de navigation à appliquer. La représentation de l'environnement que se fait le robot est faite par cette couche. Elle est également responsable de la génération de trajectoire pour répondre aux objectifs de haut niveau demandé par la mission que remplit Hexapode. Cette couche n'est pas actuellement implantée, mais les concepts y étant relié ont été planifiés, le tout sera présenté à la section 4.5.6.

Étant donné l'aspect modulaire des fonctions et des processus, il est aisément possible d'ajouter de nouveaux éléments. En effet, lors du développement de l'architecture de l'algorithme, un souci particulier a été porté à l'insertion d'élément de mise à jour en cas d'ajout de nouvelles composantes. Cela augmente une flexibilité à la programmation en vue de permettre une continuité au travail qui a été fait sur Hexapode. Le développement d'un robot marcheur pourrait être le projet d'une vie et n'est donc jamais réellement terminé. Tout comme dans la nature, où des générations d'individus génèrent l'évolution progressive des vivants, des améliorations sont toujours envisageables. C'est la même chose dans le domaine de la robotique. L'idée derrière le développement de tout robot devrait être de lui permettre une possible évolution.

4.5.4 Couche d'interface matérielle

La couche d'interface matérielle permet le contact du contrôleur avec les composants physiques du robot. Elle gère tout ce qui a trait au système électronique et électromécanique. C'est en quelque sorte le système nerveux d'Hexapode. Cette couche se loge au plus creux de la programmation, ayant influence sur les systèmes moteurs et les systèmes d'acquisition des capteurs. Les processus de cette couche sont sensibles à leur positionnement temporel. En effet, il est primordial pour assurer le bon fonctionnement de tous les systèmes que les processus soient prioritaires par rapport à ceux des autres couches.

4.5.4.1 Générateur d'échelon PWM

Les sorties de la couche d'interface matérielle sont les dix-huit consignes envoyées au servomoteur. Cette consigne est générée par le périphérique de génération d'échelon du microcontrôleur. La manière dont fonctionne le générateur PWM est simple. Il faut d'abord initialiser le périphérique en définissant les paramètres aux adresses propres au registre. Un des paramètres d'entrées du générateur PWM est le module, c'est-à-dire le compte des événements. Ces événements arrivent tous les multiples de un, deux, quatre ou huit coups de l'horloge du processeur. La valeur que prend le générateur PWM correspond au nombre d'événements qui sont comptés en plaçant la sortie en état haut, soit la durée de l'échelon.

La fréquence la plus basse pouvant être atteinte avec le plus grand module est en mode «center align» (compte vers le haut et compte vers le bas pour doubler le temps du cycle) avec la plus grande division d'événement (8) est 76 Hz. Les servomoteurs nécessitent une fréquence de rafraîchissement de 50 Hz, ce qui est encore plus lent que cette fréquence. L'astuce utilisée est de compter plusieurs sous-périodes pour faire une période complète. Plus de détails sur la manière de contrôler les servomoteurs ont été présentés à la section 4.2.2 sur la mobilité, notamment l'explication de la correspondance entre l'orientation et la durée de l'échelon à envoyer comme consigne.

La temporisation de tout l'algorithme du robot découle d'une seule source. Bien que dix-huit servomoteurs soient utilisés, cette temporisation vient d'une seule interruption qui est lancée à chacune des fins de période du canal A0 du générateur PWM. Cette

interruption se produit précisément à 250 Hz. Puisqu'elle est prioritaire sur tous les autres processus, les tâches qui sont effectuées lorsqu'elle survient doivent être le plus succinctes possible. Une fois sur cinq, les valeurs PWM correspondant à la largeur d'échelon désirées est chargée, les autres fois que l'interruption se produit, la valeur zéro est envoyée pour ne pas générer d'échelon. L'interruption incrémente aussi une variable compteur qui servira dans le programme principal à effectuer les autres tâches selon des temporisations moins critiques.

4.5.4.2 Utilisation des compteurs

Comme uniquement 12 canaux de générateur PWM sont présents, certains des canaux du périphérique des compteurs servent aussi de générateur d'échelon. Leur fonctionnement est assez semblable à celui des générateurs PWM dédiés, à l'exception du fait que les comptes pour le temps de l'échelon et le temps mort sont additionnés pour donner le module total. Ceci présente l'inconvénient de générer un échelon (impulsion) d'une largeur d'un événement même si la valeur est posée à 0. Ceci est logiquement incontournable, cependant la présence des «latches» (annexe D.2) permet de contourner ce problème. Cette puce électronique placée à la sortie des canaux du générateur PWM et des compteurs, a la propriété de permettre la gestion de la valeur des sorties par une de ses pattes. Il est alors possible de s'assurer d'une valeur nulle de sortie en utilisant la puce avec une sortie digitale GPIO venant du microcontrôleur. Les résultats permettent le bon fonctionnement des servomoteurs sans problèmes.

4.5.4.3 Problème géométrique inverse

Une des entrées de la couche d'interface matérielle est la position cartésienne effective de chacune des six pattes. Cette couche a la responsabilité du positionnement des pattes. Pour ce faire, la position cartésienne des extrémités des pattes doit être convertie en des valeurs articulaires théoriques à l'aide de la résolution du problème géométrique inverse. L'algorithme de résolution qui est implanté a une structure qui ressemble à celle présentée à la section 4.2.1.1. Il importe que l'algorithme de résolution du problème géométrique inverse soit optimisé pour permettre sa résolution en temps réel. C'est pourquoi les variables du cas général se trouvant constantes ont été précalculées. Certains termes ont aussi été simplifiés sans perte de précision. La gestion des branches de solution passe par le fait de se conformer aux limites articulaires qui ont été imposées

lors du design. Au bout du compte, un maximum d'une seule des quatre branches de solution peut survivre. Les valeurs articulaires qui en émergent serviront d'angle de consigne théorique.

4.5.4.4 Librairie `math.h`

Comme les variables à décimale flottante ne sont pas directement supportées par le microcontrôleur, l'environnement de programmation ne supporte pas directement toutes les fonctions standard de la librairie «`math.h`». Ceci est un inconvénient de taille puisque les fonctions trigonométriques et les fonctions d'arrondissement font partie de celles qui ne sont pas supportées. Les librairies «`sinus`», «`arcsinus`», «`cosinus`», «`arccosinus`», «`tangente`», «`arctangente`» et «`atan2`», ainsi que «`ceil`» ont donc été reprogrammées en faisant des compromis entre la précision devant être atteinte, le temps de calcul et l'espace mémoire nécessaire. Les résultats obtenus se sont avérés satisfaisants puisqu'ils ne demandent pas une charge trop lourde de calculs.

4.5.4.5 Correction des consignes

Puisqu'il a été démontré que la consigne d'orientation des servomoteurs doit comporter une erreur pour qu'ils fournissent un couple, il est alors nécessaire de modifier artificiellement la valeur de la consigne pour que la position soit plus proche de la valeur désirée. Les conditions du chargement de la patte doivent pouvoir être approximées afin de permettre une meilleure orientation. Des modèles de chargement d'une plus ou moins grande précision permettent la correction.

Ce qui est actuellement implanté et fonctionnel sur Hexapode est l'ajout d'une valeur corrective proportionnelle à l'orientation de la patte. Dans tous les cas d'orientation, il faut effectuer la gestion du couple résiduel venant de l'équilibrage statique. Particulièrement lorsque la patte doit être relevée, elle doit vaincre le couple du ressort en torsion ainsi que la charge venant de son propre poids, faisant augmenter le couple nécessaire à l'articulation représentée par la variable articulaire θ_2 de la figure 3.2.

L'équilibrage statique implanté est fort utile lorsque la patte doit soutenir le robot, mais les effets du ressort sont néfastes dans le cas où la patte n'effectue pas de soutien. Les pattes devant recevoir la correction sont donc indiquées par la deuxième entrée de la

couche d'interface matérielle, soit le marqueur d'état des pattes. Dans cette correction uniquement, les effets du ressort en torsion sont tenus en compte. Les couples nécessaires pour vaincre le ressort sont de loin supérieurs à ceux nécessaires pour vaincre les effets de la gravité sur les membres devant être relevés. La correction liée à la masse de la patte dans des conditions de pose près de la configuration orthogonale standard (section 4.4.4.1) a donc été retirée de la correction pour alléger les calculs devant être effectués par le microcontrôleur.

Un modèle plus complexe de correction pourrait être implanté pour plus de précision. Un tel modèle de correction encore plus complet a été élaboré de façon théorique, mais n'a pas été implanté pour l'instant sur le robot. La valeur de la correction pourrait être ajustée à l'aide d'un traitement à l'aide de la logique floue et des réseaux de neurones sur des valeurs d'entrées internes et externes. Les mesures des capteurs de pression à l'extrémité des pattes (décrit à la section 4.5.1.2) permettraient de connaître de façon qualitative l'état du contact et donc donnerait une bonne idée de la force normale qui est appliquée. Les paramètres physiques du robot, tels que les forces issues des masses des membres, la pose et le nombre de pattes au sol dans le polygone de support servent également d'entrée au système de correction par logique floue et réseau de neurones. La sortie de ce système de correction jouerait directement sur la valeur de la consigne qui serait envoyée aux servomoteurs. Cette correction permettrait de corriger plus efficacement la consigne et permettrait une meilleure gestion de l'orientation des fémurs. De plus, ce type de correction pourrait avoir pour effet de permettre à Hexapode de franchir des terrains où le relief impose des points de contact qui ne sont pas seulement dans un plan. Les points des polygones de support théorique n'auraient pas à être modifiés, mais les pattes s'adapteraient d'elles-mêmes à la variation de hauteur.

Cependant, cette voie a été explorée de façon théorique seulement. L'implantation sur le robot et des expérimentations réelles n'ont pas encore été réalisées. Si tout était implanté, la charge de calcul résultante qui devrait être opérée en temps réel dépasserait la puissance de calcul du microcontrôleur et nécessiterait l'ajout d'une autre unité de calcul afin de distribuer la charge de tous les processus.

La figure de l'organigramme de l'algorithme 4.17 dresse le tableau de la couche d'interface matérielle en incluant la correction complexe expliquée plus tôt. La concentration des résultats se fait dans la boîte réflexe et correction qui est le système d'intelligence primaire de cette couche. Comme dans l'approche biomimétique, les réflexes

des vivants sont logés creux dans le système nerveux. La poursuite des travaux sur le robot Hexapode passe par la continuité en ce sens.

4.5.5 Couche de coordination

Pour que la marche soit stable et efficace, il faut effectuer la gestion des polygones de support et de contact. La couche de coordination remplit ce rôle et guide les interactions entre les pattes pour former des groupes de pattes. Il est primordial que les séquences des mouvements soient cohérentes pour que le déplacement du robot soit le plus fluide possible.

La gestion des polygones est le meilleur moyen connu pour s'assurer de l'atteinte des objectifs. Dans la littérature, le positionnement des pattes se fait principalement au niveau du contrôle de l'orientation des actionneurs. Dans le cas d'Hexapode, c'est la couche d'interface matérielle qui gère l'orientation, il devient alors possible de concentrer les efforts de la couche de coordination pour générer les paramètres des polygones. Ces paramètres sont le nombre de pattes qui doit servir d'appuis et la position des extrémités des pattes.

De façon analytique, des séquences adaptables de déplacement sont créées. Leur optimisation s'en trouve grandement facilitée puisque le travail se fait en position cartésienne, il est alors aisé de prévoir le résultat du déplacement de façon intuitive. De plus, le fait de séparer la gestion du positionnement des pattes indépendamment de l'orientation des servomoteurs, permettrait de changer le design mécanique du robot en ne réalisant que des modifications mineures à la couche de coordination.

La phase de génération des polygones survient aussitôt que les conditions de support sont modifiées. Un nouveau polygone est établi lorsqu'une patte tombe en contact ou perd son contact avec le sol. Cette situation peut arriver selon des paramètres qui sont contrôlés ou selon une situation imprévue. Lorsque les données sur l'état de contact sont modifiées, l'algorithme traitera différemment les déplacements de la patte en question.

4.5.5.1 Stabilité

Pour parvenir à des démarches stables, il faut que le centre du polygone de support soit le plus près possible du centre de masse de tout le robot. Ceci permet de répartir la charge de la façon la plus uniforme sur le robot, ce qui a été discuté à la section sur l'étude du chargement 4.4.4.1. L'éloignement de tous les points de contact du polygone de support par rapport au centre de masse du robot a aussi tendance à augmenter la stabilité. Cependant, ceci a un effet négatif puisque les couples à l'articulation θ_2 de la figure 3.2 deviennent plus grands proportionnellement à l'agrandissement des bras de levier qui soutiennent les points de contact et les articulations. Même si le polygone de support est plus grand et que la stabilité est plus grande, un compromis doit être fait pour limiter le couple aux actionneurs. Le polygone de support ayant été déterminé comme le plus efficace est celui qui se retrouve lorsque les pattes sont initialement dans la pose orthogonale standard (section 4.4.4.1). Hexapode a six pattes dont l'extrémité peut être positionnée selon les coordonnées cartésiennes. Pour être réalisable, la position doit se trouver dans l'espace défini par les limites articulaires qui ont été choisies lors du design du robot. Les déplacements des polygones sont donc limités et une optimisation doit être faite pour obtenir l'efficacité maximale des espaces de travail. C'est pourquoi le fait de passer par la pose orthogonale standard est une bonne idée, elle est le centre de cet espace de travail. En utilisant cette pose pour un des incréments du déplacement, il est possible d'optimiser le couple des actionneurs ainsi que les déplacements possibles par rapport à cette configuration. Bien sur, certaines situations imposent un polygone différent pour surmonter des conditions spécifiques, dans ce cas Hexapode peut modifier les paramètres de la géométrie pour se conformer aux conditions optimales venant de sa représentation de l'environnement.

4.5.5.2 Lois de Wilson

Les travaux de D.M. Wilson ont permis de construire un modèle descriptif catégorisant les divers modes de marche rencontrés chez les insectes (voir section 2.2.1 et [7]). Des lois ou règles de fonctionnement qui régissent l'ensemble des mouvements généralement rencontrés lors de déplacement d'insectes marcheurs à six pattes sur des surfaces horizontales. Ces expérimentations ont été faites avec des insectes vivants et permettent d'intégrer des notions biomimétiques dans le domaine des robots marcheurs. Les lois sont étonnamment simples, mais elles permettent d'obtenir des démarches performantes. La stabilité résultant de ces démarches peut être contrôlée de façon efficace

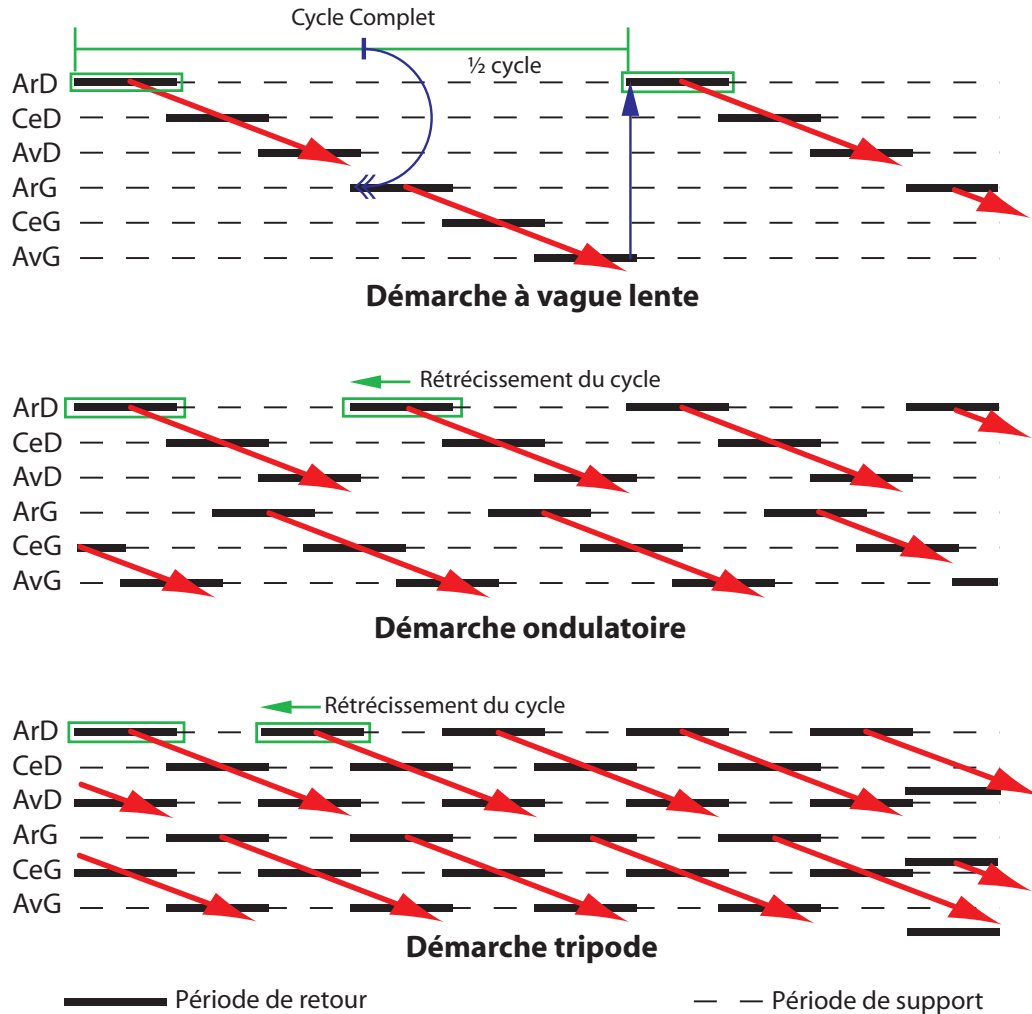


FIG. 4.19 – Représentation visuelle des séquences.

en changeant les paramètres d'entrée du modèle. L'intégration des travaux de Wilson par l'implantation des lois permet d'atteindre les objectifs fixés : la génération de démarches stables et adaptables selon les conditions du terrain.

Les lois de Wilson ont été remaniées pour être le plus facilement compréhensibles et intégrables au robot Hexapode. Les conséquences de chacune des lois sont différentes et aucune d'entre elles ne peut être négligée pour que tout fonctionne bien. La figure 4.19 montre des démarches résultant de l'application des lois.

Sur la figure 4.19, l'appellation «Ar» représente le segment arrière, «Ce» est pour le segment central et «Av» est pour le segment avant. Les lettres «D» et «G» sont

pour les côtés droit et gauche du robot, respectivement. Les cinq lois de Wilson sont énoncées ci-dessous, puis sont expliquées en fonction de la représentation visuelle des séquences telles qu'utilisée pour Hexapode.

1. La marche est constituée de deux phases : la période de support et la période de retour. La période de retour se produit lorsque la patte n'est intentionnellement plus en position pour soutenir le robot. La période de support se produit lorsque la patte soutient le robot et est donc intentionnellement dans une position pour être en contact avec le sol. Ces deux phases alternent et ne peuvent se chevaucher pour qu'un mouvement soit possible. L'addition des deux phases constitue un cycle complet. L'extension directe de cette règle est l'appartenance des pattes à un seul type de polygone à la fois. Sur la figure 4.19, cette loi entraîne une alternance continue des deux phases.
2. Une vague de phase de retour, sur un même coté, se propage des pattes arrières vers les pattes avants. Cette règle est représentée sur la figure 4.19 par les flèches descendantes diagonales. En tout temps, les séquences des pattes d'un côté gardent cette propriété, ce qui améliore la facilité d'implantation des séquences sur le robot. Cette règle donne l'allure temporelle de la démarche en imposant un ordre relatif d'actions entre les pattes situées sur un même côté du robot. Cette règle est celle qui donne le mouvement ressemblant à une vague de pattes qui se relèvent et qui se propagent dans la démarche des insectes. Cette vague est d'autant plus marquée lorsque le nombre de pattes est grand, notamment chez les millepattes.
3. La phase de retour d'une patte ne peut être amorcée avant que la patte directement en arrière de celle-ci n'ait atteint la phase de support. Cette condition est essentielle pour assurer un équilibre quasi statique au robot avant d'entamer la prochaine enjambée. Cette couche sert d'inhibiteur de la phase de retour afin de s'assurer qu'au moins trois pattes soient en contact avec le sol. Sur le prototype expérimental, cette règle est gérée intrinsèquement par le fait que la commande envoyée au servomoteur pour que la patte soit relevée, doit être plus grande que la commande pour soutenir le robot à l'aide de l'équilibrage statique. Puisque les transitions brusques de consigne doivent être évitées, la transition entre les phases de support et de retour s'effectue en ayant les deux pattes en contact avec le sol au même moment pour un bref moment. Bien que ce comportement ait tendance à faire varier légèrement la hauteur du corps du robot avec le sol, cette pratique est bénéfique puisqu'elle permet de minimiser les chances de glissement de l'extrémité de la nouvelle patte de soutien lors de son arrivée au sol.

4. Les pattes du même segment alternent en phase. Les segments sont les paires de pattes qui sont en avant, au centre ou en arrière du robot. L'alternance de la phase est le concept qui donne l'allure temporelle de la démarche, en imposant un ordre d'action entre les pattes du même segment du robot. Sur la figure 4.19, cette loi se montre par la flèche demi-circulaire qui fait correspondre le début de la période de retour de la patte arrière gauche (ArG) au milieu du cycle complet de la patte arrière droite (ArD). Cette loi permet physiquement de faire des pas. Chez les bipèdes, cette alternance est encore plus marquée puisque durant la marche, qu'une seule jambe ne peut être en phase de retour en même temps sans quoi la chute est inévitable. Pour la course, les conditions quasi statiques ne sont plus appliquées, mais remplacées par des conditions dynamiques.
5. La durée d'une phase de retour est constante. C'est la durée du cycle total qui varie. La durée de la phase de support diminue donc lorsque la fréquence des cycles augmente. En diminuant cette durée et en gardant la même distance à parcourir dans chacune des phases, la poussée s'exécute donc plus rapidement, ce qui accélère la démarche du robot. Le résultat du rétrécissement des cycles permet de changer la vitesse d'avance et la répartition des pattes entre les polygones. Cette loi a donc une influence sur le nombre de pattes qui sont au sol en même temps. Pour plus de stabilité, la vitesse du robot est diminuée et en même temps le nombre de pattes au sol est augmenté. Le mode le plus lent est la démarche à vague lente et met en phase de retour une seule patte à la fois, pour un ratio temporel d'un pour six. Le mode le plus rapide est la démarche tripode et met en phase de retour trois pattes en même temps. Le ratio de cette démarche est d'un pour deux, soit d'imposer des durées égales à la phase de support et à la phase de retour. Le rétrécissement du cycle revient visuellement à faire glisser les vagues de phase de retour de manière à les rapprocher les unes des autres. Le cas de la démarche tripode vient montrer un chevauchement de deux vagues qui se déroulent en même temps.

Un oscillateur donne la cadence qui synchronise les temps utiles dans la démarche. Chez les êtres vivants, cet oscillateur provient habituellement des pulsations cardiaques [8]. Dans le cas d'Hexapode, cet oscillateur est fonction d'une horloge interne permettant la gestion temporelle de l'exécution des programmes. Les paramètres modifiables pour agir sur le contrôle de la démarche sont donc : la durée de la période de retour et la durée du cycle total. Pour que le système puisse subir des variations ponctuelles, une horloge par patte est implémentée. Cette horloge donne un repère temporel à la gestion

des pattes. Dans des conditions normales, avec les paramètres ci-dessus, le choix d'une patte initiatrice et les lois de Wilson, il est possible de générer les horloges de toutes les pattes.

4.5.5.3 Paramètres de la marche

Les déplacements qu'effectuent les extrémités des pattes d'un même polygone sont fonction des entrées de la couche de coordination venant de la couche décisionnelle. Les paramètres complets de la marche sont :

- La direction de la marche se mesure par rapport à l'axe X du repère du robot. Bien qu'Hexapode ait des débattements articulaires permettant de plus grands déplacements dans la direction l'axe X, il lui est tout à fait possible d'effectuer des trajectoires dans toutes directions. La grandeur des pas doit seulement être ajustée en conséquence.
- La vitesse de l'extrémité des pattes durant la phase de retour. La vitesse de cette phase est ajustée puisque sa durée est constante, contrairement à celle de la phase de support qui change. Avec la vitesse et la durée, il est possible de gérer la grandeur des pas qui seront effectués. Dans le cas des essais expérimentaux, ce paramètre a été optimisé afin de tirer le meilleur parti des servomoteurs. Ces derniers travaillent mieux dans une certaine plage de vitesse. Dans le cas où l'orientation de la consigne soit atteinte, les servomoteurs s'arrêtent, rendant le mouvement moins fluide.
- La durée d'un cycle total a un effet sur la vitesse de l'extrémité de la patte durant la phase de support. Le ratio de la durée de la phase de retour et la durée du cycle total permet de déterminer les horloges de toutes les pattes.
- La rotation à effectuer durant la trajectoire est le paramètre qui sert au robot à appliquer des rotations au polygone de support et inversement au polygone de retour. L'axe de rotation utilisé est l'axe Z dans le repère du corps. La rotation totale s'effectue en imposant la matrice de rotation respective à tous les points de chacun des polygones.
- L'orientation du polygone de support par rapport au repère du corps du robot. Ce paramètre permet d'orienter le corps pour passer sur des obstacles ou des plans inclinés en gardant l'orientation optimale durant la navigation.
- La géométrie initiale du polygone de support. Le fonctionnement des déplacements des polygones passe par des multiples d'écartements incrémentaux par rapport à la posi-

tion initiale du polygone de support. La différence par rapport au temps d'horloge centrale d'une phase donne son écart avec cette position initiale. En utilisant cette méthode, les polygones auront toujours tendance à se recentrer par rapport à cette position et donc d'éviter que les incréments sortent les extrémités des pattes de l'espace de travail.

- La patte initiatrice est la patte qui permet l'initialisation des horloges de toutes les pattes. Sa valeur d'horloge et les lois de Wilson donnent la temporisation pour toute la couche de coordination.
- La vitesse des transitions est le paramètre qui gère la rapidité avec laquelle les nouvelles consignes venant de la couche décisionnelle seront appliquées. Comme la fluidité des mouvements est un élément essentiel dans la marche désirant être produite par Hexapode, il importe que ce paramètre existe. Le calcul de la nouvelle position qui devrait être appliquée est réalisé. La différence de cette position par rapport à la position actuelle est répartie en un nombre d'incrémentes spécifié pour être appliqué comme correction progressive pour que la trajectoire concorde avec celle nouvellement commandée. Bien entendu, la modification de la trajectoire ne peut déformer le polygone de support. Les modifications sont donc appliquées dans le polygone de retour. Dans le cas d'un évènement requérant une intervention immédiate, la correction peut s'effectuer en une seule itération dans le polygone de retour. Cette opération peut se faire à la limite en cessant le déplacement du polygone de support et en générant le polygone de remplacement à l'aide d'itérations dans le polygone de retour jusqu'à ce que la nouvelle configuration soit atteinte ou que les conditions justifiant une telle procédure soient éliminées.

Tous ces paramètres sont les entrées venant de la couche décisionnelle et ont une grande influence sur les résultats de la navigation. Ils sont modifiés en fonction de la représentation de l'environnement que se fait Hexapode ou que se fait le pilote qui remplace la couche décisionnelle. Les sorties de cette couche sont les positions effectives des extrémités des six pattes ainsi qu'un vecteur contenant l'information sur l'appartenance de chacune des pattes au type de polygone dont elle fait partie.

4.5.5.4 Trajectoire des pattes

La trajectoire que subit une patte dans un mode de fonctionnement standard est composée de quatre étapes, formant un trapèze. Ceci est représenté à la figure 4.20.

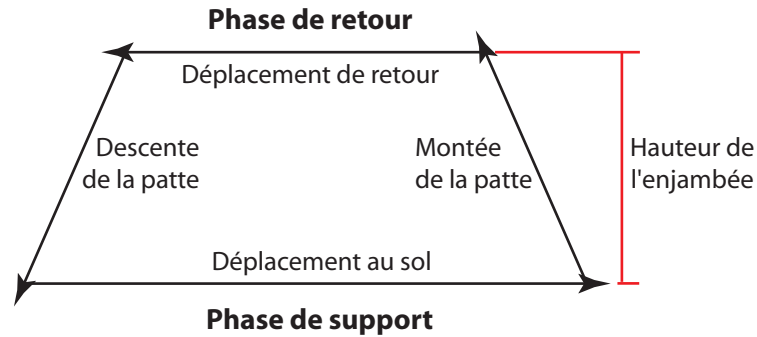


FIG. 4.20 – Trajectoire de l'extrémité d'une patte.

1. La partie de la trajectoire du bas est celle de la phase de support. C'est elle qui permet de déplacer le polygone de support. L'orientation de ce déplacement est paramétrisable en jouant avec l'orientation du polygone de support par rapport au repère du corps. La position des points des extrémités qui composent la trajectoire est fonction du nombre, de la grandeur et de l'orientation des incréments de déplacement qui sont appliqués sur le polygone de support.
2. Lorsque la patte passe dans la phase de retour, elle se relève dans la direction de l'axe Z et retourne vers la pose du polygone initial pour les autres axes. La trajectoire de relevé prend un quart de la durée de la période de retour. À la fin de cette trajectoire, la patte est en position relevée en s'étant légèrement déplacée horizontalement.
3. La patte poursuit son retour en effectuant une trajectoire horizontale dans la position relevée. Ceci est utile pour passer par-dessus les obstacles potentiels.
4. L'extrémité de la patte redescend pour atteindre le prochain point de contact. Ce point de contact doit être stable, sans quoi le polygone de support ne pourra soutenir le robot.

Une fois que le contact à l'extrémité de la patte est établi, la patte a complété le cycle et le processus recommence à la phase de support. Ceci est le cycle standard qui est appliqué aux pattes. Il permet des poses stables si la phase de support est bien gérée par la couche de coordination. Il est bien entendu possible de modifier la trajectoire du déplacement au sol pour toutes les pattes du polygone de support dans le cas où un déplacement spécial du corps serait nécessaire. Ces trajectoires sont fonctionnelles dans la plupart des cas et leurs utilisations sont conformes à la théorie des polygones.

4.5.6 Couche décisionnelle

Les explications données pour cette couche reflètent sur ce qui a été conceptualisé, mais qui n'a pas encore été implanté sur le robot expérimental. Le fait de présenter ces concepts donne un guide pour une marche à suivre suggérée pour la poursuite des travaux sur Hexapode. Du temps et de l'énergie ont été investis afin d'élaborer les grandes lignes de ce que pourrait être la couche décisionnelle, c'est pourquoi leur explication est justifiée.

Les entrées de cette couche sont de hauts niveaux. Souvent abstraites, elles sont les objectifs globaux de la mission. Les tâches résultant d'objectifs aussi généraux que de se promener et de trouver de l'eau sur Mars, réparer cette structure endommagée ou s'assurer de la sécurité d'un bâtiment en faisant une surveillance visuelle pièce par pièce d'un édifice par exemple seront certainement le quotidien des robots de demain. Les robots deviendront des créatures artificielles avec lesquelles humains pourront collaborer pour faire des activités complexes ou tout simplement effectuer des tâches qu'eux seuls ont la capacité de réaliser. Cependant, pour que tout cela soit possible, il faudra qu'ils soient capables de décortiquer ces objectifs en une série de sous-tâches. C'est sur la couche décisionnelle que repose la responsabilité de prévoir les meilleures actions à accomplir afin d'atteindre ces objectifs externes. De façon autonome par le robot, ces objectifs doivent être transformés en une série d'actions d'où émergent des trajectoires théoriques.

Dans le cadre du projet Hexapode, les objectifs globaux pourraient être des endroits à atteindre ou des cibles à visiter. Une des missions que pourrait accomplir Hexapode pourrait être, par exemple, d'aller chercher un objet pourvu d'un marqueur et de le ramener à un endroit précis. La trajectoire théorique nécessaire à la réussite de cet objectif serait planifiée par le robot lui-même, par exemple en reliant son point actuel au point objectif par une droite ou une courbe dont les paramètres peuvent être déterminés à l'avance. Le retour se ferait en référant aux difficultés rencontrées qu'il aurait mémorisées.

Les conditions de la surface sur laquelle devra se produire la navigation sont le deuxième groupe d'entrées qui auront une influence sur les sorties de la couche décisionnelle. En effet, les obstacles devront être soit contournés, soit franchis. Hexapode doit se faire une représentation de son environnement afin qu'il soit en mesure de

prendre les décisions éclairées sur ce qu'il faut faire pour atteindre ses objectifs. Cette représentation de l'environnement est possible grâce à des capteurs qui lui servent de sens, il serait possible de penser à la présence de capteur de proximités ultrasoniques, de capteurs tactiles ou même d'une caméra permettant à hexapode de voir le monde. L'acquisition de ces capteurs doit être faite par le robot et entraîne des charges de calculs de tailles variables selon le type d'entrées. Par exemple, un système de positionnement à ultrason fournit des signaux qui sont plus complexes à interpréter qu'un capteur de contact, mais amène des informations de nature différentes sur l'environnement. Un des systèmes de capteurs les plus utiles est une paire de caméras puisqu'elles permettent une perception spatiale en trois dimensions.

Cependant, le traitement de l'information venant d'un tel système est hors de portée pour le microcontrôleur utilisé actuellement. Il serait possible d'ajouter une deuxième unité de calcul, venant gérer les paramètres de haut niveau (cerveau) alors que le microcontrôleur actuel servirait à remplir les tâches de moyen et de bas niveau (cervelet et système nerveux). Les résultats des traitements de l'information des capteurs alimenteraient les processus menant à la représentation de l'environnement. Ces processus seraient en mesure d'ajuster les paramètres de la marche, comme la durée de la phase de retour et du cycle complet, la géométrie du polygone de support et la patte qui initialisera les nouvelles horloges pour toutes les autres. Ces paramètres sont les entrées de la couche de coordination. La représentation de l'environnement sert aussi à déterminer la présence d'obstacles, qu'il faudra contourner ou affronter. Ces obstacles amènent des perturbations de la trajectoire, il en émerge une trajectoire modifiée et adaptée par rapport à la trajectoire théorique calculée plus tôt. Cette trajectoire est aussi une entrée pour la deuxième couche qui saura en prendre compte.

Pour que l'intelligence du robot puisse apprendre et se développer, des processus de renforcement seront appliqués lorsque les actions accomplies auront porté fruit. Par exemple, le robot connaissant ses objectifs et ses cibles se verrait assigner des points selon son degré de performance dans la recherche de cette tâche. Des techniques d'algorithmie permettant l'apprentissage et la prise de décision comme les réseaux de neurones et la logique floue seraient propices à être utilisées.

4.5.6.1 Limitations

La programmation du microcontrôleur d'hexapode a été réalisée en gardant en tête d'améliorer les performances et de repousser les limites du microcontrôleur. Toutes les informations utiles à conserver et qui définissent les paramètres d'Hexapode sont enregistrées dans une structure. La structure en langage C est le format le plus polyvalent. La structure est définie une seule fois et des pointeurs passent à la fonction des adresses-mémoires correspondant aux sections utiles de la structure. En langage C, il y a les variables locales aux fonctions, les variables globales qui sont effectives partout dans le code. Lors de l'ouverture d'une fonction, les variables locales sont créées, ce qui demande d'allouer de la mémoire et d'écrire la valeur de la variable dans cet espace. Ceci demande du temps et entraîne une surcharge inutile de calculs. La technique du pointeur est fort utile puisqu'elle évite cette opération. De plus, elle présente l'avantage de pouvoir pointer sur des sous-branches de la structure comme argument des fonctions. Cela permet de générer des fonctions plus polyvalentes qui peuvent être utilisées sur toutes les branches plutôt que d'avoir à réécrire une fonction spécifique pour chaque branche. Le programme en langage C utilisé pour générer cette structure se trouve à l'annexe C.2 on y montre les informations qui y sont enregistrées.

Bien que les possibilités du microcontrôleur soient étendues, il a ses limites. Le plus déterminant est la taille de la pile de mémoire qui peut être utilisée pour la gestion des processus d'appel de fonction. En effet, tout comme la structure qui contient les données, les processus sont constitués de fonctions générales qui appellent des sous-fonctions. Souvent, afin que les paramètres de tout le robot soient modifiés, une fonction est appelée pour chacune des pattes. Cette façon de faire est efficace du point de vue algorithmique, mais oblige des sauvegardes des états internes du microcontrôleur.

Également, puisque la programmation du microcontrôleur s'effectue en langage C plutôt qu'en C++, il est impossible d'utiliser les objets ou d'autres outils algorithmiques de haut niveau. Le langage C est idéal pour la programmation simple et efficace, mais est tout de même plus limité que les langages de programmation de plus haut niveau. Tout fonctionne très bien pour l'implantation des couches d'interface matérielle et de coordination. Cependant, cela se complique lorsque la couche décisionnelle inclut des algorithmes nécessitant de l'algorithmie plus complexe et plus de puissance de calcul.

4.5.6.2 Remplacement de la couche décisionnelle.

La solution pour pallier à cette carence est d'abaisser le niveau d'intelligence directement embarqué sur la plate-forme et de le remplacer par une interface se trouvant à l'extérieur. Spécifiquement, ce qui a été implanté à Hexapode est de construire une console de téléopération permettant à un utilisateur de donner en temps réel les paramètres de la marche dont a besoin la couche de coordination. Pour les premières promenades d'Hexapode, quatre paramètres ont été spécifiquement contrôlables.

- L'avance des déplacements sur le polygone de retour de chacun des incréments de l'algorithme. Ce paramètre varie entre 0 et 2 mm par itération.
- La direction de l'avance par rapport à l'axe X du repère du robot entre -180 degrés et 180 degrés (figure 3.2).
- La rotation que doit effectuer Hexapode dans une série d'incrément. Le débattement de ce paramètre est fixé à plus ou moins 30 degrés. Ceci permet d'effectuer une demi rotation en 6 périodes de retour, soit en un minimum de 12 secondes.
- Le ratio des phases, soit la durée du cycle complet par rapport à la durée de la période de retour.

En utilisant ces paramètres, il est possible d'avoir un contrôle très efficace en temps réel des déplacements. Certains paramètres ont été ajustés à l'avance par rapport à ceux qui sont contrôlés, d'autres découlent de l'optimisation de leur valeur par rapport aux paramètres donnés en consigne.

- La durée de la période de retour est définie pour optimiser la vitesse des servomoteurs. Comme discuté plus tôt (section 4.2.2), une vitesse trop lente entraîne des arrêts et des démarrages dans les actionneurs, alors qu'une vitesse trop rapide ne permettrait pas de réaliser la consigne. Il y a une itération à tout les 50Hz. La durée de la période de retour optimale est de 100 itérations. La vitesse maximale pouvant être atteinte est donc de $2mm/iteration \times 100iteration \times 50Hz = 100mm/sec$. Ceci est jugé suffisant, les objectifs du projet n'étaient pas de construire un robot rapide mais d'effectuer des démarches stables. Pour obtenir une marche plus stable ce paramètre est limité à 1mm.
- La patte initialisatrice des séquences est la patte avant gauche. Pour que le robot avance dans des conditions où il ne rencontre pas d'obstacle particulier, le choix de la patte initialisatrice n'est pas critique pour les déplacements. Dans le cas où

- la perte d'un contact à l'une des extrémités des pattes du polygone de support est rencontrée, ce paramètre doit être modifié à l'intérieur de l'algorithme.
- Le polygone de support initial est celui qui est adopté dans la pose orthogonale standard.
 - L'orientation du polygone par rapport au corps le place parallèlement au plan formé des axes X-Y du repère du robot.
 - Il est possible de faire varier la vitesse d'avance du robot ou de modifier le nombre de pattes au sol avec le paramètre de phase. La stabilité du robot est fonction de la vitesse de la démarche. En utilisant une phase de deux on obtient la démarche tripode et en utilisant une phase de six on obtient la démarche à vague lente tel que discuter à la section 4.5.5.2.
 - La hauteur des enjambées est fixée à l'avance pour permettre de passer par-dessus la plupart des obstacles. L'ajout de ce paramètre dans la console aurait pu se faire, mais son utilité n'a pas été jugée critique pour l'instant. L'enjambé utilisé est de 40mm ce qui permet de passer par dessus les obstacles de 25mm utilisés lors des expérimentations.

Il a été envisagé initialement que la console soit une boîte externe composée de potentiomètres. Chacun des paramètres de la marche aurait été interfacé par l'acquisition des signaux analogiques des potentiomètres montés comme diviseurs de tension. Cependant, les possibilités qu'offre un ordinateur externe ont fait que cette console a été réalisée à l'aide du générateur d'interface utilisateur (GUI) et du module Simulink du logiciel Matlab dans sa version R14. Quatre barres de défilement permettent de faire le choix des paramètres de la marche. Il y a aussi une fenêtre numérique qui affiche simplement la valeur qui est appliquée ou qui peut servir à la saisie de cette valeur par le clavier numérique. Enfin, un bouton de remise à zéro de chacun des paramètres est ajouté.

Simulink est un environnement de programmation graphique qui permet à plusieurs processus de s'exécuter dans un cadre se rapprochant du parallélisme. Ces processus se voient gérés par le logiciel d'une manière à ce que l'exécution soit séquentielle, mais en alternant plusieurs chemins en même temps pour l'exécution. La temporisation des ordinateurs ayant Windows ou Linux comme système d'exploitation est incertaine puisqu'il n'est pas permis à l'utilisateur de gérer lui-même ce qui est envoyé au processeur. Il est donc impossible de fonctionner en temps réel de façon précise. Le résultat est qu'avec le bloc «Real Time» il est plus probable d'obtenir des sorties au bon moment temporelle-

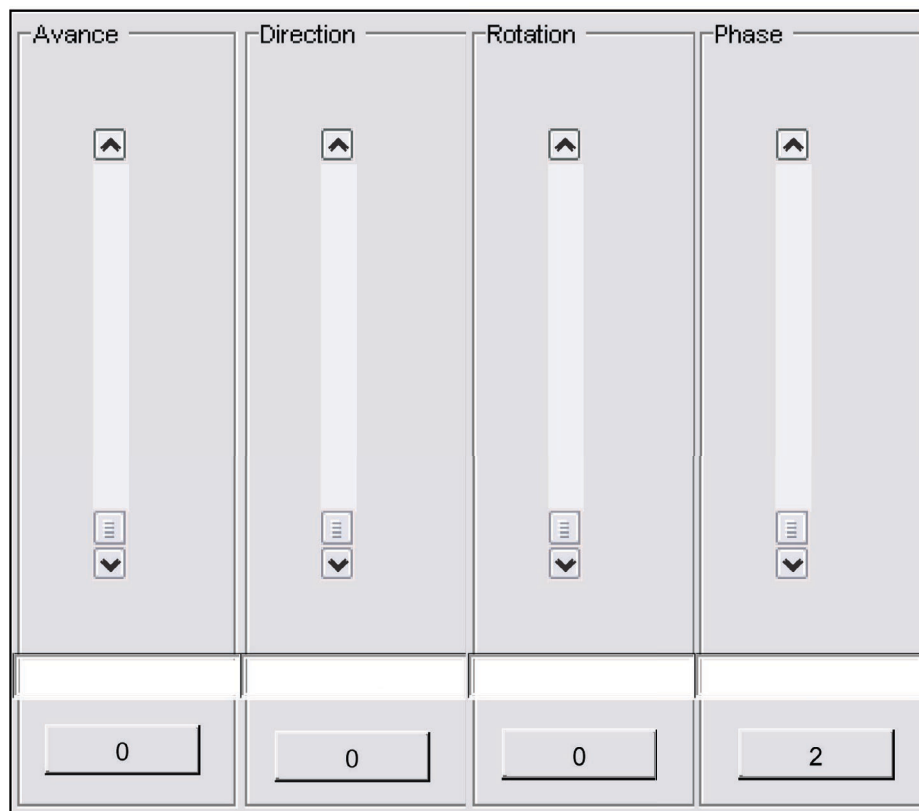


FIG. 4.21 – Console de téléopération.

ment puisque la priorité des processus liés à Matlab se voit augmentée. Dans le cas du microcontrôleur, la temporisation est en temps réel puisque le programme choisi quelle action se produit à différents moments grâce à la génération d'interruptions.

L'interface de la console est minimaliste pour l'instant, cependant elle pourra être adaptée facilement au besoin. La figure 4.21 montre le résultat de cette console de téléopération.

À court terme, d'autres boutons seront ajoutés pour gérer l'orientation du polygone de support, le nombre d'incrément de transition pour que la nouvelle consigne soit en vigueur, ainsi que la hauteur des enjambées de chaque pas. La boîte d'outil «Virtual reality» de Matlab permettrait aussi de se servir des entrées d'une manette de jeu pour contrôler Hexapode. Grâce au lien avec l'ordinateur, la poursuite des travaux sur Hexapode pourra se servir de la puissance de calcul d'un ordinateur pour effectuer les opérations qui demandent trop de temps au processeur pour être effectuées directement sur le microcontrôleur.

Pour que le robot puisse communiquer avec l'extérieur, la façon la plus efficace de faire est de se servir de la communication sérielle. Le protocole RS-232 est utilisé. Il est alors possible d'échanger des informations avec les périphériques externes. Des routines permettant la communication sérielle dans Simulink ont été trouvées sur les forums de discussion du produit. Leur utilisation est simple et efficace et a permis la communication avec le robot. Ce programme envoie une nouvelle commande uniquement lorsque les paramètres de la marche sont changés, ceci a pour but de ne pas surcharger inutilement le microcontrôleur et de permettre à ce dernier de fonctionner avec le maximum d'autonomie.

Avec cette console, l'opérateur qui choisit les déplacements qu'effectuera Hexapode remplace l'intelligence artificielle de haut niveau. L'utilisateur doit se faire une représentation de l'environnement et penser à la meilleure trajectoire pour atteindre les objectifs de mission. La personne qui gère les paramètres de la marche doit aussi se soucier des réactions aux stimuli extérieurs afin de permettre l'adaptation à l'environnement. La console pourrait devenir bidirectionnelle et retourner les valeurs des capteurs présents sur le robot afin que soit optimisée la trajectoire. La représentation de l'environnement est l'élément qui permettra au robot, par l'intermédiaire de son opérateur, de s'adapter aux nouvelles conditions. Cette trajectoire est ensuite transformée pour tenir compte de la représentation de l'environnement que se fait le robot à l'aide de ses capteurs. Les obstacles sur le terrain impliquent des perturbations de la stabilité qui devront être compensées par le robot pour remplir sa mission avec brio.

Chapitre 5

Conclusion

Cette dernière partie du mémoire fait un retour sur les notions importantes perçues tout au long du projet et qui ont été présentées dans le mémoire. Des pistes pour des applications où le robot marcheur Hexapode pourrait être utilisé sont présentées. Finalement, une ouverture est dégagée dans le but de montrer la vision qu'a voulu laisser le chercheur sur l'avenir de la robotique mobile biomimétique.

5.1 Retour sur les éléments essentiels du mémoire

L'objectif principal du projet Hexapode était d'explorer le domaine de la robotique mobile des marcheurs. Cet objectif a été atteint au long du processus qui a

mené au développement d'un robot à six pattes qui utilise la marche comme moyen de déplacement. Le prototype qui a été développé est pleinement fonctionnel et a permis de vérifier et de valider les concepts théoriques et expérimentaux qu'implique la génération de démarches stables et efficaces. La validation de l'atteinte des objectifs

Ce mémoire a présenté les étapes qui ont été déterminantes pour parvenir au prototype Hexapode. Tout d'abord, les objectifs particuliers du projet ont été mis en évidence (chapitre). Les concepts et notions liés au projet ont été développés en détail pour permettre de mieux comprendre ce qu'implique un tel projet (chapitre 1). Notamment, le concept de biomimétisme consiste à s'inspirer des solutions qu'a utilisées la nature au fil de son évolution pour résoudre les problèmes (section 1.1).

Les différentes morphologies permettant la marche ont été étudiées (section 1.2.1). Il en est ressorti qu'il est possible de comparer le corps et les membres des marcheurs à des mécanismes à topologie variable (section 1.2.2). Ceci permet alors l'étude du problème naturel de la marche d'un point de vue robotique. Les notions liées à la modélisation des points de contact à l'extrémité des pattes (section 1.2.3) ont permis de bâtir des modèles qui permettront d'utiliser la définition des polygones de contact et de support. Ces polygones sont les notions clefs utiles pour le regroupement des pattes en vue de produire des séquences menant au déplacement du corps du robot (section 1.2.4). L'utilisation judicieuse des polygones entraîne la stabilité des déplacements ce qui est important pour l'atteinte des objectifs de la marche (section 1.2.5).

La notion des polygones est un élément nouveau dans le domaine des robots marcheurs, aucun concept semblable ne se retrouvait à la connaissance de l'auteur dans la littérature. La marche se présente sous diverses formes, il est utile de les classer en deux types : la marche dynamique et la marche quasi statique (section 1.2.6). Le type de marche qui est utilisé pour le projet Hexapode est la marche quasi statique, ce type de marche a la particularité de permettre une décomposition de la marche en une série de poses où chacune d'entre elles est stable et donc en équilibre statique.

Avant de débiter la construction des prototypes du robot marcheur, l'étude de diverses morphologies d'animaux a été réalisée (chapitre 2). L'approche biomimétique a guidé le développement du robot, l'étude des marcheurs vivants a donc été faite pour permettre d'approfondir le fonctionnement des membres des marcheurs et pour permettre de voir les possibilités propres aux types de pattes en jeu. Les insectes sont

ressortis comme les grands champions tant pour leurs avantages morphologiques que pour leur simplicité à être reproduits par la robotique. Ils sont particulièrement bien adaptés à leur environnement, leur mobilité et leur autonomie sont également prodigieuses (section 2.1.2.2). Hexapode n'est pas le seul robot qui s'inspire des insectes pour leur design, la littérature en présente d'autres (section 2.2). Cependant, Hexapode présente des caractéristiques distinctives. Les diverses recherches dans le domaine ont servi à concevoir des améliorations et à faire avancer le domaine par le présent projet.

Les critères d'évaluations qui ont été utilisés pour vérifier la performance des modèles sont d'une importance primordiale (section 3.1). Tout le développement menant à Hexapode s'est fait en se servant de ces critères comme paramètres à optimiser. Ces critères d'évaluation de la performance sont :

- la mobilité,
- l'autonomie,
- l'adaptabilité.

Les prémisses de la conception d'Hexapode ont été faites par rapport à la morphologie des fourmis (section 3.2). C'est-à-dire l'architecture des membres et la mise à l'échelle de leurs dimensions. Le premier modèle qui a émergé de cette morphologie a servi uniquement à donner une idée des déplacements pouvant être créés avec cette architecture (section 3.3). Ce modèle a été assemblé à l'aide d'un module du logiciel Pro-Engineer pour être manipulé comme un pantin et ainsi donner l'approche intuitive nécessaire aux mouvements. Un prototype de design de robot permettant la réduction du nombre d'actionneurs a été réalisé (section 3.4).

Un premier modèle de patte a été analysé, il comportait seulement deux actionneurs. L'extrémité de la patte était positionnée dans un plan parallèle à la direction d'avance privilégiée du robot. Comme deux actionneurs étaient utilisés, il était possible de contrôler deux degrés de liberté en translation à l'extrémité de la patte. L'étude du problème géométrique direct du mécanisme a montré qu'il n'était pas viable de le construire.

L'architecture qui a été finalement retenue pour le robot Hexapode est une version simplifiée, à trois degrés de liberté, de la fourmi. Une patte de validation utilisant cette architecture, les mêmes actionneurs et la même unité de calcul a été mise de l'avant

(section 4.1.2). Cette patte a servi de banc d'essai pour effectuer les tests nécessaires menant aux corrections requises pour la création du prototype final d'Hexapode.

L'ensemble du développement et de la mise au point d'Hexapode s'est fait en tenant compte des trois critères de performance. Ces critères étaient la base qui justifiait l'amélioration des modèles en vue du prototype qui a été construit. La mobilité d'Hexapode (section 4.2) passe par son architecture à trois degrés de liberté par patte. L'algorithme permettant la résolution du problème géométrique inverse a dû être élaboré pour permettre sa résolution en temps réel. Le contrôle des actionneurs qui permet l'orientation des membres du robot est aussi rattaché à ce critère. Grâce aux servomoteurs RC, il est possible pour Hexapode d'effectuer les diverses poses qui découlent des démarches quasi statiques (section 4.2.2). Le design mécanique et la fabrication d'Hexapode ont également été des étapes critiques du projet (section 4.3.2), puisque les dimensions et la masse des membres ont des répercussions importantes sur tous les aspects de la marche. Le corps d'Hexapode est fabriqué en CPVC et les pattes en ABS selon la technologie FDM.

L'amélioration des performances par rapport au critère d'autonomie est passé par l'implantation de l'équilibrage statique. Ceci a permis d'obtenir une minimisation du couple que doivent fournir les actionneurs (section 4.4). La vérification de l'impact des variations de l'architecture et de l'orientation du premier axe de la patte a été effectuée. L'étude du chargement des pattes du polygone de support a été utile pour parvenir à l'équilibrage statique partiel à l'aide de ressorts en torsion (section 4.4.4.1). L'utilisation de l'équilibrage statique pour un robot marcheur est un des éléments nouveaux du projet par rapport à ce qui avait été trouvé dans la littérature. L'étude du chargement a également été utile pour déterminer les dimensions optimales des membres et de la géométrie du corps d'Hexapode.

L'optimisation de la conception en fonction du critère d'adaptabilité a aussi été réalisée. Pour ce faire, des capteurs servent de sens au robot (section 4.5.1). Un micro-contrôleur lui sert de cerveau. Grâce à cette unité de calcul qui est embarquée directement sur son dos, Hexapode est capable de faire la gestion des processus nécessaires à sa navigation de façon informatique (section 4.5.2). Ceci a été fait en passant par l'approche de la programmation hiérarchique (section 4.5.3). Trois couches algorithmiques distinctes sont mises en valeur et viennent fournir les outils nécessaires afin de permettre au robot de se déplacer : la couche d'interface matérielle, la couche de

coordination et la couche décisionnelle.

La couche d'interface matérielle est celle qui est à la base de tous les processus qui relient les composantes physiques d'Hexapode en passant par la programmation du microcontrôleur (section 4.5.4). Cette couche permet de gérer les signaux qui sont envoyés aux servomoteurs. Elle fait également l'acquisition des divers capteurs qui sont ou seront installés sur le robot.

La couche de coordination s'occupe de gérer les séquences qui mènent aux démarches capables de produire des déplacements stables et efficaces (section 4.5.5). Cette couche passe par l'implantation des lois de Wilson (section 4.5.5.2). Grâce à des règles simples, il est possible de regrouper les pattes pour former les polygones de support et de retour et de générer les trajectoires des extrémités des pattes qui sont nécessaires pour effectuer des déplacements.

Les paramètres de la marche qui permettent la navigation par la couche de coordination, sont fournis par la console de téléopération qui sert de couche décisionnelle à Hexapode (section 4.5.6). Par cette console, il est possible de diriger le robot à une certaine vitesse, d'effectuer des rotations et de faire varier le ratio de la durée du cycle total par rapport à la durée de la phase de retour. L'utilisateur peut donc explorer un terrain en fournissant un minimum d'informations au robot. L'utilisateur peut donc planifier des trajectoires selon sa représentation du terrain. Il devient alors possible au robot marcheur Hexapode de remplir des missions complexes.

5.2 Application

Hexapode se veut un prototype de créature artificielle : une fourmi géante. Les possibilités émergentes d'un mode de transport utilisant la marche sont multiples. Optimalement, ce type de robot pourra s'acquitter de tâches comme faire l'exploration de toutes surfaces ou naviguer dans un environnement encombré.

Hexapode a été conçu plus spécifiquement pour franchir les restrictions qui limitent d'autres robots ou encore pour des applications où la mobilité, l'autonomie et l'adaptabilité sont primordiales. Associée au biomimétisme, la mise en valeur des critères de

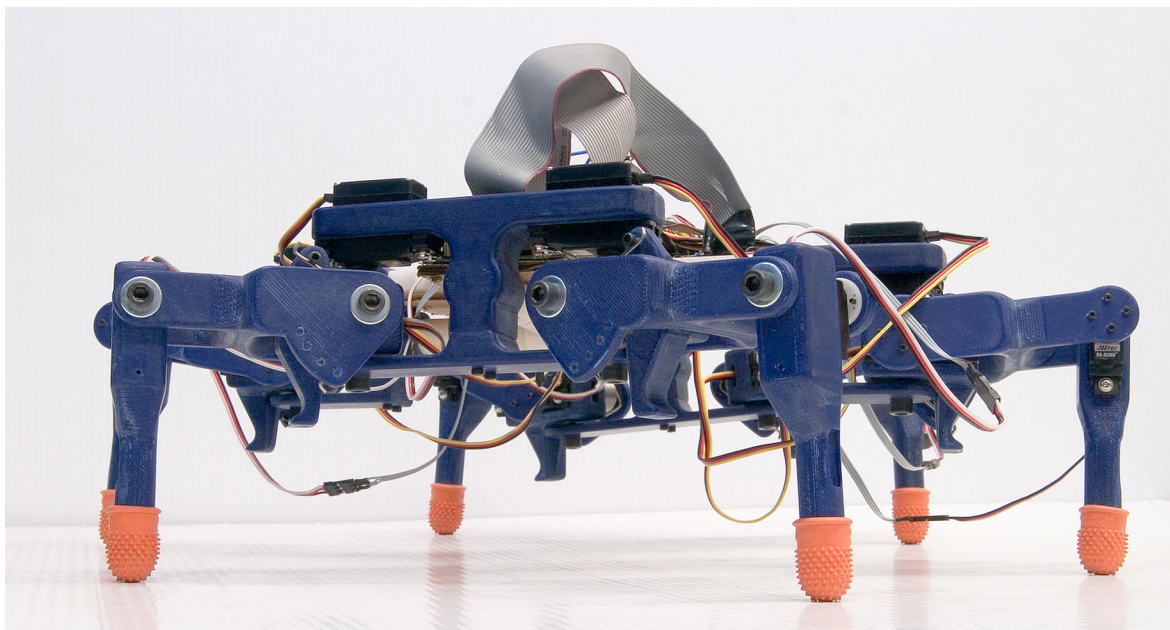


FIG. 5.1 – Hexapode, version finale.

performance que nous avons initialement retenue, a permis de générer une solution viable aux problèmes de déplacements sur des surfaces dont le relief n'est pas connu à l'avance.

5.2.1 Domaine spatial

Une des voies d'utilisation qui pourrait être visée, est l'exploration de terrains de toutes sortes, terrestres ou non. Le robot Hexapode pourrait être un premier pas vers le développement de modules utilisés dans un cadre spatial. Les futures versions d'Hexapode pourraient devenir les spécialistes de l'exploration de surfaces de toute sorte.

L'environnement de Mars, de la Lune ou d'autres planètes seraient tout à fait le type de mission dont pourrait se charger Hexapode. Dans une optique d'exploration planétaire, il serait possible d'introduire la notion de coopération entre plusieurs robots similaires pour effectuer des tâches complexes. Comme chez les fourmis, les robots pourraient même être de physionomies quelque peu différentes, certains individus se spécialisant dans des tâches particulières afin de donner au groupe des possibilités encore plus grandes pour remplir leurs objectifs. Il pourrait y avoir des robots plus agiles pour agir comme éclaireurs, d'autres plus forts pour transporter les instruments

ou ramener des échantillons. Des groupes de robots marcheurs pourraient aussi être envoyés pour déployer des bases planétaires en attendant les astronautes.

Il serait aussi possible d'imaginer des escouades de robots-araignées voyageant autour d'une station spatiale ou d'un satellite à l'aide de bout de patte à électro-aimant. Ces robots pourraient servir au système d'entretien de telles installations. Leur utilisation serait moins coûteuse que de faire effectuer les tâches par des humains. L'architecture propre aux robots à pattes leur permettrait de se promener à des endroits exigus et où le bras canadien n'a pas accès. Encore une fois dans ce cas, les robots pourraient travailler en équipe pour effectuer des tâches plus exigeantes, par exemple pour transporter le matériel pour les réparations.

Comme des applications spatiales sont imaginées, les aspects liés à la réduction de la consommation énergétique sont très importants. Le robot doit donc pouvoir fonctionner avec des ressources énergétiques réduites au maximum. Pour que le coût du lancement soit minimisé, la masse et les dimensions du robot doivent être minimisées. Les systèmes mécaniques, électroniques et informatiques embarqués sur le robot doivent avoir une fiabilité à toute épreuve. Dans le cas de problèmes, des procédures de sauvetage doivent pouvoir être lancées à partir de la Terre. Comme dans le cas du robot martien Spirit qui s'est fait reformater sa mémoire flash à des millions de kilomètres. Il est évident que le prototype actuel d'Hexapode n'est pas celui qui serait envoyé sur la Lune ou sur Mars, mais peut-être qu'un de ses descendants le sera.

5.2.2 Domaine terrestre

Les robots marcheurs sont notamment conçus pour de se déplacer dans des environnements difficiles. Certains espaces terrestres entrent aussi dans cette catégorie. Hexapode pourrait faire la collecte d'informations biologiques à des endroits trop à risque pour les scientifiques. Il serait possible de l'envoyer dans des lieux ayant subi une contamination biologique, chimique ou même nucléaire. Il pourrait être utilisé de façon autonome ou par téléopération, ce qui vient décupler les champs d'application possible.

Explorateur né, il pourrait patrouiller des régions pour identifier les transformations du milieu, en envoyant des données avec une fréquence plus grande que celle venant

d'équipes de chercheurs qui ne peuvent être partout à la fois. Hexapode pourrait aussi être utilisé pour faire la fouille du site d'une catastrophe, son faible poids lui donnerait l'avantage de diminuer les risques d'effondrement des structures et conséquemment les risques pour la vie des sauveteurs : il augmenterait alors aussi les chances de retrouver des survivants.

Dans le cas où la mission du robot serait de tracer une carte du terrain, il sera possible d'y parvenir en lui couplant un émetteur GPS et se servant des coordonnées des polygones de contact juxtaposées selon les informations données par l'inclinomètre sur l'orientation du corps du robot. Des zones d'intérêt pourraient être identifiées à l'aide des capteurs de proximité.

Des applications urbaines sont aussi possibles. Les descendants d'Hexapode pourraient aussi être utilisés comme robot domestique à la maison. Des robots de ce type commencent à faire leur apparition. D'abord comme jouet, puis en l'intégrant à la domotique, il deviendrait l'interface mobile du système. Il pourrait veiller à la sécurité des malades ou des personnes âgées ou même faire des patrouilles de garde dans les musées ou dans les endroits requérant de la surveillance. Puisqu'il pourrait être amené à travailler avec le public, l'apparence du robot doit être soignée pour le rendre visuellement attrayant.

Les présentes recherches ont été faites en vue de développer une plate-forme de robot mobile marcheur qui sera plus tard modifiée et adaptée plus spécifiquement à la tâche qui lui sera attribuée. Bien entendu, beaucoup de travail subsiste avant qu'Hexapode fasse des expéditions dans l'espace ou que des millions d'exemplaires envahissent les tablettes des magasins. L'ampleur du travail n'est pas un obstacle, mais doit être vu comme un défi à relever.

5.3 La fin ou le début des recherches ?

Qui parle de construire des créatures artificielles doit s'assurer qu'un maximum d'autonomie, de mobilité et d'adaptabilité puisse leur être assignées. Bientôt sera révolu le temps où les robots ne sont que de simples outils programmables. Les robots doivent pouvoir sortir de l'environnement clos des usines où ils ne sont pas utilisés à leur plein

potentiel. Comme discuté dans la section des applications envisageable pour Hexapode, les robots domestiques seront souhaités dans les résidences dans un avenir plus ou moins lointain. Actuellement, on peut penser au robot bipède marcheur «Robosapien» ou au chien artificiel «Aibo» de Sony qui émule un vrai chien et qui peut même apprendre de nouveaux comportements.

Lors de compétitions de Robocup des équipes d'étudiants à travers le monde mettent au point des équipes de robot qui jouent au soccer sans qu'il n'y ait d'intervention humaine. L'intelligence artificielle des robots est en train d'émerger. Le temps du cloisonnement des disciplines sera bientôt révolu.

Les exemples du passé montrent les bouleversements des systèmes de production où les systèmes mécaniques régnaient en maître en étant composés de mécanismes voués à une seule tâche. Ces chaînes de montage, souvent inutilement compliqués, se voyaient perdre leur avantage dans des productions d'un nombre plus restreint d'unités. Ces lignes de production ont souvent été remplacées par des systèmes plus flexibles. Par programmation, il est possible de changer les comportements de la mécanique. Un système de contrôle est souvent plus efficace et beaucoup plus adaptable que ce que peut faire la mécanique seule.

Il est alors possible d'extrapoler, à partir de cet exemple, pour y intégrer des notions de programmation plus avancées pour résoudre des tâches encore plus complexes. Il pourrait être avantageux dans beaucoup de contextes d'augmenter l'adaptabilité et l'autonomie des machines.

L'intelligence artificielle se trouve dans l'algorithmie, dans la capacité des machines à être dotées d'intelligence abstraite. La capacité de raisonnement des systèmes modifiera les limites de ce qu'il est présentement possible d'accomplir sur terre et dans l'espace. Pourquoi avoir à se déplacer sur de grandes distances alors qu'il est possible d'utiliser un système en téléopération. Un chirurgien est déjà capable d'opérer un patient à l'autre bout de la planète en passant par internet et ce n'est qu'un début. Si on poursuit la réflexion sur le concept. Un système intelligent pourrait le prémunir de faux mouvements et pourrait même éventuellement effectuer seul l'opération. La base de données provenant de tous les autres cas d'opération précédents serait probablement plus utile que la documentation qu'un chirurgien trouve dans les dessins de ses livres d'anatomie.

Ce qui limite l'émergence des robots dans toutes les sphères de la société, c'est que leur développement est sans cesse relégué au niveau d'outil perfectionné. Les mécaniciens font de bons corps, les électriciens font de bons systèmes de contrôle et les programmeurs tentent de développer des esprits artificiels. Il est cependant assez rare que tous ces aspects soient mis en commun pour former un être qui s'approche d'une créature artificielle. Les Japonais ont une sérieuse longueur d'avance dans ce domaine. Les golems, les tetsujins et les hommes mécaniques font depuis longtemps partie de leur culture. Des budgets sont alloués pour développer en ce sens. Même l'entreprise privée y trouve son compte et y voit un potentiel de développement à long terme, pensons à Asimo de Honda ou au Papero de Nec.

Si les Japonais voient en les robots une voie intéressante pour la recherche, dans mon cas, j'y vois un potentiel d'évolution. L'humain est mortel, physiquement faible et imparfait, il a utilisé son intelligence pour adapter son environnement à ses besoins. Aujourd'hui il y a une solution technologique à presque tous les maux. L'humain évolue lentement, dans le cas des robots, il y a le potentiel de corriger cette lacune. Le corps des robots est conçu de pièces qui peuvent être changées lors de bris. Il est donc possible de construire des systèmes qui pourraient se réparer et s'améliorer à l'infini. Le problème actuel de cette stratégie vient du fait que les technologies électriques et informatiques ne sont pas encore assez matures pour être efficaces sans tomber rapidement dans la désuétude. Des progrès incroyables ont été faits depuis les vingt dernières années, la poursuite de ces développements laisse envisager des solutions pour donner un système locomoteur, nerveux ainsi qu'une capacité de traitement assez fort pour mener à des concepts abstraits.

On peut faire l'analogie entre l'évolution des systèmes coopératifs et l'évolution en général. Lorsque les premiers humains se sont regroupés pour pratiquer la chasse, certains individus ont pu vaquer à d'autres occupations. Ceux qui ne chassaient pas, se concentraient sur ce qui a conduit à l'émergence de la société. Certains ont pris le temps de développer le langage oral, d'autres ont peint les murs des grottes de façon artistique, certains étaient affectés à la surveillance du feu pour ne pas qu'il s'éteigne. Le surplus de capacité de main-d'oeuvre a permis l'émergence de nouvelles possibilités. La même chose devrait se produire avec les capacités matérielles informatiques. Lorsque les processeurs seront assez puissants pour pouvoir résoudre plus de problèmes que seulement la résolution de calculs de positionnement ou le calcul de trajectoires simples, il y aura émergence d'entités artificielles qui auront d'autres propriétés que celles qui

sont actuellement données aux robots d'usine.

Hexapode est une créature artificielle, l'embryon potentiel d'une créature autonome encore plus complexe. Les objectifs personnels du chercheur le poussaient à créer un tout, où seraient mises en commun toutes les notions de design mécanique, de conception de circuit électronique et de programmation sur microcontrôleur. Hexapode est la somme d'éléments mécatroniques. Il est l'aboutissement des connaissances ayant été acquises jusqu'à présent dans ce domaine. Certaines parties du projet mériteraient d'être continuées, les concepts menants à leur implantation sont élaborés, il ne reste qu'à les intégrer au prototype.

À la lecture de ce mémoire, ceux qui voudront poursuivre le travail commencé ont des pistes de solutions assez précises pour le faire. Le développement d'une créature artificielle est une tâche qui ne peut être complètement terminée, tout comme la nature d'ailleurs qui est toujours en processus itératif d'amélioration avec les générations de vivants qui se succèdent depuis des milliards d'années. La vie artificielle n'est que le prolongement de cette évolution. Actuellement, les humains tentent de faire franchir à cette nouvelle forme de vie les premières marches de l'ascension, il est à prévoir que dans un avenir plus ou moins éloigné, ce seront les robots eux-mêmes qui feront le développement de leurs descendants.

Bibliographie

- [1] Wikipedia, The free encyclopedia
«<http://fr.wikipedia.org/wiki/Biomim%C3%A9tisme>», Janvier 2003.
- [2] Honda, «<http://asimo.honda.com>»
- [3] Sony, «<http://www.sony.net/SonyInfo/QRIO/>»
- [4] Gosselin, C., Notes pour le cours «Mécanique des manipulateur», GMC-64388, 2004.
- [5] Wisse, M., «Essentials of dynamic walking, Analysis and design of two-legged robots», 2004.
- [6] Collins, S. H., Ruina A., «A Bipedal Walking Robot, with Efficient and Human-Like Gait», ICRA 2005.
- [7] Wilson, D.M., «Insect walking», Annual review of entomology, volume 11, pages 103-122, 1966.
- [8] Pearson, K., «Characteristics of leg movements and patterns of coordination in locust walking on rough terrain», International Journal of Robotics Research, no.3, pages 101-112, 1976.
- [9] Cruse, H., «A new model describing the coordination pattern of the leg of a walking stick insect», Biological Cybernetics, no.32, pages 107-113, 1979.
- [10] Dean, J., Cruse, H., «Modelling the control of walking in insects», Biological Motion, Springer, 1990.

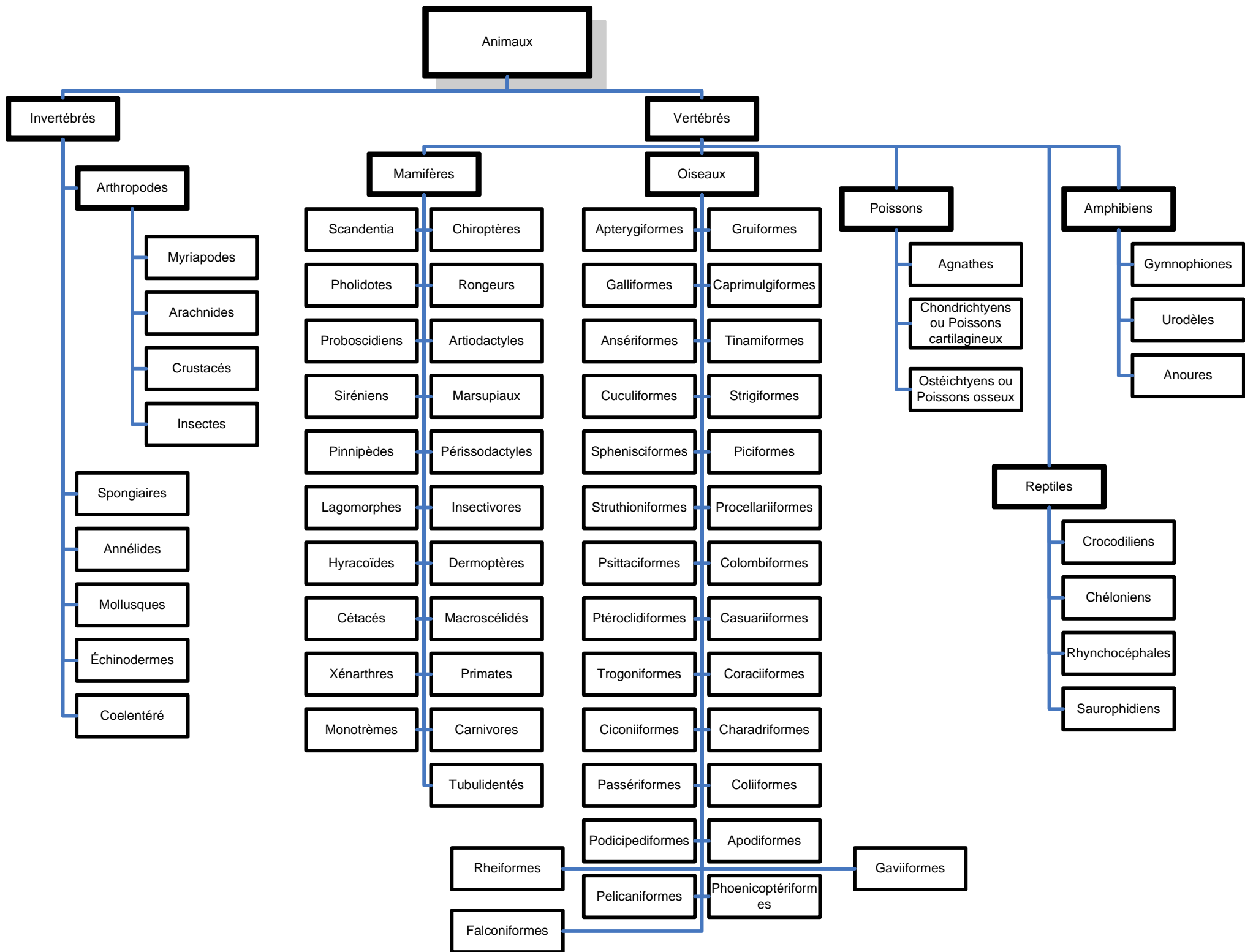
- [11] Brooks, R.A., «A Robot That Walks ; Emergent Behaviors from a Carefully Evolved Network», MIT AI Lab Memo 1091, Février 1989.
- [12] Angle, C., Brooks, R.A., «Small Planetary Rovers», Proceedings of IEEE International Conference on Intelligent Robots and Systems, 1990.
- [13] Angle, C., «Design of an Artificial Creature», Masters Thesis, MIT Department of Electrical Engineering and Computer Science, Juin 1991.
- [14] Ferrell, C., «Robust Agent Control of an Autonomous Robot with Many Sensors and Actuators», MIT Artificial Intelligence Lab Technical Report 1443, 1993.
- [15] Ferrell, C., «A Comparison of Three Insect-Inspired Locomotion Controllers, Robotics and Autonomous Systems», no.16, pages 2-4, 1995.
- [16] Kirchner, F., Spenneberg, D., «Omni-directional walking in multipod- robots based on feedback driven oscillators and local reflex mechanisms», Proceedings of the fourth International Conference on Climbing and Walking Robots, Berns, Karsten pages 643-650, 2001.
- [17] Spenneberg, D., Kirchner, F., De Gea, J., «Ambulating Robots for Exploration in Rough Terrain on Future Extraterrestrial Missions», 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation, ASTRA 2004.
- [18] Kennedy, B., Agazarian H., Cheng Y., «LEMUR : Legged Excursion Mechanical Utility Rover», Autonomous Robots, vol.11, pages 201-205, 2001.
- [19] Lewinger, W.A., «Insect-Inspired, Actively Compliant Robotic Hexapod», International Conference on Climbing and Walking Robots (CLAWAR), London, UK, pages 13-15, Septembre 2005.
- [20] Goulet, M., Gosselin, C.M., «Hexapode : un robot explorateur tout terrain», Comptes-Rendus CCToMM Symposium on Mechanisms, Machines, and Mechatronics, St-Hubert, Québec, 2005.
- [21] Goulet, M., Gosselin, C.M., «Hexapode : un robot marcheur», CSME Transactions, 2006.
- [22] Florent, Jacques, Garnier, Yves, Mathivet, Éric, «Encyclopédie Larousse de la nature, la foret et la faune», Édition Larousse, isbn : 2-03-152112-8, 370 pages, 1993.
- [23] Laliberté, T., Gosselin, C.M., Côté, G., «Rapid prototyping of mechanisms», IF-ToMM Tenth World Congress on the Theory of Machines and Mechanisms, pages 959-964, Juin 1999.

- [24] Bowling, Allan, «Mobility and dynamic performance of legged robots», IEEE international conference on robotics and automation, Barcelona, Spain, Avril 2005.
- [25] Freescale, «DSP56F801-7UM 16-Bit Digital Signal Processor Users Manual», révision 6.0, 727 pages, Octobre 2004.
- [26] Wang, J., Gosselin, C.M., «Static balancing of spatial three-degree-of-freedom parallel mechanisms», Mechanism and Machine Theory, Volume 34, No. 3, pages. 437-452, 1999.
- [27] Wang, J., Gosselin, C.M., «Static balancing of spatial three-degree-of-freedom parallel mechanisms», Mechanism and Machine Theory, Volume 34, No. 3, pages 437-452, 1999.

Annexe A

Animaux

La figure présentée est un arbre de classification du règne animal.



Annexe B

CD-rom d'accompagnement

Sur le CD-rom qui accompagne le mémoire se trouvent des fichiers complémentaires au projet Hexapode, sous format informatique.

B.1 Contenu du disque d'accompagnement

- « EquilibreStatique / EquilibreStatique.m », fichier de calcul matlab utilisé dans les études de chargement.
- « Dsp / Polyedres-Rotation-Matlab-Final / Polyedres-Rotation.mcp », fichier-projet Codewarrior téléchargé dans le microcontrôleur dans la version finale du robot.

- « Dsp / Polyedres-Rotation-Matlab-Final / InterfaceDSPmatlab », répertoire contenant les fichiers nécessaire à la consigne sous Matlab-Simulink par le port RS-232.
- « Images / *.jpg », banque d’images liées à Hexapode.
- « Pieces », répertoire contenant des pages de catalogue sur les divreses composantes.
- « Presentations-Poster-Articles / HexapodeCCToMMarticle.pdf », article «Hexapode : un robot explorateur tout terrain», référence bibliographique [20].
- « Presentations-Poster-Articles / HexapodeCCToMMpowerPoint.pdf », support informatique de la présentation effectuée lors du «CCToMM Symposium on Mechanisms, Machines, and Mechatronics» en mai 2005.
- « Presentations-Poster-Articles / HexapodeCriaqPowerPoint.pdf », support informatique de la présentation effectuée lors du «FÉA CRIAQ» en septembre 2005.
- « Presentations-Poster-Articles / HexapodeFilDesEvenements.pdf », article du journal Le Fil des événements sur Hexapode.
- « Presentations-Poster-Articles / HexapodePosterLabOratoire.pdf », affiche utilisée lors de la présentation effectuée lors du «Lab-Oratoire Public» en janvier 2005.
- « Presentations-Poster-Articles / HexapodeSCGMarticle.pdf », article «Hexapode : un robot marcheur», référence bibliographique [21].
- « Presentations-Poster-Articles / HexapodeSeminaireGMCpowerPoint.pdf », support informatique de la présentation effectuée lors du cours :«Séminaire en génie mécanique» en mai 2005.
- « Presentations-Poster-Articles / HexapodeMemoire-MathieuGoulet.pdf », Une copie électronique du présent mémoire.
- « Presentations-Poster-Articles / HexapodeLeSoleil.pdf », article du journal Le Soleil sur Hexapode.
- « Pro-E / Hexapode-V1-6-DeployableNoir / Hexapode-V3-2-Final », répertoire contenant les fichiers pro-engineer menant au prototype utilisant les mécanismes déployables.
- « Pro-E / Hexapode-V3-2-Final », répertoire contenant les fichiers pro-engineer menant au prototype final d’Hexapode.
- « Video / HexapodeRobotMarcheur.m2v », premier vidéos montrant Hexapode effectuer une trajectoire et exécuter trois démanches, en format mpeg2.
- « Video / HexapodeRobotMarcheur.mov », premier vidéos montrant Hexapode

- effectuer une trajectoire et exécuter trois démanches, en format Quick Time.
- « Video / Gait-Trajectory-dub.mov », second vidéo montrant Hexapode effectuer une trajectoire et exécuter trois démanches.
 - « Video / HexapodePatteLego.mov », vidéo montrant le prototype de patte en lego.
 - « Video / GaitsDivX.avi », vidéo en format DivX, inclus dans Gait-Trajectory-dub.mov.
 - « Video / TrajectoryDivX.avi », vidéo en format DivX, inclus dans Gait-Trajectory-dub.mov.

Annexe C

Programmation

Dans cette annexe sont présentées des sections des programmes ayant été utilisés dans le cadre du projet Hexapode.

C.1 Calcul du chargement

```
% statiqueHexapode (forme vectorielle)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Etude statique du chargement sur le robot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

clear all; clc; close all; digits(25);
%-----%
% Initialisation des parametres
%-----%
% Force de contact au pattes
syms NAvGx NAvGy NAvGz real 'd';
NAvG = [NAvGx; NAvGy; NAvGz];
syms NCeDx NCeDy NCeDz real 'd';
NCeD = [NCeDx; NCeDy; NCeDz];
syms NArGx NArGy NArGz real 'd';
NArG = [NArGx; NArGy; NArGz];

%-----%
% Reaction corps-coxa
syms RBCAvGx RBCAvGy RBCAvGz real;
RBCAvG = [RBCAvGx; RBCAvGy; RBCAvGz];
syms RBCCeDx RBCCeDy RBCCeDz real;
RBCCeD = [RBCCeDx; RBCCeDy; RBCCeDz];
syms RBCArGx RBCArGy RBCArGz real;
RBCArG = [RBCArGx; RBCArGy; RBCArGz];
syms MBCAvGx MBCAvGy MBCAvGz real;
MBCAvG = [MBCAvGx; MBCAvGy; MBCAvGz];
syms MBCCeDx MBCCeDy MBCCeDz real;
MBCCeD = [MBCCeDx; MBCCeDy; MBCCeDz];
syms MBCArGx MBCArGy MBCArGz real;
MBCArG = [MBCArGx; MBCArGy; MBCArGz];

%-----%
% Reaction femur-tibia
syms RFTAvGx RFTAvGy RFTAvGz real;
RFTAvG = [RFTAvGx; RFTAvGy; RFTAvGz];
syms RFTCeDx RFTCeDy RFTCeDz real;
RFTCeD = [RFTCeDx; RFTCeDy; RFTCeDz];
syms RFTArGx RFTArGy RFTArGz real;
RFTArG = [RFTArGx; RFTArGy; RFTArGz];
syms MFTAvGx MFTAvGy MFTAvGz real;

```



```

MFTAvG = [MFTAvGx; MFTAvGy; MFTAvGz];
syms MFTCeDx MFTCeDy MFTCeDz real;
MFTCeD = [MFTCeDx; MFTCeDy; MFTCeDz];
syms MFTArGx MFTArGy MFTArGz real;
MFTArG = [MFTArGx; MFTArGy; MFTArGz];

%-----%
% Reaction coxa-femur
syms RCFAvGx RCFAvGy RCFAvGz real 'd';
RCFAvG = [RCFAvGx; RCFAvGy; RCFAvGz];
syms RCFCeDx RCFCeDy RCFCeDz real;
RCFCeD = [RCFCeDx; RCFCeDy; RCFCeDz];
syms RCFArGx RCFArGy RCFArGz real;
RCFArG = [RCFArGx; RCFArGy; RCFArGz];
syms MCFAvGx MCFAvGy MCFAvGz real;
MCFAvG = [MCFAvGx; MCFAvGy; MCFAvGz];
syms MCFCeDx MCFCeDy MCFCeDz real;
MCFCeD = [MCFCeDx; MCFCeDy; MCFCeDz];
syms MCFArGx MCFArGy MCFArGz real;
MCFArG = [MCFArGx; MCFArGy; MCFArGz];

%-----%
% PGI
%-----%
g = 9.806E-3;
prec = [eps;eps;eps].*rand(3,1);

% Bras de levier en module des longueurs
a = [0,80,100];
b = [0,0,0];
alpha = [pi/2,0,0];

for o = 1:3
    HD(:, :, o) = [a;b;alpha];
end

```

```

% Position des centre de masse dans les repere des membrures
oCDMcorps = [0;0;0];
%oCDMtibia = [23;0;0];
oCDMtibia = [50;0;0];
oCDMfemur = [40;0;0];
oCDMcoxa = [0;0;0];

dRtibia(:, :) = [a(3);b(3);0];
dRfemur = [a(2);b(2);0];
dRcoxa = [a(1);b(1);0];

dRcorps = [106,0,-106;...
           45,-118,45;...
           0,0,0];

%-----%
% PGI_hexapode
%-----%
% Position des extremite des pattes
cas =1;

if cas == 1 % Orthogonal
pos = [106,0,-106;...
       125,-198,125;...
       -100,-100,-100];

elseif cas == 2 % extreme ortho
pos = dRcorps + [0,0,0;...
                150.71,-150.71,150.71;...
                -70.711,-70.711,-70.711];

elseif cas == 3 % extreme total
pos = dRcorps + [106,0,-106;...
                106.57,-150.71,106.57;...
                -70.711,-70.711,-70.711];
end

```

```

%-----%

Qcorps = eye(3); % Orientation du corps
[art,Qhex,Phex] = PGI_hexapode(pos,HD,dRcorps,Qcorps);

mTibia = 47;
mFemur = 62;
mCoxa = 75;
mCorps = 580 + 413;
mtot = 6*(mTibia+mFemur+mCoxa)+mCorps
Wcoxa = Qcorps*[0;0;-mCoxa*g];
Wfemur = Qcorps*[0;0;-mFemur*g];
Wtibia = Qcorps*[0;0;-mTibia*g];
Wcorps = Qcorps*[0;0;-mCorps*g];

Wtot = Wcorps + 6*(Wcoxa+Wfemur+Wtibia);
%-----%
% Position des centre de masse dans le repere global
LBcdmCAvG = Qcorps*dRcorps(:,1);
LBcdmCCeD = Qcorps*dRcorps(:,2);
LBcdmCArG = Qcorps*dRcorps(:,3);

% Femur
LCFcdmAvG = Phex(:, :, 2, 1)*oCDMfemur;
LCFcdmCeD = Phex(:, :, 2, 2)*oCDMfemur;
LCFcdmArG = Phex(:, :, 2, 3)*oCDMfemur;

LFcdmTAvG = Phex(:, :, 2, 1)*(dRfemur-oCDMfemur);
LFcdmTCeD = Phex(:, :, 2, 2)*(dRfemur-oCDMfemur);
LFcdmTArG = Phex(:, :, 2, 3)*(dRfemur-oCDMfemur);

LFAvG = Phex(:, :, 2, 1)*dRfemur;
LFCeD = Phex(:, :, 2, 2)*dRfemur;
LFArG = Phex(:, :, 2, 3)*dRfemur;

% Tibia

```

```

LFTcdmAvG = Phex(:,:,3,1)*oCDMtibia;
LFTcdmCeD = Phex(:,:,3,2)*oCDMtibia;
LFTcdmArG = Phex(:,:,3,3)*oCDMtibia;

LTcdmNAvG = Phex(:,:,3,1)*(dRtibia-oCDMtibia);
LTcdmNCeD = Phex(:,:,3,2)*(dRtibia-oCDMtibia);
LTcdmNArG = Phex(:,:,3,3)*(dRtibia-oCDMtibia);

LTAvg = Phex(:,:,3,1)*(dRtibia);
LTCeD = Phex(:,:,3,2)*(dRtibia);
LTAvg = Phex(:,:,3,3)*(dRtibia);

%-----%
% Systeme d'equation
%-----%
% Globale
%-----%
% Force sur dirrection Z seulement
NAvg(1:2) = [0;0];
NCeD(1:2) = [0;0];
NArG(1:2) = [0;0];

RBCAvG(1:2) = [0;0];
RBCCeD(1:2) = [0;0];
RBCArG(1:2) = [0;0];

RCFAvg(1:2) = [0;0];
RCFCeD(1:2) = [0;0];
RCFArG(1:2) = [0;0];

RFTAvg(1:2) = [0;0];
RFTCeD(1:2) = [0;0];
RFTAarG(1:2) = [0;0];

% Moment en X et Y seulement
MBCAvG(3) = 0;

```

```

MBCCeD(3) = 0;
MBCArG(3) = 0;

MCFAvG(3) = 0;
MFCFeD(3) = 0;
MCFArG(3) = 0;

MFTAvG(3) = 0;
MFTCeD(3) = 0;
MFTArG(3) = 0;

% Contraintes globales
SFGLOB = (Wcorps + 6*(Wcoxa + Wfemur + Wtibia) + NAvG + NCeD + NArG);
SMGLOB = (cross(LBcdmCAvG+LFAvG+LTAvg,NAvg) + ...
    cross(LBcdmCCeD+LFCeD+LTCeD,NCeD) + ...
    cross(LBcdmCArG+LFArG+LTAvg,NArG) + ...
    ...
    cross(LBcdmCAvG,Wcoxa) + ...
    cross(LBcdmCCeD,Wcoxa) + ...
    cross(LBcdmCArG,Wcoxa) + ...
    ...
    cross(LBcdmCAvG+LCFcdmAvG,Wfemur) + ...
    cross(LBcdmCCeD+LCFcdmCeD,Wfemur) + ...
    cross(LBcdmCArG+LCFcdmArG,Wfemur) + ...
    ...
    cross(LBcdmCAvG+LFAvG+LFTcdmAvG,Wtibia) + ...
    cross(LBcdmCCeD+LFCeD+LFTcdmCeD,Wtibia) + ...
    cross(LBcdmCArG+LFArG+LFTcdmArG,Wtibia));

% Resolution pour force de contact en Z
n = solve(SFGLOB(3),SMGLOB(1),SMGLOB(2));
NAvgz = eval(n.NAvGz);
NCeDz = eval(n.NCeDz);
NArGz = eval(n.NArGz);
check = NAvGz+NCeDz+NArGz;

```

```

%-----%
% Autres equations de contrainte interne
% Corps B
%-----%
% Somme force
SFB = RBCAvG + RBCCeD + RBCArG + Wcorps + ...
      3*(Wcoxa + Wfemur + Wtibia) + prec;

% Somme moment (bras de levier au cdm)
SMB = (cross(LBcdmCAvG,RBCAvG) + cross(LBcdmCCeD,RBCCeD) + ...
      cross(LBcdmCArG,RBCArG) + MBCAvG + MBCCeD + MBCArG) + prec;

%-----%
% Patte
%-----%
% Coxa C
% Somme force
SFCAvG = (-RBCAvG + RCFAvG + Wcoxa) + prec;
SFCCeD = (-RBCCeD + RCFCeD + Wcoxa) + prec;
SFCArG = (-RBCArG + RCFArG + Wcoxa) + prec;

% Somme moment (bras de levier au cdm)
SMCAvG = (-MBCAvG + MCFAvG) + prec;
SMCCeD = (-MBCCeD + MCFCeD) + prec;
SMCArG = (-MBCArG + MCFArG) + prec;

%-----%
% Femur F
% Somme force
SFFAvG = (-RCFAvG + RFTAvG + Wfemur) + prec;
SFFCeD = (-RCFCeD + RFTCeD + Wfemur) + prec;
SFFArG = (-RCFArG + RFTArG + Wfemur) + prec;

% Somme moment (bras de levier au cdm)
SMFAvG = (cross(-LCFcdmAvG,-RCFAvG) + cross(LFcdmTAvG,RFTAvG) - ...
      MCFAvG + MFTAvG) + prec;

```

```

SMFCeD = (cross(-LCFcdmCeD,-RCFCeD) + cross(LFcdmTCeD,RFTCeD) - ...
  MFCFeD + MFTCeD) + prec;
SMFArG = (cross(-LCFcdmArG,-RCFArG) + cross(LFcdmTArG,RFTAarG) - ...
  MCFArG + MFTAarG) + prec;

%-----%
% Tibia T
% Somme force
SFTAavG = (-RFTAavG + NAvG + Wtibia) + prec;
SFTCeD = (-RFTCeD + NCeD + Wtibia) + prec;
SFTAarG = (-RFTAarG + NArG + Wtibia) + prec;

% Somme moments (bras de levier au cdm)
SMTAvG = (cross(-LFTcdmAvG,-RFTAavG) - MFTAavG + cross(LTcdmNAvG,NAvG))...
  + prec;
SMTCeD = (cross(-LFTcdmCeD,-RFTCeD) - MFTCeD + cross(LTcdmNCeD,NCeD))...
  + prec;
SMTArG = (cross(-LFTcdmArG,-RFTAarG) - MFTAarG + cross(LTcdmNArG,NArG))...
  + prec;

%-----%
% Resolution
%-----%

sol = solve(...
  SFB(3),...
  SMB(1),SMB(2),...
  ...
  SFCAvG(3),...
  SMCAvG(1),SMCAvG(2),...
  SFFAvG(3),...
  SMFAvG(1),SMFAvG(2),...
  SFTAavG(3),...
  SMTAvG(1),SMTAvG(2),...
  ...
  SFCCeD(3),...

```

```

SMCCeD(1),SMCCeD(2),...
SFFCeD(3),...
SMFCeD(1),SMFCeD(2),...
SFTCeD(3),...
SMTCeD(1),SMTCeD(2),...
...
SFCArG(3),...
SMCArG(1),SMCArG(2),...
SFFArG(3),...
SMFArG(1),SMFArG(2),...
SFTArG(3),...
SMTArG(1),SMTArG(2));

NAvGz = eval(sol.NAvGz);
NCeDz = eval(sol.NCeDz);
NArGz = eval(sol.NArGz);

check = NAvGz+NCeDz+NArGz;

MBCAvGx = eval(sol.MBCArGx);
MBCCeDx = eval(sol.MBCCeDx);
MBCArGx = eval(sol.MBCAvGx);

MBCArGy = eval(sol.MBCArGy);
MBCCeDy = eval(sol.MBCCeDy);
MBCArGy = eval(sol.MBCAvGy);

MCFAvGx = eval(sol.MCFArGx);
MCFCeDx = eval(sol.MCFCeDx);
MCFArGx = eval(sol.MCFAvGx);

MCFAvGy = eval(sol.MCFArGy);
MCFCeDy = eval(sol.MCFCeDy);
MCFArGy = eval(sol.MCFAvGy);

MFTAvGx = eval(sol.MFTArGx);

```



```

MFTCeDx = eval(sol.MFTCeDx);
MFTArGx = eval(sol.MFTAvGx);

MFTAvGy = eval(sol.MFTAvGy);
MFTCeDy = eval(sol.MFTCeDy);
MFTArGy = eval(sol.MFTArGy);

RBCArGz = eval(sol.RBCArGz);
RBCCeDz = eval(sol.RBCCeDz);
RBCAvGz = eval(sol.RBCAvGz);

RCFAvGz = eval(sol.RCFAvGz);
RCFCeDz = eval(sol.RCFCeDz);
RCFArGz = eval(sol.RCFArGz);

RFTAvGz = eval(sol.RFTAvGz);
RFTCeDz = eval(sol.RFTCeDz);
RFTArGz = eval(sol.RFTArGz);

MCFAvG_patte = Phex(:, :, 2, 1)'*MCFAvG;
subs(MCFAvG_patte)

MCFCeD_patte = Phex(:, :, 2, 2)'*MCFCeD;
subs(MCFCeD_patte)

MCFArG_patte = Phex(:, :, 2, 3)'*MCFArG;
subs(MCFArG_patte)

%-----%
affichage = 1;
if affichage == 1
    figure;
    hold on;
    plot3([LBcdmCAvG(1), LBcdmCCeD(1), LBcdmCArG(1), LBcdmCAvG(1)], ...
          [LBcdmCAvG(2), LBcdmCCeD(2), LBcdmCArG(2), LBcdmCAvG(2)], ...
          [LBcdmCAvG(3), LBcdmCCeD(3), LBcdmCArG(3), LBcdmCAvG(3)], 'k-');

```

```

plot3([0, LBcdmCAvG(1), LBcdmCAvG(1)+LFAvG(1), ...
      LBcdmCAvG(1)+LFAvG(1)+LTAvg(1)],...
      [0, LBcdmCAvG(2), LBcdmCAvG(2)+LFAvG(2), ...
      LBcdmCAvG(2)+LFAvG(2)+LTAvg(2)],...
      [0, LBcdmCAvG(3), LBcdmCAvG(3)+LFAvG(3), ...
      LBcdmCAvG(3)+LFAvG(3)+LTAvg(3)], 'o-b');

plot3([0, LBcdmCCeD(1), LBcdmCCeD(1)+LFCeD(1), ...
      LBcdmCCeD(1)+LFCeD(1)+LTCeD(1)],...
      [0, LBcdmCCeD(2), LBcdmCCeD(2)+LFCeD(2), ...
      LBcdmCCeD(2)+LFCeD(2)+LTCeD(2)],...
      [0, LBcdmCCeD(3), LBcdmCCeD(3)+LFCeD(3), ...
      LBcdmCCeD(3)+LFCeD(3)+LTCeD(3)], 'o-r');

plot3([0, LBcdmCArG(1), LBcdmCArG(1)+LFArG(1), ...
      LBcdmCArG(1)+LFArG(1)+LTAvg(1)],...
      [0, LBcdmCArG(2), LBcdmCArG(2)+LFArG(2), ...
      LBcdmCArG(2)+LFArG(2)+LTAvg(2)],...
      [0, LBcdmCArG(3), LBcdmCArG(3)+LFArG(3), ...
      LBcdmCArG(3)+LFArG(3)+LTAvg(3)], 'o-g');

% Femur
plot3(LBcdmCAvG(1)+LCFcdmAvG(1), LBcdmCAvG(2)+LCFcdmAvG(2),...
      LBcdmCAvG(3)+LCFcdmAvG(3), 'xk');
plot3(LBcdmCCeD(1)+LCFcdmCeD(1), LBcdmCCeD(2)+LCFcdmCeD(2),...
      LBcdmCCeD(3)+LCFcdmCeD(3), 'xk');
plot3(LBcdmCArG(1)+LCFcdmArG(1), LBcdmCArG(2)+LCFcdmArG(2),...
      LBcdmCArG(3)+LCFcdmArG(3), 'xk');

% Tibia
plot3(LBcdmCAvG(1)+LFAvG(1)+LFTcdmAvG(1), LBcdmCAvG(2)+LFAvG(2)+...
      LFTcdmAvG(2), LBcdmCAvG(3)+LFAvG(3)+LFTcdmAvG(3), 'xk');
plot3(LBcdmCCeD(1)+LFCeD(1)+LFTcdmCeD(1), LBcdmCCeD(2)+LFCeD(2)+...

```

```

    LFTcdmCeD(2),LBcdmCCeD(3)+LFCeD(3)+LFTcdmCeD(3),'xk');
plot3(LBcdmCArG(1)+LFArG(1)+LFTcdmArG(1),LBcdmCArG(2)+LFArG(2)+...
    LFTcdmArG(2),LBcdmCArG(3)+LFArG(3)+LFTcdmArG(3),'xk');

xlabel('X'); ylabel('Y'); zlabel('Z');
axis equal;
% view(3)
view([0,0,180]);

% Affichage du cercle de contact
[cC,r] = cercle3pts(pos(1:2,:))';
t = 0:0.01:2*pi+0.01;
for o = 1:size(t,2)
    x(o) = cC(1) + r*cos(t(o));
    y(o) = cC(2) + r*sin(t(o));
end
plot3(x,y,ones(1,size(x,2))*pos(3,1),'c-');
plot3(cC(1),cC(2),pos(3,1),'c+');

% Affichage du cercle d'ancrage
[cA,r] = cercle3pts(dRcorps(1:2,:))';
t = 0:0.01:2*pi+0.01;
for o = 1:size(t,2)
    x(o) = cA(1) + r*cos(t(o));
    y(o) = cA(2) + r*sin(t(o));
end
plot3(x,y,ones(1,size(x,2))*pos(3,1),'m-');
plot3(cA(1),cA(2),pos(3,1),'m+');

distCentre = norm(cA-cC)
end

```

C.2 Architecture de la structure d'Hexapode

```

#ifndef PATTE_H
#define PATTE_H

/* ----- STRUCT Patte ----- */
// ---- Standard ----
typedef struct
{
// Offset de la fixation de la patte p/r au centre du robot
float Xoff, Yoff, Zoff;
float Xortho, Yortho, Zortho;

}STD;

// ---- Position articulaire imposee ----
typedef struct
{
// Position
float EXTREMEdeg[2]; // Valeurs limites aux articulation (degre)
float CALpwOFFSET; // position centrale p/r à theta = 0 (Pulse width)
float CALpwOFFSETload; // position centrale p/r à theta = 0 (Pulse width)
float CALpwMULT; // Facteur multiplicatif pour les valeur d'angles
float CALpwCOTE; // Facteur multiplicatif pour les valeur d'angles

float THETAdeg; // Valeur articulaire
int THETApw; // Largeur impulsion (commande PWM)

}ARTICULATION;

// ---- Patte unique ----
typedef struct
{
int ADCval;

```

```

int ADCdiv;
float ADCmoy;

}CONTACT;

// ---- Patte unique ----
typedef struct
{
float X, Y, Z;
float dXtrajTOT, dYtrajTOT, dZtrajTOT;
float dXtrajINC, dYtrajINC, dZtrajINC;
int clock; // Compteur d'horloge
float PHIact;
int mode; //newmode;
int sol;
int INCchange;
int transition;

ARTICULATION coxa, femur, tibia;
STD std;
CONTACT contact;

}PATTE;

// ---- Patte unique ----
typedef struct
{
float dXpousse, dYpousse, dZpousse;
float dXretour, dYretour, dZretour, dZpas;
float Zpas, PHItot;

float phase;

int Ttot; // Periode complete
int Tretour; // Periode du mode de retour
int Tpousse; // Periode du mode de pousse

```

```
int INCnb;  
//int newmode;  
  
}POLY;  
  
// ---- Robot complet ----  
typedef struct  
{  
PATTE AvG;  
PATTE AvD;  
PATTE CeG;  
PATTE CeD;  
PATTE ArG;  
PATTE ArD;  
POLY poly;  
int sol;  
  
}HEXAPODE;  
  
#endif // PATTE_H
```

Annexe D

Pièces

Dans cette annexe sont présentées les fiches de caractéristiques des pièces qui ont été utilisées dans le prototype final du robot marcheur Hexapode.

D.1 Composantes mécaniques

Dans cette section sont présentées les composantes mécaniques qui constituent le prototype final d'Hexapode.

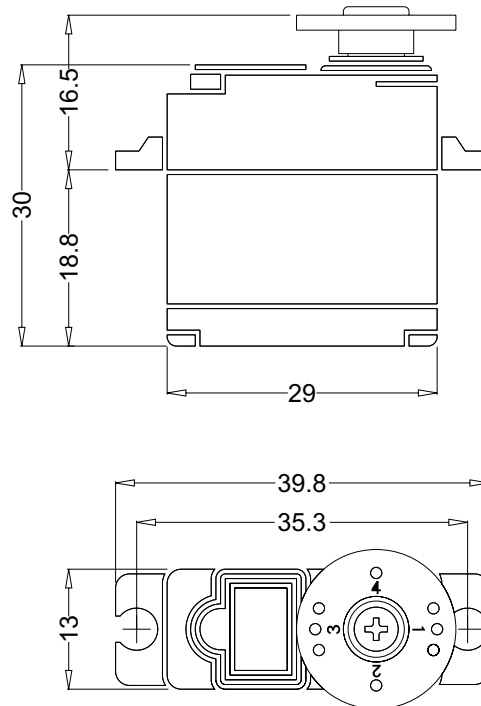
- Moteur Hitec HS85mg
- Moteur Hitec HS225mg

- Palier lisse Igus g300 de type GFM-0405-04
- Ressorts en torsion Associated spring Raymond T059-360-459

ANNOUNCED SPECIFICATIONS OF HS-85MG⁺ MIGHT MICRO METAL GEAR SERVO

1. TECHNICAL VALUES

CONTROL SYSTEM	: +PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	: 4.8V TO 6.0V	
OPERATING TEMPERATURE RANGE	: -20° C TO +60° C	
TEST VOLTAGE	: AT 4.8V	: AT 6.0V
OPERATING SPEED	: 0.16sec/60° AT NO LOAD	: 0.14sec/60° AT NO LOAD
STALL TORQUE	: 3.0kg.cm(41.66oz.in)	: 3.5kg.cm(48.60oz.in)
OPERATING ANGLE	: 40° ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	: CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
CURRENT DRAIN	: 8mA/IDLE AND 240mA NO LOAD RUNNING	
DEAD BAND WIDTH	: 8usec	
CONNECTOR WIRE LENGTH	: 160mm (6.29in)	
DIMENSIONS	: 29x13x30mm (1.14x0.51x1.18in)	
WEIGHT	: 21.9g (0.77oz)	



2. FEATURES

- 3-POLES FERRITE MOTOR
- ONE BALL BEARING
- DIRECT POTENTIOMETER DRIVE
- HYBRID I.C
- 4-METAL GEARS

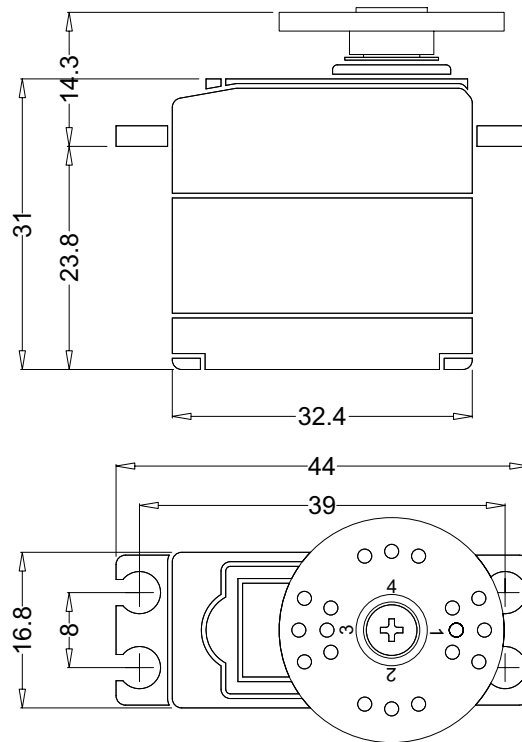
3. APPLICATIONS

- FAST ELECTRIC BOATS
- ROCKET PLANES AND SAILPLANE WINGS

ANNOUNCED SPECIFICATION OF HS-225MG MIGHT MINI METAL GEAR SERVO

1. TECHNICAL VALUES

CONTROL SYSTEM	: +PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	: 4.8V TO 6.0V	
OPERATING TEMPERATURE RANGE	: -20° C TO +60° C	
TEST VOLTAGE	: AT 4.8V	AT 6.0V
OPERATING SPEED	: 0.14sec/60° AT NO LOAD	0.11sec/60° AT NO LOAD
STALL TORQUE	: 3.9kg.cm(54.15oz.in)	4.8kg.cm(66.65oz.in)
OPERATING ANGLE	: 45°/ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	: CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
CURRENT DRAIN	: 8mA/IDLE AND 300mA NO LOAD RUNNING	
DEAD BAND WIDTH	: 8usec	
CONNECTOR WIRE LENGTH	: 300mm(11.81in)	
DIMENSIONS	: 32.4x16.8x31mm(1.27x0.66x1.22in)	
WEIGHT	: 31g (1.09oz)	



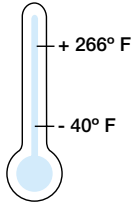
2. FEATURES

- 3-POLE FERRITE MOTOR
- LONG LIFE POTENTIOMETER
- 1-OILITE BUSHING AND 1-BALL BEARING
- INDIRECT POTENTIOMETER DRIVE
- 3-METAL GEARS
- HYBRID I.C

3. APPLICATIONS

- AIRCRAFT UP TO 10 POUNDS
- STEERING SERVO FOR ON-ROAD 1/12TH 1/10 ELECTRIC
- FAST ELECTRIC BOATS

Price Index



3 Styles
More than 900 Dimensions
Inner diameters
From 1/8 - 3 in. (1 - 150 mm)

Structure of the Part Number:

G S I - 0 2 0 3 - 0 3

d1 d2 b1 (s-Thrust)

Dimension: I - Inch (in 1/16")
M - Metric (in mm)

Type: S - Sleeve
F - Flange
T - Thrust Washer

Material



When to use iglide® G300 plain bearings:

- When you need an economical all-around performance bearing
- For above average loads
- For low to average running speeds
- When the bearing needs to run on different shaft materials
- For oscillating and rotational movements



When not to use iglide® G300 plain bearings:

- When mechanical reaming of the wall surface is necessary
 - iglide® M250
- When the highest wear resistance is necessary
 - iglide® L280
- If temperatures are constantly greater than 266°F
 - iglide® T500, F

Table 2.1: Material Data

General Properties	Unit	iglide® G300	Testing Method
Density	g/cm ³	1.45	
Color		dark gray	
Max. moisture absorption at 73°F / 50% r.h.	% weight	0.7	DIN 53495
Max. moisture absorption	% weight	4.0	
Coefficient of friction, dynamic against steel	μ	0.08 - 0.15	
p x v-value, max. (dry)	psi x fpm	12,000	

Mechanical Properties

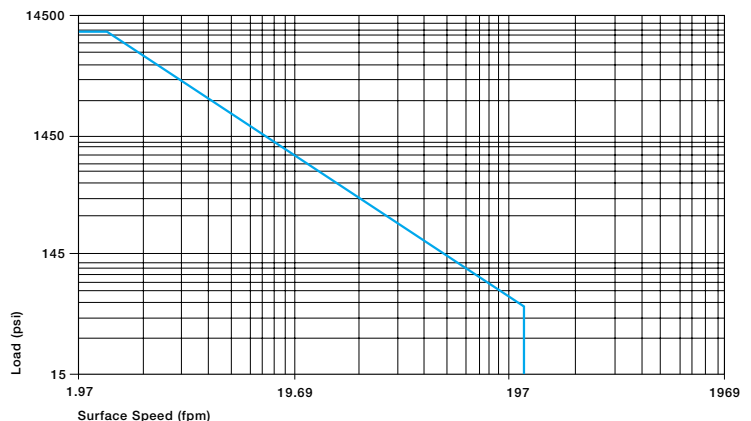
Modulus of elasticity	psi	1,131,000	DIN 53457
Tensile strength at 68°F	psi	30,450	DIN 53452
Compressive strength	psi	11,310	
Max. static surface pressure (68°F)	psi	11,600	
Shore D-hardness		81	DIN 53505

Physical and Thermal Properties

Max. long-term application temperature	°F	266	
Max. short-term application temperature	°F	428	
Min. application temperature	°F	-40	
Thermal conductivity	(W/m x K)	0.24	ASTM C 177
Coefficient of thermal expansion (at 73°F)	(K ⁻¹ x 10 ⁻⁵)	9	DIN 53752

Electrical Properties

Specific volume resistance	Ωcm	> 10 ¹³	DIN IEC 93
Surface resistance	Ω	> 10 ¹¹	DIN 53482



Graph 2.1: Permissible p x v - values for iglide® G300 running dry against a steel shaft, at 68°F

iglide® G300

Telephone 1-888-803-1895
Fax 1-401-438-7680

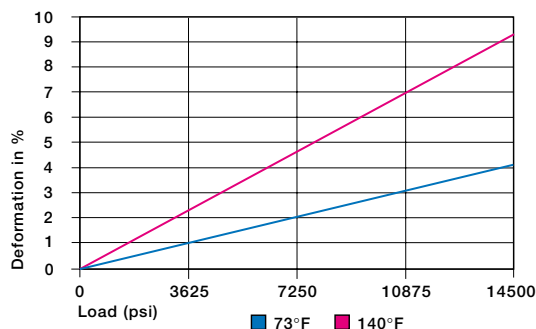
igus®, inc.
USA/Canada
Mexico

Internet: <http://www.igus.de>
email: info@igus.de

Compressive Strength

Picture 2.2 shows the elastic deformation of iglide® G300 during radial loading. At the maximum permissible load of 11,600 psi, the deformation is less than 5%. The plastic deformation is minimal up to a pressure of approximately 14,500 psi. However, it is also a result of the cycle time.

➤ Compressive Strength, Page 1.12



Graph 2.2: Deformation under load and temperature

Permissible Surface Speeds

iglide® G300 has been developed for low to medium surface speeds. The maximum values shown in Table 2.2 can only be achieved at low pressure loads. At the given speeds, friction can cause a temperature increase to maximum permissible levels. In practice, though, this temperature level is rarely reached, due to varying application conditions.

➤ Surface speed, Page 1.14
➤ p x v Value, Page 1.15

fpm	Rotating	Oscillating	Linear
Continuous	196	295	787
Short-term	393	492	984

Table 2.2: Maximum running speed

Temperatures

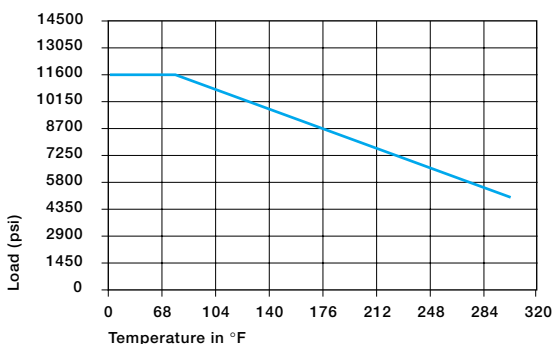
Application temperatures affect the properties of plain bearings greatly. The short-term maximum temperature is 428°F, this allows the use of iglide® G300 plain bearings in heat treating applications in which the bearings are not subjected to additional loading.

With increasing temperatures, the compressive strength of iglide® G300 plain bearings decreases. The graph 2.3 shows this inverse relationship. However, at the long-term maximum temperature of 266°F, the permissible surface pressure is still above 5,800 psi. The ambient temperatures that are prevalent in applications also has an effect on the bearing wear. With increasing temperatures, the wear increases and this effect is notable starting at the temperature of 248°F.

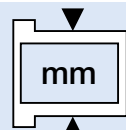
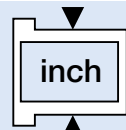
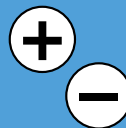
➤ Application Temperatures, Page 1.16

iglide® G300	Application Temperature
Minimum	- 40 °F
Max. long-term	+ 266 °F
Max. short-term	+ 428 °F

Table 2.3: Temperature limits for iglide® G300



Graph 2.3: Recommended maximum permissible static surface pressure of iglide® G300 as a result of temperature



Installation Tolerances

iglide® G300 plain bearings are meant to be oversized before being pressfit. After proper installation into a recommended housing bore, the inner diameter adjusts to meet our specified tolerances. Please adhere to the catalog specifications for housing bore and recommended shaft sizes. This will help to ensure optimal performance of iglide® plain bearings.

Please contact an iglide® technical expert for support.

- Tolerance Table, Page 1.24
- Testing Methods, Page 1.25

iglide® G300

Telephone 1-888-803-1895
Fax 1-401-438-7680

igus®, inc.
USA/Canada
Mexico

Internet: <http://www.igus.de>
email: info@igus.de

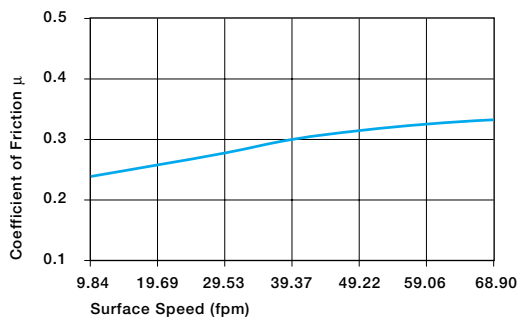
2.4

Friction and Wear

Similar to wear resistance, the coefficient of friction μ also changes with the load. The coefficient of friction decreases with increasing loads, whereas an increase in surface speed causes an increase of the coefficient of friction. This relationship explains the excellent results of iglide® G300 plain bearings for high loads and low speeds (See Graph 2.4 and 2.5).

The friction and wear are also dependent, to a large degree, on the shaft partner. Shafts that are too smooth, increase both the coefficient of friction and the wear of the bearing. For iglide® G300, a ground surface with an average roughness $R_a = 32$ rms is recommended (See Graph 2.6).

- Coefficients of friction and surfaces, Page 1.17
- Wear Resistance, Page 1.18

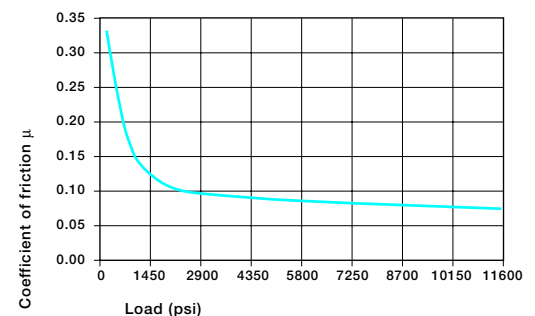


Graph 2.4: Coefficient of friction of iglide® G300 as a result of the running speed; $p = 108$ psi

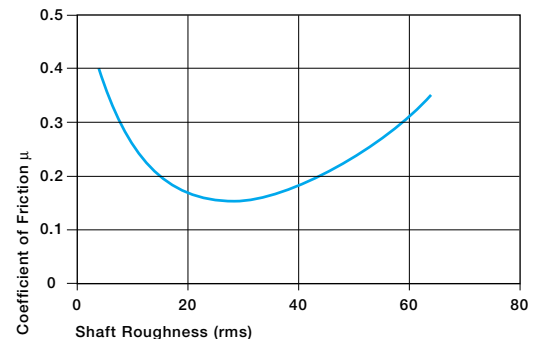
iglide® G300 Dry Grease Oil Water

Coefficient of Friction μ	Dry	Grease	Oil	Water
	0.08 - 0.15	0.09	0.04	0.04

Table 2.4: Coefficient of friction for iglide® G300 against steel ($R_a = 40$ rms, 50 HRC)



Graph 2.5: Coefficient of friction of iglide® G300 as a result of the load



Graph 2.6: Coefficient of friction as result of the shaft surface (Shaft - Cold Rolled Steel)

Shaft Materials

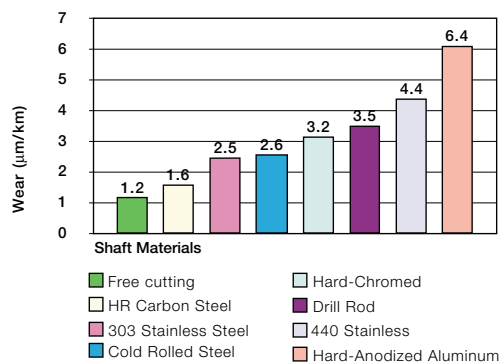
Graph 2.7 and 2.8 show results of testing different shaft materials with plain bearings made of iglide® G300.

In Graph 2.7 it is observed that iglide® G300 can be combined with various shaft materials. The simple shaft materials of free-cutting steel and HR Carbon Steel have proven best at low loads. This helps to design cost-effective systems, since both iglide® G300 and the sliding partner are economically priced.

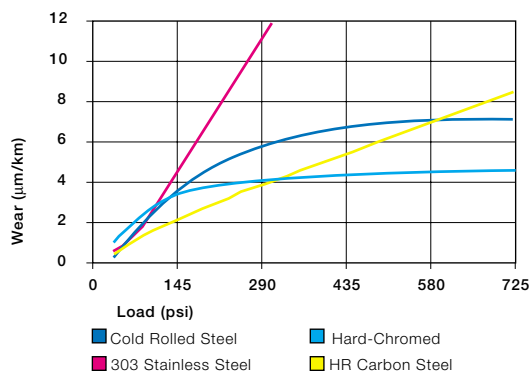
It is important to notice that with increasing loads, the recommended hardness of the shaft increases. The “soft” shafts tend to wear more easily and thus increase the wear of the overall system. If the loads exceed 290 psi, it is important to recognize that the wear rate (the slope of the curves) clearly decreases with the hard shaft materials.

The comparison of rotational movements to oscillating movements shows that iglide® G300 can provide advantages in oscillating movements. The wear of the bearing is smaller for equivalent conditions. The higher the load, the larger the difference. This means that iglide® G300 can be used for oscillating movements that are well above the given maximum load of 11,600 psi. For these loads, the use of hardened shafts is recommended. In addition to the shaft materials presented here, many others have been tested. If the shaft material you plan on using is not contained in the test results presented here, please contact us.

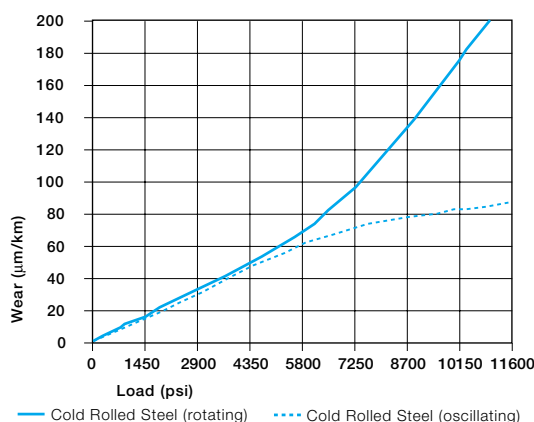
► Shaft Materials, Page 1.21



Graph 2.7: Wear of iglide® G300, rotating with different shaft materials, load p = 108 psi, v = 98 fpm



Graph 2.8: Wear with different shaft materials in rotational operation, as a result of the load



Graph 2.9: Wear for pivoting and rotating applications with shaft material Cold Rolled Steel 1018, as a result of the load

Chemical Resistance

iglide® G300 plain bearings have strong resistance to chemicals. They are also resistant to most lubricants.

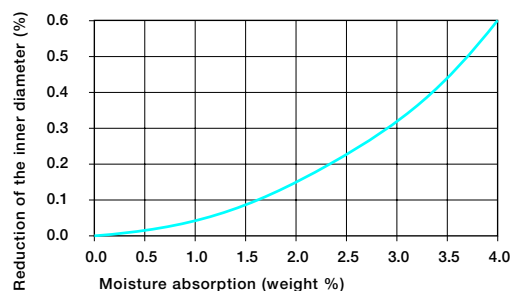
iglide® G300 plain bearings are not affected by most weak organic and inorganic acids.

The moisture absorption of iglide® G300 plain bearings is approximately 1% in the standard atmosphere. The saturation limit submerged in water is 4%. This must be taken into account for these types of applications.

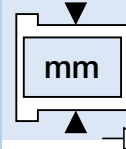
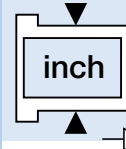
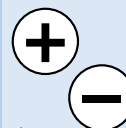
► Chemicals Table, Page 1.26

Medium	Resistance
Alcohols	Resistant
Chlorinated hydrocarbons	Resistant
Esters	Not Resistant
Greases, oils	Resistant
Ketone	Conditionally Resistant
Fuels	Resistant
Weak acids	Conditionally Resistant
Strong acids	Not Resistant
Weak lyes	Conditionally Resistant
Strong lyes	Conditionally Resistant

Table 2.5: Chemical resistance of iglide® G300- detailed list, Page 35.1



Graph 2.10: Effect of moisture absorption on iglide® G300 plain bearings



Radiation Resistance

Plain bearings made from iglide® G300 are resistant to radiation up to an intensity of 3×10^5 Gy.

UV Resistance

iglide® G300 plain bearings are permanently resistant to UV-radiation.

Vacuum

iglide® G300 plain bearings outgas in a vacuum. Use in a vacuum environment is only possible for dehumidified bearings.

Electrical Properties

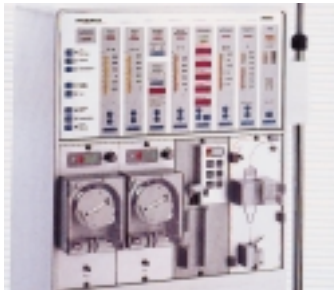
iglide® G300 plain bearings are electrically insulating.

iglide® G300

Specific volume resistance	> $10^{13} \Omega\text{cm}$
Surface resistance	> $10^{11} \Omega$

Table 2.6: Electrical properties of iglide® G300

Application Examples



Picture 2.1: Reliable under high load, wear-resistant during continuous rotational use



Picture 2.2: Tested at a load of 4046 lbs for 10,000 cycles, resulted in no measurable wear



Picture 2.3: The pneumatic rotational drive unit in steam lines at temperatures up to 275°F



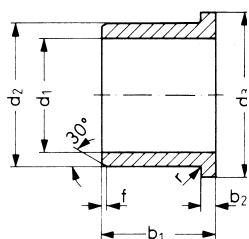
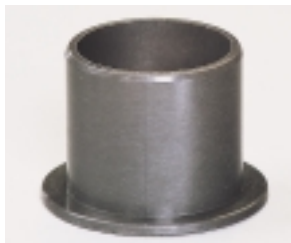
Picture 2.4: Vibrations, dirt, and temperatures up to 266°F characterize the area surrounding the engine



Picture 2.5: Conveyor chains: Through edge loading, short-term surface pressures of over 7,250 psi can occur



Picture 2.6: iglide® G300 plain bearings have proven themselves in control levers and pedals of farm tractors and construction vehicles


 $r = \max. 0.5$

Based on I.D.

 $f = 0.5 \rightarrow d_1 \leq 10$
 $f = 0.8 \rightarrow 10 < d_1 \leq 30$
 $f = 1.2 \rightarrow d_1 > 32$
Length Tolerance (b1)

Length (mm)	Tolerance (h13) (µm)
1 to 3	-0 / -140
> 3 to 6	-0 / -180
> 6 to 10	-0 / -220
> 10 to 18	-0 / -270
> 18 to 30	-0 / -330
> 30 to 50	-0 / -390
> 50 to 80	-0 / -460

Dimensions according to ISO 3547-1 and special dimensions

Part Number	d1 ¹⁾		d2	d3	b1	b2	I.D. After Pressfit		Housing Bore		Shaft Size	
	after Pressfit in Ø H7						Max.	Min.	Max.	Min.	Max.	Min.
GFM-1012-035	10.0	+0.025 +0.083	12.0	18.0	3.5	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-04	10.0	+0.025 +0.083	12.0	18.0	4.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-05	10.0	+0.025 +0.083	12.0	18.0	5.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-06	10.0	+0.025 +0.083	12.0	18.0	6.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-07	10.0	+0.025 +0.083	12.0	18.0	7.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-09	10.0	+0.025 +0.083	12.0	18.0	9.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-10	10.0	+0.025 +0.083	12.0	18.0	10.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-12	10.0	+0.025 +0.083	12.0	18.0	12.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-15	10.0	+0.025 +0.083	12.0	18.0	15.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1012-17	10.0	+0.025 +0.083	12.0	18.0	17.0	1.0	10.098	10.040	12.018	12.000	10.000	9.964
GFM-1213-03	12.0	+0.016 +0.059	13.0	17.0	3.0	0.5	12.059	12.016	13.018	13.000	12.000	11.957
GFM-1213-12	12.0	+0.016 +0.059	13.0	17.0	12.0	0.5	12.059	12.016	13.018	13.000	12.000	11.957
GFM-1214-06	12.0	+0.032 +0.102	14.0	20.0	6.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-07	12.0	+0.032 +0.102	14.0	20.0	7.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-09	12.0	+0.032 +0.102	14.0	20.0	9.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-10	12.0	+0.032 +0.102	14.0	20.0	10.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-11	12.0	+0.032 +0.102	14.0	20.0	11.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-12	12.0	+0.032 +0.102	14.0	20.0	12.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-15	12.0	+0.032 +0.102	14.0	20.0	15.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-17	12.0	+0.032 +0.102	14.0	20.0	17.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-20	12.0	+0.032 +0.102	14.0	20.0	20.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1214-24	12.0	+0.032 +0.102	14.0	20.0	24.0	1.0	12.102	12.032	14.018	14.000	12.000	11.957
GFM-1315-06	13.0	+0.032 +0.102	15.0	22.0	6.0	1.0	13.102	13.032	15.018	15.000	13.000	12.957
GFM-1416-03	14.0	+0.032 +0.102	16.0	22.0	3.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-04	14.0	+0.032 +0.102	16.0	22.0	4.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-06	14.0	+0.032 +0.102	16.0	22.0	6.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-08	14.0	+0.032 +0.102	16.0	22.0	8.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-12	14.0	+0.032 +0.102	16.0	22.0	12.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-17	14.0	+0.032 +0.102	16.0	22.0	17.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1416-21	14.0	+0.032 +0.102	16.0	22.0	21.0	1.0	14.102	14.032	16.018	16.000	14.000	13.957
GFM-1516-02	15.0	+0.016 +0.059	16.0	20.0	2.0	0.5	15.059	15.016	16.018	16.000	15.000	14.957
GFM-1516-025	15.0	+0.016 +0.059	16.0	20.0	2.5	0.5	15.059	15.016	16.018	16.000	15.000	14.957
GFM-1516-03	15.0	+0.016 +0.059	16.0	20.0	3.0	0.5	15.059	15.016	16.018	16.000	15.000	14.957
GFM-1516-15	15.0	+0.016 +0.059	16.0	20.0	15.0	0.5	15.059	15.016	16.018	16.000	15.000	14.957
GFM-1517-04	15.0	+0.032 +0.102	17.0	23.0	4.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-045	15.0	+0.032 +0.102	17.0	23.0	4.5	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-05	15.0	+0.032 +0.102	17.0	23.0	5.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-09	15.0	+0.032 +0.102	17.0	23.0	9.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-12	15.0	+0.032 +0.102	17.0	23.0	12.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-17	15.0	+0.032 +0.102	17.0	23.0	17.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-1517-20	15.0	+0.032 +0.102	17.0	23.0	20.0	1.0	15.102	15.032	17.018	17.000	15.000	14.957
GFM-151824-32	15.0	+0.032 +0.102	18.0	24.0	32.0	1.0	15.102	15.032	18.018	18.000	15.000	14.957
GFM-1618-09	16.0	+0.032 +0.102	18.0	24.0	9.0	1.0	16.102	16.032	18.018	18.000	16.000	15.957
GFM-1618-12	16.0	+0.032 +0.102	18.0	24.0	12.0	1.0	16.102	16.032	18.018	18.000	16.000	15.957
GFM-1618-17	16.0	+0.032 +0.102	18.0	24.0	17.0	1.0	16.102	16.032	18.018	18.000	16.000	15.957
GFM-1618-21	16.0	+0.032 +0.102	18.0	24.0	21.0	1.0	16.102	16.032	18.018	18.000	16.000	15.957
GFM-1622-12	16.0	+0.032 +0.102	22.0	25.0	12.0	1.0	16.102	16.032	18.018	18.000	16.000	15.957

Flange - MM

iglide® G300

 Telephone 1-888-803-1895
 Fax 1-401-438-7680

 igus®, inc.
 USA/Canada
 Mexico

 Internet: <http://www.igus.com>
 email: webmaster@igus.com

TORSION SPRINGS

Stock sizes in stainless steel and music wire

SPEC stainless steel torsion springs are widely useful to store and release energy of rotation or to maintain a pressure over a short distance. Our stock selection includes stainless steel torsion springs with four end positions and music wire springs with three end positions as shown in the drawings.

SPEC torsion springs are normally used over a supporting mandrel or arbor. Suggested mandrel sizes allow about 10% clearance at the deflections listed. If greater deflections are used, the arbor size should be reduced. Sufficient room (minimum axial space) must be provided in the assembly for the spring to function properly.

SPEC torsion springs should be used in the direction that winds the coils. In the unwinding direction the maximum load is lower because of residual stresses.

Torque values listed are suitable for average conditions. These values can be increased about 20% for static conditions with only slight setting.

Material

Stainless Steel

Commercial Type 302 ASTM-A313 or AMS 5688 (chemical and physical only)

Music Wire

ASTM-A228 or AMS 5112

Certificate of compliance available on request. Certificate of chemical analysis available at additional charge.

Ends

Straight torsion ends are standard.

Special end treatment available. Allow additional time.

Finish

Plain finish is standard. Allow additional time for special finishes.

Direction of Helix

Must be specified by adding a suffix to the catalog number. Use L for left-hand wound, or R for right-hand wound. **See illustration below.**

Tolerances

Torque is for reference only and should not be specified.

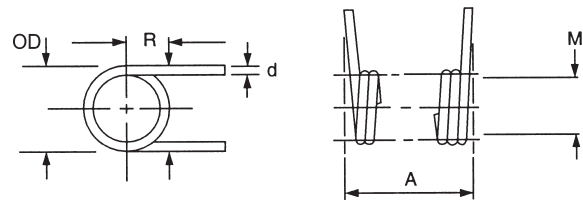
O.D. ±5% Stainless Steel

±2% Music Wire

For inspection purposes, the load should be applied at 1/2 leg length (E). Using other lengths appreciably alters the active length of wire and affects the test results.

The reference torque values listed can be translated to the approximate direct load by use of the formula $P = M/E_n$ where P is the load applied at the new length E_n . Example: for part T012-090-055, what is the load when $E_n = 0.187$? $P = M/E_n = 0.047/0.187 = 0.251b$.

Torque values at intermediate deflections can be computed by direct proration. Example: For part T030-180-250, the torque at 90° deflection is 0.312 in-lb (35.3 N-mm)



Position of Legs

STAINLESS STEEL SPRINGS

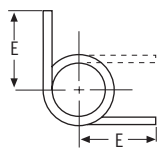


Fig 1. 90° Deflection

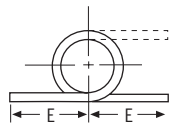


Fig 2. 180° Deflection

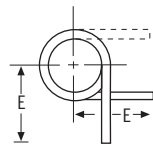


Fig 3. 270° Deflection

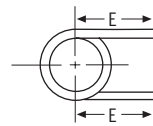


Fig 4. 360° Deflection

MUSIC WIRE SPRINGS

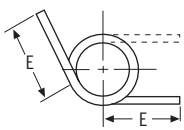


Fig 5. 120° Deflection

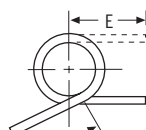


Fig 6. 210° Deflection

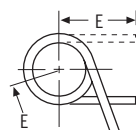
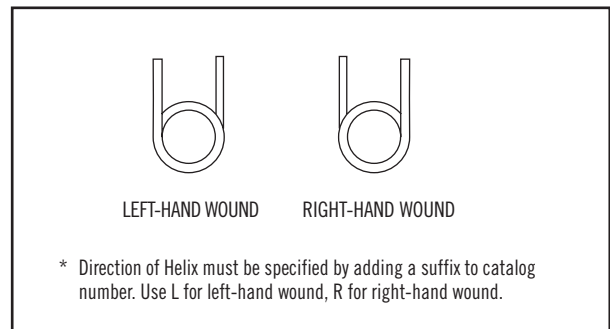


Fig 7. 300° Deflection

- d= Wire diameter
- OD= Outside diameter
- T= Torque
- R= Loaded position all parts
- M= Recommended Mandrel Size
- E= Leg length (from centerline)
- A= Min Axial Length (Axial space)
- Fig= Position of legs
- Deg°= Degrees deflection



TORSION SPRINGS - STAINLESS STEEL

Catalog Number	d Wire Diameter		OD Outside Diameter		Fig. Position of Ends Fig.	Deflection Degree	T Approximate Torque Reference		Test Point 1/2 E		M Suggested Mandrel Size		E Leg Length		A Min. Axial Space**	
	in	mm	in	mm			in-lb	N-mm	in	mm	in	mm	in	mm	in	mm
T012-090-055			0.093	2.36	1	90					0.055	1.40			0.054	1.37
T012-180-067			0.109	2.77	2	180			0.187	4.75	0.067	1.70	0.375	9.52	0.090	2.29
T012-270-062	0.012	0.30	0.102	2.59	3	270					0.062	1.57			0.139	3.53
T012-180-109			0.166	4.22	2	180	0.047	5.31			0.109	2.77			0.066	1.68
T012-270-109			0.170	4.32	3	270					0.109	2.77			0.090	2.29
T012-360-109			0.174	4.42	4	360			0.250	6.35	0.109	2.77	0.500	12.70	0.114	2.90
T014-090-063			0.124	3.15	1	90					0.063	1.60			0.067	1.70
T014-180-078			0.133	3.38	2	180					0.078	1.98			0.105	2.67
T014-270-063	0.014	0.36	0.124	3.15	3	270	0.070	7.91			0.063	1.60			0.161	4.09
T014-180-109			0.194	4.93	2	180					0.109	2.77			0.077	1.96
T014-270-125			0.201	5.11	3	270			0.375	9.52	0.125	3.18	0.750	19.05	0.105	2.67
T014-360-125			0.204	5.18	4	360					0.125	3.18			0.133	3.38
T015-090-062			0.110	2.79	1	90					0.062	1.57			0.068	1.71
T015-180-078			0.130	3.30	2	180			0.250	6.35	0.078	1.98	0.500	12.70	0.113	2.86
T015-270-078	0.015	0.38	0.124	3.15	3	270	0.093	10.51			0.078	1.98			0.173	4.39
T015-180-109			0.183	4.65	2	180					0.109	2.77			0.083	2.10
T015-270-109			0.199	5.05	3	270			0.375	9.52	0.109	2.77	0.750	19.05	0.113	2.86
T015-360-109			0.207	5.26	4	360					0.109	2.77			0.143	3.62
T017-090-093			0.160	4.06	1	90					0.093	2.36			0.081	2.06
T017-180-093			0.172	4.37	2	180			0.250	6.35	0.093	2.36	0.500	12.70	0.128	3.24
T017-270-093	0.017	0.43	0.160	4.06	3	270	0.117	13.22			0.093	2.36			0.196	4.97
T017-180-156			0.249	6.32	2	180					0.156	3.96			0.094	2.39
T017-270-156			0.259	6.58	3	270			0.375	9.52	0.156	3.96	0.750	19.05	0.128	3.24
T017-360-140			0.235	5.97	4	360					0.140	3.56			0.170	4.32
T018-090-109			0.177	4.50	1	90					0.109	2.77			0.081	2.06
T018-180-109			0.164	4.17	2	180			0.250	6.35	0.109	2.77	0.500	12.70	0.153	3.89
T018-270-109			0.160	4.06	3	270					0.109	2.77			0.228	5.79
T018-180-140	0.018	0.46	0.216	5.49	2	180	0.140	15.82			0.140	3.56			0.117	2.97
T018-270-156			0.245	6.22	3	270					0.156	3.96			0.150	3.81
T018-360-156			0.233	5.92	4	360			0.375	9.52	0.156	3.96	0.750	19.05	0.210	5.33
T020-090-109			0.191	4.85	1	90					0.109	2.77			0.095	2.41
T020-180-109			0.179	4.55	2	180					0.109	2.77			0.170	4.32
T020-270-093	0.020	0.51	0.175	4.45	3	270	0.187	21.10			0.093	2.36			0.250	6.35
T020-180-140			0.242	6.15	2	180					0.140	3.56			0.130	3.30
T020-270-172			0.268	6.81	3	270			0.500	12.70	0.172	4.37	1.000	25.40	0.165	4.19
T020-360-156			0.254	6.45	4	360					0.156	3.96			0.250	6.35
T021-090-109			0.186	4.72	1	90					0.109	2.77			0.095	2.40
T021-180-109			0.185	4.70	2	180			0.375	9.52	0.109	2.77	0.750	19.05	0.179	4.53
T021-270-109	0.021	0.53	0.184	4.67	3	270	0.218	24.60			0.109	2.77			0.266	6.76
T021-180-156			0.247	6.27	2	180					0.156	3.96			0.137	3.47
T021-270-187			0.283	7.19	3	270			0.500	12.70	0.187	4.75	1.000	25.40	0.179	4.53
T021-360-187			0.271	6.88	4	360					0.187	4.75			0.242	6.15
T023-090-109			0.204	5.18	1	90					0.109	2.77			0.109	2.77
T023-180-109			0.191	4.85	2	180			0.375	9.52	0.109	2.77	0.750	19.05	0.196	4.98
T023-270-109	0.023	0.58	0.187	4.75	3	270	0.308	34.80			0.109	2.77			0.288	7.30
T023-180-156			0.259	6.58	2	180					0.156	3.96			0.150	3.81
T023-270-156			0.251	6.38	3	270			0.500	12.70	0.156	3.96	1.000	25.40	0.219	5.55
T023-360-172			0.271	6.88	4	360					0.172	4.37			0.265	6.72
T025-090-140			0.235	5.97	1	90					0.140	3.56			0.113	2.86
T025-180-140			0.224	5.69	2	180			0.375	9.52	0.140	3.56	0.750	19.05	0.213	5.40
T025-270-140	0.025	0.64	0.219	5.56	3	270	0.375	42.40			0.140	3.56			0.316	8.03
T025-180-203			0.304	7.72	2	180					0.203	5.16			0.163	4.13
T025-270-218			0.340	8.64	3	270					0.218	5.54			0.213	5.40
T025-360-218			0.324	8.23	4	360					0.218	5.54			0.288	7.32
T028-090-156			0.267	6.78	1	90					0.156	3.96			0.133	3.38
T028-180-140			0.249	6.32	2	180			0.500	12.70	0.140	3.56	1.000	25.40	0.238	6.05
T028-270-140	0.028	0.71	0.245	6.22	3	270	0.515	58.20			0.140	3.56			0.350	8.89
T028-180-203			0.340	8.64	2	180					0.203	5.16			0.182	4.62
T028-270-203			0.329	8.36	3	270					0.203	5.16			0.266	6.76
T028-360-218			0.355	9.02	4	360					0.218	5.54			0.350	8.89

TO ORDER: WWW.ASRAYMOND.COM OR SEE PHONE & FAX NUMBERS ON BACK COVER



** Space needed on application to allow for operation of the springs.
This dimension does not refer to the length of the coils.

TORSION SPRINGS - STAINLESS STEEL

Catalog Number	d Wire Diameter		OD Outside Diameter		Fig. Position of Ends Fig.	Deflection Degree	T Approximate Torque Reference		Test Point 1/2 E		M Suggested Mandrel Size		E Leg Length		A Min. Axial Space**	
	in	mm	in	mm			in-lb	N-m	in	mm	in	mm	in	mm	in	mm
T030-090-172			0.281	7.14	1	90					0.172	4.37			0.135	3.43
T030-180-172			0.272	6.91	2	180					0.172	4.37			0.255	6.48
T030-270-172	0.030	0.76	0.270	6.86	3	270	0.625	70.60			0.172	4.37			0.380	9.65
T030-180-250			0.394	10.01	2	180					0.250	6.35			0.195	4.95
T030-270-250			0.377	9.58	3	270					0.250	6.35			0.280	7.11
T030-360-250			0.410	10.41	4	360			0.500	12.70	0.250	6.35	1.000	25.40	0.346	8.79
T032-090-172			0.288	7.32	1	90					0.172	4.36			0.152	3.86
T032-180-156			0.270	6.86	2	180					0.156	3.96			0.272	6.91
T032-270-156	0.032	0.81	0.264	6.71	3	270	0.820	92.70			0.156	3.96			0.432	10.97
T032-180-218			0.366	9.30	2	180					0.218	5.54			0.208	5.28
T032-270-218			0.354	8.99	3	270					0.218	5.54			0.304	7.72
T032-360-234			0.382	9.70	4	360					0.234	5.95			0.368	9.35
T035-090-187			0.315	8.00	1	90					0.187	4.75			0.158	4.00
T035-180-187			0.303	7.70	2	180					0.187	4.75			0.298	7.56
T035-270-187	0.035	0.89	0.311	7.90	3	270	1.000	113.00			0.187	4.75			0.442	11.23
T035-180-281			0.450	11.43	2	180					0.281	7.14			0.228	5.78
T035-270-281			0.435	11.05	3	270					0.281	7.14			0.333	8.45
T035-360-312			0.471	11.96	4	360					0.312	7.92			0.405	10.29
T038-090-234			0.386	9.80	1	90					0.234	5.94			0.180	4.57
T038-180-218			0.368	9.35	2	180			0.625	15.88	0.218	5.54	1.250	31.75	0.323	8.20
T038-270-218	0.038	0.97	0.353	8.97	3	270	1.190	134.50			0.218	5.54			0.475	12.07
T038-180-312			0.487	12.37	2	180					0.318	8.08			0.247	6.27
T038-270-312			0.477	12.12	3	270					0.312	7.92			0.361	9.17
T038-360-328			0.514	13.06	4	360					0.328	8.33			0.475	12.07
T040-090-187			0.309	7.85	1	90					0.187	4.75			0.220	5.59
T040-180-218			0.348	8.84	2	180					0.218	5.54			0.380	9.65
T040-270-218	0.040	1.02	0.358	9.09	3	270	1.375	155.40			0.218	5.54			0.550	13.97
T040-180-343			0.518	13.16	2	180					0.343	8.71			0.260	6.60
T040-270-343			0.511	12.98	3	270			1.000	25.40	0.343	8.71	2.000	50.80	0.380	9.65
T040-360-343			0.507	12.88	4	360					0.343	8.71			0.508	12.90
T045-090-203			0.357	9.07	1	90					0.203	5.16			0.259	6.58
T045-180-218			0.377	9.58	2	180			0.625	15.88	0.218	5.54	1.250	31.75	0.428	10.86
T045-270-234	0.045	1.14	0.382	9.70	3	270	2.000	226.00			0.234	5.94			0.608	15.43
T045-180-359			0.575	14.61	2	180					0.359	9.12			0.293	7.44
T045-270-359			0.556	14.12	3	270			1.000	25.40	0.359	9.12	2.000	50.80	0.428	10.86
T045-360-359			0.549	13.94	4	360					0.359	9.12			0.563	14.29
T048-090-218			0.375	9.53	1	90					0.218	5.54			0.264	6.71
T048-180-250			0.404	10.26	2	180			0.625	15.88	0.250	6.35	1.250	31.75	0.456	11.58
T048-270-250	0.048	1.22	0.416	10.57	3	270	2.500	282.00			0.250	6.35			0.660	16.76
T048-180-406			0.618	15.70	2	180					0.406	10.31			0.312	7.92
T048-270-406			0.600	15.24	3	270					0.406	10.31			0.456	11.58
T048-360-406			0.594	15.09	4	360					0.406	10.31			0.610	15.49
T051-090-234			0.408	10.36	1	90					0.234	5.94			0.293	7.44
T051-180-250			0.430	10.92	2	180					0.250	6.35			0.485	12.32
T051-270-266	0.051	1.30	0.439	11.15	3	270	2.900	328.00			0.266	6.76			0.689	17.49
T051-180-344			0.556	14.12	2	180					0.344	8.74			0.383	9.72
T051-270-359			0.571	14.50	3	270					0.359	9.12			0.536	13.60
T051-360-406			0.628	15.95	4	360					0.406	10.31			0.638	16.19
T054-090-296			0.484	12.29	1	90					0.296	7.52			0.310	7.87
T054-180-312			0.509	12.93	2	180			1.000	25.40	0.312	7.92	2.000	50.80	0.512	13.00
T054-270-312	0.054	1.37	0.514	13.06	3	270	3.275	370.00			0.312	7.92			0.715	18.16
T054-180-421			0.654	16.61	2	180					0.421	10.69			0.405	10.29
T054-270-437			0.664	16.87	3	270					0.437	11.10			0.567	14.40
T054-360-453			0.694	17.63	4	360					0.453	11.51			0.705	17.91
T059-090-296			0.499	12.67	1	90					0.296	7.52			0.340	8.64
T059-180-328			0.526	13.36	2	180					0.328	8.33			0.561	14.24
T059-270-328	0.059	1.50	0.537	13.64	3	270	4.200	475.00			0.328	8.33			0.797	20.23
T059-180-437			0.681	17.30	2	180					0.437	11.10			0.445	11.30
T059-270-453			0.699	17.75	3	270					0.453	11.51			0.620	15.74
T059-360-459			0.709	18.01	4	360					0.456	11.66			0.797	20.23



TORSION SPRINGS - STAINLESS STEEL

Catalog Number	d Wire Diameter		OD Outside Diameter		Fig. Position of Ends	Deflection Degree	T Approximate Torque Reference		Test Point 1/2 E		M Suggested Mandrel Size		E Leg Length		A Min. Axial Space**	
	in	mm	in	mm			Fig.	in-lb	N-mm	in	mm	in	mm	in	mm	in
T063-090-343			0.560	14.22	1	90					0.343	8.71			0.362	9.19
T063-180-359			0.591	15.01	2	180					0.359	9.12			0.600	15.24
T063-270-375	0.063	1.60	0.600	15.24	3	270	5.150	582.00			0.375	9.53			0.851	21.60
T063-180-500			0.767	19.48	2	180					0.500	12.70			0.475	12.07
T063-270-516			0.784	19.91	3	270					0.516	13.11			0.662	16.80
T063-360-516			0.798	20.27	4	360					0.516	13.11			0.851	21.60
T070-090-359			0.593	15.06	1	90					0.359	9.12			0.400	10.16
T070-180-390			0.625	15.88	2	180					0.390	9.91			0.665	16.89
T070-270-390	0.070	1.78	0.639	16.23	3	270	7.000	791.00			0.390	9.91			0.945	24.00
T070-180-515			0.810	20.57	2	180					0.515	13.08			0.525	13.34
T070-270-531			0.826	20.98	3	270					0.531	13.49			0.735	18.67
T070-360-546			0.843	21.41	4	360			1.000	25.40	0.546	13.87	2.000	50.80	0.945	24.00
T075-090-375			0.635	16.13	1	90					0.375	9.53			0.430	10.92
T075-180-422			0.675	17.15	2	180					0.422	10.72			0.713	18.10
T075-270-500	0.075	1.91	0.700	17.78	3	270	8.750	989.00			0.440	12.70			1.013	25.72
T075-180-484			0.775	19.69	2	180					0.484	12.29			0.640	16.26
T075-270-531			0.825	20.96	3	270					0.531	13.49			0.863	21.91
T075-360-640			0.975	24.77	4	360					0.640	16.26			1.050	26.67
T078-090-406			0.678	17.22	1	90					0.406	10.31			0.450	11.43
T078-180-453			0.728	18.49	2	180					0.453	11.51			0.741	18.82
T078-270-453	0.078	1.98	0.728	18.49	3	270	9.750	1102.00			0.453	11.51			1.030	26.16
T078-180-500			0.803	20.40	2	180					0.500	12.70			0.663	16.84
T078-270-546			0.853	21.67	3	270					0.546	13.87			0.897	22.78
T078-360-578			0.903	22.94	4	360					0.578	14.68			1.131	28.73
T085-090-422			0.715	18.16	1	90					0.422	10.72			0.489	12.42
T085-180-469			0.760	19.30	2	180					0.469	11.91			0.808	20.52
T085-270-500	0.085	2.16	0.790	20.07	3	270	12.000	1356.00	1.250	31.75	0.500	12.70	2.500	63.50	1.148	29.15
T085-180-641			0.994	25.25	2	180					0.641	16.28			0.638	16.21
T085-270-672			1.040	26.42	3	270					0.672	17.07			0.893	22.67
T085-360-688			1.054	26.77	4	360					0.688	17.48			1.148	29.15
T095-090-453			0.776	19.71	1	90					0.453	11.51			0.546	13.87
T095-180-531			0.869	22.07	2	180					0.531	13.49			0.903	22.94
T095-270-578	0.095	2.41	0.925	23.50	3	270	16.000	1808.00	1.500	38.10	0.578	14.68	3.000	76.20	1.283	32.58
T095-180-734			1.131	28.73	2	180					0.734	18.64			0.713	18.11
T095-270-797			1.203	30.56	3	270					0.797	20.24			0.998	25.34
T095-360-813			1.225	31.12	4	360					0.813	20.65			1.283	32.58
T105-090-500			0.848	21.54	1	90					0.500	12.70			0.604	15.34
T105-180-609			0.982	24.94	2	180					0.609	15.47			0.998	25.35
T105-270-703	0.105	2.67	1.090	27.69	3	270	21.000	2373.00	1.750	44.45	0.703	17.86	3.500	88.90	1.418	36.00
T105-180-813			1.248	31.70	2	180					0.813	20.65			0.788	20.02
T105-270-891			1.342	34.09	3	270					0.891	22.63			1.103	28.00
T105-360-906			1.369	34.77	4	360					0.906	23.01			1.418	36.00
T115-090-594			0.978	24.84	1	90					0.594	15.09			0.661	16.79
T115-180-641			1.043	26.49	2	180					0.641	16.28			1.093	27.76
T115-270-688	0.115	2.92	1.086	27.58	3	270	28.000	3164.00			0.688	17.48			1.553	39.43
T115-180-859			1.347	34.21	2	180					0.859	21.82			0.863	21.92
T115-270-938			1.435	36.45	3	270					0.938	23.83			1.208	30.67
T115-360-969			1.465	37.21	4	360					0.969	24.61			1.553	39.43
T125-090-591			0.989	25.12	1	90					0.591	15.01			0.844	21.44
T125-180-666			1.082	27.48	2	180					0.666	16.92			1.438	36.53
T125-270-751	0.125	3.18	1.189	30.20	3	270	32.000	3616.00	2.000	50.80	0.751	19.08	4.000	101.60	1.938	49.21
T125-180-885			1.356	34.44	2	180					0.885	22.48			1.188	30.18
T125-270-1013			1.516	38.51	3	270					1.013	25.73			1.563	39.69
T125-360-1084			1.605	40.77	4	360					1.084	27.53			1.938	49.21
T135-090-666			1.102	27.99	1	90					0.666	16.92			0.911	23.14
T135-180-735			1.189	30.20	2	180					0.735	18.67			1.553	39.45
T135-270-825	0.135	3.43	1.301	33.05	3	270	40.000	4519.00			0.825	20.96			2.093	53.15
T135-180-977			1.491	37.87	2	180					0.977	24.82			1.283	32.59
T135-270-1112			1.660	42.16	3	270					1.112	28.24			1.688	42.86
T135-360-1188			1.755	44.58	4	360					1.188	30.18			2.093	53.15

TO ORDER: WWW.ASRAYMOND.COM OR SEE PHONE & FAX NUMBERS ON BACK COVER



** Space needed on application to allow for operation of the springs.
This dimension does not refer to the length of the coils.

D.2 Composantes électroniques

Dans cette section sont présentées les composantes électroniques qui constituent le prototype final d'Hexapode.

- Latch National semiconductor MM74HC573
- Force Sensing Resistor Model 400
- Schéma électronique du Servopod de Newmicros
- Microcontrôleur Motorola DSP56F807

MM74HC573 3-STATE Octal D-Type Latch

General Description

The MM74HC573 high speed octal D-type latches utilize advanced silicon-gate P-well CMOS technology. They possess the high noise immunity and low power consumption of standard CMOS integrated circuits, as well as the ability to drive 15 LS-TTL loads. Due to the large output drive capability and the 3-STATE feature, these devices are ideally suited for interfacing with bus lines in a bus organized system.

When the LATCH ENABLE (LE) input is HIGH, the Q outputs will follow the D inputs. When the LATCH ENABLE goes LOW, data at the D inputs will be retained at the outputs until LATCH ENABLE returns HIGH again. When a HIGH logic level is applied to the OUTPUT CONTROL OC input, all outputs go to a HIGH impedance state, regardless

of what signals are present at the other inputs and the state of the storage elements.

The 74HC logic family is speed, function and pinout compatible with the standard 74LS logic family. All inputs are protected from damage due to static discharge by internal diode clamps to V_{CC} and ground.

Features

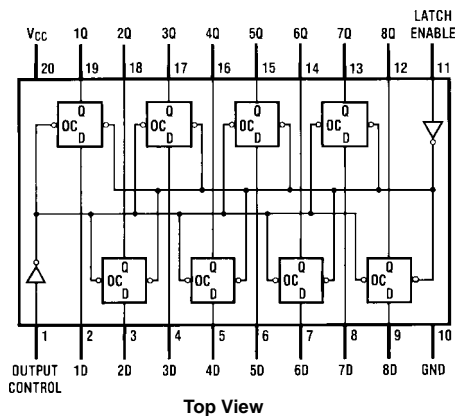
- Typical propagation delay: 18 ns
- Wide operating voltage range: 2 to 6 volts
- Low input current: 1 μ A maximum
- Low quiescent current: 80 μ A maximum (74HC Series)
- Compatible with bus-oriented systems
- Output drive capability: 15 LS-TTL loads

Ordering Code:

Order Number	Package Number	Package Description
MM74HC573WM	M20B	20-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-013, 0.300 Wide
MM74HC573SJ	M20D	20-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
MM74HC573MTC	MTC20	20-Lead Thin Shrink Small Outline Package (TSSOP), JEDEC MO-153, 4.4mm Wide
MM74HC573N	N20A	20-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

Connection Diagram



Truth Table

Output Control	Latch Enable	Data	Output
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

H = HIGH Level
L = LOW Level
 Q_0 = Level of output before steady-state input conditions were established.
Z = High Impedance
X = Don't Care

Absolute Maximum Ratings (Note 1)

(Note 2)

Supply Voltage (V_{CC})	-0.5 to +7.0V
DC Input Voltage (V_{IN})	-1.5 to $V_{CC}+1.5V$
DC Output Voltage (V_{OUT})	-0.5 to $V_{CC}+0.5V$
Clamp Diode Current (I_{IK}, I_{OK})	± 20 mA
DC Output Current, per pin (I_{OUT})	± 35 mA
DC V_{CC} or GND Current, per pin (I_{CC})	± 70 mA
Storage Temperature Range (T_{STG})	-65°C to +150°C
Power Dissipation (P_D)	
(Note 3)	600 mW
S.O. Package only	500 mW
Lead Temperature (T_L)	
(Soldering 10 seconds)	260°C

Recommended Operating Conditions

	Min	Max	Units
Supply Voltage (V_{CC})	2	6	V
DC Input or Output Voltage (V_{IN}, V_{OUT})	0	V_{CC}	V
Operating Temperature Range (T_A)	-40	+85	°C
Input Rise or Fall Times (t_r, t_f)			
$V_{CC} = 2.0V$		1000	ns
$V_{CC} = 4.5V$		500	ns
$V_{CC} = 6.0V$		400	ns

Note 1: Absolute Maximum Ratings are those values beyond which damage to the device may occur.

Note 2: Unless otherwise specified all voltages are referenced to ground.

Note 3: Power Dissipation temperature derating — plastic "N" package: -12 mW/°C from 65°C to 85°C.

DC Electrical Characteristics (Note 4)

Symbol	Parameter	Conditions	V_{CC}	$T_A = 25^\circ C$			$T_A = -40$ to $85^\circ C$	$T_A = -55$ to $125^\circ C$	Units
				Typ	Guaranteed Limits				
V_{IH}	Minimum HIGH Level Input Voltage		2.0V		1.5	1.5	1.5	V	
			4.5V		3.15	3.15	3.15	V	
			6.0V		4.2	4.2	4.2	V	
V_{IL}	Maximum LOW Level Input Voltage		2.0V		0.5	0.5	0.5	V	
			4.5V		1.35	1.35	1.35	V	
			6.0V		1.8	1.8	1.8	V	
V_{OH}	Minimum HIGH Level Output Voltage	$V_{IN} = V_{IH}$ or V_{IL} $ I_{OUT} \leq 20 \mu A$	2.0V	2.0	1.9	1.9	1.9	V	
			4.5V	4.5	4.4	4.4	4.4	V	
			6.0V	6.0	5.9	5.9	5.9	V	
		$V_{IN} = V_{IH}$ or V_{IL} $ I_{OUT} \leq 6.0$ mA $ I_{OUT} \leq 7.8$ mA	4.5V	4.2	3.98	3.84	3.7	V	
			6.0V	5.7	5.48	5.34	5.2	V	
V_{OL}	Maximum LOW Level Output Voltage	$V_{IN} = V_{IH}$ or V_{IL} $ I_{OUT} \leq 20 \mu A$	2.0V	0	0.1	0.1	0.1	V	
			4.5V	0	0.1	0.1	0.1	V	
			6.0V	0	0.1	0.1	0.1	V	
		$V_{IN} = V_{IH}$ or V_{IL} $ I_{OUT} \leq 6.0$ mA $ I_{OUT} \leq 7.8$ mA	4.5V	0.2	0.26	0.33	0.4	V	
			6.0V	0.2	0.26	0.33	0.4	V	
I_{IN}	Maximum Input Current	$V_{IN} = V_{CC}$ or GND	6.0V		± 0.1	± 1.0	± 1.0	μA	
I_{OZ}	Maximum 3-STATE Output Leakage Current	$V_{OUT} = V_{CC}$ or GND $OC = V_{IH}$	6.0V		± 0.5	± 5.0	± 10	μA	
I_{CC}	Maximum Quiescent Supply Current	$V_{IN} = V_{CC}$ or GND $I_{OUT} = 0 \mu A$	6.0V		8.0	80	160	μA	
ΔI_{CC}	Quiescent Supply Current per Input Pin	$V_{CC} = 5.5V$ $V_{IN} = 2.4V$ or 0.4V (Note 4)	OE	1.0	1.5	1.8	2.0	mA	
			LE	0.6	0.8	1.0	1.1	mA	
			DATA	0.4	0.5	0.6	0.7	mA	

Note 4: For a power supply of 5V $\pm 10\%$ the worst-case output voltages (V_{OH} and V_{OL}) occur for HC at 4.5V. Thus the 4.5V values should be used when designing with this supply. Worst-case V_{IH} and V_{IL} occur at $V_{CC} = 5.5V$ and 4.5V respectively. (The V_{IH} value at 5.5V is 3.85V.) The worst-case leakage current (I_{IN} , I_{CC} , and I_{OZ}) occur for CMOS at the higher voltage and so the 6.0V values should be used.

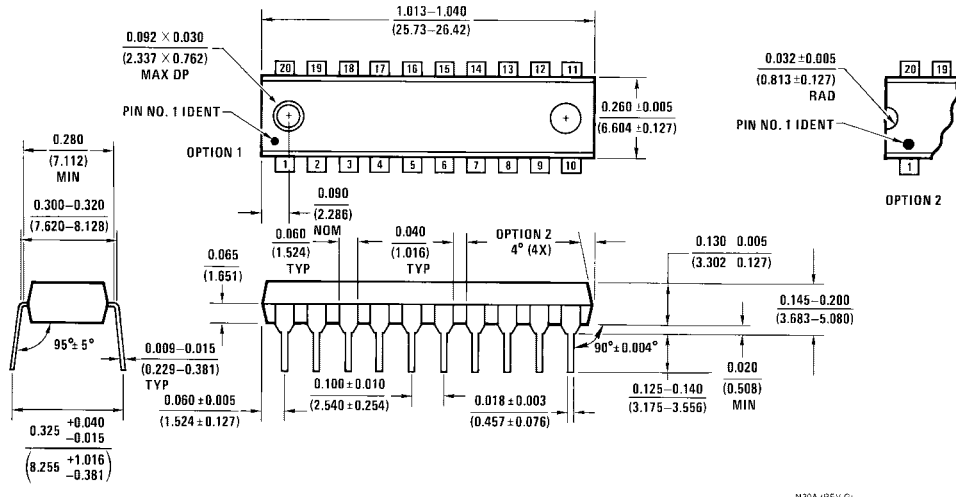
AC Electrical Characteristics								
$V_{CC} = 5V, T_A = 25^\circ C, t_r = t_f = 6 ns$								
Symbol	Parameter	Conditions	Typ	Guaranteed Limit	Units			
t_{PHL}, t_{PLH}	Maximum Propagation Delay, Data to Q	$C_L = 45 pF$	16	20	ns			
t_{PHL}, t_{PLH}	Maximum Propagation Delay, LE to Q	$C_L = 45 pF$	14	22	ns			
t_{PZH}, t_{PZL}	Maximum Output Enable Time	$R_L = 1 k\Omega$ $C_L = 45 pF$	15	27	ns			
t_{PHZ}, t_{PLZ}	Maximum Output Disable Time	$R_L = 1 k\Omega$ $C_L = 5 pF$	13	23	ns			
t_S	Minimum Set Up Time, Data to LE		10	15	ns			
t_H	Minimum Hold Time, LE to Data		2	5	ns			
t_W	Minimum Pulse Width, LE or Data		10	16	ns			
AC Electrical Characteristics								
Symbol	Parameter	Conditions	V_{CC}	$T_A = 25^\circ C$		$T_A = -40 \text{ to } 85^\circ C$	$T_A = -55 \text{ to } 125^\circ C$	Units
				Typ	Guaranteed Limits			
t_{PHL}, t_{PLH}	Maximum Propagation Delay Data to Q	$C_L = 50 pF$	2.0V	45	110	138	165	ns
			2.0V	58	150	188	225	ns
		$C_L = 150 pF$	4.5V	17	22	28	33	ns
			4.5V	21	30	38	40	ns
		6.0V	15	19	24	29	ns	
t_{PHL}, t_{PLH}	Maximum Propagation Delay, LE to Q	$C_L = 50 pF$	2.0V	46	115	138	165	ns
			2.0V	60	155	194	233	ns
		$C_L = 150 pF$	4.5V	14	23	29	35	ns
			4.5V	21	31	47	47	ns
		6.0V	12	20	25	30	ns	
6.0V	19	27	34	41	ns			
t_{PZH}, t_{PZL}	Maximum Output Enable Time	$R_L = 1 k\Omega$	2.0V	55	140	175	210	ns
		$C_L = 150 pF$	2.0V	67	180	225	270	ns
			4.5V	15	28	35	42	ns
		4.5V	24	36	45	54	ns	
t_{PHZ}, t_{PLZ}	Maximum Output Disable Time	$R_L = 1 k\Omega$	2.0V	40	125	156	188	ns
		$C_L = 150 pF$	4.5V	13	25	31	38	ns
			6.0V	12	21	27	32	ns
		6.0V	9	13	16	19	ns	
t_S	Minimum Set Up Time Data to LE		2.0V	30	75	95	110	ns
			4.5V	10	15	19	22	ns
			6.0V	9	13	16	19	ns
t_H	Minimum Hold Time LE to Data		2.0V		25	31	38	ns
			4.5V		5	6	7	ns
			6.0V		4	5	6	ns
t_W	Minimum Pulse Width LE, or Data		2.0V	30	80	100	120	ns
			4.5V	9	16	20	24	ns
			6.0V	8	14	18	20	ns
t_{TLH}, t_{THL}	Maximum Output Rise and Fall Time, Clock	$C_L = 50 pF$	2.0V	25	60	75	90	ns
			4.5V	7	12	15	18	ns
			6.0V	6	10	13	15	ns
C_{PD}	Power Dissipation Capacitance (Note 5) (per latch)	OC = V_{CC}		5				pF
		OC = GND		52				pF
C_{IN}	Maximum Input Capacitance			5	10	10	10	pF

AC Electrical Characteristics (Continued)

Symbol	Parameter	Conditions	V _{CC}	T _A = 25°C		T _A = -40 to 85°C	T _A = -55 to 125°C	Units
				Typ	Guaranteed Limits			
C _{OUT}	Maximum Output Capacitance			15	20	20	20	pF

Note 5: C_{PD} determines the no load dynamic power consumption, $P_D = C_{PD} V_{CC}^2 f + I_{CC} V_{CC}$, and the no load dynamic current consumption, $I_S = C_{PD} V_{CC} f + I_{CC}$.

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



**20-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide
Package Number N20A**

Fairchild does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and Fairchild reserves the right at any time without notice to change said circuitry and specifications.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

www.fairchildsemi.com

FSR[®] Integration Guide & Evaluation Parts Catalog With Suggested Electrical Interfaces

Force Sensing Resistors [®] – An Overview of the Technology	Page 3
Force vs. Resistance.....	Page 3
Force vs. Conductance.....	Page 4
FSR Integration Notes – A Step-by-Step Guide to Optimal Use	Page 6
FSR Usage Tips – The Do’s and Don’ts	Page 8
Evaluation Parts Catalog – Descriptions and Dimensions	Page 9
General FSR Characteristics	Page 12
Simple FSR Devices and Arrays.....	Page 12
For Linear Pots	Page 13
Glossary of Terms	Page 14
Suggested Electrical Interfaces - Basic FSRs	Page 16
FSR Voltage Divider	Page 16
Adjustable Buffers	Page 17
Multi-channel FSR to Digital Interface	Page 18
FSR Variable Force Threshold Switch	Page 19
FSR Variable Force Threshold Relay Switch	Page 20
FSR Current-to-Voltage Converter	Page 21
Additional FSR Current-to-Voltage Converters	Page 22
FSR Schmitt Trigger Oscillator	Page 23

Interlink Electronics manufactures custom FSR devices to meet the needs of specific customer applications. FSR devices can be produced in almost any shape, size, and geometry. To discuss custom design or to obtain a quote, contact Interlink Electronics at (805) 484-8855.

Force Sensing Resistors

An Overview of the Technology

Force Sensing Resistors (FSR) are a polymer thick film (PTF) device which exhibits a decrease in resistance with an increase in the force applied to the active surface. Its force sensitivity is optimized for use in human touch control of electronic devices. FSRs are not a load cell or strain gauge, though they have similar properties. FSRs are not suitable for precision measurements.

Force vs. Resistance

The force vs. resistance characteristic shown in Figure 2 provides an overview of FSR typical response behavior. For interpretational convenience, the force vs. resistance data is plotted on a log/log format. These data are representative of our typical devices, with this particular force-resistance characteristic being the response of evaluation part # 402 (0.5" [12.7 mm] diameter circular active area). A stainless steel actuator with a 0.4" [10.0 mm] diameter hemispherical tip of 60 durometer polyurethane rubber was used to actuate the FSR device. In general, FSR response approximately follows an inverse power-law characteristic (roughly 1/R).

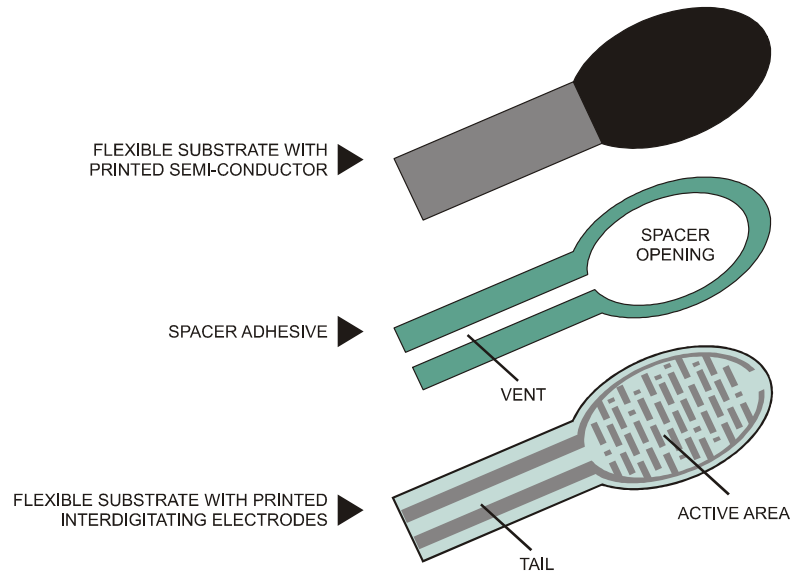


Figure 1: FSR Construction

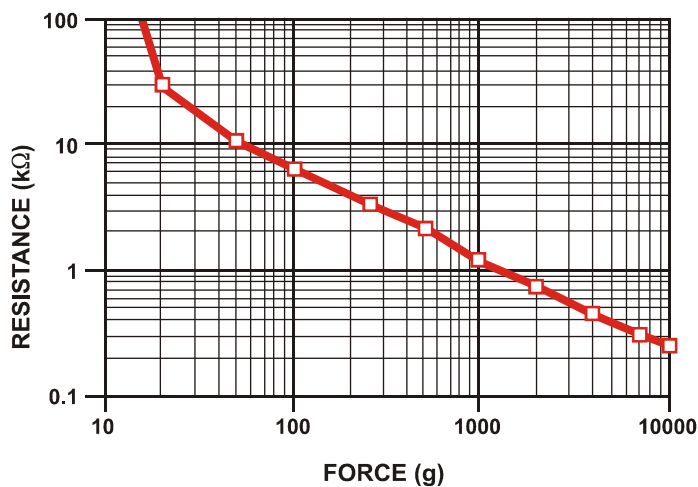


Figure 2: Resistance vs. Force

Referring to Figure 2, at the low force end of the force-resistance characteristic, a switch-like response is evident. This turn-on threshold, or ‘break force’, that swings the resistance from greater than 100 kΩ to about 10 kΩ (the beginning of the dynamic range that follows a power-law) is determined by the substrate and overlay thickness and flexibility, size and shape of the actuator, and spacer-adhesive thickness (the gap between the facing conductive elements). Break force increases with increasing substrate and overlay rigidity, actuator size, and spacer-adhesive thickness. Eliminating the adhesive, or keeping it well away from the area where the force is being applied, such as the center of a large FSR device, will give it a lower rest resistance (e.g. stand-off resistance).

At the high force end of the dynamic range, the response deviates from the power-law behavior, and eventually saturates to a point where increases in force yield little or no decrease in resistance. Under these conditions of Figure 2, this saturation force is beyond 10 kg. The saturation point is more a function of pressure than force. The saturation pressure of a typical FSR is on the order of 100 to 200 psi. For the data shown in Figures 2, 3 and 4, the actual measured pressure range is 0 to 175 psi (0 to 22 lbs applied over 0.125 in²). Forces higher than the saturation force can be measured by spreading the force over a greater area; the overall pressure is then kept below the saturation point, and dynamic response is maintained. However, the converse of this effect is also true, smaller actuators will saturate FSRs earlier in the dynamic range, since the saturation point is reached at a lower force.

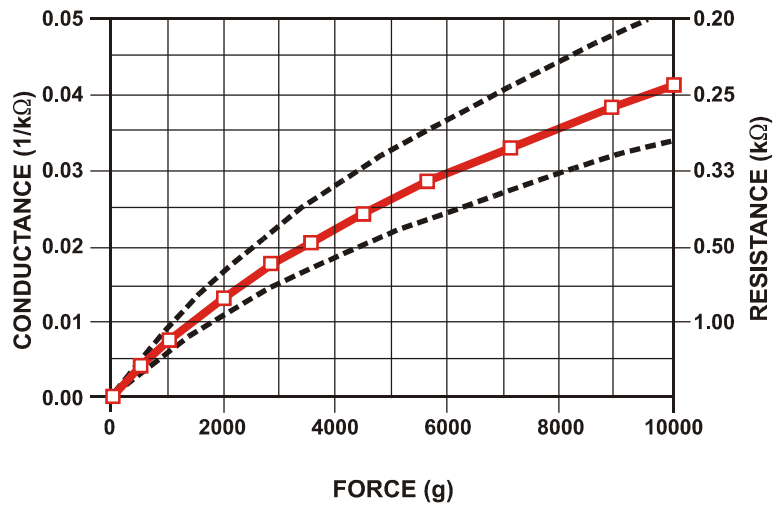


Figure 3:
Conductance vs. Force (0-10Kg)

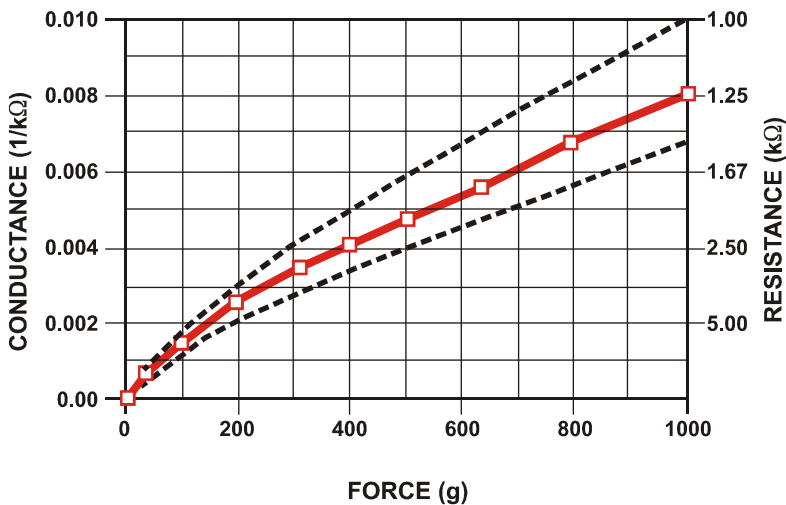


Figure 4:
Conductance vs. Force (0-1Kg) Low Force Range

Force vs. Conductance

In Figure 3, the conductance is plotted vs. force (the inverse of resistance: $1/r$). This format allows interpretation on a linear scale. For reference, the corresponding resistance values are also included on the right vertical axis. A simple circuit called a current-to-voltage converter (see page 21) gives a voltage output directly proportional to FSR conductance and can be useful where response linearity is desired. Figure 3 also includes a typical part-to-part repeatability envelope. This error band determines the maximum accuracy of any general force measurement. The spread or width of the band is

strongly dependent on the repeatability of any actuating and measuring system, as well as the repeatability tolerance held by Interlink Electronics during FSR production. Typically, the part-to-part repeatability tolerance held during manufacturing ranges from $\pm 15\%$ to $\pm 25\%$ of an established nominal resistance.

Figure 4 highlights the 0-1 kg (0-2.2 lbs) range of the conductance-force characteristic. As in Figure 3, the corresponding resistance values are included for reference. This range is common to human interface applications. Since the conductance response in this range is fairly linear, the force resolution will be uniform and data interpretation simplified. The typical part-to-part error band is also shown for this touch range. In most human touch control applications this error is insignificant, since human touch is fairly inaccurate. Human factors studies have shown that in this force range repeatability errors of less than $\pm 50\%$ are difficult to discern by touch alone.

FSR Integration Notes

A Step-by-Step Guide to Optimal Use

For best results, follow these seven steps when beginning any new product design, proof-of-concept, technology evaluation, or first prototype implementation:

1. Start with Reasonable Expectations (Know Your Sensor)

The FSR sensor is not a strain gauge, load cell or pressure transducer. While it can be used for dynamic measurement, only qualitative results are generally obtainable. Force accuracy ranges from approximately $\pm 5\%$ to $\pm 25\%$ depending on the consistency of the measurement and actuation system, the repeatability tolerance held in manufacturing, and the use of part calibration.

Accuracy should not be confused with resolution. The force resolution of FSR devices is better than $\pm 0.5\%$ of full use force.

2. Choose the Sensor that Best Fits the Geometry of Your Application

Usually sensor size and shape are the limiting parameters in FSR integration, so any evaluation part should be chosen to fit the desired mechanical actuation system. In general, standard FSR products have a common semiconductor make-up and only by varying actuation methods (e.g. overlays and actuator areas) or electrical interfaces can different response characteristics be achieved.

3. Set-up a Repeatable and Reproducible Mechanical Actuation System

When designing the actuation mechanics, follow these guidelines to achieve the best force repeatability:

- Provide a consistent force distribution. FSR response is very sensitive to the distribution of the applied force. In general, this precludes the use of dead weights for characterization since exact duplication of the weight distribution is rarely repeatable cycle-to-cycle. A consistent weight (force) distribution is more difficult to achieve than merely obtaining a consistent total applied weight (force). As long as the distribution is the same cycle-to-cycle, then repeatability will be maintained. The use of a thin elastomer between the applied force and the FSR can help absorb error from inconsistent force distributions.
- Keep the actuator area, shape, and compliance constant. Changes in these parameters significantly alter the response characteristic of a given sensor. Any test, mock-up, or evaluation conditions should be closely matched to the final use conditions. The greater the cycle-to-cycle consistency of these parameters, the greater the device repeatability. In human interface applications where a finger is the mode of actuation, perfect control of these parameters is not generally possible. However, human force sensing is somewhat inaccurate; it is rarely sensitive enough to detect differences of less than $\pm 50\%$.
- Control actuator placement. In cases where the actuator is to be smaller than the FSR active area, cycle-to-cycle consistency of actuator placement is necessary. (Caution: FSR layers are held together by an adhesive that surrounds the electrically active areas. If force is applied over an area which includes the adhesive, the resulting response characteristic will be drastically altered.) In an extreme case (e.g., a large, flat, hard actuator that bridges the bordering adhesive), the adhesive can present FSR actuation

- Keep actuation cycle time consistent. Because of the time dependence of the FSR resistance to an applied force, it is important when characterizing the sensor system to assure that increasing loads (e.g. force ramps) are applied at consistent rates (cycle-to-cycle). Likewise, static force measurements must take into account FSR mechanical setting time. This time is dependent on the mechanics of actuation and the amount of force applied and is usually on the order of seconds.

4. Use the Optimal Electronic Interface

In most product designs, the critical characteristic is Force vs. Output Voltage, which is controlled by the choice of interface electronics. A variety of interface solutions are detailed in the TechNote section of this guide. Summarized here are some suggested circuits for common FSR applications.

- For FSR Pressure or Force Switches, use the simple interfaces detailed on pages 16 and 17.
- For dynamic FSR measurements or Variable Controls, a current-to-voltage converter (see pages 18 and 19) is recommended. This circuit produces an output voltage that is inversely proportional to FSR resistance. Since the FSR resistance is roughly inversely proportional to applied force, the end result is a direct proportionality between force and voltage; in other words, this circuit gives roughly linear increases in output voltage for increases in applied force. This linearization of the response optimizes the resolution and simplifies data interpretation.

5. Develop a Nominal Voltage Curve and Error Spread

When a repeatable and reproducible system has been established, data from a group of FSR parts can be collected. Test several FSR parts in the system. Record the output voltage at various pre-selected force points throughout the range of interest. Once a family of curves is obtained, a nominal force vs. output voltage curve and the total force accuracy of the system can be determined.

6. Use Part Calibration if Greater Accuracy is Required

For applications requiring the highest obtainable force accuracy, part calibration will be necessary. Two methods can be utilized: gain and offset trimming, and curve fitting.

- Gain and offset trimming can be used as a simple method of calibration. The reference voltage and feedback resistor of the current-to-voltage converter are adjusted for each FSR to pull their responses closer to the nominal curve.
- Curve fitting is the most complete calibration method. A parametric curve fit is done for the nominal curve of a set of FSR devices, and the resultant equation is stored for future use. Fit parameters are then established for each individual FSR (or sending element in an array) in the set. These parameters, along with the measured sensor resistance (or voltage), are inserted into the equation to obtain the force reading. If needed, temperature compensation can also be included in the equation.

7. Refine the System

Spurious results can normally be traced to sensor error or system error. If you have any questions, contact Interlink Electronics' Sales Engineers to discuss your system and final data.

FSR Usage Tips *The Do's and Don'ts*

- **Do** follow the seven steps of the FSR Integration Guide.
- **Do**, if possible, use a firm, flat and smooth mounting surface.
- **Do** be careful if applying FSR devices to curved surfaces. Pre-loading of the device can occur as the two opposed layers are forced into contact by the bending tension. The device will still function, but the dynamic range may be reduced and resistance drift could occur. The degree of curvature over which an FSR can be bent is a function of the size of the active area. The smaller the active area, the less effect a given curvature will have on the FSR's response.
- **Do** avoid air bubbles and contamination when laminating the FSR to any surface. Use only thin, uniform adhesives, such as Scotch[®] brand double-sided laminating adhesives. Cover the entire surface of the sensor.
- **Do** be careful of kinks or dents in active areas. They can cause false triggering of the sensors.
- **Do** protect the device from sharp objects. Use an overlay, such as a polycarbonate film or an elastomer, to prevent gouging of the FSR device.
- **Do** use soft rubber or a spring as part of the actuator in designs requiring some travel.
- **Do not** kink or crease the tail of the FSR device if you are bending it; this can cause breaks in the printed silver traces. The smallest suggested bend radius for the tails of evaluation parts is about 0.1" [2.5 mm]. In custom sensor designs, tails have been made that bend over radii of 0.03" (0.8 mm). Also, be careful if bending the tail near the active area. This can cause stress on the active area and may result in pre-loading and false readings.
- **Do not** block the vent. FSR devices typically have an air vent that runs from the open active area down the length of the tail and out to the atmosphere. This vent assures pressure equilibrium with the environment, as well as allowing even loading and unloading of the device. Blocking this vent could cause FSRs to respond to any actuation in a non-repeatable manner. Also note, that if the device is to be used in a pressure chamber, the vented end will need to be kept vented to the outside of the chamber. This allows for the measurement of the differential pressure.
- **Do not** solder directly to the exposed silver traces. With flexible substrates, the solder joint will not hold and the substrate can easily melt and distort during the soldering. Use Interlink Electronics' standard connection techniques, such as solderable tabs, housed female contacts, Z-axis conductive tapes, or ZIF (zero insertion force) style connectors.
- **Do not** use cyanoacrylate adhesives (e.g. Krazy Glue[®]) and solder flux removing agents. These degrade the substrate and can lead to cracking.
- **Do not** apply excessive shear force. This can cause delamination of the layers.
- **Do not** exceed 1mA of current per square centimeter of applied force (actuator area). This can irreversibly damage the device.

Evaluation Parts

Descriptions and Dimensions

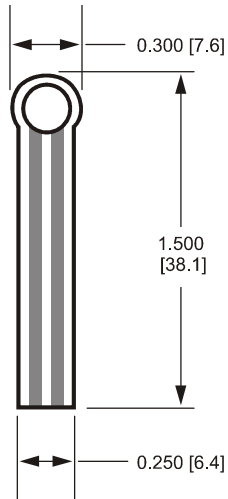


Figure 5:
Part No. 400 (0.2" Circle)

Active Area: 0.2" [5.0] diameter

Nominal Thickness: 0.012" [0.30 mm]

Material Build:

Semiconductive layer

0.004" [0.10] PES

Spacer adhesive

0.002" [0.05] Acrylic

Conductive layer

0.004" [0.10] PES

Rear adhesive

0.002" [0.05] Acrylic

Connector options

- No connector
- Solder Tabs (not shown)
- AMP Female connector

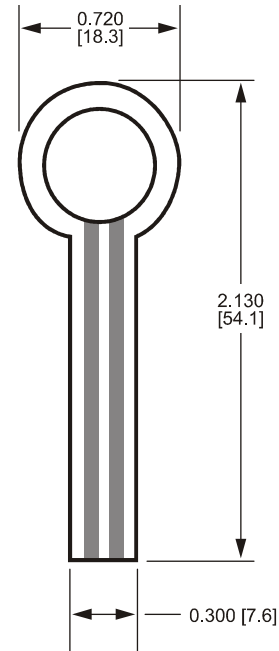


Figure 6:
Part No. 402 (0.5" Circle)

Active Area: 0.5" [12.7] diameter

Nominal thickness: 0.018" [0.46 mm]

Material Build:

Semiconductive Layer

0.005" [0.13] UItem

Spacer Adhesive

0.006" [0.15] Acrylic

Conductive Layer

0.005" [0.13] UItem

Rear Adhesive

0.002" [0.05] Acrylic

Connector

- No connector
- Solder Tabs (not shown)
- AMP Female connector

Dimensions in brackets: millimeters • Dimensional Tolerance: ± 0.015 " [0.4] • Thickness Tolerance: $\pm 10\%$

General FSR Characteristics

These are typical parameters. The FSR is a custom device and can be made for use outside these characteristics. Consult Sales Engineering with your specific requirements.

Simple FSR Devices and Arrays

PARAMETER	VALUE	NOTES
Size Range	Max = 20" x 24" (51 x 61 cm) Min = 0.2" x 0.2" (0.5 x 0.5 cm)	Any shape
Device thickness	0.008" to 0.050" (0.20 to 1.25 mm)	Dependent on materials
Force Sensitivity Range	< 100 g to > 10 kg	Dependent on mechanics
Pressure Sensitivity Range	< 1.5 psi to > 150 psi (< 0.1 kg/cm ² to > 10 kg/cm ²)	Dependent on mechanics
Part-to-Part Force Repeatability	± 15% to ± 25% of established nominal resistance	With a repeatable actuation system
Single Part Force Repeatability	± 2% to ± 5% of established nominal resistance	With a repeatable actuation system
Force Resolution	Better than 0.5% full scale	
Break Force (Turn-on Force)	20 g to 100 g (0.7 oz to 3.5 oz)	Dependent on mechanics and FSR build
Stand-Off Resistance	> 1MΩ	Unloaded, unbent
Switch Characteristic	Essentially zero travel	
Device Rise Time	1-2 msec (mechanical)	
Lifetime	> 10 million actuations	
Temperature Range	-30°C to +70°C	Dependent on materials
Maximum Current	I mA/cm ² of applied force	
Sensitivity to Noise/Vibration	Not significantly affected	
EMI / ESD	Passive device	
Lead Attachment	Standard flex circuit techniques	

Melinex®	A brand of polyester(PET). A substrate with lower temperature resistance than Ulterm® or PES, but with excellent flexibility and low cost. Similar to Mylar™.
Part or Device	The FSR. Consists of the FSR semiconductive material, conductor, adhesives, graphics or overlays, and connectors.
PES	Polyethersulfone. A transparent substrate with excellent temperature resistance, moderate chemical resistance, and good flexibility.
Pin Out	The descriptions of a FSR's electrical access at the connector pad (tail).
Repeatability	The ability to repeat, within a tolerance, a previous response characteristic.
Response Characteristic	The relationship of force or pressure vs. resistance.
Saturation Pressure	The pressure level beyond with the FSR response characteristic deviates from its inverse power law characteristic. Past the saturation pressure, increases in force yield little or no decrease in resistance.
Sensor	Each area of the FSR device that is independently force sensitive (as in an array).
Solder-tabs	Stitched on AMP connectors with tab endings. Suitable for direct PC board connection or for soldering to wires.
Space and Trace	The widths of the gaps and fingers of the conductive grid; also called pitch.
Spacer Adhesive	The adhesive used to laminate FSR devices tighter. Dictates stand-off.
Stand-off	The gap or distance between the opposed polymer film layers when the sensor is unloaded and unbent.
Stand-off Resistance	The FSR resistance when the device is unloaded and unbent.
Substrate	Any base material on which the FSR semi-conductive or metallic polymers are printed. (For example, polyetherimide, polyethersulfone and polyester films).
Tail	The region where the lead out or busing system terminates. Generally, the tail ends in a connector.
Ulterm®	Polyetherimide (PEI). A yellow, semi-transparent substrate with excellent temperature and chemical resistance and limited flexibility.

Interlink Electronics, Inc. holds international patents for its Force Sensing Resistor technology. FSR is a trademark and Force Sensing Resistors is a registered trademark of Interlink Electronics. Interlink and the six dot logotype are registered marks of Interlink Electronics.

Ulterm and Lexan are registered trademarks of G.E., Melinex is a registered trademark of ICI, and Mylar is a trademark of E.I. Dupont & Co.

Suggested Electrical Interfaces Basic FSRs

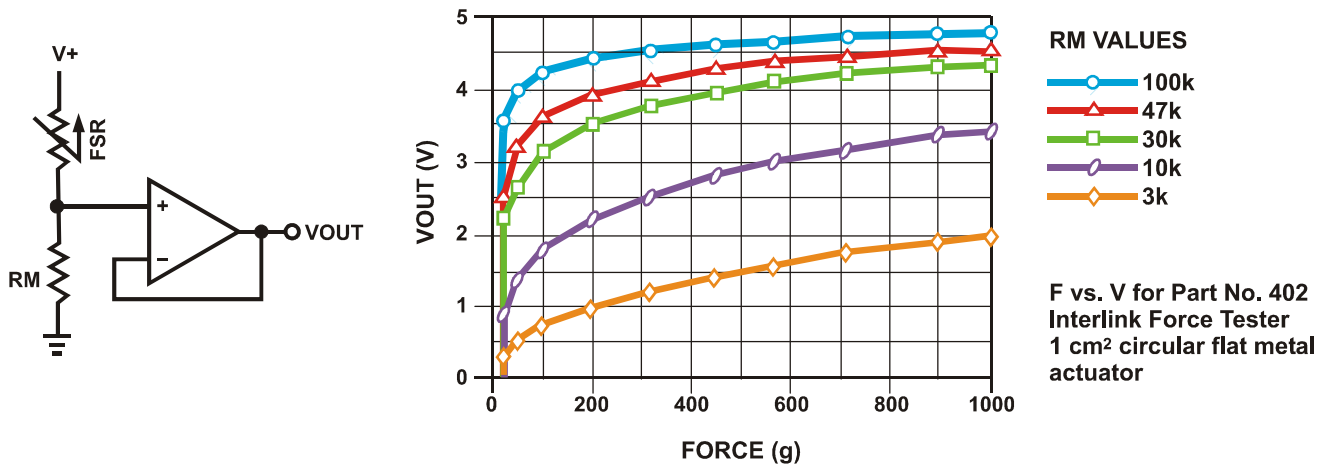


Figure 9
FSR Voltage Divider

FSR Voltage Divider

For a simple force-to-voltage conversion, the FSR device is tied to a measuring resistor in a voltage divider configuration. The output is described by the equation:

$$V_{OUT} = (V+) / [1 + R_{FSR}/R_M].$$

In the shown configuration, the output voltage increases with increasing force. If R_{FSR} and R_M are swapped, the output swing will decrease with increasing force. These two output forms are mirror images about the line $V_{OUT} = (V+) / 2$.

The measuring resistor, R_M , is chosen to maximize the desired force sensitivity range and to limit current. The current through the FSR should be limited to less than 1 mA/square cm of applied force. Suggested op-amps for single sided supply designs are LM358 and LM324. FET input devices such as LF355 and TL082 are also good. The low bias currents of these op-amps reduce the error due to the source impedance of the voltage divider.

A family of FORCE vs. V_{OUT} curves is shown on the graph above for a standard FSR in a voltage divider configuration with various R_M resistors. A $(V+)$ of +5V was used for these examples.

FSR Current-to-Voltage Converter

In this circuit, the FSR device is the input of a current-to-voltage converter. The output of this amplifier is described by the equation:

$$V_{OUT} = V_{REF} \cdot [-R_G/R_{FSR}].$$

With a positive reference voltage, the output of the op-amp must be able to swing below ground, from 0V to $-V_{REF}$, therefore dual sided supplies are necessary. A negative reference voltage will yield a positive output swing, from 0V to $+V_{REF}$.

$$V_{OUT} = (-R_G \cdot V_{REF}) / R_{FSR}.$$

V_{OUT} is inversely proportional to R_{FSR} . Changing R_G and/or V_{REF} changes the response slope. The following is an example of the sequence used for choosing the component values and output swing:

For a human-to-machine variable control device, like a joystick, the maximum force applied to the FSR is about 1kg. Testing of a typical FSR shows that the corresponding R_{FSR} at 1kg is about $4.6k\Omega$. If V_{REF} is $-5V$, and an output swing of 0V to $+5V$ is desired, then R_G should be approximately equal to this minimum R_{FSR} . R_G is set at $4.7k\Omega$. A full swing of 0V to $+5V$ is thus achieved. A set of FORCE vs. V_{OUT} curves is shown in Figure 15 for a standard FSR using this interface with a variety of R_G values.

The current through the FSR device should be limited to less than 1 mA/square cm of applied force. As with the voltage divider circuit, adding a resistor in parallel with R_{FSR} will give a definite rest voltage, which is essentially a zero-force intercept value. This can be useful when resolution at low forces is desired.

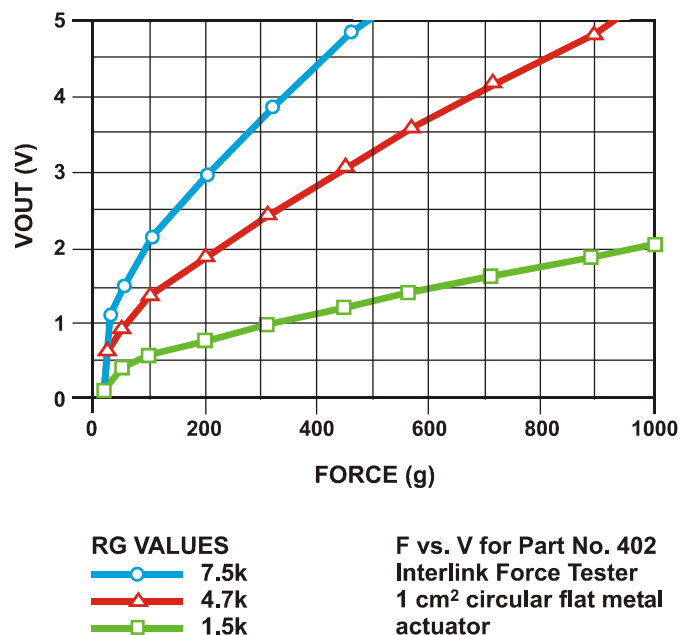
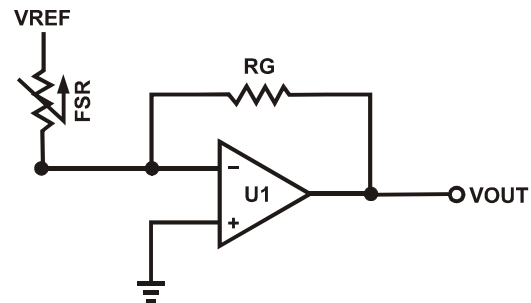


Figure 15
FSR Current-to-Voltage Converter

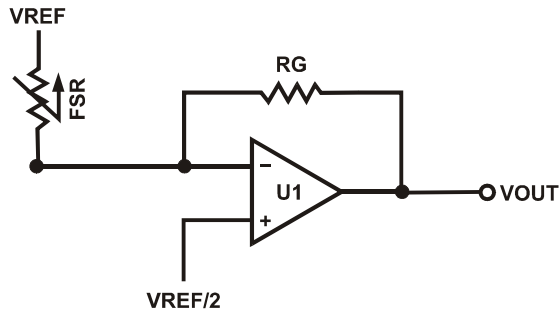


Figure 16
Add'l FSR Current-to-Voltage Converter

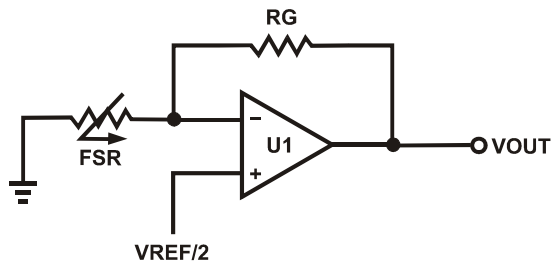


Figure 17
Add'l FSR Current-to-Voltage Converter

Additional FSR Current-to-Voltage Converters

These circuits are a slightly modified version of the current-to-voltage converter detailed on the previous page. Please refer to it for more detail.

The output of Figure 16 is described by the equation:

$$V_{OUT} = [V_{REF}/2] * [1 - R_G/R_{FSR}]$$

The output swing of this circuit is from (VREF/2) to 0V. In the case where RG is greater than RFSR, the output will go into negative saturation.

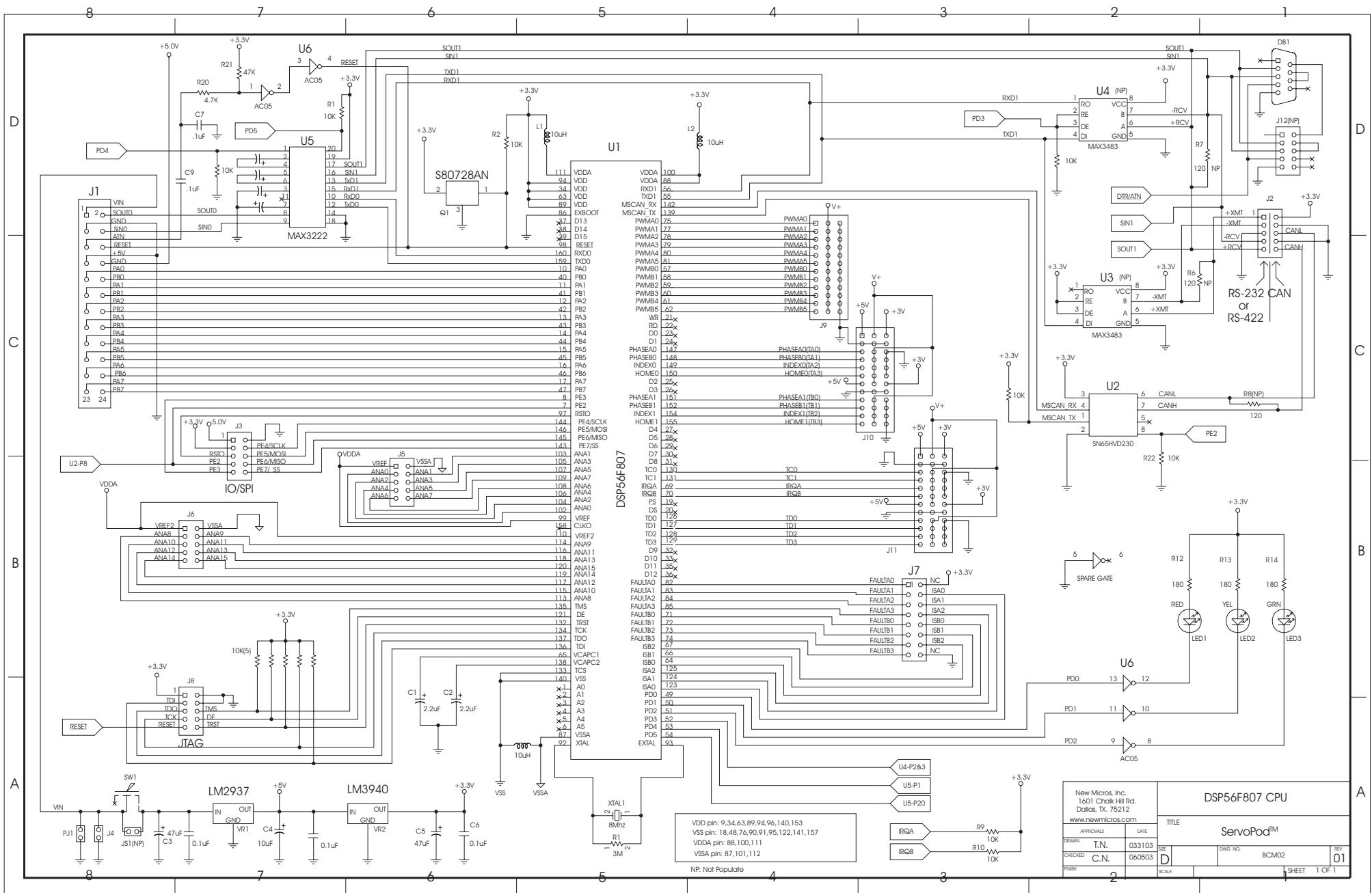
The output of Figure 17 is described by the equation:

$$V_{OUT} = V_{REF}/2 * [1 + R_G/R_{FSR}]$$

The output swing of this circuit is from (VREF/2) to VREF. In the case where RG is greater than RFSR, the output will go into positive saturation.

For either of these configurations, a zener diode placed in parallel with RG will limit the voltage built up across RG. These designs yield one-half the output swing of the previous circuit, but only require single sided supplies and positive reference voltages. Like the preceding circuit, the current through the FSR should be limited to less than 1 mA/square cm of applied force.

Suggested op-amps are LM358 and LM324.



U1

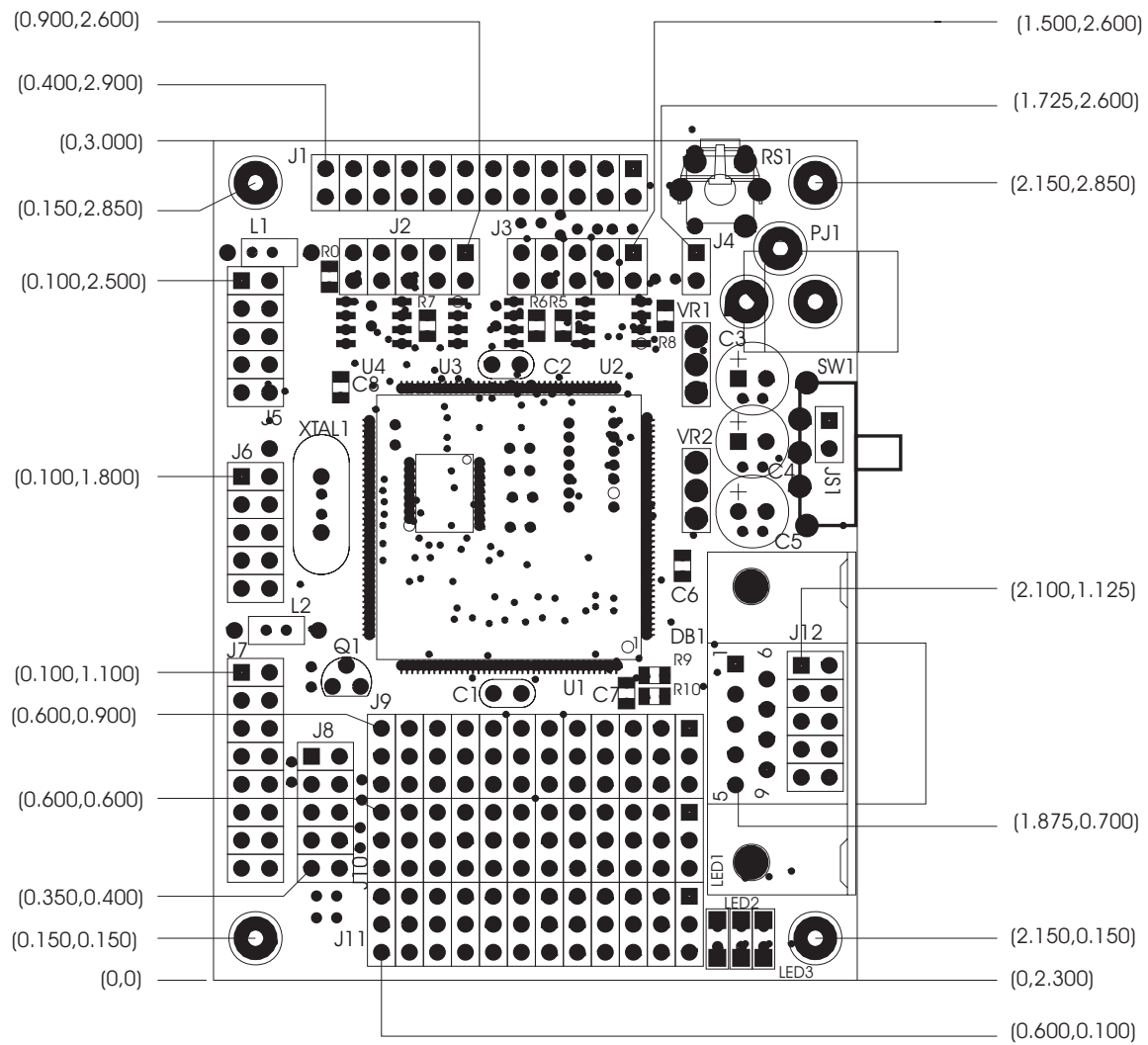
- VDDA 100
- VDD 88
- VDD 66
- TXD1 65
- VDD 63
- VDD 82
- EVBOOT 84
- D13 87
- D14 88
- D15 89
- RESET 90
- RXD0 120
- RXD0 159
- PA0 10
- PR0 40
- PA1 11
- PB1 41
- PA2 12
- PR2 42
- PA3 13
- PB3 43
- PA4 14
- PR4 44
- PA5 15
- PB5 45
- PA6 16
- PR6 46
- PA7 17
- PB7 47
- FE3 48
- FE2 49
- RSTIO 97
- INDEX1 154
- INDEX1(TB2) 155
- INDEX1(TB3) 156
- HOME1 155
- HOME1(TA3) 156
- HOME1(TB3) 157
- HOME1(TB3) 158
- D5 28x
- D6 28x
- D7 35x
- DR 31x
- IC0 130
- IC1 131
- IRGA 69
- IRGB 70
- PS 18x
- DS 20x
- TD0 129
- TD1 127
- TD2 128
- TD3 129
- D9 35x
- D10 35x
- D11 35x
- D12 35x
- FAULT0 82
- FAULT1 83
- FAULT2 84
- FAULT3 85
- FAULT0 71
- FAULT1 72
- FAULT2 73
- FAULT3 74
- ISB2 67
- ISB1 66
- ISB0 64
- IS4 125
- IS1 124
- IS0 123
- PD0 69
- PD1 50
- PD2 51
- PD3 52
- PD4 53
- PD5 54
- EXTAL 93
- VSS 140
- A0 141
- A1 142
- A2 143
- A3 144
- A4 145
- A5 146
- VSSA 92
- XTAL 93
- VDDA 100
- VDD 88
- VDD 66
- TXD1 65
- VDD 63
- VDD 82
- EVBOOT 84
- D13 87
- D14 88
- D15 89
- RESET 90
- RXD0 120
- RXD0 159
- PA0 10
- PR0 40
- PA1 11
- PB1 41
- PA2 12
- PR2 42
- PA3 13
- PB3 43
- PA4 14
- PR4 44
- PA5 15
- PB5 45
- PA6 16
- PR6 46
- PA7 17
- PB7 47
- FE3 48
- FE2 49
- RSTIO 97
- INDEX1 154
- INDEX1(TB2) 155
- INDEX1(TB3) 156
- HOME1 155
- HOME1(TA3) 156
- HOME1(TB3) 157
- HOME1(TB3) 158
- D5 28x
- D6 28x
- D7 35x
- DR 31x
- IC0 130
- IC1 131
- IRGA 69
- IRGB 70
- PS 18x
- DS 20x
- TD0 129
- TD1 127
- TD2 128
- TD3 129
- D9 35x
- D10 35x
- D11 35x
- D12 35x
- FAULT0 82
- FAULT1 83
- FAULT2 84
- FAULT3 85
- FAULT0 71
- FAULT1 72
- FAULT2 73
- FAULT3 74
- ISB2 67
- ISB1 66
- ISB0 64
- IS4 125
- IS1 124
- IS0 123
- PD0 69
- PD1 50
- PD2 51
- PD3 52
- PD4 53
- PD5 54
- EXTAL 93
- VSS 140
- A0 141
- A1 142
- A2 143
- A3 144
- A4 145
- A5 146
- VSSA 92
- XTAL 93

DSP56F807

New Micros, Inc. 1401 Chock Hill Rd. Dallas, TX 75212		TITLE DSP56F807 CPU	
www.newmicros.com		ServoPod™	
APPROVALS	DATE	DESIGN NO.	REV
DRAWN T.N.	033103	BCM02	01
CHECKED C.N.	060503	SCALE	SHEET 1 OF 1

VDD pin: 9,34,63,89,94,96,140,153
 VSS pin: 18,48,76,90,91,95,122,141,157
 VDDA pin: 88,100,111
 VSSA pin: 87,101,112

NP: Not Populate



Board Size: 2.3" x 3.0"
 Dimensions Inches

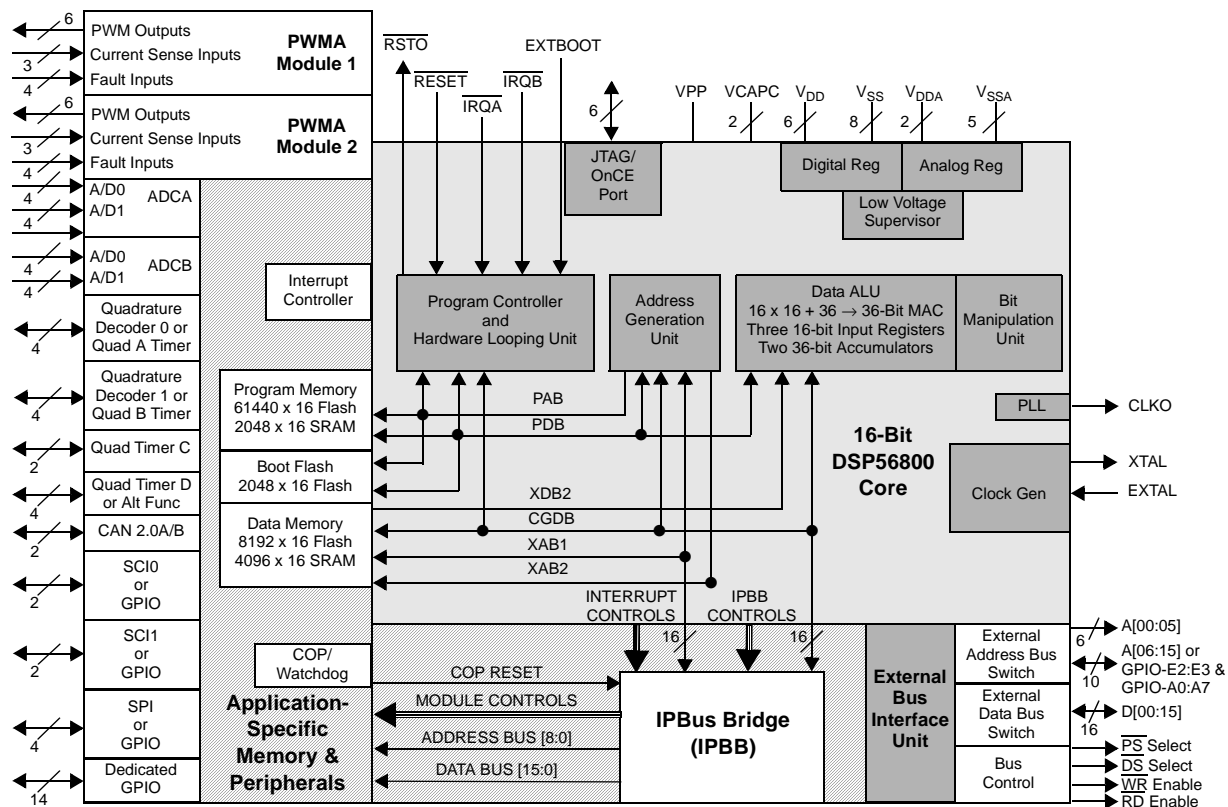


DSP56F807

Product Brief

DSP56F807 16-bit Digital Signal Processor

- Up to 40 MIPS at 80 MHz core frequency
- DSP and MCU functionality in a unified, C-efficient architecture
- Hardware DO and REP loops
- MCU-friendly instruction set supports both DSP and controller functions: MAC, bit manipulation unit, 19 addressing modes
- 60K × 16 bit words Program Flash
- 2K × 16 bit words Program RAM
- 8K × 16 bit words Data Flash
- 4K × 16 bit words Data RAM
- 2K × 16 bit words BootFLASH
- Up to 64K × 16 bit words external Program and Data memory
- Two, 6 channel PWM Modules
- Four, 4 channel, 12-bit ADCs
- Two Quadrature Decoders
- CAN 2.0 A/B Module
- Two Serial Communication Interfaces
- Serial Peripheral Interface
- Up to four General Purpose Quad Timers
- JTAG/OnCE for debugging
- 14 Dedicated GPIO, 18 Shared GPIO
- 160-pin LQFP or 169 PBGA Packages



DSP56800 Digital Signal Processing Core Features

- Efficient 16-bit DSP56800 family DSP engine with dual Harvard architecture
- As many as 40 Million Instructions Per Second (MIPS) at 80 MHz core frequency
- Single-cycle 16×16 -bit parallel Multiplier-Accumulator (MAC)
- Two 36-bit accumulators including extension bits
- 16-bit bidirectional barrel shifter
- Parallel instruction set with unique DSP addressing modes
- Hardware DO and REP loops
- Three internal address buses and one external address bus
- Four internal data buses and one external data bus
- Instruction set supports both DSP and controller functions
- Controller style addressing modes and instructions for compact code
- Efficient C Compiler and local variable support
- Software subroutine and interrupt stack with depth limited only by memory
- JTAG/OnCE Debug Programming Interface

DSP56F807 Memory Features

- Harvard architecture permits as many as three simultaneous accesses to program and data memory
- On-chip memory including a low cost, high volume flash solution
 - $60K \times 16$ bit words of Program Flash
 - $2K \times 16$ bit words of Program RAM
 - $4K \times 16$ bit words of Data RAM
 - $8K \times 16$ bit words of Data Flash
 - $2K \times 16$ bit words of BootFLASH
- Off-chip memory expansion capabilities programmable for 0, 4, 8, or 12 wait states
 - As much as $64 K \times 16$ data memory
 - As much as $64 K \times 16$ program memory

DSP56F807 Peripheral Circuit Features

- Two Pulse Width Modulator modules (PWMA & PWMB) each with 6 PWM outputs, 3 Current Sense inputs, and 4 Fault inputs, fault tolerant design with deadtime insertion, supports both center and edge aligned modes
- Four, 12-bit, Analog to Digital Convertors (ADC) which support two simultaneous conversions with quad, 4-pin multiplexed inputs, ADC and PWM modules are in sync
- Two Quadrature Decoders (Quad Dec0 & Quad Dec1) each with 4 inputs (or two additional Quad Timers A & B)
- Two dedicated General Purpose Quad Timers totalling 6 pins: Timer C with 2 pins and Timer D with 4-pins
- CAN 2.0 A/B Module with 2-pin ports for transmit and receive

- Two Serial Communication Interfaces (SCI0 & SCI1) each with 2 pins (or four additional GPIO lines)
- Serial Peripheral Interface (SPI) with configurable four-pin port (or four additional GPIO lines)
- Computer-Operating Properly (COP) Watchdog timer
- Two dedicated external interrupt pins
- Fourteen dedicated General Purpose I/O (GPIO) pins, eighteen multiplexed GPIO pins
- External reset pin for hardware reset
- JTAG/On-Chip Emulation (OnCE™) for unobtrusive, processor speed-independent debugging
- Software-programmable, Phase Lock Loop-based frequency synthesizer for the DSP core clock

Energy Information

- A single 3.3V power supply
- Fabricated in high-density CMOS with 5-volt tolerant, TTL-compatible digital inputs
- On-chip regulators for digital and analog circuitry to lower cost and reduce noise
- Wait and Stop modes available

DSP56F807 Description

The DSP56F807 is a member of the DSP56800 core-based family of Digital Signal Processors (DSPs). It combines, on a single chip, the processing power of a DSP and the functionality of a microcontroller with a flexible set of peripherals to create an extremely cost-effective solution. Because of its low cost, configuration flexibility, and compact program code, the DSP56F807 is well-suited for many applications. The DSP56F807 includes many peripherals that are especially useful for applications such as: motion control, smart appliances, steppers, encoders, tachometers, limit switches, power supply and control, automotive control, engine management, noise suppression, remote utility metering, industrial control for power, lighting, automation.

The DSP56800 core is based on a Harvard-style architecture consisting of three execution units operating in parallel, allowing as many as six operations per instruction cycle. The microprocessor-style programming model and optimized instruction set allow straightforward generation of efficient, compact code for both DSP and MCU applications. The instruction set is also highly efficient for C/C++ Compilers to enable rapid development of optimized control applications.

“Best in Class” Development Environment

The SDK (Software Development Kit) provides fully debugged peripheral drivers, libraries and interfaces that allow programmers to create their unique C application code independent of component architecture. The CodeWarrior Integrated Development Environment is a sophisticated tool for code navigation, compiling, and debugging. A complete set of evaluation modules (EVMs) and development system cards will support concurrent engineering. Together, the SDK, CodeWarrior, and EVMs create a complete, scalable tools solution for easy, fast, and efficient development.

Product Documentation

The four documents listed in Table 1 are required for a complete description and proper design with the DSP56F807. Documentation is available from local Motorola distributors, Motorola semiconductor sales offices, Motorola Literature Distribution Centers, or online at www.motorola.com/SPS/DSP.

Table 1. DSP56F807 Chip Documentation

Topic	Description	Order Number
DSP56800 Family Manual	Detailed description of the DSP56800 family architecture, and 16-bit DSP core processor and the instruction set	DSP56800FM/D
DSP56F801/803/805/807 User's Manual	Detailed description of memory, peripherals, and interfaces of the DSP56F807, DSP56F803, DSP56F805, and DSP56F807	DSP56F801-7UM/D
DSP56F807 Technical Data Sheet	Electrical and timing specifications, pin descriptions, and package descriptions	DSP56F807/D
DSP56F807 Product Brief	Summary description and block diagram of the DSP56F807 core, memory, peripherals and interfaces (this document)	DSP56F807PB/D

Ordering Information

Consult a Motorola Semiconductor sales office or authorized distributor to order parts.


Table 2. DSP56F807 Ordering Information

Part	Supply Voltage	Package Type	Pin Count	Frequency (MHz)	Order Number
DSP56F807	3.0–3.6 V	Plastic Quad Flat Pack (LQFP)	160	80	DSP56F807PY80

MFAX and OnCE™ are trademarks of Motorola, Inc.

This document contains information on a new product. Specifications and information herein are subject to change without notice.

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part.

Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

All other tradenames, trademarks, and registered trademarks are the property of their respective owners.

How to reach us:

USA/EUROPE/Locations Not Listed: Motorola Literature Distribution; P.O. Box 5405, Denver, Colorado, 80217
1-303-675-2140 or 1-800-441-2447

JAPAN: Motorola Japan, Ltd.; SPS, Technical Information Center, 3-20-1, Minami-Azabu, Minato-ku, Tokyo 106-8573 Japan. 81-3-3440-3569

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd., Silicon Harbour Centre, 2 Dai King Street, Tai Po Industrial Estate, 2 Tai Po, N.T., Hong Kong. 852-26668334

Customer Focus Center: 1-800-521-6274

Mfax™: RMFAX0@email.sps.mot.com

–TOUCHTONE 1-602-244-6609
–US & Canada ONLY 1-800-774-184
–<http://sps.motorola.com/mfax/>

HOME PAGE: <http://www.motorola.com/SPS/DSP>

