

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE



FACULTE DU GENIE ELCTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

MEMOIRE

En vue de l'Obtention Du Diplôme de Master
INSTRUMENTATION BIOMEDICALE

Thème

**CONCEPTION ET REALISATION D'UN POUSSE
SERINGUE PROGRAMMABLE A BASE D'ARDUINO**

Présenté par :

Mr.MOUHOUS MOHAND

Mr.SEDDOUD YACINE

Encadré par :

Mr.TAHANOUT MOHAMMED

Proposé par :

Mr. DJIOUA SMAIL

Mémoire soutenu publiquement le 26/09/2016 devant le jury composé de :

Mr. LAGHROUCHE

Mr. ACHOUR

Mr. BENNAMANE

Mr. TAHANOUT

Professeur à UMMTO, Président

Maitre de Conférence à UMMTP, Examineur

Maitre de Conférence à UMMTO, Examineur

Maitre de Conférence à UMMTO, Promoteur

Remerciements

Avant de commencer la présentation de ce travail, on profite de l'occasion pour remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet de fin d'études.

*Nous exprimons nos remerciements à notre promotrice **M^{me} TAHANOUI** et notre encadreur **Mr DJIOUA** d'avoir accepté de nous encadrer pour notre projet de fin d'études, ainsi que pour leurs soutiens, leurs disponibilités, leurs précieux conseils et pour leurs confiance placé en nous.*

Nous remercions les membres du jury pour l'intérêt qu'ils ont bien voulu porter à notre travail.

Nous réservons ici une place particulière pour remercier vivement nos familles pour leurs affections et leurs soutiens continu. Et à tous ceux qui, d'une manière ou d'une autre, nous ont aidés et encouragés à la réalisation de ce modeste travail.

Merci à tous et à toutes.

SOMMAIRE

Sommaire

Introduction.....	1
Chapitre 1.....	2
1.1 Historique	3
1.2. Généralités sur la perfusion	3
1.3. Les différentes techniques de perfusion	5
1.3.1. La perfusion continue	5
1.3.2. La perfusion courte intermittente	6
1.4. Les différents systèmes de perfusion	6
1.5. Paramètres d'injection	12
1.6. Condition de perfusion et sécurité	13
1.7. Surveillance de la pression de perfusion	13
1.8. Cahier des charges	13
Chapitre 2.....	15
2.1. Schéma synoptique	16
2.2. Fonctionnement	17
2.2.1. Moteur	17
2.2.2. Capteurs	21
2.2.3. Interface	22
2.2.4. Microcontrôleur (μ C)	23
2.2.5. Alimentation	25

2.3. Les caractéristiques techniques du moteur	25
2.3.1. Moteur pas à pas	26
2.3.2. Circuit de commande en puissance	27
2.3.3. Le signal de commande du moteur pas à pas	29
2.3.4. Carte de commande	31
i) Présentation du matériel	32
ii) Présentation du logiciel	34
2.3.5. Capteurs.....	35
i) Jauge de contrainte	35
ii) Capteur de position potentiométrique	36
iii) Capteur switch	37
Chapitre 3.....	38
3.1. Partie matériel	39
3.1.1. Microcontrôleur	39
3.1.2. Commande moteur	40
3.1.3. Interface	41
i) Afficheur	41
ii) Clavier	42
iii) Buzzer et LED	43
3.1.4. Capteur	43
3.1.5. Alimentation	44
3.2. Partie Logiciel	45

3.2.1. Initialisation	46
3.2.2. Menu	47
i) Dernière saisie enregistrée	47
ii) Autres menus	49
3.2.3. Traitement des données par le μ C et commande du moteur	53
3.2.4. Organigramme complet	54
3.3. Réalisation du système	56

Introduction

Face à l'augmentation de la demande de soins, du vieillissement de la population et de la recrudescence des maladies chroniques, les besoins du personnel médical n'ont jamais été autant grandissant. Parmi les complications rencontrées dans les hôpitaux de notre pays, on a constaté un grand manque dans le domaine de la perfusion, que ce soit en équipements ou en personnels qualifiés en maintenance. La perfusion classique par gravité nécessite excessivement la présence d'un infirmier et n'apporte pas de solutions efficaces lors de plusieurs interventions avec une précision de perfusion et en durée de temps plus ou moins longue.

Profitant de notre expérience en électronique biomédical acquise durant notre formation, nous proposons une solution adéquate pour alléger le personnel médical des interventions de perfusions, à savoir, un système de perfusion automatique autonome (semi autonome) à base d'un système programmable. Cette solution est conçue pour apporter une meilleure précision de perfusion et travailler sur de longues durées. Pour mener à bien notre travail, nous avons utilisé la carte ARDUINO MEGA 2560 R3 basée sur le microcontrôleur ATmega2560.

Pour présenter notre réalisation, nous avons rédigé notre mémoire de Master en trois chapitres. Le premier chapitre, s'intitule les techniques de la perfusion » dans lequel nous avons expliqué l'utilité et la fonction du pousse seringue, ainsi que les différents paramètres médicaux à respecter. Le deuxième chapitre est réservé à la « conception du système » dans lequel nous avons donné tous les détails de la conception, que ce soit la plateforme générale (schéma synoptique) ou les composants avec notre choix spécifique de ces composants. Le dernier chapitre dévoile les organigrammes de notre conception logiciel, le circuit électronique avec les différents masques de gravure, ainsi que l'ensemble des tests réalisés pour calibrer le fonctionnement du système.

CHAPITRE I

Les techniques
de perfusion

Chapitre 1

Les techniques de perfusion

1.1. Historique [1]

La circulation sanguine fut décrite pour la première fois par William Harvey et c'était en 1628, c'est ce qui a permis à Sir Christophen Wren en 1657 d'effectuer la première injection intraveineuse chez l'homme et cela à l'aide d'une plume d'oie reliée à une vessie de porc. Grâce à tous ces progrès, Gallie et Harris décrivent pour la première fois la perfusion en continu en 1935. Dans le but de faciliter la pratique, Carrel, en 1940, introduisit l'utilisation d'une pompe électrique

1.2. Généralités sur la perfusion

A l'hôpital, la prise en charge d'un malade débute fréquemment par la pose d'une perfusion. Cette dernière est une technique d'injection intraveineuse, elle permet une injection lente et continue d'une substance médicamenteuse ou de sang dans un organisme ou un organe.

L'administration parentérale des médicaments est un mode d'administration transversal : elle concerne l'ensemble des unités de soins d'un établissement, Elle permet [1] :

- D'assurer la survie en cas d'urgence
- De maintenir un équilibre nutritionnel
- De maintenir l'équilibre hydro-électrolytique lors de périodes opératoires
- De mettre en place des thérapies à débit continu :
 - Antibiothérapie
 - Antalgique
 - Chimiothérapie.

Chapitre I : Les techniques de perfusion

La voie intraveineuse est la plus couramment utilisée en raison de l'abondance des veines et de leurs accès faciles. L'effet est très rapide et n'engendre pas de dégradation des principes actifs dans le système digestif.

La perfusion peut se faire à l'aide de cathéters périphériques (figure 1.1) (cathéter court, cathlon), de cathéters centraux (figure 1.2) ou encore des chambres à cathéters implantables (figure 1.3). [4]

La perfusion intraveineuse prend d'avantage puisqu'elle permet d'administrer des fluides et des drogues aux patients incapables d'avalier des préparations orales, à ce aussi qui ont un problème d'absorption gastro-intestinale, et puis lorsque l'état général ne permet pas une prise de drogue normale.

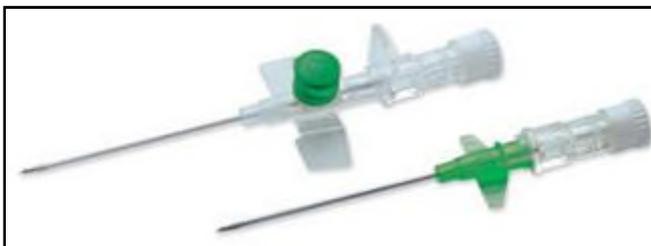


Fig.1.1 : Cathéter central [3]

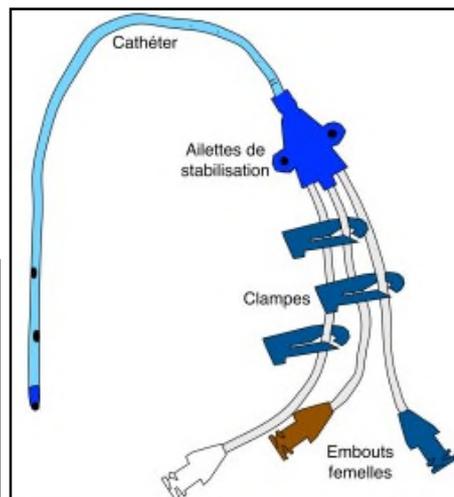


Fig. 1.1 : Cathéter périphérique [3]

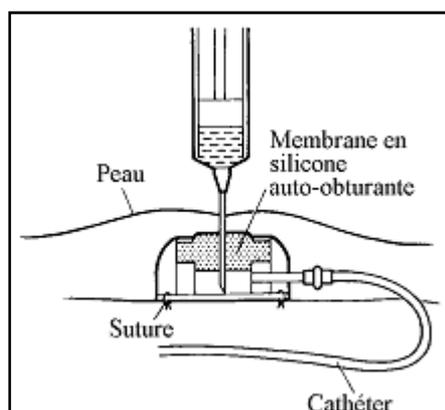


Fig. 1.2 : chambre à cathéters implantable [3]

Chapitre I : Les techniques de perfusion

La durée d'une perfusion, la quantité du liquide à perfuser et la viscosité du produit vont déterminer la voie vasculaire dans laquelle la perfusion pourra être mise. Les principaux vaisseaux utilisés et les plus fréquents pour une perfusion sont divisés en deux types : voies veineuses périphériques et voies veineuses centrales. [2]

- La voie périphérique : (veine superficielle) vue ou palpée, est une voie de courte durée d'utilisation (24/72h). Le médicament passe directement dans la circulation veineuse.
- La voie centrale : (veine profonde non vue, située par des repères anatomiques ou échographiques), elle est réservée aux médecins (diplômés ou en formation sous la responsabilité d'un collègue diplômé). La personne qui pose ou qui met en service la perfusion doit en assurer la surveillance pendant toute la durée d'utilisation et/ou transmettre les informations significatives afin de permettre le suivi de la prise en charge.

1.3. Les différentes techniques de perfusion

Principalement, on distingue trois combinaisons de perfusion : [1]

La perfusion en bolus : En médecine et en pharmacie, le terme bolus ou bol, désigne une dose de médicament ou de produit de contraste que l'on doit administrer au complet d'un seul coup, en l'utilise pour les médicaments en urgence, ou en chimiothérapie.

1.3.1. La perfusion continue :

C'est l'administration de médicaments à vitesse constante et dont la concentration sérique des médicaments est stable, dans un intervalle du temps régulier.

En fonction de cette méthode là (perfusion continue) que toute la suite de notre étude et conception du système prendre paramètre.

Exemple : Antalgie du patient cancéreux, sédation du patient de réanimation, grand brûlé.

1.3.2. La perfusion courte intermittente :

Intermédiaire entre la perfusion en bolus et la perfusion continue. Perfusion courte de l'ordre de 30 à 60mn, toutes les 4, 6, 8 ou 12 heures selon la pharmacocinétique des médicaments (ex : Antibiothérapie du patient infecté).

1.4. Les différents systèmes de perfusion

Comme le montre le schéma dans la figure (1.4) ci-dessous, il existe deux grands modes de perfusion : par gravité et par pression. [1]

Fig. 1.3 : Les différents dispositifs de la perfusion [1].

La figure (1.5) montre le dispositif de perfusion par gravité. Avec ou sans régulateur de débit, c'est le système le plus simple et le plus couramment utilisé. Le contenant du soluté est situé en hauteur par rapport à l'abord vasculaire, idéalement à environ 1 mètre 80 du sol, et le soluté s'écoule grâce à la masse de la colonne de liquide et entre dans la veine du patient avec une pression motrice proportionnelle à la différence de hauteur entre les deux. [1]

Son principal inconvénient est d'être très peu précis car le volume d'une goutte varie en fonction des différents paramètres. Du fait de certain inconvénient majeur, ce mode de perfusion doit être réservé aux médicaments qui nécessitant pas une précision et une stabilité du débit de perfusion. [2]

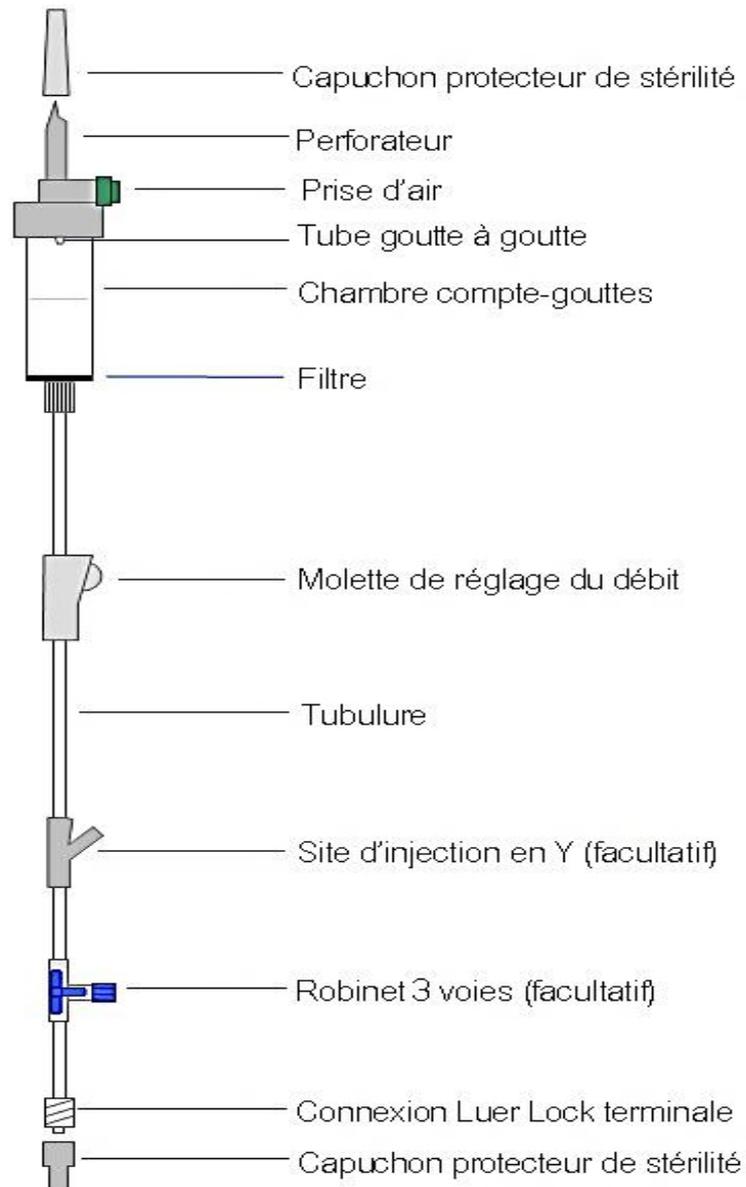


Fig. 1.4 : système de perfusion par gravité

Chapitre I : Les techniques de perfusion

Pour ce qui est de la perfusion par pression, on va illustrer deux structures :

Comme le montre la figure (1.6), les pompes volumétriques sont des appareils injectant des fluides d'un volume supérieur à 60 ml, appeler aussi pompes à perfusion, elles administrent de grands volumes de solutions injectables (médicaments ou nutriments) dans le système circulatoire d'un patient, et cela se fait par voie parentérale, pour des volumes inférieure à 60ml en utilise des pousse seringue. [6]

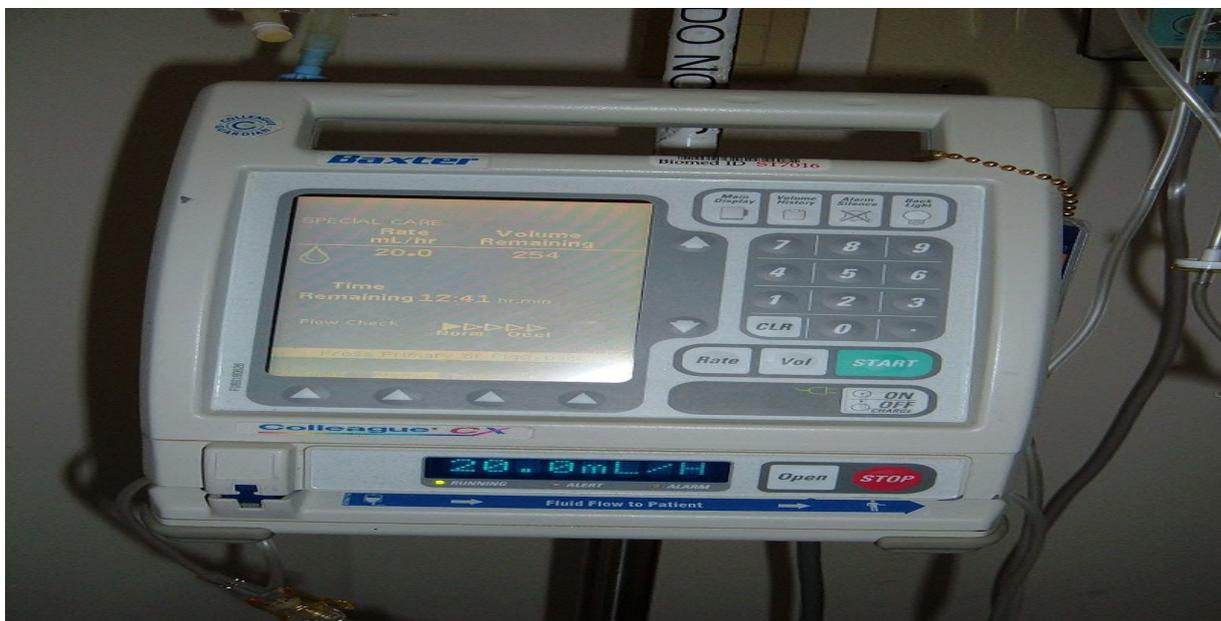


Fig.1. 5 : Pompe volumétrique

Les poussettes seringue représentés dans la figure (1.7) sont réputé pour leurs grandes utilisation, précision ainsi que leurs fiabilités, afin de subvenir au besoin des différents utilisateurs du corps médical, on a pris d'avantage l'étude et la conception d'un pousse seringue.

Chapitre I : Les techniques de perfusion

Les poussettes seringueuses utilisent l'énergie mécanique électrique d'un moteur qui pousse sur le piston de la seringue. Là aussi divers facteurs influent sur la précision du débit en sortie du dispositif. Le plus important est le diamètre interne de la seringue, c'est sur cette donnée que se base la seringue électrique afin de déterminer la vitesse d'avancement du piston. Le diamètre interne des seringues varie d'une marque à l'autre et de leurs quantités, il est donc primordial de bien spécifier le type de seringue à usage unique utilisée avant toute perfusion.



Fig.1.6 : Différents types de poussettes seringueuses

Il faut veiller également à ne pas changer la position de la seringue lors de la perfusion, des études ont en effet constaté que lors de la verticalité de certaines seringues, l'administration d'un bolus de produit pouvait se produire. Le volume de ce bolus atteint parfois plus de 2 ml, et peut être ensuite suivi d'une période pendant laquelle aucun médicament n'est perfusé au patient. [6]

Chapitre I : Les techniques de perfusion

La taille de la seringue est également un facteur de variation de débit, il est toujours conseillé notamment lorsque l'on utilise de faibles débits d'utiliser une seringue de petit diamètre.

Une étude a démontré qu'à faible débit (1 ml/h), le choix d'une seringue de 20 ml réduisait le risque de bolus et de période sans débit par rapport à une seringue de 50 ml. Les normes en vigueur exigent que les seringues utilisées aient un diamètre interne précis d'environ 1 %.

La tolérance de variation du débit admis au niveau du moteur d'une seringue pulsée est de + ou -1 %. [2]

Toutes ces données techniques vont nous amener à concevoir une base de données qui sera utilisée par le programme pour bien contrôler l'opération d'injection.

Pour conclure et justifier notre choix, La perfusion par pression présente plusieurs avantages, et parmi eux on cite les suivants : [2]

- Réglage aisé et faible débit.
- Possibilité de perfuser de petites quantités de solutés.
- Surveillance plus réduite.
- Une précision chirurgicale en débit et durée.

Chapitre I : Les techniques de perfusion

1.5. Paramètres d'injection [1]

Critères		Gravité	Pompe volumétrique	Pousse seringue
Précision (%)		20 < P < 50	5	2
Gamme de débit (ml/h)			0,1 à 1000	0,1 à 1200
Volume maximum (ml)		Limite flacon ou poche	Limite flacon ou poche	60
Mode	Continu	X	X	X
	Intermittent		X	X
	Circadien		X	X
	PCA*		X	X
	NPT*		X	X
	Proportionnel au poids		X	X
Prix consommables		Faible	Moyen à élever pour cassette	Faible
Avantages		Faible coût, rapides à mettre en place, peu de matériel nécessaire	Grande précision, perfusion rapide de grand volume	Grande précision, perfusion rapide de grand volume, marché de consommable non captif
Inconvénients		Peut être imprécise, dépendant de la pression veineuse et de la hauteur de perfusion, pas de détection d'air	Coût dispositif et tubulaires, marché de consommables captif	Coût

* PCA : Analgésie Contrôlée par le Patient

NPT : Nutrition Parentérale Totale

1.6. Condition de perfusion et sécurité

Afin de pouvoir s'assurer qu'une perfusion se fasse dans les normes, un infirmier responsable doit contrôler les différents éléments de perfusion tel que le cathéter et ses tubulaires, et veiller au bon fonctionnement de l'appareil à utiliser.

1.7. Surveillance de la pression de perfusion

Ce pousse seringue est équipé d'un système de surveillance exclusif de la pression de perfusion qui surveille en permanence les paramètres de pression à laquelle la seringue est actionnée pendant la perfusion. Si le déroulement de la perfusion est gêné, par exemple par un robinet fermé ou une occlusion du cathéter, au point que la pression de perfusion atteigne le seuil d'alarme prééglé, le pousse-seringue s'arrête automatiquement et une alarme se déclenche.

1.8. Cahier des charges

On a opté pour une tâche bien précise, proposer une conception robotique (mécanique-électrique), qui sera un pousse seringue autonome (électrique programmé), il sert à injecter et contrôler une solution (médicament) à un patient, ainsi que la prévention en cas de panne. Afin de pouvoir mener à bien notre projet, nous avons proposé le cahier de charge représenté dans le tableau suivant :

Chapitre I : Les techniques de perfusion

Type de seringue utilisé	BD Plastipac 20ml
Adaptation au personnel médical	Niveau du personnel : - Infirmier Réglages : - Débit Comptes rendus : - Témoin visuel de fonctionnement sur secteur ou sur Batterie, de mise sous tension - Alarme sonore en fin de perfusion, si problème.
Source d'énergie	Electrique - 220V AC - 12V DC sur batterie, autonomie 8h
Débit de perfusion	Débit réglable : 1 ml minimum 20 ml maximum

CHAPITRE II

Conception du système

Chapitre 2

Conception du système

2.1. Schéma synoptique

Le pousse seringue électrique comporte deux éléments clé : un mécanisme électromécanique, qui est responsable d'injecter le médicament dans le corps, et un dispositif électronique (microcontrôleur) qui sert à contrôler le mécanisme via une interface d'échange d'information et prévenir les défaillances. Afin de pouvoir bien comprendre le fonctionnement du pousse seringue et simplifier sa réalisation, nous donnons le schéma synoptique de la figure (2.1) ci-dessous représentant les différents composants principaux et leurs différentes liaisons :

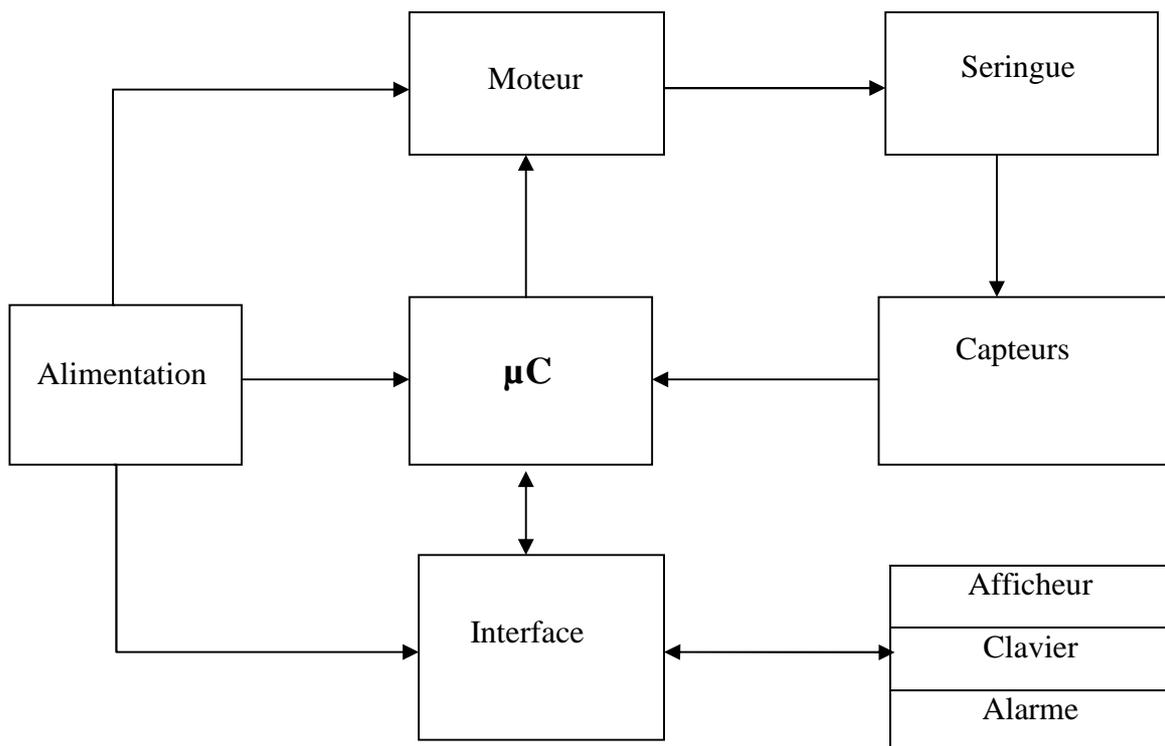


Fig. 2.1 : Schéma synoptique.

Un pousse seringue est une pompe programmable qui permet l'injection de médicaments à un patient, vu le schéma synoptique, notre conception reposera sur trois paliers principaux : mécanique, électrique et programmation.

- **Partie mécanique** : consiste à faire pousser le piston en utilisant le mouvement rotatif d'un moteur et un ensemble de dispositif mécanique qui permettent de transformer cette rotation en énergie mécanique translatoire (des engrenages, des supports, des écrous), et un corps qui porte la seringue (le boîtier).
- **Partie électrique (prévention)** : cette partie comporte un ensemble de capteurs (bande résistive, switch, jauge de contrainte) qui nous permettent de recueillir des données techniques (données électriques : tension, résistance, capacité...) via des phénomènes physiques (pression, distance, position...), et une plateforme informatique (système microcontrôleur) qui permettent de gérer l'ensemble du mécanisme. Dans cette partie on se focalise essentiellement sur les paramètres de localisation.
- **Partie interface** : elle permet l'échange d'informations entre environnement interne et externe, et pour cela on utilise un périphérique d'entrées (un clavier), des périphériques de sorties (un écran LCD, des LED, des alarmes sonores ...).

2.2. Fonctionnement

Suivant le schéma synoptique, chaque partie est un élément clés de notre conception, en premier lieu, on verra les définitions globales de chaque corps.

2.2.1. Moteur

Un moteur est un composant de conversion d'énergie électrique en énergie mécanique. En électronique, généralement on rencontre trois types de moteurs : moteur à courant continu (DC), servomoteur et moteur pas-à-pas.

Chapitre II : Conception du système

Moteur à courant continu : il s'appelle aussi Machine à Courant Continu. Le moteur à courant continu est composé de deux parties principales : le rotor (partie qui tourne) et le stator (partie qui ne tourne pas ou statique). En électrotechnique le stator s'appelle aussi inducteur et le rotor s'appelle l'induit.



Fig. 2.2 : Moteur à courant continu

Servomoteur : c'est un moteur un peu particulier, puisqu'il comporte une électronique de commande. Le nom vient en fait du latin *servus* qui signifie esclave. De manière semblable aux moteurs à courant continu, les servomoteurs disposent d'un axe de rotation qui se trouve au centre de la roue. Cet axe de rotation est en revanche entravé par un système de bridage. Les servomoteurs ont donc l'avantage d'être *asservis* en position angulaire. Cela signifie, que l'axe de sortie du servomoteur respectera une consigne d'orientation que vous lui envoyez en son entrée. [8]



Fig.2.3 : Servomoteur

Chapitre II : Conception du système

Moteur pas-a-pas : [7] Pour en terminer avec les différents types des moteurs qui existent, nous allons parler d'un moteur un peu particulier qui regroupe la technologie des deux

Moteurs précédant, et qui est cependant très utilisé dans le domaine de la robotique et tout ce qui touche à la précision d'un mouvement. Pour ce type de MoteurPasàpas, on va illustrer deux modèles :

- Moteur pas à pas à aimants permanent :
 - Moteur pas à pas unipolaire.
 - Moteur pas à pas bipolaire.
- Moteur pas à pas à reluctance variable.



Fig.2.4 : Moteur pas à pas

Chapitre II : Conception du système

Pour justifier notre choix du moteur, on a dû faire un tableau comparatif : [7]

Remarque : cité le servomoteur n'est pas obligatoire car ce dernier est limité dans son déplacement, il est conçu pour travailler entre 0° et 180° (sans modification).

	Avantages	Désavantages
Moteur pas à pas	<ul style="list-style-type: none">• Fonctionnement en boucle ouverte (pas de régulation nécessaire)• Pas besoin de codeur incrémental• Précision des pas élevés• Moteurs pas à pas linéaires existants	<ul style="list-style-type: none">• Electronique de commande plus complexe que pour le moteur DC• Puissance inférieure à encombrement égal
Moteur DC	<ul style="list-style-type: none">• Réglage de la vitesse aisée• Commande électrique simple• Puissance élevée pour un faible encombrement	<ul style="list-style-type: none">• Régulation nécessaire• Codeur incrémental nécessaire• Transformation rotative linéaire nécessaire (courroie...)

Un moteur pas à pas peut fonctionner en boucle ouverte, puisque c'est un moteur-position, c'est-à-dire que la position de l'arbre est commandée par les valeurs des courants dans les enroulements. Un servomoteur fonctionne en boucle fermée : la commande dépend de la différence entre la position demandée et la position constatée.

Les principaux composants qui constituent la mécanique du pousse seringue, se repose essentiellement sur :

Un moteur qui permet un déplacement linéaire du piston d'une seringue via une vis sans fin qui tourne grâce à une série d'engrenages et qui permet le bon déplacement du piston afin de transférer la solution contenue vers la circulation d'un patient.

Vis sans fin : c'est un cylindre comportant une cannelure hélicoïdale (parfois plusieurs), la faisant ressembler à une tige filetée. Associée à un pignon, elle constitue un engrenage gauche (les deux axes ne sont pas dans le même plan), dans lequel elle se comporte comme une roue à une dent (ou plus, selon le nombre de cannelures). On appelle aussi parfois ce système roue et vis sans fin.

2.2.2. Capteurs

Notre réalisation se compose d'un ensemble de capteurs (bande résistive, switch, jauge de contrainte) qui nous permettrons de recueillir des données techniques (données électriques : tension, résistance, capacité...) via des phénomènes physiques (pression, distance, position...), et une plateforme informatique (un ensemble de programmes) qui permet de gérer l'ensemble du mécanisme.

Nous avons vu dans la présentation générale, qu'il était nécessaire de surveiller un certain nombre de paramètres afin de garantir une perfusion correcte. Cependant, tous ces capteurs seront pilotés par un programme. Lorsque le programme fonctionne normalement (exécute correctement ces instructions) il envoie régulièrement des informations sur ses bus de contrôle.

Par contre si le microcontrôleur se trouve bloqué dans une boucle, l'activité sur les bus disparaît (les signaux ne changent plus d'état) et il est nécessaire d'avertir l'infirmier en déclenchant une alarme (sonore, lumineuse).

2.2.3. Interface

Elle permet l'échange d'information entre environnement intérieure et extérieure, et pour cela on a besoin de périphériques d'entrer (le clavier...), et de périphérique de sortie (un afficheur LCD, des LED, des alarmes...).

Interface d'entrée : afin de pouvoir introduire les différents paramètres dans notre appareil, tel que le réglage de débit et le temps d'administration, on a besoin d'un clavier qui fera office d'interface de contrôle entre l'infirmier et l'appareil (pousse seringue).

Interface de sortie : pour répondre aux différents besoins des infirmiers en matière de protection et d'avertissement en cas de problème lié à l'appareil, on a dû introduire différents éléments d'avertissements tel qu'une alarme pour signaler si un problème se présente dans l'appareil tel qu'un dysfonctionnement de l'appareil ou une coupure d'électricité, des LED témoins qui indiqueront l'état de fonctionnement de l'appareil et son alimentation (sur secteur ou sur batterie), et un afficheur LCD illustré dans la figure (2.5) ci-dessous qui permettra la simplification d'utilisation du pousse seringue et qui affichera les différents réglages introduits par l'utilisateur (infirmier).

Pour notre cas on a choisi d'utiliser un afficheur LCD 20*4 soit 20 caractères et 4 lignes et cela dans le but de n'afficher que le strict nécessaire.



Fig.2.5 : Afficheur LCD 20x4

2.2.4. Microcontrôleur (μ C)

En générale le μ C est un circuit intégré qui rassemble les éléments essentiels d'une structure programmable (ordinateur, ...), processeur, mémoires (mémoire morte pour le programme, mémoire vive pour les données), unités périphériques et interfaces d'entrées-sorties. Les microcontrôleurs se caractérisent par un plus haut degré d'intégration, une plus faible consommation électrique, une vitesse de fonctionnement plus grande (de quelques mégahertz jusqu'à plus d'un gigahertz) et un coût réduit par rapport aux microprocesseurs polyvalents utilisés dans les ordinateurs personnels.

Dans notre application on exploite un autre type de μ C, qui est une plateforme Arduino, c'est elle qui pilotera toute l'électronique de notre système, notre choix est basé sur ces paramètres-là :

Chapitre II : Conception du système

Avantages : multiplateforme (linux, mac os ou Windows), programmation simple, claire et intuitive, entièrement open-source : hardware et software, les « shields » : ajoutent beaucoup de possibilités, sans soudure, le plus important : on branche et ça marche ! Et en plus de tout cela, on peut avoir les différentes versions de l'Arduino avec un prix moindre.

Inconvénients : puissance du μC bridée par les fonctions simples d'emploi, prix des shields assez élevé en général, à part tout cela, on n'a pas trouvé un vrai inconvénient à cette carte.

Vu ces avantages et inconvénients, l'Arduino est plus adéquate pour notre conception, et dans notre cas, on utilisera un Arduino MEGA 2560 R3 qu'on peut apercevoir dans la figure (2.6) ci-dessous, doté d'un microcontrôleur ATmega328.

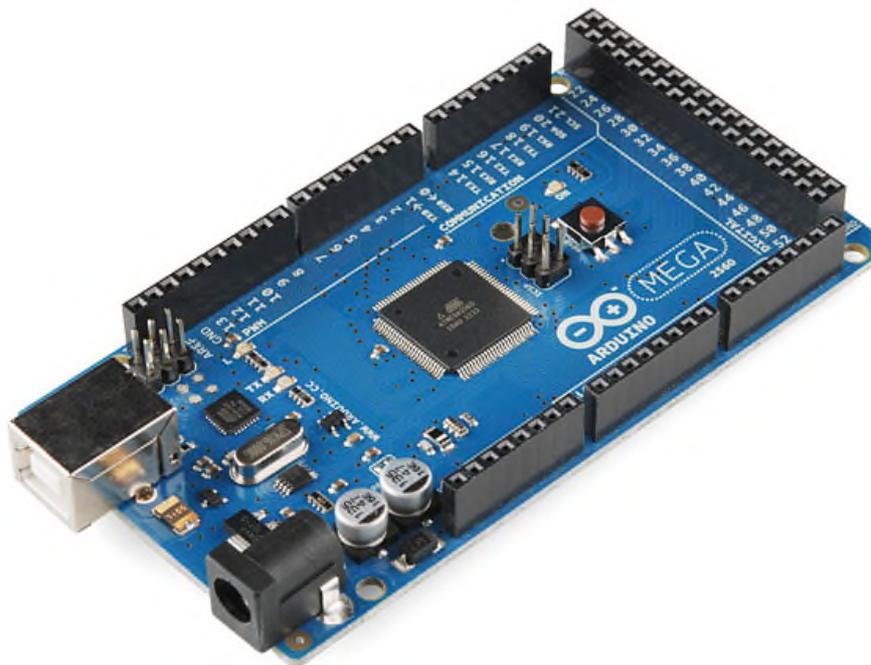


Fig.2.6 : Arduino MEGA2560 R3

2.2.5. Alimentation

Afin de pouvoir faire une bonne alimentation pour notre appareil, nous aurons besoin de différents éléments qui pourront alimenter en même temps notre carte Arduino, le moteur pas à pas, l'afficheur LCD ainsi que les différents capteurs présents dans l'appareil.

Le pousse seringue est un appareil utilisé dans un environnement hospitalier et dont le fonctionnement doit être de manière permanente et cela, même en cas de coupure de courant,

Pour ce faire, nous installerons une batterie qui prendra le relais de l'alimentation du secteur dès qu'une coupure d'électricité ce produit, afin de pouvoir choisir qu'elle type de batterie utilisée, nous aurons besoin de faire quelque test sur l'appareil et connaitre les différents paramètres tel que le courant, la tension d'alimentation, et surtout le plus important, la durée dont l'appareil doit rester en fonctionnement.

Ces différents éléments cités là-dessus, l'alimentation et la batterie seront plus détaillé dans le 3^{ème} chapitre.

2.3. Les caractéristiques techniques du moteur

Pour notre application, le moteur pas à pas est donc avantageux par rapport au moteur DC et au SV. En effet, la précision des pas ainsi que le fait que la régulation n'est pas obligatoire, sont des avantages pour la réalisation de notre système.

2.3.1. Moteur pas à pas

Le moteur pas à pas est un moteur qui tourne en fonction des impulsions électriques reçues dans ses bobinages. L'angle de rotation minimal entre deux modifications des impulsions électriques s'appelle un pas. On caractérise un moteur par le nombre de pas par tour (c'est à dire pour 360°). Les valeurs courantes sont 48, 100 ou 200 pas par tour.

Le principe de fonctionnement des moteurs pas à pas repose sur la commutation successive des enroulements stator (ou phase). Pour cela, une impulsion électrique est traduite par un séquenceur agissant sur une électronique de commutation (drivers ou transistors de puissance) qui distribue les polarités dans les enroulements. Une seule commutation provoque un seul pas quelle que soit la durée de l'impulsion (supérieur à une valeur minimale).

Les moteurs les plus couramment rencontrés présentes des pas de : [7]

- 0.9° soit 400 pas par tour ;
- 1.8° soit 200 pas par tour ;
- 3.6° soit 100 pas par tour ;
- 7.5° soit 48 pas par tour ;
- 15° soit 24 pas par tour ;

L'électronique pilotant un moteur pas à pas peut se diviser en 3 fonctions :

- L'alimentation du moteur avec ses contraintes de tensions, courants et puissances à dissiper. C'est l'électronique de puissance.
- Le séquenceur qui gère la chronologie des impulsions.
- L'oscillateur

2.3.2. Circuit de commande en puissance

Vu qu'on a précisé la plateforme qui pilotera le système (une carte Arduino), alors :

En pratique une carte Arduino n'a pas les performances électriques pour piloter un moteur. Ces derniers, quand ils tournent, génèrent tout un tas de parasites qui peuvent être des surtensions très grandes par rapport à leur tension d'alimentation. De plus, le courant qu'ils demandent est bien trop grand par rapport à ce que peut fournir une sortie numérique d'une carte Arduino (environ 40 mA), pour cela, on doit utiliser un module (driver) électronique qui est le A4988.

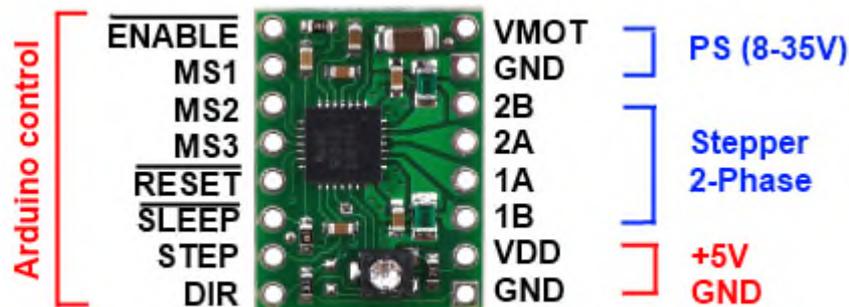


Fig.2.7 : Le Driver A4988.

Le module A4988 ne va pas uniquement protéger la carte Arduino mais aussi nous permettre d'avantage, la commande du moteur est bien efficace à grande vitesse en gardant le couple. (Le couple du moteur va baisser très fortement avec la vitesse en utilisant un simple circuit électronique). Ce même driver se constitue d'une logique qui nous permet aussi de gérer et d'engendrer des divisions de pas qui est la résolution (le plein pas, demi pas, quart de pas, huitième de pas et seizième de pas).

Chapitre II : Conception du système

Un pilote micro-stepping tel que Le A4988 permet d'obtenir une plus grande résolution en autorisant des positions intermédiaires dans un pas. Cela est rendu possible en modulant intelligemment la quantité de courant dans les bobines du moteur pas-à-pas.

Par exemple : piloter un moteur en mode "1/4 de pas" permet d'obtenir 800 micro-steps(micro-pas) sur un moteur prévu pour 200 pas, et cela en utilisant 4 niveau de courants différents pour chacun des micro-pas.

Pour bien illustrer ça, voilà quelque caractéristique technique :

- ✓ Interface de contrôle de pas et de direction simplissime.
- ✓ Un empattement de 2.54mm.
- ✓ 5 résolutions différentes de pas
- ✓ Contrôle en courant ajustable, un potentiomètre permet de fixer le courant maximum à l'aide d'un potentiomètre. Cela permet d'utiliser une tension supérieure à celle recommandée par le moteur (puisque le courant est limité) et permet ainsi d'atteindre un débit de pas plus élevé (et donc une vitesse plus élevée).
- ✓ Une commande de hachage intelligente qui sélectionne automatiquement le bon mode de décroissance de courant (fastdecay = baisse/chute rapide du courant ou slow decay = chute lente).
- ✓ Protection (arrêt) en cas de surchauffe, sous tension, surcharge, sur-courant, court-circuit. Utile en cas de surcharge/blocage moteur (car le courant va monter en flèche (pic-courant), ce qui doit provoquer la mise en protection).

La résolution du module :

MS1	MS2	MS3	Résolution Micro-pas
Low	Low	Low	Pas complet
High	Low	Low	Demi-pas
Low	High	Low	Quart de pas
High	High	Low	Huitième de pas
High	High	High	Seizième de pas

2.3.3. Le signal de commande du moteur pas à pas :

Pour permettre le bon fonctionnement de notre moteur pas à pas qui fonctionne grâce à des impulsions, nous avons étudié deux signaux, qui peuvent être générés par la carte, (signal PWM représenté dans la figure (II.8) et un signal carré), et cela afin de connaître leur rôle, et la principale différence entre ces deux signaux.

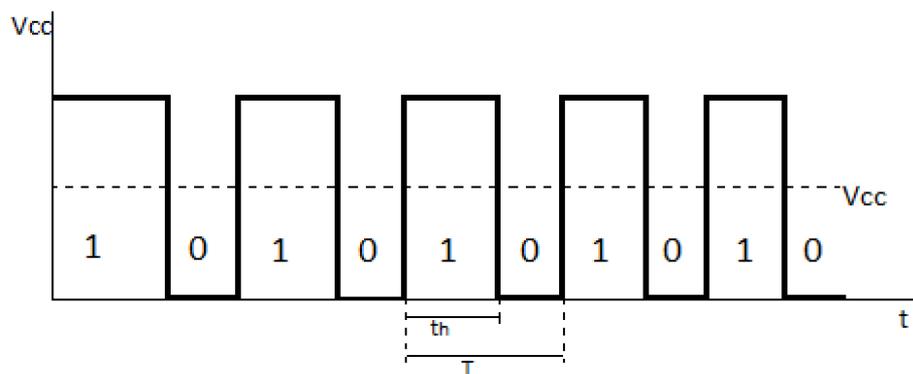


Fig. 2.8 : Signal PWM

Chapitre II : Conception du système

La PWM qui signifie Pulse With Modulation ou en français Modulation à Largeur d'Impulsion (MLI) est un signal numérique de période « T » caractérisé par la durée de l'impulsion « t_h » (pour laquelle la tension est celle de l'état haut) et d'un rapport cyclique (α) variable, ce rapport cyclique α étant la proportion d'état haut par rapport au total du signal, s'exprime en % comme suit :

$$\alpha = 100 \times \frac{t_h}{T}$$

- Si $t_h = 0$ alors $\alpha = 0 \%$; V_{moy} est nul.
- Si $t_h = T$ alors $\alpha = 100 \%$; V_{moy} est égale à V_{cc} .

Donc en faisons varier le rapport cyclique α , la vitesse variera selon l'augmentation ou la diminution de ce rapport, que lui aussi influence sur la valeur moyenne récupérer.

Après une étude du signal PWM, nous avons constaté que ce type de signal est surtout utilisé pour le contrôle des moteurs à courant continu, et cela en faisant varier la valeur moyenne, qui aussi varie la vitesse de rotation, ce mode de fonctionnement est destiné pour les moteurs CC avec une fréquence fixe, et vu que notre signal de commande impose deux commandes restreintes (rapport cyclique 50%, et fréquence variable), nous avons donc opté pour un signal carré.

Les impulsions carrées de notre signal interagissent avec la rotation du rotor du moteur pas à pas, cette rotation se fait lorsqu'on coupe le courant dans une phase tout en déclenchant l'alimentation de l'autre phase, cela est figuré dans la figure (2.9). Dans ce mode, le courant ne circule que dans une phase à la fois (mode One Phase ON, Full Step ou pas complet). Le sens du courant est inversé à chaque alternance. Ce mode est aussi appelé excitation ondulée.

La génération des signaux fréquents se fait avec un chronomètre-compteur ou bien ce qu'on appelle dans le langage programmable les Timer. La plus grande précision sera obtenue avec un Timer 16 bits. Dans notre carte Mega les broches qui délivrent les signaux fréquents sont : 2, 3, 4, 5, 6 à 13.

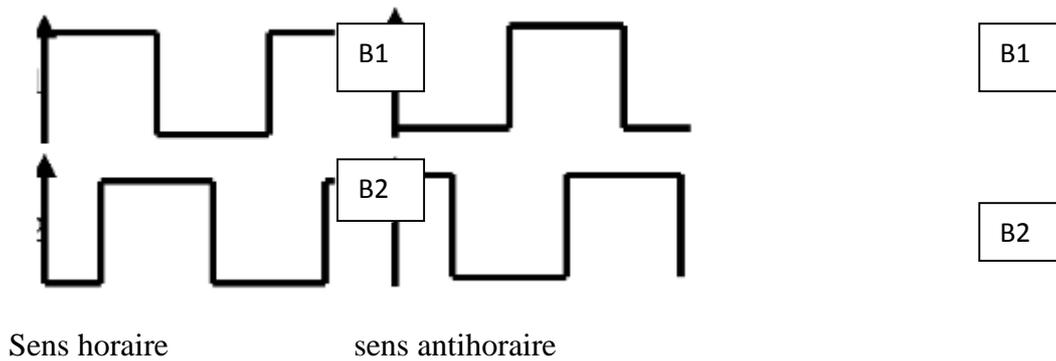


Fig. 2.9 : Signal de commande du moteur pas à pas

2.3.4. Carte de commande

Afin de pouvoir commander notre moteur pas à pas, l'utilisation d'une carte de commande polyvalente est primordiale, elle doit donc être à la fois, simple (en programmation), intuitive (compréhension du fonctionnement) et surtout petite afin d'occuper le moins d'espace possible, c'est pour cela qu'on opte pour l'utilisation d'une carte Arduino MEGA. C'est une carte basée sur le microcontrôleur ATmega2560.

Il en existe plusieurs types de carte Arduino, toutes différentes les unes des autres, parmi ces cartes-là, on peut trouver celles qui sont les plus utilisées selon leurs caractéristiques, et qui sont : la Mega, Uno, Duemilanove, Leonardo, Mega Adk et la Nano.

Pour notre projet, comme nous l'avons cité précédemment, notre choix s'est porté sur la carte Arduino MEGA en raison de sa simplicité de fonctionnement et surtout du nombre de broches disponibles.

Afin de pouvoir éclaircir plus le fonctionnement de notre carte, nous présentons ses différentes caractéristiques qui sont les suivantes :

i) Présentation du matériel

Il s'agit d'une carte électronique sous forme d'un support plan, flexible ou rigide, généralement composé d'époxy ou de fibre de verre. Elle possède des pistes électroniques disposées sur une, deux ou plusieurs couches (en surface et/ou en interne) qui permettent la mise en relation électronique des composants électroniques. Chaque piste relie tel composant à tel autre, de façon à créer un système électronique qui fonctionne et qui réalise les opérations demandées. Cette carte est basée autour d'un microcontrôleur Atmega du fabricant Atmel de référence ATMega 2560.

L'ATmega2560 est un microcontrôleur de la famille AVR dont la programmation peut être réalisée en langage C. La figure 2.9 ci-dessous nous montre ses différents composants. [12]

La carte Arduino MEGA 2560 est basée sur un ATMega2560 cadencé à 16 MHz. Sa mémoire de 256KB et son grand nombre d'E/S font de ce circuit compatible DIL30 un

Élément idéal pour les systèmes embarqués ou pour des applications robotiques nécessitant du multitâches.

La MEGA 2560 peut se programmer avec le logiciel Arduino. Le contrôleur ATMega2560 contient un bootloader de 8 KB qui permet de modifier le programme sans passer par un programmeur

Chapitre II : Conception du système

Caractéristiques principales :

- Alimentation : via port USB ou 5Vcc régulée ou 6 à 20 V non régulée
- Microprocesseur : ATmega2560
- Mémoire flash : 256KB dont 8KB pour le bootloader
- Mémoire SRAM : 8KB
- Mémoire EEPROM : 4KB
- 54 broches d'E/S dont 14 PWM
- Intensité par E/S: 40 mA
- 16 entrées analogiques
- Horloge : 16 MHz
- Bus série, I2C et SPI
- Gestion des interruptions
- Fiche USB : mini-USB B
- Dimensions : 18,54 x 43,18 mm

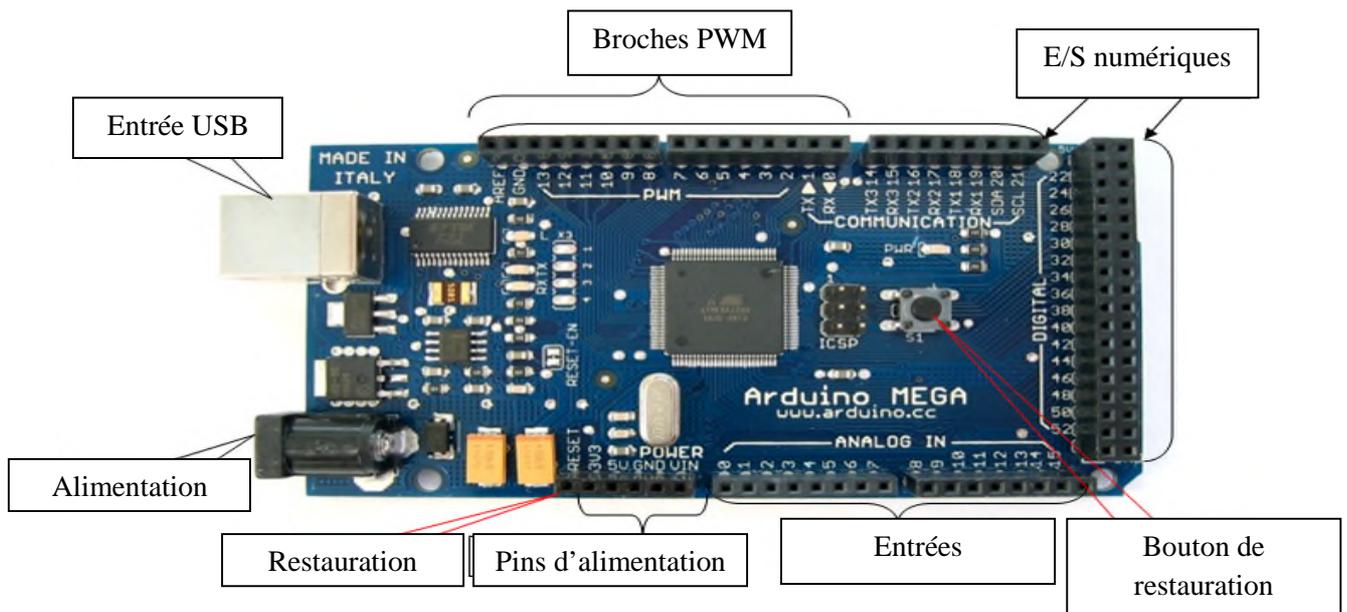


Fig. 2.9 : Carte Arduino MEGA 2560 (vue de face)

ii) Présentation du logiciel

Le logiciel de programmation des modules Arduino est une application Java, libre et multiplateforme, servant d'éditeur de code et de compilateur, et qui peut transférer le firmware et le programme au travers de la liaison série (RS-232, Bluetooth ou USB selon le module). Il est également possible de se passer de l'interface Arduino, et de compiler et uploader les programmes via l'interface en ligne de commande.

Le langage de programmation utilisé est le C++, compilé avec avr-g++, et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes Arduino, à toute personne maîtrisant le C ou le C++.

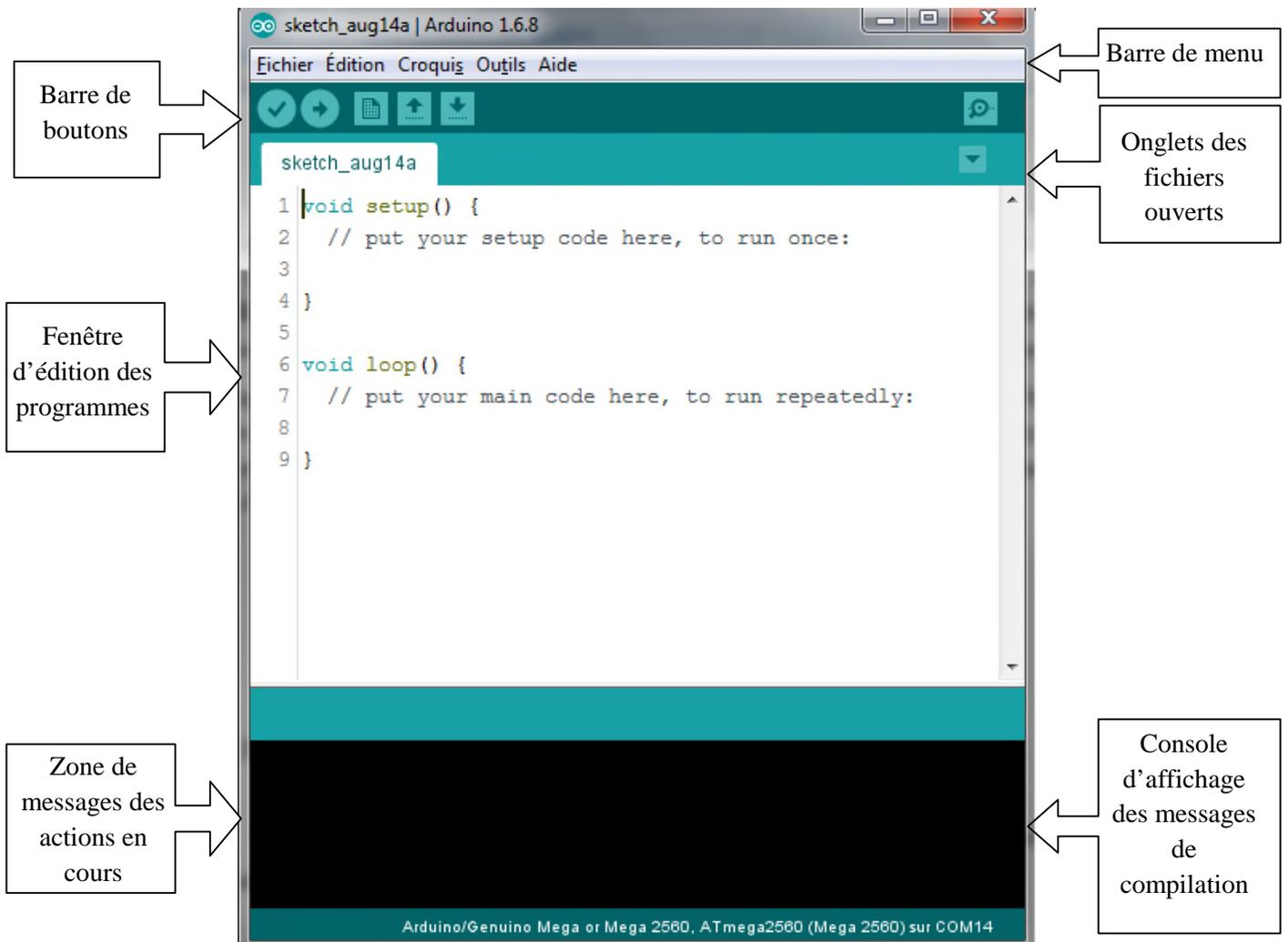


Fig. 2.10 : Logiciel Arduino

2.3.5. Capteurs

Les capteurs utilisés sont les suivants : jauge de contrainte, capteur de position potentiométrique, capteur switch.

i) **Jauge de contrainte**

Elle repose sur le principe d'un fil que l'on déforme. Le fil, très fin, est placé préférentiellement de manière longitudinale par rapport à la déformation comme illustré dans la figure 2.11. En agissant par traction ou compression sur le fil, celui-ci délivre une certaine valeur de résistance par rapport à son état repos. Cette variation de longueur modifie la résistance électrique du fil. On mesure alors cette variation de résistance entre l'état repos et l'état sous contrainte. Dans notre cas, elle est utile pour détecter la surpression au niveau du piston. [10]

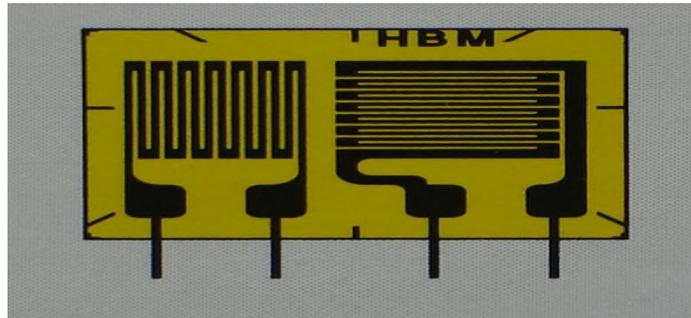


Fig.2.11 : Jauge de contrainte.

ii) Capteur de position potentiométrique

Le capteur de position potentiométrique est constitué d'une piste résistive et d'une piste conductrice séparée par une lame d'air, l'ensemble ne dépassant pas 2 mm d'épaisseur, la figure 2.12 nous illustre les composants de cette bande. Utilisé en diviseur de tension, une simple pression d'environ 1 N, au doigt ou grâce à un curseur, permet d'obtenir une information de position avec une résolution quasi-infinie. Son support autocollant le rend très facile à mettre en œuvre. Grâce à son étanchéité, Il peut être intégré dans de nombreuses applications même sévères, ou la propreté est une priorité. [9]

L'utilisation de ce capteur est donc adéquate à notre réalisation afin de pouvoir déterminer la position du piston de la seringue avec exactitude.



Fig. 2.12 : Capteur de position potentiométrique

iii) Capteur switch

Afin de pouvoir détecter la présence d'une seringue sur son support de fixation, et de détecter si ce dernier est sur sa position zéro qui est la position d'initialisation, nous avons utilisé des switch comme illustré dans la figure 2.13. [11]

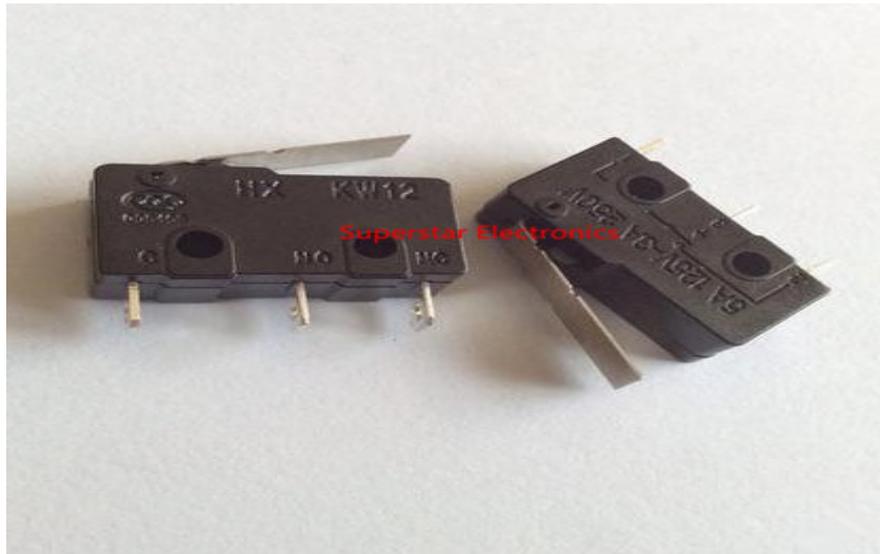


Fig.2.13 : Un switch

CHAPITRE III

Réalisation du système

²Chapitre 3

Réalisation du système

La réalisation de notre système se compose de deux parties principales, une partie matérielle et une partie logiciel que voici :

3.1. Partie matérielle

Cette partie est essentielle au bon déroulement de notre réalisation, car c'est dans celle-ci qu'on présente le circuit. Ce dernier rassemble tous les éléments de notre système tel que la carte de commande, l'afficheur et les différents boutons. Pour pouvoir faire fonctionner tous ces composants et permettre une bonne rotation du moteur pas à pas, nous avons dimensionné une alimentation adéquate pour fournir les courants et les tensions nécessaires.

3.1.1. Microcontrôleur

Le microcontrôleur qui équipe notre Arduino Mega est le ATmega 2560, ici nommé U1, les différentes liaisons de celui-ci vers les autres périphériques tel que les interfaces et le moteur sont illustrés dans la figure 3.1 suivante :

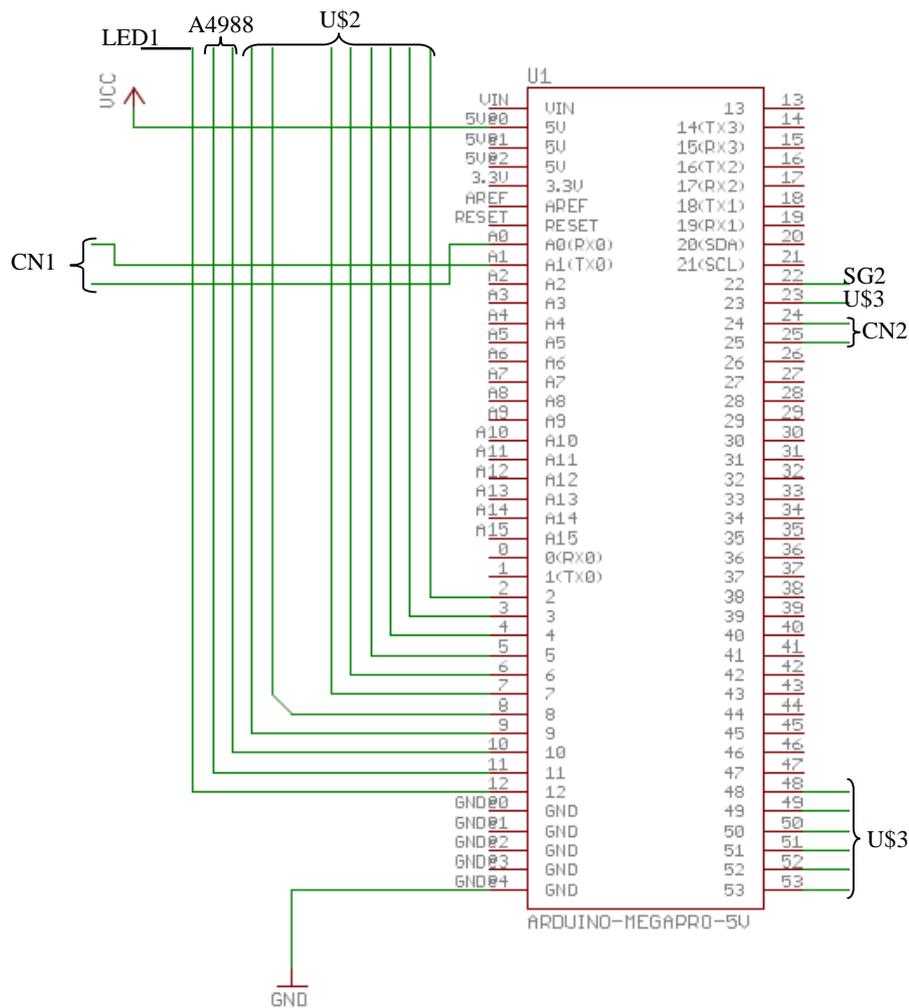


Fig. 3.1 : Schéma de branchement de l'Arduino Mega

3.1.2. Commande moteur

Le module A4988 qui sert à commander le moteur pas à pas dont sa partie logique est alimentée en 5V et aussi doté d'une alimentation allant de 8V à 35V est d'un courant pas phase (continu) allant de 1.2A à 2.3A destinée au différent bobinage du moteur pas à pas, le branchement de ce module est comme suite :

- Entrées :
- step (pas) :10
 - Dir (direction) :11
 - Sleep :Reset

Sorties : -A1 et B1 : alimente la première bobine
 -A2 et B2 : alimente la deuxième bobine

Ces branchements sont représentés dans la figure 3.2 suivante :

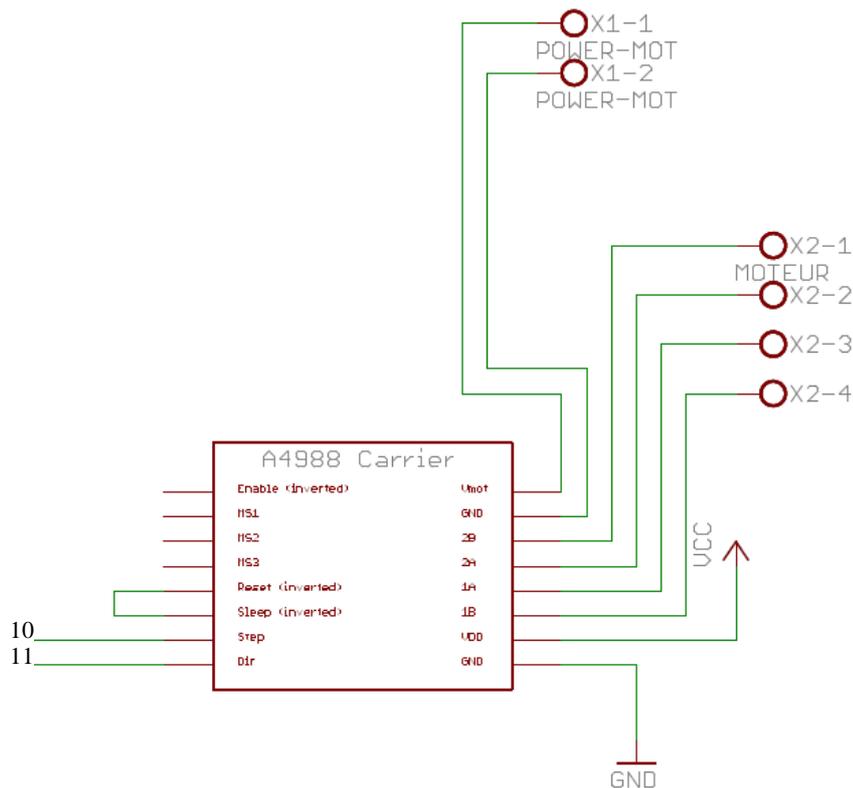


Fig. 3.2 : Schéma de branchement du A4988 et moteur pas à pas

3.1.3. Interface

Les différentes interfaces utilisées ainsi que leurs différents branchements sont les suivant :

i) Afficheur

L'afficheur utilisé U\$3 est un HD44780 LCD, il est alimenté par Vcc et ses branchements vers l'Arduino sont les suivant :

-DB4, DB5, DB6, DB7: 50, 51, 52, 53

-RS, E: 48, 49

Ces différents branchements sont représentés dans la figure 3.3 suivante :

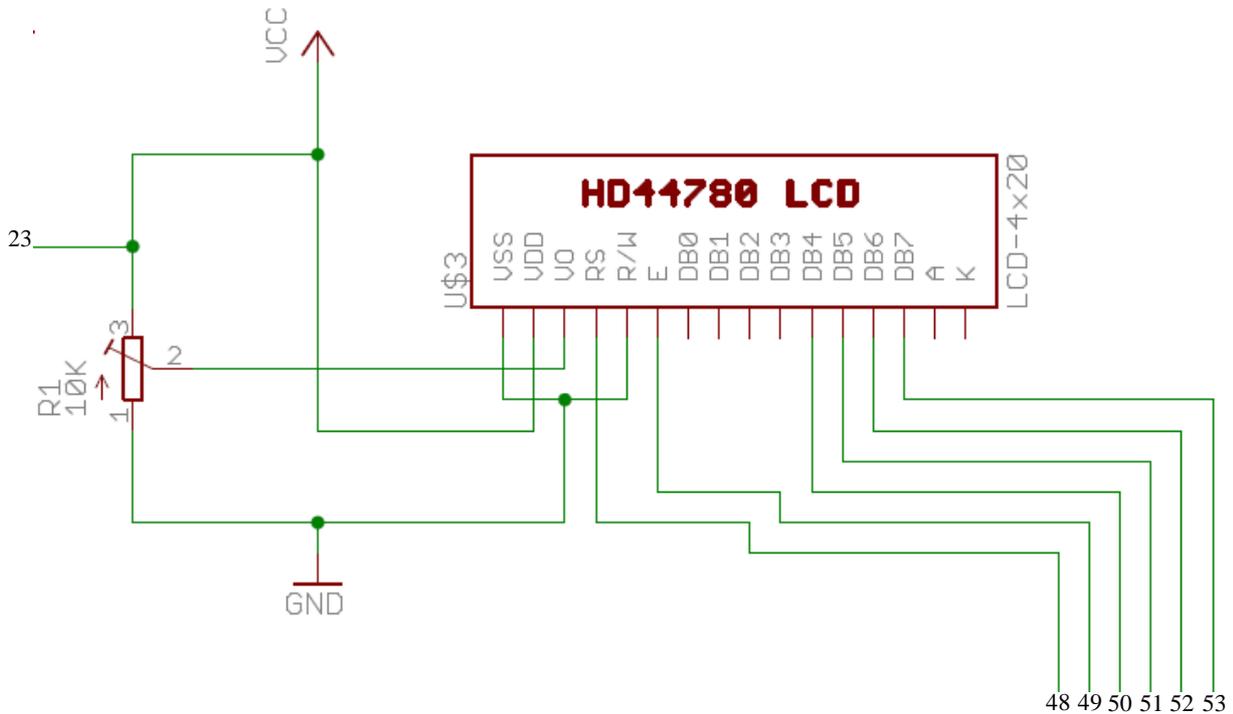


Fig. 3.3 : Branchement de l'afficheur LCD

ii) Clavier :

Le clavier matriciel 4x4 nommé U\$2 est branché comme suit :

1 à 8 vers entrée Arduino 2 à 9

Ces différents branchements sont représentés dans la figure 3.4 suivante :

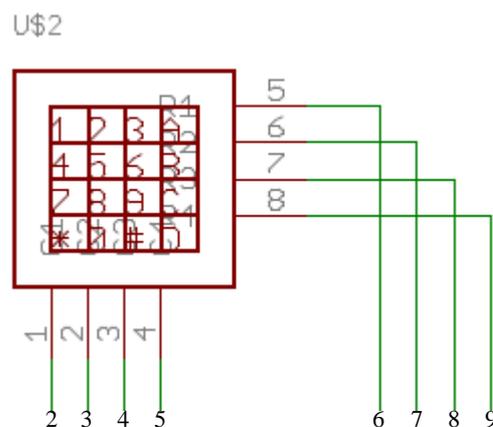


Fig. 3.4 : Branchement du clavier matriciel 4x4

iii) Buzzer et LED :

- Le Buzzer ou alarme ici nommé SG2, est relié à la broche 22.
 - La LED ici nommé LED1 clignote à chaque fois qu'il y'a un problème dans l'appareil et cela afin de prévenir l'utilisateur, cette LED est relié à la broche 12.
- Leurs différents branchements sont représentés dans les figures 3.5 et 3.6 suivantes :

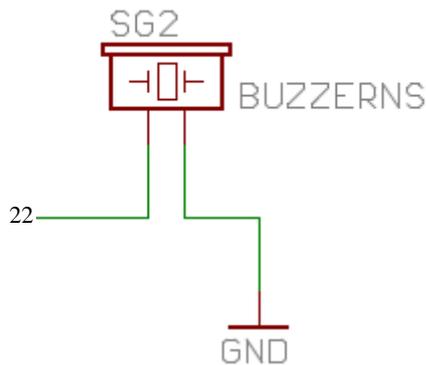


Fig. 3.5 : Branchement du Buzzer

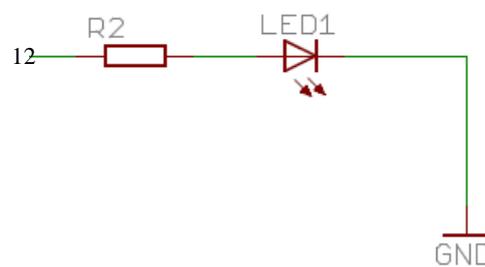


Fig. 3.6 : Branchement de la LED

3.1.4. Capteur :

- Le capteur de position : CN1, branché à l'entrée de l'Arduino A0 et A1, ces branchements sont représentés dans la figure 3.7.
- Capteur de présence (fin de course) : CN2, relié à l'Arduino par les broches 24 et 25, représenté dans la figure 3.8.

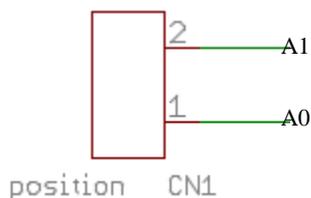


Fig. 3.7 : Branchement de CN1

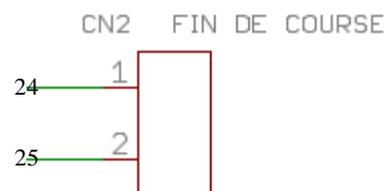


Fig. 3.8 : Branchement de CN2

3.1.5. Alimentation :

Pour décortiquer le circuit d'alimentation de la figure 3.17, on s'est concentré sur les blocs importants.

Bloc de redressement : notre conception est alimentée par un bloc d'alimentation sous tension secteur (220V, 50Hz), puisque le circuit électronique fonctionne en continu, ce qui impose aux circuits d'alimentation un système de redressement de tension.

Une alimentation autonome : vu que le système doit finir sa fonction (perfusion) en toute condition, l'utilisation d'une batterie est plus que nécessaire. Il y a toujours la présence d'un premier condensateur relié directement à la sortie du redresseur, appelé condensateur Tampon, il empêche l'alternance redressée de descendre à zéro volt (0V), cela augmente la tension continue moyenne et diminue par conséquent la tension d'ondulation résiduelle.

Une tension stable : pour stabiliser notre alimentation, on a utilisé un régulateur de tension qui est le circuit intégré (IC1. 7805), et un ensemble de condensateurs C1 et C2 dont leurs valeurs respectives sont de 220 μ F et 47 μ F et cela afin de réduire les interférences et ondulations (bruits).

Cette alimentation est représentée dans la figure 3.9 suivante :

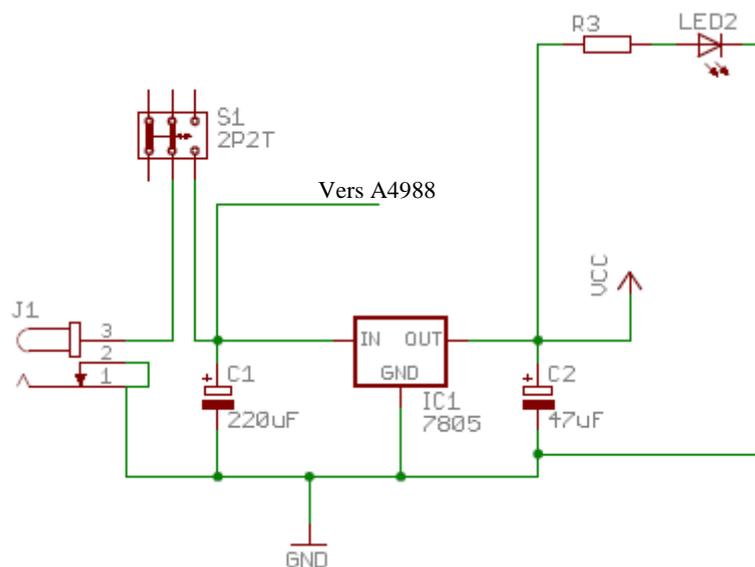


Fig. 3.9: Circuit d'alimentation

3.2. Partie logicielle

L'organigramme de la figure 3.10 constitue la base de notre programme principal ainsi que les différentes parties qui permettent de réaliser la commande de notre système.

Pour permettre le bon déroulement du programme, nous avons divisé notre organigramme en différents blocs.

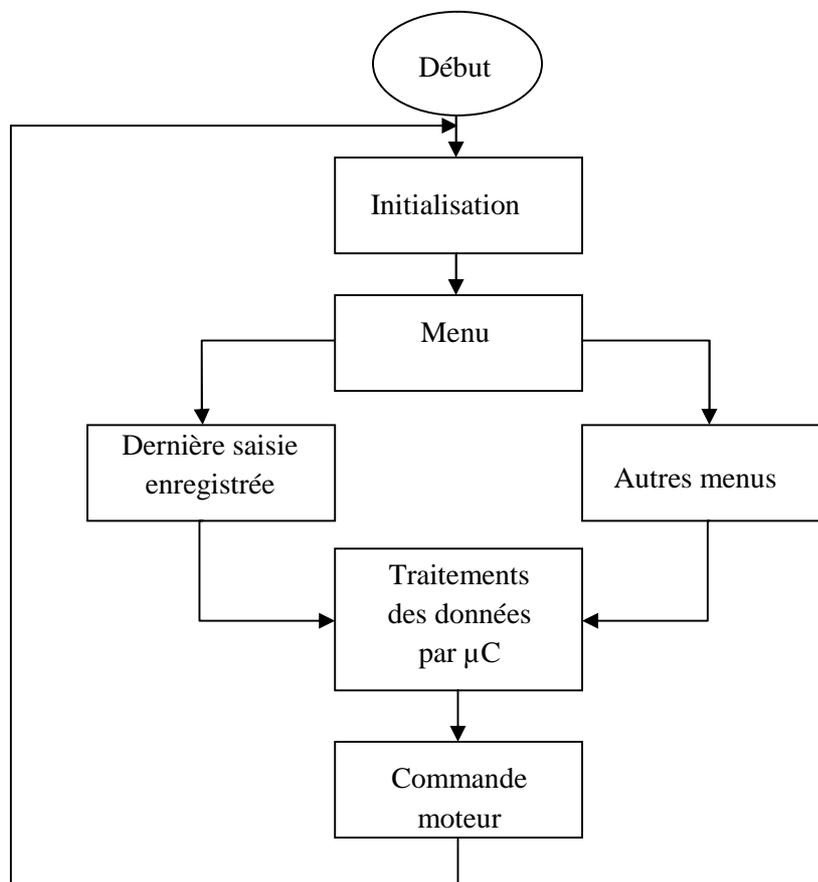


Fig.3.10 : Organigramme général du système

Chapitre III : Réalisation du système

En décortiquant les différents blocs de l'organigramme précédent, on interagit avec différents boutons et différents capteurs dont notre appareil se compose ; chaque bouton et chaque capteur a un rôle spécifique et sont défini comme suit :

- **Valider** : Sert à valider ou à confirmer une saisi ou une donnée.
- **Annuler** : Sert à annuler une donnée ou une saisie erronée.
- **Menu** : Pour entrer dans le menu.
- **Marche** : Pour faire marcher le moteur ou à relancer le programme en cas d'une pause.
- **Pause** : Pour faire arrêter le programme et le mettre en pause.
- **Arrêt** : Arrêt d'urgence en cas de problème dans l'appareil ou avec le patient.
- **FC** : Fin de Course, sert à connaitre si l'écrou est revenu à sa position initial ou pas
- **CP** : Capteur de Position, c'est une bande résistive situé sous l'écrou qui est doté d'un curseur, en appuyant avec ce dernier sur la bande, celle-ci détermine la position de l'écrou et nous donne sa position exacte.
- **PM** : Position Moteur, les pas que le moteur pas à pas effectue sont contrôlés par le microcontrôleur, cela nous donne sa position exacte selon le nombre de pas effectué.
- **PS** : Position Seringue, switch placé sous la seringue servant à détecter si celle-ci est positionnée ou pas.
- **BT 0, ...9** : Bouton de 0 jusqu'à 9 ; afin de pouvoir saisir les différents paramètres, notre appareil est muni d'un clavier numérique.

3.2.1. Initialisation

Dans ce bloc on voit les différents paramètres pris en charge par le programme afin d'effectuer l'initialisation du système et de permettre le bon fonctionnement de l'appareil.

Juste après l'allumage de l'appareil, le système effectue une vérification de la position du poussoir auquel est fixé le piston de la seringue et les différents capteur, d'abord le poussoir avance jusqu'à la fin de la vis sans fin pour vérifier le capteur de position et revient au début jusqu'au fin de course; après cela, le poussoir est remis à la position zéro si nécessaire, et enfin affiche si une seringue est bien placé sur l'appareil ou pas ; l'organigramme de la figure 3.1 Ici-dessous nous montre cette étape :

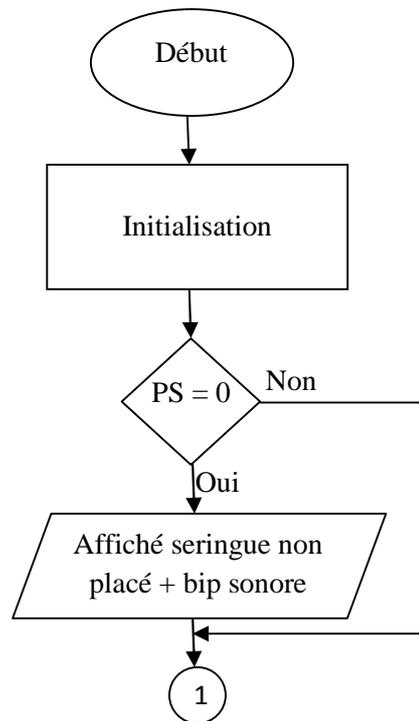


Fig.3.11 : Organigramme de l'initialisation

Le symbole ○ revoie à la suite de l'organigramme.

3.2.2. Menu

Dans cette partie qui se divise en deux, on verra les différents menus auxquelles l'utilisateur aura à faire et les différentes fonctionnalités disponibles.

i) Dernière saisie enregistrée

Juste après le démarrage de l'appareil, en cas d'une utilisation fréquente ou répétée d'une seule valeur, celle-ci s'affiche sur l'écran de l'appareil comme représentée dans la figure 3.4 qui correspond au premier menu ; ainsi, l'opérateur ou l'infirmier peut l'utilisée directement sans avoir à modifier les valeurs ou à interagir à chaque fois avec l'appareil. L'opérateur a donc juste à appuyer sur le bouton « Valider » pour la validation de cette valeur et commencé la perfusion comme démontré dans l'organigramme de la figure 3.12.

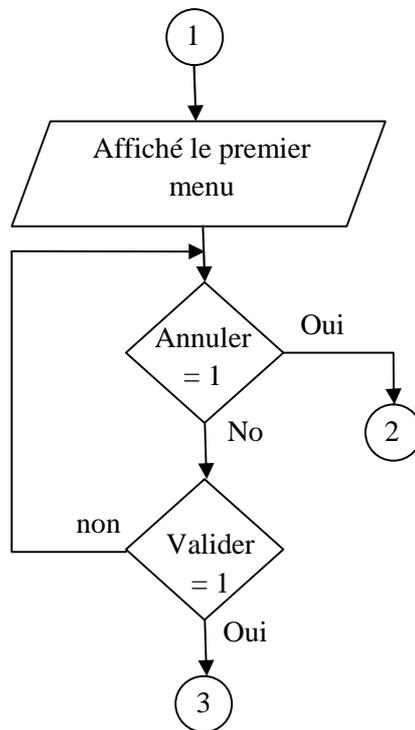


Fig. 3.12 : Organigramme du menu de la dernière saisie enregistré

La figure suivante correspond au premier menu qui affiche différents éléments dont la date et l'heure situées en haut de l'afficheur, l'ancienne valeur utilisée à la deuxième et troisième ligne et enfin les boutons disponibles à l'appui pour l'annulation de la valeur ou de sa validation.

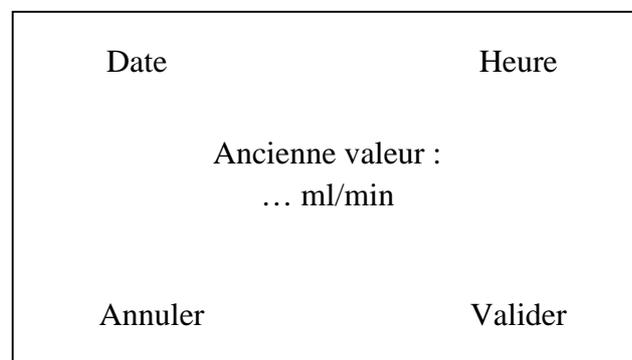


Fig.3.13 : Premier menu

ii) Autres menus

Si l'opérateur annule l'ancienne valeur enregistrée, alors un deuxième menu s'affiche, celui-ci fera partie d'un ensemble de menus qui constituera ce bloc.

Ce deuxième menu est aussi constitué de quatre lignes comme illustré dans la figure 3.14, la première affichera la date et l'heure. La deuxième, en appuyant sur le bouton « 1 », l'opérateur pourra modifier la date et l'heure, la troisième, en appuyant sur le bouton « 2 », l'opérateur pourra saisir les données nécessaires pour la perfusion, et la quatrième ligne affiche les boutons disponibles dont le bouton « 1 » et « 2 ».

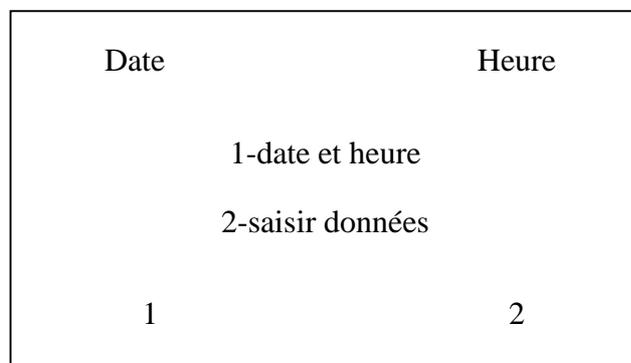


Fig. 3.14 : Deuxième menu

En appuyant sur le bouton « 2 », l'opérateur pourra voir affiché un troisième menu dont l'affichage sera aussi constitué de quatre lignes, la première affichera la date et l'heure comme les menus précédents, la deuxième et troisième, des valeurs préenregistrées tel que « 10ml/h » pour le bouton « 1 » et « 20ml/h » pour le bouton « 2 », et enfin la troisième ligne affichera les autres données à saisir en appuyant sur le bouton « 3 » comme illustré dans la figure 3.15 ci-dessous :

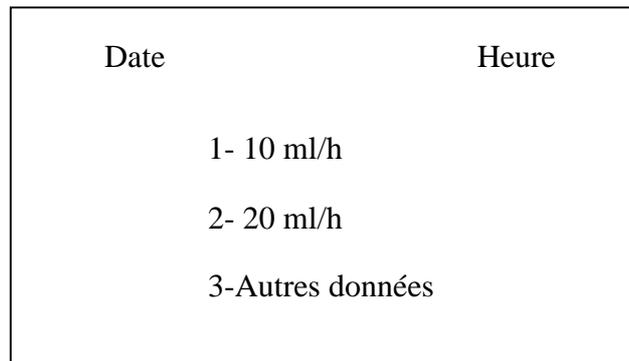


Fig. 3.15 : Troisième menu

Si l'opérateur souhaite faire une perfusion avec une valeur préenregistrée, il suffira qu'il appui sur le bouton correspondant à celle-ci, sinon, si une autre dose doit être administré, l'opérateur sélectionnera les autres données à saisir en appuyant sur le bouton « 3 », cela lui affichera un quatrième menu représenté dans la figure 3.16 ci-dessous :

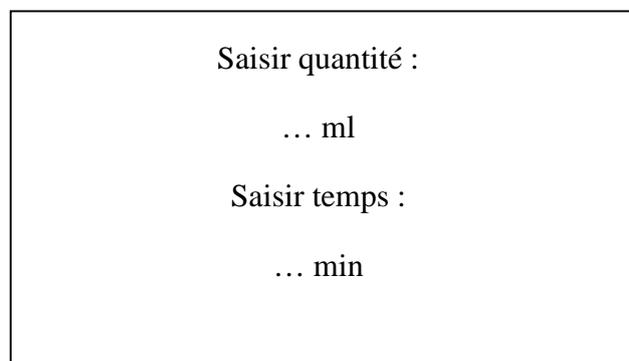
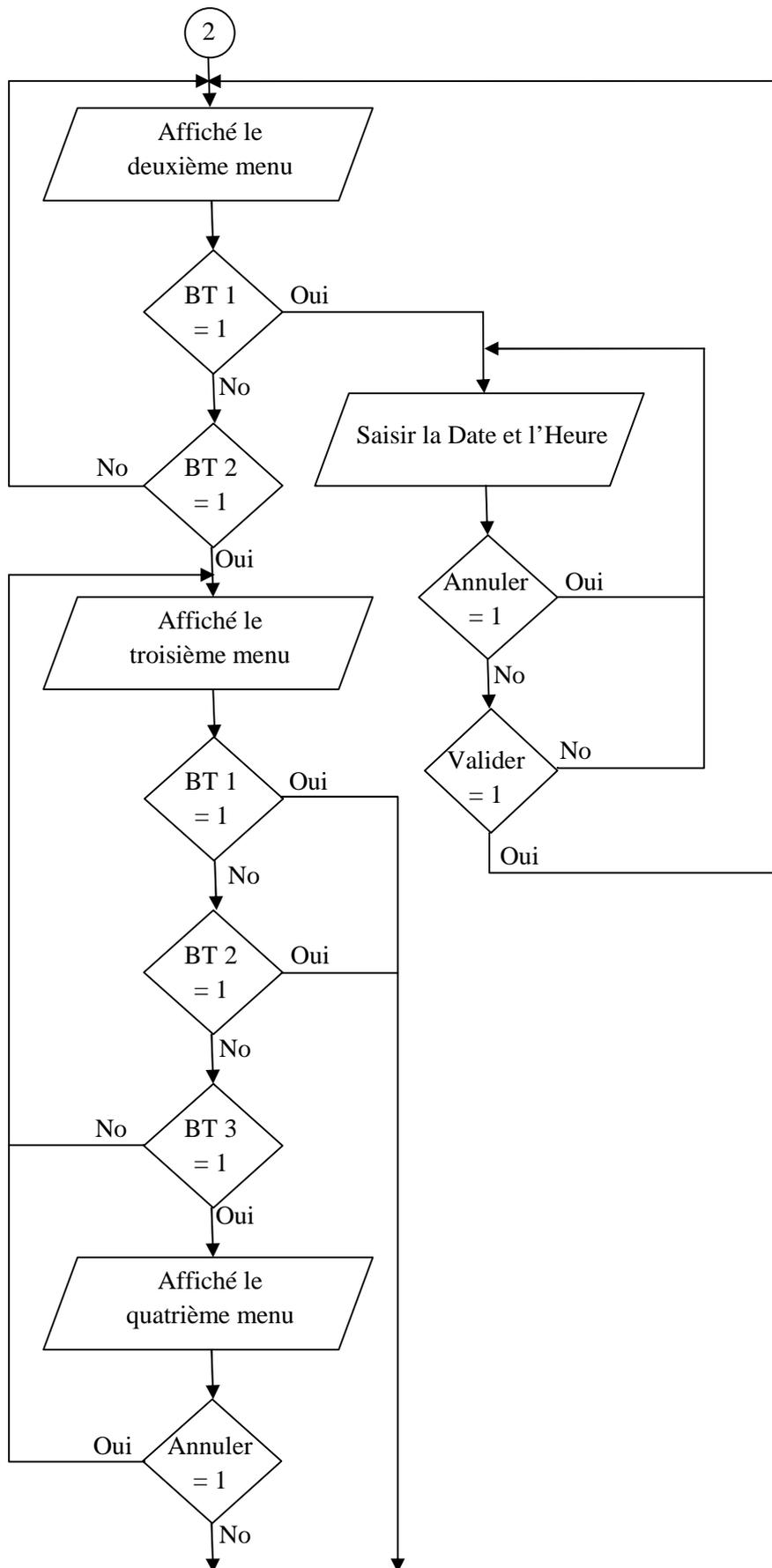


Fig. 3.16 : Quatrième menu

Le bloc des autres menus de l'organigramme qui prend en charge cet affichage est détaillé dans l'organigramme de la figure 3.17 suivante :



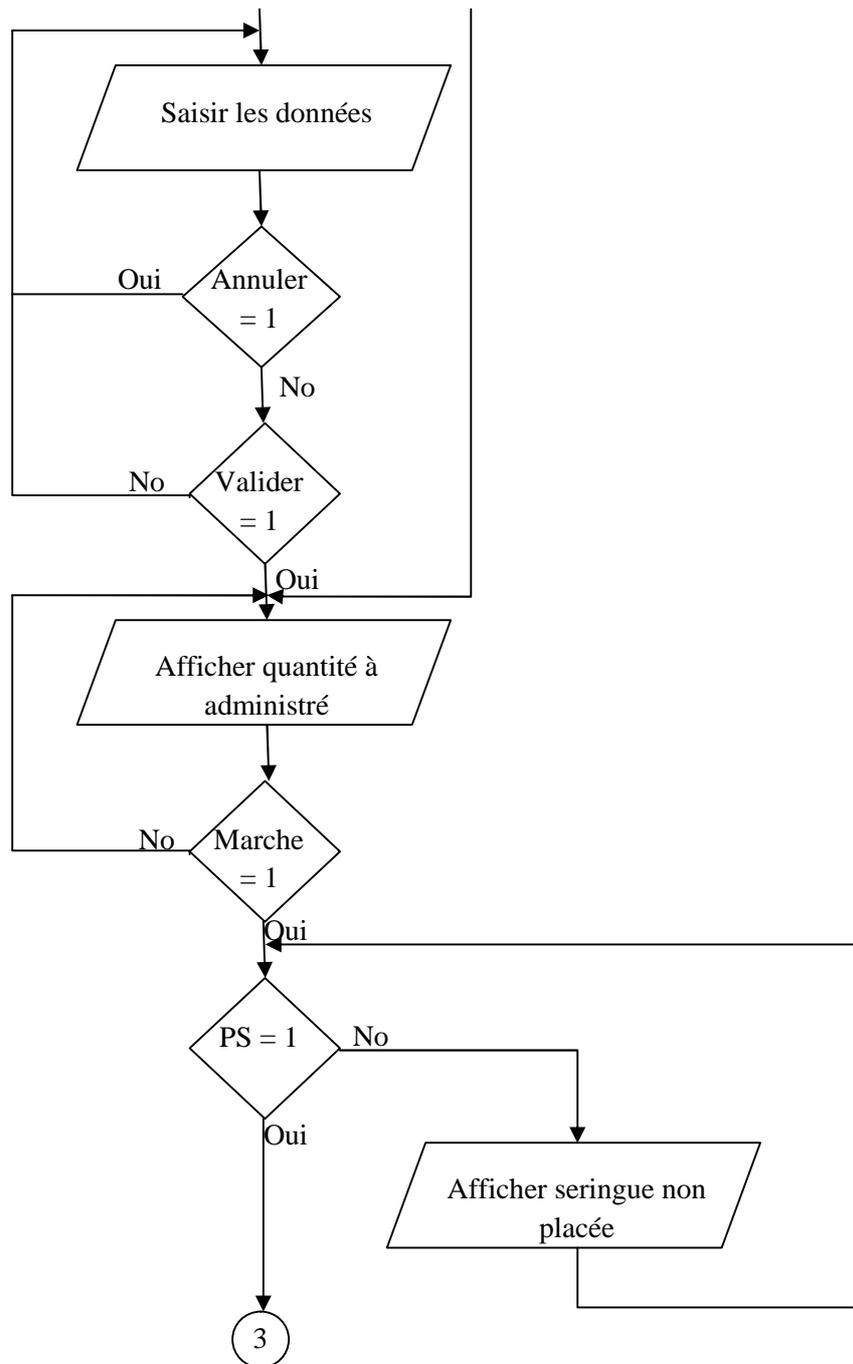
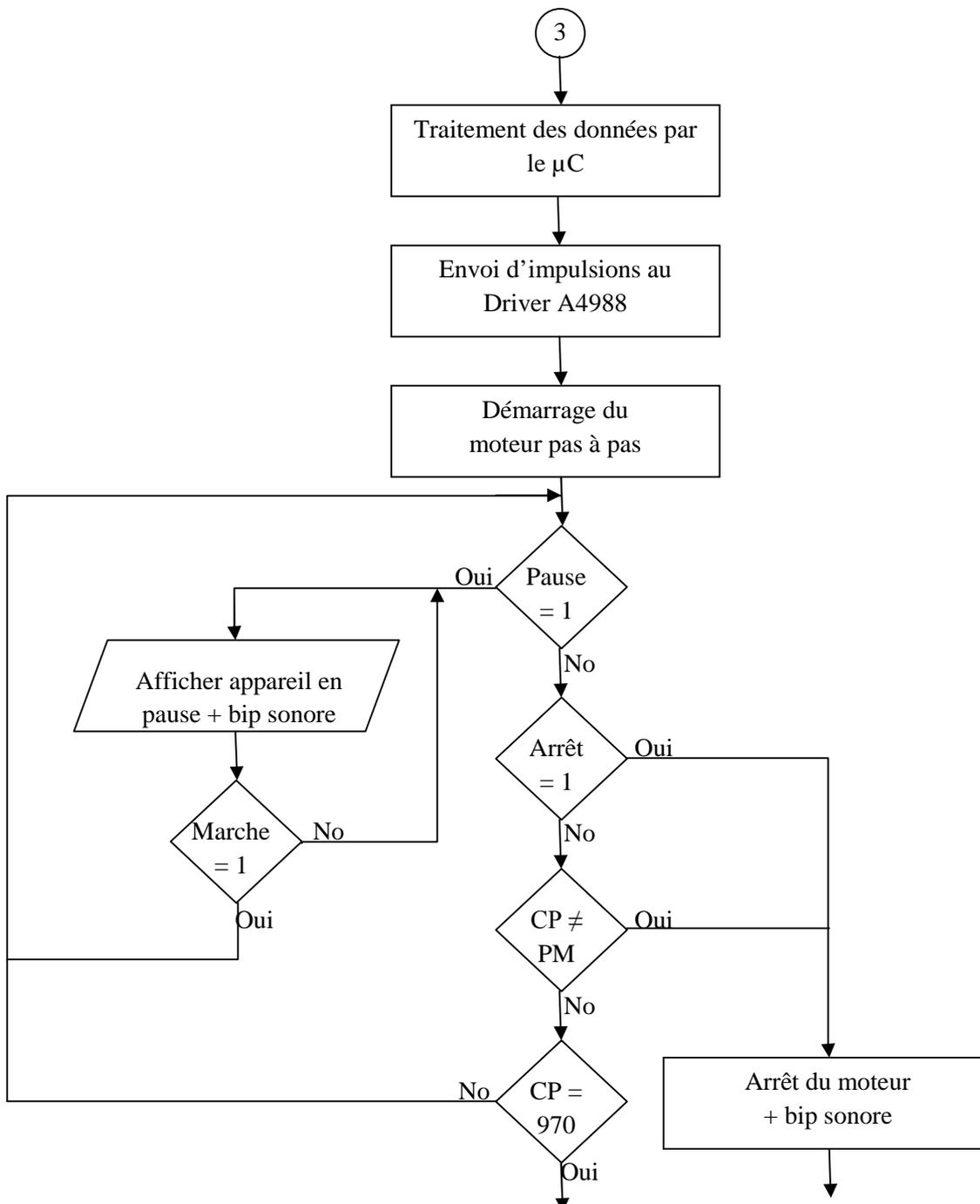


Fig. 3.17 : Organigramme des autres menus

Après que l'opérateur ait pu saisir les différents paramètres nécessaires à la perfusion, il suffira que celui-ci, après affichage de la valeur saisie, appuie sur le bouton « Marche » pour que les données soient transférées au microcontrôleur (μC) ; mais juste avant cela, le système vérifie si une seringue est placée sur l'appareil à l'aide d'un bouton poussoir, sinon, si celui-ci n'est pas appuyé, un affichage indiquera que la seringue est non placée ; l'appareil ne pourra ensuite démarrer que si une seringue est bien positionnée sur son support.

3.2.3. Traitement des données par le μC et commande du moteur

Après l'introduction des différentes données nécessaires à la perfusion, celle-ci seront envoyées vers le μC pour leurs traitements et ainsi, des impulsions seront envoyées vers le Driver A4988 qui enverra à son tour des impulsions électriques afin de contrôler la direction et le nombre de pas que le moteur pas à pas devra faire ; cela est représenté dans l'organigramme de la figure 3.18 suivante :



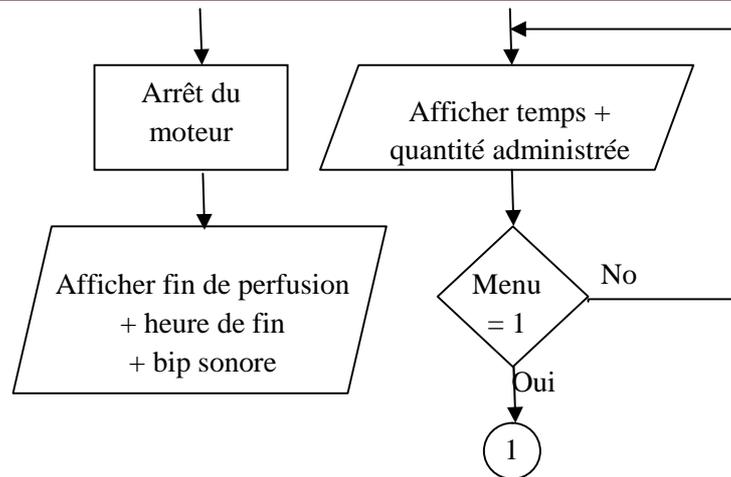


Fig. 3.18 : Organigramme de traitement par le µC et la commande du moteur

3.2.4. Organigramme complet

C'est à partir de ces différents organigrammes qu'on peut faire notre programme et ainsi lancer la réalisation de notre système, mais avons cela, voyons l'organigramme complet qui représente notre programme, celui-ci est représenté dans la figure 3.19 de la page suivante :

3.3. Réalisation du système

Pour pouvoir bien assembler les différents composants de notre système, nous avons fait un schéma de principe représenté dans la figure 3.20 qui illustre les différents éléments et comment leurs assemblages a été fait.

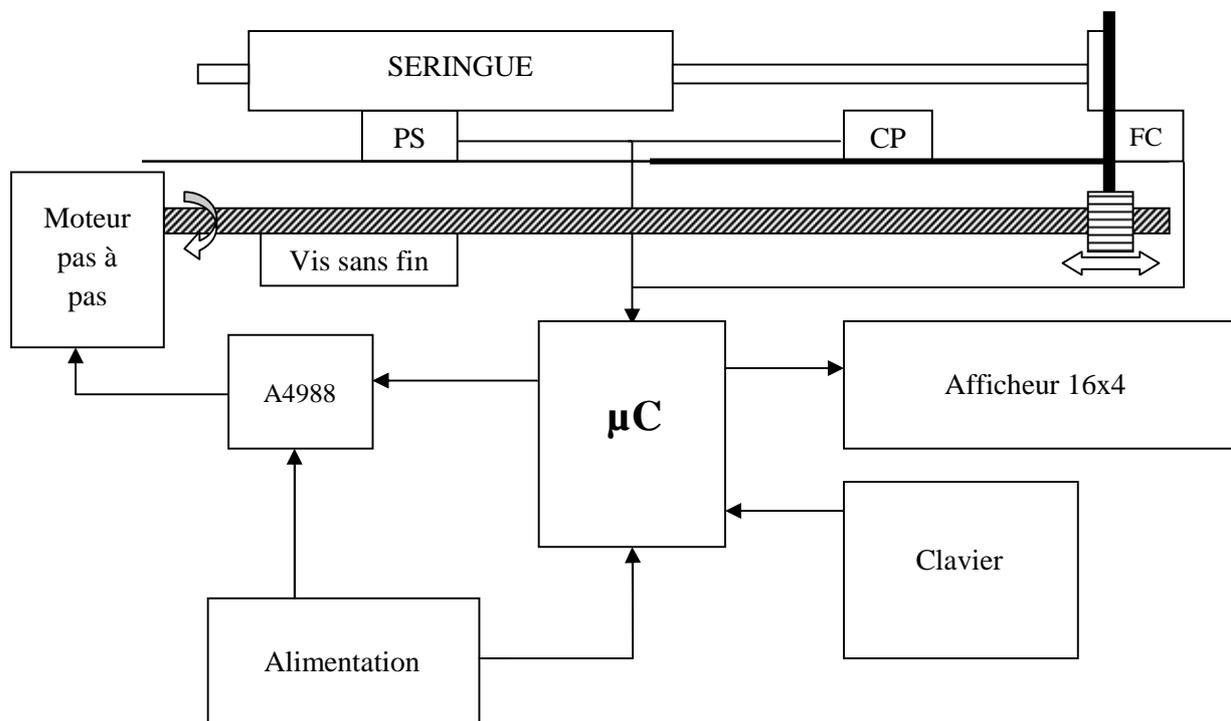


Fig. 3.20 : Schéma de principe

Après avoir fait le circuit du système, qui est constitué de deux faces, on a réalisé son typon dont les deux faces dessus et dessous sont respectivement représentées dans la figure 3.21 et la figure 3.22 suivante :

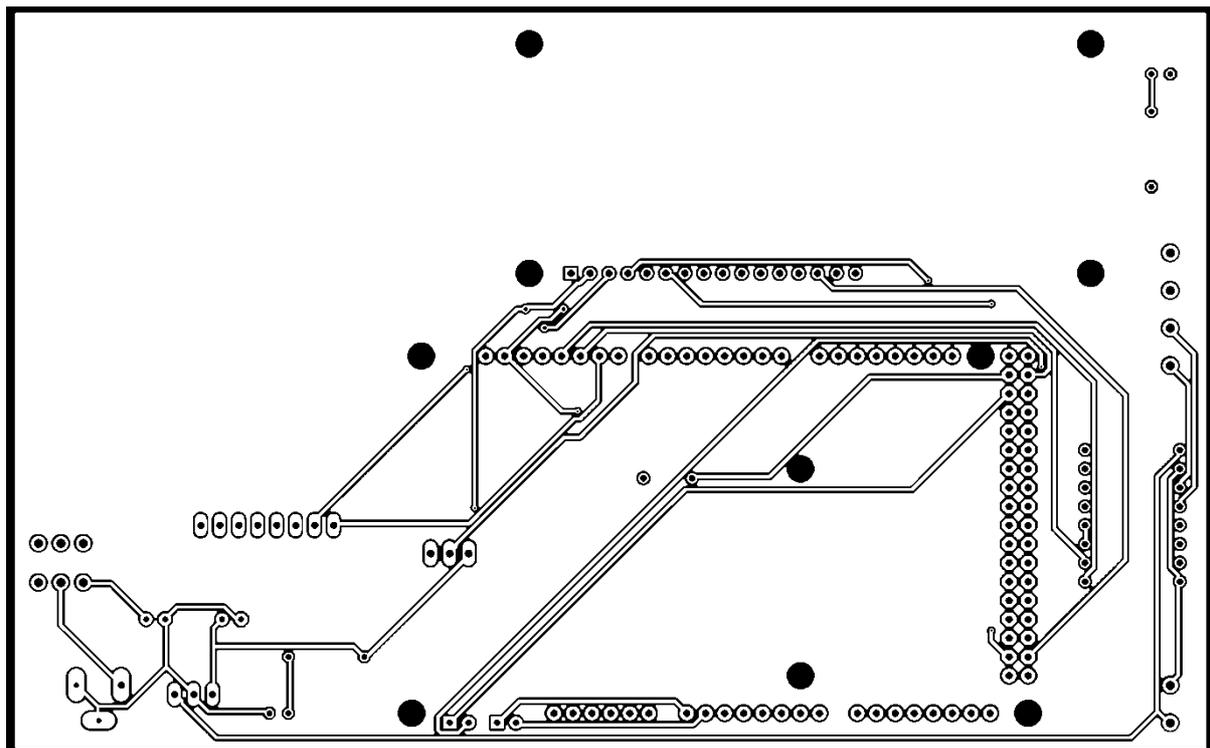


Fig. 3.21 : Typon de la face du dessus

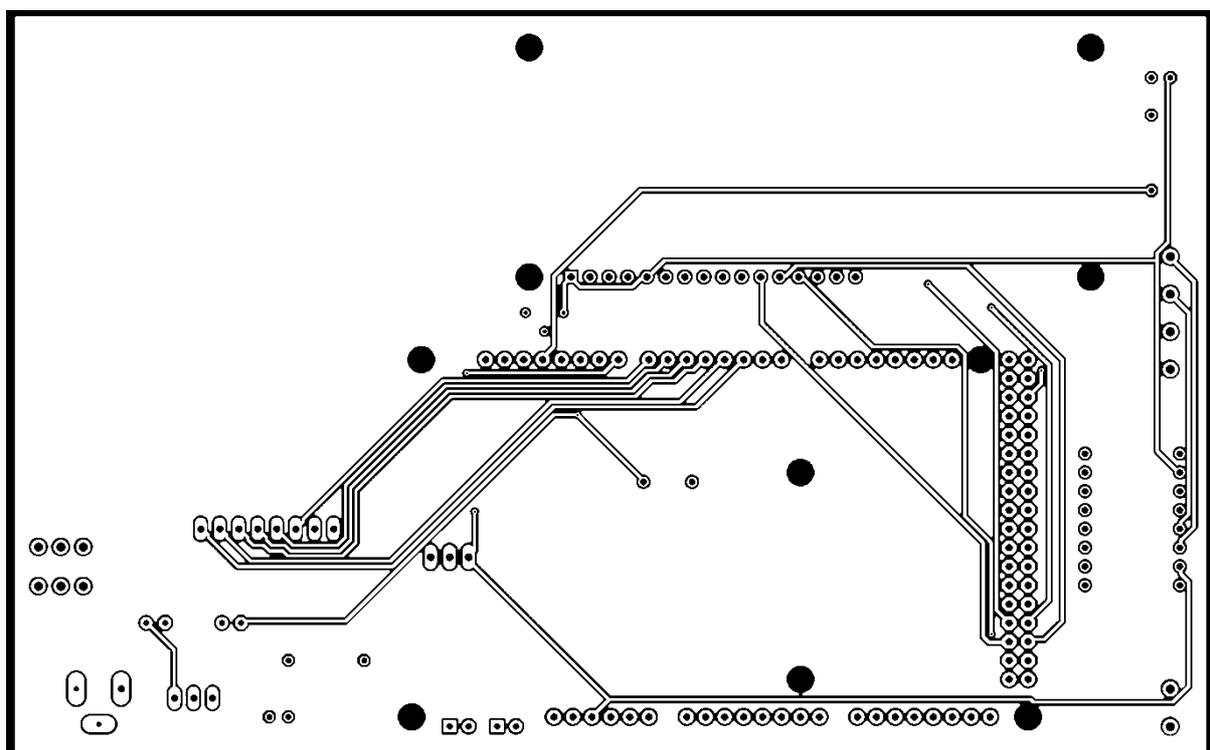


Fig. 3.22 : Typon de la face du dessous

Chapitre III : Réalisation du système

La réalisation de notre circuit imprimé double face est faite à partir de ces deux typons. Les différents composants de notre système sont reliés directement vers ce circuit à travers des connecteurs dont l'Arduino Mega se fixe directement en dessous du circuit.

Suite à une batterie de tests, on a opté pour un ensemble de paramètres, parmi eux, deux fonctions qui sont utiles pour programmer la rotation du moteur ainsi que le déplacement du piston :

- Conversion du nombre de pas par le clavier en millilitre

Une seringue de 20 CC == 6.4 centimètre.

1 millilitre = 1 centimètre cube.

1 CC == 0.32 centimètre.

Implique que : 1 millilitre == 1958 pas.

Note : les paramètres sont calibrés par rapport aux dimensions de la seringue.

- Comme il a été expliqué dans le chapitre précédent, le capteur est linéaire. Aussi la valeur du capteur de position qui est donné en millivolt et qui doit être calibré et transformée en centimètre puis en nombre de pas, pour ensuite la comparer avec le nombre de pas effectué par le moteur.

La longueur du capteur = 10 centimètre

0 centimètre = 30 millivolt

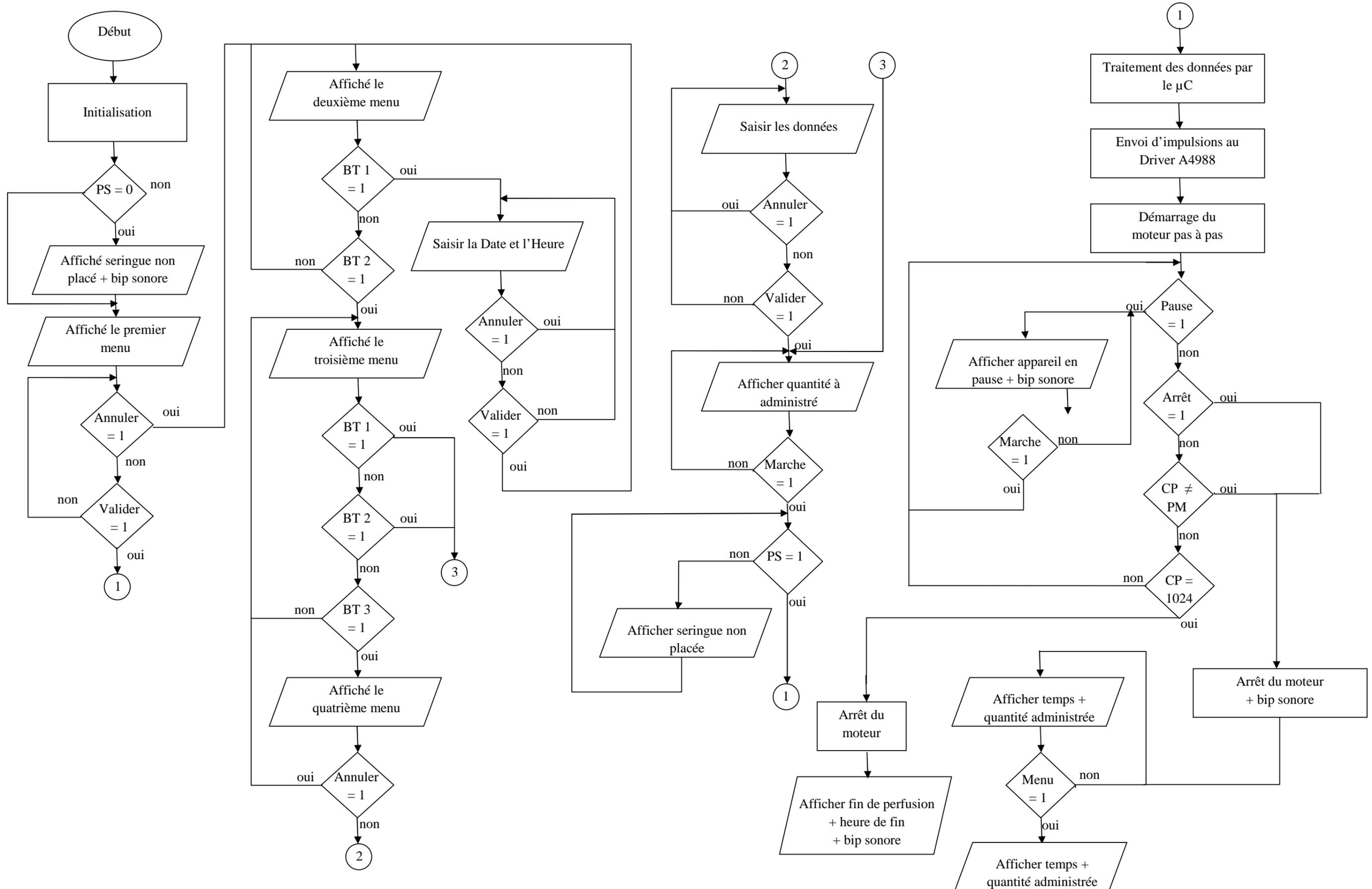
10 centimètre = 1000 millivolt

Alors : 10 centimètre = 970 millivolt

Implique que : 1 centimètre = 97 millivolt

Implique que : 1 millilitre = 1958 pas = 97 millivolt

Chapitre III : Réalisation du système



Conclusion

La disponibilité de la carte ARDUINO MEGA 2560 R3 a permis une conception relativement simple du système de perfusion. Pour une meilleure précision nous avons utilisé un moteur pas à pas pour guider l'injection. Le système réalisé permet un fonctionnement automatique sur de longues durées de temps avec des paramètres d'injection programmables et des sécurités en cas de défaillance. Le système est conçu de sorte à admettre des extensions et améliorations futures. Les tests ont montré le bon fonctionnement du système de perfusion. Toutefois, l'automatisation de notre produit nécessite une supervision humaine.

Quelques améliorations peuvent être apportées à savoir :

- L'utilisation des microcontrôleurs à la place de la carte Arduino pour réduire le coût.
- Pour plus de précision, l'utilisation d'un ARM est plus efficace.
- Création d'un système de surveillance de pression pour une meilleure sécurité.
- L'utilisation des électrovannes pour les besoins des interventions chirurgicales.
- L'utilisation d'un système de surveillance en temps réel des paramètres médicaux lors de la perfusion d'un patient.
- Intégration d'un système de reconnaissance du type de seringue pour une couverture plus importante des perfusions.
- Transmission des données de la perfusion et les alertes à distance pour une meilleure supervision.

Notre projet de fin d'étude nous a permis d'acquérir une expérience supplémentaire et un savoir-faire pour une meilleure prise en charge de conceptions électroniques futures dans le domaine de la perfusion en particulier et le domaine du médical en général.

Enfin, nous espérons que notre modeste travail puisse apporter aide et satisfaction aux futurs utilisateurs.

RÉFÉRENCES
BIBLIOGRAPHIQUES

Références bibliographiques

[1]Georgin D., NatanJ., SzymczakH., 2005 : Perfusion : Etat de l'art, contrôle qualité, *Projet Master MTS, Université de Technologie de Compiègne.*

[2] Lannoy Damien, 2011 : Optimisation de la qualité et de l'efficacité des dispositifs médicaux de perfusion simple et complexe.

[3]LORIFERNE J.F, SAADA M, BONNET F., 1990 : Abords veineux centraux - Techniques en réanimation Masson.

[4]LORIFERNE J.F, SAADA M, BONNET F, 1990 : La voie veineuse périphérique - Techniques en réanimation Masson.

[5]Joaquim Calderon (Pessac), 2004 :Gestion des voies veineuses, Journées d'Anesthésie-Réanimation Chirurgicale d'Aquitaine.

[6]https://fr.wikipedia.org/wiki/Pompe_%C3%A0_perfusion

[7]PATRICE OGUIC : Moteurs pas-à-pas et pc, 2^e édition, *ETSF Edition Techniqueset Scientifiques Française.*

[8]Frank Ebel, Markus Pany, 2011, Référence 571857 : Bases de la technique d'entraînement par servomoteur,

[9]Capteur de position,MEGATRON

:http://www.megatron.fr/composants/position/sensofoil/sensofoil_f.php

[10]Capteur jauge de contrainte, le fournisseur MEGATRON :

<http://www.directindustry.fr/prod/starrett/product-11639-809201.html>

[11]Capteur switch,MEGATRON:<http://www.axesindustries.com/details.php/id/222/capteur-de-position-boitier-de-fins-de-course-programmable.html>

[12]<https://www.arduino.cc/en/Main/ArduinoBoardMega2560#>

ANNEXE

DMOS Microstepping Driver with Translator And Overcurrent Protection

Features and Benefits

- Low $R_{DS(on)}$ outputs
- Automatic current decay mode detection/selection
- Mixed and Slow current decay modes
- Synchronous rectification for low power dissipation
- Internal UVLO
- Crossover-current protection
- 3.3 and 5 V compatible logic supply
- Thermal shutdown circuitry
- Short-to-ground protection
- Shorted load protection
- Five selectable step modes: full, $1/2$, $1/4$, $1/8$, and $1/16$

Package:

28-contact QFN
with exposed thermal pad
5 mm × 5 mm × 0.90 mm
(ET package)



Approximate size

Description

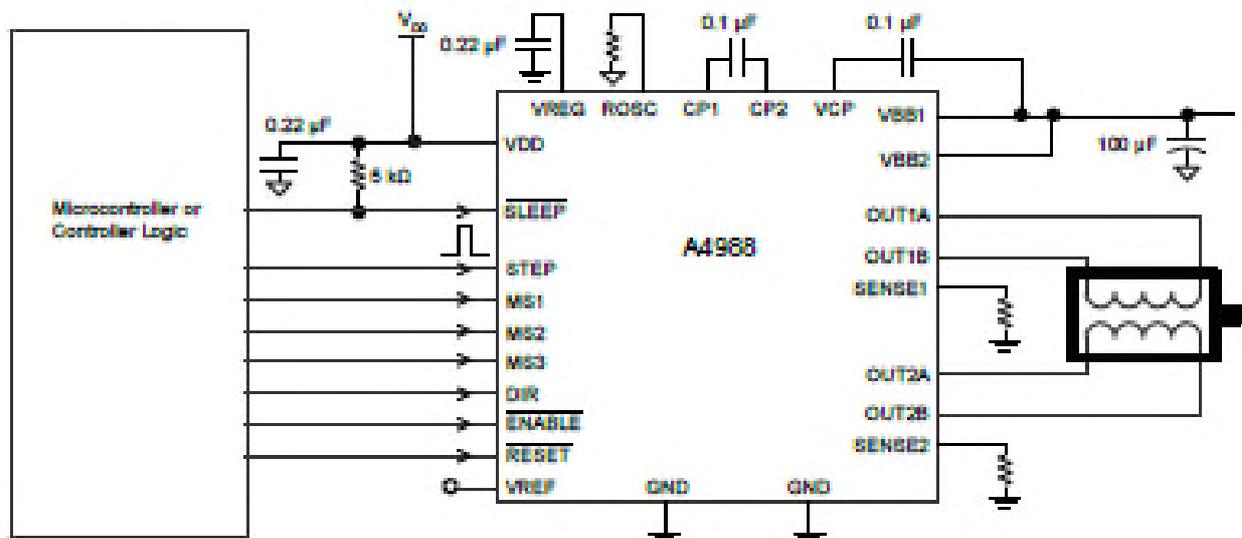
The A4988 is a complete microstepping motor driver with built-in translator for easy operation. It is designed to operate bipolar stepper motors in full-, half-, quarter-, eighth-, and sixteenth-step modes, with an output drive capacity of up to 35 V and ≈ 2 A. The A4988 includes a fixed off-time current regulator which has the ability to operate in Slow or Mixed decay modes.

The translator is the key to the easy implementation of the A4988. Simply inputting one pulse on the STEP input drives the motor one microstep. There are no phase sequence tables, high frequency control lines, or complex interfaces to program. The A4988 interface is an ideal fit for applications where a complex microprocessor is unavailable or is overburdened.

During stepping operation, the chopping control in the A4988 automatically selects the current decay mode, Slow or Mixed. In Mixed decay mode, the device is set initially to a fast decay for a proportion of the fixed off-time, then to a slow decay for the remainder of the off-time. Mixed decay current control results in reduced audible motor noise, increased step accuracy, and reduced power dissipation.

Continued on the next page...

Typical Application Diagram



MEGA 2560 R3 Datasheet



■ INTRODUCTION

MEGA 2560 R3 is an Arduino compatible board, It's based on Arduino MEGA 2560 design, So you can use BUONO MEGA 2560 as Arduino MEGA 2560, All Code, shield and IDE for Arduino MEGA 2560 R3 are also valid on BUONO MEGA 2560, Some visible improvement on hardware make BUONO more flexible and easier use, For example: 3.3V or 5V IO selectable allow you connect some 3.3V modules (such as XBee) to BUONO directly



■ THE DIFFERENCE BETWEEN Arduino MEGA 2560 AND BUONO MEGA 2560

	Arduino MEGA 2560	BUONO MEGA 2560
USB Connector	Type B Female	Micro USB connector
Operating Voltage	5V	3.3V or 5V selectable
3.3V Current	50mA	800mA
5V Current	500mA	1.8A
Input Range	7V to 12V	6V to 23V
MCU	ATMEGA2560	ATMEGA2560
Others		Ground terminal LTP for AVCC

■ FEATURES

- Inherits all of Arduino MEGA 2560's features
- Compatible to Arduino MEGA 2560's pin layout, screw hole and dimensions
- 3.3V or 5V operating voltage selectable
- More visible location of indication LEDs
- Evolved with SMD components
- Digital I/O pins 54 (of which 15 provided PWM output)
- Analog Input pins 16
- Flash memory 256 KB or which 8KB used by boot loader
- SRAM 8 KB
- EEPROM 4KB
- Clock speed 16MHz
- Micro USB connection make the cable can be mixed use with most digital products such as cell phone
- 5V DC/DC provide 95% efficiency
- 6V to 23V wide range input
- Provide max 1.8A for peripheral circuits
- Ground terminal make measurement easier

■ PACKAGE LIST

- 1x BUONO MEGA 2560
- 1x Micro USB cable
- 4x Screw and nut

■ CONTACT US

540 Mill River LN San Jose,
CA 95134

E-mail: support@inhaus.com

Web: www.inhaus.com

