Imperial College London

DE2 Electronics 2

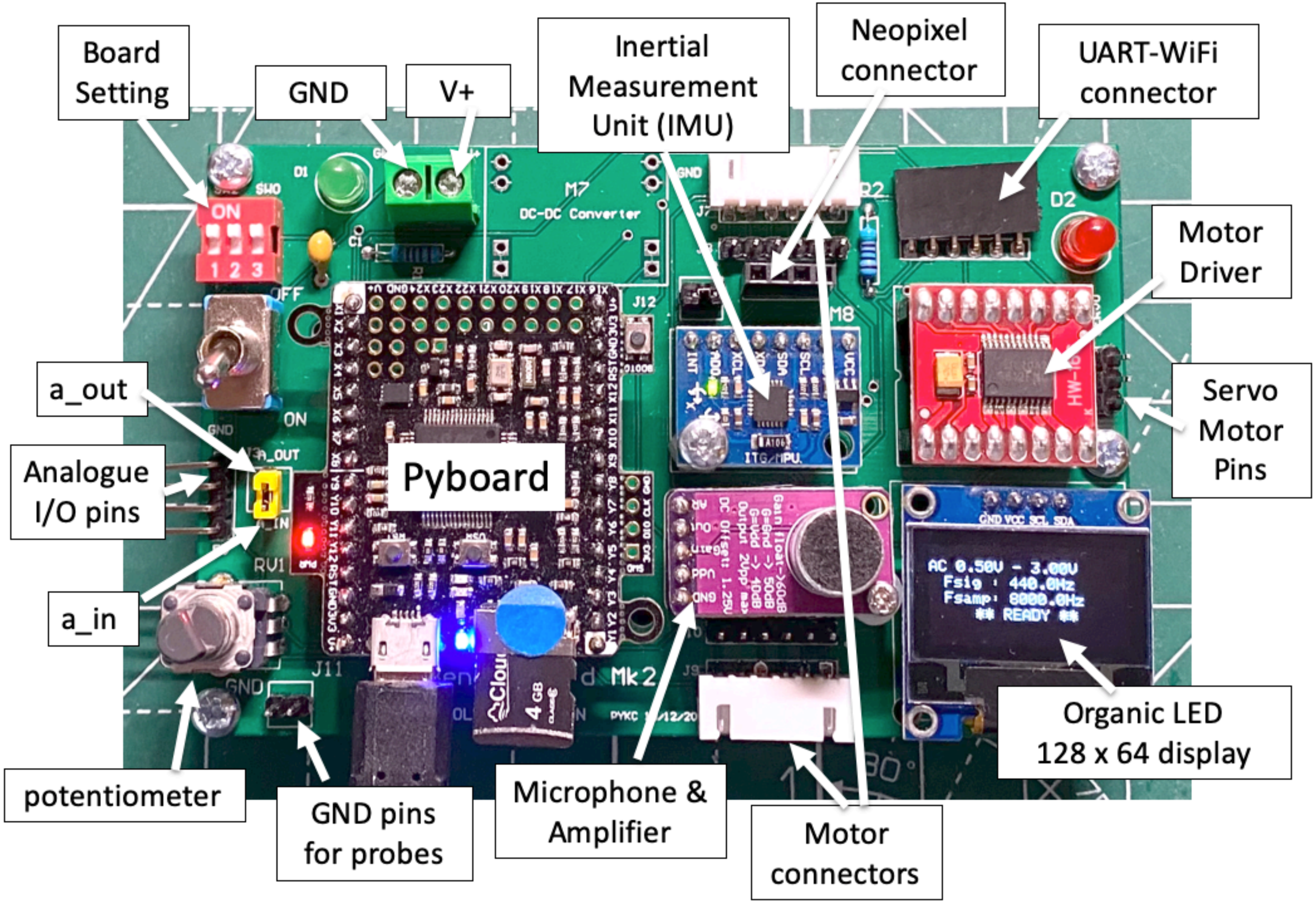
Tutorial 1

Lab 1 - PyBench Board & Spectral Analysis

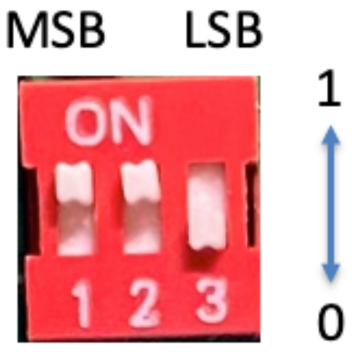
Peter Cheung
Dyson School of Design Engineering

URL: www.ee.ic.ac.uk/pcheung/teaching/DE2_EE/
E-mail: p.cheung@imperial.ac.uk

What's on the Board

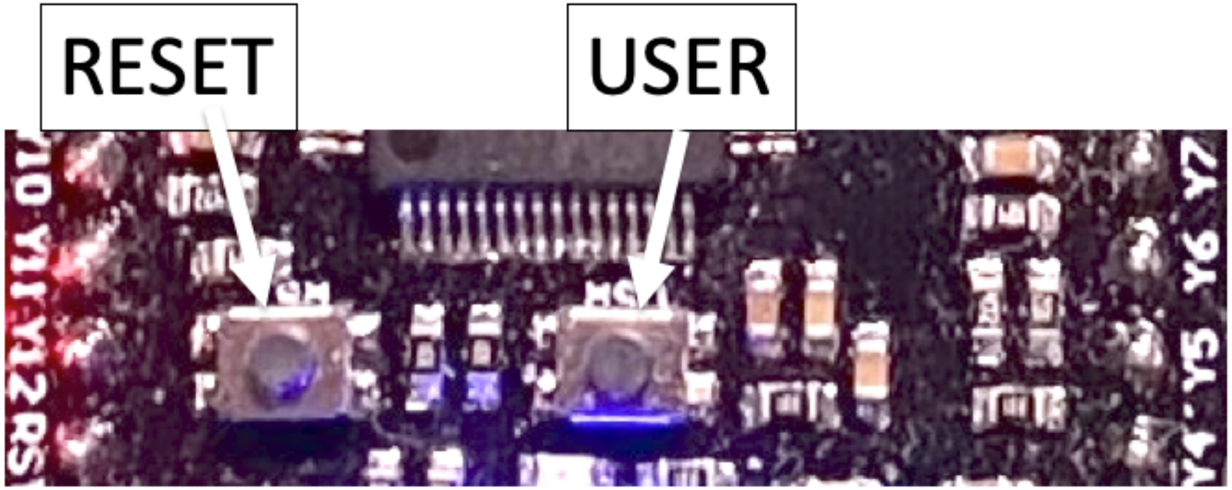


Board setting

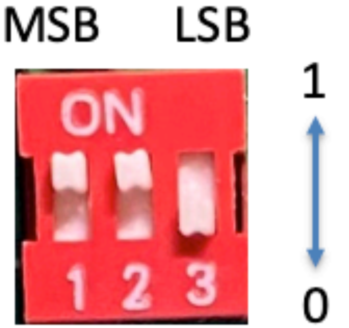


Setting = 110_2 or 6

Setting	Function
000 to 100	Reserved for user
101	Bulb Board Calibration
110	Pybench Self Test
111	Pybench v3.0

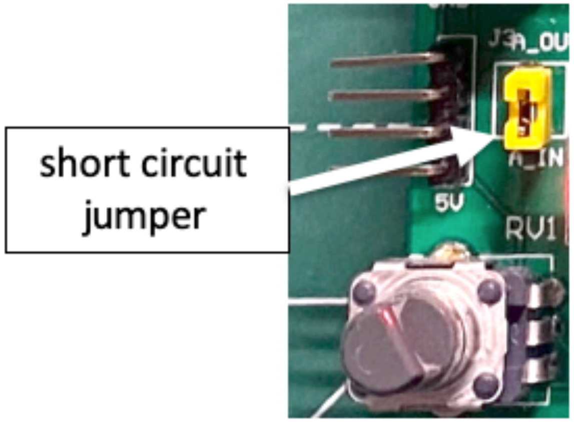


Self-test – Verify PyBench system works



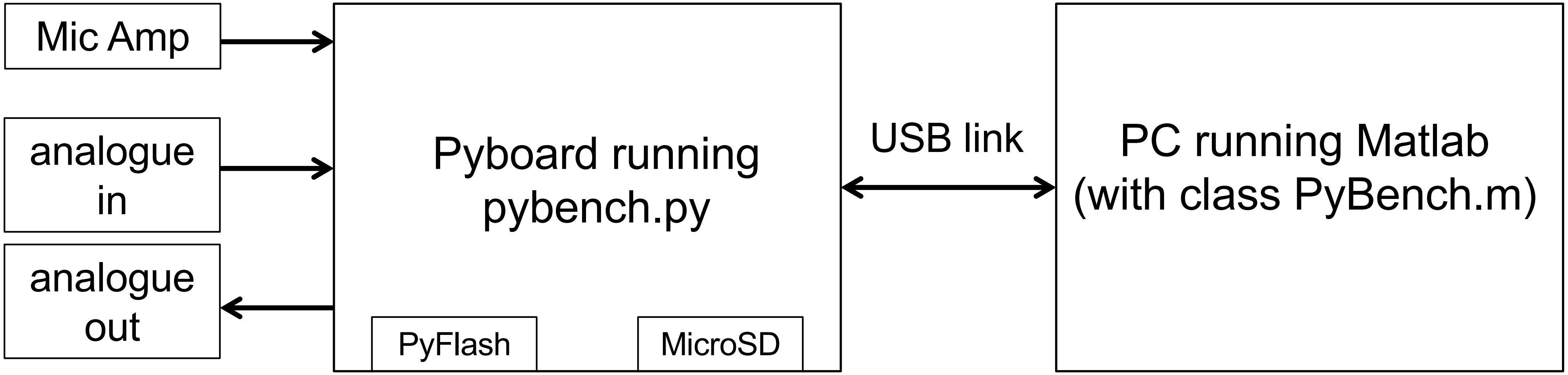
Setting = 110_2 or 6

Tests	Function
1	Test microphone
2	Test accelerometer (IMU)
3	~~Test motor and drive chip~~
4	Test ADC/DAC



short circuit jumper

How PyBench works?


Mic Amp
analogue in
analogue out
Pyboard running pybench.py
PyFlash
MicroSD
USB link
PC running Matlab (with class PyBench.m)

- Look for a serial link on computer:

```
ports = serialportlist;       % find all serial port
```

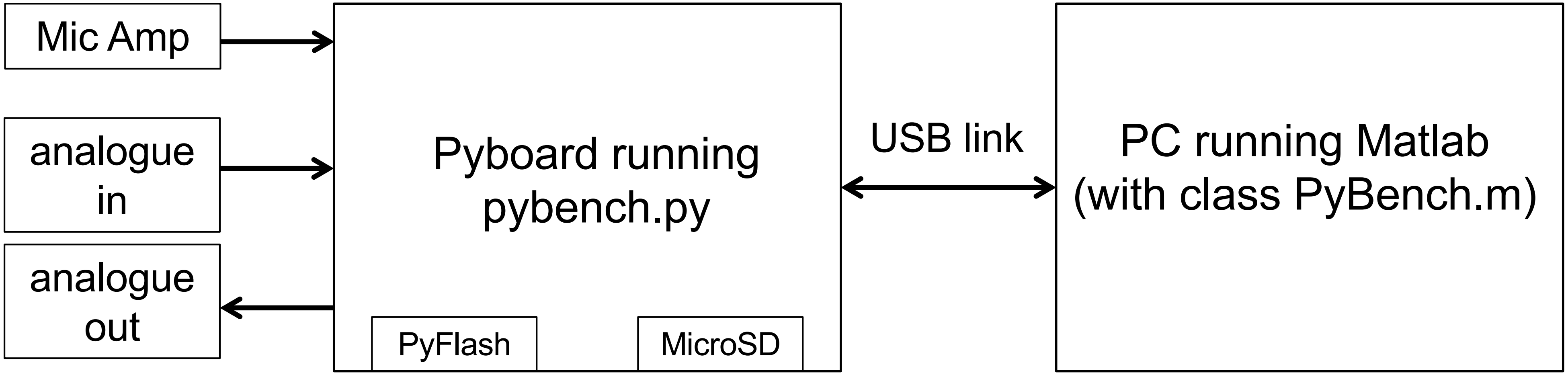
- Last one should be the one we want to use. This should be be **ports(end).**

- Create an object **pb** for the PyBench Board:

```
pb = PyBench(ports(end));    % create a PyBench object
```

- Control the Board via “methods”, e.g. **pb.set_max_v (2.5).**

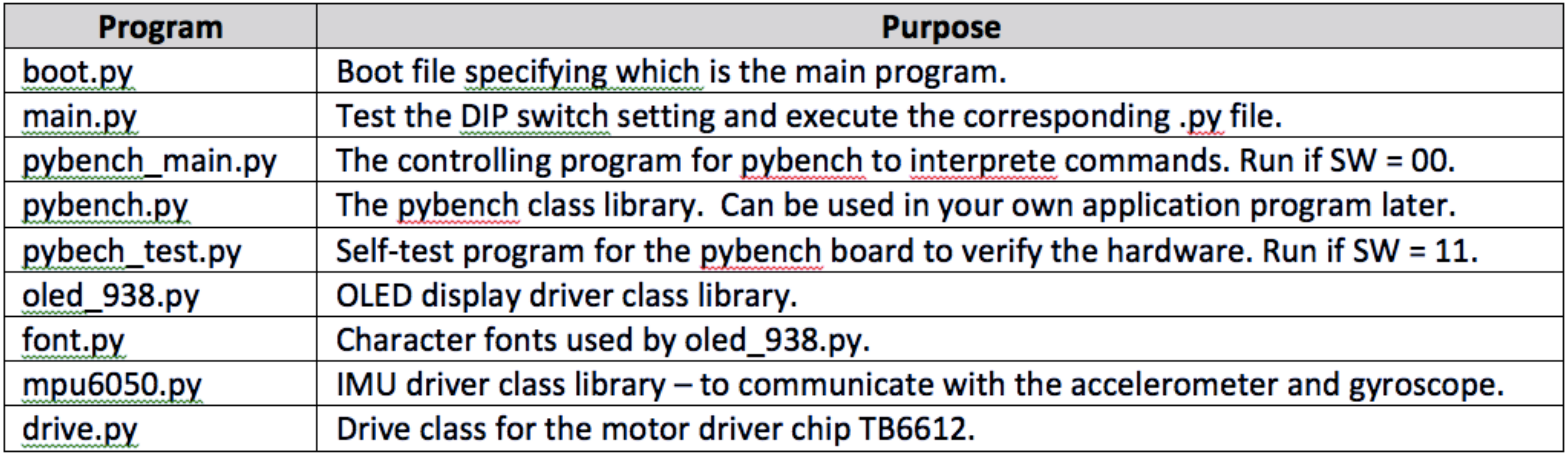
pb.set_max_v(2.5) explained


Mic Amp
analogue in
analogue out
Pyboard running pybench.py
PyFlash
MicroSD
USB link
PC running Matlab (with class PyBench.m)

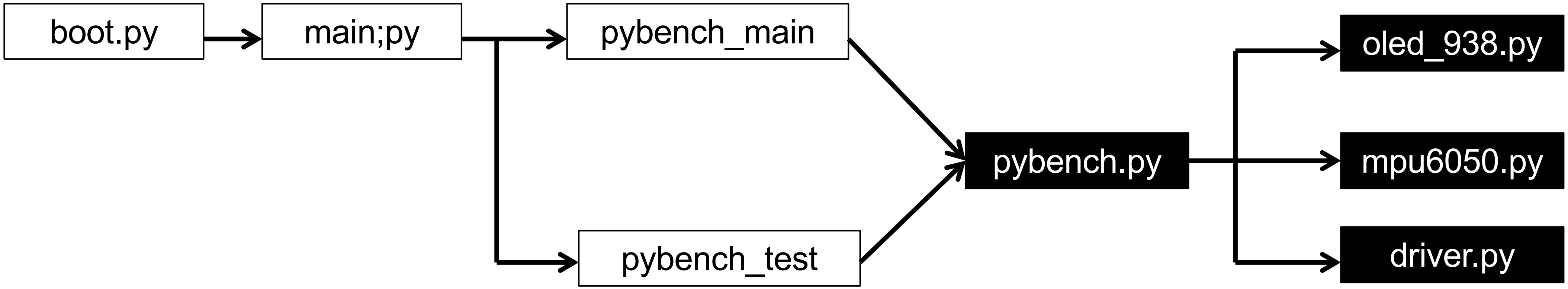
Here is what happens when you used this Matlab command: **pb = pb.set_max_v (2.5).**

- PC sends three bytes to PyBench board via USB link as serial data. First byte is a command character. In this case, 'X', followed by the value of voltage as two bytes. First byte is int (4096 * (v/3.3) / 256), and second byte is int (4096 * (v/3.3)) mod 256.
- All along, PyBoard is running a Python program (pybench.py) listening for a command. The BLUE LED is ON in this state. Waiting for a event such as a character to arrive is known as "**polling**".
- When it receives the command (3 bytes), the pybench.py code sets the maximum voltage of the ADC to 2.5V.

What are stored in the MicroSD card?



Program	Purpose
boot.py	Boot file specifying which is the main program.
main.py	Test the DIP switch setting and execute the corresponding .py file.
pybench_main.py	The controlling program for pybench to interprete commands. Run if SW = 00.
pybench.py	The pybench class library. Can be used in your own application program later.
pybech_test.py	Self-test program for the pybench board to verify the hardware. Run if SW = 11.
oled_938.py	OLED display driver class library.
font.py	Character fonts used by oled_938.py.
mpu6050.py	IMU driver class library – to communicate with the accelerometer and gyroscope.
drive.py	Drive class for the motor driver chip TB6612.


boot.py
main;py
pybench_main
pybench_test
pybench.py
oled_938.py
mpu6050.py
driver.py