Table des matières

Introduction générales 1

Chapitre I : notion sur les microcontrôleurs

I. Introduction.....2

II. Les microcontrôleurs.....2

1. Généralité 2
2. Les avantages des microcontrôleurs 2
3. La composition d'un microcontrôleur 3

III. Les PICs.....4

1. Structure d'un PIC 4
2. Structure minimale d'un PIC 4
3. Identification d'un PIC 5
4. Le choix du microcontrôleur 6
5. Le microcontrôleur PIC 16F88..... 6

5.1. Caractéristiques générales……..……..6

5.2. Schéma fonctionnel…...……..……..7

5.3. brochage du pic 16F88..……..……....8

5.4. Organisation du 16F88……8

5.5. Les ports entrée/sortie……....11

5.5.1 Particularité du port A ……12

5.5.2 Particularité du port B ……12

5.6. Le convertisseur……...……..13

5.7. L'oscillateur……...……..…..13

6. Le module MSSP et le bus I2C..... 15

6.1. Le module MSSP……..……15

6.2. présentation de bus I²C…...……..……15

6.2.1 Start condition……..16

6.2.2 Transmission d'un bit ……16

6.2.3 Stop condition……17

6.2.4 Remarque sur le Start et le Stop condition……17

6.2.5 L'acknowledge ……17

6.2.6 L'adresse et le bit R/W ……17

6.3. Le module MSSP en mode I²C……..……18

6.3.1 Transmission d'un octet……..19

6.3.2 Réception d'un octet ……..……19

6.3.3 Les registres de configuration……..……19

6.3.3 Le module SSP du PIC 16F88 ……..……19

VI. CONCLUSION ……..……20

Chapitre II : mise en forme et acquisition des données

I. Introduction.....21

II. Système d'acquisition de données ……..……21

1. définition 21

1.1Une Chaine d'acquisition comporte 21

1.2 Schéma bloc d'une chaine d'acquisition 21

1.3 Carte d'acquisition de donnée..... 22

1.4 Schéma bloc d'une carte d'acquisition 22

III. La conversion analogique numérique.....22

1. Identification de la fonction C.A.N 22

2. Symbole d'un C.A.N……..……..……22

3. Principe ……..……..……23

4. Echantillonnage…..……..……23

5. Bloqueur 24

5.1 Numérisation..... 25

5.1.1. Principe 25

5.1.2. Résolution 25

5.1.3. Réalisation 25

IV. Le filtrage analogique 26
1. Introduction 26
2. Etude du filtre en régime sinusoïdal 26
2.1 Filtre actif et filtre passif 27
2.2 Les principaux types de filtres (idéaux) 27
2.3 Filtres réels 29
3. Filtre anti-repliement 30
V. Filtrage numérique d'un signal lent avec un microcontrôleur 30
1. Introduction 30
2. Définitions générales 31
3. Filtre numérique à « moyenne glissante » 31
4. Réponse en fréquence d'un filtre numérique 33
VI. La gestion de la liaison RS232 34
1. communication série asynchrone a travers le port série RS232 34
2. Protocole de transmission 34
VII. conclusion 35

Chapitre III : Etude et réalisation de la carte d'acquisition 7 entrées analogiques

I. Introduction 36
1. Présentation 36
II. SHEMA ELECTRIQUE DE LA CARTE 36
III. PRINCIPE DE FONCTIONNEMENT 37
1. Les filtres anti-repliements 37
2. Le convertisseur analogique - numérique (ADC) 38
2.1 Caractéristiques principales du module ADC du PIC 16F88 38
2.2 Mise en œuvre 38

2.2.1. Choix des canaux d'entrées 38
2.2.2. Choix des tensions de référence 40
2.2.3. Choix du format du résultat de la conversion 41
2.2.4. Choix de la fréquence d'horloge du convertisseur ADC..... 42
2.2.5. Mise en service de l'interruption du convertisseur ADC 43
2.3 Remarques et conseils..... 47
2.3.1. Calcul de la période d'échantillonnage (TE)..... 47
2.3.2. Echantillonnage de signaux lents..... 47
2.3.3. Fréquence d'échantillonnage et nombre de canaux..... 47
3. Le convertisseur analogique - numérique (ADC) 47
IV. LA LIAISON RS232 50
1. Principe d'adaptation PIC- RS232 50
2. Protocole de communication entre l'ordinateur et la carte 52
V. L'APPLICATION *ADC108.exe* POUR WINDOWS.....53
1. Procédure d'utilisation 54
VI. LITE DE MATERIEL.....54
1. la carte d'alimentation 55
VII. SIMULATION AVEC ISIS-PROTEUS.....55
VIII. CONCLUSION.....55
Chapitre IV : Application de la carte d'acquisition 7 voies
I. Introduction.....57
II. Etude pratique.....57
III. Application.....59
IV. Problématique.....60
VI. Résultats de conversion.....62
V. Conclusion.....69
Conclusion générale 70
Références bibliographies.....71

Liste des figures

Figure I. 1 : Structure interne d'un microcontrôleur 3
Figure I. 2: PIC 16F88 6
Figure I. 3: Schéma fonctionnel de PIC 16F88 7
Figure I. 4: Brochage du 16F88 8
Figure I. 5: organisation de mémoire programme 8
Figure I. 6: structure d'adressage direct 9
Figure I. 7: Structure d'adressage indirect 10
Figure I. 8: structure de l'oscillateur interne 14
Figure I. 9: support physique de bus I2C 15
Figure I. 10: adresse I2C (1010100). 16
Figure I. 11: structure de la condition start. 16
Figure I.12: Transmission d'un bit de donnée 16
Figure I.13: structure de la condition stop 17
Figure I.14: le bit acknowledge 17
Figure I.15: le bit not acknowledge 17
Figure I.16: l'adresse et le bit R/W 18
Figure I.17: Le module MSSP en mode I²C 18
Figure II. 1: Structure d'une chaine d'acquisition 21
Figure II. 2: Structure d'une carte d'acquisition 22
Figure II. 3: Symbole de la fonction Conversion Analogique / Numérique. 22
Figure II. 4: Principe du chaîne de conversion analogique-numérique. 23
Figure II. 5: exemple de signal échantillonné 24
Figure II. 6: structure de fonction de filtre 26
Figure II. 7: 2a (filtre passif) et 2b (filtre actif) 27
Figure II. 8: structure de filtre passe bas 27
Figure II. 9: structure de filtre passe haut 28
Figure II. 10: structure de filtre passe-bande 28
Figure II. 11: structure de filtre coupe-bande 29
Figure II. 12: structure de filtre réel 29
Figure II. 13: structure de diagramme de BODE de filtre réel 30
Figure II. 14: schéma fonctionnel d'une liaison série asynchrone de la norme RS232 .. 34
Figure II. 15: la liaison série croisée 34
Figure II. 16: Exemple de transmission série 35
Figure III. 1 : circuit de la carte sur plaque d'essai 36
Figure III. 2 : schéma électrique de la carte 37
Figure III. 3: diagramme de BODE de fréquence de coupure. 37
Figure III. 4: ADC du PIC 16F88 41
Figure III. 5 : échantillonneur-bloqueur de l'ADC 45
Figure III. 6 : Réponse en fréquence du filtre numérique la plage 0 à 250 Hz. 49
Figure III. 7 : Réponse en fréquence du filtre numérique la plage 0 à 640 Hz 49

Figure III. 8 : Réponse en fréquence du filtre numérique la plage 640 Hz à Fe / 2 = 1280 Hz 50
Figure III. 9 : Circuit intégré MAX232 51
Figure III. 10 : conversion TTL – RS232 51
Figure III. 11 : l'application *ADC108.exe* 53
Figure III. 12 : la réalisation de la carte sur plaque d'essai 55
Figure III. 13: schéma électrique de la carte d'alimentation 55
Figure IV. 1: Schéma de principe pour l`acquisition des signaux lents 57
Figure IV. 2: Schéma explicatif de l`application ADC108.exe 59
Figure IV. 3: La configuration des paramètres de la liaison RS232 60
Figure IV. 4: résultat de la conversion du signal sinusoïdale 1.093v du canal 0 62
Figure IV. 5: résultat de la conversion du signal triangulaire 4.678v du canal 0 63
Figure IV. 6: résultat de la conversion du signal rectangulaire 4.613v du canal 0 64
Figure IV. 7: résultat de la conversion de 2.246v d'entrée appliqué au canal1 65
Figure IV. 8: résultat de la conversion 3.500v d'entrée appliqué au canal 2 66
Figure IV. 9: résultat de la conversion 1.527v d'entrée appliqué au canal 3 67
Figure IV. 10: résultat de la conversion 3.350v d'entrée appliqué au canal 4 68
Figure IV. 11: résultat de la conversion 4.692v d'entrée appliqué au canal 6 69

Liste des tableaux

1.1. Caractéristiques du PIC 16F88 7
3.1. Choix des canaux.....39
3.2. Choix des tensions de référence 40
3.3. Choix de fréquence d'horloge d'ADC.....42
3.4. Fréquence maximum d'horloge du PIC 16F88 43
3.5. Sélection du canal à échantillonner 44

Rapport-Gratuit.com

Liste des abréviations

I²C: Inter – **I**ntegrated **C**ircuit

SPI: **S**erial **P**eripheral **I**nterface

MSSP: **M**aster **S**ynchronous **S**erial **P**ort

SSP: **S**ynchronous **S**erial **P**ort

EEPROM: **E**lectrical **E**rasable **P**rogrammable **R**ead **O**nly **M**emory

SDA: **S**erial **D**ata **l**ine

SCL: **S**erial **C**lock **l**ine

SAD: **S**ystème d'**A**cquisition de **D**onnées

VDD: broche d'alimentation positive du microcontrôleur pic 16f88

VSS : broche de masse (0V) du microcontrôleur

Introduction génerale

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Introduction générale :

Dans notre travail que nous proposons dans ce mémoire consiste à étudier et élaborer un montage électronique destine pour une usage médical.

La carte réalisé aura pour fonction de traiter des signaux lents (température, pression,…etc.) qui sont par la nature analogique, ce traitement est en fait une acquisition, dans le cas échéant, une amplification, une conversion A/N, et un traitement numérique des données.

Les cartes d`acquisitions sont conçues pour répondre à une ou plusieurs fonctions, elles peuvent se suffire à elle-même dans le cas où toutes les fonctions sont embarquées sur le même circuit ou être extensibles a l'aide des cartes additionnelles qui sont rajoutées à l'aide de bus.

Dans le domaine du traitement de l'information, les convertisseurs Analogique/ Numérique ont une grande utilité. Les tests de ces composants toujours plus performants deviennent de plus en plus sophistiqués. C'est pourquoi l'utilisation d'outils puissantes et conviviats, tel que l'informatique, est devenu nécessaire.

L'objectif de notre projet est la réalisation d'une nouvelle version de carte d'acquisition, plus utilisable, qui puisse fonctionner sur PC.

Dans notre projet de fin d'étude, nous proposons l'étude et réalisation d'une carte d'acquisition 7 voies analogiques pour des signaux lents les résultats de conversion sont affichés sur PC via la liaison RS 232

Dans le premier chapitre de notre mémoire, nous rappelons les notions principales des microcontrôleurs et les différents caractéristiques de pic 16F88 ainsi que le module SSP et le bus I^2C

Ensuite dans le deuxième chapitre nous développons la mise en forme et l'acquisition des données, la conversion analogique ainsi que la gestion de RS232

Et dans le chapitre 3 nous présentons la carte d'acquisition 7 voies analogiques, leurs schéma électrique, le principe de fonctionnement de la carte, le protocole de communication avec le PC et la simulation avec ISIS

Enfin dans le dernier chapitre nous citons les essais obtenues ainsi que les résultats de conversion sur l'application windows.

Chapitre I
notion sur les microcontrôleurs

I. Introduction

L'évolution des systèmes électroniques amène de plus en plus souvent les concepteurs à remplacer l'électronique câblée à base de nombreux circuits intégrés par un circuit programmable qui rempli à lui seul toutes les fonctions. Les microcontrôleurs appartiennent à cette famille de circuits.

Le microcontrôleur est implanté sur le système technique et relié aux différents capteurs et actionneurs avec éventuellement des circuits d'interfaçage appropriés. Une liaison avec un ordinateur permet de programmer le circuit et de le tester.

Le microcontrôleur est un véritable petit ordinateur dans un module hybride. Outre le processeur, le module dispose d'une horloge, d'une mémoire de travail (RAM), d'une mémoire pour le programme non volatile (EEPROM), de lignes d'entrées sorties binaires et analogiques et d'un port de communication avec un ordinateur (PC).

II. Les microcontrôleurs [1]

1. Généralité :

Un microcontrôleur est une unité de traitement de l'information de type microprocesseur contenant tous les composants d'un système informatique, à savoir microprocesseur, des mémoires et des périphériques (ports, timers, convertisseurs…). Chaque fabricant a sa ou ses familles de microcontrôleur. Une famille se caractérise par un noyau commun (le microprocesseur, le jeu d'instruction…).

2. Les avantages des microcontrôleurs :

L'utilisation des microcontrôleurs pour les circuits programmables à plusieurs points forts. Il suffit pour s'en persuader, d'examiner la spectaculaire évolution de l'offre des fabricants de circuits intégrés en ce domaine depuis quelques années. Nous allons voir que le nombre d'entre eux découle du simple sens.

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

- Cette intégration a aussi comme conséquence immédiate de simplifier le tracé du circuit imprimé puisqu'il n'est plus nécessaire de véhiculer des bus d'adresses et de donnée d'un composant à un autre.
- L'augmentation de la fiabilité du système puisque, le nombre des composants diminuant, le nombre des connexions composants/supports ou composants/circuits imprimer diminue.
- Le microcontrôleur contribue à réduire les coûts à plusieurs niveaux :
  - Moins cher que les autres composants qu'il remplace.
  - Diminuer les coûts de main d'œuvre.
- Réalisation des applications non réalisables avec d'autres composants.

3. *La composition d'un microcontrôleur :*

Un circuit microcontrôleur doit contenir dans un seul boîtier tous Les éléments de bases qu'on verra par la suite. En effet, pour l'analyse des divers systèmes réalisés avant l'avènement des microcontrôleurs, les fabricants des circuits intégrés a affinées un peu la définition de ce qu'il fallait intégrer pour arriver à un schéma type analogue à la figure suivante :

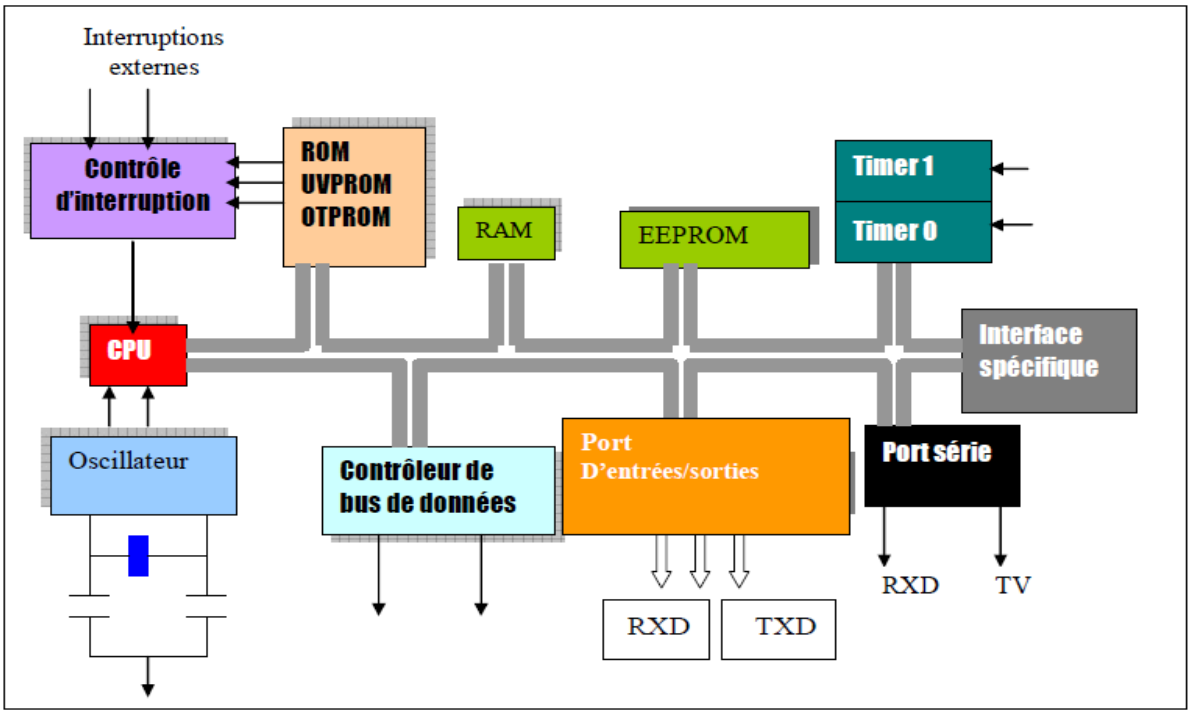


Figure1.1 : *Structure interne d'un microcontrôleur.* [1]

III. Les PICs [1]

1. *Structure d'un PIC :*

Les PIC, au même titre que les microprocesseurs, sont composés essentiellement de registres ayant chacun une fonction bien définie. Les PIC possèdent également des périphériques intégrés, tels qu'une mémoire EEPROM, un timer, des ports d'entrées/ sorties ou bien encore un convertisseur analogique/numérique.

Selon le type de PIC utilisé, on retrouvera en interne un certain nombre de registres et périphériques possédant des caractéristiques différentes. Les différences de caractéristiques selon le PIC utilisé sont :

- La taille de la RAM interne ;
- La mémoire EEPROM intégrée ;
- Le type de mémoire programme : FLASH, EPROM ou OTP et la taille de celle-ci.
- Le timer intégré ;
- Les convertisseurs analogique/numérique intégrés.

2. *Structure minimale d'un PIC :*

La structure minimale d'un PIC est constituée des éléments ci-dessous :

- Une mémoire de programme contient le code binaire correspondant aux instructions que doit exécuter le microcontrôleur. La capacité de cette mémoire est variable selon les PIC
- Une mémoire RAM sauvegarde temporairement des données. Sa capacité est aussi variable selon les PIC
- Une Unité Arithmétique et Logique (UAL ou ALU en anglais) est chargée d'effectuer toutes les opérations arithmétiques de base (addition, soustraction, etc.) ainsi que les opérations logiques de base (ET, OU logique, etc.).
- Des ports d'entrées/sorties permettent de dialoguer avec l'extérieur du microcontrôleur, par exemple pour prendre en compte l'état d'un interrupteur (entrée logique), ou encore pour commander un relais (sortie logique).
- Un registre compteur de programme (CP ou PC en anglais), est chargé de pointer l'adresse mémoire courante contenant l'instruction à réaliser par le microcontrôleur. Le contenu du registre PC évolue selon le pas de programme.

- Un registre pointeur de pile (PP ou SP en anglais) est essentiellement utilisé lorsque l'on réalise un sous-programme. Le pointeur de pile est chargé de mémoriser l'adresse courante que contient le compteur de programme avant le saut à l'adresse du sous-programme. Lorsque le sous-programme est terminé, le pointeur restitue l'adresse sauvegardée vers le compteur de programme.
- Un registre d'instruction contient tous les codes binaires correspondant aux instructions à réaliser par le microcontrôleur. Le PIC 16F88 comporte 35 instructions.
- Un registre d'état est en relation avec l'UAL et permet de tester le résultat de la dernière opération effectuée par le microcontrôleur. Selon la dernière opération effectuée, des bits sont positionnés dans le registre d'état et ceux-ci peuvent être testés à l'aide d'une instruction de branchement pour effectuer des sauts conditionnels.
- Une horloge système permet de cadencer tous les échanges internes ou externes au microcontrôleur.

La famille des PICs est subdivisée en 3 grandes familles : La famille ***Base-Line***, qui utilise des mots d'instructions de 12 bits, la famille ***Mid-Range***, qui utilise des mots de 14 bits, et la famille ***High-End***, qui utilise des mots de 16 bits.

3. Identification d'un PIC :

Pour identifier un PIC, nous utilisons simplement son numéro : les 2 premiers chiffres indiquent la catégorie du PIC, **16** indique un PIC Mid-Range, 18 indique PIC high-Range. Vient ensuite parfois une lettre **L** : Celle-ci indique que le PIC peut fonctionner avec une plage de tension beaucoup plus tolérante. Ensuite, nous trouvons : **C** indique que la mémoire programme est une EPROM ou plus rarement une EEPROM, **CR** pour indiquer une mémoire de type ROM ou **F** pour indiquer une mémoire de type *FLASH*. À ce niveau, on rappelle que seule une mémoire FLASH ou EEPROM est susceptible d'être effacée. Finalement nous trouvons sur les boîtiers le suffixe « -XX » dans lequel XX représente la fréquence d'horloge maximale que le PIC peut recevoir. Par exemple –04 pour un 4MHz. Notons dès à présent que les PICs sont des composants STATIQUES, c'est à dire que la fréquence d'horloge peut être abaissée jusqu'à l'arrêt complet sans perte de données et sans dysfonctionnement.

4. *Le choix du microcontrôleur :*

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

Le PIC 16F88 possède en plus des instructions très puissantes donc un programme à développer réduit, une programmation simple grâce au mode série. En fait la cause principale du choix du 16F88 est qu'il dispose de l'option du convertisseur A/D pour satisfaire côté acquisition.

5. *Le microcontrôleur PIC 16F88 :* [3]

Nous allons maintenant s'intéresser à la structure interne du PIC 16F88, avec lequel nous avons travaillé.

Le 16F88 est un microcontrôleur de MICROCHIP, fait partie intégrante de la famille des Mid-Range (16) dont la mémoire programme est de type flash (F) de type 88 et capable d'accepter une fréquence d'horloge maximale de 20Mhz.

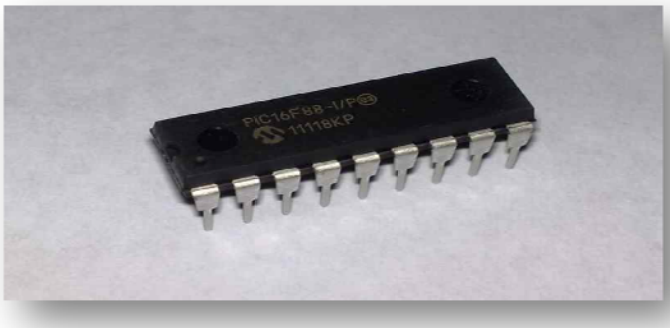


Figure 1.2 : PIC 16F88

5.1. *Caractéristiques générales:* [3]

caractéristiques	16F88
Broches	18
E/S max	16
µy flash	4 KO
µy E²PROM	256 O
CAN	7
PWM	1 DE 10 BITS
TIMER r	3
Comparateurs	2
Interruption	13
oscillateur	20 MHz MAX
Port série	USART/SSP

Tableau 1.1: *Caractéristiques du PIC 16F88.*

5.2. Schéma fonctionnel :

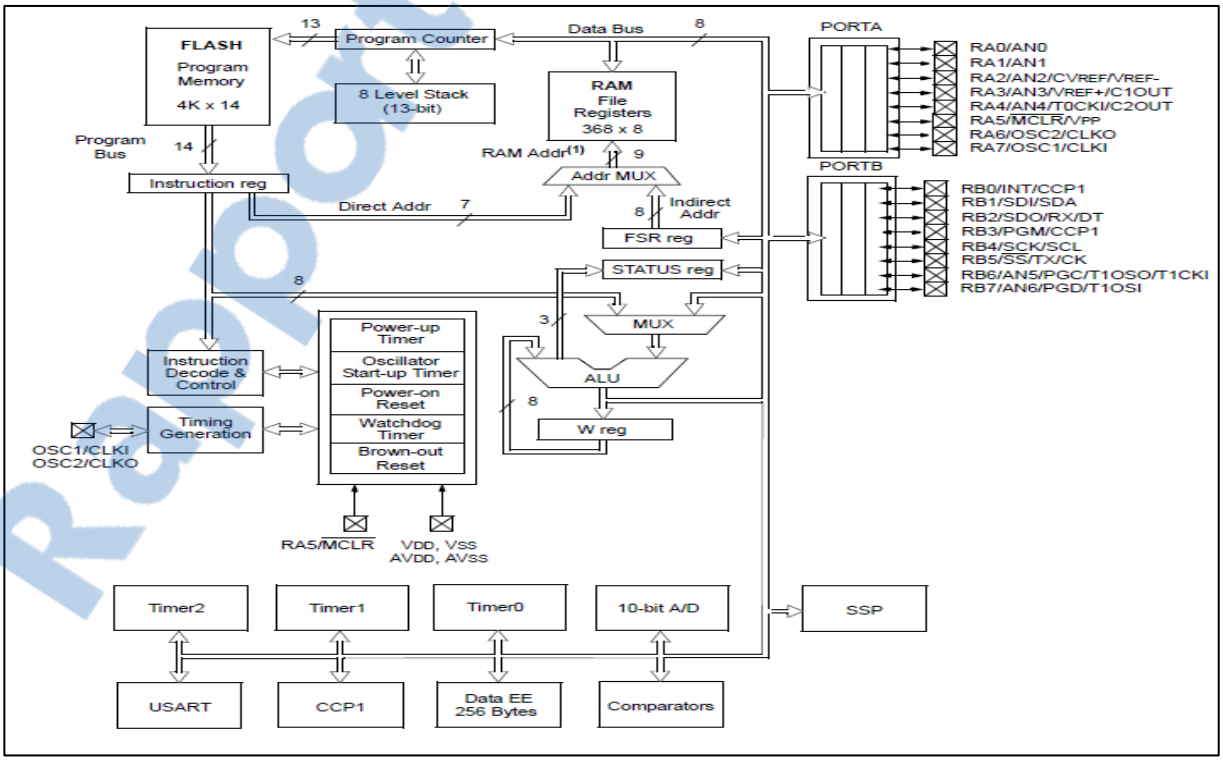


Figure 1.3 : Schéma fonctionnel de PIC 16F88 [14]

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

5.3. Brochage du PIC 16F88 :

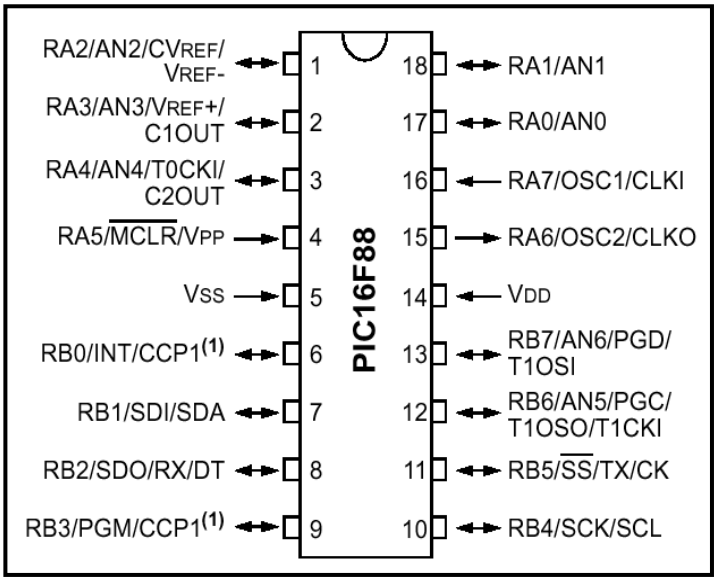


Figure 1.4 : Brochage du 16F88 [9]

5.4. *Organisation du 16F88* : [2]

La mémoire du 16F88 est divisée en 3 parties Pour mieux le connaître regardons de près son Schéma fonctionnel**.**

a- **La mémoire programme :**

La mémoire programme est constituée de 14 bits. C'est dans cette zone que nous allons écrire notre programme.

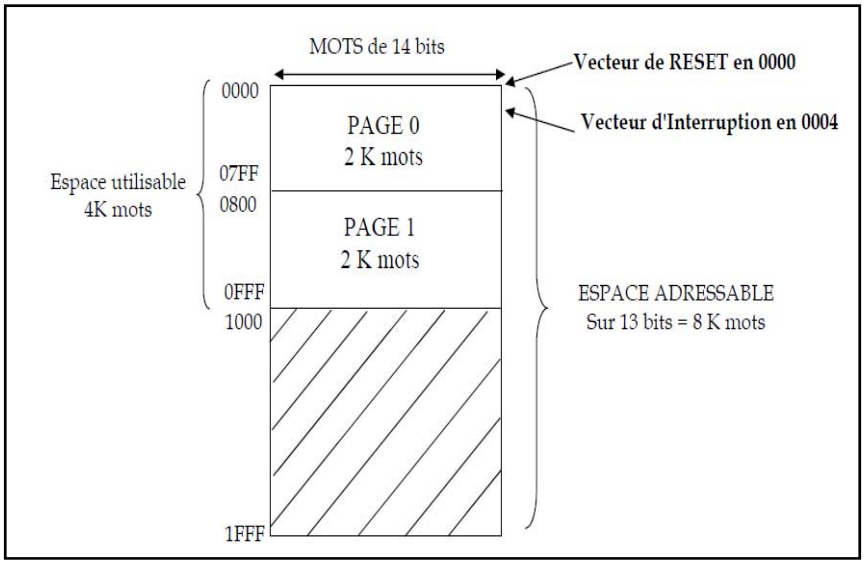


Figure 1.5 : organisation de mémoire programme

b- **La mémoire EEPROM :**

La mémoire EEPROM (**E**lectrical **E**rasable **P**rogrammable **R**ead **O**nly **M**emory), est constituée de 256 octets que nous pouvons lire et écrire depuis notre programme. Ces octets sont conservés après une coupure de courant et sont très utiles pour conserver des paramètres semi permanents.

c- **La mémoire Ram et organisation :**

La mémoire RAM est celle que nous allons sans cesse utiliser. Toutes les données qui y sont stockées sont perdues lors d'une coupure de courant. La mémoire RAM disponible du 16F88 est de 368 octets.

Pour accéder à la RAM, on dispose de deux modes d'adressage :

d- **Adressage DIRECT**

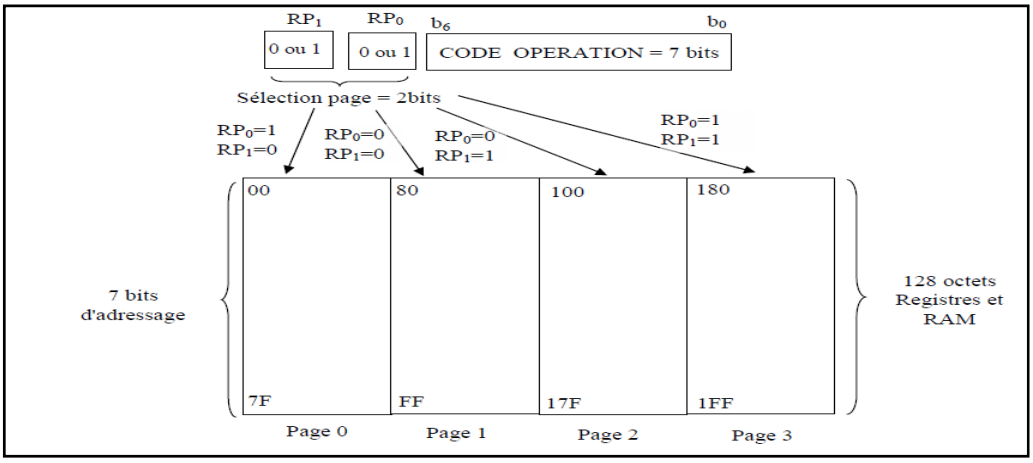


Figure 1.6 : structure d'adressage direct

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

e- **Adressage INDIRECT**

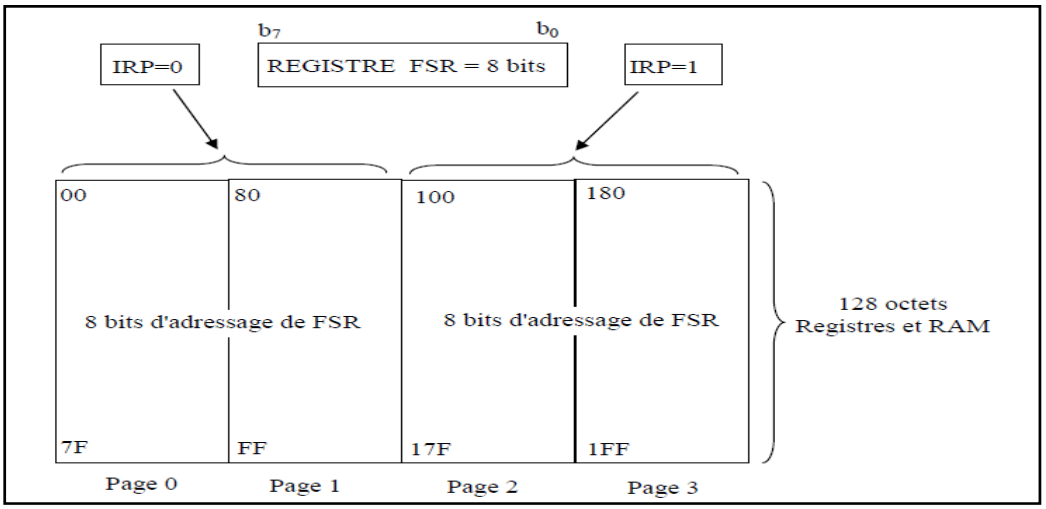


Figure 1.7 : Structure d'adressage indirect.

f- **Watchdog :**

Cette fonction est capable de surveiller le bon fonctionnement du programme que le micro contrôleur exécute. Le rôle du Watchdog (ou chien de garde) est de "reseter" le micro contrôleur si 1 'on ne remet pas à zéro périodiquement (à intervalle définissable) un registre interne grâce à 1 'instruction clrwdt (clear watchdog), si le programme tourne par exemple dans une boucle sans fin (qui est un bug dans le programme) la fonction de watchdog va permettre de remettre à 0 le micro contrôleur afin de relancer le programme.

g- **TIMER :**

Ils sont au nombre de 3. Un timer permet de compter le temps, ou des impulsions. Chaque timer possède des spécificités précises. La compréhension à 100% des timers est assez difficile. Heureusement, la plupart des compilateurs nous permettent de nous en servir de manière relativement transparente.

❖ **Timer 0**

Ce timer est le timer de base, et se configure via le registre OPTION_REG. Timer 8 bit, le timer 0 est un compteur pouvant être initialisé à une valeur donnée, via le registre TMR0 (charger la valeur désirée dans ce registre). Lorsque l'interruption du timer 0 est activée, le passage de 0xFF à 0x00 du timer provoque l'activation du flag du timer0.

❖ **Timer 1**

Second timer, il est sur 16 bits. Ce registre se configure via T1CON. Tout comme le timer 0, le timer 1 est initialisable en chargeant la valeur désirée dans les 2 registres dédiés: TMR1H pour l'octet supérieur, et TMR1L pour l'octet inférieur. A noter que l'entrée timer 1 (autre que T1CKI), est en fait une porte logique inverseuse.

❖ **Timer 2**

Troisième timer du PIC, et second timer 8 bits, le timer 2 est notamment utilisé pour générer la PWM. Il possède également, en plus d'un prediviseur, un postdiviseur. Il est, lui, initialisable via le registre TMR2. La période du timer 2 peut être configurée via PR2 (1 octet).

5.5. Les ports entrée/sortie :[3]

On dispose de 16 broches d'entrées/sorties, chacune configurables soit en entrée soit en sortie (PORTA, PORTB).

Dans cette partie, nous allons voir les différentes possibilités de chaque patte, en les désignant par leur numéro. A noter, qu'une seule fonction est disponible à la fois, par patte.

PATTE 1: E/S numérique RA2, CAN2, CVREF (référence comparateur), Vref- (référence basse CAN), entrée + du comparateur 2

PATTE 2: E/S numérique RA3, CAN3, VREF+ (référence haute CAN), sortie du comparateur 1, entrée + du comparateur 1

PATTE 3: E/S numérique RA4, CAN4, entrée de comptage timer 0, sortie du comparateur 2

PATTE 4: E numérique RA5, reset

PATTE 5: Masse

PATTE 6: E/S numérique RB0, interruption externe, sortie PWM

PATTE 7: E/S numérique RB1, entrée data SPI, data I²C

PATTE 8: E/S numérique RB2, sortie data SPI, entrée asynchrone, détection synchrone

PATTE 9: E/S numérique RB3, sortie PWM

PATTE 10: E/S numérique RB4, horloge SPI, horloge I²C

PATTE 11: E/S numérique RB5, sélection esclave SPI, sortie asynchrone, horloge synchrone

PATTE 12: E/S numérique RB6, CAN5, sortie oscillateur timer1, entrée horloge timer 1

PATTE 13: E/S numérique RB7, CAN6, entrée oscillateur externe timer 1

PATTE 14: Alimentation positive, 5V

PATTE 15: S numérique RA6, entrée quartz, sortie horloge

PATTE 16: E numérique RA7, entrée quartz, entrée horloge externe

PATTE 17: E/S numérique RA0, CAN0, entrée – du comparateur 1

PATTE 18: E/S numérique RA1, CAN1, entrée – du comparateur 2

Les E/S se configurent via le registre TRISA, TRISB.

5.5.1. Particularité du port A :

Ce port bidirectionnel est constitué de 8 bits. Le registre de direction correspondant est TRISA. Quand on écrit un "1" dans TRISA, le bit correspondant du PORTA est configuré en ENTREE, et le driver de sortie est placé en haute impédance. Si on écrit un "0", le port devient une SORTIE, et le contenu du latch correspondant est chargé sur la broche sélectionnée. Les bit 0 à 4 du Port peuvent servir d'I/O digitales ou bien être les 5 premières entrées du convertisseur analogique/numérique. C'est par le registre ANSEL que sera fait le choix. On étudiera ce registre avec le CAN. Le bit 5 peut être soit une entrée digitale soit la commande MCLR qui est le reset. C'est par le mot de configuration que le choix de cette broche sera fait. Les bits 6 et 7 peuvent être soit des I/O digitales soit les broches servant à relier le quartz de l'oscillateur. Quand on veut récupérer ces broches, il faudra faire fonctionner le PIC sur son oscillateur interne. C'est par le mot de configuration, comme pour PA5 que le choix de ces broches sera fait.

5.5.2. Particularités du port B :

Il comporte 8 bits. Le registre de direction correspondant est TRISB.

Si on écrit un "1" dans le registre TRISB, le driver de sortie correspondant passe en haute impédance. Si on écrit un "0", le contenu du Latch de sortie correspondant est recopié sur la broche de sortie. Chaque broche du PORT B est munie d'un tirage au +VDD que l'on peut mettre ou non en service en mode entrée uniquement. On active cette fonction par la mise à "0" du bit 7 dans le registre OPTION.

Au reset, le tirage est désactivé. Il est inactif quand le port est configuré en sortie. Les 4 broches PB7 PB6 PB5 et PB4 provoquent une interruption sur un changement d'état si elles sont configurées en ENTREE. Cette possibilité d'interruption sur un changement d'état associé à la fonction de tirage configurable sur ces 4 broches, permet l'interfaçage facile avec un clavier. Cela rend possible le réveil du PIC en mode SLEEP par un appui sur une touche du clavier On doit remettre à zéro le Flag de cette interruption (RBIF = bit 0 du registre INTCON) dans le programme d'interruption.

5.6. Le convertisseur : [3]

Le CAN est un périphérique intégré destiné à mesurer une tension et la convertir en nombre binaire qui pourra être utilisé par un programme.

Notre 16F88 travaille avec un convertisseur analogique/numérique qui permet un échantillonnage sur 10 bits. Le signal numérique peut donc prendre 1024 valeurs possibles.

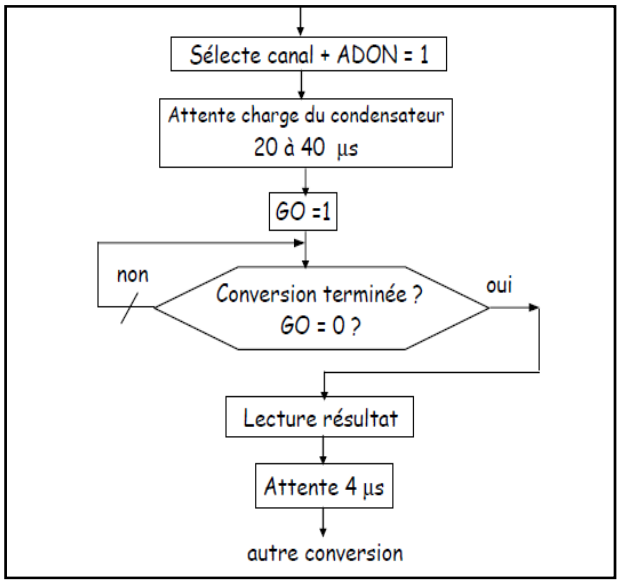
Les 5 premières entrées sont sur le Port A en RA0, RA1, RA2, RA3 et RA4, et le 2 dernières sur le Port B en RB6 et RB7.

Les tensions de référence haute et basse peuvent être choisies par programmation parmi: VDD ou la broche PA3 pour VREF + et VSS ou la broche PA2 pour VREF - .

Les 4 registres utilisés par le module convertisseur A/D sont :

- ADRESH: MSB des 10 bits du résultat.
- ADRESL: LSB des 10 bits du résultat.
- ADCON0: registre de contrôle n°0 du convertisseur.
- ADCON1: registre de contrôle n°1 du convertisseur.

Exemple d'un programme : [2]



5.7. L'oscillateur : [3]

L'horloge système peut être réalisée soit avec un quartz, soit avec une horloge extérieure, soit avec un circuit RC. Dans ce dernier cas, la stabilité du montage est limitée.

Notre PIC peut fonctionner sous les mêmes modes classiques que les 16F84 ou 16F877, c'est à dire:

- Mode LP : Avec des quartzs de fréquence basse (200 KHz max).
- Mode XT: Avec des quartzs de fréquence max 4 MHz.
- Mode HS: Avec des quartzs de fréquence max 20 MHz.
- Mode RC: Avec un condensateur et une résistance extérieure.

Il possède de nouveaux modes:

- Mode ECIO: Avec une horloge externe entrant sur PA6.
- Mode INTIO1 : Oscillateur interne. $Fosc/4$ sur PA6 et PA7 libre pour I/O.
- Mode INTIO2 : Oscillateur interne. PA6 et PA7 libres pour des I/O.

Le choix du mode sera fait par les 3 bits $Fosc0$ Fosc1 et $Fosc2$ du mot de configuration.

a- Le bloc oscillateur interne

Le 16F88 possède un bloc oscillateur qui produit un signal de 31,25 KHz et un signal de 8 MHz. Le premier sert à piloter le "watchdog". Le signal à 8 MHz qui passe à travers un pré diviseur, peut servir d'horloge système. L'oscillateur étant divisé par 4, on aura une horloge cycle maximum de 2 MHz soit un temps de cycle de 500 ns. Le choix du pré diviseur pour le signal de 8 MHz est fait par les 3 bits IRCF0 IRCF1 et IRCF2 du registre OSCCON.

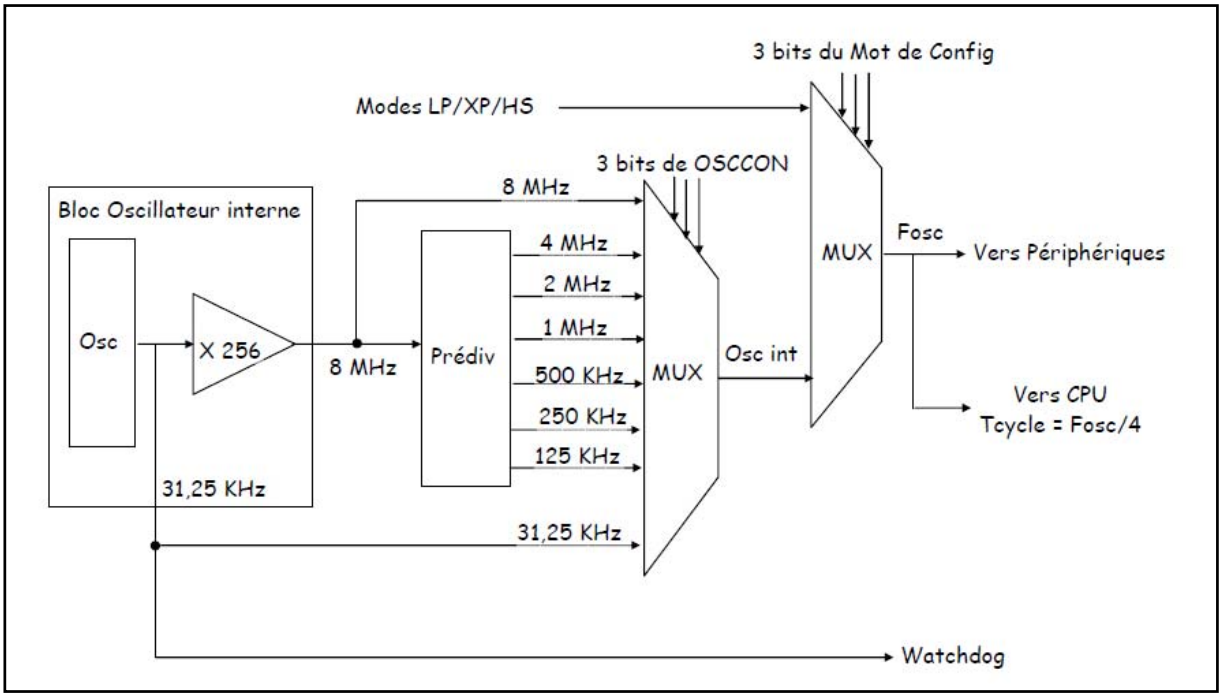


Figure 1.8 : structure de l'oscillateur interne [2]

6. *Le module MSSP et le bus I2C* [15]

6.1. le module MSSP :

Le MSSP est un des deux modules de communication série du PIC 16F88. Il permet d'échanger des données en mode synchrone avec d'autres circuits qui peuvent être des microcontrôleurs, des mémoires EEPROM série, des convertisseurs A/N, des modules d'affichage… Il peut fonctionner selon deux modes : le mode **SPI** (**S**erial **P**eripheral **I**nterface) et le mode **I²C** (**I**nter-**I**ntegrated **C**ircuit).

6.2. Présentation du bus I2C :

Le bus I²C (Inter-Integrated Circuit) est un bus populaire développé par la société Philips dans les années 1980.

Le bus I2C est un **bus série synchrone** bifilaire :

- **SDA** (Serial Data Line) : ligne de données bidirectionnelle
- **SCL** (Serial Clock Line) : horloge de synchronisation bidirectionnelle

N.B : Cela fait 3 fils en comptant la masse.

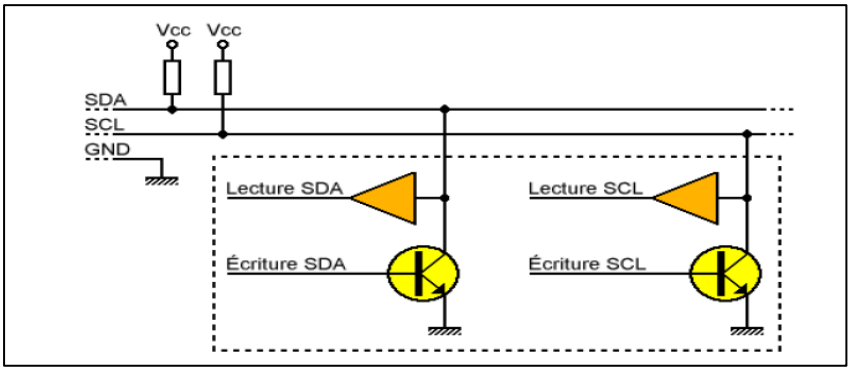


Figure 1.9 : support physique de bus I²C

Le bus I²C est contrôlé par un maître (c'est généralement un microcontrôleur, par exemple un PIC 16F88). Le bus I²C peut avoir plusieurs maîtres (on parle alors de mode multi-maîtres), mais à un instant donné, un seul maître contrôle le bus I²C. Le bus I²C possède un ou plusieurs esclaves (un bus I2C sans esclaves ne sert à rien). **Un esclave est identifié par une adresse unique** (sur 7 bits ou parfois 10 bits) configurer

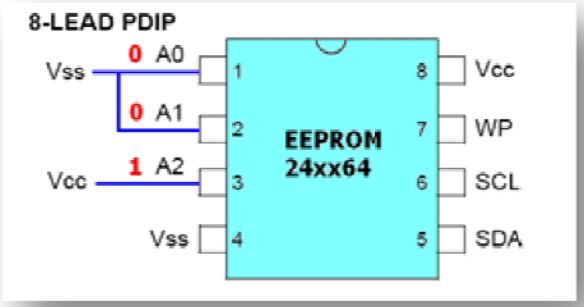


Figure 1.10 : adresse I²C (1010100)

6.2.1. Start condition : [15]

Au début d'une séquence d'échange, le master génère un start condition (S) avant de commencer l'échange de données. Au repos, les lignes SCL et SDA sont à l'état haut (relâchées). Pour générer un start, le master place d'abord la ligne SDA à 0, ensuite il place SCK à 0.

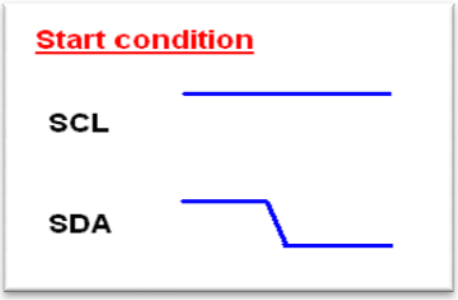


Figure 1.11 : structure de la condition start

6.2.2. Transmission d'un bit : [15]

On place le bit à transmettre sur la ligne SDA ensuite on envoie une impulsion d'horloge sur la ligne SCK. C'est cette impulsion qui informe le slave qu'il doit lire la donnée sur SDA

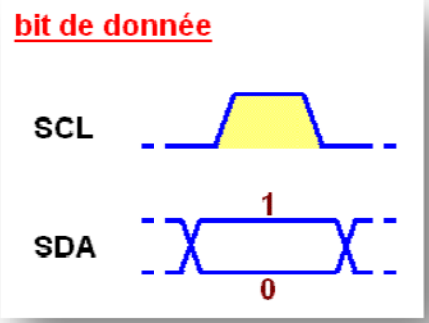


Figure 1.12 : Transmission d'un bit de donnée

6.2.3. Stop condition : [15]

A la fin d'une séquence d'échange, le master génère un stop condition (P) après lequel le bus est de nouveau libre. Pour cela, à partir de la situation SDA=0, SCK=0, le master commence par placer SCK à 1 et place ensuite SDA à 1.

Figure 1.13 : structure de la condition stop

6.2.4. Remarque sur le Start et le Stop condition :

Une séquence de transmission peut contenir plusieurs Starts conditions avant de rencontrer un Stop Condition. On parle de repeated Start condition. Un Stop condition est toujours synonyme de FIN de transmission.

6.2.5. L'acknowledge : [15]

L'acknowledge est l'accusé de réception. Il est placé par le circuit qui reçoit sur la ligne SDA juste après la réception du 8ème bit. C'est l'émetteur qui le lit de la même façon qu'on lit un bit ordinaire. SDA=0 → acknowledge positif (ACK), SDA=1 → acknowledge négatif (NOACK).

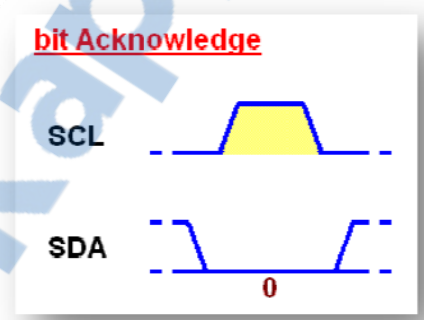


Figure 1.14: le bit acknowledge

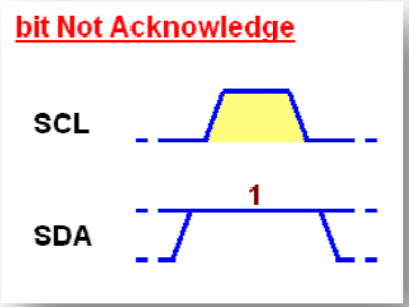


Figure 1.15: le bit not acknowledge

Rapport-Gratuit.com

6.2.6. L'adresse et le bit R/W :

Comme on peut brancher plusieurs composants sur un bus I2C, il est nécessaire de définir une adresse unique pour chacun. Elle est codée sur 7 bits A6 A5 A4 A3 A2 A1 A0. Le master qui démarre une séquence d'échange envoie l'adresse du slave juste après le *start condition*. Comme il y a seulement 7 bits, le master envoie à la 8ème position le bit R/W pour indiquer au slave s'il désire une émission (R/W=0) ou une réception (R/W=1).

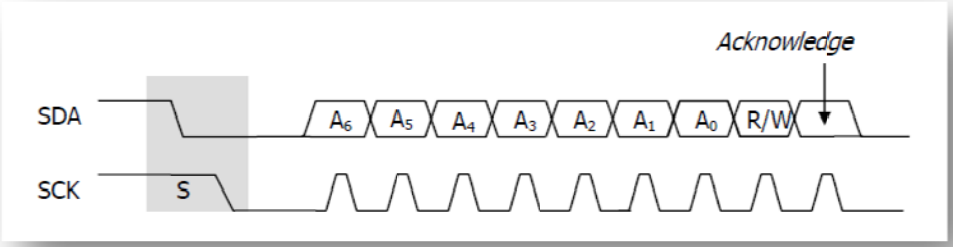


Figure 1.16: l'adresse et le bit R/W.

Le standard I²C supporte maintenant l'adressage sur 10 bits. Comme avec 7 bits on peut adresser jusqu'à 128 composants, cela nous suffira largement.

6.3. Le module MSSP en mode I²C : [15]

Le module MSSP du PIC peut être configuré en master ou en slave. Il utilise les broches RC3/SCL (Horloge) et RC4/SDA (données). Ces broches doivent être configurées en **ENTREE** à l'aide du registre TRISC et doivent être munie de résistances de pull-up externes nécessaire au fonctionnement I²C. Les fréquences d'horloges supportées sont 100 kHz, 400 kHz et 1 MHz L'accès au module en lecture et écriture se fait à l'aide du registre tampon (buffer) SSPBUFF. La transmission et la réception se fait à l'aide du registre à décalage SSPSR auquel nous n'avons pas directement accès.

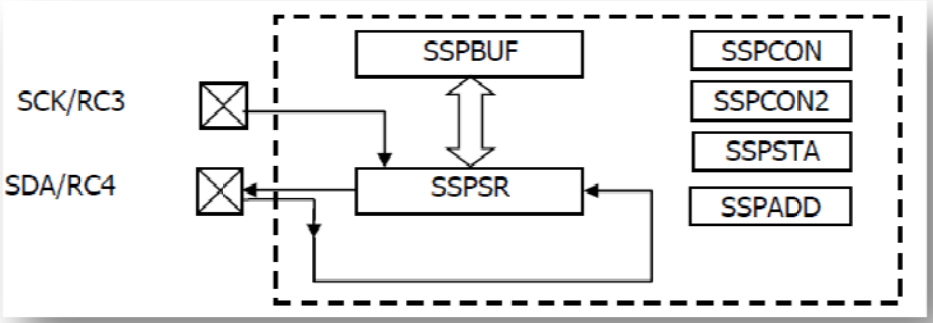


Figure 1.17 : Le module MSSP en mode I²C.

6.3.1. Transmission d'un octet : [15]

Pour transmettre un octet, il suffit de le copier dans le registre SSPBUF, et le module MSSP s'occupe du reste. Au moment de l'écriture dans SSPBUF, le bit BF passe à 1 et la transmission commence. A la fin de la transmission, le bit SSPSTAT.BF repasse à 0 et le drapeau d'interruption PIR1.SSPIF passe à 1. Le bit BF apparaît donc comme un bit très important, c'est lui qui nous permet de savoir si le registre SSPBUF est libre ou non Si on tante d'écrire dans SSPBUF alors que BF=1, le bit SSPCON.WCOL passe à 1 pour indiquer une collision et l'écriture n'a pas lieu.

6.3.2. Réception d'un octet : [15]

A la fin de la réception d'un octet, celui-ci est transféré dans SSPBUF, l'indicateur SSPSTAT.BF et le drapeau d'interruption PIR1.SSPIF passent à 1. BF repasse automatiquement à 0 au moment de la lecture de SSPBUF alors que SSPIF soit être remis à 0 par soft. Si le PIC termine la réception d'un octet avant que l'octet précédent qui se trouve dans SSPBUF n'ait été lu, on a un Overflow qui sera signalé par le drapeau SSPOV. Le transfert n'a pas lieu, l'octet arrivé est perdu.

6.3.3. Les registres de configuration : [15]

Le control du module se fait à l'aide des registres :

- SSPCON : registre de control (BANK0)
- SSPCON2 : registre de control (BANK1)
- SSPSTAT : registre d'état (BANK1)
- SSPADD : registre d'adresse (BANK1)

6.3.4. Le module SSP du PIC 16F88 [15]

Le module SSP (Synchronous serial port) du PIC 16F88 gère le bus I2C, mais seulement en mode esclave. C'est la principale différence avec le module MSSP (Master Synchronous serial port) du PIC 16F87X, qui gère également (au sens hardware) le mode maître et le mode multi-maîtres. Cependant, le module SSP du PIC 16F88 permet d'implanter de manière logicielle le mode I2C maître ou multi-maîtres.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

VI. Conclusion

Au niveau du ce chapitre, nous avons présenté les microcontrôleurs en général et le PIC 16F88 particulièrement qui est le PIC avec lequel on travail dans ce projet ; nous avons cité quelques caractéristiques importantes ainsi que leur module de communication entre la carte d'acquisition réalisée (qui sera présentée plus tard) et l'ordinateur.

Chapitre II
mise en forme et acquisition des données

I. Introduction

Nous présenterons dans ce chapitre quelques principes généraux qui permettraient de mettre en œuvre des phénomènes intervenant dans l'acquisition de données, les filtres analogiques, les filtres numériques, la convection analogique et la liaison RS232

II. Système d'acquisition de données

1. Définition :

Le système d'acquisition des données (S.A.D) est un ensemble d'éléments matériel et logiciel destiné à recueillir des données physiques par l'intermédiaire de capteurs.

1.1. Une Chaine d'acquisition comporte : [4]

- ✓ Une source d'information (le patient par exemple).
- ✓ Les capteurs capables de transformer l'information physiologique en une grandeur électrique.
- ✓ Les circuits de mise en forme réalisant les fonctions d'amplification, de filtrage, de calibration etc…
- ✓ La carte d'acquisition des données proprement dite permettant l'interfaçage homme machine.
- ✓ Un support logiciel réalisant l'acquisition, l'affichage, le traitement et la transmission des données conformément a un protocole de communication.

1.2.Schéma bloc d'une chaine d'acquisition :

Une chaine d'acquisition peut se représenter selon le Schéma bloc suivant :

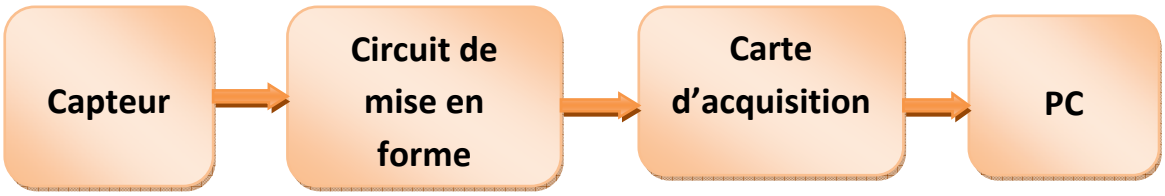


Figure 2.1 : Structure d'une chaine d'acquisition

1.3.Carte d'acquisition de donnée : [14]

Ensemble de circuits électroniques ou interface d'entrée/sortie, destiné à traiter l'information venant des capteurs: échantillonnage, digitalisation, etc.

1.4.Schéma bloc d'une carte d'acquisition :

Une carte d'acquisition numérique peut se représenter selon la figure suivante :

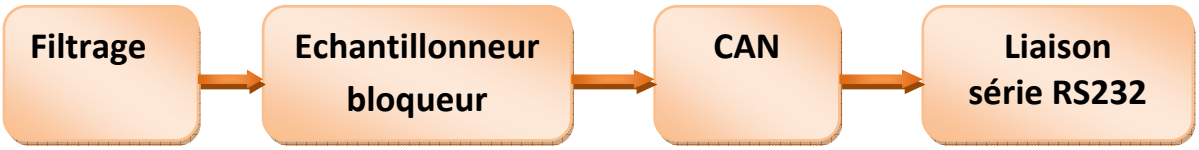


Figure 2.2 : Structure d'une carte d'acquisition

III. La conversion analogique numérique. [13]

1. Identification de la fonction C.A.N :

On appelle **C**onvertisseur **A**nalogique **N**umérique (**C.A.N.**) tout dispositif électronique qui transforme une grandeur analogique d'entrée ue en un nombre binaire de sortie N proportionnel à cette grandeur ue. En anglais, le Convertisseur Analogique Numérique est appelé **A**nalogic **D**igital **C**onversion (**A.D.C.**)

2. Symbole d'un C.A.N

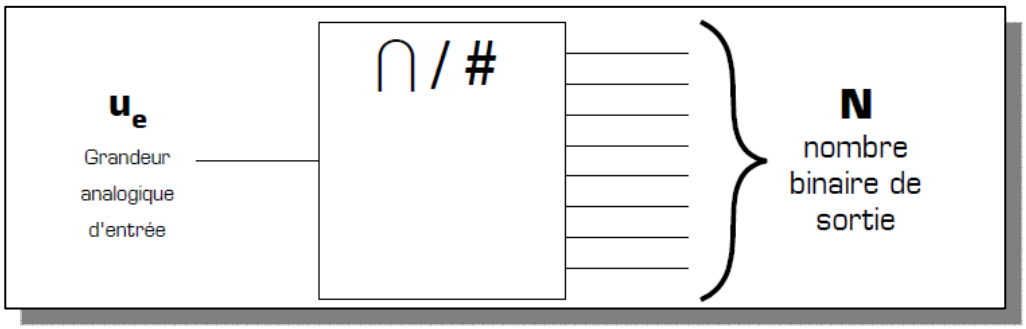


Figure 2.3 : Symbole de la fonction Conversion Analogique / Numérique

Le signe ∩ indique que la grandeur est de type **analogique** (il s'agit ici de l'entrée).

Le signe # indique que la grandeur est de type **numérique** (il s'agit ici de la sortie).

3. Principe

L'obtention d'un signal numérique à partir d'un signal analogique (exemple : microphone) nécessite de mettre en œuvre une chaîne de conversion analogique-numérique. Le convertisseur analogique-numérique a pour fonction de faire correspondre à un signal analogique un signal numérique de sortie.

Cette chaîne comporte typiquement la structure fonctionnelle suivante :

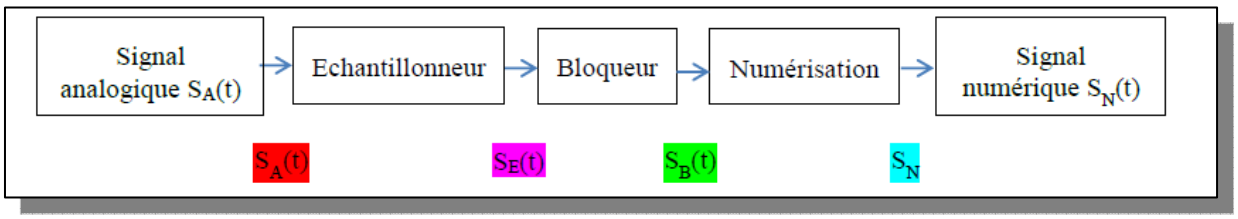


Figure 2.4 : Principe du chaîne de conversion analogique-numérique [13]

Les fonctions réalisées par chaque élément sont les suivantes :

- Echantillonneur : Acquérir la grandeur analogique à un instant t ;
- Bloqueur : Maintenir cette grandeur pendant la durée de la numérisation ;
- Numérisation : Convertir le signal analogique bloqué en un nombre (sur N bits).

4. Echantillonnage

L'échantillonneur a pour rôle de prélever périodiquement la valeur du signal analogique. Il réalise une « photographie » du signal qui « fige » son image jusqu'à la « prise » suivante. L'intervalle de temps entre 2 instants consécutifs d'échantillonnage constitue la période d'échantillonnage, notée Te, du système de conversion analogique numérique.

Le choix de cette période est imposée à la fois par la nature du signal que l'on veut numériser mais aussi par le temps de met le convertisseur analogique numérique pour réaliser sa conversion. Usuellement, on utilise la notion de fréquence d'échantillonnage telle que

$$\mathbf{F_E = \frac{1}{T_E}}$$

Un choix judicieux de cette fréquence est imposé par l'application du théorème de Shannon. Celui-ci précise qu'elle doit être au moins égale au double de la plus grande des fréquences du signal analogique.

Le signal échantillonné, à l'instant t (qui est un multiple entier de la période d'échantillonnage), est noté de la façon suivante que :

$$\mathbf{SE(t)=SE\ (nTE)=SA(t)}$$

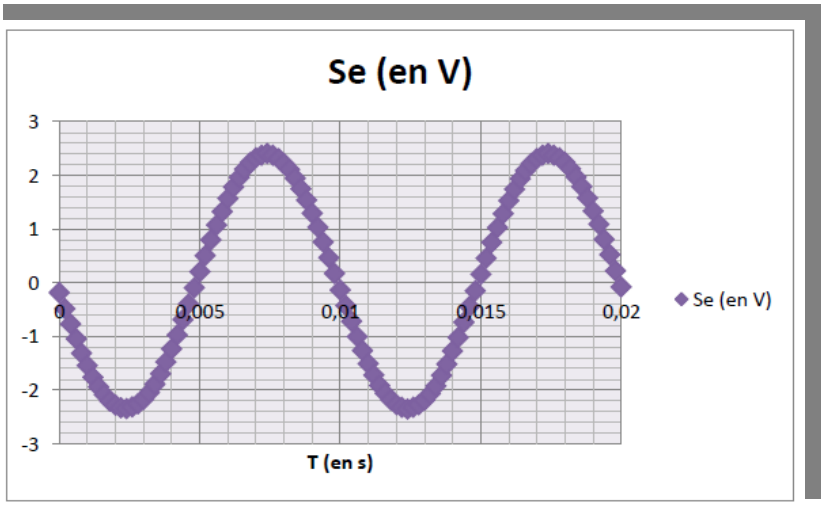


Figure 2.5 : exemple de signal échantillonné

La figure ci-dessus représente l'allure du signal échantillonné avec une période d'échantillonnage de 200 µs. A chaque instant (multiple de 200 µs), une valeur du signal analogique est acquise (comme une photo).

D'une manière générale, un signal véhiculant une information contient un certain nombre de fréquences correspondant à son spectre. Par nature, pour chaque fréquence, l'amplitude est aléatoire. Il est donc nécessaire de considérer une bande de fréquence dans laquelle se trouvent de façon pertinente les informations relatives au signal. On fixe ainsi une fréquence maximale du signal analogique et par conséquent on accepter de « perdre » une partie de l'information.

Ce choix doit être pertinent en trouvant un compromis entre rapidité de conversion et qualité de numérisation.

5. Bloqueur :

Le bloqueur a pour fonction de maintenir constant le signal échantillonné $S_E(t)$ afin de permettre au convertisseur analogique numérique de le numériser. Ce signal est maintenu constant à la valeur $S_B(t)$ jusqu'à l'échantillonnage de la valeur suivante.

Actuellement les bloqueurs agissent comme une mémoire qui garde constante la valeur échantillonnée ($S_B(t)$=constante), on parle alors de bloqueur d'ordre 0. On trouve aussi des bloqueurs d'ordre supérieur (1, 2..), lorsque l'évolution du signal $S_B(t)$ est assimilable à des décroissances exponentielles ou paraboliques.

5.1. Numérisation :

5.1.1. Principe :

La numérisation est la transformation d'un signal analogique issu du bloqueur en un signal numérique. Elle est réalisée par un convertisseur analogie numérique (en anglais : ADC) qui convertit le signal bloqué en un nombre. La sortie du convertisseur est raccordée ensuite à un système numérique (ordinateur, analyseur de spectre numérique…)

Cette numérisation réalise la quantification du signal analogique bloqué en lui associant un nombre. Ce nombre est compris entre 0 et $\mathbf{2^N}$-**1** valeurs possibles où N est le nombre de bits en sortie du convertisseur. Ainsi pour 8 bits, on a $\mathbf{2^8}$ valeurs différentes possibles en sortie soit 256 valeurs différentes. Les nombres associés vont alors de la valeur 0 jusqu'à 255.

5.1.2. Résolution :

La résolution du convertisseur, notée r, est l'intervalle de tension analogique permettant de passer du nombre n à (n+1). Pour une tension analogique d'entrée allant de 0 V à une amplitude V_{MAX} , on a $\boldsymbol{r = \frac{V_{MAX}}{2^N - 1}}$ **.** Si cette tension d'entrée varie de 0 à 10 V et que l'on utilise un convertisseur analogique numérique de 8 bits, on a une résolution de 39,2 mV. En d'autres termes, pour que la valeur numérique en sortie du convertisseur varie, il faut la tension d'entrée ait varié de 39,2 mV.

5.1.3. Réalisation :

Il existe plusieurs principes pour réaliser des convertisseurs analogiques numériques qu'ils soient construit en éléments discrets (ADI, portes logiques et éléments résistifs) ou obtenus avec des circuits intégrés.

✓ Ces principes sont les suivants :

- **Conversion par rampe** : il s'agit de comparer la grandeur analogique à celle d'une rampe de tension. Lors de l'égalité entre les deux, on note la valeur issue d'un compteur. Cette technique est lente mais précise. On l'emploie lors de l'acquisition de mesures .
- **Conversion par approximations successives :** il s'agit de comparer le signal analogique à celui issu d'un convertisseur numérique analogique (transformation d'un signal numérique en un signal analogique) qui s'incrémente automatique. A l'égalité des valeurs, on note applique la tension issue du CNA à l'entrée d'un CAN afin d'obtenir le signal numérique recherché. Cette technique est utilisée pour les signaux audio

- **Convertisseur flash :** ceux sont des convertisseurs rapides mais coûteux pouvant être utilisés pour la vidéo ou l'analyse de signaux.

IV. Le filtrage analogique [8]

1. Introduction :

D'une manière générale, un filtre est un système linéaire dont le rôle est de modifier la composition spectrale d'un signal sans y ajouter de nouvelles composantes. Il permet le renforcement ou l'atténuation d'une ou plusieurs bandes de fréquences.

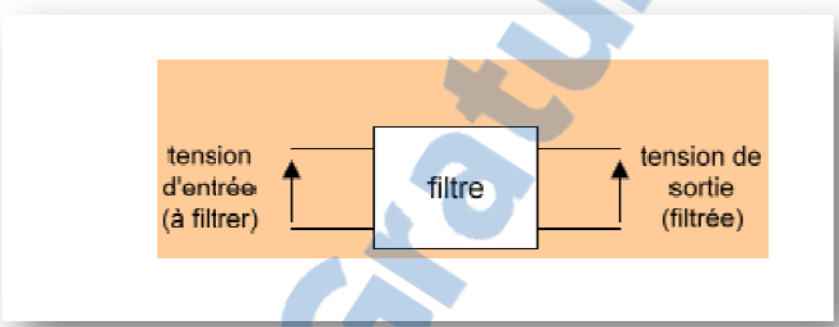
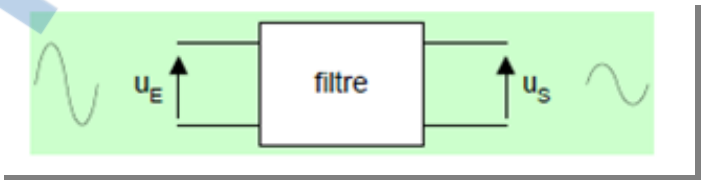


Figure 2.6 : structure de fonction de filtre

Un filtre est un circuit dont le comportement dépend de la fréquence. Un filtre est un circuit linéaire.

 Si la tension d'entrée est sinusoïdale alors la tension de sortie est sinusoïdale de même fréquence.

2. Etude du filtre en régime sinusoïdal



La principale caractéristique d'un filtre est sa ***réponse en fréquence*** : $A_V(f)$

A_V Désigne l'amplification en tension :

$$A_V = \frac{\hat{u}_S}{\hat{u}_E} = \frac{\text{amplitude de la tension de sortie}}{\text{amplitude de la tension d'entrée}}$$

2.1.Filtre actif et filtre passif :

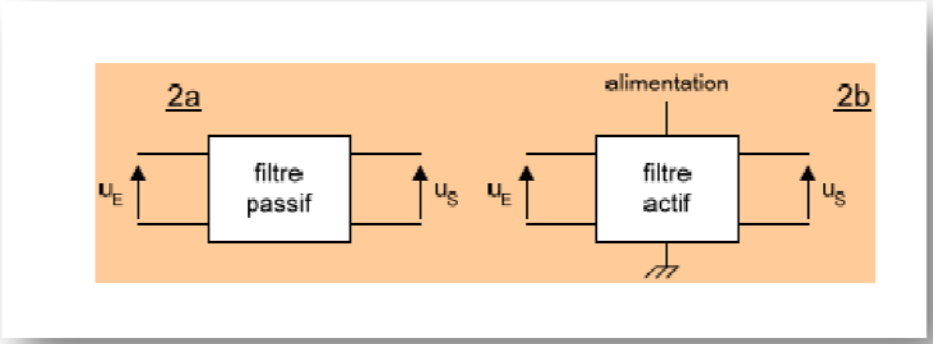
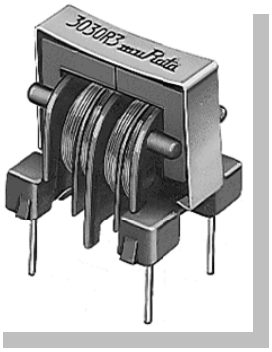


Figure 2.7 : 2a (filtre passif) et 2b (filtre actif)

- filtre passif : on y trouve résistances, bobines et condensateurs.



- filtre actif : alimentation externe, transistors, A.O.

2.2. Les principaux types de filtres (idéaux) :

a- Filtre passe-bas

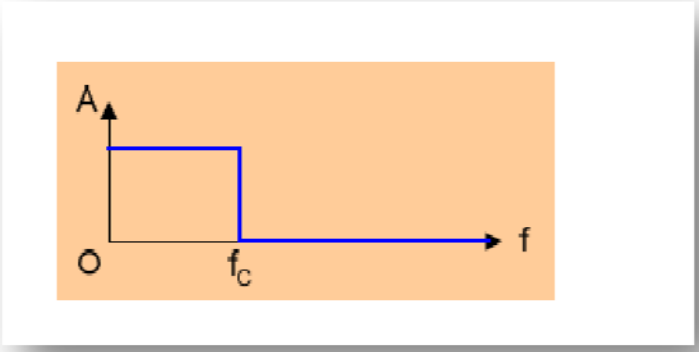


Figure 2.8 : structure de filtre passe bas

Ce filtre ne laisse passer que les basses fréquences du signal d'entrée.

Les hautes fréquences sont donc filtrées. La limite entre BF et HF est appelée *fréquence de coupure* f_c

- ✓ La *bande passante* est la gamme de fréquence non filtrée : **BP = [0,f_c]**
- ✓ A noter que les signaux continus (f = 0) ne sont pas filtrés.

b- Filtre passe-haut :

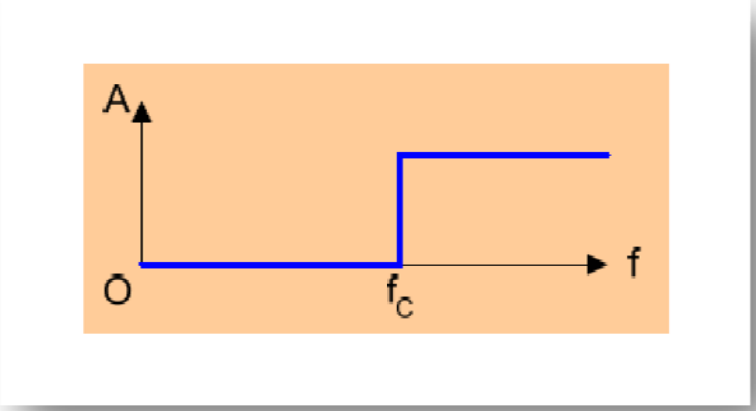


Figure 2.9 : structure de filtre passe haut

Ce filtre ne laisse passer que les hautes fréquences. **BP = [f_c, ∞[**

c- Filtre passe-bande :

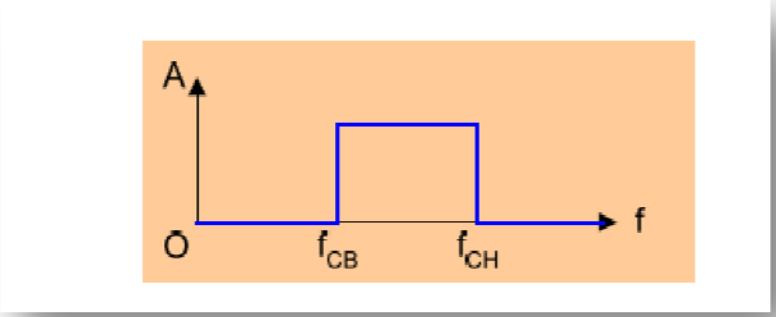


Figure 2.10 : structure de filtre passe-bande

Ce filtre ne laisse passer qu'une bande de fréquences.

Il possède deux fréquences de coupure :

- la fréquence de coupure basse
- et la fréquence de coupure haute

BP = [f_{CB},f_{CH}]

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

d- Filtre coupe-bande (ou réjecteur de bande) :

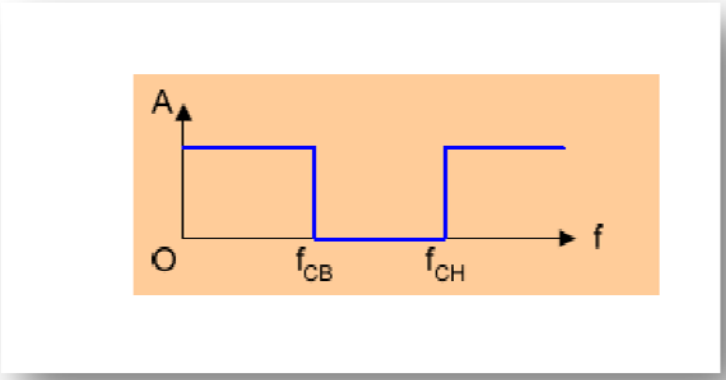


Figure 2.11 : structure de filtre coupe-bande

2.3. Filtres réels :

Prenons l'exemple d'un filtre passe-bande :

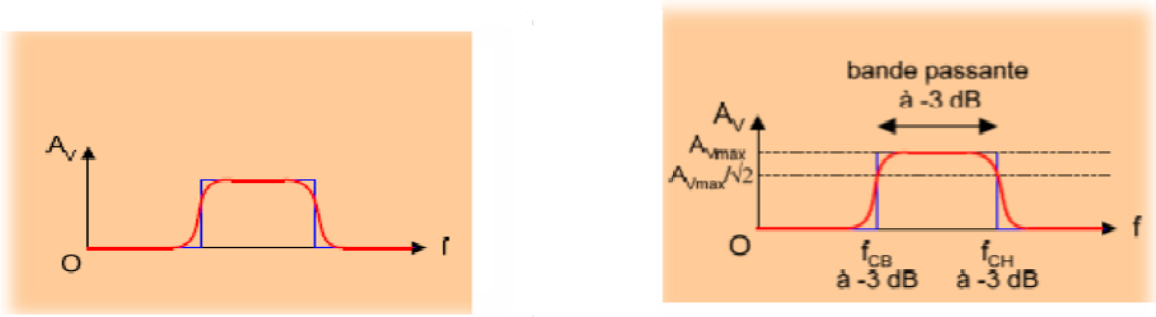


Figure 2.12 : structure de filtre réel

Les fréquences de coupure « a – 3 dB » sont définies de la manière suivante :

ce sont les fréquences qui correspondent à l'amplification maximale divisée par $\sqrt{2}$

$$A_V(f_c) = \frac{A_{V\max}}{\sqrt{2}}$$

- **Diagramme de Bode du gain :**

Le diagramme de Bode donne le gain en fonction de la fréquence (ou de la pulsation).

L'échelle des fréquences est logarithmique :

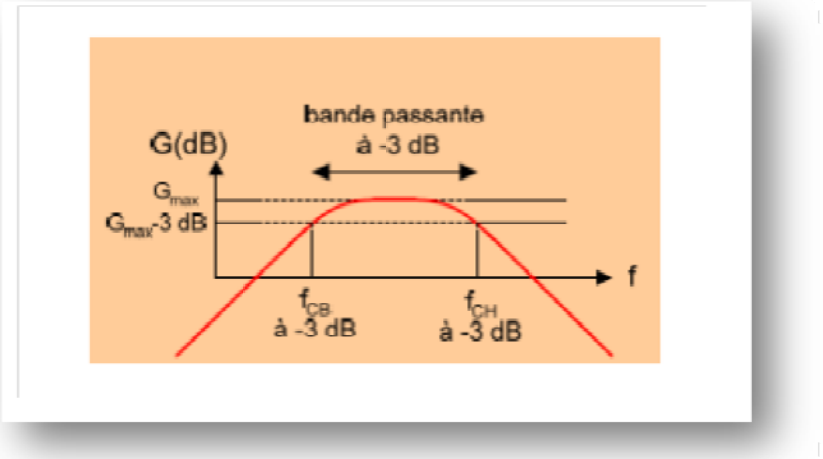
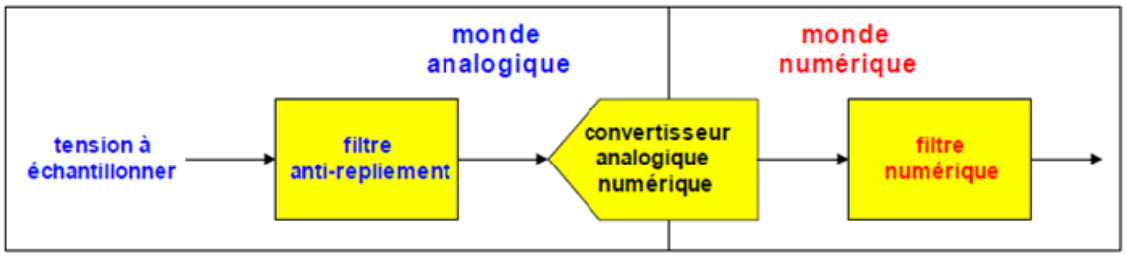


Figure 2.13 : structure de diagramme de BODE de filtre réel

3. Filtre anti-repliement [16]



C'est un filtre analogique passe-bas de fréquence de coupure **Fe / 2** (dans l'idéal). Il est utilisé généralement en amont de convertisseur analogique – numérique pour éliminer le problème de repliement de signal.

Le filtre anti-repliement permet de travailler dans la zone « utile » du filtre numérique, c'est-à-dire dans la bande de fréquences située entre 0 Hz et 500 Hz

V. Filtrage numérique d'un signal lent avec un microcontrôleur. [16]

1. Introduction :

On suppose dans cette partie que le signal à échantillonner est un **signal d'évolution lente** (de période très supérieure à la seconde). Ce pourra être le signal provenant (par exemple d'un capteur de température analogique (LM335) ou d'un capteur de pression « MPX2200 »)

2. Définitions générales :

Il existe plusieurs types de filtres numériques linéaires :

- Filtre numérique non récursif (ou filtre à réponse impulsionnelle finie RIF)
- Filtre numérique récursif (ou filtre à réponse impulsionnelle infinie RII)

Nous ne nous intéresserons qu'aux filtres numériques « non récursifs », dont voici l'expression générale :

$$\boxed{y_k = \sum_{\ell=0}^{n-1} h_\ell x_{k-\ell}}$$

$\mathbf{x_{k-l}}$ Représente la valeur numérique fournie par le convertisseur analogique – numérique à l'instant $\boldsymbol{t = kT_e - lT_e}$

$\boldsymbol{y_k}$ Représente la valeur numérique en sortie du filtre numérique à l'instant $\boldsymbol{t = kT_e}$

Un filtre numérique est caractérisé par :

→ n : le nombre d'échantillons (c'est un nombre entier, supérieur à 1).

→ $h_0, h_1, h_2, \dots \dots \dots \dots . . h_{n-1}$: n coefficients réels.

$\boldsymbol{T_e}$ est la période d'échantillonnage du convertisseur analogique – numérique.

$\boldsymbol{F_e = 1/T_e}$ est la fréquence d'échantillonnage.

Par définition, la fonction de transfert en z du filtre s'écrit :

$$\boxed{H(z) = \sum_{\ell=0}^{n-1} h_\ell z^{-\ell}}$$

3. Filtre numérique à « moyenne glissante » :

Pour des raisons de simplicité, nous choisirons un filtre numérique non récursif avec des coefficients identiques, tels que :

$$\boldsymbol{h_{l=} \frac{1}{n}}$$

D'où :

$$\boxed{\begin{aligned} y_k &= \sum_{\ell=0}^{n-1} \frac{1}{n} x_{k-\ell} \\ H(z) &= \frac{1 + z^{-1} + z^{-2} + \dots + z^{-(n-1)}}{n} \end{aligned}}$$

En fait, **ce filtre numérique calcule la valeur moyenne des n derniers échantillons**.

On parle aussi de **filtre à « moyenne glissante ».**

Toujours pour des raisons de simplicité, le nombre d'échantillons n sera une puissance de 2.

- Par exemple avec n= 2^3= 8

$$y_k = \frac{X_k + X_{k-1} + X_{k-2} + X_{k-3} + X_{k-4} + X_{k-5} + X_{k-6} + X_{k-7}}{8}$$

$$H(Z) = \frac{1 + Z^{-1} + Z^{-2} + \cdots + Z^{-7}}{8}$$

On suppose que le convertisseur analogique – numérique a une résolution de 10 bits (nombre binaire de 10 bits), avec une fréquence d'échantillonnage de 1 kHz (c'est-à-dire une conversion toutes les 1 ms).

Voici un exemple des 8 dernières conversions de ce convertisseur :

		Binaire naturel	Décimal
x_{k-7}	mesure à l'instant t – 7 ms	01 11111110	510
x_{k-6}	mesure à l'instant t – 6 ms	10 00001001	521
x_{k-5}	mesure à l'instant t – 5 ms	01 11111101	509
x_{k-4}	mesure à l'instant t – 4 ms	10 00000011	515
x_{k-3}	mesure à l'instant t – 3 ms	10 00000000	512
x_{k-2}	mesure à l'instant t – 2 ms	10 00001010	522
x_{k-1}	Avant dernière mesure (à l'instant t – 1 ms)	01 11111011	507
x_k	Dernière mesure (à l'instant t)	01 11111111	511

La dispersion des mesures est en partie due à la présence de bruit.

Un microcontrôleur sait très bien faire les additions (en binaire).

Par exemple le microcontrôleur Microchip PIC 16F88, possède l'instruction **addwf** qui lui permet d'additionner 2 nombres de 8 bits, en un cycle d'horloge (soit 0,2 µs avec une horloge à quartz de 20 MHz).

La somme de 8 nombres de 10 bits donne un résultat sur 10 + 3 = 13 bits :

$$x_k + x_{k-1} + x_{k-2} + x_{k-3} + x_{k-4} + x_{k-5} + x_{k-6} + x_{k-7}$$
$$= 10000\ 00001011 \text{ (en binaire naturel)}$$
$$= 4017 \text{ (en décimal)}$$

Il faut maintenant faire une division pour obtenir la valeur moyenne.

Un microcontrôleur standard, comme le PIC 16F88, ne possède pas d'instruction de division (ni d'instruction de multiplication).

Le PIC 16F88 doit reconstruire la division uniquement avec des instructions d'additions, ce qui donne lieu à une routine assez complexe (et lente). Heureusement, la division devient triviale quand on divise par une puissance de 2.

Pour diviser par 2^n, on décale la virgule de n bits vers la gauche.

(C'est la même chose en décimal : pour diviser par 10^n, on décale la virgule de n chiffres vers la gauche : 6594 / 1000 = 6,594).

Ainsi, pour diviser par 8, il suffit de décaler la virgule de 3 bits vers la gauche :

10000 00001011 ⟶ 10 00000001,011

Ce qui donne pour la partie entière :

10 00000001

= 513 en décimal

Et pour le reste : 011 = 0,5×0 + 0,25×1 + 0,125×1 = 0,375

Finalement : **y_k = 513,375** (4107 / 8 = 513,375)

On remarquera que la valeur moyenne sur 8 conversions donne un nombre de 13 bits.

On augmente ainsi, de manière logicielle, la résolution du convertisseur analogique – numérique (on passe de 10 à 13 bits).

4. Réponse en fréquence d'un filtre numérique

Voici l'expression générale de la fonction de transfert :

$$H(f) = \sum_{\ell=0}^{n-1} h_\ell e^{-2\pi j f \ell T_e}$$

f : fréquence en hertz du signal échantillonné (supposé sinusoïdal)

Avec notre filtre à moyenne glissante $(\boldsymbol{h_l = \frac{1}{n}})$

$$H(f) = \sum_{\ell=0}^{n-1} \frac{1}{n} e^{-2\pi j f \ell T_e}$$

que l'on peut écrire plus simplement :

$$H(f) = \frac{1}{n} \frac{\sin(2\pi f \cdot \frac{nT_e}{2})}{\sin(2\pi f \cdot \frac{T_e}{2})} e^{-2\pi j f \frac{(n-1)}{2} T_e}$$

VI. La gestion de la liaison RS232 [14]

1. communication série asynchrone a travers le port série RS232

Les liaisons séries permettent la communication entre deux systèmes numériques en limitant le nombre de fils de transmission. La liaison série aux normes de RS232 est utilisée dans tous les domaines de l'informatique. Elle est de type asynchrone, c'est-a-dire qu'elle ne transmet pas le signal de l'horloge.

Le schéma fonctionnel est le suivant :

Figure 2.14 : schéma fonctionnel d'une liaison série asynchrone de la norme RS232

La transmission série nécessite au moins 2 fils de communication, l'un pour la transmission (Tx) et l'autre pour la réception (Rx) et un fil de masse.

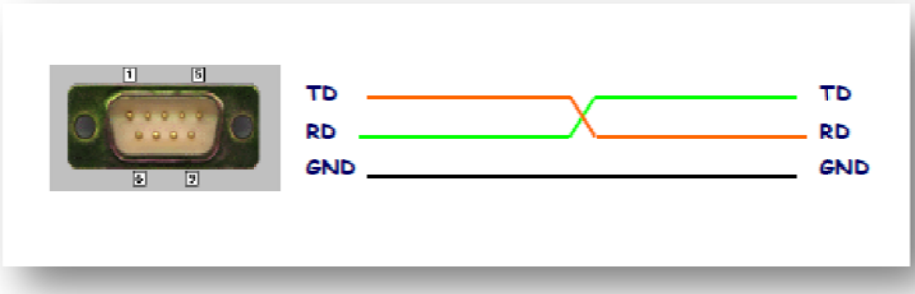


Figure 2.15 : la liaison série croisée

2. Protocole de transmission

Afin que les éléments communicants puissent se comprendre, il est nécessaire d'établir un protocole de transmission. Ce protocole devra être le même pour les deux éléments afin que la transmission fonctionne correctement.

Paramètres rentrant en jeu :

- Longueur des mots : 7 bits (ex : caractère ascii) ou 8 bits

- La vitesse de transmission : les différentes vitesses de transmission son réglables a partir de 110 bauds (bits par seconde) de la façon suivante : 110 bds, 150 bds, 300 bds, 600 bds, 1200 bds, 2400 bds, 4800 bds, 9600 bds.

- Parité : le mot transmis peut être suivi ou non d'un bit de parité qui sert a détecter les erreurs éventuelles de transmission. Il existe deux types de parité.
  - parité paire : le bit ajoute a la donnée est positionne de telle façon que le nombre des états 1 soit paire sur l'ensemble donne + bit de parité ex : soit la donnée 11001011 contenant 5 état 1, le bit de parité paire est positionne a 1, ramenant ainsi le nombre de 1 a 6.
  - parité impaire : le bit ajoute a la donnée est positionne de telle façon que le nombre des états 1 soit impaire sur l'ensemble donne + bit de parité ex : soit la donnée 11001011 contenant 5 état 1, le bit de parité paire est positionne a 0, laissant ainsi un nombre de 1 impaire.
- Bit de start : la ligne au repos est a l'état logique 1 pour indiquer qu'un mot va être transmis la ligne passe a l'état bas avant de commencer le transfert. Ce bit permet de synchroniser l'horloge du récepteur.
- Bit de stop : après la transmission, la ligne est positionnée au repos pendant 1, 2 ou 1,5 période d'horloge selon le nombre de bits de stop.

Le bit de start apparait en premier dans la trame puis les données (poids faible en premier), la parité éventuelle et le (les) bit(s) de stop. Exemple :

Soit a transmettre en parité paire, avec 2 bits de stop, le caractère B dont le codage ascii est 1000010(2) la trame sera la suivante :

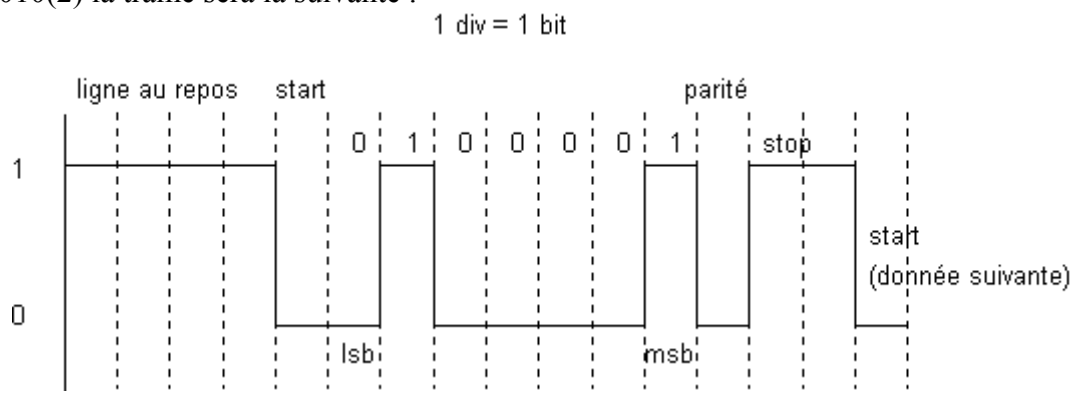


Figure 2.16 : Exemple de transmission série

VII. Conclusion

Ce chapitre a été consacré essentiellement à la description de manière plus appropriée du système d'acquisition utilisée dans notre projet de fin d'étude et des données échangées ainsi que les phénomènes intervenant dans la partie mise en forme comme le filtrage que se soit numérique ou analogique et l'échantillonnage. Dans le chapitre suivant nous allons voir plus en détaille la carte utilisée.

Rapport-Gratuit.com

Chapitre III

Etude et réalisation de la carte d'acquisition

I. INTRODUCTION

1. Présentation :

Il s'agit d'une carte d'acquisition à 7 entrées analogiques pour signaux lents, qui conviendra pour l'acquisition de températures, de pressions atmosphériques, etc. ...

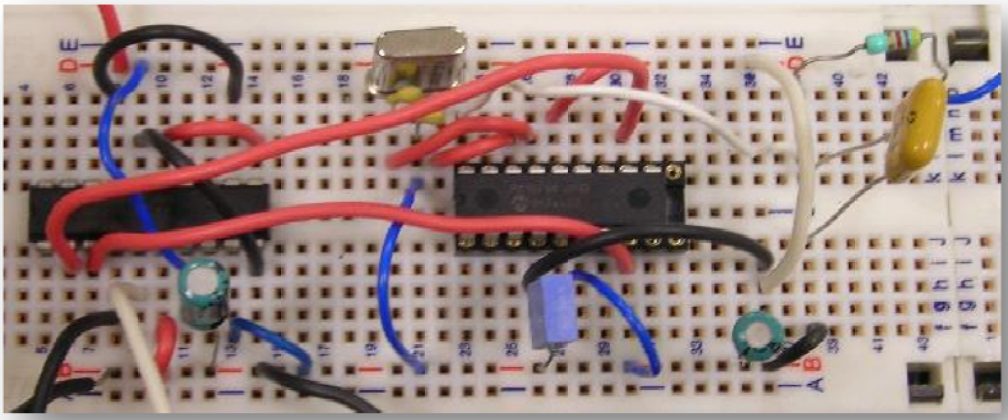


Figure 3.1 : circuit de la carte sur plaque d'essai

- La fréquence d'acquisition est de 1 Hz maximum par voie (c'est-à-dire 60 acquisitions maximum par minute).
- Le convertisseur analogique - numérique (module interne du microcontrôleur PIC 16F88).a les caractéristiques suivantes :
    - Plage de tension analogique : 0,000 V à 5,000 V
    - Résistance interne de la source de tension à échantillonner : 5000 ohms maximum
    - Résolution numérique : 10 bits (5 mV)

N.B. Un filtrage numérique améliore la résolution : 1 mV en pratique

- La carte se connecte à un ordinateur (via une liaison RS232).

II. SHEMA ELECTRIQUE DE LA CARTE

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

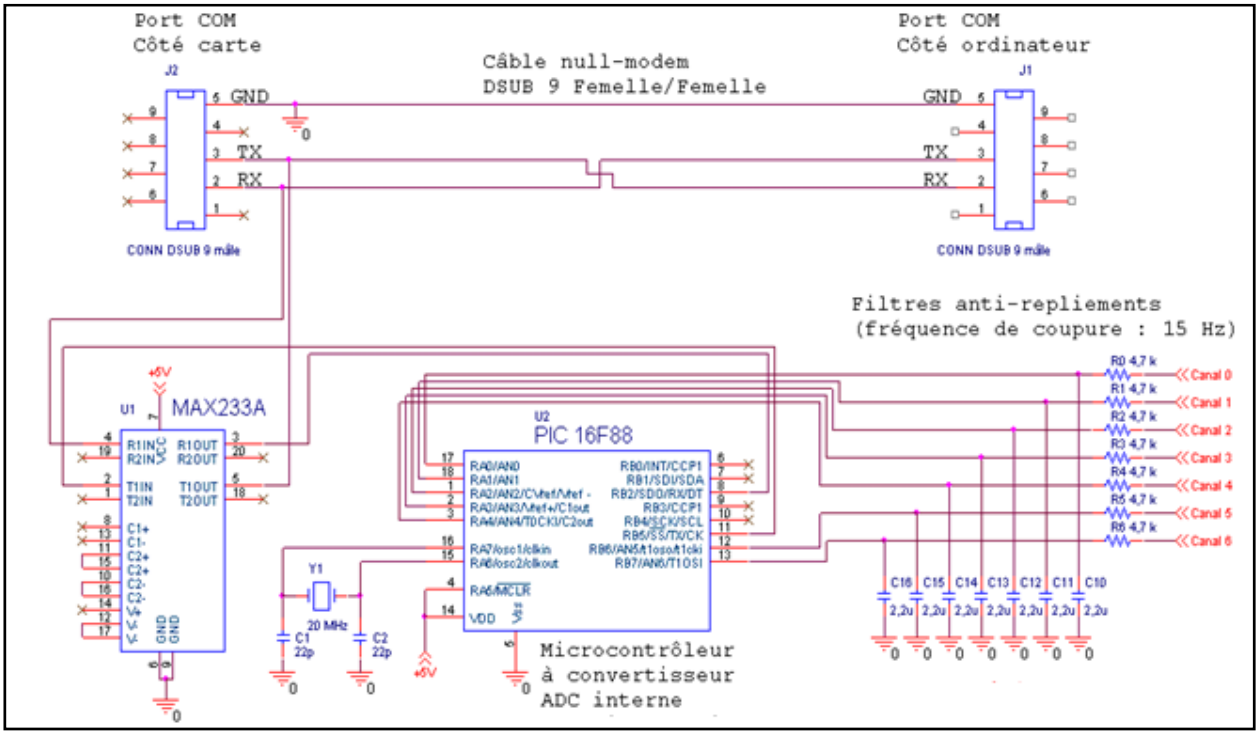


Figure 3.2 : schéma électrique de la carte

III. PRINCIPE DE FONCTIONNEMENT

1. Les filtres anti-repliements :

Un filtre anti-repliement est intercalé entre la tension à mesurer (qui provient généralement d'un capteur) et le canal d'entrée de l'ADC du PIC.

Il s'agit ici d'un simple filtre analogique passe-bas

Fréquence de coupure à – 3 dB :

R = 4,7 kΩ
C = 2,2 µf

$$f_c = \frac{1}{2\pi RC} \approx 15\ \text{Hz}$$

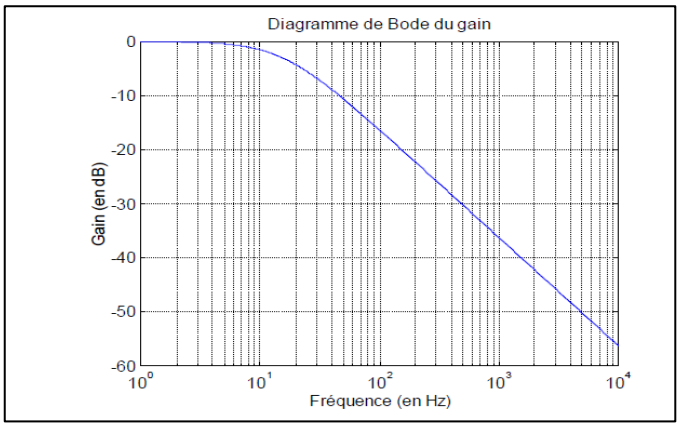


Figure 3.3 : diagramme de BODE de fréquence de coupure

2. Le convertisseur analogique - numérique (ADC) :

ADC = **A**nalog to **D**igital **C**onverter = convertisseur analogique - numérique (CAN en français). Le rôle du convertisseur analogique - numérique est de transformer une tension analogique en un nombre binaire (proportionnel à la tension analogique).

Le microcontrôleur PIC 16F88 a l'avantage de posséder un module ADC interne, contrairement au PIC 16F84A ou 16F628A (il faut alors adjoindre un ADC externe, ce qui complique le schéma électrique et la programmation du PIC).

2.1. Caractéristiques principales du module ADC du PIC 16F88 :

- Plage de tension analogique : 0,000 V à 5,000 V
- Résistance interne de la source de tension à échantillonner : 5000 ohms maximum
- Résolution numérique : 10 bits (5 mV)
- type : à approximations successives
- résolution : 10 bits
- nombre de canaux : 7 (multiplexés)
- période d'échantillonnage : environ 42 µs

→ Principaux registres relatifs au module ADC :

- **ANSEL** (adresse 0x9B, banque 1)
- **ADCON0** (adresse 0x1F, banque 0)
- **ADCON1** (adresse 0x9F, banque 1)
- **ADRESH** (adresse 0x1E, banque 0)
- **ADRESL** (adresse 0x9E, banque 1)

2.2. Mise en œuvre :

➔ **Etape 1 : Configuration**

2.2.1. Choix des canaux d'entrées :

On peut utiliser jusqu'à 7 canaux d'entrées (mais à un instant donné, un seul canal est relié à l'ADC).

Numéro de canal	Broche
0	RA0/AN0
1	RA1/AN1
2	RA2/AN2/VREF-
3	RA3/AN3/VREF+
4	RA4/AN4
5	RB6/AN5
6	RB7/AN6

Tableau 3.1 : choix des canaux

Supposons que l'on ait besoin de 4 des 7 canaux (canal 0, 1, 2 et 5). Il faut configurer les 4 broches correspondantes comme **entrée analogique** :

On commence par configurer ces 4 broches en **entrée** avec les registres spéciaux **TRISA** et **TRISB** (banque 1), puis on configure les entrées comme entrée de type analogique avec le registre spécial **ANCEL** (banque 1).

```
bsf  STATUS, RP0 ; passage en banque 1
bsf  TRISA, 0 ; configuration de la broche RA0 en entrée
bsf  TRISA, 1 ; configuration de la broche RA1 en entrée
bsf  TRISA, 2 ; configuration de la broche RA2 en entrée
bsf  TRISB, 6 ; configuration de la broche RB6 en entrée
movlw  B'00101111' ; W = B'00100111'
movwf  NSEL ; (ANSEL) = B'00100111'
```

- **bit 7** du registre ANSEL = X : non implémenté
- **bit 6** du registre ANSEL = 0 : configuration de la broche **RB7**/AN6 comme entrée/sortie **numérique**
- **bit 5** du registre ANSEL = 1 : configuration de la broche RB6/**AN5** comme **entrée analogique** (canal 5)
- **bit 4** du registre ANSEL = 0 : configuration de la broche **RA4**/AN4 comme entrée/sortie **numérique**

- **bit 3** du registre ANSEL = 0 : configuration de la broche **RA3**/AN3/VREF+ comme entrée/sortie **numérique**
- **bit 2** du registre ANSEL = 1 : configuration de la broche RA2/**AN2**/VREF- comme **entrée analogique** (canal 2)
- **bit 1** du registre ANSEL = 1 : configuration de la broche RA1/**AN1** comme **entrée analogique** (canal 1)
- **bit 0** du registre ANSEL = 1 : configuration de la broche RA0/**AN0** comme **entrée analogique** (canal 0)

2.2.2. Choix des tensions de référence :

Quatre combinaisons sont possibles.

- Elles dépendent de 2 bits du registre spécial **ADCON1** (banque 1) :

VCFG1 (bit 5 du registre ADCON1)	VCFG0 (bit 4 du registre ADCON1)	Tension analogique d'entrée (référence haute)	Tension analogique d'entrée (référence basse)	Remarques
0	0	VDD	VSS	
0	1	VDD	RA2/AN2/**VREF-**	La broche RA2 doit être configurée comme entrée analogique : **bsf TRISA, 2** **bsf ANSEL, 2** Le canal 2 n'est plus disponible.
1	0	RA3/AN3/**VREF+**	VSS	La broche RA3 doit être configurée comme entrée analogique : **bsf TRISA, 3** **bsf ANSEL, 3** Le canal 3 n'est plus disponible.
1	1	RA3/AN3/**VREF+**	RA2/AN2/**VREF-**	Les broches RA2 et RA3 doivent être configurées comme entrée analogique : **bsf TRISA, 2** **bsf TRISA, 3** **bsf ANSEL, 2** **bsf ANSEL, 3** Les canaux 2 et 3 ne sont plus disponibles.

Tableau 3.2 : choix des tensions de référence

VDD = broche d'alimentation positive du microcontrôleur PIC 16F88

VSS = broche de masse (0 V) du microcontrôleur PIC 16F88

Exemple:

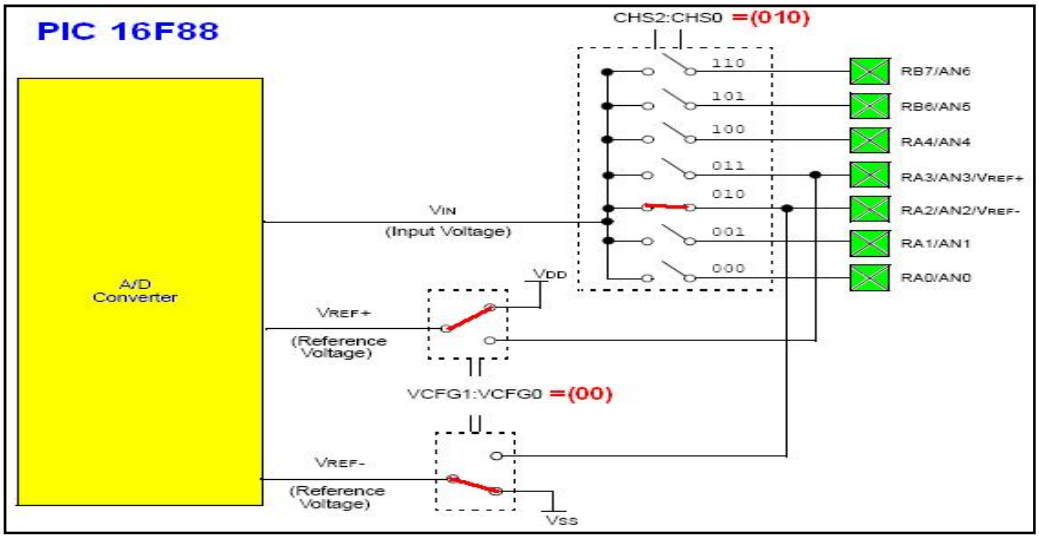


Figure 3.4 : ADC du PIC 16F88

Le microcontrôleur PIC 16F88 est alimenté avec une tension stabilisée de +5,00 V (VDD).

bsf STATUS, RP0 ; passage en banque 1

bcf ADCON1, VCFG0 ; VCFG0 = 0

bcf ADCON1, VCFG1 ; VCFG1 = 0

Avec VCFG1 = 0 et VCFG0 = 0, la plage d'entrée du convertisseur ADC est [VSS à VDD] = [0 V à 5,00 V].

Le convertisseur a une résolution de 10 bits, soit 2^10 = 1024 pas de 4,88 mV (5 V /1024).

0 mV à 4,88 mV => 0x000 (D'0')

4,88 mV à 9,77 mV => 0x001 (D'1')...4,995 à 5,000 V => 0x3FF (D'1023' ou B'11 11111111')

2.2.3. Choix du format du résultat de la conversion :

Le convertisseur ADC fournit un nombre binaire naturel de 10 bits (b9 b8 b7 b6 b5 b4 b3 b2 b1 b0).

b0 = LSB (bit de poids faible)

b9 = MSB (bit de poids fort)

Deux registres (2 x 8 bits) sont donc nécessaires pour stocker le résultat de la conversion (10 bits). Ce sont les registres :

- **ADRESH** (en banque 0)
- **ADRESL** (en banque 1)

Deux formats sont disponibles suivant la valeur du bit ADFM (bit 7 du registre spécial **ADCON1**, situé en banque 1) :

ADFM (bit 7 du registre spécial ADCON1)	Instruction	Registre ADRESH	Registre ADRESL	Exemple
0	bcf ADCON1, ADFM	(b9 b8 b7 b6 b5 b4 b3 b2)	(b1 b0 0 0 0 0 0 0)	D'758' = 0x2F6 (ADRESH) = 0xBD (ADRESL) = 0x80
1	bsf ADCON1, ADFM	(0 0 0 0 0 0 b9 b8)	(b7 b6 b5 b4 b3 b2 b1 b0)	D'758' = 0x2F6 (ADRESH) = 0x02 (ADRESL) = 0xF6

ADFM = A/D Result Format select bit

2.2.4. Choix de la fréquence d'horloge du convertisseur ADC

Le choix de la fréquence d'horloge du convertisseur ADC se fait avec 2 bits du registre spécial **ADCON0** (banque 0) et 1 bit du registre spécial **ADCON1** (banque 1) :

ADCS2 (bit 6 du registre ADCON1)	ADCS1 (bit 7 du registre ADCON0)	ADCS0 (bit 6 du registre ADCON0)	F AD Fréquence d'horloge du convertisseur ADC	T AD Période d'horloge du convertisseur ADC
0	0	0	FOSC / 2	2 x T OSC
0	0	1	FOSC / 8	8 x T OSC
0	1	0	FOSC / 32	32 x T OSC
0	1	1	F RC	T RC
1	0	0	FOSC / 4	4 x T OSC
1	0	1	FOSC / 16	16 x T OSC
1	1	0	FOSC / 64	64 x T OSC
1	1	1	F RC	T RC

Tableau 3.3 : choix de fréquence d'horloge d'ADC

- F OSC est la fréquence de l'horloge du microcontrôleur PIC 16F88.

Par exemple, avec un oscillateur à quartz externe de 20 MHz et

(ADCS2 ADCS1 ADCS0) = (010) :

F OSC = 20 MHz (T OSC = 50 ns)

F AD = F OSC / 32 = 625 kHz

T AD = 1,6 µs

- F RC est la fréquence de l'oscillateur RC interne du microcontrôleur PIC 16F88.
- Pour assurer un fonctionnement correct du convertisseur ADC, Microchip indique que la période **T AD doit être comprise entre 1,6 µs et 6,4 µs**.

Pour une conversion plus rapide, on prendra (si possible) : **TAD = 1,6 µs**

Cela impose une contrainte sur la fréquence de l'horloge du microcontrôleur PIC 16F88 :

ADCS2 (bit 6 du registre ADCON1)	ADCS1 (bit 7 du registre ADCON0)	ADCS0 (bit 6 du registre ADCON0)	F OSC Maximum
0	0	0	1,25 MHz (T AD = 1,6 µs)
0	0	1	5 MHz (T AD = 1,6 µs)
0	1	0	20 MHz (T AD = 1,6 µs)
1	0	0	2,5 MHz (T AD = 1,6 µs)
1	0	1	10 MHz (T AD = 1,6 µs)
1	1	0	20 MHz (T AD = 3,2 µs)

Tableau 3.4 : fréquence maximum d'horloge du PIC 16F88

2.2.5. Mise en service de l'interruption du convertisseur ADC

Uniquement dans le cas où vous voulez utiliser la technique des interruptions (ce que je fais personnellement).

bcf STATUS, RP0 : passage en banque 0

bcf PIR1, ADIF : on efface le drapeau de l'interruption du convertisseur ADC

bsf STATUS, RP0 : passage en banque 1

bsf PIE1, ADIE : autorisation de l'interruption du convertisseur ADC

bsf PEIE, INTCON : autorisation des interruptions des périphériques (dont le module ADC)

bsf GIE, INTCON : autorisation globale des interruptions

- **Le drapeau ADIF est automatiquement mis à 1 à la fin de la phase de conversion.**

  → Une interruption est alors déclenchée.

Dans la routine d'interruption, on pourra lire et traiter le résultat de la conversion (registres **ADRESH** et **ADRESL**). On n'oubliera pas d'effacer le drapeau ADIF !

➔ Etape 2- Mise en service du convertisseur ADC

Pour cela, il faut mettre à 1 le bit ADON du registre **ADCON0** :

bcf STATUS, RP0 ; passage en banque 0

bsf ADCON0, ADON ; ADON = 1

Remarque : Vous aurez compris qu'il faut mettre le bit ADON à 0 pour rendre hors service le convertisseur ADC :

bcf ADCON0, ADON ; ADON = 0

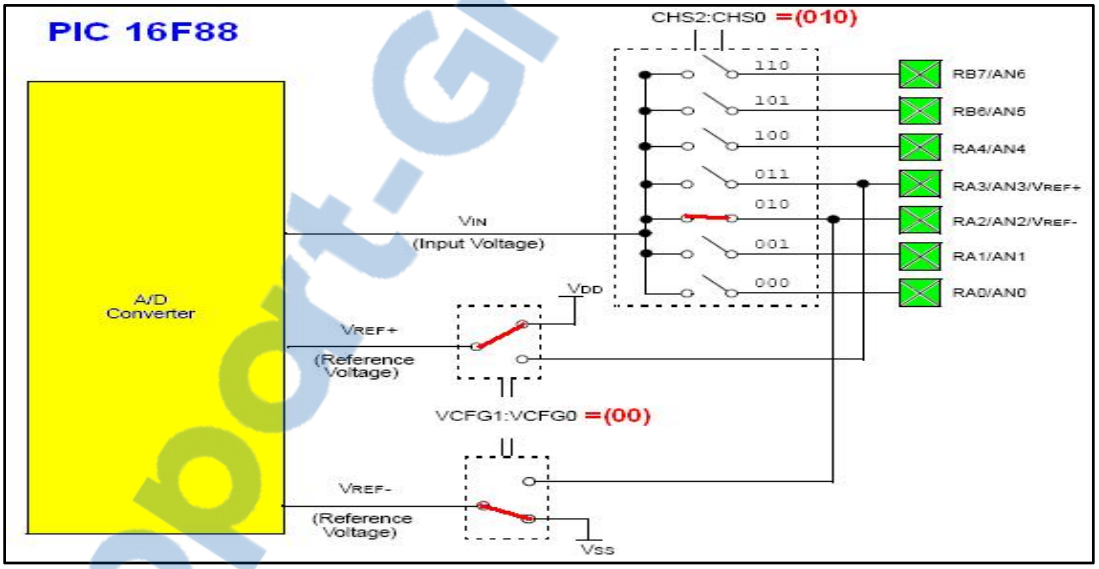
➔ Etape 3- Sélection du canal à échantillonner

La sélection du canal à échantillonner se fait avec 3 bits du registre spécial **ADCON0** (banque0)

CHS2 (bit 5 du registre ADCON0)	CHS1 (bit 4 du registre ADCON0)	CHS0 (bit 3 du registre ADCON0)	Canal sélectionné
0	0	0	canal 0 (broche RA0)
0	0	1	canal 1 (broche RA1)
0	1	0	canal 2 (broche RA2)
0	1	1	canal 3 (broche RA3)
1	0	0	canal 4 (broche RA4)
1	0	1	canal 5 (broche RB6)
1	1	0	canal 6 (broche RB7)

Tableau 3.5 : Sélection du canal à échantillonner

Exemple :



Pour échantillonner le canal 2 :

bcf STATUS, RP0 ; passage en banque 0

bcf ADCON0, CHS2 ; CHS2 = 0

bsf ADCON0, CHS1 ; CHS1 = 1

bcf ADCON0, CHS0 ; CHS0 = 0

✓ **Remarque** : Pour échantillonner plusieurs canaux (canal 0, 1, 2 et 5), on utilise la technique du multiplexage : Echantillonnage du canal 0, puis échantillonnage du canal 1, puis

Rapport-Gratuit.com

échantillonnage du canal 2, puis échantillonnage du canal 5, puis échantillonnage du canal 0, etc…

➔ Etape 4- Attente pendant la phase d'acquisition

Pendant la phase d'acquisition vous devez programmer une temporisation de 19,7 µs (environ).

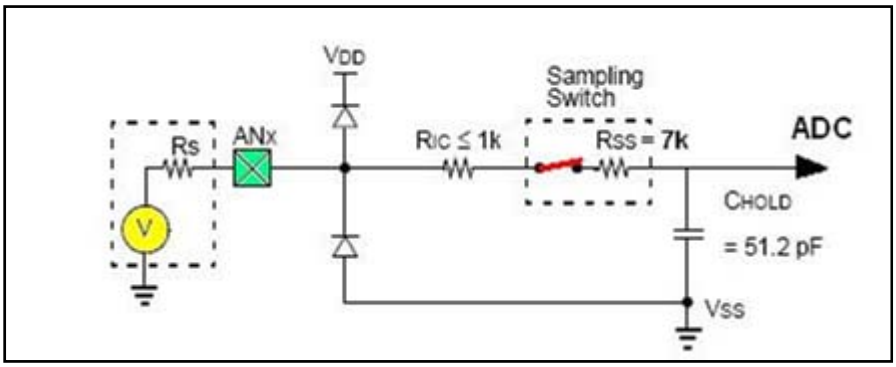


Figure 3.5 : échantillonneur-bloqueur de l'ADC

En effet, la phase d'acquisition consiste à charger le condensateur de maintien **C HOLD** avec la tension présente à l'entrée (V). Mais la durée de charge du condensateur augmente avec la résistance interne RS de la source de tension.

Microchip recommande donc d'avoir une **résistance interne de source inférieure à 10 kiloohms**.

Dans ce cas, **19,7 µs** suffisent à charger suffisamment le condensateur **(> 99,95 % de V).**

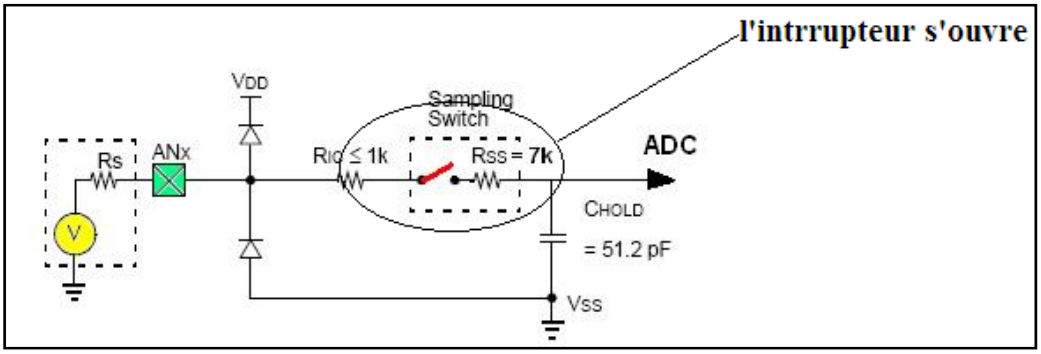
➔ Etape 5- Lancement de la phase de conversion

Une fois la phase d'acquisition terminée (**19,7 µs** environ), vous pouvez lancer la phase de conversion en mettant à 1 le bit GO du registre **ADCON0** :

bcf STATUS, RP0 ; passage en banque 0

bsf ADCON0, GO ; GO = 1

La conversion dure 12 x T AD soit **19,2 µs** dans le meilleur des cas **(T AD = 1,6 µs).**



Pendant la phase de conversion, l'interrupteur électronique est ouvert.

Le condensateur de maintien C HOLD conserve alors à ses bornes la tension V.

Cette tension est constante (aux fuites près) pendant toute la phase de conversion.

➔ Etape 6- Attente de la fin de la phase de conversion

A la fin de la conversion, **le drapeau ADIF de l'interruption du convertisseur ADC est automatiquement mis à 1.** Si vous utilisez cette interruption, le programme saute immédiatement vers la routine d'interruption.

Si vous n'utilisez pas la technique des interruptions, il faut tester le bit GO du registre **ADCON0**.

En effet, **le bit GO est automatiquement remis à 0 en fin de conversion**.

➔ Etape 7 - Lecture du résultat de la conversion

On peut maintenant lire et traiter le résultat de la conversion (registres **ADRESH** et **ADRESL**)

➔ Etape 8 - Attente avant une nouvelle acquisition

Microchip indique qu'il faut attendre 2 x T AD (3,2 µs) entre la fin de la phase de conversion et un nouveau cycle d'acquisition.

En interne, après 2 x T AD, l'interrupteur électronique de l'échantillonneur-bloqueur est fermé, et le processus d'acquisition commence :

➔ Etape 9 - Nouvelle acquisition

On saute à l'**étape 3** (pour un changement de canal). Autrement, on recommence à l'**étape 4**.

2.3.Remarques et conseils :

2.3.1. Calcul de la période d'échantillonnage (TE)

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MEMOIRES

19,7 µs (phase d'acquisition)

+ 12 x T AD = 19,2 µs (phase de conversion)

+ 2 x T AD = 3,2 µs (temps mort)

TE = 42,1 µs (avec T AD = 1,6 µs)

2.3.2. Echantillonnage de signaux lents

Le convertisseur du PIC 16F88 convient pour l'acquisition de signaux lents (par exemple, suivi de la température ambiante d'une salle à partir d'un capteur LM335).

On pensera à intercaler entre le capteur et le canal d'entrée de l'ADC du PIC, un **filtre anti-repliement** (filtre analogique passe-bas de fréquence de coupure FE / 2).

Il est également conseillé d'effectuer un **filtrage numérique** (passe-bas) de façon à enlever le maximum de bruit (en particulier le bruit de ronflement du secteur 50 Hz). Ce filtrage numérique est réalisé de manière logicielle par le PIC (pendant la phase de traitement du résultat de la conversion).

2.3.3. Fréquence d'échantillonnage et nombre de canaux

Supposons que le module ADC travaille avec une fréquence d'échantillonnage de F E = 8 kHz.

On obtient ainsi 8000 échantillons par seconde. Si n canaux sont utilisés, cela fait 8000/n échantillons par seconde et par canal. Ainsi, avec 7 canaux, chaque canal est échantillonné à seulement **F E / n** = 1,14 kHz. Ce n'est pas gênant pour l'acquisition de signaux lents (on pourra connecter 7 capteurs de température), mais c'est rédhibitoire pour des signaux rapides (donc pas de son stéréo car cela fait 2 canaux au lieu d'un seul pour un son mono.

3. Le filtrage numérique

Un filtrage numérique (passe-bas) est effectué de façon à enlever le maximum de bruit (en particulier le bruit de ronflement du secteur 50 Hz). Ce filtrage numérique est réalisé de manière logicielle par le microcontrôleur PIC 16F88 (pendant la phase de traitement du résultat de la conversion de l'ADC).

Caractéristique du filtre numérique :

- Filtre numérique non récursif à moyenne glissante (sur 256 échantillons)
- Période d'échantillonnage nominale : Te = 390,625 µs
- Fréquence d'échantillonnage nominale : $F_e = 1 / T_e = 2{,}560\ 000$ kHz

Le filtre numérique calcule la moyenne arithmétique sur 256 échantillons (filtre numérique à moyenne glissante).

- Choix de la fréquence d'échantillonnage du filtre :

La fréquence d'acquisition est de 1 Hz maximum par voie.

Comme il y a 7 voies, cela laisse 140 ms entre deux acquisitions. Il faut que **nTe** soit un multiple entier de la période du secteur (20 ms), tout en restant inférieur à 140 ms :

On prendra : **nTe = 100 ms** (soit 5 fois la période du secteur).

On a intérêt à choisir n le plus grand possible.

Te est limitée par la rapidité du convertisseur ADC interne du microcontrôleur PIC 16F88.

Dans notre cas, la durée de conversion est de 42 µs.

On prendra : **n = 256** (puissance de 2)

D'où :

Te = 100 ms / 256 = **390,625 µs** (ce qui est bien supérieur à 42 µs)

F_e = 2560,000 Hz

L'horloge du microcontrôleur PIC 16F88 est cadencée par un quartz de fréquence nominale 20 MHz : 1 cycle correspond donc à 0,2 µs.

390,625 µs = 1953,125 cycles

On arrondit à 1953 cycles :

Te = 390,6 µs

Fe = 2560,16 Hz

La tolérance sur le quartz est ± 100 ppm.

Finalement : **Fe = (2560,16 ± 0,26) Hz**

La tolérance sur la fréquence d'échantillonnage peut être négligée.

Par la suite, on retiendra la valeur de 2560 Hz

- Fréquences rejetées par le filtre numérique :

  f = k ´ 2560 / 256 avec k entier

  f = k ´ 10 Hz

Tous les multiples entiers de 10 Hz sont complètement filtrés par le filtre numérique :

10, 20, 30, 40, **50**, 60, …. **100**, …, **150** Hz …

On élimine ainsi les harmoniques du secteur (50 Hz).

N.B. Les multiples de 60 Hz sont également éliminés (on rappelle que 60 Hz est la fréquence du secteur aux USA et en Grande Bretagne).

nTe = 100 ms

= 5 fois la période du secteur (50 Hz en France)

= 6 fois la période du secteur (60 Hz aux USA).

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

- Réponse en fréquence du filtre numérique

Fe = 2560 Hz **n = 256 échantillons**

✸ **Zoom de la plage 0 à 250 Hz :**

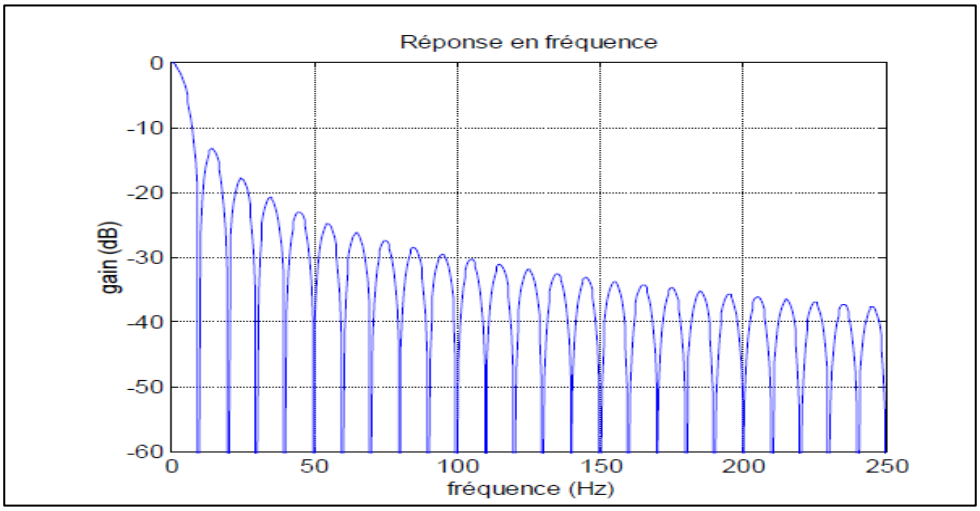


Figure 3.6 : Réponse en fréquence du filtre numérique la plage 0 à 250 Hz

✸ **Plage 0 à 640 Hz :**

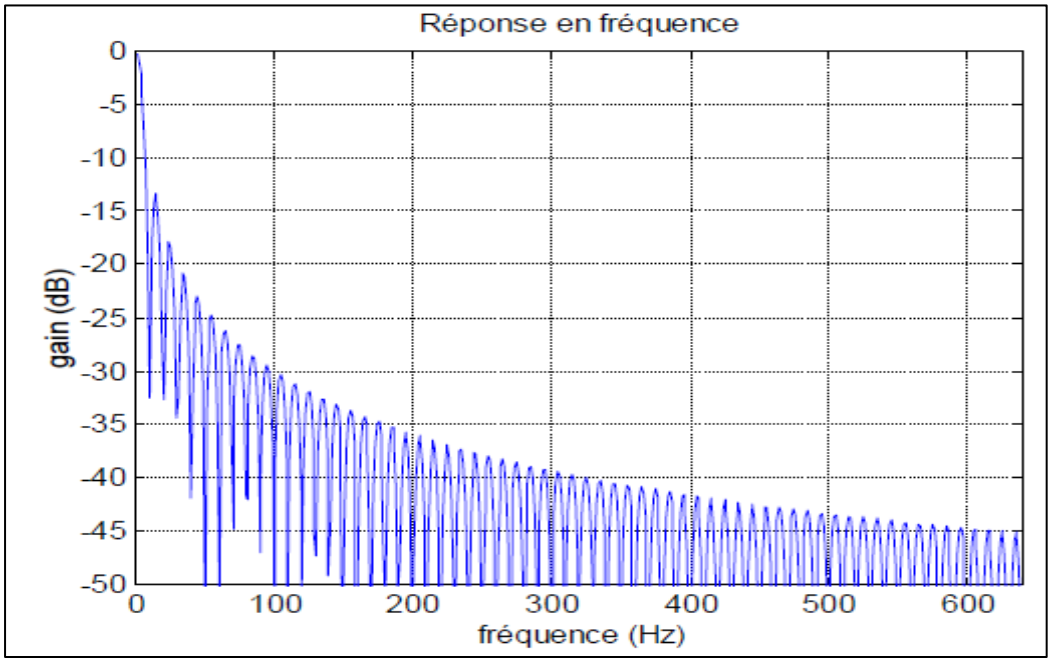


Figure 3.7 : Réponse en fréquence du filtre numérique la plage 0 à 640 Hz

✸ **Plage 640 Hz à F_e / 2 = 1280 Hz :**

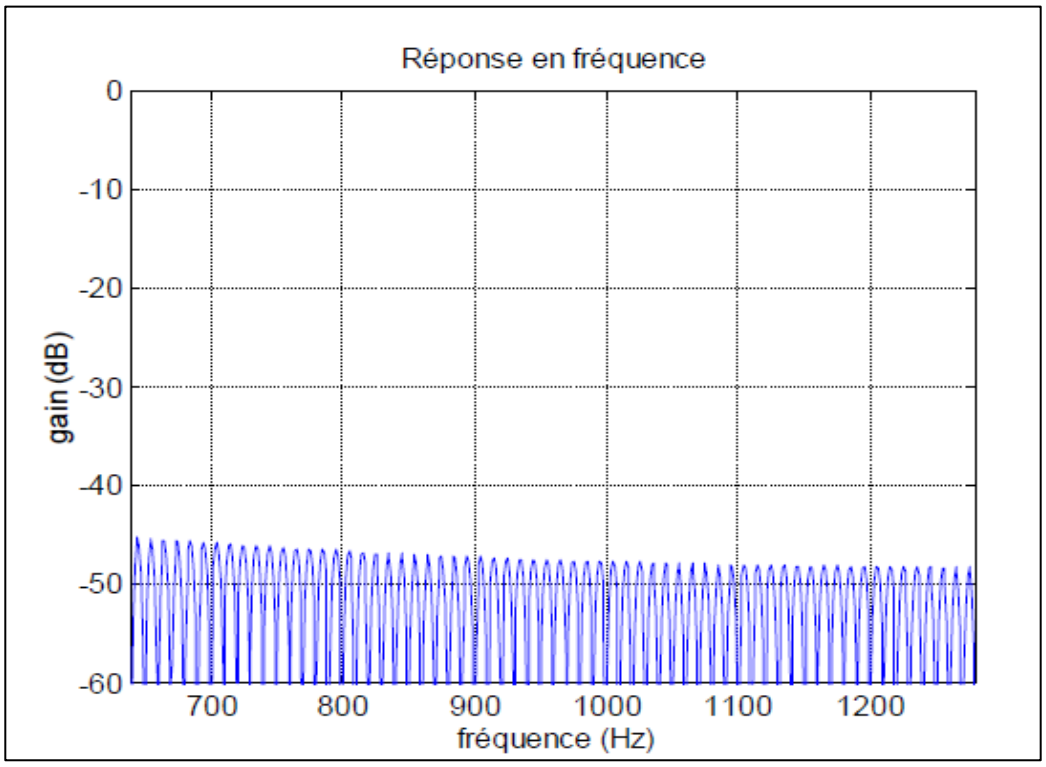


Figure 3.8 : Réponse en fréquence du filtre numérique la plage 640 Hz à Fe / 2 = 1280 Hz

IV. LA LIAISON RS232

La communication de cette carte avec le PC se fait à travers une connexion série (DB9), reliée au microcontrôleur par ses broches 2 et 3 (RX, TX) via un circuitMAX232 dont le rôle est l'adaptation des signaux TTL/CMOS.

1. Principe d'adaptation PIC- RS232

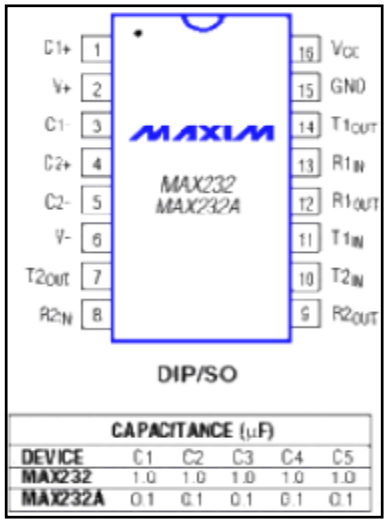
Passons maintenant au principe d'adaptation entre le PIC et le port serie Rs232.

Le PIC utilise les niveaux 0V et 5V pour définir respectivement des signaux « 0 » et « 1 ». La norme RS232 définit des niveaux de +12V et –12V pour établir ces mêmes niveaux.

Nous aurons donc besoin d'un circuit charge de convertir les niveaux des signaux entre PIC et PC. Le pin TX du PIC émettra en 0V/5V et sera convertie en +12V/-12V vers notre PC. La ligne RX du PIC recevra les signaux en provenance du PC, signaux qui seront converti du +12V/-12V en 0V/5V par notre circuit de pilotage du bus. Notons que la liaison étant full-duplex, émission et réception sont croisées, chaque fil ne transitant l'information que dans un seul sens.

Nous utiliserons le célèbre circuit MAX232 pour effectuer cette adaptation de niveaux. Ce circuit contient un double convertisseur à double direction. Autrement dit, il dispose de :

- 2 blocs, dénommes T1 et T2, qui convertissent les niveaux entres en 0V/5V en signaux sortis sous +12V/-12V. En réalité, on n'a pas tout a fait +12V et -12V, mais plutôt de l'ordre de +8,5V/-8,5V ce qui reste dans la norme RS232.
- 2 blocs, dénommes R1 et R2, qui convertissent les niveaux entres en +12V/-12V en signaux sortis sous 0V/5V.



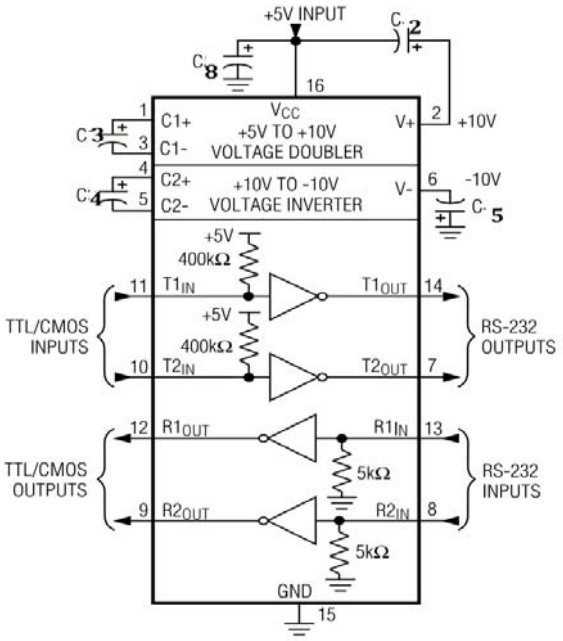


Figure 3.9 : Circuit intégré MAX232

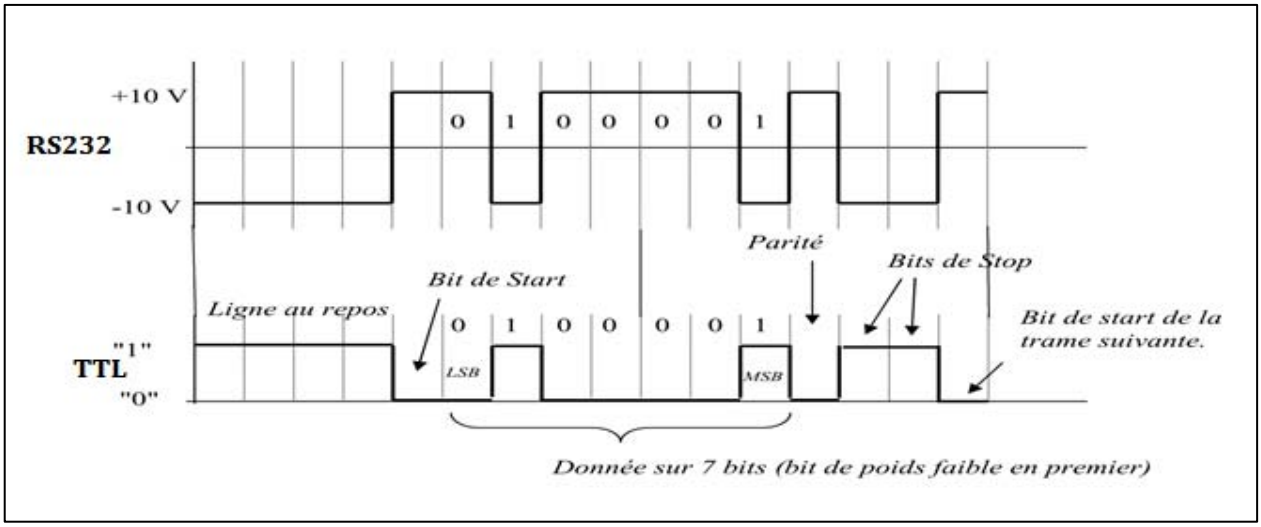


Figure 3.10 : conversion TTL – RS232

2. Protocole de communication entre l'ordinateur et la carte :

Seul l'ordinateur prend la parole (maître).

Le microcontrôleur PIC 16F88 se contente de répondre (esclave).

L'ordinateur envoie 1 octet (via la liaison RS232) toute les 140 ms.

Cet octet contient le numéro du canal à échantillonner :

- canal 0 -> 0x00
- canal 1 -> 0x01
- canal 2 -> 0x02
- canal 3 -> 0x03
- canal 4 -> 0x04
- canal 5 -> 0x05
- canal 6 -> 0x06

Le processus est cyclique : voie 0 puis voie 1 ... puis voie 6 puis voie 0 ...

Chaque voie est ainsi échantillonnée toutes les 7 x 140 = 980 ms (d'où une fréquence d'acquisition d'environ 1 Hz). Une fois l'octet reçu, le PIC 16F88 sélectionne le canal indiqué, et lance une série de 256 conversions (une toute les 390,6 µs).Cela prend : 256 x 390,6 µs = 100 ms

Le microcontrôleur PIC effectue la moyenne des 256 conversions (filtrage numérique).

Sachant que la résolution du convertisseur ADC est 10 bits, la somme nécessite un nombre de 18 bits.

La moyenne correspond à la somme avec la virgule décalée de 8 bits vers la gauche.

Par exemple :

Somme brute (18 bits) : 10 10010010 01000101 = 168 517 (en décimal). On décale la virgule de 8 positions : 10 10010010, 01000101

Partie entière (10 bits) : 10 10010010 = 658 (en décimal)

01000101 = 1/4 + 1/64 + 1/256 = 0,26953125

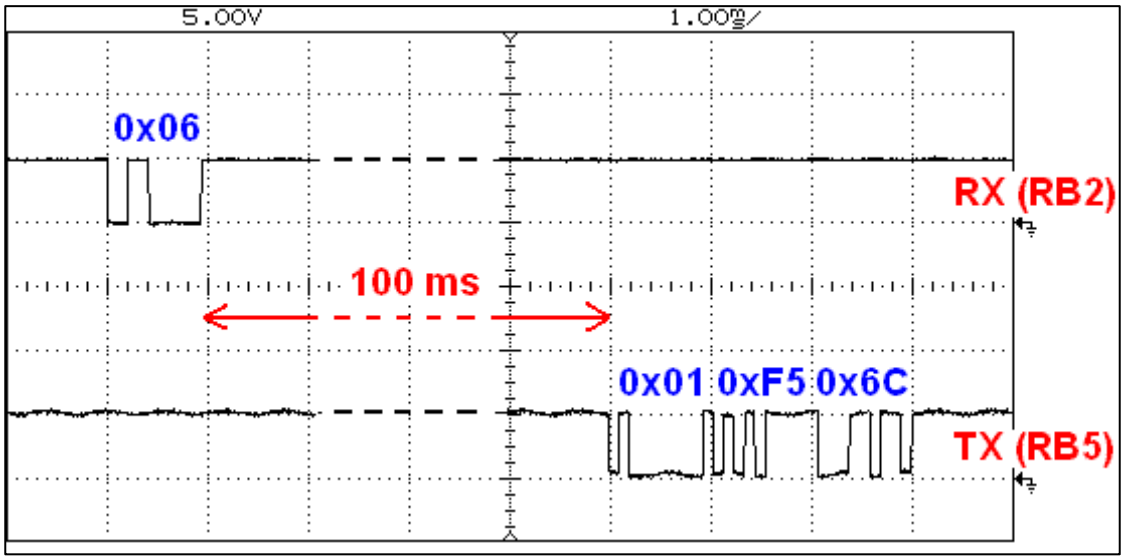
En définitive : 168 517 / 256 = 658,26953125

Notez que la résolution est améliorée : on passe de 10 bits à 18 bits.

On pratique, on restera prudent, et on considérera que la résolution est d'au moins 12 bits (soit une résolution de l'ordre de 5 V / 4096 ≈ 1 mV). La moyenne est envoyée telle quelle vers l'ordinateur sous la forme de 3 octets :

- 1er octet : (000000 D9 D8)
- 2ème octet : (D7 ... D0)
- 3ème octet : (D-1 ... D-8)

➢ **Exemple d'oscillogramme :**



L'ordinateur envoie vers le PIC un octet : 0x06 <=> voie 6

100 ms plus tard, le PIC renvoie 3 octets :

0x01 0xF5 0x6C <=> 0x01F56C <=> 128 364 (en décimal)

128 364 / 256 = 501,42 <=> (501,42 / 1023) x 5 volts = **2,451 volts**

40 ms plus tard (140 - 100 = 40 ms), l'ordinateur envoie un nouvel octet (0x00) pour demander l'échantillonnage de la voie 0, et le cycle recommence ...

V. L'APPLICATION *ADC108.exe* POUR WINDOWS

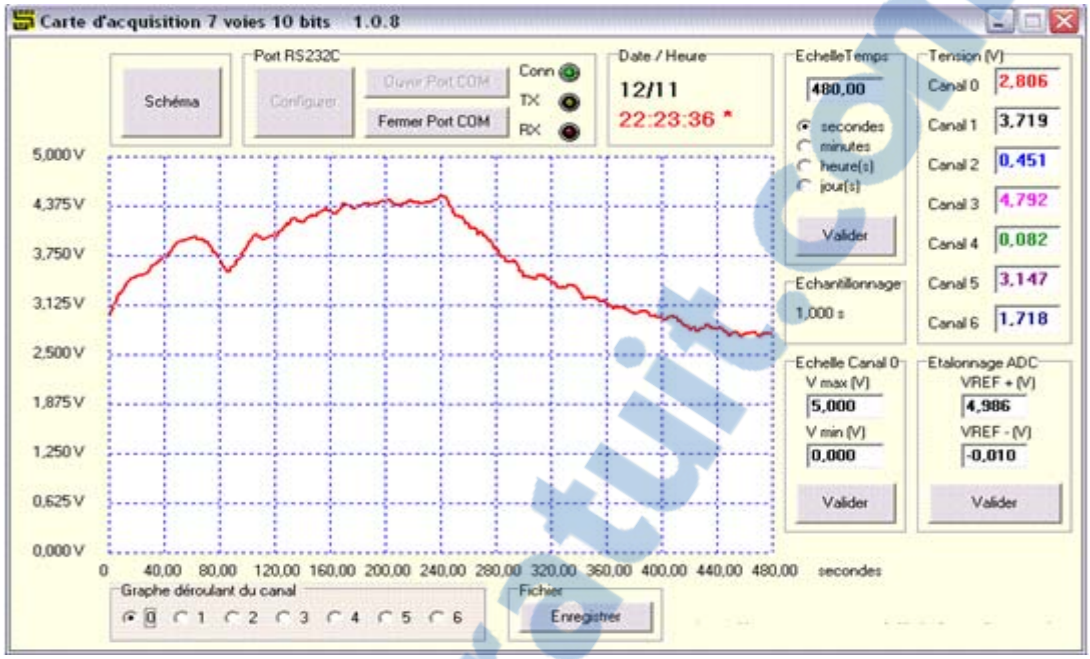


Figure 3.11 : l'application *ADC108.exe*

Cette application a été créée avec l'environnement C++ Borland Builder 5.0

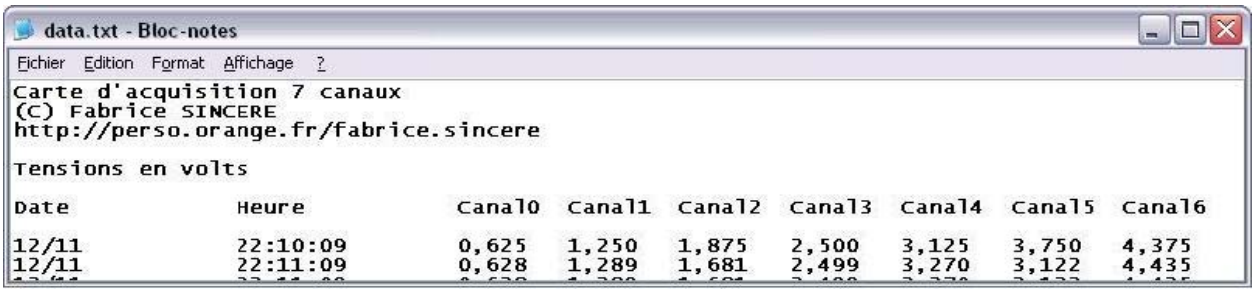
1. Procédure d'utilisation

- Brancher le câble "null-modem" entre l'ordinateur et la carte (hors tension)
- Mettre la carte sous tension
- Ouvrir l'application
- Configurer les paramètres de la liaison RS232C :
  - 9600 bauds/s
  - 8 bits de données
  - Pas de bit de parité
  - 1 bit de STOP
  - Pas de contrôle de flux
- Ouvrir le port

La tension actuelle de chaque voie doit s'afficher (avec une mise à jour toutes les 980 ms).

Il y a la possibilité de sauvegarder les données dans un fichier texte :

Rapport-Gratuit.com



data.txt - Bloc-notes

Fichier Edition Format Affichage ?

```
Carte d'acquisition 7 canaux
(C) Fabrice SINCERE
http://perso.orange.fr/fabrice.sincere

Tensions en volts

Date            Heure           Canal0  Canal1  Canal2  Canal3  Canal4  Canal5  Canal6

12/11           22:10:09        0,625   1,250   1,875   2,500   3,125   3,750   4,375
12/11           22:11:09        0,628   1,289   1,681   2,499   3,270   3,122   4,435
```

Ce fichier pourra ensuite être exploité avec LibreOffice, Excel, Matlab ...

VI. LITE DE MATERIEL

- microcontrôleur PIC 16F88 (boîtier PDIP)
- 1 circuit intégré MAX233A (interface RS232C <-> TTL/CMOS)
- 1 connecteur SubD 9 broches mâle
- 7 résistances 4,7 k
- 7 condensateurs de 2200 nF (ou 1000 nF)
- 1 quartz 20 MHz
- 1 condensateur électrochimique de 100 µF (filtrage)
- 1 condensateur électrochimique de 10 µF (filtrage)
- 1 condensateur électrochimique de 1 µF (filtrage)
- 4 condensateurs de 100 nF (filtrage)
- 2 condensateurs de 22 pF
- 1 source d'alimentation continue +12 V (ou une pile de 9 V)
- 1 régulateur 7805 (boîtier TO220)

1. la carte d'alimentation :

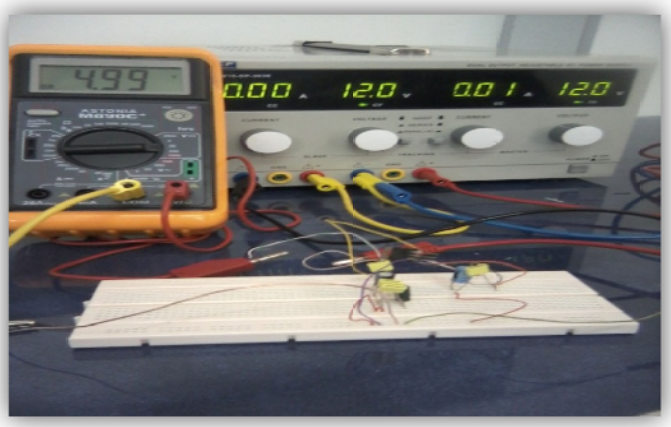


Figure 3.12 : la réalisation de la carte sur plaque d'essai

Schéma électrique :

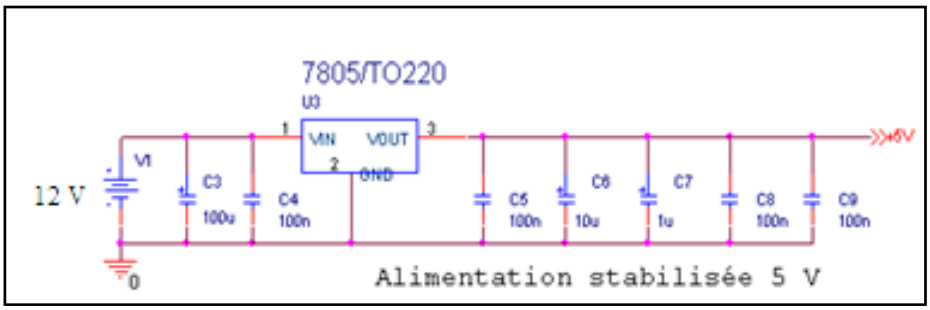


Figure 3.13 : schéma électrique de la carte d'alimentation

Notre carte doit être alimentée par une alimentation stabilisée qui fournit par 5V a la sortie pour alimenter notre PIC.

La carte d'alimentation réalisée contient :

Un régulateur de tension 7805 : qui fournit une sortie 5v pour une tension d'entrée de 12v (on peut utiliser une pile de 9V). Les condensateurs C3 et C4 de (100nF) permettent de filtrer la tension d'entrée pour le régulateur, et les condensateurs C5 (100nF), C6 (10µF), C7 (1µF), C8 (100nF), C9 (100nF) permettent de lisser la sortie du régulateur pour la charge.

VII. SIMULATION AVEC ISIS-PROTEUS

Proteus est une suite logicielle de CAO électronique éditée par la société **Labcenter Electronics**. **ISIS** est la composante de Proteus qui permet la création de schémas et la simulation électrique. La grande force de **ISIS** est de pouvoir simuler le comportement d'un microcontrôleur (PIC, Atmel, 8051, ARM, HC11...) et de son interaction avec les composants qui l'entourent.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

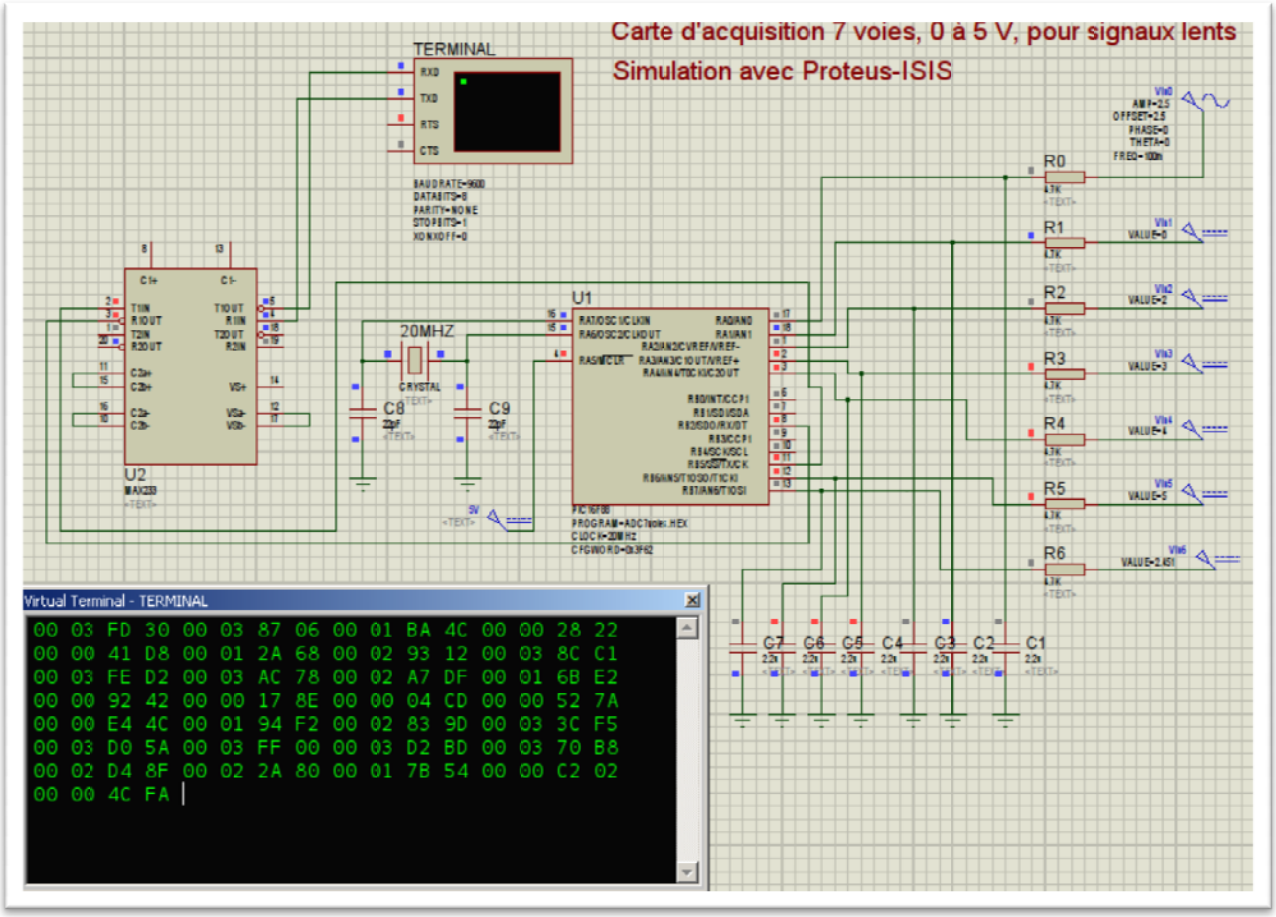


Figure 3.14 : schéma de simulation avec ISIS

Cette simulation nous permet de tester le circuit électrique de la carte au coté logiciel avant de passer à la pratique pour éliminer toutes les erreurs

VIII. CONCLUSION

Donc dans ce chapitre on a vue les différents éléments constituants la carte d'acquisition réalisé, leurs fonctionnement ainsi que le principe de la transmission série et celui de l'adaptation en ligne.

Nous verrons dans le chapitre suivant la partie pratique et les essais obtenues ainsi que les résultats de conversion sur l'application windows.

Chapitre IV

L'application de la carte d'acquisition 7 voies

I. INTRODUCTION

Dans ce chapitre on va entamer a la partie pratique c a d la réalisation de la carte d`acquisition a 7 voies analogique dans la plaque d`essai puis la maquette et faire l`acquisition des signaux converties des différents canaux dans un ordinateur XP par une application ADC.exe et enfin la mémorisation de ces signaux.

II. ETUDE PRATIQUE

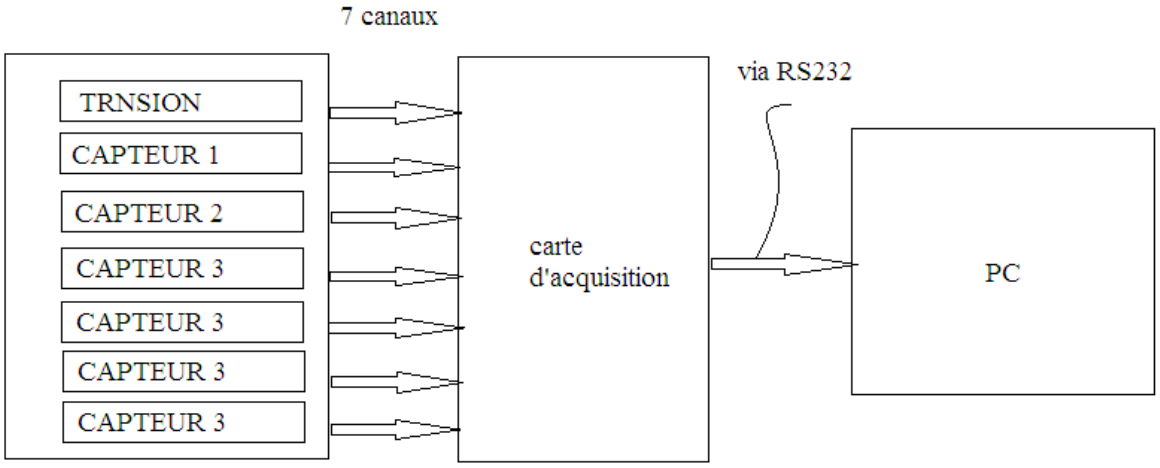


Figure 4.1 : Schéma de principe du système pour l`acquisition des signaux lents

A l`entrée on a plusieurs capteurs (exp : température, pression…etc.)

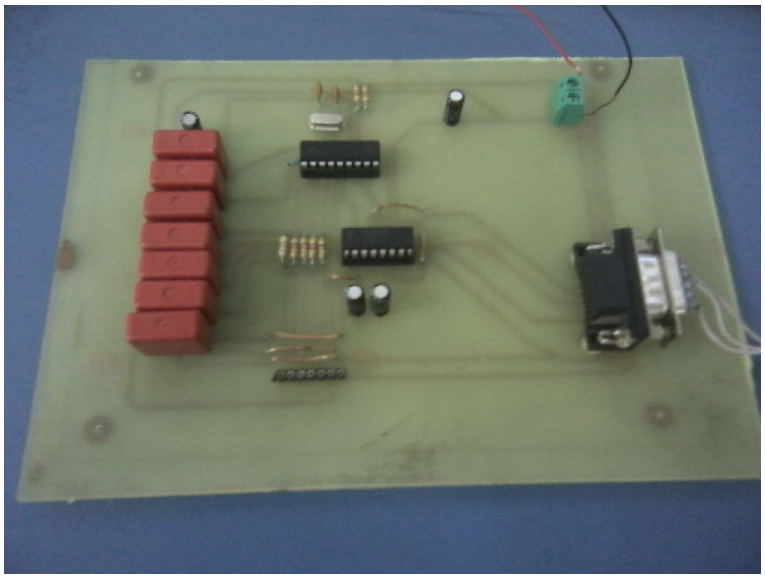
La température et la pression représente des signaux lents, chaque capteur acquis des données séparé par un intervalle import dans le temps (40s) entre un données et un autre

Chaque capteur peut donner une information différentes (ex : des relevés de température dans des différentes parties du corps ou plusieurs patient

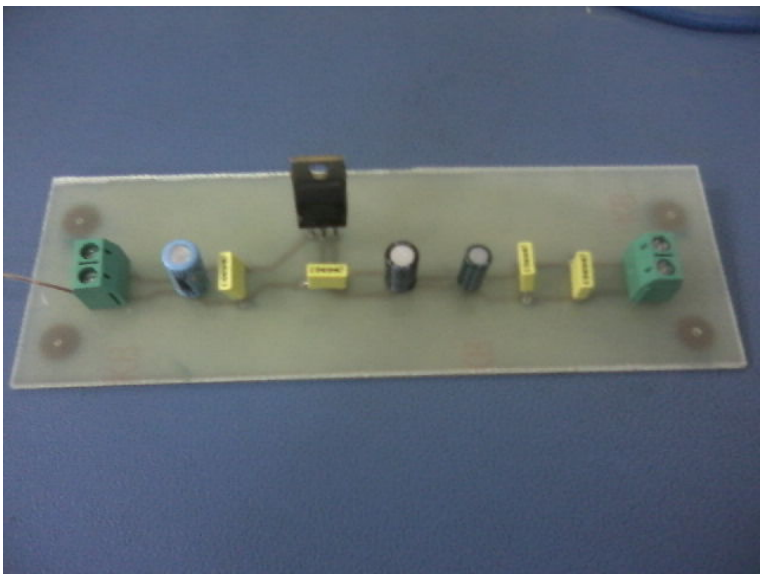
La carte d`acquisition réalisée nous permettre d`afficher sur PC via RS232 différente résultats obtenues (différentes température).

Dans notre cas on a utilisé à la place des capteurs des tensions continues car on ne s`intéresse pas à l`étude des capteurs dans notre projets

III. La carte d`acquisition a 7voies analogiques pour signaux lents



IV. Alimentation stabilisée



V. Application

On suit les étapes suivantes :

- On Branche le câble "null-modem" entre l'ordinateur et la carte (hors tension)
- On met la carte sous tension par (5v)
- On introduit une tension continue de valeurs maximale ne dépasse pas de la tension d`alimentation 5v dans un canal choisit.
- Ouvrir l`application ADC.exe.
- Faire la configuration des paramètres de la liaison RS 232.
- Ouvrir le port COM

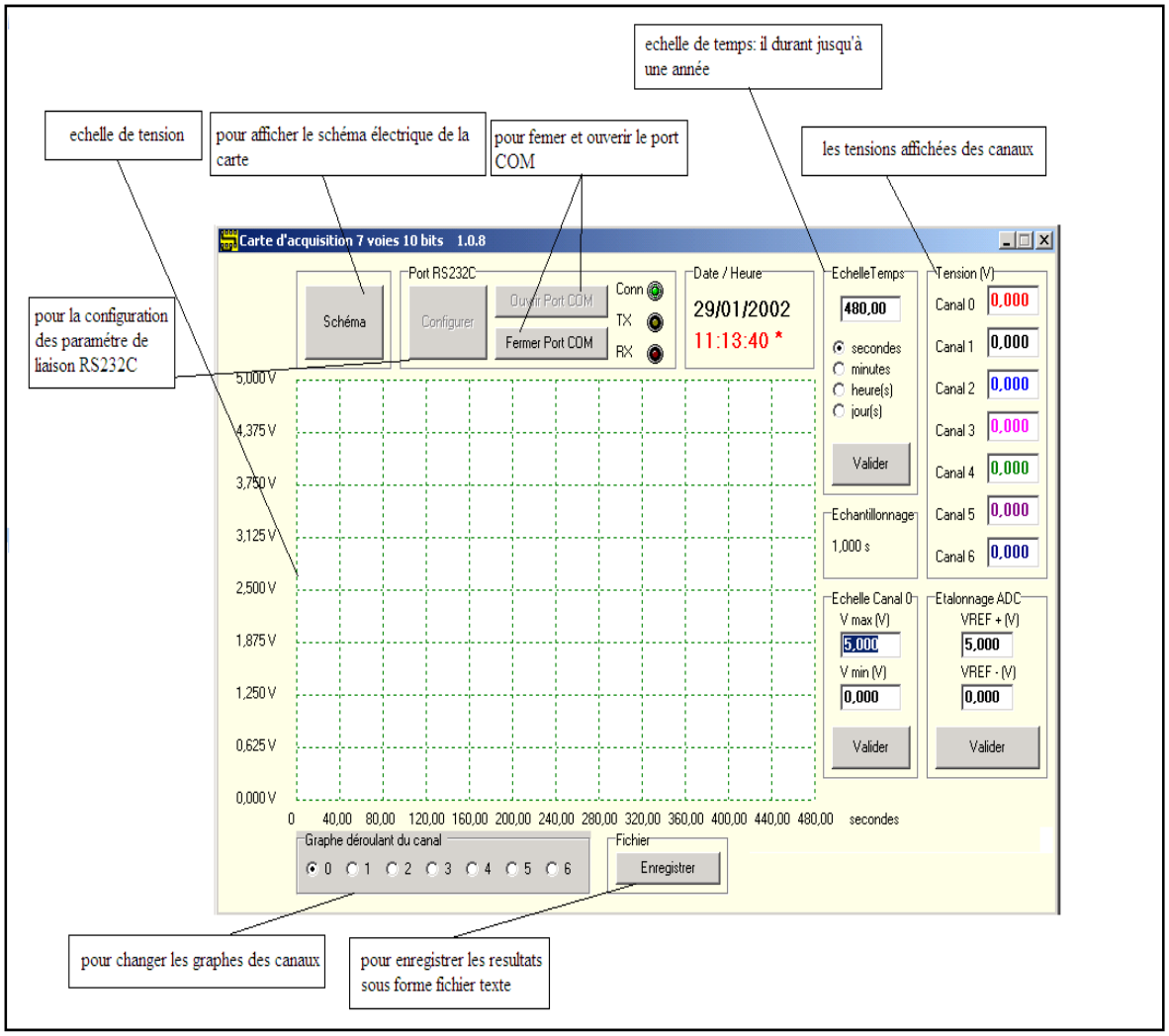


Figure 4.2 : Schéma explicatif de l`application ADC108.exe

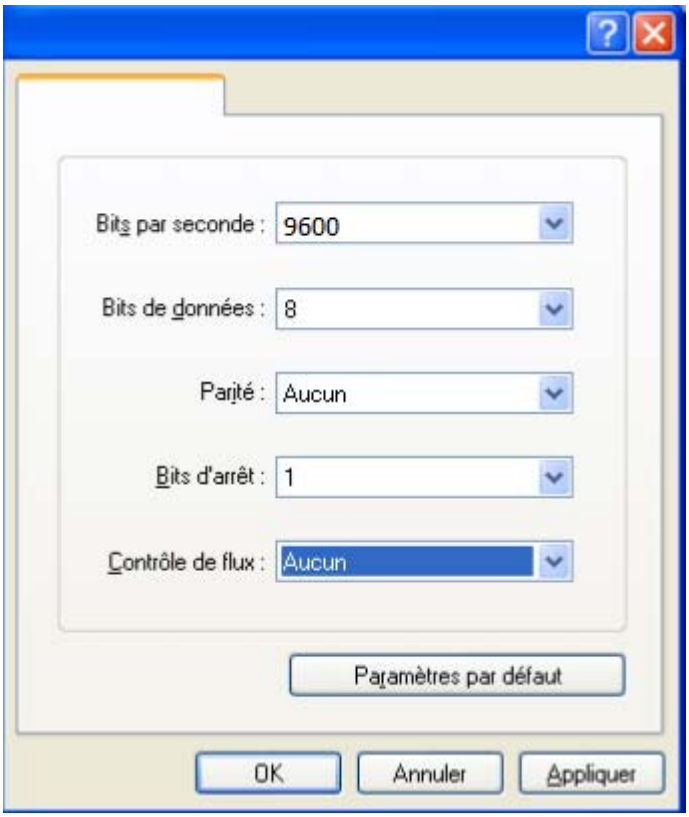
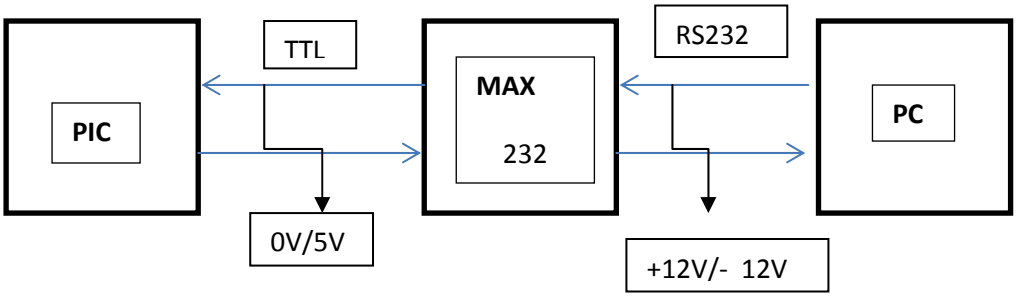


Figure 4.3 : La configuration des paramètres de la liaison RS232

VI. Problématique

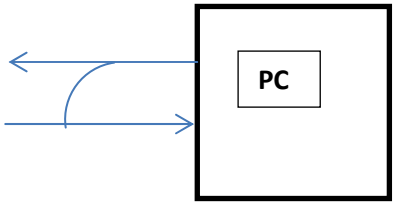
En premier essai de carte on n'aura aucun résultat affiché sur application donc

On a fait des testes pour vérifier le bon fonctionnement de notre carte d`acquisition on a commencé par la liaison RS232puis le MAX232 et PIC16F88 qui est programmer bien en avant et on a constaté que ils n`y a pas des erreurs donc ces composants fonctionnent très bien et finalement on a trouvé qu`n` ils y a pas des condensateurs de filtrage autour du pic donc il faut ajouter deux condensateurs pour construire un filtre analogique pour rendre le signal entre (information entrée) exploitable qui fait élimination du bruits (resaux50hz)



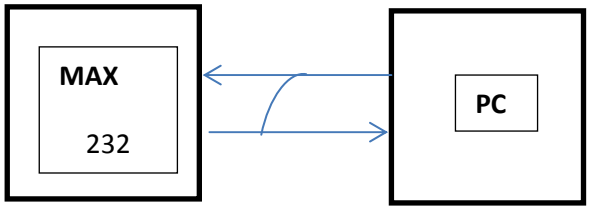
A. Le premier test :

Tous d`abord on va tester la liaison **RS232**



- On enlève le max232 et on court-circuité les deux pattes de max232 qui ont connectées par TX et RX de la liaison RS232 (pattes 14 et 13) et ainsi que les pattes de RS232 (RTS, CTS) et il faut être connecté
- Et on prend les résultats dans le hepert terminale du microordinateur (chaque ordinateur contient un hepert terminal).
- On résulte qu'on envoie un caractère, le hepert terminale affiche le caractère envoyé.

B. Le deuxième teste PIC



- On enlève le pic 16f88 puis on court-circuité les deux pattes (patte8 reçoit les données et patte11envoit les données).
- On constate lorsqu'on envoie une donnée, on reçoit la donnée elle même

C. Le troisième test

Le teste du signal d`entré :

Pour que le signal d`entre analogique soit exploitable (élimination du bruit) à l'entrée du pic 16f88, il faut ajouter un filtre passif qui contient deux condensateurs de 100nf 470uf en parallèle qui est alimenter par 5v.

VII. Résultats et discutions :

Apres l`application de tension continue aux différentes entrées de notre carte d`acquisition les résultats représentées comme suit :

1) Application sur canal0 :

On a appliqué une tension analogique de différents natures (sinusoïdales, rectangulaires , triangulaires) par un générateur de tension et voici les résultats obtenues :

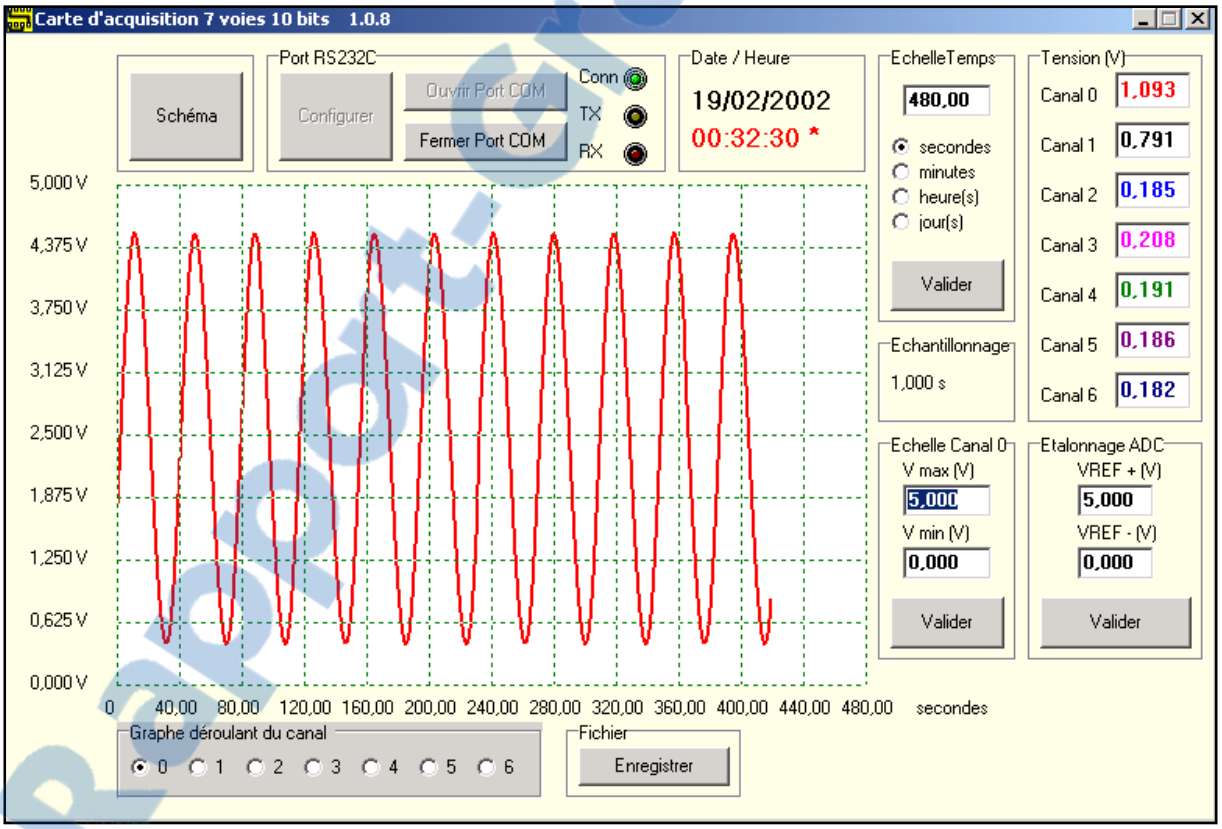


Figure 4.4 : résultat de la conversion 1.093v d'entrée appliquée au canal0

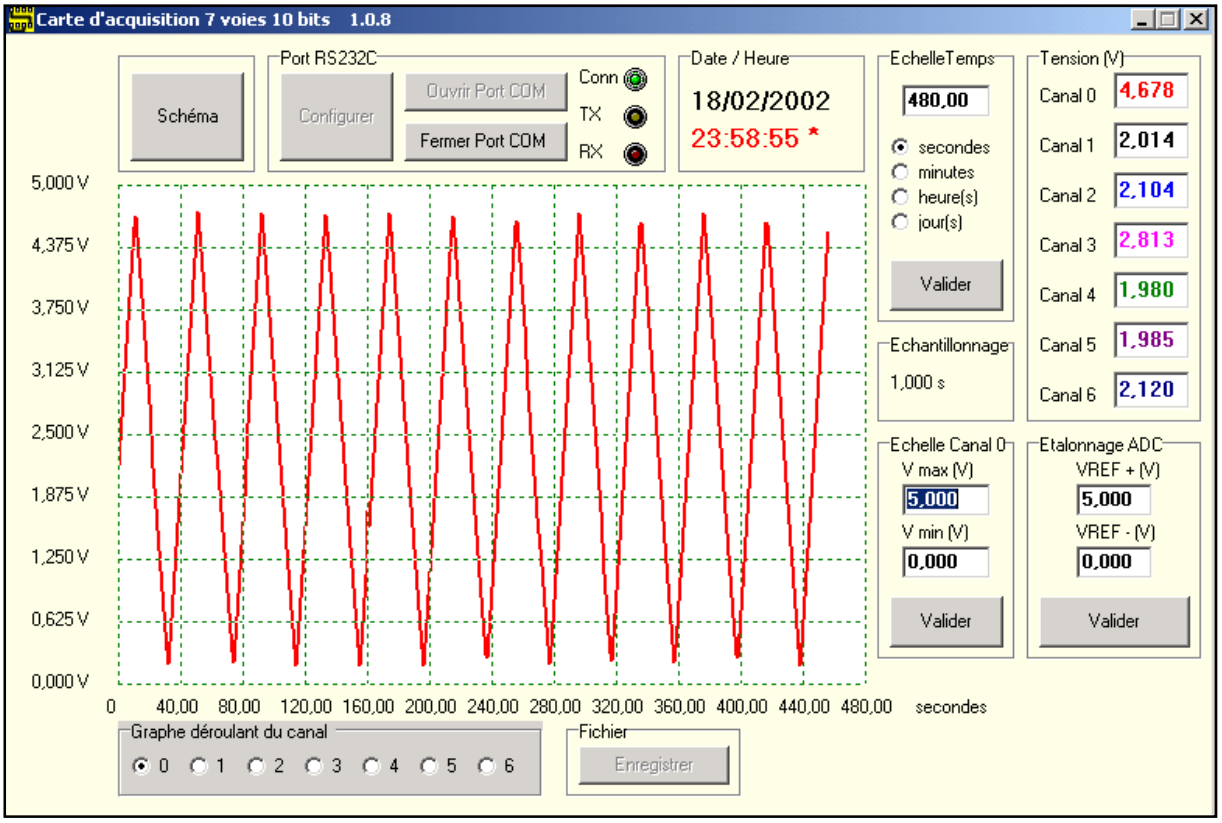


Figure 4.5 : résultat de la conversion 4.678v d'entrée appliquée au canal0

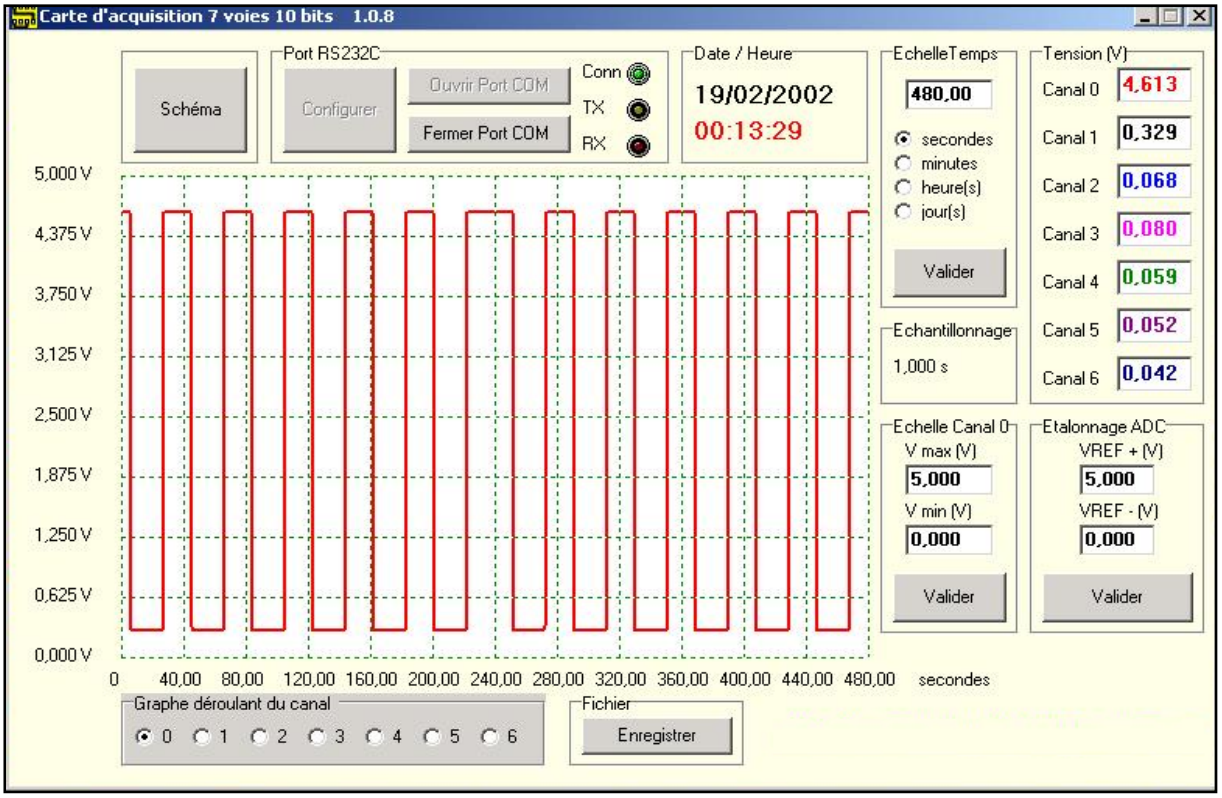


Figure 4.6 : résultat de la conversion 4.613v d'entrée appliquée au canal0

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

2) Application sur canal1 :

On applique une tension continue qui varie de 1.3v de 4.99v à

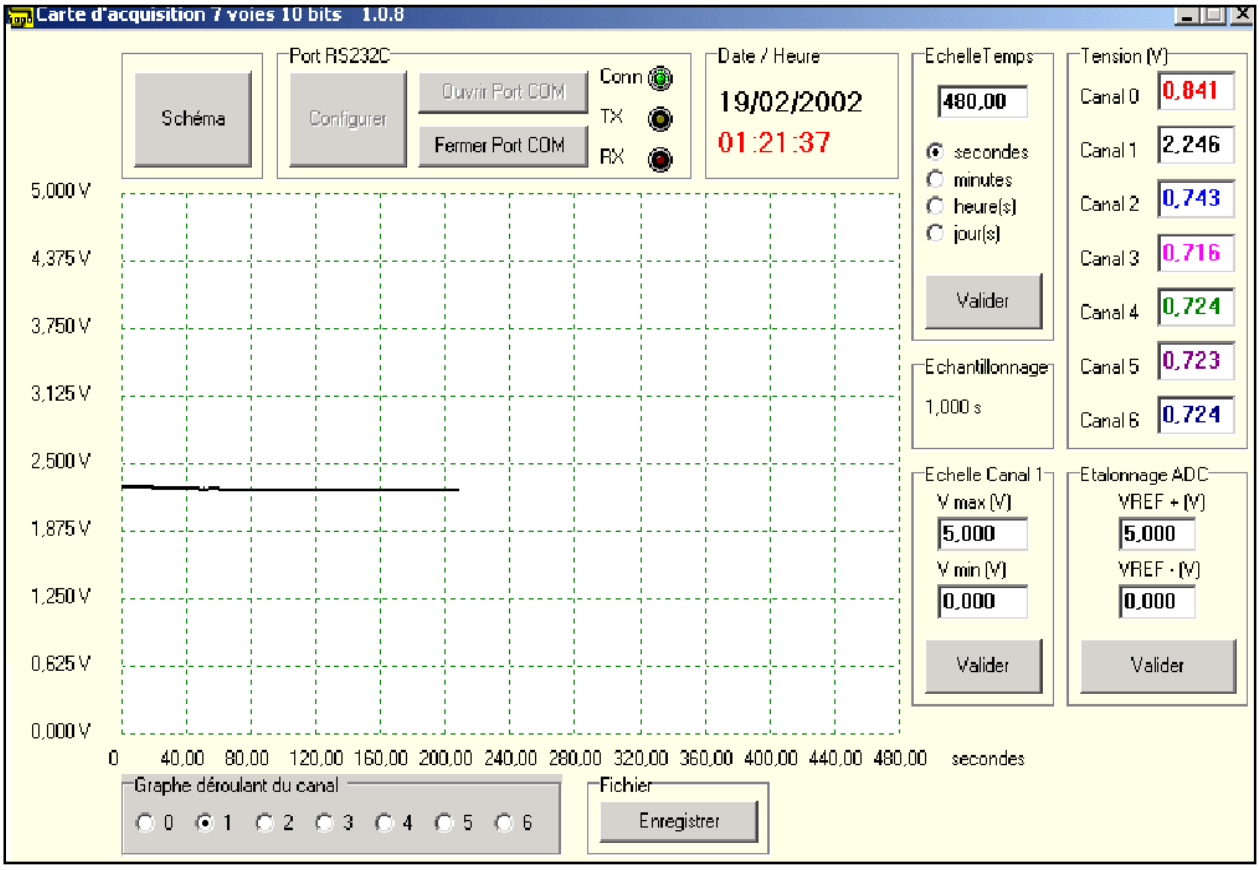


Figure 4.7 : résultat de la conversion 2.246v d'entrée appliquée au canal1

3) Application sur canal2 :

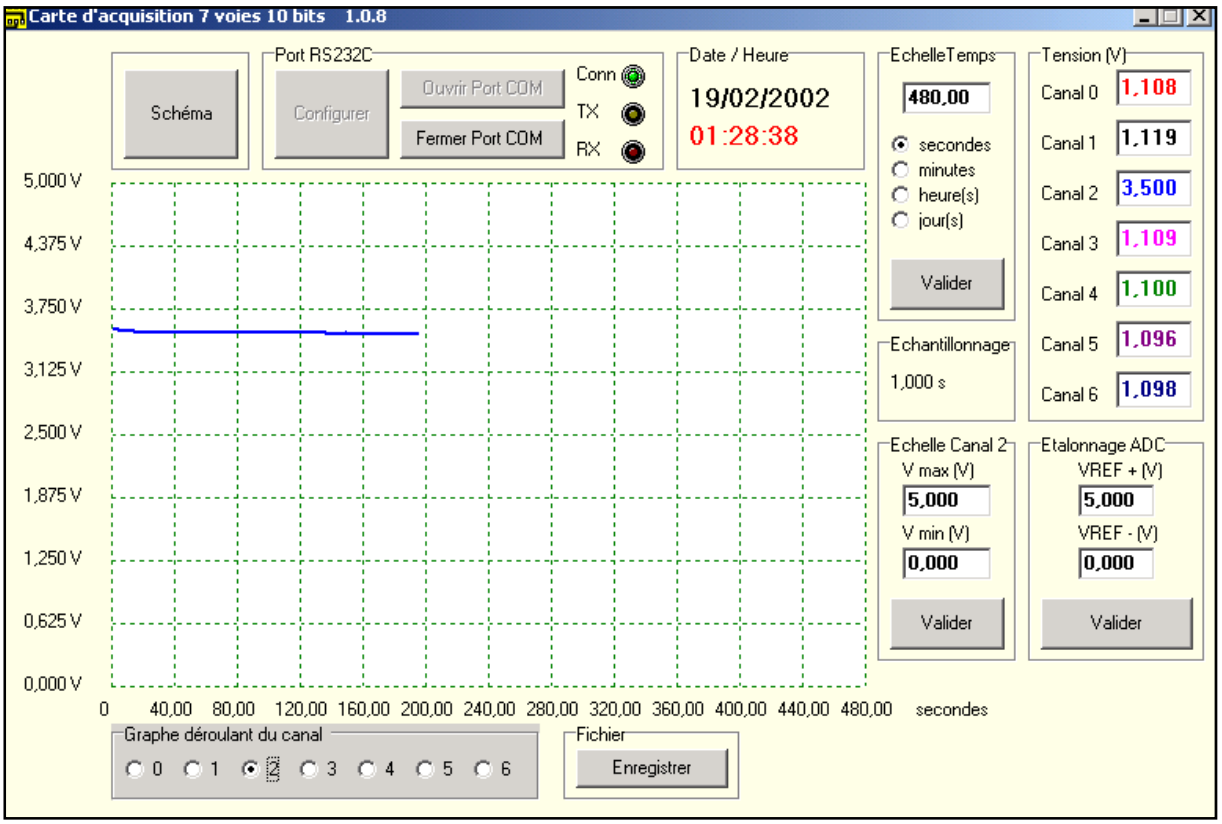


Figure 4.8 : résultat de la conversion 3.500v d'entrée appliquée au canal2

4) Application sur canal3 :

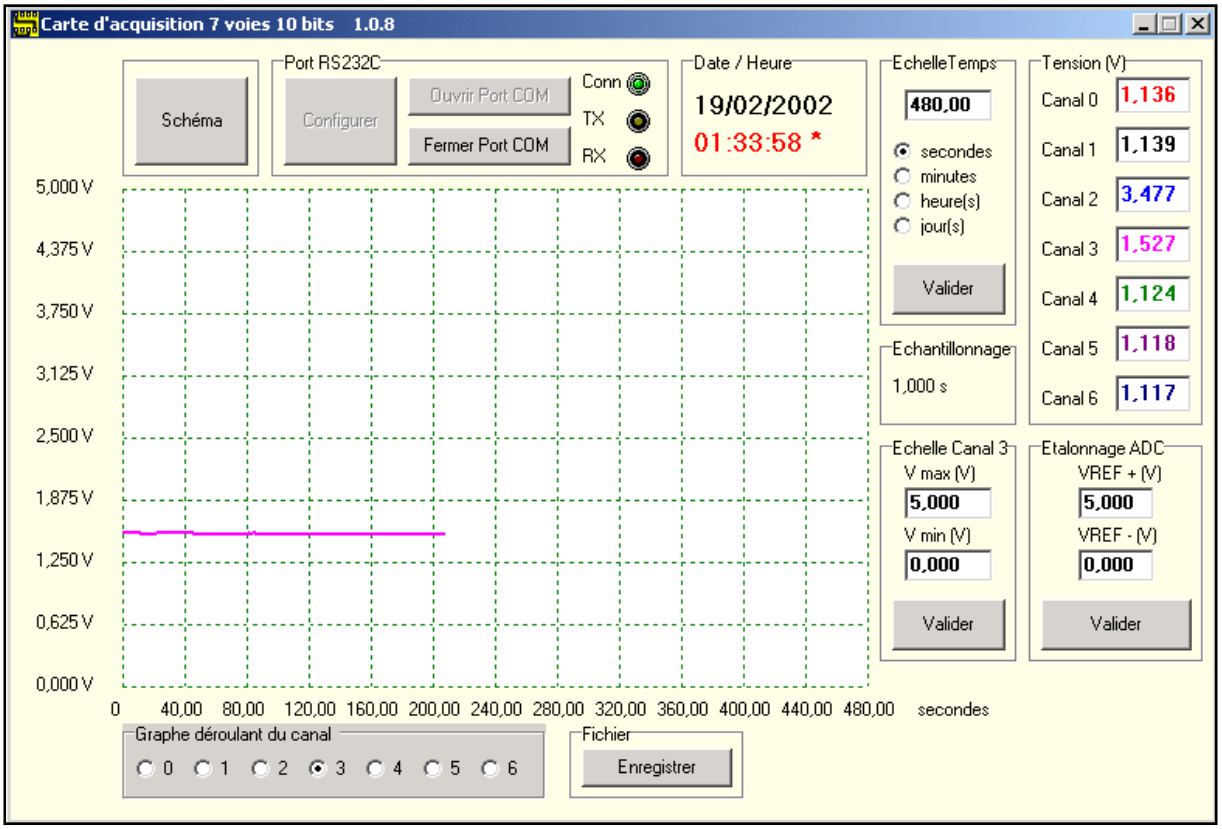


Figure 4.9 : résultat de la conversion 1.527v d'entrée appliquée au canal3

5) Application sur canal4 :

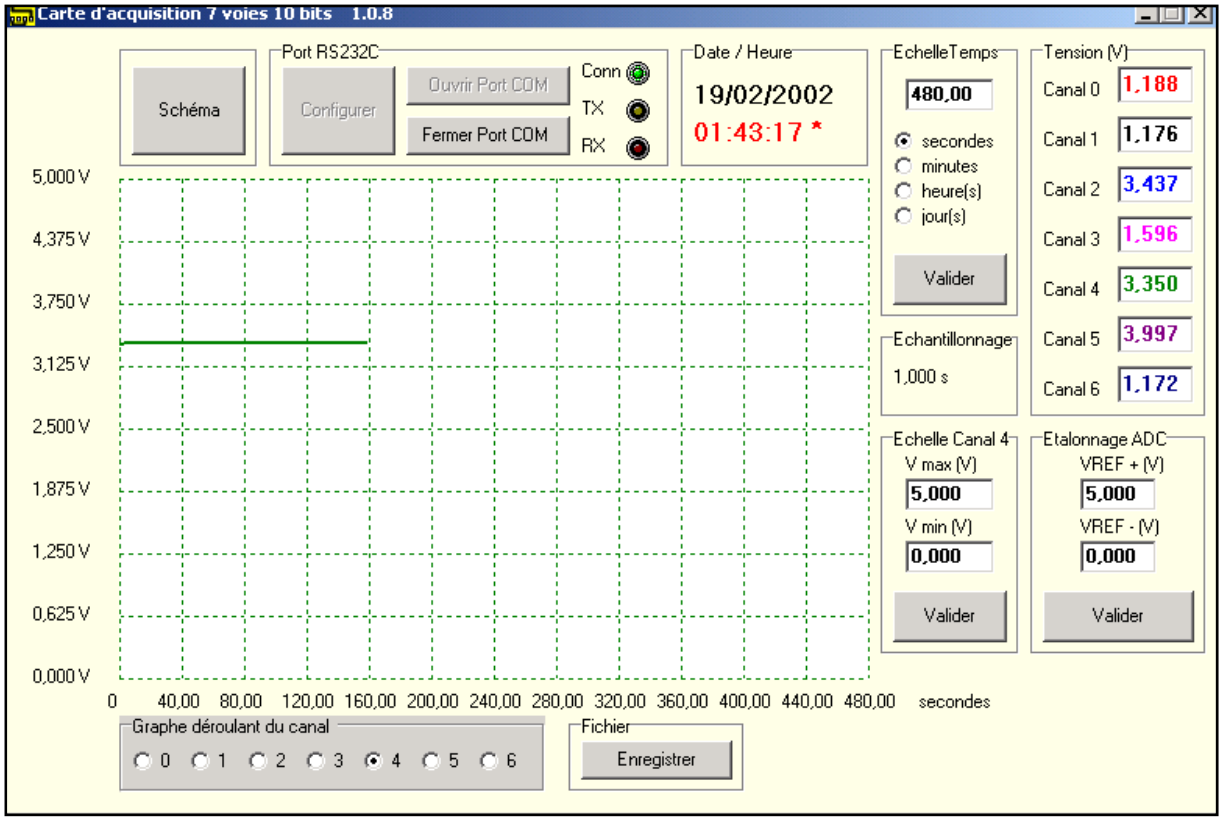


Figure 4.10 : résultat de la conversion 3.350v d'entrée appliquée au canal4

6) Application sur canal6 :

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

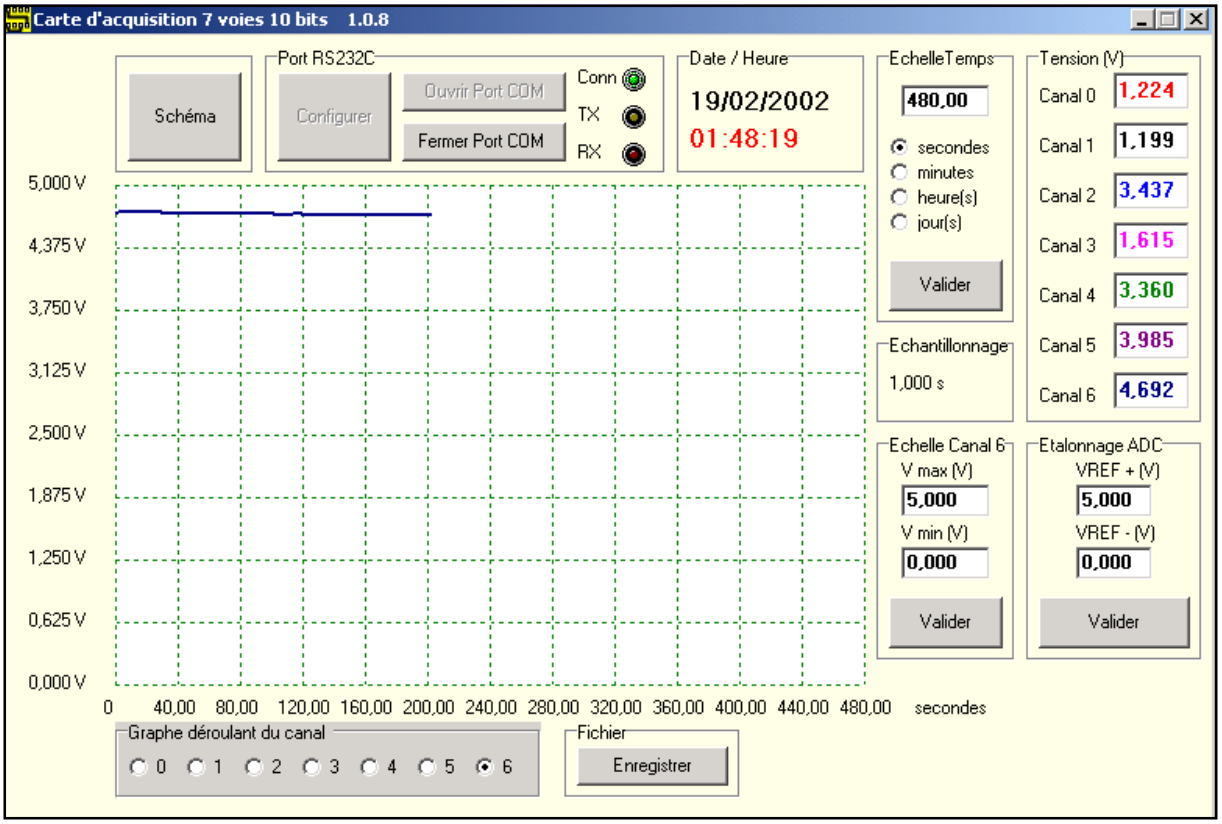


Figure 4.11 : résultat de la conversion 4.692v d'entrée appliquée au canal6

VIII. Conclusion

Apres les résultats obtenues on peut dire que la carte elle donne des meilleures résultats puis que elle arrive a donnée une ligne droite quand on applique une tension continue qui ne dépasse pas 5v a l`entrées de chaque canaux et même lorsque on applique des tensions des différentes natures(sinusoïdales ,rectangulaires ,triangulaires) à partir du GBF générateur de tension au niveaux de différents canaux on obtient des signaux similaires à l`entrée .

Conclusion générale

CONCLUSION GENERALE

Le travail qui nous a été confié dans le cadre de notre mémoire de master en électronique biomédical est un travail de développement technologique ayant pour objectif la réalisation d'une carte d'acquisition pour des signaux lents par exemple le signal de température et de pression atmosphérique est un signal d'évolution lente (de période très supérieure à la seconde). Notre carte est basée sur le PIC 16F88. C'est pour ça l'élaboration de ce modeste travail nous a permis d'accéder à plusieurs domaines. L'utilisation du PIC nous a introduits au domaine de µC, à leurs mises en œuvre et à leur système de développements ainsi que les différentes fonctionnalités et avenages qu'ils offrent.

Ce travail nous a permet aussi d'apprendre le logiciel ISIS pour la simulation.

La problématique de ce travail nous a permet de tester la fonctionnalité de quelques éléments de notre carte comme par exemple le PIC, le MAX232 et le DB9 nous avons fait des testes pour savoir si le problème posé aux cours de notre travail est à cause de mal communication entre le PC et la carte ou bien le PIC ou le MAX232 ne fonctionne pas

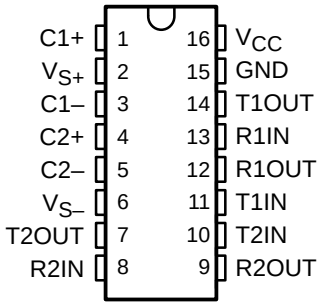
Ce projet nous a permis d'approfondir nos connaissances théoriques et pratique en instrumentation médicale

Annexe

RapportGratuit.com

- **Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28**
- **Operate With Single 5-V Power Supply**
- **Operate Up to 120 kbit/s**
- **Two Drivers and Two Receivers**
- **±30-V Input Levels**
- **Low Supply Current . . . 8 mA Typical**
- **Designed to be Interchangeable With Maxim MAX232**
- **ESD Protection Exceeds JESD 22**
  – 2000-V Human-Body Model (A114-A)
- **Applications**
  - **TIA/EIA-232-F**
  - **Battery-Powered Systems**
  - **Terminals**
  - **Modems**
  - **Computers**

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
	SOIC (D)	Tube	MAX232D	MAX232
		Tape and reel	MAX232DR	
	SOIC (DW)	Tube	MAX232DW	MAX232
		Tape and reel	MAX232DWR	
	SOP (NS)	Tape and reel	MAX232NSR	MAX232
–40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
	SOIC (D)	Tube	MAX232ID	MAX232I
		Tape and reel	MAX232IDR	
	SOIC (DW)	Tube	MAX232IDW	MAX232I
		Tape and reel	MAX232IDWR	

† 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.

LinASIC is a trademark of Texas Instruments.

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.

Copyright © 2002, Texas Instruments Incorporated

PIC16F87/88

18/20-Pin Enhanced FLASH MCUs with nanoWatt Technology

Low Power Features:

- Power Managed modes:
  - Primary RUN XT, RC oscillator, 87 μA, 1 MHz, 2V
  - RC_RUN 7 μA, 31.25 kHz, 2V
  - SEC_RUN 14 μA, 32 kHz, 2V
  - SLEEP 0.2 μA, 2V
- Timer1 oscillator 1.3 μA, 32 kHz, 2V
- Watchdog Timer 0.7 μA, 2V
- Two-Speed Oscillator Start-up

Oscillators:

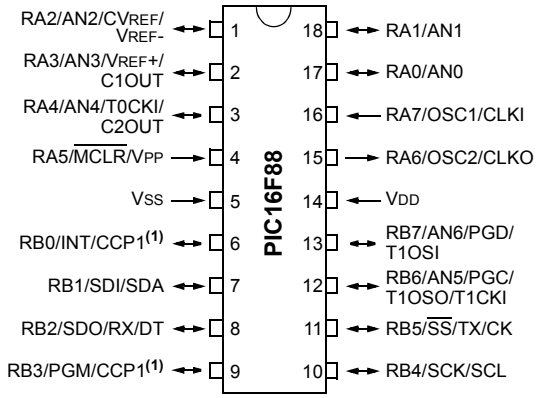
- Three Crystal modes:
  - LP, XT, HS up to 20 MHz
- Two External RC modes
- One External Clock mode:
  - ECIO up to 20 MHz
- Internal oscillator block:
  - 8 user selectable frequencies: 31 kHz, 125 kHz, 250 kHz, 500 kHz, 1 MHz, 2 MHz, 4 MHz, 8 MHz

Peripheral Features:

- Capture, Compare, PWM (CCP) module:
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit, 7-channel Analog-to-Digital Converter
- Synchronous Serial Port (SSP) with SPI™ (Master/Slave) and I^2C™ (Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection:
  - RS-232 operation using internal oscillator (no external crystal required)
- Dual Analog Comparator module:
  - Programmable on-chip voltage reference
  - Programmable input multiplexing from device inputs and internal voltage reference
  - Comparator outputs are externally accessible

Pin Diagram

18-Pin DIP, SOIC



Note 1: The CCP1 pin is determined by CCPMX in Configuration Word1 register.

Special Microcontroller Features:

- 100,000 erase/write cycles Enhanced FLASH program memory typical
- 1,000,000 typical erase/write cycles EEPROM data memory typical
- EEPROM Data Retention: > 40 years
- In-Circuit Serial Programming™ (ICSP™) - via two pins
- Processor read/write access to program memory
- Low Voltage Programming
- In-Circuit Debugging via two pins
- Extended Watchdog Timer (WDT):
  - Programmable period from 1 ms to 268s
- Wide operating voltage range: 2.0V to 5.5V

Device	Program Memory		Data Memory		I/O Pins	10-bit A/D (ch)	CCP (PWM)	USART	Comparators	SSP	Timers 8/16-bit
	FLASH (bytes)	# Single Word Instructions	SRAM (bytes)	EEPROM (bytes)							
PIC16F87	7168	4096	368	256	16	0	1	Y	2	Y	2/1
PIC16F88	7168	4096	368	256	16	1	1	Y	2	Y	2/1

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Références bibliographiques

Référence bibliographique :

[1] [ebook browse.com/mi/microcontrôleur-cours-pdf]

[2] daniel.menesplier.free.fr/doc/pic%20 16F88.pdf

[3] Diablo tronic 2.pagesperso-orange.fr/files/9_essentiel_des_pic-PDF document écrit par Alexandre GALODE

[4] hebergement.U-psud.fr/…/poly_manuel_acquisition 2012-2013

[5] www.bh-automation.fr/download/.../cv_sakli_MOUADH.pdf

[6] thierryperisse.free.fr/documents/projet_linel/…projet TP- stationmétéo.pdf

[7] dynelectrons.net76.net/montage/thermomètre à base du pic 16F88.pdf

[8] fabrice sincere.pagesperso-orange.fr/…/cours%20 filtrage% numérique

[9] www.microchip.com/download/endevicedoc/30487C.pdf

[10] aboulo.pagesperso-orange.fr/files/TP/le microcontroleur.pdf

[11] hearygfx.blogsport .com

[12] « étude et réalisation d'une carte d'acquisition analogique de l'ECG via la carte son du PC » mémoire électronique biomédical université Abou Bakr Belkaid-Tlemcen 2009-2010

[13] « exploration cardiovasculaire par étude corrélative des activités électrique et hémodynamique cardiaque et l'activité rhéologique pariétale » mémoire de magister électronique biomédical université Abou Bakr Belkaid-Tlemcen 2011-2012

[14]projret.PFE\fabrice.sincere.pagesperso-orange.fr\cm_electronique\projet_pic \aidememoire\16F88_ADC\ADC_16F88.htm

[15] projet_pic/aidememoire/16F876A_bus_I2C/bus_I2C_16F876A.htm#SSP

[16] projet_pic\carte acquisition 7 voies\carte_acquisition7voies.htm

RESUME :

L'objectif de ce mémoire de Master en Electronique Biomédical est la réalisation d'une carte d'acquisition 7 voies analogiques pour signaux lents. Les fonctions essentielles de la carte électronique réalisée sont : la conversion, le filtrage analogique et numérique. Les résultats obtenus de conversion de chaque voie sont affichées sur PC à partir d'une application de windows ***ADC108.exe.*** Cette application fonctionne sous windows XP il ya d'autre possibilité de sauvegarder les données sur un fichier textes. L'avantage principal de ce projet c'est que on peut acquérir par exemple la temperature d'un ou plusieurs patients dans des différentes parties du corps pendant des jours et même des années selon l'application qu'elle a été créée avec l'environnement C++ Borland Builder 6.0.

Mots clés : carte d'acquisition, pic 16F88, signal lent, RS232, MAX232

ABSTRACT :

Key words : acquisition card, pic 16F88,slow signal, RS232, MAX232