



HAL
open science

Systeme de diagnostic et de paramétrage à distance d'un ascenseur

Achraf Nasser Eddine

► **To cite this version:**

Achraf Nasser Eddine. Systeme de diagnostic et de paramétrage à distance d'un ascenseur. Electronique. 2013. dumas-01261622

HAL Id: dumas-01261622

<https://dumas.ccsd.cnrs.fr/dumas-01261622>

Submitted on 25 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



MEMOIRE

présenté en vue d'obtenir

le **DIPLOME d'INGENIEUR CNAM** **en Electronique**

Réalisé par

Achraf NASSER EDDINE

Systeme de diagnostic et de paramétrage à distance d'un
ascenseur

Soutenu le 06 Février 2013

JURY

PRESIDENT : M. Michel TERRE

MEMBRES : M. Khaled ITANI

M. Dany MERHEJ

M. Haissam HAJJAR

M. Mohamad ALWAN

Remerciement

En préambule à ce mémoire, je souhaitais adresser mes remerciements les plus sincères à tout le corps enseignant pour leurs soutiens et leurs efforts au service des auditeurs du Cnam au Liban.

Je tiens à remercier le président du jury M. le Professeur Michel Terré, Directeur de l'Eicnam, pour avoir bien voulu accepter de présider le jury.

Je remercie le chef du département Génie électronique M. Khaled Itani pour les conseils et directives qu'il m'a prodigué tout au long de la réalisation de ce mémoire.

J'adresse également mes remerciements à Dr. Chaouki DIAB, Directeur Adjoint à l'ISAE qui ne s'est pas attardé à me guider par des conseils précieux.

Je tiens à remercier Docteur Dani Merhej pour l'aide qu'il m'a fournie depuis le début du travail et pour ses avis toujours éclairés, ainsi que pour son attitude très confiante envers moi. Je tiens à le remercier, tant pour sa contribution à l'étude théorique et pratique et à la présentation des idées décrites dans ce projet fin d'études, que pour son soutien et pour la confiance qu'il m'a témoignée tout au long de ce projet.

Je remercie également l'ingénieur Hassan Nasser Eddine le directeur de l'entreprise « Myzer Group » qui a permis la réalisation de ce projet au sein de sa compagnie et l'ingénieur Lary Eid chef du département technique, pour son aide technique concernant le fonctionnement de l'ascenseur.

Je remercie M. Ousama Ghanem ainsi que M. Mahmoud Amhaz, pour leur compétence et leurs dynamismes qui m'ont offert une ouverture vers le domaine de l'informatique que j'ai beaucoup appréciée.

Enfin, je remercie toute personne qui m'a, de près ou de loin, accordé la faveur de son aide dans la réalisation de ce travail. Qu'ils trouvent ici, l'expression de mon sincère gratitude et ma profonde reconnaissance.

Liste des abréviations

ARP	: Address Resolution Protocol
CCP	: Capture Compare PWM module
CERN	: Conseil Européen pour la Recherche nucléaire
CGI	: Common Gateway Interface
CMOS	: Complementary Metal Oxide Semiconductor
DDNS	: Dynamic Domain Name System
DHCP	: Dynamic Host Configuration Protocol
DMA	: Direct memory access
DNS	: Domain Name System
DSL	: Digital Subscriber Line
ECCP	: Enhanced Capture Compare PWM module
EEPROM	: Electrically Erasable Programmable Read-Only Memory
EPROM	: Erasable Programmable Read-Only Memory
FTP	: File Transfer Protocol
HTML	: Hyper Text Markup Language
HTTP	: Hyper Text Transfer Protocol
ICANN	: Internet Cooperation for Assigned Names and Numbers
IP	: Internet protocol
ISP	: Internet Service Provider
MAC	: Media access control
MCU	: Multipoint Control Unit
MIIM	: Media Independent Interface Management
MIT	: Massachusetts Institute of Technology
NCSA	: American Center for Supercomputing Applications
PHY	: Physical layer
POP3	: Post Office Protocol
RAM	: Random Access Memory
RISC	: Reduced instruction set computer
ROM	: Read Only Memory
SMTP	: Simple Mail Transfer Protocol
SSI	: Server Side Include
TCP	: Transmission Control Protocol
TPIN+	: Differential Plus Twisted-Pair Input
TPIN-	: Differential Minus Twisted-Pair Input
TPOUT+	: Differential Plus Twisted-Pair Output
TPOUT-	: Differential Minus Twisted-Pair Output
TQFP	: Thin Quad Flat Pack
UDP	: User Datagram Protocol
URL	: Uniform Resource locator

Sommaire

REMERCIEMENT	1
LISTE DES ABREVIATIONS	2
SOMMAIRE	3
INTRODUCTION	6
CHAPITRES 1 : FAMILLES D'ASCENSEURS	8
1-Historique	8
2-Types d'ascenseurs modernes	9
2.1-Les ascenseurs à traction à câbles	9
2.1.1-Avantages et inconvénients des ascenseurs à câbles	10
2.2-Les ascenseurs hydrauliques	11
2.2.1-Avantages et inconvénients des ascenseurs hydrauliques	11
3-La société Autinor	12
4-Schéma bloc du projet	13
5-Conclusion	13
CHAPITRE 2 : MICROCONTROLEUR PIC	15
1- Microprocesseurs	15
2-Microcontrôleurs	16
2.1-Structure interne d'un microcontrôleur	16
3-Avantages des microcontrôleurs	17
4-Les microcontrôleurs PIC	17
5-PIC18F87J60	18
5.1-Caractéristiques	19
5.2-Module ETHERNET	20
5.2.1- Brochage du module Ethernet	22
5.2.2-Configuration des « DELs » du réseau.	22
5.2.3-Oscillateur	22
5.2.4-Interface externe Ethernet	23
5.2.5-Registres du module ETHERNET	23
5.2.6-Registres du bloc PHY	24
6-Langage de programmation utilisé	25
6.1-Compilateurs C pour PIC	26
6.1.1-Compilateur MikroC Pro	26

7- Carte de développement utilisée	27
8-Conclusion	28
CHAPITRE 3 : SYSTEMES EMBARQUES ET RESEAUX ETHERNET	29
1-Avantages implicites de l'utilisation du réseau Ethernet	29
2-Microcontrôleurs et réseau Ethernet	30
3-Application réseau	30
4-Protocoles de contrôle de transmissions et de datagramme utilisateur	31
5-Protocole IP: adresse et routage	34
6-Adresse IP statique et dynamique	36
7-Nom de domaine	37
8-Nom de domaine dynamique	38
8.1-Configuration du DDNS	39
9-Conclusion	44
CHAPITRE 4 : REALISATION DE L'INTERFAÇAGE WEB	45
1-Langages de programmation utilisée	45
1.1-URL et HTTP	46
1.2-HTML	46
1.3-Applet Java et Java Script	46
1.4-PHP	49
1.5-MySQL	49
2-Interface web réalisée	50
2.1-Base de données	51
2.2-Formes réalisées	53
3-Conclusion	61
CHAPITRE 5 : IMPLEMENTATION DU SYSTEME EMBARQUE	62
1-Circuit d'alimentation	64
1.1-Circuit 5V	64
1.2-Circuit 3,3 V	64
2-Reset	65
2.1-Reset de la carte mère de l'ascenseur	65
2.2-Reset du PIC	66
3-Entrée Indicateur d'erreur	67
4-Interface réseau	67
5-Carte mère BG-15 de l'ascenseur et sa carte de configuration BG-17	68

5.1-Fonctionnalités de la carte BG-17 -----	69
6-Fonctionnement du Microcontrôleur -----	71
6.1-Les ports entrées/sortie-----	71
6.2-Organigramme du programme principal (main)-----	72
CONCLUSION GENERALE-----	78
ANNEXES-----	79
Annexe I : Fiche technique du microcontrôleur PIC18F87J60 -----	79
Annexe II : Offres d'un IP fixe et dynamique -----	86
1-Offre d'un IP fixa proposé par la société Cyberia -----	86
2-Offre d'un nom de domaine dynamique proposé par "DynDNS.com"-----	88
Annexe III : Fiches techniques des composants utilisés -----	89
1-Régulateur 5 V LM7805 -----	89
2-Régulateur 3 V MC33268DT-3.3 -----	90
3-Circuit ULN2803A-----	91
3- Opto-coupleur 4N35 -----	92
4- Interface réseau (connecteur Ethernet RJ45)-----	94
BIBLIOGRAPHIE-----	98
LISTE DES FIGURES -----	99
LISTE DES TABLEAUX -----	100
RESUME -----	101
ABSTRACT-----	101

Introduction

Avec la montée de l'urbanisation de notre société, l'ascenseur est devenu désormais un système indispensable pour répondre aux exigences modernes de notre vie en matière d'autonomie, de mobilité, d'accessibilité et de rapidité. Il est ainsi un élément essentiel des immeubles résidentiels, des bureaux, des musées, des aéroports, des centres de soins, des bâtiments publics, etc... Il contribue ainsi à gagner du temps, faciliter les déplacements, le transport et les courses.

De ce fait, la panne d'un ascenseur est devenue un problème majeur, surtout que dans beaucoup de situations l'ascenseur est unique dans son bâtiment pour des raisons de place ou d'économie. Limiter le temps « hors service » de l'ascenseur devient ainsi une question primordiale. L'utilisation des nouvelles technologies de communication permet de transmettre des informations en temps réel. Ces informations facilitent et accélèrent par l'analyse des données qu'elles contiennent la prise de bonnes décisions.

Dans ce contexte, l'objectif de notre projet consiste à réaliser un système embarqué d'alerte et de diagnostic préliminaire à distance d'un ascenseur, à travers une interface web sécurisé. Dès qu'une erreur est détectée, le système envoie une alerte par e-mail à un système de surveillance, indiquant l'ascenseur en panne et le numéro de l'erreur. Le personnel chargé du service pourra alors faire un diagnostic préliminaire (voire même réparer l'erreur) avant l'envoi du personnel au site. Le diagnostic préliminaire peut aussi se faire à partir de n'importe quel endroit possédant une connexion internet.

En plus, nous doterons notre projet d'une petite base de données permettant la gestion des différents ascenseurs des clients, et de faire des recherches historiques sur les erreurs par système d'ascenseur.

Le client de notre projet est la société « Myzer Group » qui est l'agent de la marque française Autinor (fournisseur d'équipements pour ascenseurs). Le premier module de notre projet sera installé chez son client « Le Mall (Zara – Sin El Fil) ».

Organisation du mémoire

Ce mémoire se compose de 5 chapitres :

Le premier chapitre donne une présentation succincte des systèmes d'ascenseurs ainsi que de la société Autinor (notre projet étant installé sur un de ses produits). Le chapitre se termine par un schéma bloc général de notre projet.

Le second chapitre décrit le microcontrôleur utilisé dans l'implémentation matérielle de notre système embarqué, un PIC18F87j60. Nous présenterons brièvement ses caractéristiques et nous détaillerons son interface réseau qui nous permettra d'implémenter un serveur web embarqué. Nous présenterons ensuite le langage de programmation adopté pour ce PIC, ainsi que la carte de développement utilisée.

Le troisième chapitre introduit les protocoles réseaux permettant la communication entre notre serveur embarqué et les réseaux Ethernet locaux et sur Internet. Nous présenterons de même la configuration nécessaire pour permettre l'accès à notre serveur embarqué selon le service DDNS (*dynamic domain name system*).

Le quatrième chapitre présente les langages de programmations de l'interfaçage web utilisé dans notre projet. Nous détaillerons ensuite l'interface web implémentée. Nous présenterons ses différentes pages (formes), leurs fonctionnalités et leurs liens avec la base de données réalisée.

Le cinquième et dernier chapitre présente l'implémentation matérielle et logiciel de notre système embarqué, comprenant un serveur web pour la communication distante, et les interfaces nécessaires pour le contrôle, l'affichage et la communication (locale) avec la carte mère de l'ascenseur.

Chapitres 1 : Familles d'ascenseurs

L'ascenseur est une invention qui date de 1852, date de la réalisation du premier ascenseur. Il est devenu ensuite, indispensable avec l'expansion des immeubles et des bâtiments.

Dans ce chapitre nous allons présenter brièvement les différentes familles d'ascenseurs en analysant leurs avantages et inconvénients. On va ensuite présenter la société Française Autinor vu que notre premier module sera installé sur un de ses produits. Le chapitre se termine par un schéma bloc général de notre projet.

1-Historique

En 1743 De Velay installa au château de Versailles et au palais Mazarin des « chaises volantes » équilibrées au moyen d'un contrepoids, que l'occupant pouvait faire monter ou descendre par sa propre force en tirant sur une corde. À la même époque est installé au Mont-Saint-Michel un monte-vivres actionné par une grande roue en bois à l'intérieur de laquelle on enfermait des prisonniers qui en assuraient la rotation.[1]

En 1853, Elisha Otis dota l'ascenseur d'un système limiteur de vitesse déclenchant un système appelé frein parachute, stoppant la cabine et assurant la sécurité des personnes en cas de rupture du câble. En 1857, il en équipa l'appareil qui pouvait transporter 450 kg (soit 5 à 6 personnes) à la vitesse très lente de 0,20 mètre par seconde dans un grand magasin à New York.[1]

Les premiers ascenseurs pour le public des gratte-ciels du tournant du XIXe siècle au XXe siècle n'autorisaient que la montée aux étages et non la descente qui devait se faire par les escaliers. [1]

Le premier immeuble résidentiel à être équipé fut le Haughwort Building à New York en 1859. [1]

2-Types d'ascenseurs modernes

On distingue de nos jours essentiellement deux types de familles d'ascenseurs (fig. 1.1):

- les ascenseurs à traction à câbles,
- les ascenseurs hydrauliques.

En règle générale, ces deux types utilisent l'énergie électrique pour déplacer les cabines verticalement (moteur électrique continu ou alternatif).

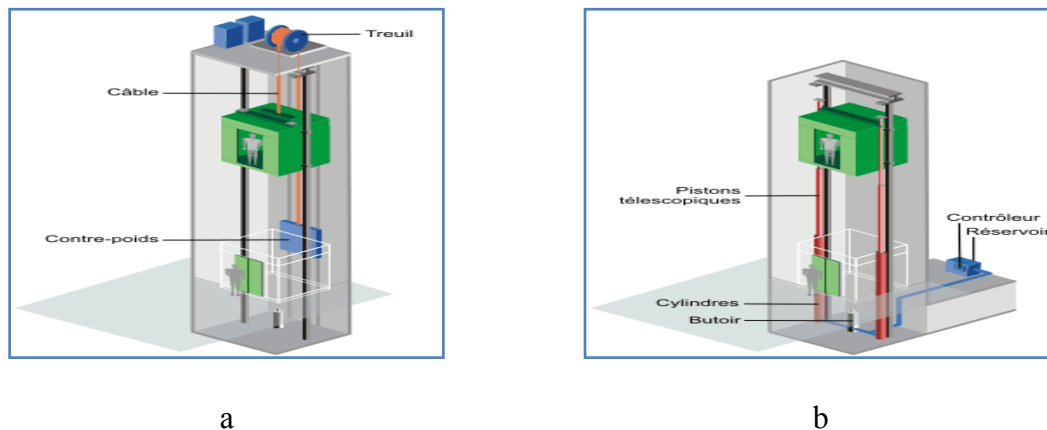


Fig. 1.1 : Types de familles d'ascenseurs. a) Ascenseur à câbles. b) Ascenseur hydraulique

2.1-Les ascenseurs à traction à câbles

Les ascenseurs à traction à câbles sont les types d'ascenseurs que l'on rencontre le plus, notamment dans les bâtiments tertiaires.

Ils se différencient entre eux selon le type de motorisation :

- à moteur-treuil à vis sans fin,
- à moteur-treuil planétaire (fig. 1.2.a),
- à moteur à attaque directe (fig. 1.2.b) (couramment appelé "Gearless" ou sans treuil),...

Quel que soit le type, les ascenseurs à traction à câbles comprennent généralement :

- une cabine,
- un contrepoids,
- des câbles reliant la cabine au contrepoids,

- des guides,
- un système de traction au-dessus de la cage de l'ascenseur,...

2.1.1-Avantages et inconvénients des ascenseurs à câbles

Ci-dessous, on cite les principaux avantages et inconvénients des ascenseurs à câbles :

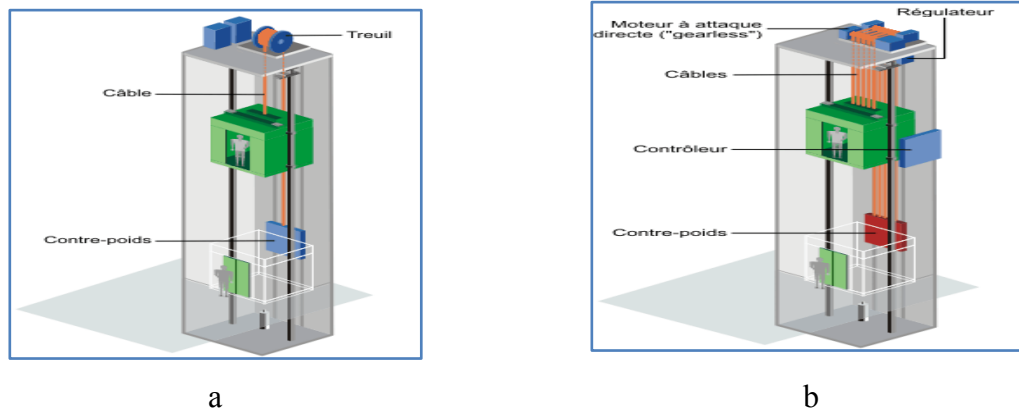


Fig. 1.2 : Ascenseurs à traction à câbles: a) Ascenseur à moteur-treuil. b) Ascenseur à moteur à attaque directe

Avantages :

- course verticale pas vraiment limitée.
- suivant le type de motorisation précision au niveau de la vitesse et du déplacement.
- rapidité de déplacement.
- bonne efficacité énergétique
- pas de souci de pollution.

Inconvénients :

- en version standard, nécessite un cabanon technique en toiture.
- exigence très importante sur l'entretien.

2.2-Les ascenseurs hydrauliques

Comme toute machine hydraulique une pompe met sous pression l'huile qui pousse un piston hors de son cylindre vers le haut (Fig. 1.3). Pendant la descente, un by-pass (vanne) permet de laisser sortir l'huile du cylindre vers le réservoir.

Les ascenseurs hydrauliques sont utilisés en général pour satisfaire des déplacements relativement courts de l'ordre de 15 à 18 mètres maximums.

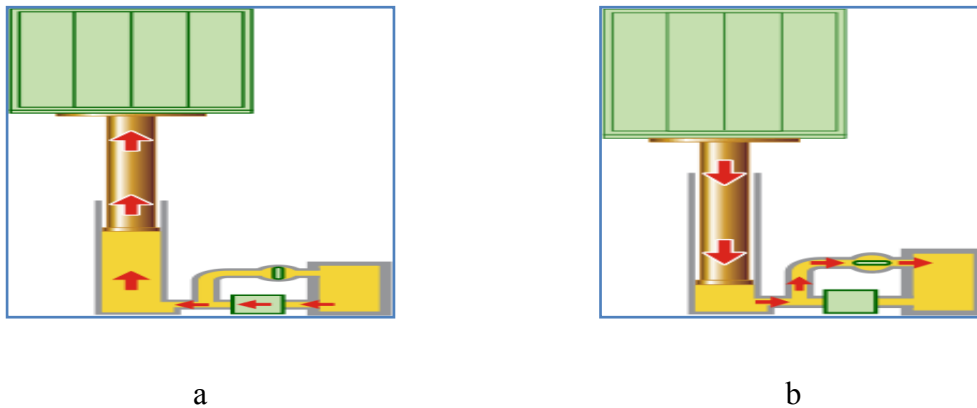


Fig. 1.3 : Principe du mécanisme d'un ascenseur hydraulique

2.2.1-Avantages et inconvénients des ascenseurs hydrauliques

Ci-dessous, nous citons les principaux avantages et inconvénients des ascenseurs hydrauliques :

Avantages :

- Grande précision au niveau du déplacement.
- Réglage facile de la vitesse de déplacement.
- Cabanon de machinerie pas nécessaire.
- Faible hauteur sous dalle (Head room)
- Implantation facile dans un immeuble existant.
- Inconvénients : Course verticale limitée à une hauteur entre 15 et 18 m.
- Risque de pollution du sous-sol.
- Consommation importante en énergie.
- Nécessité de renforcer la dalle de sol.

3-La société Autinor

Entreprise créée en 1967, Autinor consacre la première phase de son développement à la conception et la commercialisation de dispositifs de commande d'ascenseur.

Sa technologie flexible s'adapte rapidement et facilement aux réglementations locales dans chaque pays. Autinor exporte plus de 40% de son Chiffre d'Affaires.

La diversité géographique de ses clients (fig. 1.4) n'a fait qu'enrichir ses compétences, l'adaptabilité et la polyvalence de ses produits. [2]

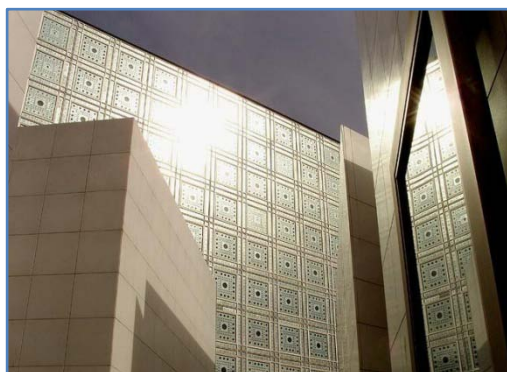


Fig. 1.4 : Diversité géographique des clients de la société Autinor

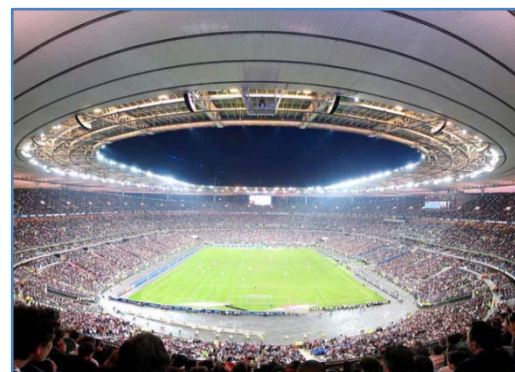
Autinor s'est engagé dans une démarche globale de développement durable. Ses produits sont aujourd'hui sélectionnés pour leur faible consommation énergétique.

Parallèlement, ils développent une gamme complète de solutions technologiques privilégiant les énergies renouvelables.[2]

L'ascenseur Autinor est utilisé dans plusieurs bâtiments mondiaux (fig. 1.5).[2]



Institut du monde arabe-Paris



Stade de France –Saint Denis



Opéra royal-Londres



Commission européenne-Bruxelles

Fig. 1.5 : Bâtiments mondiaux utilisant l'ascenseur Autinor.

4-Schéma bloc du projet

La figure 1.6 représente un schéma bloc global de notre projet. Chaque ascenseur sera relié à un de nos modules contenant un serveur web embarqué qui permet l'échange des informations entre la carte mère de l'ascenseur d'une part et l'équipe de maintenance d'autre part.

En cas d'erreur, notre module envoie un email d'alerte à un système de surveillance centralisé (responsable de maintenance), indiquant le numéro et la description de l'erreur. Le responsable de maintenance peut ainsi savoir l'erreur et faire un diagnostic préliminaire avant l'envoi du personnel de service au site.

Le diagnostic préliminaire peut aussi se faire à partir de n'importe quel endroit possédant une connexion internet (*e.g.* Smartphone, Tablet, etc.), et ceci à travers une interface web sécurisée.

En plus, notre système sera doté d'une petite base de donnée permettant la gestion des données des ascenseurs, du personnels de service, ainsi que d'effectuer des recherches datées sur l'historiques des erreurs par système d'ascenseur. Ceci peut éventuellement faciliter les diagnostics et augmenter la performance du personnel de service.

5-Conclusion

Dans ce chapitre nous avons présenté l'environnement dans lequel sera réalisé notre projet. Dans le second chapitre nous parlerons du microcontrôleur utilisé dans l'implémentation matérielle de notre système embarqué, un PIC18F87j60, du langage de programmation adopté pour ce PIC, ainsi que la carte de développement utilisée.

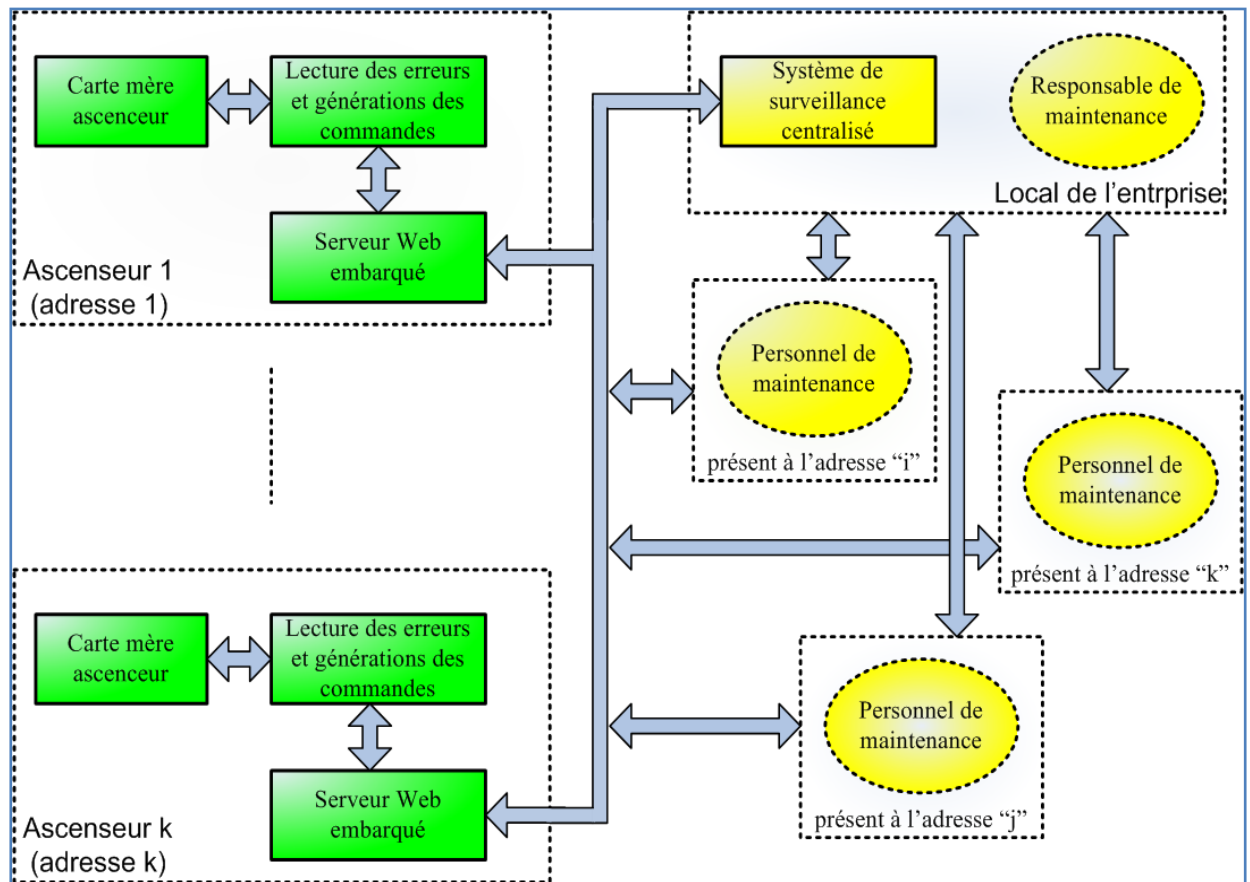


Fig.1.6 : Schéma bloc du projet.

Chapitre 2 : Microcontrôleur PIC

Les microcontrôleurs PIC appartient à une large famille de microcontrôleurs fabriquée par la société Microchip. Pour la domotique, les microcontrôleurs 8 bits ont la majorité du marché en ce moment [3]. Parmi les plus performants chez Microchip, sont ceux de la famille du PIC18FXXj60 fonctionnant à basse tension 3.3V et possédant un module Ethernet intégré permettant l'interfaçage à travers un serveur web.

Dans ce chapitre nous allons introduire brièvement l'architecture interne générale des microprocesseurs/microcontrôleur. Nous détaillerons les caractéristiques et les différents périphériques du PIC18F87j60 que nous utiliserons dans notre projet, spécialement son module Ethernet. Ensuite nous présenterons, l'environnement de développement avec le langage MikroC Pro de la société Mikroelectronika utilisant le langage C comme langage de programmation pour le pic. Ce compilateur va être utilisé pour la programmation de notre système embarqué.[3]

1- Microprocesseurs

Déjà trente ans...

La découverte du microprocesseur date aujourd'hui de près de trente ans. En effet la fabrication du premier circuit commence en 1970, où la société INTEL met au point le premier microprocesseur le 4004. [3]

Depuis, leur puissance de calcul et l'intégration des transistors qui les constituent n'ont pas cessé d'évoluer. On trouve désormais les microprocesseurs dans la plupart des applications, que ce soit, pour piloter une centrale, à l'intérieur d'un ordinateur ou bien pour remplacer le programmeur d'une machine à laver. [3]

Les microprocesseurs ne sont jamais employés seuls, mais des circuits périphériques sont toujours associés avec eux pour pouvoir être intégrés au sein d'une application. Un des avantages d'un montage à base de microprocesseur par rapport à un montage en logique câblé, est sa souplesse d'emploi. En effet, nous entrons dans le domaine de la logique programmable ou le fonctionnement du montage dépend d'un programme logé dans une mémoire, celui-ci peut être modifié pour changer les équations régissant l'application, sans toutefois entrainer de changement au niveau du câblage des entrées sorties. [3]

2-Microcontrôleurs

Les microcontrôleurs sont de véritables micro-ordinateurs intégrés sur une puce de silicium qui comportent en plus de l'unité centrale (*i.e.* le microprocesseur), de la mémoire, des ports d'entrée-sortie, des interfaces séries (e.g. RS-232), ainsi qu'une unité de gestion de temps (e.g. timers) et d'événements (e.g. interruption).

Les signaux d'entrée-sortie du microcontrôleur peuvent être facilement interfacés à des coupleurs optiques afin d'interfacer des capteurs et des actionneurs industriels.

Pratiquement tous les fabricants de microprocesseurs (Motorola, Intel, Hitachi, Texas Instrument, Toshiba, ST Microélectronique, SGS-Thomson, etc.) proposent une ou plusieurs gammes de microcontrôleurs. Les microcontrôleurs 4 bits servent essentiellement à des tâches simples. De tels microcontrôleurs sont par exemple utilisés au sein d'objets ménagers grand public, tels que des cuisinières, machines à laver ou aspirateurs. Les microcontrôleurs 8 bits sont capables de répondre à des exigences plus élevées et sont utilisés pour la commande de dispositifs informatiques tels que des joysticks, tablettes graphiques et modems. Ils sont également utilisés pour la programmation de petits robots ainsi que pour l'acquisition de données (convertisseurs A/D, etc.). Les microcontrôleurs 16/32 bits sont utilisés pour la commande de machines ou le contrôle de processus, lorsque les contraintes temps réel sont sévères ou lorsque les algorithmes de régulation nécessitent une puissance de calcul importante. Des variantes de microcontrôleurs avec canaux d'accès mémoire direct offrant un grand débit entre mémoire et entrées-sorties sont utilisés dans les applications multimédia et pour le contrôle d'imprimantes laser.

2.1-Structure interne d'un microcontrôleur

Un microcontrôleur possède au minimum les éléments suivants:

- Unité centrale, généralement constituée par un microprocesseur, a pour rôle d'exécuter le programme qui va donner vie à l'application.
- Mémoire de programme à lecture seule (e.g. ROM, EEPROM, EPROM) (contenant le programme à exécuter, cette mémoire possède la particularité de sauvegarder les informations même en absence de tension)
- Mémoire de donnée vive RAM (Random access memory) permet de stocker les informations temporaires tant que le circuit est alimenté

- Ports d'entrées-sorties permettant au processeur d'accéder au monde extérieur (interface clavier, communication série, interrupteur, etc...)
- bus système comporte les lignes permettant de relier entre eux le processeur, la mémoire et les entrées-sorties. Il comprend les lignes d'adresses provenant du processeur, les lignes de données bidirectionnelles et les lignes de contrôle.

3-Avantages des microcontrôleurs

Les points forts des microcontrôleurs sont nombreux, ils :

- Intègrent dans un seul boîtier plusieurs éléments séparés.
- Simplifient le tracé du circuit imprimé.
- Augmentent la fiabilité du système.
- Diminuent la consommation d'énergie par rapport à l'équivalent réalisé en circuit traditionnel.
- Moins cher que les composants qu'ils remplacent.

4-Les microcontrôleurs PIC

De nombreux fabricants se sont implantés sur le marché des microcontrôleurs Motorola, ST Microelectronics, WSI, etc ...

De sa part, la société américaine « Microchip technology » a mis au point un microcontrôleur CMOS : "Le PIC".

Ce microcontrôleur très utilisé à l'heure actuelle est un compromis entre simplicité d'emploi et prix de revient. Il fait partie de la famille des circuits RISC (Reduced Instruction Set Computer), caractérisée par leur vitesse d'exécution et leur jeu d'instruction réduit. [3]

Les microcontrôleurs PIC se conforment à l'architecture Harvard (Figure 2.1) : ils possèdent une mémoire de programme et une mémoire de données séparées. La plupart des instructions occupent un seul mot dans la mémoire de programme. La taille de ces mots dépend du modèle du PIC, tandis que la mémoire de données est organisée en octets.

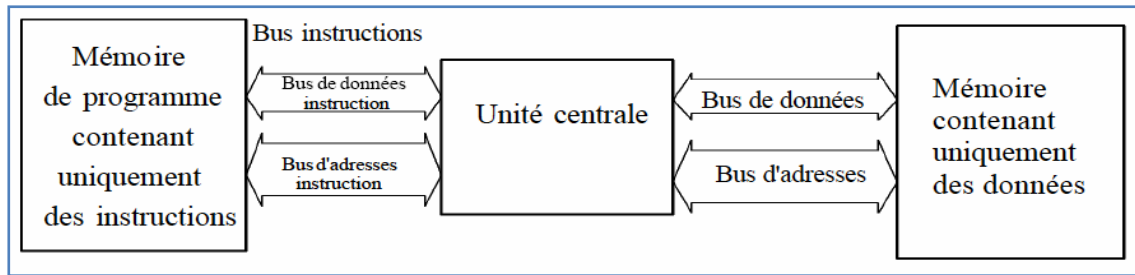


Fig.2.1 : Schéma bloc de l'architecture Harvard.

Il existe de nombreuses versions de PIC possédant chacune des caractéristiques différentes, ces versions peuvent être classifiées en trois grandes familles:

- *Base-Line*(12 bits d'instructions)
- *Mid-Range*(14 bits d'instructions)
- *High-Range*(16 bits d'instructions)

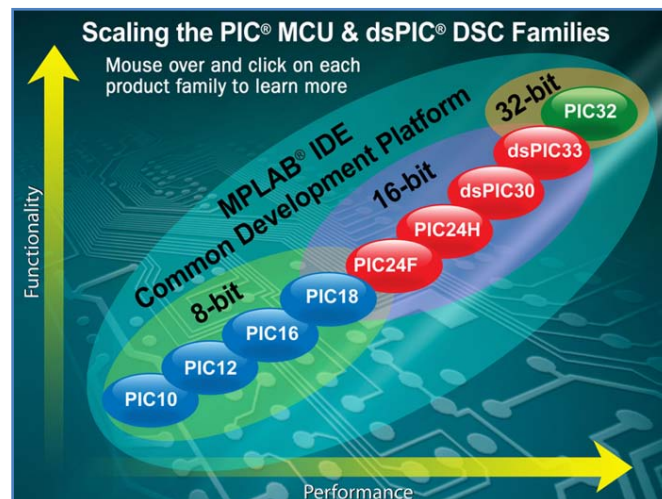


Fig.2.2 : Différentes familles de PIC[4].

Dans notre projet le PIC18F87j60 sera utilisé, il appartient à la famille PIC18FXXJ60. Les sections suivantes détaillent l'architecture de ce processeur et justifient son choix pour la réalisation de notre projet.

5-PIC18F87J60 [5]

La famille du PIC18F87j60 présente une nouvelle gamme de microcontrôleurs à basse tension. Elle présente des possibilités de calcul de haute performance et un jeu complet de fonctionnalités à un prix très compétitif. Ces caractéristiques font de cette famille un choix logique pour beaucoup des applications haut de gamme où le coût joue un rôle primordial.

5.1-Caractéristiques

Le PIC18F87j60 se présente dans un boîtier TQFP « *Thin Quad Flat Pack* » de 84 broches caractérisé par :

- Une mémoire de programme de taille 128 Koctets.
- Un compteur de programme.
- Neuf ports d'entrées-sorties.
- Une unité arithmétique et logique.
- Une mémoire RAM (3808 octets).
- Un système d'horloge.
- Un registre de décodage des instructions (75 instructions).
- Un registre d'état.
- Un registre de travail.
- Cinq modules Timer : 2×8 bit et 3×16 bit
- Deux modules CCP (Capture/Compare/PWM).
- Trois modules ECCP (Enhanced CCP).
- Des périphérique de communication : 2×EUSART (Enhanced USART)
- Un convertisseur analogique numérique : 15 canaux d'entrées multiplexés.
- Un système de Reset, MCLR.
- Un chien de garde (Watch dog).
- Un module Ethernet : 10Base-T (ce module est détaillé dans la section suivante). La présence du module Ethernet avec ses exigences d'avoir une horloge de 25 MHz nécessite la présence d'un oscillateur primaire pour fournir cette fréquence au microcontrôleur.

La figure 2.3 représente le schéma bloc de la structure interne du PIC18F87j60.

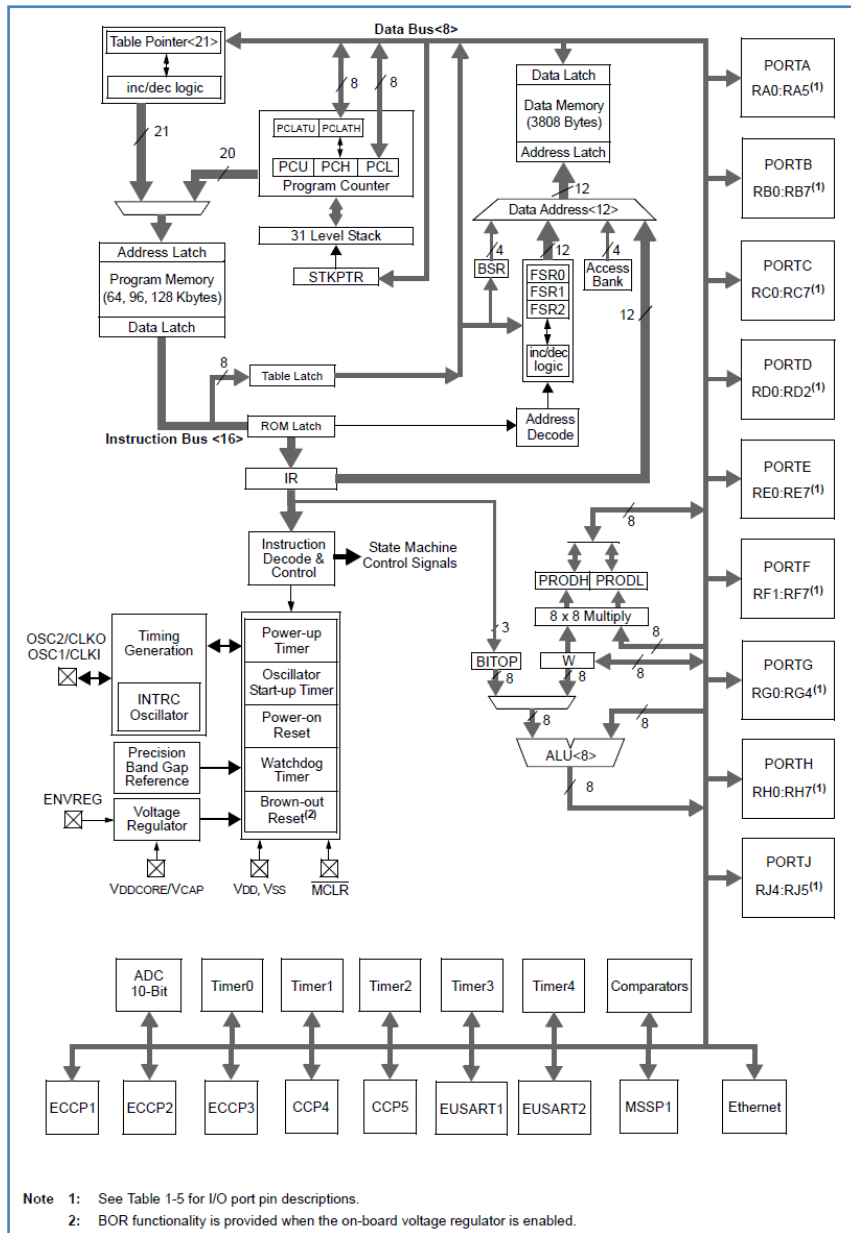


Fig.2.3 : Schéma bloc du PIC 18F87j60.

5.2-Module ETHERNET

Tous les membres de la famille PIC18FXXJ60 disposent d'un module contrôleur Ethernet intégré. C'est une solution de connectivité complète, y compris à la fois les implémentations complètes de «Media Access Control» (MAC) et de l'Émetteur-récepteur de la couche physique (PHY). Deux transformateurs d'impulsions et quelques composants passifs sont tout ce qui est requis pour connecter le microcontrôleur directement à un réseau Ethernet.

Le module Ethernet répond à toutes les spécifications de la norme IEEE 802.3 pour les connections 10-BaseT à paire torsadée. Il intègre un certain nombre de régimes de

filtrage de paquets pour limiter les paquets entrants. Il fournit également un module interne DMA «*Direct Memory Access*» pour les données rapides, et un contrôle d'erreur IP par «*checksum*». Des dispositions sont également prises pour deux sorties de LED pour indiquer l'activité de liaison et réseau. Un schéma bloc du module est donné dans la figure 2.4.

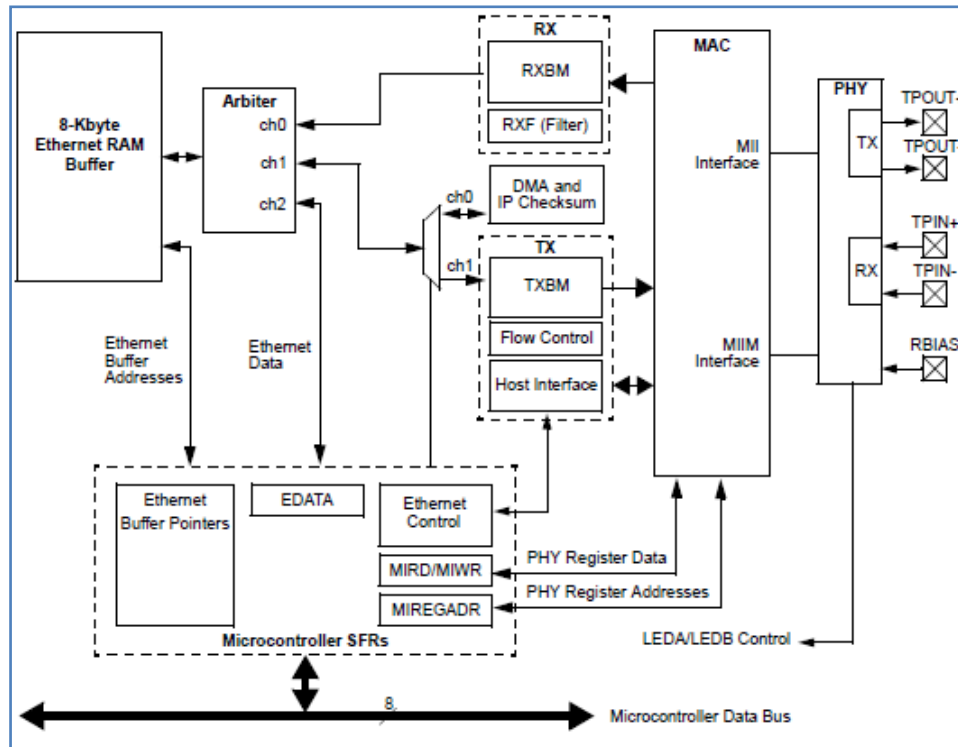


Fig.2.4 : Schéma bloc du module ETHERNET.

Le module Ethernet se compose de cinq blocs:

1. Le module émetteur-récepteur (PHY) qui code et décode les données analogiques présentes sur l'interface de paire torsadée.
2. Le module MAC qui implémente la compatibilité avec la logique MAC de l'IEEE 802.3 et fournit l'interface de gestion MIIM (*Media independent interface management*) pour contrôler la PHY.
3. Un bloc indépendant de 8-Koctet de RAM tampon pour stocker les paquets qui ont été reçus et les paquets qui doivent être transmis.
4. Un arbitre pour contrôler l'accès à la mémoire RAM lorsqu'une demande est faite à partir du microcontrôleur, de la DMA, des blocs émettre ou récepteur.
5. Un registre d'interface qui fonctionne comme un interpréteur des commandes et des signaux d'état internes entre le module et le microcontrôleur.

5.2.1- Brochage du module Ethernet

La famille des PIC18FXXj60 offrent une interface dédiée de 4-pins pour le module Ethernet. Aucun autre périphérique n'est multiplexée avec ces broches. Ces broches sont les suivantes:

- TPIN +: Entrée différentielle positive à paire torsadée.
- TPIN -: Entrée différentielle négatives à paire torsadée.
- TPOUT +: Sortie différentielle positive à paire torsadée.
- TPOUT -: Sortie différentielle négative à paire torsadée.

Le module Ethernet possède sa propre source de tension et de masse pour le module PHY. La tension est la même que VDD et VSS pour le microcontrôleur, cependant les broches ne sont pas connectées en interne. Pour que le module Ethernet fonctionne correctement, la tension positive et la masse doivent être connectées à ces broches. De même, toutes les broches d'alimentation et de masse sur le boîtier du microcontrôleur doivent être connectées de manière externe, à la même source d'alimentation, sans filtrage entre le microcontrôleur et le module Ethernet.

5.2.2-Configuration des « DELs » du réseau.

Le module PHY fournit des sorties séparées pour contrôler les indicateurs d'Ethernet, LEDA et LEDB. Ces leds sont multiplexées avec les broches RA0 et RA1 du PORTA. Leur utilisation comme sorties est activée en modifiant le bit de configuration ETHLED (registre CONFIG3H <2>).

Lorsqu'ils sont configurés en tant que sorties, RA0/LEDA et RA1/LEDB ont une capacité suffisante (jusqu'à 25 mA) pour alimenter directement les leds. Les broches doivent toujours être configurées pour fournir du courant aux leds (configuration en « source »). Les utilisateurs devraient également configurer les broches comme sorties.

5.2.3-Oscillateur

Le module Ethernet est conçu pour fonctionner à 25 MHz. Cette fréquence est fournie par l'horloge principale du microcontrôleur :

- soit avec un Crystal 25 MHz relié aux broches OSC1 et OSC2,
- ou une source d'horloge externe connectée à la broche OSC1.

5.2.4-Interface externe Ethernet

Le module Ethernet nécessite l'installation de plusieurs composants à l'extérieur. Ces composants doivent être connectés comme l'indique la figure 2.5. Sur les broches « TPIN+, TPIN- » et « TPOUT+, TPOUT- », des transformateurs d'impulsions sont obligatoires.

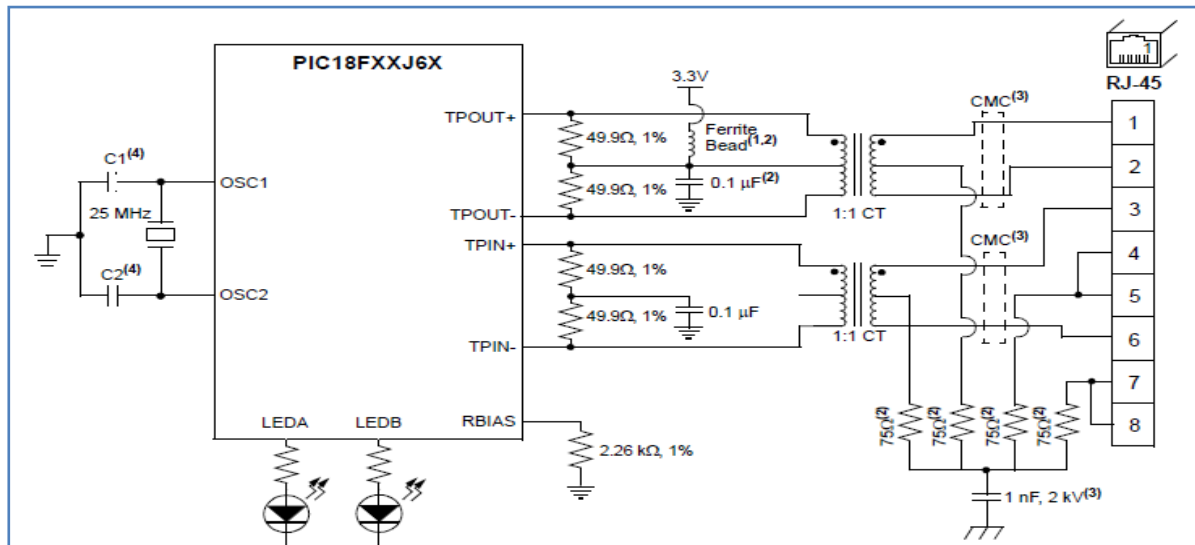


Fig.2.5 : Composants externes nécessaires pour le fonctionnement du module ETHERNET.

Les transformateurs d'impulsion devraient répondre aux exigences d'isolement de l'IEEE 802.3. Les circuits de résistances et de capacités reliés aux broches d'émission et de réception minimisent les réflexions du signal.

Des courants relativement élevés sont nécessaires pour alimenter l'interface à paire torsadée. Ainsi tous les fils doivent être aussi courts que possible, pour réduire la perte résistive. Si les lignes de données différentielles ne peuvent pas être courtes, elles doivent être acheminées de façon à avoir une impédance caractéristique de 100Ω.

5.2.5-Registres du module ETHERNET

Le module Ethernet utilise trois mémoires indépendantes pour ses opérations:

- Une RAM tampon qui stocke les données reçues en paquets afin de les préparer pour la transmission.

- Un ensemble de registres spéciales (SFRs :special function registers) 8-bits, utilisés pour contrôler le module et transmettre les données entre le module et le microcontrôleur.
- Un ensemble séparé de registres 16-bits utilisé uniquement pour le contrôle de la couche physique (PHY).

La mémoire tampon et les registres de contrôle PHY sont entièrement contenus dans le module Ethernet et ne peuvent pas être accessibles directement par le microcontrôleur. Les données sont transférées entre le microcontrôleur et le module Ethernet à l'aide de la mémoire tampon et des registres SFR. Les relations entre les registres SFRs et l'espace mémoire du module Ethernet sont présentés dans la figure 2.6.

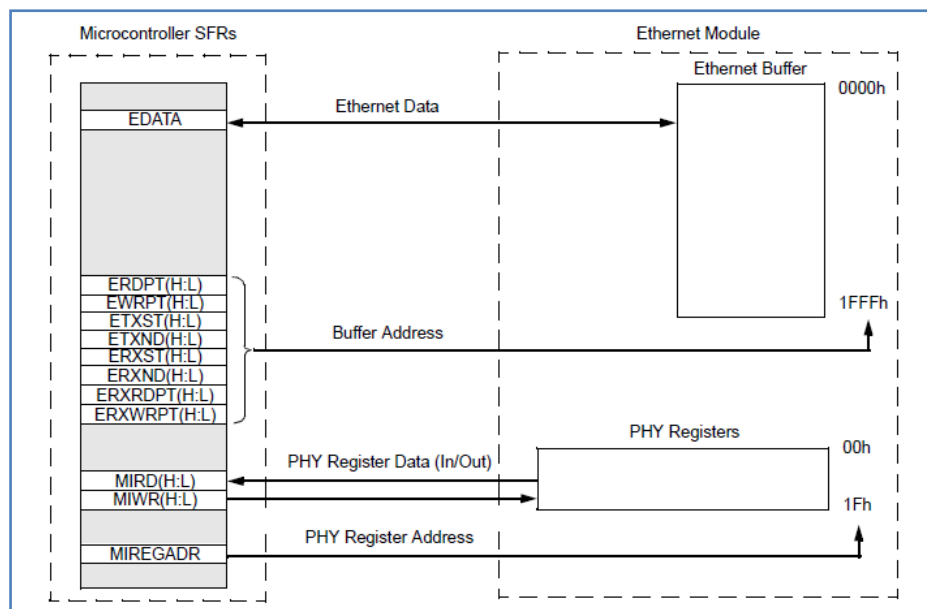


Fig. 2.6 : Relation entre les registres SFRs et l'espace mémoire du module Ethernet

5.2.6-Registres du bloc PHY

Ces registres fournissent la configuration, le contrôle et l'état de fonctionnement du bloc PHY. Tous ces registres sont de 16 bits. Ils sont accessibles avec une adresse sur 5 bits, ce qui fait 32 registres possibles. Parmi eux, seulement 7 adresses sont mises en œuvre. Ils ne sont accessibles que par l'intermédiaire de l'Interface de gestion MII.

6-Langage de programmation utilisé

Le microcontrôleur exécute le programme chargé dans sa mémoire programme. Il s'agit du code exécutable composé d'une séquence de zéros et de uns qui est organisé en 12, 14 ou 16-bit de mots, en fonction de l'architecture du microcontrôleur PIC. Chaque mot est considéré par le processeur comme une commande qui est exécutée pendant le fonctionnement du microcontrôleur. Toutes les instructions que le microcontrôleur est capable de reconnaître forment ce qu'on appelle ensemble d'instructions du microcontrôleur. [6]

Les microcontrôleurs sont programmés en langage assembleur. Les instructions du langage d'assemblage sont représentées sous formes d'abréviations significatives appelées mnémoniques, et le processus de leur transformation (passage en code exécutable) s'effectue par un programme spécial appelé assembleur. Le principal avantage de ce langage de programmation est sa simplicité, c'est à dire chaque instruction du programme correspond à une opération élémentaire relié à un emplacement dans la mémoire du microcontrôleur. Il permet un contrôle complet de ce qui se passe au sein de la puce. [6]

Cependant, la programmation en assembleur n'est pas très adaptée pour les applications de grande taille. En effet, les programmeurs préfèrent toujours un langage de programmation proche de la langue utilisée dans la vie quotidienne. Par conséquence, les langages de programmation appelés « haut niveau » ont été créés. L'un d'eux est le langage C (figure 2.7). L'avantage principal de ces langages est la simplicité de l'écriture du programme.

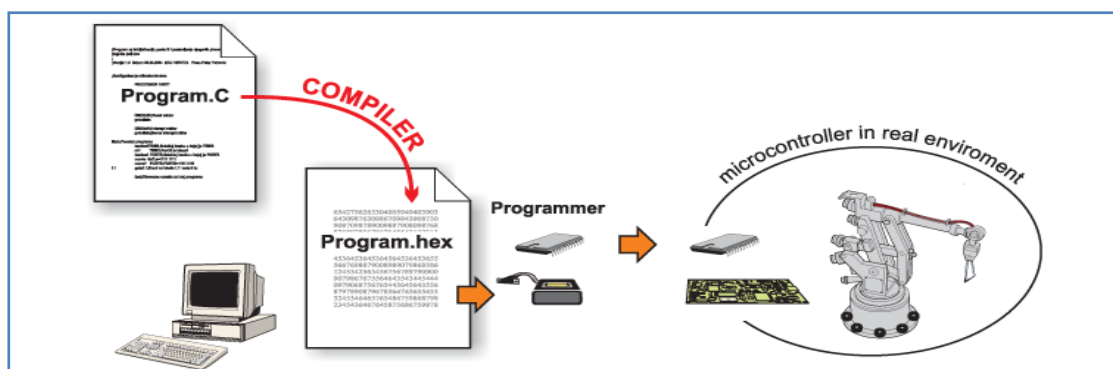


Fig.2.7:Compilation d'un programme en langage C.

6.1-Compilateurs C pour PIC

Plusieurs compilateurs en langage C existent pour programmer le microcontrôleur PIC. Ils ont été conçus par différentes constructeurs qui utilisent le pic dans leurs cartes de développement. Ci-dessous quelques compilateurs les plus utilisés :

- Mplab C18 Conçu par la société Microchip.
- CCS conçu encore par Microchip et qui est moins complexe que C18.
- MikroC Pro pour PIC conçu par la société MikroElektronika.

C'est ce dernier compilateur qui va être utilisé dans notre projet.

6.1.1-Compilateur MikroC Pro

Le mikroC PRO pour PIC (figure 2.8) est un compilateur puissant, riche en fonctionnalités et en outils de développements. Ce compilateur bénéficie d'une prise en main très facile et fournit au programmeur une solution simple pour développer ses applications pour des systèmes embarqués, sans compromettre les performances ou le contrôle.

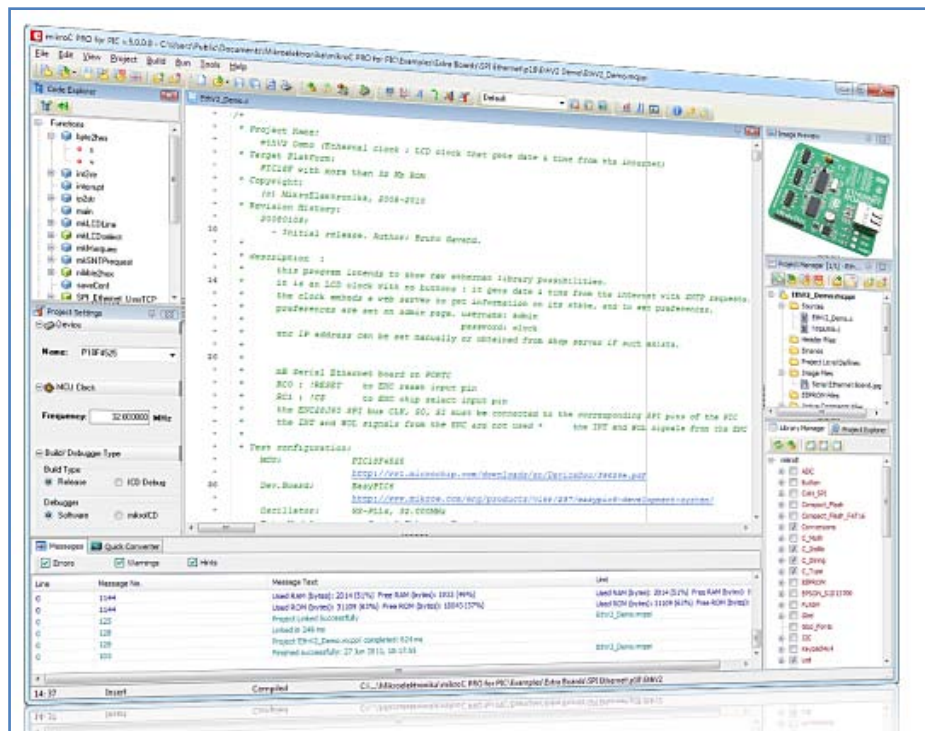


Fig.2.8 : Compilateur MikroC PRO.

Il comporte plusieurs outils intégrés (mode simulateur, terminal de communication, gestionnaire 7 segments, analyseur statistique, correcteur d'erreur, explorateur de code...) (figure 2.9). Il gère la plupart des périphériques rencontrés dans l'industrie (bus I2C, 1Wire, SPI, RS485, Bus CAN, cartes compact Flash, signaux PWM, afficheurs LCD et 7 segments...);

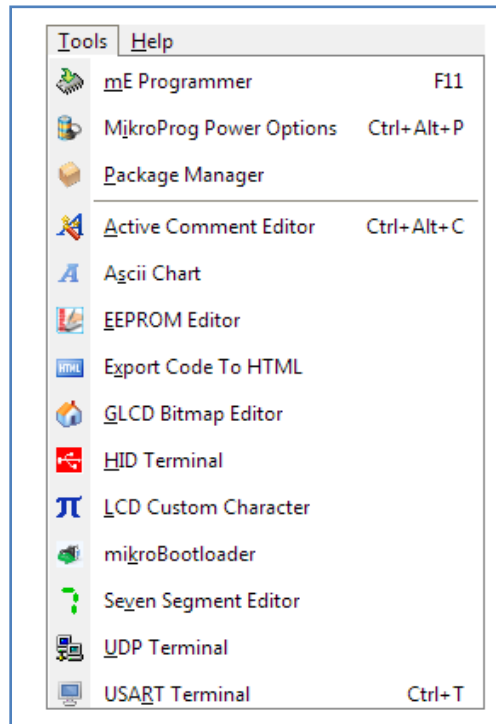
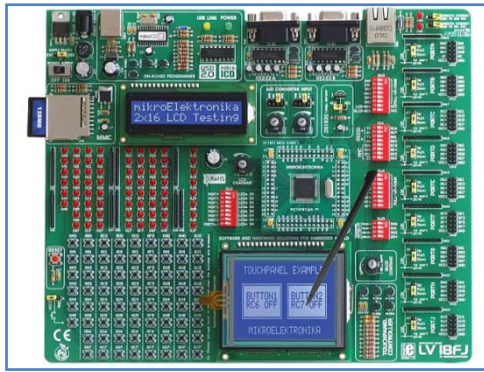


Fig.2.9 : Outils du compilateur MikroC PRO.

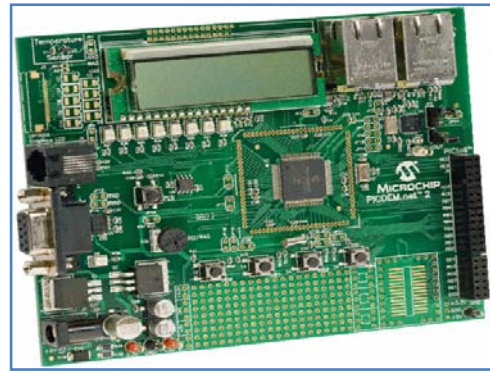
7-Carte de développement utilisée

Plusieurs cartes de développement utilisent la famille PIC18FXXj60 comme microcontrôleur et permettent la réalisation d'applications utilisant le module Ethernet. Celles qui sont les plus utilisables sont illustrées dans la figure 2.10.

Au cours de notre projet, nous utilisons la carte de développement LV18Fj. Cette carte possède un nombre important de périphériques qui nous aident à réaliser et à déboguer notre application. Elle est aussi compatible avec le compilateur MikroC PRO que nous trouvons plus pratique pour créer des applications Web que le compilateur de la société Microchip.



a



b

Fig. 2.10 : Cartes de développement comportant le module web :

a) LV18Fj conçu par MikroElektronika. b) PICDEM.net2.

8-Conclusion

Dans ce chapitre nous avons présenté brièvement le microcontrôleur PIC18F87j60 de la famille PIC18Fxxj60 et le compilateur Mikroc PRO utilisée pour la programmation de ce PIC. Nous avons présenté l'interface Ethernet du microcontrôleur et ses connexions physiques. Dans le chapitre suivant nous allons présenter les systèmes embarqués, les principaux protocoles Ethernet et la méthode de la configuration utilisée dans notre projet pour utiliser le service web Dynamic DNS.

Chapitre 3 : Systèmes embarqués et réseaux Ethernet

Ce chapitre présente succinctement les protocoles réseaux permettant la communication entre notre serveur embarqué et les réseaux Ethernet locaux et sur Internet. Nous présenterons de même les étapes effectuées pour l'obtention d'un nom de domaine dynamique ainsi que la configuration du modem DSL (*digital subscriber line*) nécessaire pour permettre l'accès à notre serveur embarqué selon le service DDNS (*dynamic domain name system*).

Un système embarqué est un dispositif qui a l'intelligence d'ordinateur et qui est dédié à l'exécution d'une tâche unique, ou un groupe de tâches liées. Les systèmes embarqués sont souvent utilisés pour exécuter des fonctions de surveillance et de contrôle. Ces systèmes sont qualifiés d'embarqués car leur code programme fait une partie intégrante du matériel.

Ethernet est la technologie de réseau utilisée dans de nombreux bureaux et maisons. Elle permet aux ordinateurs et d'autres matériels de communiquer et de partager des ressources.

Pour de nombreuses années, les systèmes embarqués et les réseaux Ethernet existaient dans des applications distinctes. Ethernet était disponible uniquement pour les ordinateurs de bureau et d'autres gros ordinateurs. Les systèmes embarqués qui avaient besoin d'échanger des informations avec ces ordinateurs, utilisaient des interfaces à faible vitesse et/ou courte portée.

Mais avec l'évolution des technologies et du marché, il est maintenant possible pour les systèmes embarqués de communiquer dans des réseaux Ethernet locaux ainsi que sur l'Internet. Les communications réseau peuvent rendre un système embarqué plus puissant et plus facile à contrôler. Un système embarqué peut ainsi héberger un site Web (web server embarqué), envoyer et recevoir des e-mails, envoyer et télécharger des fichiers et échanger d'informations avec d'autres ordinateurs connectés via une interface réseau.

1-Avantages implicites de l'utilisation du réseau Ethernet

- Ethernet est le réseau le plus largement déployé dans les bureaux et les bâtiments résidentiels ou industriels. Ainsi Ethernet est omniprésent.

- Ethernet assure la facilité de l'infrastructure, de l'interopérabilité et l'évolutivité de développement de notre module.
- Une fois que l'équipement est connecté à un réseau Ethernet, il peut être surveillée ou contrôlée par l'Internet.

2-Microcontrôleurs et réseau Ethernet

Grace à son faible coût, sa grande largeur de bande, sa stabilité, sa sécurité et sa compatibilité à travers différents dispositifs, Ethernet est devenu le standard de facto de l'accès au réseau pour les microcontrôleurs 32, 16 et même 8-bits.

Avec le réseau Ethernet, les microcontrôleurs PIC peuvent répondre aux besoins du marché de l'Ethernet intégré. Comme le présente le schéma bloc de la figure 3.1, un microcontrôleur PIC et un contrôleur Ethernet permettent de relier n'importe quel dispositif au *World Wide Web*. Ils contribuent ainsi à la surveillance, au contrôle des dispositifs sur Internet.

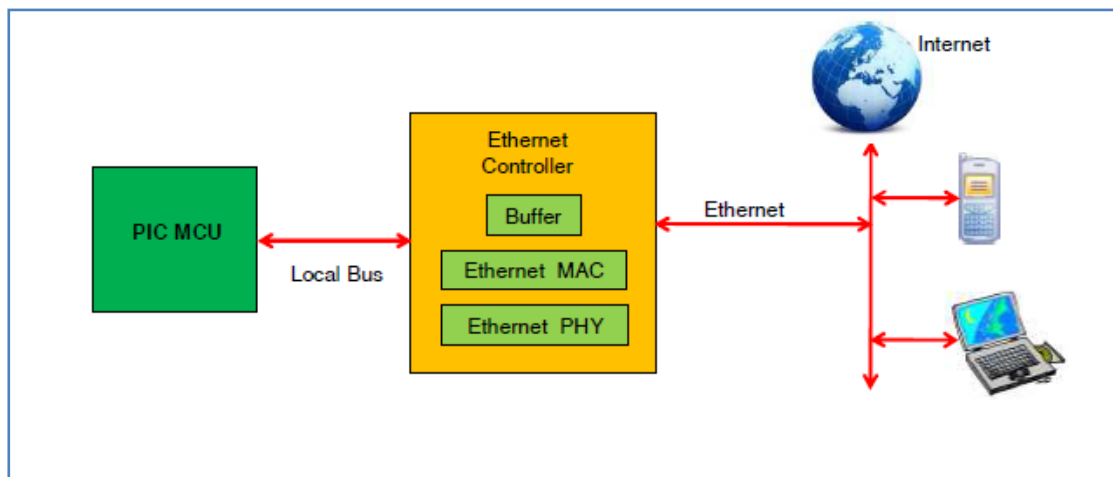


Fig.3.1: Accès sur internet via PIC MCU.

3-Application réseau [7]

Une « application réseau » utilisant l'Ethernet fournit les données à envoyer sur le réseau et utilise les données reçues à partir du réseau. Cette application a souvent une interface utilisateur qui permet aux utilisateurs de demander des données à partir d'un ordinateur connecté sur le réseau, ou à fournir des données à envoyer sur le réseau.

Dans un système embarqué, l'interface utilisateur peut simplement permettre la configuration de base et les fonctions de surveillance, tandis que le système effectue ses communications sans intervention de l'utilisateur.

Les données que l'application envoie et reçoit peuvent être par exemple : un seul octet, une ligne de texte, une demande d'une page Web (le contenu d'une page Web), une image, des données binaires, etc.

Les données envoyées par une application (coté émetteur) suivent un protocole (un ensemble de règles), permettant à l'application (coté récepteur) de comprendre ce qu'il faut faire avec ces données. Ces applications peuvent utiliser un protocole standard tel que le protocole « *hyper text transfer protocol* »(HTTP) pour demander et envoyer des pages Web, le « *File Transfer Protocol* » (FTP) pour transférer des fichiers, le « *Simple Mail Transfer Protocol* » (SMTP) ou le « *Post Office Protocol* » (POP) pour l'envoi de messages.

Dans un système embarqué, l'« application réseau » peut être un module de surveillance qui périodiquement lit et enregistre les lectures de capteurs analogiques ou discrets (tout ou rien). Elle peut être aussi une application de contrôle qui utilise les données des capteurs et les données réseaux pour commander des moteurs, des relais, ou d'autres circuits.

Un système embarqué peut fonctionner comme un serveur Web qui reçoit et répond aux demandes de pages Web. Ceci permet aux utilisateurs de fournir des entrées de commande, ou de configuration, et de visualiser presque en temps réel les données envoyés par le système embarqué.

L'application réseau peut prendre en charge plusieurs processus ou tâches. Par exemple, un système unique peut héberger une page Web et fournir un serveur FTP qui rend les fichiers disponibles pour téléchargement.

4-Protocoles de contrôle de transmissions et de datagramme utilisateur

Un réseau de communication comprend souvent des informations supplémentaires pour transmettre les données à leur destination de façon efficace et sans erreurs. Le protocole « *Transmission Control Protocol* » (TCP) ajoute de

l'information pour la vérification des erreurs, le contrôle de flux, et l'identification des ports des applications réseau. [7]

La vérification des erreurs aide le récepteur à détecter les données reçues qui ne correspondent pas à ce qui a été envoyé. Le contrôle de flux d'information permet à l'expéditeur de déterminer si le récepteur est prêt à recevoir des données. Une valeur numérique identifie l'application au niveau de son port et aide à l'acheminement correct des données. [7] Le protocole TCP effectue toutes ces fonctions.

L'Internet et les réseaux locaux utilisent le protocole TCP. Le système d'exploitation Windows et autres supportent TCP. Les systèmes embarqués dédiés réseau, comprennent généralement des bibliothèques pour le support TCP.

Lors de l'envoi des données en utilisant TCP, la couche application transmet les données à envoyer, les valeurs qui identifient la source de données et la destination à la couche TCP. La couche TCP crée un segment TCP qui se compose d'un en-tête suivi par les données de l'application (figure 3.2). [7]

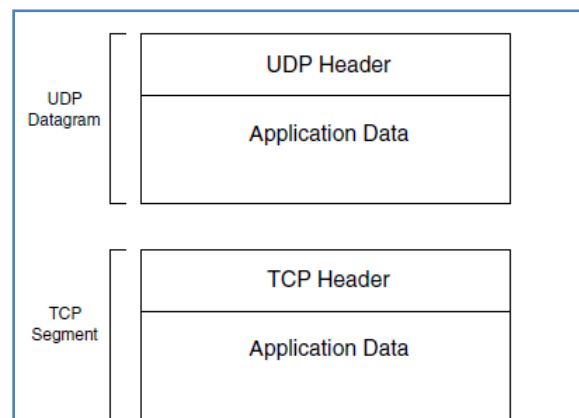


Fig3.2: Couches TCP et UDP.

L'en-tête est une structure définie par des champs contenant les données servant à la vérification des erreurs, le contrôle de flux et le routage du message vers le port correct au point de destination. La couche TCP ne modifie pas le message à envoyer, elle place le message seulement dans la partie de données du segment TCP. Le segment TCP encapsule, ou fournit un conteneur pour les données reçues de la couche d'application. Puis la couche TCP passe le segment vers la couche IP pour la transmission sur le réseau. Dans l'autre sens, la couche TCP reçoit un segment de la couche IP, supprime l'en-tête TCP, et passe le segment vers le port spécifié dans l'en-tête TCP (figure 3.3).[7]

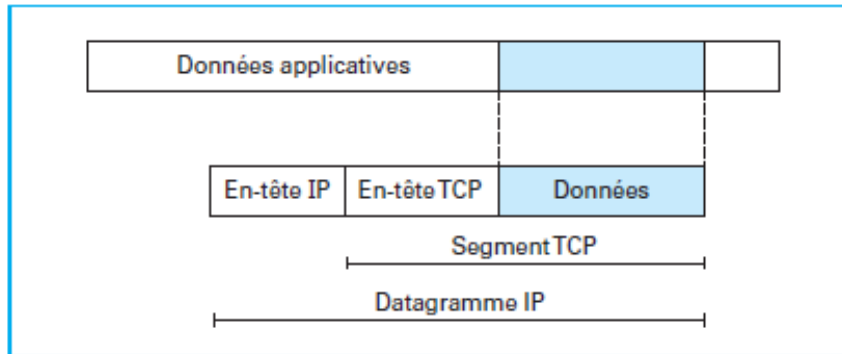


Fig.3.3 : Encapsulation des données avec TCP.

Dans le modèle de référence en couches protocolaires de l'Internet, TCP se situe au niveau de la couche 4 (figure 3.4). Il est généralement mis en œuvre par le système d'exploitation, de sorte que celui-ci offre ces services de transport aux applications à travers une interface de programmation.[8]

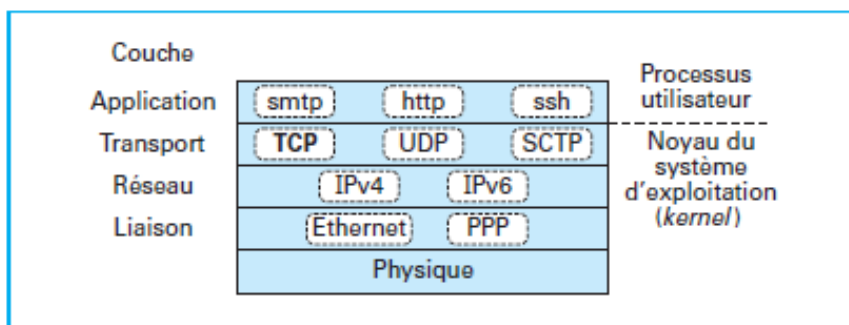


Fig.3.4: Modèle en couches protocolaires de l'Internet.

TCP offre aux applications et aux protocoles qui l'utilisent un canal transparent, sans erreurs et bidirectionnel (en mode « full duplex ») qui transporte une séquence d'octets. C'est un protocole de **point à point** (il n'est pas adapté au « multicast ») et de **bout en bout** : seules les deux extrémités participant à la communication, ou **connexion**, déroulent le protocole (figure 3.5).[8]

Une alternative plus simple à TCP est le protocole de datagramme utilisateur (UDP). Comme pour le segment TCP, un datagramme UDP comporte un en-tête, suivi d'une partie de données qui contient les données d'application. UDP comprend des champs pour spécifier les ports et une option de vérification des erreurs, mais pas de support pour le contrôle de flux. Windows et les systèmes embarqués supportent le protocole UDP.

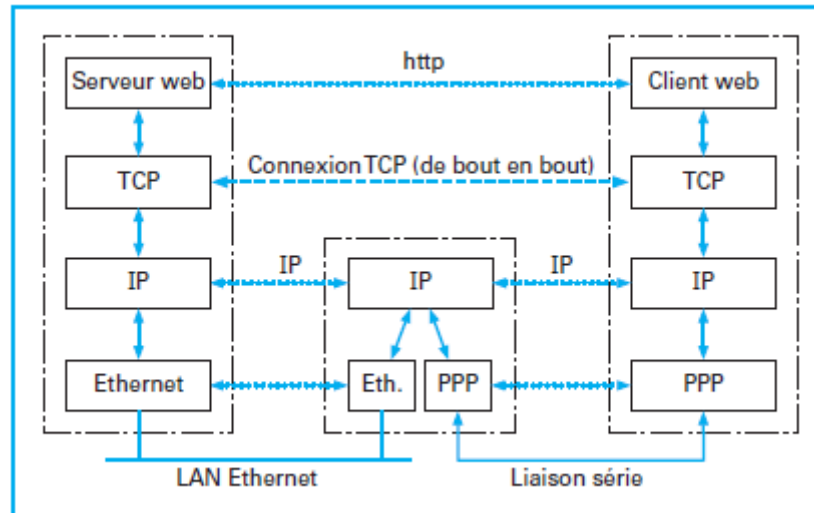


Fig.3.5: Interconnexion des équipements et connexions TCP.

Dans certains réseaux, les communications peuvent ne pas utiliser la couche TCP / UDP entièrement. Par exemple, un réseau local de systèmes embarqués peut ne pas avoir besoin de contrôle de flux ou de contrôle d'erreur supplémentaire au-delà de ce que la trame Ethernet fournit. Dans ces cas, une application peut communiquer directement avec une couche inférieure de la pile de protocoles de réseau, tel que la couche IP. [7]

5-Protocole IP: adresse et routage

La couche du protocole Internet (IP) peut aider à rendre les données à la destination, même si les sources et destinations sont sur différents réseaux locaux. Comme le nom l'indique, le protocole Internet permet aux ordinateurs de communiquer sur l'Internet les uns avec les autres. Il est étroitement lié à TCP et UDP; les réseaux locaux qui utilisent TCP et UDP utilisent également IP.

Le terme TCP/IP fait ainsi référence à des communications qui utilisent les protocoles TCP et IP. Le terme peut également se référer de façon plus générale à la suite de protocoles qui incluent TCP, IP et UDP.

Dans les réseaux Ethernet, une adresse matérielle unique identifie chaque interface sur le réseau. Les adresses IP sont plus souples car elles ne sont pas spécifiques à un type de réseau. Un message qui utilise le protocole IP peut voyager à travers différents types de réseaux, y compris Ethernet, Token Ring et les réseaux sans fil.[7]

En envoyant un message, la couche TCP passe le segment TCP, la source et les adresses de destination à la couche IP. La couche IP encapsule le segment TCP (ou UDP) dans un datagramme IP (figure 3.6).L'en-tête comprend les champs pour la source et la destination de l'adresse IP, la vérification d'erreur de l'en-tête, le routage, et une valeur qui identifie le protocole (TCP ou UDP) utilisé par la partie de données.

A la réception d'un message, la couche IP reçoit un datagramme IP, elle effectue un contrôle d'erreur et utilise la valeur du protocole pour déterminer l'acheminement de la partie de données.

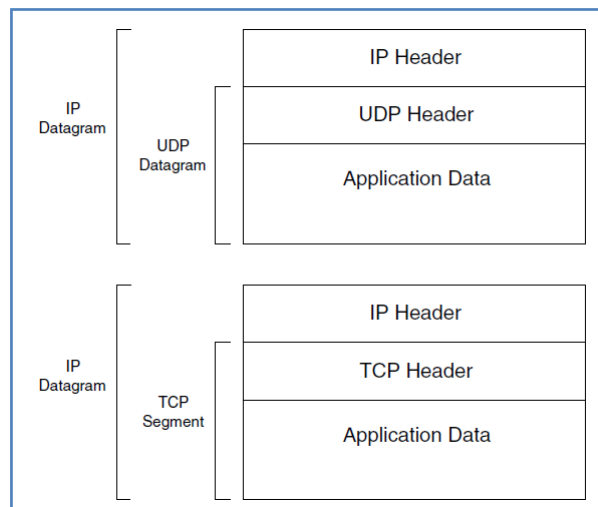


Fig.3.6 : Couche IP avec UDP ou TCP.

Dans l'en-tête IP, l'adresse IP de la source et de la destination identifie les ordinateurs émettant et recevant. Chaque ordinateur sur un réseau qui utilise l'adressage IP doit avoir une adresse unique au sein du réseau, ou des réseaux, que l'ordinateur peut communiquer avec. Les réseaux locaux peuvent utiliser des adresses IP en trois blocs réservés pour les réseaux locaux. Un ordinateur qui communique sur l'Internet doit avoir une adresse différente de l'adresse des autres ordinateurs sur Internet.

La version courante du protocole Internet (IPv4) utilise un nombre de quatre octets, en notation décimale, séparée par un point (e.g. 123.45.67.8). Cette adresse est constituée de trois champs :[9]

1. Une adresse de réseau, qui identifie de manière unique une organisation.
2. Une adresse de sous-réseau, qui identifie un sous-réseau à l'intérieur de cette organisation.
3. Une adresse de système qui identifie un nœud unique sur ce sous-réseau.

La taille de ces champs varie en fonction de la taille de l'organisation, mais ils doivent occuper un total de quatre octets. Le premier nombre détermine la taille de l'adresse de réseau, en utilisant les classes suivantes. [9]

1. Classe A, 0 à 127, un octet
2. Classe B, 128 à 191, deux octets
3. Classe C, 192 à 233, trois octets

La Société pour l'attribution des noms de domaine (voir section 7) et des numéros sur Internet (ICANN : *Internet Corporation for Assigned Names and Numbers*) attribue des blocs d'adresses IP aux fournisseurs de services Internet et d'autres qui peuvent à leur tour affecter une partie de leurs adresses à d'autres utilisateurs.

Trois protocoles sont souvent utilisés avec IP pour l'affectation et l'apprentissage IP:

- Le protocole « *Dynamic Host Configuration Protocol* » (DHCP)
- Le protocole « *Domain name system* » (DNS).
- Le protocole « *Address Resolution Protocol* » (ARP).

Un ordinateur fonctionnant comme un serveur DHCP peut utiliser DHCP pour attribuer des adresses IP aux ordinateurs d'un réseau local.

Un ordinateur qui veut connaître l'adresse IP d'un domaine tel que « google.com » peut utiliser le protocole DNS pour demander des informations à partir d'un ordinateur en fonctionnant en serveur DNS.

Un ordinateur qui veut apprendre l'adresse matérielle Ethernet qui correspond à une adresse IP dans un réseau local peut diffuser une requête ARP pour obtenir cette information.

6-Adresse IP statique et dynamique

Chaque ordinateur qui veut communiquer sur l'Internet doit avoir une adresse IP. L'ordinateur reçoit généralement cette adresse de son fournisseur de service Internet (Internet Service Provider: ISP). L'adresse IP fournie par l'ISP peut être statique ou dynamique. Une adresse IP statique reste la même jusqu'à ce que quelqu'un la change explicitement. Une adresse IP dynamique peut changer à chaque démarrage ou connexion avec le réseau Internet. [7]

Un système embarqué peut stocker une adresse IP statique dans la mémoire. Le système peut aussi recevoir une adresse IP statique ou dynamique d'un serveur DHCP au démarrage ou connexion avec le réseau [7]. Dans notre module c'est la première méthode qui a été utilisé comme nous allons le présenter dans la suite du chapitre.

Ci-dessous une liste des principaux ISP au Liban:

- OGERO
- Cyberia
- Wise
- Sodetel
- IDM
- Terranet
- Lynx

7-Nom de domaine

Un nom de domaine est constitué d'un nom qui est unique dans son domaine racine, suivi d'un point et le nom du domaine racine. En voici quelques exemples : myzergroup.com, Isae.edu.lb.

Les domaines racines les plus utilisés : «. com », «. edu »,«. gov »,«. net »et«. org ». Dans les dernières années, d'autres ont été ajoutées.

Un nom de domaine peut également contenir le code du pays comme domaine de premier niveau à droite du domaine racine. Par exemple: isae.edu.**lb**

Un ou plusieurs noms à gauche du domaine principal peuvent identifier un sous-ensemble. Par exemple: **si**.isae.edu.lb. L'ordre des noms des sous-ensembles indique leur hiérarchie. Dans l'exemple ci-dessus, « si » est un sous-ensemble de isae.edu.lb.

Les lettres www précédant généralement un nom de domaine précisent que la demande devrait être routée vers le serveur Web du domaine: www.isae.edu.lb.

De nombreux domaines sont configurés de telle sorte que les lettres www sont facultatives. À la réception d'une requête HTTP ne comportant pas« www », le logiciel du domaine transmet la requête au serveur Web par défaut [7].

Lors d'une requête http (e.g. navigateur), le client DNS parle à l'un des serveurs DNS déjà configuré (par DHCP ou configuration statique). Lorsque le serveur DNS renvoie une adresse IP, le navigateur affichera par exemple [9]:

Contacting: myzergroup.com

Suivie par:

Web site found. Waiting for reply...

Le contenu de la page Web est transféré au navigateur. La vitesse de la réponse peut parfois rendre difficile à suivre cette séquence de messages. Ceci montre la puissance du système DNS de l'Internet mondial.

En résumé, les étapes pour trouver un site Web en utilisant le service DNS sont les suivantes:

1. Le résolveur DNS local envoie une requête DNS pour un nom de domaine (e.g. myzergroup.com).
2. Le serveur DNS renvoie l'adresse IP correspondante.
3. La requête HTTP est envoyée par le navigateur avec l'IP reçue.
4. La première réponse vient du site Web (site Web trouvé). [9]

8-Nom de domaine dynamique

Le service de nom de domaine dynamique (Dynamic DNS : DDNS) fournit une méthode pour mettre à jour une adresse IP dynamique. Il permet ainsi aux utilisateurs d'un nom de domaine fixe d'accéder à un réseau et à ses ressources en les attribuant dynamiquement des adresses IP.

Chaque périphérique sur une configuration DDNS dispose d'une adresse de protocole internet qui peut changer aléatoirement après chaque connexion. Il reste actif sur la même connexion Internet au moyen d'un système de noms de domaine dynamique.

L'inverse se produit avec une configuration de protocole Internet statique, où un utilisateur fournisseur de services Internet attribue une adresse IP unique pour chaque périphérique sur un réseau à partir d'un ensemble d'adresses IP non utilisées lors de la connexion.

Dans notre projet c'est la méthode DDNS qui a été utilisée parce qu'elle est beaucoup moins chère que l'utilisation d'une adresse IP statique. Une simple comparaison des tarifs des deux méthodes justifie ce choix :

- Prix d'une adresse IP statique proposé par la société Cyberia : 2100 \$ chaque année. (voir annexe II).
- Prix d'adresse dynamique (avec 30 noms de domaine) proposé par DynDNS.com : 15 \$ chaque année. (voir annexe II).

8.1-Configuration du DDNS

Cette section détaille les étapes effectuée pour créer un compte sur le site « DynDNS.com » afin d'obtenir un nom de domaine dynamique ainsi que la configuration nécessaire du modem DSL pour permettre l'accès à notre serveur embarqué selon le service DDNS.

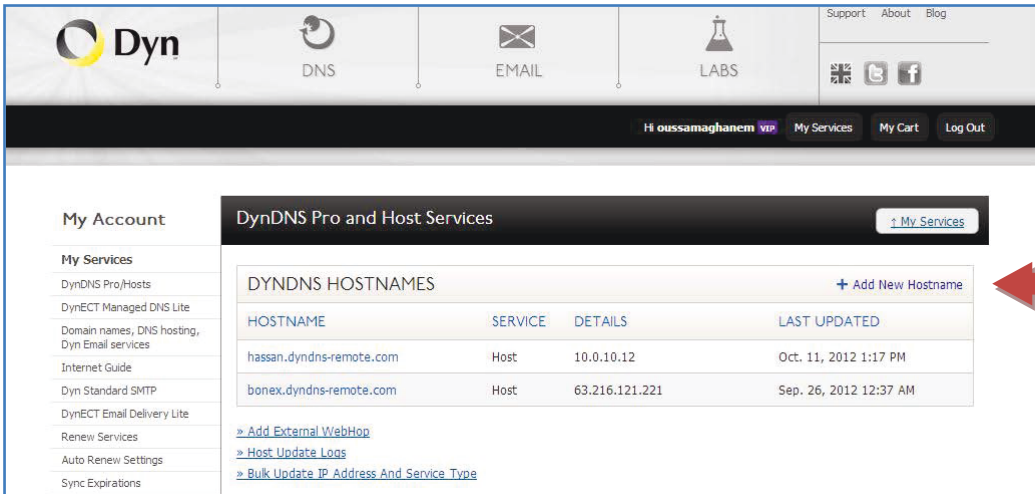
Notons cependant que l'ISP doit permettre le service DDNS pour pouvoir accéder à notre serveur web embarqué (Ce n'était pas par exemple le cas de l'ISP Cyberia).

Etape 1:

Nous avons en premier lieu créé un compte sur le site « DynDNS.com ». Puis nous nous sommes connectés à travers ce compte.

Etape 2:

Nous avons créé un nouvel « nom de hôte » (*Host name*) en cliquant sur « Add New Hostname » comme la présente la figure 3.7.



The screenshot shows the DynDNS.com user interface. The user is logged in as 'oussamaghanem'. The main content area is titled 'DynDNS Pro and Host Services'. Underneath, there is a section for 'DYNDNS HOSTNAMES' with a '+ Add New Hostname' button. A table lists existing hostnames:

HOSTNAME	SERVICE	DETAILS	LAST UPDATED
hassan.dyndns-remote.com	Host	10.0.10.12	Oct. 11, 2012 1:17 PM
bonex.dyndns-remote.com	Host	63.216.121.221	Sep. 26, 2012 12:37 AM

Below the table, there are links for 'Add External WebHop', 'Host Update Logs', and 'Bulk Update IP Address And Service Type'. A red arrow points to the '+ Add New Hostname' button.

Fig.3.7: Création d'un nouvel hôte sur le site DynDNS.com.

Nous avons utilisé « zara » comme nom de l'hôte et « dyndns-remote.com » comme nom de domaine. Zara (Le Mall – Sin EL Fil) étant le nom du magasin dans lequel nous avons installé notre module. Le variable TTL représente le temps de rafraichissement de l'IP (figure3.8)

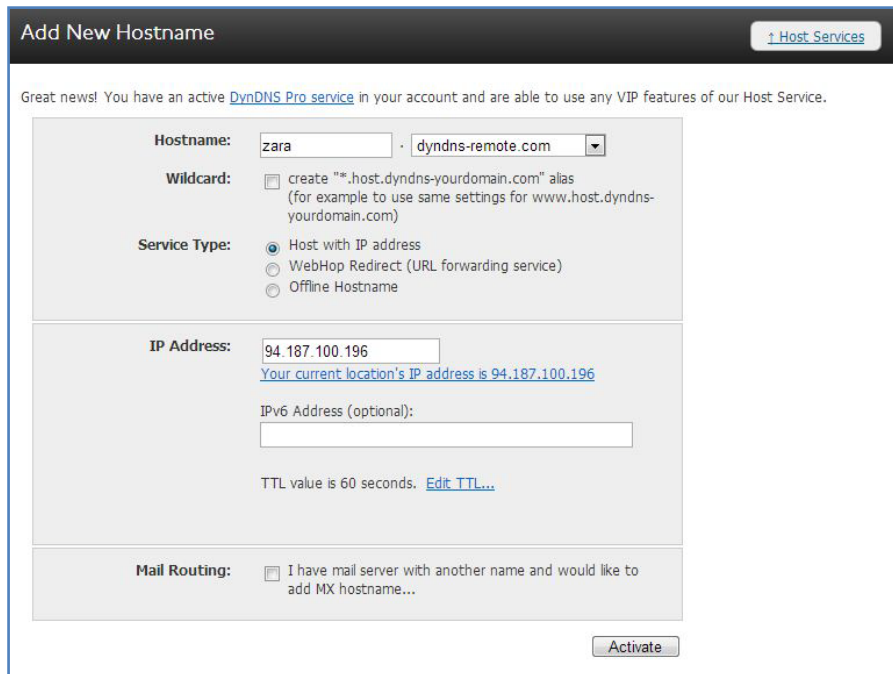


Fig.3.8: Création de l'hôte « zara ».

Maintenant un nouvel hôte s'ajoute aux précédents. Notre module sera accédé à l'adresse.

"zara.dyddns-remote.com",

Etape 3:

Cette étape représente la configuration du routeur DSL chez notre client « Zara ». Nous nous sommes connecté à l'adresse IP du router (*Gateway IP adress*). Nous avons ensuite accédé à la boîte à outils « toolbox » et nous avons cliqué sur « Dynamic DNS » (figure 3.9). Nous avons besoin de configurer ici le nom de l'hôte créée ci-dessus.

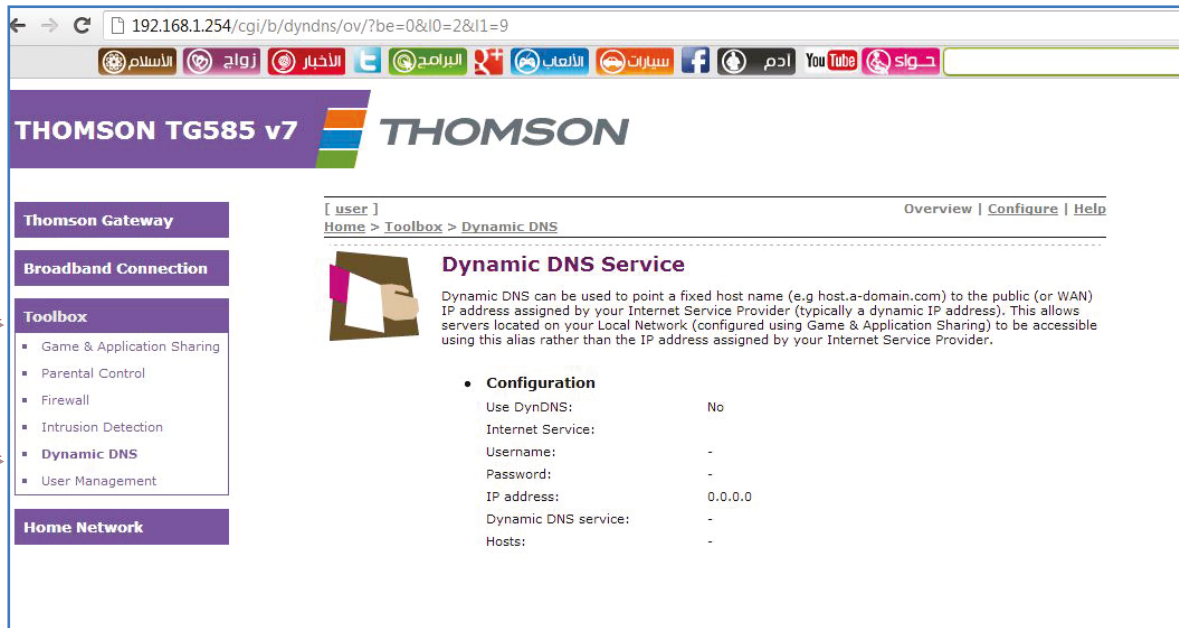


Fig.3.9: Configuration du DDNS sur le routeur.

Etape 4:

La fenetre de configuration du « *Dynamic DNS* » apparait (figure 3.10), les informations concernant le nom et le mot de passe doivent être remplis. Dans la case de service nous choisissons le « *dyndns* », ce qui correspond à « *dyndns.com* ».

Dans la case « *HOST* » nous fixons le non de l'hôte déjà créé :

"zara.dyndns-remote.com"

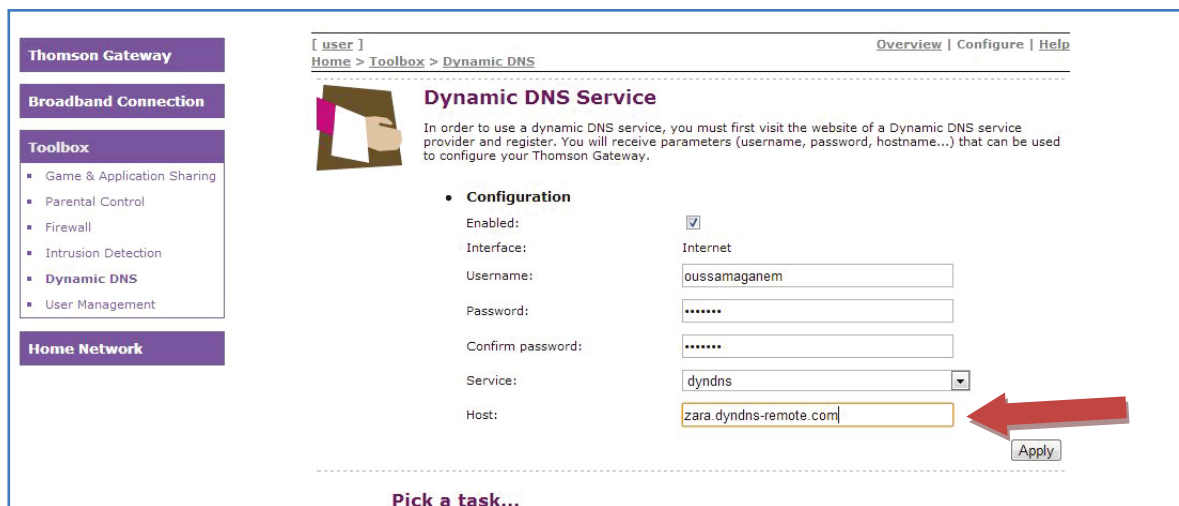


Fig.3.10: Configuration de l'hôte sur le DDNS du routeur.

Nous cliquons sur le bouton « *Apply* » pour valider.

Le résultat de la configuration du «*Dynamic DNS* » apparaît avec la nouvel hôte et un message «*update in progress* » (figure 3.11).

THOMSON TG585 v7 **THOMSON**

[user] Overview | Configure | Help
Home > Toolbox > Dynamic DNS

Dynamic DNS Service

Dynamic DNS can be used to point a fixed host name (e.g host.a-domain.com) to the public (or WAN) IP address assigned by your Internet Service Provider (typically a dynamic IP address). This allows servers located on your Local Network (configured using Game & Application Sharing) to be accessible using this alias rather than the IP address assigned by your Internet Service Provider.

Configuration

Use DynDNS:	Yes
Internet Service:	Internet
Username:	oussamaganem
Password:	*****
IP address:	94.187.100.196
Dynamic DNS service:	dyndns
Hostname:	zara.dyndns-remote.com (Update in progress)

Fig.3.11:Résultat de la configuration du DDNS du routeur.

Etape 5:

Dans cette étape nous allons créer la nouvelle application qui va utiliser l'adresse du nouvel hôte. Dans la boîte à outils nous choisissons «*Game & Application sharing* » pour créer une nouvelle application. Nous lui assignons un nom «*SystèmeEmbarque*» (figure 3.12).

THOMSON TG585 v7 **THOMSON**

[user] Help
Home > Toolbox > Game & Application Sharing > New Game or Application

New Game or Application

Enter the name of the new game or application.

Name:

Select how you want to define the new game or application.

Clone Existing Game or Application
 ABC (Another Bittorent Client)
 Manual Entry of Port Maps

Next Cancel

Fig.3.12: Nouvelle application.

Puis nous cliquons sur «*Manual Entry of Port Maps* » pour choisir le port sur lequel les données vont être transmises. Nous avons choisi pour notre application le port 8899 (figure 3.13).

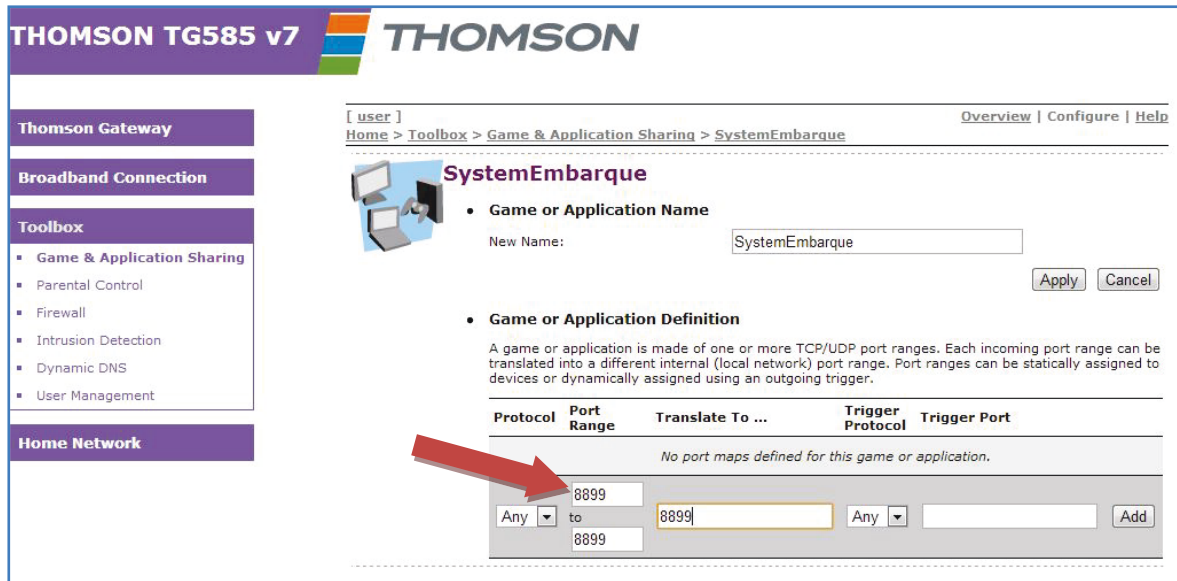


Fig.3.13: Choix du port de l'application créé.

Etape 6:

Dans cette étape nous allons assigner l'application créée à l'adresse IP de notre module (voir chapitre 5 concernant l'IP du serveur web embarqué). Dans la boîte à outils, sur le lien « Game & Application sharing », nous fixons le nom de l'application déjà créée "SystemEmbarque" et l'adresse IP physique fixe attribuée à notre serveur embarquée lors de la programmation (en MikroC). Nous cliquons sur le bouton « Add » pour valider (figure 3.14).

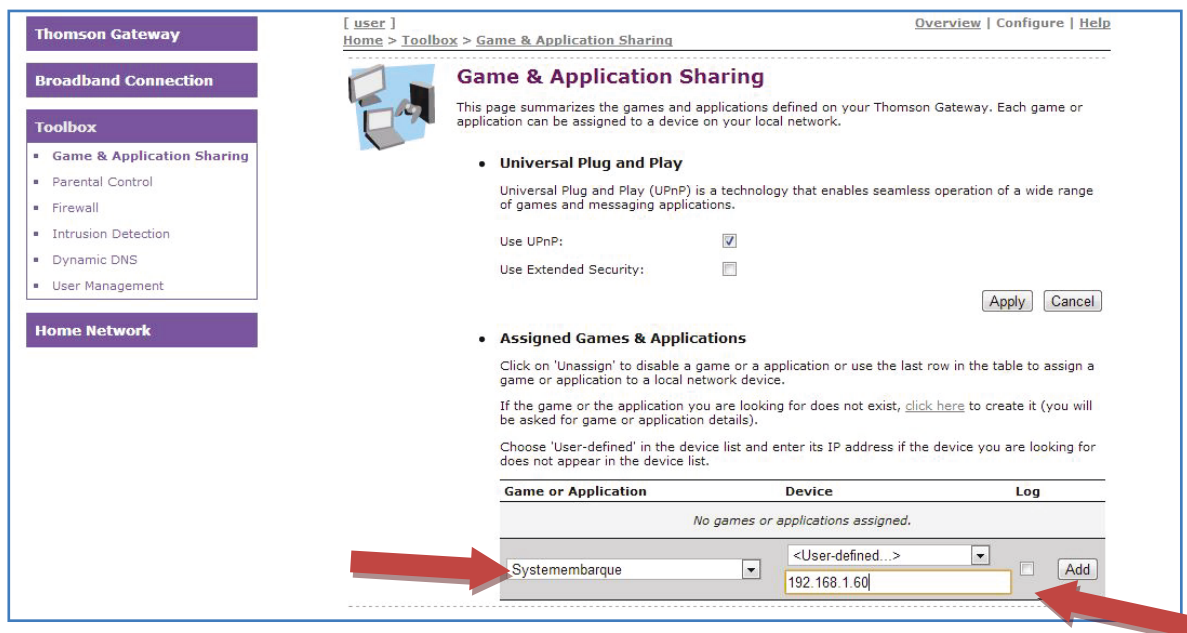


Fig.3.14: Union de l'adresse IP avec l'application créée.

La boîte de l'application apparaît avec le nom de l'application et l'adresse IP assignée (figure 3.15).

The screenshot shows the Thomson Gateway web interface. The left sidebar contains navigation menus for 'Thomson Gateway', 'Broadband Connection', 'Toolbox', and 'Home Network'. The 'Toolbox' menu is expanded, showing options like 'Game & Application Sharing', 'Parental Control', 'Firewall', 'Intrusion Detection', 'Dynamic DNS', and 'User Management'. The main content area is titled 'Game & Application Sharing' and includes a breadcrumb trail: 'Home > Toolbox > Game & Application Sharing'. Below the title, there is a description of the service and two main sections: 'Universal Plug and Play' and 'Assigned Games & Applications'. The 'Assigned Games & Applications' section contains a table with the following data:

Game or Application	Device	On	Edit	Unassign
Systemembarque	192.168.1.60	<input checked="" type="checkbox"/>	Edit	Unassign
ABC (Another Bittorrent Client)	A-10-PC	<input type="checkbox"/>	Add	

Fig.3.15: L'application avec l'adresse IP de la carte utilisée.

Après cette étape la configuration est complète. Ceci permettra à notre serveur web embarqué de se connecter au réseau en utilisant le service DDNS.

9-Conclusion

Dans ce chapitre, nous avons décrit les principaux protocoles utilisés pour la communication sur internet. Nous avons présenté la configuration nécessaire pour permettre l'accès à notre serveur embarqué selon le service DDNS. Cette méthode a été utilisée afin de réduire les coûts et obtenir un contrôle à distance, à travers le web, à un prix très abordable.

Dans le chapitre suivant nous allons parler de l'« interface web » qui a été réalisée. Nous allons présenter les différents langages de programmation utilisés pour obtenir cette interface et détailler les différentes formes (pages) réalisées.

Chapitre 4 : Réalisation de l'Interfaçage web

Dans nos jours, beaucoup de systèmes embarqués s'intègrent des serveurs Web, pour un affichage direct à partir des navigateurs réseaux ou pour la communication avec d'autres serveurs. Les pages hébergées par les serveurs (y compris ceux des systèmes embarqués) sont susceptibles d'afficher le contenu dynamique qui peut changer chaque fois que la page est demandée. Plusieurs exemples existent sur le contenu dynamique: lecture de capteurs discrets ou analogiques, affichage de la date et l'heure, etc... Certaines pages permettent également aux utilisateurs de fournir des données ou commandes au serveur, qui les traite et retourne un résultat sur la page Web ou effectue une commande appropriée à travers le système embarqué dans lequel il est intégré.

Il ya plusieurs façons d'ajouter un contenu dynamique aux pages Web. Par exemple, les applications programmées en C peuvent utiliser le « *Common Gateway Interface* » (CGI), Les applications programmées en Java utilisent des « servlets » java.

Dans ce chapitre nous allons présenter les principaux langages d'interfaçages Web utilisés dans notre projet. Nous allons ensuite décrire l'interfaçage implémenté sur le serveur. Nous présenterons les différentes pages, leurs fonctionnements et leurs liens avec la base de données utilisée. Le serveur du système embarqué sera présenté au chapitre 5.

1-Langages de programmation utilisés

Le « World Wide Web »normalement en abrégé "le web" ou parfois "le net" est une large collection de documents électroniques composés chacun d'un ensemble de pages liées écrites en HTML. Les documents sont stockés dans des fichiers sur des milliers d'ordinateurs distribués sur le réseau de l'Internet mondial.

Les concepts sous-jacents du Web ont été conçus en 1989 par Tim Berners-Lee qui travaillait aux laboratoires du Conseil européen pour la recherche nucléaire, CERN. Un accord a été signé en 1994 entre le CERN et l'institut de technologie du Massachusetts (MIT) pour mettre en place un consortium dont le but était de développer davantage le web et standardiser les protocoles qui lui sont associés. Le centre américain pour les applications des super-ordinateurs (NCSA) a également

apporté une contribution majeure à l'usage des navigateurs Web basés sur une interface graphique. Depuis ce temps de nombreux développements ont eu lieu. Le Web est maintenant la principale source de données transférées via Internet.[10]

1.1-URL et HTTP

Un localisateur uniforme de ressource (*Uniform Resource Locator (URL)*) comprend le nom du fichier et l'emplacement du serveur sur Internet sur lequel le fichier est stocké. Le protocole HTTP est utilisé par le programme navigateur (coté client) pour communiquer avec un programme coté serveur.

Ainsi URL et HTTP définissent la façon dont le contenu de chaque page Web est affiché sur l'écran PC de l'utilisateur, et la mise en place des hyperliens avec d'autres pages.

1.2-HTML

Le « *Hyper Text Markup Language* »(HTML), est le langage standard utilisé pour écrire les pages web. HTML est un langage de balisage qui est utilisé pour décrire la méthode d'affichage d'un document ou d'une page sur l'écran de l'ordinateur à l'aide d'un navigateur. [10]

Les commandes de balisage ou de format sont connues en tant que directives pour HTML et la majorité sont exprimées à l'aide d'une paire de clés (ou *tags*<>). Différentes types de directive précisent comment le texte doit être présentée sur l'écran. Il y a des directives qui permettent de spécifier une image ou un segment audio ou vidéo dans une page et comment celles-ci doivent être affichées, etc... [10]

Le tableau 4.1 représente les principales balises utilisées dans une page HTML. Il convient cependant de souligner, que le HTML est continuellement en cours de révision. [10]

1.3-Applet Java et Java Script

Un inconvénient d'utiliser uniquement du HTML pour écrire des pages web, c'est qu'il est relativement difficile d'intégrer de nouvelles fonctionnalités dans ces pages (par exemple un nouveau décodeur audio), puisque cela nécessiterait des modifications du code du navigateur. [10]

Pour surmonter cette contrainte, il est possible de mettre en œuvre dans une page des

Tab IV.I: Sélection des directives ou balises du format HTML.

Ouverture de la balise	Fermeture de la balise	Utilisation
<HTML>	</HTML>	Spécifie le début et la fin de du document ou de la page.
<HEAD>	</HEAD>	Spécifie le début et la fin de l'en-tête.
<TITLE>	</TITLE>	Spécifie le titre de la page. Il n'est pas affiché comme faisant partie de la page
<BODY>	</BODY>	Spécifie le début et la fin du contenu de la page affichée.
<I>	</I>	Affiche le texte correspondant en gras et en Italique.
		Spécifie le début et la fin d'une liste non ordonnée
		Spécifie le début et la fin d'une liste ordonnée
		Spécifie le début d'un élément énuméré
 		Spécifie le début d'une nouvelle ligne
<P>		Spécifie le début d'un nouveau paragraphe
		Spécifie un lien vers une autre page
		Spécifie un lien vers un autre point dans la même page
<!-->	-->	Début et de fin des commentaires

parties de code autonomes (sous-programmes), connu par « applets », et qui sont indépendants de l'interpréteur HTML. Ensuite, de la même manière qu'une image est accédée et affichée lorsque son étiquette ou balise est interprétée (par l'interpréteur

HTML), un applet est identifiée par une étiquette et lorsque cela est interprété, le code de l'applet est accédée et exécuté. [10]

L'avantage d'utiliser des applets, c'est que puisque chaque applet est un programme autonome, alors en mettant les parties d'une page qui contient un code susceptible de changer sous forme d'applets, tous les changements qui peuvent se produire sont incorporés dans le serveur et non pas dans le navigateur.

Un exemple d'un langage de programmation qui est utilisé pour produire des applets qui sont téléchargés à partir d'un serveur est "**Java**".

Il est également possible d'inclure du code Java dans une page HTML directement. Le langage utilisé est appelée "**JavaScript**". Ce dernier a été présenté comme le parfait complément du langage de programmation Java, puisque les deux langages peuvent communiquer ensembles à l'intérieur d'une fenêtre de navigateur.[11]

Aujourd'hui, JavaScript est omniprésent dans l'Internet. Il est estimé aujourd'hui que plus que 25 pourcent des pages Web sur l'Internet contiennent du code JavaScript. Certains d'entre elles utilisent ce code pour réaliser certains effets à l'aide de scripts très simples, telles que le contrôle de la souris et la fixation de la barre d'état du navigateur. D'autres contiennent des menus dynamiques complexes et des formes automatiques. Aucun autre langage de script ne se rapproche à la domination du JavaScript dans l'environnement du navigateur Web.

Le succès de JavaScript vient de sa facilité d'utilisation. Par exemple, une simple déclaration de JavaScript peut créer un effet intéressant dans le navigateur, et un petit nombre de déclarations est tout ce qui est nécessaire pour un environnement totalement interactif du navigateur.[11]

Les utilisateurs n'ont plus besoin d'outils spéciaux pour créer ces effets. Un simple éditeur de texte (*e.g.* Notepad), est tout ce qui est nécessaire. Si les utilisateurs peuvent créer des pages web en HTML, donc ils possèdent tous les outils dont ils auront besoin pour commencer.

Avec la sortie de la version 2.0, JavaScript est devenue utilisable à côté d'autres langages de programmation tels que C++ et Visual Basic. Microsoft a inclus une version de JavaScript 2.0, appelée JScript.NET. Ce dernier peut accéder à presque toutes les classes .NET et peut être utilisé pour créer des formulaires d'applications Windows.[11]

1.4-PHP [12]

Côté serveur, des langages script tels que le PHP sont utilisés. La simplicité de PHP à intégrer des liens au programme de base de données MySQL et à créer rapidement des éléments dynamiques sur les sites Web, a contribué largement à augmenter le nombre de ses utilisateurs.

Avec PHP, c'est une simple question d'intégrer une activité dynamique dans les pages Web. Du point de vue développeur, tout ce que vous avez à faire est d'écrire un code tel que le suivant:

```
<?php  
echo ". Bonjour tout le monde.";?>
```

La balise de début, `<? Php`, indique au serveur Web qu'il faut interpréter en PHP tous le code qui suit jusqu'à la balise `?>`. En dehors, tout est envoyé au client en format HTML directement.

Avec PHP, les développeurs Web ont un langage assez rapide et qui s'intègre de façon transparente avec le code HTML. L'utilisation de PHP, permet un contrôle illimité sur le serveur Web, e.g. le traitement d'une carte de crédit, l'ajout des détails d'un utilisateur à la base de données, ou la recherche des informations à partir d'un site Web. Tout peut se faire à partir du même fichier PHP dans lequel le code HTML réside lui-même.

1.5-MySQL [12]

MySQL est un système de base de données rapide et puissant, pourtant facile à utiliser. L'association de PHP avec MySQL permet de stocker et de récupérer les données. Ainsi les utilisateurs possèdent les éléments fondamentaux nécessaires pour le développement des sites Web.

Dans les premiers jours du Web, de nombreux sites utilisaient des fichiers texte pour stocker les données telles que les noms des utilisateurs et leurs mots de passe. Mais cette approche pouvait causer des problèmes si le fichier n'était pas correctement verrouillé contre la corruption de plusieurs accès simultanés. En outre, la taille du fichier peut facilement de telle sorte qu'il ne devienne difficile à gérer, sans parler de

la difficulté d'essayer de fusionner des fichiers et d'effectuer des recherches complexes de n'importe quel type...

C'est là que les bases de données relationnelles avec des requêtes bien structurées sont devenues essentielles. MySQL, étant libre d'utiliser, est ainsi installé sur un grand nombre de serveurs Web sur Internet. Il s'agit d'une gestion de base de données robuste et exceptionnellement rapide.

Dans MySQL les utilisateurs peuvent avoir une ou plus de tables qui contiennent des données. Par exemple, supposons qu'ils travaillent sur une table appelée utilisateurs, possédant des colonnes pour le nom, le prénom et l'e-mail, et ils souhaitent maintenant ajouter un autre utilisateur. Une simple commande peut être utilisée pour faire cela:

```
INSERT INTO VALUES utilisateurs ('Smith', 'Jean', 'jsmith@mysite.com');
```

Il est tout aussi facile de rechercher des données. Supposons que les utilisateurs possèdent une adresse e-mail pour un utilisateur et qu'ils ont besoin de chercher son nom. Pour ce faire, ils peuvent utiliser une requête MySQL tel que:

```
SELECT nom, prénom FROM utilisateurs WHERE email = 'jsmith@mysite.com';
```

MySQL retournera ensuite Smith, John et d'autres paires de nom/prénom qui peuvent être associés avec cette adresse e-mail dans la base de données.

Nous pouvons aussi réaliser avec MySQL des commandes plus complexes que « INSERT » et « SELECT ». Par exemple, nous pouvons joindre plusieurs tables en fonction de différents critères, demander les résultats dans une variété de différents ordres, faire des recherches partielles, et beaucoup plus.

L'utilisation de PHP, permet tous ces appels directement à MySQL. Cela signifie que les utilisateurs peuvent enregistrer les résultats dans des tableaux pour le traitement, effectuer des recherches multiples, chacune dépendante des résultats retournées par des recherches précédentes, etc...

2-Interface web réalisée

Dans cette partie nous présenterons une description détaillée du fonctionnement de l'application Web utilisée dans notre projet.

La figure 4.1 représente l'organigramme du programme d'interfaçage. Cet organigramme représente les différents pages présentes dans notre interface web et le contenu de chacune d'elle. Il présente aussi les liens entre certain de ces pages et la base de données. Cet organigramme fonctionne d'une façon sécurisé et donne des accès privilégiés aux administrateurs de la compagnie.

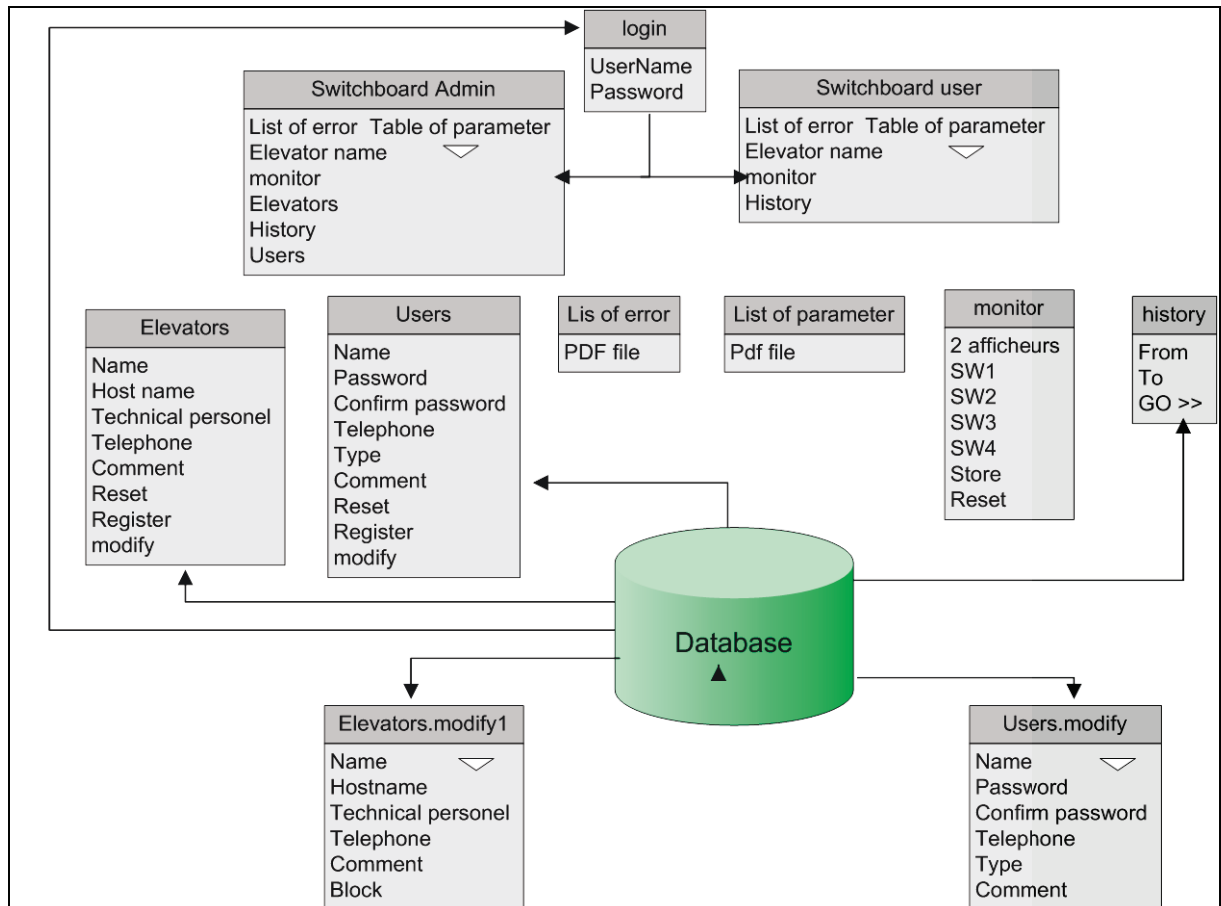


Fig.4.1 : Organigramme du programme d'interfaçage.

Une description détaillée de ces pages (formes) est présentée dans la partie 2.2.

2.1-Base de données

Le tableau IV.II représente les principaux tableaux utilisés dans la base de données et les principales informations stockées dans ces tableaux.

Cette base de données est formée de 5 tables. Premièrement nous avons la table d'ascenseur dans laquelle nous sauvegardons le nom de chaque ascenseur, le personnel (du client) à contacter, et son numéro de téléphone. Dans cette table nous avons associé un numéro unique pour chaque nouvel ascenseur afin qu'il soit appelé

par ce numéro à l'intérieur de la base de données. Ce numéro s'auto incrémente d'une façon croissante à chaque association d'un nouvel ascenseur.

Cette table est reliée à la table de l'historique des erreurs. Dans cette dernière nous saisissons le temps et la date d'erreur correspondant à celles du serveur hôte (notre

Tab.IV.II : Liste des tables utilisées dans la base donnée.

#	Nom de la table	Stockage des informations reliées à
1	elevator	Ascenseur
2	historyErr	Historique des erreurs
3	errorlist	Liste des erreurs
4	members	Utilisateurs
5	loginAttempts	Contrôle de l'accès des utilisateurs

hébergeur web étant le « a2ahost.com »). Dans cette table aussi nous avons associé (comme dans la première table) un numéro unique et qui s'auto incrémente d'une façon croissante à chaque insertion d'erreur.

La troisième table enregistre l'erreur et sa description. À chaque défaut notre programme ajoute l'erreur à la base de données d'une façon descendante. Les descriptions des erreurs sont présentées dans un fichier au format Excel ce qui permet d'effectuer une comparaison afin de présenter la signification de l'erreur lors d'une requête. Encore ici, nous avons associé dans cette table un numéro unique.

La table des utilisateurs, "members", représente les utilisateurs, leurs mots de passe et leurs téléphones. Dans ce tableau nous avons encore associé un numéro par défaut à chaque utilisateur.

La dernière table « loginAttempts » représente le nombre d'accès de chaque utilisateur. Elle est reliée à la table précédente où chaque utilisateur doit écrire son nom et son mot de passe d'une façon identique à celle présenté dans la table. En cas d'erreurs, il ne peut pas accéder la page web demandée. Cette dernière table est utilisée pour

compter le numéro d'essais d'accès de chaque utilisateur afin d'empêcher une connexion après 5 essais erronés consécutifs.

La figure 4.2 représente les différentes relations entre les tables de la base de données.

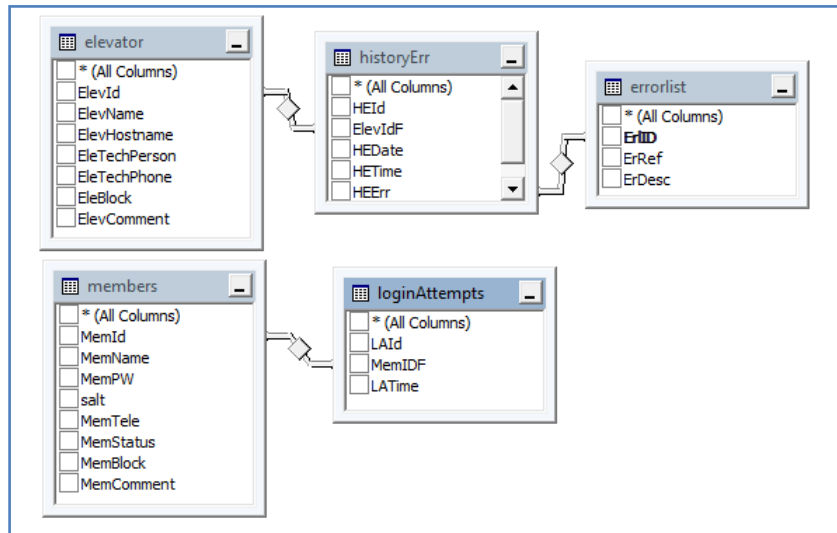


Fig.4.2: Relations entre les différentes tables de la base de données.

2.2-Formes réalisées

Dans cette section nous présenterons les différentes formes de notre interface web, ces formes utilisent chacune un code « PHP » (voir tableau IV.III) pour l'association de chaque forme au code PHP correspondant.

Tab.IV.III: Liste des codes PHP utilisés.

Formes	Non du code	Description
A	login.php	Page de connexion
B	switchboard.php	Page d'accueil et le conseil interrupteur
C	Monitor.php	Surveillance d'un ascenseur spécifique.
D	newelevator.php	Ajout et modification des informations des ascenseurs.
E	elevhistory.php	Aperçu l'erreur d'un ascenseur spécifique et sa date de défaut.
F	newuser.php	Ajout et modification des informations des utilisateurs.

L'accès à ces formes se fait à travers le site web de la compagnie "myzergroup", « www.myzergroup.com », en cliquant sur le bouton "Elevatortcontrole" de la page principale

2.2.1-Forme A: Login (figure 4.3)



Fig.4.3 : Interface de connexion d'utilisateur.

Cette page contient le formulaire d'accès. Pour se connecter à l'application Web, nous devons faire entrer le nom d'utilisateur et le mot de passe dans les champs correspondants, puis appuyer sur le bouton «login».

Si les informations remplies correspondent au nom d'utilisateur et à son mot de passe stocké dans la base de données, la page « switchboard » sera accédée. Sinon nous obtenons à l'écran le message "ErrorLogging In!".

Si uniquement le mot de passe est incorrect, le système va enregistrer le nom d'utilisateur avec la date et l'heure dans la table "loginAttempts".

Dans notre script PHP, si un utilisateur effectue cinq essais de connexion erronés et consécutifs, son compte sera verrouillé et il ne sera capable de se connecter de nouveau qu'après une période de 2 heures. En plus aucun lien vers n'importe quelle forme n'est possible qu'à travers la page du "Login". Ceci contribue à la sécurisation de notre application web.

2.2.2-Forme B: Switch Board (figure 4.4)

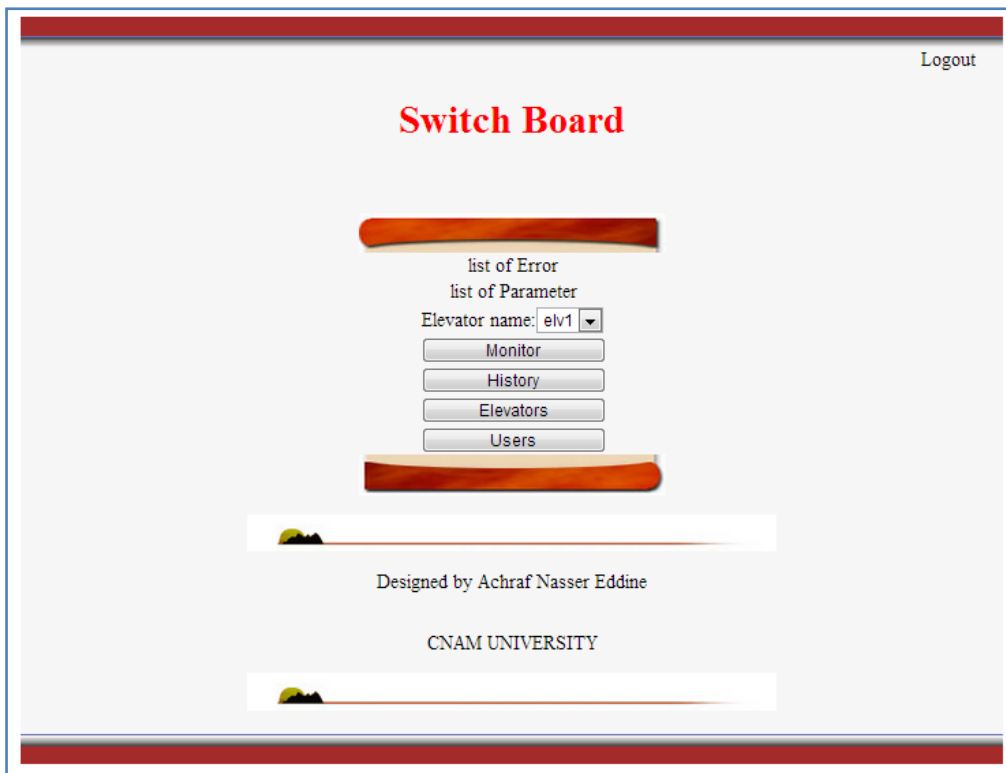


Fig.4.4 : Interface de gestion des ascenseurs.

Cette forme représente l'interface principale de notre application. Elle contient tous les boutons nécessaires pour accéder aux différents ascenseurs.

Ci-dessous une liste détaillée des formes ouvertes lors de l'appui sur les différents liens et boutons de cette interface:

- "List of error" : Ce lien ouvre un fichier pdf qui contient une liste des numéros d'erreurs avec leur description.
- "Table of parameter" : Ce lien ouvre un fichier pdf qui contient un tableau contenant une description des paramètres et les numéros qui lui sont associés.
- "Elevatormame" : Cette liste déroulante contient tous les ascenseurs enregistrés dans la base de données. Elle nous permet de choisir le nom de l'ascenseur auquel nous voulons accéder.
- "monitor" : Ce bouton permet d'entrer dans la page de contrôle et de surveillance à distance de l'ascenseur choisi dans « Elevatormame » (Forme C).

- "Elevators": Ce bouton permet de modifier la liste des ascenseurs disponibles : ajout ou blocage. (forme D).
- "History" : Ce bouton ouvre une liste contenant l'historique des erreurs correspondantes à l'ascenseur choisi dans « Elevatorname » (forme E)
- "Users" : Ce bouton permet de modifier de modifier la liste des utilisateurs disponibles : ajout ou blocage. (forme F).
- "logout" : Ce bouton sert à quitter cette page et revenir à la page initiale du site « www.myzergroup.com »

2.2.3-Forme C : monitor (figure 4.5)



Fig.4.5 : Interface de contrôle et de surveillance de l'ascenseur.

Cette forme représente l'interface de contrôle et de surveillance de l'ascenseur. Elle est implémentée sur le serveur embarqué. Avec le système embarqué installé dans le local de l'ascenseur, cet interface remplace la carte BG-17 (Carte locale de commande et diagnostique de l'ascenseur fournie par le constructeur Autinor tout en fournissant un accès à distance. Elle contient tous les boutons qui nous permettent de contrôler la carte mère de l'ascenseur. Elle contient aussi deux digits qui affichent le contenu des 2 afficheurs 7 segments présents dans le système embarqué avec un fonctionnement quasi temps réel. (Voir chapitre suivant pour plus de détails concernant l'implémentation matérielle et logicielle du système embarquée, ainsi que les détails électrique de la carte mère BG-17)

Ci-dessous une représentation détaillée des fonctions exécutées lors de l'appui des différents boutons de cette interface:

- SW1 : Ce bouton force un niveau logique zéro sur la ligne INRH de la carte mère BG15 de l'ascenseur durant 50 ms. Ceci simule l'appui sur le bouton

poussoir SW1 de la carte BG17. Visuellement ceci qui permet d'incrémenter le premier digit.

- SW2 : Ce bouton force un niveau logique zéro sur la ligne INRB de la carte mère BG15 de l'ascenseur durant 50 ms. Ceci simule l'appui sur le bouton poussoir SW2 de la carte BG17. Visuellement ceci qui permet d'incrémenter le second digit.
- SW3: Chaque appui sur ce bouton complémente l'état logique de la ligne MODIF de la carte mère BG15. Visuellement nous verrons la modification du texte situé à côté du bouton SW3 de «MODIF » à « ADR/DON » et inversement.
- SW4 : Chaque appui sur ce bouton complémente l'état logique de la ligne PAR/RAM de la carte mère BG15. Visuellement nous verrons la modification du texte situé à côté du bouton SW4 de «PAR » à « RAM » et inversement.
- Store: Ce bouton permet de simuler un forçage un appui simultané des deux boutons SW1 et SW2 pendant 1 seconde (fonctionnement nécessaire pour le stockage de paramètres sur la carte mère BG-15).
- Reset : Ce bouton permet un reset global (*power on reset*) de la carte mère de l'ascenseur en coupant son alimentation 24V pendant 5 s.

2.2.4-Forme D : Elevators (figure 4.6)



The screenshot shows a web interface titled "Elevators" with a "Logout" link in the top right corner. Below the title, there is a blue instruction: "make sure to fill all information". The form contains several input fields: "Name:", "Host name:", "Technical person:", and "Telephone:", each followed by a text input box. Below these is a larger "Comment" text area. At the bottom of the form, there are four buttons: "Reset", "Register", "modify", and "List of elevator". A "Home" link with a house icon is located at the bottom center of the page.

Fig.4.6 : Interface d'ajout et de mise à jour des informations de l'ascenseur dans la base de données.

Cette page permet de modifier la liste des ascenseurs disponibles dans la base de données : ajout ou blocage

Les informations à remplir lors d'un ajout sont les suivantes :

- "Name" : nom identifiant l'ascenseur.
- "Host name": nom de la hôte dynamique prédéfini dans le site « DynDNS.com » et configurée dans le routeur localement attaché à l'ascenseur (voir chapitre 3, Section 8).
- "Technicalpersonel": nom de la personne responsable du service de l'ascenseur chez le client de la société « MyzerGroup ».
- "Telephone": numéro du téléphone cette personne.
- "Comment": commentaire à propos de l'ascenseur (si nécessaire).

Après avoir fini le remplissage des informations concernant le nouvel ascenseur, nous cliquons sur le bouton "Register" pour valider.

Le bouton "reset" sert à réinitialiser à blanc le formulaire.

Le bouton "modify" permet de modifier les informations d'un ascenseur déjà présent dans la base de données. Dans la case correspondant à « Name » apparaît une liste déroulante des noms des ascenseurs déjà présents dans la base de données. Dans ce cas l'utilisateur est capable de modifier les informations de l'ascenseur ou de bloquer son accès.

Si une case est laissée vide, à l'exception de la case "comment", le système envoie un message pour remplir les champs manquants.

Le bouton "Home" sert à quitter cet écran et revenir à la forme B "switchboard". Le bouton "logout" sert à quitter l'application et revenir à la page initiale du site « www.myzergroup.com ».

2.2.5-Forme E : History (figure 4.7)

Cette forme affiche une liste de l'historique des erreurs correspondantes à l'ascenseur choisie dans la forme B "Switchboard" (Lors de l'appui sur le bouton "History").

L'utilisateur peut faire alors une recherche datée, de la date "From" à la date "To", en appuyant sur le bouton « go>> ».

Logout

History

Elevator Name: elv2-Date from: 2012-10-05-Date To: 2012-10-11

From: 2012-10-05 To: 2012-10-11 Go>>>>

Name	Date yyyy-mm-dd	Error Num	Error Description
elv2	2012-10-05 19:52:30	1	DEFAUT de masse.
elv2	2012-10-05 19:53:00	10	inversion du sens de rotation (DETECTION par capteur).
elv2	2012-10-11 16:41:31	11	INTERPRETATION impossible du capteur ou fusion du/des fusibles fu3, fu4.
elv2	2012-10-11 20:05:01	11	INTERPRETATION impossible du capteur ou fusion du/des fusibles fu3, fu4.

Home

Fig.4.7 : Liste historique des erreurs de l'ascenseur.

Les étapes à suivre sont les suivantes :

- sélectionner la date de début de la liste déroulante "From".
- sélectionner la date de fin de la liste déroulante "To".
- Appuyer sur le bouton "Go >>" pour afficher le résultat correspond aux critères sélectionnés.

Le bouton "Home" sert à quitter cet écran et revenir à la forme B "switchboard". Le bouton "logout" sert à quitter l'application et revenir à la page initiale du site « www.myzergroup.com ».

2.2.6-Forme F : Users (figure 4.8)

Cette page permet de modifier la liste des utilisateurs (personnel de MyzerGroup) disponibles dans la base de données : ajout ou blocage.

Les informations à remplir lors d'un ajout sont les suivantes :

- "Name" : nom de l'utilisateur.
- "Password": son mot de passe.
- "Confirm password": confirmation du mot de passe.
- "Telephone": numéro du téléphone de l'utilisateur.

- "Type": type d'utilisateurs : administrateur ou utilisateurs simple
- "Comment": commentaire à propos de l'utilisateur (si nécessaire).



Fig.4.8 : Interface des informations des utilisateurs dans la base de données.

Un administrateur peut accéder à toutes les fonctions présentées dans le système d'interfaçage. Un simple utilisateur peut accéder uniquement aux pages: "monitor", "List of error", "Table of parameter" et "History". Ainsi uniquement l'administrateur possède le droit d'ajouter ou de bloquer un utilisateur dans la base de données.

Après avoir fini le remplissage des informations concernant le nouvel utilisateur, nous cliquons sur le bouton "Register" pour valider.

Le bouton "reset" sert à réinitialiser à blanc le formulaire.

Si une case est laissée vide, à l'exception de la case "comment", le système envoie un message pour remplir les champs manquants.

Le bouton "modify" permet de modifier les informations d'un utilisateur déjà présent dans la base de données. Dans la case correspondant à « Name » apparaît une liste déroulante des noms des utilisateurs déjà présents dans la base de données. Dans ce cas l'utilisateur (un administrateur) est capable de modifier les informations de l'utilisateur ou de bloquer son accès.

Le bouton "Home" sert à quitter cet écran et revenir à la forme B "switchboard". Le bouton "logout" sert à quitter l'application et revenir à la page initiale du site « www.myzergroup.com ».

3-Conclusion

Dans ce chapitre nous avons illustré brièvement les différents langages de programmation utilisés pour réaliser notre « interfaçage web ». Nous avons ensuite présenté en détail toutes les formes réalisées dans l'interface web, ainsi que toutes les fonctionnalités implémentées pour le contrôle à distance de l'ascenseur et la gestion de la base de données.

Dans le chapitre suivant nous allons passer à l'implémentation pratique du système embarquée réalisé et permettant d'exécuter les différentes tâches de contrôle et de diagnostic de l'ascenseur à distance.

Chapitre 5 : Implémentation du système embarqué

Ce chapitre présente l'implémentation matériel et logiciel de notre système embarqué. Sa tâche est de fournir une interface d'alerte (en cas d'erreur) et de diagnostic préliminaire à distance de l'ascenseur, à travers une interface Web sécurisée (tout en conservant l'accès local)

Notre système remplace ainsi la carte BG-17 (carte locale de commande et diagnostique de l'ascenseur enfichée sur la carte mère BG15) fournie par le constructeur Autinor, et permet un accès à distance.

Le personnel de « myzergroup » peut alors accéder à travers l'interface Web présentée dans le chapitre 4 pour faire un diagnostic préliminaire avant l'envoi de son personnel de service.

La figure 5.1 présente le schéma électrique de la carte réalisée. Le serveur Web embaqué est totalement implémenté dans le microcontrôleur PIC18F87J60.

Pour l'alimentation et les connexions électriques entre le PIC et le connecteur RJ45, nous nous sommes inspirés du schéma électrique de la carte de développement LV18Fj de la société mikroelektronika.

Du côté ascenseur de l'ascenseur, nous avons étudié le schéma électrique de sa carte mère BG-15 (interface avec BG-17) afin de réaliser un circuit d'interfaçage robuste, et fournir toutes les fonctionnalités de la carte BG-17 ainsi que quelques fonctionnalités supplémentaires tel que la possibilité de reset de la carte mère BG-15 à partir de notre interface web.

Ce chapitre présente les différentes parties du circuit réalisé, les fonctionnalités fournies ainsi que les organigrammes des différents programmes (commande et serveur web intégré)

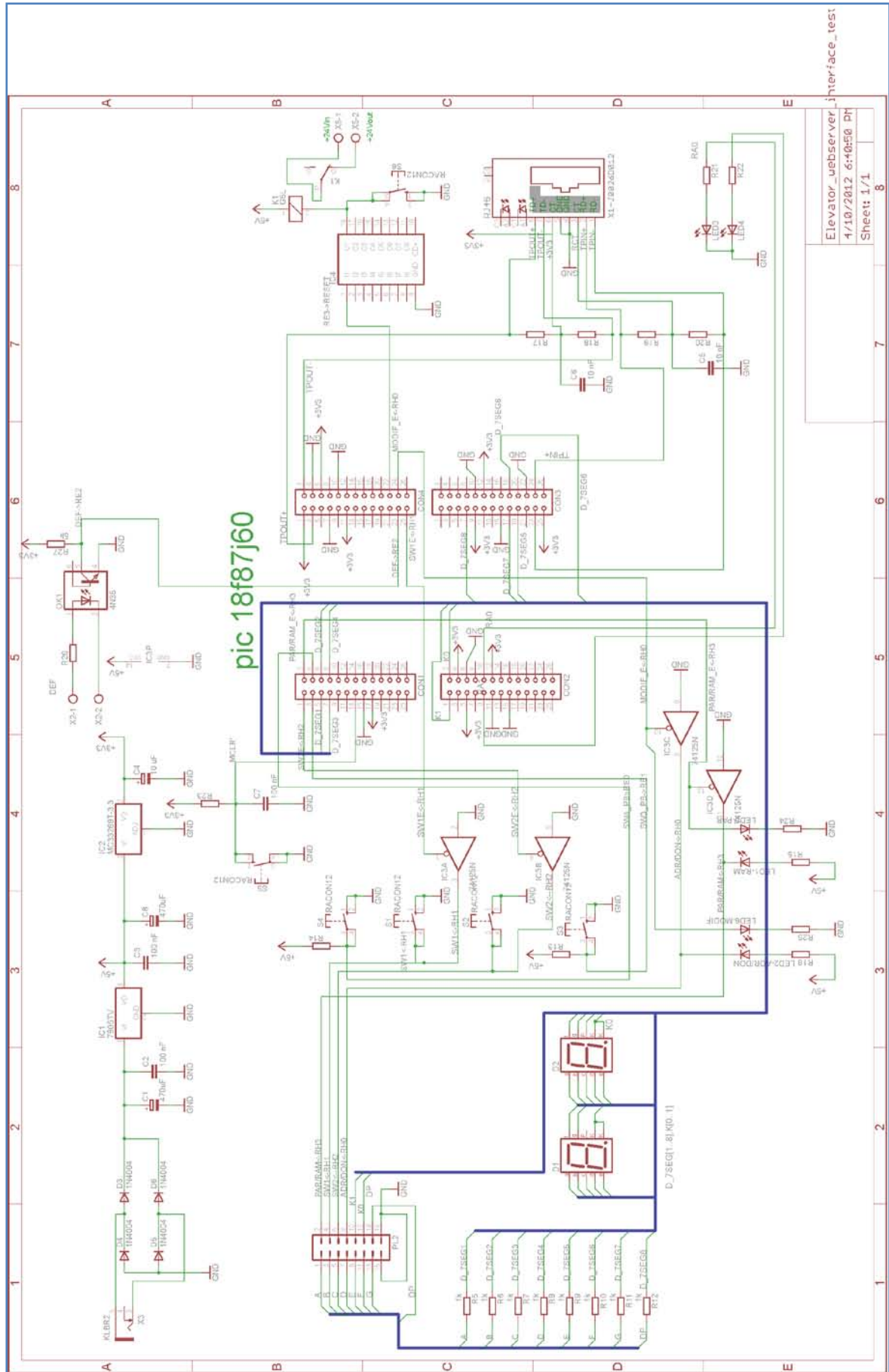


Fig.5.1 : Schéma électrique de la carte réalisée.

Elevator_webserver_interface_test
4/10/2012 6:40:50 PM
Sheet: 1/1

1-Circuit d'alimentation

Tout montage électronique nécessite une alimentation pour fonctionner. Notre montage nécessite les deux types d'alimentations suivantes:

- 5 V (relais et quelques composants interfaçant la carte mère BG-17).
- 3,3 V pour alimenter le PIC et son interface réseau.

1.1-Circuit 5V

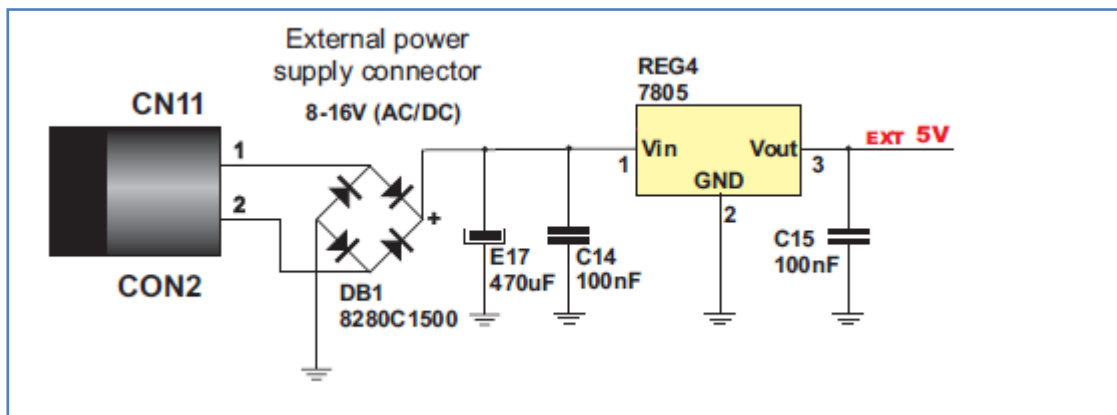


Fig.5.2 : Schéma du redresseur double alternance et du régulateur de tension 5 V.

Nous avons utilisé un redresseur double alternance (Pont de Gretz). Ceci permet une alimentation AC compris entre 7 V et 23 V, ou DC comprise entre 9 V et 32 V.

Le régulateur de tension LM7805 est utilisé pour obtenir une tension régulée 5 V à sa sortie. La figure 5.2 représente le schéma électrique du redresseur et du régulateur LM7805.

1.2-Circuit 3,3 V

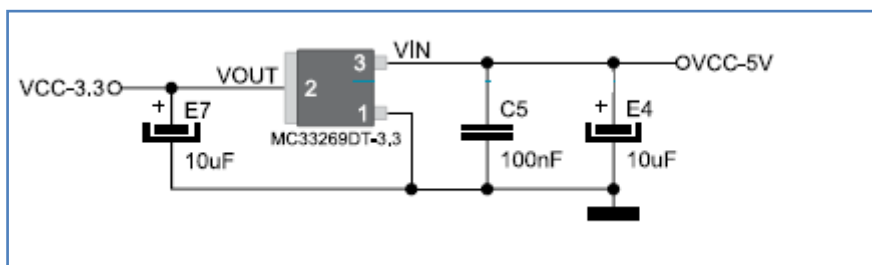


Fig.5.3 : Schéma du régulateur de tension 3,3V.

Le microcontrôleur PIC et son interface réseau doivent être alimentés par une tension de 3,3 V. Nous avons utilisé un régulateur de tension MC33269DT-3.3 pour obtenir une tension de 3,3 V à partir de l'alimentation 5V.

Le MC33269DT-3.3 fournit en sortie une tension stable de 3,3V avec une intensité de 800 mA. En outre il possède une protection thermique. Le schéma de la figure 4.2 donne un exemple de connexion du régulateur MC33269DT-3.3.

La figure 5.4 ci-dessous représente le schéma d'alimentation réalisé dans notre projet afin que les 2 types d'alimentations demandé soient disponibles

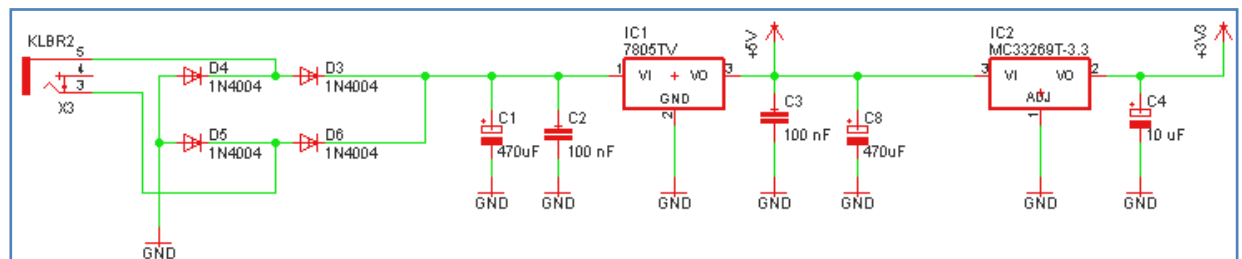


Fig.5.4 : Schématique d'alimentation de la carte.

2-Reset (carte mère BG-15 et système embarqué)

2.1-Reset de la carte mère de l'ascenseur (figure 5.5)

Cette fonctionnalité donne la possibilité d'un reset globale (*power on reset*) de la carte mère BG-15de l'ascenseur. Elle a été implémentée de deux manières différentes:

Locale : à travers un bouton poussoir sur la carte réalisée.

Distante : à partir de notre interface web (chapitre 4).

Cette fonctionnalité est nouvelle par rapport à la carte BG-17 du constructeur Autinor. Elle a été demandée par le personnel de « myzergroup » parce que selon leurs expériences, il y a quelques erreurs de la carte mère qui nécessitent uniquement un simple reset pour les corriger. Ce qui leur permet une réparation du problème à distance sans aucun déplacement.

L'implémentation matérielle de cette fonctionnalité consiste à utiliser une relai 5 V pour couper l'alimentation 24 V de la carte mère de l'ascenseur. Pour commander ce

relais à l'aide du PIC, nous avons utilisé un circuit ULN2803A qui permet d'alimenter la bobine du relais avec un courant pouvant atteindre 500 mA et une de 50 V.

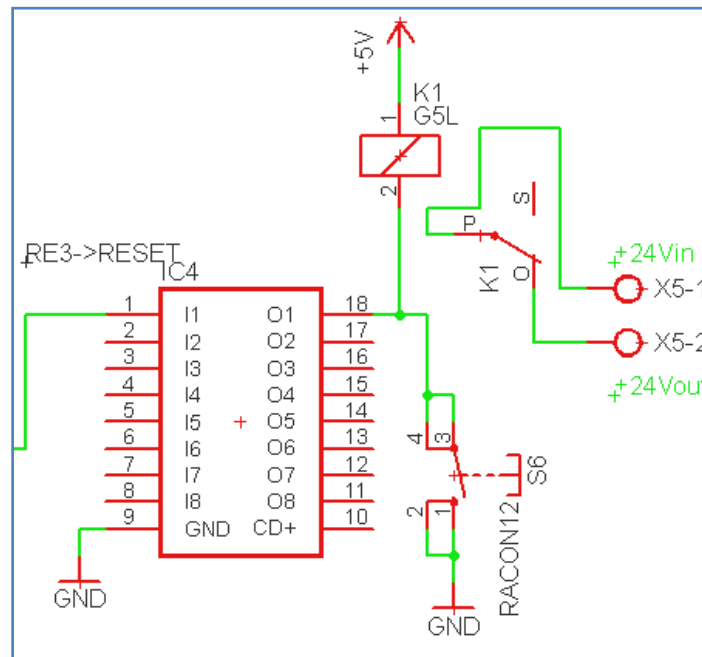


Fig.5.5 : Schéma électrique du reset.

2.2-Reset du PIC(MCLR)

La broche MCLR (Master Clear) permet de réaliser un reset manuel du microcontrôleur du PIC quand elle est placée à 0V. La figure ci-dessous représente le circuit utilisé pour le MCLR.

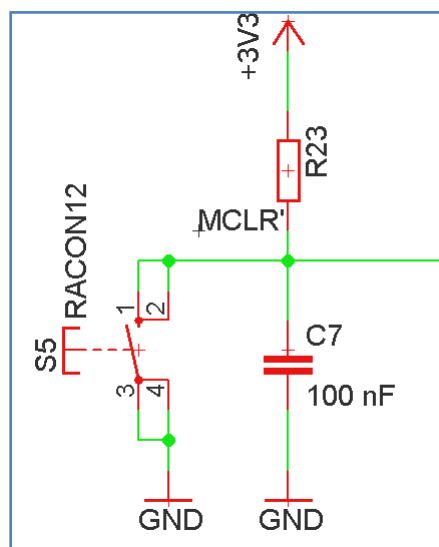


Fig.5.6: Schéma électrique du circuit MCLR du PIC.

3-Entrée Indicateur d'erreur

A l'occurrence d'une erreur (défaut) dans la carte mère de l'ascenseur, elle envoie en sortie un signal indicateur d'erreur de 24 V.

Ce signal a été utilisé comme entrée à notre système. Pour adapter les niveaux de tensions, on a utilisé un opto-coupleur 4N35 dont l'entrée est reliée au signal indicateur d'erreur 24V de la carte mère avec une résistance adéquate. La sortie de l'opto coupleur a été reliée à la broche RE2 (Port E, 3eme bit) du PIC.

A la suite de cette indication le PIC envoie un email à un personnel de la société« myzergroup » afin d'alerter de ce défaut.

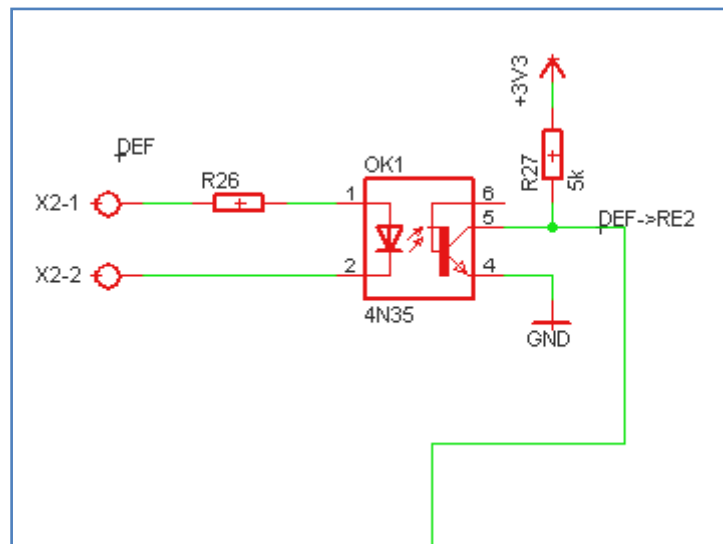


Fig.5.7 : Schéma électrique de l'indicateur d'erreur.

4-Interface réseau (Connecteur Ethernet RJ45)

Dans notre carte nous utilisons un connecteur Ethernet RJ45 (contenant les transformateurs d'impulsions) représenté dans la figure 5.8. Ce connecteur permet de relier le microcontrôleur à un LAN (Local Area Network).

Aucun réglage n'est nécessaire pour son utilisation. Sa connexion est réalisée à partir des broches TPOUT+, TPOUT-, TPIN+ et TPIN- du microcontrôleur (figure 5.9). L'activation des 2 LEDs à travers les broches RA0 et RA1 du PIC18F87j60 indiquent le fonctionnement de la connexion Ethernet.[13]

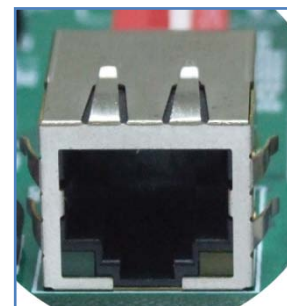


Fig.5.8 : Connecteur Ethernet RJ45.

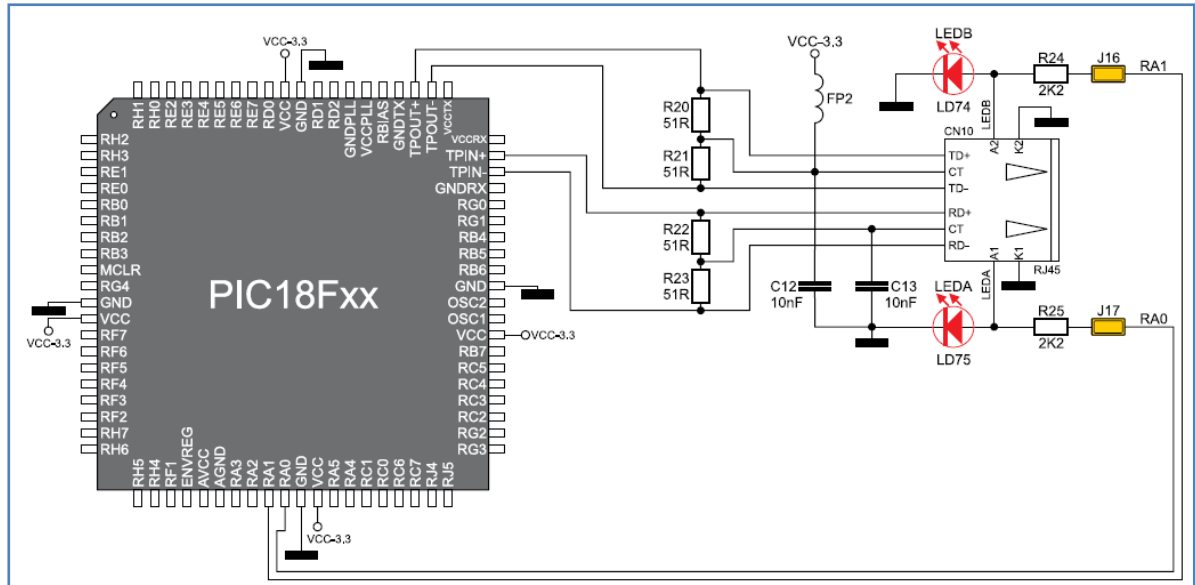


Fig.5.9 : Schéma de connexion de l'interface réseau.

5-Carte mère BG-15 de l'ascenseur et sa carte de configuration BG-17

La figure 5.10 présente le schéma de connexions externes de la carte mère de l'ascenseur nommé BG-15. Une petite carte appelé BG-17 s'enfiche sur cette carte

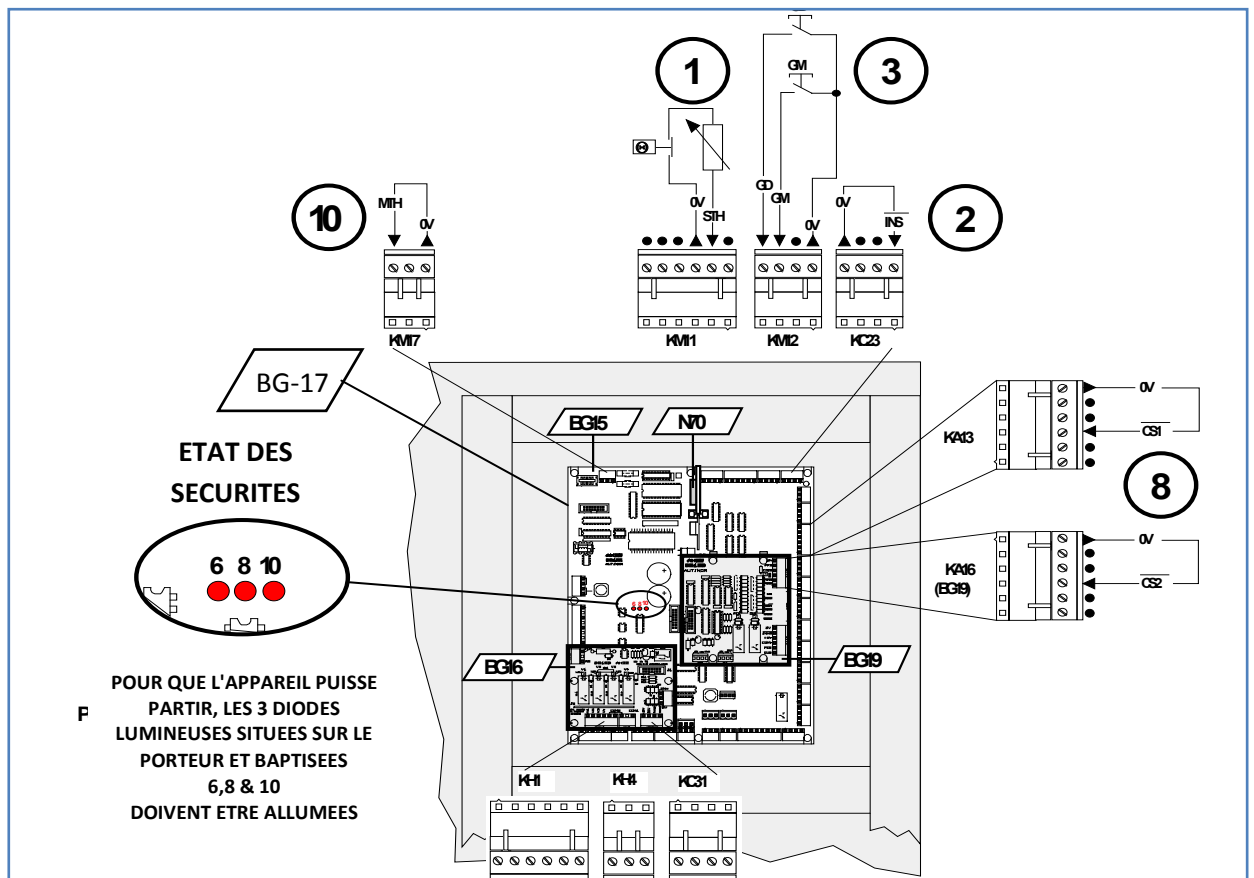


Fig.5.10 : Carte BG-15.

mère et permet sa configuration. La figure 5.11 présente l'interface utilisateur de la carte BG-17. Notre système embarqué va remplacer la carte BG-17 tout en fournissant les mêmes fonctionnalités (voire plus) en accès local et distant.

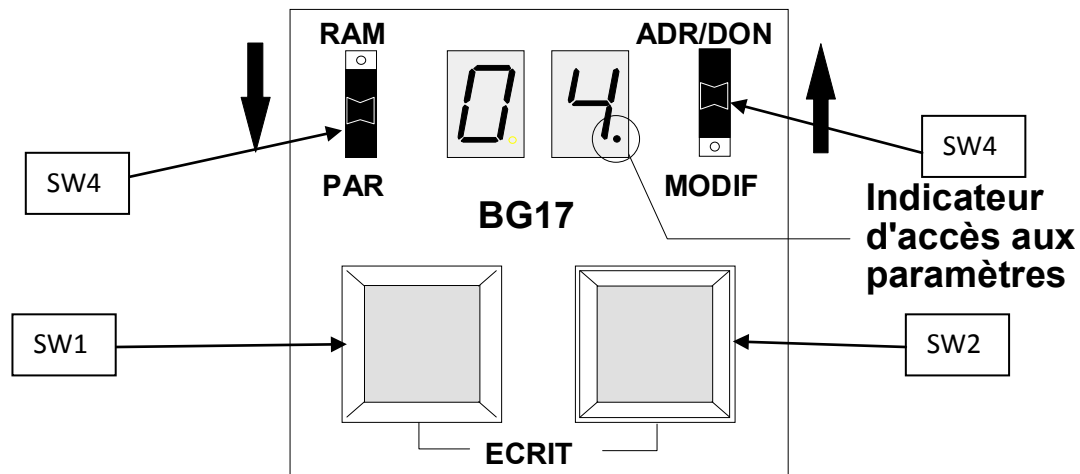


Fig.5.11 : Interface utilisateur de la carte BG-17

5.1-Fonctionnalités de la carte BG-17

Boutons poussoir :

Cette carte possède 2 boutons poussoirs et 2 boutons inverseurs qui permettent la configuration de la carte mère de l'ascenseur (figure 5.12).

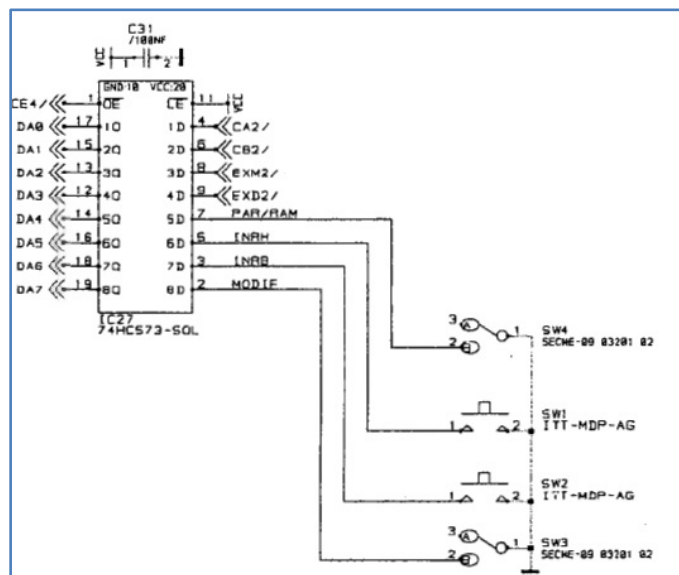


Fig.5.12 : Schéma électriques des boutons de la carte BG-17

Le fonctionnement de ces boutons est décrit ci-dessous :

- **SW1**: Ce bouton poussoir force un niveau logique zéro sur la ligne INRH de la carte mère BG15 de l'ascenseur (haute impédance lorsqu'il est relâché).
- **SW2** : Ce bouton poussoir force un niveau logique zéro sur la ligne INRB de la carte mère BG15 de l'ascenseur(haute impédance lorsqu'il est relâché).
- **SW3** : Permet de mettre la ligne MODIF à zéro ou à haute impédance.
- **SW4** : Permet de mettre la ligne PAR/RAM à zéro ou à haute impédance.

Ces 4 boutons ont été implémentés sur la carte de notre système embarqué (voir fig. 5.1 à la page 63).L'implémentation du niveau « haute impédance » a été obtenue en utilisant un buffer 3 états, le 74HCT125.Le choix de la famille HCT dans notre circuit électrique est justifié afin d'obtenir des niveaux de tensions compatibles avec les sorties 3.3V du microcontrôleur PIC 18F87J60.

Les fonctionnalités de ces 4 boutons seront de mêmes accessibles à partir de l'interface web réalisée (voir chapitre 4 figure 4.5) et ceci pour permettre une gestion à distance.

Afficheurs sept segments

Les deux afficheurs sept segments de la carte BG-17 sont à cathode commune (figure 5.13). Ces 2 afficheurs sont multiplexés à l'aide des sorties de sélection K0 et K1.

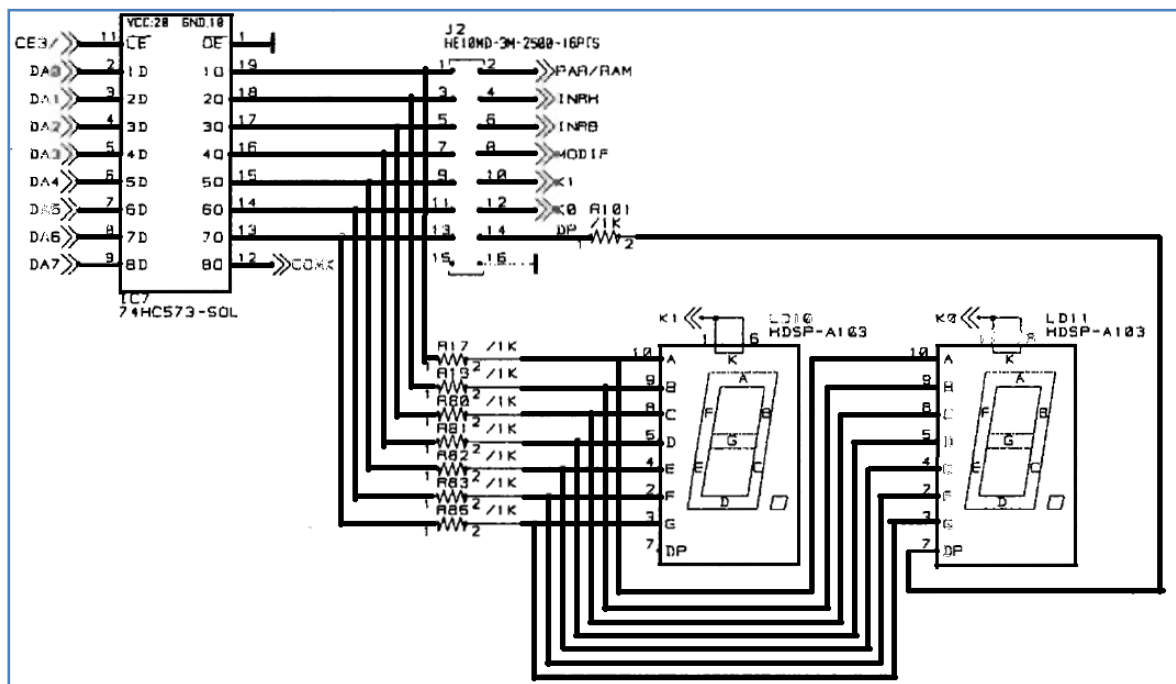


Fig.5.13 : Schéma de connexion des 2 afficheurs 7 segments de la carte BG-17.

Ces afficheurs indiquent, entre autres, le numéro de l'erreur. En cas de default, ce numéro va être transféré par email avec sa description au département de service de la société Myzer Group.

Ce numéro sera aussi affiché par l'interface web réalisée (voir chapitre 4 figure 4.5) pour permettre une gestion à distance.

6-Fonctionnement du Microcontrôleur

Le microcontrôleur PIC18F87j60 va assurer toutes les fonctionnalités présentées ci-dessus et va permettre leur accès à travers l'internet.

6.1-Les ports entrées/sortie

Plusieurs ports du microcontrôleur ont été utilisés pour exécuter ces. Le tableau V.I représente les différentes entrées et sorties du PIC utilisée sur notre carte et leur affectation.

Tab V.I : Liste des entrées et sortis du PIC.

Broche utilisé	Configuration	Fonction
PortB0	Entrée	7segments-D1 (A)
PortB1	Entrée	7segments-D2 (B)
PortB2	Entrée	7segments-D3 (C)
PortB3	Entrée	7segments-D4 (D)
PortB4	Entrée	7segments-D5 (E)
PortB5	Entrée	7segments-D6 (F)
PortB6	Entrée	7segments-D7 (G)
PortB7	Entrée	7segments-Dp (DP)
PortH4	Entrée	K0
PortH5	Entrée	K1
PortH1	Sortie	SW1
PortH2	Sortie	SW2
PortH0	Sortie	Led PAR-RAM
PortH3	Sortie	Led ADR/DON-MODIF
PortE1	Entrée	SW3 (bouton-poussoir)
PortE0	Entrée	SW4 (bouton-poussoir)
PortE2	Entrée	Indicateur d'erreur
PortE3	Sortie	Reset de la carte mère BG-15 à partir du logiciel
PortA0	Sortie	LedA (Ethernet)
PortA1	Sortie	LedB (Ethernet)

Le port A est configuré en sortie afin d'utiliser les deux broches A0 et A1 pour l'indication du transfert des données à travers le connecteur RJ45.

Le port B est utilisé comme entrée pour capturer les données envoyées par la carte mère BG-15 aux deux afficheurs sept segments.

Le port H est configuré comme sortie à l'exception des deux broches H4 et H5. Les broches de sorties sont utiliser pour simuler le fonctionnement des boutons de la carte BG-17. Les broches H4 et H5 sont utilisés pour lire l'état des deux bits de sélection K0 et K1 utilisés pour le multiplexage des deux afficheurs 7 segments.

Le port E est configuré comme entrée à l'exception de la broche E3. E0 et E1 sont utilisées pour lire les états locaux des deux boutons poussoir SW3 et SW4 (voir figure 5.1). En appuyant localement sur ces deux boutons, leurs états sont premièrement lus par le pic qui génère ultérieurement adéquate à envoyer vers la carte mère de l'ascenseur. Nous avons utilisé cette méthode afin d'avoir une synchronisation entre le fonctionnement manuelle de la carte et le fonctionnement à travers l'interface logiciel du web. La broche E2 est utilisée pour détecter l'indication de l'erreur (envoyée par la carte mère). Finalement la broche E3 est utilisée pour commander le système de reset de la carte mère.

6.2-Organigramme du programme principal (main)

La figure 5.14 représente l'organigramme complet de notre programme. Premièrement nous avons commencé par la configuration des principaux paramètres. A l'état initial du système les boutons et l'indicateur sont posés aux niveaux logiques 1. Nous avons encore configurés les paramètres de l'Ethernet : l'adresse IP, l'adresse MAC, l'adresse du routeur, du DNS et l'adresse masque IP ainsi que le mode de transfert Ethernet. Dans notre cas, c'est le mode full duplex qui a été utilisé.

Nous commençons par détecter les données et les sorties de multiplexages, k0 et k1, envoyées par la carte mère BG-15 aux deux afficheurs. Si k1 est égale à zéro nous mettons le contenu des données dans une variable aff1 et si k2 est égale à zéro dans la variable aff2.

L'afficheur 1 ne doit pas contenir un point décimal, c'est pourquoi nous l'avons éliminé dans notre programme.

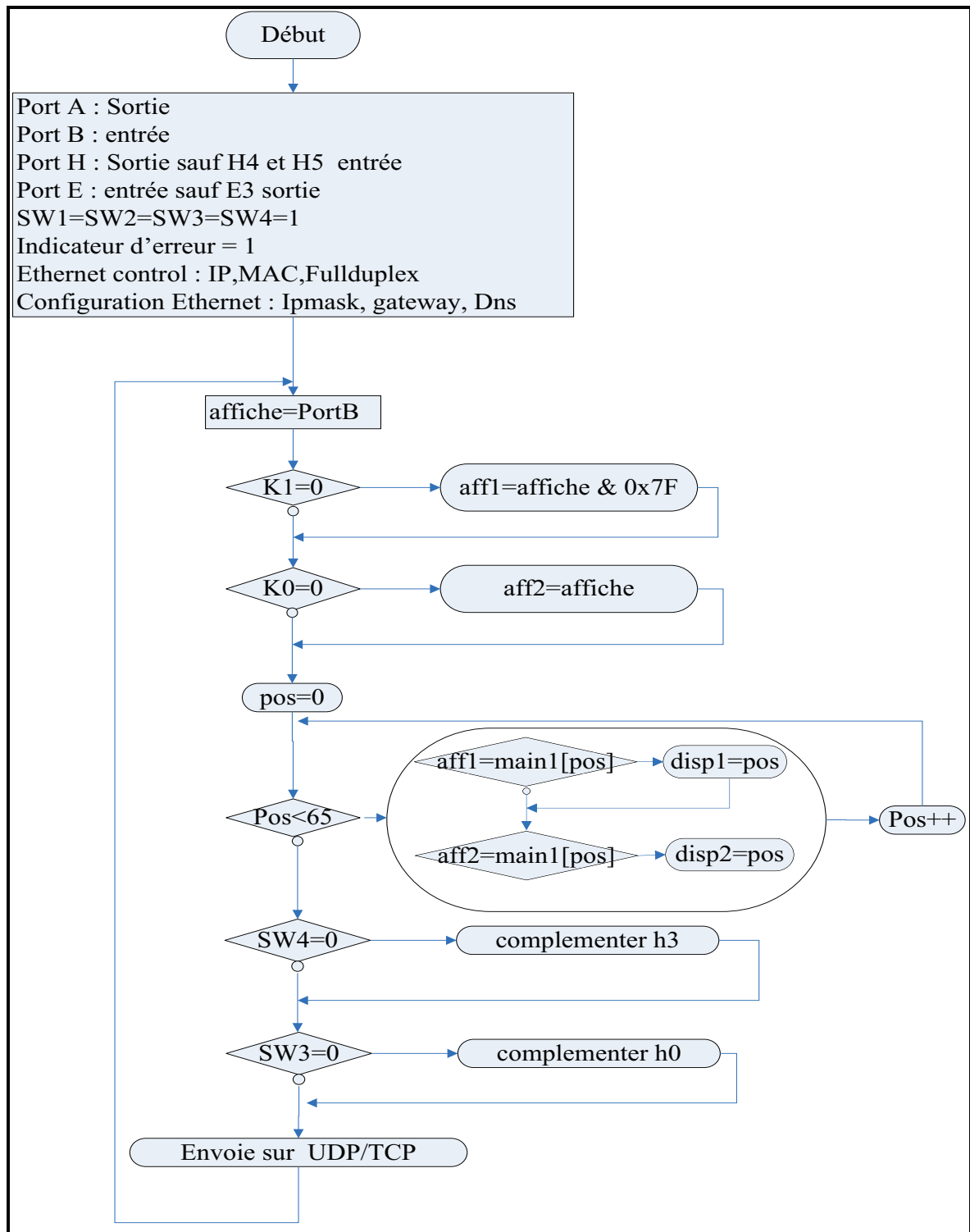


Fig.5.14 : Organigramme du programme principal (main).

Après une comparaison est faite entre le contenu de chacun des variables aff1 et aff2 et un vecteur appelé « main1 » contenant sous forme d'entier décimale toutes les données possibles que peuvent contenir chacun des afficheurs 7 segment. Cette comparaison permet de choisir la position, « dans le vecteur main1 », dans laquelle se trouve la valeur des afficheurs. Cette position est alors transmise à la page HTML qui

contient encore un vecteur identique à « main1 ». De cette manière, la page HTML est utilisée uniquement pour afficher le résultat sans faire aucune comparaison dans cette page (la taille de la page HTML étant limitée).

Ensuite nous avons lu les états des deux boutons SW3 et SW4 présents localement sur notre carte, et nous avons commandé en fonction des valeurs lues, les leds visualisant leurs états. Par exemple l'état initiale du bouton SW4 est RAM et la led1-RAM (voir figure 5.1) sur notre carte est allumée et la led5-PAR est éteinte ; en appuyant sur SW4 les états des leds sont complétés. Pour avoir une synchronisation entre la carte et le programme d'interfaçage web, nous avons transmis les commandes locales des deux boutons SW3 et SW4 au microcontrôleur qui, à son tour, commande les entrées de la carte mère de l'ascenseur et mémorise l'état de ces deux boutons.

Enfin nous avons transmis toutes ces données vers UPD/TCP pour être transférées à travers l'internet. L'organigramme de la fonction UPD/TCP est présenté dans la figure 5.14.

Nous commençons dans cette fonction par initialiser la variable entière « len » à zéro. Cette variable va correspondre à la longueur de la chaîne de caractère à transmettre. Cette chaîne correspond au vecteur dynamique « dyna » et va contenir toutes nos informations.

Le transfert de donnée se faisant uniquement à travers le port 8899, on doit commencer par vérifier si les informations arrivent à travers ce port. Sinon on reste dans une boucle d'attente.

Une fois des données reçues, on acquiert les 10 premiers octets. Les quatre premiers octets correspondent à la méthode de transmission. Nous avons utilisé uniquement la méthode "GET". Le cinquième octet indique le mode de fonctionnement qui va suivre.

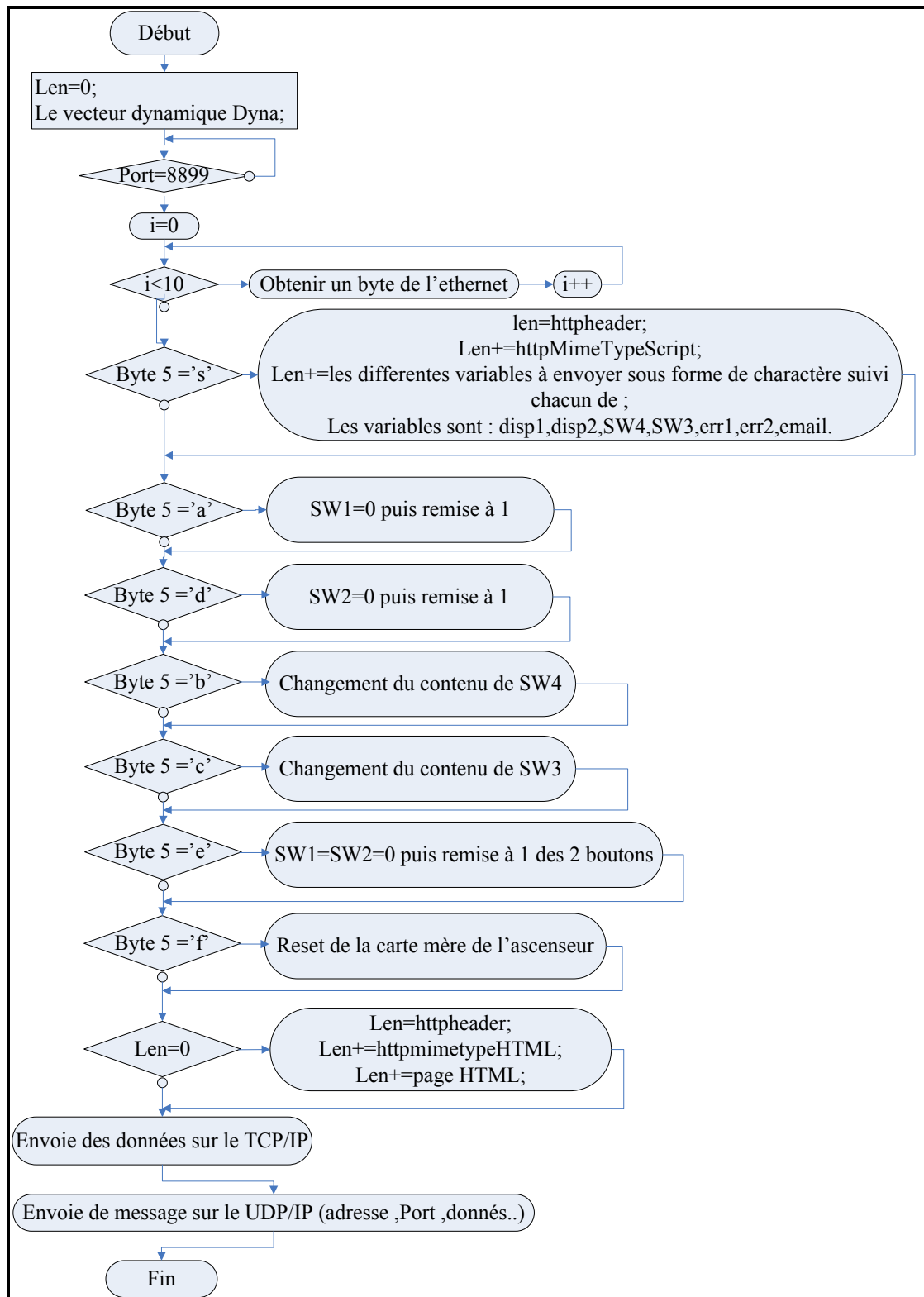


Fig.5.15 : Organigramme de la fonction de transmission sur l'internet.

Notre programme possède 8 modes de fonctionnement détaillés ci-dessous.

Mode 1: Si le contenu de l'octet 5 correspond au code du caractère 's'.Le programme envoie l'entête (header) et son type .Ceci permet l'affichage de la chaine de caractère sur la page d'interfaçage web. Puis le programme traite les différentes variables qui

vont être reçues sous forme de caractères en les stockant dans le vecteur dynamique « dyna ». Par exemple pour traiter la variable « SW4 », nous mettons dans le vecteur « dyna » la déclaration de SW4 sur la page HTML, 'var SW4', dans ce cas SW4 est défini comme variable puis nous ajoutons le contenu de SW4 sous forme de caractère et nous terminons la chaîne par un ';'. La même procédure est répétée pour les autres variables. Notons cependant que pour le cas particulier des 2 variables « disp1 » et « disp2 », le programme envoie la position du caractère dans un tableau « look up » (vecteur main1), et suivant cette position il affiche le caractère convenable. L'apparition d'un défaut est indiquée en positionnant la variable « email » à 1. Cette variable indique la nécessité d'envoyer un email au service de maintenance de « Myzer Group ». Les variables err1 et err2 contiennent le numéro de l'erreur à l'instant du défaut. Ce numéro est sauvegardé dans la base de données qui se trouve dans la page web afin d'avoir un historique concernant les erreurs de chacun des ascenseurs. Ce même numéro est aussi envoyé par email au service de maintenance.

Mode 2: Si le contenu de l'octet 5 correspond au code du caractère 'a', il indique une utilisation du bouton poussoir SW1 de l'« interface web ». Dans ce cas le système doit envoyer un niveau logique bas vers la ligne correspondante sur la carte mère de l'ascenseur pour une durée de 50 ms.

Mode 3: Si le contenu de l'octet 5 correspond au code du caractère 'd', il indique une utilisation du bouton poussoir SW2 de l'« interface web ». Dans ce cas le système doit envoyer un niveau logique bas vers la ligne correspondante sur la carte mère de l'ascenseur pour une durée de 50 ms.

Mode 4: Si le contenu de l'octet 5 correspond au code du caractère 'b', il indique une utilisation du bouton poussoir SW4 de l'« interface web ». Dans ce cas le système doit compléter la ligne correspondante sur la carte mère de l'ascenseur et mettre à jour les leds de visualisation correspondantes.

Mode 5: Si le contenu de l'octet 5 correspond au code du caractère 'c', il indique une utilisation du bouton poussoir SW3 de l'« interface web ». Dans ce cas le système doit compléter la ligne correspondante sur la carte mère de l'ascenseur et mettre à jour les leds de visualisation correspondantes.

Mode 6: Si le contenu de l'octet 5 correspond au code du caractère 'e', il indique une utilisation du bouton poussoir « store » de l'« interface web ». Ce bouton simule

l'appui simultané des deux boutons SW1 et SW2 pour une durée d'une seconde. Ce bouton apparaît uniquement sur le programme d'interfaçage et n'existe pas sur la carte de notre système embarqué (ni sur la carte BG-17) car manuellement nous pouvons appuyer sur les deux boutons en même temps.

Mode 7: Si le contenu de l'octet 5 correspond au code du caractère 'f'. Le programme effectue un reset (Power on Reset) de la carte mère de l'ascenseur en coupant son alimentation 24V.

Mode 8: Si la valeur de la variable « len » est égale à zéro, elle indique l'établissement de la connexion web et la nécessité de visualiser la page HTML. Notre serveur embarqué envoie le header, son type et la page HTML principale.

Après le traitement des différents modes, notre programme transmet les données sur le TCP/IP pour être envoyées sur le réseau, enfin il retourne le contenu de l'entier len.

Le programme utilise ensuite la fonction UDP pour transmettre quelques informations tels le numéro d'adresse IP, le numéro du port de la source et de la destination, ces informations sont stockées dans le vecteur dynamique 'dyna'. Ces informations sont transmises pour permettre au destinataire de reconnecter le système à travers l'UDP en cas de coupure de la connexion TCP.

Conclusion générale

Nous avons présenté dans ce mémoire la réalisation d'un système embarqué de surveillance et contrôle à distance, comprenant un serveur web embarqué, ainsi qu'une interface web sécurisée. Ce système, relié à la carte mère d'un ascenseur, nous a permis d'obtenir les fonctionnalités suivantes :

- a. La visualisation en temps réel de l'état actuel de l'ascenseur.
- b. La possibilité de configuration et paramétrage à distance de la carte mère de l'ascenseur.
- c. La génération d'un appel d'alerte en cas d'erreur avec indication de l'erreur en question.
- d. L'affichage et l'archivage de la liste des erreurs dans une base de données.

Grâce à ce système, nous pouvons faciliter et accélérer la procédure de réparation de l'ascenseur, voire même dans certain cas le réparer, sans avoir à se déplacer physiquement. L'accès à notre système est pratiquement universel : tout dispositif possédant une connexion internet et un navigateur Web peut être utilisé.

Notons enfin que malgré que notre applications a été limitée pour interfacer une carte mère d'un ascenseur, elle contient tous les éléments nécessaires, que ça soit au niveau matériel ou logiciel, pour implémenter des systèmes domotiques à distance, domaine en plein essor ces dernières années.

Annexes

Annexe I : Fiche technique du microcontrôleur PIC18F87J60(premières pages)

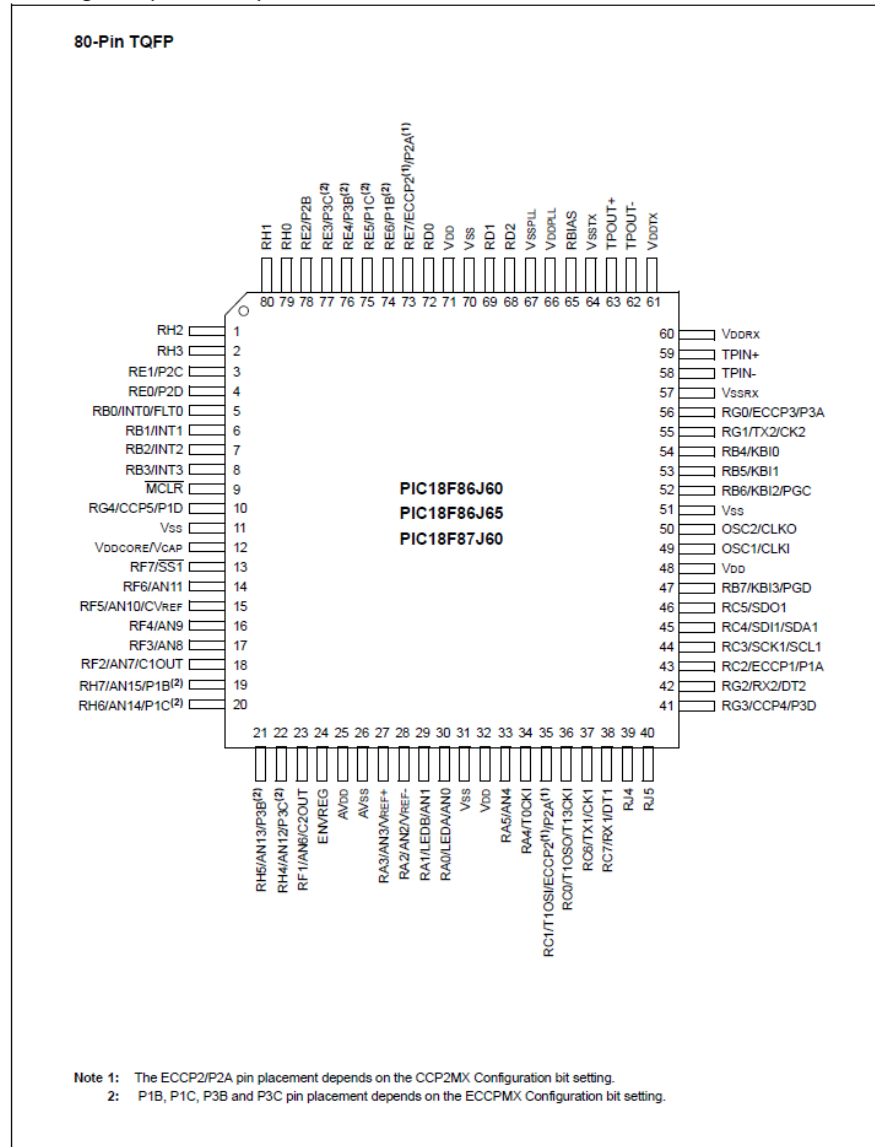


PIC18F97J60 Family Data Sheet

64/80/100-Pin High-Performance,
1-Mbit Flash Microcontrollers
with Ethernet

PIC18F97J60 FAMILY

Pin Diagrams (Continued)



PIC18F97J60 FAMILY

Table of Contents

1.0	Device Overview	7
2.0	Oscillator Configurations	39
3.0	Power-Managed Modes	45
4.0	Reset	53
5.0	Memory Organization	67
6.0	Flash Program Memory	95
7.0	External Memory Bus	105
8.0	8 x 8 Hardware Multiplier	117
9.0	Interrupts	119
10.0	I/O Ports	135
11.0	Timer0 Module	163
12.0	Timer1 Module	167
13.0	Timer2 Module	173
14.0	Timer3 Module	175
15.0	Timer4 Module	179
16.0	Capture/Compare/PWM (CCP) Modules	181
17.0	Enhanced Capture/Compare/PWM (ECCP) Modules	189
18.0	Ethernet Module	205
19.0	Master Synchronous Serial Port (MSSP) Module	255
20.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)	301
21.0	10-Bit Analog-to-Digital Converter (A/D) Module	325
22.0	Comparator Module	335
23.0	Comparator Voltage Reference Module	341
24.0	Special Features of the CPU	345
25.0	Instruction Set Summary	359
26.0	Development Support	409
27.0	Electrical Characteristics	413
28.0	Packaging Information	449
29.0	DC and AC Characteristics Graphs and Tables	455
	Appendix A: Revision History	457
	Appendix B: Device Differences	457
	Index	459
	The Microchip Web Site	471
	Customer Change Notification Service	471
	Customer Support	471
	Reader Response	472
	Product Identification System	473

PIC18F97J60 FAMILY

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC18F66J60
- PIC18F66J65
- PIC18F67J60
- PIC18F86J60
- PIC18F86J65
- PIC18F87J60
- PIC18F96J60
- PIC18F96J65
- PIC18F97J60

This family introduces a new line of low-voltage devices with the foremost traditional advantage of all PIC18 microcontrollers – namely, high computational performance and a rich feature set at an extremely competitive price point. These features make the PIC18F97J60 family a logical choice for many high-performance applications where cost is a primary consideration.

1.1 Core Features

1.1.1 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC18F97J60 family offer five different oscillator options, allowing users a range of choices in developing application hardware. These options include:

- Two Crystal modes, using crystals or ceramic resonators.
- Two External Clock modes, offering the option of a divide-by-4 clock output.
- A Phase Lock Loop (PLL) frequency multiplier, available to the external oscillator modes, which allows clock speeds of up to 41.667 MHz.
- An internal RC oscillator with a fixed 31 kHz output which provides an extremely low-power option for timing-insensitive applications.

The internal oscillator block provides a stable reference source that gives the family additional features for robust operation:

- **Fail-Safe Clock Monitor:** This option constantly monitors the main clock source against a reference signal provided by the internal oscillator. If a clock failure occurs, the controller is switched to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.
- **Two-Speed Start-up:** This option allows the internal oscillator to serve as the clock source from Power-on Reset, or wake-up from Sleep mode, until the primary clock source is available.

1.1.2 EXPANDED MEMORY

The PIC18F97J60 family provides ample room for application code, from 64 Kbytes to 128 Kbytes of code space. The Flash cells for program memory are rated to last 100 erase/write cycles. Data retention without refresh is conservatively estimated to be greater than 20 years.

The PIC18F97J60 family also provides plenty of room for dynamic application data with 3808 bytes of data RAM.

1.1.3 EXTERNAL MEMORY BUS

In the unlikely event that 128 Kbytes of memory are inadequate for an application, the 100-pin members of the PIC18F97J60 family also implement an External Memory Bus (EMB). This allows the controller's internal program counter to address a memory space of up to 2 Mbytes, permitting a level of data access that few 8-bit devices can claim. This allows additional memory options, including:

- Using combinations of on-chip and external memory up to the 2-Mbyte limit
- Using external Flash memory for reprogrammable application code or large data tables
- Using external RAM devices for storing large amounts of variable data

1.1.4 EXTENDED INSTRUCTION SET

The PIC18F97J60 family implements the optional extension to the PIC18 instruction set, adding eight new instructions and an Indexed Addressing mode. Enabled as a device configuration option, the extension has been specifically designed to optimize reentrant application code originally developed in high-level languages, such as C.

1.1.5 EASY MIGRATION

Regardless of the memory size, all devices share the same rich set of peripherals, allowing for a smooth migration path as applications grow and evolve.

PIC18F97J60 FAMILY

1.2 Other Special Features

- **Communications:** The PIC18F97J60 family incorporates a range of serial communication peripherals, including up to two independent Enhanced USARTs and up to two Master SSP modules, capable of both SPI and I²C™ (Master and Slave) modes of operation. In addition, one of the general purpose I/O ports can be reconfigured as an 8-bit Parallel Slave Port for direct processor-to-processor communications.
- **CCP Modules:** All devices in the family incorporate two Capture/Compare/PWM (CCP) modules and three Enhanced CCP (ECCP) modules to maximize flexibility in control applications. Up to four different time bases may be used to perform several different operations at once. Each of the three ECCP modules offers up to four PWM outputs, allowing for a total of twelve PWMs. The ECCP modules also offer many beneficial features, including polarity selection, programmable dead time, auto-shutdown and restart and Half-Bridge and Full-Bridge Output modes.
- **10-Bit A/D Converter:** This module incorporates programmable acquisition time, allowing for a channel to be selected and a conversion to be initiated without waiting for a sampling period and thus, reducing code overhead.
- **Extended Watchdog Timer (WDT):** This enhanced version incorporates a 16-bit prescaler, allowing an extended time-out range. See **Section 27.0 "Electrical Characteristics"** for time-out periods.

1.3 Details on Individual Family Members

Devices in the PIC18F97J60 family are available in 64-pin, 80-pin and 100-pin packages. Block diagrams for the three groups are shown in Figure 1-1, Figure 1-2 and Figure 1-3.

The devices are differentiated from each other in four ways:

1. Flash program memory (three sizes, ranging from 64 Kbytes for PIC18FX6J60 devices to 128 Kbytes for PIC18FX7J60 devices).
2. A/D channels (eleven for 64-pin devices, fifteen for 80-pin pin devices and sixteen for 100-pin devices).
3. Serial communication modules (one EUSART module and one MSSP module on 64-pin devices, two EUSART modules and one MSSP module on 80-pin devices and two EUSART modules and two MSSP modules on 100-pin devices).
4. I/O pins (39 on 64-pin devices, 55 on 80-pin devices and 70 on 100-pin devices).

All other features for devices in this family are identical. These are summarized in Table 1-1, Table 1-2 and Table 1-3.

The pinouts for all devices are listed in Table 1-4, Table 1-5 and Table 1-6.

PIC18F97J60 FAMILY

TABLE 1-1: DEVICE FEATURES FOR THE PIC18F97J60 FAMILY (64-PIN DEVICES)

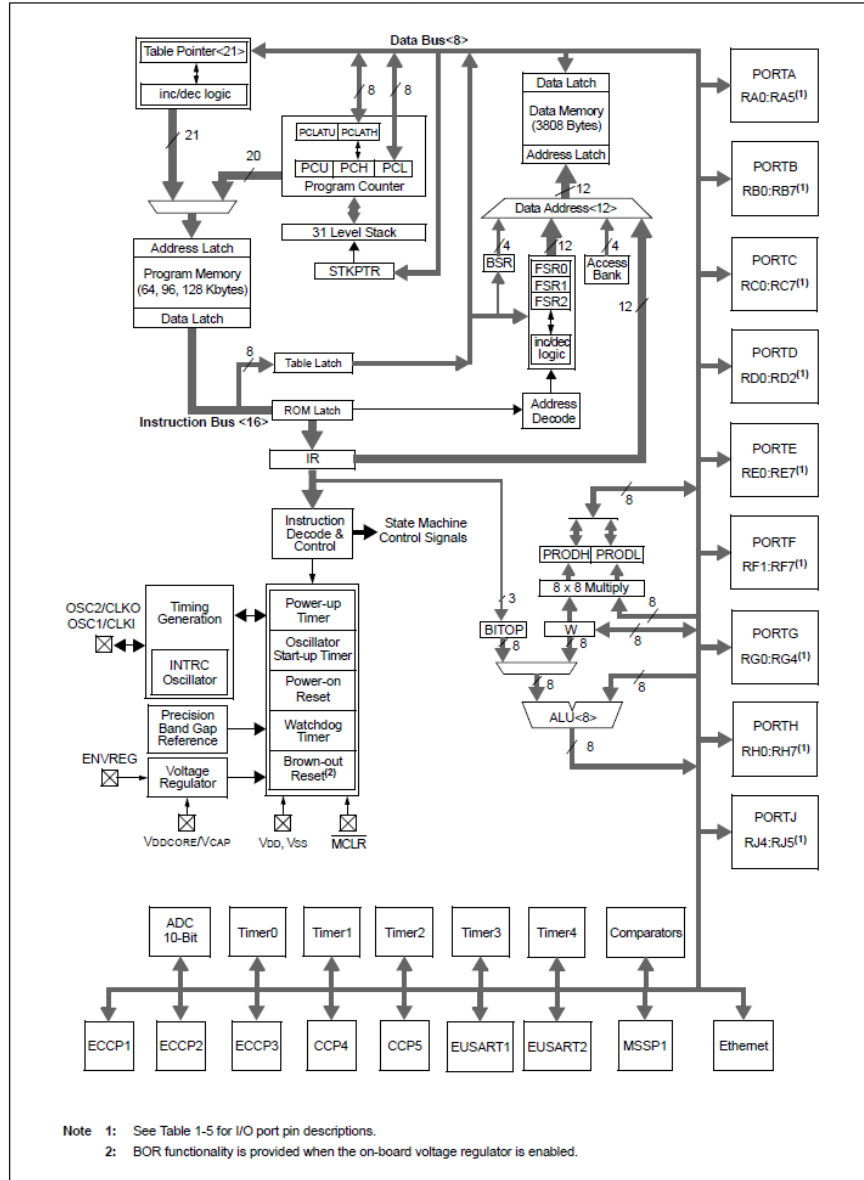
Features	PIC18F66J60	PIC18F66J65	PIC18F67J60
Operating Frequency	DC – 41.667 MHz	DC – 41.667 MHz	DC – 41.667 MHz
Program Memory (Bytes)	64K	96K	128K
Program Memory (Instructions)	32764	49148	65532
Data Memory (Bytes)	3808		
Interrupt Sources	26		
I/O Ports	Ports A, B, C, D, E, F, G		
I/O Pins	39		
Timers	5		
Capture/Compare/PWM Modules	2		
Enhanced Capture/Compare/PWM Modules	3		
Serial Communications	MSSP (1), Enhanced USART (1)		
Ethernet Communications (10Base-T)	Yes		
Parallel Slave Port Communications (PSP)	No		
External Memory Bus	No		
10-Bit Analog-to-Digital Module	11 Input Channels		
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow, MCLR, WDT (PWRT, OST)		
Instruction Set	75 Instructions, 83 with Extended Instruction Set Enabled		
Packages	64-Pin TQFP		

TABLE 1-2: DEVICE FEATURES FOR THE PIC18F97J60 FAMILY (80-PIN DEVICES)

Features	PIC18F86J60	PIC18F86J65	PIC18F87J60
Operating Frequency	DC – 41.667 MHz	DC – 41.667 MHz	DC – 41.667 MHz
Program Memory (Bytes)	64K	96K	128K
Program Memory (Instructions)	32764	49148	65532
Data Memory (Bytes)	3808		
Interrupt Sources	27		
I/O Ports	Ports A, B, C, D, E, F, G, H, J		
I/O Pins	55		
Timers	5		
Capture/Compare/PWM Modules	2		
Enhanced Capture/Compare/PWM Modules	3		
Serial Communications	MSSP (1), Enhanced USART (2)		
Ethernet Communications (10Base-T)	Yes		
Parallel Slave Port Communications (PSP)	No		
External Memory Bus	No		
10-Bit Analog-to-Digital Module	15 Input Channels		
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow, MCLR, WDT (PWRT, OST)		
Instruction Set	75 Instructions, 83 with Extended Instruction Set Enabled		
Packages	80-Pin TQFP		

PIC18F97J60 FAMILY

FIGURE 1-2: PIC18F86J60/86J65/87J60 (80-PIN) BLOCK DIAGRAM



Annexe II : Offres d'un IP fixe et dynamique

1-Offre d'un IP fixe proposé par la société Cyberia

CYBERIA LEBANON

Transmog Inc s.a.l. -

Cyberia
7th floor, Hamra Square Bldg. Block A
Hamra Street, Hamra
Beirut, Lebanon
Tel: + (961) 1 744 101
Fax: + (961) 1 744 102

Friday, December 09, 2011
Re: Your request for Full Time Internet Connection

Mr. Hassan Nasereddine
Beirut, Lebanon
E-mail: hassansd1@muzergroup.com
Tel: + (961) 3 693870

Dear Mr. Nasereddine

QUOTE FOR AN INTERNET LINK

Following your request to inquire about the SOHO Broadband Internet Connection, please find attached Cyberia's proposal to provide you with the above mentioned service.

Should you need any additional information, or would like to inquire about any item in the proposal, please do not hesitate to contact me.

Sincerely,

Wassim El-Khoury
Account Manager
Transmog Inc s.a.l.

Transmog Inc. s.a.l.
Capital: LBP 8102.158.000
C.R.: 68241
Hamra Square center
Hamra street
P. O. Box: 113-6141 Hamra
Beirut - Lebanon
Phone: (+961-1) 744 101
Fax: (+961-1) 744 102
info@cyberia.net.lb
www.cyberia.net.lb



SOHO - Broadband Internet Connection

Service Description

The Small Office – Home Office broadband internet connection solution is a dedicated Asymmetric link between the customer and Cyberia. The bandwidth itself at Cyberia is shared among all the SOHO customers at a contention ratio set to deliver the best performance and ensure burstability of the speed up to 512 Kbps on the upload traffic and 2048 Kbps on the download traffic.

This solution is best suitable for small businesses, whether run from home or office who need to be always connected to the Internet at high speed. All applications available on the Internet including VPN connections are available over the SOHO link without any restrictions.

Service Characteristics

- Up to 2 Mbps download speed
- Fixed real IPs
- Connection via a dedicated microwave link
- 24 hour support
- Installation in 7 business days (a site survey is required before the installation)

Advantages

Since the communication in a Full Time Connection is not routed through a switched telephone network like the normal dial-up services, various advantages are provided to frequent or heavy users including:

Lower costs

- Fixed Running Costs as against incrementing costs borne by dial-up subscribers especially when access to the Internet is required during those periods of the day when phone calls are charged by time.
- Diminishing Costs are experienced as users increase since a single SOHO connection may provide access to more than one user at a fixed cost whereas charges for calls tend to increase as the number of users increases.

Pricing

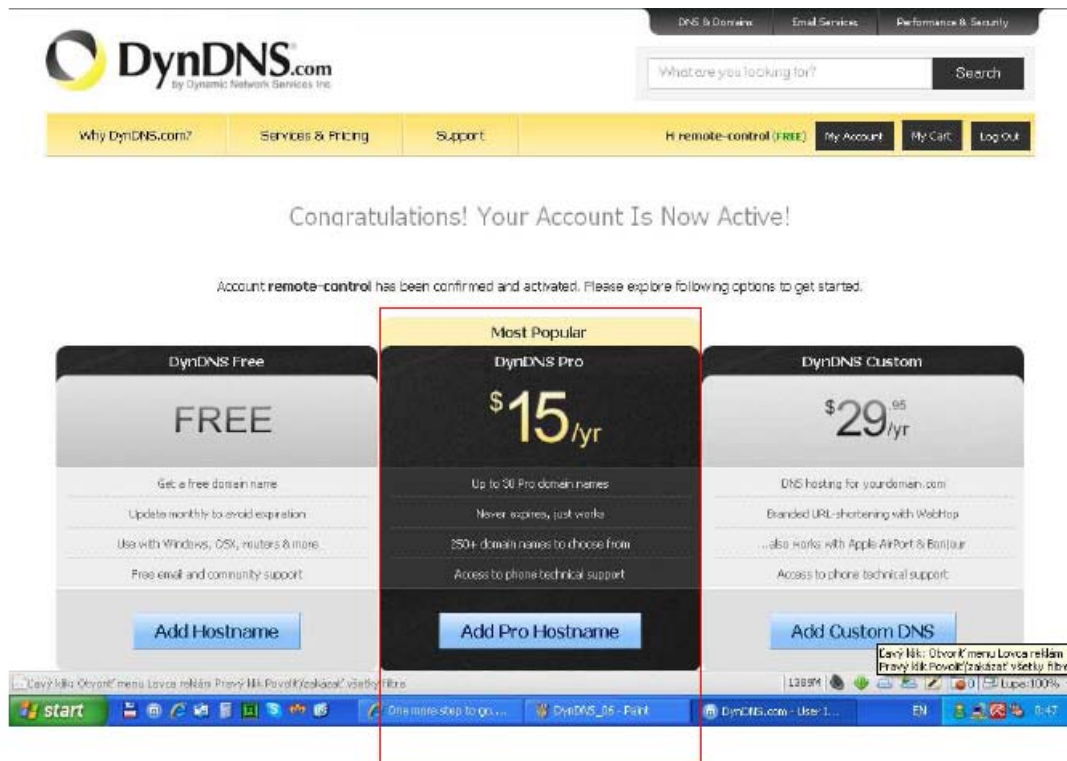
The table below summarizes the startup cost and the monthly recurring charges.

Item	Description	Price	
		256 Kbps up 1024 Kbps down	
Setup Fees	Setup, installation		\$300.00
Monthly Recurring Charges	Broadband Monthly Recurring Charge		\$150.00

Note: The above prices are subject to a 10% VAT



2-Offre d'un nom de domaine dynamique proposé par "DynDNS.com"



Annexe III : Fiches techniques des composants utilisés (Quelques pages)

1-Régulateur 5 V LM7805

KEC SEMICONDUCTOR TECHNICAL DATA

KIA7805AP~KIA7824AP BIPOLAR LINEAR INTEGRATED CIRCUIT

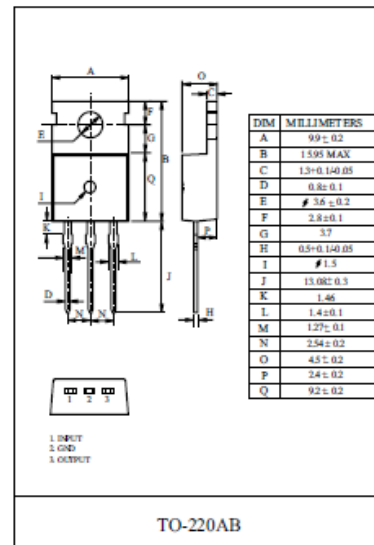
THREE TERMINAL POSITIVE VOLTAGE REGULATORS 5V, 6V, 7V, 8V, 9V, 10V, 12V, 15V, 18V, 20V, 24V.

FEATURES

- Internal Thermal Overload Protection.
- Internal Short Circuit Current Limiting.
- Output Current up to 1.5A.
- Satisfies IEC-65 Specification. (International Electronical Commission).
- Package is TO-220AB

LINE-UP

ITEM	OUTPUT VOLTAGE (Typ.)	UNIT
KIA7805AP	5	V
KIA7806AP	6	
KIA7807AP	7	
KIA7808AP	8	
KIA7809AP	9	
KIA7810AP	10	
KIA7812AP	12	
KIA7815AP	15	
KIA7818AP	18	
KIA7820AP	20	
KIA7824AP	24	



MAXIMUM RATINGS (Ta=25°C)

CHARACTERISTIC	SYMBOL	RATING	UNIT
Input Voltage	V _{IN}	35	V
		40	
Power Dissipation-1 (No Heatsink)	AP	P _{D2}	W
Power Dissipation-2 (Infinite Heatsink)	AP	P _{D2}	
Operating Junction Temperature	T _j	-40 ~ 150	°C
Storage Temperature	T _{stg}	-55 ~ 150	°C
Maximum Junction Temperature	T _{j(max)}	150	°C

2-Régulateur 3 V MC33268DT-3.3



Advance Information
**Low Dropout Positive
 Fixed and Adjustable
 Voltage Regulators**

The MC33269 series are low dropout, medium current, fixed and adjustable, positive voltage regulators specifically designed for use in low input voltage applications. These devices offer the circuit designer an economical solution for precision voltage regulation, while keeping power losses to a minimum.

The regulator consists of a 1.0 V dropout composite PNP-NPN pass transistor, current limiting, and thermal shutdown.

- 3.3 V, 5.0 V, 12 V and Adjustable Versions
- Space Saving DPAK, SOP-8 and SOT-223 Power Packages
- 1.0 V Dropout
- Output Current in Excess of 800 mA
- Thermal Protection
- Short Circuit Protection
- Output Trimmed to 1.0% Tolerance
- No Minimum Load Requirement for Fixed Voltage Output Devices

ORDERING INFORMATION

Device	Operating Temperature Range	Package
MC33200D	T _J = -40° to +125°C	SOP-8
MC33200DT		DPAK
MC33200ST		SOT-223
MC33200T		Insertion Mount
MC33200D-3.3		SOP-8
MC33200DT-3.3		DPAK
MC33200ST-3.3		SOT-223
MC33200T-3.3		Insertion Mount
MC33200D-5.0		SOP-8
MC33200DT-5.0		DPAK
MC33200T-5.0		Insertion Mount
MC33200D-12		SOP-8
MC33200DT-12		DPAK
MC33200T-12		Insertion Mount
MC33200ST-12		SOT-223

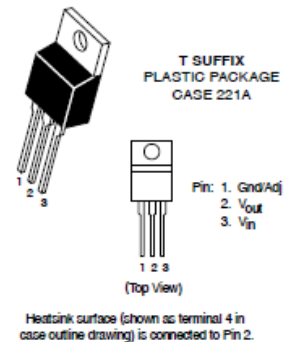
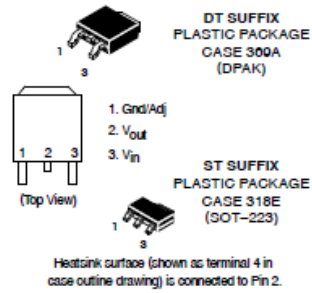
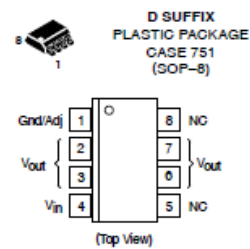
DEVICE TYPE/NOMINAL OUTPUT VOLTAGE

MC33200D	Adj	MC33200D-5.0	5.0 V
MC33200DT	Adj	MC33200DT-5.0	5.0 V
MC33200T	Adj	MC33200T-5.0	5.0 V
MC33200ST	Adj	MC33200ST-5.0	5.0 V
MC33200D-3.3	3.3 V	MC33200D-12	12 V
MC33200DT-3.3	3.3 V	MC33200DT-12	12 V
MC33200T-3.3	3.3 V	MC33200T-12	12 V
MC33200ST-3.3	3.3 V	MC33200ST-12	12 V

Order this document by MC33269/D

MC33269

800 mA
 LOW DROPOUT
 THREE-TERMINAL
 VOLTAGE REGULATORS



This document contains information on a new product. Specifications and information herein are subject to change without notice.

© Motorola, Inc. 1997

Rev 4

3-Circuit ULN2803A

- 500-mA Rated Collector Current (Single Output)
- High-Voltage Outputs . . . 50 V
- Output Clamp Diodes
- Inputs Compatible With Various Types of Logic
- Relay Driver Applications
- Compatible with ULN2800A Series

description/ordering information

The ULN2803A is a high-voltage, high-current Darlington transistor array. The device consists of eight npn Darlington pairs that feature high-voltage outputs with common-cathode clamp diodes for switching inductive loads. The collector-current rating of each Darlington pair is 500 mA. The Darlington pairs may be connected in parallel for higher current capability.

Applications include relay drivers, hammer drivers, lamp drivers, display drivers (LED and gas discharge), line drivers, and logic buffers. The ULN2803A has a 2.7-kΩ series base resistor for each Darlington pair for operation directly with TTL or 5-V CMOS devices.

ULN2803A
DARLINGTON TRANSISTOR ARRAY

SLRS049E – FEBRUARY 1997 – REVISED JULY 2006

DW OR N PACKAGE
(TOP VIEW)

1B	1	10	1C
2B	2	17	2C
3B	3	16	3C
4B	4	15	4C
5B	5	14	5C
6B	6	13	6C
7B	7	12	7C
8B	8	11	8C
GND	9	10	COM

ORDERING INFORMATION

T _A	PACKAGE†	ORDERABLE PART NUMBER	TOP-SIDE MARKING
-40°C to 85°C	PDIP (N)	Tube of 20	ULN2803AN
	SOIC (DW)	Tube of 40	ULN2803ADW
		Reel of 2000	ULN2803ADWRL

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

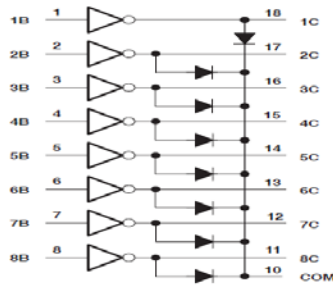
TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 2006, Texas Instruments Incorporated

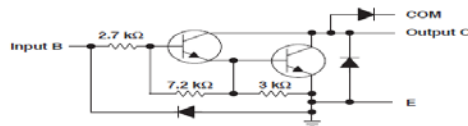
**ULN2803A
DARLINGTON TRANSISTOR ARRAY**

SLRS049E – FEBRUARY 1997 – REVISED JULY 2006

logic diagram



schematic (each Darlington pair)



2

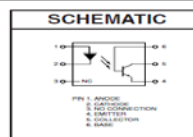


3- Opto-coupleur 4N35



**GENERAL PURPOSE 6-PIN
PHOTOTRANSISTOR OPTOCOUPLEDERS**

4N25 4N37	4N26 H11A1	4N27 H11A2	4N28 H11A3	4N35 H11A4	4N36 H11A5
--------------	---------------	---------------	---------------	---------------	---------------



DESCRIPTION

The general purpose optocouplers consist of a gallium arsenide infrared emitting diode driving a silicon phototransistor in a 6-pin dual in-line package.

FEATURES

- Also available in white package by specifying -M suffix, eg. 4N25-M
- UL recognized (File # E90700)
- VDE recognized (File # 94766)
 - Add option V for white package (e.g., 4N25V-M)
 - Add option 300 for black package (e.g., 4N25.300)

APPLICATIONS

- Power supply regulators
- Digital logic inputs
- Microprocessor inputs


**GENERAL PURPOSE 6-PIN
PHOTOTRANSISTOR OPTOCOUPLEDERS**

4N25 4N37	4N26 H11A1	4N27 H11A2	4N28 H11A3	4N35 H11A4	4N36 H11A5
ABSOLUTE MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise specified)					
Parameter	Symbol	Value	Units		
TOTAL DEVICE					
Storage Temperature	T_{STG}	-55 to +150	°C		
Operating Temperature	T_{OPR}	-55 to +100	°C		
Lead Solder Temperature	T_{SOL}	260 for 10 sec	°C		
Total Device Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	250	mW		
		3.3 (non-M), 2.94 (-M)			
EMITTER					
DC/Average Forward Input Current	I_F	100 (non-M), 60 (-M)	mA		
Reverse Input Voltage	V_R	6	V		
Forward Current - Peak (300 μs , 2% Duty Cycle)	$I_F(\text{PK})$	3	A		
LED Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	150 (non-M), 120 (-M)	mW		
		2.0 (non-M), 1.41 (-M)	mW/°C		
DETECTOR					
Collector-Emitter Voltage	V_{CEO}	30	V		
Collector-Base Voltage	V_{CBO}	70	V		
Emitter-Collector Voltage	V_{ECO}	7	V		
Detector Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	150	mW		
		2.0 (non-M), 1.76 (-M)	mW/°C		

4- Interface réseau (connecteur Ethernet RJ45)

PULSEJACK™ 1x1 Tab-DOWN RJ45



Description:

10/100 Base-TX RJ45 1x1 Tab-DOWN with LEDs 8-pin (J0 series) integrated magnetics connector (ICM), designed to support applications, such as SOHO (ADSL modems), LAN-on-Motherboard (LOM), Hub and Switches.



J0006D21BNL shown

Features and Benefits:

- RoHS peak wave solder temperature rating 260°C
- Suitable for CAT 5 & 6 Fast Ethernet Cable or better UTP

Electrical Performance Summary:

- Meets or exceeds IEEE 802.3 standard for 100Base-T
- 350 µH minimum OCL with 8mA bias current
- Minimum 1500Vrms isolation per IEEE 802.3 requirement

Electrical Specifications @ 25°C — Operating Temperature 0°C to +70°C

RoHS Compliant Part Number	Turns Ratios		Mech. Drawing	EMI Fingers	LEDs ¹ (L/R)	Insertion Loss (dB TYP)	Return Loss (dB TYP) 100W ±15W					Crosstalk (dB TYP)			Common Mode Rejection (dB TYP)		Hipot (VDC) 1 MIN
	TX	RX					1-65 MHz	1-10 MHz	10-30 MHz	30-60 MHz	60-80 MHz	1-30 MHz	30-60 MHz	60-100 MHz	1-50 MHz	50-150 MHz	
J0006D01BNL	1CT:1	1CT:1	A	No	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0006D21NL	1CT:1	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0006D21BNL	1CT:1	1CT:1	B	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0011D01NL	1CT:1	1CT:1	C	No	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0011D01BNL	1CT:1	1CT:1	A	No	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0011D21NL	1CT:1	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0011D21BNL	1CT:1	1CT:1	B	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0011D21ENL	1CT:1	1CT:1	B	YES	G/G	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0012D21NL	1CT:1	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0018D21NL	1CT:1.414	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0018D21ENL	1CT:1.414	1CT:1	B	YES	G/G	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0024D21NL	1CT:2	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0024D21BNL	1CT:2	1CT:1	B	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D01NL	1CT:1	1CT:1	D	No	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D01BNL	1CT:1	1CT:1	A	No	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D01ENL	1CT:1	1CT:1	A	No	G/G	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D01FNL ²	1CT:1	1CT:1	A	No	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D21NL	1CT:1	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D21BNL	1CT:1	1CT:1	B	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D21ENL	1CT:1	1CT:1	B	YES	G/G	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D21FNL ²	1CT:1	1CT:1	B	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0026D21GNL ³	1CT:1	1CT:1	B	YES	YG/G	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0033D21NL	1.25CT:1	1CT:1	E	YES	N/A	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0035D21BNL	1CT:1	1CT:1	F	YES	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	
J0073D01BNL	1CT:2.5	1CT:1	A	No	GY	-1	-20	-16	-12	-10	-40	-35	-30	-30	-20	2250	

Notes: 1. LEDs Left/Right: G=green, Y=yellow, N/A=none, YG=Bi-color LED Yellow Green. 2. LEDs with internal resistor. 3. Bi- color Left LED.

RJ45 Durability Testing Rating

Part Number	Mating Force (MAX)	Unmating Force (MAX)	Durability	Plug to Jack Retention (MIN)
J0 Series	5 lbs./2.268 kgs.	5 lbs./2.268 kgs.	750 Insertions	20 lbs./9.072 kgs.

RJ45 Material Specification

Part Number	Shield		Material	Contact		Solder Area ²	Housing		MSL ³ Rating
	Material	Finish		Plating Area	Material		Specification		
J0 Series	Brass	10-20m inches Nickel over 10-20m inches Brass	Phosphor Bronze	Nickel underplating and selective gold plating 15µ inches	120µ inches Sn90/Pb10 over 50µ inches nickel	Thermoplastic	UL 94 V-0	1	

Notes: 1. Connector dimensions comply with FCC dimension requirements. 3. MSL = Moisture Sensitivity Level rating from 1 to 5 (highest = 1, lowest = 5)
2. RoHS parts are tin matte finish over nickel.

USA 858 674 8100 • Germany 49 7032 7806 0 • Singapore 65 6287 8998 • Shanghai 86 21 62787060 • China 86 755 33966678 • Taiwan 886 3 4643715

www.pulseeng.com

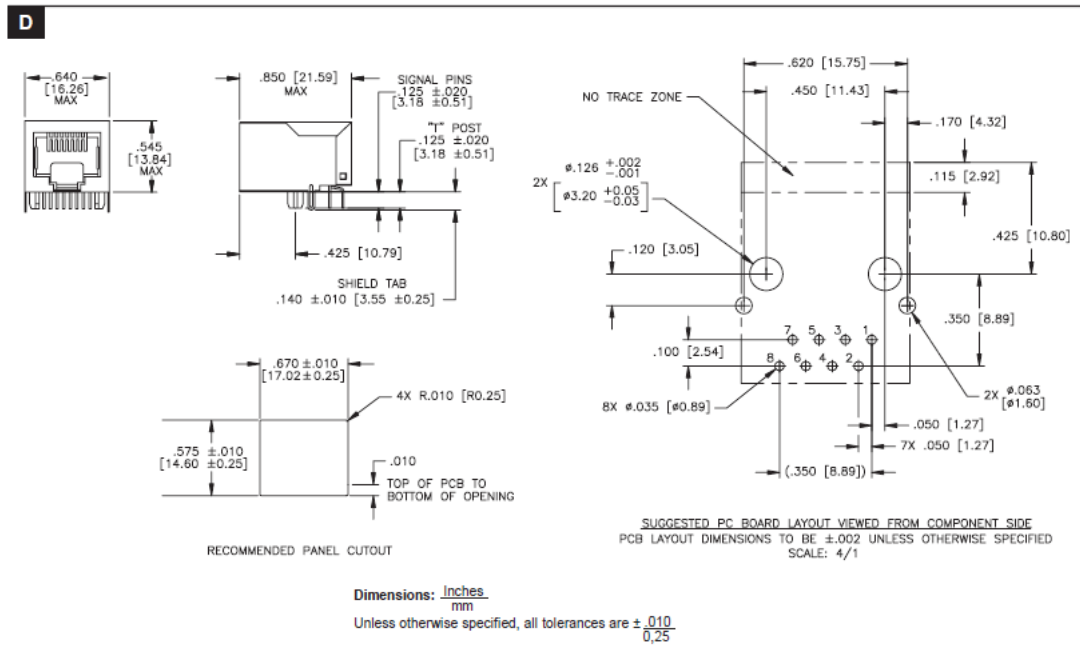
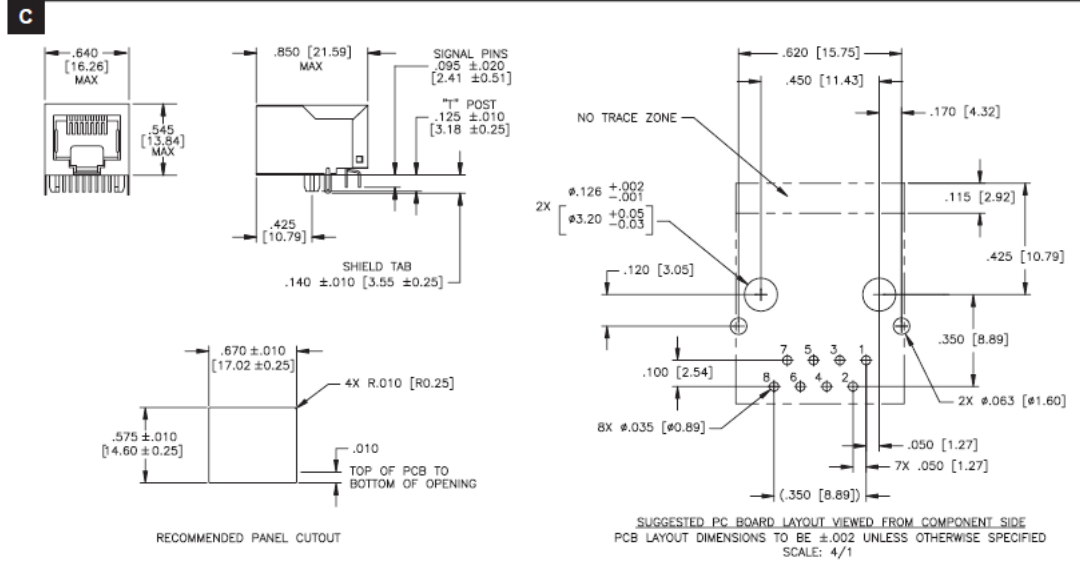
1

J403.I (8/10)

PULSEJACK™
1x1 Tab-DOWN RJ45



J0 Series Mechanicals



USA 858 674 8100 • Germany 49 7032 7806 0 • Singapore 65 6287 8998 • Shanghai 86 21 62787060 • China 86 755 33966678 • Taiwan 886 3 4643715
www.pulseeng.com 3 J403.1 (8/10)

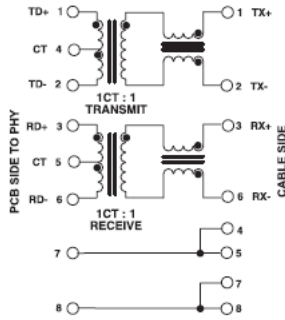
PULSEJACK™
1x1 Tab-DOWN RJ45



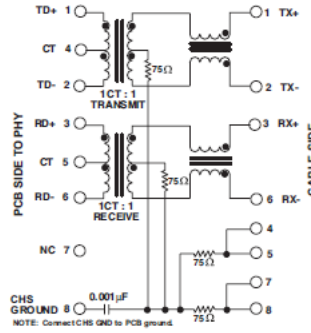
J0 Series Electrical Schematics

www.pulseeng.com

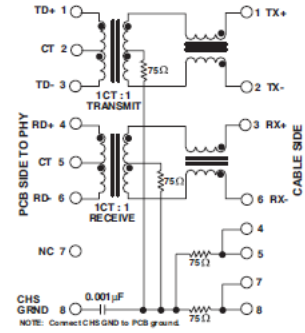
J0006



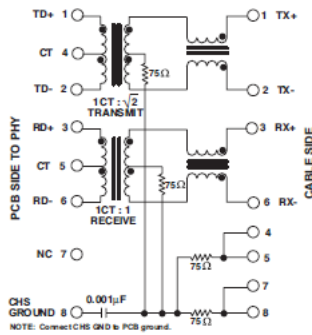
J0011



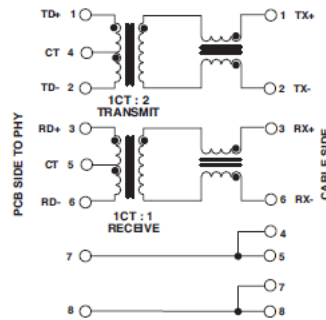
J0012



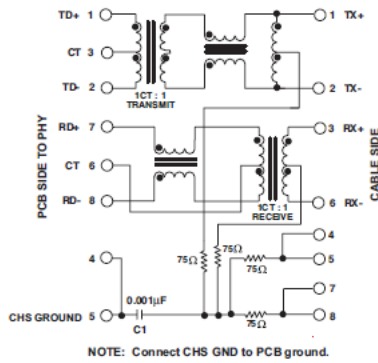
J0018



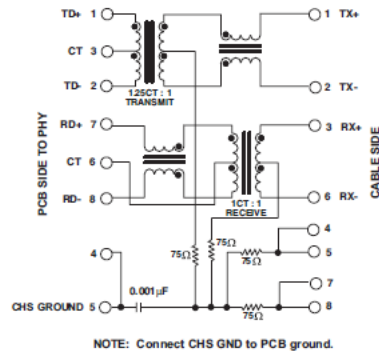
J0024



J0026



J0033



USA 858 674 8100 • Germany 49 7032 7806 0 • Singapore 65 6287 8998 • Shanghai 86 21 62787060 • China 86 755 33966678 • Taiwan 886 3 4643715

www.pulseeng.com

5

J403.I (8/10)

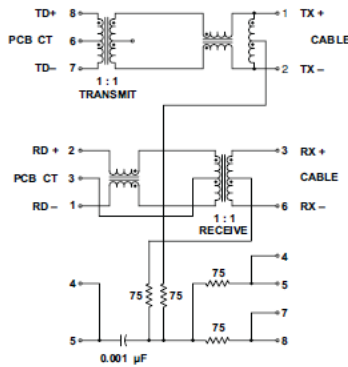
PULSEJACK™
1x1 Tab-DOWN RJ45



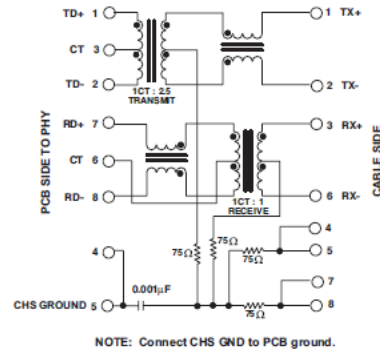
www.pulseeng.com

J0 Series Electrical Schematics (continued)

J0035

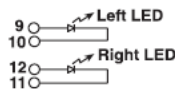


J0073

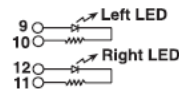


LED Configuration

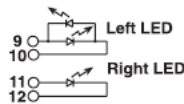
LED = "B & E" Specification
(LEDs without internal resistors)



LED = "F" Specification
(LEDs with internal resistors)



LED = "G" Specification
(Bi-color LED option with internal resistors)



Standard LED	Wavelength	Forward* V(MAX)	(TYP)
Yellow	585 nm	2.5 V	2.1 V
Green	565 nm	2.5 V	2.2 V

* Using an internal resistor within the LED increases the voltage rating of the diode from 2.5 V to 5.0 V (assumes bias current = 20 mA).

For More Information:

Pulse Worldwide Headquarters
12220 World Trade Dr.
San Diego, CA 92128
U.S.A.

Pulse Europe
Einsteinstrasse 1
D-71083 Herrenberg
Germany

Pulse China Headquarters
B402, Shenzhen Academy of
Aerospace Technology Bldg.
10th Kejinan Rd.
High-Tech Zone
Nanshan District
Shenzhen, PR China 518057

Pulse North China
Room 2704/2705
Super Ocean Finance Ctr.
2067 Yan An Rd. West.
Shanghai 200336
China

Pulse South Asia
135 Joo Seng Rd.
#03-02
PM Industrial Bldg.
Singapore 368363

Pulse North Asia
No. 26, Kao Ching Rd.
Yang Mei Chen
Taoyuan Hsien
Taiwan
R. O. C.

www.pulseeng.com

Tel: 858 674 8100
Fax: 858 674 8262

Tel: 49 7032 7806 0
Fax: 49 7032 7806 135

TEL: 86 755 33966678
FAX: 86 755 33966700

Tel: 86 21 62787060
Fax: 86 2162786973

Tel: 65 6287 8998
Fax: 65 6280 0080

Tel: 886 3 4643715
Fax: 886 3 4641911

Performance warranty of products offered on this data sheet is limited to the parameters specified. Data is subject to change without notice. Other brand and product names mentioned herein may be trademarks or registered trademarks of their respective owners.
© Copyright, 2010, Pulse Engineering, Inc. All rights reserved.

www.pulseeng.com

J403.I (8/10)

Bibliographie

- [1] <http://fr.wikipedia.org/wiki/Ascenseur>, Ascenseur :consulté le 01/10/2012.
- [2] ST-Autinor : Site de présentation de l'entreprise,2012, <http://www.autinor.com/>.
- [3] MAYEUX P., Cours sur le PIC, 2012, <http://p.may.chez-alice.fr> .
- [4] ST-Microchip : Site de présentation de l'entreprise , 2012 .
<http://www.microchip.com/>
- [5] ST-Microchip : Fiche technique concernant la famille de PIC 18FXXJ60, 2007.
- [6] ST-MikroElektronika : " PIC Microcontrollers-Programming in C" ,Ouvrage concernant les microcontrôleur PIC et son langage de programmation ,2011.
- [7] AXELSON J.,2003.*Embedded Ethernet and internet complete*, Lakeview Research LLC, Madison , 497 p.
- [8] ROS D., 2008. Protocole de transport TCP. *Techniques de l'Ingénieur*. **TE 7570**,1-15.
- [9] BENTHAM J., 2002. *TCP/IP LEAN Web Servers for Embedded Systems*, CMP Books ,535 p.
- [10] HALSHALL F., 2001. *Multimedia communications* .Addison-wsley , CWS, 1034 p.
- [11] DUFFY S.,2003. *How to do Everything with JavaScript* .McGraw-Hill/Osborne, 369 p.
- [12] NIXON R.,2009. *Learning PHP, MySQL, and JavaScript*.Mukherji S., O'REILLY ,528 p.
- [13] ST-MikroElektronika : Fiche technique concernant la carte de développement LV18F.

Liste des figures

Figure 1.1 : Types de familles d'ascenseurs	9
Figure 1.2 : Ascenseurs à traction à câbles	10
Figure 1.3 : Principe du mécanisme d'un ascenseur hydraulique.....	11
Figure 1.4 : Diversité géographique des clients de la société Autinor	12
Figure 1.5 : Bâtiments mondiaux utilisant l'ascenseur Autinor	12
Figure 1.6 : Schéma bloc du projet	14
Figure 2.1 : Schéma block de l'architecture Harvard	18
Figure 2.2 : Différentes familles de PIC	18
Figure 2.3 : Schéma bloc du PIC 18F87j60	20
Figure 2.4 : Schéma bloc du module ETHERNET	21
Figure 2.5 : Composants externes nécessaires pour le fonctionnement du module ETHERNET.....	23
Figure 2.6 : Relation entre les registres SFRs et l'espace mémoire du module Ethernet.....	24
Figure 2.7 : Compilation d'un programme en langage C	25
Figure 2.8 : Compilateur Mikroc PRO	26
Figure 2.9 : Outils du compilateur Mikroc PRO	27
Figure 2.10 : Cartes de développement comportant le module web	28
Figure 3.1: Accès sur internet via PIC MCU.....	30
Figure 3.2 : Couches TCP et UDP	32
Figure 3.3 : Encapsulation des données avec TCP	33
Figure 3.4 : Modèle en couches protocolaires de l'Internet	33
Figure 3.5 : Interconnexion des équipements et connexions TCP	34
Figure 3.6 : Couche IP avec UDP ou TCP	35
Figure 3.7 : Création d'un nouvel hôte sur le site DynDNS.com	39
Figure 3.8 : Création de l'hôte« zara »	40
Figure 3.9 : Configuration du DDNS sur le routeur	41
Figure 3.10 : Configuration de l'hôte sur le DDNS du routeur	41
Figure 3.11 : Résultat de la Configuration sur le DDNS du routeur	42
Figure 3.12 : Nouvelle application	42
Figure 3.13 : choix du port de l'application créé	43
Figure 3.14 : Union de l'adresse IP avec l'application créée	43
Figure 3.15 : L'application avec l'adresse IP de la carte utilisée	44
Figure 4.1 : Organigramme du programme d'interfaçage	51
Figure 4.2 : Relations entre les différentes tables de la base de données	53
Figure 4.3 : Interface de connexion d'utilisateur	54
Figure 4.4 : Interface de gestion des ascenseurs	55
Figure 4.5 : Interface de contrôle et de surveillance de l'ascenseur	56
Figure 4.6 : Interface d'ajout et de mise à jour des informations de l'ascenseur dans la base de données..	57
Figure 4.7 : Liste historique des erreurs de l'ascenseur	59
Figure 4.8 : Interface des informations des utilisateurs dans la bade de données	60
Figure 5.1 : Schéma électrique de la carte réalisée	63

Figure 5.2 : Schéma du redresseur double alternance et du régulateur de tension 5 V	64
Figure 5.3 :Schéma du régulateur de tension 3,3V	64
Figure 5.4 :Schématique d'alimentation de la carte	65
Figure 5.5 :Schéma électrique du reset	66
Figure 5.6 :Schéma électrique du circuit MCLR du PIC	66
Figure 5.7 :Schéma électrique de l'indicateur d'erreur	67
Figure 5.8 :Connecteur Ethernet RJ45	67
Figure 5.9 :Schéma du module Ethernet	68
Figure 5.10 :Carte BG-15	68
Figure 5.11 :Carte BG-17	69
Figure 5.12 : Schéma électriques des boutons de la carte BG-1	69
Figure 5.13 : Schéma de connexion des 2 afficheurs 7 segments de la carte BG-17	70
Figure 5.14 : Organigramme du programme principal (main)	73
Figure 5.15 : Organigramme de la fonction de transmission sur l'internet	75

Liste des Tableaux

Tableau IV.I :Sélection des directives ou balises du format HTML	47
Tableau IV.II: Liste des tables utilisées dans la base donnée	52
Tableau IV.III: Liste des codes PHP utilisés.....	53
Tableau V.I: Liste des entrées et sortis du PIC	71

Résumé

De nos jours le marché des systèmes embarqués suffisamment puissants pour permettre l'exécution de services web est en pleine expansion. Ces dispositifs permettent, avec des coûts de production peu élevés, de faciliter les tâches de surveillance et de contrôle à distance. Dans ce contexte, le cahier de charge de ce projet consiste à réaliser un système embarqué d'alerte et de diagnostic préliminaire à distance d'un ascenseur, à travers une interface web sécurisé. L'alerte consiste à envoyé un e-mail à un système de surveillance centralisé, indiquant l'ascenseur en panne et le numéro de l'erreur. Le personnel chargé du service pourra alors faire un diagnostic préliminaire avant l'envoi du personnel au site. Le diagnostic préliminaire peut aussi se faire à partir de n'importe quel endroit possédant une connexion internet. Ceci aide à limiter les temps « hors service » de l'ascenseur et à faciliter, par l'analyse des données reçues, la prise des bonnes décisions. Le client de notre projet est la société « Myzer Group ». Notre premier module réalisé sera installé sur un de ses ascenseurs de la française Autinor, chez son client "Le Mall (Zara – Sin El Fil). ”

Mots-clés : Microcontrôleur PIC18F87J60, serveur web embarqué, protocole TCP/IP, Dynamic DNS, ascenseur, interface Web sécurisée.

Abstract

Nowadays the market of embedded systems with enough powerful to allow the execution of web services is in full expansion. These devices help, with low production costs, to facilitate the tasks of remote monitoring and control. In this context, the goal of this project is to make an embedded system for alert and preliminary remote diagnosis of an elevator, through a secure web interface. The alert is an e-mail sent to a centralized monitoring system, indicating the elevator down and the error number. The service staff can then make a preliminary diagnosis before sending the personnel to the site. Preliminary diagnosis can also be done from anywhere with an internet connection. This helps to limit the “down time” of the elevator and facilitate, through the analysis of the data received, the making of the right decisions. The client of our project is the company "Myzer Group." Our first completed module will be installed on one of its' lifts of the French brand AUTINOR, at his client site "Le Mall (Zara - Sin El Fil).”

Keywords: PIC18F87j60 microcontroller, embedded web server, TCP/IP protocol, Dynamic DNS, elevator, secure web interface.