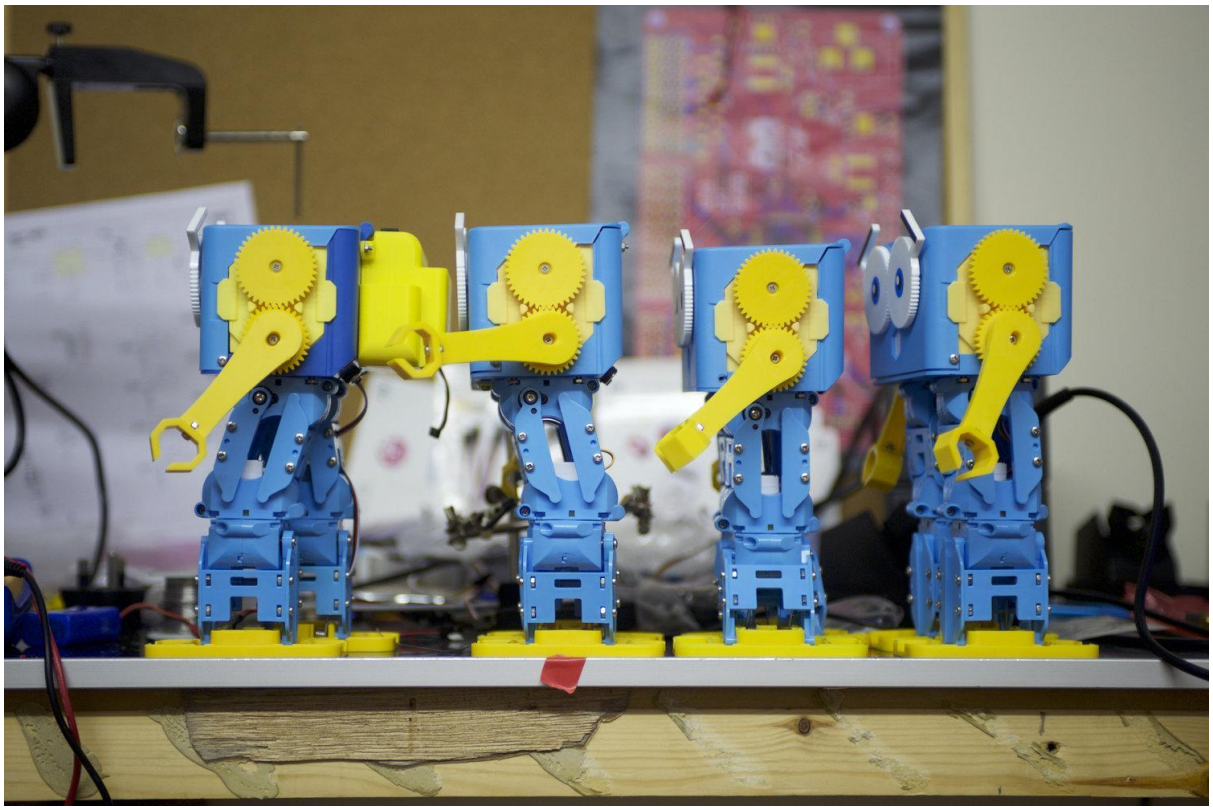


ROBOT MARTY

Introduction à la programmation en Scratch

Traduit par Génération Robots

Source : https://robotical.io/

Ce document explique de manière progressive comment prendre en main le robot Marty à travers la programmation sous Scratch.

Quelques vidéos vous permettront de voir la réalisation par le robot des différents extraits de code :
https://www.youtube.com/playlist?list=PLiBgsWjIRfInizbzagjLnoQjoeb0psS33

Dans la version PDF de ce guide, vous pourrez y accéder directement en cliquant sur le lien “Video” dans chaque section.

De nouveaux projets pour débutants vont être publiés prochainement, et devraient apparaître sur www.robotical.io/scratch rapidement.

NOTES

Pour pouvoir contrôler votre robot Marty via l'interface Scratch, il faut vous assurer que votre ordinateur et votre robot sont bien connectés au même réseau Wifi. Vous trouverez toutes les informations nécessaires dans le guide de démarrage (à part).

I.Commandes basique

Habituez-vous à contrôler votre robot Marty avec Scratch en jouant avec les blocs et fonctions intégrés.

1.Get Ready

Le bloc "Get Ready" activera les moteurs, redonnera sa position initiale à Marty et lui fera bouger les sourcils. Le bloc "Turn off motors" coupera les moteurs. Il est préférable de le placer à la fin du script.

2.Chaîne de commande plus longue (Vidéo)

3.Boucles (Vidéo)

Parfois vous voulez faire la même chose plusieurs fois, mais copier/coller le code ne fait pas propre et prends beaucoup de temps. Amusez-vous avec le bloc “repeat” afin de vous habituer aux boucles.

4.Définir une fonction (Vidéo)

L’autre façon de ne pas répéter du code est de créer une “fonction” à l’aide du bloc Define.

II.Application : Faites marcher Marty !

Il y a une fonction de marche déjà intégrée, mais ce serait trop facile... Pour cet exercice, vous devez apprendre à Marty à marcher. Vous allez devoir penser à l'équilibre et comment chaque articulation doit bouger afin de créer un mouvement complexe.

1.Faites un pas (Vidéo)

Le mouvement de base pour faire un pas - dans ce cas là, avec la jambe droite.

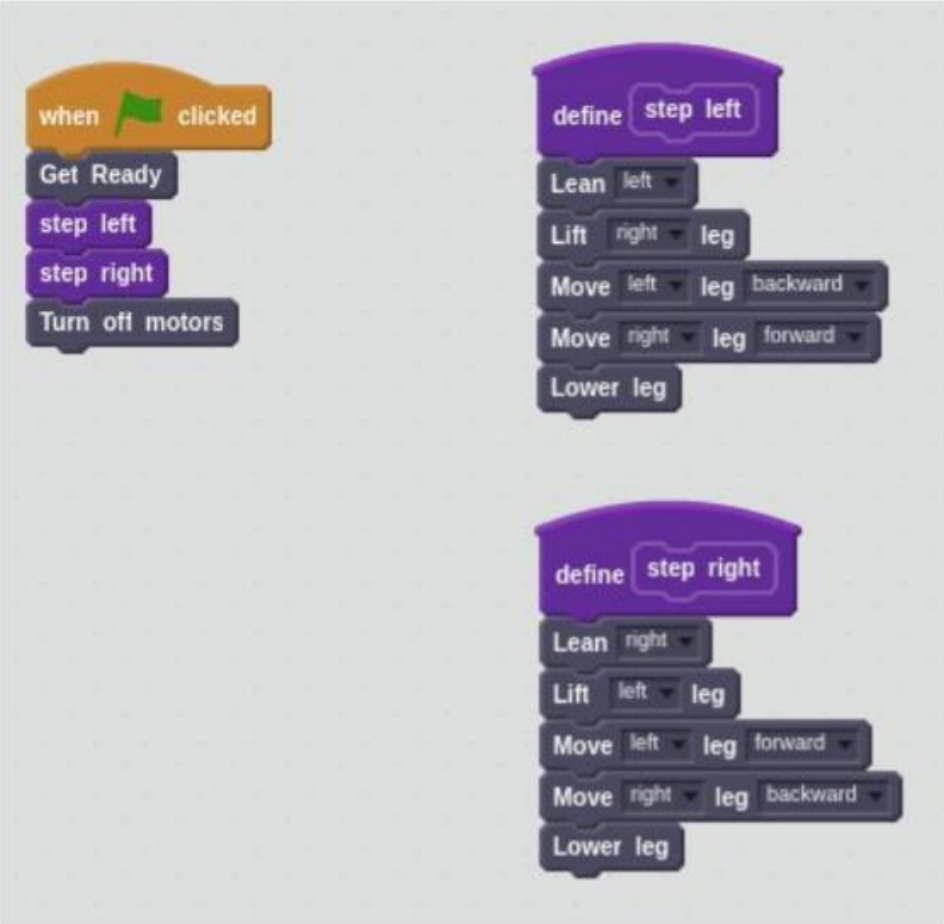
2.Faites-en deux (Vidéo)

Puis ajouter un pas avec l'autre jambe

.

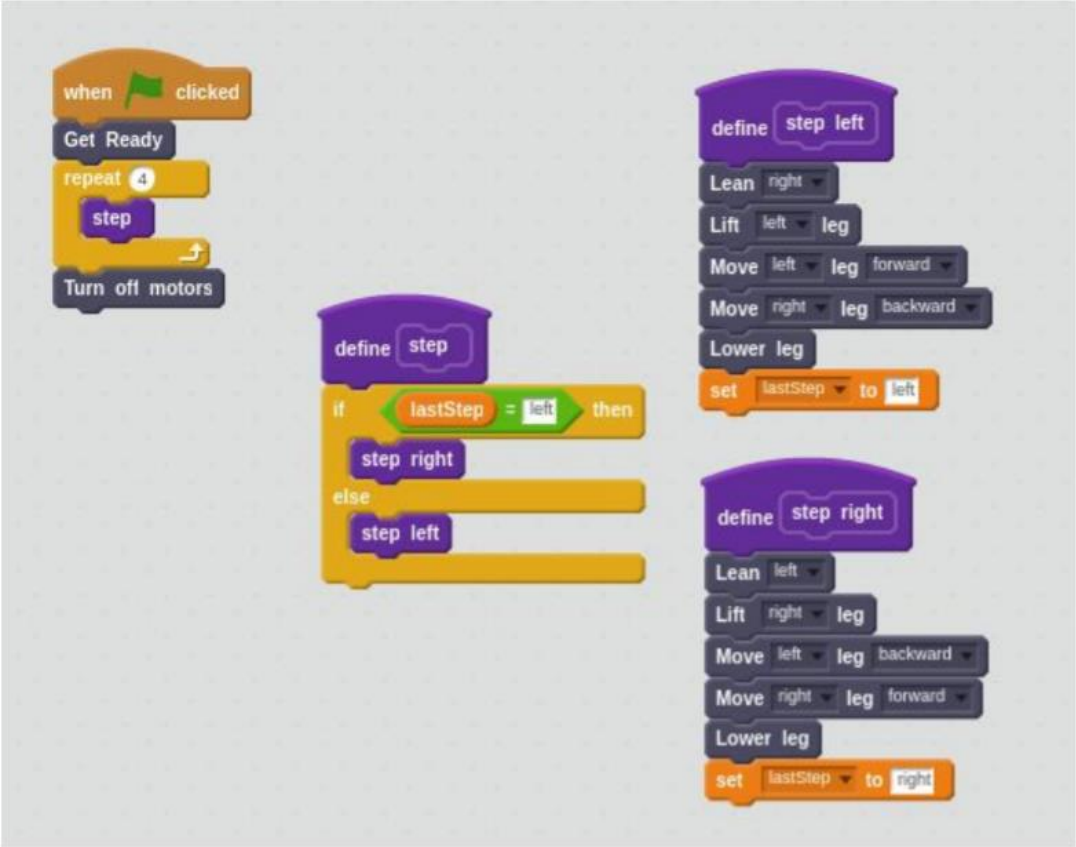
3.Créez des fonctions ! (Vidéo)

Déplacez chaque pied individuellement avec sa propre fonction.



4.Une seule fonction pour tous les bouger ! (Vidéo)

Il serait plus élégant de simplement dire “step” plutôt que spécifier à chaque fois quelle jambe doit bouger. Cela peut être fait en créant une variable qui garde une trace de quel pied a été déplacé en dernier.



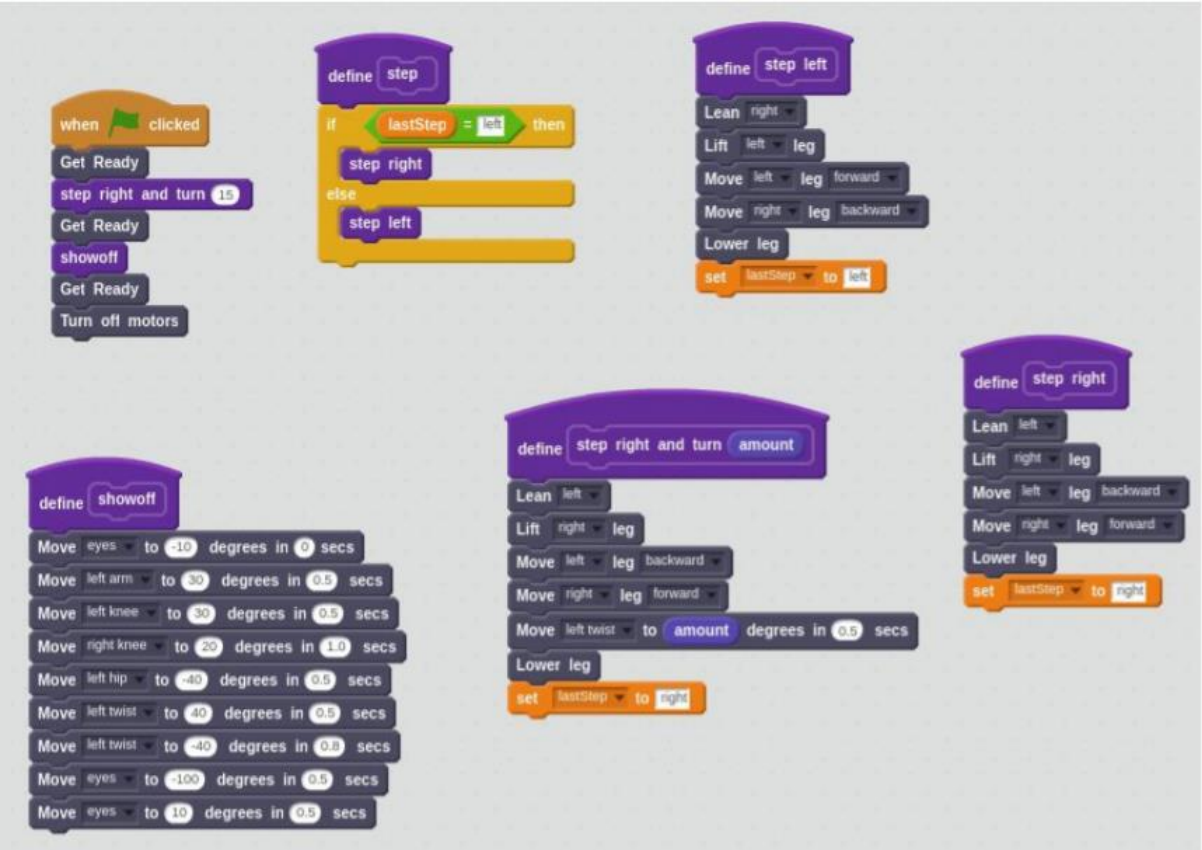
III. Aller plus loin : créez vos propres mouvements

1.Le bloc Move Joint (Vidéo)

Ce bloc permet un contrôle plus fin de chaque articulation, et donc de réaliser des mouvements personnalisés. Amusez-vous avec le bloc “Move Joint” afin de créer un mouvement plus complexe !

2.Un tournant décisif (Vidéo)

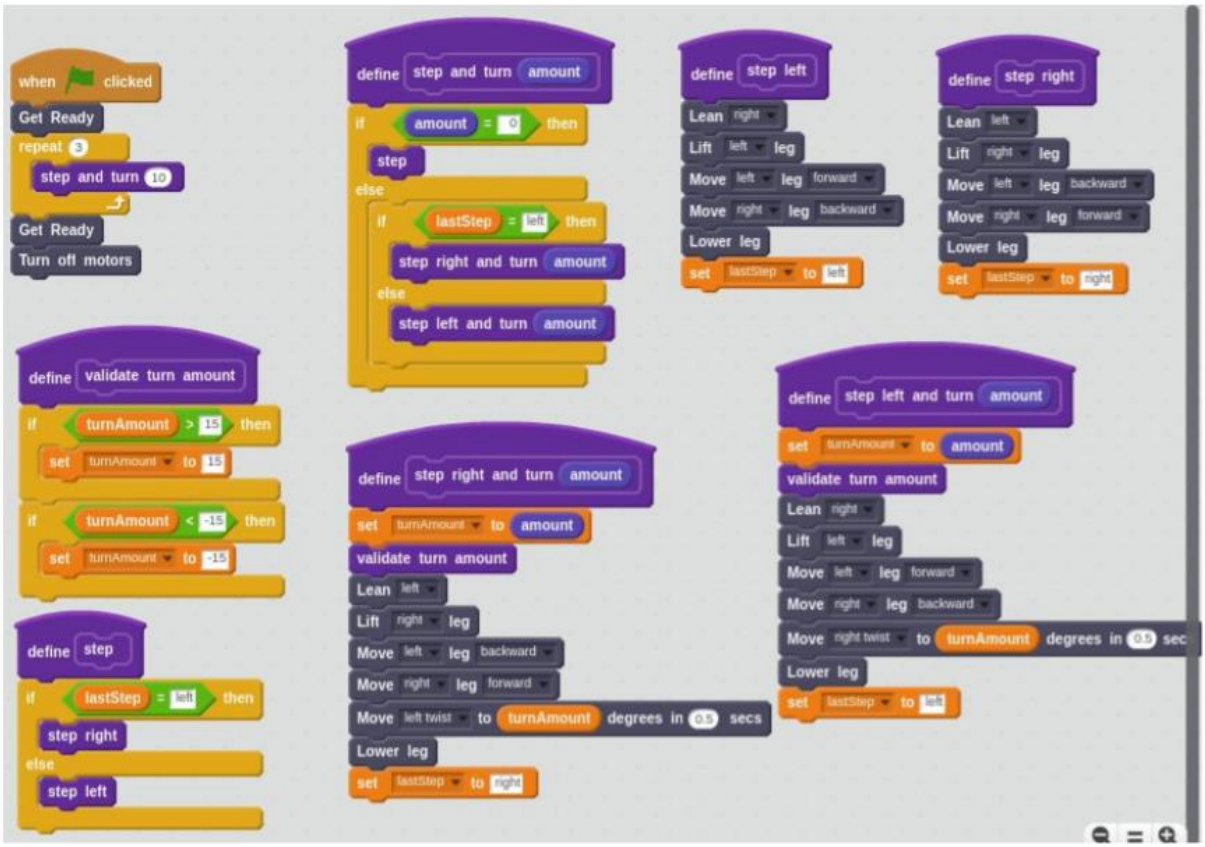
Vous allez créer une fonction permettant au robot de tourner sur lui-même. Cette fonction aura un argument indiquant de combien le robot doit tourner.



3.Validation de paramètres et optimisation du code (Vidéo)

Plusieurs choses se produisent dans ce code :

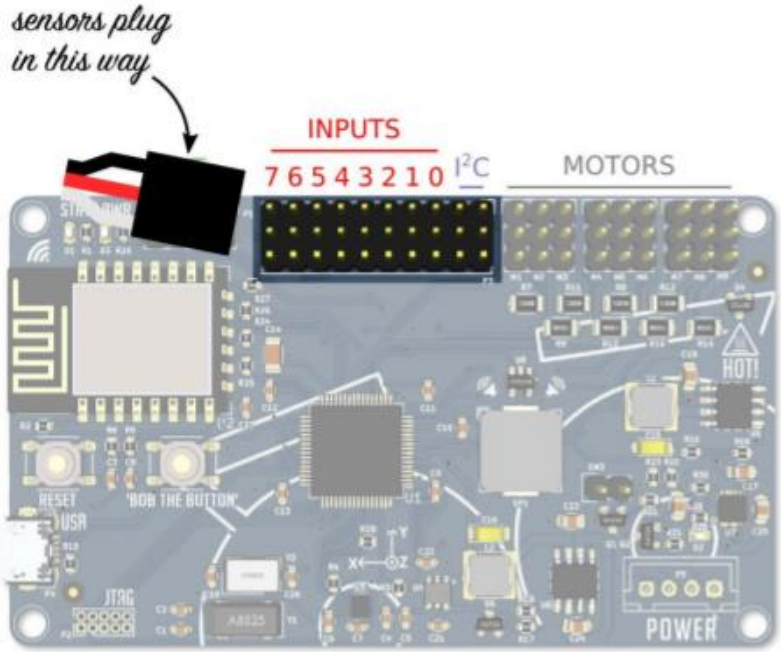
- Validation de paramètres : si vous dites à Marty de trop tourner, ses pieds se heurteront l'un à l'autre et ce ne sera pas joli-joli. Afin d'éviter d'envoyer une mauvaise commande, on peut "limiter" la dose du virage.
- Création de fonctions plus générales. Les fonctions peuvent appeler d'autres fonctions - dans ce cas la fonction "step and turn" englobe la fonction "step", en plus d'ajouter quelques fonctionnalités.



IV.Capteurs

Le robot Marty a plusieurs capteurs intégrés - capteur de courant électrique sur la plupart des moteurs, accéléromètre pour calculer l'inclinaison et l'accélération, et un ensemble de ports auxquels des capteurs comme des capteurs de contact peuvent être ajoutés.

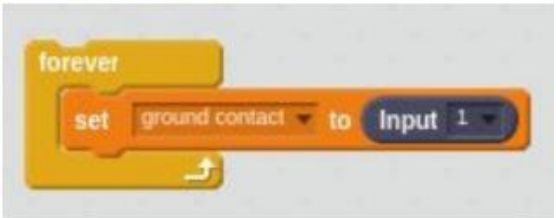
Le schéma de câblage électronique ci-après montre comment les capteurs peuvent être fixés, mais ces exemples supposent qu'un capteur de contact connecté au port 0 se trouve à l'avant d'un pied, et qu'un capteur de contact connecté au port 1 se trouve en bas d'un pied.



Connecter les capteurs au robot Marty. Les câbles vont avec le fil noir en haut, et les nombres correspondent au bloc "Input *n*" dans Scratch.

1.Lire la valeur d'un capteur (Vidéo)

Le bloc "Input" vous laissera lire l'état actuel d'un switch. En assignant la lecture d'un capteur à une variable, vous pouvez observer le résultat dans Scratch.
Le switch lira "1" lorsqu'il est activé, et "0" lorsqu'il ne l'est pas.



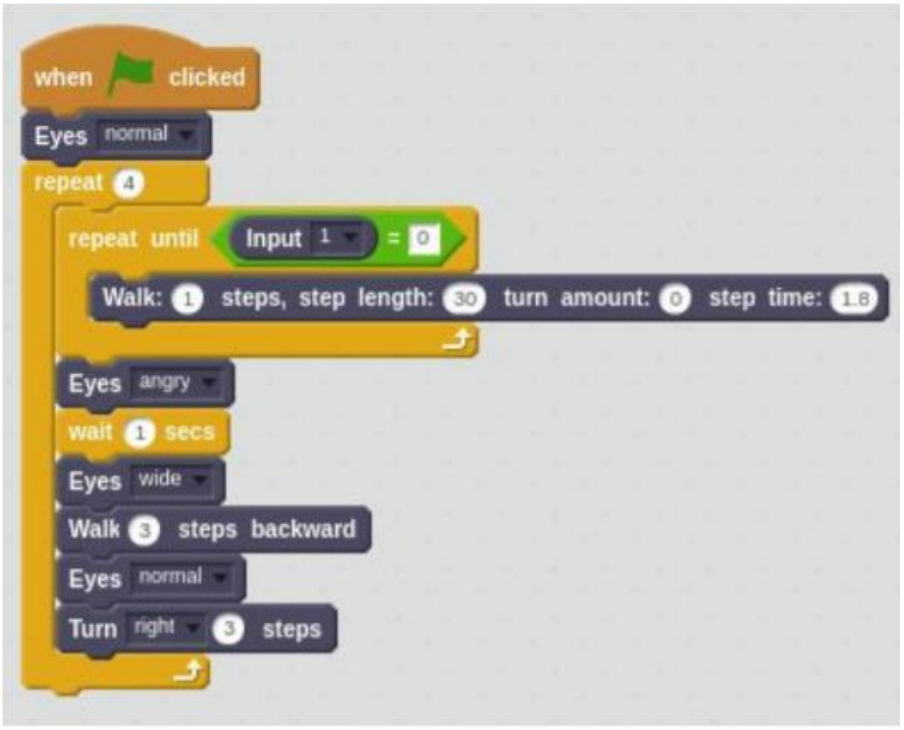
2.Marcher jusqu'au bord (Vidéo 1) (Vidéo 2)

Cet exemple part du principe qu'il y a un capteur de contact sous l'un des pieds et connecté au port 1.
A la fin de chaque pas, le switch peut être vérifié pour voir s'il est activé. S'il est à "1", le pied touche le sol, s'il est à "0" alors il ne le touche pas !

Nous pouvons utiliser ce code pour voir si Marty est arrivé à un rebord.

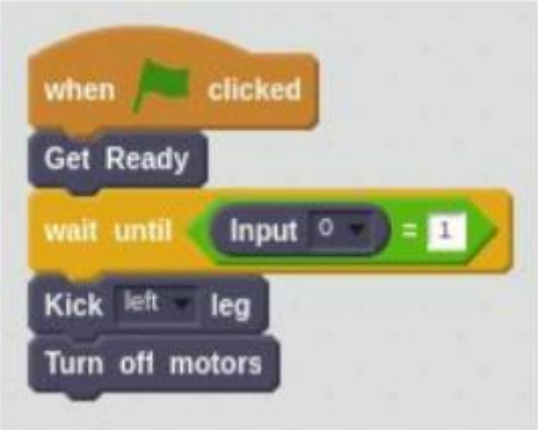
3.Revenir du bord (Vidéo)

Faire de grands yeux quand on arrive au bout de la table, c'est bien joli, mais ça ne nous aide pas beaucoup. Ajoutez quelques blocs pour faire faire quelque chose d'utile à Marty lorsqu'il arrive à un bord. Dans cet exemple, il reculera et se tournera avant de reprendre sa marche.



4.Réagir à la pression d'un bouton (Vidéo)

Dans cet exemple, on suppose la présence d'un capteur de contact au bout d'un pied, et connecté au port 0.
Il s'agit d'un exemple simple d'une réaction à un bouton poussé. Marty attendra jusqu'à ce que quelque chose touche le capteur de contact et donnera un coup de pied.



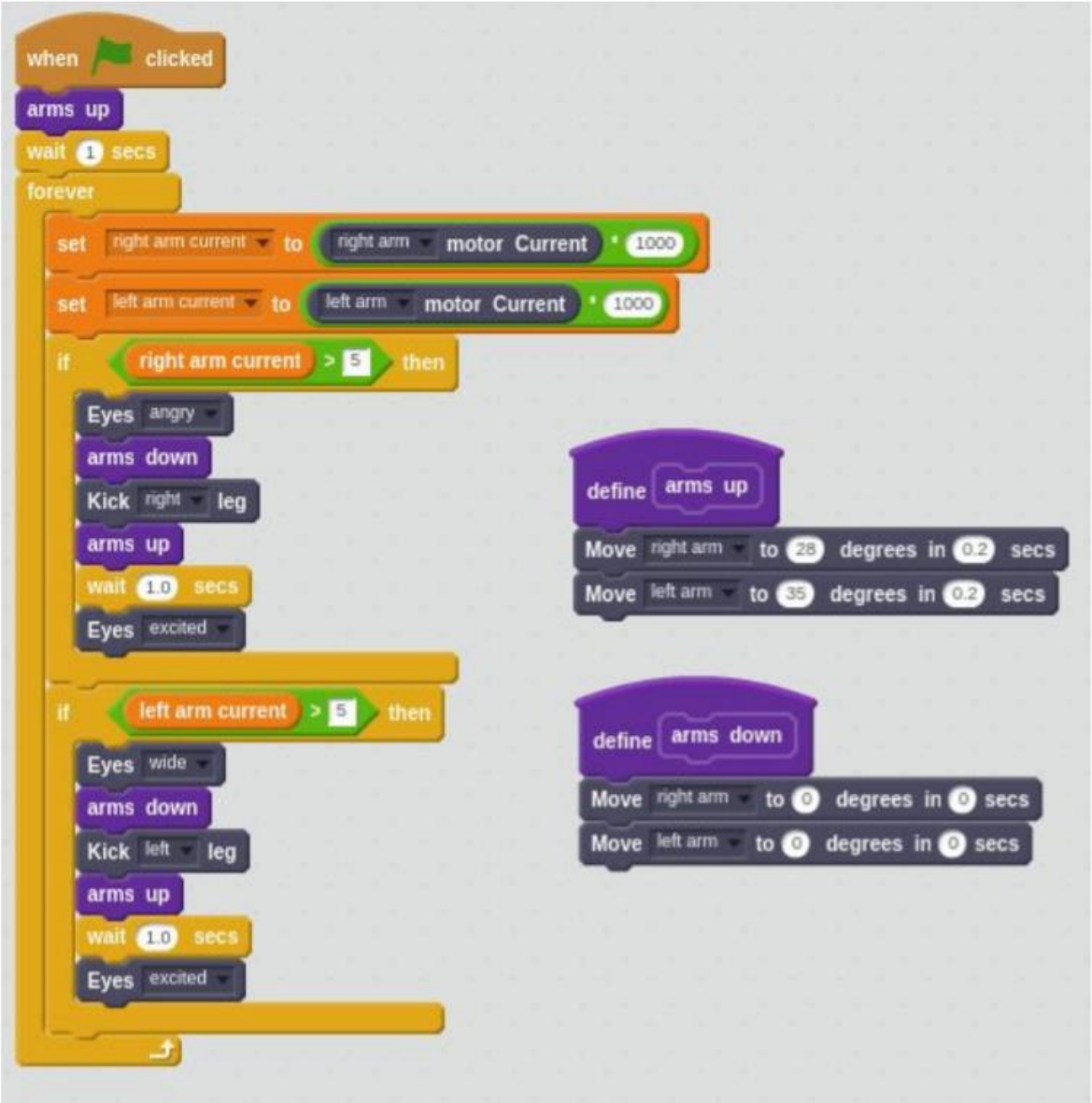
5.Lire le courant d'un moteur (Vidéo)

Avoir un aperçu du courant d'un moteur vous laisse voir à peu près combien de couple un moteur produit (de manière simplifiée, cela permet de voir de combien le moteur "force").

Le capteur renvoie une petite valeur. Pour la rendre exploitable, vous devrez donc la multiplier comme montré ci-dessous.
Dans cet exemple, nous lions le couple du bras droit à la position du sourcil - poussez le bras de Marty ou mettez quelque chose de lourd dessus et il se mettra en colère !

6.Utiliser le courant du moteur comme entrée (Vidéo)

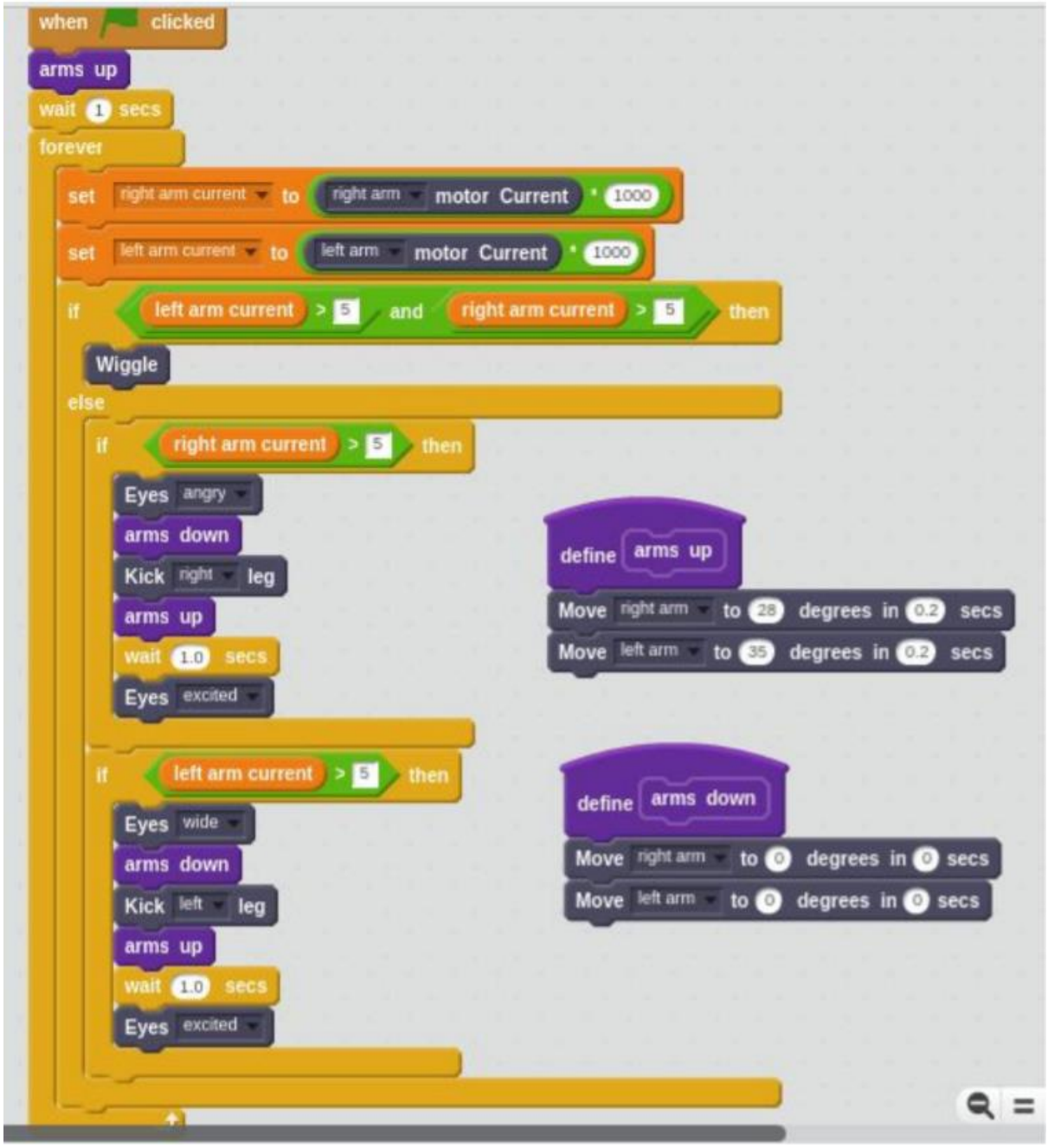
Dans cet exemple, nous donnons des instructions à Marty en tirant l'un de ses bras vers le bas. Tirer le bras droit lui fera donner un coup de pied avec sa jambe droite, et tirer le bras gauche lui fera donner un coup de pied avec sa jambe gauche.


when clicked
arms up
wait 1 secs
forever
set right arm current to right arm motor Current * 1000
set left arm current to left arm motor Current * 1000
if right arm current > 5 then
Eyes angry
arms down
Kick right leg
arms up
wait 1.0 secs
Eyes excited
if left arm current > 5 then
Eyes wide
arms down
Kick left leg
arms up
wait 1.0 secs
Eyes excited
define arms up
Move right arm to 28 degrees in 0.2 secs
Move left arm to 35 degrees in 0.2 secs
define arms down
Move right arm to 0 degrees in 0 secs
Move left arm to 0 degrees in 0 secs

V. Fonctionnalités supplémentaires

1.Comportement AND (Vidéo)

La cerise sur le gâteau : vous pouvez dire à Marty de faire quelques chose de spécial lorsque *les deux* bras sont tirés en même temps.



2.Afficher des valeurs dans des graphiques (Vidéo)

Une autre fonctionnalité très pratique de Scratch est d'afficher des lectures de capteurs sous forme de graphique.

Dans cet exemple, nous affichons la valeur de l'un de capteur de courant du moteur.
Pour faire ceci, nous créons un lutin sous la forme d'un simple cercle, puis nous l'animons avec un stylo.