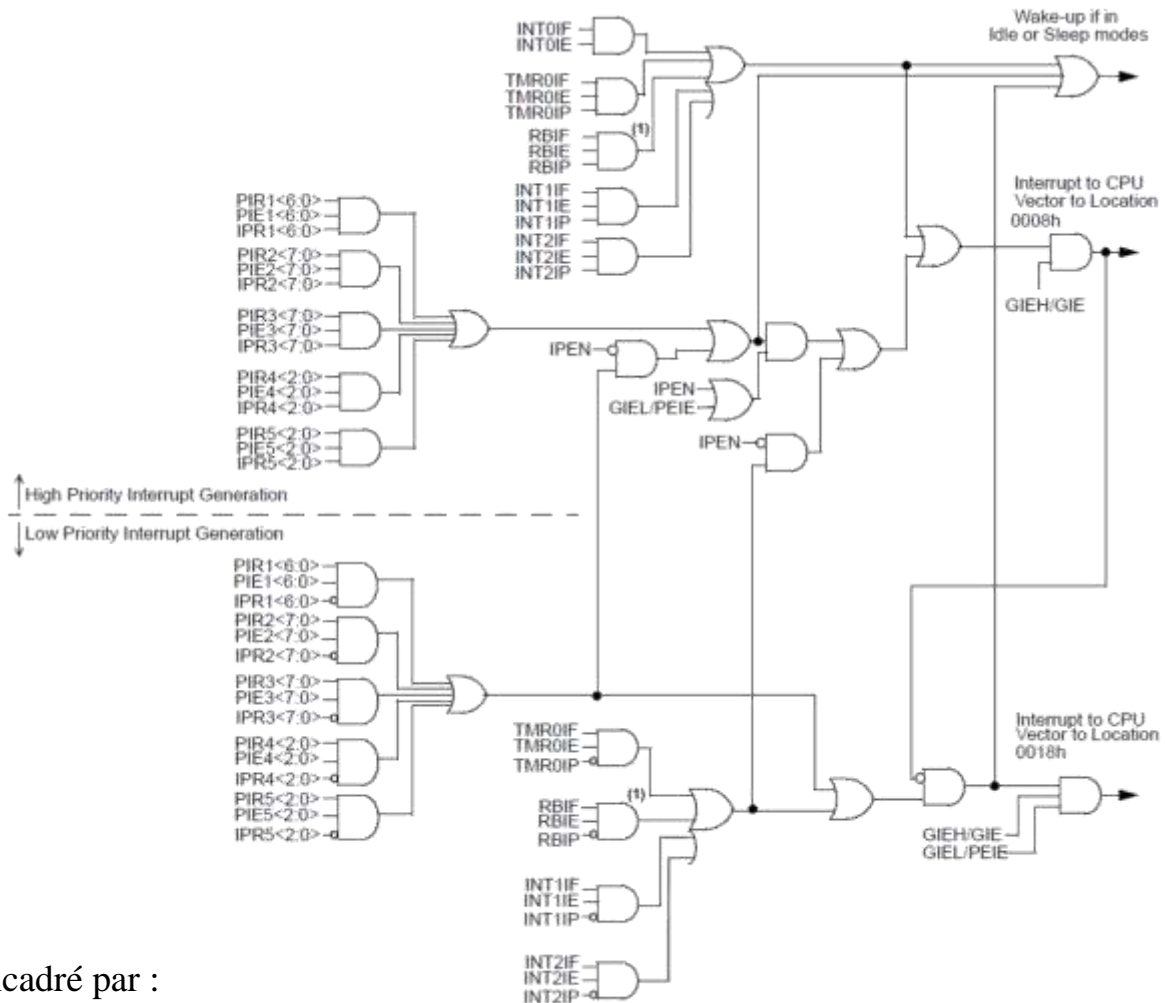


TP 3 – Informatique embarquée

GESTION DES INTERRUPTION



Encadré par :

EMHARRAF Mohamed

1. Objectifs

L'objectif principal de ce TP est de comprendre et de tester le fonctionnement des différentes interruptions externes disponibles sur le microcontrôleur PIC18F45K22. Les étudiants apprendront à configurer et exploiter les interruptions INT et RBI (Interrupt-on-Change des ports RB) pour gérer des événements externes en temps réel.

2. Exemple: (interruption INT0)

Dans cet exemple, nous exploiterons l'interruption **INT0** pour faire clignoter le port **C**.

1. **Lancer le compilateur mikroC PRO for PIC** : ouvrez le logiciel et préparez-vous à créer un nouveau projet.
2. **Créer un nouveau projet** : sélectionnez l'option "New Project". Configurez les paramètres comme suit :
 - Microcontrôleur : **PIC18F45K22**
 - Fréquence : **8 MHz**
 - Spécifiez également le chemin d'enregistrement du projet pour une organisation optimale.
3. **Écriture du code** : nous allons maintenant écrire notre programme en C. La première étape consiste à initialiser les registres de configuration nécessaires pour activer et gérer l'interruption **INT0** ainsi que le port **C**.

La première chose à faire est d'initialiser les registres de configuration :

```
TRISC = 0;
ANSELB=0;
INTCON2|= 0x40; // INTEDG0=1
INTCON |= 0x90; //GIE=1, INT0IE = 1
```

Dans le programme principal on fixe la valeur par défaut de port (0 par exemple)

```
LATC = 0x00;
```

Dans la fonction d'interruption, on doit clignote le port et initialiser le flag d'interruption

```
void interrupt()
{ LATC=~LATC;
  delay_ms(1000);
  INTCON &=0xFC; //INT0F=0 }
```

Le programme complet prend la forme suivante :

```
void main()
{ TRISC = 0;
  INTCON2|= 0x40; // INTEDG0=1
  INTCON |= 0x90; //GIE=1, INT0IE = 1
  while(1)
```

```

    { LATC = 0x00; }
void interrupt()
{ LATC=~LATTC;
  delay_ms(1000);
  INTCON &=0xFC; //INT0F=0}

```

Exécution du programme :

Charger le programme compilé sur la carte, puis cliquer sur le bouton RB0 pour inverser la valeur de PORTC pendant une seconde.

3. Manipulations

1. Donner le programme Permet d'incrémenter la valeur de PORTC a chaque 0,5s, et mettre à zéro le port si l'interruption INT0 est activé.
2. Donner le programme Permet de clignoter de PORTC au rythme de 1s, et charger le contenu de PORTD dans le PORTC si l'interruption RBI est activé.
3. On désire commander deux LEDs à l'aide de deux sources d'interruption INT0 et INT1, concevoir le programme permet d'allumer les LEDs PORTD pour une durée de 1s, si INTO active et les allumées pour une durée de 4s suite à l'activation de INT1 (avec même niveau de priorité).
4. On désire commander les LED de PORTA par plusieurs endroits à l'aide de l'interruption RBI, Une impulsion sur l'un des boutons poussoir de RBI, les LED s'allume, une deuxième impulsion les LED s'éteint.
5. Programme permet de calcule la somme de deux variables (var1, var2) et afficher résultat sur le PORTD.

L'activation de l'interruption INT0 mettre à zéro les variable var1 et var2.

L'activation de l'interruption INT1 charge la valeur de PORTC dans VAR1

L'activation de l'interruption INT2 charge la valeur de PORTA dans VAR2

L'interruption INT0 doit être prioritaire par rapport au INT1 et INT2.