1. System Design:

The main structure consists of packet generation, sender, switch, and receiver tasks.

Packet Generation Task:

Generates a packet with a random length between [L1, L2] and a fixed header that contains a random destination, sender ID, and sequence number. After identifying all packet parameters and passing them to the dynamically allocated object, we pass this packet to a node that will be added to the front of a linked list modeling the transmission buffer between the packet generation task and the sender task. This task is controlled by a random time delay between [T1,T2], which manages the time between two packet generations.

Sender Task:

Sends packets to the queue between the sender and the switch, waiting for acknowledgment from the receiver before removing the packet from the transmission buffer. If the ACK does not arrive within the timeout Tout, it will retransmit this packet. After four retransmissions without the received ACK, it will free this packet and focus on the next one.

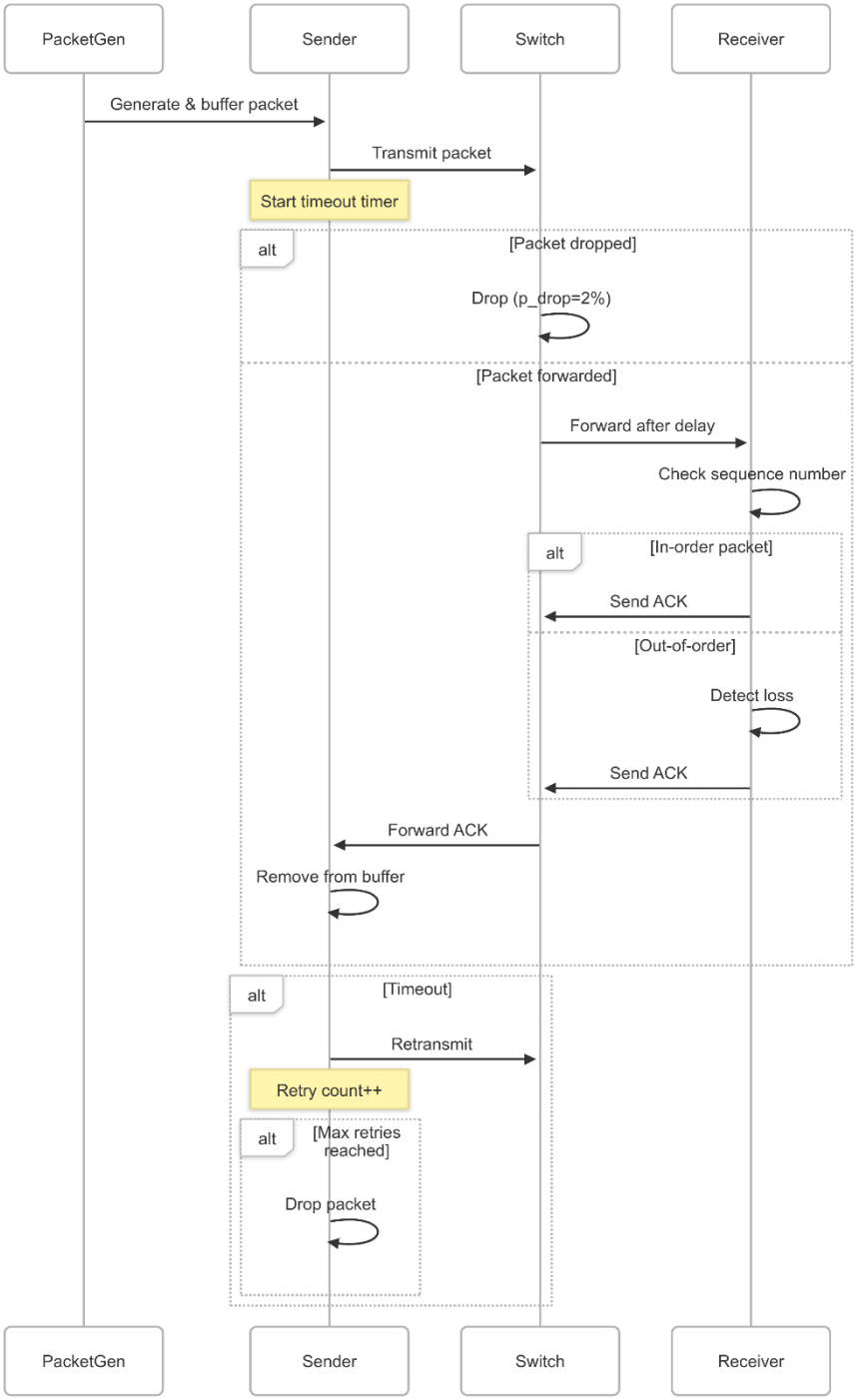
Switch Task:

Generates randomly the transmission and propagation delays and the losses in the link for the packet and the acknowledgment. Receives the packet from the sender by a queue and sends it to its destination receiver by a specific queue (each receiver has a specific queue between it and the switch). Receives the ACK from the receiver and sends it to the sender to make sure that the packet has arrived.

Receiver Task:

Checks the sequence of the received packet to determine whether it matches the expected one. If it arrived correctly, it sends an ACK to the sender. If it is larger, it indicates that a packet or some packets were lost; the receiver waits for the new packet and sends its ACK. If it is smaller, this means it is duplicated; the receiver ignores it and sends an ACK for the older one (expected - 1) to inform the sender to disregard this packet.

The sequence diagram of the system:



Data Structures Used:

Packet		
uint32_t	seq_num	
uint16_t	length	
uint8_t	sender_id	1 or 2
uint8_t	dest	3 or 4
uint8_t	payload	variable length

Packet

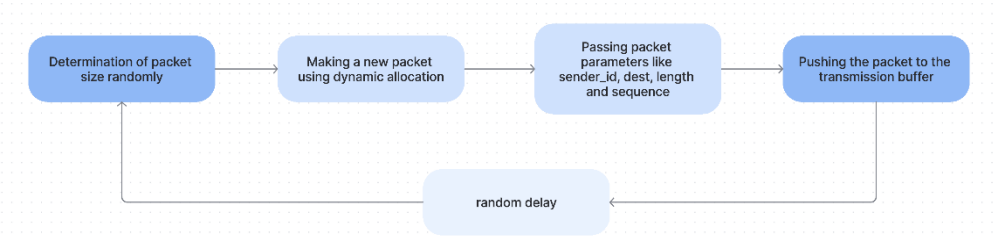
Ack_Packet		
uint32_t	seq_num	
uint8_t	sender_id	3 or 4
uint8_t	dest	1 or 2
uint8_t	padding	fills to 40 bytes

ACK

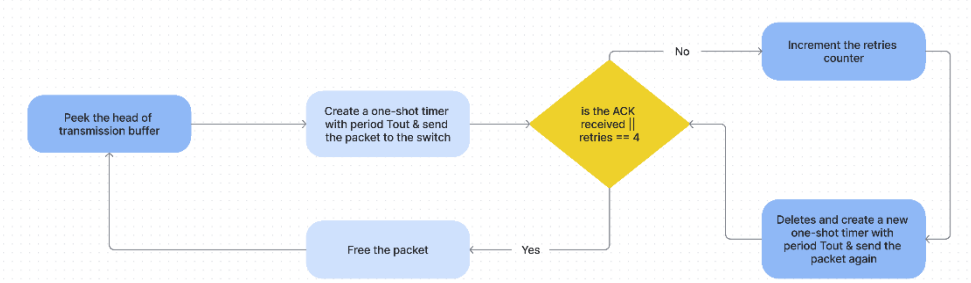
BufferNode	
Packet	pointer
pkt	uint8_t
retries	TimerHandle_t
timer	BufferNode
pointer	next

BufferNode

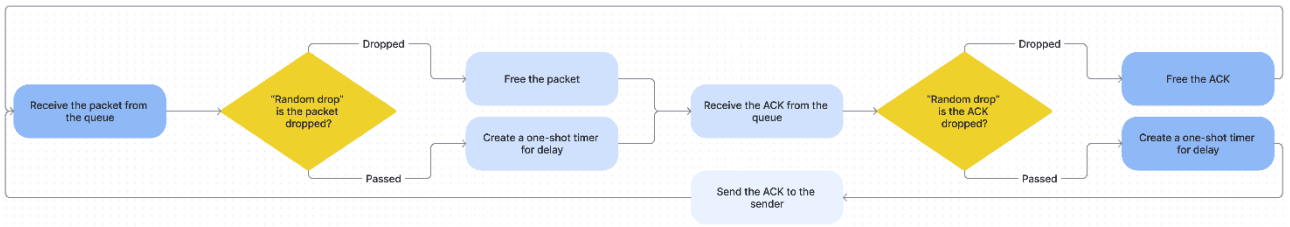
Packet Generation Task Flowchart:



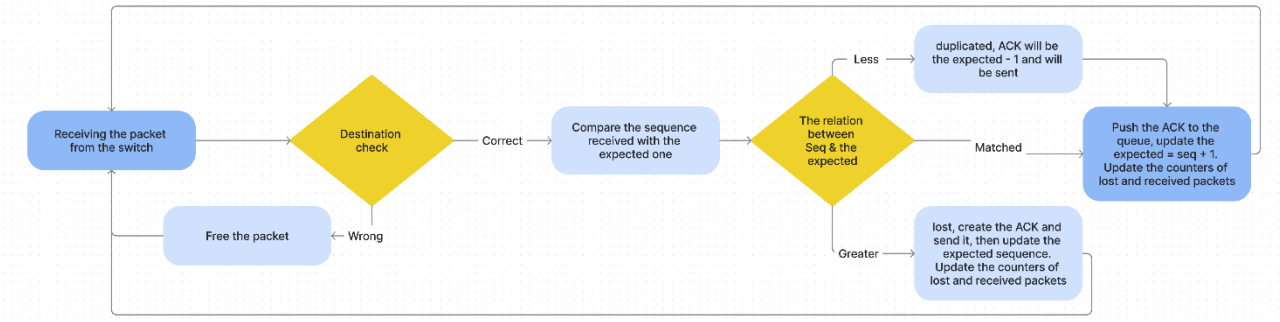
Sender Task Flowchart:



Switch Task Flowchart:



Receiver Task Flowchart:

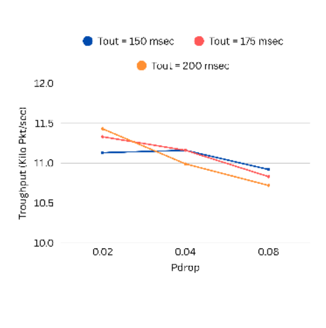


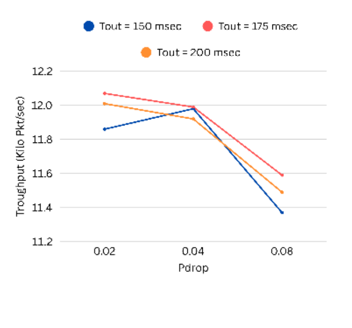
2. Results:

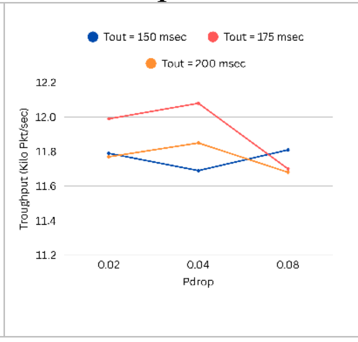
Tout	Pdrop	Throughput	Dropped packets due to retransmission limit	Avg transmission per packet
N = 1				
150 msec	0.02	11.13KP/sec	0	1.48
150 msec	0.04	11.16KP/sec	0	1.49
150 msec	0.08	10.92KP/sec	2	1.49
175 msec	0.02	11.33KP/sec	0	1.02
175 msec	0.04	11.16KP/sec	0	1.06
175 msec	0.08	10.83KP/sec	0	1.11
200 msec	0.02	11.43KP/sec	0	1
200 msec	0.04	10.99KP/sec	0	1.06
200 msec	0.08	10.72KP/sec	0	1.10
N = 2				
150 msec	0.02	11.86KP/sec	0	1.45
150 msec	0.04	11.98KP/sec	0	1.46
150 msec	0.08	11.37KP/sec	0	1.56
175 msec	0.02	12.07KP/sec	0	1.05
175 msec	0.04	11.99KP/sec	1	1.07
175 msec	0.08	11.59KP/sec	0	1.15
200 msec	0.02	12.01KP/sec	0	1.05
200 msec	0.04	11.92KP/sec	0	1.07
200 msec	0.08	11.49KP/sec	0	1.16
N = 4				
150 msec	0.02	11.79KP/sec	0	1.44
150 msec	0.04	11.69KP/sec	0	1.5
150 msec	0.08	11.81KP/sec	0	1.53
175 msec	0.02	11.99KP/sec	0	1.05
175 msec	0.04	12.08KP/sec	0	1.07
175 msec	0.08	11.70KP/sec	0	1.13
200 msec	0.02	11.77KP/sec	0	1.05
200 msec	0.04	11.85KP/sec	0	1.08
200 msec	0.08	11.68KP/sec	0	1.13
N = 8				
150 msec	0.02	12.13KP/sec	0	1.04
150 msec	0.04	12.08KP/sec	0	1.08
150 msec	0.08	11.74KP/sec	0	1.12
175 msec	0.02	12.13KP/sec	0	1.05
175 msec	0.04	11.89KP/sec	0	1.08
175 msec	0.08	11.89KP/sec	0	1.14
200 msec	0.02	11.91KP/sec	0	1.05
200 msec	0.04	11.97KP/sec	0	1.05
200 msec	0.08	11.75KP/sec	2	1.12

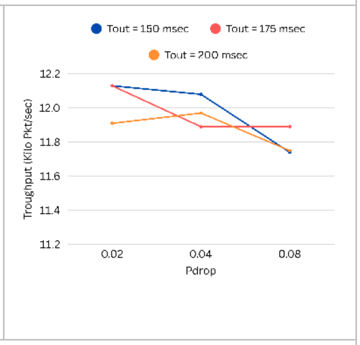
N = 1	N = 2	N = 4	N = 8

Throughput Vs Pdrop

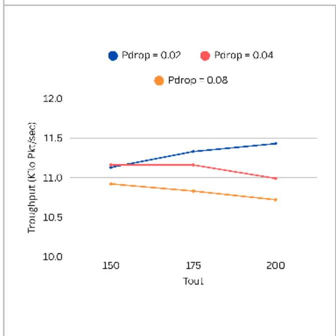


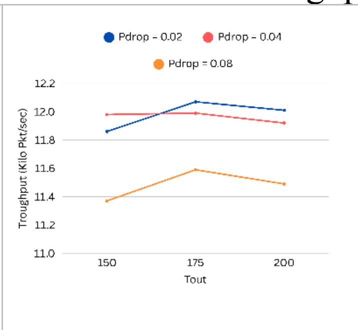


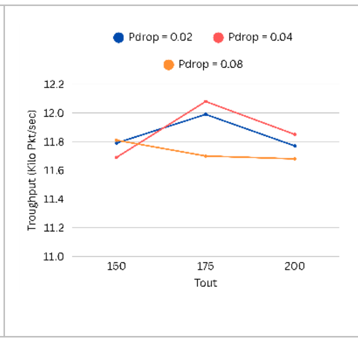


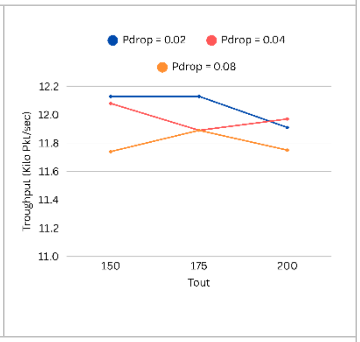


Throughput Vs Tout

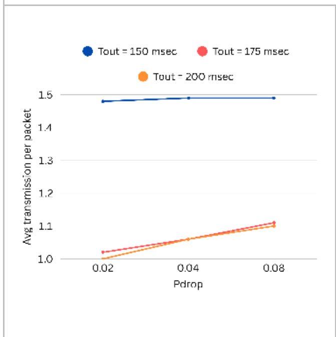


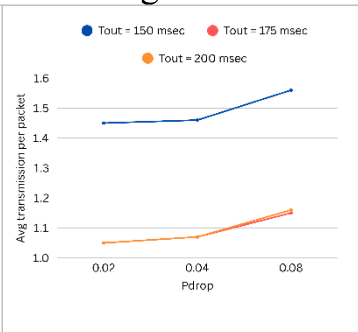


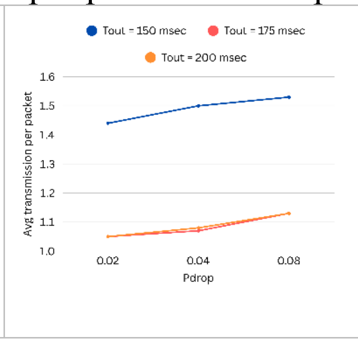


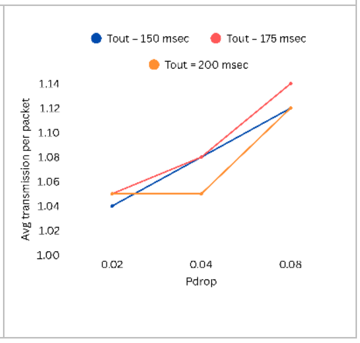


Average transmission per packet Vs Pdrop









3. Results insights:

For send-and-wait with N=1:

- Throughput decreases as Pdrop increases.
- Tout (175–200 ms) maximizes throughput at moderate Pdrop.
- Very small Tout wastes retransmissions, while very large Tout acts as an idle channel.

Go-Back-N (N>1):

- Higher throughput than in S&W, especially at moderate Pdrop.
- Gains diminish beyond N almost equal to 8, since the link becomes saturated.
- Optimal Tout shifts downward with larger N (less waiting time needed when multiple ACKs might return in rapid succession).

Note: All above results where under the condition of no propagation delay (D = 0). However, we can confirm that the system functions under D = 5 ms. We found that the effects of adding a propagation delay is a lower throughput (around 8 to 9 KPps), and higher number of Dropped packets due to retransmission limit (reaching 150 in some tests)