

Data Networks

UdS and IMPRS-CS

Lecture 5: Data Link-Layer Problems

Thanks to Eugene Ng, Rice University

Bit Stream Encoding

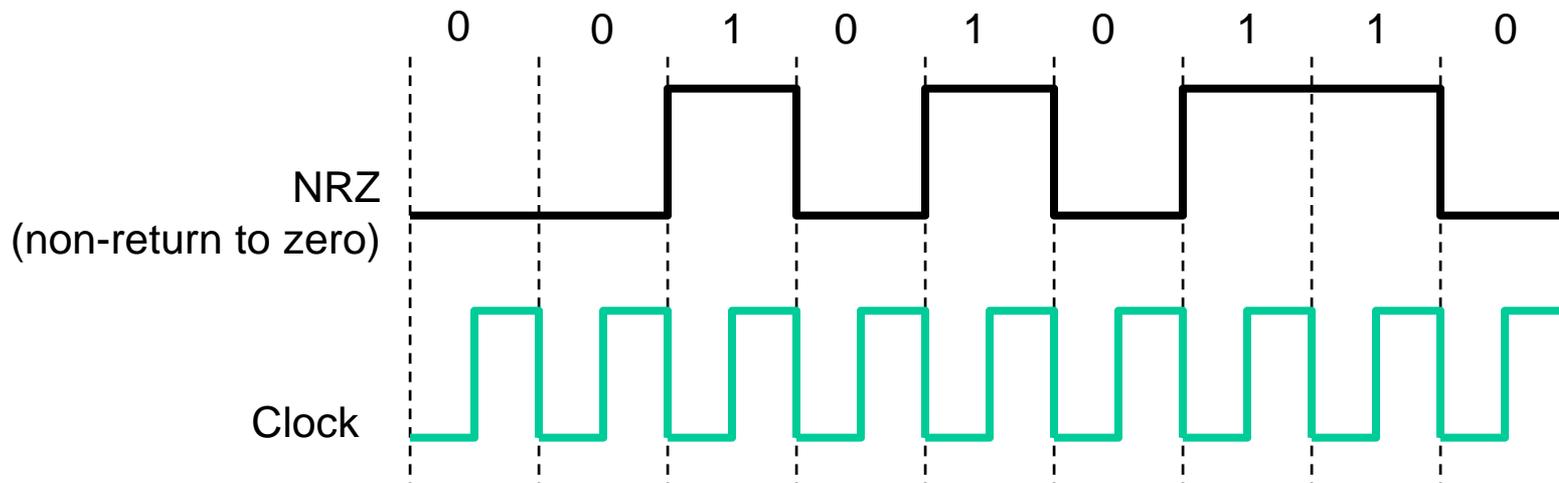
- Specify how bits are represented in the analog signal
 - This service is provided by the **physical** layer
- Challenges:
 - Efficiency: ideally, bit rate is maximized
 - Robust: avoid de-synchronization between sender and receiver when there is a large sequence of 1's or 0's

Assumptions

- We use two discrete signals, high and low, to encode 1 and 0
- The transmission is synchronous, i.e., there is a clock used to sample the signal
- If the amplitude and duration of the signals is large enough, the receiver can do a reasonable job of looking at the distorted signal and estimating what was sent.

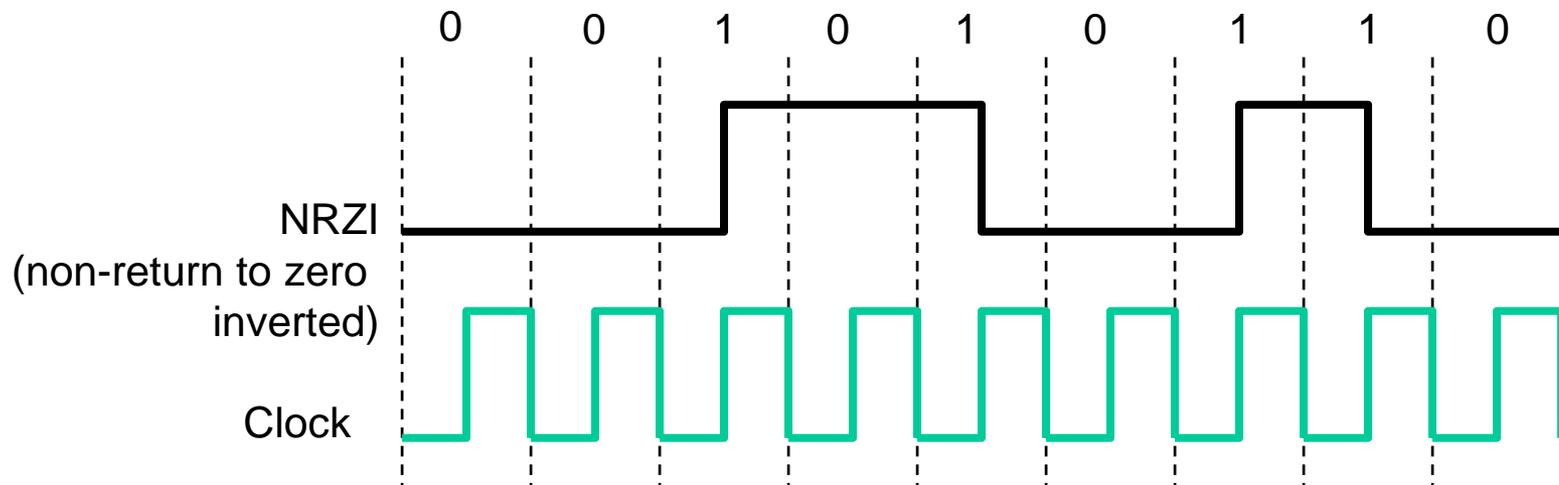
Non-Return to Zero (NRZ)

- 1 → high signal; 0 → low signal
- Disadvantages: when there is a long sequence of 1's or 0's
 - Sensitive to clock skew, i.e., difficult to do clock recovery



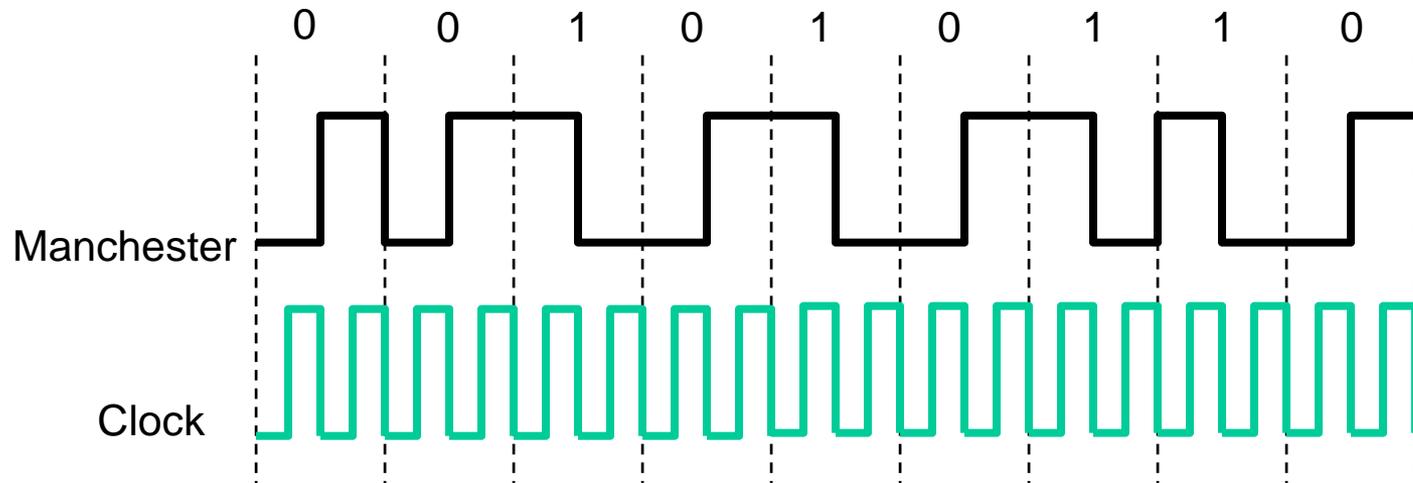
Non-Return to Zero Inverted (NRZI)

- 1 → make transition; 0 → stay at the same level
- Solve previous problems for long sequences of 1's, but not for 0's



Manchester

- 1 → high-to-low transition; 0 → low-to-high transition
- Addresses clock recovery problems
- Disadvantage: signal transition rate doubled
 - I.e. useful data rate on same physical medium halved
 - Efficiency of 50%



4-bit/5-bit (100Mb/s Ethernet)

- Goal: address inefficiency of Manchester encoding, while avoiding long periods of low signals
- Solution:
 - Use 5 bits to encode every sequence of four bits such that no 5 bit code has more than one leading 0 and two trailing 0's
 - Use NRZI to encode the 5 bit codes
 - Efficiency is 80%

4-bit	5-bit	4-bit	5-bit
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Framing

- Specify how **blocks** of data are transmitted between two nodes connected on the same physical media
 - This service is provided by the **data link** layer
- Challenges
 - Decide when a frame starts/ends
 - If use special delimiters, differentiate between the true frame delimiters and delimiters appearing in the payload data

Byte-Oriented Protocols: Sentinel Approach



- STX – start of text
- ETX – end of text
- Problem: what if ETX appears in the data portion of the frame?
- Solution
 - If ETX appears in the data, introduce a special character DLE (Data Link Escape) before it
 - If DLE appears in the text, introduce another DLE character before it
 - Like in C programming, “Say `\"Hello\\\"`”, (`\\` is the escape character)

Byte-Oriented Protocols: Byte Counting Approach

- Sender: insert the length of the data (in bytes) at the beginning of the frame, i.e., in the frame header
- Receiver: extract this length and decrement it every time a byte is read. When this counter becomes zero, we are done

Bit-Oriented Protocols



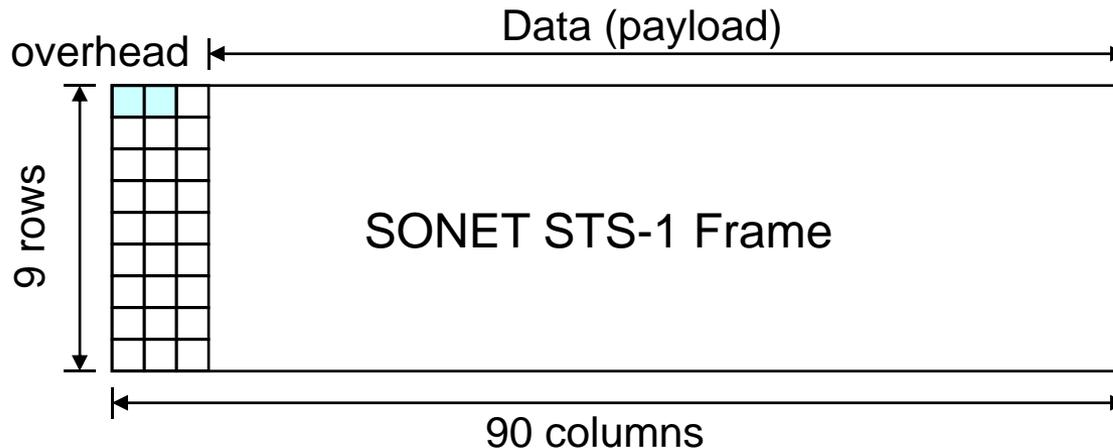
- Both start and end sequence can be the same
 - E.g., 01111110 in HDLC (High-level Data Link Protocol)
- Sender: in data portion inserts a 0 after five consecutive 1s
 - “Bit stuffing”
- Receiver: when it sees five 1s makes decision on the next **two** bits
 - If next bit 0 (this is a stuffed bit), remove it
 - If next bit 1, look at the next bit
 - If 0 this is end-of-frame (receiver has seen 01111110)
 - If 1 this is an error, discard the frame (receiver has seen 01111111)

Clock-Based Framing (SONET)

- SONET (Synchronous Optical NETWORK)
- Developed to transmit data over optical links
 - Example: SONET STS-1: 51.84 Mbps
- SONET maintains clock synchronization across several adjacent links to form a path

STS-1 Frame

- First two bytes of each frame contain a special bit pattern that allows to determine where the frame starts
- No bit-stuffing is used, frame is fixed size
- Receiver looks for the special bit pattern every 810 bytes
 - Size of frame = $9 \times 90 = 810$ bytes



Clock-Based Framing (SONET)

- Details:
 - Bits are encoded using NRZ
 - To avoid long sequences of 0's or 1's the payload is XOR-ed with a special 127-bit pattern with many transitions from 1 to 0