Data Link Layer

Reference:
Sections 6.1, 8.1, 8.2 and 8.3 of Stallings
Chapter 6 Forouzan
Study Guide 5

Review of Lecture 4

- List and briefly define important factors that can be used in evaluating or comparing the various digital-to-digital encoding techniques.

Review of Lecture 4

- List and briefly define important factors that can be used in evaluating or comparing the various digital-to-digital encoding techniques.
- Answer - Signal spectrum (frequency, dc, distortion), Clocking (synchronise), error detection (built in error detection), signal interference and noise immunity (superior performance in the presence of noise), cost and complexity (higher signal rate to achieve a given data rate).

Review of Lecture 4

- Define biphase encoding and describe two biphase techniques.

Review of Lecture 4

- Define biphase encoding and describe two biphase techniques.
- Answer - A biphase scheme requires at least one transition per bit time and may have as many as two transitions. Manchester and differential Manchester (discuss).

Review of Lecture 4

- What does the sampling theorem tell us concerning the rate of sampling required for an analog signal?

Review of Lecture 4

- What does the sampling theorem tell us concerning the rate of sampling required for an analog signal?
- Answer - The sampling rate must be higher than twice the highest signal frequency.

Review of Lecture 4

- What are the differences among angle modulation, PM, and FM?

Review of Lecture 4

- What are the differences among angle modulation, PM, and FM?
- Answer - Frequency modulation (FM) and phase modulation (PM) are special cases of angle modulation. For PM, the phase is proportional to the modulating signal. For FM, the derivative of the phase is proportional to the modulating signal.

Lecture 5 Objectives

- **Understand the reasons for having a data link layer;**
- **describe serial transmission techniques;**
- **describe the generic structure of a data link layer frame;**
- **understand the reasons for using multiplexing;**
- **describe Frequency Division Multiplexing;**
- **describe Time Division Multiplexing; and**
- **describe Statistical Time Division Multiplexing.**

Introduction

- The Data Link Layer provides the functionality to manage and control the data link, as the Physical Layer cannot provide addressing, error checking, link management, flow control or a technique to share the data link.

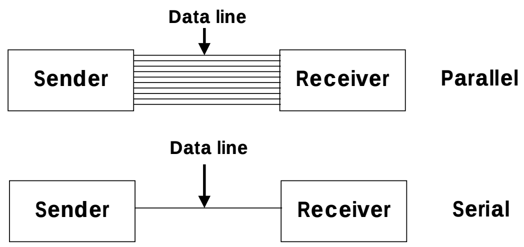
Introduction ….

- The generic structure of the Data Link Layer frame includes:
  - Addressing information, hence a source and destination address can be indicated.
  - Error checking information, this ensures corrupt frames are not accepted.
  - Control onformation, to support link management.

Serial vs Parallel Transmission

- Serial Transmission
  - data are transmitted one signal element at a time over a single path
  - Used in most data/information exchange
- Parallel Transmission
  - A group of bits is transmitted simultaneously over a bundle of lines in a cable
  - Used in communication within a computer system
  - Normally used where distance between the two devices are short

Serial vs Parallel Transmission



Parity Bit

- Parity bit provides a simple error detection mechanism. It does not detect the position of the error
- Type: Odd and even parity
  - Odd parity-the number of binary ‘1’ in the bit stream has to be equal to an odd number. If not the case, an error has occurred during transmission
  - Example: 10101101 -> does not contain any error, 10101100 -> contains error because the number of ‘1’ s is 4 (even number)

Parity Bit …

  - Even parity - the number of binary ‘1’ in the bit stream has to be an *even* number
  - Example, 11001100 does not contain error, 11001101 contains error
- ASCII character transmission uses 8 bits, 7 bit to represent the actual character, 1 bit for parity bit.
- What is the parity bit for 1000001? If odd parity bit is used in transmission (1 or 0?)
  - Answer: 1 , the bits become 10000011

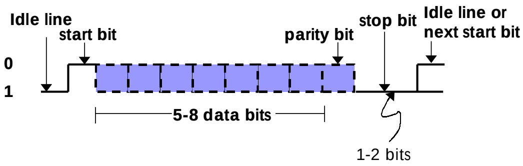
Synchronization in transmission

- Problem arises if sender and receiver clocks are not precisely aligned. Receiver will sample received data at a different rate
- Synchronization between sending and receiving end is very important
- Synchronization can be achieved through:
  - asynchronous transmission
  - synchronous transmission
  - determined by whether transmitter and receiver clocks are independent (asynchronous) and synchronized (synchronous)

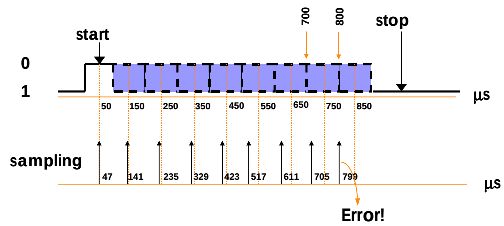
Asynchronous Transmission

- Avoid timing problem by not sending long stream of bits
- Data are transmitted one character at a time
- Receiver will resynchronize at the beginning of each character
- Synchronization is maintained within each character with start and stop codes
  - idle state - no data on line, equivalent to “1”
  - start bit - equivalent to “0”
  - data stream - 5 to 8 bits
  - parity bit - one bit for error detection
  - stop bit - equivalent to “1” of duration 1, 1,5 or 2 bit

Asynchronous Transmission …



Asynchronous Transmission …



100 µs bit duration and receiver samples at every 94 µs

Asynchronous Transmission…

- Example, a communication system needs to transmit the ASCII characters ‘ABC’ using even parity bit. The stop bit (1) is one bit-time and the start bit (0) starts the timing sequence for the next nine elements.
  - 0100000101 0100010111 0100001111 1111

Asynchronous Transmission…

- Provides a modest synchronization
- Overhead of 2 or 3 bits per char (~20%)
- Overhead can be reduced by increasing the number of characters sent between the start and stop bits, but larger blocks lead to greater cumulative timing error

Synchronous Transmission

- A block of data is transmitted in a steady stream without start or stop bits
- Clocks must be synchronized
  - Can use separate clock line
    - subject to impairments
    - good over short distance, not over long distance
  - Embed clock signal in data
- Need another level of synchronization
  - Flags to determine beginning and end of the block, normally called preamble & postamble

preamble	data	postamble

- Preamble+data+postamble=Frame

Synchronous Transmission …

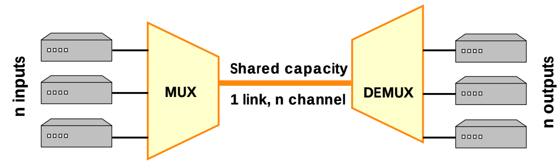
- Generic Fields
  - Frame Start Field: a sequence of bits at the beginning indicating the start of the frame
  - Address Field: address of source and/or destination
  - Length/Type/Control Field: indicates length, type or control field for sequencing and providing acknowledgement
  - Data Field: contains data
  - Frame Check Sequence: for error detection and control
- An example, HDLC in later lecture

Synchronous Transmission …

- Far more efficient than asynchronous transmission
- On average, asynchronous transmission requires 20% or more overhead
- Example of a well known synchronous transmission is HDLC. It contains 48 bits of controls, preamble and postamble
  - for a data frame of 1000 character, overhead=0.6%

Multiplexing

- A function that allows multiple transmission sources to ***share*** a larger common transmission capacity



Why Multiplexing ?

- The higher the data rate, the more cost effective the transmission facility
  - the cost per-kbps declines with an increase in the data rate
- Not many individual data communication devices requires high data rate support

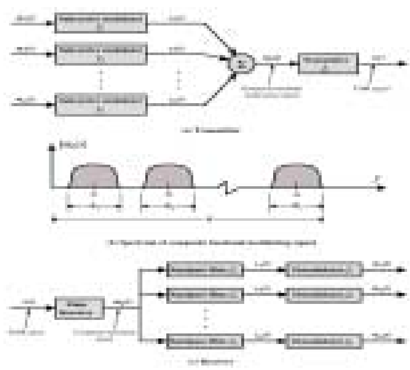
Types of Multiplexing

- Frequency Division Multiplexing (FDM)
- Time Division Multiplexing (TDM)
- Statistical Time division Multiplexing (STDM)

Frequency Division Multiplexing

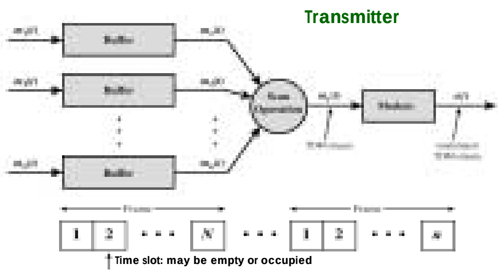
- Division of a transmission link into multiple channels by splitting the frequency band
- Used when useful bandwidth of the link is greater than required bandwidth of signals to be transmitted
- Each signal is modulated to a different carrier frequency
- Carrier frequencies are separated by guard bands so that signals do not overlap

Frequency Division Multiplexing…

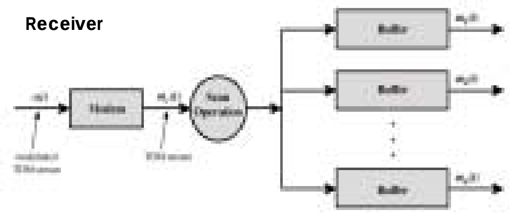


Time-Division Multiplexing

- Time slots on a shared medium are assigned to devices on a fixed, predetermined basis



Time-Division Multiplexing…



- Used when data rate of the medium exceeds data rate of digital signal to be transmitted

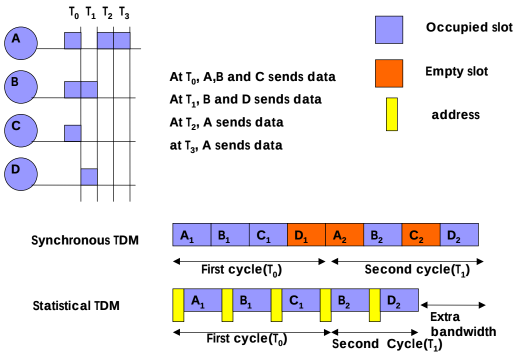
Time-Division Multiplexing…

- Each buffer typically one bit or one character in length
- Buffer is scanned in round-robin fashion
- Scan operation are sufficiently rapid that each buffer is emptied before more data can arrive
- In each frame, one or more slots is dedicated to each data source
- Sequence of slots dedicated to one source, from frame to frame, is called a channel
- Flow control and error control are provided on a per-channel basis using HDLC on a per-channel basis
- many slots are wasted

Statistical TDM

- Time slots are assigned to a device dynamically on demand
- For n input, only ***k (k<n)*** time slots are available
- Multiplexer scans input channels and collects data until the frame is full, ***does not send empty slots***
- Positional significance of the slots is lost. So, *source address is included in each time slot*
- Address information introduces more overhead
- May cause problems during peak periods
  - buffer in multiplexer to hold temporary excess data
  - keep buffer size to minimum to reduce delay

Statistical TDM…



Example: Multiplexing

- Adopted from Stallings q8. 11 (7^{th} & 6^{th} ed)
- A character- interleaved TDM is used to combine the data streams of a number of 110- bps and five of 300 bps asynchronous terminals for data transmission over a 4800-bps digital line. Each terminal sends asynchronous characters consisting of 7 data bits, 1 parity bit, 1 start bit and 1 stop bit. Assume that one synchronization character is sent every 19 data characters and, in addition, at least 3 percent of the line capacity is reserved for pulse stuffing to accommodate variations from the various terminal
- Determine:
  - Number of bits per character
  - Data rate available for data transmission
  - Number of 110- bps terminal that can be accommodated

Example: Multiplexing…

- Number of bits per- character
  - 7 + 1 + 1 + 1 = 10 bits
- Data rate available for data transmission
  - (100%- 3%) x 4800 = 4656 bps
- Number of 110- bps terminals
  - 300 bps terminals require => 5 x 300 = 1500 bps
  - No of chars sent per second = 4656 : 10 = 465.6 chars per second
  - 1 SYNC char for every 19 data chars (assume that no SYNC char sent at the start) => 465 : 20 = 23 chars of SYNC per second
  - 23 chars = 230 bits
  - Remaining available data rate = 4656 – (1500+ 230) = 2926 bps
  - No of terminals = 2926 : 110 = 26.6 => 26 terminals