

FRAMING

SUBJECT: DATA COMMUNICATION & OPTICAL
FIBERS

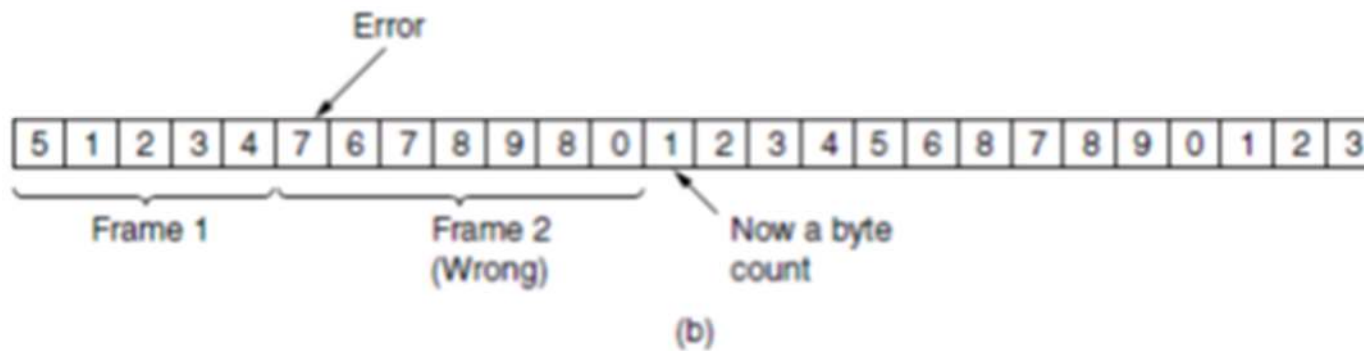
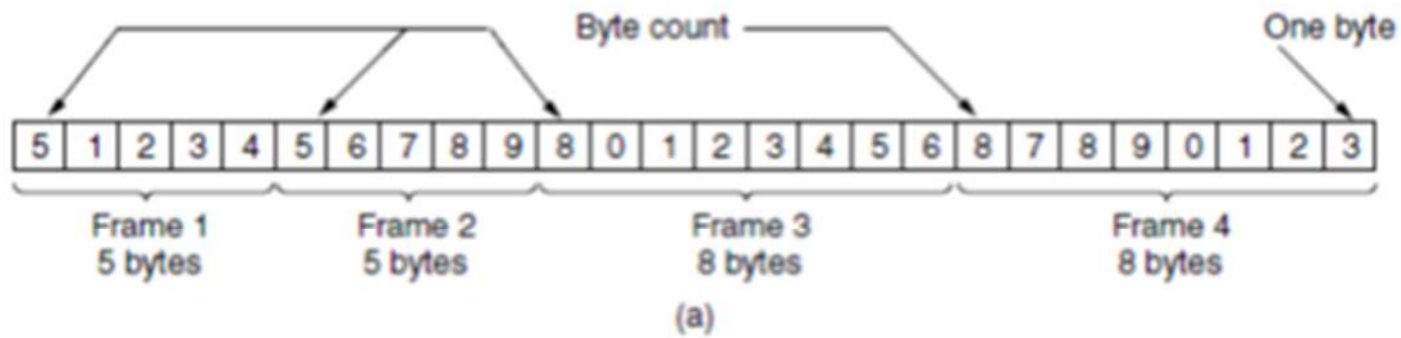
SAVIYA VARGHESE
Dept of COMPUTER APPLICATION
2020-21

BYTE COUNT:

- ⦿ The first **framing** method uses a field in the header to specify the number of bytes in the frame.
- ⦿ When the data link layer at the destination sees the **byte count**, it knows how many bytes follow and hence where the end of the frame is.



FRAMES OF SIZES 5,5,8,8(FIGURE(a)-WITHOUT ERRORS FIGURTE(b)-WITH ONE ERROR



Errors

- ⦿ Problem shown in (b).
- ⦿ Transmission error changed 5 to 7. All frames now out of synch.
- ⦿ Even if we detect error, we have no way of recovering - of finding where next frame starts.



- ⦿ A **checksum** is a small-sized block of data derived from another block of digital data for the purpose of detecting errors that may have been introduced during its transmission or storage.
- ⦿ By themselves, checksums are often used to verify data integrity but are not relied upon to verify data authenticity.



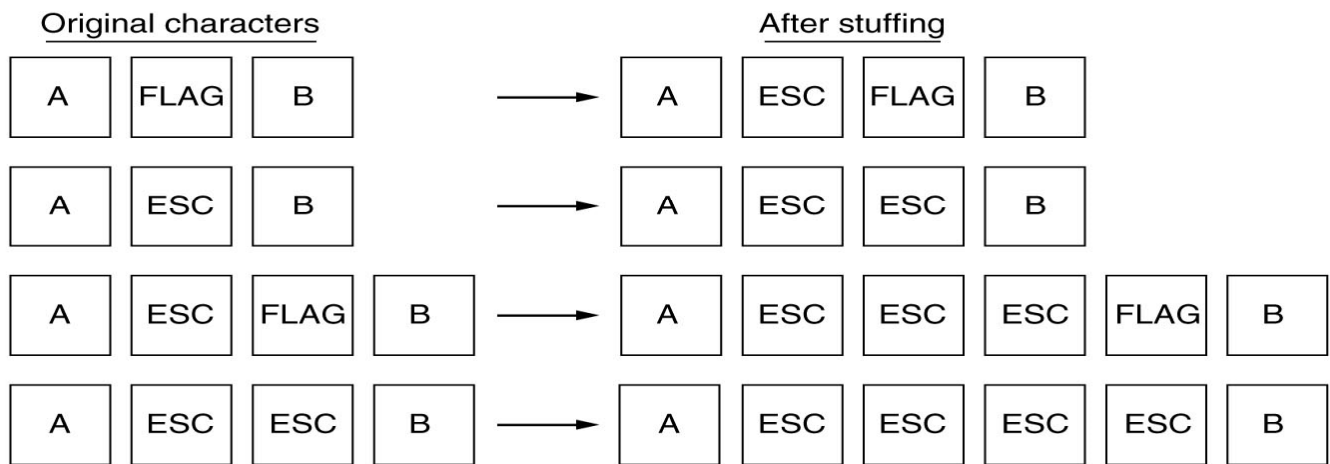
FLAG BYTES WITH BYTE STUFFING

- Each frame starts with special start and end bytes (**flag bytes**). Here will imagine it as same byte, FLAG.
After error, can always find start of next frame.





(a)



(b)



FLAG BITS WITH BIT STUFFING

- ◉ Byte stuffing specifies char format (e.g. 8 bits per char).
To allow arbitrary no. of bits per char, use stuffing at *bit*-level rather than at byte-level.
- ◉ Each frame begins and ends with bit pattern 01111110 (6 1's)
If 5 1's in a row in data, stuff a 0 in so will never be 6 in a row.
Stuff it in always - *whether the next char was going to be a 1 or not.*
De-stuffer removes the 0's after any 5 1's.

