

Aujourd'hui, le but était de me servir de mon programme de réveil "final" (sous-entendu, avec l'allumage des LED, le bouton SNOOZE et le bouton d'arrêt) mis au point pour le piezo buzzer afin de reproduire la même chose pour mon nouveau système sonore qui est bien plus performant car il s'agit de hauts parleurs de casques.

J'ai choisit de me diriger vers ce système là sur les conseils de M; Masson pour plusieurs raison. La première est que le son produit est bien plus agréable, mais surtout, cela permet la lecture de fichier MP3 et WAV et donc, rend possible l'implémentation dans notre réveil de bruits d'ambiance (orage, pluie...) contrairement au piezo buzzer qui avait un bruit trop mécanique. Un autre avantage non négligeable est que le lecteur MP3 utilisé possède sa propre puce et donc, la lecture d'une musique ne gêne pas le fonctionnement de ma carte arduino.

Pour ce faire j'ai récupéré un module MP3 catalex qui est branché à une enceinte, sur lequel je me suis bien renseignée (grâce à la documentation fournie par M. Masson et [ce site](#) ou encore [celui-là](#)).

Une fois les commandes de bases de ce module ainsi que la compréhension de programmes simples l'utilisant acquises, j'ai commencé par réécrire un programme SNOOZE basique dans lequel j'ai part la suite implémenté mon bouton d'arrêt de l'alarme et ma bande LED.

Un gros plus de ce système comparé au précédent avec le piezo buzzer est que je ne possède plus aucun delay lors de la lecture de la musique, les seuls delay() restant étant ceux utilisés pour l'augmentation de la luminosité de la bande LED. Cela rend donc mon programme bien plus fiable.

Ensuite, j'ai transformer le bouton du snooze en un capteur de vibrations, ce qui permettra à l'utilisateur de devoir "secouer", ou simplement taper le réveil afin d'enclencher le retardement du lever. Pour se faire, j'ai d'abord écouter les explications de M. Masson sur la notion d'interruption avec arduino puis après avoir regardé le fonctionnement de ce module sur [ce site](#), j'ai reproduit le fonctionnement du programme sur ce site ([Utilisation des interruptions avec Arduino • AranaCorp](#)) pour le faire correspondre au mien.

Après avoir testé le fonctionnement du capteur avec une simple LED qui s'allumait et s'éteignait, j'ai pu connecter le tout à mon programme final.

Bien que les temps de ce programme restent à définir, je joins le code (avec capteur) sur le Github.

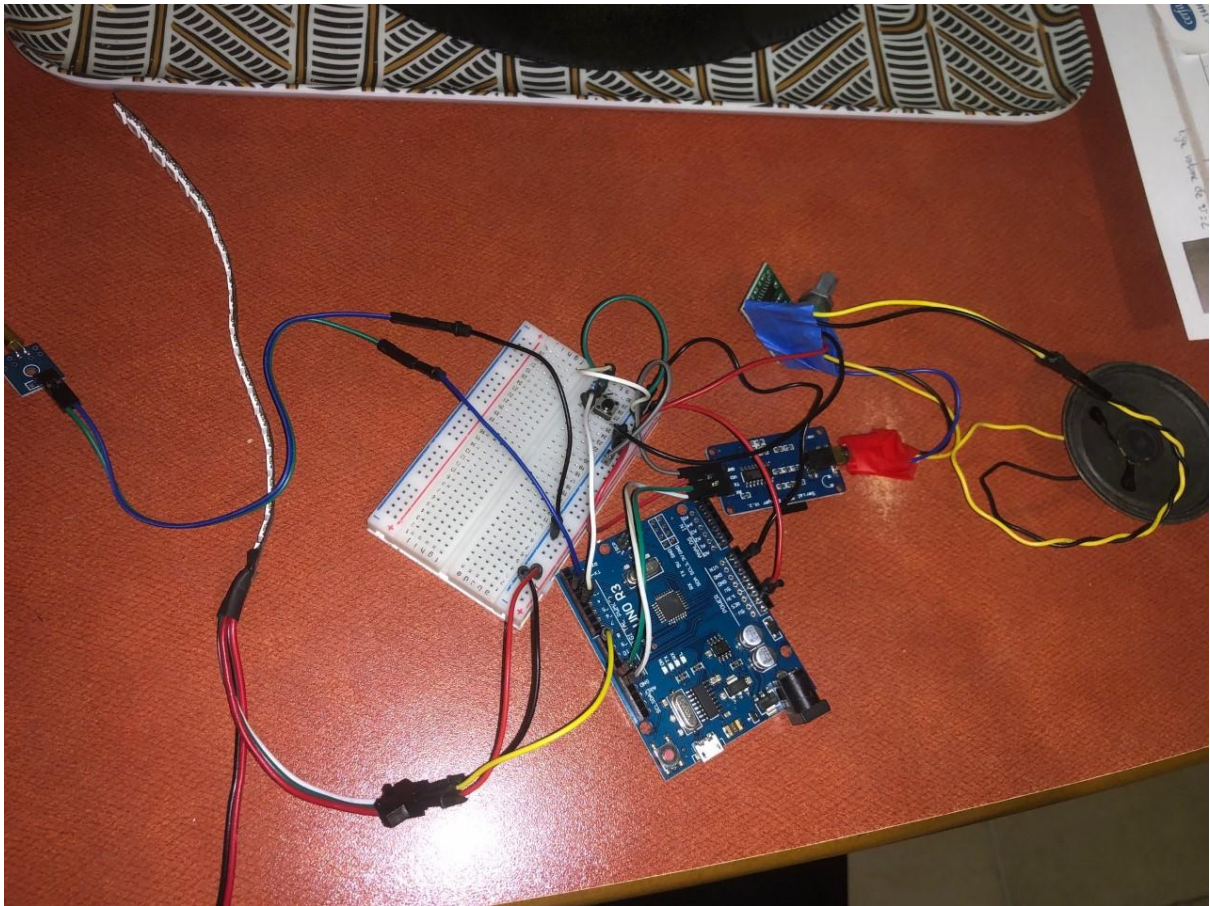


image du montage avec le système sonore sur la droite et la bande LED + capteur de vibrations sur la gauche. Le bouton d'arrêt est quant à lui sur la breadboard.