REPUBLIQUE ALGERENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Badji Mokhtar Annaba
Faculté de technologie
Département d'électronique

**Travaux Pratiques Mp Mc**

# TP n° 3

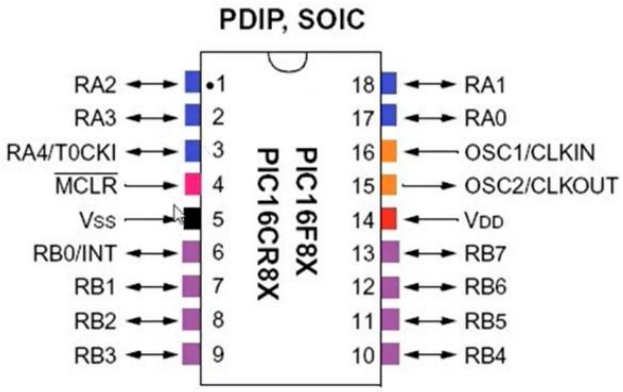
## Manipulation des entrées/sorties sur un PIC
## Programme assembleur pic16f84

Ces Travaux pratiques ont été réalisés pour le module TP Microprocesseurs et Microcontrôleurs de la troisième année licence automatique,Pour l'année universitaire 2024/2025 au département d'electronique, Université Badji Mokhtar ANNABA.

**Par Dr. MERABTI Nardjes**

**Objectif** : Ce TP a pour objectif de vous initier à la programmation en assembleur. Nous allons aborder les concepts de base à travers la création d'un programme simple sur MPLAB comme par exemple faire allumer une led LED avec le PIC16F84 en appuyons sur un interrupteur

**RAPPEL:**




- ✓ VDD et VSS: Borne d'alimentation +5V et 0V. C'est ici que seront branchés le régulateur de tension et la masse.
- ✓ MCLR : "Reset" du circuit.
- ✓ OSC1 et OSC2 : Brôches reçevant le montage horloge.
- ✓ RA0 à RA4 : Port d'E/S A.
- ✓ RB0 à RB7 : Port d'E/S B

**Rappel sur les bases de la programmation en assembleur**

1. **Architecture du processeur :** Chaque processeur possède son propre jeu d'instructions. Il est donc essentiel de connaître l'architecture spécifique pour laquelle vous écrivez du code.
2. **Registres :** Les registres sont des petites mémoires présentes dans le processeur utilisé pour stocker temporairement des données.
3. **Instructions :** Les instructions en assembleur sont des commandes simples qui indiquent au processeur quelle opération effectuer.
4. **Mémoire :** La mémoire est utilisée pour stocker les données et le code du programme.
5. **Assemblage :** Le processus de conversion du code assembleur en code machine s'appelle l'assemblage.

**Les étapes pour écrire un programme en assembleur**

1. **Écrire le code :** Utiliser un éditeur de texte pour écrire les instructions en assembleur.

2. **Assembler le code :** Utiliser un assembleur pour convertir le code en code machine.
3. **Lier le code :** Si nécessaire, lier le code avec d'autres fichiers pour créer un programme exécutable.
4. **Exécuter le programme :** Exécuter le programme sur l'ordinateur cible.

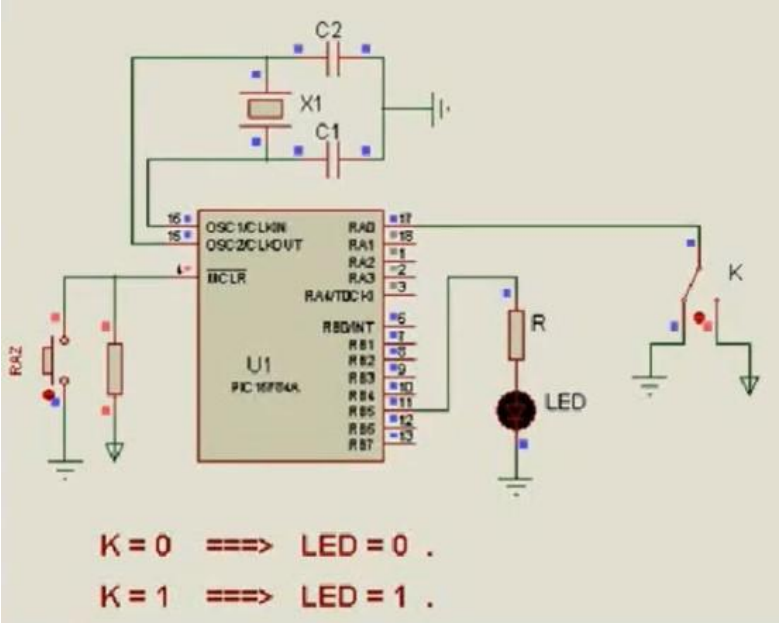
**Partie 1:**

➢ Réaliser un circuit simple avec un PIC16F84 où :

- Une LED est connectée sur la broche RB5.
- Un interrupteur est connecté sur la broche RA0.
- Lorsque vous appuyez sur l'interrupteur, la LED s'allume.

Donc:

✓ **RA0 :** Entrée numérique (interrupteur)
✓ **RB5 :** Sortie numérique (LED)




➢ **Rocopier le code 1 et le code 2 sur MPLAB et corriger les erreurs**

**CODE 1:**

```
; Configuration du PIC,

BSF STATUS, RPO
MOVLW B'00000000'
MOVWF TRISB
MOVLW B'00000'
MOVWF TRISA
BCF STATUS,RP0

;Test de l'interrupteur K
Test
BTFSS PORTA,0
GOTO ZERO
GOTO UN
; Eteindre la LED : Ecriture de ZERO
ZERO
MOVLW B'11111111'
MOVWF PORTB
GOTO TEST

; Allumer la LED : Ecriture de UN
UN
MOVLW B'11111111'
MOVWF PORTB
GOTO TEST
END
```

- **CODE 2:**

```
; Configuration des ports
BSF STATUS, RP0 ; Banque de registres haute
MOVLW B'00000000' ; Configuration de RA0 en entrée
MOVWF TRISA
MOVLW B'00000001' ; Configuration de RB5 en sortie
MOVWF TRISB ;
BCF STATUS, RP0 ; Banque de registres basse

;Programme principal
LOOP
BTFSC PORTA, 0 ; Si RA0 est à 1 (interrupteur enfoncé)
BSF PORTB, 5 ; Allume la LED sur RB5
GOTO LOOP
BCF PORTB, 5 ; Éteint la LED
```

GOTO LOOP

**Partie2:**

Ecrire un programme en assembleur sous MPLAB qui permet d'allumer une LED Rouge.

Ajouter deux autres Led (verte et orange) et faire cilgnoter les trois Led

Nous souhaitons créer un programme qui :

- Allume successivement une LED rouge, verte et orange.
- Utilisez une temporisation pour créer un effet de clignotement .