TD 4

# Programme et directives

Les microcontrôleurs sont des composants programmables. Ils font ce que leur dit de faire le programme et rien d'autre.

Le programme est une suite d'instructions. Elles sont codées en binaire (code machine) pour pouvoir être exécutées par le microcontrôleur ; Pour écrire un programme, il existe plusieurs solutions (langages) mais chacune nécessite une forme de traduction.

**Programmation en langage assembleur** : Avant la construction d'un programme, il est recommandé de réaliser un algorigramme qui représente le cheminement du programme à écrire. Cela va faciliter la programmation.

**Les directives les plus utilisées :**

Les directives de l'assembleur sont des instructions qu'on ajoute dans le programme et qui seront interprétées par l'assembleur.

- **LIST** : Permet de définir un certain nombre de paramètres comme le processeur utilisé (p), la base par défaut pour les nombres (r), le format du fichier hex à produire (f) ainsi que d'autres paramètres.

Exemple: LIST p=16F84A, r=dec, f=inhx8m

- **INCLUDE** : permet d'insérer un fichier source. Par exemple le fichier **p16f84A.inc** qui contient la définition d'un certain nombre de constante comme les noms des registres ainsi que les noms de certains bits.

Exemple: **# INCLUDE**< p16f84A.inc>

**INCLUDE**p16f84A.inc

- **__CONFIG** : permet de définir les 14bits de configuration qui seront copié dans l'EEPROM de configuration lors de l'implantation du programme dans le PIC (protection de code, type d'oscillateur, chien de garde et temporisation du départ)

Chargée de cours : Dr. K.Chaker
Chargée de TD : Dr. N.Merabti
License Automatique.
Module Mp/Mc.

| 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CP | CP | CP | CP | CP | CP | CP | CP | CP | CP | PWRTE | WDTE | FOSC1 | FOSC0 |

**FOSC1** : **FOSC0** Sélection du type d'oscillateur pour l'horloge :
11 : Oscillateur **RC ;**
10 : Oscillateur HS (High speed) : **quartz** haute fréquence jusqu'à **10 MHz** ;
01 : Oscillateur XT, c'est le mode le plus utilisé, **quartz** jusqu'à **4 MHz ;**
00 : Oscillateur LP (Low power), consommation réduite, jusqu'à **200 kHz.**

**WDTE** validation du timer**WDT** (chien de garde) :
1 : WDT validé ;
0 : WDT inhibé.

**PWRTE** validation d'une temporisation à la mise sous tension :
1 : temporisation inhibée ;
0 : temporisation validée.

**CP** Protection en lecture du code programme :
1 : pas de protection ;
0 : protection activée

Voici les valeurs et leurs définitions :

```
;       _CP_ON          Code protection ON : impossible de relire
;       _CP_OFF         Code protection OFF
;       _PWRTE_ON       Timer reset sur power on en service
;       _PWRTE_OFF      Timer reset hors-service
;       _WDT_ON         Watch-dog en service
;       _WDT_OFF        Watch-dog hors service
;       _LP_OSC         Oscillateur quartz basse vitesse
;       _XT_OSC         Oscillateur quartz moyenne vitesse
;       _HS_OSC         Oscillateur quartz grande vitesse
;       _RC_OSC         Oscillateur à réseau RC
```

Ex: __CONFIG _CP_OFF & _HS_OSC & _PWRTE_OFF & _WDT_OFF

Dans cet exemple le bit CP est à 0, le bit WDT (watchdog) est à 0, le bit PWRTE est à 1 et le terme HS définit le type d'horloge utilisé.

**Exemple** : **__CONFIG** B'11111111111001'

**__CONFIG** H'3FF9'

• **EQU** : permet de définir une constante ou une variable :

Exemple : XX **EQU** 0x20

Chaque fois que le compilateur rencontrera **XX**, il la remplacera par la constante 0x20.

• **#DEFINE** : définit un texte de substitution

**Exemples** :

**#DEFINE** led1 PORTA,0 ; led1 est remplacé par PORTA,0

**#DEFINE** Bp2 PORTA,3 ; Bp2 est remplacé par PORTA,3

• **ORG** : permet de choisir l'adresse de début, dans laquelle sera logée la première instruction de votre programme.

**Exemple** : **ORG** H'0000' ;

Le programme commencera à l'adresse **0000H** de la mémoire programme du PIC.

**Si** on utilise **un sous-programme d'interruption**, il faut éviter d'utiliser dans le programme principal l'adresse **0004H**. Dans ce cas il faut loger le programme principal à une autre adresse que **0000H**. La fonction **GOTO** permet de renvoyer vers une étiquette ou une adresse.

**Exemple** : **ORG** H'0000' ; initialisation du PIC

**GOTO** start ; va à l'étiquette start

La syntaxe **ORG**, peut être utilisée plusieurs fois, ce qui permet de loger des sous programmes à des emplacements mémoire définie.

• **;** : le compilateur ignore tout ce qui suit un point-virgule (; ).

**Exemple** : ; je peux écrire des commentaires après le point-virgule

• **END** : précise où doit s'arrêter l'assemblage, les instructions situées après sont ignorées.

**Les étiquettes (Labels)** : permettent de remplacer une adresse en format numérique.

**Exemple** : GOTO **tempo**

**Format des nombres** : L'assembleur reconnaît les nombres en décimal, hexadécimal, binaire ou octal. Pour préciser la base il faut utiliser les préfixes précisés dans le tableau ci-dessous :

| Base | Préfixe | Exemple |
|---|---|---|
| **Décimal** | **D**'nnn'<br>.nnn | **D**'39'<br>.39 |
| **Hexadécimal** | **H**'nn'<br>**0x**nn | **H**'2F'<br>**0x**2F |
| **Binaire** | **B**'….' | **B**'01100101' |

**Exercice1:** Configurer les broches: RB0, RB3, RB6, RB7 du PORT B en entrée, et les broches RB1, RB2, RB4, RB5 en sortie.

**Exercice2 :** Configurer les broches RA0, RA1, RA2 du PORT A en entrée et les broches RA3, RA4 du PORT A en sortie.

**Exercice 3 :** Écrire un programme qui additionne deux valeurs en RAM ("val1" et "val2") et met le résultat dans une variable 8 bits "Res", en utilisant les directives de la compilation

**Exercice4 :** Ecrire un programme qui permet d'Allumer et éteindre une LED par deux boutons poussoirs

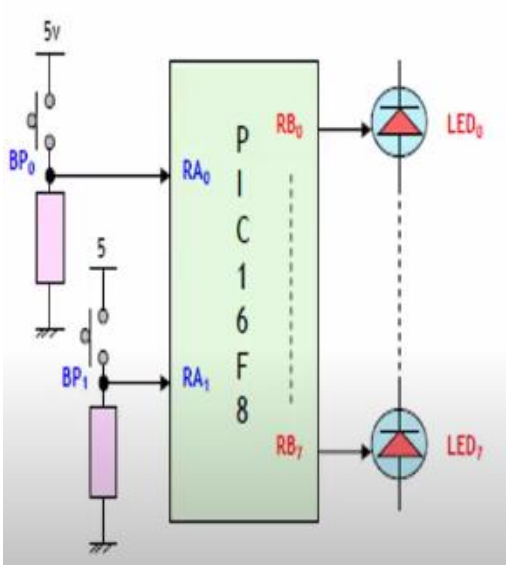
**Exercice5 :**

Ecrire un programme qui permet d'allumer 8 LEDs par deux boutons poussoirs $BP_0$ et $BP_1$

- Une action sur le bouton poussoir $BP_1$ allume toutes les LEDs

- Une action sur le bouton poussoir $BP_0$ éteint toutes les LEDs

Sachant qu'au repos les LEDs sont éteintes.




Chargée de cours : Dr. K.Chaker
Chargée de TD : Dr. N.Merabti
License Automatique.
Module Mp/Mc.