Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

FS 2018 Prof. R. Wattenhofer

Computer Engineering II

Solution to Exercise Sheet Chapter 5

Basic

1 MAC Addresses vs. IP Addresses

a)
- They operate on different layers in the network stack: link layer vs network layer.
- Different size (6 bytes vs 4 or 16 bytes) and notation
- Assigned by hardware manufacturer vs by network administrator or DHCP
- Used for routing vs used as unique identifier (esp. before an IP address is assigned)

b) MAC addresses are impractical for routing on the Internet as they are not grouped by network or location. (Instead they are grouped by manufacturer.)

c) Some kind of unique name is required to be able to execute any meaningful protocol when first joining a network.

2 Escape Sequences

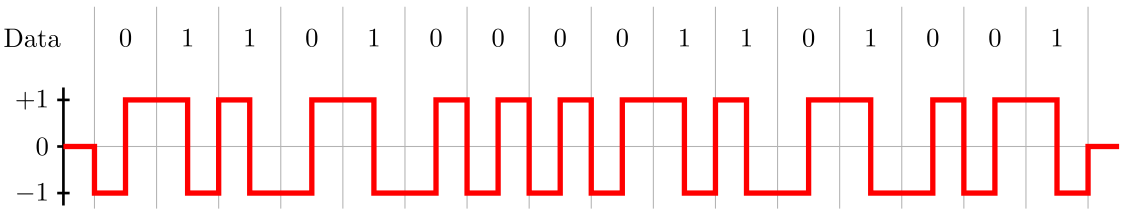
a) It is never possible to be sure a string was escaped, but some escaping schemes allow telling when a string has not been escaped, namely if it contains invalid byte sequences, i.e., `Y`z where $z \notin \{\mathtt{A}, \mathtt{B}\}$.

b) In software strings are usually parsed from the start, hence joining in the middle of an ongoing transmission as in the physical layer is not a concern. This means that the delimiter `X` may occur in the string as part of an escape sequence without being mistaken as the delimiter. Of course, escape sequences may still not start with an `X`.

c) `\"Oh no,\" Jon said, \"my cat \\\"Garfield\\\" is locked outside in the rain!\"`

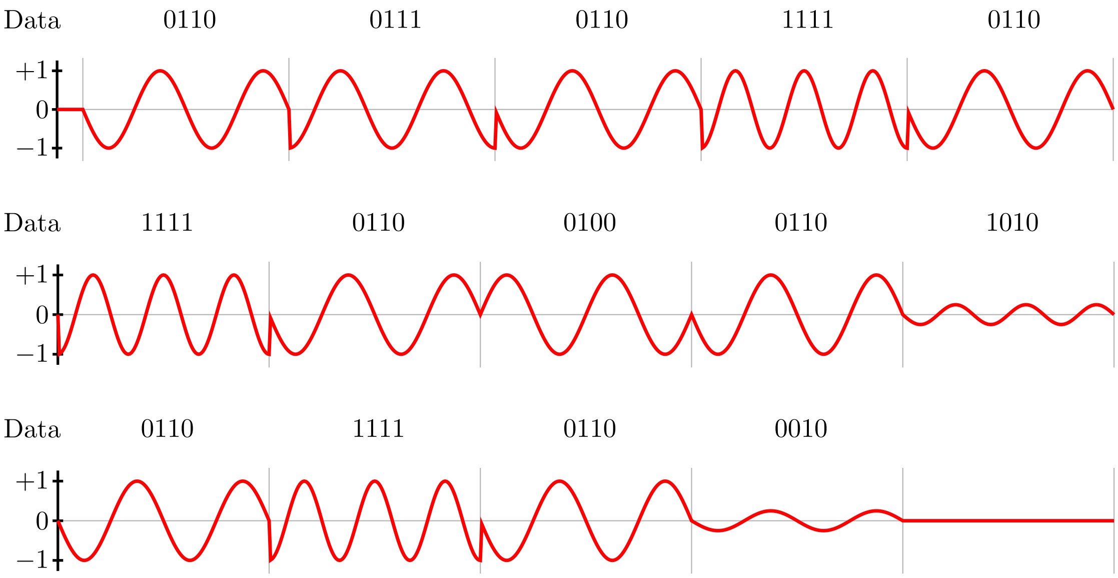
3 Manchester Decoding

The bits are `0110100001101001` (in order).
$01101000_2 = 104 =$ `ascii('h')`, $01101001_2 = 105 =$ `ascii('i')`.
Hence, the message is `hi`.



4 AM/FM/PM Demodulation

The symbols are `0110 0111 0110 1111 0110 1111 0110 0100 0110 1010 0110 1111 0110 0010` (in order). The message reads `goodjob`.



5 SINR

$$SINR_{A1} = \frac{7\,\text{nW}}{31\,\text{nW} + 0.01\,\text{nW}} = 0.226 < \beta$$
$$SINR_{A2} = \frac{52\,\text{nW}}{10\,\text{nW} + 5\,\text{nW}} = 3.467 < \beta$$
$$SINR_{B1} = \frac{31\,\text{nW}}{7\,\text{nW} + 0.01\,\text{nW}} = 4.422 > \beta$$
$$SINR_{B2} = \frac{10\,\text{nW}}{52\,\text{nW} + 5\,\text{nW}} = 0.175 < \beta$$

Thus, only the link $s_B \to r_2$ transmits and receives successfully.

Advanced

6 Bit Stuffing

Note that we just list example solutions here.

a) We append to every occurence of the string `01111` (a substring of `S`) the bit `1` (preventing `S` from occurring). Note that this operation is trivially reversible and hence allows for easy decoding.

b) The problem is that the `0` at the end of `S` may combine with the start of the packet into another instance of `S`. In principle, the same thing could happen at the end of the packet with the leading `0` of `S`, but this is not an issue for our solution from a).

Solution: Just add a `0` to the front of the packet *before* performing the bit insertions. Clearly, this operation is also reversible.

7 Code Division Decoding

a) The received signal was created from the following input:

Code	Data Bits
$(+,+,-,-)$	0010
$(+,-,+,-)$	1010

b) The received signal was created from the following input:

Code	Data Bits
$(+,+,+,+)$	1000
$(+,+,-,-)$	1110
$(+,-,+,-)$	0011
$(+,-,-,+)$	1100

c) In W_2, shifted codes may look like other codes. For example, $(+,+,-,-)$ may be created from $(+,-,-,+)$ and $(+,+,+,+)$ may be created from $(+,+,-,-)$. As a result, one signal can spoil the correlation values of other signals.

d) Choose long pseudorandom sequences as codes. This makes it impossible for one code to be mistaken as another. The downside is that such a scheme reduces the possible throughput. An additional benefit is that the codes' lengths can be chosen arbitrarily, which makes it possible to account even for severe noise.

e) This task is mostly easily solved by deducting the possible offsets for each code from alignments of the peaks. Once a code's offset has been identified, its bits may be determined and the contribution of the code may be subtracted from the signal. This does not work in general, but the specific signal given in the task allows for it. The table below shows one of the easier decoding orderings.

Code	Data Bits	Offset
$(+,+,+,+)$	10101	-1
$(+,-,+,-)$	01010	-2
$(+,-,-,+)$	01010	0
$(+,+,-,-)$	10001	0

8 Path Loss Sandwich

a) This is but an example solution. (LU stands for Length Unit, PU for Power Unit)

$$
\begin{aligned}
c &= 1\,\text{LU} \\
b &= 10\,\text{LU} \\
a &= 1337\,\text{LU} \\
P_C &= 1\,\text{PU} \\
P_B &= 15\,\text{PU} \\
P_A &= 300\,\text{PU} \\
SINR_A &= \frac{\frac{300}{2695^2}}{\frac{1}{1348^2} + \frac{15}{1358^2}}\,\text{PU} = 4.756 > \beta \\
SINR_B &= \frac{\frac{15}{21^2}}{\frac{1}{11^2} + \frac{300}{1358^2}}\,\text{PU} = 4.036 > \beta \\
SINR_C &= \frac{\frac{1}{1^2}}{\frac{15}{11^2} + \frac{300}{1348^2}}\,\text{PU} = 8.056 > \beta
\end{aligned}
$$

b) In practice, such specific node constellations are rare unless specifically set up by hand. Additionally, minimum signal reception strengths increase the required transmission strengths, such that most hardware is unable to offer the required transmit power settings spanning values as low as 1 PU and as high as 300 PU.