Student Online Teaching Advice Notice

The materials and content presented within this session are intended solely for use in a context of teaching and learning at Trinity.

Any session recorded for subsequent review is made available solely for the purpose of enhancing student learning.

Students should not edit or modify the recording in any way, nor disseminate it for use outside of a context of teaching and learning at Trinity.

Please be mindful of your physical environment and conscious of what may be captured by the device camera and microphone during videoconferencing calls.

Recorded materials will be handled in compliance with Trinity's statutory duties under the Universities Act, 1997 and in accordance with the University's policies and procedures.

Further information on data protection and best practice when using videoconferencing software is available at https://www.tcd.ie/info_compliance/data-protection/.

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

CSU33031 Computer Networks

High-level Data Link Control (HDLC)

Stefan Weber

email: sweber@tcd.ie

Office: Lloyd 1.41

HDLC

- ISO 33009, ISO 4335, Used initially in X.25

1979, ISO 3309

- It's old – so, why should we care?
  - Implements framing, addressing
  - Implements flow control mechanisms

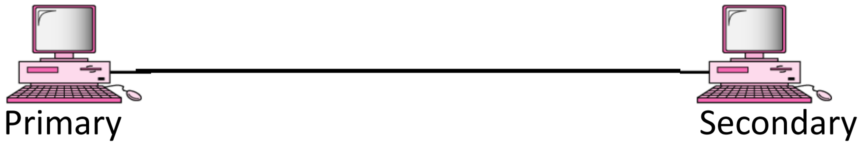
- Do we have to learn it by heart?
  - **No** – learn the principles – not the frame layouts!

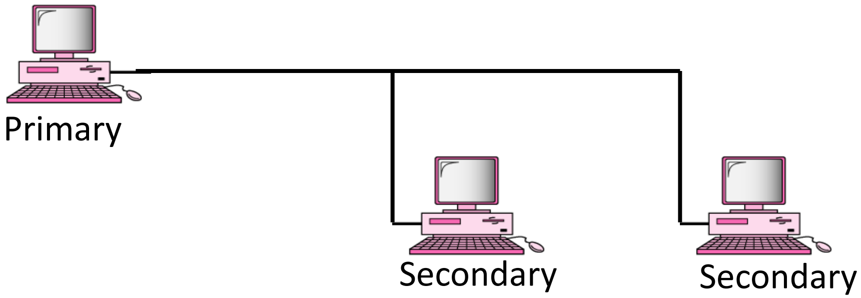
HDLC Station Types

- Primary station
  - Controls operation of link
  - Frames issued are called commands
- Secondary station
  - Under control of primary station
  - Frames issued called responses
- Combined station
  - Combination of primary and secondary station
  - May issue commands and responses

HDLC Station Types

Unbalanced Configuration

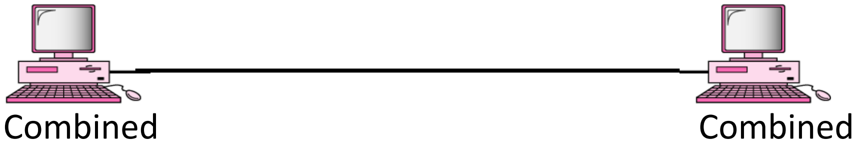




Two types of stations:

- Primary station
- Secondary station

Balanced Configuration



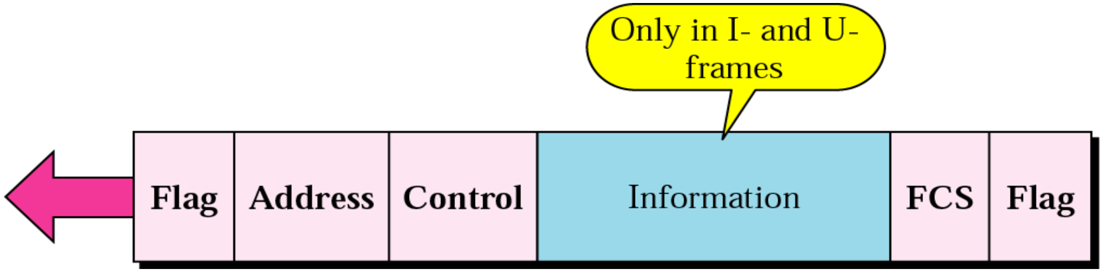
One type of stations:

- Combined station

HDLC Modes

- Three modes:
  - Normal Response Mode (NRM)
  - Asynchronous Response Mode (ARM)
  - Asynchronous Balanced Mode (ABM)

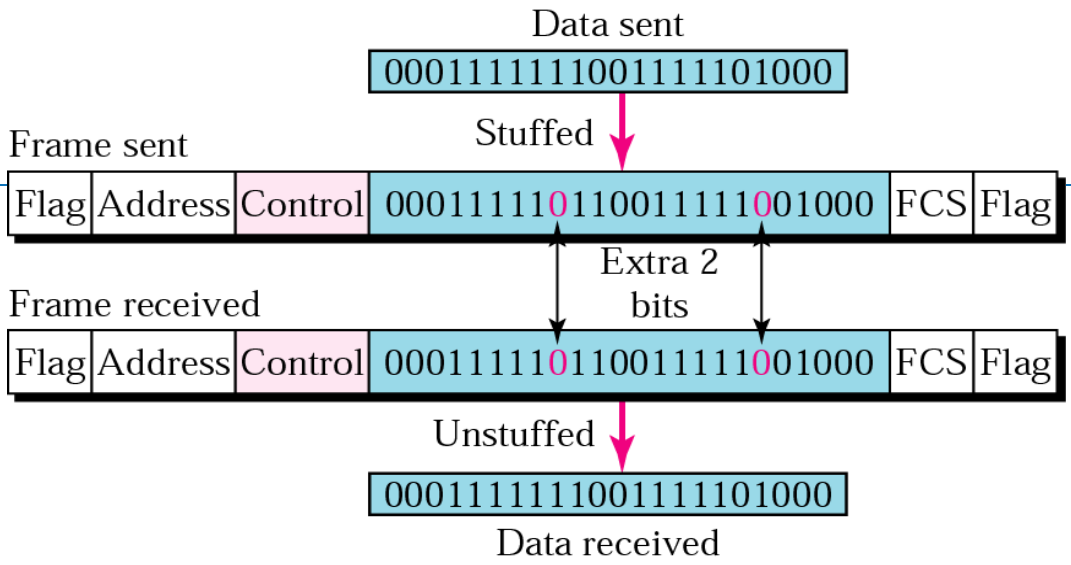
HDLC frame



- Flag= 01111110
  - specifies beginning and end of frame
- Address
  - specifies secondary station
  - as either sender or receiver
- Control
  - specifies type of frame and seq.&ack. number
- Frame Check Sequence (FCS)
  - either 16- or 32-bit CRC

* Figure is courtesy of B. Forouzan

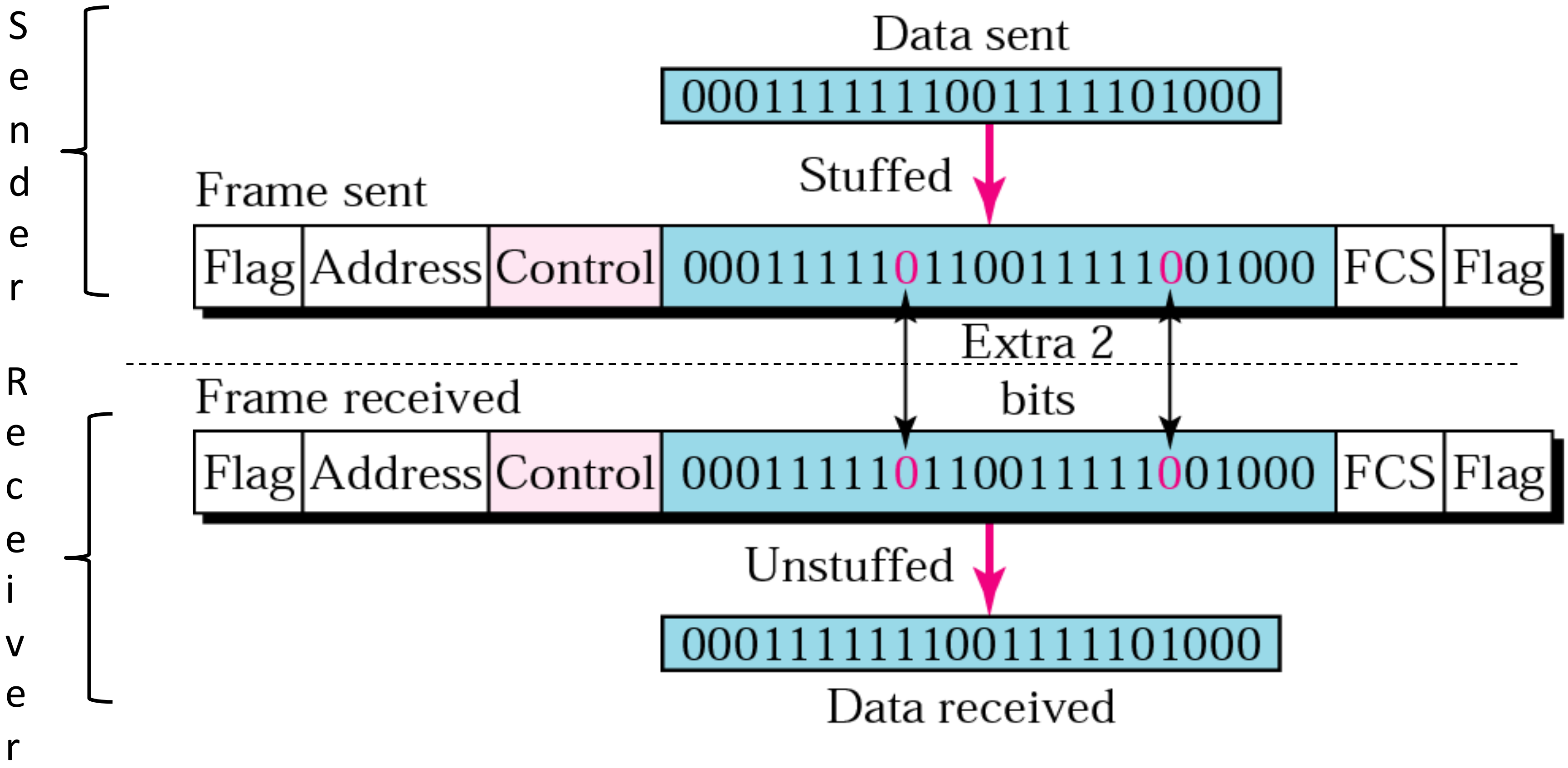
Bit-Stuffing



- Bit stuffing used to avoid confusion with data containing same combination as flag 01111110
  - 0 inserted after every sequence of five 1s
  - If receiver detects five 1s
    - it checks next bit
    - If 0, it is deleted
    - If 1 and seventh bit is 0, accept as flag
    - If sixth and seventh bits 1, sender is indicating abort

* Figure is courtesy of B. Forouzan

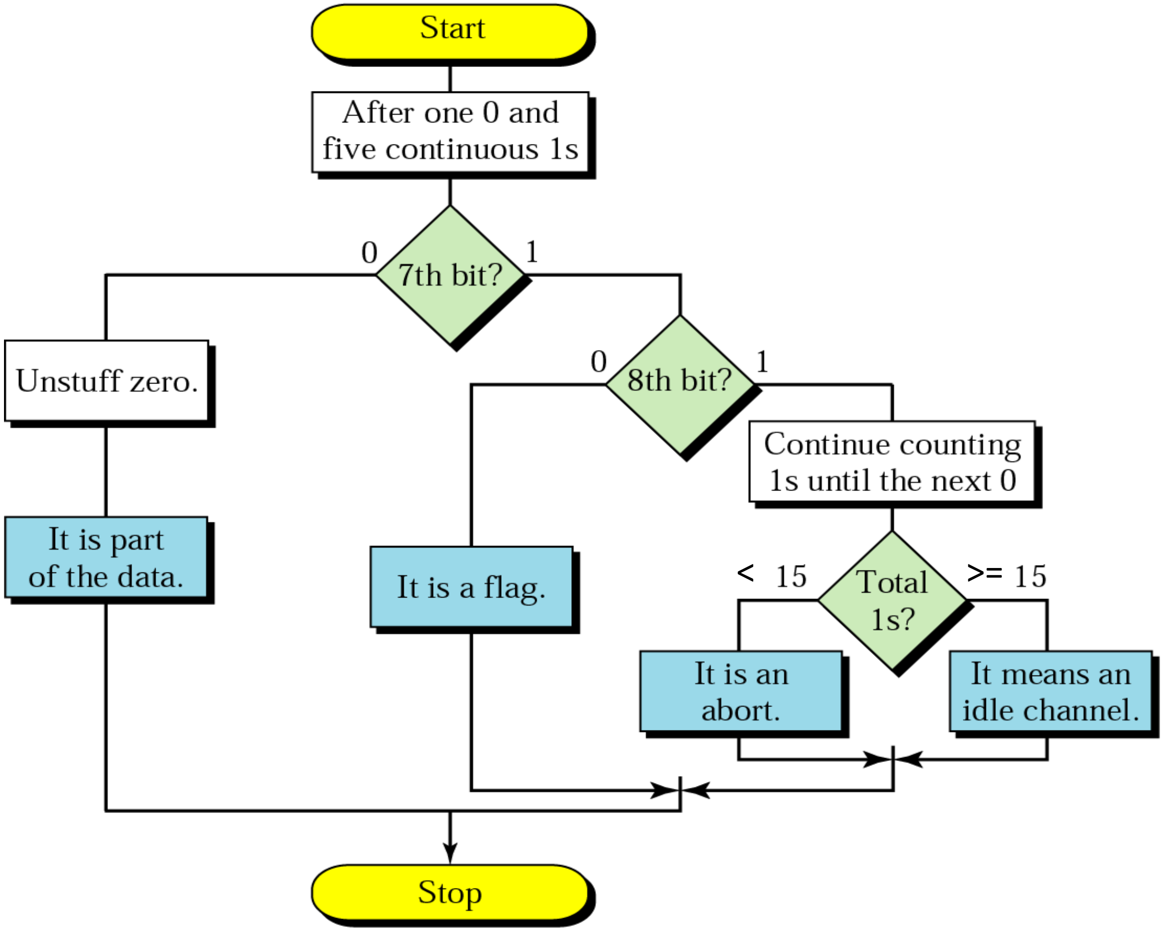
Bit Stuffing



Process of adding 0 whenever there is a flag or escape sequence in the text.

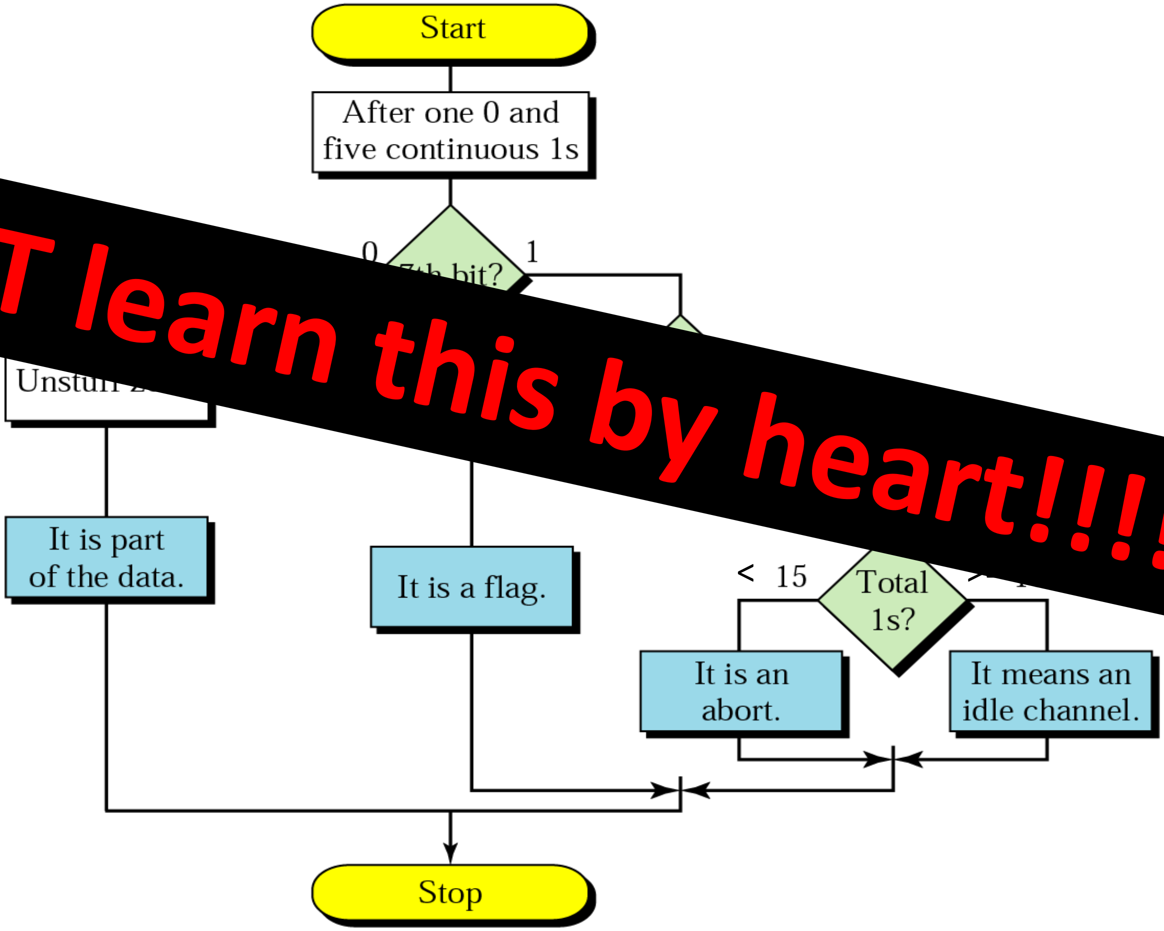
* Figure is courtesy of B. Forouzan

Bit stuffing in HDLC



* Figure is courtesy of B. Forouzan

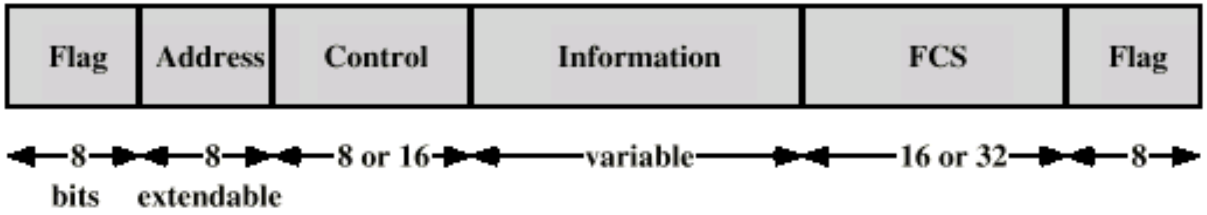
Bit stuffing in HDLC



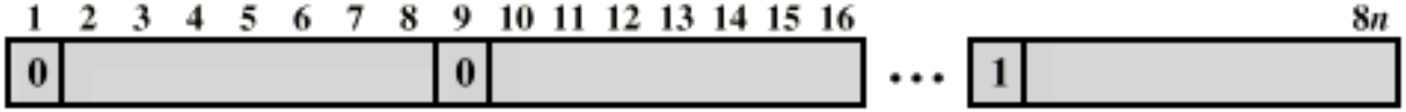
Do NOT learn this by heart!!!!!

* Figure is courtesy of B. Forouzan

Address Field

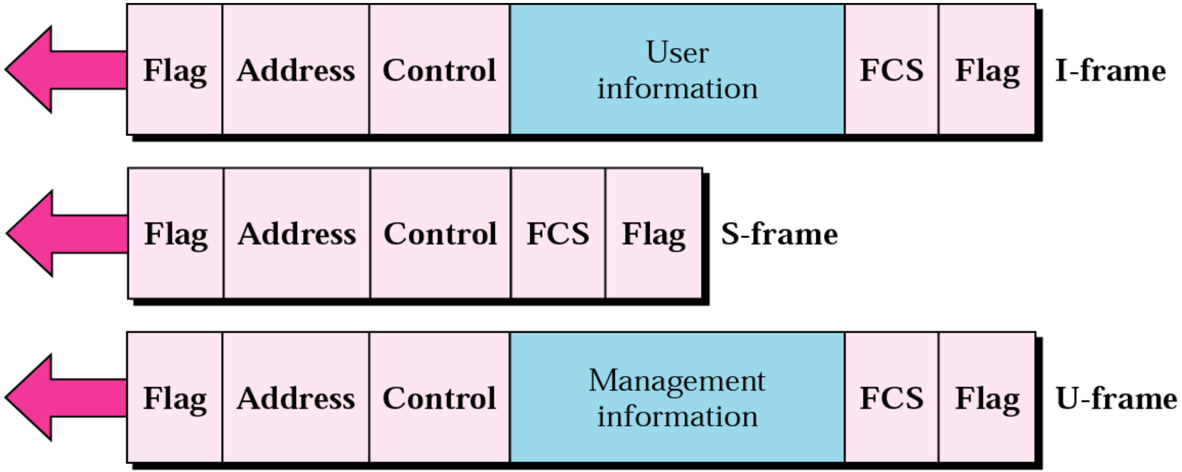


- Usually 8 bits long
- May be extended to multiples of 7 bits
  - LSB of each octet indicates that it is the last octet (1) or not (0)
- All ones (11111111) is broadcast



* Figure is courtesy of W. Stallings

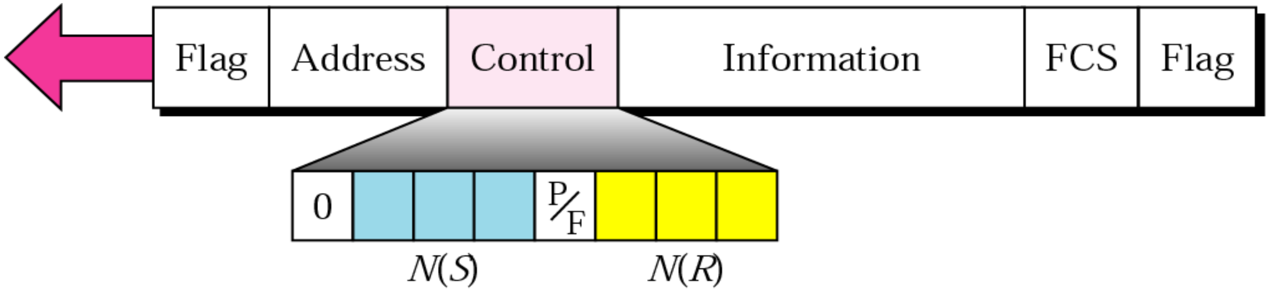
HDLC Frame Types



- I-Frame: Information Transfer Format
  - Control= 0 ? ? ? ? ? ? ?
- S-Frame: Supervisory Format
  - Control= 1 0 ? ? ? ? ? ?
- U-Frame: Unnumbered Format
  - Control= 1 1 ? ? ? ? ? ?

* Figure is courtesy of B. Forouzan

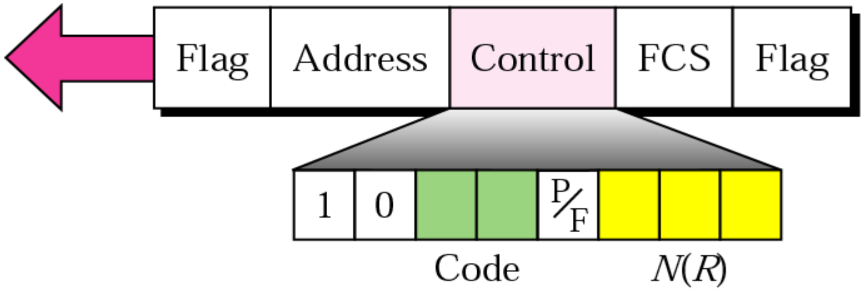
I-Frame



- N(S)
  - Sequence Number of Sender
- N(R)
  - Sequence Number of Receiver
- P/F
  - Poll/Final bit
  - Set by Primary station as request for information
  - Set by Secondary station to signal response or to signal final frame of a transmission

* Figure is courtesy of B. Forouzan

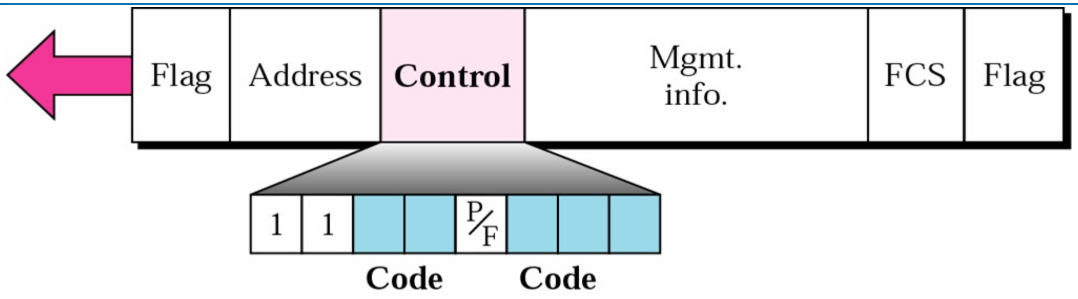
S-Frame Control Field



- Code 00 = Receive Ready (RR)
  - Acknowledge frames & waiting for more
- Code 10 = Receive Not Ready (RNR)
  - Acknowledge frames & busy right now
- Code 01 = Reject (REJ)
  - Go-Back-N NAK
- Code 11 = Selective Reject (SREJ)
  - Selective Repeat NAK

* Figure is courtesy of B. Forouzan

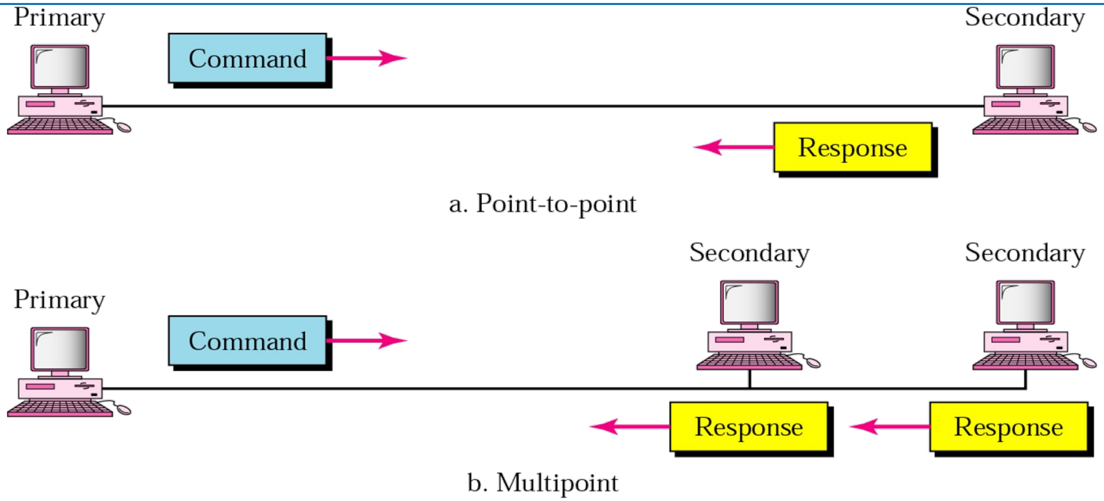
U-Frame Control Field



Code	Command/Response	Meaning
00 001	SNRM	Set normal response mode
11 100	SABM	Set asynchronous balanced mode
00 100	UP	Unnumbered poll
00 000	UI	Unnumbered information
00 110	UA	Unnumbered acknowledgment
00 010	DISC	Disconnect
10 000	SIM	Set initialization mode
11 001	RSET	Reset
11 101	XID	Exchange ID
10 001	FRMR	Frame reject

* Figure is courtesy of B. Forouzan

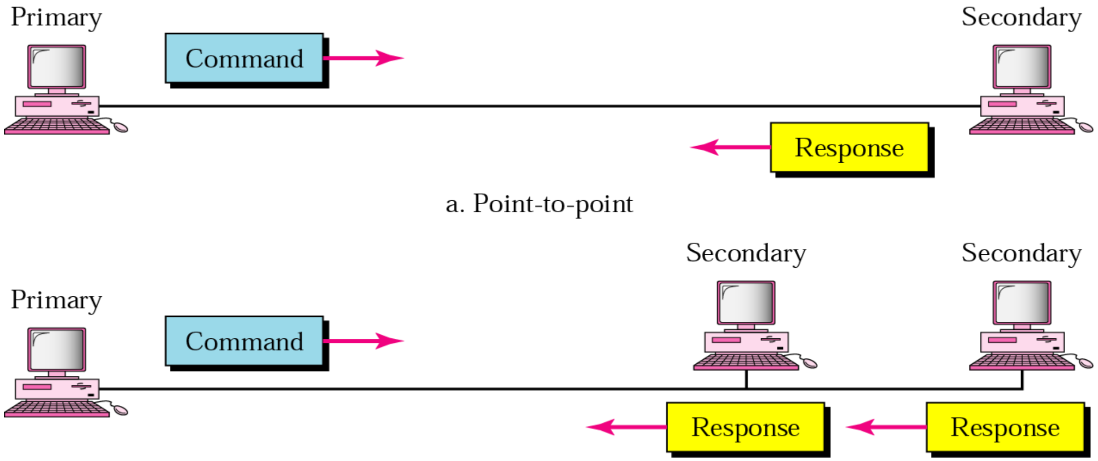
Normal Response Mode (NRM)



- Master/Slave architecture
- Unbalanced configuration
- Primary initiates transfer to secondary
- Secondary may only transmit data in response to command from primary
- Used on multi-drop lines

* Figure is courtesy of B. Forouzan

Asynchronous Response Mode (ARM)



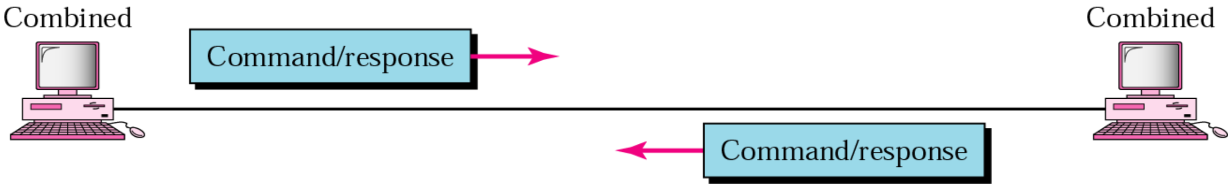
a. Point-to-point

b. Multipoint

- Unbalanced configuration
- Secondary may initiate transmission without permission form primary
- Primary responsible for line
- Rarely used

* Figure is courtesy of B. Forouzan

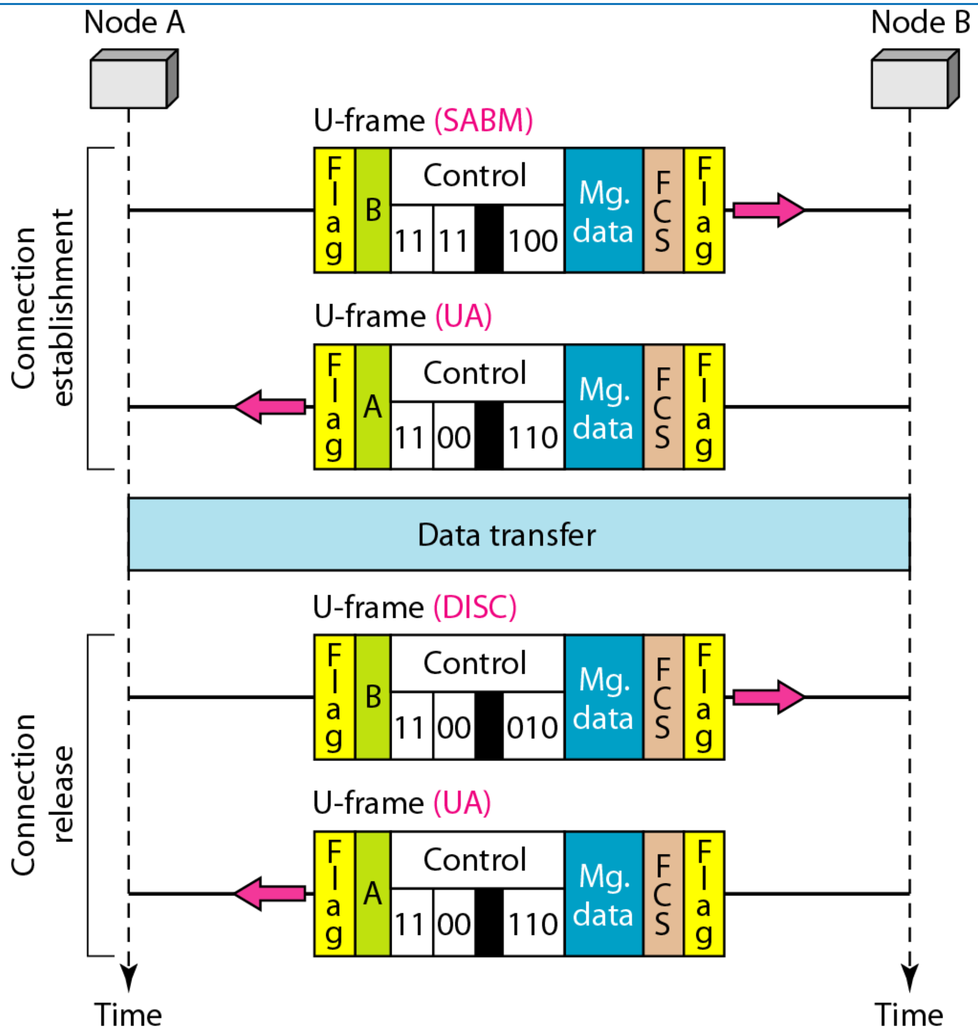
Asynchronous Balanced Mode (ABM)



- Balanced configuration
- Either station may initiate transmission without receiving permission
- Most widely used
- No polling overhead

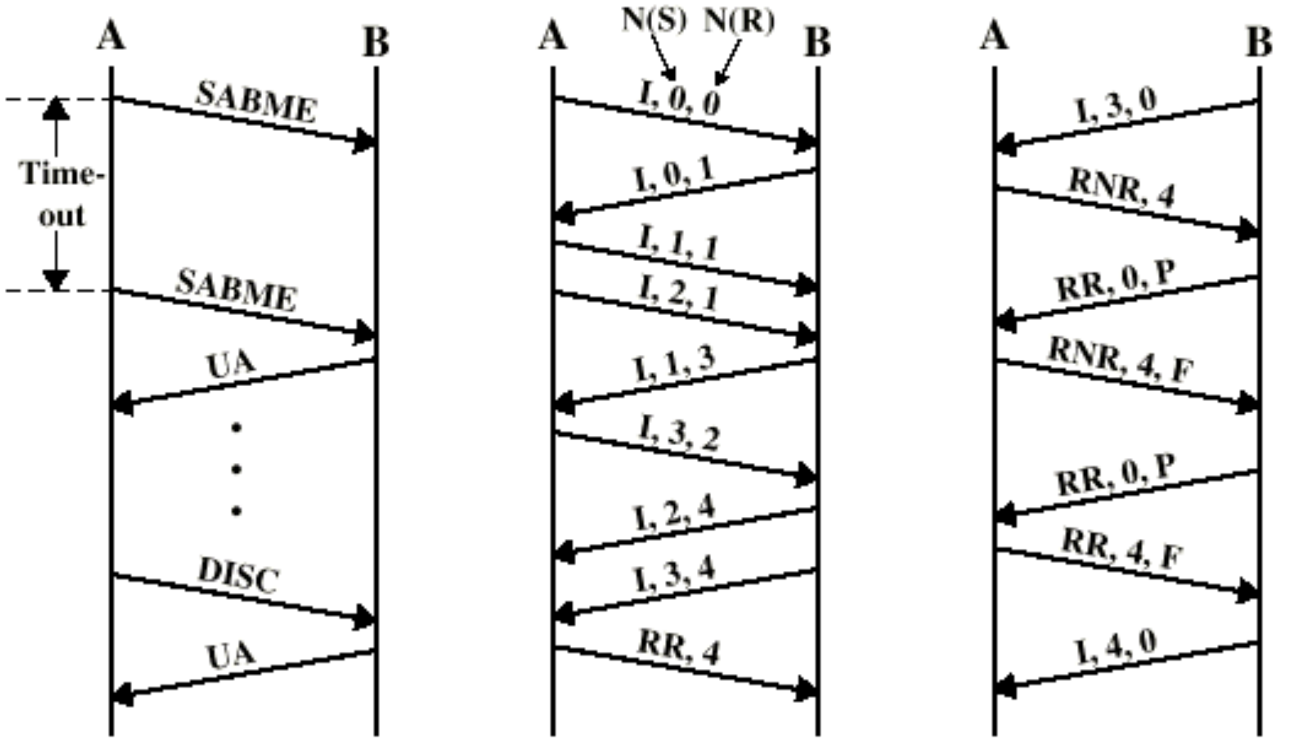
* Figure is courtesy of B. Forouzan

Connection & Disconnection



* Figure is courtesy of B. Forouzan

Examples of Operation



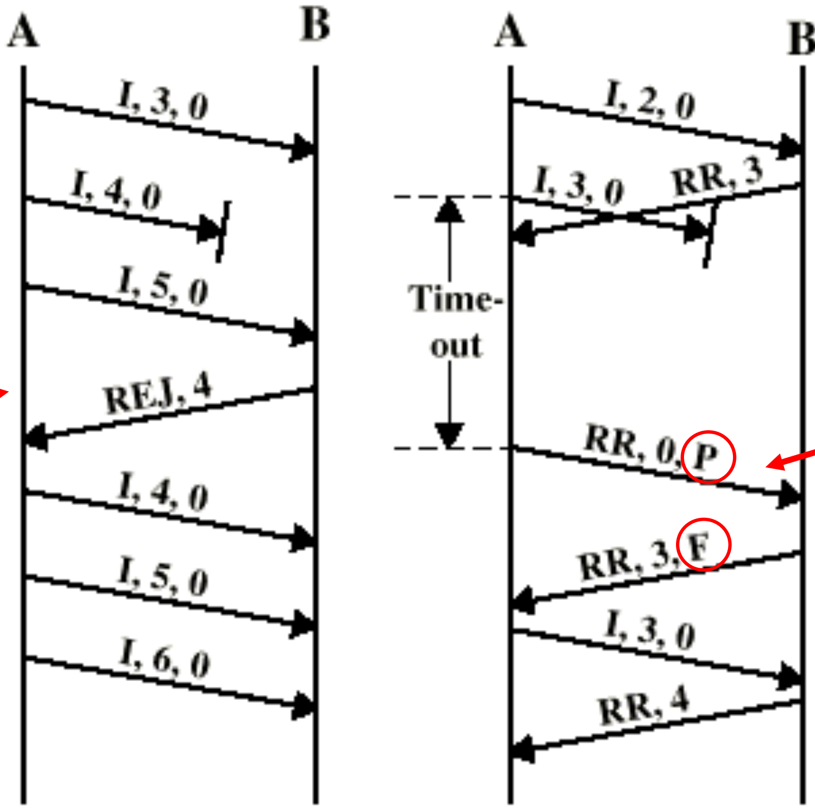
(a) Link setup and disconnect (b) Two-way data exchange (c) Busy condition

SABME	Set asynchronous balanced mode	RR	Receive Ready
I	Information	RNR	Receive Not Ready
UA	Unnumbered acknowledgment	REJ	Reject
DISC	Disconnect	SREJ	Selective Reject

tesy of W. Stallings

Example

Received 5 but didn't receive 4



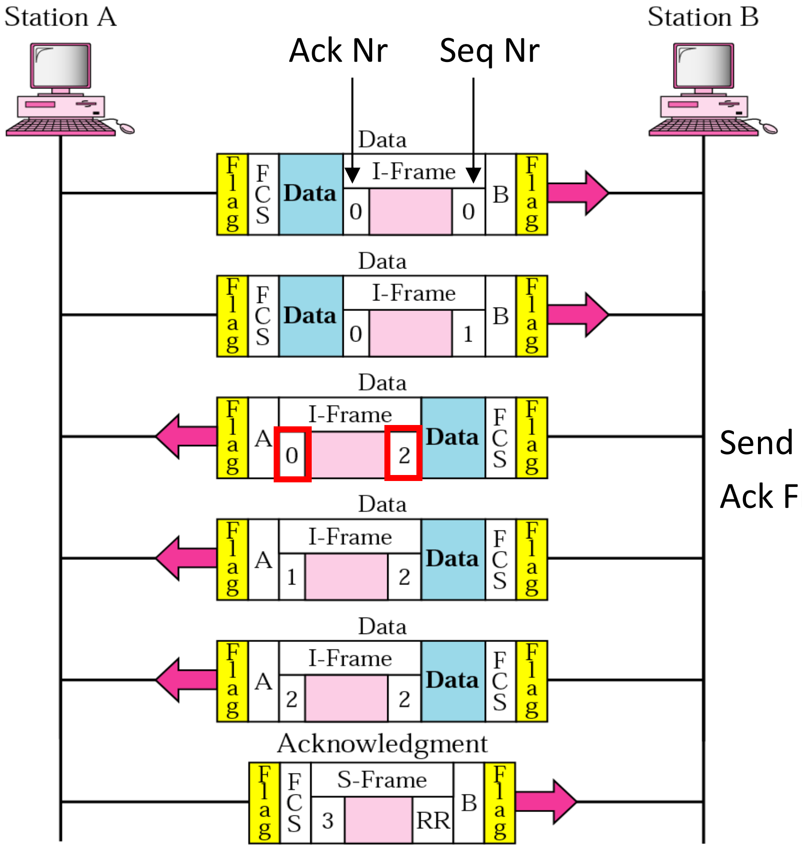
(d) Reject recovery

(e) Timeout recovery

Hello?? Anybody home??

* Figure is courtesy of W. Stallings

Piggybacking without Error

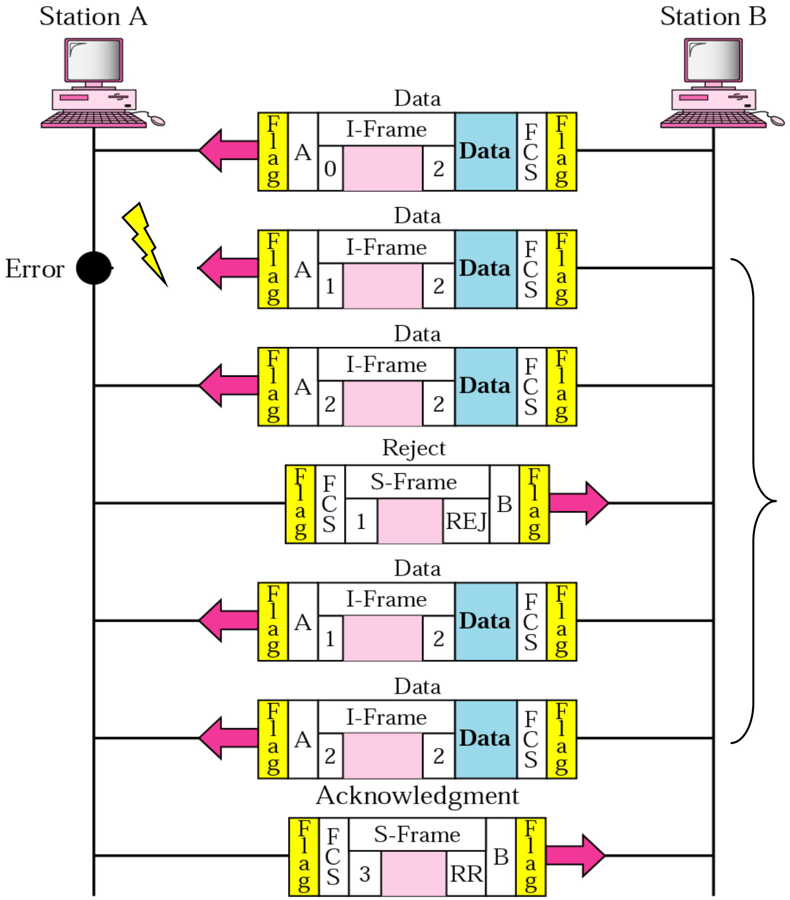


Send Frame 0 &

Ack Frame 0+1

* Figure is courtesy of B. Forouzan

Piggybacking with Error



Go-Back-N

* Figure is courtesy of B. Forouzan

Summary: HDLC

- Three station types
  - Primary station
  - Secondary station
  - Combined station
- Operation modes
  - Normal response mode
  - Asynchronous response mode
- Three frame types
  - I-Frame: Information Transfer Format
  - S-Frame: Supervisory Format – Flow Control
  - U-Frame: Unnumbered Format – Connection setup/term./etc
- Bit-Stuffing - to avoid confusion of data and flag

HDLC – Why?

- ‘should give you a feeling for a protocol

- It includes most of the basic mechanisms
  - Framing
  - Addressing
  - Bit-stuffing
  - Flow/Error control

- Once you can run through HDLC in your head, you understand the basics of link layer protocols

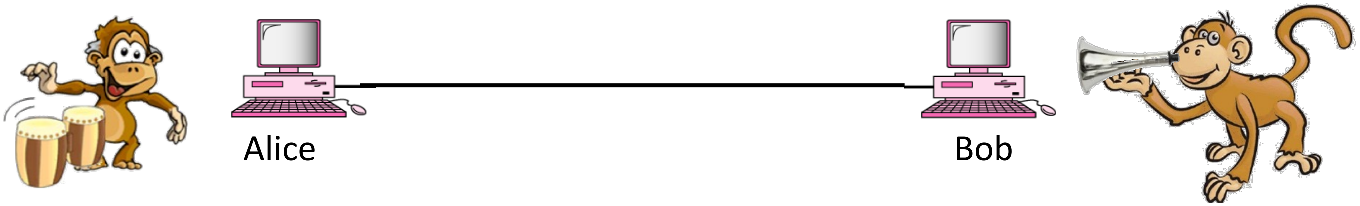
Binary Example

01111110011111000111101101010110101111110

0111111001111010110001101110110101111110

01111110011111000110000101011110101111110

0111111001111010110001101110110101111110



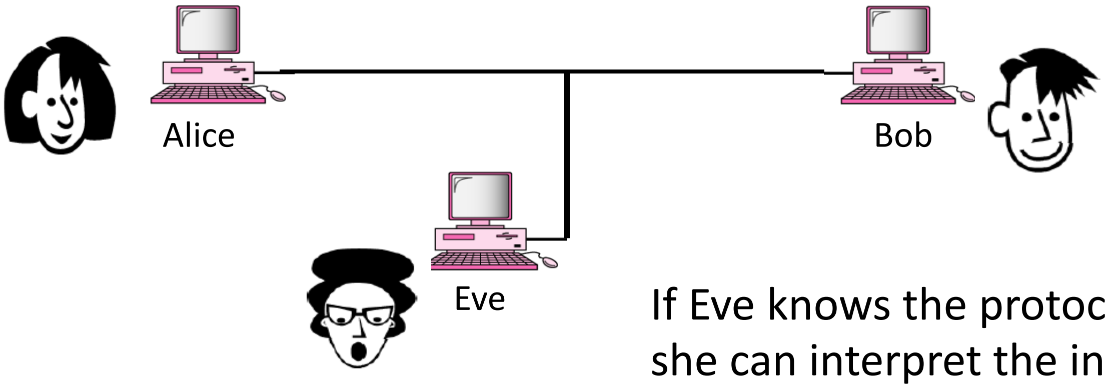
Binary Example

```
01111110011111000111101101010110101111110

0111111001111010110001101110110101111110

01111110011111000110000101011110101111110

0111111001111010110001101110110101111110
```



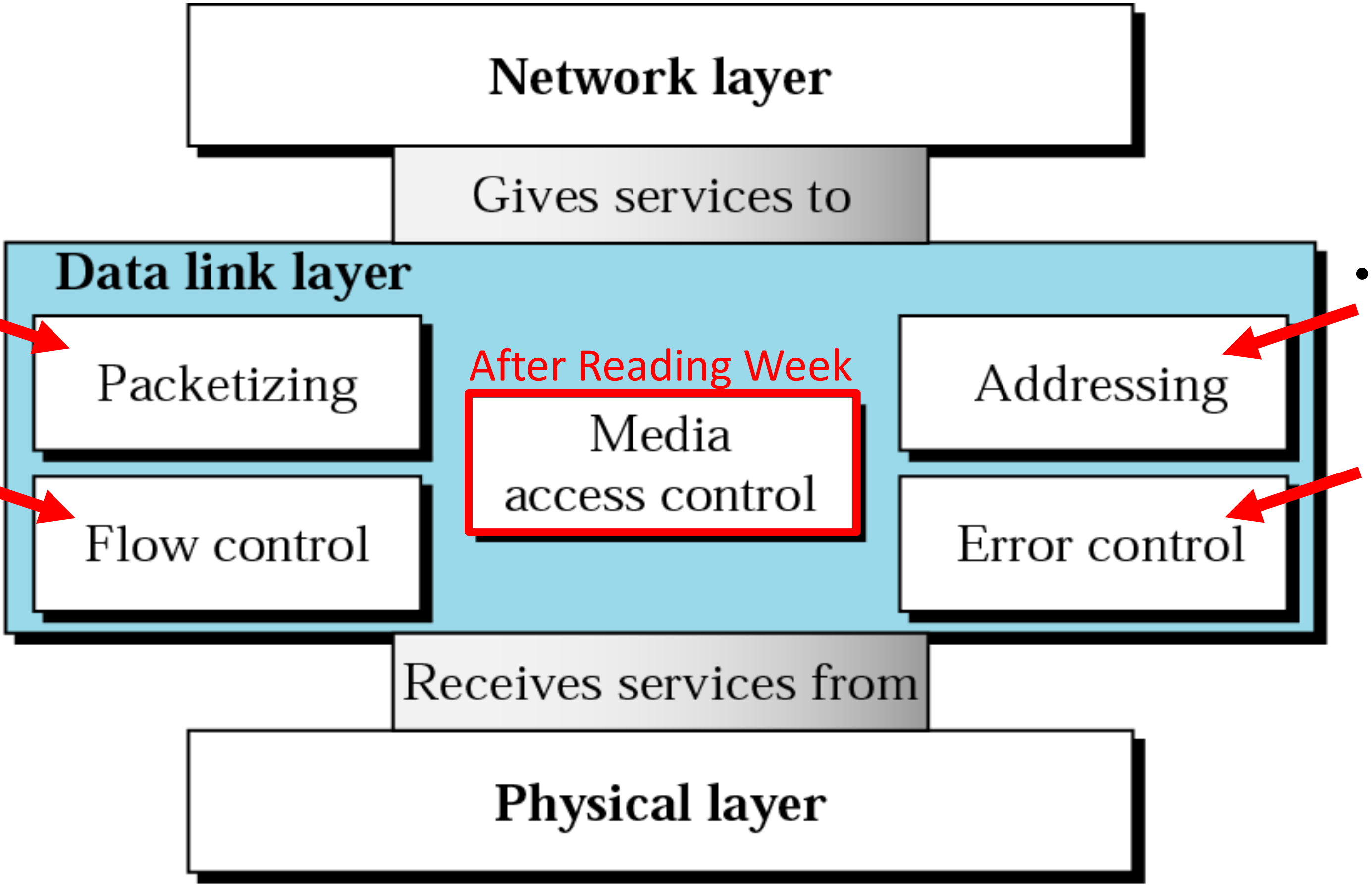
If Eve knows the protocol,
she can interpret the information

Link Layer

•Flag Bytes

•Frame Format

•Options for Stop&Wait, Selective Repeat, etc

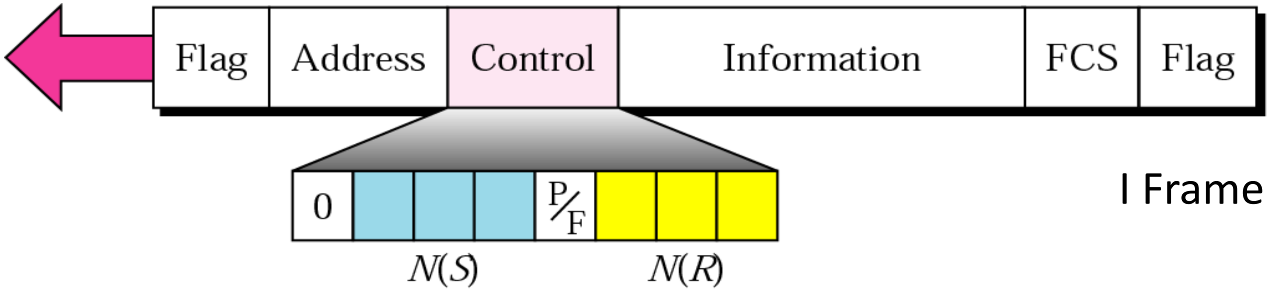


•Destination Address

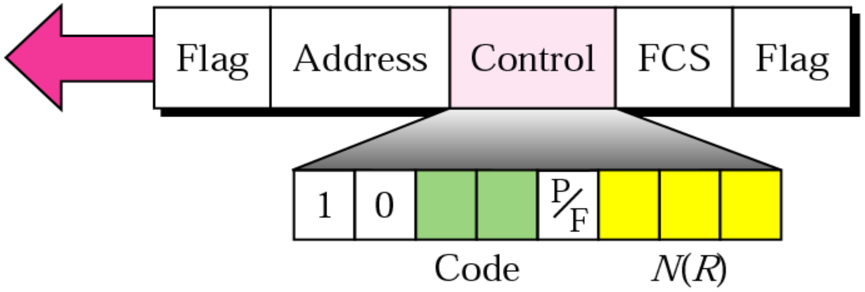
•CRC

* Figure is courtesy of B. Forouzan

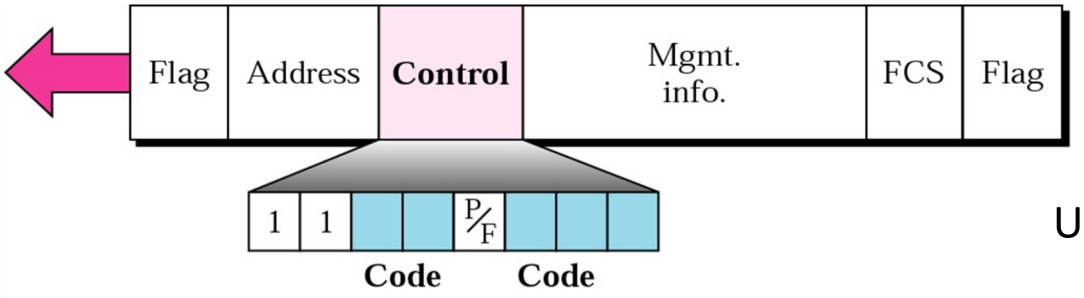
HDLC Frames



I Frame: Transfer of data



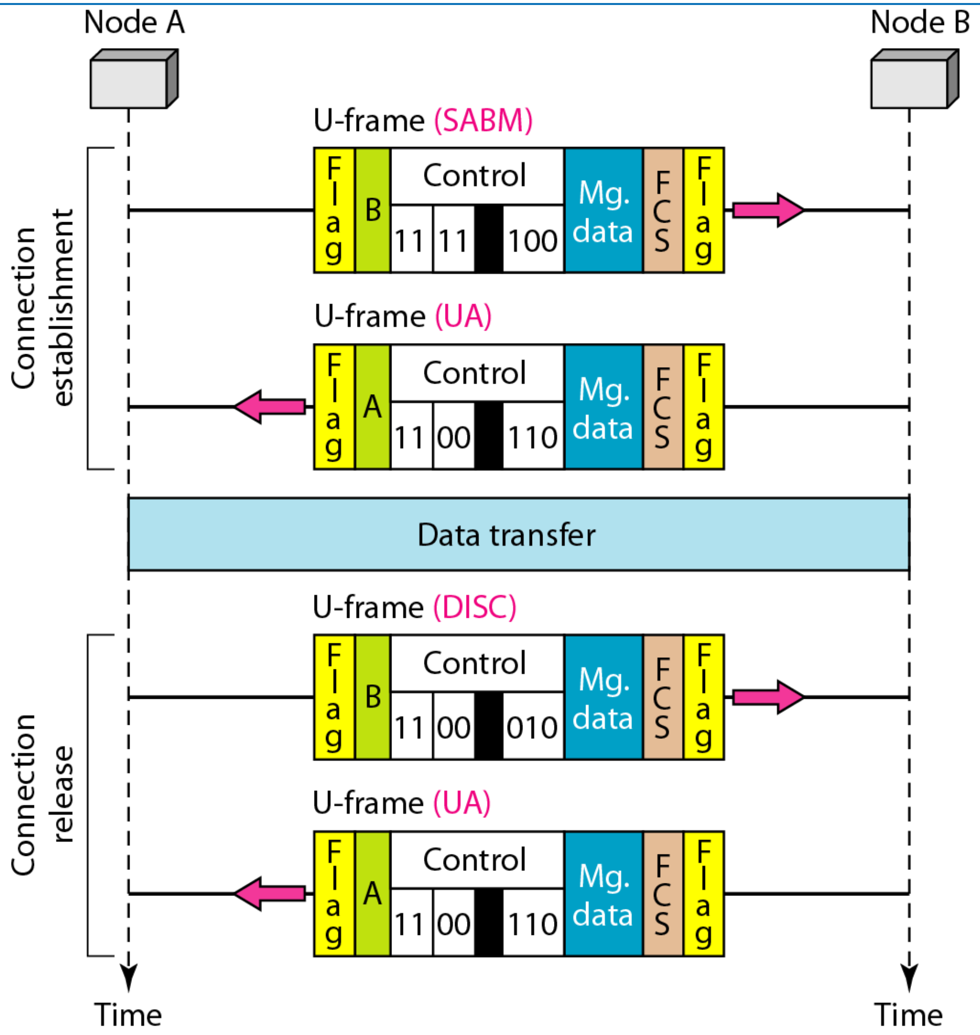
S Frame: Flow Control



U Frame: Connection Mgmt

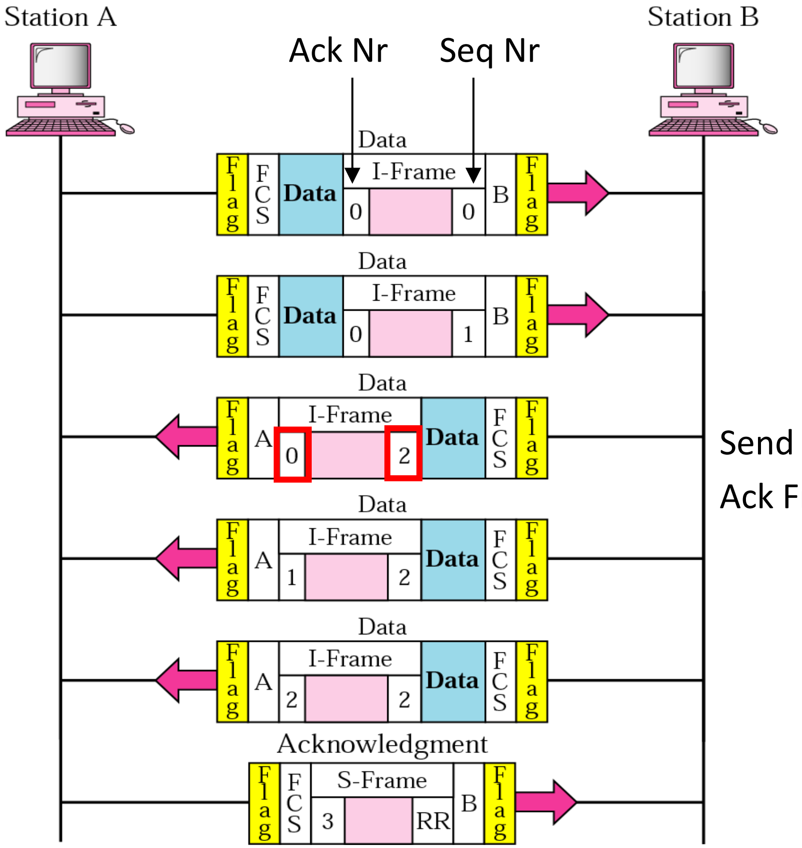
* Figure is courtesy of B. Forouzan

Connection & Disconnection



* Figure is courtesy of B. Forouzan

Piggybacking without Error



Send Frame 0 &

Ack Frame 0+1

* Figure is courtesy of B. Forouzan

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

CSU33031 Computer Networks

Error Detection Speed Run

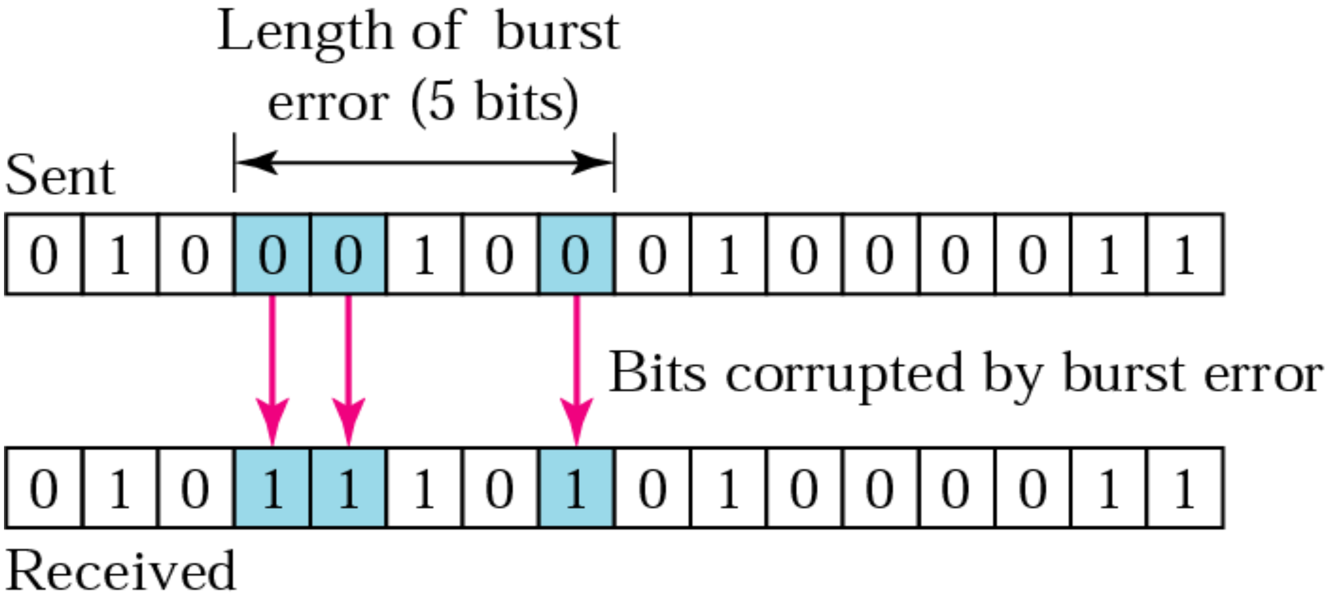
Stefan Weber

email: sweber@tcd.ie

Office: Lloyd 1.41

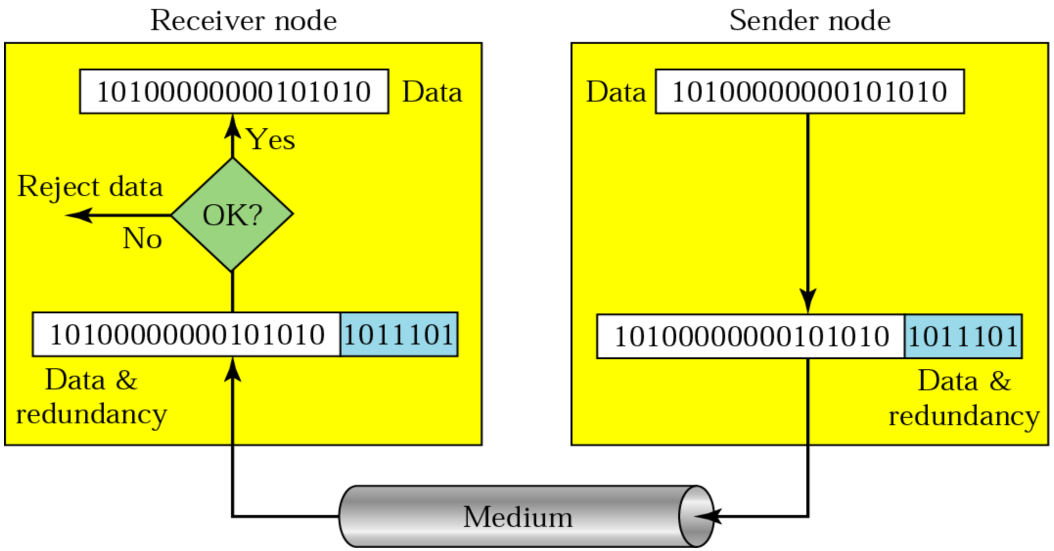
Types of Errors: Burst Error

A burst error means that 2 or more bits in the data unit have changed



* Figure is courtesy of B. Forouzan

Redundancy



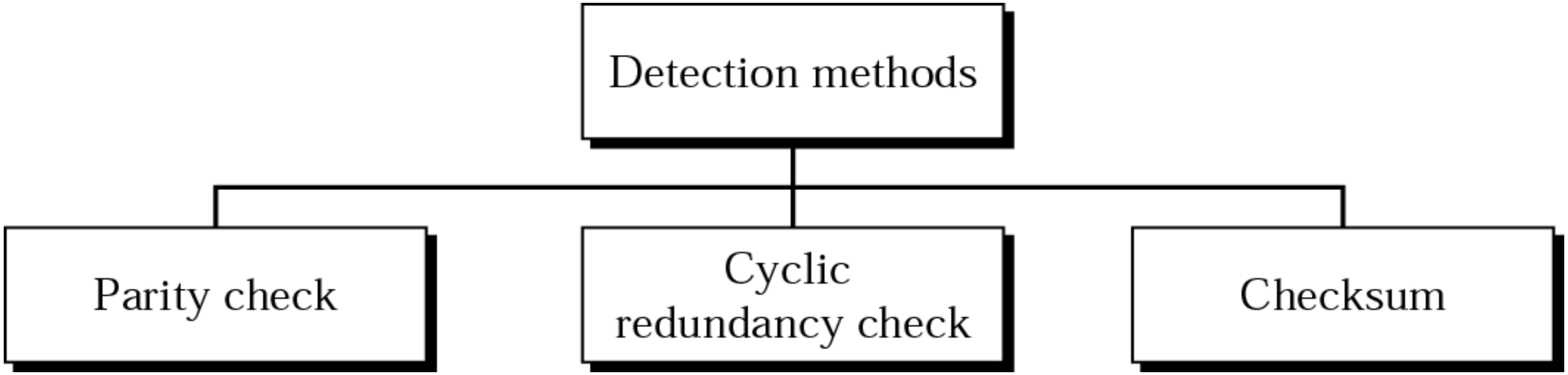
– Sender includes additional information

– Receiver verifies this information

– Example: Meet Thursday, 26th Sep   (→ Wrong!)

* Figure is courtesy of B. Forouzan

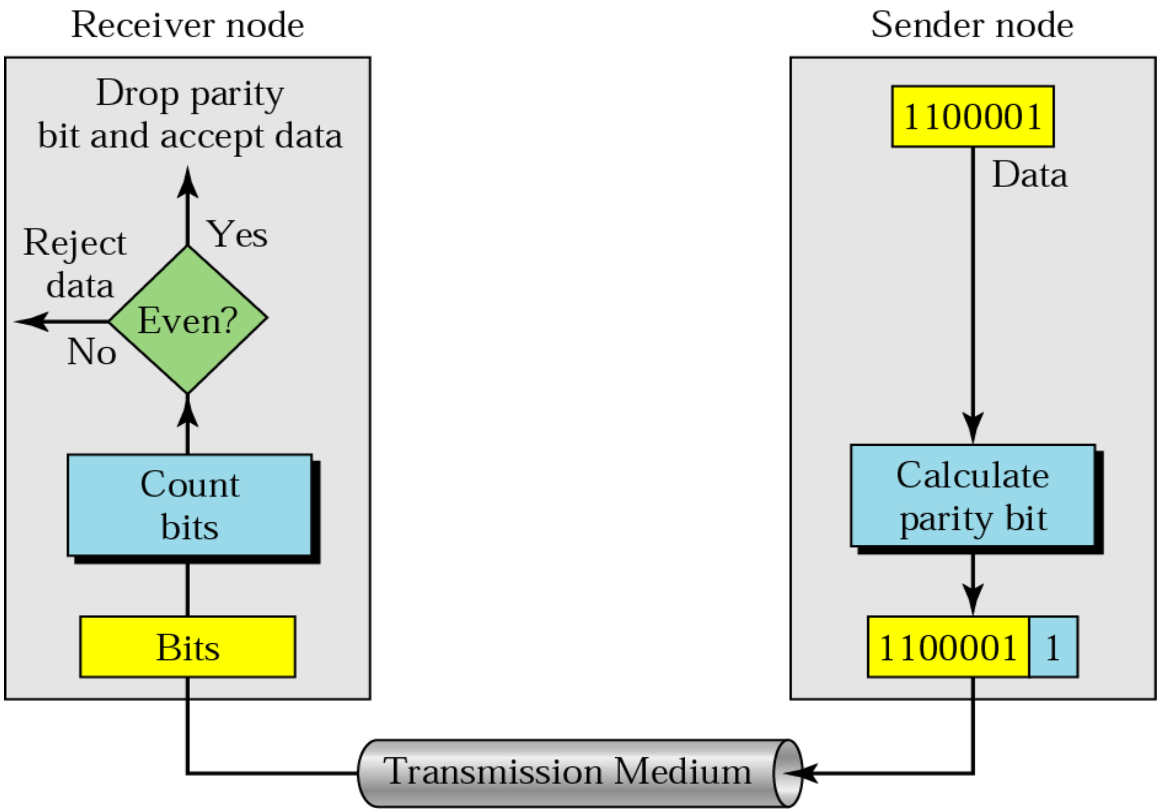
Detection of Errors



- Types of detection methods
  - Balance **detection against overhead**

* Figure is courtesy of B. Forouzan

Even-Parity Concept

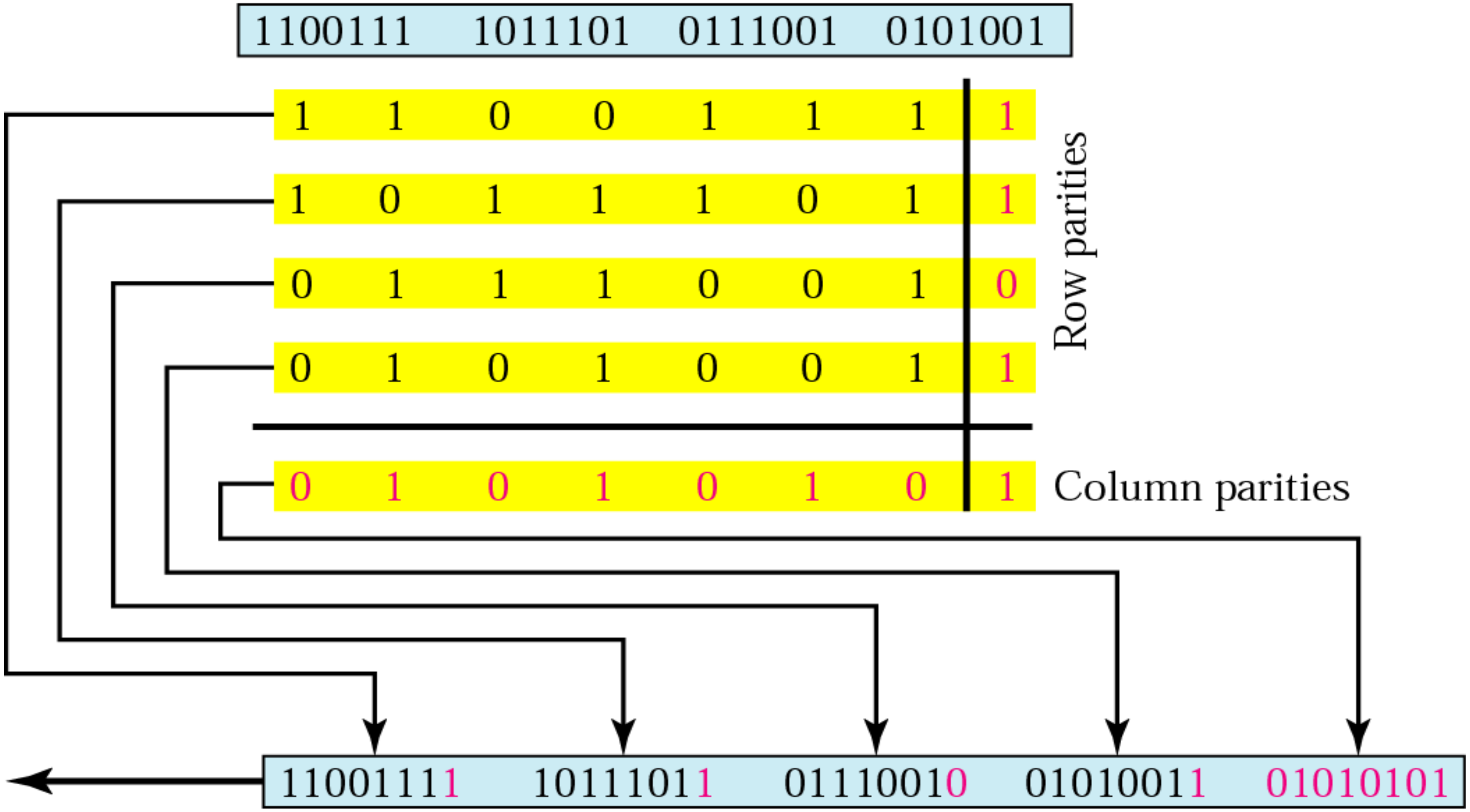


A parity bit is added to every data unit so that the total number of 1s is even (or odd for odd-parity).

* Figure is courtesy of B. Forouzan

Two-Dimensional Parity Check

In two-dimensional parity check, a block of bits is divided into rows and a redundant row of bits is added to the whole block.



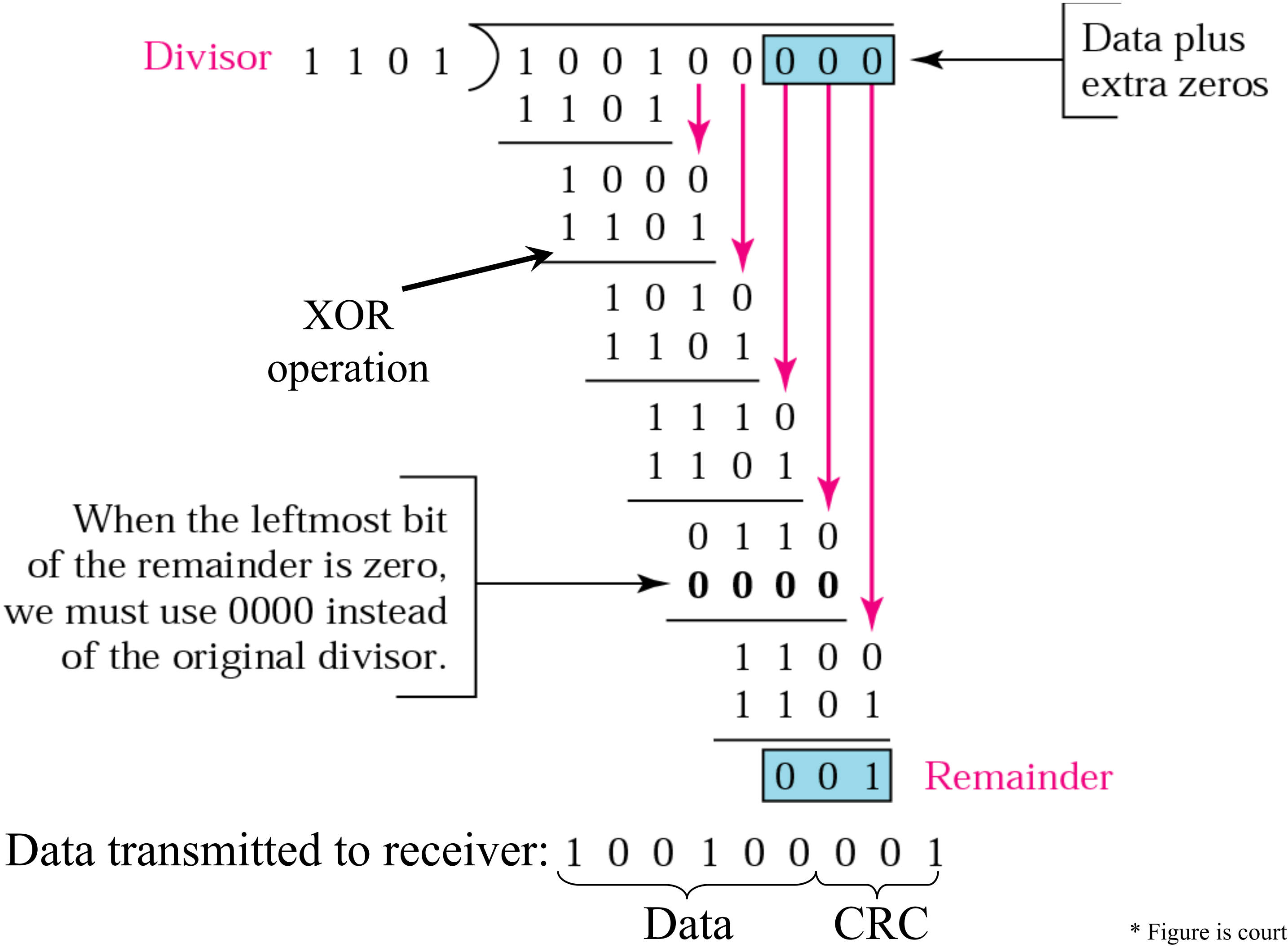
1100111 1011101 0111001 0101001

1	1	0	0	1	1	1	1	
1	0	1	1	1	0	1	1	Row parities
0	1	1	1	0	0	1	0	
0	1	0	1	0	0	1	1	
0	1	0	1	0	1	0	1	Column parities

11001111 10111011 01110010 01010011 01010101

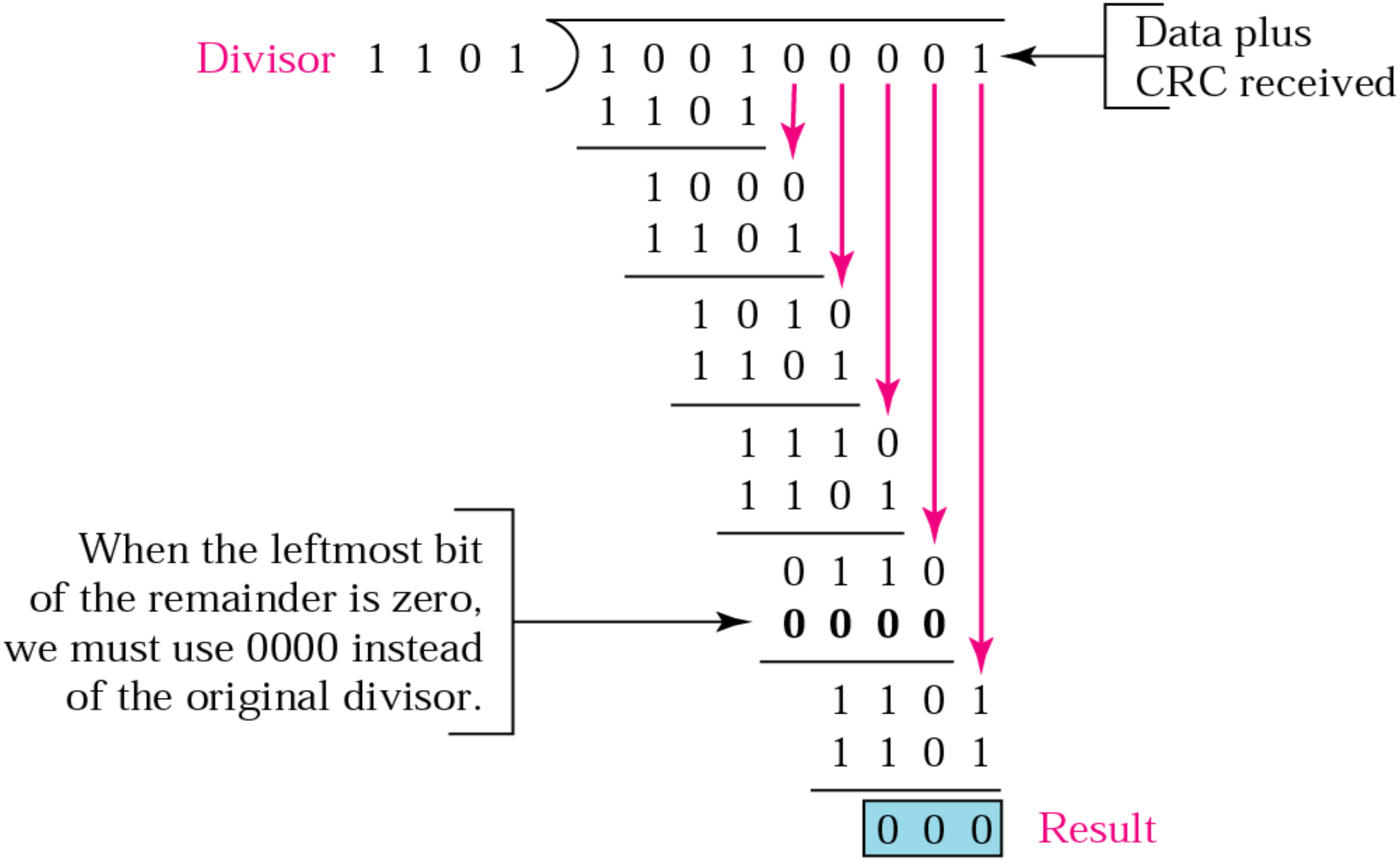
* Figure is courtesy of B. Forouzan

CRC: Sender



* Figure is courtesy of B. Forouzan

CRC: Receiver



Divisor 1 1 0 1

Data plus CRC received

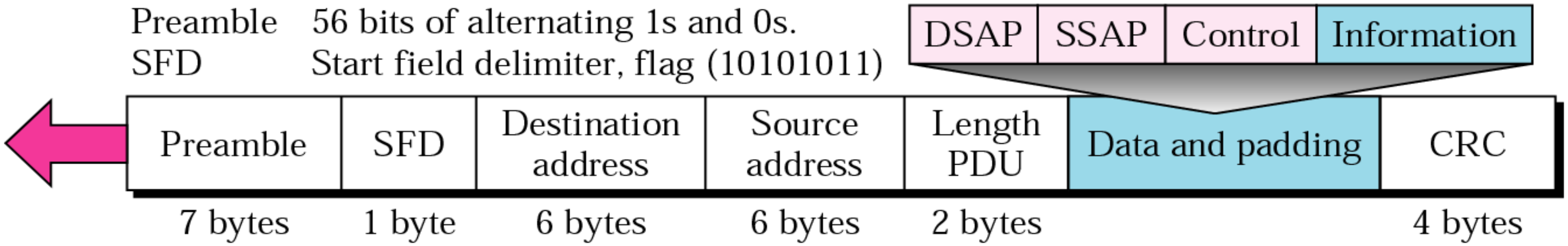
```
          1 0 0 1 0 0 0 0 1
          1 1 0 1
            1 0 0 0
            1 1 0 1
              1 0 1 0
              1 1 0 1
                1 1 1 0
                1 1 0 1
                  0 1 1 0
                  0 0 0 0
                    1 1 0 1
                    1 1 0 1
                      0 0 0
```

When the leftmost bit of the remainder is zero, we must use 0000 instead of the original divisor.

Result

* Figure is courtesy of B. Forouzan

802.3 MAC Format



- 64-bit frame preamble (10101010) used to synchronize reception
  - 7 bit preamble (10101010) + 1 start flag (10101011)

- Maximum frame length: 1536 bytes

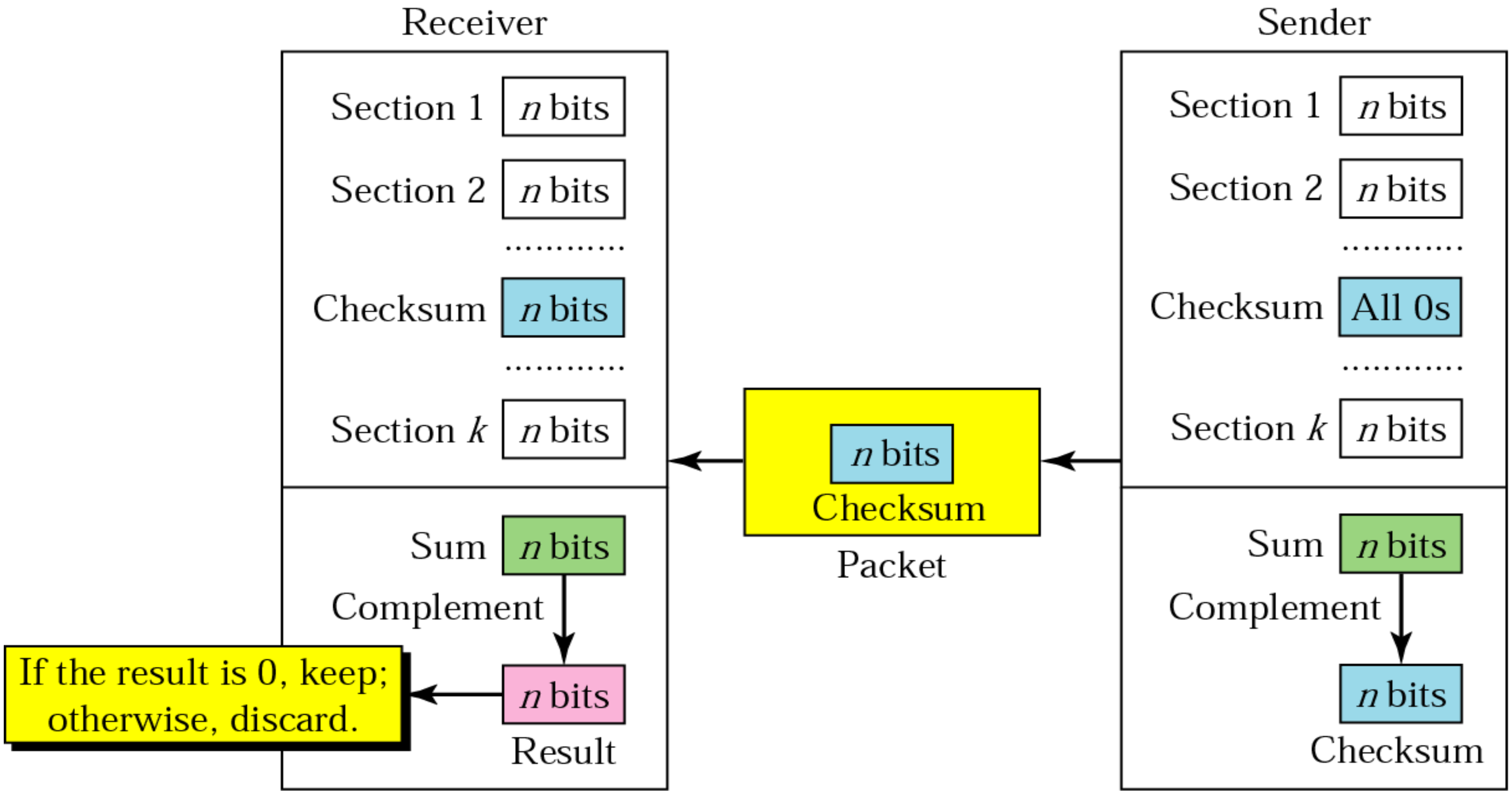
  ➪ max 1500 bytes payload

- Minimum frame length: 64 bytes

  ➪ min 46 bytes payload

* Figure is courtesy of B. Forouzan

Checksum



* Figure is courtesy of B. Forouzan

Checksum II

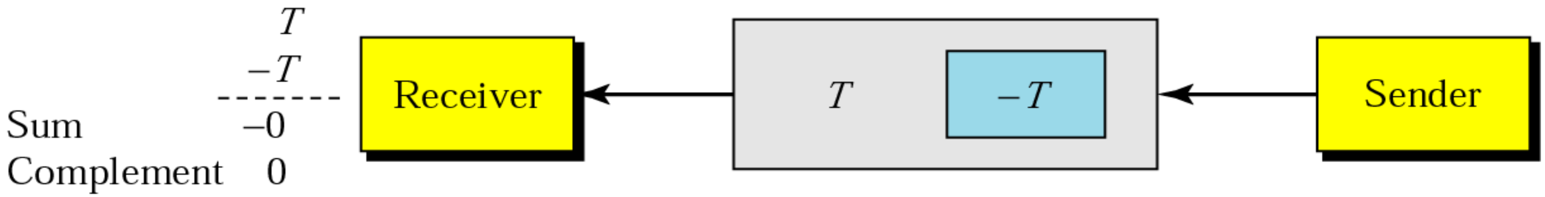
Sender:

The unit is divided into k sections, each of n bits.

All sections are added using one's complement to get the sum.

The sum is complemented and becomes the checksum.

The checksum is sent with the data.



Receiver:

The unit is divided into k sections, each of n bits.

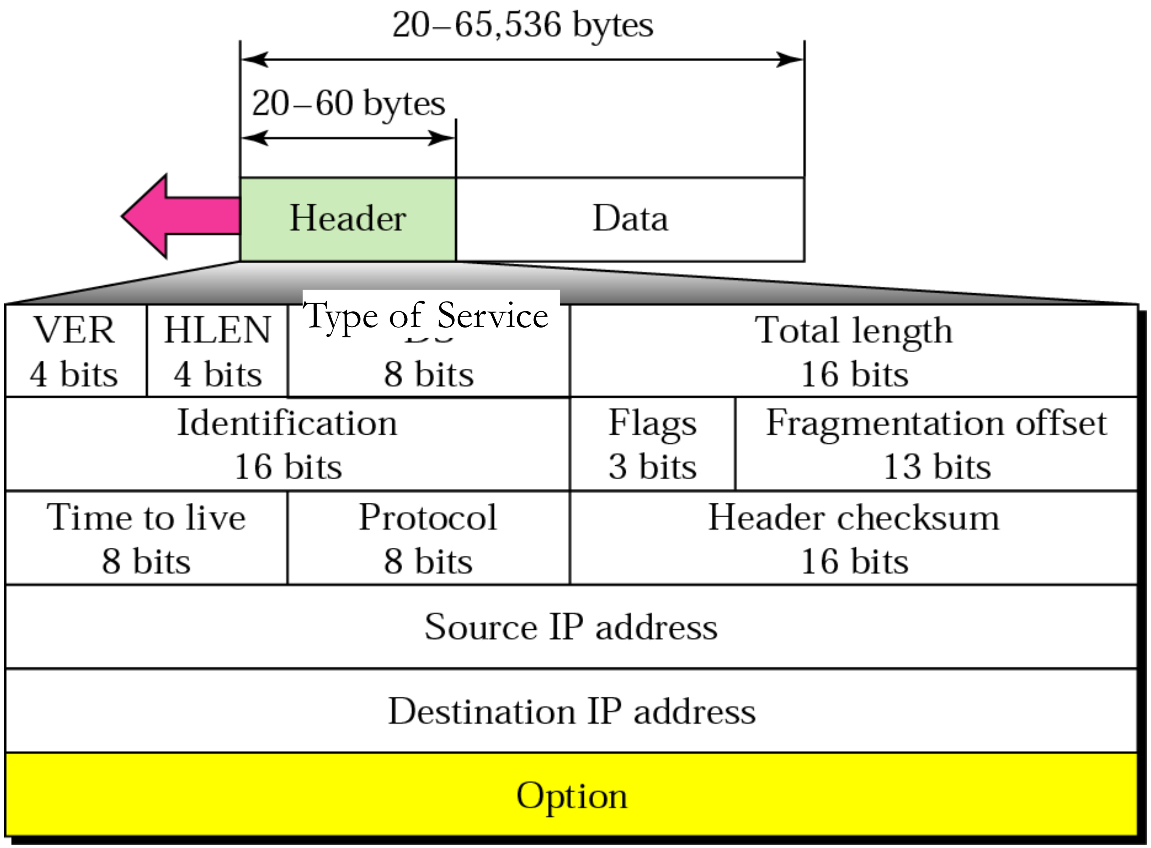
All sections are added using one's complement to get the sum.

The sum is complemented.

If the result is zero, the data are accepted: otherwise, rejected.

* Figure is courtesy of B. Forouzan

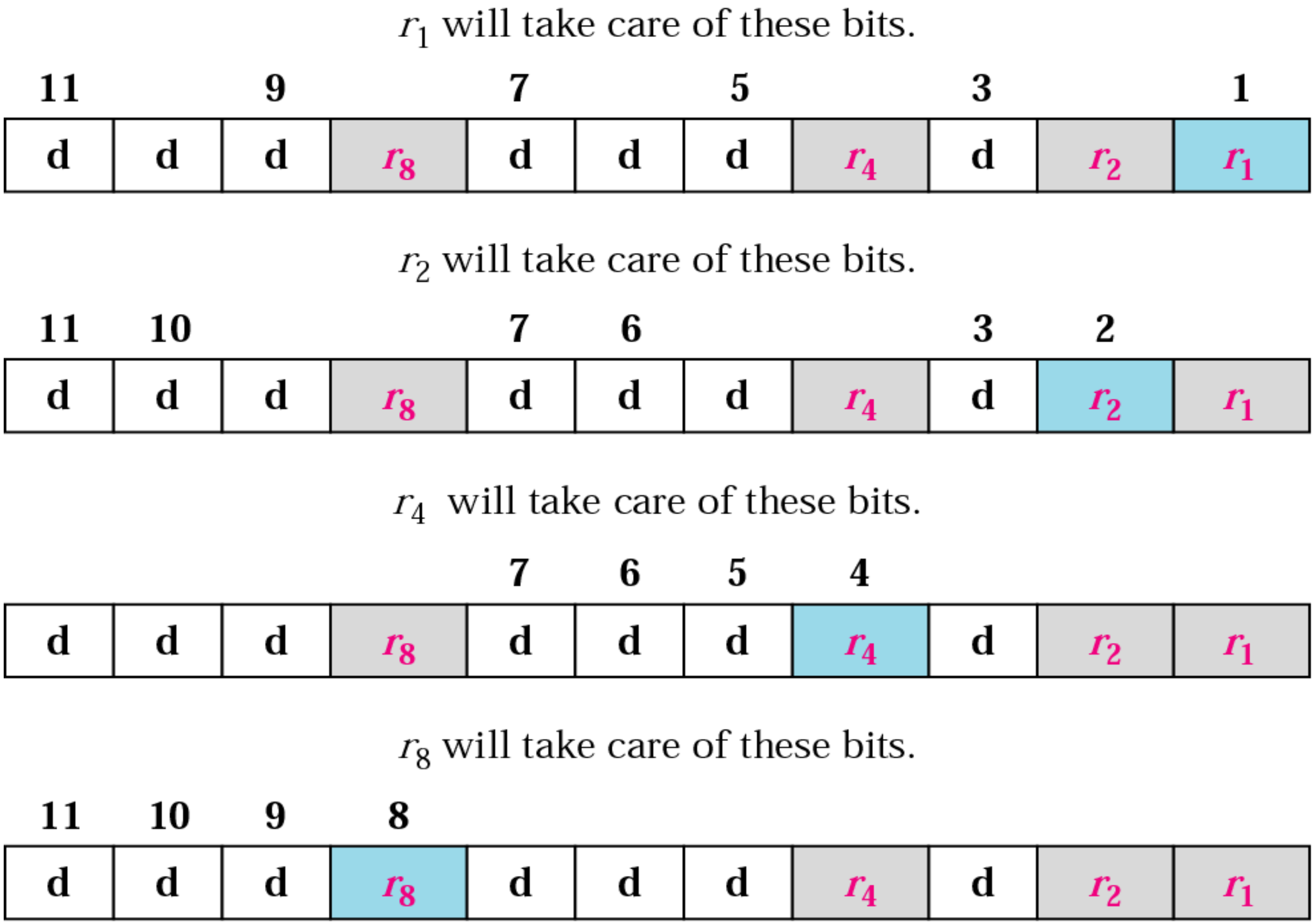
IP Datagram: Example for Checksum



VER 4 bits	HLEN 4 bits	Type of Service 8 bits	Total length 16 bits
Identification 16 bits		Flags 3 bits	Fragmentation offset 13 bits
Time to live 8 bits	Protocol 8 bits	Header checksum 16 bits	
Source IP address			
Destination IP address			
Option			

* Figure is courtesy of B. Forouzan

Redundancy Bits Calculation



r_1 will take care of these bits.

11 9 7 5 3 1
d d d r_8 d d d r_4 d r_2 r_1

r_2 will take care of these bits.

11 10 7 6 3 2
d d d r_8 d d d r_4 d r_2 r_1

r_4 will take care of these bits.

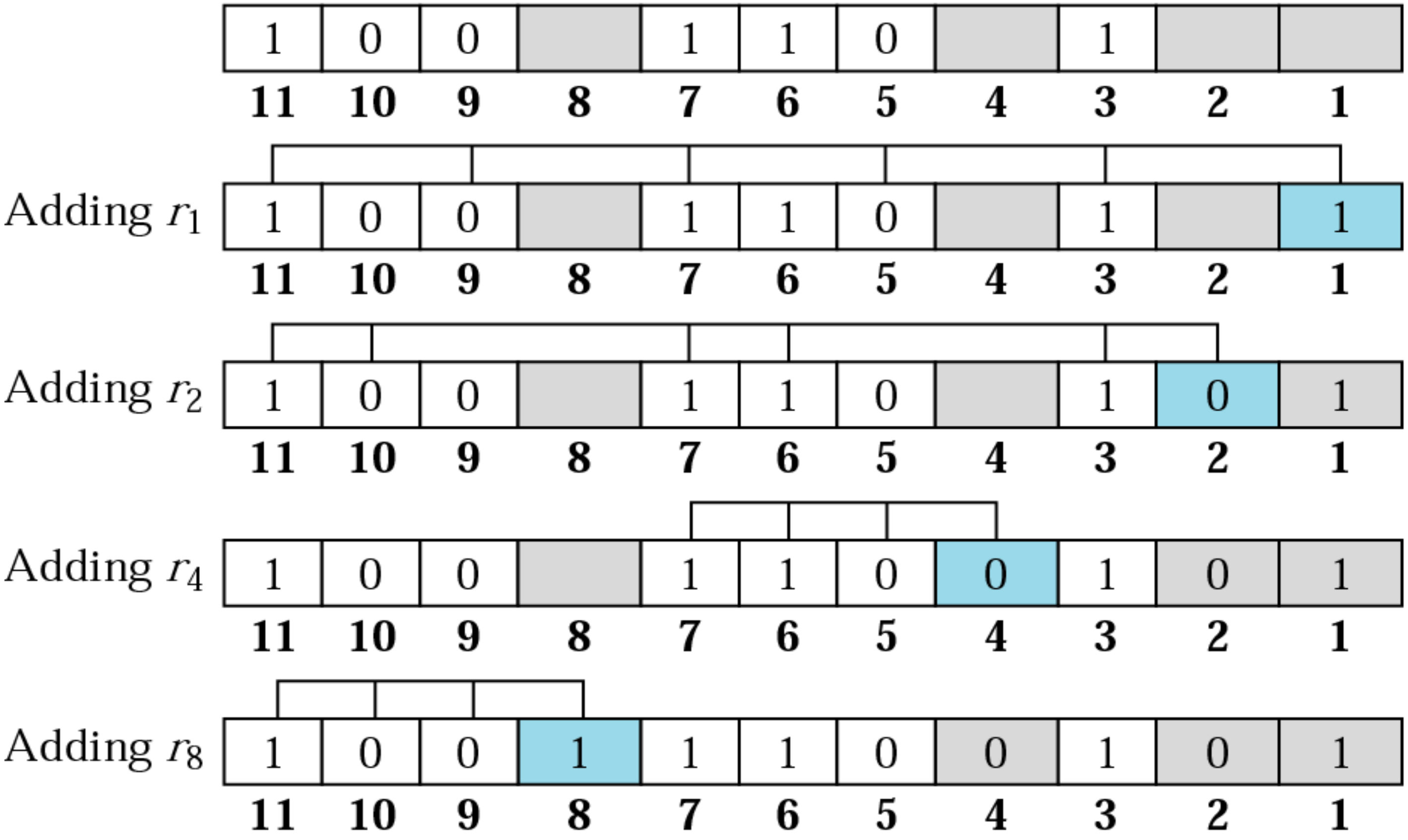
7 6 5 4
d d d r_8 d d d r_4 d r_2 r_1

r_8 will take care of these bits.

11 10 9 8
d d d r_8 d d d r_4 d r_2 r_1

* Figure is courtesy of B. Forouzan

Redundancy Bit Calculation



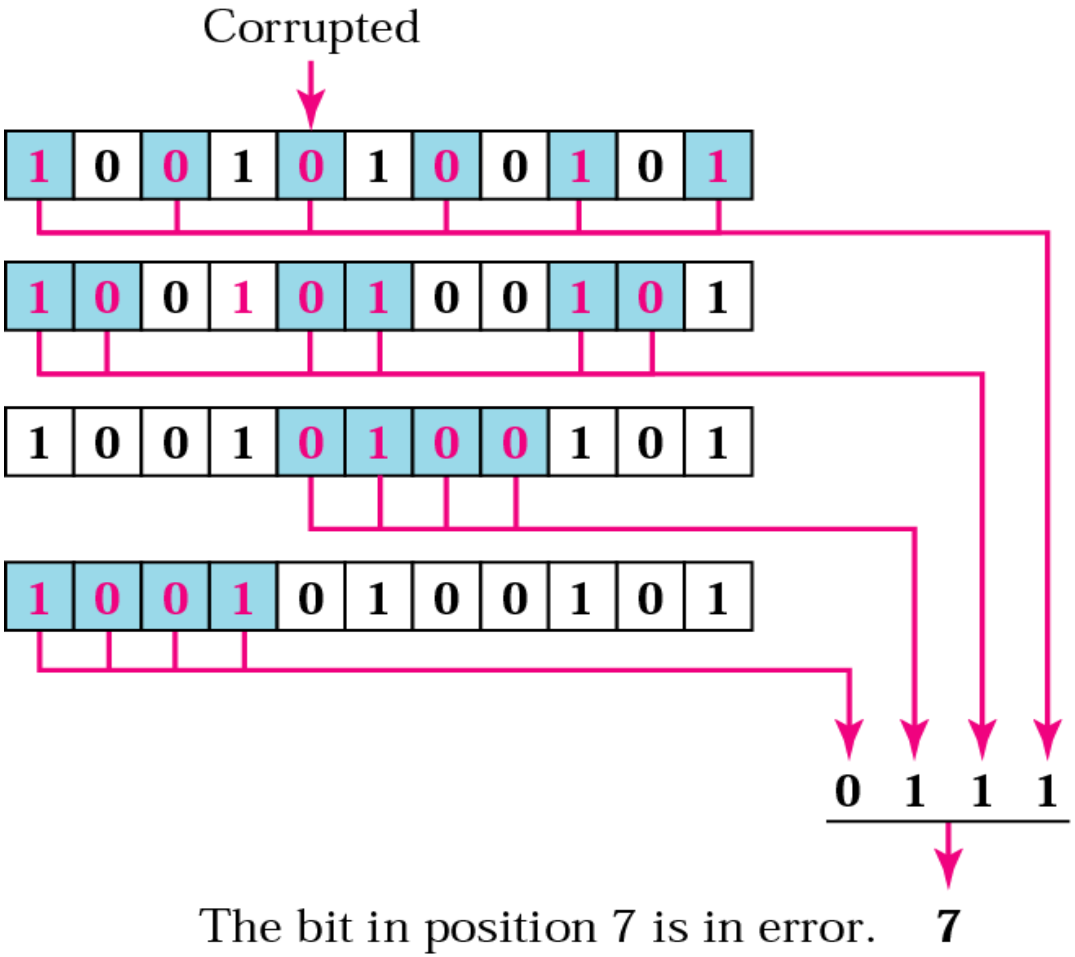
	11	10	9	8	7	6	5	4	3	2	1
	1	0	0		1	1	0		1		
Adding r_1	1	0	0		1	1	0		1		1
Adding r_2	1	0	0		1	1	0		1	0	1
Adding r_4	1	0	0		1	1	0	0	1	0	1
Adding r_8	1	0	0	1	1	1	0	0	1	0	1

Data:
1 0 0 1 1 0 1

Code:
1 0 0 1 1 1 0 0 1 0 1

* Figure is courtesy of B. Forouzan

Error Detection using Hamming Code



The bit in position 7 is in error.

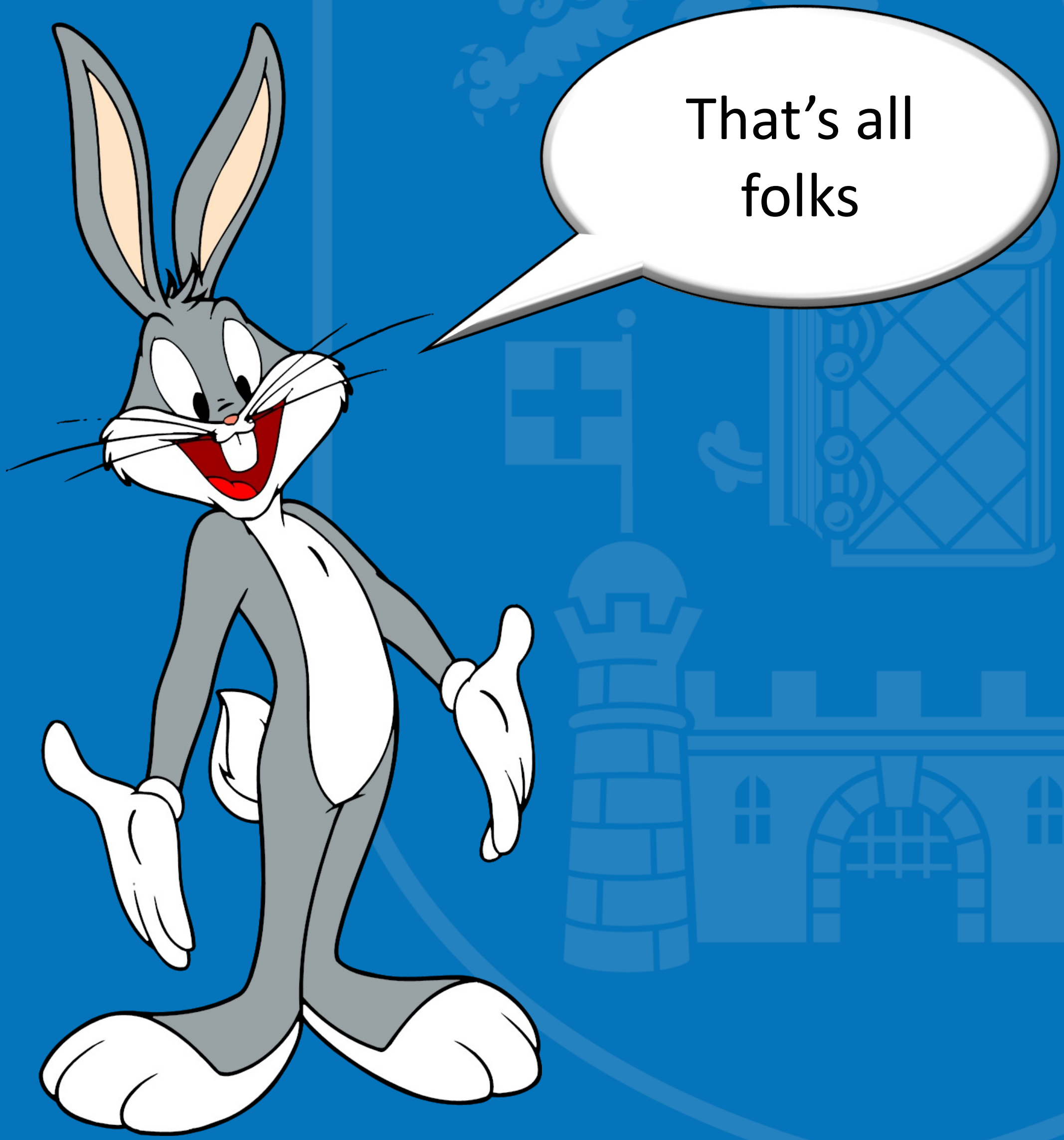
* Figure is courtesy of B. Forouzan

Summary

- Types of Errors
  - Single-Bit & Burst Errors

- Detection of Errors
  - Parity Check / 2D Parity Check
  - CRC ← Sequence of bits
  - Checksum ← Chunks of bits

- Correction of Errors
  - Error Correction by Retransmission
  - Forward Error Correction – Hamming Code


That's all folks