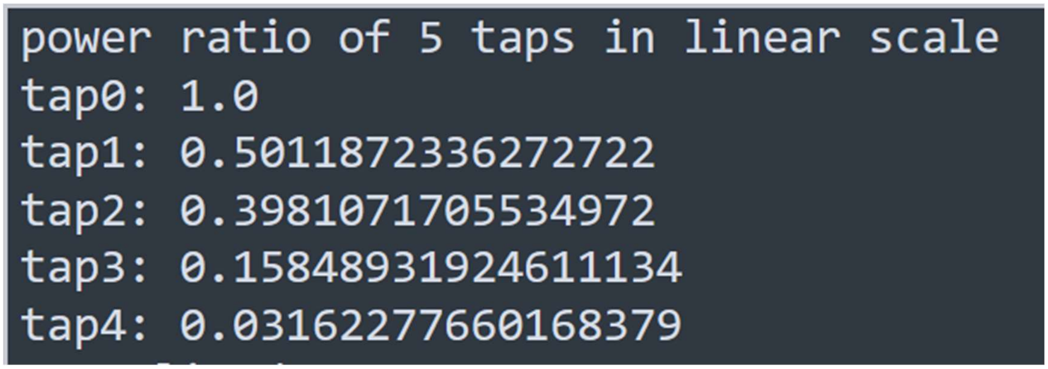
姓名:葉冠宏　　學號:r11943113

1.

請都執行 python Q1.py

(a)



```
power ratio of 5 taps in linear scale
tap0: 1.0
tap1: 0.5011872336272722
tap2: 0.3981071705534972
tap3: 0.15848931924611134
tap4: 0.03162277660168379
```

(b)

Mean excess delay: 17.60192779995999

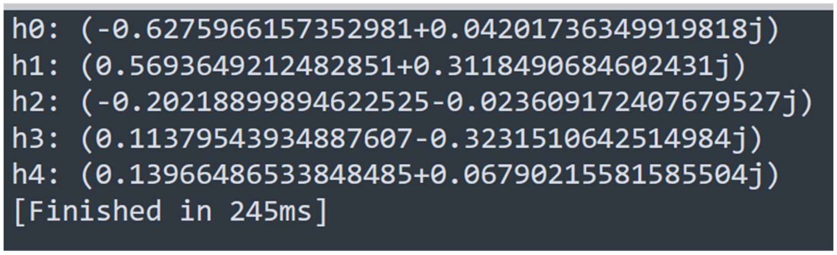
(c)

RMS excess delay: 25.60915740232992

2.

(a)

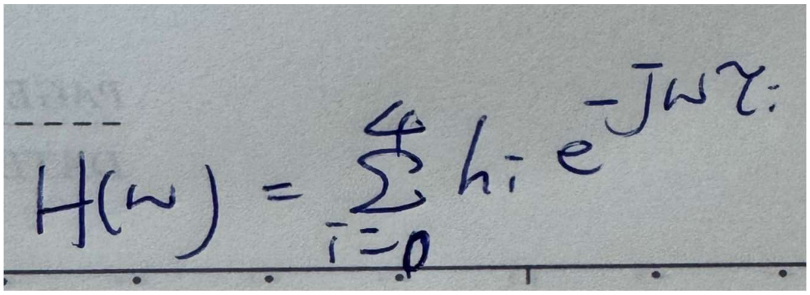
請執行 python Q2a.py



```
h0: (-0.6275966157352981+0.04201736349919818j)
h1: (0.5693649212482851+0.3118490684602431j)
h2: (-0.20218899894622525-0.023609172407679527j)
h3: (0.11379543934887607-0.3231510642514984j)
h4: (0.13966486533848485+0.06790215581585504j)
[Finished in 245ms]
```

(b)

Equation:

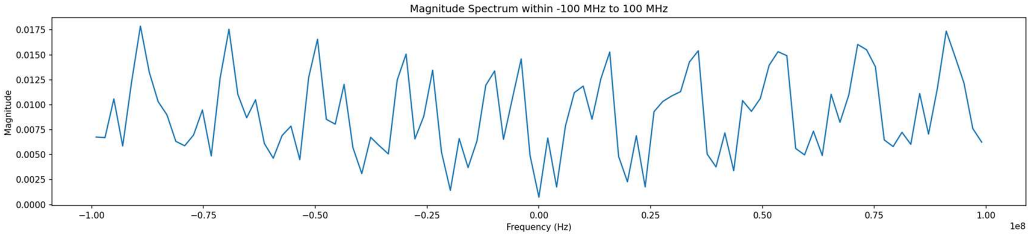


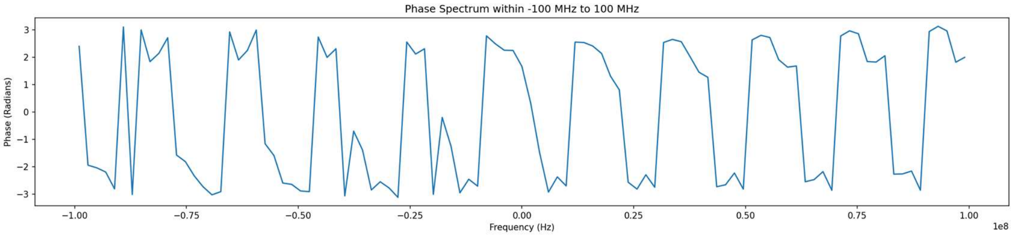
$$H(\omega) = \sum_{i=0}^{4} h_i e^{-j\omega\tau_i}$$

(c)

請都執行 python Q2c.py

magnitude and phase of channel frequency response:





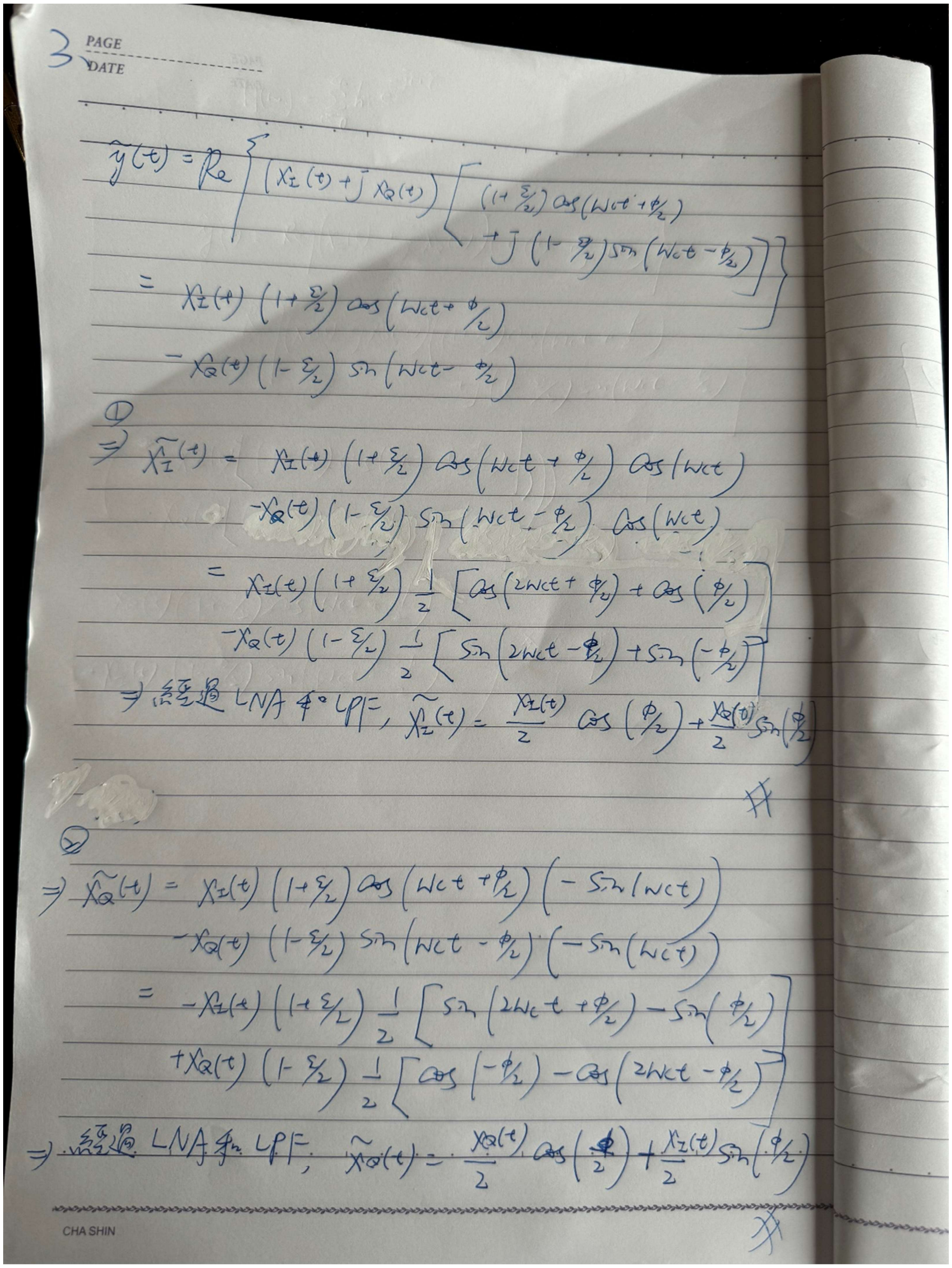
(d)

請都執行 python Q2d.py

coherence bandwidth: 7809707 Hz (= 1/(5* 25.60915740232992) *10^9)

13 subcarriers should be allocated (about 100M/ coherence bandwidth)

3.



PAGE 3

DATE

$$\tilde{y}(t) = Re\left\{ \left(x_I(t) + j\,x_Q(t)\right)\left[\left(1+\frac{\varepsilon}{2}\right)\cos\left(w_ct+\frac{\phi}{2}\right) + j\left(1-\frac{\varepsilon}{2}\right)\sin\left(w_ct-\frac{\phi}{2}\right)\right]\right\}$$

$$= x_I(t)\left(1+\frac{\varepsilon}{2}\right)\cos\left(w_ct+\frac{\phi}{2}\right) - x_Q(t)\left(1-\frac{\varepsilon}{2}\right)\sin\left(w_ct-\frac{\phi}{2}\right)$$

①

$$\Rightarrow \tilde{x}_I(t) = x_I(t)\left(1+\frac{\varepsilon}{2}\right)\cos\left(w_ct+\frac{\phi}{2}\right)\cos(w_ct) - x_Q(t)\left(1-\frac{\varepsilon}{2}\right)\sin\left(w_ct-\frac{\phi}{2}\right)\cos(w_ct)$$

$$= x_I(t)\left(1+\frac{\varepsilon}{2}\right)\frac{1}{2}\left[\cos\left(2w_ct+\frac{\phi}{2}\right)+\cos\left(\frac{\phi}{2}\right)\right] - x_Q(t)\left(1-\frac{\varepsilon}{2}\right)\frac{1}{2}\left[\sin\left(2w_ct-\frac{\phi}{2}\right)+\sin\left(-\frac{\phi}{2}\right)\right]$$

$\Rightarrow$ 經過 LNA 和 LPF, $\tilde{x}_I(t) = \frac{x_I(t)}{2}\cos\left(\frac{\phi}{2}\right) + \frac{x_Q(t)}{2}\sin\left(\frac{\phi}{2}\right)$

#

②

$$\Rightarrow \tilde{x}_Q(t) = x_I(t)\left(1+\frac{\varepsilon}{2}\right)\cos\left(w_ct+\frac{\phi}{2}\right)\left(-\sin(w_ct)\right) - x_Q(t)\left(1-\frac{\varepsilon}{2}\right)\sin\left(w_ct-\frac{\phi}{2}\right)\left(-\sin(w_ct)\right)$$

$$= -x_I(t)\left(1+\frac{\varepsilon}{2}\right)\frac{1}{2}\left[\sin\left(2w_ct+\frac{\phi}{2}\right)-\sin\left(\frac{\phi}{2}\right)\right] + x_Q(t)\left(1-\frac{\varepsilon}{2}\right)\frac{1}{2}\left[\cos\left(-\frac{\phi}{2}\right)-\cos\left(2w_ct-\frac{\phi}{2}\right)\right]$$

$\Rightarrow$ 經過 LNA 和 LPF, $\tilde{x}_Q(t) = \frac{x_Q(t)}{2}\cos\left(\frac{\phi}{2}\right) + \frac{x_I(t)}{2}\sin\left(\frac{\phi}{2}\right)$

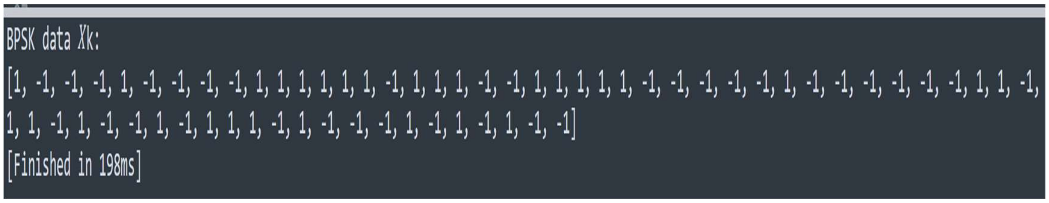
CHA SHIN

#

4.

請都執行 python Q4.py

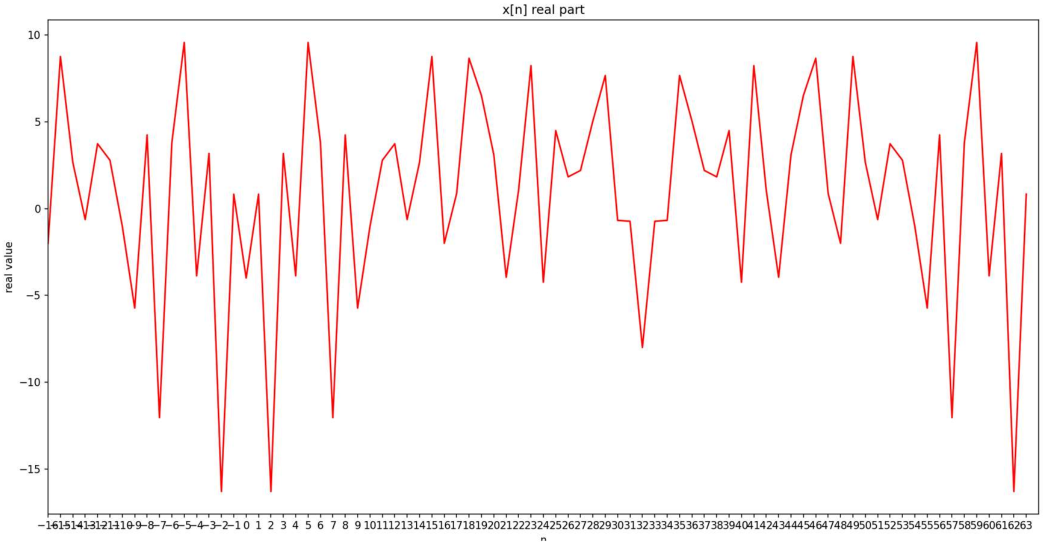
(a)



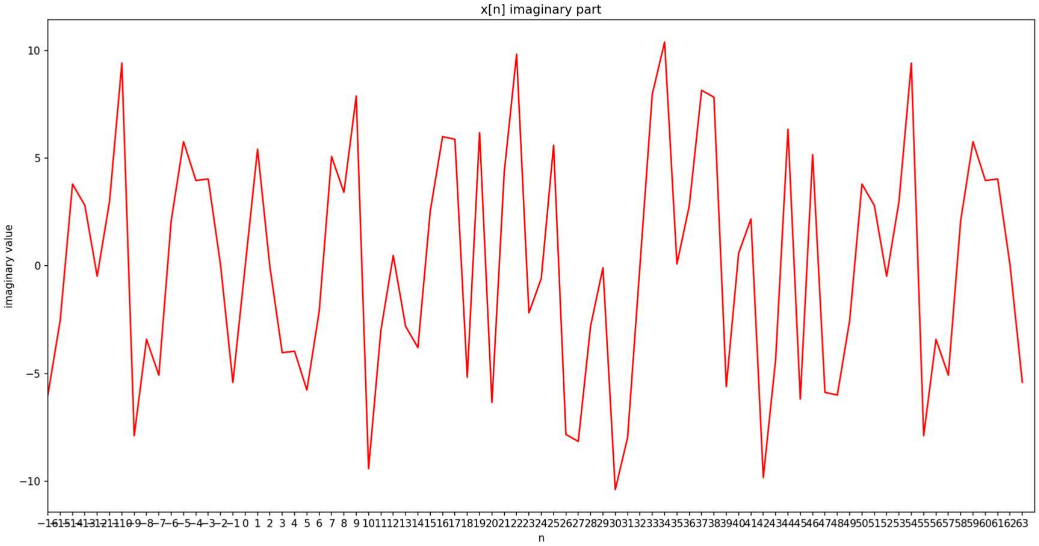
```
BPSK data Xk:
[1, -1, -1, -1, 1, -1, -1, -1, -1, 1, 1, 1, 1, 1, 1, -1, 1, 1, 1, -1, -1, 1, 1, 1, 1, 1, -1, -1, -1, -1, -1, 1, -1, -1, -1, -1, -1, -1, 1, 1, -1,
1, 1, -1, 1, -1, -1, 1, -1, 1, 1, 1, -1, 1, -1, -1, -1, 1, -1, 1, -1, 1, -1, -1]
[Finished in 198ms]
```

(b)

X[n] Real part:

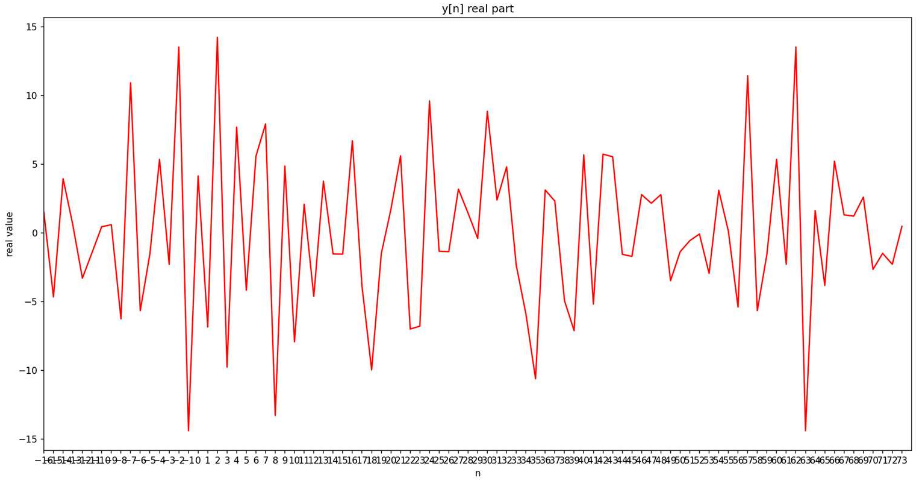


X[n] Imaginary part:

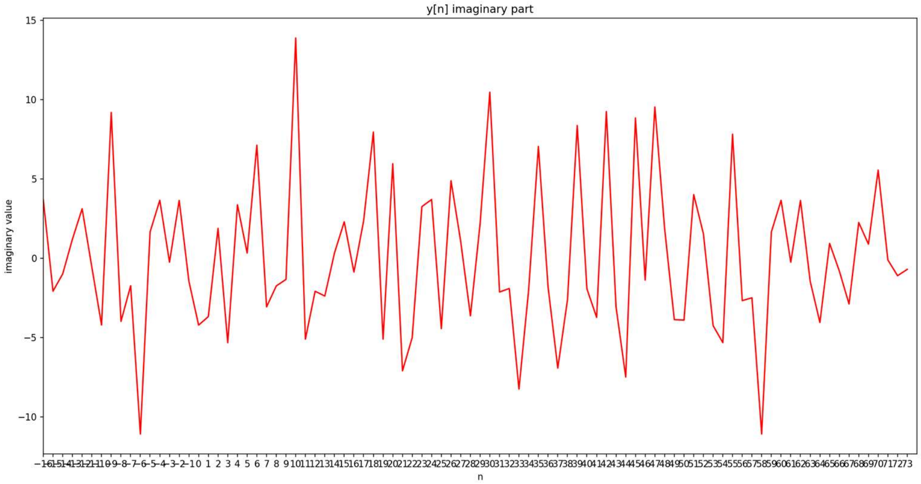


(c)

Y[n] real part:

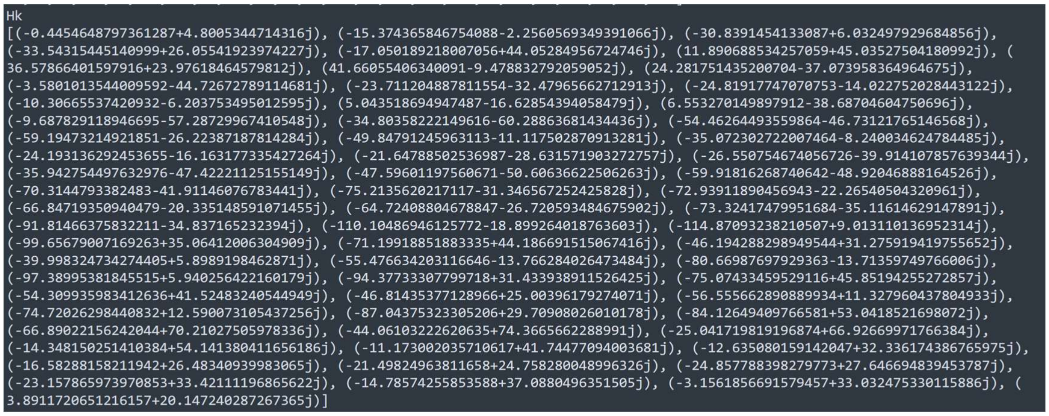


Y[n] imaginary part:



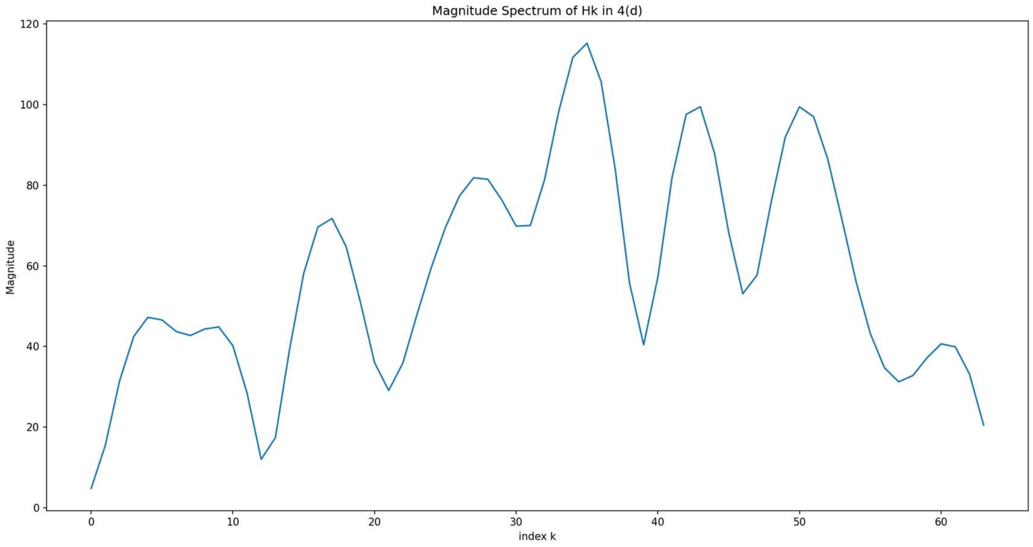
(d)

Hk:

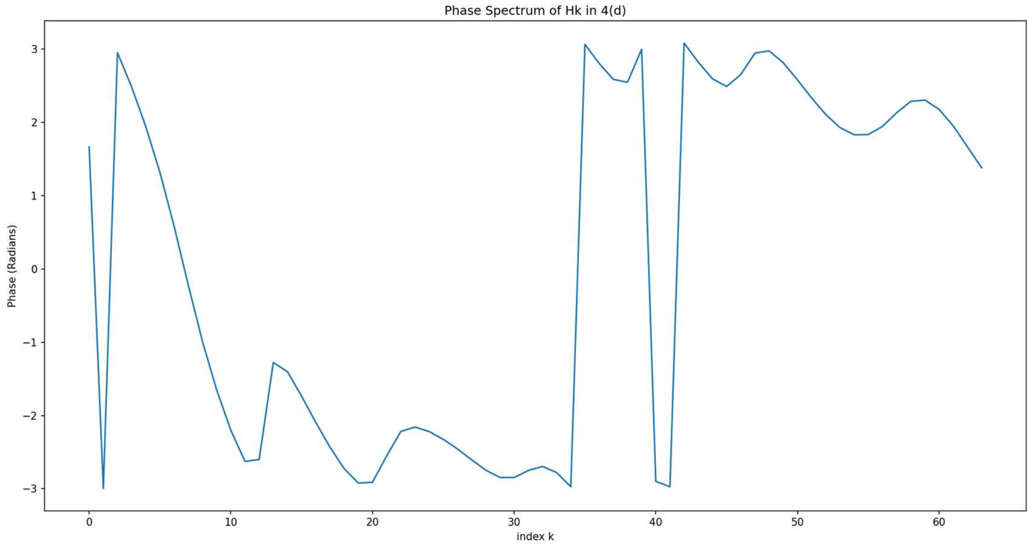


```
Hk
[(-0.4454648797361287+4.8005344714316j), (-15.374365846754088-2.2560569349391066j), (-30.8391454133087+6.032497929684856j),
(-33.54315445140999+26.05541923974227j), (-17.050189218007056+44.05284956724746j), (11.890688534257059+45.03527504180992j), (
36.57866401597916+23.97618464579812j), (41.66055406340091-9.478832792059052j), (24.281751435200704-37.073958364964675j),
(-3.5801013544009592-44.72672789114681j), (-23.711204887811554-32.47965662712913j), (-24.81917747070753-14.022752028443122j),
(-10.30665537420932-6.203753495012595j), (5.043518694947487-16.62854394058479j), (6.553270149897912-38.68704604750696j),
(-9.687829118946695-57.28729967410548j), (-34.80358222149616-60.28863681434436j), (-54.46264493559864-46.73121765146568j),
(-59.19473214921851-26.22387187814284j), (-49.84791245963113-11.117502870913281j), (-35.072302722007464-8.240034624784485j),
(-24.193136292453655-16.163177335427264j), (-21.64788502536987-28.631571903272757j), (-26.550754674056726-39.914107857639344j),
(-35.942754497632976-47.42221125155149j), (-47.59601197560671-50.60636622506263j), (-59.91816268740642-48.92046888164526j),
(-70.3144793382483-41.91146076783441j), (-75.2135620217117-31.346567252425828j), (-72.93911890456943-22.26540504320961j),
(-66.84719350940479-20.335148591071455j), (-64.72408804678847-26.720593484675902j), (-73.32417479951684-35.11614629147891j),
(-91.81466375832211-34.837165232394j), (-110.10486946125772-18.899264018763603j), (-114.87093238210507+9.013110136952314j),
(-99.65679007169263+35.06412006304909j), (-71.19918851883335+44.186691515067416j), (-46.194288298949544+31.275919419755652j),
(-39.998324734274405+5.8989198462871j), (-55.476634203116646-13.766284026473484j), (-80.66987697929363-13.71359749766006j),
(-97.38995381845515+5.940256422160179j), (-94.37733307799718+31.433938911526425j), (-75.07433459529116+45.85194255272857j),
(-54.309935983412636+41.52483240544949j), (-46.81435377128966+25.00396179274071j), (-56.555662890889934+11.327960437804933j),
(-74.72026298440832+12.590073105437256j), (-87.04375323305206+29.70908026010178j), (-84.12649409766581+53.0418521698072j),
(-66.89022156242044+70.21027505978336j), (-44.06103222620635+74.3665662288991j), (-25.041719819196874+66.92669971766384j),
(-14.348150251410384+54.141380411656186j), (-11.173002035710617+41.74477094003681j), (-12.635080159142047+32.336174386765975j),
(-16.58288158211942+26.48340939983065j), (-21.49824963811658+24.758280048996326j), (-24.857788398279773+27.646694839453787j),
(-23.157865973970853+33.42111196865622j), (-14.78574255853588+37.0880496351505j), (-3.1561856691579457+33.032475330115886j), (
3.8911720651216157+20.147240287267365j)]
```

Magnitude of Hk:



Phase of Hk:



(e)

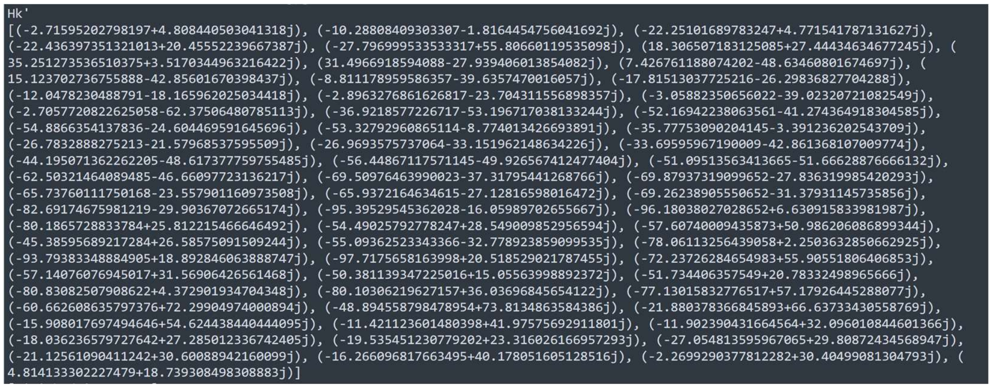
在 Q2(c)，我們是去把 multipath fading channel h[n]直接去做傅立葉轉換，得到的 spectrum Hk，其 x 軸是用 Hertz 來去表示。

而在 Q4(d)的過程中，我們利用的原理是根據 y[n] = h[n]⊗$x[n]$ 同時對等號左右兩邊左右兩邊做傅立葉轉換得到 Yk = Hk x Xk，然後我們用 Yk/ Xk 去求得每個不同 k 的時候的 Hk。也就是說我們是用 received signal 的頻譜和 input 的頻譜去求得 Hk 的頻譜，其 x 軸可以用 index 表示，代表說當每個不同 input Xk 的 index k 時，其所得到的 Hk 是多少。

如果我們去把 Q2(c)的頻譜做適當的接合會發現，其和 Q4(d)的頻譜是一樣的。

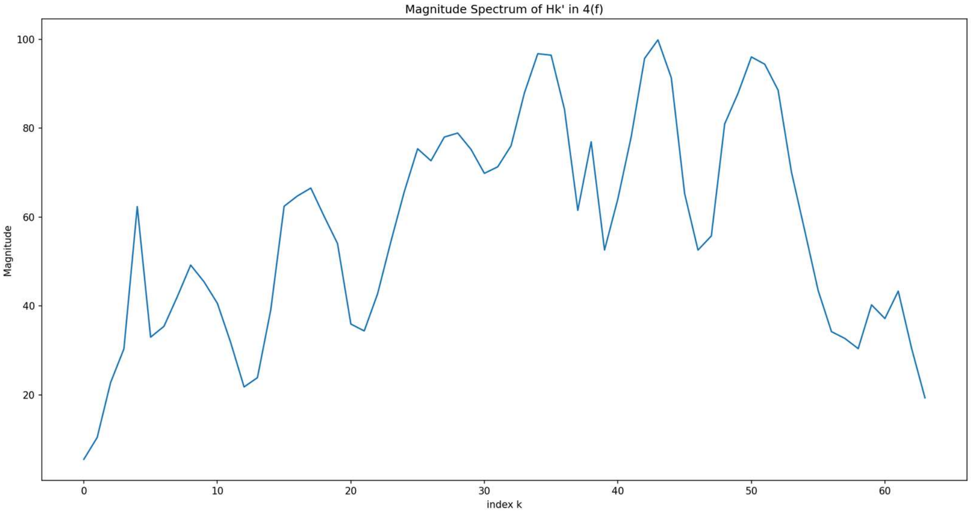
(f)

Hk’:

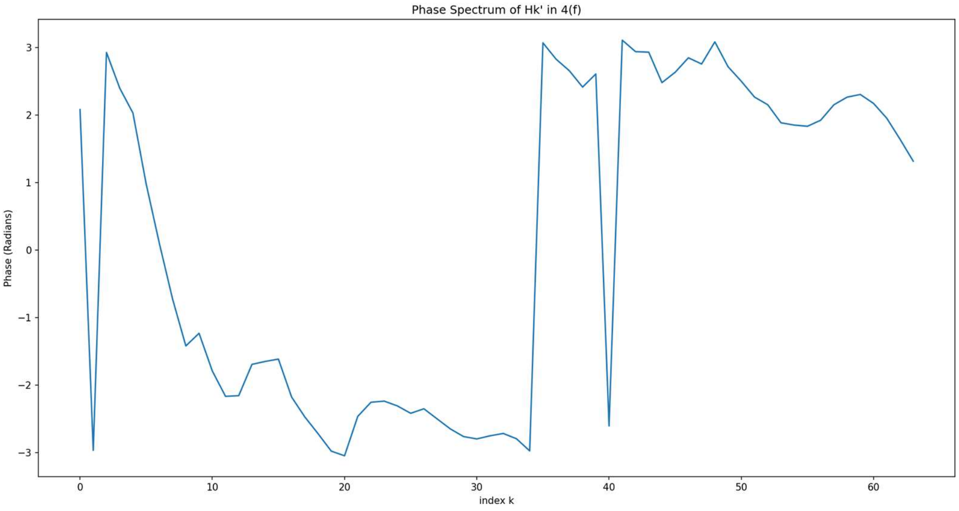


```
Hk'
[(-2.71595202798197+4.808440503041318j), (-10.28808409303307-1.8164454756041692j), (-22.25101689783247+4.771541787131627j),
(-22.436397351321013+20.45552239667387j), (-27.796999533533317+55.80660119535098j), (18.306507183125085+27.44434634677245j), (
35.251273536510375+3.5170344963216422j), (31.4966918594088-27.939406013854082j), (7.426761188074202-48.63460801674697j), (
15.123702736755888-42.85601670398437j), (-8.811178959586357-39.6357470016057j), (-17.81513037725216-26.29836827704288j),
(-12.0478230488791-18.165962025034418j), (-2.8963276861626817-23.704311556898357j), (-3.05882350656022-39.02320721082549j),
(-2.7057720822625058-62.37506480785113j), (-36.9218577226717-53.196717038133244j), (-52.16942238063561-41.274364918304585j),
(-54.8866354137836-24.604469591645696j), (-53.32792960865114-8.774013426693891j), (-35.77753090204145-3.391236202543709j),
(-26.7832888275213-21.57968537595509j), (-26.9693575737064-33.151962148634226j), (-33.69595967190009-42.861368107009774j),
(-44.195071362262205-48.617377759755485j), (-56.44867117571145-49.926567412477404j), (-51.09513563413665-51.66628876666132j),
(-62.50321464089485-46.66097723136217j), (-69.50976463990023-37.31795441268766j), (-69.87937319099652-27.836319985420293j),
(-65.73760111750168-23.557901160973508j), (-65.9372164634615-27.12816598016472j), (-69.26238905550652-31.37931145735856j),
(-82.69174675981219-29.90367072665174j), (-95.39529545362028-16.05989702655667j), (-96.18038027028652+6.630915833981987j),
(-80.1865728833784+25.812215466646492j), (-54.49025792778247+28.549009852956594j), (-57.60740009435873+50.986206086899344j),
(-45.38595689217284+26.58575091509244j), (-55.09362523343366-32.778923859099535j), (-78.06113256439058+2.2503632850662925j),
(-93.79383348884905+18.892846063888747j), (-97.7175658163998+20.518529021787455j), (-72.23726284654983+55.90551806406853j),
(-57.14076076945017+31.56906426561468j), (-50.381139347225016+15.05563998892372j), (-51.734406357549+20.78332498965666j),
(-80.83082507908622+4.372901934704348j), (-80.10306219627157+36.03696845654122j), (-77.13015832776517+57.17926445288077j),
(-60.662608635797376+72.29904974000894j), (-48.894558798478954+73.8134863584386j), (-21.880378366845893+66.63733430558769j),
(-15.908017697494646+54.624438440444095j), (-11.421123601480398+41.97575692911801j), (-11.902390431664564+32.096010844601366j),
(-18.036236579727642+27.285012336742405j), (-19.535451230779202+23.316026166957293j), (-27.054813595967065+29.80872434568947j),
(-21.12561090411242+30.60088942160099j), (-16.266096817663495+40.178051605128516j), (-2.2699290377812282+30.40499081304793j), (
4.814133302227479+18.739308498308883j)]
```

Magnitude of Hk’:



Phase of Hk’:



(g)
我們可以觀察到在 Hk 的 frequency magnitude 和 phase 中，整個頻譜顯現的較為平滑，而在 Hk’ 的 frequency magnitude 和 phase 中，整個曲線中會有許多不同的突起。
在 Q4(f)的計算過程中，由於你在 x[n]階段並沒有加入 cyclic prefix，因此在計算 convolution 的時候，當 y[n]比較小的時候，對於 transmitted waveform 前段的資訊並沒有和 cyclic form(也就是 waveform 後段)的資訊納入一起計算，因此造成整個頻譜不夠平滑，這就是為什麼 Q4(f) channel frequency response 不夠恰當。
而在 Q4(d)的做法中，我們在 x[n]的過程中有加入 Cyclic prefix，只是在最後已經 convolution 完得到 y[n]之後才去 remove cyclic prefix，這部份並不會影響當初 convolution 計算過程納入不同 h[n]或 x[n]的資訊，你只是擷取 y[n]某一時段的訊號而已。而由於你在 x[n]的過程有先加入 cyclic prefix，因此你在計算和 h[n] convolution 的時候，即使在計算 n 較前段，也可以將完整前後段的 x[n]資訊納入計算。
在實際應用中，加入 cyclic prefix，我們可以避免前一個 symbol 不會影響到下一個 symbol，而使所關注的 symbol 頻譜可以完整被保留好。