 La petite coccinelle qui a peur

Ce projet vous permettra d'utiliser l'information provenant d'une cellule photo-résistive fixée à un servomoteur afin de créer une « petite coccinelle » simple qui a peur lorsqu'on s'en approche. La résistance électrique d'une cellule photo-résistive varie selon la lumière qu'elle reçoit. Judicieusement placée sous une lampe, cette petite « bête » pourra ainsi « détecter » lorsqu'une main s'en approche, puisque cette dernière fera de l'ombre sur le capteur, la cellule. Une commande sera ensuite envoyée au servomoteur afin de faire reculer la cellule qui y est attachée.

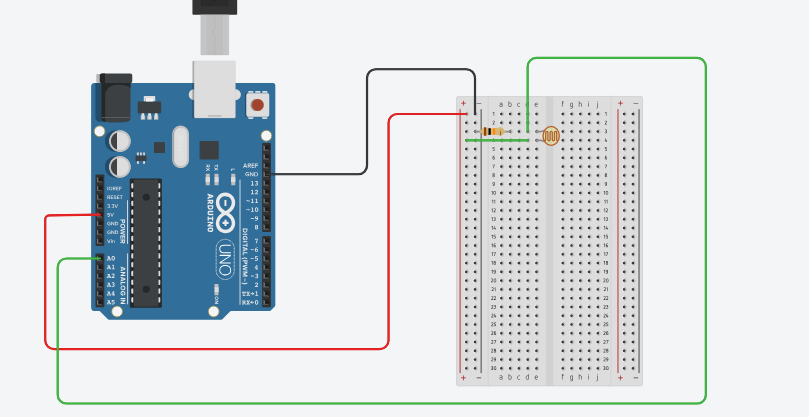
*Éléments nécessaires pour réaliser ce projet*

|  |  |
| --- | --- |
| * une cellule photo-résistive ; * une résistance de 10 kΩ ; * un servomoteur. | * des pinces crocodiles ; * quelques fils de connexion. |

1. Acquisition de l’information

Dans cette première étape, nous travaillerons sur la partie « Acquérir l’information » de ce projet. Nous connecterons la cellule photosensible à l’entrée analogique A0 de la carte Arduino et afficherons l'information qui y est associée.

Montage



Effectuer le montage ci-contre



Programme d’acquisition

#include <Servo.h> // inclure la librairie pour l'utilisation des servomoteurs

Servo servo1; // créer l'objet "servo1" pour référer à notre servomoteur

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// définition des variables

int pinCapteur = A0; // variable pour désigner quelle entrée de l'Arduino est utilisée pour le capteur

int valeurCapteur = 0; // variable pour contenir la valeur lue sur le capteur.

void setup()



{

Serial.begin(9600); // établir la connexion série à 9600 baud

}

void loop()

{

valeurCapteur = analogRead(pinCapteur); // lire la pin analogique A0 et mettre la valeur dans valeurCapteur

Serial.print(« valeurCapteur = »); // communiquer au moniteur série la valeur du capteur.

Serial.println(valeurCapteur); // communiquer au moniteur série la valeur du capteur.

delay(200); // laisser un cours délai pour éviter le trop-plein d'informations

}

Copier le programme en langage Arduino ci-dessus dans un fichier vierge sur le logiciel Arduino.

Transférer le programme sur la carte Arduino

puis ouvrir le moniteur (retour d’information à l’écran)

Eclairer la coccinelle puis relever les valeurs renvoyées sur le moniteur avec et sans l’ombre faite par la main.



Coccinelle éclairée 🡪 ValeurCapteur = ?????

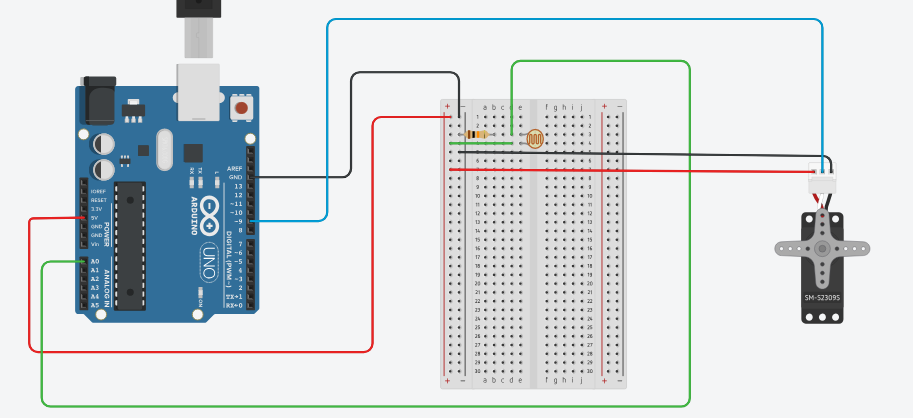
Coccinelle à l’ombre 🡪 ValeurCapteur = ?????

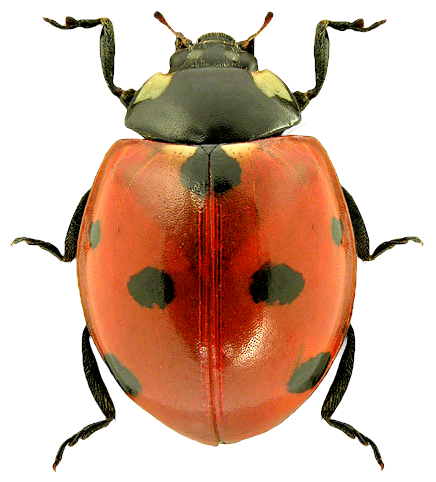
1. Fuite de la coccinelle – pilotage de la chaîne d’énergie



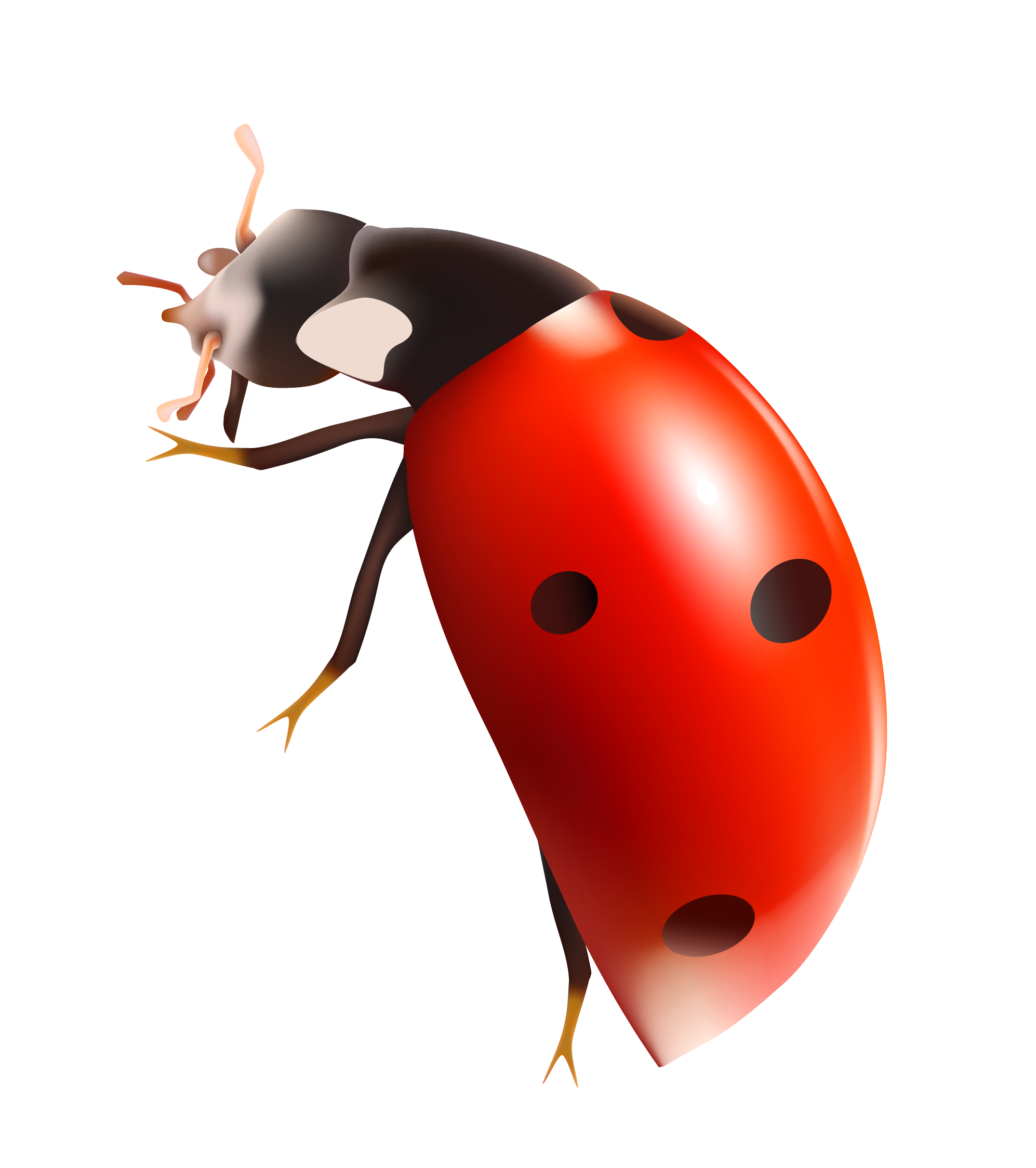
Dans cette deuxième étape, nous travaillerons sur la partie « Chaîne d’énergie » et son pilotage. La conversion d’énergie électrique en mécanique de rotation qui assure le déplacement de la coccinelle est réalisée par un servomoteur. Un servomoteur est moteur dont la position angulaire de l’arbre est asservie à une consigne donnée sous forme de signal rectangulaire MLI (équivalant au signal de sortie analogique de la carte Arduino) . Nous connecterons le servomoteur à la carte Arduino et une valeur de position aléatoire lui sera envoyée afin de faire fuir la coccinelle.

Montage





Effectuer le montage ci-dessus. La photorésistance sera placée sur la coccinelle.



Programme d’acquisition

#include <Servo.h> // inclure la librairie pour l'utilisation des servomoteurs

Servo servo1; // créer l'objet "servo1" pour référer à notre servomoteur

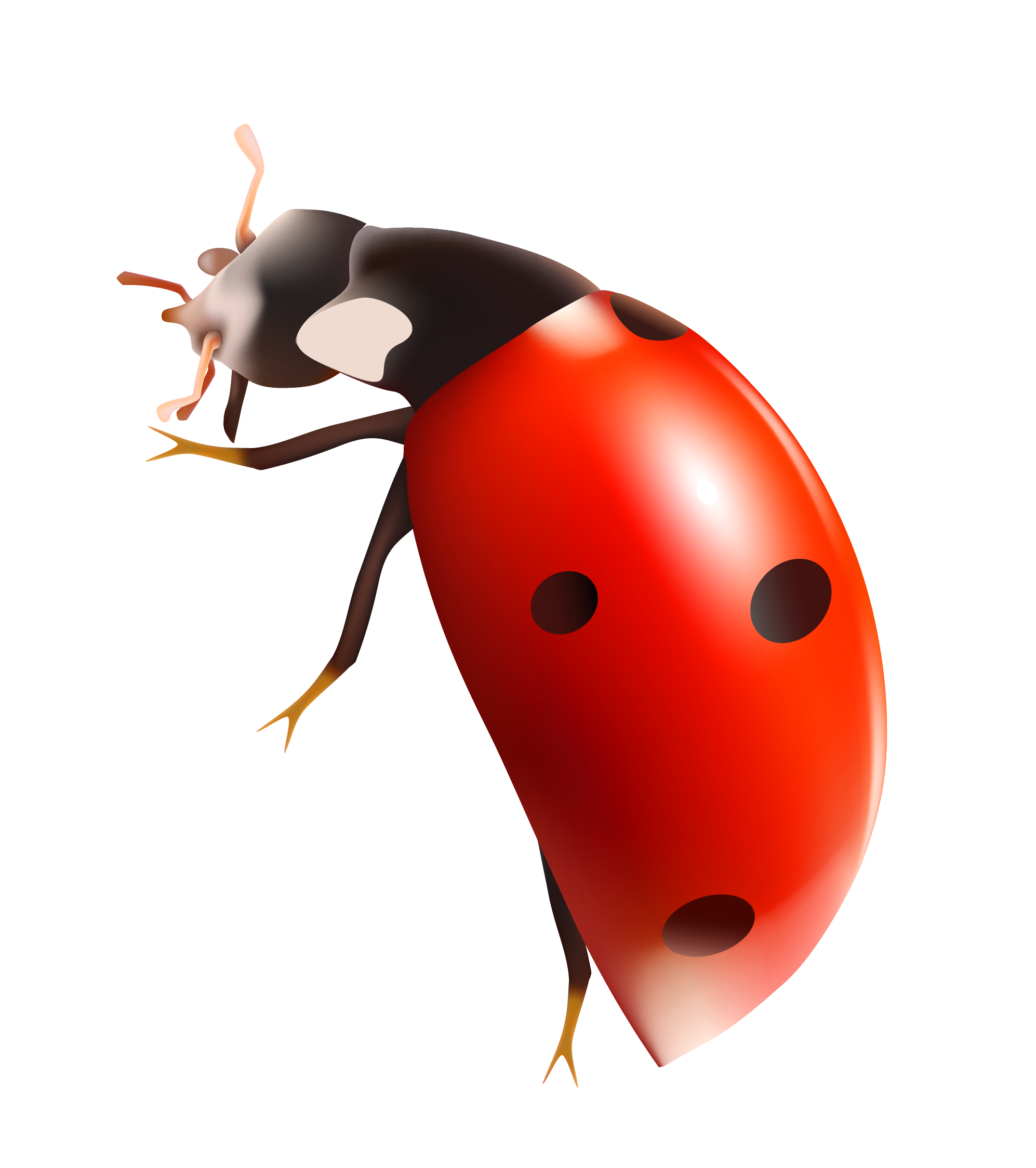
// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// définition des variables

int pinCapteur = A0; // variable pour désigner quelle pin de l'Arduino est utilisée pour le capteur

int valeurCapteur = 0; // variable pour contenir la valeur du capteur.

int servoPos = 0; // variable pour contenir et spécifier la position en degrés (0-180)



void setup()

{

Serial.begin(9600); // établir la connexion série à 9600 baud

servo1.attach(9); // associer la PIN 9 au servomoteur

}

void loop()

{

valeurCapteur = analogRead(pinCapteur); // lire la pin analogique A0 et mettre la valeur dans valeurCapteur

Serial.print(« valeurCapteur = »); // communiquer au moniteur série la valeur du capteur.

Serial.println(valeurCapteur); // communiquer au moniteur série la valeur du capteur.

if(valeurCapteur< ?????????)

{

servoPos = random(181); // attribuer une valeur aléatoire comprise entre 0 et 180 à servoPos

servo1.write(servoPos); // spécifier la position au servomoteur

}

delay(200); // laisser un cours délai pour éviter le trop-plein d'informations

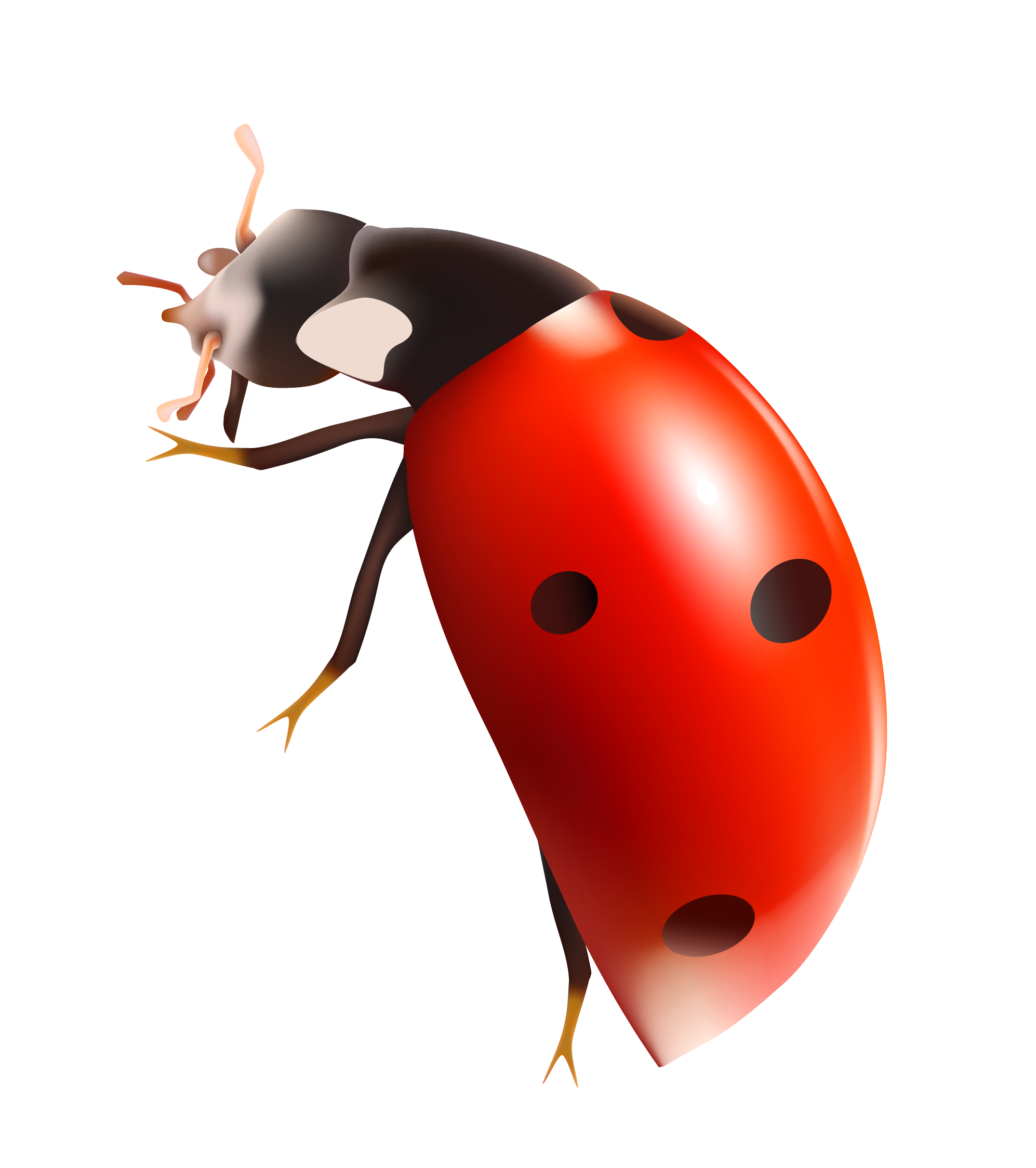
}

Rajouter au programme les lignes en rose vous changerez ici if(valeurCapteur< ?????????) les ?????? par une valeur de lecture du capteur qui doit déclencher la fuite de la coccinelle.

Transférer le programme sur la carte Arduino

puis ouvrir le moniteur (retour d’information à l’écran)

Effectuer les essais. Existe-t-il des écarts entre le fonctionnement souhaité et le fonctionnement obtenu ?



La réalisation de votre projet est-elle validée ?

Vont-ils enfin me laisser tranquille ?!!



