

Fast Ethernet Simulation

This project simulates a basic Fast Ethernet network using C++. It includes key components such as Station and Switch to demonstrate frame transmission and handling in a network.

Project Structure

FastEthernetSimulation/

```
├── common.h      // Definitions of shared types like `Frame`
├── station.h     // Header file for the `Station` class
├── station.cpp   // Implementation of the `Station` class
├── switch.h     // Header file for the `Switch` class
├── switch.cpp   // Implementation of the `Switch` class
├── main.cpp     // Entry point for the simulation
├── station_log.txt // Logs activity of individual stations (generated during runtime)
└── switch_log.txt  // Logs activity of the switch (generated during runtime)
```

Key Classes and Their Responsibilities

1. Frame (Defined in common.h)

- Represents a network frame with the following fields:
 - sequenceNumber: A unique identifier for the frame.
 - source: The source station ID.
 - destination: The destination station ID.
 - payload: The actual data.
 - priority: A number representing the priority of the frame.

2. Station (Defined in station.h and station.cpp)

- Represents a network station capable of sending and receiving frames.
- Responsibilities:
 - Sending frames to the switch.
 - Receiving frames from the switch.
 - Logging station activity.

3. Switch (Defined in switch.h and switch.cpp)

- Represents a network switch responsible for forwarding frames between stations.
- Responsibilities:
 - Managing registered stations.
 - Forwarding frames to the correct destination.
 - Logging switch activity.

How to Run

Prerequisites

- A C++ compiler supporting C++11 or later (e.g., GCC, Clang, MSVC).
- Visual Studio (if using Windows) or any compatible IDE.

Steps

1. Clone or Download the Repository

2. `git clone https://github.com/your-username/FastEthernetSimulation.git`

`cd FastEthernetSimulation`

3. Open the Project

- If using Visual Studio:
 - Open the .sln file or create a new Visual Studio project.
 - Add all .h and .cpp files to the project.
- If using a command line:

`g++ -std=c++11 -o simulation main.cpp station.cpp switch.cpp`

4. Build and Run the Project

- In Visual Studio: Press Ctrl+F5 to build and run.
- On the command line:

`./simulation`

5. Output Logs

- Check the generated station_log.txt and switch_log.txt for detailed activity logs.

Example Workflow

Input

The main.cpp defines a simple workflow:

1. Three stations (SP1, SP2, SP3) are registered with the switch.
2. Two frames are sent:
 - From SP1 to SP2.
 - From SP2 to SP3.
3. The switch processes and forwards these frames to the appropriate stations.

Output

- **Switch Log (switch_log.txt):**
- Forwarding frame 1 from SP1 to SP2

Forwarding frame 2 from SP2 to SP3

- **Station Logs:**
- SP1_log.txt:

Sent frame 1 to switch.

- SP2_log.txt:
- Received frame 1 from SP1.

Sent frame 2 to switch.

- SP3_log.txt:

Received frame 2 from SP2.

Troubleshooting

Common Errors

1. **Undefined Identifier Errors:**
 - Ensure all header files are included in the .cpp files.
2. **Function Signature Mismatch:**
 - Ensure the function declarations in header files match the definitions in implementation files.
3. **File Access Issues:**
 - Ensure the program has write access to generate log files.

Future Enhancements

- Add support for error detection and retransmission of frames.
- Implement advanced features like VLANs and QoS.
- Visualize the network activity using a GUI.

License

This project is licensed under the MIT License. See the LICENSE file for more details.