Chapter 6 — Digital Data Communication Techniques

CSE 3213

Asynchronous and Synchronous Transmission

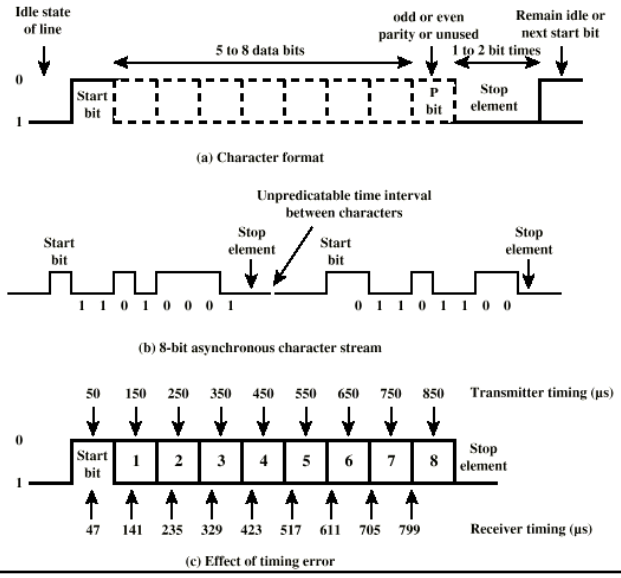
- Timing problems require a mechanism to synchronize the transmitter and receiver.
- Receiver samples the medium at the center of each bit time.
- Transmitter's and receiver's clocks may not be precisely aligned.
- Example (discussed in class)
- Two solutions
  - —Asynchronous
  - —Synchronous

Asynchronous Transmission

- Data transmitted on character at a time
  - —5 to 8 bits
- Timing only needs maintaining within each character
- Resynchronize with each character

3

Asynchronous (diagram)



4

Asynchronous - Behavior

- Idle state = binary 1
- In idle state, receiver looks for transition 1 to 0
- Start bit = binary 0
- Then samples next 5 – 8 intervals (character length)
- Stop element = binary 1 (min length = 1, 1.5 or 2 bits)
  No maximum length specified for stop element (why?)
- Then looks for next 1 to 0 for next char

- Simple, cheap
- Overhead of 2 or 3 bits per char (~20%)
- Good for data with large gaps (keyboard)

5

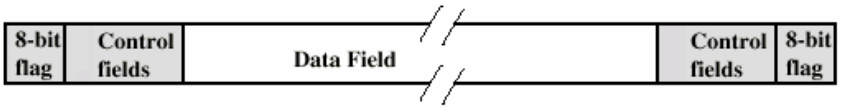
Synchronous - Bit Level

- Block of data transmitted without start or stop bits
- Clocks must be synchronized
- Can use separate clock line
  - —Good over short distances
  - —Subject to impairments
- Embed clock signal in data
  - —Manchester encoding (digital signals)
  - —Carrier frequency, phase (analog signals)

6

Synchronous - Block Level

- Need to indicate start and end of block
- Use preamble and postamble
  - —e.g. series of SYN (hex 16) characters
  - —e.g. block of 11111111 patterns ending in 11111110
- More efficient (lower overhead) than asynchronous transmission



7

Types of Error

- An error occurs when a bit is altered between transmission and reception
- Single bit errors
  - — One bit altered
  - — Adjacent bits not affected
  - — White noise
- Burst errors
  - — Contiguous sequence of *B* bits in which the first, the last and any number of intermediate bits in error
  - — Caused by impulse noise or fading in wireless channels
  - — Effect greater at higher data rates

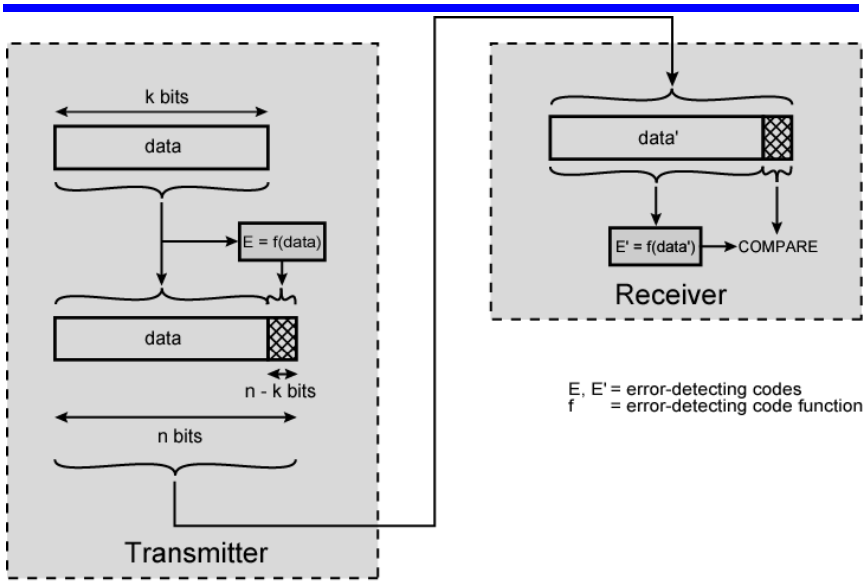
8

Error Detection

- Additional bits (error detection code) added by transmitter for error detection
- Recalculated and checked by receiver
- Note: there is still chance of undetected errors
- Parity Check
  - —Appending a parity bit to the end of a data block
  - —Value of parity bit is such that character has even (even parity) or odd (odd parity) number of ones
  - —An even number of bit errors goes undetected

9

Error Detection Process



Cyclic Redundancy Check

- Given *k* bits of data,
  generate a sequence F of *j* bits (FCS)
  using a predetermined divisor P of (*j*+1) bits
- Transmit a frame of *k+j* bits (data + FCS) which will be exactly divisible by divisor P
- Receiver divides frame by divisor P
  - —If no remainder, assume no error
- Simpler but equivalent method: receiver repeats the steps the sender did. If getting the same FCS, assume no error.

11

CRC

3 equivalent procedures:

- Modulo-2 arithmetic
- Polynomials
- Digital logic (not covered)

Examples (discussed in class)

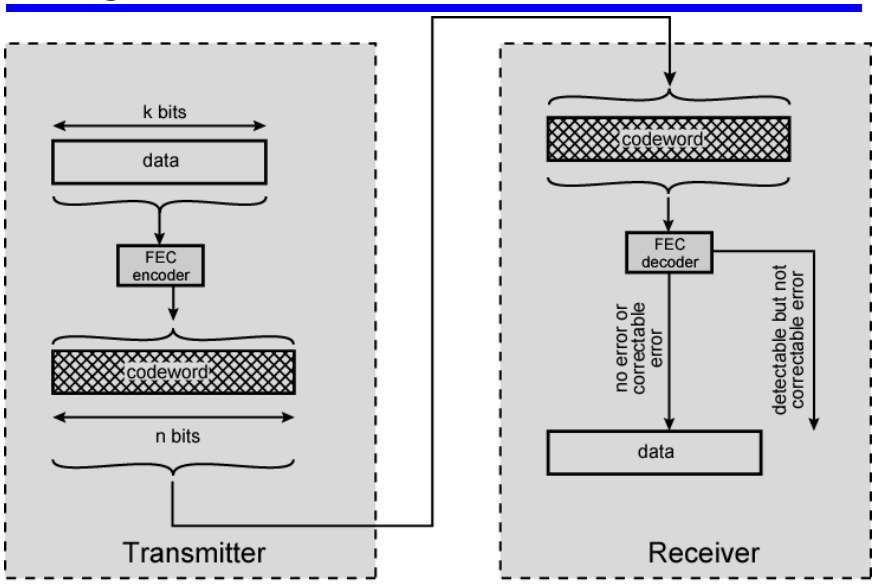
12

Error Correction

- Correction of detected errors usually requires data block to be retransmitted (see Chapter 7)
- Not appropriate for wireless applications
  - —Bit error rate is high
    - Lots of retransmissions
  - —Propagation delay can be long (satellite) compared with frame transmission time
    - Would result in retransmission of frame in error plus many subsequent frames
- Need to correct errors on basis of bits received

13

Error Correction Process Diagram



Error Correction Process

- Each *k* bit block mapped to an *n* bit block (*n*>*k*)
  - Codeword
  - Forward error correction (FEC) encoder
- Codeword sent
- Received bit string similar to transmitted but may contain errors
- Received code word passed to FEC decoder
  - If no errors, original data block output
  - Some error patterns can be detected and corrected
  - Some error patterns can be detected but not corrected
  - Some (rare) error patterns are not detected
    - Results in incorrect data output from FEC

15

How Error Correction Works

- Add redundancy to transmitted message
- Can deduce original in face of certain level of error rate
- E.g. block error correction code
  - In general, add (*n* – *k*) bits to end of block
    - Gives *n* bit block (codeword)
    - All of original *k* bits included in codeword
  - Some FEC map *k* bit input onto *n* bit codeword such that original *k* bits do not appear

16

Design Considerations for Block Code

- For given values of n and k, want the largest possible value of d_{min}
- To increase d_{min} increase the number of extra bits.
- Reduce the number of extra bits to reduce bandwidth needed
- Easy to encode/decode, minimal overheads (memory, time)

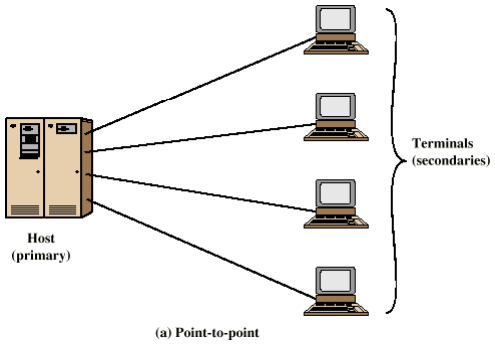
17

Line Configuration

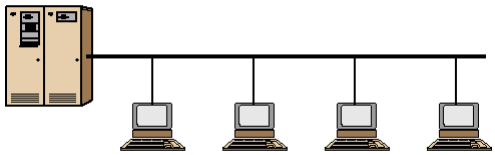
- Topology
  - Physical arrangement of stations on medium
  - Point to point
  - Multi point
    - Computer and terminals, local area network
- Half duplex
  - Only one station may transmit at a time
  - Requires one data path
- Full duplex
  - Simultaneous transmission and reception between two stations
  - Requires two data paths (or echo canceling)

18

Traditional Configurations



(a) Point-to-point



(b) Multipoint

19

Reading

- Chapter 6, Stallings' book
- Homework: Solve problems 6.13 and 6.15 in the textbook.

20