

Apprendre à programmer avec Scratch



Ce manuel introduit à l'environnement du logiciel Studuino. L'environnement de programmation Studuino étant en développement, les présentes instructions sont amenées à être modifiées ou révisées si nécessaire.

Sommaire

1. Introduction	1
1.1. Configuration.....	1
1.2. Connecter les éléments.....	1
2. Utiliser les LED	4
2.1. Connecter une LED au circuit Studuino	4
2.2. Paramétrage des ports.....	4
2.3. Créer un programme pour allumer une LED.....	6
2.4. Faire clignoter une LED.....	8
2.5. Spécifier le nombre de clignotement de la LED	10
2.6. Changer la vitesse de clignotement.....	13
3. Utiliser les avertisseurs sonores.....	15
3.1. Connecter un avertisseur sonore.....	15
3.2. Paramétrage des ports.....	15
3.3. Jouer une note	16
3.4. Jouer une mélodie.....	18
3.5. Faire une mélodie en utilisant une liste.....	21
3.5.1. Créer une liste	21
4. Interrupteurs et capteurs	30
4.1. Interrupteurs à bouton-poussoir.....	30
4.1.1. Paramétrage des ports	31
4.1.2. Tester un interrupteur à bouton-poussoir.....	31
4.1.3. Programmer.....	34
4.2. Capteurs de lumière.....	41
4.2.1. Connecter un capteur de lumière au circuit Studuino	41
4.2.2. Paramétrage des ports	41
4.2.3. Vérifier le capteur de lumière.	42
4.2.4. Programmer.....	42
4.3. Capteurs infrarouges à réflecteur	46
4.3.1. Connecter un capteur infrarouge au circuit Studuino.....	46
4.3.2. Paramétrage des ports	46

4.3.3.	Vérifier le capteur infrarouge à réflecteur.....	47
4.3.4.	Programmer.....	48
4.4.	Capteurs tactiles	53
4.4.1.	Connecter un capteur tactile au circuit Studuino	53
4.4.2.	Paramétrage des ports	53
4.4.3.	Tester le capteur tactile	54
4.4.4.	Programmer.....	55
4.5.	Capteurs sonores.....	58
4.5.1.	Connecter un capteur sonore au circuit Studuino.....	58
4.5.2.	Paramétrage des ports.....	58
4.5.3.	Tester le capteur sonore	59
4.5.4.	Programmer.....	59
5.	Moteurs à courant continu.....	65
5.1.	Contrôler un moteur à courant continu	65
5.1.1.	Connecter un moteur à courant continu au circuit Studuino.....	65
5.1.2.	Paramétrage du port.....	66
5.2.	Contrôler une voiture.....	70
5.2.1.	Programmer.....	70
5.3.	Contrôler une voiture en utilisant un accéléromètre	76
5.3.1.	Connecter un accéléromètre au circuit Studuino	77
5.3.2.	Paramétrage des ports	77
5.3.3.	Programmer.....	79
6.	Faire bouger le servomoteur	88
6.1.	Calibrer les angles du servomoteur	88
6.1.1.	Ajuster l'angle de l'arbre de transmission du servomoteur	88
6.1.2.	Connecter un servomoteur au circuit Studuino.....	89
6.1.3.	Paramétrage des ports	89
6.1.4.	Programmer.....	91
6.2.	Faire un bras robotisé qui utilise trois servomoteurs	92
6.2.1.	Assembler le robot.....	93
6.2.2.	Paramétrage des ports	95
6.2.3.	Programmer.....	95

1. Introduction

1.1. Configuration

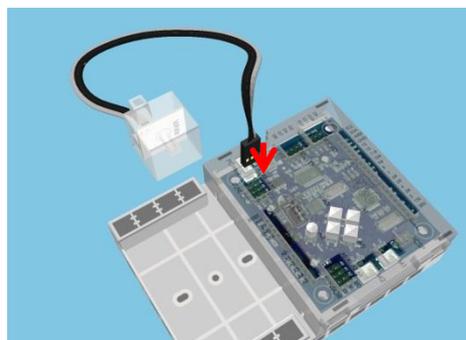
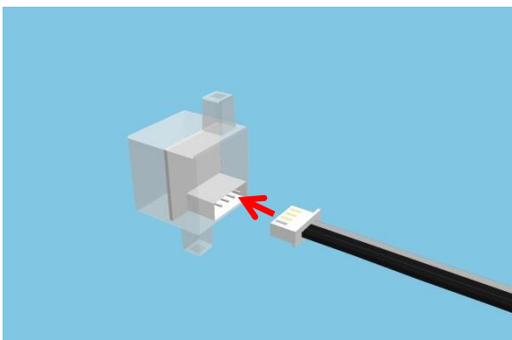
Avant de commencer, rendez-vous sur <https://www.ecolerobots.com/studuino/> pour télécharger et consulter les manuels [Découvrir Scratch](#) et [Découvrir la Programmation par icônes](#).

1.2. Connecter les éléments

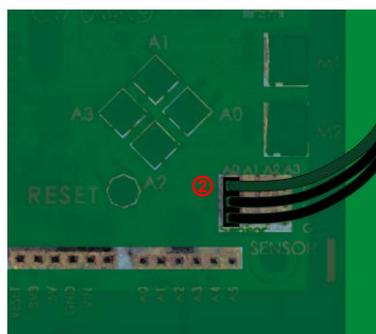
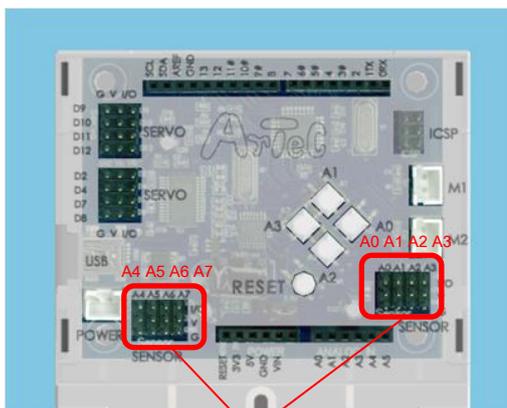
① Brancher les capteurs, avertisseurs sonores et LED

Les câbles de raccordement de capteur sont utilisés pour lier les capteurs au circuit Studuino. Les câbles se fixent aux connecteurs du capteur/ LED/ avertisseur sonore situés sur le circuit comme indiqué ci-après.

★ Le câble à trois fils est utilisé pour tous les capteurs à l'exception de l'accéléromètre qui requiert un câble à quatre fils.



Fils gris face intérieure comme indiqué.



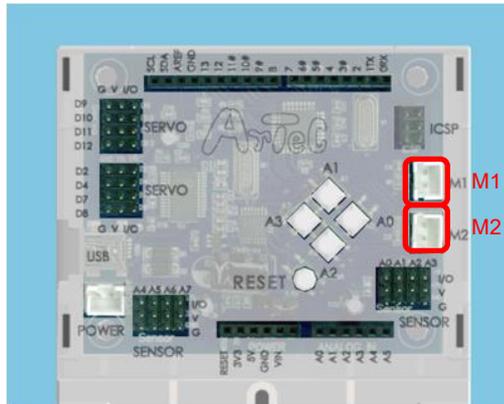
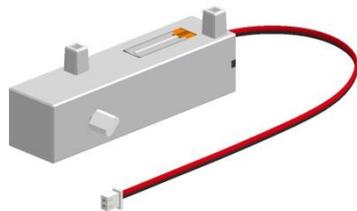
Connecteurs des capteurs/ LED/ avertisseurs sonores

- ★ Les capteurs son, lumière et photorélecteur IR se connectent de A0 jusqu'à A7.
- ★ Les capteurs tactiles, les LED et les avertisseurs sonores se connectent de A0 jusqu'à A5.
- ★ Les accéléromètres se connectent à A4 et à A5.
- ★ Les interrupteurs à bouton-poussoir A0-A3 ne peuvent pas être utilisés pendant l'utilisation d'un capteur avec les connecteurs A0-A3.

② Connecter un moteur à courant continu

L'utilisation des connecteurs M1 et M2 permettent jusqu'à deux connexions de moteurs à courant continu.

★ Les câbles de moteur à courant continu ne peuvent être adaptés au connecteur que d'une seule manière.

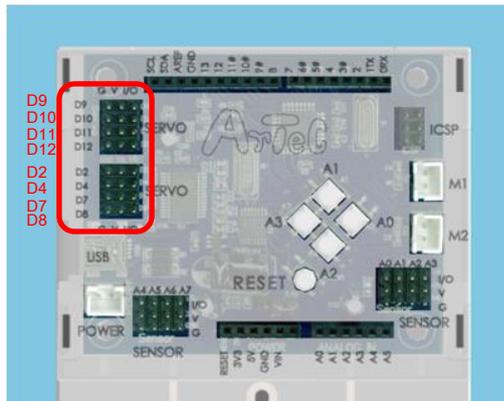
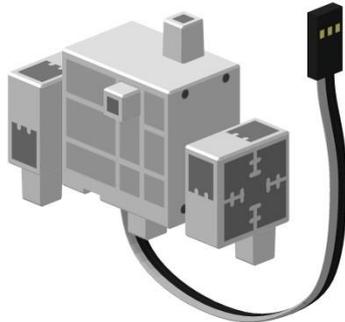


★ Le connecteur de moteur à courant continu M1 ne peut pas être utilisé en même temps que les connecteurs du servomoteur D2 et D4.

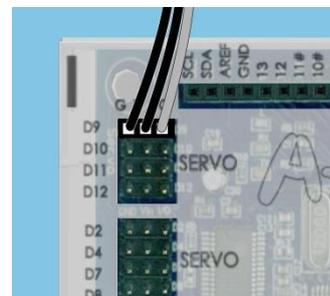
★ Le connecteur de moteur à courant continu M2 ne peut pas être utilisé en même temps que les connecteurs du servomoteur D7 et D8.

③ Connecter un servomoteur

Jusque huit servomoteurs peuvent être connectés aux connecteurs du servomoteur.



Fils gris face intérieure
comme indiqué.

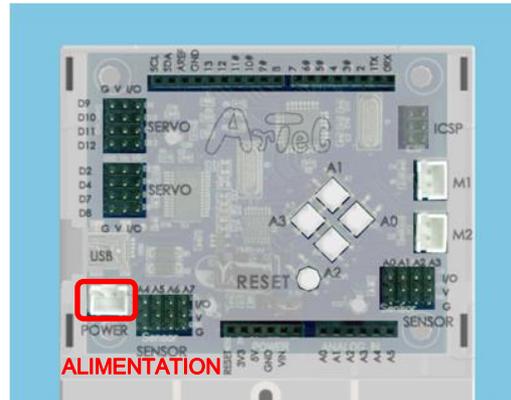


★ Le connecteur de moteur à courant continu M1 ne peut pas être utilisé en même temps que les connecteurs du servomoteur D2 et D4.

★ Le connecteur de moteur à courant continu M2 ne peut pas être utilisé en même temps que les connecteurs du servomoteur D7 et D8.

④ Connecter un boîtier de piles

Connectez le câble du boîtier de pile aux broches d'alimentation. Les capteurs/ LED/ avertisseurs sonores/ interrupteurs peuvent utiliser la tension fournie par le port USB. Toutefois, le moteur à courant continu et le servomoteur nécessitent une tension fournie par une prise de courant. Lorsque l'USB est débranché, tous les dispositifs exigent la tension d'une prise de courant pour fonctionner.

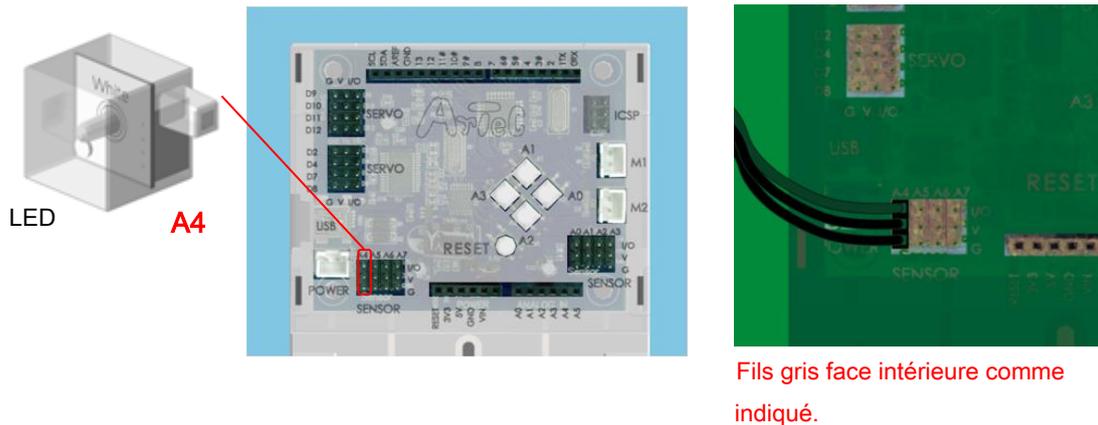


2. Utiliser les LED

Ce chapitre explique comment programmer une LED. Il aborde la connexion des dispositifs au circuit Studuino, le paramétrage des ports, la programmation de l'environnement de programmation par blocs et le transfert des données au circuit.

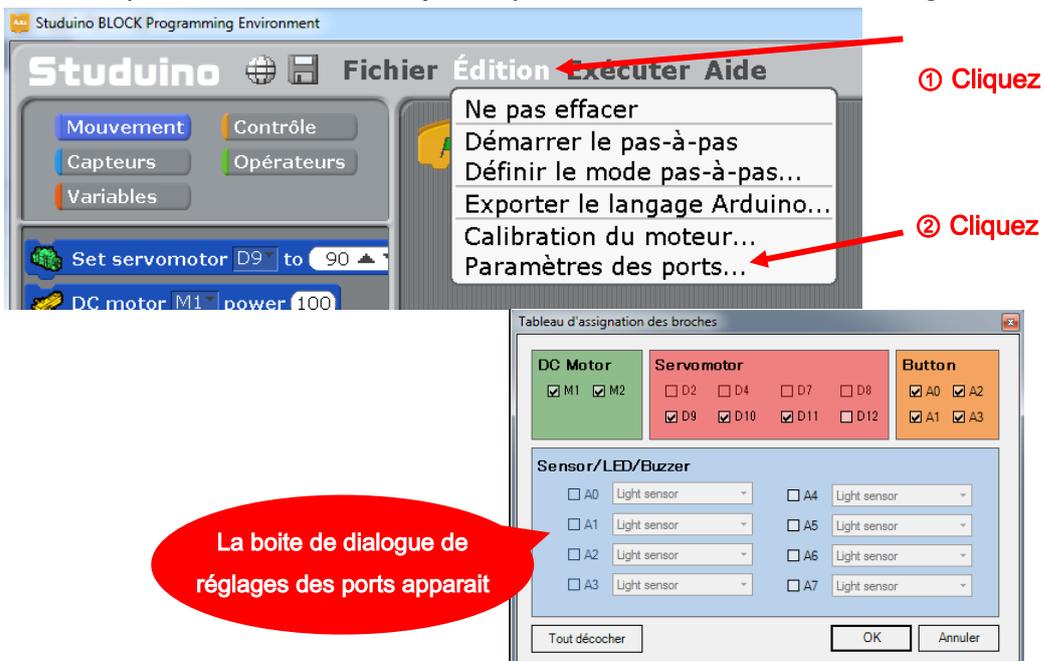
2.1. Connecter une LED au circuit Studuino

Connectez une LED au connecteur A4 (capteur/ LED/ avertisseur sonore) sur le circuit Studuino.

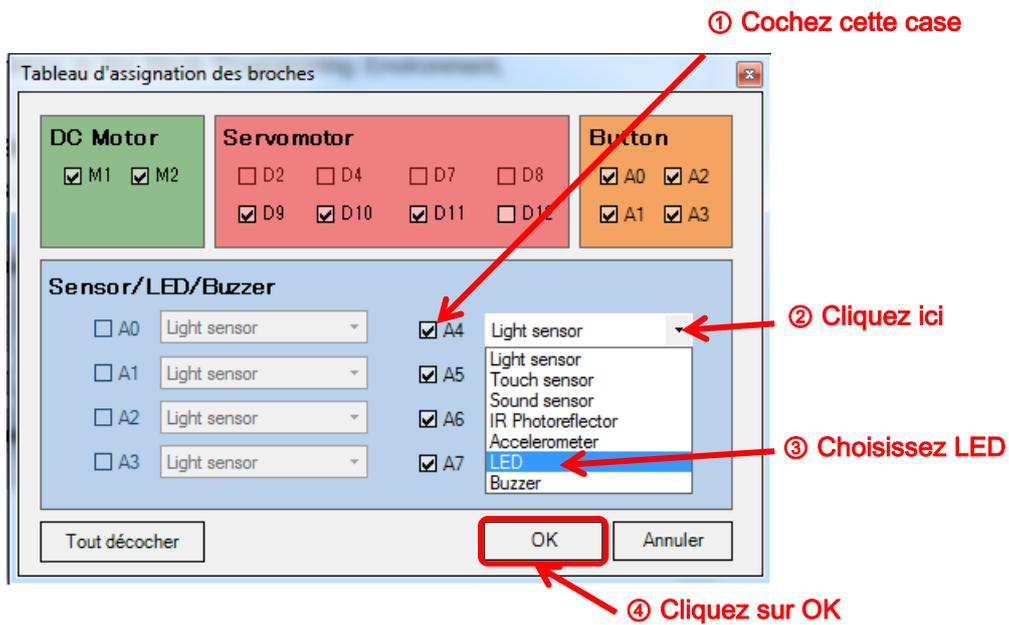


2.2. Paramétrage des ports

Avant de commencer à programmer, définissez les ports dans l'environnement de programmation par blocs pour qu'il s'accorde à votre circuit. Sélectionnez «Édition» dans le menu, puis «Paramètres des ports» pour accéder à la boîte de dialogue de réglages.



Dans la boîte de dialogue de réglage des ports, cochez A4 et choisissez LED.

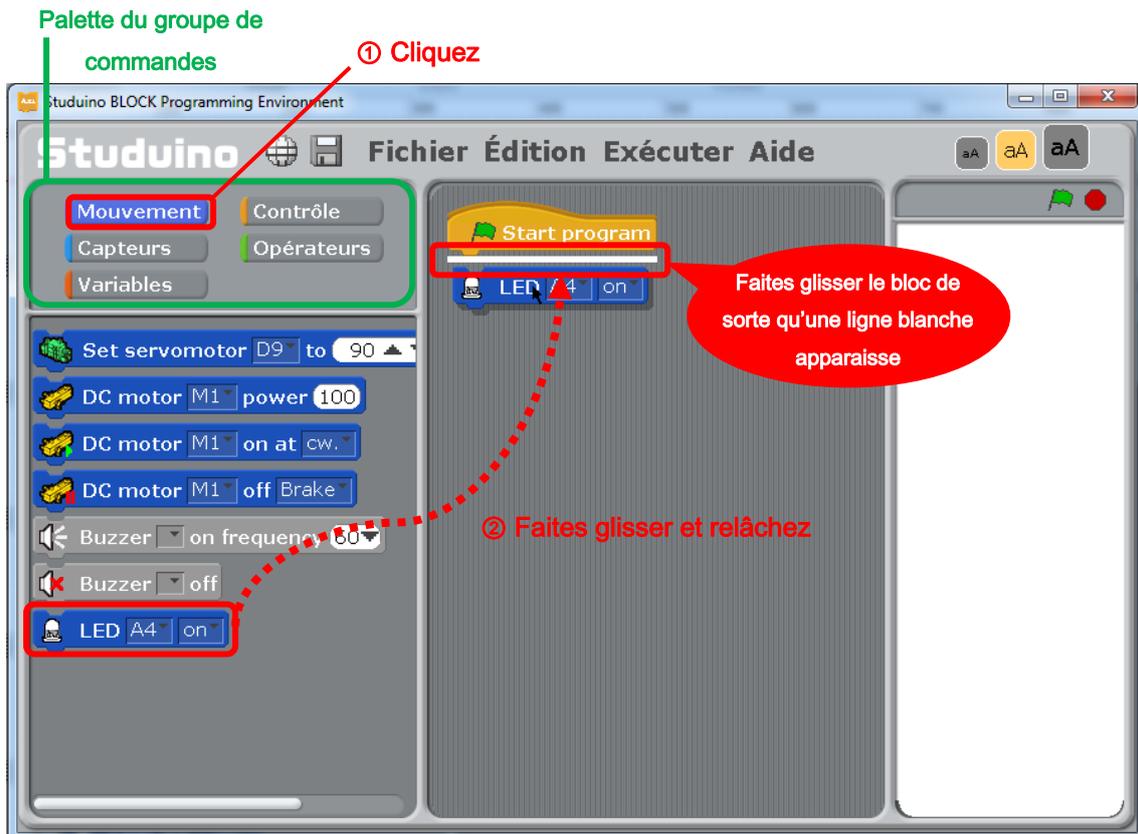


Vous avez maintenant configuré les ports pour votre circuit Studuino dans l'environnement de programmation par blocs.

2.3. Créer un programme pour allumer une LED

Maintenant, nous allons utiliser le bloc  pour apprendre les bases de la programmation.

- ① Cliquez sur l'onglet «Mouvement» dans la palette du groupe de commandes et faites glisser le bloc  situé dans la palette de blocs sous le bloc  situé dans le champ du script. Les blocs vont s'assembler.



Assurez-vous que les blocs sont assemblés convenablement !



- ★ Déplacez les blocs partout à l'intérieur du champ de script en les faisant glisser et en les relâchant. Si le bloc que vous déplacez est bien attaché à un autre bloc, vous les verrez alors bouger ensemble comme un bloc entier.

- ② Connectez le circuit Studuino au PC en utilisant le câble USB. À partir du menu, sélectionnez «Exécuter» puis «Transférer».



L'affichage de la fenêtre de dialogue ci-dessous indique que votre programme est en cours de transfert. La LED s'allume automatiquement une fois que le transfert est effectué.



L'affichage des messages ci-dessous indique que votre programme n'a pas été transféré avec succès. Vérifiez les solutions proposées et essayez à nouveau.

Message	Solution
	Assurez-vous que le circuit Studuino est correctement connecté à votre ordinateur par le câble USB.
	Présence de blocs non attachés. Supprimez tous les blocs qui ne sont pas attachés au bloc .
	Fermez toutes les applications qui peuvent entrer en communication avec votre Studuino. Si aucune autre application ne communique avec le circuit, débranchez le câble USB et appuyez sur le bouton de réinitialisation.

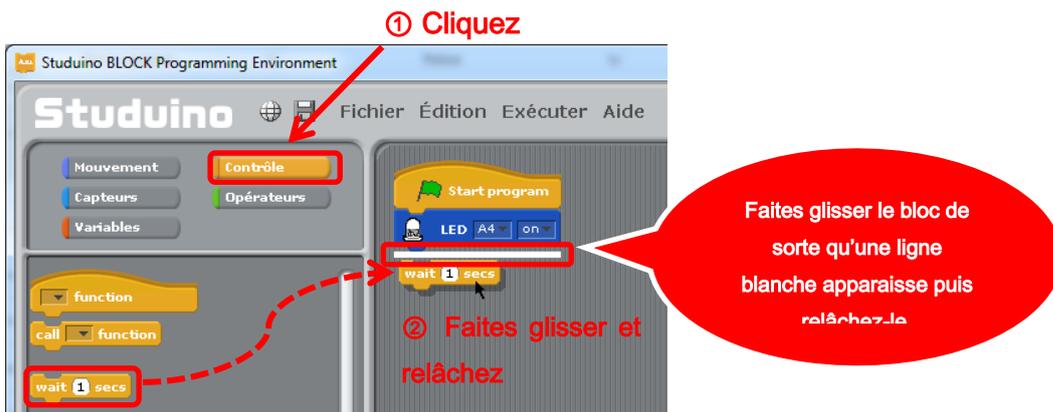
2.4. Faire clignoter une LED

Utilisez le programme LED de la section précédente et un bloc de répétition infinie

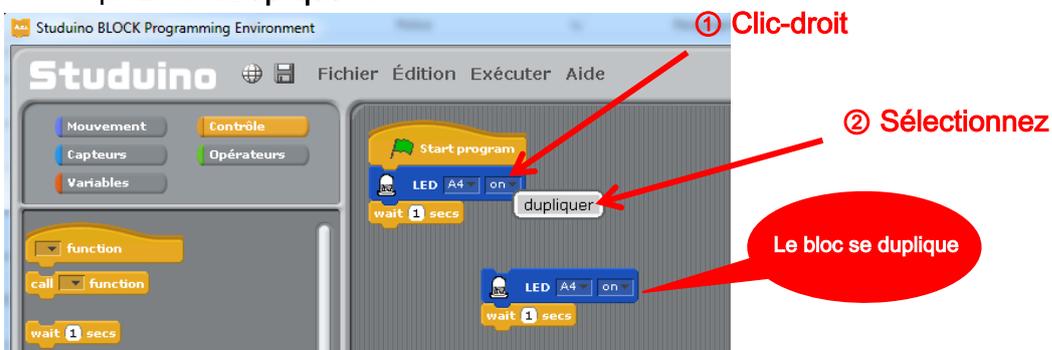


pour apprendre les boucles.

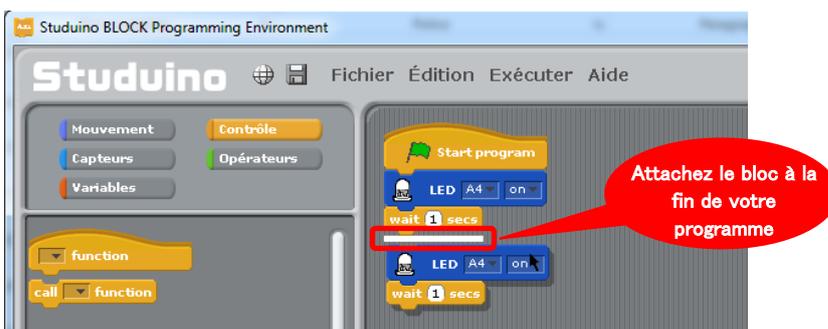
- 1 Cliquez sur l'onglet «Contrôle» dans la palette du groupe de commandes puis faites glisser un bloc **wait 1 secs** situé dans la palette de contrôle vers le champ de script. Attachez-le aux blocs de programme LED clignotante.



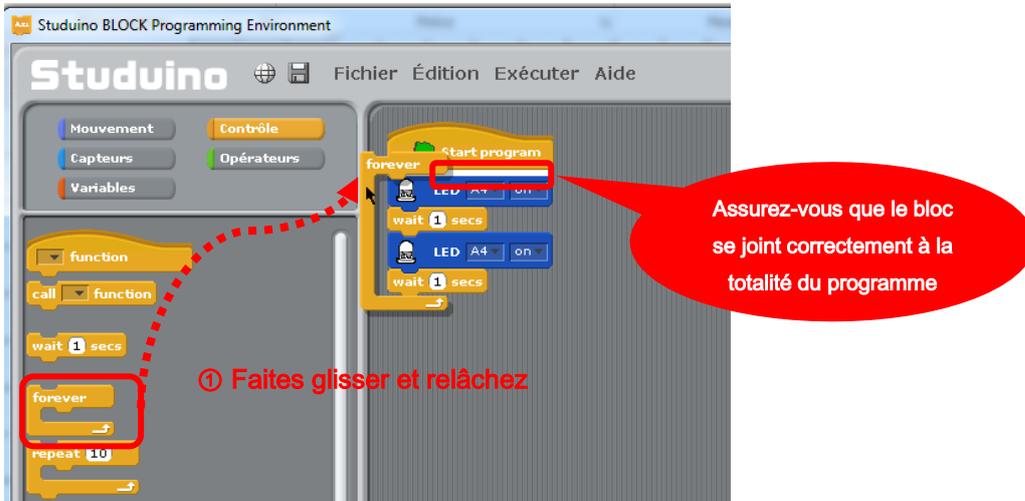
- 2 Effectuez un clic droit sur **LED A4 on** pour accéder au menu contextuel puis cliquez sur «Dupliquer».



- 3 Attachez le bloc dupliqué sous le bloc **wait 1 secs**.



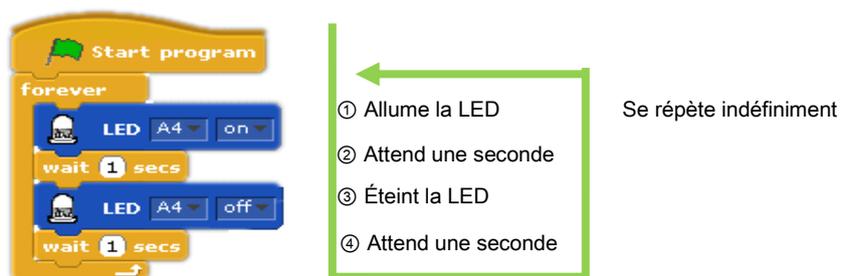
- ④ Faites glisser le bloc  de la palette «Contrôle» vers le champ de script puis relâchez-le afin que le bloc s'attache au reste du programme.



- ⑤ Réglez le bloc  le plus bas sur OFF.



Vous venez désormais de programmer votre LED pour qu'elle clignote chaque seconde de manière infinie.



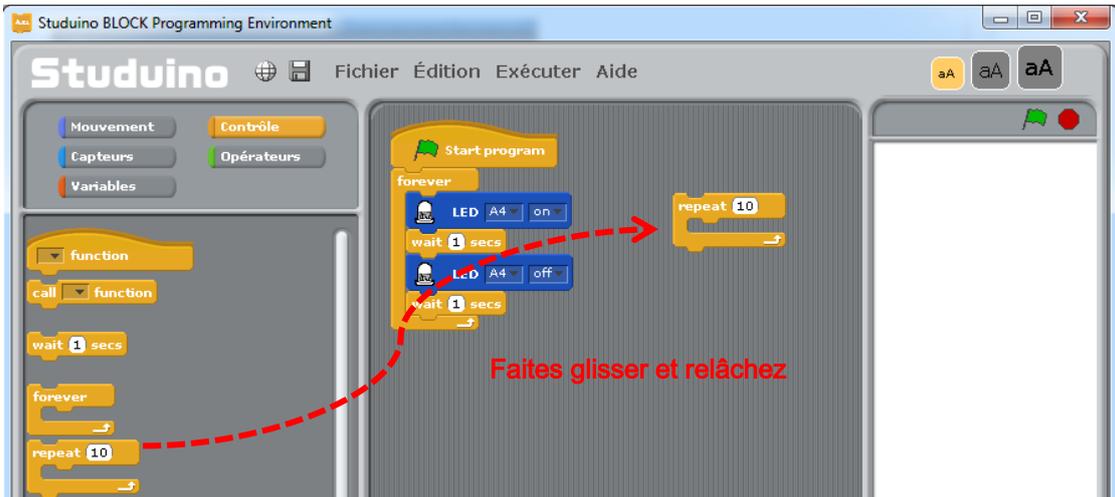
Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez «Transférer» depuis le menu «Exécuter». Une fois les données transférées, la LED du circuit clignotera automatiquement à une seconde d'intervalle.

2.5. Spécifier le nombre de clignotement de la LED

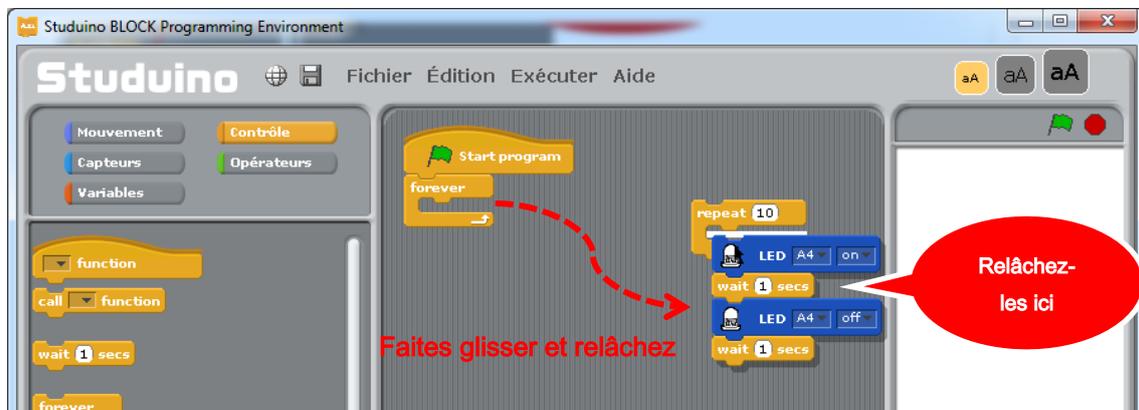
Maintenant, nous allons apprendre à contrôler le nombre de fois que nous voulons faire clignoter la LED. Utilisez le programme de clignotement de la LED de la section

précédente et un bloc  pour apprendre les boucles.

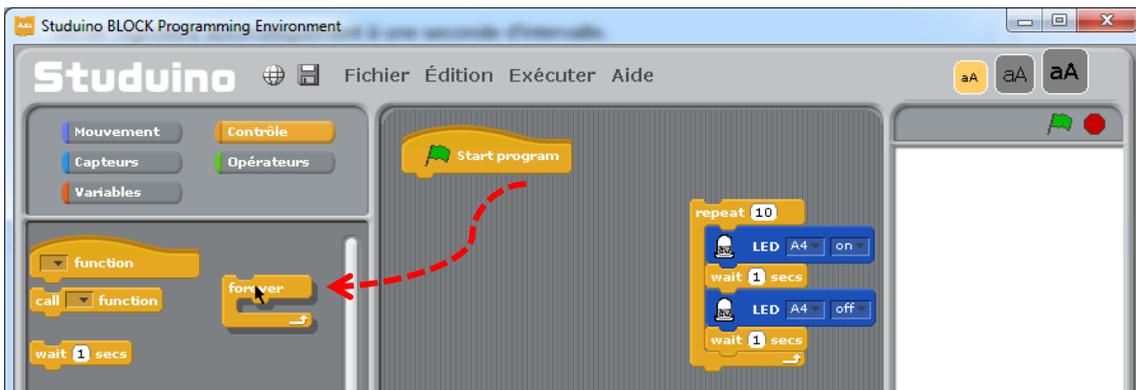
① Faites glisser un bloc  de la palette «Contrôle» vers le champ de script.



② Déplacez les blocs attachés à  vers l'ouverture de .



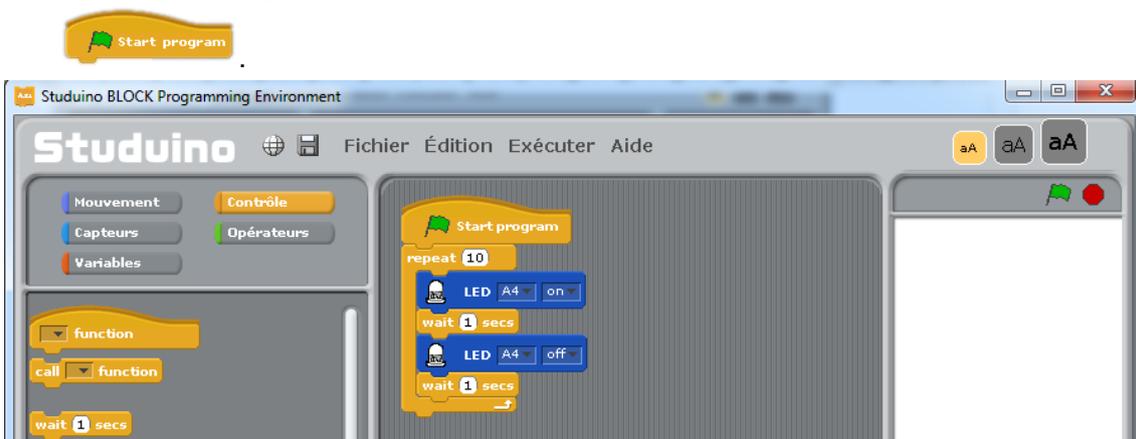
- ③ Enlevez le bloc  en le glissant vers la palette de blocs.



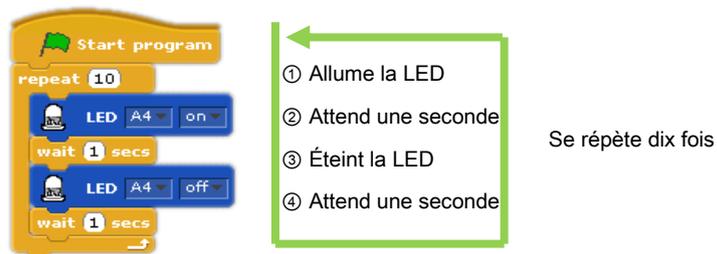
A l'exception du bloc maître , tous les blocs situés dans le champ de script peuvent être supprimés en les replaçant dans la palette de blocs.

- ★Effectuez un clic-droit sur un bloc pour afficher le menu contextuel et cliquez sur **supprimer** pour le supprimer.
- ★Sélectionnez «**Ne pas effacer**» à partir de la barre de menu pour restaurer votre action si vous l'avez effectué accidentellement.

- ④ Connectez le groupement de blocs que vous avez créé en étape 2 au bloc



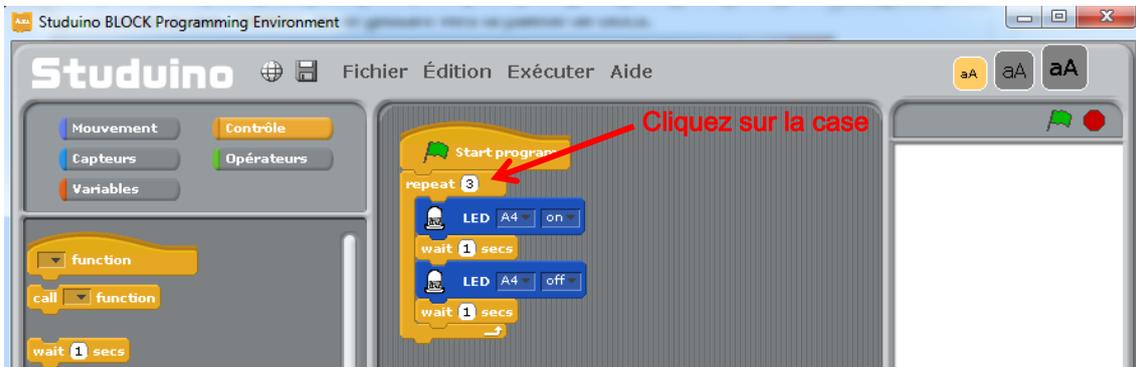
Le programme est maintenant complet. Il fera clignoter la LED à chaque seconde, dix fois de suite.



Connectez le circuit Studuino au PC en utilisant le câble USB. À partir du menu «**Exécuter**» choisissez «**Transférer**». Une fois les données transférées, la LED du circuit clignotera automatiquement 10 fois à une seconde d'intervalle.

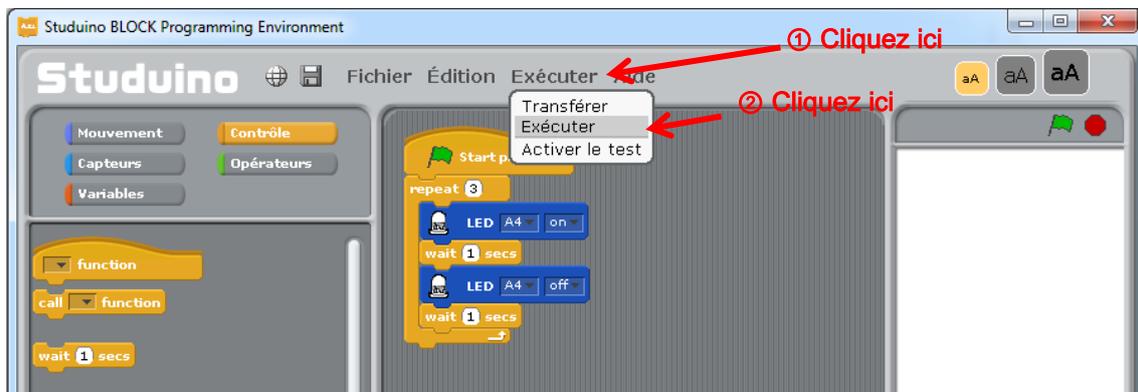
Ensuite, nous allons changer le nombre de boucles dans le bloc  afin de faire varier le nombre de clignotement.

- ⑤ Cliquez sur le nombre «10» dans le bloc  et remplacez-le par un nombre différent. La LED clignotera alors le nombre de fois que vous aurez défini.



Connectez le circuit Studuino au PC en utilisant le câble USB. À partir du menu «Exécuter», choisissez «Transférer». Une fois les données transférées, la LED du circuit clignotera automatiquement à une seconde d'intervalle.

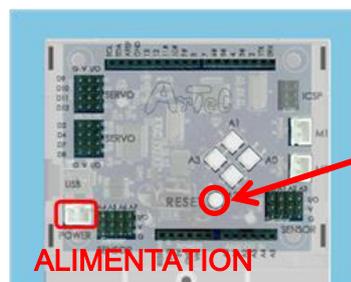
Démarrez le programme transféré en cliquant sur le menu «Exécuter» et en sélectionnant «Exécuter».



★Un programme transféré peut fonctionner sans être connecté à un PC, simplement à partir d'une batterie.

Retirez le câble USB du circuit Studuino. Connectez le boîtier de piles et activez l'interrupteur. La LED clignotera à intervalle d'une seconde durant le nombre de fois spécifié.

★Une fois le programme terminé, redémarrez-le en appuyant sur le bouton de réinitialisation du circuit.



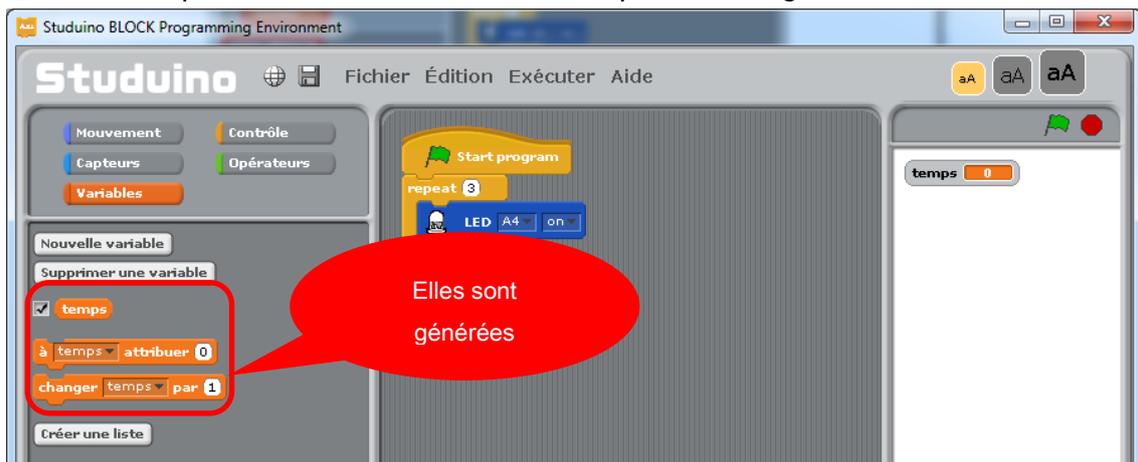
2.6. Changer la vitesse de clignotement

Maintenant, nous allons contrôler la vitesse à laquelle la LED clignote en utilisant comme base le programme de la section précédente. Nous allons le modifier en créant un programme qui utilise des blocs de variable.

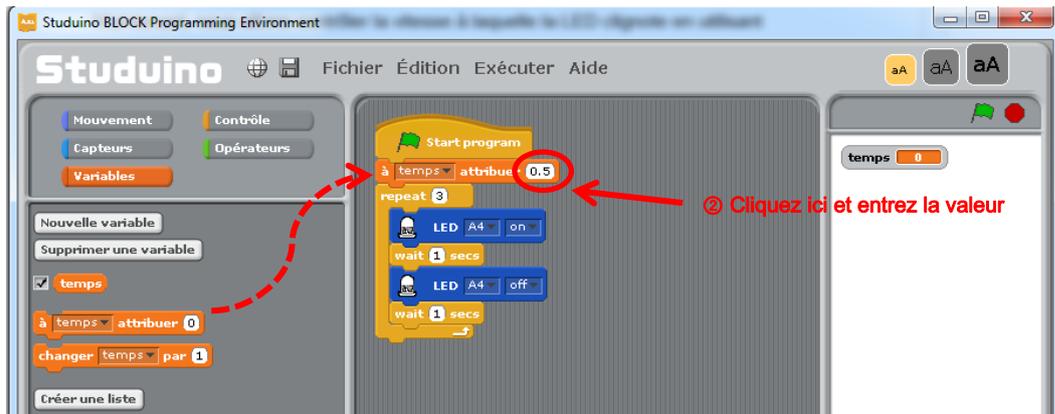
- ① À partir de la palette du groupe de commande, cliquez sur Variables puis **Nouvelle variable**. Une boîte de dialogue du paramétrage du nom apparaît. Appelez votre variable «temps» et cliquez sur OK.



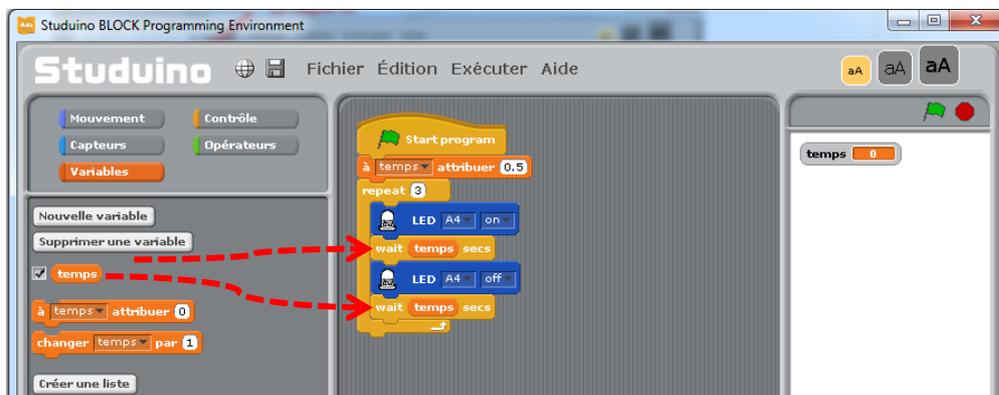
- ② Un bloc qui contrôle les variables de « temps » va être généré.



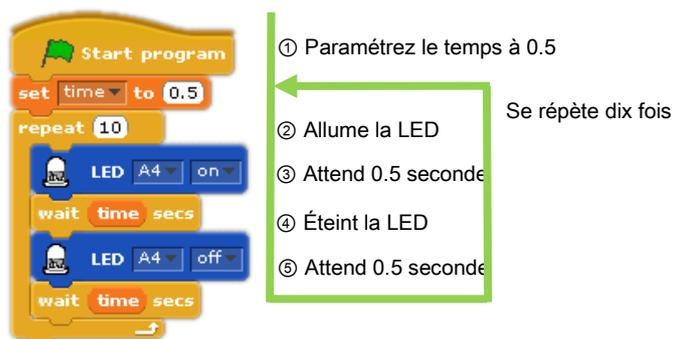
- ③ Connectez le bloc **set time to 0** au bloc **Start program** et réglez la valeur à 0.5.



- ④ Connectez le bloc **temps** au bloc **wait 1 secs**.



Le programme est achevé. Il fera donc clignoter votre LED 10 fois par intervalle de 0.5 seconde.



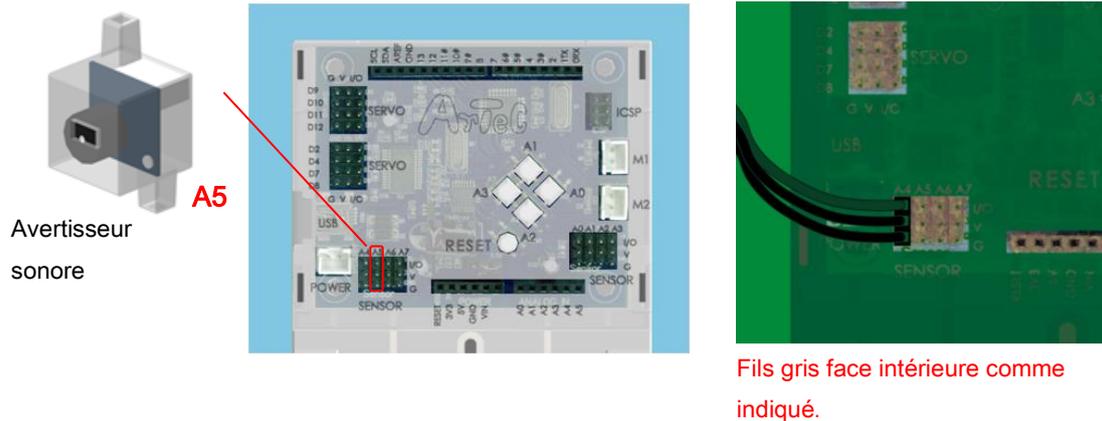
Connectez le circuit Studuino au PC en utilisant le câble USB. À partir du menu «**Exécuter**» choisissez «**Transférer**». Une fois les données transférées, la LED du circuit clignotera automatiquement 10 fois à 0.5 seconde d'intervalle.

3. Utiliser les avertisseurs sonores

Cette section explique comment utiliser un avertisseur sonore dans un programme, comment activer et interrompre le son et comment utiliser les blocs de liste et de fonction.

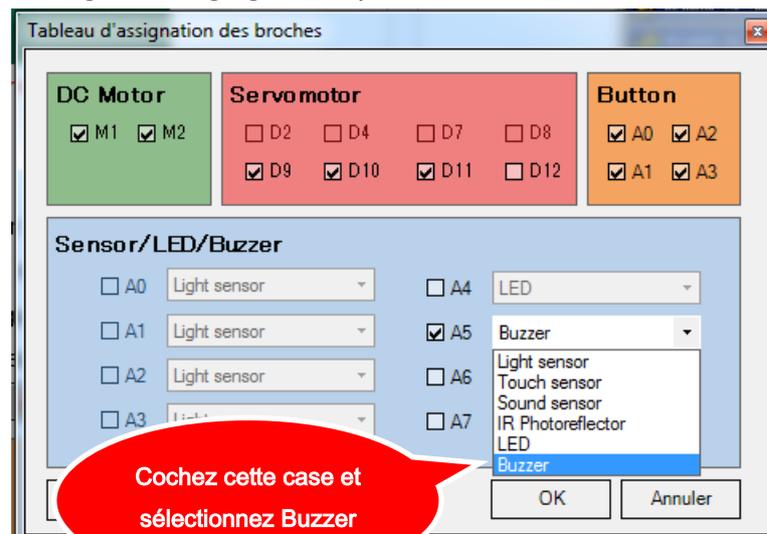
3.1. Connecter un avertisseur sonore

Connectez un avertisseur sonore au connecteur A5 (capteur/ LED/ avertisseur sonore) sur le circuit Studuino.



3.2. Paramétrage des ports

Paramétrez les ports dans l'environnement de programmation par blocs afin qu'il s'accorde au circuit. Sélectionnez «**Édition**» dans le menu, puis dans le menu déroulant sélectionnez «**Paramètres des ports**» pour accéder à la boîte de dialogue de réglages. Dans la boîte de dialogue de réglages des ports, cochez A5 et choisissez Buzzer.



3.3. Jouer une note

Maintenant, nous allons créer un programme qui permettra de faire sonner l'avertisseur sonore pendant une seconde. Pour contrôler le son, nous utiliserons le

bloc et le bloc .

★ Sélectionnez «Nouveau» sous «Fichier» dans le menu pour démarrer un nouveau programme.

① À partir de la palette «Mouvement», sélectionnez et connectez-le à .



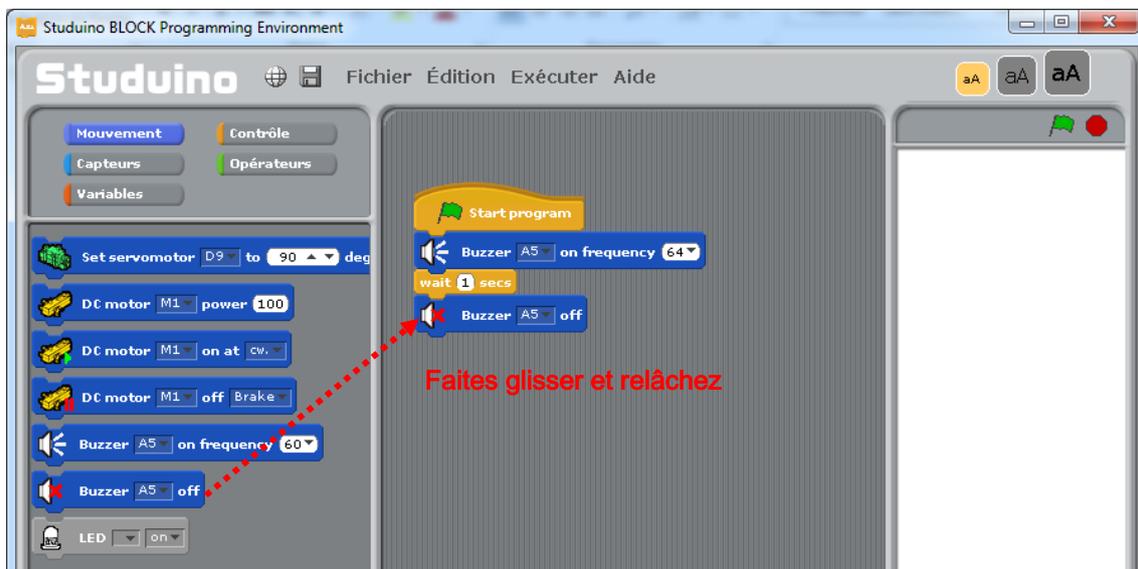
② Cliquez sur ▼ dans le bloc pour visualiser le clavier. En plaçant le curseur de la souris sur une touche du clavier, vous obtiendrez sa valeur. Maintenant cliquez sur E (64).



③ À partir de la palette «Contrôle», choisissez le bloc  et connectez-le au bloc



④ À partir de la palette «Mouvement», choisissez un bloc  et connectez-le au bloc .



Le programme est achevé. L'avertisseur sonore émettra alors la note E pendant une seconde.

Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez «Transférer» depuis le menu «Exécuter». Une fois les données transférées, l'avertisseur sonore émettra la note E pendant une seconde.

3.4. Jouer une mélodie

Créez une mélodie qui joue les notes C, C, G, G, A, A, C et qui correspondent à la mélodie «brille, brille, petite étoile». L'avertisseur sonore est capable de jouer cette mélodie en émettant ces sons les uns à la suite des autres.

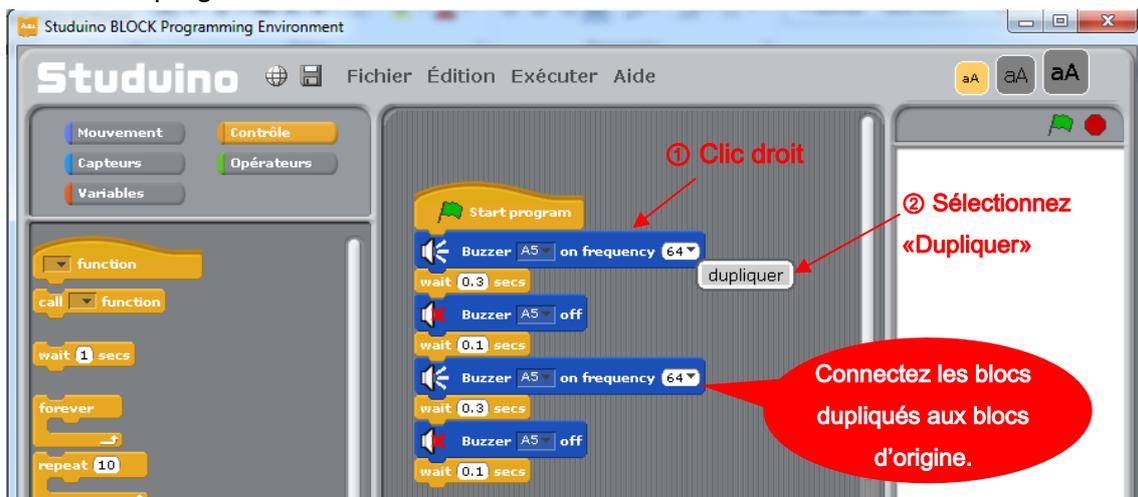
- ① Ajoutez un bloc **wait 1 secs** au programme que vous avez créé dans la section précédente.



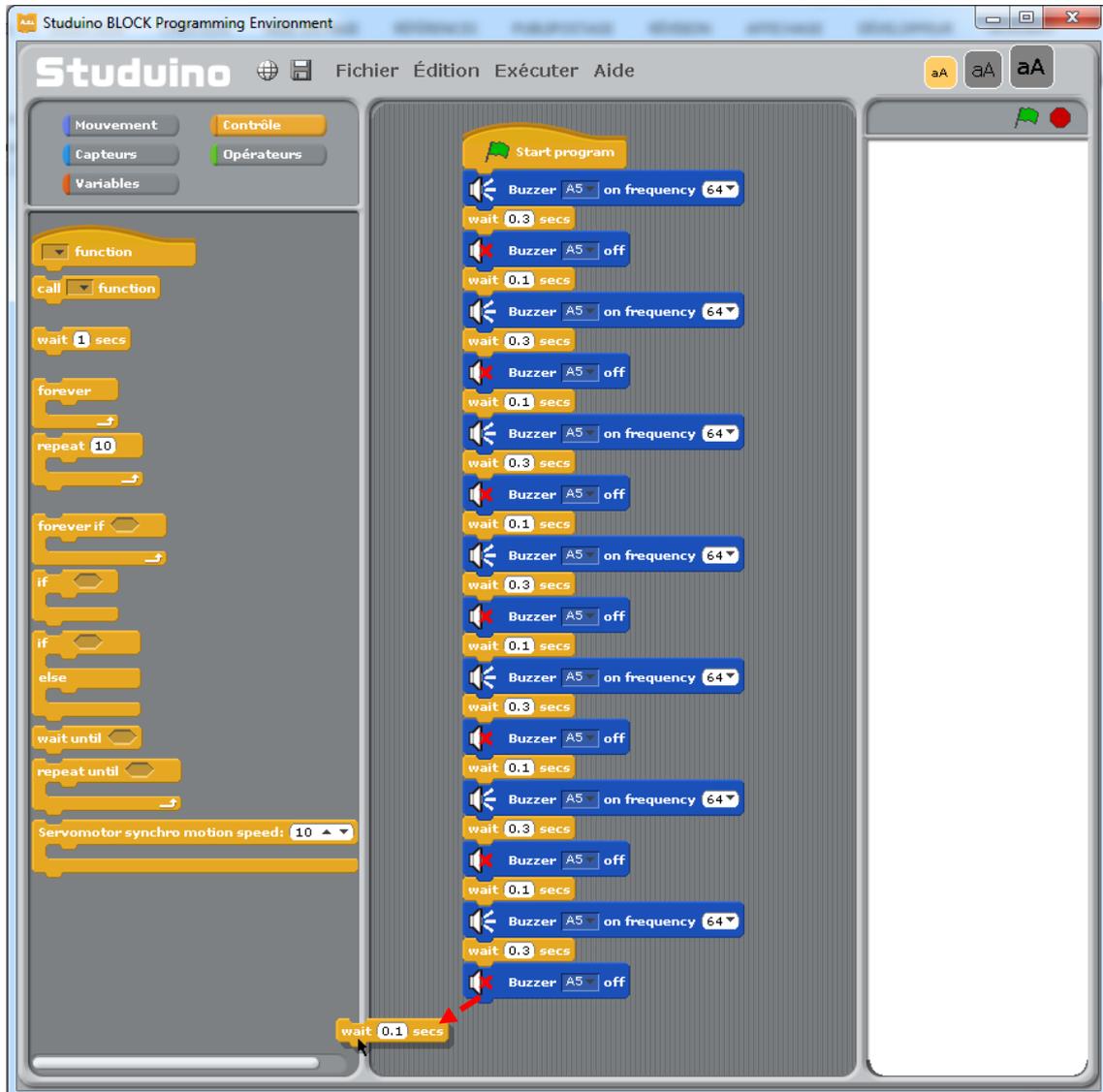
- ② Paramétrez les blocs **wait 1 secs** à 0.3 et 0.1 seconde.



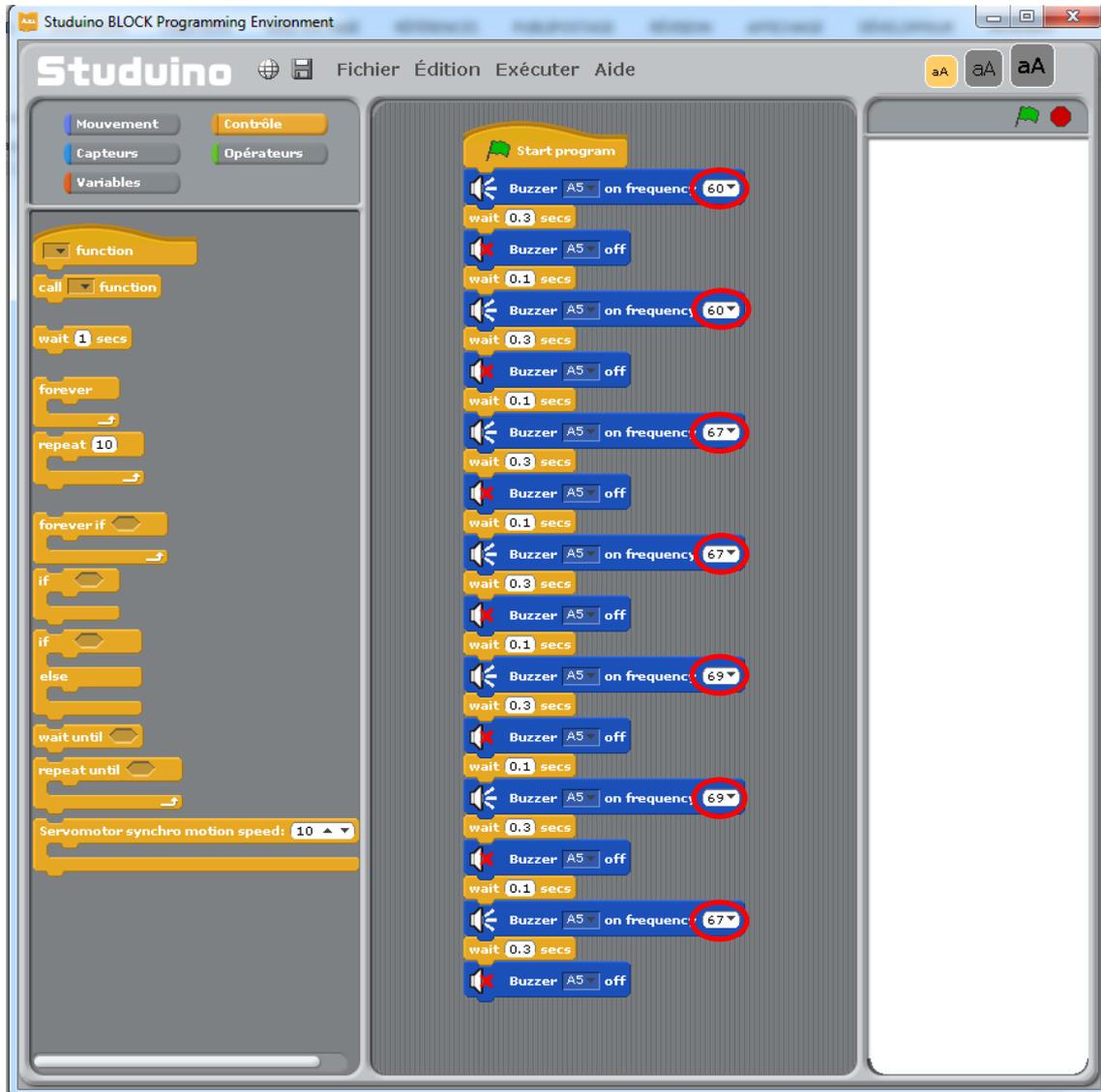
- ③ Effectuez un clic droit sur le bloc **Buzzer A5 on frequency 64** pour accéder au menu contextuel puis cliquez sur «Dupliquer». Attachez les blocs venant d'être dupliqués à votre programme.



- ④ Répétez l'étape ③ pour faire 7 paires des blocs **wait 1 secs** et **Buzzer A5 on frequency 64**. S'il y en a plus que prévu, faites simplement glisser les blocs de trop vers la palette de commande.



- ⑤ Paramétrez le bloc  sur C (60), C (60), G (66), G (66), A (67), A (67), et G (66).



Le programme est achevé. L'avertisseur sonore émet alors l'air «brille, brille, petite étoile».

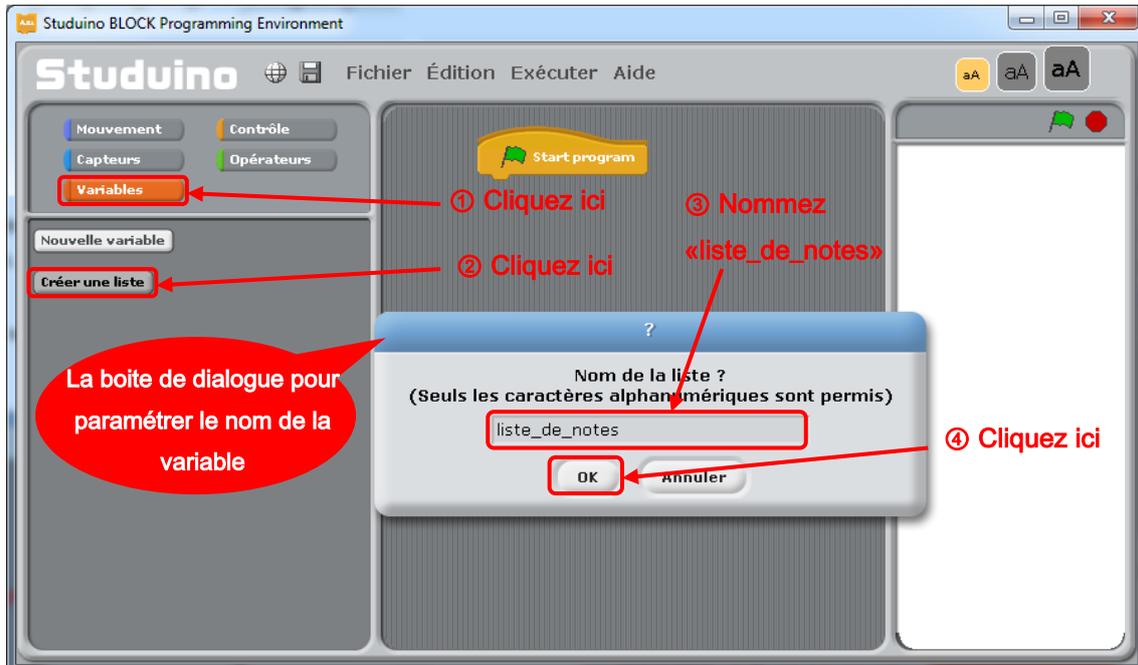
Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez «Transférer» depuis le menu «Exécuter». Une fois les données transférées, l'avertisseur sonore émettra alors l'air «brille, brille, petite étoile».

3.5. Faire une mélodie en utilisant une liste

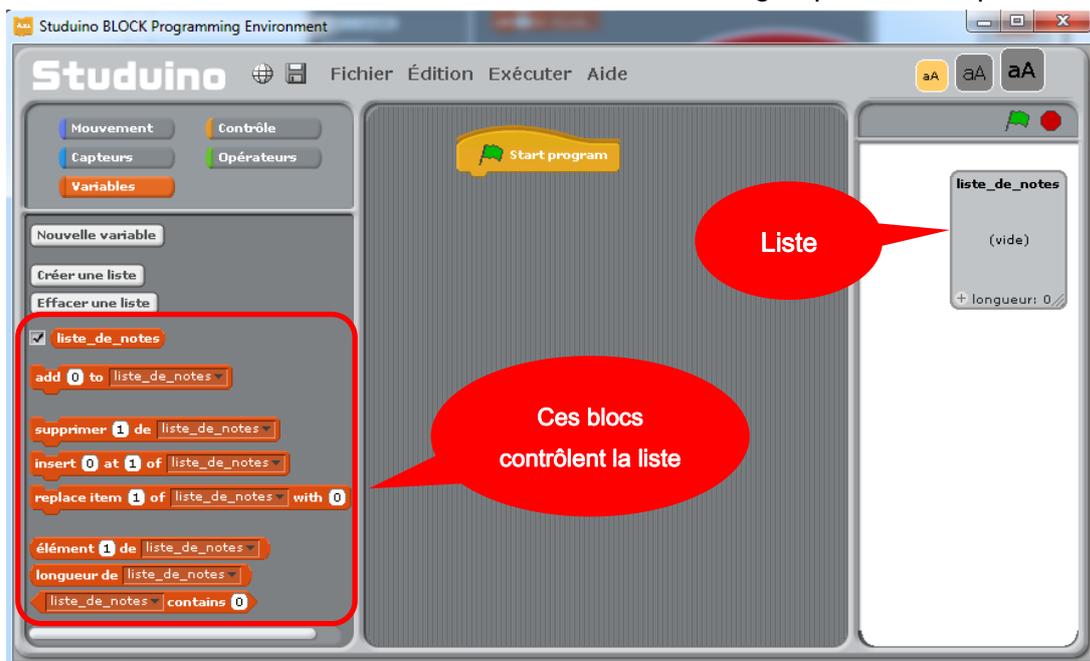
Le programme créé dans la précédente section peut être simplifié par l'utilisation d'une liste. Ce que vous aurez à faire, c'est de créer une liste.

3.5.1. Créer une liste

- ① Dans la palette «Variables», cliquez sur le bouton **Créer une liste**. Nommez la liste «liste_de_notes».



- ② Deux éléments seront alors créés : une liste vide et un groupe de blocs qui la contrôle.



Maintenant, nous allons apprendre comment chaque bloc contrôle une liste.

- ③ Faites glisser 3 blocs  vers le champ de script et assemblez-les ensemble. Paramétrez-les pour être à 0, 12 et 15 comme indiqué ci-dessous.



- ④ Cliquez sur  et les blocs connectés à  s'exécuteront dans l'ordre. Les valeurs 0, 12 et 15 apparaîtront dans la liste.

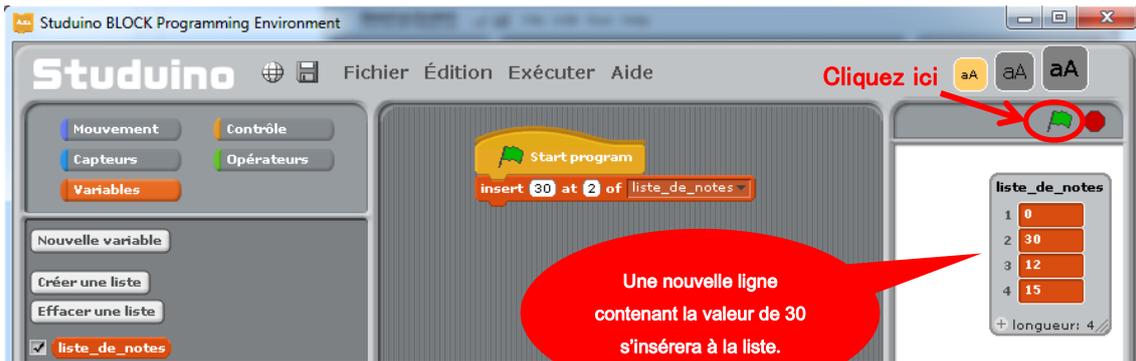


Les étapes suivantes expliqueront les processus de la liste réalisée à l'étape 4.

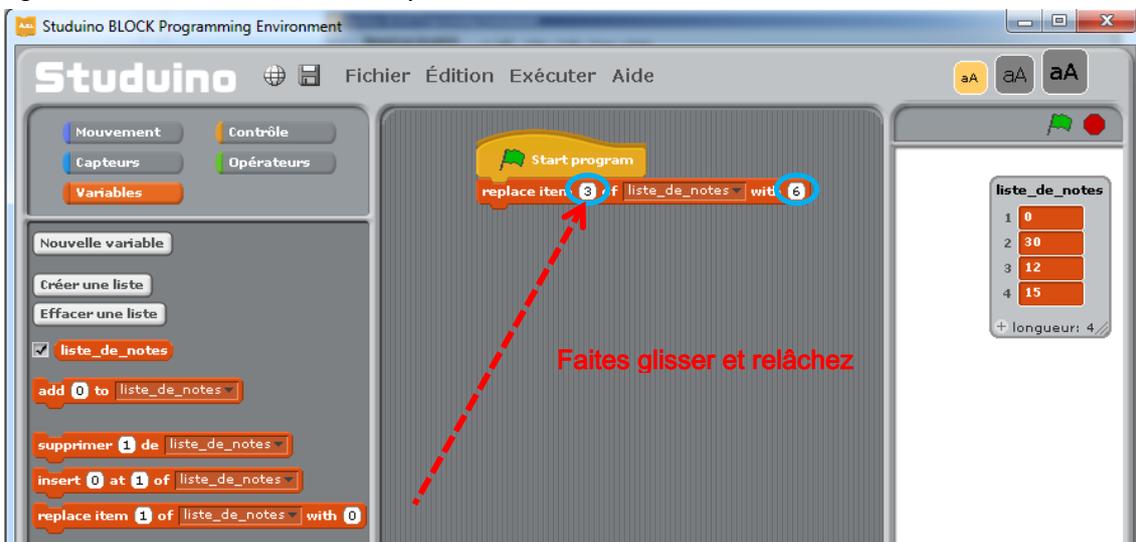
- ⑤ Supprimez les blocs créés à l'étape 3. Attachez un bloc  à  et saisissez les nombres ci-dessous à la place de 30 au niveau de la seconde ligne de la liste.



- ⑥ Cliquez sur .



- ⑦ Supprimez `insert 30 at 2 of list_note`, puis faites glisser `replace item 1 of list_note with 0` vers le champ de script. Attachez-le et paramétrez-le pour remplacer la valeur de la ligne trois dans liste_de_notes par 6.



- ⑧ Cliquez sur .



- ⑨ Ensuite, supprimez `replace item 1 of liste_notes with 0`, puis faites glisser `supprimer 1 de liste_de_notes` vers le champ de script. Attachez-le et paramétrez-le pour supprimer la valeur de la seconde ligne dans liste_de_notes.



- ⑩ Cliquez sur .



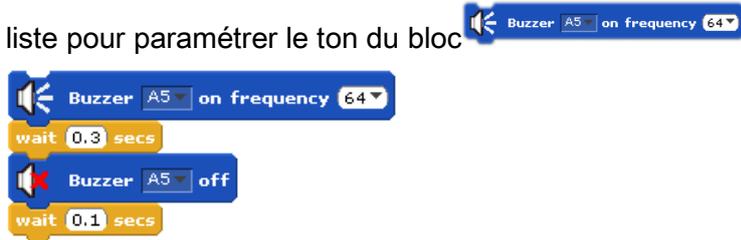
Vous pouvez utiliser les étapes listées ci-dessus pour modifier (ajouter, insérer, substituer ou supprimer) les valeurs dans une liste. Vous pouvez également accéder aux informations contenues dans la liste avec les blocs suivants.

Bloc	Processus
<code>élément 1 de liste_de_notes</code>	Obtient la valeur d'une ligne spécifiée.
<code>longueur de liste_de_notes</code>	Obtient la longueur d'une liste spécifiée.
<code>liste_de_notes contient 0</code>	Vérifie si la liste contient la valeur spécifiée.

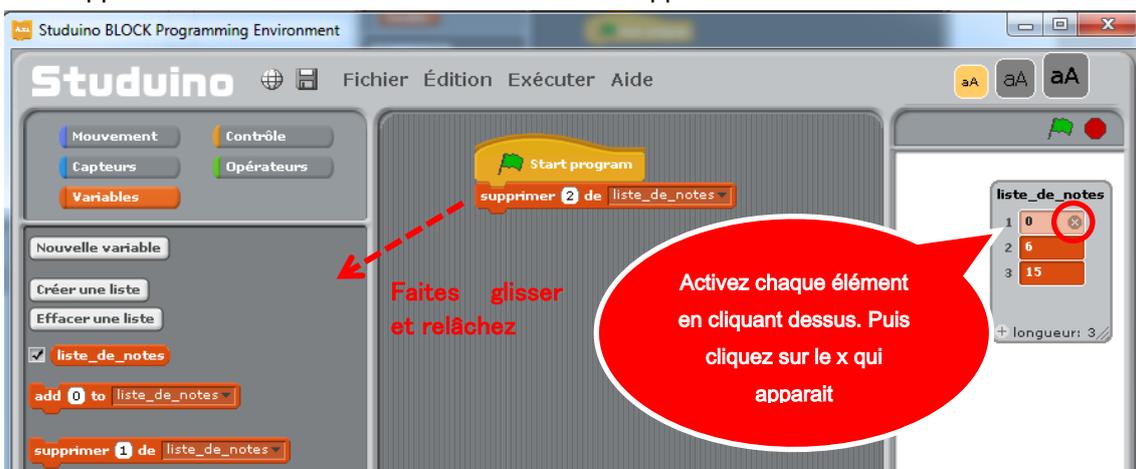
Ensuite, nous allons utiliser une liste pour recréer un programme à partir d'une mélodie jouée.

- Jouer une mélodie en utilisant une liste

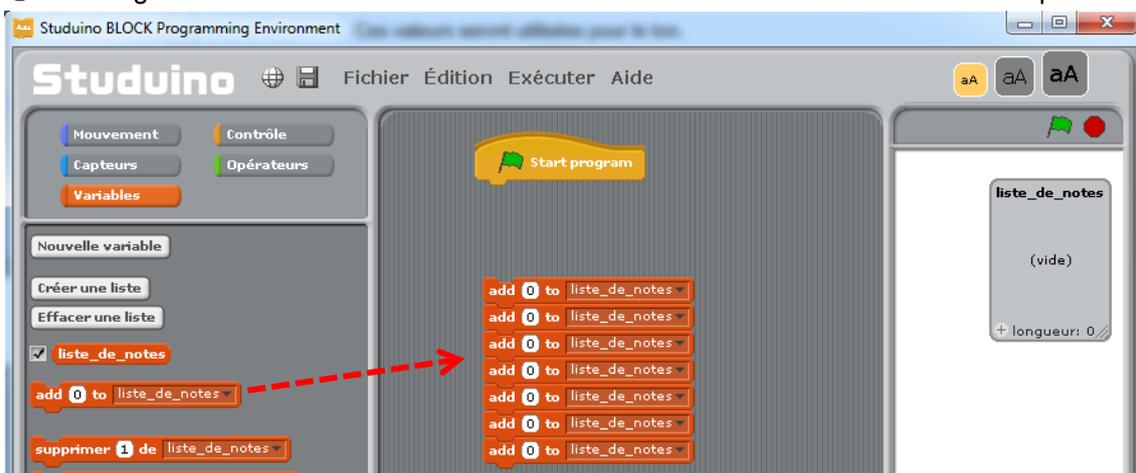
Le programme créé en 3.4. **Jouer une mélodie** est joué à sept reprises avec des tons différents en utilisant des blocs combinés. Nous allons simplifier ce programme en utilisant une liste pour paramétrer le ton du bloc



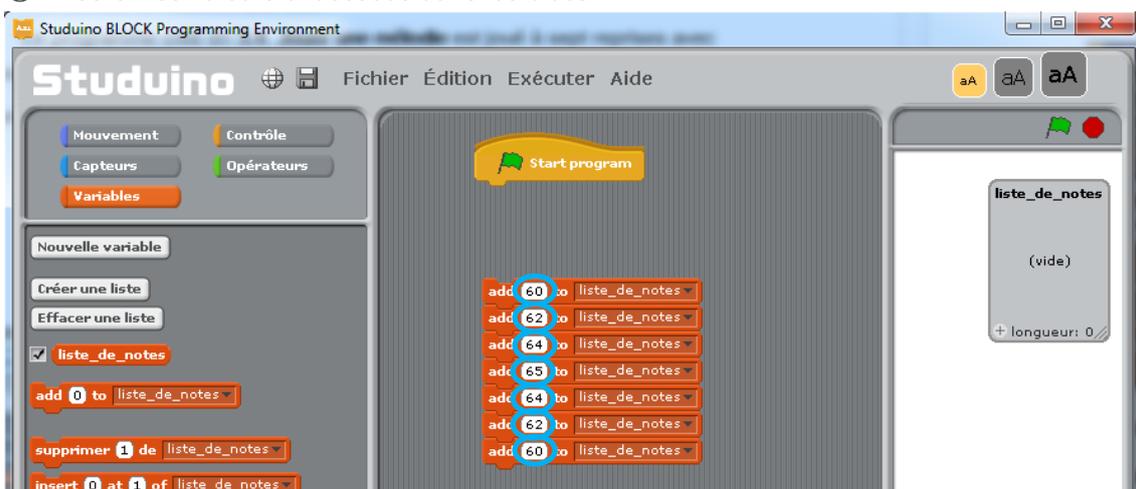
⑪ Supprimez **supprimer 1 de liste_de_notes** et ensuite supprimez tous les éléments de la liste.



⑫ Faites glisser 7 blocs **add 0 to liste_de_notes** et assemblez-les ensemble comme indiqué.

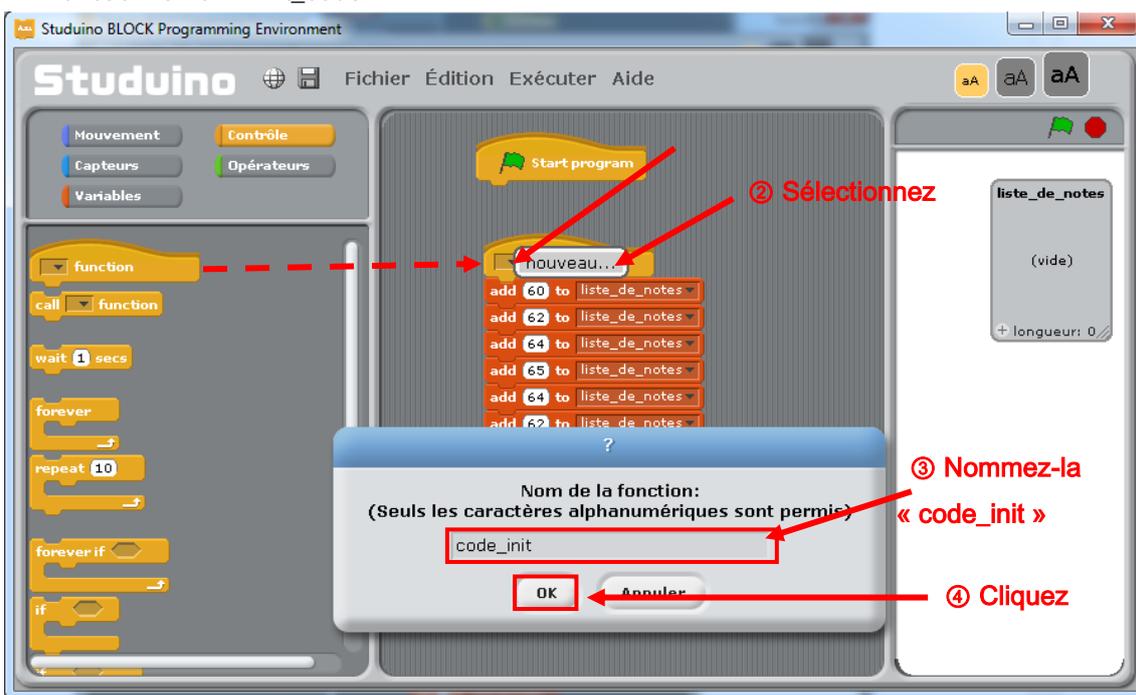


13 Insérez les valeurs ci-dessous dans les blocs `add 0 to liste_de_notes`.



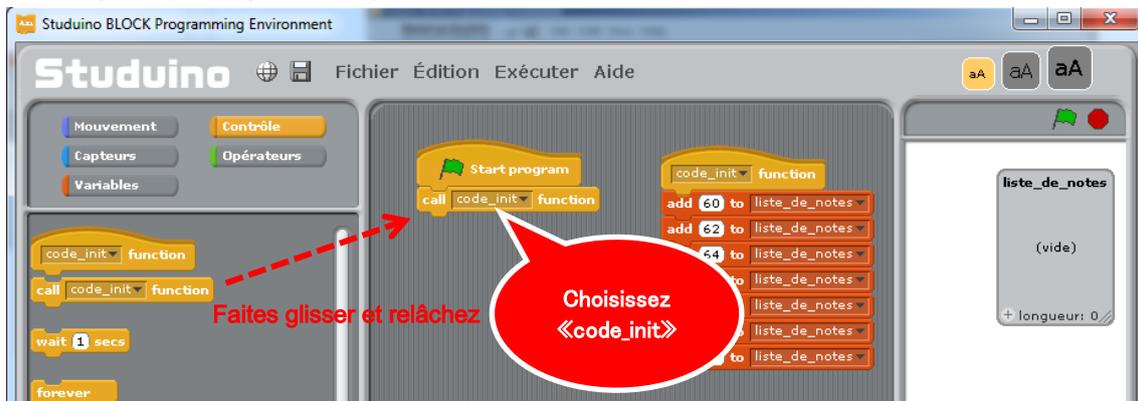
Ces valeurs seront utilisées pour la tonalité.

14 Faites glisser `function` depuis la palette «Contrôle» et collez-le au-dessus des blocs créés à l'étape 13. Cliquez sur le symbole ▼ puis sélectionnez «new ...» et donnez à cette fonction le nom «init_code.».



La fonction complétée «code_init» va mettre en place le changement de tonalité de chaque note de « brille, brille, petite étoile ». Les blocs attachés à `function` seront exécutés à la lecture du programme. Lorsque `call function` est exécuté, chaque bloc `add 0 to list` sera lu et les nombres seront ajoutés à la liste.

- ⑮ Faites glisser **call** **function** de la palette «Contrôle» et connectez-le à **Start program**. Cliquez sur le symbole ▼ puis sélectionnez la fonction code_init.



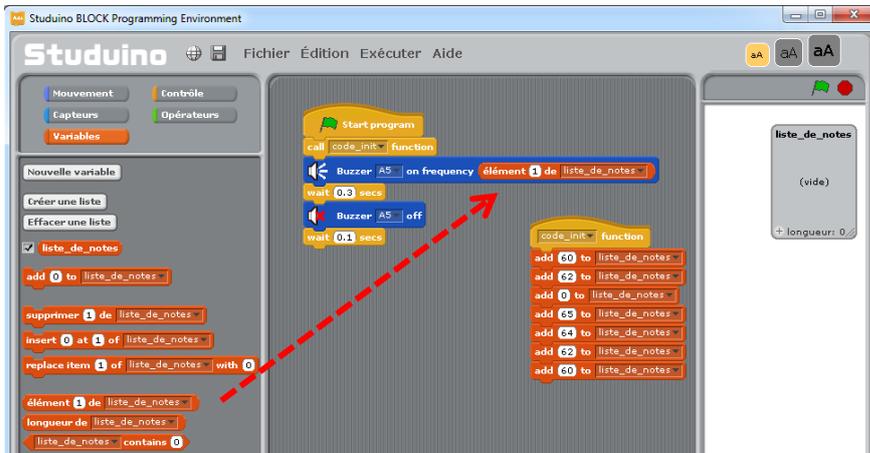
- ⑯ Faites glisser **Buzzer A5 on frequency 60** de la palette «Mouvement» et connectez-le à **Buzzer A5 off**.



- ⑰ Faites glisser deux **wait 1 secs** à partir de la palette «Contrôle» et paramétrez-les à 0.3 et 0.1.



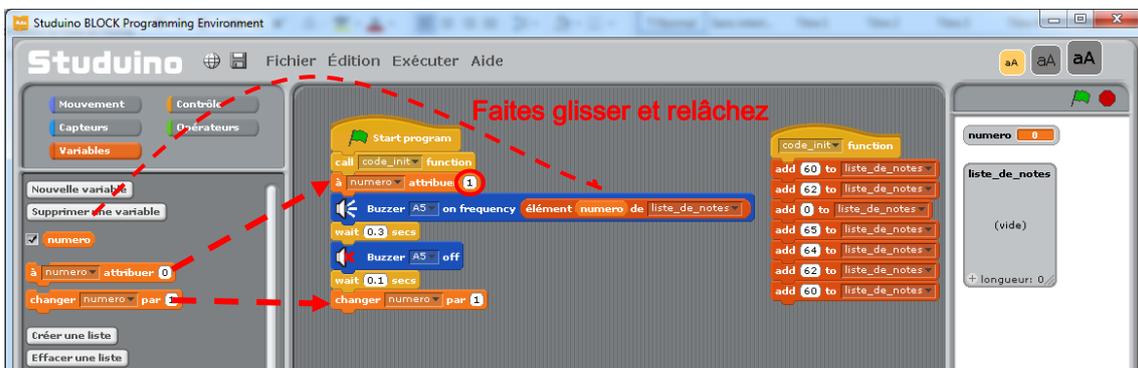
- 18) Faites glisser **élément 1 de liste_de_notes** depuis la palette «Variables» et insérez-le à **Buzzer A5 on frequency 60** comme indiqué.



- 19) Cliquez sur le bouton **Nouvelle variable** dans la palette «Variables». Nommez la variable **numero**.



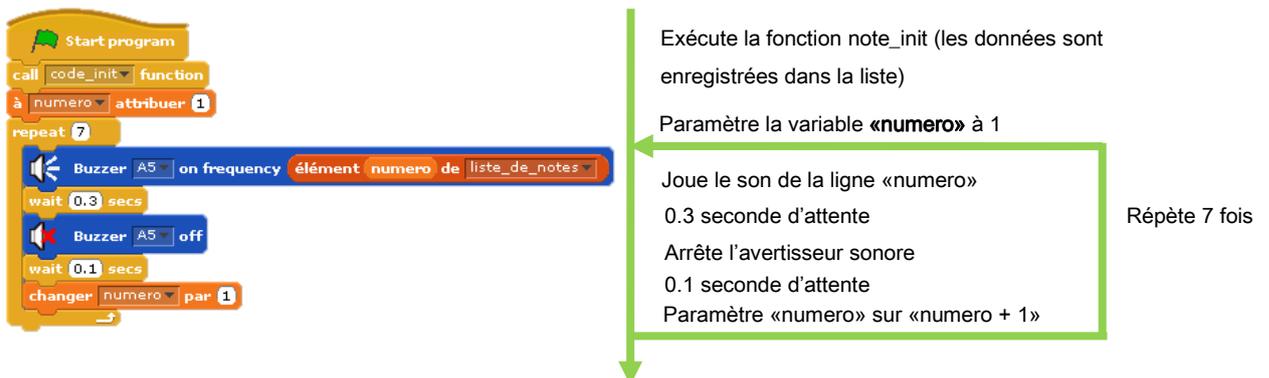
- 20) Faites glisser et placez le bloc de variable qui vient d'être créé aux places indiquées ci-dessous. Paramétrez à 1.



② Connectez un bloc  de la palette «Contrôle» et réglez-le sur 7.



Le programme que vous venez d'achever utilise une liste pour jouer le début de la comptine par l'intermédiaire des actions suivantes.



Exécute la fonction note_init (les données sont enregistrées dans la liste)
 Paramètre la variable «numero» à 1

Joue le son de la ligne «numero»
 0.3 seconde d'attente
 Arrête l'avertisseur sonore
 0.1 seconde d'attente
 Paramètre «numero» sur «numero + 1»

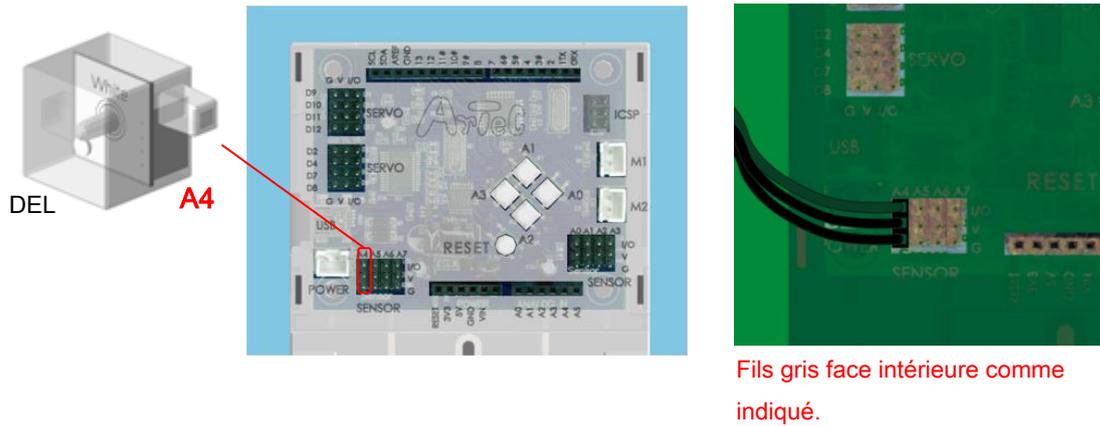
Répète 7 fois

La valeur de «numero» augmente d'une unité à chaque fois qu'une boucle est complétée, en utilisant «numero» en tant que variable pour jouer dans l'ordre chaque son enregistré.

Connectez votre Studuino au PC via le câble USB et choisissez «Transférer» depuis le menu «Exécuter». Une fois le programme transféré, l'avertisseur sonore jouera alors l'air de «brille, brille, petite étoile».

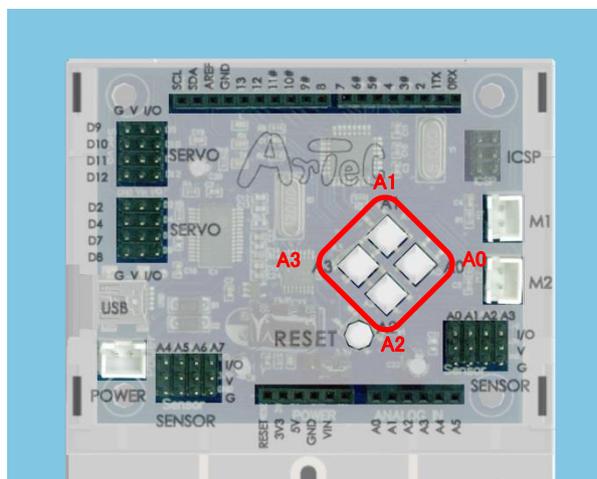
4. Interrupteurs et capteurs

Cette section explique comment créer un programme qui utilise les capteurs et les interrupteurs à bouton-poussoir. Il aborde aussi les blocs de condition et le mode test. Cet exemple de programmation se fait à partir d'une LED. Utilisez le connecteur A4 pour connecter la LED au circuit Studuino.



4.1. Interrupteurs à bouton-poussoir

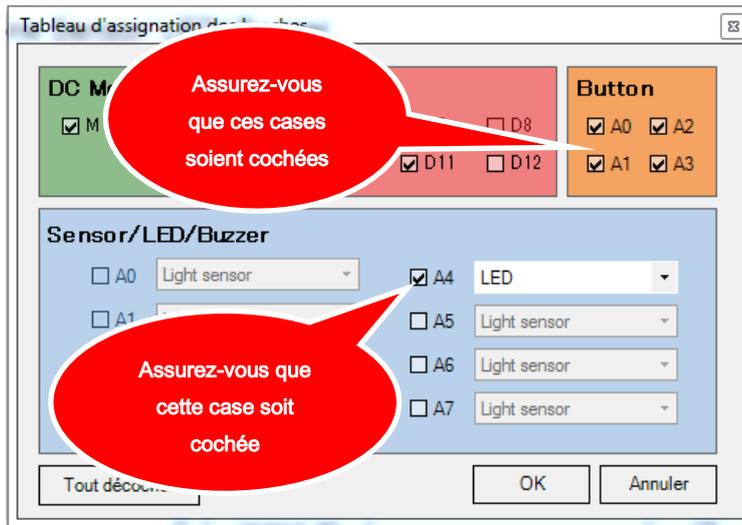
Les interrupteurs à bouton-poussoir sont montés sur le circuit Studuino. Les broches du connecteur A0 à A3 sont assignées aux interrupteurs.



4.1.1. Paramétrage des ports

Paramétrez les ports de l'environnement de programmation par blocs afin qu'il s'accorde au circuit. Sélectionnez «**Édition**» dans le menu, puis dans le menu déroulant sélectionnez «**Paramètres des ports**» pour accéder à la boîte de dialogue de réglages.

Dans la boîte de dialogue de paramétrage des ports, assurez-vous que les cases allant de A0 à A3 soient bien cochées. Pour A4, sélectionnez «LED» dans la zone Sensor/ LED/ Buzzer.

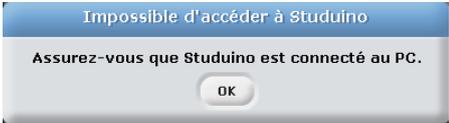
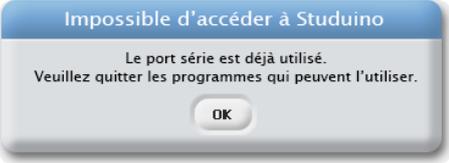


4.1.2. Tester un interrupteur à bouton-poussoir

Testez l'interrupteur à bouton-poussoir afin de vous assurer qu'il fonctionne correctement. Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «**Exécuter**».



Le mode test ne fonctionnera pas si les messages ci-dessous apparaissent.
 Testez les solutions proposées et essayez à nouveau.

Message	Solution
	<p>Assurez-vous que le circuit du Studuino est correctement connecté à votre ordinateur par le câble USB.</p>
	<p>Fermez toutes les applications qui peuvent entrer en communication avec votre Studuino. Si aucune autre application ne communique avec le circuit, débranchez le câble USB et appuyez sur le bouton de réinitialisation.</p>

Presser le bouton de réinitialisation du circuit Studuino pendant l'exécution du mode test causera la rupture de la connexion de votre Studuino vers votre PC et sera suivi du message suivant.

Message	Solution
	<p>Bien que vous puissiez continuer à programmer une fois le mode test fermé, le logiciel ne reconnaîtra plus votre Studuino. Nous vous recommandons de redémarrer le logiciel si vous voyez ce message.</p>

L'interrupteur à bouton-poussoir envoie une valeur égale à 1 à son état normal, et égale à 0 lorsque le bouton est enfoncé. Pressez les interrupteurs à bouton-poussoir de A0 jusque A3 et vérifiez si les valeurs correspondantes changent dans le tableau des capteurs du circuit.

[A0] Button	0
[A1] Button	1
[A2] Button	1
[A3] Button	1
[A4] LED	0
[A5] Pas connecté	0
[A6] Pas connecté	0
[A7] Pas connecté	0

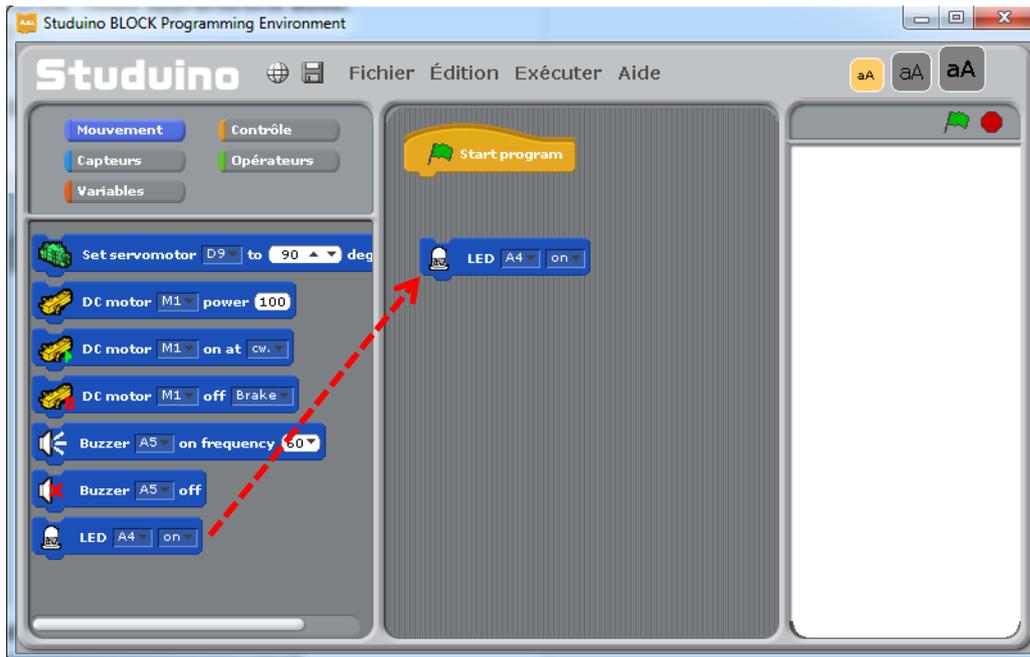
Pendant que l'interrupteur à bouton-poussoir est enfoncé, la valeur affichée indique 0.

Si les valeurs répondent correctement, alors le circuit fonctionne convenablement. Maintenant, quittez le mode test en sélectionnant «**Désactiver le test**» depuis le menu «**Exécuter**».

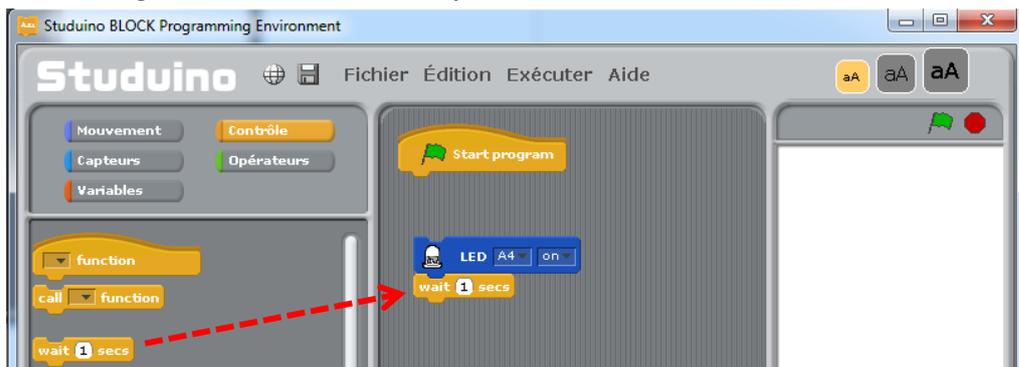
4.1.3. Programmer

Maintenant, nous allons créer un programme qui va permettre à la LED de clignoter lorsque l'interrupteur à bouton-poussoir est enfoncé. Nous apprendrons aussi comment créer un programme qui utilise le bloc .

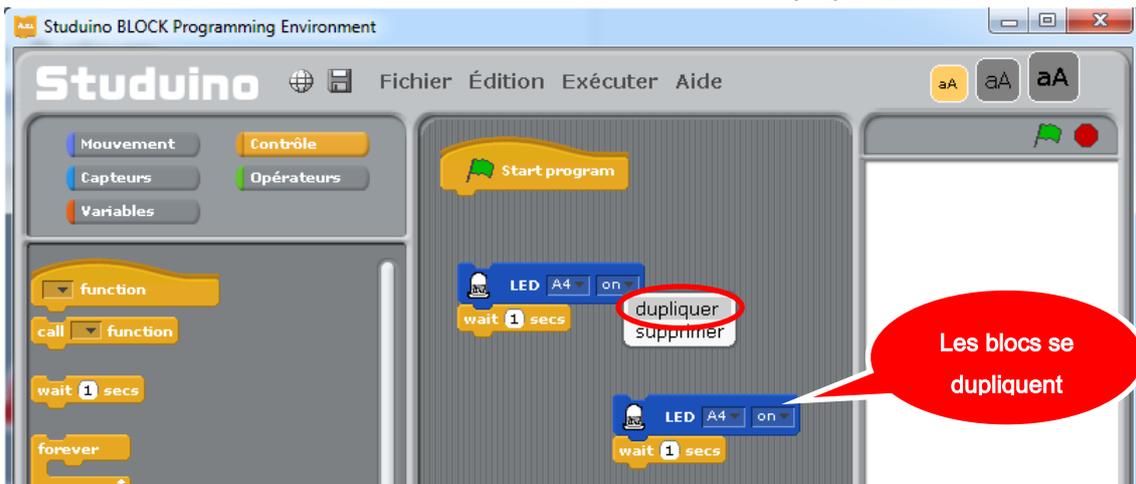
① Faites glisser un bloc  de la palette «Mouvement» vers le champ de script.



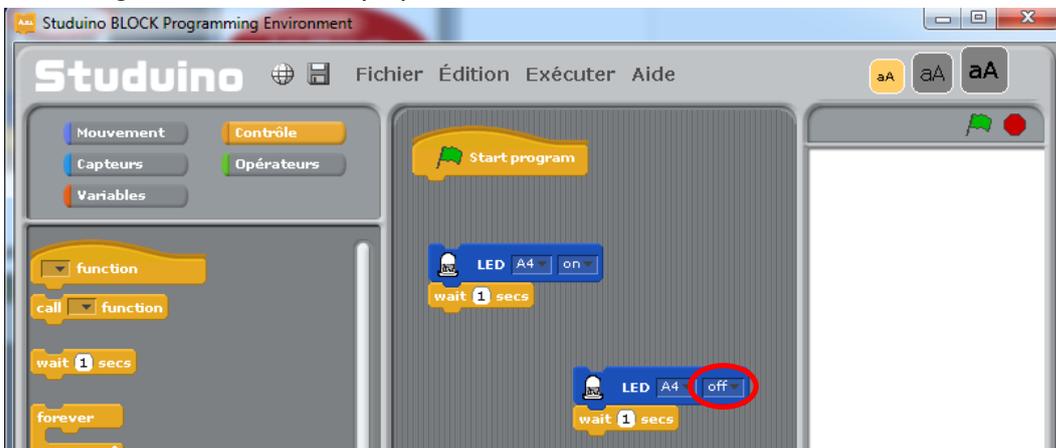
② Faites glisser  de la palette «Contrôle» et attachez-le aux autres blocs.



③ Faites un clic droit sur  et sélectionnez «Dupliquer».



④ Changez le bloc LED dupliqué de «on» à «off» .

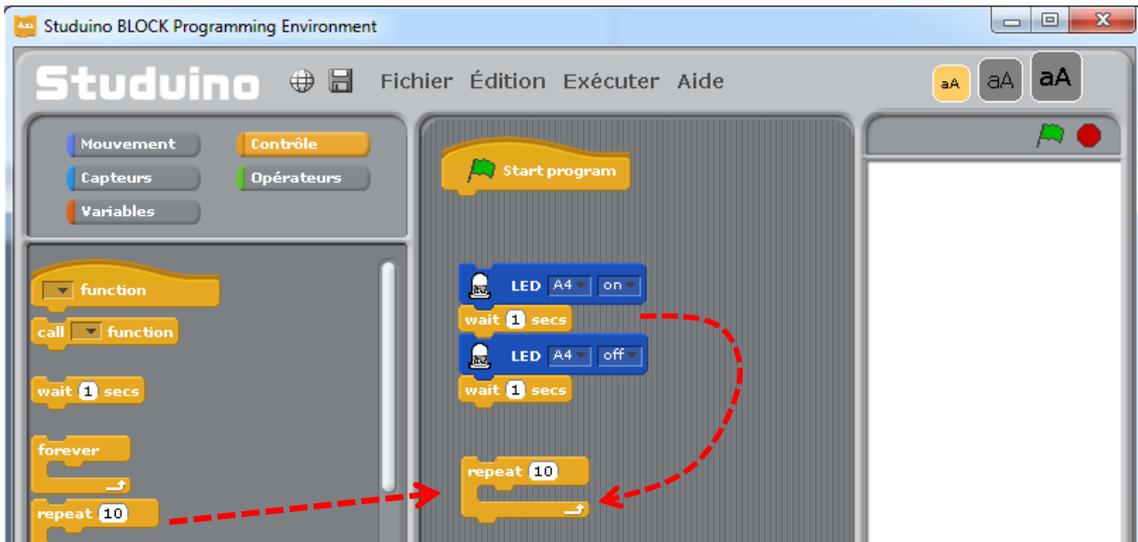


⑤ Connectez les blocs dupliqués aux blocs d'origine.

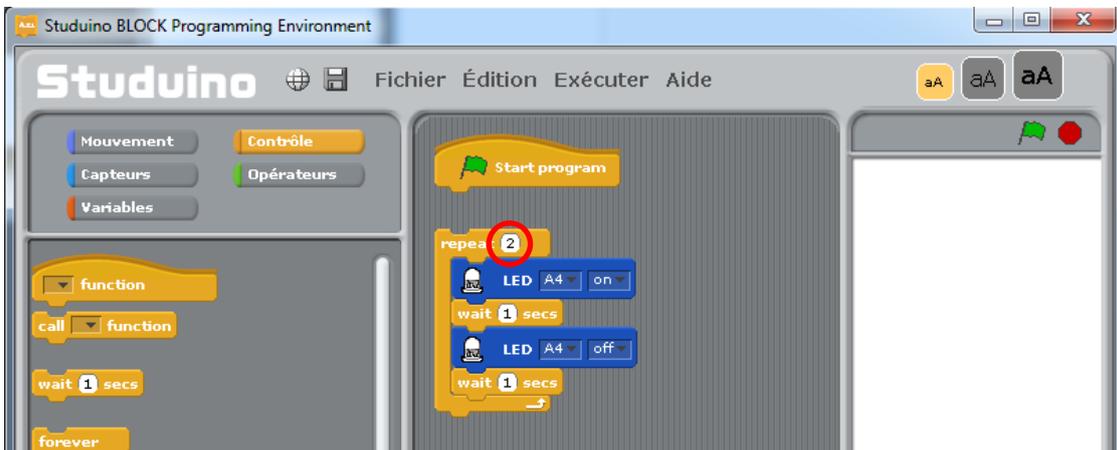


⑥ À partir de la palette «Contrôle», faites glisser  vers le champ de script.

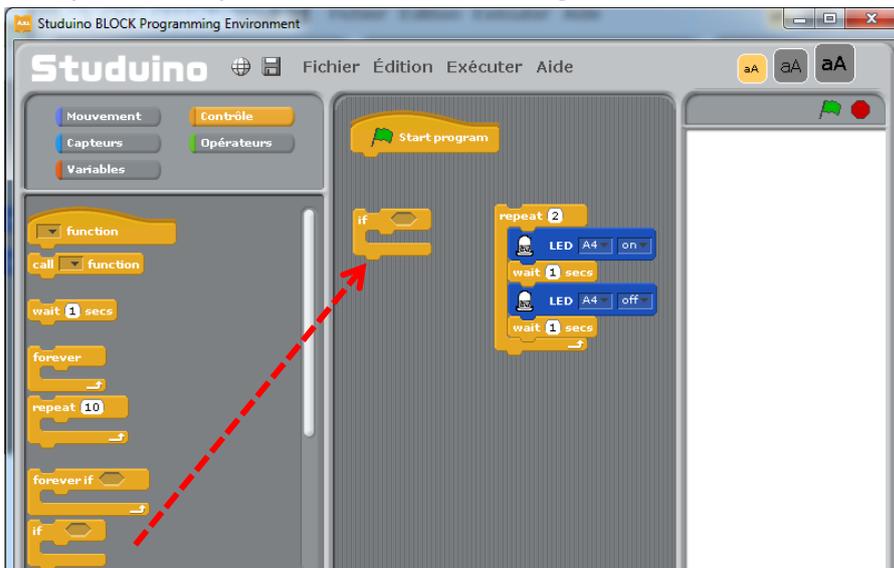
Mettez les blocs de l'étape 4 dans le bloc .



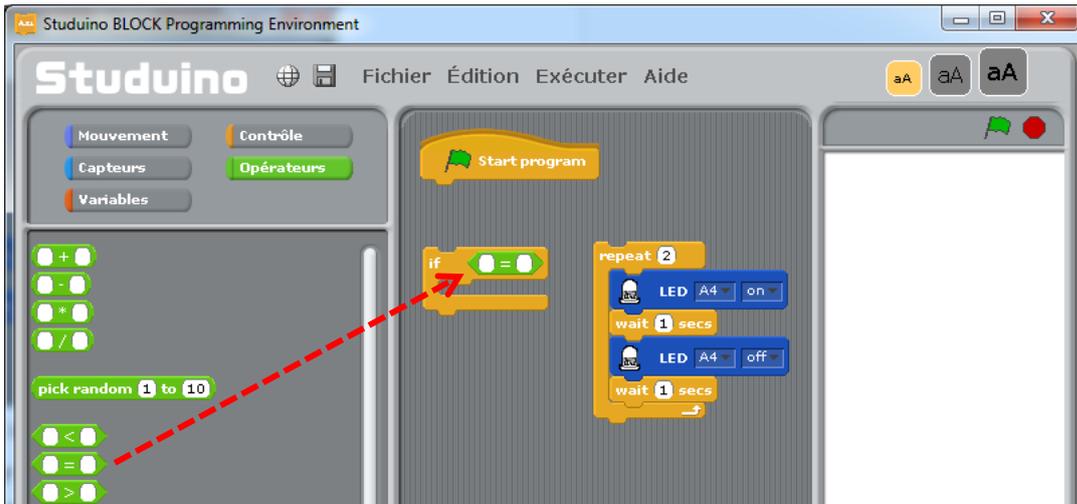
⑦ Paramétrez le nombre de boucle à 2.



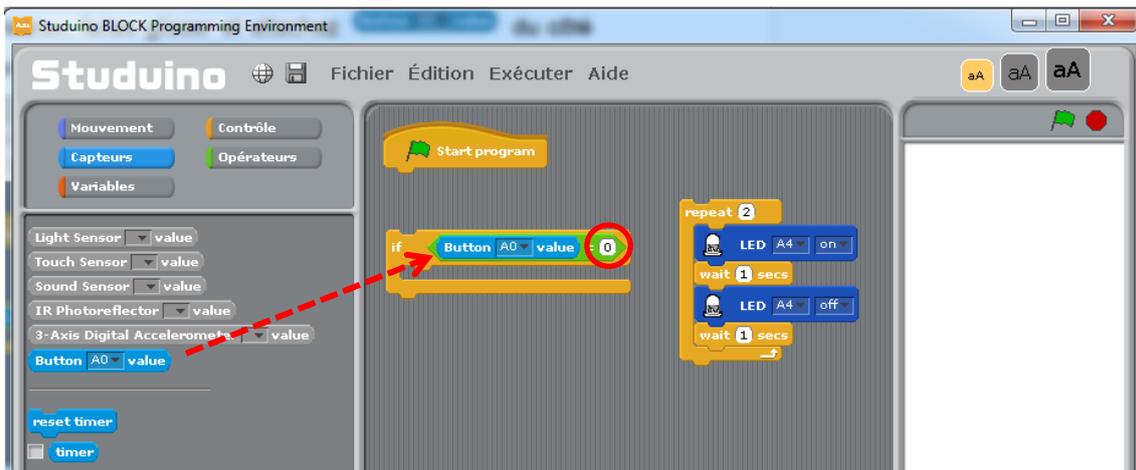
⑧ À partir de la palette «Contrôle», faites glisser  vers le champ de script.



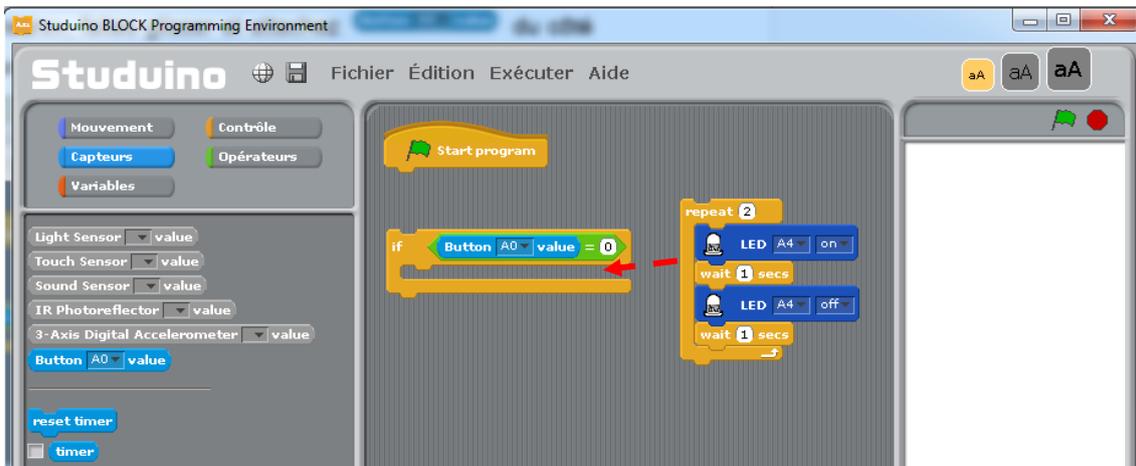
⑨ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



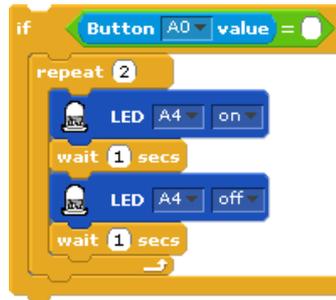
⑩ À partir de la palette «Capteurs», faites glisser et attachez  du côté gauche de . Ensuite, paramétrez le côté droit sur 0.



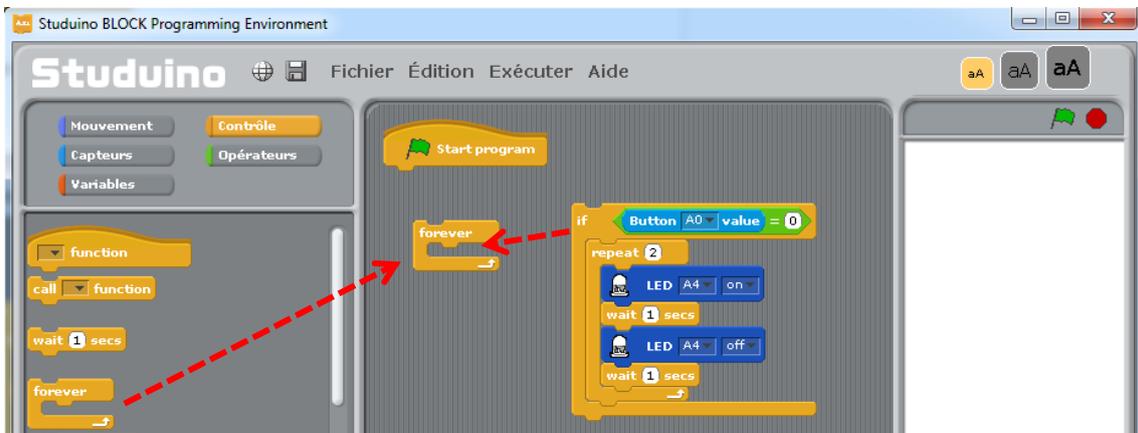
⑪ Déplacez les blocs de l'étape 6 au niveau du début de .



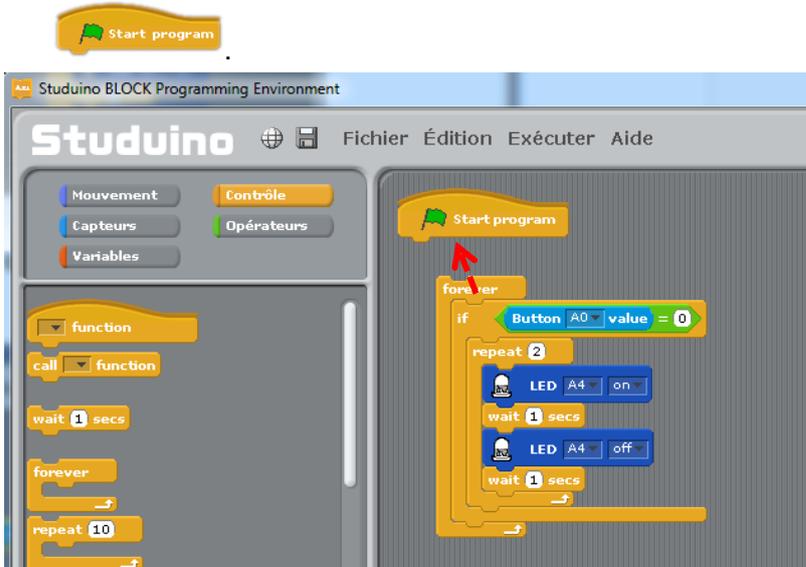
Ce groupement de blocs représente un programme qui permet à la LED de clignoter deux fois lorsque l'interrupteur à bouton-poussoir A0 est enfoncé (lorsque le bouton est en position 0).



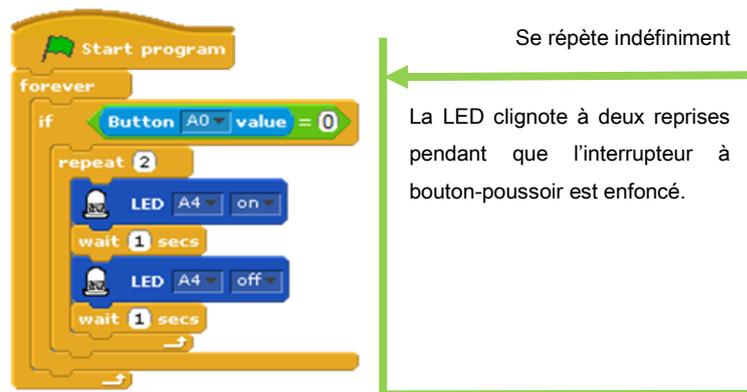
⑫ Faites glisser  vers le champ de script. Mettez en place les blocs de l'étape 10 dans le bloc .



⑬ Connectez le groupement de blocs que vous avez créé à l'étape ⑪ au bloc



Le programme est maintenant complet. Il fera clignoter la LED deux fois lorsque l'interrupteur à bouton-poussoir A0 est enfoncé.



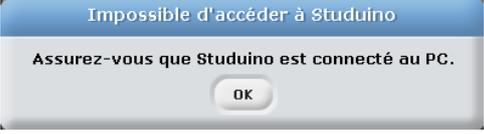
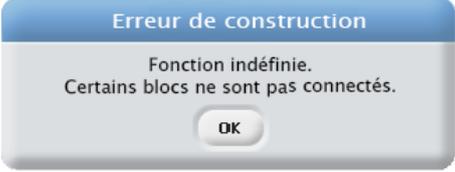
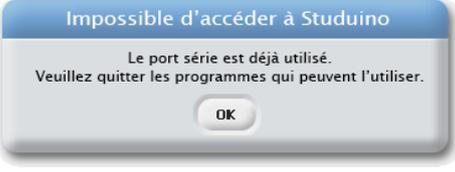
Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez «**Transférer**» depuis le menu «**Exécuter**» .



Si le message suivant apparaît et disparaît par la suite, cela signifie que le programme a été transféré avec succès au circuit. À la suite du transfert de données, la LED du circuit clignotera automatiquement à deux reprises après que l'interrupteur à bouton-poussoir A0 a été enfoncé.



L'affichage des messages ci-dessous indique que votre programme n'a pas été transféré avec succès. Testez les solutions proposées et essayez à nouveau.

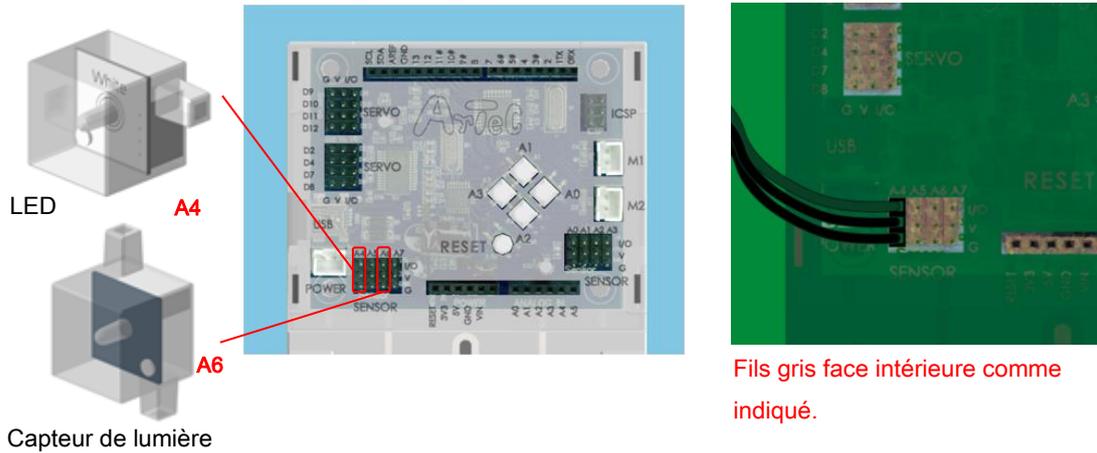
Message	Solution
	<p>Assurez-vous que le circuit du Studuino est correctement connecté à votre ordinateur par le câble USB.</p>
	<p>Supprimez tous les blocs qui ne sont pas attachés au bloc .</p>
	<p>Fermez toutes les applications qui peuvent entrer en communication avec votre Studuino. Si aucune autre application ne communique avec le circuit, débranchez le câble USB et appuyez sur le bouton de réinitialisation.</p>

4.2. Capteurs de lumière

Les capteurs de lumière sont utilisés pour détecter la luminosité.

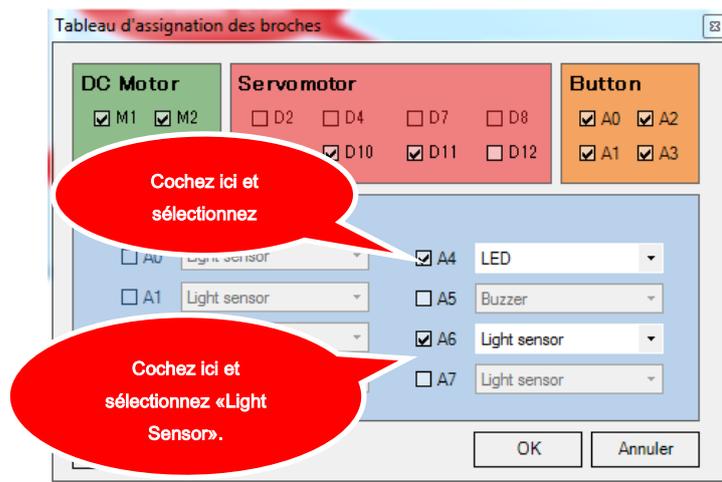
4.2.1. Connecter un capteur de lumière au circuit Studuino

Connectez votre capteur de lumière au connecteur A6 (capteur/ LED/ avertisseur sonore).



4.2.2. Paramétrage des ports

Avant de commencer à programmer, définissez les ports dans l'environnement de programmation par blocs pour qu'il s'accorde à votre circuit. Sélectionnez «**Édition**» dans le menu, puis dans le menu déroulant sélectionnez «**Paramètres des ports**» pour accéder à la boîte de dialogue des réglages. Dans la boîte de dialogue de réglage des ports, cochez A6 et choisissez «Light Sensor». Puis, cochez A4 et choisissez «LED» dans le menu déroulant.



4.2.3. Vérifier le capteur de lumière.

Trouvez la valeur du capteur de lumière. Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «**Exécuter**». Dans le mode test, le tableau des capteurs apparaît et indique "[A6] Light Sensor".

La valeur de la luminosité du capteur de lumière varie de 0 à 100 ; 0 correspond à la luminosité la plus sombre. Observez la valeur indiquée sur le tableau des capteurs pendant que vous faites varier la luminosité autour du capteur de lumière. Par exemple, faites de l'ombre au capteur et vérifiez sa valeur.

Si vous observez les valeurs varier, c'est que le capteur fonctionne convenablement. Maintenant, quittez le mode test en sélectionnant «**Désactiver le test**» depuis le menu «**Exécuter**».

Tableau des capteurs	
[A0] Button	1
[A1] Button	1
[A2] Button	1
[A3] Button	1
[A4] LED	0
[A5] Pas connecté	0
[A6] Light sensor	90
[A7] Pas connecté	0

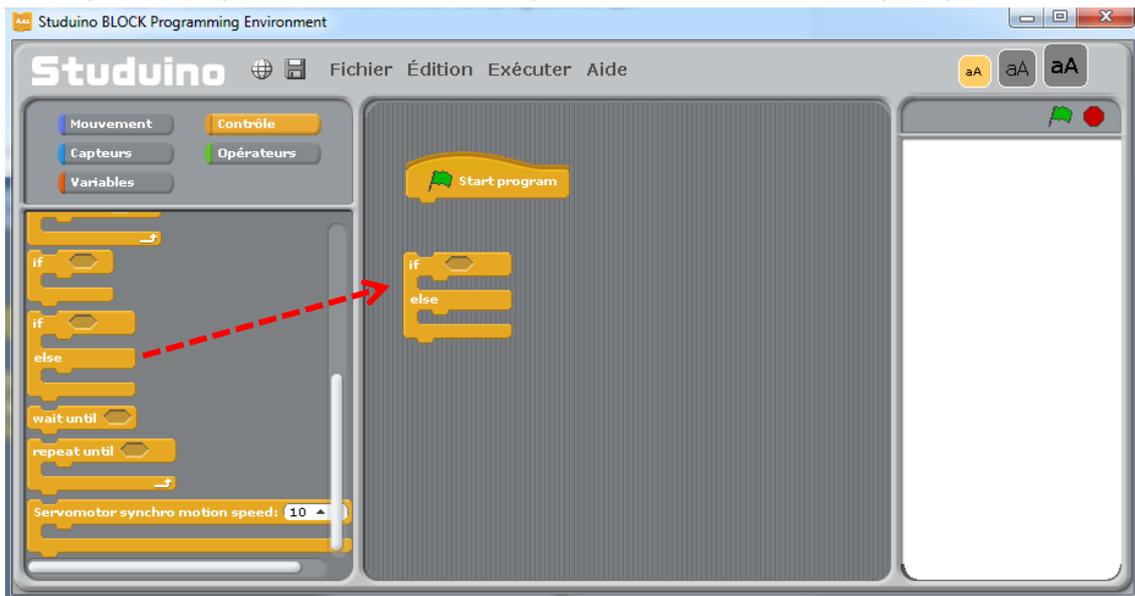
La valeur change en fonction de la luminosité autour du capteur de lumière.

4.2.4. Programmer

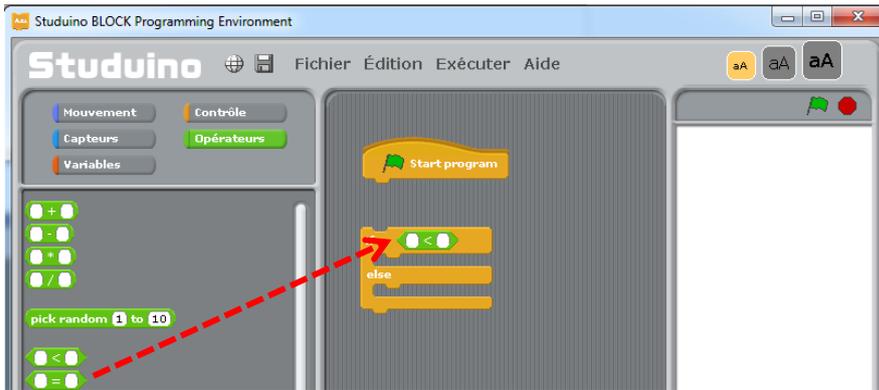
Maintenant, nous allons créer un programme qui va permettre à la LED de clignoter dans un environnement sombre. Nous apprendrons aussi comment créer un programme qui utilise le



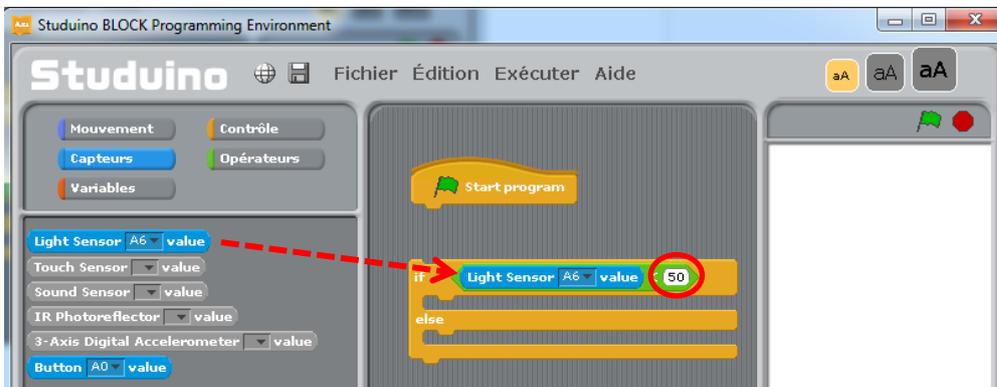
① À partir de la palette «Contrôle», faites glisser vers le champ script.



- ② À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.

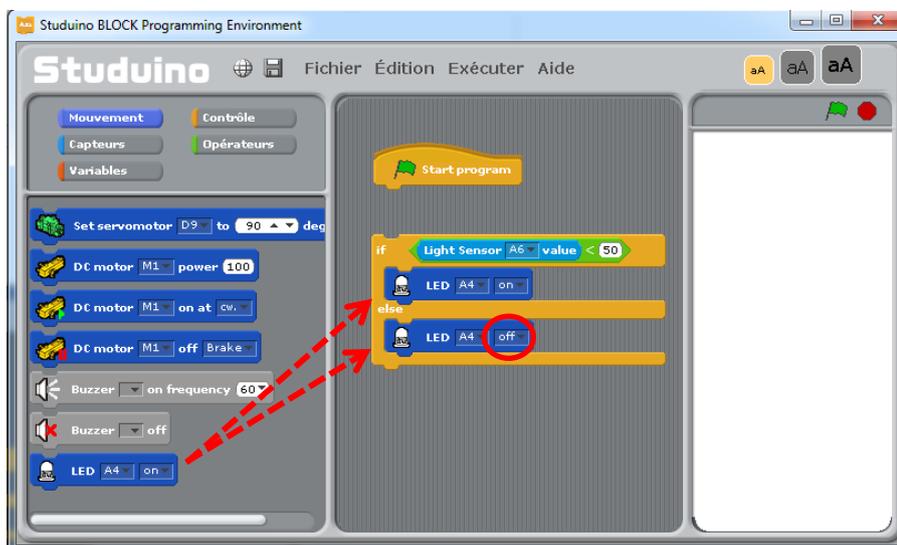


- ③ À partir de la palette «Capteurs», faites glisser  et attachez-le du côté gauche de . Paramétrez le côté droit sur 50.

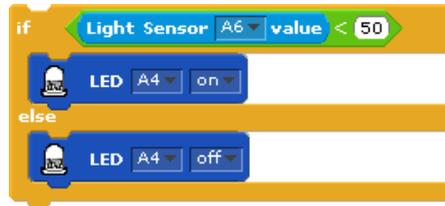


- ⚠ La valeur du capteur de lumière s'ajuste en fonction de la luminosité de la pièce. Paramétrez la valeur sur la droite en vous référant à celle indiquée par le capteur du circuit.

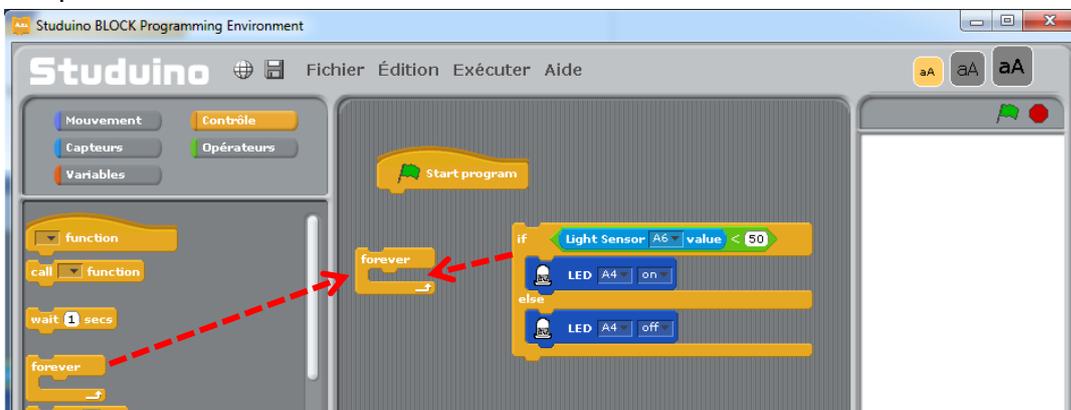
- ④ À partir de la palette «Mouvement», sélectionnez deux blocs  et placez-les dans les ouvertures de . Réglez le bloc  le plus bas sur OFF.



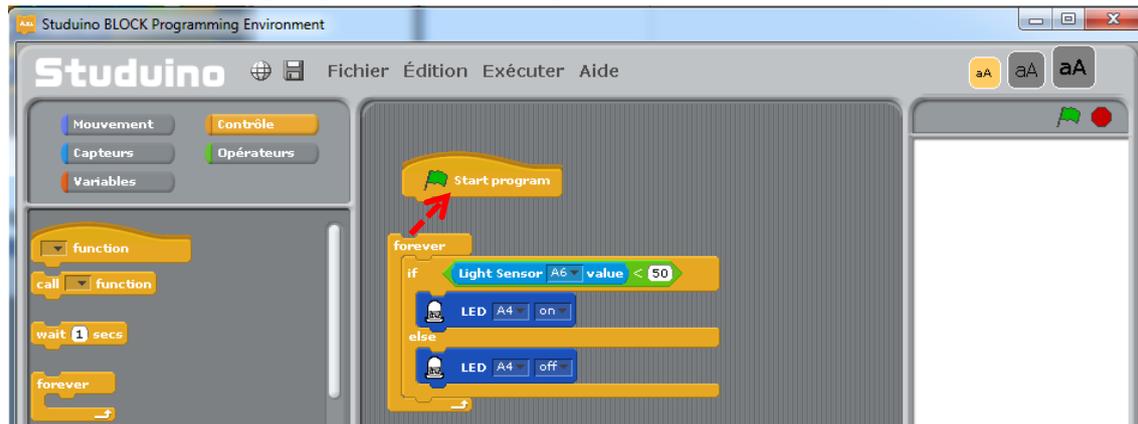
Ce groupe de blocs permettra à la LED de clignoter dans un environnement sombre (le capteur affichant une valeur inférieure à 50), et fera en sorte qu'elle ne clignote pas dans un environnement éclairé (le capteur affichant une valeur inférieure à 50).



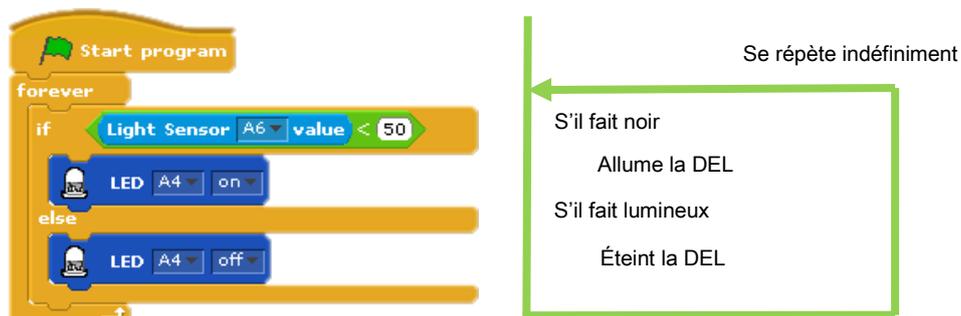
- ⑤ Faites glisser un bloc  vers le champ de script. Insérez les blocs de l'étape 4 dans le bloc .



- ⑥ Connectez le groupement de blocs que vous avez créé à l'étape 5 à .



Le programme est maintenant complet. Avec ce dernier, la LED clignotera lorsqu'il fera sombre et s'éteindra lorsqu'il fera clair.



- ⑦ Vérifiez si la LED s'allume ou s'éteint en fonction de la luminosité. Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «**Exécuter**».

- ⑧ Pendant que vous êtes en mode test, cliquez sur .



⚠ Ne déconnectez pas le câble USB pendant le mode test.

Les blocs utilisés par le programme sont indiqués par un contour blanc. Vérifiez si la LED clignote en faisant de l'ombre au capteur de lumière avec votre main.

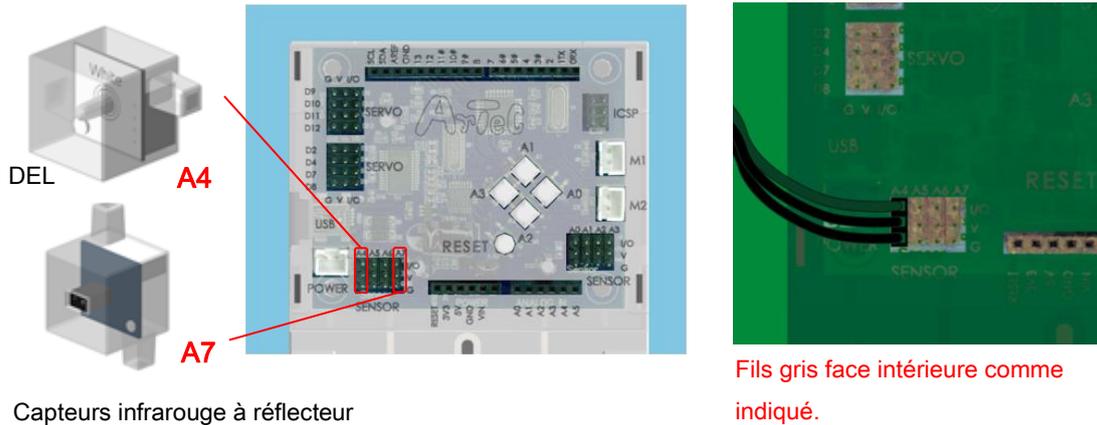
Depuis le menu, sélectionnez «Exécuter» puis «**Transférer**» afin que le programme soit transféré vers le circuit Studuino. Le programme peut désormais fonctionner sur batterie sans connexion USB à votre PC.

4.3. Capteurs infrarouges à réflecteur

Un capteur infrarouge à réflecteur permet de calculer la distance avec un objet. Pour ce faire, il émet une lumière infrarouge et mesure le résultat de la réflexion sur l'objet.

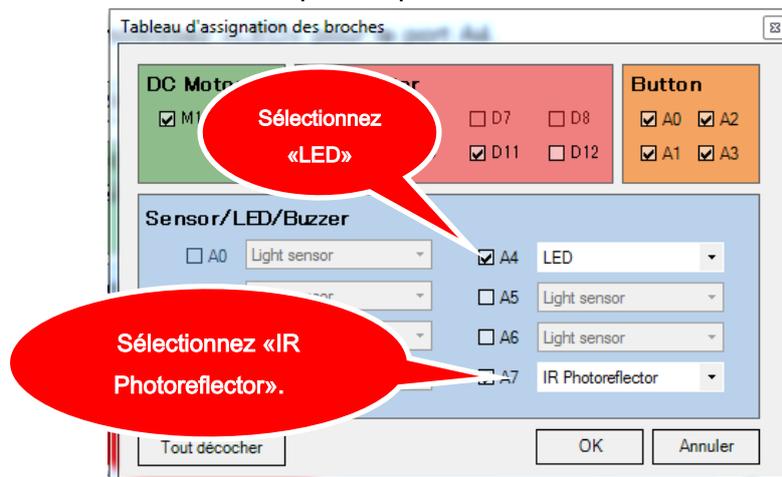
4.3.1. Connecter un capteur infrarouge au circuit Studuino

Connectez un capteur infrarouge à réflecteur au connecteur A7 (capteur/ LED/ avertisseur sonore) de votre circuit Studuino.



4.3.2. Paramétrage des ports

Paramétrez les ports dans l'environnement de programmation par blocs afin qu'il s'accorde au circuit. Cliquez «Édition» dans le menu, puis dans le menu déroulant sélectionnez «Paramètres des ports» pour accéder à la boîte de dialogue de réglages. Dans la boîte de dialogue de réglage des ports, cochez A7 et choisissez «IR Photoreflector». Choisissez «LED» pour le port A4.



4.3.3. Vérifier le capteur infrarouge à réflecteur

Maintenant, nous allons déterminer la valeur de votre capteur infrarouge à réflecteur. Connectez le circuit Studuino au PC en utilisant le câble USB. À partir du menu **Exécuter** choisissez **Activer le test**.

Dans le mode test, le tableau des capteurs apparaît et indique **[A7] IR Photoreflector**. La valeur du capteur représente la distance entre le capteur et un objet. Elle se situe entre 0 et 100. Plus la distance est courte, et plus le nombre sera grand. Faites bouger l'objet et vérifiez si la valeur indiquée sur le circuit varie bien en fonction de la distance entre le capteur et ce dernier.

Tableau des capteurs	
[A0] Button	1
[A1] Button	1
[A2] Button	1
[A3] Button	1
[A4] LED	0
[A5] Pas connecté	0
[A6] Pas connecté	0
[A7] IR Photoreflector	11

La valeur varie en fonction de la distance entre le capteur et l'objet.

Si vous observez les valeurs varier, c'est que le capteur fonctionne convenablement. Maintenant, quittez le mode test en sélectionnant **«Désactiver le test»** depuis le menu **«Exécuter»**.

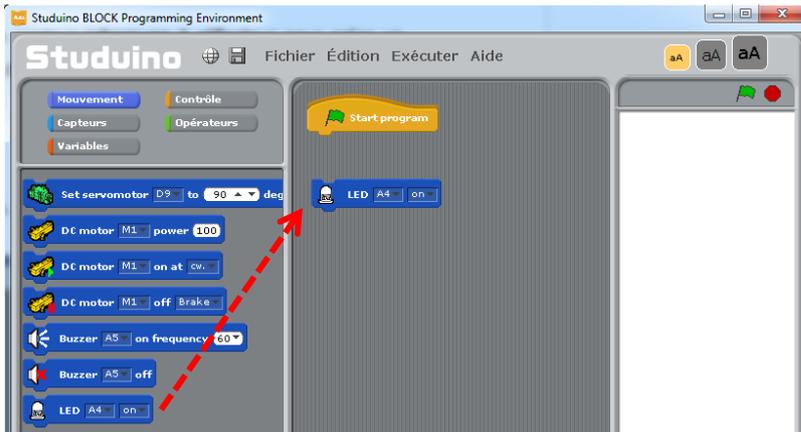
4.3.4. Programmer

Maintenant, nous allons utiliser le capteur infrarouge à réflecteur pour créer un programme qui va permettre à la LED de clignoter lorsqu'un objet se rapproche d'elle.

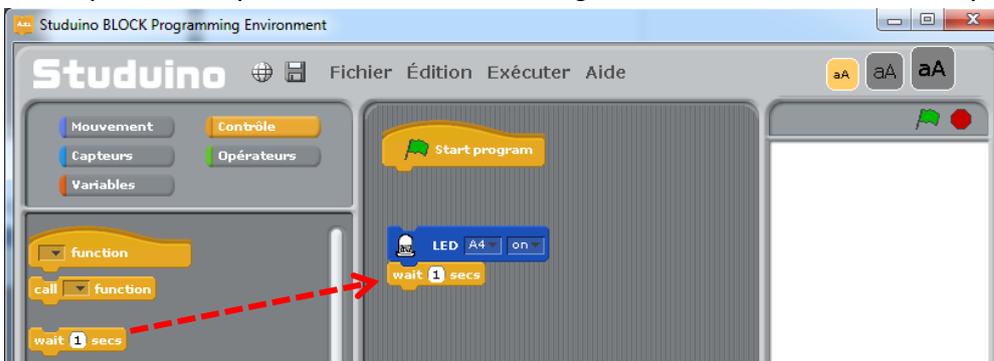
Nous apprendrons aussi comment créer un programme qui utilise le bloc



- ① À partir de la palette «Mouvement», faites glisser  vers le champ de script.



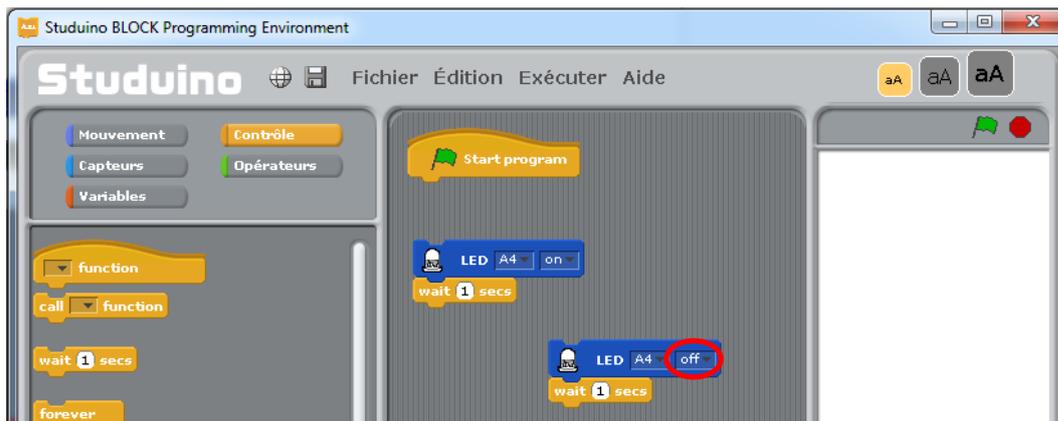
- ② À partir de la palette «Contrôle», faites glisser  vers le champ de script.



- ③ Faites un clic droit sur  et sélectionnez «Dupliquer».



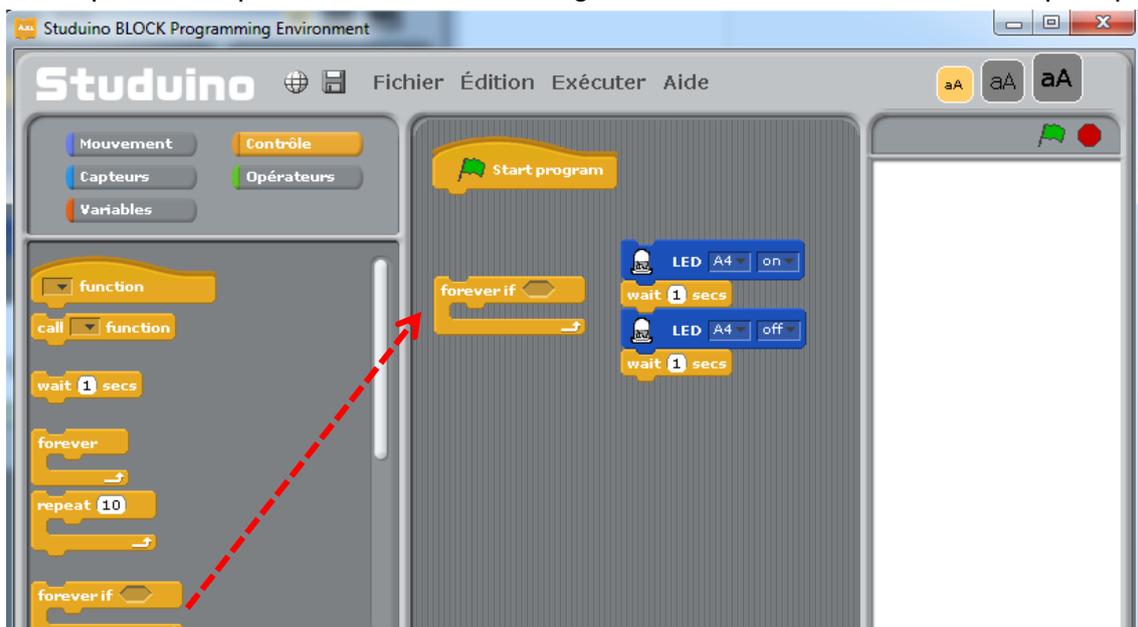
- ④ Réglez le bloc dupliqué sur «off».



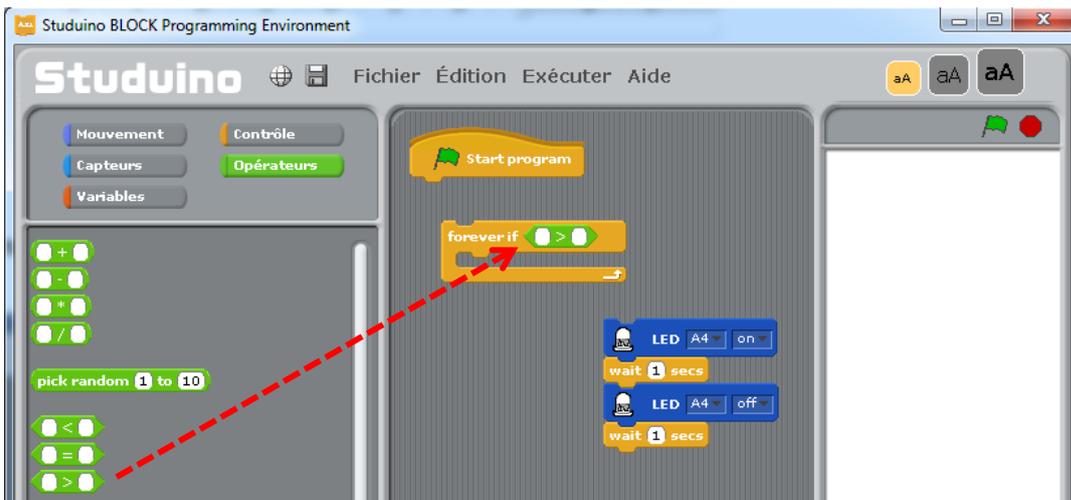
- ⑤ Connectez les blocs dupliqués aux blocs d'origine.



- ⑥ À partir de la palette «Contrôle», faites glisser  vers le champ script.

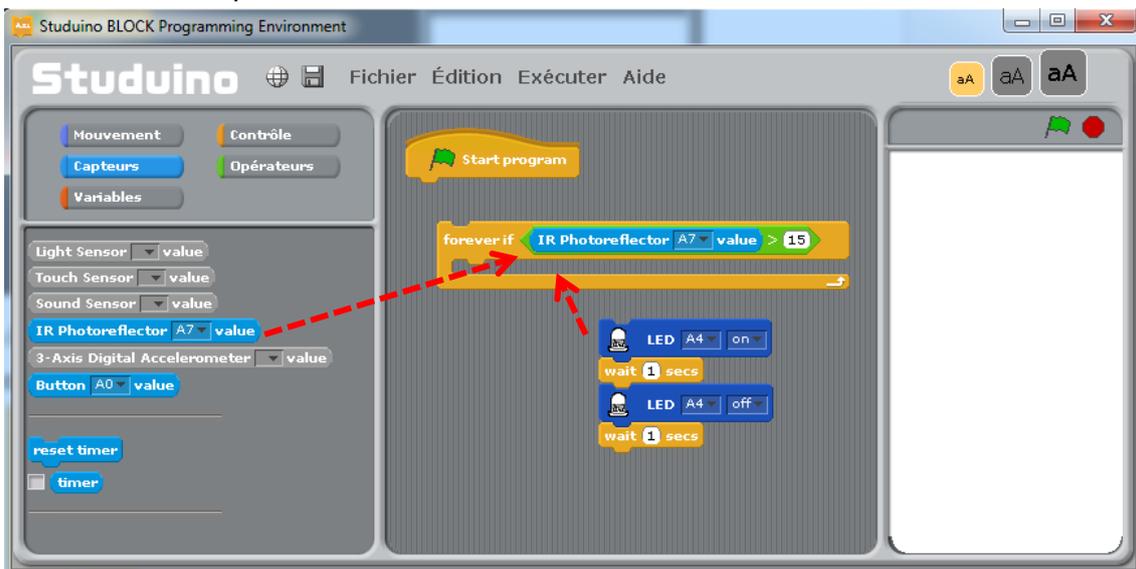


⑦ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



⑧ Faites glisser un bloc  depuis la palette «Capteurs » et attachez-le du côté gauche de . Paramétrez le côté droit sur 15. Placez les

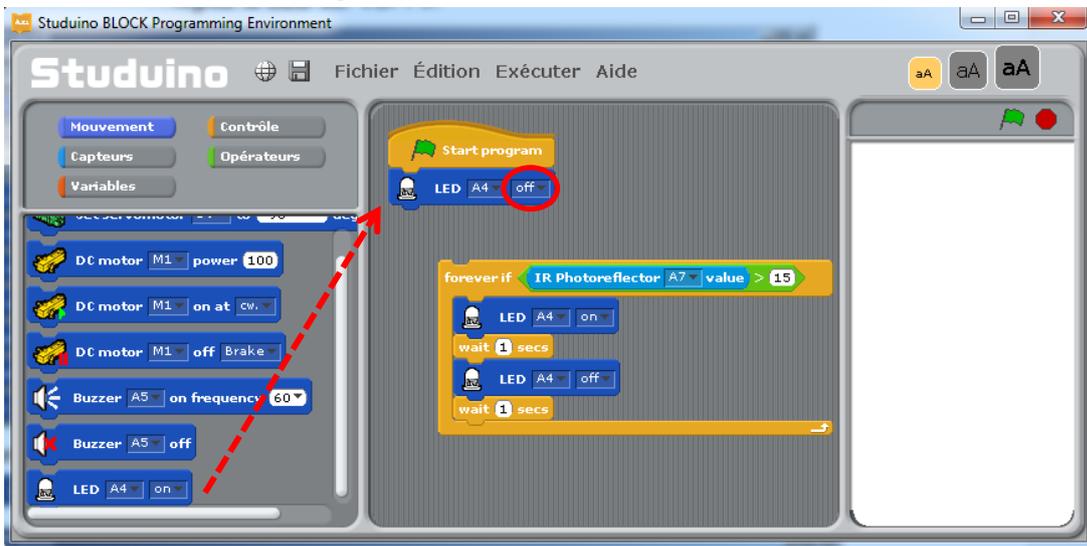
blocs de l'étape 4 dans



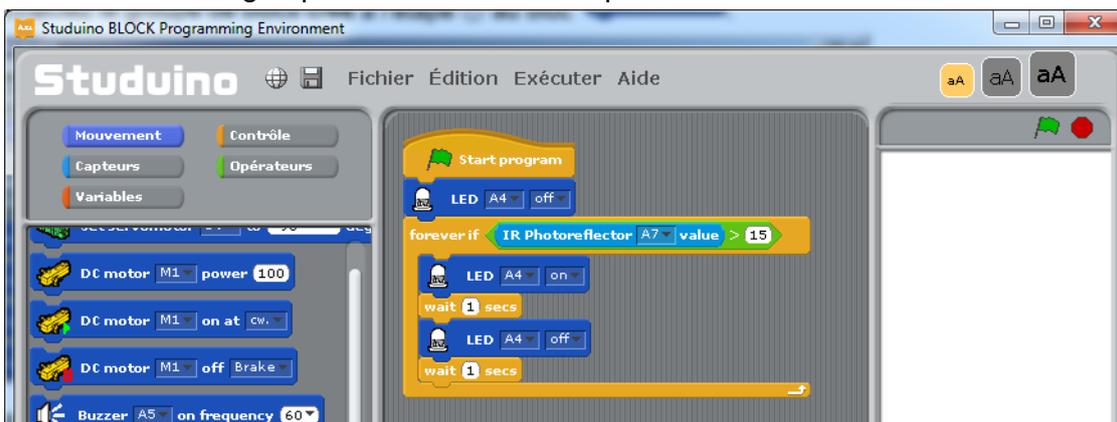
Ce groupe de blocs permet à la LED de clignoter lorsqu'un objet se trouve à proximité (la valeur du capteur est supérieure à 15).



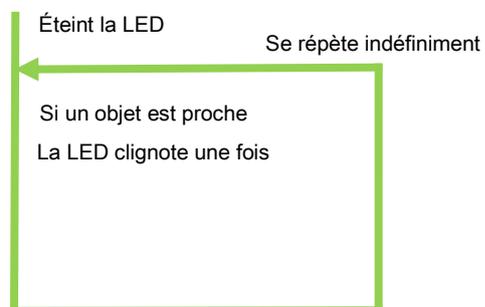
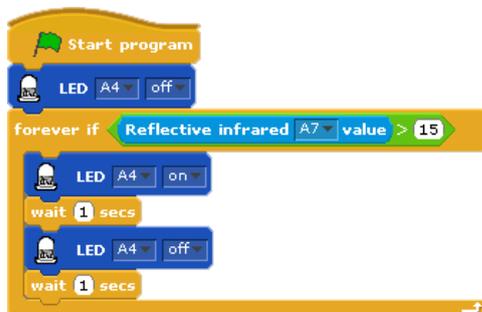
- ⑨ À partir de la palette «Mouvement», faites glisser et attachez  à . Réglez la LED sur «OFF».



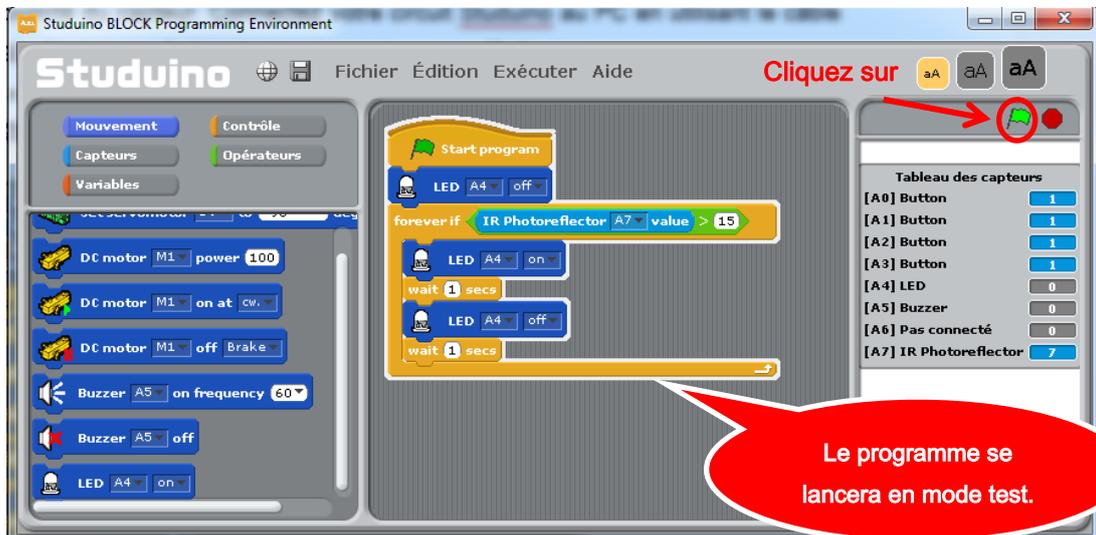
- ⑩ Connectez le groupe de blocs créé à l'étape ⑦ au bloc .



Ce programme fera clignoter la LED lorsqu'un objet sera à proximité.



- ⑪ Vérifiez si le programme fonctionne convenablement en plaçant un objet à proximité du capteur. Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «Exécuter».
- ⑫ Pendant que vous êtes en mode test, cliquez sur .



 **Ne déconnectez pas le câble USB pendant le mode test.**

Faites bouger l'objet et vérifiez si la LED clignote lorsque la valeur indiquée par le capteur devient supérieure à 15.

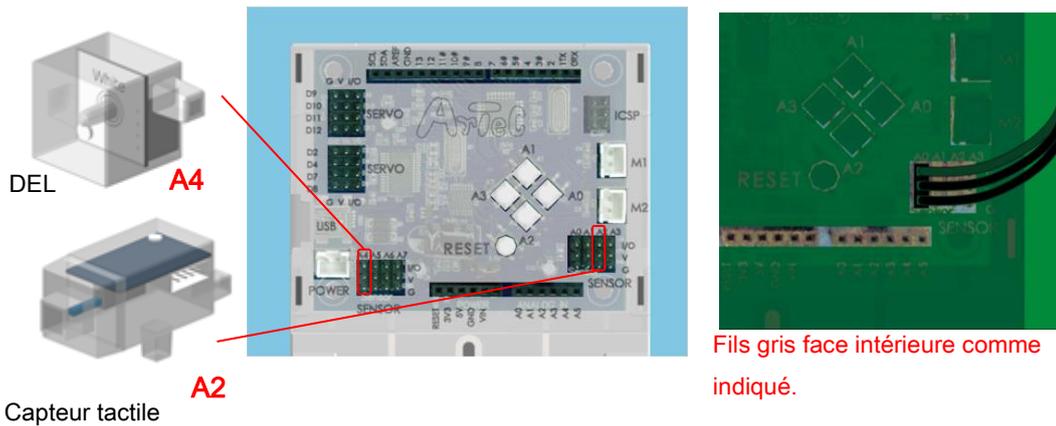
À partir du menu, sélectionnez «Exécuter» puis «Transférer» afin que le programme soit transféré vers le circuit Studuino. Le programme peut désormais fonctionner sur batterie sans la connexion USB à votre PC.

4.4. Capteurs tactiles

Un capteur tactile permet de détecter si un bouton est enfoncé. Il permet de juger certaines choses comme par exemple l'entrée en contact d'objets.

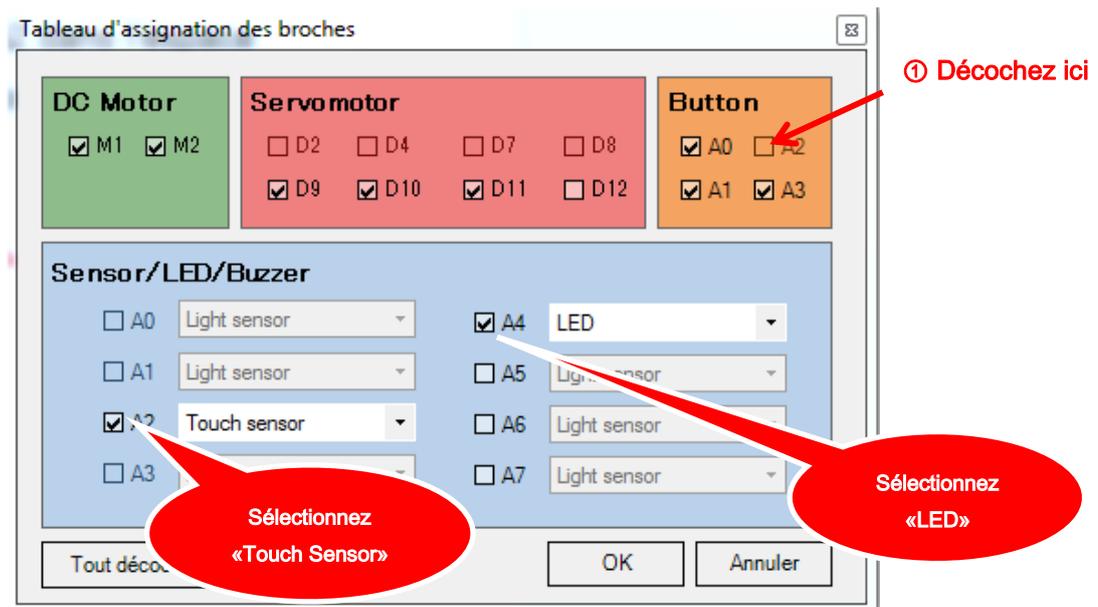
4.4.1. Connecter un capteur tactile au circuit Studuino

Connectez un capteur tactile au connecteur A2 (capteur/ LED/ avertisseur sonore) sur le circuit Studuino.



4.4.2. Paramétrage des ports

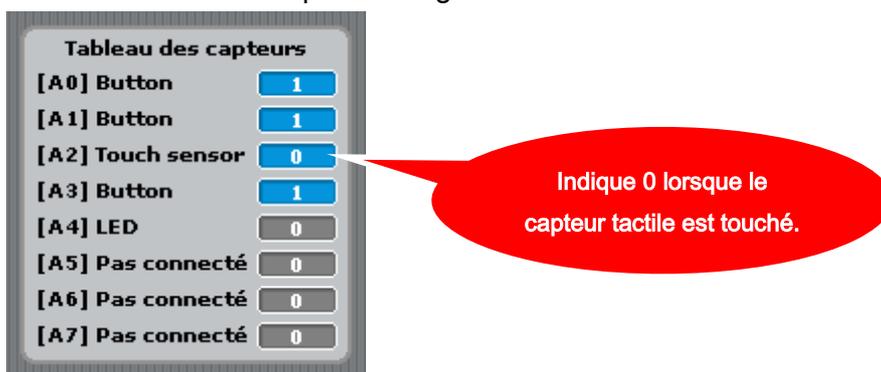
Paramétrez les ports dans l'environnement de programmation par blocs afin qu'il s'accorde à votre circuit. Sélectionnez «Édition» dans le menu, puis dans le menu déroulant sélectionnez «Paramètres des ports» pour accéder à la boîte de dialogue de réglages. Dans la boîte de dialogue de réglage des ports, décochez A2 dans l'espace «Button», cochez A2 dans l'espace «Sensor/LED/Buzzer», puis choisissez «Touch Sensor». Cochez A4 et sélectionnez «LED».



4.4.3. Tester le capteur tactile

Maintenant, nous allons déterminer la valeur de votre capteur tactile. Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «**Exécuter**». Dans le mode test, le tableau des capteurs apparaît et indique "[A2] Touch Sensor."

Le capteur tactile envoie une valeur égale à 1 à son état normal et égale à 0 lorsque le bouton est enfoncé. Touchez le capteur tactile et vérifiez si la valeur indiquée dans le tableau des capteurs réagit.

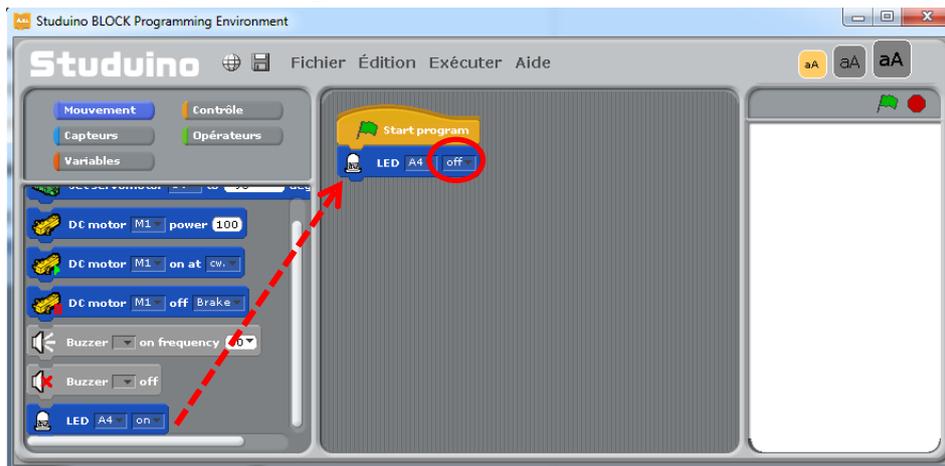


Si les valeurs répondent correctement, alors le circuit fonctionne convenablement. Maintenant, quittez le mode test en sélectionnant «**Désactiver le test**» depuis le menu «**Exécuter**».

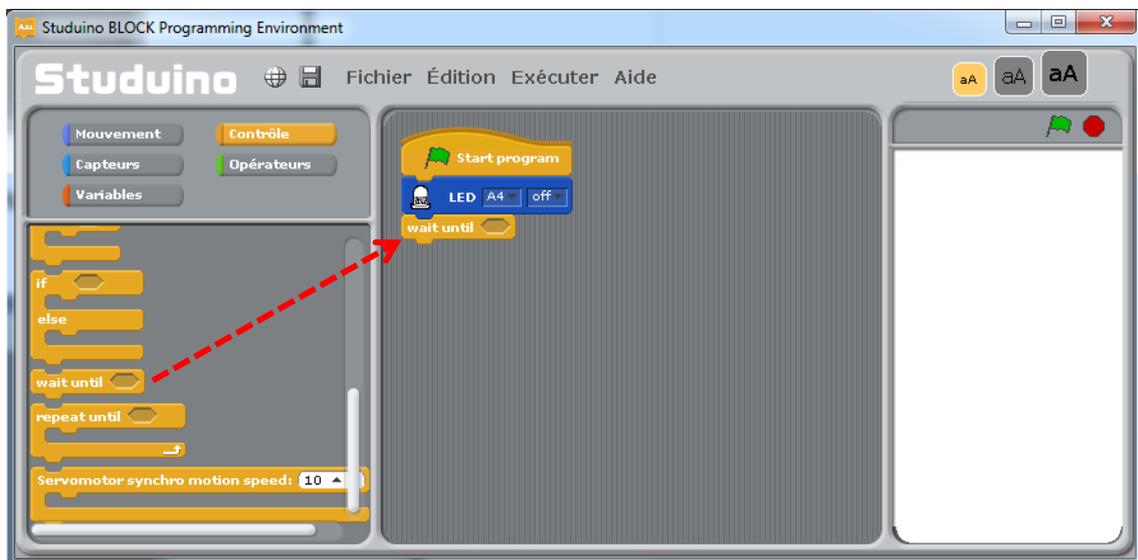
4.4.4. Programmer

Maintenant, nous allons utiliser le capteur tactile pour créer un programme qui va permettre à la LED de clignoter lorsque le capteur tactile est pressé. Nous apprendrons aussi comment créer un programme qui utilise **wait until**.

- ① À partir de la palette «Mouvement», faites glisser et attachez **LED A4 on** à **Start program**. Réglez la LED sur «OFF».



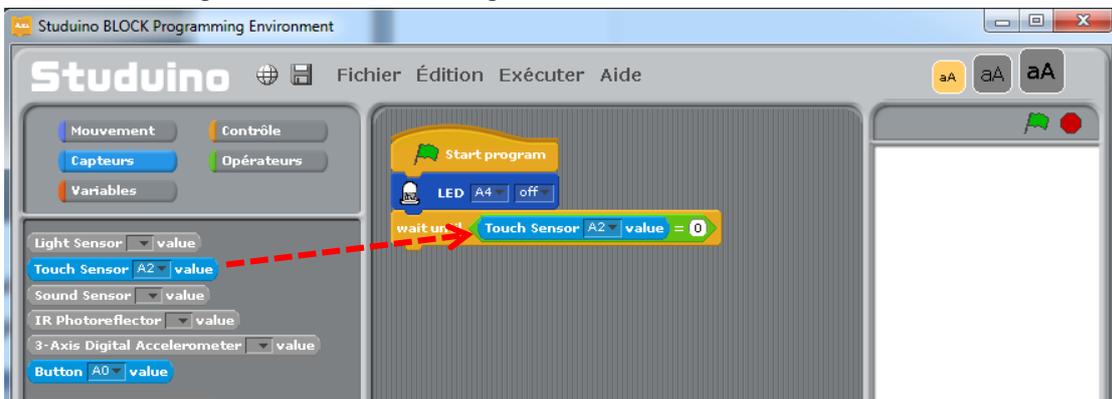
- ② À partir de la palette «Contrôle», faites glisser et attachez **wait until** à **LED A4 off**.



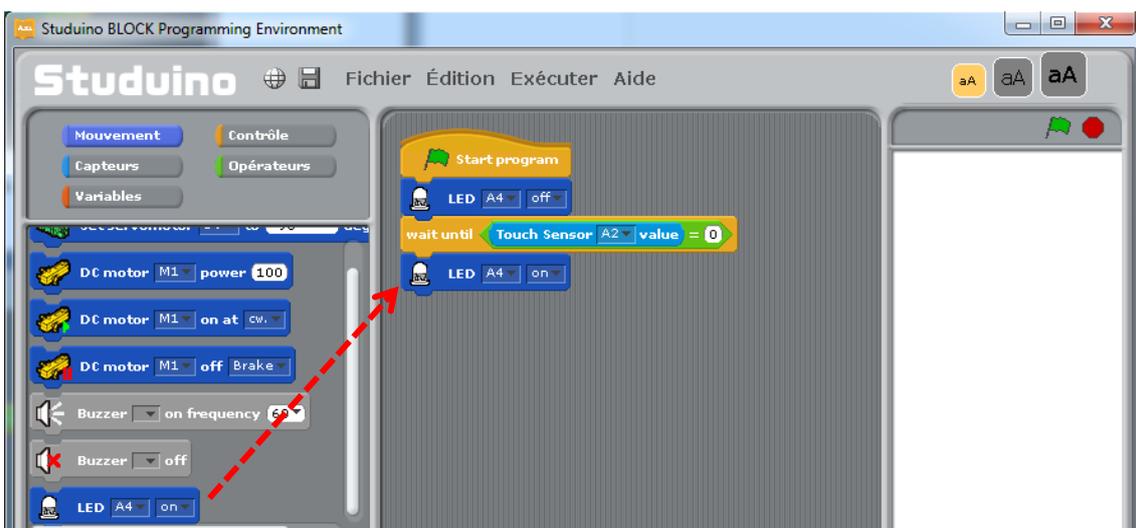
- ③ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



- ④ À partir de la palette «Capteurs», faites glisser  et attachez-le du côté gauche de . Réglez le côté droit sur 0.

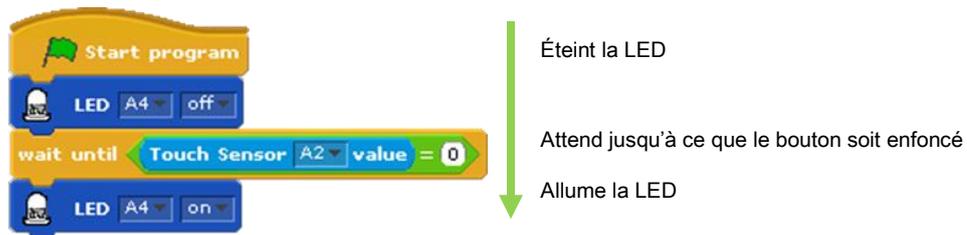


- ⑤ Faites glisser  de la palette «Mouvement» et attachez-le aux autres blocs.



Le programme est maintenant complet. Il fera clignoter la LED deux fois lorsque

le capteur tactile sera touché.



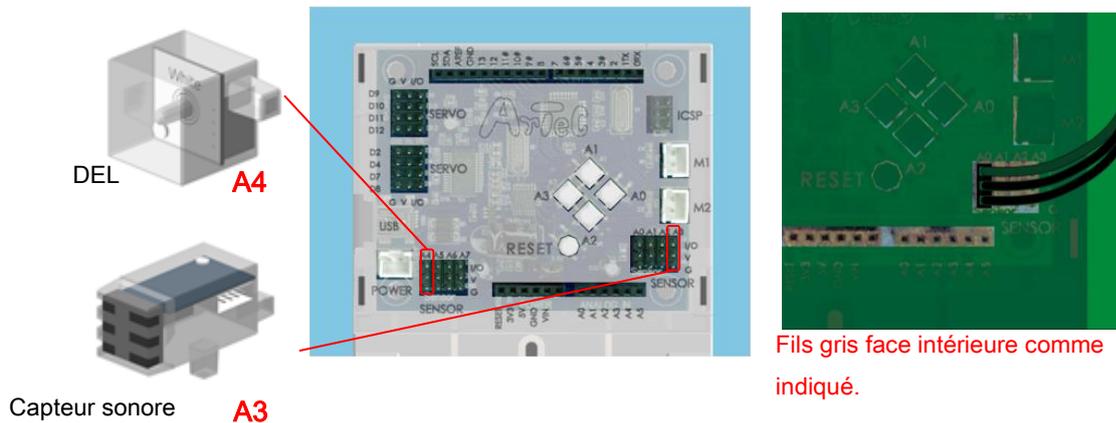
Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Transférer** depuis le menu **Exécuter**. Vérifiez si la LED s'allume bien lorsque le capteur tactile est touché.

4.5. Capteurs sonores

Les capteurs sonores mesurent le niveau sonore d'un son.

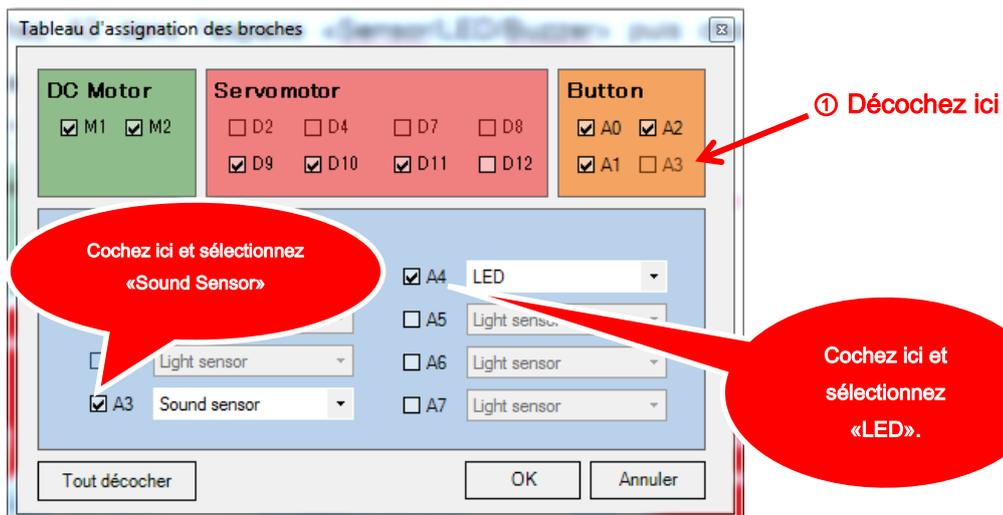
4.5.1. Connecter un capteur sonore au circuit Studuino

Connectez un capteur sonore au connecteur A3 (capteur/ LED/ avertisseur sonore) sur le circuit Studuino.



4.5.2. Paramétrage des ports

Paramétrez les ports dans l'environnement de programmation par blocs afin qu'il s'accorde à votre circuit. Sélectionnez «Édition» dans le menu, puis dans le menu déroulant sélectionnez «Paramètres des ports» pour accéder à la boîte de dialogue de réglages. Dans la boîte de dialogue de réglage des ports, décochez A3 dans l'espace «Button», cochez A3 dans l'espace «Sensor/LED/Buzzer» puis choisissez «Sound Sensor». Cochez A4 et sélectionnez «LED».



4.5.3. Tester le capteur sonore

Maintenant, nous allons déterminer la valeur de votre capteur sonore.

Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu «**Exécuter**». Dans le mode test, le circuit du capteur apparait et indique "[A3] Sound Sensor."



Les valeurs doivent varier en fonction de l'intensité du niveau sonore.

La valeur du capteur sonore représente le volume d'un son et varie de 0 à 50. Dans un environnement sans bruit, la valeur affichée indique 0. Plus le volume sonore augmente et plus grande sera la valeur indiquée. Essayez de souffler sur le capteur sonore ou de faire d'autres sons et vérifiez si les valeurs changent dans le tableau des capteurs.

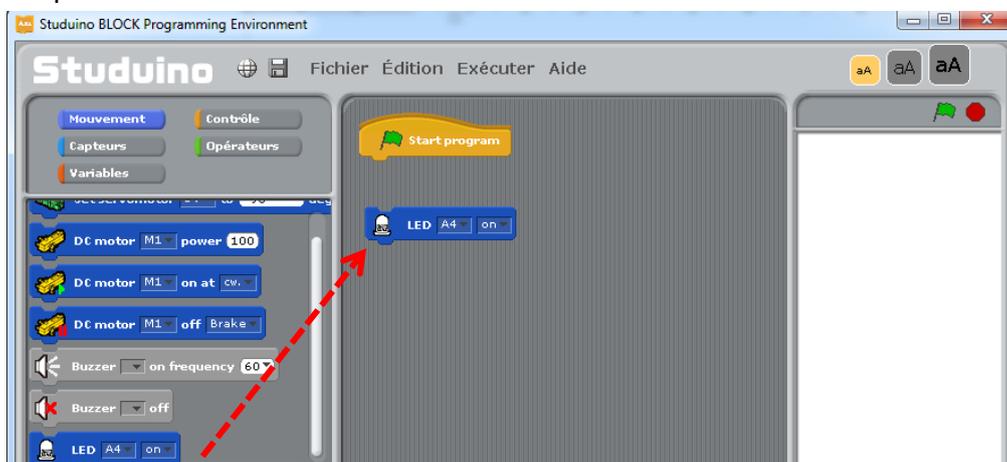
Si les valeurs répondent correctement, alors le circuit fonctionne convenablement. Maintenant, quittez le mode test en sélectionnant «**Désactiver le test**» depuis le menu «**Exécuter**».

4.5.4. Programmer

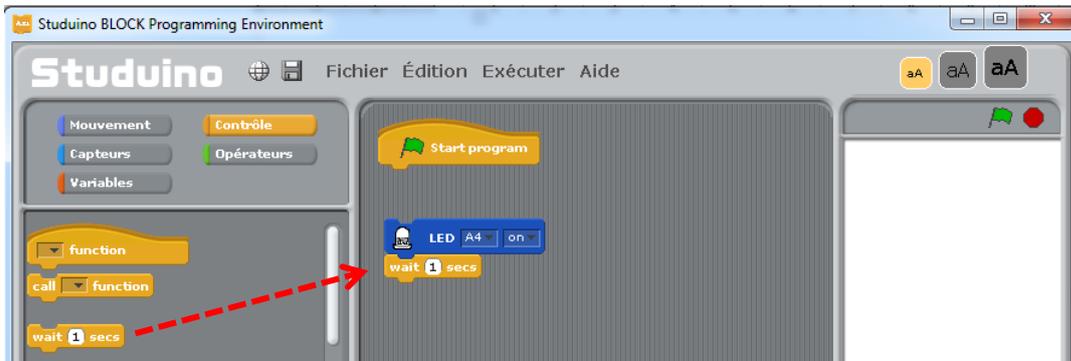
Maintenant, nous allons utiliser le capteur sonore pour créer un programme qui va permettre à la LED de s'arrêter de clignoter si vous soufflez sur le capteur sonore.

Nous utiliserons le bloc  .

① À partir de la palette «Mouvement», faites glisser  vers le champ script.



- ② À partir de la palette «Contrôle», faites glisser et attachez  au bloc de l'étape 1.



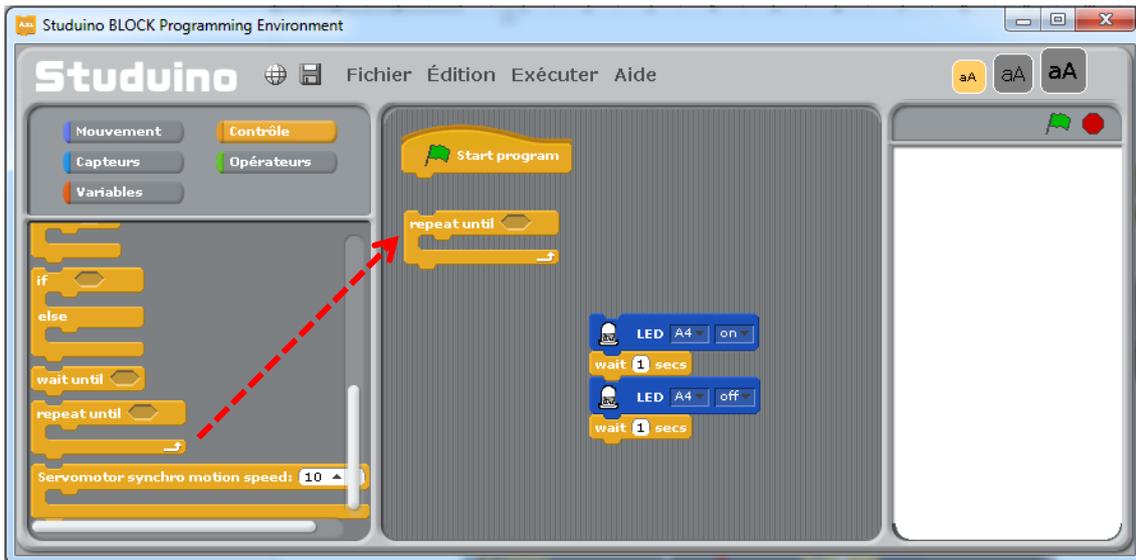
- ③ Faites un clic droit sur  et sélectionnez «Dupliquer».



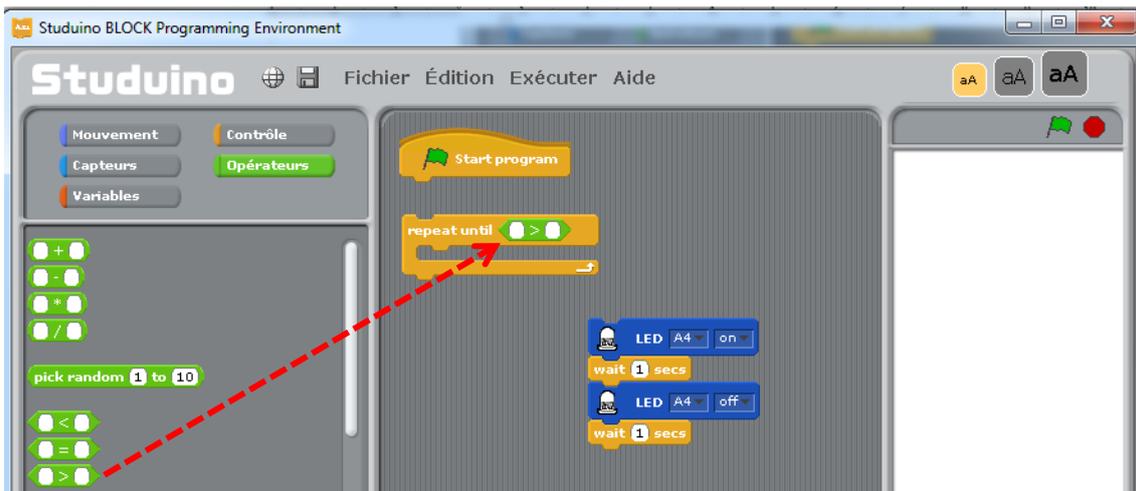
- ④ Connectez les blocs dupliqués aux blocs d'origine.



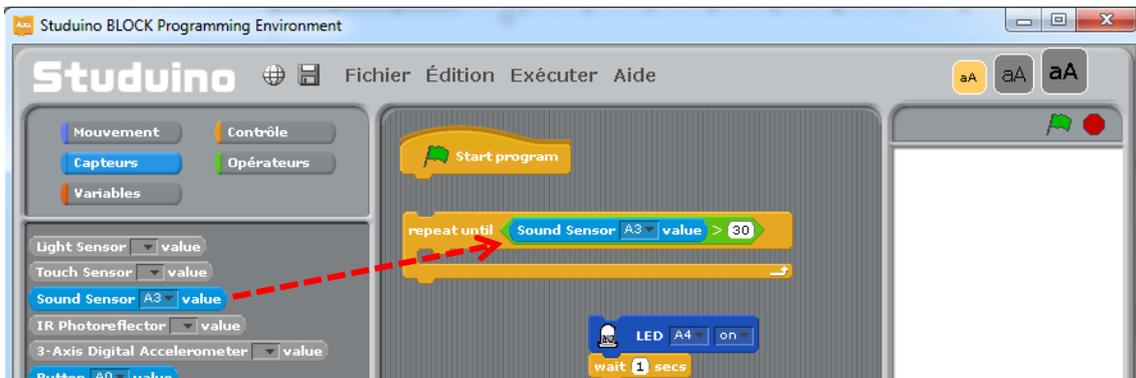
- ⑤ À partir de la palette «Contrôle», faites glisser  vers le champ script.



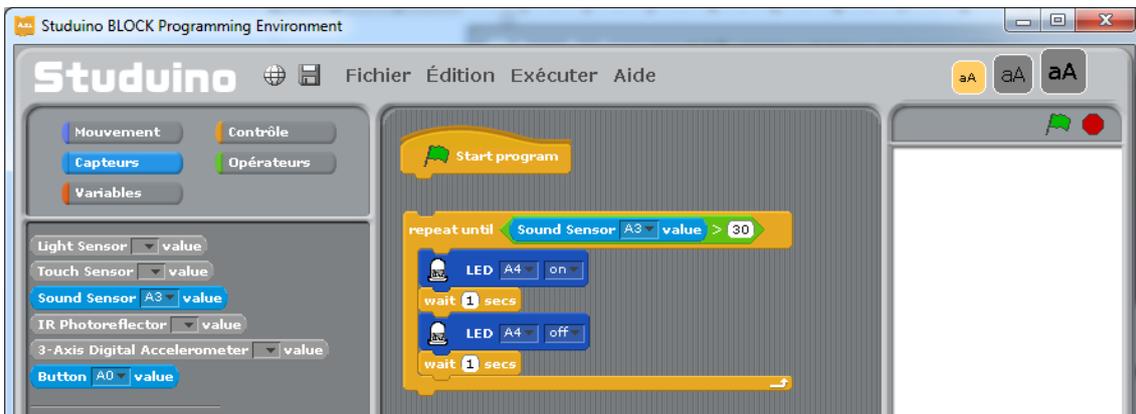
- ⑥ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



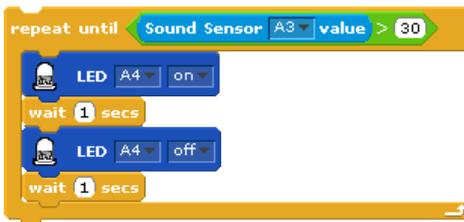
- ⑦ À partir de la palette «Capteurs », faites glisser et attachez  du côté gauche de . Paramétrez le côté droit sur 30.



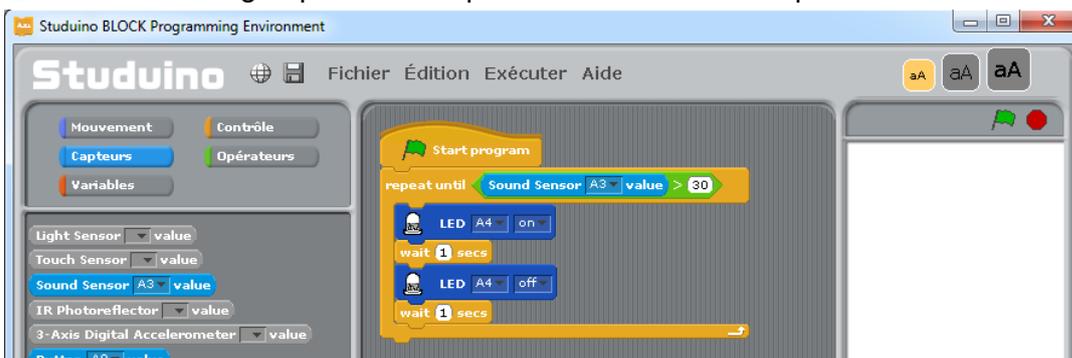
⑧ Déplacez les blocs de l'étape 4 dans l'ouverture



Ce programme fera en sorte qu'une LED qui clignote en boucle s'arrête de clignoter si vous soufflez sur le capteur sonore (lorsque la valeur indiquée par le capteur devient supérieure à 30).

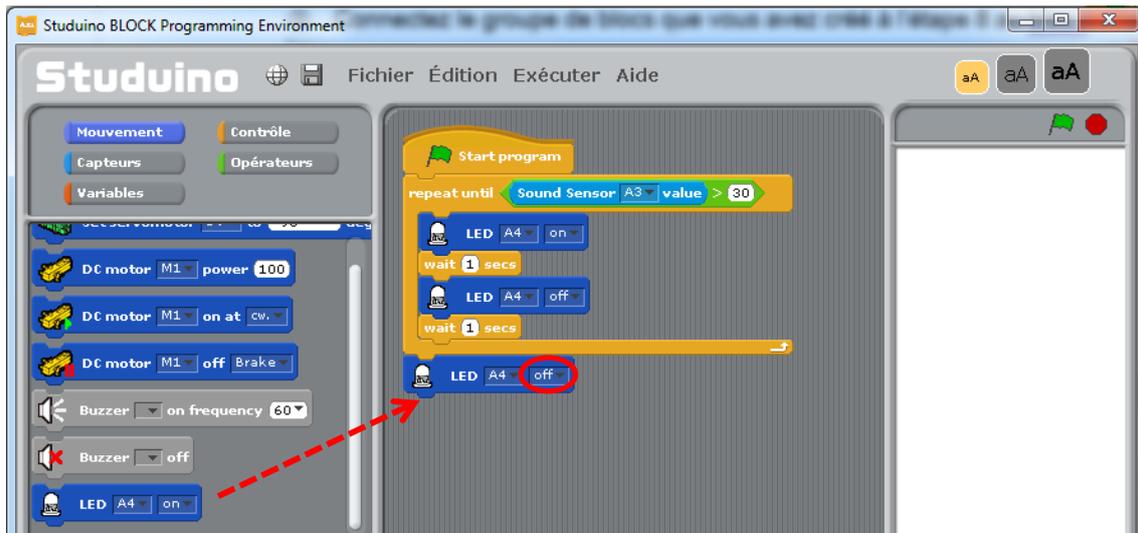


⑨ Connectez le groupe de blocs que vous avez créé à l'étape 8 au bloc

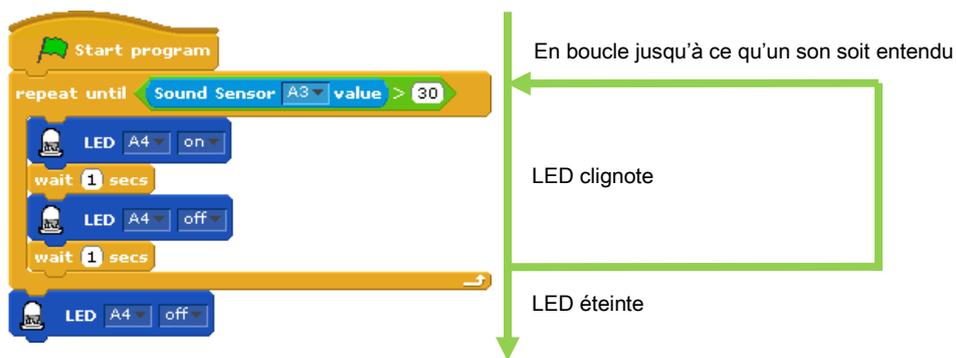


⑩ À partir de la palette «Mouvement», faites glisser et attachez Paramétrez la LED sur «OFF».



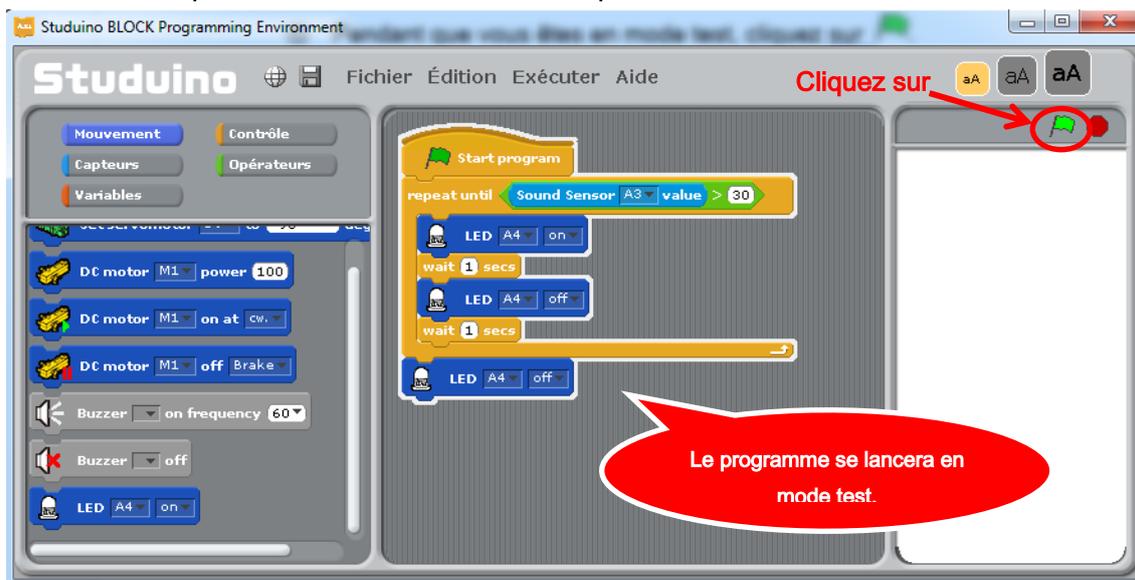


Nous venons de réaliser un programme qui permet d'arrêter qu'une LED clignote lorsque l'on souffle sur le capteur sonore.



- ⑪ Vérifiez bien que la LED s'éteint lorsque vous soufflez sur le capteur sonore. Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **Activer le test** depuis le menu « **Exécuter** ».

⑫ Pendant que vous êtes en mode test, cliquez sur .



 **Ne déconnectez pas le câble USB pendant le mode test.**

Soufflez sur le capteur sonore et vérifiez que la LED s'éteigne lorsque la valeur indiquée par le capteur sonore indique plus de 30.

À partir du menu, sélectionnez «Exécuter» puis «**Transférer**» afin que le programme soit transféré vers le circuit Studuino. Le programme peut désormais fonctionner sur batterie sans la connexion USB à votre PC.

5. Moteurs à courant continu

5.1. Contrôler un moteur à courant continu

Dans cette section, vous allez apprendre comment configurer et contrôler un moteur à courant continu par l'intermédiaire d'un programme que vous aurez élaboré.

5.1.1. Connecter un moteur à courant continu au circuit Studuino

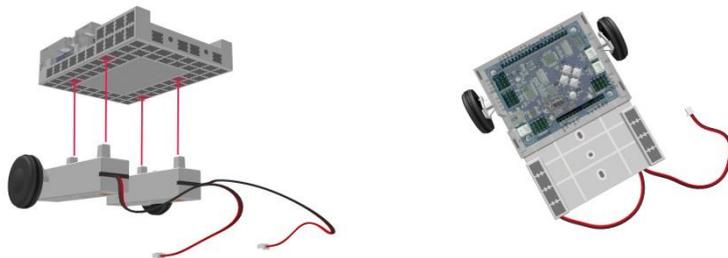
Assemblez la voiture comme indiqué ci-dessous.

(1) Attachez les roues au moteur à courant continu comme indiqué ci-dessous.

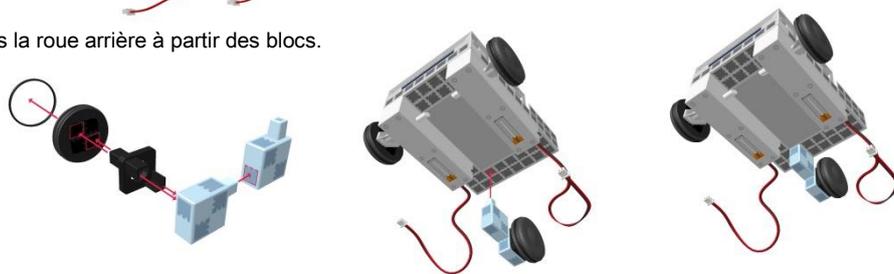
★ Faites en sorte que la roue droite et gauche soient symétriques.



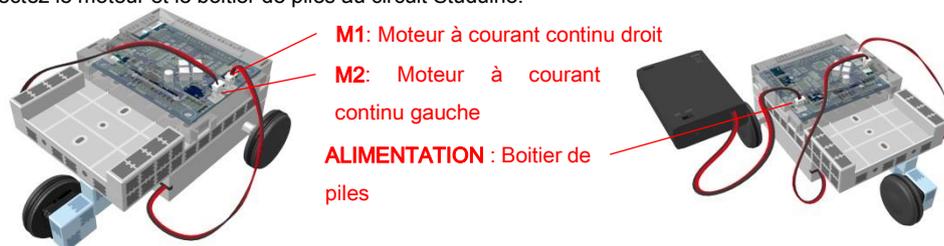
(2) Attachez les moteurs à courant continu au bas du circuit.



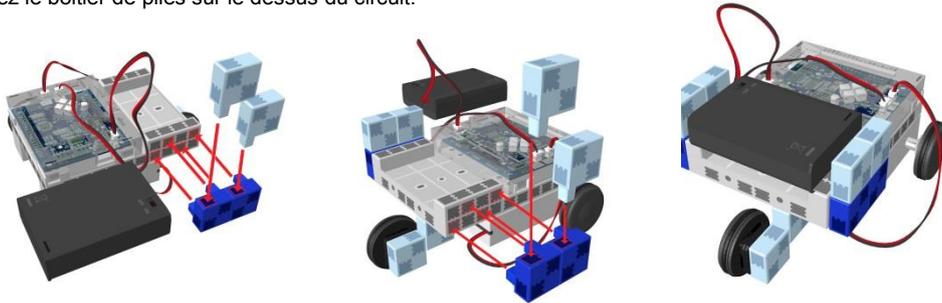
(3) Faites la roue arrière à partir des blocs.



(4) Connectez le moteur et le boîtier de piles au circuit Studuino.

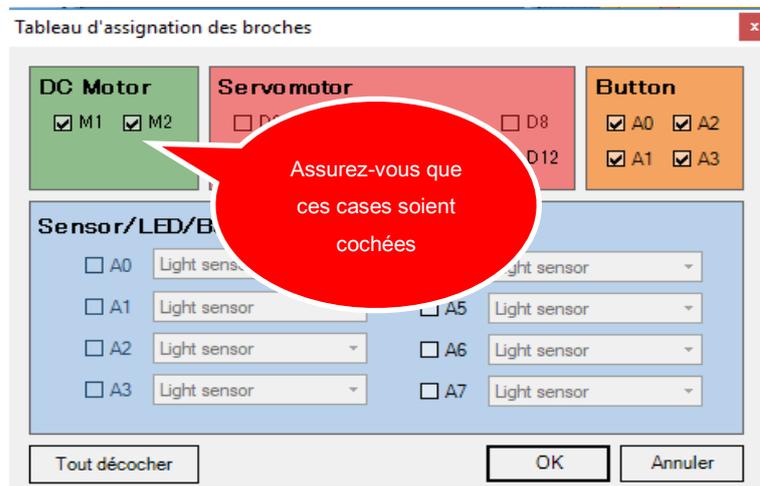


(5) Placez le boîtier de piles sur le dessus du circuit.



5.1.2. Paramétrage du port

Avant de programmer, vous aurez besoin de configurer les ports dans Scratch afin qu'il s'accorde aux pièces connectées à votre Studuino. Sélectionnez **«Édition»** dans le menu, puis sélectionnez **«Paramètres des ports»** pour accéder à la boîte de dialogue de réglages. Assurez-vous que M1 et M2 soient bien cochés dans l'espace «DC Motor».



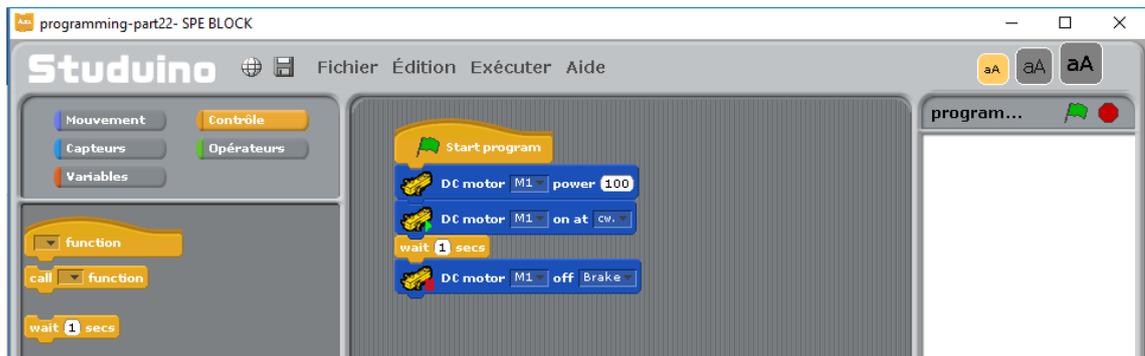
Programmer

Maintenant, nous allons créer un programme qui va permettre de contrôler le moteur à courant continu connecté à M1 et de changer les paramètres durant le mode test pour voir comment le moteur répond.

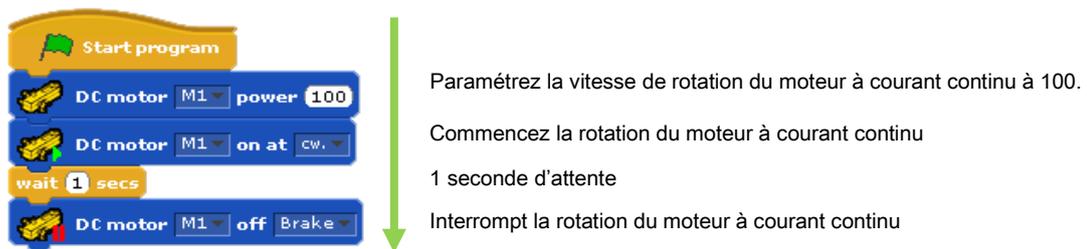
- ① À partir de la palette «Mouvement», sélectionnez ,
 et . Connectez-les ensemble.
- ② Faites glisser  de la palette «Contrôle» et insérez-le entre
 et .



③ Attachez les blocs à .

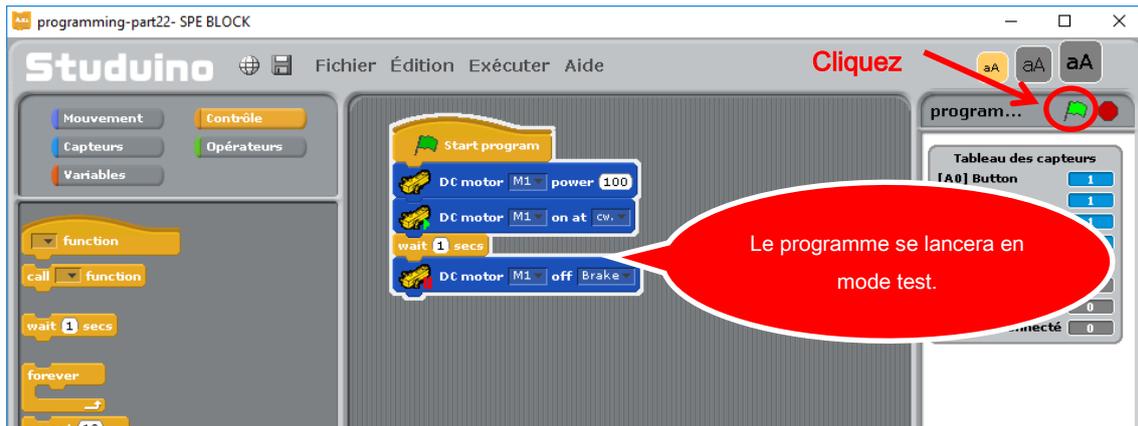


Le programme est maintenant complet. Le moteur à courant continu connecté à M1 tournera.



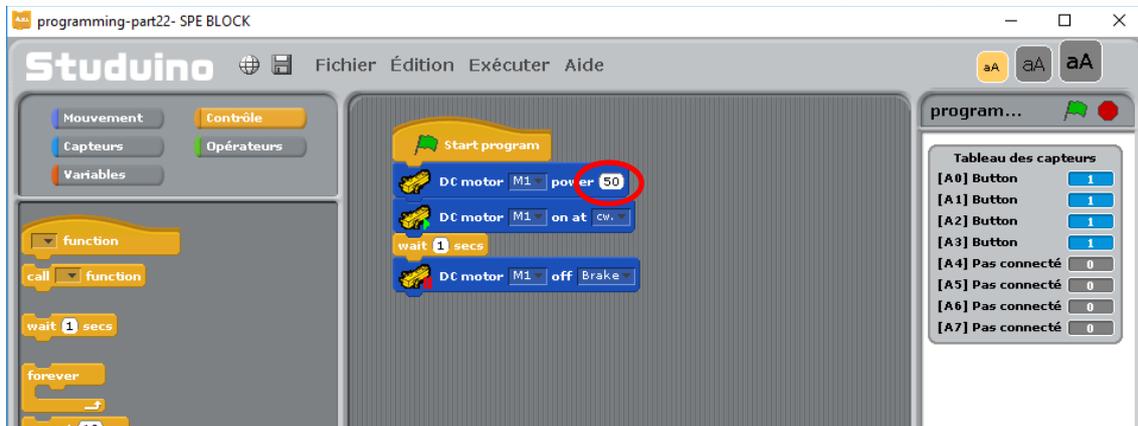
④ Vérifiez comment le moteur à courant continu tourne. Connectez votre circuit Arduino au PC en utilisant le câble USB. Choisissez «Activer le test» depuis le menu «Exécuter».

⑤ Cliquez sur  après être entré en mode test.



Le moteur à courant continu connecté à M1 tournera pendant une durée d'une seconde puis s'arrêtera.

⑥ Définissez la valeur du bloc  sur 50, puis cliquez sur .



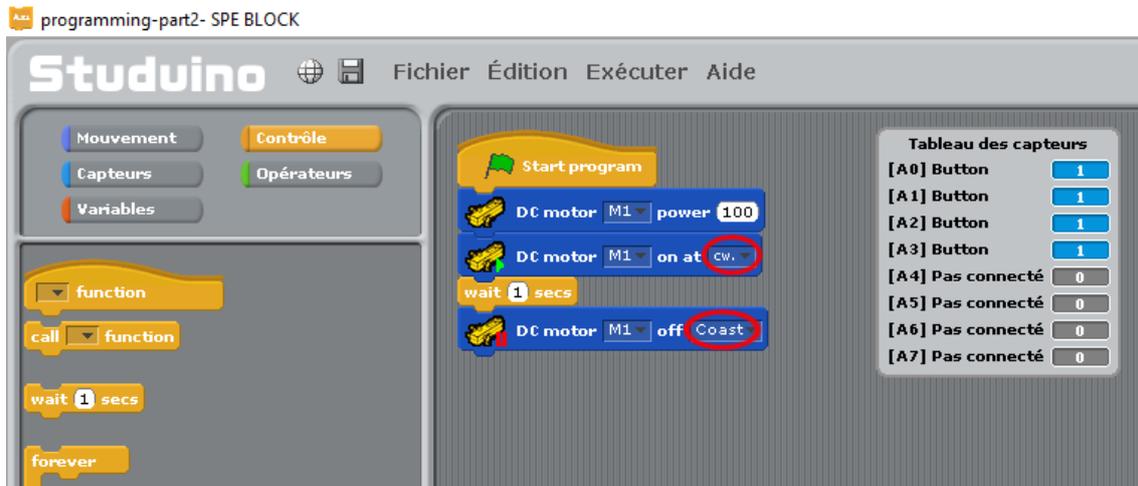
Le moteur à courant continu tournera plus lentement.

⑦ Ramenez de nouveau la valeur de  à 100 et paramétrez  sur «counterclockwise» (ccw, « dans le sens contraire des aiguilles d'une montre »). Cliquez sur .



Le moteur à courant continu tournera en sens inverse.

- ⑧ Changez  de nouveau sur «clockwise». Définissez  sur «coast», puis cliquez sur .



Le moteur à courant continu tournera dans le sens des aiguilles d'une montre (vers l'avant) et exécutera un arrêt en rotation libre.

- ★ La valeur de 100 correspond à la vitesse de rotation maximum du moteur à courant continu. Saisir une valeur supérieure à 100 ne fera pas tourner le moteur plus rapidement.
- ★ Le moteur à courant continu ne tournera pas si la vitesse est configurée en dessous d'un certain seuil.

Le paramétrage de la vitesse représente l'intensité du courant électrique envoyé au moteur à courant continu. Afin que votre moteur à courant continu fonctionne, il faut un montant minimum de courant électrique.

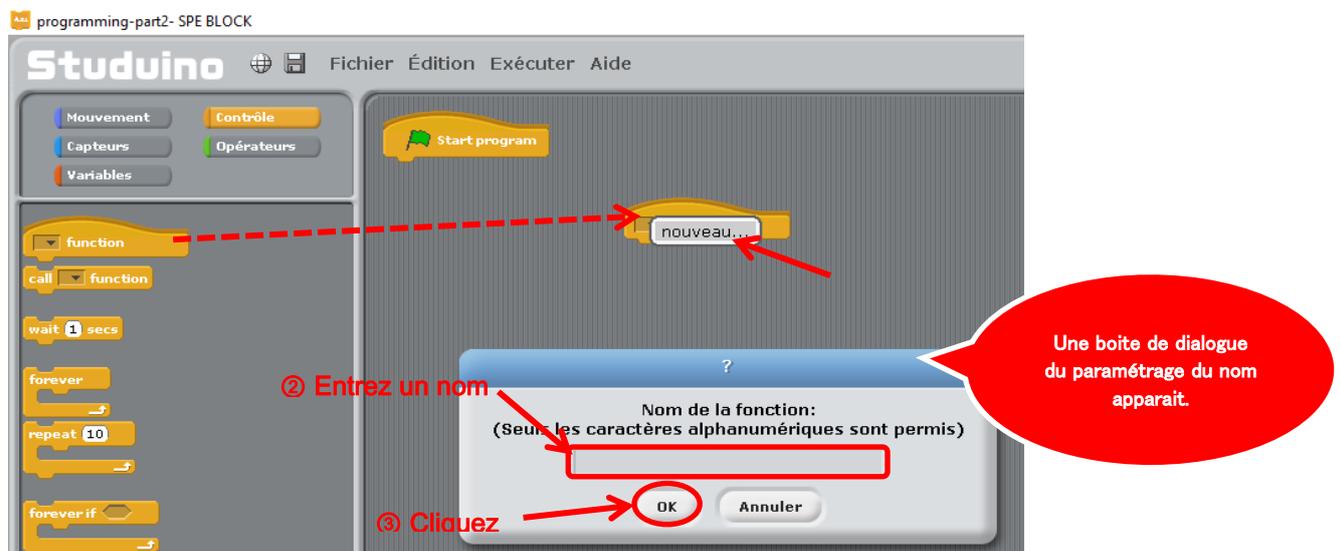
5.2. Contrôler une voiture

Maintenant, nous allons créer un programme qui va utiliser deux moteurs à courant continu afin de faire avancer, reculer et tourner une voiture.

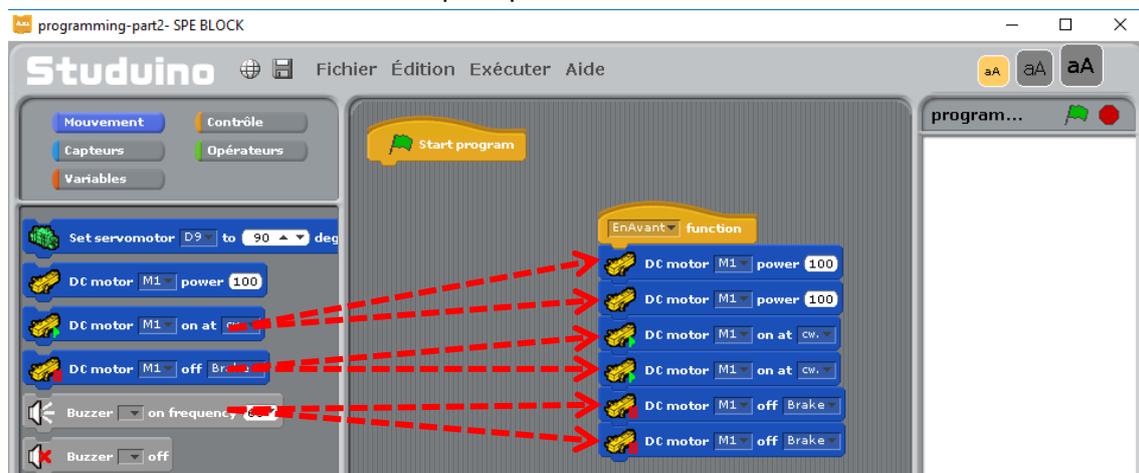
5.2.1. Programmer

Tout d'abord, nous allons créer un programme qui va permettre à la voiture d'avancer.

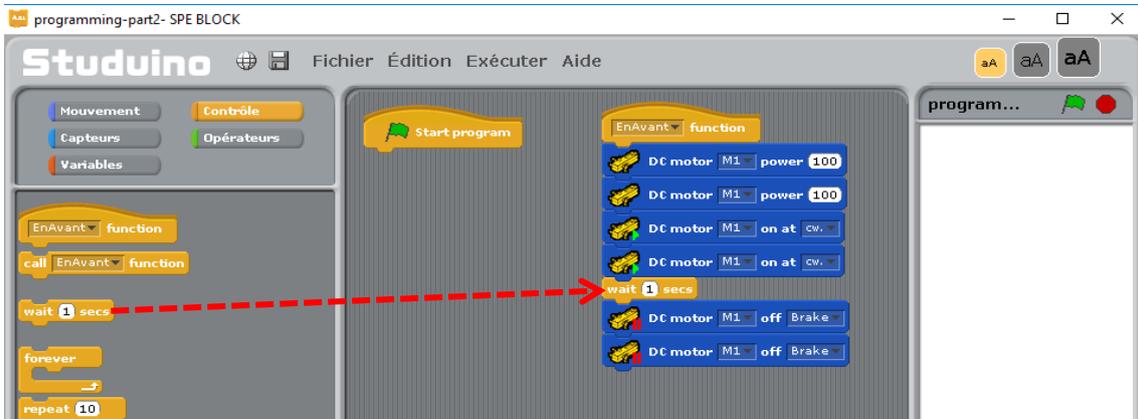
- ① À partir de la palette « Contrôle », choisissez . Sélectionnez «nouveau...» en cliquant sur ▼. Nommez la fonction «EnAvant» et cliquez sur OK.



- ② À partir de la palette «Mouvement», choisissez 2 blocs ,  et , puis connectez-les au bloc  dans le champ script les uns à la suite des autres.



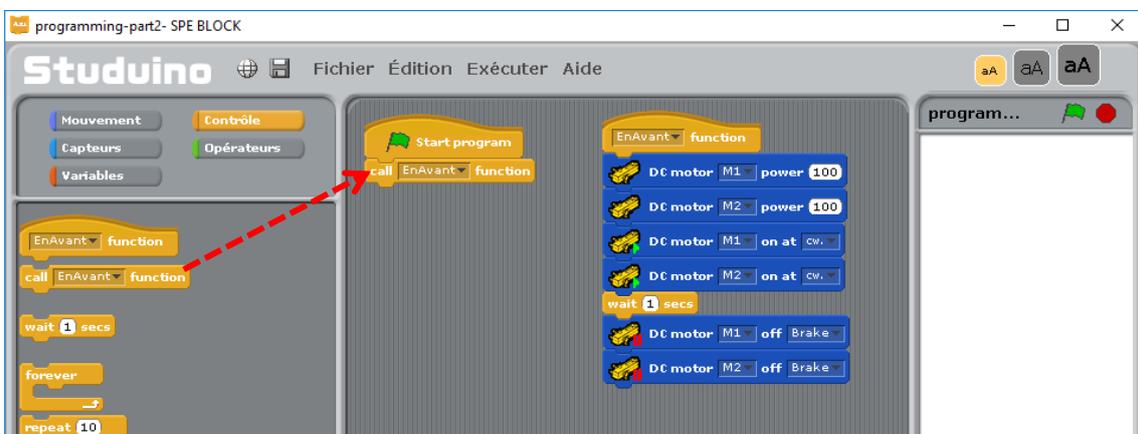
Faites glisser **wait 1 secs** depuis la palette «Contrôle» et insérez-le entre



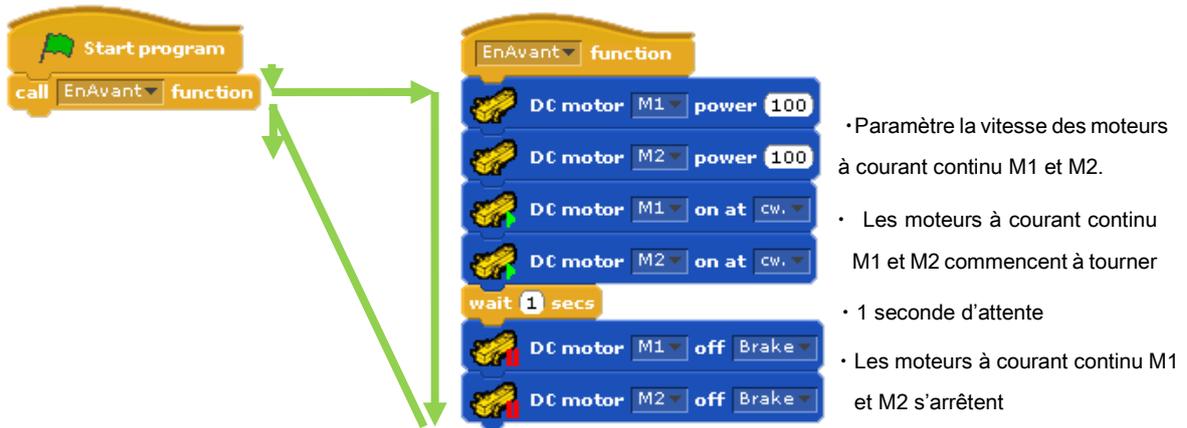
③ Dans l'un des blocs **DC motor M1 power 100**, **DC motor M1 on at cw.** et **DC motor M1 off Brake.**, passez de M1 à M2.



④ Attachez **call EnAvant fonction** à **Start program**.



Le programme est maintenant complet.

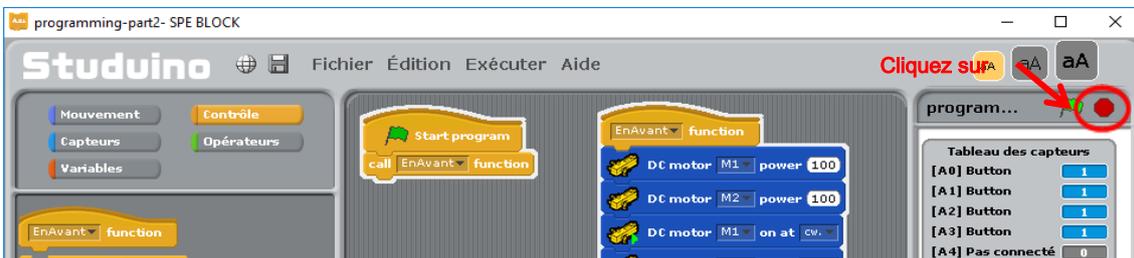


- Maintenant observez comment la voiture se déplace. Connectez le circuit Studuino au PC en utilisant le câble USB. Choisissez «Activer le test» depuis le menu «Exécuter».
- Cliquez sur  après être entré dans le mode test.



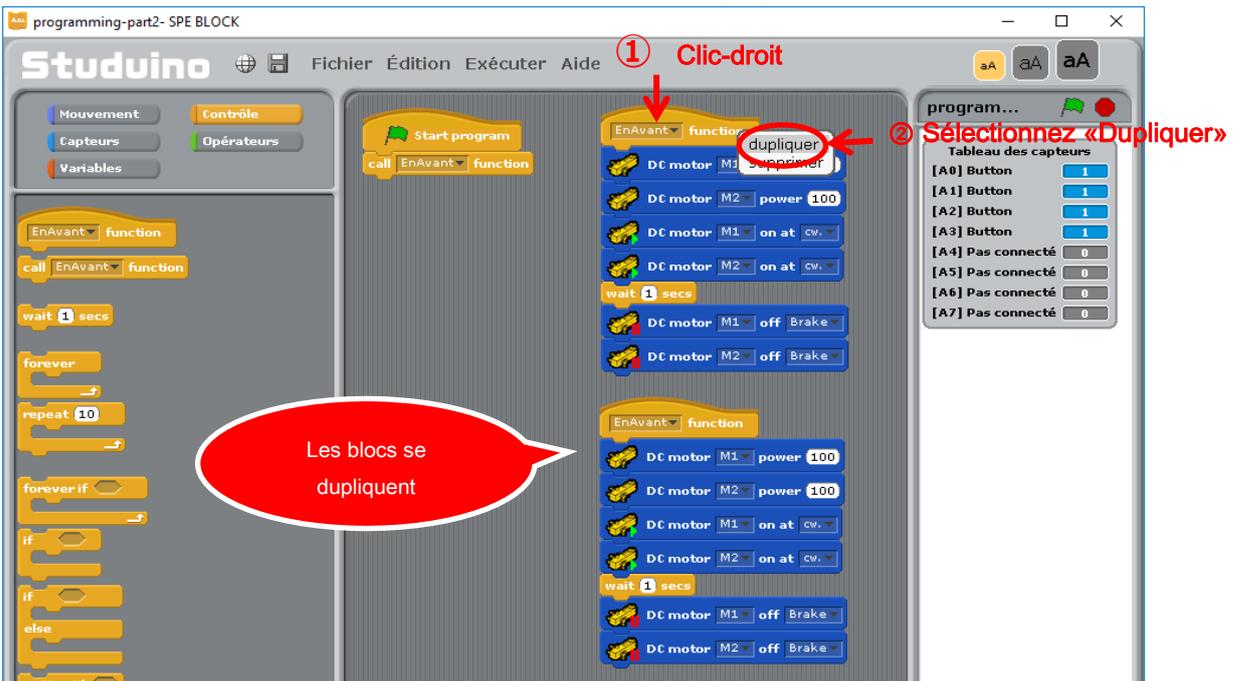
La voiture avancera pendant une seconde puis s'arrêtera.

- Cliquez sur  pour arrêter le mode test.



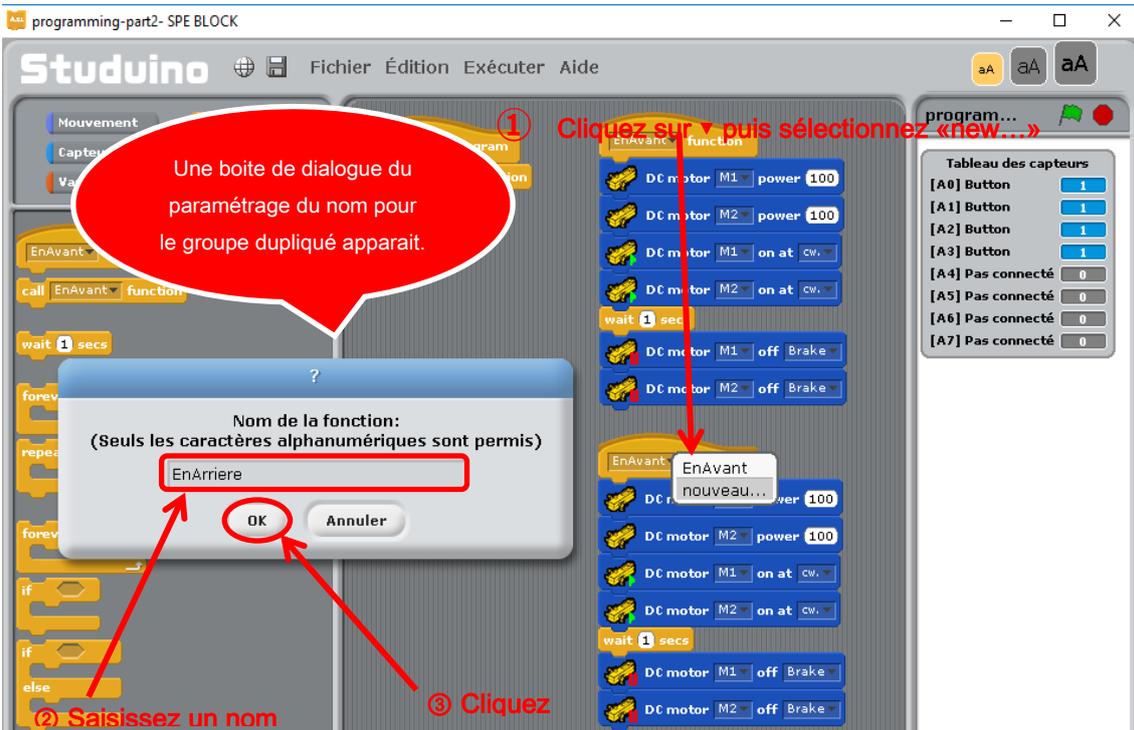
Ensuite, nous allons créer un programme qui va permettre à la voiture de reculer.

⑧ Clic droit sur **EnAvant** fonction et sélectionnez «Dupliquer».

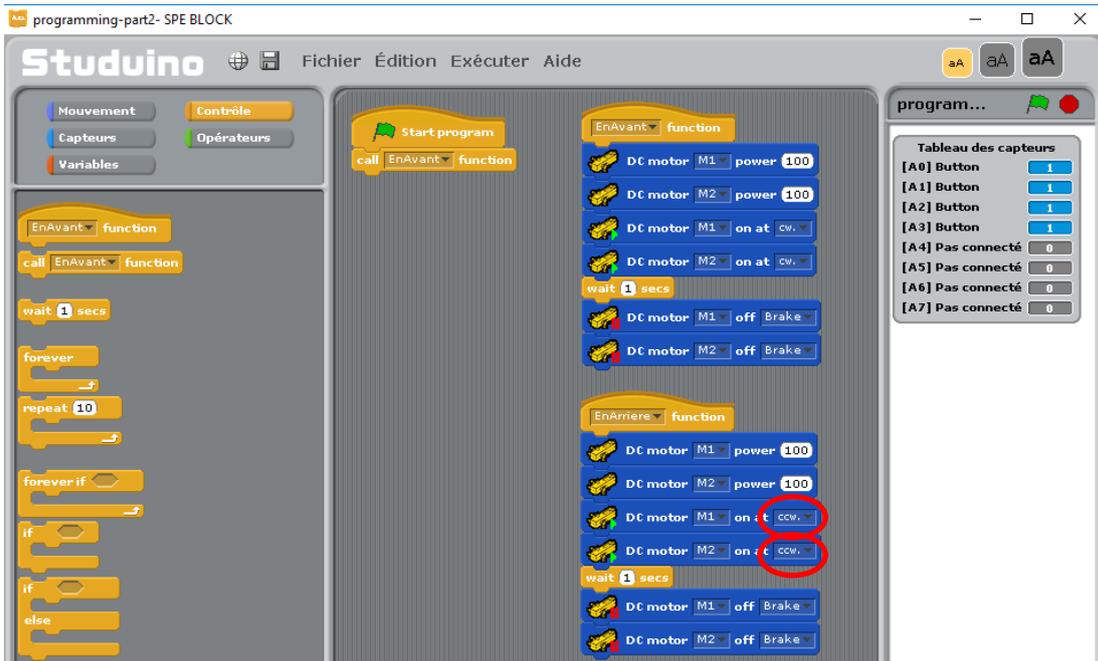


⑨

Cliquez sur ▼ au niveau du **EnAvant** fonction dupliqué puis sélectionnez «nouveau...». Nommez la fonction «EnArriere» et cliquez sur OK.



- ⑩ Dans le groupe nommé «EnArriere», paramétrez le sens de la rotation sur «counterclockwise» sur  et .

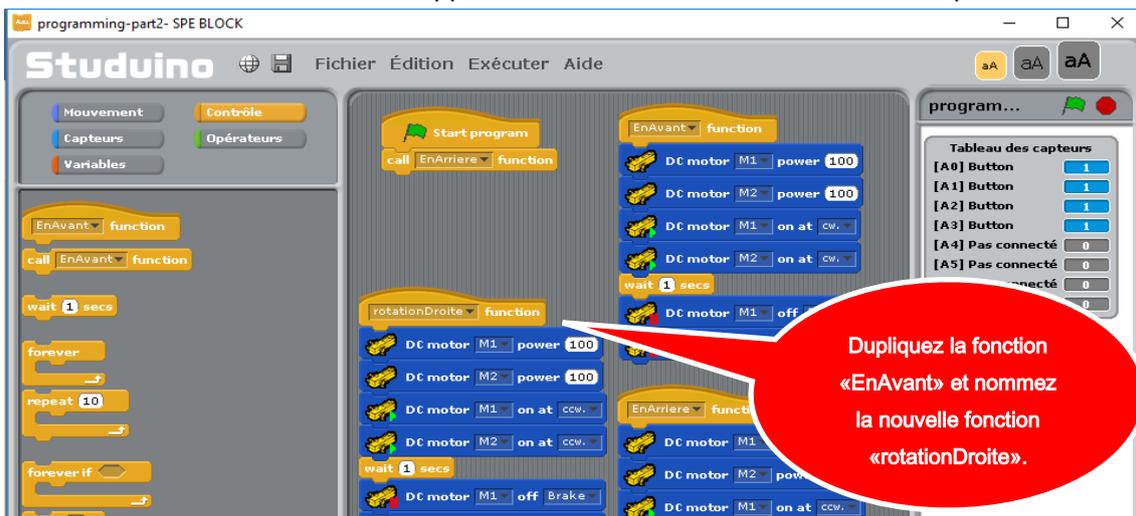


- ⑪ Cliquez sur ▼ dans  puis sélectionnez «EnArriere».

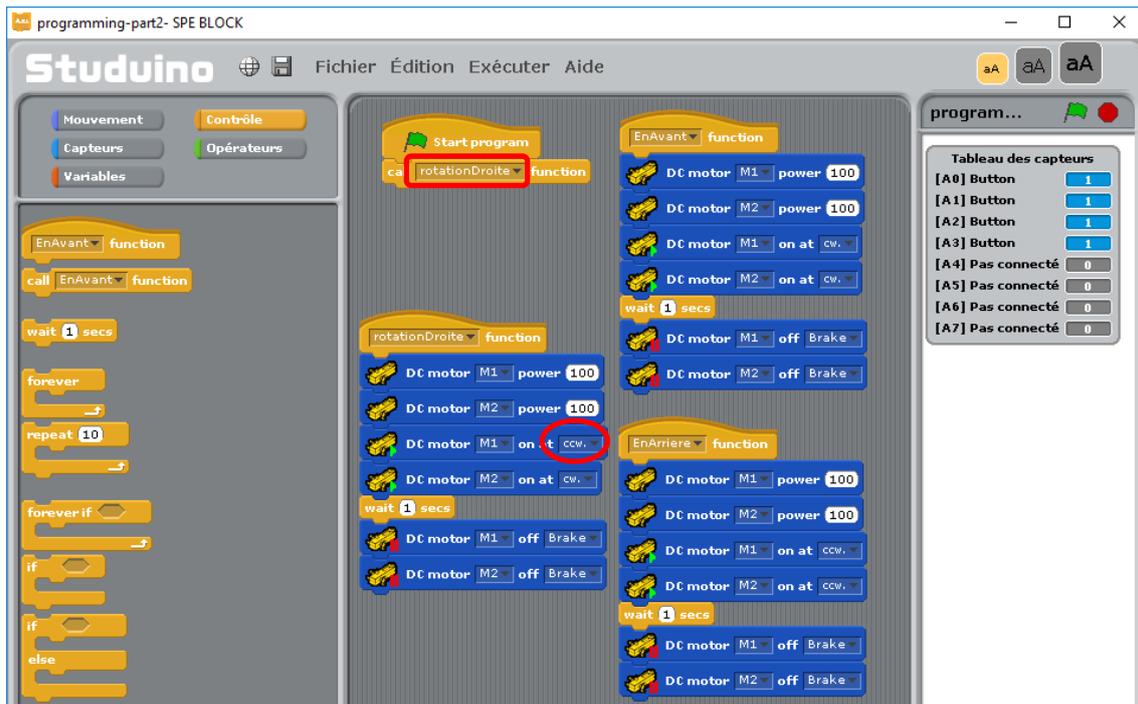
Le programme est maintenant complet. Lorsque le programme exécute les étapes 6 et 7, la voiture recule pendant une seconde puis s'arrête.

Ensuite, nous allons créer un programme qui va permettre à la voiture de tourner dans le sens des aiguilles d'une montre.

- ⑫ Créez une nouvelle fonction appelée «rotationDroite» en suivant les étapes 8 et 9.



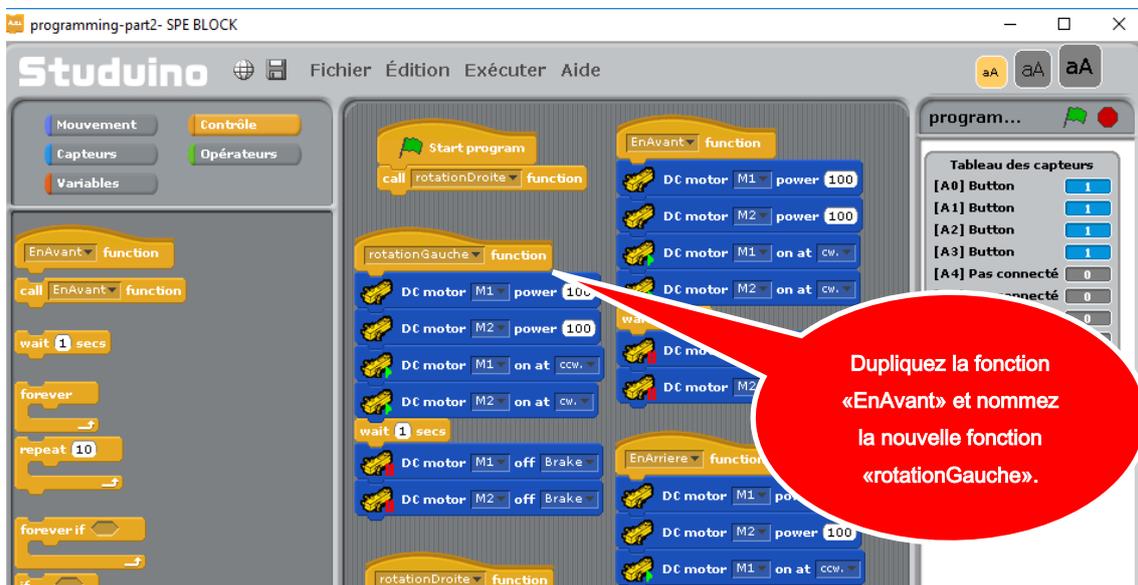
- ⑬ Paramétrez le bloc  de la première ligne sur «counterclockwise». Sélectionnez «rotationDroite» en cliquant sur ▼ au niveau du bloc .



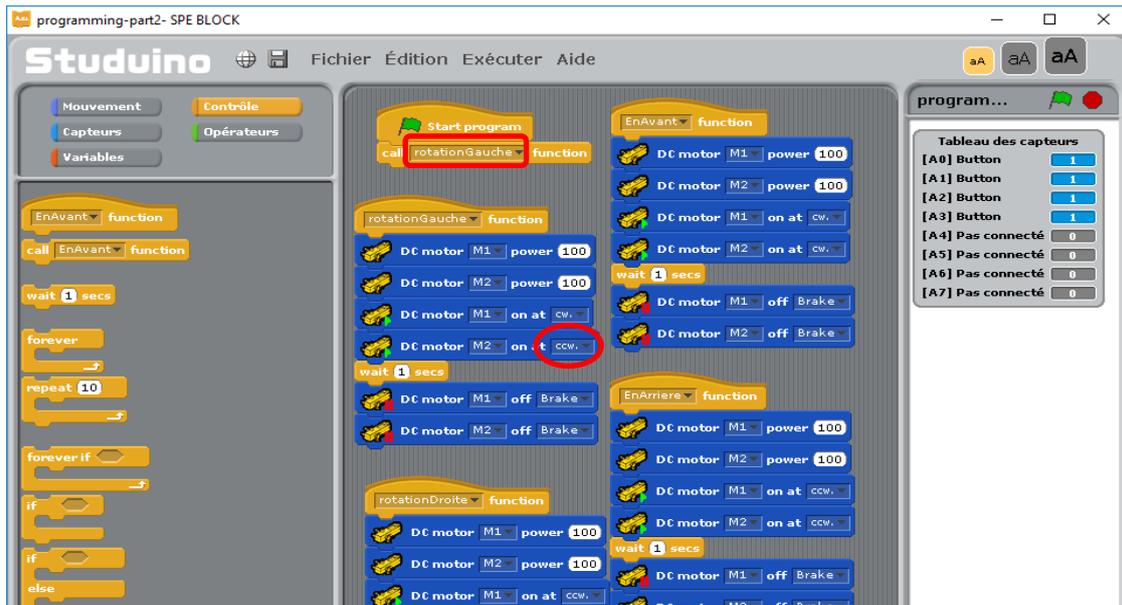
Le programme est maintenant complet. Suivez les étapes 6 et 7. Votre voiture tournera dans le sens des aiguilles d'une montre pendant une seconde puis s'arrêtera.

Ensuite, nous allons créer un programme qui va permettre à la voiture de tourner dans le sens contraire des aiguilles d'une montre.

⑭ Créez une nouvelle fonction appelée «rotationGauche» en suivant les étapes 8 et 9.



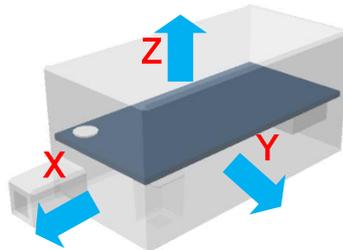
⑮ Paramétrez le bloc  de la seconde ligne sur «counterclockwise». Sélectionnez «rotationGauche» en cliquant sur ▼ au niveau du  bloc.



Le programme est maintenant complet. Suivez les étapes 6 et 7. Vérifiez que la voiture tourne dans le sens contraire des aiguilles d'une montre pendant une seconde puis s'arrête.

5.3. Contrôler une voiture en utilisant un accéléromètre

Maintenant, nous allons créer un programme qui utilise un accéléromètre pour contrôler la voiture. Basez votre programme sur la section précédente. Un accéléromètre fonctionne en calculant les accélérations sur les axes X, Y et Z. Il calcule les variations de vitesse sur une période de temps donné.



Comme l'accéléromètre bouge sur différents axes, chaque valeur change comme indiqué ci-dessous.



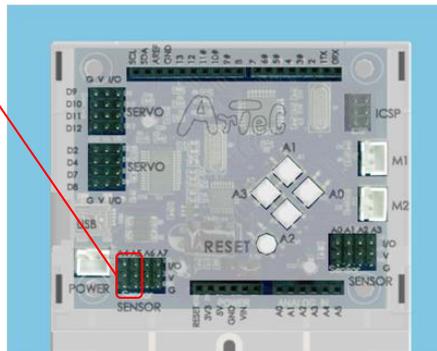
Un accéléromètre fixe dans une direction dans laquelle il indique autre chose que 0 car il détecte une accélération gravitationnelle. Les objets sont attirés vers le sol en raison

de la gravité de la terre. Cette force est utilisée pour mesurer les angles (bascule) par rapport au sol.

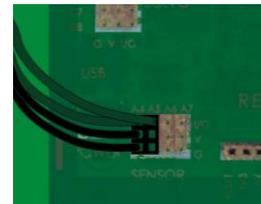
★ L'accélération gravitationnelle est la force exercée par la gravité sur un objet.

5.3.1. Connecter un accéléromètre au circuit Studuino

Connectez un accéléromètre au connecteur A4 (capteur/ LED/ avertisseur sonore) sur le circuit Studuino.

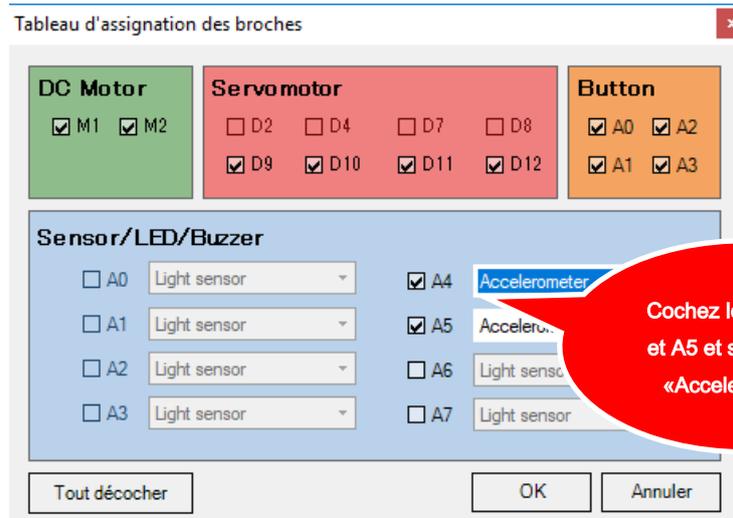


Les quatre fils de l'accéléromètre se connectent sur A4 et A5.



5.3.2. Paramétrage des ports

Avant de commencer à programmer, définissez le port d'information de l'environnement de programmation par blocs pour qu'il s'accorde à votre circuit. Sélectionnez «Édition» dans le menu, puis dans le menu déroulant sélectionnez «Paramètres des ports...» pour accéder à la boîte de dialogue de réglages. Dans la boîte de dialogue de réglages du port, cochez A4 et A5 et choisissez accéléromètre.



Tester l'accéléromètre

Trouvez la valeur de l'accéléromètre. Connectez votre circuit Studuino au PC en utilisant le câble USB. Choisissez **«Activer le test»** depuis le menu **«Exécuter»**. Dans le mode test, le circuit du capteur apparait et indique "[A4/A5] Accelerometer".

Les valeurs de l'accéléromètre mesurent l'inclinaison sur une tranche allant de 0 à 100 et où 50 correspond à un niveau parfait sur le sol. Lorsque l'inclinaison devient positive, la valeur tend à se rapprocher de 100 tandis que pour une inclinaison négative elle tend vers 0. Inclinez l'accéléromètre et notez les variations de valeur.

Tableau des capteurs	
[A0] Button	1
[A1] Button	1
[A2] Button	1
[A3] Button	1
[A4/A5] Accelerometer (X)	52
[A4/A5] Accelerometer (Y)	75
[A4/A5] Accelerometer (Z)	51
[A6] Pas connecté	0
[A7] Pas connecté	0



Les valeurs changent en fonction de l'inclinaison.

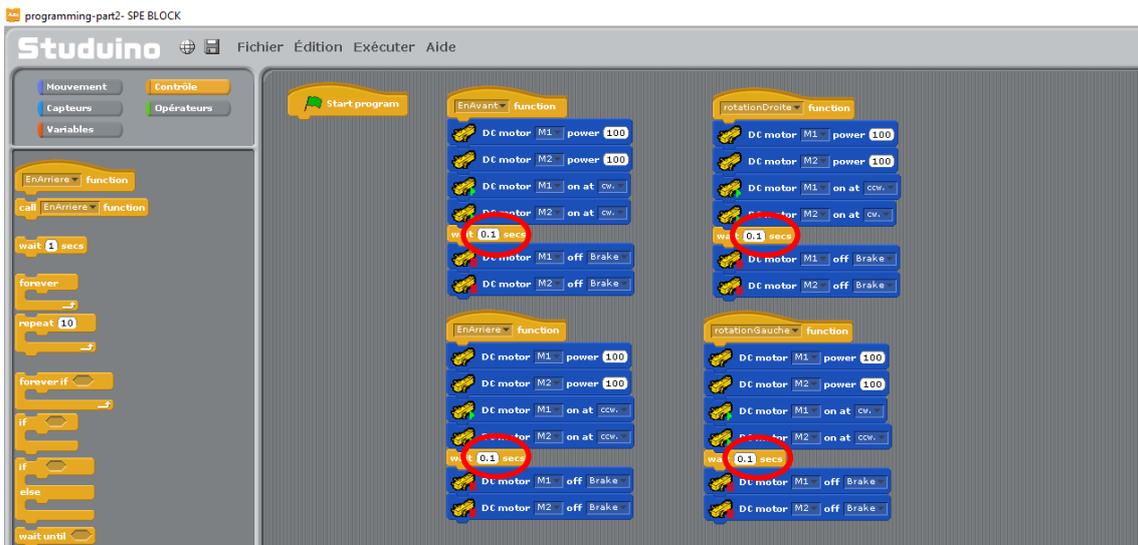
Maintenant, quittez le mode test en sélectionnant **«Désactiver le test»** depuis le menu **«Exécuter»**.

5.3.3. Programmer

Pour ce chapitre, nous utiliserons le programme abordé dans la section 5.2.1 qui permettait à la voiture d'avancer, de reculer et de pivoter.



① Définissez la valeur du bloc **wait 1 secs** sur 0.1.



② Paramétrez  et  sur «Coast».

programming-part2- SPE BLOCK

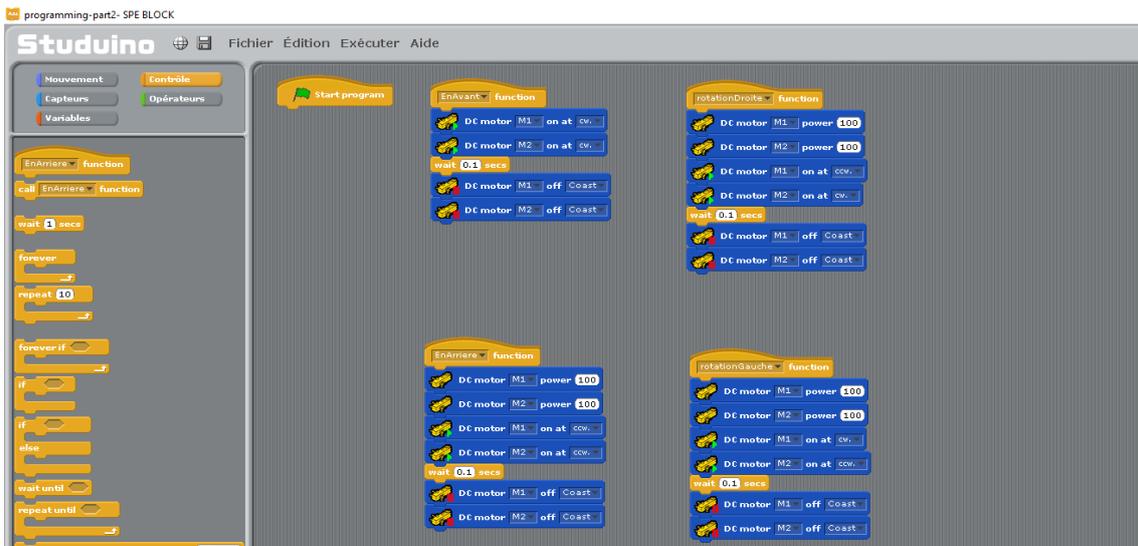


③ Dans le groupe de fonction «EnAvant», déconnectez  et  , puis faites regliser  vers la palette de bloc.

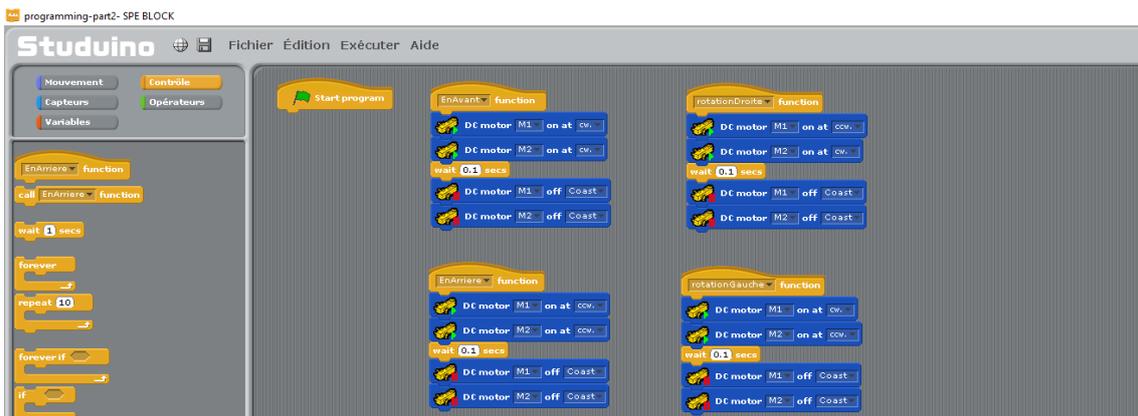
programming-part2- SPE BLOCK



④ Attachez à  un bloc .



⑤ Suivez les étapes ③ et ④ pour supprimer le bloc   des autres groupes de fonction.



- ⑥ À partir de la palette «Mouvement», faites glisser deux  et attachez-les à . Paramétrez le second  sur M2 comme indiqué.



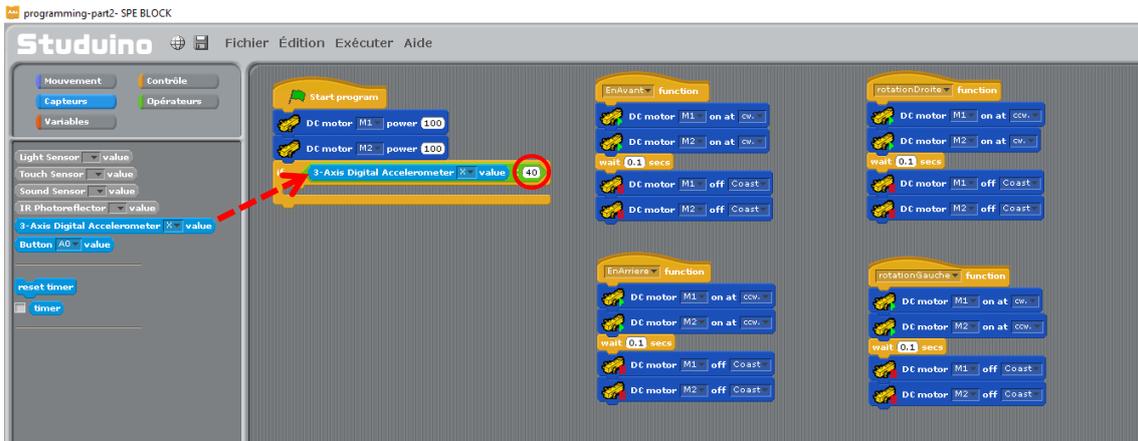
- ⑦ À partir de la palette «Contrôle», faites glisser et attachez  comme indiqué.



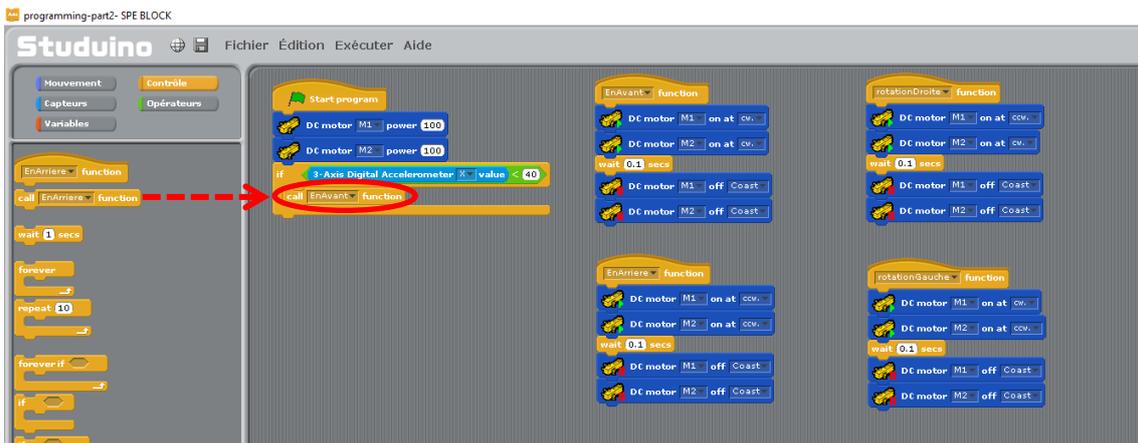
- ⑧ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



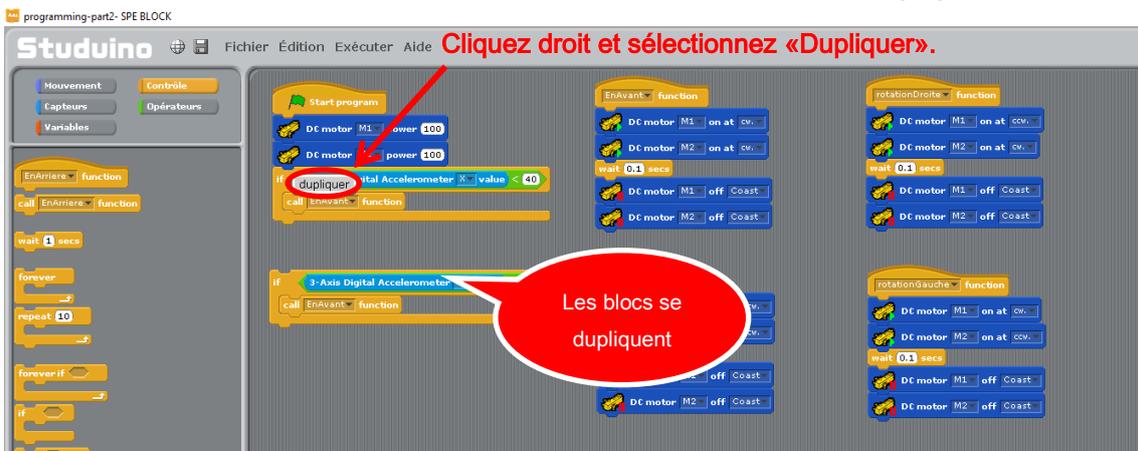
- ⑨ À partir de la palette «Capteurs », faites glisser et attachez **3-Axis Digital Accelerometer** du côté gauche de . Ensuite, paramétrez le côté droit sur 40.



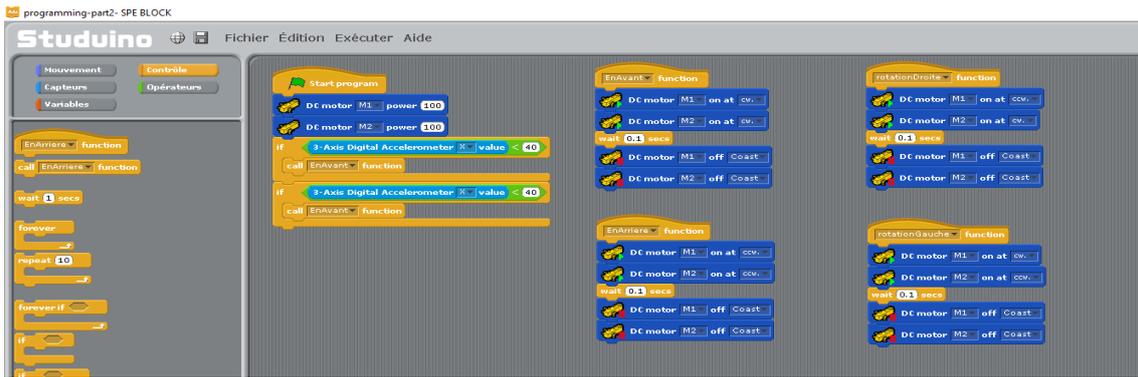
- ⑩ À partir de la palette «Contrôle», faites glisser et attachez **call EnAvant** au niveau du début de , puis sélectionnez la fonction « EnAvant».



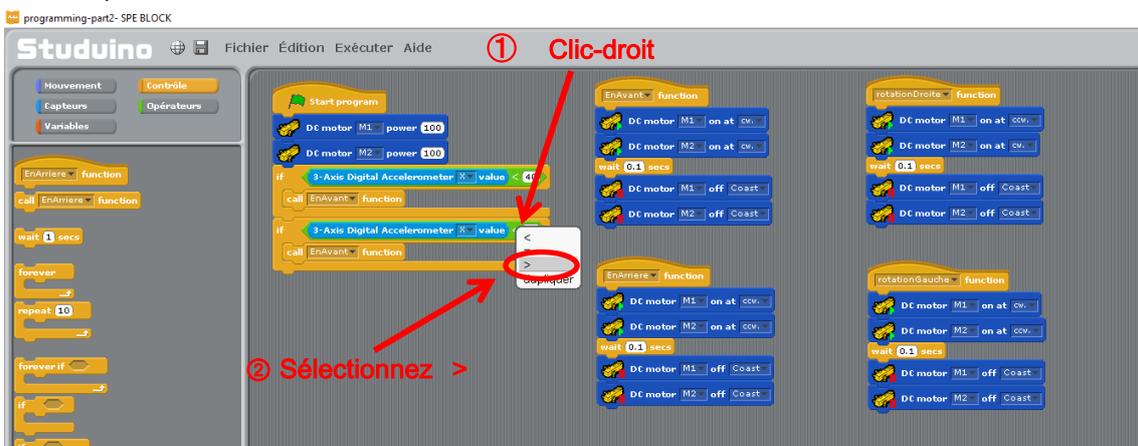
- ⑪ Clic droit sur  et sélectionnez «Dupliquer».



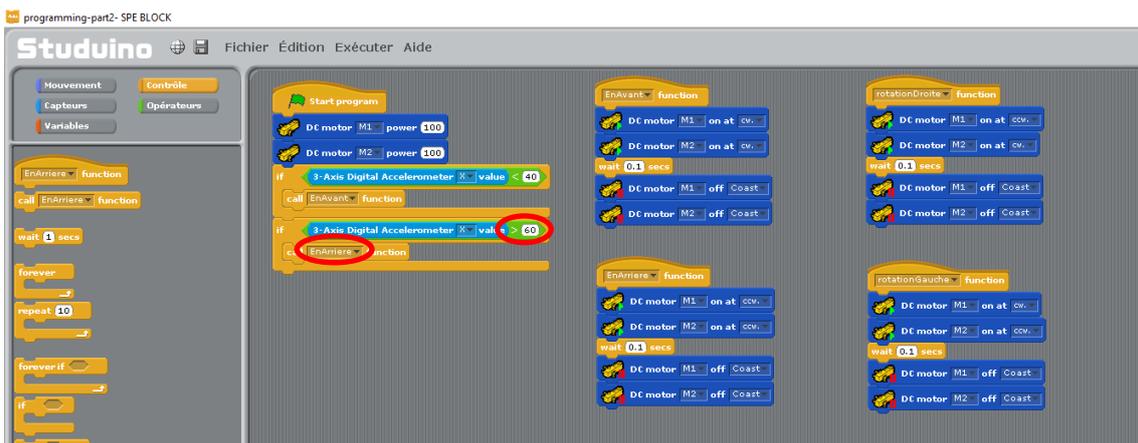
12 Connectez les blocs dupliqués aux blocs d'origine.



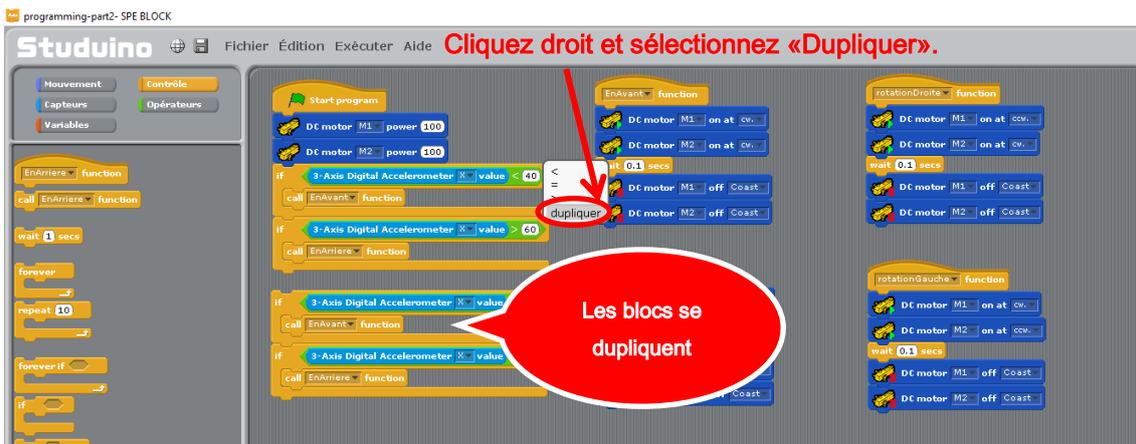
13 Cliquez droit sur  et sélectionnez >.



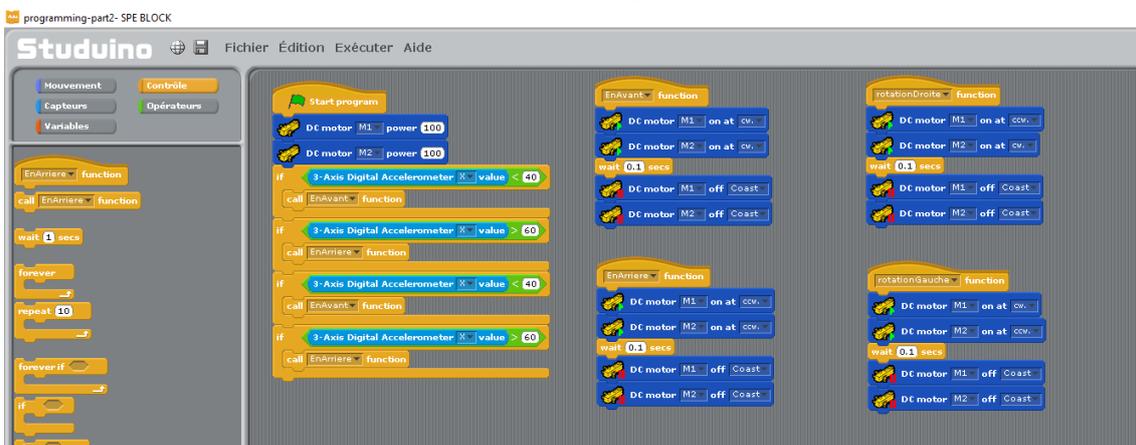
14 Paramétrez le côté droit de  sur 60 et paramétrez  sur la fonction «EnArriere».



- 15 Cliquez droit sur  et sélectionnez «Dupliquer».



- 16 Connectez les blocs dupliqués aux blocs d'origine.



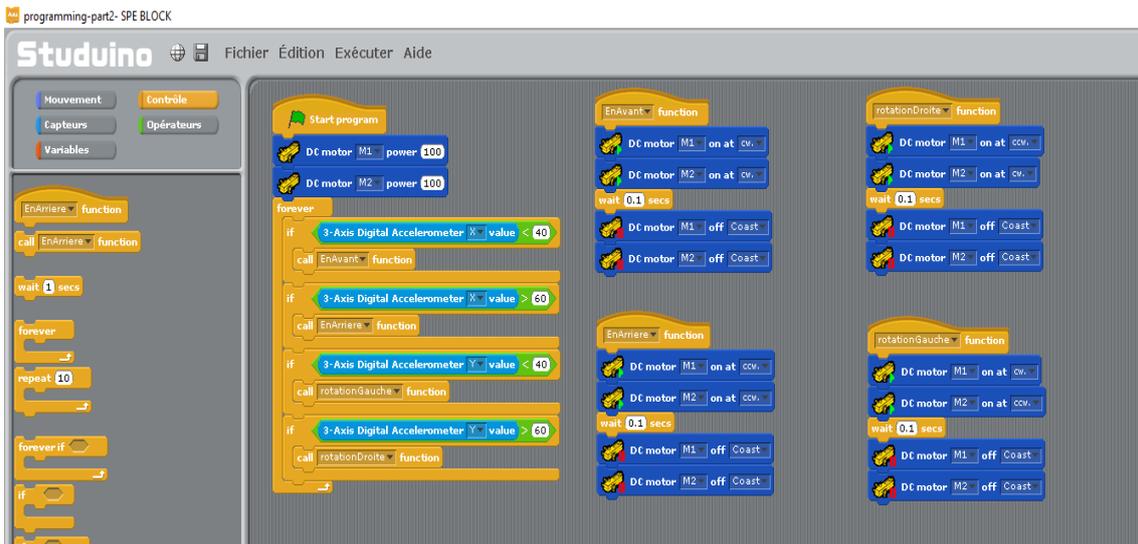
- 17 Paramétrez les blocs dupliqués  sur Y. Comme indiqué, paramétrez un bloc  sur «rotationGauche» et l'autre sur «rotationDroite».



⑧ Faites glisser  et insérez-le au groupe de blocs en dessous de .



Le programme est maintenant complet. Il utilisera l'accéléromètre pour contrôler la voiture.



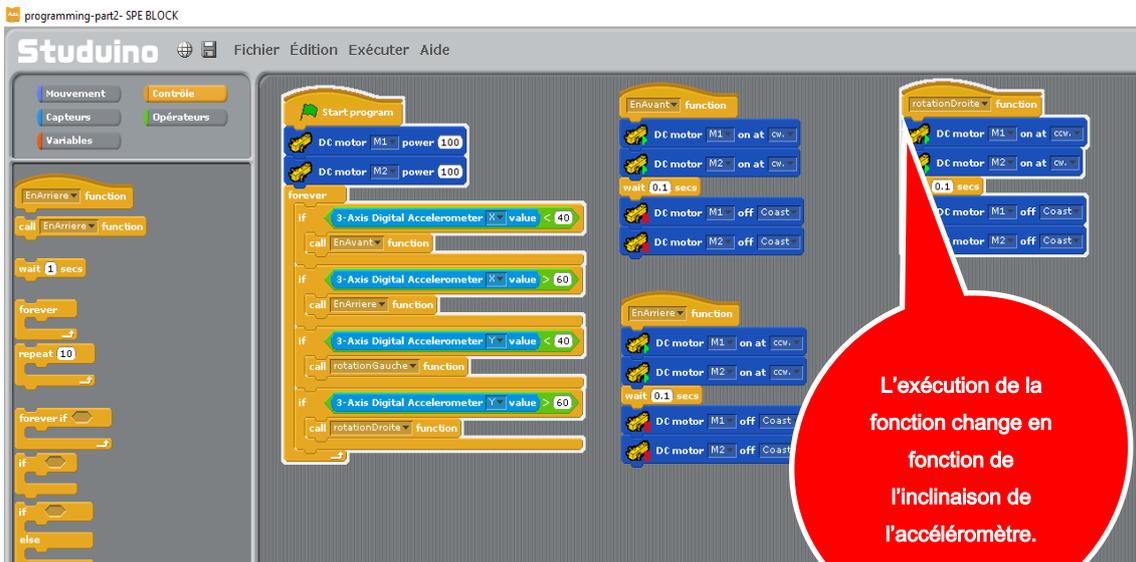


Paramétrez la vitesse des moteurs à courant continu M1 et M2 sur 100.

- Lorsque l'accéléromètre est incliné vers l'avant, il exécute la fonction «EnAvant»
- Lorsque l'accéléromètre est incliné vers l'arrière, il exécute la fonction «EnArriere»
- Lorsque l'accéléromètre est incliné vers la gauche, il exécute la fonction «rotationGauche »
- Lorsque l'accéléromètre est incliné vers la droite, il exécute la fonction «rotationDroite»

⑩ Observez comment la voiture réagit lorsque l'accéléromètre est incliné. Connectez le circuit Studuino au PC en utilisant le câble USB. Activez votre boîtier de piles, puis choisissez «**Activer le test**» depuis le menu «**Exécuter**».

⑪ Cliquez sur  après être entré dans le mode test.



La voiture se déplace en fonction de l'inclinaison de l'accéléromètre.

⑫ Après avoir fait les vérifications de la voiture dans le mode test, choisissez «**Transférer**» depuis le menu «**Exécuter**».

6. Faire bouger le servomoteur

Cette section explique comment créer un programme pour servomoteur basique, comment fonctionne le servomoteur et comment utiliser les blocs «Liste» et «Opérateurs».

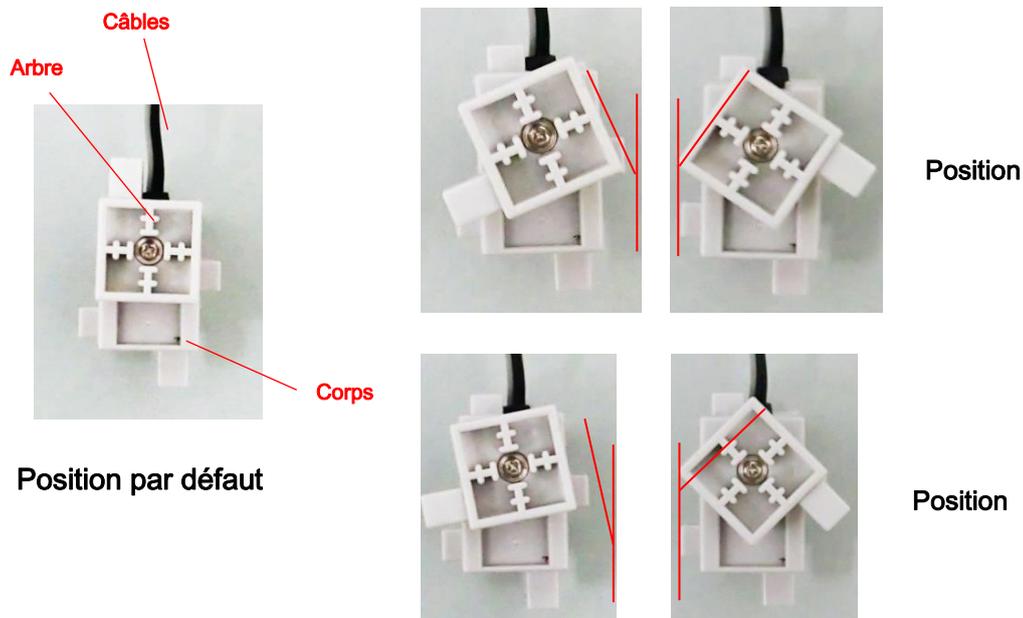
6.1. Calibrer les angles du servomoteur

Chaque servomoteur est différent. Même si vous paramétrez chaque angle pour qu'ils soient identiques aux autres, il peut y avoir de légères différences (appelées glissement) au niveau des angles de votre moteur. Cet écart peut être corrigé en calibrant votre servomoteur.

6.1.1. Ajuster l'angle de l'arbre de transmission du servomoteur

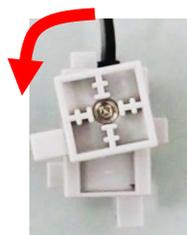
Avant de fixer le servomoteur, vérifiez que l'arbre de transmission du servomoteur est correctement attaché en utilisant la méthode indiquée ci-dessous.

En faisant pivoter l'arbre de transmission trop loin sur la gauche ou sur la droite, par rapport à sa position par défaut, vous pourriez déplacer de sa place la portion du bloc de l'arbre de transmission.



Afin de ne pas endommager les engrenages intérieurs, veillez à utiliser un niveau de force raisonnable lorsque vous manipulez l'arbre de transmission. Pour éviter d'endommager les engrenages, seul le bloc bouge lorsqu'il est tourné.

Si vous tournez l'arbre de transmission lorsqu'il est en position anormale, vous entendrez un «clic» indiquant qu'il est retourné à sa position par défaut.

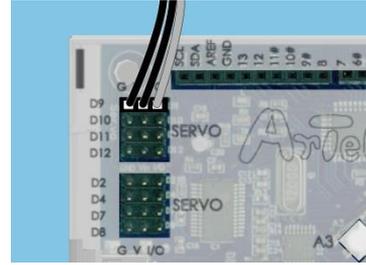
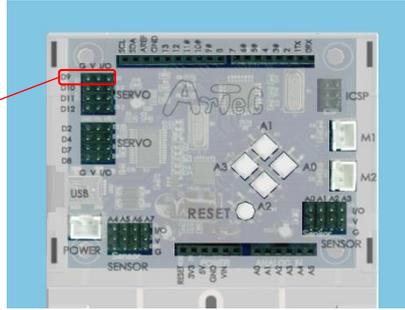
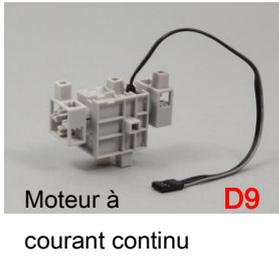


★Évitez d'appliquer une pression trop importante lorsque vous le tournez. Cela peut sérieusement endommager le servomoteur.

★Des ajustements mineurs de l'angle pourront être effectués plus tard grâce à l'utilisation d'un logiciel.

6.1.2. Connecter un servomoteur au circuit Studuino

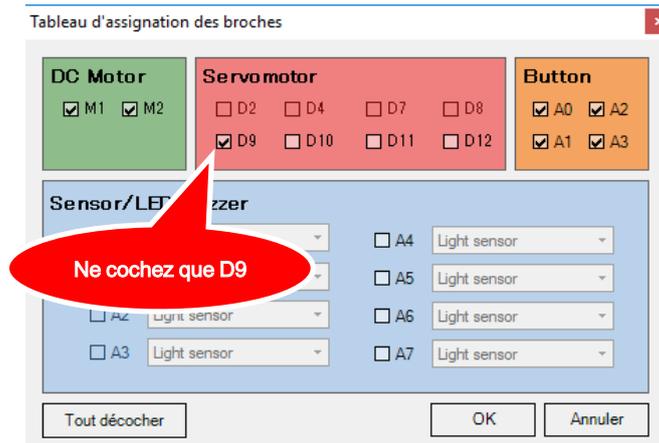
Connectez le servomoteur à D9 sur l'unité Studuino



Assurez-vous que les câbles de raccordement de capteur soient insérés correctement. Le câble gris doit être tourné vers la carte.

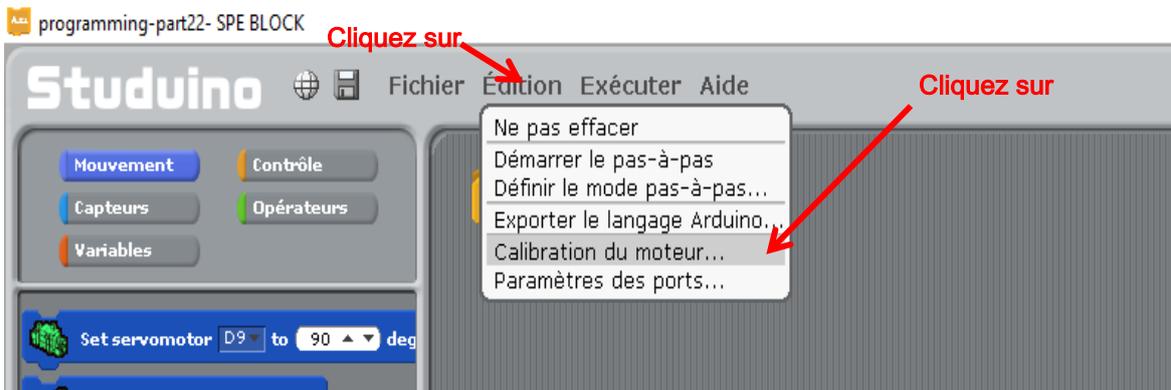
6.1.3. Paramétrage des ports

Avant de commencer à programmer, définissez les ports dans l'environnement de programmation par blocs pour qu'il s'accorde à votre circuit. Sélectionnez «**Édition**» dans le menu, puis dans le menu déroulant sélectionnez «**Paramètres des ports**» pour accéder à la boîte de dialogue de réglages. Assurez-vous que seule la case D9 soit cochée dans la zone «servomotor» comme indiqué ci-dessous.



① Choisissez «**Édition**» dans le menu, puis sélectionnez «**Calibration du moteur...**».

Assurez-vous que le câble USB soit bien connecté au circuit.



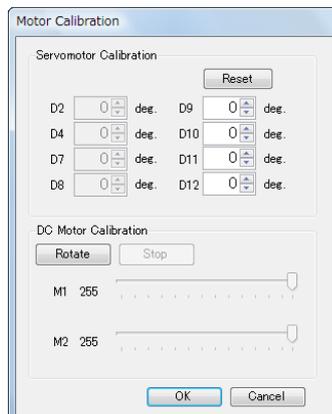
Lorsque «**Calibration du moteur...**» est sélectionné, une fenêtre s'ouvre. Tous les angles des servomoteurs sont paramétrés sur 90 et le logiciel ira automatiquement dans le

mode test comme indiqué ci-dessous.

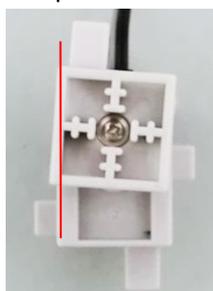
★ Assurez-vous que la source d'énergie soit bien émise par le boîtier de piles et que ce dernier soit bien connecté au circuit.



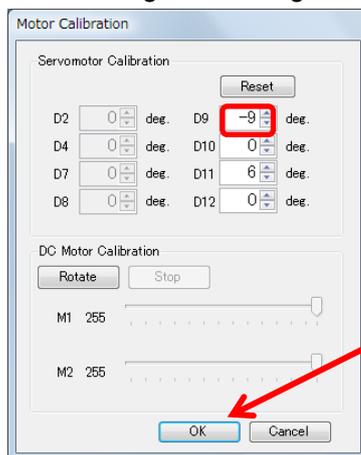
L'arbre de transmission est à 90 degrés et le corps mis à niveau.



- ② Lorsque l'arbre de transmission est légèrement décalé de son angle de 90 degrés, la valeur de mesure peut être affinée dans la fenêtre d'étalonnage du servomoteur. Assurez-vous de n'entrer que les valeurs qui engendrent un angle de 90 degrés.



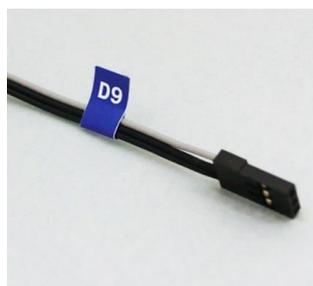
L'arbre de transmission et le corps ont besoin d'être ajustés.



Cliquez sur

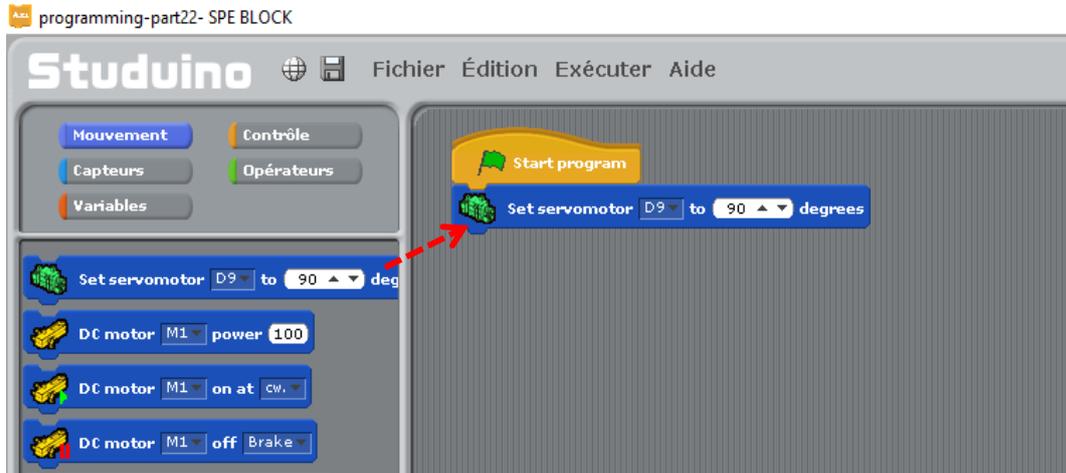
★ Les valeurs doivent être entrées de nouveau si vous changez les connecteurs du servomoteur.

Lorsque l'étalonnage de l'angle du servomoteur est complété, collez une étiquette indiquant le même nombre que le connecteur du servomoteur afin de l'identifier plus facilement.

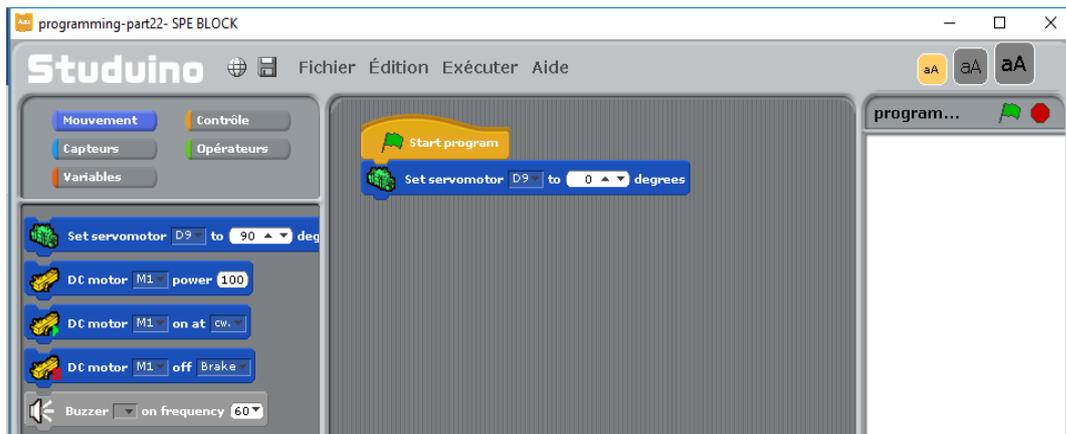


6.1.4. Programmer

- ① À partir de la palette «Mouvement», sélectionnez  et connectez-le à .



- ② Changez 90 en 0 dans . Connectez le circuit Studuino au PC en utilisant le câble USB. Activez votre boîtier de piles, puis choisissez «Activer le test» depuis le menu «Exécuter».
- ③ Cliquez sur  après être entré dans le mode test.



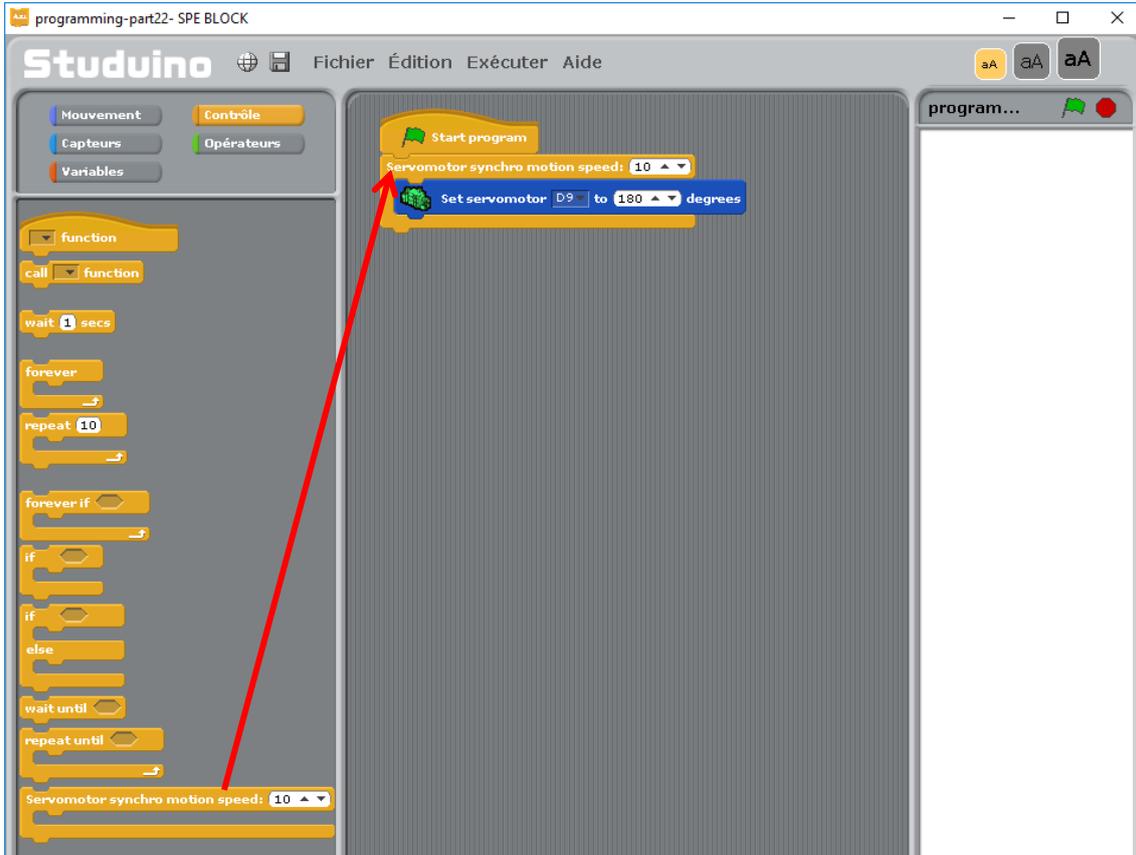
Assurez-vous que le servomoteur bouge sur 0 degré.

- ④ Changez 0 en 180 sur , puis cliquez sur . Assurez-vous que le servomoteur bouge sur 180 degrés.

⑤ Insérez



dans



⑥ Entrez la valeur 180 dans le bloc



et cliquez sur 

afin de vérifier si le moteur bouge pour atteindre un angle de 180 degrés.



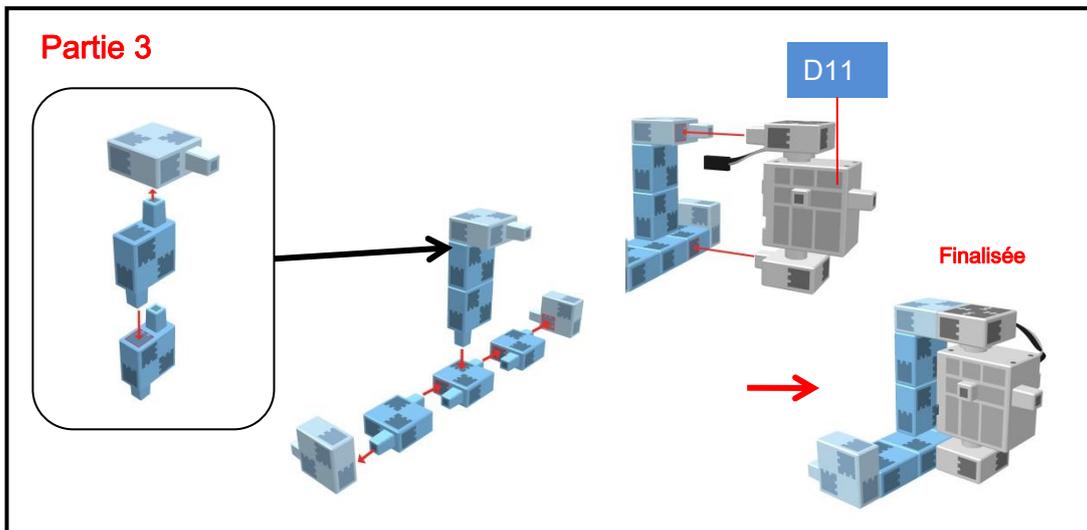
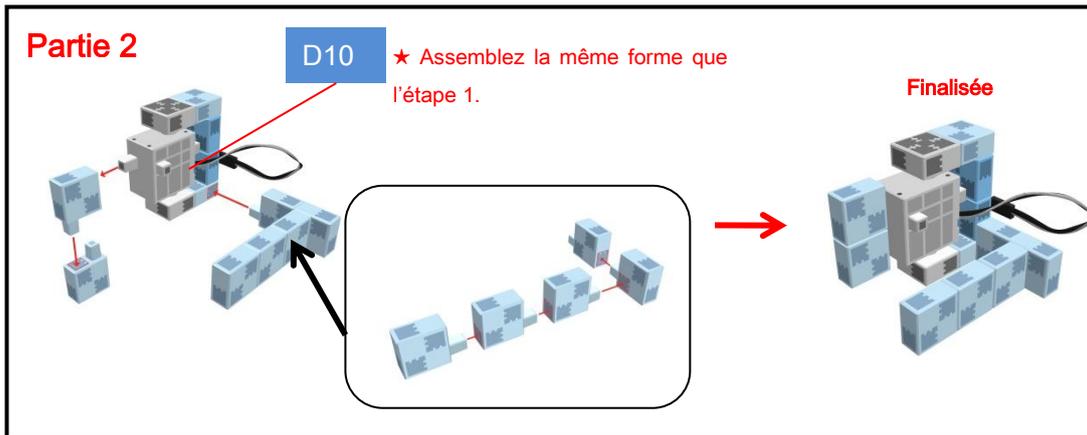
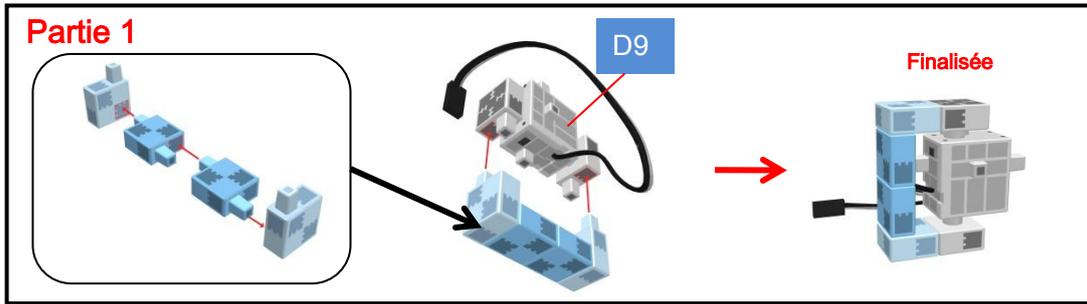
Le bloc peut contenir plusieurs blocs de servomoteurs, permettant ainsi aux servomoteurs de bouger à des angles spécifiques en même temps. Les autres processus ne peuvent pas être utilisés pendant que le servomoteur est en mouvement.

6.2. Faire un bras robotisé qui utilise trois servomoteurs

Maintenant, nous allons construire un bras robotisé utilisant trois servomoteurs et qui nous permettra de saisir un objet. Nous utiliserons les interrupteurs à bouton poussoir du circuit pour contrôler ce bras. Ce programme requiert l'utilisation des blocs «variables», «opérateurs», et «button».

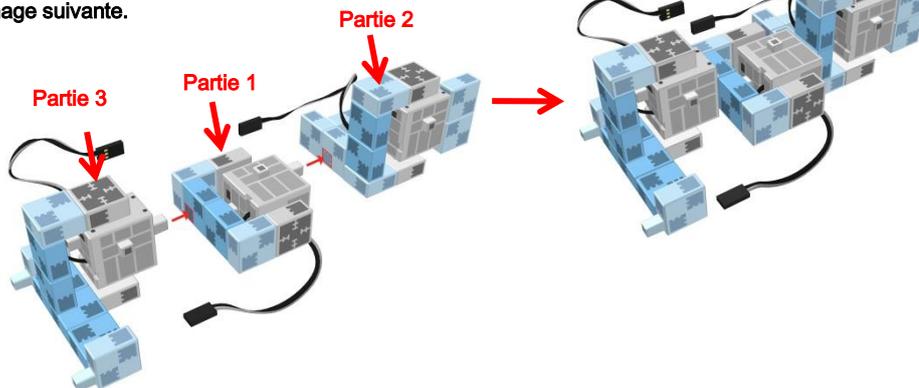
6.2.1. Assembler le robot

Assemblez le bras robotisé comme indiqué ci-dessous.

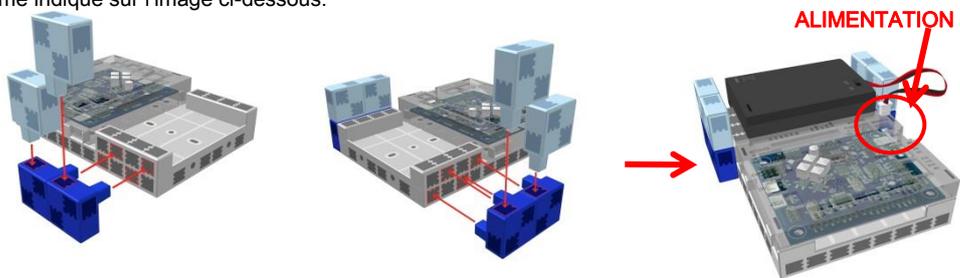


Assembler le bras

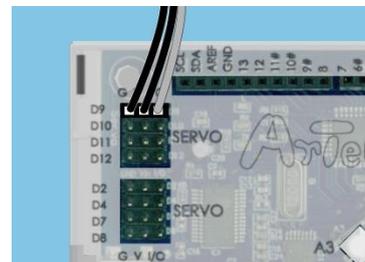
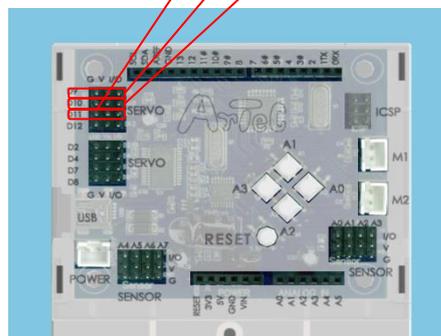
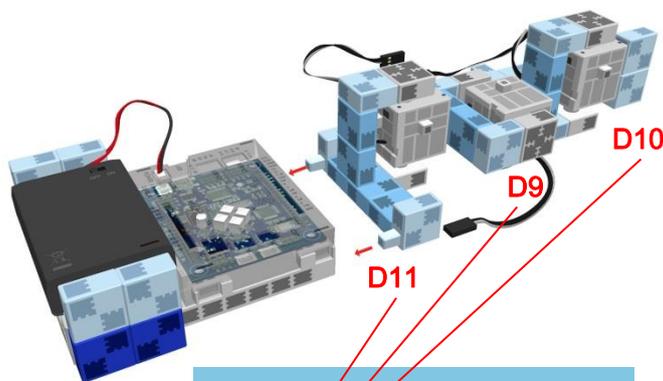
(1) Assemblez la partie 1 avec la 3 comme indiqué sur l'image suivante.



(2) Connectez le boîtier de piles à la surface de la carte de circuit imprimé en le connectant à l'alimentation comme indiqué sur l'image ci-dessous.



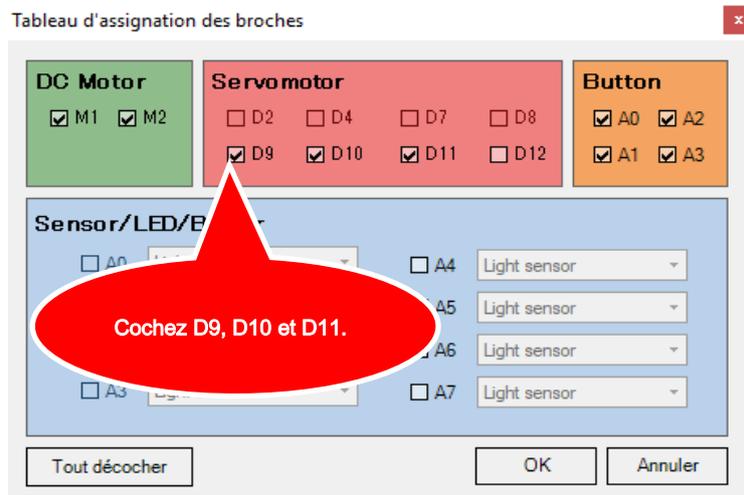
(3) Pour fixer le bras, connectez chaque servomoteur au connecteur.



Assurez-vous que les câbles de raccordement de capteur soient insérés correctement. Le câble gris doit être sur le dessus.

6.2.2. Paramétrage des ports

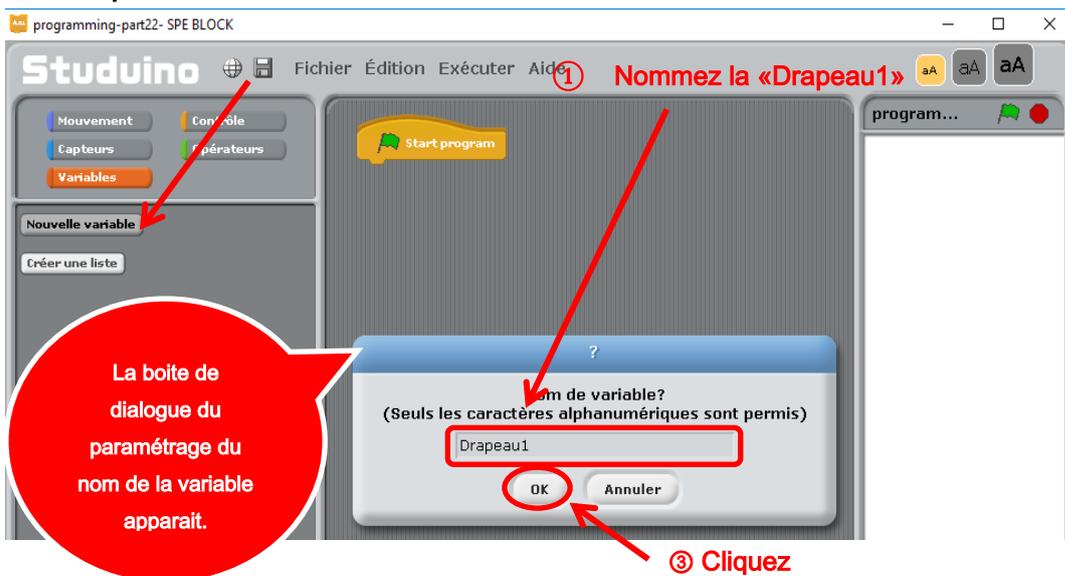
Avant de commencer à programmer, définissez les ports dans l'environnement de programmation par blocs pour qu'il s'accorde à votre circuit. Sélectionnez «**Édition**» dans le menu, puis dans le menu déroulant sélectionnez «**Paramètres des ports...**» pour accéder à la boîte de dialogue de réglages. Assurez-vous que les cases D9, D10 et D11 soient cochées dans la zone «Servomotor».



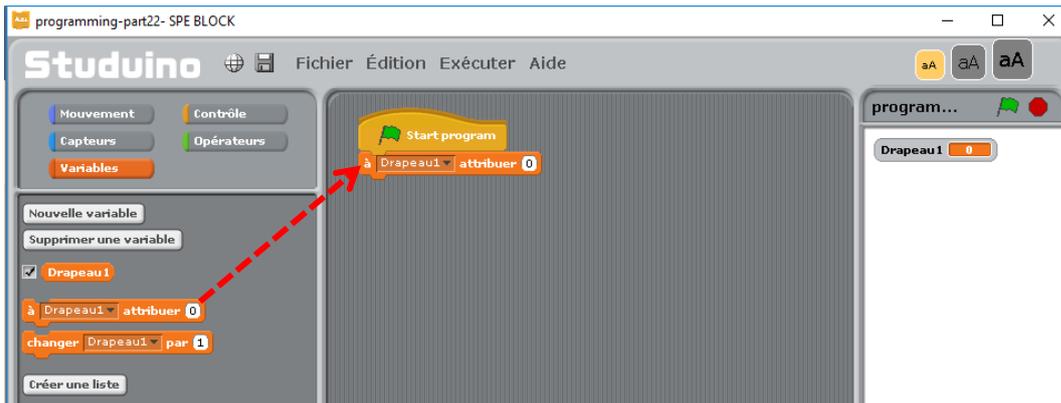
6.2.3. Programmer

Observez comment un servomoteur bouge.

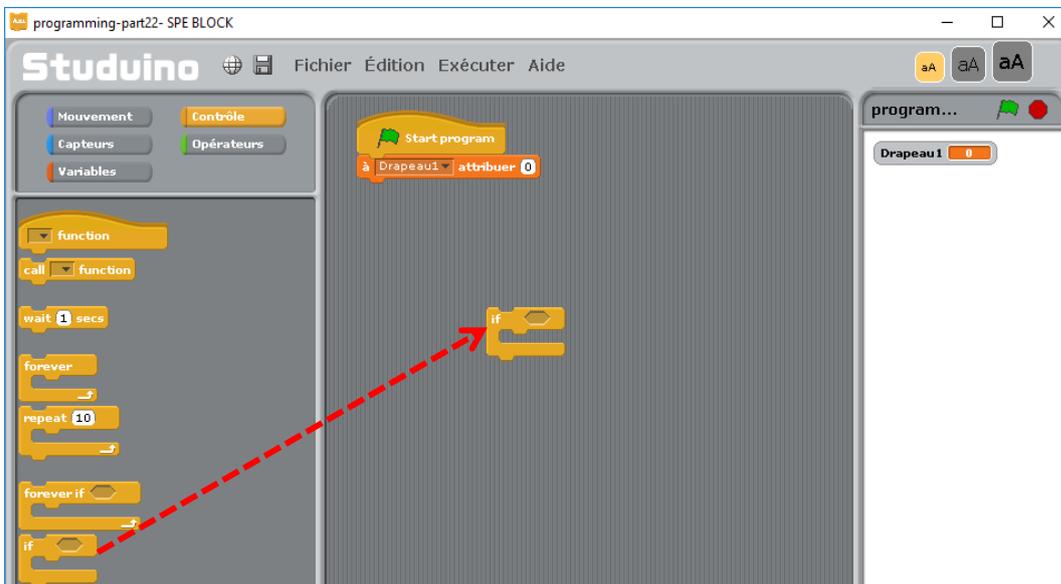
- 1 Cliquez sur le bouton «Nouvelle variable» dans la palette «Variables». Entrez «**Drapeau1**» comme nom de variable.



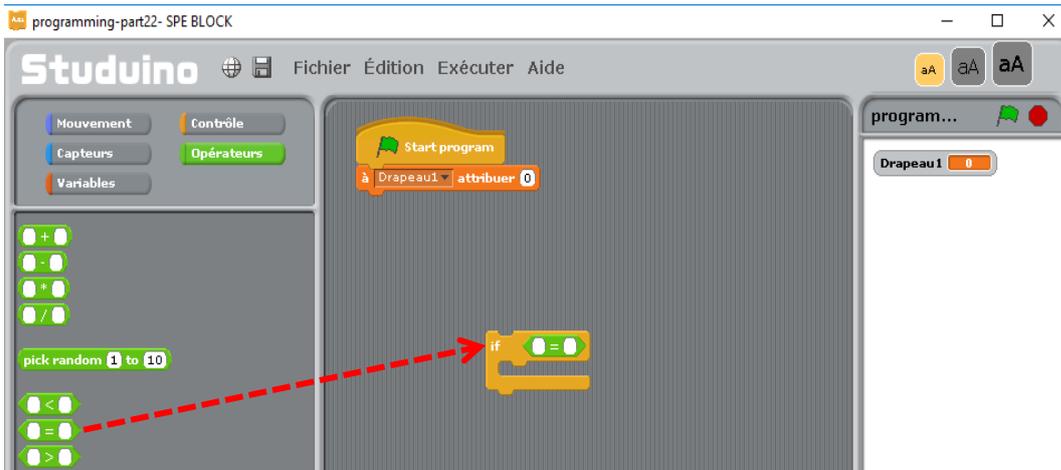
- 2 Faites glisser  de la palette «Variables» et connectez-le à .



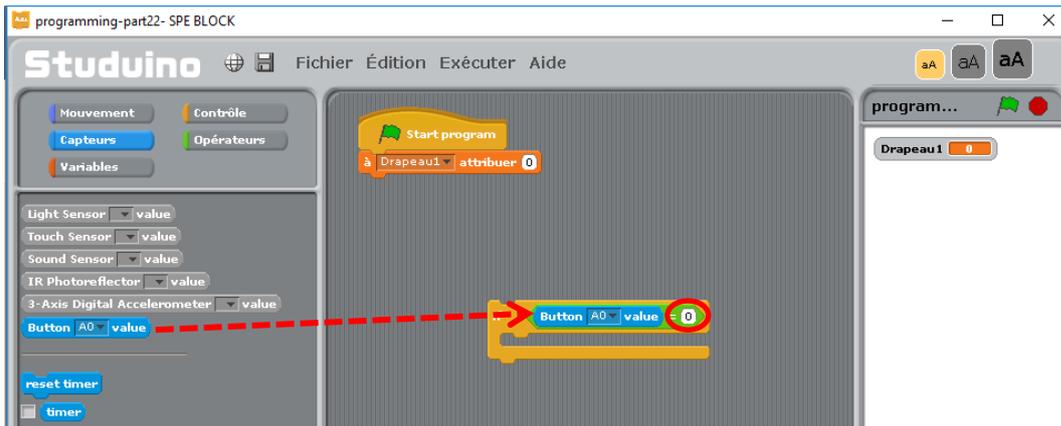
- ③ Maintenant, nous allons créer un processus qui change la valeur de «Drapeau1» lorsque le bouton A0 est touché. À partir de la palette «Contrôle», faites glisser  vers le champ script.



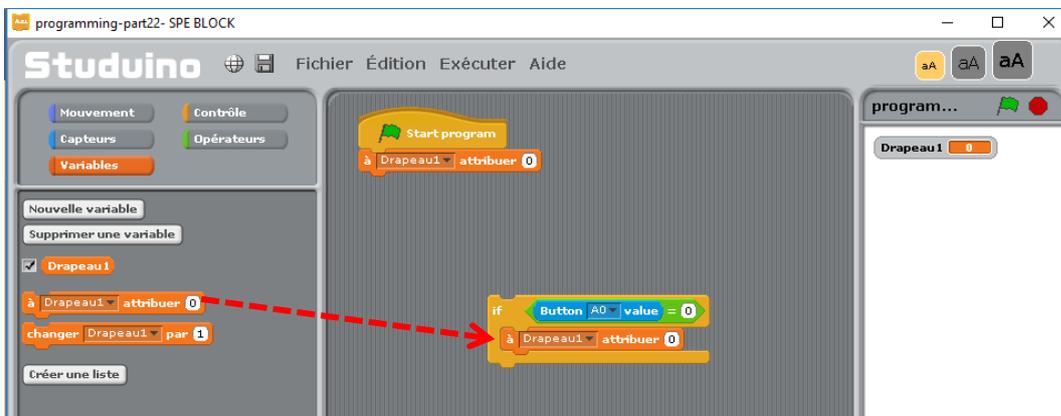
- ④ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



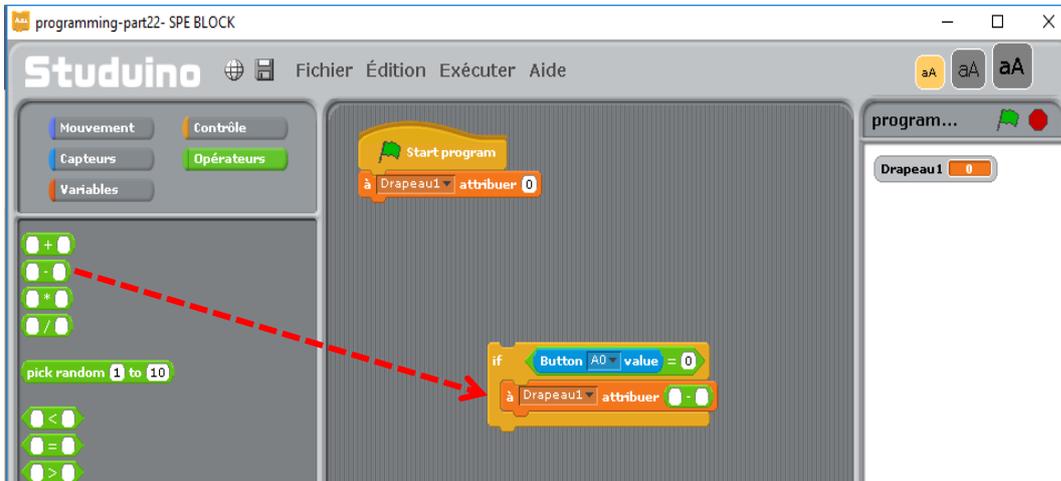
- ⑤ À partir de la palette «Capteurs», faites glisser  et attachez-le du côté gauche de . Ensuite, paramétrez le côté droit sur 0.



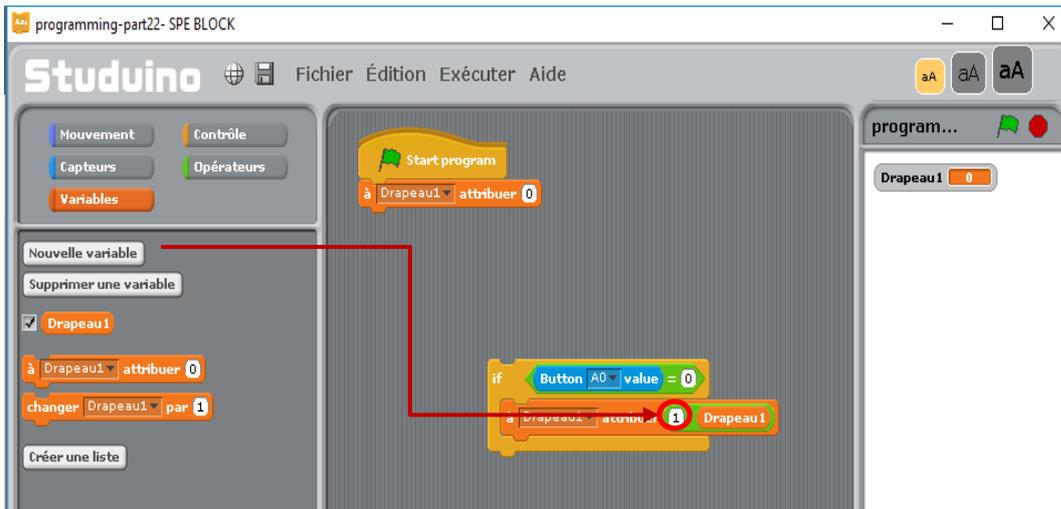
⑥ À partir de la palette «Variables», sélectionnez  et déplacez-le dans .



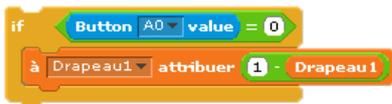
⑦ À partir de la palette «Opérateurs», faites glisser et attachez  comme indiqué.



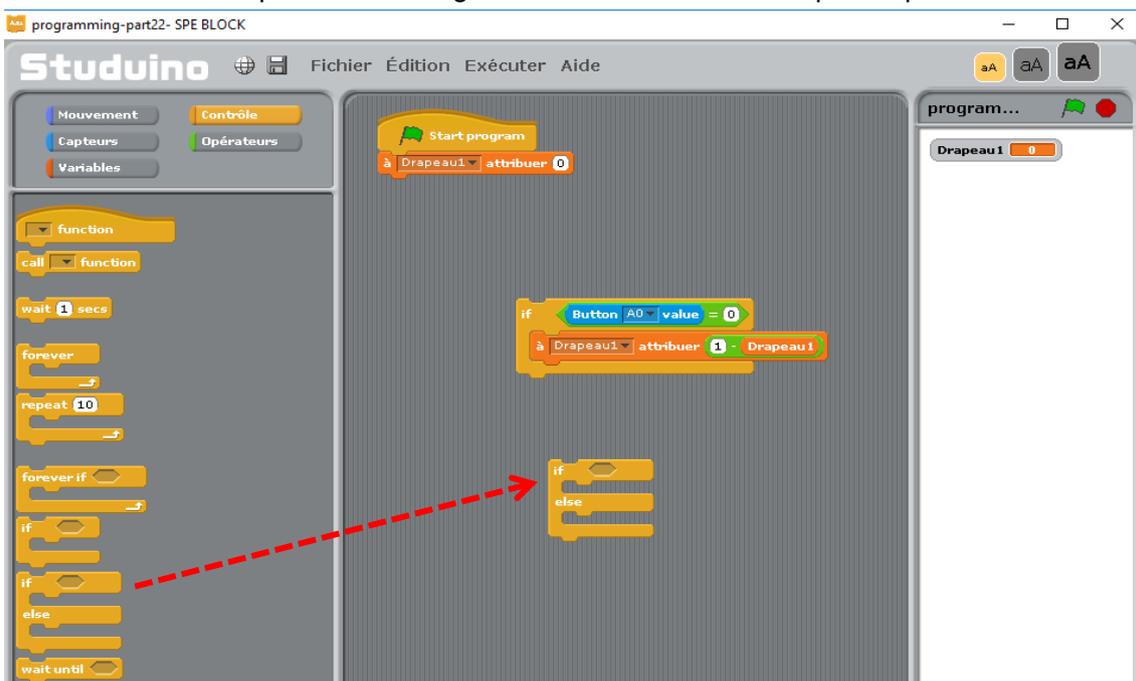
⑧ À partir de la palette «Variables», faites glisser et attachez  du côté droit de . Ensuite, paramétrez le côté gauche sur 1.



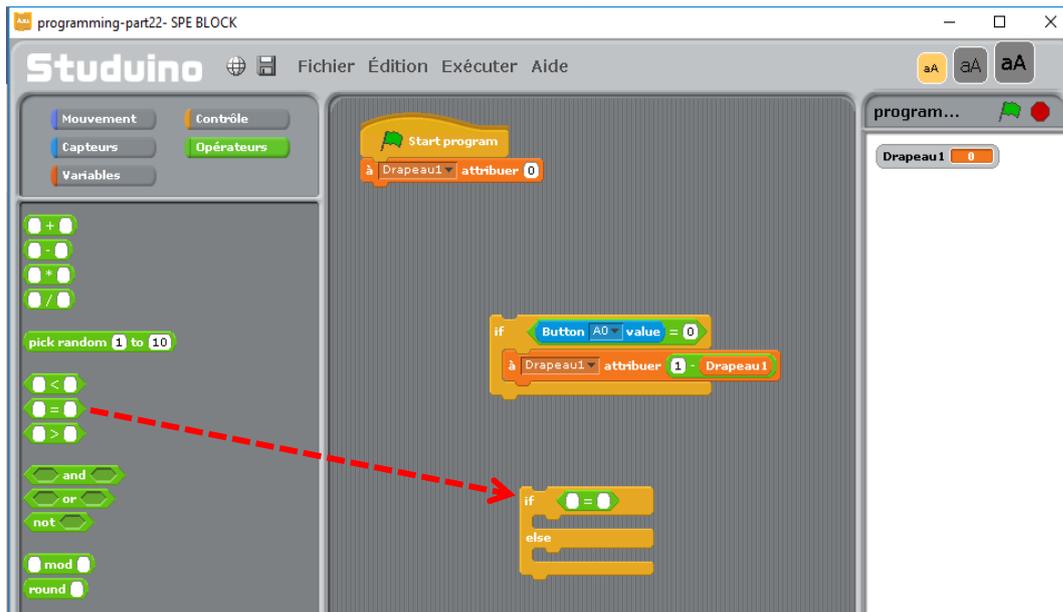
Le processus est maintenant complet. Ce dernier change la valeur de «Drapeau1» lorsque le bouton A0 est touché. Si la valeur de «Drapeau1» était égale à 1 lorsque le bouton était touché, alors la valeur changeait en 0. Si la valeur était égale à 0, alors elle devenait 1.



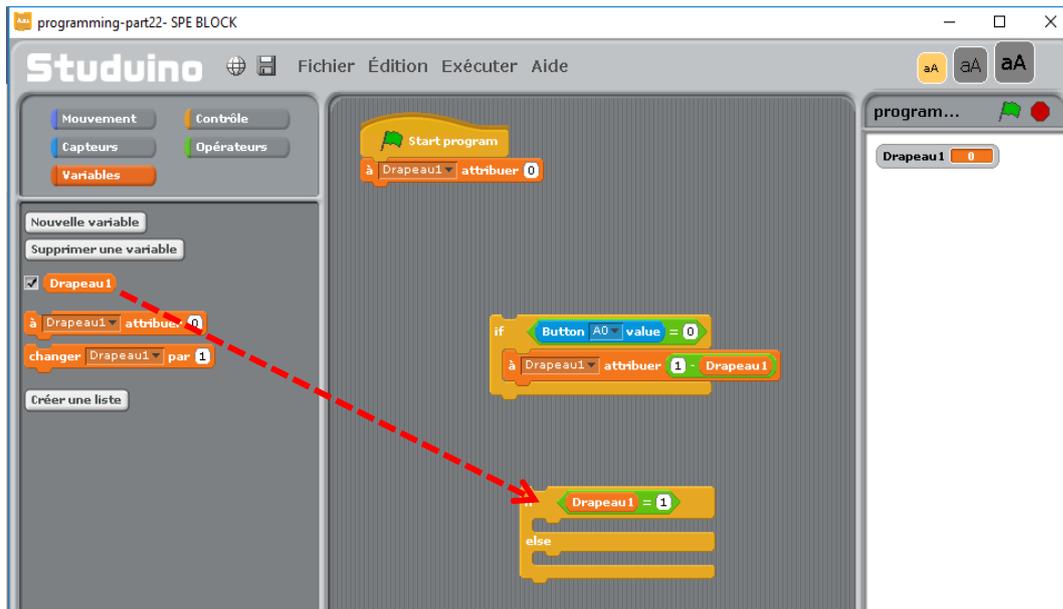
Maintenant, nous allons créer un processus qui change l'angle du servomoteur en fonction de la valeur de «Drapeau1». Faites glisser le bloc  depuis la palette «Contrôle».



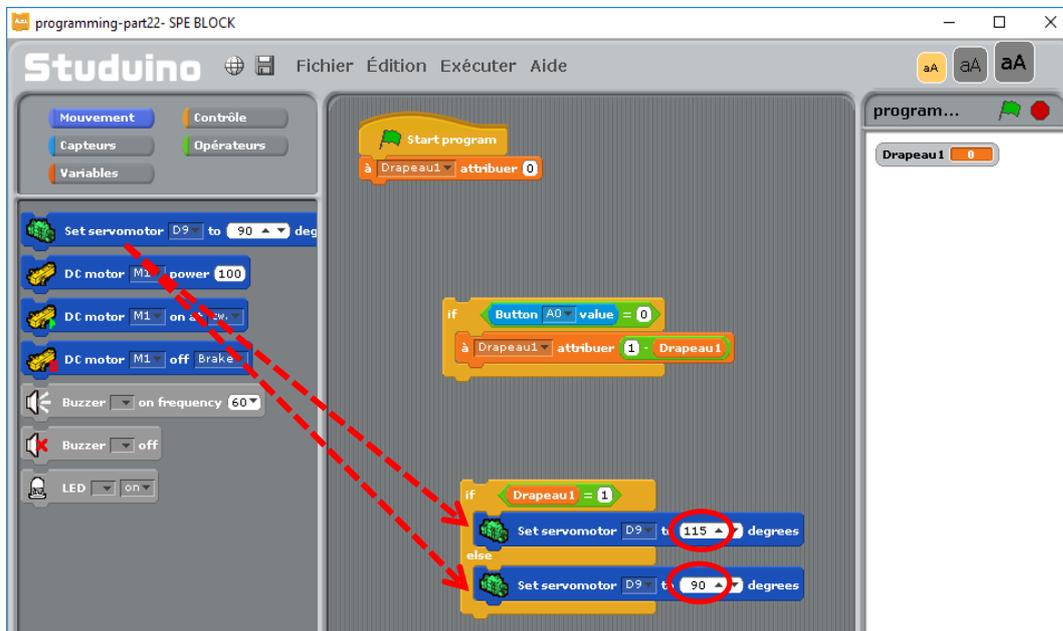
⑨ À partir de la palette «Opérateurs», faites glisser  vers le champ script.



- ⑩ À partir de la palette «Variables», faites glisser et attachez **Drapeau1** du côté gauche de . Ensuite, paramétrez le côté droit sur 1.

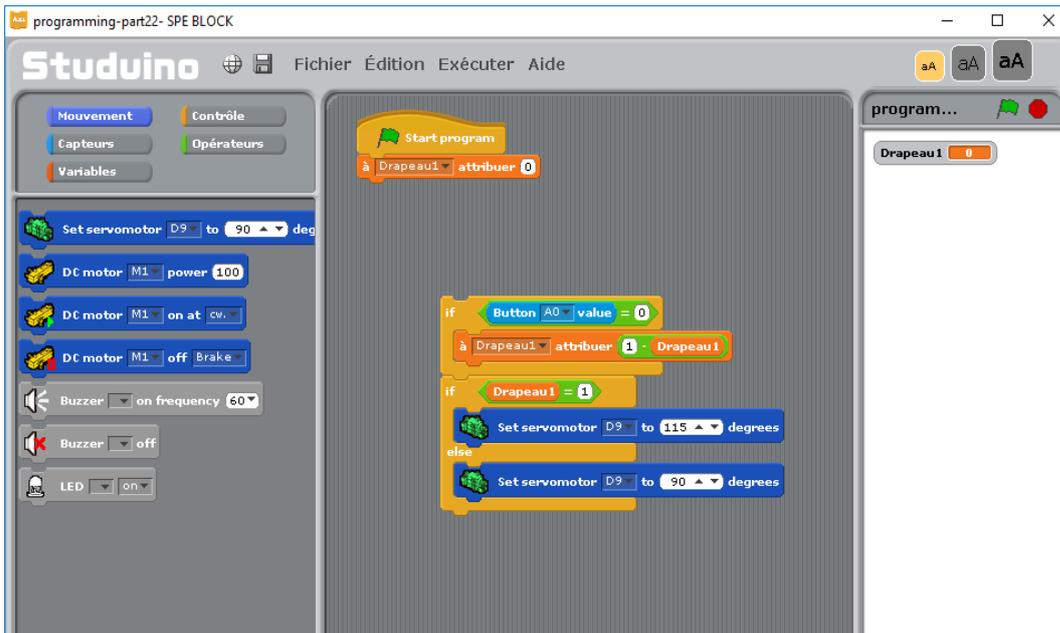


- ⑪ À partir de la palette «Mouvement», sélectionnez  et  et déplacez-le dans . Entrez les valeurs 115 et 90 dans les blocs attachés comme indiqué ci-après.

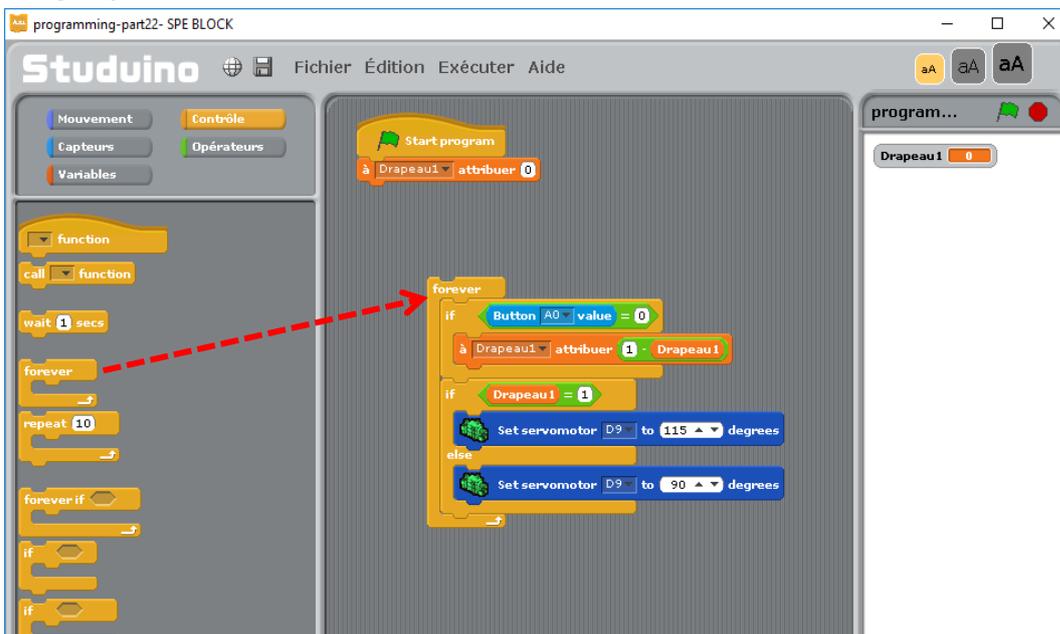


Le processus est maintenant complet. Ce dernier change l'angle du servomoteur en fonction de la valeur de «Drapeau1».

- ⑫ Connectez le groupe de blocs que vous venez de créer aux autres blocs comme indiqué.

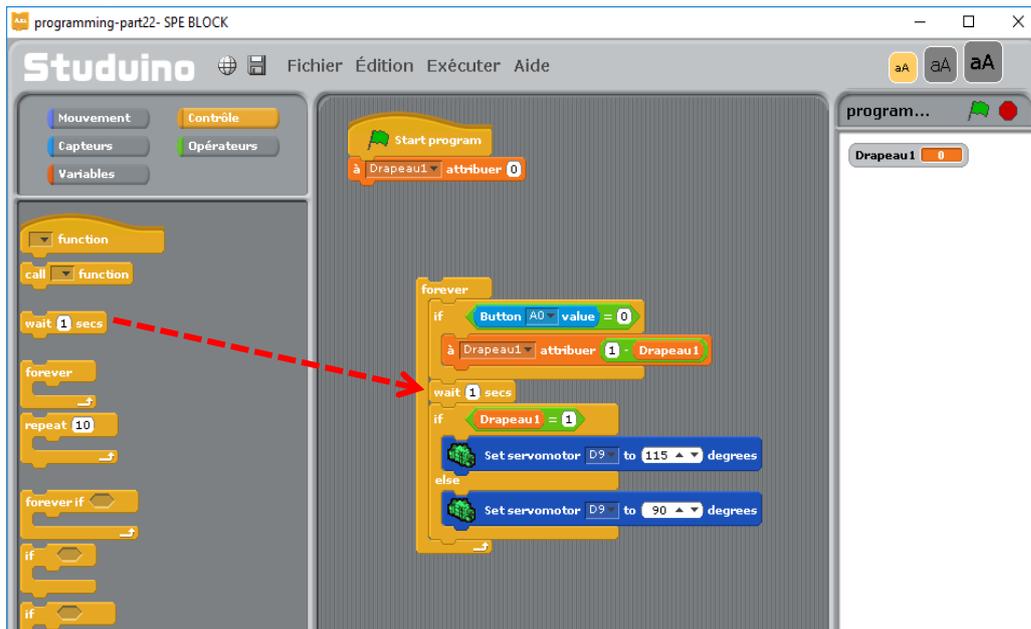


- ⑬ Faites glisser  depuis la palette «Contrôle» et utilisez-le pour entourer le groupe de blocs.

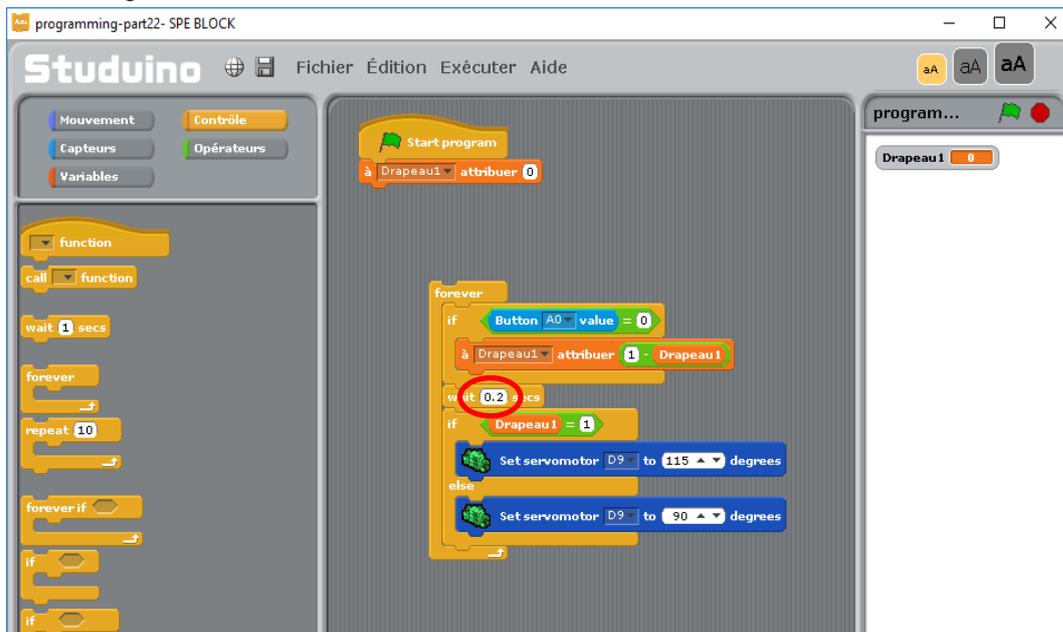


À l'étape précédente, vous avez créé un processus qui change l'angle du servomoteur lorsque l'interrupteur à bouton poussoir est touché. Ce programme utilise une fonction de boucle. Tous les processus sont exécutés aussitôt que la commande est donnée, puis passe immédiatement à l'étape suivante. Cela engendre la répétition en continu du processus tant que le bouton est touché. Chaque boucle change la valeur de «Drapeau1» entraînant ainsi un nouvel angle du servomoteur.

- ⑭ Faites glisser **wait 1 secs** de la palette «Contrôle» et insérez-le dans la position indiquée ci-dessous.

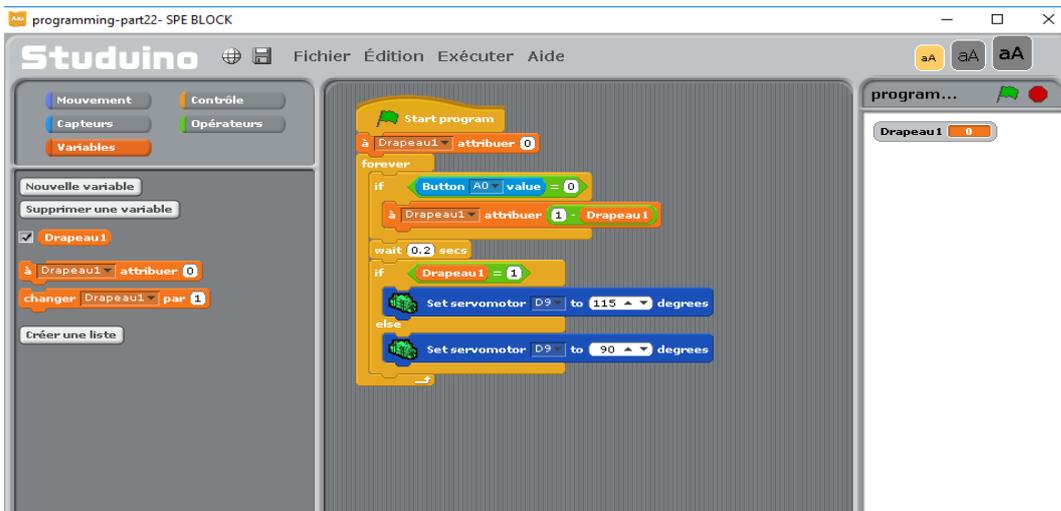


- ⑮ Changez la durée de **wait 1 secs** à 0.2 seconde.



Ainsi, toutes les 0.2 secondes, l'état de l'interrupteur à bouton poussoir sera vérifié.

⑯ Connectez le groupe à **à Drapeau1 attribuer 0** comme indiqué.

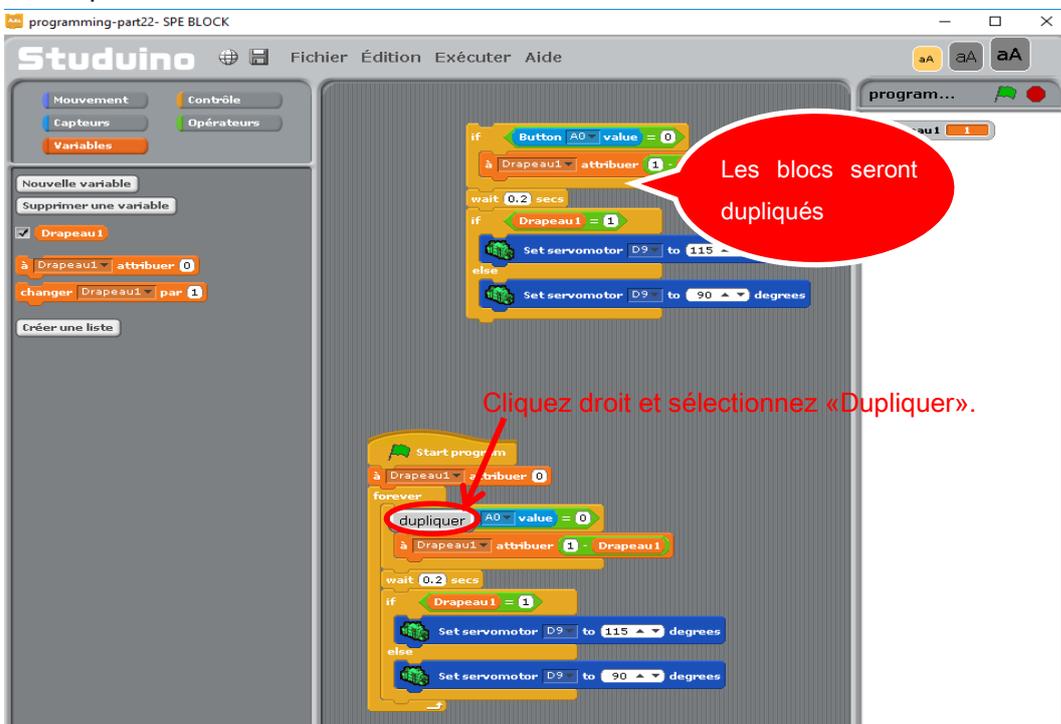


Le programme est maintenant complet. Ce dernier change l'angle du servomoteur connecté à D9 de 115 degrés à 90 lorsque l'interrupteur à bouton poussoir est touché.

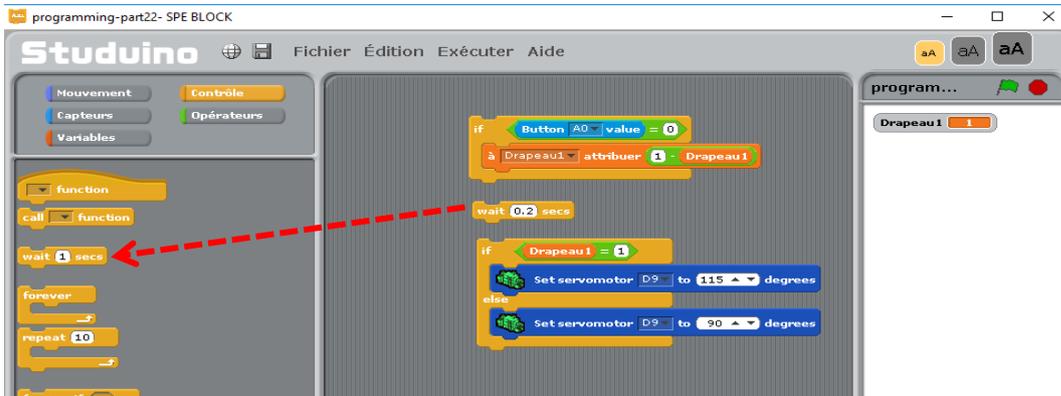
⑰ Choisissez «**Transférer**» depuis le menu «**Exécuter**» pour observer comment votre programme fonctionne.

⑱ Répétez les étapes ci-dessus pour recréer le même programme pour les deux autres servomoteurs. Faites glisser le programme terminé vers le bas et dupliquez le bloc en

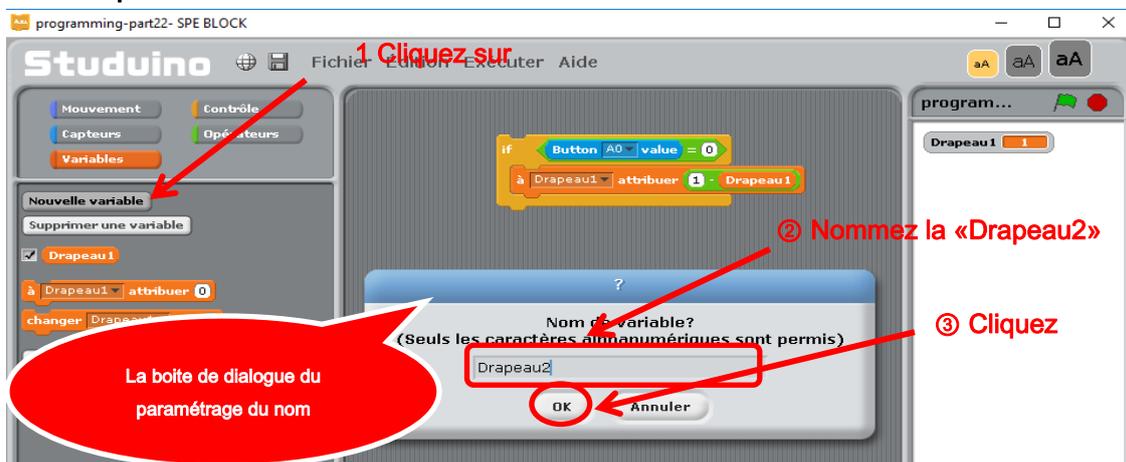
cliquant droit sur le bloc



- ⑱ Séparez les blocs dupliqués. Faites glisser et relâchez **wait 1 secs** dans la palette «Contrôle».



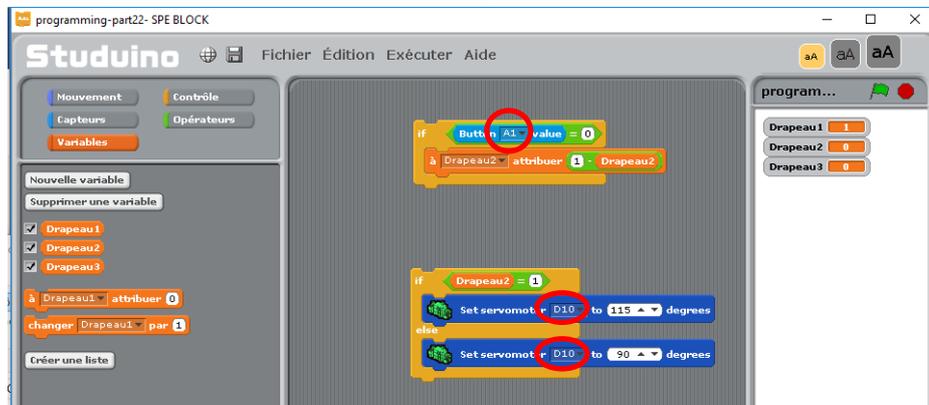
- ⑳ Cliquez sur le bouton «Nouvelle variable» dans la palette «Variables». Entrez «Drapeau2» comme nom de variable.



- 21 Remplacez le nom «Drapeau1» par «Drapeau2» dans le bloc dupliqué.



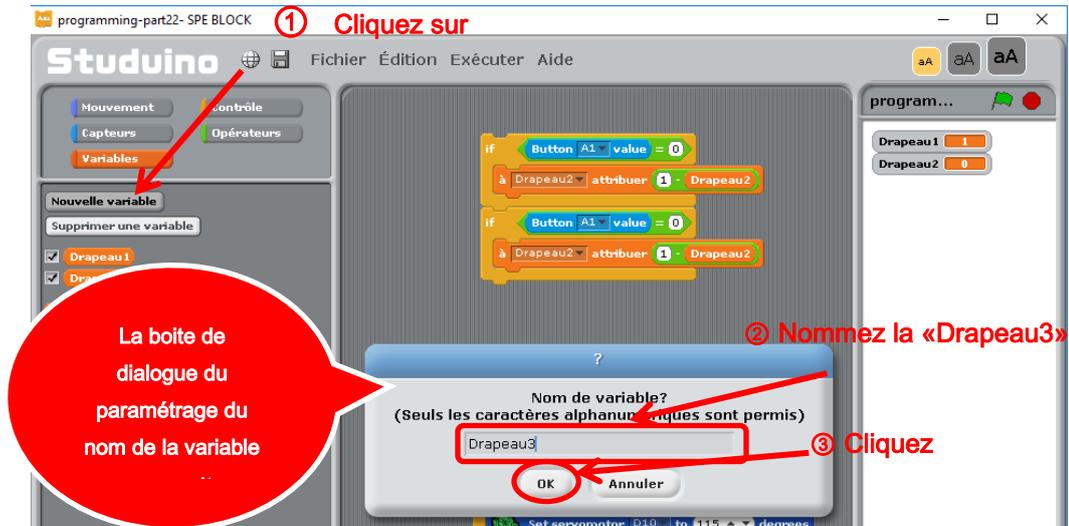
- 22 Faites glisser **Drapeau2** sur le côté droit de **0-0**. Le bloc déjà présent réapparaîtra. Toutefois il est préférable de le supprimer en faisant un clic-droit dessus. Ou vous pouvez simplement le faire glisser dans la zone palette pour le supprimer.
- 23 Dupliquez le bloc **Set servomotor D9 to 90 degrees** afin de faire deux nouvelles duplications des blocs puis attachez-les aux blocs originaux **Button A0 value**.



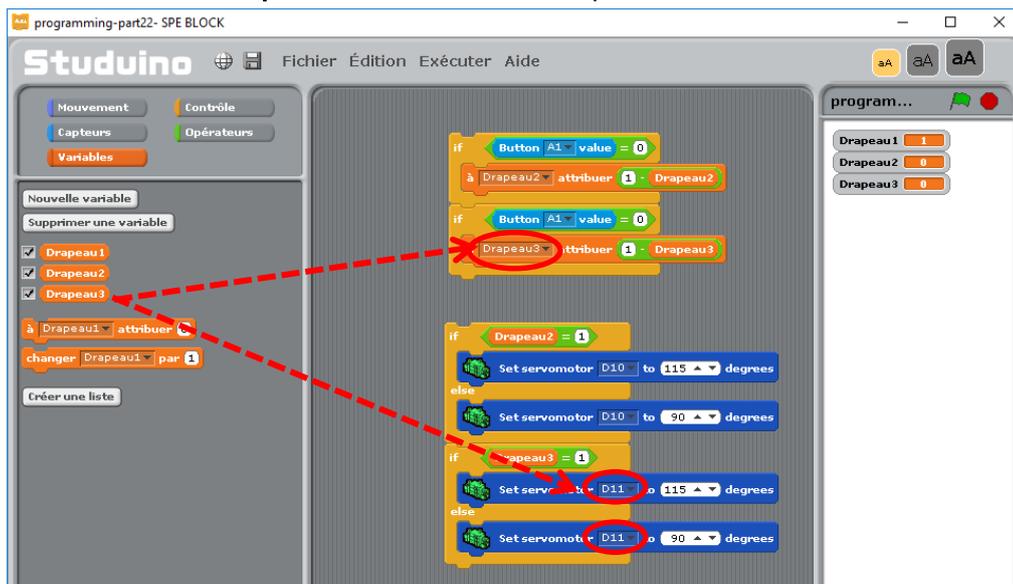
- 24 Dupliquez de nouveau les deux blocs venant d'être dupliqués et attachez-les aux blocs originaux.



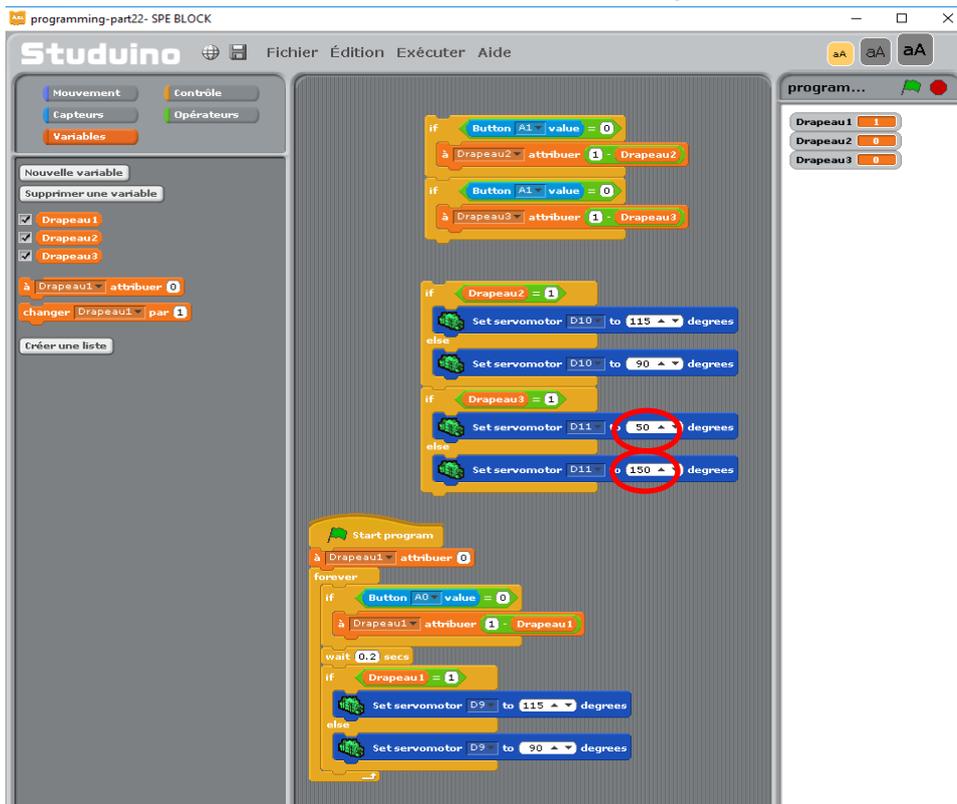
25 Paramétrez le troisième servomoteur. Créez une variable «Drapeau3».



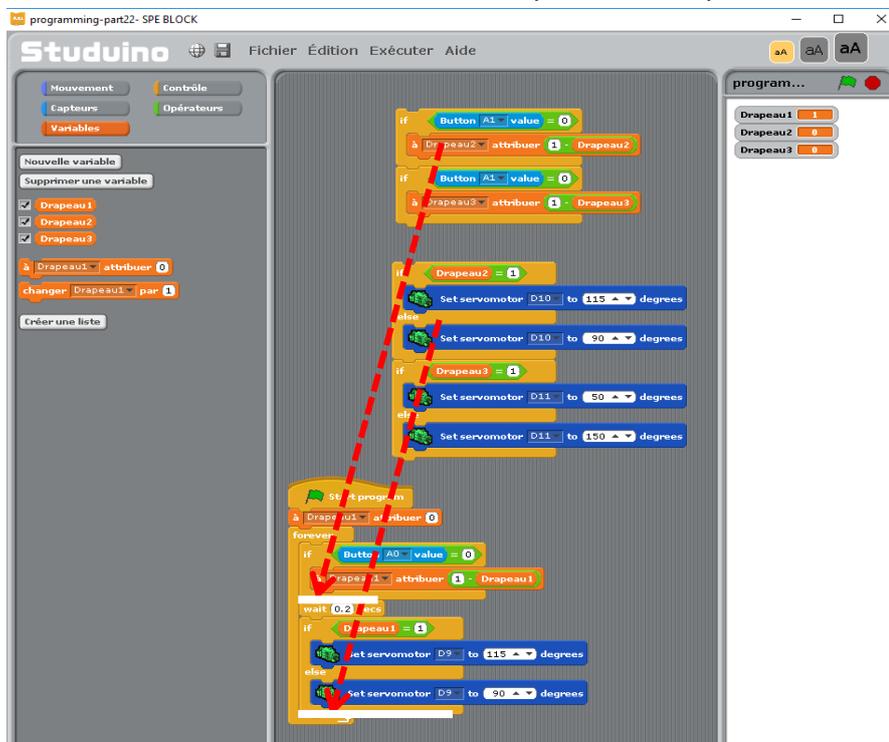
Suivez exactement les mêmes étapes que pour le servomoteur deux. Toutefois, nommez cette fois-ci la variable «Drapeau3» et choisissez D11 pour le servomoteur.



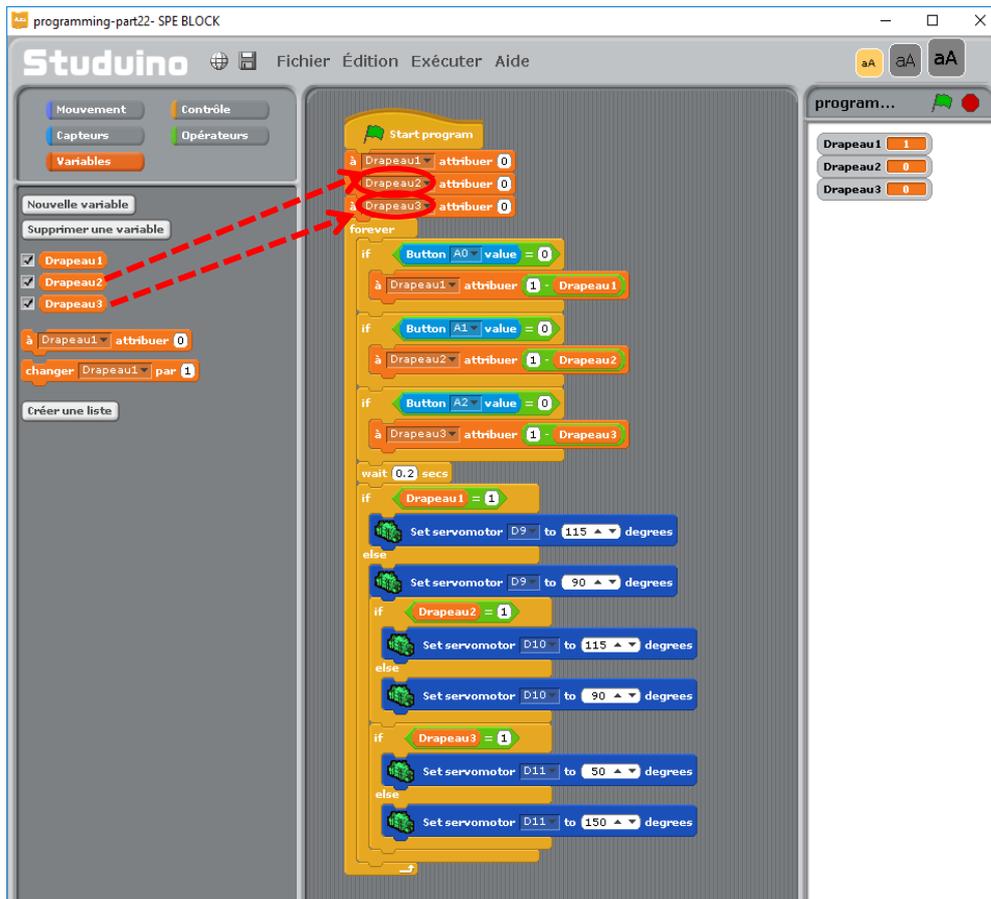
26 Changez les variables des blocs du servomoteur «Drapeau3» à 50 et 150 degrés.



27 Attachez les blocs de manière similaire aux positions indiquées ci-dessous.



28 Faites glisser deux blocs  et attachez-les aux positions indiquées ci-dessous. Sélectionnez «Drapeau2» et «Drapeau3».



Le programme est maintenant complet. Ce dernier change l'angle des servomoteurs connectés à D9, D10 et D11 lorsque les interrupteurs à bouton poussoir A0, A1 ou A2 sont touchés. Choisissez «**Transférer**» depuis le menu «**Exécuter**». Lorsque le transfert sera terminé, toucher A0, A1, ou A2 fera bouger le bras.