

# Document mis en page à rendre pour le projet

Ce document doit contenir les éléments suivants :

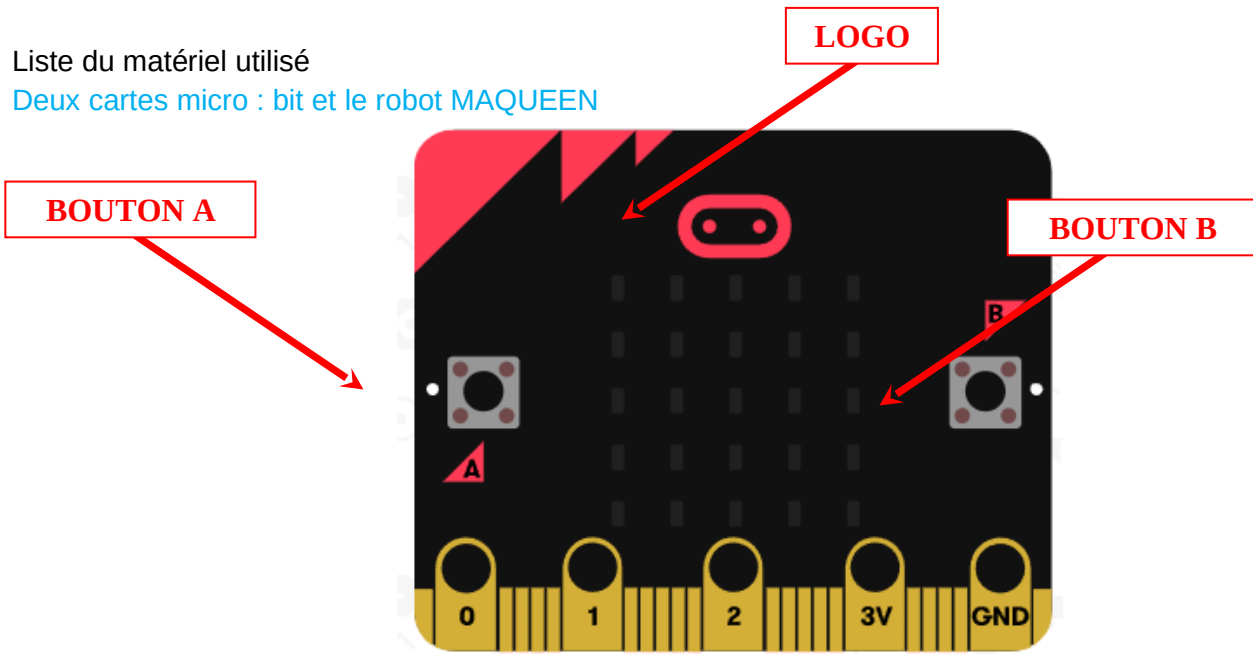
Noms et prénoms des élèves du groupe  
Ilyes El Yaakoubi et Axel Roger-Bourdeix.

Titre du projet  
Manette pour pouvoir télécommander voiture.

Ce que nous voulons faire :

Pouvoir commander la voiture MAQUEEN uniquement avec la carte micro : bit à distance à l'aide de signaux radios.

Liste du matériel utilisé  
Deux cartes micro : bit et le robot MAQUEEN



Les capteurs et actionneurs utilisés

Description fonctionnelle :

- BOUTONS
  - o Le bouton A :  
Lorsqu'on appuie sur le bouton A la voiture tourne à gauche
  - o Le bouton B :  
Lorsqu'on appuie sur le bouton A la voiture tourne à droite
  - o Le bouton A + B :  
Lorsqu'on appuie sur les deux boutons A et B en même temps la voiture avance
- L'inclinaison de la carte micro : bit :
  - o Le logo vers le haut :  
La voiture accélère.
  - o Le logo vers le bas :  
La voiture ralenti.

CARTE « ROBOT MAQUEEN »

```
au démarrage
radio définir groupe 1
afficher texte "vroom"
définir vitesse à 100
```

Au démarrage de la carte ou après appuie sur le bouton reset

On choisit le groupe radio pour la communication entre les deux cartes

FAIT défiler le texte « VROOM »  
On initialise la valeur de la vitesse du robot MAQUEEN

```
quand une donnée est reçue par radio receivedString
si receivedString = "AVANT" alors
  montrer LEDs
  pause (ms) 100
  effacer l'écran
  Moteur All dir horaire vitesse vitesse
si receivedString = "gauche" alors
  montrer LEDs
  pause (ms) 100
  effacer l'écran
  Moteur M1 dir antihoraire vitesse vitesse
  Moteur M2 dir horaire vitesse vitesse
si receivedString = "droite" alors
  montrer LEDs
  pause (ms) 100
  effacer l'écran
  Moteur M1 dir horaire vitesse vitesse
  Moteur M2 dir antihoraire vitesse vitesse
si receivedString = "moins" alors
  définir vitesse à vitesse - 5
  afficher texte "--"
si receivedString = "plus" alors
  définir vitesse à vitesse + 5
  afficher texte "++"
```

← Si le message radio reçu est « AVANT »  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT

ATTENDS pendant 100 millisecondes  
Puis efface l'écran  
On fait tourner les deux moteurs dans le sens horaire à vitesse

← Si le message radio reçu est « gauche »  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT mais positionné à gauche

ATTENDS pendant 100 millisecondes  
Puis efface l'écran  
On fait tourner le moteur droite dans le sens horaire à vitesse  
On fait tourner le moteur gauche dans le sens antihoraire à vitesse

← Si le message radio reçu est « droite »  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT mais positionné à droite

ATTENDS pendant 100 millisecondes  
Puis efface l'écran  
On fait tourner le moteur gauche dans le sens horaire à vitesse  
On fait tourner le moteur droite dans le sens antihoraire à vitesse

← Si le message radio reçu est « moins »  
On diminue la vitesse de 5  
On fait défiler le texte « - - »

← Si le message radio reçu est « plus »  
On augmente la vitesse de 5  
On fait défiler le texte « + + »

```

1 def on_received_string(receivedString):
2     global vitesse
3     if receivedString == "AVANT":
4         basic.show_leds("""
5             . . # . .
6             | | | | . # # # .
7             | | | | . . # . .
8             | | | | . . # . .
9             | | | | . . # . .
10            """)
11        basic.pause(100)
12        basic.clear_screen()
13        maqueen.motor_run(maqueen.Motors.ALL, maqueen.Dir.CW, vitesse)
14    if receivedString == "gauche":
15        basic.show_leds("""
16            . # . . .
17            | | | | # # # . .
18            | | | | . # . . .
19            | | | | . # . . .
20            | | | | . # . . .
21            """)
22        basic.pause(100)
23        basic.clear_screen()
24        maqueen.motor_run(maqueen.Motors.M1, maqueen.Dir.CCW, vitesse)
25        maqueen.motor_run(maqueen.Motors.M2, maqueen.Dir.CW, vitesse)
26    if receivedString == "droite":
27        basic.show_leds("""
27        basic.show_leds("""
28            . . . # .
29            | | | | . . # # #
30            | | | | . . . # .
31            | | | | . . . # .
32            | | | | . . . # .
33            """)
34        basic.pause(100)
35        basic.clear_screen()
36        maqueen.motor_run(maqueen.Motors.M1, maqueen.Dir.CW, vitesse)
37        maqueen.motor_run(maqueen.Motors.M2, maqueen.Dir.CCW, vitesse)
38    if receivedString == "moins":
39        vitesse = vitesse - 5
40        basic.show_string("--")
41    if receivedString == "plus":
42        vitesse = vitesse + 5
43        basic.show_string("++")
44    radio.on_received_string(on_received_string)
45
46    vitesse = 0
47    radio.set_group(1)
48    basic.show_string("vroom")
49    vitesse = 100

```

CARTE « RADIOCOMMANDE »

Au démarrage de la carte ou après appuie sur le bouton reset

On choisit le groupe radio pour la communication entre les deux cartes

FAIT défiler le texte « VROOM »

```
au démarrage
radio définir groupe 1
afficher texte "vroom"
définir vitesse à 100

lorsque logo vers le haut
définir vitesse à vitesse + 5
envoyer la chaîne "plus" par radio
afficher texte "++"

lorsque logo vers le bas
définir vitesse à vitesse - 5
envoyer la chaîne "moins" par radio
afficher texte "--"

toujours
si bouton A + B est pressé alors
envoyer la chaîne "AVANT" par radio
montrer LEDs
pause (ms) 100
effacer l'écran

si bouton A est pressé alors
envoyer la chaîne "gauche" par radio
montrer LEDs
pause (ms) 100
effacer l'écran

si bouton B est pressé alors
envoyer la chaîne "droite" par radio
montrer LEDs
pause (ms) 100
effacer l'écran
```

← Si les boutons A et B sont appuyés alors  
La carte envoie le message « AVANT » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT

ATTENDS pendant 100 millisecondes  
Puis efface l'écran

← Si le boutons A est appuyé alors  
La carte envoie le message « gauche » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT mais positionner vers la gauche

ATTENDS pendant 100 millisecondes  
Puis efface l'écran

← Si le boutons B est appuyé alors  
La carte envoie le message « droite » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT mais positionner vers la droite

ATTENDS pendant 100 millisecondes  
Puis efface l'écran

```

1 def on_gesture_logo_up():
2     global vitesse
3     vitesse = vitesse + 5
4     radio.send_string("plus")
5     basic.show_string("++")
6 input.on_gesture(Gesture.LOGO_UP, on_gesture_logo_up)
7
8 def on_gesture_logo_down():
9     global vitesse
10    vitesse = vitesse - 5
11    radio.send_string("moins")
12    basic.show_string("--")
13 input.on_gesture(Gesture.LOGO_DOWN, on_gesture_logo_down)
14
15 vitesse = 0
16 radio.set_group(1)
17 basic.show_string("vroom")
18 vitesse = 100
19
20 def on_forever():
21     if input.button_is_pressed(Button.AB):
22         radio.send_string("AVANT")
23         basic.show_leds("""
24             . . # . .
25             | | | | . # # # .
26             | | | | . . # . .
27             | | | | . . # . .
28             | | | | . . # . .
29         """)
30         basic.pause(100)
31         basic.clear_screen()
32     if input.button_is_pressed(Button.A):
33         radio.send_string("gauche")
34         basic.show_leds("""
35             . # . . .
36             | | | | # # # . .
37             | | | | . # . . .
38             | | | | . # . . .
39             | | | | . # . . .
40         """)
41         basic.pause(100)
42         basic.clear_screen()
43     if input.button_is_pressed(Button.B):
44         radio.send_string("droite")
45         basic.show_leds("""
46             . . . # .
47             | | | | . . # # #
48             | | | | . . . # .
49             | | | | . . . # .
50             | | | | . . . # .
51         """)
52         basic.pause(100)
53         basic.clear_screen()
54 basic.forever(on_forever)

```

```
[Au démarrage]
[Radio] configurer Canal 7
Puissance 6
Taille des données 32
Groupe 1
afficher "VROOM"
mettre vitesse à 100

Répéter indéfiniment
[Radio] si une chaîne est reçue dans stringData alors
si stringData = AVANT alors
  afficher l'image
  attendre 100 milliseconde(s)
  [Maqueen] contrôler le robot avancer vitesse vitesse

[Radio] si une chaîne est reçue dans stringData alors
si stringData = GAUCHE alors
  afficher l'image
  attendre 100 milliseconde(s)
  [Maqueen] contrôler le moteur droit direction vitesse vitesse
  [Maqueen] contrôler le moteur gauche direction vitesse vitesse

[Radio] si une chaîne est reçue dans stringData alors
si stringData = DROITE alors
  afficher l'image
  attendre 100 milliseconde(s)
  [Maqueen] contrôler le moteur gauche direction vitesse vitesse
  [Maqueen] contrôler le moteur droit direction vitesse vitesse

[Radio] si une chaîne est reçue dans stringData alors
si stringData = MOINS alors
  afficher "-"
  attendre 100 milliseconde(s)
  mettre vitesse à vitesse - 5

[Radio] si une chaîne est reçue dans stringData alors
si stringData = PLUS alors
  afficher "+"
  attendre 100 milliseconde(s)
  mettre vitesse à vitesse + 5
```

```
1 from microbit import *
2 import radio
3 import time
4
5 radio.on()
6
7 radio.config(channel = 7, power = 6, length = 32, group=1)
8 display.show('VROOM')
9 vitesse = 100
10
11 while True:
12     stringData = radio.receive()
13     if stringData:
14         if stringData == 'AVANT':
15             led_image = Image('00900:09990:00900:00900:00900')
16             display.show(led_image)
17             time.sleep_ms(100)
18             i2c.write(0x10, bytearray([0x00, 0x0, vitesse]))
19             i2c.write(0x10, bytearray([0x02, 0x0, vitesse]))
20     stringData = radio.receive()
21     if stringData:
22         if stringData == 'GAUCHE':
23             led_image = Image('09000:99900:09000:09000:09000')
24             display.show(led_image)
25             time.sleep_ms(100)
26             i2c.write(0x10, bytearray([0x02, 0x0, vitesse]))
27             i2c.write(0x10, bytearray([0x00, 0x1, vitesse]))
28     stringData = radio.receive()
29     if stringData:
30         if stringData == 'DROITE':
31             led_image = Image('00090:00999:00090:00090:00090')
32             display.show(led_image)
33             time.sleep_ms(100)
34             i2c.write(0x10, bytearray([0x00, 0x0, vitesse]))
35             i2c.write(0x10, bytearray([0x02, 0x1, vitesse]))
36     stringData = radio.receive()
37     if stringData:
38         if stringData == 'MOINS':
39             display.show('--')
40             time.sleep_ms(100)
41             vitesse = vitesse - 5
42     stringData = radio.receive()
43     if stringData:
44         if stringData == 'PLUS':
45             display.show('++')
46             time.sleep_ms(100)
47             vitesse = vitesse + 5
48
```

CARTE « RADIOCOMMANDE »

Au démarrage de la carte ou après appuie sur le bouton reset

On choisit le groupe radio pour la communication entre les deux cartes

FAIT défiler le texte « VROOM »  
On initialise la valeur de la vitesse du robot MAQUEEN

- ← Si les boutons A et B sont appuyés alors  
La carte envoie le message « AVANT » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT
  
- ATTENDS pendant 100 millisecondes  
Puis efface l'écran
  
- ← Si les boutons A est appuyé alors  
La carte envoie le message « GAUCHE » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT MAIS POSITIONNER A GAUCHE
  
- ATTENDS pendant 100 millisecondes  
Puis efface l'écran
  
- ← Si les boutons B est appuyé alors  
La carte envoie le message « DROITE » par onde radio  
AFFICHE LE DESSIN D'UNE FLECHE VERS L'AVANT MAIS POSITIONNER A DROITE
  
- ATTENDS pendant 100 millisecondes  
Puis efface l'écran
  
- ← Si la carte micro : bite est incliné avec le logo en haut  
La carte envoie le message « PLUS » par onde radio  
FAIT défiler le texte « ++ »
  
- ← Si la carte micro : bite est incliné avec le logo en bas  
La carte envoie le message « MOINS » par onde radio  
FAIT défiler le texte « -- »

```
1 From microbit import *
2 import radio
3 import time
4
5 radio.on()
6
7 radio.config(channel = 7, power = 6, length = 32, group=1)
8 display.show('VROOM')
9 vitesse = 100
10
11 while True:
12     if button_a.is_pressed() and button_b.is_pressed():
13         radio.send('AVANT')
14         led_image = Image('00900:09990:00900:00900:00900')
15         display.show(led_image)
16         time.sleep_ms(100)
17         display.clear()
18     if button_a.is_pressed():
19         radio.send('GAUCHE')
20         led_image = Image('09000:99900:09000:09000:09000')
21         display.show(led_image)
22         time.sleep_ms(100)
23         display.clear()
24     if button_b.is_pressed():
25         radio.send('DROITEE')
26         led_image = Image('00090:00999:00090:00090:00090')
27         display.show(led_image)
28         time.sleep_ms(100)
29         display.clear()
30     if accelerometer.current_gesture() == 'up' :
31         radio.send('PLUS')
32         display.show('++')
33     if accelerometer.current_gesture() == 'down' :
34         radio.send('MOINS')
35         display.show('--')
36
```

Extensions ou développements possibles du projet

Rajouter des boutons sur la carte radio commande pour créer des fonctions comme réaliser un demi-tour, des slaloms,...

Modifier le programme de la carte radio pour n'utiliser que « son inclinaison »

---