



## Le Petit Gaillard (mBlock)

### Objectifs :

- Préparer le Petit Gaillard pour son utilisation
- Programmer le Petit Gaillard pour faire différentes postures et mouvements

### Matériel :

- Morceaux de Petit Gaillard (1 corps, 1 support, 2 bras, 2 avant-bras)
- 4 micro-servomoteurs positionnels+ attaches
- Fils
- Carte Arduino Uno +fil USB
- Fil d'alimentation 12 V ou bloc-piles + piles
- Ordinateur
- Platine d'essai

### Déroulement :

- Monter le robot sur sa base
- Positionner les moteurs et attacher les bras et les avant-bras sur le robot
- Programmer le robot pour différents mouvements de danse

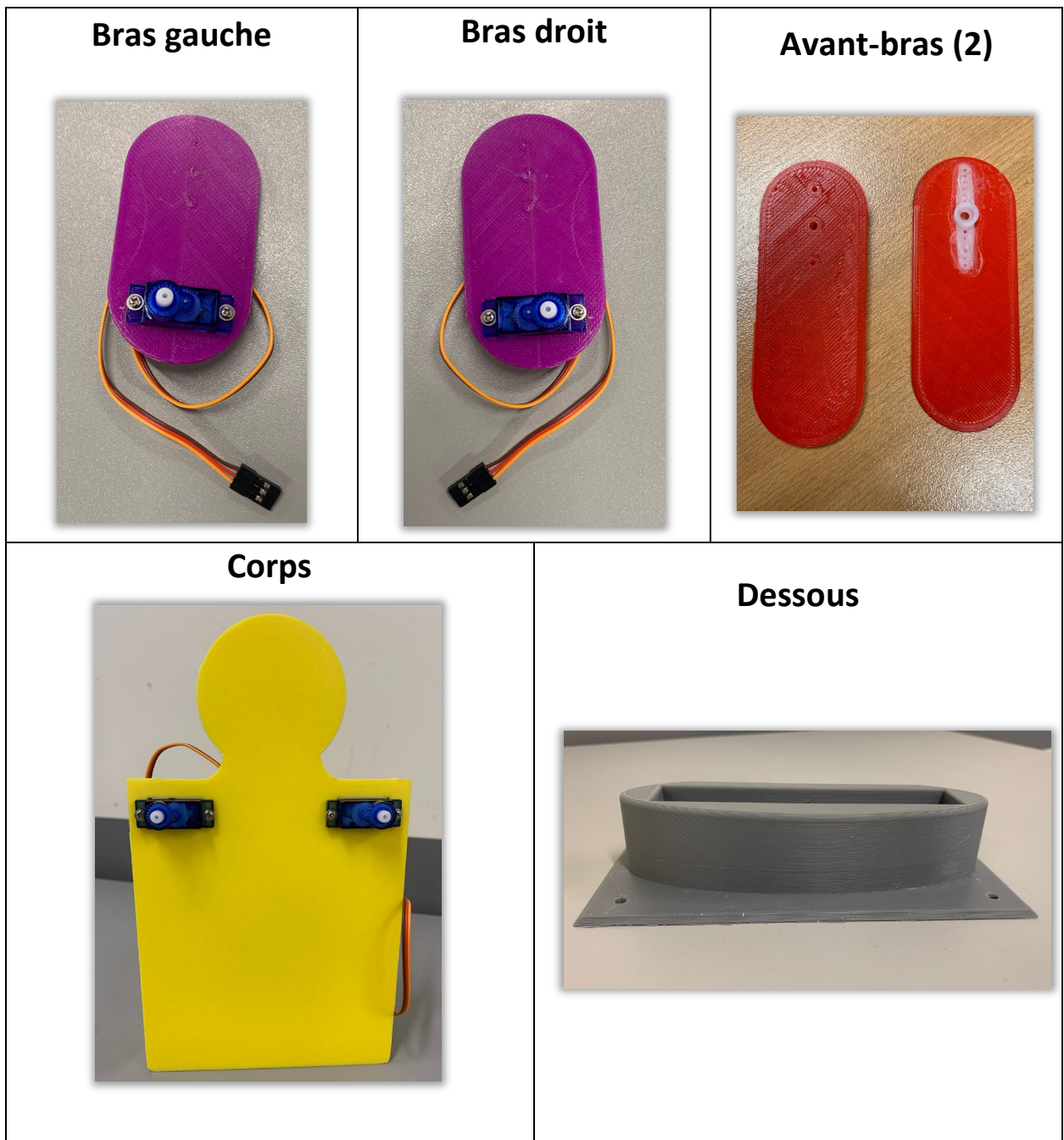
### Le Petit Gaillard



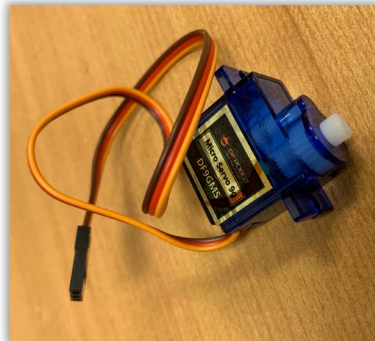
Le principe de base du Petit Gaillard est assez simple. Quatre servomoteurs servent à faire bouger les bras. Chaque servomoteur permet une rotation de son arbre de 180 degrés. Il y a un moteur pour chaque coude et un moteur pour chaque épaule.

En programmant la position des servomoteurs, on peut faire bouger les bras du robot.

**Parties du robot :**



### Servomoteurs (4 déjà installés)



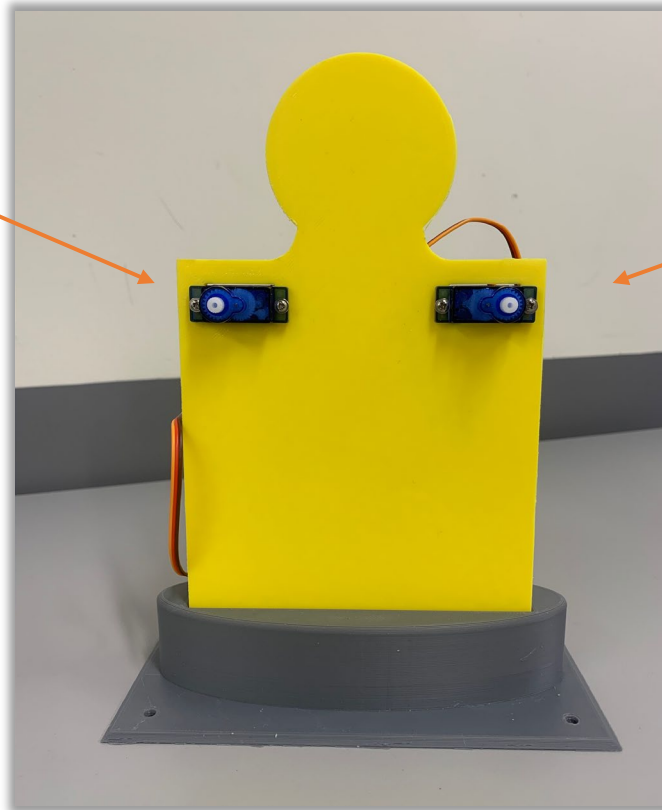
### Attaches supplémentaires (2)



### Préparation

1. La première étape consiste à placer le corps et dans le support pour que le robot se tienne debout. Les deux servomoteurs des épaules sont déjà en place sur le corps. On obtient ceci :

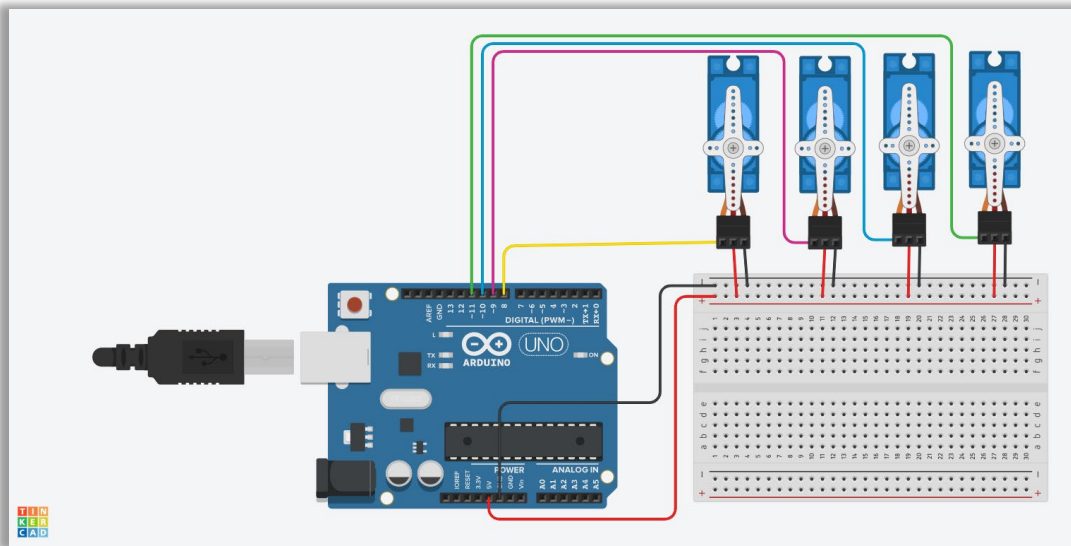
Gauche



Droite

**Dans le présent document, la gauche du robot est votre gauche et non la sienne.**

2. Brancher les servomoteurs comme suit :



Le tableau suivant vous indique quel moteur brancher dans quelle broche digitale de la carte Arduino :

Servomoteur	Broche
Épaule Gauche	8
Épaule Droite	9
Coude Gauche	10
Coude Droit	11

Pour brancher les servomoteurs, il faut noter que le fil rouge doit aller sur la broche +5V, le fil noir ou brun sur la broche GND et le fil orange ou jaune sur la broche digitale qui contrôle le moteur en question.

Petit truc :

Bien s'assurer que les fils passent derrière le robot et que la carte Arduino aussi est derrière le robot. Cela évitera que les fils n'entravent le mouvement des bras.

3. Dans Arduino, écrire le code suivant :

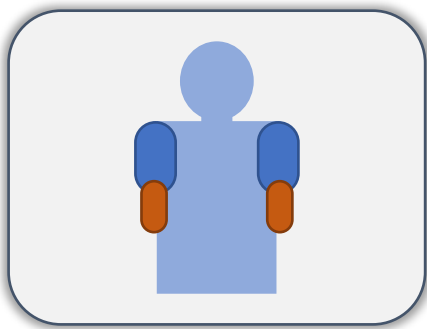
```
#include <Servo.h>

Servo epauleGauche;
Servo epauleDroite;
Servo coudeGauche;
Servo coudeDroit;

void setup(){
    epauleGauche.attach(8);
    epauleDroite.attach(9);
    coudeGauche.attach(10);
    coudeDroit.attach(11);
}

void loop(){
    epauleGauche.write(180);
    epauleDroite.write(0);
    coudeGauche.write(180);
    coudeDroit.write(0);
}
```

4. Téléverser ensuite le code dans la carte Arduino.
5. Une fois que les moteurs sont immobiles (ça devrait être assez rapide), placer d'abord les bras et ensuite les avant-bras pour obtenir la posture suivante :



6. À la suite du code précédent, ajouter les blocs nécessaires pour obtenir le code suivant et le téléverser dans la carte.
- 7.

```

#include <Servo.h>

Servo epauleGauche;
Servo epauleDroite;
Servo coudeGauche;
Servo coudeDroit;

void setup(){
    epauleGauche.attach(8);
    epauleDroite.attach(9);
    coudeGauche.attach(10);
    coudeDroit.attach(11);
}

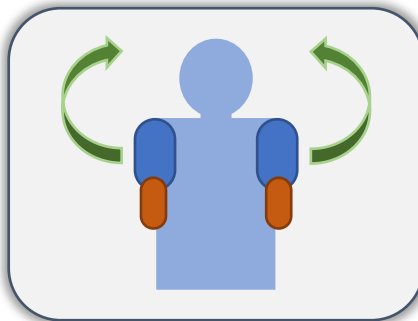
void loop(){
    epauleGauche.write(180);
    epauleDroite.write(0);
    coudeGauche.write(180);
    coudeDroit.write(0);

    delay(1000);

    epauleGauche.write(0);
    epauleDroite.write(180);
    coudeGauche.write(180);
    coudeDroit.write(0);
}

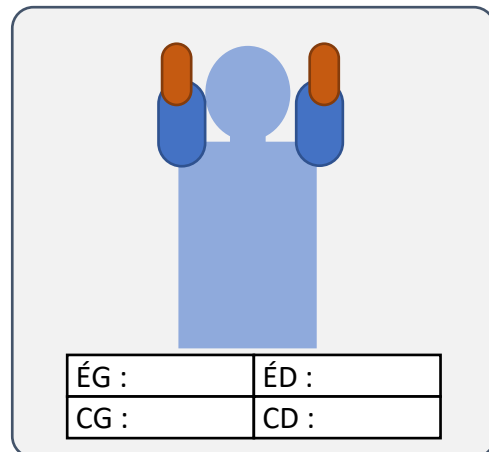
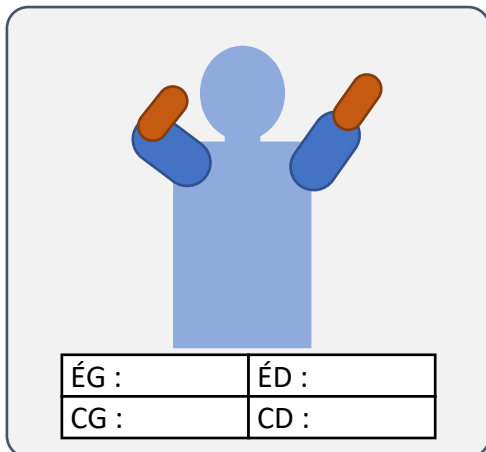
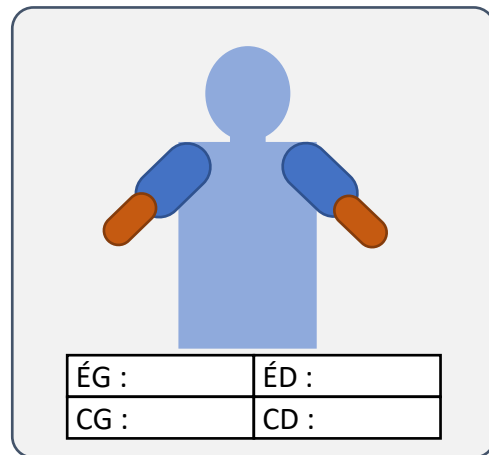
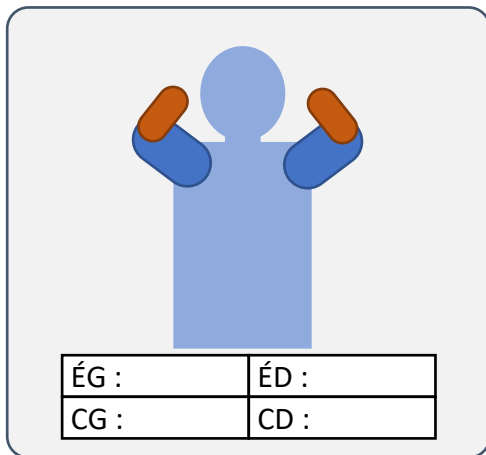
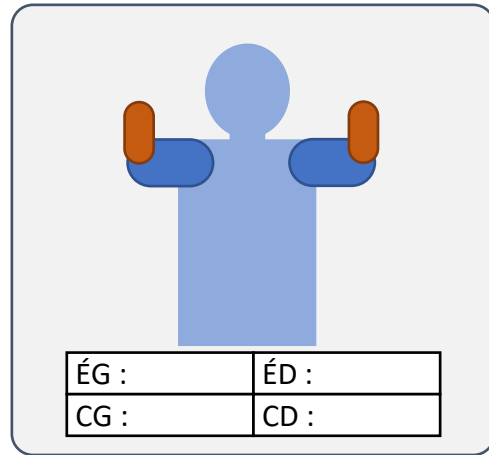
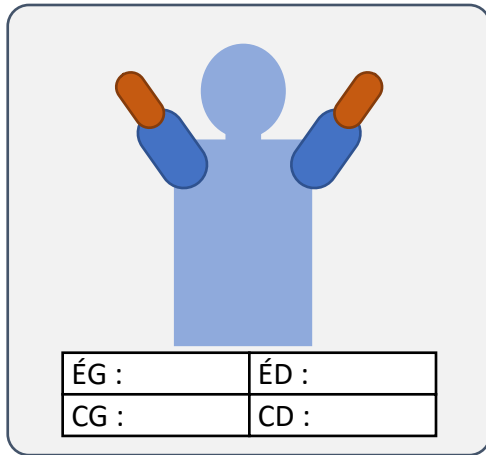
```

Les deux bras devraient passer de la position complètement en bas à la position complètement en haut, en tournant vers l'extérieur, comme suit :





Sinon, vérifier les branchements, enlever les bras et les avant-bras du robot et recommencer à l'étape 3.

8. À partir du programme précédent, faire un programme pour reproduire chacune des postures suivantes du robot. Noter aussi les valeurs des angles qui correspondent :



9. Inventez vos propres postures et inscrire les angles qui correspondent :

	
ÉG :	ÉD :
CG :	CD :

	
ÉG :	ÉD :
CG :	CD :

10. En modifiant le code utilisé pour l'App-IA avec les 3 DEL, programmer le robot pour 3 postures différentes au choix lorsque l'App-IA reconnaîtra les couleurs Bleu, Rouge ou Vert.

**Défi ultime :**

Faire un autre modèle d'IA permettant de reconnaître 4 classes d'images ou plus et l'entraîner. Programmer le robot pour qu'il ait une posture différente pour chacune de ces postures.