

# 音波產生器

Github 網址(內有程式碼):

<https://github.com/ChristopherChen070535/Wave-generator>

撰寫人:

國立台北大學通訊工程學系三年級陳品鈞

## ➤ 專題簡介

這個程式是用來生成不同類型波形的 16 位 PCM WAV 音訊檔案以及對應的 SQNR (信號到量化雜訊比) 檔案。它支援的波形類型包括正弦波、鋸齒波、雙極方波和三角波。

程式會從命令行讀取參數，包括取樣率、量化位元數、波形類型、頻率、振幅、持續時間、輸出 WAV 檔案名稱以及 SQNR 檔案名稱。然後它會生成相應的 WAV 檔案並計算 SQNR，將結果寫入到 SQNR 檔案中。

## ➤ 成果及內容

### ● 程式輸入:

輸入之指令:

```
Wave-generator fs m wavetype f A T 1> fn.wav 2> sqnr.txt
```

Wave-generator 表示檔名，並且根據下圖 1 之參數產生波型，並計算 SQNR(Signal-to-quantization-noise ratio)

參數	意義	單位	applicable values
fs	取樣率	Hz	8000/16000/22050/44100
m	sample size	bit	8/16/32
wavetype	波形種類	無	字串: sine/sawtooth/square/triangle
f	訊號頻率	Hz	取決於 fs
A	振幅	相對於TRS端子最大輸入電壓 (Remark 1)	[0.0, 1.0]
T	產生的音訊長度	second	possible real value

圖 1:輸入之參數

### ● 程式輸出結果:

1. 輸入指令: `./Wave-generator 44100 16 sine 440 0.5 5`  
`output.wav sqnr.txt`

下圖 1 為輸入指令後所產生的 44100Khz 之 sine 波，而圖 3 則為計算後並存入文字檔之 SQNR。

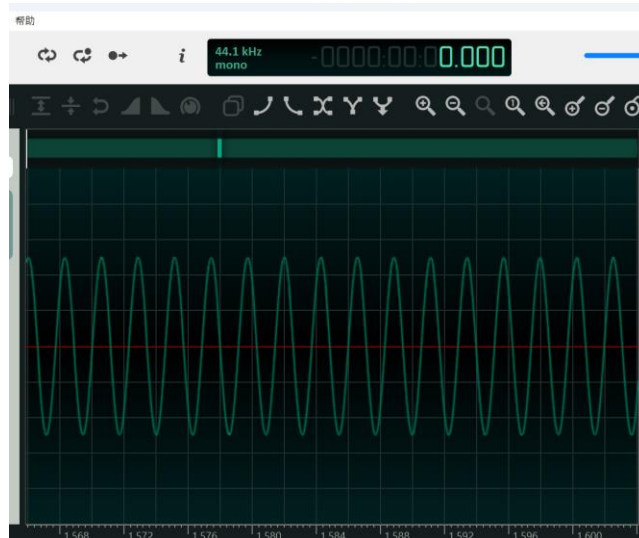
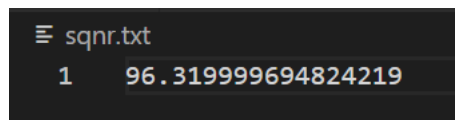


圖 2:44100Khz 之 sine 波

A screenshot of a text editor window showing the contents of a file named 'sqnr.txt'. The file contains a single line of text: '1 96.319999694824219'. The text is displayed in a monospaced font on a dark background.

```
sqnr.txt
1 96.319999694824219
```

圖 3:計算出之 SQNR

2. 輸入指令: `./Wave-generator 16000 16 sawtooth 440 0.5`  
`5 output.wav sqnr.txt`

下圖 4 為輸入指令後所產生的 16000hz 之 sawtooth 波，而圖 5 則為計算後並存入文字檔之 SQNR，因為輸入的 samplesize 和 1 相同故 SQNR 數值不變。

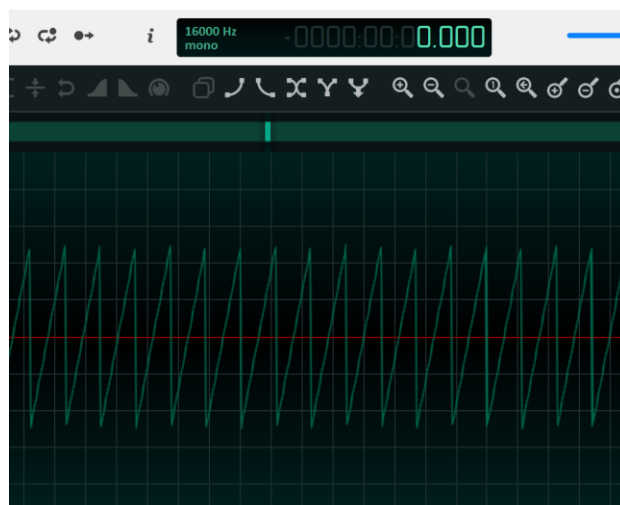


圖 4:16000HZ 的鋸齒波

```
sqr.txt
1 96.319999694824219
```

圖 5:計算出之 SQNR

3. 輸入指令: `./Wave-generator 8000 32 bipolarsquare 440 0.5 5 output.wav sqr.txt`

下圖 6 為輸入指令後所產生的 8000hz 之 bipolarsquare 波，而圖 7 則為計算後並存入文字檔之 SQNR，因為輸入的 samplesize 為 32 故 SQNR 數值改變。



圖 6:8000HZ 的 bipolarsquare 波

```
sqr.txt
1 192.639999389648440
```

圖 7:計算出之 SQNR

4. 輸入指令: `./Wave-generator 22050 8 triangle 440 0.5 5 output.wav sqr.txt`

下圖 8 為輸入指令後所產生的 22050hz 之三角波，而圖 9 則為計算後並存入文字檔之 SQNR，因為輸入的 samplesize 為 8 故 SQNR 數值改變。

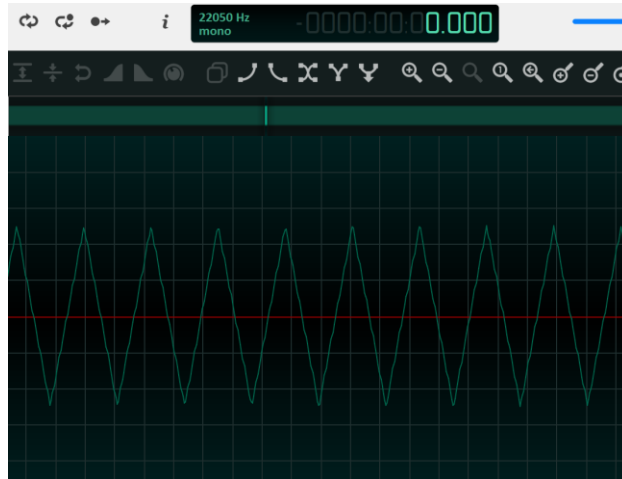


圖 8:22050HZ 的 triangle 波

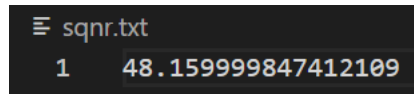


圖 9:計算出之 SQNR

➤ 程式內容:

1. Sine 波的產生:因為正弦波為連續曲線，所以根據正弦波的公式  $y(t) = A \times \sin(2\pi ft)$  將其轉換為程式可得如下圖 10，其中 SHRT\_MAX 為<limits.h>庫中的常量，表示 16 位有符號整數的最大值，主要是將結果轉換為 16 位的整數形式，作為波形的樣本。

```
// Function to generate a sine wave
int16_t generateSineWave(double t, double amplitude, double frequency) {
    return (int16_t)(amplitude * SHRT_MAX * sin(2.0 * M_PI * frequency * t));
}
```

圖 10:正弦波之程式

2. Sawtooth 波的產生:鋸齒波的形狀類似於鋸齒，這一部分的計算我主要是將頻率和時間乘積的相對位置的值映射到 -1 到 1 之間。由於鋸齒波形是一個逐漸增加再突然下降的形式，因此我將取餘數操作的結果乘以 2 再減去 1，這樣得到的值域為 -1 到 1 之間，最後再乘以振幅，即符合鋸齒波形的特性，程式之實現如下圖 11

```
// Function to generate a sawtooth wave with a specified frequency
int16_t generateSawtoothWave(double t, double amplitude, double frequency) {
    return (int16_t)(amplitude * SHRT_MAX * (2.0 * fmod(t * frequency, 1.0) - 1.0));
}
```

圖 11:鋸齒波之程式

3. Square 波的產生:方波特性為在每個週期內具有兩個固定的值，通常是正值和負值。其程式如下圖 12，首先我利用正弦波的週期

特性生成方波的輪廓，這裡的 frequency 是方波的頻率，t 是時間。接著我將正負的數值都調整為相同，亦即正值都相等，且負值都相等，這樣就確保了在每個週期內方波的取值在 1 和 -1 之間切換。

```
// Function to generate a bipolar square wave with a specified frequency
int16_t generateSquareWave(double t, double amplitude, double frequency) {
    double value = sin(2.0 * M_PI * frequency * t);
    return (int16_t)(amplitude * SHRT_MAX * (fmod(value, 1.0) >= 0.5 ? 1.0 : -1.0));
}
```

圖 12: 方波之程式

4. Triangle 波的產生: 如下圖 13，首先， $1.0 - 4.0 \times (fabs(fmod(t \times frequency, 1.0)))$  計算了時間 t 在一個週期內與中點的距離，並且將結果取絕對值，以計算出其在時域的相對位置。然後，將這個距離乘以 4，使其值域為 -2 到 2，然後減去 1，這樣得到的值域就是從 -1 到 1 之間。

```
// Function to generate a triangle wave with a specified frequency
int16_t generateTriangleWave(double t, double amplitude, double frequency) {
    return (int16_t)(amplitude * SHRT_MAX * (1.0 - 4.0 * fabs(fmod(t * frequency, 1.0) - 0.5)));
}
```

圖 13: 三角波之程式

5. SQNR 之計算: SQNR (如下圖 14) 代表了信號的功率與量化雜訊的功率之比，其單位為分貝 (dB)。在這段程式碼中，SQNR 被設置為一個固定值，其計算基於位元深度。

$$SNR(dB) = 10 \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right)$$

圖 14: SQNR 之公式

- **心得及感想:** 此次專題我學習到了很多關於音訊處理應用於程式之知識，像是如何寫 WAV 的標頭檔，還有如何將波型之概念轉換為程式等，雖然不是很難的程式，但因為先前沒有接觸過相關的概念，所以花了我一些時間完成，讓我獲益良多。