Network Computing courses

Maël Auzias

ENSIBS - UBS

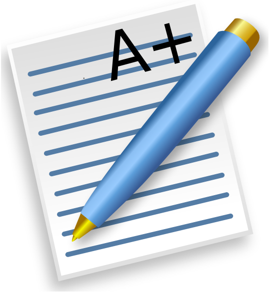
October 2014

Course details

Objectives

- How do *computers* communicate?
- What are the mechanisms **under** an HTTP request or a telegram message?
- Networks are all around us, better study them!

Course details



Evaluation

- Short test at the beginning of every lesson (5 min) ?
- Project
- Final exam (1 hour)
- All same weighting

Material

- Slides available at teaching.auzias.net (github too)

Presentation Outline

Definitions

- **Network:** an **interconnected** group or system

Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks** RFC791 (September 1981)

Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks** RFC791 (September 1981)
- **IP:** Internet **Protocol** provides the functions necessary to deliver a package of bits from a source to a destination over a network

Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks** RFC791 (September 1981)
- **IP:** Internet **Protocol** provides the functions necessary to deliver a package of bits from a source to a destination over a network
- **(world wide) Web: network** consisting of a collection of Internet websites using HTTP

Definitions

- **HTTP:** Hypertext Transfer **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems draft HTTP2 (July 2014)

Definitions

- **HTTP:** Hypertext Transfer **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems draft HTTP2 (July 2014)
- **FTP:** File Transfer **Protocol** promotes sharing of files, encourages the use of remote computers RFC959 (October 1985)

Definitions

- **HTTP:** Hypertext Transfer **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems draft HTTP2 (July 2014)
- **FTP:** File Transfer **Protocol** promotes sharing of files, encourages the use of remote computers RFC959 (October 1985)
- **TCP:** Transmission Control **Protocol** is intended for use as a highly reliable host-to-host RFC761 (January 1980)

Definitions

- **HTTP:** Hypertext Transfer **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems draft HTTP2 (July 2014)
- **FTP:** File Transfer **Protocol** promotes sharing of files, encourages the use of remote computers RFC959 (October 1985)
- **TCP:** Transmission Control **Protocol** is intended for use as a highly reliable host-to-host RFC761 (January 1980)
- **UDP:** User Datagram **Protocol** provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism RFC768 (August 1980)

Definitions

- **HTTP:** Hypertext Transfer **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems draft HTTP2 (July 2014)
- **FTP:** File Transfer **Protocol** promotes sharing of files, encourages the use of remote computers RFC959 (October 1985)
- **TCP:** Transmission Control **Protocol** is intended for use as a highly reliable host-to-host RFC761 (January 1980)
- **UDP:** User Datagram **Protocol** provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism RFC768 (August 1980)
- **RFC:** Request For Comments (Internet Draft (ID), RFC, Internet Standard)

Definitions

- **Router:** network **hardware** providing routing services

Definitions

- **Router:** network **hardware** providing routing services
- **Routing: algorithm processed** to decide where to forward a packet

Definitions

- **Router:** network **hardware** providing routing services
- **Routing: algorithm processed** to decide where to forward a packet
- **Forwarding: action** of moving a packet from one NIC to another

Definitions

- **Router:** network **hardware** providing routing services
- **Routing: algorithm processed** to decide where to forward a packet
- **Forwarding: action** of moving a packet from one NIC to another
- **NIC:** Network Interface Card
- **Switch (hub):** network **hardware** connecting systems using packet switching

Definitions

- **Router:** network **hardware** providing routing services
- **Routing: algorithm processed** to decide where to forward a packet
- **Forwarding: action** of moving a packet from one NIC to another
- **NIC:** Network Interface Card
- **Switch (hub):** network **hardware** connecting systems using packet switching
- **Packet switching:** forward-like method regardless of the content (destination-based)

Definitions

- **Router:** network **hardware** providing routing services
- **Routing: algorithm processed** to decide where to forward a packet
- **Forwarding: action** of moving a packet from one NIC to another
- **NIC:** Network Interface Card
- **Switch (hub):** network **hardware** connecting systems using packet switching
- **Packet switching:** forward-like method regardless of the content (destination-based)
- **NAT:** Network Address Translation, router modifying IP address into another IP address.

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Response: application message** destined for a client (*result*)

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Response: application message** destined for a client (*result*)
- **Fat client: application** where most functions are processed by the client itself

Definitions

- **Node (network):** any entity that can send packets to/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Response: application message** destined for a client (*result*)
- **Fat client: application** where most functions are processed by the client itself
- **Thin client: application** where most functions are carried out on a central server

What kind of network is it?

- **BAN:** Body Area Network

What kind of network is it?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks

What kind of network is it?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)

What kind of network is it?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)
- **MAN:** Metropolitan Area Networks, can cover a whole city

What kind of network is it?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)
- **MAN:** Metropolitan Area Networks, can cover a whole city
- **WAN:** Wide Area Networks cover a broad area (Internet)

Topologies

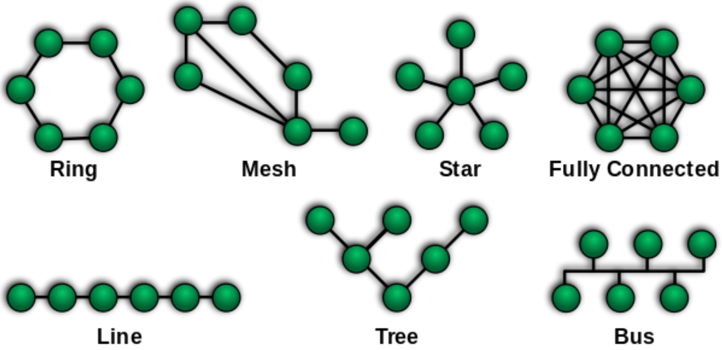


Figure: upload.wikimedia.org

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].
- **Star:** all messages go through the same central node, reducing network failure.

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.
- **Bus:** all nodes are connected to the same media. Only one can send a packet at a time, which all others then receive.

[1] Hong Kong protesters use a mesh network to organize

Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network[1].
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.
- **Bus:** all nodes are connected to the same media. Only one can send a packet at a time, which all others then receive.
- **Tree:** hierarchical topology, such as a binary tree.

[1] Hong Kong protesters use a mesh network to organize

Bonus

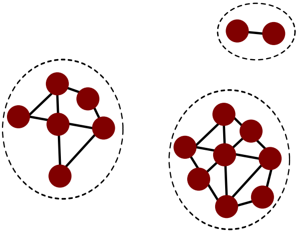


Figure: Disconnected MANET illustration [1]

Bonus

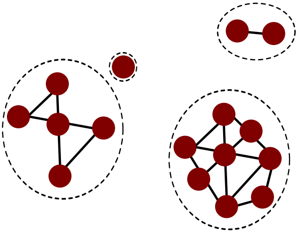


Figure: Store-carry-and-forward [1]

Bonus

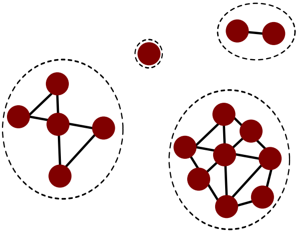


Figure: Store-carry-and-forward [1]

Bonus

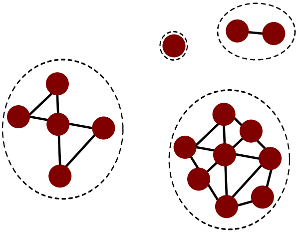


Figure: Store-carry-and-forward [1]

Bonus

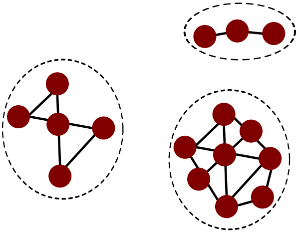


Figure: Store-carry-and-forward [1]

HTTP request/response example

Enter getbootstrap.com in your browser

HTTP request/response example

Enter getbootstrap.com in your browser

Source	Destination	Protocol	Length	Info
192.168.0.48	208.67.222.222	DNS	76	Standard query 0x4797 A getbootstrap.com
208.67.222.222	192.168.0.48	DNS	108	Standard query response 0x4797 A 192.30.252.154 A 192.30.252.153

Figure: DNS request/response

HTTP request/response example

Enter getbootstrap.com in your browser

Source	Destination	Protocol	Length	Info
192.168.0.48	208.67.222.222	DNS	76	Standard query 0x4797 A getbootstrap.com
208.67.222.222	192.168.0.48	DNS	108	Standard query response 0x4797 A 192.30.252.154 A 192.30.252.153

Figure: DNS request/response

Source	Destination	Protocol	Length	Info
127.0.0.1	127.0.0.13	TCP	74	36159 > http [SYN] Seq=0 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=12
127.0.0.13	127.0.0.1	TCP	74	http > 36159 [SYN, ACK] Seq=0 Ack=1 Win=43690 Len=0 MSS=65495 SACK_PERM
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=1 Ack=1 Win=43776 Len=0 TSval=122257 TSecr=12225
127.0.0.1	127.0.0.13	HTTP	356	GET /index.html HTTP/1.1
127.0.0.13	127.0.0.1	TCP	66	http > 36159 [ACK] Seq=1 Ack=291 Win=44800 Len=0 TSval=122259 TSecr=122
127.0.0.13	127.0.0.1	HTTP	354	HTTP/1.1 200 OK (text/html)
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=291 Ack=289 Win=44800 Len=0 TSval=122259 TSecr=1
127.0.0.1	127.0.0.13	HTTP	357	GET /favicon.ico HTTP/1.1
127.0.0.13	127.0.0.1	HTTP	565	HTTP/1.1 404 Not Found (text/html)
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=582 Ack=788 Win=45952 Len=0 TSval=122269 TSecr=1

Figure: HTTP request/response

How do messages reach their destination?

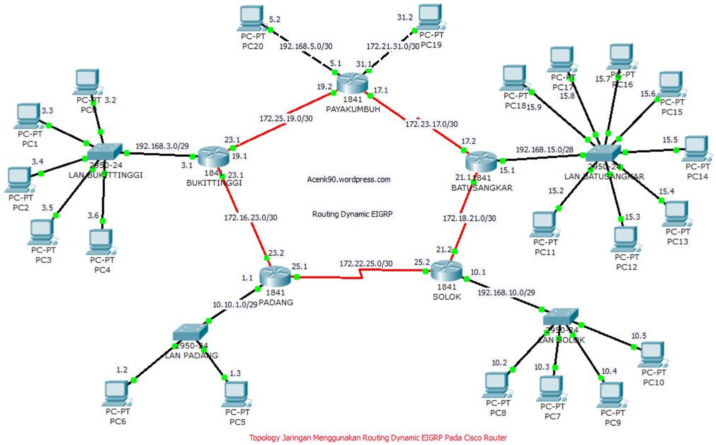


Figure: acenk90.files.wordpress.com

More like this...

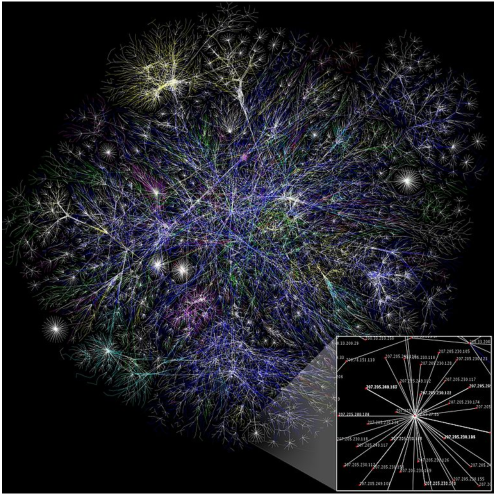


Figure: wikimedia.org

How does it work? From signal to application...

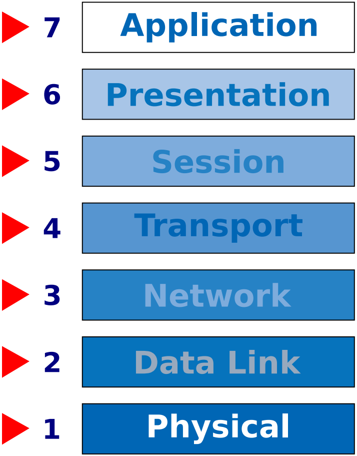
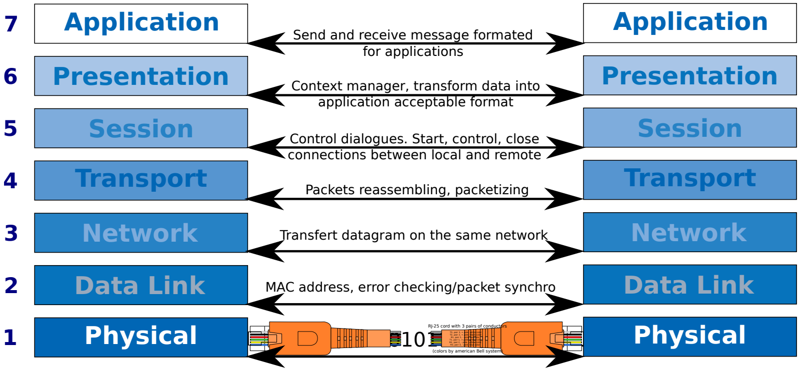
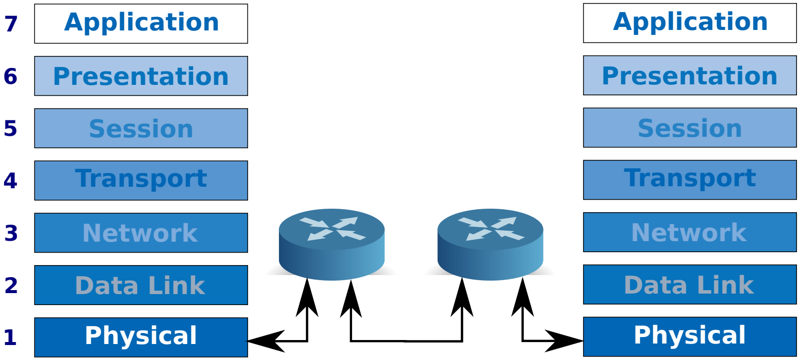


Figure: OSI model

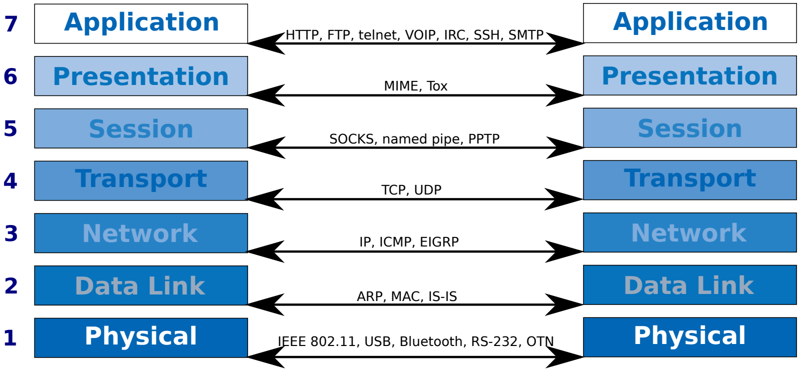
Nth layer communicate with Nth layer..



.. thanks to 3-th layers



One single protocol, one single layer



Encapsulation

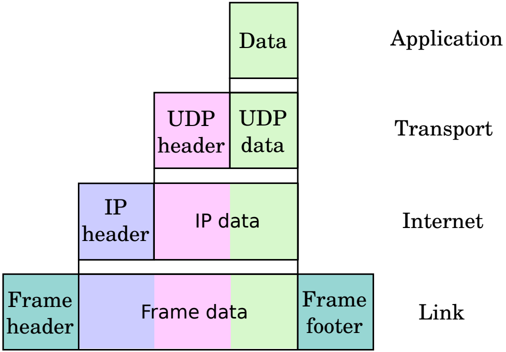


Figure: Encapsulation

Presentation Outline

Aims

- Interface data link layer,

Aims

- Interface data link layer,
- (De)Encode,

Aims

- Interface data link layer,
- (De)Encode,
- Transmit: 1 after 0 (after 0 or 1, after 0... or 1)

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): $<$100Gbit/s

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): $<$100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): $<$50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): <100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): <50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)
- IEEE 802.15.1 (a.k.a. Bluetooth): <1 Mbit/s

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): <100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): <50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)
- IEEE 802.15.1 (a.k.a. Bluetooth): <1 Mbit/s
- IEEE 802.15.4 (a.k.a. ZigBee): <250 kbit/s

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): $<$100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): $<$50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)
- IEEE 802.15.1 (a.k.a. Bluetooth): $<$1 Mbit/s
- IEEE 802.15.4 (a.k.a. ZigBee): $<$250 kbit/s
- IEEE 802.16 (a.k.a. Wi-Max): $<$40 Mbit/s

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): $<$100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): $<$50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)
- IEEE 802.15.1 (a.k.a. Bluetooth): $<$1 Mbit/s
- IEEE 802.15.4 (a.k.a. ZigBee): $<$250 kbit/s
- IEEE 802.16 (a.k.a. Wi-Max): $<$40 Mbit/s
- IEEE 1394 (a.k.a. Firewire): $<$3200 Mbit/s

Hardware medium

- IEEE 802.3 (a.k.a. Ethernet): <100Gbit/s
- IEEE 802.11 (a.k.a. Wi-Fi): <50 Mbit/s (802.11ad goes up to 6.75 Gbit/s)
- IEEE 802.15.1 (a.k.a. Bluetooth): <1 Mbit/s
- IEEE 802.15.4 (a.k.a. ZigBee): <250 kbit/s
- IEEE 802.16 (a.k.a. Wi-Max): <40 Mbit/s
- IEEE 1394 (a.k.a. Firewire): <3200 Mbit/s
- USB, serial port such as RS-232...

Hardware medium: IEEE 802.3 (Ethernet)

Figure: RJ45 connector

Hardware medium: IEEE 802.15.1 (Bluetooth)

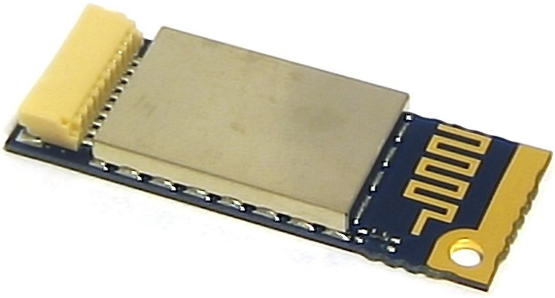


Figure: Bluetooth card

Hardware medium: IEEE 802.15.4 (ZigBee)

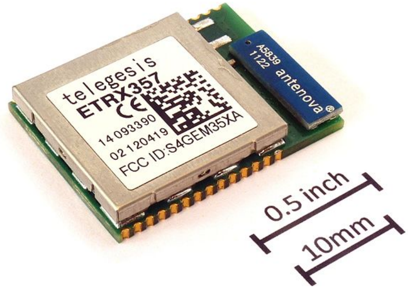


Figure: ZigBee card

Hardware medium: IEEE 802.16 (Wi-Max)

Figure: Wi-Max antenna

Hardware medium: IEEE 1394 (Firewire)

Figure: Firewire connector

Encoding

- **MLT3 (Multi-Level Transmit):** state change for 1s over 3 levels, stay in the same state for 0s

Encoding

- **MLT3 (Multi-Level Transmit):** state change for 1s over 3 levels, stay in the same state for 0s
- **AMI (Alternate Mark Inversion):** state 0 for 0s, state +/-1 for 1s

Encoding

- **MLT3 (Multi-Level Transmit):** state change for 1s over 3 levels, stay in the same state for 0s
- **AMI (Alternate Mark Inversion):** state 0 for 0s, state +/-1 for 1s
- **Manchester:** voltage transition (rising/falling edge mean 1/0)

Encoding

- **MLT3 (Multi-Level Transmit):** state change for 1s over 3 levels, stay in the same state for 0s
- **AMI (Alternate Mark Inversion):** state 0 for 0s, state +/-1 for 1s
- **Manchester:** voltage transition (rising/falling edge mean 1/0)
- **BMC (Biphase Mark Code):** change its state for 1s, stay on the same state for 0s

Encoding

- **MLT3 (Multi-Level Transmit):** state change for 1s over 3 levels, stay in the same state for 0s
- **AMI (Alternate Mark Inversion):** state 0 for 0s, state +/-1 for 1s
- **Manchester:** voltage transition (rising/falling edge mean 1/0)
- **BMC (Biphase Mark Code):** change its state for 1s, stay on the same state for 0s
- and so on...

Encoding: Multi-Level Transmit

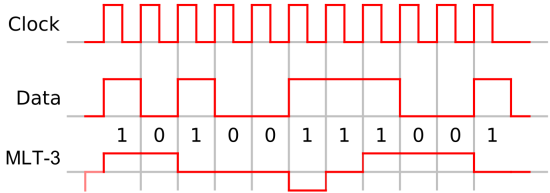


Figure: Multi-Level Transmit

Encoding: Alternate Mark Inversion

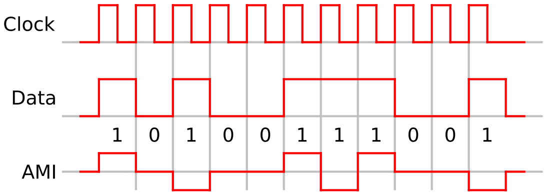


Figure: Alternate Mark Inversion

Encoding: Manchester

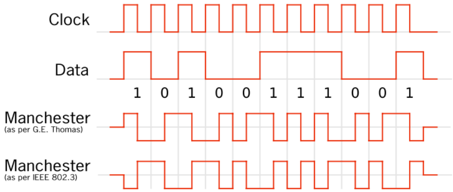


Figure: Manchester

Encoding: Biphase Mark Code

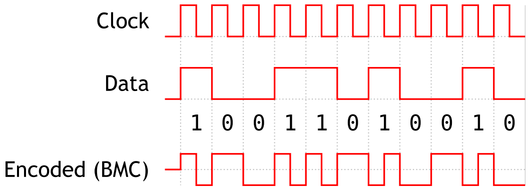


Figure: Biphase Mark Code

Transmitting

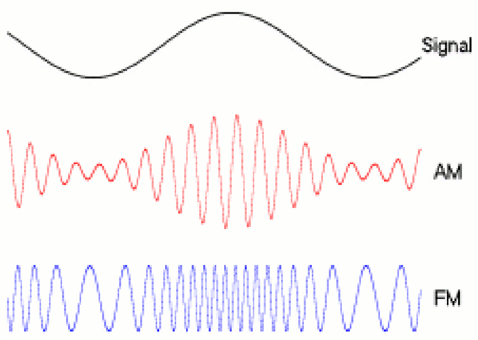


Figure: Amplitude and phase modulation

Error detection

- Repetition (hum...)

Error detection

- Repetition (hum...)
- Parity (XOR)

Error detection

- Repetition (hum...)
- Parity (XOR)
- Checksum

Error detection

- Repetition (hum...)
- Parity (XOR)
- Checksum
- CRC (Cyclic redundancy check): with a polynomial divison

Error detection

- Repetition (hum...)
- Parity (XOR)
- Checksum
- CRC (Cyclic redundancy check): with a polynomial divison
- Hash

Error detection

- Repetition (hum...)
- Parity (XOR)
- Checksum
- CRC (Cyclic redundancy check): with a polynomial divison
- Hash
- and so on...

Error correcting

- Repetition (again)

Error correcting

- Repetition (again)
- Hamming

Error correcting

- Repetition (again)
- Hamming
- MDPC (Multidimensional parity-check code)

Correction: MDPC

Raw data to send: 0x01 02 03 04

0x01	0x02	0x03
0x03	0x04	0x07
0x04	0x06	

Figure: Data received with MDPC

Data sent (with MDPC): 0x01 02 03 03 04 07 04 06

Aims

- Interface network layer,

Aims

- Interface network layer,
- Delivery to unique(?) hardware addresses,

Aims

- Interface network layer,
- Delivery to unique(?) hardware addresses,
- Framing,

Aims

- Interface network layer,
- Delivery to unique(?) hardware addresses,
- Framing,
- Data transfer

Layer composition (of its two sublayers)

1. Logical Link Control (LLC):
   - end to end flow control
   - end to end error control
   - (transmitting/receiving) protocols, over MAC sublayer, multiplexing

Layer composition (of its two sublayers)

1. Logical Link Control (LLC):
   - end to end flow control
   - end to end error control
   - (transmitting/receiving) protocols, over MAC sublayer, multiplexing
2. Media Access Control (MAC):
   - physical (hardware) addressing
   - collision detection and retransmission
   - data packet scheduling (and queuing)
   - QoS
   - VLAN

Carrier Sense Multiple Access with Collision Avoidance

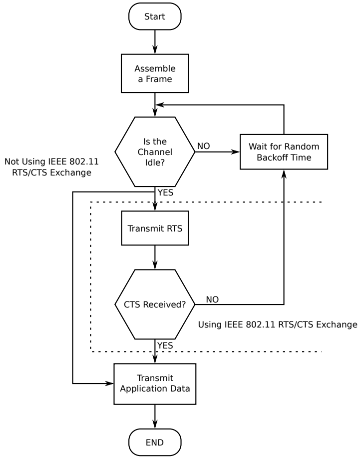


Figure: CSMA CA

Layer 2 Ethernet packet

MAC dest. (6)	MAC src. (6)	VLAN tag* (4)	Ethertype (2)
Payload (42-1500)		Frame check sequence (4)	

Figure: Layer 2 Ethernet packet

optional, Content (size in bytes)

Ethertype 0x	Protocol
0800	IPv4
0806	ARP
0842	Wake-on-LAN
86dd	IPv6

Figure: Data received with MDPC

ARP example

0000	ff	ff	ff	ff	ff	ff	fa	ba	00	ab	ab	af	08	06	00	01
0010	08	00	06	04	00	01	fa	ba	00	ab	ab	af	ac	11	22	37
0020	00	00	00	00	00	00	ac	11	00	f9	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00				

Figure: ARP request

MAC address destination MAC address source Ethertype Hardware type Protocol type OpCode (1 request, 2 reply) IP address source IP address destination

ARP example

0000	ff	ff	ff	ff	ff	ff	fa	ba	00	ab	ab	af	08	06	00	01
0010	08	00	06	04	00	01	fa	ba	00	ab	ab	af	ac	11	22	37
0020	00	00	00	00	00	00	ac	11	00	f9	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00				

Figure: ARP request

MAC address destination MAC address source Ethertype Hardware type Protocol type OpCode (1 request, 2 reply) IP address source IP address destination

ARP example

0000	fa	ba	00	ab	ab	af	be	be	00	00	eb	eb	08	06	00	01
0010	08	00	06	04	00	01	be	be	00	00	eb	eb	ac	11	00	f9
0020	fa	ba	00	ab	ab	af	ac	11	22	37	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00				

Figure: ARP reply

MAC address destination MAC address source Ethertype Hardware type Protocol type OpCode (1 request, 2 reply) IP address source IP address destination

ARP example

0000	fa	ba	00	ab	ab	af	be	be	00	00	eb	eb	08	06	00	01
0010	08	00	06	04	00	01	be	be	00	00	eb	eb	ac	11	00	f9
0020	fa	ba	00	ab	ab	af	ac	11	22	37	00	00	00	00	00	00
0030	00	00	00	00	00	00	00	00	00	00	00	00				

Figure: ARP reply

MAC address destination MAC address source Ethertype Hardware type Protocol type OpCode (1 request, 2 reply) IP address source IP address destination

Aims

- Interface transport layer,

Aims

- Interface transport layer,
- Host addressing,

Aims

- Interface transport layer,
- Host addressing,
- End-to-end packet transmission (data link? Connectionless? Switch? Router?),

Aims

- Interface transport layer,
- Host addressing,
- End-to-end packet transmission (data link? Connectionless? Switch? Router?),
- Routing, load balancing

Concepts

- IP addressing fundamentals,

Concepts

- IP addressing fundamentals,
- Classfull IP addressing,

Concepts

- IP addressing fundamentals,
- Classfull IP addressing,
- Subnet masks,

Concepts

- IP addressing fundamentals,
- Classfull IP addressing,
- Subnet masks,
- Variable length subnet masks (VLSM),

Concepts

- IP addressing fundamentals,
- Classfull IP addressing,
- Subnet masks,
- Variable length subnet masks (VLSM),
- Classless inter-domain routing (CIDR).

IP addressing fundamentals

IP address

32 bits (4x4 bytes)

mask	
Networks part	Host part

Figure: IP address parts

IP addressing fundamentals

Masks

- Separates network and host bits,

IP addressing fundamentals

Masks

- Separates network and host bits,
- MSB **always** are ones and then zeros! 255.254.255.0 is not possible,

IP addressing fundamentals

Masks

- Separates network and host bits,
- MSB **always** are ones and then zeros! 255.254.255.0 is not possible,
- Indicates how many bits are used for the network part:
  - A 8-bit mask leaves 24 bits for the hosts,
  - A 16-bit mask leaves 16 bits for the hosts,
  - A 24-bit mask leaves 8 bits for the hosts,
  - A N-bit mask leaves 32-N bits for the hosts.

IP addressing fundamentals

Masks

- Separates network and host bits,
- MSB **always** are ones and then zeros! 255.254.255.0 is not possible,
- Indicates how many bits are used for the network part:
  - A 8-bit mask leaves 24 bits for the hosts,
  - A 16-bit mask leaves 16 bits for the hosts,
  - A 24-bit mask leaves 8 bits for the hosts,
  - A N-bit mask leaves 32-N bits for the hosts.
- Two different mask (differences seen further):
  - Network mask,
  - Subnet mask.

IP addressing fundamentals

IP address

32 bits (4x4 bytes)

Networks part	Host part

Figure: IP address parts and mask

IP addressing fundamentals

IP address

32 bits (4x4 bytes)

ones mask	zeros mask
Networks part	Host part

Figure: IP address parts and mask

IP addressing fundamentals

Is that a host?

- Network address,

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,
- Broadcast address.

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,
- Broadcast address.

Within the same network

- All addresses have the same network bits,

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,
- Broadcast address.

Within the same network

- All addresses have the same network bits,
- All nodes have different host bits,

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,
- Broadcast address.

Within the same network

- All addresses have the same network bits,
- All nodes have different host bits,
- Network address has zeros for host bits,

IP addressing fundamentals

Is that a host?

- Network address,
- Nodes,
- Broadcast address.

Within the same network

- All addresses have the same network bits,
- All nodes have different host bits,
- Network address has zeros for host bits,
- Broadcast address has ones for host bits.

Example: network 1

Mask /24	255 11111111	255 11111111	255 11111111	0 00000000
Network address	192 11000000	168 10101000	1 00000001	0 00000000
First nodes address	192 11000000	168 10101000	1 00000001	1 00000001
Last nodes address	192 11000000	168 10101000	1 00000001	254 11111110
Broadcast address	192 11000000	168 10101000	1 00000001	255 11111111

Figure: IP address example 1

Example: network 2

Mask /16	255 11111111	255 11111111	0 00000000	0 00000000
Network address	172 10101100	17 00010001	0 00000000	0 00000000
First nodes address	172 10101100	17 00010001	0 00000000	1 00000001
Last nodes address	172 10101100	17 00010001	255 11111111	254 11111110
Broadcast address	172 10101100	17 00010001	255 11111111	255 11111111

Figure: IP address example 2

Formula

How many ~~hosts~~ nodes with a N-bit mask?

$2^{32-N} - 2$

Formula

How many ~~hosts~~ nodes with a N-bit mask?

$2^{32-N} - 2$, the -2 moves out network and broadcast addresses which are not nodes.

Formula

How many ~~hosts~~ nodes with a N-bit mask?

$2^{32-N} - 2$, the -2 moves out network and broadcast addresses which are not nodes.

- 24-bit mask: $2^{32-24} - 2 = 2^8 - 2 = 254$ nodes

Formula

How many ~~hosts~~ nodes with a N-bit mask?

$2^{32-N} - 2$, the -2 moves out network and broadcast addresses which are not nodes.

- 24-bit mask: $2^{32-24} - 2 = 2^8 - 2 = 254$ nodes
- 16-bit mask: $2^{32-16} - 2 = 2^{16} - 2 = 65.534$ nodes

Formula

How many ~~hosts~~ nodes with a N-bit mask?

$2^{32-N} - 2$, the -2 moves out network and broadcast addresses which are not nodes.

- 24-bit mask: $2^{32-24} - 2 = 2^8 - 2 = 254$ nodes
- 16-bit mask: $2^{32-16} - 2 = 2^{16} - 2 = 65.534$ nodes
- 8-bit mask: $2^{32-8} - 2 = 2^{24} - 2 = 16.777.214$ nodes

Public and private addresses

Public

- Most of IP addresses

[2] Internet Assigned Numbers Authority

Public and private addresses

Public

- Most of IP addresses
- Registered ISP and large organizations inherit blocks of public addresses from IANA[2]

[2] Internet Assigned Numbers Authority

Public and private addresses

Public

- Most of IP addresses
- Registered ISP and large organizations inherit blocks of public addresses from IANA[2]
- Usage of not registered public addresses is forbidden.

Private

- Privates addresses are A, B and C classes (see after)

[2] Internet Assigned Numbers Authority

Public and private addresses

Public

- Most of IP addresses
- Registered ISP and large organizations inherit blocks of public addresses from IANA[2]
- Usage of not registered public addresses is forbidden.

Private

- Privates addresses are A, B and C classes (see after)
- No registration needed

[2] Internet Assigned Numbers Authority

Public and private addresses

Public

- Most of IP addresses
- Registered ISP and large organizations inherit blocks of public addresses from IANA[2]
- Usage of not registered public addresses is forbidden.

Private

- Privates addresses are A, B and C classes (see after)
- No registration needed
- Not routed across the Internet

[2] Internet Assigned Numbers Authority

Public and private addresses

Public

- Most of IP addresses
- Registered ISP and large organizations inherit blocks of public addresses from IANA[2]
- Usage of not registered public addresses is forbidden.

Private

- Privates addresses are A, B and C classes (see after)
- No registration needed
- Not routed across the Internet
- Proxy, NAT and private addresses solved IPv4 shortage.

[2] Internet Assigned Numbers Authority

Classful IP Addressing

Class	A	B	C
First octet	1 - 126	128 - 191	192 - 223
First octet pattern 0b	0*	10*	110*
Network mask	255.0.0.0 /8	255.255.0.0 /16	255.255.255.0 /24
IP addresses range	1.0.0.0 126.0.0.0	128.0.0.0 191.255.0.0	192.0.0.0 223.255.255.0
Number of nodes	16777214	65534	254

Figure: Three main classes

Where did 127.0.0.0/8 go ?!

Classful IP Addressing

Class D

- First octet: 224 - 239

Classful IP Addressing

Class D

- First octet: 224 - 239
- First octet pattern: 1110*

Classful IP Addressing

Class D

- First octet: 224 - 239
- First octet pattern: 1110*
- Theses IP addresses are multicast addresses.

Classful IP Addressing

Class D

- First octet: 224 - 239
- First octet pattern: 1110*
- Theses IP addresses are multicast addresses.

Class E

- Everything left

Classful IP Addressing

Class D

- First octet: 224 - 239
- First octet pattern: 1110*
- Theses IP addresses are multicast addresses.

Class E

- Everything left
- Experimental class.

Reserved addresses

- 0.0.0.0 used in routing (seen further)

Reserved addresses

- 0.0.0.0 used in routing (seen further)
- 127.0.0.0/8: loopback addresses (127.0.0.1 - 127.255.255.254).

Course details

Course details

Course details

References

Maurice J. Khabbaz, Assi Chadi M., and Fawaz Wissam F.
Disruption-Tolerant Networking: A Comprehensive Survey on Recent Developments and Persisting Challenges.
IEEE communications surveys and tutorials, 2012.