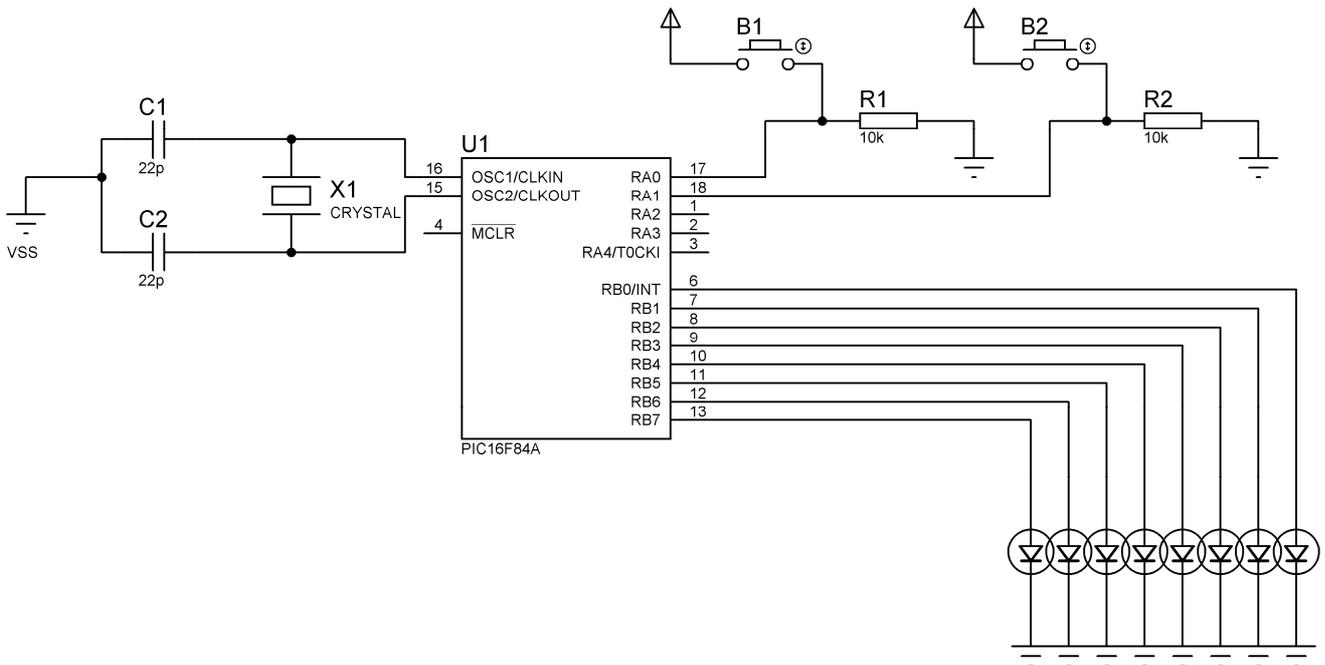


# TRAVAUX DIRIGES N° : 1

## AFFICHAGE DES DONNEES

### Exercice 1.1 :

Soit le montage suivant permettant de commander 8 diodes LED :



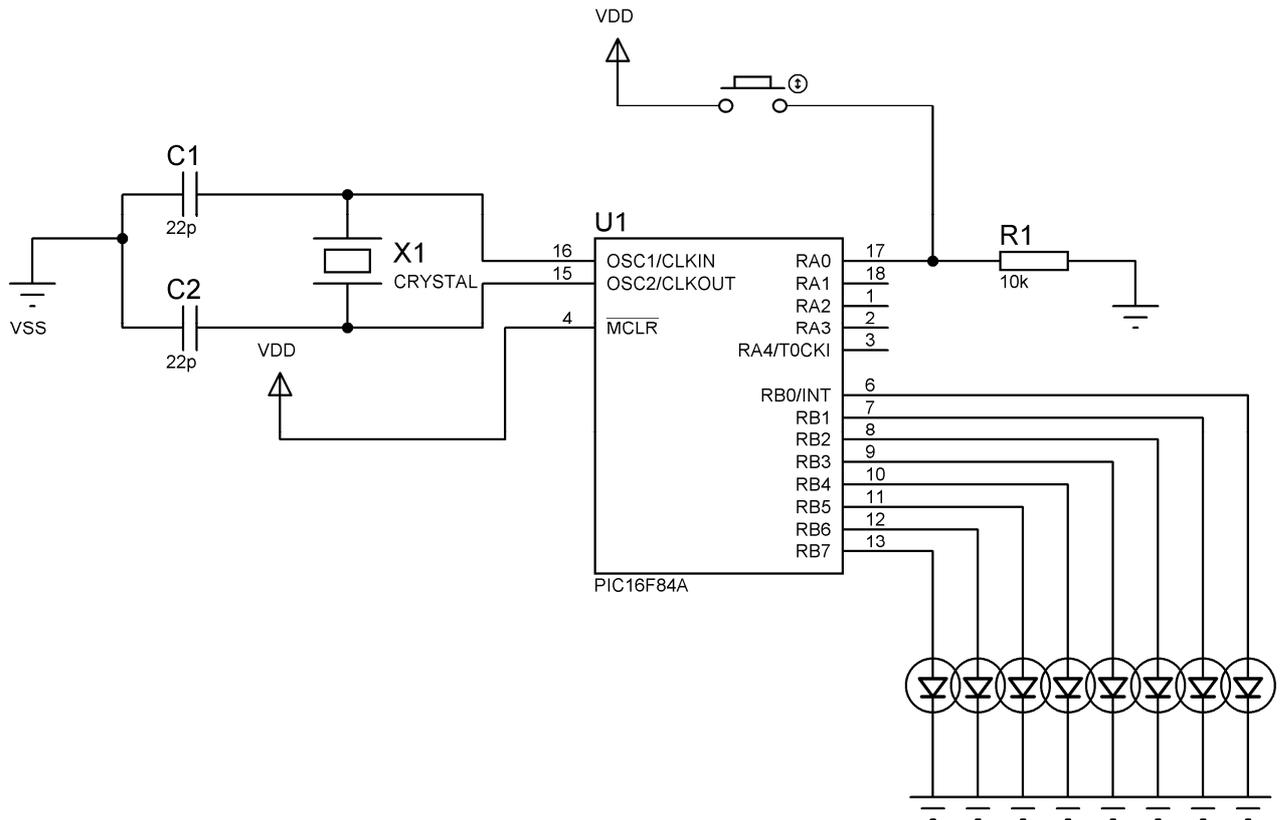
On veut maintenant réaliser plusieurs animations sur les LEDs commandées par les boutons poussoirs B1 et B2 selon le tableau suivant :

N° séquence	B1	B2	Séquence
0	1	1	L0-L1-L2-L3-L4-L5-L6-L7
1	1	0	L7-L6-L5-L4-L3-L2-L1-L0
2	0	1	L7/L0-L6/L1-L5/L2-L4/L3
3	0	0	L0/L1-L2/L3-L4/L5-L6/L7

Ecrire un programme en C permettant de commander les diodes LED.

### Exercice 1.2 :

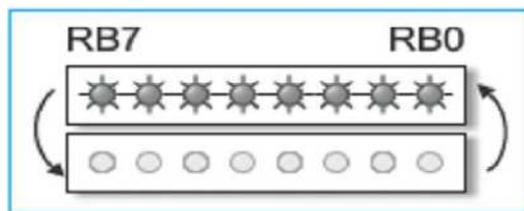
Soit le montage suivant permettant de commander 8 diodes LED :



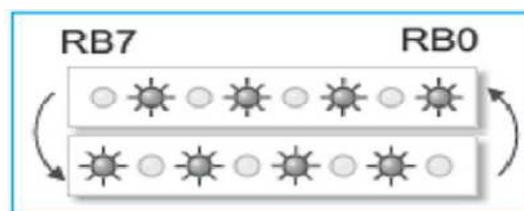
**Question 1 :**

On propose de faire clignoter LEDs branchées au port B du microcontrôleur de deux façons différentes :

- Si RA0 = 0, tous les LEDs clignotent avec une période d'une seconde.



- Si RA0 = 1, les LEDs doivent clignoter chaque 500 ms comme indiqué dans la figure ci-dessous.



Ecrire un programme en C permettant de commander les diodes LED.

**Question 2 :**

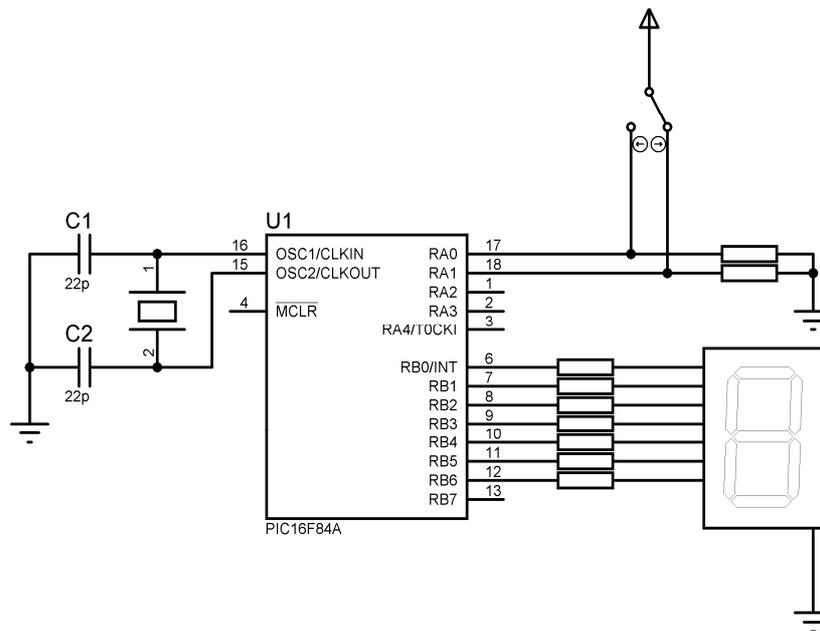
- Si RA0=0, les 8 diodes LED clignotent pendant 2 secondes,
- Si RA0=1, on obtient le cycle répétitif suivant :

Diodes LED allumés	Durée
D0D1	1s
D2D3	2s
D4D5	3s
D6D7	4s
Aucune diode	1s

Ecrire un programme en C permettant de commander les diodes LED.

**Exercice 1.3 :**

On désire commander un afficheur à 7 segments afin de réaliser un compteur modulo 10 à l'aide d'un microcontrôleur 16F84A.



1. Ecrire un programme C qui permet de réaliser le fonctionnement suivant :

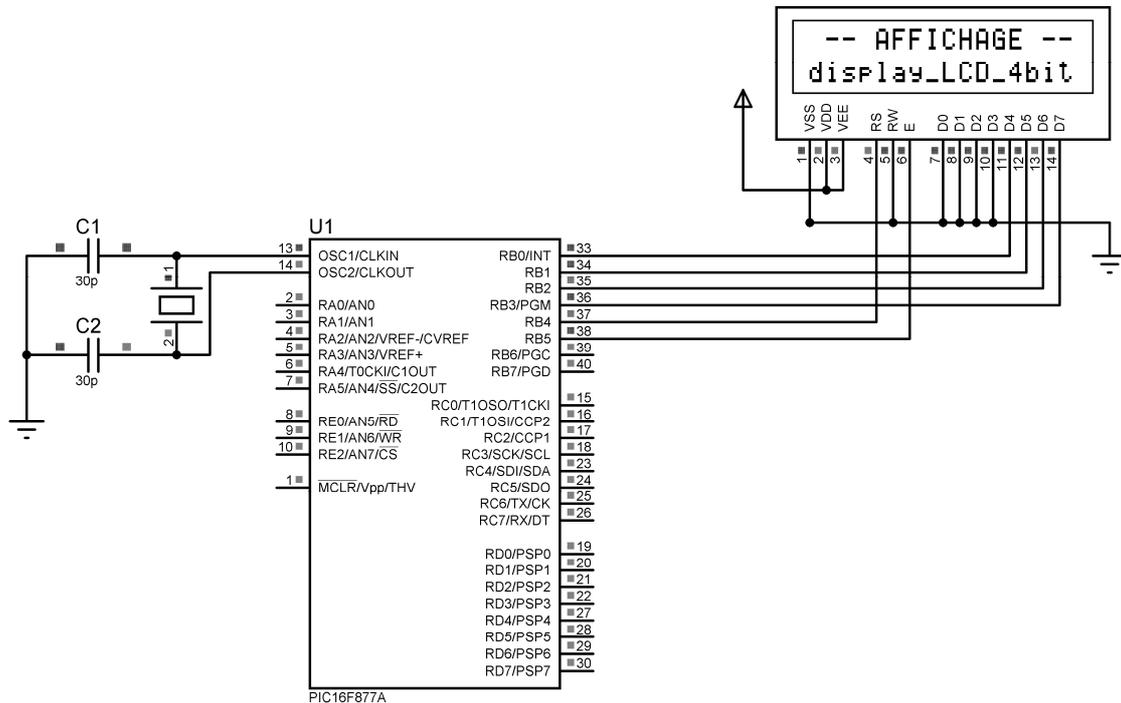
- **Compteur modulo 10** commandé par la position 1 du switcher (RA0).
- **Décompteur modulo 10** commandé par la position 2 du switcher (RA1).

2. Ecrire un programme C qui permet de réaliser le fonctionnement suivant :

- **Compteur** des nombres pairs commandé par la position 1 du switcher.
- **Compteur** des nombres impairs commandé par la position 2 du switcher.

**Exercice 1.4 :**

Soit le montage suivant :

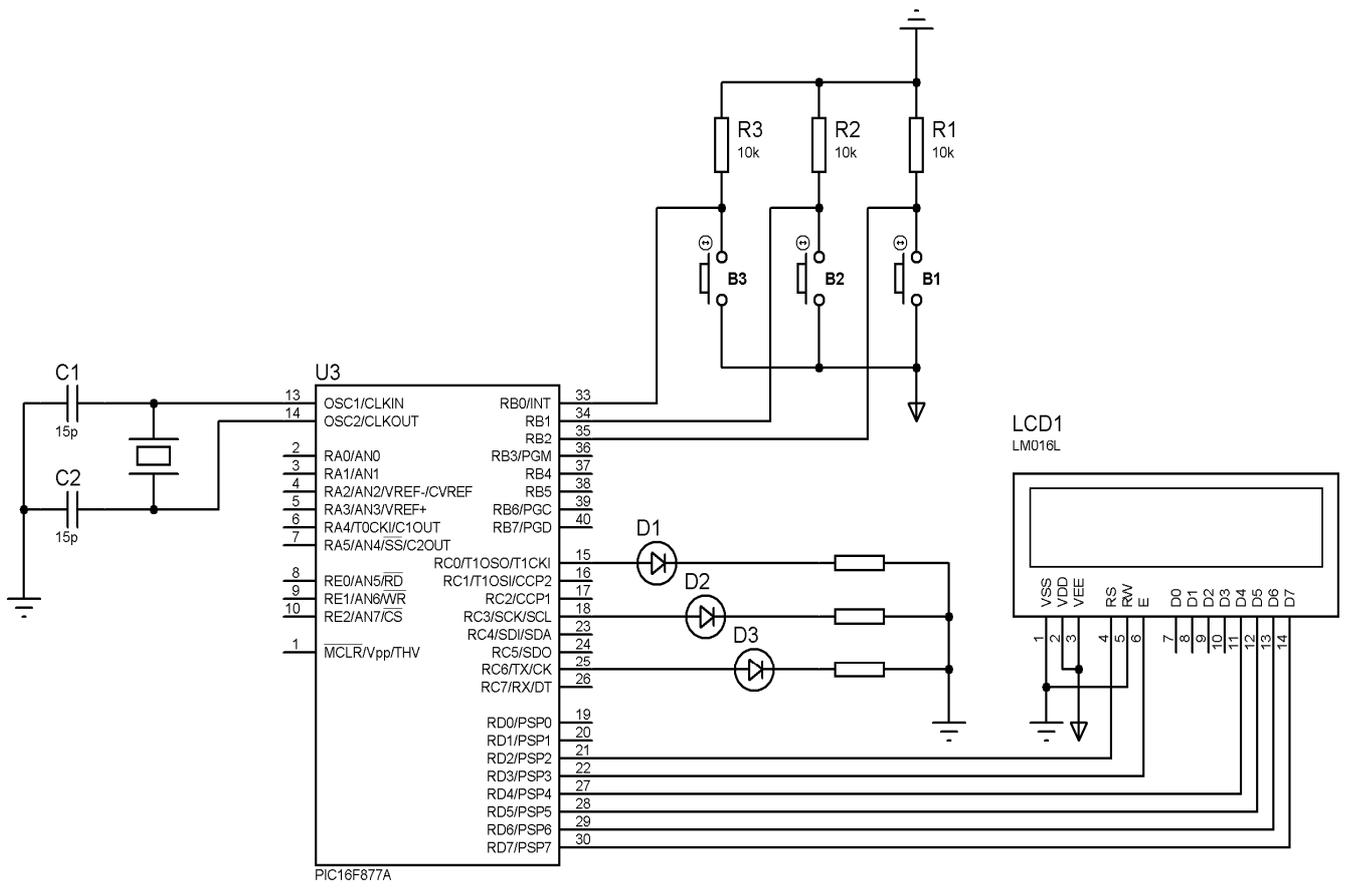


Ecrire un programme en C qui affiche sur l'écran LCD le message ci-dessus.

**Exercice 1.5 :**

Soit le montage de la figure ci-après. Ecrire un programme en C qui répond au fonctionnement suivant :

- Un appui sur le bouton B1 seul, l'afficheur affiche 'BONJOUR' et la diode D1 clignote.
- Un appui sur le bouton B2 seul, l'afficheur affiche 'BONSOIR' et la diode D2 clignote.
- Un appui sur le bouton B3 seul, l'afficheur affiche 'SALAM' et la diode D3 clignote.
- Pour les autres cas, rien ne se passe.



## TRAVAUX DIRIGES N° : 2

### SYSTEMES D'ENTREE DES DONNEES (1)

**Exercice 2.1 :**

Soient les équations suivantes d'un système logique combinatoire :

$$S_1 = A + B + \bar{C}.\bar{B},$$

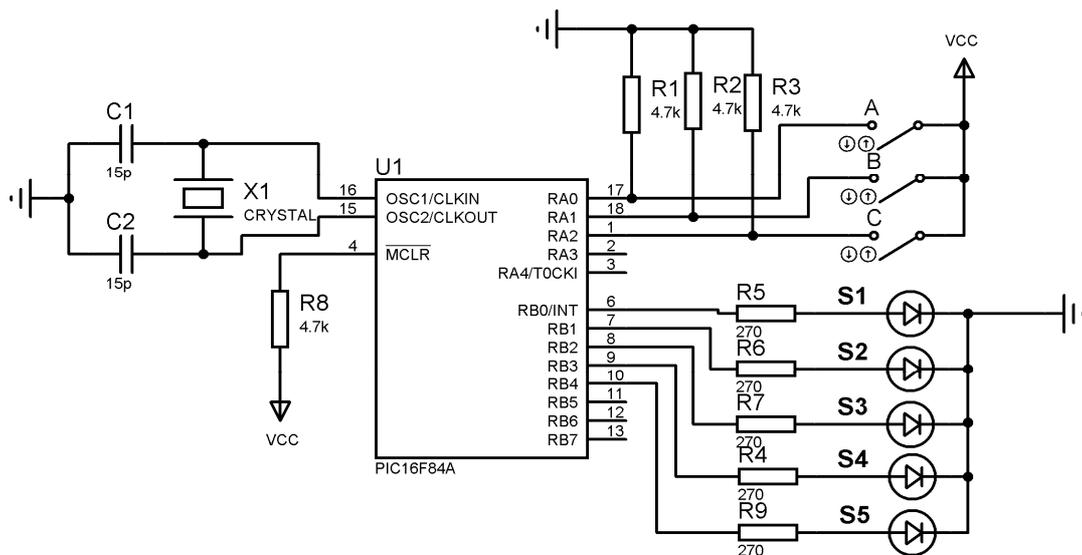
$$S_2 = A.\bar{B} + \bar{C}.B + C,$$

$$S_3 = \overline{A + B + C}.\bar{B}$$

$$S_4 = A + \bar{C}.B,$$

$$S_5 = B.\bar{C}.A$$

On propose le schéma de simulation suivant :



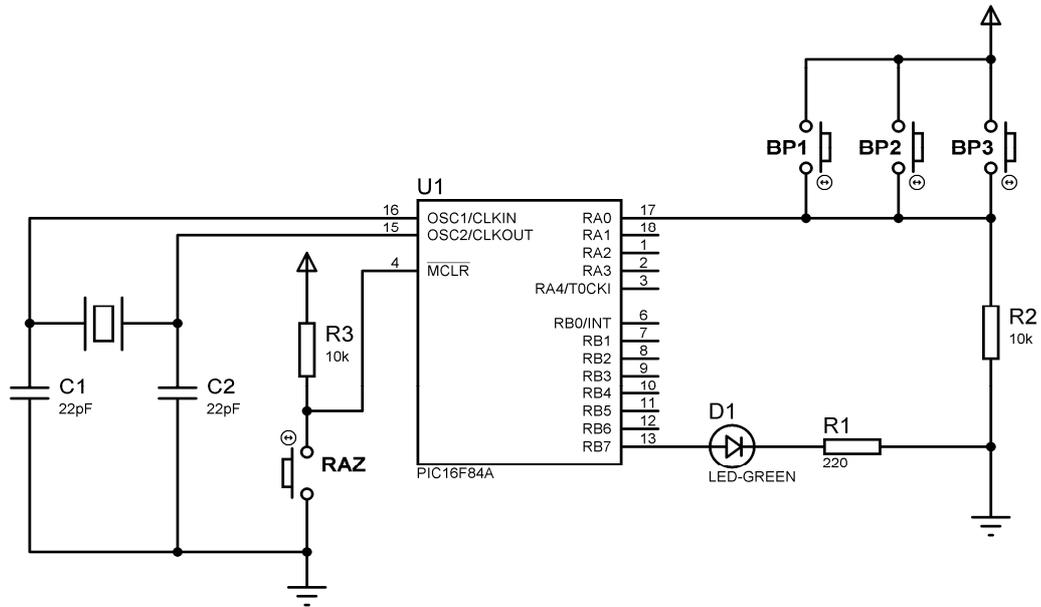
Ecrire un programme en C qui permet d'implanter le système combinatoire précédent.

**Exercice 2.2 :**

On désire commander une LED par plusieurs endroits à l'aide d'un microcontrôleur 16F84A via des boutons poussoir BPi (BP1, BP2 et BP3).

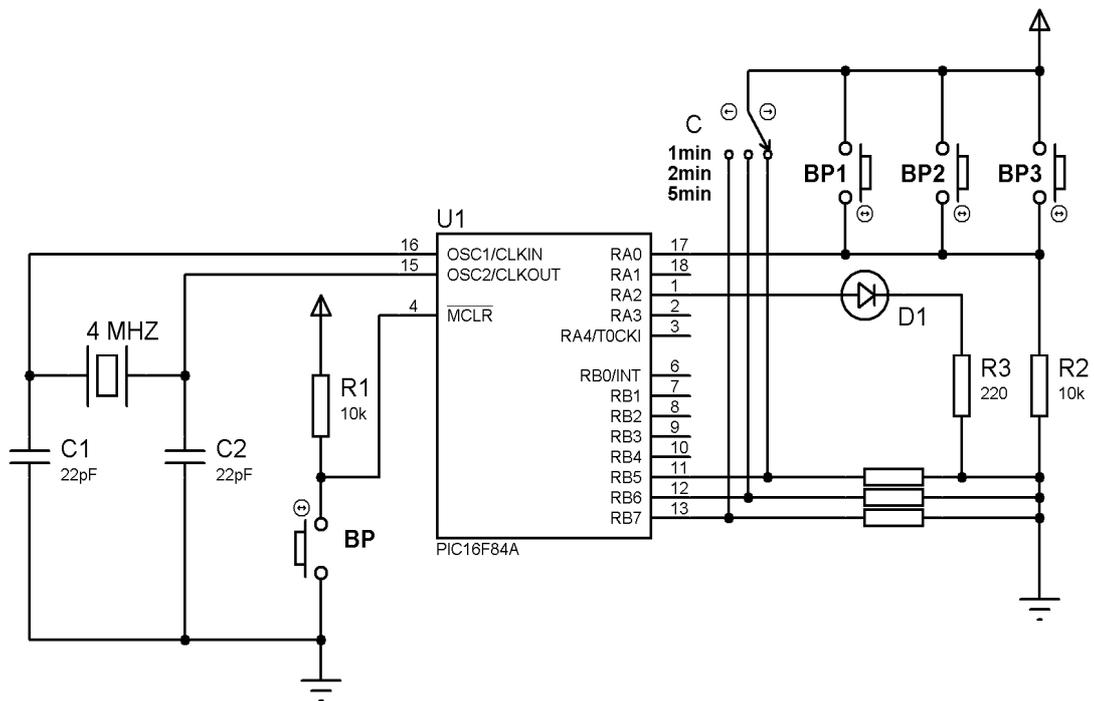
Une impulsion sur l'un des boutons poussoir BPi, la LED D s'allume, une deuxième impulsion la LED s'éteint.

Ecrire un programme C qui répond au cahier de charge.



**Exercice 2.3 :**

On désire réaliser une minuterie d’escalier à 3 temps réglable (1min, 3min, 5min) à l’aide d’un microcontrôleur 16F84A.



Le principe de fonctionnement est le suivant :

- Une impulsion sur l’un des boutons poussoir BP<sub>i</sub>, la LED D s’allume pendant un temps T puis s’éteint.
- Deux impulsions successives est sans effet.
- Le temps est réglable par un commutateur C à trois positions.

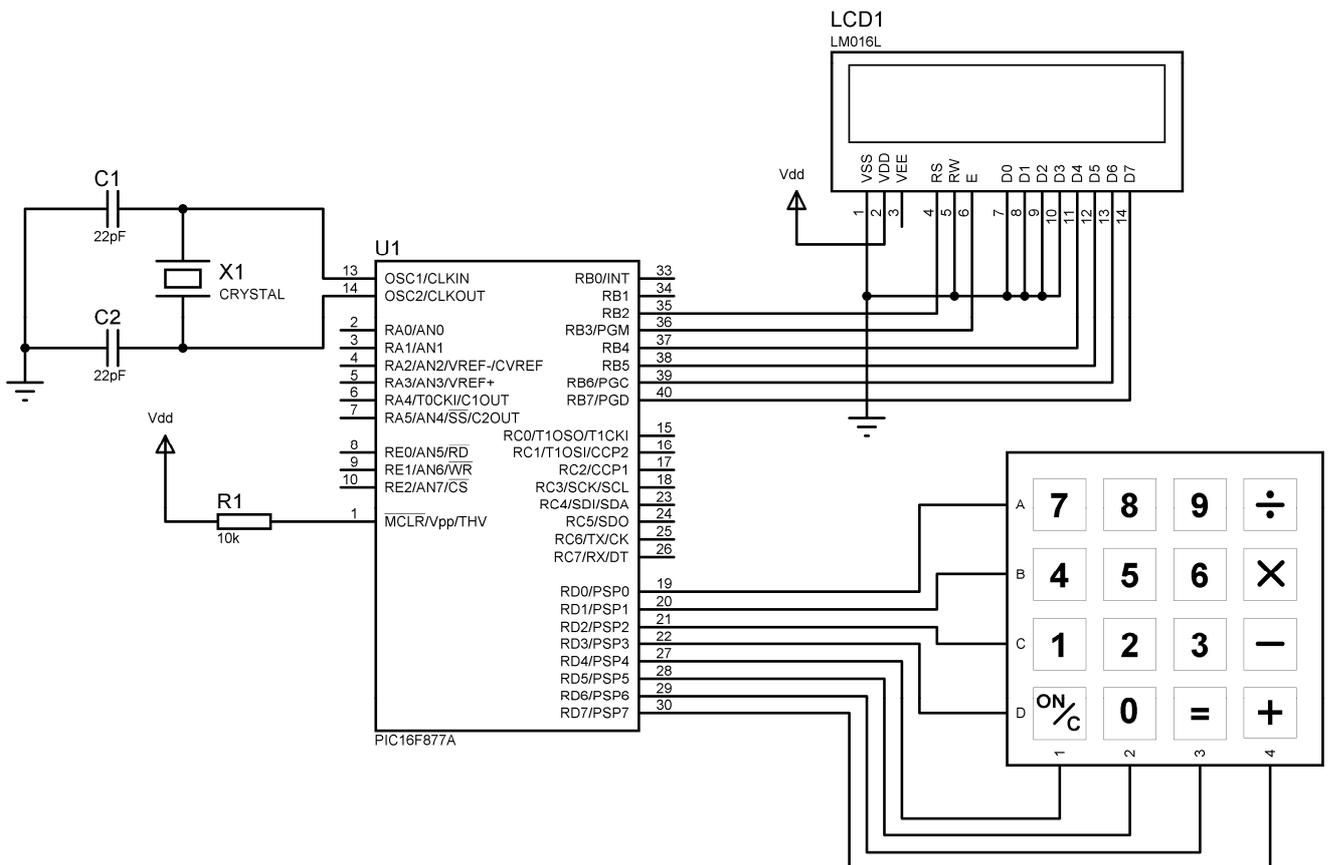
Ecrire un programme C qui répond au cahier de charge.

## TRAVAUX DIRIGES N° : 3

### SYSTEMES D'ENTREE DES DONNEES (2)

#### Exercice 3.1 :

Ce circuit illustre l'utilisation d'un clavier et un écran LCD alphanumérique.



Ecrire un programme qui lit un caractère à partir du clavier et l'affiche sur l'écran LCD.

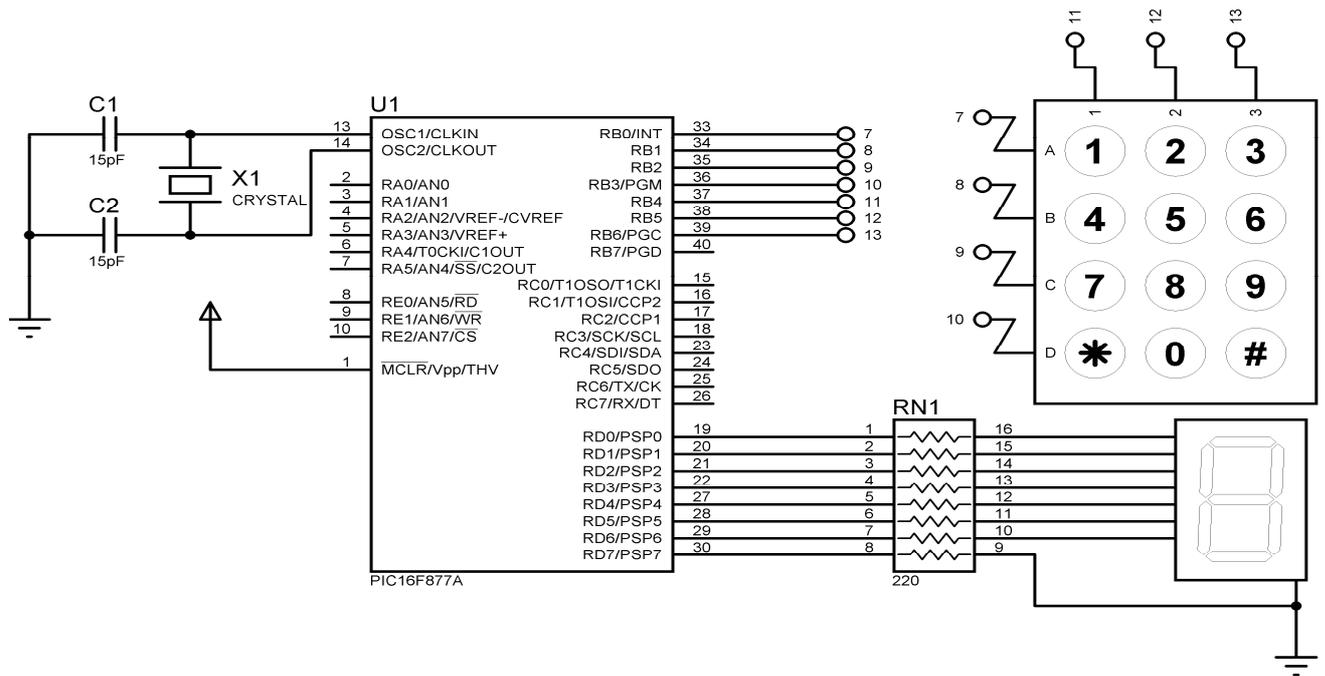
#### Exercice 3.2 :

On désire commander un afficheur à 7 segments à l'aide d'un microcontrôleur 16F877A via un clavier à 12 touches.

Le principe de fonctionnement est le suivant :

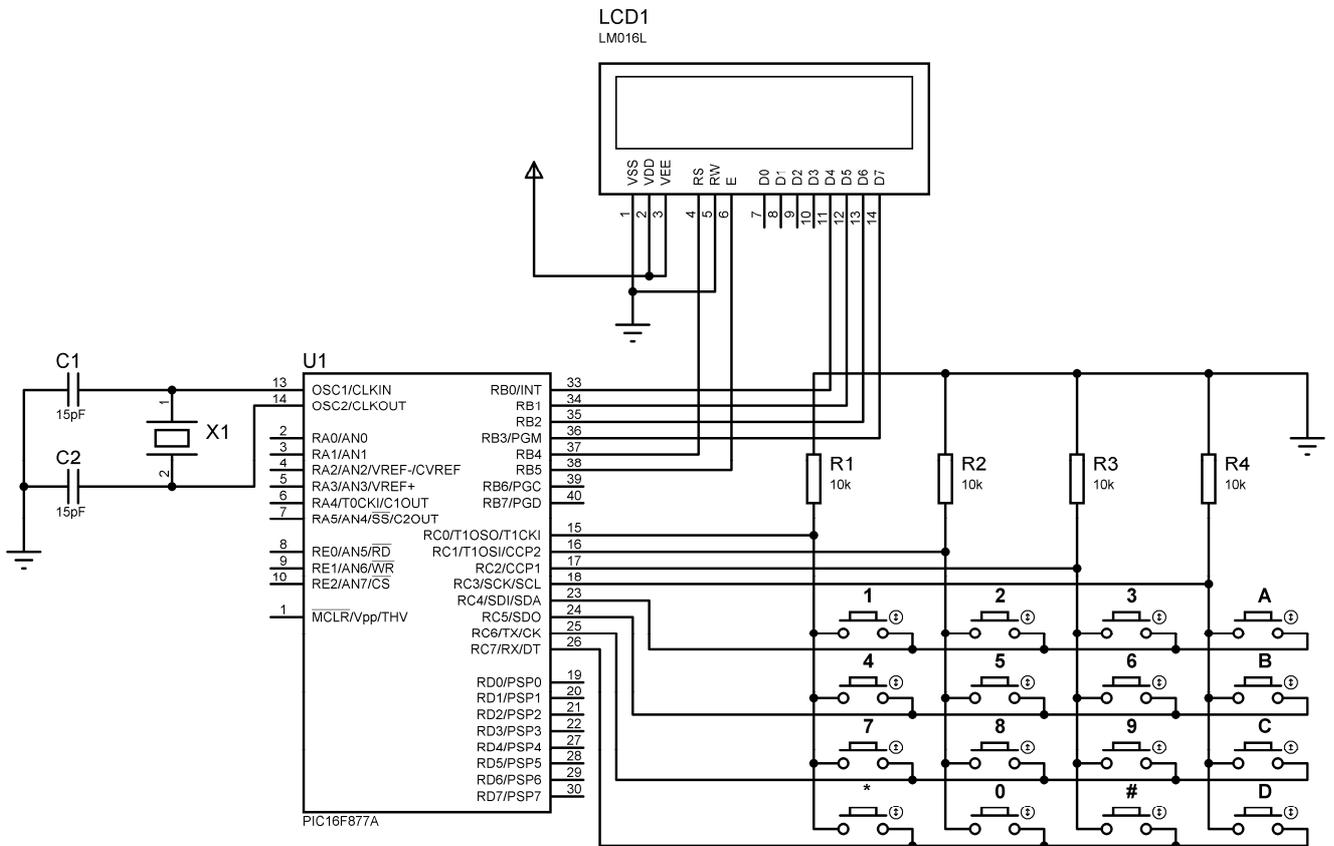
- L'appui sur une touche de 0 à 9 affiche sa valeur sur l'afficheur 7 segments.
- L'appui sur la touche (\*) affiche la lettre E sur l'afficheur 7 segments.
- L'appui sur la touche (#) affiche la lettre F sur l'afficheur 7 segments.

Ecrire un programme qui répond à ce cahier des charges.



**Exercice 3.3 :**

Soit le circuit suivant composé d'un microcontrôle PIC 16F877A muni d'un afficheur LCD et 16 boutons poussoirs jouant le rôle d'un clavier matriciel :



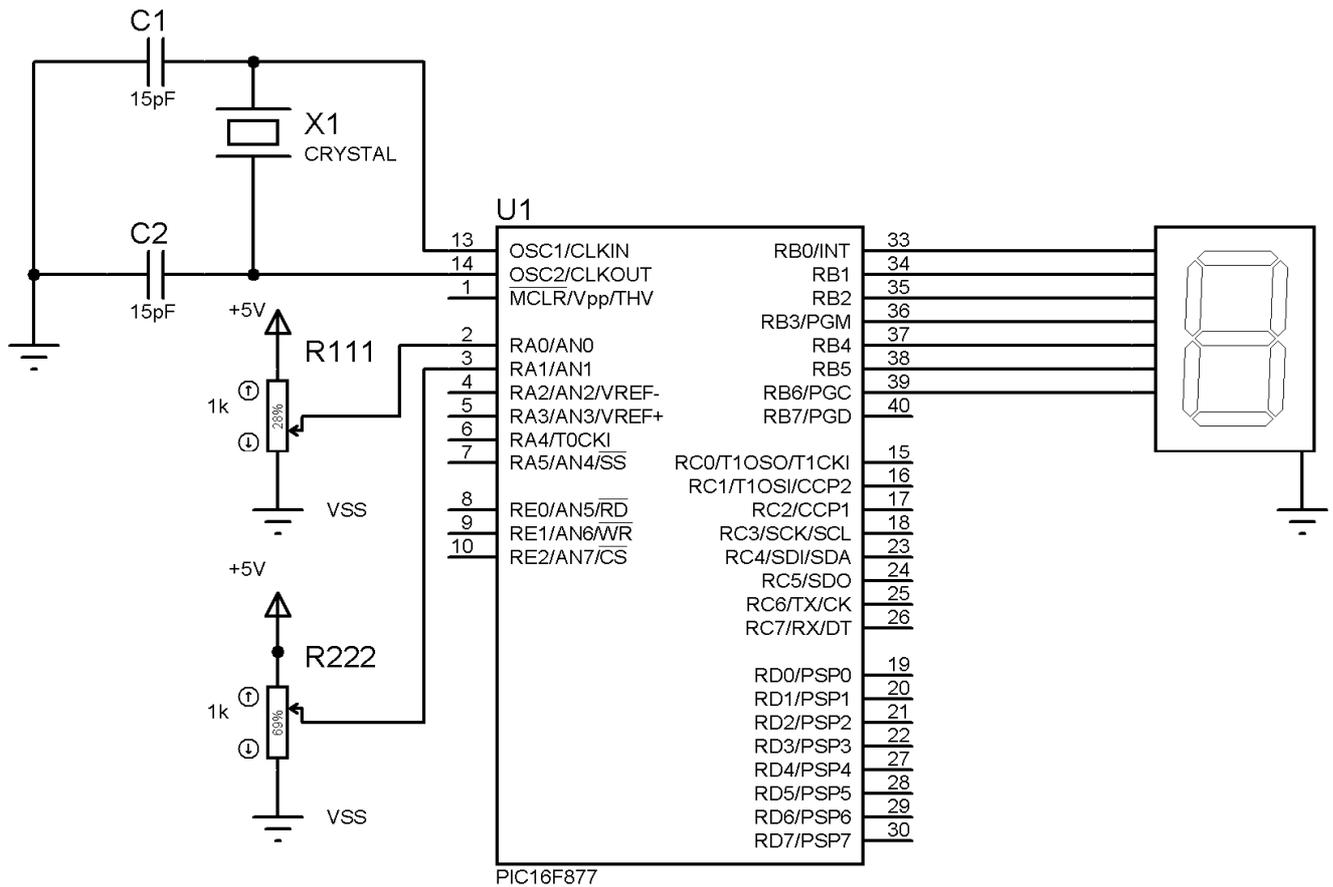
Ecrire un programme qui lit un caractère à partir du clavier et l'affiche sur l'écran LCD.

## TRAVAUX DIRIGES N° : 4

### CONVERTISSEUR ANALOGIQUE NUMERIQUE

#### Exercice 4.1

Soit le montage suivant :



On veut réaliser un comparateur entre les valeurs de deux **tensions analogiques** appliquées sur chacune des pins RA0 et RA1 via deux potentiomètres.

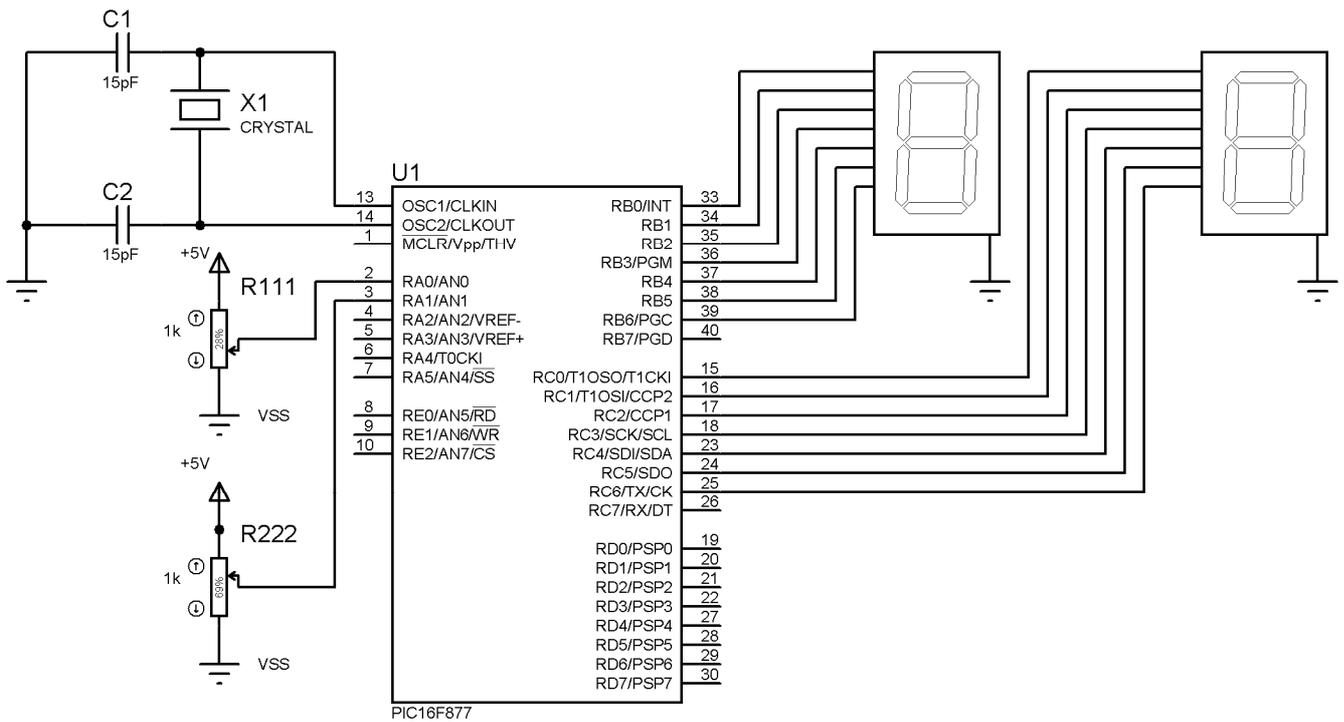
Ecrire un programme en C qui répond au cahier des charges suivant :

- Si tension 0 > tension 1 **alors** afficher la lettre **S**.
- Si tension 0 = tension 1 **alors** afficher la lettre **E**.
- Si tension 0 < tension 1 **alors** afficher la lettre **I**.

L'affichage des résultats se fera sur un afficheur 7 segments comme le montre la figure ci-dessus.

#### Exercice 4.2

Soit le montage suivant :



Ecrire un programme en C qui répond au fonctionnement suivant :

- Si  $RA0 < RA1$ , on doit faire un comptage modulo **20**.
- Si  $RA0 > RA1$ , on doit faire un décomptage modulo **20**.
- Si  $RA0 = RA1$ , on affiche la lettre H sur les deux afficheurs.

L’affichage des résultats se fera sur un afficheur 7 segments.

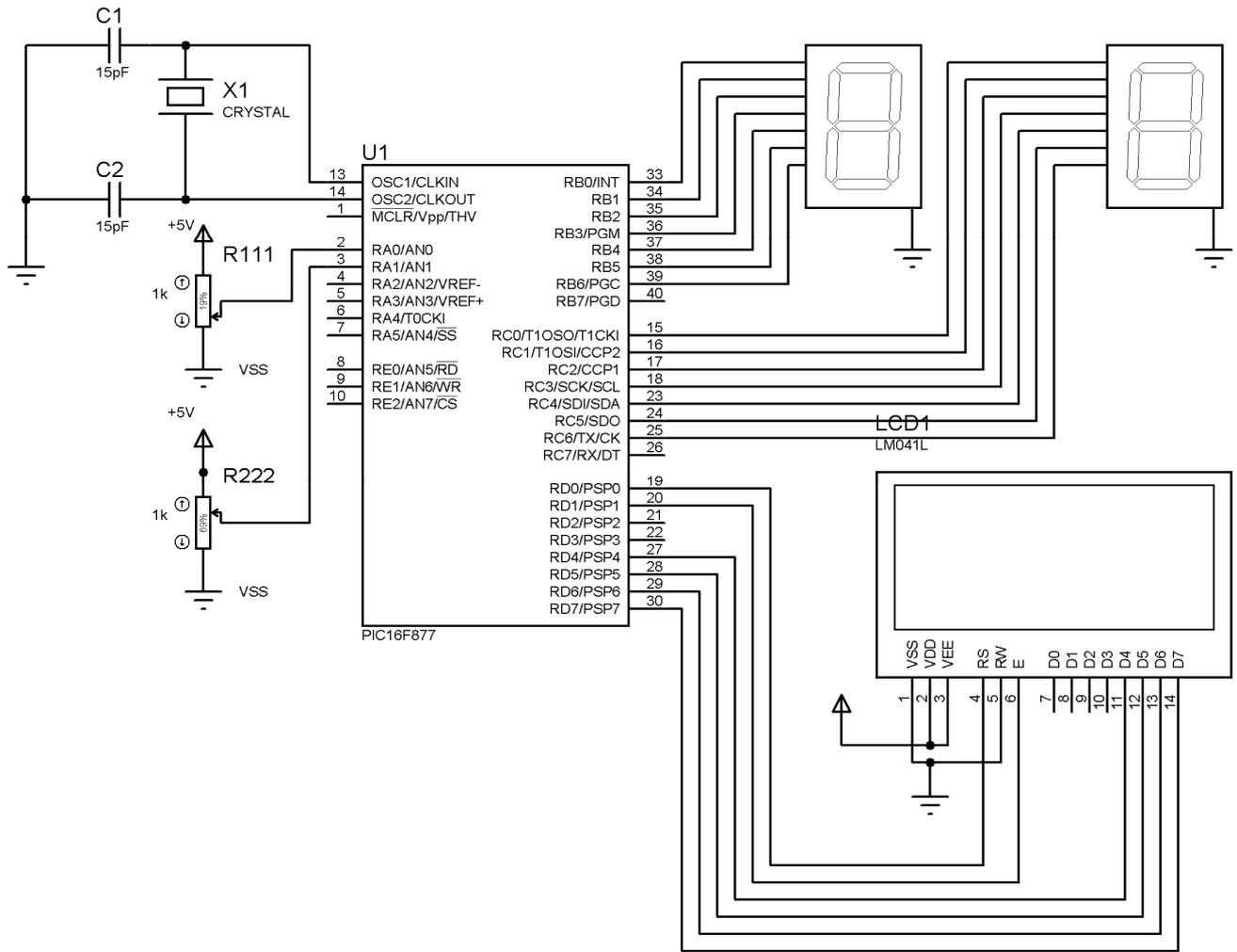
**Exercice 4.3 :**

Soit le montage ci-après. Ecrire un programme en C qui répond au fonctionnement suivant :

- Si tension 0 > tension 1 **alors** afficher sur chaque afficheur 7 segments la lettre **S**.
- Si tension 0 = tension 1 **alors** afficher sur chaque afficheur 7 segments la lettre **E**.
- Si tension 0 < tension 1 **alors** afficher sur chaque afficheur 7 segments la lettre **I**.

Le programme doit afficher aussi sur le LCD 16x4 :

- Ligne 1 : le message « Comparateur »,
- Ligne 2 : la valeur de la tension 0,
- Ligne 3 : la valeur de la tension 1,
- Ligne 4 : la valeur de la tension **maximale**.

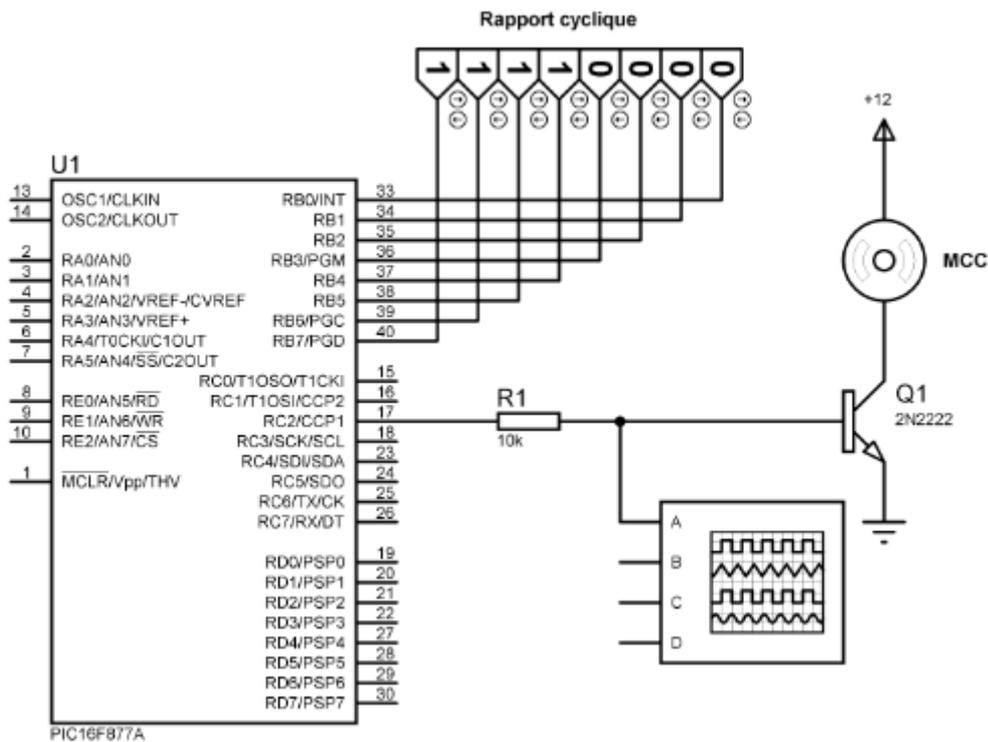


## TRAVAUX DIRIGES N° : 5

### MODULATION A LARGEUR D'IMPULSION

#### Exercice 5.1 :

Pour faire varier la vitesse d'un moteur à courant continu, on vous propose le circuit de la figure ci-dessous :



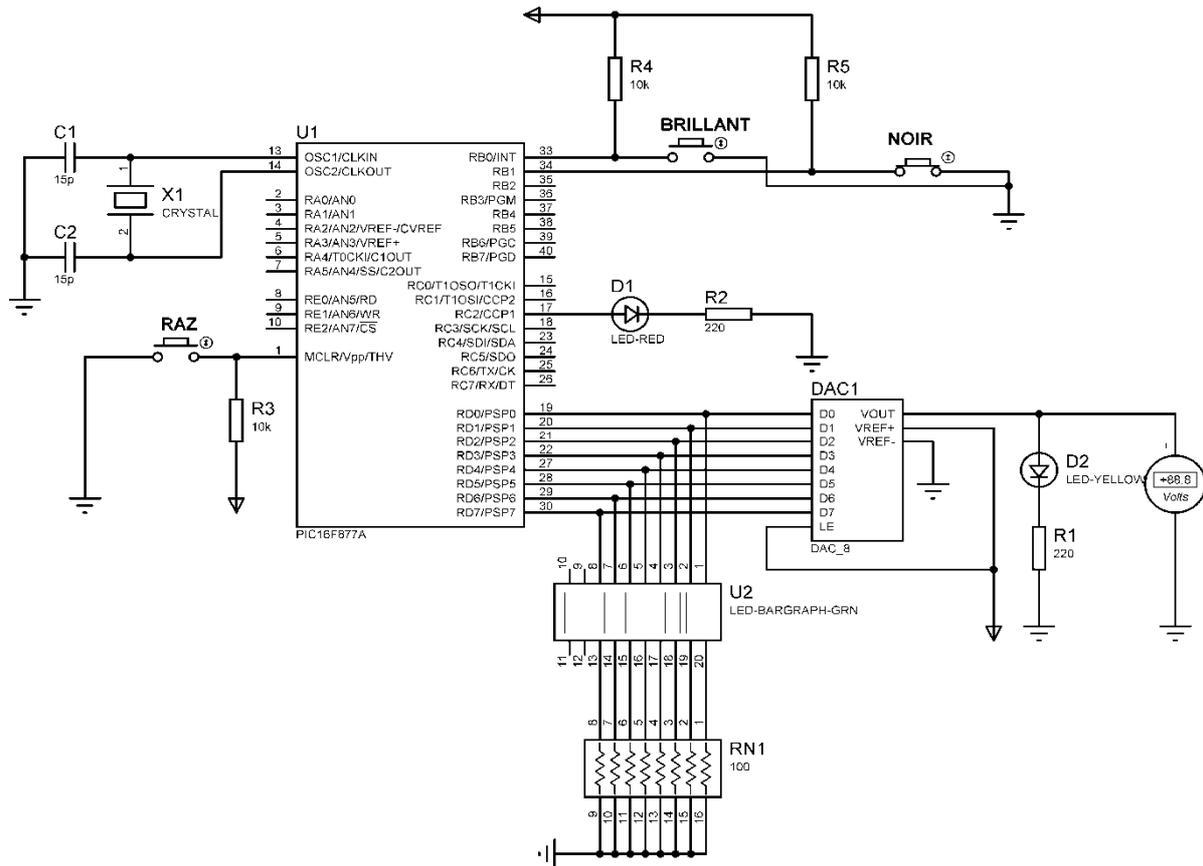
Ecrire un programme qui lit une valeur numérique sur le PORTB sous forme d'un nombre codé sur 8 bits. Ce nombre constituera le rapport du cyclique d'un signal PWM qui va être généré pour commander un moteur à courant continu.

#### Exercice 5.2 :

Pour rendre les choses plus intéressantes, la durée d'impulsions à la sortie CCP1 (PORTC, 2) peut être changée avec les touches symboliquement marqués comme "BRILLANT" et "NOIR", tandis que la durée est considérée comme la combinaison binaire sur le PORTD.

La diode D1 signale que le module CCP1 est activé. Dans le montage réel son état corresponde à l'état de la diode D2 pendant la simulation.

- Un appui sur le bouton BRILLANT fait augmenter le rapport cyclique,
- Un appui sur le bouton NOIR fait diminuer le rapport cyclique.

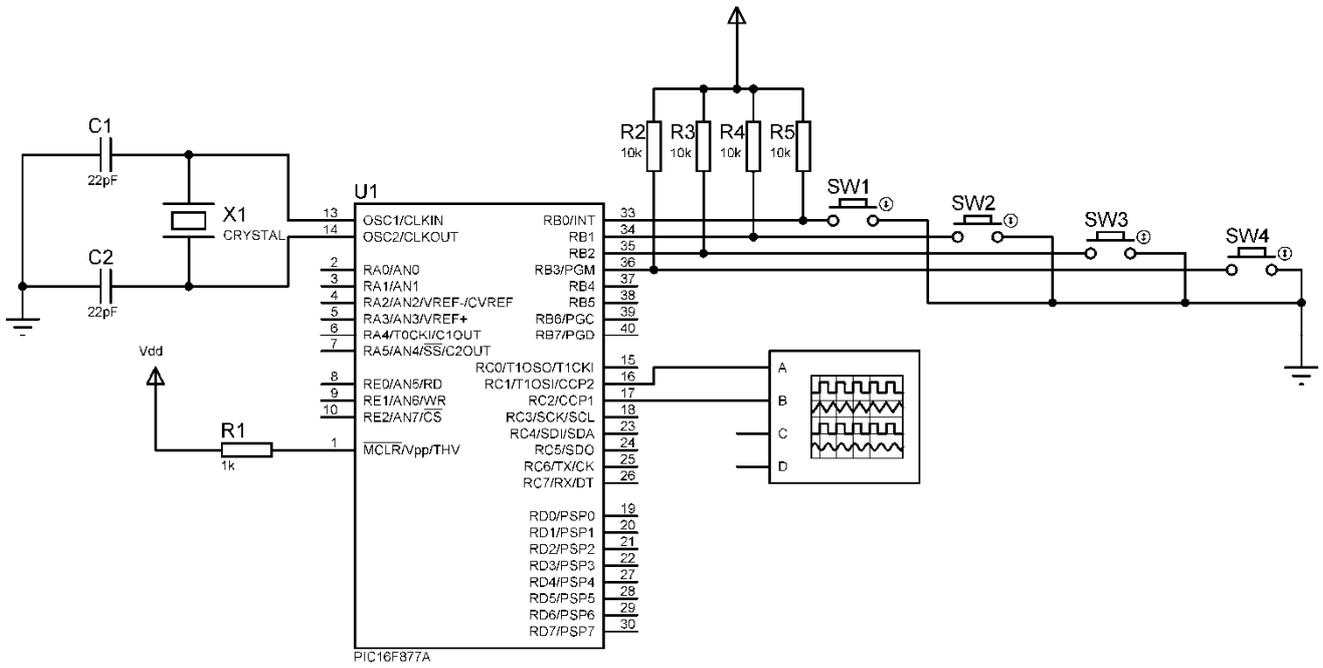


Ecrire un programme en MikroC qui répond au cahier des charges.

**Exercice 5.3 :**

Soit le schéma de la figure ci-après. Quatre commutateurs (SW1, SW2, SW3 et SW4) sont reliés aux quatre bits de poids faible du PORTB du microcontrôleur avec les résistants à tirage vers le haut de 10K. Ces commutateurs sont utilisés pour commander le rapport cyclique du PWM qui est généré par les modules CCP.

- Un appui sur SW1 augmente le rapport cyclique du PWM produit par CCP1,
- Un appui sur SW2 diminue le rapport cyclique du PWM produit par CCP1,
- Un appui sur SW3 augmente le rapport cyclique du PWM produit par CCP2,
- Un appui sur SW4 diminue le rapport cyclique du PWM produit par CCP2.



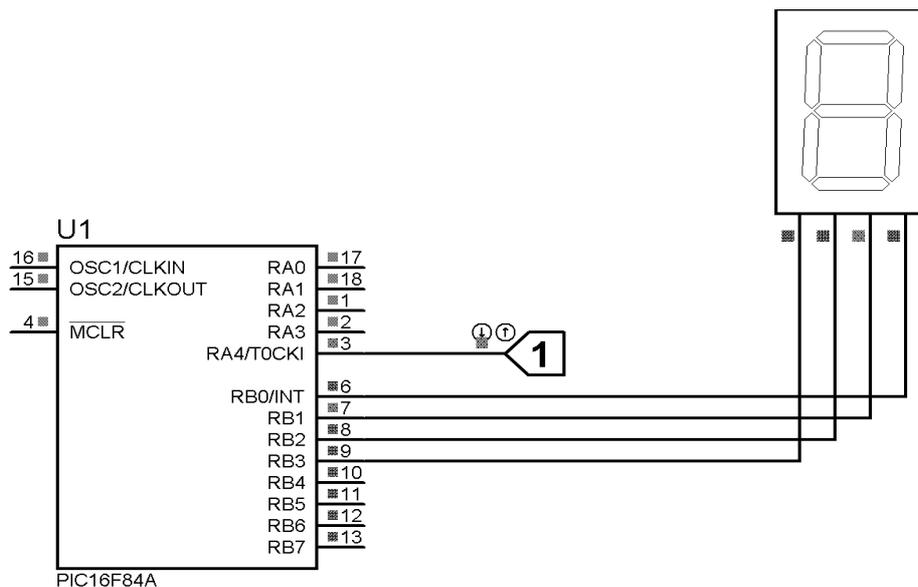
Ecrire un programme en MikroC qui répond au cahier des charges ci-dessus.

## TRAVAUX DIRIGES N° : 6

### LES TIMERS EN MIKROC

#### Exercice 6.1 :

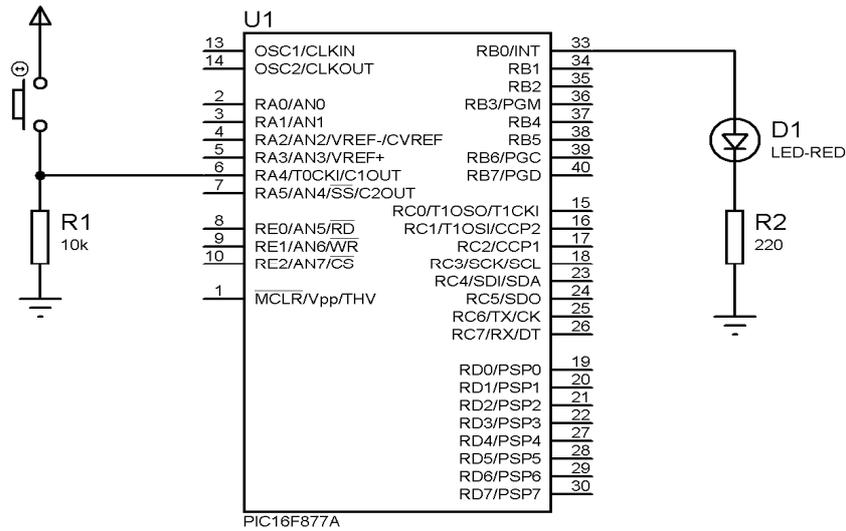
On désire réaliser un compteur modulo 10 en utilisant le Timer TMR0 d'un PIC 16F84A. Le compteur est incrémenté à chaque front montant. On propose le schéma de la figure ci-dessous :



1. Indiquer si le mode de fonctionnement du TMR0 est compteur ou temporisateur.
2. Ecrire un programme en MikroC permettant de compléter la fonction désirée.
5. Modifier le programme précédent pour réaliser un compteur modulo 16 et l'incrémentatation se fait chaque 2 fronts descendants.

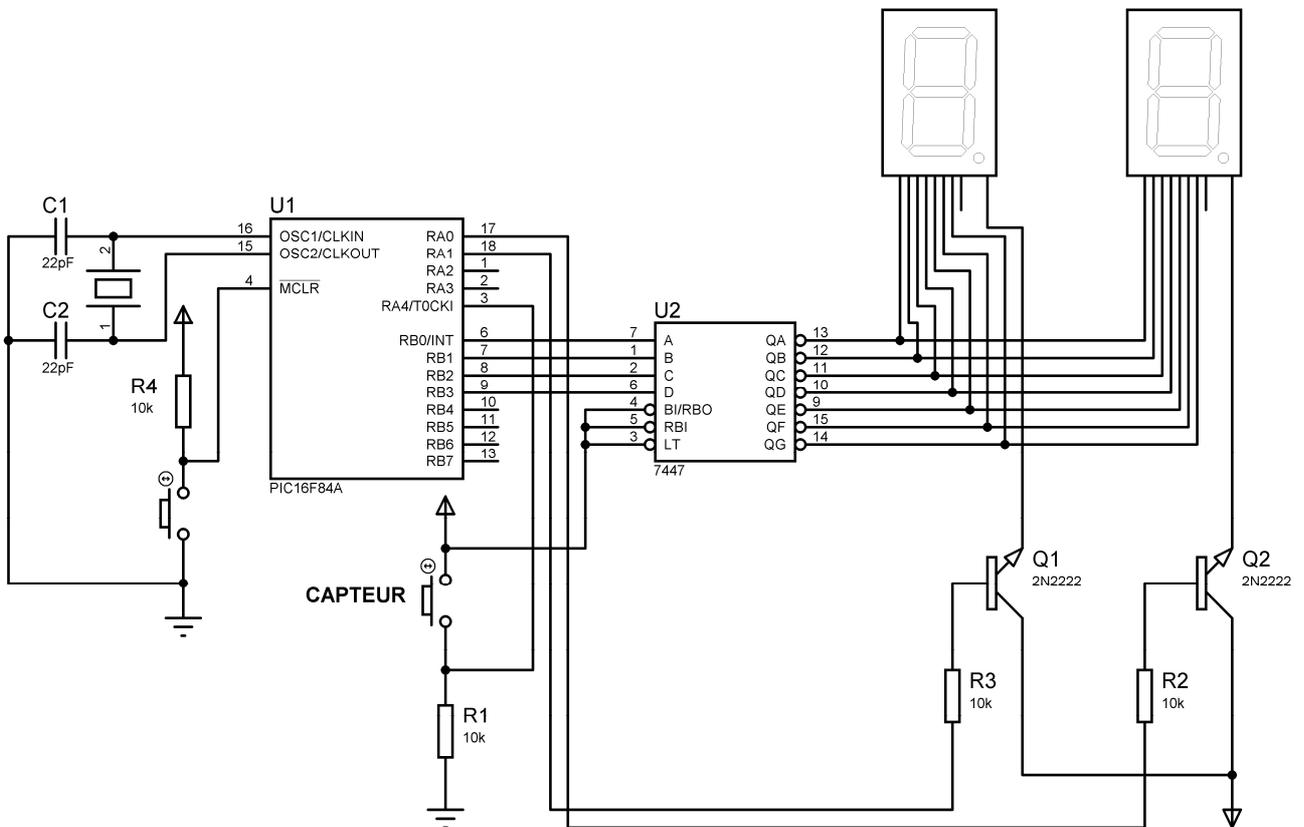
#### Exercice 6.2 :

Selon le circuit de la figure ci-dessous, écrire un programme en C qui permet de compter par le module TIMER0 les fronts montants arrivant sur la broche RA4 et d'allumer la diode chaque fois que 10 fronts sont comptés.



**Exercice 6.3 :**

On désire réaliser un compteur modulo 100 en utilisant le Timer TMR0 d'un PIC 16F84A. Le compteur est incrémenté à chaque front descendant de RA4. On adopte l'affichage multiplexé puisqu'on dispose d'un seul décodeur BCD – 7Segments et deux afficheurs. Ecrire un programme en MikroC.

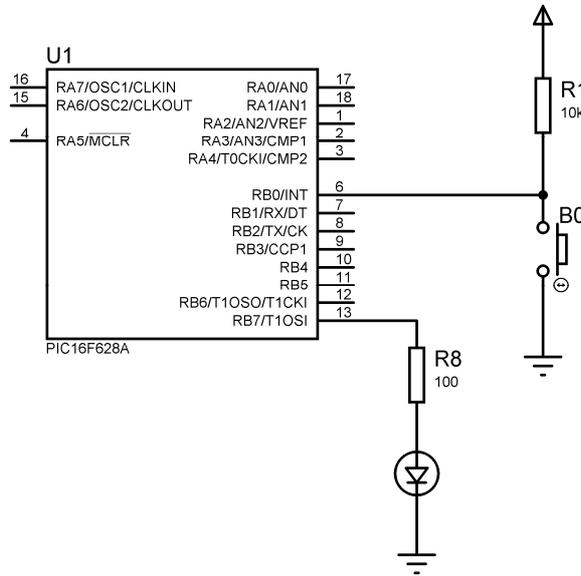


## TRAVAUX DIRIGES N° : 7

### LES INTERRUPTIONS EN MIKROC

#### Exercice 7.1 :

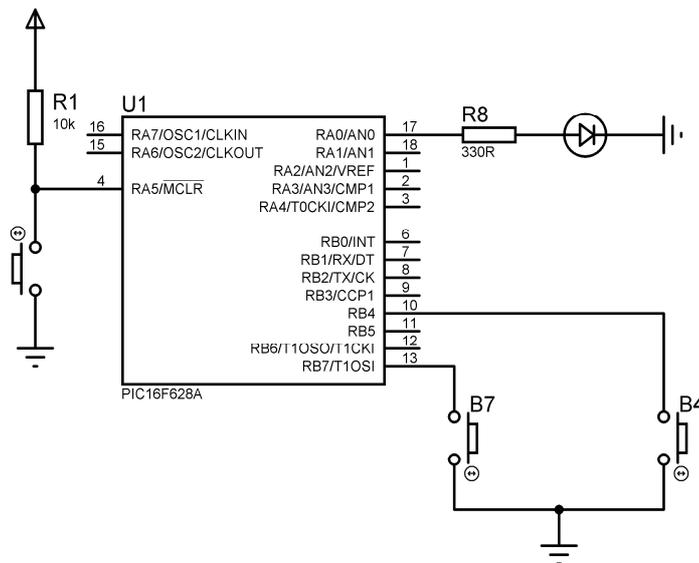
On donne le montage de la figure suivante :



Ecrire un programme qui accepte une interruption sur RB0 et à chaque impulsion sur le bouton poussoir B0 inverse l'état de la diode LED.

#### Exercice 7.2 :

Soit le montage de la figure suivante :

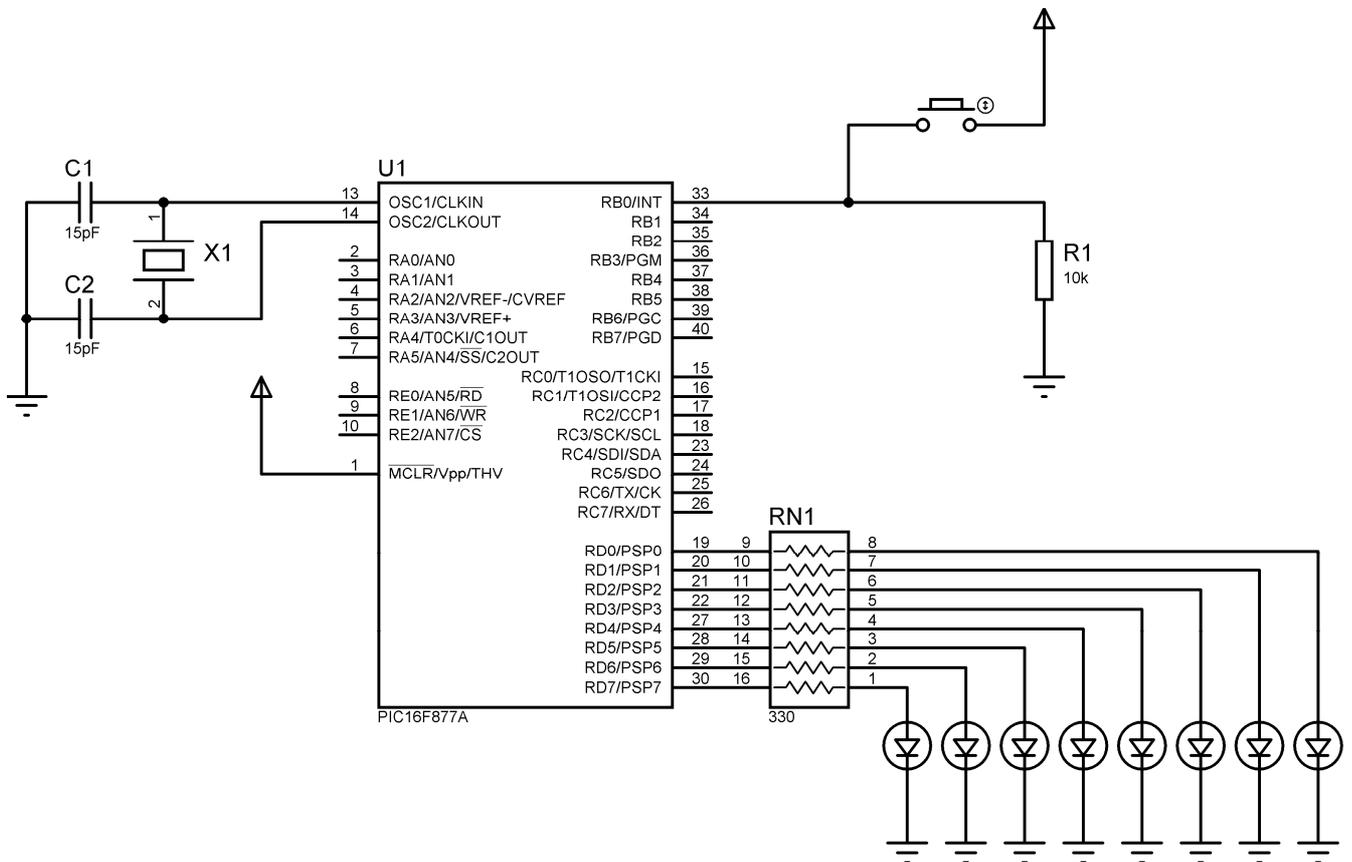


Ecrire un programme qui accepte des interruptions sur RB4 et RB7 et :

- A chaque impulsion sur le bouton poussoir B4 allume la diode LED,
- A chaque impulsion sur le bouton poussoir B7 éteint la diode LED.

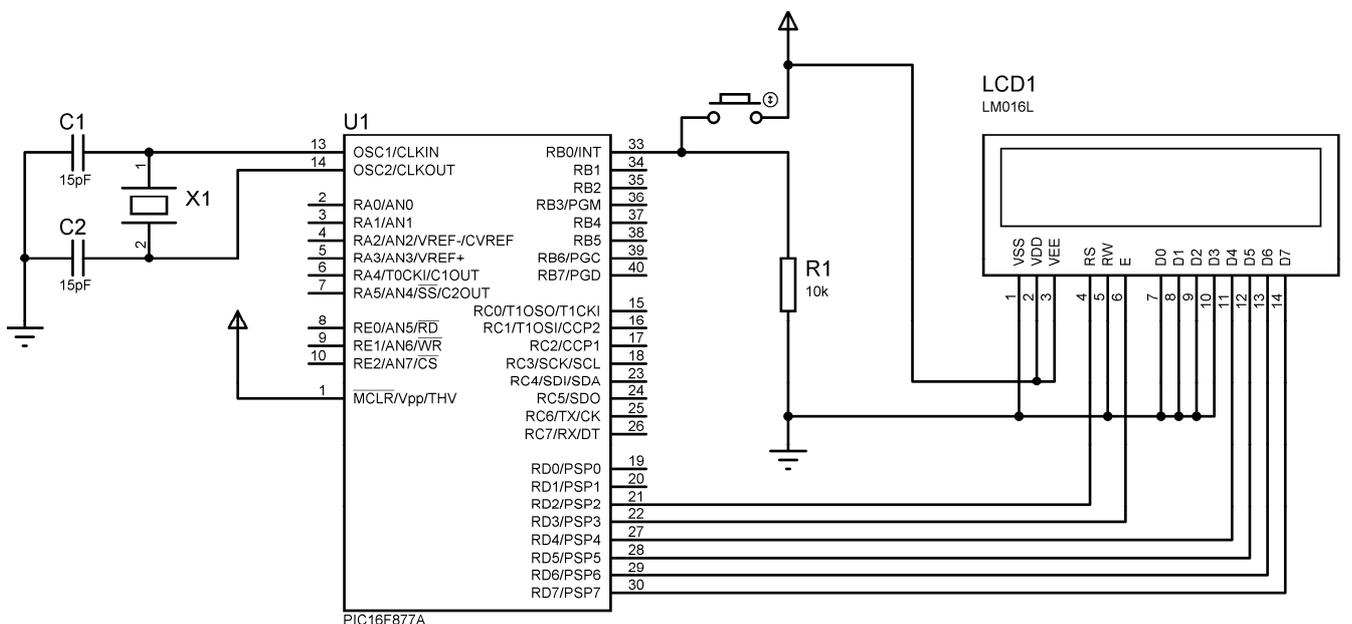
**Exercice 7.3 :**

1. On désire réaliser un compteur binaire 8 bits et afficher le résultat sur 8 diodes LED. Pour réaliser ce cahier des charges, on propose le circuit de la figure suivante.



Ecrire un programme en MikroC qui à chaque impulsion sur le bouton lié à l'entrée RB0 incrémente le compteur et affiche le résultat sur le PORTD.

2. Modifier le programme précédent afin d'afficher le résultat du comptage sur un afficheur LCD selon le montage suivant :



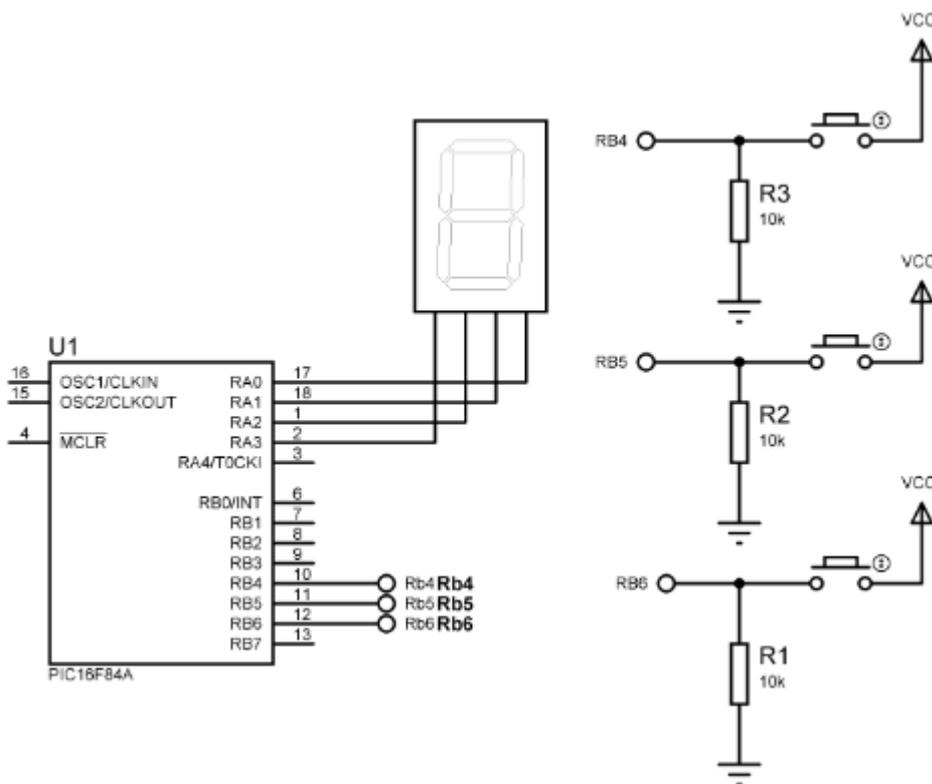
**Exercice 7.4 :**

Le but est de réaliser un décompteur dont le point de départ est géré par des interruptions générées par le changement d'état sur RB4, RB5 et RB6.

- Si on appuie sur RB6 alors le décompteur réalise le cycle suivant : F, E, D, ..., 0
- Si on appuie sur RB5 alors le décompteur réalise le cycle suivant : A, 9, 8, ..., 0
- Si on appuie sur RB4 alors le décompteur réalise le cycle suivant : 7, 6, 5, ..., 0
- Sinon le décompteur réalise le cycle par défaut : 5, 4, 3, ..., 0.

Pour réaliser ce cahier des charges, on vous propose le montage de la figure 2. Pour simplifier l'affichage des résultats du décomptage, on utilise un afficheur 7 segments-BCD.

En effet, cet afficheur affiche les valeurs de 0 à F (en hexadécimal) directement sans passer par la table de décodage des 7 segments.



Ecrire un programme C qui accepte des interruptions sur les entrées RB4, RB5 et RB6 et affiche sur l'afficheur 7 segments-BCD le cycle de décomptage correspondant.