Performing actions

1.1 Experimental purpose

In this course, we will learn how to control DOGZILLA realize stabilizing mode.

1.2 Experimental preparation

The functions of dogzilla Python library involved in this course.

Dogzilla robot dog integrates a 9-axis IMU posture sensor, and adjusts its posture to keep its back level according to the data fed back by the posture sensor.

The functions of dogzilla Python library involved in this course include:

imu(mode)：In the self stable state, the robot dog will automatically adjust the posture angle to keep the back at the horizontal position. It is not allowed to manually set the posture angle when it is turned on.

mode 0: close、1: open

1.3 Experimental process

Open the jupyterlab client and find the code path:

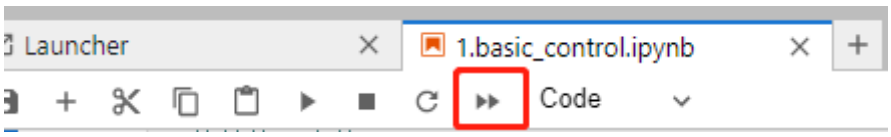
```
DOGZILLA/Samples/2_Control/5.stabilizing_mode.ipynb
```

By default g_ENABLE_CHINESE=False, if you need to display Chinese, please set g_ENABLE_CHINESE=True.

```
# 中文开关，默认为英文 Chinese switch. The default value is English
g_ENABLE_CHINESE = False

Name_widgets = {
    'Stabilizing_mode': ("Stabilizing_mode", "自稳模式"),
    'ON':("ON", "已开启"),
    'OFF':("OFF", "已关闭"),
}
```

Click the following icon to run all cells, and then pull to the bottom to see the generated controls.



布局控件并显示 Layout widgets and display them

```
output = widgets.Output()
box_display = widgets.VBox([button_Stable, output])
display(box_display)
```


```
Button clicked: Stabilizing_mode ON
Button clicked: Stabilizing_mode OFF
Button clicked: Stabilizing_mode ON
```

Click stabilizing_Mode button, open the self stable mode of the robot dog, and prompt to open it. Click again to close it.

1.4 Summary

In this course, we use JupyterLab controls DOGZILLA realize stabilizing mode.

The robot dog's self stabilization mode can keep the back level in the case of small slope or uneven road surface.