EECS, University of Ottawa

ELG5374 –Fall 2021

Computer Communication Network

Digital Data Communication Techniques

Note: some material in the slides has been taken from various other sources 1-1

1

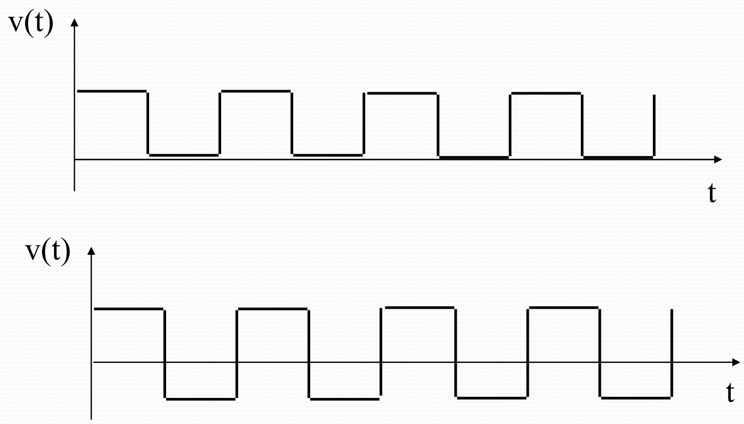
Outline

- Synchronization
- Asynchronous and synchronous transmission
- Error detection/correction techniques
- Line Configuration

1-2

2

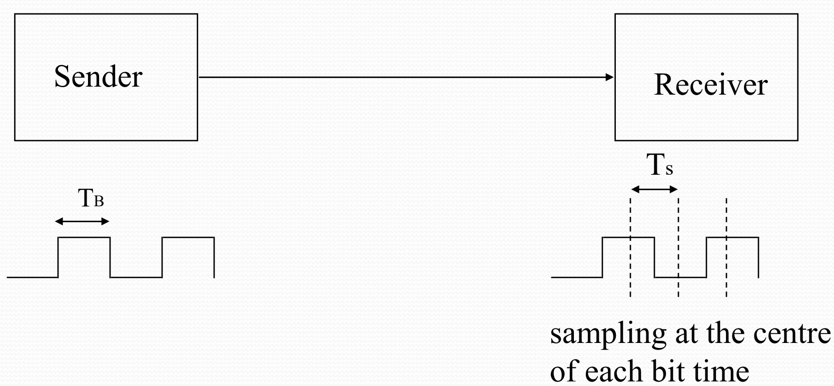
Digital Data/Digital Signals



1-3

3

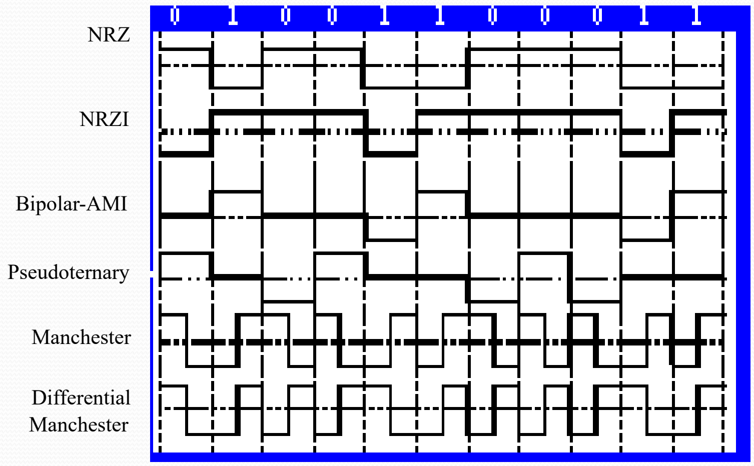
SYNCHRONIZATION



1-4

4

Some encoding techniques



1-5

5

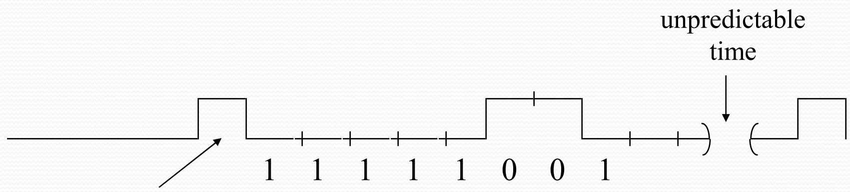
SYNCHRONIZATION

- loss of synchronization
  - In practice T_B and T_S are not equal. The result is that the timing of the receiver may slowly drift relative to the received signal.

1-6

6

SYNCHRONIZATION



1-7

7

SYNCHRONIZATION

- loss of synchronization
  - Solution:
    - data is sent in bit sequences called frames
    - the receiving clock is started at the beginning of each bit sequence
  - Question:

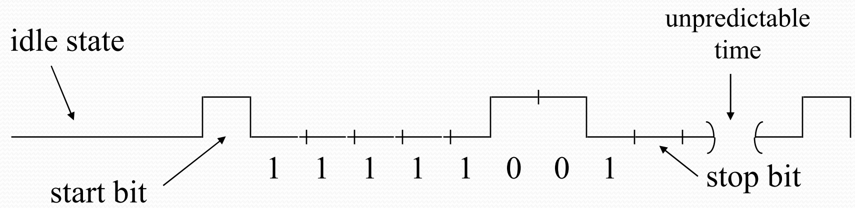
    Since synchronization needs to be kept only for the duration of the frame, what is the length of the frame that will allow us to avoid loss of synchronization?

1-8

8

ASYNCHRONOUS TRANSMISSION

- Timing or synchronization must only be maintained within each character; the receiver has the opportunity to resynchronize at the beginning of each new character.



SYNCHRONIZATION

- loss of synchronization

  example:
  - a frame consists of 11 bits
  - assume that the synchronization at the start of the first bit is late at most 10% of T_B

We must fulfill the following 2 conditions:

$$(10+\frac{1}{2})\times T_S + 0.1\times T_B < 11\times T_B \quad \text{and} \quad (10+\frac{1}{2})\times T_S > 10\times T_B$$

These are satisfied if: $\left|\frac{T_s - T_B}{T_B}\right| < 3.8\%$

ASYNCHRONOUS TRANSMISSION

- Timing requirements are modest. Sender and receiver are synchronized at the beginning of every character (8 bits if ASCII)
  - high overhead

$$overhead = \frac{control\_bits}{total\_bits}$$

1-11

11

SYNCHRONOUS TRANSMISSION

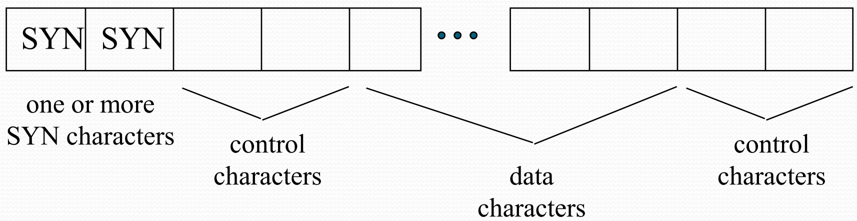
- In this mode, blocks of characters or bits are transmitted. Each block begins with a preamble and ends with a postamble
  - 2 types:
    - character-oriented
    - bit-oriented

1-12

12

SYNCHRONOUS TRANSMISSION

- Character-oriented
  - the frame consists of a sequence of characters



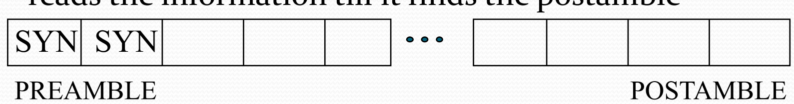
SYN is a unique bit pattern that signals the receiver the beginning of a block

1-13

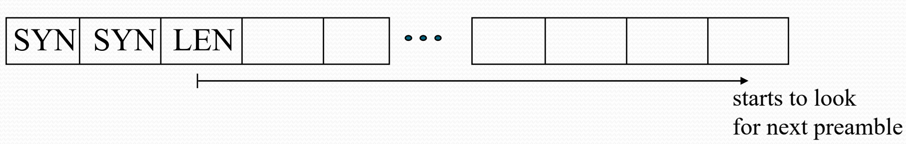
13

SYNCHRONOUS TRANSMISSION

- character-oriented - 2 approaches
  - the receiver having detected the beginning of the block reads the information till it finds the postamble



the receiver having received the preamble, looks for extra information regarding the length of the frame

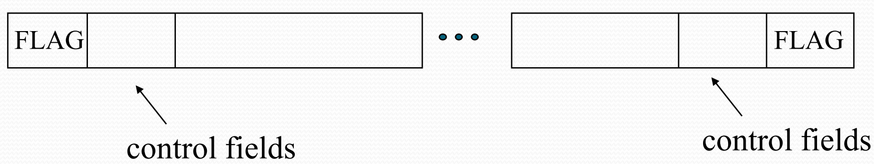


1-14

14

SYNCHRONOUS TRANSMISSION

- Bit oriented
  - In this mode, the frame is treated as a sequence of bits. Neither data nor control information is interpreted in units of x-bit characters
  - a special bit pattern indicates the beginning of a frame
  - the receiver looks for the occurrence of the flag



1-15

15

Dealing with presence of errors

- Detect presence of errors (error detection)
- Try to correct them (error correction)
- If no correction have the mechanism to request retransmission (use of Automatic Repeat Request)

1-16

16

Random Errors

- An error occurs when a bit is altered between transmission and reception
- Random, statistically uniformly spread errors
  - occurrence of an error does not increase the probability that other bits, close to the one in error, while be in error
  - white noise is producing such errors
- For low BER and frames of “reasonable” length, most framers would experience no error or 1 error at most.
  - example: BER = 10^{-6} and length of frame=[1000bytes*8=] 8,000 bits
  - probability of receiving a frame correctly = [1-10^{-6}]^{8,000}>0.992
  - probability of a frame having a single error=8,000*receiving a frame correctly = 10^{-6}*[1-10^{-6}]^{7,999}* {8,000}=0.007873
  - Probability of having more than 1 error<[1-0.992-0.007873=]0.000127

1-17

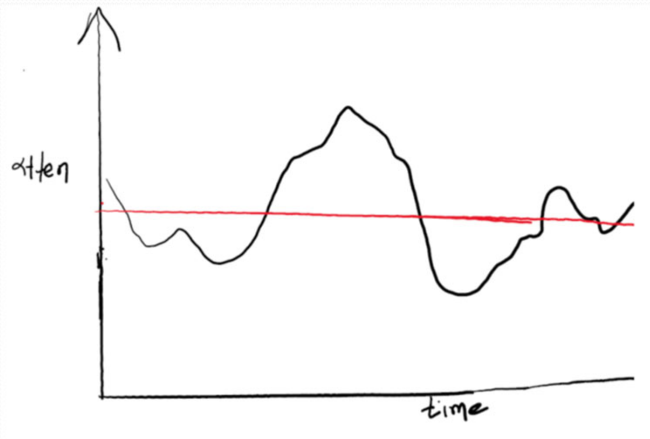
17

Burst Errors

- Occurrence of an error having occurred in the sequence, means bits preceding/following the one in error have higher probability than the average bit error probability to be in error
- Error strings (clusters of errored bits closely located in the sequence) form
- Some channel related impairments producing error bursts
  - impulsive noise
  - “slow” fading/shadowing in wireless (relevance of bit rate to average time/distribution channel attenuation remains below certain level)
  - effect greater at higher data rates

1-18

18



1-19

19

Error detection and control

Objective

- detect and correct errors that occur in the transmission of frames

Types of errors

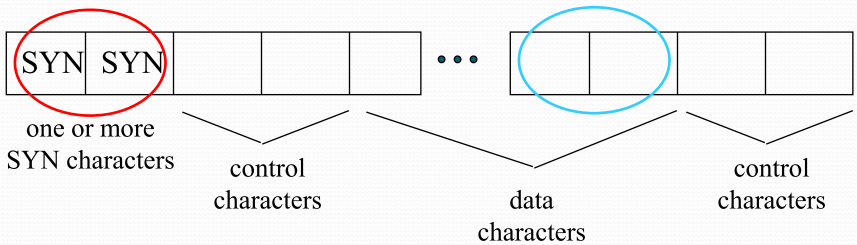
- lost frame: a frame that the receiver does not receive (e.g., because starting of frame/clock extraction is not achieved due to excessive signal attenuation, increased levels of noise…)
- damaged frame: a frame that the receiver receives, but some of its bits are in error

1-20

20

Impact of error location

The frame consists of a sequence of characters



SYN is a unique bit pattern that signals the receiver the beginning of a block

1-21

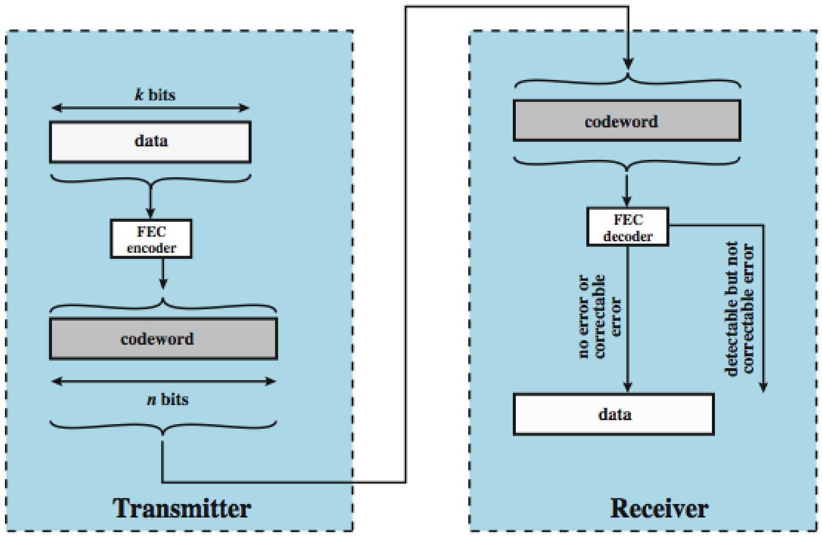
21

Error Detection

1-22

22

Error Correction/Detection Process



1-23

23

How Error Correction & Detection Works

- Adds redundancy to transmitted message
- Can deduce original despite some errors
- Example: block error correction code
  - map *k* bit input onto an *n* bit codeword
  - each distinctly different
  - if get error assume codeword sent was closest to that received
- means have reduced effective data rate
- most of work concerning error correction & detection is making use of Galois field algebra (Boolean algebra - mod_2 arithmetic – is a case of it)

1-24

24

Code rate & minimum distance

$$\text{Code rate } r = \frac{\#\text{ of information bits in a block}}{\#\text{ of total bits in a block}} = \frac{k}{n}$$

The bandwidth expansion is $1/r = n/k$

The energy per channel bit (E_c) is related to energy per information bit (E_b) through $E_c = rE_b$

Minimum distance $(d_{\min})$: Minimum number of positions in which any 2 codewords differ.

1-25

25

Correctable and detectable errors

- A block code can correct at least μ errors if $d_{min} >= 2\mu + 1$
- => if dmin=3, then μ=1. If there is only one error in the block, it can be corrected.
- A block code can detect any error pattern if the received n bits do not correspond to a codeword.
- If there are λ errors in the n-bits codeword, the existence of errors is detected with certainly if $\lambda < d_{min}$.
- However, even when $\lambda >= d_{min}$, many of the corrupted blocks can still be detected.
- Out of the 2^n possible n-bit combinations, only 2^k codewords can be generated, thus, there are $2^n - 2^k = 2^k(2^{(n-k)}-1)$ prohibited combinations.
- Above statement applies when no error correction is used.

1-26

26

A simple block code: (7,4) Hamming Code

Message	Codeword
0000	000 0000
1000	110 1000
0100	011 0100
1100	101 1100
0010	111 0010
1010	001 1010
0110	100 0110
1110	010 1110
0001	101 0001
1001	011 1001
0101	110 0101
1101	000 1101
0011	010 0011
1011	100 1011
0111	001 0111
1111	111 1111

- Minimum Humming distance?
- Error Correction capability?
- Error detection w/o error correction?
- Error detection with error correction?

27

27

A simple block code: (7,4) Hamming Code

Message	Codeword
0000	000 0000
1000	110 1000
0100	011 0100
1100	101 1100
0010	111 0010
1010	001 1010
0110	100 0110
1110	010 1110
0001	101 0001
1001	011 1001
0101	110 0101
1101	000 1101
0011	010 0011
1011	100 1011
0111	001 0111
1111	111 1111

- Minimum Humming distance?
  3
- Error Correction capability?
  1
- Error detection w/o error correction? **3**
- Error detection with error correction? **0**

1-28

28

"Popular" Error Detection Techniques

- Parity Checks
- Longitudinal redundancy checks (LRC)
- Cyclic redundancy checks (CRC)

1-29

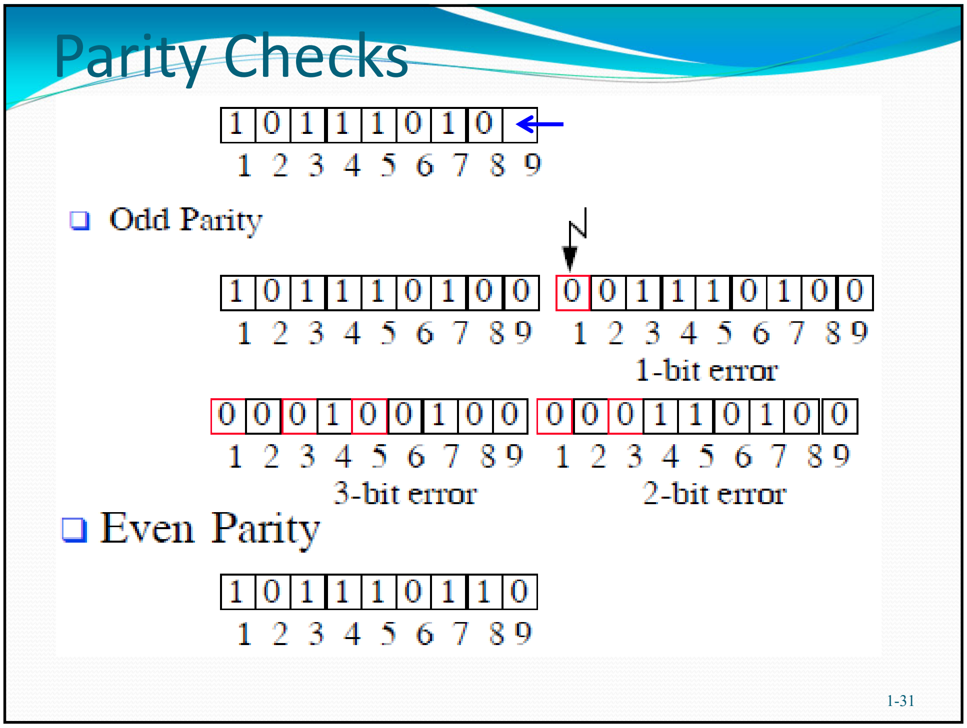
29

Simple Error Detection Scheme

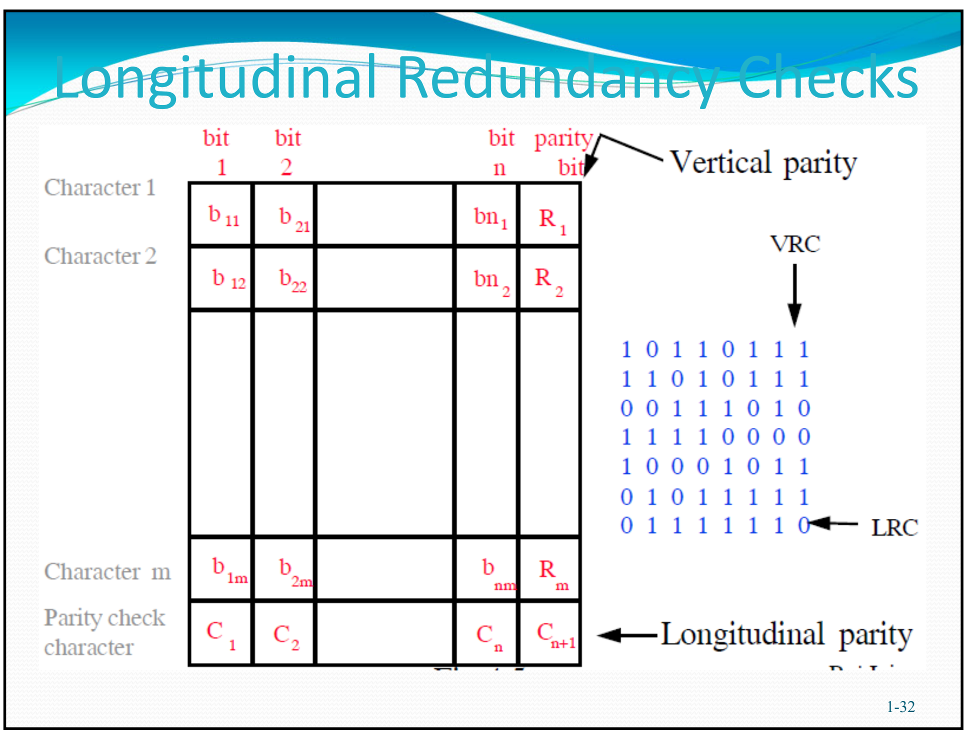
- Parity check
  - Value of parity bit is such that character has even (even parity) or odd (odd parity) number of ones
  - Even number of bit errors goes undetected

1-30

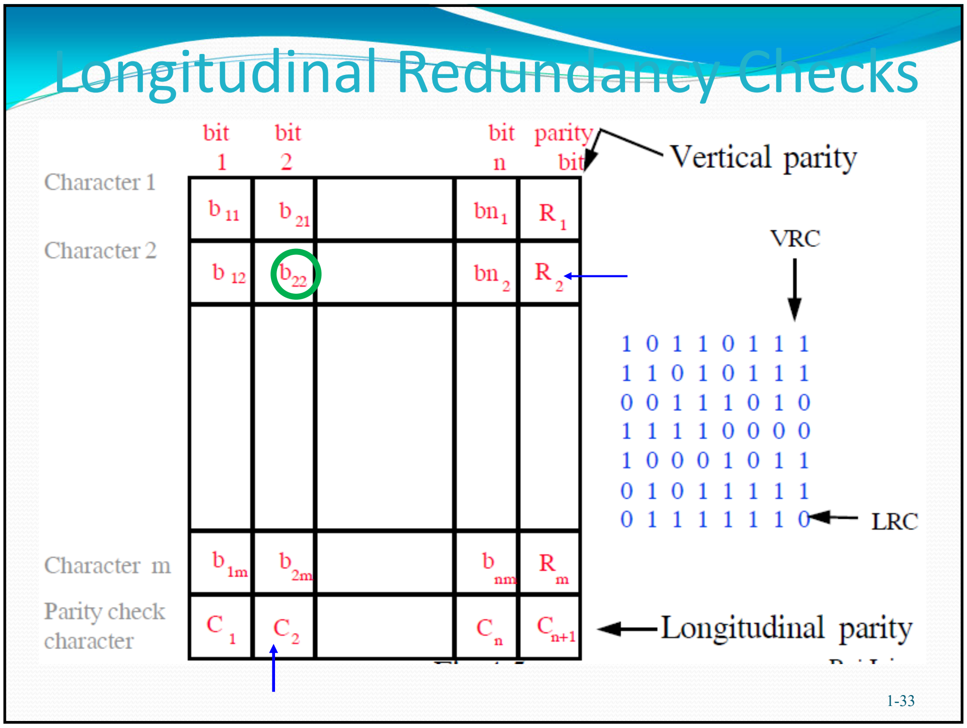
30


Parity Checks
1 0 1 1 1 0 1 0
1 2 3 4 5 6 7 8 9
Odd Parity
1 0 1 1 1 0 1 0 0
1 2 3 4 5 6 7 8 9
0 0 1 1 1 0 1 0 0
1 2 3 4 5 6 7 8 9
1-bit error
0 0 0 1 0 0 1 0 0
1 2 3 4 5 6 7 8 9
3-bit error
0 0 0 1 1 0 1 0 0
1 2 3 4 5 6 7 8 9
2-bit error
Even Parity
1 0 1 1 1 0 1 1 0
1 2 3 4 5 6 7 8 9
1-31

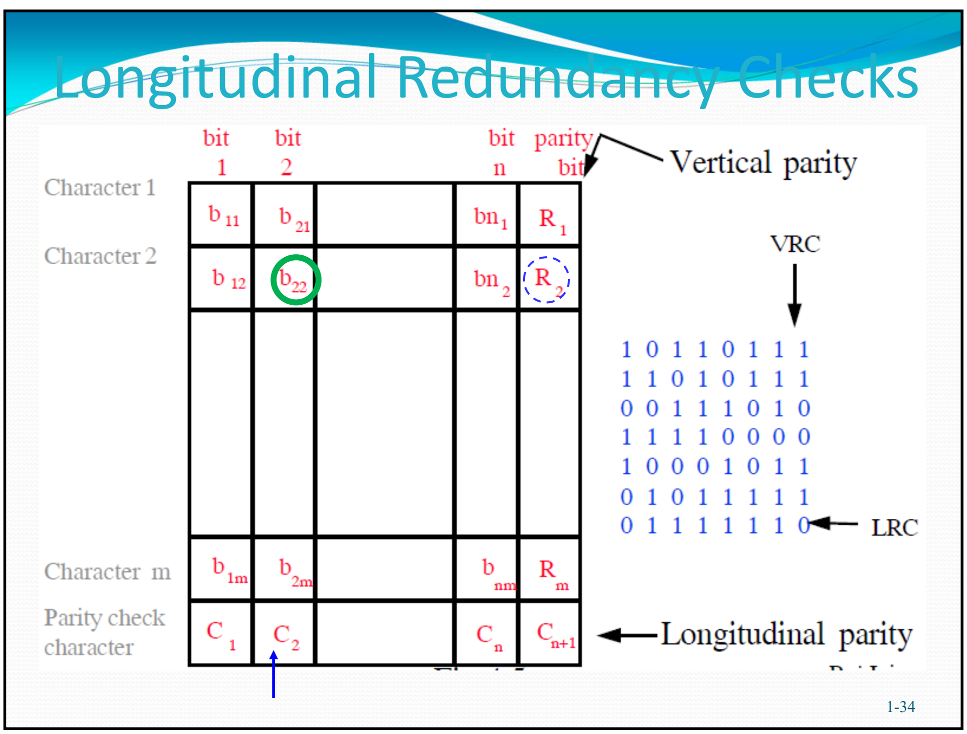
31


Longitudinal Redundancy Checks
bit 1
bit 2
bit n
parity bit
Vertical parity
Character 1
Character 2
Character m
Parity check character
b 11
b 21
bn 1
R 1
b 12
b 22
bn 2
R 2
b 1m
b 2m
b nm
R m
C 1
C 2
C n
C n+1
VRC
1 0 1 1 0 1 1 1
1 1 0 1 0 1 1 1
0 0 1 1 1 0 1 0
1 1 1 1 0 0 0 0
1 0 0 0 1 0 1 1
0 1 0 1 1 1 1 1
0 1 1 1 1 1 1 0
LRC
Longitudinal parity
1-32

32

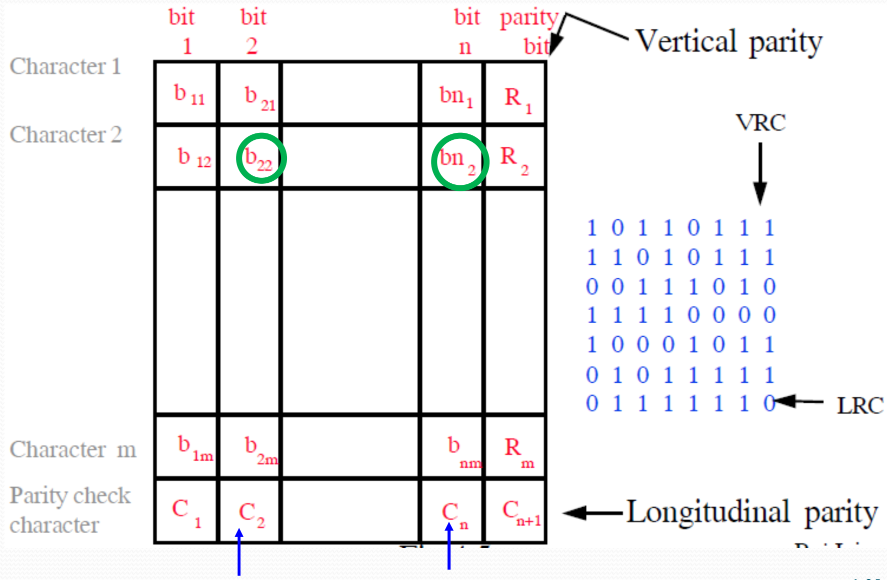

Longitudinal Redundancy Checks
bit 1
bit 2
bit n
parity bit
Vertical parity
Character 1
Character 2
Character m
Parity check character
b 11
b 21
bn 1
R 1
b 12
b 22
bn 2
R 2
b 1m
b 2m
b nm
R m
C 1
C 2
C n
C n+1
VRC
1 0 1 1 0 1 1 1
1 1 0 1 0 1 1 1
0 0 1 1 1 0 1 0
1 1 1 1 0 0 0 0
1 0 0 0 1 0 1 1
0 1 0 1 1 1 1 1
0 1 1 1 1 1 1 0
LRC
Longitudinal parity
1-33

33


Longitudinal Redundancy Checks
bit 1
bit 2
bit n
parity bit
Vertical parity
Character 1
Character 2
Character m
Parity check character
b 11
b 21
bn 1
R 1
b 12
b 22
bn 2
R 2
b 1m
b 2m
b nm
R m
C 1
C 2
C n
C n+1
VRC
1 0 1 1 0 1 1 1
1 1 0 1 0 1 1 1
0 0 1 1 1 0 1 0
1 1 1 1 0 0 0 0
1 0 0 0 1 0 1 1
0 1 0 1 1 1 1 1
0 1 1 1 1 1 1 0
LRC
Longitudinal parity
1-34

34

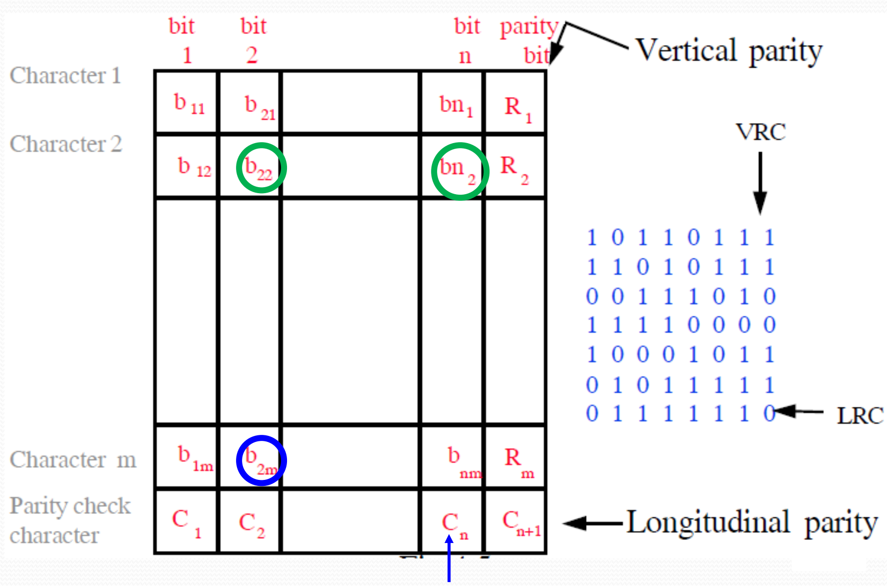
Longitudinal Redundancy Checks



1-35

35

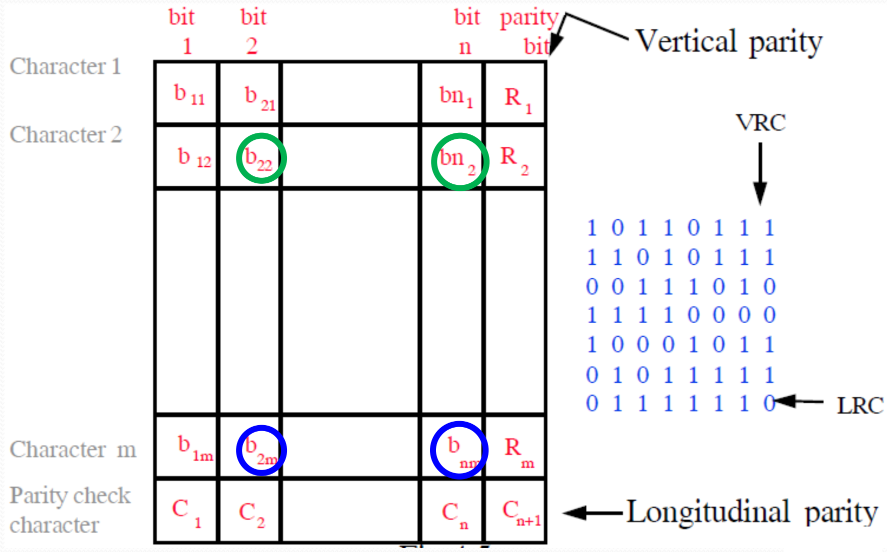
Longitudinal Redundancy Checks



36

36

Longitudinal Redundancy Checks

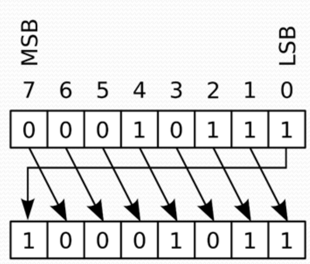


1-37

37

Cyclic Redundancy Check

- Based on cyclic error-correcting codes



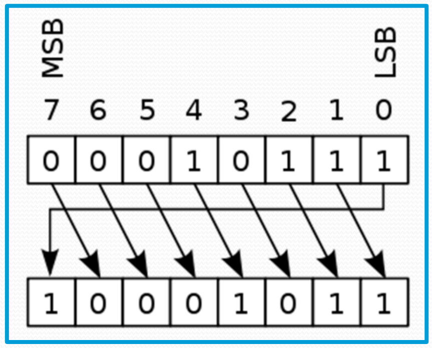
- For a block of k bits the transmitter generates n bit sequence
- n insert redundancy in the codeword
- Transmitter transmits the $k+n$ bits
- Receiver uses error detection process to decide if there were errors in the received sequence or otherwise

1-38

38

Cyclic Codes

- **“Cyclic code** is a block code, where the circular shifts of each codeword gives another codeword that belongs to the code”.
- “They are error-correcting codes having algebraic properties that are convenient for efficient error correction & detection”.



1-39

39

Fundamentals of CRC coding

- CRC codes treat bit strings as representations of polynomials with coefficients of 0 and 1 only (modulo 2 arithmetic)

$$11001 \leftrightarrow 1 * X^4 + 1 * X^3 + 0 * X^2 + 0 * X^1 + 1 * X^0 = X^4 + X^3 + 1$$

- Polynomial arithmetic is done modulo 2
  - subtraction and addition are similar to EXCLUSIVE OR
  - division is similar to the one in decimal except the subtraction is done modulo 2
- Make sure you are familiar with mod2 arithmetic/algebra

1-40

40

Addition/Subtraction

Modulo 2 addition/subtraction is performed using an exclusive OR (xor) operation on the corresponding binary digits of each operand.

$0 \pm 0 = 0;\quad 0 \pm 1 = 1;\quad 1 \pm 0 = 1;\quad 1 \pm 1 = 0$

Multiplication

```
   1011
 x 0101
 ________
   1011
  0000
 1011
0000
_________
0100111
```

1-41

41

Division

Modulo 2 division can be performed in a manner similar to arithmetic long division. Subtract the denominator (the bottom number) from the leading parts of the enumerator (the top number). Proceed along the enumerator until its end is reached. Remember that we are using modulo 2 subtraction. For example, we can divide 100100110 by 10011 as follows:

```
          10001 remainder 101
10011|100100110
      10011
          10110
          10011
            101
```

1-42

42

CRC: Basic Idea

- The sender and receiver agree upon "a generator polynomial", **G(x)**, in advance.
- The sender appends a checksum (corresponds to the *n* redundancy bits) to the end of the (only data) frame, represented by the **M(x)** polynomial, in a way that the polynomial **T(x),** representing the {data + checksum bits} frame, is divisible by **G(x).**
- Upon receipt of the frame, the receiver (generates and) divides **H(x)** by **G(x)** using modulo 2 division.
- **H(x)** is the polynomial corresponding to the received sequence.
- if there is a remainder, there has been transmission error.

1-43

43

How to compute the checksum

$11001 \leftrightarrow 1 * X^4 + 1 * X^3 + 0 * X^2 + 0 * X^1 + 1 * X^0 = X^4 + X^3 + 1 = M(X)$

- If *n* is the degree of G(x), then append *n* zero to the low order end of the frame; the resulting frame corresponds to the polynomial $X^n M(x)$.

$11001000 \xleftrightarrow{n=3} 1 * X^7 + 1 * X^6 + 0 * X^5 + 0 * X^4 + 1 * X^3 + 0 * X^2 + 0 * X^1 + 0 * X^0 = X^7 + X^6 + X^3 = X^3 * M(X)$

- Divide G(x) into $X^n M(x)$ using modulo 2 division.

$$\frac{X^n M(X)}{G(X)} \rightarrow D(X); R(X)$$

- *D(X):* divisor; *R(X)*: remainder

$$X^n M(X) = G(X) \otimes D(X) \oplus R(X)$$

1-44

44

How to compute the checksum

- Subtract the remainder from $X^n M(x)$ using modulo 2 subtraction/addition.
- The result is the checksumed frame's polynomial , T(x).

$$T(x) = X^n M(X) \oplus R(X)$$

- The frame corresponding to ***T(x)*** is transmitted.

$$T(X) = X^n M(X) \oplus R(x) = [D(X) \otimes G(X) \oplus \boxed{R(x)] \oplus R(X)} \Leftrightarrow$$

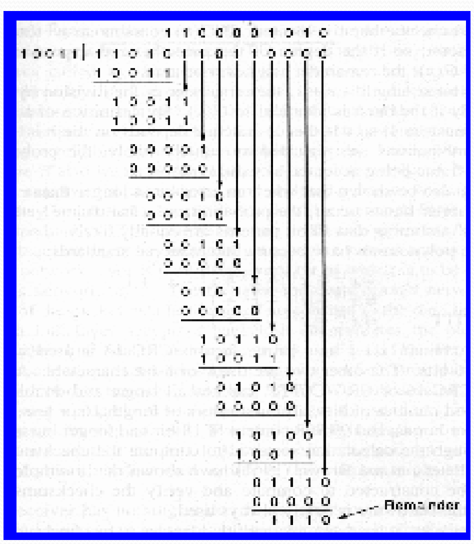
$$\Leftrightarrow T(X) = D(X) \otimes G(X) \oplus O \Leftrightarrow \boxed{\frac{T(X)}{G(X)} = D(X)}$$

G(X) divides perfectly T(X) (remainder = *O*)

1-45

45

CRC: an Example



- Frame: 1101011011
- Generator: 10011

1-46

46

CRC Error Detection

- Let us assume that some transmission errors occur
- Instead of receiving **T(x)**, the receiver will receive **H(x)=T(x)⊕E(x)**
- If there are k “1” bits in E(x), (it is most probable that) k bit errors have occurred
- the receiver computes (T(x) ⊕ E(x))/G(x)=E(x)/G(x)
- If G(x) contains two or more terms, (i.e. n>1) all single errors will be detected.
- single-bit error means $E(X) = X^{m-1}$, where $0 < m \leq n + k$

1-47

47

Single error and $G(x)=X^n$

- $H(X) = T(X) \oplus E(X)$ where
- $E(X) = O$ if no bit errors occur
- $E(X) = X^{m-1}$ if only the m-th bit of the *[k+n]*-bit long frame is reversed ($0 < m \leq n + k$; the larger the value of m is, the more significant the location of the bit within the frame is)
- $E(X) = L(X) \otimes G(X) \oplus F(X)$
- $H(X) = T(X) \oplus E(X) = T(X) \oplus L(X) \otimes G(X) \oplus F(X)$
- $\dfrac{H(X)}{G(X)} = D(X) \oplus L(X) \oplus \dfrac{F(X)}{G(X)}$

Error will be detected if $\dfrac{F(X)}{G(X)} \neq O$

For *m-1>=n*, $L(X) = X^{m-n-1}$ and $\dfrac{F(X)}{G(X)} = O$. Error is not detected.

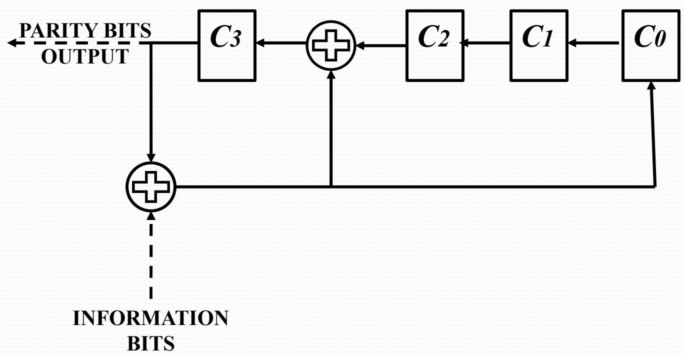
For *m-1<n*, *L(x)=O* and $\dfrac{F(X)}{G(X)} \neq O$. Error is detected.

1-48

48

Shift register circuit

$11001 \leftrightarrow 1 * X^4 + 1 * X^3 + 0 * X^2 + 0 * X^1 + 1 * X^0 =$
$X^4 + X^3 + 1 = G(X)$

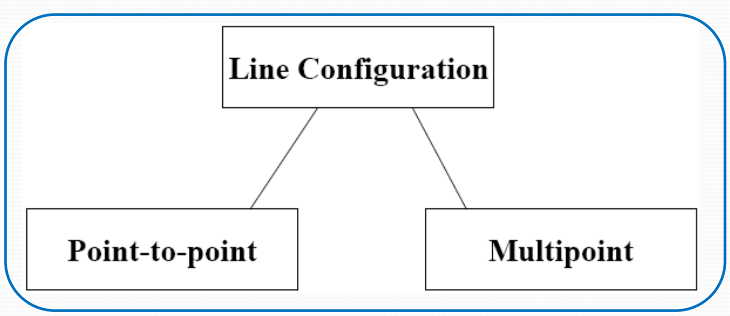


1-49

49

Line Configuration

- Topology
  - refers to physical arrangement of stations on the medium
  - two topologies are commonly used
    - point-to-point
    - multipoint



1-50

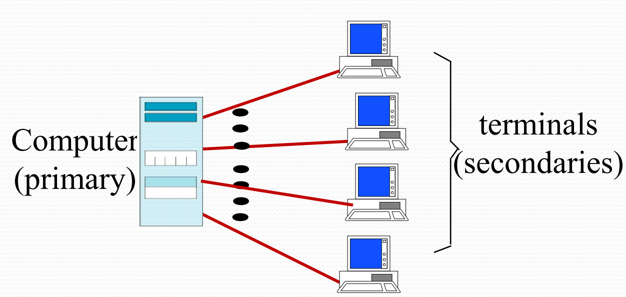
50

Line Configuration - Topology

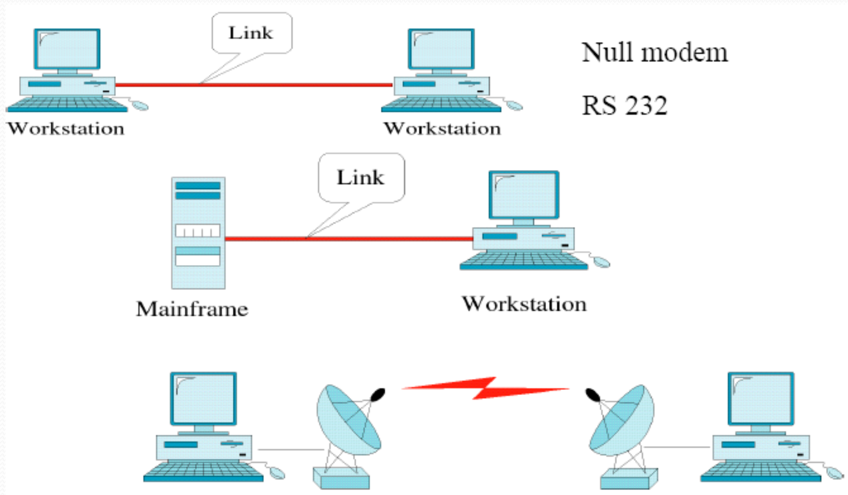
- point to point - two stations
  - traditionally mainframe computer and terminals
  - between two routers / computers
- multi point - multiple stations
  - typically, a local area network (LAN)

Point-to-point topology

- a separate transmission line from the computer to each terminal
- the computer must have an I/O port for each terminal



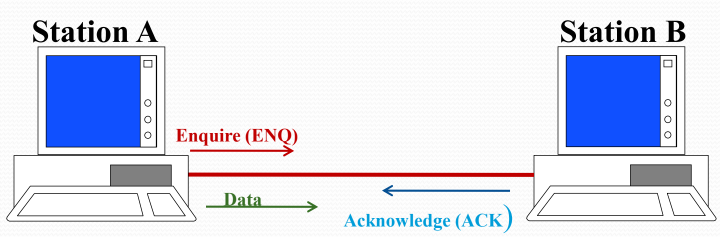
Examples of Point-to-point topology



1-53

53

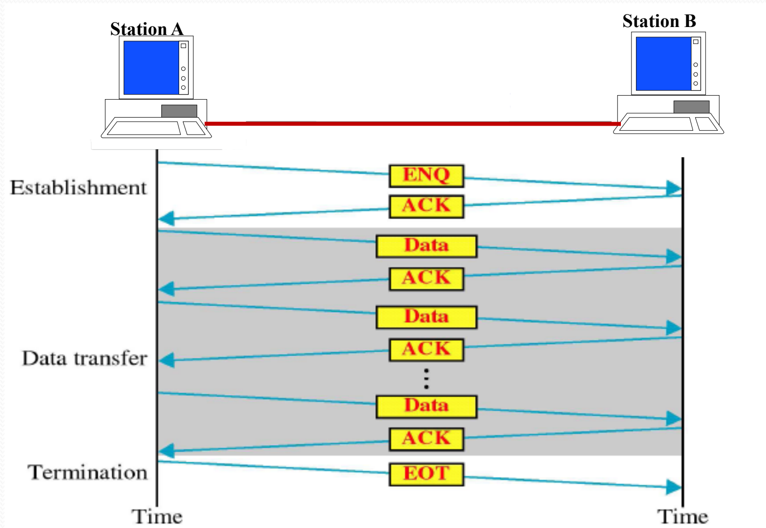
Line Discipline: Point-to-Point



1-54

54

Line Discipline: Point-to-Point

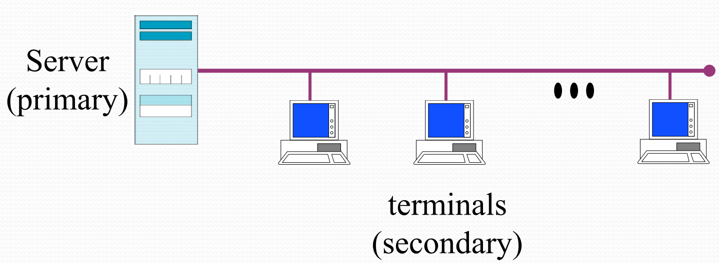


1-55

55

Point to Multipoint topology

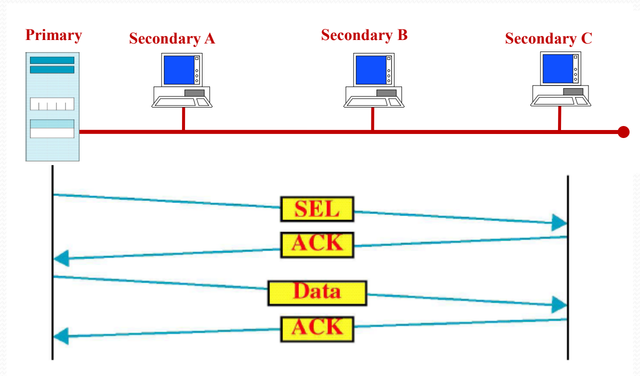
- A single "line" is needed
- The computer needs only a single I/O port
  - e.g. Ethernet, Token Ring, WiFi



1-56

56

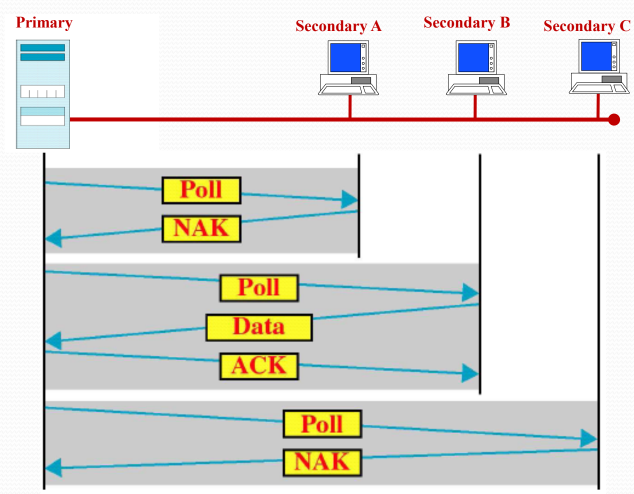
Select Line Discipline



1-57

57

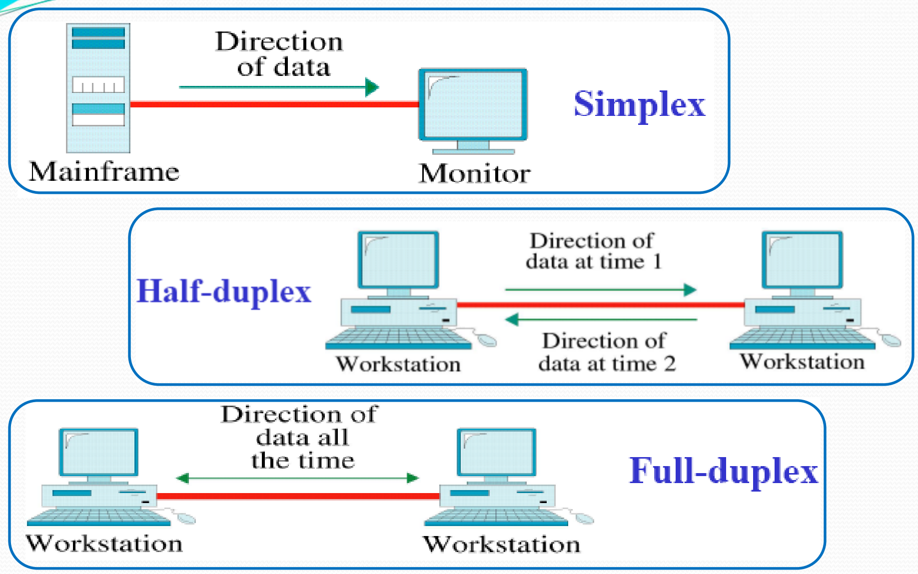
Poll Line Discipline



1-58

58

Transmission Mode



1-59

59

Line Configuration - Duplex

- classify data exchange as half or full duplex
- half duplex (two-way alternate)
  - only one station may transmit at a time
  - requires one data path
- full duplex (two-way simultaneous)
  - simultaneous transmission and reception between two stations
  - requires two data paths
    - separate media or frequencies used for each direction
  - or echo canceling

1-60

60

END

1-61

61