

Mbed HW 01 Report

Digital Input and Output

109033130 唐振家

一、Lab Description

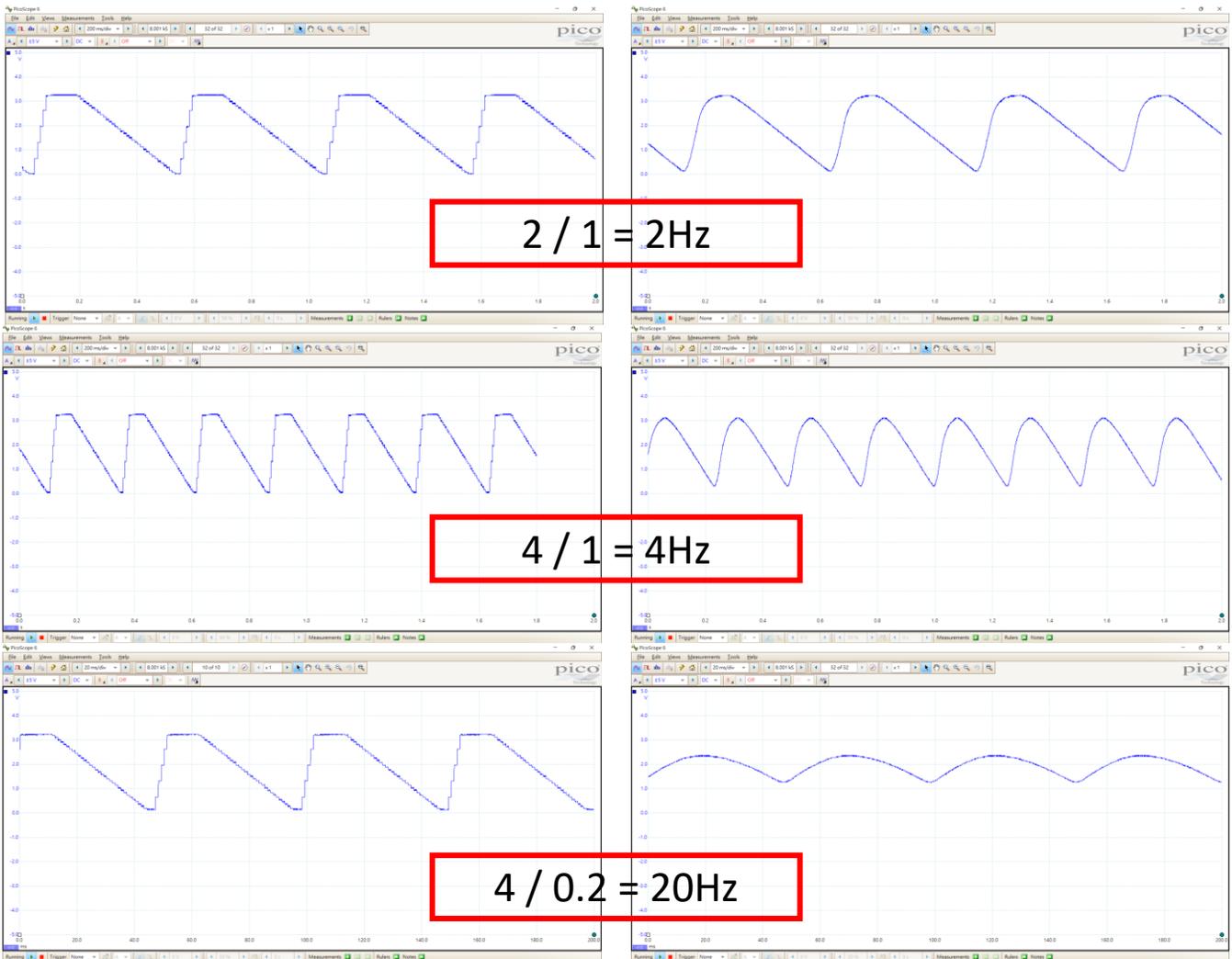
1、(Measure) Picoscope.

說明：

我的學號末碼為30，所以經過修正，做31的圖，analog output以間格0.02為區間，從0加到1，將波形切成 $1 / 0.02 = 50$ 份，前10%衝上1，維持到30%之後線性降到0。以下是我的波形經過以及未經過filter的結果。

頻率部份我設2、4、8、20，這是因為將波切成50份並用sleep的方式做delay就會受到限制，最短只能做到1毫秒， $1 * 50 \text{ ms} / 1 \text{ 個波} = 20\text{hz}$ 為上限，而其他頻率則是因為 $1000/50$ 能夠整除。所以可以做出來。

$f_c = \frac{1}{2\pi RC} \cdot 10R = 2000\Omega, C = 10\mu F$ ，算出Cut-off frequency為7.95hz，可以看出來在20Hz(大於8許多，超出Cut-off frequency)，振幅明顯下降。



一、Lab Description

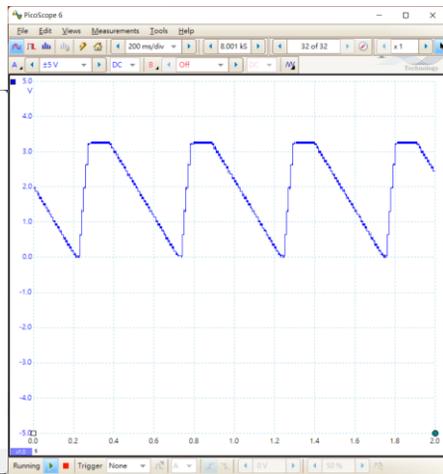
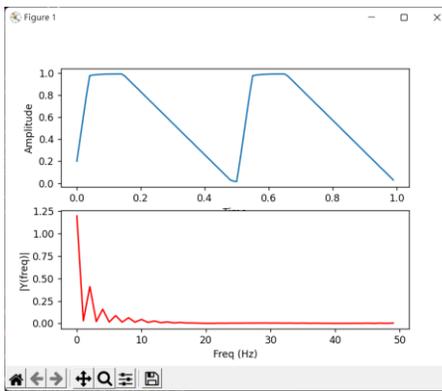
2、(Analysis 1) Python program

取樣用了一點技巧，試過取50個樣點，繪製出來的頻譜圖並不是太好，於是決定取100個點，因為取樣頻率為波形繪製頻率的兩倍，所以必須要在波形頻率的一半就取一次樣點，用變數counter計算經過幾個點，每經過frequency的一半個波，就取一個樣本點，例如頻率為20，就是在每10個點中取一個，將FFT.py黨中的sampling rate改成100即可。

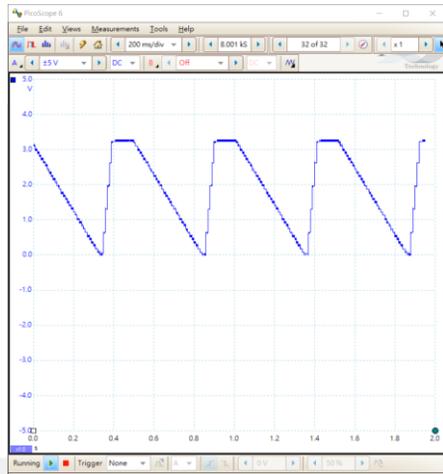
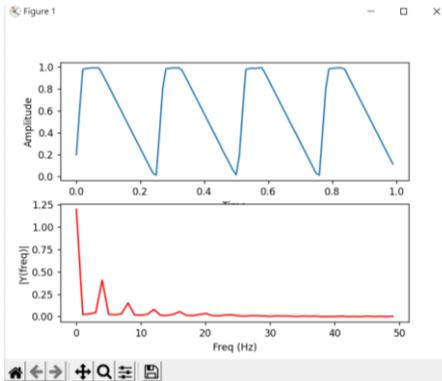
以下是使用FFT做取樣波形呈現的結果，為經過filter之前。

```
if (counter % (Fretype[type]/2) == 0 && counter < 100 * Fretype[type] / 2) {
    sample++;
    ADCdata[sample] = Ain;
} else if (sample == 99) {
    for (sample = 0; sample < 100; sample++) {
        printf("%f\r\n", ADCdata[sample]);
    }
}
counter++;
//1000ms / 50 circles / freq
ThisThread::sleep_for(1000/50/Fretype[type]);
}
```

2Hz



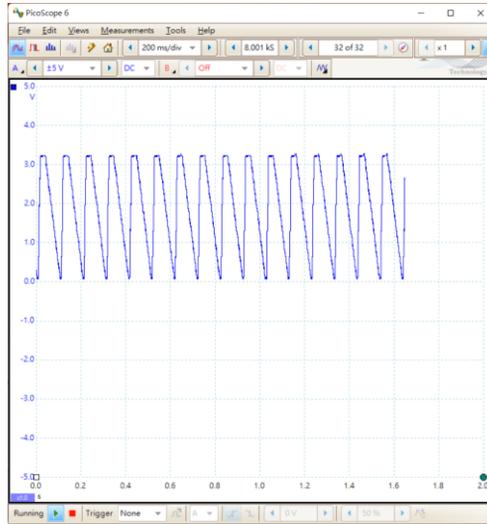
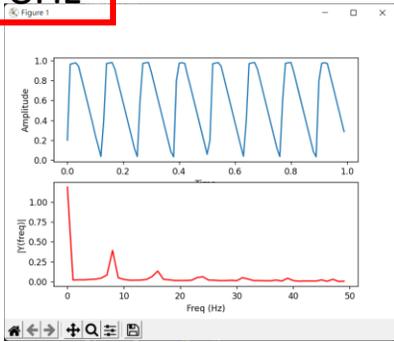
4Hz



一、Lab Description

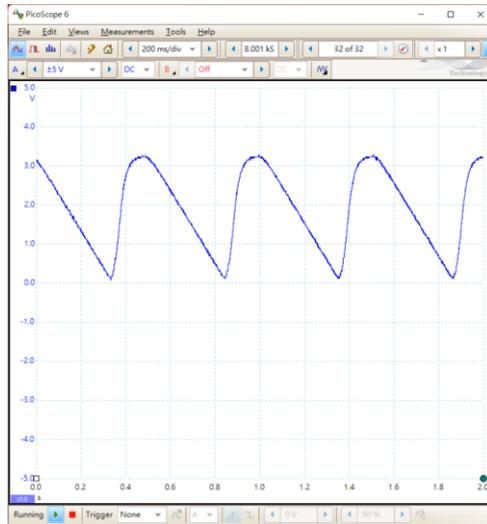
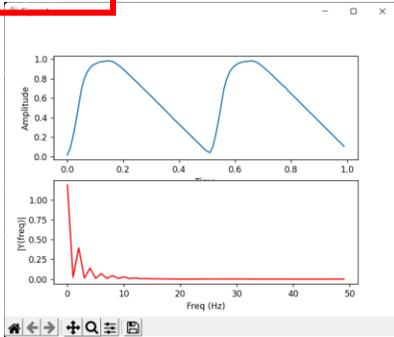
3、(Analysis 1) Python program

8Hz

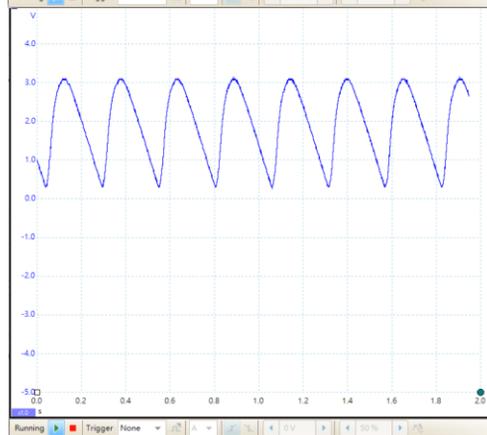
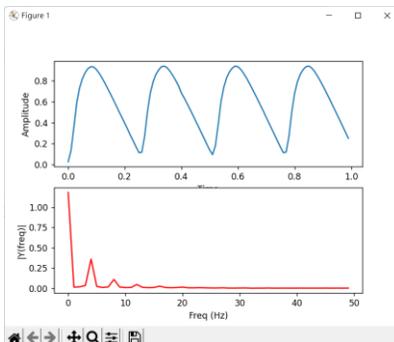


以下是使用FFT做取樣波形呈現的結果，為經過filter之後。

2Hz



4Hz

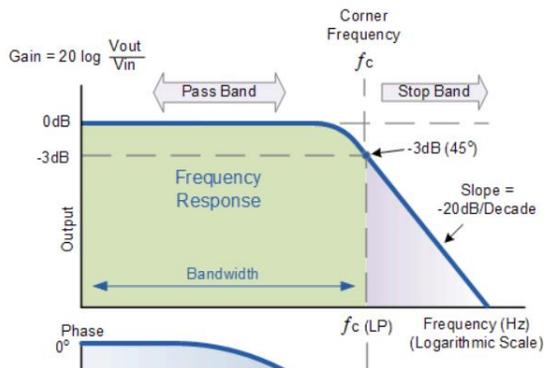


一、Lab Description

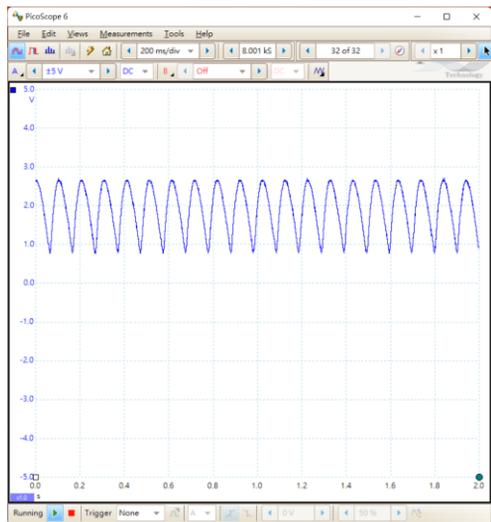
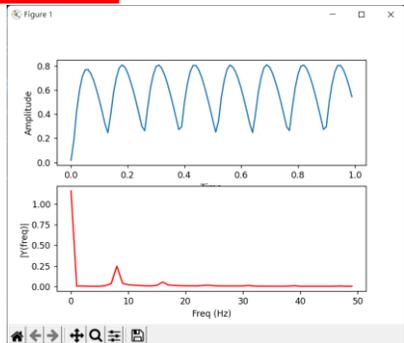
3、(Analysis 1) Python program

以下是使用FFT做取樣波形呈現的結果，為經過filter之後。

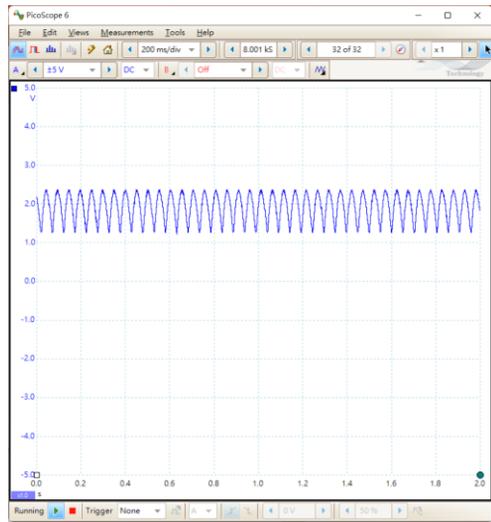
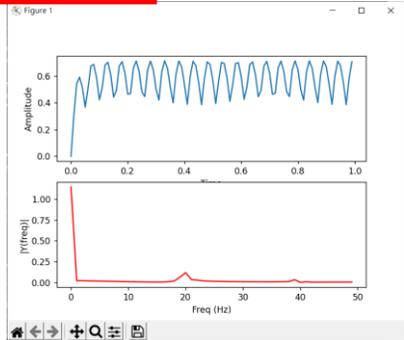
看出8Hz在Cut-off frequency臨界點，振幅已經略為縮小，而20Hz則是非常明顯超出Pass bond，進入stop bond，振幅原本應該高於3V，剩下約1V



8Hz



20Hz



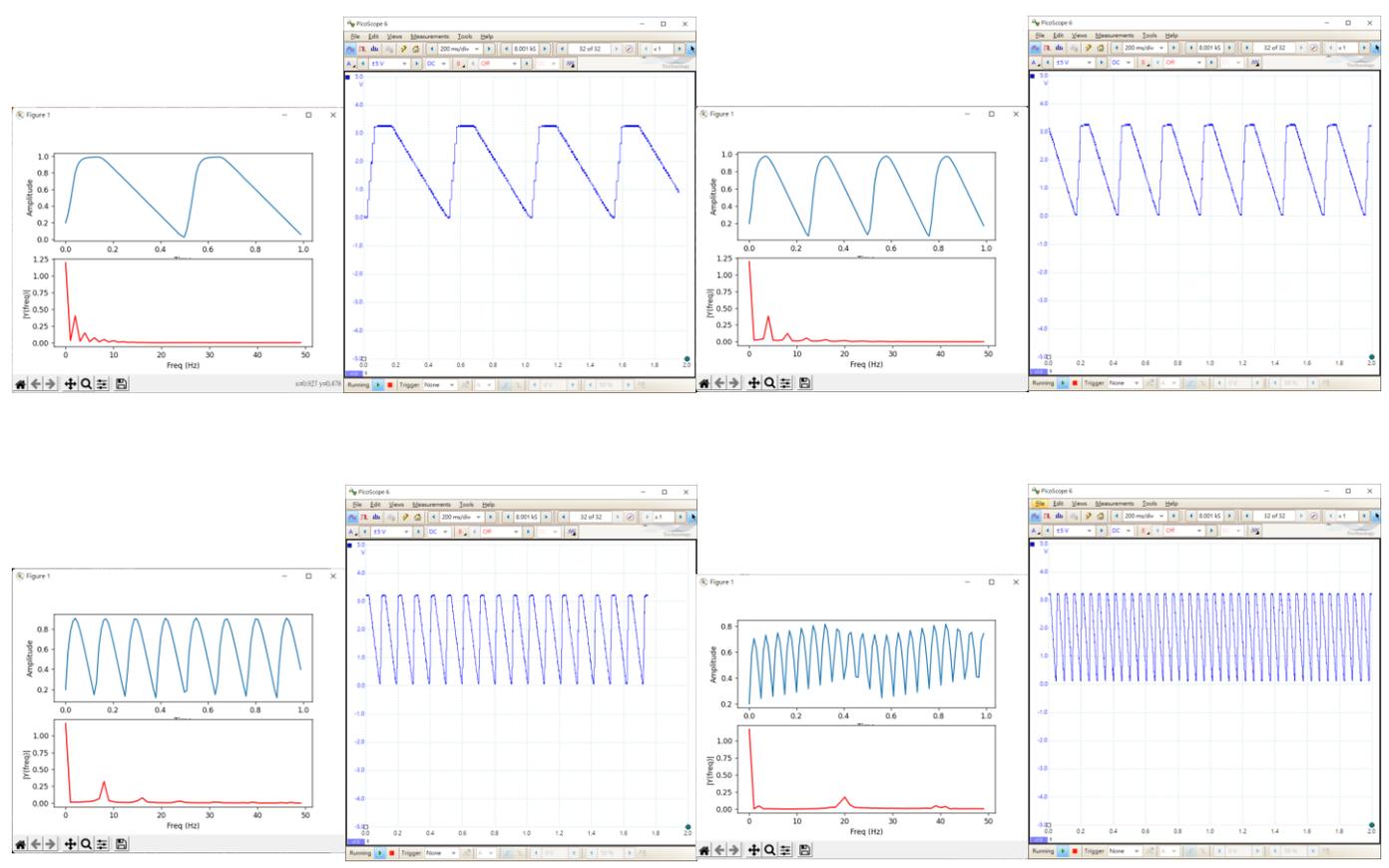
一、Lab Description

4、(Analysis 2) Python program

說明：

以下是在Python檔案中直接利用一階簡單低通濾波公式計算出來的結果，加上前一向後除以二，我們可以看見Picoscope中的振幅即使是在20Hz也仍然沒有受到影響，依然大於3V，而經過計算後的頻譜呈現較為平滑，在8Hz臨界值的時候跟經過真正的filter比較起來，都是振幅縮到0.2~0.8倍左右。但是20Hz時明顯用py模擬的計算結果會與真正的filter差異很大，實際上對振幅的減少會在0.4~1之間，而模擬則是在0.2~0.8之間。

```
for x in range(0, int(Fs)):  
    line = s.readline() # Read an echo string from B_L4S5I_IOT01A terminated with '\n'  
    if x == 0:  
        y[x] = float(line)  
    else:  
        y[x] = (float(line) + y[x-1]) / 2  
    # print line
```



三、遇到的問題

取樣的間隔計算問題

原本頻譜圖會跟實際頻率差2倍，透過觀察Picoscope的波形頻率，與想像相符，經過檢查發現，我原本取樣時，是以頻率數量作為間隔，例如頻率為20，就以20為間隔取一次，但這樣實際取樣時間卻是2秒，嘗試改python檔中的Ts(sampling interval)參數，卻無法執行，於是只好透過更改取樣計算，頻率為20，就以10為間隔取一次。

四、討論

學科與實驗結合

我是在這學期才修訊號與系統，剛好能將知識用在學科上，電子學也學到濾波器的一些相關知識，學到的東西能夠被實現的感覺非常好。