Chapter 5

Link Layer

How are packets exchanged between two neighboring nodes?

5.1 Addressing

Definition 5.1 (Link Layer)**.** *The **link layer** deals with the transmission of packets between two neighboring nodes, i.e., **single-hop**.*

Remarks:

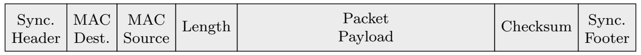
- The link layer is the bottom layer of the internet protocol suite.
- The concepts of the link layer can be grouped into two parts: the *medium access control* (MAC) dictates access patterns to the underlying wired or wireless medium. The *physical layer* (PHY) specifies the encoding of the data stream on the medium. Some layering models such as the Open Systems Interconnection (OSI) model treat these two parts as separate layers, and some models are even more detailed and split up both parts into multiple layers each.

Definition 5.2 (MAC Addresses)**.** *To identify nodes below the network layer, **MAC addresses** are used. A MAC address consists of 6 bytes and is typically formatted as 6 2-digit hexadecimal numbers separated by hyphens or colons, e.g.,* `00:21:cc:63:e8:5f`*.*

Remarks:

- Every network interface device is assigned a worldwide unique address by the manufacturer. However, many devices also support overriding this address through software.
- On the link layer only MAC addresses are valid as source and destination addresses for packets – IP addresses are a concept introduced above the link layer and can hence not be used on the link layer.

Definition 5.3 (Link Layer Packets)**.** ***Link layer packets**, also called **frames**, have additional fields to mark the exact frame in time they occupy during transmission: a synchronization header and in some protocols also a synchronization footer are added, containing predefined bit sequences any listener can recognize as the start (or the end) of a packet.*



Sync. Header	MAC Dest.	MAC Source	Length	Packet Payload	Checksum	Sync. Footer

Figure 5.4: Typical link layer packet.

Remarks:

- When an IP packet is transmitted, it makes up the payload of a link layer packet.
- Some protocols operate directly on the link layer, i.e., they do not send IP packets and address nodes directly by their MAC addresses. We will see an example of such a protocol below (Protocol 5.7).
- There are also devices operating strictly below the network layer. The most prominent among them is the *switch*.

Definition 5.5 (Switches)**.** *A **switch** is a central network node with the task of mediating traffic between its neighbors. Unlike routers, switches are unaware of IP addresses and operate on the link layer only.*

Remarks:

- Without the need for routing, subnets or port forwarding tables, switch hardware can be a simpler and cheaper alternative to routers for connecting a set of nodes locally.
- In its most basic form, a switch simply copies any incoming packet to all other connected neighbors without any inspection or modification of the packet. A switch does not have a MAC address.
- However, typically, a switch also keeps track of what source MAC addresses were received on each of its physical ports. If a packet arrives with a known destination MAC address, the switch can forward the packet to that port only.
- A port of a switch does not necessarily have to be connected to an IP aware node – it is also possible to chain switches. This means, a switch might need to internally assign several MAC addresses to a single port.
- In the wireless setting, as every packet is broadcasted by the very nature of the medium, the concept of switches is superfluous. However, devices called *repeaters* may extend the reach of a wireless network by rebroadcasting any received packets. As repeaters are not even aware of MAC addresses, they only operate on the physical layer (PHY).

Definition 5.6 (Broadcast MAC Address)**.** *The MAC address* `ff:ff:ff:ff:ff:ff` *is the designated **broadcast address** on the link layer. When used as a packet's destination address, any node hearing the packet will process it.*

Protocol 5.7 (ARP)**.** *The **Address Resolution Protocol** is used to find out the MAC address belonging to a given IPv4 address.*

Algorithm 5.8 ARP lookup for an IP address a

```
Send a query containing $a$ to ff:ff:ff:ff:ff:ff (broadcast)
if there is a node with IP address $a$ then
    That node responds with its MAC address
else
    After some timeout, conclude that $a$ does not exist or is not reachable
end if
```

Protocol 5.9 (NDP)**.** *The **Neighbor Discovery Protocol** offers the functionality of ARP for IPv6. It also includes additional features, such as the detection of duplicate addresses.*

Remarks:

- Address resolution is the main way to obtain destination MAC addresses – the routing done on the network layer merely outputs IP addresses.
- Caching is used, hence ARP/NDP look-ups are only made for new IP addresses.
- ARP/NDP packets are not IP packets – they are their own kind of packet.
- Nowadays often all traffic of nodes at the "edge" of the network (such as personal computers and smartphones) is routed over a *gateway* router. As the gateway is the only direct neighbor of edge nodes, they never contact any MAC address apart from the gateway. However, when connecting nodes through a switch, e.g., at LAN parties, ARP/NDP are vital to make newly plugged in nodes reachable.

Definition 5.10 (Global Broadcast Address, IPv4)**.** *The IP address `255.255.255.255` is the designated **global broadcast address** on the network layer for IPv4. When used as a packet's destination address, any node hearing the packet will process it.*

Remarks:

- A router receiving a packet with a broadcast destination will echo the packet to all connected devices.
- Certain routers will drop broadcast packets, for example, routers belonging to an ISP – broadcasting a packet to everybody on the Internet is not a reasonable operation.

Protocol 5.11 (DHCP)**.** *The **Dynamic Host Configuration Protocol** is used to automatically assign unused IP addresses to newly connecting network participants. To do so, one node in the network runs a designated DHCP server.*

Algorithm 5.12 Acquiring an unused IP address using DHCP

```
DHCP client sends a request with its MAC address to 255.255.255.255,
    using source address 0.0.0.0
DHCP server decides on an unused IP address $a$ and marks $a$ as "reserved"
    and replies with the offer for $a$
DHCP client notes the IP address of the DHCP server and replies, this time
    directly, that it accepts $a$
DHCP server marks $a$ as in use and replies with a final confirmation
DHCP client receives the confirmation and uses $a$ as its IP from then on
```

Remarks:

- DHCP is strictly speaking an application layer protocol as it builds upon UDP.
- As broadcast IP addresses cannot resolve to MAC addresses, the broadcast MAC address `ff:ff:ff:ff:ff:ff` is used.
- The DHCP server may base its choice of the offered IP address on the joining device's MAC address, and assign a returning device its previous IP address.
- In addition to unused IP addresses, the DHCP server often also distributes other configuration data such as its subnet mask (the block of local addresses, e.g., 192.168.0.0/24), the gateway node's address and the preferred DNS server's address.
- If no DHCP server is present, unique IP addresses as well as the network configuration have to be set manually for every participant.
- There are two separate versions of DHCP, for IPv4 addresses and IPv6 addresses respectively, fulfilling the same purpose. IPv6 defines several specialized broadcast addresses; for DHCP the address `ff02::1:2` is used.

5.2 Wireless Phenomena

Let us briefly discuss some of the differences between wired and wireless communication.

As radio transmissions are electromagnetic waves, their propagation is reminiscent of other waves we experience in everyday life such as light and sound. For example, phenomena such as shadowing, reflection and even diffraction are observable in radio waves.

Definition 5.13 (Half-Duplex, Full-Duplex)**.** ***Half-duplex** devices are **not** able to both send and receive at the same time. **Full-duplex** devices can send and receive simultaneously.*

Remarks:

- Wireless devices are typically *half-duplex*, which means they cannot receive anything while sending.
- In contrast, wired communication is usually *full-duplex*, i.e., both ends of the cable may send and receive at the same time.
- The most prominent special property of the wireless medium is that by nature any transmission is a broadcast, i.e., any wireless receiver physically within range will receive a sent message, not just the intended recipient. "Within range" means that the signal is sufficiently stronger than the ambient electromagnetic noise as well as interfering signals. This can be modeled by the *signal-to-interference-plus-noise ratio*.

Definition 5.14 (SINR)**.** *The* ***signal-to-interference-plus-noise ratio*** *(SINR) is a model for the quality of a received signal. It is defined as:*

$$SINR = \frac{S}{I+N} \overset{!}{>} \beta$$

- *S: the strength of the signal to be received*
- *I: the sum of the interference caused by other transmissions*
- *N: the ambient noise*
- *β: the* ***SINR threshold*** *which needs to be cleared for successful signal reception.*

Remarks:

- This formula may be evaluated at each receiver separately to determine whether it can correctly decode the signal.
- The SINR threshold β depends on hardware and encoding.
- Physics dictate that in vacuum an electromagnetic signal's strength diminishes quadratically with distance traveled. When permeating other materials such as air or concrete walls the signal is weakened even more quickly. This effect is called *fading*.
- There exist detailed models predicting the effect of not only fading but also wave propagation phenomena such as shadowing and reflection, but these are beyond the scope of this lecture.
- Due to the different travel times of the signal over different paths, the received signal may be the sum of several components delayed by different amounts. This is called *multipath*. For example, the received signal may consist of the direct line-of-sight component of the sent signal plus a component with a longer travel time reflected off a wall.
- By using nodes with multiple antennas, multipath can be exploited to transmit and decode multiple signals at once, increasing throughput. Such schemes are the foundation of the field of *MIMO transmissions* (multiple-input, multiple-output).

- In general, it is desirable to use lower transmit powers when possible, as this reduces power consumption as well as interference caused to other nearby wireless links. However, standards designed for throughput, such as Wireless LAN, often rather prefer to use the highest available transmit power to maximize the achieved SINR, as this allows employing more efficient encodings.

5.3 Medium Access Control (MAC)

Definition 5.15 (Multiple Access)**.** ***Multiple access*** *describes a setting in which multiple devices use a shared medium to communicate. It also describes the problem of avoiding deterioration of service caused by the collisions of transmissions in such a setting.*

Remarks:

- Collision mainly concerns wireless networks nowadays. In the past, sometimes bus network structures were used, i.e., every node was connected to the same bus cable, exhibiting similar problems for wired networks.

Definition 5.16 (Medium Division)**.** *By subdividing the medium into separate domains, in each of which only one device may send at a time, collisions can be prevented from occurring in the first place. Such subdivisions may be done in several ways:*

- ***Space division:*** *segment the area of operation such that fading prevents any two potential senders' signals from colliding. Examples: AM/FM radio, GSM.*
- ***Frequency division:*** *segment the available frequency spectrum into multiple frequencies bands that can be used in parallel. However, note that usually a device cannot listen on multiple frequencies simultaneously. Examples: Wireless LAN, Bluetooth.*
- ***Time division:*** *segment time into time slots, in each of which only one device may send as designated by some kind of schedule. Examples: Bluetooth, GSM.*
- ***Code division:*** *stretch the signal and xor it with a pseudorandom bit sequence unique to each sender. Knowing the pseudorandom bit sequences, a receiver can then distinguish simultaneously arriving superimposed signals. Examples: GPS, UMTS/3G.*

Remarks:

- Typically, multiple kinds of division are combined to reach a desired level of sender separation. Bluetooth, for example, makes heavy use of time and frequency division through the use of its strict scheduling and frequency hopping.

Definition 5.17 (Carrier Sensing)**.** ***Carrier sensing*** *or* ***clear channel assessment*** *(CCA) is a technique to prevent collisions from occurring by listening to the medium (the "carrier") for a short while before sending, such that one might pick up on an already ongoing transmission.*

Remarks:

- If no other transmission is detected, sending is performed immediately. If another transmission is detected, sending is postponed as it is assumed a collision would occur wiping out both packets. Before the next sending attempt carrier sensing is performed again.

Definition 5.18 (Hidden Terminal Problem)**.** *Due to the fading nature of the wireless medium one may not always hear the other senders during carrier sensing, even though at the intended recipient the signals of the senders would collide. This is referred to as the* ***hidden terminal problem****.*

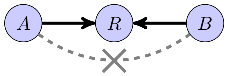


Figure 5.19: The hidden terminal problem: senders A and B can reach the recipient R, but they cannot hear each other. Hence carrier sensing cannot avoid collisions.

Definition 5.20 (Exposed Terminal Problem)**.** *The* ***exposed terminal problem*** *is the opposite of the hidden terminal problem: two close senders trying to send to different recipients may sense each others' signals and avoid sending simultaneously even though each receiver would be able to receive its signal perfectly well.*

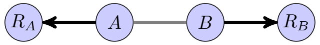


Figure 5.21: The exposed terminal problem: senders A and B could send to their respective recipients R_A and R_B simultaneously, but believe it would cause collisions due to carrier sensing.

Protocol 5.22 (RTS/CTS)**.** ***Request To Send / Clear To Send*** *is a packet exchange proposed as a solution to the hidden and exposed terminal problems.*

Remarks:

- RTS/CTS solves the hidden terminal problem as a receiving node's CTS will allow exactly one of its neighbors to send.
- The exposed terminal problem is also solved as long as the CTS messages do not interfere with other ongoing transmissions. For instance, assuming the setup from Figure 5.21, if B was already transmitting, A may not be able to hear a CTS message from R_A.

Algorithm 5.23 RTS/CTS

```
Before sending, the sender sends out a short RTS packet
If the intended recipient hears the RTS packet, it answers with a short CTS packet
If the sender receives the CTS packet, it begins transmission, otherwise it assumes it is not clear to send and tries again later
Other nodes hearing the CTS abstain from sending for some time since they know one of their neighbors is about to receive a packet from somewhere else
```

Definition 5.24 (Collision Response)**.** *The counterpart to collision avoidance is the approach of detecting collisions and responding to them after the fact.*

Remarks:

- Collisions are usually detected by immediately following up every successfully received packet with an acknowledgment (ACK) packet back to the sender. If the sender does not receive the ACK it will assume its packet got lost and try again.
- Collisions can also be detected as they occur, if the devices support simultaneous sending and receiving (i.e., are *full-duplex*) and the medium guarantees for multiple senders to hear each other (common for wired bus networks, but usually does not apply to wireless networks).
- Even though carrier sensing may prevent collisions from actually destroying packets and thus reducing the network throughput, the response is usually similar to reacting to a collision after it occurred: wait for some amount of time and then retry.
- Making a good choice for the amount of time to postpone the sending is not trivial.

Definition 5.25 (Backoff Time, Backoff Strategy)**.** *The time waited before retrying an unsuccessful transmission is called the* ***backoff time****. Ways to choose a backoff time are called* ***backoff strategies****.*

Remarks:

- Using a fixed duration as backoff time is not advisable: If two conflicting senders employ the same backoff strategy, their sending attempts would keep conflicting. Thus, feasible backoff strategies require a random component.
- Thought experiment: n nodes all try to send at the same time towards a single receiver. All transmissions start at the start of a time slot and have exactly the length of the time slot. How would a strategy maximize the probability of exactly one node sending at a time?

Protocol 5.26 (Slotted Aloha)**.** *In every time slot, every node transmits with probability* $1/n$*.*

Theorem 5.27. *Using Protocol 5.26 allows one node to transmit alone after expected time e.*

Proof. The probability for success, i.e., that the number of transmitting nodes X is exactly 1, is

$$\Pr[X = 1] = n \cdot \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \approx \frac{1}{e},$$

where the last approximation is a result from Theorem 5.28 for sufficiently large n. Hence, if we repeat this process e times, we can expect one success. □

Theorem 5.28. *We have*

$$e^t \left(1 - \frac{t^2}{n}\right) \leq \left(1 + \frac{t}{n}\right)^n \leq e^t$$

for all $n \in \mathbb{N}, |t| \leq n$. Note that

$$\lim_{n \to \infty} \left(1 + \frac{t}{n}\right)^n = e^t.$$

Remarks:

- The origin of the name is the ALOHAnet protocol which was developed at the University of Hawaii to wirelessly connect the islands.

- Protocol 5.26 also works in an unslotted time model, with a factor 2 penalty, i.e., the probability for a successful transmission will drop from $\frac{1}{e}$ to $\frac{1}{2e}$. Essentially, each slot is divided into t small time slots with $t \to \infty$ and the nodes start a new t-slot long transmission with probability $\frac{1}{2nt}$.

- Protocol 5.26 requires knowledge of the number of senders n. What if we don't know n?

Algorithm 5.29 Random exponential backoff

```
$i \leftarrow 0$
Attempt sending
while sending unsuccessful do
    $i \leftarrow i + 1$
    Pick a value from the interval $[0, c^i]$ uniformly at random and wait that
    many time units
    Attempt sending again
end while
```

Remarks:

- c is some constant, often 2.

- We are going to see backoff protocols also in different contexts, e.g. in Chapter 10 as a way to deal with lock contention.

- Growing the range of values $[0, c^i]$ after every failed transmission attempt allows the system to adapt dynamically to the number of senders, as each sender spreads out its transmissions more when the number of senders is large, but still does not waste too much time when the number of senders is small.

- Both Aloha and random exponential backoff waste slots, in which more or fewer than one sender send. If it is possible to coordinate a schedule implicitly or explicitly, the frequency of successful transmissions can be improved significantly.

Definition 5.30 (Duty Cycling)**.** *Nodes in a network may agree on periods of time in which no messages are exchanged. During these periods the nodes may remain in a low-power sleep mode to conserve energy. This called **duty cycling**.*

Remarks:

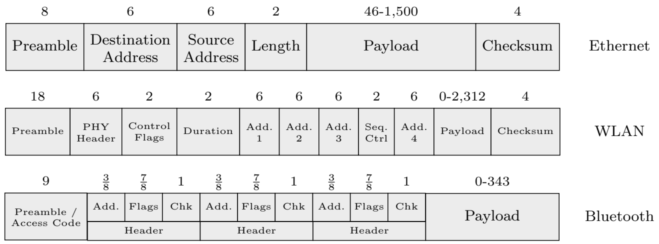
- Duty cycling is especially interesting to mobile devices without a constant power supply. As wireless devices consume a significant amount of energy both when transmitting and when only listening, shutting down the wireless hardware when it is not needed has become a priority.

- As wireless communication requires both the sender and the receiver to be awake at the same time to be successful, such shutting down needs to be carefully coordinated as not to carelessly lose packets.

- In networks coordinated by a central access point, the most straightforward way is to have the access point synchronize all participants and declare some wake-up schedule. Whenever a scheduled wake-up is reached, nodes power on to exchange messages. As soon as a node knows it won't need to participate in any more traffic until the next wake-up it can go to sleep.

5.4 Physical Layer (PHY)

Once we decide when to send, when not to send, and whom to address, all that remains of the link layer are several physical considerations. The link layer packet format used for almost all the wired connections of the Internet today is Ethernet v2. Wireless protocols such as Wireless LAN (following the IEEE 802.11 standard) and Bluetooth define their own packet formats. Figure 5.31 shows some examples of some common packet types.



8	6	6	2	46-1,500	4	
Preamble	Destination Address	Source Address	Length	Payload	Checksum	Ethernet

18	6	2	2	6	6	6	2	6	0-2,312	4	
Preamble	PHY Header	Control Flags	Duration	Add. 1	Add. 2	Add. 3	Seq. Ctrl	Add. 4	Payload	Checksum	WLAN

9	$\frac{3}{8}$	$\frac{7}{8}$	1	$\frac{3}{8}$	$\frac{7}{8}$	1	$\frac{3}{8}$	$\frac{7}{8}$	1	0-343	
Preamble / Access Code	Add.	Flags	Chk	Add.	Flags	Chk	Add.	Flags	Chk	Payload	Bluetooth
	Header			Header			Header				

Figure 5.31: The complete physical representations of typical Ethernet, Wireless LAN and Bluetooth packets. The number above each field corresponds to its length in bytes.

Remarks:

- Wireless LAN uses several address fields, to express packets being forwarded by a base station or a repeater.
- The Bluetooth header is repeated three times and only contains one 3-bit address, as more is not required in Bluetooth's network topologies of at most 8 nodes, in which all packets are either sent or received by the network's master node.
- All these formats have preambles and checksums in common. These two features we will discuss in the remainder of this section.

Definition 5.32 (Syncword/Preamble)**.** *To establish clearly detectable packet boundaries, i.e., when exactly a packet begins and when it ends, physical layer implementations typically specify a fixed sequence of bits or bytes to be transmitted at the start of every packet called* ***syncword*** *or* ***preamble****.*

Remarks:

- To specify the end of the packet, either the length of the packet is encoded in the packet's header or another packet end sequence (a "footer") is attached to the end of the packet.
- It may be of interest to make the syncword unique, i.e., not let it appear as part of packet's body. This way, participants freshly joining or just waking up can be certain a new packet started when they hear the syncword.

Definition 5.33 (Bit/Byte Stuffing)**.** ***Bit stuffing*** *and* ***byte stuffing*** *are techniques representing bit resp. byte sequences, which should be unique to the packet boundaries (such as syncwords), within a packet's body in a way such that these sequences do not occur in the packet's body.*

Remarks:

- Naturally, whatever technique is employed should be reversible: the receiver should be able to restore the original content of the packet's body.
- For simplicity's sake, for the remainder of this section consider the case of a critical byte sequence consisting of only some byte `X`. The results generalize to arbitrary critical bit and byte sequences.
- First, consider the naive approach: We cannot simply replace every occurrence of `X` in the body with another byte `Y`, as we would not be able to distinguish these `Y`s from bytes which were originally `Y`.

Definition 5.34 (Escape Sequences)**.** *Given some critical byte `X`, we choose a byte $Y \neq X$ as* ***escape byte*** *and use it to define two* ***escape sequences*** *consisting of two bytes each, say, `YA` and `YB` ($A \neq X$, $B \neq X$, $A \neq B$). The sender replaces every `Y` in the original body with `YA` and every `X` with `YB`. The receiver in turn performs the substitution in reverse.*

Remarks:

- This scheme is correct: the encoded body does not contain any `X`, and decoding will always yield the original body.
- The sequence $\mathtt{Y}z$ in the encoded body is undefined for values of $z \notin \{\mathtt{A}, \mathtt{B}\}$.
- The general concept of escape sequences is also frequently used in software. For example, to encode a quotation mark we use a backslash as escape character, e.g., `"Herman \"Babe\" Ruth"`. In web addresses `%` is used to escape the bytes of "illegal" characters, e.g., `%20` for spaces and `%E2%98%83` for the unicode snowman.
- The main disadvantage of this simple scheme is that it may cause the packet's body to become a lot longer than it originally was – up to twice as long!

Definition 5.35 (Consistent Overhead Byte Stuffing)**.** *Treat the original body as a sequence of byte strings $s_0, s_1, \ldots, s_n$ separated by the forbidden `X` byte, then alternatingly send the length of a string and the string itself: $|s_0|, s_0, |s_1|, s_1, \ldots, |s_n|, s_n$. The receiver can then reconstruct the original body by joining the strings back together with `X`s in between.*

Remarks:

- If there are multiple subsequent `X` in the original body, s_i may be an empty string.
- If we assume that the strings are short, then encoded body is always exactly 1 byte longer than the original, no matter how often `X` occurs.
- We need to avoid using `X` in a length value. This can be accomplished adding 1 to all length values $\geq$ `X`.

Definition 5.36 (Checksums)**.** *Another common feature is the inclusion of a checksum over the whole packet, including header and payload.*

Remarks:

- Computing a checksum typically entails xoring together all input bits several times following certain patterns to obtain a checksum of 1-4 bytes.
- Checksum algorithms usually require only a single pass over the data and are simple to enough to allow performing computation and checking of these checksums in hardware.
- Checksums on the link layer serve multiple purposes. For one, it would be a shame if a unreliable wireless link corrupted a packet which traveled across the globe – resending it from its source node would be a waste. Further, link layers are interested in also having checksums for non-data packets (such as those for connecting and disconnecting, synchronizing schedules or RTS/CTS).
- Higher layers in the network stack may employ additional checksums, such that they may be used on unreliable link layers.
- IPv6, as opposed to IPv4, no longer includes a checksum and expects the underlying link layer to employ reliable error detection.
- There also exist *error correcting codes* which allow not only detecting but also correcting a certain amount of bit errors. In practice, they are used only in certain Wireless LAN versions; usually, it is assumed that most packets are transmitted either completely without errors or damaged beyond repair.

Definition 5.37 (MTU)**.** *Every link layer implementation specifies a* ***maximum transmission unit****, the maximum link layer payload size this link layer supports.*

Remarks:

- For Ethernet this value is 1,500 bytes, for Wireless LAN it usually is 2,312 bytes, and for Bluetooth it usually is 672 bytes, using a higher transmission rate for the payload.
- It is the network layer's responsibility to ensure it creates no packets larger than the MTU.
- IPv4 and IPv6 support fragmenting oversized transport layer packets into several network layer packets, using fields in the IP header to indicate the number of fragments (Definition 2.22).
- Since Ethernet and Wireless LAN packets are common, an MTU of 1,500 bytes has become commonplace in many applications and frameworks.
- So how do we actually transmit a packet?

Definition 5.38 (Line Coding)**.** ***Line coding*** *is a physical encoding representing a data bit stream as a series of values from $\{-1, 0, +1\}$. When transmitting, each value is to be held for 1 time unit on the line before moving on to the next value in the series.*

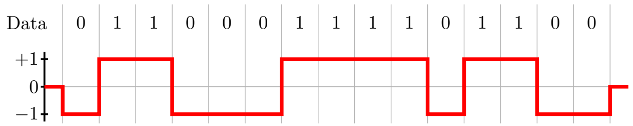


Figure 5.39: Simple line coding.

Remarks:

- Figure 5.39 shows the simplest kind of line coding: mapping every '0' bit to -1 and every '1' bit to $+1$.
- Such simple codings have two disadvantages when there is a long string of equal bits: 1) It is hard to verify that a signal is still being sent during these periods. 2) If the clocks of sender and receiver are not running at exactly the same rate, the receiver may count a different number of consecutively equal bits than what the sender intended to send.
- One workaround is to have nodes agree on a maximum number of permitted equal bits in a row. This requires encoding the data in a way that the resulting data bit stream exhibits the desired behavior.
- Another disadvantage of this coding is that there may be an undesirable bias towards $+1$ or -1, i.e., the mean value may not be 0.

Definition 5.40 (Manchester Coding)**.** ***Manchester coding*** *is a kind of line coding, in which every bit is represented by two values: '0' bits by first -1 then $+1$, and '1' bits by first $+1$ then -1.*

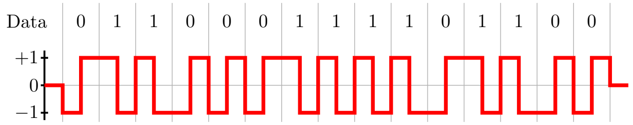


Figure 5.41: Manchester coding.

Remarks:

- Manchester coding solves the aforementioned problems with long runs of the same bit. In particular, the receiver may use the ongoing signal's edges to keep its clock in sync. It also exhibits no bias towards towards $+1$ or -1.

Definition 5.42 (Modulation)**.** *Expressing data bits as changes in the properties of a regular periodic waveform, the* ***carrier signal****, is called* ***modulation****.*

Definition 5.43 (Amplitude Modulation, AM)**.** ***Amplitude modulation*** *is a modulation which expresses data by varying the carrier signal's amplitude.*

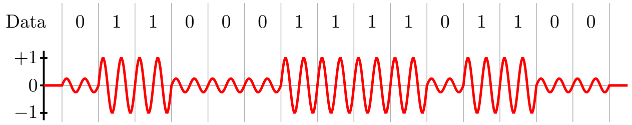


Figure 5.44: Amplitude modulation.

Remarks:

- The reception of amplitude modulated signals suffers greatly from noise, shadowing and signal transposition. For example, if the signal is reflected from a surface to reach a location behind a corner, the signal's power is decreased, which means the received amplitude value also decreases.

Definition 5.45 (Frequency Modulation, FM)**.** ***Frequency modulation*** *is a modulation which expresses data by varying the carrier signal's frequency.*

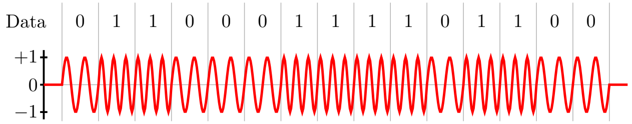


Figure 5.46: Frequency modulation.

Remarks:

- The frequency is usually varied only by small amounts, staying within a narrow frequency band.

- As opposed to amplitude modulated signals, frequency modulated signals are very robust to noise, which is one of the main reasons for the popularity of FM.

Definition 5.47 (Phase Modulation, PM)**.** ***Phase modulation*** *is a modulation which expresses data by varying the carrier signal's phase.*

Definition 5.49 (Symbols)**.** *Multiple data bits may be grouped into* ***symbols*** *before being encoded. This allows making use of the ability to represent more than 2 states at a time in the coding.*

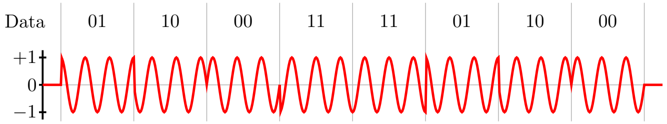


Figure 5.48: Phase modulation with 2-bit symbols.

Remarks:

- In reality, signals must be narrow-band, so "jumps" as they occur in Figure 5.48 must be avoided.

- The three modulation schemes presented above are often combined to express more different values with a single symbol.

- Encodings and modulations much more involved than the ones presented here have been designed optimized for parameters such as throughput or feasible SINR thresholds.

- A modulation encoding more data bits within the same time and frequency band raises the SINR threshold required for successful reception and vice versa.

Chapter Notes

ARP was introduced in RFC 826 [4] in 1982, the IPv6 counterpart NDP in RFC 4861 [6] in 2007.

DHCP was introduced as a more flexible application layer only replacement for the lower level Reverse Address Resolution Protocol (RARP) in RFC 1541 [5] in 1993.

Slotted ALOHA was one of the protocols developed for ALOHAnet [1], a project which aimed to wirelessly connect the Hawaiian island in 1971. As the name implies there also exists a slotless version of the protocol, which – while not requiring global time-slotting – suffers from a higher chance of collisions.

Consistent Overhead Byte Stuffing [2] was proposed to address the problem of naïve byte stuffing algorithms doubling the length of packets if their content happened to contain many instances of the reserved byte.

Manchester Coding, also known as *phase encoding*, while widely used at lower data rates (e.g. 10 MBit/s Ethernet) as an easy way to deal with clock errors, is less popular for higher data transmissions due to frequency-related problems [3].

This chapter was written in collaboration with Michael König.

Bibliography

[1] Norman Abramson. The aloha system: another alternative for computer communications. In *Proceedings of the November 17-19, 1970, fall joint computer conference*, pages 281–285. ACM, 1970.

[2] Stuart Cheshire and Mary Baker. Consistent overhead byte stuffing. *IEEE/ACM Transactions on Networking (TON)*, 7(2):159–172, 1999.

[3] Inc. Cisco Systems. Ethernet technologies. `http://docwiki.cisco.com/wiki/Ethernet_Technologies`. Accessed: 2017-03-22.

[4] David C. Plummer. RFC 826: An Ethernet Address Resolution Protocol – or – Converting Network Protocol Addresses, 1982.

[5] Bucknell University R. Droms. RFC 1541: Dynamic Host Configuration Protocol, 1993.

[6] E. Nordmark Sun Microsystems W. Simpson Daydreamer H. Soliman Elevate Technologies T. Narten, IBM. RFC 4861: Neighbor Discovery for IP version 6 (IPv6), 2007.