

Projet Arduino: Maison connectée

Rapport de séance n°3 (5/01/2022)

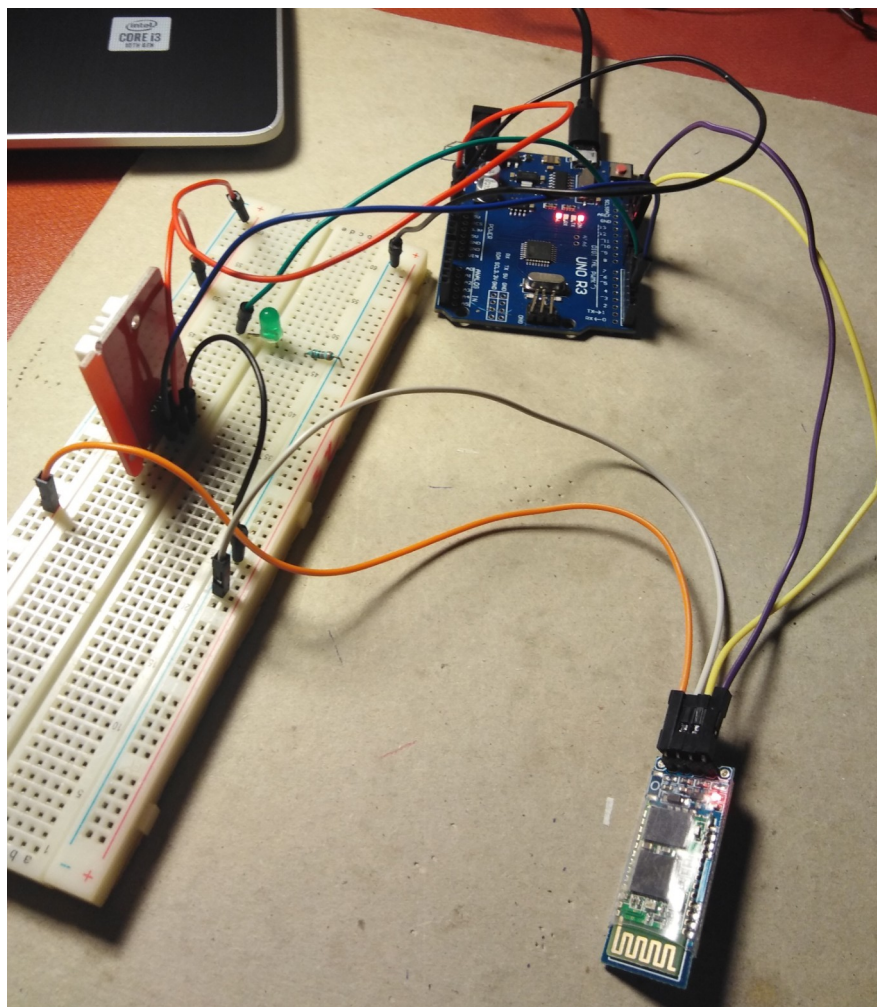
Par Nathan PERDOUX

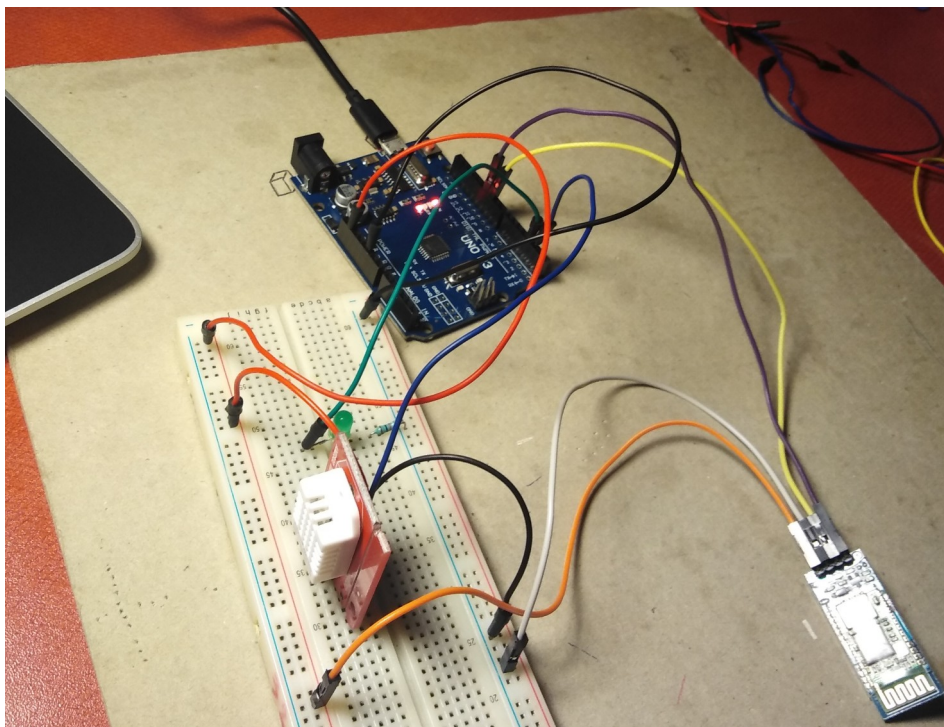
- Partie 1: Connexion module AM2302 (capteur température/hygrométrie) et Bluetooth + interface Android

Au cours de la dernière séance, j'ai câblé un montage présentant une LED qui s'allume dans le cas où la température mesurée par le module AM2302 dépasse un certain seuil que j'ai fixé au préalable. La LED est une sorte de témoin de chauffe c'est-à-dire que l'allumer simule la mise en route d'un chauffage.

Désormais, je dois connecter mon module Bluetooth à la carte Uno afin de pouvoir donner des instructions comme allumer/éteindre le chauffage (i.e. la LED) depuis mon smartphone. J'aimerais aussi pouvoir mettre la gestion du chauffage en mode automatique ou en mode manuel. En mode automatique, le chauffage s'allume si le seuil est dépassé uniquement et en mode manuel, je peux éteindre et allumer le chauffage à ma guise et ce sans prendre en compte la température ambiante.

Voici des photos du montage une fois le module Bluetooth ajouté:

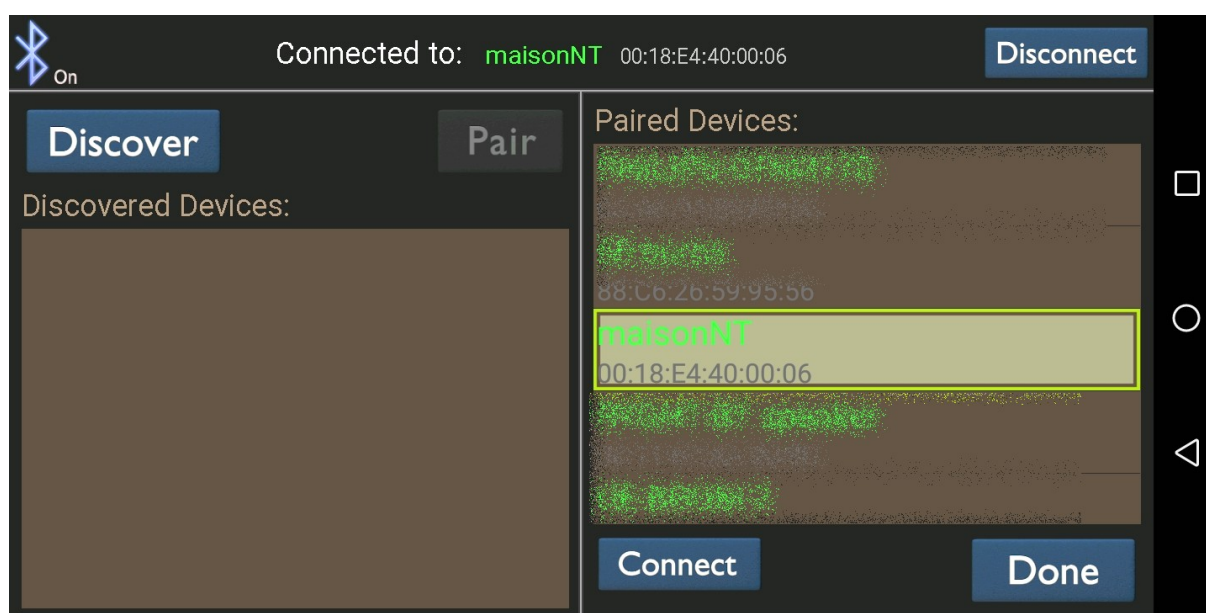




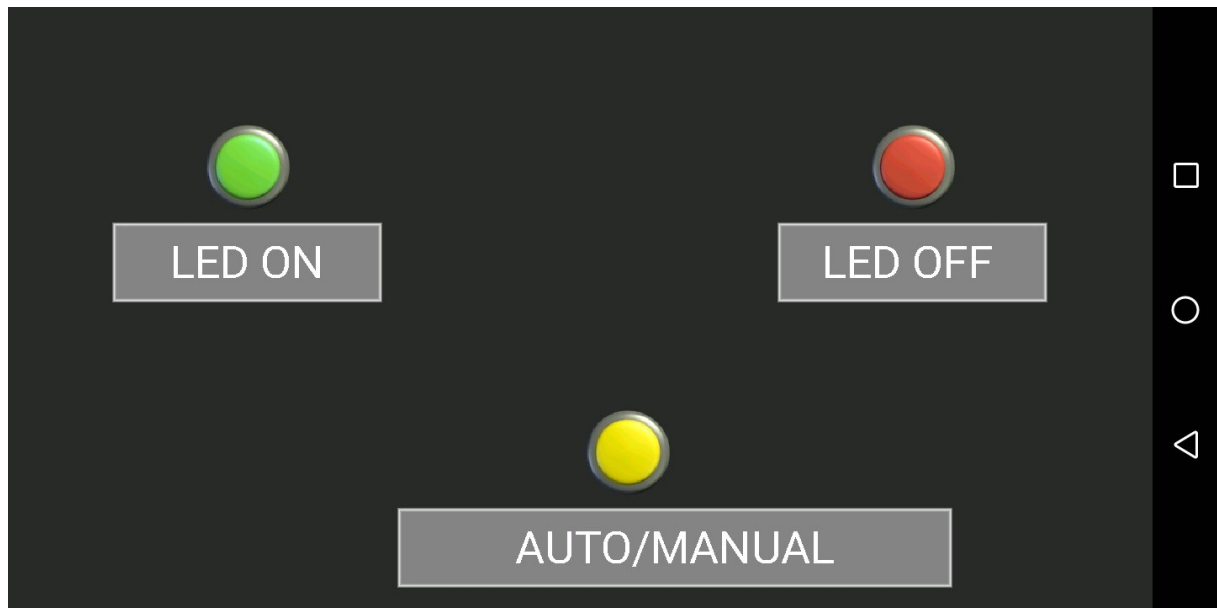
Voici un tableau récapitulant les connexions aux I/O de la carte:

I/O de la carte Uno	Composant connecté
2	Broche DAT (data) de l'AM2302
3	LED (borne +)
10	TX du module BT
11	RX du module BT

A présent, je vais utiliser l'application Android *Bluetooth Electronics* pour créer un panneau (panel sur l'application) afin de pouvoir commander les fonctionnalités activer ou désactiver le mode auto et allumer ou éteindre le chauffage lorsque le mode manuel est allumé. Je me connecte au module que j'ai provisoirement renommé «maisonNT».



J'ai créé le panneau suivant comprenant 3 boutons : le vert pour allumer le chauffage en mode manuel, le rouge pour allumer le chauffage en mode manuel et le jaune pour activer/désactiver le mode automatique.



Lorsque le bouton vert est pressé, le smartphone envoie au module BT le caractère «O» qui est transmis à la carte Uno via la liaison RX/TX. Pour le bouton rouge, c'est le caractère «C» et pour le bouton jaune c'est le caractère «A». Le code commenté est trouvable [ici](#), sur le GitHub de notre projet. C'est en fait une «fusion» du code utilisé pour le module HC-06 et de celui utilisé pour l'AM2302.

- Partie 2: Connexion moteur pas-à-pas, module photorésistance et Bluetooth + interface Android

J'entreprends à présent la conception d'un montage permettant de faire tourner un moteur pas-à-pas d'un certain nombre de tour (que j'aurai choisi au préalable) dans un sens si le module photorésistance détecte une intensité lumineuse au-dessus d'un certain seuil et dans l'autre si l'intensité lumineuse mesurée est en-dessous du seuil. Je peux ainsi simuler l'ouverture et la fermeture d'un store selon la lumière à l'extérieur de la maison. En fait, il s'agit du même objectif qu'avec l'AM2302 et la température mais avec des composants différents. Les codes et l'interface Android auront donc la même structure.

Voyons quels sont les composants que j'utilise aujourd'hui:

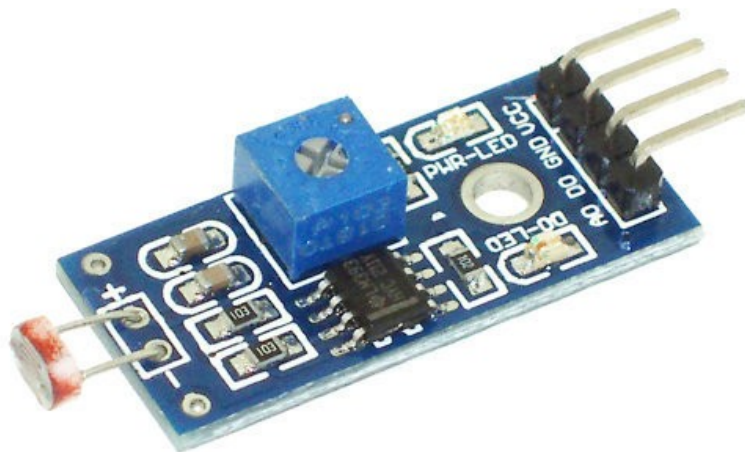


Figure 1: Module Arduino photorécepteur

Il mesure la luminosité ambiante à l'aide du capteur à son extrémité (rouge). Il possède 4 broches: VCC (5V), GND (masse), D0 (état numérique i.e. 0 ou 1, on fixe le seuil avec le potentiomètre qui est la partie bleu clair) et A0 (état analogique compris entre 0 et 1023 avec 0 très lumineux et 1023 très sombre). Nous utiliserons A0 et fixerons le seuil d'ouverture fermeture du store à 230 arbitrairement pour le moment.

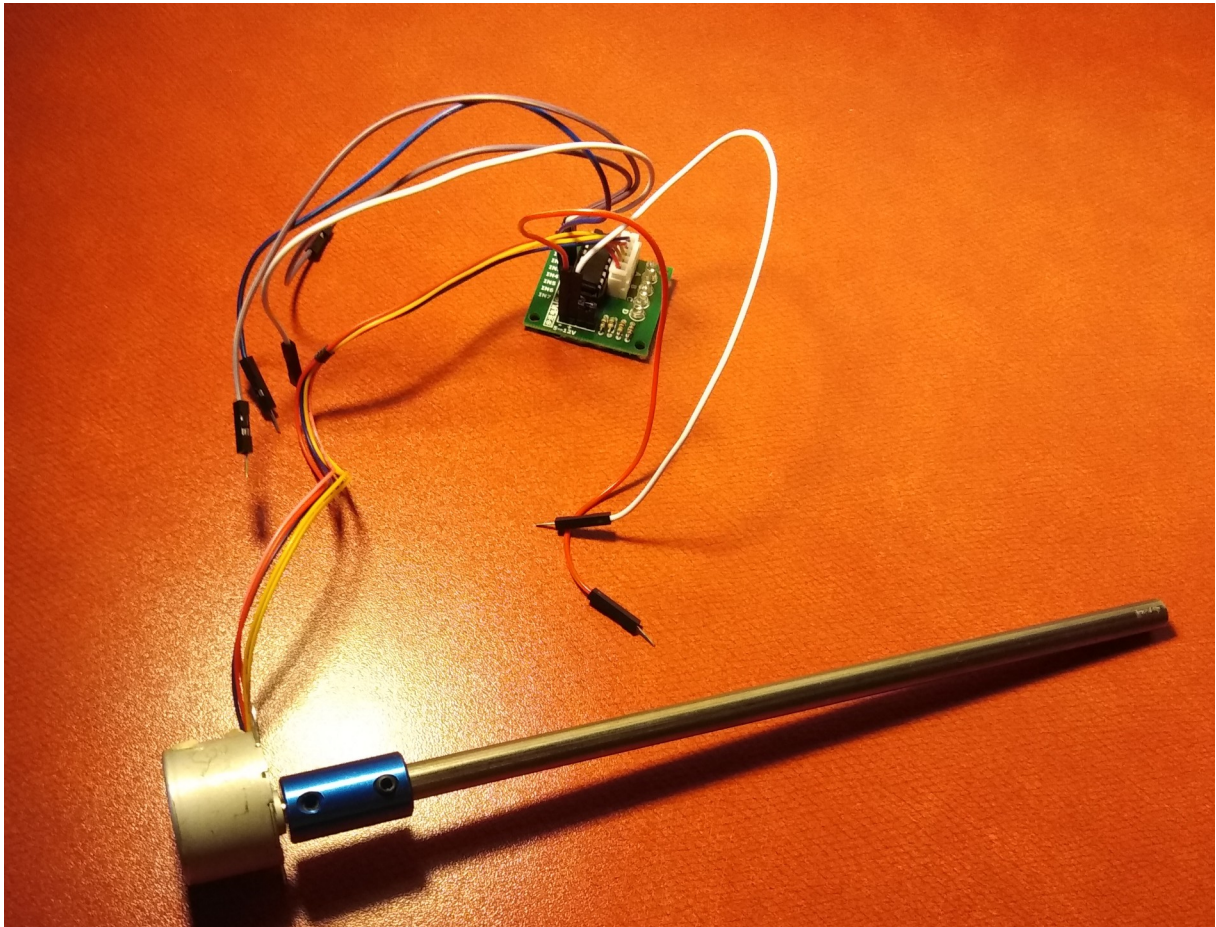
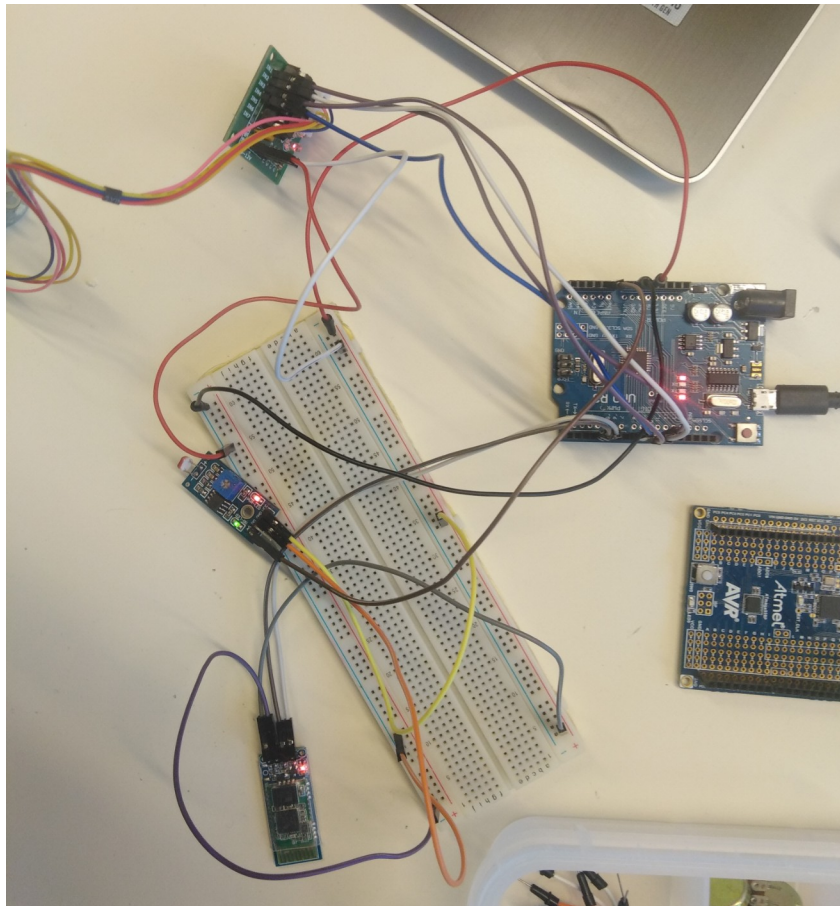
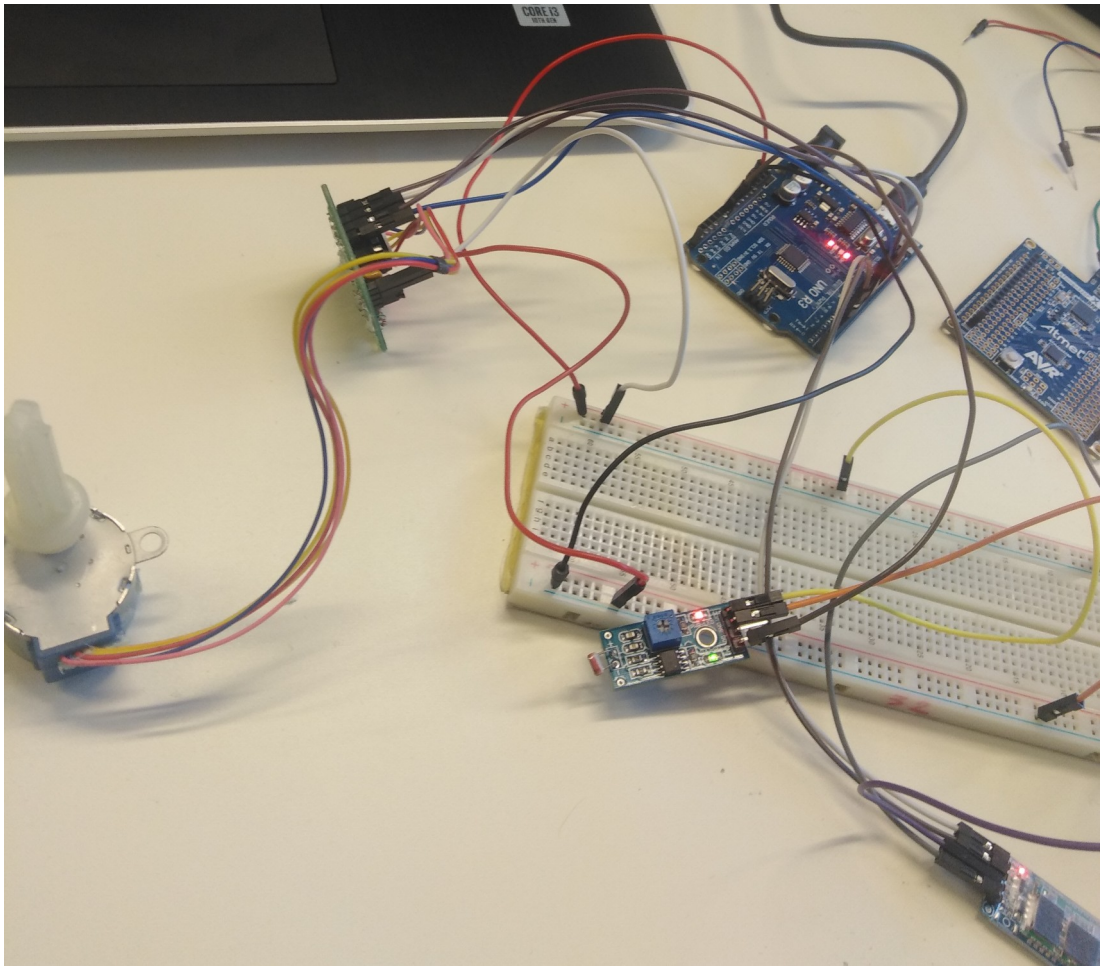


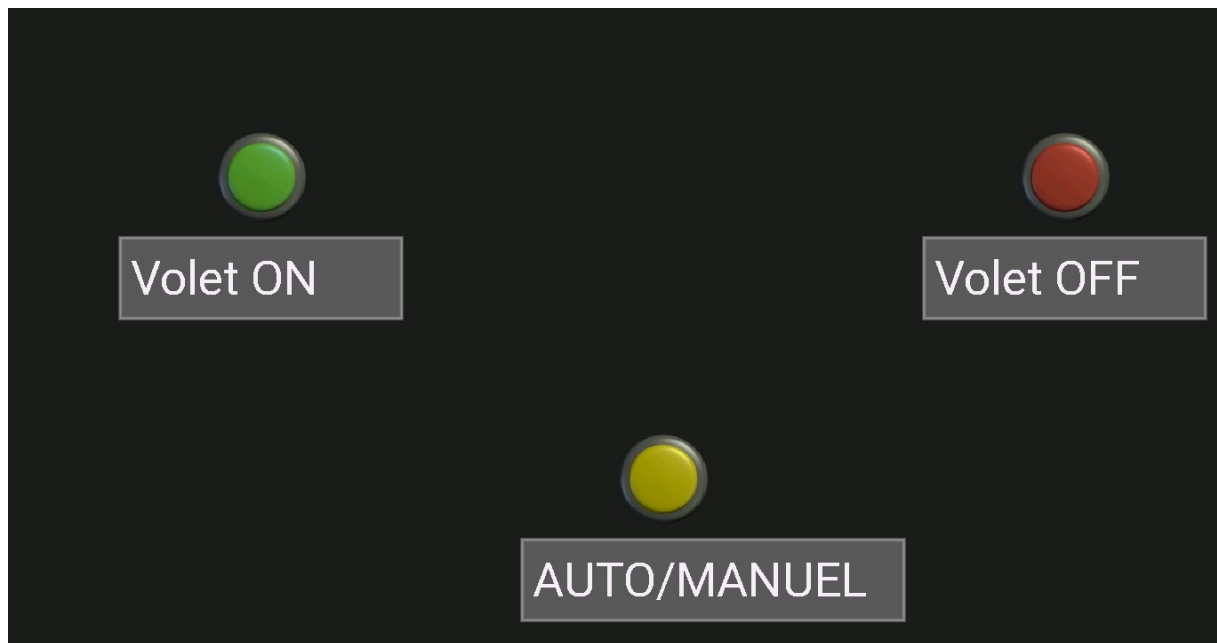
Figure 1: Moteur pas-à-pas

Le moteur pas-à-pas présente l'avantage de pouvoir être contrôlé avec précision (distance et vitesse). Le moteur en lui-même est relié à une carte comportant une puce ULN2003apg qui est un réseau de transistors. Un axe en fer est fixé au stator. Nous fixerons à cet axe de la toile qui s'enroulera sur celui-ci lorsque le store est ouvert et qui se déroulera jusqu'à atteindre le sol lorsque le store est ouvert.

J'utilise le code que Thomas m'a donné pour faire fonctionner le moteur (il est trouvable [ici](#)). Ensuite, comme pour la partie 1, «je fusionne» le code avec celui du module HC-06 qui est trouvable [ici](#). Le montage est le suivant (l'axe en fer n'est pas encore fixé):



Pour ce qui est de l'interface Android, c'est la même que pour la partie 1:



Après tests, tout fonctionne bien !