

Link Layer (I)

Overview

- The data link layer has two components:
 - Data Link Control:
 - addressing
 - “MAC” addresses used in frame headers to identify source, destination
 - error detection
 - errors caused by signal attenuation, noise
 - receiver detects presence of errors:
 - error correction
 - receiver signals sender for retransmission or drops frame
 - receiver identifies and corrects bit error(s) without resorting to retransmission (forward error correction: receiver executes error-correcting code that generates the actual frame. e.g. Humming Codes, Binary convolution Code, Reed-Solomon Code, Low-Density Parity-Check Code)
 - flow control
 - pacing between adjacent sending and receiving nodes
 - Media Access Control (MAC)

Multiple Access Protocols

- Media Access Control enforces a methodology to allow multiple devices access a shared media network.
 - A shared media network → single shared broadcast channel
 - Multiple devices → two or more simultaneous transmissions by nodes:
 - interference
 - collision: if node receives two or more signals at the same time
 - Methodology → multiple access protocol
 - distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- Note: communication about channel sharing must use channel itself!
 - no out-of-band channel for coordination

An ideal multiple access protocol

- given: broadcast channel of rate R bps
- goals:
 - high utilization: when one node wants to transmit, it can send at rate R
 - fairness: when M nodes want to transmit, each can send at average rate R/M
 - fully decentralized

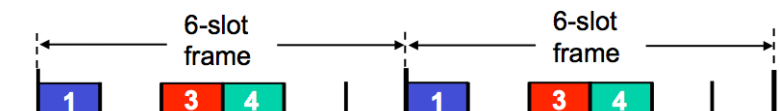
- no special node to coordinate transmissions
- no synchronization of clocks, slots
- simple

Three broad classes

- **channel partitioning**
 - divide channel into smaller “pieces” (time slots, frequency, code)
 - allocate piece to node for exclusive use
- **random access**
 - channel not divided, allow collisions
 - “recover” from collisions
- **“taking turns”**
 - nodes take turns, but nodes with more to send can take longer turns

TDMA

- **TDMA: time division multiple access**
- channel partitioning
- divide time into time frames, and then divide each frame into N time slots
- each node gets fixed length slot (length = one packet transmission time)
- when a node has packets to send, it transmits the packet during the assigned time slot in the revolving time frames
- unused slots go idle
- example: 6-host LAN, 1, 3, 4 have packets to send, slots 2, 5, 6 idle

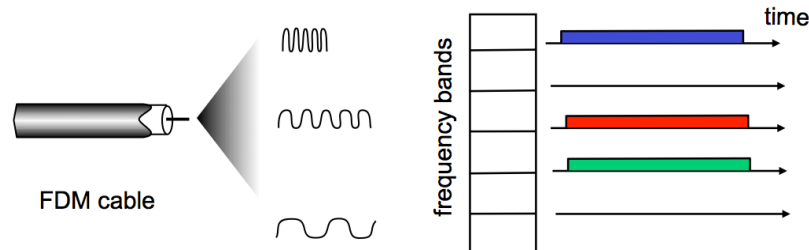


- advantages:
 - eliminates collisions
 - perfectly fair (transmission rate for each node R/N bps per frame time)
- disadvantages:
 - a node is limited to an average rate R/N bps even when only one node with packets to send
 - a node must wait for its turn in the transmission sequence, even if it is the only one with packets to send

FDMA

- **FDMA: frequency division multiple access**

- channel partitioning
- divide the R bps channel into different frequencies (each with R/N bps)
- each node assigned a frequency
- a node uses the assigned frequency to transmit packets
- example: 6-host LAM, 1, 3, 4 have packet to send, frequency bands 2, 5, 6 idea

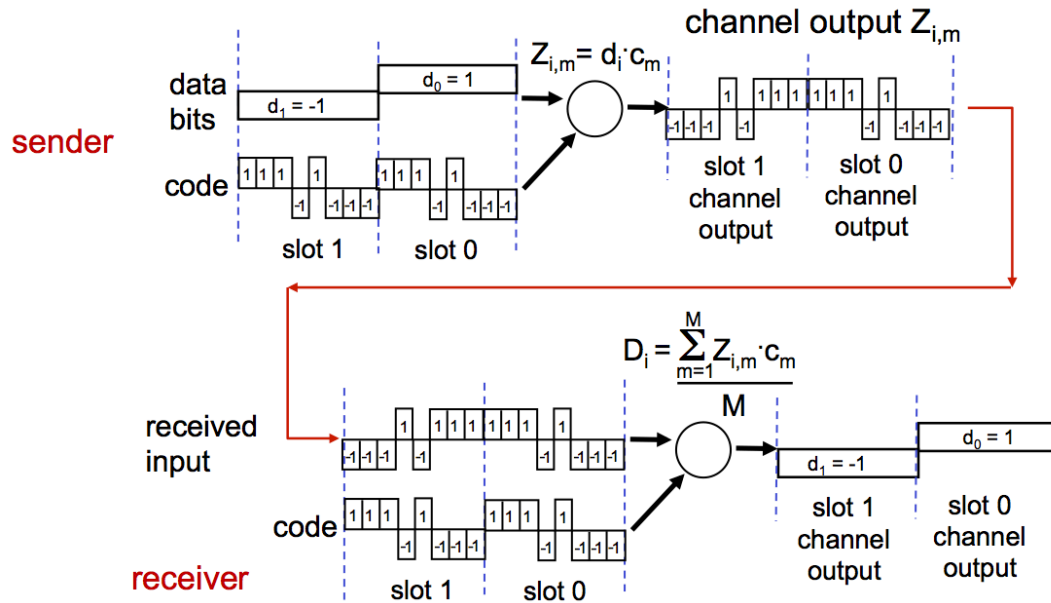


- advantages:
 - same as TDMA
- disadvantage:
 - a node is limited to an average rate R/N bps even when only one node with packets to send
- improvement:
 - no need to wait till the assigned slot in next time frame to transmit

CDMA

- **CDMA: code division multiple access**
- channel partitioning
- unique “code” assigned to each user; i.e., code set partitioning
- all users share same frequency, but each user has own “chipping” sequence (i.e., code, a signal that changes at a much faster rate than the original sequence of data bit) to encode data
 - each bit being sent is encoded by multiplying the bit by the code
- allows multiple users to “coexist” and transmit simultaneously with minimal interference (if codes are “orthogonal”)
- used in 2G network, and widely used in wireless network in rural area and telephony network

- example: one sender (ideal case)



encode:

$$Z_{i,m} = d_i \cdot c_m$$

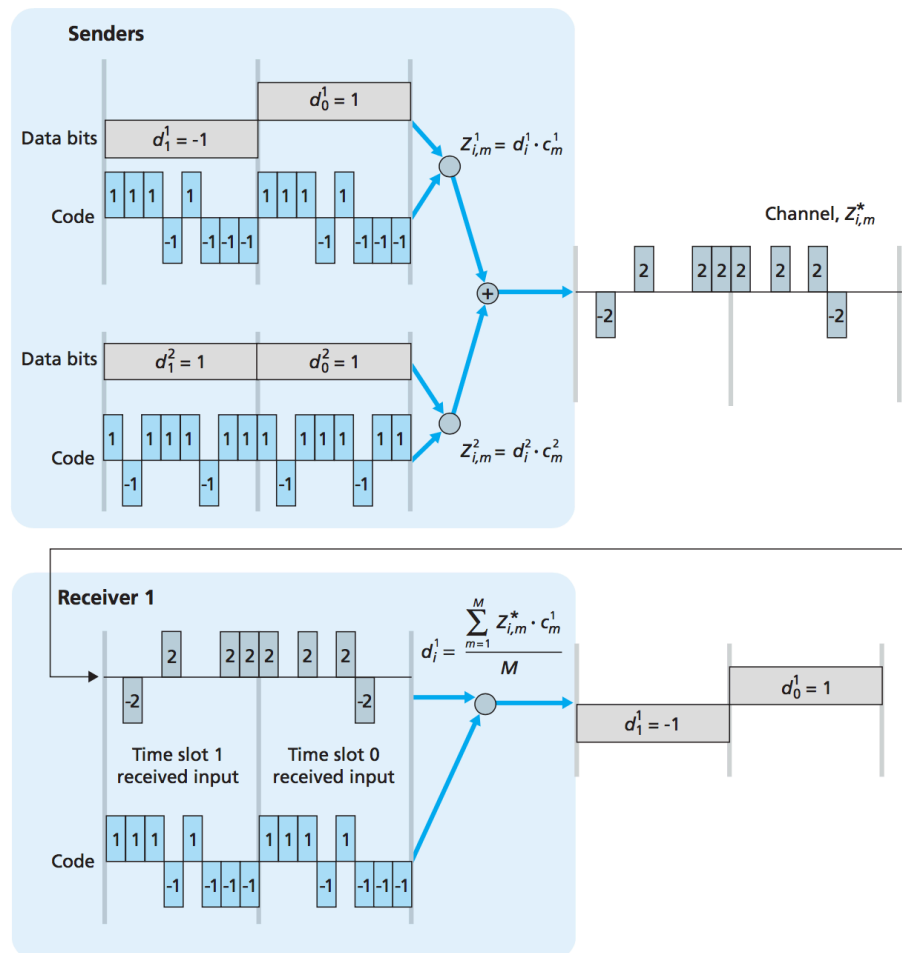
decode:

$$d_i = \frac{1}{M} \sum_{m=1}^M Z_{i,m} \cdot c_m$$

where d_i is the data bit, the bit-transmission time of d_i is divided into M mini-slot. $Z_{i,m}$ is the output of the encoder for the m th mini-slot of the transmission time, c_m is the m th bit of the code

- Note: for mathematic simplicity, data bit 0 is represented in -1.

- example: two sender (conflict)



encode:
$$z_{i,m}^* = \sum_{s=1}^N z_{i,m}^s$$

decode:
$$d_i = \frac{1}{M} \sum_{m=1}^M z_{i,m}^* \cdot c_m$$

where N is the number of senders

- Note:

- The CDMA **codes must be carefully chosen** in order for the CDMA receivers to be able to extract a particular sender's signal.
- The above discussion has **assumed that the received signal strengths from various senders are the same**.

Disclaimer: Notes adapted from the textbook and online resources.