

EECS2020\_HW4

108032053 陳凱揚

## 1. Plot the signal in time domain and frequency domain

(magnitude spectrum)

下圖 1-1 及圖 1-2 分別為”sister\_8sec.wav”在 time domain 和 frequency domain 的圖形，而在聆聽音樂時可以發現大鼓在 8 秒鐘的音樂中約擊打了 8 次，其頻率約為 1Hz；且在 2.8 秒~2.84 秒間可以聽到一個完整的擊打聲，因此我們分別放大圖形去尋找鼓聲的波形在不同 domain 上的樣子為何。如下圖 1-3 為 time domain 的放大圖，可以看出波形在時間為 2.81 秒~2.83 秒時，相較於其他時間的波形較為雜亂、震盪較大，推測其應為大鼓的波形疊加至原始音樂上所導致；而下圖 1-4 為 frequency domain 的放大圖，推測大約在 1Hz 上方的 peak 代表著大鼓的擊打聲。

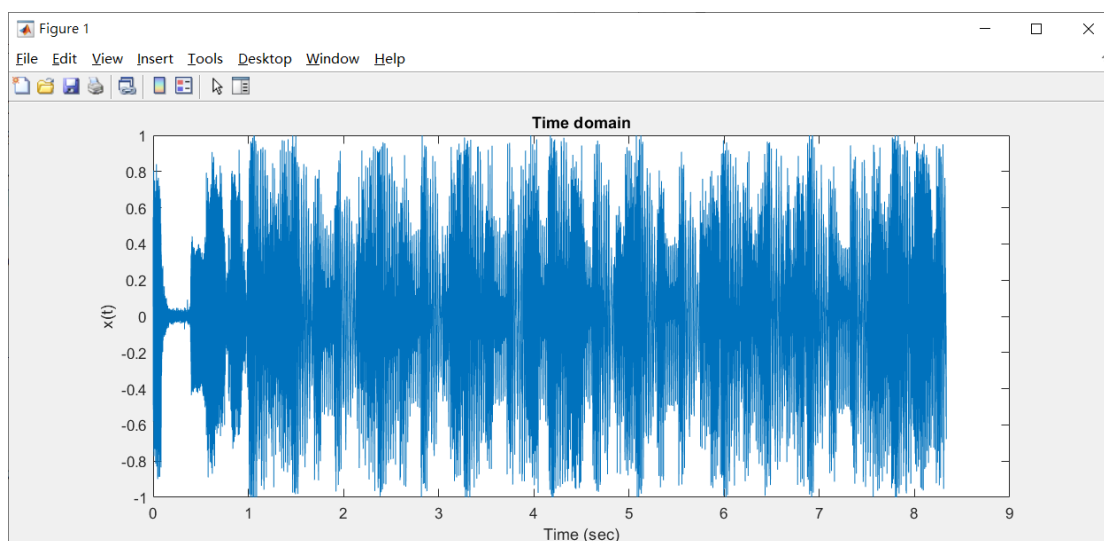


圖 1-1

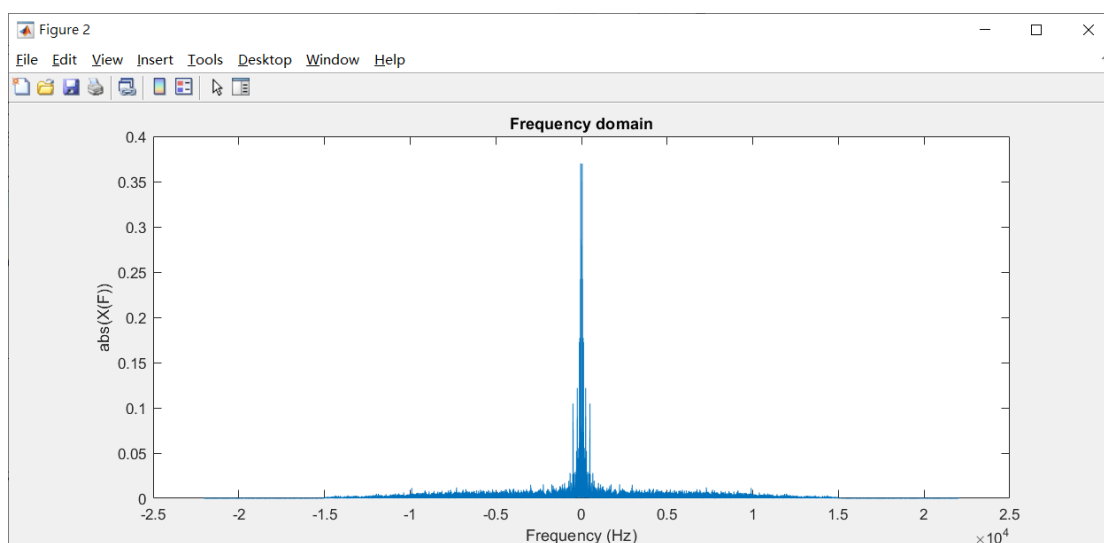


圖 1-2

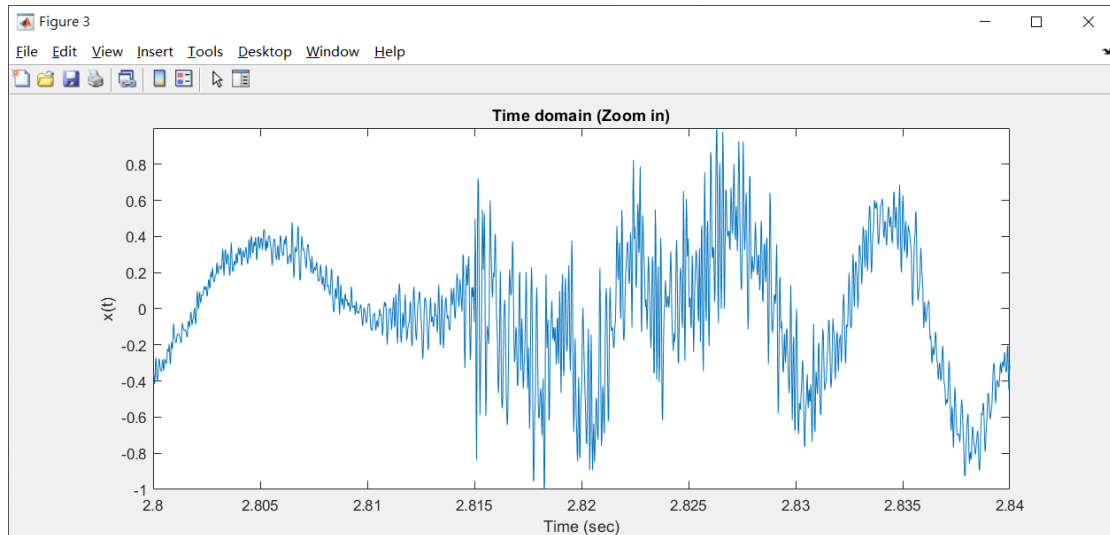


圖 1-3

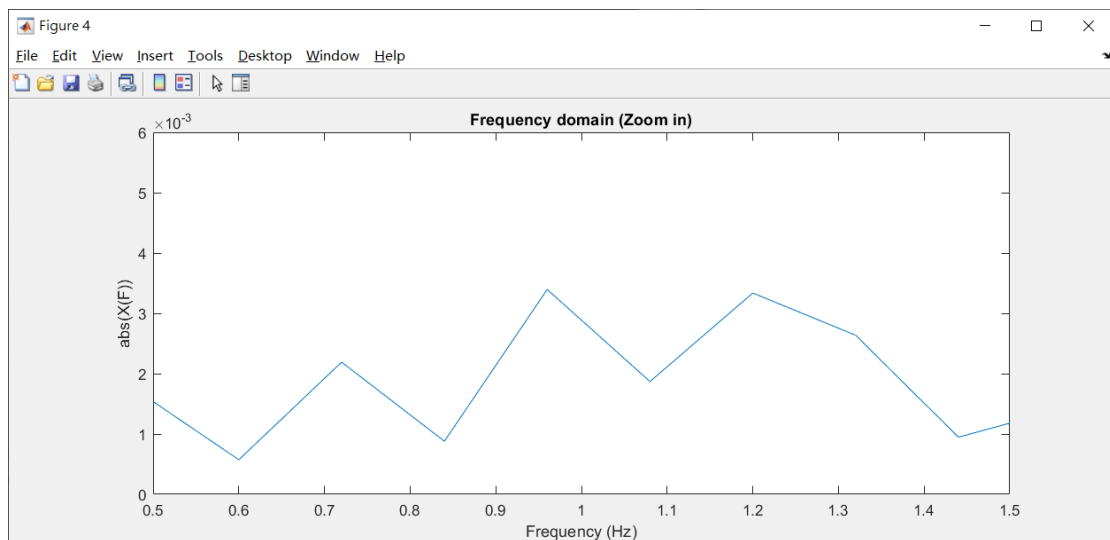


圖 1-4

## 2. Filter the music

- (a) 下圖 2-1~圖 2-6 分別為以不同的 filter order 所做出截斷頻率為 4000Hz 的 LPF 的 impulse response，而圖 2-7~圖 2-12 則為 frequency response。從這些圖形中可以看出隨著 order 的增加，impulse response 會越接近一個 digital sinc function，frequency response 可以看出圖形越接近一個 ideal LPF，其截斷頻率處的斜率絕對值會越大。而透過查詢 `fir1()` 的詳細介紹後，可以知道其功能是透過 window 的方式，以 FIR 來近似 IIR 的 filter，當 order 越大，window 觀測的範圍大小也會越大，其產生出來的 filter 也會越接近 ideal filter。此外，我們也了解到了這個函式的 window 並不是我們平常所用的 rectangular window，而是以 hamming window 的方式來減少 ripple 的

產生，其方式如下： $w[n] = \begin{cases} \frac{1}{2} [1 + \cos(\frac{\pi n}{M})] & , -M \leq n \leq M \\ 0 & , otherwise \end{cases}$ 。

最後，下圖 2-13~圖 2-18 為原始訊號及被 LPF 過濾後的訊號的 magnitude spectrum 比較圖，可以看出 LPF 的確發揮了功能，當 order 越大，效果也越明顯。

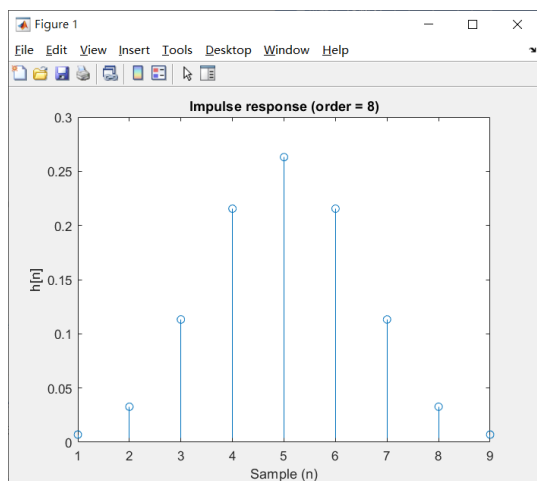


圖 2-1

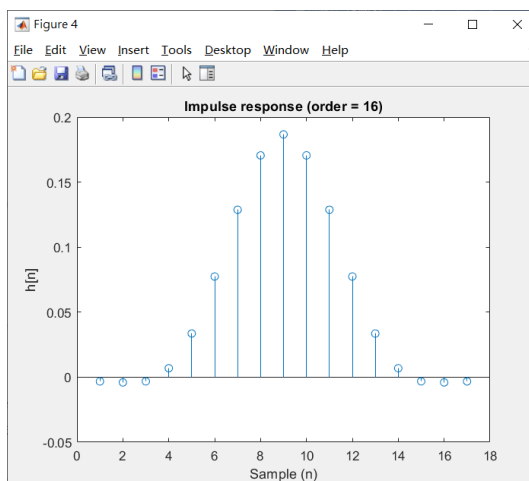


圖 2-2

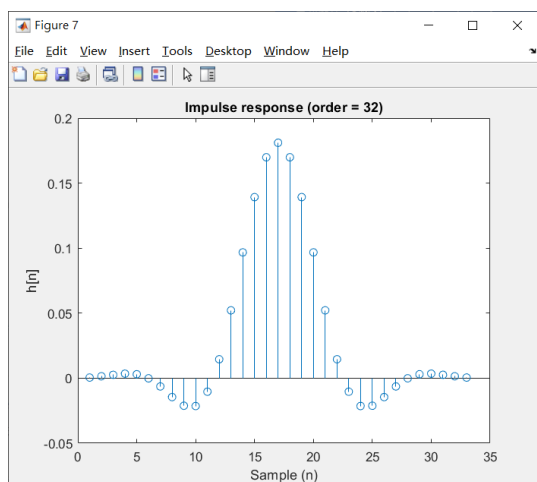


圖 2-3

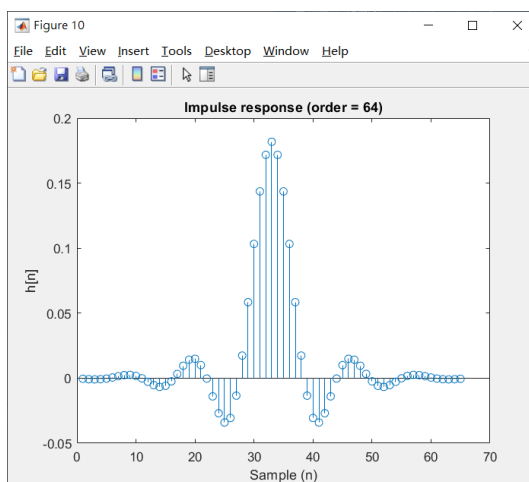


圖 2-4

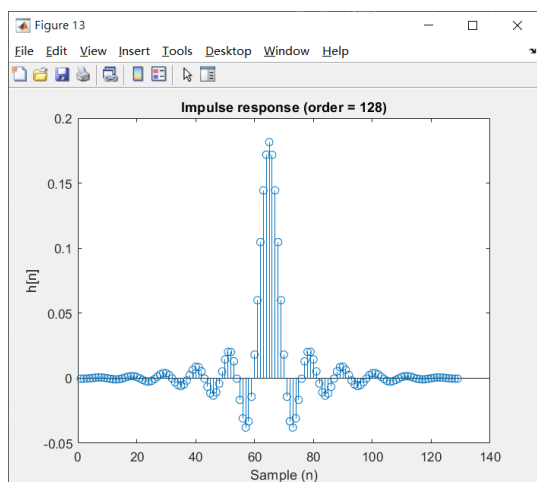


圖 2-5

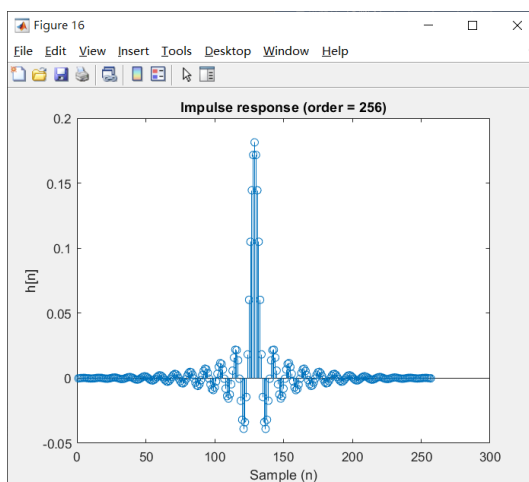


圖 2-6

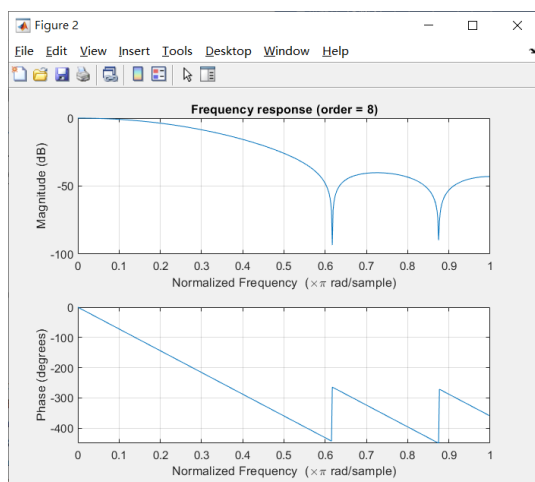


圖 2-7

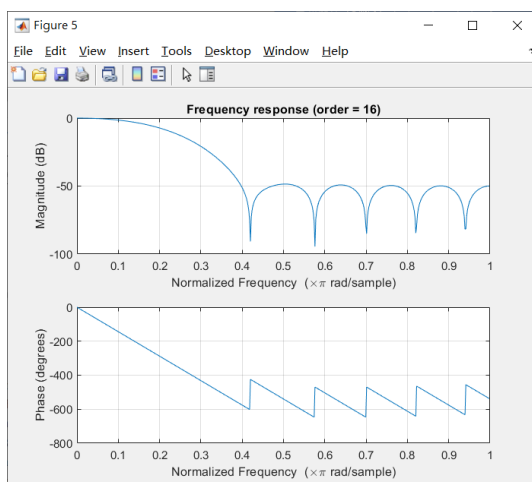


圖 2-8

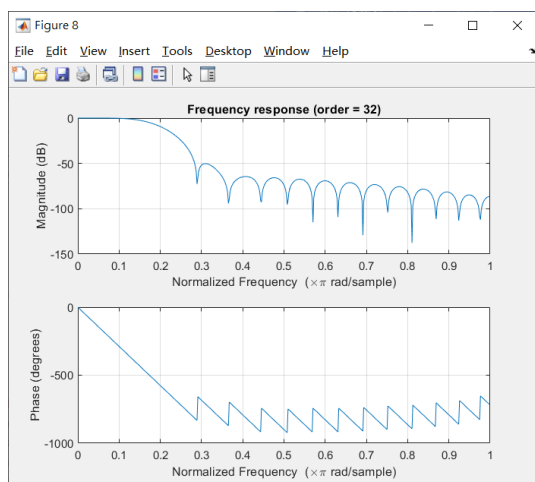


圖 2-9

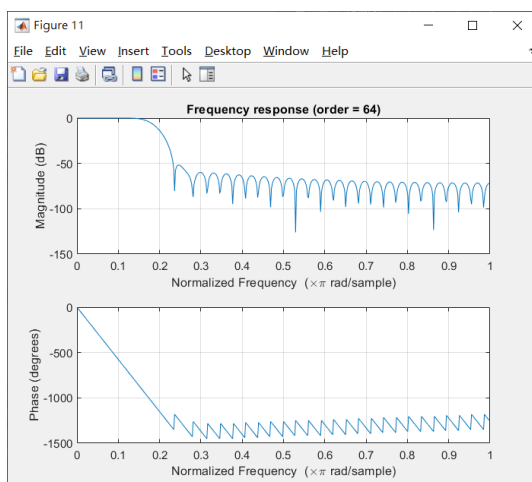


圖 2-10

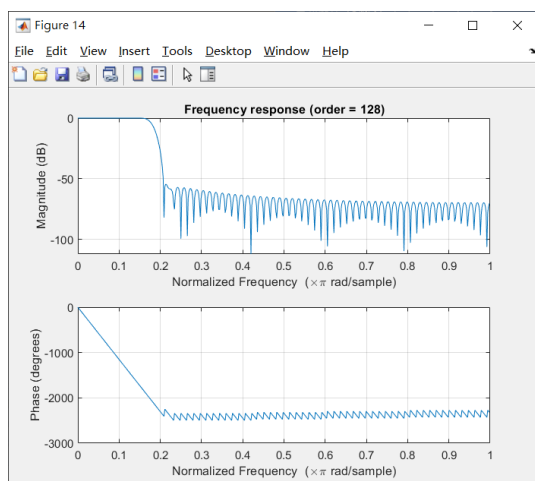


圖 2-11

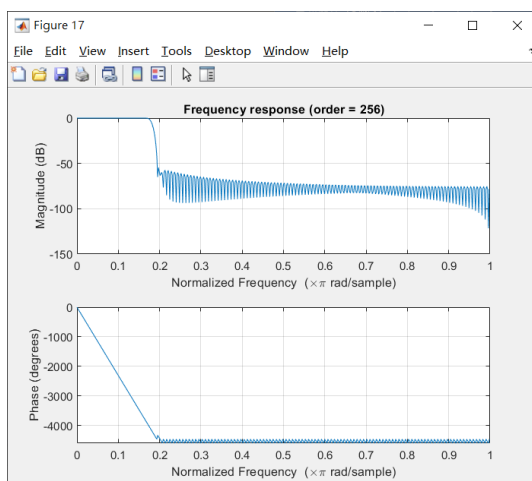


圖 2-12

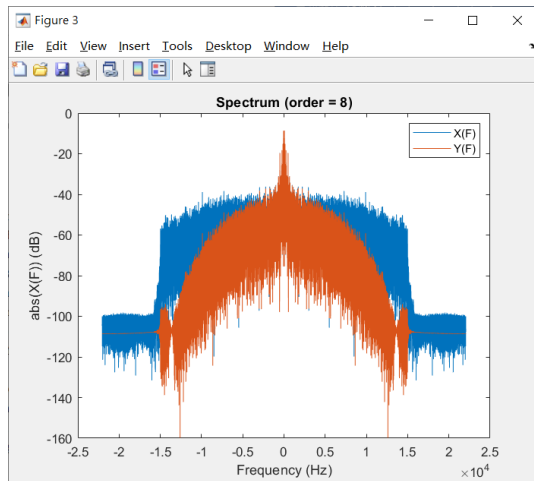


圖 2-13

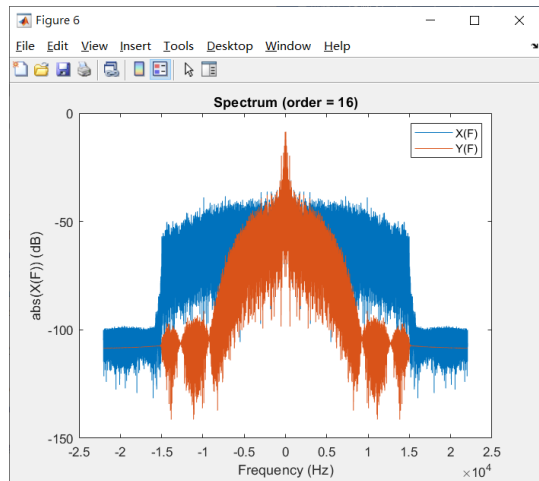


圖 2-14

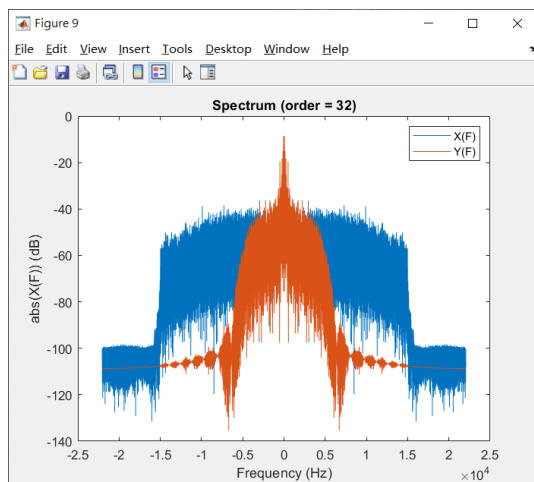


圖 2-15

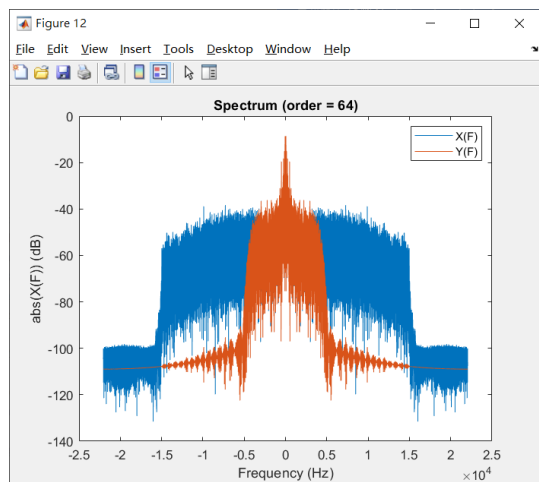


圖 2-16

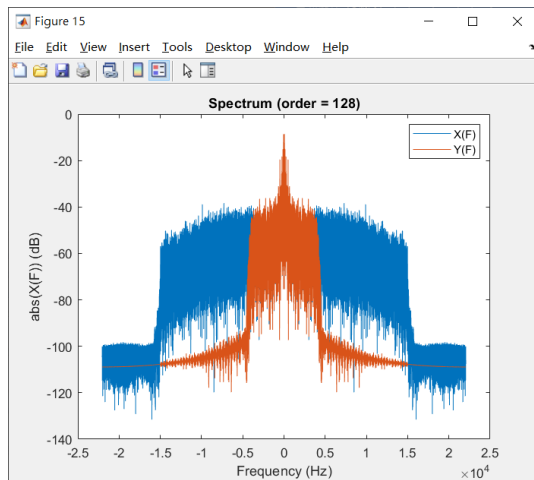


圖 2-17

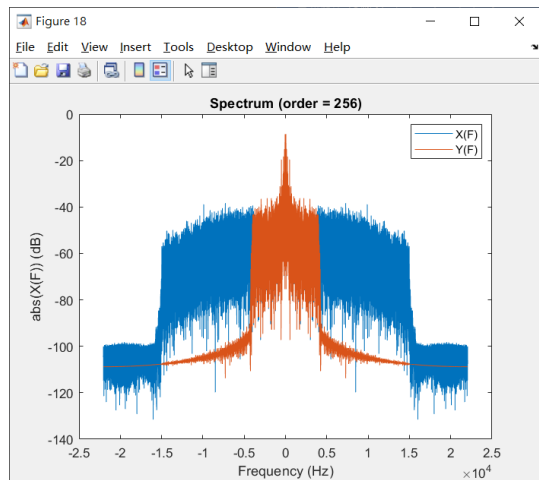


圖 2-18

(b) 我所設計的 LPF 和 HPF 的截斷頻率皆為 4000Hz，下圖 2-19、圖 2-20 分別為其 frequency response。通過 LPF 後的音訊，其人聲的部分會較為明顯，背景音樂的部分則比較弱，有些樂器甚至幾乎消失聲音了；而通過 HPF 的音訊，人聲的部分幾乎消失了，主要聲音為較高頻的銅鈸。此外，由於 filter 的 impulse response 的長度為 order+1，且由(a)小題

可知當 order 越大，截斷頻率處的落差會越大，因此當 filter 的長度越長，此 filter 就會越接近 ideal filter，輸出的音訊更接近於我們想要的結果，不想要的 frequency 也殘留的越少。

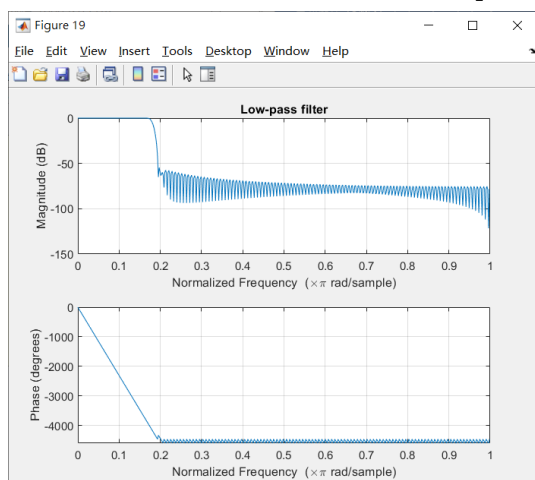


圖 2-19

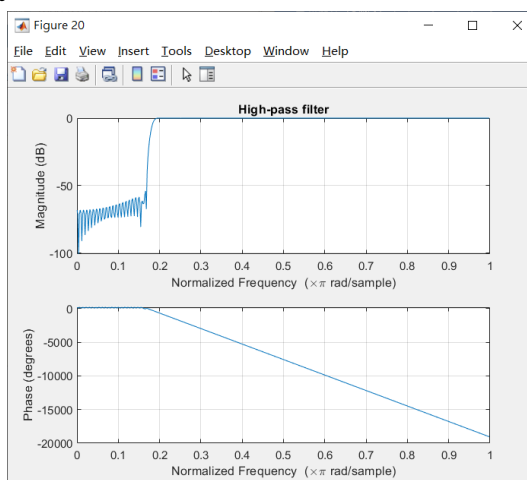


圖 2-20

- (c) 下圖為我們所設計之截斷頻率為 400Hz~9000Hz 的 band-stop filter 的 frequency response，通過此 filter 後輸出的音訊，盡可能地減少了人聲的部分，使背景音樂相對來說較大聲。若我們知道人聲主要出現在哪些頻率範圍，即可透過對應的 band-stop filter，來消除掉這些人聲，使音訊變成只有背景音樂的卡拉 ok 版本。

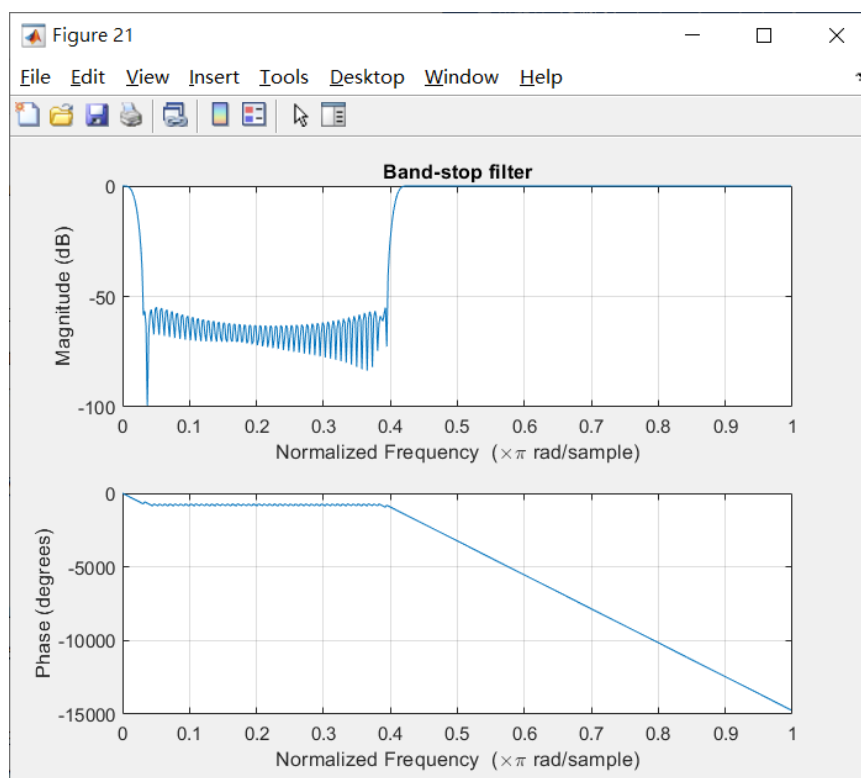


圖 2-21

### 3. Re-sampling the ECG and make the ECG audible

(a) 我們希望將取樣頻率  $F_s = 1800$  的音訊 resample 成  $F_s' = 44100$  的音訊，

由  $F_s' = \frac{I}{D} F_s$  的關係式可知  $I = 49, D = 2$ ，下圖 3-1 為講義介紹的兩種

resample 方式，我所採用的是 System 1，並將兩個 LPF 合成一個 LPF，其 block diagram 如下圖 3-2 所示。此外，由於講義上是以  $F_s$  做標準化，函式  $fir1()$  是以  $F_s/2$  做標準化，因此  $F_{cut}$  必須做調整，如下：

令  $\Omega$  為 absolute angular frequency,  $\omega, \omega'$  分別為講義、 $fir1()$  所使用的 normalized angular frequency,  $f, f'$  為 normalized frequency。

$$\Omega = F_s * \omega = \frac{F_s}{2} * \omega' \rightarrow \omega = \frac{1}{2} \omega' \rightarrow f' = 2f$$

$$\omega_{cut} = \min\left(\frac{\pi}{I}, \frac{\pi}{D}\right) \rightarrow f_{cut} = \min\left(\frac{1}{2I}, \frac{1}{2D}\right) \rightarrow f'_{cut} = \min\left(\frac{1}{I}, \frac{1}{D}\right)$$

下圖 3-3、圖 3-4 分別為我們所做的 resample 的輸出訊號及函式  $resample()$  所做的輸出訊號的 magnitude spectrum，而圖 3-5、圖 3-6 為放大後的結果，可看出其圖形相當接近，都有消除掉由 upsampling 所產生的 image，使我們在做 interpolation 和 decimation 時，不會在 frequency domain 產生 aliasing，令訊號在 resample 後依然保持著原始資訊。

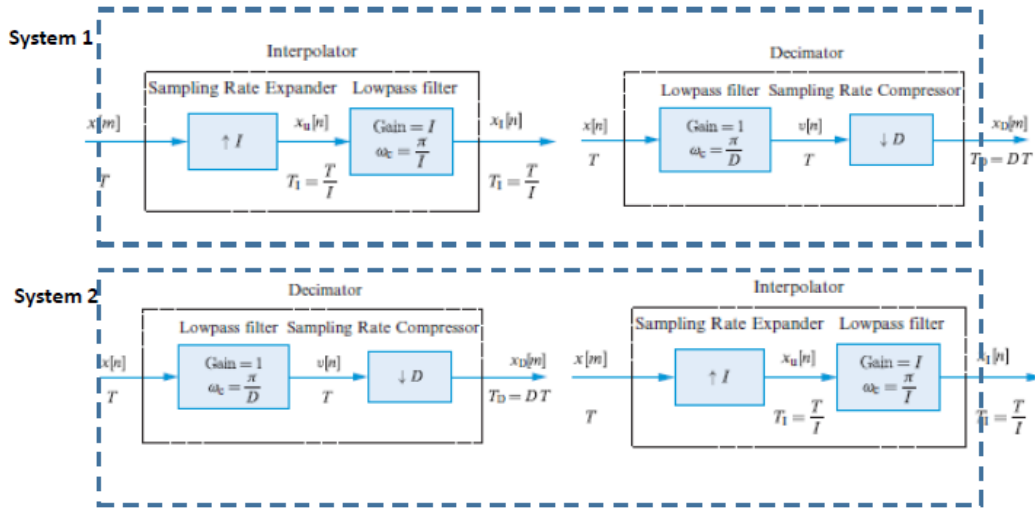


圖 3-1



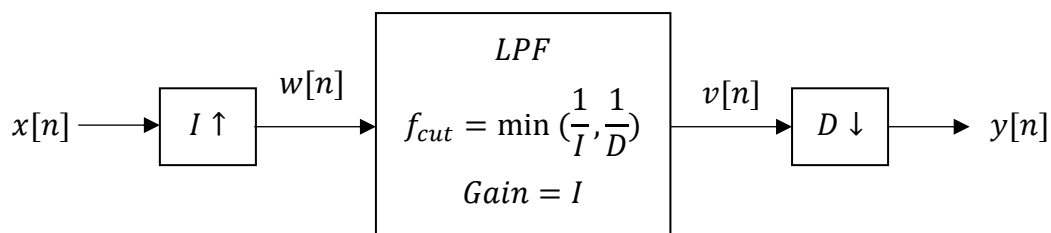


圖 3-2

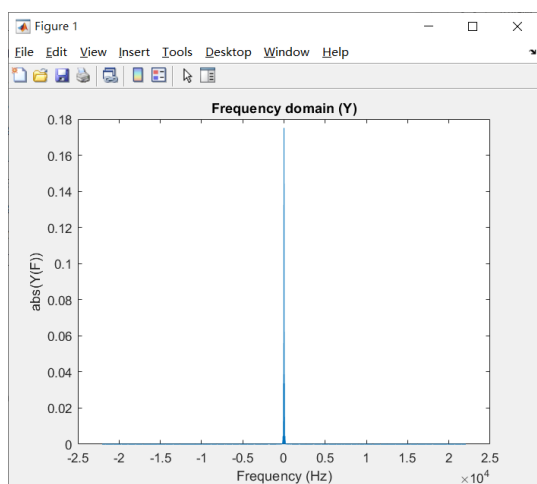


圖 3-3

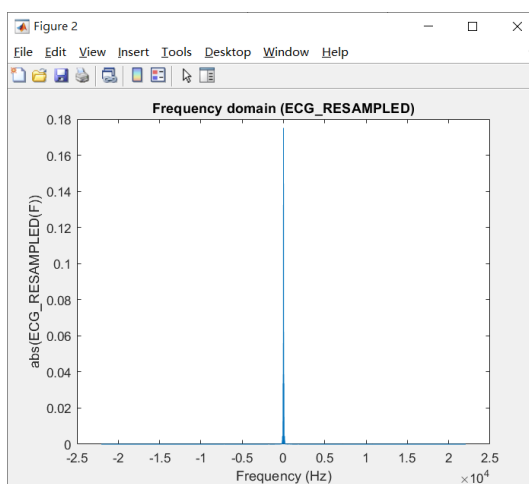


圖 3-4

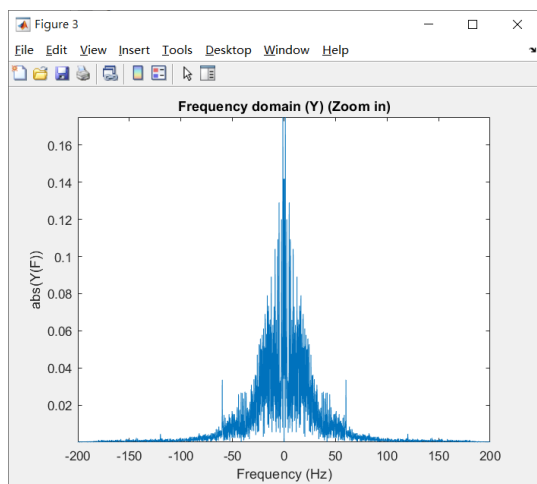


圖 3-5

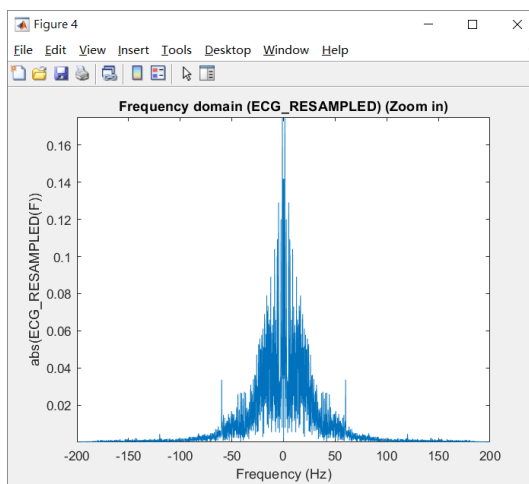


圖 3-6

- (b) 由(a)小題的圖 3-3、圖 3-5 可以看出訊號的主要頻率落在 $-100\text{Hz} \sim 100\text{Hz}$ 間，這對於我們的喇叭及耳機來說太過低頻，喇叭在此頻率區間的 frequency response 可能相當小，使我們幾乎聽不到 ECG 的聲音。因此我採取了 amplitude modulation 的方式將頻率在 frequency domain 進行位移，令原始訊號為  $s(t)$ ， $p(t) = \cos(2\pi F_{\text{shift}}t)$ ，處理後的訊號  $r(t) = s(t) * p(t)$ ，且取  $F_{\text{shift}} = 1000\text{Hz}$ ，下圖 3-7 為  $r(t)$  的 magnitude spectrum，可以看出主要頻率位於  $1000\text{Hz}$  和  $-1000\text{Hz}$  處，喇叭在此頻率的 frequency response 較大，因此我們即可以正常的音量大小聽到清楚的 ECG 音訊。

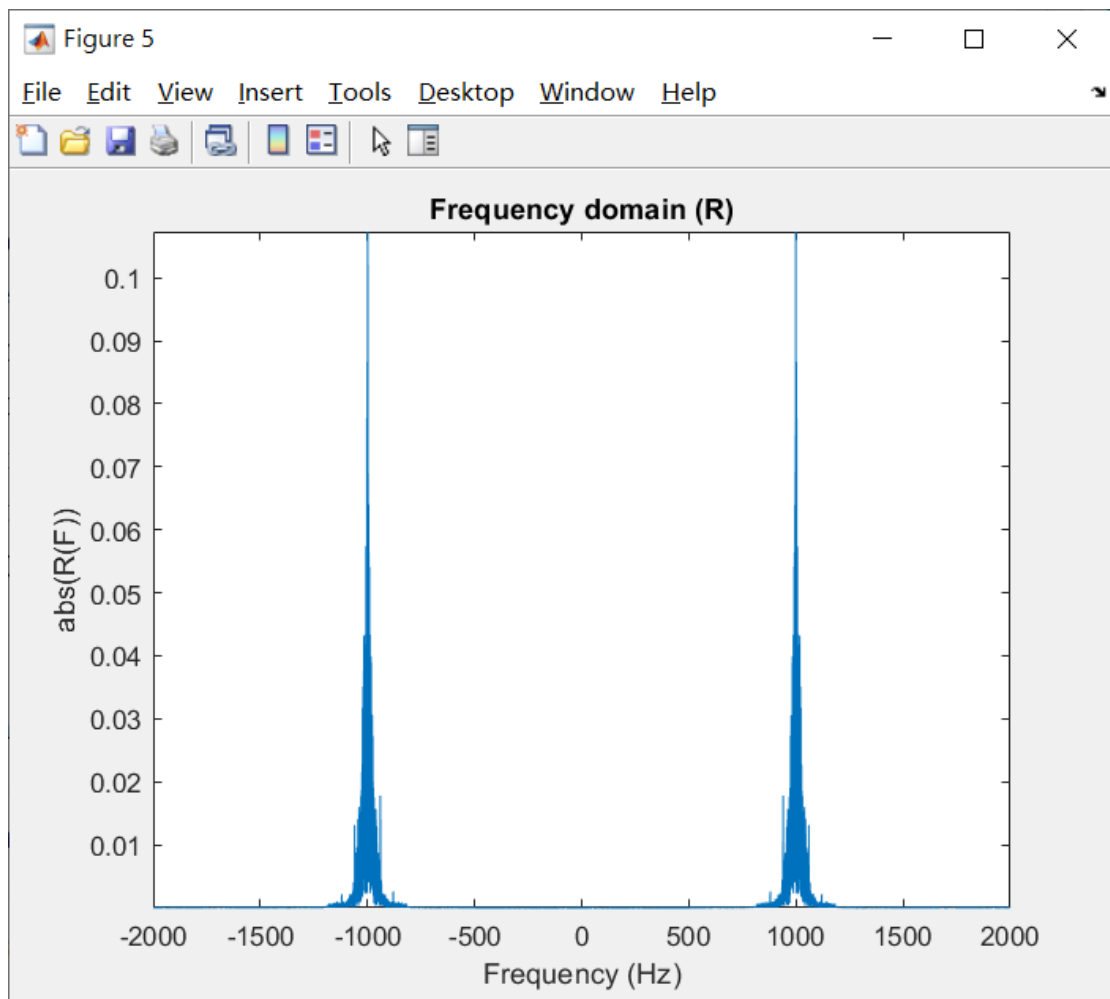


圖 3-7