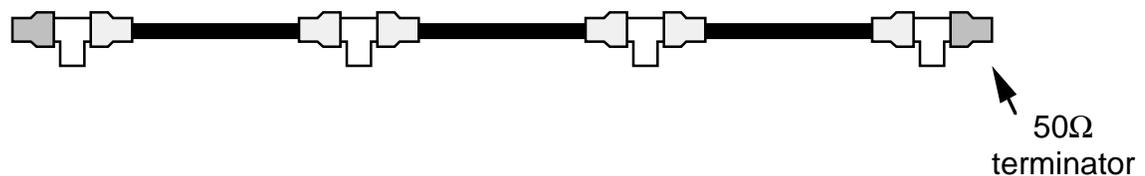# Broadcast Networks:
ETHERNET

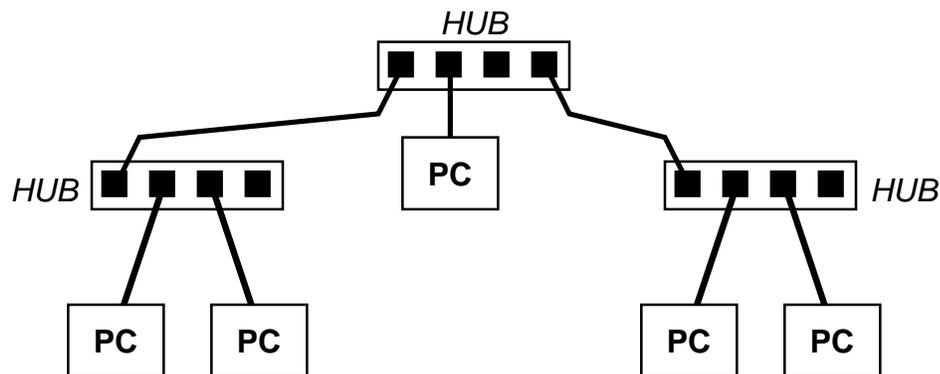**10 Base 2**   10Mbps   "Thin" coax, 185m max.

10 Base 5   10Mbps   "Thick" coax, 500m max.
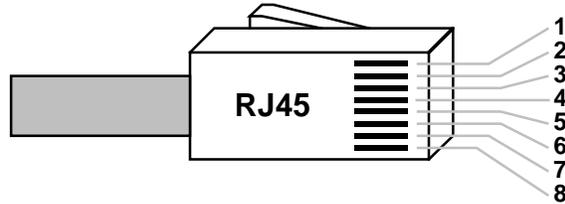
50Ω
terminator

10 Base T   10Mbps   Unshielded Twisted Pair
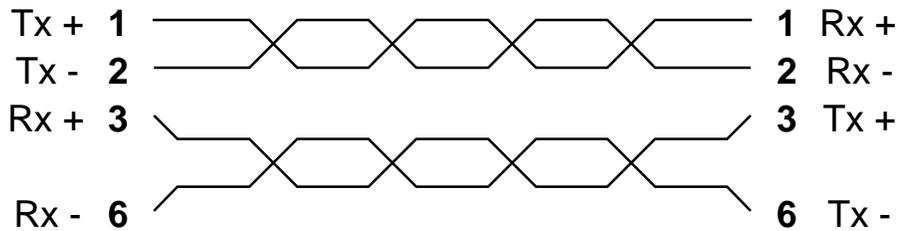
100 Base T   100Mbps CAT-5 UTP

HUB

HUB   PC   *HUB*

PC   PC   PC   PC

❏ **Max. 100m each cable**
❏ **Max. 4 hubs between any two PCs**

10 Base T Wiring

RJ45

1
2
3
4
5
6
7
8

PC: "MDI"
(media-dependent interface)

Hub: "MDI-X"
(media-dependent interface – crossover)

Tx + **1** _____ **1** Rx +
Tx - **2** _____ **2** Rx -
Rx + **3** _____ **3** Tx +

Rx - **6** _____ **6** Tx -

Must ensure pairs are twisted together!
Pick a colour scheme – and stick to it. e.g.

White/Orange 1
Orange 2
White/Green 3
Blue 4
White/Blue 5
Green 6
White/Brown 7
Brown 8

This is the colour scheme recommended in the **comp.dcom.cabling** *LAN wiring FAQ*

PC

Tx + **1** ——
Tx - **2** ——
Rx + **3** ——

Rx - **6** ——

PC

—— **1** Tx +
—— **2** Tx -
—— **3** Rx +

—— **6** Rx -

Ethernet Frames

MAC Header	Data

◀——— Max 1500 bytes ———▶
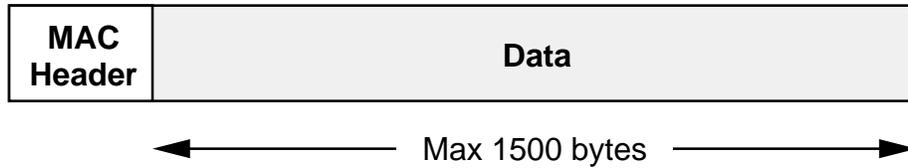
- ❏ **Card identified by "MAC address"**
- ❏ **Unique 48-bit binary number**
- ❏ **Burned into card by manufacturer**

  e.g. `00:80:F4:C7:3B:24`

- ❏ **All cards respond to "broadcast address"**

  `FF:FF:FF:FF:FF:FF`

IP Encapsulation

MAC Header	IP Header	IP Data

◀——— MTU = 1500 ———▶

Source MAC addr ***Dest MAC addr***

**We know the source MAC address
(our own) – but how do we know
the MAC address of the recipient?**
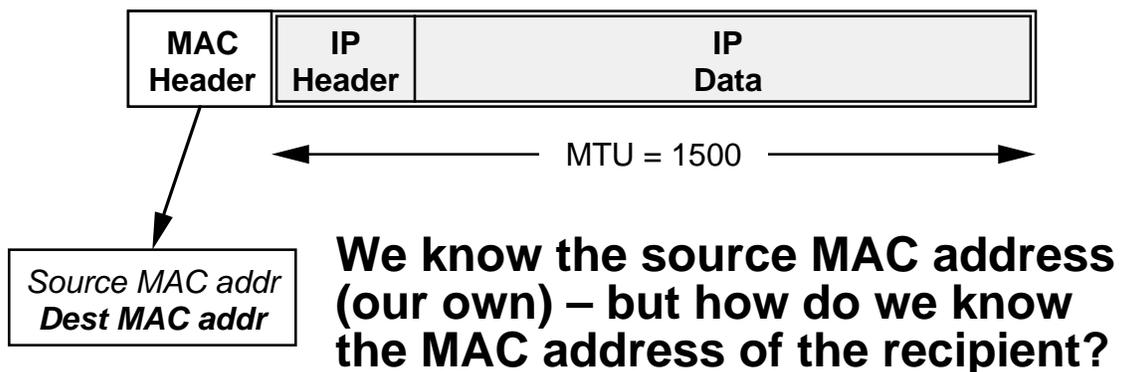
ARP - Address Resolution Protocol

We want to send a datagram to w.x.y.z

❏ **Send BROADCAST "ARP request: *w.x.y.z*"**

❏ **Machine with this IP number sends ARP response**

❏ **The ARP response contains that machine's MAC address (source MAC addr)**

❏ **So that's the MAC address we use to send the IP datagram**

NOTES:

❏ **You never ARP for a machine outside of your own network** – you ARP for the gateway that you want to forward via instead

❏ **For efficiency, every machine keeps a cache of ARP replies; they time out after typically 15 minutes** (in case the network changes)

```
arp -an          Show ARP cache
arp -d w.x.y.z   Delete cache entry
```

❏ **ARP packets are <u>not</u> IP datagrams!**