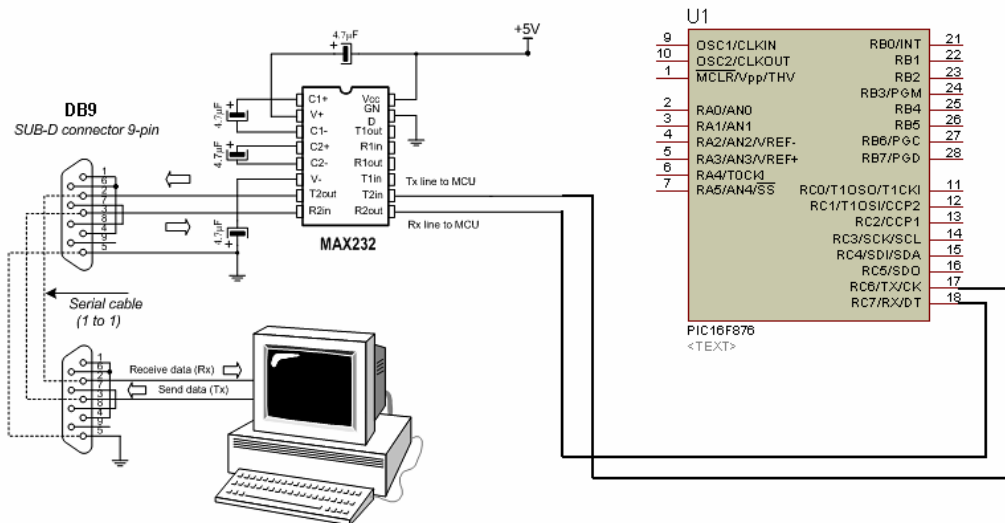


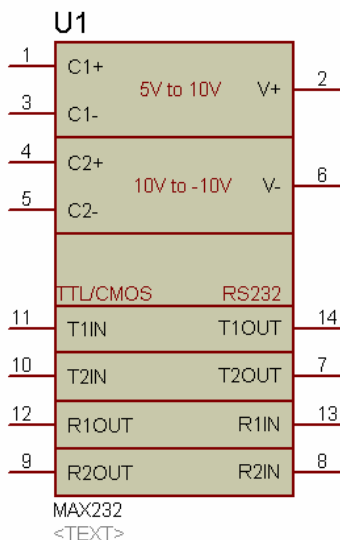
TD1 CAO

Projet :

On désire réaliser une carte électronique à base d'un microcontrôleur de la famille PIC 16F876 et qui assure une liaison série du type RS232 avec un PC.



On utilisera le logiciel ISIS pour réaliser le schéma de la carte, mais la bibliothèque des composants ne contient pas le circuit intégré MAX232, on vous demande de **créer ce composant**, de **compléter le schéma** et de **réaliser le circuit imprimé** de la carte en simple face.



MAX232 :

- Les broches **VCC** et **GND** ont respectivement les numéros **16** et **15**.
- L'empreinte physique est du type DIL16.

NB :

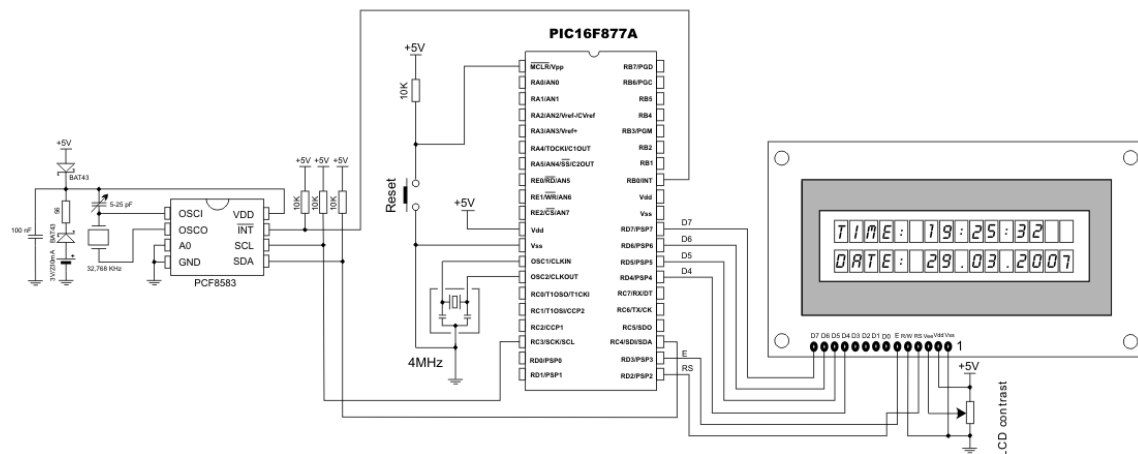
N'oubliez pas de prévoir un **connecteur** pour **alimenter** la carte !

TD2 CAO

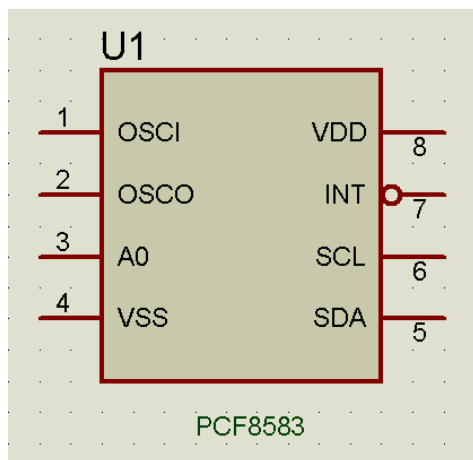
Projet :

On désire réaliser une carte électronique à base d'un microcontrôleur de la famille PIC 16F877A et qui assure une liaison série du type I2C avec un module d'horloge temps réel PCF8583 afin d'afficher le temps et la date sur un afficheur LCD.

Example of connecting real time clock PCF8583



1) On utilisera le logiciel ISIS pour réaliser le schéma de la carte, mais la bibliothèque des composants ne contient pas le circuit intégré PCF8583, on vous demande de **créer ce composant** et de **réaliser le circuit imprimer** de la carte en simple face.



PCF8583 :

- L'empreinte physique est du type DIL08.

NB :

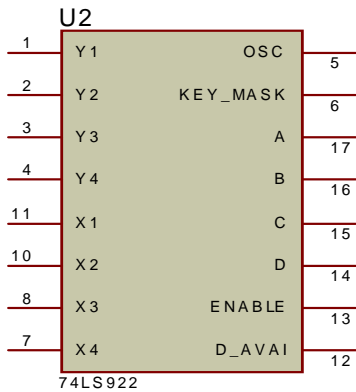
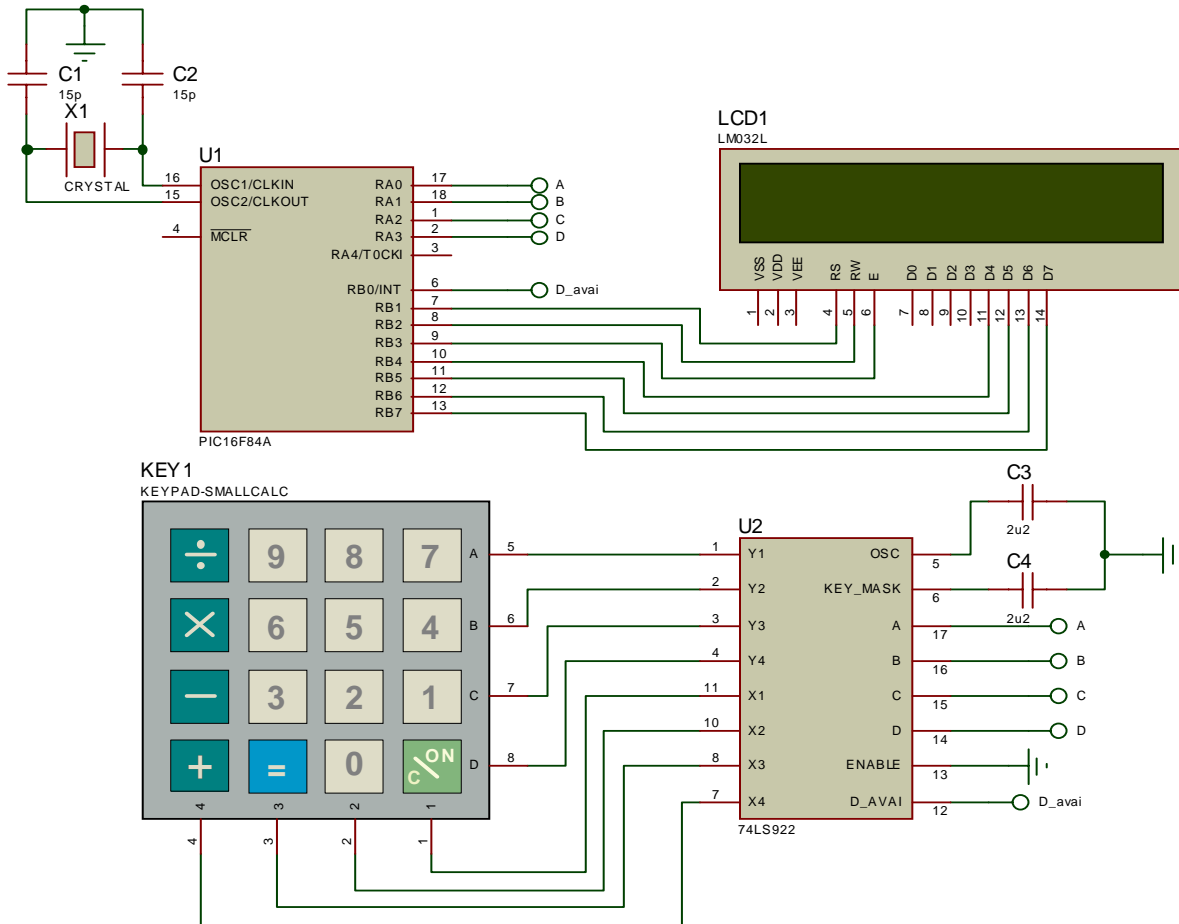
N'oubliez pas de prévoir un **connecteur** pour **alimenter** la carte !

2) Ecrire un programme en µpascal qui permet d'afficher sur le LCD le message suivant :

Nous somme le :
9 /12/ 2007

TD3 CAO

Projet : On désire réaliser une calculatrice à base d'un microcontrôleur de la famille PIC 16F84A.. On utilisera le logiciel ISIS pour réaliser le schéma de la carte, mais la bibliothèque des composants ne contient pas le circuit intégré 74922 (décodeur de clavier), on vous demande de créer ce composant et de réaliser le circuit imprimé de la carte en double face.



- Les broches **VCC** et **GND** ont respectivement les numéros **18** et **9**.
- L'empreinte physique est du type **DIL18**.

NB :
 N'oubliez pas de prévoir un **connecteur** pour **alimenter** la carte !

TD4 CAO

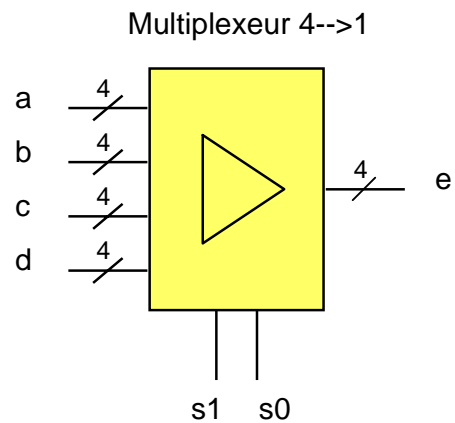
Questions de cours : (6 pts.)

- 1) Citer les différentes phases de réalisation d'un projet en électronique.
- 2) Donner les avantages et les inconvénients de la conception assistée par ordinateur en électronique.

Exercice 1: (6 pts.)

- 1) Compléter le programme en VHDL suivant afin de décrire un multiplexeur 4 vers 1 donné dans la figure 1

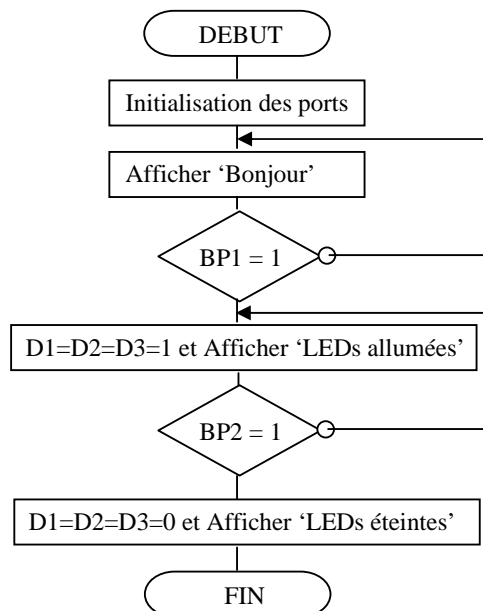
```
library ieee;  
use ieee.std_logic_1164.all;  
entity MUX4 is  
port(.....  
.....  
.....  
end MUX4;  
architecture ARCHMUX4 of MUX4 is  
begin  
e <= a when.....  
.....  
.....  
end ARCHMUX4;
```



Exercice 2: (8 pts.)

On désire simuler le fonctionnement d'une carte à base du microcontrôleur PIC 16F876 avec le logiciel ISIS, le schéma est donné dans la **figure 2**.

On vous demande d'écrire un programme en μ PASCAL qui permet de simuler le fonctionnement de la carte suivant l'organigramme donné ci-dessous.



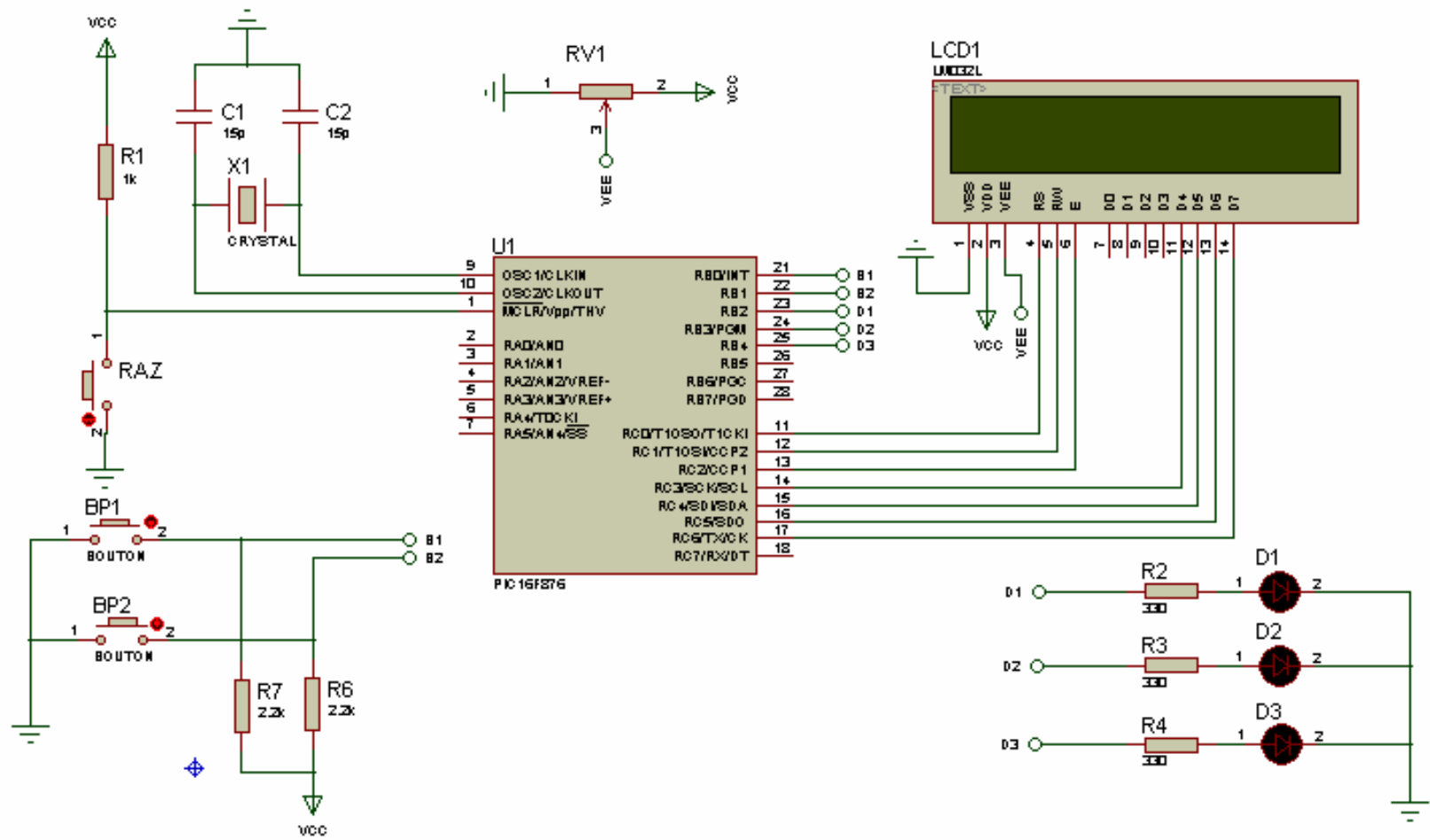


Figure 2

TD5 CAO

Questions de cours : (2 pts.)

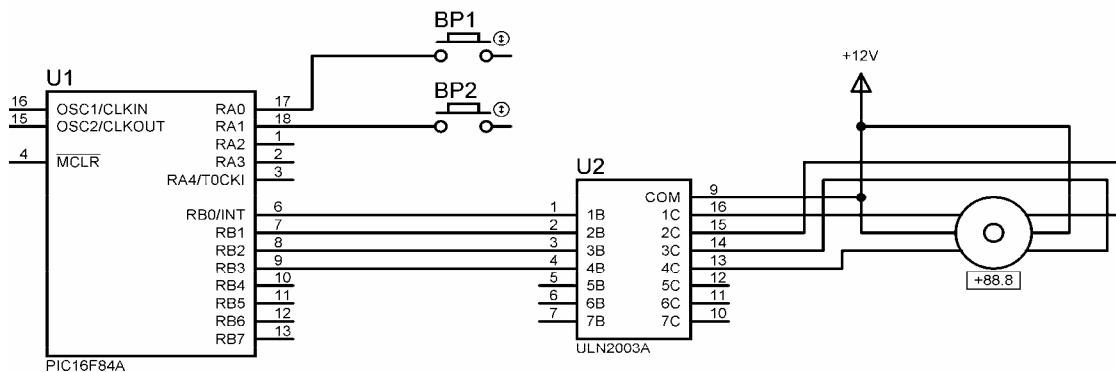
1) Donner la structure d'une description VHDL .

Exercice 1: (5 pts.)

Ecrire une description VHDL d'un additionneur 1 bit.

Exercice 2: (7 pts.)

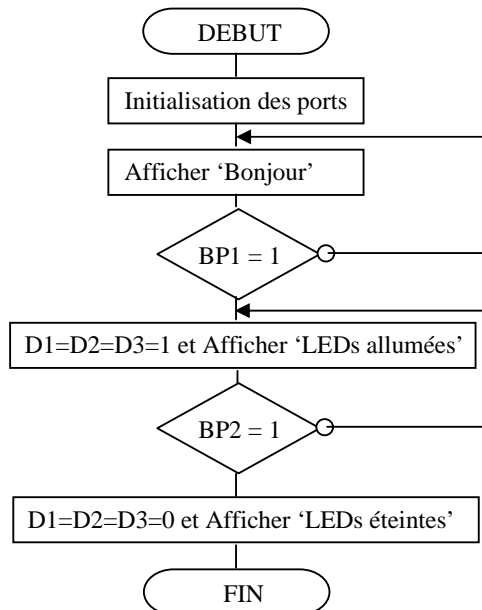
- 1) Compléter le schéma de la figure suivante.
- 2) Ecrire un programme en μ Pascal pour commander le moteur dans les deux sens de rotation



Exercice 3: (6 pts.)

On désire simuler le fonctionnement d'une carte à base du microcontrôleur PIC 16F876 avec le logiciel ISIS. On utilise alors un afficheur LCD, deux boutons poussoirs BP1 et BP2 et trois diode LEDs :D1,D2,D3

On vous demande de proposer une solution et d'écrire un programme en μ PASCAL qui permet de simuler le fonctionnement de la carte suivant l'organigramme donné ci-dessous.



ANNEXE :

```

function Button(var port : byte; pin, time, active_state : byte) : byte;
procedure Lcd_Config(var port : byte; const RS, EN, WR, D7, D6, D5, D4 : byte);
procedure Lcd_Out(row, col : byte; var text : array[255] of char);
procedure SetBit(var register : byte; rbit : byte);
procedure ClearBit(var register : byte; rbit : byte);
function TestBit(register, rbit : byte)

```

