CS 378: Computer Networks Lab Fall 2024

Lab 3: Hands-On MAC Layer

Daya Swaroop Chavali (22b0939)
Nimish Milind Manware (22b0944)
Jayesh Jadhav (22b1056)
Chandrakant Pradhan (22b1023)

Introduction

Only two nodes implementation is done

In this lab, we will implement the MAC layer using the **CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)** protocol. The goal is to efficiently transmit data between nodes using an audio-based physical layer, built upon the system designed in Lab 2. This MAC layer ensures that multiple nodes can share the medium without data collisions by employing collision avoidance mechanisms.

Frames

Each frame would have following fields.

- **Preamble**
  Similar to the last lab, we use a **preamble** to mark the beginning of the message/frame and signal readiness for transmission. It includes alternating bits followed by two consecutive 1 (i.e. 101010101011). Then there would be field for MAC addresses for sender and receiver each.
- **MAC Address**
  Both will take 2 bits only (destination in a transmitted frame can take values 0,1,2 only and -1 is to be discarded).
- **Length**
  Then there would be the field for length bits (5 bits).
- **Payload**
  This would the the be the actual message to be sent.
- **CRC**
  This would be helpful for error detection and correction.

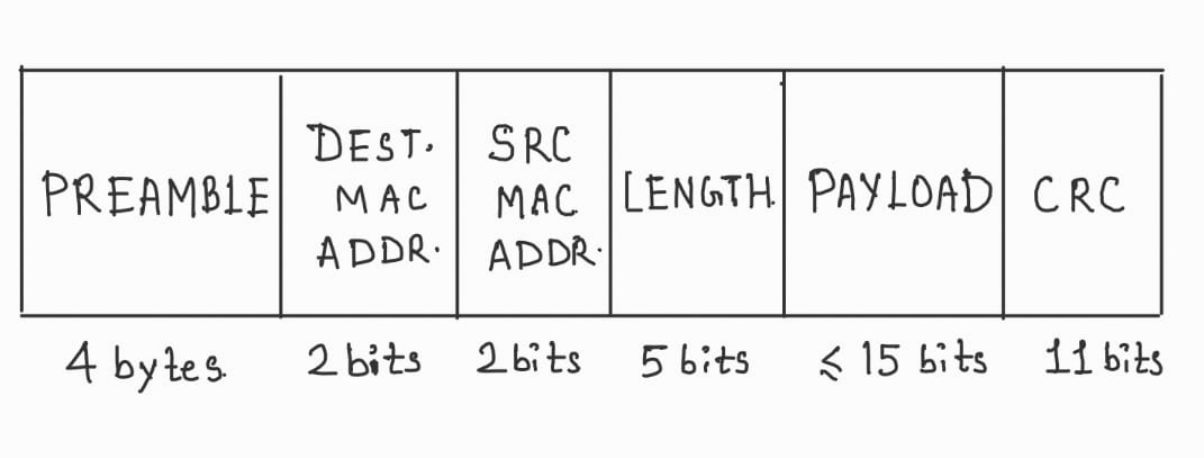

Figure 1: Frame Partioning

MAC Layer Design

1. CSMA/CA Protocol

We will be using the CSMA/CA protocol which prevents the collisions as discussed in the class.

- **Carrier Sensing**: Each node will be listening to the shared medium (audio channel) before transmitting. If the channel is idle, the node proceeds to the backoff phase. If busy, the node waits until the medium becomes free. The carrier sensing would be implemented as follows: Each machine would be actively listening and processing the data which is audio that is being listened by the machine. If there is no significant frequency intensity corresponding to either 0 or 1 for a preset time period,the Carrier is considered free.

- **Collision Avoidance**: After sensing that the medium is idle, the node waits for a random backoff period before transmitting its message. This ensures that if multiple nodes are ready to transmit, they won't collide immediately.

- **Transmission and Acknowledgment**: After waiting for the backoff period, the node transmits its message. After which the node waits for the acknowledgment form the receiver for a preset amount of time (T0).In case of broadcast message the sender would wait for ACK from all the nodes. The receiver sends an acknowledgment (ACK) back, confirming successful reception. If no ACK is received within the waiting period the message is considered to be collided and the message is retransmitted , T0 is set to T0 *2 , and a state variable N (initially 0) indicating number of missed ACK is incremented by 1. If N isgreater than a maximum, the connection is deemed lost and all the messages whose to address is that particular device would be removed from the queue .

This protocol avoids collisions before they occur, particularly useful in wireless (audio-based) networks where direct collision detection is difficult.

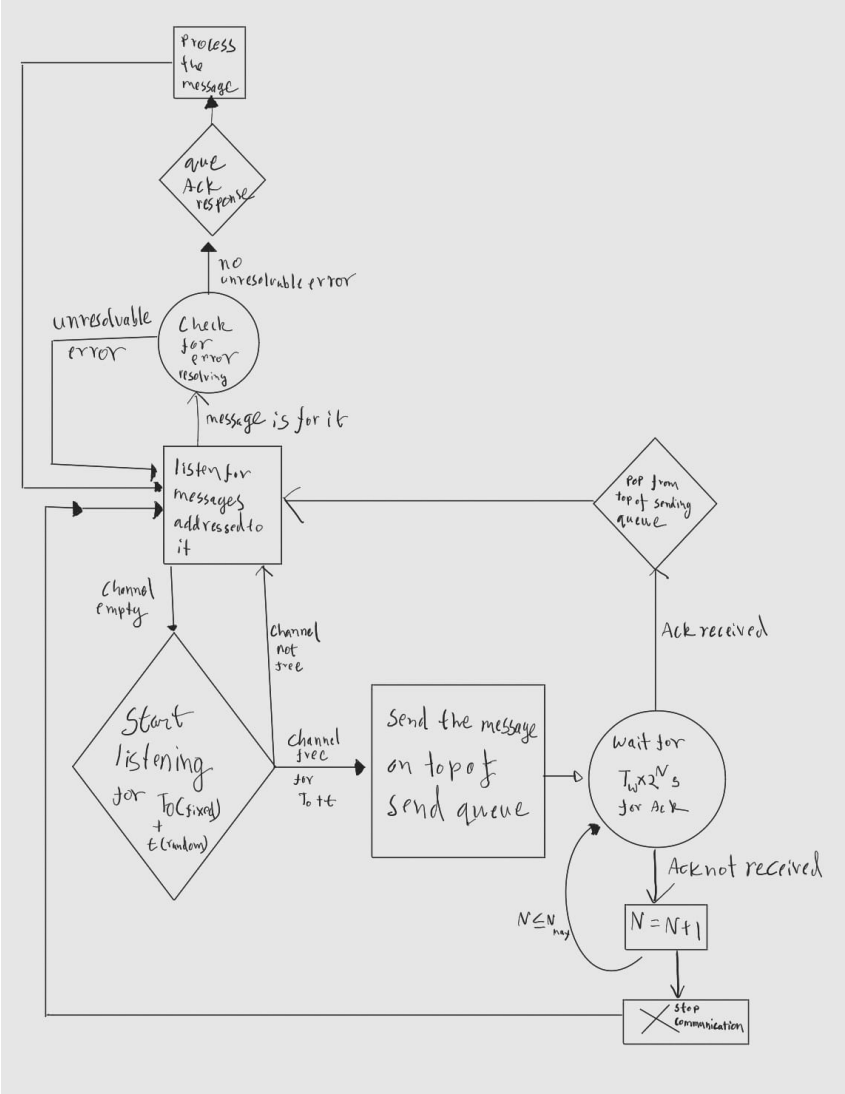

Figure 2: Enter Caption

2. Node Setup

- Each node is assigned a unique MAC address (either 1, 2). These MAC addresses are used to identify the source and destination of the message. The assignment will be done by changing a variable in the program.
- Each node listens to the medium, waits for its turn if it has a message to send, and ensures no simultaneous transmission.

3. Input and Output Format

- **Input**: The input will be stored in a queue at the start of the program. When the trigger is sent the message should be added to a ready to send queue. This will be attempted to be sent immediately if possible according to MAC protocol.
- **Output**: When a message is successfully sent or received, the node prints a timestamped log in the format [SENT] or [RECVD] along with the MAC addresses and the message.

4. Message Receiving Procedure

- Each node actively listens for the messages in the channel and whenever a channel is appeared to be active it start to process the information on the channel. It sees that if the message is for that node or not and then proceeds to check for the error presence in the message with the help of CRC.
- After verifying that the message is correct it sends the Acknowledgement accordingly. The Acknowledgement will consists of only the MAC Address of the sender and the receiver. The Acknowledgement transmission would also follow the message transmission protocol.

5. Handling Collisions

Although collisions are unlikely due to the avoidance mechanisms, if two nodes start transmission simultaneously (due to simultaneous backoff periods), retransmission is triggered. The backoff time is randomized to prevent repeated collisions as described earlier.

6. Timing and Synchronization

- All nodes synchronize their clocks using NTP before the experiment to ensure consistency in timestamps.
- The backoff intervals are adjusted dynamically to account for changes in network congestion.

7. Error Detection and Correction

The message includes a CRC for error detection. In case the CRC check fails, the message is discarded (and no ACK sent), and retransmission is triggered.