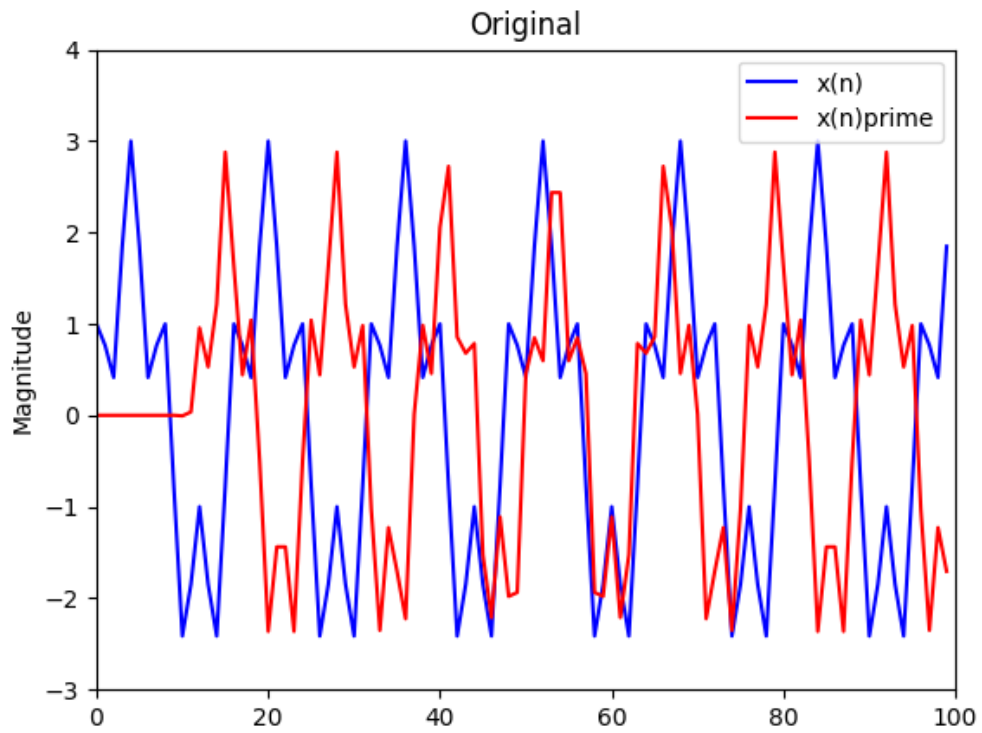
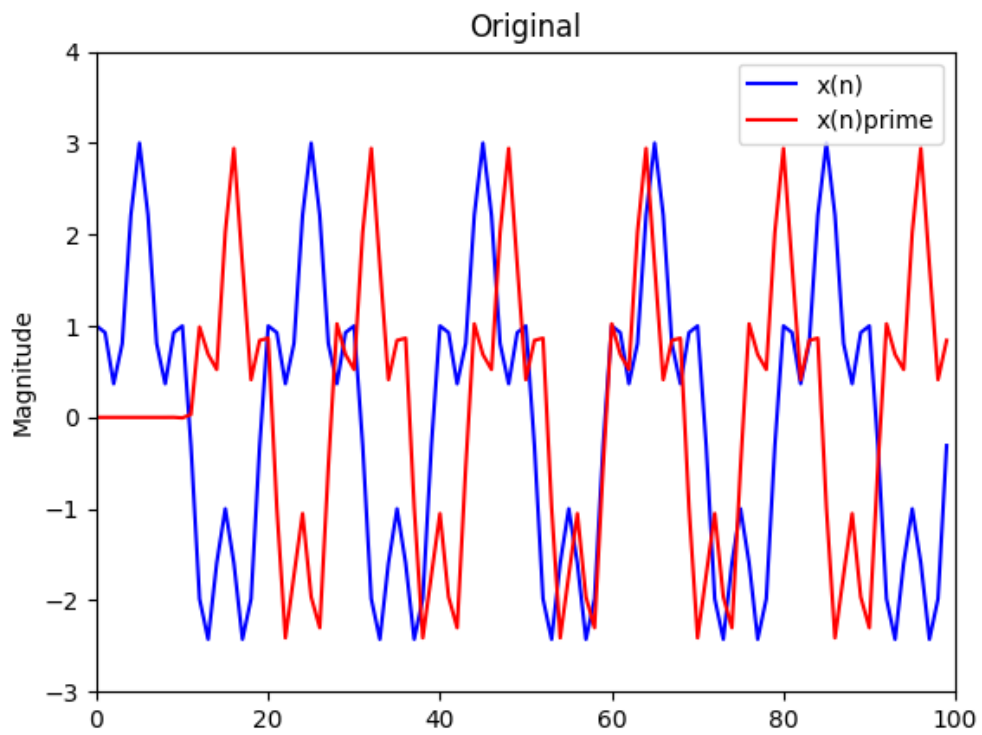


HW3-6

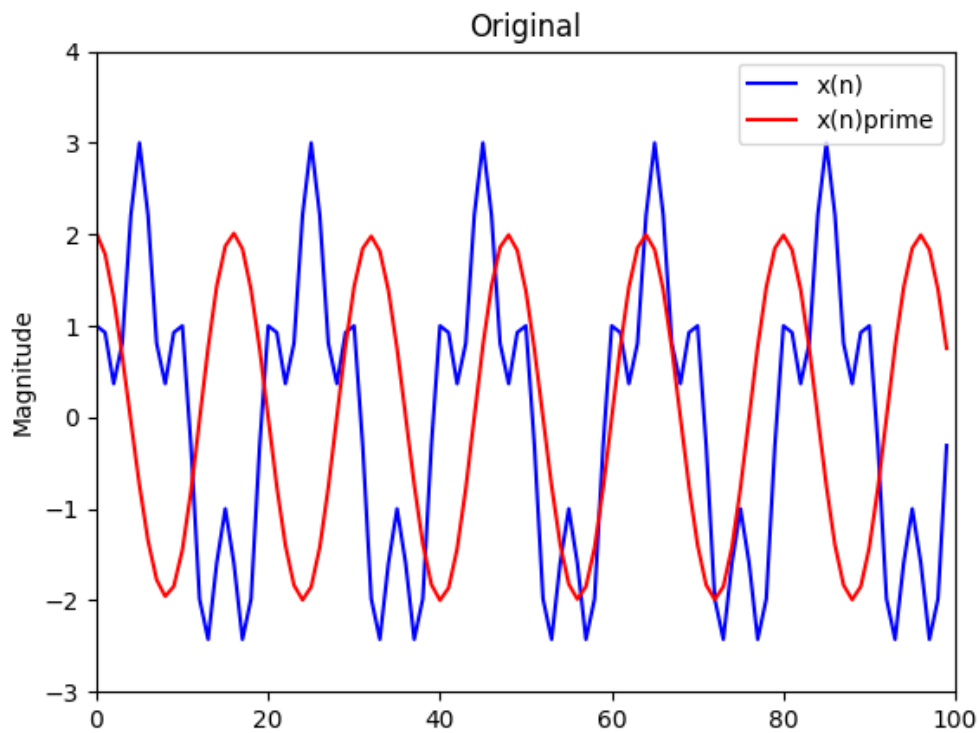
a)



b)



c)



程式碼

```

1 > import ...
4 def plot(fs):
5     xt= 2*np.sin( 2 * np.pi * f1 /fs * t) + np.cos( 2 * np.pi * f2 /fs * t )
6     yt=scipy.signal.upfirdn(bl,xt,U,D)
7     plt.plot( *args: t[0:100],xt[0:100], 'b', label='x(n)')
8     plt.plot( *args: t[0:100],yt[0:100], 'r', label='x(n)prime')
9     plt.title("Original")
10    plt.ylabel("Magnitude")
11    plt.xlim( *args: 0, 100)
12    plt.ylim( *args: -3, 4)
13    plt.legend()
14    plt.show()

1 usage
15 def plot2(fs):
16     xt= 2*np.sin( 2 * np.pi * f1 /fs * t) + np.cos( 2 * np.pi * f2 /fs * t )
17     xt1=scipy.signal.decimate(xt,D)
18     yt=scipy.signal.resample(xt1,U*1000)
19     plt.plot( *args: t[0:100],xt[0:100], 'b', label='x(n)')
20     plt.plot( *args: t[0:100],yt[20:120], 'r', label='x(n)prime')
21     plt.title("Original")
22     plt.ylabel("Magnitude")
23     plt.xlim( *args: 0, 100)
24     plt.ylim( *args: -3, 4)
25     plt.legend()
26     plt.show()

27 t = np.linspace( start: 0, stop: 4999, num: 5000)
28 f1,f2=500,2000
29 fs1,fs2,fs3=8000,10000,10000
30 U,D=4,5
31 N=24*D
32 bl = scipy.signal.firwin(N,1/D,window=('kaiser',N+1))
33 bl=U*bl
34 plot(fs1)
35 plot(fs2)
36 plot2(fs3)

```

3-6ANS:先下採樣一開始就濾掉高頻信號(或者說是雜訊)，在上採樣的話也只剩低頻訊號。