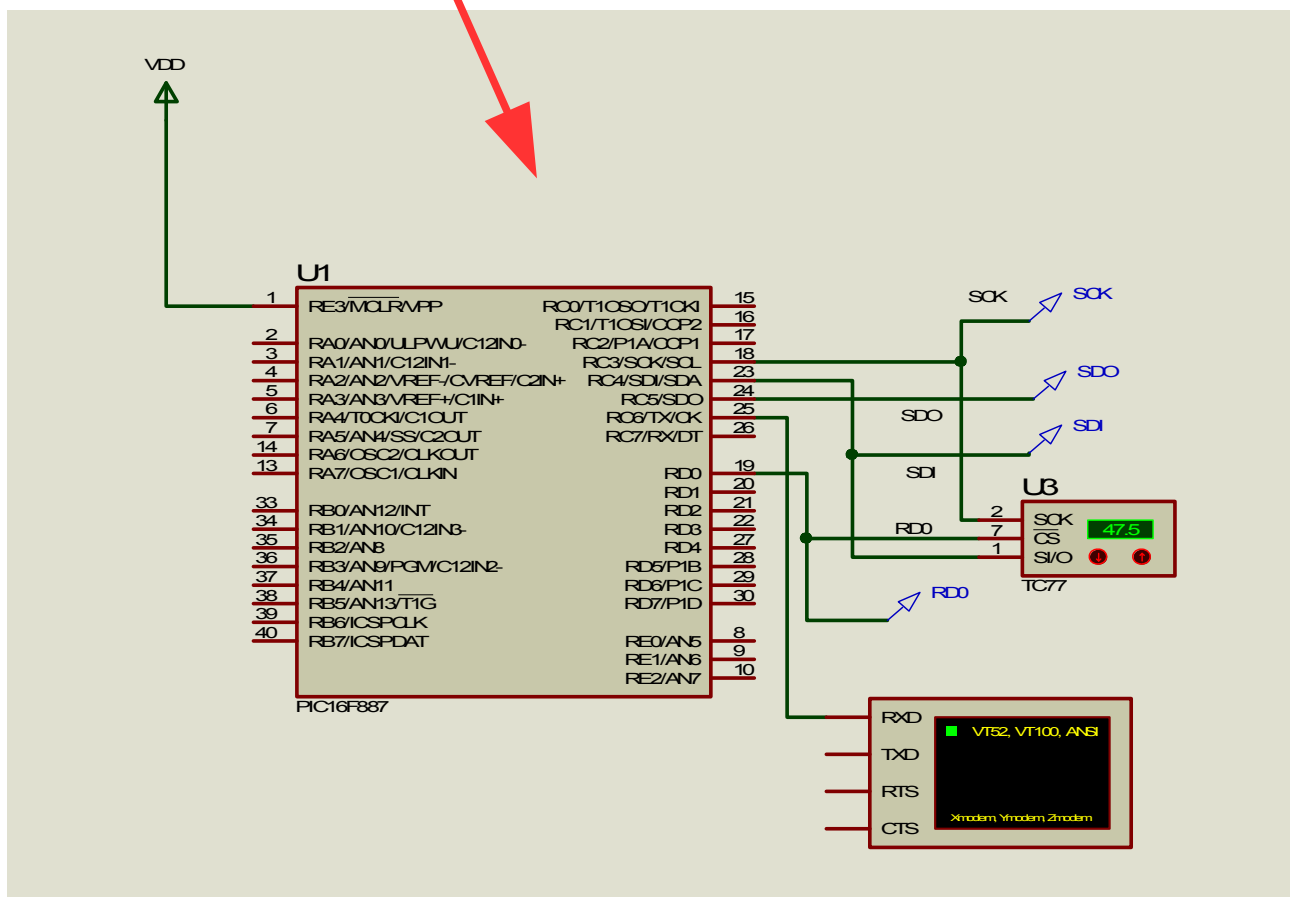


Mise en situation :

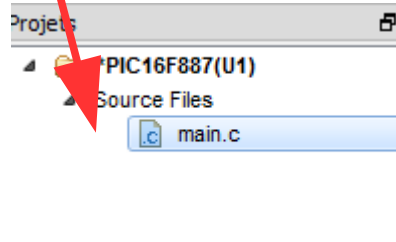
On souhaite mesurer la température de la salle 24 en utilisant plusieurs capteurs (TC77 possédant un bus SPI, LM35,..) et un affichage soit sur un Terminal soit sur un écran graphique avec un bus SPI (Nokia 5110).

Première partie : Acquisition température avec un capteur TC77 et affichage sur Terminal

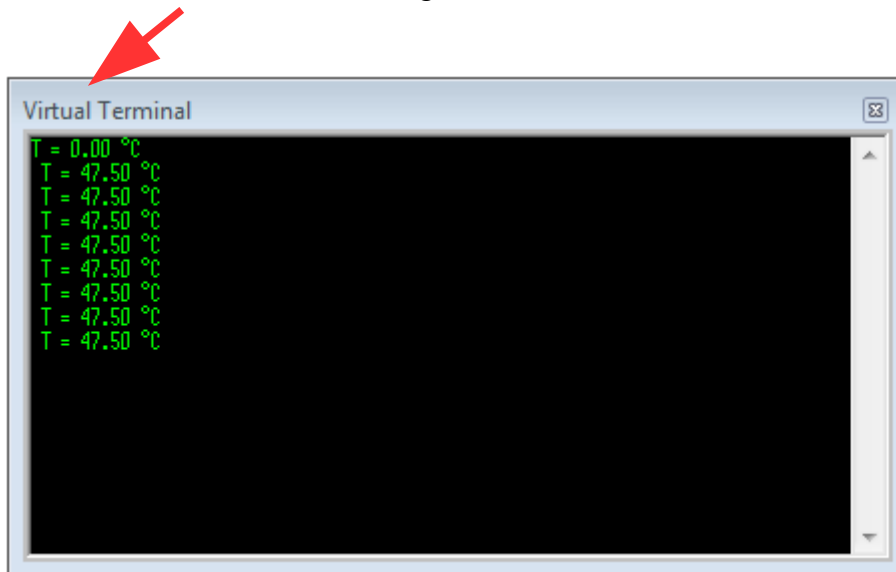
- Saisir le schéma ci dessous avec le microcontrôleur PIC16F887, un capteur de température TC77 et un Terminal.



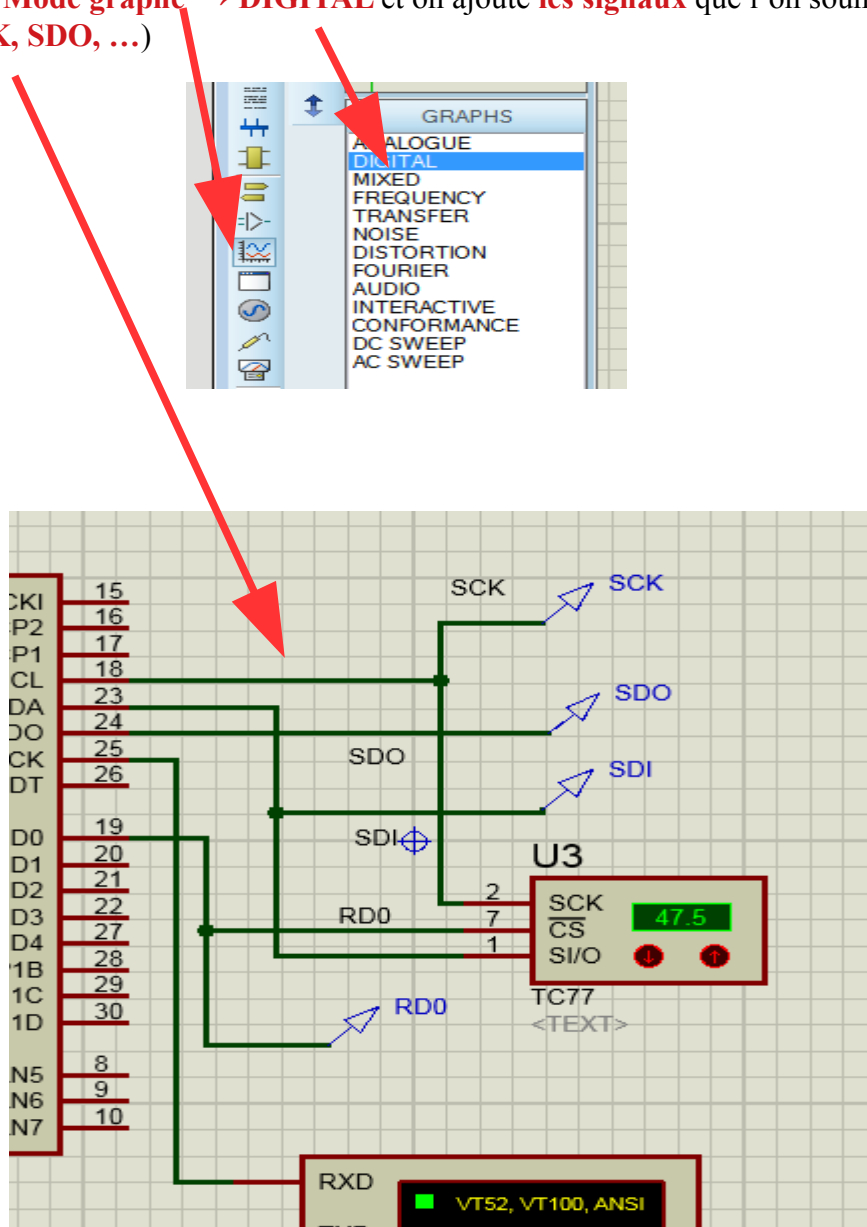
- Copier les bibliothèques **RS232.C** et **SPI.C** dans le même dossier que le projet.
- Ajouter toutes les bibliothèques (**RS232.C,SPI.C**) au projet au fichier **main.c_a_completer** avec les directives **#include "RS232.C"** et **#include "SPI.C"**.
- Compléter le fichier de la fonction principale **main.c_a_completer** et le sauvegarde sous le nom **main.c**
- Ajouter le fichier **main.c** au projet pour afficher la température sur un Terminal.



- Pour calculer la température, on utilise la formule donnée par le constructeur et correspondant à la température: **$((TempMSB \ll 8 \mid TempLSB) \gg 3) * 0.0625$**
- Compiler le fichier main.c en mode release
- Simuler le fonctionnement et corriger si nécessaire.



- On souhaite relever les valeurs des **2 octets** renvoyés par le capteur TC77, pour cela on clique sur **Mode graphe → DIGITAL** et on ajoute **les signaux** que l'on souhaite observer (ici : **SCK, SDO, ...**)



- Relever les valeurs des 2 octets (**MSB** → Octet de poids fort et **LSB** → Octet de poids faible) correspondant à la température et calculer sa valeur en utilisant la formule donnée par le constructeur : **$((TempMSB \ll 8 \mid TempLSB) \gg 3) * 0.0625$**

(Voir le document constructeur du capteur et les extraits donné ci dessous)

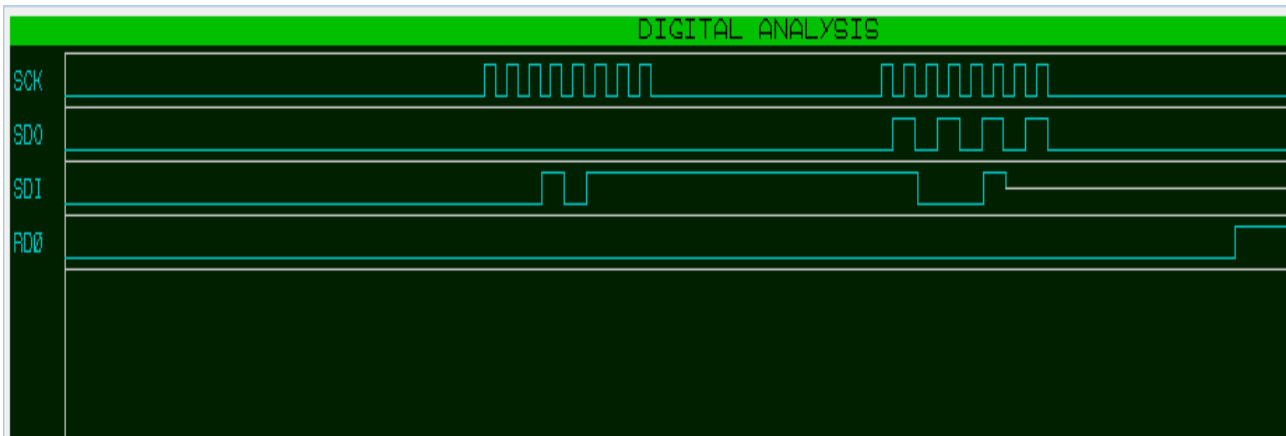


TABLE 4-1: REGISTERS FOR TC77

Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value at Powerup/Reset
CONFIG	C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0	XXXX/XXXX 0000/0000
TEMP	T12	T11	T10	T9	T8	T7	T6	T5	T4	T3	T2	T1	T0	*	x	x	1111/1111 0000/0*XX
M_ID	0	1	0	1	0	1	0	0	0	0	0	0	0	0	x	x	0101/0100 0000/00XX

TABLE 1: TC77 DIGITAL REGISTERS

Register	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value at Power-up/Reset
Configuration (Read/Write)	C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0	Continuous Temperature Conversion Mode **
Temperature (Read Only)	T15 (2 ⁸) ***	T14 (2 ⁷)	T13 (2 ⁶)	T12 (2 ⁵)	T11 (2 ⁴)	T10 (2 ³)	T9 (2 ²)	T8 (2 ¹)	T7 (2 ⁰)	T6 (2 ⁻¹)	T5 (2 ⁻²)	T4 (2 ⁻³)	T3 (2 ⁻⁴)	T2 *	T1 x	T0 x	Temp. = -2°C
Manufacturer ID (Read Only)	0	1	0	1	0	1	0	0	0	0	0	0	0	0	x	x	Bit 15 to Bit 8 = 54 hex

Legend: * Temperature Bit 2 = 0 during power-up; otherwise, bit 2 = 1
 ** C15:C0 = XXXX/XXXX 1111/1111 (Shutdown mode)
 C15:C0 = XXXX/XXXX 0000/0000 (Continuous Conversion mode)
 *** Temperature Register Bit 15 is the sign bit. If Bit 15 is equal to '1', the temperature is negative (T < 0°C). If Bit 15 is equal to '0', the temperature is positive (T ≥ 0°C).

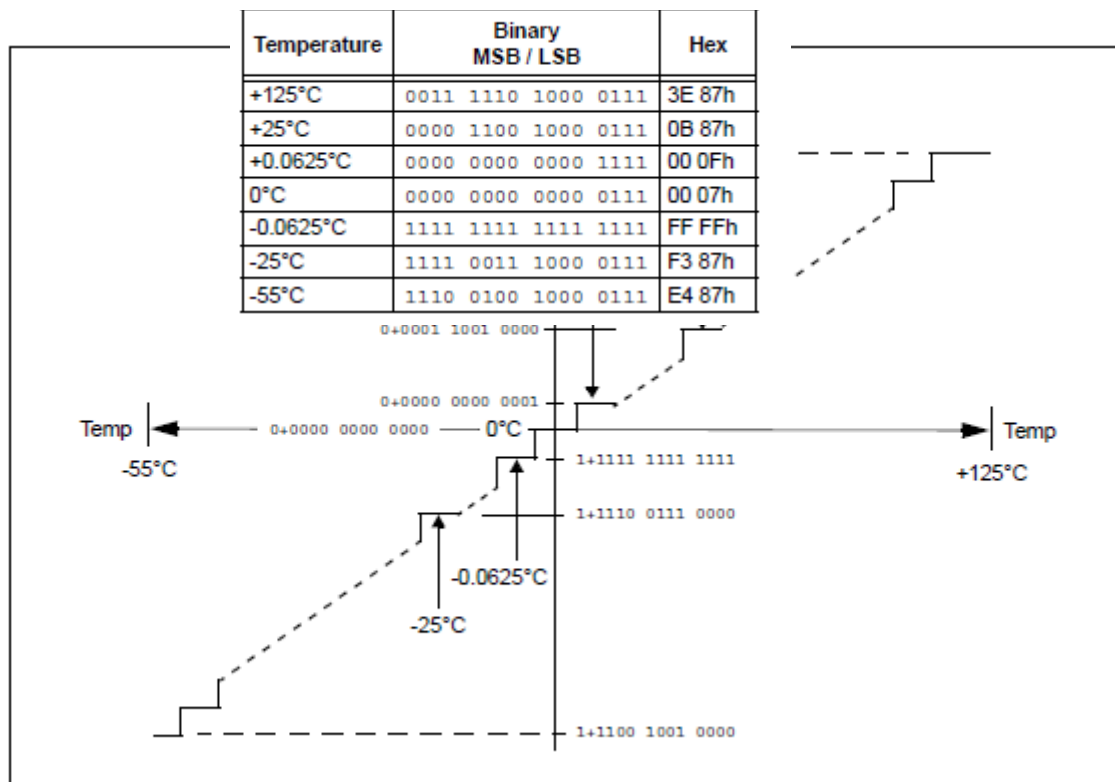
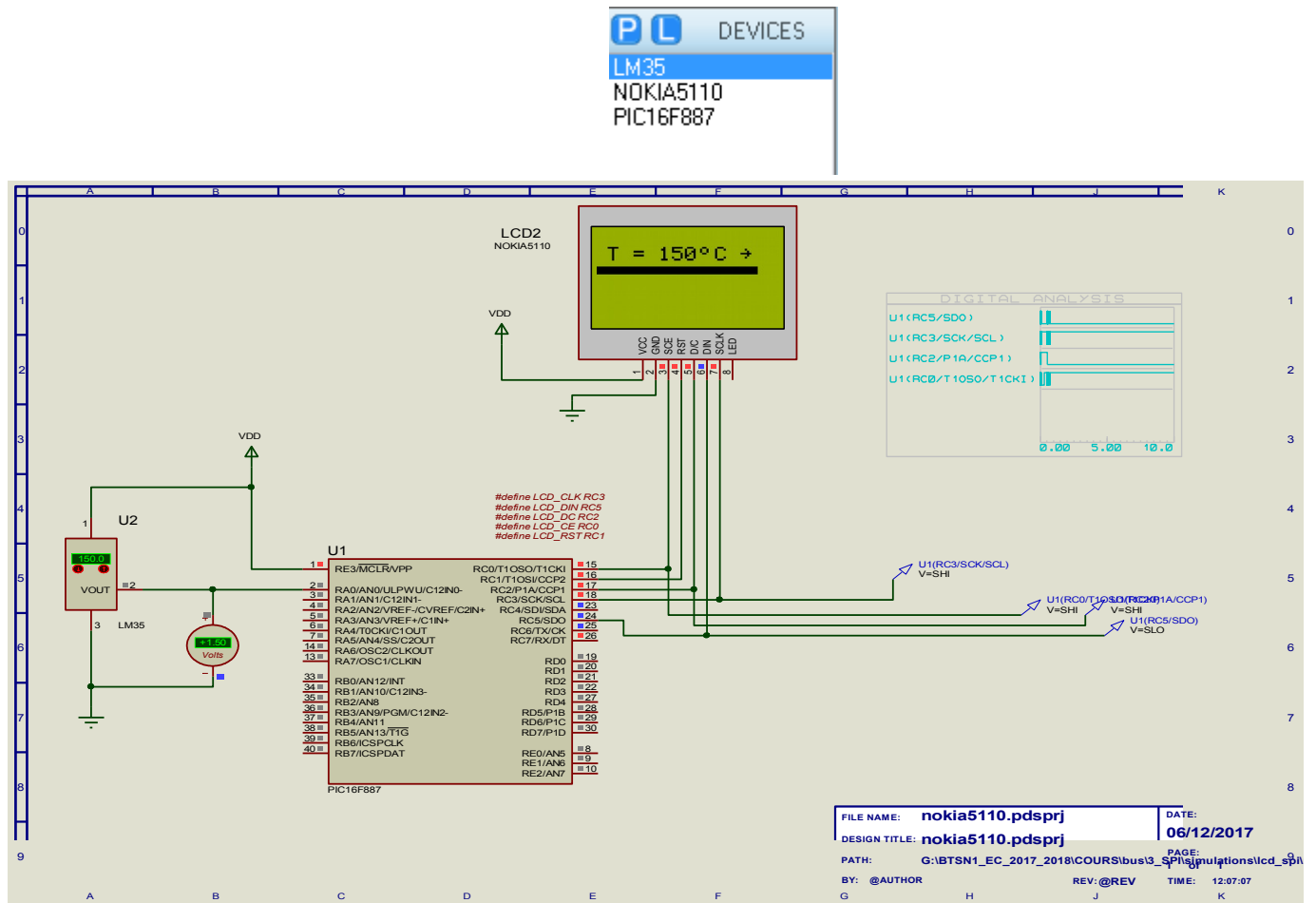


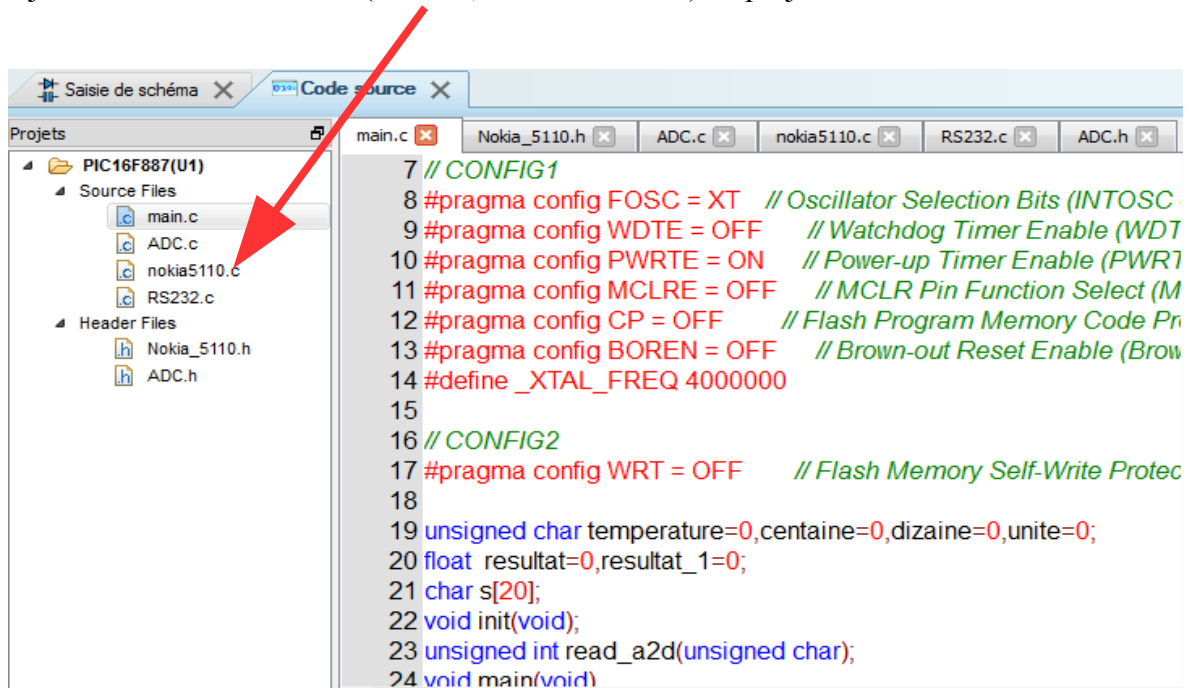
FIGURE 3-1: Temperature To Digital Transfer Function (Non-linear Scale).

Deuxième partie : Acquisition température avec un capteur LM35 et affichage sur écran graphique SPI.

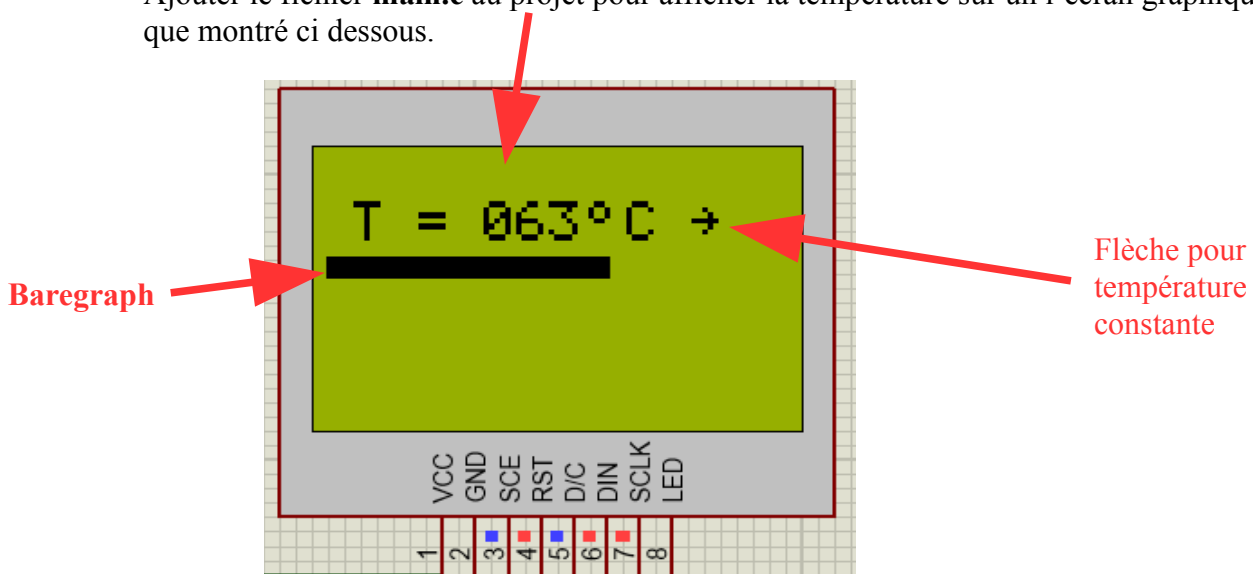
- Saisir le schéma ci dessous avec le microcontrôleur PIC16F887, un capteur de température LM35 et un afficheur graphique NOKIA5110.



- Ajouter toutes les bibliothèques (**ADC.c**, **nokia5110.c**....) au projet.



- Compléter le fichier de la fonction principale main() **afficheur_graphique_a_completer.c** et le sauvegarde sous le nom **main.c**
- Ajouter le fichier **main.c** au projet pour afficher la température sur un l'écran graphique tel que montré ci dessous.



- On souhaite ajouter un barregraphe qui va de 0 à 100 % pour des températures variants de 0 à 150 °C. En utilisant la fonction **nokia_bargraph_H(char Ligne,int Value)**, ajouter cette fonctionnalité dans le programme.
- On souhaite également ajouter des flèches pour indiquer les variations de la température (→ : température constante, ↑ : température en hausse ou ↓ pour température en baisse). En utilisant les fonctions **nokia_print_....arrow()**, ajouter cette fonctionnalité.
- **Simuler le fonctionnement et corriger si nécessaire.**

Troisième partie : *Acquisition température avec un capteur TC77 et affichage sur écran graphique SPI.*

- On souhaite utiliser uniquement un capteur TC77 et un écran graphique Nokia5110 pour afficher la température de la salle 24.
- En utilisant les travaux des 2 premières parties, réorganiser les schémas et les programmes.
- *Simuler le fonctionnement et corriger si nécessaire*