1、Control buzzer

1、Control buzzer
- 1.1Experimental goals
- 1.2 Experiment preparation
- 1.3 experiment procedure
- 1.4 Experiment summary

1.1Experimental goals

This course mainly learns how to control the robot's buzzer, automatically turn off the buzzer after 1 second, keep buzzing, and turn off the buzzer.

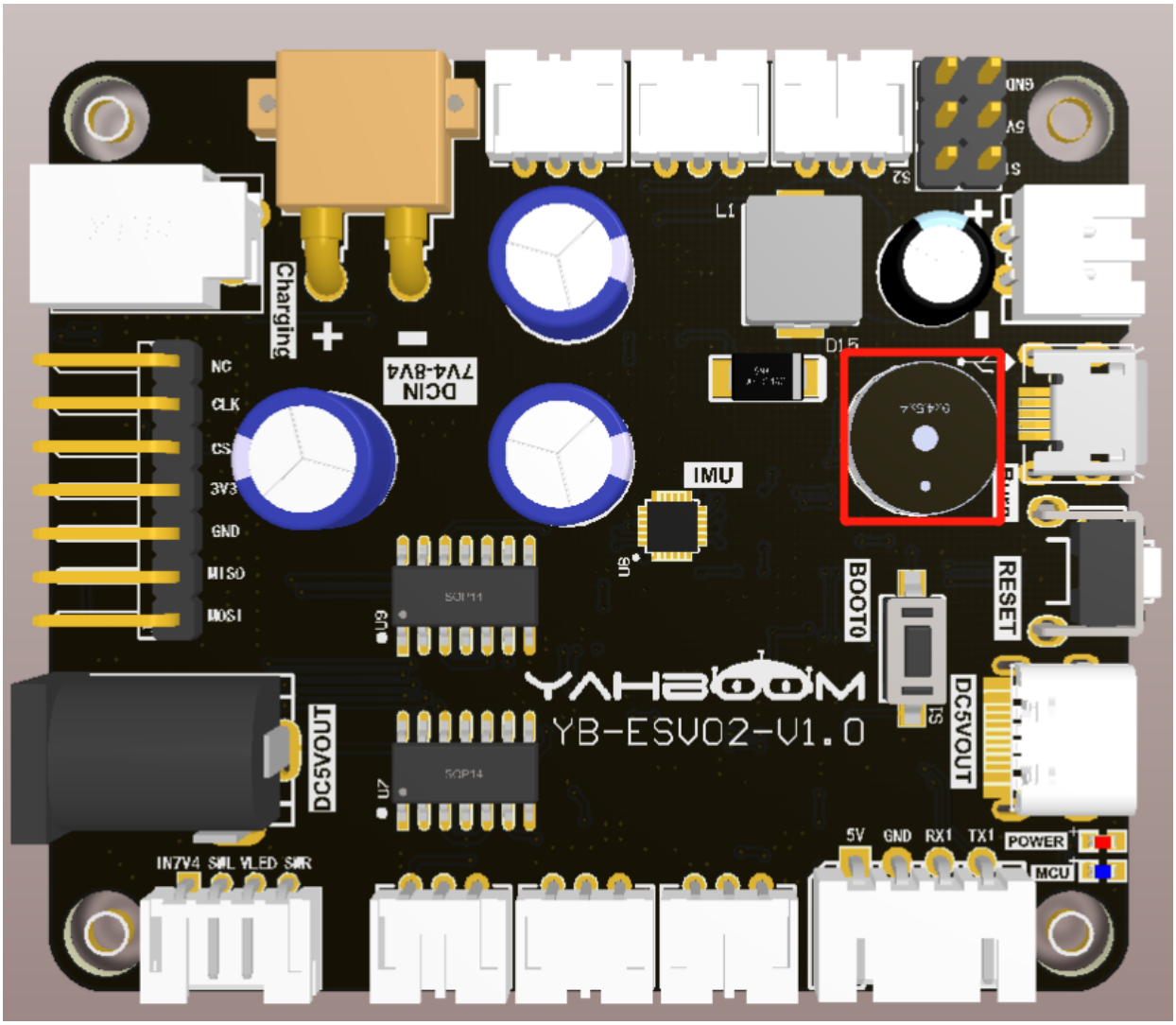
1.2 Experiment preparation

The functions of the Muto hexapod robot Python library involved in this course are:

buzzer(timeout)：Control the buzzer to sound.

The parameter timeout represents the timeout time of the whistle, and the value range is [0, 255]. If timeout=0, it means turning off the buzzer; timeout=255, means the buzzer keeps sounding; other values mean it will automatically turn off after timeout*100ms.

The location of the buzzer on the expansion board is as shown in the figure below：



1.3 experiment procedure

Open the jupyterLab client and find the code path：

```
muto/Samples/Control/1.buzzer.ipynb
```

Import and create a new Muto object g_bot from MutoLib.

```
from MutoLib import Muto
g_bot = Muto()
```

Turn on the buzzer to sound, and then automatically turn off after a delay of 1000 milliseconds.

```
timeout_100ms = 10
g_bot.buzzer(timeout_100ms)
```

Turn on the buzzer and it will not turn off automatically.

```
g_bot.buzzer(255)
```

Turn off the buzzer.

```
g_bot.buzzer(0)
```

1.4 Experiment summary

The buzzer control timeout interval is 100 milliseconds. After the timeout period, the buzzer automatically turns off. Within the effective range of the control buzzer, there are two special values, 0 and 255, which respectively indicate turning off and continuously turning on the buzzer. Other values will automatically turn off after timeout*100 milliseconds.