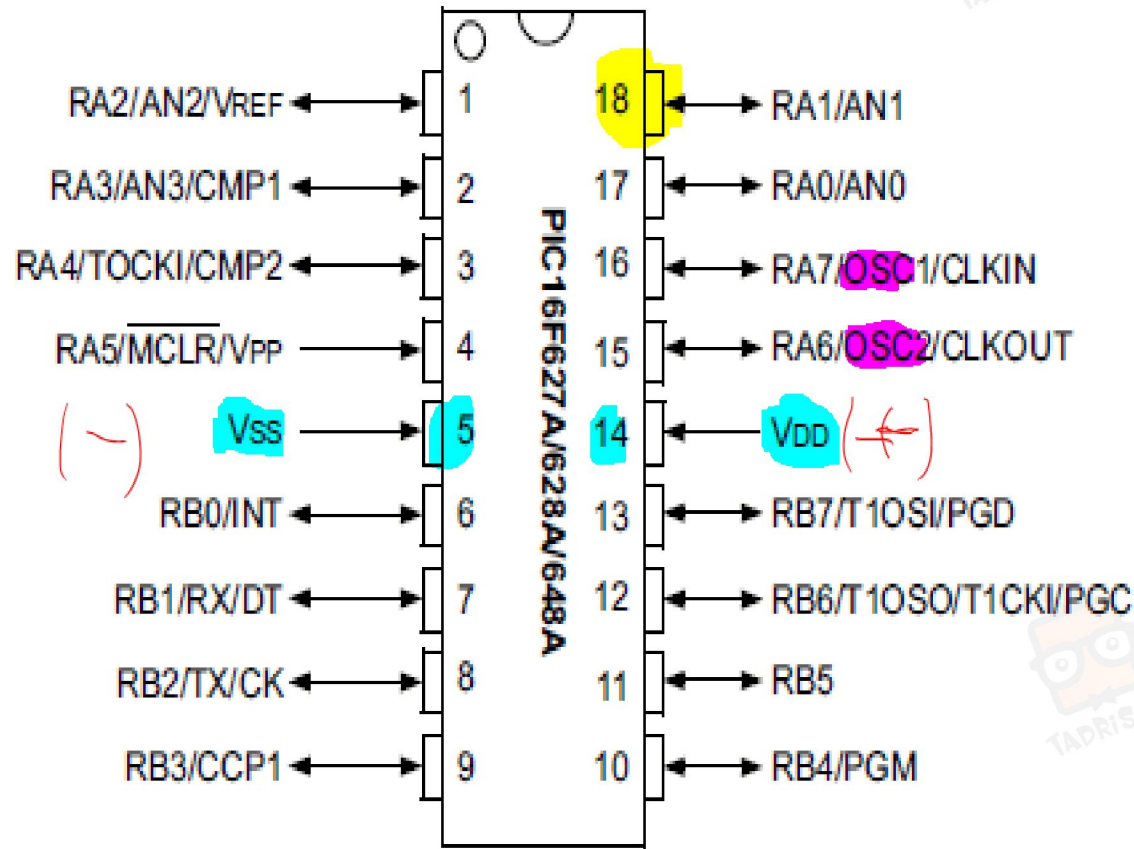


## Exercice 1 :

On donne la représentation du microcontrôleur



1- Identifier les numéro des pins du porta

RA0: 17, RA1: 18, RA2: 1, RA3: 2, RA4: 3, RA5: 4, RA6: 15, RA7: 16.

2- Dans ce microcontrôleur existe-il d'autre port ? Si oui donner le nom.

Oui, le port B.

3- Donner le rôle des pins (5, 14).

les broches d'alimentation.

4- L'entrée 5 (MCLR) sert à quoi ?

c'est l'entrée de reset : initialiser le programme.

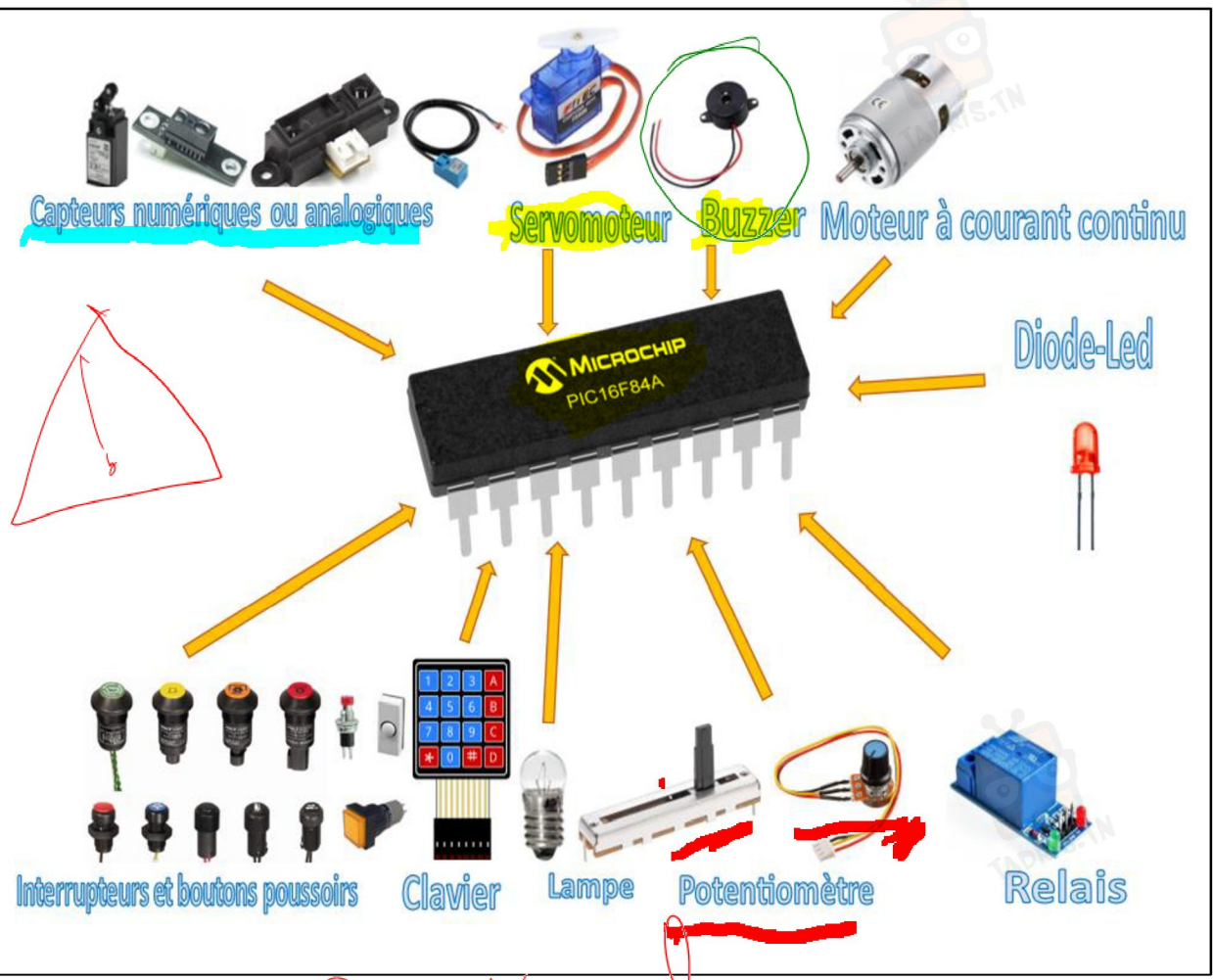
5- Ou doit on brancher l'oscillateur pour ce circuit ?

sur les broches 15 et 16.



$E \rightarrow 1$   
 $S \rightarrow 0$  } TRIS

## Exercice 2 :



La figure ci-contre représente un microcontrôleur et les périphériques qu'on peut y associer avec.

1- Classer ses périphériques selon leur type entrée ou sortie. E(entrée) ou S (sortie) dans la case de chaque élément.

Capteur	E	Servomoteur	S	Buzzer	S	Moteur DC	S	Led	S
Bouton et interrupteur	E	Clavier	E	Lampe	S	Potentiomètre	E	Relais	S

2- Sur le PORTD on branche les éléments suivants :

- RD0 : bouton poussoir**
- RD1 : Diode LED**
- RD2 : capteur**
- RD3 : relais**
- RD4 : moteur**
- RD5 : Transistor**
- RD6 : interrupteur**
- RD7 : Lampe**

Retrouver la configuration du PORTD

	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
<b>TYPE(E/S)</b>	S	E	S	S	S	E	S	E
<b>TRISD</b>	0	1	0	0	0	1	0	1

TRISD = 0b...0...1...00101.

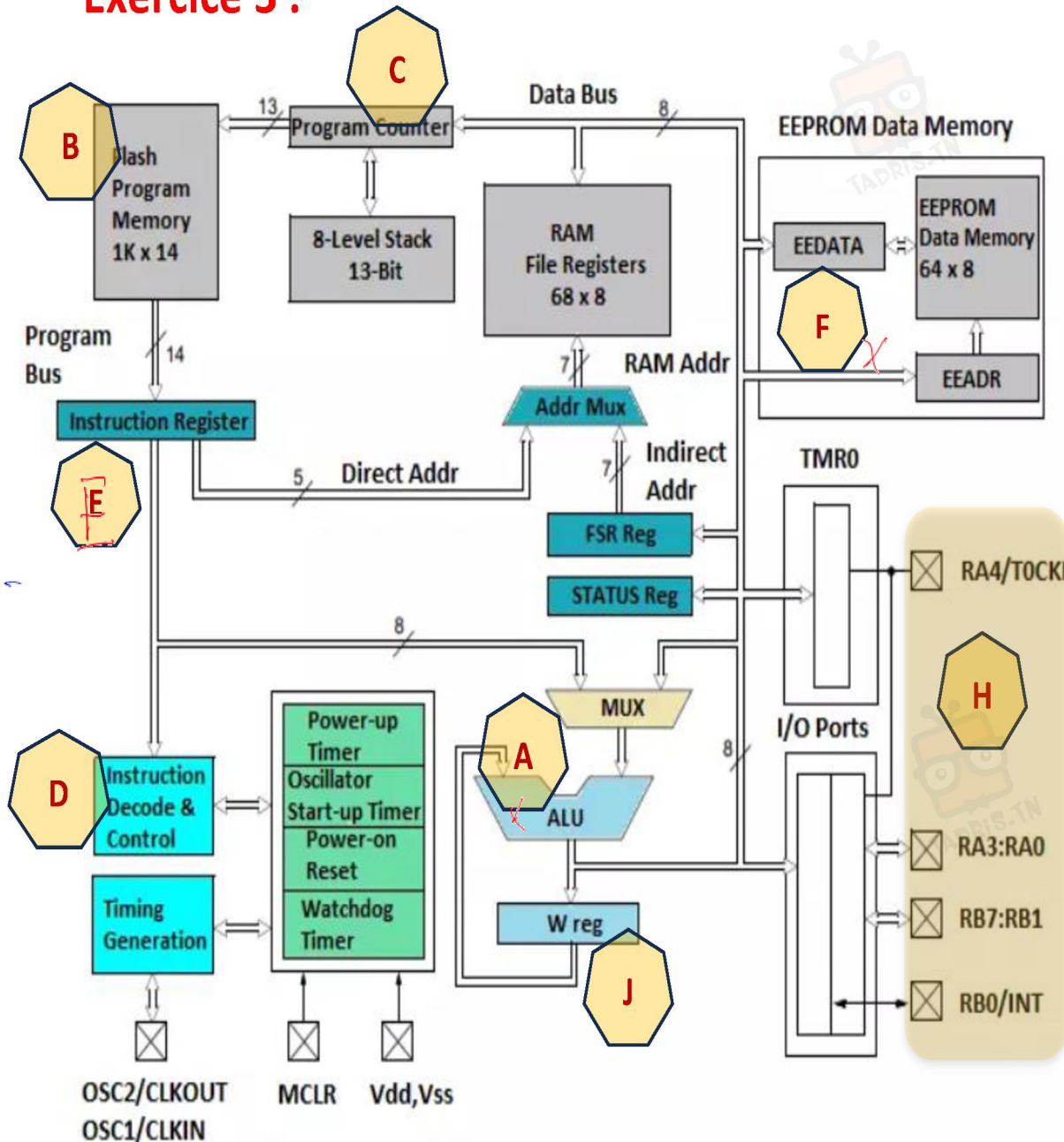
3- Donner la signification des instructions suivantes

TRISB = 0x00  
 ..... tout le port B est configuré en sortie .....  
 TRISA = 0b010011  
 RA0, RA1, RA4 : configuré comme des entrées, le reste des sorties

0x → base hexadécimale  
 0b → base binaire



### Exercice 3 :



1- Lorsque plusieurs entrées/sorties logiques sont associées matériellement, on dit qu'elles forment un port. Barrer la mauvaise réponse.

~~RA0 RA1~~ -- ~~RB1 RB3~~ --

a. Vrai

~~b. Faux~~

2- Les broches RA0/AN0 et RA1/AN1 du PIC16F87 peuvent être raccordées à un µC : barrer la mauvaise réponse.

a. Un détecteur de mouvement ✓

~~b. Une lampe de 220v~~

~~d. Un étage de puissance pour la commande en d'un moteur monophasé ✓~~

Relais

3- Donner le nom des mémoires qui se trouvent dans un microcontrôleur.

Mémoire programme, Mémoire de données, RAM.

4- Donner le nom de l'élément qui permet d'effectuer les calculs dans un microcontrôleur.

UAL : unité Arithmétique et logique

5- C'est quoi le rôle du registre W.

enregistre le dernier résultat de l'UAL

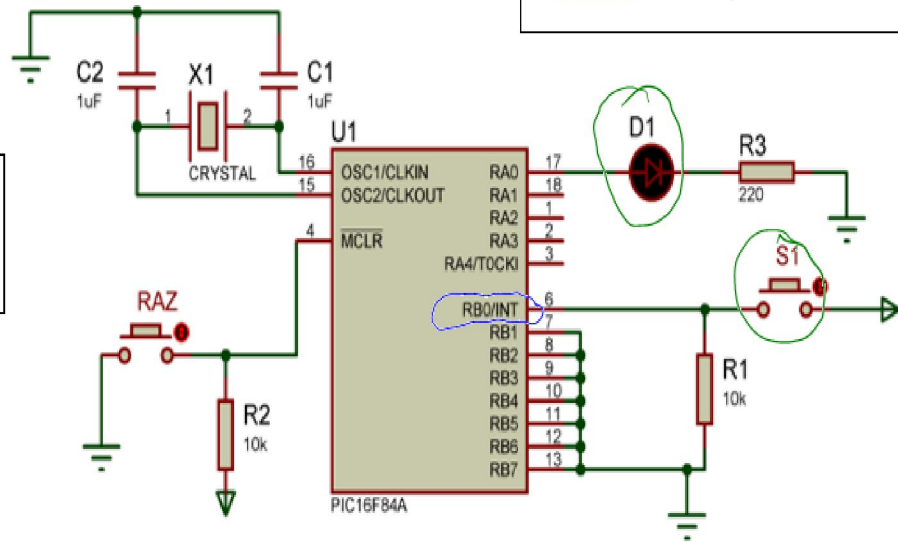
6- Donner l'ordre d'exécution d'une instruction, d'après les repères sur la figure.

C-B-E-D-F-A-J-H

## Exercice 4 :

On désire commander une diode LED (D1) à l'aide d'un bouton (S1) en utilisant un microcontrôleur 16F84A comme donnée par la figure suivante :

Une adresse non utilisée sera considérée comme une sortie



S1=1 → D1=1

S1=0 → D1=0

1-Configurer le registre TRISA en se référant à la figure. (Complétez avec « 0 » ou « 1 ».)

RA4	RA3	RA2	RA1	RA0
0	0	0	0	0

TRISA = 0b00000

2-Configurer le registre TRISB en se référant à la figure. (Complétez avec « 0 » ou « 1 ».)

RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
0	0	0	0	0	0	0	1

TRISB = 0b00000001  
= 0x01

3-Compléter les programmes répondant aux critères de fonctionnement.

Solution N°1 avec des variables

```

sbit S1 at RB0_bit;
sbit D1 at portA.B0;
void main()
{
    TRISA=0b00000;
    TRISB=0x01;
    PortA=0;
    while (1) // boucle infinie
    {
        if (S1==1)
            D1=1;
        else
            D1=0;
    }
}
    
```

Solution N°2 sans variables

```

void main()
{
    TRISA=0b00000;
    TRISB=0x01;
    PortA=0;
    while (1)
    {
        if (portB.B0==1)
            portA.B0=1;
        else
            portA.B0=0;
    }
}
    
```



في دارك... إتهنوني على قرابتك إصغارك

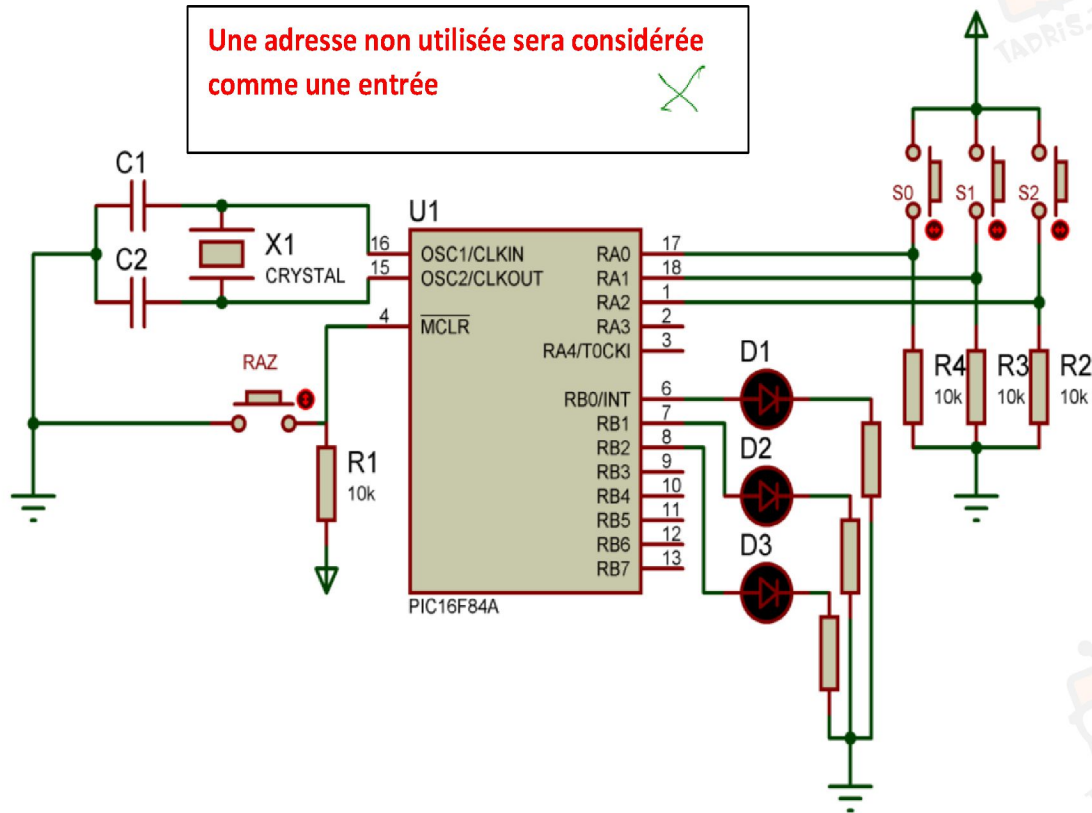




## Exercice 5 :

Un circuit à base de 16F84A, comporte 3 boutons poussoirs (S0, S1 et S2) et trois LED (D1, D2 et D3).

Une adresse non utilisée sera considérée comme une entrée



2- Donner la configurations des registres TRISA et TRISB.

$TRISA = 0b11111111; = 0x1F$

$TRISB = 0b11111000; = 0xF8$

1- Le fonctionnement du circuit précédant est décrit avec le tableau suivant :

S2	S1	S0	D1	D2	D3
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	0	1
1	1	1	1	0	1

- Compléter les programmes répondant aux critères de fonctionnement.

sbit S0 at RA0\_bit; sbit S1 at RA1\_bit;

sbit S2 at RA2\_bit; sbit D1 at RB0\_bit;

sbit D2 at RB1\_bit; sbit D3 at RB2\_bit;

void main(){

trisa=0x1F.....; trisb=0xF8.....; portb=0.....;

for(;;)// .....

{

if((S2==0)&&(S1==1)&&(S0==1)) {D1=1; D2=0; D3=0;}

else if((S2==1)&&(S1==0)&&(S0==1)) {D1=0; D2=1; D3=0;}

else if((S2==1)&&(S1==1)&&(S0==0)) {D1=0; D2=0; D3=1;}

else if((S2==1)&&(S1==1)&&(S0==1)) {D1=1; D2=0; D3=1;}

else {D1=0; D2=0; D3=0;}

}}

déclaration.

configuration

initialisation

boucle infinie

programme