

```

1 # Quadruped_0.py - MicroPython Version of ROBOTIS TSK3 code for Quadruped
2 # Remote Control with ENGINEER App
3 # All Rights Reserved - CNT Robotics LLC 2020
4
5 from pycm import *
6
7 def Touched():
8     global touch_coordinate, screen_coord_x, screen_coord_y
9     touch_coordinate = (int) (smart.read8(10310))
10    if (touch_coordinate > 0):
11        screen_coord_x = (int) ((touch_coordinate - 1) % 5 + 1) # Need to cast results to Integers
12        screen_coord_y = (int) ((touch_coordinate / 5) + 1)
13
14 def TorqueOffAllDynamixel():
15     global torque_on
16     torque_on = False
17     dxlbus.torque_off() # Torque Off All Dynamixels
18
19 def TorqueOnAllDynamixel():
20     global torque_on
21     torque_on = True
22     UpdateAllDynamixelPosition()
23     DXL(254).torque_on() # Torque On All Dynamixels
24
25 def ParseButton():
26     global touch_coordinate, screen_coord_x, screen_coord_y, motion_page_number, prev_touch_coordinate
27     global torque_on, SpeedUp
28     if (torque_on == True):
29         ParseJoystick()
30     if ((screen_coord_x > 3) and (touch_coordinate != 0)):
31         if ((touch_coordinate >= 14) and (torque_on == True)):
32             smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
33             motion_page_number = (int) (((touch_coordinate - 11) % 5 - 2) + (((touch_coordinate - 1) / 5) * 2 - 3)) # Need to cast results to Integers
34             PlayMotion()
35         elif ((touch_coordinate == 4) and (prev_touch_coordinate == touch_coordinate)):
36             if (torque_on == True):
37                 TorqueOffMelody()
38                 TorqueOffAllDynamixel()
39             else:
40                 TorqueOnMelody()
41                 TorqueOnAllDynamixel()
42                 InitPose()
43         elif ((touch_coordinate == 17) and (prev_touch_coordinate != touch_coordinate)):
44             SpeedUp = (int) ((SpeedUp + 1) % 2)
45             SetMotionPlaySpeed()
46             smart.write8(10130, (SpeedUp + 1) ) # set Background Image on Mobile Device = SpeedUp + 1
47         elif ((touch_coordinate == 0) and (prev_touch_coordinate != touch_coordinate)):
48             InitPose()
49
50 def SetMotionPlaySpeed():
51     global SpeedUp
52     if (SpeedUp == 0):
53         DXL(200).write8(64, 100) # set Motion Speed to 100 i.e. "regular" speed from Motion File
54     else:
55         DXL(200).write8(64, 120) # set Motion Speed to 120 i.e. sped up by 20%
56
57 def InitPose():

```

```

58     motion.play(1)
59
60 def UpdateAllDynamixelPosition():
61     DXL(200).write8(65, 3) # Address 65 undocumented by ROBOTIS
62     while (DXL(200).read8(65) == True):
63         delay(0) # essentially do nothing until Addr 65 reads False
64
65 def TorqueOnMelody():
66     buzzer.melody(0)
67     buzzer.wait()
68
69 def TorqueOffMelody():
70     buzzer.melody(1)
71     buzzer.wait()
72
73 def ParseJoystick():
74     global touch_coordinate, motion_page_number
75     motion_page_number = 0
76     if ((touch_coordinate >= 11) and (touch_coordinate <= 13)):
77         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
78         motion_page_number = touch_coordinate - 3
79         PlayMotion()
80     elif ((touch_coordinate >= 21) and (touch_coordinate <= 23)):
81         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
82         motion_page_number = touch_coordinate - 10
83         PlayMotion()
84     elif (touch_coordinate == 6):
85         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
86         motion_page_number = 16 # Turn Left
87         PlayMotion()
88     elif (touch_coordinate == 8):
89         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
90         motion_page_number = 17 # Turn Right
91         PlayMotion()
92     elif (touch_coordinate == 16):
93         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
94         motion_page_number = 14 # Left
95         PlayMotion()
96     elif (touch_coordinate == 18):
97         smart.write8(10350, 10) # SMART Vibrate for 0.1 sec.
98         motion_page_number = 15 # Right
99         PlayMotion()
100
101 def PlayMotion():
102     global motion_page_number
103     motion.play((int) (motion_page_number))
104     WaitForMotionComplete()
105
106 def WaitForMotionComplete():
107     global torque_on
108     while (DXL(200).read8(68) == True): # check Motion Status
109         if (torque_on == False):
110             break
111         delay(0)
112
113 # Main program starts here
114 smartphone_connected = False
115 touch_coordinate = 0

```

```
116 prev_touch_coordinate = 0
117 SpeedUp = 0
118 screen_coord_x = 0
119 screen_coord_y = 0
120 torque_on = False
121 motion_page_number = 0
122 melody_number = 0
123
124 while True:
125     if (smartphone_connected == False):
126         if (smart.is_connected() == True):
127             smart.wait_connected()
128             smart.write8(10010, 2) # set Landscape Display mode on ENGINEER App
129
130             SpeedUp = 0
131             SetMotionPlaySpeed()
132             smart.write8(10130,(SpeedUp+1)) # set Background Image = SpeedUp + 1
133
134             TorqueOnAllDynamixel()
135             InitPose()
136
137             smartphone_connected = True
138         else:
139             delay(0) # do nothing
140     else:
141         Touched() # get Touch Area 1 status
142         ParseButton()
143         prev_touch_coordinate = touch_coordinate
144
145
146
147
148
149
```