

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي و البحث العلمي

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE MINISTÈRE DE
L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

جامعة سعد دحلب البليدة
UNIVERSITE SAAD DAHLEB DE BLIDA



كلية الهندسة - دائرة الإلكترونيك
FACULTÉ DES SCIENCES DE L'INGÉNIEUR
DEPARTEMENT D'ELECTRONIQUE

**MÉMOIRE DE
PROJET DE FIN D'ETUDES
POUR L'OBTENTION DU DIPLÔME DE
MASTER EN ÉLECTRONIQUE**

OPTION : AUTOMATIQUE

**MISE EN ŒUVRE D'UN KIT DIDACTIQUE DE COMMANDE
D'ASCENCEUR.**

Proposé par : M^{elle}. CHENTIR AMINA

Présenté par : DJOUAT AHMED

N° :

Session: Octobre 2011



Dédicaces

Je dédie ce modeste travail à :

*Mes chers parents pour leur encouragement
et leur soutien durant toutes mes années
d'études.*

Mes sœurs: Madiha, Siham et Assia.

Toute ma grande famille.

*Mes amis: Mourad, Mohamed, Halim, Abd
elhak, Abd errahmène, Yassin*

*Et tous mes amis de la promotion Master
Automatique 2010-2011*

A tous qui connaissent Ahmed Anis.





Remerciements

Tout d'abord je remercie DIEU de m'avoir donné la force et le courage d'accomplir ce travail.

Je souhaite remercier chaleureusement Melle Chentir Amina, ma promotrice, pour l'attention constante qu'elle a portée à ce travail. Merci Melle Chentir de m'avoir soutenu et supporté pendant cette période de projet.

Je remercie vivement tous les membres de l'équipe Parole de m'avoir permis de travailler au sein de leur laboratoire.

J'exprime également mes sincères remerciements aux membres de jury d'avoir accepté l'examination de mon modeste travail.

Enfin, je tiens aussi à exprimer ma reconnaissance à mes chers parents pour leur patience, leur tolérance, et leur soutien moral.



Résumé

Notre projet de fin d'études intitulé « Mise en œuvre d'un kit didactique de commande d'ascenseur » consiste à étudier, à concevoir et à réaliser des cartes de commande d'un ascenseur. Ces cartes sont :

- ✚ Une carte de commande câblée.
- ✚ Une carte de commande par microcontrôleur PIC 16F877.

Abstract

Our final graduation project entitled: "Implementation of an educational kit elevator control" consists in studying, designing and carrying out cards of control of an elevator. These cards are:

- ✚ A control board cable.
- ✚ A control board by microcontroller PIC 16F877.

ملخص

لدينا مشروع تخرج بعنوان: "تنفيذ عنصر تحكم تعليمي لمصعد كهربائي" يتمحور على دراسة شاملة, وفهم دقيق وتطبيق فعلي لبطاقات التحكم بالمصعد والتي هي :

✚ بطاقة التحكم بالأسلاك الكهربائية .

✚ بطاقة التحكم عن طريق المتحكم PIC 16F877 .

Introduction générale

Les systèmes automatisés sont de plus en plus présents dans notre quotidien comme dans le monde industriel. La technologie, dont l'objectif est d'apporter aux élèves une culture et une ouverture sur le monde technologique actuel, aborde les systèmes automatisés au cycle central ainsi qu'au cycle d'orientation.

Grâce au développement de la microélectronique, on a vu l'apparition de plusieurs familles de composants puissants capables de gérer une multiplicité de processus. Le microcontrôleur, qui est d'ailleurs le résultat de différentes expériences acquises dans la conception des microprocesseurs, fait partie de ces composants. Il en est de même pour les systèmes automatisés destinés à être utilisés dans le domaine de la robotique, qui s'avère un domaine de plus en plus prometteur au service de l'être humain ; devenu incapable de subvenir aux besoins nombreux et pressants de la vie actuelle.

Depuis la plus haute antiquité, les hommes ont cherché un moyen mécanique leur permettant de réduire considérablement les efforts nécessaires au déplacement de charges dans le sens vertical.

C'est seulement au XIXe siècle que le transport vertical des personnes et des charges prend toute son importance, en raison notamment de l'industrialisation. En 1853 naît le concept d'ascenseur moderne, grâce à l'invention de l'Américain Elisha Graves Otis. Ce maître mécanicien met au point le parachute, système de sécurité destiné à retenir les monte-charge en cas de rupture de câble. Il en démontre la pertinence lors d'une exposition dans le hall du Crystal Palace à New-York.

Progressivement, l'ascenseur va s'industrialiser pour mieux répondre aux besoins d'une urbanisation croissante qui va atteindre son apogée durant les années 1970. S'il devient moins esthétique, l'ascenseur prend toute son importance en permettant aux immeubles d'habitation ou de bureaux de s'élever. Dans le même temps, l'installation d'ascenseurs commence à se généraliser dans les immeubles d'habitation existants.

A la fin des années 80, l'industrie de l'ascenseur intègre toutes les technologies mises en œuvre dans d'autres secteurs. C'est le cas, par exemple, de l'électronique de puissance qui permet le contrôle direct des moteurs à courant alternatif par la variation de tension et de fréquence.

Désormais programmables, les ascenseurs sont plus confortables, moins gourmands en énergie, plus spacieux et plus accessibles aux personnes âgées et handicapées. Cette décennie a été aussi marquée par l'apparition des ascenseurs panoramiques qui rencontrent un

engouement auprès des architectes. Ils s'installent tout aussi bien en façade qu'au cœur d'un bâtiment et apportent de nouvelles dimensions dans la conception architecturale (mouvement, point de vue sur les sites).

A la fin des années 90, les directives Européennes « nouvelles approches » ouvrant la possibilité de conception plus originale, alliées aux innovations technologiques permettant une grande compacité des équipements, ont favorisé l'apparition de concept tel que l'ascenseur sans local de machine.

L'ascenseur d'aujourd'hui est plus communicant. Les ascensoristes développent des systèmes d'information et de communication de plus en plus performants permettant un dialogue avec l'extérieur. Avec ces systèmes, associés à une organisation adaptée du service de maintenance, il est possible de mieux satisfaire les exigences accrues des clients et des usagers en matière d'assistance aux personnes et de disponibilité de l'ascenseur.

Les nouvelles technologies en matière de système d'entraînement (haut rendement et régulation performante) et de gestion de plus en plus sophistiquée et intelligente du trafic conduisent à des ascenseurs ayant de meilleures performances énergétiques et donc de plus en plus respectueux de l'environnement.

La technologie dans le monde informatique est adaptée. A cet niveau, les services et les sociétés informatiques doivent donc suivre et accompagner cette dynamique, de ce fait, ils sont amenés à revoir leurs démarches informatiques afin d'avoir une meilleure réactivité, soit face à la concurrence, soit pour préserver une avance. C'est dans ce cadre que s'inscrit notre projet de fin d'étude.

En effet, ce projet consiste à étudier, à concevoir et à réaliser des cartes de commande d'un ascenseur, soient :

- Une carte à commande câblée.
- Une carte de commande par microcontrôleur PIC 16F877.

Notre projet va comporter essentiellement 3 chapitres. Le premier chapitre sera consacré aux principes de base du microcontrôleur PIC16F877 et ses applications. Le deuxième chapitre va présenter une étude théorique du montage avec une représentation de l'ascenseur.

En fin le troisième chapitre sera consacré à la simulation des différentes cartes de commande réalisées par le logiciel PROTEUS, à dégager les résultats et conclusions aux quels on a abouti au terme de notre étude et faire une réalisation pratique pour les deux cartes de commande. Nous terminerons par une conclusion générale où nous ferons le bilan global de notre travail.

1-1 Introduction

Les PICs sont des microcontrôleurs à architecture RISC (Reduce Instructions Construction Set), ou encore composants à jeu d'instructions réduits.

L'avantage est que plus on réduit le nombre d'instructions, plus leur décodage sera rapide ce qui augmente la vitesse de fonctionnement du microcontrôleur.

Les microcontrôleurs PICS sont des composants **STATIQUES**, c'est à dire que la fréquence d'horloge peut être abaissée jusqu'à l'arrêt complet sans perte de données et sans dysfonctionnement. Ceci par opposition aux composants **DYNAMIQUE**, dont la fréquence d'horloge doit rester dans des limites précises [1].

Dans ce chapitre, on va étudier le microcontrôleur, car il est l'élément de base de notre carte de commande.

1-2 Définition

Un microcontrôleur se présente sous la forme d'un circuit intégré réunissant tous les éléments d'une structure à base de microprocesseur. Voici généralement ce que l'on trouve à l'intérieur d'un tel composant :

- Un microprocesseur (C.P.U.).
- De la mémoire de donnée (RAM et EEPROM).
- Des mémoires programmées :
 - ROM dont le contenu est programmée lors de sa fabrication.
 - PROM programmable électriquement une seule fois par le développeur (appelée aussi OTPROM).
 - EPROM programmable électriquement et effaçable aux U-V (appelée aussi UVPRM).
 - EEPROM programmable et effaçable électriquement.
- Des interfaces parallèles pour la connexion des entrées / sorties.
- Des interfaces séries (synchrone ou asynchrone) pour le dialogue avec d'autres unités.
- Des timers pour générer ou mesurer des signaux avec une grande précision temporelle.
- Des convertisseurs analogique / numérique pour le traitement de signaux analogiques.

Un PIC n'est rien d'autre qu'un microcontrôleur, c'est à dire une unité de traitement de l'information de type microprocesseur à laquelle on a ajouté des périphériques internes permettant de réaliser des montages sans nécessiter l'ajout de composants externes.

L'horloge fournie au microcontrôleur **PIC** est pré divisé par 4 au niveau de celle-ci. C'est cette base de temps qui donne le temps d'un cycle. Si on utilise par exemple un quartz de 4MHz, on obtient donc 1000000 de cycles/seconde, or, comme le microcontrôleur **PIC** exécute pratiquement une instruction par cycle, hormis les sauts, cela donne une puissance de l'ordre de 1MIPS (**1 Million d'Instructions Par Seconde**).

1-3 Les différentes familles du microcontrôleur PICs

Il y en a trois grandes familles de PICs :

- La famille **Base Line**, qui utilise des mots d'instructions de **12 bits**.
- La famille **Mid-Range**, qui utilise des mots de **14 bits** (et dont font partie la 16F84 et 16F876).
- La famille **High-End**, qui utilise des mots de **16 bits**.

Tous les microcontrôleurs PICs Mid-Range ont un jeu de 35 instructions et 4 sources d'interruptions :

- Interruption externe commune avec la broche RB0.
- Interruption due au TIMER.
- Interruption sur changement d'état des broches de port RB0 à RB1.
- Interruption de fin d'écriture en EEPROM.

1-4 Identification du microcontrôleur PIC

Pour identifier un microcontrôleur PIC, on utilise simplement son numéro. Les 2 premiers chiffres indiquent la catégorie du microcontrôleur **PIC**, **16** indique une **PIC Mid-Range** [2].

Vient ensuite parfois une lettre

- **L**: Celle-ci indique que le microcontrôleur **PIC** peut fonctionner avec une plage de tension plus tolérante.
- **C** indique que la mémoire programme est une **EPROM** ou plus rarement une **EEPROM**
- **CR** pour indiquer une mémoire de type **ROM**
- **F** pour indiquer une mémoire de type **FLASH**

1-5 Architecture du microcontrôleur PIC

Le schéma ci-dessous de la **Figure1.1** représente l'architecture simplifiée d'un PIC et les principaux blocs fonctionnels présents à l'intérieur.

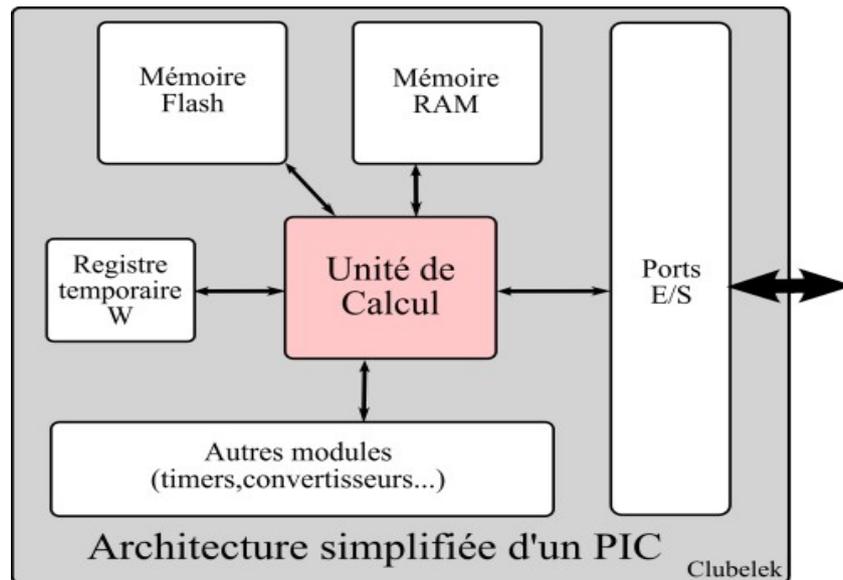


Figure 1.1 : Architecture interne de pic [1]

Et la description des blocs fonctionnels de cette figure est comme cela :

- ❖ **Mémoire flash** : C'est une mémoire réinscriptible qui conserve ses données lorsque le PIC n'est pas alimenté. Elle est utilisée pour stocker le programme. A chaque ligne du programme est attribuée une adresse qui permettra à l'unité de calcul de se repérer.
- ❖ **Mémoire RAM** : C'est une mémoire volatile (qui s'efface quand le PIC n'est plus alimenté). Les variables utilisées au cours du programme sont stockées à cet endroit.
- ❖ **Unité de Calcul** : C'est le cœur du microcontrôleur. Ici se déroulent toutes les opérations à une vitesse définie par la fréquence d'horloge (fréquence d'oscillation divisée par 4).
- ❖ **Registre temporaire W** : C'est ici qu'est stockée la valeur nécessaire pour une opération de l'unité de calcul.
- ❖ **Ports E/S (Entrées/Sorties)** : Ce sont les unités qui font le lien entre ce qui se passe à l'intérieur du PIC et l'extérieur.
- ❖ **Modules annexes** : Toute les fonctions annexes (timers, comparateurs, convertisseurs analogiques/numériques.)

1-6 Choix du microcontrôleur

1-6-1 Pourquoi un microcontrôleur PIC 16F877 ?

Le choix d'un microcontrôleur est important car c'est de lui que dépendent en grande partie les performances, la taille, la facilité d'utilisation et le prix du montage [1].

D'après le **Tableau 1.1** ci-dessous le PIC16F877 est un microcontrôleur de MICROCHIP, qui fait partie intégrante de la famille des Mid Range (16) dont la mémoire programme est de type flash (F) de type 877 et capable d'accepter une fréquence d'horloge maximale de 4Mhz.

Tableau 1.1 : les caractéristiques d'un PIC 16F877 [1]

PIC	FLASH	RAM	EEPROM	I/O	A/D	PORT //	Port série
16F877	8K	368 octets	258	33		NON	USART/ MSSP

Le microcontrôleur 16F877, en boîtier DIP 40 broches (**Figure 1.2**), peut fonctionner à une fréquence maximale de 20 MHz pour un cycle d'instruction de 200ns.



Figure 1.2 : photos réelle d'un PIC16F877

Il est constitué des éléments suivants :

- 8 K mots de 14 bits de mémoire programme du type Flash,
- 256 octets de EEPROM ,
- 368 octets de RAM ,
- 2 Timers de 8 bits et un Timer de 16 bits,
- Un convertisseur A / N 10 bits avec 5 entrées multiplexées,
- Un chien de garde (WatchDog),
- Une interface série synchrone (SPI),
- Une interface série asynchrone (USART).
- 5 ports E/S :

- ❖ Un port A de 6 bits (RA0 à RA5).
- ❖ Un port B de 8 bits (RB0 à RB7).
- ❖ Un port C de 8 bits (RC0 à RC7).
- ❖ Un port D de 8 bits (RD0 à RD7).
- ❖ Un port E de 3 bits (RE0 à RE3).

1-6-2 Les différents registres du PIC

Port A : c'est un port d'entrée/sortie, il contient 6 pins d'entrées /sorties de RA0 à RA5 répartie sur deux registres : le registre PORTA et le registre TRISA. Le bit 6 et 7 ne sont pas implémentés, ils seront lus comme « 0 ».

Au moment de reset on doit forcer une valeur dans le registre ADCON1, pour pouvoir utiliser ce port en entrée/sortie de type générale.

Port B : le port B est le port bidirectionnel de 8 bits. Toutes les broches sont compatibles TTL. Ce registre fonctionne avec la même manière que le port A et TRISA, mais concernant bien entendu les 8 pins de RB. Si on parle de la particularité du port B, puisque les entrées du port B peuvent être connectées à une résistance de rappel (+5v) de manière interne, la sélection s'effectuant par le bit 7 du registre option.

Port C : c'est le port à 8 bits bidirectionnel. Le registre de direction correspondant est le registre TRIS C qui assure des liaisons de communication avec le périphérique externe au microcontrôleur.

Port D : ce port n'est présent que sur le 16F877, il fonctionne d'une façon identique aux autres, dans son mode de fonctionnement général.

Le registre TRISD comporte donc les 8 bits de direction. Les 8 pins I/O, en mode entrée, sont de type « trigger de Schmitt ». Le fonctionnement de ce port dépend de la valeur placée dans TRISE.

Port E : Ce port n'est présent que sur le 16F877. Il ne comporte que 3 pins, RE0, RE1 et RE2 contrairement aux autres ports, les bits non concernés de TRISE sont implémentés pour d'autres fonctions.

On remarque que les pins REX peuvent être utilisés comme pins d'entrées analogiques. C'est le registre ADCON1 qui détermine si ce port sera utilisé comme port I/O ou comme port analogique.

Ces registres spéciaux font partie du SFR (registres fondamentaux) et configurent le microcontrôleur. Certains registres initialisent les périphériques alors que d'autres sont utilisés par le CPU.

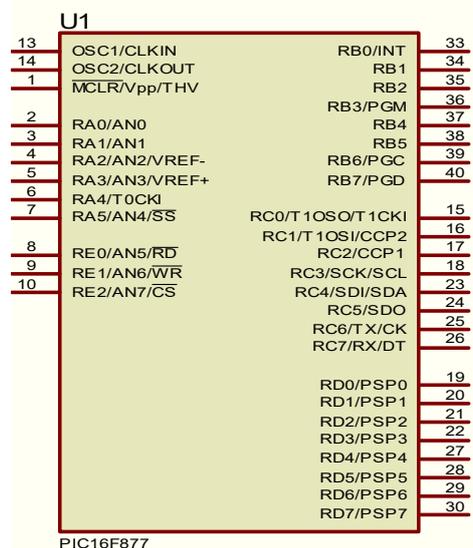


Figure 1.3 : structure générale d'un pic 16F877 [2]

1-7 Les particularités électriques de microcontrôleur PIC 16F877

Ces broches de fonctionnement (Figure 1.3) sont les broches qui permettent au microcontrôleur de fonctionner. Elles doivent être obligatoirement connectées pour qu'elles fonctionnent. Nous les avons classées de la façon suivante:

1-7-1 Les broches d'alimentation

Le PIC 16F877 a quatre broches d'alimentation. Les deux pins pour le Vcc sont le n°12 et n° 31, et pour le Vdd sont les pins n° 11 et n° 32. Il lui suffit une alimentation de 5V. Les pins d'alimentation sont placés d'une part et d'autre en position centrale du PIC. Là où on trouve le quartz, qui peut être remplacé par un résonateur ou par un simple réseau RC. Les condensateurs de découplage, du fait de la fréquence plus importante du quartz utilisé, sont de valeur environ 15pF.

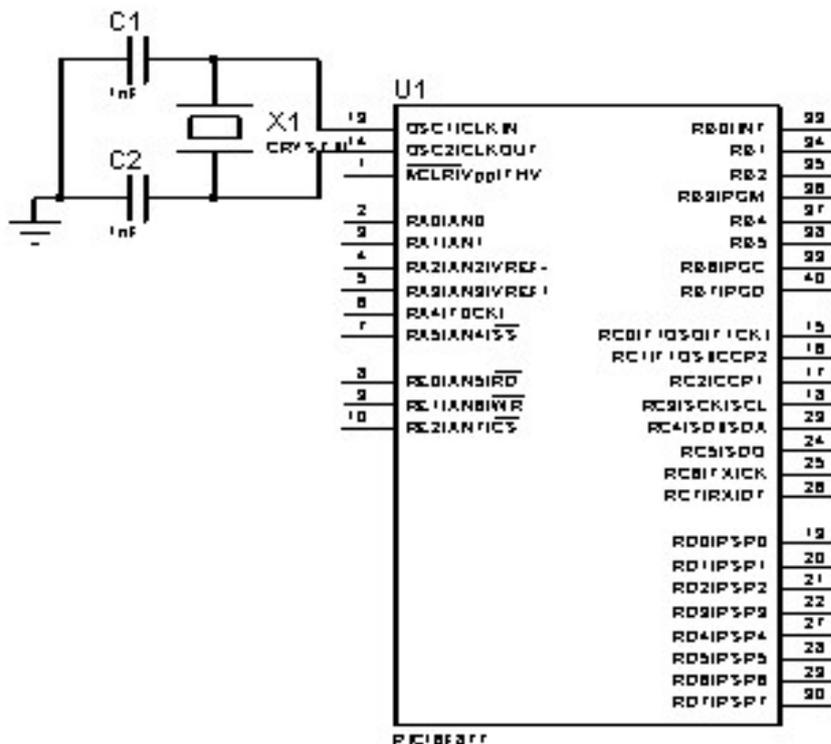


Figure 1.4 : Brochage du circuit d’oscillation PIC16F877 [1]

1-7-2 Les broches de quartz

Dans le cas du 16F877 le quartz utilisé est typiquement un quartz de 4MHz, c'est-à-dire qu'il va fournir 4millions d'impulsions par seconde. Le temps qui s'écoule entre deux impulsions s'appelle un cycle d'horloge qui dure donc 10ns Le quartz de 1 MHz est connecté aux broches osc1 (n° 13) et osc2 (n° 14) comme nous montre la **Figure 1.4**. Il est aussi possible d'utiliser des fréquences plus basses en utilisant un circuit oscillateur externe (mais cela n'a pas d'intérêt en ce qui nous concerne).

1-7-3 La broche de réinitialisation

Une broche particulière du microprocesseur est la broche MCLR (Master CLEAR). Cette broche est d'avantage une broche de contrôle que de fonctionnement. Cette broche a pour effet de provoquer la réinitialisation du microprocesseur lorsqu'elle est connectée à 0. Ceci provoque l'arrêt de programme qui va recommencer à la première instruction. L'ensemble des registres du microprocesseur vont par ailleurs être également remis à leur valeur de départ. Pour que le processeur fonctionne, il faut donc que cette broche soit connectée à 5V.

Un bouton poussoir est prévu pour le reset par l'utilisateur. L'appui sur le bouton poussoir va mettre transitoirement la broche MCLR à (0).

1-7-4 Les ports d'entrées/ sorties

Toutes les autres broches du 16F877A sont des broches de ports. Elles peuvent aussi avoir d'autres rôles si les modules internes du PIC sont actifs. Si on fait le total cela fait 33 broches d'entrées- sorties (E/S) disponibles. D'où les gros avantages du 16F877A est :

- ❖ Le sens E/S des broches est configuré dans un registre appelé TRIS
- ❖ Un registre de données attachées à chaque port.
- ❖ On peut brancher une LED via une résistance directement sur les broches de port en sortie.

1-8 Conclusion

Au cours de ce chapitre, on a étudié les différentes caractéristiques du microcontrôleur PIC 16F877 qui sera le composant essentiel de notre carte de commande de l'ascenseur. Nous avons choisit le 16F877 pour sa multitude de fonction et le nombre considérable d'entrée sortie selon notre besoin. Cela va nous aider dans la réduction du nombre des composants, la réduction de la carte en plus de sa disponibilité sur le marché. ET pour plus d'informations sur le PIC 16F877, voir l'annexe 1.

2-1 Introduction

La fonction globale d'un système automatisé est une modification sur l'environnement pour laquelle de l'énergie est mobilisée. Sa finalité est d'apporter une valeur ajoutée à la matière d'œuvre (énergie, information, matière) qui est mise en jeu. Les applications sont vastes : industrie, environnement, domotique (système de régulation et de sécurité), gestion à distance, etc..... . On identifie quatre entités dans un système automatisé: l'opérateur, la partie commande, la partie opérative et les flux qui le traversent. Ces flux mettent en avant deux chaînes fonctionnelles : chaîne d'acquisition : détecter et communiquer (notion de capteurs) et chaîne d'action : produire et transformer (notion d'actionneurs). Parmi les systèmes automatisés nous avons l'ascenseur comme nous montre la **Figure 2.1**. Dans un ascenseur, l'ensemble électromécanique (cabine, moteur, portes,...) constitue la partie opérative, les boutons d'appel, la logique (combinatoire, séquentielle, programmée) constituent la partie commande [3].

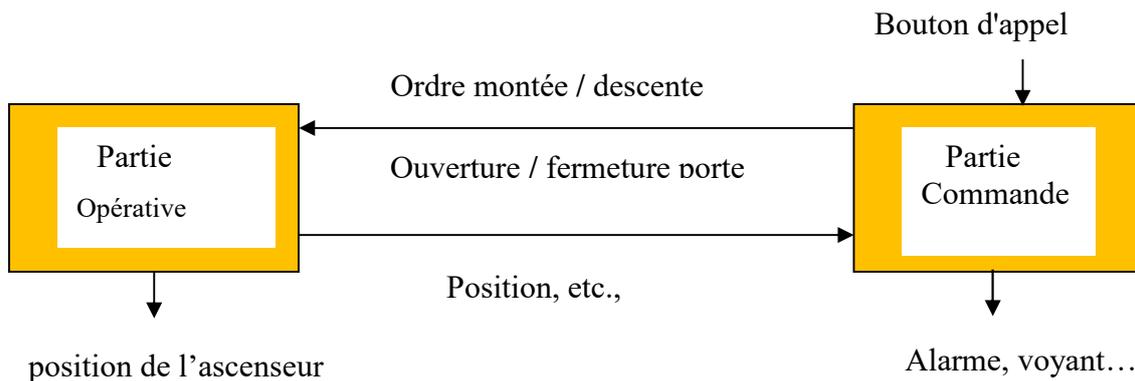


Figure 2.1 : modélisation de l'ascenseur [3]

2-2 Présentation de l'ascenseur

C'est un dispositif utilisé pour transporter des personnes entre différents étages d'immeuble. Il est constitué d'une cabine qui se meut dans une cage généralement verticale. La cabine est suspendue à un câble et contrebalancée par un contrepoids. Son déplacement se fait le long de guides grâce à un moteur entraînant le câble [4]. D'après la **Figure 2.2** l'ascenseur se compose principalement de :

LEXIQUE ASCENSEUR

LOCAL DES MACHINES (SUPERIEUR)

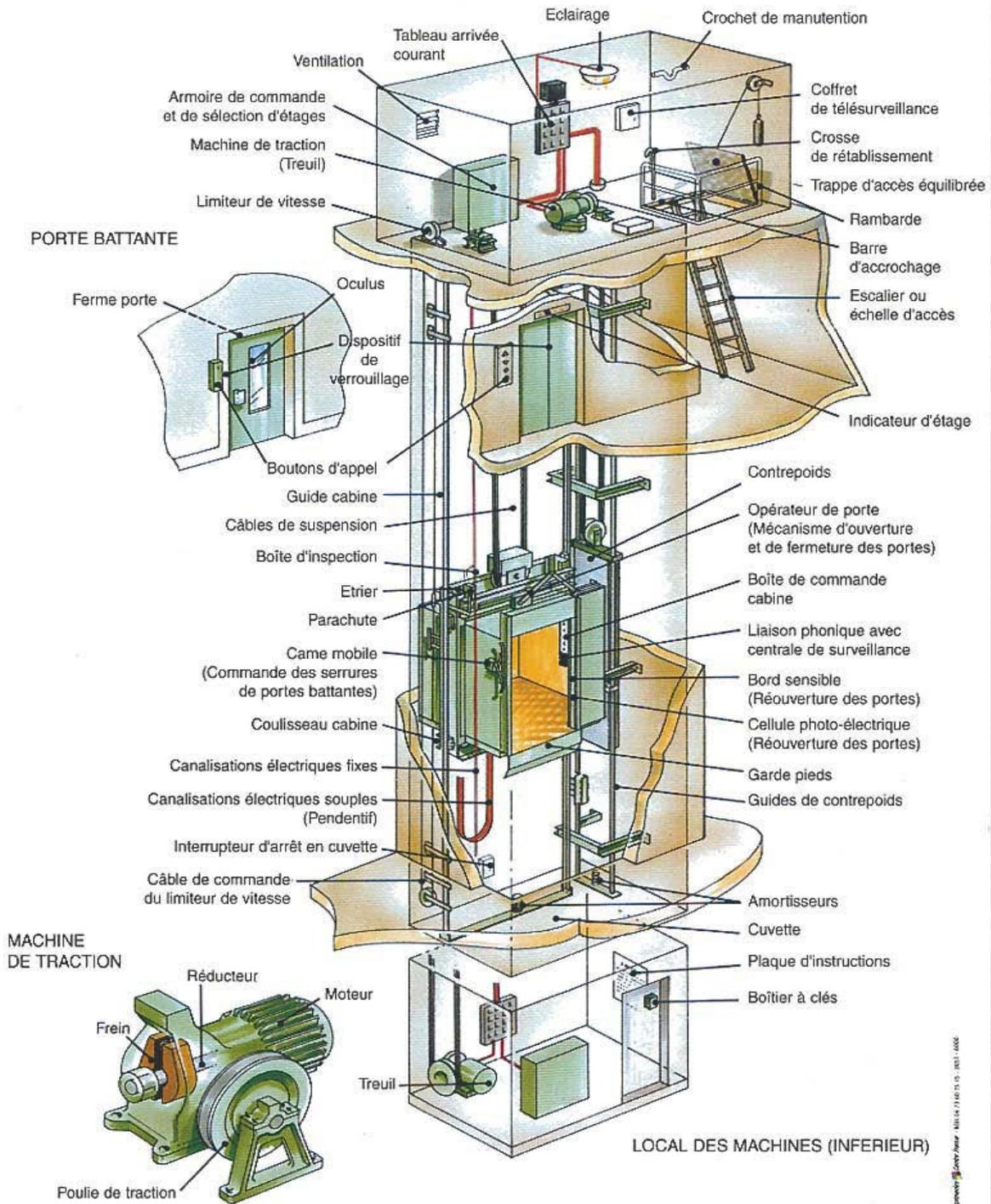


Figure 2.2 : présentation technique de l'ascenseur [4]

La cabine :

Elément composé d'un planché de parois et d'un toit destiné à accueillir les personnes et les marchandises. (La partie visible de l'ascenseur) cet élément est inséré et fixé dans un cadre appelé suspension cabine. La course de la cabine est limitée par 4 capteurs de positions (FC1, FC2, FC3, et FC4) situés à l'étage1, étage2, étage3 et à l'étage4 alors que l'état de la porte est indiqué par des capteurs de portes (CS0, CS1, CS2, CS3).

Câble de traction :

Câble généralement en acier, destiné à suspendre la cabine au contre poids et à déplacer celle-ci.

Le contre poids :

Elément destiné à contre balancer le poids de la suspension métallique contenant des gueuzes en fonte destinées à l'alourdir. Lorsque la cabine d'ascenseur monte, le contre poids descend.

La sécurité :

Pour assurer la sécurité des usagés, différents dispositifs sont prévus à l'ascenseur pour provoquer l'arrêt du moteur en cas d'anomalie. On peut citer:

- ❖ Le parachute: destiné à arrêter et maintenir à l'arrêt la cabine sur ces guides en cas de sur- vitesse à la descente ou de rupture de la gaine.
- ❖ Verrouillage des portes lorsque la cabine se met en mouvement. L'ouverture de ces derniers n'est permise que lorsque la cabine est en face.

Les entrées/sorties :

Après consultation du cahier de charge, on a pu conclure qu'il nous faut 8 entrées et 12sorties :

❖ **Les signaux de sorties :**

- Sens de rotation :

Ce sont les deux signaux de rotation (Sens1 et Sens2) pour commander la montée ou la descente de la cabine.

- Vitesse :

C'est le signal (vitesse) de rotation du moteur (lent=0 ou rapide=1).

- Allume cabine :

C'est un voyant qui permet de commander l'allumage de la cabine.

- Les témoins :

Ce sont les quatre voyants situés aux quatre étages de la cage (LED1, LED2, LED3 et LED4).

❖ Les signaux d'entrées :

- Bouton d'appel:

Quatre poussoirs sont situés aux quatre étages de la cabine pour appeler cette dernière (Appel1, Appel2, Appel3, Appel 4).

- Arrêt d'urgence:

Poussoir qui permet de bloquer l'ascenseur en cas d'anomalie.

- Test portes :

C'est le voyant (PORTE) qui indique l'ouverture de l'une des portes (Led allumée : portes fermées et inversement).

- Présence étages :

Ce sont des voyants (PE1, PE2, PE3 et PE4) qui indiquent la position de la cabine.

PE1 : la cabine est au premier étage...

- Limite haut et bas :

Ce sont des voyants (LIMITEB et LIMITEH) qui indiquent le dépassement de la cabine de sa limite.

Le schéma fonctionnel de notre ascenseur est représenté sur la **Figure 2.3**.

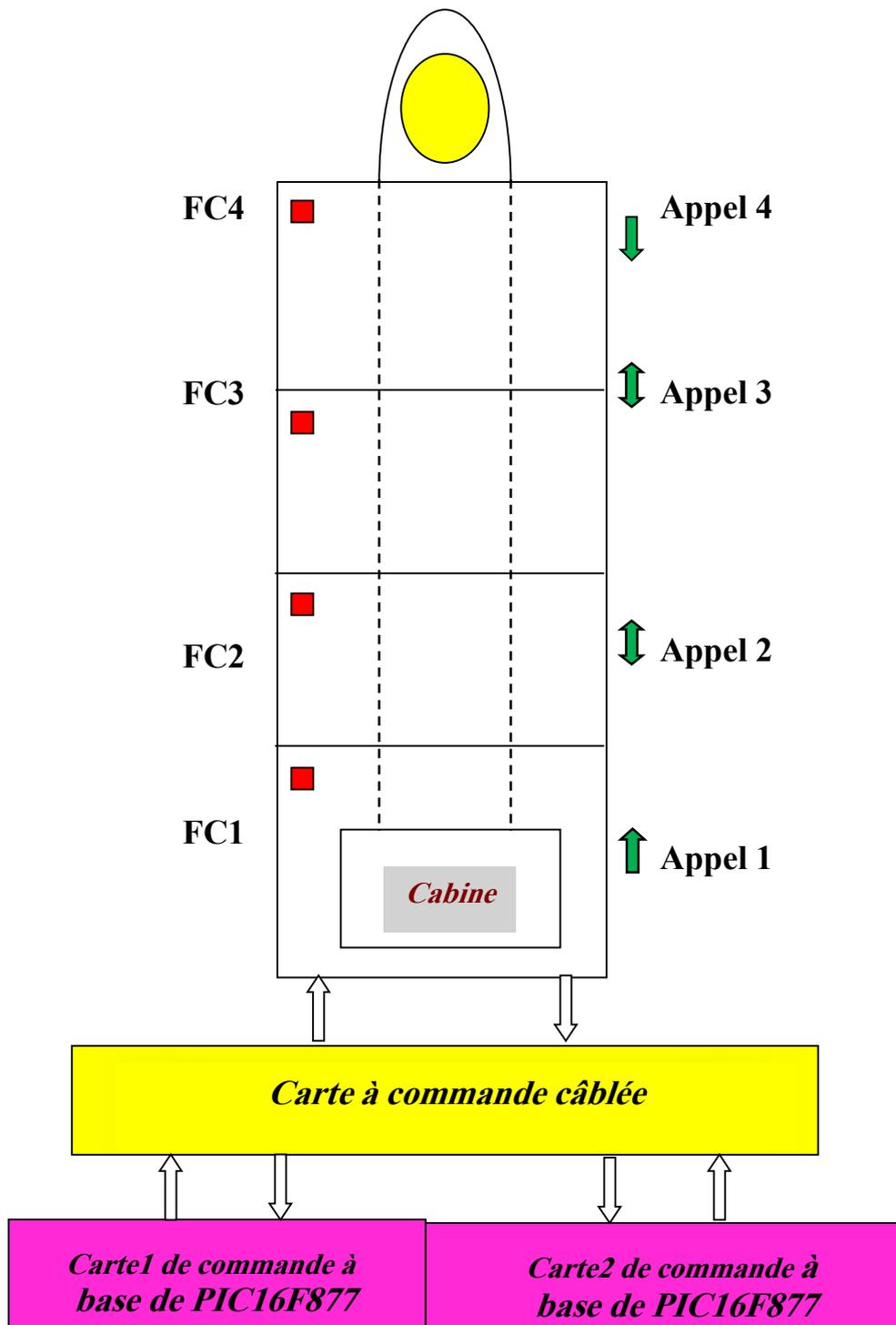


Figure 2.3 : Schéma fonctionnel de l'ascenseur [1]

2-3 Etude du fonctionnement de l'ascenseur

Cette partie consiste à étudier le fonctionnement de l'ascenseur. Pour cela, nous avons choisi l'étude de spécification qui est un processus d'abstraction qui consiste à représenter les aspects essentiels des systèmes.

2-3-1 Spécification des données et des évènements

Le tableau ci-dessous représente une spécification des données et des évènements de notre ascenseur (avec Cmd = Commande).

Tableau 2.1: Spécification des données et des évènements [1]

	Type	Nature	Bord
Cmd-monté	Sortie	discrète	Moteur
Cmd-descente	Sortie	discrète	Moteur
Appel cabine	Entrée	discrète	Bouton d'appel
Présence cabine	Entrée	discrète	Capteur cabine
Etat de porte	Entrée	discrète	Capteur porte
Marche/Arrêt	Entrée	Evénement	opérateur
Message AFF	Sortie	discrète	Afficheur LCD

2-3-2 Diagramme de contexte

La **Figure 2.4** ci-dessous représente le diagramme de contexte de notre projet.

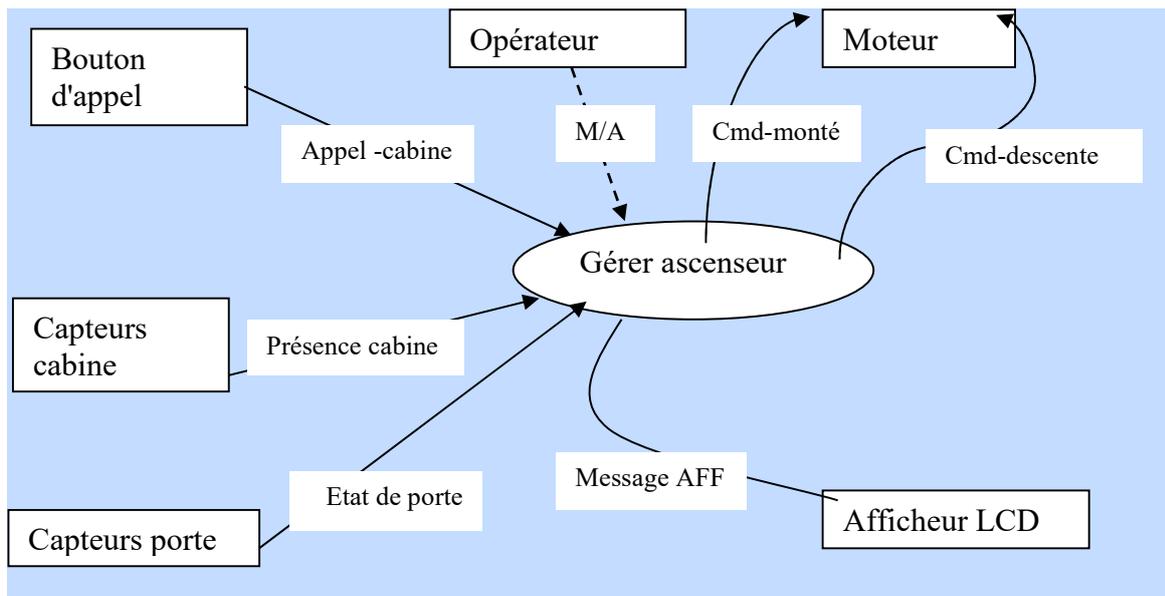


Figure 2.4 : Diagramme de contexte

Ce diagramme contient la fonction principale du système avec les composants complémentaires qui modélisent l'environnement du système. Il permet d'identifier ce qui est à la charge du système et ce qui est à la charge de l'environnement.

2-3-3 Diagramme préliminaire

Ce diagramme permet de représenter la décomposition du système au sous systèmes. Les fonctions des sous systèmes sont représentées par transformation de données comme nous montre la **Figure 2.5** suivante :

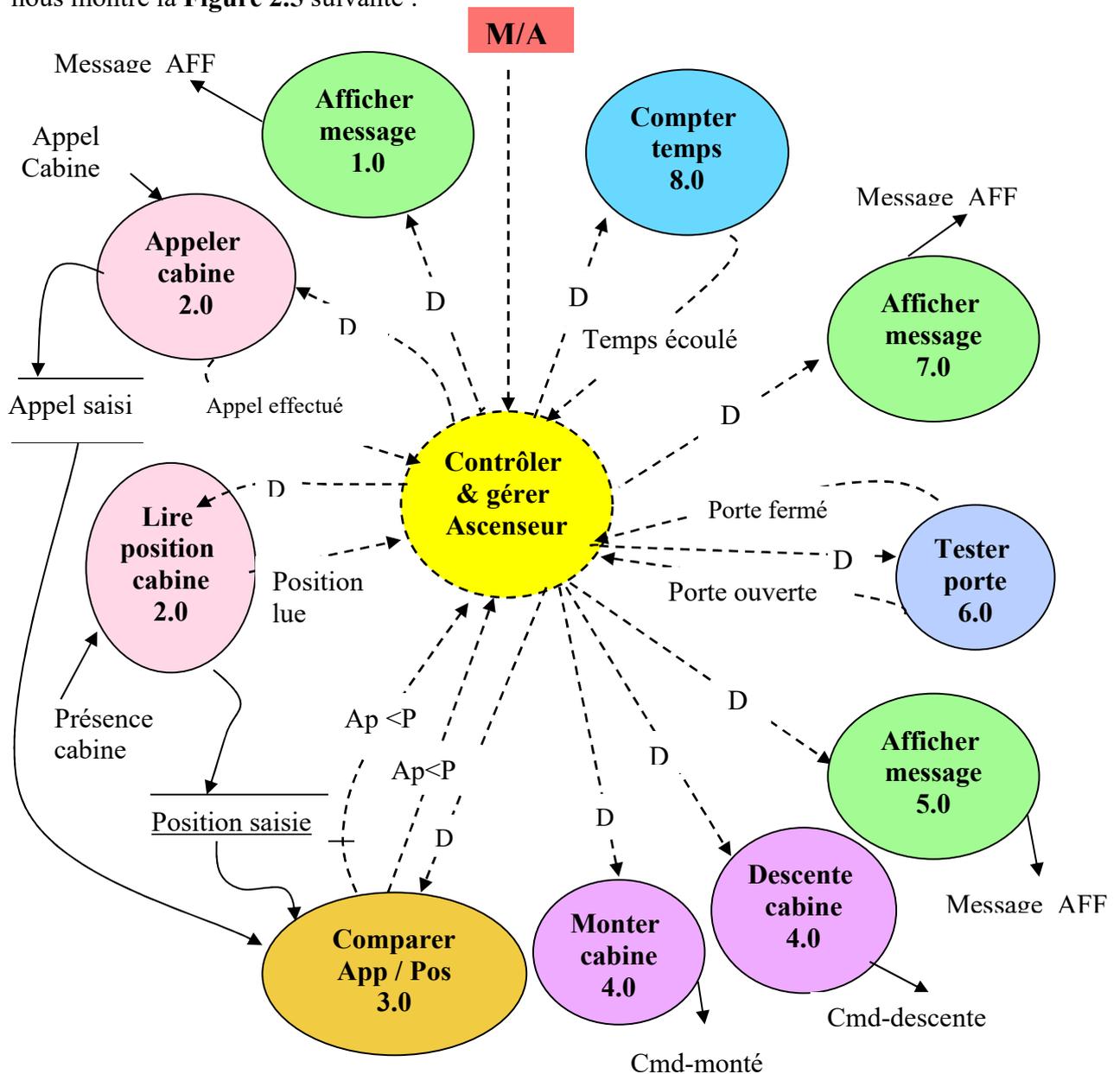


Figure 2.5: Diagramme préliminaire

Le diagramme préliminaire explique les étapes essentielles pour commander notre ascenseur.

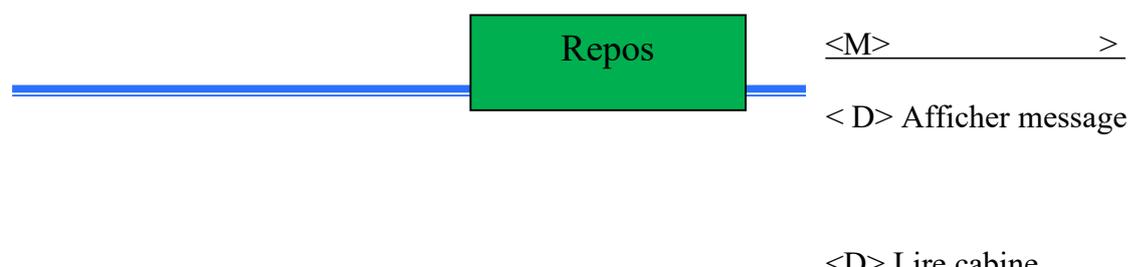
Ces étapes sont :

- Etape1 : après l'ordre de marche (**M/A**) donc l'initialisation du microcontrôleur, l'afficheur affiche le message d'accueil suivant (bien venue...).
- Etape2 : le microcontrôleur lit la position de la cabine actuelle et reçoit l'ordre de l'appel désiré.
- Etape3 : le microcontrôleur compare entre la position de la cabine et l'appel effectué.
- Etape4 : suivant l'étape 3 le microcontrôleur donne les ordres pour commander la cabine (la monter ou la descendre).
- Etape5 : le rôle de l'afficheur dans cette étape est l'affichage des étages désirés (étage1, étage2, ...).
- Etape6 : le microcontrôleur donne les ordres pour gérer la porte (ouvrir et fermer la porte).
- Etape7 : suivant l'étape 6 l'afficheur affiche le message suivant (porte ouverte ou bien porte fermée).
- Etape8 : cette étape représente la fin du cycle (temps écoulé), donc le microcontrôleur revient à l'étape 2 pour recevoir une autre demande d'appel.

(D = veut dire Donner des ordres, Ap = Appel, P =Position, AFF= message d'AFFicheur).

2-3-4 Diagramme d'état transition

Le diagramme de la **Figure 2.6** permet de représenter les différentes étapes de fonctionnement du système, que nous allons implémenter dans notre carte de commande basée sur le PIC16F877.



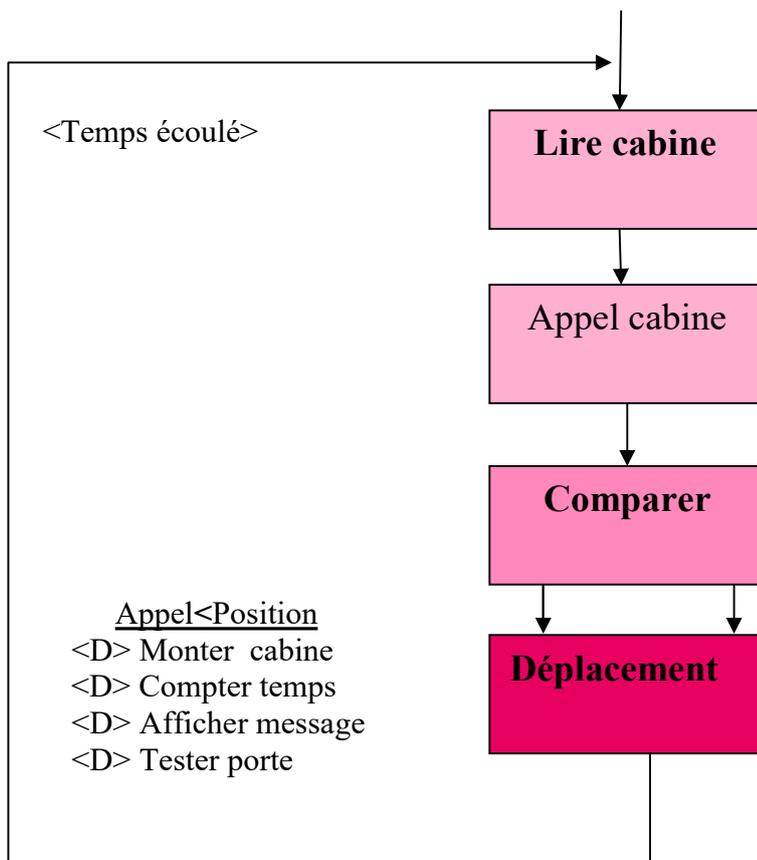


Figure 2.6: Diagramme d'état transition

2-4 Etude de la carte d'alimentation stabilisée

Les systèmes électriques ont besoin toujours d'une alimentation, cette dernière peut être stabilisée, symétrique ou variable.

2-4-1 Description de la carte d'alimentation

Notre montage nécessite une carte d'alimentation stabilisée symétrique fournissant les tensions suivantes :

- 5V pour alimenter le PIC16F877, l'élément essentiel de la carte de commande.
- $\pm 12V$ pour alimenter le circuit de puissance intégré dans la carte de commande câblée.
- 15V pour la vitesse du moteur de rotation.
- La masse (GND).

Pour réaliser cette carte d'alimentation, il nous faut donc :

- Un transformateur abaisseur, qui fournit sur son secondaire une tension alternative très inférieure à celle du secteur (220V/12V) (transformateur à point milieu).
- Un pont redresseur (diodes en pont de Graëtz), qui fournit en sortie une tension non plus alternative mais redressée.
- les capacités de filtrage, qui réduisent l'ondulation de la tension issue du pont redresseur.
- les régulateurs de tension, dont le rôle est de stabiliser le potentiel de sortie par une certaine valeur (5V, 12V, -12V et 15V).
- les capacités de découplage pour la protection des régulateurs de tension.

2-4-2 Schéma synoptique de la carte d'alimentation

La **Figure 2.7** nous montre le schéma synoptique de notre carte d'alimentation :

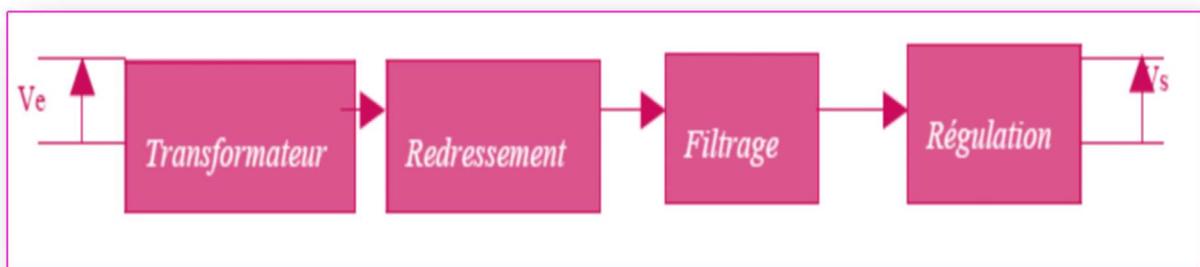


Figure 2.7 : Schéma synoptique de la carte d'alimentation

Avec (V_e : la tension d'entrée et V_s : la tension de sortie).

2-4-3 Choix du transformateur

Notre Carte de développement nécessite une tension d'alimentation de (5v, 15v,-12v et +12v) avec un courant nominale de 0.5A, donc il faut choisir un transformateur à point milieu dont la tension au secondaire est à 12V avec un courant nominal de 1A.

2-4-4 Choix de pont de redressement

Le choix de pont de diode est basé essentiellement sur :

- La tension inverse maximale de diode.
- Le courant moyen direct.

2-4-5 Choix des condensateurs

❖ Choix des condensateurs de filtrage:

Pour obtenir une tension presque constante, il faut brancher un ou plusieurs condensateurs en parallèle juste après le pont redresseur. Plus la valeur de la capacité est élevée plus le filtrage sera meilleur. Les deux principaux critères à considérer dans le choix d'un condensateur sont :

- sa capacité
- sa tension de service

❖ Choix des condensateurs de découplage :

On les choisit de telles sortes qu'ils servent à améliorer la stabilité du régulateur.

2-4-6 Choix de régulateur de tension

Un régulateur de tension est un composant semi-conducteur dont le rôle consiste à rendre quasi continue une tension qui présente une ondulation issue d'un pont redresseur et à stabiliser sa valeur. On a choisit la famille LM78xx (ou LM79xx) qui possède les caractéristiques suivantes :

- Courant de sortie 1A.
- Protection thermique interne contre les surcharges.
- Aucun composant externe nécessaire.
- Plage de sécurité pour le transistor de sortie.
- Limitation interne du courant de court-circuit.

Donc pour notre carte d'alimentation on utilise les régulateurs de tension suivants :

- ❖ 7805 : régulateur de tension qui fournit une tension continue de +5v.
- ❖ 7812 : régulateur de tension qui fournit une tension continue de +12v.
- ❖ 7815 : régulateur de tension qui fournit une tension continue de +15v.
- ❖ 7912 : régulateur de tension qui fournit une tension continue de -12v.

Pour plus d'informations, voir annexe 2.

2-4-7 Schéma électrique de la carte d'alimentation

La **Figure 2.8** représente le schéma électrique de la carte d'alimentation implémentée sur le logiciel Proteus.

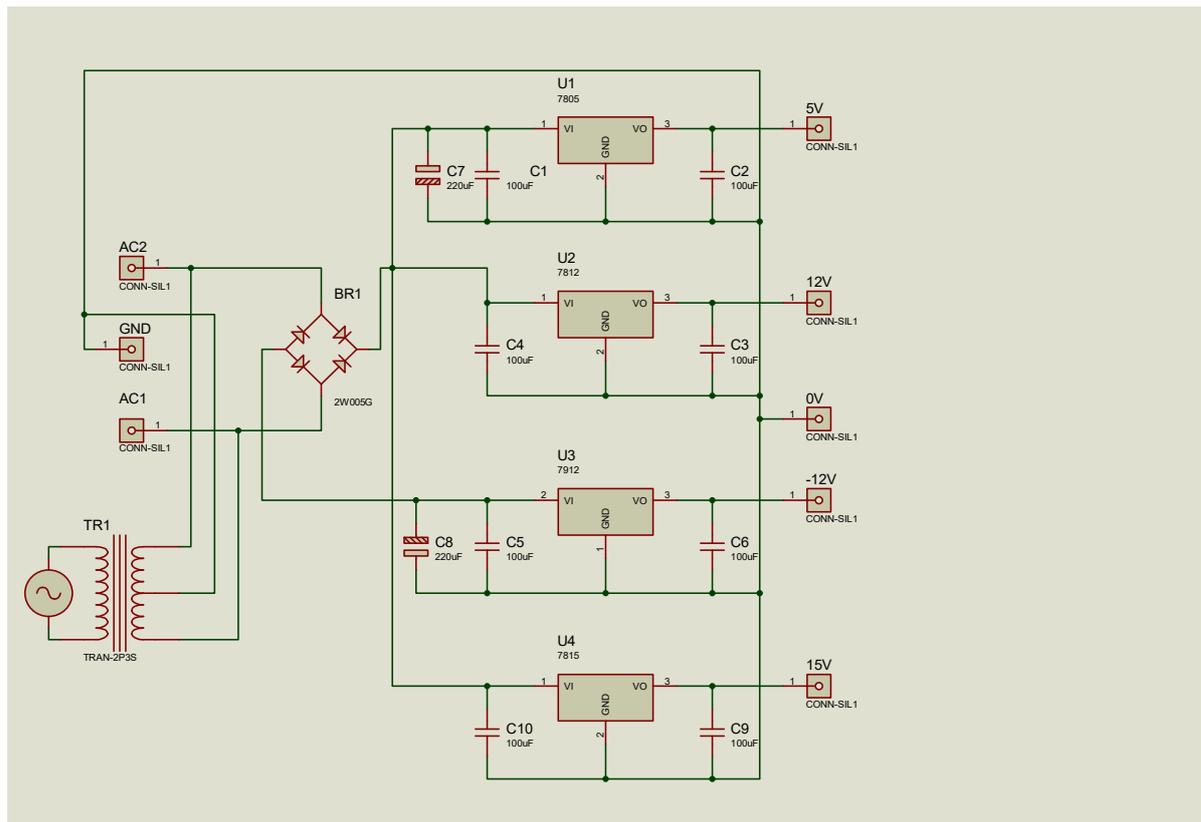


Figure 2.8 : Schéma électrique de la carte d'alimentation

2-5 Carte à commande câblée

Cette carte a pour but de servir d'interface d'entrée/sortie à la commande et au contrôle de la cage d'ascenseur. Elle permet d'avoir directement accès aux différents signaux d'entrée et de sortie.

2-5-1 Schéma synoptique

La **Figure 2.9** représente le schéma synoptique de la carte à commande câblée.

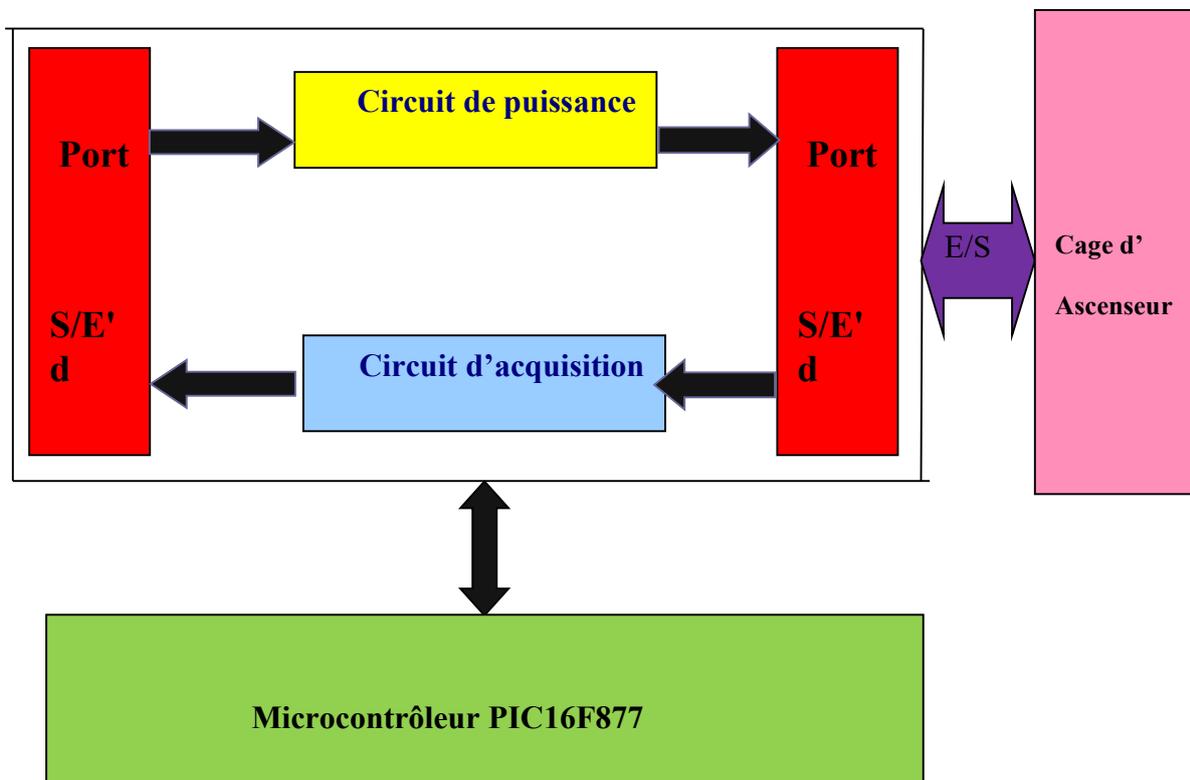


Figure 2.9 : Schéma synoptique de carte de commande câblée

2-5-2 Etude du Circuit de puissance

Dans cette carte on utilise des relais qui représentent le circuit de puissance comme le montre la **Figure 2.10** avec :

- ❖ RL1, pour contrôler les deux vitesses du moteur (Lent et rapide).
- ❖ RL2, RL3 pour le verrouillage mécanique en cas d'urgence avec RL2 : Limite bas et RL3 : Limite haut (de la cage d'ascenseur).
- ❖ RL4, RL5 pour commander les deux sens de mouvement de la cabine (RL4 pour la montée et RL5 pour la descente).

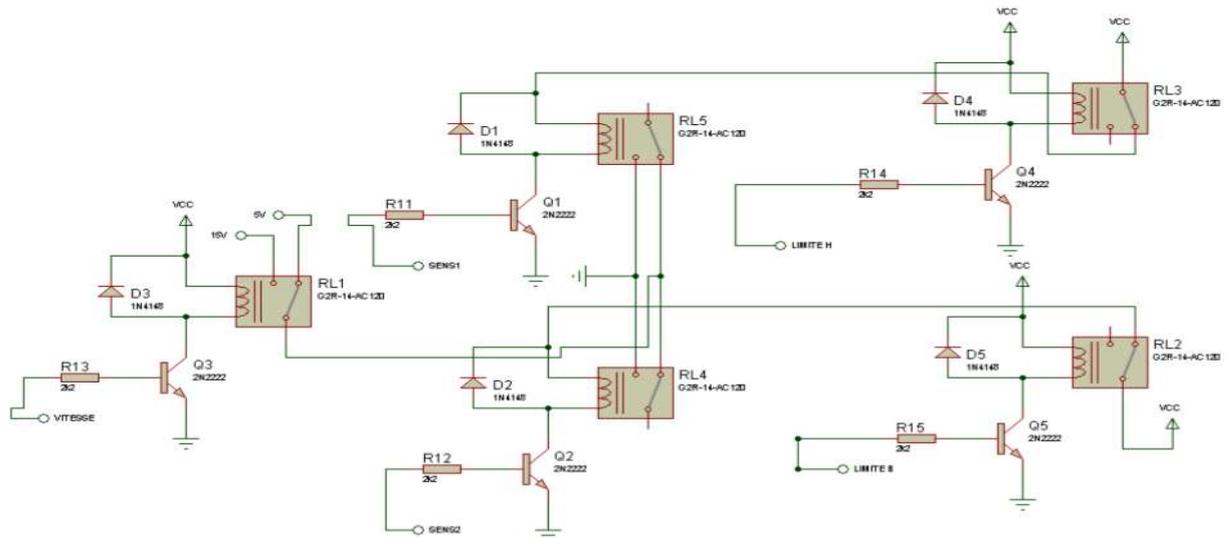


Figure 2.10: Circuit de puissance de carte de commande câblée

2-5-3 Etude du Circuit d'acquisition

Ce circuit comporte les différents signaux d'entrée/sortie comme le montre la **Figure 2.11**. Parmi les applications de notre carte, l'affichage des états logiques des signaux qui sera effectué par des diodes (Leds). Pour la commande manuelle, nous avons utilisé des pins. Les Leds sont des diodes qui diffèrent de celle de Silicium, or cette dernière a une tension de seuil de 0.6V, par contre la diode LED a une tension de seuil presque le double qui atteint, pour les Leds rouges 1.5V, 2V pour les vertes et les oranges...

Pour le bon fonctionnement et la clarté de la luminosité des diodes (Leds), on a assuré la chute nécessaire de tension à travers des résistances de protection de (330Ω) comme le montre la figure ci-dessous.

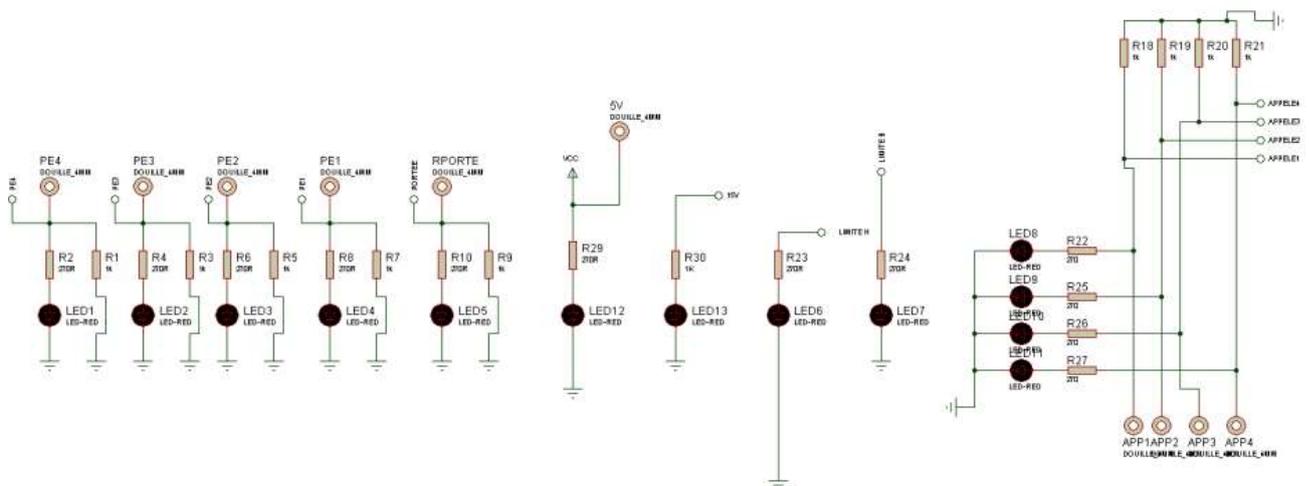
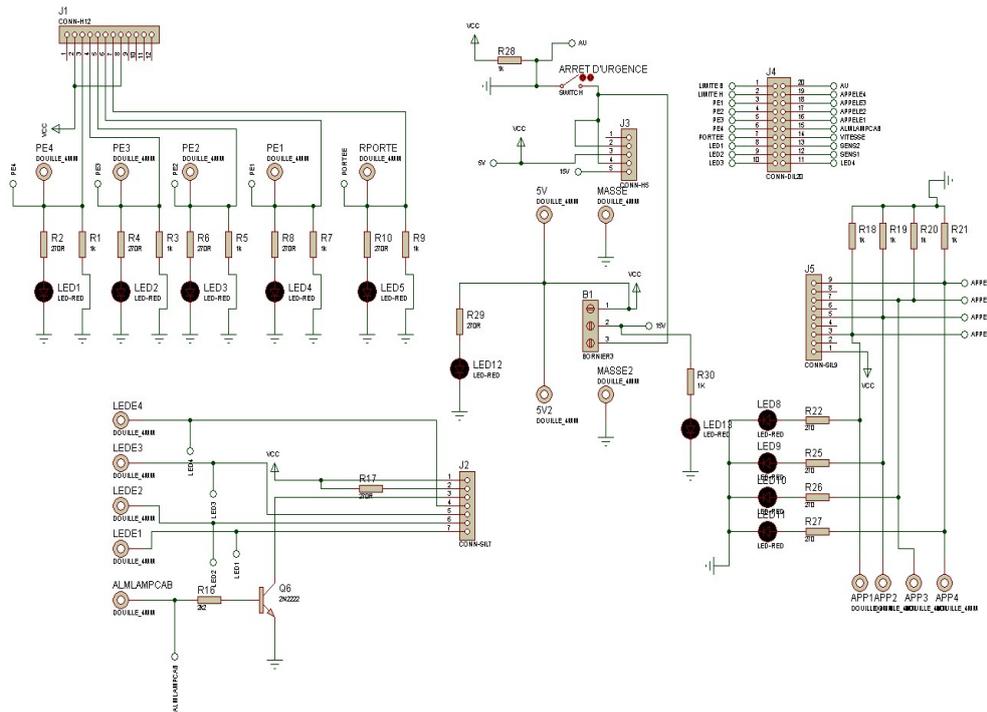


Figure 2.11 : Circuit d'acquisition de carte de commande câblée

2-5-4 Port d'entrées/Sorties

Ces ports d'entrées/Sorties sont représentés dans la **Figure 2.12** sous la forme de connecteurs (J1 : 12 broches, J2 : 7 broches, J3 : 5 broches et J5 : 9 broches) pour établir la connexion entre l'ascenseur et notre carte (la carte de commande câblée) et J4 (20 broches) pour la connexion de notre carte avec les deux autre cartes (carte de commande basée sur le PIC16F877et la carte des boutons poussoirs).



Fi

Figure 2.12: Les ports d'entrées/Sorties de la carte de commande câblée

2-6 Etude de la première Carte de commande à base de microcontrôleur PIC16F877

Cette carte est connectée avec la carte d'entrée/sortie à partir du connecteur de 20 broches, qui sont liés avec le microcontrôleur. Cette carte a pour but de commander l'ascenseur à base de microcontrôleur 16F877.

2-6-1 Schéma synoptique

La **Figure 2.13** représente le schéma synoptique de la première carte de commande à base du PIC 16F877.

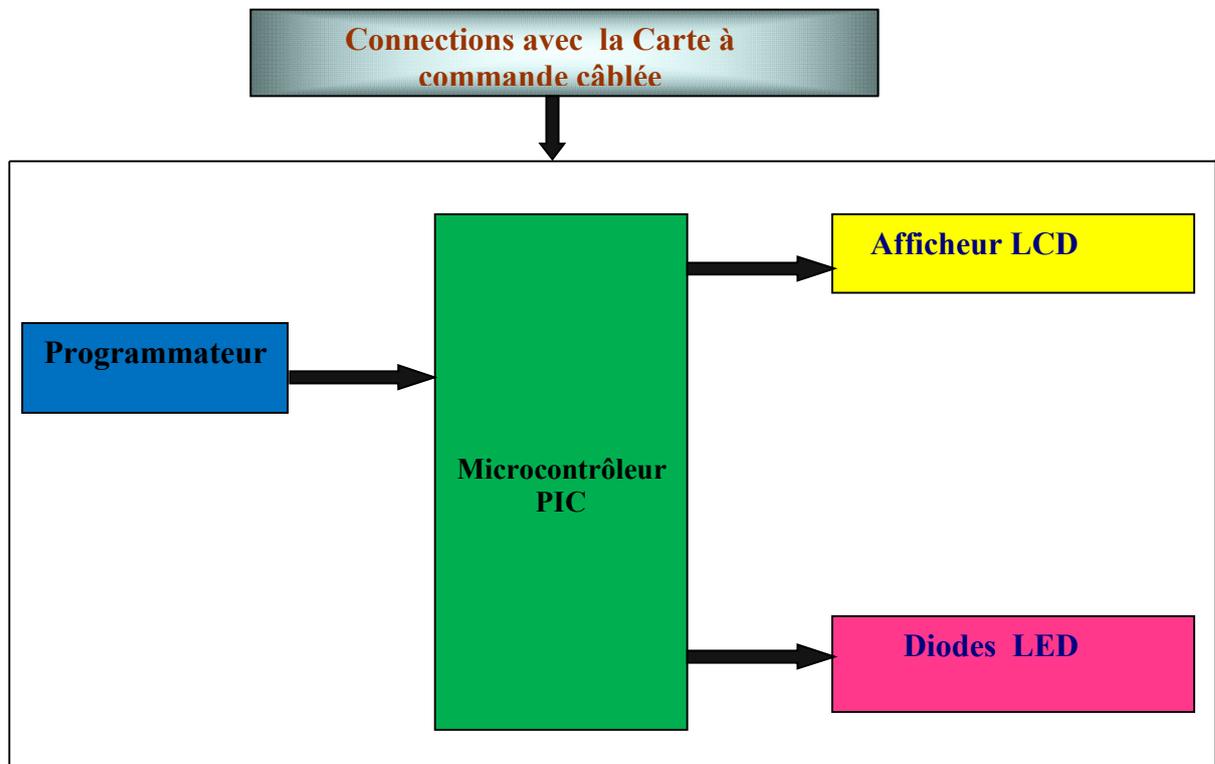


Figure 2.13 : Schéma synoptique de la première carte de commande à base du PIC 16F877

2-6-1-1 Diodes Leds

On a utilisé des diodes (Leds) rouges de potentiel de 1.5V et vertes de potentiel de 2V dans cette partie, pour afficher les états logiques de la porte et l'allumage de la cabine de l'ascenseur.

2-6-1-2 Afficheur LCD

➤ Présentation :

Dans notre projet, on a utilisé un afficheur LCD 2*16 digits à logique intégrée que l'on trouve aujourd'hui très facilement sur le marché. C'est un moyen d'affichage des informations. Et comme il ne fait aucune interprétation des codes de commandes des afficheurs, il est compatible avec tous les modèles existants, de 1 ou 2 lignes de 16 ou 20 caractères. Il existe deux interfaces normalisées : une version « parallèle » et une autre « série » or dans cette étude nous avons utilisé la version parallèle qui est composée de 2 lignes et de 16 caractères [5].

➤ Une interface normalisée :

Tous les afficheurs LCD à logique intégrée utilisent la même interface sans qu'aucune norme rédigée pour cela. Cette interface est de type parallèle et son brochage avec le nom de ses signaux est indiqué dans le **Tableau 2.2** suivant :

Tableau 2.2 : brochage <<normalisée>> des afficheurs LCD à logique intégrée [5].

N° de pastille	Appellation	Fonction
1	Vss	Masse
2	Vdd	Alimentation +5V
3	V0	Contraste afficheur
4	Rs	Sélection commande/donnée
5	R/W	Lecture/écriture
6	E	Validation des données
7	D0	Donnée D0 [poids faible]
8	D1	Donnée D1
9	D2	Donnée D2
10	D3	Donnée D3
11	D4	Donnée D4
12	D5	Donnée D5
13	D6	Donnée D6
14	D7	Donnée D7 [poids fort]

 **Broche 1** : masse.

 **Broche 2** : Vcc.

 **Broche 3** : luminosité

 **Broche 4 RS (Registre Select)** : sert à dire au module dans quel registre il doit écrire les données présentes sur les broches des données.

Si RS = 0 ; le module sait que c'est une instruction et va donc l'écrire dans le registre d'instruction.

Si RS = 1 ; le module sait que c'est un code caractère et va donc l'écrire dans le registre de donnée.

- ✚ **Broche 5 R/W (Read /WRITE) :** sert à dire au module, si les broches des données sont en <<entrée>> ou en <<sortie>>, autrement dit si les données sont <<écrites>> vers le module ou <<lues>> depuis le module.
- ✚ **Broche 6 E (Enable):** sert à valider les données ou les instructions écrites sur les broches des données du module. La validation se fait sur le front descendant
- ✚ **Broche 7 à 14 :** utilisées pour le transfert des données ou des instructions. Le transfert peut se faire sur 8 bits, toutes les broches sont alors utilisées, ou sur 4 bits, dans ce cas, seules les broches 11 à 14 sont utilisées.

➤ Connexion de l’afficheur sur la carte :

La **Figure 2.14** nous montre la connexion de l’afficheur avec le PIC16F877.

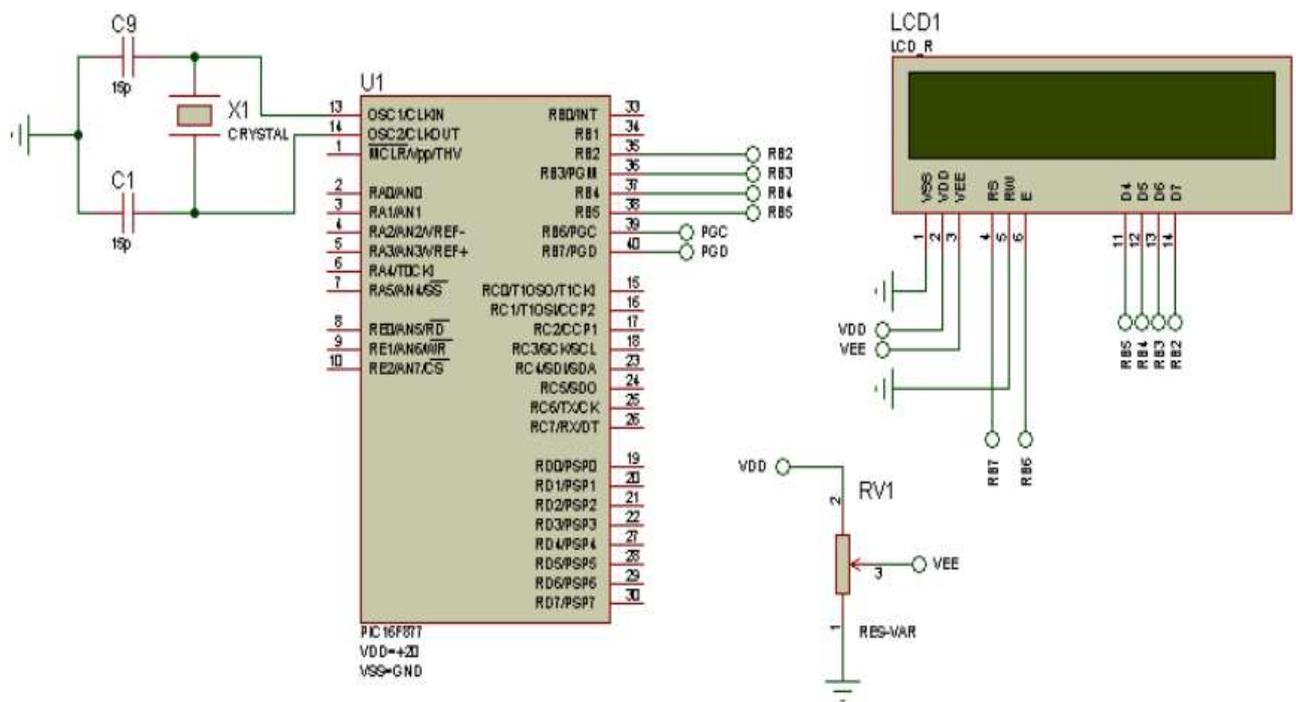


Figure 2.14 : Connexion de l’afficheur sur la carte de commande à base du PIC16F877

2-6-1-3 Programmeur

Notre carte est à base de microcontrôleur PIC, et pour qu’elle soit fonctionnelle, il faut programmer le microcontrôleur PIC.

On peut réaliser un programmeur intégré dans la carte de développement, mais pour faciliter la tâche, On a utilisé un programmeur isolé. Comme le montre la **Figure 2.15** suivante:

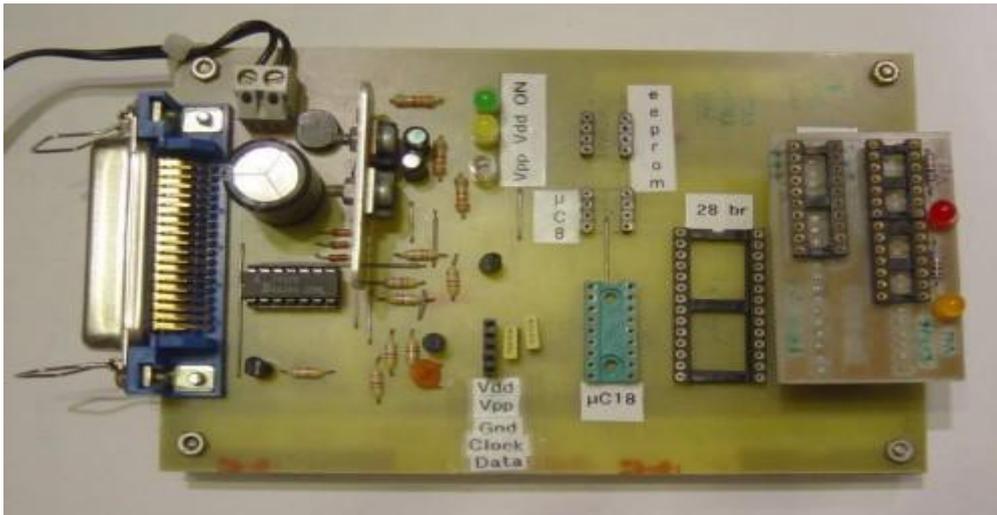


Figure 2.15 : Carte de programmeur

2-6-2 Schéma électrique

La Figure 2.16 représente le schéma électrique de la première carte de commande à base du PIC16F877. Elle est utilisée pour afficher les états logiques de la porte, l’allumage de la cabine, l’étage désiré et le message d’accueil à travers l’afficheur LCD et les Leds rouges.

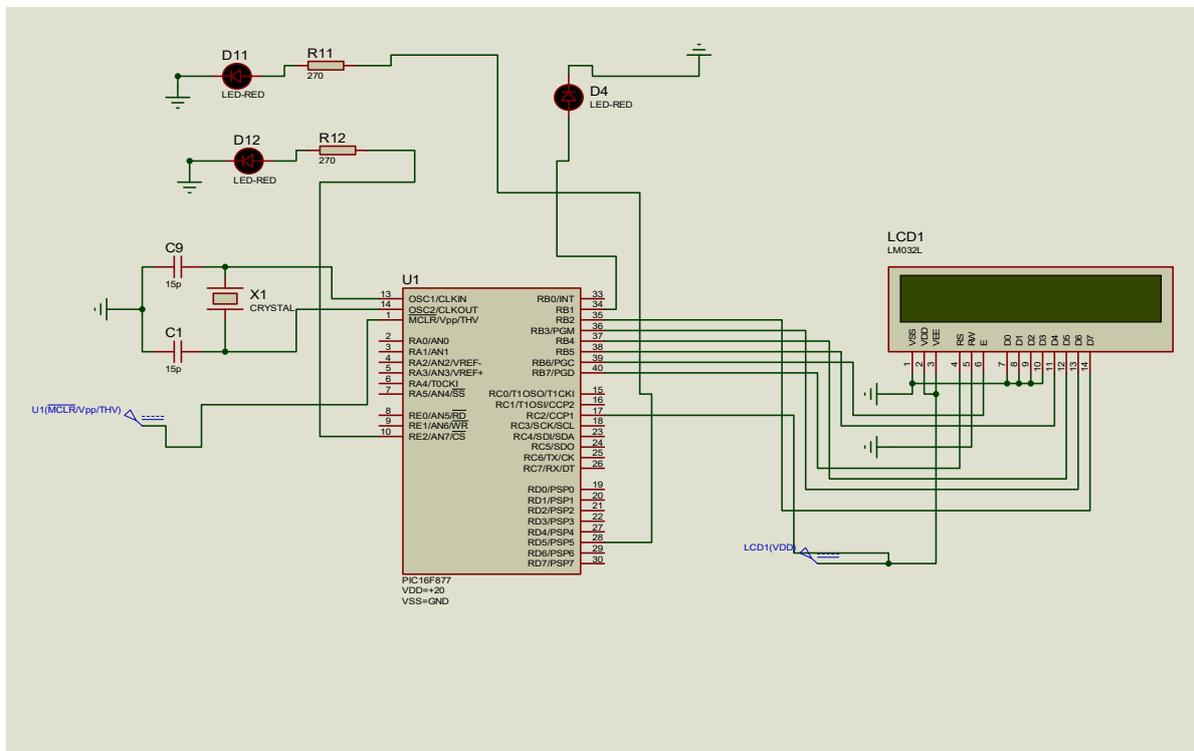


Figure 2.16 : schéma électrique de la première carte de commande de PIC

2-7 Etude de la deuxième Carte de commande à base de microcontrôleur PIC16F877

Cette carte qui fait partie de la carte de commande à base de microcontrôleur PIC16F877 est connectée avec la carte d'entrée/sortie à partir du connecteur de 20 broches. Elle a pour rôle de commander le sens de rotation du moteur à travers les capteurs de position et les boutons d'appels.

2-7-1 Schéma synoptique

La **Figure 2.17** représente le schéma synoptique de la deuxième carte de commande à base du PIC16f877.

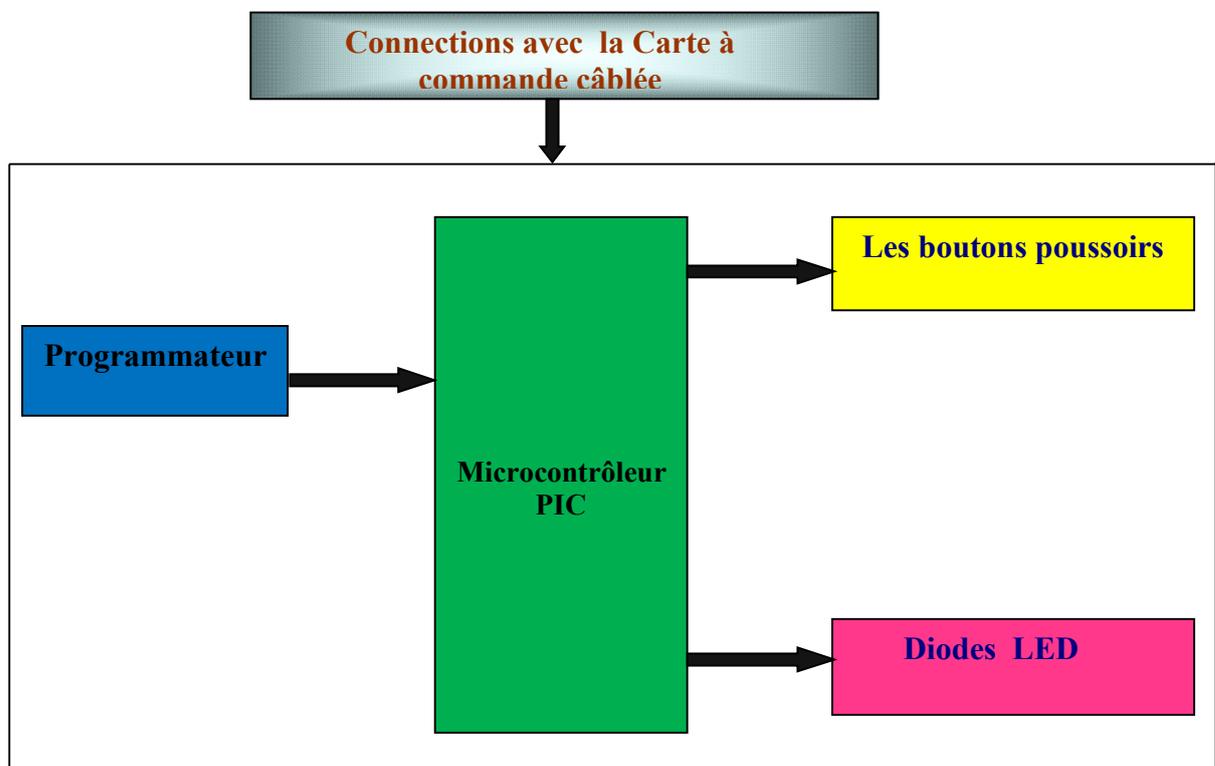


Figure 2.17 : Schéma synoptique de la deuxième carte de commande à base du PIC 16F877

2-7-1-1 Diodes Leds

Cette fois ci, les diodes (Leds) rouges sont utilisées pour afficher les états logiques suivants :

- ✚ Les quatre étages de l'ascenseur (1^{ère} étage, 2^{ème} étage, 3^{ème} étage et 4^{ème} étage).
- ✚ Les deux sens de rotation du moteur (sens1 et sens2).
- ✚ La vitesse du moteur, etc.....

2-7-1-2 Les boutons poussoirs

On a utilisé les boutons poussoirs dans cette partie pour remplacer :

- ✚ les capteurs de position de la cabine dans chaque étage de l’ascenseur.
- ✚ Les boutons d’appel de chaque étage de l’ascenseur.

2-7-2 Schéma électrique

La **Figure 2.18** représente le schéma électrique de la deuxième carte de commande à base du PIC16F877 implémenté sur le logiciel Proteus. Cette carte est utilisée pour afficher l’état logique de chaque étage (l’étage désiré), les deux sens de rotation du moteur et la vitesse à travers les Leds rouges. Cette carte contient aussi les boutons poussoirs qui représentent les capteurs de position avec les boutons d’appels de chaque étage.

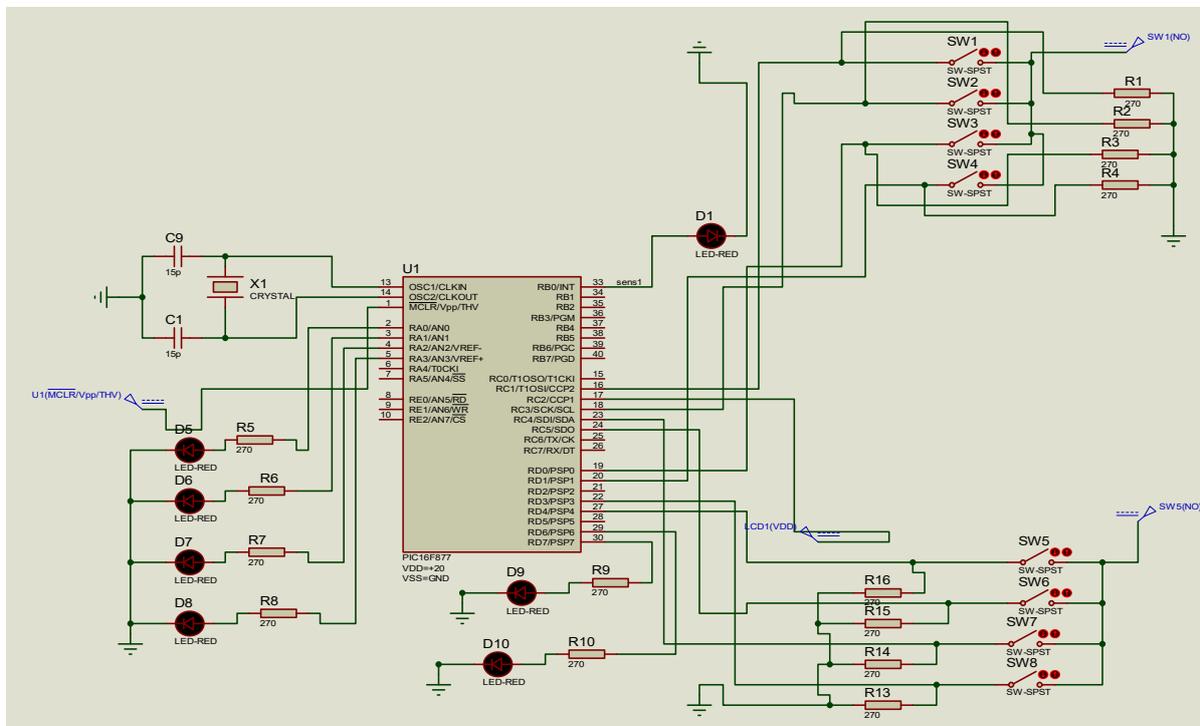


Figure 2.18 : schéma électrique de la deuxième carte de commande du PIC16F877

2-8 Conclusion

Dans ce chapitre, on a présenté les différents entrées/sorties de notre système. Il est clair qu’il y a plusieurs solutions pour commander l’ascenseur. Parmi ces solutions on peut en citer deux : la commande câblée et la commande par microcontrôleur. Ces derniers feront l’objet d’étude du prochain chapitre.

3-1 Introduction

Dans le chapitre précédent, on a étudié les caractéristiques du microcontrôleur PIC 16F877 qui est l'élément le plus intéressant pour nos cartes. Dans ce chapitre, nous allons présenter l'étude de réalisation et la simulation de notre montage.

La simulation et la programmation de nos cartes de commandes représentent le niveau le plus haut dans la hiérarchie de la conception de ce système. La programmation d'un microcontrôleur, définit le principe de fonctionnement de celui-ci. Les cartes électroniques exécutent les instructions décidées par le programme, c'est pour cette raison précise, que l'automatisation s'appuie sur des outils informatiques aussi bien dans les phases de conception, de programmation, de modélisation (géométrique, cinématique, dynamique, contraintes, etc.), que de simulation. Ces précieux logiciels sont des aides indispensables pour la maîtrise globale d'un système automatique. A ce jour, ces logiciels sont presque exclusivement réservés à des spécialistes. En particulier, les outils permettant d'aborder les aspects de simulation ou de programmation des microcontrôleurs.

Dans ce chapitre, on va donner aussi une présentation du logiciel Proteus, les différentes cartes réalisées sur Proteus et finalement une présentation du logiciel Mikropascal.

3-2 Présentation du logiciel Proteus

Proteus est un outil professionnel de saisie de schémas et de conception de circuits imprimés avec placement automatique, routage et rapports. Il est composé principalement de deux outils ISIS et ARES [6].

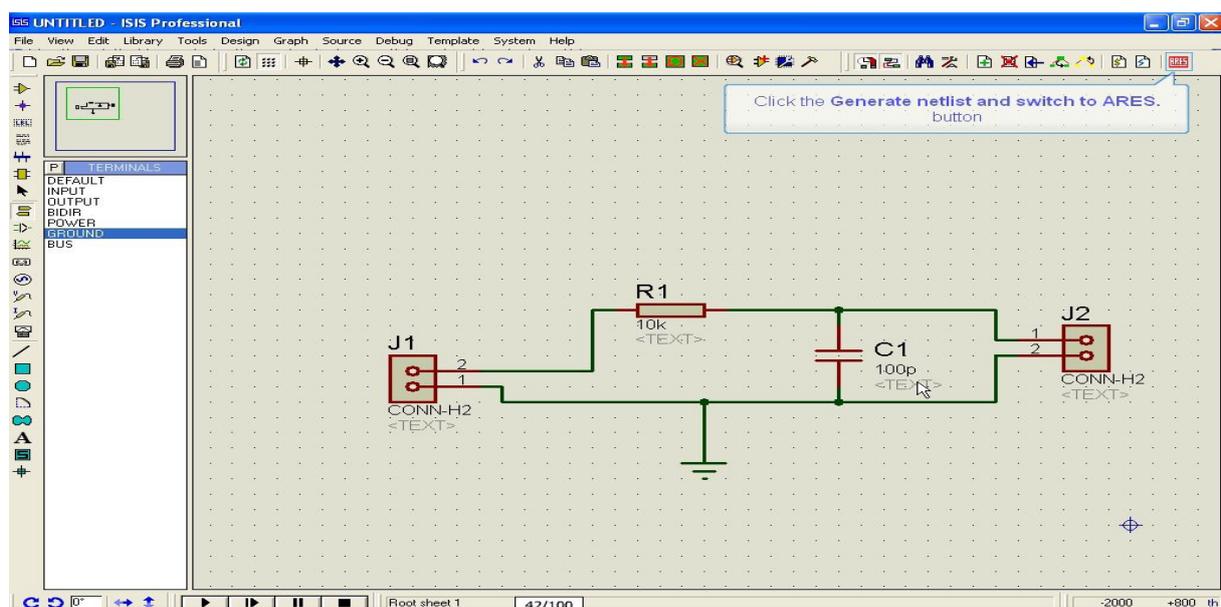


Figure 3.1 : la fenêtre ISIS Professional

ISIS

Comme le montre la **Figure 3.1**, ISIS est beaucoup plus qu'un logiciel de saisie de schéma classique. Au cœur du système PROTEUS, ISIS est conçu pour répondre parfaitement aux besoins des deux fonctions d'un schéma :

- La saisie rapide de projets très complexes destinés aux simulations et aux conceptions de cartes.
- La création de très beaux schémas destinés aux publications.

Cet excellent logiciel nous a permis de faire la simulation de la partie électronique de notre carte de commande. Il comporte une bibliothèque riche en microcontrôleurs qu'on peut programmer, et des composants électroniques de tout genre.

Il nous a permis de voir si nos cartes fonctionnent correctement ou pas sans passer à la réalisation donc faire la différence facilement entre la panne de réalisation et l'erreur de programmation.

ARES

Le module ARES de PROTEUS permet de dessiner manuellement ou automatiquement des circuits imprimés comme le montre la **Figure 3.2**.

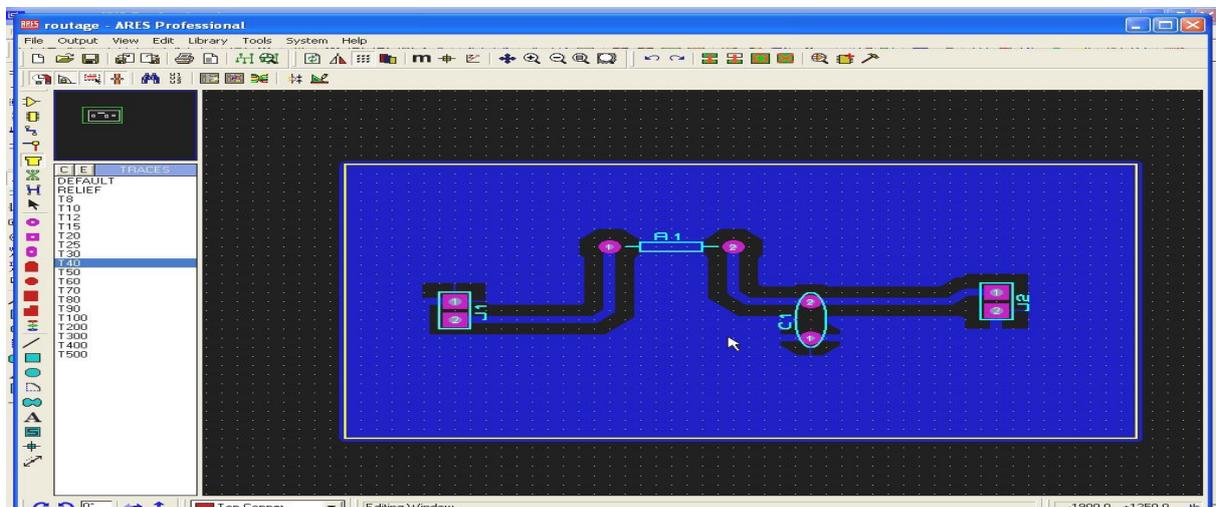


Figure 3.2 : la fenêtre ARES Professional

Note : pour plus d'informations sur l'utilisation du logiciel Proteus voir l'annexe 3.

3-3 Implémentation des différentes cartes réalisées sur Proteus

Les étapes nécessaires permettant de voir un programme s'exécuter sur un PIC sont :

-  Saisir le schéma électronique à réaliser sur un logiciel de CAO (dans notre cas ISIS).

- ✚ Produire le circuit de routage de la carte électronique à réaliser sur le logiciel ARES.
- ✚ Produire la carte électronique, puis implanter tous les composants.
- ✚ Ecrire un programme en langage Mikropascal et le sauvegarder avec l'extension **.ppas**
- ✚ Compiler ce programme. Le résultat est un fichier exécutable avec l'extension **.hex** contenant une suite d'instructions compréhensibles par le pic.
- ✚ Transplanter le fichier **.hex** dans la mémoire programme du PIC (mémoire flash) à l'aide d'un programmeur adéquat.
- ✚ Mettre le PIC dans son montage final, mettre sous tension et vérifier le fonctionnement.

On va commencer par les trois cartes réalisées sur PROTEUS et qui sont les suivantes :

3-3-1 Carte de commande câblée

La carte de commande câblée comme la montre la **Figure 3.3** est basée essentiellement sur le circuit de puissance composé de cinq relais, le circuit d'acquisition sous forme des LED rouges et finalement des connecteurs de (20, 9,7 broches, etc.....) qui représentent les ports Entrée /Sortie de notre système.

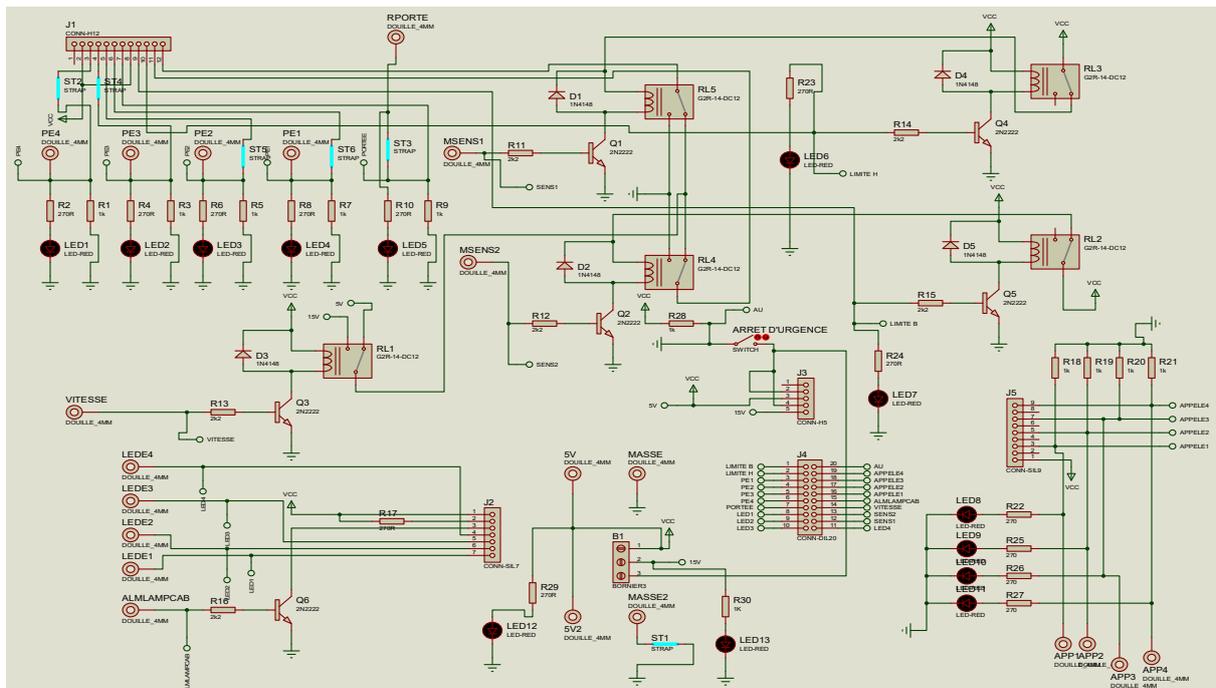


Figure 3.3 : la carte de commande câblée

3-3-2 carte de commande à base du microcontrôleur PIC16F877

Cette carte comme nous montre la **Figure 3.4** est composée de deux cartes de commande : la première carte contient le microcontrôleur PIC connecté avec l’afficheur LCD et trois LED rouges pour le (son, test porte, allumage de la cabine) et la deuxième carte contient le PIC connecté avec les boutons poussoirs qui représentent les capteurs de position de chaque étage, les boutons d’appel de chaque étage aussi et des LED rouges pour afficher les états logiques suivants :

- 🔌 Indicateur d’étage, on a 4 étages donc 4 indicateurs.
- 🔌 Les deux sens de rotation du moteur (Sens 1 et Sens 2).
- 🔌 La vitesse du moteur.
- 🔌 Le dépassement de la cabine : limite haut (LIMITE H) et limite bas (LIMITE B).
- 🔌 L’arrêt d’urgence.

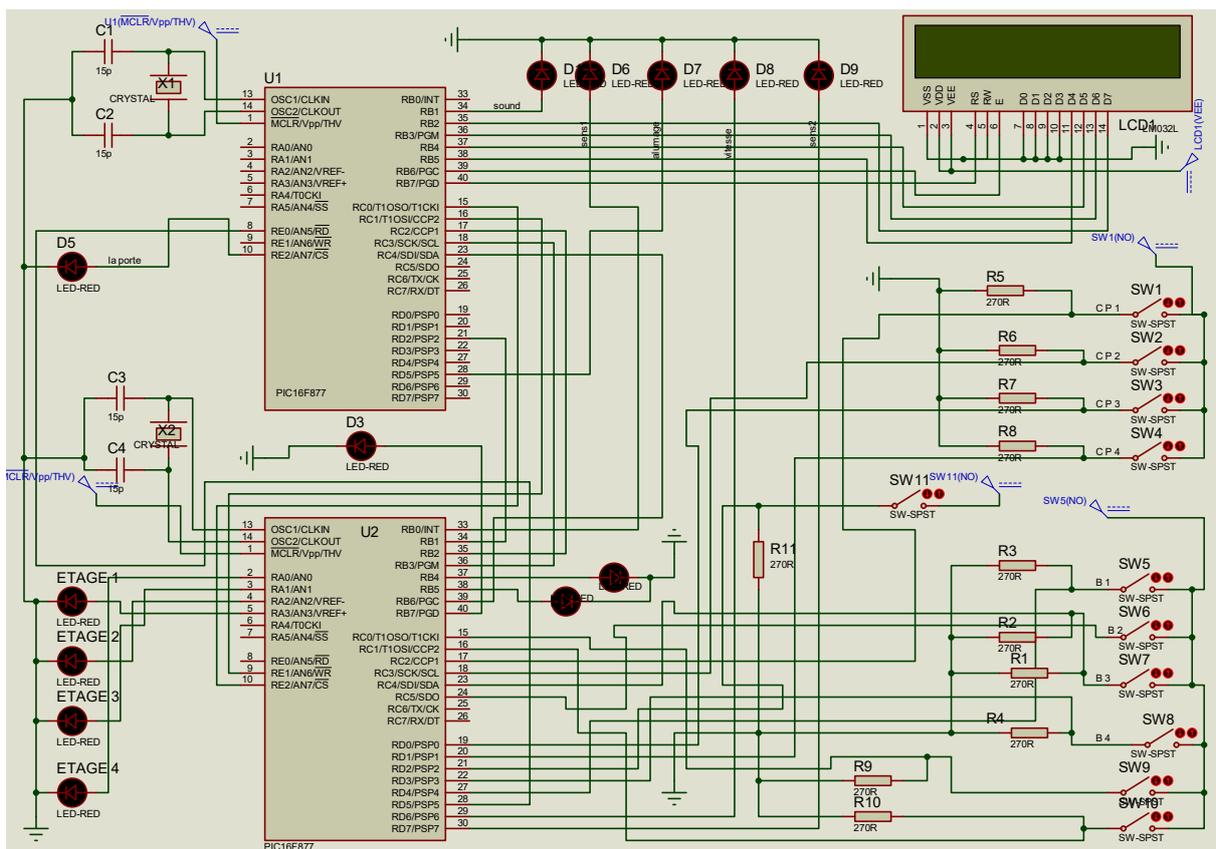


Figure 3.4 : la carte de commande basée sur le microcontrôleur PIC16F877

3-3-3 Carte des boutons poussoirs

Le schéma électrique de la **Figure 3.5** représente la carte des boutons poussoirs qui contient :

- ✚ 4 boutons poussoirs pour remplacer les 4 capteurs de position d'étage.
- ✚ 4 boutons poussoirs pour les 4 boutons d'appel.
- ✚ 3 boutons poussoirs pour remplacer les capteurs : d'Arrêt d'urgence, limite haut (LIMITE H) et limite bas (LIMITE B).
- ✚ Un connecteur de 18 broches pour la connexion avec les autres cartes.

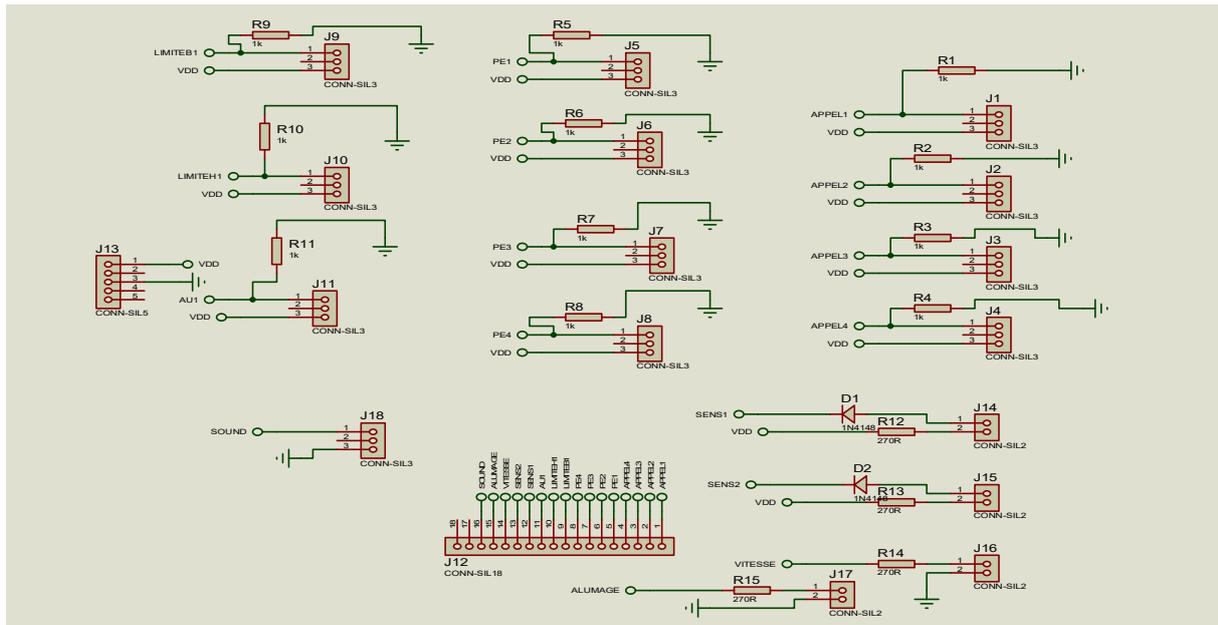


Figure 3.5 : La carte des boutons poussoirs

3-4 Présentation du logiciel Mikropascal

Le « Mikropascal » est un compilateur pour PIC conçu par la société « Mikroelektronika ». Ce compilateur PASCAL nouvelle génération " Micropascal" pour microcontrôleurs PIC bénéficie d'une prise en main très intuitive et d'une ergonomie sans faille. Ces très nombreux outils intégrés (mode simulateur, gestionnaire 7 segments,...) associés à sa capacité à pouvoir gérer la plupart des périphériques rencontrés dans l'industrie (RS, Bus CAN, afficheur LCD et 7 segments...) ont fait de lui un outil de développement incontournable, puissant et riche de dispositifs pour des microcontrôleurs de la famille PIC [7].

Le compilateur Micropascal nous permet de développer rapidement des applications complexes comme suit :

- ✚ Ecrire le code source de pascal en utilisant le rédacteur intégré de code (les aides de code et de paramètre, accentué de syntaxe, correction automatique, etc....)

- ✚ Employer les bibliothèques Micropascal incluses pour accélérer nettement le développement : acquisition de données, mémoire, affichage, conversions.
- ✚ Surveiller la structure de programme, variables, et fonctions dans l'explorateur de code.
- ✚ Inspecter l'écoulement de programme et corriger la logique exécutable avec le programme de mise au point intégré.
- ✚ Obtenir les rapports et les graphiques détaillés : carte de RAM et de ROM, coder les statiques, impression d'assemblage, etc....

3-4-1 Description du logiciel Micropascal

La Figure 3.6 représente une description détaillée de l'interface du logiciel MikroPascal :

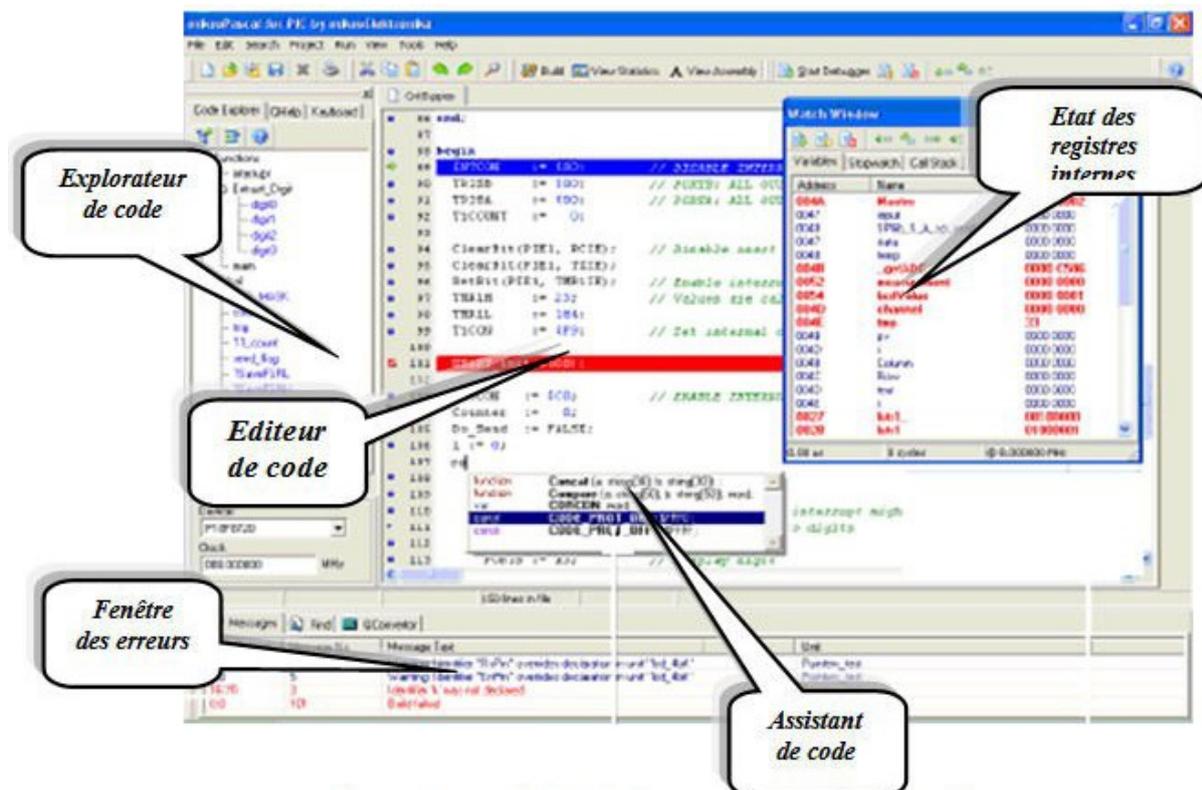


Figure 3.6 : description de l'interface du logiciel MikroPascal [8]

3-4-1-1 La barre d'outils

La barre d'outils contient les instructions suivantes :

-  Nouveau projet
-  Charger un projet existant et changer ses paramètres (ex : horloge, son nom,...)

-  Ouvrir un projet
-  Fermer un programme
-  Sauvegarder les programmes en cours
-  Sauvegarder un fichier existant sous un autre nom.
-  Sauvegarder le programme en cours
-  Annuler la dernière action
-  Rétablir la dernière action
-  Imprimer
-  Une nouvelle page

3-4-1-2 Création d'un projet sur MikrPascal

❖ Première étape

Cliquer sur l'icône « nouveau projet » comme le montre la **Figure 3.7** ou sélectionner le menu « projet » à « new projet ».

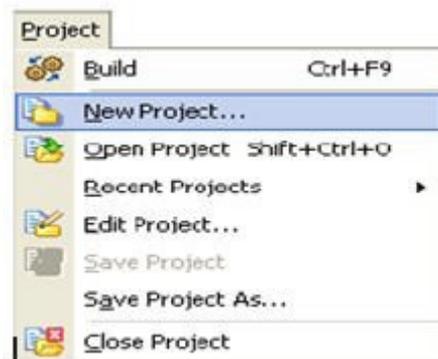


Figure 3.7 : ouvrir un nouveau projet

❖ Deuxième étape

Le Mikropascal organise des applications en projet, se composant d'un fichier projet simple (extension.PPA) et un ou plusieurs (.ppas) fichiers sources. Nous pouvons compiler des fichiers source seulement s'ils font partie d'un projet.

Le fichier projet diffuse les informations suivantes :

-  Nom de projet et description facultative ;

- ✚ Dispositif de cible ;
- ✚ Drapeaux de dispositif (mot de config) ;
- ✚ Horloge de dispositif ;
- ✚ Liste de fichiers source de projet avec des chemins.

La création d'un nouveau projet est simple. Il suffit de remplir la boîte de dialogue (**Figure 3.8**) avec les valeurs désirées (Project name and description, location, devise, clock, config word). Après la configuration de notre projet, on clique sur OK.

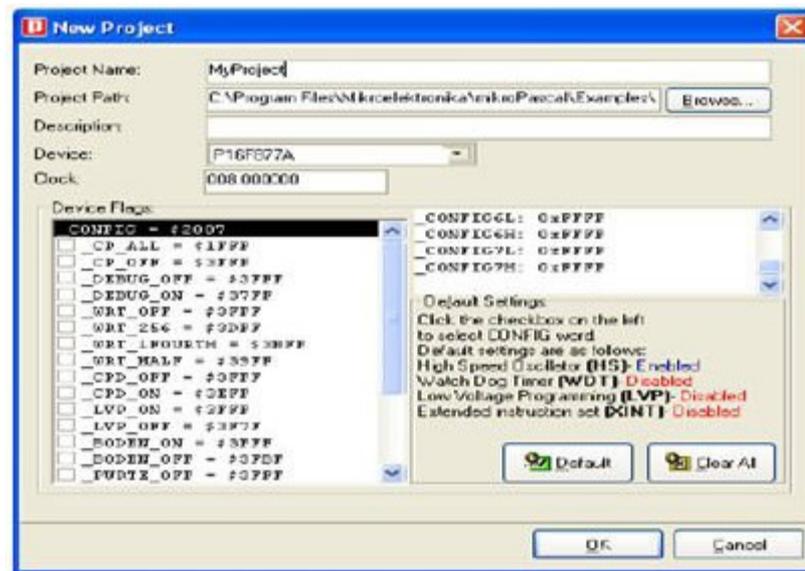


Figure 3.8 : fenêtre de nouveau projet

❖ Troisième étape

Lors de la fermeture de la fenêtre précédente, on peut écrire notre programme sur l'éditeur de code comme suit (**Figure 3.9**) :

```

lcd_demo.ppas
1
2 Program lcd;
3 Begin
4 TRISB :=0 ;
5 While true do
6 Begin
7 PORTB :=$FF ;
8 Delay_ms(1000) ;
9 PORTB :=0 ;
10 Delay_ms(1000) ;
11 end ;
12 end.

```

Figure 3.9 : saisie du programme dans l'éditeur de code.

❖ Quatrième étape

Après l'exécution du programme, on clique sur le menu « Project build » (**Figure 3.10**).

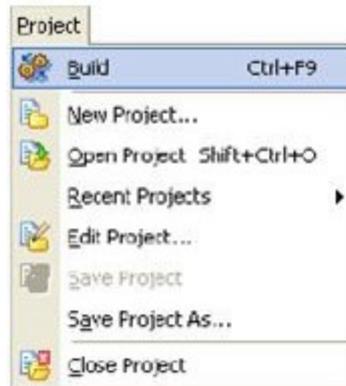


Figure 3.10 : compilation du programme

La barre de progrès nous informera au sujet du statut de compilation. S'il y a des erreurs, on nous l'annoncera dans la fenêtre d'erreur.

Si aucune erreur n'est produite, la fenêtre des erreurs nous informe... (Voir la **Figure 3.11** ci-dessous)

 A screenshot of a 'Messages' window in a software application. The window has a title bar with 'Messages', 'Find', and 'QConverter' icons. Below the title bar is a table with four columns: 'Line/Column', 'Message No.', 'Message Text', and 'Unit'. The table contains three rows of data:

Line/Column	Message No.	Message Text	Unit
0.0	100	Success	
0.0	101	Used ROM: 70 (1%)	Used RAM: 3 (1%)
0.0	102	Free ROM: 8121 (99%)	Free RAM: 395 (99%)

Figure 3.11 : fenêtre d'erreur

3-4-2 Editeur de code

Extrêmement convivial, l'éditeur du « MikroPascale » dispose d'une multitude de fonctions qui permettront aussi bien aux utilisateurs novices comme aux professionnels de travailler très rapidement.

3-4-2-1 L'assistant de code

Ce dernier nous permettra après avoir taper les premières lettres d'une instruction d'ouvrir une fenêtre en tapant CTRL+ ESPACE afin d'obtenir la liste de toutes les instructions commençant par les mêmes lettres. Il nous suffira ensuite de cliquer sur celle qui nous convient pour l'importer directement dans l'éditeur (**Figure 3.12**).

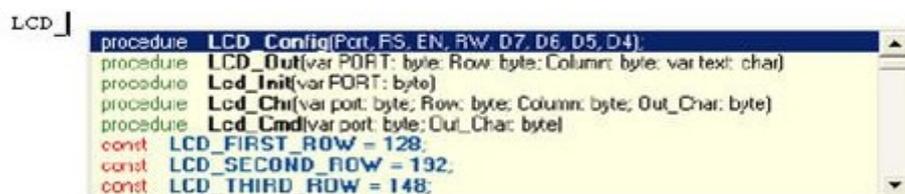


Figure 3.12 : assistant de code

3-4-2-2 L'assistant de paramètres

Ce dernier nous permettra après avoir saisi le début d'une fonction nécessitant des paramètres (**Figure 3.13**), d'afficher la syntaxe de ces paramètres (le premier paramètre à écrire s'affiche en caractères gras). Après avoir saisi ce premier paramètre, le second s'affiche alors à son tour en caractères gras).

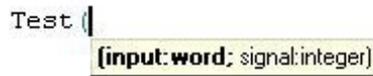


Figure 3.13 : assistant de paramètres.

3-5 Réalisation pratique et tests

On va tester la réalisation pratique de la carte de commande basée sur le microcontrôleur PIC16F877 de la **Figure 3.4** à travers la simulation sur le logiciel Proteus.

- ✚ On lance la simulation en cliquant sur le bouton Play.
- ✚ L'afficheur LCD affiche le message d'accueil suivant : « bien venue, sélectionnez la position de la cabine et choisissez un étage ».comme nous montre la **Figure 3.14**.

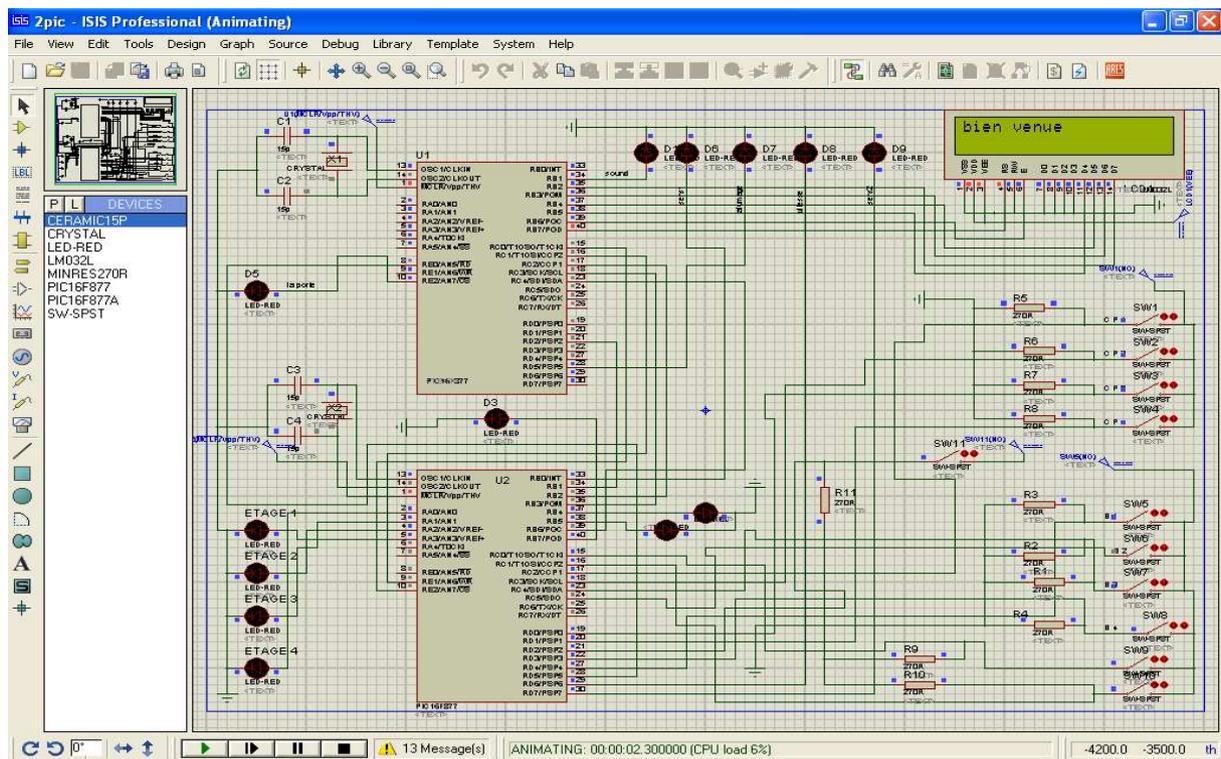


Figure 3.14 : « message d'accueil »

- ✚ On va choisir la position de la cabine à travers les quatre Switchs de position. Donc nous avons 4 possibilités suivant les 4 étages de notre ascenseur. Au même temps, on va choisir aussi un étage (l'étage désiré) à l'aide des 4 Switchs qui représentent les

boutons d'appel de chaque étage .Dans notre exemple de la **Figure 3.15** on a fait le cas suivant :

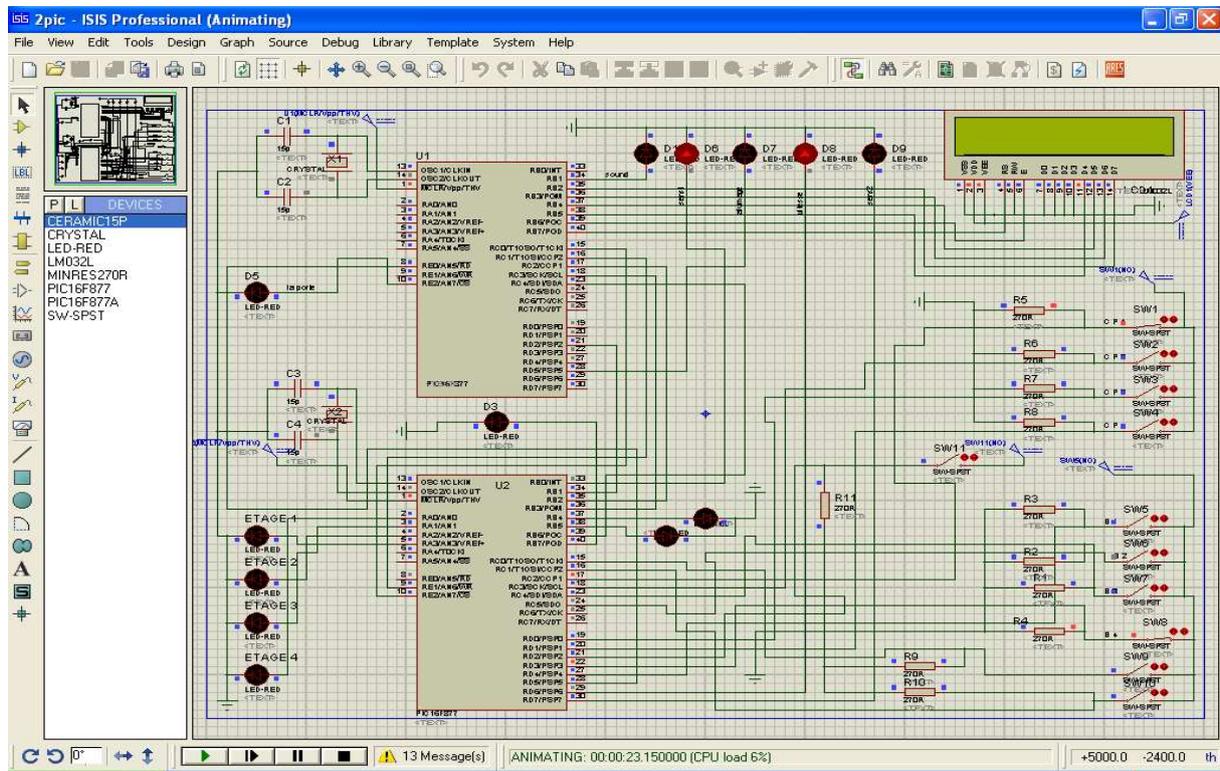


Figure 3.15 : première étape « message d'accueil »

Dans cet exemple la cabine se trouve à l'étage 1 car on a appuyé sur Switch 1 de la position de la cabine et on a choisi l'étage 4 à travers son bouton d'appel.

- Les deux LED (sens1 de la rotation du moteur et la vitesse du moteur) s'allument pendant 20 secondes cela veut dire que l'ascenseur est monté de l'étage1 jusqu'à l'étage4.
- En suite la LED de l'étage 4 s'allume et l'afficheur LCD affiche le message suivant « étage 4» comme le montre la **Figure 3.16** pendant 5 secondes ce qui veut dire que l'ascenseur est arrivée à l'étage 4.
- Après cette étape, il faut ouvrir la porte donc la LED de la porte s'allume et l'afficheur affiche le message suivant : « porte ouverte » comme le montre la **Figure 3.17** pendant 5 secondes en suite la LED doit être éteinte et l'afficheur affiche le message suivant : « porte fermée » comme le montre la **Figure 3.18**.
- Finalement, la LED d'allumage de la cabine va s'allumer après la fermeture de la porte comme le montre la **Figure 3.19**.

Ce cycle se répète dans les cas où l'ascenseur est en état de montée (de 1 à 2, 2 à 3, 3 à 4, 1 à 4, etc.....) avec une durée du temps précise dans chaque cas .Dans les cas où l'ascenseur est en état de descente (de 4 à 1, 4 à 3, 4 à 2, 3 à 1, 3 à 2, etc.....) la LED du sens 2 de rotation du moteur s'allume toute seule sans la LED de la vitesse pour consommer l'énergie du moteur en suite les autres étapes(2,3,4 et 5) se répètent comme l'exemple précédent.

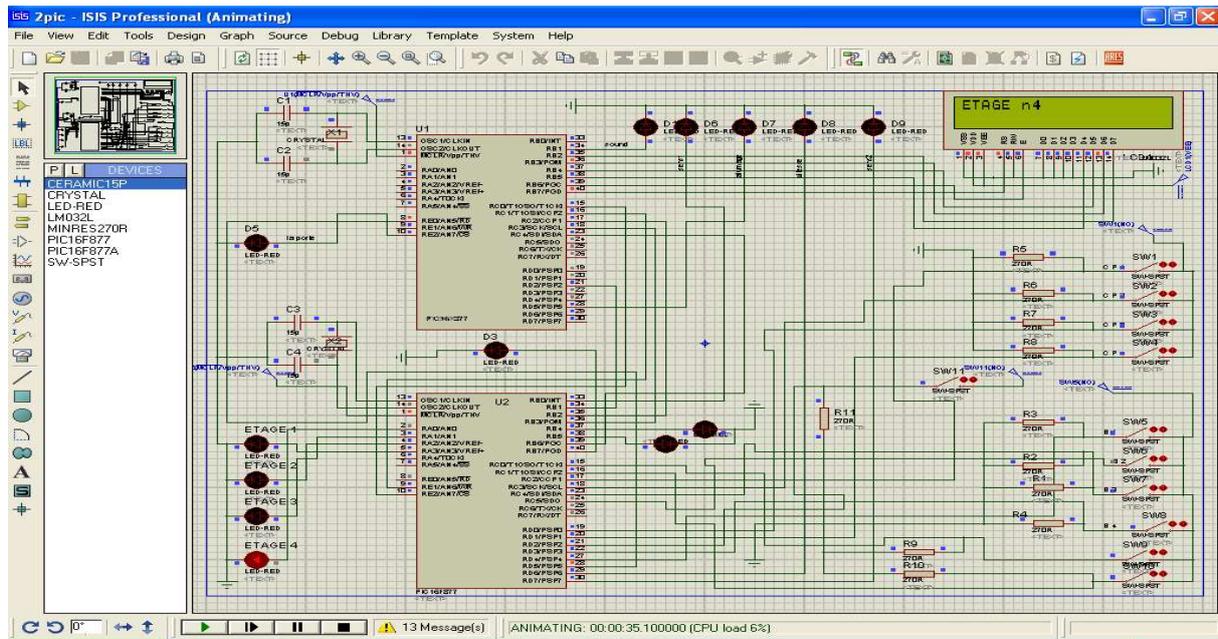


Figure 3.16 :deuxième étape « l'arrivée à l'étage désiré »

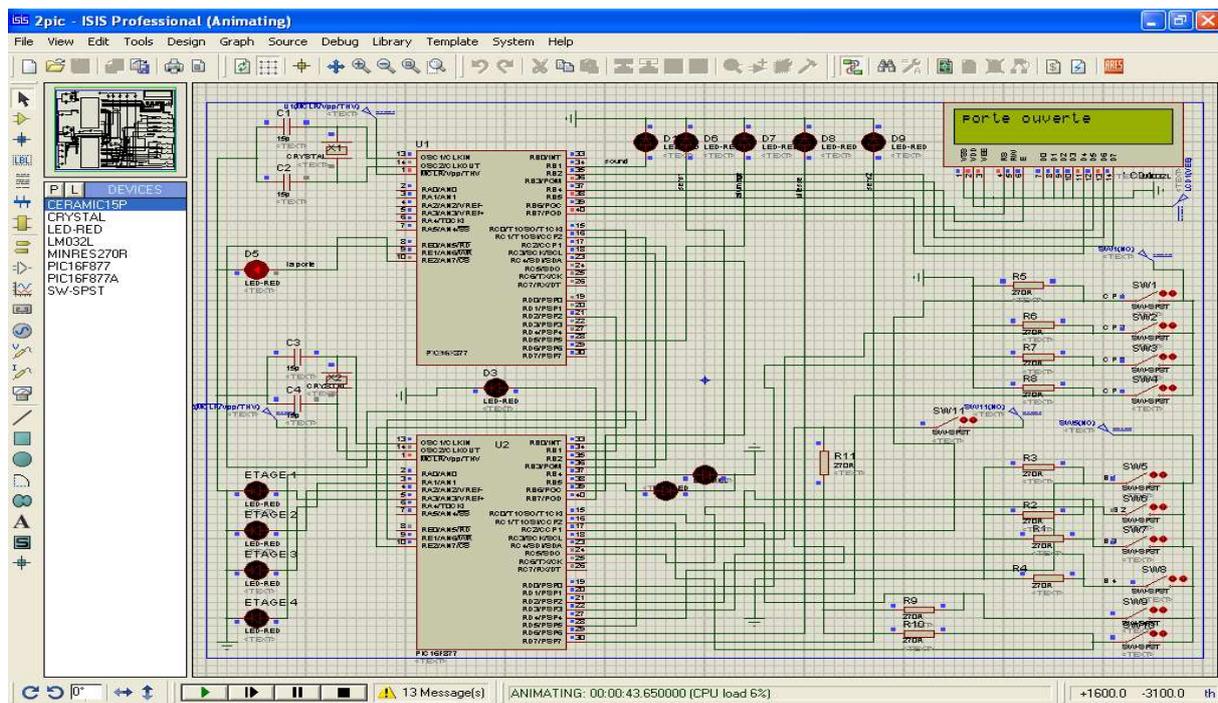


Figure 3.17 :troisième étape « ouvrir la porte »

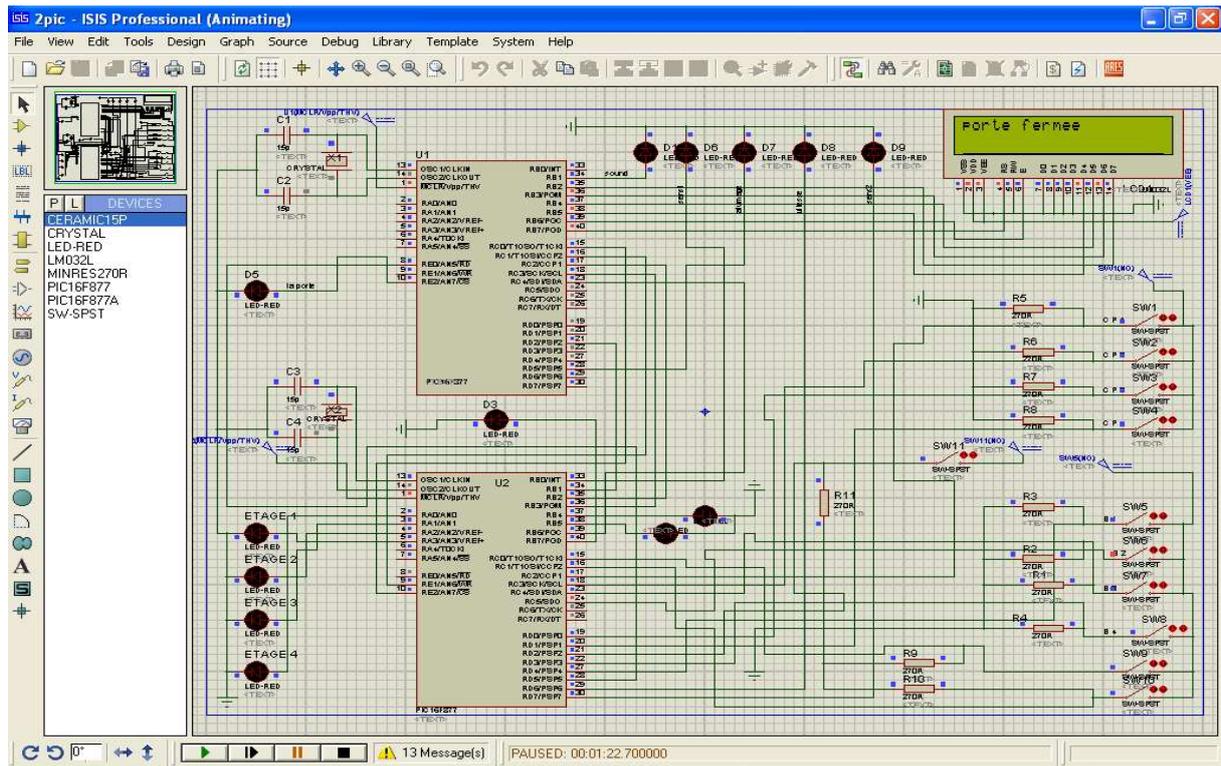


Figure 3.18 :quatrième étape « fermeture de la porte »

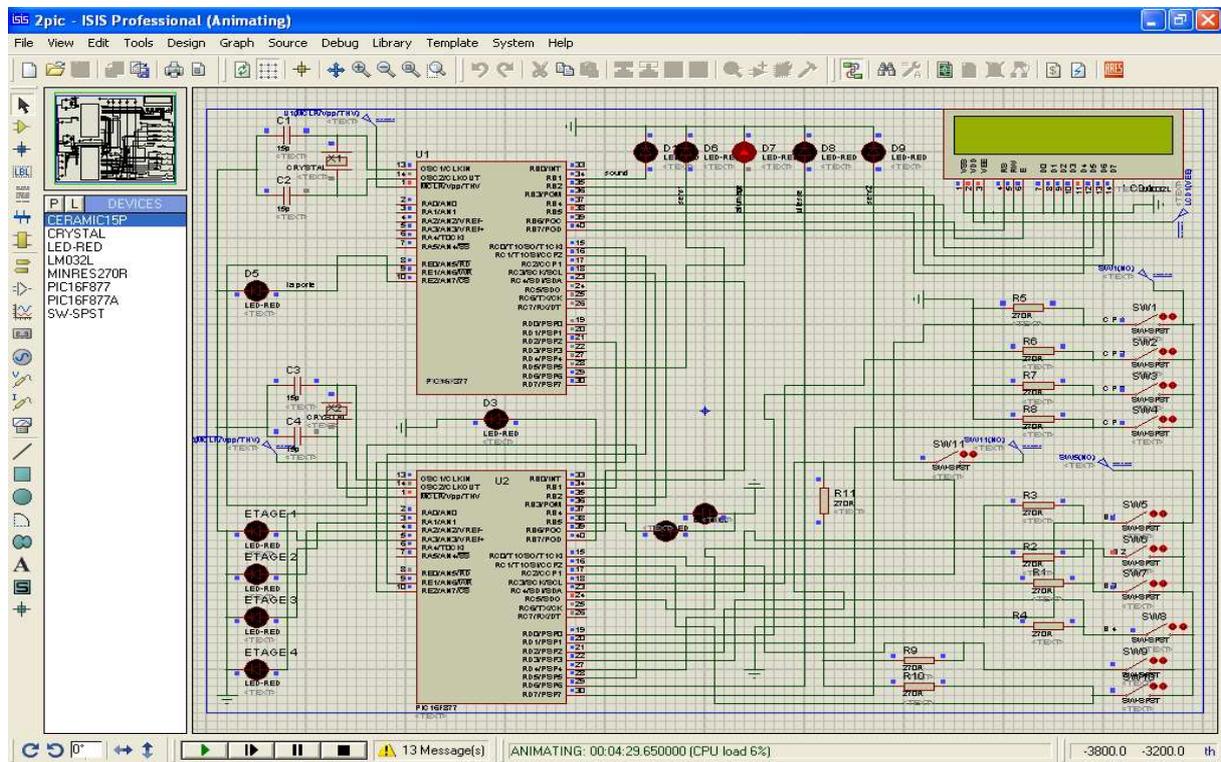


Figure 3.19 :cinquième étape « allumage de la cabine »

 L'arrêt d'urgence :

Si nous avons cliqué sur un bouton poussoir parmi les boutons suivants : « Arrêt d'urgence », « LIMITE H » ou bien « LIMITE B » nous aurons la **Figure 3.20** donc : une LED s'allume pendant un certain temps et l'afficheur LCD affiche le message « ARRET URGENT » avec une alerte dans le bit son (SOUND).

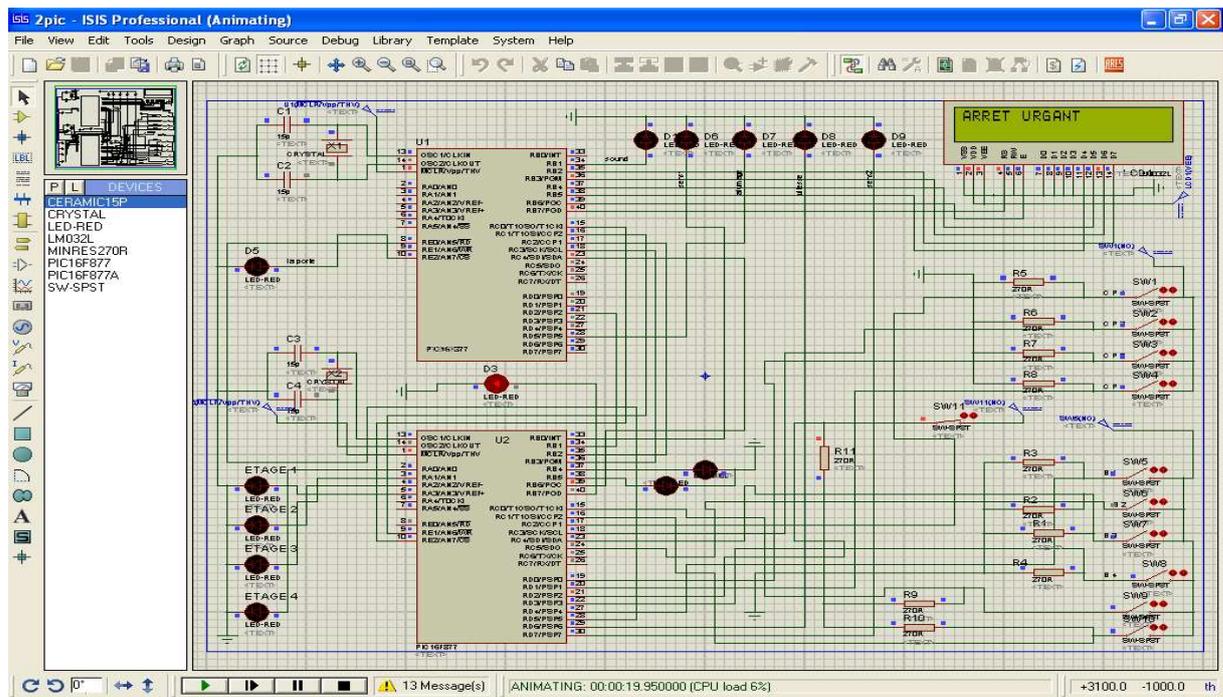


Figure 3.20 :L'arrêt d'urgence

Nous sommes ensuite passés à la réalisation pratique de notre montage et nous avons bien constaté réellement le test de toutes les opérations décrites précédemment.

Note : l'annexe 3 contient les images réelles des différentes cartes réalisées avec les circuits imprimés correspondants.

3-6 Conclusion

Dans ce chapitre nous avons étudié la simulation et la réalisation des cartes de commande avec la présentation des différents blocs et comparer entre elles :

- La commande câblée est simple à l'utilisation et moins chère, mais nous ne pouvons pas y faire des modifications, de même elle présente un encombrement des fils.
- La commande à base de microcontrôleur PIC 16F877 est de faible consommation et présente un circuit imprimé peu complexe avec un coût réduit.

Nomenclature

Carte d'alimentation :

- ✓ Un transformateur (AC220V, 50Hz) à point milieu (12V, 0V, -12V) qui délivre un courant de 1000 mA.
- ✓ Un pont de diode (2W005G) pour le redressement.
- ✓ Les capacités de filtrage et de découplage qui sont :
C1=C2=C3=C4=C5=C6=C9=C10= 100uF, C7=C8= 220uF
- ✓ Les régulateurs de tension : U1(7805), U2(7812), U3(7912), U4(7815).

Carte de commande câblée :

- ✓ 5 relais (RL1, RL2, RL3, RL4, RL5) de type (G2R-14-DC12).
- ✓ Les résistances de 1K Ω : (R1,R2,R5,R7,R9,R18,R19,R20,R21,R28,R30).
- ✓ Les résistances de 270 Ω :
(R2,R4,R6,R8,R10,R17,R22,R23,R24,R25,R26,R27,R29).
- ✓ Les résistances de 2K Ω : (R11, R13, R14, R15, R16).
- ✓ Les diodes de type (1N4148) qui sont : (D1, D2, D3, D4, D5).
- ✓ Les transistors bipolaires de type (2N222) qui sont (Q1, Q2, Q3, Q4, Q5, Q6).
- ✓ 13 LED rouge.
- ✓ 5 Connecteurs : J1=CONN-H12, J2= CONN-H7, J3= CONN-H5, J4=CONN-DIL20, J5=CONN-H12.
- ✓ 21 Pins pour la commande manuelle.
- ✓ 1 Switch « arrêt d'urgence ».
- ✓ 1 Bornier3 (B1).

Carte de commande à base du microcontrôleur PIC16F877 :

- ✓ Un afficheur LCD 2*16 bits.
- ✓ 2 Microcontrôleurs de type PIC 16F877.
- ✓ 4 Capacités de 15pF (C1, C2, C3, C4).
- ✓ 2 Quartz de 4 MHz.
- ✓ 13 LED rouges.

- ✓ 11 Résistances de 270Ω :(R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11).
- ✓ Une résistance variable de $2K\Omega$.
- ✓ 11 Boutons poussoirs.
- ✓ 1 Bornier2.
- ✓ 1 Connecteur de type (CONN-DIP30).

Liste des abréviations

PIC	: Peripheral Interface Controller
RISC	: Reduced Instruction Set Computer
CPU	: Central Processing Unit
RAM	: Random Access Memory
EEPROM	: Electrically Erasable Programmable Read Only Memory
PROM	: Programmable Read Only Memory
OTPROM	: One Time Programmable Read Only Memory
EPROM	: Erasable Programmable Read Only Memory
UVPROM	: Ultra Violet Programmable Read Only Memory
MIPS	: Million Instruction Per Second
I/O	: Input/Output
A/N	: Analogique/Numérique
SPI	: Serial Peripheral Interface
USART	: Universal Asynchronous Receiver Transmitter
E/S	: Entrée/Sortie
TTL	: Transistor-Transistor Logic
SFR	: Special Function Register
RC	: Résistance Capacité
VCC	: Voltage Common- Collector
VDD	: Voltage Drain- Drain
Ox	: Oscillation
MCLR	: Mater CLEAR
LED	: Light Emitting Diode

M/A	: Marche/Arrêt
AFF	: Afficheur
LCD	: Liquid Cristal Display
Cmd	: Commande
D	: Donner des ordres
App	: Appel
Pos	: Position
M	: Marche
GND	: Ground
Ve	: Tension d'entrée
Vs	: Tension de sortie
RL	: Relais
R/W	: Read/Write
RS	: Regitre Select
E	: Enable
ISIS	: Intelligent Schematic Input System
CAO	: Conception Assisté par Ordinateur
CAN	: Convertisseur Analogique Numérique
PPI	: Programmable Peripheral Interface

Liste des figures et tableaux

Liste des figures :

Figure 1.1 : Architecture interne de pic.....	5
Figure 1.2 : photos réelle d'un PIC16F877.....	6
Figure 1.3 : structure générale d'un pic 16F877.....	8
Figure 1.4 : Brochage du circuit d'oscillation PIC16F877.....	9
Figure 2.1 : modélisation de l'ascenseur.....	11
Figure 2.2 : présentation technique de l'ascenseur.....	12
Figure 2.3 : Schéma fonctionnel de l'ascenseur.....	15
Figure 2.4 : Diagramme de contexte.....	16
Figure 2.5 : Diagramme préliminaire.....	17
Figure 2.6 : Diagramme d'état transition.....	19
Figure 2.7 : Schéma synoptique de la carte d'alimentation.....	20
Figure 2.8 : Schéma électrique de la carte d'alimentation.....	22
Figure 2.9 : Schéma synoptique de carte de commande câblée.....	23
Figure 2.10 : Circuit de puissance de carte de commande câblée.....	24
Figure 2.11 : Circuit d'acquisition de carte de commande câblée.....	24
Figure 2.12 : Les ports d'entrées/Sorties de la carte de commande câblée.....	25
Figure 2.13 : Schéma synoptique de la 1 ^{ère} carte de commande à base du PIC16F877.....	26
Figure 2.14 : Connexion de l'afficheur sur la carte de commande à base du PIC16F877.....	28
Figure 2.15 : Carte de programmeur.....	29
Figure 2.16 : schéma électrique de la première carte de commande de PIC.....	29
Figure 2.17 : Schéma synoptique de la 2 ^{ème} carte de commande à base du PIC 16F877.....	30
Figure 2.18 : schéma électrique de la deuxième carte de commande du PIC16F877.....	31
Figure 3.1 : la fenêtre ISIS Professional.....	32
Figure 3.2 : la fenêtre ARES Professional.....	33
Figure 3.3 : la carte de commande câblée.....	34
Figure 3.4 : la carte de commande basée sur le microcontrôleur PIC16F877.....	35
Figure 3.5 : La carte des boutons poussoirs.....	36
Figure 3.6 : description de l'interface du logiciel MikroPascal.....	37
Figure 3.7 : ouvrir un nouveau projet.....	38
Figure 3.8 : fenêtre de nouveau projet.....	39

Figure 3.9 : saisie du programme dans l'éditeur de code.....	39
Figure 3.10 : compilation du programme.....	40
Figure 3.11 : fenêtre d'erreur.....	40
Figure 3.12 : assistant de code.....	40
Figure 3.13 : assistant de paramètres.....	41
Figure 3.14 : « message d'accueil ».....	41
Figure 3.15 : première étape « choisir un étage ».....	42
Figure 3.16 : deuxième étape « l'arrivée à l'étage désiré ».....	43
Figure 3.17 : troisième étape « ouvrir la porte ».....	43
Figure 3.18 : quatrième étape « fermeture de la porte ».....	44
Figure 3.19 : cinquième étape « allumage de la cabine ».....	44
Figure 3.20 : L'arrêt d'urgence.....	45

Liste des tableaux :

Tableau 1.1 : les caractéristiques d'un PIC 16F877.....	6
Tableau 2.1: Spécification des données et des évènements.....	16
Tableau 2.2 : brochage <<normalisée>> des afficheurs LCD à logique intégrée.....	27

Table des Matières

Remerciements

Dédicaces

Résumés

Table des matières

Liste des figures et tableaux

Liste des abréviations

Introduction générale..... 1

Chapitre 1 : Généralités sur les microcontrôleurs

1-1 Introduction.....	3
1-2 Définition.....	3
1-3 Les différentes familles du microcontrôleur PICs.....	4
1-4 Identification du microcontrôleur PIC.....	4
1-5 Architecture du microcontrôleur PIC	4
1-6 Choix du microcontrôleur	5
1-6-1 Pourquoi un microcontrôleur PIC 16F877 ?	5
1-6-2 Les différents registres du PIC.....	7
1-7 Les particularités électriques du microcontrôleur PIC 16F877.....	8
1-7-1 Les broches d'alimentation.....	8
1-7-2 Les broches de quartz.....	9
1-7-3 La broche de réinitialisation.....	9
1-7-4 Les ports d'entrées / sorties.....	10
1-8 Conclusion.....	10

Chapitre 2 : Etude théorique du montage

2-1 Introduction	11
2-2 Présentation de l'ascenseur.....	11
2-3 Etude du fonctionnement de l'ascenseur.....	16
2-3-1 Spécification des données et des évènements.....	16
2-3-2 Diagramme de contexte	16
2-3-3 Diagramme préliminaire.....	17
2-3-4 Diagramme d'état de transition.....	18
2-4 Etude de la carte d'alimentation stabilisée.....	19
2-4-1 Description de la carte d'alimentation.....	19
2-4-2 Schéma synoptique de la carte d'alimentation.....	20
2-4-3 Choix du transformateur.....	20
2-4-4 Choix de pont de redressement.....	21
2-4-5 Choix des condensateurs.....	21
2-4-6 Choix de régulateur de tension.....	21
2-4-7 Schéma électrique de la carte d'alimentation.....	22

2-5 Carte à commande câblée	22
2-5-1 Schéma synoptique.....	23
2-5-2 Etude du circuit de puissance.....	23
2-5-3 Etude du circuit d'acquisition.....	24
2-5-4 Port d'entrées / sorties.....	25
2-6 Etude de la 1 ^{ère} carte de commande à base de microcontrôleur PIC16F877.....	25
2-6-1 Schéma synoptique.....	26
2-6-1-1 Diodes Leds.....	26
2-6-1-2 Afficheur LCD.....	26
2-6-1-3 Programmeur.....	28
2-6-2 Schéma électrique.....	29
2-7 Etude de la 2 ^{ème} carte de commande à base de microcontrôleur PIC16F877.....	30
2-7-1 Schéma synoptique.....	30
2-7-1-1 Diodes Leds.....	30
2-7-1-2 Les boutons poussoirs.....	31
2-7-2 Schéma électrique.....	31
2-8 Conclusion.....	31

Chapitre 3 : Réalisation et tests

3-1 Introduction.....	32
3-2 Présentation du logiciel Proteus.....	32
3-3 Implémentation des différentes cartes réalisées sur Proteus.....	33
3-3-1 Carte de commande câblée.....	34
3-3-2 Carte de commande à base du microcontrôleur PIC16F877.....	35
3-3-3 Carte des boutons poussoirs.....	36
3-4 Présentation du logiciel MikroPascal.....	36
3-4-1 Description du logiciel MicroPascal.....	37
3-4-1-1 La barre d'outils.....	37
3-4-1-2 Création d'un projet sur MicroPascal.....	38
3-4-2 Editeur de code.....	40
3-4-2-1 L'assistant de code.....	40
3-4-2-2 L'assistant de paramètres.....	41
3-5 Réalisation pratique et tests.....	41
3-6 Conclusion.....	45
Conclusion générale.....	46
Bibliographie.....	47
Annexes	
Nomenclature	

Conclusion générale

Notre projet de fin d'études intitulé « mise en œuvre d'un Kit didactique de commande d'ascenseur » est basée essentiellement sur deux solutions : la commande câblée et la commande par microcontrôleur PIC 16F877. La réalisation de ce type de projet permet de mettre en pratique un grand nombre de connaissances acquises au fil des études et engendre la grande satisfaction de voir notre travail se concrétiser réellement.

On est arrivé à réaliser les deux cartes de commande et comparer entre elles, d'abord la commande câblée qui est simple à l'utilisation et moins chère, mais où nous ne pouvons pas y faire des modifications, de même elle présente un encombrement des fils. Ensuite la commande à base de microcontrôleur PIC 16F877 qui est de faible consommation et qui présente un circuit imprimé peu complexe avec un coût réduit, on peut aussi y modifier le programme pour suivre le cahier de charge dans cette carte facilement.

La manipulation de ce projet nous a appris la maîtrise de l'outil de programmation des microcontrôleurs à travers le compilateur « MicroPascal », le routage des différents circuits imprimés avec le logiciel des schémas électriques « ARES » et la simulation des cartes de commande à travers le logiciel ISIS.

On peut proposer une troisième solution pour commander notre ascenseur avec une carte de commande par PC. Cette carte aura pour but de commander la cage d'ascenseur par PC avec une interface parallèle 8255 + port parallèle. Le P.P.I (Programmable Peripheral Interface) 8255, est un circuit périphérique programmable qui contient un registre de contrôle, un registre d'état et trois ports d'entrée/sortie A, B, C (divisé en deux: C-BAS et C-HAUT). Le port parallèle est une interface qui sert avant toutes choses à communiquer, entre un ordinateur et une imprimante, et le grand avantage de ce port est qu'il est composé de 17 lignes utiles et qui pour certaines d'entre elles sont bidirectionnels, donc son utilisation dans le domaine de l'électronique peut-être très variée. Malheureusement, nous n'avons pas pu trouver le matériel essentiel soit le P.P.I 8255 pour réaliser cette carte.

Finalement, nous souhaitons beaucoup de courage et de réussite à ceux qui nous suivront dans ce type de projets.

LES MICROCONTROLEURS (PICs de Microchip)

Le schéma fonctionnel de la **Figure 1.5** représente une architecture de "Von Neumann" (Commune à la plupart des microprocesseurs) où la mémoire programme partage le même bus que la mémoire de donnée. L'architecture de "Harvard", qui dispose de bus distincts pour les données et pour le programme, est plus rarement utilisée.

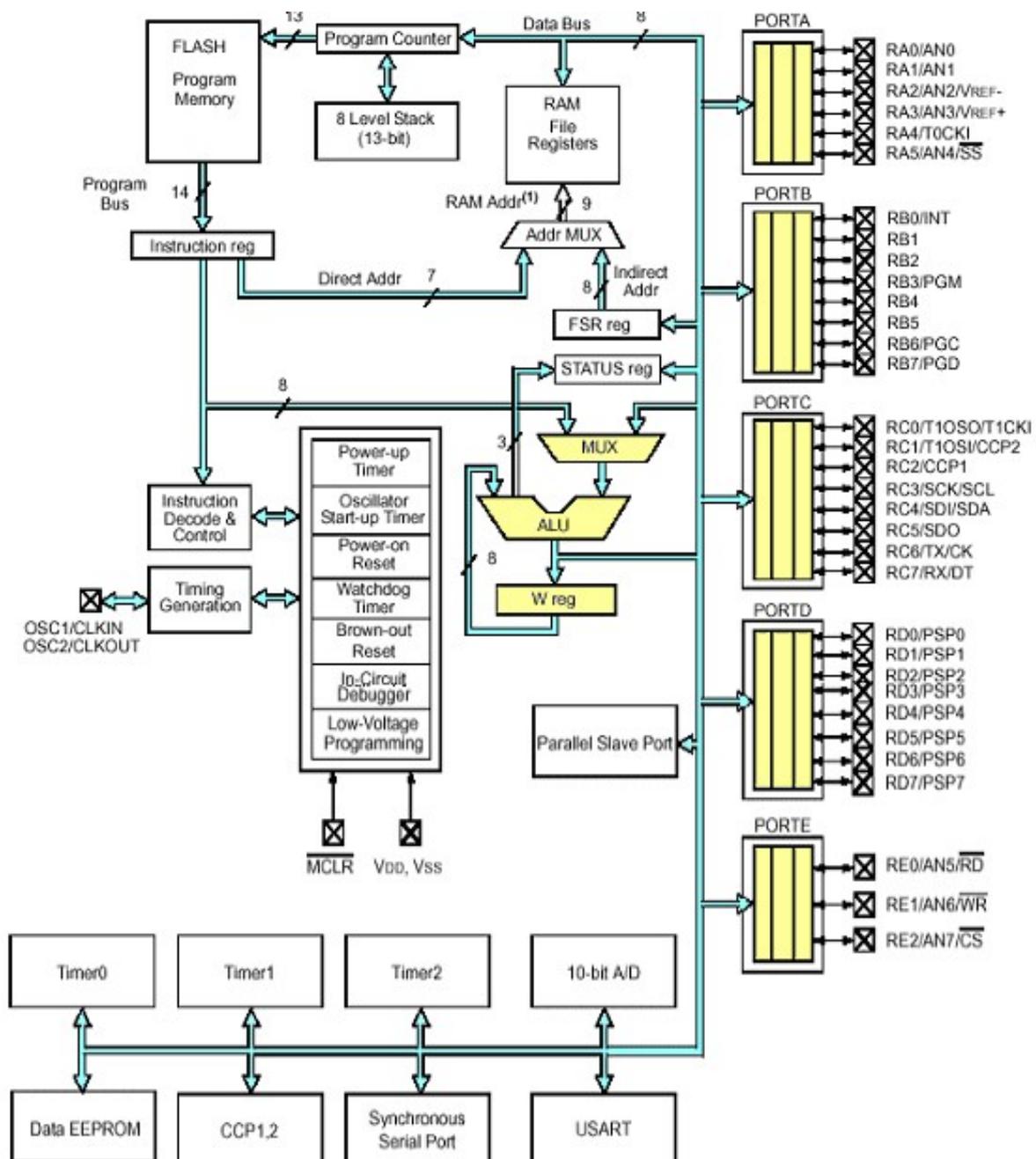


Figure 1.5 : Structure interne du 16F877

Les éléments de base du PIC 16F876/877

La Figure 1.6 montre les éléments de base du PIC 16F877

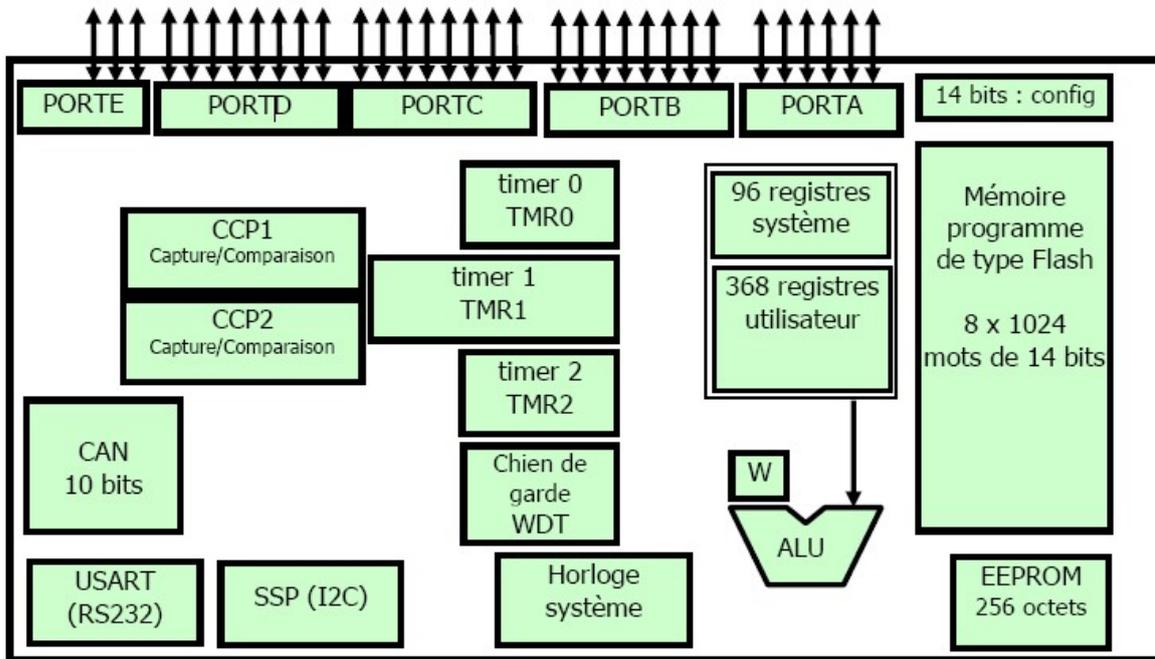
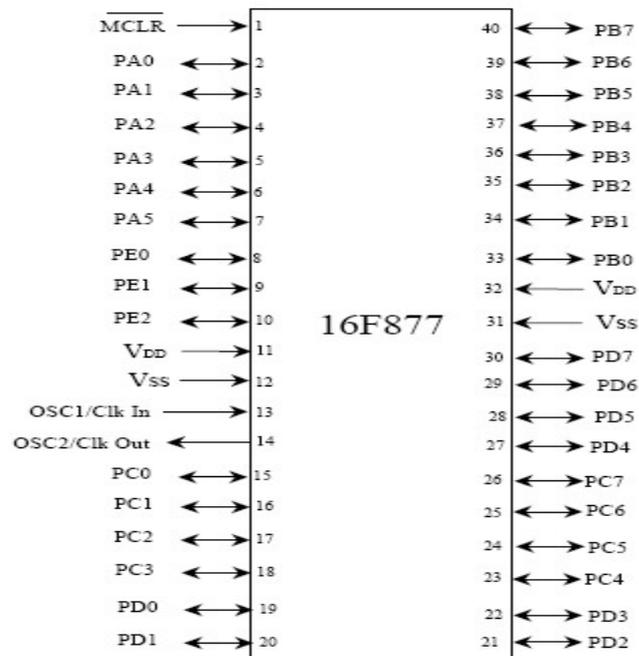


Figure 1.6 : Les éléments constitutifs du PIC 16F877

Le brochage du PIC 16F877 :

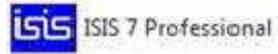
La Figure 1.7 montre le brochage du PIC16F877



. **Figure 1.7** : Le brochage du PIC 16F877

Les Logiciels Utilisés

Prise en main du logiciel ISIS (Intelligent Schematic Input System)

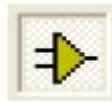


1. Démarrer le logiciel

Lancer depuis le raccourci du bureau ou du menu **Démarrer** de votre ordinateur le logiciel

Proteus – ISIS :  ISIS 7 Professional

2. Création de la liste des composants



➤ Sélectionner l'icône composant :

➤ Dans la fenêtre du sélectionneur d'objets, cliquer sur P 

- Une nouvelle fenêtre (**Pick Devices**) comme le montre la **Figure 3.1** de choix de composants s'ouvre : Trouver les composants désirés à partir d'un **mot clé**, ou dans la **catégorie** proposée, puis dans la **sous-catégorie** (par défaut choisissez dans la sous-catégorie **Generic**).



Figure 3.1 : fenêtre du **Pick Devices**

- Cliquer une fois pour faire apparaître le composant, s'il convient double cliquer pour qu'il s'inscrive sur la liste de travail **DEVICES**.
- Procéder ainsi, pour placer dès le début du TP tous les composants que vous aurez besoin par la suite sur la liste de travail, puis fermer la fenêtre en cliquant sur **OK**.
- Dans notre cas il faut sélectionner les composants « RES », « CAP » et « SWITCH », qui se mettent dans le « panier ».

3. Placement des composants sur le schéma

- Sélectionner le composant à placer dans la liste faite précédemment et le placer dans la zone de travail de la fenêtre d'édition (voir **Figure 3.2**).

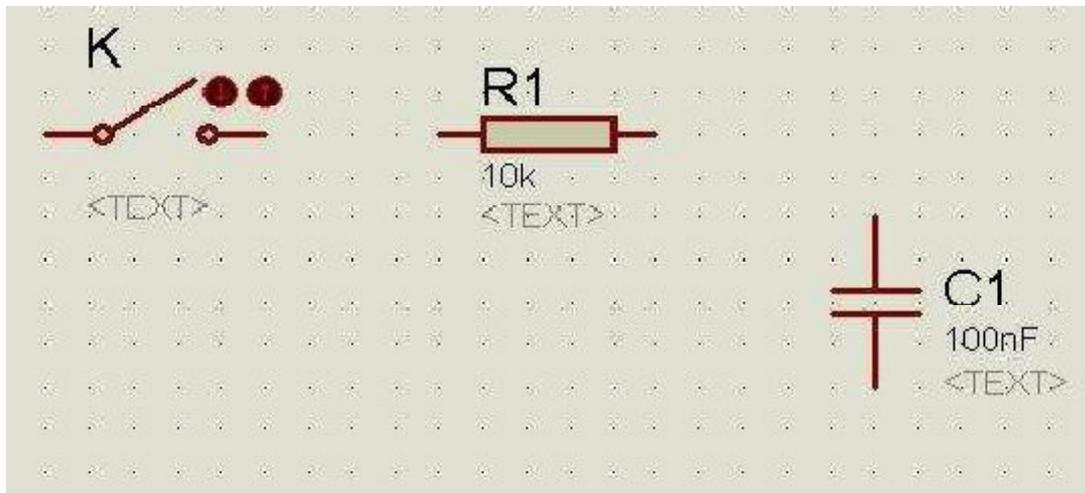


Figure 3.2 : le schéma de placement des composants

- Eventuellement le tourner ou lui faire faire une symétrie à l'aide des outils d'orientation du menu de la boîte à outils (**Figure 3.3**):

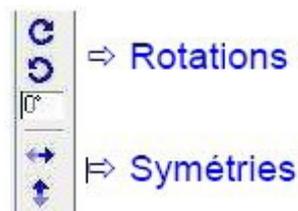


Figure 3.3 : les outils d'orientation

- Les composants placés, il faut définir leurs paramètres : Cliquer gauche sur le composant, le symbole devient rouge, re cliquer gauche, une fenêtre pour éditer les propriétés s'ouvre.
 - Nommez-les (R1, C1 et K) et affectez les valeurs suivantes : R1 = 10kF et C1= 100nF

4. Placement des différentes connexions externes

Pour GROUND; OUTPUT ; INPUT...Après avoir cliqué sur cette

fonction  Sélectionnez la connexion et après l'avoir mise dans le « panier », la placer sur le schéma. Pour notre schéma, nous utiliserons la masse (GROUND).

5. Placement des alimentations et des générateurs

Il faut maintenant placer les différentes alimentations ou générateurs utilisés.

- Cliquer sur et sélectionner un générateur (DC, SINE, PULSE...).
- Ensuite cliquer sur le schéma pour le placer (ici Pulse = rectangle).
- Les alimentations ou générateurs placés, il faut leur donner un nom et régler les différents paramètres qui sont fonctions du type de générateurs choisi.

Cliquer gauche sur le générateur, le symbole devient rouge, recliquer gauche, une fenêtre pour éditer les propriétés s'ouvre (**Figure 3.4**).

- Donner un nom (VE).
- Régler l'amplitude des tensions : $V_{Emin} = 0V$ et $V_{Emax} = 10V$
- Régler pour avoir un signal carré de fréquence 200 Hz.

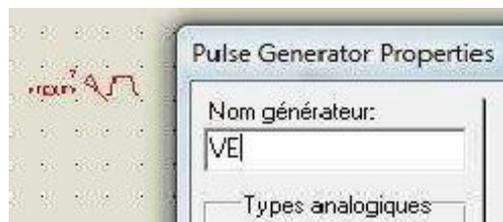


Figure 3.4 : fenêtre du Pulse Generator Properties

6. Réalisations des connexions

Il nous reste à relier les connexions entre les différents symboles.

- Sélectionner dans le menu  Aller sur l'extrémité d'un composant, cliqué, le début de la liaison est créée, puis aller cliquer sur le composant à relier pour finir la liaison, le logiciel positionne seul le chemin (**Figure 3.5**).

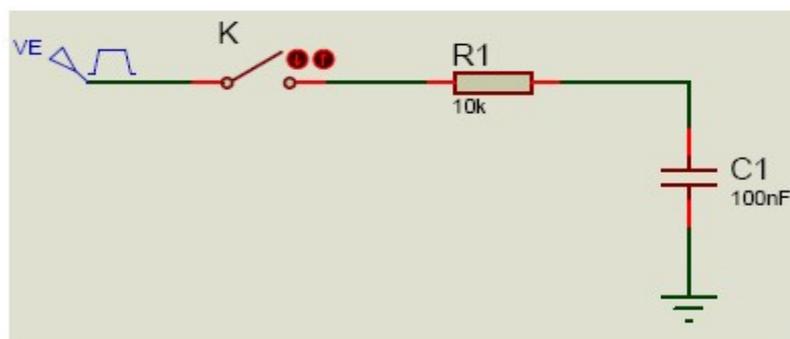


Figure 3.5 : le schéma de liaison

7. Placement des instructions des mesures

Il faut choisir le ou les appareils de mesure que l'on souhaite utiliser en cliquant sur l'icône :

Comme le montre la **Figure 3.6**



Figure 3.6 : fenêtre des appareils de mesure

Pour un voltmètre continu (**DC VOLTMETER**), comme pour un voltmètre réel, il mesure une DDP (différence de potentiel) entre les 2 points connectés, il faut donc le relier à 2 points. Comme le montre la **Figure 3.7** :

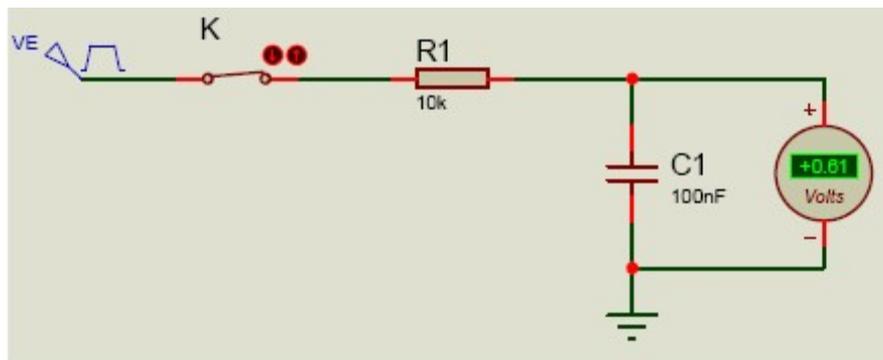


Figure 3.7

8 .Animation-simulation du fonctionnement

Pour lancer l'animation. Cliquer sur la flèche, comme sur un appareil multimédia, Fermer l'interrupteur **K** en cliquant sur la flèche de commande, il est maintenant possible de lire les informations données par le voltmètre en temps réel.

9 .utilisation avancée des appareils de mesure

Réaliser le schéma de la **Figure 3.8**, avec $R1 = 10 \text{ k}\Omega$ et $C1 = 100 \text{ nF}$. Le générateur VE devra être paramétré ainsi : VE compris entre 0 V et 10 V ; forme rectangulaire de rapport cyclique 0,5 (soit une largeur d'impulsion de 50%) ; fréquence = 200Hz. Placer les instruments de mesures suivants : un ampèremètre (**DC AMMETER**) ; un voltmètre (**DC VOLTMETER**) et un oscilloscope.

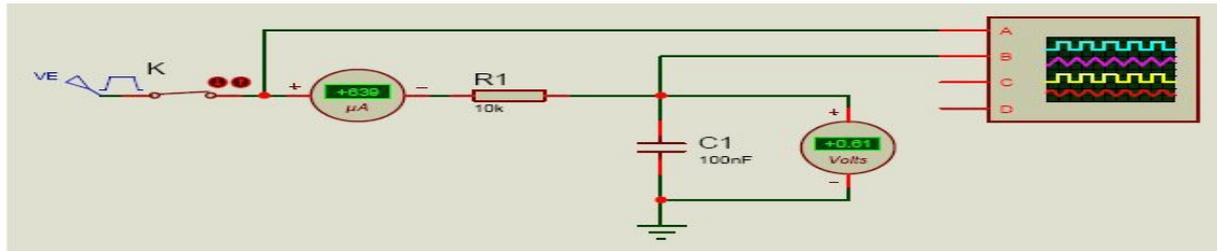


Figure 3.8

Lancer la simulation.

Si l'écran de l'oscilloscope n'apparaît pas, vous devez arrêter la simulation, cliquer gauche une fois sur l'oscilloscope (il devient rouge) et aller sur le menu **Mise au point** puis valider sur **Réinitialiser fenêtres popup**. Vous pouvez alors relancer la simulation, elle se déroule en temps réel comme le montre la **Figure 3.9**. Procéder en suite aux réglages :

- Mettre les **voies C et D** de l'oscilloscope sur OFF.
- Régler la base de temps de l'oscilloscope pour avoir plusieurs périodes sur l'écran.
- Mettre la **voie A** sur DC et régler la sensibilité avec le commutateur, au besoin bien repositionner le signal avec la molette.
- Mettre la **voie B** sur DC ou AC pour supprimer la composante continue, régler la sensibilité pour avoir une bonne amplitude, au besoin bien repositionner le signal avec la molette.

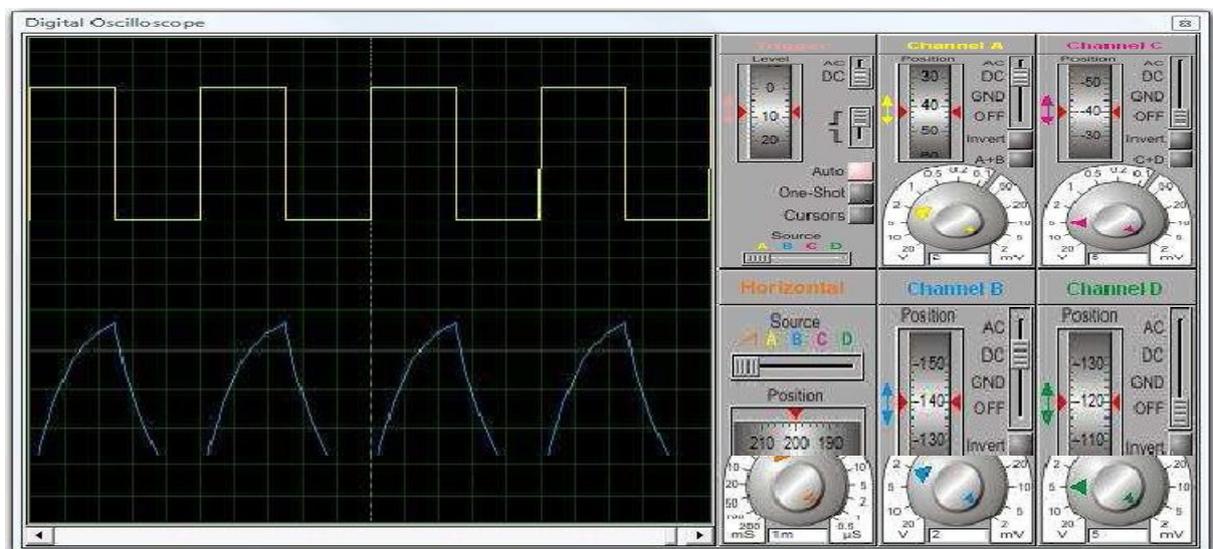


Figure 3.9 : le schéma de l'oscilloscope

Prise en main du logiciel ARES



A partir de l'environnement ISIS Sélectionner l'icône rouge à droite :  comme le montre la Figure 3.10 :

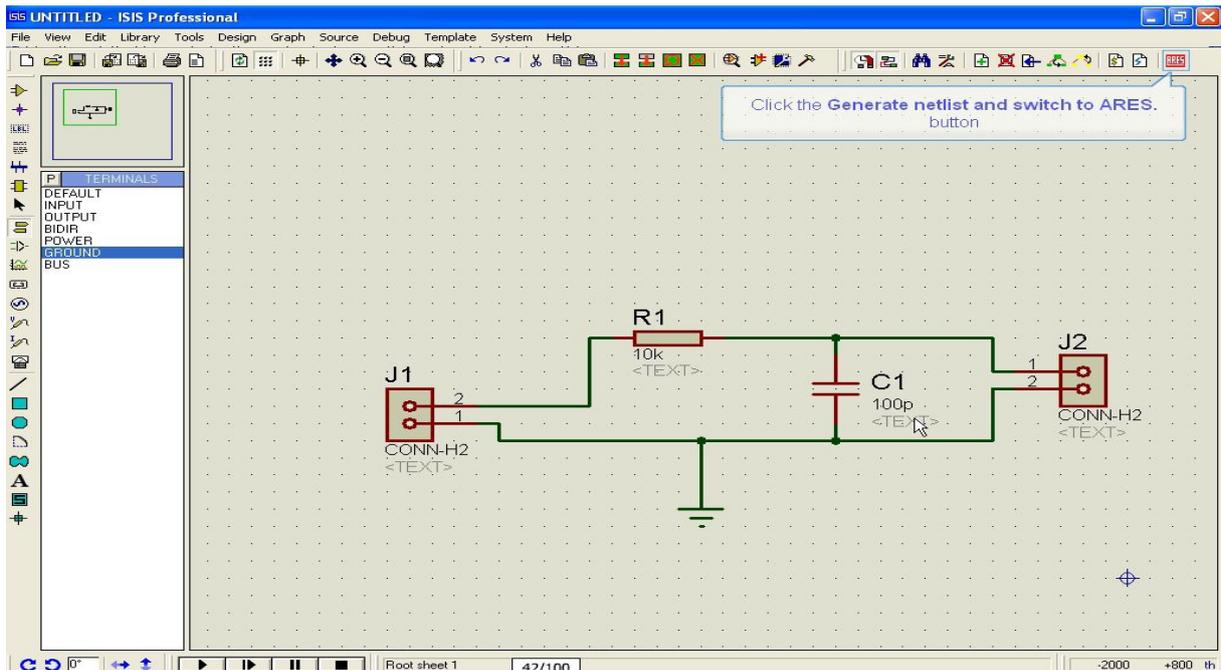


Figure 3.10



Sélectionner l'icône composant:

Lorsque on termine de mettre les composants click sur l'icône « tools » de la Figure 3.11 :

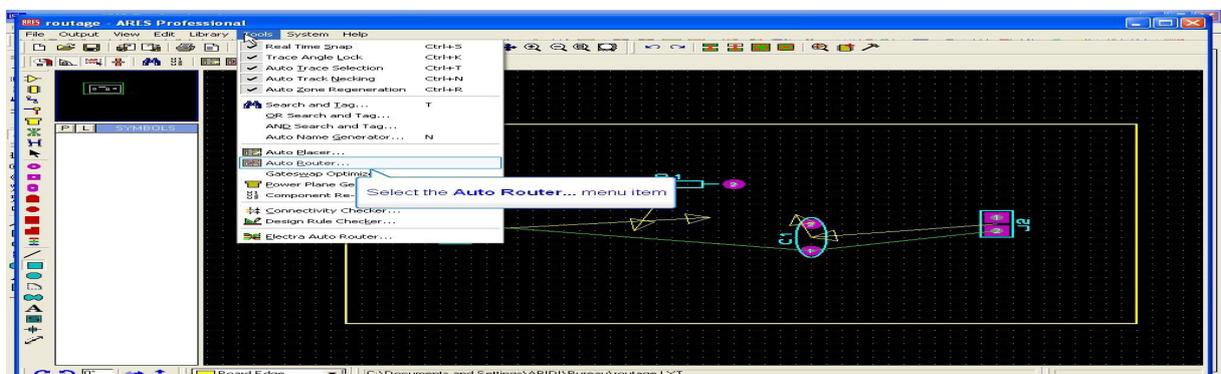


Figure 3.11

Click sur « Edit Stratégies » de la **Figure 3.12** :

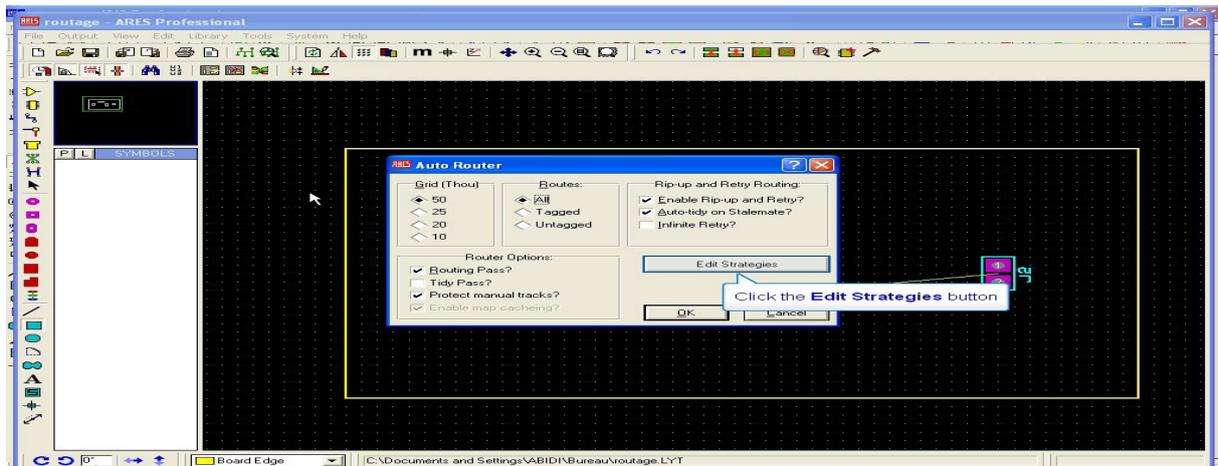


Figure 3.12

Puis réglé les paramètres comme le montre la **Figure 3.13** suivante :

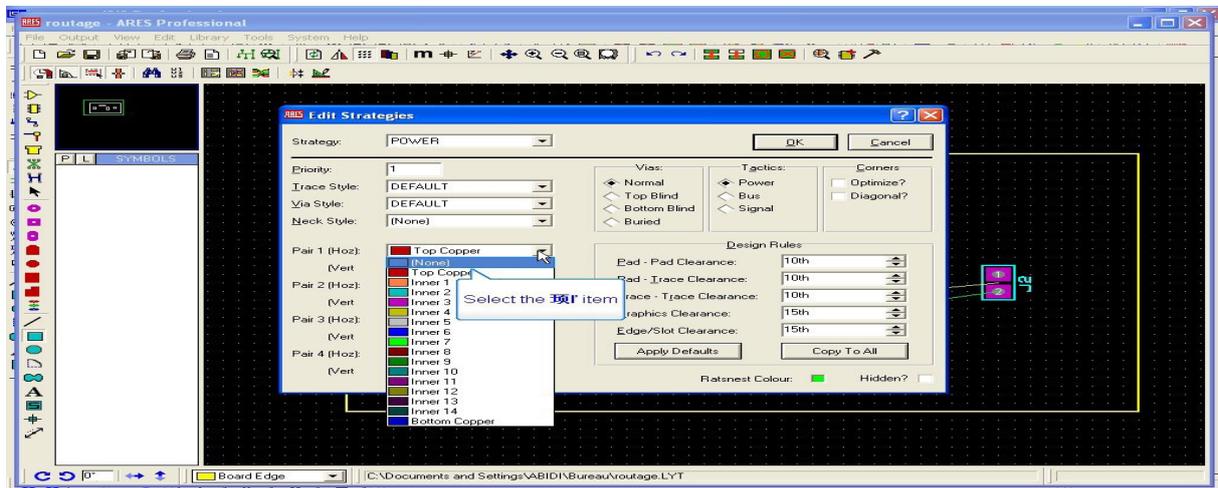


Figure 3.13

Click sur OK de la **Figure 3.14**.

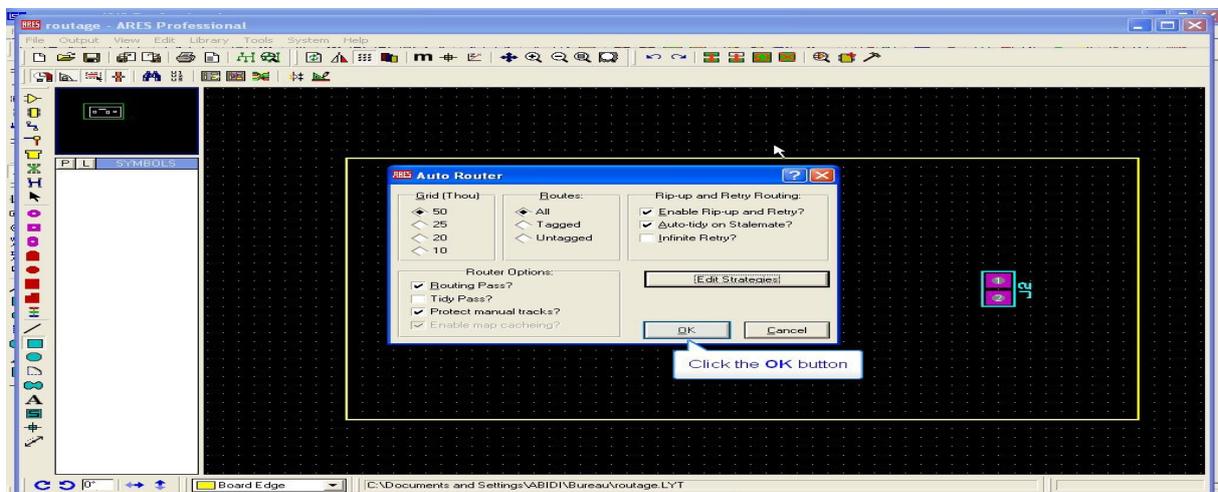


Figure 3.14

En fin vous trouvez le routage d'ARES suivant comme le montre la Figure 3.15 :

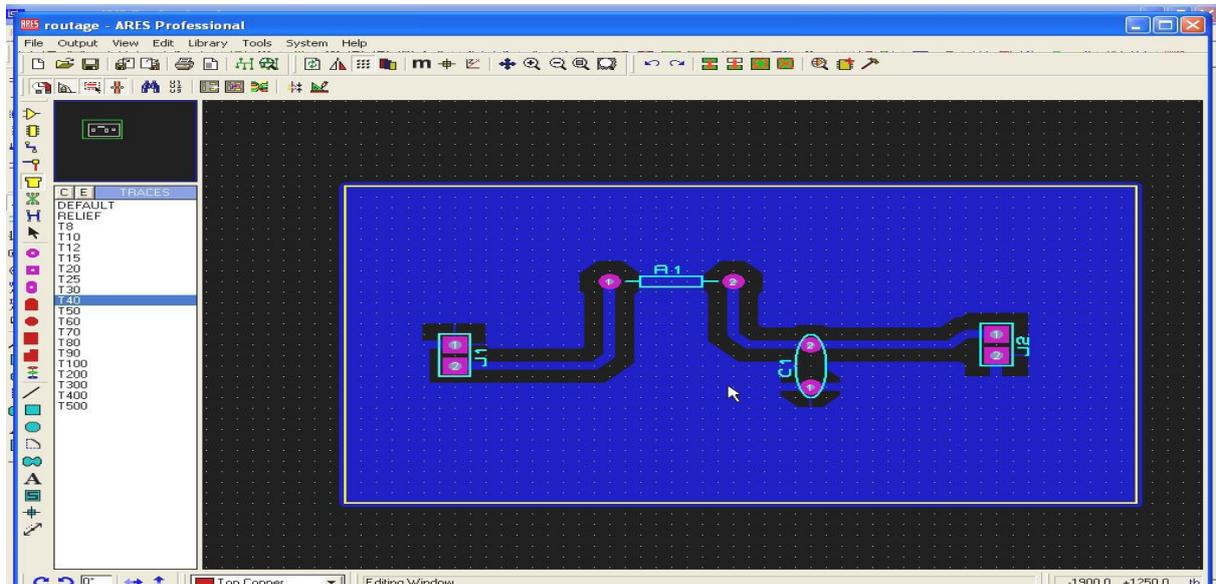


Figure 3.15 : le schéma de routage

Prise en main du logiciel IC-PROG IC-PROG



Figure 3.16

Définition

IC-PROG (Figure 3.16) est un programme qui nous permet de transférer un fichier compilé vers un PIC ou une mémoire.

C'est le logiciel qu'avec on va s'aider dans la programmation du PIC 16F877.

Utilisation

Après le lancement de notre logiciel la première étape à faire est de choisir le PIC qu'on va programmer 16F877 comme le montre la **Figure 3.17**.

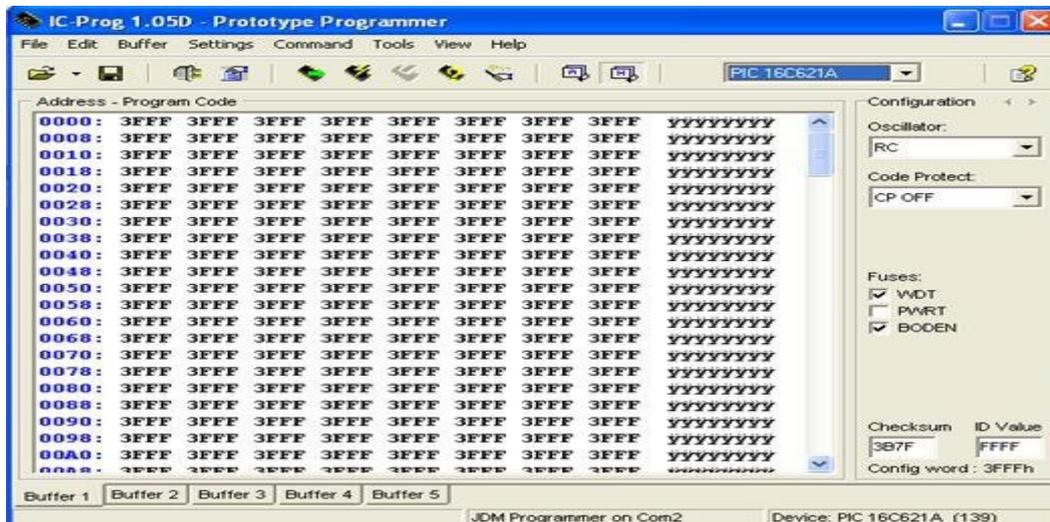


Figure 3.17

Configuration D'IC-PROG

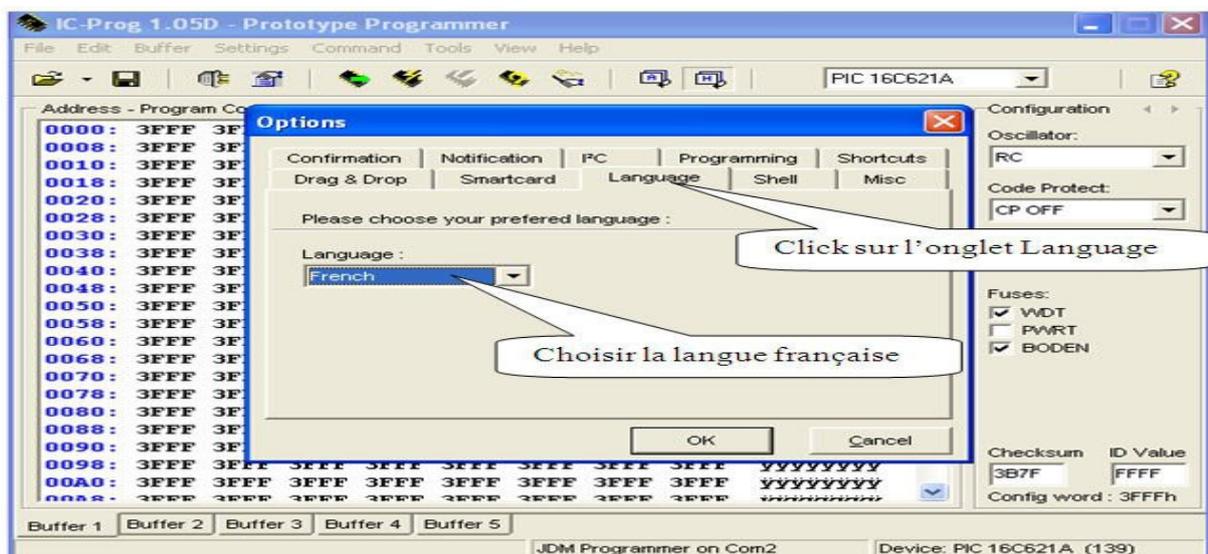


Figure 3.18

Cette page de la **Figure 3.18** décrit une méthode d'initialisation d'ICPROG sous Windows XP Pro et XP Home Edition (merci à Yves qui a testé ICPROG avec la version Home

Edition). Assurez-vous dans un premier temps de disposer des trois fichiers (**Figure 3.19**) suivants dans votre répertoire contenant ICPROG, tel que :



Figure 3.19

Celui de gauche, représente le fichier d'aide ICPROG.CHM, celui de droite l'exécutable du logiciel ICPROG14.EXE et enfin c ICPROG.SYS nécessaire au fonctionnement sous Windows XP.

Remarque : ICPROG14.EXE correspond à la version V1.4 d'ICPROG. Les nouvelles versions d'ICPROG disponibles sur le site de l'auteur portent un nom différent. Utilisez toujours la dernière version d'ICPROG proposée au téléchargement.

Les circuits imprimés

La **Figure 3.20** représente le schéma de circuit imprimé de la carte de commande basée sur le microcontrôleur PIC16F877 implémenté sur le logiciel ARES.

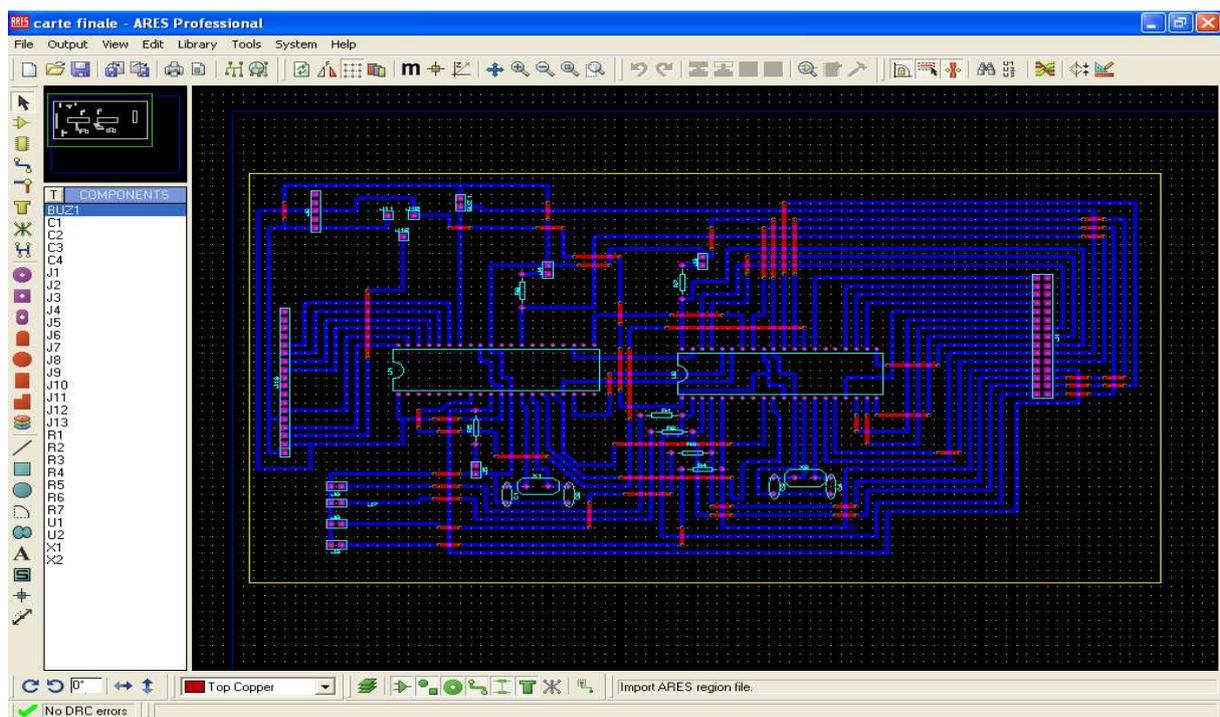


Figure 3.20 : le circuit imprimé de la carte de commande à base du PIC16F877

La **Figure 3.21** représente le schéma de circuit imprimé de la carte des boutons poussoirs implémenté sur le logiciel ARES.

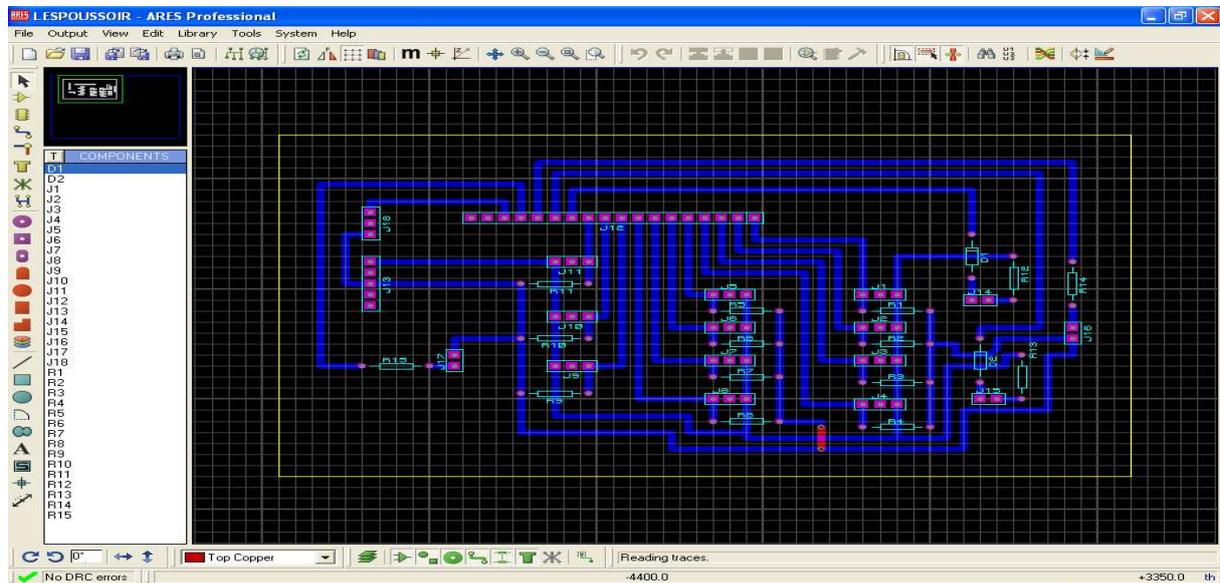


Figure 3.21 : le circuit imprimé de la carte des boutons poussoirs.

La **Figure 3.22** représente le schéma de circuit imprimé de la carte de commande câblée implémenté sur le logiciel ARES.

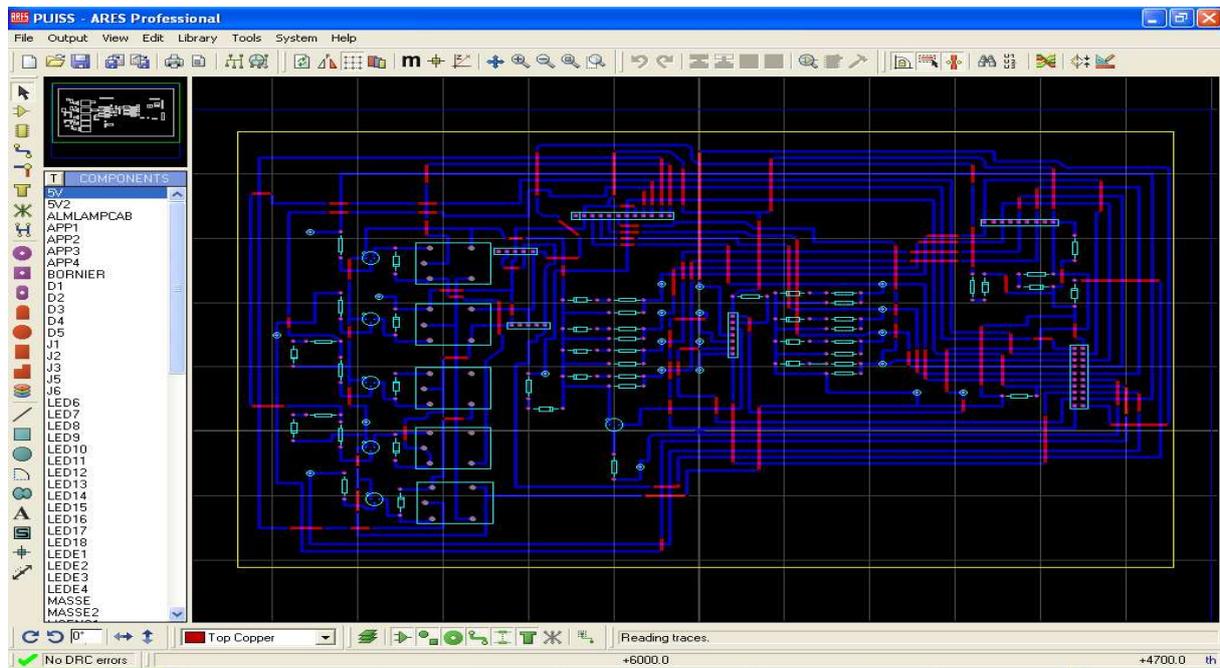
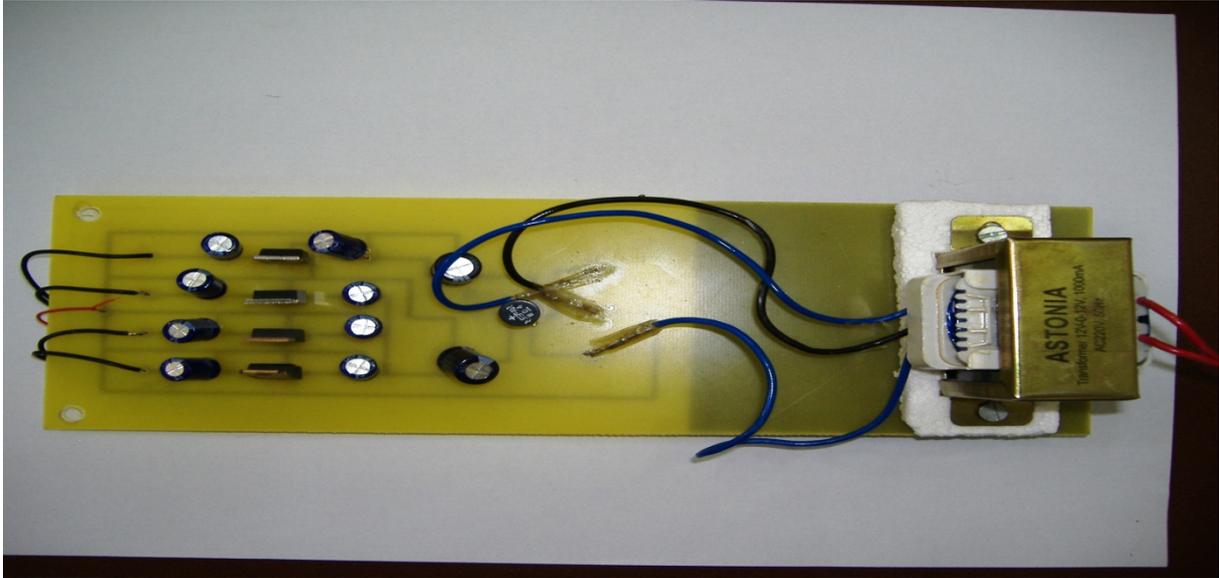
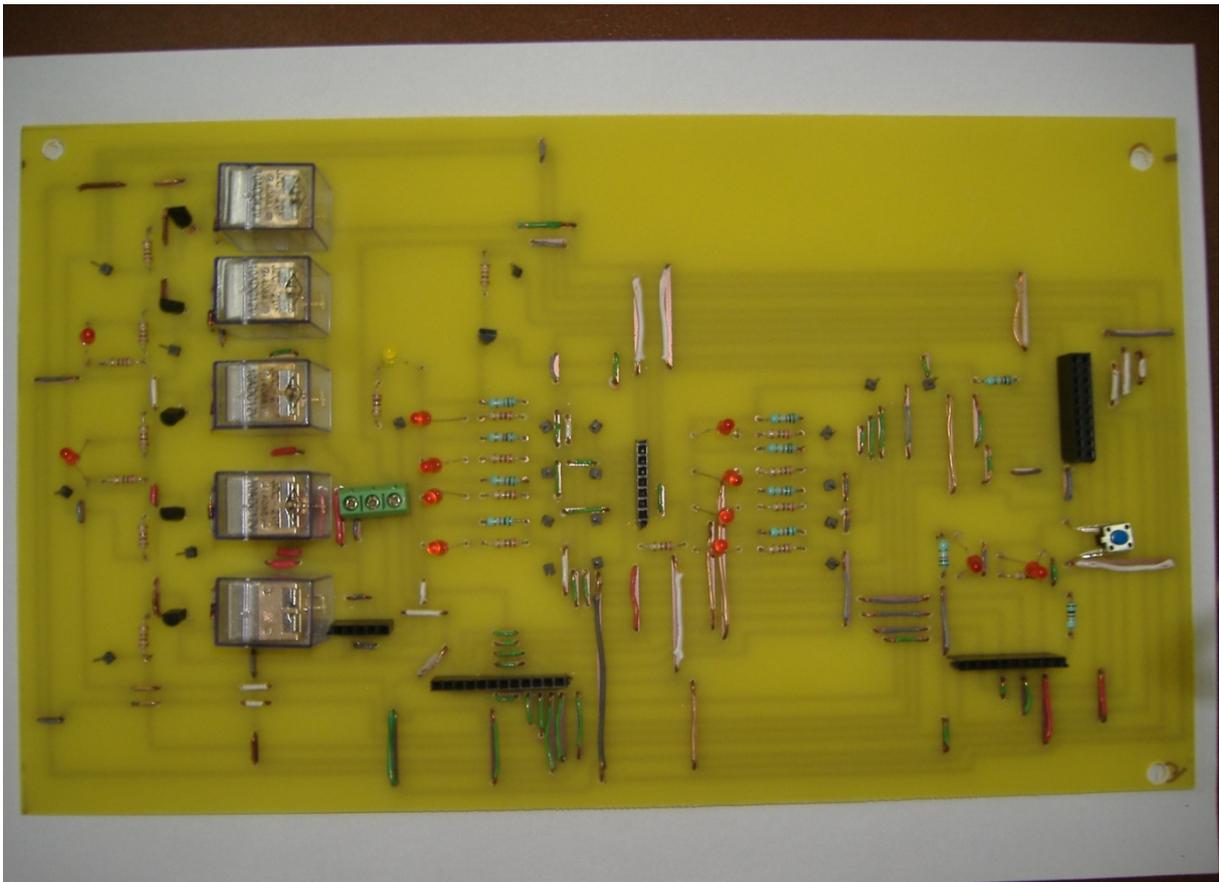


Figure 3.22 : le circuit imprimé de la carte de commande câblée.

Les photos réelles des différentes cartes réalisées**Figure 3.23** : photos réelle de la carte d'alimentation.**Figure 3.24** : photos réelle de la carte de commande câblée.

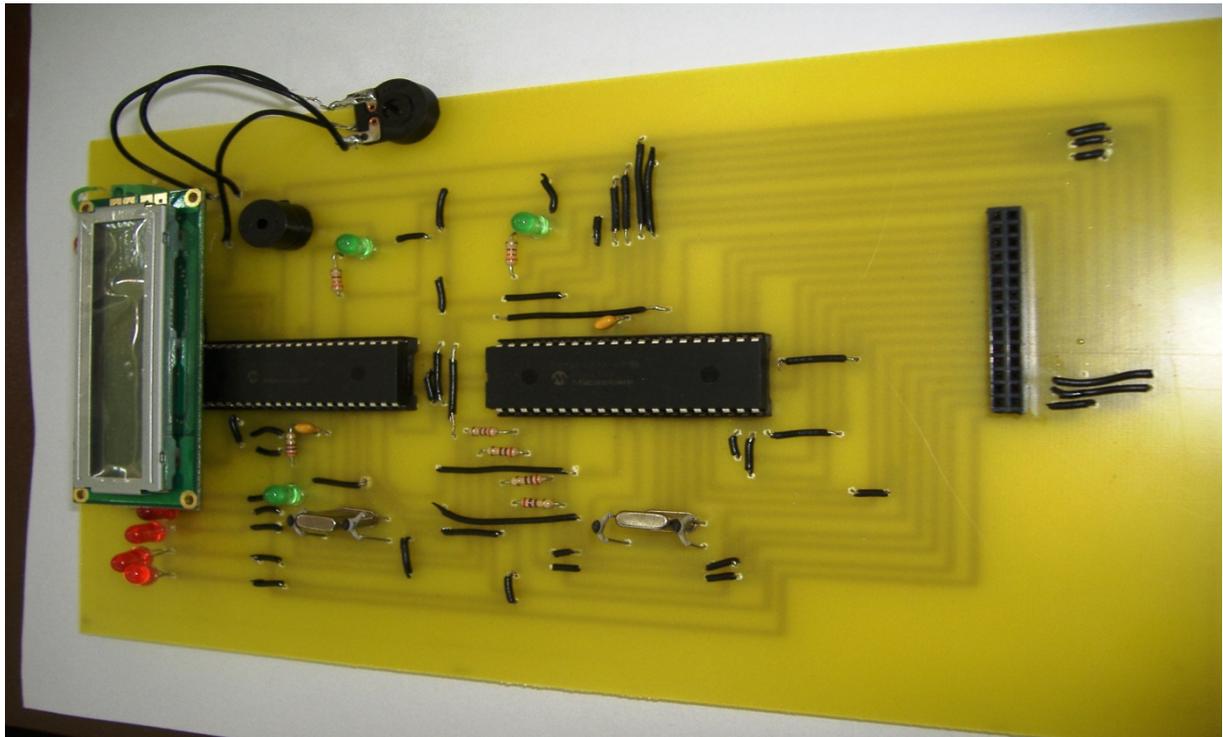


Figure 3.25 : photos réelle de la carte de commande basée sur le PIC 16F877.

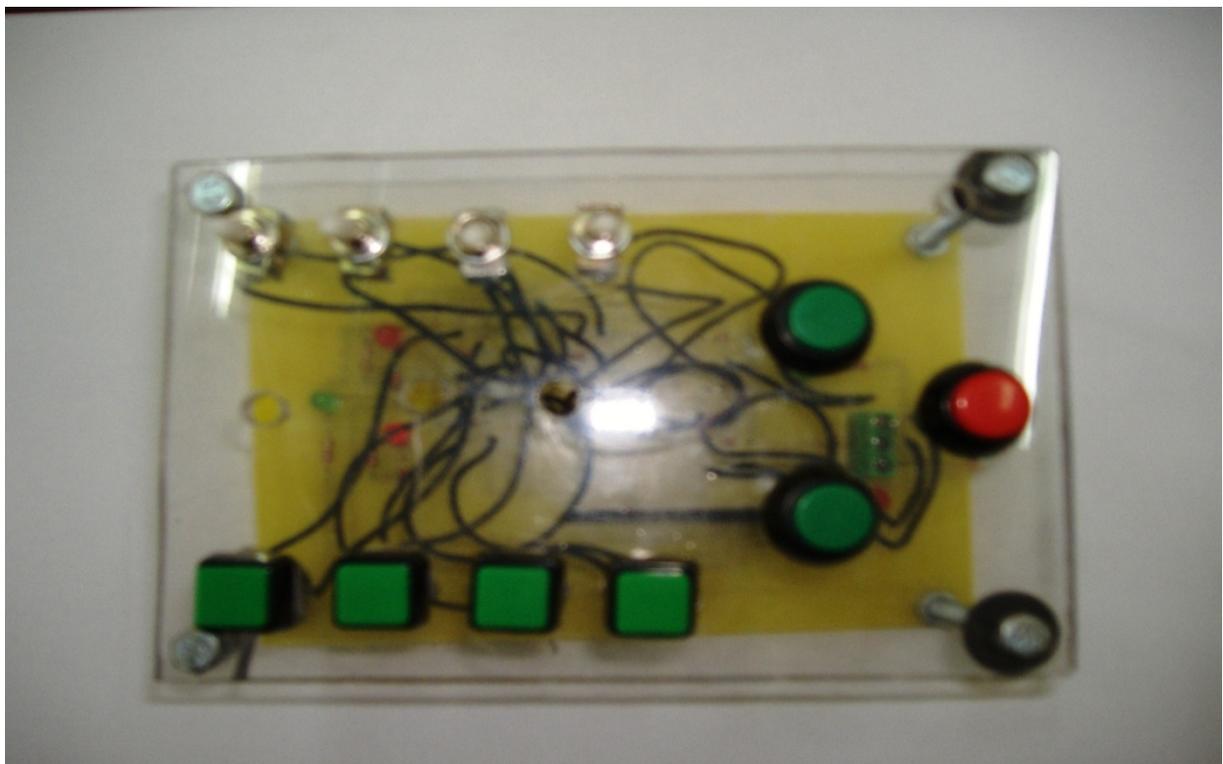


Figure 3.26 : photos réelle de la carte des boutons pousoirs.

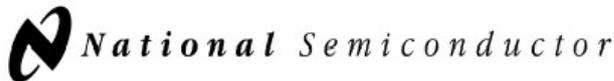


Figure 3.27 : photos réelle du montage final (position 1).



Figure 3.28 : photos réelle du montage finale (position 2).

Le régulateur LM7812



May 2000

LM78XX Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expended to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the out-

put, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

Features

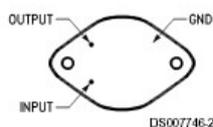
- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

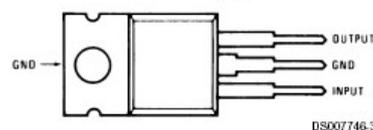
Connection Diagrams

Metal Can Package
TO-3 (K)
Aluminum



Bottom View
Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)

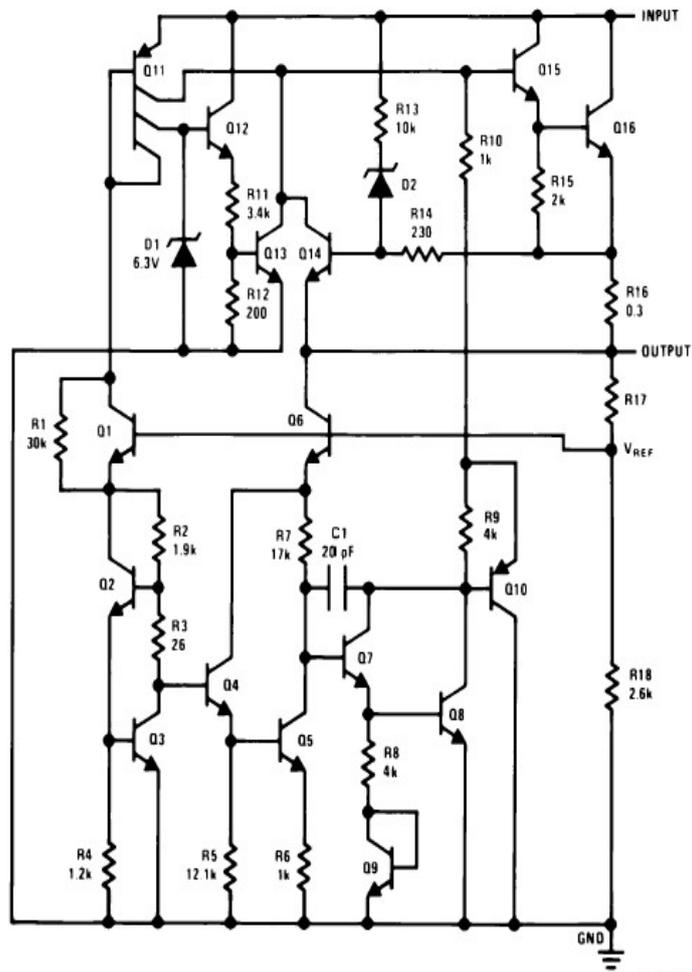


Top View
Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B

LM78XX Series Voltage Regulators

LM78XX

Schematic



DS007746-1

LM78XX

Absolute Maximum Ratings (Note 3)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Input Voltage
 ($V_O = 5V, 12V$ and $15V$) 35V
 Internal Power Dissipation (Note 1) Internally Limited
 Operating Temperature Range (T_A) 0°C to $+70^\circ\text{C}$

Maximum Junction Temperature
 (K Package) 150°C
 (T Package) 150°C
 Storage Temperature Range -65°C to $+150^\circ\text{C}$
 Lead Temperature (Soldering, 10 sec.)
 TO-3 Package K 300°C
 TO-220 Package T 230°C

Electrical Characteristics LM78XXC (Note 2)

$0^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$ unless otherwise noted.

Output Voltage			5V			12V			15V			Units	
Input Voltage (unless otherwise noted)			10V			19V			23V				
Symbol	Parameter	Conditions	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max		
V_O	Output Voltage	$T_J = 25^\circ\text{C}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V	
		$P_D \leq 15\text{ W}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.75		5.25	11.4		12.6	14.25		15.75	V	
		$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$			($7.5 \leq V_{\text{IN}} \leq 20$)			($14.5 \leq V_{\text{IN}} \leq 27$)			($17.5 \leq V_{\text{IN}} \leq 30$)	V	
ΔV_O	Line Regulation	$I_O = 500\text{ mA}$	$T_J = 25^\circ\text{C}$	3	50		4	120		4	150	mV	
			ΔV_{IN}			($7 \leq V_{\text{IN}} \leq 25$)			($14.5 \leq V_{\text{IN}} \leq 30$)			($17.5 \leq V_{\text{IN}} \leq 30$)	V
		$0^\circ\text{C} \leq T_J \leq +125^\circ\text{C}$	ΔV_{IN}			50			120			150	mV
			ΔV_{IN}			($8 \leq V_{\text{IN}} \leq 20$)			($15 \leq V_{\text{IN}} \leq 27$)			($18.5 \leq V_{\text{IN}} \leq 30$)	V
		$I_O \leq 1\text{ A}$	$T_J = 25^\circ\text{C}$			50			120			150	mV
			ΔV_{IN}			($7.5 \leq V_{\text{IN}} \leq 20$)			($14.6 \leq V_{\text{IN}} \leq 27$)			($17.7 \leq V_{\text{IN}} \leq 30$)	V
ΔV_O	Load Regulation	$T_J = 25^\circ\text{C}$	$5\text{ mA} \leq I_O \leq 1.5\text{ A}$	10	50		12	120		12	150	mV	
			$250\text{ mA} \leq I_O \leq 750\text{ mA}$			25			60			75	mV
		$5\text{ mA} \leq I_O \leq 1\text{ A}, 0^\circ\text{C} \leq T_J \leq +125^\circ\text{C}$			50			120			150	mV	
I_Q	Quiescent Current	$I_O \leq 1\text{ A}$	$T_J = 25^\circ\text{C}$		8			8			8	mA	
			$0^\circ\text{C} \leq T_J \leq +125^\circ\text{C}$			8.5			8.5			8.5	mA
ΔI_Q	Quiescent Current Change	$5\text{ mA} \leq I_O \leq 1\text{ A}$	$T_J = 25^\circ\text{C}, I_O \leq 1\text{ A}$		0.5			0.5			0.5	mA	
			$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$			1.0			1.0			1.0	mA
			$(7.5 \leq V_{\text{IN}} \leq 20)$			($14.8 \leq V_{\text{IN}} \leq 27$)			($17.9 \leq V_{\text{IN}} \leq 30$)			V	
V_N	Output Noise Voltage	$T_A = 25^\circ\text{C}, 10\text{ Hz} \leq f \leq 100\text{ kHz}$	$I_O \leq 500\text{ mA}, 0^\circ\text{C} \leq T_J \leq +125^\circ\text{C}$		1.0			1.0			1.0	mA	
			$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$			($7 \leq V_{\text{IN}} \leq 25$)			($14.5 \leq V_{\text{IN}} \leq 30$)			($17.5 \leq V_{\text{IN}} \leq 30$)	V
$\frac{\Delta V_{\text{IN}}}{\Delta V_{\text{OUT}}}$	Ripple Rejection	$f = 120\text{ Hz}$	$I_O \leq 1\text{ A}, T_J = 25^\circ\text{C}$	62	80		55	72		54	70	dB	
			or $I_O \leq 500\text{ mA}$			62			55			54	dB
			$0^\circ\text{C} \leq T_J \leq +125^\circ\text{C}$			($8 \leq V_{\text{IN}} \leq 18$)			($15 \leq V_{\text{IN}} \leq 25$)			($18.5 \leq V_{\text{IN}} \leq 28.5$)	V
R_O	Dropout Voltage	$T_J = 25^\circ\text{C}, I_{\text{OUT}} = 1\text{ A}$		2.0			2.0			2.0		V	
			Output Resistance			8			18			19	m Ω

Afficheur LCD

Voilà les deux figures suivantes (2.20 et 2.21) qui nous montrent l'afficheur LCD avec son schéma fonctionnel.



Figure 2.1 : photos réel d'un afficheur LCD

WM-C1602M.pdf - Adobe Reader

Fichier Edition Affichage Document Outils Fenêtre Aide

1 / 1 110% Rechercher

WM-C1602M

16x2 CHARACTERS

FEATURE:

- 1.5 x 7 Dots with Cursor
- Built in Controller (HD44780 or Equivalent)
- +5V Power Supply (Also Available for +3V)
- 1/16 Duty Cycle

ELECTRICAL CHARACTERISTICS:

Item	Sym.	Condition	Min.	Typ.	Max.	Unit.
Supply Voltage for LCD	$V_{DD}-V_o$	Ta=25°C	----	4.5	----	V
Supply Current	I_{DD}	$V_{DD}=5V$	----	2.0	3.0	mA

DISPLAY CHARACTER ADDRESS CODE:

Display position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

INTERFACE PIN CONNECTIONS:

1	2	3	4	5	6	7	8
VSS	V_{DD}	V_o	RS	R/W	E	DB0	DB1
9	10	11	12	13	14	15	16
DB2	DB3	DB4	DB5	DB6	DB7	LED A	LEDK

Version	L1
EL. & No Backlight	73.2 mm
LED Backlight	71.2 mm

EL. & No Backlight LED Backlight DETAIL "A"

Figure 2.2 : le schéma fonctionnel de l'afficheur LCD

- [1] H. ABIDI, M. MEDDEB et S. MIZEN, « Mise en œuvre d'un kit didactique de commande d'un ascenseur ». Thèse en électronique, Tunisie, 2008.
- [2] B. BEGHYN, « Les microcontrôleurs PIC ». Edition Lavoisier, ISBN 2-7462-0764-8, Décembre 2003.
- [3] L. BRES, E. MARTINEZ et Y. OLIVERI, « L'enseignement des Systèmes automatisés pilotés par ordinateur », Thèse de Doctorat, Spécialité : Electronique, IUFM Aix Marseille, France, 1999.
- [4] Présentation Technique De L'ascenseur. www.ac-reims.fr/datice/stilp/batiment_industrie/.../rpf_mcta.pdf
- [5] <http://stielec.ac-aix-marseille.fr>
- [6] R. TAYEB CHERIF, T. MEDJADJ et S. MOKRANE, « Etude et réalisation d'un Fantôme automatique pour la dosimétrie absolue ». Thèse d'ingénieur en électronique, Option : contrôle, Institut d'électronique, USDB, Octobre 2007.
- [7] <http://www.technologuepro.com>
- [8] <http://www.mikroe.com>