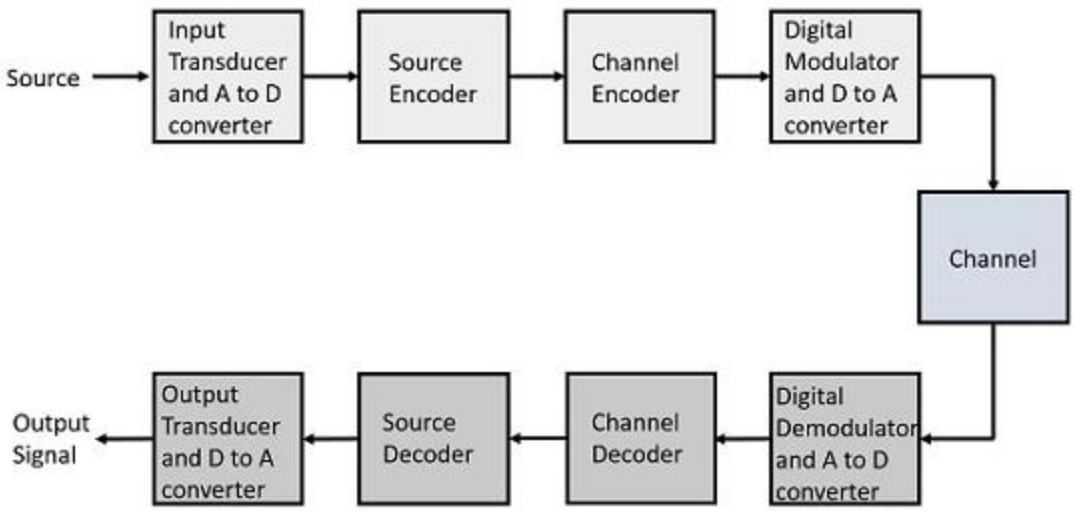
🔬 LAB THÍ NGHIỆM THÔNG TIN SỐ

MÔ PHỎNG HỆ THỐNG TRUYỀN THÔNG SỐ VỚI ĐIỀU CHẾ ASK, FSK và PSK

Biên soạn và hướng dẫn: TS. Trần Văn Líc

Version: 1.0.2-2025



Basic Elements of a Digital Communication System

Mục tiêu

- Hiểu rõ **5 khối chính** trong hệ thống **Digital Communication**:
  1. **Mã hóa nguồn (Source Coding)**
  2. **Mã hóa kênh (Channel Coding)**
  3. **Ghép kênh (Multiplexing) (nếu cần)**
  4. **Điều chế (Modulation)**
  5. **Truyền dẫn & Giải điều chế (Transmission & Demodulation)**
- Thực hành mô phỏng truyền dữ liệu số bằng điều chế số cơ bản **FSK, ASK, PSK.**
- Quan sát ảnh hưởng của **nhiễu** và **mã hóa kênh** đến chất lượng truyền dẫn.

🛠 Yêu cầu phần mềm

- MATLAB / Simulink hoặc Python (NumPy + Matplotlib)

🔹 LAB1: Mã hóa nguồn (Source Coding)

📌 **Mô tả:** Chuyển chuỗi ký tự ASCII thành dạng nhị phân.

📌 **Thực hiện**

Nhập một chuỗi ký tự "HELLO" và chuyển đổi thành nhị phân (mỗi ký tự 8 bit).

📌 **Python Code**

```
text = "HELLO"
binary_data = ''.join(format(ord(c), '08b') for c in text)
print("Binary Data:", binary_data)
```

📌 **Kết quả mong đợi (chụp lại màn hình kết quả)**

Binary Data: .....

🔹 LAB2: Mã hóa kênh (Channel Coding)

📌 **Mô tả:** Thêm mã Hamming (7,4) để sửa lỗi trong kênh truyền.

📌 **Python Code**

```
from commpy.channelcoding import hamming
data = [1, 0, 1, 1]
encoded_data = hamming.encode(data, 3)
print("Hamming Encoded Data:", encoded_data)
```

📌 **Kết quả mong đợi (chụp lại màn hình kết quả)**

Dữ liệu gốc: .....

Mã Hamming (7,4): .....

📌 **Thực hiện lại với mã Hamming (7,11). So sánh và nhận xét so với mã (7,4)**

🔹 LAB 3: Ghép kênh TDM (Multiplexing)

📌 **Mô tả:** Kết hợp hai luồng dữ liệu trên cùng một kênh.

📌 **Python Code**

```
text1 = "HELLO"

text2 = "WORLD"


binary_data1 = ''.join(format(ord(c), '08b') for c in text1)

binary_data2 = ''.join(format(ord(c), '08b') for c in text2)


muxed_data = ''.join([b1 + b2 for b1, b2 in zip(binary_data1, binary_data2)])

print("TDM Multiplexed Data:", muxed_data)
```

📌 **Kết quả mong đợi**

TDM Multiplexed Data: .....

🔹 LAB 4: Điều chế FSK (Modulation)

📌 **Mô tả:** Điều chế dữ liệu đã ghép kênh bằng **FSK, ASK, PSK**

📌 **Python Code**

```
import numpy as np

import matplotlib.pyplot as plt


Fs = 10000

T = 1/Fs

t = np.arange(0, 0.01, T)


fsk_signal = np.array([])


for bit in muxed_data:

    freq = 2000 if bit == '0' else 5000
```

```
    fsk_signal = np.append(fsk_signal, np.sin(2 * np.pi * freq * t))


plt.plot(fsk_signal[:500])
plt.title("TDM FSK Modulated Signal")
plt.show()
```

📌 **Kết quả mong đợi: (chụp lại màn hình kết quả)**

.....
.....
.....
.....

🚀 **Quan sát tần số thay đổi theo từng bit 0 và 1.**

🔹 LAB 5: Giải điều chế & kiểm tra lỗi

📌 **Mô tả:** Thêm nhiễu AWGN, sau đó giải điều chế và kiểm tra lỗi.

📌 **Python Code**

python

CopyEdit

```
noise = np.random.normal(0, 0.5, len(fsk_signal))
rx_signal = fsk_signal + noise


plt.plot(rx_signal[:500])
plt.title("FSK with Noise")
plt.show()
```

🚀 **Quan sát tín hiệu bị méo do nhiễu.**

.....
.....
.....
.....

📌 **Giải điều chế**

```
power_f1 = np.sum(np.abs(np.sin(2 * np.pi * 2000 * t) * rx_signal))

power_f2 = np.sum(np.abs(np.sin(2 * np.pi * 5000 * t) * rx_signal))


received_bit = 0 if power_f1 > power_f2 else 1

print("Received Bit:", received_bit)
```

✅ **Khôi phục lại chuỗi bit ban đầu. Kiểm tra kết quả có đúng với tín hiệu ban đầu hay không ? (chụp lại màn hình kết quả)**

.....
.....
.....
.....

❓ **Câu hỏi sau LAB**

1. **Mã hóa Hamming (7,4)** hoạt động như thế nào để sửa lỗi?
2. Nếu tăng nhiễu trong kênh truyền, điều gì xảy ra với tín hiệu?

◆ LAB 6: ĐIỀU CHẾ ASK

YÊU CẦU LAB: Thực hiện lại tất cả các bài LAB trên với điều chế ASK. So sánh giữa FSK với ASK. Nhận xét ?

◆ LAB 7: ĐIỀU CHẾ PSK

YÊU CẦU LAB: Thực hiện lại với PSK. so sánh giữa FSK với PSK. Nhận xét ?

◆ LAB 8: ĐIỀU CHẾ 16-QAM

YÊU CẦU LAB: Thực hiện lại với PSK. so sánh giữa FSK với 16-QAM. Nhận xét ?