**USMAN MARUF**

**3041120**

**COMPUTER ENGINEERING**

**4.1**

**a.** function [xx, tt] = beat(A, B, fc, delf, fsamp, dur)

%BEAT compute samples of the sum of two cosine waves

% usage:

% [xx, tt] = beat(A, B, fc, delf, fsamp, dur)

%

% A = amplitude of lower frequency cosine

% B = amplitude of higher frequency cosine

% fc = center frequency

% delf = frequency difference

% fsamp = sampling rate

% dur = total time duration in seconds

% xx = output vector of samples

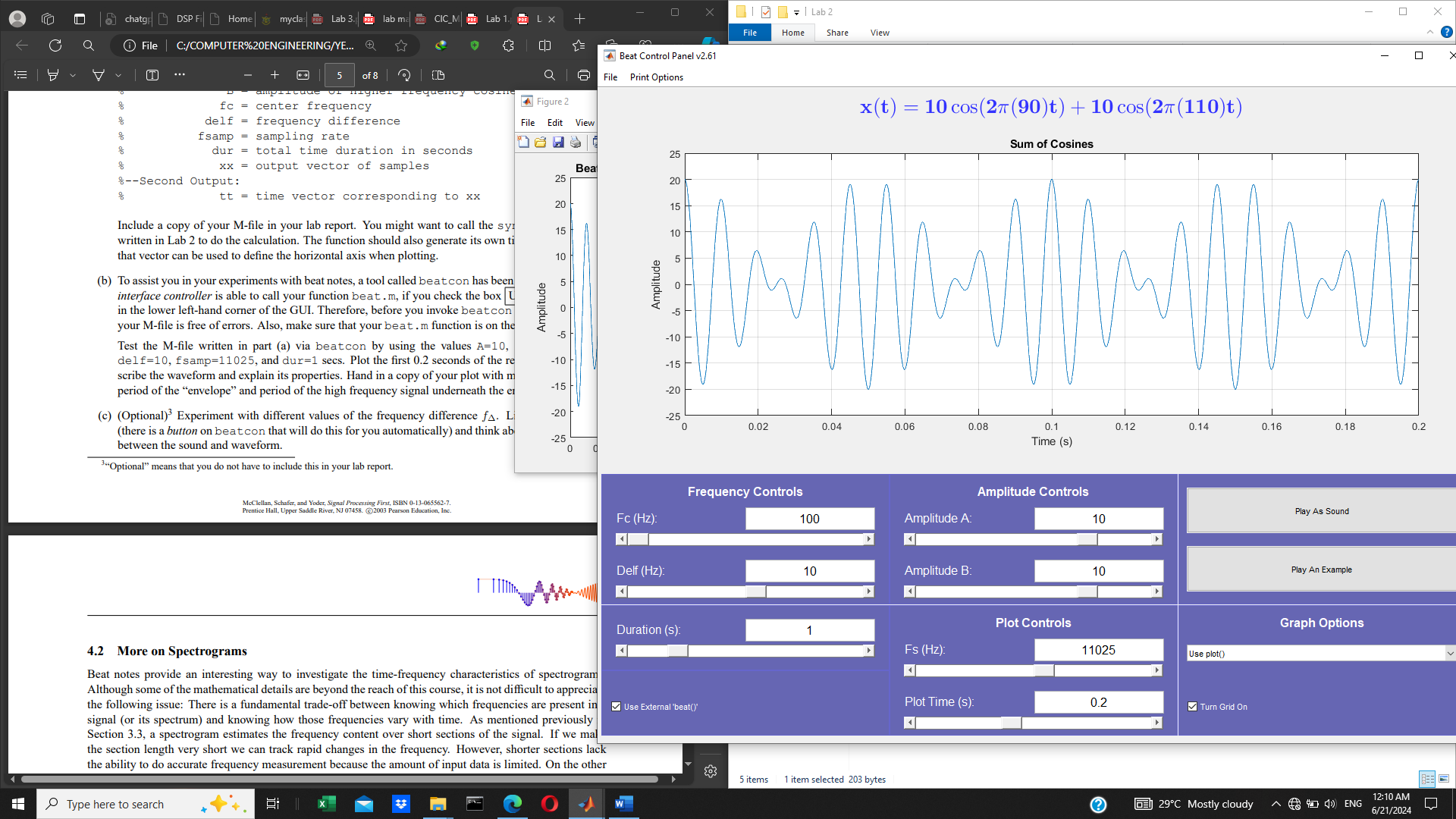
%--Second Output:

% tt = time vector corresponding to xx

tt=0:1/fsamp:dur;

xx=A\*cos(2\*pi\*(fc-delf)\*tt) + B\*cos(2\*pi\*(fc+delf)\*tt);

**b.**

****

**4.2**

**a.** delf=32; %Hz

dur = 0.2; %s

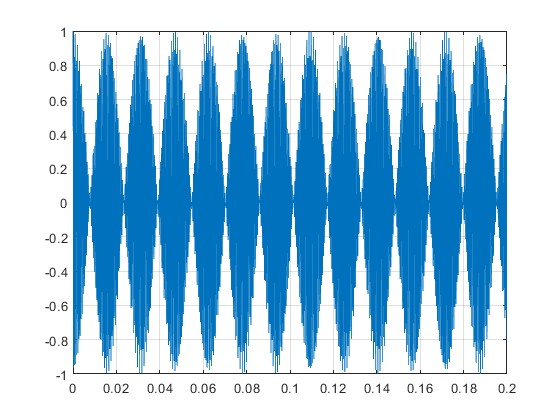
fsamp = 11025; %Hz

fc = 2000; %Hz

tt=0:1/fsamp:dur;

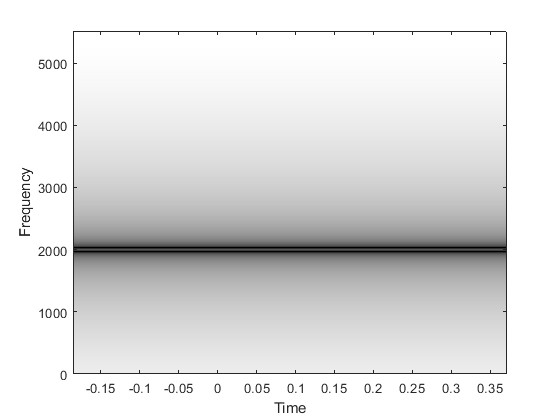
xx=cos(2\*pi\*delf\*tt).\*cos(2\*pi\*fc\*tt);

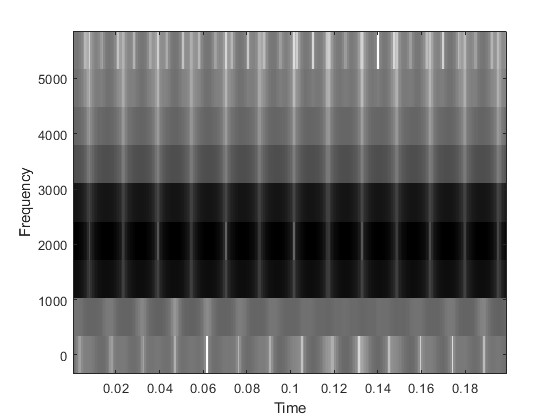
plot(tt,xx), grid on;



**c.** With the long section length, it is difficult to see how the frequency content changes with time. Also, the correct frequencies are present in the spectrogram.

**d.** With short section length, the spectrogram shows how the spectrum varies with time.

****

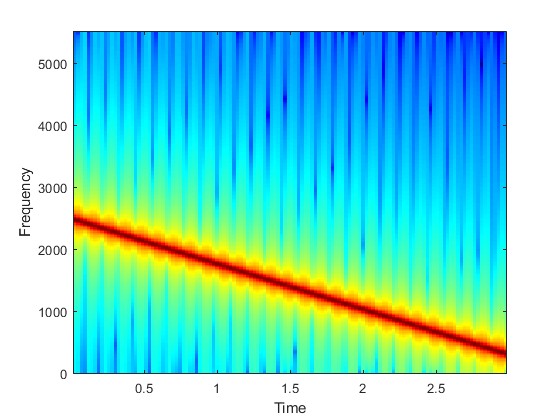
****

**4.3** [a,b]=mychirp(5000,300,3,11025);

soundsc(a,fsamp);

The frequency movement is linear. It chirps down as time passes on.

specgram(a,512,fsamp);



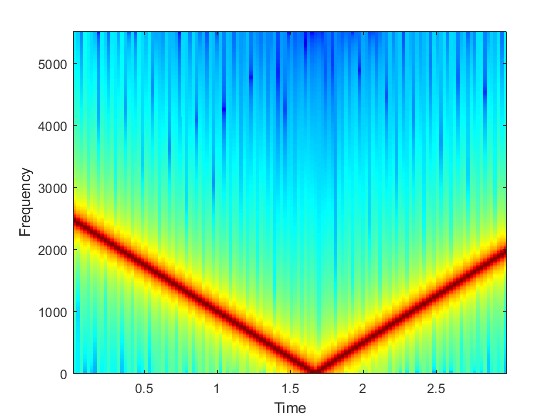
**4.4** fsamp=11025;

[a,b]=mychirp(3000,-2000,3,11025);

soundsc(a,fsamp);

The frequency movement is also linear. It chirps down for some time before chirping up.

specgram(a,512,fsamp);

****