

## Guide de mise en marche du châssis AlphaBot2 pour Arduino.

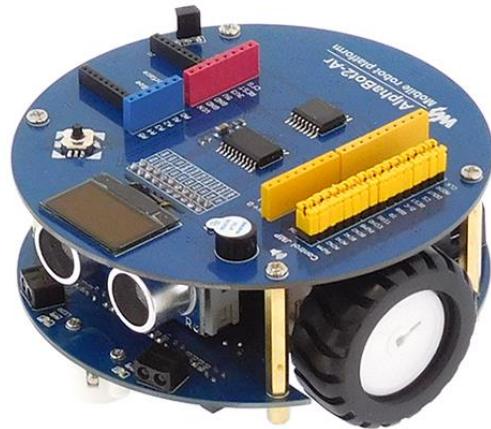
### Matériel nécessaire :

- 1 x carte [Uno ou compatible](#)
  - 1 x châssis AlphaBot2-Ar
  - 2 x [accus Li-Ion 14500](#)
  - 1 x [chargeur d'accus Li-Ion](#)
  - 1 x module [Bluetooth Xbee](#)
  - Un cordon USB en fonction du modèle de votre carte Uno
  - L'IDE Arduino installé sur votre ordinateur
- 

### Présentation du châssis :

Châssis circulaire en kit AlphaBot2-Ar sans soudure comportant le nécessaire pour la réalisation d'un projet robotique à base d'Arduino Uno ou compatible (**non incluse**).

Ce châssis dispose de deux plateformes embarquant plusieurs modules: des leds RGB, un support Xbee (module Xbee Bluetooth, non inclus), un afficheur OLED, un mini HP, un récepteur IR, un détecteur à ultrasons, des capteurs de proximité et des suiveurs de lignes.



Cette base peut être pilotée de plusieurs façons:

- Avec une télécommande infrarouge (inclus).
- Par une liaison Bluetooth via un module Xbee (non inclus).
- De façon autonome, en développant vous-même un programme via l'IDE Arduino utilisant les différents capteurs présents sur la plateforme et le capteur à ultrasons (suiveur de ligne, détection d'obstacle). D'autres capteurs peuvent être ajoutés grâce aux E/S de la carte Uno restant accessibles.

Pour se déplacer, cette plateforme comporte deux motoréducteurs avec deux roues en caoutchouc et deux roues omnidirectionnelles contrôlées par un double pont en H TB6612FNG.

Ce châssis dispose d'un bouton marche/arrêt et d'un support pour deux accus Li-Ion 14500 3,7 Vcc (non inclus, voir [09500](#)) alimentant la plateforme et la carte Arduino.

## Téléchargement et installation des librairies :

Les différents exemples présentés dans ce guide de démarrage nécessite le téléchargement et l'installation de Librairies et de programmes compatible Arduino®.

Rendez-vous sur ce lien :

[https://drive.google.com/open?id=1mdie5WoBiOJntO3dxM5ROL8Dhtq\\_MkMF](https://drive.google.com/open?id=1mdie5WoBiOJntO3dxM5ROL8Dhtq_MkMF)

Téléchargez le fichier \*.ZIP en cliquant sur l'icône :



Une fois le fichier ZIP téléchargé, placez-vous dans votre répertoire ou sont téléchargé les fichiers.

Extraire le fichier .ZIP avec un logiciel du type : [WinRAR®](#), [7Zip®](#).

Vous devez obtenir deux répertoires :



Effectuez un copier/coller du **contenu** du dossier « **libraries** » vers :

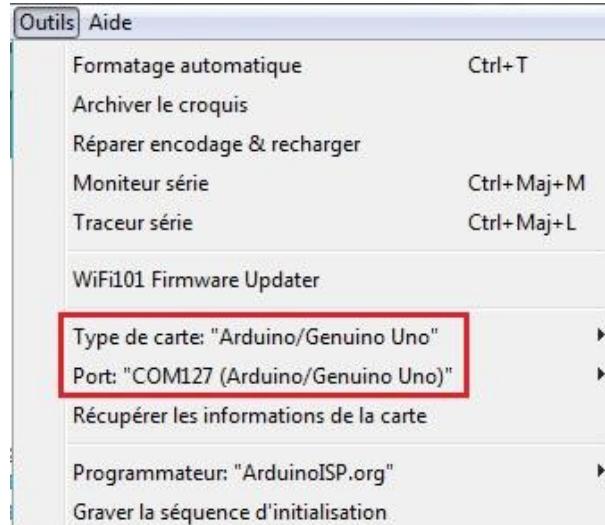
**C:\Utilisateurs\\*Votre utilisateur\*\Mes documents\Arduino\libraries**

Le dossier « **demo** » contient les différents programmes exemples permettant le fonctionnement du châssis Alphabot2-Ar.

## Essai des moteurs :

Ouvrez le dossier **demo\Run\_Test** et exécutez Run\_Test.ino avec l'IDE Arduino. Raccordez votre carte Uno à votre PC avec un cordon USB.

Selectionner le port et le type de carte correspondant à votre carte :



Compilez et téléversez le programme dans la carte Uno :



Une fois le programme envoyé, le robot va se déplacer lentement vers l'avant. La faible vitesse de déplacement est due à l'intensité trop faible délivrée par le port USB du PC.

En déconnectant le port USB et en mettant l'inverseur du châssis sur position ON, le robot effectuera un déplacement rectiligne avec une bonne vitesse.

## Essai avec le joystick 4 direction + bouton-poussoir :

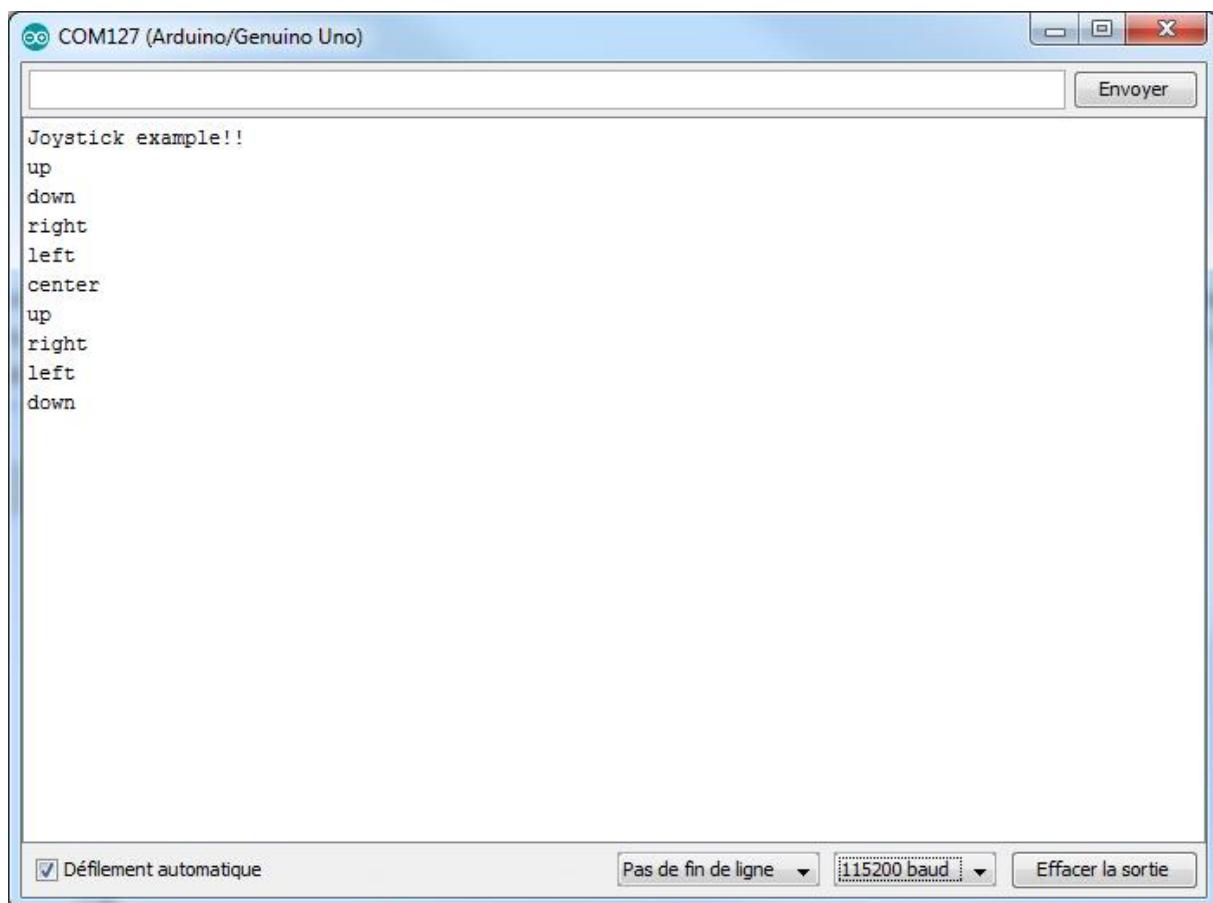
Ouvrez le programme Joystick.ino, compilez et téléversez-le dans la carte Uno.

Débranchez le cordon USB et allumez le châssis.



Le châssis se dirigera maintenant en fonction de la direction appuyée sur le joystick. Le buzzer s'active lorsque le joystick est manipulé.

En rebranchant le cordon USB et en exécutant le moniteur série (CTRL+MAJ+M), vous devrez obtenir un retour des différentes directions demandées via le bouton poussoir. Le moniteur série doit être réglé sur 115200 bauds.



The screenshot shows the Arduino Serial Monitor window titled "COM127 (Arduino/Genuino Uno)". The text area displays the following data:

```
Joystick example!!
up
down
right
left
center
up
right
left
down
```

The bottom of the window shows the following settings and buttons:

- Défilement automatique
- Pas de fin de ligne
- 115200 baud
- Effacer la sortie

## Télécommande Infrarouge :

Ouvrez dans l'IDE Arduino le programme nommé **IR.ino**. Compilez et téléversez ce programme dans la carte Uno.

Une fois ce programme envoyé vous pourrez diriger le robot grâce à la télécommande IR incluse.

Attribution des touches :

2 : en avant

8 : en arrière

4 : tourner à gauche

6 : tourner à droite

Volume +/- : augmente ou diminue la vitesse de déplacement

EQ : remise à zéro de la vitesse

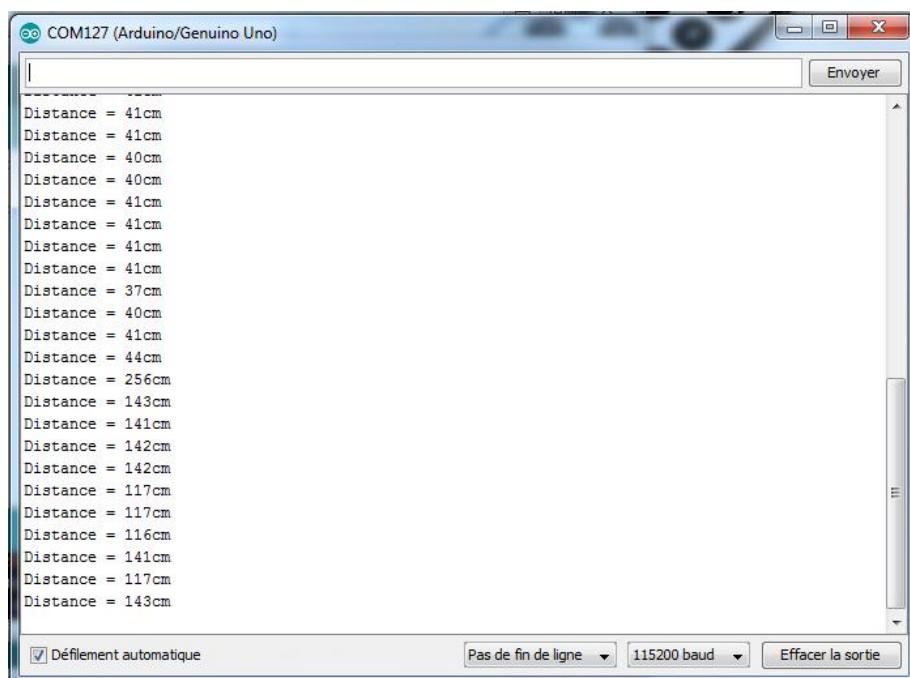
## Mesure de distance avec le capteur US :

Ouvrez dans l'IDE Arduino le programme **Ultrasonic\_Ranging.ino**.

Compilez et téléversez le programme dans la carte Uno.

Lancez maintenant le moniteur série (CTRL+MAJ+M) et réglez le sur 115200 bauds.

Vous devriez voir la distance mesurée par le capteur US apparaître dans le moniteur série.



## Détection d'obstacle avec le capteur US :

Ouvrez dans l'IDE Arduino le programme **Ultrasonic-Infrared-Obstacle-Avoidance.ino**.

Compilez et téléversez le programme dans la carte Uno.

Placez le robot au sol et mettez le sous tension. Le robot devrait aller tout droit et changer de direction dès la détection d'un obstacle.

## Test des capteurs suiveurs de lignes :

Ouvrez dans l'IDE Arduino le programme **TRSensorExample.ino**.

Compilez et téléversez le programme dans la carte Uno.

Les capteurs de lignes permettent la détection d'une ligne noire sur un fond blanc. Pour cet exemple, il est conseillé d'utiliser de l'adhésif noir d'environ 15 mm de large sur une feuille A4 blanche.

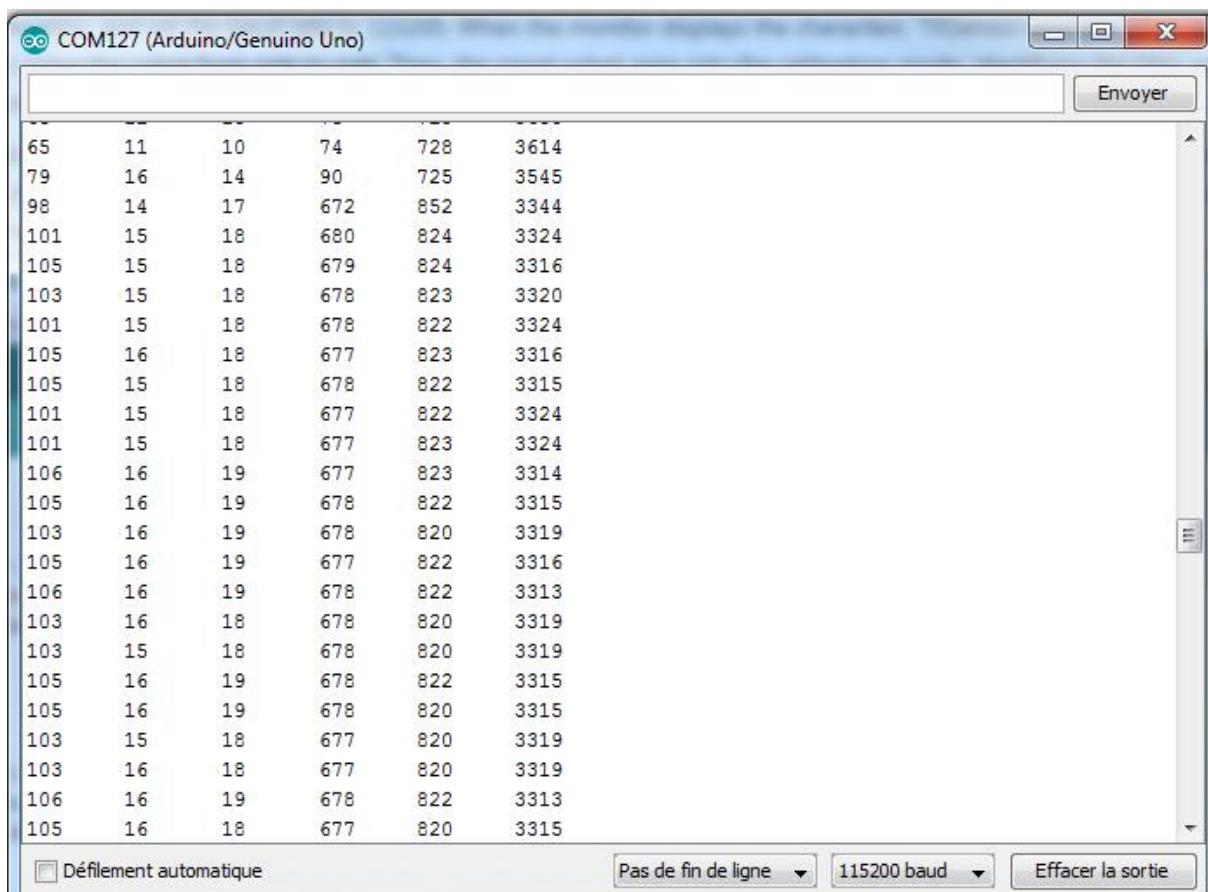
Ouvrez le moniteur série (CTRL+MAJ+M) et réglez le débit à 115200 bauds. Pendant que l'Arduino paramètre les capteurs, déplacer le robot de droite à gauche sur la ligne noire afin d'obtenir une bonne calibration.

Une fois le message « **calibration done** » passé, vous verrez apparaître dans le moniteur série les valeurs min et max de chaque capteur :

Par exemple :

```
TRSensor example
calibrate done
44 56 87 43 47 <- valeurs mini
620 946 945 935 941 <- valeurs maxi
```

Après une calibration de quelques seconds, les mesures des 5 capteurs apparaissent dans le moniteur série :



1	2	3	4	5	6
65	11	10	74	728	3614
79	16	14	90	725	3545
98	14	17	672	852	3344
101	15	18	680	824	3324
105	15	18	679	824	3316
103	15	18	678	823	3320
101	15	18	678	822	3324
105	16	18	677	823	3316
105	15	18	678	822	3315
101	15	18	677	822	3324
101	15	18	677	823	3324
106	16	19	677	823	3314
105	16	19	678	822	3315
103	16	19	678	820	3319
105	16	19	677	822	3316
106	16	19	678	822	3313
103	16	18	678	820	3319
103	15	18	678	820	3319
105	16	19	678	822	3315
105	16	19	678	820	3315
103	15	18	677	820	3319
103	16	18	677	820	3319
106	16	19	678	822	3313
105	16	18	677	820	3315

La première colonne est la mesure du 1<sup>er</sup> capteur de gauche, la 5ème colonne celle du capteur le plus à droite et la 3eme colonne est la valeur du capteur central.

Les valeurs varie de 0 à 1000 les valeurs les plus élevées sont celles ou les capteurs son situés les plus proches de la ligne noire.

La **6ème colonne** indique la position de la ligne noire :

- vers **2000**, la ligne noire est vers le milieu du robot.
- Vers **0**, la ligne noire est vers le côté gauche du robot.
- Vers **4000**, la ligne noire est vers le côté droit du robot.

## Test des leds RGB :

Ouvrez dans l'IDE Arduino le programme **WS2812.ino**. Compilez et téléversez le programme dans la carte Uno.

Pour modifier les couleurs, modifiez les valeurs RGB de chaque leds. Pour qu'une led éclaire rouge le code RGB est (255, 0, 0), pour du vert (0, 255, 0).

Exemple de programme avec toute les Leds sur rouge :

```
#include <Adafruit_NeoPixel.h>

#define PIN 7

Adafruit_NeoPixel RGB = Adafruit_NeoPixel(4, PIN, NEO_GRB + NEO_KHZ800);

void setup() {
    // put your setup code here, to run once:
    RGB.begin();
    RGB.setPixelColor(0, RGB.Color(0, 255, 0));
    RGB.setPixelColor(1, RGB.Color(255, 0, 0));
    RGB.setPixelColor(2, RGB.Color(0, 255, 0));
    RGB.setPixelColor(3, RGB.Color(255, 0, 0));
    RGB.show();
}

void loop() {
    // put your main code here, to run repeatedly:
}
```

## Test de l'afficheur OLED :

Ouvrez dans l'IDE Arduino le programme **oled.ino**.

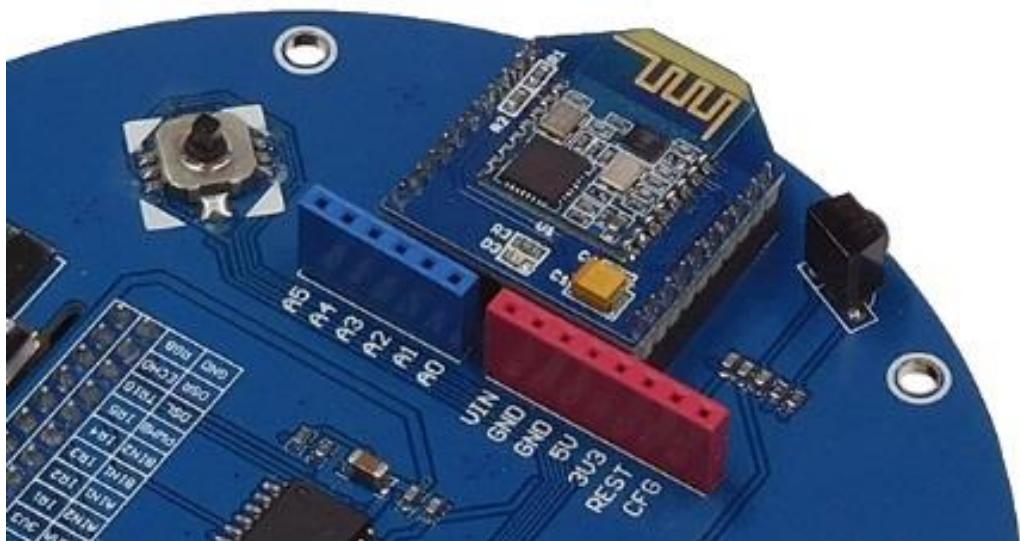
Compilez et téléversez le programme dans la carte Uno.

Le logo AlphaBot2 doit apparaître sur l'écran OLED.

## Contrôle via une liaison Bluetooth :

Ouvrez dans l'IDE Arduino le programme **Bluetooth.ino**. Compilez et téléversez le programme dans la carte Uno.

Installez le module Xbee Bluetooth (34250 conseillé) dans le châssis. Une fois le module Xbee installé il sera impossible de téléverser un programme dans la carte Uno, le Module Xbee utilisant le port série pour communiquer avec la carte Uno.



Cette utilisation requiert une application à installer sur votre smartphone ou tablette, voici les liens de téléchargement :

Application **iOS** : [Téléchargement](#)

Application **Android**, APK à installer manuellement sur votre smartphone\* : [Téléchargement](#)

\* A télécharger directement sur le smartphone. Activez les **sources inconnues** dans Paramètres -> Paramètres avancés -> Sécurité pour pouvoir installer l'application.

Une fois le robot alimenté, lancez l'application sur votre smartphone et connectez-vous au module Xbee. Le Bluetooth doit être activé sur votre smartphone.

Dans la liste, sélectionnez l'appareil avec comme adresse MAC : **00:0E:0E...**

**Ensuite, vous devrez configurer les touches de déplacement :**

Appuyez sur « **Remote Control** » et ensuite sur « **Config** ».

Appuyez sur **Forward (keydown)** et saisissez : **Forward**

Appuyez sur **Forward (keyup)** et saisissez : **Stop**

Appuyez sur **Backward (keydown)** et saisissez : **Backward**

Appuyez sur **Backward (keyup)** et saisissez : **Stop**

Appuyez sur **Left (keydown)** et saisissez : **Left**

Appuyez sur **Left (keyup)** et saisissez : **Stop**

Appuyez sur **Right (keydown)** et saisissez : **Right**

Appuyez sur **Right (keyup)** et saisissez : **Stop**

Appuyez sur **Low** et saisissez : **Low**

Appuyez sur **Medium** et saisissez : **Medium**

Appuyez sur **High** et saisissez : **High**

Revenez en arrière, sur le menu disposant des flèches directionnelles.

En appuyant sur les touches vous déplacerez votre robot dans la direction désirée. Suivant la touche appuyée, les Leds RGB doivent s'allumer de différentes façons.

Les touches Low, Medium et High permettent de choisir la vitesse de déplacement du robot.

# GO TRONIC

ROBOTIQUE ET COMPOSANTS ÉLECTRONIQUES

Si vous rencontrez des problèmes, merci de nous contacter par courriel à :

[sav@gotronic.fr](mailto:sav@gotronic.fr)