DSP Lab 1

DSP Lab 1: Introduction to Arduino Platform

電機 19 紀伯翰 104061171

1a. Specification:

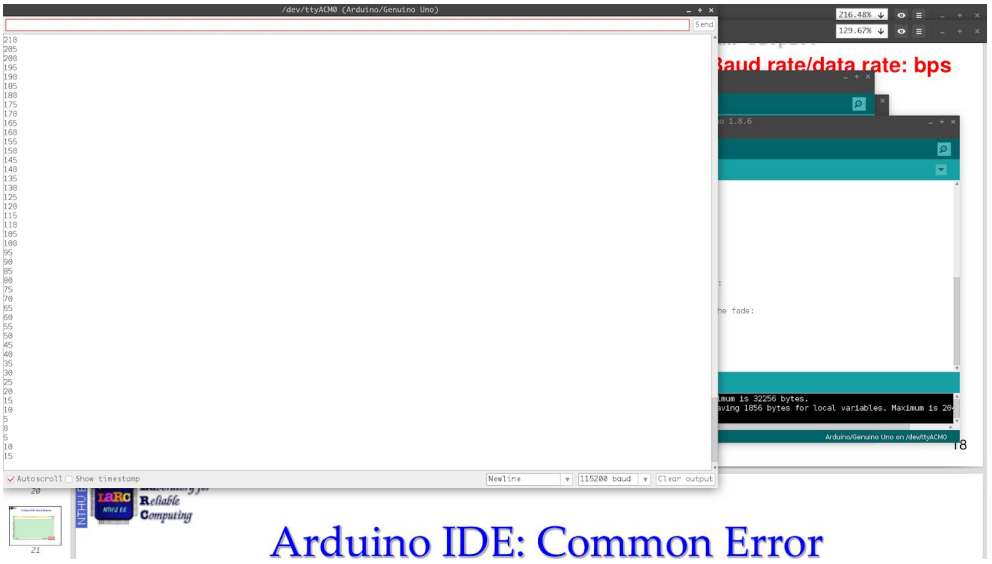
See the Arduino training manual and finish the labs till class 7。

1b. Implementing:

Fade:

https://drive.google.com/file/d/1pPkFRMwhFW6MvT8_04OWgwPVe0ydh8Vc/view?usp=sharing

Serial:



2a.Specification:

(Sampling and Aliasing) Connect the function generator (Picoscope, see PICOSCOPE.pdf)
to the ADC of the Arduino board. Set sine wave as the ADC input waveform with frequency
of 100 Hz. Use the following frequencies (500 Hz, 200 Hz, 100 Hz, 80 Hz) to sample the input waveform.

1) Plot the waveforms with the provided Matlab codes.

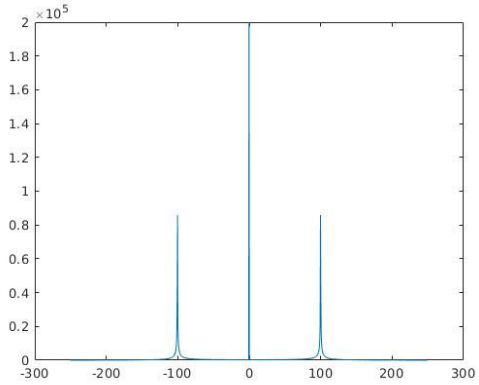
2) Use FFT to show the spectra of the saved samples.

3) Discuss about the aliasing issues.

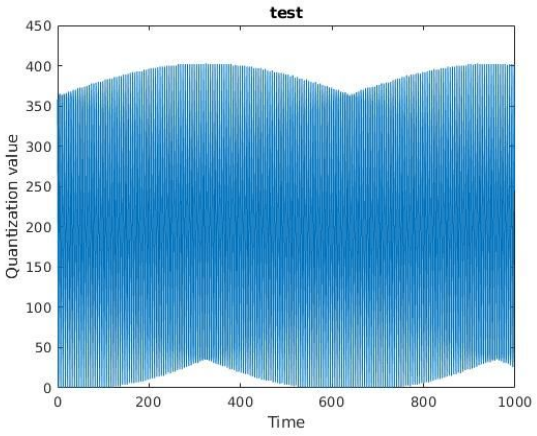
b.implementing:

500hz

Frequency domain

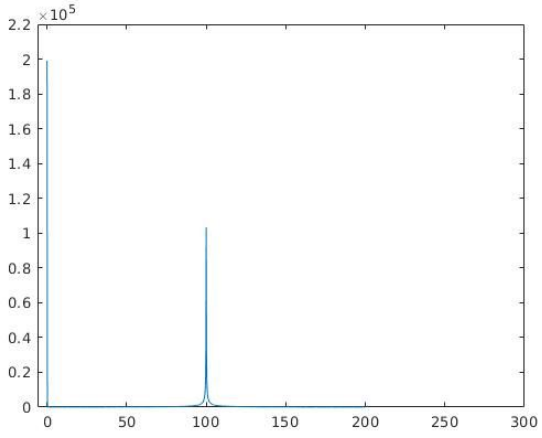


Time domain

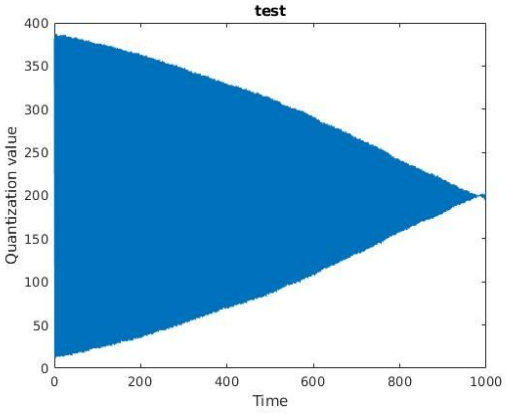


200hz

Frequency domain

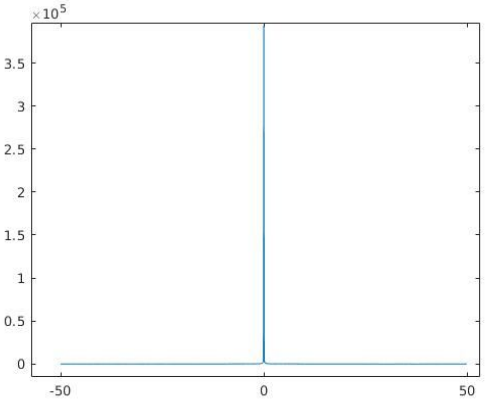


Time domain

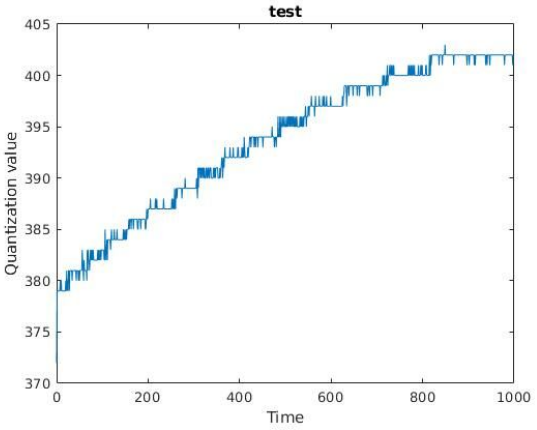


100hz

Frequency domain

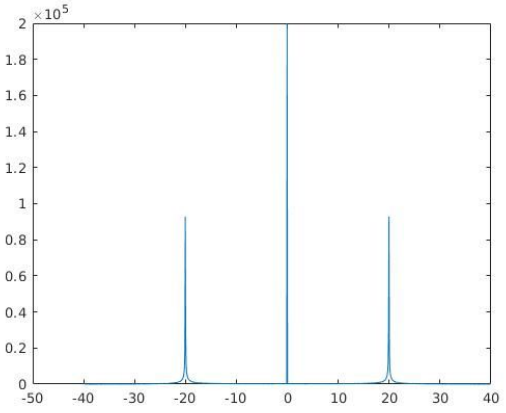


Time domain

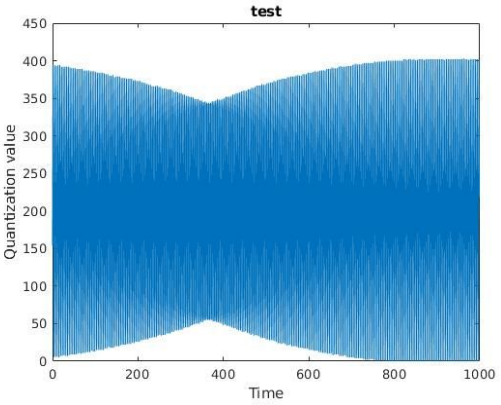


80hz

Frequency domain



Time domain



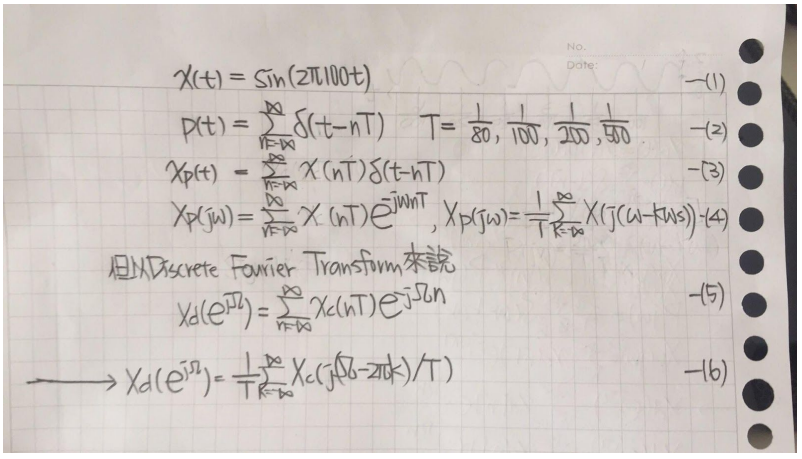
2c. Discussion:

在frequency domain聚集在frequency 0的位置的impulse signal是offset 1V 造成的現象,
Quantization value 都介於0-400的原因是因為arduino板子支援0-5v,但picoscope的工作電壓在0-2v所以說Quantization value頂多位於400上下,即使10bit也就是有1024個states下,也quatization的值頂多出現在400上下.

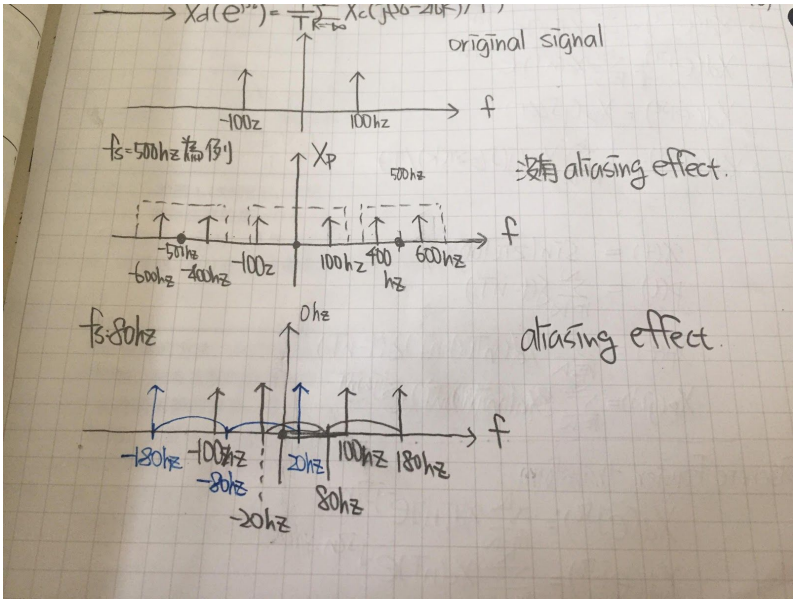
Aliasing issue:

四張圖show出了膺頻效應,
當取樣頻率小於輸入訊號頻率的兩倍,output的訊號在時域上會distortion.
所謂的distortion是指時域上output的訊號除了有放大縮小或是shift的效果之外,還有其他的效果造成輸出的訊號奇形怪狀也就是所謂的失真,而在頻域上會有訊號被抵銷甚至是疊加的狀況,以80hz取樣頻率的圖為例,照理來說,input為100hz的sine訊號,他在頻譜的分佈應是在100hz和-100hz上會各有一根delta signal.但因為取樣的時候頻域的區間比sine函數區間還窄導致在20hz和-20hz也出現了delta signal,並且時域的訊號distortion了.
取樣頻率100nz的圖之所以沒有delta function是因為那兩根會在f的[-100,100]上
200hz的取樣頻率為Nyquist frequency 的臨界值,所以可以看到在它frequency domain的100hz, -100hz上有signal.但因為其實analog的sine訊號並無法精確的作到100hz會有點誤差,所以200hz的取樣頻率會使它在frequency domain之f介於[-100,100]上和500hz的圖相同.

以下是訊號取樣的數學式子過程:



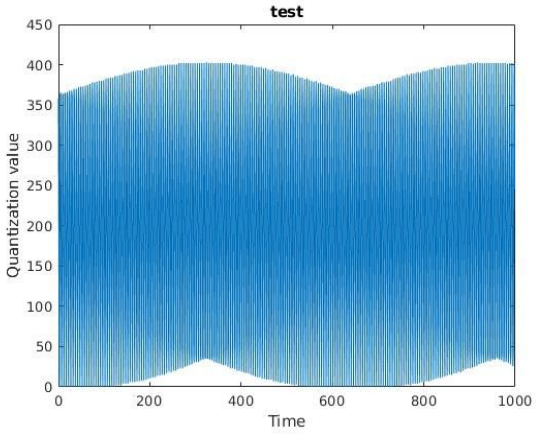
已知圖中 (1)為input signal, (2)為sample function
藉由上圖(4)式可以知道說output signal在頻域上是原本的訊號疊加上shift n倍的fs的訊號,
如圖



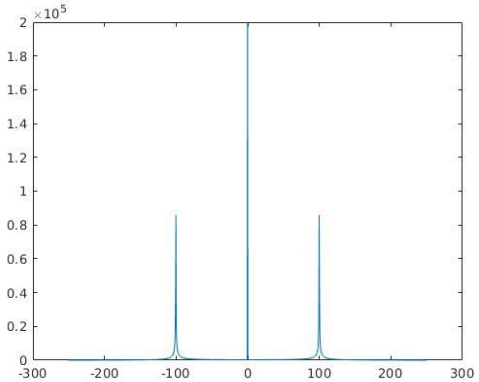
所以80hz的sample rate 才會在頻譜的-20,20hz上會有兩根delta function(先不管正負)

3a.Specification:
(Quantization) Use 10 bits, 8 bits, 5 bits, and 3 bits to sample the oscillator waveform.
1) Plot the waveforms with the provided Matlab codes.
2) Use FFT to show the spectra of the saved samples.

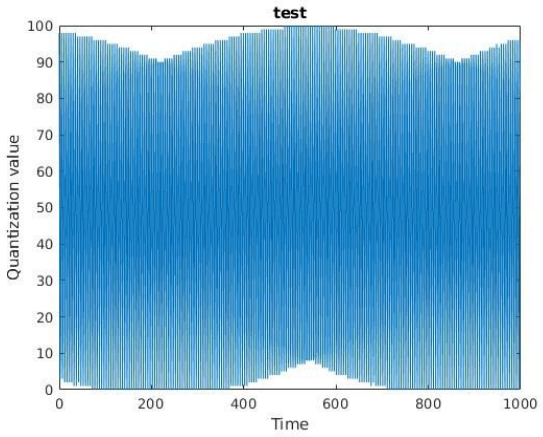
3b.implementing:
10bit
fs = 500
time domain

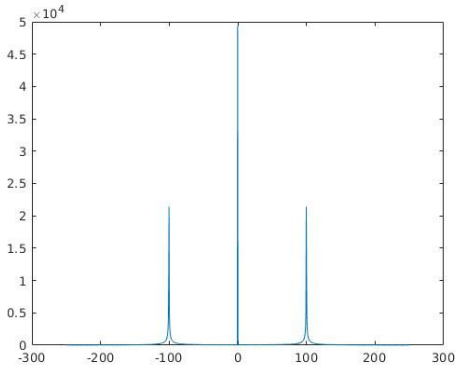


Frequency domain

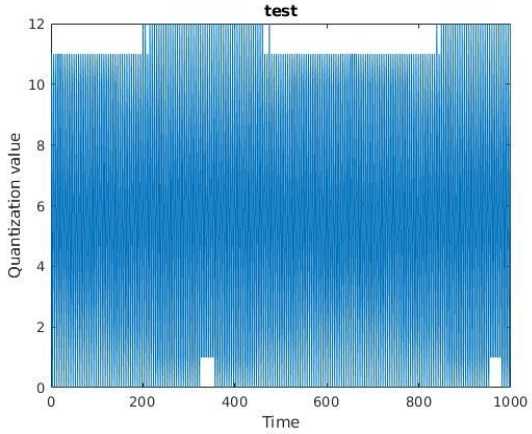


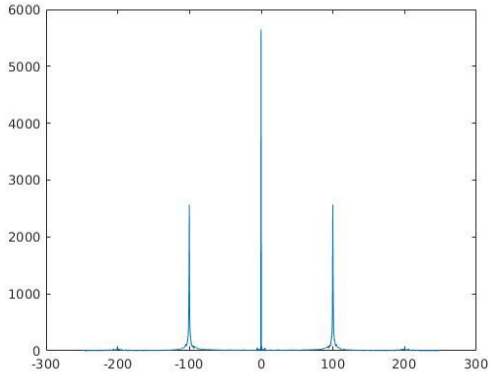
8bit





5bit

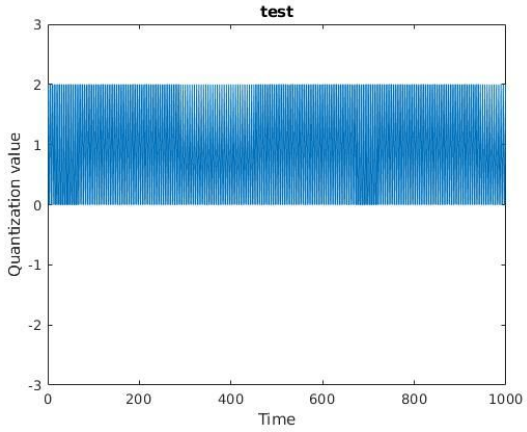


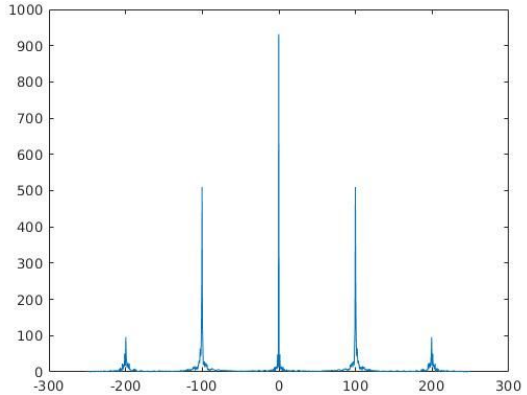


3bit

TIme domain:

Frequency domain:





3c.Discussion:

不同的quantization會使time domain上的訊號變得方方角角的,而在frequency domain上每一個delta 訊號更是產生了毛邊,以及多了雜訊的干擾.至於會出現邊邊角角的原因在於quatization 的 每一個interval太廣了,變成說當訊號假如高於這個區間的上界一點點它就歸屬於下一個區間,使得時域的訊號變成邊邊角角的.

Extra Question:

1a. Specification:

Why does the sampled sine signal (fs = 500 Hz or 200 Hz) look like an amplitude-modulated (AM) sine wave (i.e., 100 Hz sine carrier with amplitude modulated envelope)? not like an expected sine wave?

1c Discussion

以頻譜來說,取樣的頻譜並不是和sine產生的頻譜相同,假設fs =500 那sine的兩根delta signal會以500hz的週期一直在頻譜上出現和sine頻譜上只有兩根的圖是不一樣的,它們只是在頻譜中f介於[-100,100]之間相同而已,

也因為如此當頻譜inverse fourier transform回來之後,會出現訊號上有延遲像是AM的modulation出現.

為何是AM Modulation?

首先以下是AM Modulation示意圖:

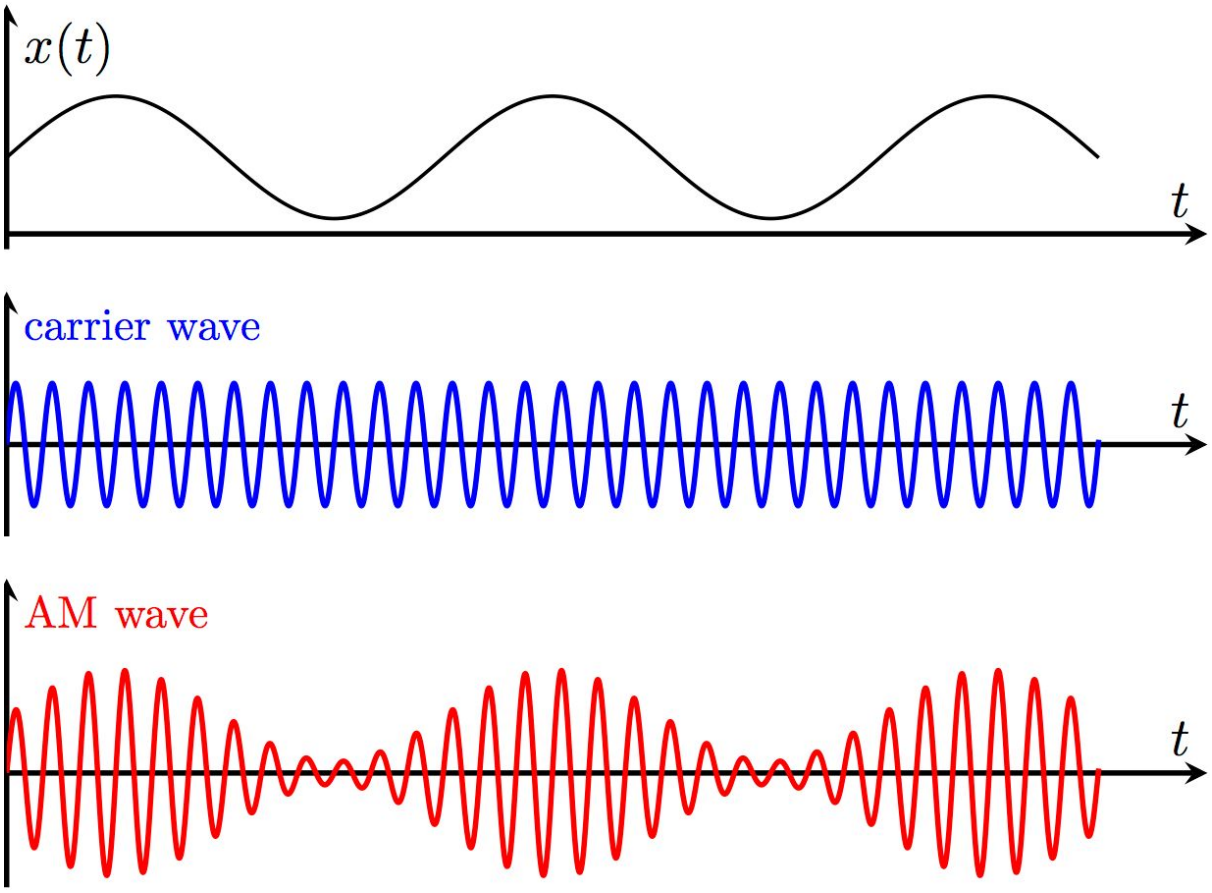


Figure 1.

先把狀況分成以下兩種:

(1)sample rate > input frequency

(2)sample rate< input frequency

當我們使用fs = 500,200的時候,sample rate > input frequency,所以以上圖中x(t)的圖來看它會在input signal一個週期內取到至少一個點以上,那當我們蒐集完全部的點時,在用matlab作圖時,matlab做的工作只是把那些點的數值全部連線起來,那以500hz來說,它會在input 的一個週期內取到至多5個點,然後每個週期都會取5個點,然後有規律的按照100hz的input signal的重複取樣5個點,產生了一個100hz的envelope.而振幅因為是取樣所以會隨著100hz的input signal有變化,就如同是將fs當成是一個振幅是1的cosine或sine載波頻率為500hz,200hz,然後input 是頻率為100hz的sine signal,做的一個AM調變.

還有一個原因可以從頻譜來看,當我們做取樣的時候,如下圖 (沒有aliasing effect)

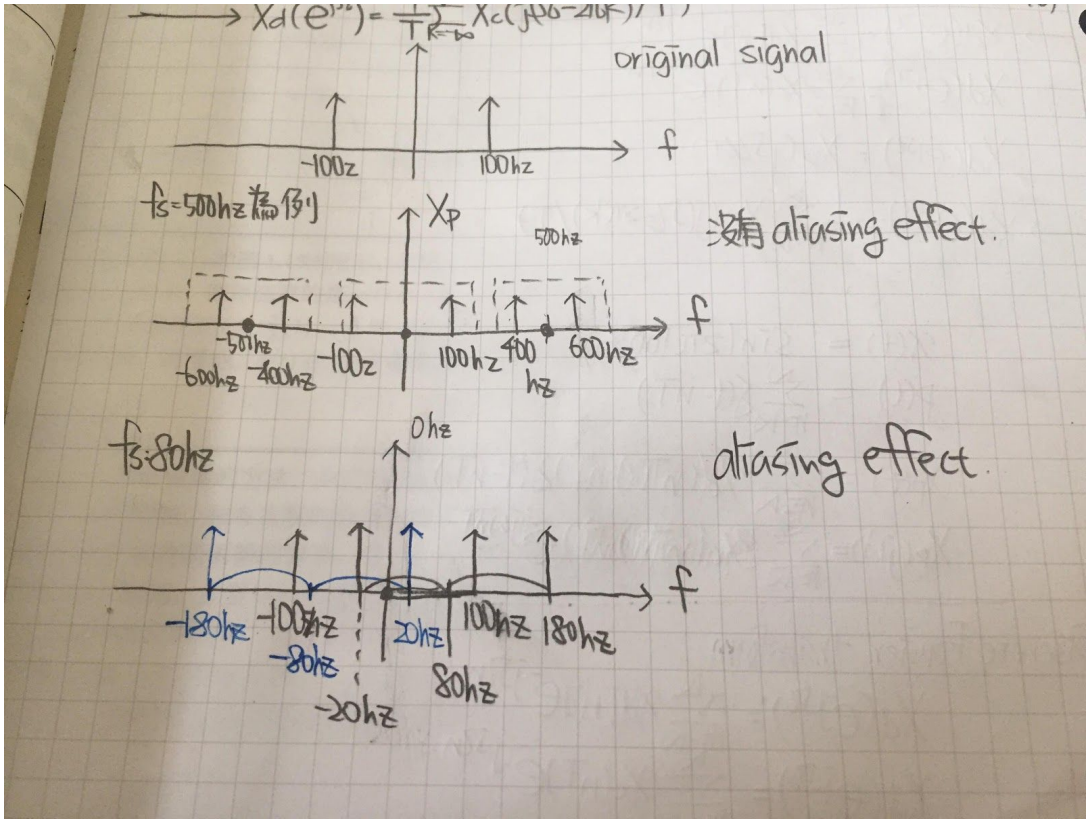
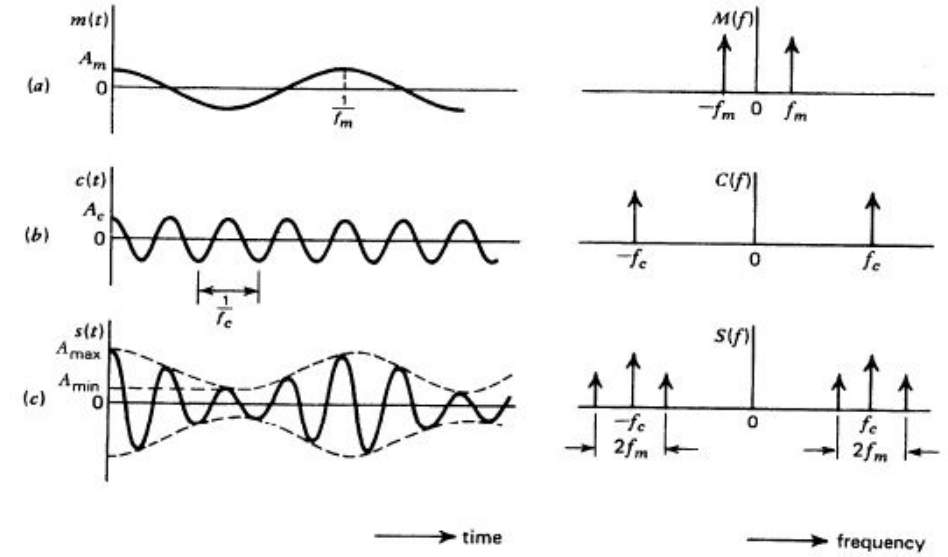


Figure 2.



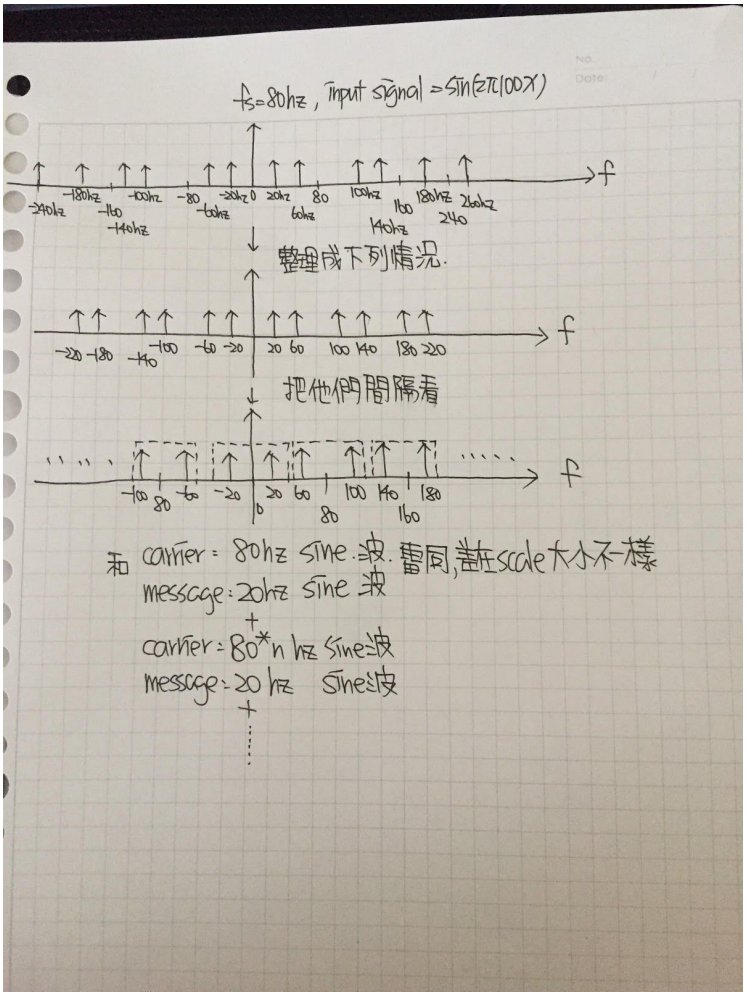
Illustrating the time-domain (on the left) and frequency-domain (on the right) characteristics of standard amplitude modulation produced by a single tone. (a) Modulating wave. (b) Carrier wave. (c) AM wave.

Figure 3

我們可以看到他的頻譜是每500hz會在兩端各距離100hz的地方產生sine訊號的delta function(先不管正負).當我們再做AM 調變的時候,我們是把兩個頻率小的delta訊號(input以sine,cosine為例)藉由載波(頻率比較大的訊號,carrier)送到高頻去,可是頻譜的高度會因為載波為cosine而乘上1/2,但初略來說頻譜分佈(撇除低頻來看)是和取樣之後的頻譜圖是雷同的,也是把低頻的東西藉由載波送到高頻去,不同的是AM並無低頻訊號以及因為取樣在frequency domain會與原訊號做convolution,所以取樣的頻譜會以

週期為500hz重複出現兩根delta function(input以sine為例),因此取樣的頻譜還比AM多了很多週期的delta function.可以把那些很多週期的delta function理解成載波頻率為n*500hz的訊號和input signal進行疊加.

(2)sample rate< input frequency
當sample rate 小於input frequency,情況變得有點特殊,以頻譜來看,我們知道它會aliasing,
如圖:



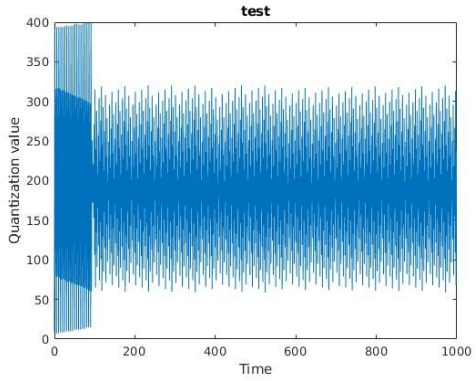
變成是以message 為 100-80=20hz的signal,配上carrier 為 80*n hz的signal做疊加,才形成output signal.

2a.Specification:

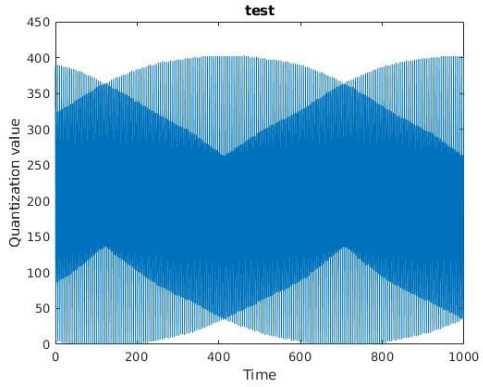
In the experiment, you're using the highest baud rate (i.e., 115200 bps) of the serial port.
What would happen if you set a smaller baud rate, e.g., 9600 bps or 19200 bps?
2a-1
基於Arduino上的程式碼 (如下), 擷取波形顯示在PC的螢幕上包含二個步驟 (1) DATA ACQUISITION (2) DATA TRANSFER
DATA ACQUISITION at sampling rate, e.g., M Hz, and assume each data is represented by "int": 2 Bytes => DATA Acquisition at M*2*8 bps (bit per second)
DATA TRANSFER at Baud rate, e.g., N bps
想一想
Can DATA TRANSFER rate smaller than DATA ACQUISITION rate? e.g., can N < M*2*8 bps?

baud rate:是指硬體上的傳輸速率,實驗中,arduino透過port和電腦作到資料傳輸,而傳輸的速率事實上是有協定的,市面上以及網路查到的是RS232傳輸協定還有很多種協定,而傳輸速率就以baud rate來命名.
baud rate 又稱 鮑率,假設baud rate = 9600的話,代表傳輸速率為每秒最多可以傳9600個bit,在電腦和arduino間進行數據傳輸,實驗中,當我以9600的baud rate進行傳輸的時候,在取樣頻率為250hz的時候,time domain會出現資料點變少的問題

2b.implementing
Baud rate = 9600 bit/s
Sample rate = 250hz



但當我將baud rate調成19200 bit/s時:



圖又恢復正常了.

2c.Discussion
會出現這樣的原因在於 每秒取樣的資料量多於每秒最多可以傳輸的資料量.(DATA ACQUISITION rate > DATA TRANSFER at baud rate)
透過arduino內部函式,可以知道Serial.println()到底上傳了多少byte的資料量到Serial上
假設quantization的值為 295,每一個數字各代表一個char所以是各一個byte,因為println會自動加上'\r' 和 '\n',所以以295這個值為例就是5個bytes,假設每次取樣的值都是3個位數,而取樣頻率為250hz,
250 x 5 x 8 = 10000 bits,也就是說每秒會傳輸10000個bit遠遠大於9600bit,造成圖形上出現怪異的狀況.
而baud rate設 19200 bit/s 則會高於取樣頻率,則訊號恢復成正常樣貌.

4. Conclusion

這個lab學了不少東西,包括使用Serial 傳輸 data的細節,baud rate影響資料傳輸,去細查matlab以及arduino的document相關的函式來查看println究竟上傳了多少資料量到Serial中,以及手動實做有關aliasing還有sample,ADC的問題,實做時,才會去徹底的研究函式到底做了哪些事情,以及signal的形狀造成的原因會去做一個探討,在實做上會發現,現實世界會出現很多誤差,造成理論值出現了些許的錯誤並且需要修正,不過找到原因解決其實蠻好玩的.

5.Reference

-Arduino 線上document
-講義上的教學還有提供的 code
-matlab 線上document
-通訊系統一

MATLAB CODE

```
clear all;
fclose('all');

serialobj=instrfind;
if ~isempty(serialobj)
   delete(serialobj)
end
clc;clear all;close all;
s1 = serial('/dev/ttyACM0');  %define serial port
s1.BaudRate=9600;     %define baud rate

disbuff=nan(1,1000);

fopen(s1);
clear data;
N_point = 1000;
fs=220;   %sample rate
time=[0:1:999];
figure
h_plot=plot(nan,nan);
hold off
tic
for i= 1:N_point
   data=fscanf(s1);%read sensor
   %whos data;
   %disp(data);
   y(i) = str2double(data);
   if y(i) > 360
   disp(y(i));
   end
   if i<=1000
```

```
    disbuff(i)=y(i);

    else
    disbuff=[disbuff(2:end) y(i)];
    end

    if i>1
    set(h_plot,'xdata',time,'ydata',disbuff)
    title('test');
    xlabel('Time');
    ylabel('Quantization value');
    drawnow;
    figure(1);
    end

end
final=(abs(fft(disbuff)));
%final = fftshift(abs(fft(disbuff))); %shift to -100,+100
figure(2)

plot(((0:999)*fs/1000),final)
%plot(((0:999)-(1000+1)/2)*fs/1000,final) %plot signals which shift to -100,+100
xlim([-6,300]) %To show 0 value apparently
toc
% close the serial port
fclose(s1);
```

Arduino code

```
int pot = A0;
int val = 0;
int start_time;
int now_time;


void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(pot,INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:

  start_time = micros();
```

```
  val = analogRead(pot);    //read analog input

  //val = map(val,0,1023,0,255);    //mapping
  //Serial.println(sizeof(val));
  //Serial.println(val,BIN);
  Serial.println(val,DEC);
  //Serial.println(size_t(Serial.print(val,DEC)),DEC);     //print the size used

  now_time = micros();

  while(now_time-start_time<4545.45){    //sample rate  //fs = 220 tiime = 1/220
    now_time = micros();
    //Serial.println(now_time,DEC);
  }

}
```
