```
1   from tkinter import *
2   import time
3   import os
4   import math
5   import motors as motor
6
7   # variables globales
8
9
10
11  joyXprev,joyYprev = 0,0
12  modulo = 0
13  angulo = 0
14  mouse_inscreen = 0
15  joyX = 0
16  joyY = 0
17
18  fwd = 0
19
20  #motor.rightMotor(100,1)
21  #motor.leftMotor(100,1)
22
23
24
25  '''
26  ################################ MOTORES
    ##########################################################################
27  GPIO.setmode (GPIO.BOARD)
28
29  RM_A = 10
30  RM_B = 8
31  LM_A = 3
32  LM_B = 5
33
34  pin_pwm_L = 7
35  pin_pwm_R = 12
36  GPIO.setup(RM_A ,GPIO.OUT)
37  GPIO.setup(RM_B,GPIO.OUT)
38  GPIO.setup(LM_A, GPIO.OUT)
39  GPIO.setup(LM_B, GPIO.OUT)
40
41  GPIO.setup(pin_pwm_L,GPIO.OUT)
42  GPIO.setup(pin_pwm_R,GPIO.OUT)
43
44  pwm_L = GPIO.PWM(pin_pwm_L,100)
45  pwm_R = GPIO.PWM(pin_pwm_R,100)
46
47  pwm_L.start(100)
48  pwm_R.start(100)
49
50
51
52  ############################## FUNCIONES VARIAS
    ##########################################################################
53  def leftMotor (velL):
54      pwm_L.ChangeDutyCycle(velL)
55      if fwd:
56          GPIO.output(LM_A,True)
57          GPIO.output(LM_B,False)
58
59      else:
60          GPIO.output(LM_A,False)
61          GPIO.output(LM_B,True)
62
63      print(velL)
64
65
66
67
68  def rightMotor (velR):
```

```
69          pwm_R.ChangeDutyCycle(velR)
70          if fwd:
71              GPIO.output(RM_A,True)
72              GPIO.output(RM_A,False)
73          else:
74              GPIO.output(RM_A,False)
75              GPIO.output(RM_A,True)
76          print (velR)
77
78      '''
79
80      def calculate_module (x,y):
81          module = x*x + y*y
82          module = math.sqrt(module)
83          return module
84
85      def create_circle(x, y, r, canvasName): #center coordinates, radius
86          x0 = x - r
87          y0 = y - r
88          x1 = x + r
89          y1 = y + r
90          return canvasName.create_oval(x0, y0, x1, y1)
91
92      def algulo ():
93          if joyX == 0:
94              return 90
95          else:
96              angulo = math.atan(joyY/joyX)
97          return angulo
98
99      def joystick (vector): # joystick
100         fwd = 0
101         joyX = vector.x*(2/3)-100
102         joyY = vector.y*(-2/3)+100
103         mouse_inscreen = 1
104         if joyX > 100 or joyX < -100 :
105             joyX = 0
106             joyY = 0
107             mouse_inscreen = 0
108
109         if joyY > 100 or joyY < -100 :
110             joyX = 0
111             joyY = 0
112             mouse_inscreen = 0
113
114         if mouse_inscreen == 1 and calculate_module(joyX,joyY) < 100:
115             modulo = calculate_module(joyX,joyY)
116             canvas.delete(ALL)
117             control_screen.delete(ALL)
118             canvas.create_line(150,150,vector.x, vector.y,width=3,fill='blue')
119             canvas.create_text(vector.x,vector.y+10, text = ("x = ",int (joyX),"y =
                ",int(joyY),"modulo = ",int(calculate_module(int (joyX),int(joyY))) ),
                fill = 'green')
120             create_circle(153,153,149,canvas)
121             control_screen.create_text(130,15, text = ("x = ",int (joyX),"y =
                ",int(joyY),"modulo = ",int(calculate_module(int (joyX),int(joyY))) ),
                fill = 'green')
122
123             leftV = 0
124             rightV = 0
125             angulo = 0
126
127             if joyX != 0:
128                 angulo = math.atan(joyY/joyX)
129             else :
130                         if joyY > 0:
131                                 leftV = int(modulo)
132                                 rightV = int(modulo)
133                                 fwd = 1
134                         else:
```

```
135                                 leftV = int(modulo)
136                                 rightV = int(modulo)
137                                 fwd = 0
138
139
140             if joyX > 0 and joyY >= 0: # cuadrante 1 giro alante derecha
141
142                     leftV = int(modulo)
143                     rightV = int((100 - math.cos(angulo)*100) * modulo / 100)
144                     fwd = 1
145
146
147             if joyX < 0 and joyY >= 0:# cuadrante 2 alante izquierda
148                 leftV = int((100 - math.cos(angulo)*100) * modulo / 100)
149                 rightV = int(modulo)
150                 fwd = 1
151
152
153             if joyX < 0 and joyY <= 0: #cuadrante 3
154                 leftV = int((100 - math.cos(angulo)*100) * modulo/100)
155                 rightV = int(modulo)
156                 fwd = 0
157
158
159             if joyX > 0 and joyY <= 0: # cuadrante 4
160                 leftV = int(modulo)
161                 rightV = int((100 - math.cos(angulo)*100) * modulo / 100)
162                 fwd = 0
163
164             motor.rightMotor(rightV,fwd)
165             motor.leftMotor(leftV,fwd)
166             #print (leftV,rightV)
167
168
169
170         #print ("click en:",int(joyX),int(joyY))
171         #print (mouse_inscreen)
172
173     def stop ():
174         joyX = 0
175         joyY = 0
176         module = 0
177         canvas.delete(ALL)
178
179
180
181
182     ############################### PANTALLA
        ##########################################################################
183
184     root = Tk()
185     root.title("DeltaDynamics'robot")
186     root.geometry("600x600")
187     root.config(bg="#3DC3D8")
188     root.config(bd=10)
189     root.config(relief="groove")
190
191
192     canvas = Canvas(width=300, height=300, bg='white')
193     canvas.place(x = 100, y =10 )
194     canvas.bind("<B1-Motion>", joystick)
195     canvas.config(relief="groove")
196     canvas.config(bd=1)
197     create_circle(153,153,149,canvas)
198     canvas.pack()
199
200     botonera = Frame()
201     botonera.place(x=0, y=350)
202     botonera.config(width=600, height=100)
203     botonera.config(bg ="#A53EF1")
```

```
204  botonera.config(relief="groove")
205
206  control_screen = Canvas(width = 300, height=30, bg='white')
207  control_screen.pack(pady = 5)
208
209  #botones
210
211  button_stop = Button(botonera,padx=50, pady=20, text="STOP", command = stop)
212  button_stop.pack(side = LEFT)
213
214  button_stop = Button(botonera,padx=50, pady=20, text="esto", command = stop)
215  button_stop.pack(side = LEFT)
216
217  button_stop = Button(botonera,padx=50, pady=20, text="dos cosas", command = stop)
218  button_stop.pack(side = LEFT)
219
220
221  root.mainloop()
222
```