

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement supérieur et de la Recherche scientifique

Université Djilali Bounaama de Khemis Miliana



Faculté des Sciences et de la Technologie

Département de Technologie

Mémoire de Projet de Fin d'études

En vue de l'obtention du diplôme de **MASTER**

Filière : **Génie Éclectique**

Spécialité : **Automatique et Informatique Industrielle**

Thème :

**Conception et réalisation d'une interface
homme-machine (IHM), Application a une machine
d'impression semi-automatique**

Présenté par :

HOUACHEMI Lotfi

RAIS Benyoucef

Soutenu le : 22/09/2024 Devant les membres de jury :

Président : M. M.FEKIR

Examineur : M. S.IKNI

Encadreur : M. S.MAHDAB

Année Universitaire : 2023-2024

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciement :

En préambule à ce mémoire nous remercions Dieu qui nous aide et nous donne la patience et le courage durant ces longues années d'étude.

En second lieu, nous tenons à remercier notre Encadreur Mr.MAHDAB Salim pour ses précieux conseils et son assistance tout au long de la période de travail.

Nos respects et notre gratitude vont également aux membres du jury qui nous ont fait l'honneur de juger ce travail.

Enfin, on remercie tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

Merci.

Dédicace :

Je dédie ce travail :

*À mes chers parents **Benmira** et **Zohra**, je vous suis infiniment reconnaissant pour votre soutien inébranlable tout au long de ce parcours académique.*

Votre encouragement constant et votre foi en mes capacités ont été ma plus grande source de motivation.

*À mon frère bien-aimé **Abdeldjalil** (اللهم ارحمه), qui n'est plus parmi nous aujourd'hui, je dédie ce travail. Ton souvenir et ton inspiration continuent de guider chacun de mes pas.*

*À mon cher frère **Zakaria**, ainsi qu'à sa femme et à son adorable fils **Abdeldjalil**, votre soutien et votre compréhension ont été précieux durant cette période intense.*

À Tous ceux qui ont contribué à la réussite de ce travail.

À tout ma famille & mes amies & tous mes camarades.

Merci.

Lotfi.

Je dédie ce travail :

*À mes très chers parents, **MOSTEFA** et **AHLAM** que je n'arriverais jamais à leurs exprimer mon amour sincère, qu'ils ont jamais dit non à mes exigences, qui n'ont épargné aucun effort pour me rendre heureux.*

*À mes chères frères **IMAD** et **WAIL** et **IHAB** ainsi que ma petite sœur **KAMILIA**, qui n'ont pas cessée de m'encourager et soutenir que le dieu les protège et leurs offre la chance et le bonheur.*

À toute ma famille, source d'espoir et de motivation.

À tous mes amis et mes camarades de la promotion 2019 MASTER en automatique et informatique industrielle.

À tous ceux qui ont sacrifié leur temps pour la science et à tous ceux qui utilisent la science pour le bien et la prospérité de l'humanité.

Merci.

Benyoucef

Résume :

Ce projet traite de la conception et de la réalisation d'une machine d'impression sur vêtements semi-automatique, contrôlée par un Arduino via une interface homme-machine (IHM). L'objectif était d'automatiser partiellement une machine de pressage thermique manuelle pour améliorer son efficacité et sa convivialité. Les améliorations incluent l'intégration d'un contrôleur ARDUINO et d'un système pneumatique, ainsi que la possibilité de régler la température, le temps de pressage et de compter les nombres des cycles de production via l'interface homme machine.

Mots clés : Machine d'impression - ARDUINO – IHM – Pression – Température.

ملخص:

يتناول هذا المشروع تصميم وإنشاء آلة طباعة ملابس نصف أوتوماتيكية، يتم التحكم فيها بواسطة أردوينو عبر واجهة الإنسان والآلة (أش أم أي) كان الهدف هو أتمتة آلة الضغط الحراري اليدوية جزئيًا لتحسين كفاءتها وسهولة استخدامها. تشمل التحسينات دمج وحدة تحكم أردوينو ونظام هوائي، بالإضافة إلى القدرة على ضبط درجة الحرارة ووقت الضغط وحساب عدد دورات الإنتاج عبر واجهة. **كلمات مفتاحية:** آلة طباعة – أردوينو – واجهة الإنسان والآلة – الضغط – درجة الحرارة

Abstract:

This project deals with the design and creation of a semi-automatic clothing printing machine, controlled by an Arduino via a human-machine interface (HMI). The goal was to partially automate a manual heat pressing machine to improve its efficiency and user-friendliness. Improvements include the integration of an ARDUINO controller and pneumatic system, as well as the ability to adjust temperature, pressing time and count numbers production cycles via the HMI.

Keywords: Printing machine - ARDUINO – HMI – Pressure – Temperature.

Liste des abréviations :

PLC : Programmable Logic Controller

A P I: Automate Programmable Industrial

LD: Ladder

FBD: Function Block Diagram.

GRA: Grafecet

LED: Light Emitting Diode.

IDE : Integrated Development Environment

SPA : Système de Production Automatisée

CPU : Central Processing Unit

VCC : Voltage Common Collector

TD : Température détecté

TS : Température saisie

PR : Pression programmée

TP : Temp de pose

T.PRO : Température programmée

CY : Nombre de cycle

GND: Ground

LCD: Liquid Cristal Display

Sommaire :

i.	Chapitre 1 : Généralité sur les automates	4
	I.1. Introduction	4
	I.2. Définition d'API :	4
	I.3. Quelques fabricant des automates :	5
	I.4. Rôles de l'API dans un système de production :	5
	I.5. Application des automates programmables :	6
	I.6. Architecture des automates :	6
	I.6.4. Structure extérieure d'un API :	6
	I.6.5. Structure interne d'un API :	7
	I.7. Fonctionnement d'un A.P.I :	9
	I.8. Critères de choix d'un automate programmable industriel :	10
	I.9. Programmation d'un API :	10
	I.9.1. Ladder (langage contact) :	10
	I.9.2. GRAFCET :	11
	I.9.3. FBD (Function block diagram) :	12
	I.10. Conclusion :	12
ii.	Chapitre 2 : Arduino & machine d'impression	13
	II.1. Introduction :	13
	II.2. Histoire :	13
	II.3. ARDUINO :	14
	II.3.1. Définition d'Arduino :	14
	II.3.2. Application :	14
	II.3.3. Différents types des cartes d'Arduino :	14
	II.4. Arduino UNO :	15
	II.4.1. Définition :	15
	II.4.2. Avantage d'Arduino UNO :	16

II.4.3. Schéma des portes d'Arduino UNO :.....	16
II.4.4. Microcontrôleur ATmega328P :.....	18
II.4.5. Caractéristiques d'ATMega328p :.....	19
II.4.6. Software :	19
II.4.7. ARDUINO IDE :.....	19
II.5. Interface détaillée :.....	20
II.5.1. Les afficheurs LCD :.....	20
II.5.2. Ecran LCD 16x02 & Clavier :.....	21
II.5.3. La fonctionnent des boutons :	22
II.5.4. Caractéristiques principales du Shield :	23
II.5.5. Connexion et Montage du Shield :.....	23
II.6. Présentations du projet :	24
II.6.1. Définition de machine :.....	24
II.6.2. Les principaux composants de la machine :.....	25
II.6.3. Les caractéristiques techniques de la machine :	26
II.6.4. Paramètres Conseillées :.....	26
II.6.5. Mode d'emploi :	27
II.7. Conclusion :	28
III. Chapitre 3 : Organigramme & programme	29
III.1. Introduction :.....	29
III.2. L'organigramme :.....	29
III.2.1. Les entres & les sorties :	29
III.3. Programme :.....	32
III.4. Conclusion :	44
IV. Chapitre 4 : Réalisation & Applications Pratiques	45
IV.1. Introduction :.....	45
IV.2. Réalisation électronique :	45
IV.2.1. Schéma électrique :.....	45

IV.2.2. Description des composants :.....	46
IV.2.3. La Sonde :.....	46
IV.2.4. Transistor :	47
IV.2.5. Relais :.....	48
IV.2.6. Résistance Chauffante :.....	49
IV.2.7. Moteur centralisé : (Actionneur)	50
IV.2.8. Montage électronique réel :.....	50
IV.3. Réalisation mécanique :.....	52
IV.4. Teste & résultat :	52
IV.5. Conclusion :	54
v. Conclusion générale :	55

Table des figures :

FIGURE I-1 : AUTOMATE PROGRAMMABLE INDUSTRIEL (66).....	4
FIGURE I-2 : MITSUBISHI FX1N-40MR (67).....	5
FIGURE I-3 : ROLES DE L'API DANS UN SYSTEME DE PRODUCTION (4).....	5
FIGURE I-4 : AUTOMATE COMPACT LOGO 230 V (7).....	6
FIGURE I-5 : AUTOMATE MODULAIRE SIEMENS (8).....	7
FIGURE I-6 : STRUCTURE INTERNE D'UN API (9).....	8
FIGURE I-7:CYCLE D'UN API (72).....	9
FIGURE I-8: QUELQUES SYMBOLES DE LANGAGE LADDER (15).....	11
FIGURE I-9 : EXEMPLE SUR GRAFCET (17).....	11
FIGURE I-10: FBD PROGRAMMATION (19).....	12
FIGURE II-1: LOGO D'ARDUINO (23).....	14
FIGURE II-2: DIFFERENTS CARACTERISTIQUE DES DIFFERENTES CARTES D'ARDUINO (68).....	15
FIGURE II-3: DEFERENTS TYPES D'ARDUINO (26).....	15
FIGURE II-4: ARDUINO UNO (28).....	16
FIGURE II-5: SCHEMA DES PORTES D'ARDUINO UNO (30).....	17
FIGURE II-6: ATMEL ATMEGA328P (31).....	18
FIGURE II-7: STRCTURE GENERALE D'ARDUINO IDE.....	20
FIGURE II-8 : AFFICHEUR LCD (38).....	20
FIGURE II-9: LCD 16*2 (69).....	20
FIGURE II-10 : ECRAN LCD 16X02 SHIELD (39).....	21
FIGURE II-11: MACHINE D'IMPRESSION SEMI-AUTOMATIQUE (42).....	24
FIGURE II-12: MACHINE D'IMPRESSION (41).....	25
FIGURE III-1: ORGANIGRAMME D'UNE MACHINE D'IMPRESSION.....	31
FIGURE IV-1 :CRCUIT ELECTRIQUE.....	45
FIGURE IV-2 : BU806 TRANSISTOR (49).....	48
FIGURE IV-3 : SYMBOLE D'UN RELAIS (51).....	49
FIGURE IV-4 : RELAIS G5LC-1 (53).....	49
FIGURE IV-5 : BOUGIE PRECHAUFFAGE.....	50
FIGURE IV-6 : MOTEUR CENTRALISE (55).....	50
FIGURE IV-7 : MNTAGE REEL ARDUINO-LCD SHIELD.....	51
FIGURE IV-8 : MONTAGE ELECTRONIQUE COMPLETE.....	51
FIGURE IV-9: VUE A COTE.....	52
FIGURE IV-10 : VUE DE FACE.....	52
FIGURE IV-11 : ETAPE 1.....	52
FIGURE IV-12 : ETAPE 2.....	52
FIGURE IV-13 : ETAPE 3.....	53
FIGURE IV-14 : ETAPE 4.....	53
FIGURE IV-15 : ETAPE 5.....	53
FIGURE IV-17 : DESCENDE D'ACTIONNEUR.....	54

Introduction Générale

Introduction générale :

L'évolution rapide des technologies numériques et de l'automatisation a radicalement transformé divers secteurs industriels, y compris celui de l'impression sur textiles. La demande croissante pour des vêtements personnalisés et de haute qualité pousse les entreprises à chercher des solutions innovantes pour améliorer la précision, la vitesse et l'efficacité de leurs processus de production. Traditionnellement, l'impression sur vêtements a été un processus manuel exigeant en main-d'œuvre et en temps.

Cependant, avec l'émergence de technologies d'automatisation, il est désormais possible de concevoir des systèmes plus efficaces et fiables.

Ce projet de fin d'études se concentre sur le développement d'un système d'impression semi-automatisé, utilisant un Arduino UNO pour le contrôle de la machine et une interface homme-machine (IHM) pour faciliter l'interaction utilisateur.

L'objectif principal est de concevoir et de mettre en œuvre une solution capable de répondre aux exigences de précision et de répétabilité dans l'impression sur vêtements, tout en offrant une interface conviviale et intuitive pour les opérateurs.

L'impression manuelle sur vêtements, bien qu'artisanale et flexible, présente plusieurs inconvénients, notamment la variabilité des résultats, le temps de production prolongé et la dépendance aux compétences de l'opérateur. Ces limitations peuvent entraîner des coûts supplémentaires et des délais de production accrus. En réponse à ces défis, l'automatisation partielle du processus d'impression apparaît comme une solution prometteuse pour améliorer la qualité et l'efficacité.

Le choix de l'Arduino Uno pour ce projet s'explique par sa polyvalence, sa facilité d'utilisation et sa compatibilité avec une multitude de capteurs et d'actionneurs nécessaires pour le processus d'impression.

L'interface homme-machine (IHM) joue un rôle crucial dans la simplification de la programmation et de la gestion des tâches par l'utilisateur. Une IHM bien conçue permet de réduire les erreurs humaines, d'améliorer l'efficacité de l'opérateur et d'offrir une expérience utilisateur intuitive.

L'intégration de l'Arduino Uno avec une IHM permet de créer un système où les utilisateurs peuvent facilement configurer les paramètres d'impression, surveiller le processus et effectuer des ajustements en temps réel.

Les principaux objectifs de ce projet sont :

- Concevoir une machine d'impression semi-automatisée capable de fonctionner avec une précision et une répétabilité élevée.
- Intégrer l'Arduino Uno avec une IHM pour créer une interface utilisateur intuitive et efficace.
- Réduire le temps de production et améliorer la qualité des impressions sur vêtements grâce à l'automatisation partielle.

Pour atteindre ces objectifs, la méthodologie suivante sera adoptée :

- Étudier les technologies actuelles et les meilleures pratiques dans le domaine de l'impression sur textiles et de l'automatisation.
- Développer les schémas et les prototypes de la machine d'impression en utilisant l'Arduino Uno et des composants compatibles.
- Créer une interface utilisateur intuitive en utilisant des logiciels compatibles avec l'Arduino.

En explorant les principes de l'automatisation et en intégrant des technologies modernes, ce projet vise à démontrer le potentiel d'une approche semi-automatisée pour améliorer les processus d'impression sur textiles.

Cette introduction présente le contexte général du projet et pose les bases des objectifs et des méthodes qui seront détaillés dans les sections suivantes :

Chapitre 1 : Généralités sur les automates :

Dans ce chapitre nous donnons certaines généralités (définitions, types, etc....) Sur les automates programmables.

Chapitre 2 : Arduino & machine d'impression

Dans ce chapitre, nous présentons ARDUINO, détaillons l'interface homme-machine IHM et présentons le schéma de notre projet.

Chapitre 3 : Organigramme & programme

Ce chapitre est consacré à l'organigramme et le programme d'Arduino.

Chapitre 4 : Réalisation & Applications Pratiques

Ce chapitre est consacré au mode d'emploi, à la réalisation électronique et mécanique, ainsi qu'aux tests de la réalisation.

Chapitre 1

Généralité sur Les

Automates Industriel

I. Chapitre 1 : Généralité sur les automates

I.1. Introduction

L'automatisation industrielle est un domaine clé dans la modernisation des processus de production. Au cœur de cette révolution se trouvent les automates programmables (AP), également connus sous le nom de contrôleurs logiques programmables (PLC). Les AP sont des dispositifs électroniques utilisés pour contrôler des machines et des processus industriels. Ils offrent une solution fiable et flexible pour l'automatisation, en remplaçant les systèmes de commande traditionnels basés sur des relais et des circuits câblés.

Les automates programmables jouent un rôle crucial dans de nombreux secteurs industriels, notamment la fabrication, la chimie, l'aéronautique et bien d'autres. Leur capacité à exécuter des tâches complexes avec précision et répétabilité en fait des outils indispensables pour les ingénieurs et les techniciens. Ce chapitre explore les principes fondamentaux des automates programmables, leur fonctionnement, leurs structures, et leur application dans divers contextes industriels.

I.2. Définition d'API :

Un automate programmable industriel (API) est un appareil électronique configurable, spécifiquement conçu pour être utilisé dans des environnements industriels. Il permet l'automatisation de diverses tâches en pilotant des actionneurs et des pré-actionneurs grâce à des signaux logiques, analogiques ou numériques. L'API offre une grande flexibilité et peut être adapté à de nombreuses applications, que ce soit en termes de traitement, de composants ou de langage de programmation. (1)



Figure I-1 : Automate Programmable Industriel. (1)

I.3. Quelques fabricant des automates :

Voici quelques exemples d'automates proposés par différents fabricants : (2)

- Siemens (S7-200, S7-300, S7-400, S7-1200, LOGO...)
- Schneider électrique (TSX 17/37/57, TSX micro, premium...)
- Mitsubishi (MELSEC FX1S/FX1N, sériel, système Q...)
- Omron (ZEN, CPM 1A/2A/2C, CS1, CJ1...)
- Rockwell Automation (micrologix 1200/1400, SLC-500, SLC 5000...)
- ABB (AC500, AC800C, S500...)
- Yokugawa (FCN, FCN-RTU, FCJ...)



Figure I-2 : Mitsubishi FX1N-40MR (1)

I.4. Rôles de l'API dans un système de production :

L'API est un dispositif de commande qui transmet des signaux aux actionneurs en fonction des données reçues des capteurs du système matériel. Il agit en suivant un algorithme programmé spécifiquement.

Dans un système bouclé, L'API opère sur les éléments matériels pour la production, incluant la fabrication, l'emballage, etc.

De plus en plus, l'API gère également l'énergie, incluant donc la production de matières premières ou d'objets, dans une partie opérative élargie. (3)

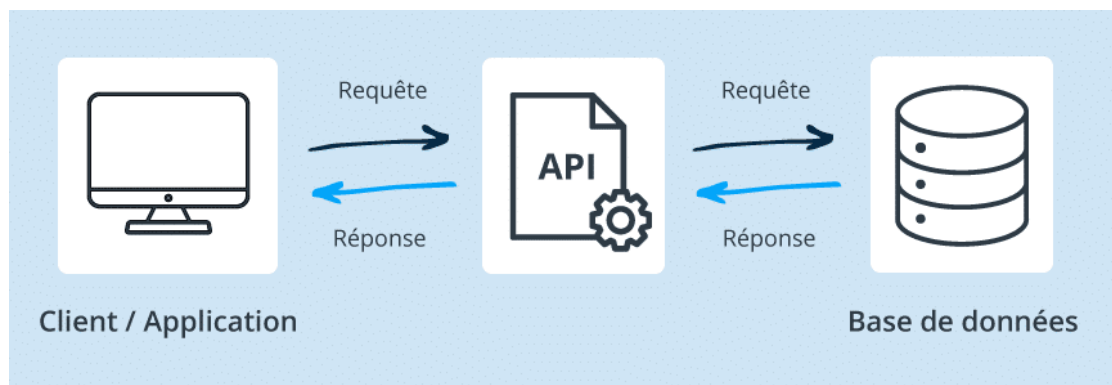


Figure I-3 : Rôles de l'API dans un système de production (4)

I.5. Application des automates programmables :

Les automates programmables sont présents dans beaucoup de domaines soit par exemple les domaines suivants (5) :

- Commande de machines (Transport de matières, stockage, emballage).
- Régulation du processus (chimie, pétrochimie, traitement des eaux).
- Contrôle de systèmes (production et distribution d'énergie).
- Automatisation du bâtiment (chauffage, climatisation, alarme, éclairage).
- Transport (Feux de signalisation- les radars d'aéroportes)

I.6. Architecture des automates :

I.6.4. Structure extérieure d'un API :

Les automates peuvent être de type **compact** ou **modulaire** : (6)

API compactes : Simples, intégrées et adaptées aux applications de base où l'espace est limité.

API modulaires : Flexibles, adaptables et idéales pour des systèmes complexes nécessitant une personnalisation et une extensibilité.

I.6.4.a/ Compact :

Les modules de programmation (comme le LOGO de Siemens, le ZELIO de Schneider ou le MILLENIUM de Crouzet) se distinguent des micro-automates. Ils intègrent un processeur, une alimentation, des entrées et sorties, et sont capables de réaliser des fonctions avancées telles que le comptage rapide ou la gestion d'entrées/sorties analogiques. Bien qu'ils puissent être équipés d'extensions limitées, ces dispositifs sont principalement conçus pour contrôler de petits systèmes automatisés. (6)



Figure I-4 : Automate compact LOGO 230 V (7)

I.6.4.b/ Modulaire :

Le processeur, l'alimentation et les interfaces d'entrées/sorties sont situés dans des modules distincts fixés sur des racks avec un "fond de panier" (bus et connecteurs).

Ces systèmes sont utilisés dans des automatismes complexes qui demandent une puissance, une capacité de traitement et une flexibilité élevée. (8)

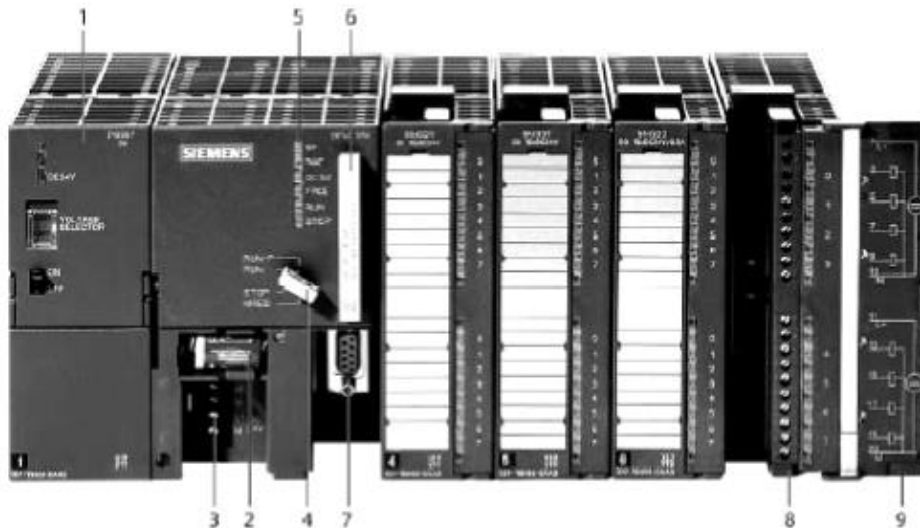


Figure I-5 : Automate Modulaire Siemens (8)

1. Module d'alimentation
2. Pile de sauvegarde
3. Connexion au 24V cc
4. Commutateur de mode (à clé)
5. LED de signalisation d'état et de défauts
6. Carte mémoire
7. Interface multipoint (MPI)
8. Connecteur frontal
9. Volet en face avant

I.6.5. Structure interne d'un API :

Les API comportent cinq principales parties : (6)

- **Module d'alimentation** : il assure la distribution d'énergie aux différents modules.

- **Unité centrale** : À base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).
- **Le bus interne** : Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.
- **Mémoires** : Un système à processeur est toujours accompagné d'un ou plusieurs types de mémoires comme :
 - Mémoire de travail
 - Mémoire ROM
 - Mémoire système
 - Mémoire RAM non volatile
 - Mémoire de chargement
- **Interfaces d'entrées / sorties** :
 - Interface d'entrée : Cette fonctionnalité permet de recevoir les informations du SPA ou du IHM, de les mettre en forme (filtrage, ...) et de les isoler électriquement grâce à l'optocoupleur.
 - Interface de sortie : Elle permet de commander les divers prés actionneurs et éléments de signalisation du SPA tout en assurant l'isolement électrique.

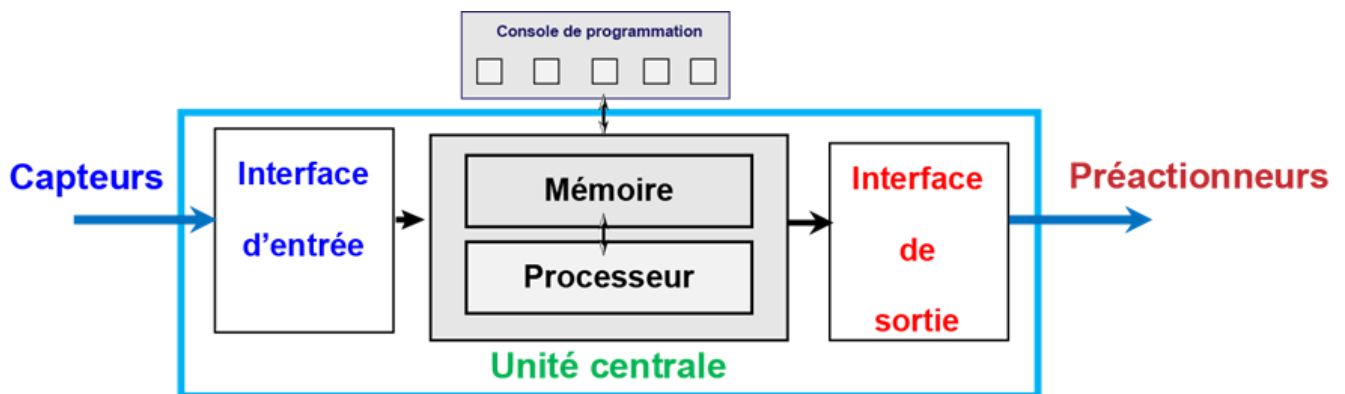


Figure I-6 : Structure interne d'un API (9)

I.7. Fonctionnement d'un A.P.I :

Lorsque l'API est en fonctionnement, trois phases se succèdent (10) :

- **PHASE 1 : ACQUISITION DES ENTREES :**

Durant cette phase brève de quelques microsecondes :

- Les états logiques des entrées sont capturés et enregistrés dans une zone spécifique de la mémoire de données.
- Le programme n'est pas exécuté.
- Les sorties ne sont pas mises à jour.

- **PHASE 2 : TRAITEMENT DU PROGRAMME :**

Durant cette phase brève de quelques microsecondes :

- Les instructions du programme sont exécutées une à une.
- Lorsque l'état d'une entrée doit être lu, le programme utilise la valeur stockée dans la mémoire de données.
- Le programme détermine l'état des sorties et stocke ces valeurs dans une zone de la mémoire de données réservée aux sorties.
 - Les entrées ne sont pas surveillées en continu par l'automate
 - Les sorties ne sont pas mises à jour.

Pendant cette phase, seule la mémoire de données et la mémoire programme sont mises à contribution.

- **PHASE 3 : AFFECTATION DES SORTIES :**

Durant cette phase qui dure quelques microsecondes :

- Les états des sorties mémorisés précédemment dans la mémoire de données sont reportés sur le module de sorties.
- Les entrées ne sont pas scrutées.
- Le programme n'est pas exécuté.



Figure I-7:Cycle d'un API (10)

I.8. Critères de choix d'un automate programmable industriel :

Il revient à l'utilisateur d'établir le cahier des charges de son système et de chercher sur le marché l'automate le mieux adapté à ses besoins : (11)

- Avoir les compétences et l'expérience nécessaire pour programmer la gamme d'automate.
- Le nombre et le type d'entrées et de sorties nécessaires.
- Les moyens de sauvegarde.
- La communication envisagée avec les autres systèmes.
- La fiabilité et la robustesse.
- Le coût d'investissement, de fonctionnement, de maintenance de l'équipement.
- Les capacités de traitement de la CPU.

I.9. Programmation d'un API :

Les programmes dans les systèmes à microprocesseur sont écrits en code machine, c'est-à-dire en une série de nombres binaires qui sont des instructions. (12)

Les types de langages de programmation API les plus connues sont :

I.9.1.Ladder (langage contact) :

Le langage Ladder, également appelé langage à relais, à réseau en échelle ou à contact, est utilisé pour programmer les fonctions logiques d'un API. (13)

Pour programmer ses fonctions, il faut juste copier et transcrire les équations logiques sous formes des schémas électriques à l'aide de symboles placés.

Les symboles graphiques utilisés dans Ladder sont : relais, bobine, contacts, des blocs fonctionnels organisé en réseaux (13)

Ce langage est composé deux barres verticales qui représentent les lignes d'alimentations et de réseaux. Chaque réseau correspond à une équation logique. (14)

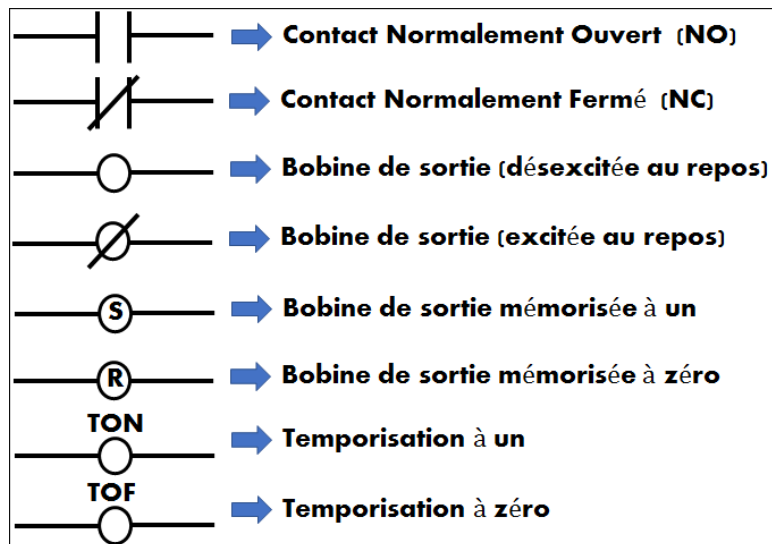


Figure I-8: Quelques symboles de langage Ladder (15)

I.9.2.GRAFCET :

La programmation des API est simplifiée grâce à l'utilisation d'un langage graphique spécialisé appelé GRAFCET (Graphe de Commande Étape).

Ce langage est spécifiquement conçu pour la programmation des API. Les différentes actions sont représentées par des cases, reliées entre elles par des lignes indiquant la progression des opérations. Le passage d'une étape à une autre ne se fait que lorsque l'étape précédente est active et que la transition a été validée. (16)

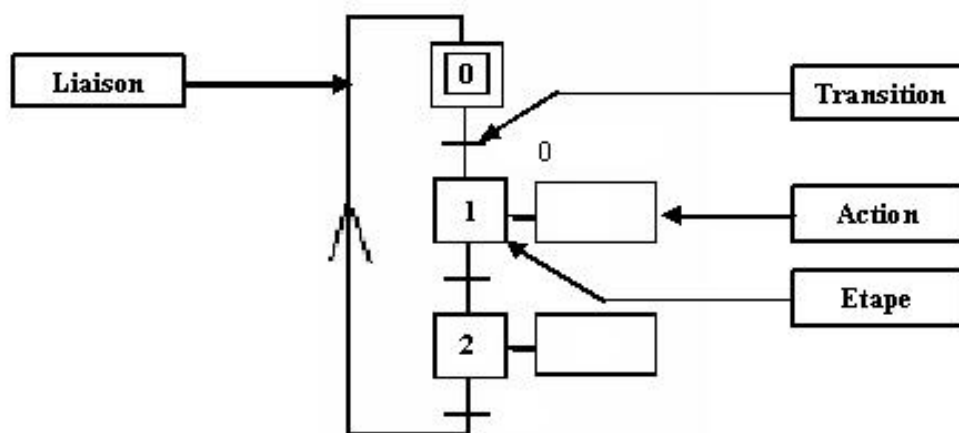


Figure I-9 : Exemple sur GRAFCET (17)

I.9.3.FBD (Function block diagram) :

Le langage FBD est un langage graphique. Il permet la construction d'équations complexes à partir des opérateurs standards, de fonctions ou de blocs fonctionnels. (18)

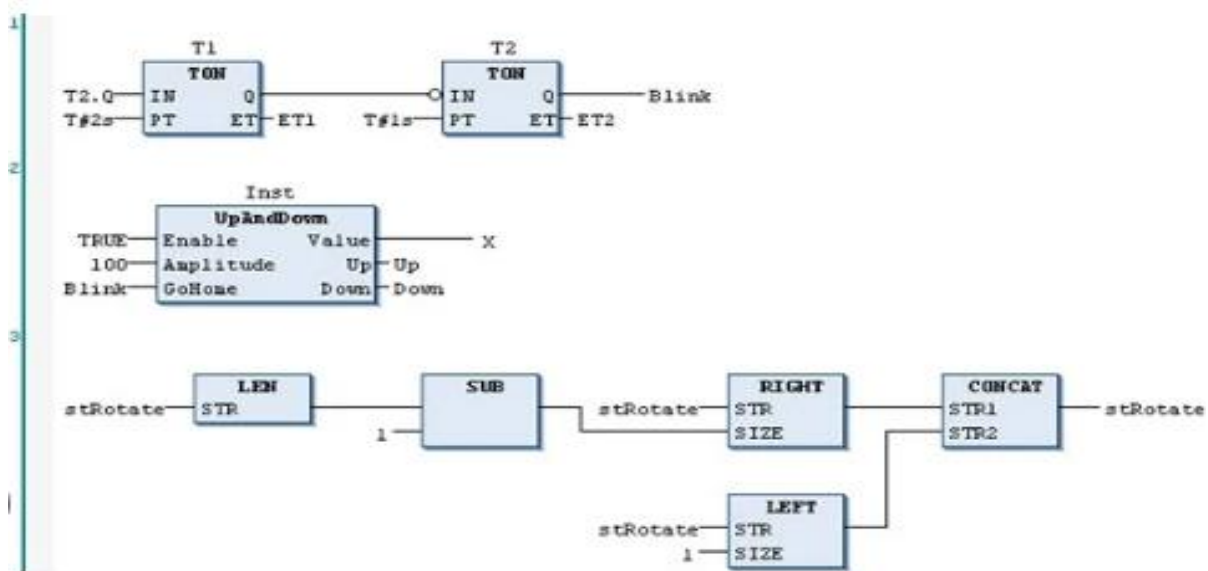


Figure I-10: FBD Programmation (19)

I.10. Conclusion :

Ce premier chapitre a fourni une exploration approfondie des automates programmables industriels (API), en mettant en lumière leur rôle crucial dans l'automatisation industrielle moderne. Nous avons couvert les concepts de base et l'architecture des API, en détaillant leur alimentation, leur rôle, et les différents composants de l'unité centrale.

Les principes de fonctionnement des API ont été expliqués, ainsi que leurs critères de choix pour diverses applications industrielles.

Nous avons également abordé les différents langages de programmation utilisés pour configurer les API, tels que le LADDER, le GRAFCET et le FBD.

En conclusion, les automates programmables se révèlent être des outils indispensables pour toute industrie cherchant à optimiser ses processus de production. Leur capacité à s'adapter rapidement aux besoins changeants et à exécuter des tâches avec une précision inégalée fait d'eux un élément essentiel de l'automatisation moderne.

Chapitre 2

ARDUINO & machine d'impression

II. Chapitre 2 : Arduino & machine d'impression

II.1. Introduction :

Au fil du temps, l'Arduino a acquis une importance capitale dans de nombreux projets, qui vont du plus simple au plus complexe. Par conséquent, une grande communauté bénéficie de cette plateforme libre de droits. Il représente un pont tendu entre le monde réel et le monde numérique, et offre la possibilité d'élargir les interactions entre les êtres humains et les machines, ainsi que les relations entre l'environnement et les machines. (20)

Dans ce deuxième chapitre, nous avons exploré la plateforme Arduino, mettant en lumière sa facilité d'utilisation, ses composants et son interface de programmation conviviale. Nous avons également présenté en détail notre projet d'impression semi-automatisée sur vêtements, en décrivant le schéma du système.

L'objectif principal de ce projet est de concevoir une solution efficace pour l'impression textile en utilisant ARDUINO UNO et une IHM pour contrôler le processus d'impression.

II.2. Histoire :

L'équipe de l'école de Design d'Interaction d'Ivrea (Italie) a développé le projet Arduino. Leur principal défi à cette période (2003-2004) était la complexité et le coût des outils requis pour la création de projets interactifs (entre 80 et 100 euros). Le prototypage était principalement utilisé dans les domaines de l'ingénierie, de la robotique et des domaines techniques. Ils s'attachent donc à concevoir un matériel moins coûteux et plus simple à utiliser. (21)

En 2003, Heranado Barragan, pour sa thèse de fin d'études, avait entrepris le développement d'une carte électronique dénommée « Wiring », accompagnée d'un environnement de programmation libre et ouvert. Cette carte a donc inspiré le projet Arduino (2005) et conçu par une équipe de professeurs et d'étudiants (David Mellis, Tom Igoe, Gianluca Martino, David Caurtielles, Massimi Banzi et Nicholas Zambetti) (22)

II.3. ARDUINO :

II.3.1. Définition d'Arduino :

ARDUINO est une plate-forme pour créer des objets interactifs à usage créatif. Il s'agit d'un circuit imprimé en matériel libre (dont les prototypes sont disponibles en licence libre) qui comprend un microcontrôleur qui peut être programmé pour analyser et générer des signaux électriques, permettant ainsi d'accomplir différentes tâches. Grâce à cet environnement matériel et logiciel, l'utilisateur a la possibilité de concevoir ses projets en expérimentant directement. (20)



Figure II-1: Logo d'ARDUINO (23)

II.3.2. Application :

Le système Arduino nous offre la possibilité de réaliser de nombreuses tâches, qui sont utilisées dans tous les domaines. Pour vous donner quelques exemples, vous pouvez : (24)

- Contrôler les appareils domestiques
- Communiquer avec l'ordinateur
- Commande un robot de dessin
- Contrôle de la température ambiante
- Un jeu de lumières
- Maison intelligente

II.3.3. Différents types des cartes d'Arduino :

Il existe plusieurs types de carte Arduino à savoir carte Arduino Mega2560, carte Arduino MEGA et carte Arduino UNO. Cette dernière est utilisée pour la réalisation de notre projet. (25)

Chapitre 2 : ARDUINO & machine d'impression

Arduino	Processeur	Flash ko	SRAM ko	EEPROM ko	Broches d'E/S numériques	...avec PWM	Broches d'entrées analogique	Type d'interface USB
Diecimila	ATmega168	16	0,5	1	14	6	6	FTDI
Duemilanove	ATmega168 /328p	16/32	0,5/1	1/2	14	6	6	FTDI
UNO	Atmega 328p	32	1	2	14	6	6	ATMega8U2
Leonardo	Atmega32u4	32	1	2,5	20	7	12	ATMega32U4
Mega	Atmega1280	128	4	8	54	15	16	FTDI
Mega2560	Atmega2560	256	4	8	54	15	16	ATMega8U4

Figure II-2: Différents caractéristique des différentes cartes d'Arduino (25)

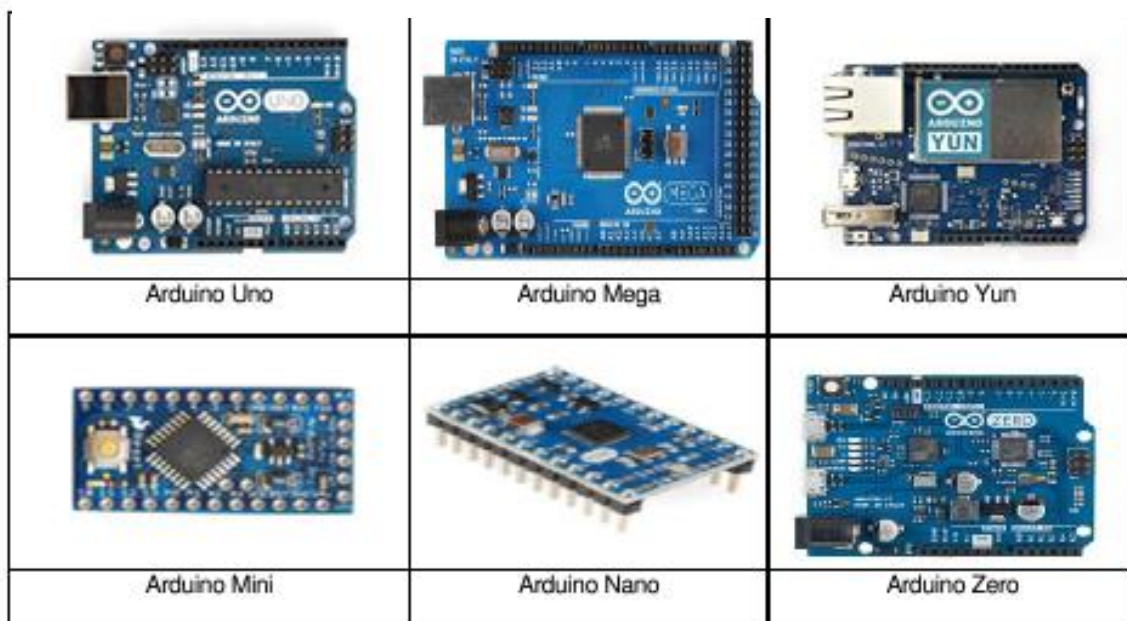


Figure II-3: Différents types d'Arduino (26)

II.4. Arduino UNO :

II.4.1. Définition :

La carte Arduino UNO est un microcontrôleur programmable basé sur le processeur ATmega328, permettant de contrôler divers éléments tels que (des moteurs ou des LED...Etc.). Elle dispose de plusieurs "ports" qui permettent, entre autres, une connexion à un ordinateur ou une alimentation externe.

Cette carte a été développée par une équipe comprenant Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, David Mellis et Nicholas Zambetti. (27)



Figure II-4: ARDUINO UNO (28)

II.4.2. Avantage d'Arduino UNO :

Il y a de nombreux avantages d'Arduino UNO par rapport les autres modules : (29) :

- Le prix : Les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes.
- Clair et simple : L'utilisation de l'environnement de programmation Arduino est simple pour les débutants.
- Logiciel Open Source et extensible : Arduino et le langage Arduino sont libres sous licence open source, accessibles à des programmeurs expérimentés.
- Multi plateforme : Le logiciel Arduino, écrit en JAVA, tous les systèmes d'exploitation supportent l'Arduino comme : Windows, Macintosh et Linux.

II.4.3. Schéma des portes d'Arduino UNO :

Généralement, tout module électronique avec une interface de programmation est continuellement reposé sur la structure d'un ou nombreux circuits programmables. (30)

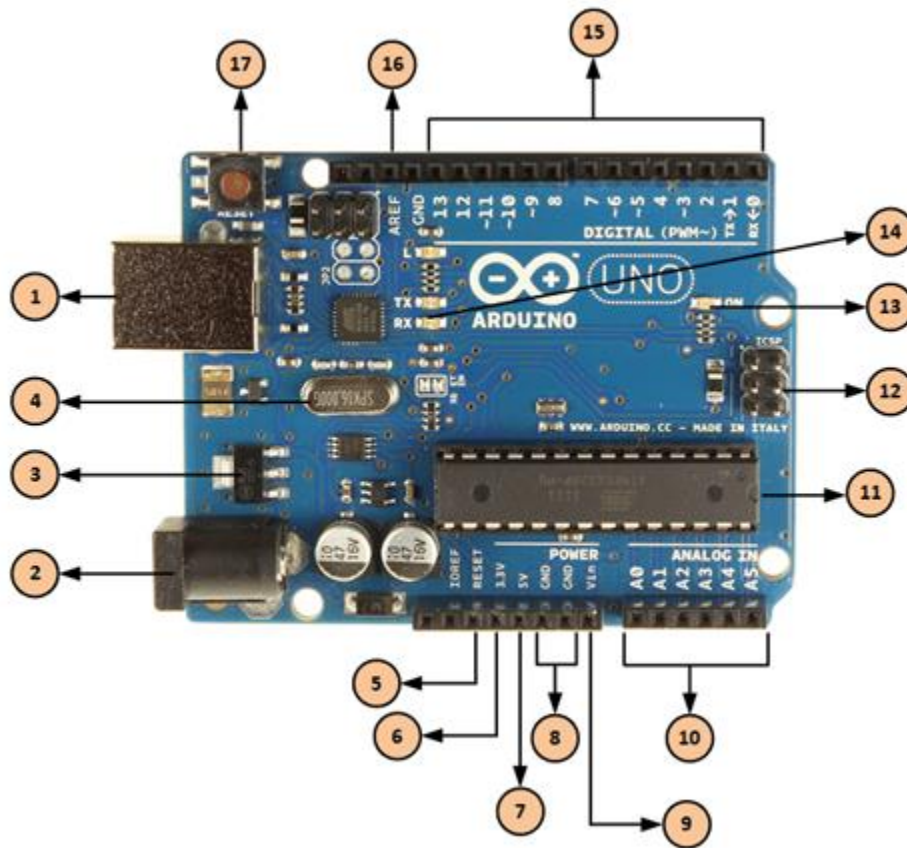


Figure II-5: Schéma des ports d'Arduino UNO (30)

- 1- **Alimentation / Programmation par USB** : Il est possible d'alimenter la carte Arduino en utilisant un câble USB connecté à votre ordinateur.
- 2- **Alimentation via connecteur Jack DC** : La carte Arduino peut être directement alimentée par ce connecteur Jack DC. Ce connecteur (2) est relié au régulateur de tension intégré à la carte. L'alimentation via ce connecteur (2) doit être comprise entre 5 et 12 V.



- 3- **Régulateur de tension** : La fonction du régulateur de tension (3) est de contrôler la tension d'alimentation de l'Arduino pour la stabiliser à la bonne tension du microcontrôleur et de chaque élément de la carte.
- 4- **Oscillateur à quartz** : Les propriétés électromécaniques du quartz lui permettent de vibrer à des fréquences très précises. Ce projet aide Arduino UNO à calculer les données de temps. La fréquence d'ARDUINO est 16 MHz.
- 5- **Arduino Reset** : Vous pouvez redémarrer l'Arduino par "reset" en connectant un bouton externe sur la broche de la carte Arduino.
- 6- **3.3V** : Broche d'alimentation de tension 3.3 Volts.
- 7- **5V** : Broche d'alimentation de tension 5 Volts.

- 8- **GND** : Terre/masse 0v.
- 9- **Vin** : Cette pin est utilisée pour alimenter l'Arduino à partir d'une source de tension externe.
- 10- **Broches analogiques** : Ces broches sont utilisées pour lire des signaux analogiques provenant de capteurs tels que des capteurs d'humidité ou de température (A0 jusqu'à A5).
- 11- **Microcontrôleur principal** : c'est la base de l'Arduino, Les microcontrôleurs proviennent généralement d'ATMEL.
- 12- **Connecteur ICSP** : Ce connecteur permet de programmer le microcontrôleur directement au niveau des couches les plus basses (bootloader, code ASM, etc.)
- 13- **Indicateur LED d'alimentation** : Lorsque vous branchez l'Arduino sur une source d'alimentation, cette LED doit s'allumer pour indiquer que votre carte est correctement alimentée.
- 14- **LED's TX et RX** : TX (émission) et RX (réception).
- 15- **Entrées/Sorties numériques** : La carte Arduino UNO possède 14 broches d'Entrées / Sorties numériques (15), dont 6 peuvent fournir une sortie PWM (Pulse Width Modulation).
- 16- **Broche AREF** : AREF est l'acronyme de "Analog Reference". Cette broche est de temps en temps utilisée pour définir une tension de référence externe (entre 0 et 5v).

II.4.4. Microcontrôleur ATmega328P :

L'ATMega328P est l'un des microcontrôleurs les plus connus, notamment parce qu'il constitue le cœur des cartes Arduino.

Ce composant de la famille AVR peut être programmé en C/C++ pour effectuer des tâches telles que la lecture ou l'émission de signaux logiques ou analogiques via ses ports. (31)

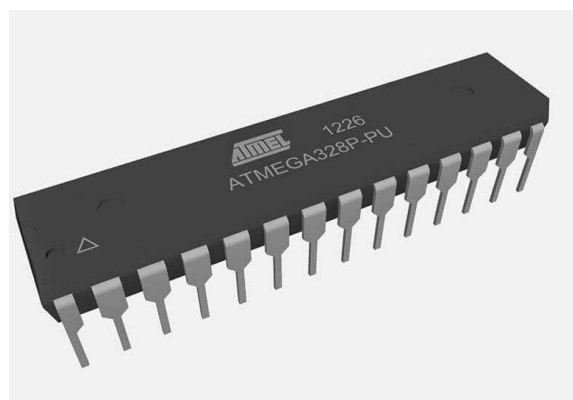


Figure II-6: Atmel ATmega328p (32)

II.4.5. Caractéristiques d'ATMega328p :

Le microcontrôleur ATMega328P présente les caractéristiques suivantes : (33)

- Architecture 8 bits performante
- Fréquence de fonctionnement jusqu'à 20 MHz
- Mémoire flash de 32 Ko
- SRAM de 2 Ko et EEPROM de 1 Ko
- Interfaces série, numériques et analogiques
- Modes de faible consommation d'énergie
- Polyvalence et facilité d'utilisation

II.4.6. Software :

Tout comme il existe de nombreux logiciels pour simuler l'utilisation de la carte Arduino Uno, il existe également une variété de logiciels pour programmer et développer des projets avec Arduino. Parmi ces logiciels, nous pouvons citer l'Arduino IDE (Integrated Development Environment). (34)

II.4.7. ARDUINO IDE :

L'Arduino IDE est un environnement de développement intégré (IDE) libre et open source, conçu spécifiquement pour la programmation des microcontrôleurs Arduino.

Il propose une large bibliothèque de fonctions et de codes pré-écrits, simplifiant ainsi la conception de projets électroniques. Ce logiciel est indispensable pour les amateurs, les étudiants et les professionnels engagés dans des projets de microcontrôleurs. (35)

Les programmes Arduino comportent trois parties : (36)

- Déclaration des variables et constantes globales.
- Fonction Setup () : exécutée au démarrage du programme ou lors de l'appui sur le bouton Reset.
- Fonction Loop () : boucle exécutée sans fin (tant que l'Arduino est sous tension)

NB : Il est possible de télécharger ARDUINO IDE sur ce lien : (37)

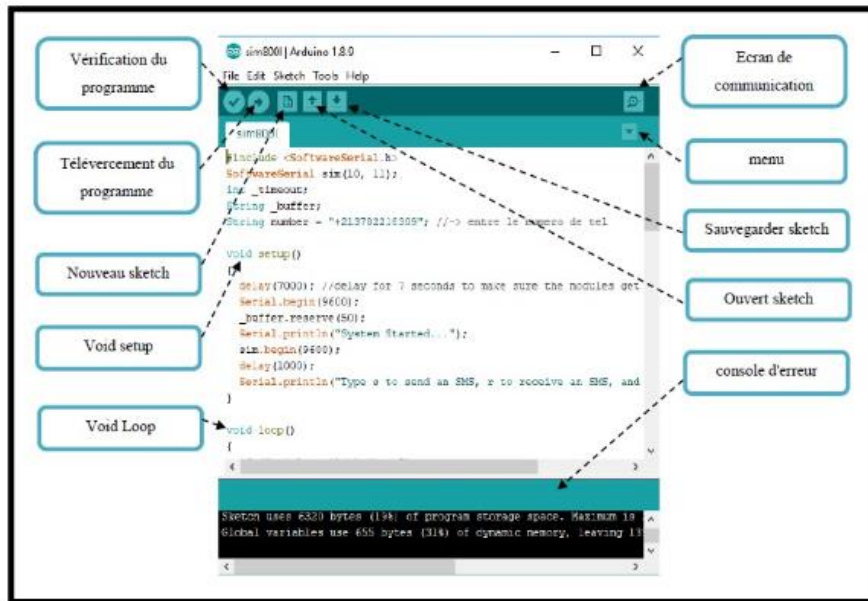


Figure II-7: Structure générale d'ARDUINO IDE

II.5. Interface détaillée :

II.5.1. Les afficheurs LCD :

Les afficheurs LCD (Liquid Crystal Display) sont essentiels dans les applications nécessitant la visualisation de paramètres, car ils servent d'interface Homme/Machine. Ils sont couramment utilisés dans les projets avec des microcontrôleurs, offrant une interface conviviale pour l'utilisateur.

De plus, les afficheurs LCD sont utiles pendant la phase de développement des programmes, car ils permettent d'afficher facilement les valeurs des variables pour le débogage et l'analyse. (38)



Figure II-8 : Afficheur LCD (39)

Ils existent différents types d'afficheur selon leur vue de l'extérieur. Mais généralement ils accomplissent le même rôle, afficher des caractères alphanumériques sur son écran. Chaque caractère possède son code ASCII.

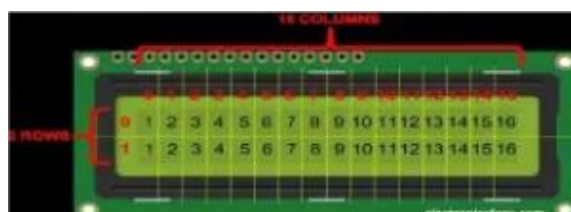


Figure II-9: LCD 16*2 (39)

L'afficheur possède 8 entrées, de DO à D7 et ils sont liés aux broches du microcontrôleur. Le PIC18F4520 est l'acteur qui envoie les signaux à travers les broches liés aux entrées de l'afficheur, et dans ce dernier le message sera traduit par des caractères.

II.5.2. Ecran LCD 16x02 & Clavier :

L'écran LCD 16x02 (Shield) est conçu pour être utilisé avec les cartes Arduino compatibles, fournissant une interface intuitive pour la gestion des menus et la sélection des options.

Il présente des caractères blancs sur un écran LCD bleu-noir. L'écran est équipé d'un clavier comprenant cinq boutons : sélection, haut, droite, bas et gauche. (40)

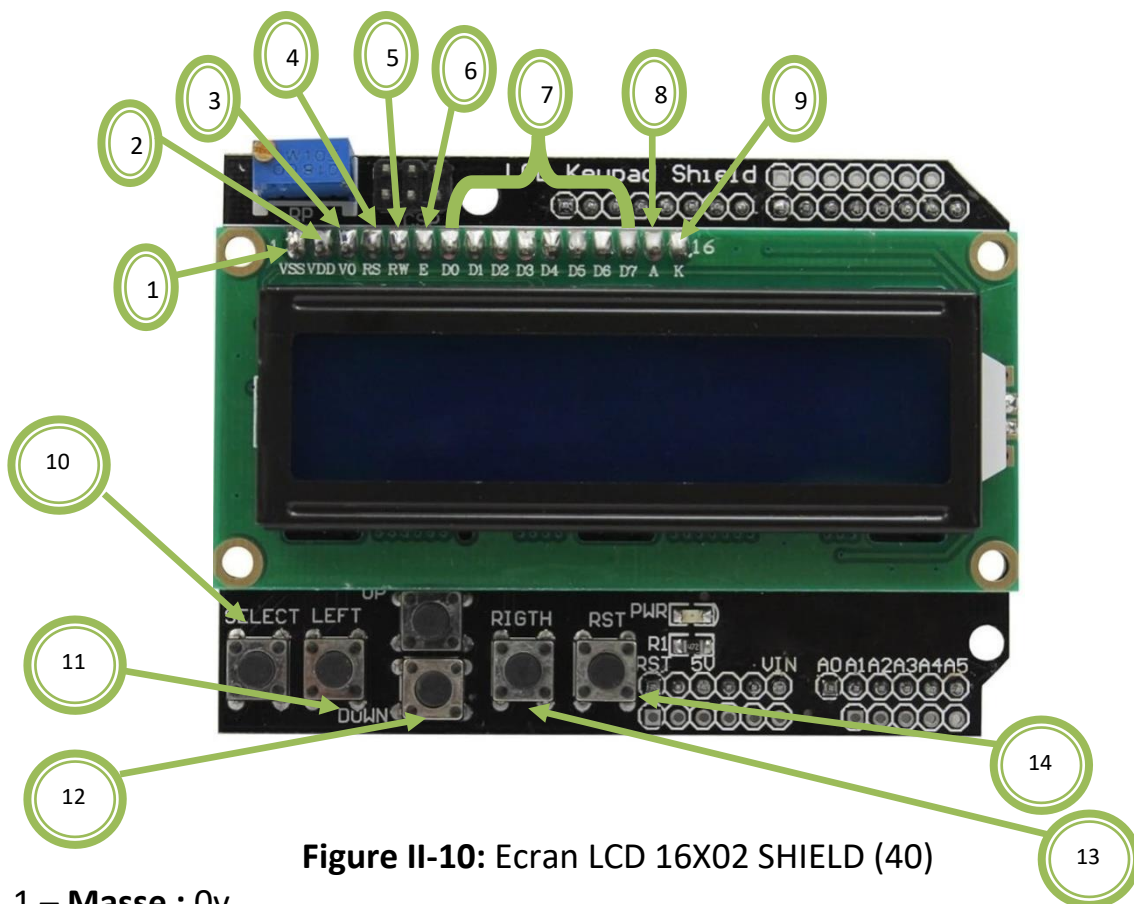


Figure II-10: Ecran LCD 16X02 SHIELD (40)

- 1 – Masse : 0v.
- 2 – Vcc : +5v de l'afficheur.
- 3 – VO : Réglage de contraste entre 0 et +5V.
- 4 – Rs : Si RS=1 caractère (donnée) ; Si RS=0 instruction (commande).
- 5 – R/W : Si R/W=0 écriture vers l'afficheur ; Si R/W=1 lecture de l'afficheur
- 6 – E : Entrée de validation.
- 7 – D0-D7 : Transfert des données.
- 8 – A : Anode de rétro éclairage (+5 V).

9 – K : Cathode du rétro éclairage (masse).

10 - **Select bouton.**

11 – **Gauche bouton.**

12 – **Haut/bas bouton.**

13 – **Droit bouton.**

14 – **Rest bouton.**

II.5.3. La fonctionnent des boutons :

Les boutons d'un LCD Shield (6 boutons) sont des composants intégrés à un écran LCD et qui servent d'interface utilisateur pour interagir avec un microcontrôleur d'Arduino.

En réalité, chaque bouton poussoir est câblé à un réseau de résistances qui forment un pont diviseur de tension relativement complexe.

Pour savoir quel bouton a été appuyé, il suffit de faire appel à la fonction `analogRead()` sur la broche **A0** et comparer la valeur lue par rapport à une fourchette de valeurs types : (41)

Nous obtenons obtient les valeurs suivantes :

Bouton :	Valeur lue avec <code>analogRead(A0)</code> :	Voltage :(v)
Aucun	1023	5
UP (haut)	132	0.64
DOWN (bas)	380	1.85
RIGHT (droite)	0	0
LEFT (gauche)	480	2.34
SELCTE (sélection)	721	3.52
Rest	/	/

Pour calcule la valeur de voltage pour chaque bouton, nous utilisons cette règle

v_{MA} : Valeur maximale associée dans `analogRead()` = 1023

C_{AN} : Convertisseur Analogique – Numérique = 5v

A_b : Valeur lue avec `analogRead (A0)`

V_x : Voltage de bouton

1023 → 5v

132 → V_x

$V_x \approx 0.64 \text{ v}$

II.5.4. Caractéristiques principales du Shield :

- **Dimensions** : 16 colonnes et 2 lignes (42)
- **Compatibilité** : S'installe directement sur l'Arduino
- **Rétroéclairage** : Contrôlé par une broche dédiée
- **Ajustement du contraste** : Via un potentiomètre intégré

II.5.5. Connexion et Montage du Shield :

La connexion d'un écran LCD 16x2 Shield à l'Arduino est simplifiée car le Shield est conçu pour s'installer directement sur l'Arduino sans nécessiter de câblage complexe.

II.5.5.a/ Broches utilisées :

Les broches suivantes de l'Arduino sont utilisées pour contrôler l'écran LCD :
(43)

Broche Ecran :	Broche ADUINO :
D4 (données)	D4
D5 (données)	D5
D6 (données)	D6
D4 (données)	D7
RS (signal de contrôle données / commandes)	D8
EN (signal début transmission)	D9
Rétro-éclairage	D10

II.5.5.b/ Ajustement du contraste de l'écran :

L'écran LCD intégré au Shield comporte un potentiomètre multi-tours, situé en haut à gauche de la carte. Ce composant permet d'ajuster le contraste de l'affichage, c'est-à-dire la clarté et la lisibilité des caractères affichés à l'écran. Un contraste mal réglé peut rendre l'affichage difficile à lire, soit en raison d'une trop grande saturation des pixels, soit en raison d'une luminosité excessive. (42)

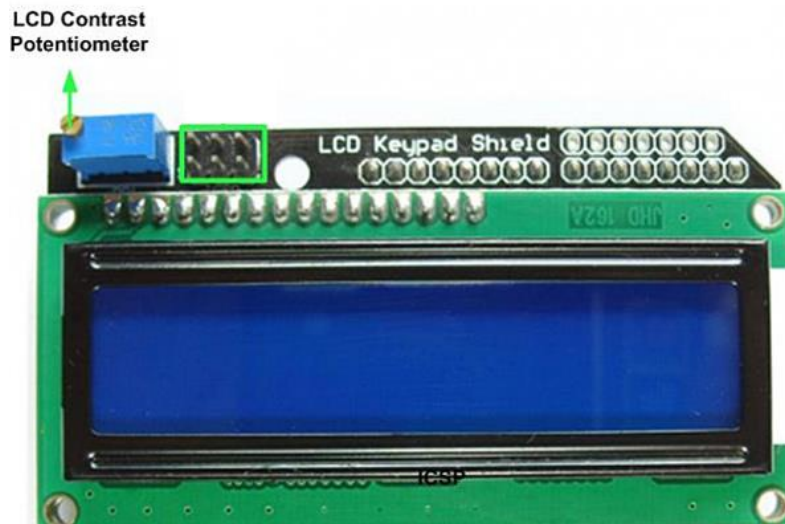


Figure II-11 : LCD contraste Potentiomètre (40)

II.6. Présentations du projet :

II.6.1. Définition de machine :

La machine d'impression semi-automatique développée dans le cadre de ce projet est un système contrôlé par une carte Arduino UNO, Commandée par une IHM conçu pour l'impression précise sur divers types de tissus.

Elle intègre plusieurs composants et fonctionnalités essentiels pour assurer une qualité d'impression élevée et reproductible.

Une base en forme de "U" est intégrée à notre machine, ce qui permet d'imprimer de différents textiles sur la plaque inférieure. (44)

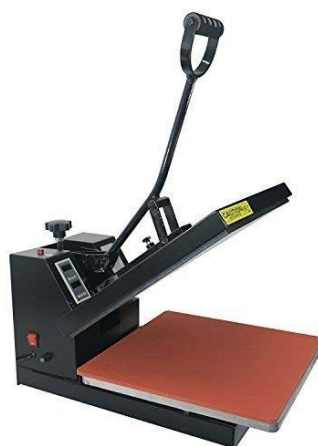


Figure II-12: Machine d'impression semi-automatique (45)

II.6.2. Les principaux composants de la machine :

Voici tous les éléments qui composent la presse pneumatique dans la figure ci-dessous : (44)

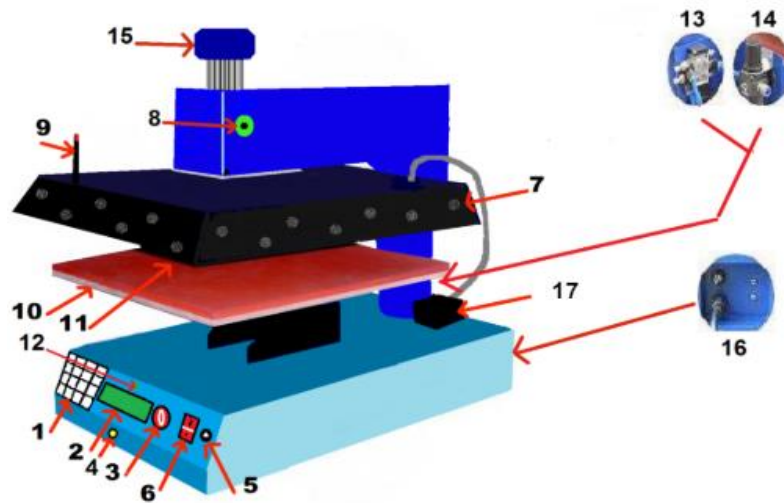


Figure II-13: Machine d'impression (44)

1. Clavier 4*4
2. Interface Homme machine IHM
3. Arrêt d'urgence
4. Buzzer 1500 ms
5. Sélecteur départ cycle manuel
6. Interrupteur principal
7. Capot de sécurité
8. Boutons poussoir départ cycle
9. Poignée du titoir
10. Plateau inférieur en mousse
11. Plateau chauffant
12. LED verte
13. Limiteurs de débit pour descente et remontée vérin
14. Capteur d'air
15. Vérin
16. Fusible et cordon secteur
17. Connecteur du plateau chauffant

II.6.3. Les caractéristiques techniques de la machine :

La machine semi-automatique comporte plusieurs paramètres qui lui permettent d'accomplir ses fonctionnalités, grâce à des caractéristiques soigneusement étudiées.

Voici quelques-unes de ces caractéristiques techniques :

- Poids en ordre de marche : 8 kg
- Hauteur : 410 mm
- Largeur : 33 mm
- Dimension du plateau : 290*400 mm
- Alimentation électrique : 220VAV / 12 VDC
- Ampérage : 3.8A
- Puissance : 836 W
- Fréquence : 50/60 Hz
- Température à affichage digital : par matériel
- Durées : 3s
- Pression : 1 bar

II.6.4. Paramètres Conseillés :

Ces valeurs ne sont présentées qu'à titre indicatif ; elles peuvent varier selon les matériaux et doivent absolument être contrôlées avant chaque impression :
(46)

Type Paramètres	Polyester	Coton	Soie
Température	160C – 180C	150C -170C	120C -140C
Pression	1 bar	1 bar	1 bar
Temp de pose	3 s	3s	3s

NB : Dans notre programme, nous avons fixé une plage de température de **35-55°C** pour permettre un préchauffage progressif, car la bougie chauffante prend du temps à atteindre la température optimale pour l'impression.

II.6.5.Mode d'emploi :

Dans cette section, nous examinerons la manière d'utiliser la machine. Qui sera responsable de la mise en relation entre une interface d'une machine (IHM) et l'opérateur :

1. Il faut positionner la machine sur une surface plane et stable, puis vérifier le courant (le courant de référence est de 220 VAC/12 VDC) et connecter la machine. Assurez-vous également de brancher le câble USB de l'Arduino dans le PC pour établir la communication.
2. Mettez l'appareil sous tension.
3. Attendez que la machine initialise tous ses composants. Un message indiquant « WELCOME » s'affichera.
4. Appuyez sur le bouton "DROITE" pour entrer dans le menu de sélection du type de tissu.
5. Appuyez sur le bouton correspondant pour sélectionner le tissu souhaité : UP pour Polyester / RIGHT pour Coton / DOWN pour Soie.
6. Pour le **polyester** :
 - a. La température actuelle est affichée : « TEMPERATURE = 25 C ».
 - b. Utilisez les boutons "HAUT" et "BAS" pour ajuster la température cible (160C-180C).
 - c. Appuyez sur le bouton "DROITE" pour confirmer la température :
 - a. Si $TS < Min$ & $TS > Max$: afficher le message « FAILED » et retourner à l'étape a.
 - b. Si $TS > Min$ & $TS < Max$
 - c. Passez à la pression, au temps de pose et au nombre de cycles.
 - d. Si $TA < TS$: afficher le message « 1 2 3 ... » et faire descendre le moteur en affichant « merci ».
 - e. Fin d'impression et retour à l'étape 5 pour un nouveau cycle (c'est-à-dire en incrémentant le compteur de cycles avec $cycles = cycles + 1$).
7. Pour le **Coton** :
 - a. La température actuelle est affichée : « TEMPERATURE = 25 C ».
 - b. Utilisez les boutons "HAUT" et "BAS" pour ajuster la température cible (150C-170C).
 - c. Appuyez sur le bouton "DROITE" pour confirmer la température :
 - a. Si $TS < Min$ & $TS > Max$: afficher le message « FAILED » et retourner à l'étape a.
 - b. Si $TS > Min$ & $TS < Max$
 - c. Passez à la pression, au temps de pose et au nombre de cycles.

- d. Si $TA < TS$: afficher le message « 1 2 3 ... » et faire descendre le moteur en affichant « merci ».
 - e. Fin d'impression et retour à l'étape 5 pour un nouveau cycle (c'est-à-dire en incrémentant le compteur de cycles avec $cycles = cycles + 1$).
8. Pour le **Soie** :
- a. La température actuelle est affichée : « TEMPERATURE = 25 C ».
 - b. Utilisez les boutons "HAUT" et "BAS" pour ajuster la température cible (120C-140C).
 - c. Appuyez sur le bouton "DROITE" pour confirmer la température :
 - a. Si $TS < Min$ & $TS > Max$: afficher le message « FAILED » et retourner à l'étape a.
 - b. Si $TS > Min$ & $TS < Max$
 - c. Passez à la pression, au temps de pose et au nombre de cycles.
 - d. Si $TA < TS$: afficher le message « 1 2 3 ... » et faire descendre le moteur en affichant « merci ».
 - e. Fin d'impression et retour à l'étape 5 pour un nouveau cycle (c'est-à-dire en incrémentant le compteur de cycles avec $cycles = cycles + 1$).

II.7. Conclusion :

Ce chapitre a offert une compréhension détaillée des concepts techniques et pratiques pour la création d'une machine d'impression semi-automatique contrôlée par Arduino.

Nous avons défini l'Arduino et son rôle central dans les projets électroniques, détaillé son interface et ses composants, tels que l'écran LCD KEYPORD, les boutons de contrôle.

De plus, nous avons expliqué le mode d'emploi de la machine, incluant les étapes de mise sous tension, de sélection du type de tissu, de configuration des paramètres, de démarrage et de fin du cycle de sérigraphie, garantissant ainsi une utilisation optimale et sécurisée.

Chapitre 3

Organigramme & programme

III. Chapitre 3 : Organigramme & programme

III.1. Introduction :

Dans les chapitres précédents, nous avons abordé la conception globale de la machine d'impression, en incluant les aspects mécaniques, électroniques, et les interfaces utilisateur.

Ce chapitre présente une analyse détaillée du fonctionnement de la machine d'impression en utilisant un organigramme structuré et des programmes spécifiques pour chaque étape du processus.

L'objectif principal est de fournir une vue d'ensemble claire et compréhensible des différentes opérations impliquées dans la machine, en mettant en lumière les interactions entre les composants matériels et logiciels.

III.2. L'organigramme :

Il ne nous reste plus qu'à créer un organigramme, qui est un dessin représentatif d'un algorithme que nous utiliserons pour déterminer le travail du code machine au moyen de formes géométriques sur la base desquelles nous suivrons les étapes de programmation.

Pour les créer, nous définirons les entrées et les sorties, puisqu'elles échangent des données entre le microcontrôleur et ses périphériques associés. De cette façon, le système peut réagir aux changements de son environnement.

III.2.1. Les entres & les sorties :

III.2.1.a/ Les entrées :

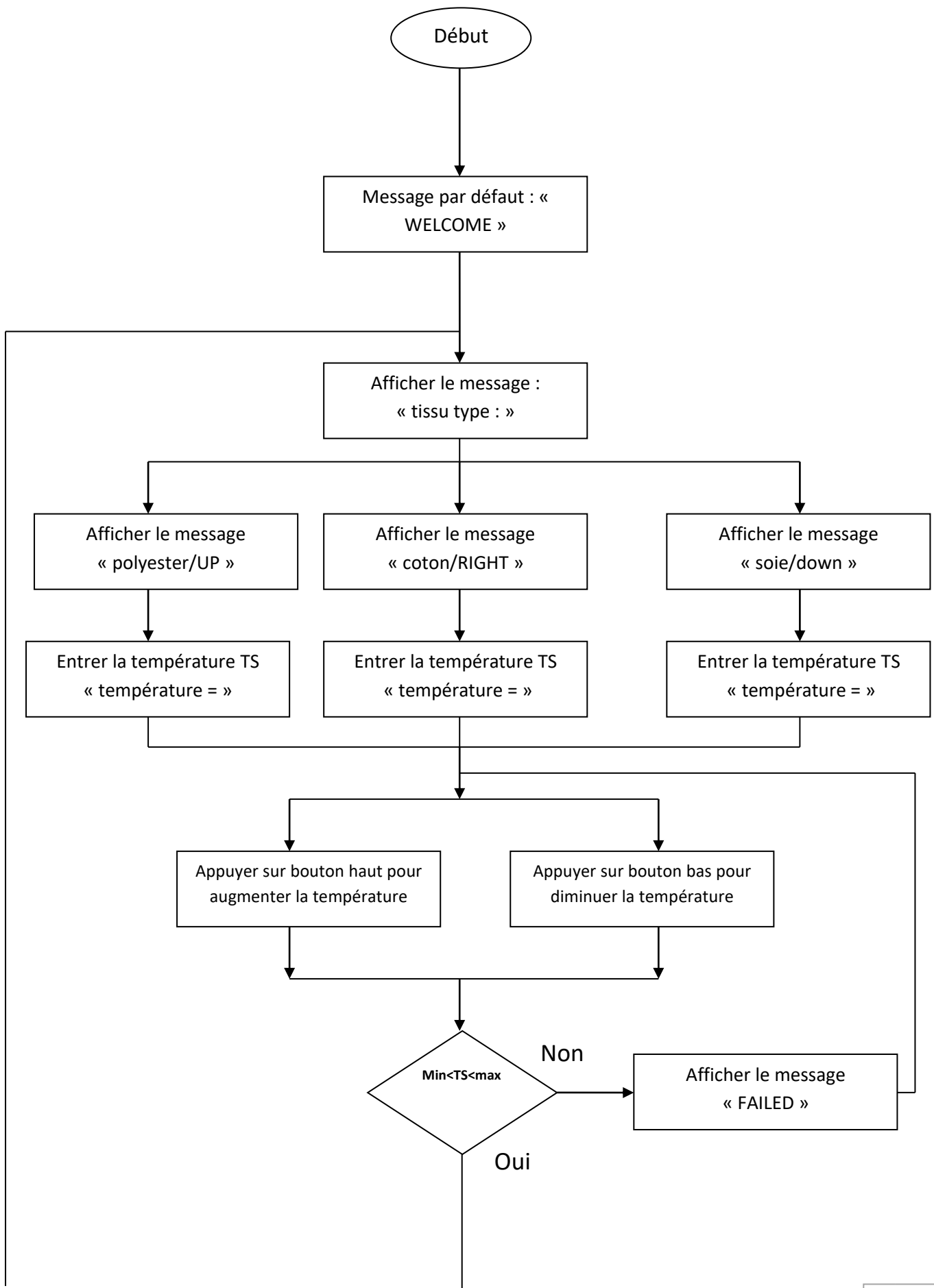
- Les boutons de LCD 16*02
- Sonde de température PTC

III.2.1.b/ Les sorties :

- Ecran LCD I2c
- Resistance chauffante
- Moteur (Actionneur)

Chapitre 3 : Organigramme & programme

Maintenant on peut dessiner notre organigramme :



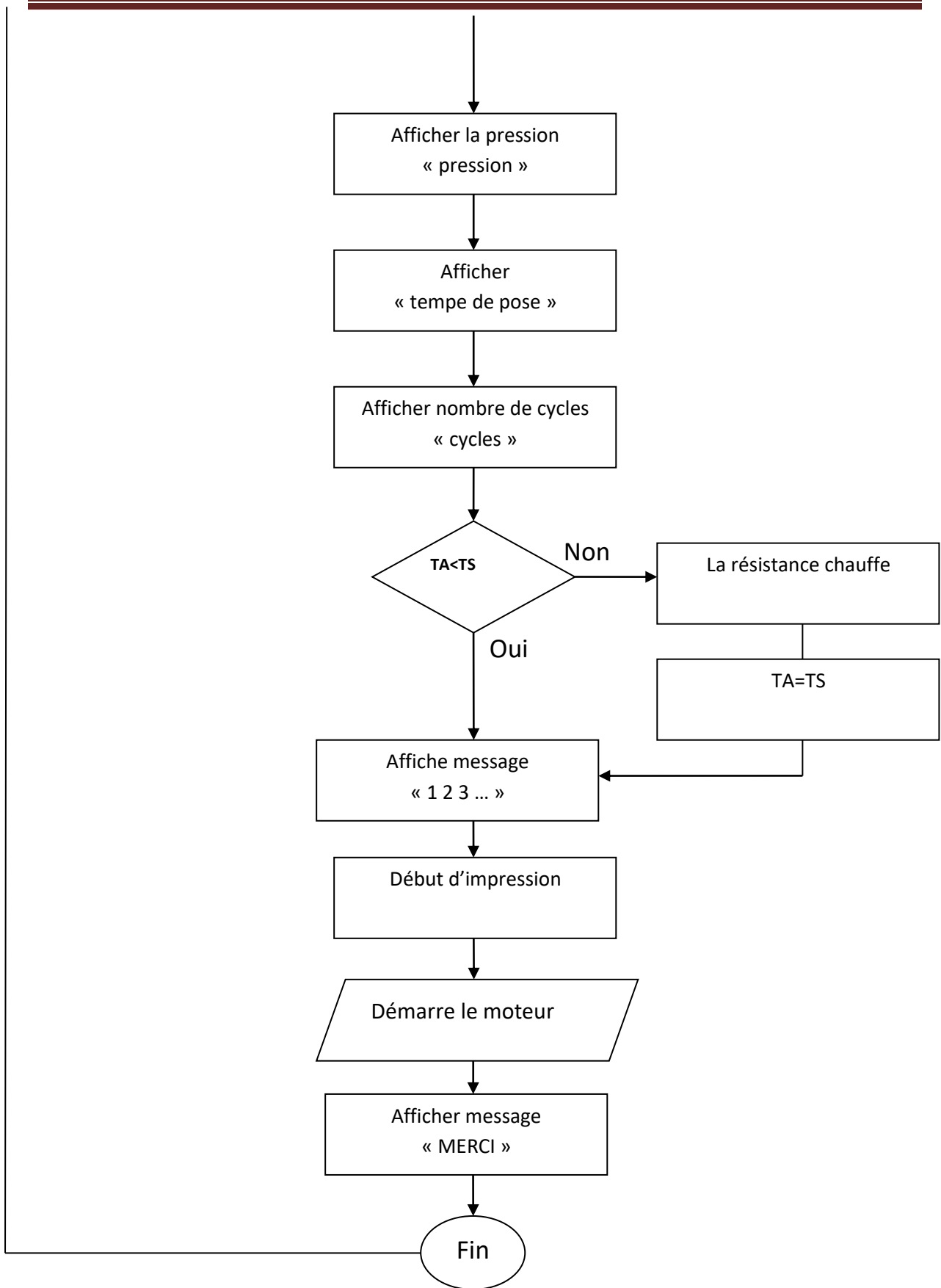


Figure III-1: Organigramme d'une machine d'impression

III.3. Programme :

Le programme permet à l'utilisateur de sélectionner un type de tissu des vêtements, de configurer la température, la pression, le temps de pose et le nombre de cycles, et de contrôler une résistance chauffante et un moteur en fonction des lectures de température.

1 –Le programme commence par inclure la bibliothèque pour gérer l'écran LCD puis nous avons définie l'emplacement des broches pour l'écran comme ça LiquidCrallcd(pin_RS, pin_EN, pin_d4, pin_d5, pin_d6, pin_d7) :

```
//Appeler bib de LCD
#include<LiquidCrystal.h>

//Instalation les pins de lcd
LiquidCrallcd(8, 9, 4, 5, 6, 7);
```

2 –Les broches pour le capteur de température (sonde) 'ta', l'état des boutons de LCD 'buttonState', l'actionneur de bras (moteur) 'moteur', la LED 'led' et la résistance chauffante (bougie préchauffage) 'ress' sont définies.

```
//La sonde
const int tempPin = A2; //Analogique pin (la sonde)
float te;
float tvolt;
float ta; //Température actuelle

//Les 5 boutons
const int buttonPin = A0; // Analogique pin (buttons)

//Les Sorties :
int moteur = 11; // Moteur pin
int ress = 13; // Résistance chauffante
```

3 – Plusieurs variables sont initialisées pour gérer les lectures de température, les états des boutons, le choix du tissu des vêtements '**choix**', l'état du menu de chaque tissu '**menuState**', la température initiale saisie '**ts**', la pression '**press**', une plage de températures comprenant la température minimale '**min**' et maximale '**max**', le nombre de cycles de chaque impression '**cycles**', ainsi que le temps de pose '**tp**'//Les Variables

```
int choix = 0;
// Fabrique choix: polyester=0, Cotton=1, soie=2
int menuState = 0; // Menu
int ts = 25; // Température initiale
int press = 1; // La valeur de la pression
int min = 40; // Min valeur de température
int max = 55; // Max valeur de température
int cycles = 0; // Nombres de cycles
int tp = 15; // Temp de pose (s)
```

3 – dans la dernière partie : Le moniteur série est démarré avec une fréquence de 9600 bauds puis on peut déclarer les dimensions de notre LCD (16*02) « l'écran est initialisée » après nous avons configurée Les broches comme entrées '**temppin**'&'**buttonPin**' ou sorties '**moteur**', '**ress**'&'**led**' selon leur utilisation, à la fin un message de bienvenue ("WELCOME") est affiché et à la fin.

```
void setup(){
    Serial.begin(9600);

    lcd.begin(16, 2);
    lcd.clear();

    // Pin setup
    pinMode(temppin, INPUT);
    pinMode(buttonPin, INPUT);
    pinMode(moteur, OUTPUT);
    pinMode(ress, OUTPUT);
    pinMode(led, OUTPUT);

    // Affiche message 'WELCOME'
    lcd.setCursor(0, 0);
    lcd.print("WELCOME");
}
```


4 – dans le boucle principale `'loop'` : L'état du bouton est lu en utilisant `analogRead ()`, La température actuelle est lue à partir de la sonde et calculée en utilisant les formules appropriées, l'affichage de température sur Serial monitor permettre de suivi de température de bougie, la LED est allumée pendant 3 s c'est-à-dire le cycle a commencé !

```
voidloop(){
  //lire la valeur de bouton
  intbuttonState = analogRead(buttonPin);

  //lire la temperature
  te = analogRead(temppin);
  tvolt = te * (0.00488); // 0.00488=5v/1023
  ta = (tvolt / 0.01477982452) - 20; //
  Serial.println(ta); //Afficher la Température Actuelle dans SM
  delay(1500);

  digitalWrite(led, HIGH);
  delay(3000);
```

5 - En fonction de l'état du menu `'menuState'` différentes actions sont exécutées, Si `'menuState'` est 0, un menu de sélection du type de tissu est affiché.

Les utilisateurs peuvent choisir entre Polyester, Coton ou Soie en utilisant les boutons.

Selon le type de tissu sélectionné (Polyester, Coton ou Soie), la fonction correspondante (`'polyester()' , 'coton()' , 'soie()'`) est appelée.

```
//Programme Principale
switch(menuState){
  // etape 1: choix de tissu
  case0:
  if(buttonState<50){ // bouton droite pressé
  menuState = 1;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Tissu Type:");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Polyester/UP"); //1er choix
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Coton/RIGHT"); //2eme choix
  delay(2000);
  lcd.clear();
```

```
lcd.setCursor(0, 0);
lcd.print("Soie/DOWN"); //3eme choix
delay(3500);
}
break;
// etape 2: après choisir

case1:
if(buttonState<250&&buttonState>= 50){ // haut bouton pressé
choix = 0;
polyester(buttonState);
}elseif(buttonState<50){ // droite bouton pressé
choix = 1;
coton(buttonState);
}elseif(buttonState<400&&buttonState>= 250){ // bas bouton pressé
choix = 2;
soie(buttonState);
}
break;
}
}
```

6 –Pour chaque tissu : polyester, coton et soie :

A – polyester :

Cette fonction affiche la température actuelle et permet de l'ajuster en utilisant les boutons Haut et Bas (haut pour augmenter+ / bas pour diminuer la valeur-) le bouton gauche ramène à menu principal, Le bouton Droite passe à l'affichage et à la configuration de la pression '**press**', du temps de pose '**tp**' et du nombre de cycles '**cycles**'

La température est lue en continu '**ta**' et comparée à la température saisie '**ts**'. La résistance chauffante est activée ou désactivée en conséquence.

Si la température actuelle est dans la plage définie, la LED '**Led**' et le moteur '**moteur**' sont activés pendant **15 secondes**.

```
//1er choix
voidpolyester(intbuttonState){
menuState = 2;
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Polyester:");
delay(2000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp=");
lcd.setCursor(6, 0);
lcd.print(ts);
lcd.setCursor(8, 0);
lcd.print(" C");
```

```
while(menuState == 2){
  buttonState = analogRead(buttonPin);

  //saie la température
  if(buttonState>= 50&&buttonState<250){ // haut Button pressé
    ts++;
    lcd.setCursor(6, 0);
    lcd.print(" ");
    lcd.setCursor(6, 0);
    lcd.print(ts);
  }elseif(buttonState>= 250&&buttonState<400){ // bas button pressé
    ts--;
    lcd.setCursor(6, 0);
    lcd.print(" ");
    lcd.setCursor(6, 0);
    lcd.print(ts);
  }elseif(buttonState>= 400&&buttonState<600){ //gauche button pressé
    menuState = 1; // retour a 0
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Tissu Type:");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Polyester/UP");
    delay(3500);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Coton/RIGHT");
    delay(3500);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Soie/DOWN");
    delay(3500);
  }elseif(buttonState<50){ // bouton droite
    if(ts< min || ts> max){
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("FAILED");
      delay(2000);
    }

    // rtour à la page précédente
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Temp=");
    lcd.setCursor(6, 0);
    lcd.print(ts);
    lcd.setCursor(8, 0);
```

```
lcd.print(" C");
}else{
// afficher de la pression
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Pression=");
lcd.setCursor(10, 0);
lcd.print(press);
lcd.setCursor(11, 0);
lcd.print(" BAR");
delay(3000);

// temp de pose
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("temp de pose=");
lcd.setCursor(13, 0);
lcd.print(tp);
lcd.setCursor(15, 0);
lcd.print("S");
delay(3000);

// nombre de cycles
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("cycles=");
lcd.setCursor(8, 0);
lcd.print(cycles);
delay(3000);

while(ta <ts){
te = analogRead(temppin);
tvolt = te * (0.00488);
ta = (tvolt / 0.01477982452) - 5;

lcd.clear();
lcd.setCursor(1, 0);
lcd.print("TA=");
lcd.setCursor(4, 0);
lcd.print(ta);
lcd.setCursor(7, 0);
lcd.print("C");

if(ta <ts){
digitalWrite(ress, HIGH);
}else{
digitalWrite(ress, LOW);

lcd.clear();
```

```
lcd.setCursor(0, 0);
lcd.print("1 2 3 . . .");
delay(2000);

digitalWrite(moteur, HIGH);
delay(3000);
digitalWrite(moteur, LOW);
cycles++;
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("MERCI");
delay(2000);

menuState = 0;
break; // quitter
}
}
}
}
delay(1000);
}
}
```

B – Coton :

La fonction `coton(int buttonState)` permet de configurer et de contrôler la température '**ts**', la pression '**press**', le temps de pose '**tp**' et le nombre de cycles '**cycles**' pour le tissu en coton dans la machine d'impression :

```
// 2EME choix: cotton
void coton(int buttonState){
  menuState = 2;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Coton:");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Temp=");
  lcd.setCursor(6, 0);
  lcd.print(ts);
  lcd.setCursor(8, 0);
  lcd.print(" C");

  while(menuState == 2){
    buttonState = analogRead(buttonPin);
    // saisie la température
    if(buttonState >= 50 && buttonState < 250){ // haut button pressé
      ts++;
      lcd.setCursor(6, 0);
      lcd.print(" ");
    }
  }
}
```

```
lcd.setCursor(6, 0);
lcd.print(ts);
}elseif(buttonState>= 250&&buttonState<400){ // bas button pressé
ts--;
lcd.setCursor(6, 0);
lcd.print(" ");
lcd.setCursor(6, 0);
lcd.print(ts);
}elseif(buttonState>= 400&&buttonState<600){ // gauche button pressé
menuState = 1;
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Tissu Type:");
delay(2000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Polyester/UP");
delay(3500);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Coton/RIGHT");
delay(3500);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Soie/DOWN");
delay(3500);
}elseif(buttonState<50){ // bouton droite
if(ts< min || ts> max){
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("FAILED");
delay(2000);

// retour au page précédent
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp=");
lcd.setCursor(6, 0);
lcd.print(ts);
lcd.setCursor(8, 0);
lcd.print(" C");
}else{

// afficher de la pression
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Pression=");
lcd.setCursor(10, 0);
lcd.print(press);
```

```
lcd.setCursor(11, 0);
lcd.print(" BAR");
delay(3000);

// temp de pose
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("temp de pose=");
lcd.setCursor(13, 0);
lcd.print(tp);
lcd.setCursor(15, 0);
lcd.print("S");
delay(3000);

// nombre de cycles
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("cycles=");
lcd.setCursor(8, 0);
lcd.print(cycles);
delay(3000);

while(ta < ts){
te = analogRead(temppin);
tvolt = te * (0.00488);
ta = (tvolt / 0.01477982452) - 5;

lcd.clear();
lcd.setCursor(1, 0);
lcd.print("TA=");
lcd.setCursor(4, 0);
lcd.print(ta);
lcd.setCursor(7, 0);
lcd.print("C");

if(ta < ts){
digitalWrite(ress, HIGH);
}else{
digitalWrite(ress, LOW);

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("1 2 3 . . .");
delay(2000);

digitalWrite(moteur, HIGH);
delay(3000);
digitalWrite(moteur, LOW);
cycles++;
```

```
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("MERCI");
delay(2000);

menuState = 0; // retour
break; // quitter
}
}
}
}
delay(1000);
}
}
```

C– Soie :

La fonction **soie (int buttonState)** est utilisée pour configurer automatiquement les paramètres adaptés au tissu en soie dans la machine d'impression. Elle permet de définir des valeurs spécifiques pour la température de fonctionnement ('ts'), la pression ('press'), le temps de pose ('tp'), et le nombre de cycles ('cycles').

```
// soie
voidsoie(intbuttonState){
  menuState = 2;
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("soie:");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Temp=");
  lcd.setCursor(6, 0);
  lcd.print(ts);
  lcd.setCursor(8, 0);
  lcd.print(" C");

  while(menuState == 2){
    buttonState = analogRead(buttonPin);

    // saisie la température
    if(buttonState>= 50&&buttonState<250){ // haut button pressé
      ts++;
      lcd.setCursor(6, 0);
      lcd.print(" ");
      lcd.setCursor(6, 0);
```



```
lcd.print(ts);
}elseif(buttonState>= 250&&buttonState<400){ // bas button pressé
ts--;
lcd.setCursor(6, 0);
lcd.print(" ");
lcd.setCursor(6, 0);
lcd.print(ts);
}elseif(buttonState>= 400&&buttonState<600){ // gauche button pressé
menuState = 1; // retour au menu
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Tissu Type:");
delay(2000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Polyester/UP");
delay(3500);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Coton/RIGHT");
delay(3500);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Soie/DOWN");
delay(3500);
}elseif(buttonState<50){ // bouton droite
if(ts< min || ts> max){
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("FAILED");
delay(2000);

// retour à la page précédente
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp=");
lcd.setCursor(6, 0);
lcd.print(ts);
lcd.setCursor(8, 0);
lcd.print(" C");
}else{

// afficher de la pression
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Pression=");
lcd.setCursor(10, 0);
lcd.print(press);
lcd.setCursor(11, 0);
```

```
lcd.print(" BAR");
delay(3000);

// temp de pose
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("temp de pose=");
lcd.setCursor(13, 0);
lcd.print(tp);
lcd.setCursor(15, 0);
lcd.print("S");
delay(3000);

// nombre de cycles
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("cycles=");
lcd.setCursor(8, 0);
lcd.print(cycles);
delay(3000);

while(ta < ts){
  te = analogRead(temppin);
  tvolt = te * (0.00488);
  ta = (tvolt / 0.01477982452) - 5;

  lcd.clear();
  lcd.setCursor(1, 0);
  lcd.print("TA=");
  lcd.setCursor(4, 0);
  lcd.print(ta);
  lcd.setCursor(7, 0);
  lcd.print("C");

  if(ta < ts){
    digitalWrite(ress, HIGH);
  }else{
    digitalWrite(ress, LOW);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("1 2 3 . . .");
    delay(2000);

    digitalWrite(moteur, HIGH);
    delay(3000);
    digitalWrite(moteur, LOW);
    cycles++;
    lcd.clear();
```

```
lcd.setCursor(0, 0);  
lcd.print("MERCI");  
delay(2000);  
  
menuState = 0;  
break;  
}  
}  
}  
}  
delay(1000);  
}  
}  
//fin d programme
```

III.4. Conclusion :

Cette étude vise à fournir une compréhension détaillée du fonctionnement prévu de la machine d'impression, tout en établissant une base solide pour la suite du projet. En décomposant chaque étape et en analysant le programme détaillé, nous avons créé un cadre fiable pour le développement et la maintenance de la machine d'impression. Ce travail constitue une étape préparatoire cruciale dans notre projet, garantissant une performance de haute qualité et la capacité de s'ajuster aux exigences futures.

Chapitre 4

Réalisation &

Applications Pratiques

IV. Chapitre 4 : Réalisation & Applications Pratiques

IV.1. Introduction :

Dans ce chapitre, nous allons détailler la conception et réalisation de la partie électrique et mécanique de notre projet. L'objectif principal est de fournir une description complète des composants utilisés, des schémas de montage, ainsi que des étapes de réalisation pour garantir une compréhension claire et exhaustive du système développé pour notre machine d'impression.

Ce chapitre sera divisé selon les étapes suivantes :

- Réalisation électronique
- Réalisation mécanique
- Teste et résultat

IV.2. Réalisation électronique :

IV.2.1. Schéma électrique :

La figure ci-dessous représente le circuit électrique d'une machine d'impression conçu à l'aide du programme PROTUS :

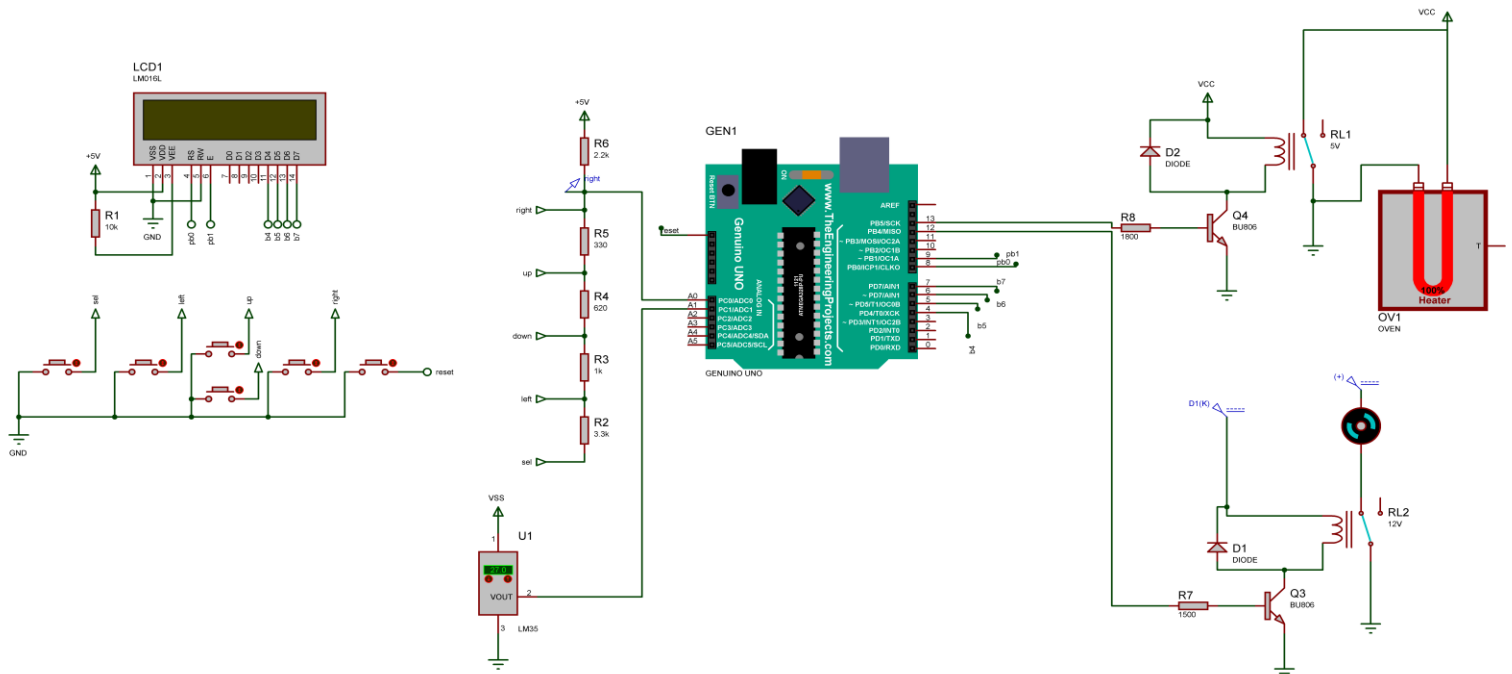


Figure IV-1 :Crcuit électrique

IV.2.2. Description des composants :

Nous allons détailler les composants clés utilisés dans notre projet. Chaque composant joue un rôle crucial dans le bon fonctionnement de la machine d'impression :

IV.2.3. La Sonde :

Les sondes sont des capteurs qui permettent de transformer une grandeur physique ici la température en une grandeur électrique ou un signal électrique. (47)



Figure IV-2 : Sonde de température (48)

Le champ d'application de ces sondes couvre une large plage de travail allant de -200 jusqu'à 1000 °C avec des résistances de quelques Ohms à plusieurs Kilo-ohms. (47)

Nous utilisons une sonde de température CTP pour notre projet :

IV.2.3.a/ Sonde de température CTP :

La sonde de température à CTP (Coefficient de Température Positif) est un dispositif de mesure de la température qui utilise des semi-conducteurs. Sa résistance varie en fonction de la température : elle augmente de manière uniforme avec l'élévation de la température. (49)

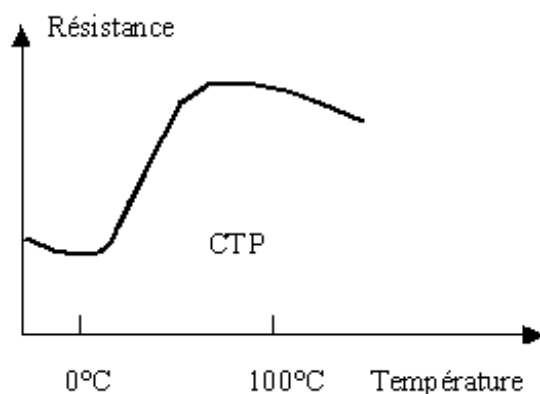


Figure IV-3 : L'évolution de la résistance d'une sonde CTP en fonction de la température $U_R = f(T)$

La résistance d'une sonde en cuivre en fonction de la température peut être exprimée par la formule suivante : (50)

$$R_t = R_0(1 + \alpha \cdot t)$$

R_t : Résistance en ohms à la température t

R_0 : la résistance à 0°C

α : est le coefficient de température du cuivre ($0,00393^\circ\text{C}^{-1}$)

t : est la température en $^\circ\text{C}$

IV.2.3.b/ **Caractéristiques :**

Les caractéristiques de la Sonde de température PTC sont : (51)

- Résistance interne $R_S=3.9\text{ k}\Omega$
- Sonde cylindre.
- Plage de -40 à $+100^\circ\text{C}$.
- Boîtier en cuivre.
- Thermo-conductrice et résistante à l'humidité
- Très précise.

IV.2.4. **Transistor :**

Un transistor est un composant électronique qui peut fonctionner comme un interrupteur ou un amplificateur. Il est constitué de matériaux semi-conducteurs et dispose de trois bornes : la base (B), le collecteur (C) et l'émetteur (E). En contrôlant le courant ou la tension à la borne de la base, on peut réguler le courant entre le collecteur et l'émetteur, ce qui permet de contrôler des signaux électriques plus grands. (52)

Il existe deux types de transistor : (52)

IV.2.4.a/ Types de Transistor :

• Transistors bipolaires :

- 1- **NPN** : Le courant circule du collecteur (N) vers l'émetteur (N) lorsque la base (P) est alimentée.
- 2- **PNP** : Le courant circule de l'émetteur (P) vers le collecteur (P) lorsque la base (N) est alimentée

• Transistors à effet de champ :

- 1 - **JFET** (en anglais : Junction Field Effect Transistor)
- 2- **MOSFET** (en anglais : Metal-Oxide-Semiconductor Field Effect Transistor) :
 - a - MOSFET à enrichissement
 - b - MOSFET à appauvrissement

NB : Nous utilisons pour notre projet un transistor NPN BU806 :

Ce transistor a été utilisé car nous avons une application à faible puissance et à fréquence moyenne.

IV.2.4.b/ Transistor NPN :

Le transistor NPN tire son nom de la disposition de ces couches : une couche de matériau de type N (négatif), suivie d'une couche de matériau de type P (positif), puis à nouveau d'une couche de type N. (53)



Figure IV-2 : BU806 Transistor (54)

IV.2.5. Relais :

Un relais est un dispositif électromécanique utilisé pour commuter un circuit électrique en utilisant un signal de commande. Le relais fonctionne comme un

interrupteur contrôlé à distance, permettant de contrôler des charges électriques plus importantes avec un petit signal de commande. (55)

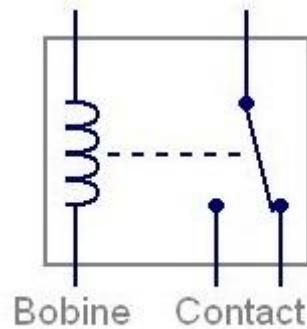


Figure IV-3 : Symbole d'un relais (56)

NB : Nous utilisons pour notre projet un relais Omron G5LC-1 :

IV.2.5.a/Relais OMRON G5LC-1 :

Le relais Omron G5LC-1 est un relais unipolaire, normalement ouvert (en anglais : SPST - Single Pole Single Throw), compact de faible puissance et conçu pour des applications de commutation générales. (57)



Figure IV-4 : Relais G5LC-1 (58)

IV.2.5.b/ Caractéristiques :

- Tension de la bobine : 12 v DC/220v AC
- Capable de commuter des charges AC et DC : 10A
- Offre une bonne isolation entre la bobine et les contacts

IV.2.6.Résistance Chauffante :

Resistance chauffante est un composant électrique conçu pour convertir l'énergie électrique en chaleur. (59)

NB : Nous utilisons pour notre projet une bougie préchauffage d'auto.



Figure IV-5 : bougie préchauffage

IV.2.7.Moteur centralisé : (Actionneur)

Le moteur centralisé 24V est un type de moteur électrique utilisé principalement dans les systèmes de contrôle automatisé et les systèmes de verrouillage pour les applications domestiques et industrielles. (60)



Figure IV-6 : Moteur centralisé (60)

IV.2.8.Montage électronique réel :

Celle-ci est le montage électronique réel :

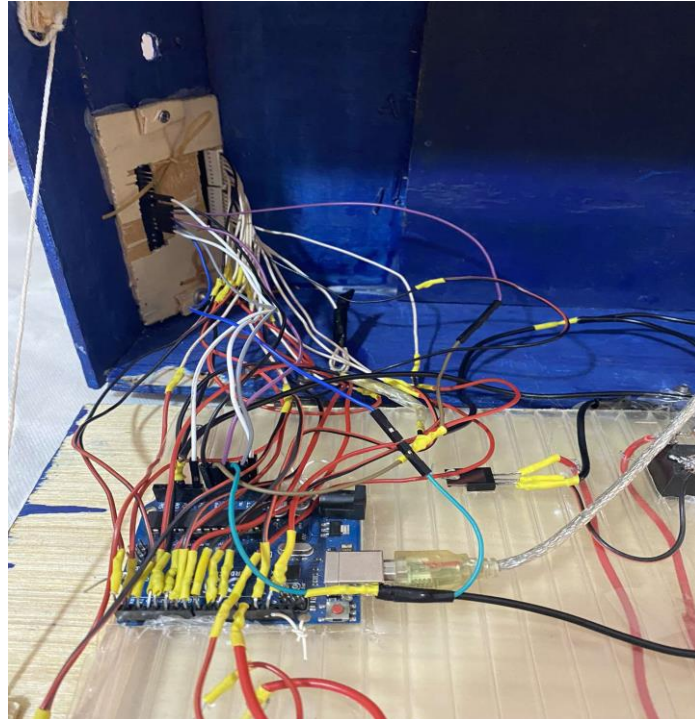


Figure IV-7 : Montage réel ARDUINO-LCD SHIELD

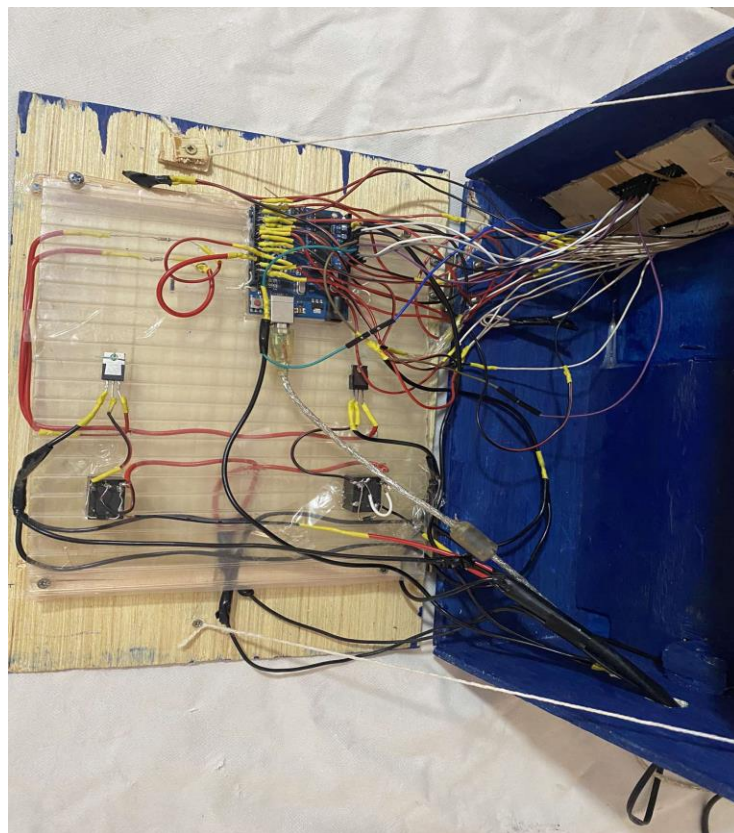


Figure IV-8 : Montage électronique complète

IV.3. Réalisation mécanique :

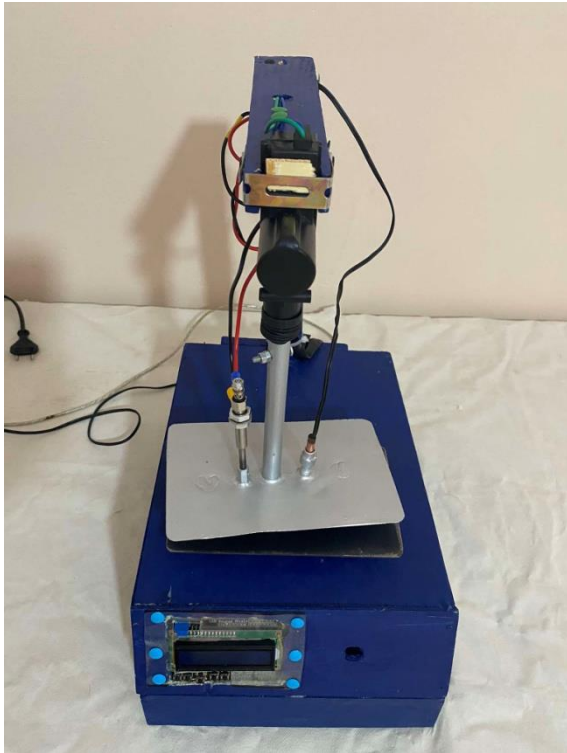


Figure IV-10 : Vue de face

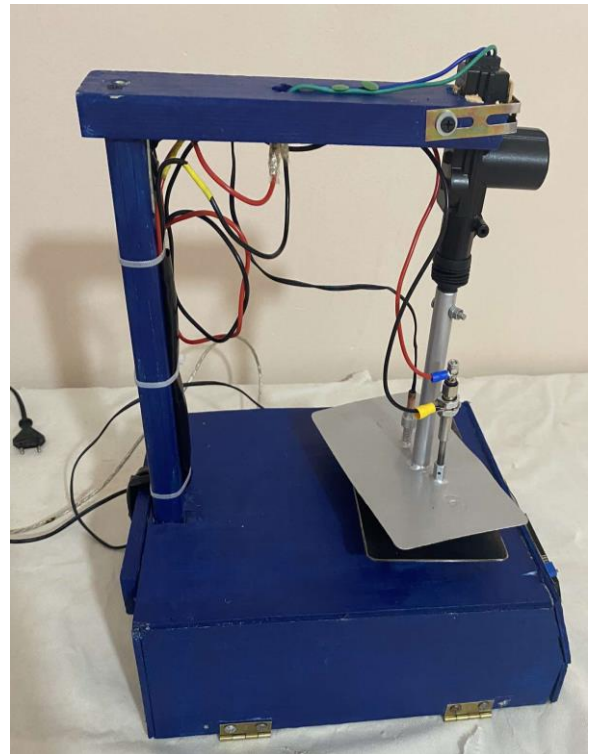


Figure IV-9: Vue à coté

IV.4. Teste & résultat :

- Après avoir connecte l'Arduino avec le PC Par un câble spéciale et Mettez l'appareil sous tension, LCD va allumez et afficher un message « WELCOME » :



Figure IV-11 : Etape 1

- Il existe 3 types de tissu (polyester/coton/soie) et peuvent être sélectionnés avec les boutons (up-right-down) :



Figure IV-12 : Etape 2

- Entrez la valeur de température :



Figure IV-13 : Etape 3

- Affichiez les valeurs de « pression, temps de pose et nombre de cycles »



Figure IV-14 : Etape 4

- Affichiez la température actuelle « ta » :



Figure IV-15 : Etape 5

- Actionneur descende :

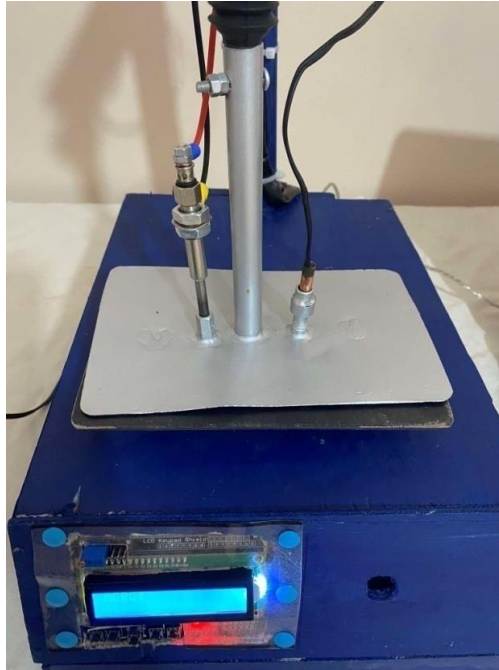


Figure IV-16 : Descende d'actionneur

IV.5. Conclusion :

Dans ce chapitre, nous avons présenté la réalisation complète de notre machine semi-automatique contrôlée par Arduino.

Cela inclut la création d'un mode d'emploi pour une utilisation facile, le montage électronique détaillé avec des photos réelles des composants intégrés, ainsi que le montage mécanique, où nous avons assuré la solidité et l'efficacité de la structure. Les tests effectués, accompagnés de photos réelles, ont validé le bon fonctionnement de l'ensemble, confirmant que la machine répond aux spécifications techniques et est prête pour une utilisation opérationnelle.

Conclusion Générale

v. Conclusion générale :

Pour conclure ce mémoire, nous avons exploré en profondeur la conception et la réalisation d'une machine d'impression semi-automatique contrôlée par Arduino. Ce projet a été guidé par une série d'étapes techniques et pratiques, démontrant l'application concrète des connaissances en automatique, électronique et programmation.

Dans le 1^{er} chapitre, nous avons examiné les Automates Programmables, ou PLC, qui sont des dispositifs spécialisés utilisés principalement pour le contrôle et l'automatisation des processus industriels. Nous avons détaillé leur architecture, leurs capacités de traitement, et leur utilisation dans des environnements industriels pour la surveillance et le contrôle des machines et des lignes de production.

Dans le 2^{eme} chapitre, Nous avons continué par définir l'Arduino comme une plateforme centrale dans le domaine du prototypage électronique, soulignant son rôle essentiel en tant que contrôleur principal de notre machine. L'interface Arduino a été minutieusement étudiée, mettant en lumière l'interaction entre ses divers composants tels que l'écran LCD et les boutons de contrôle.

Le 3^{eme} chapitre a porté sur le programme Arduino et l'organigramme détaillé qui décrit le fonctionnement du système. Ce chapitre a mis en lumière la logique de contrôle et les algorithmes développés pour gérer les différentes tâches de la machine d'impression. En conclusion, ce mémoire témoigne d'un parcours d'apprentissage significatif dans les domaines de l'automatique & informatique, marqué par la création réussie d'une machine d'impression grâce à l'intégration efficace de la technologie Arduino & l'interface homme machine IHM.

Dans le 4^{ème} chapitre, nous avons détaillé la réalisation électronique et mécanique de la machine. Le schéma électrique a été conçu avec Proteus, incluant des composants comme la sonde de température, le relais, le transistor, et la bougie chauffante. Le montage électronique a ensuite été réalisé, suivi de la conception mécanique, garantissant un assemblage fonctionnel et fiable de la machine.

Bibliographie

Bibliographie :

1. F., Manual. Les Automates Programmables Industriels (API). *technologuepro*. [En ligne] 18-11-2017. <https://www.technologuepro.com/cours-automate-programmable-industriel/Les-automates-programmables-industriels-API.htm>.
2. Introduction aux A.P.I / ELEC13. [En ligne] <https://elec13.wordpress.com/2016/03/06/introduction-aux-a-p-i/>.
3. Kadem Chérifa, Ouail Isma. Automatisation et télégestion d'une station de pompage. Société des Eaux et d'Assainissement d'Alger : Techniques de l'ingénieur, 2017, Vol. 8.
4. Qu'est ce qu'une API ? *open-prod*. [En ligne] open-prod. <https://open-prod.com/lexique/api-definition/>.
5. INTRO AUX API'S. [En ligne] <https://elec13.wordpress.com/2016/03/06/introduction-aux-a-p-i/>.
6. GONZAGA, Alain. *LES AUTOMATES PROGRAMMABLES INDUSTRIELS*. [En ligne] https://www.geea.org/IMG/pdf/LES_AUTOMATES_PROGRAMMABLES_INDUSTRIELS_pour_GEEA.pdf.
7. [En ligne] <https://www.pop-industrie.com/automate-compact-logo-230-v-didactique-produit-8308-11>.
8. [En ligne] https://www.memoireonline.com/02/12/5423/m_Renovation-de-la-soudeuse-Soudronic-de-latelier-de-fabrication-des-fts-neufs21.html.
9. AUTOMATES PROGRAMMABLES INDUSTRIELS (A.P.I). [En ligne] <https://tech3elec.education.tn/API.html>.
10. LES AUTOMATES PROGRAMMABLES INDUSTRIELS (API) . *Automatismes* . [En ligne] https://www.uvt.rnu.tn/resources-uvt/cours/Automatismes/chapitre2_api.pdf.
11. Oussama, LEKHAL. Conception d'une prise de pièces automatisée à la base d'une API supervisée par un IHM. BLIDA : s.n., 2018-2019.

12. [En ligne] https://fac.umc.edu.dz/ista/pdf/cours/L3-GIM_R%C3%A9seau%20automatescours1_Programmation%20des%20API.pdf.
13. AUTOMATION. [En ligne] <https://www.automation-sense.com/blog/automatisme/le-langage-ladder-pour-lesnuls.html>.
14. LADDER. *TECHNOLOGIE*. [En ligne] <https://sii-technologie.ac-normandie.fr/IMG/pdf/ladder.pdf>.
15. [En ligne] https://www.researchgate.net/figure/Quelques-symboles-de-langage-Ladder_fig32_349039684.
16. TIA PORTAL V12. [auteur du livre] Yousfi Lounis Mellali Sofyane. *étude de l'automatisation et de la supervision d'un procédé de lavage de filtres Niagara à CEVITAL*. université Abderrahmane MIRA de Bejaia : s.n.
17. [En ligne] http://philippe.berger2.free.fr/automatique/cours/G7/le_grafcet.htm.
18. [En ligne] https://product-help.schneider-electric.com/Machine%20Expert/V1.1/fr/SoMProg/SoMProg/FBD_LD_IL_Editor/FBD_LD.
19. [En ligne] https://product-help.schneider-electric.com/Machine%20Expert/V1.1/fr/SoMProg/SoMProg/FBD_LD_IL_Editor/FBD_LD_IL_Editor-4.htm.
20. TAVERNIER, CHRISTAN. *Arduino Maitriser sa programmation et ses cartes d'interface*. PARIS : édition DUNOD, 2014.
21. Abderrezzak, Yahi. *Surveillance à distance de la consommation d'énergie via un système IoT*. Université Aboubakr Belkaïd – Tlemcen : s.n., 2017-2018.
22. ASTALASEVEN, ESKIMON et OLYTE. *Arduino pour bien commencer en électronique et en programmation*. [En ligne] duzero.com.
23. [En ligne] <https://fr.wikipedia.org/wiki/Arduino>.
24. [En ligne] <https://www.ionos.fr/digitalguide/serveur/know-how/projets-arduino/#:~:text=de%20cookies%20ici.->

,Station%20m%C3%A9t%C3%A9o%20(professionnel),%2C%20et%20affiche%20 l'heure..

25. Djebri Abdelkader, Bellache Okba. *Réalisation d'une station météo a base d'une carte arduino*. SAAD DAHLEB UNIVERSITE - BLIDA : s.n., 2013-2014.

26. [En ligne] https://www.memoireonline.com/10/20/11898/m_Station-meteorologique-sur-arduino3.html.

27. ZIOUECHE DJAWED, MEBREK SALAH EDDINE. *Réalisation d'un capteur de présence piloté par Arduino*. Université Aboubakr Belkaïd, Faculté de Technologie Tlemcen : s.n., 2016.

28. [En ligne] <https://www.algeria.ubuy.com/fr/product/3V6KZCHAG-arduino-uno-r3-compatible-atmega328p-ch340-usb-microcontroller-board>.

29. I. Bradai, G. Roumaïssa. « *Etude, Simulation d'un Thermomètre Numérique à base d'Arduino* ». Université Ain Temouchent : s.n., 2021.

30. ISLAM, ZAR HAMZA & MELLOUK. *Conception et réalisation d'un contrôleur de température automatique par carte Arduino*. Université Aboubakr Belkaïd – Tlemcen : s.n., 2022.

31. Cotenceau, B. Carte ARDUINO UNO Microcontrôleur ATMega328. Polytech Angers : s.n., 2019-2020.

32. [En ligne] <https://www.indiamart.com/proddetail/atmega328-microcontrollers-ic-22855326748.html>.

33. BENHALIMA Oussama, KHEDIM Youcef. *REALISATION D'UN TUNNEL DE DESINFECTION CONTRE COVID 19 ALIMENTE PAR UN SYSTEME MULTISOURCES*. UNIVERSITÉ IBN-KHALDOUN DE TIARET : DÉPARTEMENT DE GENIE ELECTRIQUE-AII, 2019/2020.

34. [En ligne] <https://www.cours-gratuit.com/cours-arduino/tutoriel-environnement-de-programmation-arduino-ide>.

35. arduino blaise pascal. [En ligne] <https://arduino.blaise-pascal.fr/presentation/logiciel/>.

36. [En ligne] <https://www.auditsi.eu/?p=7085>.

37. [En ligne] <https://github.com/arduino/arduino-ide/releases/tag/2.3.2>.
38. [En ligne] <https://technospace.wordpress.com/2012/06/24/serrure-codee-avec-pic-16f877/>.
39. [En ligne] <https://www.aranacorp.com/fr/gerez-un-ecran-lcd-16x2-avec-arduino/>.
40. LCD, Les écrans. Using 1602 LCD Keypad Shield W/ Arduino [+Practical Projects]. [En ligne] VENDREDI 31 MARS 2023.
https://zestedesavoir.com/tutoriels/686/arduino-premiers-pas-en-informatique-embarquee/748_laffichage-une-autre-maniere-dinteragir/3443_les-ecrans-lcd/.
41. [En ligne] <https://www.carnetdumaker.net/articles/utiliser-une-shield-lcd-de-dfrobots-avec-une-carte-arduino-genuino-et-la-bibliotheque-liquidcrystal/>.
42. Les écrans LCD alphanumériques. [En ligne]
<https://www.locoduino.org/spip.php?article97>.
43. [En ligne] <https://www.carnetdumaker.net/articles/utiliser-une-shield-lcd-de-dfrobots-avec-une-carte-arduino-genuino-et-la-bibliotheque-liquidcrystal/>.
44. MOUSSAOUI Abdelhak, TOUATI Youcef. *Presse à chaud pneumatique* . Dr Yahia Fares Université de Médéa : s.n., 2022/2023.
45. [En ligne] https://aptlovevs.shop/product_details/59076632.html.
46. SPI. [En ligne]
https://www.facebook.com/permalink.php/?story_fbid=1398560307101971&id=1394407720850563.
47. Types de sondes de températures en climatisation. [En ligne]
<https://www.abcclim.net/sondes-ptc-ntc-climatisation-chauffage.html#:~:text=Les%20sondes%20sont%20des%20capteurs,thermocouples%2C%20les%20sondes%20%C3%A0%20r%C3%A9sistance..>
48. [En ligne] <https://www.directindustry.fr/fabricant-industriel/capteur-temperature-ptc-162791.html>.

49. [En ligne] chrome-extension://efaidnbnmnnibpcajpcglclefindmkaj/https://www.lyc-valdedurance.ac-aix-marseille.fr/extra/projet_ballon2008/mesures/CTN.pdf.
50. Temperature Coefficient of Resistance. [En ligne] <https://byjus.com/jee/temperature-coefficient-of-resistance/>.
51. SONDE NTC. [En ligne] <https://www.guilcor.fr/sonde-a-thermistance-ntc/5818-capteur-de-temperature-ntc-a-boitier-en-cuivre.html>.
52. *Le transistor bipolaire*. limoges : Ecole nationale supérieure d'ingénieurs de limoges ENSIL.
53. [En ligne] <https://alltransistors.com/transistor.php?transistor=23394>.
54. [En ligne] https://www.alldatasheet.com/view.jsp?Searchword=Bu806&gad_source=1&clid=Cj0KCQjwzva1BhD3ARIsADQuPnWHq1VLJDpartfC3UDovmvITASLLFiGcFQsV9mSfiCnZG7EVIJ5p7waAo-QEALw_wcB.
55. Leibson, Steve. Comparaison entre les contacteurs et les relais électromécaniques : comment choisir l'outil adapté à l'application. [En ligne] 24 07 2018. <https://www.digikey.fr/fr/articles/electromechanical-relays-versus-contactors#:~:text=Les%20relais%20%C3%A9lectrom%C3%A9caniques%20permettent%20de,commande%20et%20la%20puissance%20commut%C3%A9e..>
56. [En ligne] https://sti2d.ecolelamache.org/le_relais_lectromcanique.html.
57. [En ligne] <https://rtmfm.cnrs.fr/wp-content/uploads/2022/01/Tutoriel-11-comprendre-les-relais.pdf>.
58. [En ligne] <https://www.ebay.co.uk/itm/323595500436>.
59. [En ligne] <https://www.refsas.com/tout-savoir-sur-les-resistances-electriques-chauffantes/#:~:text=La%20r%C3%A9sistance%20chauffante%20est%20donc,par%20conduction%20ou%20par%20rayonnement..>
60. [En ligne] <https://www.amazon.fr/syst%C3%A8me-verrouillage-Central-voiture-serrure/dp/B0CCKWN1PS>.

61. Michel, GILLES. « *Architecture et application des automates programmables* ». Paris : DUNOD, 1988.
62. [En ligne] [https://www.ionos.fr/digitalguide/serveur/know-how/projets-arduino/#:~:text=de%20cookies%20ici,-,Station%20m%C3%A9t%C3%A9o%20\(professionnel\),%2C%20et%20affiche%20l'heure..](https://www.ionos.fr/digitalguide/serveur/know-how/projets-arduino/#:~:text=de%20cookies%20ici,-,Station%20m%C3%A9t%C3%A9o%20(professionnel),%2C%20et%20affiche%20l'heure..)
63. Oussama, Ayad Benyounes & Tari. *Simulation et réalisation d'un circuit détecteur de gaz interfacé à la carte Arduino*. TLEMCEM : Université Abou bakrBelkaïd, Faculté de Technologie, 2019/2020.
64. Djebbari Abdeldkader, Bellache Okba. *Réalisation d'une station météo a basé d'une carte ARDUINO-UNO*. Université Saad Dahlab BLIDA : s.n., 2013/2014.
65. amine, BOUMESHED Mohamed. *Réalisation d'un PID à base ARDUINO*. BLIDA : Université SAAD DAHLAB de BLIDA, Faculté de Technologie, 2014.
66. ZERROUKI Mohamed Amine, NESNAS Riadh. *Conception et réalisation d'un système de commande d'une habitation*. TIZI OUZOU : Université Mouloud Mammeri Faculté de genie électrique et informatique , 2018.
67. arduino., Documentation. [En ligne] <https://www.arduino.cc>.
68. LED ET CALCUL DE LA RÉSISTANCE SÉRIE. [En ligne] 29 06 2011. <https://www.astuces-pratiques.fr/electronique/led-et-calcul-de-la-resistance-serie>.
69. Cours : Les afficheurs LCD. *sciencesappliquees*. [En ligne] <https://sciencesappliquees.com/noncato/346-cours-les-afficheurs-lcd>.
70. Sonde NTC. [En ligne] <https://www.guilcor.fr/sonde-a-thermistance-ntc/5818-capteur-de-temperature-ntc-a-boitier-en-cuivre.html>.
71. [En ligne] <https://eib-dz.com/>.
72. [En ligne] <https://www.999mitsubishi.com/mitsubishi-fx1n-fx1n-40mr-es>.
73. Djebbari Abdelkader, Bellache Okba. *Réalisation d'une station météo a base d'une carte arduino*. Saad Dahleb Université - BLIDA : s.n., 2013/2014.

74. [En ligne] <https://arduino.blaisepascal.fr/ecran-lcd/>.

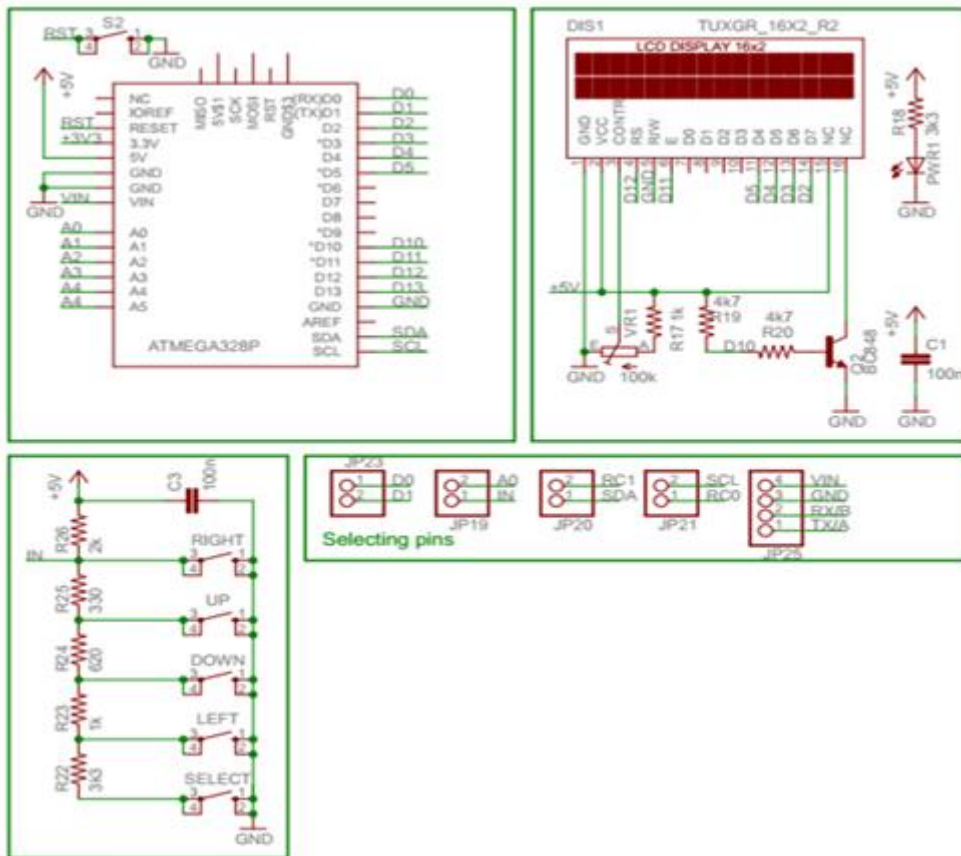
75. [En ligne] <https://www.tinkercad.com/things/iO8njOsvB4L-grand-blorryhabbi/editel?sharecode=CEPAL98sPtdFnN1kFoeRerYhaJ4LKx4pJahE8f4OSzU>.

76. guilcor. [En ligne] <https://www.guilcor.fr/sonde-a-thermistance-ntc/5818-capteur-de-temperature-ntc-a-boitier-en-cuivre.html>.

77. L'Automate Programmable Industriel. [En ligne] <https://bpmeiprades.com/cours/la-partie-commande/lessons/fonctionnement-dun-a-p-i/>.

ANNEXE

3 – Circuit diagramme des boutons de LCD SHIELD



4 – DATASHEET de la diode BU806 :

MAXIMUM RATINGS

Rating	Symbol	BU806	Unit
Collector–Emitter Voltage	V_{CEO}	200	Vdc
Collector–Emitter Voltage	V_{CEV}	400	Vdc
Collector–Base Voltage	V_{CBO}	400	Vdc
Emitter–Base Voltage	V_{EBO}	6.0	Vdc
Collector Current — Continuous	I_C	8.0	Adc
— Peak		15	
Emitter–Collector Diode Current	I_F	10	Adc
Base Current	I_B	2.0	Adc
Total Device Dissipation, $T_C = 25^\circ\text{C}$	PD	60	Watts
Derate above $T_C = 25^\circ\text{C}$		0.48	
Operating and Storage Junction Temperature Range	T_J, T_{stg}	-65 to 150	$^\circ\text{C}$