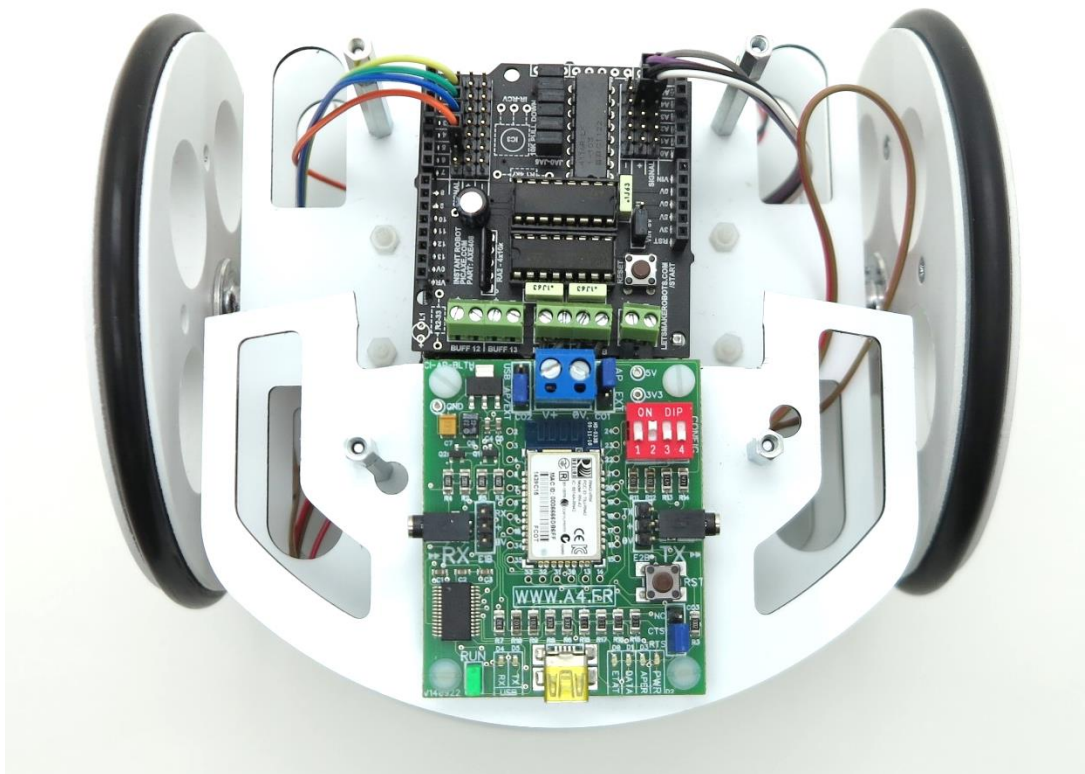


# RoboCoda

Plateforme robotique modulaire

**Option Bluetooth  
avec PICAXE et ApplInventor**





PICAXE est une marque de la Sté Revolution Education.

AutoProg est un système développé par la Sté A4, qui utilise des microcontrôleurs PICAXE.



L'ensemble des ressources numériques disponibles autour de nos projets et maquettes sont téléchargeables librement et gratuitement sur [www.a4.fr](http://www.a4.fr)

La duplication de ce dossier est autorisée sans limite de quantité au sein des établissements scolaires, aux seules fins pédagogiques, à la condition que soit cité le nom de l'éditeur : Sté A4.

La copie ou la diffusion par quelque moyen que ce soit à des fins commerciales n'est pas autorisée sans l'accord de la Sté A4.



APP INVENTOR est un environnement de programmation orientée objet, accessible aux non-initiés pour concevoir des applications Android.



Edité par la société A4 Technologie  
Tél. : 01 64 86 41 00 - Fax : 01 64 46 31 19  
[www.a4.fr](http://www.a4.fr)

## SOMMAIRE

<b>1. Introduction.....</b>	<b>2</b>
1.1. Prérequis souhaitables .....	2
1.2. Organisation de ce dossier .....	2
<b>2. Eléments nécessaires .....</b>	<b>3</b>
2.1. Matériels .....	3
2.2. Logiciels.....	3
2.3. Ressources complémentaires .....	3
<b>3. Le module Bluetooth .....</b>	<b>4</b>
3.1. Configuration .....	4
3.2. Témoins lumineux.....	4
<b>4. Tableau d'affectation des Entrées/Sorties PICAXE .....</b>	<b>5</b>
<b>5. Mise en service des applications.....</b>	<b>6</b>
5.1. Programmes pour smartphone Android .....	6
<b>6. Fiche technique N°1 – RC_B1_Avancer .....</b>	<b>8</b>
<b>7. Fiche technique N°2 – RC_B2_Pilotage .....</b>	<b>11</b>
<b>8. Fiche technique N°3 – RC_B3_Accelerometre.....</b>	<b>14</b>
<b>9. Fiche technique N°4 – RC_B4_Reception_donnee .....</b>	<b>17</b>
<b>10. Fiche technique N°2 – RC_B5_Choix_pilotage.....</b>	<b>20</b>

# 1. Introduction

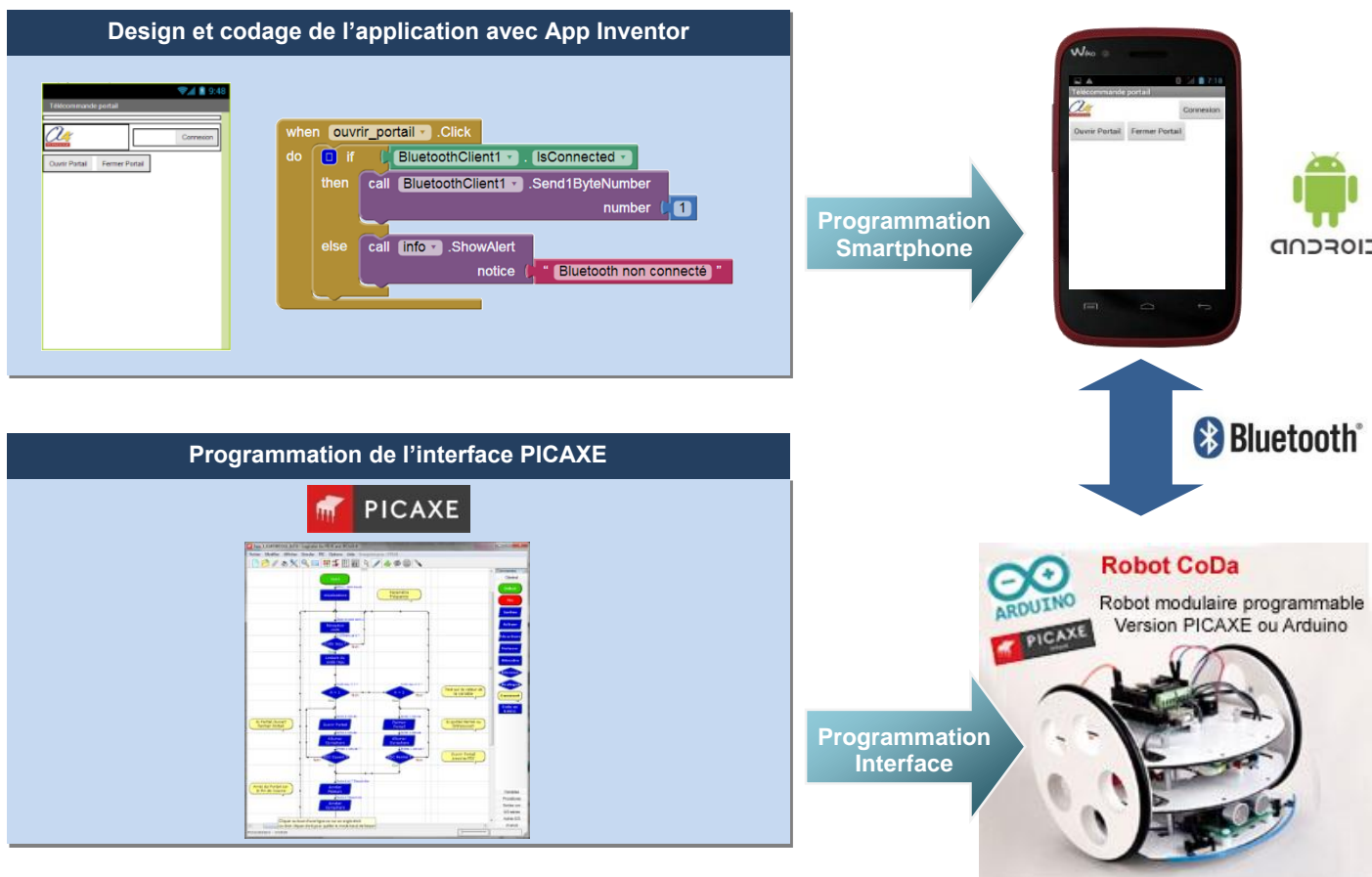
Ce document illustre l'utilisation du module Bluetooth au travers d'exemples basés sur **le robot CoDa d'A4** équipée avec **l'option module Bluetooth**.

Les applications proposées permettent d'interagir entre une application chargée sur **Smartphone Android** et la maquette pilotée par une interface programmable **PICAXE AXE401**

Les programmes sont développés avec les IDE\* :

- **App Inventor 2** pour les applications Android ;
- **PICAXE Editor 6** pour les applications robotique ;

\* IDE : Environnement de Développement Intégré



## 1.1. Prérequis souhaitables

La mise en œuvre des applications suppose que l'utilisateur ait des notions de base autour des logiciels et matériels utilisés. Il est utile de maîtriser les programmes de bases du robot CoDa avant de s'initier avec l'option Bluetooth

## 1.2. Organisation de ce dossier

Les applications se présentent sous forme de fiches classées par ordre de difficulté croissante.

Elles permettent de découvrir la manière d'établir une communication bidirectionnelle entre le smartphone et l'interface de pilotage du robot.

Des propositions de modifications sont suggérées à la fin de chaque fiche.

Les éléments ou briques de programmes qui permettent d'assurer la communication en Bluetooth entre le smartphone et l'interface sont similaires d'une application à l'autre.

Ces briques peuvent apparaître complexes, cependant leur maîtrise n'est pas indispensable pour appréhender les applications.

L'annexe de ce dossier propose des explications complémentaires autour des applications.

## 2. Eléments nécessaires

---

### 2.1. Matériels

- **Robot CoDa** monté avec la carte **PICAXE AXE401**
- **Option Bluetooth** montée sur robot CoDa
- **Câble de programmation USB** pour la programmation de la carte PICAXE.

### 2.2. Logiciels

- IDE PICAXE Logicator ou PICAXE Editor6 (<http://www.picaxe.com/Software>) pour la programmation de l'interface AutoProg.
- IDE App Inventor 2 (<http://appinventor.mit.edu/explore/>) pour la programmation des applications pour smartphone.

L'environnement fonctionne sur le cloud. Il est hébergé sur un serveur.

Un compte Gmail est nécessaire pour l'utiliser.

*\* IDE : Environnement de Développement Intégré*

### 2.3. Ressources complémentaires

Des ressources complémentaires sont disponibles sur [www.a4.fr](http://www.a4.fr)

- Dossier Technique Robot CoDa
- Dossier App Inventor 2 pour prendre en main AppInventor 2
- Guide d'utilisation PICAXE Logicator

D'autres ressources sont également disponibles sur internet. Vous pouvez entrer les mots clés suivants pour les localiser avec un moteur de recherche :

Tuto app inventor 2

Tuto picaxe

Tuto arduino uno

### 3. Le module Bluetooth

Le module Bluetooth développé par A4 Technologie permet de convertir le protocole Bluetooth en protocole de communication type Série qui est le mode de communication classique utilisé avec PICAXE ou Arduino. Ce module accepte différentes configurations.

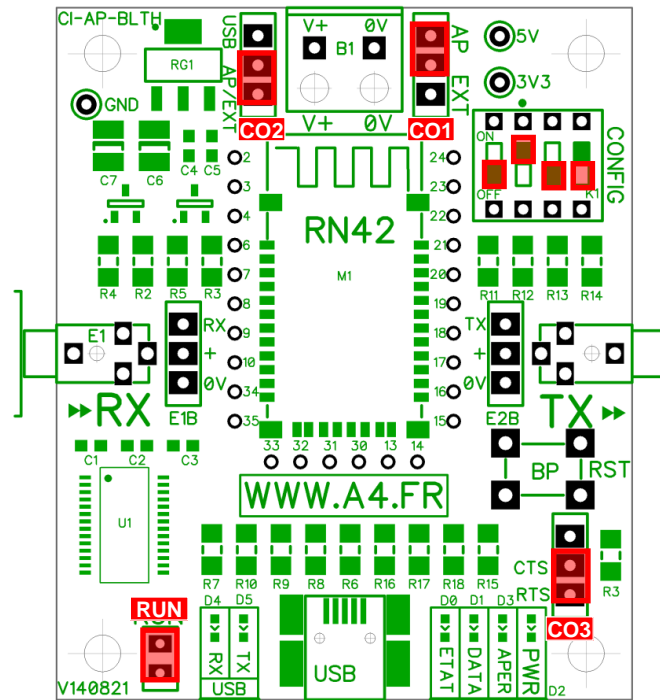
En mode avancé, il peut être configuré au travers d'une liaison par connexion USB à un PC ou par l'envoi de commandes au travers de ses liaisons RX et TX.

La documentation technique du module Bluetooth décrit en détail les fonctionnalités du module.

Elle est téléchargeable sur [www.a4.fr](http://www.a4.fr)

#### 3.1. Configuration

Positionner les cavaliers et interrupteurs comme indiqué par les positions repérées en rouge ci-dessous.



- Le cavalier repéré **RUN** est utilisé lors de la mise au point de programmes avec **Arduino**. Il doit être ôté pour permettre le téléversement du programme puis doit être remis lors de l'utilisation.
- La mise au point de programmes avec **PICAXE** ne nécessite pas d'ôter ce cavalier pour transférer le programme.
- Les cavaliers **CO1** et **CO2** permettent de sélectionner le mode d'alimentation du module Bluetooth. Dans la configuration ci-dessus, son alimentation provient directement de l'interface AutoProg ou AutoProgUno au travers des cordons de liaison avec le module ; ils sont positionnés respectivement sur AP et sur AP/EXT.
- Le cavalier **CO3** est utilisé en mode avancé pour relier ou dissocier les signaux CTS et RTS nécessaires au fonctionnement du module Bluetooth. Ici, il est positionné sur CTS/RTS.
- Les interrupteurs **CONFIG** permettent de paramétrer le mode de fonctionnement du module Bluetooth. Ici, l'interrupteur n°2 est positionné sur ON pour sélectionner une vitesse de transmission des données à 9600 bauds.

#### 3.2. Témoins lumineux

**PWR** indique que le module est sous tension.

**APER** indique que le module est associé avec un matériel Bluetooth.

**DATA** indique qu'il y a un flux de données entre le module et l'appareil avec lequel il est connecté.

**ETAT** indique que le module est opérationnel. L'affichage clignotant indique qu'il n'est pas opérationnel.

**USB RX** indique qu'il y a un flux de données sur la liaison USB du PC vers le module.

**USB TX** indique qu'il y a un flux de données sur la liaison USB du module vers le PC.

## 4. Tableau d'affectation des Entrées/Sorties PICAXE

Ci-dessus le tableau résumant la connexion entre les broches de la carte Shield AXE408 et la carte de commande PICAXE AXE401. Ce tableau sert principalement à faire le lien entre le câblage des capteurs/actionneurs et la programmation du microcontrôleur.

N° Broche AXE 408 (Shield moteur)	MODULE CAPTEUR/ACTIONNEURS	N° Broche PICAXE (Carte de commande)
0 (RX)	Module Bluetooth (TX)	C.7
1 (TX)	Module Bluetooth (RX)	C.6
2	Détecteur de ligne Droit	B.2
3	Détecteur de ligne Central	B.0
4	Détecteur de ligne Gauche	B.1
5	Module capteur à ultrasons	B.5
6	Broche libre	B.6
7	Broche libre	B.7
8	Direction moteur droit	C.0
9	Puissance moteur droit	C.1
10	Puissance moteur gauche	C.2
11	Direction moteur gauche	C.5
12	Broche libre	C.4
13	Broche libre	C.3
A0	Broche libre	A.0
A1	Broche libre	A.1
A2	Broche libre	A.2
A3	Broche libre	A.3
A4	Microrupteur gauche	B.3
A5	Microrupteur droit	B.4

## 5. Mise en service des applications

Les applications proposées dans les fiches suivantes sont créées avec différents IDE (Environnement de Développement Intégré).

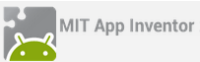
Pour chaque application, un ensemble de fichiers à charger dans le smartphone Android et dans l'interface programmable PICAXE est proposé.


Une description et des explications sont proposées pour chaque application.

Les éléments clés sont mis en évidence et des suggestions de modification sont proposées en vue d'adapter les programmes à un nouveau contexte d'utilisation.

Il est indispensable que la fonction Bluetooth soit mise en service dans les paramètres du smartphone ou de la tablette Android.


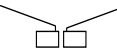


### 5.1. Programmes pour smartphone Android

	<a href="http://appinventor.mit.edu/explore/">http://appinventor.mit.edu/explore/</a> Cet IDE est gratuit. Il fonctionne sur le cloud. Il ne nécessite pas d'installation en local. Un compte Gmail est indispensable pour accéder à son espace de développement.
<b>Matériel associé</b>	<ul style="list-style-type: none"><li>- Smartphone ou tablette Android</li><li>- Cordon de liaison USB avec le PC pour transférer les programmes dans le smartphone.</li></ul>
<b>Fichiers</b>	Fichiers <b>.APK</b> (application) à copier et à installer dans le smartphone. Fichiers <b>.AIA</b> (code source) à ouvrir dans App Inventor 2.
<b>Notes</b>	Un guide de prise en main de l'IDE App Inventor 2 est proposé en téléchargement sur <a href="http://www.a4.fr">www.a4.fr</a> La page d'accueil d'App Inventor propose un nombre important de tutoriels (en anglais). Le mot-clé « Tuto App Inventor » tapé dans un moteur de recherche renvoie vers de nombreuses ressources autour d'App Inventor.

	<b>PICAXE Logicator</b> <a href="http://www.picaxe.com/Software/PICAXE/Logicator-for-PICAXE/">http://www.picaxe.com/Software/PICAXE/Logicator-for-PICAXE/</a> ou <b>PICAXE Editor6</b> <a href="http://www.picaxe.com/Software/PICAXE/PICAXE-Editor-6/">http://www.picaxe.com/Software/PICAXE/PICAXE-Editor-6/</a> Cet IDE est gratuit. Il doit être installé sur un PC.
<b>Matériel associé</b>	<ul style="list-style-type: none"><li>- Interface programmable AutoProg ou autre carte PICAXE compatible.</li><li>- Cordon de programmation PICAXE USB pour transférer les programmes dans l'interface AutoProg.</li></ul>
<b>Fichiers</b>	Fichiers <b>.PLF</b> à ouvrir dans l'IDE PICAXE et à transférer dans l'interface.
<b>Notes</b>	<a href="http://www.a4.fr">www.a4.fr</a> propose de nombreuses ressources autour de PICAXE ; notamment le Guide d'utilisation de Logicator et les dossiers AutoProg Il existe d'autres IDE compatibles PICAXE (par exemple Scratch).






### Liste des symboles utilisés pour les capteurs:

(Voir le chapitre Montage / Assemblage pour la mise en œuvre des capteurs)

Symbole	Description
	Aucun capteur
	Module capteurs microrupteurs (détection de contact avec un obstacle).
	Module capteurs infrarouges (détection de marquage au sol).
	Module capteur à ultrasons (détection d'obstacle à distance).



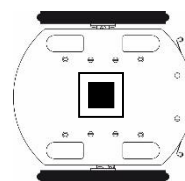
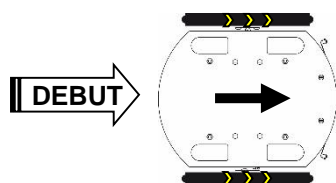
## Liste des programmes

Fiche	Capteur	Description	Fichier programme
N°1		Envoyer une instruction du Smartphone au robot	<a href="#">RC_B1_Avancer</a>
N°2		Piloter le robot avec le Smartphone	<a href="#">RC_B2_Pilotage</a>
N°3		Piloter le robot avec l'accéléromètre du Smartphone	<a href="#">RC_B3_Accelerometre</a>
N°4		Recevoir une donnée émise par le module Bluetooth	<a href="#">RC_B4_Reception_donnee</a>
N°5		Etablir un mode de fonctionnement manuel et automatique	<a href="#">RC_B5_Choix_pilotage</a>

## 6. Fiche technique N°1 – RC\_B1\_Avancer

But de l'application	Se déplacer en avant et s'arrêter avec les commandes du Smartphone
Notions de programmation abordées	Pilotage du robot sans fil par connexion Bluetooth à un appareil Android
Programme associé	Code source PICAXE Editor 6 <b>RC_B1_Avancer.plf</b> Code source App Inventor <b>RC_B1_Avancer.aia</b> Application Android <b>RC_B1_Avancer.apk</b> (Fichiers téléchargeables sur <a href="http://www.a4.fr">www.a4.fr</a> )
Capteur mis en jeu	Aucun

Illustration



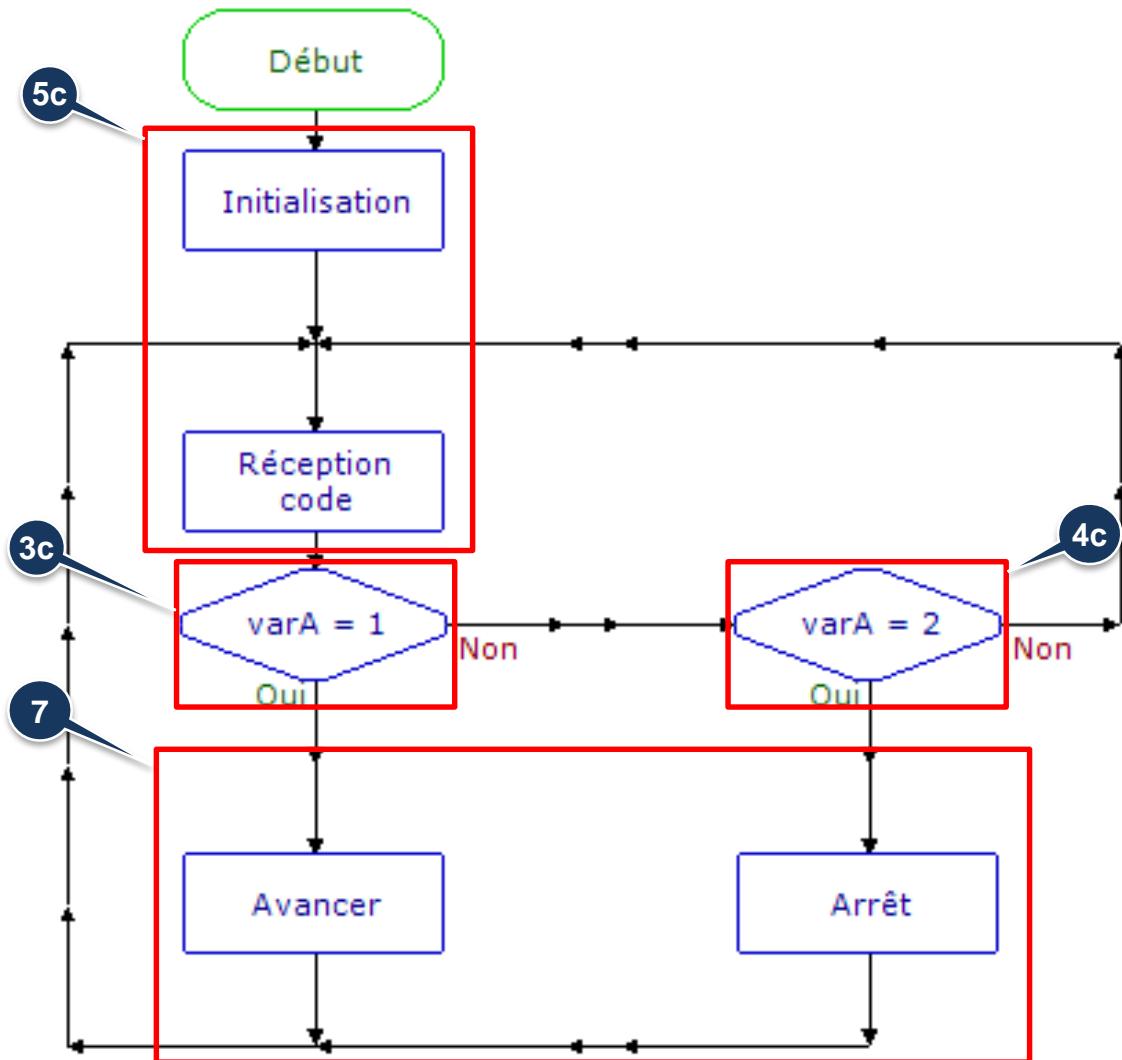
Suggestions de modifications	Modifier le programme pour que le robot avance lorsque le bouton avancer est appuyé et s'arrête au relâchement
------------------------------	--

**IDE App Inventor2 (fenêtre Designer)**

- 1 Titre de l'application.
- 2 Image insérée dans l'application.
- 3a Avance du robot en envoyant le code « 1 ».
- 4a Arrêt du robot en envoyant le code « 2 ».
- 5a Gestion de la communication en Bluetooth.
- 6 Éléments non visibles de l'application. Des explications sont données en Annexe.

**IDE App Inventor2 (fenêtre Blocks)**

- 3b **Déclenchement de la commande avancer**  
Lorsque bouton nommé **avancer** est cliqué :  
Le smartphone envoie le code « 1 » par liaison Bluetooth.
- 4b **Déclenchement de la commande arret**  
Lorsque bouton nommé **arret** est cliqué :  
Le smartphone envoie le code « 2 » par liaison Bluetooth.
- 5b **Gestion de la communication en Bluetooth**  
Ces blocs sont nécessaires pour gérer la communication en Bluetooth. Ils sont communs à toutes les applications proposées dans ce dossier.  
  
*Note : on peut déployer la visualisation de ces blocs dans App Inventor 2 en effectuant un clic droit sur le bloc souhaité et en sélectionnant « Expand Block ».*



3c

Vérification du code reçu (code « 1 »).

4c

Vérification du code reçu (code « 2 »).

5c

Gestion et attente de réception des données émises par le smartphone.

7

Gestion de la marche et l'arrêt du robot.

## 7. Fiche technique N°2 – RC\_B2\_Pilotage

---

But de l'application	<b>Piloter le robot dans toutes les directions. Le robot doit se déplacer seulement sur l'appui du bouton et doit s'arrêter au relâchement.</b>
Notions de programmation abordées	Contrôler la direction du Robot, nouvelle instruction sur le déclenchement d'une séquence App Inventor
Programme associé	Code source PICAXE Editor 6 <b>RC_B2_Pilotage.plf</b> Code source App Inventor <b>RC_B2_Pilotage.aia</b> Application Android <b>RC_B2_Pilotage.apk</b> <b>(Fichiers téléchargeables sur <a href="http://www.a4.fr">www.a4.fr</a>)</b>
Capteur mis en jeu	Aucun
Suggestions de modifications	Ajouter une possibilité de régler la vitesse des moteurs directement sur l'interface de l'application.

**1** Titre de l'application.

**2** Image insérée dans l'application.

**3a** Avance du robot lors de l'appui du bouton en envoyant le code « 1 ».

**4a** Arrêt du robot lors du relâchement du bouton en envoyant le code « 0 ».

**5a** Gestion de la communication en Bluetooth.

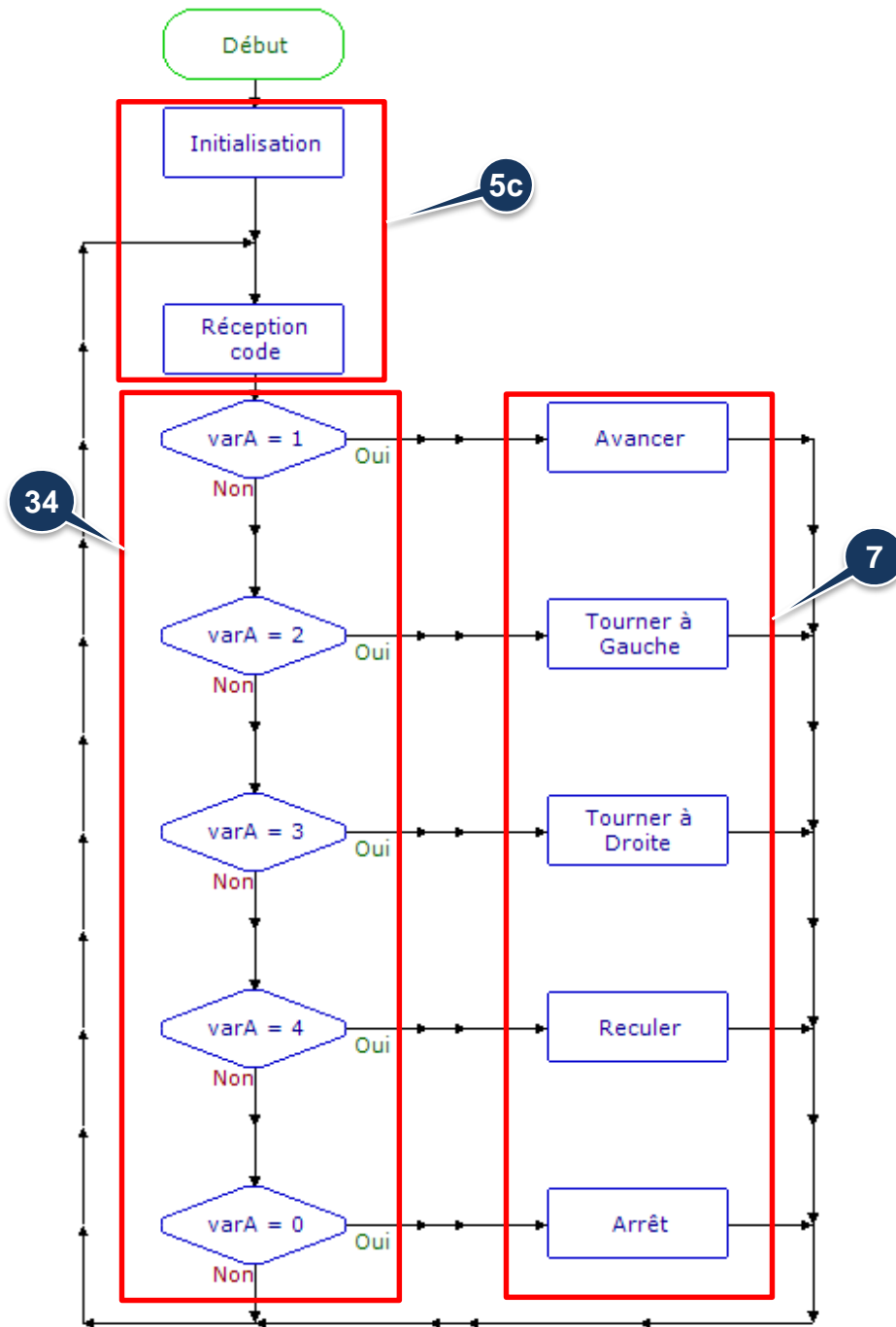
**6** Éléments non visibles de l'application. Des explications sont données en Annexe.

**3b** Avancer :  
**Lorsque le bouton nommé avancer est enfoncé :**  
 Envoyer le code « 1 » par liaison Bluetooth.  
 Cette instruction est répétée pour les 3 autres commandes de pilotage du robot avec un code envoyé différent pour chaque instruction.

**4b** Arrêt du robot :  
**Lorsque le bouton nommé avancer est relâché**  
 envoyer le code « 0 » par liaison Bluetooth.  
 Cette instruction est répétée pour les 3 autres commandes de pilotage avec le code « 0 »

**5b** Gestion de la communication en Bluetooth  
 Ces blocs sont nécessaires pour gérer la communication en Bluetooth.  
 Ils sont communs à toutes les applications proposées dans ce dossier.

*Note : on peut déployer la visualisation de ces blocs dans App Inventor 2 en effectuant un clic droit sur le bloc souhaité et en sélectionnant « Expand Block ».*



34

Vérification du code reçu par le module Bluetooth et transmis par le smartphone

5c

Gestion et attente de réception des données émises par le smartphone.

7

Gestion du pilotage du robot

## 8. Fiche technique N°3 – RC\_B3\_Accelerometre

---

But de l'application	<b>Piloter le robot dans toutes les directions en fonction de l'inclinaison de l'appareil</b>
Notions de programmation abordées	Contrôler la direction du Robot à l'aide du capteur Accéléromètre
Programme associé	Code source PICAXE Editor 6 <b>RC_B3_Accelerometre.plf</b> Code source App Inventor <b>RC_B3_Accelerometre.aia</b> Application Android <b>RC_B3_Accelerometre.apk</b> <b>(Fichiers téléchargeables sur <a href="http://www.a4.fr">www.a4.fr</a>)</b>
Capteur mis en jeu	Aucun
Suggestions de modifications	Modifier la vitesse du robot selon l'inclinaison de l'appareil Android.



**IDE App Inventor2 (fenêtre Designer)**

- 1 Titre de l'application.
- 2 Image insérée dans l'application.
- 3a Variable d'inclinaison de l'appareil. X, Y et Z afficheront l'inclinaison de chaque axe de l'appareil.
- 4a Capteur accéléromètre
- 5a Gestion de la communication en Bluetooth.
- 6 Éléments non visibles de l'application. Des explications sont données en Annexe.

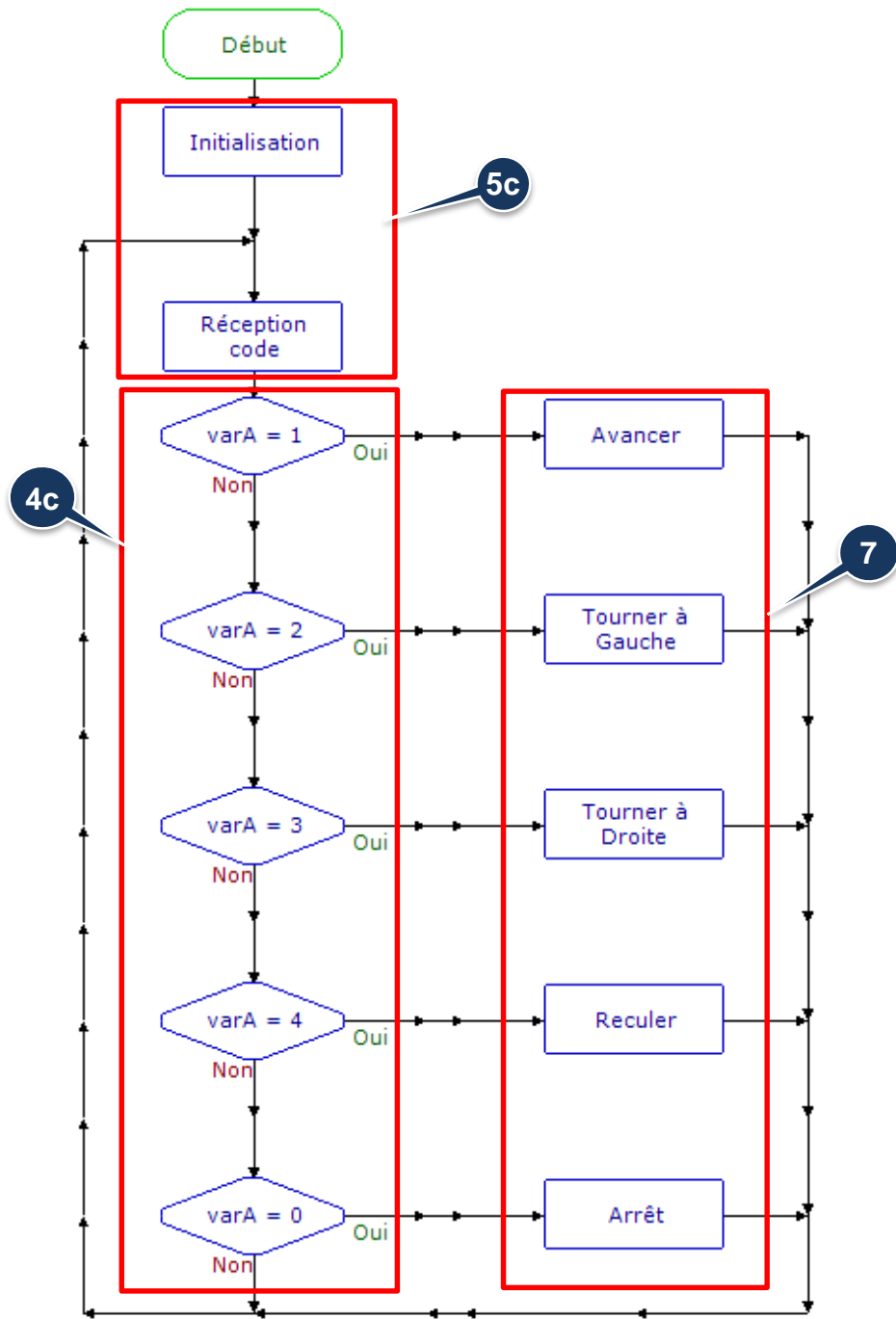
**IDE App Inventor2 (fenêtre Blocks)**

**3b Affichage des variables :**  
Lorsque l'inclinaison de l'appareil change, les variables X, Y et Z prennent les valeurs d'inclinaison.

**4b Envoi du code :**  
Dans cette partie le code est envoyé en permanence, sans action nécessaire sur un bouton.  
Lorsque les valeurs de l'accéléromètre changent, l'appareil Android envoie automatiquement un code au module Bluetooth


**5b Gestion de la communication en Bluetooth**  
Ces blocs sont nécessaires pour gérer la communication en Bluetooth. Ils sont communs à toutes les applications proposées dans ce dossier.

*Note : on peut déployer la visualisation de ces blocs dans App Inventor 2 en effectuant un clic droit sur le bloc souhaité et en sélectionnant « Expand Block ».*



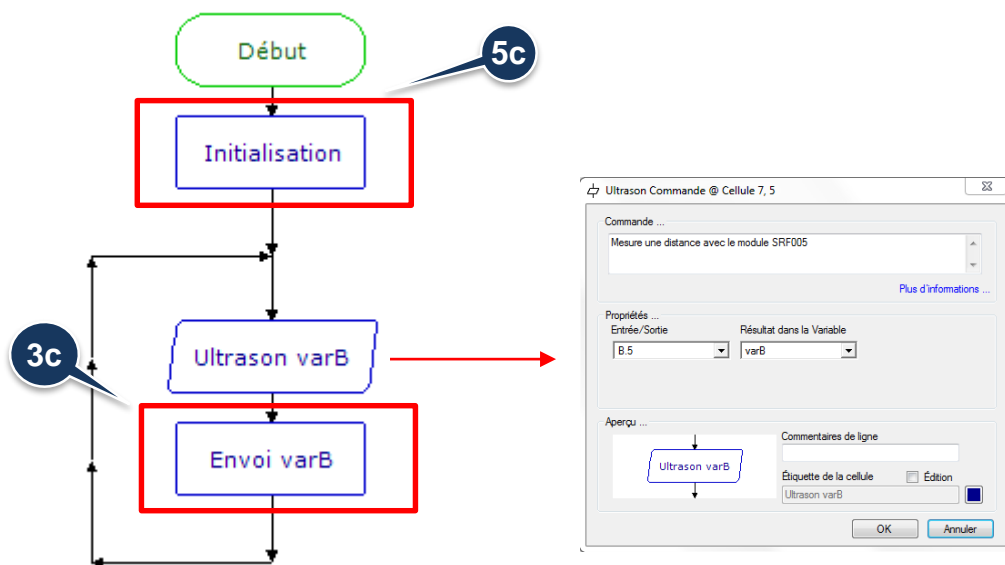
- 4c** Vérification du code reçu par le module Bluetooth et transmis par le smartphone
- 5c** Gestion et attente de réception des données émises par le smartphone.
- 7** Gestion du pilotage du robot

## 9. Fiche technique N°4 – RC\_B4\_Reception\_donnee

But de l'application	Réceptionner des données reçues par les capteurs du robot.
Notions de programmation abordées	Emission de données sur PICAXE, réception de données sur App Inventor.
Programme associé	Code source PICAXE Editor 6 <b>RC_B4_Reception_donnee.plf</b> Code source App Inventor <b>RC_B4_Reception_donnee.aia</b> Application Android <b>RC_B4_Reception_donnee.apk</b> (Fichiers téléchargeables sur <a href="http://www.a4.fr">www.a4.fr</a> )
Capteur mis en jeu	 Capteur à Ultrasons
Suggestions de modifications	Afficher un texte sur l'application lorsque la distance reçue est inférieure à 10 cm.

- 1 Titre de l'application.
- 2 Image insérée dans l'application.
- 3a Variable qui stockera la valeur envoyée par le module Bluetooth.
- 4a Insertion d'une nouvelle horloge pour vérifier la réception de donnée sur l'appareil.
- 5a Gestion de la communication en Bluetooth.
- 6 Éléments non visibles de l'application. Des explications sont données en Annexe.

- 3b Affichage de la donnée  
La donnée reçue est stockée et est affichée dans la variable **distance\_US**.
- 4b Initialisation d'une nouvelle horloge  
**A chaque coup d'horloge**, si le Bluetooth est connecté, l'appareil scrute pour voir si une donnée a été émise par le module Bluetooth.
- 5b Gestion de la communication en Bluetooth  
Ces blocs sont nécessaires pour gérer la communication en Bluetooth. Ils sont communs à toutes les applications proposées dans ce dossier.  
  
*Note : on peut déployer la visualisation de ces blocs dans App Inventor 2 en effectuant un clic droit sur le bloc souhaité et en sélectionnant « Expand Block ».*



Configuration de la commande Ultrason


3c

Envoi la valeur de la variable B à l'appareil Android

5c

Initialisation du PICAXE pour l'envoi des données

## 10. Fiche technique N°2 – RC\_B5\_Choix\_pilotage

But de l'application	<b>Fusionner deux modes de pilotage :</b> <ul style="list-style-type: none"><li>- <b>Un pilotage Automatique</b> qui entrainera le robot à suivre une ligne.</li><li>- <b>Un pilotage manuel</b>, où l'utilisateur reprend le contrôle du robot. Inclure une sécurité. Stopper le robot lorsque le capteur à Ultrasons détecte un obstacle.</li></ul>
Notions de programmation abordées	Rendre compatible plusieurs modes de pilotages. Gérer l'envoi et la réception de donnée en simultanée
Programme associé	Code source PICAXE Editor 6 <b>RC_B5_Choix_pilotage.plf</b> Code source App Inventor <b>RC_B5_Choix_pilotage.aia</b> Application Android <b>RC_B5_Choix_pilotage.apk</b> (Fichiers téléchargeables sur <a href="http://www.a4.fr">www.a4.fr</a> )
Capteur mis en jeu	 Capteur à Ultrasons

- 1 Titre de l'application.
- 2a Application RC\_B2\_Pilotage.
- 3a Ajout de deux boutons qui détermineront le choix du pilotage.
- 4a Application RC\_B4\_Reception\_donnee.
- 5a Gestion de la communication en Bluetooth.
- 6 Éléments non visibles de l'application. Des explications sont données en Annexe.

**Choix du mode de pilotage :**

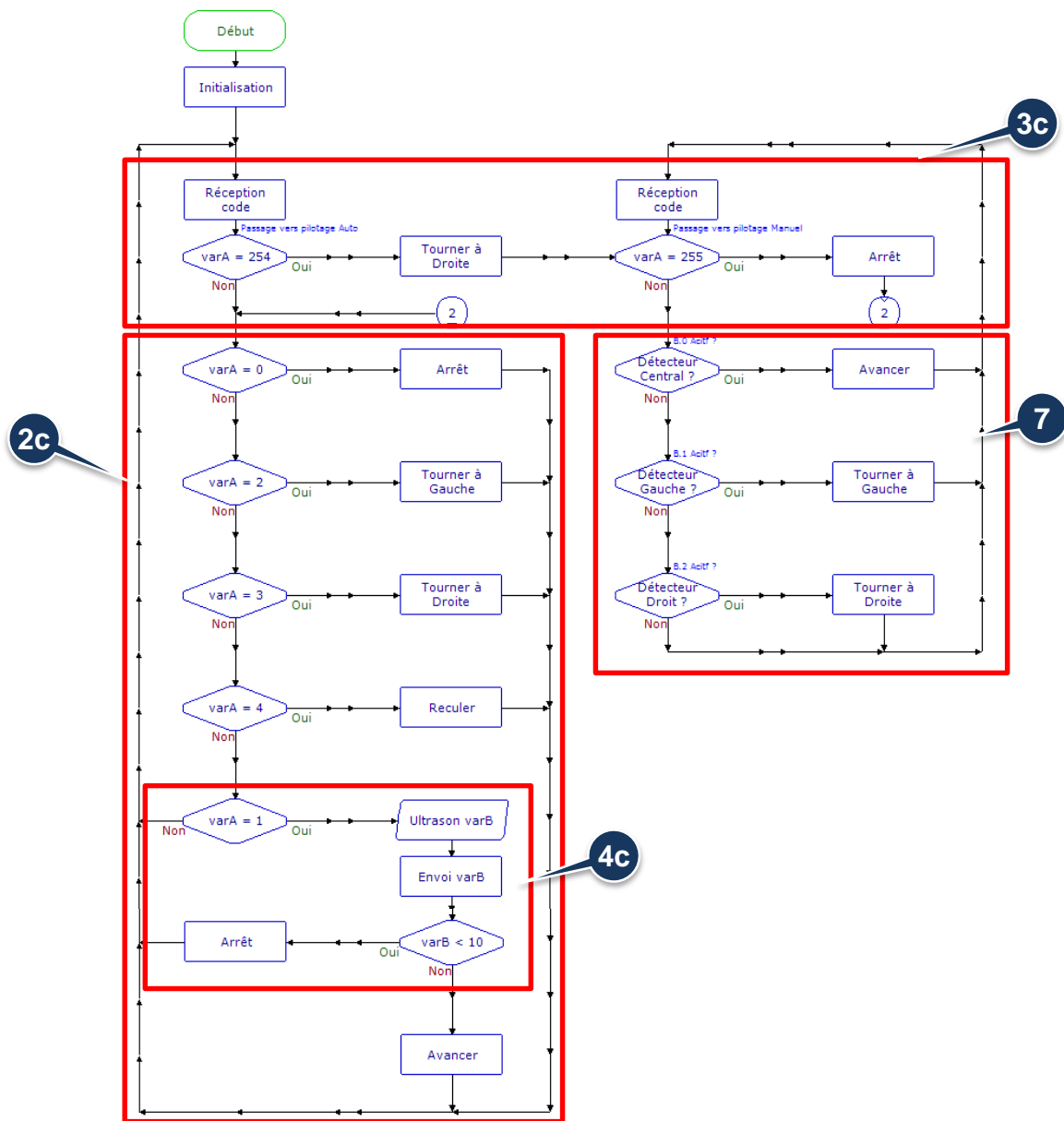
Lorsque le bouton pilotage\_auto est appuyé, le Smartphone envoie le code « 254 ».

Lorsque le bouton pilotage\_manuel est appuyé, le Smartphone envoie le code « 255 ».

**Gestion de la communication en Bluetooth**

Ces blocs sont nécessaires pour gérer la communication en Bluetooth. Ils sont communs à toutes les applications proposées dans ce dossier.

*Note : on peut déployer la visualisation de ces blocs dans App Inventor 2 en effectuant un clic droit sur le bloc souhaité et en sélectionnant « Expand Block ».*



- 2c Programme du pilotage manuel.
- 3c Gestion du pilotage automatique ou manuel.
- 4c Gestion de la sécurité Ultrason.
- 7 Pilotage automatique (suivi de ligne).