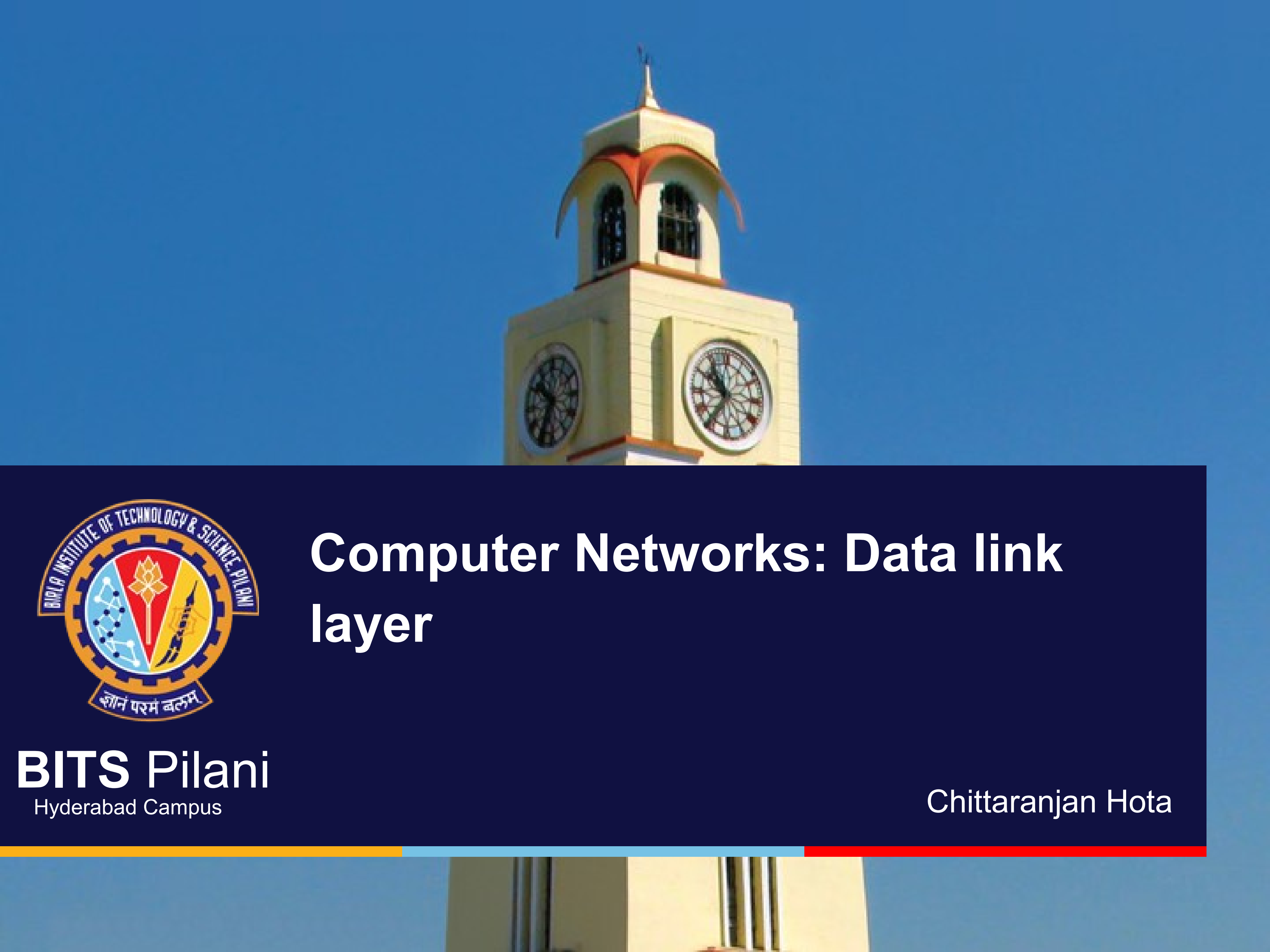


Computer Networks: Data link layer

BITS Pilani
Hyderabad Campus

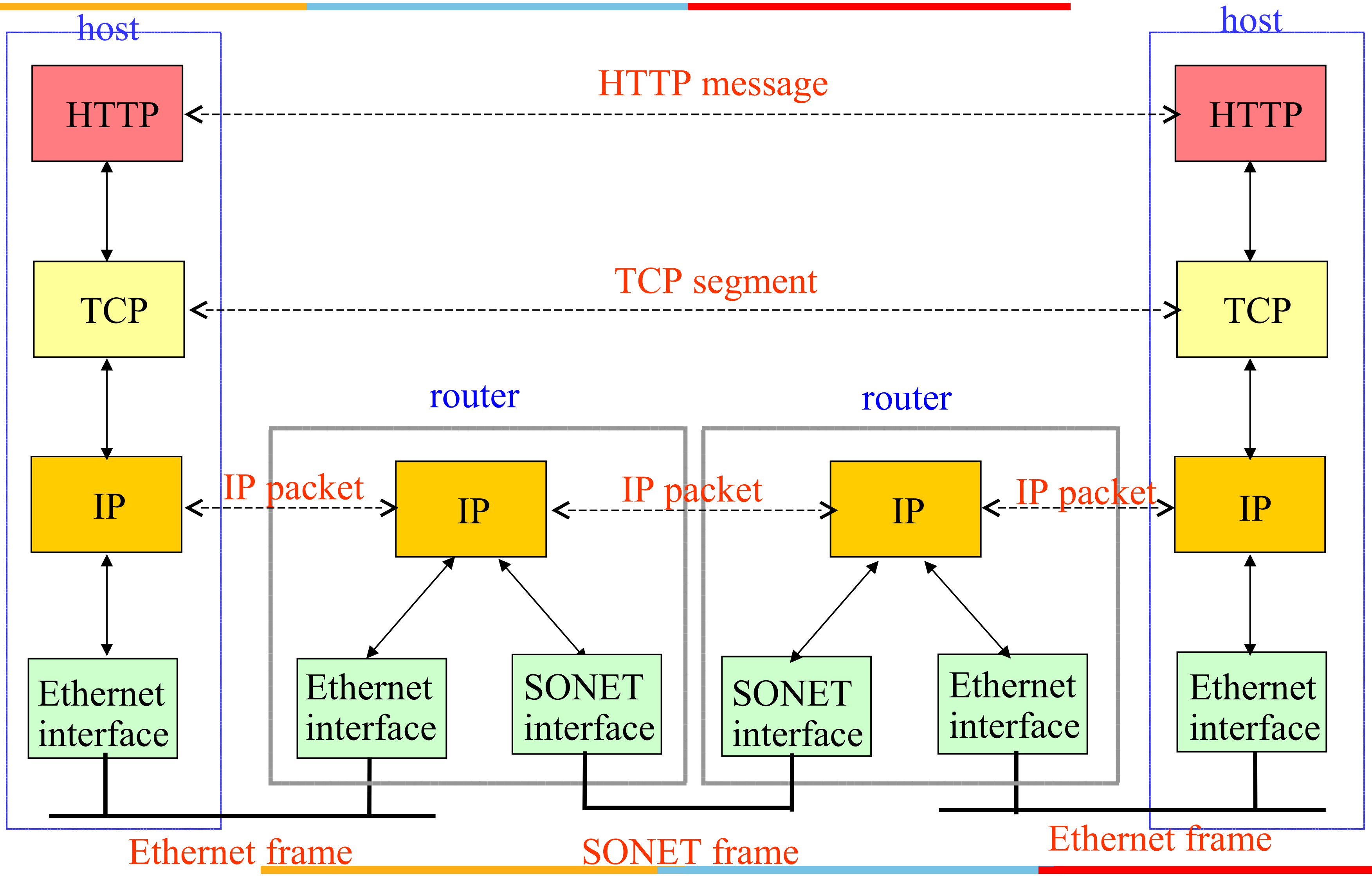
Chittaranjan Hota

Link layer Context

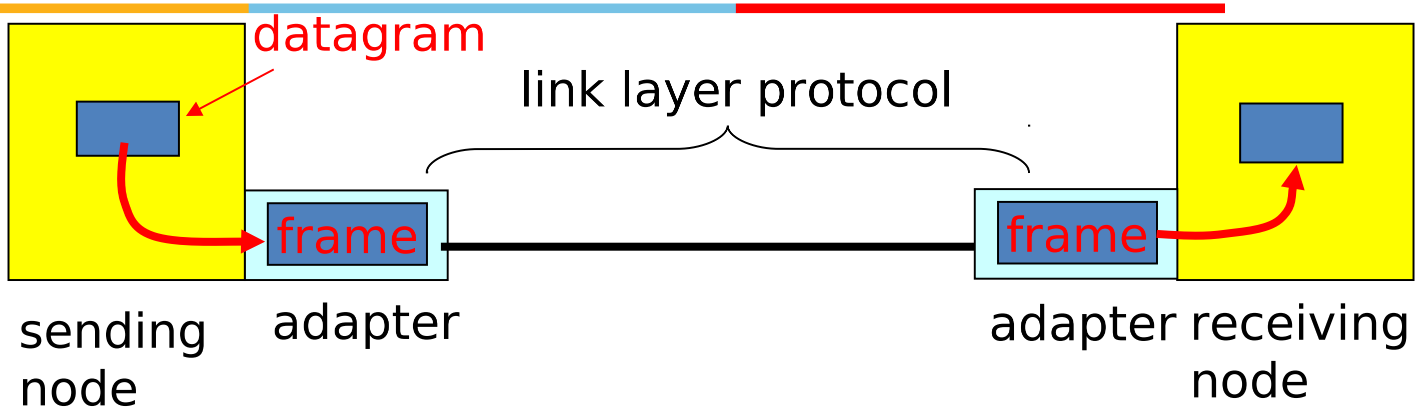
- Datagram transferred by different link protocols over different links.
  - e.g., Ethernet on first link, frame relay on second, and 802.11 on third etc.
- Each link protocol provides different services
  - e.g., may or may not provide reliable data transfer over link

- transportation analogy
- trip from Delhi to Bangalore
  - air: Delhi to Hyderabad
  - train: Hyderabad to Chennai
  - air: Chennai to Bangalore
- tourist = datagram
- transport segment = communication link
- transportation mode = link layer protocol
- travel agent = routing algorithm

Data link layer



Where is link layer implemented?

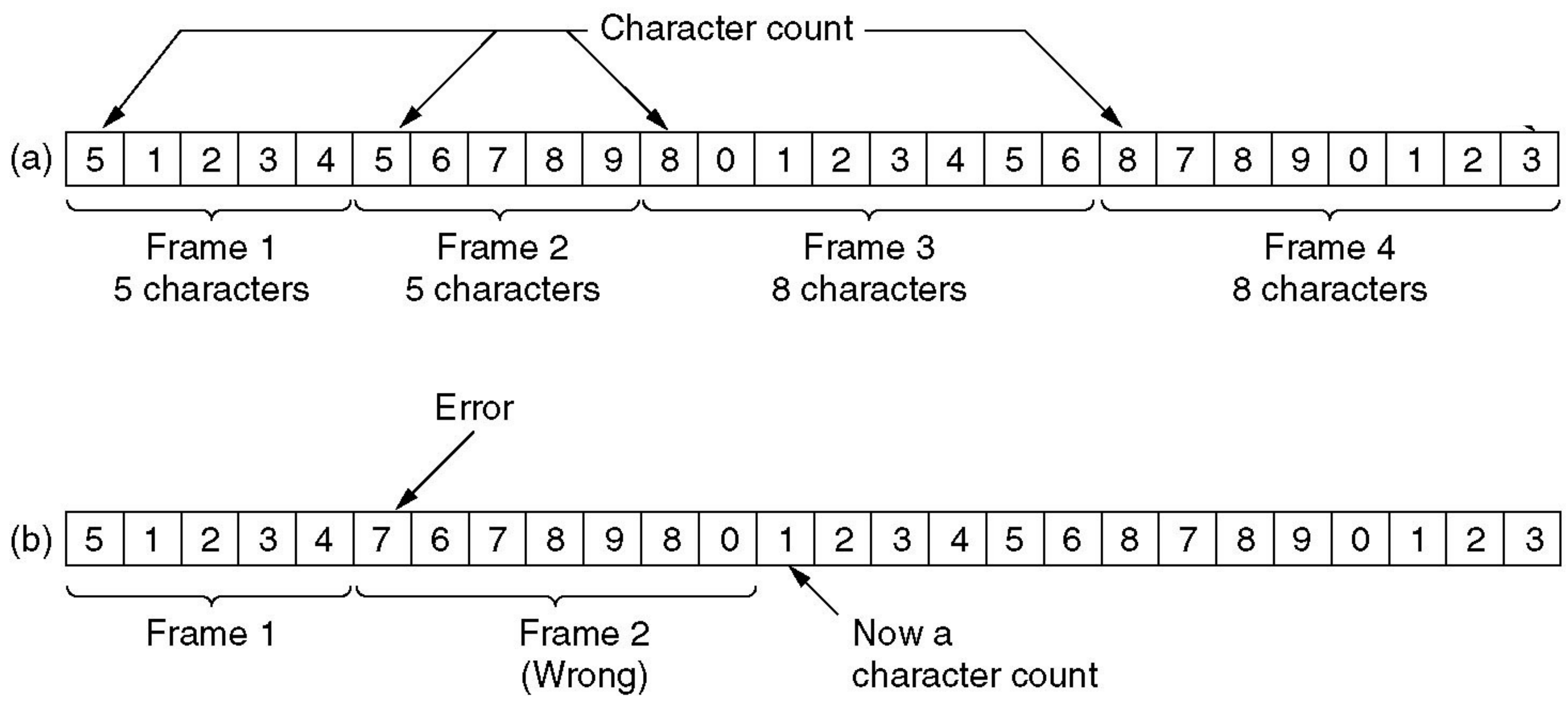


- Link layer implemented in adaptor (network interface card)
  - Ethernet card, PCMCI card, 802.11 card
- Sending side:
  - Encapsulates datagram in a frame
  - Adds error checking bits, flow control, etc.
- Receiving side
  - Looks for errors, flow control, etc.
  - Extracts datagram and passes to receiving node

Link layer services

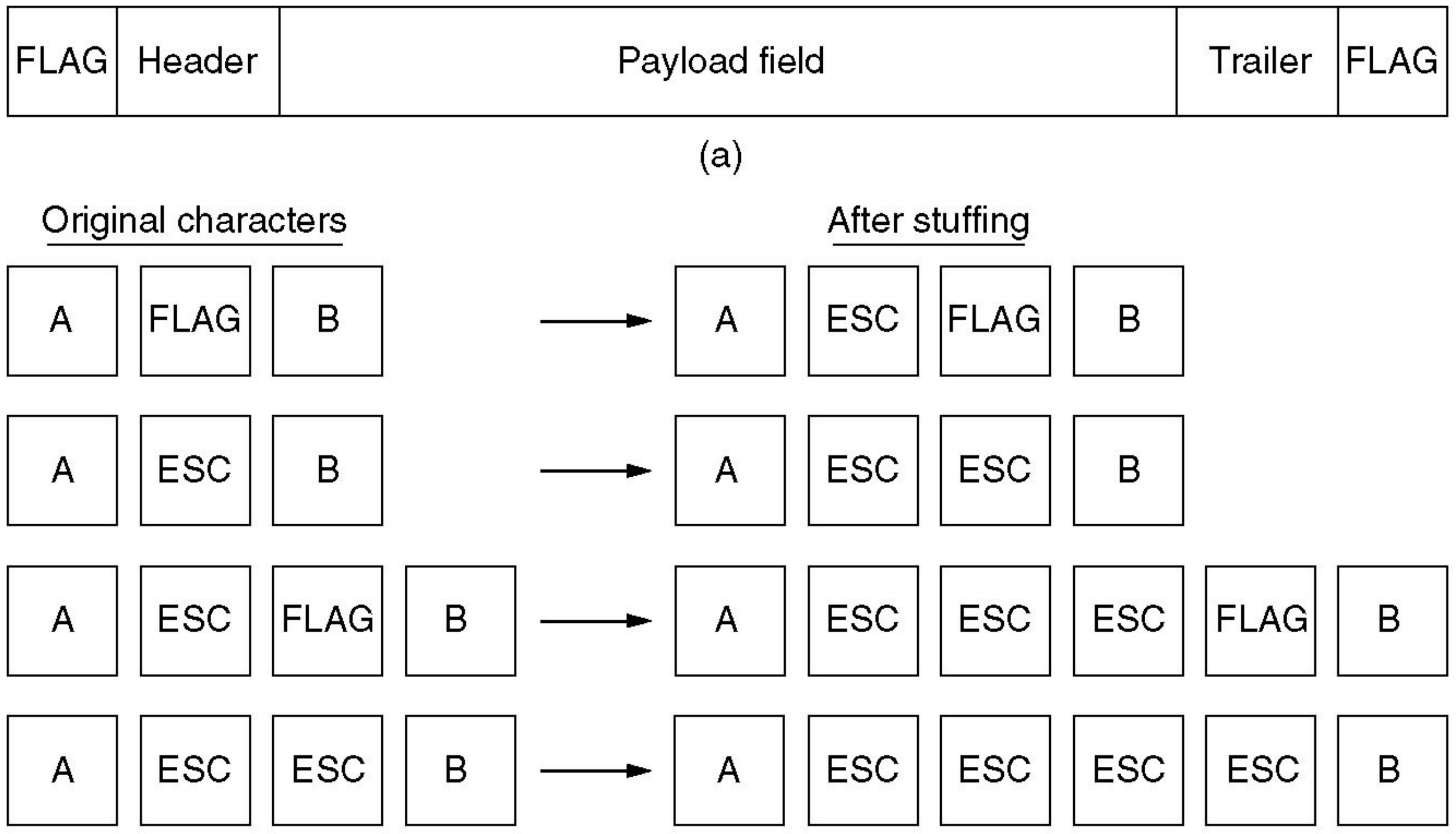
- Reliable delivery and flow control
- Framing
- Error detection and correction
- Medium access

Framing: Character count



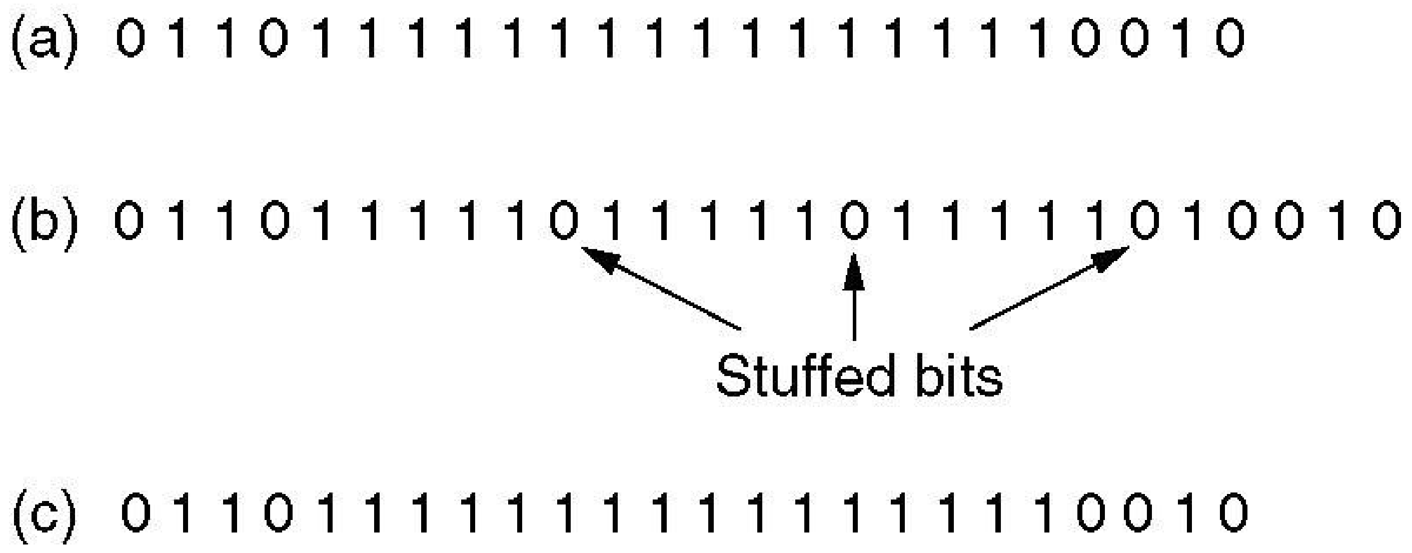
A character stream. (a) Without errors. (b) With one error.

Framing: Byte Stuffing



(b)

Framing: bit stuffing

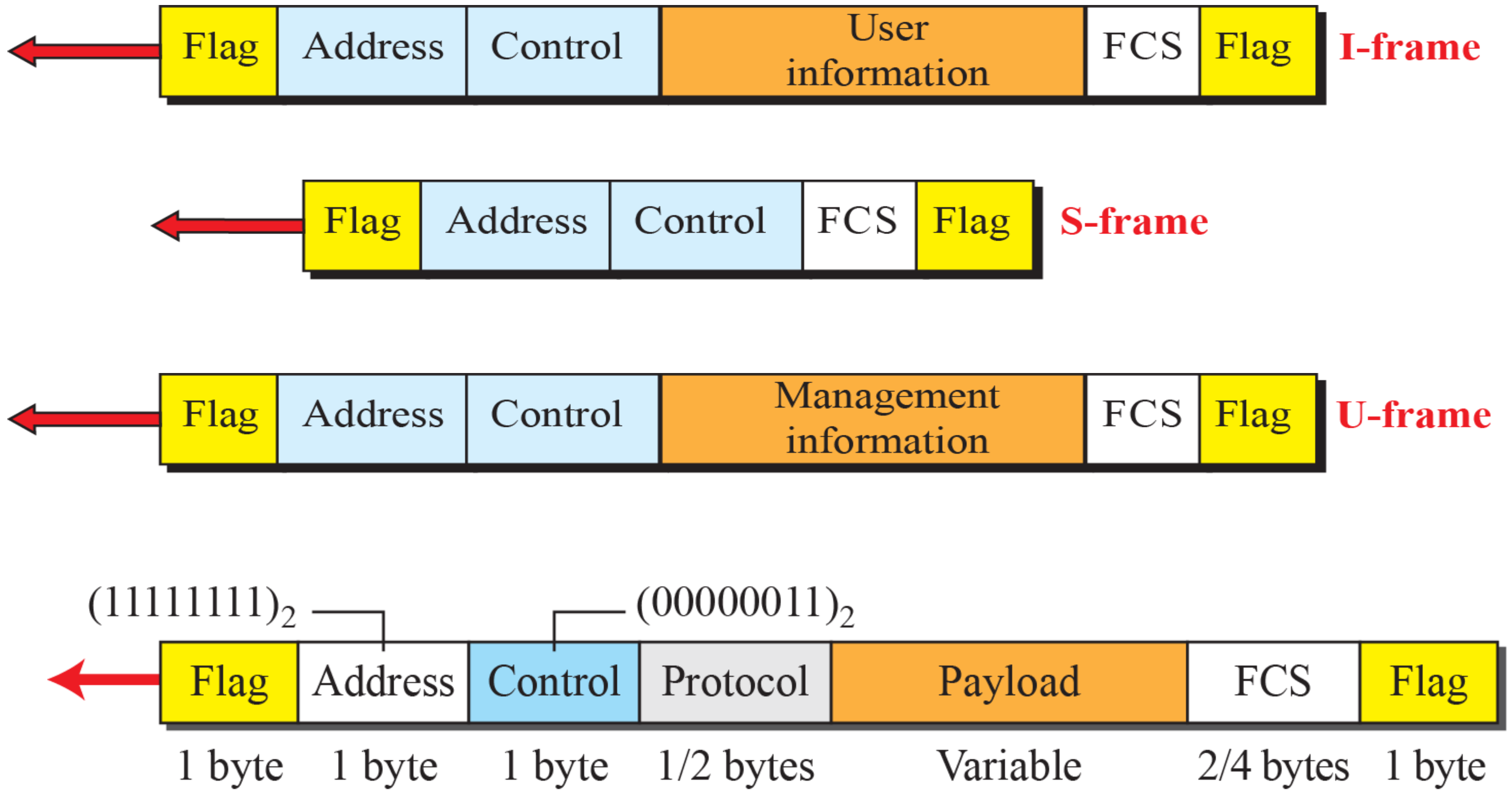

(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0
(b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0
Stuffed bits
(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

a) The original data.

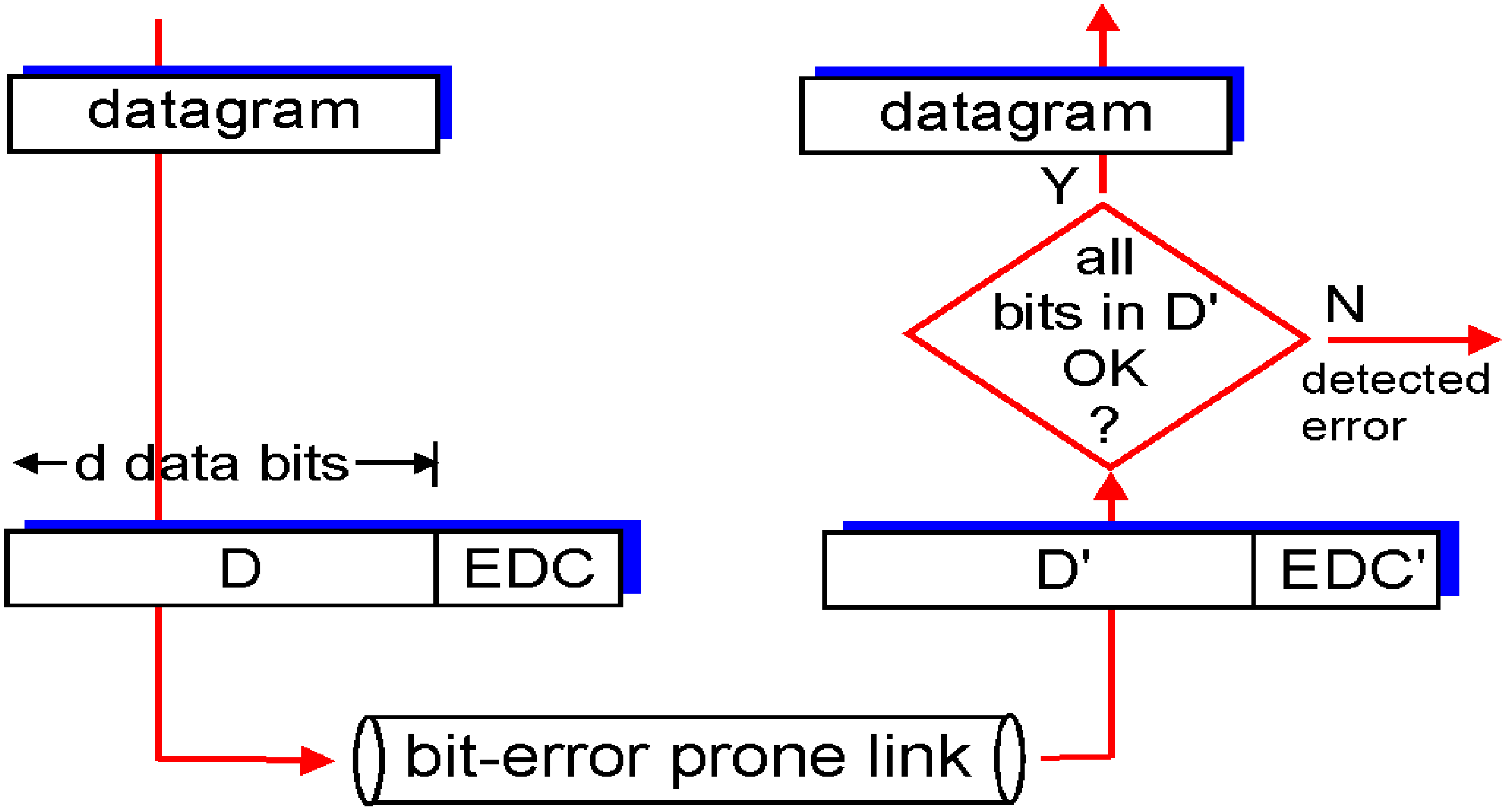
(b) The data as they appear on the line.

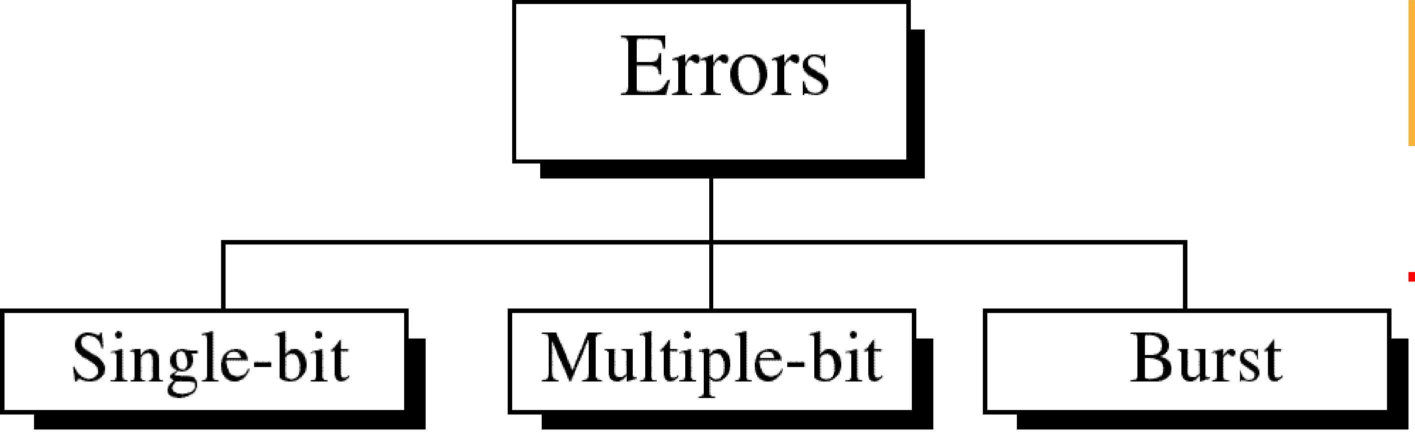
(c) The data as they are stored in receiver's memory after destuffing.

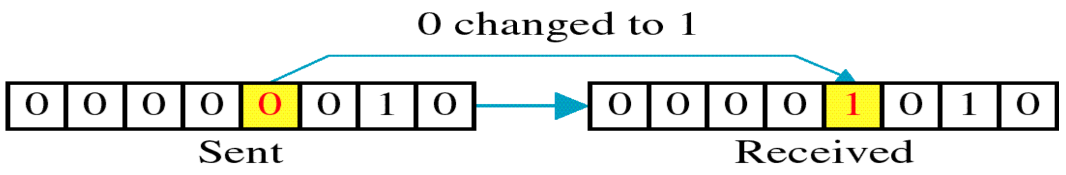
HDLC and PPP Frame formats

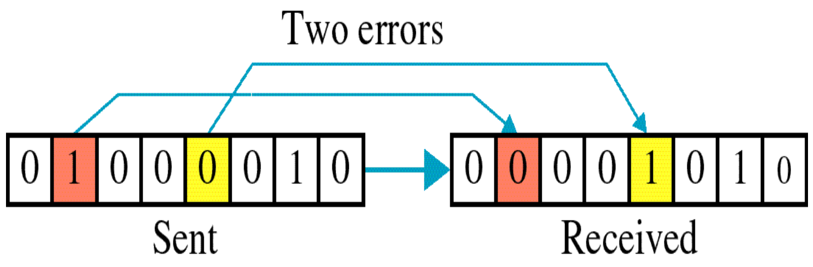


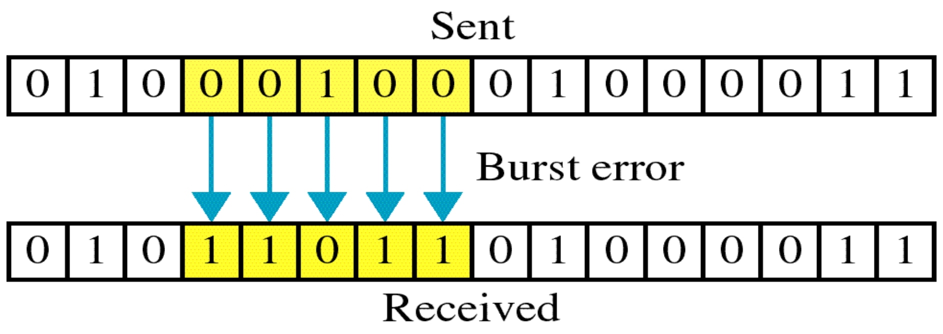
Error detection and Correction




Errors
Single-bit
Multiple-bit
Burst


0 changed to 1
0 0 0 0 0 0 1 0
Sent
0 0 0 0 1 0 1 0
Received


Two errors
0 1 0 0 0 0 1 0
Sent
0 0 0 0 1 0 1 0
Received

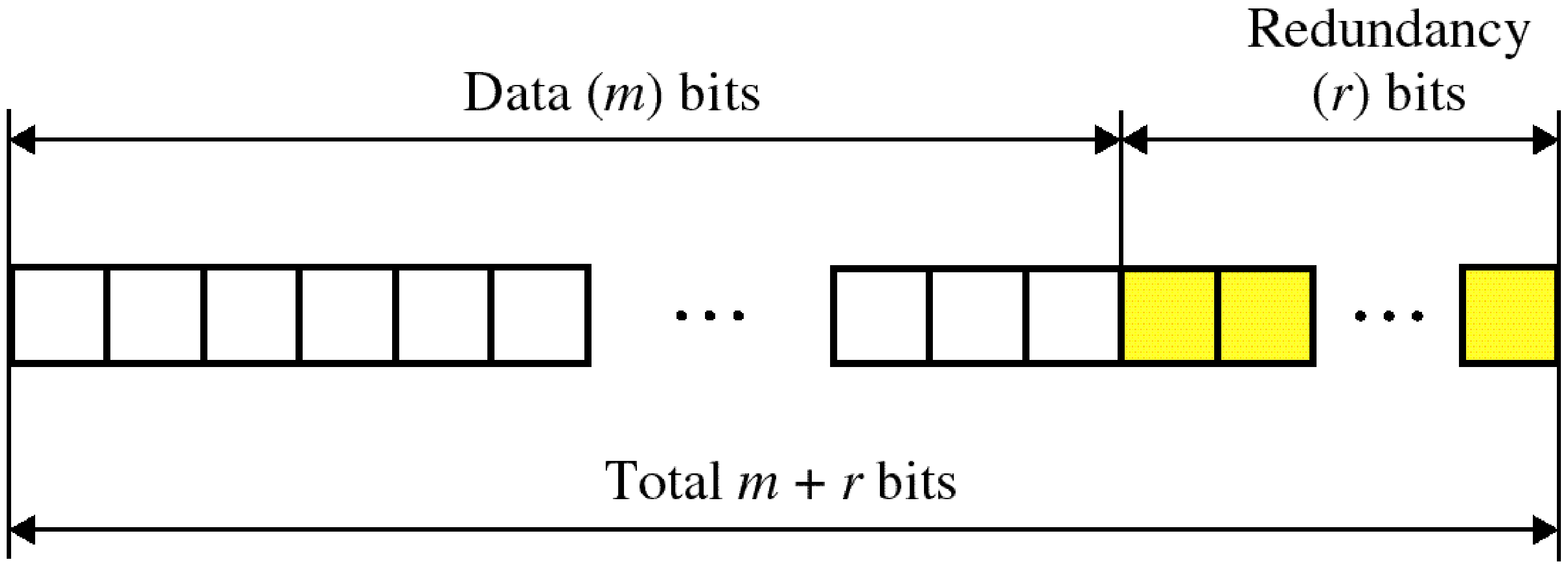

Sent
0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1
Burst error
0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 1
Received

Parity Check codes

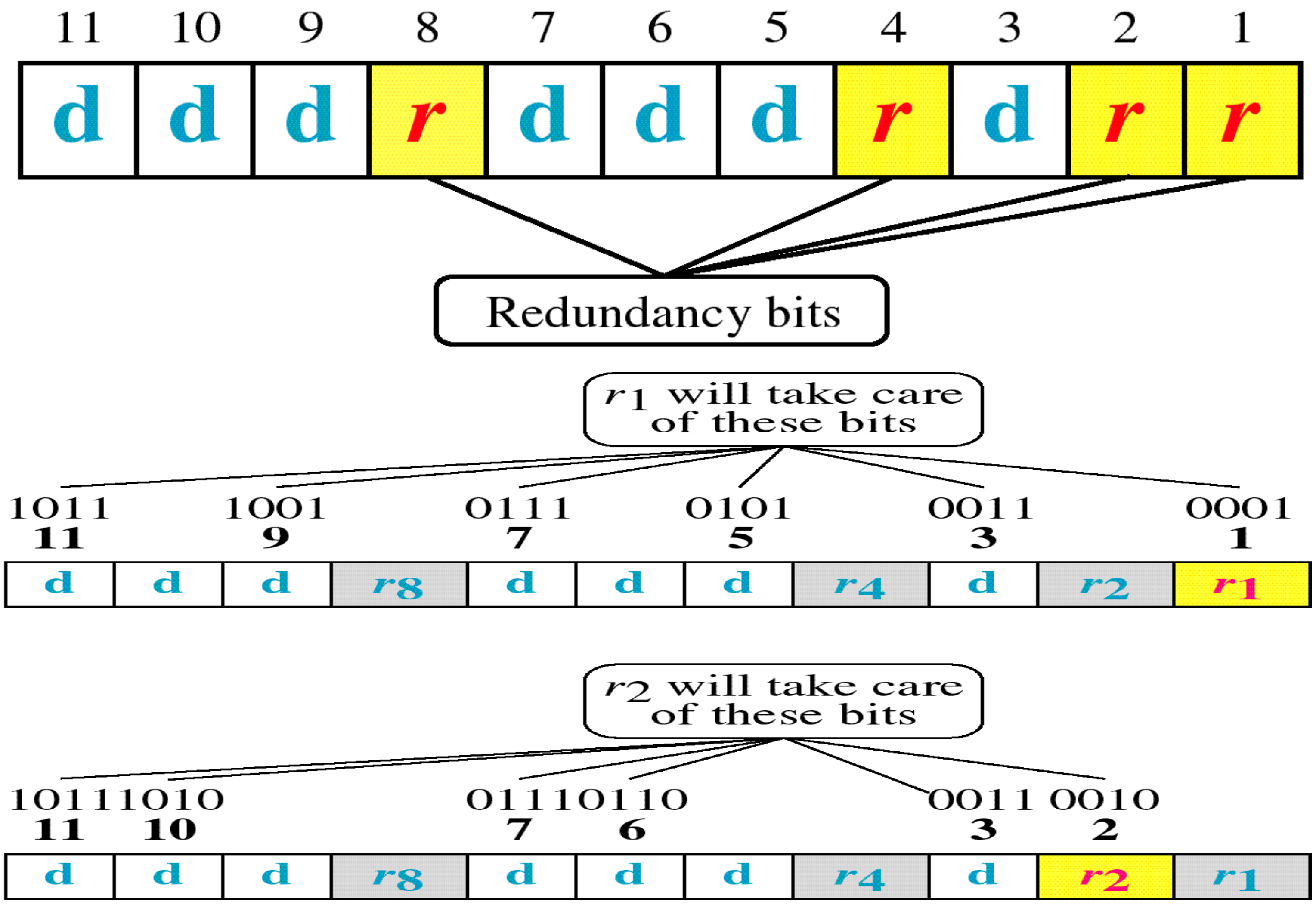
Datawords	*Codewords*	*Datawords*	*Codewords*
00	000	10	101
01	011	11	110

Datawords	**Codewords**	*Datawords*	**Codewords**
0000	**00000**	1000	**10001**
0001	**00011**	1001	**10010**
0010	**00101**	1010	**10100**
0011	**00110**	1011	**10111**
0100	**01001**	1100	**11000**
0101	**01010**	1101	**11011**
0110	**01100**	1110	**11101**
0111	**01111**	1111	**11110**

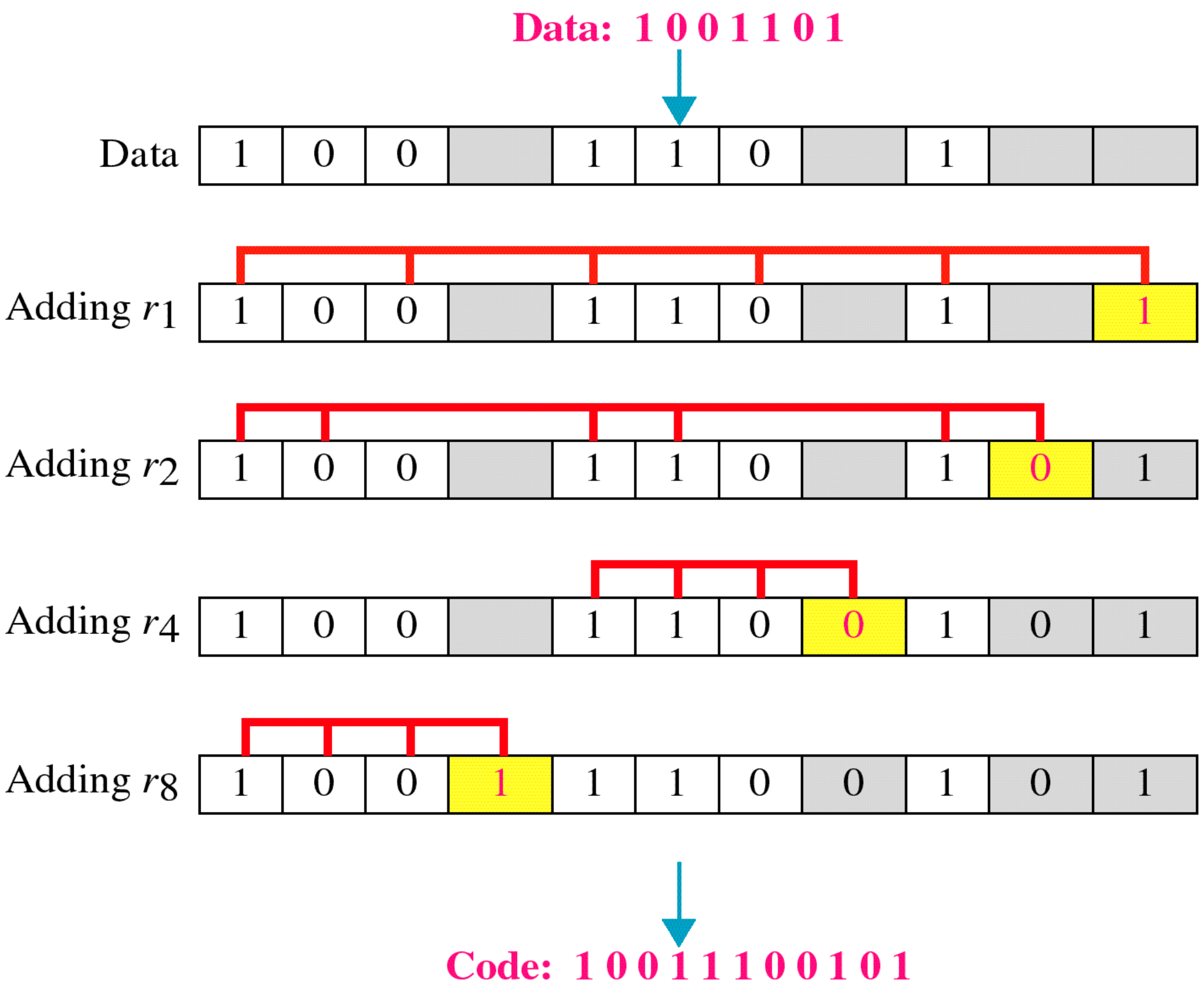
Error Correction



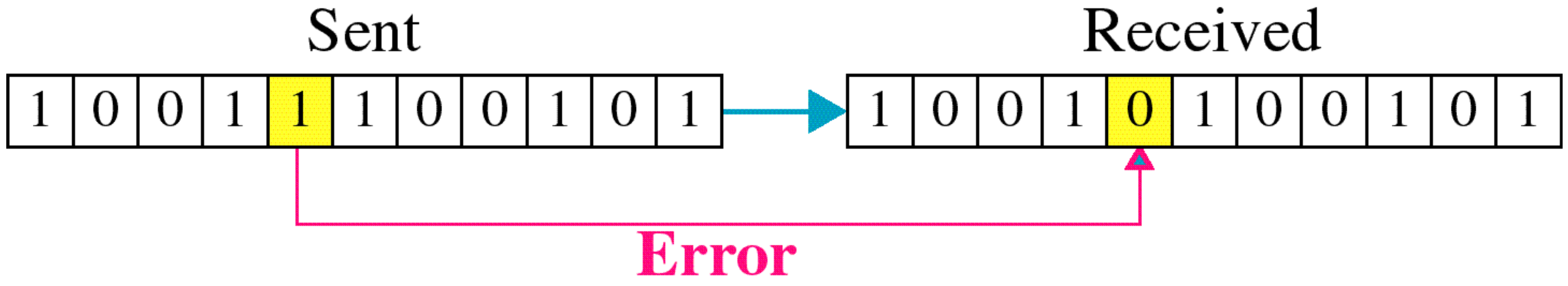
Hamming Code



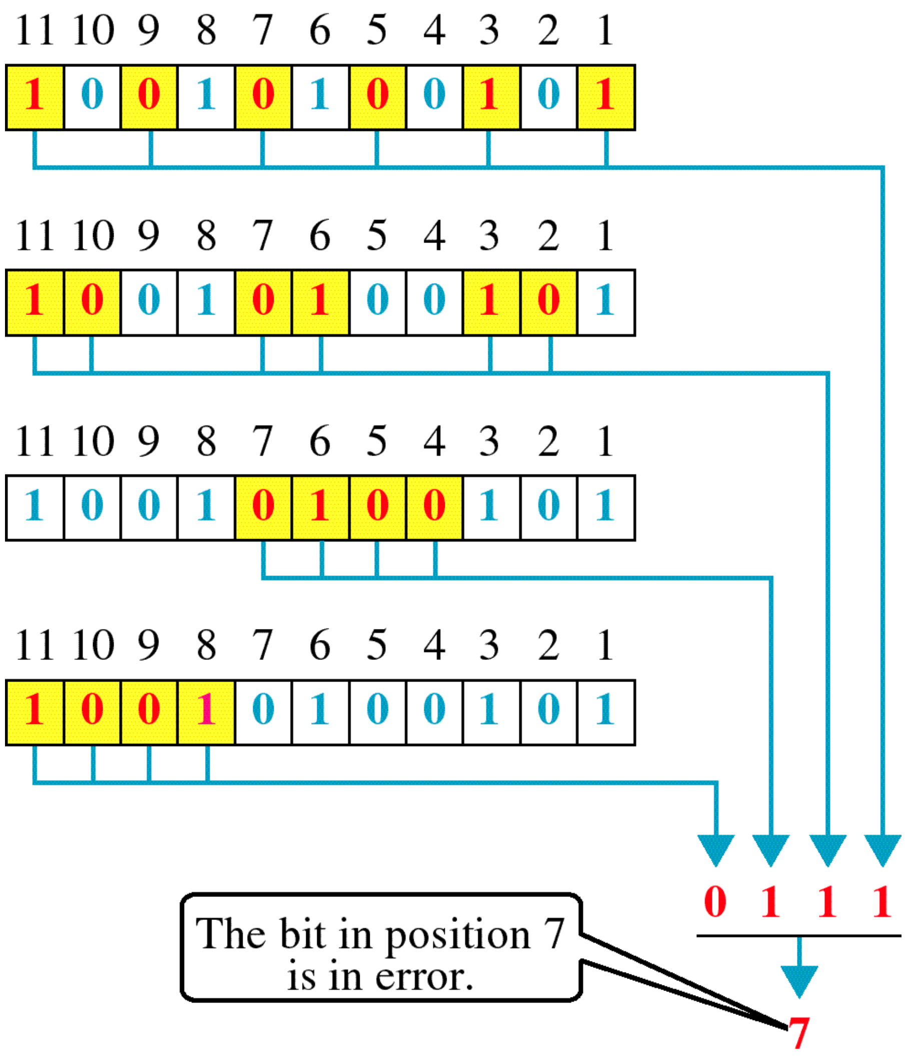
Hamming code example



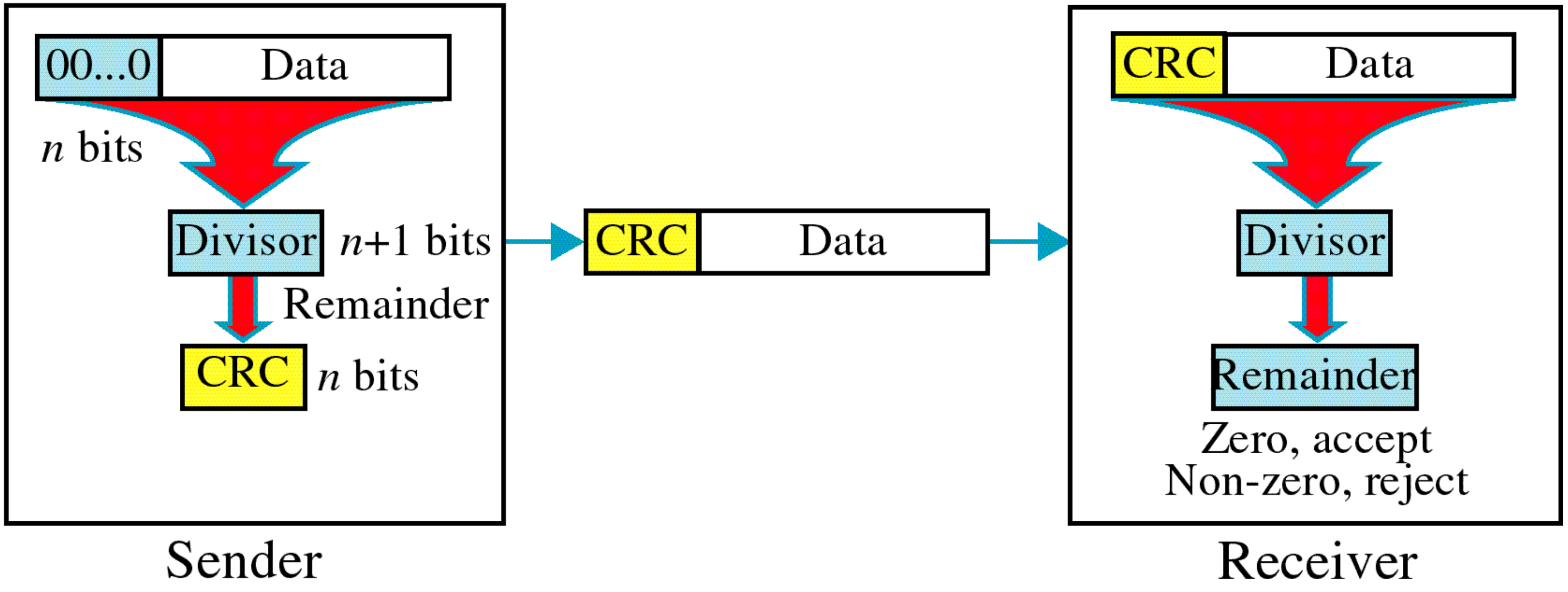
Single bit error



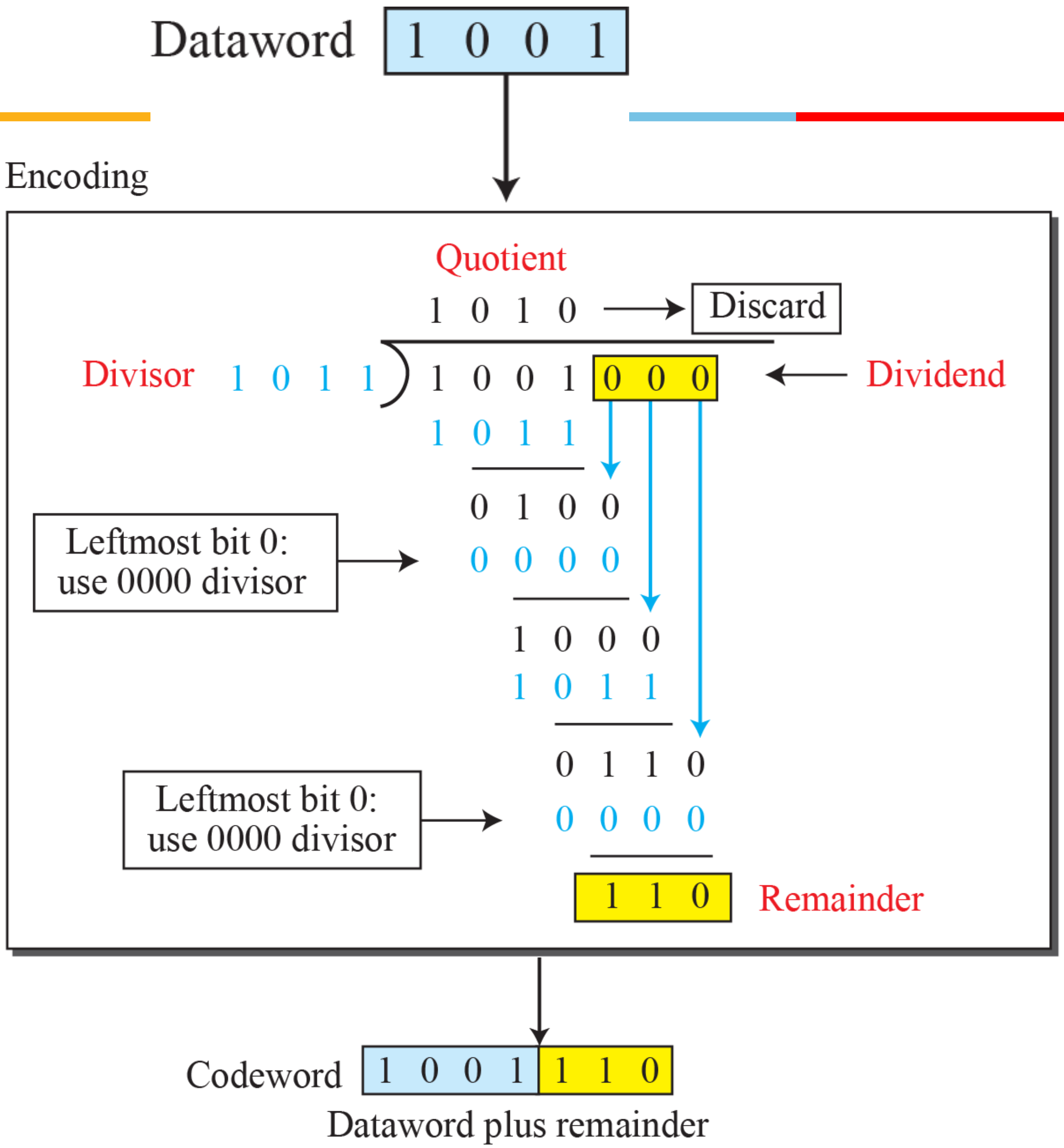
Error detection



Cyclic Redundancy Check



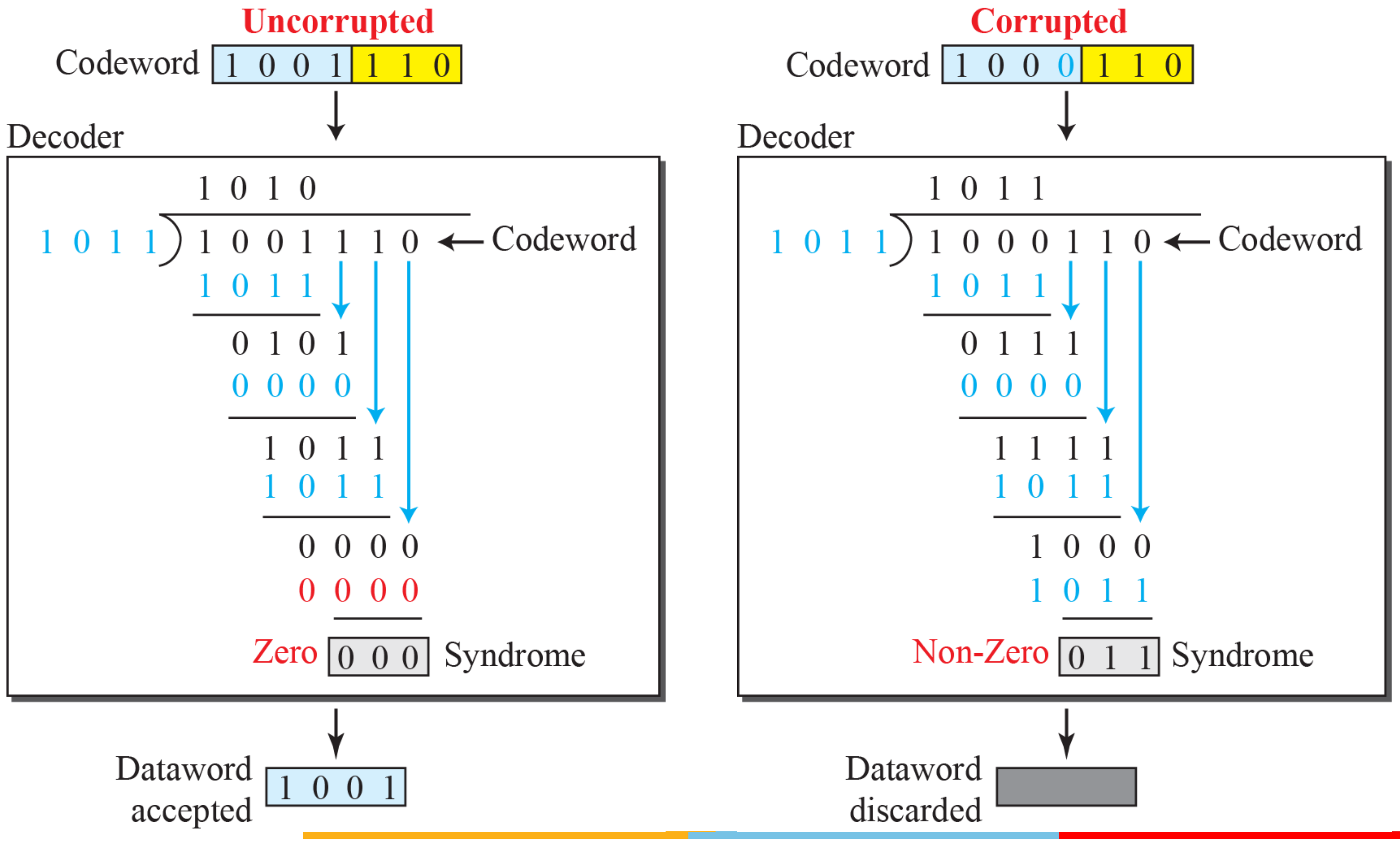
Example



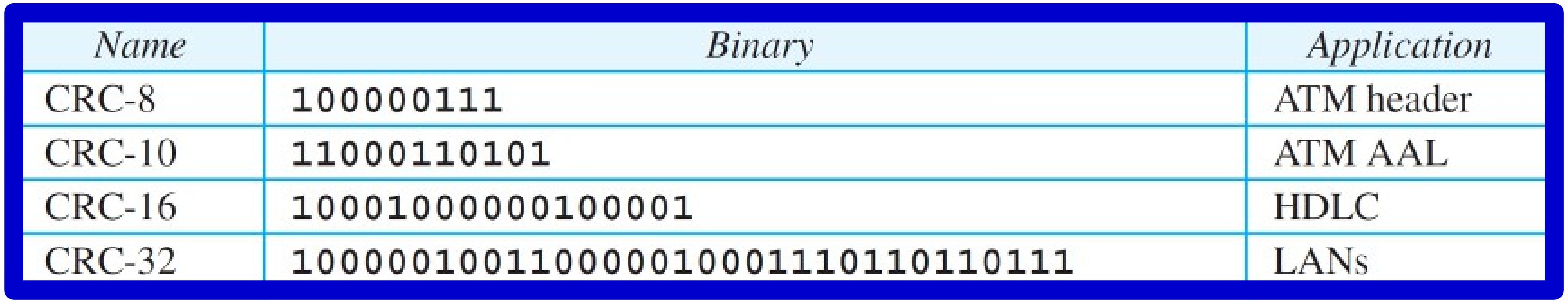
Note:

Multiply: AND
Subtract: XOR

Continued…


Uncorrupted
Codeword 1 0 0 1 1 1 0
Decoder
1 0 1 0
1 0 1 1) 1 0 0 1 1 1 0 ← Codeword
1 0 1 1
0 1 0 1
0 0 0 0
1 0 1 1
1 0 1 1
0 0 0 0
0 0 0 0
Zero 0 0 0 Syndrome
Dataword accepted 1 0 0 1
Corrupted
Codeword 1 0 0 0 1 1 0
Decoder
1 0 1 1
1 0 1 1) 1 0 0 0 1 1 0 ← Codeword
1 0 1 1
0 1 1 1
0 0 0 0
1 1 1 1
1 0 1 1
1 0 0 0
1 0 1 1
Non-Zero 0 1 1 Syndrome
Dataword discarded

Error detection in Practice



Name	*Binary*	*Application*
CRC-8	100000111	ATM header
CRC-10	11000110101	ATM AAL
CRC-16	10001000000100001	HDLC
CRC-32	100000100110000010001110110110111	LANs

- CRCs are widely used on links
  - Ethernet, 802.11, ADSL, Cable …
- Checksum used in Internet
  - IP, TCP, UDP … but it is weak
- Parity
  - Is little used