

## PROJET DE FIN D'ETUDES

Filière :

Ingénieurs des Travaux de l'Etat

# CONCEPTION ET REALISATION D'UN TRANSMETTEUR TELEPHONIQUE D'ALARME A MICROCONTROLEUR

Elaboré par :

**Mounir KERFAI**

**Lotfi SOUSSI**

Encadré par :

**Fethi TLILI**

**Zouhaier BEN ABDALLAH**

Année universitaire : 2001 / 2002

**Conception et réalisation d'un transmetteur Téléphonique d'alarme à microcontrôleur****RESUME**

L'objectif de ce travail consiste à l'étude, la conception et la réalisation d'un transmetteur téléphonique d'alarme permettant d'émettre via le réseau téléphonique un message vocal de détresse prés enregistré dans une mémoire en cas de disfonctionnement ou d'infraction du système télé-surveillé. Ce message est envoyé à un ou plusieurs correspondants dont les numéros téléphoniques sont préprogrammés et mémorisés dans une EEPROM. La commande de l'ensemble est assurée par un microcontrôleur et l'affichage des différents états de la ligne ainsi que ceux du système télé-surveillé par un afficheur à cristaux liquides.

**MOTS CLES**

Transmetteur – Alarme – Télé surveillance – Microcontrôleur - DTMF  
Afficheur LCD - EEPROM - Programmation - Mémoire analogique

## **AVANT- PROPOS**

Ce projet de fin d'étude à été réalisé à l'Ecole Supérieure des Télécommunications de Tunis « SUP'COM » dans le cadre de la formation continue cours du soir pour l'obtention du grade d'Ingénieur des travaux.

Au terme de ce travail nous tenons à remercier nos encadreurs Monsieur Fethi TLILI Assistant à l'Ecole Supérieure des Télécommunications et Monsieur Zouhaier Ben ABDALLAH Ingénieur des Travaux enseignant à l'Institut Supérieur des Etudes Technologiques en Communications, pour leurs disponibilités et conseils durant toute la période.

Nos remerciements s'adresse aussi à tous nos professeurs et à tous ceux qui nous ont aider pour aboutir à ce résultat.

# CHAIER DES CHARGES

## Titre du projet :

Etude, conception et réalisation d'un transmetteur téléphonique d'alarme commandé par microcontrôleur.

## Position du problème

La mise en œuvre de systèmes de télésurveillance à microcontrôleur se trouve confronté à des contraintes imposées d'une part par le réseau téléphonique telle que caractéristiques des lignes téléphoniques terminales, la signalisation téléphonique, la numérotation et d'autre part par l'utilisation des systèmes numériques telle que microprocesseur et microcontrôleur. Il est donc recommandé d'étudier les caractéristiques du réseau téléphonique puis celles des microcontrôleurs avant de commencer l'étude et la conception de notre transmetteur téléphonique d'alarme.

## Travail demander :

- Analyse des caractéristiques des ligne téléphoniques
- Etude de la signalisation terminale
- Etude des microcontrôleurs
- Etude d'un système programmable
- Etude des afficheurs à cristaux liquides
- Etude des mémoires
- Conception et réalisation de l'interface ligne

- Conception et réalisation de la carte de commande
- Programmation et mise en marche

### **Caractéristiques du transmetteur :**

- Connexion par une prise RJ11 à la ligne téléphonique
- Entrée pour plusieurs interfaces de détection d'alarmes
- Sortie sirène
- Alimentation : 220V/50Hz, 12V DC
- Batterie de secours
- Programmation des N° téléphoniques à partir d'un poste simple
- Mémoire pour le stockage des numéros téléphoniques°
- Afficheur LCD à 2 lignes et 16 caractères

## **RESUME**

L'objectif de ce travail consiste à l'étude, la conception et la réalisation d'un transmetteur téléphonique d'alarme permettant d'émettre via le réseau téléphonique un message vocal de détresse prés enregistré dans une mémoire en cas de dysfonctionnement ou d'infraction du système télé-surveillé. Ce message est envoyé à un ou plusieurs correspondants dont les numéros téléphoniques sont préprogrammés et mémorisés dans une EEPROM. La commande de l'ensemble est assurée par un microcontrôleur et l'affichage des différents états de la ligne ainsi que ceux du système télé-surveillé par un afficheur à cristaux liquides.

## **MOTS CLES**

Transmetteur – Alarme – Télé surveillance – Microcontrôleur - DTMF  
Afficheur LCD - EEPROM - Programmation - Mémoire analogique

# SOMMAIRE

## AVANT PROPOS

## RESUME & MOTS CLES

## SOMMAIRE

## INTRODUCTION GENERALE

### CHAPITRE I : GENERALITES

I-1- Introduction.....	1
I-2- Caractéristiques de ligne téléphonique.....	2
I-3- Généralités sur les microcontrôleurs.....	4
I-3-1- Structure d'un système programmable.....	4
I-3-2- Architecture interne d'un microcontrôleur.....	6
I-3-3- Les avantages des microcontrôleurs.....	8
I-3-4- Le choix d'un microcontrôleur .....	8
I-4- Les microcontrôleurs de la famille MICROCHIP (PIC ).....	9
I-4-1- Les architectures Harvard ou Von Neumann.....	9
I-4-2- Les architectures RISC ou CISC .....	10
I-5- Les afficheurs.....	10
I-5-1 Les afficheurs à segments.....	10
I-5-2- Les afficheurs à matrice de points.....	11
I-5-3- Les afficheurs à cristaux liquides.....	11
I-6- Les mémoires.....	12
I-6-1 Les mémoires à accès aléatoire.....	12
I-6-2 Les mémoires à accès séquentiel.....	13
I-6-3 Caractéristiques principales des mémoires.....	13
I-7- Les mémoires analogiques.....	14
I-8-	
Conclusion.....	15

### CHAPITRE II : PRINCIPE DE FONCTIONNEMENT ET CHOIX DES COMPOSANTS

II-1- Introduction.....	16
II-2- Principe de fonctionnement du dispositif.....	16
II-3- Le microcontrôleur PIC 16F84.....	18
II-3-1- Les signaux disponibles .....	18
II-3-2- L'architecture interne.....	19
II-4- Le programmeur de PIC .....	24
II-5- L'afficheur à cristaux liquides.....	24
II-6- Le multiplexeur/ dé multiplexeur CD4052.....	28
II-7- La mémoire 24C01.....	29
II-8- Le décodeur DTMF SSI202 ( CD22202 ).....	29
II-9- L'encodeur DTMF TCM5089.....	29
II-10- Le décodeur de tonalité LM567.....	30
II-11- L'amplificateur opérationnel $\mu$ A741.....	30
II-12- L'amplificateur TBA820.....	30
II-13- La mémoire analogique ISD2590.....	30
II-14- Conclusion.....	30

## CHAPITRE III : PROGRAMMATION ET FONCTIONNEMENT ELECTRONIQUE

III-1- Introduction.....	31
III-2- Le système de développement de logiciel.....	31
III-3- Les modes d'adressage.....	32
III-4- Le jeu d'instruction.....	32
III-5- Algorithme.....	36
III-6- Fonctionnement électronique.....	37
III-6-1- Alimentation.....	37
III-6-2- Détection d'une alarme et prise de ligne.....	37
III-6-3- Détection et reconnaissance de la tonalité.....	39
III-6-4- Amplification des signaux DTMF.....	40
III-6-5- La mémorisation d'un message.....	40
III-6-6- Le module microcontrôleur.....	41
III-7- Conclusion.....	41



## **CHAPITRE IV : REALISTION PRATIQUE**

IV-1- Introduction.....	48
IV-2- Programmation du microcontrôleur.....	48
IV-3- Le circuit imprimé.....	48
IV-4- Implantation des composants.....	49
IV-5- Mises au point et mode d'utilisation .....	49
IV-5-1- Mises au point.....	49
IV-5-2- Mode d'utilisation.....	49
IV-5-3- Estimation des coûts.....	50
IV-6- Conclusion.....	51

## **CONCLUSION GENERALE**

## **BIBLIOGRAPHIE**

## **ANNEXES**

# INTRODUCTION GENERALE

Les télécommunications ont connu ces dernières années un progrès technologique important qui a conduit à l'introduction de nouveaux services et à l'amélioration de la qualité des services fournis. Pour assurer ces prestations nous enregistrons une grande diversité d'équipement allant de la paire de fils et les circuits simples à composants discrets jusqu'aux réseaux hauts débit et les calculateurs les plus puissants. Les bandes de fréquences dans lesquelles opèrent ces différents systèmes vont de quelques centaines de Hertz à quelques Giga Hertz.

Ce développement technologique a permis de moderniser, de sécuriser, et de permettre aux différents utilisateurs une demande plus en plus accrue de moyen de sécurité et de confort permettant la télésurveillance, les télécommandes et autres.

Dans le cadre de ce projet nous avons mis au point un équipement permettant d'émettre via le réseau téléphonique un message vocal de détresse prés enregistré dans une mémoire en cas de disfonctionnement ou d'infraction du système télé-surveillé. Ce message est envoyé à un ou plusieurs correspondants dont les numéros téléphoniques sont préprogrammés et mémorisés dans une EEPROM. La commande de l'ensemble est assurée par un microcontrôleur et l'affichage des différents états de la ligne ainsi que ceux du système télé-surveillé par un afficheur à cristaux liquides

Dans le premier chapitre, nous avons étudié les caractéristiques des lignes, les différents signaux téléphoniques et enfin nous terminons par une étude détaillée des microcontrôleurs, des afficheurs à cristaux liquides et des mémoires.

Dans le chapitre suivant nous avons présenté le synoptique de notre transmetteur, d'étudier l'architecture interne du microcontrôleur et d'étudier les différentes interfaces nécessaires à notre système d'alarme.

Dans le troisième chapitre, et après avoir pris connaissance des différents composants et outils nécessaire pour le développement des modules constituant le système d'alarme téléphonique, nous avons décrit le mode d'adressage, la programmation du microcontrôleur PIC 16F84 et le fonctionnement électronique.

Le dernier chapitre nous avons décrit les différentes étapes de la réalisation pratique du système d'alarme téléphonique. En effet, après les études théoriques et la réalisation des prototypes des différentes interfaces, les résultats des essais et tests effectués étaient concluants, ce qui nous a permis de réaliser notre module.

# CHAPITRE I : GENERALITES

## I-1- Introduction

Le réseau téléphonique constitue le support de transmission des informations émanant de notre transmetteur téléphonique d'alarme ce qui donne lieu à des contraintes dues au raccordement des terminaux sur les lignes téléphoniques. Ceci nous a conduit à prendre en considération les caractéristiques imposées par celle-ci ainsi que des caractéristiques des composants que nous allons choisir pour réaliser notre transmetteur.

Dans ce chapitre nous allons étudier les caractéristiques des lignes, les différents signaux téléphoniques et enfin nous terminons par une étude détaillée des microcontrôleurs, des afficheurs à cristaux liquides et des mémoires.

## I-2- Caractéristiques de ligne téléphonique

Au repos, une ligne téléphonique présente, entre ses deux fils, une tension continue de 48V à 50V, dont l'un des pôles est relié à la terre. Pour faire sonner le poste, le central peut superposer une tension alternative d'environ 85V/25 ou 50Hz à cette alimentation continue. Un simple condensateur permet de séparer ces deux tensions lorsque cela est nécessaire.

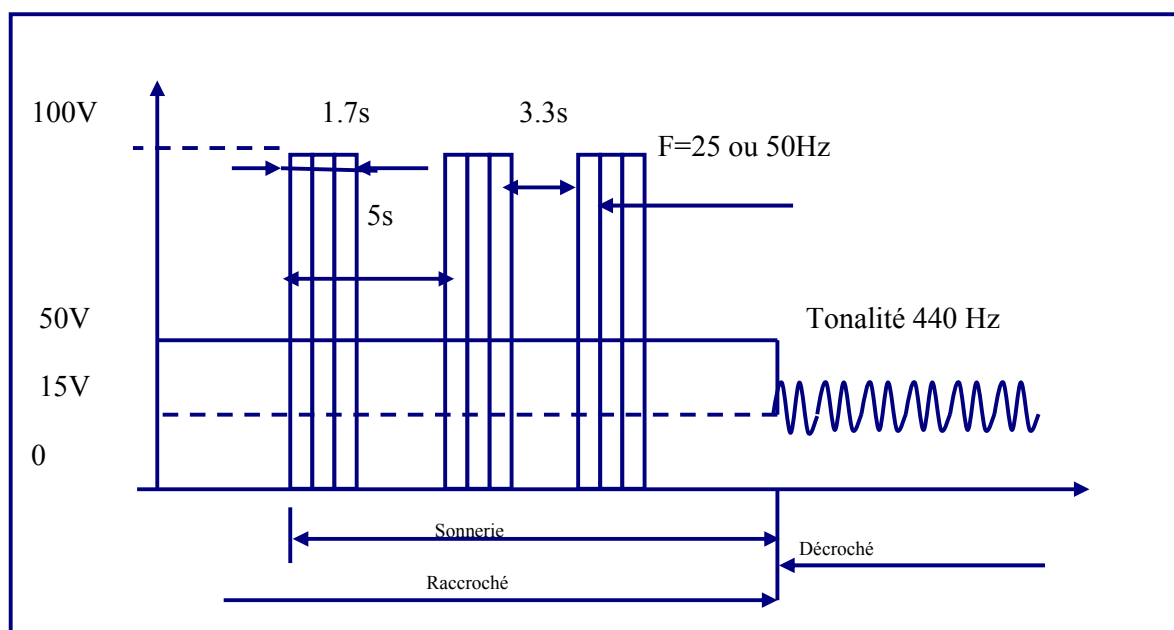
Lorsqu'on décroche le poste ( ou prise de ligne par un système automatique ), un circuit approprié consomme un courant de 35 mA environ sur la ligne, une tonalité de fréquence 440 Hz « invitation à numéroté » doit être obtenue. Elle se transforme en « occupation » si aucun numéro n'a été composé dans les vingt secondes (ligne classée « en faux appel »).

Pour numéroté, il faut utiliser un cadran ou un clavier : le cadran rotatif fonctionne par brèves coupures du courant « de boucle » de 35 mA, avec un « rapport cyclique de 33/66 ms, c'est-à-dire à une fréquence de 10 Hz. A chaque chiffre composé correspond un nombre égal de coupures, sauf pour le « zéro » qui correspond à 10 impulsions. Certains claviers, dits « décimaux », reproduisent ce fonctionnement[5].

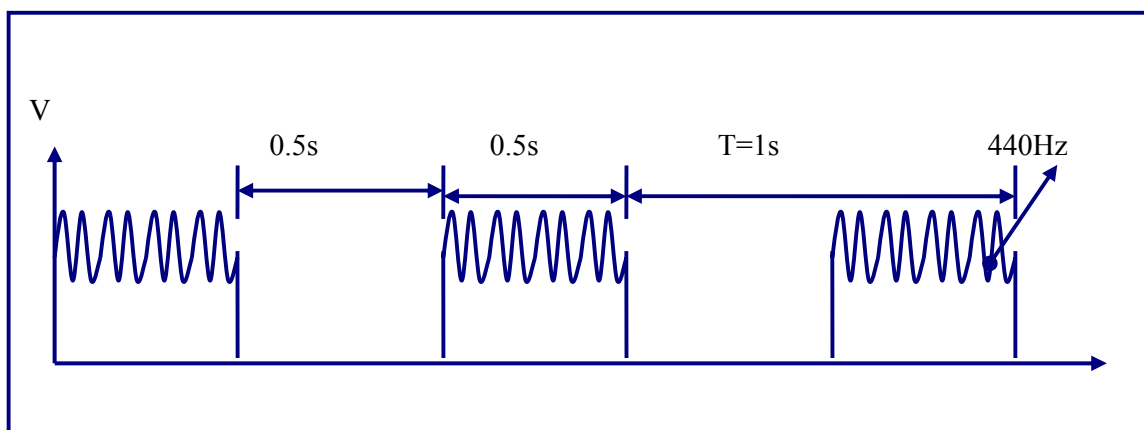
Les claviers les plus modernes sont cependant du type multifréquences (DTMF) : à chaque touche du clavier téléphonique correspond à l'addition de deux fréquences sinusoïdales dont les valeurs sont indiquées dans le tableau suivant :

Fréquence	1209 Hz	1366 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

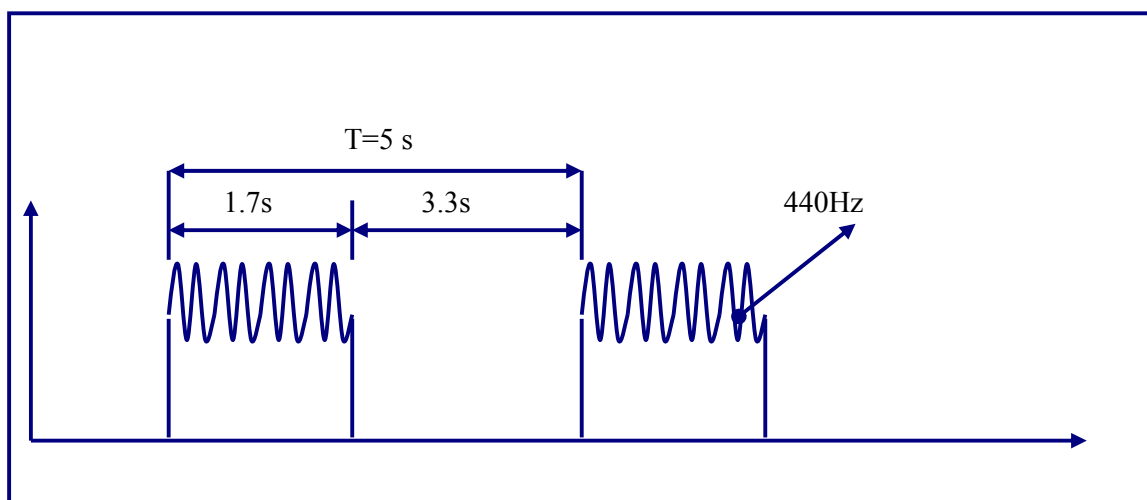
Les figures : I-1, I-2 et I-3 représentent les caractéristiques des signaux d'une ligne téléphonique :



*FIGURE I-1 : Evolution des potentiels*



*FIGURE I-2 : Signal d'occupation*



*FIGURE I-3 : Signal de retour d'appel*

## **I-3- Généralités sur les microcontrôleurs**

Le microcontrôleur est né lorsqu'on avait besoin de systèmes intelligents et moins encombrants qui sortent de la classique logique câblé à une logique programmable et souple, ainsi que, pour répondre aux besoins et aux applications domestiques et industrielles évoluées.

Directement issu de la technologie des microprocesseurs, le microcontrôleur est en fait le regroupement dans un seul boîtier de tout un système informatique : Unité centrale ( le microprocesseur), mémoires(ROM, RAM,...) et circuits d'interface ( ports E/S, timers...)

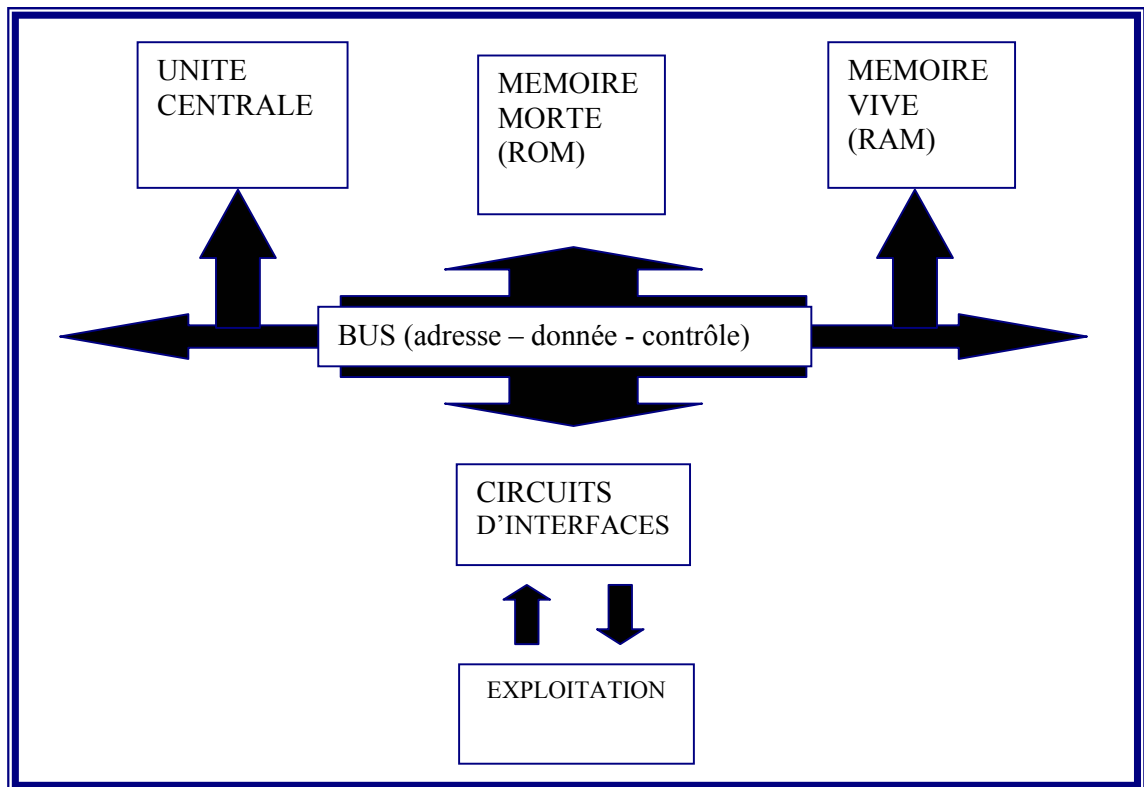
On a plusieurs types de microcontrôleurs, selon le fabricant et la famille. Chaque fabricant caractérise une famille par un noyau commun, qu'est la partie intelligente, (le microprocesseur et le jeu d'instruction); mais ces circuits se diffèrent en ce qui concerne la capacité mémoire et interfaces, tout en gardant la possibilité de s'adapter plus au moins à certains taches[1].

### **I-3-1- Structure d'un système programmable**

Tout système programmable a une structure qui contient les mêmes éléments du système informatique classique :

- L'unité centrale
- La mémoire morte (ROM)
- La mémoire vive (RAM)
- Les circuits d'interfaces
- Un bus d'interconnexion

La figure suivante représente la structure d'un système programmable :



*FIGURE I-4 : Structure général d'un système programmable*

L'unité centrale, généralement constitué par un microprocesseur plus au moins évolué, exécute le programme qui va donner vie à l'application. Le programme est contenu dans le deuxième élément qui est la mémoire morte ( ROM ). Cette mémoire peut être constituée de diverses façons : Mémoire programmé par masque, mémoire UVPROM ou EEPROM.

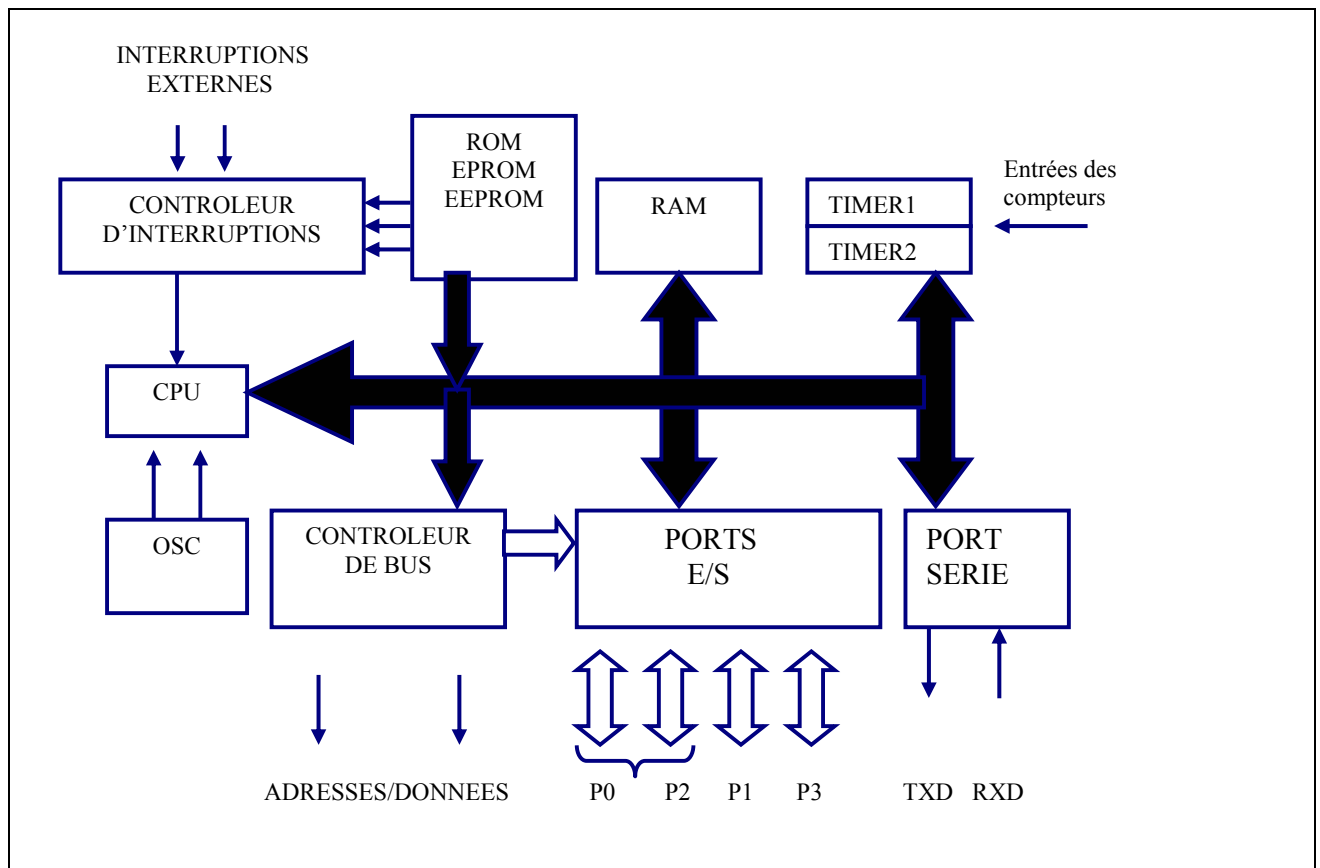
Pour pouvoir travailler correctement, le microprocesseur a souvent besoin de stocker des données temporaires dans la mémoire vive(RAM) qui est de faible taille. Le dernier élément qui assure le rôle le plus important dans une application susceptible de faire appel à un microcontrôleur est tout ce qui concerne les circuits d'interface avec l'exploitant.

Le propre d'un microcontrôleur est en effet d'intégrer, autant que possible, dans un seul boîtier, l'ensemble de ces fonctions[1].



### I-3-2- Architecture interne d'un microcontrôleur

Un microcontrôleur doit contenir dans un seul boîtier tous les éléments décrits dans la figure ci dessus. En effet, pour l'analyse des divers systèmes réalisés avant l'avènement des microcontrôleurs, les fabricants de circuit intégrés ont affiné un peu la définition de ce qu'il fallait intégrer pour arriver à un schémas type analogue à celui de la figure suivante :



*FIGURE I-5 : Architecture interne d'un microcontrôleur*

L'unité centrale a été bien souvent simplifiée par rapport à celle des microprocesseurs classiques. En contre partie, des instructions de manipulations des bits, très utile pour définir les entrées/sorties, lui ont été ajoutées. L'unité centrale peut contenir un très grand nombre de registres internes qui servent de mémoire vive. Concernant la mémoire morte, elle ne pouvait

qu'être programmée par masque. A l'utilisateur potentiel du microcontrôleur de commander donc un nombre de pièces identique très important. C'était acceptable pour une réalisation en grande série mais pas pour des fabrications limitées. Un certain nombre de microcontrôleurs sont disponible sans ROM. Puis, les technologies d'intégration progressent, les fabricants ont appris à placer sur la puce de mémoire programmable électriquement et effaçable aux ultraviolets (UVPROM) ou de la mémoire programmable et effaçable électriquement (EEPROM), on a vu également apparaître des microcontrôleurs dits OTPROM (ONE TIME PROM) programmable une seule fois.

On trouve donc à l'heure actuelle au moins cinq types différents de microcontrôleur :

- Microcontrôleur avec ROM programmable par masque
- Microcontrôleur sans aucune ROM
- Microcontrôleur avec UVPROM ou EEPROM
- Microcontrôleur avec de l'OTPROM
- Microcontrôleur avec mélange de ces combinaisons

Pour ce qui est de la mémoire vive ou RAM, tous les microprocesseurs disposent de cette mémoire interne de taille assez faible. De même on rencontre généralement les éléments de base suivants :

- Des lignes d'entrées/sorties parallèles en nombre variable selon la vocation et la taille du boîtier ( un problème de nombre maximum de pattes se posent très vite avec l'accroissement du nombre de ces lignes ).
- Au moins une interface d'entrée/sortie asynchrone, plus au moins évoluée selon les circuits
- Un ou plusieurs timers internes dont les possibilités peuvent être très variables mais qui fonctionnent en compteurs, dé compteurs, générateurs d'impulsions programmables, etc....
- Parfois, un ou des convertisseurs A/N précédés de multiplexeurs pour offrir plusieurs voies
- Parfois aussi, un convertisseur N/A[1].

### **I-3-3- Les avantages des microcontrôleurs**

Les points forts des microcontrôleurs sont nombreux et bien réels. Il suffit, pour s'en persuader, d'examiner la spectaculaire évolution de l'offre des fabricants de circuits intégrés en ce domaine depuis quelques années.

Tout d'abord, un microcontrôleur intègre dans un seul et même boîtier ce qui, avant, nécessitait une dizaine d'éléments séparés. Il en résulte donc une diminution évidente de l'encombrement et du circuit imprimé.

Cette intégration a aussi comme conséquence immédiate de simplifier le tracé du circuit imprimé, puisqu'il n'est plus nécessaire de véhiculer des bus d'adresses et de données d'un composant à un autre.

Il en découle bien évidemment une augmentation de la fiabilité du système et une diminution de la consommation.

Le microcontrôleur contribue à réduire le coût et il permet, enfin, à certaines applications de voir le jour[1].

### **I-3-4- Le choix d'un microcontrôleur**

Pour réaliser une application bien déterminer, il nous faut connaître les différents fabricants et leurs produits précis. En effet, toute la difficulté du choix d'un microcontrôleur pour une application donnée réside dans la sélection des bons circuits. Le choix du microcontrôleur est dicté par deux critères principaux :

- l'adaptation de son architecture interne aux besoins de l'application comme la présence du convertisseur A/N ou timer disposant d'un mode particulier

- le fait d'avoir ou non un système de développement. En effet, si l'on ne possède rien, on peut se laisser guider par le premier critère en comparant tous les investissements développement à prévoir. Si l'on est déjà équipé mieux vaut choisir un circuit un peu moins bien adapté[1].

## **I-4- Les microcontrôleurs de la famille MICROCHIP (PIC )**

Lors de leur première introduction sur le marché, les microcontrôleurs PIC de la société américaine Arizona Microchip, ont défrayé la chronique. En effet, s'étaient les premiers microcontrôleurs RISC utilisant de surcroît une architecture interne de type Harvard. Ces deux particularités permettent à ces microcontrôleurs de se distinguer par des vitesses de travail très importantes par rapport à leurs homologues classiques de même taille[2].

### **I-4-1- Les architectures Harvard ou Von Neumann**

Quasiment tous les microcontrôleurs actuels utilisent une architecture interne dite de Von Neumann, c'est à dire en fait une architecture comme celle que l'on rencontre encore dans la majorité des micro-ordinateurs. La mémoire dite programme contient en fait des instructions et des données l'on ne dispose que d'un bus, appelé bus de données, qui véhicule tour à tour les codes des instructions et les données qui leur sont associées. Cette architecture pose quelques problèmes dès que l'on veut faire fonctionner l'ensemble rapidement.

Il est alors préférable de faire appel à une structure dite Harvard dans la quelle les instructions et les données sont clairement différenciées et véhiculées sur des bus différents. Les résultats obtenus, en termes de vitesse d'exécution des programmes, peuvent par contre être impressionnants.

Les microcontrôleurs PIC 16FXX utilisent donc une architecture Harvard et ils font également appel à une architecture de type RISC[2].

## **I-4-2- Les architectures RISC ou CISC**

RISC (Reduced Instruction Set Computer) signifie circuit à jeu d'instructions réduit, un vrai circuit de type RISC doit en effet disposer d'une structure de pipeline qui lui permet au minimum d'exécuter une instruction alors qu'il est déjà entrain de rechercher la suivante.

Ce principe de pipeline permet encore d'accroître la vitesse d'exécution par rapport aux microprocesseurs classiques, appelés par opposition de type CISC (Complex Instruction Set Computer ). En outre, un vrai circuit RISC doit en principe exécuter toutes les instructions à la même vitesse, c'est à dire en un cycle d'horloge instruction.

Les PIC 16FXX sont donc des circuits RISC ce qui contribue à accroître encore leur vitesse de travail. Cela présente aussi un avantage pour des programmeurs potentiels, car nous n'avons plus à apprendre une centaine d'instructions différentes comme avec la majorité des microcontrôleurs actuels mais seulement 33 ou 35 instructions. Ce jeu d'instruction réduit est quasiment aussi performant, en termes de capacité de code, que le jeu au premier abord beaucoup plus complet des microcontrôleurs conventionnels[2].

## **I-5- Les afficheurs**

Il existe trois types très connus des afficheurs :

- Les afficheurs à segments
- Les afficheurs à matrice de points
- Les afficheurs à cristaux liquides

### **I-5-1 Les afficheurs à segments**

Les afficheurs à segments sont obtenus par association dans un même boîtier, d'un certain nombre de segments permettant la représentation de chiffres ou de lettres,

Les segments composant les afficheurs sont réalisés avec des diodes électro luminescentes à anode commune ou à cathode commune

## **I-5-2- Les afficheurs à matrice de points**

Chaque point est constitué d'une diode électro luminescente et la sélection de certaines de ces diodes parmi les autres permettent la représentation des caractères alphanumériques

## **I-5-3- Les afficheurs à cristaux liquides**

Les afficheurs à cristaux liquides sont constitués par un cristal se trouvant entre deux armatures réfléchit ou non la lumière ambiante s'il se trouve excité par un champ électrique convenable, on peut retenir trois principes :

### **I-5-3-1- La diffusion dynamique**

Les premiers afficheurs à cristaux liquides était réalisés suivant ce principe : On applique au cristal liquide homogène une tension supérieure à une tension de seuil à partir de la quelle se produit des distortions dans le cristal qui n'est plus transparent et diffuse alors de la lumière incidente, le contraste entre la zone transparente et la zone diffuse est alors visible.

### **I-5-3-2- Nématique en hélice**

Les afficheurs actuels utilisent ce principe qui est basé sur la polarisation de la lumière due à des forces diélectriques provoquées par le champ électrique, l'avantage principal est que la consommation est très faible mais le contraste obtenu dépend de l'angle de la vision de l'afficheur .

### **I-5-3-3- Dichroïque**

C'est un des procédés les plus récents, il est basé sur la polarisation et la réfraction de la lumière et permet suivant la tension appliquée d'obtenir de couleurs différentes.

On distingue deux types d'afficheurs à cristaux liquides :

- Les afficheurs numériques qui permettent d'afficher un ou plusieurs chiffres formés par un ensemble de sept segments
- Des afficheurs alphanumériques qui permettent d'afficher une ou plusieurs lignes de textes, les différents caractères sont formés à l'aide d'une matrice de points, généralement de cinq points sur sept

On trouve également des afficheurs graphiques formés d'une matrice rectangulaire de points, couramment des afficheurs comportant jusqu'à 680x400 points, ces afficheurs sont utilisés dans les ordinateurs portatifs et dans la télévision à écran géant[4].

## **I-6- Les mémoires**

Une mémoire est un dispositif permettant de stocker des informations ( données ), lorsqu'on veut accéder rapidement à une donnée, celle-ci se trouvera à une adresse précise dans le dispositif, on utilisera ainsi une mémoire de type accès aléatoire. Lorsque cette rapidité n'est pas nécessaire, on pourra utiliser des mémoires de types séquentiels.

### **I-6-1 Les mémoires à accès aléatoire**

#### **I-6-1-1 Les mémoires mortes**

Elles sont à lecture seule. Les informations ne sont pas perdues à la coupure de l'alimentation des circuits.

- ROM : ( Read-Only-Memory : mémoire à lecture seule ). Elle a été programmée par le fabricant de circuits intégrés par masquage et son programme ne peut être modifié.
- PROM : ( Programmable ROM : ROM programmable ). Elle est livrée non enregistrée par le fabricant. L'utilisateur programme sa PROM à l'aide d'un programmeur. Mais lorsque celle-ci est programmée, on ne peut plus l'effacer.
- EPROM : ( Erasable PROM : PROM effaçable ). Par rapport à la PROM, l'avantage est que l'utilisateur pourra s'il le désire effacer son EPROM en la soumettant à un rayonnement ultra-violet intense.

- EEPROM : ( Electrically erasable ROM : ROM effaçable électriquement ). Ce sont les mémoires mortes les plus récentes, elles sont programmables et effaçables électriquement par l'utilisateur.

### **I-6-1-2 Les mémoires vives**

On trouve des RAM ( Random Access Memory ) : Mémoire à accès aléatoire statiques ou dynamiques. Les informations seront perdues à la coupure de l'alimentation des circuits.

### **I-6-2 Les mémoires à accès séquentiel**

Dans ces mémoires on accède pas directement au mot recherché, les informations doivent défiler jusqu'à ce qu'on arrive au point recherché. C'est le cas de tous les mémoires de masse comme la bande magnétique, les disquettes et les disques durs où on utilise le principe de l'aimantation rémanente d'une zone de d'une bande magnétique.

### **I-6-3 Caractéristiques principales des mémoires**

#### **I-6-3-1 Capacité de la mémoire**

C'est le nombre total de bits que le circuit peut emmagasiner. Les données se trouvent formatées en général 1, 4 ou 8 bits.

#### **I-6-3-1 Réponses de la mémoire**

- Le temps d'accès : Temps qui s'écoule entre l'instant où l'adresse et les signaux nécessaires au cycle de lecture sont appliqués à la mémoire et l'instant où la donnée se trouve en sortie.
- Le temps de cycle : temps minimum pour effectuer un cycle de lecture et un cycle d'écriture.



## I-7- Les mémoires analogiques

Ce type de mémoire est fabriqué par la société Integrated Devices Technology, s'interface directement avec un micro électret pour l'enregistrement d'un message et de le restituer par haut parleur. On trouve quatre variantes de circuits intégrés :

- ISD1012 : Capacité d'enregistrement de 12 secondes
- ISD1016 : Capacité d'enregistrement de 16 secondes
- ISD1020 : Capacité d'enregistrement de 20 secondes
- ISD2590 : Capacité d'enregistrement de 90 secondes

Le circuit comporte une technique de mémorisation analogique, un circuit de préamplification pour le microphone, un contrôle automatique de gain, des filtres et un amplificateur de sortie[3].

*FIGURE I-6 : Structure interne de la mémoire analogique (ISD)*

## I-8- Conclusion

Le présent chapitre nous a permis d'étudier les adaptations nécessaires à prévoir dans notre transmetteur téléphonique d'alarme en tenant compte des caractéristiques des lignes téléphoniques et de différents composants.

Dans le chapitre suivant nous allons consacrer notre étude aux différentes interfaces nécessaires à l'adaptation de la ligne téléphonique aux modules constituant le transmetteur, pour cela un choix minutieux des composants à utiliser s'impose.

## **CHAPITRE II : PRINCIPE DE FONCTIONNEMENT ET CHOIX DES COMPOSANTS**

### **II-1- Introduction**

Dans ce chapitre nous allons présenter le synoptique de notre transmetteur, d'étudier l'architecture interne du microcontrôleur et d'étudier les différentes interfaces nécessaires à notre système d'alarme.

### **II-2- Principe de fonctionnement du dispositif**

Le présent transmetteur téléphonique d'alarme est destiné à fonctionner sous le contrôle d'un dispositif de détection par ouverture ou fermeture de contact ou des interfaces de détection d'incendies, présence.etc... Quel que soit le principe de sollicitation, le montage entame un cycle complet :

- Prise de la ligne téléphonique
- Composition automatique du premier numéro de téléphone préalablement programmé
- Composition automatique du second numéro de téléphone préalablement programmé si le premier s'avère occupé
- Emission d'un message d'alerte préalablement enregistré
- Libération de la ligne téléphonique

La durée globale de ce cycle est réglable, et grâce à des boutons poussoir il est possible à tout moment de faire cesser un cycle en cours.

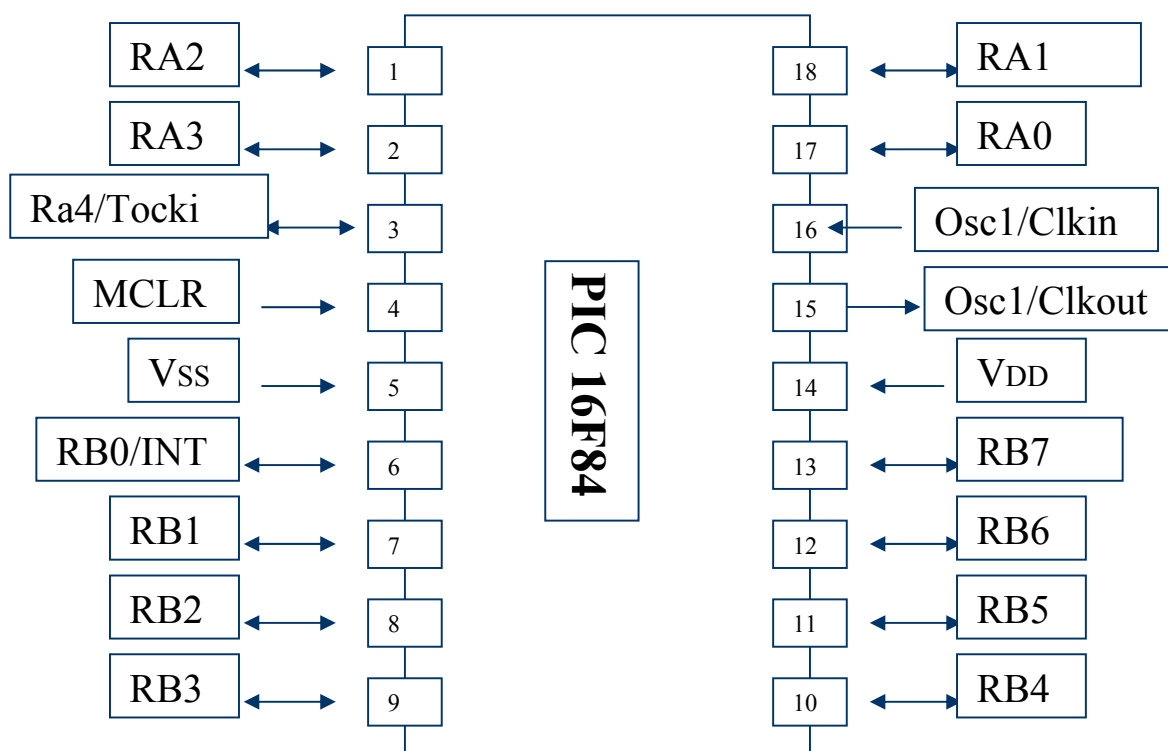


## II-3- Le microcontrôleur PIC 16F84

Le microcontrôleur PIC 16F84 est un circuit dont la mémoire est de type EEPROM, c'est à dire, programmable et effaçable électriquement. Il dispose en outre d'une mémoire EEPROM de données.

### II-3-1- Les signaux disponibles

#### II-3-1-1- Le brochage :



*FIGURE II-2 : Brochage du PIC 16F84*

### II-3-1-2- Le brochage :

- \* **Vss** et **Vdd** sont les pattes d'alimentation.
- \* **OSC1** et **OSC2** sont les pattes d'horloge.
- \* **MLCR** est l'entrée de reset.
- \* **RA0** à **RA3** sont les pattes d'entrées /sorties 0 à 3 du port A.
- \* **RA4** est la patte d'entrée/sortie 4 du port A commune avec l'entrée d'horloge externe du timer 0.
- \* **RB0** est l'entrée/sortie 0 du port B commune avec une entrée d'interruption externe.
- \* **RB1** à **RB7** sont les entrées/sorties 1 à 7 du port B. Les pattes Rb4 à Rb7 supportent en outre la fonction interruption sur changement d'état.

### II-3-2- L'architecture interne

La mémoire de programme du 16F84 est organisée en mots de 14 bits. La taille de la pile est de huit niveaux et les registres sont organisés de la même façon. Le programme counter comporte 13 bits et peut donc adresser 8 Ko de mémoire. Le vecteur de reset est placé en 0000 tandis que le vecteur d'interruption se trouve à l'adresse 0004. La mémoire de programme utilisateur proprement dite démarre à partir de 0005 et s'étend donc jusqu'à 03FF.

La figure suivante décrit l'architecture interne du PIC 16F84 :

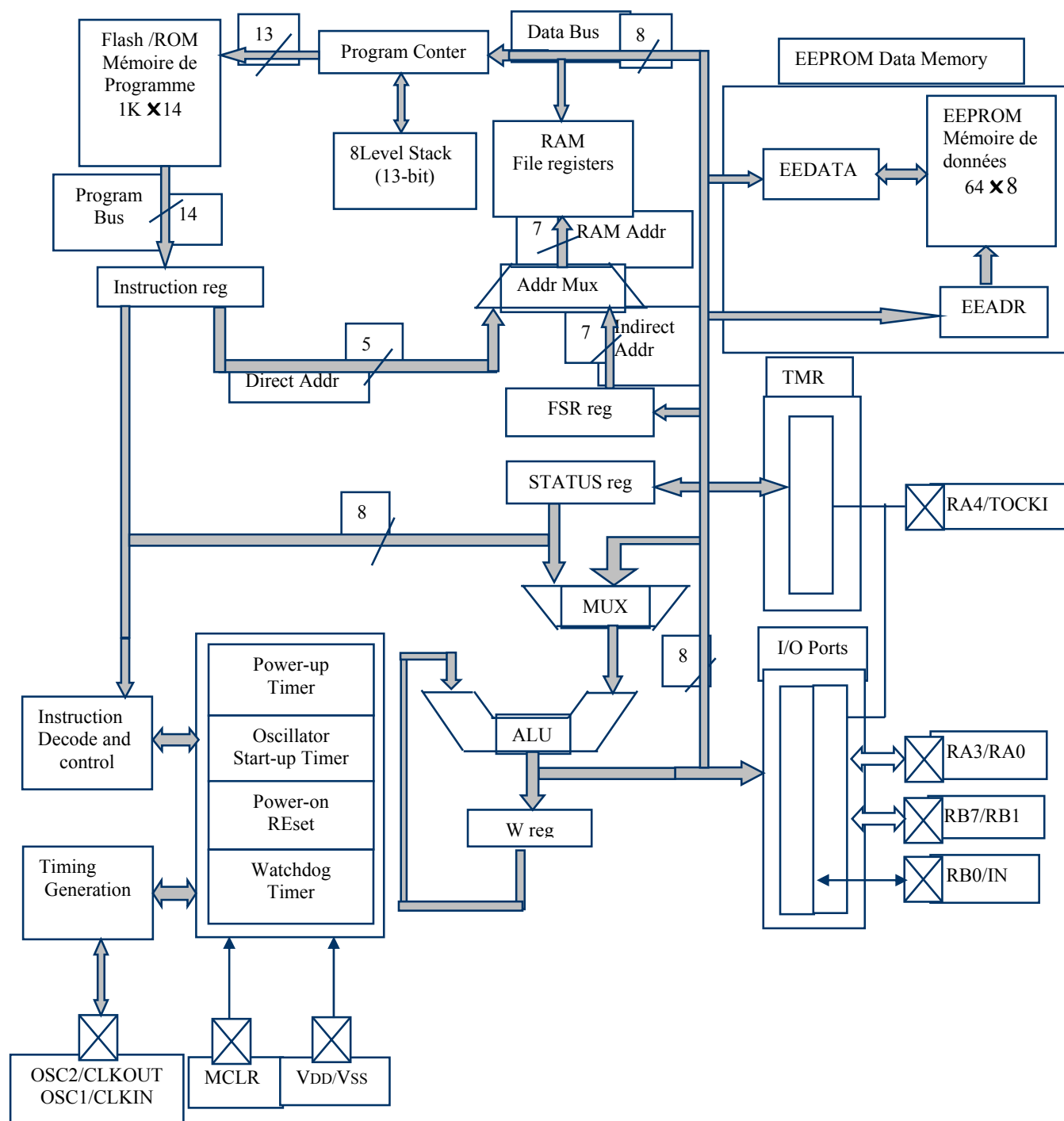


FIGURE II-3 :Architecture interne du PIC 16F84

### **II-3-2-1- Les registres internes**

Ils sont nombreux en raison de la présence de ressources internes également nombreuses. Ces registres sont accessibles de la même façon et sans restriction, ceci afin de faciliter la programmation du circuit. Ces registres peuvent être : des registres d'états, des registres d'adresses ou des registres de sélection de registres.

### **II-3-2-2- Les ports parallèles**

Le PIC 16F84 dispose de deux ports parallèles appelés port A et port B.

Les lignes de ces ports peuvent être programmées individuellement en entrées ou en sorties et s'utilisent quasiment de la même façon.

#### **II-3-2-2-1- Le port A**

Le port A dispose d'une largeur de cinq bits. Les lignes RA0 à RA3 sont munies d'un buffer de sortie CMOS. La ligne RA4 quant à elle adopte une structure un peu différente. En effet, la sortie est de type collecteur ouvert et l'entrée est munie d'un trigger de schmitt. Elle est commune avec l'entrée externe du timer 0. Le sens de travail de toutes les lignes de ce port est contrôlé par le registre TRISA dans lequel un bit à 0 place la ligne correspondante en sortie et un bit à 1 la place en entrée. Suite, à un reset, tous les bits du registre TRISA sont évidemment mis à un.

#### **II-3-2-2-2- Le port B**

Le port B est un port bidirectionnel 8 bits complets dont une seule ligne est partagée avec une autre ressource interne. Les structures internes des lignes RB0 à RB3 et des lignes RB4 à RB7 sont différentes. La raison d'être de cette différence tient au fait qu'il est possible de programmer la génération d'une interruption lors d'un changement d'état de l'une quelconque des lignes RB4 à RB7.



Toutes les lignes du port B disposent d'un pull up ou résistance de tirage au niveau haut de forte valeur. Ce pull up peut être validé ou non grâce au bit RBPU du registre OPTION du timer 0. La validation ou non de ce pull up est commune à toutes les lignes du port B. Le sens de travail de toutes les lignes de ce port est contrôlé par le registre TRISB et identique dans son principe à celui du port A.

### **II-3-2-3- Le timer**

Le timer du PIC 16F84 est constitué d'un registre 8 bits qui peut être lu et écrit. En mode timer, le registre augmente d'une unité à chaque cycle d'horloge instruction. Compte tenu de la présence dans le timer d'une circuiterie de synchronisation avec l'horloge interne du processeur, le timer ne peut plus fonctionner lorsque ce dernier est en mode SLEEP. Pour faire sortir le PIC de ce mode il génère une interruption.

### **II-3-2-4- Le timer chien de garde**

Le timer chien de garde du PIC 16F84 est muni de son propre oscillateur autonome à celle R-C interne. Il ne nécessite donc aucun composant externe et continu à fonctionner même lorsque l'horloge du PIC est arrêtée lors d'une instruction SLEEP par exemple. Si on ne souhaite pas utiliser le chien de garde, il est possible de l'invalider par écriture d'un 0 dans le bit spécifique du circuit lors de sa programmation.

### **II-3-2-5- La mémoire EEPROM de données**

Le PIC 16F84 contient une EEPROM de données de 64 octets. Cette mémoire ne faut pas partie de l'espace adressable et n'est pas accessible en lecture et écriture qu'au travers de deux registres : un registre pour les données et un registre pour les adresses. Deux registres de contrôle permettent quant à eux de définir le mode de fonctionnement de cette mémoire.

Cette mémoire EEPROM n'utilise aucune source de haute tension externe et fonctionne pour toute la plage d'alimentation autorisée pour le PIC 16F84. Sa durée typique de

programmation est de 10 ms. Elle est contrôlée par un timer interne déchargeant ainsi de toute mesure de temps par logiciel.

Le fait de vouloir écrire à une adresse déjà occupée efface automatiquement le contenu qui s'y trouve avant d'y placer la nouvelle donnée. De ce fait, aucune commande d'effacement spécifique n'est prévue.

Le principe d'utilisation de cette EEPROM en lecture est fort simple et respecte la chronologie suivante :

- écriture de l'adresse à lire dans le registre EEADR,
- mise à 1 du bit RD du registre de contrôle EECON1,
- Lecture de la donnée ainsi adressée dans le registre EEDATA.

L'écriture dans cette mémoire, qui est en fait une programmation puisque nous sommes en présence d'une EEPROM, est un peu plus complexe pour d'évidentes raisons de sécurité. Elle doit se dérouler de la façon suivante :

- ✦ Ecriture de l'adresse ou on souhaite écrire dans le registre EEADR.
- ✦ Ecriture de la donnée dans le registre EEDATA.

### **II-3-2-6- Le reset**

Le reset du PIC 16F84 peut être provoqué par plusieurs sources :

- Une mise sous tension du circuit.
- Une action sur la patte MLCR alors que le circuit est en mode normal ou en mode SLEEP.
- Un débordement du timer chien de garde alors que le circuit est en mode normal ou en mode SLEEP.

Le comportement du circuit et l'état des registres affectés par un reset sont différents selon celle de ces situations qui se produisent.

### **II-3-2-7 Les différentes sources d'interruptions**

Une interruption du PIC 16F84 peut être validée ou non par mise à 1 des bits correspondants au registre INTCON. Chaque source d'interruption potentielle dispose de son propre flag ou drapeau qui fonctionne toujours, que les interruptions correspondantes soient validées ou non. Une interruption peut : soit faire sortir le PIC du mode SLEEP, soit générer une véritable interruption s'il est en fonctionnement normal[2].

## **II-4- Le programmeur de PIC**

Ce programmeur permet de programmer une large gamme de microcontrôleurs PIC, parmi les quelles se situe le PIC 16F84. Il se connecte simplement à l'interface parallèle d'un PC. De plus, il travaille avec le logiciel "ICPROG" indispensable à sa programmation et qui peut être gratuitement télécharger à partir d'un site Internet. Ce logiciel de programmation est très simple, il s'agit en effet d'un programme à présentation pseudo graphique avec menus déroulant et boîtes de dialogue. Il permet les manipulations classiques sur tous les programmeurs de ce type : sélection du circuit à programmer, test de virginité, programmation, vérification, plus un certain nombre de fonctions annexes moins importantes.

## **II-5- L'afficheur à cristaux liquides ( Annexe VI )**

Les afficheurs à cristaux liquides sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement. Ces afficheurs disposent d'une interface parallèle dont le brochage et le nom des signaux sont indiqués dans le tableau suivant :

<b>N° de pastille</b>	<b>Appellation</b>	<b>Fonction</b>
<b>1</b>	<b>Vss</b>	<b>Masse</b>
<b>2</b>	<b>Vdd</b>	<b>Alimentation +5v</b>
<b>3</b>	<b>V0</b>	<b>Contraste afficheur</b>
<b>4</b>	<b>RS</b>	<b>Sélection commandes/données</b>
<b>5</b>	<b>R/W</b>	<b>Lecture/écriture</b>
<b>6</b>	<b>E</b>	<b>Validation des données</b>
<b>7</b>	<b>D0</b>	<b>Donnée D0 (poids faible)</b>
<b>8</b>	<b>D1</b>	<b>Donnée D1</b>
<b>9</b>	<b>D2</b>	<b>Donnée D2</b>
<b>10</b>	<b>D3</b>	<b>Donnée D3</b>
<b>11</b>	<b>D4</b>	<b>Donnée D4</b>
<b>12</b>	<b>D5</b>	<b>Donnée D5</b>
<b>13</b>	<b>D6</b>	<b>Donnée D6</b>
<b>14</b>	<b>D7</b>	<b>Donnée D7 (poids fort)</b>

*FIGURE II-4 : Description des broches de l'afficheur LCD*

L'alimentation se fait sous une tension unique de 5V et un potentiomètre ajustable permet de régler la tension, appelée  $V_o$ , qui fixe le contraste de l'afficheur.

Le fonctionnement de cette interface parallèle n'est pas complexe compte tenu de la signification des signaux, en effet :

- > **RS** permet d'indiquer au module si on lui envoie une donnée à afficher ( RS au niveau haut ) ou une commande ( RS au niveau bas ).
- > **R/W** permet d'indiquer au module si on lui envoie une donnée, c'est à dire si on écrit dans l'afficheur ( R/W au niveau bas ) ou si on lit une donnée provenant du module ( R/W au niveau haut ).
- > **E** est la ligne de validation des données. L'afficheur lisant les données ( ou fournissant des données stables lorsqu'il est en lecture ) sur le front descendant de ce signal.
- > **D0** à **D7** sont les lignes de données sur lesquelles transitent les données à l'afficheur ou les commandes transmises à l'afficheur, selon l'état de RS. Ces lignes de données peuvent fonctionner en mode 8 bits, auquel cas elles sont lues « d'un coup » lors d'un front descendant de E ou en mode 2 x 4 bits. Dans ce dernier cas, chaque mot de 8 bits est envoyé en deux fois sur D7 à D4. Le premier groupe de 4 bits correspond aux 4 bits de poids forts et le second groupe aux bits de poids faibles.
- > **Vss** c'est la masse.
- > **Vdd** alimentation +5V.
- > **Vo** contraste afficheur.

En ce qui concerne notre application nous avons choisi un afficheur LCD de 2 lignes de 16 caractères dont les différentes commandes sont présentées dans le tableau suivant :

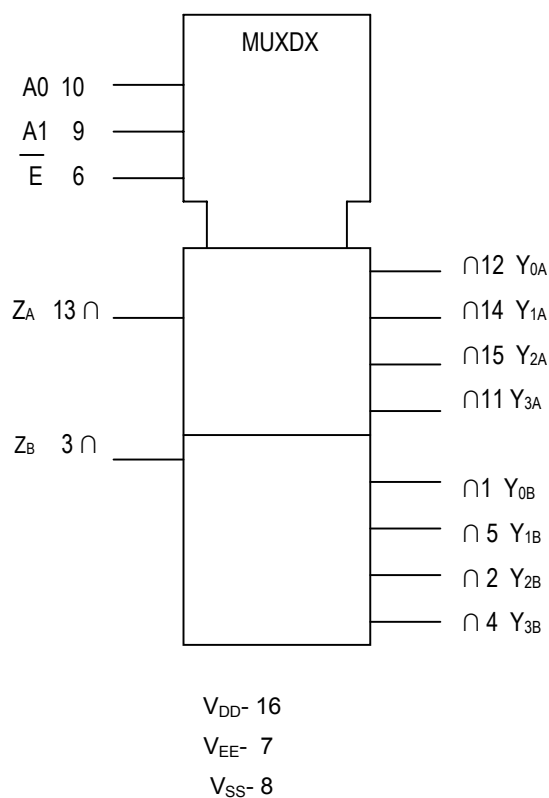
INSTRUCTION			BUS DE DONNEES								DESCRIPTION
	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0	
Effacement	0	0	0	0	0	0	0	0	0	1	Effacer la mémoire d’affichage, mais pas celle du générateur de caractère :place le curseur en position 00 (home)
Retour du curseur	0	0	0	0	0	0	0	0	1	x	Placer le curseur à la position initiale. Les décalages sont annulés. L’adresse 00 redevient la position supérieur à gauche.
Décalage du texte et(ou)curseur	0	0	0	0	0	0	0	1	ID	S	Indiquer la direction de déplacement du curseur(ID) et si le texte suit ou non le curseur (S).
Marche-Arret affichage, curseur, clignotement	0	0	0	0	0	0	1	D	C	B	Affichage D(display), curseur C ,Clignotement B (blink).
Décalage affichage ou curseur	0	0	0	0	0	1	SC	RL	x	x	Décale le curseur(S=0)ou le texte(SC=1) à gauche/droite (Right/Left.)
Initialisation	0	0	0	0	1	DL	N	x	x	x	Largeur du bus de données (Data Length) nombre de lignes utilisées (N).
Adresse du générateur du caractères	0	0	0	1	Caractère			Rangée			Indique la rongée ( de000 à111) et le caractère du où écrire les cinq bits de poids faible de l’octet de données qui suit.
Adresse mémoire	0	0	1	Adresse							Place la valeur adresse dans le compteur d’adresse, les données qui suivent sont en ASCII.
Lecture d’une adresse et du drapeau occupé, busy flag	0	0	BF	Adresse							Lit l’état de l’afficheur et l’adresse du curseur.
Ecrire de données	1	0	Données								
Lecture de données	1	1	Données								

FIGURE II-5 : Commandes de l'afficheur LCD

Le jeu de caractères d'un afficheur « LCD » est compatible ASCII ce qui facilite la tâche lors de l'élaboration du programme[6].

## II-6- Le multiplexeur/ dé multiplexeur CD4052

Le circuit intégré CD4052 renferme dans un seul boîtier deux multiplexeur/ dé multiplexeur quatre voies, il va être utilisé d'une façon qu'on peut exploiter toutes les entrées/sorties du microcontrôleur. En effet le nombre de ports du PIC16F84 est insuffisant pour gérer toutes les interfaces nécessaires pour le fonctionnement du système d'alarme[3].



*FIGURE II-6 : Structure du multiplexeur/ dé multiplexeur*

Entrées			Voie EN FONCTION		
— E	A <sub>1</sub>	A <sub>0</sub>			
L	L	L	Y <sub>0A</sub> – Z <sub>A</sub>	;	Y <sub>0B</sub> – Z <sub>B</sub>
L	L	H	Y <sub>1A</sub> – Z <sub>A</sub>	;	Y <sub>1B</sub> – Z <sub>B</sub>
L	H	L	Y <sub>2A</sub> – Z <sub>A</sub>	;	Y <sub>2B</sub> – Z <sub>B</sub>
L	H	H	Y <sub>3A</sub> – Z <sub>A</sub>	;	Y <sub>3B</sub> – Z <sub>B</sub>
H	X	X	AUCUN		

*FIGURE II-6 : Tableau de des fonctions du multiplexeur/ dé multiplexeur*

## II-7- La mémoire 24C01 ( Annexe I )

La mémoire EEPROM 24C01 assure le stockage des données et garde sa mémorisation même en cas d'absence prolongée de l'alimentation, sa capacité relativement faible est largement suffisante pour programmer deux numéros téléphoniques de huit chiffres. Ce type de mémoire utilise la liaison série pour la lecture et l'écriture des données binaires.

## II-8- Le décodeur DTMF SSI202 ( Annexe II )

Le circuit intégré SSI202 piloté par un quartz externe à 3.579545MHz en parallèle avec une résistance de 1Mohms permet la traduction sur quatre bits d'un signal DTMF reconnu conforme. Ces données vont être traitées par le microcontrôleur dont le but, est d'écriture le numéro téléphonique désiré dans l'EEPROM 24C01.

## II-9- L'encodeur DTMF TCM5089 ( Annexe II )

Il s'agit d'un encodeur DTMF, ce circuit intégré comporte des entrées lignes et colonnes. Si on relie simultanément une entrée rangée et une entrée colonne à l'état bas, la sortie délivre un signal musical ( fréquence vocale ) composé de deux fréquences sinusoïdales normalisées. La base de temps interne qui génère les fréquences DTMF avec toute la précision requise est pilotée par un quartz externe à 3.579545MHz. Les signaux d'entrées seront délivrés par le microcontrôleur.



## **II-10- Le décodeur de tonalité LM567 ( Annexe III )**

Le circuit intégré LM567 est un décodeur de tonalité. Il permet de mettre en évidence une fréquence donnée en agissant sur le réglage de la période de son oscillateur interne par l'intermédiaire des composants périphériques. Sa sortie passe de l'état haut à l'état bas lors de la présence d'une fréquence égale à celle de l'oscillateur interne sur son entrée. Il servira comme indicateur de l'état de la ligne téléphonique pour le microcontrôleur.

## **II-11- L'amplificateur opérationnel $\mu$ A741 ( Annexe II )**

Le circuit intégré  $\mu$ A741 est amplificateur opérationnel qui peut être monté en amplificateur des signaux disponibles dans la ligne téléphoniques.

## **II-12- L'amplificateur TBA820 ( Annexe IV )**

Le circuit intégré TBA820 est un amplificateur audio ( basse fréquence ) de faible puissance ( environ 1.2W ) dont le rôle consiste à donner aux signaux DTMF la puissance nécessaire pour les injecter dans la ligne téléphonique.

## **II-13- La mémoire analogique ISD2590 ( Annexe V )**

La mémoire analogique ISD2590 permette la mémorisation d'un message vocal de durée 90 secondes, ce message peut être changé à tout moment par l'utilisateur.

## **II-14- Conclusion**

Ce chapitre nous a permis de choisir le microcontrôleur PIC 16F84, vue, sur son jeu d'instructions réduit, sa rapidité d'exécution des instructions et son coût faible. La méthode et les outils nécessaires à sa programmation ainsi que le fonctionnement électronique de notre transmetteur téléphonique d'alarme seront traité dans le chapitre suivant.

## **CHAPITRE III : PROGRAMMATION ET FONCTIONNEMENT ELECTRONIQUE**

### **III-1- Introduction**

Après avoir pris connaissance des différents composants et outils nécessaire pour le développement des modules constituant le système d'alarme téléphonique, la description du mode d'adressage, la programmation du microcontrôleur PIC 16F84 et le fonctionnement électronique feront l'objet du présent chapitre.

### **III-2- Le système de développement de logiciel**

Un système de développement comporte en premier lieu un assembleur et un ou des compilateurs adaptés au langage évolué que l'on souhaite utiliser pour programmer.

L'assembleur traduit les instructions écrites en utilisant les mnémoniques du langage machine en code binaire exécutable par le microcontrôleur. La suite des mnémoniques s'appelle le listing ou code source du programme alors que le code binaire s'appelle l'objet ou l'exécutable.

Le compilateur quant à lui traduit les instructions écrites en langage évolué qui constituent aussi ce que l'on appelle le listing ou code source, en code binaire exécutable par le microcontrôleur qui constitue le code objet.

Dans un système de développement bien conçu, les deux programmes, assembleur et compilateur, doivent coexister et être utilisables l'un l'autre sans difficulté. En effet, s'il est normal de faire des calculs un tant soit peu complexe en utilisant le langage évolué, les entrées/sorties performantes et rapides ne se conçoivent, surtout dans un microcontrôleur

qu'écrites en assembleur. Il est donc primordial que le compilateur utilisé pour le langage évolué s'interface parfaitement avec l'assembleur de façon à pouvoir écrire des sous-programmes en langage machine au sein même du programme en langage évolué.

Une fois le programme de l'application est écrit et assemblé ou compilé sur une machine, on est en possession d'un binaire exécutable.

### **III-3- Les modes d'adressage**

Le premier mode, qui est en général appelé immédiat par la majorité des fabricants, est celui dans lequel la donnée manipulée par l'instruction est codée avec l'instruction elle-même. La donnée en question s'appelle ici un intervalle.

On rencontre ensuite un mode d'adressage que l'on pourrait appeler direct. En effet, la mémoire RAM est en fait divisée en registres spécifiques et en un ensemble de registres à usage général, ce mode consiste donc à coder le nom du ou des registres concernés directement dans l'instruction.

On dispose également, d'un mode d'adressage de type bit qui permet de manipuler un bit individuel dans n'importe quel registre. Il est à noter que ce mode d'adressage ne s'utilise jamais seul mais est toujours couplé avec le mode d'adressage direct.

Enfin, le dernier mode d'adressage qu'est dit le mode le plus puissant, est le mode indirect qui consiste en l'adressage à un registre par l'intermédiaire d'un autre.

### **III-4- Le jeu d'instruction**

Chaque instruction suivie de sa description est traitée dans ce qui suit, notant que chacune est codée sur 14 bits pour le cas du PIC 16F84.

La notion adoptée pour les données et adresses manipulées par les instructions est la suivante :

- f représente un numéro de registre, codé sur sept bits dans le cas du 16F84.
- B représente un numéro de bit, codé sur trois bits. Le bit 0 est toujours le bit de poids le plus faible comme il se doit en notation logique cohérente.
- K représente une donnée, son codage en nombre de bits est variable selon l'instruction.

### ▸ **Description des instructions**

- ◆ **ADDWF** : ajoute le contenu de W et le contenu de F et place le résultat dans W.
- ◆ **ANDLW** : effectue un ET logique entre le contenu de W et le littéral K et place le résultat dans K ; ( K représente une donnée ).
- ◆ **ANDWF** : effectue un ET logique entre le contenu de W et le contenu de F et place le résultat dans W.
- ◆ **BCF** : met à zéro le bit numéro b de F ( le numéro est codé sur 3 bits ).
- ◆ **BSF** : met à un le bit numéro b de F.
- ◆ **BTFSC** : si le bit numéro b est nul, l'instruction qui suit celle-ci est ignorée, donc le BTFSC demande deux cycles pour s'exécuter.
- ◆ **BTFSS** : si le bit numéro b de F est à un, l'instruction qui suit celle-ci est ignorée, le BTFSS demande deux cycles pour s'exécuter.
- ◆ **CALL** : sauvegarde l'adresse de retour sur la pile puis appelle le sous-programme situé à l'adresse chargée dans le PC.
- ◆ **CLRF** : met le contenu de F à zéro.
- ◆ **CLRW** : met le contenu de W à zéro.
- ◆ **CLRWDT** : met à zéro le registre de comptage du timer chien de garde ainsi que le pré-diviseur.
- ◆ **COMF** : complémente F bit à bit, le résultat est placé de nouveau dans F.
- ◆ **DECF** : diminue le contenu de F d'une unité, le résultat est placé de nouveau dans F.

- ♦ **DECFSZ** : diminue le contenu de F d'une unité, le résultat est placé à nouveau dans F. Si le résultat est nul l'instruction suivante est ignorée et, dans ce cas, cette instruction dure deux cycles.
- ♦ **GOTO** : appelle le sous programme situé à l'adresse chargée dans le PC.
- ♦ **INCF** : augmente le contenu de F d'une unité, le résultat est placé à nouveau dans f.
- ♦ **INCFSZ** : augmente le contenu de F d'une unité, le résultat est placé à nouveau dans F. Si le résultat est nul l'instruction suivante est ignorée, dans ce cas, cette instruction dure deux cycles.
- ♦ **IORLW** : effectue un OU logique inclusif entre le contenu de W et littéral K et place le résultat dans W.
- ♦ **IORWF** : effectue un OU logique inclusif entre le contenu de W et contenu de F et place le résultat dans F.
- ♦ **MOVF** : déplace le contenu de F dans f ou dans W.
- ♦ **NOP** : ne fait que consommer du temps machine ( un cycle dans ce cas ).
- ♦ **OPTION** : charge le registre OPTION avec le contenu de W.
- ♦ **RETFIE** : charge le PC avec la valeur qui se trouve au sommet de la pile assurant ainsi un retour d'interruption, cette instruction dure deux cycles.
- ♦ **RETLW** : charge W avec le littéral K puis charge le PC avec la valeur qui se trouve au sommet de la pile effectuant ainsi un retour de sous-programme.
- ♦ **RETURN** : charge le PC avec la valeur qui se trouve au sommet de la pile effectuant ainsi un retour de sous-programme, cette instruction dure deux cycles.
- ♦ **RLF** : rotation à gauche d'un bit du contenu de F en passant par le bit de retenue C. Le résultat est placé dans F ou W suivant l'indication du bit d.
- ♦ **RRF** : rotation à droite d'un bit du contenu de F en passant par le bit de retenue C. Le résultat est placé dans F ou W suivant l'indication du bit d.
- ♦ **SLEEP** : place le circuit en mode sommeil avec arrêt de l'oscillateur.
- ♦ **SUBLW** : soustrait le contenu de W du littéral du K et place le résultat dans W, soustraction par la méthode du complément à deux.
- ♦ **SUBWF** : soustrait le contenu de W du contenu de f et place le résultat dans W, soustraction par la méthode du complément à deux.
- ♦ **SWAPF** : échange les quatre bits de poids forts avec les quatre bits de poids faibles de f et place le résultat dans f.

- ♦ **TRIS** : charge le contenu de W dans le registre TRIS du port f.
- ♦ **XORLW** : effectue un OU logique exclusif entre le contenu de W et le littéral K et place le résultat dans W.
- ♦ **XORWF** : effectue un OU logique exclusif entre le contenu de W et le contenu de f et place le résultat dans f[2].

## **III-5- Algorithme**

## III-6- Fonctionnement électronique

### III-6-1- Alimentation ( Figure III - I )

L'énergie est prélevée du secteur 220V par l'intermédiaire d'un transformateur abaisseur de tension qui délivre sur son enroulement secondaire un potentiel alternatif de l'ordre de 12 V. Un pont de diode redresse les deux alternances tandis que la capacité C1 réalise un filtrage pour produire un courant légèrement ondulé de 12 à 15V, aussitôt acheminé sur l'entrée d'un régulateur 7812 qui délivre sur sa sortie un potentiel continu, stabilisé à 12V. Après filtrage par la capacité C4, cette tension est par la suite appliquée à l'entrée d'un régulateur 7805 pour délivrer un potentiel de 5V. Les capacités C3 et C6 apportent un complément de filtrage par contre C2 et C5 sont des capacités de découplage. La LED L2 dont le courant est limité par la résistance R3 témoigne la mise sous tension du dispositif.

Une coupure du secteur 220V provoque le blocage du transistor T1 comportant sur son circuit collecteur le bobinage d'un relais 2RT REL1. L'état de repos correspond à la mise en service du batterie de secours signalé par l'allumage du LED1 monté en série avec la résistance R2. la diode D1 protège le transistor des surtensions liées au phénomène de self qui se manifestent lors des coupures.

### III-6-2- Détection d'une alarme et prise de ligne (Figure III - II )

La porte NAND A de CI-I dans sa position de repos a ses deux entrées soumises à un état haut permanent. L'entrée 2 est reliée à un état haut par l'intermédiaire du contrôle d'ouverture, tandis que l'entrée 1 est soumise à un état haut par le biais de R6. Il en résulte un état bas de repos, ou de veille sur la sortie de cette porte. Si l'une ou l'autre de ces deux entrées sont soumises même brièvement à un état bas soit par rupture de contrôle d'ouverture ou fermeture ou encore par l'appui sur BP1, un état haut devient disponible sur la sortie 3 de la porte NAND A DE CI-I et pris en compte par le circuit dérivateur C8, R8 et D2 pour produire une brève impulsion positive correspondant à la charge de C8 à travers R8.



Les portes NOR A et B de CI-II forment une bascule monostable. A l'état de repos, une telle bascule présente sur sa sortie un état bas permanent, tandis que la sortie de la porte A est à l'état haut. Dès que la bascule reçoit une impulsion positive sur son entrée de commande par le dispositif de dérivation C8, R8, la sortie de la bascule passe à l'état haut et restera dans cette situation pendant une durée proportionnelle au produit  $(VAR1 + R9) \times C9$ , c'est la durée du cycle d'alarme.

En appuyant sur BP2 on soumet l'entrée 6 de la porte NOR B de CI-II à l'état haut, la sortie passe aussitôt à l'état bas et la bascule monostable se place ainsi prématurément sur sa position de repos. Le même phénomène se produit au moment de la mise sous tension du montage, suite à une coupure de courant secteur par exemple. En effet grâce à la charge de C10 à travers R10 une impulsion positive apparaît sur l'entrée 6 de la porte B de CI-II. Cette disposition assure la mise en position de repos de la bascule de manière automatique afin que les instabilités qui se créent au moment de l'établissement de l'alimentation ne viennent pas générer accidentellement un cycle d'alarme.

Le transistor T2 comportant sur son circuit collecteur le bobinage d'un relais 2RT REL1 est commandée par l'état haut délivré par la bascule. La fermeture de REL2 est signalée par la LED3 montée en série avec R12 et protégée des surtensions liées aux phénomènes de self qui se manifestent lors des coupures par la diode D3.

Dès que le relais REL2 se ferme, la ligne téléphonique voit ses deux polarités réunies par une résistance VAR2. le potentiel passe alors de 50V, caractéristique de la ligne libre, à une valeur de l'ordre de 10V ce qui correspond à une ligne prise.

Aussi la fermeture du REL2 provoque le déclenchement d'une sirène contrôlée par l'interrupteur INT1.

### III-6-3- Détection et reconnaissance de la tonalité (Figure III - III )

Le circuit intégré référencé CI-III est un 741 monté en amplificateur. Les signaux disponibles dans la ligne téléphonique sont ( DTMF et tonalités ) acheminés sur l'entrée inverseuse par l'intermédiaire de C11 et de R13. La résistance variable VAR3 réalise le contre réaction nécessaire au fonctionnement en amplificateur fermé. Le gain s'exprime par la relation  $G=(VAR3 + R16)/R13$ .

Le pont formé par les résistances R14 et R15 fixe le potentiel de repos de sortie de l'amplificateur à une valeur de l'ordre de 2.5V et grâce à la variable VAR3 il est possible de régler le gain en potentiel de cet étage. Le transistor T3 est monté en émetteur commun et polarisé de façon à présenter sur son collecteur un potentiel nul en l'absence des signaux.

Les signaux disponibles à la sortie de CI-III se trouvent sur le collecteur de T3 sous la forme des créneaux. Les portes B et C de IC-I montées en trigger de Schmidt, délivrent sur la sortie des créneaux dont les fronts ascendants et descendants se caractérisent par une allure verticale. Les résistances R22 et R23 constituent un pont diviseur et on dispose alors de créneaux dont l'amplitude est calibrée à une valeur de 0.3V imposée par les caractéristiques d'entrées de CI-IV ( LM567 ) qui a pour rôle de mettre en évidence une fréquence donnée.

L'ajustable VAR4 et la capacité C17 sont à la base d'une période d'oscillations internes définie par la relation  $T= 1.1 \times VAR4 \times C17$ . Lorsque la fréquence du signal d'entrée (440Hz) acheminé par l'intermédiaire de C14 est la même que celle qui est déterminée par le réglage du curseur de VAR4, la sortie passe à l'état bas. Pour une fréquence différente, cette sortie reste à l'état haut imposé par la présence de R24. La diode \* témoigne des différents états de la sortie de LM567, pris en compte par le microcontrôleur pour l'exécution de ses tâches ( programmation du numéro téléphonique, contrôlée par l'interrupteur INT2 et d'assurer le bon déroulement du cycle d'alarme).

### **III-6-4- Amplification des signaux DTMF (Figure III - IV )**

Les signaux DTMF générés par le circuit intégré TCM5089 sont recueillis par l'ajustable VAR6, dont le curseur permet de prélever une fraction plus ou moins importante de l'amplitude. Par l'intermédiaire de C20, ces signaux sont introduits dans l'entrée de CI-V qui est un amplificateur audio de faible puissance ( TBA820 ). Ce dernier est piloté par quelques composants externes qui stabilisent et règlent son fonctionnement. Ainsi la capacité C22 stabilise le phénomène de réjection, tandis que C25 assure la compensation de fréquence.

Le gain propre de l'amplificateur est fixe grâce à R28 et C21, il est possible de régler l'importance de l'amplitude du signal de sortie en agissant sur le curseur de l'ajustable VAR6. Les signaux DTMF ainsi amplifiés sont injectés dans la ligne téléphonique par l'intermédiaire de C24, qui bloque également les composantes continues de la ligne.

### **III-6-5- La mémorisation d'un message (Figure III - V )**

La mémoire vocale constituée par le circuit intégré CI-VI permet l'enregistrement d'un message désiré de durée allant jusqu'à 90 secondes, les signaux enregistrés sont injectés dans la ligne téléphonique par l'intermédiaire d'un transformateur assurant une isolation galvanique de rapport 1/1. La fonction enregistrement ou lecture est choisie avec l'interrupteur INT4, et déclenchée par le bouton poussoir BP3. Si on souhaite arrêter avant que la fin de la mémoire disponible ne soit atteinte, on appuie de nouveau sur BP3. Tout appui ultérieur sur ce dernier fera alors repartir la lecture ou l'enregistrement de cet endroit. Si par contre on veut arrêter la lecture ou l'enregistrement et redémarrer au début de la mémoire lors de la prochaine utilisation, on appuie sur le bouton poussoir BP4. Tout appui ultérieur sur BP3 fera alors repartir la lecture ou l'enregistrement au début de la mémoire.

Dans le cas où plusieurs messages ont été enregistrés les uns à la suite des autres ( actions répétées sur BP3 ), ces messages sont considérés comme indépendants les uns des autres et en mode lecture, chaque action sur BP3 les fera lire tour à tour. L'écoute des messages enregistrés est possible grâce à la fermeture de l'interrupteur INT3.

### **III-6-6- Le module microcontrôleur (Figure III - VI )**

Le composant principale de ce module est le microcontrôleur PIC16F84, piloté par un quartz de 4 KHz, assure la gestion d'affichage, la communication série avec l'EEPROM est la génération des signaux de commande aux différents éléments constituant le système d'alarme.

En entrée sur ses ports nous trouvons les quatre multiplexeur/ dé multiplexeur commandés par les ports RB6 et RB7 pour gérer tour à tour : L'encodeur DTMF, le décodeur DTMF, l'afficheur LCD de 2 lignes de 16 caractères commandé en mode 4 bits avec son contraste réglable grâce à une résistance variable de 22 K $\Omega$  et la mémoire EEPROM. Le fonctionnement électronique de ce module assure les tâches suivantes :

- Programmation de deux numéros à huit chiffres à partir d'un poste téléphonique : les données à traiter sont fournies par le décodeur DTMF ( SSI202)
- Sauvegarde des données dans une EEPROM 24C01
- Vérification des deux numéros téléphoniques programmés
- Gestion de l'affichage
- Détection d'alarme et le test de l'état de la ligne téléphonique ( occupation ) : signaux disponibles à la sortie de CI-IV (LM567 ) : Port RA4
- Exécution des commandes de la numérotation à travers l'encodeur DTMF ( TCM5089 )
- Commande de la libération de la ligne téléphonique : Port RB5
- Commande de la prise de la ligne téléphonique : Port RB4
- Commande de la mémoire vocale ( ISD2590 ) : Port RA1
- Interruption en cours d'exécution des tâches (RESET manuelle grâce à un bouton poussoir BP5).

### **III-7- Conclusion**

L'étude détaillée de la programmation du microcontrôleur et du principe de fonctionnement des différents étages électroniques sont indispensables à la réalisation pratique et la mise au point de notre système téléphonique d'alarme qui vont faire l'objet du dernier chapitre.

## **CHAPITRE IV : REALISTION PRATIQUE**

### **IV-1- Introduction**

Le présent chapitre décrit les différentes étapes de la réalisation pratique du système d'alarme téléphonique. En effet, après les études théoriques et la réalisation des prototypes des différentes interfaces, les résultats des essais et tests effectués étaient concluants, ce qui nous a permis de réaliser notre module.

### **IV-2- Programmation du microcontrôleur ( Annexe VII )**

Pour programmer le micxrôcontroleur PIC16F84 , nous avons utilisé l'diteur de texte DOS pour écrire le listing et le sauvegarder dans un fichier nommé ALARME.SRC, par la suite il vient l'assemblage et le sauvegarde de ce dernier à l'aide du Logiciel SPASM ( logiciel d'assemblage pour les microcontroleur de la famille MICROCHIP ).

Le fichier hexadécimale obtenu (ALARME.HEX) est ensuite transferer dans la zone code du microcontrôleur PIC16F84 moyennant le logiciel ICPROG par l'intermédiaire d'un un programmeur connecté au port parallèle de l'ordinateur.

### **IV-3- Le circuit imprimé ( Annexe VIII )**

Après avoir saisie les pistes moyennant le logiciel EAGLE, la planche constituant le tracé du circuits imprimé est par la suite tiré par une imprmente sur un papier translucide et interposé entre une source de rayons ultraviolets et l'epoxy présensibilisé.

Après révélation dans un bain de soude suivi d'un rinçage, le module est plongé dans un bain de perchlorure de fer pour gravure. Lorsque cette opération est achevé, ce dernier est lavé avec de l'eau. Ensuite, toutes les pastilles seront à percer. à l'aide d'un foret de 08mm de diamètre. Certains seront à agrondirpour les adapter au diamètre des connexions des composants aux quels ils sont destinés.

## **IV-4- Implantation des composants ( Annexe IX )**

Après la mise en place des straps de liaison, on a passé à l'implantation et la soudure des résistances, des capacités, des supports de circuits intégrés et des autres accessoires telle que les ajustables les relais, etc.

## **IV-5- Mises au point et mode d'utilisation**

### **IV-5-1- Mises au point**

En branchant le 220V, en appuyant sur BP1 on règle la durée totale du cycle d'alarme à la valeur de 3minutes en agissant sur le curseur de l'ajustable VAR1. et après avoir branché le montage sur la ligne téléphonique dès le début du processus, on tourne le curseur de l'ajustable VAR3 jusqu'à l'obtention d'une puissance suffisante de signal de tonalité qui sera ensuite détecté par le décodeur de tonalité en agissant sur VAR4. La LED\* témoigne la présence du signal ce qui facilite la calibration de l'oscillateur interne du CI-IV ( LM576 ) sur la fréquence de 440 Hz. Et pour avoir une puissance de signaux DTMF de valeur suffisante pour le fonctionnement correcte du module, on agit sur les curseurs des ajustables VAR5 ( DTMF vers le décodeur: SSI202 ) et VAR6 (DTMF vers amplificateur : TBA820 ). Quant au curseur de l'ajustable VAR2, sa position est le résultat de l'adaptation convenable du montage à la ligne téléphonique.

### **IV-5-2- Mode d'utilisation**

Le système d'alarme téléphonique est branché en parallèle avec un poste téléphonique, et alimenté par le secteur 220V et une batterie de secours de 12V. Un contact à ouverture est installé sur une porte ou une fenêtre connecté directe à l'entrée de détection du système.

La LED verte allumée témoigne la mise sous tension du système

La LED jaune allumée indique que la batteries de secours est en service

La LED rouge allumée indique la détection d'alarme et prise de ligne

BP1 : simulation d'alarme

BP2 : annulation d'alarme

BP3 : déclenchement de lecture ou enregistrement de message

BP4 : arrêt de lecture ou enregistrement de message

BP5 : reset du microcontrôleur

INT1 : sirène active ou silence (A/S)

INT2 : programmation des numéros téléphoniques (PROG/ALARM)

INT3 : vérification du message (HP/ALARM)

INT4 : enregistrement ou lecture (LECT/ENRG)

La programmation des numéros téléphoniques est choisie par l'interrupteur INT2 ( PROG/ALARM ) avant le décrochage du poste téléphonique, et la composition de la commande suivante : **\*#numéro à huit chiffres#**, ( Affichage : < NOVEAU NUMERO>)

Pour vérifier les numéros téléphoniques, on compose la commande suivante :

**\*#1#** pour savoir le premier numéro programmée, ( Affichage : < XXXXXXXX>)

**\*#2#** pour savoir le second numéro programmée, ( Affichage : < XXXXXXXX>)

XXXXXXXX : numéro à huit chiffres

Pour l'enregistrement d'un message d'alerte, la fonction enregistrement ou lecture est choisie avec l'interrupteur INT4 (LECT/ENRG), et déclenchée par le bouton poussoir BP3, et l'écoute du message est possible, soit à travers le haut parleur en choisissant la position de l'interrupteur INT3 sur ( HP/ALARM), soit à travers le combiné du poste téléphonique décroché.

#### **IV-5-3- Estimation des coûts**

Un facteur important qui a beaucoup d'influence sur le processus de développement de tout projet est le financement. L'étude de prix de revient représente un élément déterminant dans la conception du produit. Sachant que l'objectif du concepteur est de trouver un compromis entre l'espérance d'un produit de bonne qualité et la réduction du prix de revient, nous avons veillé,

durant l'étape de la conception à réduire au maximum le prix de revient tout en conservant la chance d'aboutir à une réalisation de bonne qualité.

D'après le tableau suivant, l'estimation du coût de la réalisation du dit système d'alarme téléphonique est d'environ : 155.340 DT TTC

Composant	Qtée	PU	PT TTC	Composant	Qtée	PU	PT TTC	Composant	Qtée	PU	PT TTC
Diode LED	5	0.080	0.400	Relais 2RT	2	6.500	13.000	EEPROM 24C01	1	4.000	4.000
Resistance	41	0.040	1.640	Transistor	3	0.300	0.900	TBA820	1	0.600	0.600
Capacité	36	0.250	9.000	Support CI	14	0.400	5.600	LM567	1	1.200	1.200
Diode	7	0.200	1.400	Régulateur	2	0.800	1.600	ISD2590	1	20.000	20.000
Bouton poussoir	5	0.700	3.500	Afficheur	1	30.000	30.000	µA741	1	0.400	0.400
Interrupteur	4	1.100	4.400	PIC16F84	1	10.000	10.000	Quartz	3	1.500	4.500
Haut parleur	1	1.500	1.500	TCM5089	1	5.000	5.000	Circuit imprimé	1	10.000	10.000
Micro-electret	1	1.000	1.000	SSI202	1	5.000	5.000	Divers	----	----	10.000
Pont de diodes B80	1	0.700	0.700	CD4052	4	0.800	3.200	-----	----	----	-----
Transfo 220/12V/1A	1	3.000	3.000	CD4001	1	0.400	0.400	-----	----	----	-----
Transfo r :1/1	1	3.000	3.000	CD4011	1	0.400	0.400	-----	----	----	-----
TOTAL 1 DT TTC			29.540	TOTAL 2 DT TTC			75.100	TOTAL 3 DT TTC			50.700
TOTAL GENERAL DT TTC								155.340			

*FIGURE IV-1 : Tableau des composants*

## IV-6- Conclusion

Le fonctionnement correcte de notre système d'alarme ne justifie pas son raccordement sur le réseau téléphonique de Tunisie Télécom, que suite à un agrément de l'opérateur ( Tunisie Télécom ), en revanche, aucune restriction, si le branchement se réalise sur un réseau privé.



# CONCLUSION GENERALE

Le présent rapport est rédigé dans le cadre de la préparation d'un projet de fin d'études à l'Ecole Supérieure des Communications de Tunis, en vue de l'obtention de diplôme d'Ingénieurs des Travaux de l'Etat en Télécommunications.

L'élaboration de ce travail a nécessité l'assimilation et la mise en œuvre de plusieurs disciplines tels que l'étude théorique du réseau téléphonique, des microcontrôleurs, des mémoires, des afficheurs, la réalisation des montages électroniques et la programmation.

En effet, nous avons étudié, et réalisé un transmetteur téléphonique d'alarme permettant d'émettre via le réseau téléphonique un message vocal de détresse prés enregistré dans une mémoire en cas de disfonctionnement ou d'infraction du système télé-surveillé. Ce message est envoyé à un ou plusieurs correspondants dont les numéros téléphoniques sont préprogrammés et mémorisés dans une EEPROM. La commande de l'ensemble est assurée par un microcontrôleur et l'affichage des différents états de la ligne ainsi que ceux du système télé-surveillé par un afficheur à cristaux liquides.

Dans ce cadre, nous avons mené en premier lieu une recherche bibliographique des principales caractéristiques de la ligne téléphonique, les microcontrôleurs, les afficheurs, et les mémoires.

En second lieu, on a étudié la programmation et le fonctionnement d'un microcontrôleur de la famille MICROCHIP et les principaux composants nécessaires à la réalisation de système.

Enfin, et grâce à la définition de l'architecture générale du module et les solutions adéquates, nous avons pu réussir la réalisation pratique du transmetteur téléphonique d'alarme objet de notre **Projet de Fin d'Etudes**.

## **BIBLIOGRAPHIE**

- [1]** Christian Tavernier  
Les Microcontrôleurs  
Dunod . Paris 1991
- [2]** Christian Tavernier  
Les Microcontrôleurs PIC, description & mise en œuvre  
Dunod. Paris 1998
- [3]** Christian Panetto  
Guide pratique des systèmes logiques  
ETSF. Paris 1986
- [4]** Claude Gallès  
Astuces et méthodes électroniques  
ETSF. Paris 1998
- [5]** Stéphane Lahier et Dominique Présent  
Transmission et réseaux  
Dunod. Paris 1999
- [6]** C.Cimelli et R.Bourgeron  
Guide du technicien en électronique  
Hachette. Paris 1999