Unit 4 - Network Layer

AGENDA

- **IPv4 Addresses**: Address Space, Notations, Classful Addressing, Classless Addressing,
- Network Address Translation (NAT)
- Need for Network Layer
- Internet as a Datagram Network
- Internet as a Connectionless Network
- IPv4: Segment Header Format,
- Datagram, Fragmentation, Checksum, Options
- **IPv6:** Advantages
- Packet Format
- Extension Headers
- Forwarding Techniques
- Forwarding Process
- Routing Table.

Network Layer

- The network layer is responsible for the delivery of individual packets from the source to the destination host.
- If a packet travels through the Internet, we need a global addressing system to help distinguish the source and destination.
- This scheme is called logical addressing. We use the term IP address to mean a logical address in the network layer of the TCP/IP protocol suite.
- The network layer adds a header that includes the logical addresses of the sender and receiver to the packet corning from the upper layer.

Network Layer

- The Internet addresses are 32 bits in length; this gives us a maximum of 2^{32} addresses. These addresses are referred to as IPv4 (IP version 4) addresses or simply IP addresses if there is no confusion.
- The need for more addresses, in addition to other concerns about the IP layer, motivated a new design of the IP layer called the new generation of IP or IPv6 (lP version 6).
- In this version, the Internet uses 128-bit addresses that give much greater flexibility in address allocation. These addresses are referred to as IPv6 (IP version 6) addresses.

IPv4 ADDRESSES

- The identifier used in the IP layer of the TCP/IP protocol suite to identify the connection of each device to the Internet is called the Internet address or **IP address**.
- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet.
- The IP address is the address of the connection, not the host or the router, because if the device is moved to another network, the IP address may be changed.
- IPv4 addresses are unique in the sense that each address defines one, and only one, connection to the Internet.
- If a device has two connections to the Internet, via two networks, it has two IPv4 addresses.

IPv4 ADDRESSES

- IPv4 is also a connection-less protocol that uses the datagram approach. Each datagram is handled independently, and each can follow a different route to the destination.
- This implies that datagrams sent by the same source to the same destination could arrive out of order.
- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.

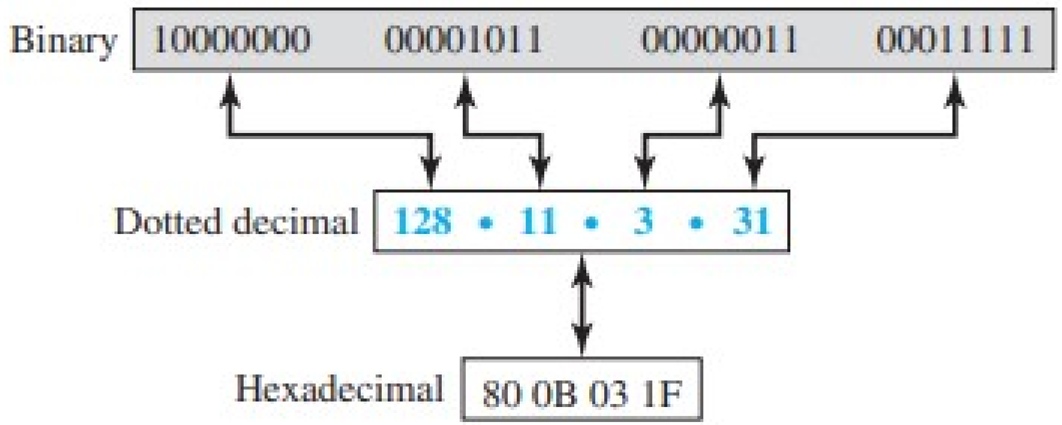
IPv4 ADDRESSES

- **Address Space**
- A protocol like IPv4 that defines addresses has an address space.
- An address space is the total number of addresses used by the protocol.
- If a protocol uses b bits to define an address, the address space is 2^b because each bit can have two different values (0 or 1).
- IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than four billion).
- If there were no restrictions, more than 4 billion devices could be connected to the Internet.

Dotted-decimal notation and binary notation for an IPv4 address

- There are three common notations to show an IPv4 address: binary notation (base 2), dotted-decimal notation (base 256), and hexadecimal notation (base 16).

Figure 18.16 *Three different notations in IPv4 addressing*



Dotted-decimal notation and binary notation for an IPv4 address

- In binary notation, an IPv4 address is displayed as 32 bits. To make the address more readable, one or more spaces are usually inserted between each octet (8 bits).
- Each octet is often referred to as a byte. To make the IPv4 address more compact and easier to read, it is usually written in decimal form with a decimal point (dot) separating the bytes.
- This format is referred to as dotted-decimal notation.
- We sometimes see an IPv4 address in hexadecimal notation. Each hexadecimal digit is equivalent to four bits. This means that a 32-bit address has 8 hexadecimal digits.

IPv4 ADDRESSES

- Change the following IPv4 addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

- We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation.

a. 129.11.11.239

b. 193.131.27.255

IPv4 ADDRESSES

- Change the following IPv4 addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 221.34.7.82

- We replace each decimal number with its binary equivalent

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

IPv4 ADDRESSES

- Find the error, if any, in the following IPv4 addresses.

a. 111.56.045.78

b. 221.34.7.8.20

c. 75.45.301.14

d. 11100010.23.14.67

Solution

- a. There must be no leading zero (045).
- b. There can be no more than four numbers. (One extra number)
- c. Each number needs to be less than or equal to 255. (301)
- d. A mixture of binary notation and dotted-decimal notation is not allowed.

IPv4 ADDRESSES

- **Hierarchy in Addressing**
- A 32-bit IPv4 address is also hierarchical, but divided only into two parts. The first part of the address, called the prefix, defines the network; the second part of the address, called the suffix, defines the node (connection of a device to the Internet).
- Figure 18.17 shows the prefix and suffix of a 32-bit IPv4 address. The prefix length is n bits and the suffix length is (32 − n) bits.
- A prefix can be fixed length or variable length. The network identifier in the IPv4 was first designed as a fixed-length prefix.
- This scheme, which is now obsolete, is referred to as classful addressing. The new scheme, which is referred to as classless addressing, uses a variable-length network prefix.

IPv4 ADDRESSES

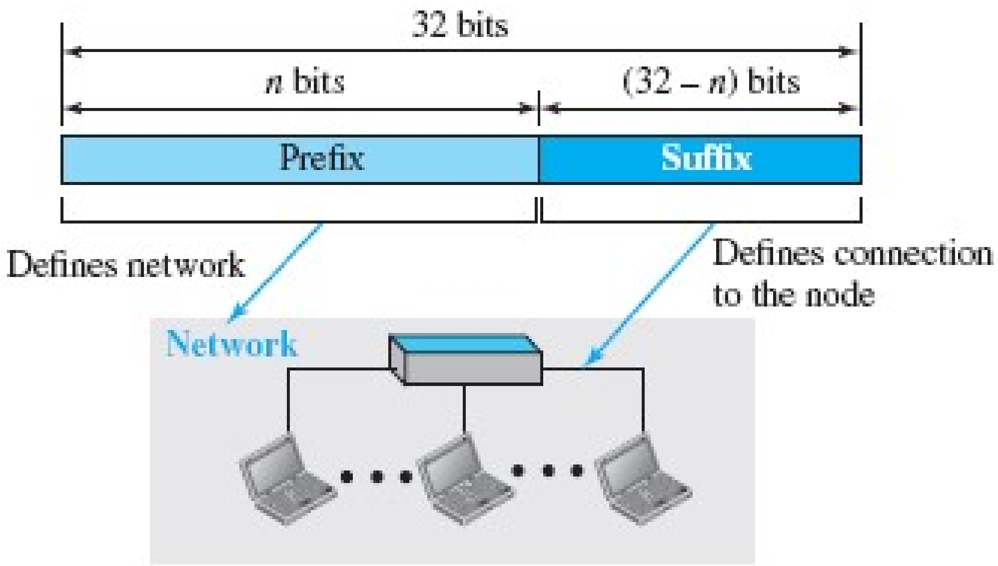


Figure 18.17 Hieararchy in Addressing

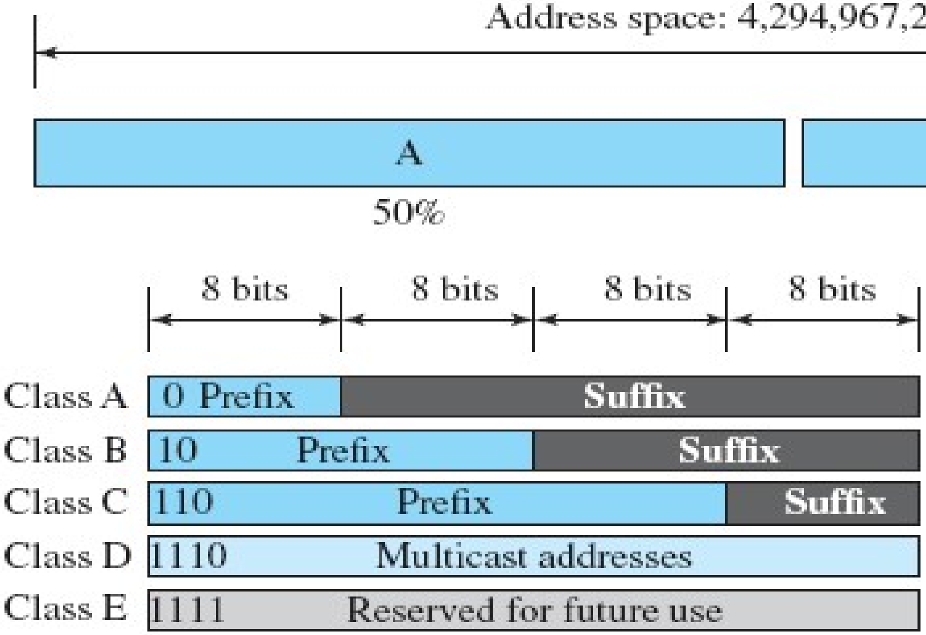
IPv4 ADDRESSES

- **Classful Addressing**
- When the Internet started, an IPv4 address was designed with a fixed-length prefix, but to accommodate both small and large networks, three fixed-length prefixes were designed instead of one (n = 8, n = 16, and n = 24).
- The whole address space was divided into five classes (class A, B, C, D, and E), as shown in Figure 18.18.
- This scheme is referred to as classful addressing.

In classful addressing, the address space is divided into five classes: A, B, C, D, and E.

Classful Addressing

Figure 18.18 *Occupation of the address space in classful addressing*



Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

Finding the classes in binary and dotted-decimal notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

Classful Addressing

- In class A, the network length is 8 bits, but since the first bit, which is 0, defines the class, we can have only seven bits as the network identifier.
- This means there are only $2^7 = 128$ networks in the world that can have a class A address.
- In class B, the network length is 16 bits, but since the first two bits, which are (10)2, define the class, we can have only 14 bits as the network identifier.
- This means there are only $2^{14} = 16{,}384$ networks in the world that can have a class B address.
- All addresses that start with (110)2 belong to class C.

Classful Addressing

- In class C, the network length is 24 bits, but since three bits define the class, we can have only 21 bits as the network identifier.
- This means there are 2^{21} = 2,097,152 networks in the world that can have a class C address.
- Class D is not divided into prefix and suffix. It is used for multicast addresses. All addresses that start with 1111 in binary belong to class E.
- Class D is used as reserved class.

Classful Addressing

- **Address Depletion**
- The reason that classful addressing has become obsolete is address depletion.
- Since the addresses were not distributed properly, the Internet was faced with the problem of the addresses being rapidly used up, resulting in no more addresses available for organizations and individuals that needed to be connected to the Internet.

In classful addressing, a large part of the available addresses were wasted.

Classful Addressing

- **Address Depletion**
- To understand the problem, let us think about class A. This class can be assigned to only 128 organizations in the world, but each organization needs to have a single network (seen by the rest of the world) with 16,777,216 nodes (computers in this single network).
- Since there may be only a few organizations that are this large, most of the addresses in this class were wasted (unused).
- Class B addresses were designed for midsize organizations, but many of the addresses in this class also remained unused.
- Class C addresses have a completely different flaw in design. The number of addresses that can be used in each network (256) was so small that most companies were not comfortable using a block in this address class.
- Class E addresses were almost never used, wasting the whole class.

Classes and Blocks

Table 19.1 ***Number of blocks and block size in classful IPv4 addressing***

Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

- One problem with classful addressing is that each class is divided into a fixed number of blocks with each block having a fixed size as shown in Table

Classes and Blocks

- We can see the flaw in this design. A block in class A address is too large for almost any organization.
- This means most of the addresses in class A were wasted and were not used.
- A block in class B is also very large, probably too large for many of the organizations that received a class B block.
- A block in class C is probably too small for many organizations.
- Class D addresses were designed for multicasting

Netid and Hostid

- In classful addressing, an IP address in class A, B, or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address.
- Figure 19.2 shows some netid and hostid bytes. The netid is in color, the hostid is in white. Note that the concept does not apply to classes D and E.
- In class A, one byte defines the netid and three bytes define the hostid. In class B, two bytes define the netid and two bytes define the hostid. In class C, three bytes define the netid and one byte defines the hostid.

Table 19.2 ***Default masks for classful addressing***

Class	*Binary*	*Dotted-Decimal*	*CIDR*
A	**11111111** 00000000 00000000 00000000	**255**.0.0.0	/8
B	**11111111 11111111** 00000000 00000000	**255.255**.0.0	/16
C	**11111111 11111111 11111111** 00000000	**255.255.255**.0	/24

Netid and Hostid

- Mask
- Although the length of the netid and hostid (in bits) is predetermined in classful addressing, we can also use a mask (also called the default mask), a 32-bit number made of contiguous 1s followed by contiguous as.
- The masks for classes A, B, and C are shown in Table 19.2. The concept does not apply to classes D and E.

Classful Addressing

- **Subnetting and Supernetting**
- To alleviate address depletion, two strategies were proposed and, to some extent, implemented: subnetting and supernetting.
- In subnetting, a class A or class B block is divided into several subnets. Each subnet has a larger prefix length than the original network.
- If all of the addresses in a network are not used, subnetting allows the addresses to be divided among several organizations.
- This idea did not work because most large organizations were not happy about dividing the block and giving some of the unused addresses to smaller organizations.
- While subnetting was devised to divide a large block into smaller ones, supernetting was devised to combine several class C blocks into a larger block to be attractive to organizations that need more than the 256 addresses available in a class C block.
- This idea did not work either because it makes the routing of packets more difficult.

Classful Addressing

- **Advantage of Classful Addressing**
- Given an address, we can easily find the class of the address and, since the prefix length for each class is fixed, we can find the prefix length immediately.
- In other words, the prefix length in classful addressing is inherent in the address; no extra information is needed to extract the prefix and the suffix.

Classful addressing, which is almost obsolete, is replaced with classless addressing.

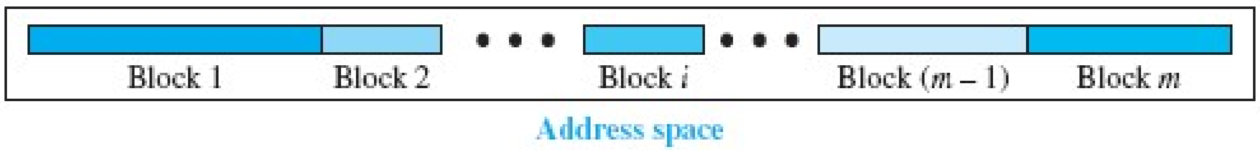
Classless Addressing

- With the growth of the Internet, it was clear that a larger address space was needed as a long-term solution.
- The larger address space, however, requires that the length of IP addresses also be increased, which means the format of the IP packets needs to be changed.
- Although the long-range solution has already been devised and is called IPv6 (discussed later), a short-term solution was also devised to use the same address space but to change the distribution of addresses to provide a fair share to each organization.
- The short-term solution still uses IPv4 addresses, but it is called classless addressing.
- In classless addressing, variable-length blocks are used that belong to no classes. We can have a block of 1 address, 2 addresses, 4 addresses, 128 addresses, and so on.

Classless Addressing

- In classless addressing, the whole address space is divided into variablelength blocks. The prefix in an address defines the block (network); the suffix defines the node (device).
- Theoretically, we can have a block of 20, 21, 22, . . . , 232 addresses (Has to be in the power of 2).

Figure 18.19 *Variable-length blocks in classless addressing*

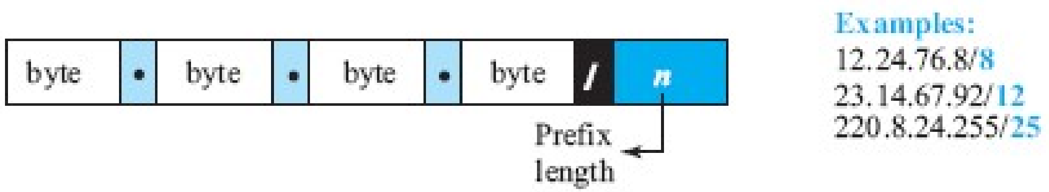


- We can have a prefix length that ranges from 0 to 32. The size of the network is inversely proportional to the length of the prefix.
- A small prefix means a larger network; a large prefix means a smaller network.
- The idea of classless addressing can be easily applied to classful addressing. In other words, classful addressing is a special case of classless addressing.

Classless Addressing

- **<u>Prefix Length: Slash Notation</u>**
- How to find the prefix length if an address is given. In this case, the prefix length, n, is added to the address, separated by a slash.
- The notation is informally referred to as slash notation and formally as **c**lassless **i**nter-**d**omain **r**outing or CIDR (pronounced cider) strategy.
- An address in classless addressing can then be represented as shown in Figure 18.20.

Figure 18.20 *Slash notation (CIDR)*



Classless Addressing

- **<u>Prefix Length: Slash Notation</u>**
- Extracting Information from an Address Given any address in the block, we normally like to know three pieces of information about the block to which the address belongs: the number of addresses, the first address in the block, and the last address.
- Since the value of prefix length, n, is given, we can easily find these three pieces of information, as shown in Figure 18.21.
- 1. The number of addresses in the block is found as $N = 2^{(32-n)}$.
- 2. To find the first address, we keep the n leftmost bits and set the (32 − n) right most bits all to 0s.
- 3. To find the last address, we keep the n leftmost bits and set the (32 − n) rightmost bits all to 1s.

Note: IP has 32 bits, and number after the slash tells you where does the network part end, host part starts. and /24 says that first 24 bits are used for network designation, while last 8 bits are used for various hosts inside that network

Example 18.1

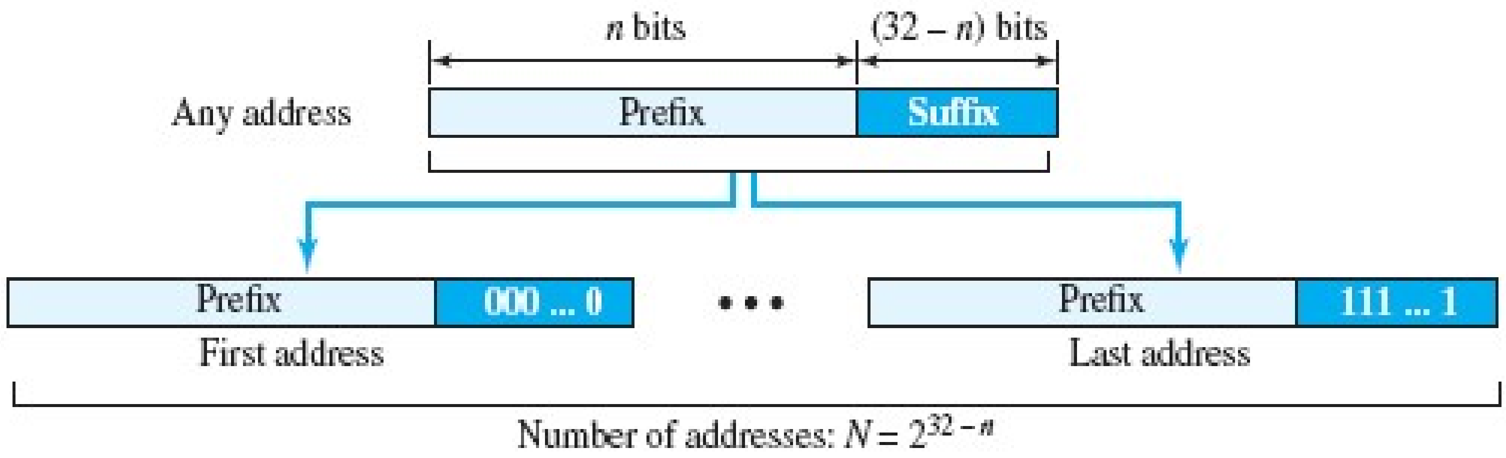
A classless address is given as 167.199.170.82/**27**. We can find the above three pieces of information as follows. The number of addresses in the network is $2^{32-n} = 2^5 = 32$ addresses.

The first address can be found by keeping the first 27 bits and changing the rest of the bits to 0s.

Address: 167.199.170.82/**27**	10100111	11000111	10101010	01010010
First address: 167.199.170.64/**27**	10100111	11000111	10101010	01000000

The last address can be found by keeping the first 27 bits and changing the rest of the bits to 1s.

Figure 18.21 *Information extraction in classless addressing*



Classless Addressing

- **Address Mask**
- Another way to find the first and last addresses in the block is to use the address mask.
- The address mask is a 32-bit number in which the n leftmost bits are set to 1s and the rest of the bits $(32 - n)$ are set to 0s.
- A computer can easily find the address mask because it is the complement of $(2^{32 - n} - 1)$.
- The reason for defining a mask in this way is that it can be used by a computer program to extract the information in a block, using the three bit-wise operations NOT, AND, and OR.
- 1. The number of addresses in the block N = NOT (mask) + 1.
- 2. The first address in the block = (Any address in the block) AND (mask).
- 3. The last address in the block = (Any address in the block) OR [(NOT (mask)].

Classless Addressing

- **Address Mask**

Example 18.2

We repeat Example 18.1 using the mask. The mask in dotted-decimal notation is 256.256.256.224. The AND, OR, and NOT operations can be applied to individual bytes using calculators and applets at the book website.

Number of addresses in the block:	N = **NOT** (mask) + 1 = 0.0.0.31 + 1 = 32 addresses
First address:	First = (address) **AND** (mask) = 167.199.170.82
Last address:	Last = (address) **OR** (**NOT** mask) = 167.199.170.255

Classless Addressing

- <u>**Address Mask**</u>

- In Example 19.5 the /28 can be represented as 11111111 11111111 11111111 11110000 (twenty-eight Is and four Os). Find
- a) The first address, b) The last address, c) The number of addresses
- **Solution**
- **a.** The first address can be found by ANDing the given addresses with the mask. ANDing here is done bit by bit. The result of ANDing 2 bits is 1 if both bits are Is; the result is 0 otherwise.
- **Address**: 11001101 00010000 00100101 00100111
- **Mask**: 11111111 11111111 11111111 11110000
- **First address**: 11001101 00010000 00100101 00100000

Classless Addressing

- **Address Mask**

- **b.** The last address can be found by ORing the given addresses with the complement of the mask. ORing here is done bit by bit.
- The result of ORing 2 bits is 0 if both bits are Os; the result is 1 otherwise. The complement of a number is found by changing each 1 to 0 and each 0 to 1
- **Address:** 11001101 00010000 00100101 00100111
- **Mask complement**: 00000000 00000000 00000000 00001111
- **Last address**: 11001101 00010000 00100101 00101111
- **c.** The number of addresses can be found by complementing the mask, interpreting it as a decimal number, and adding 1 to it.
- **Mask complement**: 000000000 00000000 00000000 00001111
- **Number of addresses**: 15 + 1 =16

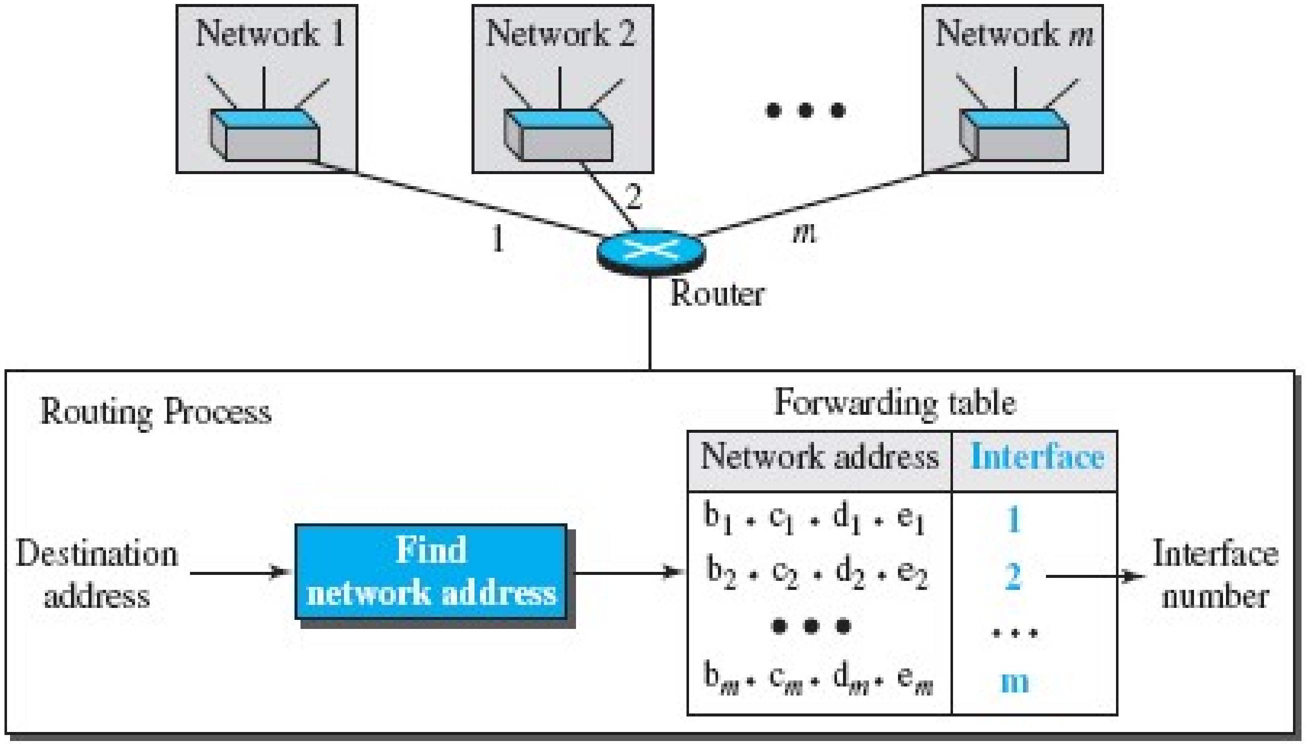
Classless Addressing

- **Network Address**
- Network address is used in routing a packet to its destination network.
- For the moment, let us assume that an internet is made of m networks and a router with m interfaces.
- When a packet arrives at the router from any source host, the router needs to know to which network the packet should be sent: from which interface the packet should be sent out.
- When the packet arrives at the network, it reaches its destination host using another strategy that we discuss later.
- Figure 18.22 shows the idea. After the network address has been found, the router consults its forwarding table to find the corresponding interface from which the packet should be sent out.
- The network address is actually the identifier of the network; each network is identified by its network address.

Classless Addressing

- **Network Address**

Figure 18.22 *Network address*



Classless Addressing

- **Network Address Block Allocation**
- The main responsibility of block allocation is given to a global authority called the Internet Corporation for Assigned Names and Numbers (ICANN).
- ICANN does not allocate addresses to individual Internet users but to an ISP (or a larger organization that is considered an ISP in this case).
- For the proper operation of the CIDR, two restrictions need to be applied to the allocated block.
- 1. The number of requested addresses, N, needs to be a power of 2. The reason is that $N = 2^{32-n}$ or $n = 32 - \log_2 N$. If N is not a power of 2, we cannot have an integer value for n.
- 2. The requested block needs to be allocated where there is an adequate number of contiguous addresses available in the address space. However, there is a restriction that the first address needs to be divisible by the number of addresses in the block.
- The reason is that the first address needs to be the prefix followed by (32 − n) number of 0s. The decimal value of the first address is then,
- first address =(prefix in decimal) × 232 − n = (prefix in decimal) □N

Example 19.10

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

a. The first group has 64 customers; each needs 256 addresses.

b. The second group has 128 customers; each needs 128 addresses.

c. The third group has 128 customers; each needs 64 addresses.

Figure 19.9 *An example of address allocation and distribution by an IS?*