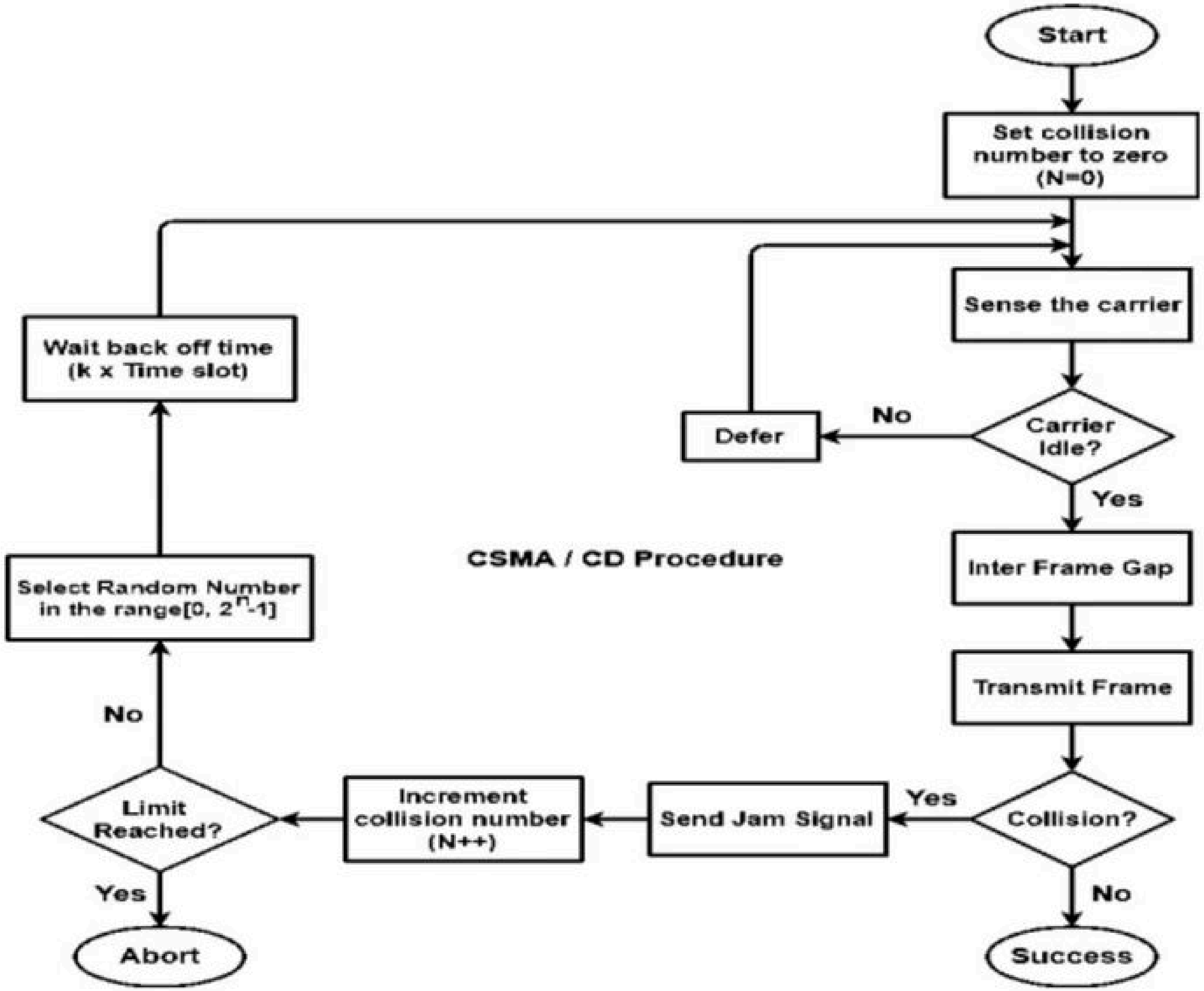
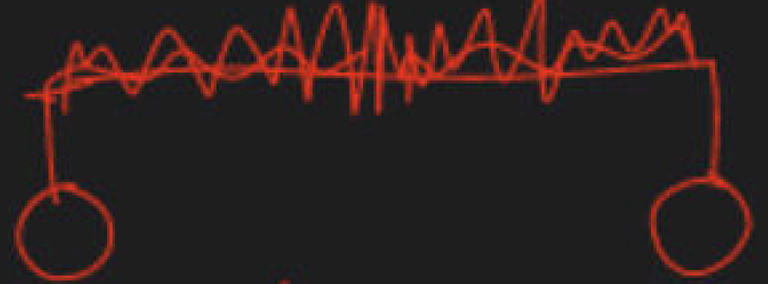
IPV6, WIFI & Doubt Clearing Session

Complete Course on Computer Networks - Part I

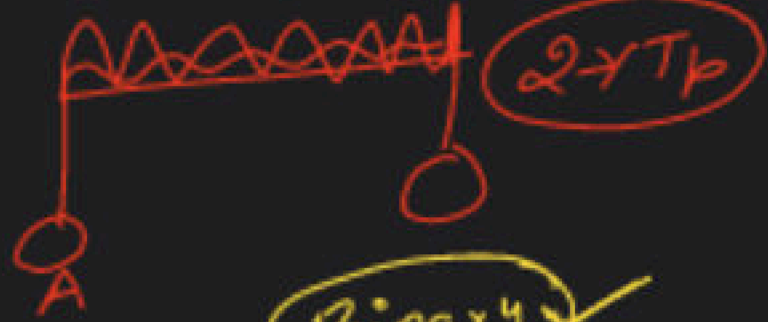
Ravindrababu RAVULA • Lesson 25 • Feb 17, 2021

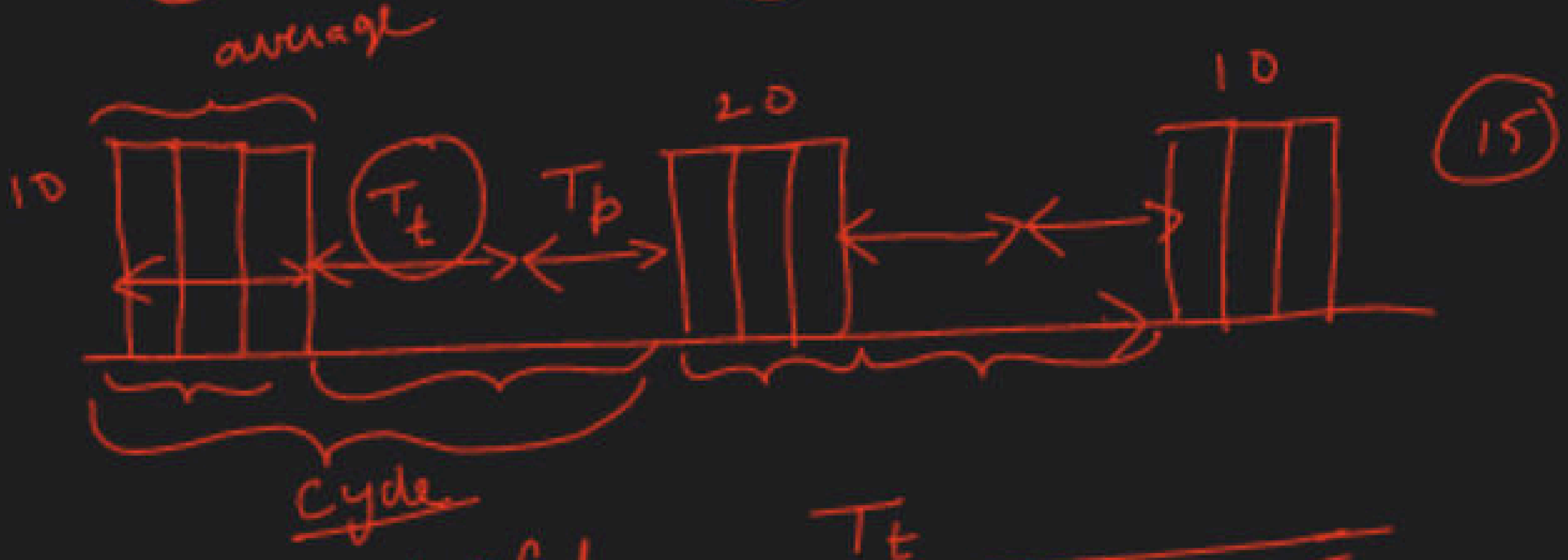

Start
Set collision number to zero (N=0)
Sense the carrier
Carrier Idle?
No
Defer
Yes
Inter Frame Gap
Transmit Frame
Collision?
Yes
Send Jam Signal
Increment collision number (N++)
Limit Reached?
No
Select Random Number in the range[0, 2^n-1]
Wait back off time (k x Time slot)
Yes
Abort
No
Success
CSMA / CD Procedure



$BO \rightarrow WT \quad K \times (0 - 2^{n-1})$

$K \times T_{slot}$





Binary

A	B	C
$n=1$	$n=3$	$n=10$
K	K	K
WT	WT	WT

$$\eta = \frac{\text{useful}}{\text{cycle}} = \frac{T_t}{C * 2 * T_p + T_t + T_p}$$

C

p'

Average

repeat

every station $\rightarrow$ p

$n \rightarrow$ stations

"p"

$n \rightarrow 1 \quad (n-1)$

tries stop

$$P_{succ} = \boxed{\left({}^{n}C_1 * p * (1-p)^{n-1}\right)}$$

$$+ \; {}^{n}C_2 * p^2 * (1-p)^{n-2}$$

n coins – p – H

$P_{succ} =$

$$\eta = \frac{T_t}{2.71 * 2 * T_p + T_p + T_t + \ldots}$$

$$\eta = \frac{1}{1+6.44a} \qquad a = \frac{T_p}{T_t}$$

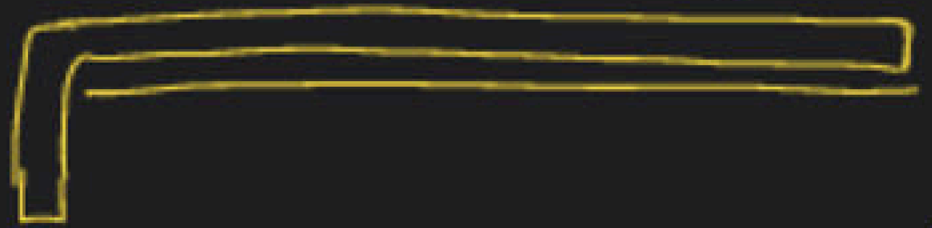
$$\eta = \frac{1}{1+6.44\left(\frac{d}{v}\right)*\left(\frac{B}{L}\right)}$$

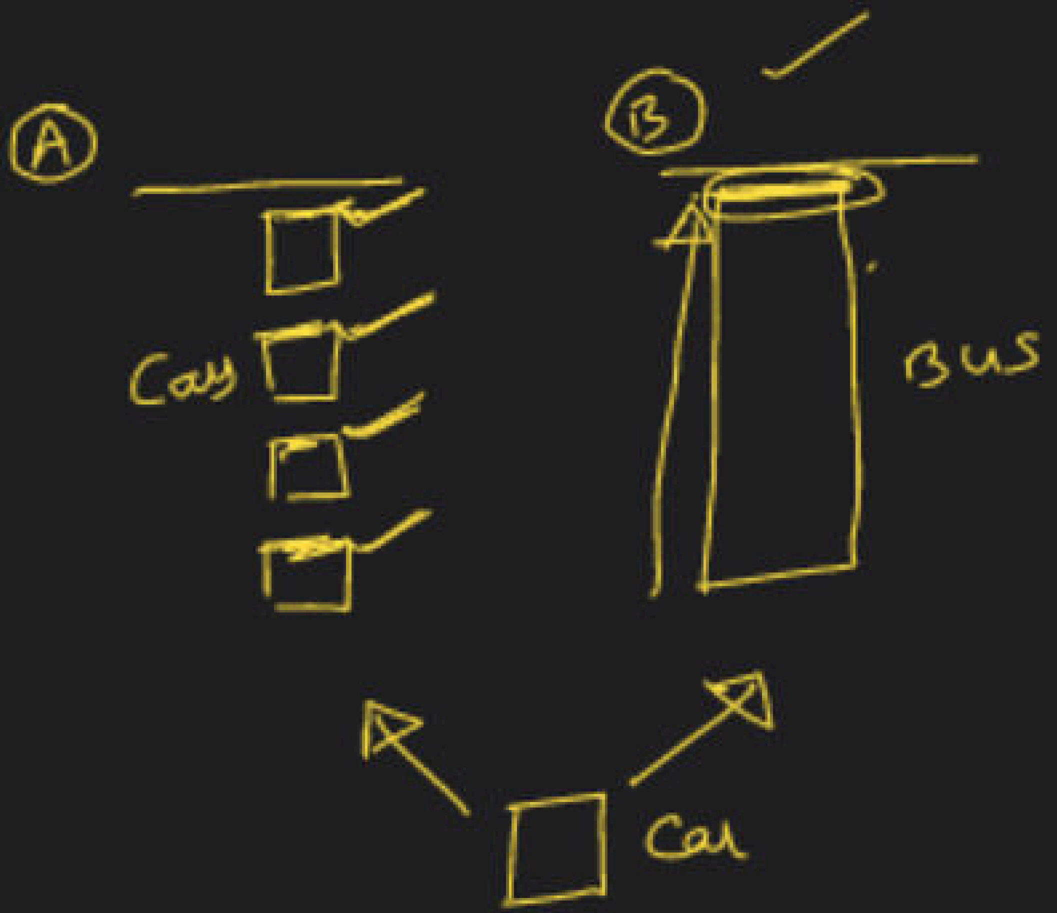
$d\uparrow \rightarrow \eta\downarrow$

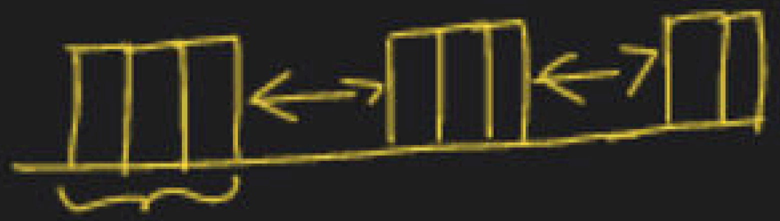
$\Rightarrow$ CSMA/CD

LANS

× WANS







Ethernet $\rightarrow$ Lan

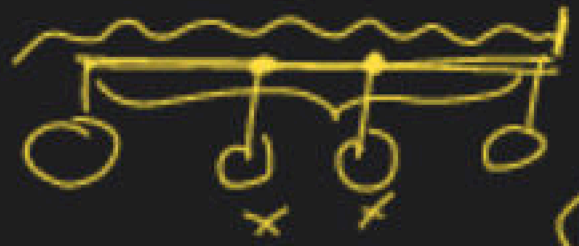
$L\uparrow \rightarrow \eta\uparrow$

10GB

$L\uparrow \rightarrow T_t$

$$\eta = \frac{T_t}{(\qquad)}$$

monopoly



1

limit $\rightarrow$ Size

$\eta\uparrow$

$L\uparrow$

Certain

Now, let us find the maximum value of $P_{successful\ transmission}$.
For maximum value, we put-

$$\boxed{\frac{dP_{successful\ transmission}}{dp} = 0}$$

On solving,

At $p = 1/n$, we get the maximum value of $P_{successful\ transmission}$

Thus,
$(P_{successful\ transmission})_{max}$
$= {}^nC_1 \times 1/n \times (1 - 1/n)^{n-1}$
$= n \times 1/n \times (1 - 1/n)^{n-1}$
$= (1 - 1/n)^{n-1}$

$(P_{successful\ transmission})_{max} = \boxed{(1 - 1/n)^{n-1}}$

$P_{succ} = n * p^1 * (1-p)^{n-1}$

$\boxed{\frac{d P_{succ}}{d p} = 0}$

$\frac{d^2 P_{succ}}{d p^2} < 0 \rightarrow$

$p = 1/n$

$P_{succ} \rightarrow$ max.

$= \cancel{n} * 1/\cancel{n} * \boxed{(1 - 1/n)^{n-1}}$

$n \rightarrow$ no of stations

$C \rightarrow$ collions

"p" $\rightarrow$ station

"n" $\rightarrow$ very large of stations

If there are sufficiently large number of stations i.e. $n \to \infty$, then we have-

$$\lim_{n\to\infty} \left(P_{\text{successful transmission}}\right)_{max} = \lim_{n\to\infty} \left(1 - \frac{1}{n}\right)^{n-1}$$

$$= \frac{1}{e}$$ → limit

Number of times a station must try before successfully transmitting the data packet

$= 1 / P_{max}$ (Using Poisson's distribution)

$= 1 / (1/e)$

$= e$

Average number of collisions that might occur before a successful transmission = e

∞ lt n→∞ ∞ → Suff large # stations

"p", "n" e 2.71

Prob succ = 1/e

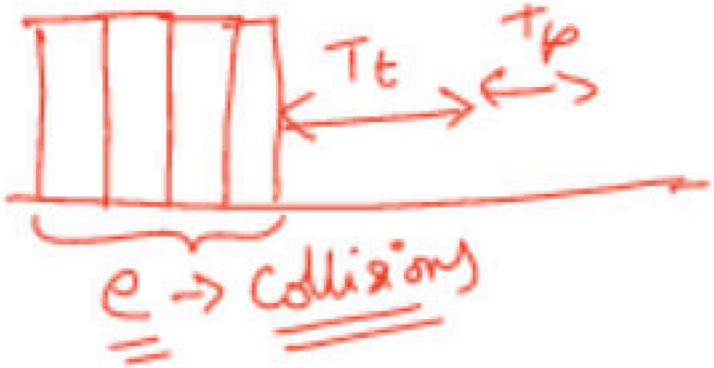
Poi dis: p → H ⇒ ? I head

1 coin → H → p

? I head H + TH + TTH + TTTT4

1/p times before Head.

$\frac{P_{succ} = 1/e}{I\ succ}$ = "e"



$\boxed{c = e}$

$\boxed{c = e}$

succ → Tn
fail → collins

After e tries

average → large

n c = e

"p"

Important Notes-

Note-01:

- CSMA / CD is used in wired LANs.
- CSMA / CD is standardized in IEEE 802.3

ether

Note-02:

CSMA/CD x CP.

- CSMA / CD only minimizes the recovery time.
- It does not take any steps to prevent the collision until it has taken place.

Important Formulas-

- Condition to detect collision: Transmission delay >= 2 x Propagation delay
- Minimum length of data packets in CSMA / CD = 2 x Bandwidth x Distance / Speed
- Efficiency of CSMA / CD = 1 / (1 + 6.44 x a) where $a = T_p / T_t$
- Probability of successful transmission = ${}^nC_1 \times p \times (1-p)^{n-1}$
- Average number of collisions before a successful transmission = e

$\left(\frac{1}{1+6.44a}\right)$

e

Back off algorithm for CSMA/CD

Back Off Time-

In CSMA / CD protocol,

- After the occurrence of collision, station waits for some random back off time and then retransmits.
- This waiting time for which the station waits before retransmitting the data is called as **back off time**.
- Back Off Algorithm is used for calculating the back off time.

Back Off Algorithm-

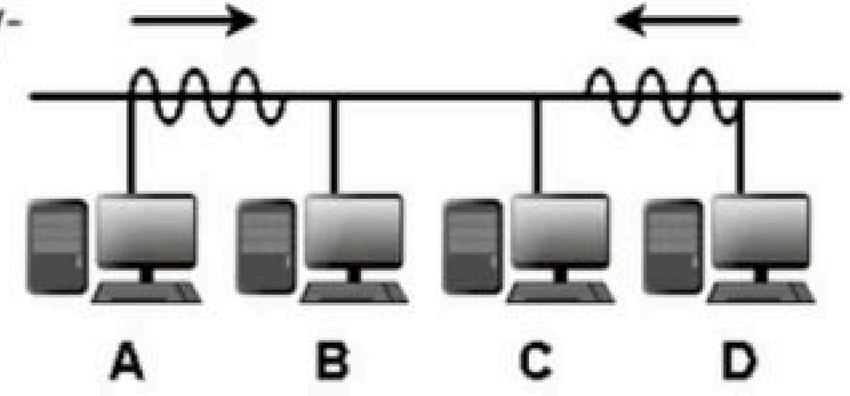
After undergoing the collision,

- Transmitting station chooses a random number in the range $[0, 2^n-1]$ if the packet is undergoing collision for the n^{th} time.
- If station chooses a number k, then-

$$\text{Back off time} = k \times \text{Time slot}$$

Example-

Consider the following scenario where stations A and D start transmitting their data simultaneously-



For simplicity,
- We consider the value of time slot = 1 unit.
- Thus, back off time = K units.

Scene-01: For 1st Data Packet Of Both Stations-

- Both the stations start transmitting their 1^{st} data packet simultaneously.
- This leads to a collision.
- Clearly, the collision on both the packets is occurring for the 1^{st} time.
- So, collision number for the 1^{st} data packet of both the stations = 1.

At Station A-

After detecting the collision,

•Station A randomly chooses a number in the range $[0, 2^1-1]$ = [0,1].
•If station A chooses the number K_A, then back off time = K_A units.

At Station D-

After detecting the collision,
•Station D randomly chooses a number in the range $[0, 2^1-1]$ = [0,1].
•If station D chooses the number K_D, then back off time = K_D units.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 1st collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 1st collision.
1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.

From here,
•Probability of station A to successfully retransmit its data after the 1^{st} collision = 1 / 4
•Probability of station D to successfully retransmit its data after the 1^{st} collision = 1 / 4
•Probability of occurrence of collision again after the 1^{st} collision = 2 / 4 = 1 / 2

Now,
•Consider case-02 occurs.
•This causes station A to successfully retransmit its 1^{st} packet after the 1^{st} collision.

Scene-02: For 2nd Data Packet Of Station A And 1st Data Packet Of Station D-

Consider after some time,
•Station A starts transmitting its 2^{nd} data packet and station D starts retransmitting its 1^{st} data packet simultaneously.
•This leads to a collision.

At Station A-

- The 2nd data packet of station A undergoes collision for the 1st time.
- So, collision number for the 2nd data packet of station A = 1.
- Now, station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A units.

At Station D-

- The 1st data packet of station D undergoes collision for the 2nd time.
- So, collision number for the 1st data packet of station D = 2.
- Now, station D randomly chooses a number in the range $[0, 2^2-1] = [0,3]$.
- If station D chooses the number K_D, then back off time = K_D units.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
0	2	•In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
0	3	•In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 2nd collision.
1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.
1	2	•In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
1	3	•In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.

From here,

- Probability of station A to successfully retransmit its data after the 2^{nd} collision = 5 / 8
- Probability of station D to successfully retransmit its data after the 2^{nd} collision = 1 / 8
- Probability of occurrence of collision again after the 2^{nd} collision = 2 / 8 = 1 / 4

Now,

- Consider case-03 occurs.
- This causes station A to successfully retransmit its 2^{nd} packet after the 2^{nd} collision.

Scene-03: For 3rd Data Packet Of Station A And 1st Data Packet Of Station D-

Consider after some time,

- Station A starts transmitting its 3rd data packet and station D starts retransmitting its 1st data packet simultaneously.
- This leads to a collision.

At Station A-

- The 3rd data packet of station A undergoes collision for the 1st time.
- So, collision number for the 3rd data packet of station A = 1.
- Now, station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A unit.

At Station D-

- The 1st data packet of station D undergoes collision for the 3rd time.
- So, collision number for the 1st data packet of station D = 3.
- Now, station D randomly chooses a number in the range $[0, 2^3-1] = [0,7]$.
- If station D chooses the number K_D, then back off time = K_D unit.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	2	•In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	3	•In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	4	•In this case, station A starts retransmitting its data immediately while station D waits for 4 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	5	•In this case, station A starts retransmitting its data immediately while station D waits for 5 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	6	•In this case, station A starts retransmitting its data immediately while station D waits for 6 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	7	•In this case, station A starts retransmitting its data immediately while station D waits for 7 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 3rd collision.

1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.
1	2	•In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	3	•In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	4	•In this case, station A waits for 1 unit of time while station D waits for 4 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	5	•In this case, station A waits for 1 unit of time while station D waits for 5 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	6	•In this case, station A waits for 1 unit of time while station D waits for 6 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	7	•In this case, station A waits for 1 unit of time while station D waits for 7 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.

From here,

- Probability of station A to successfully retransmit its data after the 3rd collision = 13 / 16
- Probability of station D to successfully retransmit its data after the 3rd collision = 1 / 16
- Probability of occurrence of collision again after the 3rd collision = 1 / 16

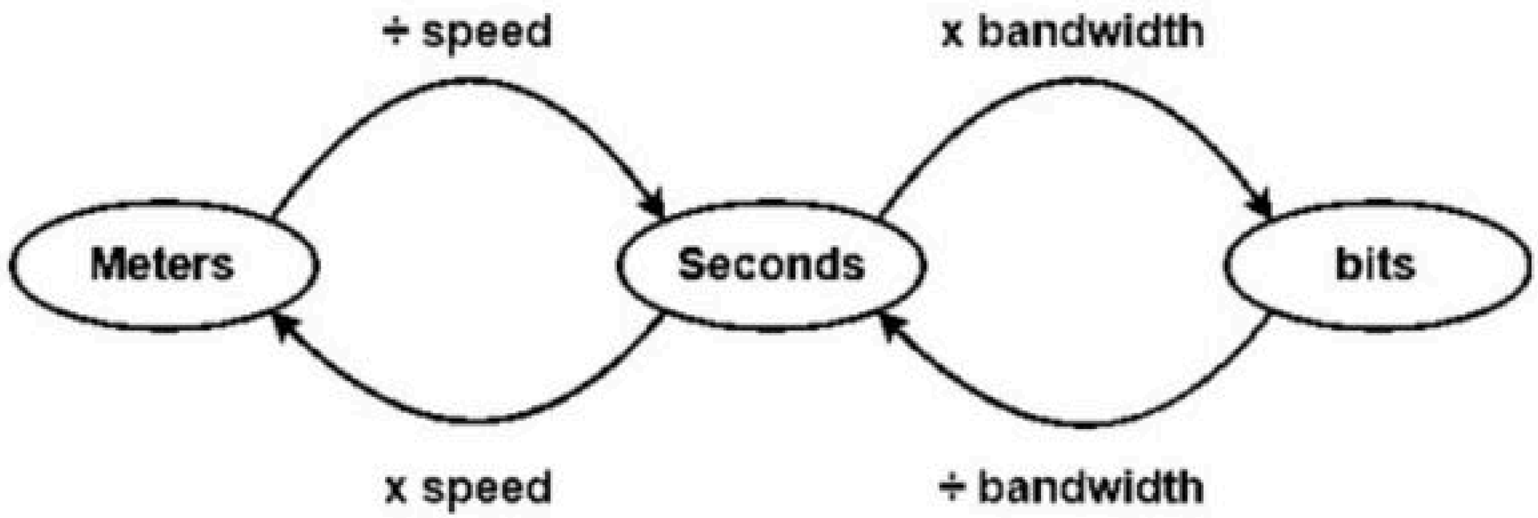
In the similar manner, the procedure continues.

Before discussing Token Passing, let us discuss few important concepts required for the discussion.

Time Conversions-

In token passing,

- Time may be expressed in seconds, bits or meters.
- To convert the time from one unit to another, we use the following conversion chart-



Token Passing Terminology-

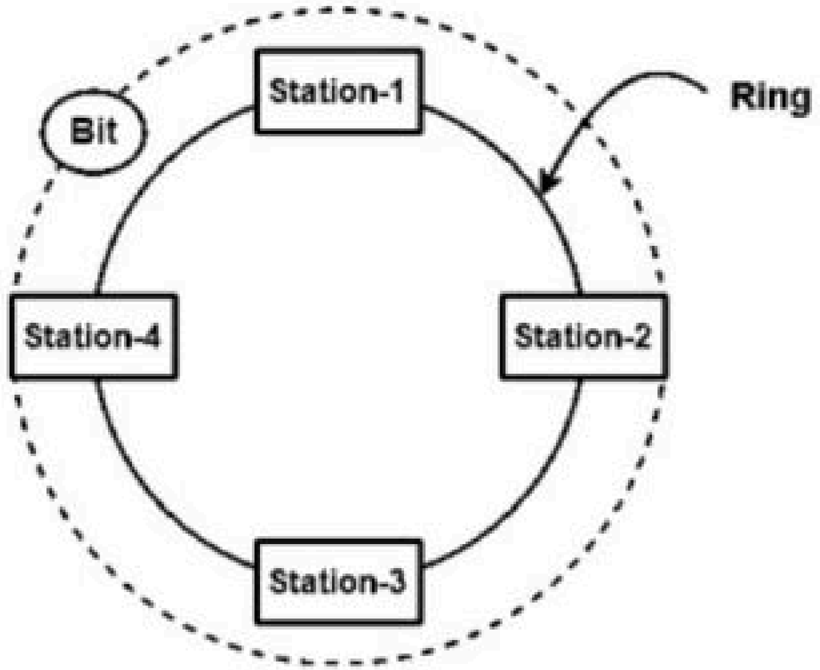
The following terms are frequently used-
1.Token
2.Ring Latency
3.Cycle Time

1. Token-

•A token is a small message composed of a special bit pattern.
•It represents the permission to send the data packet.
•A station is allowed to transmit a data packet if and only if it possess the token otherwise not.

2. Ring Latency-

Time taken by a bit to complete one revolution of the ring is called as **ring latency**.



Let us derive the expression for ring latency.

If-

- Length of the ring = d
- Speed of the bit = v
- Number of stations = N
- Bit delay at each station = b

(Bit delay is the time for which a station holds the bit before transmitting to the other side)

$$\text{Ring Latency} = \frac{d}{v} + N \times b$$

This time is taken by the bit to traverse the ring

This time is taken by the stations to hold the bit

<u>**Notes-**</u>

- d / v is the propagation delay (T_p) expressed in seconds.
- Generally, bit delay is expressed in bits.
- So, both the terms (d / v and $N \times b$) have different units.
- While calculating the ring latency, both the terms are brought into the same unit.
- The above conversion chart is used for conversion.

After conversion, we have-

$$\text{Ring Latency} = \left(\frac{d}{v} + \frac{N \times b}{B}\right) \text{ sec}$$
$$= \left(T_p + \frac{N \times b}{B}\right) \text{ sec}$$

OR

$$\text{Ring Latency} = \left(\frac{d \times B}{v} + N \times b\right) \text{ bits}$$
$$= \left(T_p \times B + N \times b\right) \text{ bits}$$

3. Cycle Time-

Time taken by the token to complete one revolution of the ring is called as **cycle time**.

If-

- Length of the ring = d
- Speed of the bit = v
- Number of stations = N
- Token Holding Time = THT

(Token Holding Time is the time for which a station holds the token before transmitting to the other side)

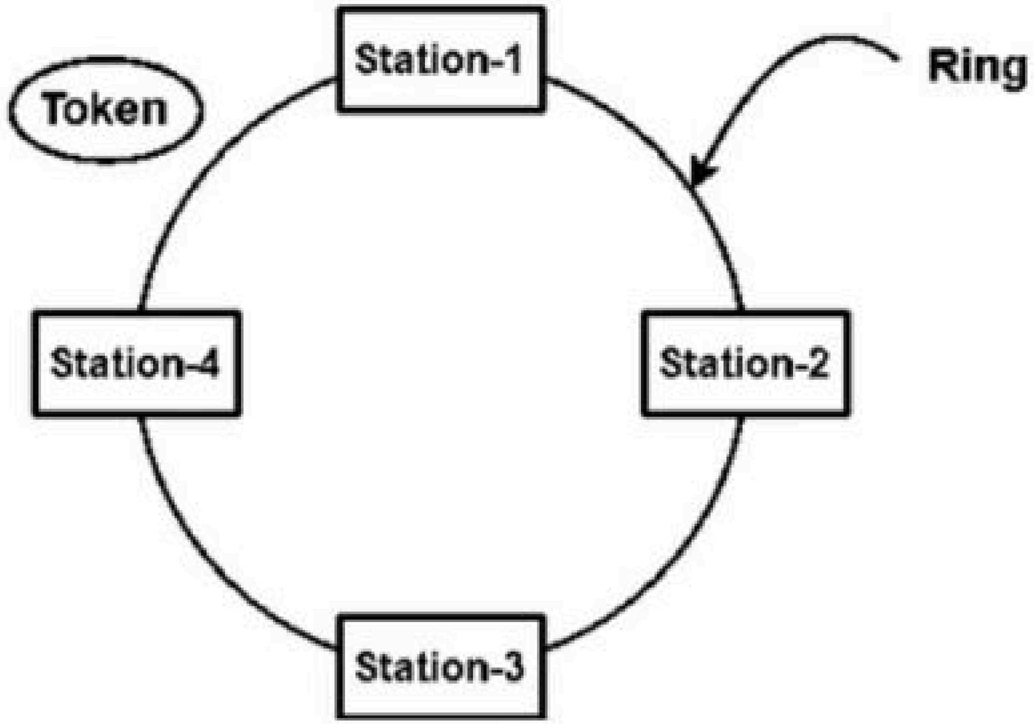
Then-

$$\boxed{\begin{aligned}\text{Cycle Time} &= \frac{d}{v} + N \times THT \\ &= T_p + N \times THT\end{aligned}}$$

Token Passing-

In this access control method,

- All the stations are logically connected to each other in the form of a ring.
- The access of stations to the transmission link is governed by a token.
- A station is allowed to transmit a data packet if and only if it possess the token otherwise not.
- Each station passes the token to its neighboring station either clockwise or anti-clockwise.



Assumptions-

Token passing method assumes-

- Each station in the ring has the data to send.
- Each station sends exactly one data packet after acquiring the token.

Efficiency-

Efficiency (η) = Useful Time / Total Time

In one cycle,

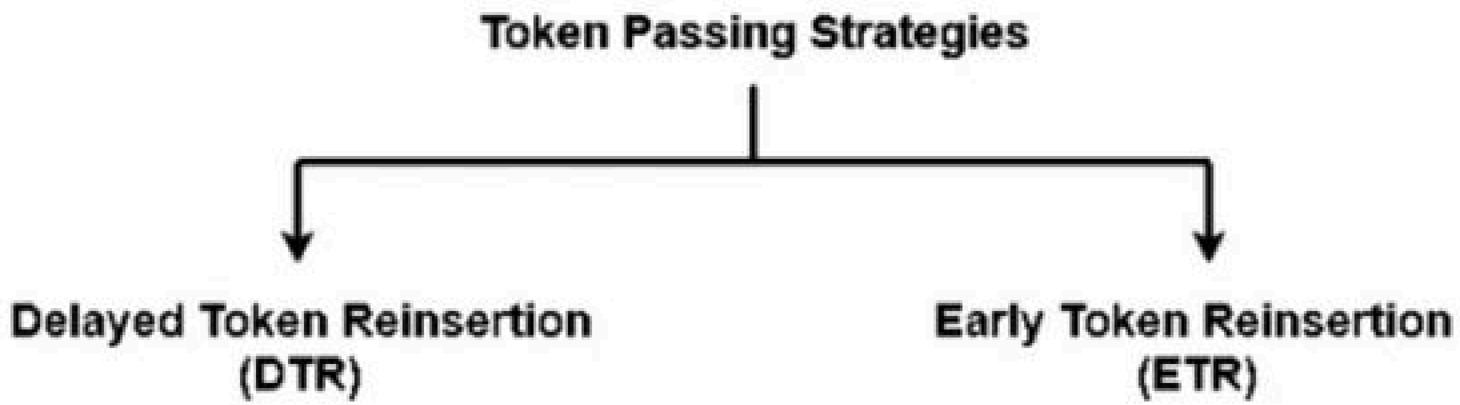
- Useful time = Sum of transmission delay of N stations since each station sends 1 data packet = $N \times T_t$
- Total Time = Cycle time = $T_p + N \times THT$

Thus,

$$\boxed{\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times THT}}$$

Token Passing Strategies-

The following 2 strategies are used in token passing-

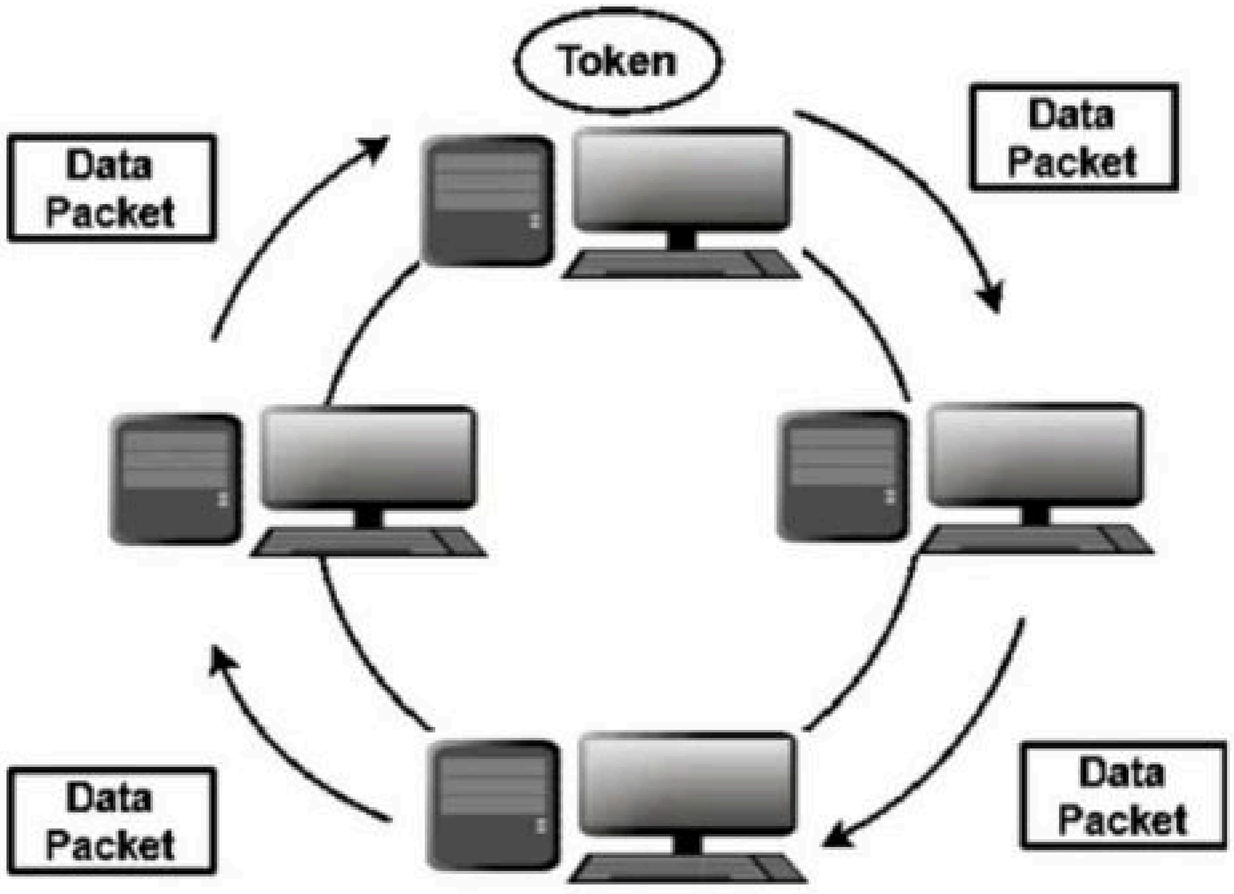


1. Delayed Token Reinsertion-

In this strategy,
•Station keeps holding the token until the last bit of the data packet transmitted by it takes the complete revolution of the ring and comes back to it.

Working-

After a station acquires the token,
•It transmits its data packet.
•It holds the token until the data packet reaches back to it.
•After data packet reaches to it, it discards its data packet as its journey is completed.
•It releases the token.



Delayed Token Reinsertion Token Passing

Token Holding Time-

$$\text{Token Holding Time (THT)} = \text{Transmission delay} + \text{Ring Latency}$$

We know,

- Ring Latency = T_p + N x bit delay
- Assuming bit delay = 0 (in most cases), we get-

$$\text{Token Holding Time} = T_t + T_p$$

Efficiency-

Substituting THT = $T_t + T_p$ in the efficiency expression, we get-

$$\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times (T_t + T_p)}$$

OR

$$\text{Efficiency } (\eta) = \frac{1}{\frac{a}{N} + (1 + a)}$$

OR

$$\text{Efficiency } (\eta) = \frac{1}{1 + \left(1 + \frac{1}{N}\right) a}$$

OR

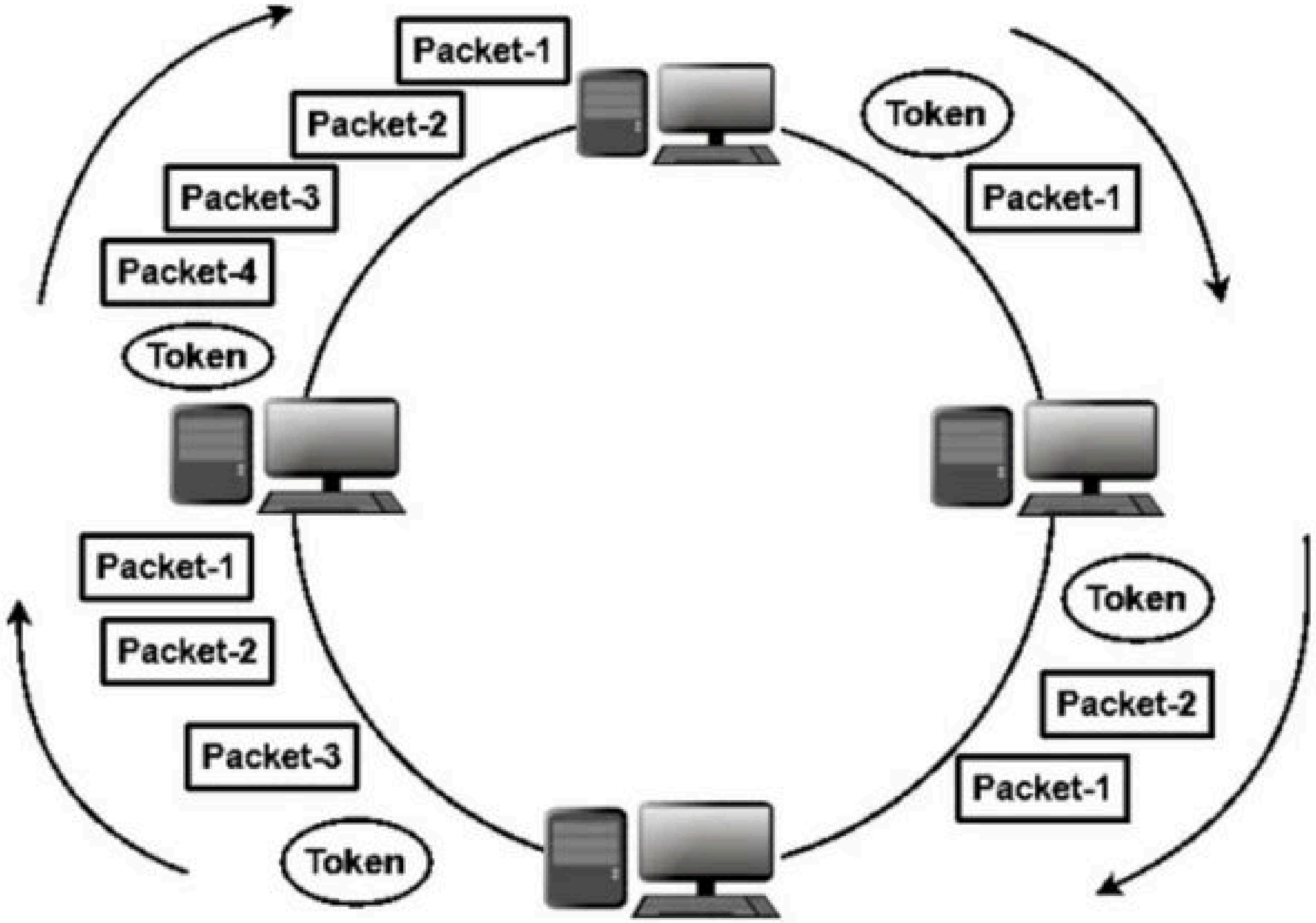
$$\text{Efficiency } (\eta) = \frac{1}{1 + \left(\frac{N + 1}{N}\right) a}$$

2. Early Token Reinsertion-

In this strategy,

•Station releases the token immediately after putting its data packet to be transmitted on the ring.

Working-



Step-01: At Station-1:

Station-1
•Acquires the token
•Transmits packet-1
•Releases the token

Step-02: At Station-2:

Station-2
•Receives packet-1
•Transmits packet-1
•Acquires the token
•Transmits packet-2
•Releases the token

Step-03: At Station-3:

Station-3
•Receives packet-1
•Transmits packet-1
•Receives packet-2
•Transmits packet-2
•Acquires the token
•Transmits packet-3
•Releases the token

Step-04: At Station-4:

Station-4
•Receives packet-1
•Transmits packet-1
•Receives packet-2
•Transmits packet-2
•Receives packet-3
•Transmits packet-3
•Acquires the token
•Transmits packet-4
•Releases the token

Step-05: At Station-1:

•Receives packet-1
•Discards packet-1 (as its journey is completed)
•Receives packet-2
•Transmits packet-2
•Receives packet-3
•Transmits packet-3
•Receives packet-4
•Transmits packet-4
•Acquires the token
•Transmits packet-1 (new)
•Releases the token

In this manner, the cycle continues.

Token Holding Time-

Token Holding Time (THT) = Transmission delay of data packet = T_t

Efficiency-

Substituting THT = T_t in the efficiency expression, we get-

$$\boxed{\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times T_t}}$$

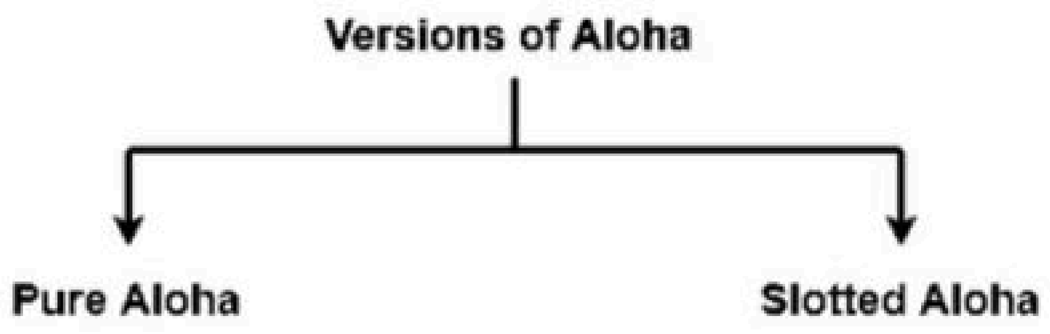
OR

$$\boxed{\text{Efficiency } (\eta) = \frac{1}{1 + \frac{a}{N}}}$$

Delay Token Retransmission (DTR)	Early Token Retransmission (ETR)
Each station holds the token until its data packet reaches back to it.	Each station releases the token immediately after putting its data packet on the ring.
There exists only one data packet on the ring at any given instance.	There exists more than one data packet on the ring at any given instance.
It is more reliable than ETR.	It is less reliable than DTR.
It has low efficiency as compared to ETR.	It has high efficiency as compared to ETR.

Aloha-

There are two different versions of Aloha-



1. Pure Aloha-

- It allows the stations to transmit data at any time whenever they want.
- After transmitting the data packet, station waits for some time.

Then, following 2 cases are possible-

Case-01:

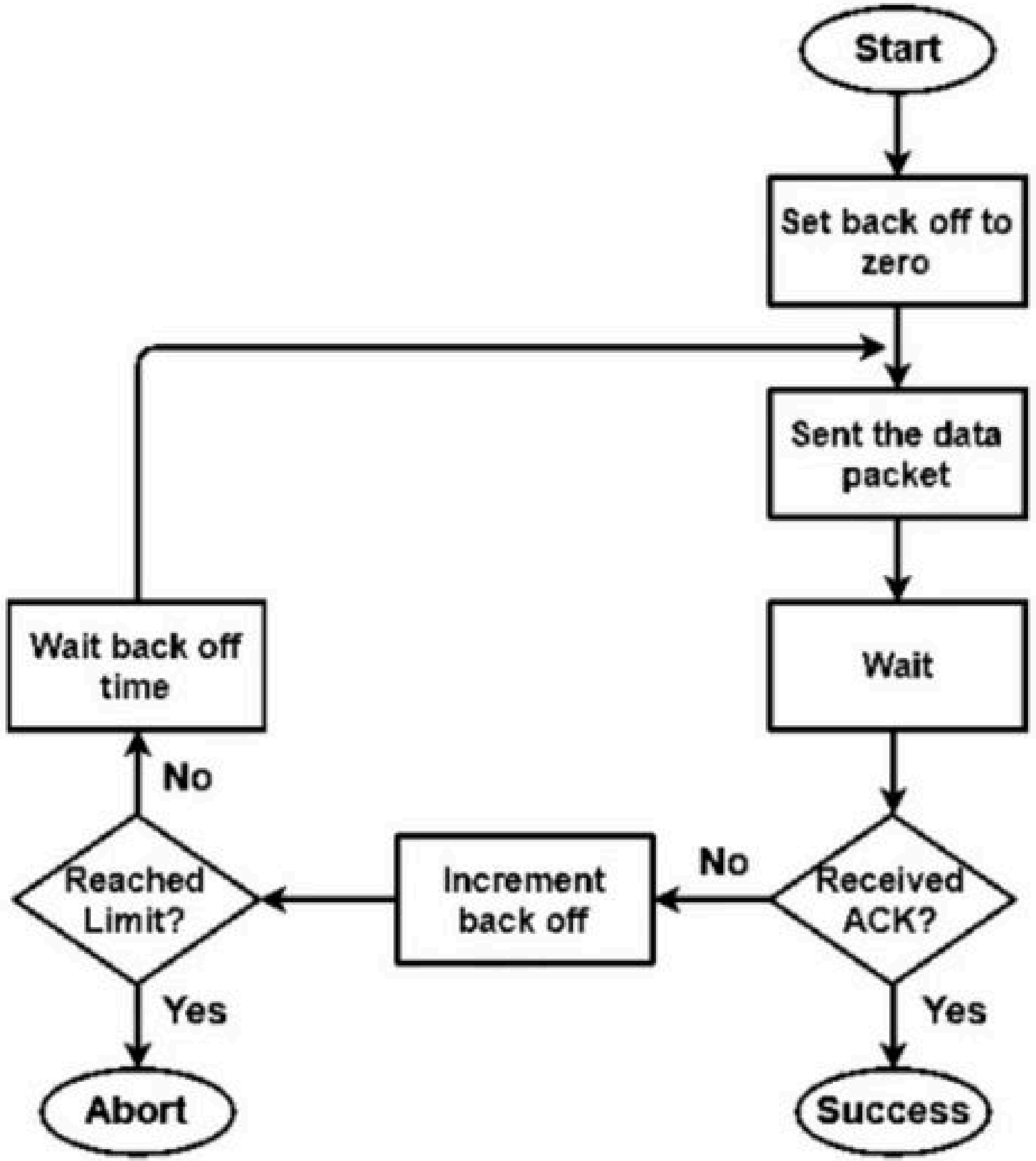
- Transmitting station receives an acknowledgement from the receiving station.
- In this case, transmitting station assumes that the transmission is successful.

Case-02:

- Transmitting station does not receive any acknowledgement within specified time from the receiving station.
- In this case, transmitting station assumes that the transmission is unsuccessful.

Then,

- Transmitting station uses a **Back Off Strategy** and waits for some random amount of time.
- After back off time, it transmits the data packet again.
- It keeps trying until the back off limit is reached after which it aborts the transmission.



Flowchart for Pure Aloha

Efficiency-

Efficiency of Pure Aloha (η) = $G \times e^{-2G}$

where G = Number of stations willing to transmit data

Maximum Efficiency-

For maximum efficiency,

- We put $d\eta / dG = 0$
- Maximum value of η occurs at $G = 1/2$
- Substituting $G = 1/2$ in the above expression, we get-

Maximum efficiency of Pure Aloha
$= 1/2 \times e^{-2 \times 1/2}$
$= 1 / 2e$
$= 0.184$
$= 18.4\%$

Thus, Maximum Efficiency of Pure Aloha (η) = 18.4%

2. Slotted Aloha-

- Slotted Aloha divides the time of shared channel into discrete intervals called as **time slots**.
- Any station can transmit its data in any time slot.
- The only condition is that station must start its transmission from the beginning of the time slot.
- If the beginning of the slot is missed, then station has to wait until the beginning of the next time slot.
- A collision may occur if two or more stations try to transmit data at the beginning of the same time slot.

Efficiency-

$$\text{Efficiency of Slotted Aloha } (\eta) = G \times e^{-G}$$

where G = Number of stations willing to transmit data at the beginning of the same time slot

Maximum Efficiency-

For maximum efficiency,

- We put $d\eta / dG = 0$
- Maximum value of η occurs at $G = 1$
- Substituting $G = 1$ in the above expression, we get-

Maximum efficiency of Slotted Aloha
$= 1 \times e^{-1}$
$= 1 / e$
$= 0.368$
$= 36.8\%$

Thus,

Maximum Efficiency of Slotted Aloha (η) = 36.8%

Pure Aloha	Slotted Aloha
Any station can transmit the data at any time.	Any station can transmit the data at the beginning of any time slot.
The time is continuous and not globally synchronized.	The time is discrete and globally synchronized.
Vulnerable time in which collision may occur $= 2 \times T_t$	Vulnerable time in which collision may occur $= T_t$
Probability of successful transmission of data packet $= G \times e^{-2G}$	Probability of successful transmission of data packet $= G \times e^{-G}$
Maximum efficiency = 18.4% (Occurs at G = 1/2)	Maximum efficiency = 36.8% (Occurs at G = 1)
The main advantage of pure aloha is its simplicity in implementation.	The main advantage of slotted aloha is that it reduces the number of collisions to half and doubles the efficiency of pure aloha.

Computer Networks

Practice questions on Access Control Methods and GATE PYQ

Problem 1: GATE2015(CS)

Consider a CSMA/CD network that transmits data at a rate of 100 Mbps (10^8 bits per second) over a 1 km (kilometer) cable with no repeaters. If the minimum frame size required for this network is 1250 bytes, what is the signal speed (km/sec) in the cable?

(A) 8000
(B) 10000
(C) 16000
(D) 20000

Solution:

Data should be transmitted at the rate of 100 Mbps.

Transmission Time >= 2*Propagation Time
= $1250*8 / (100 * 10^6)$
= 2*length/signal speed
= signal speed = $(2 * 10^3 * 100 * 10^6) / (1250 * 8)$
= $2 * 10 * (10^3)$ km/sec = 20000

D is correct.

Problem 2: GATE2016(CS)

Consider a LAN with four nodes S1, S2, S3 and S4. Time is divided into fixed-size slots, and a node can begin its transmission only at the beginning of a slot. A collision is said to have occurred if more than one node transmit in the same slot. The probabilities of generation of a frame in a time slot by S1, S2, S3 and S4 are 0.1, 0.2, 0.3 and 0.4, respectively. The probability of sending a frame in the first slot without any collision by any of these four stations is _________.

(A) 0.462

(B) 0.711

(C) 0.5

(D) 0.652

Solution:

The probability of sending a frame in the first slot
without any collision by any of these four stations is sum of following 4 probabilities
Probability that S1 sends a frame and no one else does
+ Probability thatS2 sends a frame and no one else does
+ Probability thatS3 sends a frame and no one else does
+ Probability thatS4 sends a frame and no one else does
= 0.1 * (1 - 0.2) * (1 - 0.3) *(1 - 0.4)
+ (1 -0.1) * 0.2 * (1 - 0.3) *(1 - 0.4)
+ (1 -0.1) * (1 - 0.2) * 0.3 *(1 - 0.4)
+ (1 -0.1) * (1 - 0.2) * (1 - 0.3) * 0.4
= 0.4404

Problem-3:

In a CSMA / CD network running at 1 Gbps over 1 km cable with no repeaters, the signal speed in the cable is 200000 km/sec. What is minimum frame size?

Solution-

Given-

- Bandwidth = 1 Gbps
- Distance = 1 km
- Speed = 200000 km/sec

Calculating Propagation Delay-

Propagation delay (T_p)

= Distance / Propagation speed
= 1 km / (200000 km/sec)
= 0.5×10^{-5} sec
= 5×10^{-6} sec

Calculating Minimum Frame Size-

Minimum frame size
= 2 x Propagation delay x Bandwidth
= $2 \times 5 \times 10^{-6}$ sec x 10^{9} bits per sec
= 10000 bits

Computer Networks

Error Control Methods PART 1

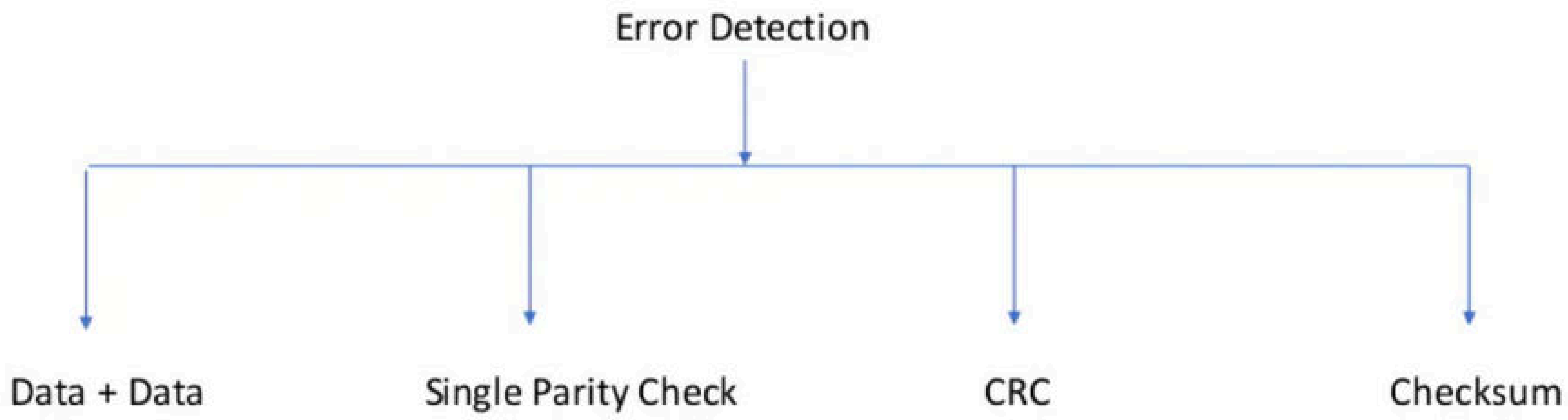
Error Handling Methods

Error Detection

Error detection is a technique that is used to check if any error occurred in the data during the transmission.

Error Correction

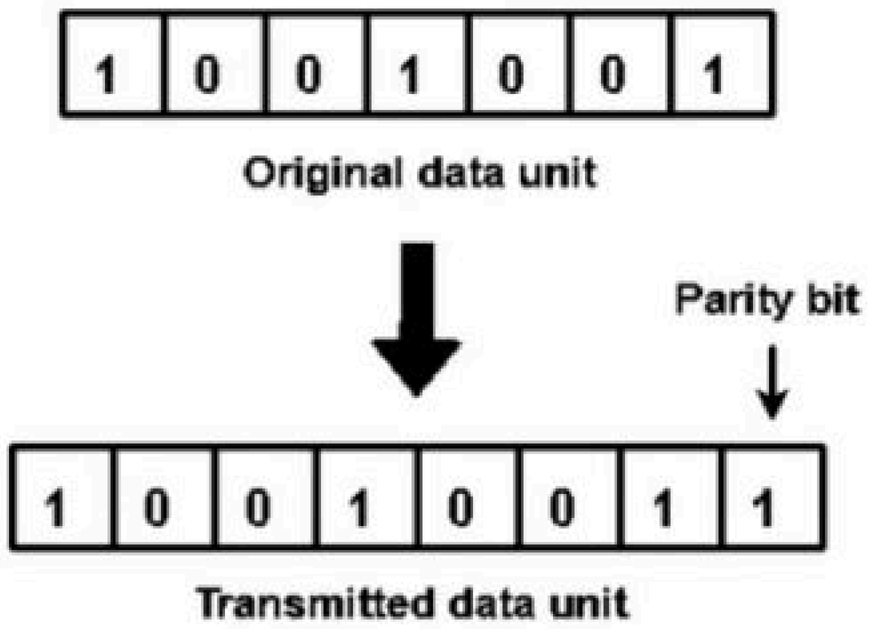
Error Correction is a technique that is used to correct error occurred in the data by its own during the transmission.


Error Detection
Data + Data
Single Parity Check
CRC
Checksum

Single Parity Check-

In this technique,

- One extra bit called as **parity bit** is sent along with the original data bits.
- Parity bit helps to check if any error occurred in the data during the transmission.



Limitation-

- This technique can not detect an even number of bit errors (two, four, six and so on).
- If even number of bits flip during transmission, then receiver can not catch the error.

Cyclic Redundancy Check-

- Cyclic Redundancy Check (CRC) is an error detection method.
- It is based on binary division.

Cyclic Generator-

Data to be sent : 1 0 1 1 0 1 1
CRC generator: 1 1 0 1

CRC generator is 4 bits
There for sender appends 3 bits of 0's to the data

Note: if CRCG= n bits then bits to be appended in data is (n-1) 0's

SENDER'S SIDE

Appended 0's

1 1 0 1 | 1 0 1 1 0 1 1 0 0 0

1 1 0 1

0 1 1 0 0 1 1 0 0 0

Go on applying XOR

Appended 0's

1 1 0 1 | 1 0 1 1 0 1 1 0 0 0

1 1 0 1

0 1 1 0 0 1 1 0 0 0

1 1 0 1

0 0 0 1 1 1 0 0 0

Go on applying XOR

Appended 0's

```
1101 | 1011011 000
       1101
       0110011000
        1101
        000111000
           1101
        000001100
```

Go on applying XOR

Appended 0's

```
1 1 0 1 ) 1 0 1 1 0 1 1 0 0 0
          1 1 0 1
          -------------------
          0 1 1 0 0 1 1 0 0 0
            1 1 0 1
          -------------------
            0 0 0 1 1 1 0 0 0
                  1 1 0 1
          -------------------
            0 0 0 0 0 1 1 0 0
                      1 1 0 1
          -------------------
            0 0 0 0 0 0 0 0 1
```

Go on applying XOR

CRC

DATA SENT : 1 0 1 1 0 1 1 0 0 1

RECEIVER'S SIDE

```
1101 ) 1011011001
       1101
       ----------
       0110011001
        1101
       ----------
        000111001
           1101
       ----------
        000001101
             1101
       ----------
        000000000
```

Go on applying XOR

CRC IS 0, DATA RECEIVED IS RIGHT!

Efficiency (η) = Useful Time / Total Time

Before a successful transmission,
•There may occur many number of collisions.
•2 x T_p time is wasted during each collision.

Thus,
•Useful time = Transmission delay of data packet = T_t
•Useless time = Time wasted during collisions + Propagation delay of data packet = c x 2 x T_p + T_p
•Here, c = Number of contention slots / collision slots.

$$\boxed{\text{Efficiency } (\eta) = \frac{T_t}{c \times 2 \times T_p + T_t + T_p}}$$

Here,
•c is a variable.
•This is because number of collisions that might occur before a successful transmission are variable.

Probabilistic Analysis shows-

Average number of collisions before a successful transmission = e

$$\text{Efficiency } (\eta) = \frac{T_t}{e \times 2 \times T_p + T_t + T_p}$$

OR

$$\text{Efficiency } (\eta) = \frac{T_t}{T_t + 6.44 \times T_p}$$

OR

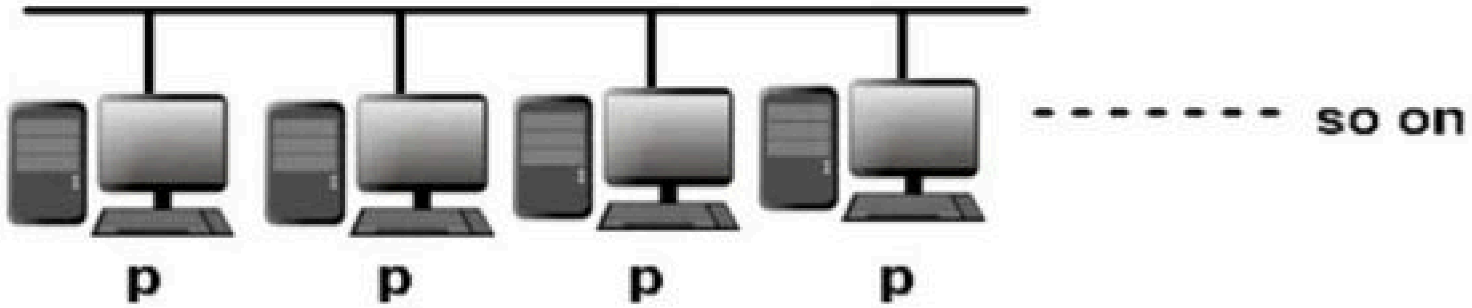
$$\text{Efficiency } (\eta) = \frac{1}{1 + 6.44 \times a}, \text{ where } a = T_p / T_t$$

Probabilistic Analysis-

Let us perform the probabilistic analysis to find the average number of collisions before a successful transmission.

Consider-

- Number of stations connected to a CSMA / CD network = n
- Probability of each station to transmit the data = p



Transmission will be successful only when-

- One station transmits the data
- Other (n-1) stations do not transmit the data.

Thus, Probability of successful transmission is given by-

$$P_{\text{successful transmission}} = {}^{n}C_1 \times p \times (1-p)^{n-1}$$

Now, let us find the maximum value of $P_{\text{successful transmission}}$.
For maximum value, we put-

$$\boxed{\frac{dP_{\text{successful transmission}}}{dp} = 0}$$

On solving,

At $p = 1/n$, we get the maximum value of $P_{\text{successful transmission}}$

Thus,
$(P_{\text{successful transmission}})_{max}$
$= {}^nC_1 \times 1/n \times (1 - 1/n)^{n-1}$
$= n \times 1/n \times (1 - 1/n)^{n-1}$
$= (1 - 1/n)^{n-1}$

$$(P_{\text{successful transmission}})_{max} = (1 - 1/n)^{n-1}$$

If there are sufficiently large number of stations i.e. $n \rightarrow \infty$, then we have-

$$\lim_{n\to\infty} \left(P_{\text{successful transmission}}\right)_{max} = \lim_{n\to\infty} \left(1 - \frac{1}{n}\right)^{n-1}$$

$$= \frac{1}{e}$$

Number of times a station must try before successfully transmitting the data packet

$= 1 / P_{max}$ (Using Poisson's distribution)

$= 1 / (1/e)$

$= e$

Average number of collisions that might occur before a successful transmission = e

Important Notes-

Note-01:

- CSMA / CD is used in wired LANs.
- CSMA / CD is standardized in IEEE 802.3

Note-02:

- CSMA / CD only minimizes the recovery time.
- It does not take any steps to prevent the collision until it has taken place.

Important Formulas-

- Condition to detect collision: Transmission delay >= 2 x Propagation delay
- Minimum length of data packets in CSMA / CD = 2 x Bandwidth x Distance / Speed
- Efficiency of CSMA / CD = 1 / (1 + 6.44 x a) where $a = T_p / T_t$
- Probability of successful transmission = ${}^{n}C_1 \times p \times (1-p)^{n-1}$
- Average number of collisions before a successful transmission = e

Back off algorithm for CSMA/CD

Back Off Time-

In CSMA / CD protocol,

- After the occurrence of collision, station waits for some random back off time and then retransmits.
- This waiting time for which the station waits before retransmitting the data is called as **back off time**.
- Back Off Algorithm is used for calculating the back off time.

Back Off Algorithm-

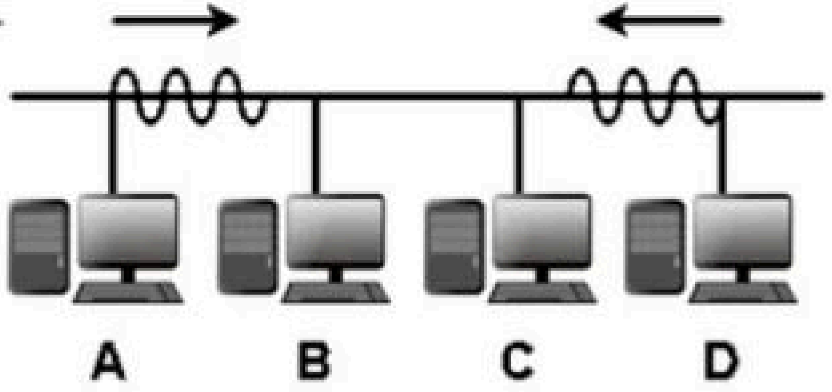
After undergoing the collision,

- Transmitting station chooses a random number in the range $[0, 2^n-1]$ if the packet is undergoing collision for the n^{th} time.
- If station chooses a number k, then-

$$\text{Back off time} = k \times \text{Time slot}$$

Example-

Consider the following scenario where stations A and D start transmitting their data simultaneously-



For simplicity,

- We consider the value of time slot = 1 unit.
- Thus, back off time = K units.

Scene-01: For 1st Data Packet Of Both Stations-

- Both the stations start transmitting their 1st data packet simultaneously.
- This leads to a collision.
- Clearly, the collision on both the packets is occurring for the 1st time.
- So, collision number for the 1st data packet of both the stations = 1.

At Station A-

After detecting the collision,

- Station A randomly chooses a number in the range $[0, 2^1-1]$ = $[0,1]$.
- If station A chooses the number K_A, then back off time = K_A units.

At Station D-

After detecting the collision,

- Station D randomly chooses a number in the range $[0, 2^1-1]$ = $[0,1]$.
- If station D chooses the number K_D, then back off time = K_D units.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 1st collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 1st collision.
1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.

From here,

- Probability of station A to successfully retransmit its data after the 1^{st} collision = 1 / 4
- Probability of station D to successfully retransmit its data after the 1^{st} collision = 1 / 4
- Probability of occurrence of collision again after the 1^{st} collision = 2 / 4 = 1 / 2

Now,

- Consider case-02 occurs.
- This causes station A to successfully retransmit its 1^{st} packet after the 1^{st} collision.

Scene-02: For 2nd Data Packet Of Station A And 1st Data Packet Of Station D-

Consider after some time,

- Station A starts transmitting its 2^{nd} data packet and station D starts retransmitting its 1^{st} data packet simultaneously.
- This leads to a collision.

At Station A-

- The 2nd data packet of station A undergoes collision for the 1st time.
- So, collision number for the 2nd data packet of station A = 1.
- Now, station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A units.

At Station D-

- The 1st data packet of station D undergoes collision for the 2nd time.
- So, collision number for the 1st data packet of station D = 2.
- Now, station D randomly chooses a number in the range $[0, 2^2-1] = [0,3]$.
- If station D chooses the number K_D, then back off time = K_D units.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
0	2	•In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
0	3	•In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 2nd collision.
1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.
1	2	•In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.
1	3	•In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 2nd collision.

From here,

- Probability of station A to successfully retransmit its data after the 2nd collision = 5 / 8
- Probability of station D to successfully retransmit its data after the 2nd collision = 1 / 8
- Probability of occurrence of collision again after the 2nd collision = 2 / 8 = 1 / 4

Now,

- Consider case-03 occurs.
- This causes station A to successfully retransmit its 2nd packet after the 2nd collision.

Scene-03: For 3rd Data Packet Of Station A And 1st Data Packet Of Station D-

Consider after some time,

- Station A starts transmitting its 3rd data packet and station D starts retransmitting its 1st data packet simultaneously.
- This leads to a collision.

At Station A-

- The 3rd data packet of station A undergoes collision for the 1st time.
- So, collision number for the 3rd data packet of station A = 1.
- Now, station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A unit.

At Station D-

- The 1st data packet of station D undergoes collision for the 3rd time.
- So, collision number for the 1st data packet of station D = 3.
- Now, station D randomly chooses a number in the range $[0, 2^3-1] = [0,7]$.
- If station D chooses the number K_D, then back off time = K_D unit.

K_A	K_D	Remarks
0	0	•In this case, both the stations start retransmitting their data immediately. •This case leads to a collision again.
0	1	•In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	2	•In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	3	•In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	4	•In this case, station A starts retransmitting its data immediately while station D waits for 4 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	5	•In this case, station A starts retransmitting its data immediately while station D waits for 5 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	6	•In this case, station A starts retransmitting its data immediately while station D waits for 6 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
0	7	•In this case, station A starts retransmitting its data immediately while station D waits for 7 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	0	•In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. •This case leads to D successfully retransmitting its data after the 3rd collision.

1	1	•In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. •This case leads to a collision again.
1	2	•In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	3	•In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	4	•In this case, station A waits for 1 unit of time while station D waits for 4 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	5	•In this case, station A waits for 1 unit of time while station D waits for 5 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	6	•In this case, station A waits for 1 unit of time while station D waits for 6 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.
1	7	•In this case, station A waits for 1 unit of time while station D waits for 7 unit of time. •This case leads to A successfully retransmitting its data after the 3rd collision.

From here,

- Probability of station A to successfully retransmit its data after the 3^{rd} collision = 13 / 16
- Probability of station D to successfully retransmit its data after the 3^{rd} collision = 1 / 16
- Probability of occurrence of collision again after the 3^{rd} collision = 1 / 16

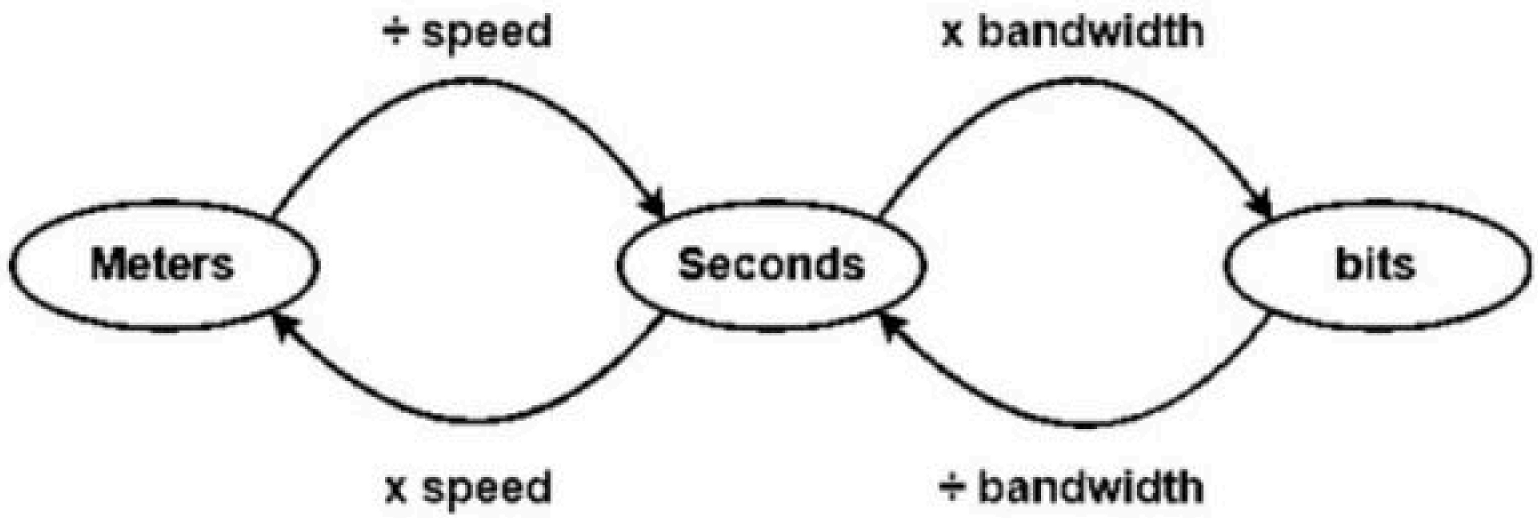
In the similar manner, the procedure continues.

Before discussing Token Passing, let us discuss few important concepts required for the discussion.

Time Conversions-

In token passing,

- Time may be expressed in seconds, bits or meters.
- To convert the time from one unit to another, we use the following conversion chart-



Token Passing Terminology-

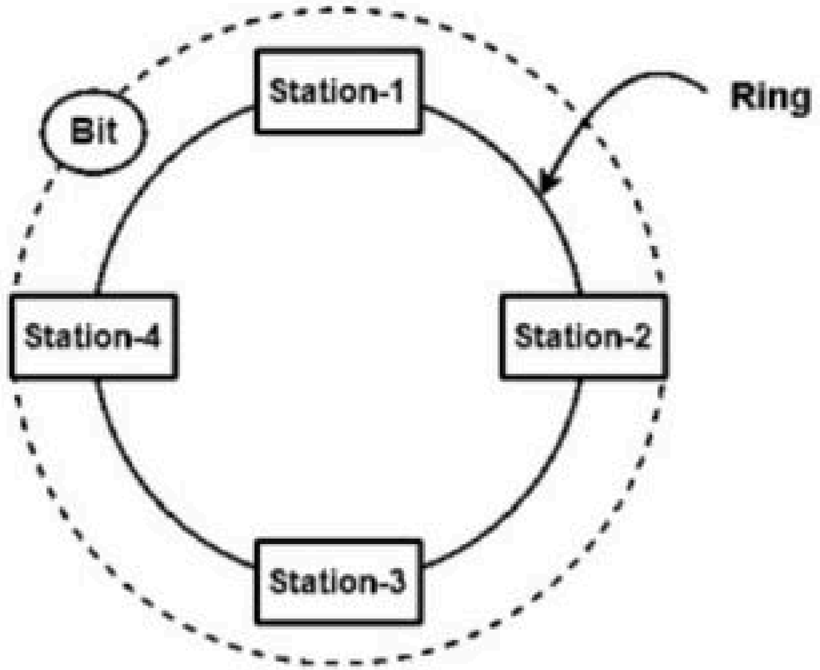
The following terms are frequently used-
1.Token
2.Ring Latency
3.Cycle Time

1. Token-

•A token is a small message composed of a special bit pattern.
•It represents the permission to send the data packet.
•A station is allowed to transmit a data packet if and only if it possess the token otherwise not.

2. Ring Latency-

Time taken by a bit to complete one revolution of the ring is called as **ring latency**.



Let us derive the expression for ring latency.

If-

- Length of the ring = d
- Speed of the bit = v
- Number of stations = N
- Bit delay at each station = b

(Bit delay is the time for which a station holds the bit before transmitting to the other side)

$$\boxed{\text{Ring Latency} = \underbrace{\frac{d}{v}}_{\text{This time is taken by the bit to traverse the ring}} + \underbrace{N \times b}_{\text{This time is taken by the stations to hold the bit}}}$$

Notes-

- d / v is the propagation delay (T_p) expressed in seconds.
- Generally, bit delay is expressed in bits.
- So, both the terms (d / v and $N \times b$) have different units.
- While calculating the ring latency, both the terms are brought into the same unit.
- The above conversion chart is used for conversion.

After conversion, we have-

$$\text{Ring Latency} = \left(\frac{d}{v} + \frac{N \times b}{B}\right) \text{ sec}$$

$$= \left(T_p + \frac{N \times b}{B}\right) \text{ sec}$$

OR

$$\text{Ring Latency} = \left(\frac{d \times B}{v} + N \times b\right) \text{ bits}$$

$$= \left(T_p \times B + N \times b\right) \text{ bits}$$

3. Cycle Time-

Time taken by the token to complete one revolution of the ring is called as **cycle time**.

If-

- Length of the ring = d
- Speed of the bit = v
- Number of stations = N
- Token Holding Time = THT

(Token Holding Time is the time for which a station holds the token before transmitting to the other side)

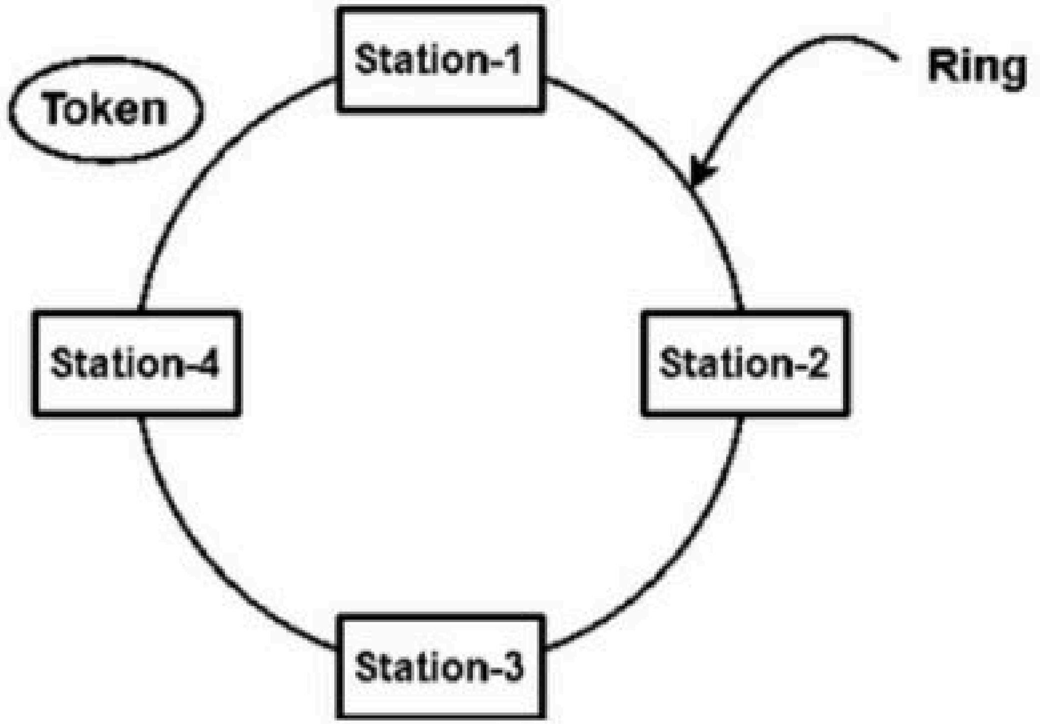
Then-

$$\boxed{\begin{aligned} \text{Cycle Time} &= \frac{d}{v} + N \times THT \\ &= T_p + N \times THT \end{aligned}}$$

Token Passing-

In this access control method,

- All the stations are logically connected to each other in the form of a ring.
- The access of stations to the transmission link is governed by a token.
- A station is allowed to transmit a data packet if and only if it possess the token otherwise not.
- Each station passes the token to its neighboring station either clockwise or anti-clockwise.



Assumptions-

Token passing method assumes-

- Each station in the ring has the data to send.
- Each station sends exactly one data packet after acquiring the token.

Efficiency-

Efficiency (η) = Useful Time / Total Time

In one cycle,

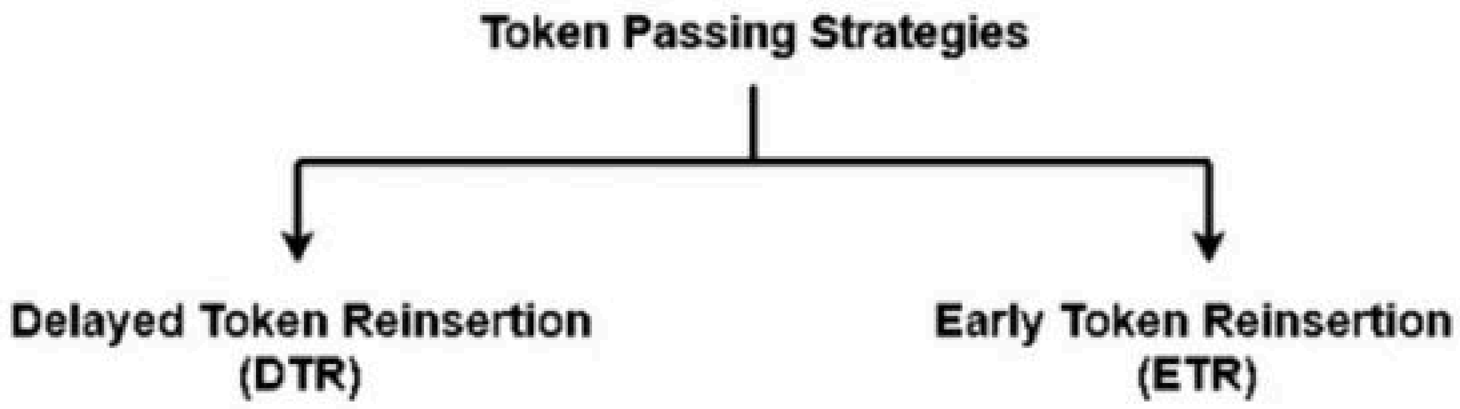
- Useful time = Sum of transmission delay of N stations since each station sends 1 data packet = $N \times T_t$
- Total Time = Cycle time = $T_p + N \times THT$

Thus,

$$\boxed{\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times THT}}$$

Token Passing Strategies-

The following 2 strategies are used in token passing-



1. Delayed Token Reinsertion-

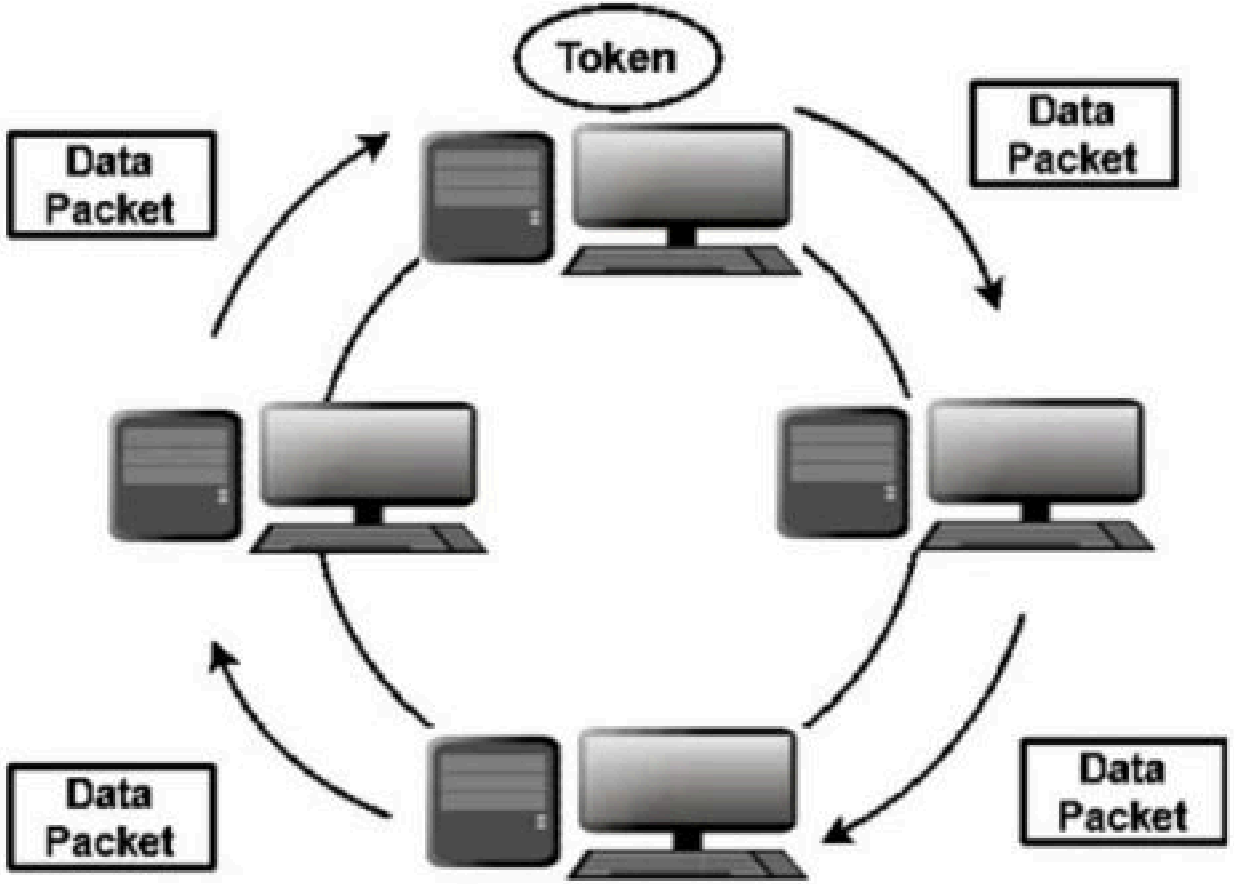
In this strategy,

- Station keeps holding the token until the last bit of the data packet transmitted by it takes the complete revolution of the ring and comes back to it.

Working-

After a station acquires the token,

- It transmits its data packet.
- It holds the token until the data packet reaches back to it.
- After data packet reaches to it, it discards its data packet as its journey is completed.
- It releases the token.



Delayed Token Reinsertion Token Passing

Token Holding Time-

Token Holding Time (THT) = Transmission delay + Ring Latency

We know,

- Ring Latency = T_p + N x bit delay
- Assuming bit delay = 0 (in most cases), we get-

Token Holding Time = $T_t + T_p$

Efficiency-

Substituting THT = $T_t + T_p$ in the efficiency expression, we get-

$$\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times (T_t + T_p)}$$

OR

$$\text{Efficiency } (\eta) = \frac{1}{\frac{a}{N} + (1 + a)}$$

OR

$$\text{Efficiency } (\eta) = \frac{1}{1 + \left(1 + \frac{1}{N}\right) a}$$

OR

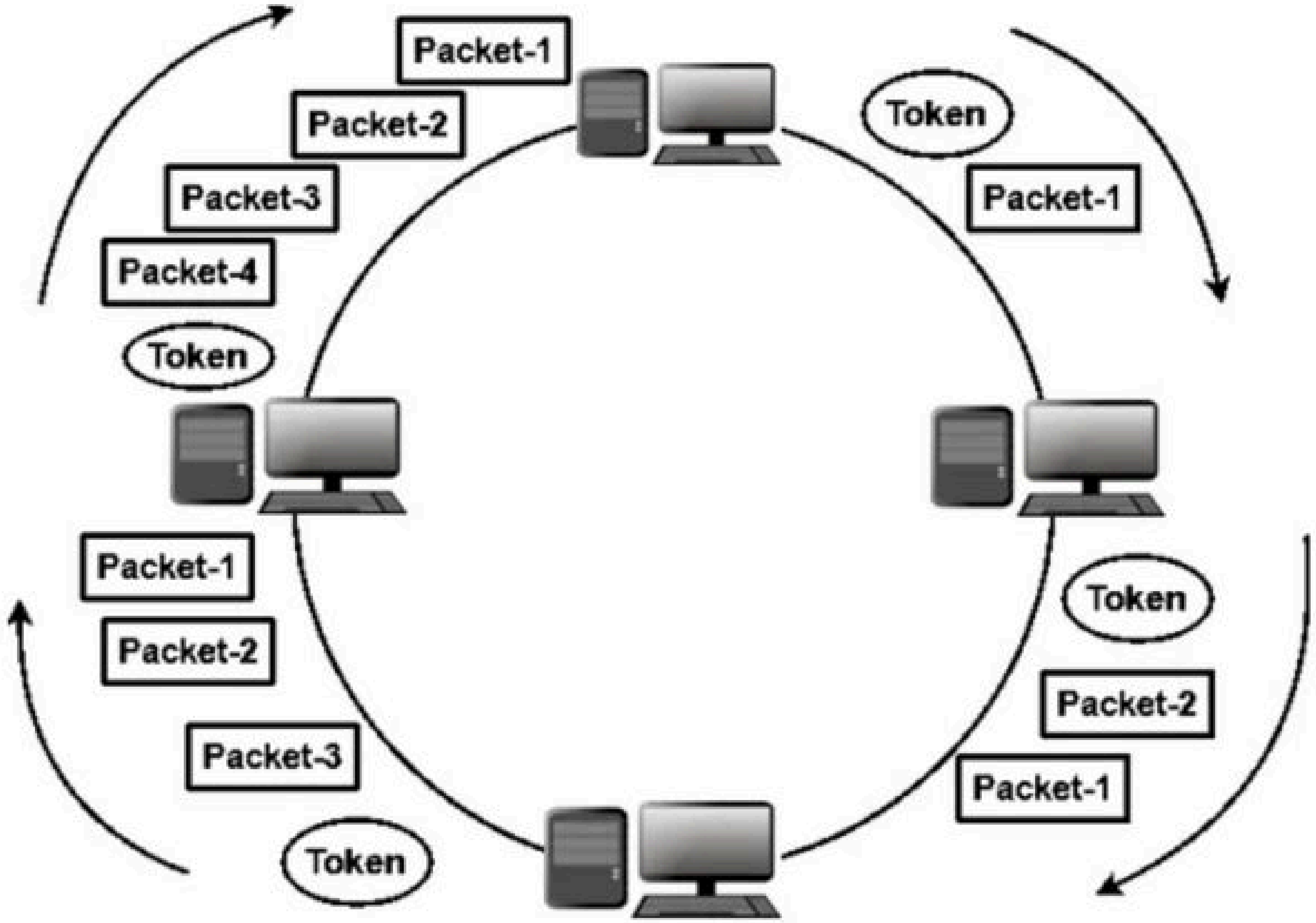
$$\text{Efficiency } (\eta) = \frac{1}{1 + \left(\frac{N + 1}{N}\right) a}$$

2. Early Token Reinsertion-

In this strategy,

•Station releases the token immediately after putting its data packet to be transmitted on the ring.

Working-



Step-01: At Station-1:

Station-1
•Acquires the token
•Transmits packet-1
•Releases the token

Step-02: At Station-2:

Station-2
•Receives packet-1
•Transmits packet-1
•Acquires the token
•Transmits packet-2
•Releases the token

Step-03: At Station-3:

Station-3
•Receives packet-1
•Transmits packet-1
•Receives packet-2
•Transmits packet-2
•Acquires the token
•Transmits packet-3
•Releases the token

Step-04: At Station-4:

Station-4
•Receives packet-1
•Transmits packet-1
•Receives packet-2
•Transmits packet-2
•Receives packet-3
•Transmits packet-3
•Acquires the token
•Transmits packet-4
•Releases the token

Step-05: At Station-1:

•Receives packet-1
•Discards packet-1 (as its journey is completed)
•Receives packet-2
•Transmits packet-2
•Receives packet-3
•Transmits packet-3
•Receives packet-4
•Transmits packet-4
•Acquires the token
•Transmits packet-1 (new)
•Releases the token

In this manner, the cycle continues.

Token Holding Time-

Token Holding Time (THT) = Transmission delay of data packet = T_t

Efficiency-

Substituting THT = T_t in the efficiency expression, we get-

$$\boxed{\text{Efficiency } (\eta) = \frac{N \times T_t}{T_p + N \times T_t}}$$

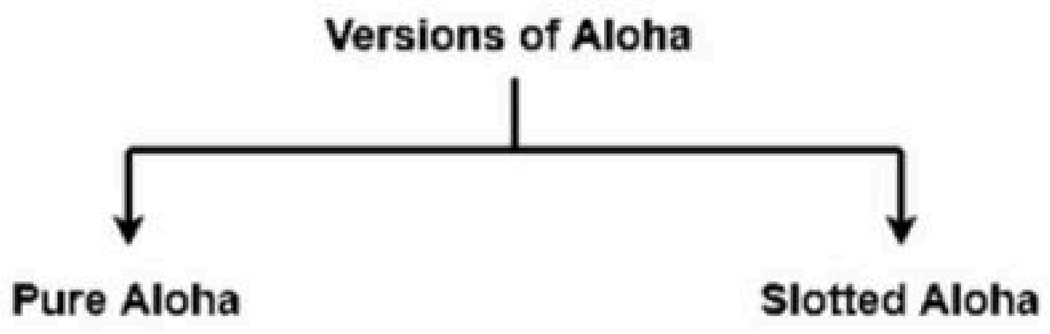
OR

$$\boxed{\text{Efficiency } (\eta) = \frac{1}{1 + \frac{a}{N}}}$$

Delay Token Retransmission (DTR)	Early Token Retransmission (ETR)
Each station holds the token until its data packet reaches back to it.	Each station releases the token immediately after putting its data packet on the ring.
There exists only one data packet on the ring at any given instance.	There exists more than one data packet on the ring at any given instance.
It is more reliable than ETR.	It is less reliable than DTR.
It has low efficiency as compared to ETR.	It has high efficiency as compared to ETR.

Aloha-

There are two different versions of Aloha-



1. Pure Aloha-

- It allows the stations to transmit data at any time whenever they want.
- After transmitting the data packet, station waits for some time.

Then, following 2 cases are possible-

Case-01:

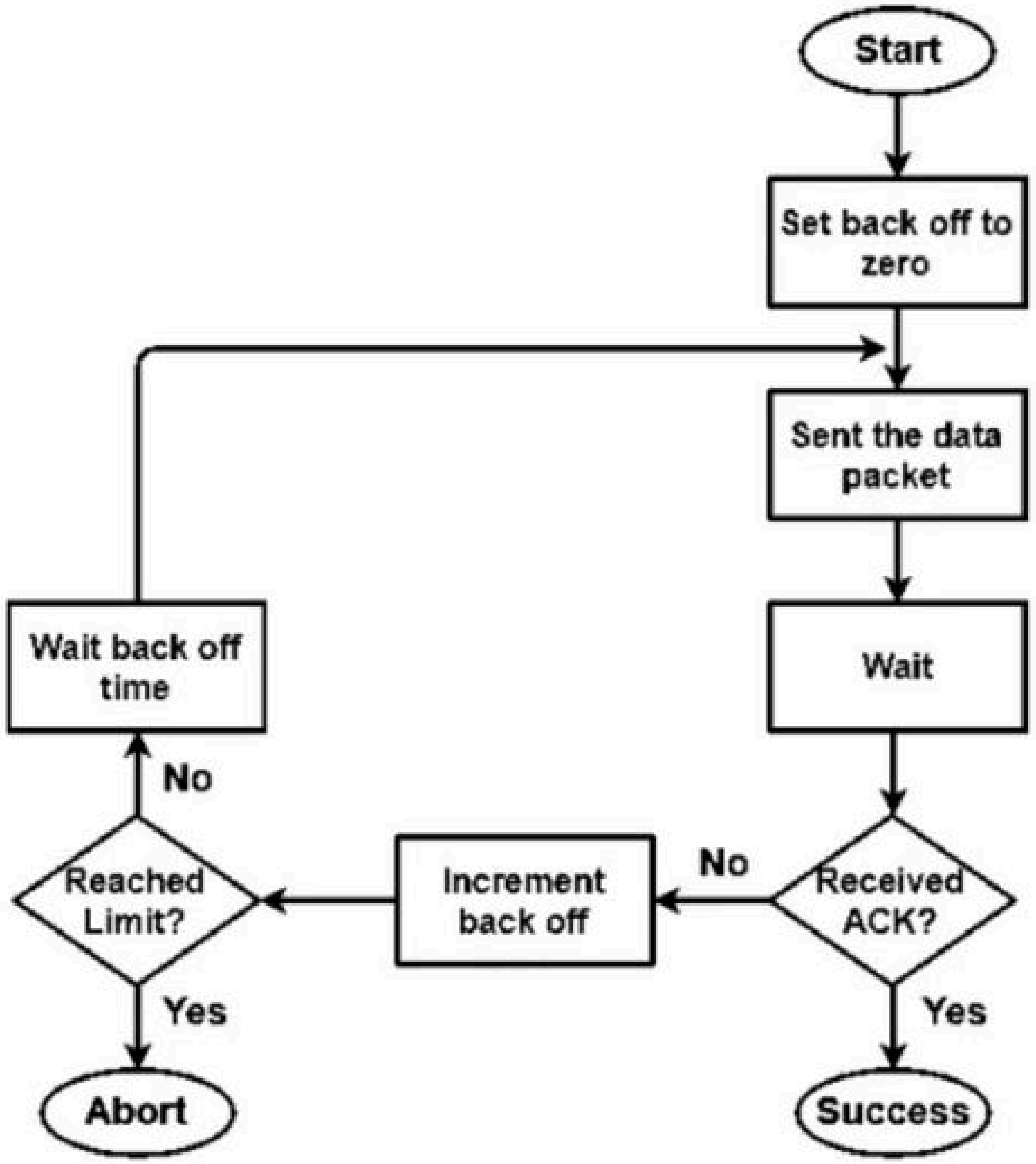
- Transmitting station receives an acknowledgement from the receiving station.
- In this case, transmitting station assumes that the transmission is successful.

Case-02:

- Transmitting station does not receive any acknowledgement within specified time from the receiving station.
- In this case, transmitting station assumes that the transmission is unsuccessful.

Then,

- Transmitting station uses a **Back Off Strategy** and waits for some random amount of time.
- After back off time, it transmits the data packet again.
- It keeps trying until the back off limit is reached after which it aborts the transmission.



Flowchart for Pure Aloha

Efficiency-

$$\text{Efficiency of Pure Aloha } (\eta) = G \times e^{-2G}$$

where G = Number of stations willing to transmit data

Maximum Efficiency-

For maximum efficiency,

- We put $d\eta / dG = 0$
- Maximum value of η occurs at $G = 1/2$
- Substituting $G = 1/2$ in the above expression, we get-

Maximum efficiency of Pure Aloha
$= 1/2 \times e^{-2 \times 1/2}$
$= 1 / 2e$
$= 0.184$
$= 18.4\%$

Thus, Maximum Efficiency of Pure Aloha (η) = 18.4%

2. Slotted Aloha-

- Slotted Aloha divides the time of shared channel into discrete intervals called as **time slots**.
- Any station can transmit its data in any time slot.
- The only condition is that station must start its transmission from the beginning of the time slot.
- If the beginning of the slot is missed, then station has to wait until the beginning of the next time slot.
- A collision may occur if two or more stations try to transmit data at the beginning of the same time slot.

Efficiency-

$$\text{Efficiency of Slotted Aloha } (\eta) = G \times e^{-G}$$

where G = Number of stations willing to transmit data at the beginning of the same time slot

Maximum Efficiency-

For maximum efficiency,
- We put $d\eta / dG = 0$
- Maximum value of η occurs at $G = 1$
- Substituting $G = 1$ in the above expression, we get-

Maximum efficiency of Slotted Aloha

$= 1 \times e^{-1}$

$= 1 / e$

$= 0.368$

$= 36.8\%$

Thus,

Maximum Efficiency of Slotted Aloha (η) = 36.8%

Pure Aloha	Slotted Aloha
Any station can transmit the data at any time.	Any station can transmit the data at the beginning of any time slot.
The time is continuous and not globally synchronized.	The time is discrete and globally synchronized.
Vulnerable time in which collision may occur $= 2 \times T_t$	Vulnerable time in which collision may occur $= T_t$
Probability of successful transmission of data packet $= G \times e^{-2G}$	Probability of successful transmission of data packet $= G \times e^{-G}$
Maximum efficiency = 18.4% (Occurs at G = 1/2)	Maximum efficiency = 36.8% (Occurs at G = 1)
The main advantage of pure aloha is its simplicity in implementation.	The main advantage of slotted aloha is that it reduces the number of collisions to half and doubles the efficiency of pure aloha.

Computer Networks

Practice questions on Access Control Methods and GATE PYQ

Problem 1: GATE2015(CS)

Consider a CSMA/CD network that transmits data at a rate of 100 Mbps (10^8 bits per second) over a 1 km (kilometer) cable with no repeaters. If the minimum frame size required for this network is 1250 bytes, what is the signal speed (km/sec) in the cable?

(A) 8000
(B) 10000
(C) 16000
(D) 20000

Solution:

Data should be transmitted at the rate of 100 Mbps.

Transmission Time >= 2*Propagation Time
= 1250*8 / (100 * 10^6)
= 2*length/signal speed
= signal speed = (2 * 10^3 * 100 * 10^6) / (1250 * 8)
= 2 * 10 * (10^3) km/sec = 20000

D is correct.

Problem 2: GATE2016(CS)

Consider a LAN with four nodes S1, S2, S3 and S4. Time is divided into fixed-size slots, and a node can begin its transmission only at the beginning of a slot. A collision is said to have occurred if more than one node transmit in the same slot. The probabilities of generation of a frame in a time slot by S1, S2, S3 and S4 are 0.1, 0.2, 0.3 and 0.4, respectively. The probability of sending a frame in the first slot without any collision by any of these four stations is _________.

(A) 0.462

(B) 0.711

(C) 0.5

(D) 0.652

Solution:

The probability of sending a frame in the first slot
without any collision by any of these four stations is sum of following 4 probabilities
Probability that S1 sends a frame and no one else does
+ Probability thatS2 sends a frame and no one else does
+ Probability thatS3 sends a frame and no one else does
+ Probability thatS4 sends a frame and no one else does
= 0.1 * (1 - 0.2) * (1 - 0.3) *(1 - 0.4)
+ (1 -0.1) * 0.2 * (1 - 0.3) *(1 - 0.4)
+ (1 -0.1) * (1 - 0.2) * 0.3 *(1 - 0.4)
+ (1 -0.1) * (1 - 0.2) * (1 - 0.3) * 0.4
= 0.4404

Problem-3:

In a CSMA / CD network running at 1 Gbps over 1 km cable with no repeaters, the signal speed in the cable is 200000 km/sec. What is minimum frame size?

Solution-

Given-

- Bandwidth = 1 Gbps
- Distance = 1 km
- Speed = 200000 km/sec

Calculating Propagation Delay-

Propagation delay (T_p)

= Distance / Propagation speed
= 1 km / (200000 km/sec)
= 0.5×10^{-5} sec
= 5×10^{-6} sec

Calculating Minimum Frame Size-

Minimum frame size
= 2 x Propagation delay x Bandwidth
= $2 \times 5 \times 10^{-6}$ sec $\times 10^{9}$ bits per sec
= 10000 bits

Computer Networks

Error Control Methods PART 1

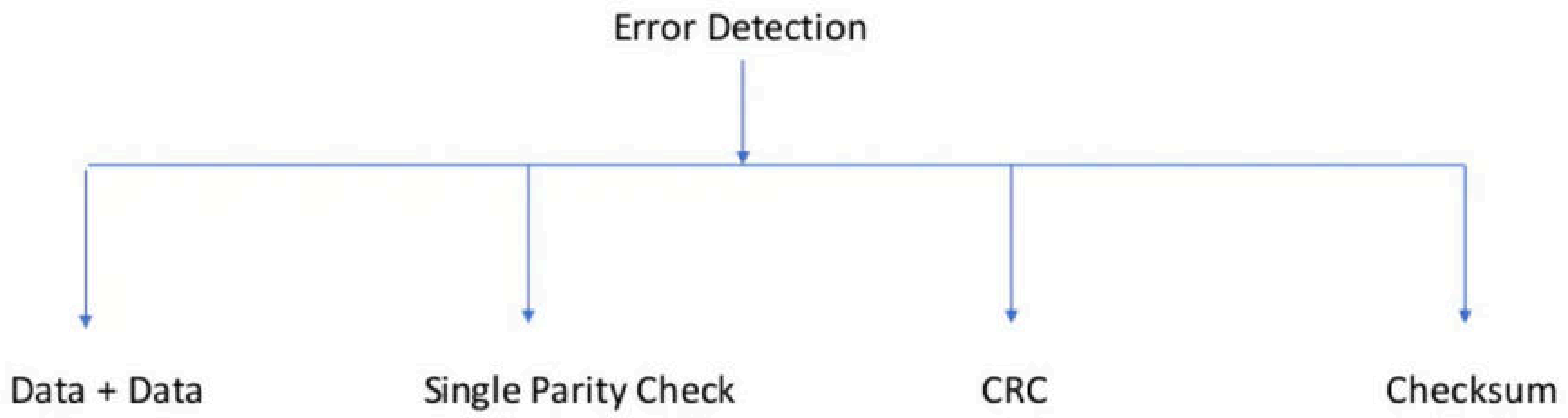
Error Handling Methods

Error Detection

Error detection is a technique that is used to check if any error occurred in the data during the transmission.

Error Correction

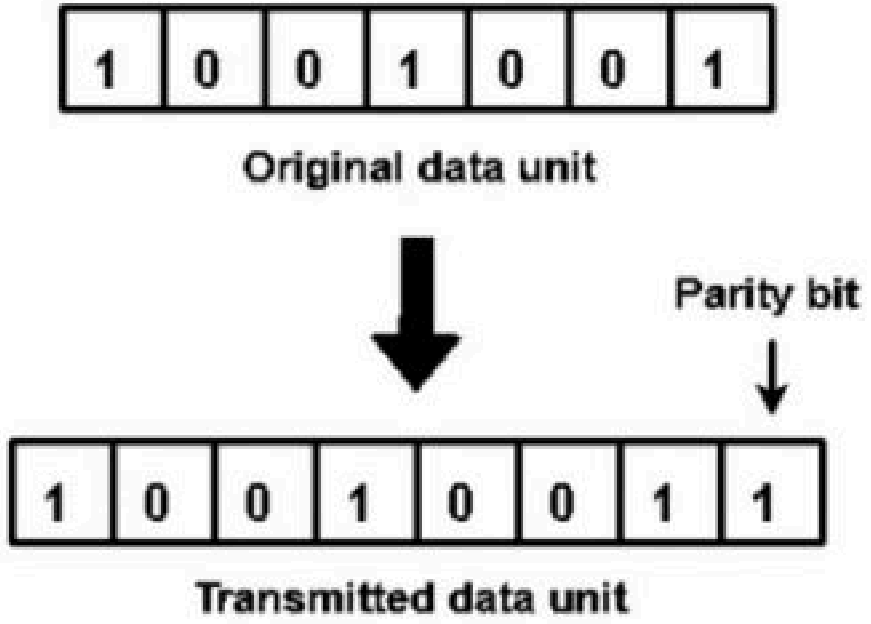
Error Correction is a technique that is used to correct error occurred in the data by its own during the transmission.


Error Detection
Data + Data
Single Parity Check
CRC
Checksum

Single Parity Check-

In this technique,

- One extra bit called as **parity bit** is sent along with the original data bits.
- Parity bit helps to check if any error occurred in the data during the transmission.



Limitation-

- This technique can not detect an even number of bit errors (two, four, six and so on).
- If even number of bits flip during transmission, then receiver can not catch the error.

Cyclic Redundancy Check-

- Cyclic Redundancy Check (CRC) is an error detection method.
- It is based on binary division.

Cyclic Generator-

Data to be sent : 1 0 1 1 0 1 1
CRC generator: 1 1 0 1

CRC generator is 4 bits
There for sender appends 3 bits of 0's to the data

Note: if CRCG= n bits then bits to be appended in data is (n-1) 0's

SENDER'S SIDE

Appended 0's

$$
\begin{array}{r|l}
1\,1\,0\,1 & 1\,0\,1\,1\,0\,1\,1\,\boxed{0\,0\,0} \\
 & 1\,1\,0\,1 \\
\hline
 & 0\,1\,1\,0\,0\,1\,1\,0\,0\,0
\end{array}
$$

Go on applying XOR

Appended 0's

```
1101 | 1011011000
       1101
       ----------
       0110011000
        1101
       ----------
        000111000
```

Go on applying XOR

Appended 0's

```
1 1 0 1 | 1 0 1 1 0 1 1 0 0 0
          1 1 0 1
          ---------------------
          0 1 1 0 0 1 1 0 0 0
            1 1 0 1
          ---------------------
            0 0 0 1 1 1 0 0 0
                  1 1 0 1
          ---------------------
            0 0 0 0 0 1 1 0 0
```

Go on applying XOR

Appended 0's

1 1 0 1 | 1 0 1 1 0 1 1 0 0 0

1 1 0 1

Go on applying XOR

0 1 1 0 0 1 1 0 0 0

1 1 0 1

0 0 0 1 1 1 0 0 0

1 1 0 1

0 0 0 0 0 1 1 0 0

1 1 0 1

CRC

0 0 0 0 0 0 0 0 1

DATA SENT : 1 0 1 1 0 1 1 0 0 1

RECEIVER'S SIDE

```
1101 ) 1011011001
       1101
       ----------
       0110011001
        1101
       ----------
        000111001
           1101
       ----------
        000001101
             1101
       ----------
        000000000
```

Go on applying XOR

CRC IS 0, DATA RECEIVED IS RIGHT!