

# Les liaisons séries



# Les liaisons séries des PIC16Fxxx

- **USART ou AUSART:**
  - Liaison synchrone ou asynchrone avec possibilité d'adressage
- **MSSP (Master Synchronous Serial Port)**
  - **SPI :**
    - Liaison synchrone sur 4 fils
  - **I<sup>2</sup>C ou MI<sup>2</sup>C :**
    - Liaison synchrone I<sup>2</sup>C esclave ou maître - esclave sur 2 fils



# L'USART

- **Addressable Universal Synchronous Asynchronous Receiver Transmitter**
- **Cette liaison peut être configurée**
  - **en mode asynchrone - full duplex**
  - **en mode synchrone maître - half duplex**
  - **en mode synchrone esclave - half duplex**
- **Elle utilise deux broches du port C RC6 et RC7**



# Les registres associés

- **TXSTA**      contrôle et état de l'émission
- **RCSTA**      contrôle et état de la réception
- **SPBRG**      générateur de la vitesse de transmission
- **TXREG**      donnée à transmettre
- **RCREG**      donnée reçue
- les registres d'interruption **PIR1** et **PIE1**



# Le registre TXSTA



1 Maître  
0 Esclave

Transmission autorisée

Haute (1) ou basse vitesse

9ème bit de données

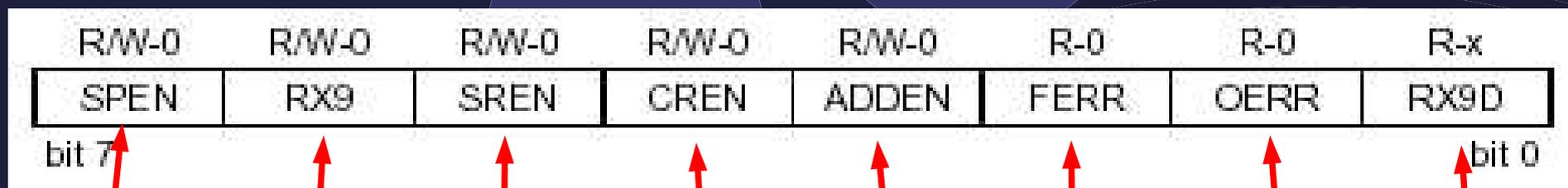
Transmission sur 8 (0) ou 9 (1) bits

Mode sync (1) ou async (0)

Etat du registre de transmission  
1 vide



# Le registre RCSTA



Validation  
de la comm.  
série

mode sync  
reception  
unique

Détection  
d'adresse  
validée

Erreur de  
bourrage

Validation de la  
réception du bit 9

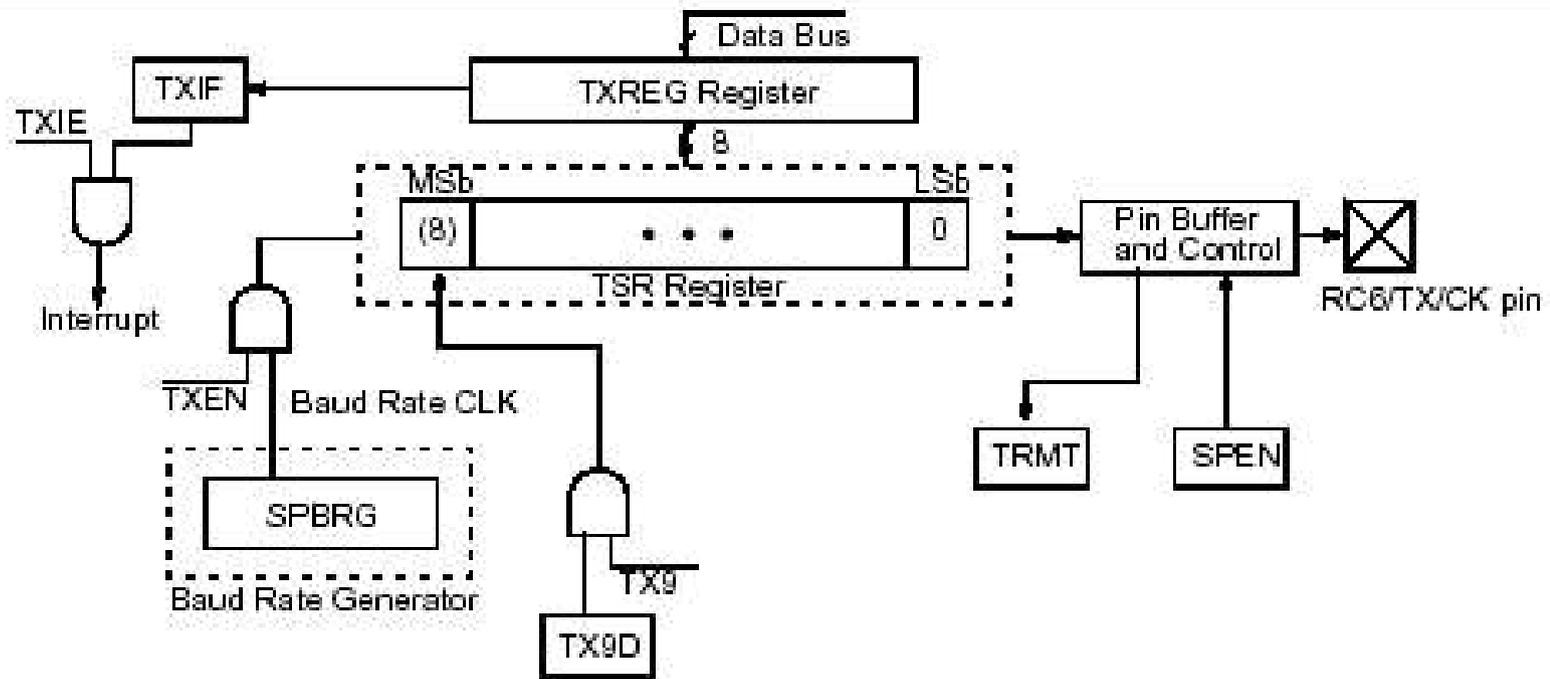
Validation de  
la réception

Erreur  
de  
trame

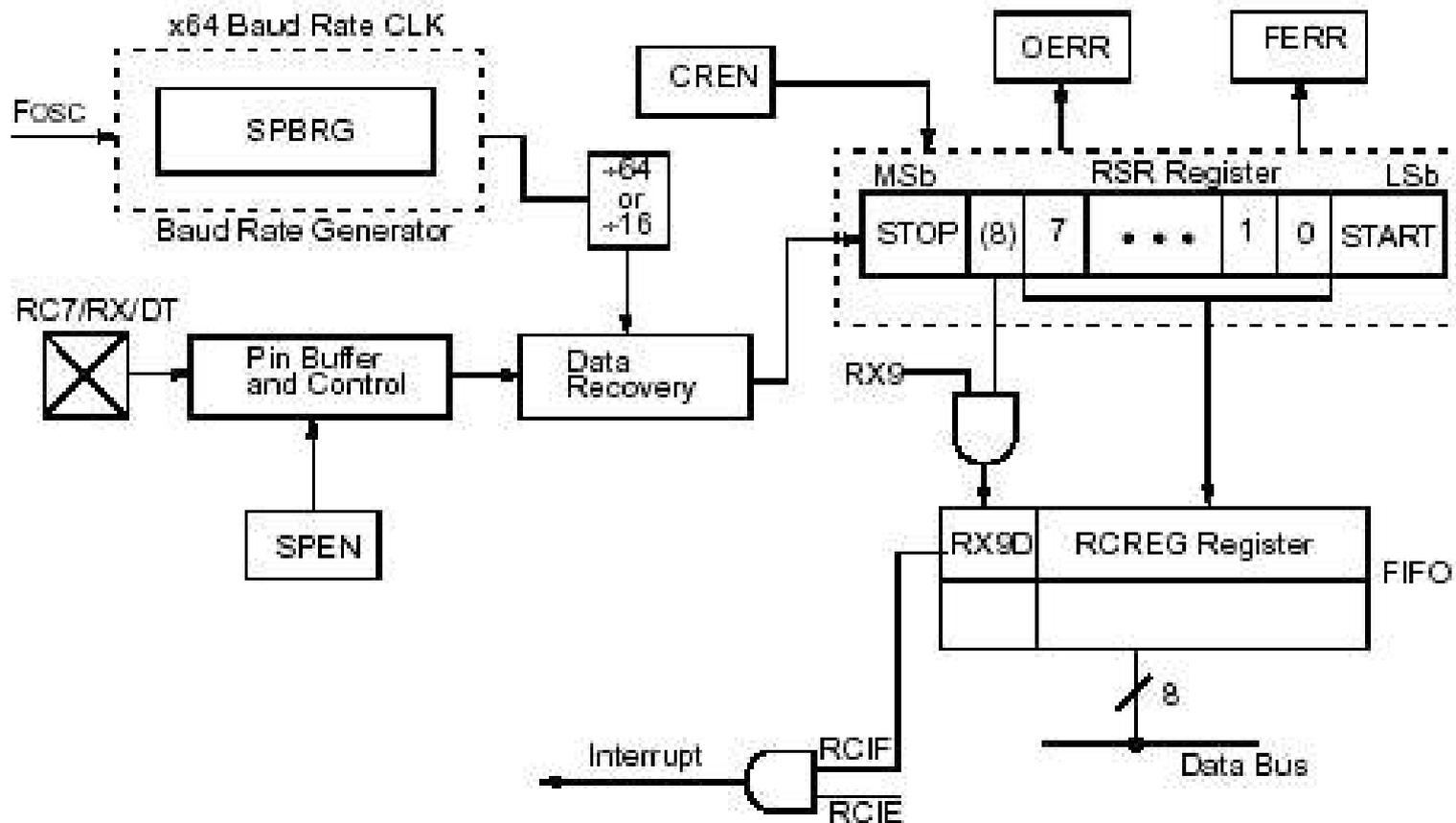
9ème bit  
reçu



# Schéma fonctionnel de la transmission



# Schéma fonctionnel de la réception



# Le module MSSP

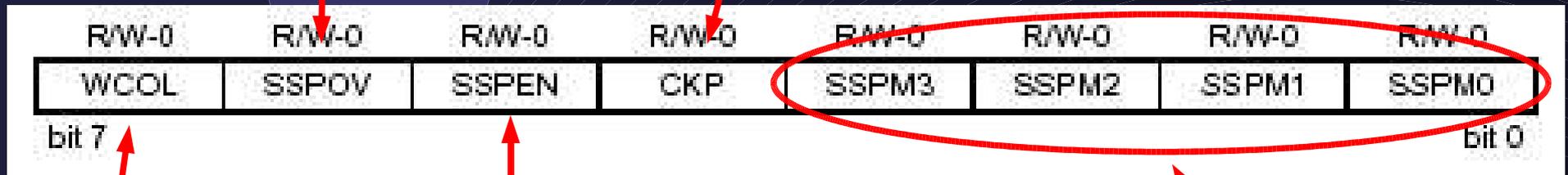
- Module port série synchrone maître
- Peut fonctionner suivant l'un des deux modes:
  - Serial Peripheral Interface SPI
  - Inter Integrated Circuit I<sup>2</sup>C
- Utilise les broches RC3, RC4 et RC5
- Le fonctionnement du module est contrôlé par les registres SSPCON et SSPCON2
- L'état de son fonctionnement est décrit par le registre SSPSTAT
- Les données sont écrites ou lues dans le registre SSPBUF



# Le registre SSPCON

Déplacement en réception

Polarité de l'horloge



Détection de collision

Validation du mode SSP

Choix du mode de fonctionnement

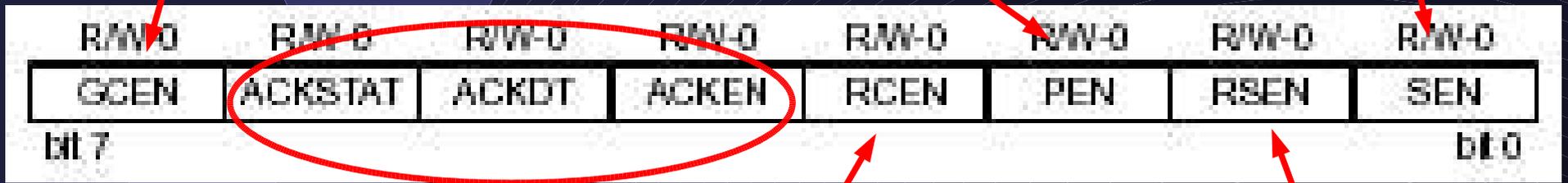


# Le registre SSPCON2 contrôle le mode I<sup>2</sup>C

Validation de l'appel général I<sup>2</sup>C esclave

Validation de la condition de Stop

Validation de la condition de Start



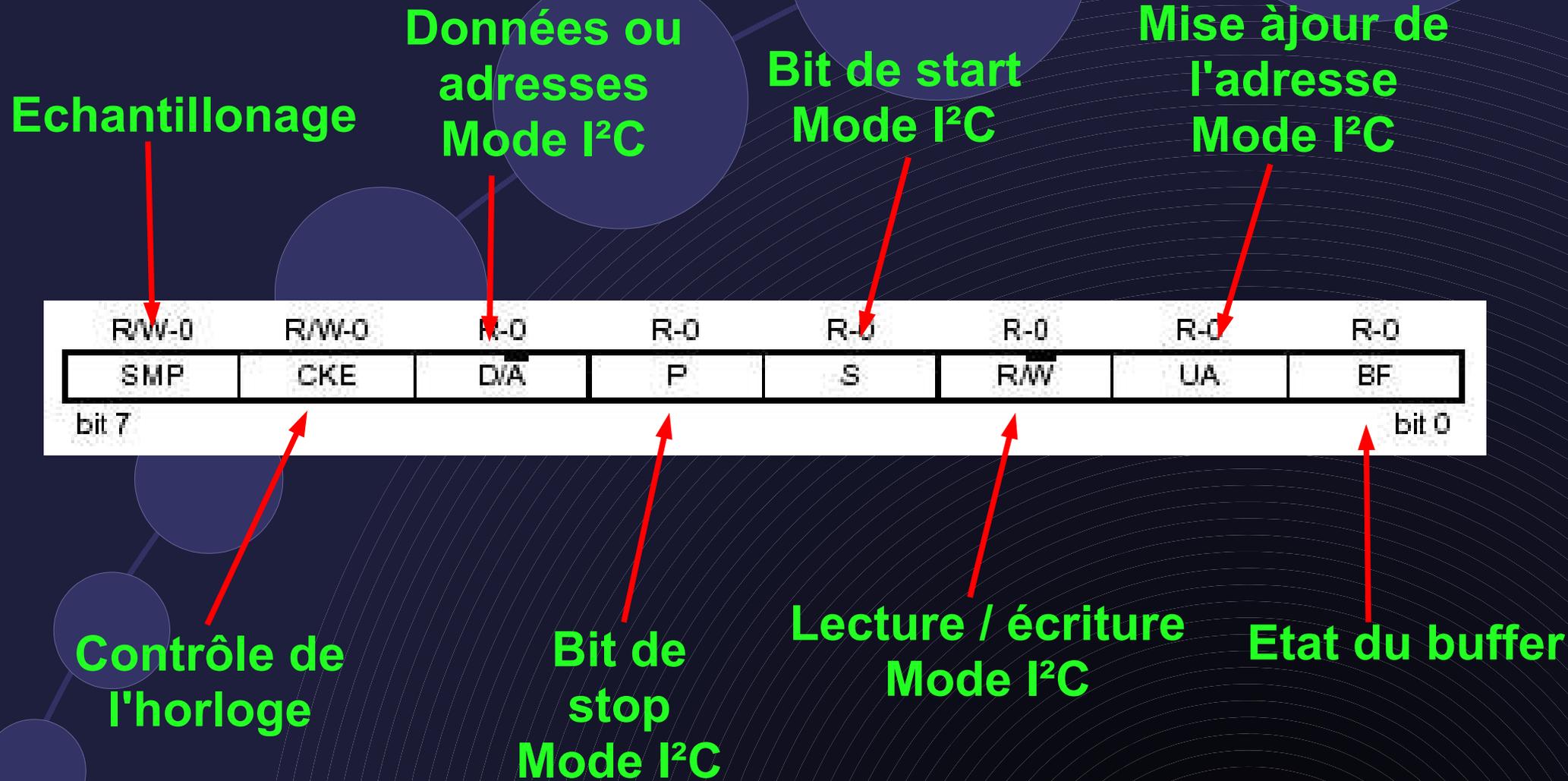
Gestion du bit d'acquiescement en I<sup>2</sup>C maitre

Validation du mode réception

Validation de la répétition de la condition de Start

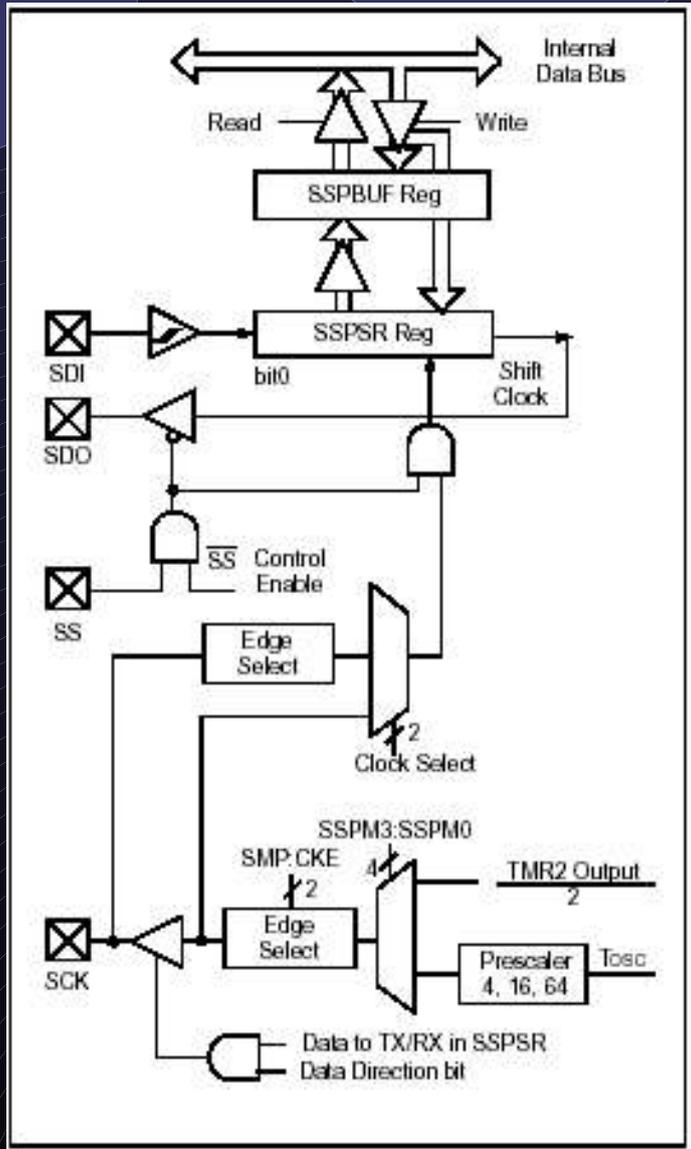


# Le registre SSPSTAT



# La liaison SPI

- Utilise 4 broches:
  - SDO (RC5) sortie des données
  - SDI (RC4) entrée des données
  - SCK (RC3) horloge
  - SS (RA5) sélection (facultative)
- Permet la sélection
  - du mode maitre ou esclave
  - de la polarité de l'horloge
  - du front actif de l'horloge
  - de la vitesse



# Pour utiliser la SPI il faut:

- Configurer TRISC pour avoir SCK et SDO en sortie et SDI en entrée
- Choisir le mode maitre ou esclave et la vitesse de l'horloge bits  $SSPM<3:0>$
- Choisir la polarité et le front de l'horloge (CKP et CKE)
- Valider le port série  $SSPEN = 1$

Une interruption peut être générée à la fin de la transmission



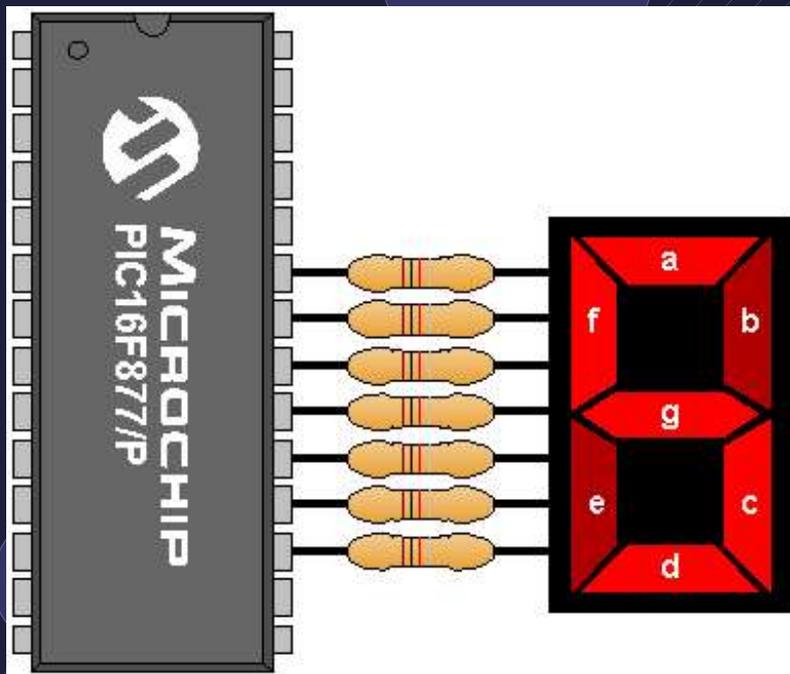
# La liaison I<sup>2</sup>C

- Peut être Esclave ou Maître/esclave
- Peut avoir 7 ou 10 bits d'adresse en esclave
- Permet le fonctionnement avec plusieurs maitres
- Son fonctionnement est contrôlé par les mêmes registres que la SPI SSPCON, SSPCON2 et SSPSTAT
- Le deux lignes RC3/SCL et RC4/SDA doivent être configurées en entrées dans Trisc



# Utilisation d'une table

- Utilisation d'une table pour la commande d'un afficheur 7 segments



```

org      0x0020          ; Page 0
movf    DisplayValue,w
call    SevenSegmentDecode
movwf   PORTB

org      0x100          ; Page 3
SevenSegmentDecode
addwf   PCL,f
retlw   b'00111111'    ; Decode 0
retlw   b'00000110'    ; Decode 1
retlw   b'01011011'    ; Decode 2
retlw   b'01001111'    ; Decode 3
retlw   b'01100110'    ; Decode 4
retlw   b'01101101'    ; Decode 5
retlw   b'01111101'    ; Decode 6
retlw   b'00000111'    ; Decode 7
retlw   b'01111111'    ; Decode 8
retlw   b'01101111'    ; Decode 9

```

