



# A-CONTEXTE

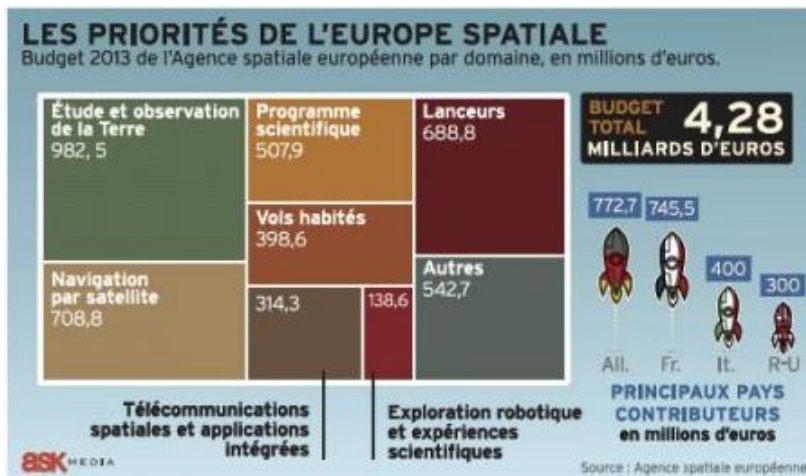
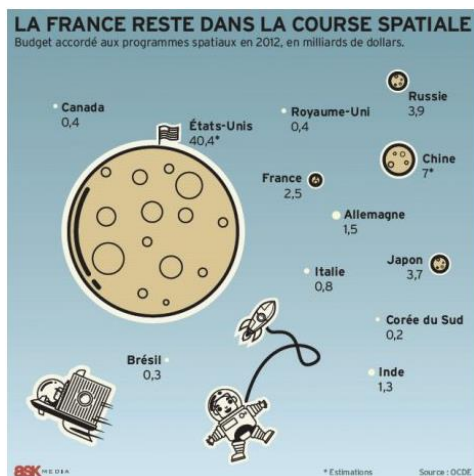
Depuis la nuit des temps l'homme a toujours voulu savoir ce qu'il y avait derrière la colline, l'océan ou sous sa surface, en haut des plus hauts sommets. Explorer l'espace n'est donc qu'une suite logique. Pour en arriver là, il a dû toujours repousser plus loin ses propres limites physiques ainsi que techniques. Grâce à cela les avancées réalisées ont apporté de nombreux bénéfices à nos sociétés, les défis relevés dans l'exploration spatiale nous permettent d'améliorer nos connaissances de l'univers dans lequel nous vivons, les technologies mises en œuvre ont un impact sociétal (création de nouvelles industries et donc d'emplois, amélioration du cadre de vie (Ex. : Positionnement grâce au GPS : cela évite de se perdre, de trouver son chemin, de porter secours, ....) et bien d'autres.

☑ **A.1 - Visionnez la vidéo suivante :** <https://youtu.be/0rRQU9dvkhc?t=11>  
en cas de problème de connexion la vidéo est sur le serveur.

☑ **A.2 – Quels sont les avantages que nous pouvons tirer de l'exploration spatiale ? prenons l'exemple de Galiléo.**



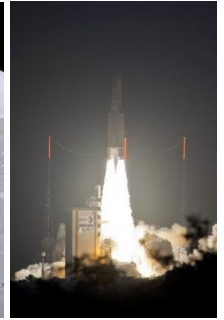
En 2013, l'agence spatiale européenne (ESA) disposait d'un budget de 4,28 milliards d'euros, dont 3,1 milliards provenant des États membres :



<https://www.la Tribune.fr/entreprises-finance/industrie/aeronautique-defense/20140120trib000810519/comment-la-conquete-spatiale-a-change-notre-vie.html>

☑ **A.3 – Quelle place occupe notre pays dans la course spatiale mondiale ?**

☑ **A.4 – Quel est la part du budget destiné à l'exploration robotique et expérience scientifique ?**



# B-EXPRESSION DU BESOIN

## Quel est le problème ?

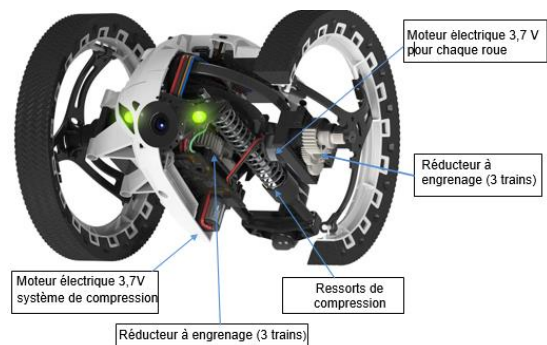
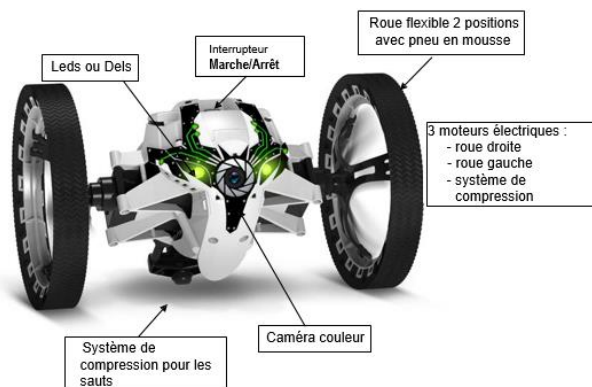
L'exploration à distance nécessite l'utilisation de robot pilotable depuis la terre. Ces robots sont dotés de capacités particulières pour franchir des obstacles, effectuer et analyser des prélèvements de roches ou d'alluvions, nous envoyer des images. Ils nous permettent d'aller là où nous ne pouvons pas aller pour le moment (Mission Rover en est un exemple, Curiosity envoyé sur la planète Mars le 6 août 2012)

## Objectif du projet : construire un robot civil

Fort de cette expérience dans l'exploration spatiale, le directeur de votre future entreprise souhaite se diversifier et créer deux nouvelles branches :

- Une, destinée au public, dite de loisirs avec un robot permettant de jouer en se déplaçant dans son environnement proche. Capable d'effectuer des sauts pour franchir des obstacles et de réaliser des images ou des films.
- Une autre, plus professionnelle dirigée vers :
  - les entreprises (exploration de canalisation souterraine pour la recherche de pannes, ...).
  - les corps de la sécurité civile permettant de porter secours à des personnes en danger (exploration de cavité ou de bâtiments effondrés, suite à tremblement de terre la recherche de personnes ensevelies ou pour explorer des sites en zone contaminée ou très dangereuse).
  - des corps à des fins militaires ou antiterroristes (exploration d'une zone de combat , en cas de prise d'otage-> la recherche de renseignements)

Jeune ingénieur sorti de l'école vous intégrez l'équipe R&D, vous êtes chargé de valider le premier prototype sorti des laboratoires.



Le dossier présentant le produit est à votre disposition et utilise des outils différents de ceux que vous avez appris à l'école. Mais ne pose pas de problème dans leur compréhension.

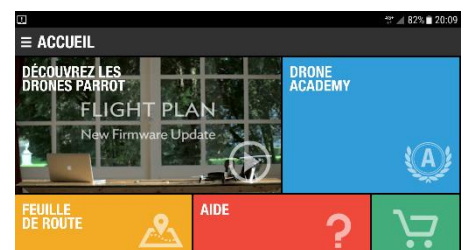
### Mise en situation

Le robot jumping Parrot Jumping Sumo est un robot à forte personnalité qui roule, fonce, zigzague, virevolte, prend des virages à 90°, fait des bonds et peut même frapper des objets. Il apporte de nouvelles sensations dans le domaine de la robotique de loisirs.

Avant de mettre en œuvre le robot, consulter les pages 2 à 6 du dossier technique du robot dans le site intranet associé à notre étude.

### Pour mettre en œuvre le robot :

- ouvrir le dossier « Guide utilisateur ».
- connecter votre smartphone au « wifi » du jumping Sumo.
- lancer l'application « Freeflight jumping ».
- piloter le robot.
- faire exécuter au jumping sumo les principales acrobaties.

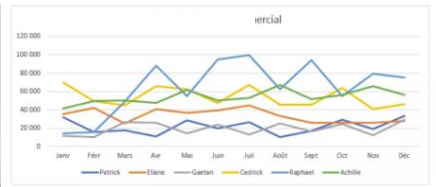
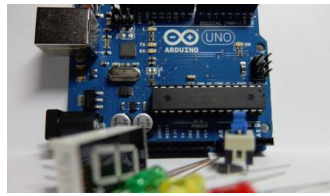
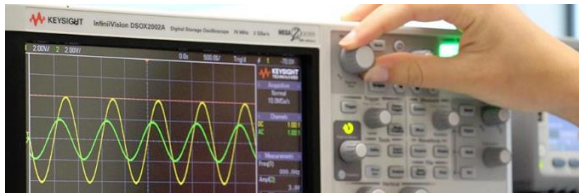


- avec l'option « feuille de route » programmez un parcours avec 4 acrobaties.

- ☒ **B.1 - A partir de l'analyse de l'extrait du cahier des charges (dossier technique du robot jumping) identifiez, relevez les niveaux des critères de la fonction principale FP1 et complétez le tableau ci-dessous :**

	Critères d'appréciation	Niveau	Flexibilité
<b>FP1</b>  Permettre aux joueurs de déplacer, de visualiser, de faire sauter, de filmer, ...	Longueur des sauts		
	Hauteur des sauts		
	Vitesse et accélération		





# C-EXPERIMENTATION

Quatre points sont donc à vérifier, vous allez donc vous répartir le travail. Chacun prendra à sa charge un des points du cahier des charges : la **hauteur** du saut, la **longueur** du saut, la **vitesse** de déplacement du robot et les **paramètres** du moteur.

Mener une expérimentation c'est d'abord bien comprendre ce que vous devez vérifier et mesurer, le protocole expérimental. Ensuite bien noter les résultats de l'expérience pour les analyser.

Pour vous aider dans cette tâche :

## ☑ C.1 - Répondez aux questions suivantes :

(Attention les réponses sont propres au point particulier que vous devez vérifier, pour ce qui est du protocole expérimental chacun de vous ira consulter les pages spécifiques à son expérimentation)

- Objectif de l'expérience :
- Paramètre à observer (à mesurer) :
- Valeur souhaitée ou attendue :
- Matériels à disposition :

## Premier élève ..... : la vitesse de déplacement

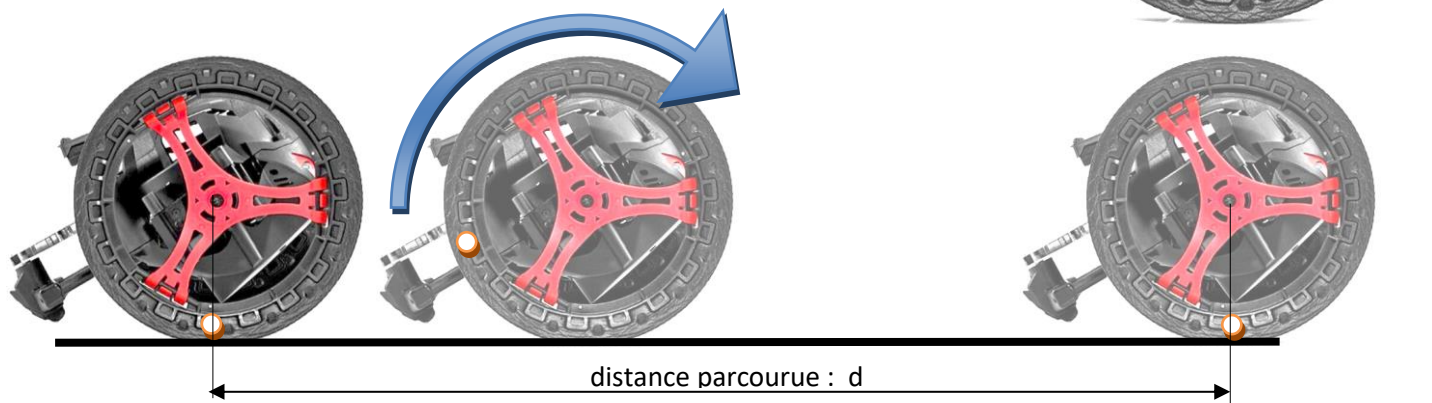
### ☑ C.2 - Mesurez la fréquence de rotation à vide de la roue.

#### • Protocole expérimental 1 :

- Collez une pastille réfléchissante sur une des roues du robot si elle n'est pas déjà présente.
- Surélevez le robot pour les roues ne soient plus en contact avec le sol.
- Pilotez le déplacement longitudinal à vitesse maximale.
- À l'aide du tachymètre fourni, mesurez la fréquence de rotation (en tr/min) de la roue du robot.

N = tr/min

- ☑ Comment détermineriez-vous la distance parcourue pour 1 tour de roue, de quelle dimension auriez-vous besoin ?



- ☑ Écrivez la relation littérale permettant de calculer la distance parcourue = .....

- ☑ Mesurez le diamètre (D) de la roue précisément :

Ø D = mm

- ☑ Calculez maintenant la distance parcourue par tour de roue d

d = mm

- ☑ Établissez puis calculez la vitesse de déplacement du robot à partir d'un calcul prenant en compte la distance parcourue par tour d et la fréquence de rotation N (en tr/min), la vitesse de déplacement du robot sera exprimée en m/min, en m/s puis en km/h.



- Protocole expérimental 2 : vérification de la vitesse atteinte par le robot

Pour vérifier ce critère de façon précise vous allez suivre un protocole basé sur l'analyse d'une image vidéo montrant le robot en déplacement horizontal. Cette analyse nécessite l'emploi du logiciel « Tracker ». Afin de gagner du temps cette vidéo a été déjà réalisée, nom du fichier : « **vitesse-3.avi** »

**Étape 1 : Pointage vidéo**

- lancez le logiciel « Tracker » et chargez la vidéo réalisée.
- réalisez le pointage vidéo du déplacement du robot (suivre le tutoriel pas à pas fourni).

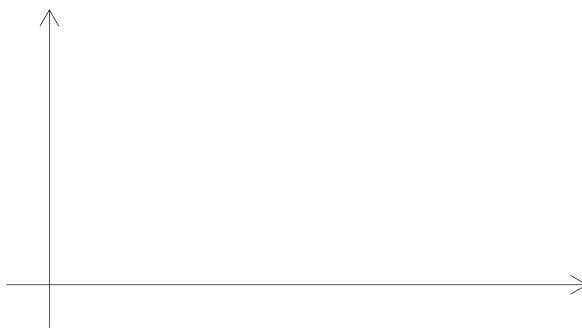
**Étape 2 :**

- tracez sur la photo ci-dessus la trajectoire suivie par le robot

**Étape 3 : Analyse des données**

- affichez le graphe représentant la vitesse du robot.

- ☒ **C.3 - Tracez ci-dessous l'allure de la courbe puis relevez la valeur de la vitesse max et le temps mis pour l'atteindre. Concluez par rapport à la validité du cahier des charges.**



$V_{\max} = \dots\dots\dots$

$t_{\max} = \dots\dots\dots$

**Conclusion :**

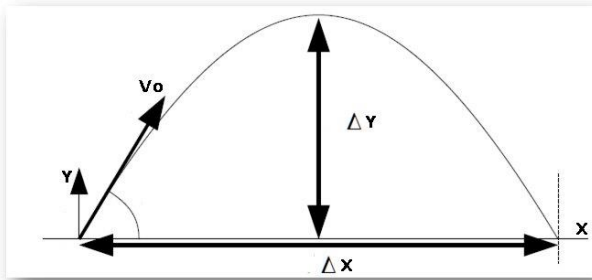
- ☒ **C.4 – Analyse :** La vitesse de déplacement du robot dépend elle du diamètre des roues ? si oui sur quel paramètre dimensionnel peut-on agir pour augmenter ou diminuer celle-ci sans toucher à la fréquence de rotation du moteur ?

# ETUDE THEORIQUE : Partie commune aux 2 autres études.

## Deuxième élève ..... : la hauteur

## Troisième élève ..... : la longueur

Comme vous avez pu l'observer le robot jumping Parrot Jumping Sumo décrit **une trajectoire parabolique** lors de ses sauts. L'étude consistera à définir la vitesse de propulsion **Vo** à donner à celui-ci ainsi que l'angle **α** afin d'obtenir le saut requis.



Une étude dynamique du jumping Parrot Jumping Sumo durant sa phase de saut nous a permis de définir l'équation de la trajectoire suivante :

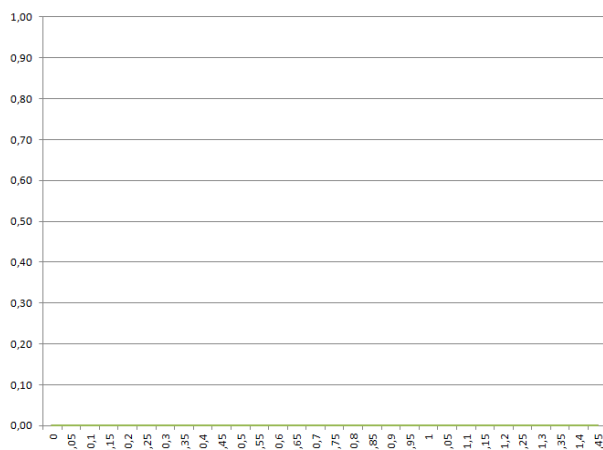
$$H(x) = Y(x) = \frac{-g}{2.Vo.Cos^2\alpha}.X^2 + X.tan\alpha$$

**H(x), Y(x)** : altitude atteinte en m

**g** : accélération de la pesanteur 9,81 m/s<sup>2</sup>    **Vo** : vitesse de propulsion du robot en m/s  
**α** : angle de la direction à donner au robot au moment du départ en °

Même si la fonction paraît compliquée n'ayez pas de souci, nous allons utiliser un tableur pour faire cette étude :

- ☒ **Ouvrez** le fichier : « **parabole.xlvs** »
- ☒ **Renseignez** les cases colorées, le tableur effectuera le reste.  
**Nota** : l'intervalle concernant la vitesse de propulsion est Vo [0 ; 4,5 m/s]  
 Pour une valeur **Vo** donnée, faites varier l'angle de propulsion **α** .....
- ☒ **Tracez** en couleur l'allure des trajectoires en fonction des 3 angles choisis.



« **parabole-vitesse.xlvs** »

- ☒ **Renseignez** les cases colorées, le tableur effectuera le reste.  
**Nota** : l'intervalle concernant la vitesse de propulsion est **Vo** [0 ; 4,5 m/s]  
 Pour une valeur **α** donnée, faites varier la vitesse de propulsion **Vo** .....
- ☒ **Tracez** en couleur l'allure des vitesses en fonction des 3 valeurs choisies.
- ☒ **C.4** – Analysez les résultats. Quel constat faites-vous sur l'influence de la vitesse sur la trajectoire du robot ?

Ouvrez le fichier :

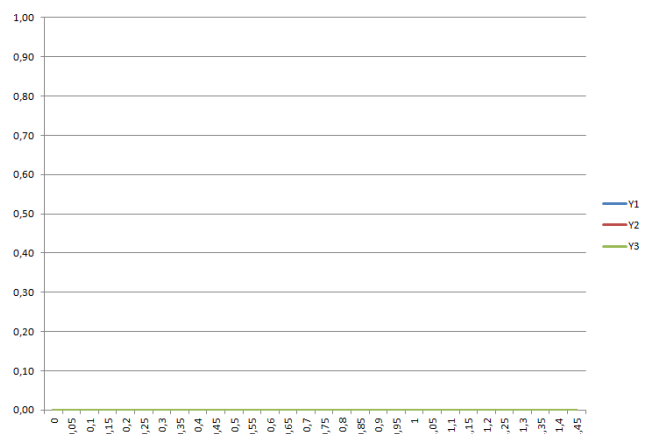
### Données

g	Vo	α
(m/s <sup>2</sup> )	(m/s)	(°)
9,81		

- ☒ **C.3** – Analysez les résultats. Quel constat faites-vous sur l'influence de l'angle d'inclinaison sur la trajectoire du robot?

### Données

g	Vo	α
(m/s <sup>2</sup> )	(m/s)	(°)
9,81		



## Deuxième élève ..... : la hauteur

- Protocole expérimental 1 : vérification de la hauteur d'un saut

Pour vérifier ce critère de façon précise vous allez suivre un protocole basé sur l'analyse d'une image vidéo montrant le robot exécuter un saut. Cette analyse nécessite l'emploi du logiciel « Tracker ». Afin de gagner du temps cette vidéo a été déjà réalisée, nom du fichier : « **Saut\_en\_hauteur\_1.avi** »

### Étape 1 : Pointage vidéo

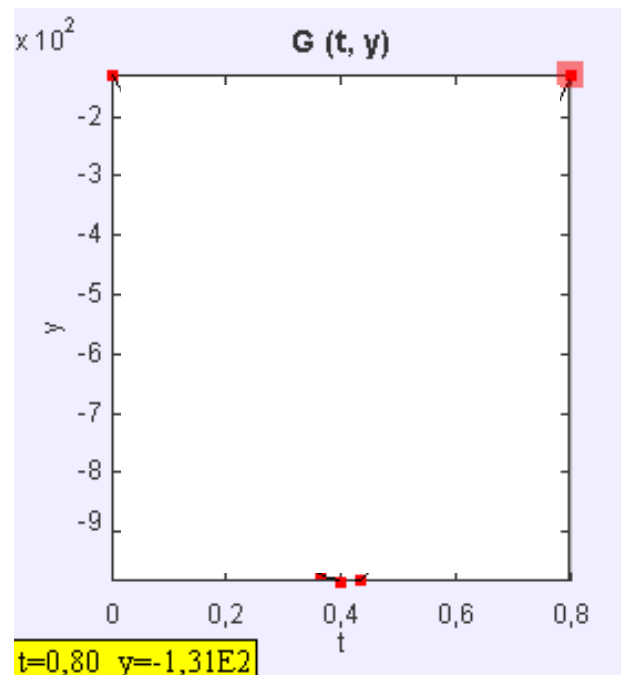
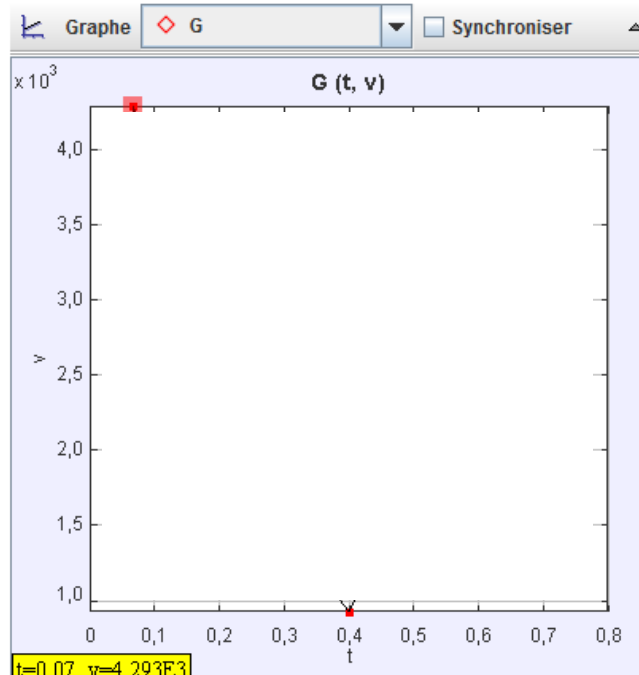
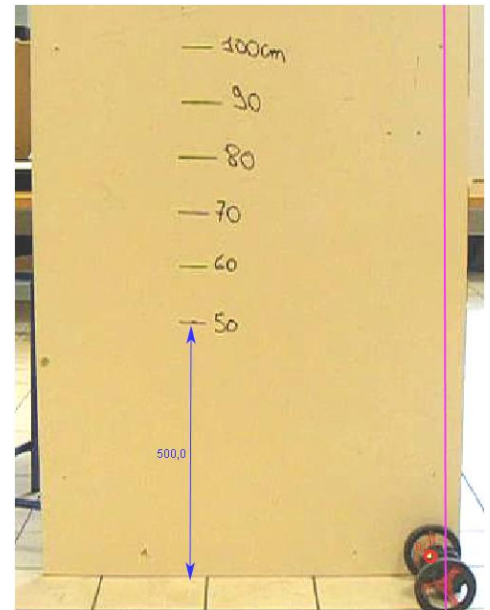
- lancez le logiciel « Tracker » et chargez la vidéo réalisée.
- réalisez le pointage vidéo du saut en hauteur du robot (suivre le tutoriel pas à pas fourni).

### Étape 2 :

- tracez sur la photo ci-dessus la trajectoire suivie par le robot

### Étape 3 : Analyse des données

- affichez le graphe représentant la vitesse du robot.
- **Mesurez** à l'aide du rapporteur, l'angle de propulsion et reportez-le sur la figure. (Voir tuto)
- **Afficher** les graphes de la position **Y** et de la vitesse **V** en fonction du temps.
- **Tracez** l'allure de ces courbes ci-dessous.
- **Repérez** les valeurs maximales atteintes.



☒ **C.5** - Concluez par rapport à la validité du cahier des charges.

☒ **C.6** - Connaissant maintenant les résultats expérimentaux, veuillez entrer les paramètres  $V$  et  $\alpha$  dans votre feuille de calcul. Quelle(s) remarque(s) pourriez-vous faire entre les valeurs trouvées sur la vidéo et celles obtenues à l'aide du modèle mathématique, une erreur de l'ordre de 10% est déjà très acceptable.

## Troisième élève ..... : la longueur

- Protocole expérimental 1 : vérification de la longueur d'un saut

Pour vérifier ce critère de façon précise vous allez suivre un protocole basé sur l'analyse d'une image vidéo montrant le robot exécuter un saut. Cette analyse nécessite l'emploi du logiciel « Tracker ». Afin de gagner du temps cette vidéo a été déjà réalisée, nom du fichier : « **saut\_longueur.trk** »

### Étape 1 : Pointage vidéo

- lancez le logiciel « Tracker » et chargez la vidéo réalisée.
- réalisez le pointage vidéo du saut en longueur du robot (suivre le tutoriel pas à pas fourni).

Nota : La capture automatique semble poser un problème (trop rapide peut être), vous procéderez ici en faisant le pointage manuellement. Pour cela, activez « Marquer par défaut » et pointez chaque position.

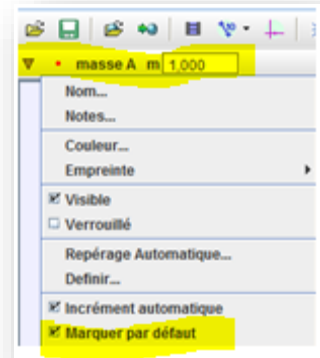
### Étape 2 :

- tracez sur la photo ci-dessus la trajectoire suivie par le robot

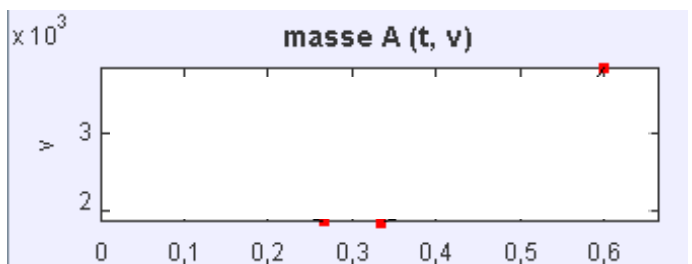
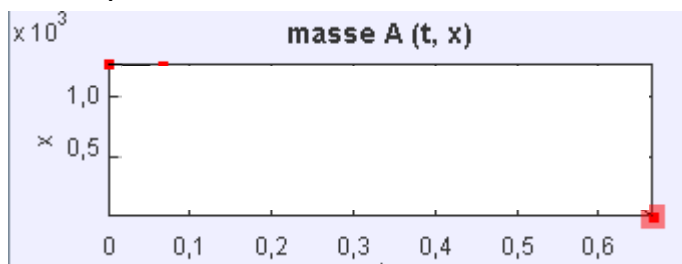
### Étape 3 : Analyse des données

- affichez le graphe représentant la vitesse du robot.

Longueur planche : 1 m



- **Mesurez** à l'aide du rapporteur l'angle de propulsion et reportez-le sur la figure. (Voir tuto)
- **Afficher** les graphes de la position **X** et de la vitesse **V** en fonction du temps.
- **Tracez** l'allure de ces courbes ci-dessous.
- **Repérez** les valeurs maximales atteintes.



- ☑ **C.7** - Concluez par rapport à la validité du cahier des charges.

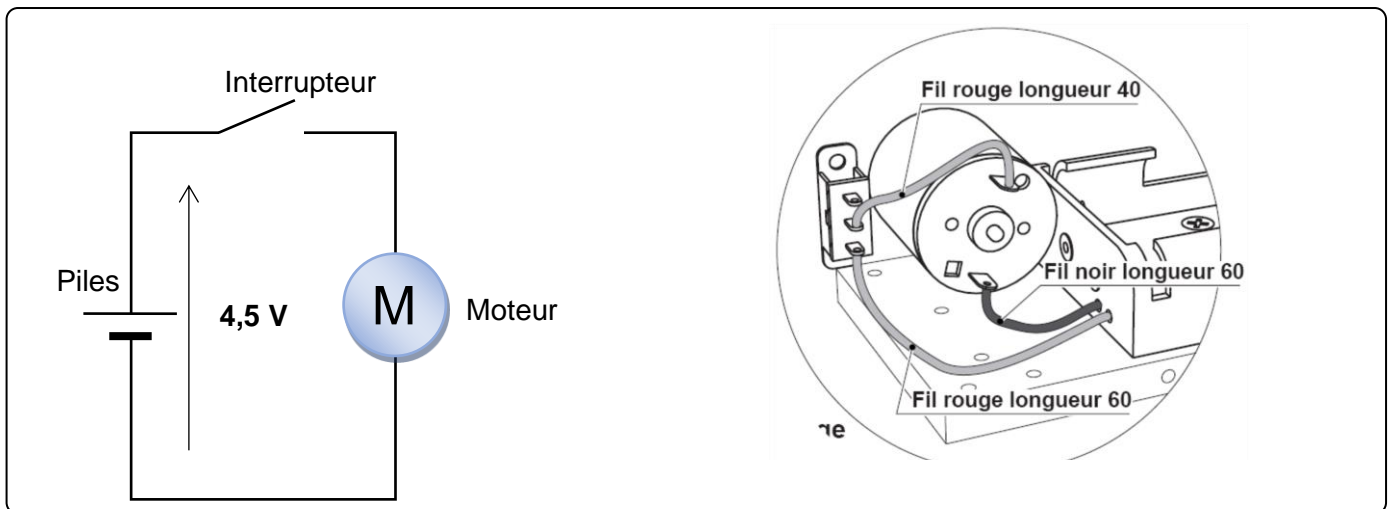


## Quatrième élève ..... : le moteur

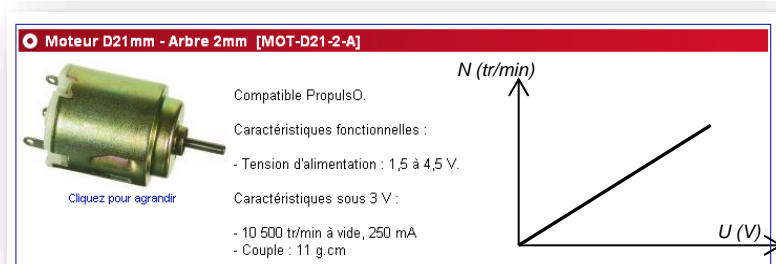
Le moteur .... Élément indispensable dès que l'on souhaite déplacer le robot par exemple. Quels sont les paramètres importants qui régissent son fonctionnement ?

### ☑ C.8- Commençons par le Schéma électrique

Le moteur est schématisé par un cercle avec en son centre la lettre « M ». Attention on ne **peut pas câbler directement le moteur** sur la carte **Arduino**. Sous peine de la détruire nous y reviendrons plus tard.



Le fournisseur du moteur électrique (Mot-D21-2-A) nous fournit les caractéristiques suivantes, comment **interprétez-vous** cette courbe, que doit-on comprendre ?

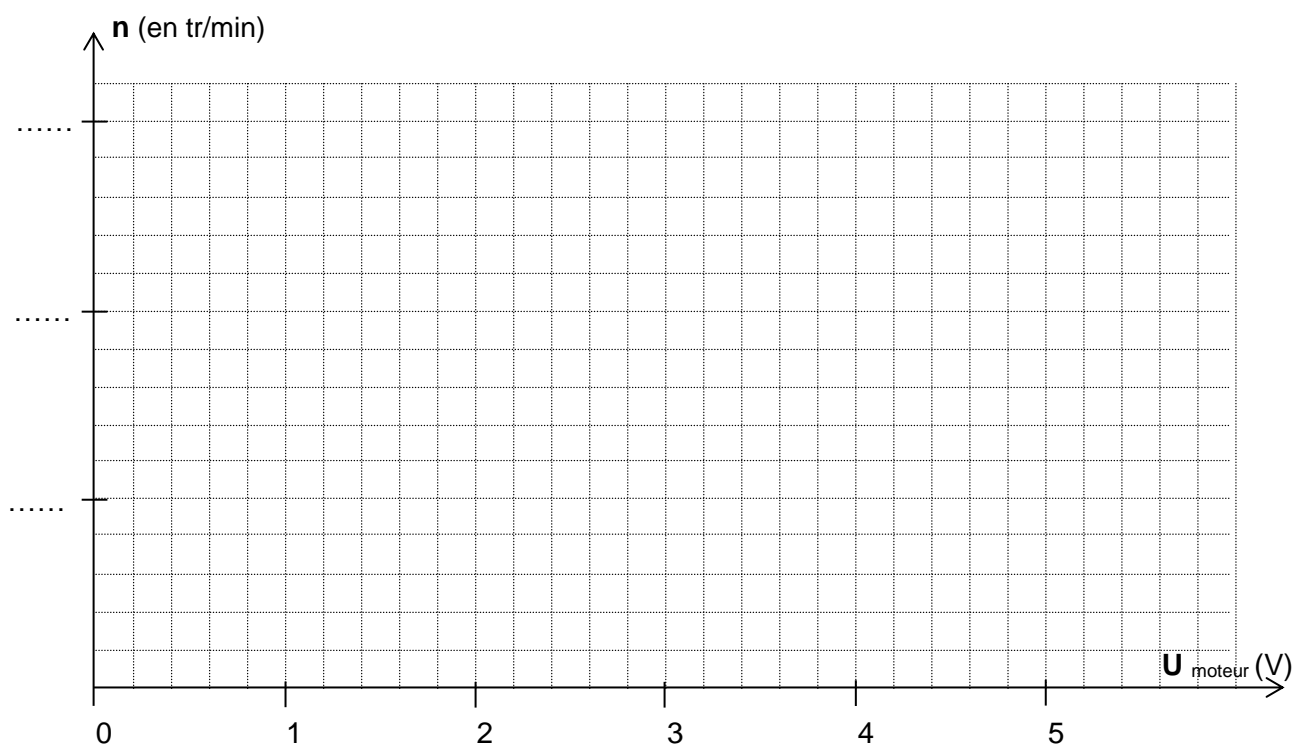


### ☑ C.9- Sur la photo ci-dessous tracez le circuit électrique vous permettant d'alimenter le moteur puis faites vérifier votre câblage par le professeur.

- Protocole expérimental 1 : vérification des caractéristiques du moteur.
  - **Réalisez** le câblage avec le matériel mis à votre disposition puis réalisez l'expérience ( faites vérifier avant !!).
  - **Coupez** l'alimentation puis **inversez** les polarités d'alimentation du moteur. Que constatez-vous
  - **Faites varier** la tension d'alimentation du moteur de 0 à 4,5 V ou plus suivant votre moteur, votre professeur vous l'indiquera.
  - **Mesurer** à l'aide d'un **tachymètre** la fréquence de rotation du moteur **n** (en tr/min) et complétez le tableau.
  - **Tracez** la courbe puis **concluez**.



Mesures à vide								
<b>U (volts)</b>	<b>0</b>	....	....	....	....	....	....	....
<b>n (en tr/min)</b>	<b>0</b>							



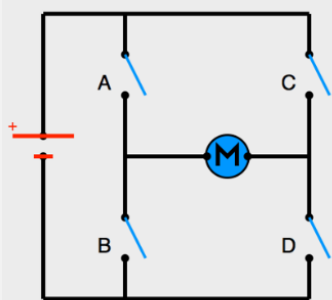
☒ **C.10-** Vos résultats sont-ils conformes à ceux du constructeur.

☒ **C.11-** Expliquez en quelques mots le fonctionnement d'un moteur.

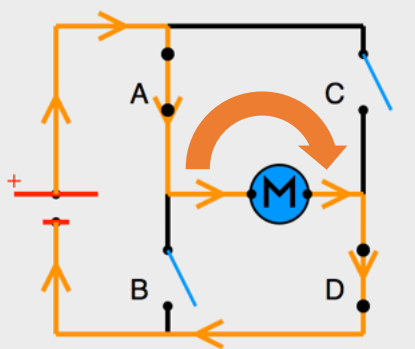
# Principe de base d'une commande moteur 2 sens de rotation.

A la base un robot est constitué de plusieurs moteurs, pour lui permettre de se déplacer il est nécessaire de pouvoir agir sur le sens de rotation pour lui permettre d'avancer, de reculer, de tourner à droite ou à gauche et pour finir de gérer sa vitesse. Ce miracle est possible grâce à un **pont en H**.

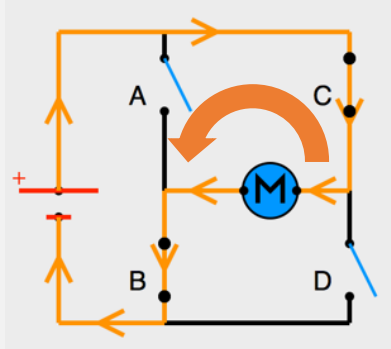
Explication :



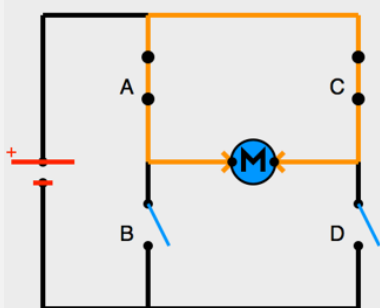
Voilà ci-contre le schéma de principe d'un **pont en H**. Il tient son nom de la forme en H du circuit autour du moteur. Pour faire simple celui-ci est constitué de 4 interrupteurs A, B, C et D la combinaison de fermeture de ces interrupteurs va entraîner un fonctionnement différent du moteur. Ici le **moteur est libre**.



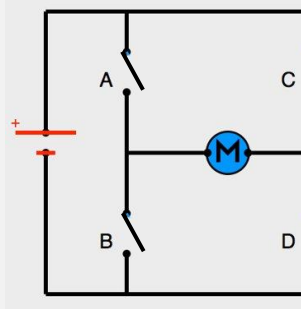
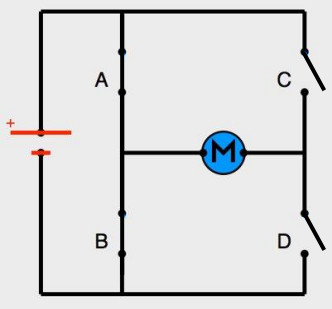
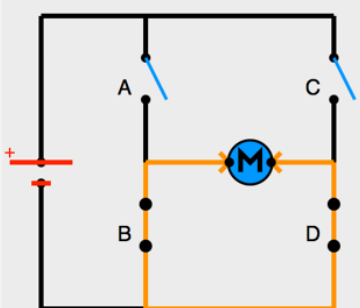
**Sens 1** : Les interrupteurs fonctionnent deux par deux. Le A est ici associé au D et sont fermés. Le courant entre par la patte gauche du moteur et sort par sa droite. Le **moteur tourne dans le sens 1**.



**Sens 2** : Les interrupteurs fonctionnent deux par deux. Le B est associé au C et sont fermés. Le courant entre par la patte droite du moteur et sort par sa gauche. Le **moteur tourne donc dans le sens inverse** !



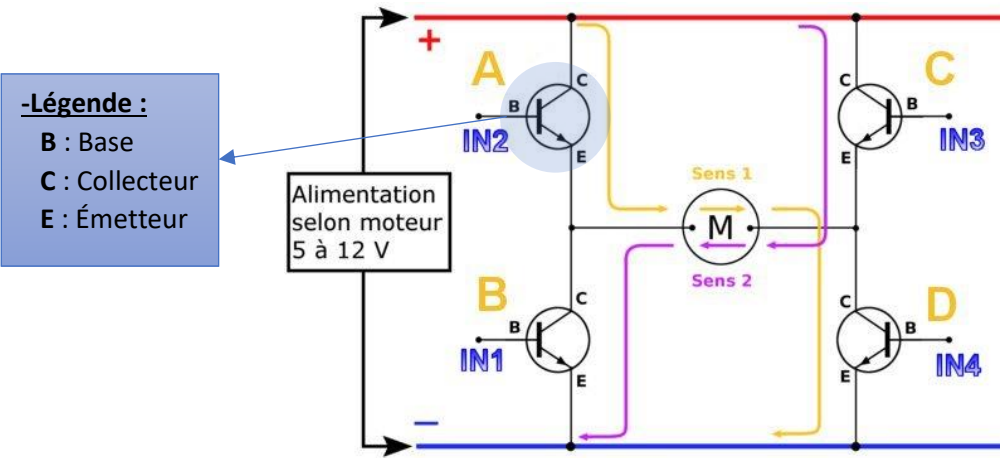
On peut aussi associer le A au C et le B au D. Lorsqu'un moteur est en roue libre (c'est-à-dire qu'il tourne à cause de sa force d'inertie mais pas à cause du courant), il génère un courant. Ce courant peut être utilisé dans le pont en H comme frein électromagnétique. Le **moteur s'envoie son propre courant à l'envers** ! Ça permet de **contrôler l'arrêt** du moteur plutôt que de le **laisser en roue libre**.



Si vous fermez tous les interrupteurs ou A et B ou C et D, vous réalisez un court-circuit. Dans le cas d'une pile, elle se décharge extrêmement vite. Dans le cas de **composants électroniques** ou d'une **carte Arduino**, ils **grillent**.

**Du coup, vous l'aurez compris, évitez de fermer tous les interrupteurs ou les combinaisons précisées ci-dessus !**

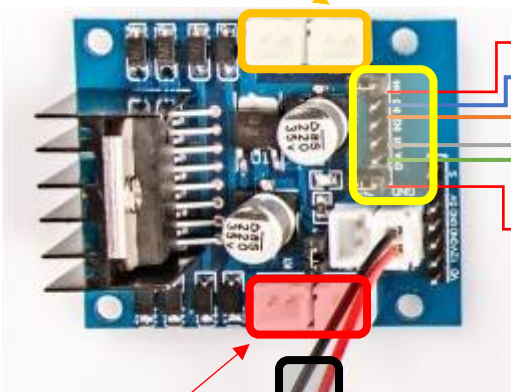
Vous comprendrez aisément qu'il est un peu fastidieux de commander un moteur, et par extension, un robot, avec des interrupteurs ! Heureusement l'électronique est là. La figure ci-dessous montre un montage classique en électronique, appelé pont en H (H bridge) ou pont double (dual bridge). Il est constitué de 4 transistors NPN appelés A, B, C et D (comme nos interrupteurs du paragraphe précédent). Pour faire tourner le moteur dans le **sens 1**, le courant doit suivre le cheminement représenté en **ORANGE**. Les transistors **A** et **D** doivent être **passants** alors que les transistors **B** et **C** doivent être **bloqués**. C'est l'inverse pour faire tourner le moteur dans le **sens 2**, le courant doit suivre le cheminement représenté en **VIOLET** ; les transistors **B** et **C** doivent être **passants** et les transistors **A** et **D** doivent être **bloqués**.



### Carte de contrôle des moteurs L298 (Double pont en H)

Deux modèles, la carte bleue permet de contrôler 4 moteurs, le modèle orange seulement 2.

Câblage Moteur(s) côté gauche



ENB : vitesse des moteurs côté gauche  
IN4 : Sens côté Gauche  
IN3 : Sens côté Gauche

D6 ~  
D8  
D7

IN2 : Sens côté Droit  
IN1 : Sens côté Droit  
ENA : vitesse des moteurs droit

D4  
D2  
D5 ~

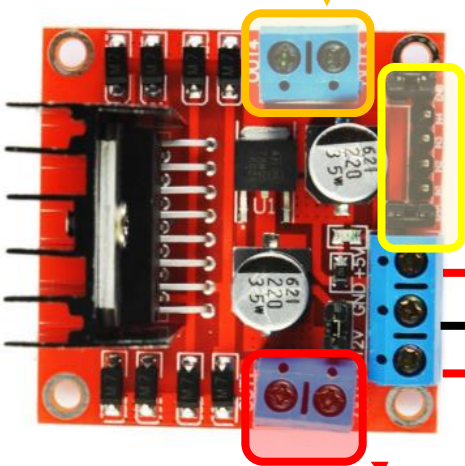
Câblage Moteur(s) côté droit

Alimentation des moteurs et de la carte Arduino par une batterie de 7,8V

	IN1 (D2)	IN2 (D4)	IN3 (D7)	IN4 (D8)
Avant (Sens 2)	HIGH	LOW	HIGH	LOW
Arrière (Sens 1)	LOW	HIGH	LOW	HIGH
Droite	LOW	HIGH	HIGH	LOW
Gauche	HIGH	LOW	LOW	HIGH

Contrôle sens des moteurs

Câblage Moteur(s) côté gauche



Identique à celui-ci-dessus

Alimentation des moteurs et de la carte Arduino par une batterie

Câblage Moteur(s) côté droit



## Partie 1 - Pilotage de 2 moteurs.

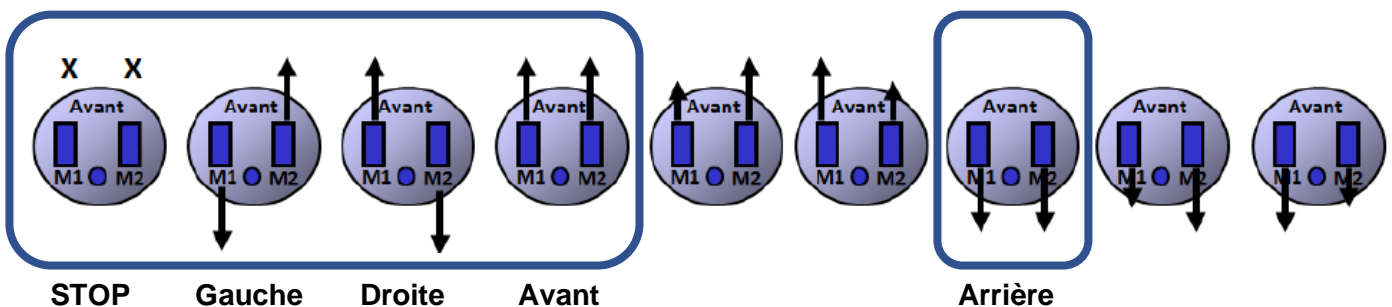
Par équipe de 2 !!



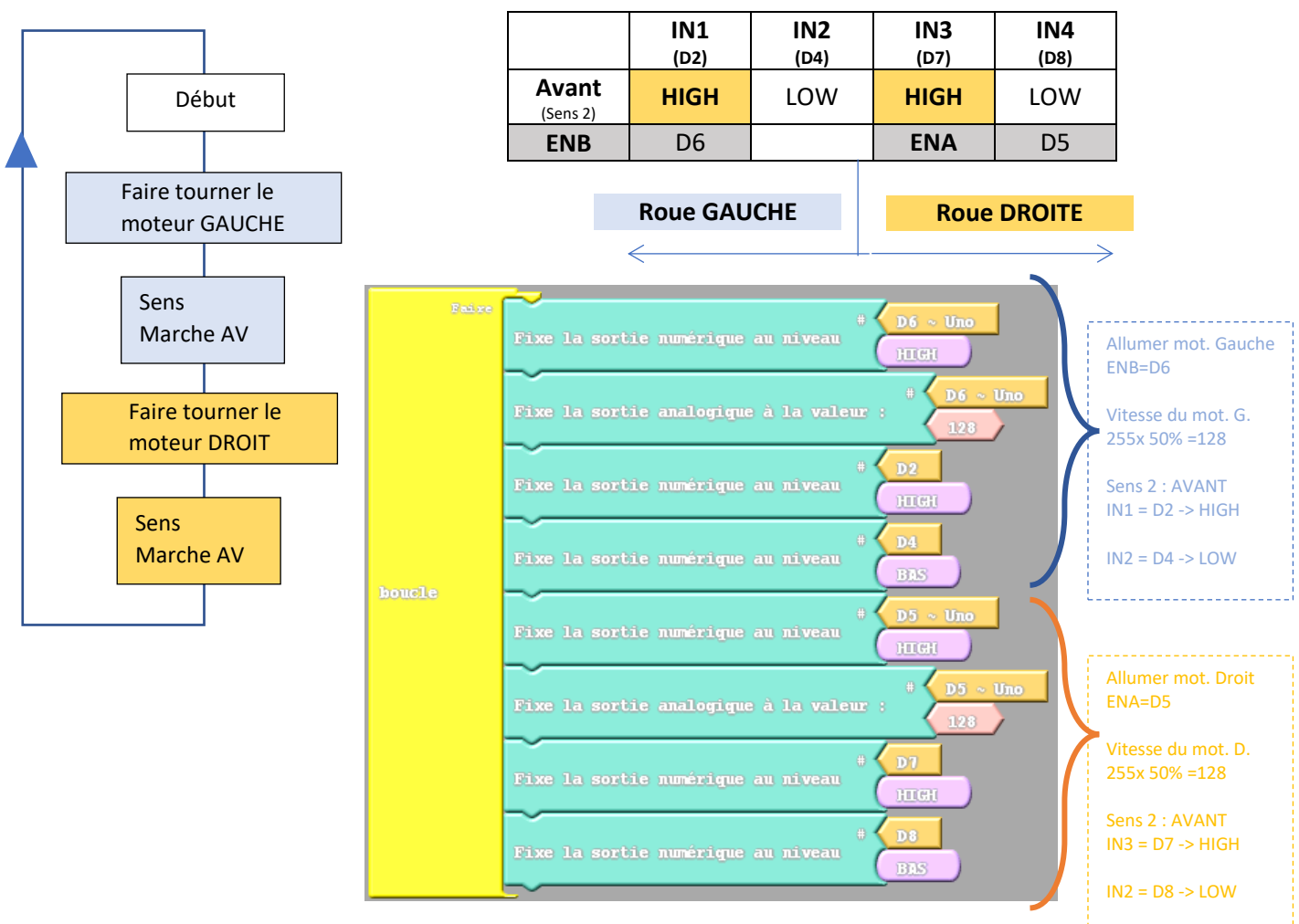
**IMPORTANT :** Ne jamais câbler un moteur sur la carte ARDUINO directement sous peine de la détériorer.

- ☑ **C.10** - Nous allons donc **réaliser le câblage du moteur** sur la carte Arduino. Reportez-vous au schéma précédent « carte de contrôle des moteurs L298 »

Pour diriger le robot, on peut définir 8 directions qui donneront de la précision dans les déplacements du robot car avec une base mécanique à 2 roues motrices, le mouvement est contrôlé par le sens de rotation et la vitesse de chaque moteur. Le pourcentage de vitesse (0 à 100%) peut être variable selon la technique du Pulse Width Modulation (PWM). Les rotations et changement de direction peuvent être très rapides ou au contraire plus progressifs. Ci-après les 8 solutions envisageables, mais nous nous limiterons aux 5 repérées :



- ☑ **C.11** - Nous allons donc **réaliser le programme** pour réaliser une marche AVANT.



## Programme

```
void setup()
{
  pinMode( 6 , OUTPUT);
  pinMode( 2 , OUTPUT);
  pinMode( 4 , OUTPUT);
  pinMode( 5 , OUTPUT);
  pinMode( 7 , OUTPUT);
  pinMode( 8 , OUTPUT);
}
```

```
void loop()
{
  digitalWrite(6 , HIGH);
  analogWrite(6 , 128);
  digitalWrite(2 , HIGH);
  digitalWrite(4 , LOW);
  digitalWrite(5 , HIGH);
  analogWrite(5 , 128);
  digitalWrite(7 , HIGH);
  digitalWrite(8 , LOW);
}
```



**Remarque importante** : ce n'est pas la vitesse des roues que la carte de contrôle L298N pilote, mais la vitesse des moteurs.

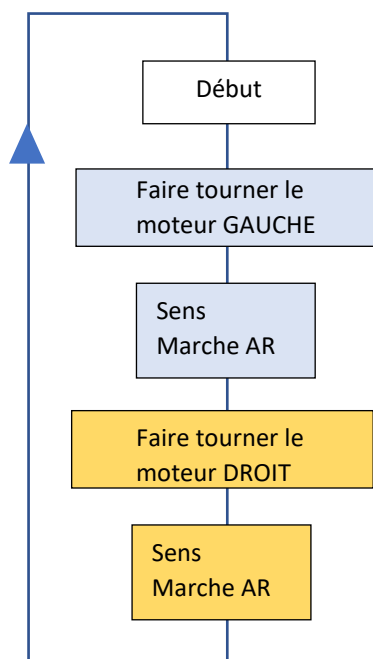
### Analyse :

il est nécessaire d'utiliser 4 blocs pour piloter un **SEUL** moteur :

- Le 1<sup>er</sup> permet de mettre le moteur en marche
- Le 2<sup>ème</sup> gère la vitesse de rotation du moteur.
- Le 3<sup>ème</sup> et le 4<sup>ème</sup> définissent le sens de rotation du robot en pilotant les transistors du pont en H.

D'autre part la gestion des numéros de broches est assez délicat pour éviter les courts-circuits.

☑ **C.12** - Vous allez donc **réaliser le programme** pour réaliser une marche **ARRIERE**.



### Travail à réaliser

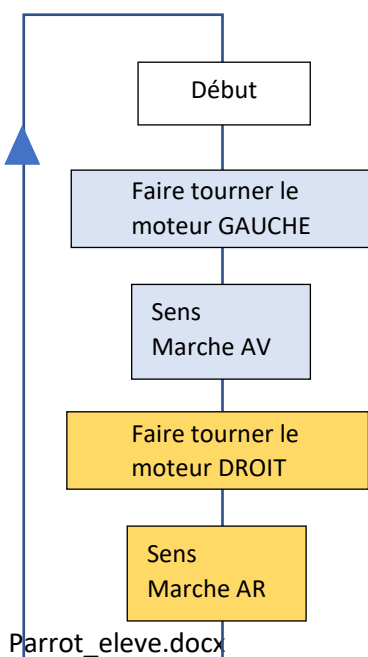
- 1- **Complétez** le tableau ci-dessous.
- 2- **Clonez** le programme précédent puis **stockez** le quelque part sur la page nous en aurons
- 3- **Apportez** les modifications nécessaires pour satisfaire le contrat.
- 4- **Enregistrez** votre travail sous un nouveau nom.

	IN1 (D2)	IN2 (D4)	IN3 (D7)	IN4 (D8)
<b>ARRIERE</b> (Sens 1)				
<b>ENB</b>	D6		<b>ENA</b>	D5

Roue GAUCHE

Roue DROITE

☑ **C.13** - Vous allez donc **réaliser le programme** pour réaliser un  $\frac{1}{4}$  de tour(s) à **DROITE**.



### Travail à réaliser

- 5- **Complétez** le tableau ci-dessous.
- 6- **Clonez** un sous-programme.
- 7- **Apportez** les modifications nécessaires pour satisfaire le contrat.
- 8- **Enregistrez** votre travail sous un nouveau nom.

	IN1 (D2)	IN2 (D4)	IN3 (D7)	IN4 (D8)
<b>DROITE</b>				
<b>ENB</b>	D6		<b>ENA</b>	D5

Roue GAUCHE

Roue DROITE

- ☑ **C.14** - Vous allez donc **réaliser le programme** pour **stopper** le robot.

### Travail à réaliser



- 9- **Complétez** le tableau ci-dessous.
- 10- **Clonez** un sous-programme puis **Apportez** les modifications nécessaires pour satisfaire le contrat.
- 11- **Enregistrez** votre travail.

	IN1 (D2)	IN2 (D4)	IN3 (D7)	IN4 (D8)
<b>ARRÊT</b>	LOW	LOW	LOW	LOW
<b>ENB</b>	D6		ENA	D5

Roue GAUCHE

Roue DROITE

- ☑ **C.15** - Vous allez donc **réaliser le programme** pour réaliser un  $\frac{1}{4}$  de tour à **GAUCHE**.

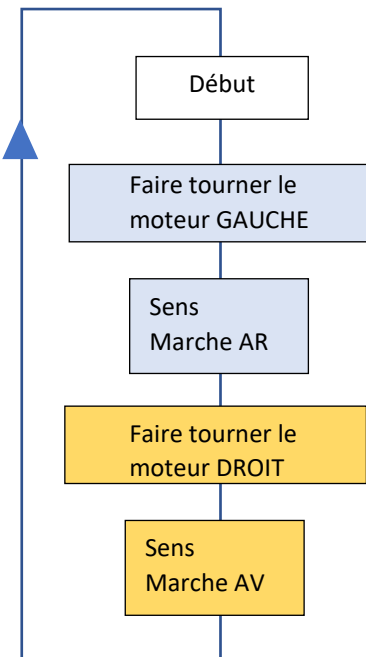
### Travail à réaliser

- 12- **Complétez** le tableau ci-dessous.
- 13- **Clonez** un sous-programme puis **Apportez** les modifications nécessaires pour satisfaire le contrat.
- 14- **Enregistrez** votre travail.

	IN1 (D2)	IN2 (D4)	IN3 (D7)	IN4 (D8)
<b>GAUCHE</b>				
<b>ENB</b>	D6		ENA	D5

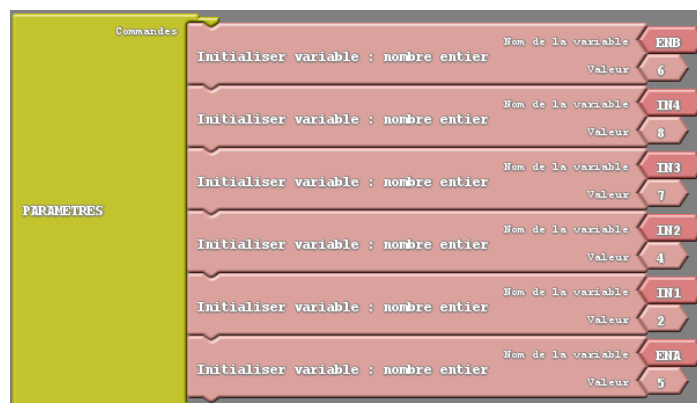
Roue GAUCHE

Roue DROITE

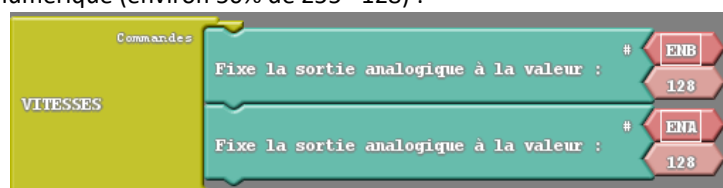


Pour rendre moins contraignante la gestion de ce programme on pourrait y apporter d'autres petites modifications :

- Créer un sous-programme « **Paramètres** » où l'on pourrait créer des variables **IN1**, **IN2**, **IN3**, **IN4**, **ENA** et **ENB** pour l'affectation des broches, l'avantage si l'on était amené à changer les connexions sur la carte Arduino pour diverses raisons (matériels supplémentaires par exemple), la mise à jour du nouveau paramètre se ferait rapidement sans avoir à chercher à l'intérieur des différentes lignes des sous-programmes.



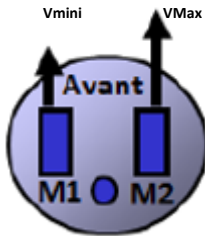
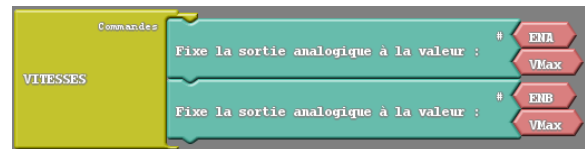
- Créer un sous-programme « **Vitesse** » où l'on pourrait donner aux variables « **ENB** » et « **ENA** » les vitesses des moteurs sous forme numérique (environ 50% de 255 = 128) :



Seul souci les moteurs n'ont qu'une plage de vitesses ... Pour y remédier on pourrait créer des sous-programmes vitesses

Vitesse 50 (%), Vitesse 75 (%), .....

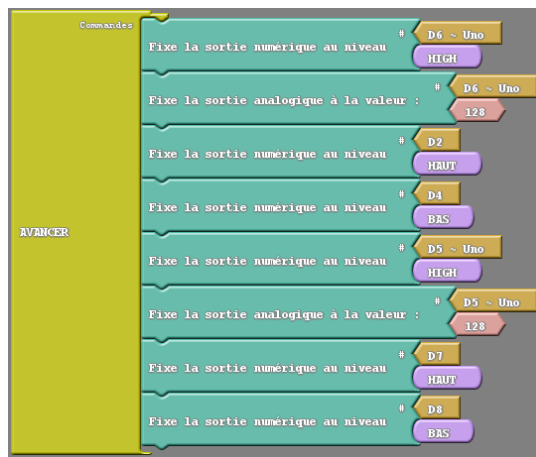
Mieux, créez deux variables supplémentaires « **VMax** » et « **Vmin** » pour changer les paramètres vitesses plus rapidement en ayant deux plages de vitesses des moteurs.



Le deuxième avantage est qu'en ne donnant pas la même vitesse à droite et à gauche on obtient des virages soit vers la gauche soit vers la droite. Une différence plus ou moins importante entre ces vitesses aura pour conséquence un virage plus ou moins prononcé.

Cela se ferait rapidement à l'intérieur de ces sous-programmes et à un seul endroit nous évitant d'avoir à rechercher les lignes adéquates à mettre à jour dans tous les sous-programmes que l'on aurait créés.

**Avant**



**Après**



## Partie 2 - Pilotage du robot.

Par équipe de 2 !!

On souhaite munir le robot d'un système « bis » de pilotage dans le cas exploration souterraine où les ondes Wifi ne passeraient pas. Un « brainstorming » sur le sujet a permis de dégager deux idées :

- La 1<sup>ère</sup> par télécommande. (Equipe 1)
- La 2<sup>ème</sup> par joystick. (Equipe 2)

Vous allez donc constituer deux équipes pour étudier chacune des solutions, les essais futurs dans les conditions difficiles définiront le choix définitif.

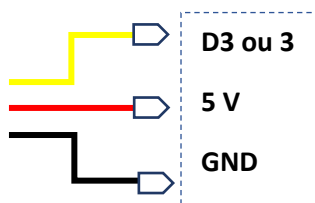
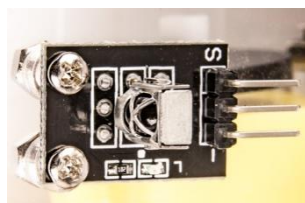
### Première équipe : la télécommande

Et si l'on utilisait une télécommande pour piloter votre robot. Pour cela il faut associer à celle-ci, un module de télécommande IR qui vous permettra de communiquer avec la machine en ayant au préalable affecté à différentes touches de celle-ci des fonctions, par exemple : Touche 1 = Avancer, Touche 2 = Reculer, ... .

**Mon Premier programme : Rechercher les codes de ma télécommande**

☒ **C.16 - Nous allons ajouter un module IR**

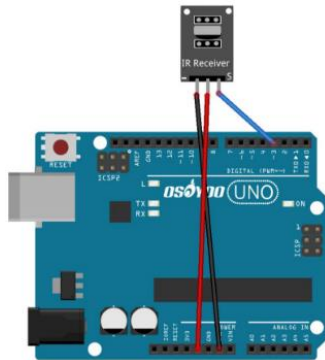
Le branchement est très simple. Quelque soit le modèle dont vous disposez, prenez votre module IR, prenez un câble Grove et réalisez le branchement suivant.



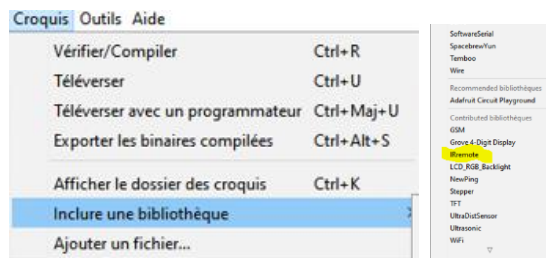
Fiche technique du composant  
Fréquence de fonctionnement : 38 KHz  
Alimentation : 2,7V à 5,5V  
Angle d'ouverture : 2 x 45°



- Reliez votre **module IR** au shield de connexion **par un câble Grove** sur la borne **D3** :

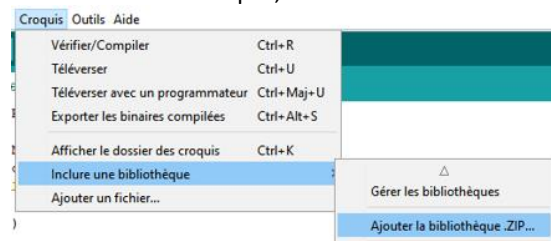


**Nota :** Pour déchiffrer les codes émis par la télécommande vous aurez besoin de la bibliothèque « IRremote ». Il est possible que la bibliothèque dont nous avons besoin ne soit pas présente dans Arduino IDE pour x raisons (nouvelle version Arduino installé, .... ). Pour le vérifier **cliquez** dans le menu sur :



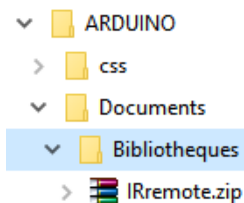
- **Listez** jusqu'en bas de la longue liste, voir surtout dans « **Contributed bibliothèques** »

Si vous ne la trouvez pas, vous allez devoir l'installer. La procédure est presque identique à ce que nous venons de faire ...



Vous trouverez ce fichier dans le dossier sur le serveur :

**Arduino/Documents/Bibliothèques/**



### Objectif du programme

1. Déterminer le code de chaque touche de la télécommande ....

- **Recopiez** le code ci-dessous et **collez** le dans l'interface Arduino
- **Téléversez** le et **procédez** aux mesures.

```
#include <IRremote.h>

int RECV_PIN = 3;
IRrecv irrecv(RECV_PIN);
decode_results results;

void setup()
{
  Serial.begin(9600);    // vitesse port série 9600 bauds
  irrecv.enableIRIn();   // Initialise le récepteur
}

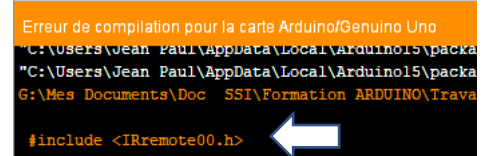
void loop() {
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX); /*valeur en
                                         Hexadécimal*/
    irrecv.resume();      // Reçoit la valeur suivante
    delay(1000);
  }
}
```

	Touche	Code	Touche	Code
	1		0	
	2		*	
	3		#	
	4		OK	
	5		→	
	6		←	
	7		↑	
	8		↓	
	9			



- **Téléversez** le programme.

Vous pouvez constater qu'il y a un problème lors du téléversement du programme .... Si vous regardez au début de celui-ci vous remarquez que le code ne correspond pas à la bibliothèque chargée dans la première partie, sans doute un bug au niveau de l'interface « **Ardublock** ». Vous devez **modifier** le code à la main. Je n'ai trouvé que cette solution pour que cela marche...



```
#include <IRremote00.h>
#include <ctype.h>
#include <Wire.h>
```



```
#include <IRremote.h>
#include <ctype.h>
#include <Wire.h>
```

## ☑ C.17 - Nous allons **revenir** à notre robot où pas mal de choses sont déjà réalisées.

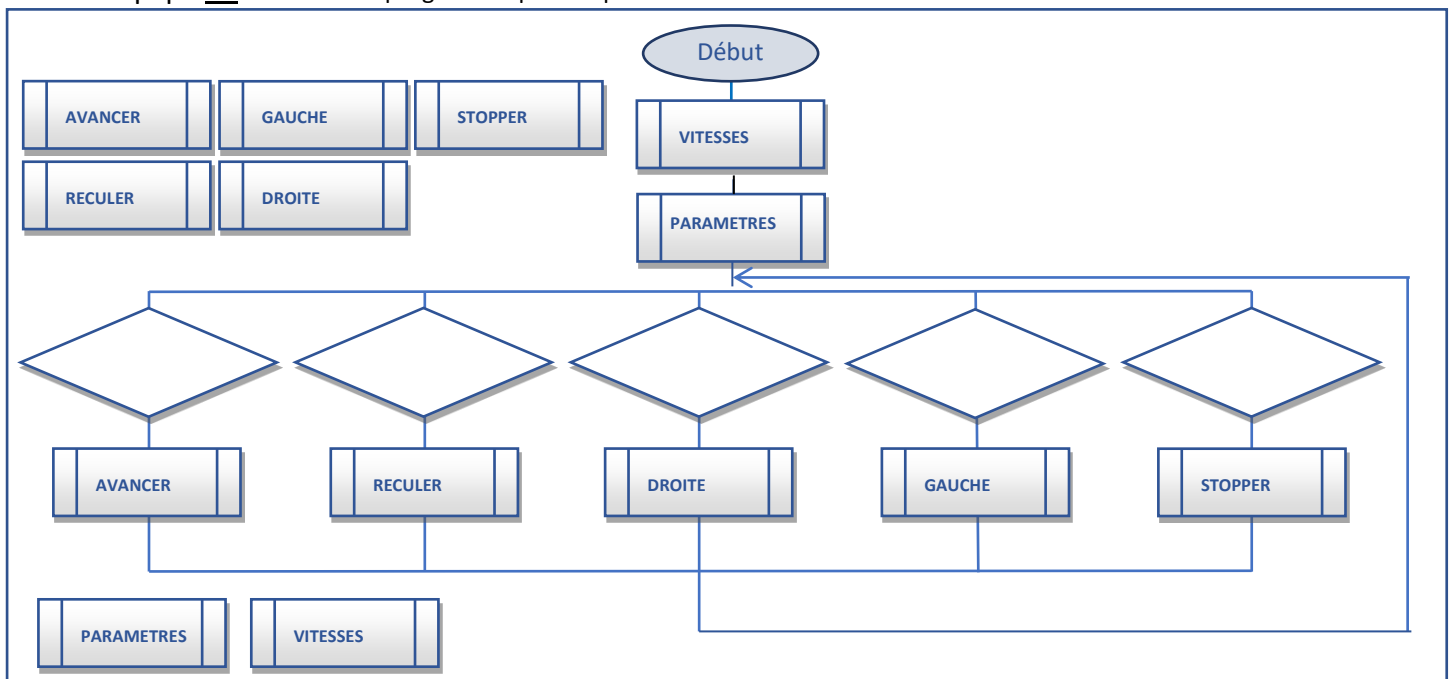
### Objectif du programme



- 1- La touche « **↑** » permettra au robot d'AVANCER.
  - 2- La touche « **↓** » permettra au robot d'RECULER.
  - 3- La touche « **←** » permettra au robot d'aller à GAUCHE.
  - 4- La touche « **→** » permettra au robot d'aller à DROITE.
  - 5- La touche « **OK** » permettra au robot de STOPPER.
- Nota : Vous pouvez aussi en choisir d'autres.



Pour l'équipe **01** : réalisez le programme pour déplacer le robot avec la télécommande.



- Déclaration du récepteur IR sur la broche **3** - bloc dans « **Communication** »
- Obtenir le code du récepteur du IR

- Affecter à la variable **IR** la valeur de **ir code** - bloc dans « **Variables/constantes** »

- Donner un nom à la variable « **Haut** » par exemple ou « **Avancer** » pour la touche « **↑** »
- Le code IR est « ..... » voir votre tableau.

Pour chaque mouvement souhaité du robot, **Indiquez** dans les différentes cases le **nom de la variable et sa valeur**

....

- Si la valeur « **IR** » captée est **égale** au code « **Centre** » le robot est à l'arrêt.

- **Téléversez le programme.**

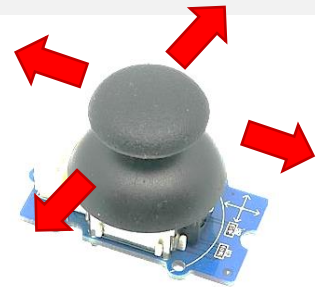
## Deuxième équipe : le joystick

### Objectif du programme



1. Déterminer les positions extrêmes du joystick  
-> suivant 4 directions (tous les 90°).

On souhaite commander le robot à l'aide d'un joystick, suivant sa position de celui-ci il se déplacera en Avant, à Droite, à Gauche, fera marche Arrière ou ne bougera pas si la manette reste au centre.



- **Réalisez** le câblage en positionnant le joystick sur le Shield sur la broche **A0** ou **0**.
- **Réalisez** le programme pour effectuer ces mesures.  
Deux façons de procéder pour connaître les coordonnées de position du joystick :
  - Soit vous **copiez/collez** le code dans l'IDE Arduino.
  - Soit vous **réalisez** le programme avec Ardublock (une fois avec X, 1 fois avec Y).



/\* Coordonnées de position du Joystick GROVE connectez le module sur A0 et A1 \*/


```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int jhoriz = analogRead(A0);
  int jvert = analogRead(A1);

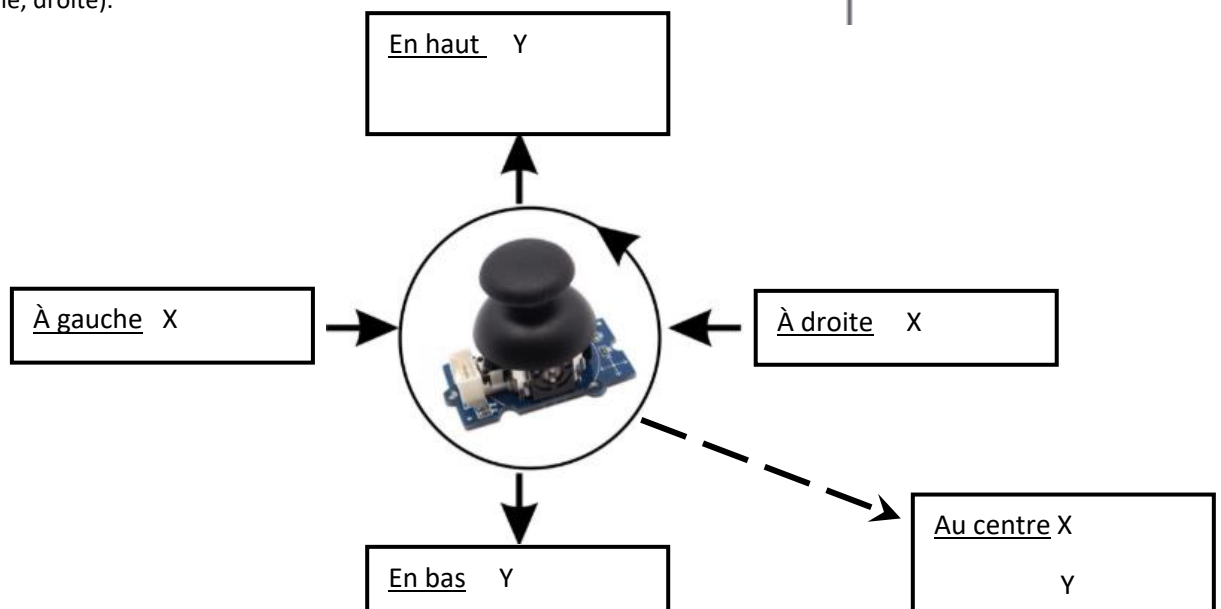
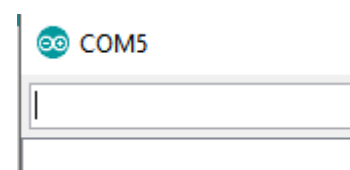
  Serial.print("Les coordonnées X et Y sont :");
  Serial.print(jhoriz, DEC);
  Serial.print(",");
  Serial.println(jvert, DEC);
  Serial.println(" ");
  delay(1000);
}
```

### Données du constructeur (cela nous permettra de vérifier)

Item	Min	Typical	Max	Unit
Working Voltage	4.75	5.0	5.25	V
Output Analog Value (X coordinate)	206	516	798	\
Output Analog Value (Y coordinate)	203	507	797	\

En ouvrant le moniteur  série on affiche donc les 2 valeurs en fonction du positionnement du Joystick, on remarque que si on ne touche pas le Joystick, on obtient 2 valeurs qui seront notre référence (au centre).

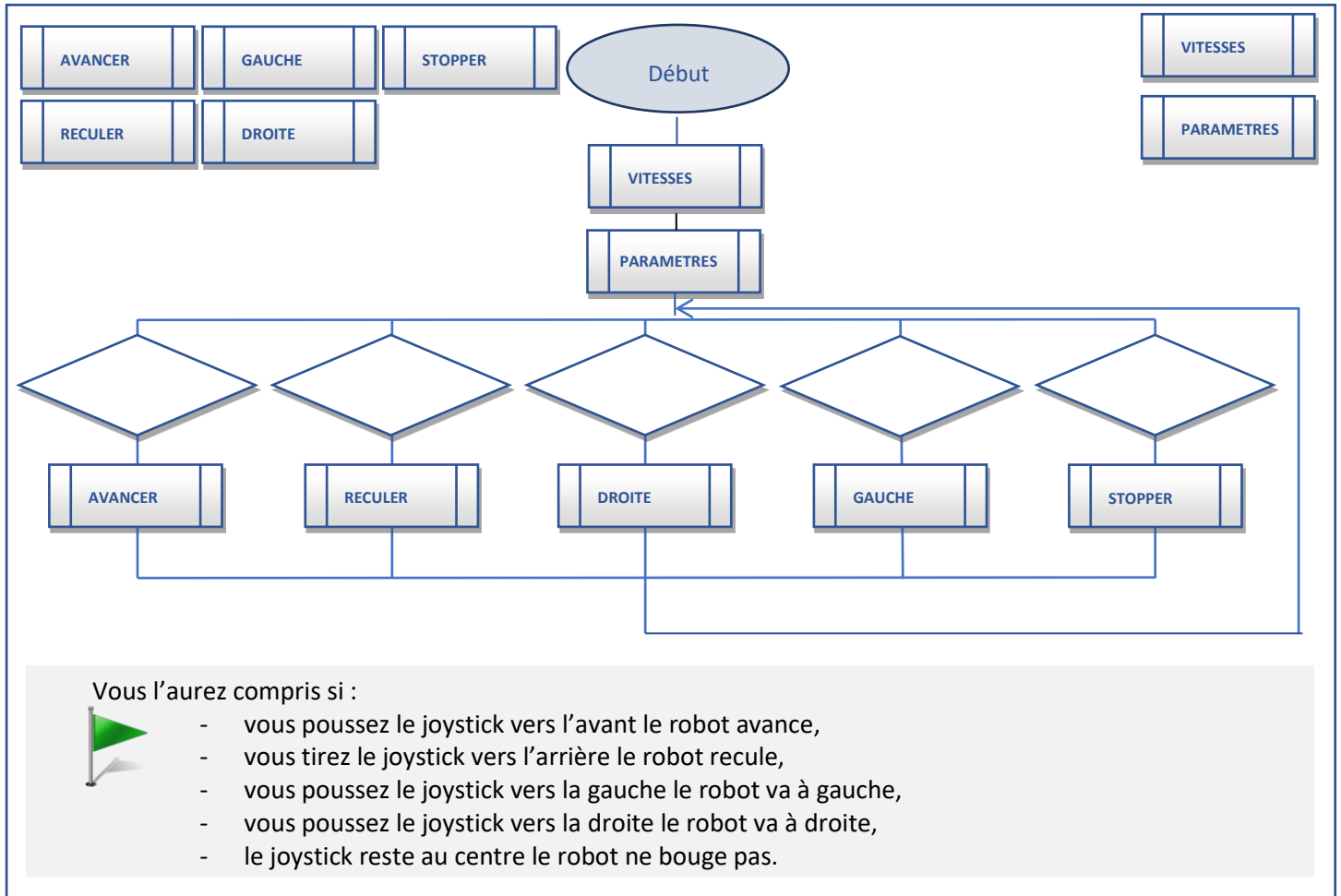
- **Relevez et positionnez** sur le graphique ci-dessous les valeurs x et y donnant la position du joystick dans les 4 directions qui nous intéressent (centre, haut, bas, gauche, droite).



☑ **C.18** – Vos résultats **sont-ils conformes** à ceux donnés par le constructeur.

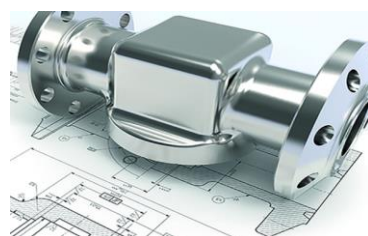
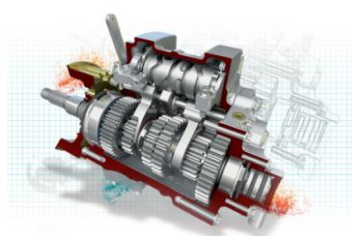
Maintenant que les limites de ce joystick sont connues nous allons adapter les amplitudes du joystick (4 directions) à la direction à prendre par le robot.

Pour l'équipe **02** : réalisez le programme pour déplacer le robot avec un joystick.



☑ **C.19** – Continuez le programme principal en complétant **la partie boucle**.





# D-CONCEPTION D'UNE SOLUTION-DAO

## Partie 4 – Fixation du/des joystick(s) sur un support ergonomique.



### Objectif

1. Définir la forme du support pour fixer le joystick.

- ☑ **D.1 – Indiquez sur la photo les dimensions du composant qui vous serviront à son implantation.**  
(Hauteur, largeur, espacement des trous de fixations, ..... ) A mesurer sur l'original.



- ☑ **D.2 – Avant de commencer avec le logiciel de DAO Solidworks, faites ci-dessous un croquis 3D et 2D de la solution que vous avez imaginé pour ce couvercle qui viendra se fixer quelque part sur la poubelle.**



# E-PRESENTATION ORALE

## Partie 5 – Je présente mes solutions



### Objectifs

1. Présenter le résultat de votre travail à l'ensemble de la classe.
2. Répartir le temps de parole entre les acteurs du groupe (au moins 5min/pers.).
3. Préparer un diaporama.

- ☒ **E.1 – Pour préparer cet oral complétez le tableau ci-dessous qui fait l'inventaire du contenu à présenter.**

Diaporama	Temps	Qui	Élément important à citer pendant cette partie
Contexte			
Situation / problème / Besoin			
Cahier des charges			

- ☒ **E.2 – Consultez la fiche des erreurs à éviter.**

### Les erreurs les plus fréquentes à éviter :

- Pour ne pas se perdre dans un nombre trop important de diapositives il convient de retenir le chiffre suivant 1 diapo pour environ 2 min d'oral
- Attention aux fautes d'orthographe.
- Ne pas lire ce qui est écrit mais commenter la diapo.
- Attention aux hauteurs de caractères