

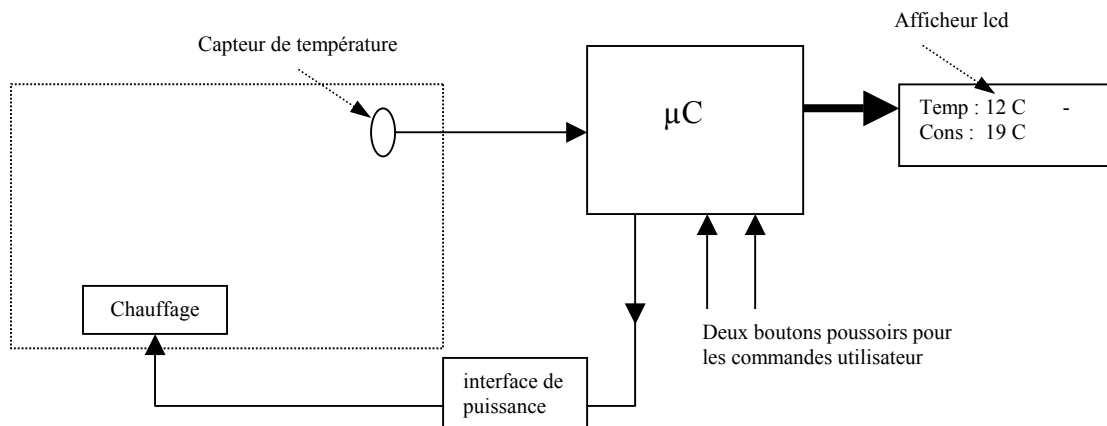
Objectifs :

- Savoir coder un régulateur tout ou rien à hystérésis**
- Programmer des calculs arithmétiques en C**
- Prendre en compte des actions utilisateurs sur des boutons**
- Réaliser un affichage convivial sur l'écran lcd**
- Mesurer les performances temporelles du programme**

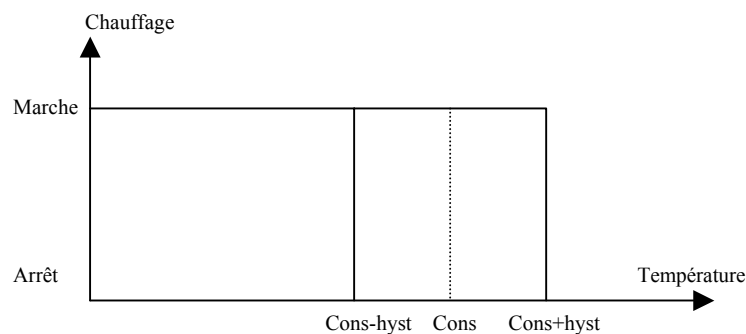
1/Présentation de l'application

Dans ce tp vous allez réaliser le logiciel embarqué (firmware) d'un régulateur de température tout ou rien à hystérésis. Ce régulateur très simple fonctionne en mesurant périodiquement la température ambiante et en commandant la mise en marche ou l'arrêt du chauffage en fonction de l'écart entre la température ambiante et la température souhaitée (consigne).

Le schéma de principe est le suivant :



La loi de régulation est la suivante :





TP n°2

Régulateur de température



La consigne et l'hystérésis valent 19°C et 4°C à la mise sous tension du système. L'utilisateur pourra changer la consigne par les boutons S2 (-1°C) et S3 (+1°C). Il sera également possible de changer l'hystérésis et de passer à un affichage en degrés Fahrenheit.

Dans un premier temps, la commande de chauffage sera simulée par l'allumage de la LED câblée sur RB3 et le capteur de température sera simulé par le potentiomètre R16 câblé sur l'entrée RA0.

2/ Acquisition et affichage de la température simulée

Le potentiomètre R16 permet de faire varier la tension analogique présente entre la broche RA0 et la masse entre 0 et 5V. Cette tension peut être convertie par le convertisseur analogique/numérique du pic pour fournir un nombre de type `unsigned char`. Une fois le convertisseur démarré par la fonction `adc_on()`, il est possible de récupérer le résultat de la conversion par un appel à la fonction `adc_read_8b()`.

- Quelle sont les limites possibles de ce nombre ?

- Expliquez pourquoi la ligne C ci-dessous permet de ramener ce nombre dans l'intervalle [-128,+127] ?

```
temp = -128 + adc_read_8b(0);
```

- Quel serait le type de la variable `temp` qui permettrait d'utiliser le moins de RAM ? (Nous utiliserons le type `int` par la suite pour pouvoir faire de calcul sur une plage plus étendue).

- Réaliser le programme permettant d'afficher en permanence la température simulée sur la première ligne de l'écran lcd

Ex : Temp : 15 C

NB : on ne cherchera pas à afficher le caractère ° sur le lcd.

3/ Modification de la consigne par l'utilisateur

A la mise sous tension, la consigne est fixée à 20 °C. Le bouton S2 fera augmenter la consigne de 1°C, le bouton S3 la fera diminuer de la même valeur. La consigne minimum est -30°C, la consigne maximum est +70°C. Si on appuie en même temps sur S2 et S3, la consigne revient à 20°C.

- Ajouter au programme précédent l'affichage de la consigne sur la deuxième ligne et la possibilité de la faire varier comme décrit ci-dessus.

Ex : Temp : 15 C

Cons : 22 C



TP n°2

Régulateur de température



4/ Régulation à hystérésis

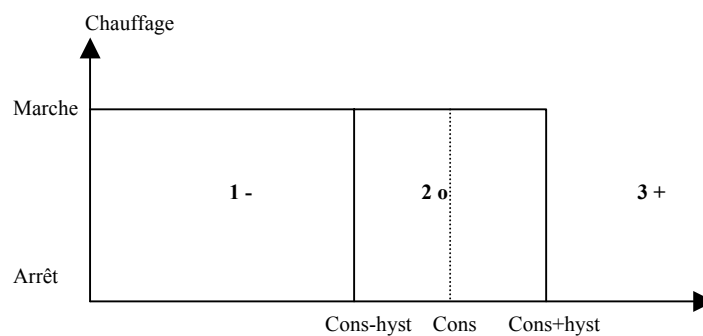
L'hystérésis est fixé à 4°C à la mise sous tension. La commande de chauffage est simulée par l'allumage de la LED câblée sur la sortie RB3.

- Ajouter au programme précédent la commande du chauffage selon la loi de la régulation définie en 1/

NB : La valeur de l'hystérésis sera stockée dans une variable pour se réserver la possibilité de le faire varier par la suite.

5/Amélioration de l'affichage .

On souhaite afficher une indication de l'état de la température effective par rapport à la consigne. Sur la dernière position de la première ligne, on va afficher les caractères -, o ou + selon que la température se situe dans les zones 1,2 ou 3 respectivement.



(Remarquez que l'algorithme est légèrement différent de la commande de chauffage)

- Ajouter cette fonctionnalité au programme.



TP n°2

Régulateur de température



6/ Modification de la valeur de l'hystérésis

Pour changer la valeur de l'hystérésis, l'utilisateur doit mettre l'appareil hors tension et appuyer sur S3 au moment de la remise sous tension.

Après cette manipulation l'affichage est le suivant :

```
Temp : 15 C +  
Hyst : 4 C
```

Les boutons S2 et S3 servent alors à augmenter ou à diminuer l'hystérésis par pas de 1°C. L'hystérésis est limité à la plage [1°C , 18°C].

L'appui simultané sur S2 et S3 permet de repasser au mode de fonctionnement normal.

-Ajouter cette fonctionnalité à votre programme.

7/ Et pour le vendre aux américains...

On doit prévoir un affichage en degré fahrenheit. La loi de conversion est :

$$t(^{\circ}C) = \frac{5}{9} \cdot (t(^{\circ}F) - 32)$$

Pour passer dans ce mode, l'utilisateur doit appuyer sur S2 lors de la mise sous tension.

- Reprendre votre programme pour ajouter cette fonctionnalité. Attention tout rester fonctionnellement identique (consigne, hystérésis, pas de variation de la consigne, etc...).

A vous d'imaginer un moyen simple de passer dans le mode de variation d'hystérésis en affichage fahrenheit.

Exemple d'affichage :

```
Temp      : 72 F +  
Preset    : 70 F
```

8/ Performance temporelle

La constante de temps du système à réguler est de l'ordre de plusieurs minutes.

Le temps de réaction de l'affichage suite à une sollicitation des boutons par l'utilisateur doit être inférieur à 500ms par ne pas être jugé gênant.

- En faisant changer d'état une sortie à chaque parcours de la boucle générale, mesurez le temps de réaction du programme et conclure.