

### III. Robot Makeblock

Tu vas réaliser plusieurs programmes pour aboutir au fonctionnement définitif du robot. Chaque programme sera testé avant la rédaction du suivant :

#### a. Déplacements

Le premier programme à pour but de vérifier les câblages et fonctionnement du pont en H et des boutons poussoir. La création de « sous-programmes » utilisables dans les programmes suivants est à privilégier. Seront testés dans ce programme : les boutons poussoirs rouge et noir, la marche avant et arrière, la rotation à droite et à gauche du robot :

- Un appui sur le bouton poussoir noir provoque le déplacement du robot vers l'avant
- 3 secondes plus tard, le robot tourne à droite (moteur droit en arrière vitesse max, moteur gauche en avant vitesse max)
- 2 secondes plus tard, le robot tourne à gauche (moteur droit en avant vitesse max, moteur gauche en arrière vitesse max)
- 2 secondes plus tard, le robot recule jusqu'à un appui sur le bouton rouge

*Astuce : Créer un variable « STOP » permettant d'enregistrer une demande d'arrêt par le bouton rouge*

#### b. Arrêt au mur

Le deuxième programme à pour but de vérifier le câblage et fonctionnement des 3 capteurs de proximité. La création de « sous-programmes » utilisables dans les programmes suivants est toujours à privilégier :

- Un appui sur le bouton poussoir noir provoque le déplacement du robot vers l'avant
- Un appui sur le bouton rouge provoque l'arrêt du déplacement du robot
- Si pendant le déplacement vers l'avant, l'un des 3 capteurs de proximité robot détecte une distance inférieure ou égale à 6 cm, le robot doit :
  - S'arrêter
  - Reculer jusqu'à ce que les 3 capteurs de proximité détectent une distance d'au moins 20 cm
  - S'arrêter

*Astuces : Faire des sous-programmes pour la lecture des capteurs ultrason*

*Une condition par capteur ultrason*

*Tester le programme avec chacun des capteurs branché seul, puis avec les 3 branchés*

#### c. Suiveur de ligne

Le troisième programme à pour but de vérifier le câblage et fonctionnement du capteur suiveur de ligne, et de coordonner les signaux envoyés par celui-ci au déplacement du robot. La création de « sous-programmes » utilisables dans les programmes suivants est toujours à privilégier. Lors du test de ce programme il sera nécessaire de régler la sensibilité du capteur de ligne (dépend de la luminosité ambiante) comme expliqué dans la datasheet.

- Un appui sur le bouton poussoir noir provoque le déplacement du robot vers l'avant avec suivi de la ligne noire :
  - Si l'élément central du capteur est au niveau haut, le robot est sur la bonne trajectoire : il faut continuer tout droit
  - Si l'élément central du capteur est au niveau bas et l'élément gauche du capteur au niveau haut, le robot s'écarte de sa trajectoire en partant sur la droite : il doit aller à gauche.
  - Si l'élément central du capteur est au niveau bas et l'élément droit du capteur au niveau haut, le robot s'écarte de sa trajectoire en partant sur la gauche : il doit aller à droite.
- Un appui sur le bouton rouge provoque l'arrêt du déplacement du robot

*Astuce : Changer de direction si seulement un des 3 éléments détecte la ligne*

#### d. Labyrinthe

Ce programme est la finalité de l'exercice. Le but est de permettre au robot de sortir seul du labyrinthe en suivant la ligne au sol. Un obstacle sera placé dans le labyrinthe pendant l'essai. Le robot devra s'arrêter devant cet obstacle et ne repartira qu'une fois l'obstacle enlevé :

- Un appui sur le bouton poussoir noir provoque le déplacement du robot vers l'avant avec suivi de la ligne noire :
  - Si l'élément central du capteur est au niveau haut, le robot est sur la bonne trajectoire : il faut continuer tout droit
  - Si l'élément central du capteur est au niveau bas et l'élément gauche du capteur au niveau haut, le robot s'écarte de sa trajectoire en partant sur la droite : il doit aller à gauche.
  - Si l'élément central du capteur est au niveau bas et l'élément droit du capteur au niveau haut, le robot s'écarte de sa trajectoire en partant sur la gauche : il doit aller à droite.
  - Si pendant un déplacement, l'un des 3 capteurs de proximité robot détecte une distance inférieure ou égale à 6 cm, le robot doit :
    - S'arrêter jusqu'à ce que les 3 capteurs de proximité détectent une distance d'au moins 20 cm

- Un appui sur le bouton rouge provoque l'arrêt du déplacement du robot

*Astuces : Créer une fonction « obstacle » qui sera appelée lors de la détection d'un obstacle, dans laquelle le robot sera arrêté, les capteurs ultrason relus toutes les 0.1sec pour être comparés à la valeur souhaitée. Si l'un des capteurs n'atteint pas la valeur voulue, rappeler cette même fonction.*

### e. BONUS : Pilotage par Bluetooth

Le dernier programme est un bonus pour les équipes les plus avancées. Le but étant de piloter le robot à l'aide du module Bluetooth et de l'application « Bluetooth Electronics ». Celle-ci doit gérer tous les déplacements possibles. Pour sécuriser le fonctionnement, la marche avant sera interdite si l'un des 3 capteurs de proximité robot détecte une distance inférieure ou égale à 10 cm. Le capteur de ligne n'est pas pris en compte. Aucune indication de plus n'est donnée pour ce programme : à toi de jouer ! Un supplément de points sur la note de TP est bien sûr à la clé !

*Astuces : Créer une nouvelle télécommande, et sélectionner « copy to panel » dans « Library » → « RC car démo » Il y a deux ports série sur le Leonardo, le Serial et le Serial1. Serial correspond au port série généré sur l'USB, Serial1 aux broches RX et TX. Ici, il faut donc utiliser la commande « serial1 »*

#### Matériel :

- Deux boutons poussoirs : un rouge « BPR » et un noir « BPN »

- Un pont en H (L293D) pilotera les 2 moteurs avec les mots clés suivants associés :

- « **rotation** » : mise en fonctionnement du moteur (broche EN du pont en H)
- « **avant** » : rotation du moteur en sens horaire...
- « **arrière** » : rotation du moteur en sens...

...suivi de « **\_gauche** » pour le moteur de gauche ou de « **\_droite** » pour le moteur de droite (vue de l'arrière du robot)

- 3 capteurs à ultrasons (HC SR-04) empêcheront les collisions. Les mots clés suivants seront associés :

- « **echo** » pour la broche « output echo »...
- « **trig** » pour la broche « trig input »...

...suivi de « **\_gauche** » pour le capteur de gauche, ou « **\_centre** » pour le capteur central, ou encore « **\_droite** » pour celui de droite (vue de l'arrière du robot)

- Un capteur suiveur de ligne (AXE121) avec les mots clés suivants associés :

- « **suiveur\_droite** » : élément détectant la ligne à droite du robot (vue de l'arrière du robot)
- « **suiveur\_centre** » : élément détectant la ligne au centre du robot (vue de l'arrière du robot)
- « **suiveur\_gauche** » : élément détectant la ligne à gauche du robot (vue de l'arrière du robot)

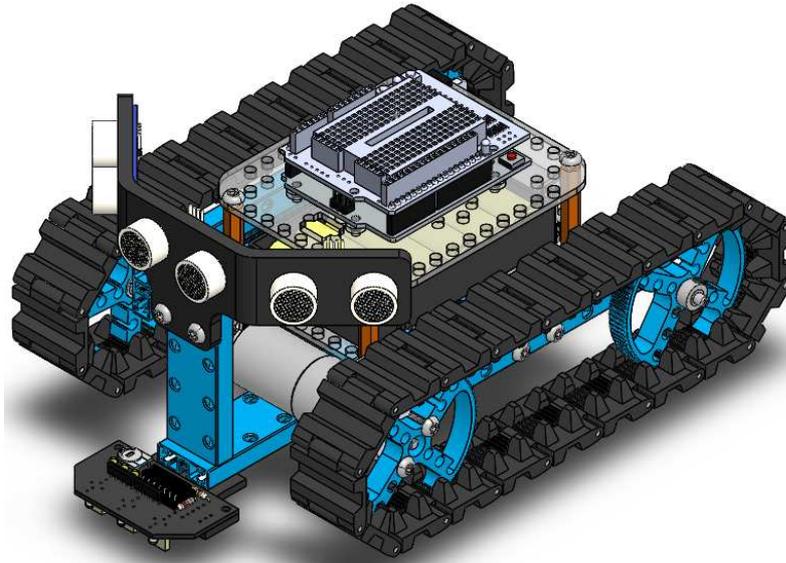
- Un module Bluetooth (HC05) permettra la connexion avec un smartphone, avec les mots clés suivants associés :

- « **Serial** » : utilisation du port série du Leonardo à 9600 bauds (broches 0 et 1)
- « **commande** » : variable de type « char » permettant d'enregistrer les instructions envoyées par Bluetooth.

1) A l'aide des datasheets « L293D », « HCSR04 », « AXE121 », « HC05 » et des exercices précédents, **complète le schéma de câblage** (utilise des couleurs différentes pour une lecture plus rapide, et trace les traits pour qu'ils ne se coupent pas grâce aux ponts déjà dessinés) **page suivante** pour :

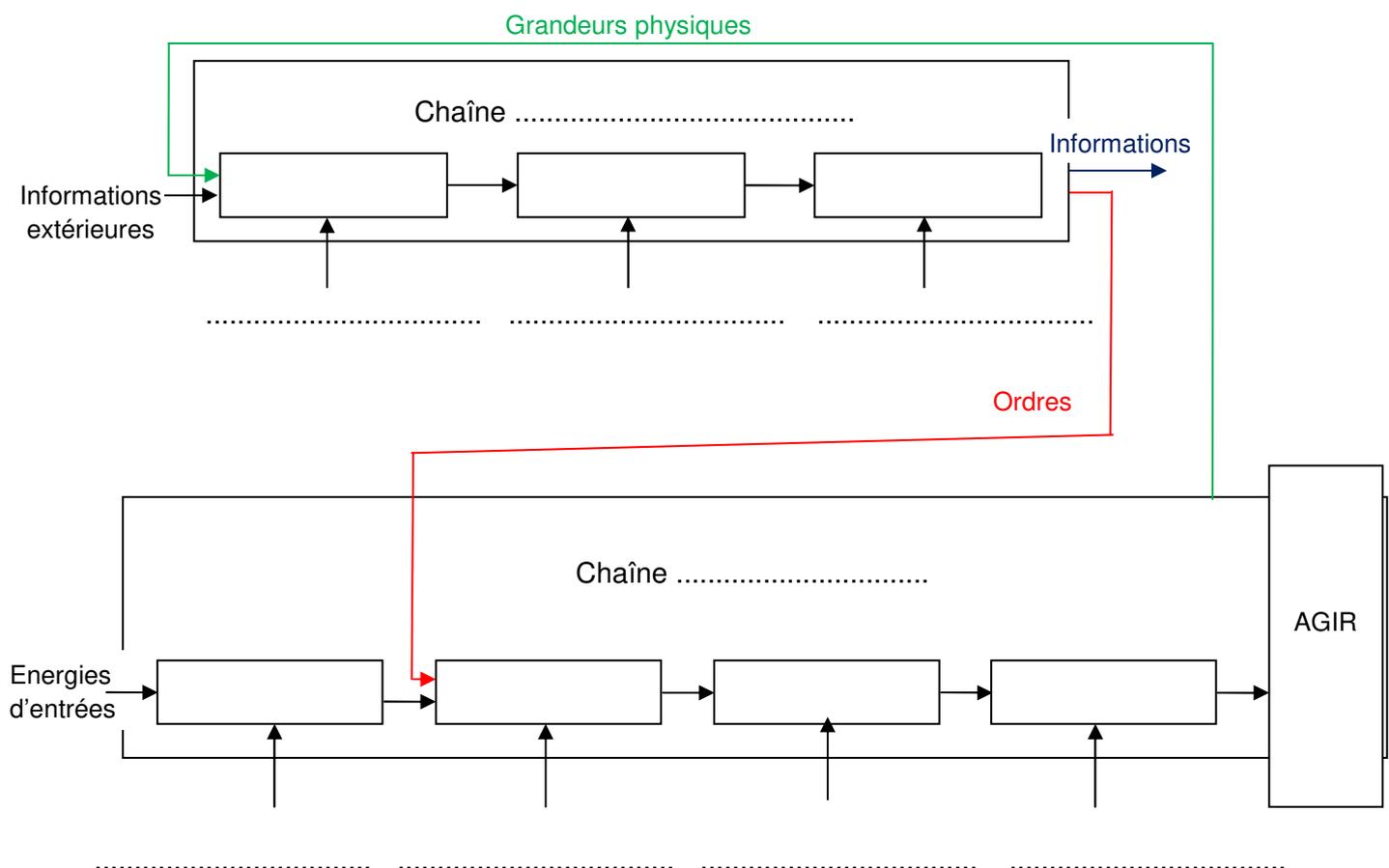


2) A partir de la notice de montage disponible dans le classeur et du matériel **Makeblock** mis à ta disposition, construis le robot suivant :



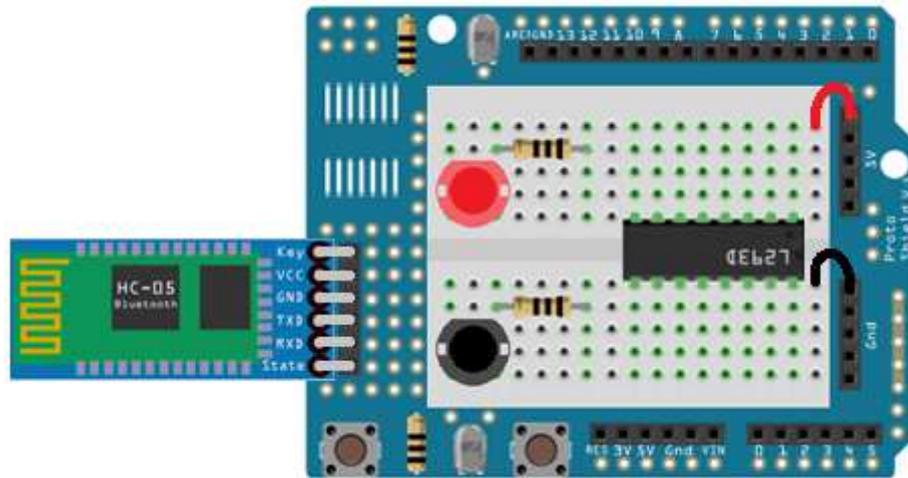
**ATTENTION A METTRE LES VIS SIMPLEMENT EN APPUI SANS SERRER :  
LES PIÈCES SONT EN ALUMINIUM OU EN PLASTIQUE !**

3) Compléter les chaînes d'informations et d'énergie du robot réalisé :



4) Faire vérifier le schéma de câblage et les chaînes d'informations et d'énergie au professeur.

5) A l'aide du matériel électronique mis à ta disposition, **place les composants sur la mini breadboard de la platine de prototypage Arduino** comme indiqué sur l'image ci-dessous. Pour la valeur des résistances, se référer à la page précédente.



6) A l'aide du schéma de câblage réalisé précédemment, **câble l'ensemble des composants électroniques sans raccorder le bloc piles à la prise d'alimentation de l'Arduino**

7) Appelle le professeur pour la vérification

**8) Réalisation et test des programmes du robot :**

Les programmes doivent être rédigés puis testés dans l'ordre. Pour chacun d'eux tu dois réaliser les instructions listées ci-dessous (de a à f).

a) Rédiger le programme (**voir détails pages 25 et 26**). Ceux-ci doivent être enregistrés sous les noms suivants :

- « **a\_deplacements** » pour le programme « **a. Déplacements** »
- « **b\_arret** » pour le programme « **b. Arrêt au mur** »
- « **c\_ligne** » pour le programme « **c. Suiveur de ligne** »
- « **d\_labyrinthe** » pour le programme « **d. Labyrinthe** »
- « **e\_bluetooth** » pour le programme « **e. BONUS : Pilotage par Bluetooth** »

b) Vérifie que le bloc piles n'alimente pas la carte Arduino, puis relie le port USB de l'Arduino à celui de l'ordinateur par l'intermédiaire du HUB USB

c) Téléverse le programme dans l'Arduino (**pense à enlever le module Bluetooth le temps du téléversement**)

d) Débranche le câble USB de la carte Arduino, puis Raccorde le bloc piles à la prise d'alimentation de l'Arduino.

f) **Vérifie que le fonctionnement du robot correspond à celui attendu. Si ce n'est pas le cas, modifie ton programme et répète les opérations b) à f), jusqu'à obtention du fonctionnement attendu.**

g) **Appelle le professeur pour la validation (tout programme non validé à cette étape est considéré comme non fait)**

**Une fois l'ensemble des programmes testés, démonte tous les composants et range-les dans leurs boites en respectant les emplacements indiqués sur les photos.**