CHAPTER 12

Multicasting and Multicast Routing Protocols

Exercises

1. If we want to find CIDR notation for a range of addresses when the first and last address in the block are given, we need to find the value of ***n*** (the prefix length). One way to do that is to find the maximum number of the same rightmost bits in both addresses (the potential *n*). If we assume that this part is the prefix, then the suffix should be all 0's in the first address and all 1's in the last address. If the two addresses pass this test, we have found *n* and the block can be written in CIDR notation with *n* as the maximum number of common bits. Note that the test also guarantees that the first address is divisible by the number of addresses in the block.

a. The block does not pass the test because there is one 1 in the suffix of the first address.

First Address:	11100000	00000000	00000010	00000000
Last Address:	11100000	00000000	11111111	11111111

This block can be split into several smaller CIDR blocks in the future.

b. The block does not pass the test because there are two 1's in the suffix of the first address.

First Address:	11100000	00000011	00000000	00000000
Last Address:	11100111	11111111	11111111	11111111

This block can be split into several smaller CIDR blocks in the future.

c. The block does not pass the test because there is one 1 in the suffix of the first address and one 0 in the suffix of the last address.

First Address:	11101010	00000000	00000000	00000000
Last Address:	11101110	11111111	11111111	11111111

This block can be split into several smaller CIDR blocks in the future.

3. We use binary notation to do transformation:

```
                                11011111    00011000    00111100    00001001
                                                ↓           ↓           ↓
00000001    00000000    01011110    00011000    00111100    00001001
```

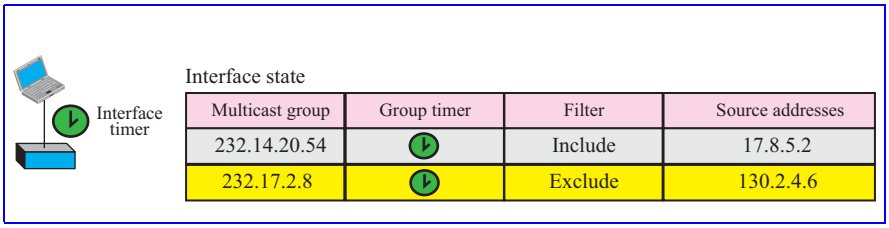
The resulting address in hexadecimal is: **01:00:5E:18:3C:09**. Using the same method, we get the same result.

5. There is no need for a *report message* to travel outside of its own network because its only purpose is to inform the next router in the spanning tree of group membership. There is no need for a *query message* to travel outside of the local network because its only purpose is to poll the local network for membership in any groups.

7. The membership report message is **[8 + 3 × (8 + 5 × 4)] = 92** bytes long. We cannot give a general formula for the size because the number of specific addresses in each record may be different.

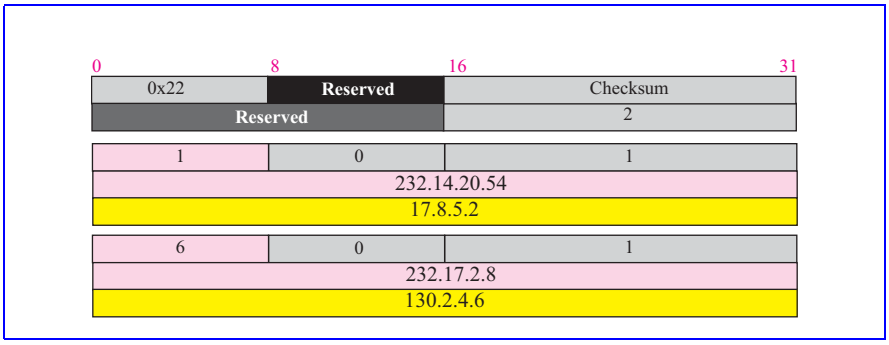
9. See Figure 12.E9.

Figure 12.E9 *Solution to Exercise 9*



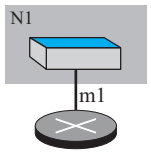
11. Figure 12.E11 shows the header and group records. We use record type 6 to exclude any message from the only source defines in the previous report.

Figure 12.E11 *Solution to Exercise 11*



13. Figure 12.E13 shows the state.

Figure 12.E13 *Solution to Exercise 13*



State for interface m1

Multicast group	Timer	Filter	Source addresses
232.77.67.60		Include	

15.

a. This is a **query** message.

b. The checksum is **0xEEFF** or $(1110\ 1110\ 1111\ 1111)_2$.

c. The group address is $(E80E1508)_{16}$ or **232.14.21.8**.

17. This is a repeat of Exercise 4, but we repeat the solution below. In each case, we find the rightmost three bytes, subtract 8 from the leftmost digit if it is greater than 8, and add result to the starting Ethernet address. Note that *a* and *b* represent the same Ethernet address.

a.

Original IP Address:					**E0**	**:**	**12**	**:**	**48**	**:**	**08**
After subtraction:							**12**	**:**	**48**	**:**	**08**
Result:	**01**	**:**	**00**	**:**	**5E**	**:**	**12**	**:**	**48**	**:**	**08**

b.

Original IP Address:					**EB**	**:**	**12**	**:**	**48**	**:**	**08**
After subtraction:							**12**	**:**	**48**	**:**	**08**
Result:	**01**	**:**	**00**	**:**	**5E**	**:**	**12**	**:**	**48**	**:**	**08**

c.

Original IP Address:					**ED**	**:**	**12**	**:**	**06**	**:**	**58**
After subtraction:							**12**	**:**	**06**	**:**	**58**
Result:	**01**	**:**	**00**	**:**	**5E**	**:**	**12**	**:**	**06**	**:**	**08**

d.

Original IP Address:					**94**	**:**	**58**	**:**	**0C**	**:**	**08**
After subtraction:							**58**	**:**	**0C**	**:**	**08**
Result:	**01**	**:**	**00**	**:**	**5E**	**:**	**58**	**:**	**0C**	**:**	**08**

19. See Tables below:

R2 Table	
Dest.	***Next-hop***
G1	**---, R1, R3**
G2	**---, R3**
G3	**---, R1**
G4	**R1, R3**
G5	**---, R1**

R3 Table	
Dest.	***Next-hop***
G1	**---, R2**
G2	**---, R2**
G3	**---, R2**
G4	**---, R2**
G5	**---, R2**

R4 Table	
Dest.	***Next-hop***
G1	**---, R1**
G2	**R1**
G3	**---, R1**
G4	**---, R1**
G5	**---, R1**

21. Router B is the designated parent router.

23. Yes, RPB creates a shortest path tree because a multicast packet reaches every network and that network receives only one copy of the packet. The leaves of the tree are the networks.