
A Layer 1 Refresher

Bits, Modulation, et al.

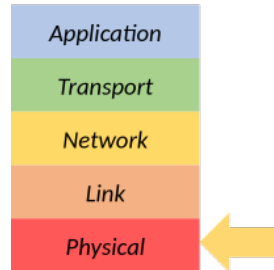
Tobias Fiebig

Technische Universität Wien

A solid blue vertical bar is positioned on the right side of the slide, extending from the top of the content area to the bottom.

Hello, Physical Layer!

Let's start with the lowest layer in the protocol hierarchy:
Layer 1, the "Physical Layer".



- Exchanging information is what makes systems “communicate”
- Digital systems communicate in “bits”
- How to get a bit transmitted?

- Multiple steps of preparation / transformation
 1. Source encoder / compression
 2. Channel encoder / CRC redundancy and block codes
 3. Line encoder / line codes
 4. Modulation
- Receiver needs to reverse these steps



- The transmission medium can either be a
 - Wire
 - Bits are modulated into voltage
 - Air
 - Bits are modulated into frequency or amplitude
- The now presented principles are agnostic to the medium
- The terms “wire” and “voltage” are interchangeably used for “medium” / “air” and “signal” / “frequency” from now.



- “Clock Recovery”
 - Property of a line code describing if sender and receiver require no dedicated wire for clock synchronization
 - Transmission parties can only “understand” each other if their en- and decoding clock are in sync
- “Running Digital Sum (RDS)”
 - Describes the share of “DC (direct current) component” on the wire
 - $\rightarrow +\infty$ or $\rightarrow -\infty$ if the line code produces high DC share on the wire
 - Around 0 implies wire is DC free, which is healthy for connected devices
- “Baud rate”
 - Describes rate at which voltage levels on the wire change
 - Higher Baud rates allow faster transmission of the same information, but are also more error-prone and hence have a lower reach

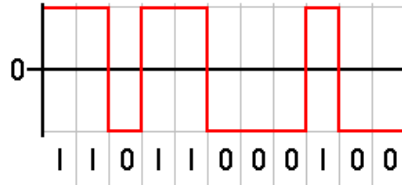


- Binary codes
 - Just translate bit into voltage, e.g.:
 - High voltage: 1
 - Low voltage: 0
- Biphase codes
 - Translate bit into voltage jumps, e.g.:
 - Voltage increase: 1
 - Voltage drop: 0
- Ternary codes
 - Introduce a third voltage level, e.g.:
 - Low voltage: (not sending)
 - High positive voltage: 1
 - High negative voltage: 0



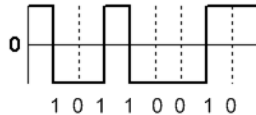
Non-Return-to-Zero (NRZ)

- Just translate bits into voltage
- High voltage: “1”
- Low voltage: “0”
- No clock recovery
- Baud rate well used for information



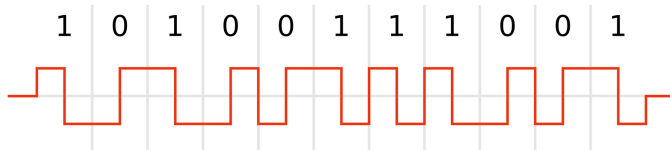
Non-Return-to-Zero Inverted (NRZI)

- Change to other voltage means “1”
- No clock recovery
- Baud rate well used for information



Manchester Code

- Introduce additional voltage jumps at the cost of transmission rate
- They are used to prepare the encoding of the actual data bit
- “0”: voltage increase after the preparation interval
- “1”: voltage drop after the preparation interval
- Clock recovery possible
- Maximum line rate efficiency: 50% (you need a bit for each bit)



Return to Zero (RZ)

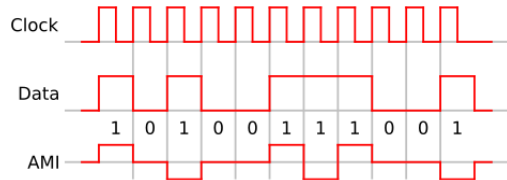
- High voltage: 1
- Low voltage: 0
- Always return to “neutral” voltage level after transmission
- Clock recovery possible
- Half of the Baud rate used for voltage level return



Ternary Code (II)

Alternate Mark Inversion (AMI)

- Neutral voltage: 0
- High and low (alternating) voltage: 1
- Neutral voltage: 0
- Always return to “neutral” voltage level after transmission
- No clock recovery
- Baud rate well used



- Introduce mapping of N data bits to M code bits
- 4b/5b code example
 - Map 4-bit data word to 5-bit code word
 - Chose 5-bit words such that any combination of sent code words
 - ensures the line is free of DC component, i.e., only chose words with $DRS = 0$
 - allows clock recovery
 - This works well with NRZ codes and helped them to establish market share
 - USB3.0 / HDMI / S-ATA use 8b/10b codes
 - 100G Ethernet uses 64b/66b codes
 - NVlink / PCIe 5.0 use 128b/1xxb codes
- 4b/3t codes also possible
 - Map a 4 bit data word to a 3 “ternary” word
 - Exclude “neutral” ternary word to avoid RDS $\rightarrow \infty$

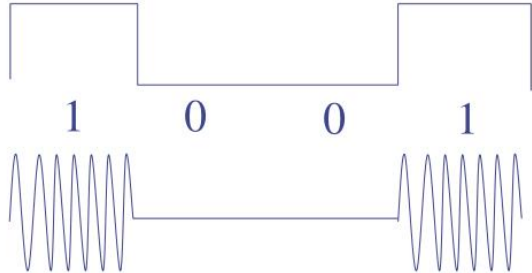


- Transformation from digital to analogue signal
- Relies on Fourier transformation
- Amplitude-shift keying
- Frequency-shift keying
- Phase-shift keying
- “Modem” is short for “Modulation and Demodulation”



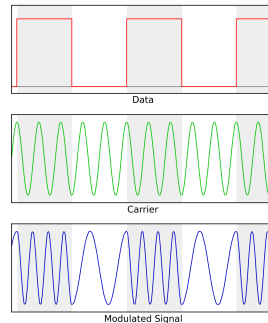
Amplitude-Shift Keying (ASK)

- Signal has constant frequency
- To modulate a “1” transmitter is connected to an oscillator
- To modulate a “0”, transmitter gets pulled to GND
- Technically easy to implement
- Low bandwidth requirements
- Prone to interference



Frequency-Shift Keying (FSK)

- Signal has constant amplitude
- Two different but constant oscillators get connected to transmitter to modulate “0” and “1”
- Technically easy to implement
- Higher bandwidths possible



Phase-Shift Keying (PSK)

- Constant frequency and amplitude
- Phase gets inverted to modulate a transition in the data message bits
- Technically a bit more complicated, especially demodulation
- Relatively robust to transmission interference

