

# BASIC COMPUTER NETWORK

Week - 6

## Data Link Layer

Universitas Kristen Wira Wacana Sumba  
Lecturer - Fajar Hariadi

## Contents

- 1 **Encoding**
- 2 **Framing**
- 3 **Error Detection**
- 4 **Reliable Transmission**

# Pendahuluan

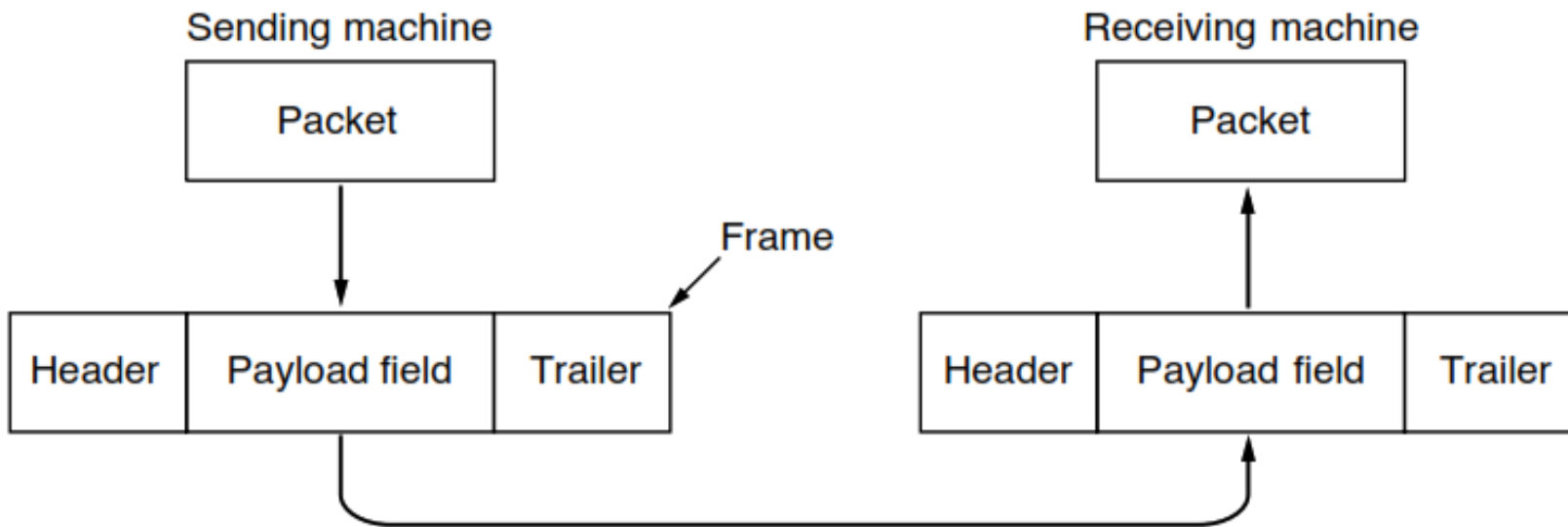
- Pada saat layer fisik sudah terhubung (kabel/nirkabel) terkadang kanal komunikasi yang dibentuk memiliki kesalahan (error) dalam mentransmisikan data
- Selain itu setiap kanal komunikasi yang digunakan memiliki batasan dalam mentransmisikan bit rate
- Selanjutnya terdapat pula delay waktu antara bit pada saat dikirim dan pada saat diterima
- Batasan-batasan ini memiliki implikasi terhadap efisiensi dalam transfer data

# Pendahuluan

- Data link layer menggunakan layanan dari physical layer di bawahnya untuk mengirim dan menerima data melalui kanal komunikasi
- Data link layer memiliki beberapa fungsi utama yaitu:
  - Menyediakan layanan interface kepada network layer
  - Mengelola frame untuk pengiriman data
  - Mendeteksi dan memperbaiki kerusakan data pada saat transmisi
  - Mengatur aliran data agar penerima yang lambat tidak dipenuhi data dari pengirim yang cepat

# Pendahuluan

- Untuk dapat bekerja sesuai dengan fungsinya, data link layer akan menerima packet dari network layer dan meng-enkapsulasi/membungkus packet menjadi sebuah frame untuk ditransmisikan
- Setiap frame akan memiliki struktur header, payload field, dan trailer

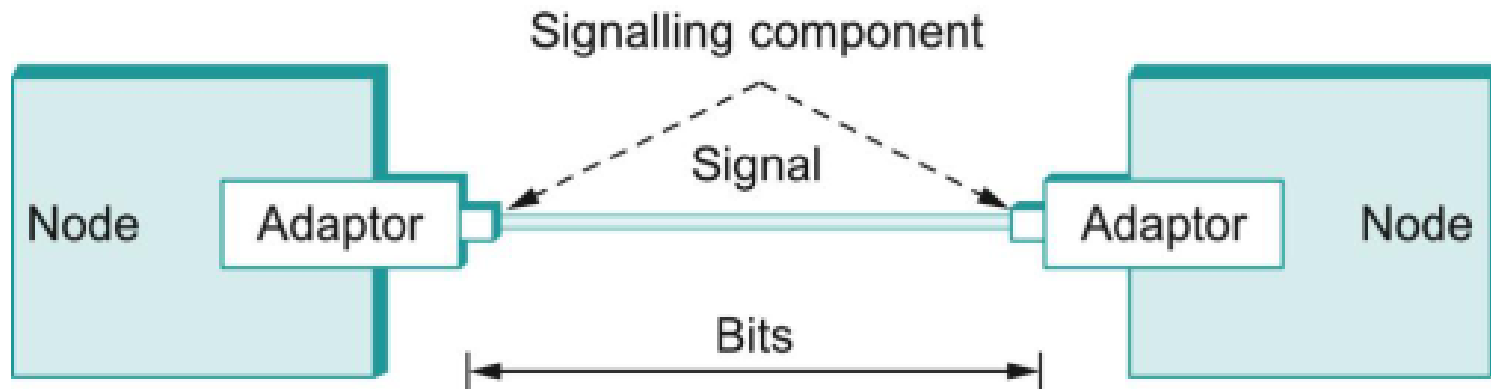


01

# Encoding

# Encoding

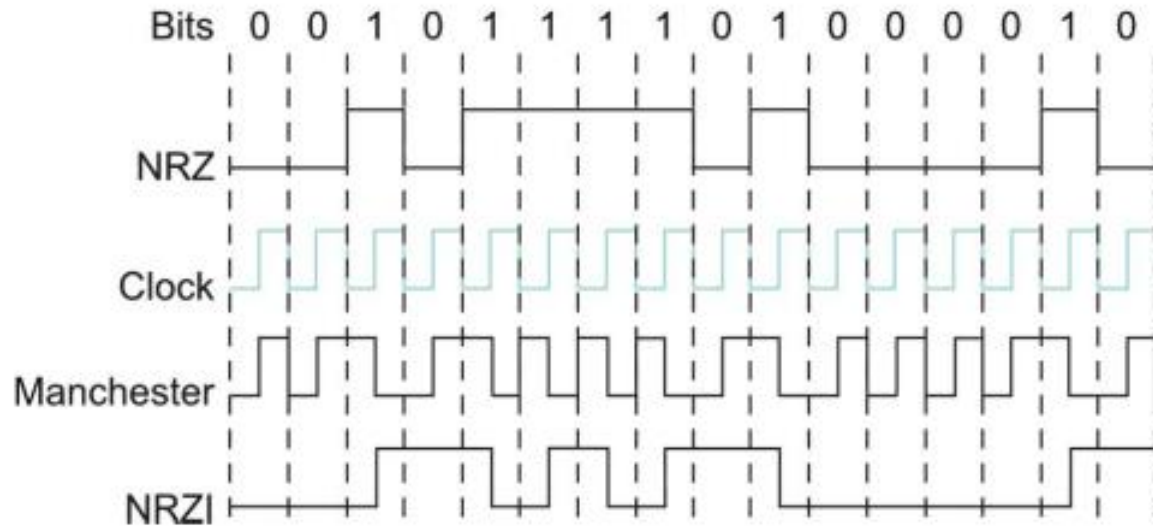
- Proses encoding merupakan proses mengkodekan sinyal dari media transmisi menjadi bit-bit data yang disebut dengan bit stream.
- Proses ini dapat berupa proses mengubah dua level tegangan pada transmisi kabel tembaga, level power yang berbeda pada kabel optic, atau dua jenis amplitudo yang berbeda pada transmisi radio



Source : Computer Networks A System Approach, Larry L. Petterson,  
Morgan Kaufmann Elsevier, 2021, page 75

# Encoding

Terdapat beberapa pendekatan dalam melakukan encoding, sama seperti pembahasan sebelumnya pada physical layer



Source : Computer Networks A System Approach, Larry L. Petterson, Morgan Kaufmann Elsevier, 2021, page 77, 79

**Table 2.2 4B/5B encoding.**

4-bit data symbol	5-bit code
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

02

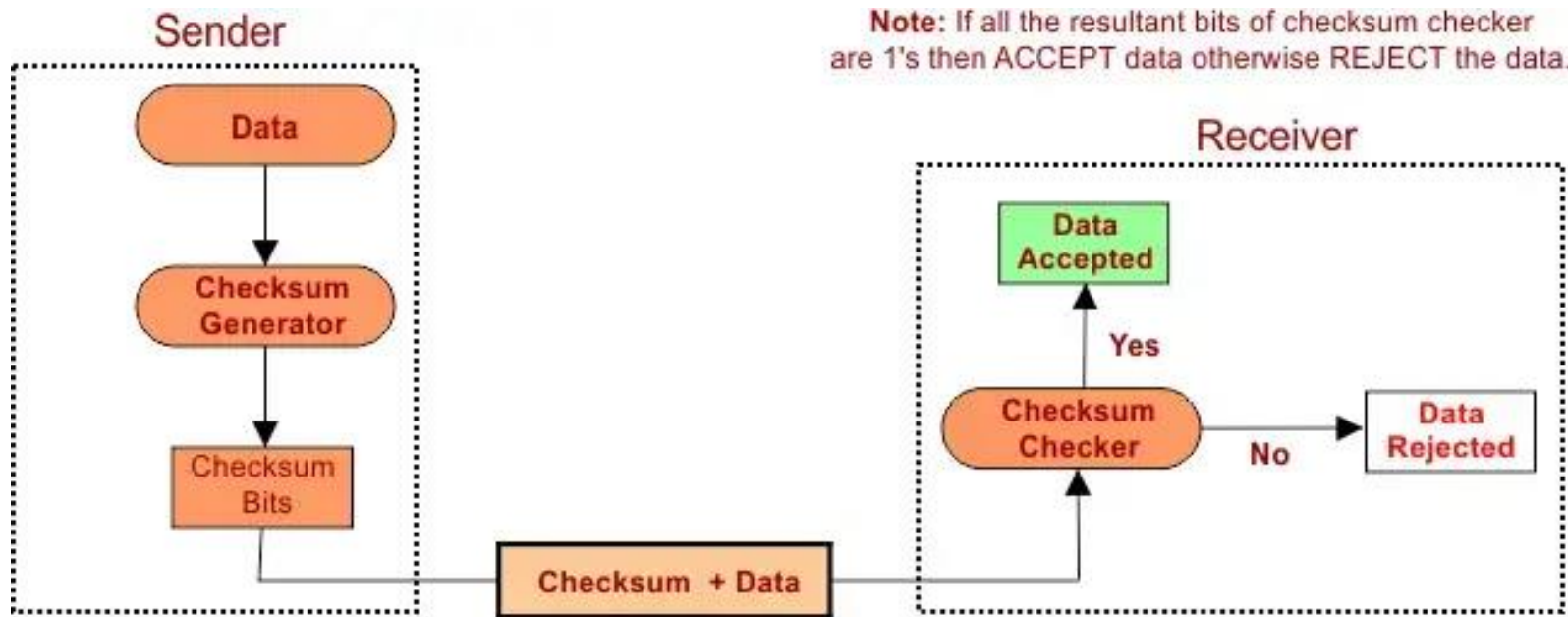
# Framing

# Framing

- Untuk dapat menyediakan layanan kepada network layer, data link layer akan menggunakan layanan dari physical layer
- Physical layer menerima bit stream dari data link dan mencoba untuk mengirim ke tujuan
- Bit stream yang diterima oleh data link layer tidak dijamin bebas dari error
- Beberapa bit bisa saja memiliki nilai yang berbeda ketika dikirim dan diterima
- Jumlah bit yang diterima dapat kurang, sama atau bahkan lebih dari jumlah yang dikirim
- Hal ini menjadi tanggung jawab dari data link layer untuk mendeteksi dan memperbaiki error

# Framing

Pendekatan yang biasa dilakukan adalah memecah bit stream menjadi frame, kemudian melakukan perhitungan checksum untuk setiap frame, dan mengirimkan frame beserta dengan hasil checksum



Source: <https://cstaleem.com/checksum-in-error-detection>

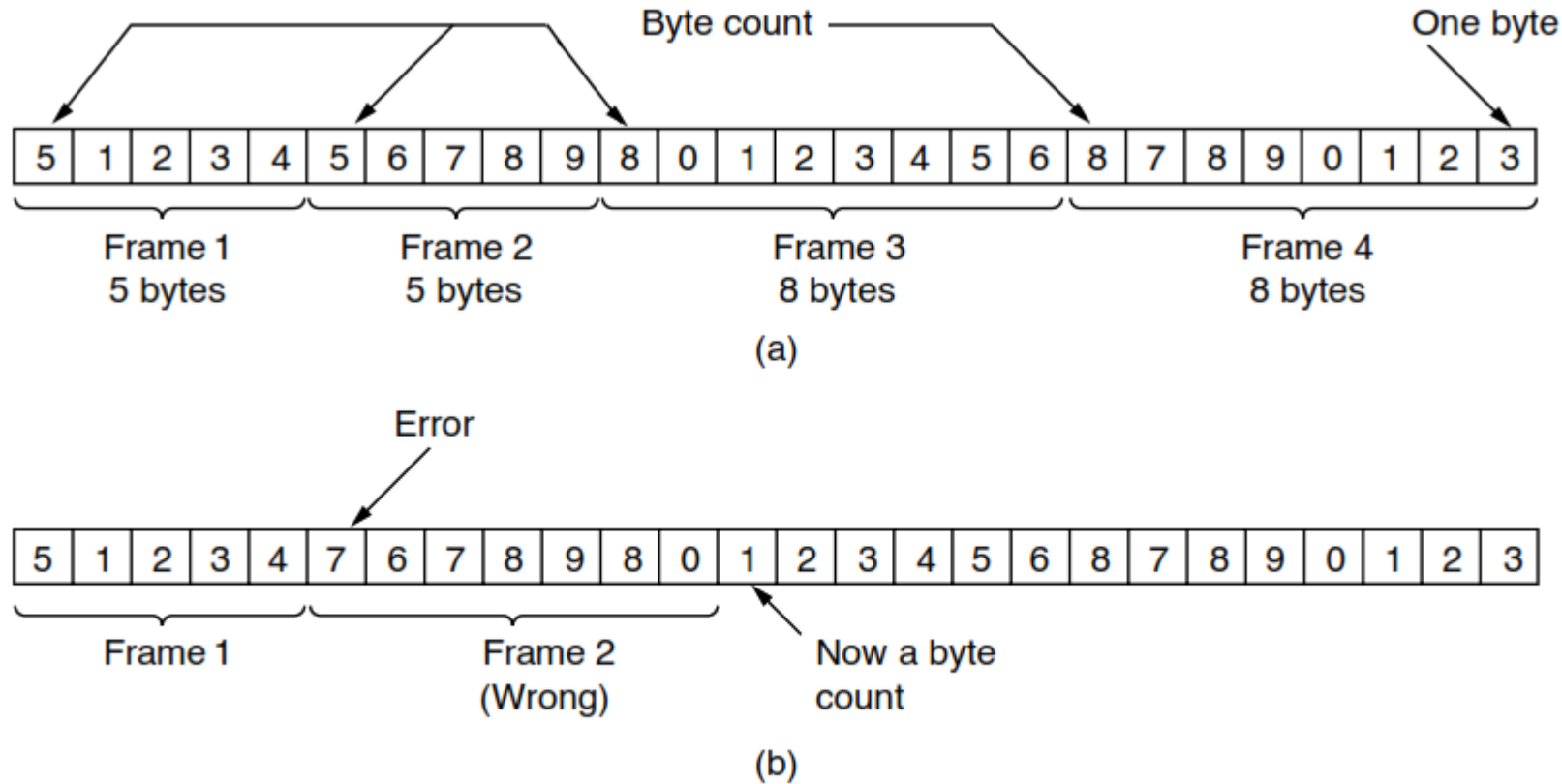
Jika terjadi kesalahan data link dapat membuang frame yang rusak atau mengirimkan laporan error kepada pengirim

Sampai ke bagian penerima dilakukan perhitungan checksum ulang, jika hasil checksum yang dilakukan berbeda dengan data checksum yang diterima maka komputer penerima tahu bahwa ada kesalahan pada saat transmisi data

# Framing

- Memecah bit stream menjadi frame memerlukan cara yang dapat memudahkan penerima dalam mengetahui awal sebuah frame dan akhir sebuah frame dari bit stream yang diterima.
- Pendekatan pertama adalah byte count, metode ini menggunakan header yang berisi jumlah spesifik byte dalam sebuah frame sehingga ketika penerima menerima frame dan membaca headernya, bagian penerima mengetahui panjang byte dari sebuah frame, dan bit berikutnya berarti sudah frame baru

# Framing



**Figure 3-3.** A byte stream. (a) Without errors. (b) With one error.

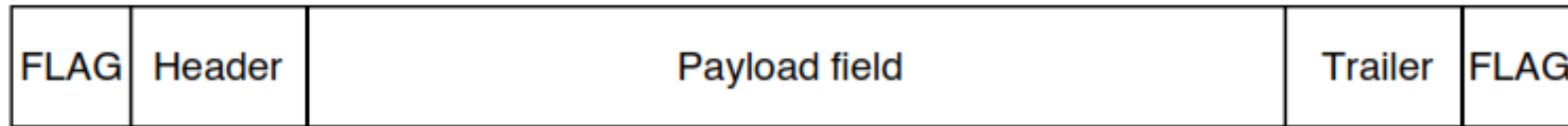
Source : Computer Networks, Andrew S. Tanenbaum, Pearson, 2021, page 206

Metode ini memiliki kelemahan dalam handle frame yang error pada saat transmisi

Contohnya ada pada bagian (b) jika ada frame yang error maka frame berikutnya tidak dapat dibaca dengan baik

# Framing

- Metode kedua dalam membentuk frame digunakan untuk menyelesaikan masalah resynchronization/ sinkronisasi ulang apabila terdapat frame error dengan menggunakan awal dan akhir frame menggunakan byte spesial yang disebut dengan flag byte



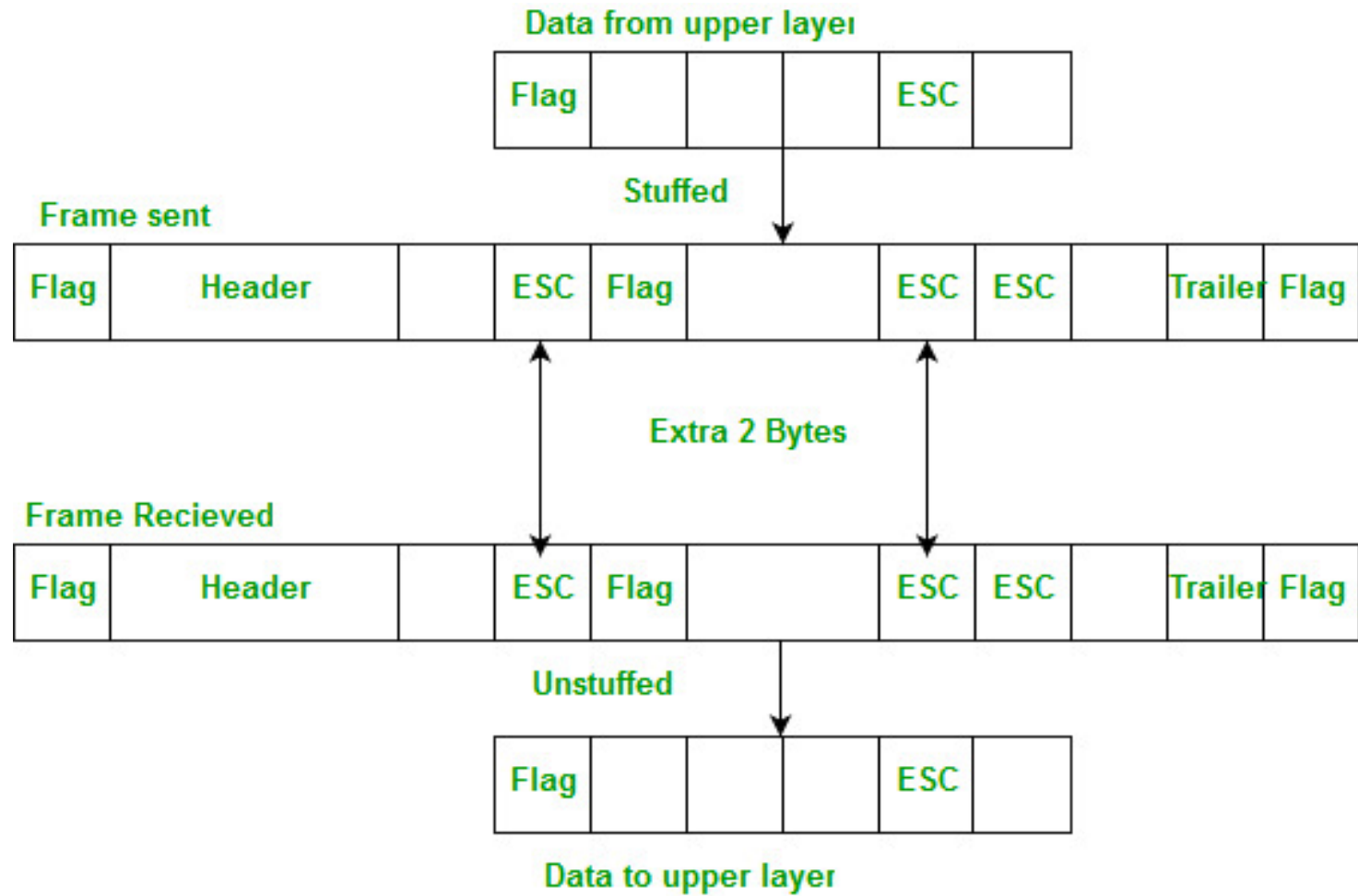
Source : Computer Networks,  
Andrew S. Tanenbaum, Pearson,  
2021, page 207

- Flag ini mengindikasikan awal dan akhir sebuah frame dan menandakan awal frame berikutnya, sehingga ketika terjadi error pada bagian penerima, sinkronisasi dapat dilakukan dengan mencari data 2 buah flag berurutan yang digunakan untuk menandakan akhir dari frame dan awal dari frame baru

# Framing

- Kelemahan penambahan flag terjadi ketika terdapat pola bit yang sama dengan byte flag pada pertengahan data yang dikirim, hal ini akan memungkinkan terjadinya kesalahan pembacaan pola frame
- Untuk menangani hal tersebut dilakukan penambahan byte yang biasanya menggunakan deret bit karakter ESC sebelum bagian yang memiliki pola sama dengan flag di tengah data
- Selain pola bit flag, pola bit ESC di tengah data juga akan diberikan tambahan ESC
- Pada sisi penerima pola bit ESC ini dihapus, sehingga data yang diperoleh sama dengan data yang dikirim metode ini disebut dengan byte stuffing

# Framing

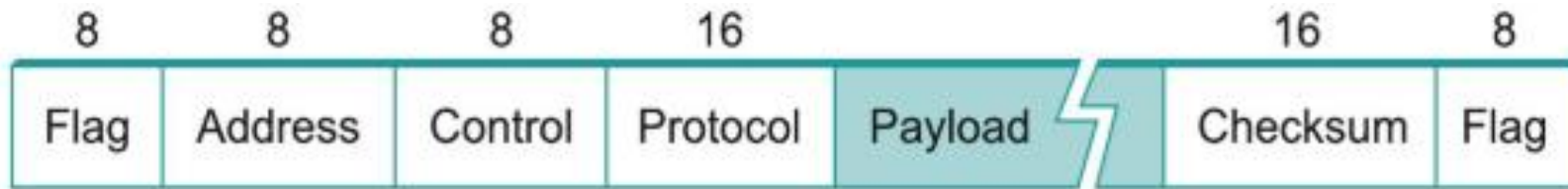


Penambahan ESC dilakukan dari sisi pengirim

Pada sisi penerima ESC yang telah ditambahkan akan dihapus

# Framing

Protocol yang menggunakan byte stuffing adalah Point-to-Point Protocol (PPP)



Source : Computer Networks A System Approach, Larry L. Petterson, Morgan Kaufmann Elsevier, 2021, page 80

- Flag berisi pola bit 01111110 - default
- Address berisi pola bit 11111111 - default
- Control berisi pola bit 00000011 - default
- Protocol merupakan informasi pada network layer, layanan mana dari network layer yang menerima frame
- Payload/information berisi data yang dikirimkan, panjangnya bervariasi namun maksimal 1500 bytes
- Checksum / CRC digunakan untuk mengidentifikasi frame error

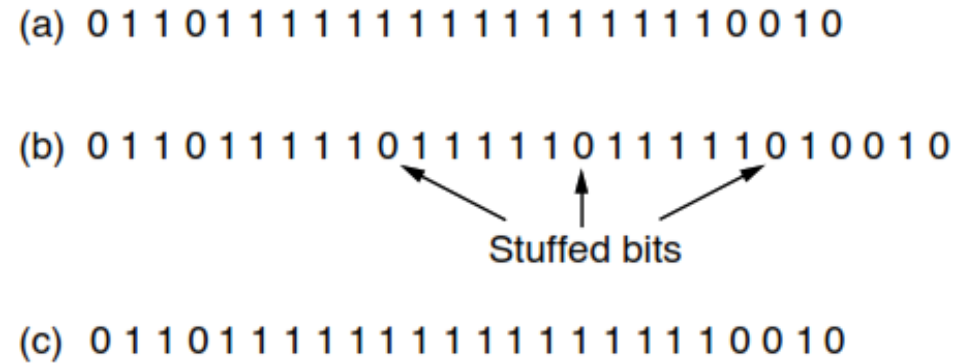
# Framing

- Pendekatan berikutnya dalam membentuk framing mengatasi permasalahan byte stuffing yaitu pola pengiriman yang terikat pada byte (8 bit) data.
- Pembuatan frame dapat dilakukan berdasarkan bit, pola kerjanya sama seperti byte stuffing hanya saja dilakukan di level bit sehingga ukuran frame dapat lebih fleksibel
- Protocol yang menggunakan bit stuffing adalah protocol High-Level Data Link Control (HDLC)

# Framing

➤ Pada bagian pengirim, apabila menemukan 5 bit (11111) muncul secara berurutan maka secara otomatis akan ditambahkan bit 0 baru kemudian dilanjutkan bit berikutnya

➤ Pada sisi penerima ketika melihat ada 5 bit secara berurutan, bit 0 berikutnya akan dihapus dan bit selanjutnya akan dianggap sebagai lanjutan dari data

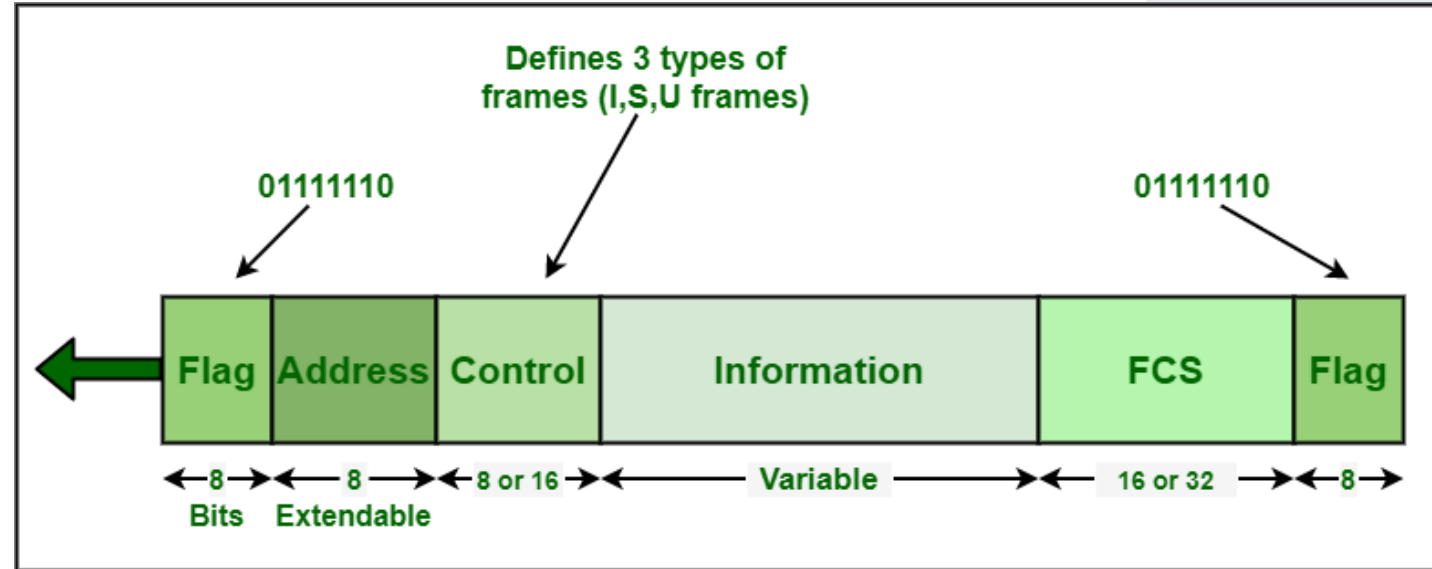


**Figure 3-5.** Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.

Source : Computer Networks, Andrew S. Tanenbaum, Pearson, 2021, page 208

# Framing

➤ Pada bagian pengirim, apabila menemukan 5 bit (11111) muncul secara berurutan maka secara otomatis akan ditambahkan bit 0 baru kemudian dilanjutkan bit berikutnya



Source : <https://www.geeksforgeeks.org/basic-frame-structure-of-hdlc/>

➤ Pada sisi penerima ketika melihat ada 5 bit secara berurutan, bit 0 berikutnya akan dihapus dan bit selanjutnya akan dianggap sebagai lanjutan dari data

03

# Error Control

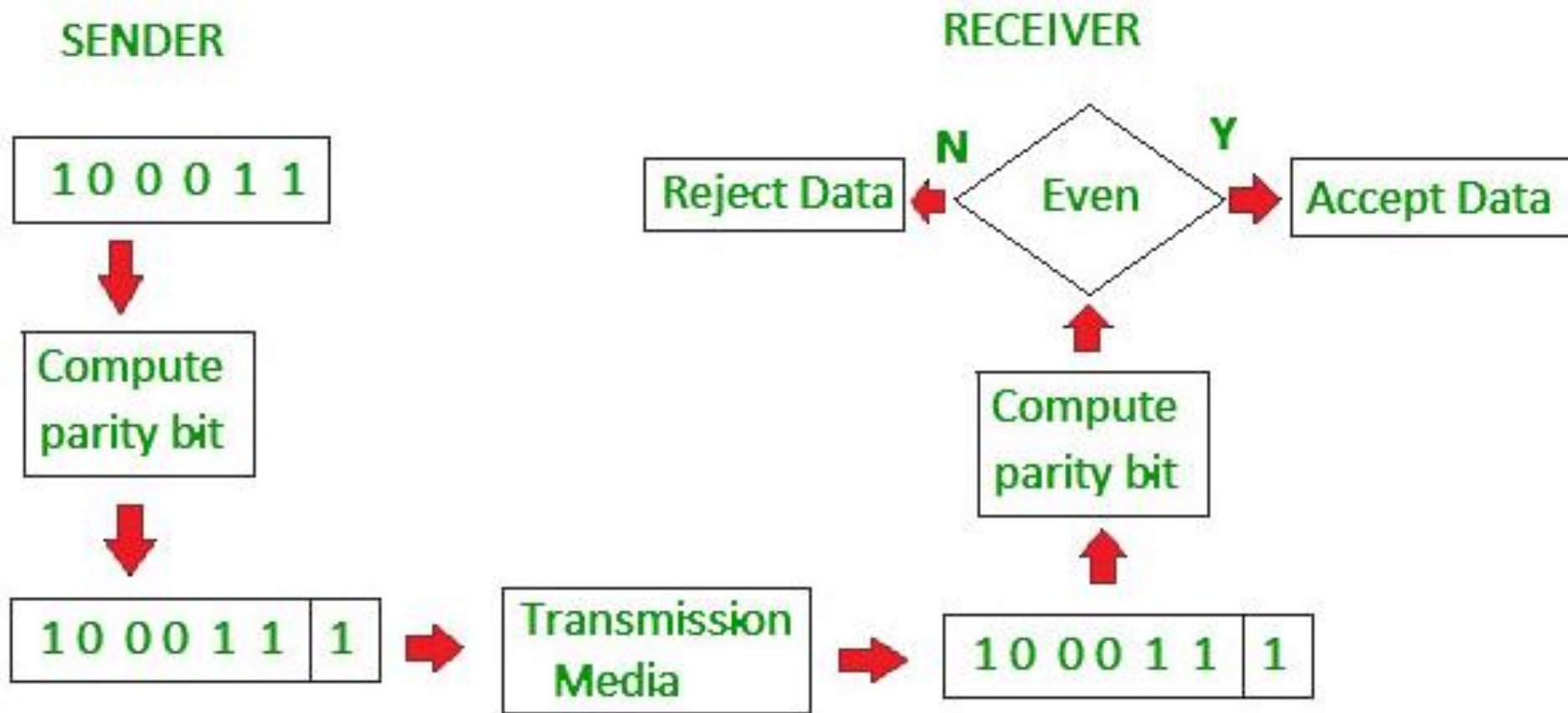
# Error Control

- Error pada frame pada umumnya terjadi ketika bit stream di transmisikan
- Terdapat beberapa metode yang digunakan dalam mendeteksi error, namun yang paling umum digunakan saat ini adalah Cyclic Redundancy Check (CRC)
- Namun kita akan membahas dari mulai dari metode yang paling sederhana hingga sampai metode cyclic redundancy check (CRC)

# Parity Checking

- Simple parity check dilakukan dengan menambahkan bit pada akhir data yang ditransmisikan, cara kerjanya adalah:
  - Bit 1 ditambahkan di akhir jika block bit memiliki jumlah bit 1 sebanyak angka ganjil
  - Bit 0 ditambahkan jika block bit memiliki jumlah bit 1 sebanyak angka genap
- Oleh karena cara kerjanya, metode ini juga sering disebut even parity checking

# Parity Checking



Dapat digunakan untuk mendeteksi error dengan jumlah bit kesalahan ganjil

Source: <https://www.geeksforgeeks.org/error-detection-in-computer-networks/>

# Two Dimensional Parity Checking

Original Data

10011001	11100010	00100100	10000100
----------	----------	----------	----------

Row parities

10011001	0
11100010	0
00100100	0
10000100	0
11011011	0

Column parities →

100110010	111000100	001001000	100001000	110110110
-----------	-----------	-----------	-----------	-----------

Data to be sent

Setiap baris akan dicek menggunakan simple parity check, selain baris setiap kolom akan dicek juga. Kedua hasil parity check akan dikirimkan

Tidak hanya mendeteksi, tapi dapat dipergunakan untuk memperbaiki jika hanya 1 bit error

Tapi jika yang error adalah parity bit maka skema deteksi dan perbaikan tidak dapat dilakukan

# Checksum

- Pendeteksian error menggunakan checksum melibatkan pembagian data menjadi beberapa ukuran yang sama dan melakukan komplemen/pembalikan nilai untuk mendapatkan nilai checksum
- Data kemudian dikirimkan bersamaan dengan hasil checksum
- Sampai di sisi penerima data dijumlahkan dengan hasil checksumnya, jika hasil penjumlahannya bernilai 1 semua atau hasil komplemennya bernilai 0 semua maka tidak terdapat kesalahan pengiriman data

# Checksum

Contoh data yang akan dikirimkan adalah 1010100100111001, dibagi menjadi byte (8 bit)

Sender Site:

```
10101001    subunit 1
00111001    subunit 2
11100010    sum (using 1s complement)
00011101    checksum (complement of sum)
```

Data transmitted to Receiver is:



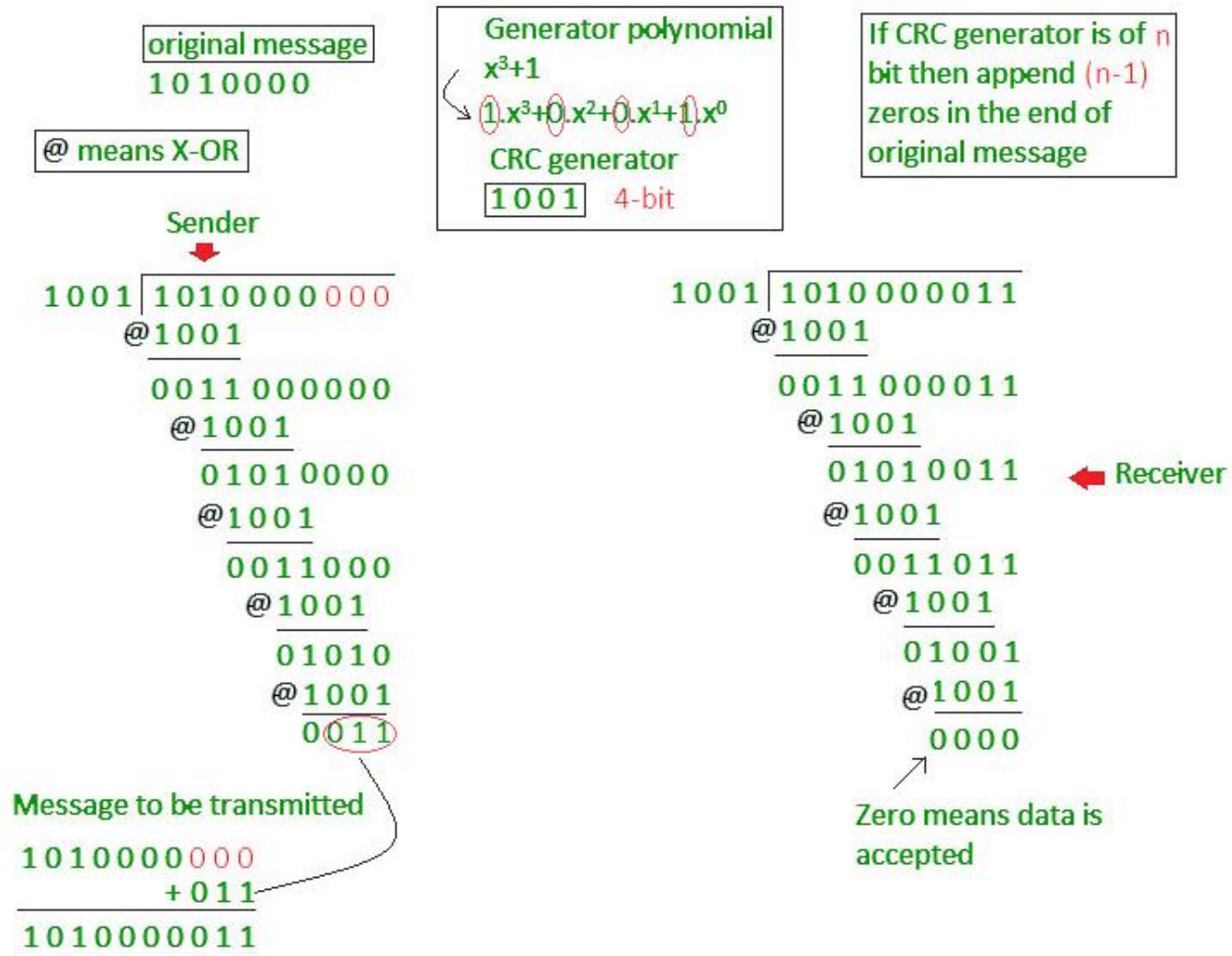
Source: <https://www.geeksforgeeks.org/error-detection-code-checksum/>

Receiver Site :

```
10101001    subunit 1
00111001    subunit 2
00011101    checksum
11111111    sum
00000000    sum's complement
```

**Result is zero, it means no error.**

# Cyclic Redundancy Check (CRC)



- Pesan asli ditambahkan bit 0 sejumlah bit key -1 di bagian akhir
- Lakukan modulo 2 division menggunakan XOR pada bit data yang ditambahkan 0
- Gantikan penambahan bit 0 dengan Sisa hasil bagi / remainder
- Pada sisi penerima lakukan hal yang sama namun menggunakan bitstream yang diterima, jika remainder bernilai 0 semua, maka data tidak memiliki error

04

# Reliable Transmission

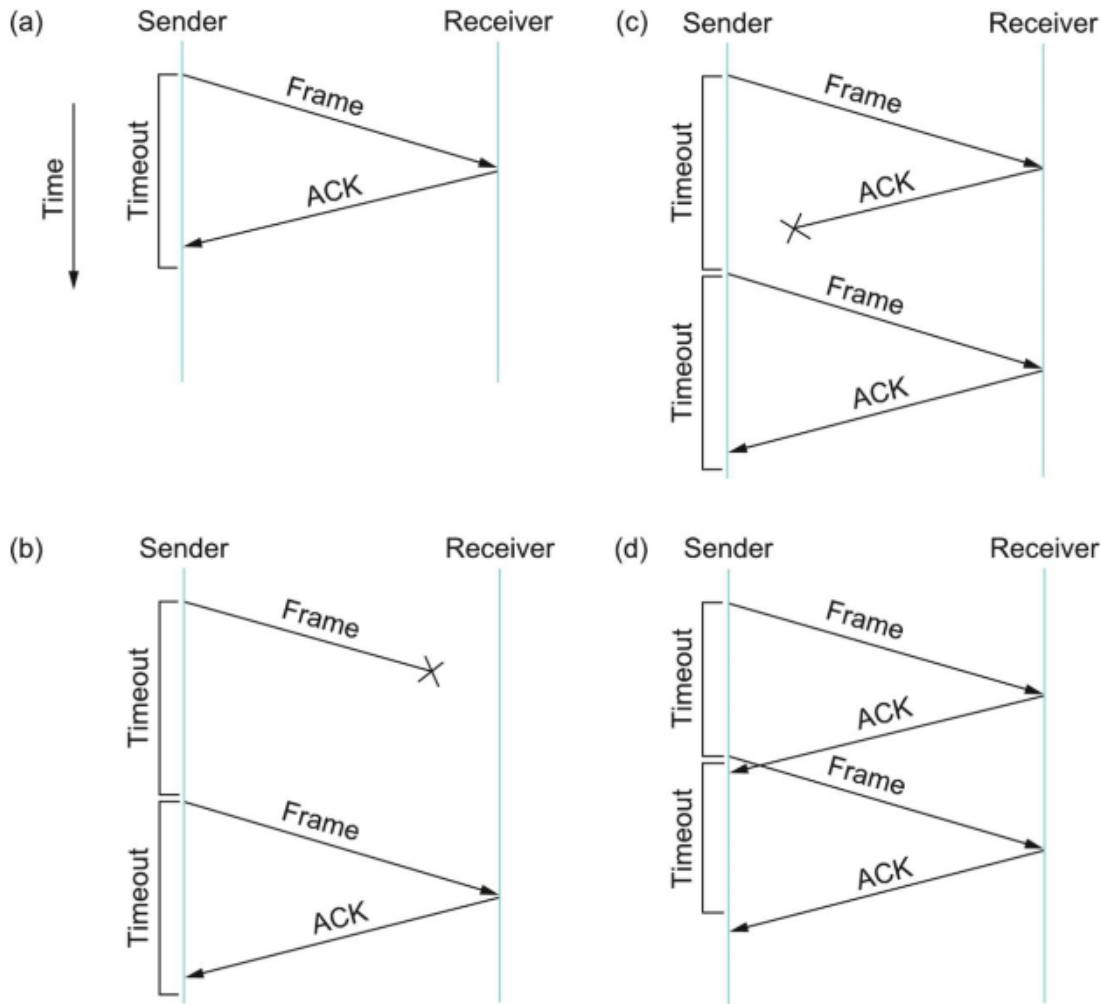
# Reliable Transmission

- CRC dapat digunakan untuk mendeteksi error, namun untuk menjamin saluran komunikasi yang handal perlu ada penanganan ketika dideteksi terjadinya error tersebut
- Pengiriman yang reliable dapat diperoleh melalui dua mekanisme dasar yaitu acknowledgments dan timeout
- Acknowledgments (ACK) merupakan frame yang digunakan penerima untuk memberitahukan kepada pengirim bahwa data yang dikirim telah diterima
- Jika pengirim tidak menerima pesan ACK ini dalam rentang waktu yang telah ditentukan maka pengirim akan mengirimkan ulang pesan yang sebelumnya telah dikirim.
- Tindakan menunggu waktu penerimaan ACK ini yang disebut dengan timeout

# Reliable Transmission

- Strategi penggunaan ack dan timeout ini sering disebut dengan Automatic Repeat Request (ARQ).
- Terdapat beberapa algoritma dalam mengimplementasikan ARQ, algoritma pertama adalah Stop and Wait
- Stop and wait bekerja dengan memastikan pengirim menerima pesan ACK terlebih dahulu baru mengirimkan frame berikutnya
- Jika pesan ACK tidak diterima dalam rentang waktu timeout, maka pengirim akan mengirimkan ulang pesan yang sebelumnya dikirim

# Stop and Wait



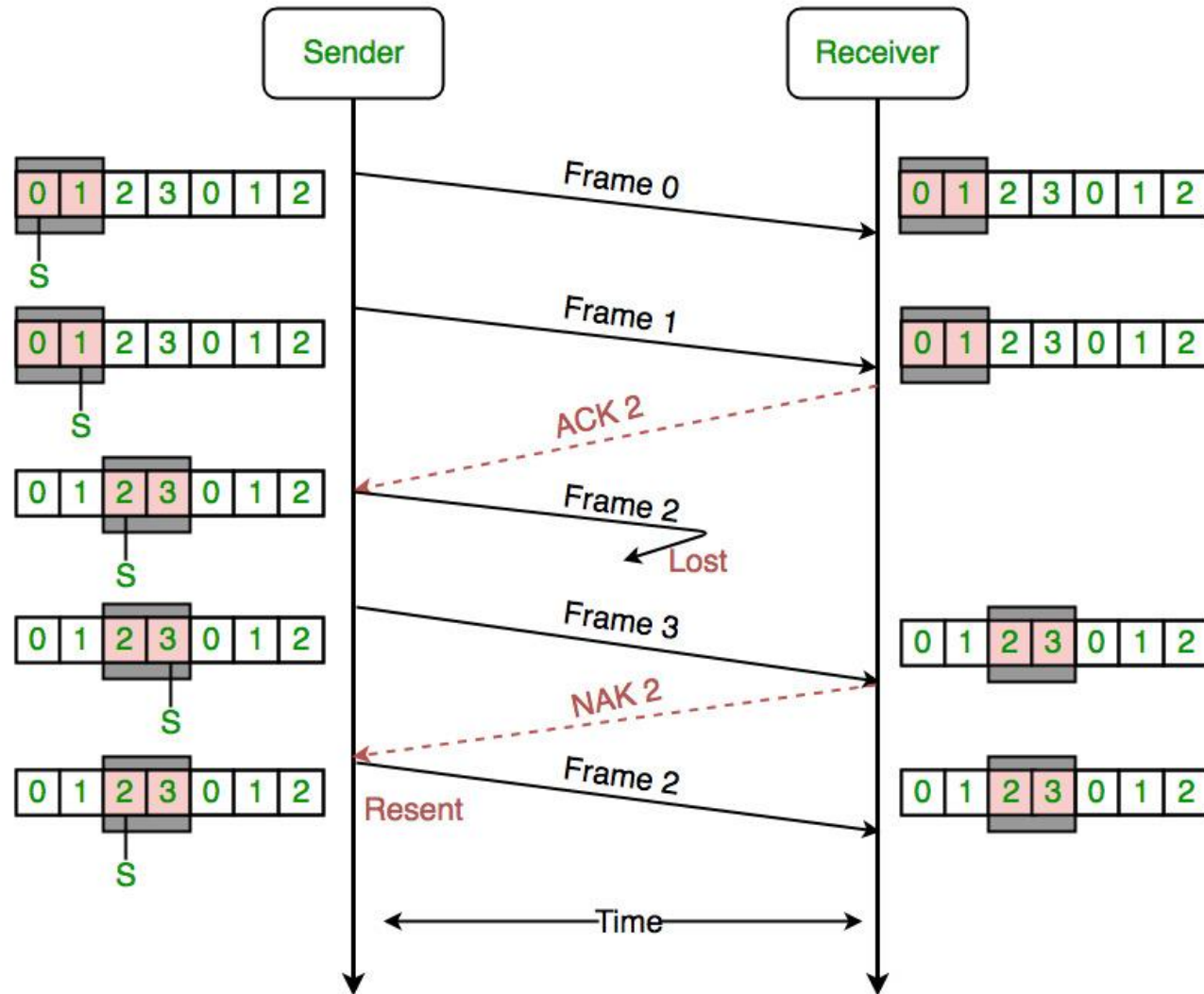
■ **FIGURE 2.14** Timeline showing four different scenarios for the stop-and-wait algorithm. (a) The ACK is received before the timer expires; (b) the original frame is lost; (c) the ACK is lost; (d) the timeout fires too soon.

Source : Computer Networks A System Approach, Larry L. Petterson, Morgan Kaufmann Elsevier, 2021, page 80

Terdapat 4 skenario yang berbeda dalam implementasi algoritma Stop and wait

Dimana keseluruhan skenario akan mengirimkan frame secara berurutan dengan memastikan frame sudah diterima terlebih dahulu baru mengirimkan frame berikutnya

# Sliding Window



Algoritma kedua adalah sliding window, dimana algoritma ini mengirimkan beberapa frame sekaligus sesuai dengan banyak frame (window size) yang telah ditentukan dan mengirimkan frame berikutnya sesuai dengan ack terakhir yang diterima

# Referensi

Computer Networks Sixth Edition, Andrew S. Tanenbaum, Pearson Education Limited, 2021

Fundamentals of Computer Networks, Matthew N. O. Sadiku, Humana Press, 2022

Computer Networks A System Approach, Larry L. Petterson, Morgan Kaufmann Elsevier, 2021

Cstaleem (Accessed: 2024, October 3). Checksum in Error Detection. <https://cstaleem.com/checksum-in-error-detection>

Geeksforgeek (Accessed: 2024, October 3). Difference between Byte stuffing and Bit stuffing. <https://www.geeksforgeeks.org/difference-between-byte-stuffing-and-bit-stuffing/>

Geeksforgeek (Accessed: 2024, October 3). Basic Frame Structure of HDLC. <https://www.geeksforgeeks.org/basic-frame-structure-of-hdlc/>

# Referensi

- Geeksforgeek (Accessed: 2024, October 3). Error Detection in Computer Networks. <https://www.geeksforgeeks.org/error-detection-in-computer-networks/>
- Geeksforgeek (Accessed: 2024, October 3). Error Detection Code – Checksum. <https://www.geeksforgeeks.org/error-detection-code-checksum/>
- Geeksforgeek (Accessed: 2024, October 3). Sliding Window Protocol | Set 3 (Selective Repeat). <https://www.geeksforgeeks.org/sliding-window-protocol-set-3-selective-repeat/>

**Week 7**

---

# Medium Access Control Sub Layer

---