

```

1
2 #Region Project Attributes
3     #AutoFlushLogs: True
4     #CheckArrayBounds: True
5     #StackBufferSize: 300
6 #End Region
7
8 ' GESTION D'UN PARKING VIRTUEL DE 12 PLACES (Maquette construite) DONT 10 sont OCCUPÉES AU
  DÉPART >>> 2 places restent libres
9 ' Marc DANIEL - via CARTE ARDUINO UNO complétée par l'utilisation du composant L293D -
  Février 2021
10 ' Maquette parking motorisée
11
12 Sub Process_Globals
13     Public Serial1 As Serial
14     Private pinButtonEntree As Pin                'broche pour le bouton d'entrée du
parking
15     Private pinButtonSortie As Pin                'broche pour le bouton de sortie du
parking
16     Private pinLEDVerte, pinLEDRouge, pinLEDbleues, pinLEDblanches As Pin 'broches pour
les LEDs
17     'LED VERTE = place(s) disponible(s) - LED ROUGE = PLUS DE PLACE(S)
18     'LEDs BLEUES = ENTRÉE D'UN VÉHICULE EN COURS
19     'LEDs BLANCHES - SORTIE D'UN VÉHICULE EN COURS
20     Private pinOuverture, pinFermeture As Pin      'broches pour les connexions
motorisation barrière
21     'pinOuverture >>> Moteur ouvrant la barrière - pinFermeture >>> Moteur fermant la
barrière (Inversion du sens de rotation grâce au composant L293D)
22     Public LEDVerte = False As Boolean
23     Public Places As UInt
24     End Sub
25
26
27 Private Sub AppStart
28     Serial1.Initialize(115200)
29     pinButtonEntree.Initialize(pinButtonEntree.A1, pinButtonEntree.MODE_INPUT_PULLUP)
30     pinButtonEntree.AddListener("pinButtonEntree_StateChanged")
31     pinButtonSortie.Initialize(pinButtonSortie.A0, pinButtonSortie.MODE_INPUT_PULLUP)
32     pinButtonSortie.AddListener("pinButtonSortie_StateChanged")
33     pinLEDVerte.Initialize(7, pinLEDVerte.MODE_OUTPUT)
34     pinLEDRouge.Initialize(9, pinLEDRouge.MODE_OUTPUT)
35     pinLEDbleues.Initialize(10, pinLEDbleues.MODE_OUTPUT)
36     pinLEDblanches.Initialize(11, pinLEDblanches.MODE_OUTPUT)
37     pinOuverture.Initialize(4, pinOuverture.MODE_OUTPUT) 'Connexion au composant L293D
borne n°2 ou 1a ou "IN1" (pour l'ouverture de la barrière
38     pinFermeture.Initialize(8, pinFermeture.MODE_OUTPUT) 'Connexion au composant L293D
borne n°7 ou 2a ou "IN2" pour la fermeture de la barrière
39     Places=2 '12 places de parking dont 10 sont déjà occupées par des véhicules = 2 places
encore disponibles
40     CallSubPlus("Depart", 0,0)
41
42 End Sub
43
44 Private Sub Depart
45     pinLEDVerte.DigitalWrite(True) 'allume la LED verte au départ de la session ou en
cas de RESET sur la carte ARDUINO UNO (il reste 2 places définies à la ligne 39)
46 End Sub

```

```
47
48 Private Sub pinButtonEntree_StateChanged(State1 As Boolean)
49     Log("État: ", State1) 'Log la valeur de State1
50     If State1 = False Then
51         If Places > 0 Then
52             pinLEDVerte.DigitalWrite(True) 'allume la LED verte (il reste des places) - NB: ↗
53             La LED verte sera allumée par défaut au départ car 2 places libres
54             pinLEDRouge.DigitalWrite(False) 'éteint la LED Rouge (Le parking n'est plus ↗
55             complet)
56             pinLEDbleues.DigitalWrite(True) 'Allumage des LEDs Bleues = Entrée d'un véhicule
57             CallSubPlus("Ouverture",500,0) ' Ouverture de la barrière
58             CallSubPlus("Pause",28000,0) 'Mouvement du véhicule (Entrée ou Sortie)
59             CallSubPlus("Fermeture", 35000, 0) ' Fermeture de la barrière
60             CallSubPlus("FinEntree",60000,0) ' Arrêt de la fermeture de la barrière + ↗
61             extinction des LEDs bleues
62             Places = Places -1
63             If Places = 0 Then
64                 pinLEDRouge.DigitalWrite(True) 'allume la LED rouge (Parking complet 0 place ↗
65                 libre)
66                 pinLEDVerte.DigitalWrite(False) 'éteint la LED Verte
67                 End If
68             End If
69             End If
70         End Sub
71
72 Private Sub pinButtonSortie_StateChanged(State2 As Boolean)
73     Log("État: ", State2) 'Log la valeur de State2
74     If State2 = False Then
75         If Places <> 12 Then
76             pinLEDVerte.DigitalWrite(True) 'allume la LED verte (il reste des places)
77             pinLEDRouge.DigitalWrite(False) 'éteint la LED Rouge
78             pinLEDblanches.DigitalWrite(True) ' Allumage des LEDs Blanches = Sortie d'un ↗
79             véhicule
80             CallSubPlus("Ouverture",500,0) ' Ouverture de la barrière
81             CallSubPlus("Pause",28000,0) 'Mouvement du véhicule (Entrée ou Sortie)
82             CallSubPlus("Fermeture", 35000, 0) ' Fermeture de la barrière
83             CallSubPlus("FinSortie",60000,0) ' Arrêt de la fermeture de la barrière + ↗
84             extinction des LEDs blanches
85             End If
86             If Places < 12 Then Places = Places + 1
87             End If
88         End Sub
89     End Sub
90
91 Private Sub Ouverture(Tag As Byte)
92     pinOuverture.DigitalWrite(True) ' ouvre la barrière pour entrée ou sortie d'un ↗
93     véhicule
94 End Sub
95
96 Private Sub Pause(Tag As Byte)
97     pinOuverture.DigitalWrite(False) 'La barrière reste ouverte - Mouvement du véhicule ↗
98     entrant ou sortant
99 End Sub
100
101 Private Sub Fermeture(Tag As Byte)
102     pinFermeture.DigitalWrite(True) 'fermeture de la barrière
103 End Sub
104
105
```

```
96 Private Sub FinEntree(Tag As Byte)
97     pinFermeture.DigitalWrite(False) 'Arrêt de la fermeture de la barrière
98     pinLEDBleues.DigitalWrite(False) 'Extinction des LEDs Bleues = Fin Entrée d'un  ↵
    véhicule
99 End Sub
100
101 Private Sub FinSortie(Tag As Byte)
102     pinFermeture.DigitalWrite(False) 'Arrêt de la fermeture de la barrière
103     pinLEDBlanches.DigitalWrite(False) 'Extinction des LEDs Blanches = Fin Sortie  ↵
    d'un véhicule
104 End Sub
105
106
107
108
109
110
111
```