

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur  
et de la Recherche Scientifique  
Université Akli Mohand Oulhadj - Bouira -  
Tasdawit Akli Muḥend Ulḥağ - Tubirett -



وزارة التعليم العالي والبحث العلمي  
جامعة أكلي محمد أولحاج  
- البويرة -

Faculté des Sciences et des Sciences Appliquées

كلية العلوم والعلوم التطبيقية

M<sup>R</sup>éférence : ...../MM/2021

المرجع : .....م / م / 2021

## Mémoire de Master

Présenté au

Département : Génie Électrique

Domaine : Sciences et Technologies

Filière : Electronique

Spécialité : Electronique des systèmes embarqués

Réalisé par :

MILOUDI Mohamed

Et

SALHI Badr Edine

## Thème

Réalisation D'une Maquette Pédagogique Electronique  
Numérique à base d'un Microcontrôleur PIC

Soutenu le: 03/07/2023

Devant la commission composée de :

Mr :	Fekik Arezki	M.C.A	Univ. Bouira	Président
	Bouhedda Ali	M.A.A	Univ. Bouira	Examineur
	Haroun Smail	M.C.B	Univ. Bouira	Encadreur

Année Universitaire: 2022-2023

## **Remerciements**

Ce travail a été effectué au sein du Département des Sciences et sciences appliquées de l'Université de Bouira.

Nous sommes reconnaissants envers notre encadreur H. Smail

Nous souhaitons adresser nos remerciements aux personnes qui nous ont apporté leurs aides, spécialement nos camarades et amis Mustapha et Omar.

Et nous souhaitons remercier spécialement nos parents à qui nous avons dédié ce travail, ainsi que toute notre famille.

Je tiens à adresser mes remerciements à mes parents pour m'avoir supporté, je tiens à remercier mon frère Smail pour son grand soutien, aussi ma sœur Nadjat et aussi mes amis Farouk, Slimane, Redouane, Wissem et Sara.

## **Résumé**

Le microcontrôleur joue un rôle crucial dans notre vie quotidienne, que ce soit en tant qu'étudiants en électronique, amateurs ou professionnels de l'électronique. Il est largement utilisé dans divers secteurs industriels et constitue une ressource essentielle pour les projets de fin d'études. Notre objectif est de concevoir une maquette électronique numérique basée sur le microcontrôleur PIC de la famille 18Fxxx, plus précisément le PIC 18F4550. Dans le premier chapitre, nous présenterons et définirons le microcontrôleur PIC 18F4550, ainsi que ses caractéristiques principales. Le deuxième chapitre se concentrera sur la conception de notre maquette, en présentant les schémas utilisés et les différents modules composant notre projet. Enfin, dans le troisième chapitre, nous aborderons la réalisation de la maquette et la programmation associée.

## **Abstract**

The microcontroller plays a crucial role in our daily lives, whether as electronics students, enthusiasts, or professionals in the field. It is widely used in various industrial sectors and serves as an essential resource for end-of-studies projects. Our goal is to design a digital electronic prototype based on the PIC microcontroller from the 18Fxxx family, specifically the PIC 18F4550. In the first chapter, we will introduce and define the PIC 18F4550 microcontroller, along with its key features. The second chapter will focus on the design of our prototype, presenting the schematics used and the different modules that make up our project. Finally, in the third chapter, we will discuss the implementation of the prototype and the associated programming.

# Table des Matières

Remerciements .....	I
Résumé .....	II
Table des Matières .....	III
Liste des Figures.....	VI
Liste des Tableaux.....	VII
Listes des Acronymes.....	VIII

## **Introduction Générale** **1**

### **Chapitre 1 : Présentation du microcontrôleur PIC 18F4550**

I.Introduction .....	2
I.1 Définition de microcontrôleur.....	2
I.2 Les avantages du microcontrôleur .....	2
I.3 Le choix d'un microcontrôleur.....	3
I.4 Définition de pic.....	4
I.4.1 Classification des pics .....	4
I.4.2 Identification du PIC .....	4
I.5 Pic 18F4550 .....	5
I.5.2 Caractéristiques principales du PIC .....	6
I.5.3 Les ports d'entrées/sorties parallèles.....	6
I.5.4 Organisation de la mémoire .....	7
I.5.4.1 La mémoire de programme .....	7
I.5.4.2 La mémoire de données.....	7
I.5.5 Alimentation du PIC 18F4550 .....	7
I.5.6 L'horloge.....	8
I.5.6.a L'horloge interne .....	8
I.5.6.a L'horloge interne .....	8
I.5.7 Le circuit Reset.....	9
I.5.8 Le Timer chien de garde (WDT).....	10
I.5.9 Le mode SLEEP .....	10
I.5.10 Le port séries synchrone maître ou MSSP .....	11
I.5.10.a Le MSSP pour SPI (Serial Peripheral Interface) .....	11
I.5.10.b Le MSSP pour I2C (Inter Integrated Circuit).....	11
I.5.11 L'USART (Universal Synchronous Asynchronous Receiver Transmitter) .....	11
I.5.12 L'USB (Universal Serial Bus).....	12
I.5.13 Convertisseur Analogique/Numérique .....	12

I.5.14 Les Timers.....	12
I.6 Conclusion.....	12

## **Chapitre 2 : Etude de la maquette électronique numérique**

II.1. Introduction.....	13
II.2. Schémas synoptique.....	13
II.3. Carte de développement .....	13
II.3.1. Connexion du PIC sur la carte .....	14
II.3.2. USB HID Bootloader.....	14
II.3.3. Connexion du RESET .....	15
II.3.4. ICSP (In-Circuit Serial Programming) .....	15
II.4. Module d'affichage.....	16
II.4.1. Afficheur LCD.....	16
II.4.1.1. Présentation .....	16
II.4.1.2. Principe de fonctionnement .....	17
II.4.1.3. Les différentes broches d'un LCD.....	17
II.4.1.4. Schéma de l'afficheur LCD sur la carte.....	18
II.4.2. Afficheurs 7 Segments 4 digits.....	18
II.4.3. Connecteur GLCD .....	19
II.5. Module d'entrées/sorties.....	20
II.5.1. Le clavier .....	20
II.5.1.1. Principe de fonctionnement .....	20
II.5.1.2. Schémas du clavier sur la carte.....	21
II.5.2. Matrice des boutons .....	21
II.5.3. Matrice de LEDs.....	22
II.5.4. Les entrées analogiques .....	23
II.5.4.1. LDR (photo résistance).....	23
II.5.4.2. Potentiomètres .....	23
II.5.4.3. LM35 .....	25
II.6. Module commande moteurs .....	25
II.6.1. L293D Contrôleur de moteur CC .....	25
II.6.2. ULN2003A Contrôleur de moteur pas à pas .....	27
II.6.3. Adaptateur de Servomoteurs.....	27
II.6.4. Relais à 2 canaux .....	28
II.7. Module d'alimentation (5V et 12V) .....	30
II.8. Conclusion.....	30

## **Chapitre3 : Réalisation et Programmation de la maquette électronique numérique**

III.1. Introduction .....	31
---------------------------	----

III.2. Coté hardware.....	31
III.2.1 Module de développement .....	31
III.2.2 Module d’affichage .....	33
III.2.3 Module d’entrées/sorties .....	34
III.2.4 Module de commandes moteurs.....	36
III.2.5 Carte d’alimentation .....	37
III.3.1 Installer le bootloader dans le PIC 18F4550.....	39
III.4 Conclusion.....	42
<b>Conclusion Générale</b>	<b>43</b>
<b>Références</b>	<b>44</b>
<b>Annexe</b>	<b>46</b>

# Liste des Figures

Figure I.1 Brochage du 18F4550.....	5
Figure I.2 Architecture interne du PIC.....	7
Figure I.3 Brochage de l'horloge externe du pic 18F4550 .....	8
Figure I.4 Les différents types de Reset .....	9
Figure I.5 : Brochage de SPI.....	11
Figure I.6 : Brochage de l'USART.....	11
Figure II.1 connexion du pic sur la carte.....	14
Figure II.2 : USB HID Bootloader sur Proteus.....	15
Figure II.3 : bouton RESET sur Proteus.....	15
Figure II.4 : connexion du ICSP sur Proteus.....	15
Figure II.5 : Carte de développement à base de PIC 18F4550 sur Proteus.....	16
Figure II.6 : schéma d'un LCD.....	17
Figure II.7 : schéma de l'afficheur LCD dans le module d'affichage sur Proteus.....	18
Figure II.8 : Schéma de l'afficheur 7 segments 4 digits sur Proteus.....	19
Figure II.9 : Connecteur GLCD sur Proteus.....	19
Figure II.10 : Module d'affichage sur Proteus.....	20
Figure II.11 : Architecture interne du clavier.....	20
Figure II.12 : Schéma du clavier sur Proteus.....	21
Figure II.13 : Schéma d'installation des boutons.....	22
Figure II.14 : matrice des boutons sur Proteus.....	22
Figure II.15 : matrice de LEDs sur Proteus.....	23
Figure II.16 : LDR sur Proteus.....	23
Figure II.17 : potentiomètre sur Proteus.....	24
Figure II.18 : LM35 sur Proteus.....	24
Figure II.19 : Module des entrées/sorties sur Proteus.....	25
Figure II.20 : L293D.....	25
Figure II.21 : Schéma du pilote L293D sur Proteus.....	26
Figure II.22 : ULN2003A.....	27
Figure II.23 : ULN2003A sur Proteus.....	27
Figure II.24 : Servomoteur.....	28
Figure II.25 : adaptateur de 2 servomoteurs sur Proteus.....	28
Figure II.26 : Relais.....	29

Figure II.27 : module a relais sur Proteus.....	29
Figure II.28 : Module commande de moteurs sur Proteus.....	29
Figure II.29 : schéma de l'alimentation stabilisée sur Proteus .....	30
Figure III.1 : Schéma du PCB de la carte développement sur Proteus .....	31
Figure III.2 : Image du circuit imprimé de la carte de développement.....	32
Figure III.3 : Image de la carte de développement.....	32
Figure III.4 : Schéma du PCB de la carte d'affichage sur Proteus.....	33
Figure III.5 : Image du circuit imprimé du module d'affichage .....	33
Figure III.6 : Image du module d'affichage.....	34
Figure III.7 : schéma PCB de la carte de I/O .....	34
Figure III.8 : Image du circuit imprimé du module de I/O.....	35
Figure III.9 : image du module de I/O.....	35
Figure III.10 : schéma PCB de la carte de commande de moteurs.....	36
Figure III.11 : Image du circuit imprimé du module de commandes de moteurs.....	36
Figure III.12 : Image de la carte de commandes de moteurs.....	37
Figure III.13 : schéma PCB de la carte d'alimentation.....	37
Figure III.14 : circuit imprimé de la carte d'alimentation .....	38
Figure III.15 : Image de la carte d'alimentation.....	38
Figure III.16 : connexion du pic sur le Pickit 3 clown.....	39
Figure III.17 : application du programmeur.....	40
Figure III.18 : importation du fichier Hex.....	40
Figure III.19 : le fichier du HID Bootloader.....	41
Figure III.20 : La fin de chargement du HID bootloader.....	41

## **Liste des Tableaux**

Tableau I.1. Tableau comparatif.....	5
Tableau II.1 : Brochage de l'interface moteur L293D.....	26



## Listes des Acronymes

PIC : Peripheral Interface Controller.

UAL : Unité Arithmétique et Logique.

CMOS : Complementary metal-oxide-semiconductor.

UVPROM : Ultra Violet Programmable Read Only Memory.

OTPROM : One-Time Programmable Read-Only Memory.

EEPROM : Electrically-Erasable Programmable Read-Only Memory.

EPROM : Erasable Programmable Read Only Memory.

MHZ : mégahertz.

E/S ,I/O : entrées/sorties.

DIL : Dual In-Line package.

VSS : the voltage applied to the transistor Source pin.

VDD : voltage applied to the Drain of the transistors.

VCC : voltage applied to the Collector of the transistors.

Mbps : mega byte per second.

LCD : Liquid Crystal Display.

LED : Light-Emitting Diode.

LDR : light dependent resistor.

USB : Universal Serial Bus.

HID : Human Interface Device.

ASCII : American Standard Code for Information Interchange.

GND : Ground.

PCB : printed circuit board.

# Introduction Générale

Une maquette pédagogique à base d'un PIC est un système qui associe un ensemble de fonction fonctions (Prog., RS232, I2C) gérées par un microcontrôleur de la famille PIC 16F877.[1]

Nous réalisons une maquette pédagogique à base d'un PIC 18f4550 qui est choisi pour ça haute performance et son port USB intégré. Nous avons choisi de faire cette maquette modulaire pour permettre à l'étudiant de faire les manipulation et la connexion des composants lui même.

Alors le but de ce projet est de réaliser une maquette pédagogique électronique numérique à base d'un microcontrôleur PIC 18f4550 pour permettre à l'étudiant de manipuler des applications de base et de réaliser des études des systemes électroniques basées sur le UC.

Dans le premier chapitre, on à présenter les microcontrôleurs spécifiquement le 18f4550, on à vu tous ces caractéristiques.

Dans le 2ème chapitre on à conçu les différents modules de la maquette en utilisant ISIS Proteus.

Dans le 3 ème chapitre on a présenter la maquette réaliser et on a programmé la maquette pour qu'elle soit prête à l'emploi

# Chapitre I :

## Présentation du microcontrôleur

### PIC 18F4550

#### **I.Introduction :**

Un objet technique contenant de l'électronique révèle souvent une fonction dont le rôle est de traiter une information : une opération arithmétique (addition, multiplication...) ou logique (ET, OU...) entre plusieurs signaux d'entrée pour qu'un signal de sortie puisse être généré. Ces fonctions peuvent être réalisées par des circuits analogiques ou logiques. Cependant, lorsque l'objet technique devient complexe et nécessite un grand nombre d'opérations de traitement de l'information, il devient simple d'utiliser la structure basée sur le microcontrôleur PIC.

#### **I.1 Définition de microcontrôleur:**

Un microcontrôleur est un dispositif de traitement de données de type microprocesseur qui comprend tous les éléments essentiels d'un système informatique, tels qu'un microprocesseur, des mémoires et des périphériques (tels que des ports, des timers, des convertisseurs, etc.). Chaque fabricant dispose généralement de sa propre gamme de microcontrôleurs. [1]

Une famille de microcontrôleurs partage généralement un noyau commun, tel que le microprocesseur et le jeu d'instructions. Les fabricants peuvent proposer une grande variété de broches, qui s'adaptent plus ou moins bien à certaines tâches. Toutefois, un programmeur qui maîtrise une famille de microcontrôleurs n'a pas besoin de connaître la façon d'utilisation de chaque membre tout seul. Il suffit de connaître les différences par rapport au membre principal de la famille. Ces différences peuvent inclure la taille de la mémoire, la présence ou l'absence de périphériques, ainsi que leur nombre. [1]

#### **I.2 Les avantages du microcontrôleur:**

Les microcontrôleurs sont une solution avantageuse pour la programmation de circuits à plusieurs points, avec des avantages bien réels. Il suffit d'observer l'évolution remarquable de l'offre proposée par les fabricants de circuits intégrés dans ce domaine ces dernières années pour en être convaincu. Nous allons examiner comment la multitude de microcontrôleurs disponibles découle de la logique la plus élémentaire.

- Dans un premier temps, un microcontrôleur rassemble dans un seul boîtier ce qui auparavant nécessitait une dizaine d'éléments séparés. Cela se traduit par une réduction notable de l'encombrement du matériel et du circuit imprimé.
- De plus, cette intégration a pour conséquence immédiate de simplifier la conception du circuit imprimé car il n'est plus nécessaire de transmettre des bus d'adresses et de données d'un composant à l'autre.
- Grâce à la diminution du nombre de composants, la fiabilité du système s'améliore car le nombre de connexions entre les composants, les supports ou les circuits imprimés diminue.
- Les microcontrôleurs contribuent à réduire les coûts à plusieurs niveaux, car ils sont moins chers que les autres composants qu'ils remplacent et permettent de réduire les coûts de main-d'œuvre.
- Les microcontrôleurs permettent également la réalisation d'applications qui ne seraient pas réalisables avec d'autres composants.

### **I.3 Le choix d'un microcontrôleur :**

La plupart des grands fabricants de circuits intégrés proposent aujourd'hui plusieurs gammes de microcontrôleurs, toutes plus performantes les unes que les autres, du moins selon leurs publicités. Il est donc légitime de se demander quelle famille de circuits à choisir, en particulier pour un industriel qui doit développer une application. Cette question est d'ailleurs souvent au centre de ses préoccupations.

Dans notre cas, étant donné que nous ne sommes pas des industriels, notre travail est simplifié. En effet, nous devons simplement prendre en compte les critères suivants :

- Le ou les circuits de la famille doivent être facilement accessibles sur le marché amateur.
- Le prix des circuits doit être abordable pour tous les budgets.
- La programmation de la mémoire morte interne (celle qui contient le programme) doit être facile.
- Enfin, les outils de développement (nous verrons dans un instant de quoi il s'agit) doivent être aussi peu coûteux que possible.

Par chance, les microcontrôleurs de la famille PIC de Microchip répondent actuellement le mieux à ces critères et connaissent un succès planétaire, étant très largement utilisés dans l'industrie. En les choisissant, nous bénéficions donc des avantages qui en découlent, notamment un vaste choix de références, une excellente disponibilité et un faible coût unitaire.

Le présent chapitre se concentre sur l'étude du microcontrôleur, qui constitue l'élément fondamental de notre maquette pédagogique.

Un microcontrôleur est un circuit intégré qui renferme en interne, dans un seul boîtier, tous les éléments qui constituent un micro-ordinateur complet. Sa structure interne inclut notamment une Unité Arithmétique et Logique (UAL), des ports d'entrées/sorties, des interfaces de communication série, des interfaces d'entrées/sorties analogiques, ainsi que des timers et une horloge temps réel. [2]

#### **I.4 Définition de pic :**

Actuellement, les microcontrôleurs sont largement utilisés dans de nombreuses applications grand public et professionnelles, et il existe plusieurs familles de ces composants. Dans les années 90, la société américaine Microchip Technology a développé un microcontrôleur CMOS appelé le PIC (Peripheral Interface Controller). Ce composant est toujours très répandu aujourd'hui grâce à son équilibre entre la facilité d'utilisation, la rapidité et le coût abordable. [1]

Il existe plusieurs versions des PIC, notamment les UVROM qui peuvent être effacés par une source de rayonnements ultraviolets, les OTPROM programmables une seule fois, ainsi que les EEPROM et flash EPROM qui peuvent être effacés électriquement. [1]

##### **I.4.1 Classification des pics: [2]**

Les modèles de microprocesseurs actuellement disponibles sur le marché peuvent être regroupés en trois familles principales, chacune comportant plusieurs références. Ces familles sont les suivantes :

- Base-line : les instructions sont encodées sur 12 bits.
- Mid-range : les instructions sont encodées sur 14 bits.
- High-end : les instructions sont encodées sur 16 bits.

##### **I.4.2 Identification du PIC:**

En général, un microcontrôleur PIC est identifié par une référence composée des éléments suivants : xx(L)XXyy-zz.

- Les deux premiers caractères "xx" indiquent la famille du composant, qui peut être "12, 14, 16, 17 ou 18".
- La lettre "L" optionnelle représente une tolérance de plage de tension plus importante.
- Les deux caractères suivants "XX" représentent le type de mémoire programme :
  - "C" indique une EPROM ou une EEPROM.
  - "F" indique une mémoire flash.
- Les deux caractères "yy" sont un identificateur spécifique au composant.
- Les deux derniers caractères "zz" indiquent la vitesse maximale du quartz de pilotage.

Le tableau suivant montre la différence entre quelques PICs. [2]

	Mem Prog en octets	SRAM en octets	EEPROM en octets	Fréquence max en MHZ	E/S	Boitier
18F2550	64K	2048	256	40	24	28
18F4550	64K	2048	256	40	35	40
16F877	114K	368	256	20	33	40
16F84	14336	224	64	20	13	18
12C508	6144	25	-	4	6	8

Tableau I.1. Tableau comparatif. [3]

### I.5 Pic 18F4550 :

Le 18F4550 classé en High-end de la société Microchip est équipé de ressources matérielles internes riches ainsi que d'une mémoire vive et d'une mémoire de programme très importantes. Il est particulièrement adapté à la programmation en langage C. Il se présente sous la forme d'un boîtier DIL (Dual In-Line Package) en plastique opaque comportant 40 broches, dont 33 sont utilisées pour les entrées/sorties, deux paires de broches pour l'alimentation (VSS et VDD), une broche pour la réinitialisation et deux broches pour l'oscillateur, une pour l'entrée et l'autre pour la sortie. Si l'oscillateur externe n'est pas utilisé, cette dernière peut être configurée en tant qu'entrée/sortie. [4]

La figure ci-dessous représente le brochage du 18F4550 :

#### 40-Pin

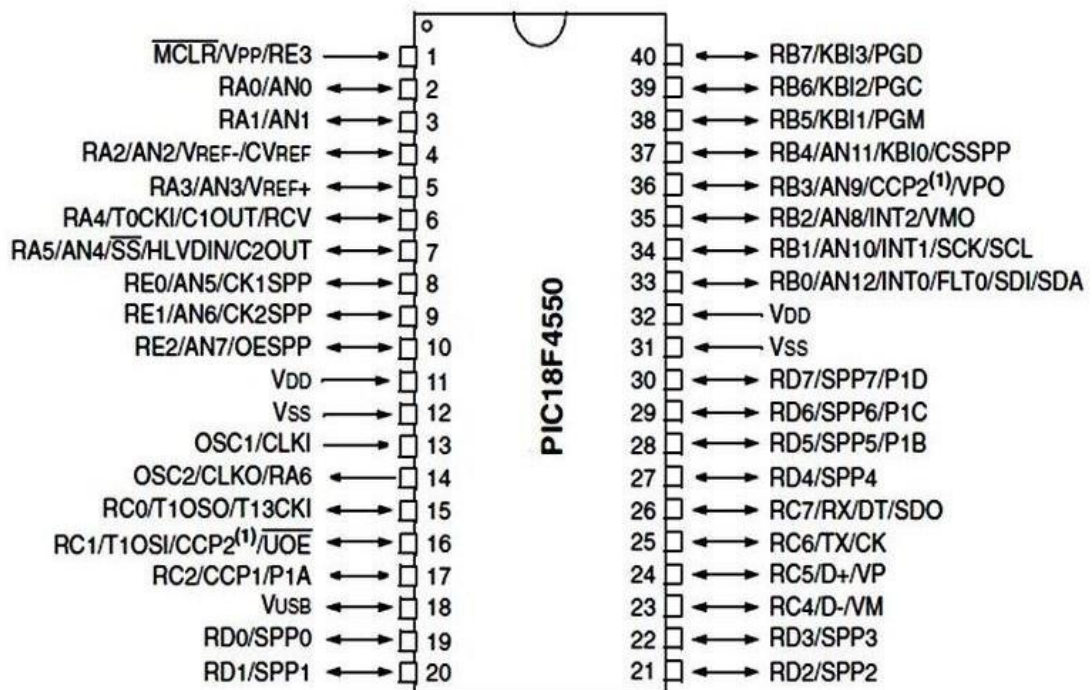


Figure I.1 Brochage du 18F4550

### **I.5.2 Caractéristiques principales du PIC :**

Le 18F4550 dispose d'une mémoire de programme de type FLASH qui peut résister à 100 000 cycles d'effacement/écriture, ainsi qu'une mémoire de données capable de supporter 1 000 000 cycles d'effacement/écriture. Les données stockées dans ces mémoires peuvent être conservées pendant au moins 40 ans. De plus, ce microcontrôleur est capable de fonctionner avec un oscillateur pouvant atteindre une fréquence de 48 MHz. [4]

Le microcontrôleur est équipé de trois sources d'interruptions externes (INTCON, INTCON2, INTCON3), qui peuvent être hiérarchisées selon leur niveau de priorité. Il est également doté de treize (13) entrées analogiques, qui permettent de convertir des signaux analogiques en signaux numériques. [4]

Le microcontrôleur est pourvu de trois modules de transmission série : le MSSP (Master Synchronous Serial Port) qui peut fonctionner en deux modes, SPI (4 fils) et I2C (3 fils), l'USART (RS232 et RS485) et l'USB.

Le microcontrôleur dispose d'un code de protection qui peut être activé pour prévenir la lecture et la copie frauduleuse de son programme à partir d'un programmeur externe. En outre, il prend en charge la programmation en circuit sous forme série (ICSP), ce qui permet de modifier le contenu de sa mémoire de programme sans avoir à le retirer de l'application où il est installé. [4]

### **I.5.3 Les ports d'entrées/sorties parallèles :**

Le PIC 18F4550 est un microcontrôleur de la famille des PICs de Microchip qui dispose de plusieurs ports d'entrée/sortie numériques. Chacun de ces ports peut être configuré en tant qu'entrée ou sortie en fonction des besoins de l'application.

**-Le port A :** est un port d'entrée/sortie numérique à 8 bits, qui est également multiplexé avec les entrées analogiques et peut être utilisé comme entrée analogique.

**-Le port B :** est un port d'entrée/sortie numérique à 8 bits, avec des broches ayant des fonctions spéciales telles que la broche RB7 qui est utilisée pour la programmation ICSP.

**-Le port C :** est un port d'entrée/sortie numérique à 8 bits, avec des broches ayant des fonctions spéciales telles que la broche RC6 qui est utilisée pour l'horloge externe.

**-Le port D :** est un port d'entrée/sortie numérique à 8 bits, avec des broches ayant des fonctions spéciales telles que la broche RD5 qui est utilisée pour le module de communication USB.

**-Le port E :** est un port d'entrée/sortie numérique à 3 bits, avec des broches ayant des fonctions spéciales telles que la broche RE0 qui est utilisée pour le module de communication USB.

Les registres de direction TRISA, TRISB, TRISC, TRISD et TRISE sont utilisés pour configurer les ports. Pour définir un port comme entrée, le bit correspondant doit être réglé sur 1 (I=in), tandis que pour un port de sortie, le bit correspondant doit être réglé sur 0 (0=out).

Il est possible de programmer les ports A et E en tant qu'entrées analogiques, à l'exception des broches RA4 et RA6. Le port B contient trois entrées pour les interruptions externes (RB0, RB1, RB2). Le port D peut être utilisé comme un port parallèle esclave (SPP). [4]

#### I.5.4 Organisation de la mémoire :

Le microcontrôleur PIC dispose d'une mémoire de programme et d'une mémoire de données. La structure Harvard des PICs permet un accès distinct à chacune de ces mémoires.

**I.5.4.1 La mémoire de programme :** La mémoire de programme du PIC 18F4550 est destinée à contenir le programme à exécuter et a une capacité de 32 Kbytes. Chaque instruction est codée sur deux octets, permettant de stocker 16K mots d'instructions. La mémoire flash est utilisée pour insérer le programme en utilisant un programmeur.

**I.5.4.2 La mémoire de données :** La mémoire de données du PIC 18F4550 est divisée en deux parties distinctes : une RAM de 2048 octets et une zone EEPROM de 256 octets. La RAM contient les registres de fonctions spéciales (SFRs), qui sont utilisés pour contrôler les opérations sur le circuit. La zone EEPROM contient des registres généraux qui sont disponibles pour une utilisation libre. [4]

La figure I.2 montre schéma simplifié d'un pic:

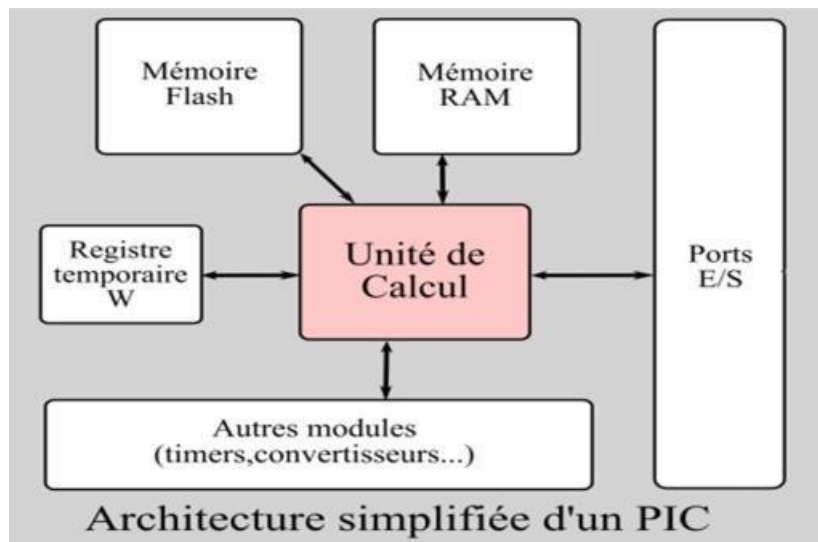


Figure I.2 Architecture interne du PIC.

#### I.5.5 Alimentation du PIC 18F4550 :

Le PIC 18F4550 nécessite une tension d'alimentation de 4,2 à 5,25 V pour un fonctionnement normal. Sa consommation de courant est inférieure à 1,6 mA sous 5V avec une horloge de 4 MHz et de 25  $\mu$ A sous 3V avec une horloge de 32 KHz. Sa RAM peut conserver son contenu même lorsque le circuit est en mode veille, avec une tension d'alimentation allant jusqu'à 1,5 V. Pour alimenter le



circuit, il faut appliquer la tension d'alimentation positive aux broches Vdd (11 et 32) et la masse aux broches Vss (12 et 31), en utilisant un condensateur de 22 à 100 nF de type céramique multicouche pour découpler les broches d'alimentation. [4]

### I.5.6 L'horloge :

Le PIC est équipé d'une horloge interne et peut également fonctionner avec une horloge externe. Les deux modes peuvent être utilisés simultanément si nécessaire. [4]

**I.5.6.a L'horloge interne :** Le PIC dispose d'un bloc oscillateur qui permet de générer deux horloges de fréquences différentes, utilisables de manière indépendante ou combinées. La première horloge utilise un oscillateur de fréquence 8 MHz, qui peut être réduite jusqu'à 31 KHz à l'aide d'un diviseur programmable. La deuxième horloge fonctionne à une fréquence fixe de 31 KHz et utilise un oscillateur RC interne. La première horloge peut être utilisée comme horloge principale si l'USB n'est pas utilisé. [4]

**I.5.6.b L'horloge externe :** Le PIC peut également être utilisé avec une horloge externe, qui est connectée aux broches OSC1/CLK1 (broche 13) et OSC2/CLK2 (broche 14). Il existe trois schémas de base pour l'utilisation de différentes horloges externes :

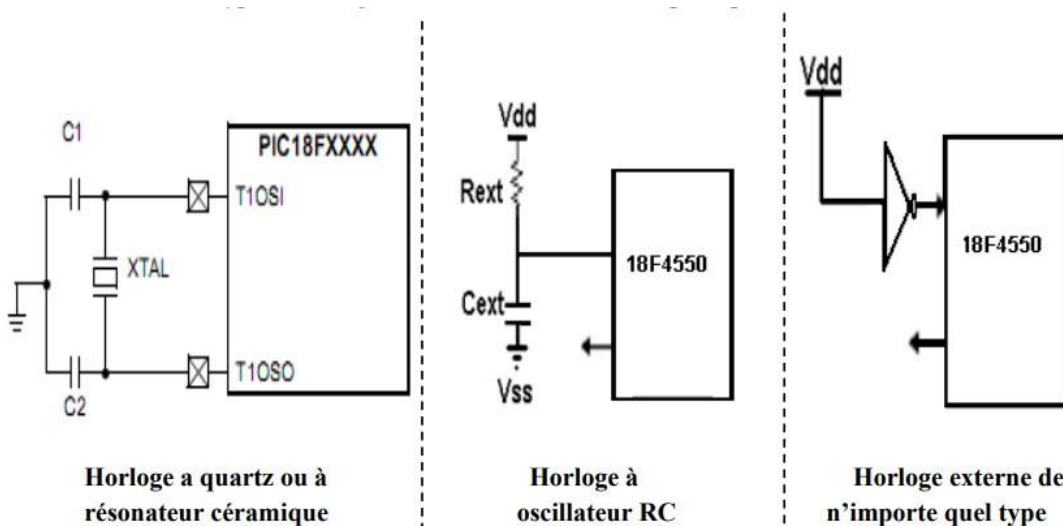


Figure I.3 Brochage de l'horloge externe du pic 18F4550

Le microcontrôleur PIC 18F4550 offre une grande flexibilité en termes d'horloges utilisables, avec huit types d'horloges différents:

- LP (Low Power) : horloge à quartz basse vitesse pour une faible consommation de courant.
- XT : pour tout quartz de fréquence comprise entre une centaine de KHz et 4MHz.
- HS (High Speed) : horloge à quartz ou à résonateur céramique haute vitesse pour toute fréquence >4MHz.
- HSPPL : horloge à quartz ou à résonateur céramique haute vitesse avec boucle de verrouillage de phase PLL interne.

- RC ou External RC : horloge pilotée par un ensemble de résistances et capacités connecté sur l'entrée CLKI, avec une sortie de signal d'horloge à une fréquence divisée par 4 sur la sortie CLKO.
- RCIO : horloge pilotée par un ensemble résistance capacité externe connecté sur l'entrée CLKI, tandis que CLK0 (RA6) sert d'entrée/sortie à usage générale.
- EC : horloge utilisant un signal d'horloge externe appliqué à CLKI, avec une sortie de signal d'horloge à une fréquence divisée par 4 sur la sortie CLK0 (RA6).
- ECIO : source d'horloge externe appliquée sur CLKI, avec CLK0 (RA6) comme entrée/sortie du port parallèle.

Pour choisir l'horloge appropriée parmi les huit options disponibles, l'utilisateur doit configurer les 3 bits correspondants (FOSC2, FOSC1, FOSC0) dans le registre de configuration (CONFIGIH). Les logiciels de programmation de PICs fournissent un panneau de configuration qui permet à l'utilisateur de choisir le type d'horloge souhaité (LP, XT, etc.) et de manipuler les bits de configuration en conséquence.

### I.5.7 Le circuit Reset :

Le processus de réinitialisation est un élément important du fonctionnement d'un système. Lorsqu'un système rencontre une erreur, la capacité de le réinitialiser est souvent nécessaire pour résoudre le problème. Il peut également être utile de réaliser une réinitialisation lors de la mise sous tension. Cette opération est réalisée en appliquant un signal logique "0" à la patte 1 du microcontrôleur. Il existe différents types de réinitialisation, tels que la réinitialisation manuelle, automatique et mixte. [4]

Les différents circuits de RESET sont figurés en-dessous :

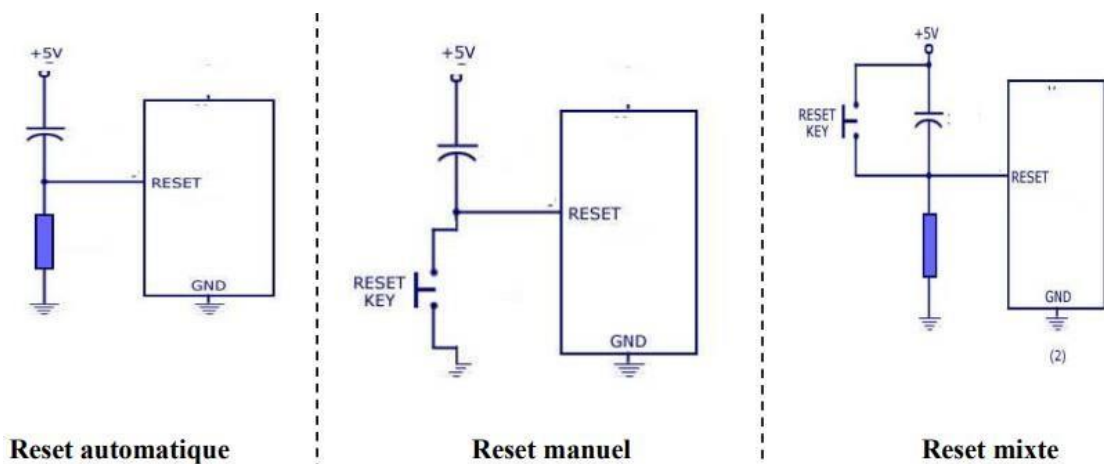


Figure I.4 Les différents types de Reset

Le microcontrôleur PIC possède plusieurs sources potentielles de réinitialisation, qui sont au nombre de huit, à savoir :

- Un reset à la mise sous tension du circuit, appelé POR (Power On Reset).
- Un reset déclenché par l'action sur la pin MCLR en mode normal.
- Un reset déclenché par l'action sur la pin MCLR en mode sommeil.
- Un reset provoqué par le débordement du timer chien de garde (WDT).
- Un reset déclenché par la détection d'une chute anormale de la tension d'alimentation, même temporaire, appelé BOR (Brown Out reset).
- Un reset provoqué par l'exécution de l'instruction de même nom (reset).
- Un reset déclenché par la saturation de la pile.
- Un reset provoqué par un débordement inférieur de la pile, lorsqu'une tentative de retirer une donnée de la pile est effectuée alors qu'elle est vide.

### **I.5.8 Le Timer chien de garde (WDT) :**

Le système de surveillance intégré dans le microcontrôleur permet de vérifier le bon déroulement du programme et de le protéger contre les blocages. Ce système est basé sur un compteur qui est incrémenté à un rythme défini par une horloge RC interne indépendante de l'horloge système. En temps normal, ce compteur est réinitialisé régulièrement pendant l'exécution du programme. Cependant, en cas de dysfonctionnement ou de blocage du programme, le compteur peut arriver à sa valeur maximale et provoquer un débordement. Dans ce cas, deux situations peuvent se produire :

- Si le microcontrôleur fonctionne normalement, le Watchdog Timer (WDT) provoque une réinitialisation (RESET).
- Si le microcontrôleur est en mode veille (SLEEP), le WDT provoque une sortie de veille (WAKE-UP), et l'exécution du programme reprend à l'endroit où elle s'était arrêtée. [4]

### **I.5.9 Le mode SLEEP :**

L'instruction SLEEP permet de mettre le microcontrôleur PIC en mode de sommeil afin de réduire sa consommation. Lorsque le microcontrôleur est en mode SLEEP, l'horloge système est arrêtée, ce qui entraîne l'arrêt de l'exécution du programme en cours. Pour sortir du mode de sommeil, il est nécessaire de provoquer un signal WAKE-UP en utilisant l'une des trois possibilités suivantes :

- Une réinitialisation externe due à la mise à 0 de l'entrée MCLR du PIC.
- Un débordement du Watchdog Timer (WDT), si celui-ci est validé.
- Une interruption générée par les broches RB0, RB1 ou RB2, si l'interruption est autorisée. [4]

### **I.5.10 Le port séries synchrone maître ou MSSP :**

Le mode MSSP (Master Synchronous Serial Port) est un port série qui permet de communiquer avec différents périphériques tels que des mémoires EEPROM série, des convertisseurs A/D, etc. Le module MSSP est capable de fonctionner en mode SPI ou I2C. [4]

**I.5.10.a Le MSSP pour SPI (Serial Peripheral Interface) :** Le mode MSSP utilisé pour le protocole SPI (Serial Peripheral Interface) permet de transmettre et recevoir des données de manière synchrone et simultanée sur 8 bits. Pour assurer la communication, trois broches sont utilisées :

- La broche de sortie série de données (SD0) - RC7.
- La broche d'entrée série de données (SDI) - RB0/SDA.
- La broche d'horloge série (SCK) - RB1/SCL/LVDIN.

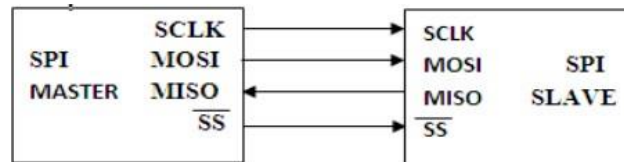


Figure I.5 : Brochage de SPI.

**I.5.10.b Le MSSP pour I2C (Inter Integrated Circuit) :** Le mode MSSP utilisé pour le protocole I2C (Inter-Integrated Circuit) permet de transmettre des données en tant que maître ou esclave en utilisant uniquement deux fils : la broche de transfert de données (SDA) pour le transfert de données (broche 33) et la broche d'horloge (SCL) pour l'horloge (broche 34). [4]

**I.5.11 L'USART (Universal Synchronous Asynchronous Receiver Transmitter) :**

Il s'agit du module EUSART (Enhanced Universal Synchronous Asynchronous Receiver Transmitter), également appelé SCI (Serial Communication Interface). Ce module permet la transmission série en mode asynchrone et synchrone. En mode asynchrone, il peut communiquer en full duplex avec des dispositifs tels que les micro-ordinateurs. En mode synchrone, il peut communiquer en half-duplex avec des périphériques externes tels que les mémoires EEPROM série. Le module utilise la patte RC6/TX pour l'émission de données et la patte RC7/RX pour la réception.

L'USART peut être configuré comme une interface :

- Asynchrone (Full-duplex).
- Synchrone-Master (Half-duplex).
- Synchrone-Slave (Half-duplex).

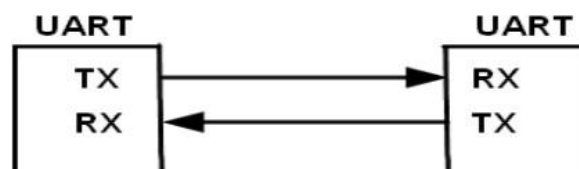


Figure I.6 : Brochage de l'USART.

**I.5.12 L'USB (Universal Serial Bus):**

L'USB (Universal Serial Bus) est une norme qui permet la connexion de périphériques externes via un port série compatible plug-and-play. Sa version 2.0 permet des débits allant jusqu'à 480 Mbps. Pour la transmission des données, la pin 24 (DATA+) est utilisée pour l'émission et la pin 23 (DATA-) est utilisée pour la réception. [4]

#### **I.5.13 Convertisseur Analogique/Numérique :**

Le convertisseur analogique-numérique (CAN) du PIC est composé de 13 entrées analogiques, dont 8 se trouvent sur le port A (RA0-RA7) et les 5 autres sur le port E (RE0-RE3). Le résultat de la conversion est encodé sur 10 bits. Ce CAN peut fonctionner même lorsque le PIC est en mode sommeil, en utilisant une horloge RC interne pour effectuer la conversion. [4]

#### **I.5.14 Les Timers:**

Tous les microcontrôleurs sont équipés d'un ou plusieurs Timers pour la gestion du temps. Le PIC18F4550 dispose de quatre Timers, à savoir Timer0, Timer1, Timer2 et Timer3, chacun étant de 8 bits ou de 16 bits. Le Timer est un circuit périphérique intégré dans le PIC, qui est activé par le programme et qui compte un nombre prédéterminé d'impulsions du quartz. Ces impulsions peuvent provenir du quartz qui cadence le microcontrôleur, ce qui permet un fonctionnement en tant que compteur de temps, ou d'une entrée TOR du PIC (par exemple, RA4 pour le Timer0, RC0 pour le Timer1), ce qui permet un fonctionnement en tant que compteur d'événement extérieur. Une fois que le nombre d'impulsions est atteint, un drapeau (flag) est activé. Les Timers ont tous un compteur (counter) et un pré-compteur (prescaler) presque identiques pour leur fonctionnement. [4]

#### **I.6 Conclusion :**

En conclusion, le microcontrôleur PIC 18F4550 est un composant électronique très puissant, offrant de nombreuses fonctionnalités et des performances de haut niveau. Grâce à ses 4 timers, ses nombreuses entrées/sorties, son convertisseur A/D et ses modules de communication série tels que l'USART, l'I2C, le SPI et l'USB, il peut être utilisé dans une variété d'applications électroniques, allant de la domotique à l'automatisation industrielle en passant par la robotique et les systèmes embarqués. Il est également facile à programmer, grâce à un grand choix de langages de programmation tels que le langage C, BASIC et d'autres environnements de développement intégrés. De plus, la large disponibilité de ressources telles que des tutoriels, des exemples de code et des bibliothèques de périphériques facilite grandement l'apprentissage et la mise en œuvre du PIC 18F4550.

En somme, le microcontrôleur PIC 18F4550 est une excellente solution pour les ingénieurs et les développeurs cherchant une solution de traitement embarquée fiable, puissante et facile à utiliser.

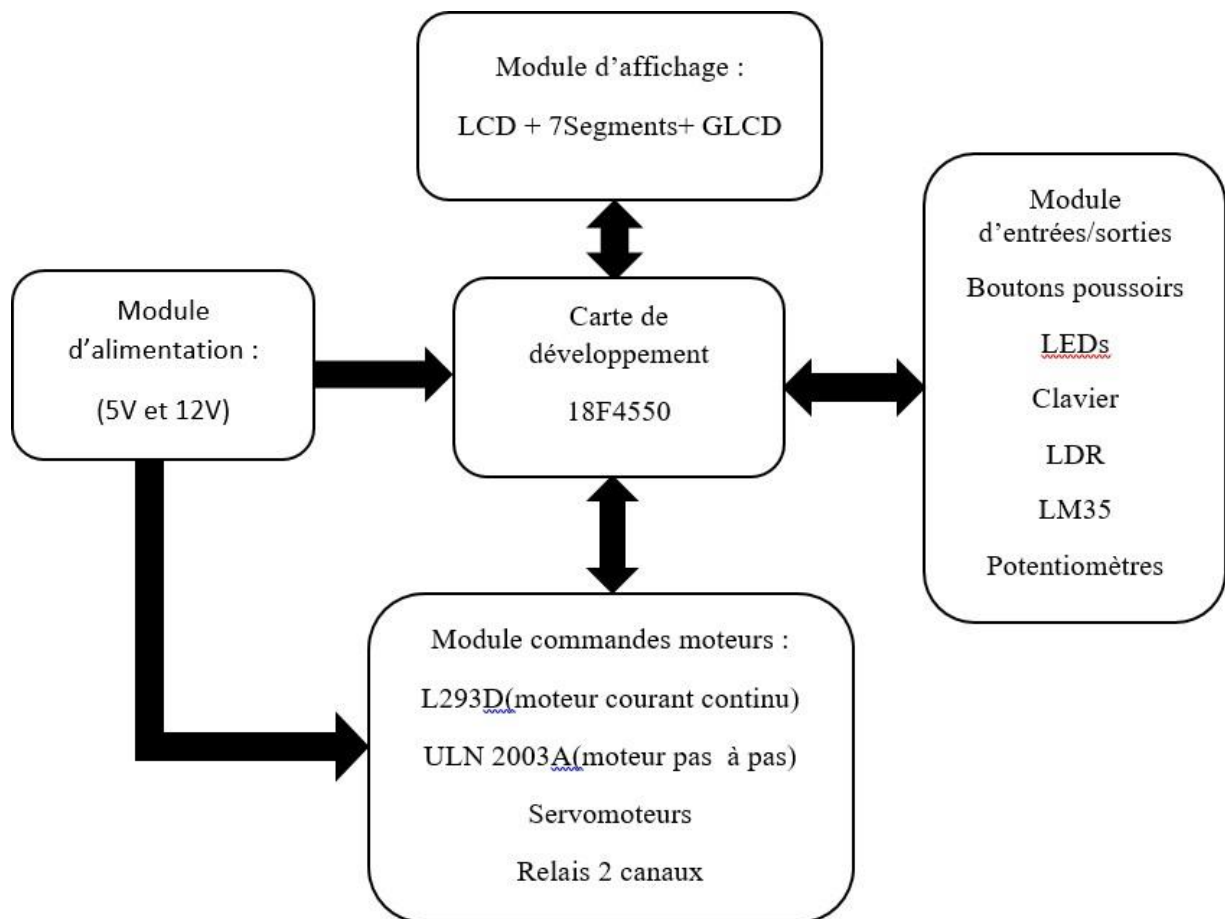
## Chapitre II:

# Etude de la maquette électronique numérique

### II.1. Introduction :

Dans ce chapitre, nous exposons en détail la solution choisie pour répondre aux spécifications de notre cahier des charges. Nous abordons la conception détaillée de chaque composant du système, visant ainsi à obtenir une schématisation complète et précise.

### II.2. Schémas synoptique :



### II.3. Carte de développement :

Une carte de développement basée sur le PIC 18F4550 est une carte électronique utilisée pour prototyper et développer des applications basées sur le microcontrôleur PIC 18F4550. Dans notre carte on a choisi Le PIC 18F4550 car est un microcontrôleur offrant une variété de fonctionnalités et d'interfaces intégrées. La programmation de cette carte de développement est réalisée à l'aide d'un

bootloader USB appelé "USB HID bootloader". Un bootloader est un logiciel permettant de charger et d'exécuter du code sur le microcontrôleur. Le bootloader USB HID permet de programmer le PIC 18F4550 via une connexion USB, facilitant ainsi le processus de mise à jour du firmware ou du programme embarqué.

En résumé, une carte de développement basée sur le PIC 18F4550 est une plateforme pour développer des applications avec ce microcontrôleur, et son bootloader USB HID facilite la programmation via une connexion USB.

### II.3.1. Connexion du PIC sur la carte :

La figure suivante indique la connexion du pic sur la carte :

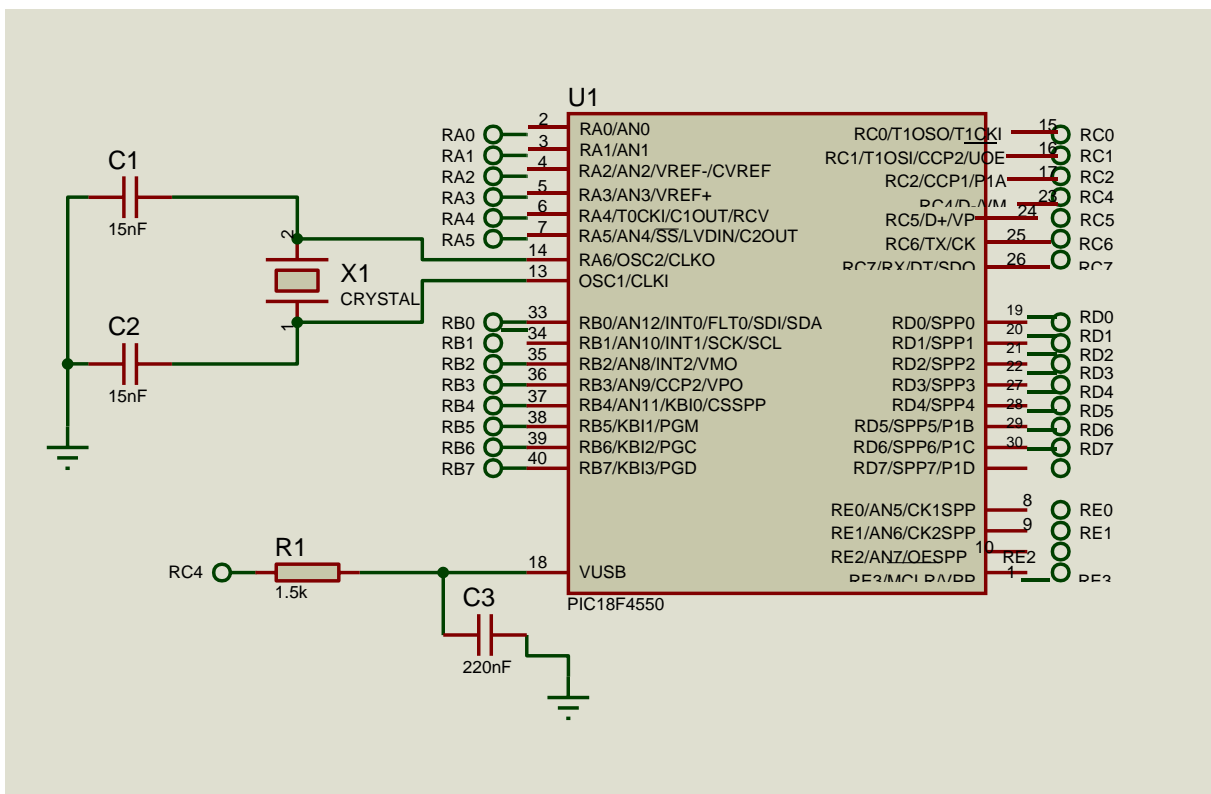


Figure II.1 : connexion du pic sur la carte.

### II.3.2. USB HID Bootloader :

Il offre la possibilité de charger un programme utilisateur dans le microcontrôleur en utilisant une connexion USB directe, sans nécessiter l'utilisation d'un module de programmation supplémentaire. Il se connecte comme le montre la figure ci-dessous :

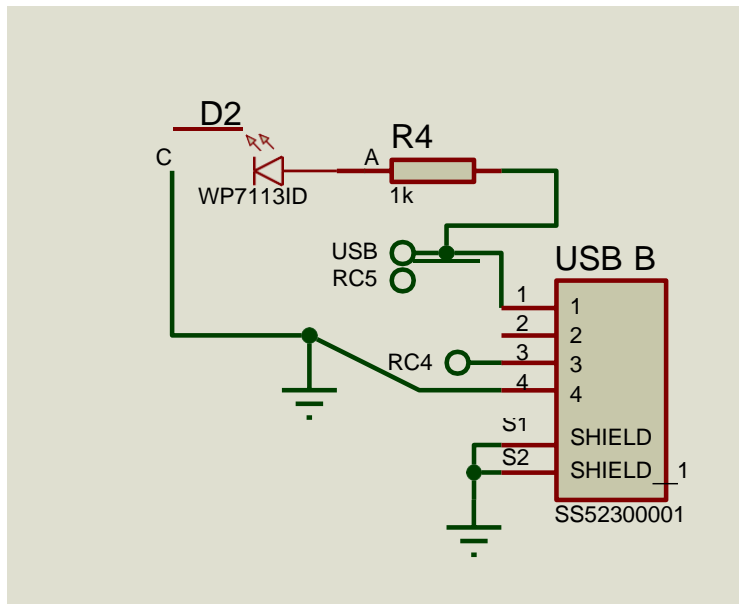


Figure II.2 : USB HID Bootloader sur Proteus.

### II.3.3. Connexion du RESET :

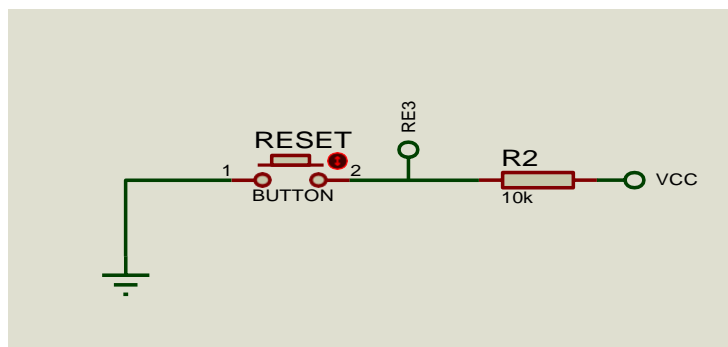


Figure II.3 : bouton RESET sur Proteus.

**II.3.4. ICSP (In-Circuit Serial Programming) :** C'est une méthode de programmation synchronisée définie par Microchip, utilisée pour programmer les microcontrôleurs PIC. Elle nécessite l'utilisation de cinq ou six broches du microcontrôleur pour écrire et lire le firmware. La sixième broche est réservée aux microcontrôleurs prenant également en charge une autre méthode de programmation appelée Low-Voltage ICSP (ICSP basse tension). [5]

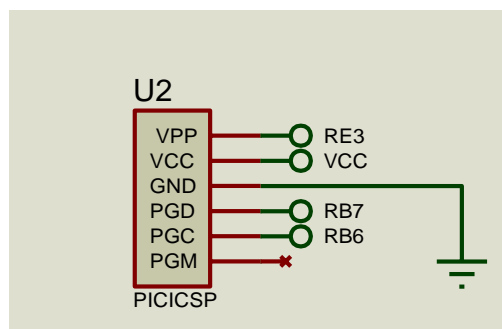


Figure II.4 : connexion du ICSP sur Proteus.



La figure ci-dessous montre le schéma de la carte de développement :

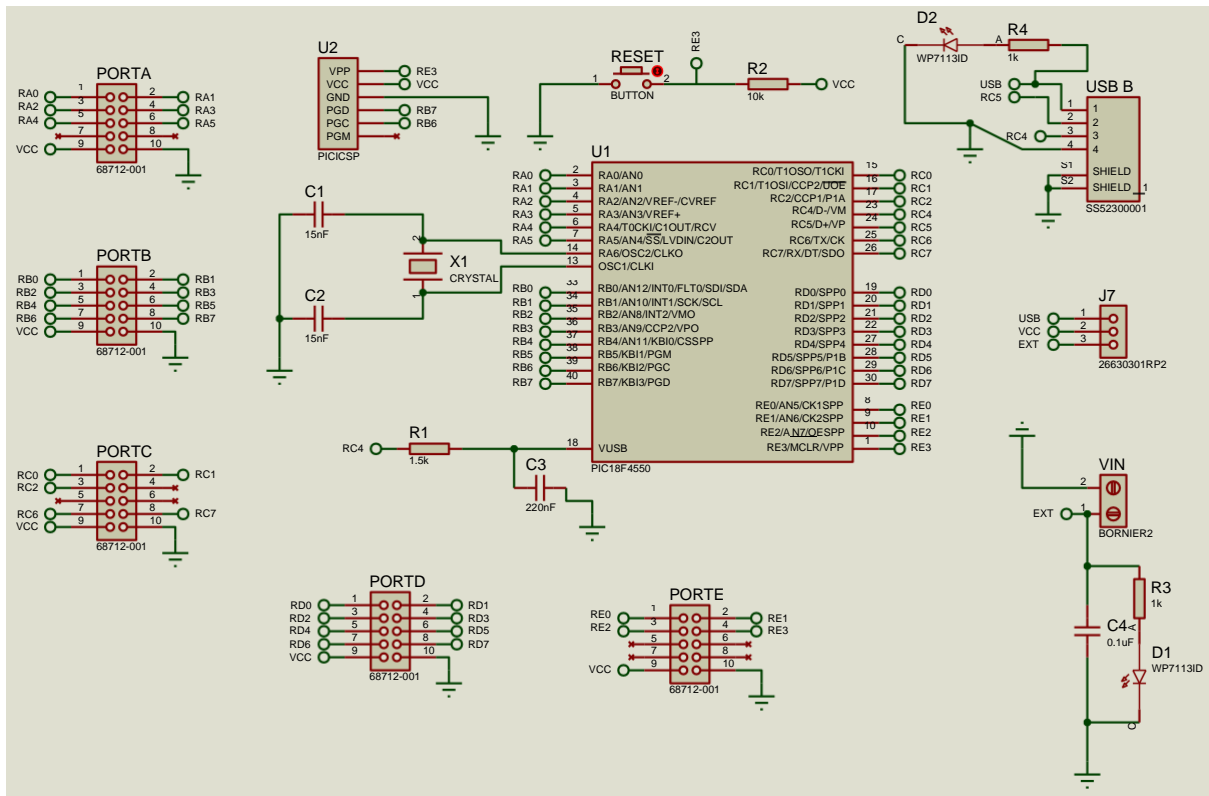


Figure II.5 : Carte de développement à base de PIC 18F4550 sur Proteus.

La carte de développement PIC 18F USB vous aidera à répondre à vos besoins de prototypage avec n'importe quel microcontrôleur Microchip de 27/40 broches prenant en charge une interface USB.

- Tous les ports se terminent par un connecteur Box distinct avec une option d'alimentation en 5 VCC
- Connecteur ICSP pour la programmation des PIC avec support ICD
- Source de Crystal 20 Mhz
- Bornier pour une alimentation extérieure

## II.4. Module d'affichage :

Ce module aide à réaliser des applications qui nécessitent l'affichage des informations. On a mis un afficheur LCD et un 7 Segments utilisable selon les besoins de l'application à développer et on a mis un connecteur pour un afficheur GLCD en cas de besoin.

### II.4.1. Afficheur LCD :

#### II.4.1.1. Présentation :

Un afficheur à cristaux liquides, également connu sous le nom d'afficheur LCD, est composé de deux lames de verre espacées d'environ 20 µm. Les caractères sont dessinés sur ces lames, formant

ainsi les motifs à afficher. L'afficheur fonctionne en appliquant une tension alternative de basse fréquence (quelques volts, généralement entre 3 et 5V) entre les deux faces, ce qui lui permet de devenir opaque. Cependant, pour une bonne visibilité, un afficheur à cristaux liquides nécessite un bon éclairage ambiant. Sa lisibilité s'améliore avec un éclairage adéquat. [1]

#### II.4.1.2. Principe de fonctionnement :

Dans notre projet, nous allons utiliser le mode 4 bits de l'afficheur LCD. Dans ce mode, seuls les 4 bits de poids fort (D4 à D7) de l'afficheur sont utilisés pour transmettre et lire les données. Les 4 bits de poids faible (D0 à D3) sont reliés à la masse, ce qui signifie qu'il nous faut sept fils supplémentaires pour contrôler l'afficheur, en plus de l'alimentation. Les données sont écrites séquentiellement, d'abord les quatre bits de poids fort, suivis des quatre bits de poids faible. Pour valider chaque demi-octet, une impulsion positive d'au moins 450 ns doit être envoyée sur la ligne E. Après chaque action sur l'afficheur, il est possible de vérifier s'il est prêt à traiter la prochaine information. Pour cela, il faut demander une lecture en mode commande et vérifier l'état du flag Busy BT. Lorsque BF=0, cela signifie que l'afficheur est prêt à recevoir une nouvelle commande ou donnée. [1]

#### II.4.1.3. Les différentes broches d'un LCD :

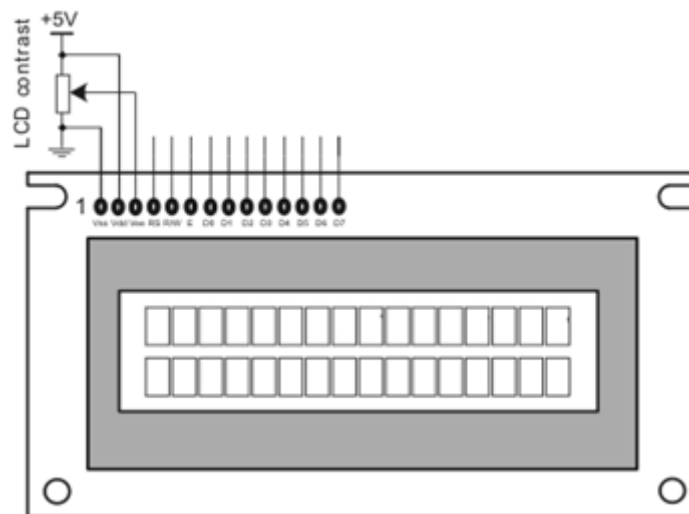


Figure II.6 : schéma d'un LCD.

Au-dessus de l'écran à cristaux liquides lui-même, on trouve une série de 14 broches ayant les rôles suivants :

- Broche 1 : La masse ;
- Broche 2 : VCC ;
- Broche 3 : luminosité ;
- Broche 5, R/W : lecture/écriture ;

Lecture	Ecriture
1	0

- Broche 6,E :commande des opérations d’écriture ou lecture ;
- Broches 7 à 14 : Utilisées pour le transfert des données ou des instructions. Ce transfert peut être effectué sur 8 bits, où toutes les broches sont utilisées, ou sur 4 bits, dans ce cas, seules les broches 11 à 14 sont utilisées.

#### II.4.1.4. Schéma de l’afficheur LCD sur la carte :

Ce module LCD offre une simple solution pour l’interfaçage de LCD avec le microcontrôleur à l’aide du connecteur « BERG » a 10 broches

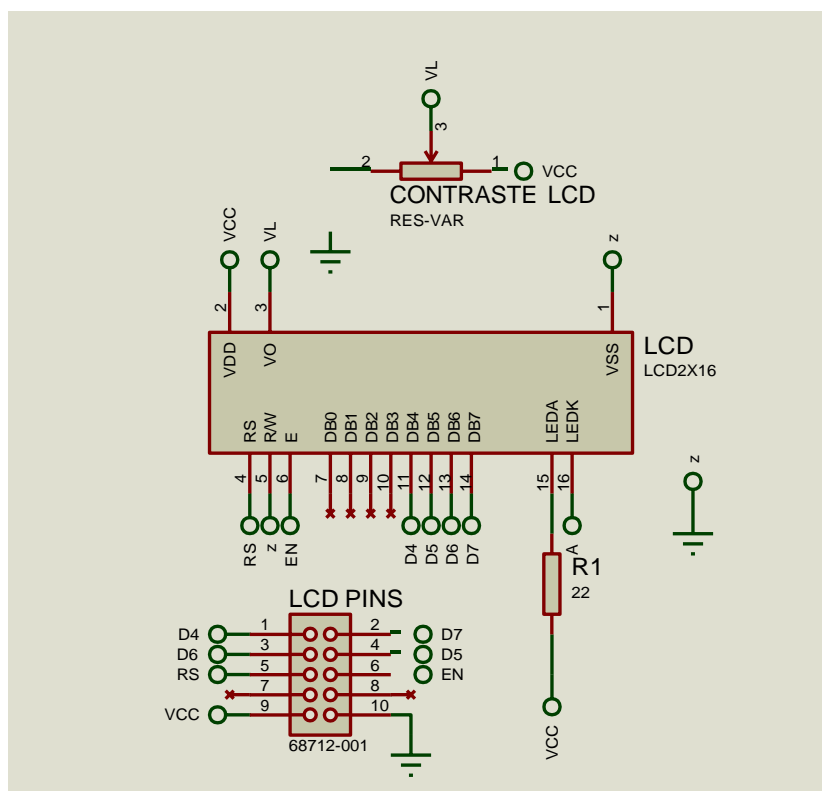


Figure II.7 : schéma de l’afficheur LCD dans le module d’affichage sur Proteus.

#### II.4.2. Afficheurs 7 Segments 4 digits :

Un afficheur 7 segments 4 digits est un type de dispositif d’affichage électronique qui se compose de quatre afficheurs 7 segments. Il permet d’afficher des chiffres décimaux de 0 à 9 en utilisant sept segments lumineux pour chaque afficheur. Ces afficheurs sont souvent utilisés dans divers appareils électroniques tels que les horloges, les compteurs et les panneaux de contrôle pour afficher des nombres ou des informations à quatre chiffres. [6]

Voici son schéma sur la carte :

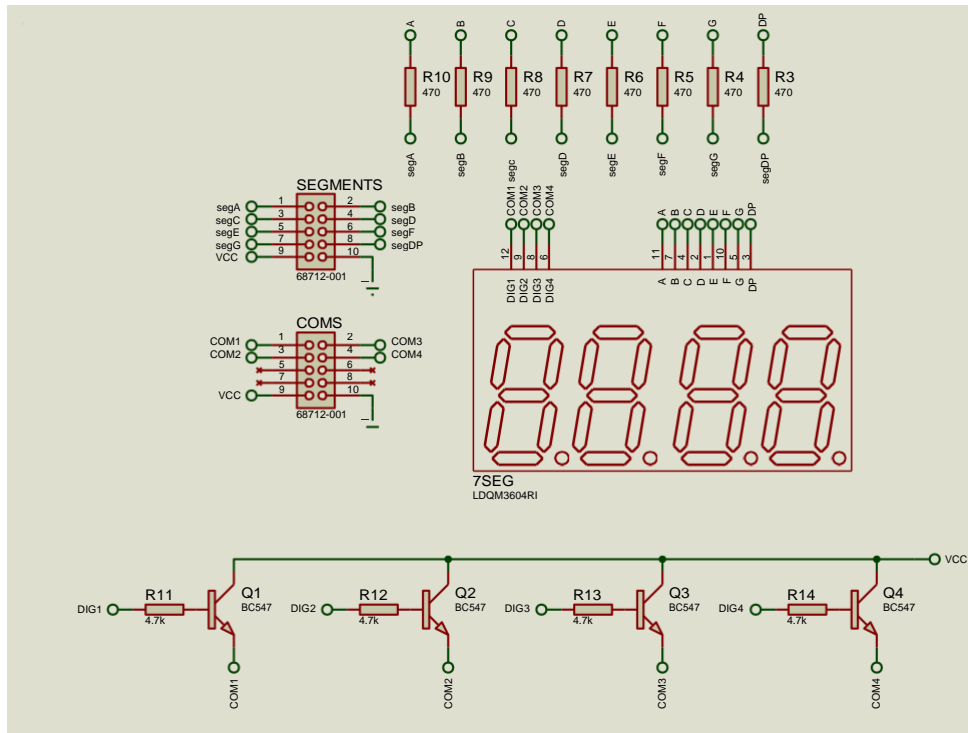


Figure II.8 : Schéma de l’afficheur 7 segments 4 digits sur le module d’affichage sur Proteus.

### II.4.3. Connecteur GLCD :

Le connecteur GLCD offre une simple solution pour l’interfaçage de l’écran GLCD en cas de besoin et voici son schéma sur la carte :

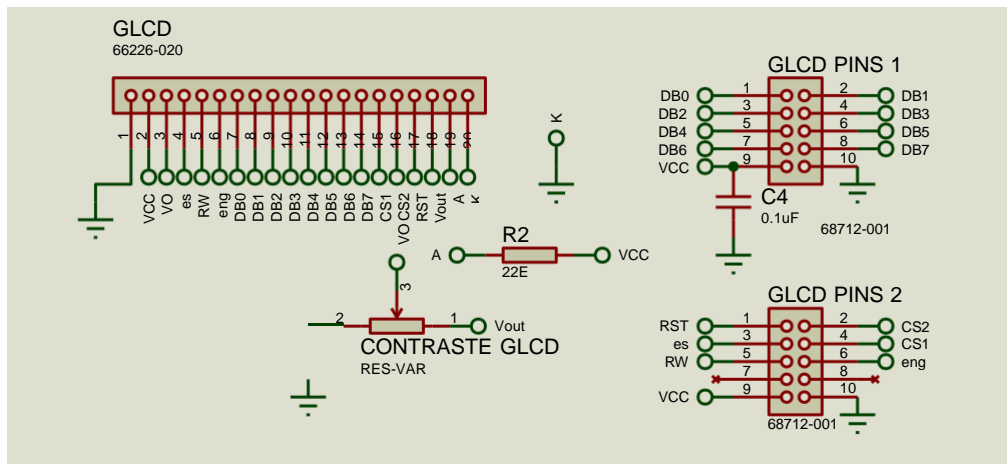


Figure II.9 : Connecteur GLCD sur Proteus.

Le schémas ci-dessous représente le module d’affichage de notre projet :



Lorsqu'une touche est enfoncée sur le clavier, cela provoque la mise à l'état logique 1 du signal D\_avai, tandis que le code ASCII correspondant est présent sur les lignes de sortie du codeur (A... D).

Le clavier est composé de 16 touches disposées selon une matrice de 4 lignes et 4 colonnes. Chaque ligne représente un groupe de 4 touches, tandis que les 4 colonnes correspondent aux pistes verticales. Lorsqu'une touche est enfoncée, elle établit un contact électrique entre une piste horizontale et une piste verticale.

Chaque touche agit comme un interrupteur normalement ouvert. Lorsqu'une touche est enfoncée, un signal électrique est envoyé vers le codeur, qui est un circuit électronique simple. Ce codeur associe à chaque signal un code, tel que le code ASCII de la touche pressée. Ce signal peut être utilisé pour générer une interruption vers le processeur, afin qu'il puisse traiter l'information reçue. Les codeurs utilisés couramment offrent également des fonctionnalités supplémentaires, comme la capacité de détecter les touches maintenues enfoncées pendant une période prolongée.

### II.5.1.2. Schémas du clavier sur la carte :

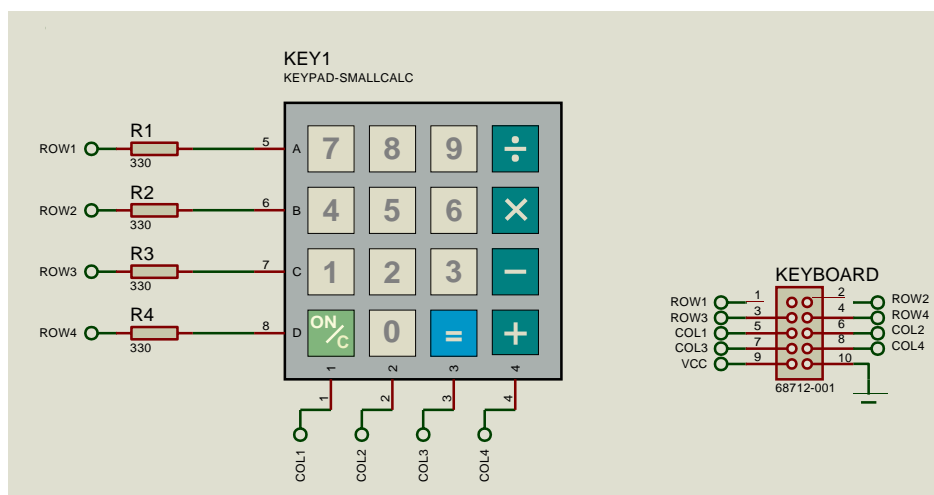


Figure II.12 : Schéma du clavier sur Proteus.

Ce schéma facilite l'interfaçage du clavier avec le microcontrôleur à l'aide du connecteur à 10 pins.

### II.5.2. Matrice des boutons :

Les boutons sont des dispositifs simples à utiliser, abordables et faciles à mettre en œuvre. Ils peuvent être de type normalement ouvert ou normalement fermé. Cependant, lors du changement d'état, les boutons peuvent être sujets à des rebonds ou du bruit.

Pour remédier à ces problèmes de rebondissement et garantir une utilisation fiable des boutons, le compilateur MikroC met à notre disposition une bibliothèque appelée "Button". Cette bibliothèque contient une fonction unique : "unsigned short Button (unsigned short \*port, unsigned short pin, unsigned short time, unsigned short active\_state)". Les paramètres de cette fonction sont les suivants

: "port" représente le port auquel le bouton est connecté, "pin" indique le bit du port correspondant au bouton, "time" est le délai en millisecondes pour filtrer le bruit, et "active\_state" est l'état logique pour lequel nous souhaitons activer le bouton. La fonction renvoie la valeur 0 si le bouton n'est pas actif et 255 s'il est actif.

L'installation des boutons peut être réalisée de deux manières : en étant actif à l'état haut (lorsque le bouton est enfoncé) ou actif à l'état bas (lorsque le bouton est relâché). [7]



Figure II.13 : Schéma d'installation des boutons : (a) actif à l'état bas ; (b) actif à l'état haut.

La matrice à 8 boutons offre une connexion facile d'un connecteur de type boîtier à 10 broches à 8 interrupteurs tactiles de petite taille, avec une sélection possible via un jumper d'un réseau de résistances d'état haut ou d'état bas connecté en parallèle.

Le schéma ci-dessous représente la matrice des boutons sur le module :

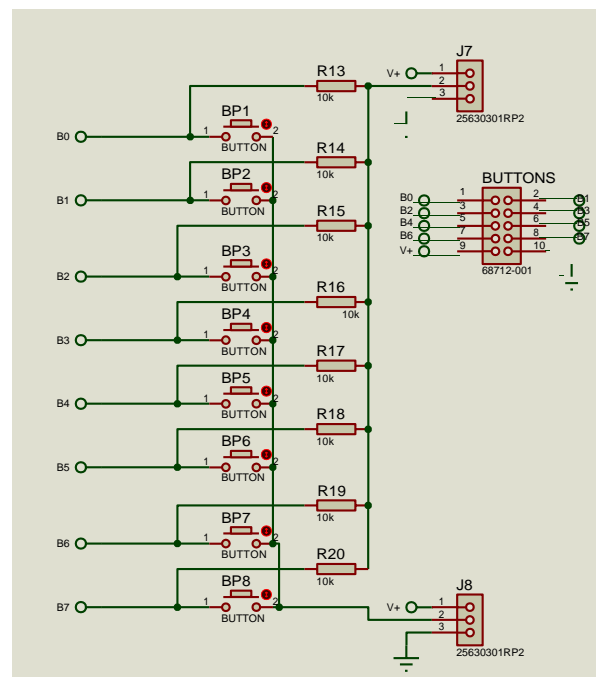


Figure II.14 : matrice des boutons sur Proteus.

### II.5.3. Matrice de LEDs :

Cette simple matrice à 8 LED offre un moyen pratique de surveiller ou d'afficher la sortie de différentes broches d'E/S pour notre projet.

Ci-dessous le schéma de la matrice sur notre module :

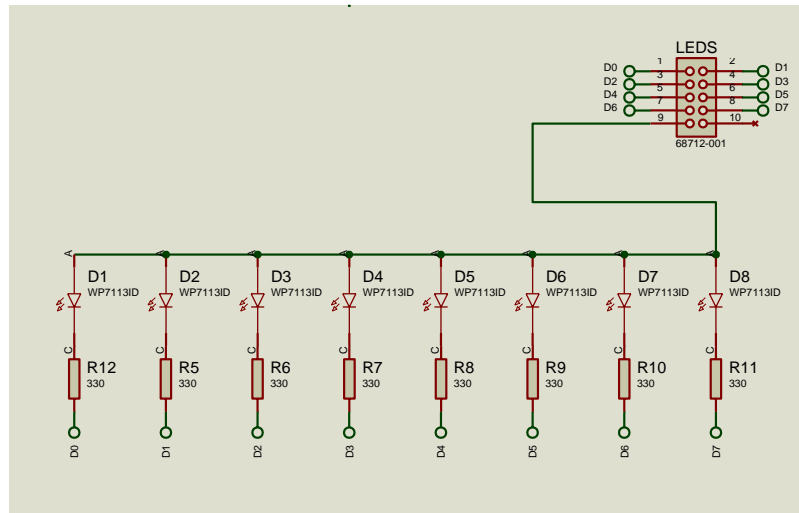


Figure II.15 : matrice de LEDs sur Proteus.

## II.5.4. Les entrées analogiques :

### II.5.4.1. LDR (photo résistance) :

Une photo résistance, également connue sous le nom de LDR (Light Dependent Resistor), est un composant électronique dont la valeur en ohms varie en fonction de l'intensité lumineuse à laquelle il est exposé. Son utilisation principale est la mesure de l'intensité de la lumière. Cependant, elle est souvent concurrencée par la photodiode, qui offre un temps de réponse plus court.

Les photo résistances sont généralement fabriquées à partir de matériaux tels que le sulfure ou le sélénium de cadmium, qui présentent des propriétés de semi-conducteurs. [1]

Voici le schéma de LDR :

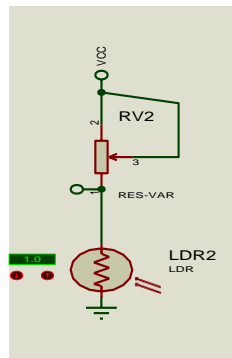


Figure II.16 : LDR sur Proteus.

### II.5.4.2. Potentiomètres :

Les potentiomètres sont des dispositifs électriques ajustables qui permettent de contrôler la résistance électrique. Ils sont couramment utilisés pour régler des paramètres variables tels que le niveau sonore, la luminosité, la vitesse, et bien d'autres, dans les circuits électroniques. Les potentiomètres se composent d'une résistance variable et d'un curseur mobile qui permet de







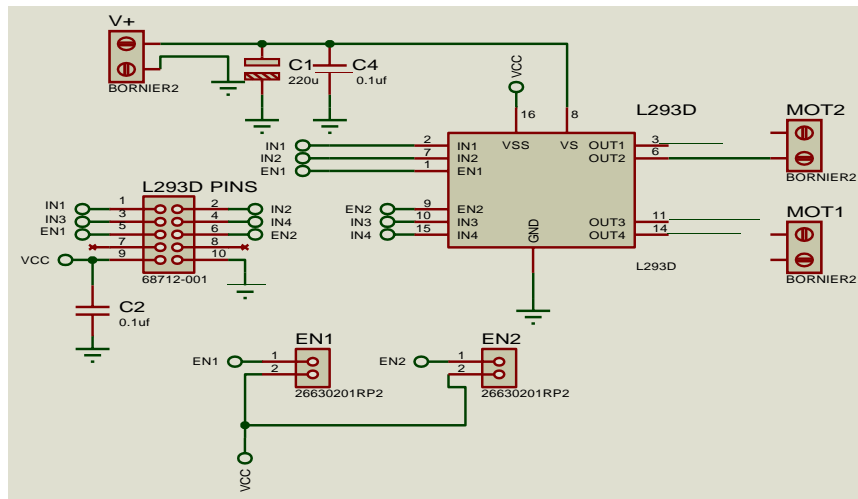


Figure II.21 : Schéma du pilote L293D sur Proteus.

On a mis 1 Bornier pour l'alimentation de 5 à 36 volts et 2 autres pour la connexion des moteurs à courant continu. On a mis 2 jumper (EN1, EN2) pour choisir le canal 1 ou 2.

Le connecteur a 10 pins et pour l'interfaçage avec la carte de développement.

Le tableau suivant donne le brochage de L293D :

N°	Broches	Signification
1	Enable 1,2	Broche d'activation du canal 1 active à l'état haut (5V)
2	Input 1(ou IN1)	Entrée canal 1
3	Output 1	Sortie canal 1
4	GND	Masse
5	GND	Masse
6	Output 2	Sortie canal 1
7	Input 2(ou IN2)	Entrée canal 1
8	Vcc2	Tension d'alimentation de sortie (4,5V ;36V)
9	Enable 3,4	Broche d'activation du canal 2 active à l'état haut (5V)
10	Input 3(ou IN3)	Entrée canal 2
11	Output 3	Sortie canal 2
12	GND	Masse
13	GND	Masse

14	Output 4	Sortie canal 2
15	Input 4(ou IN4)	Entrée canal 2
16	Vcc1	Tension d'alimentation d'entrée (4,5V ;5V)

Tableau II.1 : Brochage de l'interface moteur L293D. [11]

### II.6.2. ULN2003A Contrôleur de moteur pas à pas :

Le ULN2003A est un ensemble de transistors Darlington à haute tension et haute intensité, comprenant chacun sept paires d'émetteurs communs à collecteur ouvert. Chaque paire est capable de supporter un courant de 500mA. Des diodes de suppression sont incluses pour la commande de charges inductives, les entrées et les sorties sont disposées en opposition pour simplifier la conception de la carte. Ces dispositifs sont capables de piloter une large gamme de charges, notamment des solénoïdes, des relais, des moteurs à courant continu, des affichages LED, des lampes à filament, des têtes d'impression thermiques et des tampons de puissance élevée. [12]



Figure II.22 : ULN2003A.

Le schéma ci-dessous représente le circuit électrique de l'ULN2003A :

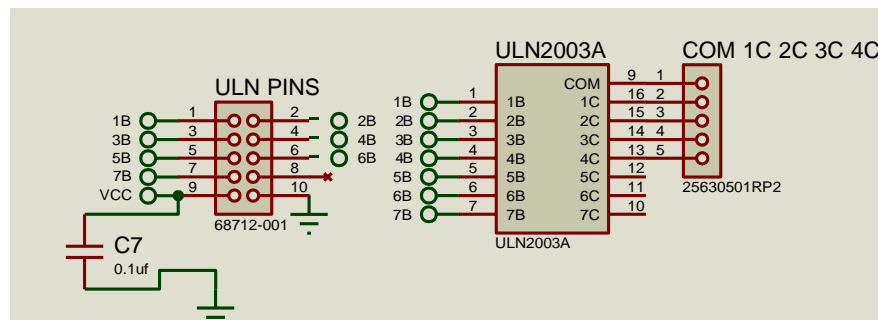


Figure II.23 : ULN2003A sur Proteus.

La schématisation de l'uln2003 est simple, on a utilisé un boîtier connecteur à 10 pins pour l'interfaçage avec le microcontrôleur et un connecteur à 5 pin pour la connexion d'un moteur pas à pas.

### II.6.3. Adaptateur de Servomoteurs :

Le servomoteur fait partie de la famille des moteurs électriques et se distingue principalement par sa capacité exceptionnelle de régulation et les possibilités de contrôle qui en découlent. Il offre la

possibilité de régler avec précision la position de l'angle, l'accélération et la vitesse. Pour fournir cette caractéristique, l'architecture fondamentale du servomoteur comprend, en plus du moteur lui-même, un capteur de position du rotor compatible. Ce capteur, également connu sous le nom de capteur de position, de rotation ou de rétroaction du moteur, est capable d'enregistrer de manière précise la position de l'arbre moteur à un instant donné. [13]



Figure II.24 : Servomoteur.

Le schéma suivant représente l'adaptateur de 2 servomoteurs avec le PIC :

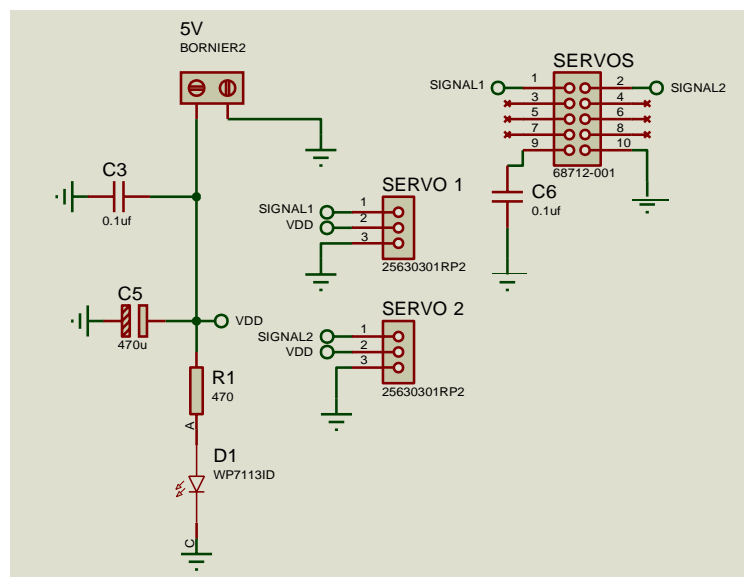


Figure II.25 : adaptateur de 2 servomoteurs sur Proteus.

Cette adaptateur offre une connexion/interfaçage de servomoteurs avec le PIC grâce au connecteur à 10 pins

On a utilisé 2 connecteur à 3 pins pour connecter 2 servomoteurs, et un bornier pour les alimenter avec une tension de 5V.

#### II.6.4. Relais à 2 canaux :

Un relais électromécanique est un composant électrique qui permet de distribuer la puissance électrique en fonction d'un signal provenant d'une partie de commande. Il offre la possibilité d'ouvrir et de fermer un circuit électrique de puissance en se basant sur une information logique. Ce dispositif assure une isolation complète (appelée isolation galvanique) entre les deux circuits, à savoir le circuit

de puissance et le circuit de commande, ce qui signifie qu'ils peuvent avoir des caractéristiques d'alimentation électrique différentes. [14]



Figure II.26 : Relais.

Le schéma ci-dessous représente le module de relais à 2 canaux :

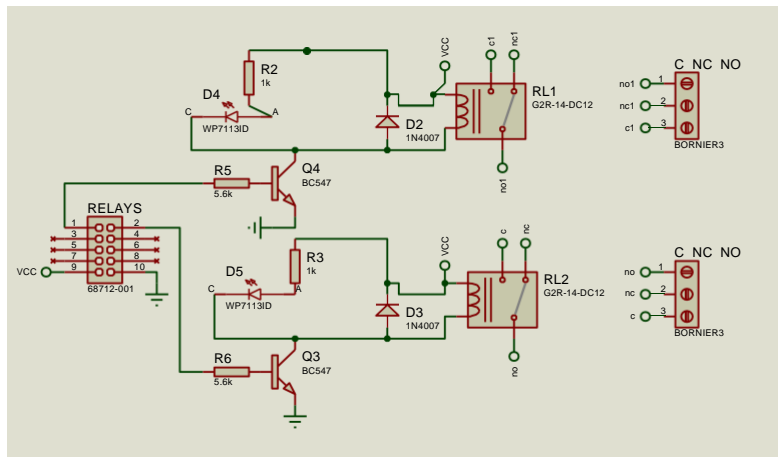


Figure II.27 : module a relais sur Proteus.

Le schéma suivant représente le module commande moteurs :

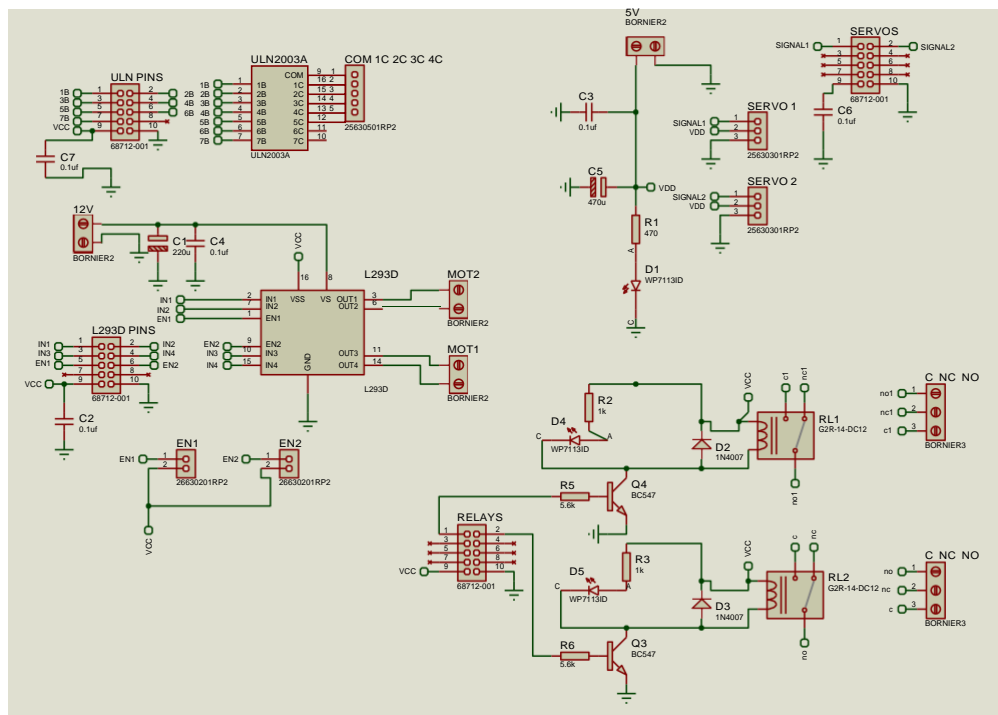


Figure II.28 : Module commande de moteurs sur Proteus.

## II.7. Module d'alimentation (5V et 12V) :

On veut réaliser un module d'alimentation stabilisée de 5V et 12V et débiter un courant de 600 mA, pour cela on suit le montage proposé ci- dessous :

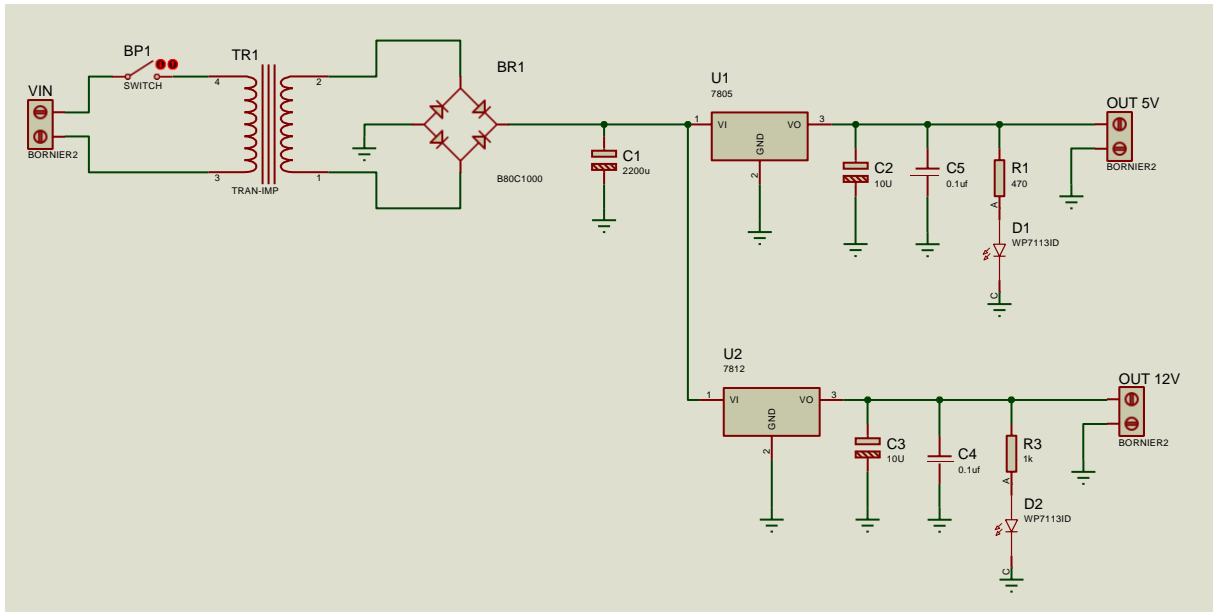


Figure II.29 : schéma de l'alimentation stabilisée sur Proteus.

Ce montage est composé de :

- Transformateur
- Interrupteur
- Un pont de diodes
- 2 régulateurs
- 5 capacités
- 2 Diode LED
- 2 résistances
- 3 borniers

## II.8. Conclusion :

En conclusion, ce chapitre a présenté de manière détaillée la solution sélectionnée pour répondre aux spécifications du cahier des charges. Nous avons examiné en profondeur la conception de chaque composant du système, ce qui nous a permis d'obtenir une schématisation complète et précise. Cette approche détaillée garantit une compréhension approfondie du fonctionnement du système et sert de base solide pour sa mise en œuvre future.

## Chapitre III:

# Réalisation et Programmation de la maquette électronique numérique

### III.1 Introduction :

Après avoir conçu les Schémas de nos cartes (modules) en utilisant ISIS Proteus, dans ce chapitre on réalise la maquette et on présente les deux côtés : hardware et software

Dans le hardware on présente la réalisation de la maquette, les circuits imprimés ; dans le coté software on présente la programmation de la carte de développement et le chargement de HID bootloader dans notre PIC.

### III.2 Coté hardware :

#### III.2.1 Module de développement :

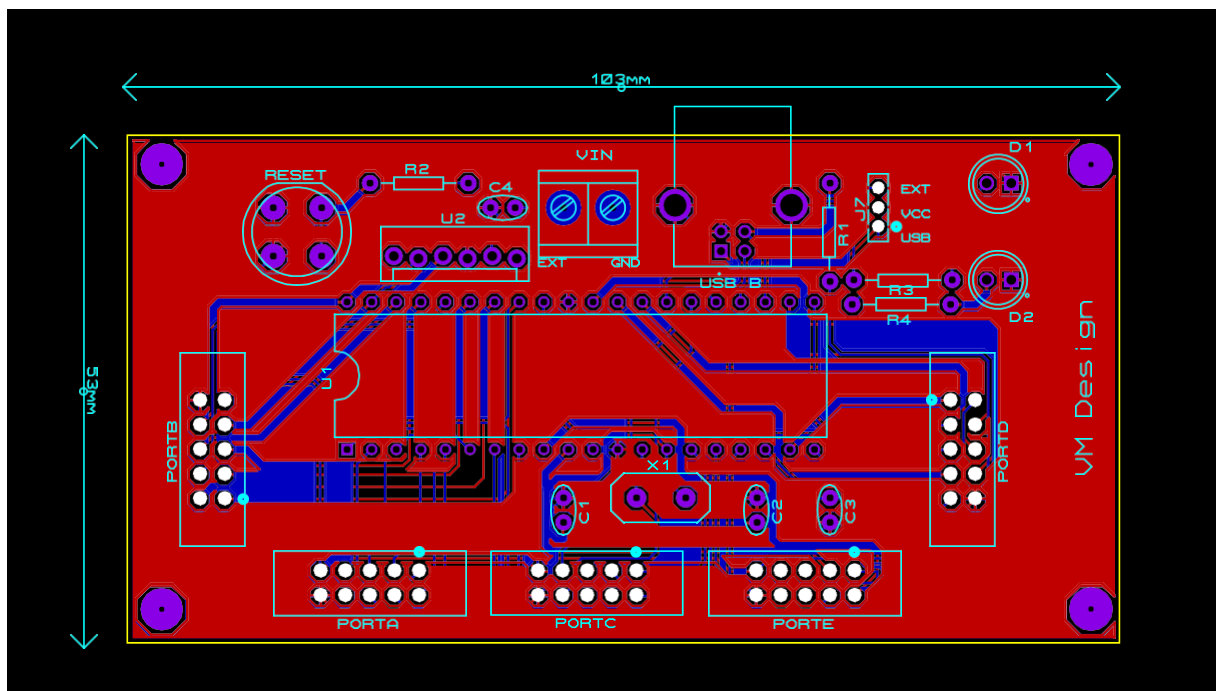


Figure III.1 : Schéma du PCB de la carte développement sur Proteus.

Ce schéma est composé de 2 couches après l'insertion de plan de masse on passe à la réalisation.



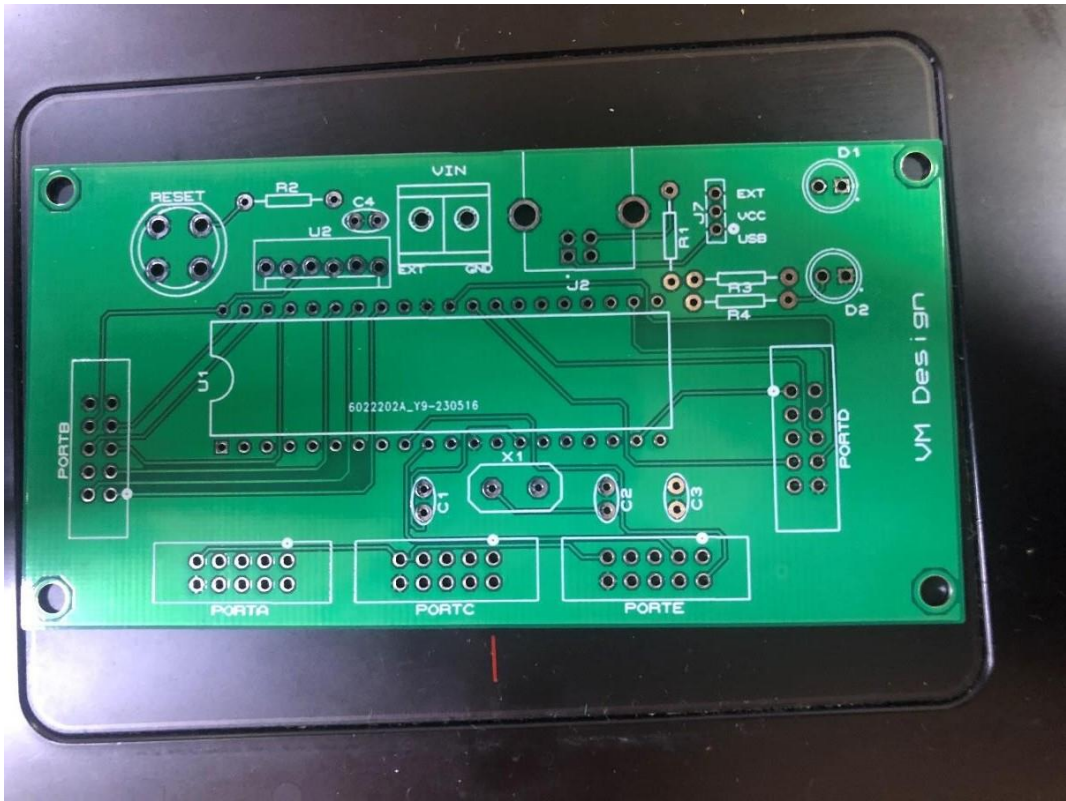


Figure III.2 : Image du circuit imprimé de la carte de développement

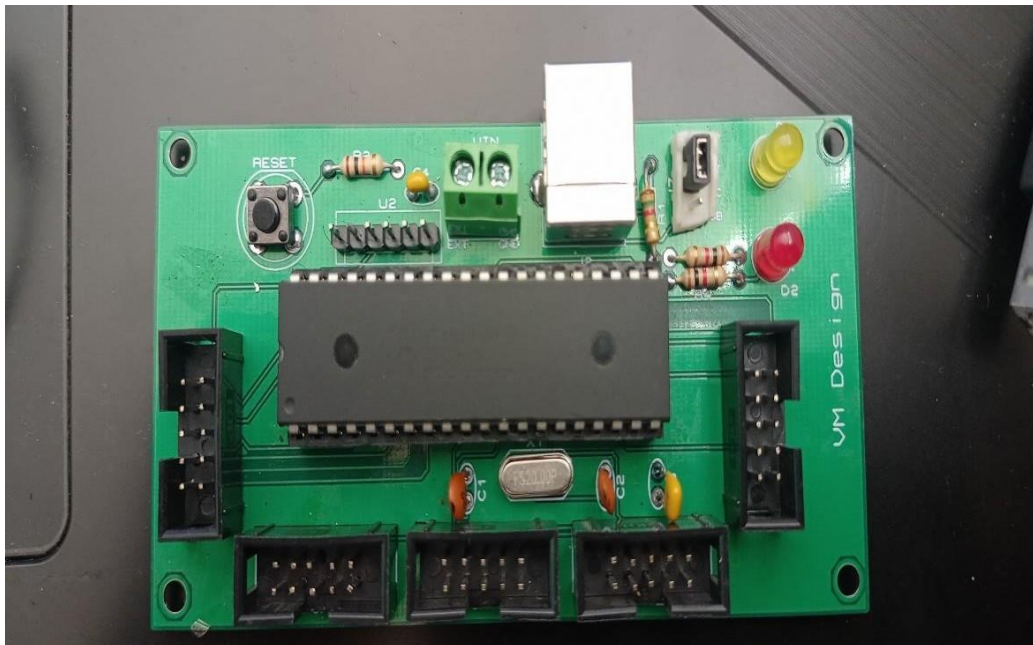


Figure III.3 : Image de la carte de développement.

### III.2.2 Module d'affichage :

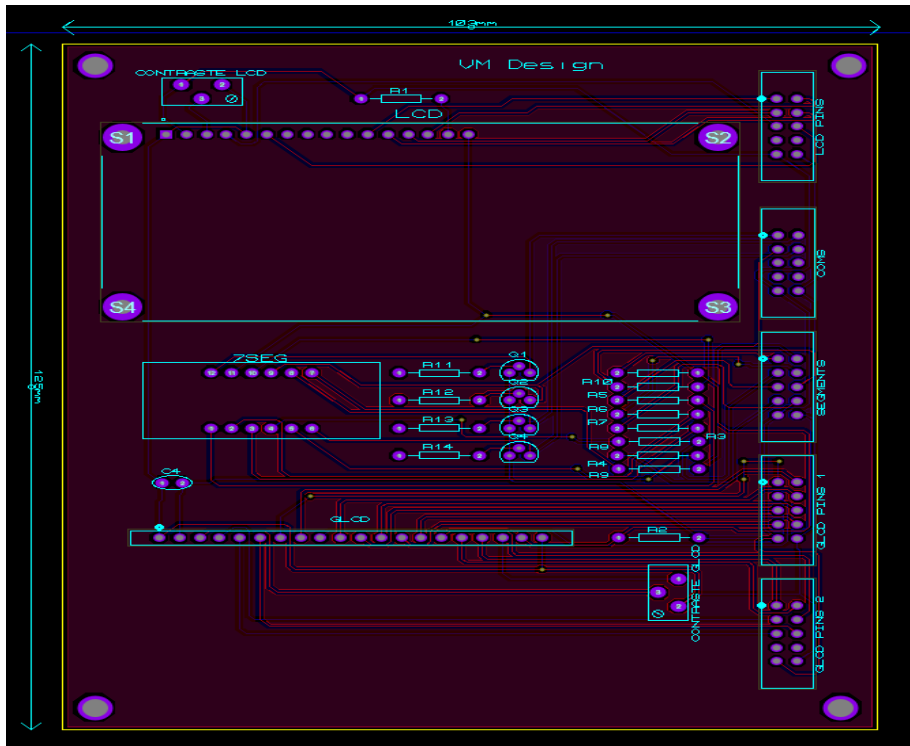


Figure III.4 : Schéma du PCB de la carte d'affichage sur Proteus.

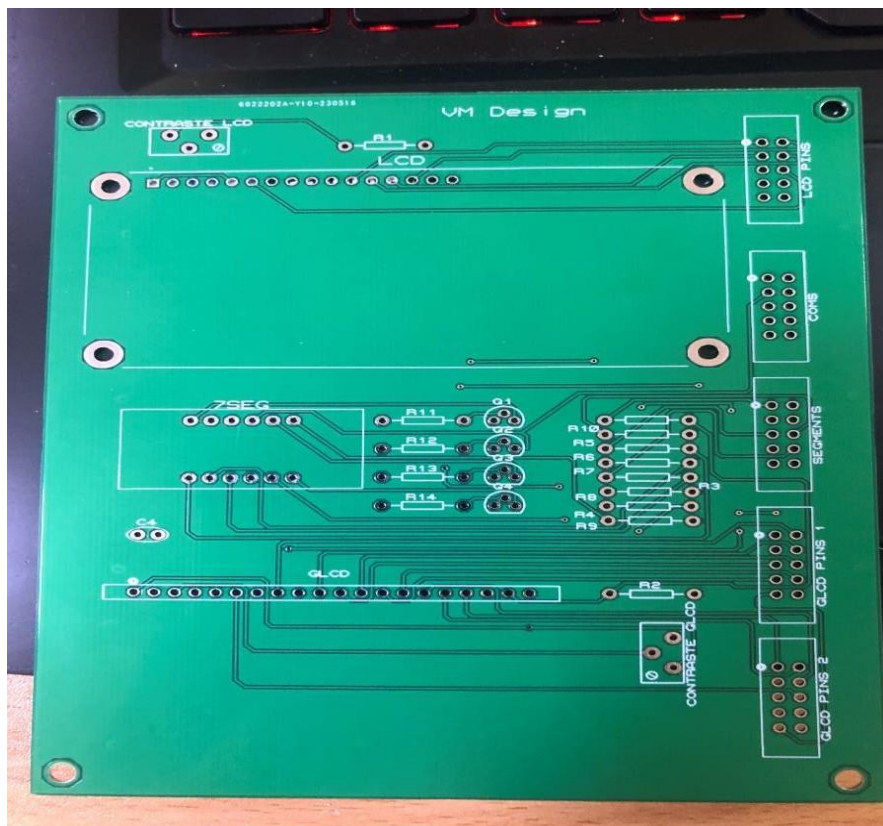


Figure III.5 : Image du circuit imprimé du module d'affichage



Figure III.6 : Image du module d'affichage.

### III.2.3 Module d'entrées/sorties :

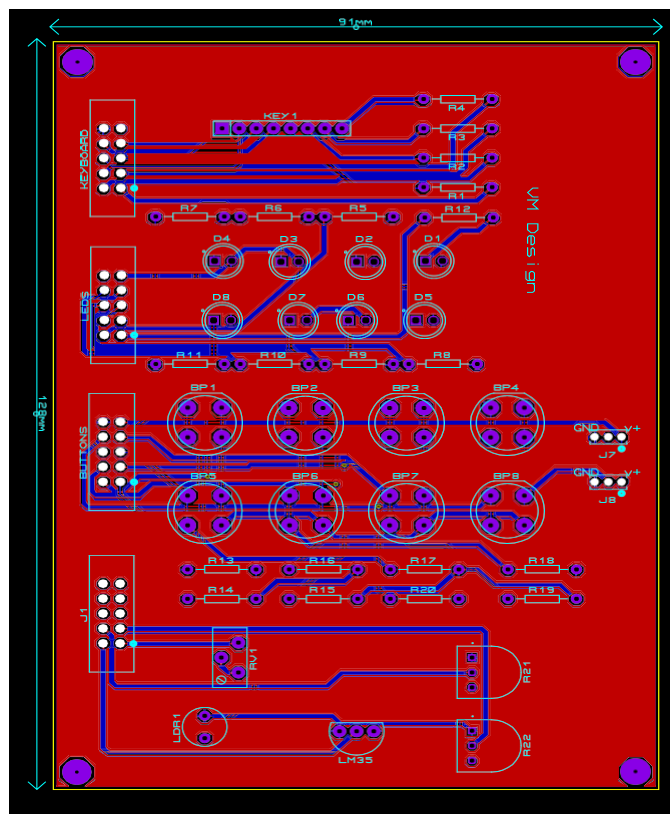


Figure III.7 : schéma PCB de la carte de I/O.

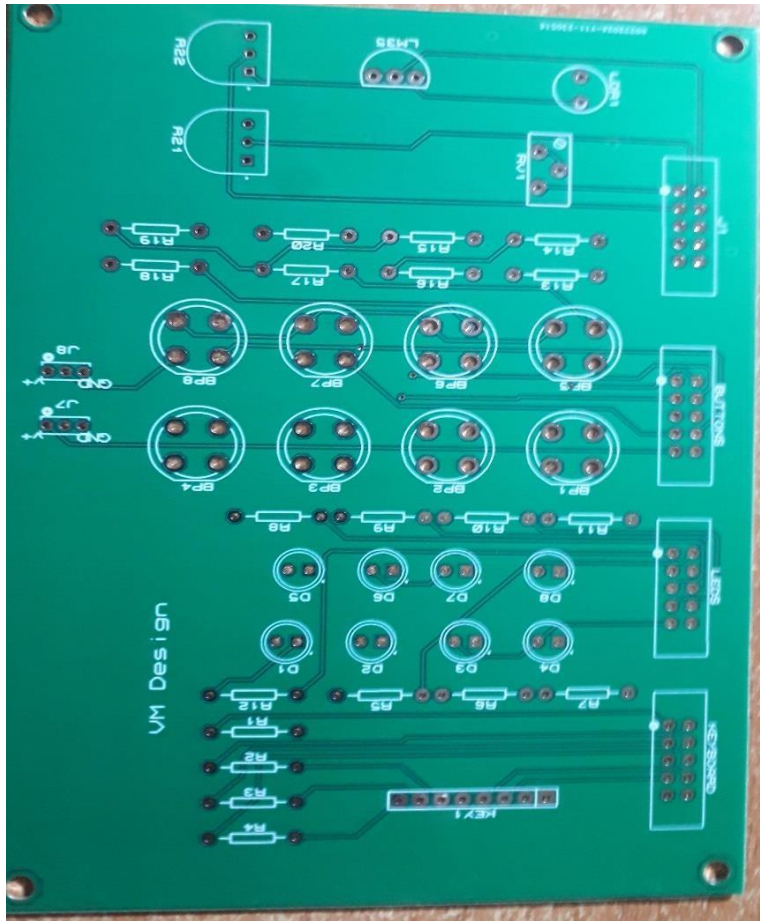


Figure III.8 : Image du circuit imprimé du module de I/O.

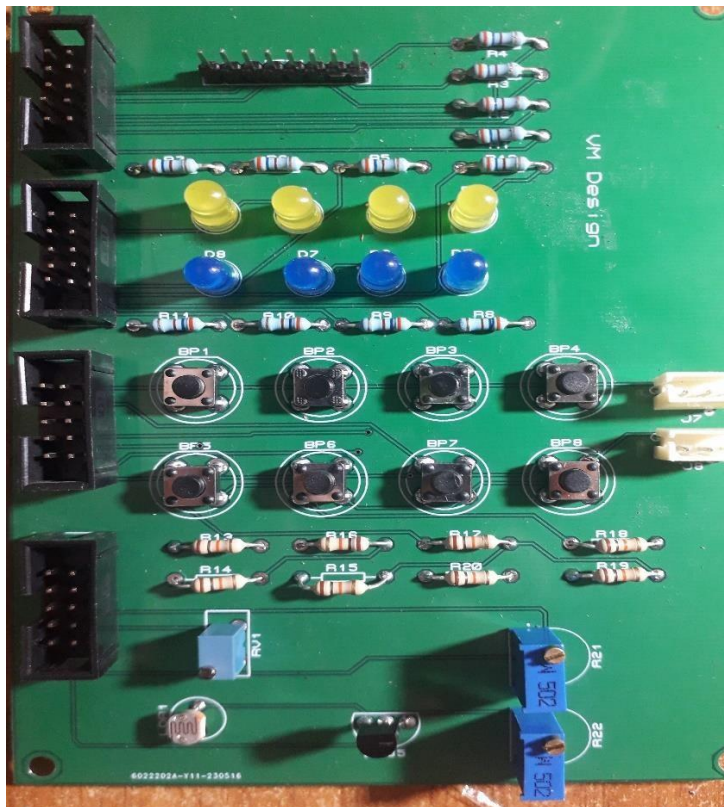


Figure III.9 : image du module de I/O.

### III.2.4 Module de commandes moteurs :

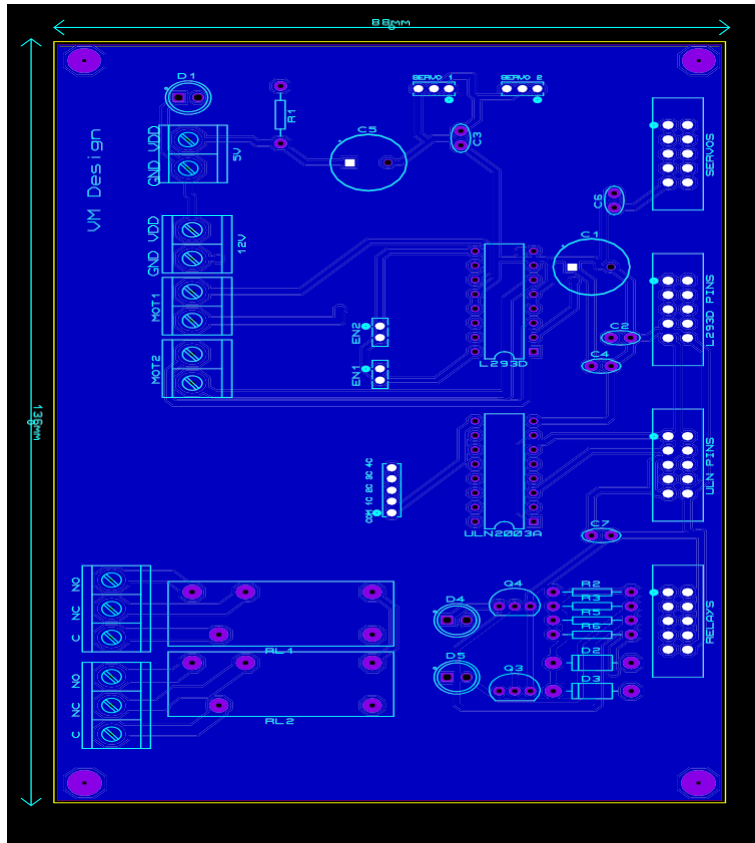


Figure III.10 : schéma PCB de la carte de commande de moteurs.

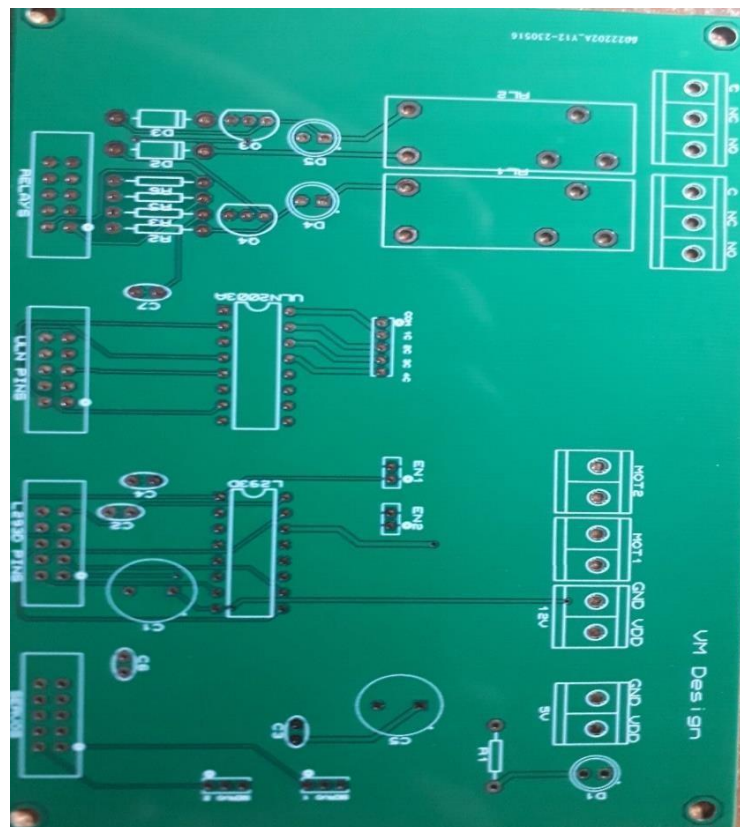


Figure III.11 : Image du circuit imprimé du module de commandes de moteurs.



Figure III.12 : Image de la carte de commandes de moteurs.

### III.2.5 Carte d'alimentation :

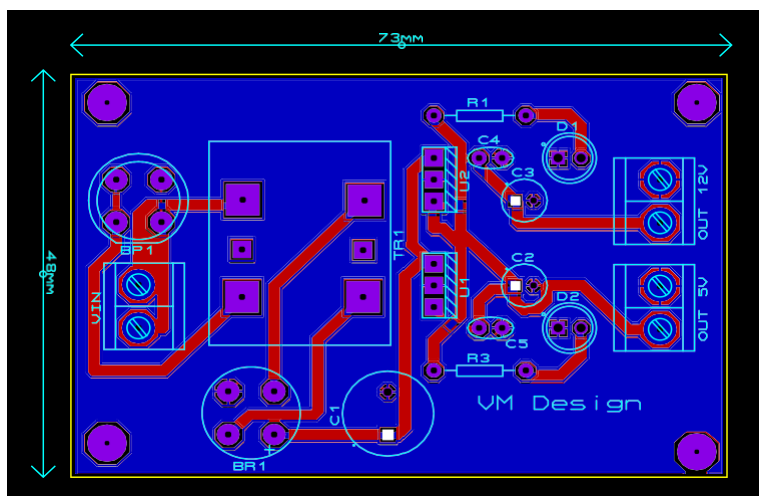


Figure III.13 : schéma PCB de la carte d'alimentation.

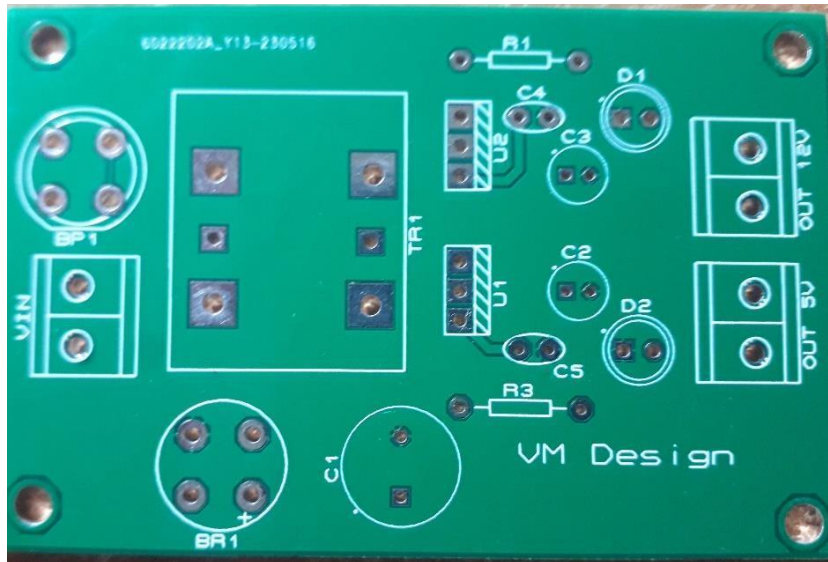


Figure III.14 : circuit imprimé de la carte d'alimentation.

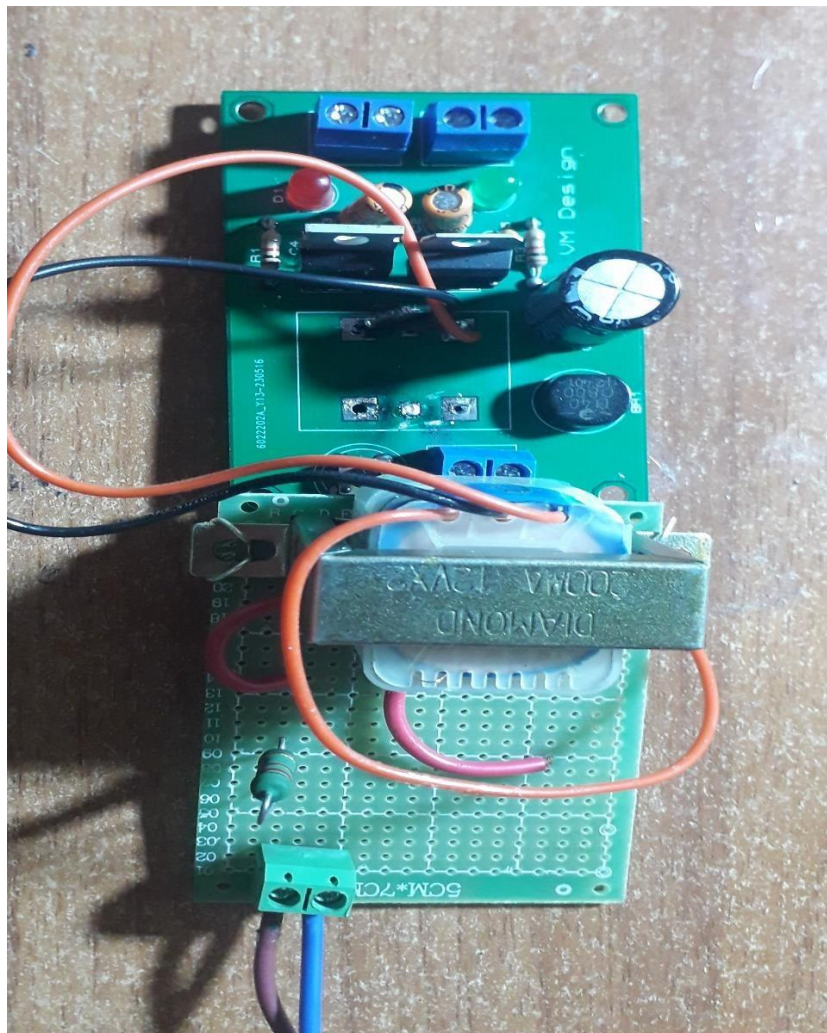


Figure III.15 : Image de la carte d'alimentation.

### III.3 coté software :

Les éléments indispensables pour le Bootloader comprennent :

- 1-Un câble USB pour relier la carte à l'ordinateur.
- 2-Un microcontrôleur préalablement programmé avec le bootloader.

#### III.3.1 Installer le bootloader dans le PIC 18F4550 :

Pour installer le programme du bootloader, il est nécessaire d'utiliser un programmeur externe tel que PicKit3, un programmeur série ou ICD3. C'est une opération unique qui n'a pas besoin d'être répétée. Vous devez flasher le programme du bootloader fourni, nommé "MCHPUSB Bootloader PIC18F4550 Family.hex", qui se trouve dans le dossier dédié au bootloader. [15]

Nous avons utilisé le PicKit 3 clown pour exécuter cette opération et les étapes sont les suivantes :

- 1- Connecter le pic 18F4550 comme le montre la figure au-dessous :

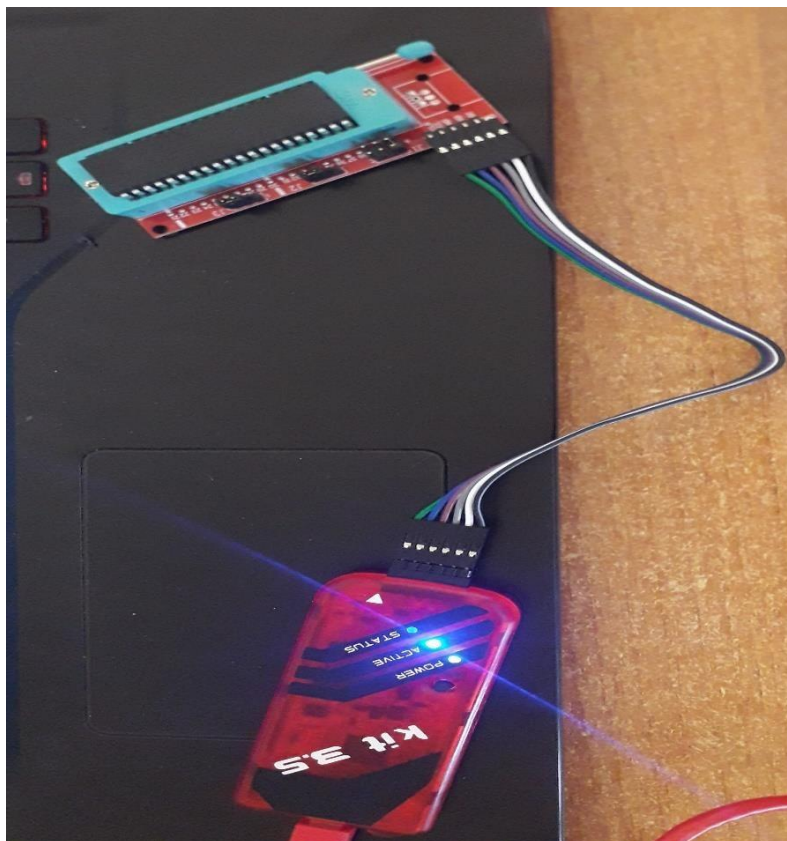


Figure III.16 : connexion du pic sur le Pickit 3 clown.

- 2- Connecter le PicKit 3 au pc
- 3- On ouvre l'application du programmeur



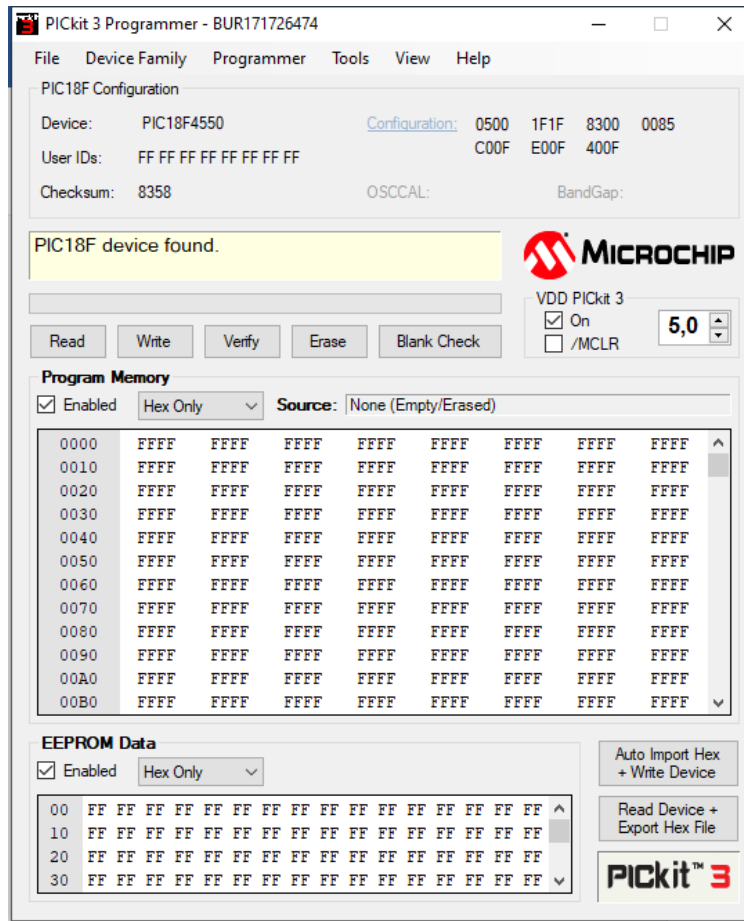


Figure III.17 : application du programmeur.

- 4- Comme vous le voyez dans la figure au-dessus le pic est reconnu maintenant on passe l'importation du fichier bootloader.hex, Pour faire cela on clique sur file puis sur import Hex comme le montre la figure suivante :

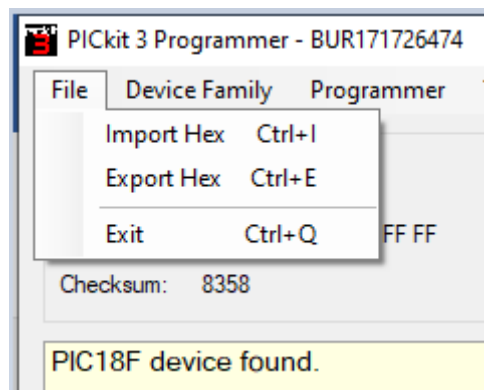


Figure III.18 : importation du fichier Hex.

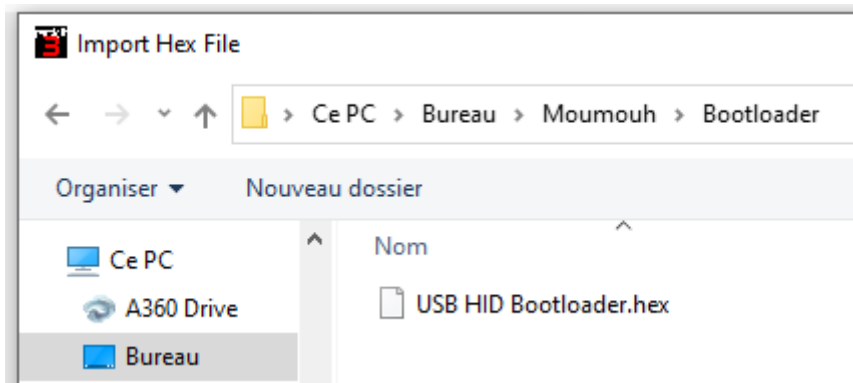


Figure III.19 : le fichier du HID Bootloader.

5- Maintenant on clique sur write et notre programme sera importer avec succès

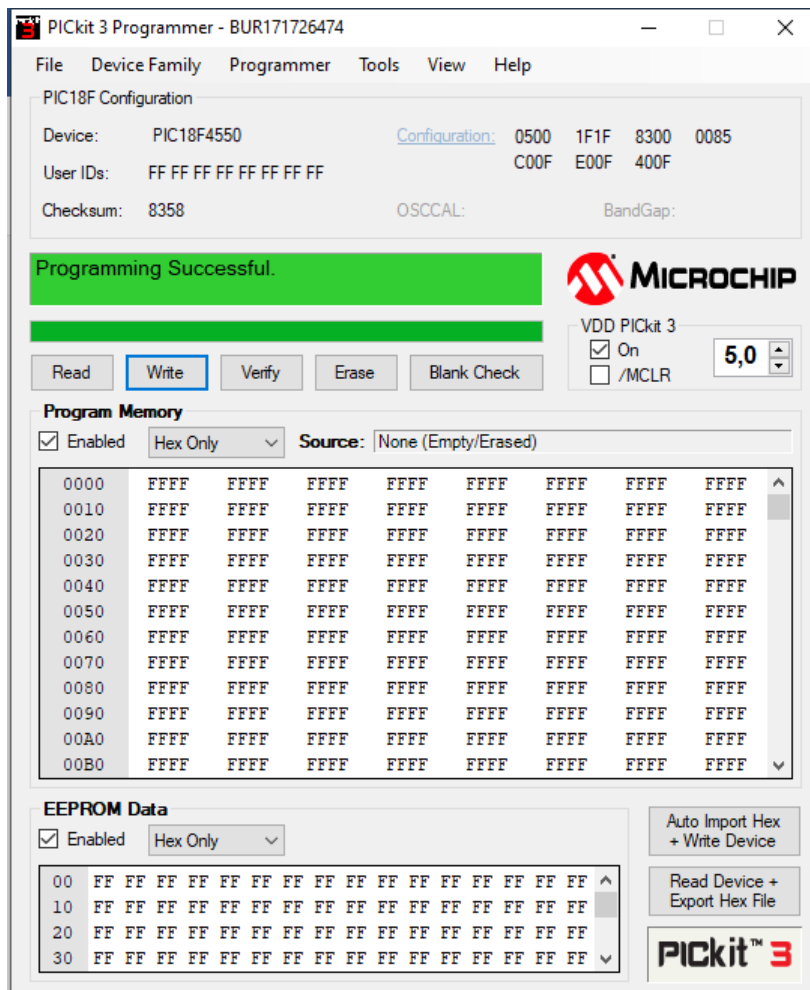


Figure III.20 : La fin de chargement du HID bootloader.

### **III.4 Conclusion :**

En conclusion du chapitre III, nous avons abordé la réalisation et la programmation de la maquette électronique numérique. Tout au long de ce chapitre, nous avons exploré les étapes nécessaires pour concrétiser notre maquette, en mettant l'accent sur la programmation. Maintenant notre carte est prête à l'emploi.

# Conclusion générale

Dans le cadre de notre projet de fin d'études, nous avons eu l'opportunité d'approfondir nos connaissances théoriques et de développer une solide expérience pratique. Notre travail s'est concentré sur la réalisation d'une maquette didactique basée sur un microcontrôleur PIC.

Ce projet nous a permis d'élargir nos compétences dans le domaine de l'électronique, notamment en ce qui concerne :

- La conception et la réalisation de cartes et de circuits imprimés, nous donnant ainsi une compréhension approfondie du processus de développement matériel.
- La création de cartes d'interface, ce qui nous a familiarisés avec les différentes méthodes de connexion entre composants électroniques.
- La programmation des microcontrôleurs PIC, nous permettant de développer des compétences dans la programmation embarquée et de comprendre le fonctionnement interne de ces dispositifs.

## Références bibliographiques

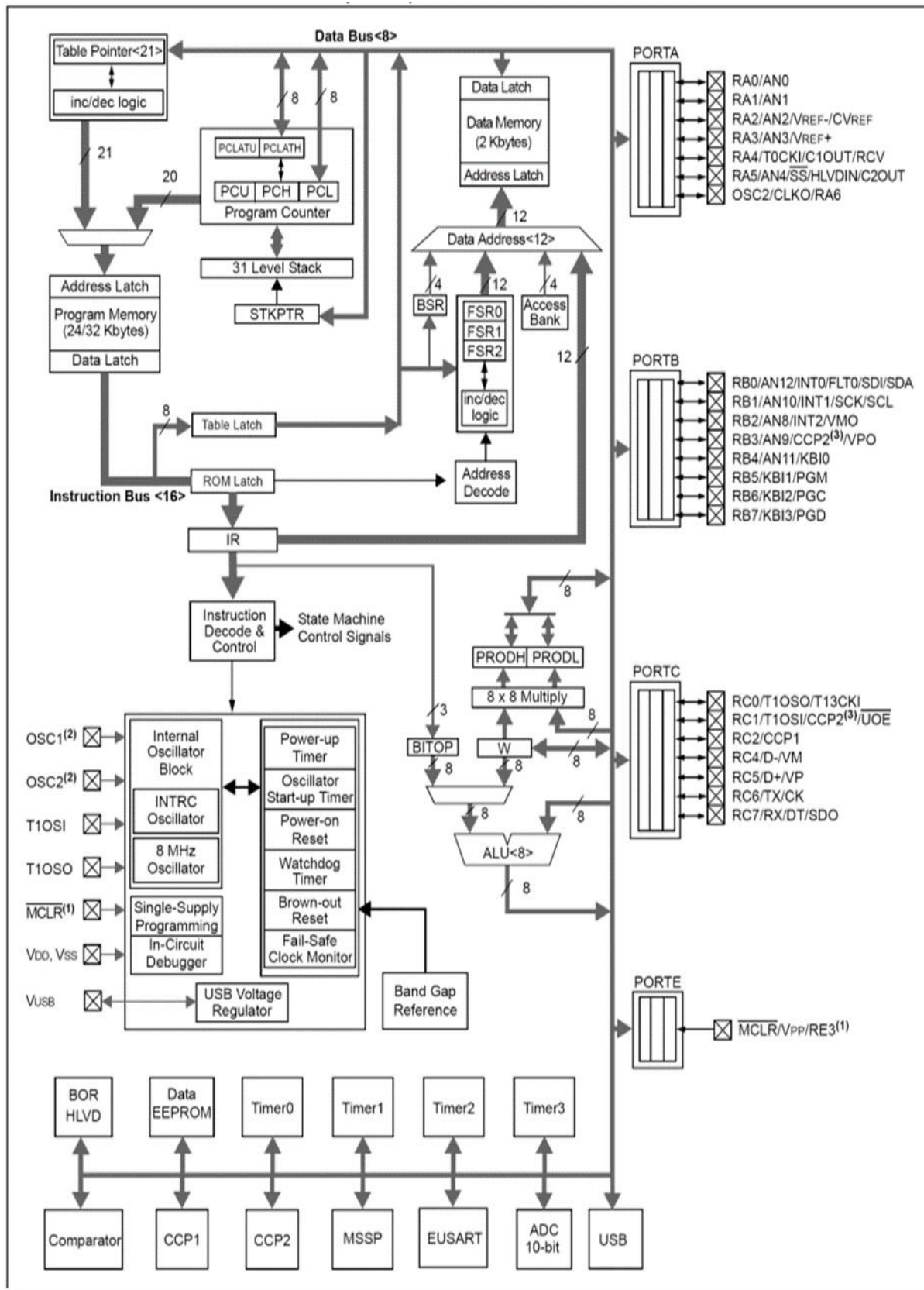
- [1] RJE.B, WAZ. R. "Projet de réalisation d'une maquette didactique à base de PIC 16F877." 2007.
- [2] BERNAOUI.S. "Étude et réalisation d'un kit de développement d'applications à base de PIC de la famille 16Fxxx." 2012.
- [3] Petitpa, P. "PIC18F4550 Datasheet." Disponible sur : <https://philpetitpa.pagesperso-orange.fr/PIC18F4550.pdf>. Consulté le 17/02/2023.
- [4] NECHICHE.S, REMICHI.H. "Étude et réalisation d'un banc de mesure pour la surveillance d'un moteur à base de la carte de développement OLIMEX." 2013.
- [5] Smolnikar, M., Mohorcic, M. "A Framework for Developing Microchip PIC Microcontroller-Based Applications." WSEAS Transactions on Advances in Engineering Education, 5(2), 87. 2008.
- [6] "Fonctionnement de l'afficheur de 7 segments à 4 digits." Disponible sur : <https://ledisrupteurdimensionnel.com/arduino/fonctionnement-de-lafficheur-de-7-segments-a-4-digits/>. Consulté le 15/03/2023.
- [7] Fekik, A. "Chapitre III: Systèmes d'entrées et affichage des données." 2018.
- [8] "The Fundamentals of Digital Potentiometers." Disponible sur : <https://www.digkey.fr/fr/articles/the-fundamentals-of-digital-potentiometers#:~:text=Le%20potentiom%C3%A8tre%20est%20un%20composant,utilisateur%20sur%20un%20arbre%20rotatif>. Consulté le 16/03/2023.
- [9] "Mesurer une température avec un capteur LM35 et une carte Arduino/Genuino." Disponible sur : <https://www.carnetdumaker.net/articles/mesurer-une-temperature-avec-un-capteur-lm35-et-une-carte-arduino-genuino/#:~:text=Qu'est%2Dce%20qu'un%20capteur%20LM35%20%3F,-Brochage%20du%20capteur&text=Le%20capteur%20de%20temp%C3%A9rature%20LM35,une%20fiabilit%C3%A9%20%20toute%20%20%C3%A9preuve>. Consulté le 16/03/2023.
- [10] "Arduino Moteur DC Shield." Disponible sur : <https://electrotoile.eu/arduino-moteur-DC-shield.php>. Consulté le 17/03/2023.
- [11] HADDI.S, CHERIFI.A. "Conception et réalisation d'un système de contrôle sans fil de moteurs à courant continu avec une interface GUI LabVIEW." 2021.

[12] "ULN2001 Datasheet." Disponible sur  
: <https://www.st.com/resource/en/datasheet/uln2001.pdf>. Consulté le 25/03/2023.

[13] "Servomoteur - Glossaire." Disponible sur : <https://harmonicdrive.de>


## **Annexe :**

**Architecture interne du PIC 18F4550 :**



**USB HID mikroBootloader Manuel :**





**User's Guide**


This is a short manual that describes the operation of *mikroBootloader application*. You will find that **five simple steps** are all it takes to successfully load your new program into the MCU.

**You should know this before you start**

- 1** *USB HID Bootloader application* may only be used with the **appropriate device USB HID Bootloader firmware version**, which is free of charge and can be downloaded from mikroElektronika's website.
- 2** Bootloader uses **USB HID class connection**, and no additional OS drivers are required for proper operation.


**1. Establish USB link**  
USB HID mikroBootloader manual

Connect your board to your PC. When device is recognized by your OS the gray USB icon will turn red, indicating that USB link is successfully established. **You have 5s to proceed to the next step.** If you wait too long the device will disconnect.




**2. Connect with MCU**  
USB HID mikroBootloader manual

Click on the **Connect** button within 5s. The chip automatically enters the Bootloader mode and is ready for further instructions. If your USB link disconnects, just reset the chip to reestablish the link. Then proceed with step 2 again.



**3. Choose HEX file**  
USB HID mikroBootloader manual


Load the program you want to upload into the chip by clicking on the **Browse for HEX** button or simply **drag and drop** your HEX on to mikroBootloader window. You'll be notified in the **History Window** that the file is opened.



**4. Start Bootloader**  
USB HID mikroBootloader manual


Start bootloader by clicking on the **Begin uploading** button. You can monitor the bootloading process in the progress bar.

After the process is completed, a new message window will inform you of the bootloading success.



**5. Reset chip**  
USB HID mikroBootloader manual

Bootloader firmware will **automatically reset the MCU**, after which you should **wait 5s** for your newly loaded program to start.



for more information visit <http://www.mikroe.com>

**MikroElektronika**  
DEVELOPMENT TOOLS | COMPILERS | BOOKS

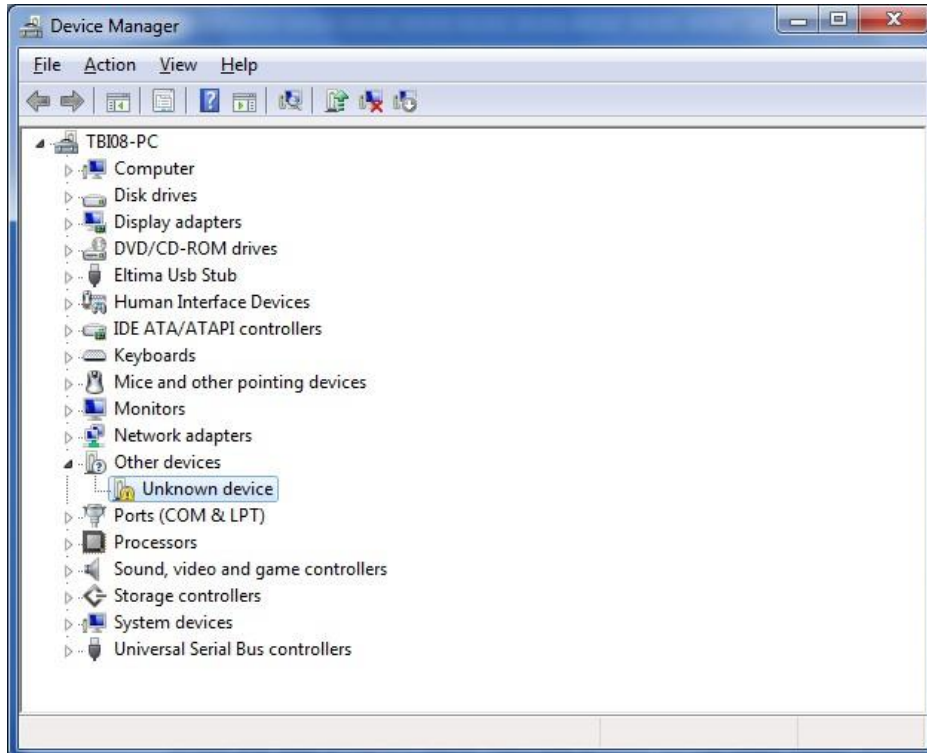
### Install Bootloader driver :

In order to install Bootloader driver, we need to enter into the Boot mode. Keep pressing the reset button while connecting the USB cable to the port. This will enter to the boot mode and you can see below status in Taskbar.

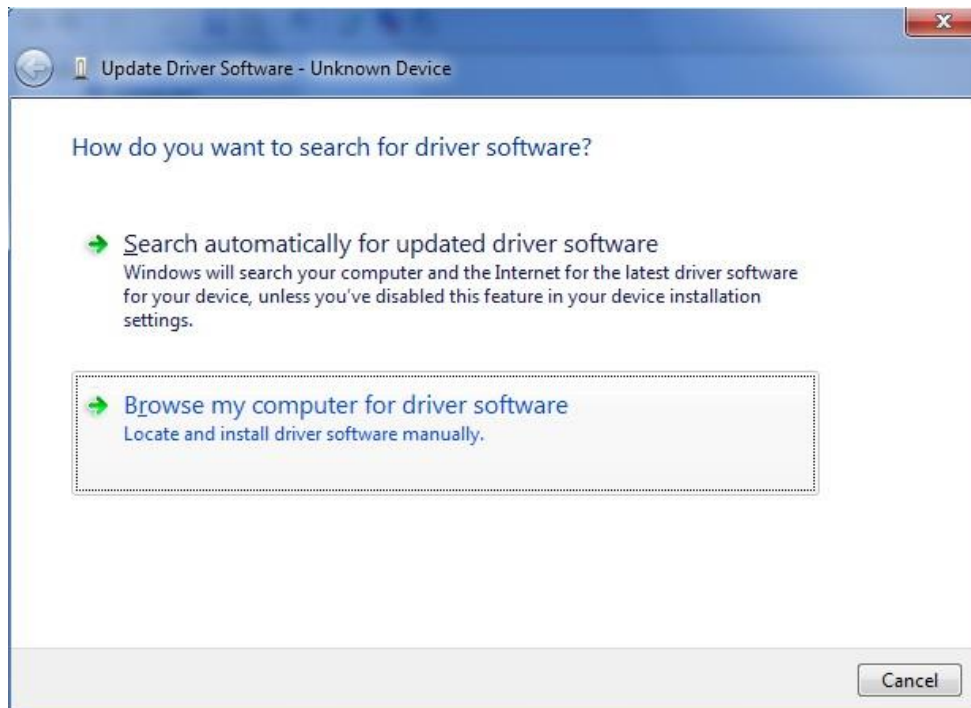


The driver installation will fail because Windows doesn't have the driver files by default. So we need to point to the required files to install it manually.

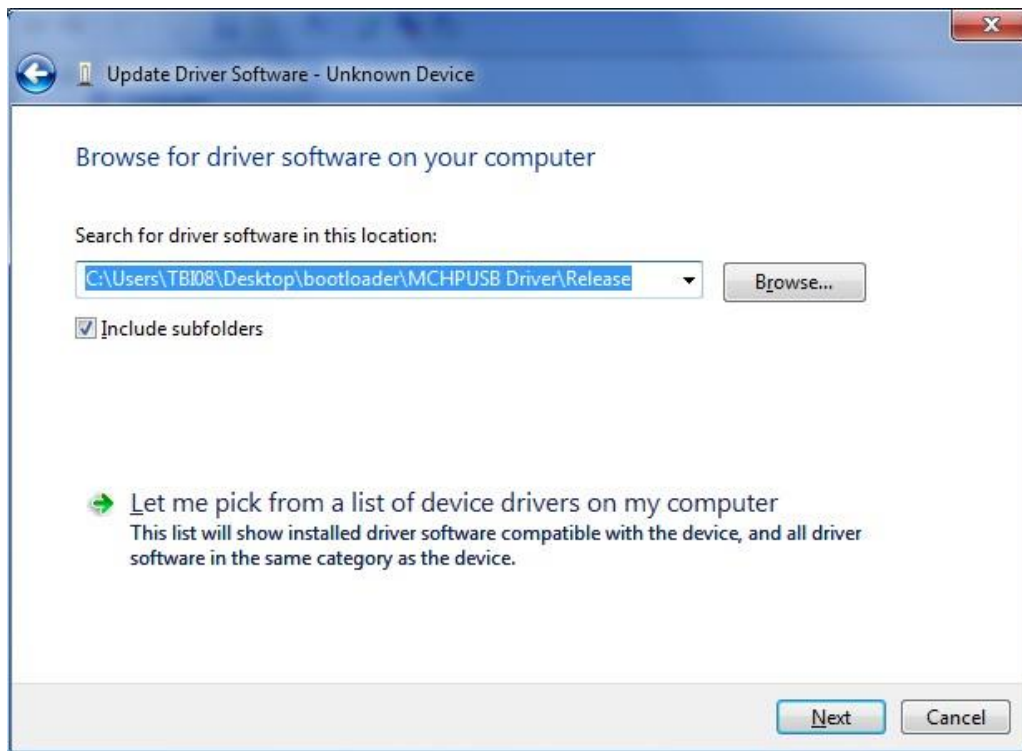
To install, take **Device Manager** (from Control panel). There you can see a section **Other Devices**. The **Unknown Device** which is listed under is our required device.



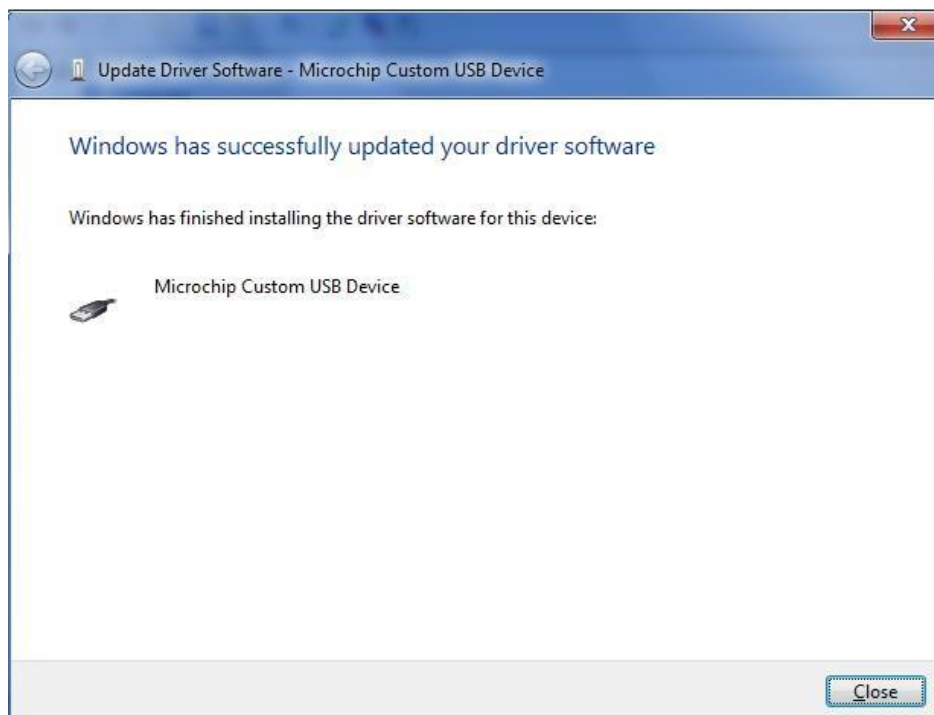
Right click on the **Unknown Device** and select **Update Driver Software**.



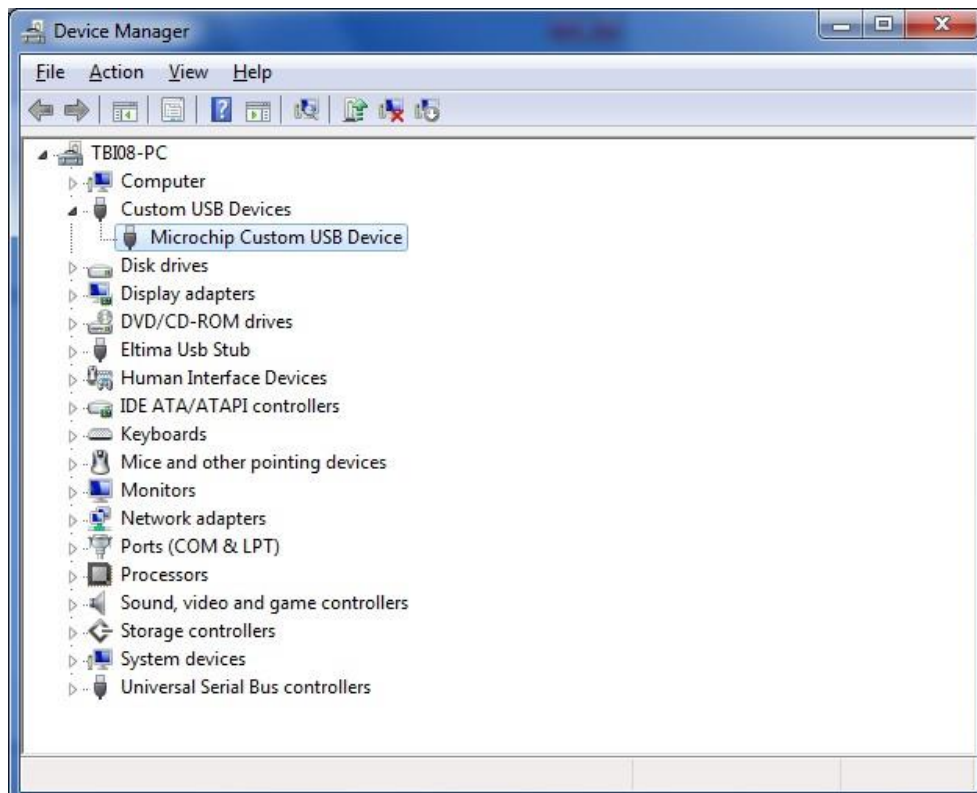
Use the browse button to point to the **bootloader\MCHPUSB Driver\Release**. Press Next. Windows may warn you that the driver is not from verified source. Select Install it anyway.



Next window will show the driver installing status. After a while, Windows would complete the installation.



Now you can examine the Device manager screen, under Custom USB Devices, there is our Microchip Custom USB Device.



Once the Bootloader is installed, start programming your projects.

