

Micro Python & Pyboard

Is it an Arduino? Is it a Raspberry Pi? No it's another prototyping platform, with a built-in accelerometer. **Les Pounder** investigates.

The success of open source hardware platforms had brought forth many new ideas and projects, and the newest kids on the block are Pyboard and Micro Python. The project is the brainchild of Damien George from Cambridge, and is a tiny ARM microcontroller platform (the Pyboard), which is programmed using a lean re-write of Python 3 (Micro Python).

The board has two main ports: one for a micro USB connection, the other for MicroSD cards, which can be used to store your scripts. Next we can see a lot of tiny holes in the board, which provide access to the microcontroller and enable us to connect many different types of electronic components. There are also two microswitches: one is reserved for resetting the board; and the other can be used in your projects. Finally there are four small LEDs that are used to show activity/power to the board, but which can also be used in your projects.

To program the board, you simply connect it to your PC with a micro USB cable. After a few seconds the inbuilt 1MiB flash storage is mounted and available for use. In the storage you will see four files, but the two main files we are interested in **boot.py** and **main.py**. In **main.py** we can write our own Python project using our favourite editor, then save, eject the drive and reset the board. Hey presto: your project is working! The inclusion of a micro SD slot enables you to add further programs to your board and save information such as sensor data to the card.

Yes, but what can it do?

The board's functionality is comparable to an Arduino, with connections for PWM (Pulse Width Modulation) and analogue-to-digital conversion. For this review we wrote a simple script called **accel.py** (<https://github.com/lesp/Micro Python>) that would illuminate two LEDs based on how far the board was tilted in an X

```

main.py - /media/4421-0015/main.py
# main.py -- put your code here!
import pyb
from time import sleep
accel = pyb.Accel()
SENSITIVITY = 6
x1_pin = pyb.Pin.board.X1
g = pyb.Pin(pyb.Pin.board.X1, pyb.Pin.OUT_PP)
x12_pin = pyb.Pin.board.X12
h = pyb.Pin(pyb.Pin.board.X12, pyb.Pin.OUT_PP)

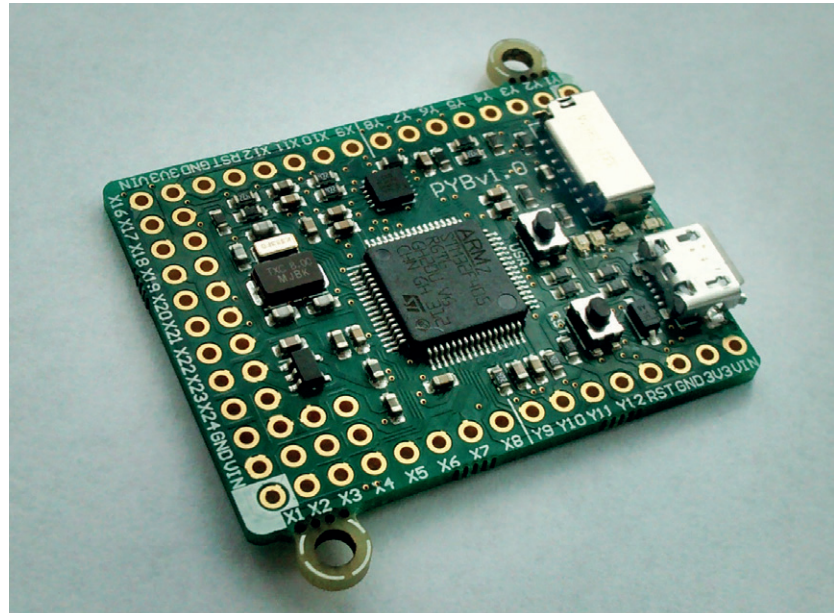
def lightup():
    g.high()

def lightup2():
    h.high()

while True:
    x = accel.x()
    y = accel.y()
    if abs(x) > SENSITIVITY:
        lightup()
        sleep(0.5)
    elif abs(y) > SENSITIVITY:
        lightup2()
        sleep(0.5)
    else:
        g.low()
        h.low()

```


The **main.py** file is just a text file on the internal memory.



The Pyboard measures just 33 x 40 mm and weighs 6g. There's a tutorial at <http://tinyurl.com/o529no5> that illustrates how easy it can be to use it with a servo.

and Y direction, accessing our favourite feature of the Pyboard: its accelerometer. We then connected the two LEDs, one to connection X1, the other to X12, as they were very near to ground (GND). It's not hard to see this being adapted to create a quadcopter.

So where does the board fit into the vast sea of products already on the market? Whereas the Raspberry Pi is a full computer, the Pyboard is a microcontroller in a similar vein to the Arduino platform, and we see it as an alternative to the Arduino for those who are already competent with Python. The use of the familiar Python language is a great boost to the board, enabling anyone to quickly prototype hardware projects.

The Pyboard is a remarkably well made piece of kit, with lots of expandability and potential for great projects. Couple this board with the lean implementation of Python 3 that is Micro Python and we have an alternative platform for Python-based hardware hacking. With none of the overheads that a Raspberry Pi has, by which we mean external devices and peripherals. If you need a small board for a single task then why not consider trying this on for size? 

DATA

Web
<http://micropython.org>
Developer
Damien George
Price
£28 (On sale from July)

LINUX VOICE VERDICT

Great connectivity, small size and lovely Python make this a great alternative to the Pi and Arduino.

