

通訊原理與實驗 LAB 04：脈波調變

1. 實驗目的：MATLAB 程序實現脈波調變與其系統設計。

2. 實驗內容：

Alec Reeves 發明脈波編碼調變 pulse code modulation (PCM)，一種類比訊號的數位化方法。PCM 將訊號的強度依照同樣的間距分成數段，然後用獨特的數位記號（通常是二進位）來量化。它是數位音頻，CD，電話和其他數位音頻應用的標準形式[1]。

在圖 4-1 中，一個正弦波（紅色曲線）被取樣和量化為 PCM。正弦波在每段固定時間內被取一次樣，即 x 軸的刻度。而每一個樣本則依照某種運演算法（在這個例子中是 ceiling function），選定它們在 y 軸上的位置。這樣便產生完全離散的輸入訊號的替代物，很容易編碼成為數位資料，以作儲存或操縱。以右圖為例，很清楚看出樣本為 8、9、11、12、13、14、15、15、15、14...等，將它們以二進位編碼，就得到一組一組的數字：1000、1001、1011、1100、1101、1110、1111、1111、1111、1110...等。

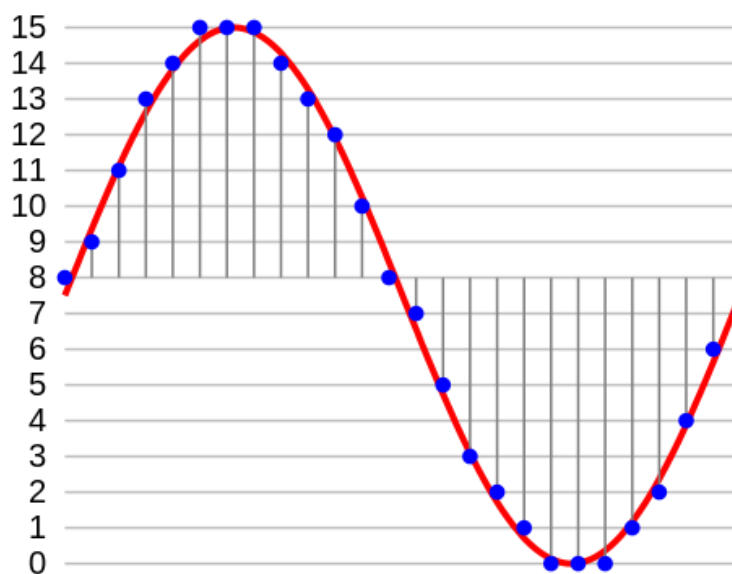


圖 4-1 PCM 正弦波（取自維基百科）

PCM 資料流(stream)是模擬信號的數位表示，其中模擬信號的幅度以均勻間隔並彩定期採樣，每個樣本在數位步長的範圍內被量化為最接近的值。PCM 資料流具有確定流對原始模擬信號的保真度的兩個基本屬性：**採樣速率**（即採樣採樣的每秒的次數）以及**位深度**（表示每個樣本的可能數位值的數量）。

```

1 % Lab4_pcm
2 - clc
3 - close all
4 - clear all
5 - N = 64;
6 - fs = 5; %採樣率為niquist速率 >= 2*訊號頻寬
7 - ts = (0:N-1)/fs; %採樣時間區間
8 - freqStep = fs/N; % 頻域的頻率的解析度
9
10 - c =input('編碼位元 = '); %每個PCM值由 c 個位元表示
11 - level = 2^c ; %共計有 2^c 階層
12 - part = -1: 2/level:1;% 取樣器的最低值-1.0 與最高值 1.0, 間隔數量為 2^c.
13 - codebook = -1.0:2/level:1+2/level;%量化器代碼簿, 量化後的數值-1,-1+1/level,-1+2/level., ...
14 - fc = 1; %產生弦波,fc = 1Hz
15 - t = (0:N-1)/(10*fc);
16 - x = cos(2*pi*fc*t); %產生原訊號
17 - ts = (0:N-1)/fs; %採樣時間區間
18 - xn = cos(2*pi*fc*ts); %取樣後訊號,
19 - [index,quants] = quantiz(xn,part,codebook);% quants 量化向量, 值為浮點數
20 - % index 階層向量, 值為整數
21 - subplot(6,1,1);
22 - plot(t,x);
23 - xlim([0, 5]); %改變x軸上顯示之範圍
24 - title('原訊號 x(t)');
25 - X = fftshift(fft(x));
26 - subplot(6,1,2);
27 - M = length(x);
28 - freq = (10*fc/M)*(-M/2:M/2-1); % 頻域的頻率刻度
29 - plot(freq,abs(X)/M);
30 - title('原訊號的FFT X(f)');
31
32 - subplot(6,1,3);
33 - stem(ts,xn);
34 - xlim([0, 5]); %改變x軸上顯示之範圍
35 - title('取樣後訊號 x(nt)');
36 - X1 =fftshift(fft(xn));
37 - freq1 = freqStep*(-N/2:N/2-1); % 頻域的頻率刻度
38 - subplot(6,1,4);
39 - plot(freq1,abs(X1)/N);

```

圖 4-2-1 pcm 程式

```

40 - title('取樣後訊號的FFT  $X_n(f)$ ');
41
42 - subplot(6,1,5);
43 - stem(ts,quants);
44 - xlim([0, 5]); %改變x軸上顯示之範圍
45 - title('量化後訊號  $v(nt)$ ');
46 - X2 = fftshift(fft(quants));
47 - subplot(6,1,6);
48 - plot(freq1,abs(X2)/N);
49 - title('量化後訊號的FFT  $V_n(f)$ ');
50

```

圖 4-2-2 pcm 程式

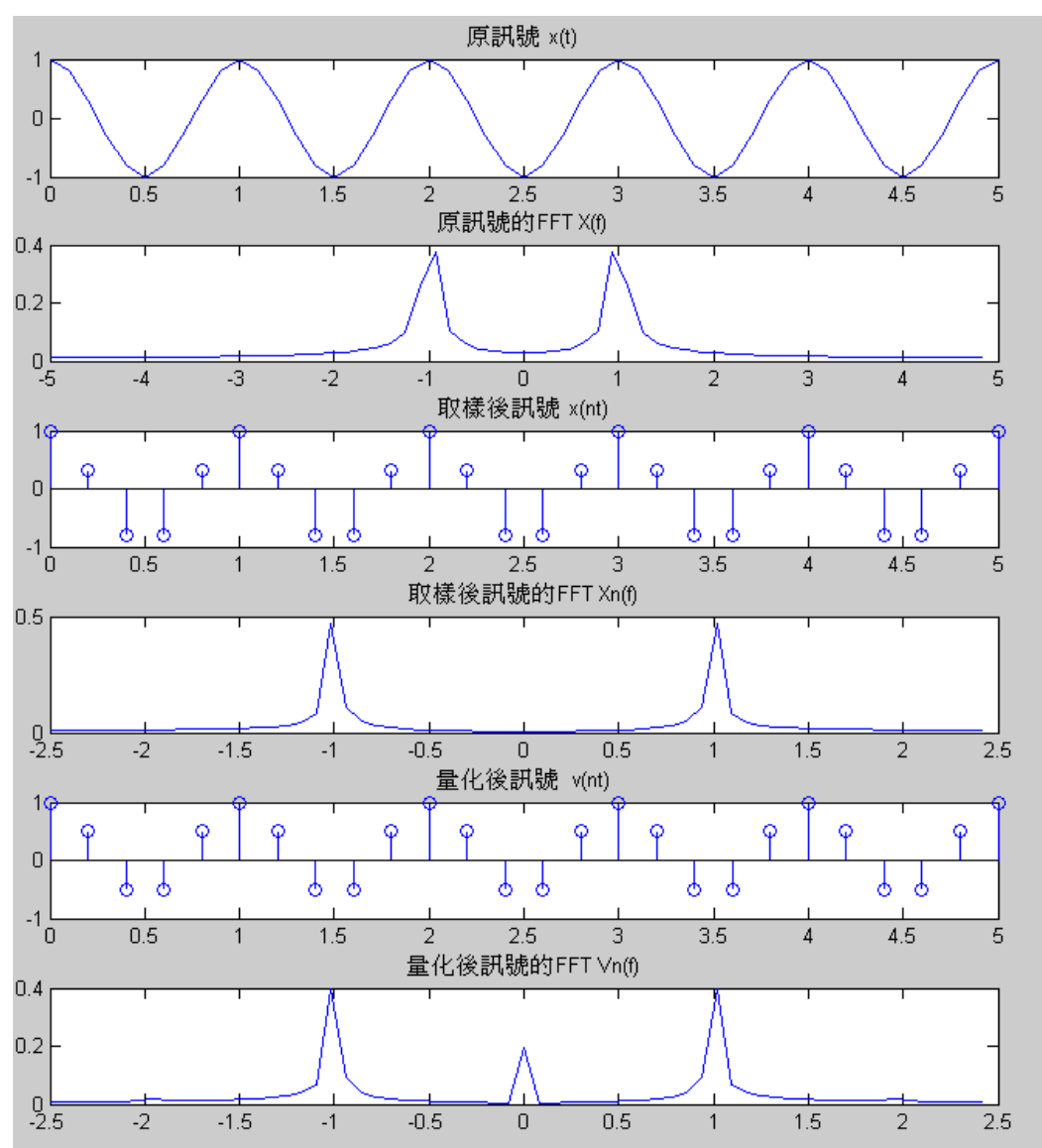


圖 4-3 圖形 (編碼位元 = 2)

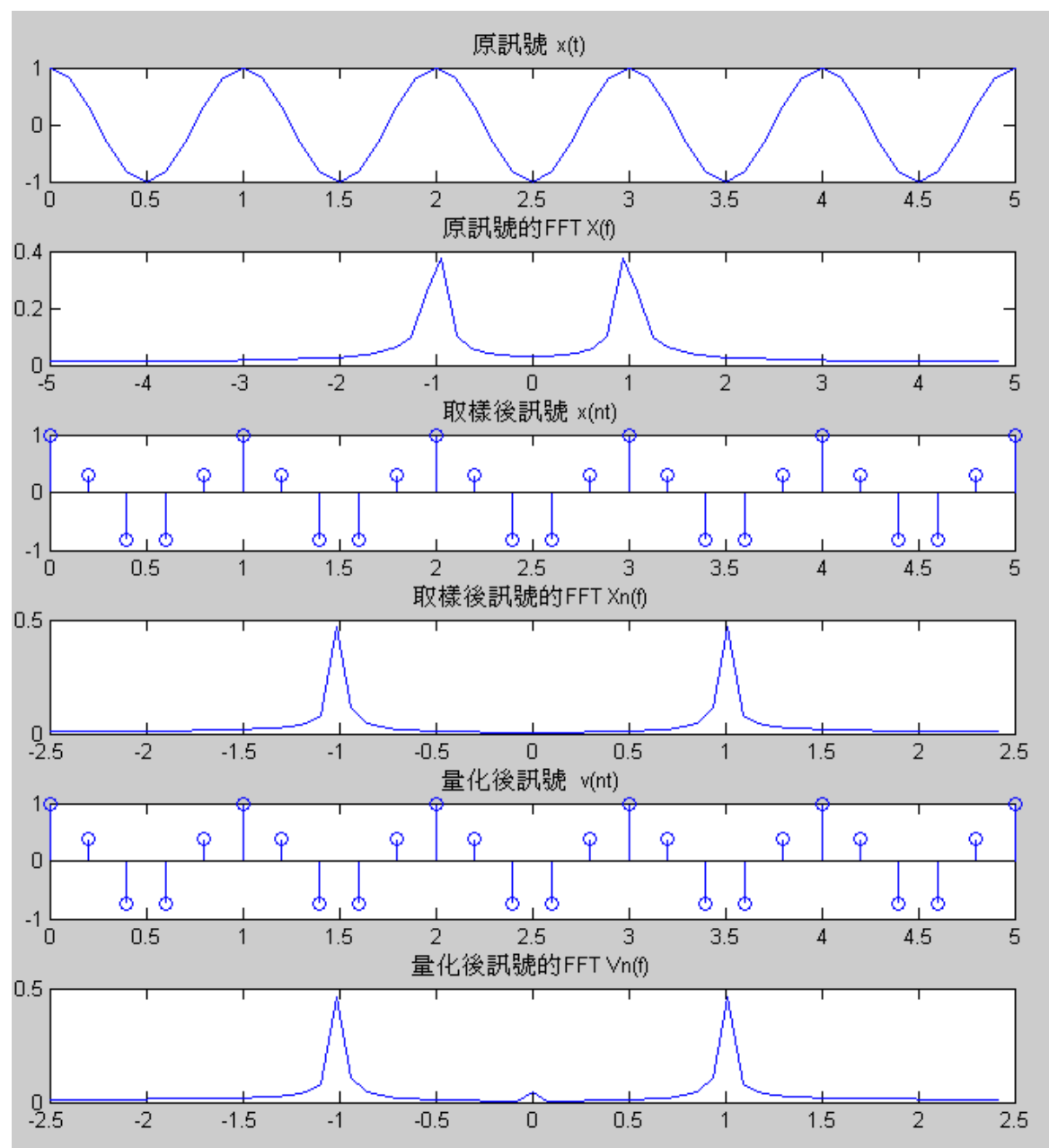


圖 4-4 圖形 (編碼位元 = 4)

指令學習

`[index, quants] = quantiz(sig, partition, codebook)` : 取樣器

sig : 訊號向量

partition : 階層分割線數量 = 階層數量-1, 如8階層則需7條階層分割線。

codebook : 階層所對應的數值, 如`[-1, -0.5, 0, 0.5]` 分別表示為4階層對應之數值。

Index : 訊號向量取樣後的階層向量。

quants : 訊號向量取樣後的階層對應之數值向量。例如 : 5

```
[index, quants] = quantiz(sig, partition, codebook)
```

```
sig = [13, 34, 84, 40, 23]
```

index = 10:10:90 ; 計有9條階層分割線，最低階為10，每隔10劃一條，直至90為止。

codebook = 10:10:100，計有10個階層所對應之數值 10 ~ 100。

```
index = [0, 3, 8, 3, 2]
```

```
quants = [10, 40, 90, 40, 30]
```

3. 作業

(1) 請依圖4-2-1 及4-2-2 程式，將弦波改為方波，頻率為2Hz，取樣頻率大於Nyquist Rate，並產生如圖4-3 (編碼位元 = 2) 及圖 4-4 (編碼位元 = 4) 等圖形，相關程式請參考以下附錄(或檔案t1.m)。

(2) **反思題**: 比較圖 4-3 圖形 (編碼位元 = 2) 及圖 4-4 圖形 (編碼位元 = 4)，為何圖 4-4 量化後訊號的FFT $V_n(f)$ 在 $f=0$ 附近處的數值小於圖 4-3，請說明原因? (當編碼位元 = 6， $f=0$ 附近處的數值更小)

(3) **反思題**: 圖 4-3 圖形 (編碼位元 = 2) 及圖 4-4 圖形 (編碼位元 = 4) 最後都將產生量化後訊號 $v(nt)$ ，請問二者 $v(nt)$ 的速率為多少？請寫出計算式。

(4) **挑戰題**: 請自行設計PCM之matlab程式，使其產生頻譜圖形足以說明AliasEffect(課本第310頁)。

附錄

● 以下為檔案 t1.m 之內容

```
1 % Lab4_hw#1
2 clc
3 close all
4 clear all
5 N = 1280;
6 fs = 5; %採樣率為niquist速率 >= 2*訊號頻寬
7 ts = (0:N-1)/fs; %採樣時間區間
8 freqStep = fs/N; % 頻域的頻率的解析度
9
10 c = input('編碼位元 = '); %每個PCM值由 c 個位元表示
11 level = 2^c; %共計有 2^c 階層
12 part = -1: 2/level:1;% 取樣器的最低值-1.0 與最高值 1.0, 間隔數量為 2^c.
13 codebook = -1.0:2/level:1+2/level;%量化器代碼簿, 量化後的數值-1,-1+2/level,-1+4/level., ...
14 fc = 2; %產生弦波,fc = 1Hz
15 t = (0:N-1)/(100*fc);
16 x = square(2*pi*fc*t); %產生原訊號
17 ts = (0:N-1)/fs; %採樣時間區間
18 xn = square(2*pi*fc*ts); %取樣後訊號,
19 [index,quants] = quantiz(xn,part,codebook);% quants 量化向量, 值為浮點數
20 % index 階層向量, 值為整數
21 plot(t,x);
22 xlim([0, 5]); %改變x軸上顯示之範圍
23 ylim([-1.1, 1.1]); %改變x軸上顯示之範圍
24 title('原訊號 x(t)');
```

