

```
1  /**
2   * Code de barrière de stationnement
3   * @author Matias Vazquez-Levi
4   */
5  #include <Servo.h>
6
7  #define PIN_SERVO_MOTOR 9
8  #define PIN_SENSOR_IR 2
9  #define PIN_LED_GREEN 8
10 #define PIN_LED_RED 6
11
12 Servo motor;
13 int angle_servo = 0;
14
15 enum State {
16     WAITING,
17     OPENING,
18     CLOSING
19 };
20
21 State etat = WAITING;
22
23 void setup()
24 {
25     // Configurer le moteur
26     motor.attach(PIN_SERVO_MOTOR);
27
28     // Configurer le capteur IR
29     pinMode(PIN_SENSOR_IR, INPUT);
30
31     // Configurer les broches de DEL
32     pinMode(PIN_LED_GREEN, OUTPUT);
33     pinMode(PIN_LED_RED, OUTPUT);
34
35     // Allumer la LED rouge
36     digitalWrite(PIN_LED_RED, HIGH);
37     digitalWrite(PIN_LED_GREEN, LOW);
38
39     delay(10); // Attendre que le moteur soit bien configuré
40     motor.write(0);
41     delay(300); // Attendre que le moteur soit bien en position avant de débiter le loop
42 }
43
44 void waiting()
45 {
46     // Lire la valeur du capteur IR
47     int value = digitalRead(PIN_SENSOR_IR);
48
49     // Si le capteur est à 0, c'est qu'un vehicule est détectée
50     if (value == 0) {
51         // Changer d'état
52         etat = OPENING;
53     } else {
54         // Allumer la LED rouge
55         digitalWrite(PIN_LED_RED, HIGH);
56     }
57 }
58
59 void opening()
60 {
61     // Ouvrir la barrière 3 degrés à la fois
62     angle_servo += 3;
63
64     // Si la barrière est complètement ouverte
65     if (angle_servo > 90) {
66
67         // Allumer la LED verte et éteindre la LED rouge
68         digitalWrite(PIN_LED_GREEN, HIGH);
69         digitalWrite(PIN_LED_RED, LOW);
```

```
70 // Attendre que la voiture passe
71 while(digitalRead(PIN_SENSOR_IR) == 0); // Tant que le capteur n'est pas à 1, attendre
72
73
74 // Attendre un peu plus en clignotant la LED rouge
75 digitalWrite(PIN_LED_GREEN, LOW);
76 delay(300);
77 digitalWrite(PIN_LED_RED, HIGH);
78 delay(300);
79 digitalWrite(PIN_LED_RED, LOW);
80 delay(300);
81 digitalWrite(PIN_LED_RED, HIGH);
82 delay(300);
83 digitalWrite(PIN_LED_RED, LOW);
84 delay(300);
85
86 // Changement d'état
87 etat = CLOSING;
88 }
89 // Mettre à jour l'angle du moteur
90 motor.write(angle_servo);
91
92 // Attendre un peu
93 delay(80);
94 }
95
96 void closing()
97 {
98 // Fermer la barrière 3 degrés à la fois
99 angle_servo -= 3;
100
101 // Si la barrière est complètement fermée
102 if (angle_servo < 0) {
103
104 // Changement d'état
105 etat = WAITING;
106
107 // S'assurer que l'angle est bien à 0
108 angle_servo = 0;
109 }
110
111 // Mettre à jour l'angle du moteur
112 motor.write(angle_servo);
113
114 // Attendre un peu
115 delay(80);
116 }
117
118 void loop()
119 {
120 // Chaque état a son propre "loop"
121 switch(etat) {
122 case WAITING: waiting(); break;
123 case OPENING: opening(); break;
124 case CLOSING: closing(); break;
125 }
126 }
```