

Compte rendu - Projet Arduino Système de surveillance

Fonctionnalités principales :



1 - Activer / désactiver le mode surveillance grâce à un bouton poussoir

Le mode par défaut est le mode normal et change dès qu'on effectue un appui mi-long sur le bouton.

On l'affiche en haut à gauche de l'afficheur (N pour Normal et S pour Surveillance).

Une autre solution : on aurait pu remplacer le bouton par un interrupteur avec 2 valeurs : mode surveillance et mode normal.



2 - Afficher la température donnée par le capteur de température en ° Celsius

On récupère la valeur analogique renvoyée par le capteur puis on incorpore cette valeur à une formule impliquant la tension pour nous permettre d'en obtenir une valeur en degrés celsius.

On l'affiche en bas à droite de l'écran LCD en prenant en compte l'affichage selon la place que prend le nombre (unité, dizaines et centaines).



3 - Afficher l'intensité lumineuse par la photorésistance en lux

On récupère la valeur analogique renvoyée par la photorésistance.

On l'affiche en bas à gauche sur l'afficheur LCD en prenant en compte l'affichage selon la place que prend le nombre (unité, dizaines et centaines).



4 - Afficher l'heure (format hh:mm) en initialisant par communication série

On peut récupérer l'heure décomposée en heures et minutes tapée par l'utilisateur dans le moniteur série puis la traiter.

On l'affiche en haut au milieu sur l'afficheur LCD.

Une autre solution : on aurait pu utiliser le module RTC DS1307 qui permet de gérer le temps avec un oscillateur à quartz et s'interface en I²C.



5 - Faire clignoter la led intégrée à la carte toutes les secondes

On fera clignoter la led intégrée de la carte grâce à la fonction `millis()` qui nous permettra de la faire clignoter toutes les secondes sans delay donc sans faire de pause pour ne pas empiéter le code.

Fonctionnalités mode surveillance :



1 - Émettre un son avec un buzzer si la température passe le seuil bas ou haut

Le seuil bas sera défini à 20°C et le seuil haut sera défini à 35°C lors de la phase de test.

Quand la température est en dessous du seuil bas, on émet un son aigu (1200 Hz) et quand elle est au dessus, on envoie un son grave (200 Hz).

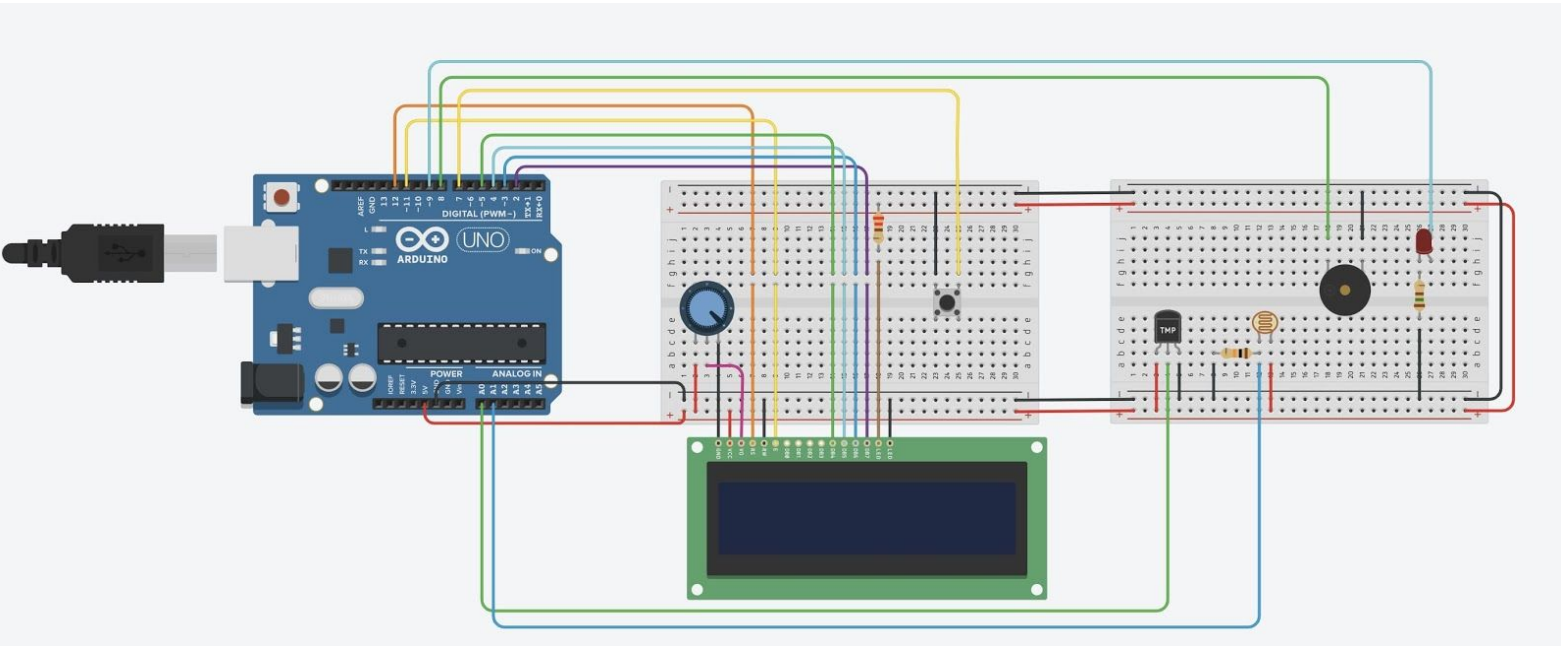
On mettra un delay de 0.1s entre chaque sonnerie.



2 - Activer une led (lumière) qui s'allume proportionnellement à la luminosité

La led s'allume proportionnellement à la luminosité avec la fonction `map()` qui renvoie une valeur mappée qui sera utilisé pour allumer la led à une certaine luminosité.

Montage :



Matériels et ports :

Afficheur LCD : 2, 3, 4, 5, 11 et 12

Bouton poussoir : 7

Capteur de température TMP36 : A0

Photorésistance GL5528 : A1

Buzzer : 8

Led : 9

Test :

Le schéma est disponible sur le site de Tinkercad

Lien : <https://www.tinkercad.com/things/41jw4qsCWxy-projet-iut>

Explications :

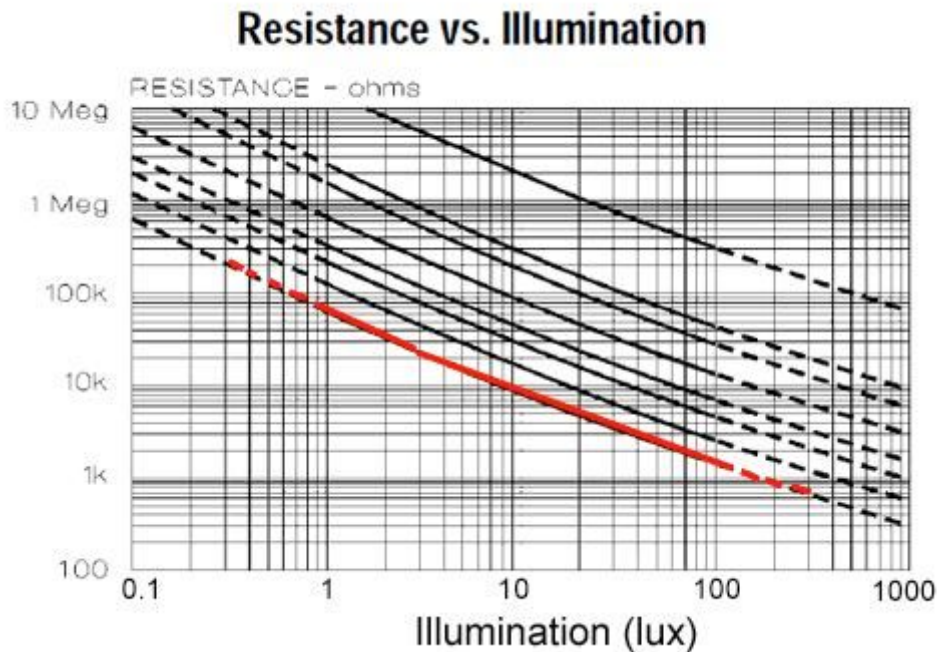
1 - Calcul lux

Une photorésistance est un composant électronique dont la résistivité (capacité à s'opposer à la circulation du courant électrique) varie en fonction de la quantité de lumière.

C'est à dire que plus elle est éclairée, plus sa résistivité diminue et inversement.

Pour faire simple c'est une résistance dont la valeur change en fonction de la lumière.

Schéma de la relation entre la résistance et l'intensité lumineuse :



On récupère ensuite la valeur sur le port analogique et on le traite dans notre programme.

Dans notre projet on utilise la photorésistance GL5528 qui peut être sensible à la chaleur et réagit différemment en fonction de la chaleur donc la mesure peut être légèrement différente mais cela n'est pas important dans notre projet.

2 - Précision horloge

On crée une variable 'secondes' qui s'incrémente à chaque tour de la boucle loop à chaque seconde et quand cette variable est à 60 on la remet à 0 et on incrémente les minutes.