

```
1 'Arduino Robot Car B4R - Pilotage d'un véhicule à distance via un dispositif Bluetooth
  ( Smartphone Android ou Manette de jeux Arduino)
2 'Programme original ARDUINO de Dejan NEDELKOVSKI converti en Visual Basic B4R par Klaus
  CHRISTL et Marc DANIEL
3 #Region Project Attributes
4     #AutoFlushLogs: True
5     #CheckArrayBounds: True
6     #StackBufferSize: 300
7 #End Region
8
9 Sub Process_Globals
10     Public Serial1 As Serial
11     Private SoftwareSerial1 As SoftwareSerial
12     Private astream As AsyncStreams
13     Private IN1, IN2, IN3, IN4, ENA, ENB As Pin
14     Private xAxis, yAxis, SpeedD, SpeedG As UInt
15     Private x=0, y=0 As UInt
16     Private xMapped As UInt
17 End Sub
18
19 Private Sub AppStart
20     Serial1.Initialize(115200)
21     Log("App started")
22     IN1.Initialize(4, IN1.MODE_OUTPUT)
23     IN2.Initialize(5, IN2.MODE_OUTPUT)
24     IN3.Initialize(6, IN3.MODE_OUTPUT)
25     IN4.Initialize(7, IN4.MODE_OUTPUT)
26     ENA.Initialize(9, ENA.MODE_OUTPUT)
27     ENB.Initialize(10, ENB.MODE_OUTPUT)
28
29     SoftwareSerial1.Initialize(9600, 11, 12) 'Software Serial port sur les broches 11 et
12
30     astream.Initialize(SoftwareSerial1.Stream, "astream_NewData", Null)
31     SpeedD=0
32     SpeedG=0
33     xAxis=0
34     yAxis=0
35 End Sub
36
37 Sub AStream_NewData (Buffer() As Byte)
38     'Valeurs par défaut s'il n'y a aucune donnée en provenance du dispositif Bluetooth
  (Soit Joystick ARDUINO soit application Android sur Smartphone)
39
40     x=510/4
41     y=510/4
42     If Buffer.Length = 2 Then
43         Log(Buffer.Length)
44         Log(Buffer(0), " / ", Buffer(1))
45         x= Buffer(0)
46         Delay(10)
47         y=Buffer(1)
48         xAxis=x*4
49         yAxis=y*4
50         Log("yAxis ", yAxis, "  xAxis ", xAxis)
51         If yAxis < 470 Then
52             'lance le moteur droit en avant
53             IN1.DigitalWrite(True)
```

```
54     IN2.DigitalWrite(False)
55     'lance le moteur gauche en avant
56     IN3.DigitalWrite(True)
57     IN4.DigitalWrite(False)
58     SpeedD=MapRange(yAxis, 470, 0, 0, 255)
59     SpeedG=MapRange(yAxis, 470, 0, 0, 255)
60     Log("en avant")
61 else if yAxis > 550 Then
62     'lance le moteur droit en arrière
63     IN1.DigitalWrite(False)
64     IN2.DigitalWrite(True)
65     'lance le moteur gauche en arrière
66     IN3.DigitalWrite(False)
67     IN4.DigitalWrite(True)
68     SpeedD=MapRange(yAxis, 550, 1023, 0, 255)
69     SpeedG=MapRange(yAxis, 550, 1023, 0, 255)
70     Log("en arrière")
71 Else
72     SpeedD=0
73     SpeedG=0
74     Log("arrêt")
75 End If
76 If xAxis < 470 Then
77     xMapped=MapRange(xAxis, 470, 0, 0, 255)
78     'Déplacement vers la gauche = diminution de la vitesse du moteur gauche +
augmentation de la vitesse du moteur droit
79     SpeedD = SpeedD - xMapped
80     SpeedG = SpeedG + xMapped
81     If SpeedD < 0 Then
82         SpeedD=0
83     End If
84     If SpeedG > 255 Then
85         SpeedG=255
86     End If
87 End If
88 If xAxis > 550 Then
89     xMapped = MapRange(xAxis, 550, 1023, 0, 255)
90     'Déplacement vers la droite = diminution de la vitesse du moteur droit,
augmentation de la vitesse du moteur gauche
91     SpeedD = SpeedD + xMapped
92     SpeedG = SpeedG - xMapped
93     If SpeedD > 255 Then
94         SpeedD = 255
95     End If
96     If SpeedG < 0 Then
97         SpeedG = 0
98     End If
99 End If
100 If SpeedD < 70 Then
101     'Suppression des bourdonnements à basse vitesse (La valeur 70 peut être
modifiée en fonction des moteurs utilisés)
102     SpeedD = 0
103 End If
104 If SpeedG < 70 Then
105     SpeedG=0
106 End If
107 End If
```

```
108     Log("SpeedG ", SpeedG)
109     ENA.analogWrite(SpeedD)
110     ENB.analogWrite(SpeedG)
111 End Sub
112
113
114
115
```