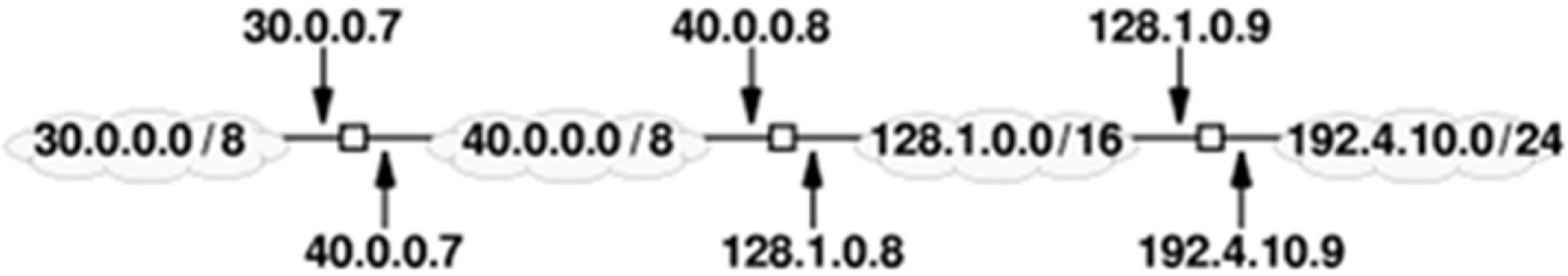
Subnet and Classless IP addressing

- A problem with *Classful addressing* is that it results in an unequal division of the *IP address space.*
- As the global Internet has grown the use of *classful addressing* has become problematic:
  - The IP address space is being exhausted,
  - Many addresses remain unused,
  - Refer to examples in class.
- Two new addressing methods were introduced to overcome this: *subnet addressing* and *classless addressing*:
  - Here the division between the prefix and suffix portions can occur on any bit boundary.

Subnet and Classless IP addressing

- To facilitate Classless Addressing an additional piece of information is allocated with each address range.
- This is known as an *address mask* or *subnet mask.*
- Masks are 32-bit values that enable the router to **compute** the network prefix from any given IP address.
- They are comprised of a contiguous sequence (unbroken sequence) of 1 bits followed by a contiguous sequence of 0 bits.
- Just like IP addresses they can be represented in dotted-decimal notation :
  - Refer to the following slide for some examples.

Example Classless IP Addressing Allocation



Network Number	Mask
30.0.0.0	255.0.0.0
40.0.0.0	255.0.0.0
128.1.0.0	255.255.0.0
192.4.10.0	255.255.255.0

- Notice how each router is assigned an IP address on each of the networks to which it attaches.

Subnet and Classless IP addressing

- Note the *address mask* or *subnet mask* for the 192.4.10.0/24 network:
  - This notation is known as Classless Inter-Domain Routing (CIDR).
- The /24 (slash 24) means that the mask is comprised of 24 ONE bits followed by 8 (32-24) ZERO bits*:*
  - In *dotted-decimal* notation this mask can be represented as:

    255.255.255.0
  - The first three octets are all ONEs and the last octet is all ZEROs.
  - The table below the network diagram shows the network address for each sub-net and its mask in *dotted-decimal notation*.

Using Address Masks to route packets

- Following on from the previous discussion on routing.
- Recall that for any given destination IP address, the router must determine the *network prefix* portion.
- Having extracted the *network prefix* the router consults its *routing table*.
- The use of Classless addressing changes the way routers calculate the *network prefix* portion of a destination IP address.

Using Address Masks to route packets

- For an incoming packet with a destination IP address the router tests the following condition: **A = = (D&M)**
- Where:
  - **A** is the IP address (network number) of networks that the router knows about,
  - **M** is the mask associated with the network, and,
  - **D** represents a destination IP address that the router needs to make a routing decision.

Address Masks

- For example consider the following:

A = 11000000 00000100 00001010 00000000

D = 11000000 00000100 00001010 00000011

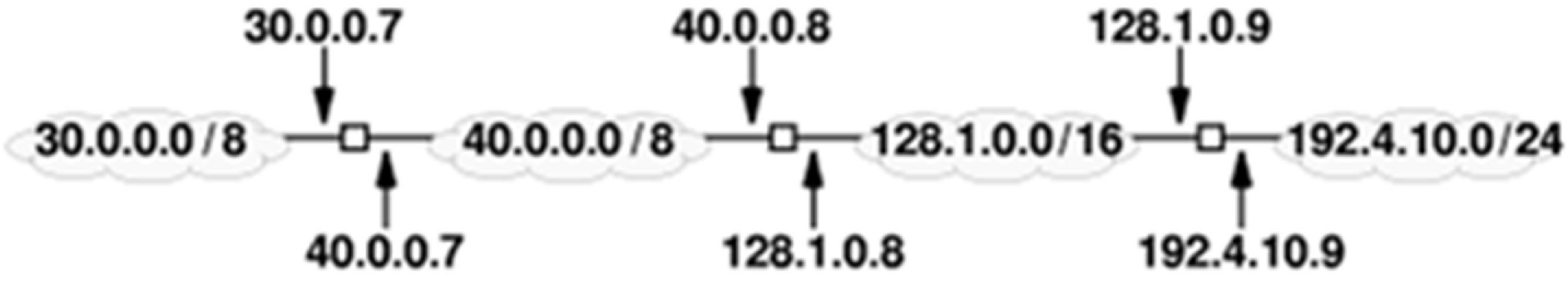
M = 11111111 11111111 11111111 00000000

- The mask, **M**, is ‘applied’ to the Destination IP address, **D**
  - i.e. **D & M**
  - The **AND** operation effectively zeros out the last eight bits of **D**.
- The result is then compared to the **A** address.

Address Masks

- If they match then the Destination IP address, **D** is said to **belong** to the network, **A**:
  - The packet containing the Destination IP address, **D**, is then **routed** towards network **A,**
  - The packet is routed to the address indicated by the ***Next Hop*** field in the routing table (refer to next slide).
- Otherwise, the next entry in the routing table will be tried using the above approach.

Example IP Routing Table using Classless Addressing



(a)

Destination	Mask	Next Hop
30.0.0.0	255.0.0.0	40.0.0.7
40.0.0.0	255.0.0.0	deliver direct
128.1.0.0	255.255.0.0	deliver direct
192.4.10.0	255.255.255.0	128.1.0.9

Address Masks

- From the previous slide notice the following:
  - Each network address is written in CIDR notation.
  - Routers have multiple IP addresses; one for each of the networks it attaches to.
  - Below the network diagram is a high-level representation of the *Routing Table* for the router in the middle.
  - The Next Hop field identifies which destination networks are directly connected and which are remotely connected

Address Masks

- The discussions in class will focus on:
  - Identifying the routing tables for each of the other routers.
  - The process of *routing* of packets arriving at each of the routers towards their final destinations.

Classless Addressing and the IP Address Space

- *Classless addressing* makes more efficient use of the IP address space
- Consider an example of a single *class B prefix* (16-bit prefix): 128.211.0.0
- Previously with *classful addressing* this network address could only be used to identify a single network comprising approximately 65K host addresses.

Classless Addressing V's IP Address Space

- With <u>*classless*</u> *addressing* the network address can be sub-divided using *network masks* to cater for <u>**sub-networks**</u> of varying sizes:
  - For example a 28-bit *address mask* can be used as follows:

    128.211.0.0/28

    128.211.0.16/28

    128.211.0.32/28

- Whilst each sub-network has the same size mask (28 bits), the network prefixes are different (and *unique).*
- In addition most of the original address is still available.

Basic Subnetting

- **<u>Sub-netting</u>** allows for creating **multiple** logical networks from a single address block:
  - Sub-nets are formed by 'borrowing' one or more of the *host-suffix* bits and using them as *network-prefix* bits.
  - This is achieved by extending the network mask.
  - The more host bits borrowed, the more sub-nets can be defined.

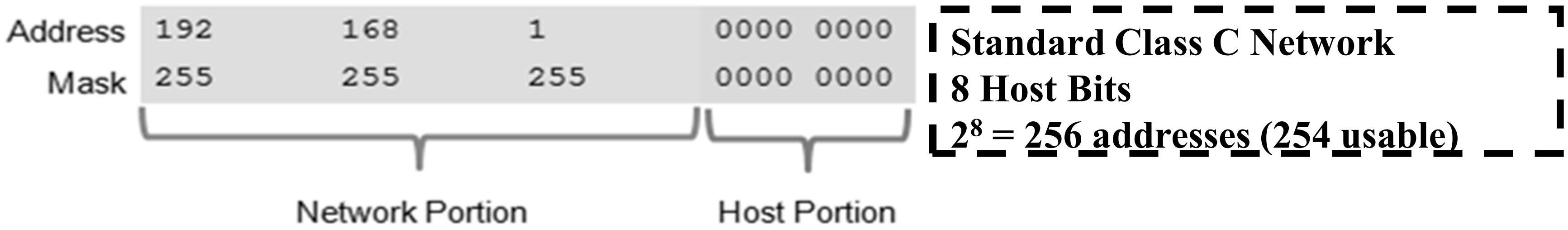
Basic Subnetting

- For each host-bit *'borrowed'*, the number of sub-networks available is doubled:
  - For example, with one borrowed host-bit, 2 sub-nets are created, with two borrowed host-bits four sub-nets are created etc.
  - However, with each host-bit borrowed, fewer **host addresses** are available per sub-net. In other words the size of the sub-networks reduces.

Basic Subnetting

- For example if **n** host-bits are borrowed:
  - The number of sub-nets created is $\mathbf{2^n}$.
  - The total number of addresses per sub-net is $\mathbf{2^m}$ (where **m** = the number of host-bits left).
  - The number of **<u>usable</u>** host addresses: $\mathbf{2^m - 2}$;
    - The first address in the block is used to identify the network,
    - The last address in the block is the *Broadcast Address* (explained in class).

Basic Subnetting



- This is a standard Class C network address:
  - The network address is 192.168.1.0/24
  - The mask in dotted-decimal notation is: 255.255.255.0
  - There are 8 host-bits which gives $\mathbf{2^8}$ (256) addresses of which 254 are usable for actual host addresses.

Basic Subnetting

- Borrowing one host-bit creates two ($2^1 = 2$) sub-nets as follows:

Original	192.	168.	1.	0	000 0000
Mask	255.	255.	255.	0	000 0000

Borrowing 1 Bit from the host portion creates 2 subnets with the same subnet mask:

Subnet 0

Network 192.168.1.**0-127/25**

Mask: 255.255.255.**128**

Subnet 1

Network 192.168.1.**128-255/25**

Mask: 255.255.255.**128**

Each sub-net has 7 Host-bits left giving 2^7 (or 128) addresses of which 126 are usable.

Basic Subnetting

- From the above calculations the following *Address Table* can be derived:

Subnet	**Network Address**	**Host Addresses**	**Broadcast Address**	**Mask**
0	192.168.1.0	192.168.1.1 – 126	192.168.1.127	/25
1	192.168.1.128	192.168.1.129 - 254	192.168.1.255	/25

Basic Subnetting using the ***Magic Number***

- An alternative approach to deriving the *Address Table* is using *The Magic Number*:
  - The magic number is the number of addresses to be created in each sub-network to include: the ***network number***, the ***broadcast address*** and, the ***host range***.
- This number can be determined from the network mask for the sub-nets to be created.

Basic Subnetting using the ***Magic Number***

- Consider the Class C address:192.168.1.0/24:
  - With 8 host bits there are 256 addresses of which 254 are usable host addresses.
- You are required to divide this address space into two equal portions to create two sub-nets:
  - To create two sub-nets one bit will need to be borrowed from the host portion.
  - This requires a sub-net mask of /25 or 255.255.255.128
  - This division will create two sub-nets each containing 128 addresses i.e. 25 network bits and 7 host bits.

Basic Subnetting using the ***Magic Number***

- To determine the Magic Number look for the right-most ***non-zero*** octet in the sub-net mask:
  - The last octet matches this criterion.
  - Subtract this octet from 256 as follows:

    256 – 128 = 128 which is *The Magic Number.*

- With the *Magic Number* the Address Table is easy to complete as per the following slide.

Basic Subnetting using the ***Magic Number***

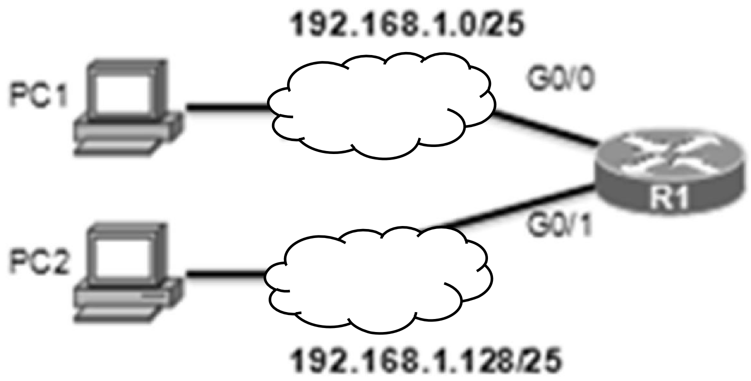
Subnet	**Network Address**	**Host Addresses**	**Broadcast Address**	**Mask**
0	192.168.1.0	192.168.1.1 – 126	192.168.1.127	/25
1	192.168.1.128	192.168.1.129 - 254	192.168.1.255	/25

- By simply adding the *Magic Number* to the starting address (192.168.1.0) the next Sub-network address can be derived.

Subnets in Use

Subnet 0

Network 192.168.1.**0-127**/**25**



Subnet 1

Network 192.168.1.**128-255**/**25**

Address Range for 192.168.1.0/25 Subnet

All Host bits = 0

Network Address	192. 168. 1. 0 000 0000	= 192.168.1.0
First Host Address	192. 168. 1. 0 000 0001	= 192.168.1.1
Last Host Address	192. 168. 1. 0 111 1110	= 192.168.1.126
Broadcast Address	192. 168. 1. 0 111 1111	= 192.168.1.127

All Host bits = 1

Address Range for 192.168.1.128/25 Subnet

All Host bits = 0

Network Address	192. 168. 1. 1 000 0000	= 192.168.1.128
First Host Address	192. 168. 1. 1 000 0001	= 192.168.1.129
Last Host Address	192. 168. 1. 1 111 1110	= 192.168.1.254
Broadcast Address	192. 168. 1. 1 111 1111	= 192.168.1.255

All Host bits = 1

Example basic sub-netting

- Given an address block of 192.168.1.0 /24, it is required to divide this network into 4 subnets:
  - Determine the power of 2 to provide 4 sub-networks i.e. $\mathbf{2^{?} = 4}$ (Note the number of sub-nets will be a power of 2).
  - Hence **two** host-bits are required to be borrowed.
  - This leaves **6 host-bits** for host addresses i.e.

    11111111.1111111.11111111.**11**000000
  - The address masks for the new sub-nets is /26 or

    255.255.255.192

Basic Subnetting

- Using the *Magic Number* approach to determine the addresses in each sub-net:
  - Look for the last octet that is non-zero.
  - The last octet matches this criterion.
  - Subtract this octet from 256 as follows:
    - 256 – 192 = 64 which is *The Magic Number*
- The Address Table can be completed as follows:

Basic Subnetting

Subnet	Network Address	Host Addresses	Broadcast Address	Mask
0	192.168.1.0	192.168.1.1 – 62	192.168.1.63	/26
1	192.168.1.64	192.168.1.65 – 126	192.168.1.127	/26
2	192.168.1.128	192.168.1.129 – 190	192.168.1.191	/26
3	192.168.1.192	192.168.1.193 - 254	192.168.1.255	/26

- See how the *Magic Number* is used to determine the *network address* for each subnet.

Subnetting a Class B Network

- Given an address block of **172.25.0.0 /16**, we wish to divide this network into 11 subnets with each subnet catering for 3000 hosts:
  - Determine the power of 2 to provide for 3000 hosts:
  - i.e. $\mathbf{2^{?} = 3000}$
  - 2^{12} is sufficient (i.e. $2^{11} = 2048$, $2^{12} = 4096$)
  - This requires four bits to be borrowed from the second octet.
  - Leaving **12 bits** for host addresses.

Subnetting a Class B Network

- So the new Subnet mask is:

  11111111.1111111.11110000.00000000

- Or, in dotted-decimal notation:

  255.255.240.0 (/20)

- Using the *Magic Number* approach to determine the addresses in each subnet.

Subnetting a Class B Network

- With a Mask for each subnet of 255.255.240.0:
  - Look for an octet that is non-zero.
  - The second-last octet matches this criterion.
  - Subtract this octet from 256 as follows:
    - 256 – 240 = 16 which is *The Magic Number*
- The Address Table can be completed as follows:

Subnetting a Class B Network

	N/W Add	Host Addresses	Broadcast Address	Mask
0	172.25.0.0	172.25.0.1 – 172.25.15.254	172.25.15.255	/20
1	172.25.16.0	172.25.16.1 – 172.25.31.254	172.25.31.255	/20
2	172.25.32.0	172.25.32.1 – 172.25.47.254	172.25.47.255	/20
3	172.25.48.0	172.25.48.1 – 172.25.63.254	172.25.63.255	/20
4	172.25.64.0	172.25.64.1 – 172.25.79.254	172.25.79.255	/20
5	172.25.80.0	172.25.80.1 – 172.25.95.254	172.25.95.255	/20
6	172.25.96.0	172.25.96.1 – 172.25.111.254	172.25.111.255	/20
7	172.25.112.0	172.25.112.1 – 172.25.127.254	172.25.127.255	/20
8	172.25.128.0	172.25.128.1 – 172.25.143.254	172.25.143.255	/20
9	172.25.144.0	172.25.144.1 – 172.25.159.254	172.25.159.255	/20
10	172.25.160.0	172.25.160.1 – 172.25.175.254	172.25.175.255	/20
11	172.25.176.0	172.25.176.1 – 172.25.191.254	172.25.191.255	/20

Special IP Addresses

- IP defines a set of special address forms that are reserved and should never be assigned to hosts
- These include:
  - **Directed Broadcast Address**. This is defined for each physical network. A suffix of all 1 bits is added to the network prefix
  - **Limited Broadcast Address**. Here an address consisting of all 1 bits will allow a broadcast on "a single wire"
  - **This Computer Address**. An IP address consisting of all zeros refers to *this computer.* Used by hosts at *boot-up* to obtain its IP address
  - **Loopback Address**. This has a network prefix 127/8; the host suffix is irrelevant but is usually set to 1 i.e. 127.0.0.1

Routers and Multi-Homed Hosts

- Routers and *multi-homed* host computers are assigned two or more IP addresses because:
  - They have connections to multiple physical networks
  - Each IP address prefix specifies only one physical network.
- A fundamental principle of the IP addressing scheme:

"An IP address does not identify a specific computer. Instead, each IP address identifies a connection between a computer and a network. A computer with multiple network connections, e.g. a router, requires one IP address for each connection."

A Router Addressing Example