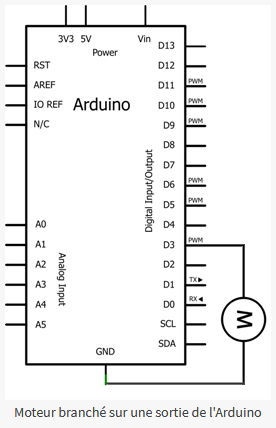
**Objectif** : **Faire démarrer un moteur à courant continu en appuyant sur un bouton poussoir et l’arrêter quand on relâche ce même bouton .Ou on peut utiliser un interrupteur à deux positions.**



1. **Rappel :**

* **Ne jamais brancher le moteur électrique directement, sans autres composants intermédiaires sur la carte ARDUINO au risque de la griller. (Car** la carte ne doit pas débiter plus de 40 mA par port de sortie soit 200mA au total**).**

***Donc* NE PAS FAIRE CE CABLAGE**

Si vous êtes en présentiel :

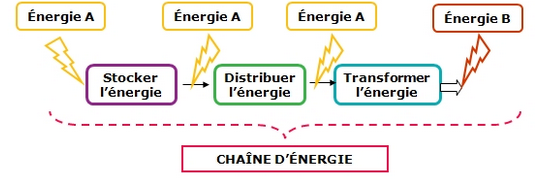
Se munir d’une carte ARDUINO et d’une plaque de câblage rapide (breadboard) , 1câble « USB », d’un moteur cc ; d’un accumulateur 9 v ; 1 sortie relay « grove » connectée en D8 ; 1 entrée bouton poussoir « grove » connectée en D7; d’une diode ; 2 fils fins à souder sur les patte du moteur.

Si vous êtes en distanciel :

Se connecter au site Tinkercad, à la place du relay « grove ».

(Prendre dans la bibliothèque sur le côté un relais unipolaire bidirectionnel « LU-5-R »).

1. **Compléter ci-dessous sur le schéma de la chaine d’énergie** 
   1. ) En rajoutant les noms des composants qui vous sont nécessaires pour réaliser les différentes fonctions dans l’objectif en début de page.
   2. ) En apportant des précisions sur le types d’énergie, elles seront notées au-dessus de chaque étiquette énergie.



**Accumulateur**

**Relais**

**Moteur**

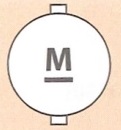
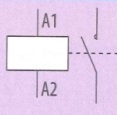
**Courant électrique**

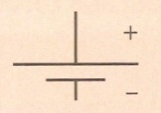
**électrique**

**électrique**

**Mécanique**

1. **Ecrire le nom de chacun des composants ci-dessous**



xle-relais-principe-de-fonctionnement-5.jpg.pagespeed.ic.isvrfvl-XT.jpg

**Moteur électrique Relais Pile Led Condensateur Bouton poussoir**

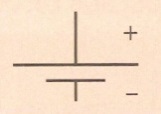
1. **Compléter le schéma de la chaîne d’information et de la chaîne d’énergie commencé ci-dessous**

Vous devez utiliser les composants proposés au bas de la page précédente pour réaliser un schéma de principes en respectant le fonctionnement demandé. Une fois le schéma terminé le montrer à l’enseignant.

**Chaîne d’information**

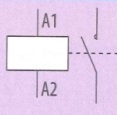
**Chaîne d’énergie**

Shield sur carte Arduino



D4

D8

 Moteur

**Bp M/A**

Bouton poussoir



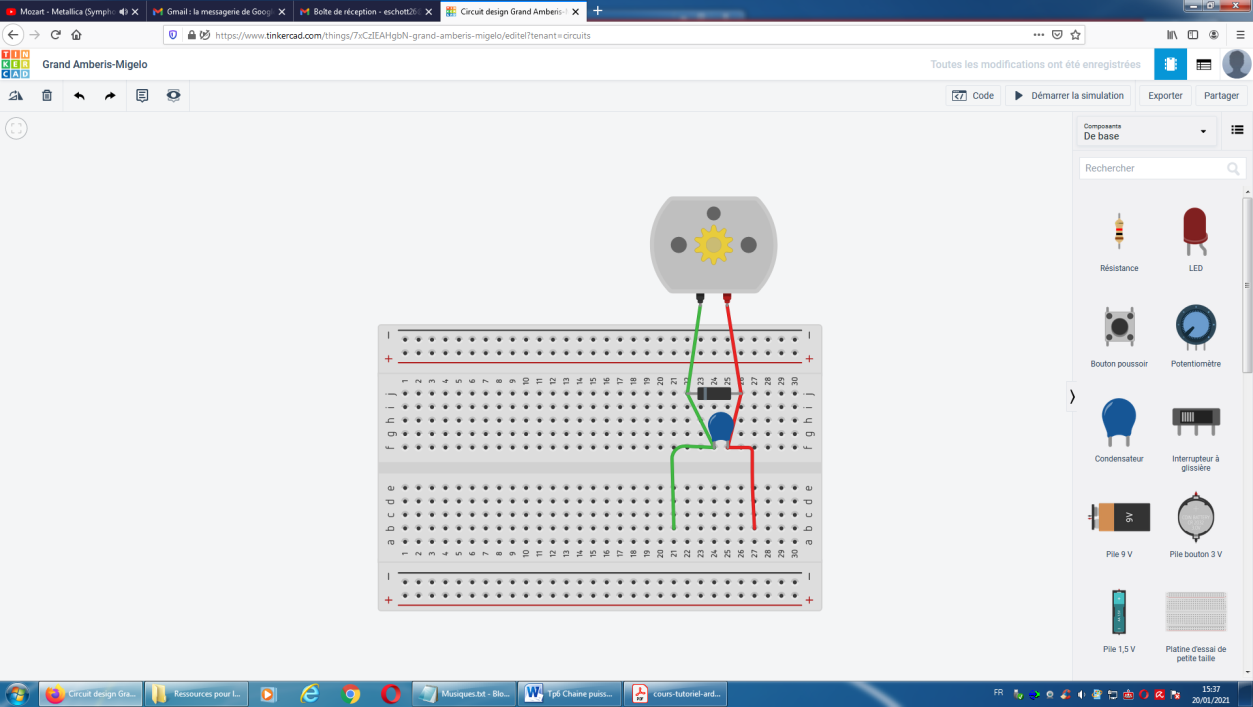
Les deux bornes vertes de la carte « Relay »

1. **Puis sur votre schéma rajoutez le bouton poussoir et le shield de la carte Arduino.**
2. **Voici des informations pour vous aider à écrire votre programme dans un câblage réel.**

|  |  |  |
| --- | --- | --- |
| Nom | Description graphique | Codes de programmation |
| 1 Entrée bouton poussoir « grove » à connecter sur le shield en D4  Cette carte d'extension Base Shield v2 est dotée de nombreux connecteurs Grove | Résultat de recherche d'images pour "button  arduino grove" | const int bouton1 = 4; //attribution de la broche 4 à la variable bouton1.  **void setup**() {  pinMode(bouton,INPUT); //initialisation de la broche 4 comme étant une entrée.  }  **void loop**() {  etat\_bouton=digitalRead(bouton);  if (digitalRead(bouton) == 1) // test de la condition, ici la variable boolèenne bouton est égale à un ou pas.  { Vous devez écrire à cette place ce que le programme doit faire quand vous avez appuyé sur le bouton }  else  { et ici c’est l’inverse bien sure } |
| 1 sortie relay « grove » à connecter sur le shield en D8 | Résultat de recherche d'images pour "relais arduino HLS8L-DC5V" | #define RELAY\_PIN 8  **void setup**() {  pinMode(RELAY\_PIN, OUTPUT); // initialize the digital pin as an output.  }  **void loop**() {  digitalWrite(RELAY\_PIN, HIGH); // Met en marche le relais, donc ferme le contact de puissance et le moteur fonctionne. } |

1. **Voici des informations pour vous aider à tracer le câblage virtuel dans un Tinkercad**

|  |  |  |
| --- | --- | --- |
| Eléments nécessaires pour réaliser le câblage virtuel dans la bibliothèque | Photo d’un composant réel. | Pour le connecter |
| Bouton poussoir avec deux paires de contacts de type N.O.  (N.O : Normalement ouvert au repos) | Résultat de recherche d'images pour "button  arduino grove" |  |
| Interrupteur coulissant à 2 positions  A gauche un contact de type N.C.  A droite un contact de type N.O.  Attention ne pas oublier de rajouter une résistance pull-down de 10 kOhms. |  |  |
| Relais unipolaire bidirectionnel  « LU-5-R » pour un fonctionnement « tout ou rien » de l’élément piloté. | Résultat de recherche d'images pour "relais arduino HLS8L-DC5V" |  |



Deux composants sont à rajouter à votre schéma dans Tinkercad (pour en prendre l’habitude) :

**La diode de roue libre** va protéger les éléments entre le moteur et la carte elle-même. Une fois installée. Elle va éliminer la surtension qui apparait dans le moteur (bobine) lors de son l’extinction, car celui-ci se transforme en génératrice de courant pendant un court instant. Il suffit alors de placer une diode en parallèle avec le moteur.

**Le condensateur** quant à lui va bloquer tous les parasites électriques créé involontairement par le moteur en évitant de les envoyer dans le circuit s’il n’est pas installé.

Explication rapide sur la nécessité d’une **résistance pull-up ou pull-down** :

Lorsqu'on utilise une entrée numérique, il est important de s'assurer que le potentiel de l'entrée « au repos » est bien celui auquel on s'attend. En effet, si on laisse l'entrée « libre », c'est-à-dire câblée à rien, le potentiel qu'elle prendra ne sera pas nécessairement 0 V. On parle alors de potentiel flottant, car l'électricité statique ambiante ou les perturbations électromagnétiques peuvent faire apparaître des valeurs très fluctuantes. On utilise alors une liaison protégée par une résistance qui va « tirer vers le haut » (5 V) ou « tirer vers le bas » (0 V) le potentiel au repos, comme une sorte d'élastique. C’est cette dernière qui sera utilisée. (Résistance de 10 kOhms).

1. **Ecrire votre programme ci-dessous et dans le logiciel Arduino et dans Tinkercad.**

const int ………………...

|  |
| --- |
|  |

**#define RELAY\_PIN 8**

**const int bouton1 = 2;**

**void setup() {**

**pinMode(bouton1,INPUT);**

**pinMode(RELAY\_PIN, OUTPUT);**

**}**

**void loop() {**

**//bouton1=digitalRead(bouton1);**

**if (digitalRead(bouton1)== 1)**

**{**

**digitalWrite(RELAY\_PIN, HIGH);**

**}**

**else**

**{**

**digitalWrite(RELAY\_PIN, LOW);**

**}**

**}**

1. **Modéliser votre câblage**

Si vous êtes en présentiel :

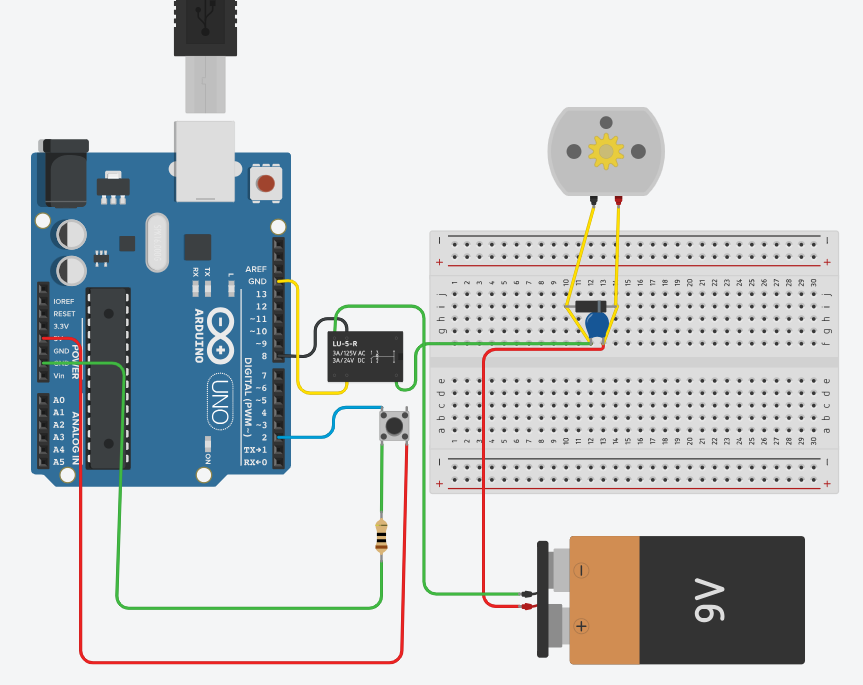
Après avoir fait le schéma.

Câbler votre montage en étant hors tension.

Obtenir l’accord de votre enseignant pour le mettre sous tension.

Si vous êtes en distanciel :

Dans Tinkercad, tracer votre schéma en utilisant les éléments de programme et les composants des paragraphes 6 & 7.



1. **Valider le, faire des copies d’écran, car le compte rendu de cette activité sera ramassée et notée.**

