Distance Vector Routing Example-

Consider-

- There is a network consisting of 4 routers.
- The weights are mentioned on the edges.
- Weights could be distances or costs or delays.

Step-01:

Each router prepares its routing table using its local knowledge.

Routing table prepared by each router is shown below.

At Router A-

Destination	Distance	Next Hop
A	0	A
B	2	B
C	∞	–
D	1	D

At Router B-

Destination	Distance	Next Hop
A	2	A
B	0	B
C	3	C
D	7	D

At Router C-

Destination	Distance	Next Hop

A	∞	–
B	3	B
C	0	C
D	11	D

At Router D-

Destination	Distance	Next Hop
A	1	A
B	7	B
C	11	C
D	0	D

Step-02:

- Each router exchanges its distance vector obtained in Step-01 with its neighbors.
- After exchanging the distance vectors, each router prepares a new routing table.

This is shown below-

At Router A-

- Router A receives distance vectors from its neighbors B and D.
- Router A prepares a new routing table as-

- Cost of reaching destination B from router A = min { 2+0 , 1+7 } = 2 via B.
- Cost of reaching destination C from router A = min { 2+3 , 1+11 } = 5 via B.
- Cost of reaching destination D from router A = min { 2+7 , 1+0 } = 1 via D.

Explanation For Destination B

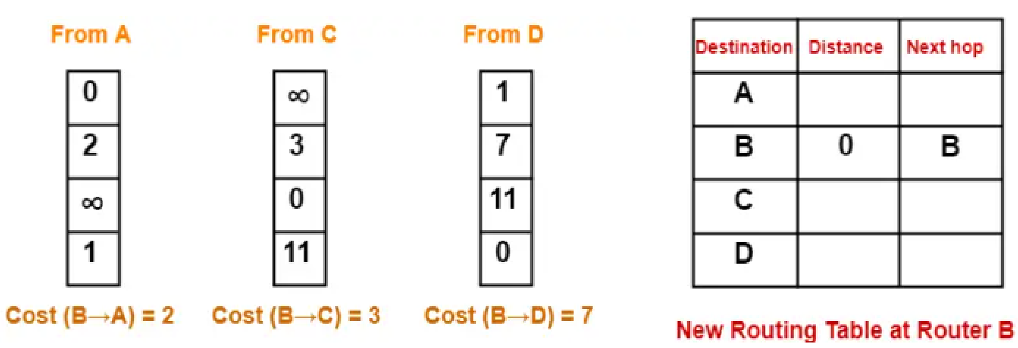
- Router A can reach the destination router B via its neighbor B or neighbor D.
- It chooses the path which gives the minimum cost.
- Cost of reaching router B from router A via neighbor B = Cost (A→B) + Cost (B→B)= **2 + 0** = 2
- Cost of reaching router B from router A via neighbor D = Cost (A→D) + Cost (D→B) = **1 + 7** = 8
- Since the cost is minimum via neighbor B, so router A chooses the path via B.
- It creates an entry (2, B) for destination B in its new routing table.
- Similarly, we calculate the shortest path distance to each destination router at every router.

Thus, the new routing table at router A is-

Destination	Distance	Next Hop
A	0	A
B	2	B
C	5	B
D	1	D

At Router B-

- Router B receives distance vectors from its neighbors A, C and D.
- Router B prepares a new routing table as-



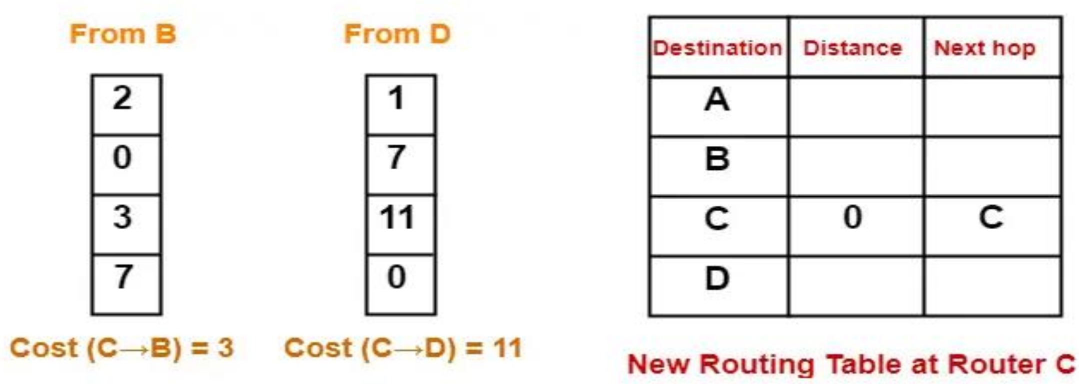
- Cost of reaching destination A from router B = min { 2+0 , 3+∞ , 7+1 } = 2 via A.
- Cost of reaching destination C from router B = min { 2+∞ , 3+0 , 7+11 } = 3 via C.
- Cost of reaching destination D from router B = min { 2+1 , 3+11 , 7+0 } = 3 via A.

Thus, the new routing table at router B is-

Destination	Distance	Next Hop
A	2	A
B	0	B
C	3	C
D	3	A

At Router C-

- Router C receives distance vectors from its neighbors B and D.
- Router C prepares a new routing table as-



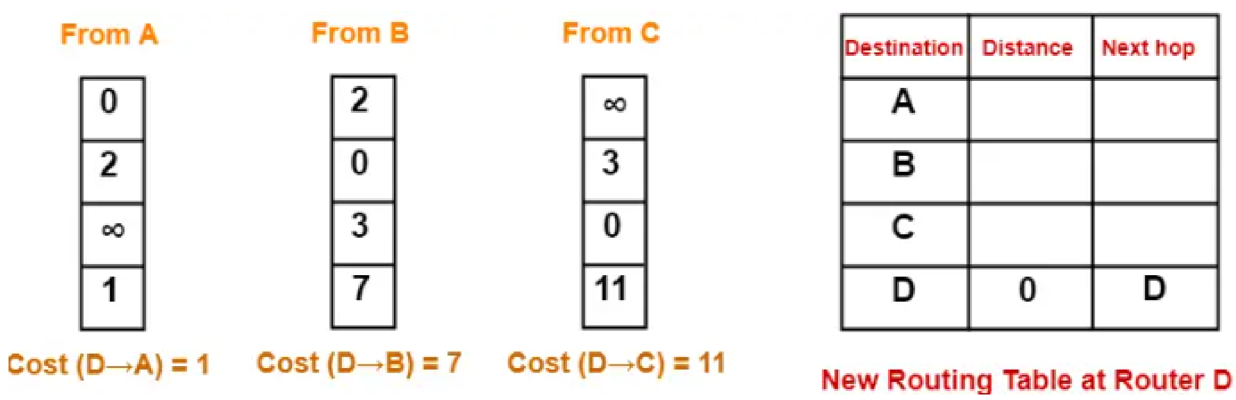
- Cost of reaching destination A from router C = min { 3+2 , 11+1 } = 5 via B.
- Cost of reaching destination B from router C = min { 3+0 , 11+7 } = 3 via B.
- Cost of reaching destination D from router C = min { 3+7 , 11+0 } = 10 via B.

Thus, the new routing table at router C is-

Destination	Distance	Next Hop
A	5	B
B	3	B
C	0	C
D	10	B

At Router D-

- Router D receives distance vectors from its neighbors A, B and C.
- Router D prepares a new routing table as-



- Cost of reaching destination A from router D = min { 1+0 , 7+2 , 11+∞ } = 1 via A.
- Cost of reaching destination B from router D = min { 1+2 , 7+0 , 11+3 } = 3 via A.
- Cost of reaching destination C from router D = min { 1+∞ , 7+3 , 11+0 } = 10 via B.

Thus, the new routing table at router D is-

Destination	Distance	Next Hop
A	1	A
B	3	A
C	10	B
D	0	D

Step-03:

Each router exchanges its distance vector obtained in Step-02 with its neighboring routers.

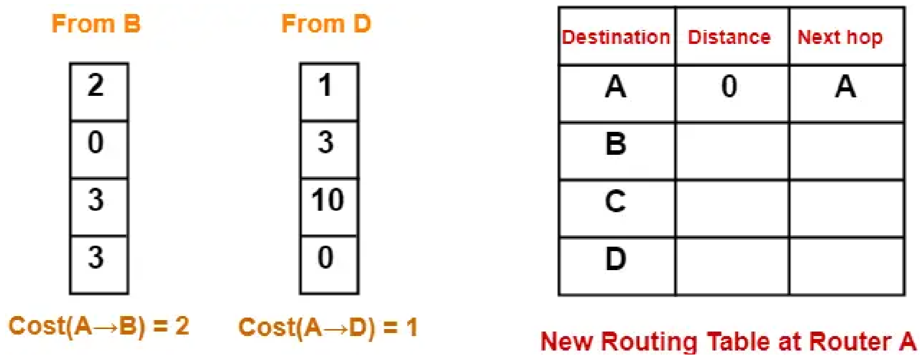
- After exchanging the distance vectors, each router prepares a new routing table.

This is shown below-

At Router A-

Router A receives distance vectors from its neighbors B and D.

- Router A prepares a new routing table as-



- Cost of reaching destination B from router A = min { 2+0 , 1+3 } = 2 via B.
- Cost of reaching destination C from router A = min { 2+3 , 1+10 } = 5 via B.
- Cost of reaching destination D from router A = min { 2+3 , 1+0 } = 1 via D.

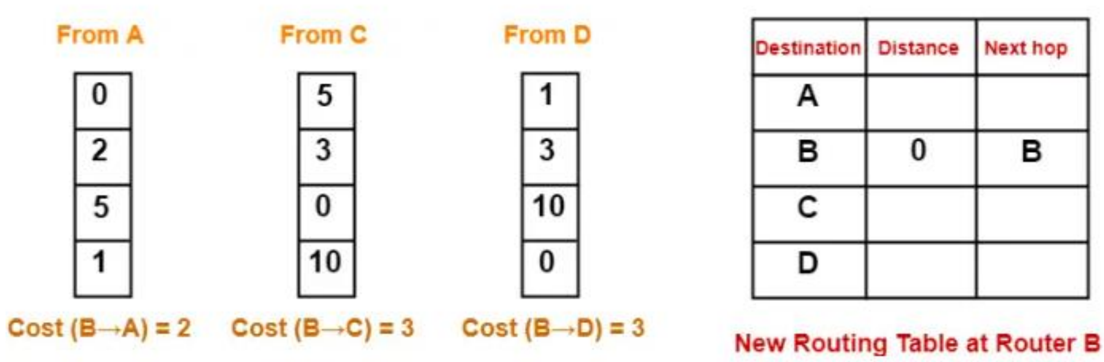
Thus, the new routing table at router A is-

Destination	Distance	Next Hop
A	0	A
B	2	B
C	5	B
D	1	D

At Router B-

Router B receives distance vectors from its neighbors A, C and D.

- Router B prepares a new routing table as-



- Cost of reaching destination A from router B = min { 2+0 , 3+5 , 3+1 } = 2 via A.
- Cost of reaching destination C from router B = min { 2+5 , 3+0 , 3+10 } = 3 via C.
- Cost of reaching destination D from router B = min { 2+1 , 3+10 , 3+0 } = 3 via A.

Thus, the new routing table at router B is-

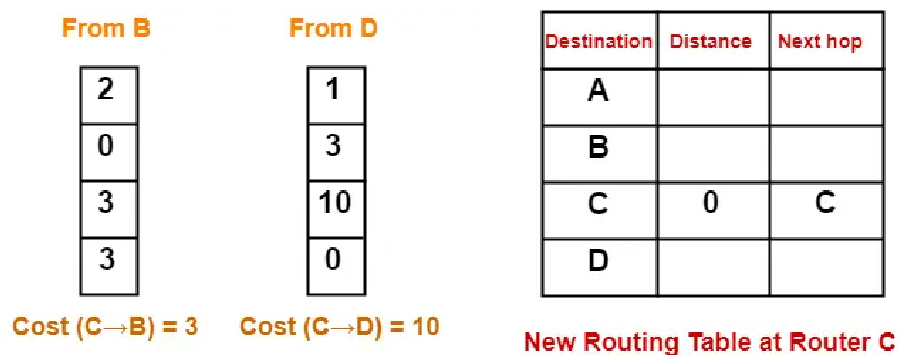
Destination	Distance	Next Hop
A	2	A

B	0	B
C	3	C
D	3	A

At Router C-

Router C receives distance vectors from its neighbors B and D.

- Router C prepares a new routing table as-



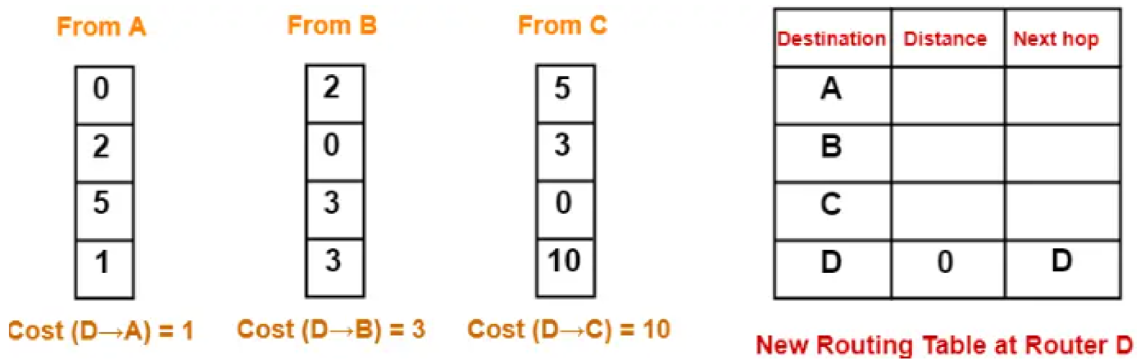
- Cost of reaching destination A from router C = min { 3+2 , 10+1 } = 5 via B.
- Cost of reaching destination B from router C = min { 3+0 , 10+3 } = 3 via B.
- Cost of reaching destination D from router C = min { 3+3 , 10+0 } = 6 via B.

Thus, the new routing table at router C is-

Destination	Distance	Next Hop
A	5	B
B	3	B
C	0	C
D	6	B

At Router D-

- Router D receives distance vectors from its neighbors A, B and C.
- Router D prepares a new routing table as-



- Cost of reaching destination A from router D = min { 1+0 , 3+2 , 10+5 } = 1 via A.
- Cost of reaching destination B from router D = min { 1+2 , 3+0 , 10+3 } = 3 via A.
- Cost of reaching destination C from router D = min { 1+5 , 3+3 , 10+0 } = 6 via A.

Thus, the new routing table at router D is-

Destination	Distance	Next Hop
A	1	A
B	3	A
C	6	A
D	0	D

These will be the final routing tables at each router.

13. Consider-

- We have a big single network having IP Address 200.1.2.0.
- We want to do subnetting and divide this network into 3 subnets.

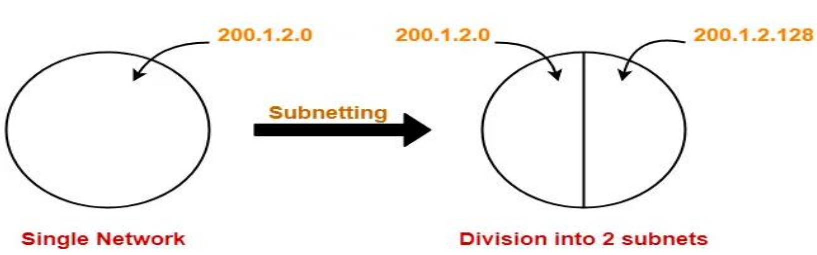
Here, the subnetting will be performed in two steps-

1. Dividing the given network into 2 subnets
2. Dividing one of the subnets further into 2 subnets

Step-01: Dividing Given Network into 2 Subnets-

The subnetting will be performed exactly in the same way as performed in Example-01.

After subnetting, we have-



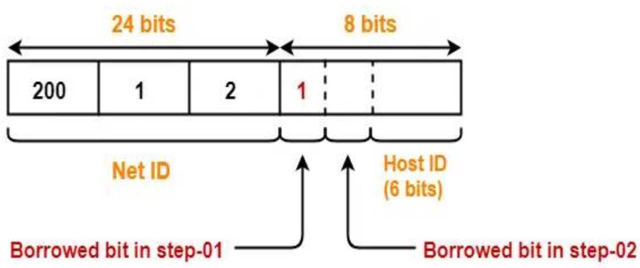
Step-02: Dividing One Subnet into 2 Subnets-

- We perform the subnetting of one of the subnets further into 2 subnets.
- Consider we want to do subnetting of the 2nd subnet having IP Address 200.1.2.128.

For creating two subnets and to represent their subnet IDs, we require 1 bit.

So,

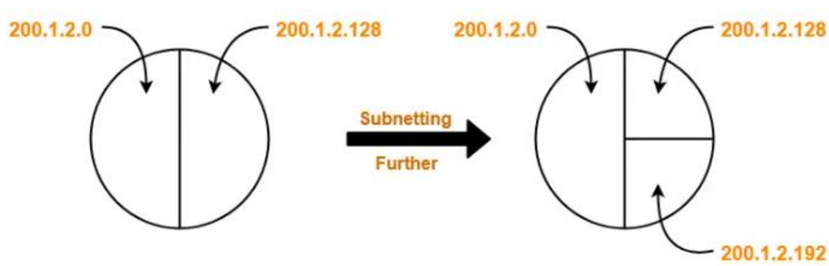
- We borrow one more bit from the Host ID part.
- After borrowing one bit, Host ID part remains with only 6 bits.



- If 2nd borrowed bit = 0, then it represents one subnet.
- If 2nd borrowed bit = 1, then it represents the other subnet.

IP Address of the two subnets are-

- 200.1.2.**10**000000 = 200.1.2.128
- 200.1.2.**11**000000 = 200.1.2.192



Finally, the given single network is divided into 3 subnets having IP Address-

- 200.1.2.0
- 200.1.2.128
- 200.1.2.192

For 1st Subnet-

- IP Address of the subnet = 200.1.2.0
- Total number of IP Addresses = 2^7 = 128
- Total number of hosts that can be configured = 128 – 2 = 126
- Range of IP Addresses = [200.1.2.**0**0000000, 200.1.2.**0**1111111] = [200.1.2.0, 200.1.2.127]
- Direct Broadcast Address = 200.1.2.**0**1111111 = 200.1.2.127
- Limited Broadcast Address = 255.255.255.255

For 2nd Subnet-

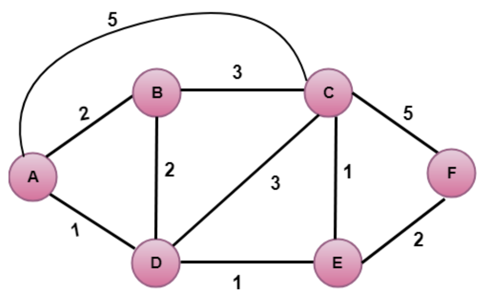
- IP Address of the subnet = 200.1.2.128
- **Total number of IP Addresses** = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**10**000000, 200.1.2.**10**111111] = [200.1.2.128, 200.1.2.191]

- Direct Broadcast Address = 200.1.2.**10**111111 = 200.1.2.191
- Limited Broadcast Address = 255.255.255.255

For 3rd Subnet-

- IP Address of the subnet = 200.1.2.192
- Total number of IP Addresses = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**11**000000, 200.1.2.**11**111111] = [200.1.2.192, 200.1.2.255]
- Direct Broadcast Address = 200.1.2.**11**111111 = 200.1.2.255
- Limited Broadcast Address = 255.255.255.255

12. Consider a network using Link state routing algorithm in its networking. The objective is to find the shortest cost path. The topology of the network is given below. Show routing table at each node initially and show how the Link State Routing algorithm would operate in this network.



Step 1:

The first step is an initialization step. The currently known least cost path from A to its directly attached neighbors, B, C, D are 2,5,1 respectively. The cost from A to B is set to 2, from A to D is set to 1 and from A to C is set to 5. The cost from A to E and F are set to infinity as they are not directly linked to A.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞

Step 2:

In the above table, we observe that vertex D contains the least cost path in step 1. Therefore, it is added in N. Now, we need to determine a least-cost path through D vertex.

a) Calculating shortest path from A to B

1. v = B, w = D
2. D(B) = min(D(B) , D(D) + c(D,B))
3. = min(2, 1+2)>
4. = min(2, 3)
5. The minimum value is 2. Therefore, the currently shortest path from A to B is 2.

b) Calculating shortest path from A to C

1. v = C, w = D
2. D(B) = min(D(C) , D(D) + c(D,C))
3. = min(5, 1+3)
4. = min(5, 4)
5. The minimum value is 4. Therefore, the currently shortest path from A to C is 4.</p>

c) Calculating shortest path from A to E

1. v = E, w = D
2. D(B) = min(D(E) , D(D) + c(D,E))
3. = min(∞, 1+1)
4. = min(∞, 2)
5. The minimum value is 2. Therefore, the currently shortest path from A to E is 2.

Note: The vertex D has no direct link to vertex E. Therefore, the value of D(F) is infinity.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞

Step 3:

In the above table, we observe that both E and B have the least cost path in step 2. Let's consider the E vertex. Now, we determine the least cost path of remaining vertices through E.

a) Calculating the shortest path from A to B.

1. v = B, w = E
2. D(B) = min(D(B) , D(E) + c(E,B))
3. = min(2 , 2+ ∞)
4. = min(2, ∞)
5. The minimum value is 2. Therefore, the currently shortest path from A to B is 2.

b) Calculating the shortest path from A to C.

1. v = C, w = E
2. D(B) = min(D(C) , D(E) + c(E,C))
3. = min(4 , 2+1)
4. = min(4,3)
5. The minimum value is 3. Therefore, the currently shortest path from A to C is 3.

c) Calculating the shortest path from A to F.

1. v = F, w = E
2. D(B) = min(D(F) , D(E) + c(E,F))
3. = min(∞ , 2+2)
4. = min(∞ ,4)
5. The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E

Step 4:

In the above table, we observe that B vertex has the least cost path in step 3. Therefore, it is added in N. Now, we determine the least cost path of remaining vertices through B.

a) Calculating the shortest path from A to C.

1. v = C, w = B
2. D(B) = min(D(C) , D(B) + c(B,C))
3. = min(3 , 2+3)
4. = min(3,5)
5. The minimum value is 3. Therefore, the currently shortest path from A to C is 3.

b) Calculating the shortest path from A to F.

1. v = F, w = B
2. D(B) = min(D(F) , D(B) + c(B,F))
3. = min(4, ∞)
4. = min(4, ∞)
5. The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E

Step 5:

In the above table, we observe that C vertex has the least cost path in step 4. Therefore, it is added in N. Now, we determine the least cost path of remaining vertices through C.

a) Calculating the shortest path from A to F.

1. v = F, w = C
2. D(B) = min(D(F) , D(C) + c(C,F))
3. = min(4, 3+5)
4. = min(4,8)
5. The minimum value is 4. Therefore, the currently shortest path from A to F is 4.

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E
5	ADEBC					4,E

Final table:

Step	N	D(B),P(B)	D(C),P(C)	D(D),P(D)	D(E),P(E)	D(F),P(F)
1	A	2,A	5,A	1,A	∞	∞
2	AD	2,A	4,D		2,D	∞
3	ADE	2,A	3,E			4,E
4	ADEB		3,E			4,E
5	ADEBC					4,E
6	ADEBCF					

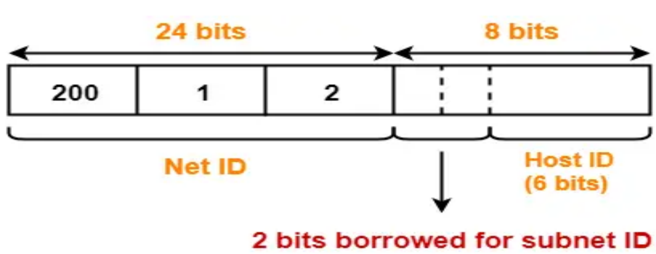
Disadvantage:

Heavy traffic is created in Line state routing due to Flooding. Flooding can cause an infinite looping, this problem can be solved by using Time-to-leave field.

13. Consider-

- We have a big single network having IP Address 200.1.2.0.
- We want to do subnetting and divide this network into 4 subnets.

Clearly, the given network belongs to class C.

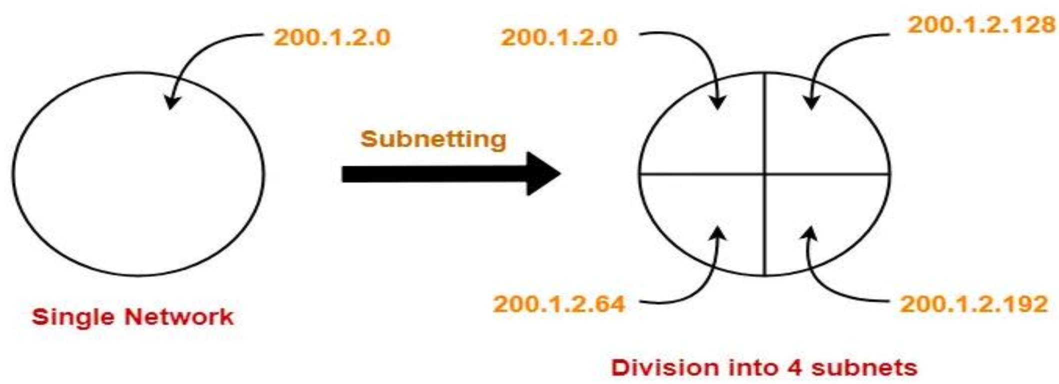


For creating four subnets and to represent their subnet IDs, we require 2 bits.

- So,
- We borrow two bits from the Host ID part.
- After borrowing two bits, Host ID part remains with only 6 bits.
- If borrowed bits = 00, then it represents the 1st subnet.
- If borrowed bits = 01, then it represents the 2nd subnet.
- If borrowed bits = 10, then it represents the 3rd subnet.
- If borrowed bits = 11, then it represents the 4th subnet.

IP Address of the four subnets are-

- 200.1.2.**00**000000 = 200.1.2.0
- 200.1.2.**01**000000 = 200.1.2.64
- 200.1.2.**10**000000 = 200.1.2.128
- 200.1.2.**11**000000 = 200.1.2.192



For 1st Subnet-

- IP Address of the subnet = 200.1.2.0
- Total number of IP Addresses = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**00**000000, 200.1.2.**00**111111] = [200.1.2.0, 200.1.2.63]
- Direct Broadcast Address = 200.1.2.**00**111111 = 200.1.2.63
- Limited Broadcast Address = 255.255.255.255

For 2nd Subnet-

- IP Address of the subnet = 200.1.2.64
- Total number of IP Addresses = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**01**000000, 200.1.2.**01**111111] = [200.1.2.64, 200.1.2.127]
- Direct Broadcast Address = 200.1.2.**01**111111 = 200.1.2.127
- Limited Broadcast Address = 255.255.255.255

For 3rd Subnet-

- IP Address of the subnet = 200.1.2.128
- Total number of IP Addresses = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**10**000000, 200.1.2.**10**111111] = [200.1.2.128, 200.1.2.191]
- Direct Broadcast Address = 200.1.2.**10**111111 = 200.1.2.191
- Limited Broadcast Address = 255.255.255.255

For 4th Subnet-

- IP Address of the subnet = 200.1.2.192
- Total number of IP Addresses = 2^6 = 64
- Total number of hosts that can be configured = 64 – 2 = 62
- Range of IP Addresses = [200.1.2.**11**000000, 200.1.2.**11**111111] = [200.1.2.192, 200.1.2.255]
- Direct Broadcast Address = 200.1.2.**11**111111 = 200.1.2.255
- Limited Broadcast Address = 255.255.255.255