

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**VIỆN ĐIỆN TỬ - VIỄN THÔNG**



## **BÁO CÁO TRAINING MATLAB**

### **TUẦN 2**

**Nguyễn Xuân Hoàng**

**Trần Thu Mai Anh**

Hà Nội, 12 - 2020



# MỤC LỤC

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT .....	ii
DANH MỤC HÌNH VẼ.....	iii
CHƯƠNG 1. CƠ SỞ LÝ THUYẾT .....	1
1.1 Tín hiệu trên miền thời gian và miền tần số.....	1
1.2 Phổ tần của tín hiệu.....	2
1.3 SNR và cách quy đổi.....	2
1.3.1 SNR.....	2
1.3.2 Cách quy đổi đơn vị dBw sang dB .....	3
1.4 Tín hiệu băng gốc và bộ lọc.....	3
1.4.1 Tín hiệu băng gốc.....	3
1.4.2 Bộ lọc .....	3
1.5 Điều chế.....	7
1.5.1 Điều chế AM .....	7
CHƯƠNG 2. THỰC HÀNH .....	10
2.1 Điều chế và giải điều chế AM.....	10
2.1.1 Vẽ tín hiệu $y(t)$ .....	10
2.1.2 Sử dụng FFT để vẽ phổ biên độ $x(t)$ , $y(t)$ .....	10
2.1.3 Vẽ tín hiệu $r(t)$ dưới tác động của nhiễu trắng và $w(t)$ .....	11
2.1.4 Thiết kế bộ lọc thông thấp.....	13
2.2 Điều chế QAM.....	18
2.2.1 Vẽ dạng tín hiệu .....	18
2.2.2 Vẽ tín hiệu đầu thu sau lọc.....	19
2.3 Kết luận .....	21

## DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

- FS: Fourier Series
- DTFS: Discrete time fourier series
- FT: Fourier transform
- DTFT: Discrete time fourier transform
- DFT: Discrete fourier transform
- LPF: Low pass filter
- HPF: High pass filter
- BPF: Band pass filter
- BSF: Band stop filter
- AM: Amplitude Modulation
- AM – DSB – SC: Amplitude Modulation – Double – SideBand Supressed Carrier
- QAM: Quadrature AM
- $\exp()$ : ký hiệu của hàm e mũ

## DANH MỤC HÌNH VẼ

Hình 1.1 Sơ đồ hệ thống trên miền thân gian với đầu vào là hàm exp .....	1
Hình 1.2 Sơ đồ biến một tín hiệu X ở miền thời gian sang miền tần số .....	2
Hình 1.3 Đặc tuyến tần số của một mạch lọc .....	4
Hình 1.4 Bộ lọc thông thấp.....	5
Hình 1.5 Bộ lọc thông cao .....	6
Hình 1.6 Bộ lọc thông dải.....	6
Hình 1.7 Bộ lọc chắn dải .....	7
Hình 1.8 Sơ đồ bộ điều chế AM .....	8
Hình 1.9 Giải điều chế AM.....	8
Hình 1.10 Bộ thu phát điều chế QAM.....	9
Hình 2.1 Tín hiệu $y(t)$ sau điều chế .....	10
Hình 2.2 Phổ biên độ của $x(t)$ và $y(t)$ .....	11
Hình 2.3 Tín hiệu $r(t)$ dưới tác động của nhiễu trắng Gauss .....	12
Hình 2.4 Phổ biên độ của tín hiệu $w(t)$ trường hợp $r(t)$ không có nhiễu trắng.....	12
Hình 2.5 Phổ biên độ của tín hiệu $w(t)$ trường hợp $r(t)$ có nhiễu.....	13
Hình 2.6 Tín hiệu khôi phục $x(t)$ bằng lọc trên miền tần số không có nhiễu.....	14
Hình 2.7 Tín hiệu khôi phục $x(t)$ bằng lọc trên miền tần số có nhiễu với $SNR = 0$ .....	14
Hình 2.8 Tín hiệu khôi phục $x(t)$ bằng lọc trên miền tần số có nhiễu với $SNR = 10$ dB .....	15
Hình 2.9 Tín hiệu khôi phục $x(t)$ bằng lọc trên miền tần số có nhiễu với $SNR = 20$ dB .....	15
Hình 2.10 Tín hiệu khôi phục $x(t)$ sử dụng hàm lọc trên Matlab không nhiễu .....	16
Hình 2.11 Tín hiệu khôi phục $x(t)$ sử dụng hàm lọc trên Matlab có nhiễu với $SNR = 0$ .....	17
Hình 2.12 Tín hiệu khôi phục $x(t)$ sử dụng hàm lọc trên Matlab có nhiễu với $SNR = 10$ .....	17
Hình 2.13 Tín hiệu khôi phục $x(t)$ sử dụng hàm lọc trên Matlab có nhiễu với $SNR = 20$ .....	18

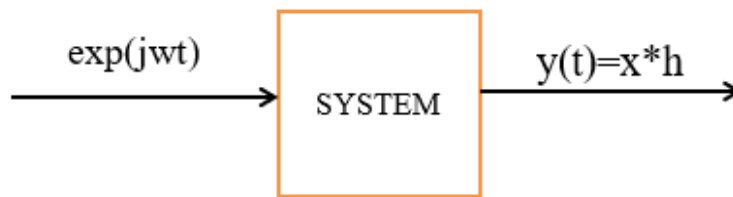
Hình 2.14 Tín hiệu phát của QAM.....	19
Hình 2.15 Tín hiệu khôi phục khi không có nhiễu.....	19
Hình 2.16 Tín hiệu khôi phục khi có nhiễu với $SNR = 0$ .....	20
Hình 2.17 Tín hiệu khôi phục khi có nhiễu với $SNR = 10$ .....	20
Hình 2.18 Tín hiệu khôi phục khi có nhiễu với $SNR = 20$ .....	21

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## 1.1 Tín hiệu trên miền thời gian và miền tần số

Tín hiệu là dạng biểu diễn của thông tin, mang thông tin và thay đổi theo thời gian. Một tín hiệu khi biểu diễn trên miền thời gian phức tạp nhưng lại đơn giản trên miền tần số.

Xét một hệ thống LTI có tín hiệu vào là một hàm e mũ như *Hình 1.1*



**Hình 1.1** Sơ đồ hệ thống trên miền thời gian với đầu vào là hàm exp

Xét tích chập:

$$y(t) = \int_{-\infty}^{\infty} x(\tau) \cdot h(t - \tau) d\tau$$

Vì tích chập có tính giao hoán nên:

$$y(t) = \int_{-\infty}^{\infty} h(\tau) \cdot x(t - \tau) d\tau = \int_{-\infty}^{\infty} h(\tau) \cdot e^{j\omega(t-\tau)} d\tau$$

$$\text{suy ra: } y(t) = e^{j\omega t} \cdot \int_{-\infty}^{\infty} h(\tau) \cdot e^{-j\omega\tau} d\tau$$

Hay:

$$y(t) = x(t) \cdot H(j\omega)$$

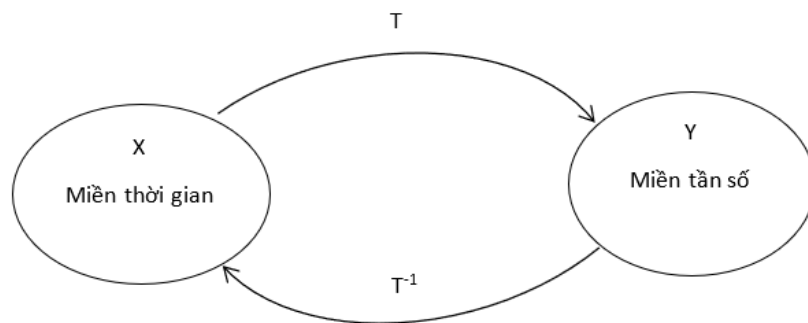
Với  $H(j\omega) = \int_{-\infty}^{\infty} h(t) \cdot e^{-j\omega t} \cdot dt$  gọi là đáp ứng tần số của hệ thống (thực chất là phép biến đổi Fourier FT).

Như vậy, nếu đầu vào  $x(t)$  ở miền thời gian là tổ hợp nhiều hàm exp  $(j\omega)$  ở các tần số khác nhau đưa vào hệ thống thì đầu ra  $y(t)$  chỉ là các phép tính nhân thông thường

chứ không còn là tích chập phức tạp. Do đó việc xét tín hiệu trên miền tần số đôi khi trực quan và dễ dàng hơn nhiều khi thực hiện trên miền thời gian với công cụ là các phép khai triển và biến đổi Fourier.

## 1.2 Phổ tần của tín hiệu

Như đã biết, các phép biến đổi toán học thực chất là phép ánh xạ biểu diễn tín hiệu từ không gian ban đầu (thực) sang miền không gian khác nhằm mục đích thuận tiện trong việc xử lý các đặc tính của tín hiệu (Hình 1.2).



**Hình 1.2 Sơ đồ biến một tín hiệu X ở miền thời gian sang miền tần số**

Một toán tử T biến đổi tín hiệu X trên miền thời gian thành các thành phần tín hiệu Y trên miền tần số. T có thể là FS (Fourier Series), DTFS (Discrete time fourier series), FT (Fourier transform), DTFT (Discrete time fourier transform) hay DFT (Discrete fourier transform).

Phân bố giá trị của các thành phần tín hiệu trên thang tần số gọi là phổ tần số (gọi tắt là phổ) tín hiệu. Trường hợp tổng quát, người ta dùng phổ biên độ và phổ pha để phân tích các đặc tính của tín hiệu.

## 1.3 SNR và cách quy đổi

### 1.3.1 SNR

SNR (Signal to noise ratio) là đại lượng dùng để so sánh mức độ tín hiệu mong muốn với mức độ ồn xung quanh. SNR được xác định bằng tỉ số giữa công suất của tín hiệu và công suất của độ ồn, thường được biểu diễn dưới dạng đơn vị deciben (dB). Nếu tỉ số cao hơn 1 : 1 (lớn hơn 0 dB) thì có nghĩa là tín hiệu nhiều hơn độ ồn.

$$SNR = \frac{P_{signal}}{P_{noise}}$$



### 1.3.2 Cách quy đổi đơn vị dBw sang dB

#### 1.3.2.1 W sang dBW

dBW là một đơn vị đo cường độ của tín hiệu được biểu thị bằng decibel so với một watt. Nó được sử dụng vì khả năng thể hiện cả giá trị công suất lớn và nhỏ trong phạm vi ngắn.

$$Power \text{ in dBW} = 10 \log \left( \frac{Power}{1W} \right)$$

Hoặc:

$$Power \text{ in W} = 10^{Power \text{ in } \frac{dBW}{10}}$$

#### 1.3.2.2 Số lần sang dB

$$\begin{aligned} SNR_{dB} &= 10 \log_{10} SNR = 10 \log_{10} \frac{P_{signal}}{P_{noise}} = 10(\log_{10} P_{signal} - \log_{10} P_{noise}) \\ &= P_{signal, dB} - P_{noise, dB} \end{aligned}$$

Hay ta có

$$\begin{aligned} SNR_{dB} &= 10 \log_{10} \left( \frac{A_{signal}}{A_{noise}} \right)^2 = 20 \log_{10} \left( \frac{A_{signal}}{A_{noise}} \right) \\ &= 20(\log_{10} A_{signal} - \log_{10} A_{noise}) \end{aligned}$$

Nên suy ra, số lần:  $\sigma = 10^{-\frac{SNR_{db}}{20}}$

## 1.4 Tín hiệu băng gốc và bộ lọc

### 1.4.1 Tín hiệu băng gốc

Tín hiệu băng gốc đề cập đến dải tần số ban đầu của tín hiệu trước khi nó chuyển đổi hoặc điều chế sang dải tần số khác. Nó chính là tín hiệu mang thông tin ban đầu mà ta muốn phát đi.

### 1.4.2 Bộ lọc

Trong xử lý tín hiệu, bộ lọc là thiết bị hoặc là quá trình loại bỏ một số thành phần hay tính năng không cần thiết của tín hiệu.

Xét một hệ thống LTI với tín hiệu đầu vào là 1 hàm tuần hoàn. Như đã xét, ta có thể biểu diễn:

$$y(t) = x(t) \cdot H(j\omega)$$

$$\text{Với } H(j\omega) = \int_{-\infty}^{\infty} h(t) \cdot e^{-j\omega t} \cdot dt$$

$$\text{Nếu đầu vào là tổ hợp của nhiều hàm tuần hoàn: } x(t) = \sum_{n=-\infty}^{\infty} A_n e^{jn\omega t}$$

$$\text{Thì } y(t) = \sum_{n=-\infty}^{\infty} A_n e^{jn\omega t} \cdot H(jn\omega) \text{ và } H(j\omega) = \int_{-\infty}^{\infty} h(t) \cdot e^{-j\omega t} \cdot dt$$

Nhận thấy,  $y(t)$  được biểu diễn dưới dạng chuỗi Fourier. Các thành phần của  $x(t)$  khi qua hệ thống LTI sẽ bị thay đổi khác nhau về biên độ và pha tùy thuộc vào  $H(j\omega)$ . Do đó hệ thống LTI đóng vai trò như là một bộ lọc tần số với  $H(j\omega)$  là đáp ứng tần số.

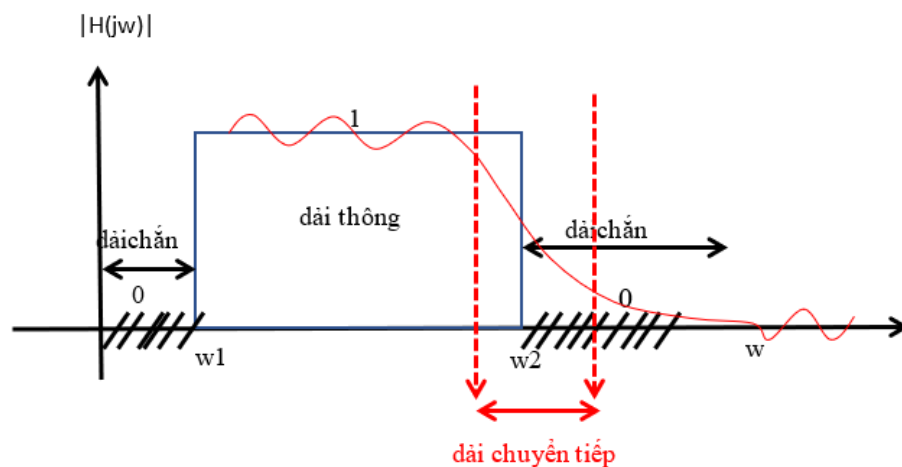
Với hệ thống LTI rời rạc, dùng tính chất của DTFT ta cũng thu được:

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega})$$

Với  $H(e^{j\omega})$  được gọi là đáp ứng tần số của hệ thống, và được tính bởi công thức:

$$H(e^{j\omega}) = FT[h(n)] = \sum_{n=-\infty}^{+\infty} h(n)e^{-j\omega n}$$

Vậy một hệ thống cho đầu vào đi qua trong 1 khoảng tần số nhất định và bị chặn lại ở những khoảng tần số còn lại thì được gọi là bộ lọc. Hình vẽ *Hình 1.3* mô tả đặc tuyến tần số cho một mạch lọc.



**Hình 1.3 Đặc tuyến tần số của một mạch lọc**

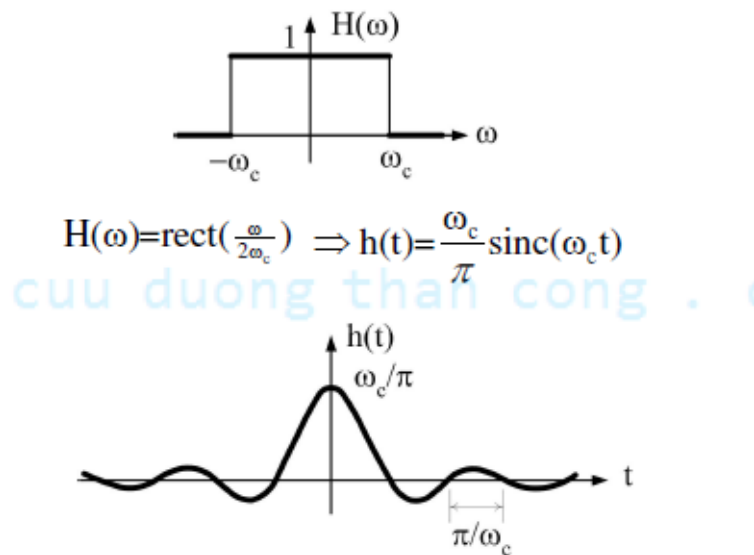
Hình 1.3 cho thấy một bộ lọc sẽ bao gồm 3 dải tần số cơ bản là dải chặn, dải thông và dải chuyển tiếp. Trong đó dải thông sẽ cho những tần số nằm trong đó đi qua, còn dải chặn sẽ chặn lại những tần số đi qua nó. Tùy vào mục đích điều chế mà ta sẽ thiết kế bộ lọc sao cho phù hợp.

Có ba bộ lọc lý tưởng, gồm:

- Bộ lọc thông thấp lý tưởng
- Bộ lọc thông cao lý tưởng
- Bộ lọc thông dải lý tưởng
- Bộ lọc chặn dải lý tưởng

#### 1.4.2.1 Bộ lọc thông thấp (LPF)

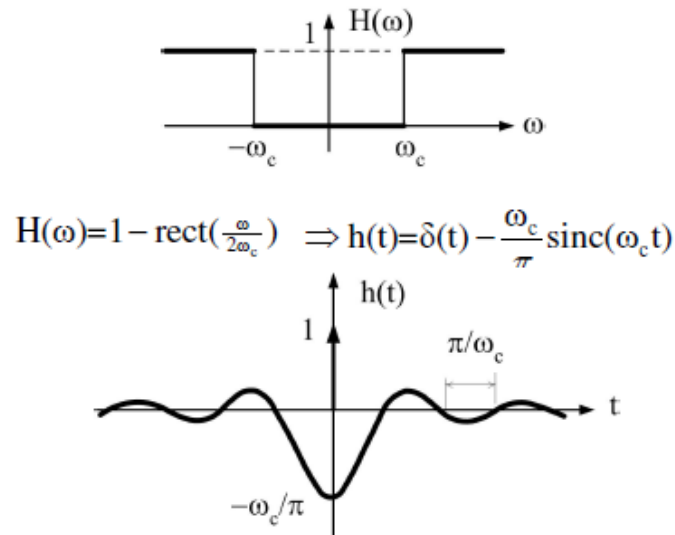
Bộ lọc thông thấp là bộ lọc cho tín hiệu có tần số thấp hơn tần số cutoff đã chọn và làm suy giảm tín hiệu có tần số cao hơn tần số cutoff. Đáp ứng tần số của bộ lọc phụ thuộc vào thiết kế của bộ lọc. Bộ lọc thông thấp là bổ sung của bộ lọc thông cao. Tham khảo mô hình LPF trong Hình 1.4 với  $H(\omega)$  là đáp ứng tần số của bộ lọc,  $h(t)$  là đáp ứng xung ở miền thời gian.



Hình 1.4 Bộ lọc thông thấp

#### 1.4.2.2 Bộ lọc thông cao (HPF)

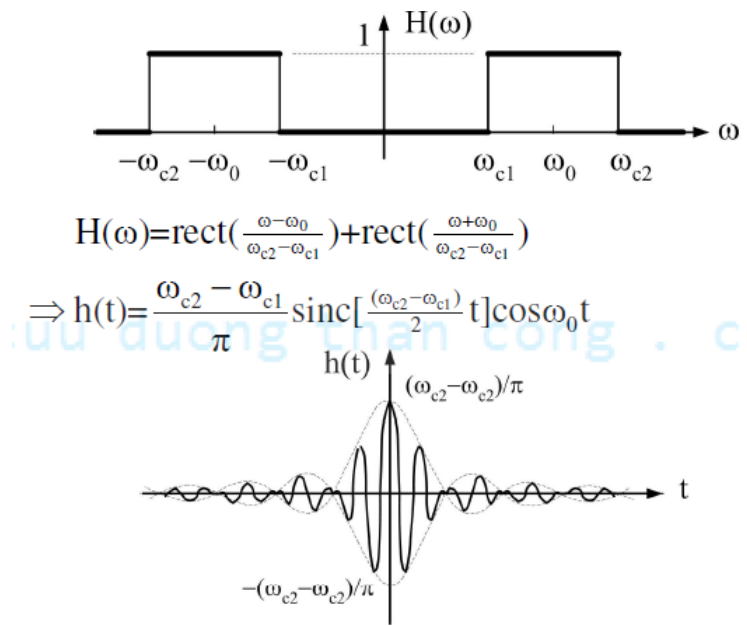
Bộ lọc thông cao (Hình 1.5) là bộ lọc cho tín hiệu có tần số cao hơn tần số cutoff và làm giảm tín hiệu có tần số thấp hơn tần số cutoff. Lượng suy giảm của tần số phụ thuộc vào thiết kế của bộ lọc. Bộ lọc thông cao thường được sử dụng cho hệ thống LTI.



Hình 1.5 Bộ lọc thông cao

#### 1.4.2.3 Bộ lọc thông dải (BPF)

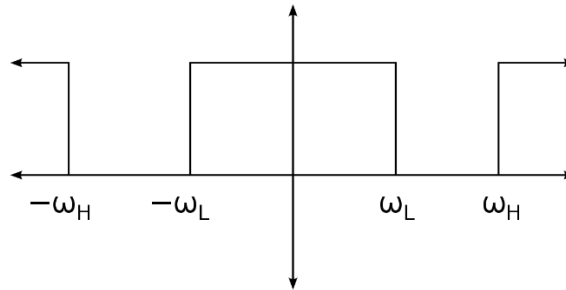
Bộ lọc thông dải là bộ lọc cho tín hiệu có tần số đi qua trong một dải tần số nào đó, được mô tả như Hình 1.6



Hình 1.6 Bộ lọc thông dải

#### 1.4.2.4 Bộ lọc chắn dải (BSF)

Bộ lọc chắn dải là bộ lọc không cho tín hiệu có tần số đi qua trong một dải tần số nào đó, được mô tả như *Hình 1.7*



**Hình 1.7 Bộ lọc chắn dải**

### 1.5 Điều chế

Điều chế là quá trình chuyển đổi phổ tần của tín hiệu cần truyền đến một vùng phổ tần khác bằng cách dùng một sóng mang để chuyên chở tín hiệu cần truyền đi. Mục đích của việc làm này là chọn một phổ tần thích hợp cho việc truyền thông tin đồng thời thỏa mãn nguyên lý bức xạ điện từ và ghép kênh theo tần số.

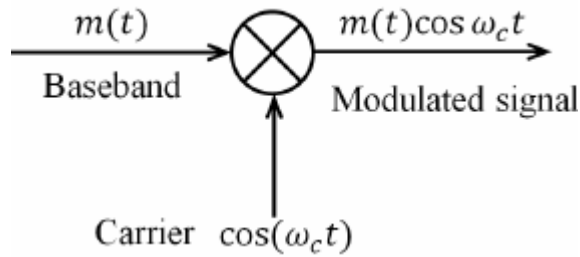
Thành phần trong tín hiệu điều chế gồm tín hiệu sóng mang (carrier wave) và tín hiệu băng gốc hay còn gọi là tín hiệu mang thông tin (informational signal).

Một cách tổng quát, phương pháp điều chế là dùng tín hiệu cần truyền làm thay đổi một thông số nào đó của sóng mang (biên độ, tần số, pha,...). Tùy vào thông số được lựa chọn mà có các phương pháp điều chế khác nhau như điều biên (AM), điều tần (FM), điều xung (PM),...

#### 1.5.1 Điều chế AM

Nguyên tắc: Nhân với tín hiệu sóng mang điều hoà để dịch phổ. Ta xét bộ điều chế AM như *Hình 1.8*. Tín hiệu băng gốc là  $m(t)$ , sau khi cho đi qua một bộ nhân với sóng ma ta thu được sóng được điều chế  $y_{AM}(t) = m(t) \cos w_c t$ . Áp dụng tính chất của phép biến đổi Fourier, ta biểu diễn tín hiệu được điều chế trên miền tần số:

$$Y_{AM}(w) = \frac{1}{2} M(w - w_c) + \frac{1}{2} M(w + w_c)$$



**Hình 1.8 Sơ đồ bộ điều chế AM**

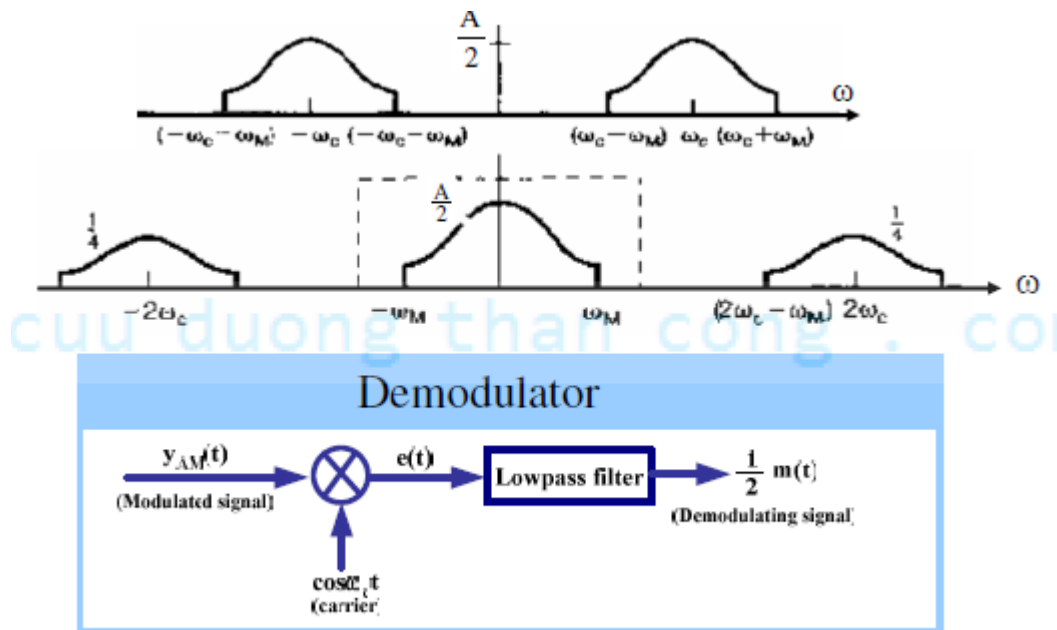
Từ phương trình, ta có thể thấy phổ tần của tín hiệu  $m(t)$  sau khi được điều chế được đẩy lên tần số cao hơn (tính chất dịch tần số).

Sau khi qua kênh thông tin, để giải điều chế ta cần thực hiện tách sóng đồng bộ  $e(t) = y_{AM} m(t) \cos \omega_c t = (m(t) \cos \omega_c t) \cos \omega_c t$ . Do Sóng mang thông tin đầu phát và đầu thu khác nhau về pha nên cần đồng bộ.

Xét trên miền tần số:

$$E(\omega) = \frac{1}{2}M(\omega) + \frac{1}{4}[M(\omega + 2\omega_c) + M(\omega - 2\omega_c)].$$

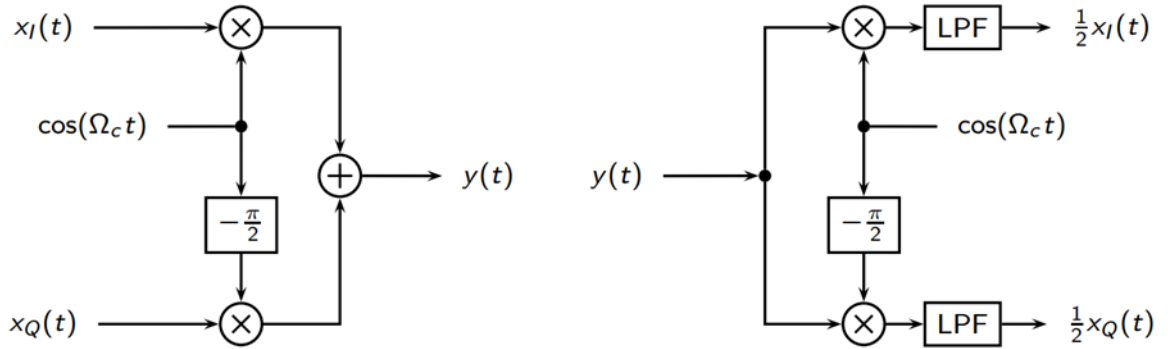
Để thu được tín hiệu ban đầu, ta sử dụng bộ lọc LPF để loại bỏ đi thành phần tần số đã bị dịch. Lưu ý rằng biên độ của tín hiệu khôi phục bằng một nửa tín hiệu mang tin ban đầu (Hình 1.9).



**Hình 1.9 Giải điều chế AM**

### 1.5.1.1 Điều chế QAM

AM – DSB – SC chiếm băng thông 2 lần so với tín hiệu mang thông tin nên không hiệu quả. QAM sử dụng 2 sóng mang cùng tần số nhưng pha trực giao (phase quadrature) để truyền tín hiệu. Hình 1.10 minh hoạt một bộ thu phát dùng điều chế QAM.



**Hình 1.10 Bộ thu phát điều chế QAM**

Tín hiệu đầu vào được tách thành hai loại là  $x_I(t)$  và  $x_Q(t)$ . Cho  $x_I(t)$  qua bộ nhân với  $\cos(\Omega_c t)$  và cho  $x_Q(t)$  qua bộ nhân với  $\cos\left(\Omega_c t - \frac{\pi}{2}\right)$ . Khi đó tín hiệu  $y(t)$  sau khi điều chế là:

$$\begin{aligned} y(t) &= x_I(t) \cdot \cos(\Omega_c t) + x_Q(t) \cdot \cos\left(\Omega_c t - \frac{\pi}{2}\right) \\ &= x_I(t) \cdot \cos(\Omega_c t) + x_Q(t) \cdot \sin(\Omega_c t) \end{aligned}$$

Lý tưởng trên kênh truyền không có nhiễu, bên bộ thu sẽ nhận được tín hiệu  $y(t)$ . Giải điều chế ra tín hiệu  $x_I(t)$  và  $x_Q(t)$  bằng cách lần lượt cho đi qua bộ nhân với  $\cos(\Omega_c t)$  và  $\sin(\Omega_c t)$

$$\begin{aligned} y(t) \cdot \cos(\Omega_c t) &= x_I(t) \cdot \cos(\Omega_c t)^2 + x_Q(t) \cdot \sin(\Omega_c t) \cdot \cos(\Omega_c t) \\ &= x_I(t) \cdot \frac{1}{2} + \frac{1}{2} \cdot x_I(t) \cdot \cos(2 \cdot \Omega_c t) + x_Q(t) \cdot \sin(2 \cdot \Omega_c t) \end{aligned}$$

Qua bộ lọc thông thấp LPF ta nhận được tín hiệu :  $\hat{x}_I(t) = \frac{1}{2} \cdot x_I(t)$ , tương tự với tín hiệu còn lại ta cũng thu được  $\hat{x}_Q(t) = \frac{1}{2} \cdot x_Q(t)$ .

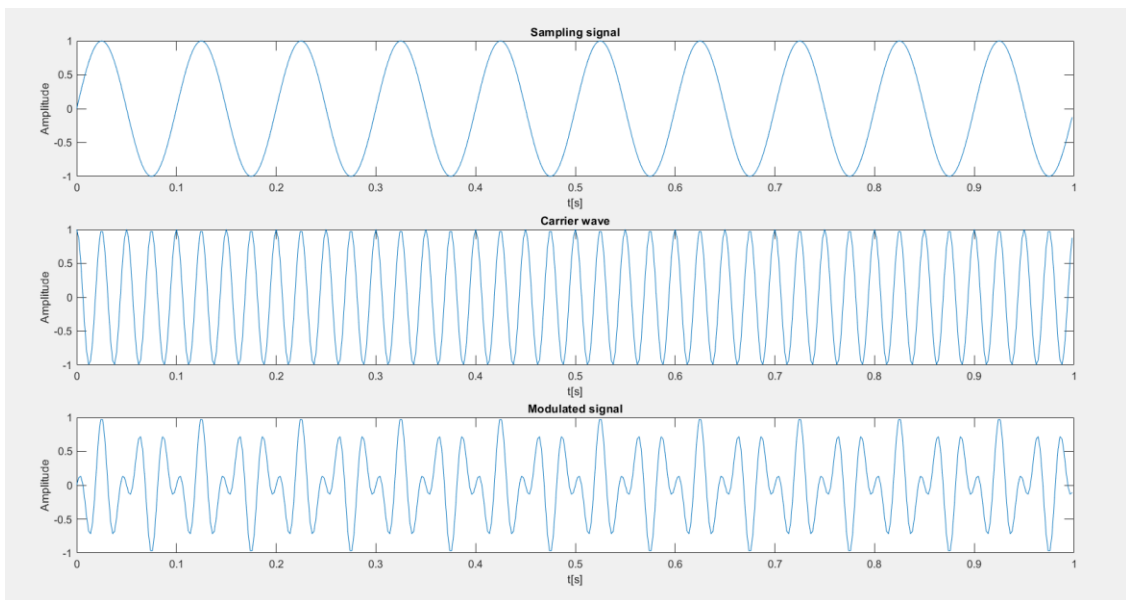
## CHƯƠNG 2. THỰC HÀNH

### 2.1 Điều chế và giải điều chế AM

#### 2.1.1 Vẽ tín hiệu $y(t)$

- Đầu vào là tín hiệu hình sin  $x(t) = \cos(2\pi f t)$  với  $f = 10$  Hz, tần số sóng mang  $f_0 = 40$  Hz,  $t$  từ 0 tới 1 s và tần số lấy mẫu là 500 Hz.
- Đầu ra là tín hiệu  $y(t)$  với  $y(t) = x(t) \cdot \cos(2\pi f_0 t)$

**Kết quả hiện ra màn hình:**



**Hình 2.1 Tín hiệu  $y(t)$  sau điều chế**

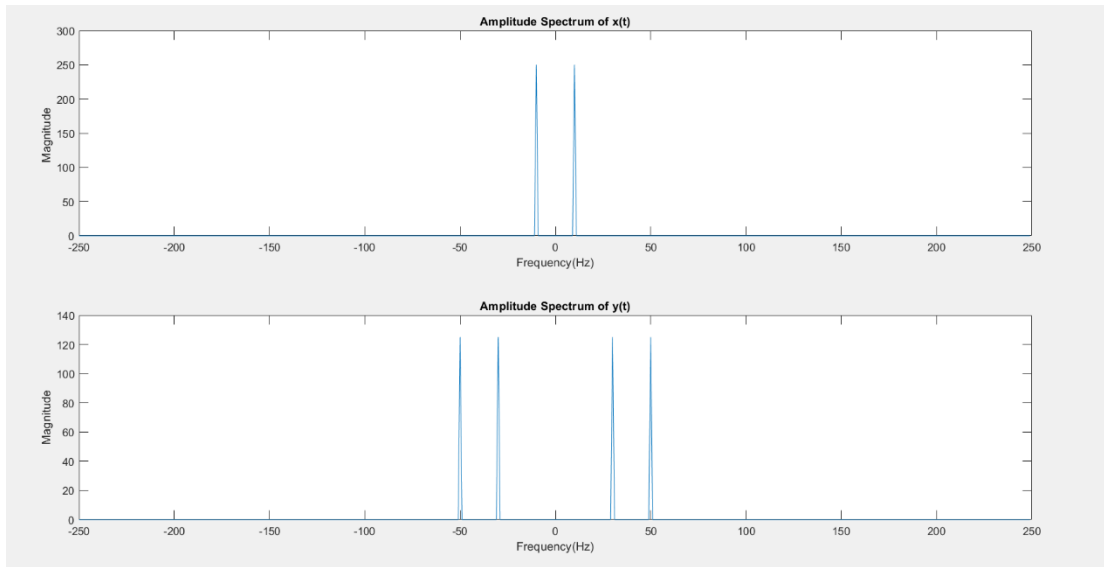
Hình 2.2 mô tả kết quả mô phỏng giải điều chế tín hiệu  $y(t)$ . Có thể thấy, tín hiệu này là một sóng có tính chu kỳ, nhưng không còn dạng của một sóng sin ban đầu.

#### 2.1.2 Sử dụng FFT để vẽ phổ biên độ $x(t)$ , $y(t)$

- Đầu vào là tín hiệu hình sin  $x(t) = \cos(2\pi f t)$  với  $f = 10$  Hz, tần số sóng mang  $f_0 = 40$  Hz,  $t$  từ 0 tới 1 s và tần số lấy mẫu là 500 Hz. Tín hiệu sau điều chế  $y(t)$  với  $y(t) = x(t) \cdot \cos(2\pi f_0 t)$
- Đầu ra là phổ biên độ các tín hiệu  $x(t)$  và  $y(t)$



c. Kết quả hiện ra màn hình:



**Hình 2.2 Phổ biên độ của  $x(t)$  và  $y(t)$**

Hình 2.2 mô tả kết quả mô phỏng phổ biên độ của tín hiệu đầu vào  $x(t)$  và tín hiệu đã qua điều chế  $y(t)$ . Do hàm  $x(t)$  ban đầu chỉ gồm 1 thành phần tần số là 10Hz nên khi sử dụng hàm  $\text{fft}()$  để phân tích phổ, có hai phổ biên độ xuất hiện tại 10Hz và -10Hz. Phân phổ âm là do tính chất của phép biến đổi Fourier nhưng thực tế ta chỉ quan tâm đến phần tần dương (thực tế tần số không âm). Sau khi qua điều chế, do tính chất dịch tần số của phép biến đổi Fourier, 1 phổ biên của tín hiệu gốc sẽ bị dịch đi 40Hz về cả hai phía. Do đó ta nhận được kết quả phổ biên ở các tần số 50Hz, -50Hz, 30Hz và -30Hz.

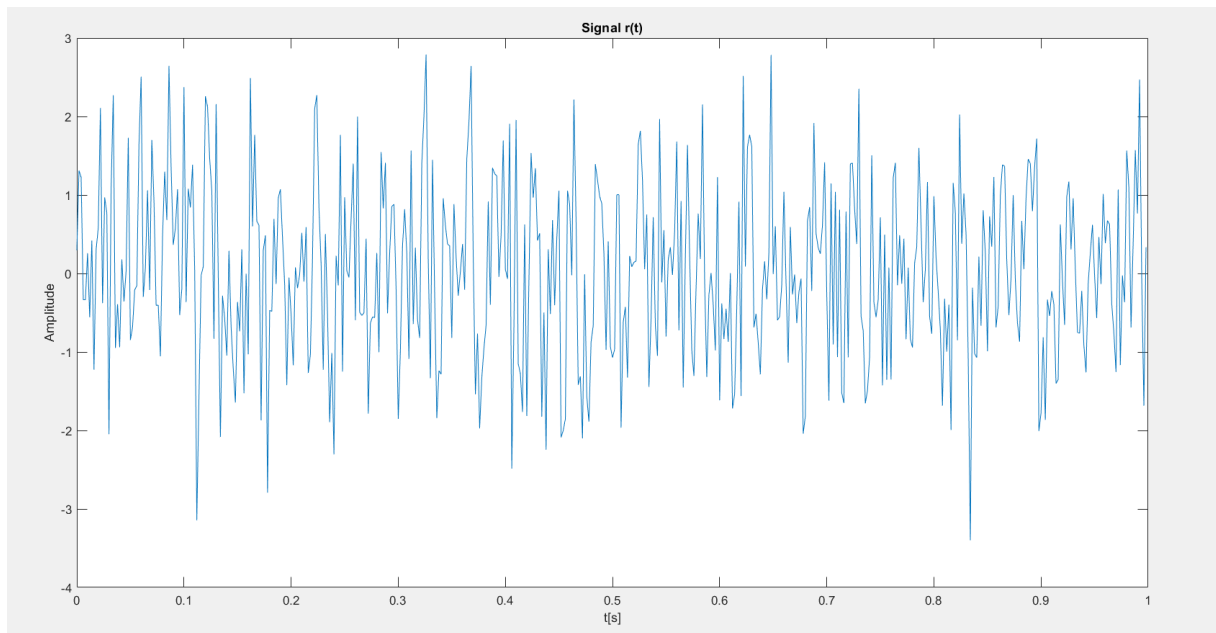
### 2.1.3 Vẽ tín hiệu $r(t)$ dưới tác động của nhiễu trắng và $w(t)$

- Đầu vào là tín hiệu sau điều chế  $y(t)$  với  $y(t) = x(t) \cdot \cos(2\pi f_0 t)$ , tỉ lệ sóng trên nhiễu  $\text{SNR} = 10 \text{ dB}$ , phương trình sóng mang là  $\cos(2\pi f_0 t)$ .
- Đầu ra là dạng tín hiệu tại Anten thu  $r(t)$  và phổ biên độ tín hiệu  $w(t)$  tại máy thu.

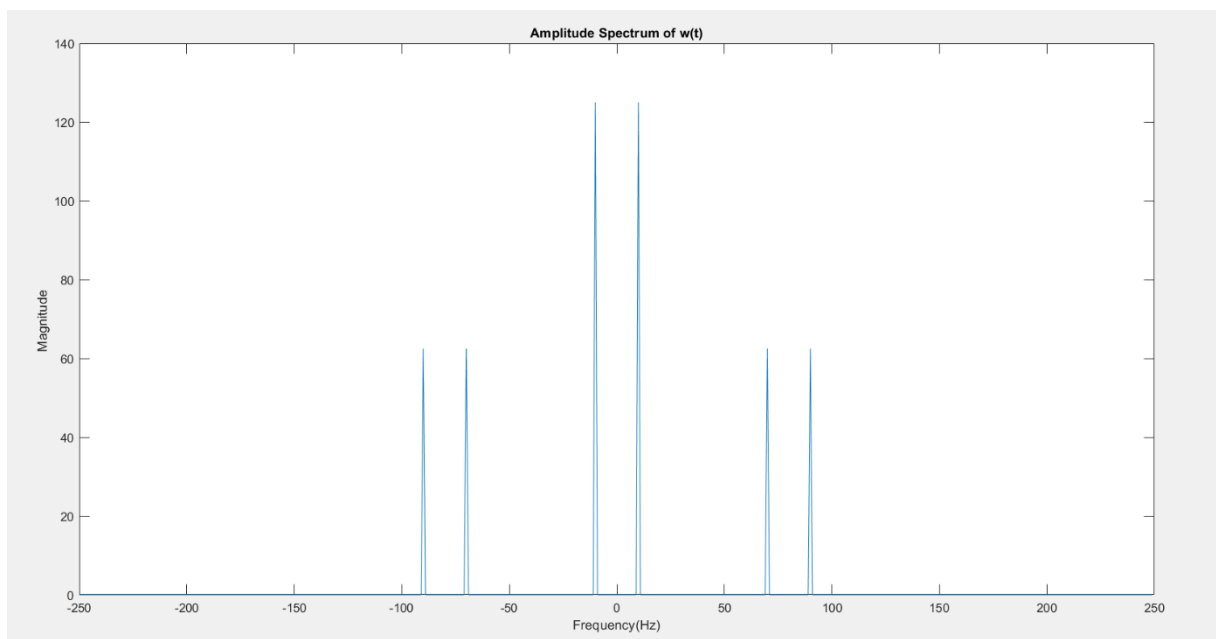
**Kết quả hiện ra màn hình:**

Hình 2.3 mô tả kết quả mô phỏng tín hiệu  $r(t)$  dưới tác động của nhiễu trắng với  $\text{SNR}=10 \text{ dB}$ . Có thể thấy lúc này, dạng tín hiệu  $r(t)$  đã khác biệt so với dạng tín hiệu  $y(t)$  ban đầu. Nếu  $\text{SNR}$  càng nhỏ, sự chênh lệch khác nhau sẽ càng lớn.

Hình 2.4 mô tả kết quả mô phỏng tín hiệu  $w(t)$  khi  $r(t)$  không có nhiễu trắng. Có thể thấy dải tần số của  $w(t)$  bao gồm 10Hz, 70Hz và 90Hz, với biên độ của các cột tần số cao (70Hz và 90Hz) giảm một nửa so với cột tần số 10Hz, thỏa mãn công thức.

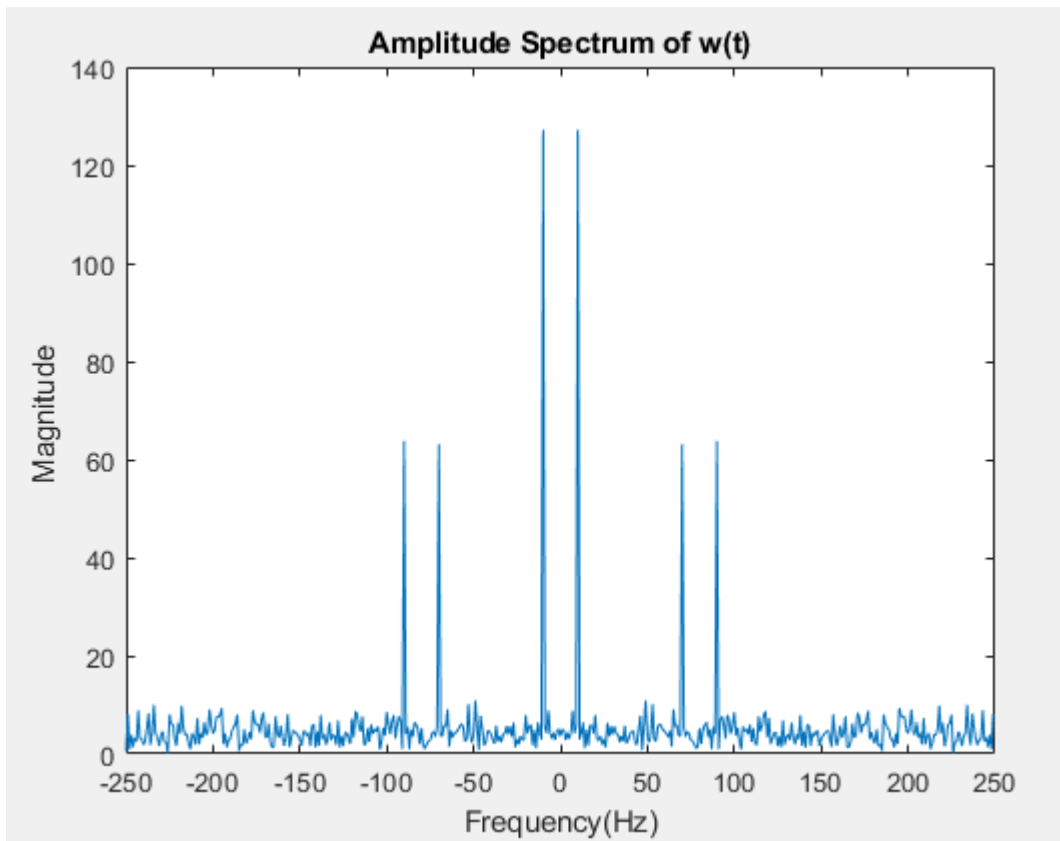


**Hình 2.3 Tín hiệu  $r(t)$  dưới tác động của nhiễu trắng Gauss**



**Hình 2.4 Phổ biên độ của tín hiệu  $w(t)$  trường hợp  $r(t)$  không có nhiễu trắng**

Nếu xét trường hợp có nhiễu với  $\text{SNR}=10\text{dB}$ , trong matlab thay  $w_t=r_t.*x_c$ ; ta được:



**Hình 2.5 Phổ biên độ của tín hiệu  $w(t)$  trường hợp  $r(t)$  có nhiễu**

Hình 2.5 mô tả kết quả mô phỏng tín hiệu  $w(t)$  khi  $r(t)$  có nhiễu trắng với  $\text{SNR} = 10 \text{ dB}$ . Có thể thấy cả khi có nhiễu, dải tần số của tín hiệu vẫn không thay đổi (gồm 10 Hz, 70 Hz và 90 Hz). Tuy nhiên, sóng nhiễu làm xuất hiện dải tần số mới từ -250 Hz tới 250 Hz, với biên độ càng lớn khi SNR càng nhỏ.

#### **2.1.4 Thiết kế bộ lọc thông thấp**

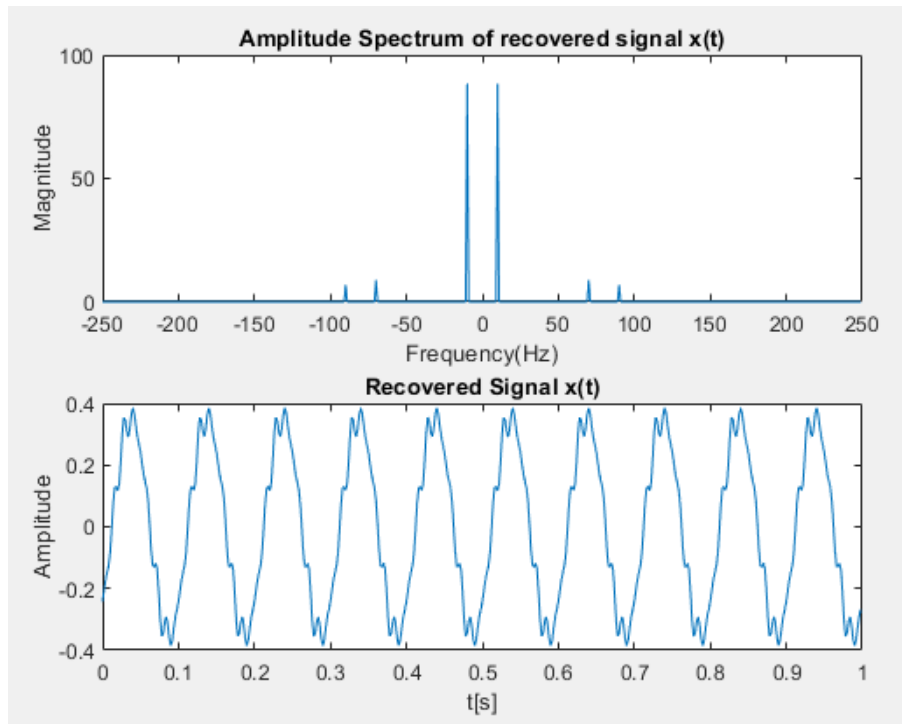
##### **2.1.4.1 Lọc trên miền tần số**

Một bộ lọc thực tế thường có giá trị thấp chứ không bằng 0 với  $H(\omega)$  ở tần số cao và giảm dần độ lớn khi tần số tăng. Ở đây ta sử dụng bộ lọc thông thấp đơn giản nhất nhưng kém hiệu quả nhất là:

$$H(\omega) = \frac{f_c}{f_c + j \cdot f}$$

Với  $f_c$  là tần số cắt,  $f$  là tần số lấy mẫu tín hiệu  $x(t)$ .

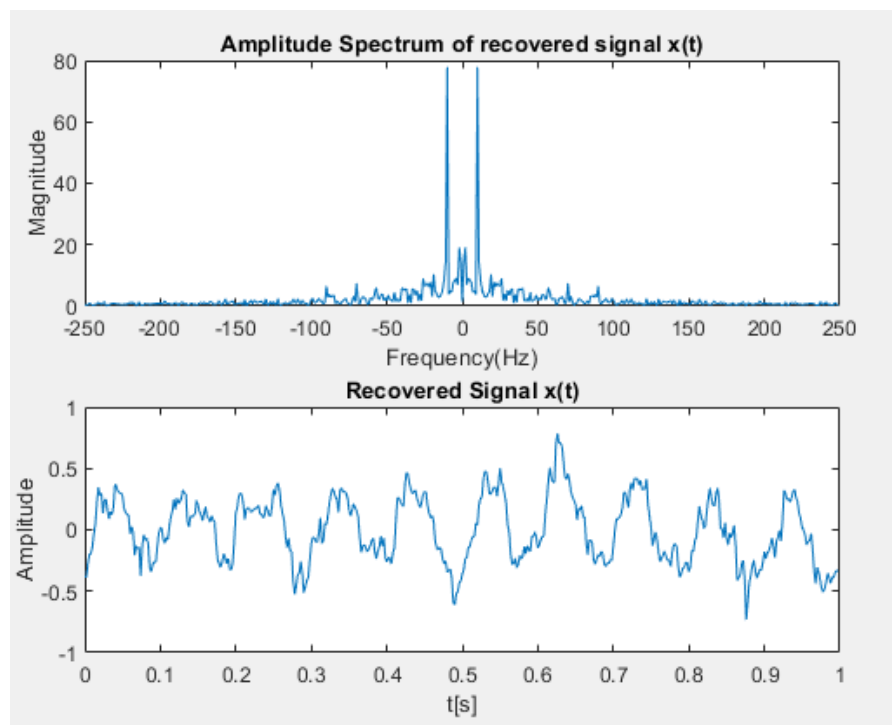
Trường hợp không có nhiễu:



**Hình 2.6 Tín hiệu khôi phục  $x(t)$  bằng lọc trên miền tần số không có nhiễu**

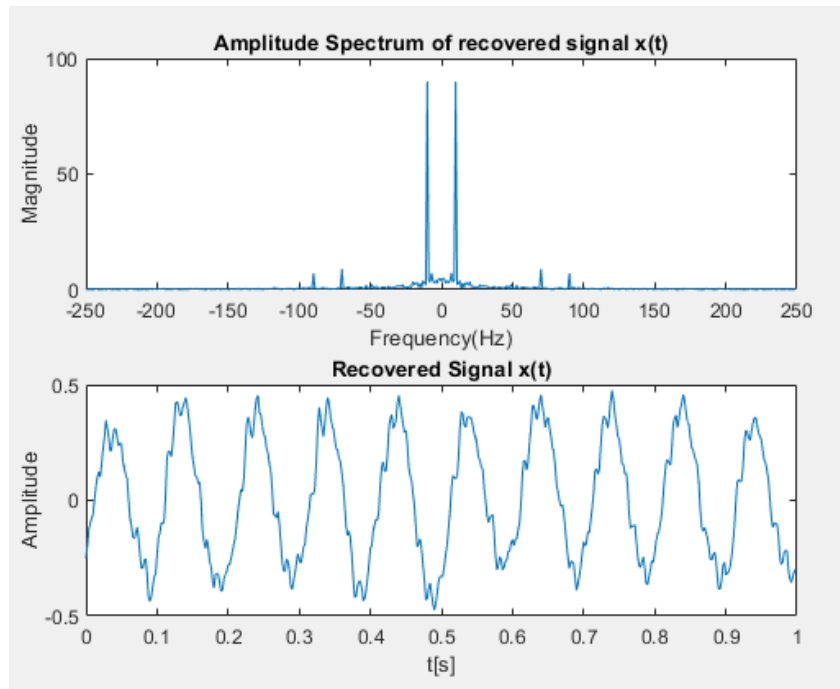
Hình 2.6 mô tả kết quả mô phỏng khôi phục  $x(t)$  bằng lọc thông thấp trên miền tần số không có nhiễu. Có thể thấy tín hiệu  $x(t)$  có dạng tương đối giống  $x(t)$  ban đầu, tuy chưa có dạng hình sin chuẩn, biên độ giảm khoảng một nửa. Cột tần số thể hiện ở hình vẽ là 10 Hz, trùng với tần số ban đầu.

Trường hợp có nhiễu:



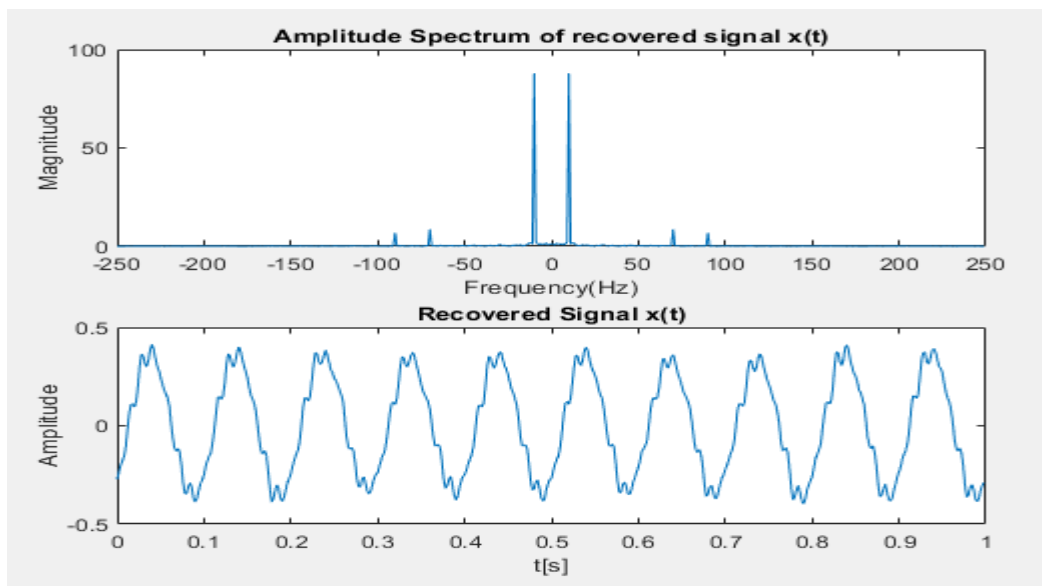
**Hình 2.7 Tín hiệu khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $SNR = 0$**

Hình 2.7 mô tả kết quả mô phỏng khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $\text{SNR}=0$ . Cột tần số 10 Hz rõ ràng, tuy vẫn có nhiều tần số nhiễu đáng kể. Tín hiệu khôi phục  $x(t)$  có dạng khác xa với  $x(t)$  ban đầu, không rõ chu kỳ và dạng hình sin chuẩn.



**Hình 2.8 Tín hiệu khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $\text{SNR} = 10$  dB**

Hình 2.8 mô tả kết quả mô phỏng khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $\text{SNR} = 10$ . Cột tần số 10 Hz rõ ràng, tần số nhiễu không đáng kể. Tín hiệu khôi phục có dạng khác xa với  $x(t)$  ban đầu, chưa rõ chu kỳ và dạng hình sin chuẩn.

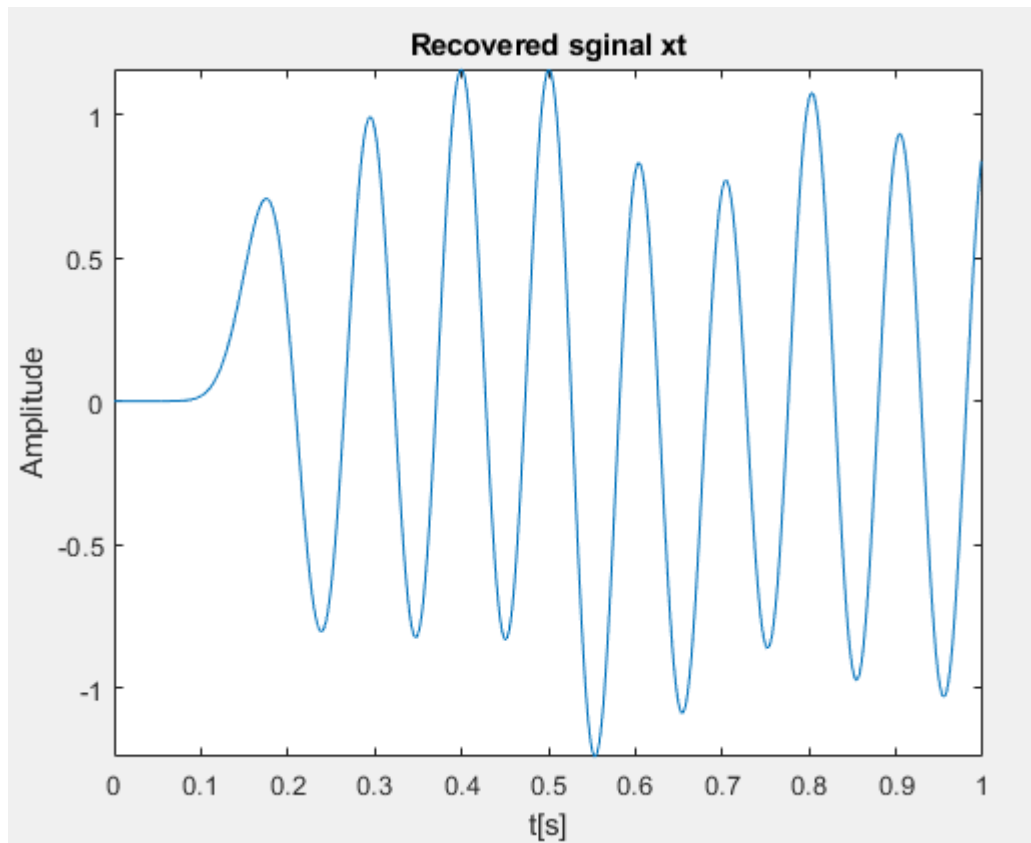


**Hình 2.9 Tín hiệu khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $\text{SNR} = 20$  dB**

Hình 2.9 mô tả kết quả mô phỏng khôi phục  $x(t)$  bằng lọc trên miền tần số có nhiễu với  $\text{SNR} = 20$ . Cột tần số 10 Hz rõ ràng, tần số nhiễu gần như là không có. Tín hiệu khôi phục có dạng khá gần với  $x(t)$  ban đầu, chưa rõ chu kì và dạng hình sin chuẩn.

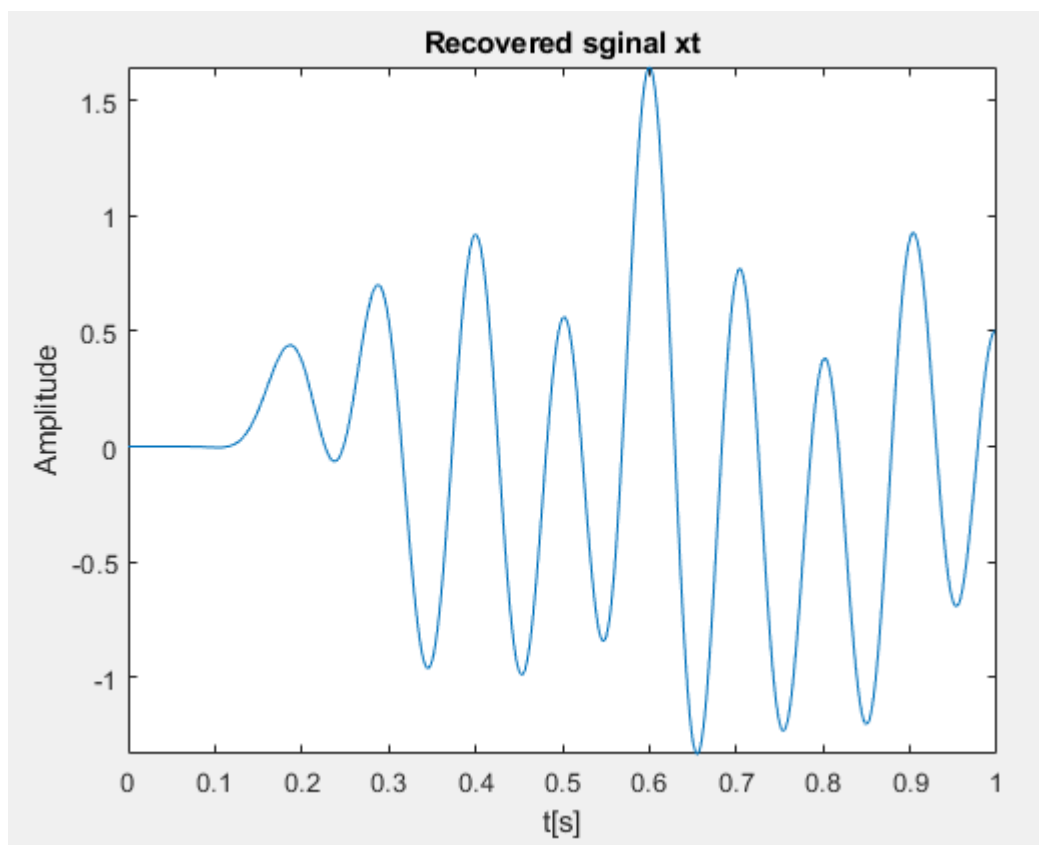
#### 2.1.4.2 Sử dụng hàm Butter có sẵn của Matlab

Trường hợp không có nhiễu:

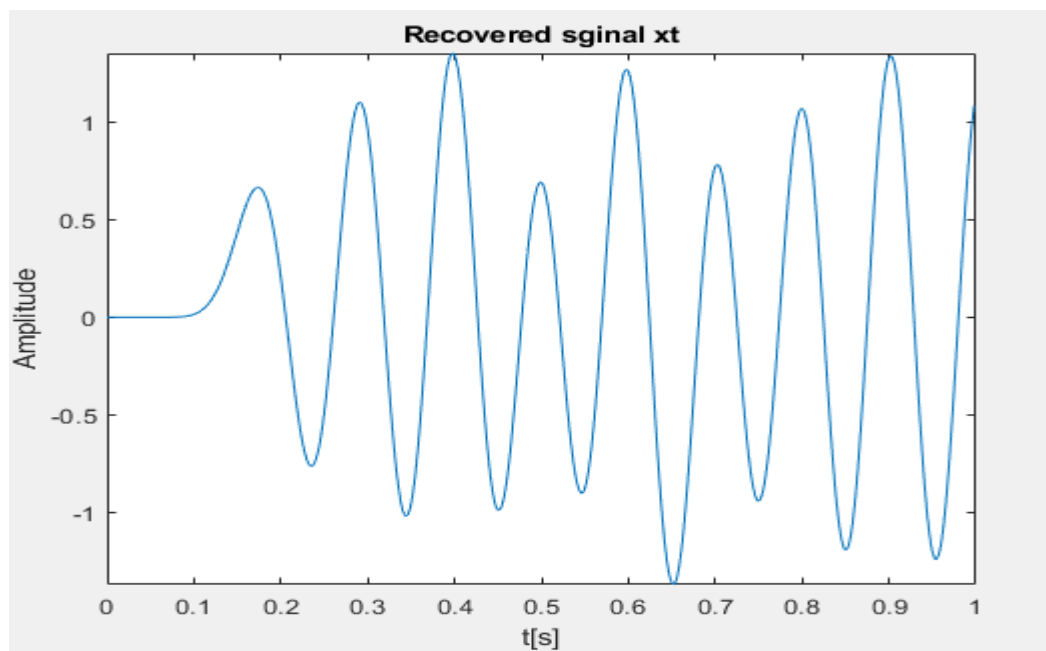


**Hình 2.10 Tín hiệu khôi phục  $x(t)$  sử dụng hàm lọc trên Matlab không nhiễu**

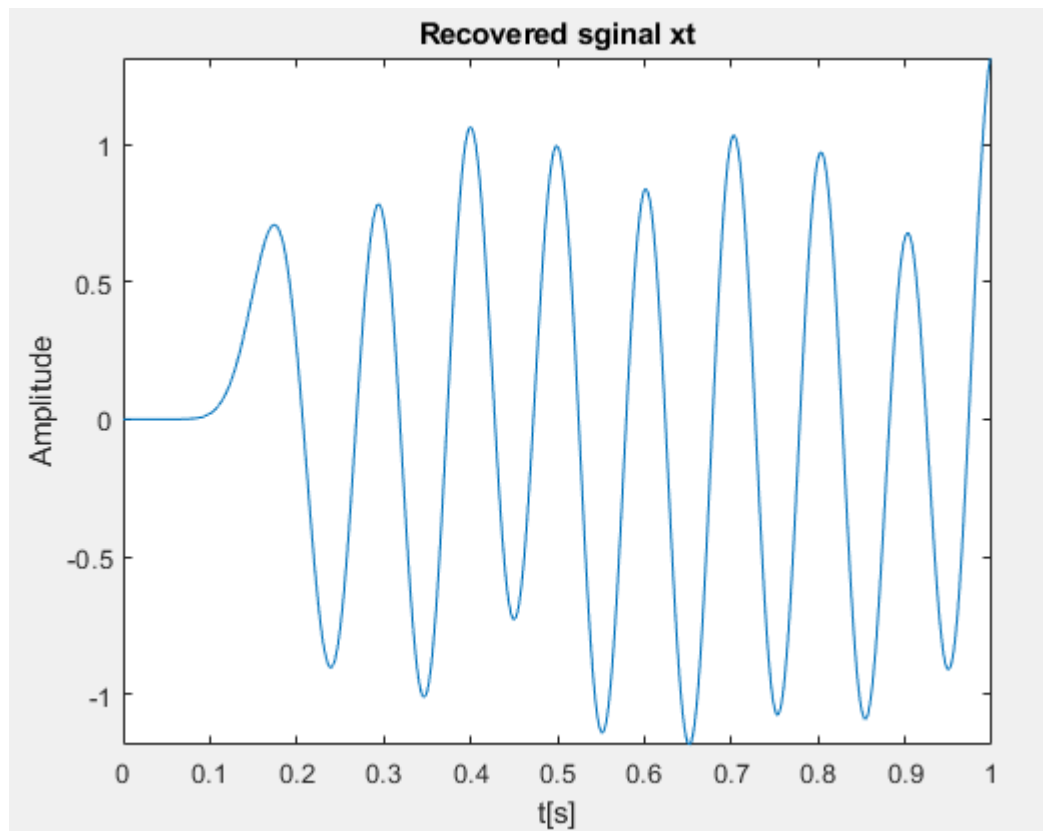
Hình 2.10 cho ta kết quả tín hiệu khôi phục sử dụng hàm `butter()` và `filter()` trên Matlab. Kết quả cho thấy với tần số  $f_c = 10\text{Hz}$  và bậc của bộ lọc sử dụng ở đây là 13 cho ta một tín hiệu hình sin chưa chuẩn xác so với lý thuyết, trường hợp không nhiễu. Ta xét các trường hợp nhiễu với SNR khác nhau ở các hình dưới đây:



**Hình 2.11 Tín hiệu khôi phục  $x(t)$  sử dụng hàm lọc trên Matlab có nhiễu với  $\text{SNR} = 0$**



**Hình 2.12 Tín hiệu khôi phục  $x(t)$  sử dụng hàm lọc trên Matlab có nhiễu với  $\text{SNR} = 10$**



**Hình 2.13 Tín hiệu khôi phục  $x(t)$  sử dụng hàm lọc trên Matlab có nhiễu với  $\text{SNR} = 20$**

Ta thấy, SNR càng lớn thì tín hiệu khôi phục càng tiệm cận trường hợp không nhiễu, đúng như lý thuyết.

## 2.2 Điều chế QAM

### 2.2.1 Vẽ dạng tín hiệu

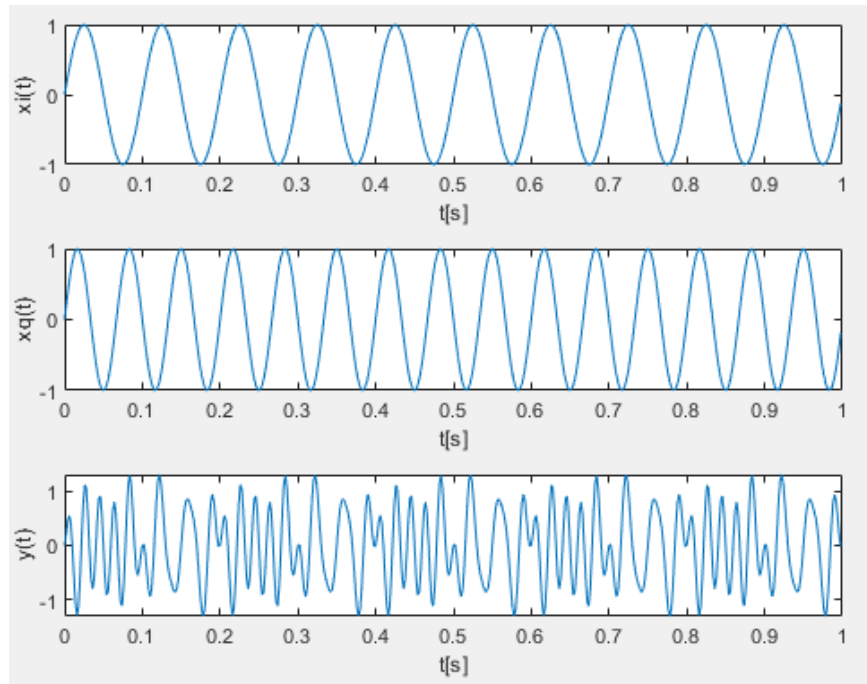
Tín hiệu băng gốc  $x_I(t)$  là một tín hiệu hình sin với tần số 10 Hz và  $x_Q(t)$  là một tín hiệu hình sin với tần số 15 Hz, sóng mang  $f_0 = 40$  Hz. Hãy vẽ dạng tín hiệu  $y(t)$  trong khoảng thời gian: 0 – 1 [s]. Chọn tần số lấy mẫu  $f_s = 500$  Hz.

Vẽ tín hiệu phát của bộ điều chế QAM, với tần số của  $x_i$  là 10 Hz; tần số của  $x_q$  là 15Hz

#### Kết quả:

Hình 2.14 vẽ ra tín hiệu phát đi của QAM là tổng của 2 tín hiệu sau khi điều chế của  $x_i$  và  $x_q$ .

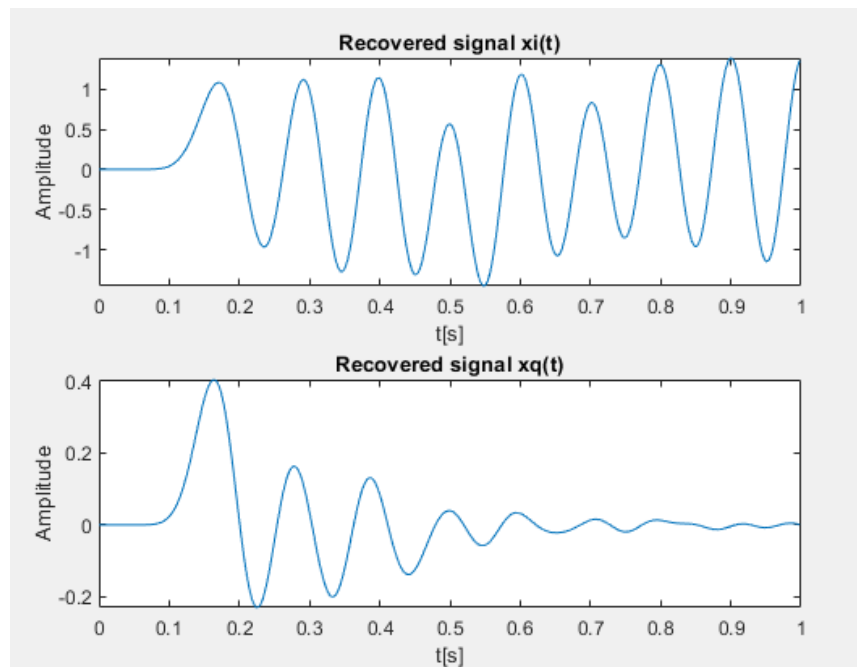




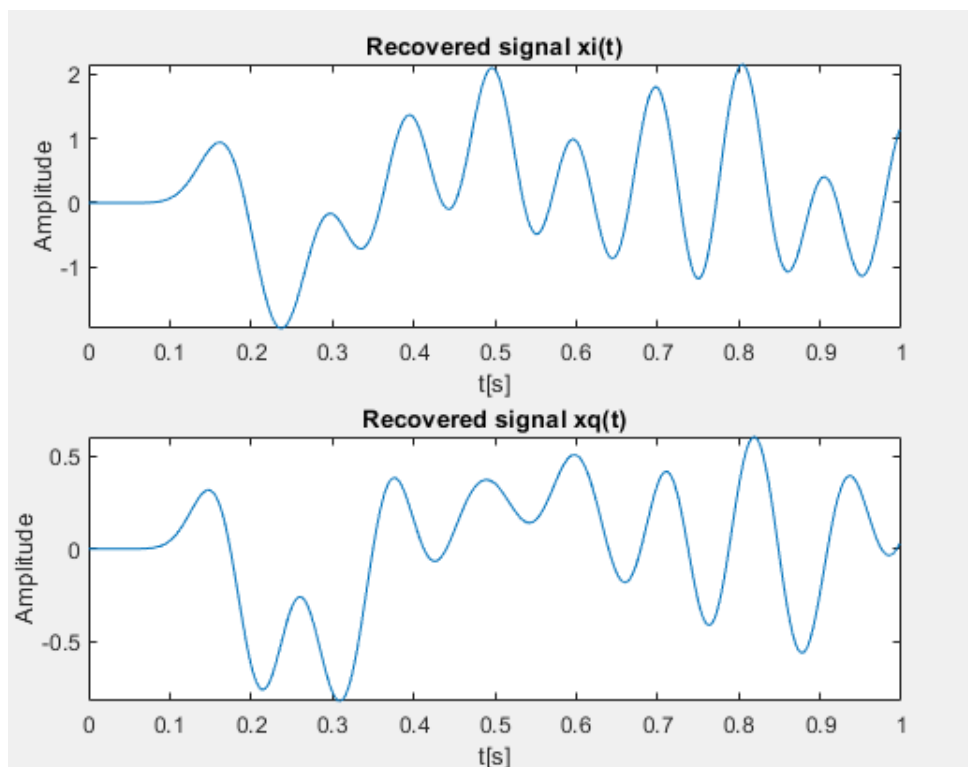
**Hình 2.14 Tín hiệu phát của QAM**

### 2.2.2 Vẽ tín hiệu đầu thu sau lọc

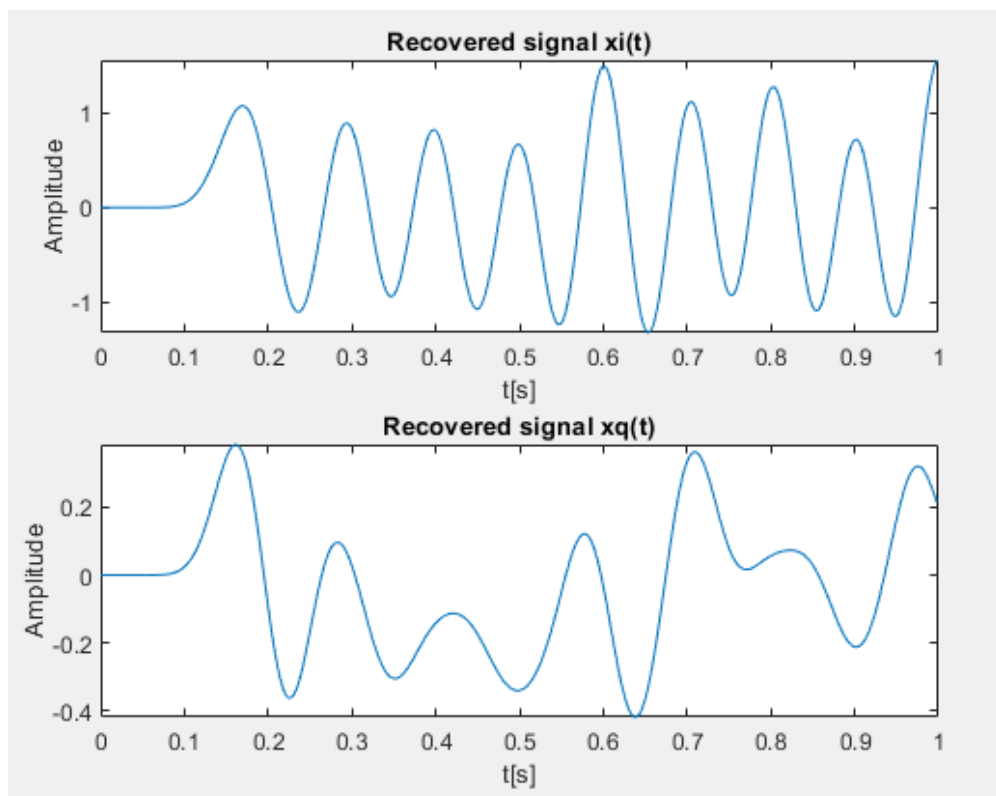
Sử dụng hàm `butter()` và `filter()` của Matlab, ta thu được tín hiệu khôi phục trường hợp không nhiễu.



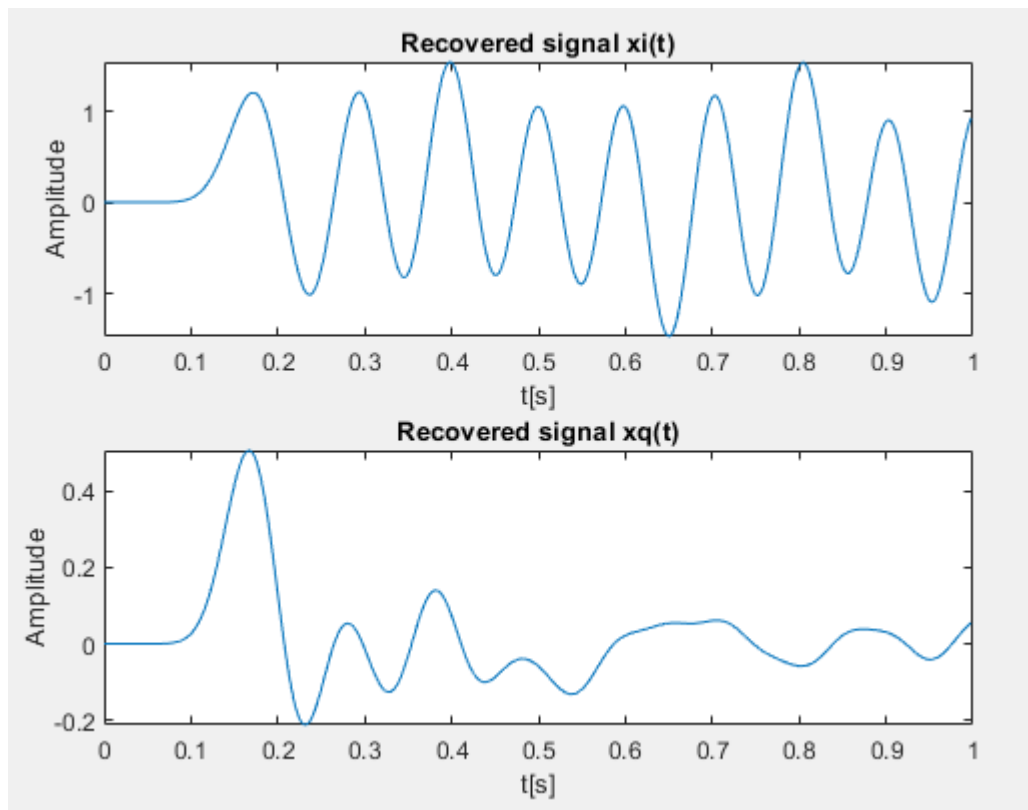
**Hình 2.15 Tín hiệu khôi phục khi không có nhiễu**



**Hình 2.16 Tín hiệu khôi phục khi có nhiễu với  $\text{SNR} = 0$**



**Hình 2.17 Tín hiệu khôi phục khi có nhiễu với  $\text{SNR} = 10$**



**Hình 2.18 Tín hiệu khôi phục khi có nhiễu với SNR = 20**

Dựa vào kết quả, ta nhận thấy rằng khi SNR càng tăng thì dạng tín hiệu khôi phục ngày càng tiệm cận trường hợp không nhiễu. Tuy nhiên tín hiệu  $x(q)$  dần mất dạng hình sin ở bậc bộ lọc lấy là 14.

## 2.3 Kết luận

- Theo lý thuyết biên độ của tín hiệu sau khi điều chế giảm đi 1 nửa. Nhưng kết quả cho thấy tùy vào tần số cắt mà ta chọn (ở đây chọn  $f_c=10\text{Hz}$ ) và có nhiễu hay không có nhiễu mà tín hiệu khôi phục có tỉ lệ sai khác so với tín hiệu băng gốc.
- Một bộ lọc thực tế luôn không lý tưởng và có sự sai khác, việc sử dụng hàm butter và filter trên matlab cũng đem lại kết quả tương tự.
- Với nhiễu trắng Gauss trên kênh truyền, nếu SNR càng lớn thì kết quả càng tốt.