

```
1 'Voiture ARDUINO - ARDUINO CAR - Utiliser Arduino UNO en association avec une carte shield ↗
   - Marc DANIEL - Novembre 2021
2 'Mise à jour de Novembre 2025 - Update November 2025
3 'YFROBOT PM-R3 Smart Car Drive Board R3 Expansion Board Multi-Function Motor Drive Board
4 #Region Project Attributes
5     #AutoFlushLogs: True
6     #CheckArrayBounds: True
7     #StackBufferSize: 300
8 #End Region
9
10
11 Sub Process_Globals
12     Public Serial1 As Serial
13     Private SoftwareSerial1 As SoftwareSerial
14     Private astream As AsyncStreams
15     Private BtnTest, DIR1, DIR2, PWMA, Lights, Klaxon As Pin
16     Private Speed As UInt
17     Private FEUX As Boolean
18     Public MonServo As Servo
19     Private SON As Boolean
20     Public Angle, L, X=0 As UInt
21 End Sub
22
23 Private Sub AppStart
24     Serial1.Initialize(115200)
25     Log("Démarrage")
26     DIR1.Initialize(4, DIR1.MODE_OUTPUT) ' Marche avant - Forward
27     DIR2.Initialize(7, DIR2.MODE_OUTPUT) ' Marche arrière - Backward
28     PWMA.Initialize(5, PWMA.MODE_OUTPUT) ' moteur arrière
29     MonServo.Attach2(3, 30, 130)         ' Direction by servo motor
30     straight_Ahead
31     Lights.Initialize(13, Lights.MODE_OUTPUT)
32     Klaxon.Initialize(2, Klaxon.MODE_OUTPUT)
33     BtnTest.Initialize(BtnTest.A0, BtnTest.MODE_INPUT_PULLUP)
34     BtnTest.AddListener("BtnTEST_StateChanged")
35     SoftwareSerial1.Initialize(9600, 11, 12)
36     astream.Initialize(SoftwareSerial1.Stream, "astream_NewData", Null)
37 End Sub
38
39 Private Sub BtnTEST_StateChanged(State As Boolean)
40
41     If State = False Then 'BoutonTEST=0
42         Log("Bouton test: ", State)
43         Depart
44     Else 'BoutonTEST=1
45         Log("Bouton test: ", State)
46     End If
47 End Sub
48
49 Sub Depart 'tests de fonctionnement des équipements après appui sur le petit bouton test
50     For x=1 To 3
51         Lights.DigitalWrite(True) 'Test des feux si installés
52         Log("Allumage des feux")
53         FEUX = True
54         Klaxon.DigitalWrite(True) 'Test du Klaxon ou Buzzer si installé
55         SON = True
56         Log("Déclenchement du Klaxon")
```

```
57     Delay(1000)
58     Lights.DigitalWrite(False)
59     FEUX = False '(Lumières éteintes par défaut)
60     Log("Extinction des feux")
61     Klaxon.DigitalWrite(False)
62     SON = False
63     Log("Arrêt du Klaxon") '(Klaxon à l'arrêt par défaut)
64     Delay(1000)
65 Next
66 'Tests sur la direction commandée par le Servomoteur
67 straight_Ahead
68 Log("Roues avant braquées à droite")
69 Delay(2000)
70 For i = 90 To 35 Step -1
71     MonServo.Write(i)
72     Delay(10)
73 Next
74 Log("Roues avant braquées à gauche")
75 Delay(2000)
76 For i = 35 To 90
77     MonServo.Write(i)
78     Delay(10)
79 Next 'Les roues avant sont alignées en ligne droite
80 Log("Alignement des roues avant")
81 DIR1.DigitalWrite(False)
82 DIR2.DigitalWrite(False)
83 Speed = 150
84 Envoi
85 Delay(3000)
86 DIR1.DigitalWrite(True)
87 DIR2.DigitalWrite(True)
88 Speed = 100
89 Envoi
90 Delay(3000)
91 Speed = 0 'Vitesse ramenée à zéro par défaut
92 Envoi
93 For x=1 To 3
94     Lights.DigitalWrite(True) 'Test des feux si installés
95     Log("Allumage des feux")
96     FEUX = True
97     Klaxon.DigitalWrite(True) 'Test du Klaxon ou Buzzer si installé
98     SON = True
99     Log("Déclenchement du Klaxon")
100    Delay(1000)
101    Lights.DigitalWrite(False)
102    FEUX = False '(Lumières éteintes par défaut)
103    Log("Extinction des feux")
104    Klaxon.DigitalWrite(False)
105    SON = False
106    Log("Arrêt du Klaxon") '(Klaxon à l'arrêt par défaut)
107    Delay(1000)
108 Next
109 End Sub 'Fin des tests de fonctionnement des équipements
110
111 Sub AStream_NewData (Buffer() As Byte)
112     'Fonctionnement de la voiture pilotée par Bluetooth
113     L= Buffer.Length
```

```
114     If L = 1 Then X = Buffer(0)
115     If L = 2 Then X = Buffer(0)
116     If L = 5 Then X = Buffer(4)
117     Log("Buffer: ", L, " bytes")
118     Log("Value of X: ", X)
119
120     Select X
121
122         Case 0 'Arrêt de la voiture
123             Speed=0
124             Envoi
125
126         Case 2 'Alignement des roues avant en ligne droite
127
128             straight_Ahead
129
130         Case 3 ' GESTION du «Klaxon» ou buzzer si connecté à la broche 3
131             If SON = False Then
132                 Klaxon.DigitalWrite(True)
133                 SON = True
134                 Log("Déclenchement du Klaxon")
135             Else
136                 Klaxon.DigitalWrite(False)
137                 SON = False
138                 Log("Arrêt du Klaxon")
139             End If
140
141         Case 4 'Rotation légère à gauche
142
143             Angle = MonServo.Read
144             If Angle < 70 Then
145                 For i = Angle To 70
146                     MonServo.Write(i)
147                     Delay(10)
148                 Next
149             End If
150             If Angle > 70 Then
151                 For i = Angle To 70 Step -1
152                     MonServo.Write(i)
153                     Delay(10)
154                 Next
155             End If
156
157
158         Case 5
159             'Rotation nette à gauche
160             Log("Rotation nette à gauche")
161
162             Angle = MonServo.Read
163             If Angle < 35 Then
164                 For i = Angle To 35
165                     MonServo.Write(i)
166                     Delay(10)
167                 Next
168             End If
169             If Angle > 35 Then
170                 For i = Angle To 35 Step -1
```

```

171         MonServo.Write(i)
172         Delay(10)
173     Next
174 End If
175
176
177 Case 6
178     'Rotation légère à droite
179     Log("Rotation légère à droite")
180     Angle = MonServo.Read
181     If Angle < 110 Then
182         For i = Angle To 110
183             MonServo.Write(i)
184             Delay(10)
185         Next
186     End If
187     If Angle > 110 Then
188         For i = Angle To 110 Step -1
189             MonServo.Write(i)
190             Delay(10)
191         Next
192     End If
193
194
195 Case 7
196     'Rotation nette à droite
197     Log("Rotation nette à droite")
198     Angle = MonServo.Read
199     If Angle < 125 Then
200         For i = Angle To 125
201             MonServo.Write(i)
202             Delay(10)
203         Next
204     End If
205     If Angle > 125 Then
206         For i = Angle To 125 Step -1
207             MonServo.Write(i)
208             Delay(10)
209         Next
210     End If
211
212 Case 8
213     GrandHuit
214
215 Case 9 ' GESTION des FEUX, LEDS, DIODES et lumières diverses si connectés broche ↗
13
216     If FEUX=False Then
217         Lights.DigitalWrite(True)
218         FEUX = True
219
220     Else
221         Lights.DigitalWrite(False)
222         FEUX = False
223         Log("Extinction des LEDs")
224         'Extinction des LEDS si elles existent et ont été connectées sur la ↗
13 broche 13
225     End If

```

```
226
227     Case 10
228         ' MARCHE AVANT - FORWARD MARCH
229         DIR1.DigitalWrite(False)
230         DIR2.DigitalWrite(False)
231         Speed = 100
232         Envoi
233
234     Case 11 ' MARCHE AVANT rapide de la voiture
235         DIR1.DigitalWrite(False)
236         DIR2.DigitalWrite(False)
237         Speed = 180
238         Envoi
239
240
241     Case 12 'MARCHE ARRIÈRE - BACKWARD
242         DIR1.DigitalWrite(True)
243         DIR2.DigitalWrite(True)
244         Speed = 90
245         Envoi
246
247
248
249     Case 13 'MARCHE ARRIÈRE rapide de la voiture - FAST BACKWARD
250         DIR1.DigitalWrite(True)
251         DIR2.DigitalWrite(True)
252         Speed= 110
253         Log("Marche arrière rapide")
254         Envoi
255     End Select
256 End Sub
257
258 Sub GrandHuit
259     ' La voiture va effectuer seule les manoeuvres sur un circuit de plusieurs mètres
260     ' de préférence à l'extérieur
261     straight_Ahead
262     DIR1.DigitalWrite(False)
263     DIR2.DigitalWrite(False)
264     Speed = 140
265     Envoi 'Démarrage
266     Delay(4000)
267     For j=1 To 5
268         Delay(4000)
269         Right_Turn
270         Delay(3000)
271         straight_Ahead
272         Delay(4000)
273     Next
274     Speed = 0
275     For x=1 To 3 '3 appels de phares + 3 coups de Klaxon
276         Lights.DigitalWrite(True)
277         Log("Allumage des feux")
278         FEUX = True
279         Klaxon.DigitalWrite(True)
280         SON = True
281         Log("Déclenchement du Klaxon")
282         Delay(1000)
```

```
283     Lights.DigitalWrite(False)
284     FEUX = False
285     Log("Extinction des feux")
286     Klaxon.DigitalWrite(False)
287     SON = False
288     Log("Arrêt du Klaxon")
289     Delay(1000)
290 Next
291 Envoi 'Fin du circuit
292 End Sub
293
294 Private Sub Envoi 'Gestion vitesse de la voiture
295     PWMA.analogWrite(Speed)
296     'PWMB.AnalogWrite(Speed) 'Second moteur non utilisé ici
297     Log("Vitesse: ",Speed)
298 End Sub
299
300 Sub straight_Ahead
301     Angle = MonServo.Read
302     If Angle < 90 Then
303         For i = Angle To 90
304             MonServo.Write(i)
305             Delay(10)
306         Next
307     End If
308     If Angle > 90 Then
309         For i = Angle To 90 Step -1
310             MonServo.Write(i)
311             Delay(10)
312         Next
313     End If
314 End Sub
315
316 Sub Right_Turn
317     For i = 90 To 125
318         MonServo.Write(i)
319         Delay(10)
320     Next
321 End Sub
```