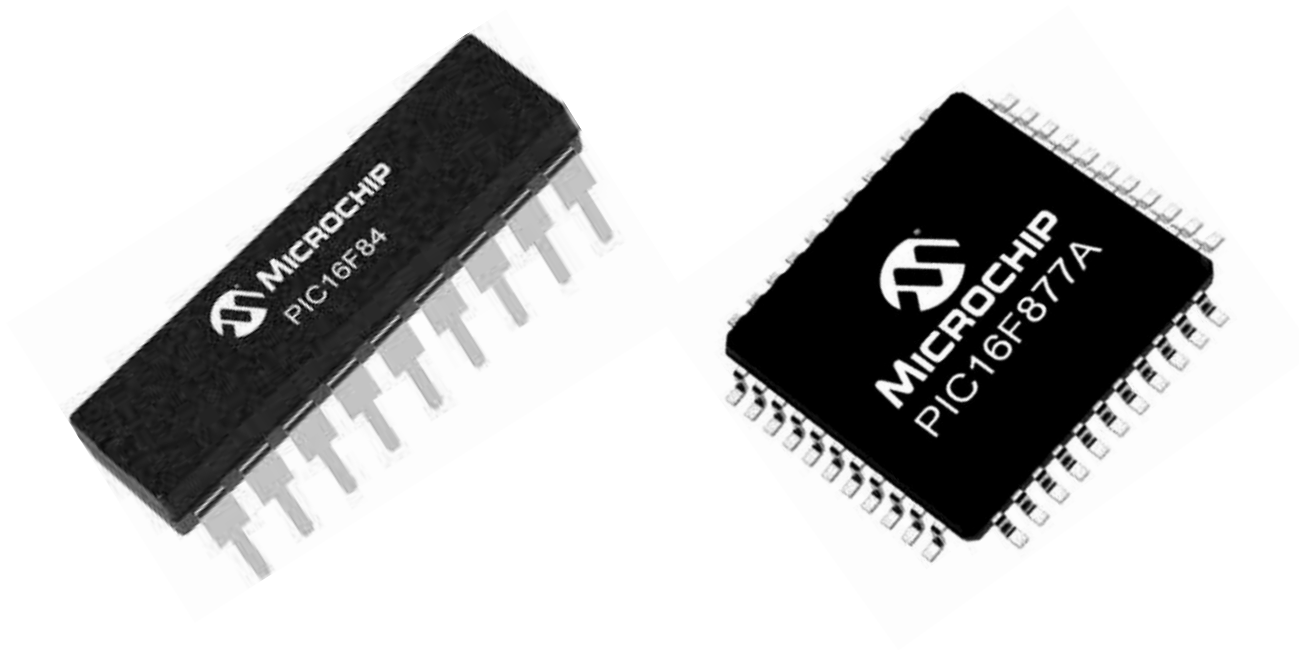


Cours: Microcontrôle



Chapitre II

Microcontrôleurs PIC



Microcontrôleurs PIC

A. Présentation d'un microcontrôleur PIC ;

Département d'informatique / FMI

Niveau : 1^{ère} Année Master GI

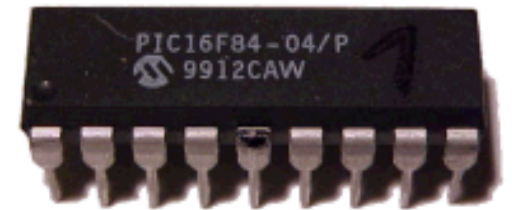
Module : Microcontrôleur

Chapitre II

- † Définition
- † Pourquoi les microcontrôleurs PIC ?
- † Famille des PIC

B. Microcontrôleur PIC16F84 ;

- † Définition ;
- † Constituants du PIC 16F84 ;
- † Brochage et fonctions des différentes pattes :
- † Architecture générale ;
- † Principe de fonctionnement ;
- † Organisation de la mémoire ; † Les ports d'entrées / sorties, Le Timer ;
- † Mise en œuvre ;
- † Jeu d'instructions .



Chapitre II

A. Présentation d'un microcontrôleur PIC

➔ Définition :

- Les microcontrôleurs PIC (**Programmable Interface Controllers**) sont des circuits électroniques qui peuvent être programmés pour effectuer une vaste gamme de tâches.
- Les PICs sont des composants RISC (Reduce Instructions Construction Set), ou encore composant à jeu d'instructions réduit.
- Les microcontrôleurs PIC sont apparus en **1993**



Chapitre II

A. Présentation d'un microcontrôleur PIC

➔ Pourquoi les microcontrôleurs PIC ?

✚ Avantages :

- Jeu d'instructions réduit (RISC): l'avantage est que plus on réduit le nombre d'instructions, **plus facile** et **plus rapide** en est le **décodage**, et **plus vite** le composant fonctionne ;
- Les microcontrôleurs PIC sont basés sur l'architecture **Harvard**, ce qui les rend **populaires** ;
- Facile à programmer ;
- Faible coût ;
- Grande disponibilité ;
- ...

Chapitre II

A. Présentation d'un microcontrôleur PIC

➔ La famille des PIC :

La famille des PIC à processeur 8 bits est subdivisée à l'heure actuelle en 3 grandes catégories :

- **Base-Line** : ils utilisent des mots d'instruction de 12 bits (ex : PIC 12F509).
- **Mid-Range** : ils utilisent des mots d'instruction de 14 bits (ex : PIC 16F84, PIC 16F877).
- **High-End** : ils utilisent des mots d'instruction de 16 bits (ex : PIC 18F4410).

Chapitre II

A. Présentation d'un microcontrôleur PIC

→ Identification d'un PIC :

Un PIC est identifié par un numéro de la forme suivant :

PIC xx (L) XX yy – zz |

- **xx** : Famille du composant (12, 14, 16, 17, 18)
- **L** : Tolérance plus importante de la plage de tension
- **XX** : Type de mémoire de programme. XX prend la lettre :
 - **C** pour une EPROM ou EEPROM ;
 - **CR** pour une PROM ;
 - **F** pour une mémoire de type Flash.
- **yy** : Identification

Chapitre II

- **zz** : Vitesse quartz de cadencement du microcontrôleur.

A. Présentation d'un microcontrôleur PIC

➔ Identification d'un PIC :

✚ Exemple :

PIC 16F887-04 |

- **16** : PIC Mid-Range
- **F** : mémoire programme est de type FLASH (F) • **877** : réinscriptible de type 877.

Chapitre II

- 04 : fréquence d'horloge de 4 Mhz.

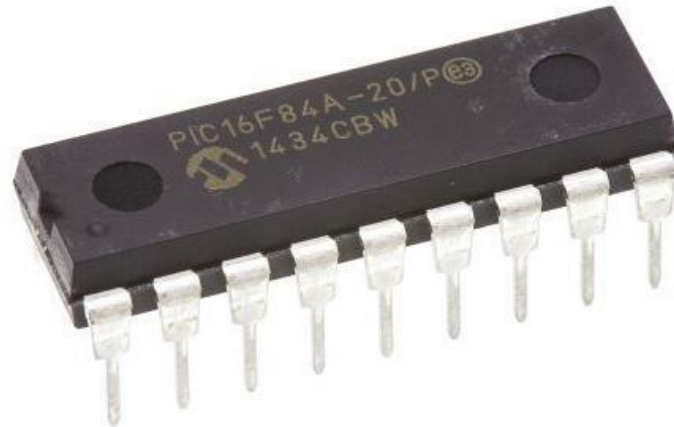


B. Microcontrôleur PIC 16F84

➔ Définition :

- Le **PIC 16F84** est un microcontrôleur à 08 bits avec un boîtier est un DIL (Dual In Line) de 2x9 pattes.
- En dépit de sa petite taille, il est caractérisé par une architecture interne qui lui confère souplesse et vitesse incomparables.

Chapitre II



B. Microcontrôleur PIC 16F84

➔ Constituants du PIC 16F84 :

Les principaux constituants du **PIC 16F84** sont :

- La mémoire de type **Flash ROM** pour le programme ;
- La mémoire de type **RAM** pour les registres ;
- La mémoire de type **EEPROM** pour les données à sauvegarder;

Chapitre II

- Des registres particuliers: de **travail (W)** et d'**état** ;
- L'Unité Arithmétique et Logique (**ALU**) ; • Les ports d'**entrées / sorties** ;
- Alimentation sous **5 Volts**.

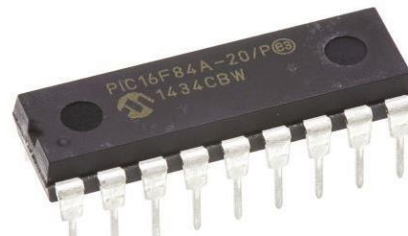
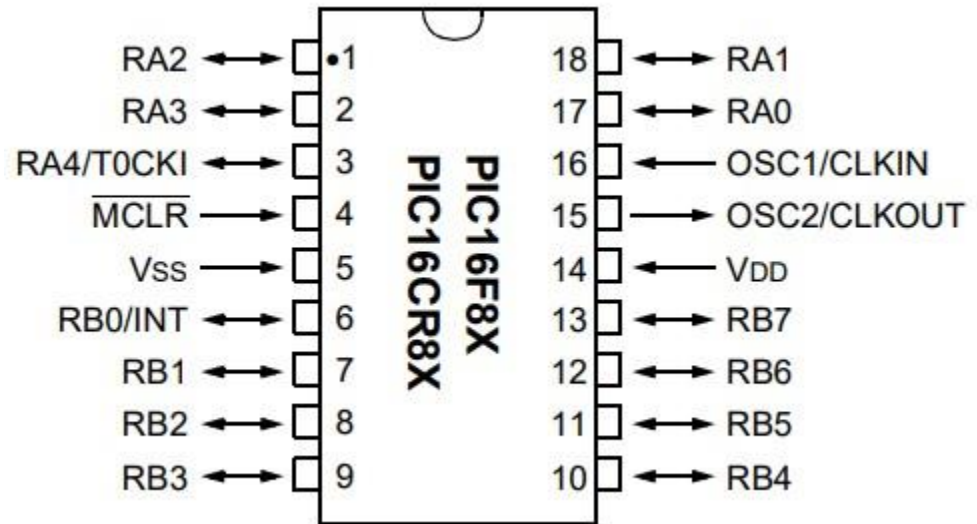
B. Microcontrôleur PIC 16F84

➔ **Brochage et fonctions des différentes pattes :**

Les principaux constituants du **PIC 16F84** sont :

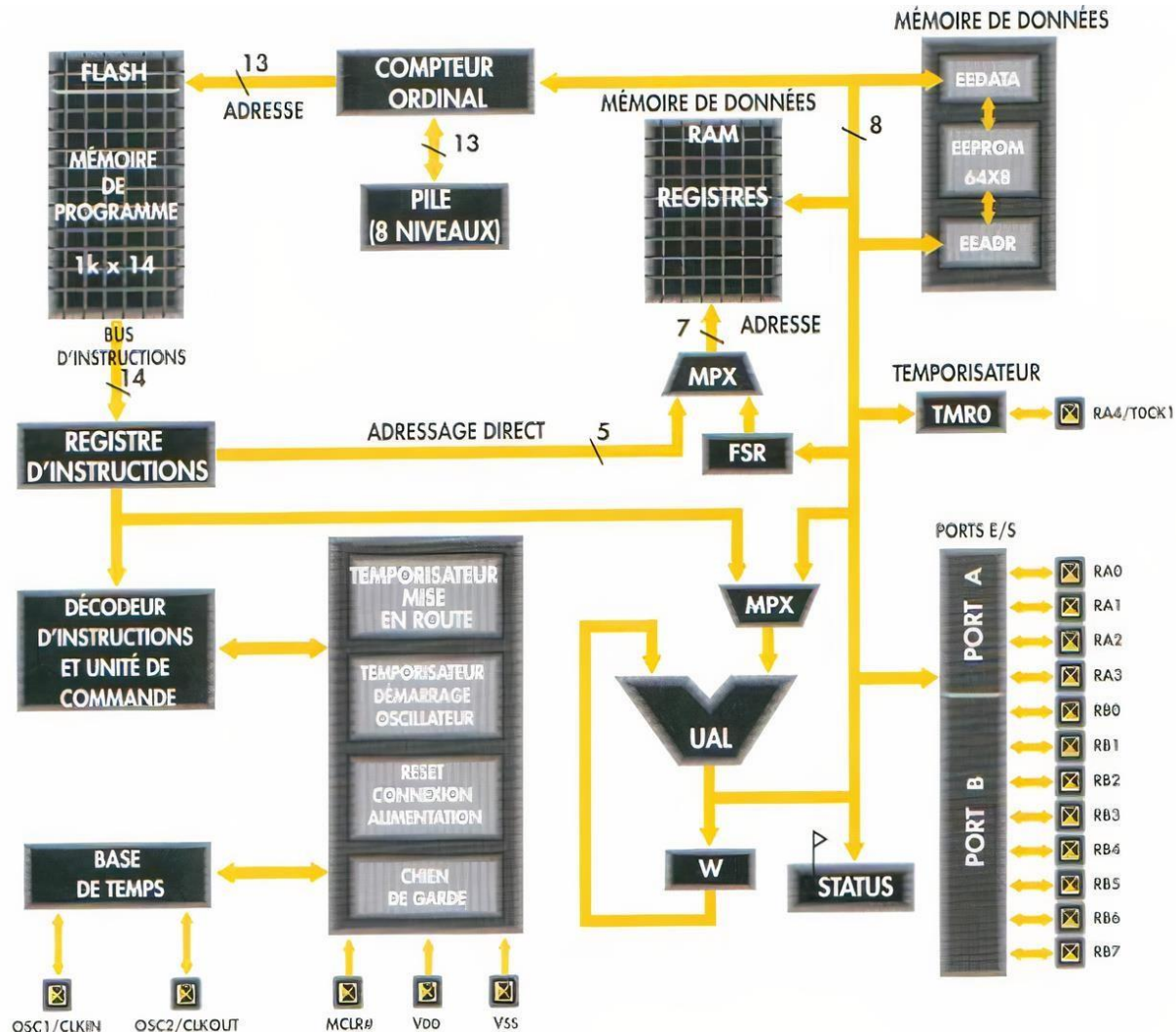
Chapitre II

- **VSS, VDD** : Alimentation
- **OSC 1,2** : Horloge
- **RA 0-4** : Port A
- **RB 0-7** : Port B
- **T0 CKL** : Entrée de comptage • **INT** : Entrée d'interruption
- **MCLR** : Reset : 0V.



Chapitre II

B. Microcontrôleur PIC 16F84



Chapitre II

➔ Architecture générale du PIC 16F84

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Architecture générale du PIC 16F84

- Un système d'initialisation à la mise sous tension (power-up timer, ...)
- Un système de génération d'horloge à partir du quartz externe (timing génération)
- Une unité arithmétique et logique (ALU)
- Une mémoire flash de programme de 1k "mots" de 14 bits
- Un compteur de programme (Program Counter)
- Un bus spécifique pour le programme (program bus)
- Un registre contenant le code de l'instruction à exécuter

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Architecture générale du PIC 16F84 (suite)

- Un bus spécifique pour les données (data bus)
- Une mémoire RAM contenant les SFR
- 68 octets de données
- Une mémoire EEPROM de 64 octets de données
- 2 ports d'entrées/sorties
- Un compteur (timer)
- Un chien de garde (watchdog)

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Principe de fonctionnement du PIC

Un microcontrôleur exécute des instructions. On définit :

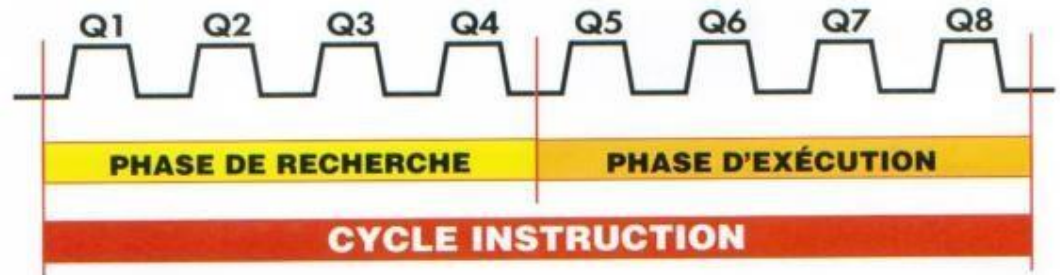
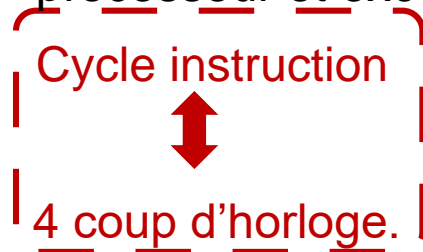
✚ Le cycle instruction

C'est le temps nécessaire à l'exécution d'une instruction. Une instruction est exécutée en deux phases :

- La **phase de recherche** du code binaire de l'instruction stocké dans la mémoire de programme

Chapitre II

- La **phase d'exécution** ou le code de l'instruction est interprété par le processeur et exécuté.



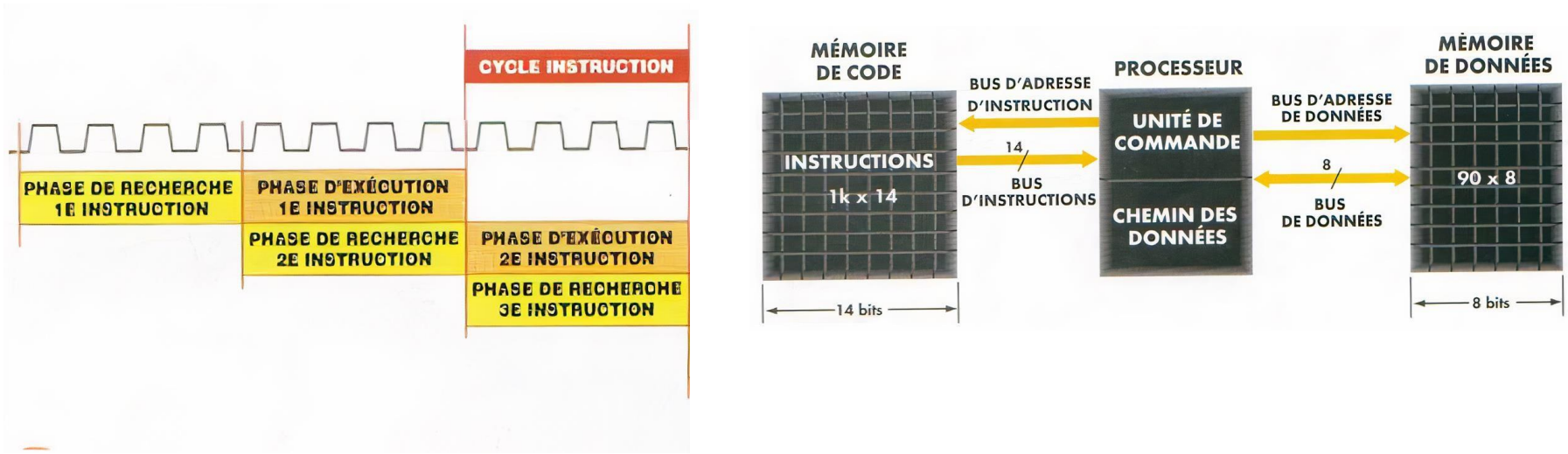
B. Microcontrôleur PIC 16F84

➔ Principe de fonctionnement du PIC

⚡ Le cycle instruction (suite)

- Les instructions issues de la mémoire de programme circulent sur un bus différent de celui sur lequel circulent les données ;
- Le processeur peut effectuer la phase de recherche d'une instruction pendant qu'il exécute l'instruction précédente

Chapitre II



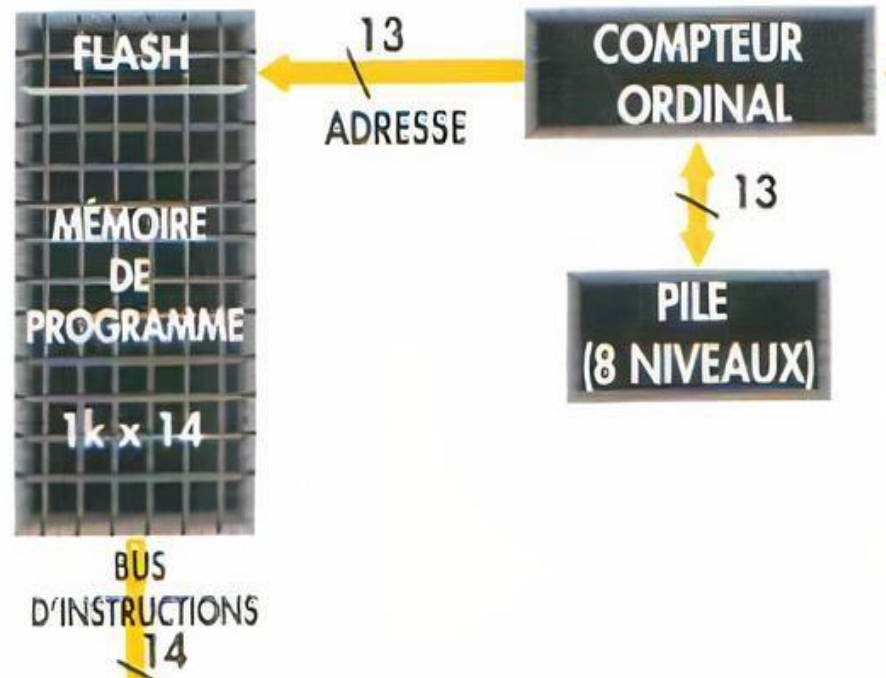
B. Microcontrôleur PIC 16F84

➔ Organisation de la mémoire 1.

Mémoire de programme (mémoire morte)

- Elle contient le programme à exécuter.

Chapitre II



Chapitre II

B. Microcontrôleur PIC 16F84

Chapitre II

→ Organisation de la mémoire

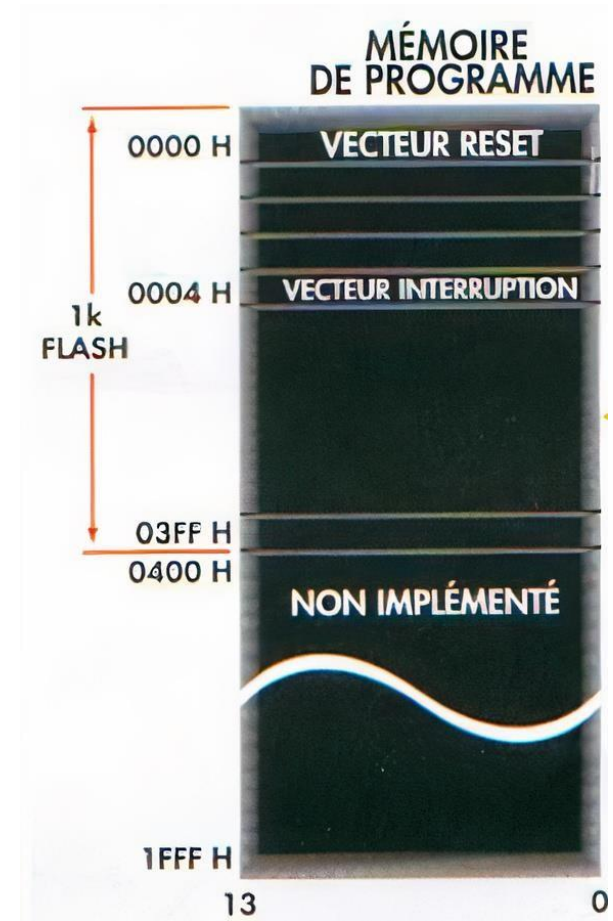
1. Mémoire de programme (mémoire morte)

- C'est une mémoire flash de 1k "mots" de 14 bits (1024 emplacements);
- L'adresse **0000** est réservée au vecteur **RESET** ;
- L'adresse **0004** est assignée au vecteur d'**interruption** et fonctionne de manière similaire à celle du vecteur de Reset.

B. Microcontrôleur PIC 16F84

→ Organisation de la mémoire

1. Mémoire de programme (mémoire morte)

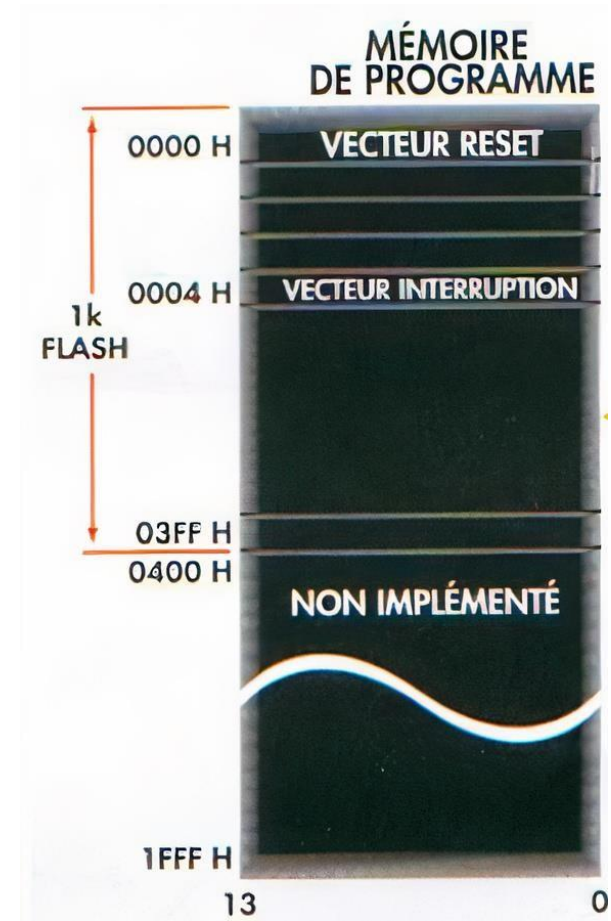


Chapitre II

- **Déroulement d'un programme** : A la mise sous tension :
- Le processeur va chercher la première instruction qui se trouve à l'adresse **0000** de la mémoire de programme ;
 - Le processeur puis va chercher la deuxième instruction à l'adresse **0001**;
 - Le processeur peut alors sélectionner l'emplacement souhaité grâce au **bus d'adresse** et il peut lire son contenu (ici l'instruction) grâce à son **bus d'instruction**;

B. Microcontrôleur PIC 16F84

➔ **Organisation de la mémoire**



Chapitre II

1. Mémoire de programme (mémoire morte)

□ Déroulement d'un programme :

Adressage s'effectue à l'aide d'un Compteur Ordinal appelé **PC** ;

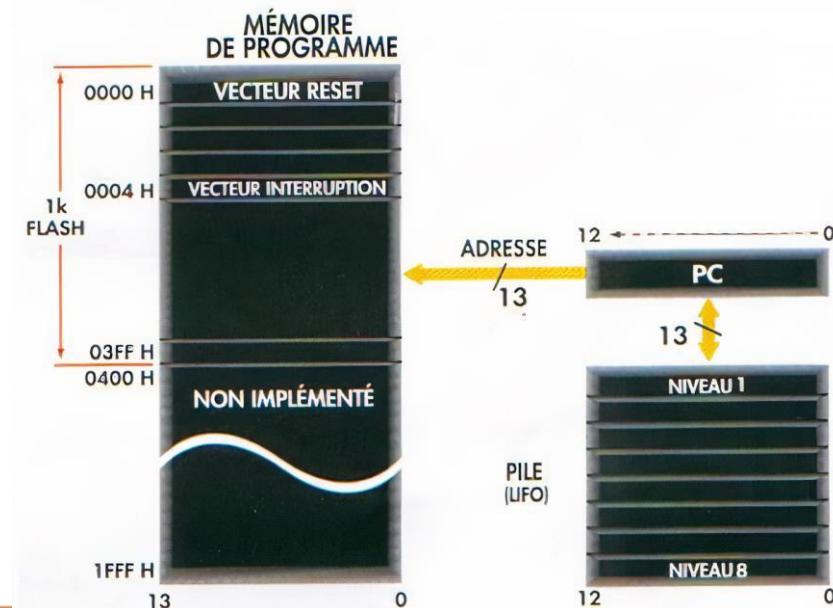
- La **pile** sert à emmagasiner de manière temporaire l'adresse d'une instruction (cas d'exécution d'un **sous programme**).

B. Microcontrôleur PIC 16F84

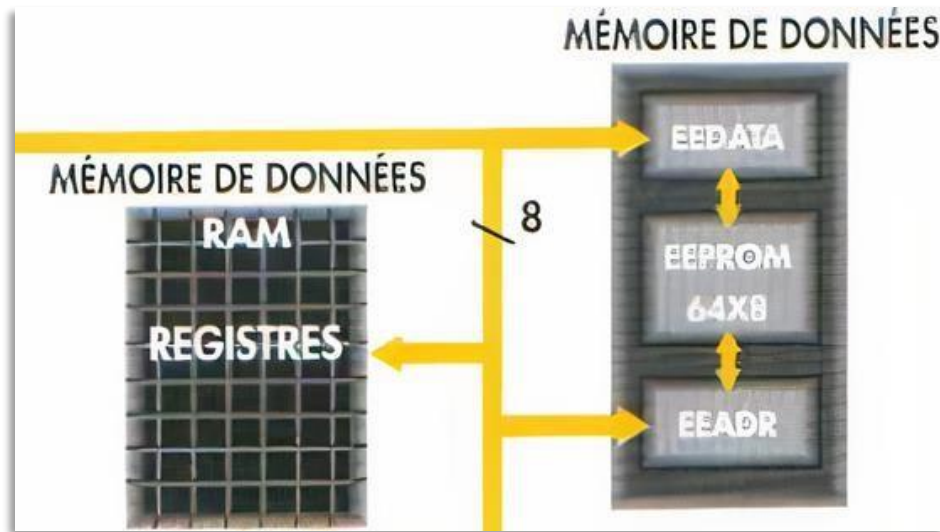
→ Organisation de la mémoire

2. La mémoire de données RAM :

Elle se décompose en deux parties de **RAM** et une zone **EEPROM**.



Chapitre II



B. Microcontrôleur PIC 16F84

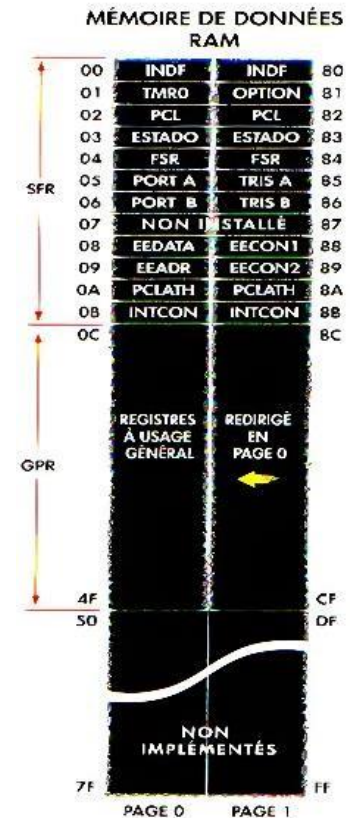
➔ Organisation de la mémoire

2. La mémoire de données RAM :

- La première partie de RAM contient les **SFRs** (*Special Function Registers*) qui permettent de contrôler le PIC;

Chapitre II

- La seconde partie de RAM contient des registres généraux **GPR (68 octets)**, libres pour l'utilisateur ;
- La mémoire est séparée en deux pages (0 et 1) ;
- Le **bus d'adresse** qui permet d'adresser la RAM est composé de **7 fils** (adresser **128 emplacements** différents);
- Chaque page de la RAM est composée de **128 octets** ;



Chapitre II

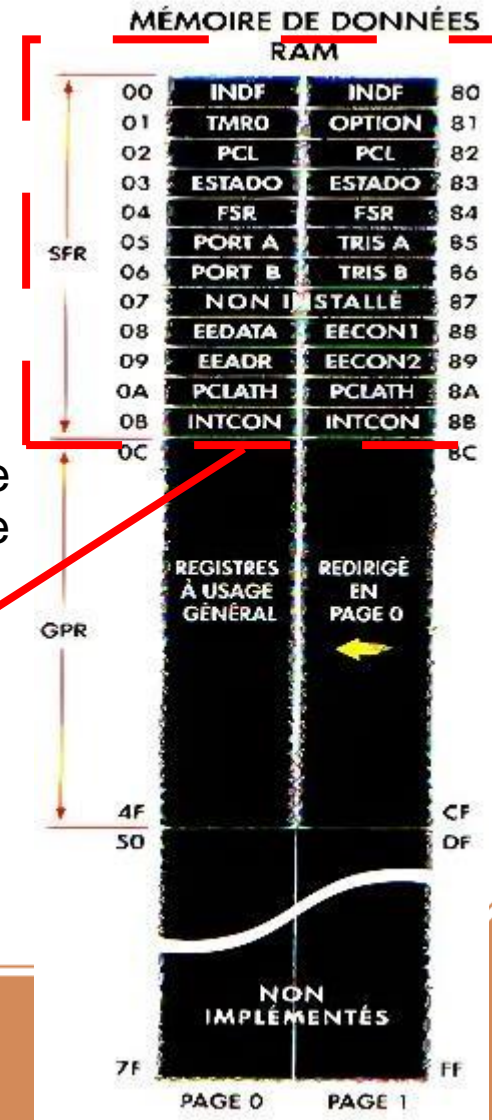
B. Microcontrôleur PIC 16F84

- Les premiers octets sont réservés pour **SFR** (Special File Register) : Ces emplacements sont en effet utilisés par le microcontrôleur pour configurer l'ensemble de son fonctionnement ;



A vertical diagram showing the memory layout of Special File Registers (SFR) for the PIC 16F84. It consists of a table with four columns: address (00 to 0B), register name, a second register name, and another address (80 to 8B). A vertical double-headed arrow on the left is labeled 'SFR'.

00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B



Chapitre II

➔ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres
spéciaux SFRs :

Chapitre II

B. Microcontrôleur PIC 16F84

→ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres



00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B

spéciaux SFRs :

Chapitre II

Registre	Fonctionnement
INDF (00h - 80h)	Utilise le contenu de FSR pour l'accès indirect à la mémoire
TMR0 (01h)	Registre lié au compteur (Timer)
PCL (02h - 82h)	Contient les poids faibles du compteur de programmes (PC)
PCLATH (0Ah-8Ah)	Contient les poids forts

Chapitre II

B. Microcontrôleur PIC 16F84

→ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres



00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B

spéciaux SFRs :

Chapitre II

Registre	Fonctionnement
STATUS (03h - 83h)	Il contient l'état de l'unité arithmétique et logique ainsi que les bits de sélection des pages (banks)
FSR (04h - 84h)	Registre de sélection de registre (adressage indirect)

Chapitre II

B. Microcontrôleur PIC 16F84

→ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres



00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B

spéciaux SFRs :

Chapitre II

Registre	Fonctionnement
PORTA (05h)	Donne accès en lecture ou écriture au port A de 5 bits : RA0 à RA4. La ligne RA4 peut être utiliser en entrée de comptage.
PORTB (06h)	Donne accès en lecture ou écriture au port B : 8 bits. la ligne RB0 peut être utiliser en entrée d'interruption.

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres



00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B

spéciaux SFRs :

Chapitre II

Registre	Fonctionnement
EEDATA (08h)	Permet l'accès aux données dans la mémoire EEPROM.
EEADR (09h)	Permet l'accès aux adresses de la mémoire EEPROM
INTCON (0Bh-8Bh)	Contrôle des interruptions
OPTION_ REG (81h)	Contient des bits de configuration pour divers périphériques

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres



00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	ESTADO	ESTADO	83
04	FSR	FSR	84
05	PORT A	TRIS A	85
06	PORT B	TRIS B	86
07	NON INSTALLÉ		87
08	EEDATA	EECON1	88
09	EEADR	EECON2	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B

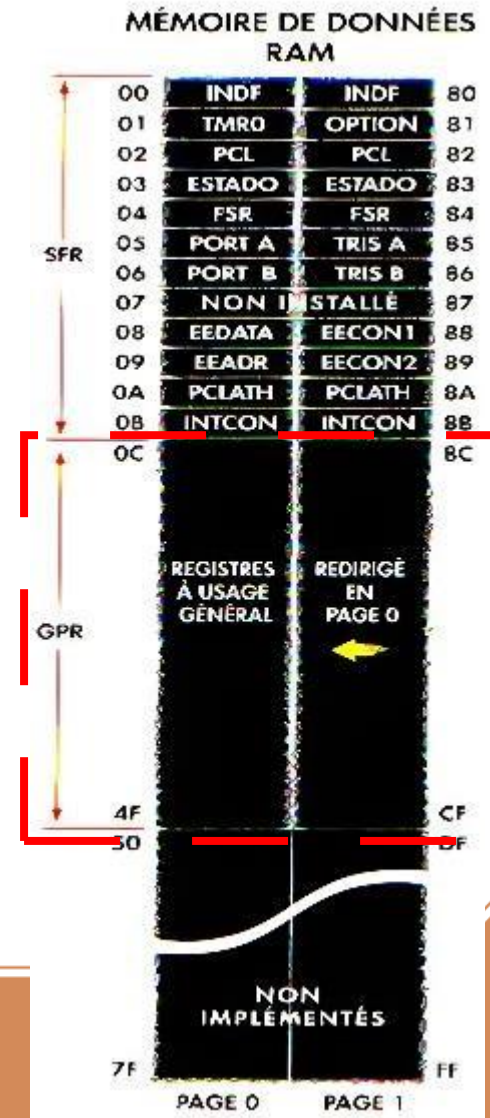
spéciaux SFRs :

Chapitre II

Registre	Fonctionnement
TRISA (85h)	Indique la direction (entrée ou sortie) du port A
TRISB (86h)	Indique la direction (entrée ou sortie) du port B
EECON1 (88h-89h)	Permet le contrôle d'accès à la mémoire EEPROM.

Chapitre II

B. Microcontrôleur PIC 16F84



Chapitre II

→ Organisation de la mémoire

2. La mémoire de données RAM ○ Registres généraux GPR :

- La RAM de données proprement dite se réduit donc à la zone notée **GPR** (**R**egistre à **U**sage **G**énérale) ;
- 68 registres en page 0 et autant en page 1 ;
- Les données écrites en page 1 sont redirigées en page 0;

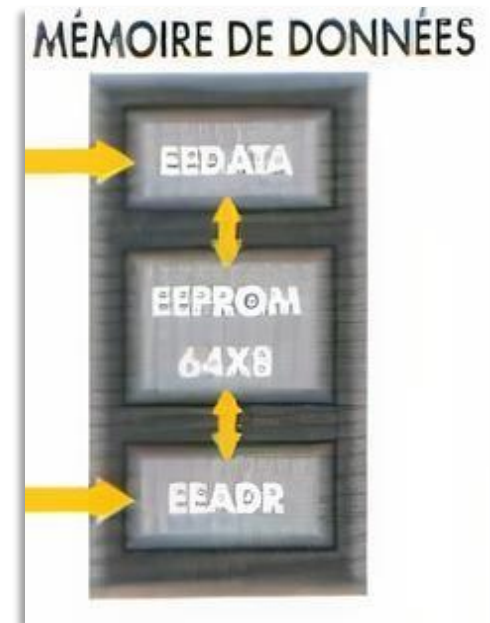
B. Microcontrôleur PIC 16F84

→ Organisation de la mémoire

Chapitre II

2. La mémoire de données RAM ○ Mémoire EEPROM :

- Le PIC 16F84 possède une zone **EEPROM** de **64 octets** accessibles en lecture et en écriture par le programme ;
- On peut y sauvegarder des valeurs, qui seront conservées même si l'alimentation est éteinte, et les récupérer lors de la mise sous tension.



Chapitre II

B. Microcontrôleur PIC 16F84

➔ Les ports d'entrées / sorties

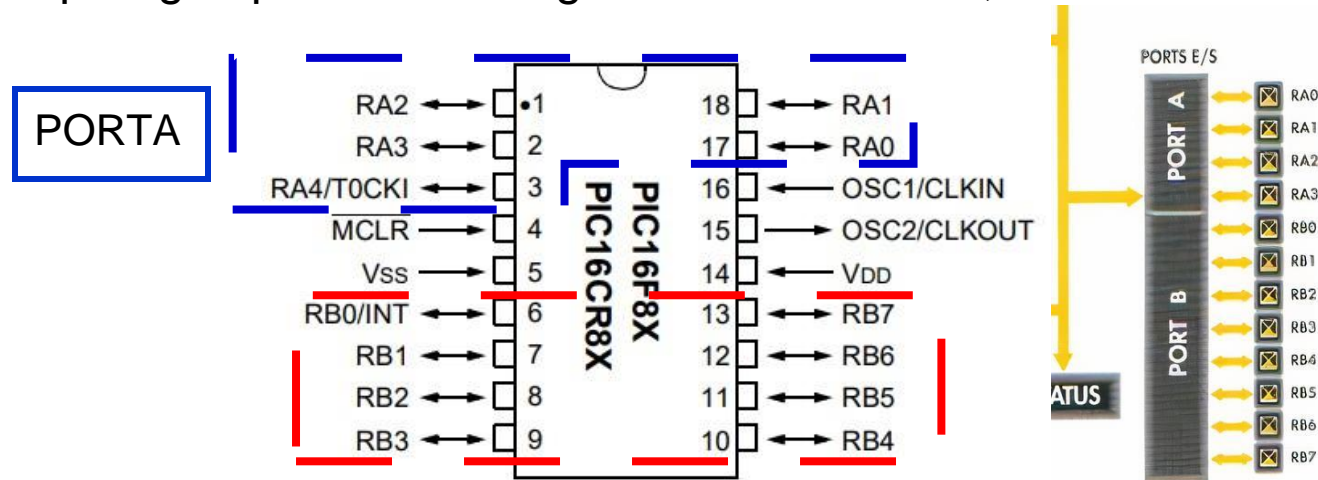
Le PIC16F84 est équipé de **13 lignes d'entrées/sorties** réparties en deux ports :

- **Le port A** : RA0 à RA4 (RA4 peut être utiliser en entrée de comptage)
- **Le port B** : RB0 à RB7 (RB0 peut être utiliser en entrée d'interruption)

PORTB

Chapitre II

- Chaque ligne peut être configurée soit en entrée, soit en sortie.



B. Microcontrôleur PIC 16F84

→ Le Timer

Dans la majeure partie des applications, il est nécessaire de contrôler le temps.

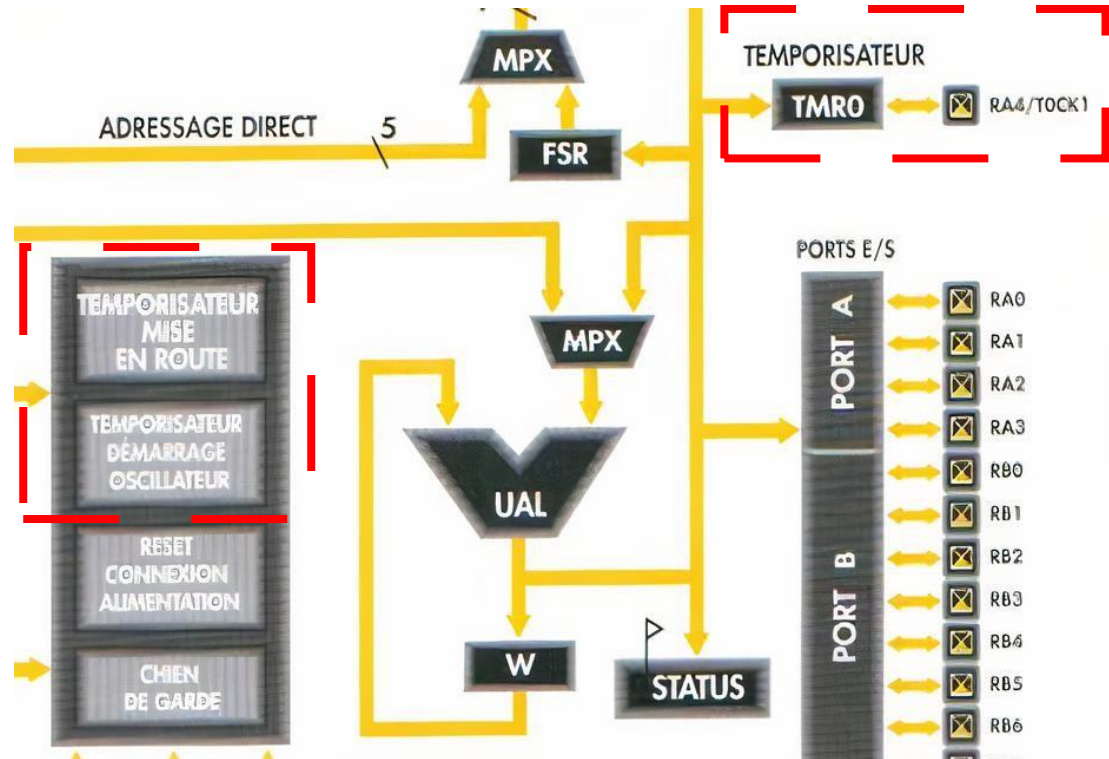
Chapitre II

- Le pic16F84 dispose de deux timers:
 - Un à usage général (le **TMR0**) ;
 - Un autre utilisé pour le chien de garde (watch dog **WDG**).

B. Microcontrôleur PIC 16F84

✈ Mise en oeuvre

L'utilisation et la mise en œuvre très simple des PICs les a rendus extrêmement populaire.



Chapitre II

- Il suffit d'alimenter le circuit par ses deux broches VDD et VSS ;
- De fixer sa vitesse de fonctionnement à l'aide d'un **quartz** (Fig. 2);
- D'élaborer un petit système pour permettre de réinitialiser le microcontrôleur sans avoir à couper l'alimentation (Fig. 3);
- Il suffit ensuite d'écrire le programme en langage assembleur sur un ordinateur grâce au logiciel **mikroC** puis de le compiler pour le transformer en langage machine et le transférer dans le PIC grâce à un programmeur.

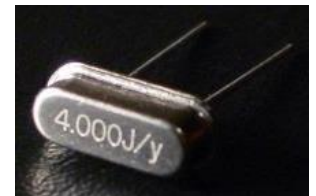
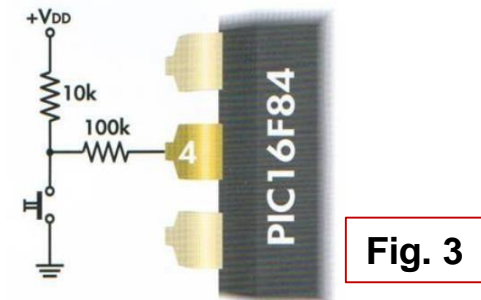
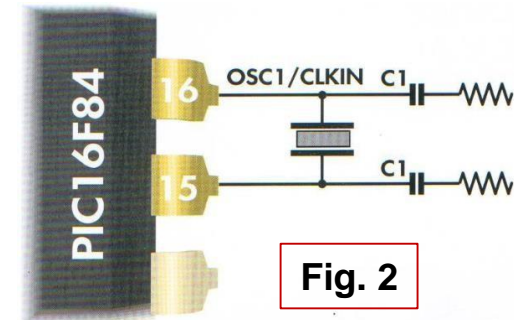


Fig. 1: Un quartz de 4 MHz

Chapitre II



B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Chapitre II

Toutes les instructions compréhensibles par les microcontrôleurs forment ce que l'on appelle le **jeu d'instructions**.

□ Règle des différents mnémoniques du jeu d'instructions :

Afin de comprendre la fonction de chaque instruction, la notation adoptée pour les données et adresses manipulées par les instructions est fort simple et est la suivante :

k : est un **littéral**, c'est-à-dire **une valeur codée sur un octet** (8 bits).

f : est le **symbole** correspondant à **un registre**. **b** : est le **numéro du bit** concerné par l'**instruction**. **d** : spécifie l'**endroit** où doit être placé le **résultat de l'opération**.

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

○ **Exemple** : Un certain nombre d'instructions (**ADDWF**, **ANDWF**, etc..) utilise une notation spéciale présentée sous la forme :

- **ADDWF f, d**

Où **f** indique le registre et où **d** peut prendre deux valeurs (**0 ou 1**), ce qui change le comportement de l'instruction. Si **d** est à **0**, le résultat est placé dans le registre de travail **W**, la valeur dans le registre **f** est alors inchangée, alors que si **d** est à **1**, le résultat est placé dans le registre **f**.

Si **d = 0** le résultat est placé dans **W**.

Si **d = 1** le résultat est placé dans **f**.

Chapitre II

Si **d = 0** **W + f → W**

Si **d = 1** **W + f → f**

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
ADDLW (Add Literal to W)	ADDLW k	On ajoute au registre de travail la valeur k et on place le résultat dans le registre de travail W
ADDWF (Add W to F)	ADDWF f, d	On ajoute le contenu de W et le contenu de f et on place le résultat dans f si d=1 ou dans W si d=0

Chapitre II

ANDLW (And Literal and W)	ANDLW k	On effectue un ET logique entre k et le contenu de W, et on place le résultat dans le registre de travail W
ANDWF (And W with F)	ANDWF f, d	On effectue un ET logique entre le contenu de W et le contenu de f , on place le résultat dans W si d=0 ou dans f si d=1

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
BCF (Bit Clear F)	BCF f, b	On met à 0 le bit b du registre f

Chapitre II

BSF (Bit Set F)	BSF f, b	On met à 1 le bit b du registre f
BTFSC (Bit Test, Skip if Clear)	BTFSC f, b	Saut de l'instruction qui suit, si le bit b de f est nul $b(f)=0$
BTFSS (Bit Test, Skip if Set)	BTFSS f, b	Saut de l'instruction qui suit si $b(f)=1$

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
-------------	---------	-----------

Chapitre II

CALL (subroutine Call)	CALL label	Appeler un sou programme (label)
CLRF (Clear F)	CLRF f	On met le contenu du registre f à 0
CLRW (Clear W)	CLRW	On met le contenu du registre W à 0
CLRWD (Clear Watch Dog Timer)	CLRWD	On met le contenu du registre du timer chien de garde à 0

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

Chapitre II

INSTRUCTION	SYNTAXE	OPERATION
COMF (Complement F)	COMF f,d	On complémente le contenu du registre f bit à bit, le résultat est placé dans f si d=1, dans W si d=0.
DECF (Decrement F)	DECF f,d	On diminue le contenu du registre f d'une unité, le résultat est placé dans f si d=1, dans W si d=0 (dans ce cas f reste inchangé).
DECFSZ	DECFSZ f,d	On diminue le contenu du registre f d'une unité, le résultat est placé dans f si d=1, dans W si d=0
GOTO	GOTO label	On effectue un saut dans le programme pour aller à l'adresse pointé par le label précisé dans GOTO
INCF (Increment F)	INCF f, d	On augmente le contenu du registre f d'une unité, le résultat est placé dans f si d=1, dans W si d=0

Chapitre II

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
INCFSZ (Increment F, Skip if Zero)	INCFSZ f, d	On augmente le contenu du registre f d'une unité, le résultat est placé dans f si d=1, dans W si d=0
IORLW (Inclusive Or literal with W)	IORLW k	On effectue un OU logique entre le contenu de W et le littéral k, le résultat est placé dans W.
IORWF (Inclusive Or W with F)	GOTO label	On effectue un saut dans le programme pour aller à l'adresse pointé par le label précisé dans GOTO

Chapitre II

MOVF (Move F)	MOVF f,d	On déplace le contenu de f dans f si d=1 ou de f dans W si d=0,
-----------------	----------	---

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
MOVLW (Move Literal to W)	MOVLW k	On charge le contenu de W avec le littéral k
MOVWF (Move W to F)	MOVWF f	On charge le contenu de f avec le contenu de W
NOP (No Operation)	NOP	On ne fait que consommer du temps machine

Chapitre II

RETFIE (Return From Interrupt)	RETFIE	On charge le PC avec la valeur qui se trouve au sommet de la pile pour revenir au programme principal lorsque l'exécution du sous programme est terminée
RETLW (Return Literal toW)	RETLW k	On charge le contenu de W avec le littéral k puis on charge le PC avec la valeur qui se trouve au sommet de la pile effectuant ainsi un retour de sous programme.

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
RETURN Return	RETURN	n charge le PC avec la valeur qui se trouve au sommet de la pile effectuant ainsi un retour de sous programme.

Chapitre II

SLEEP (Sleep)	SLEEP	On place le circuit en mode sommeil avec arrêt de l'oscillateur.
SUBLW (Subtract W from Literal)	SUBLW k	On soustrait le contenu du registre W du littéral k et on place le résultat dans W
SUBWF (Subtract W from F)	SUBWF f,d	On soustrait le contenu du registre W du contenu du registre f et on place le résultat dans W si d=0, ou dans f si d=1

B. Microcontrôleur PIC 16F84

➔ Jeu d'instructions du PIC 16F84

Les 35 instructions sont donc les suivantes :

INSTRUCTION	SYNTAXE	OPERATION
-------------	---------	-----------

Chapitre II

SWAPF (Swap F)	SWAPF f, d	On échange les quatre bits de poids forts avec les quatre bits de poids faibles et on place le résultat dans W si d=0, ou dans f si d=1
XORLW (Exclusive Or Literal with W)	XORLW k	On effectue un OU Exclusif entre W et le littéral k, le résultat est placé dans W
XORWF (Exclusive Or W with F)	XORWF f, d	On effectue un OU Exclusif entre W et le contenu de f, le résultat est placé dans W si d=0, sinon il est placé dans f
RLF	RLF f, d	Rotation à gauche du contenu du registre f.
RRF	RRF f, d	Rotation à droite du contenu du registre f.