

TD 3 : Application des jeux d'instructions PIC16F84

- **Configuration des ports :** 13 lignes d'entrées/sorties, réparties en un port de 5 lignes (PORTA) et un port de 8 lignes (PORTB)

La configuration des bits PORTA se fait dans le registre TRISA

- Configurer un bit du **PORTA** comme entrée → mettre **1** dans le bit correspondant dans TRISA
- Configurer un bit du **PORTA** comme sortie → mettre **0** dans le bit correspondant dans TRISA

TRISA	X	X	X	1	0	1	1	0
PORTA	X	X	X	RA4 (E)	RA3 (S)	RA2 (E)	RA1 (E)	RA0 (S)

La configuration des bits PORTB se fait dans le registre TRISB

- Configurer un bit du **PORTB** comme entrée → mettre **1** dans le bit correspondant dans TRISB
- Configurer un bit du **PORTB** comme sortie → mettre **0** dans le bit correspondant dans TRISB

TRISB	1	1	1	1	0	0	0	1
PORTB	RB7 (E)	RB6 (E)	RB5 (E)	RB4 (E)	RB3 (S)	RB2 (S)	RB1 (S)	RB0 (E)

1 : entrée (E), **0** : sortie (S)

- **Etapes de la configuration :**

- ❖ Accéder à la Bank 1
- ❖ Déterminer le mot à mettre dans le registre TRISx
- ❖ Mettre ce mot dans le registre de travail W
- ❖ Transférer le contenu de W dans le registre TRISx
- ❖ Accéder à la Bank 0, pour pouvoir accéder au PORTx

▪ **Le registre STATUS :**

7	6	5	4	3	2	1	0
IRP	RP1	RP0	/TO	/PD	Z	DC	C

Le registre d'état contient 8 bits :

- ✓ 5 bits, témoins (drapeaux) caractérisant le résultat de l'opération réalisée par la CPU (lecture seule)
 - **/TO** : (Time Out) débordement du timer WDT
 - **/PD** : (Power Down) caractérise l'activité du chien de garde WDT
 - **Z** : (Zéro) résultat nul pour une opération arithmétique et logique (résultat=0 → Z=1, résultat≠0 → Z=0)
 - **DC** : (Digit Carry) retenue sur un quartet (4bits)
 - **C** : (carry) retenue sur un octet (8 bits)
(Résultat > 0 → C=0, résultat < 0 → C=1)
- ✓ 1 bit : **RP0** : sélection de bank (bank 0 ou 1)
(bit 5 du registre d'état =0 → bank 0, bit 5 du registre d'état =1 → bank 1)
- ✓ 2 bits : RP1 et IRP : pas pour le pic16F84

- **Le registre W** : registre de travail (accumulateur) : il est utilisé pour faire toute sorte de calcul, le résultat d : d : c'est le registre de destination, avec deux variantes :

d=0 résultat est placé dans W, **d=1** : résultat placé dans le registre f

▪ **Cycle d'exécution d'une instruction :**

- Recherche de l'instruction
- Décodage de l'instruction
- Exécution
- Ecriture du résultat

Exercice 1

1. Mettre 0 dans W
2. Charger la valeur 20_H dans W
3. Charger la valeur de W à l'adresse 020_H
4. Charger la valeur à l'adresse 020_H dans W
5. Charger la valeur 20 dans W

Exercice 2

1. Mettre à 1 le bit 5 de la valeur à l'adresse 003_H
2. Décrémenter la valeur à l'adresse 020_H, et stocke le résultat à la même adresse
3. Décrémenter la valeur à l'adresse 020_H, et stocke le résultat dans W
4. Décrémenter la valeur à l'adresse 020_H et stocke le résultat dans l'adresse 021_H
5. Tester si le bit C du registre d'état vaut 1

Exercice 3

Ecrire un programme qui additionne deux valeurs en RAM « val1 et val2 », et met le résultat dans une variable 8bits « Res »

Exercice 4

Configurer le PORT B en entrée, lire le contenu du PORT B

Mettre le résultat dans la case mémoire d'adresse 0C_H

Exercice 5

Ecrire pour l'opération mathématique suivante, le programme qui la réalise

$$(100+15-31)/4$$

Exercice 6

Ecrire un programme PIC16f84 qui donne le résultat de multiplication de 5x10 et place le résultat dans l'adresse 0x0C.