



Jaringan Komunikasi Data E-Learning

Presents:

Data and Logical Link Control (LLC)



Minggu 5





Outline

- Error Correction (Block Parity, Hamming Code)
 - HDLC (High-Level Data Link Control)
 - Point to point Protocol
- 

Block Parity

- Sederhana, menggunakan perhitungan pariti dasar
- Menggunakan pariti baris dan kolom sebagai sarana koreksi kesalahan
- Hanya mampu mengkoreksi kesalahan 1 bit, mampu mendeteksi kesalahan lebih dari 1 bit
- Efisiensi tergantung dari ukuran baris dan kolom yang digunakan, semakin banyak baris dan kolom akan semakin banyak bit pariti

Contoh Block Parity 1

1	1	0	1	1		1	1	0	1	1	√
1	0	1	1	1		1	0	0	1	1	×
1	1	0	1	1	→	1	1	0	1	1	√
0	0	1	1	0		0	0	1	1	0	√
1	0	0	0	1		1	0	0	0	1	√
						√	√	×	√	√	

Contoh Block Parity 2

1	1	0	1	1	1	1	0	1	1	✓
1	0	1	1	1	1	0	0	1	1	✗
1	1	0	1	1	1	0	0	1	1	✗
0	0	1	1	0	0	0	1	1	0	✓
1	0	0	0	1	1	0	0	0	1	✓
					✓	✗	✗	✓	✓	✓

Contoh Block Parity 3

1	1	0	1	1	1	1	0	1	1	√
1	0	1	1	1	1	1	0	1	1	√
1	1	0	1	1	1	0	1	1	1	√
0	0	1	1	0	0	0	1	1	0	√
1	0	0	0	1	1	0	0	0	1	√
					√	√	√	√	√	√

Hamming Code

Hamming Code diciptakan oleh Richard Wesley Hamming, seorang ahli matematika Amerika



Hamming Code: Sisi Pengirim(1)

Hamming code menggunakan metode matematik modulo 2.

Langkah-langkah Hamming code di sisi pengirim :

1. Disisipkan bit-bit pariti di posisi bit 2^n

→ bit ke-1,2,4,8,16,32 dst.

Sehingga deretan bit →

P₁ **P₂** d₁ **P₃** d₂ d₃ d₄ **P₄** d₅ d₆ d₇ d₈ d₉ dst

Hamming Code: Sisi Pengirim(2)


2. Lakukan *parity check* dengan memperhatikan letak bit-bit yang diperiksa.

Ketentuan bit yang diperiksa: skip (n-1) bit, check n bit, skip n bit, check n bit, dst.. [n = posisi bit parity]

Posisi bit	1	2	3	4	5	6	7	8	9	10	11	12	13
Kategori	p1	p2	d1	p3	d2	d3	d4	p4	d5	d6	d7	d8	d9
p1	√		√		√		√		√		√		√
p2		√	√			√	√			√	√		
p3				√	√	√	√					√	√
p4								√	√	√	√	√	√




Hamming Code: Sisi Pengirim(3)

3. Lakukan langkah XOR untuk semua bit yang posisinya telah ditandai. Bit hasil XOR ini adalah bit paritynya.
 4. Data dikirimkan dengan bit-bit parity yang telah disisipkan.
- 



Hamming Code: Sisi Penerima (1)

1. Untuk menentukan posisi bit informasi dan *parity*, gunakan ketentuan seperti pada langkah 1 dan 2 metode Hamming di sisi pengirim.
 2. Lakukan proses xor untuk bit-bit sesuai ketentuan pada langkah ke-3 metode Hamming seperti di sisi pengirim.
- 

Hamming Code: Sisi Penerima (2)

Posisi bit	1	2	3	4	5	6	7	8	9	10	11	12	13	Proses Xor
Kategori	p1	p2	d1	p3	d2	d3	d4	p4	d5	d6	d7	d8	d9	
Bit informasi	0	1	1	1	0	0	1	1	0	1	1	1	1	
p1	✓		✓		✓		✓		✓		✓		✓	$p1 = 0 \text{ xor } 1 \text{ xor } 0 \text{ xor } 1 \text{ xor } 0 \text{ xor } 1 \text{ xor } 1 = 0$
p2		✓	✓			✓	✓			✓	✓			$p2 = 1 \text{ xor } 1 \text{ xor } 0 \text{ xor } 1 \text{ xor } 1 \text{ xor } 1 = 1$
p3				✓	✓	✓	✓					✓	✓	$p3 = 1 \text{ xor } 0 \text{ xor } 0 \text{ xor } 1 \text{ xor } 1 \text{ xor } 1 = 0$
p4								✓	✓	✓	✓	✓	✓	$P4 = 1 \text{ xor } 0 \text{ xor } 1 \text{ xor } 1 \text{ xor } 1 \text{ xor } 1 = 1$

Hasil xor jika dilihat dari mulai urutan pertama sampai keempat adalah 0 1 0 1. Urutan bit ini dibaca terbalik, yaitu 1010 sama dengan nilai 10 dalam desimal. Artinya, ada yang salah yaitu bit ke-10


Metode FEC Lain

- Semua metode FEC pada dasarnya menggunakan metode matematik modulo 2
- Metoda ini terus dikembangkan dengan tujuan:
 - Mendapatkan kemampuan koreksi bit yang semakin banyak
 - Dengan mengurangi jumlah bit pariti yang dibutuhkan
 - Mampu melanjutkan komunikasi walaupun sempat terputus.



Metode FEC Lain

Metoda yang umum digunakan:

- BCH Code
 - Reed Solomon Code
 - Convolutional Code
 - Trellis Code
 - Turbo Code
- 




Simple Protocols





Protokol

Protokol diperlukan untuk implementasi kontrol data link.
Protokol digunakan pada kanal noiseless (Ideal) serta
kanal noisy (real).



Protocols

Noiseless channels

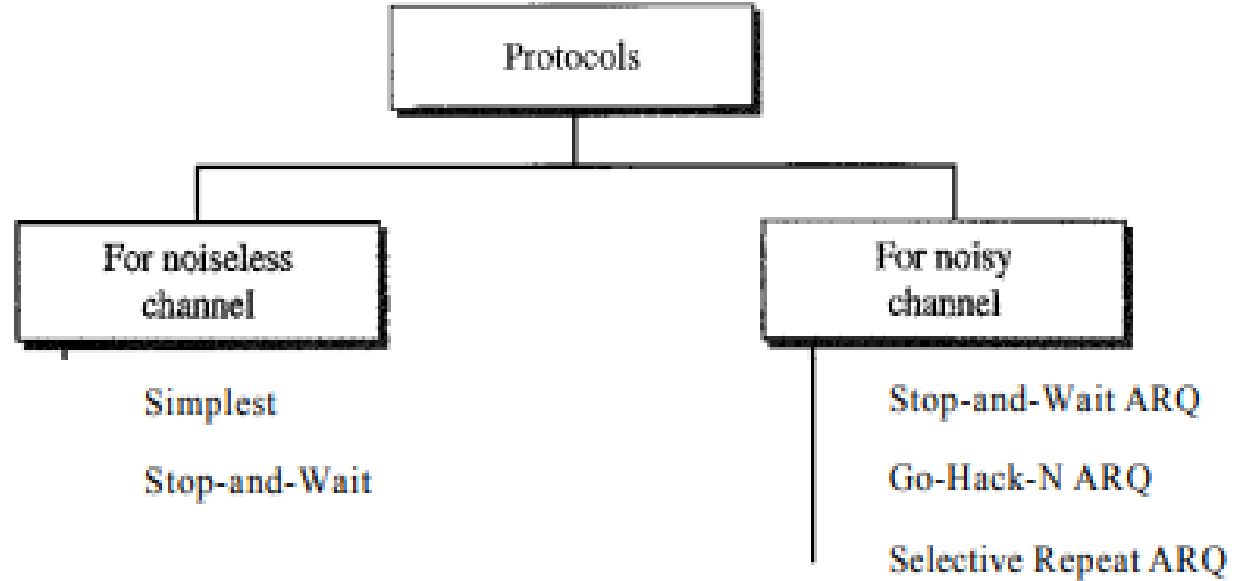
- Simplest protocol
- Stop and wait protocol

Noisy channels

- Stop and wait ARQ
- Go Back N ARQ
- Selective repeat ARQ

Piggybacking


Taxonomy of protocols



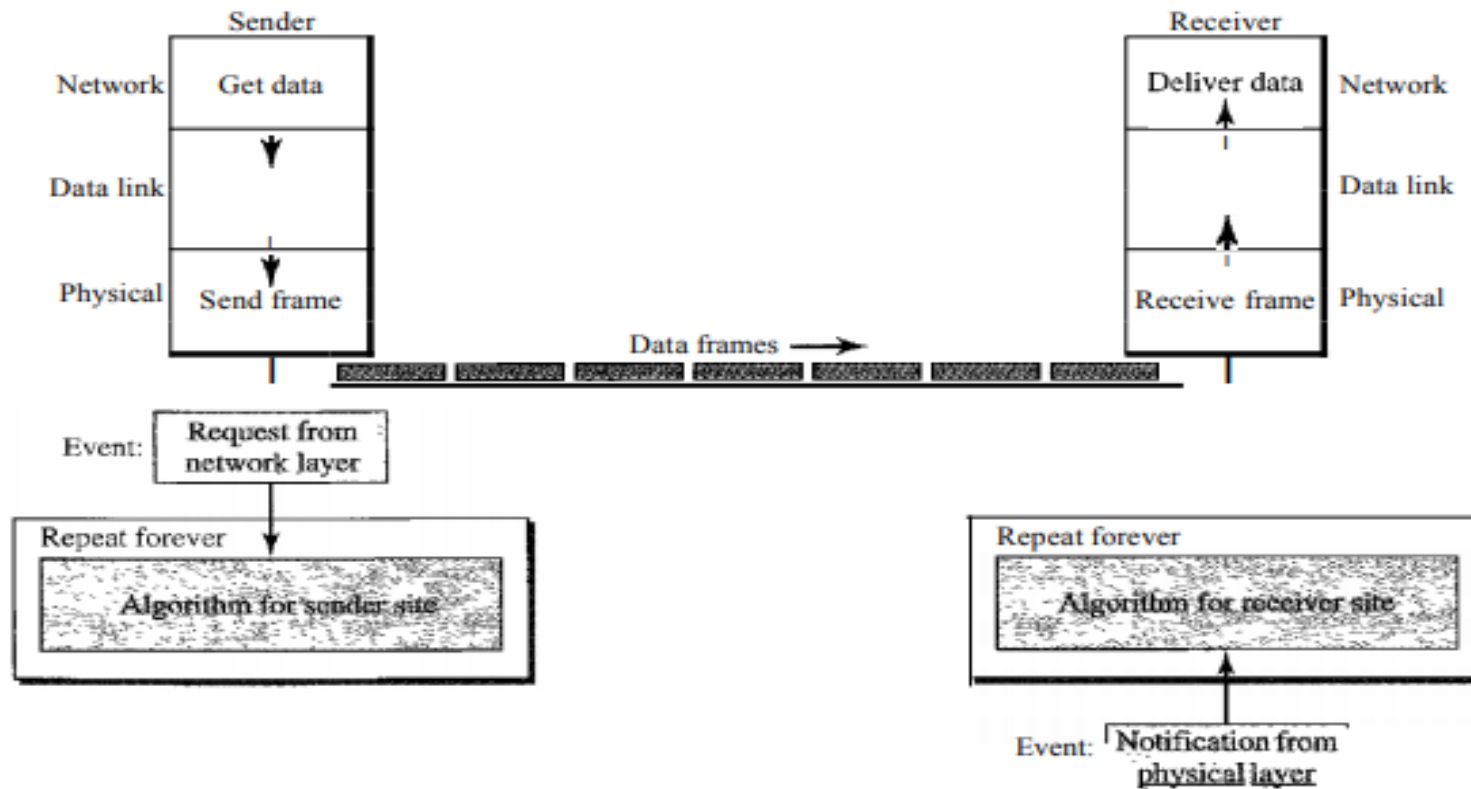


Noiseless Channel

Simplest protocol tidak memiliki kontrol flow maupun kesalahan. Protokol ini merupakan protocol searah, frame data bepergian hanya dalam satu arah, yaitu dari pengirim ke penerima.



The design of the simplest protocol with no flow or error control





Noiseless Channel

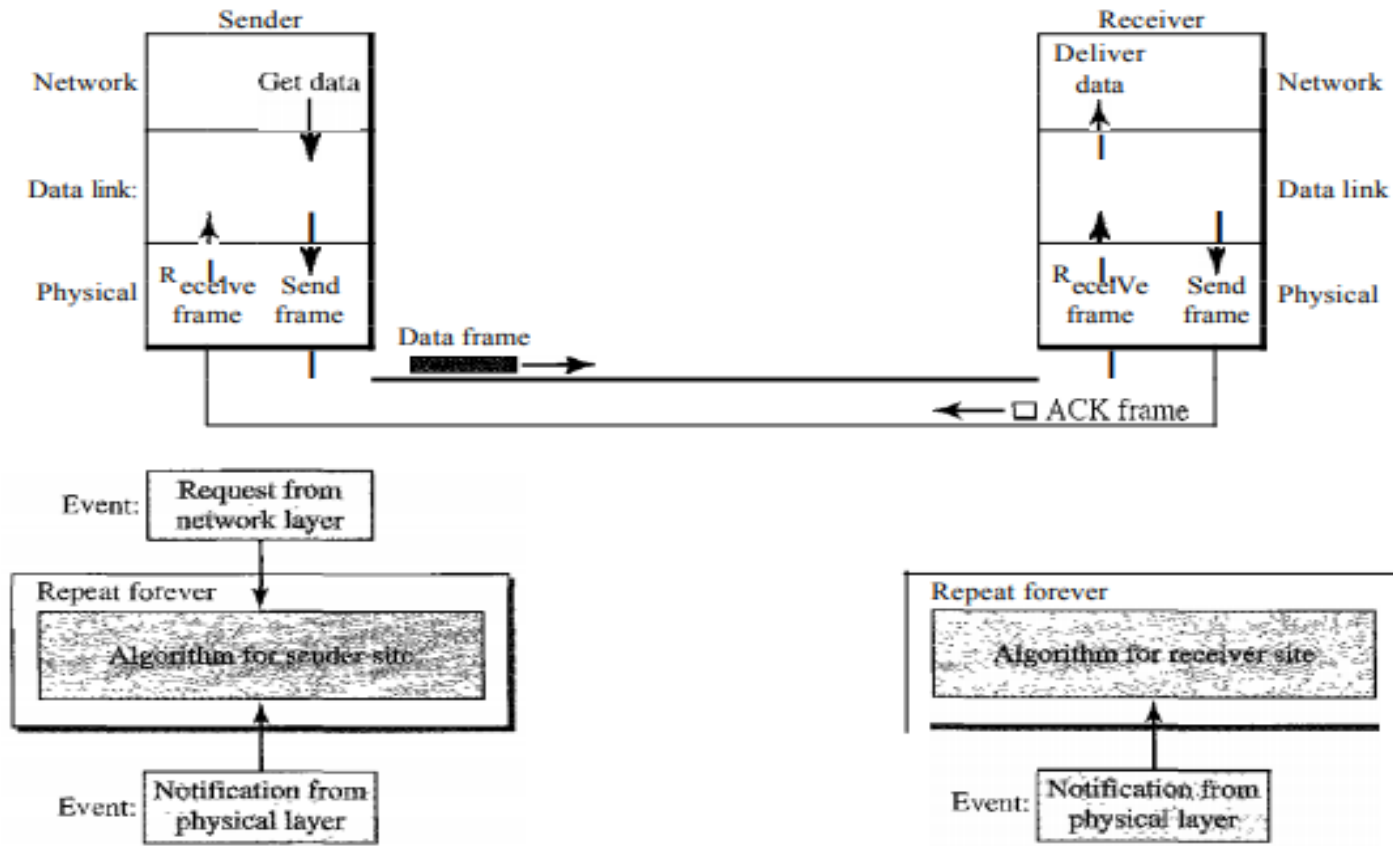
Stop and wait protocol

Pengirim mengirim satu frame, berhenti sampai pengirim menerima konfirmasi dari penerima (ok untuk melanjutkan), dan kemudian mengirim frame berikutnya.

Hal ini untuk mencegah penerima kewalahan dengan frame yang berakibat terbuangnya frame atau penolakan layanan.




Design of Stop-and-Wait Protocol



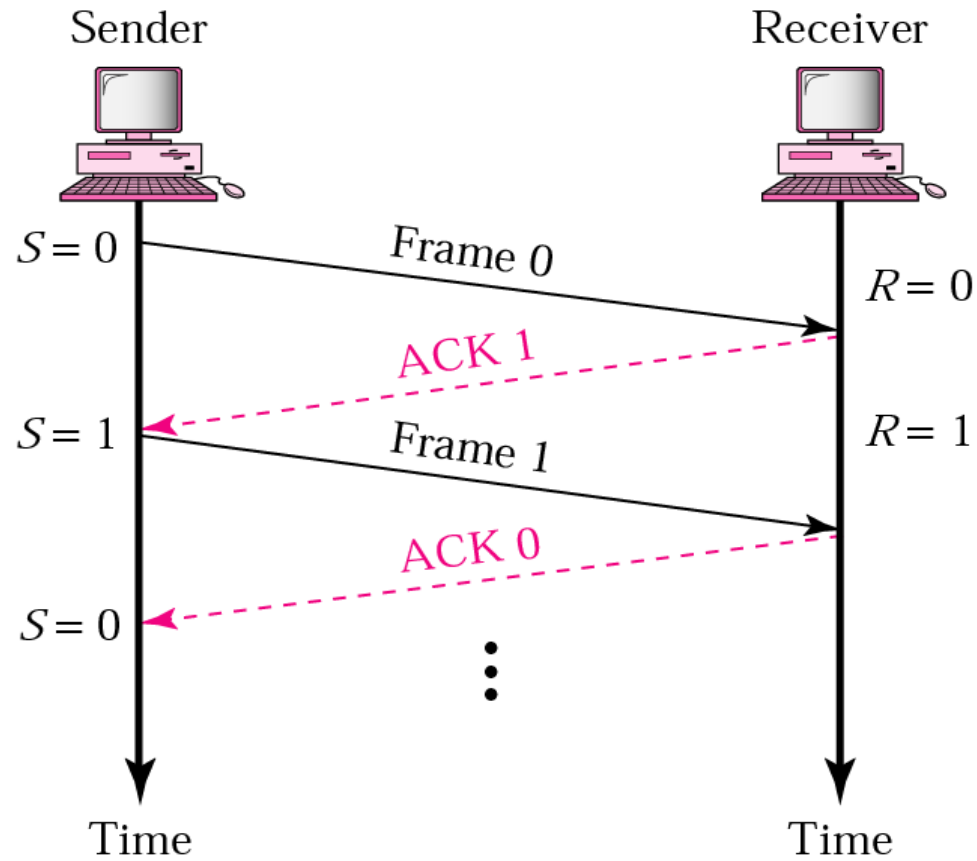


Noisy Channel

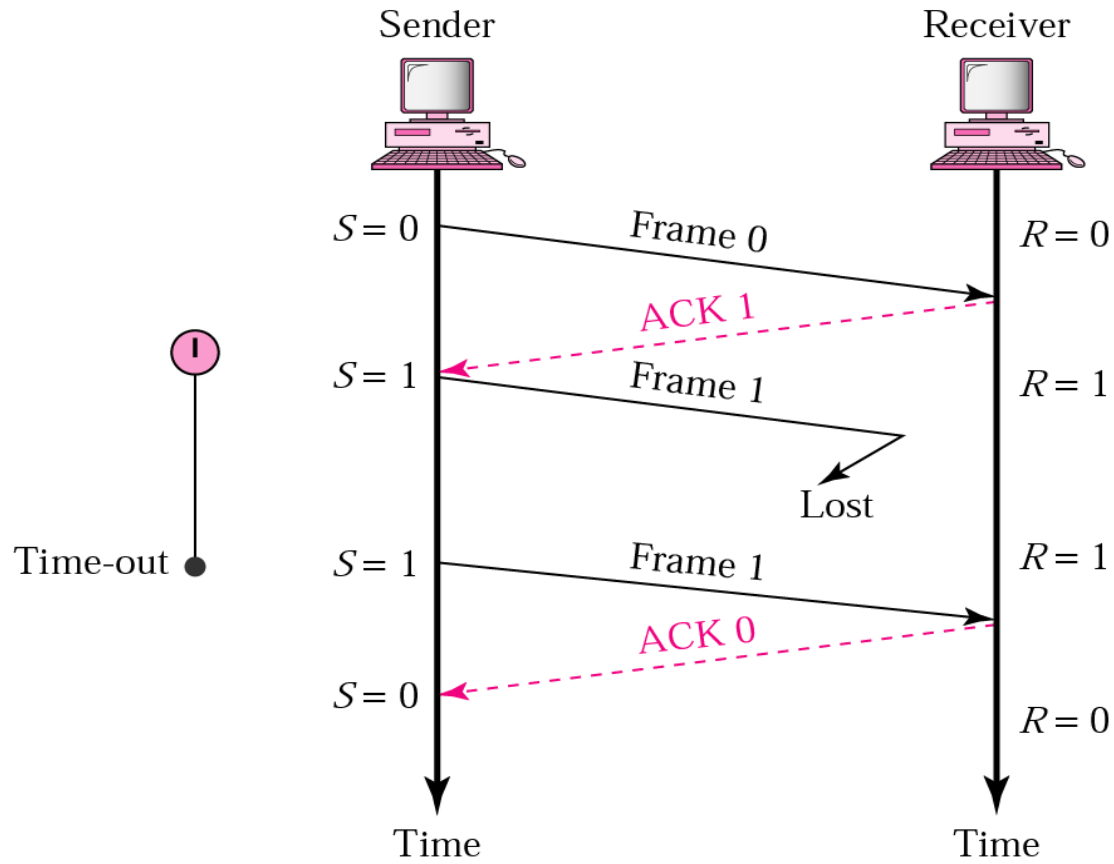
Stop and Wait ARQ (Automatic Repeat Request)/Idle ARQ

- ✓ Pengirim menyimpan salinan frame terakhir yang terkirim dan menunggu ACK untuk frame tersebut.
 - ✓ Frame selanjutnya tidak dapat dikirim hingga ACK diterima.
 - ✓ Frame diberi nomor bergantian 0 dan 1
 - ✓ Frame rusak atau hilang dikirmkan kembali
 - ✓ Berulang hingga EOT (end of transmission) dikirm
- 

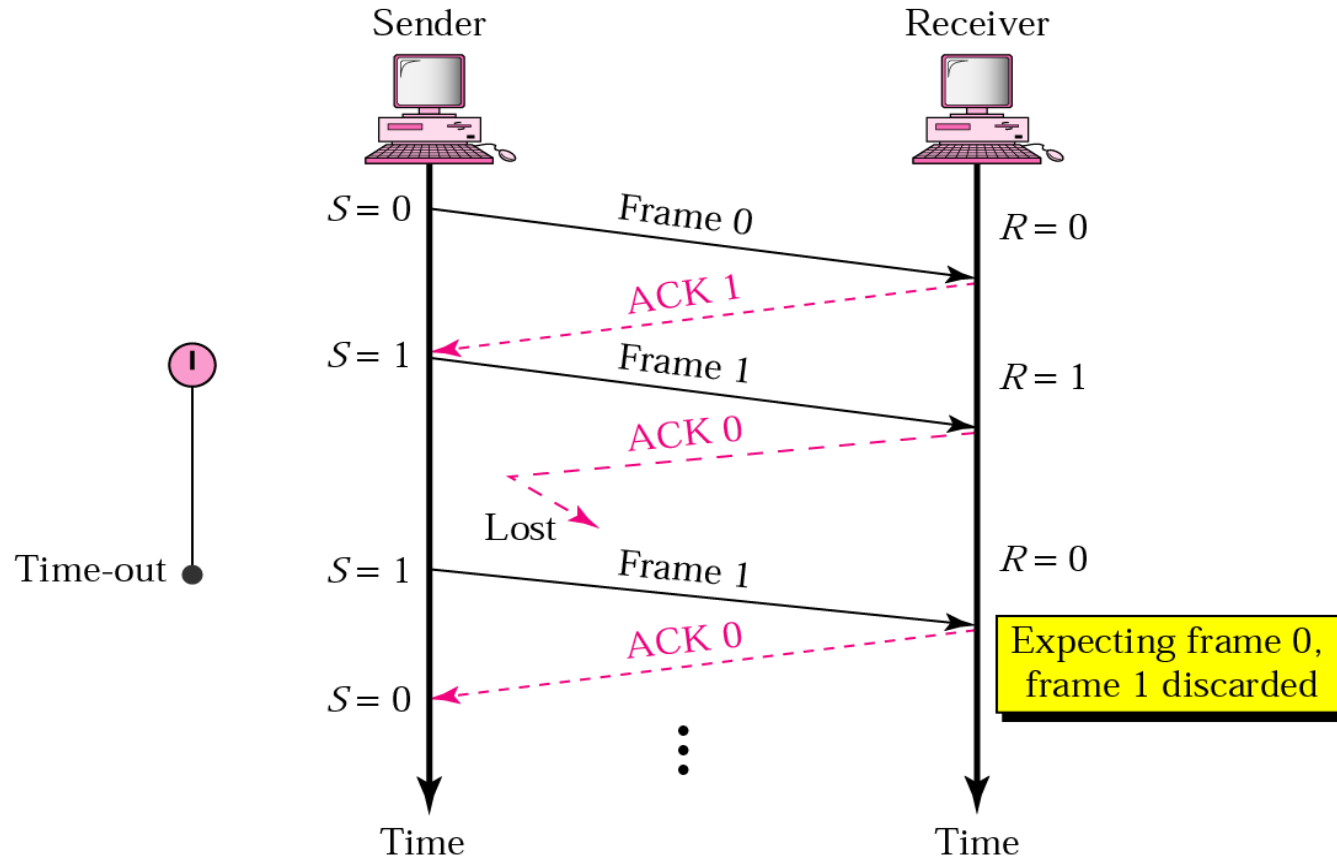
Stop-and-Wait Normal Operation



Lost or Damaged Frame




Lost Acknowledgement





Noisy Channel

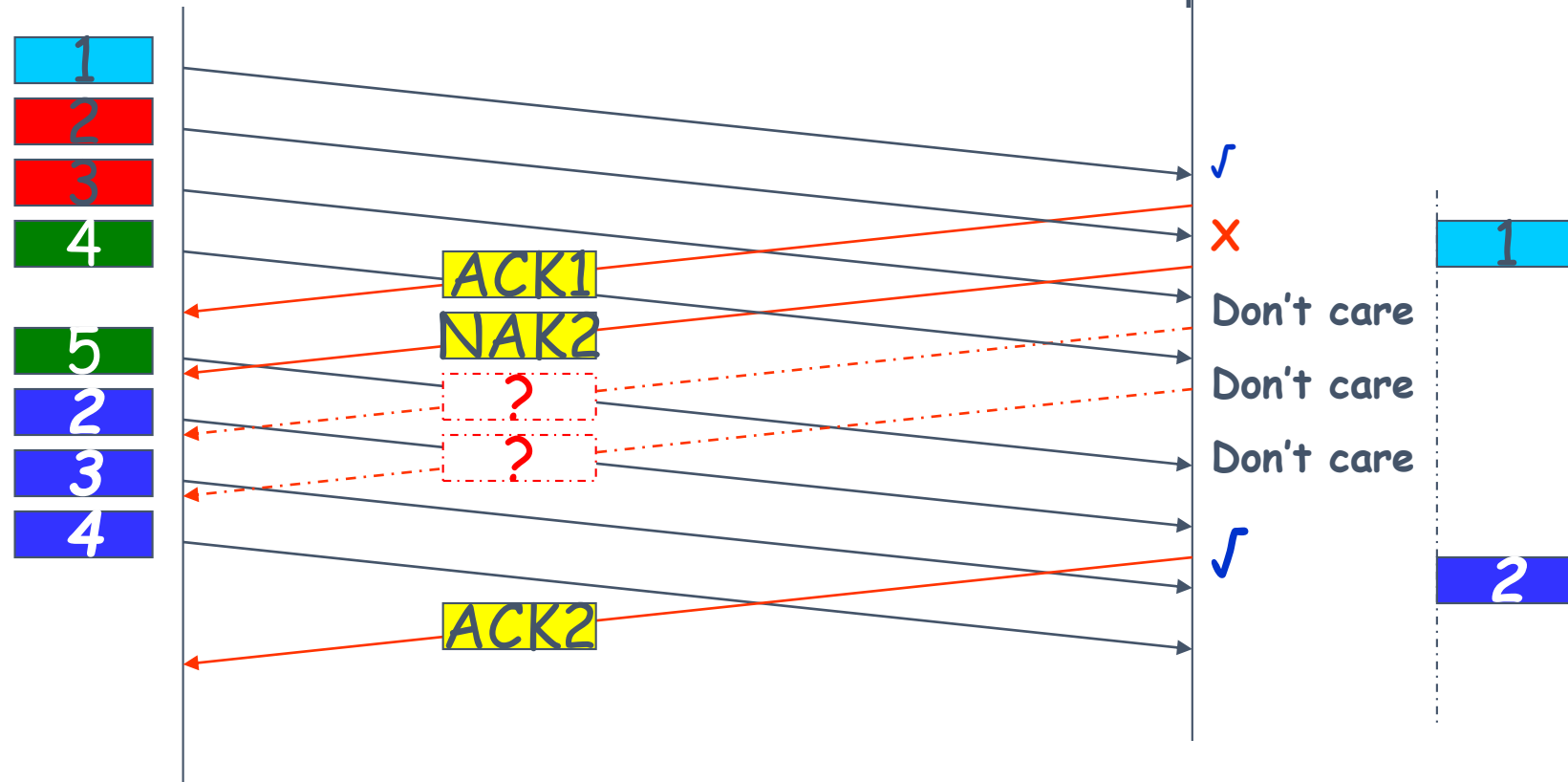
Stop and Wait ARQ (Automatic Repeat Request)

- Keuntungan: simple, setiap frame dicek dan di-ACK sebelum frame selanjutnya dikirim
 - Penomoran frame mencegah duplikasi
 - Kelemahan: tidak efisien, lambat
 - Frame dan ACK memakai seluruh bandwidth
 - Jika jarak antarperangkat jauh, waktu tunggu yang dibutuhkan cukup lama
 - Pengirim akan mengirim kembali frame jika tidak mendapat ACK
- 

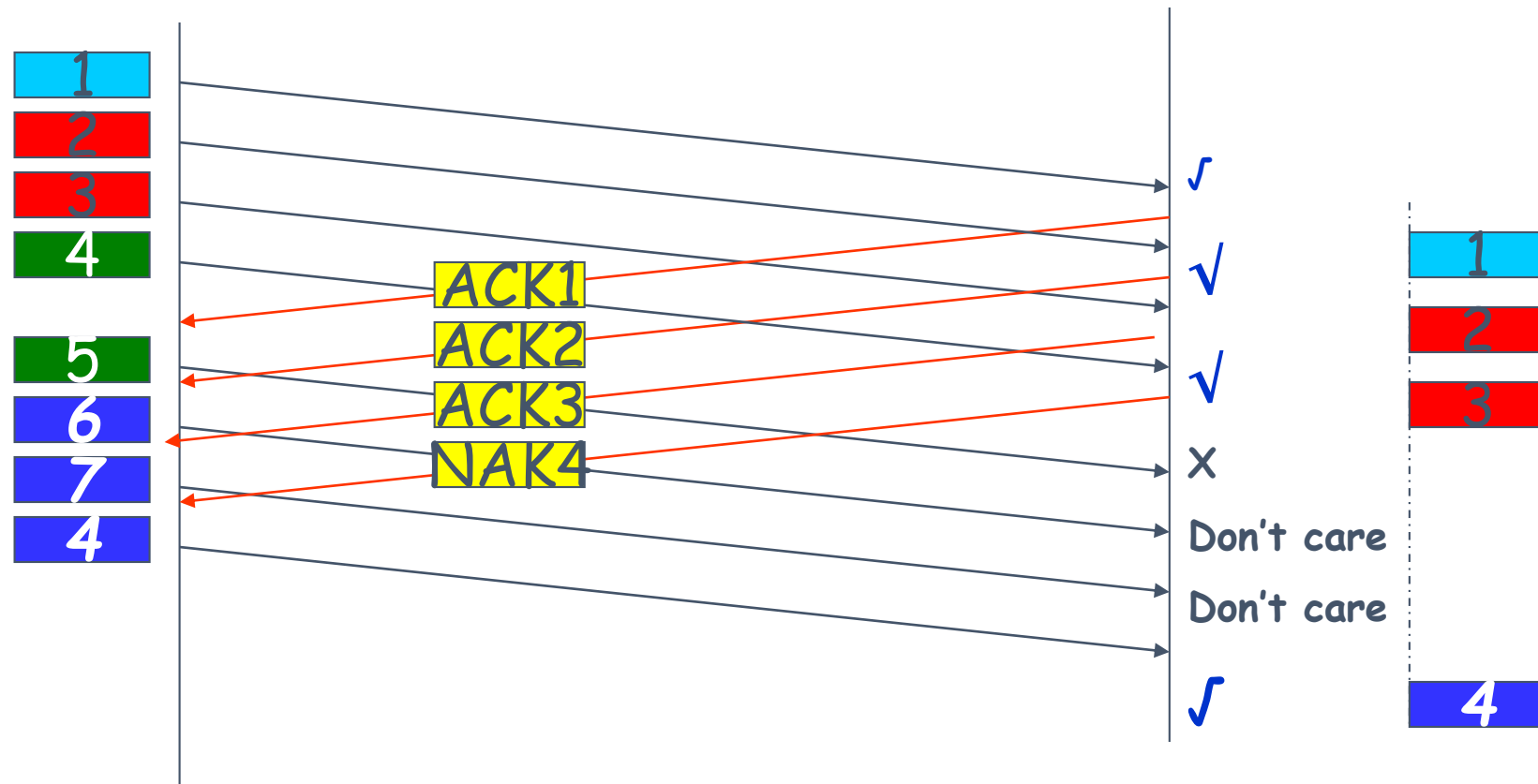
Noisy Channel

ARQ : Go Back N

- Mengirim ulang mulai dari paket yang salah
- Paket akan diterima terjaga urutannya
- Efisiensi saluran lebih rendah dari Selective Repeat



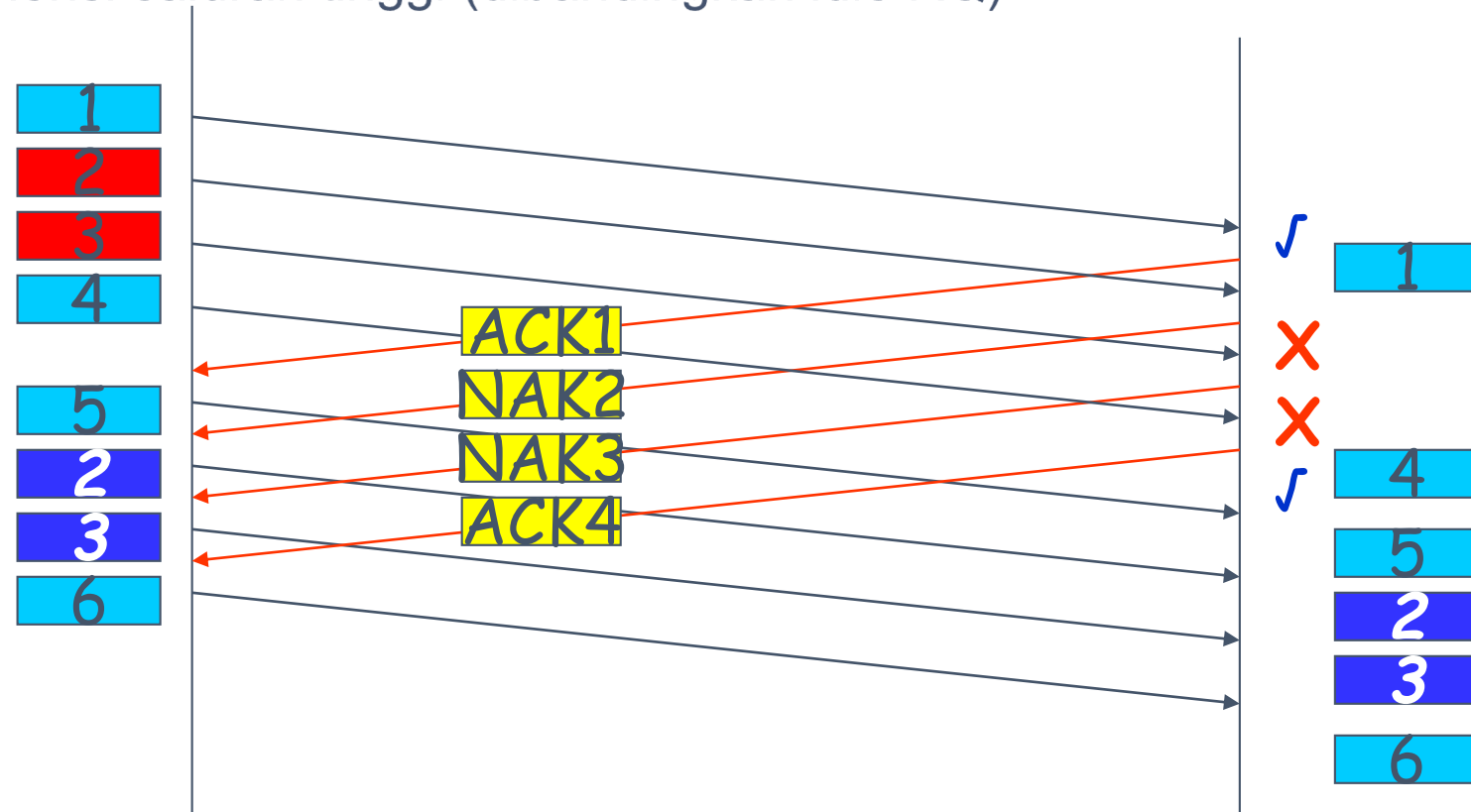
Kasus Lain Go Back N



Noisy Channel


Selective Repeat ARQ

- Hanya mengirim ulang untuk paket yang salah
- Paket diterima tidak berurutan
- Efisiensi saluran tinggi (dibandingkan idle RQ)

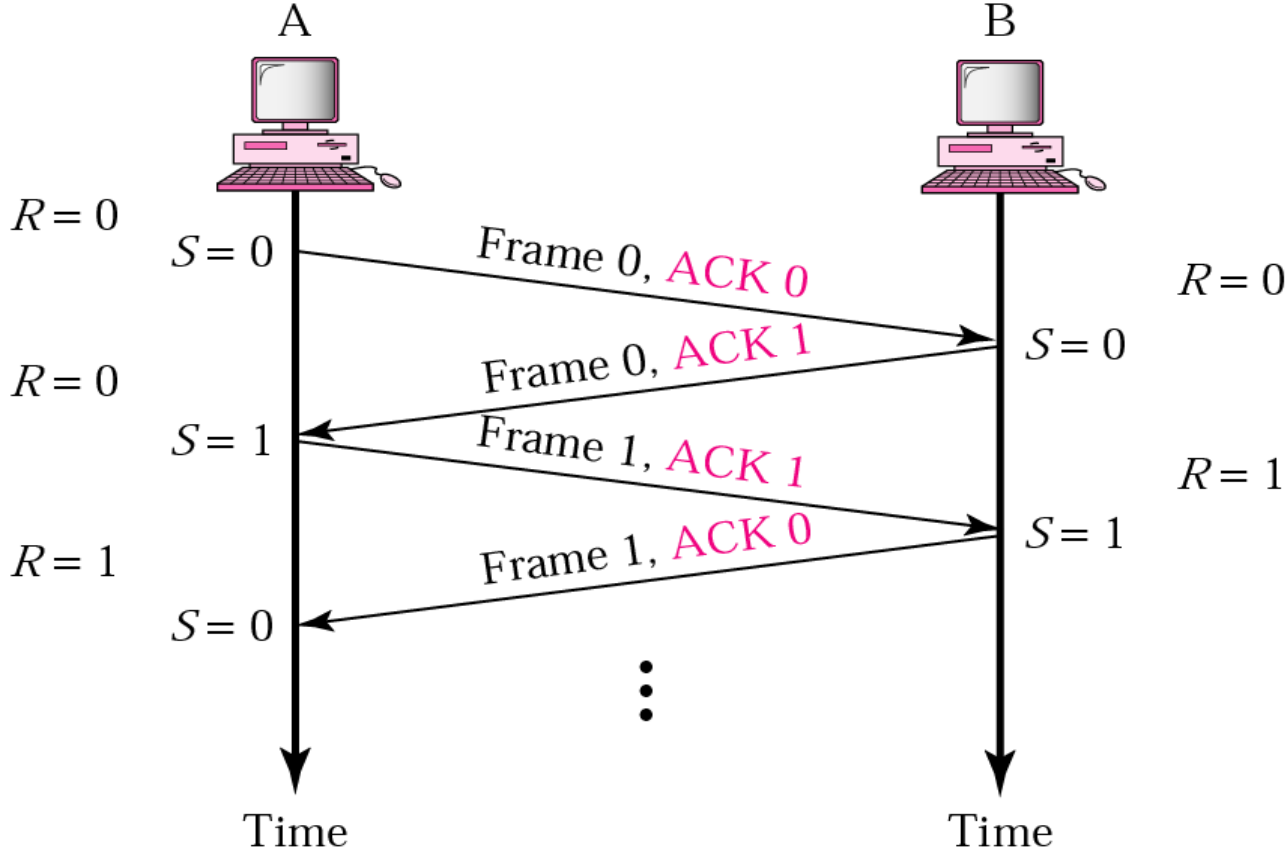




Piggybacking

- Metode yang mengkombinasikan data frame dan acknowledgement
 - Menghemat bandwidth akibat less overhead from separate data frame and ACK frame into one frame
- 

Piggybacking






HDLC

(High-Level Data Link Control)





High-Level Data Link Control

- ✓ Salah satu protocol data link control.
 - ✓ Protokol bit-oriented untuk komunikasi point-to-point dan multipoint links.
 - ✓ Mengimplementasikan mekanisme ARQ
- 



Tipe HDLC Station

Primary station

- Mengendalikan operasi link
- Frame yang dikeluarkan disebut command
- Me-maintain logical link terpisah ke tiap secondary station

Secondary station

- Di bawah kendali primary station
- Frame yang dikeluarkan disebut response

Combined station

- Dapat mengeluarkan commands dan responses
- 




Konfigurasi HDLC Link

Unbalanced

- One primary and one or more secondary stations
- Supports full duplex and half duplex

Balanced

- Two combined stations
 - Supports full duplex and half duplex
- 

Mode Transfer HDLC

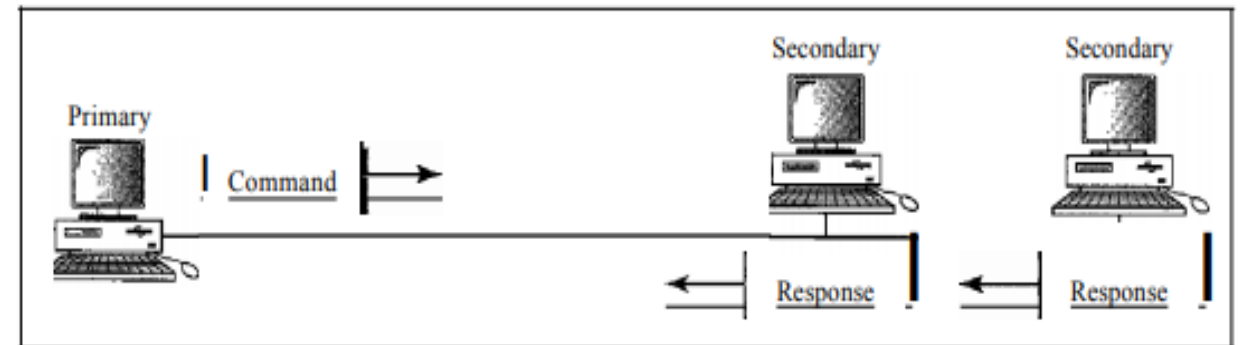
Normal Response Mode (NRM)

- Konfigurasi unbalanced
- Primary menginisiasi transfer ke secondary
- Primary station dapat mengirim commands, Secondary station hanya dapat merespon
- Digunakan untuk point-to-point dan multiple-point links

Figure 11.25 *Normal response mode*



a. Point-to-point



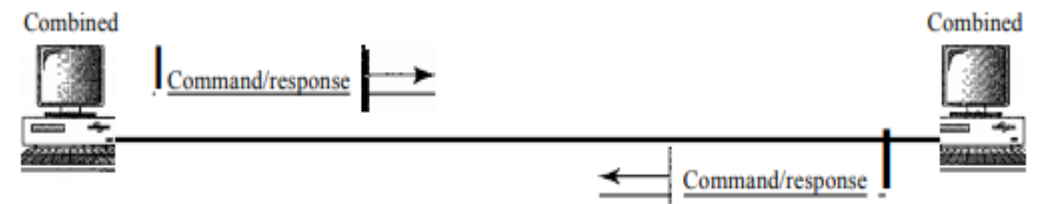
b. Multipoint

Mode Transfer HDLC (2)

Asynchronous Balanced Mode (ABM)


- Konfigurasi balanced
- point-to-point
- Setiap station dapat berfungsi sebagai primary dan secondary (acting as peers)
- Salah satu station dapat menginisiasi transmisi tanpa persetujuan
- Paling banyak digunakan

Figure 11.26 *Asynchronous balanced mode*



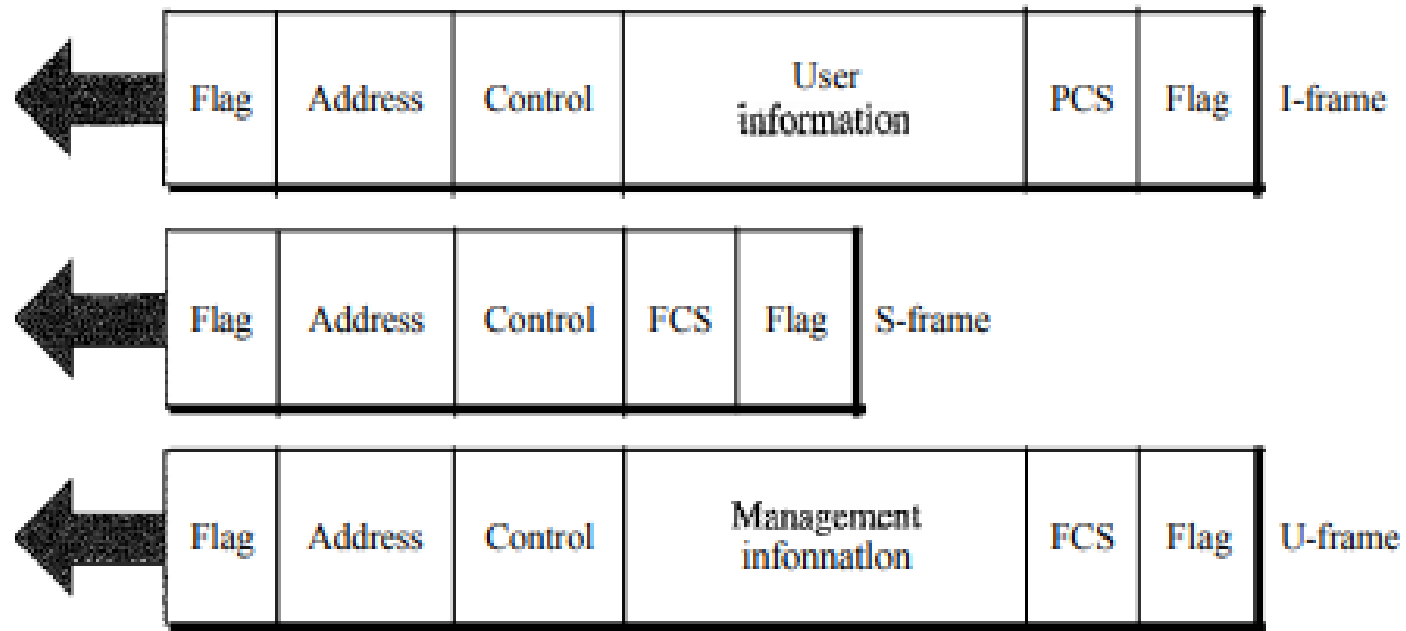


Frame HDLC

- HDLC mendefinisikan tiga jenis frames: **information frames (I-frames)**, **supervisory frames (S-frames)**, and **unnumbered frames (V-frames)**
 - I-frame digunakan untuk mengangkut data pengguna dan mengontrol informasi yang berkaitan dengan data pengguna (piggybacking).
 - S-frame hanya digunakan untuk mengirim informasi kontrol.
 - V-frame dicadangkan untuk manajemen sistem. Informasi yang dibawa oleh V-frame dimaksudkan untuk mengelola tautan itu sendiri.
- 


Frame HDLC

HDLC frames






Frame HDLC

- **Flag field.** The flag field of an HDLC frame is an 8-bit sequence with the bit pattern 01111110 that identifies both the beginning and the end of a frame and serves as a synchronization pattern for the receiver.
 - **Address field.** The second field of an HDLC frame contains the address of the secondary station. If a primary station created the frame, it contains a *to* address. If a secondary creates the frame, it contains *from* address. An address field can be 1 byte or several bytes long, depending on the needs of the network. One byte can identify up to 128 stations (1 bit is used for another purpose). Larger networks require multiple-byte address fields. If the address field is only 1 byte, the last bit is always a 1. If the address is more than 1 byte, all bytes but the last one will end with 0; only the last will end with 1. Ending each intermediate byte with 0 indicates to the receiver that there are more address bytes to come.
- 



Frame HDLC

- **Control field.** The control field is a 1- or 2-byte segment of the frame used for flow and error control. The interpretation of bits in this field depends on the frame type. We discuss this field later and describe its format for each frame type.
 - **Information field.** The information field contains the user's data from the network layer or management information. Its length can vary from one network to another.
 - **FCS field.** The frame check sequence (FCS) is the HDLC error detection field. It can contain either a 2- or 4-byte ITU-T CRC.
- 




Point-to-Point Protocol






Point-to-Point Protocol (PPP) Services

- PPP mendefinisikan format frame yang akan ditukar antar perangkat.
 - PPP mendefinisikan bagaimana dua perangkat dapat menegosiasikan pembentukan link dan pertukaran data.
 - PPP mendefinisikan bagaimana data network layer dienkapsulasi dalam data link frame.
 - PPP mendefinisikan bagaimana dua perangkat dapat saling mengautentikasi.
- 

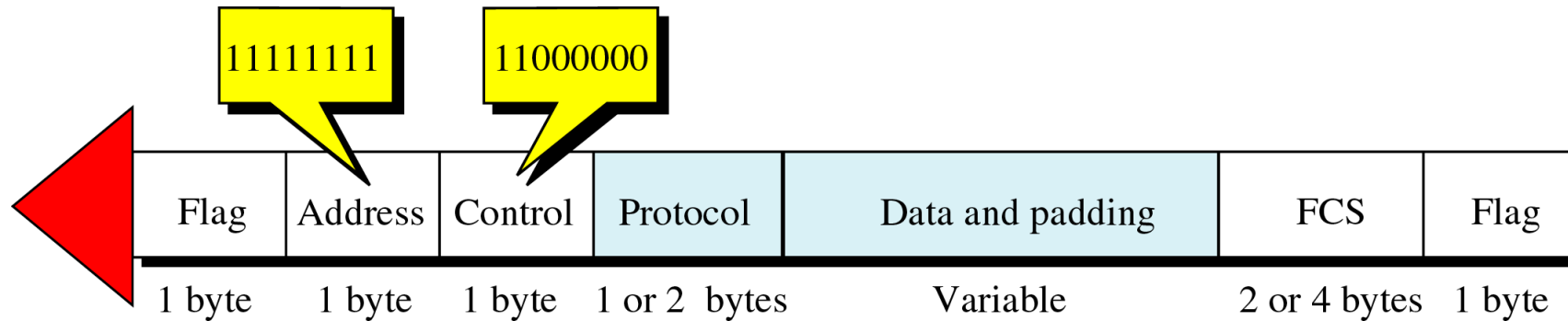


Point-to-Point Protocol (PPP) Services

- PPP menyediakan beberapa layanan network layer yang mendukung berbagai protokol network layer.
 - PPP menyediakan koneksi melalui multiple links.
 - PPP menyediakan konfigurasi alamat jaringan. Hal ini berguna ketika home user membutuhkan alamat jaringan sementara untuk terhubung ke Internet.
- 


Format Frame PPP

PPP adalah protokol berorientasi byte yang menggunakan byte stuffing dengan escape byte 01111101.






Format Frame PPP

- **Flag.** A PPP frame starts and ends with a 1-byte flag with the bit pattern 01111110. Although this pattern is the same as that used in HDLC, there is a big difference. PPP is a byte-oriented protocol; HDLC is a bit-oriented protocol. The flag is treated as a byte, as we will explain later.
 - **Address.** The address field in this protocol is a constant value and set to 11111111 (broadcast address). During negotiation (discussed later), the two parties may agree to omit this byte.
 - **Control.** This field is set to the constant value 11000000 (imitating unnumbered frames in HDLC). As we will discuss later, PPP does not provide any flow control. Error control is also limited to error detection. This means that this field is not needed at all, and again, the two parties can agree, during negotiation, to omit this byte.
- 

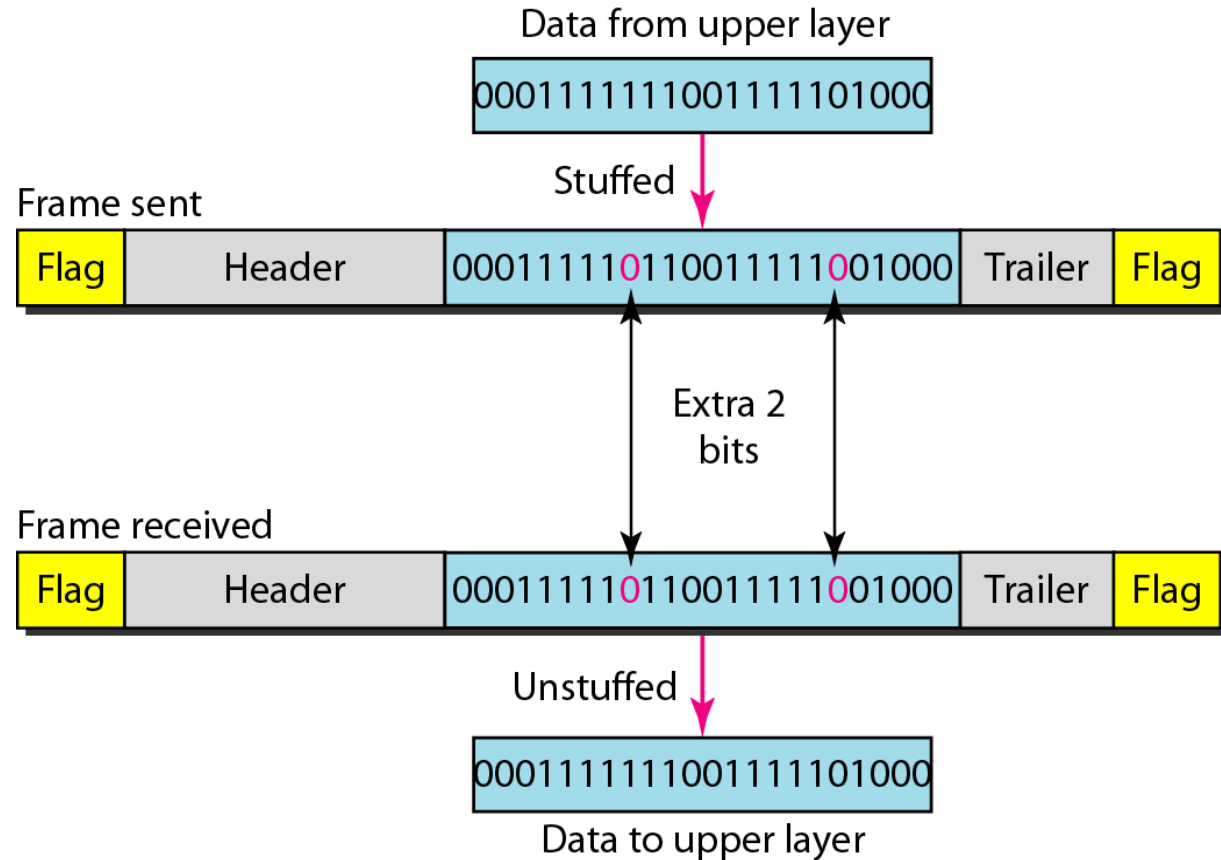


Format Frame PPP

- Protocol. The protocol field defines what is being carried in the data field: either user data or other information. We discuss this field in detail shortly. This field is by default 2 bytes long, but the two parties can agree to use only 1 byte.
 - Payload field. This field carries either the user data or other information that we will discuss shortly. The data field is a sequence of bytes with the default of a maximum of 1500 bytes; but this can be changed during negotiation. The data field is byte-stuffed if the flag byte pattern appears in this field. Because there is no field defining the size of the data field, padding is needed if the size is less than the maximum default value or the maximum negotiated value.
 - FCS. The frame check sequence (FCS) is simply a 2-byte or 4-byte standard CRC.
- 


Bit stuffing and unstuffing

Bit stuffing adalah proses menambahkan satu 0 ekstra setiap kali lima 1 berturut-turut mengikuti 0 dalam data, sehingga penerima tidak salah mengira pola 0111110 untuk sebuah flag.





Multiplexing pada PPP

- PPP menggunakan protokol lain untuk membangun link, mengautentikasi pihak-pihak yang terlibat, dan membawa data network layer.
 - Tiga set protokol adalah Link Control Protocol (LCP), dua Authentication Protocols (APs), dan beberapa Network Control Protocols (NCPs).
- 



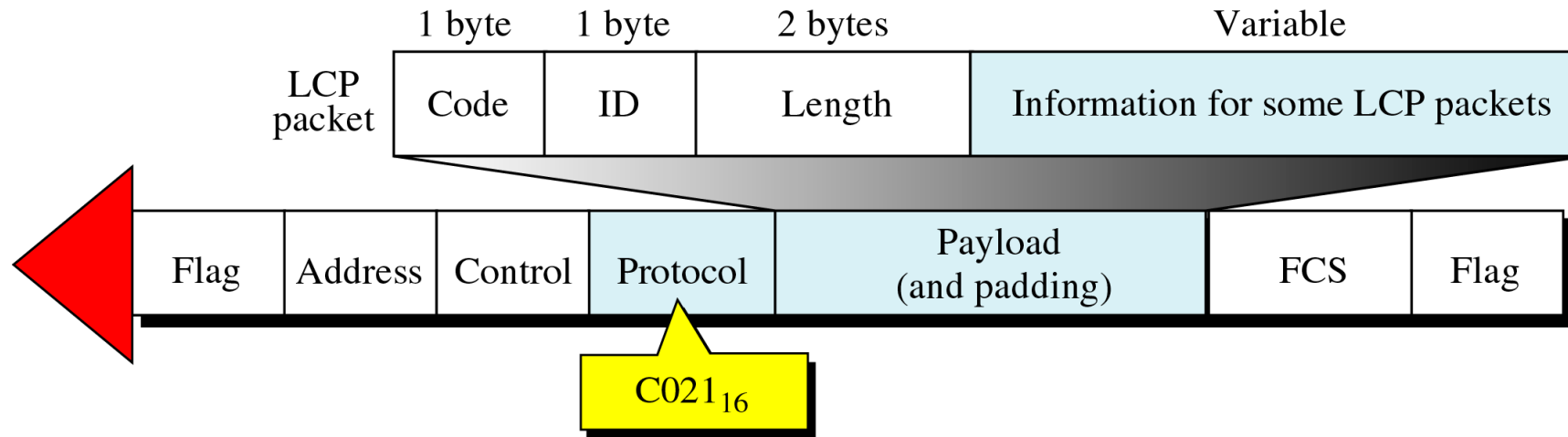
Link Control Protocol (LCP)

Bertanggung jawab untuk membangun, memelihara, mengonfigurasi, mengakhiri link, dan bernegosiasi

Semua paket LCP dibawa di payload field PPP frame
– PPP field Protocol = hex C021



Enkapsulasi Paket LCP dalam Frame




Paket LCP dan Kode

Code	Packet Type	Description
01 ₁₆	Configure-request	Contains the list of proposed options and their values
02 ₁₆	Configure-ack	Accepts all options proposed
03 ₁₆	Configure-nak	Announces that some options are not acceptable
04 ₁₆	Configure-reject	Announces that some options are not recognized
05 ₁₆	Terminate-request	Requests to shut down the line
06 ₁₆	Terminate-ack	Accepts the shut down request
07 ₁₆	Code-reject	Announces an unknown code
08 ₁₆	Protocol-reject	Announces an unknown protocol
09 ₁₆	Echo-request	A type of hello message to check if the other end is alive
0A ₁₆	Echo-reply	The response to the echo-request message
0B ₁₆	Discard-request	A request to discard the packet




Common options

Option	Default
Maximum receive unit	1500
Authentication protocol	None
Protocol field compression	Off
Address and control field compression	Off






Autentikasi

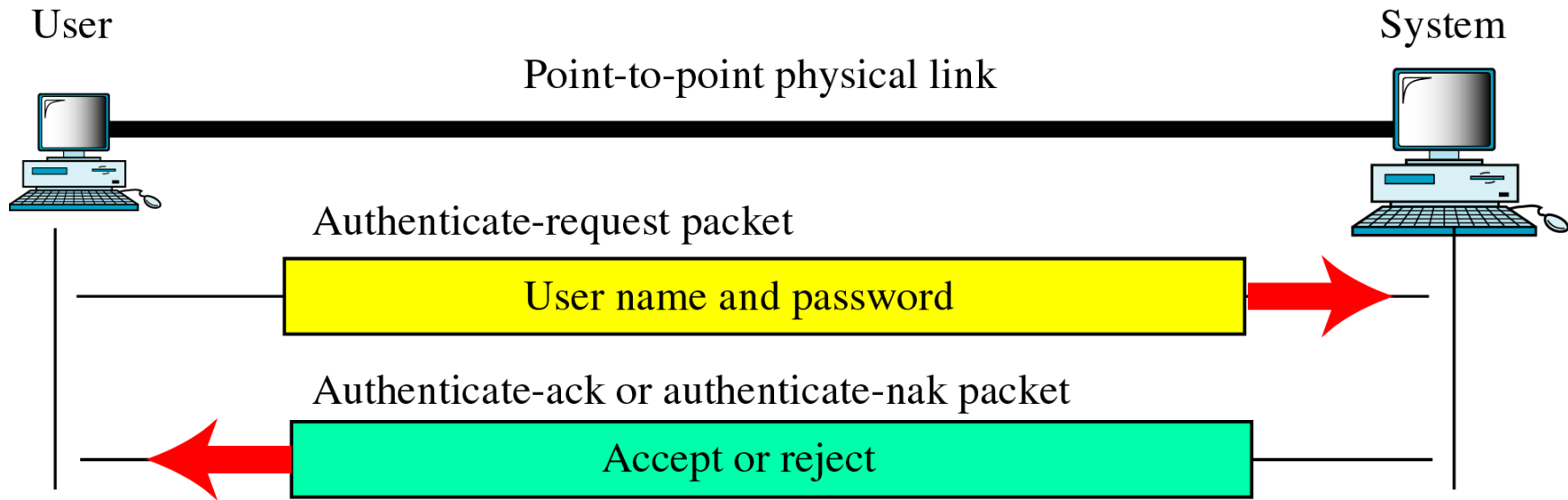
- Penting karena komunikasi dial-up
 - Dua protocol untuk autentikasi:
Password Authentication Protocol (PAP)
Challenge Handshake Authentication Protocol (CHAP)
- 



Autentikasi – PAP

- Proses dua langkah
 - 1) User mengirim ID dan password
 - 2) Sistem verifikasi
 - Paket PAP dienkapsulasi dalam PPP frame
 - Terdapat 3 tipe paket PAP
(slide berikutnya)
- 

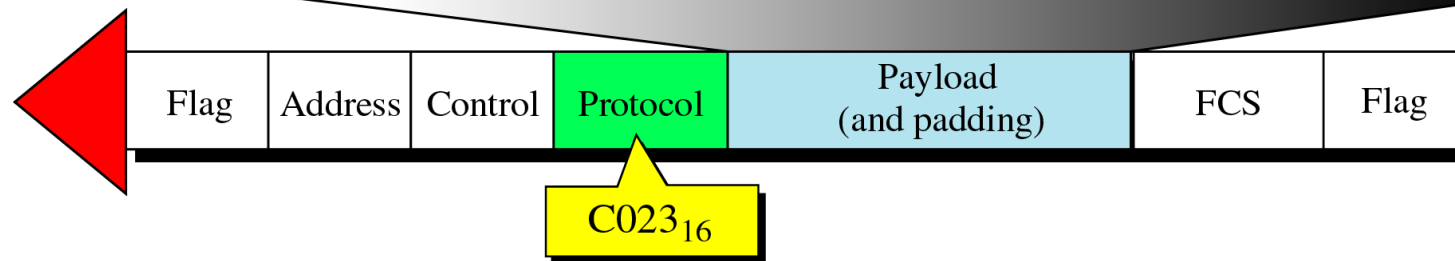
PAP



PAP Packets

PAP Packets


	1 byte	1 byte	2 bytes	1 byte	Variable	1 byte	Variable
Authenticate-request	Code = 1	ID	Length	User name length	User name	Password length	Password
Authenticate-ack	Code = 2	ID	Length	Message length	User name		
Authenticate-nak	Code = 3	ID	Length	Message length	User Name		



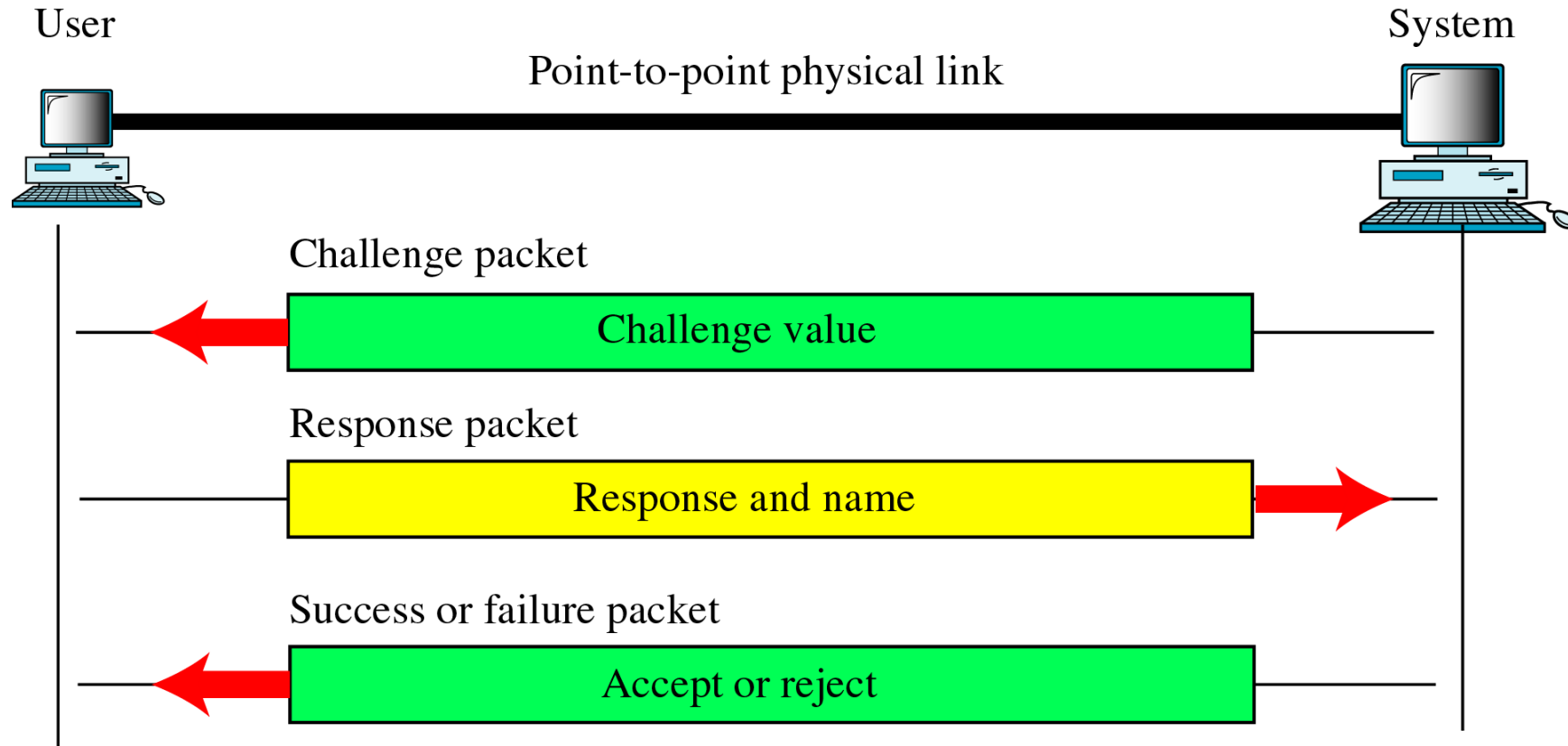


Autentikasi – CHAP

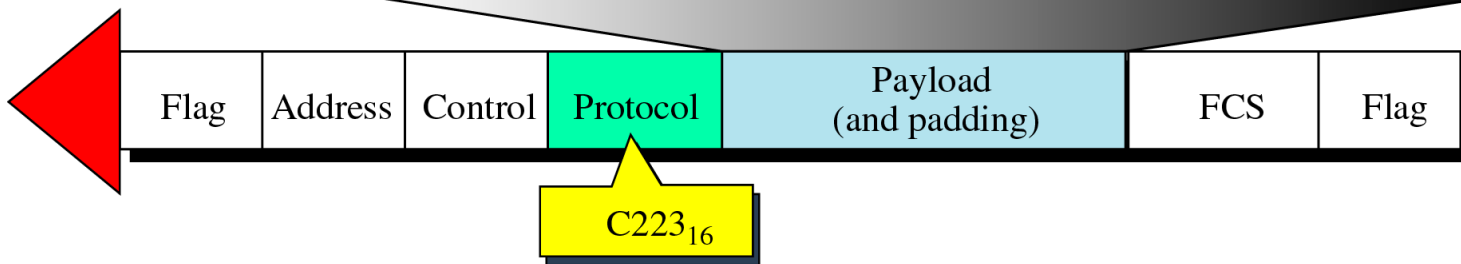
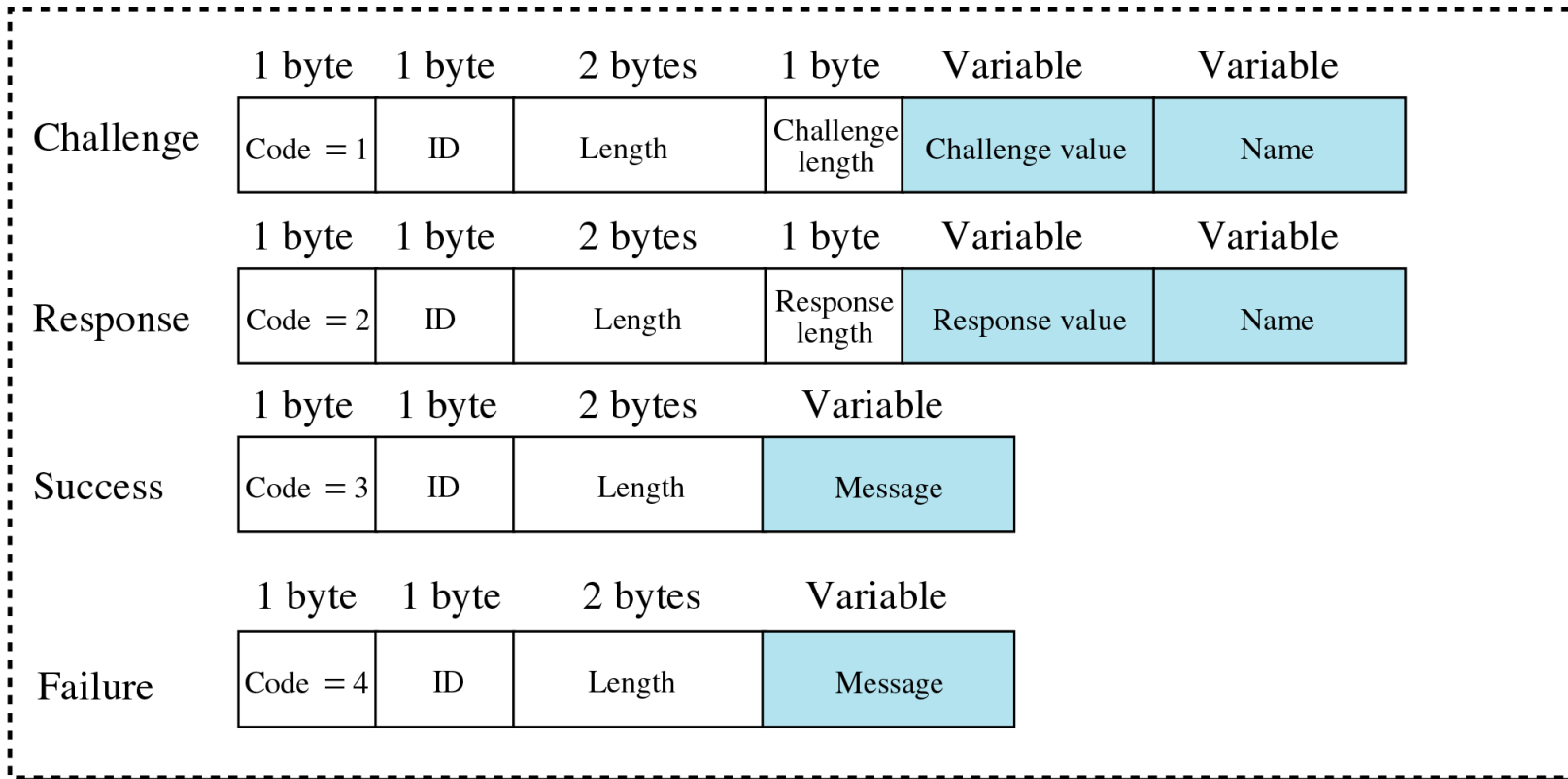
Three-way handshake

- 1) Sistem mengirimkan paket challenge
 - 2) Pengguna menerapkan fungsi yang telah ditentukan sebelumnya yang mengambil nilai challenge dan password dan menciptakan hasil
 - 3) Sistem melakukan hal yang sama; lalu bandingkan hasilnya dengan hasil pengguna
- 

CHAP




CHAP Packets





IPCP (Protokol NCP)

- Setelah link dibangun dan security ditetapkan, koneksi *network layer* perlu dibuat
 - IPCP, atau Internetwork Protocol Control Protocol, merupakan NCP (Network Control Protocol)
- 



IPCP

Seven packet types:

Configure-request (01)

Configure-ACK (02)

Configure-NAK (03)

Configure-reject (04)

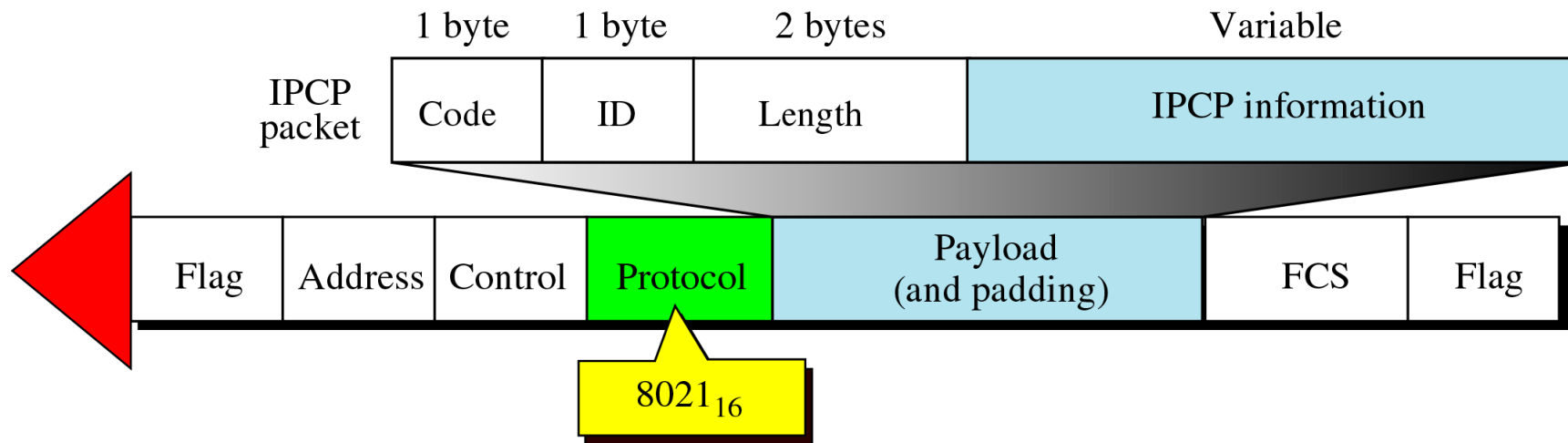
Terminate-request (05)

Terminate-ACK (06)

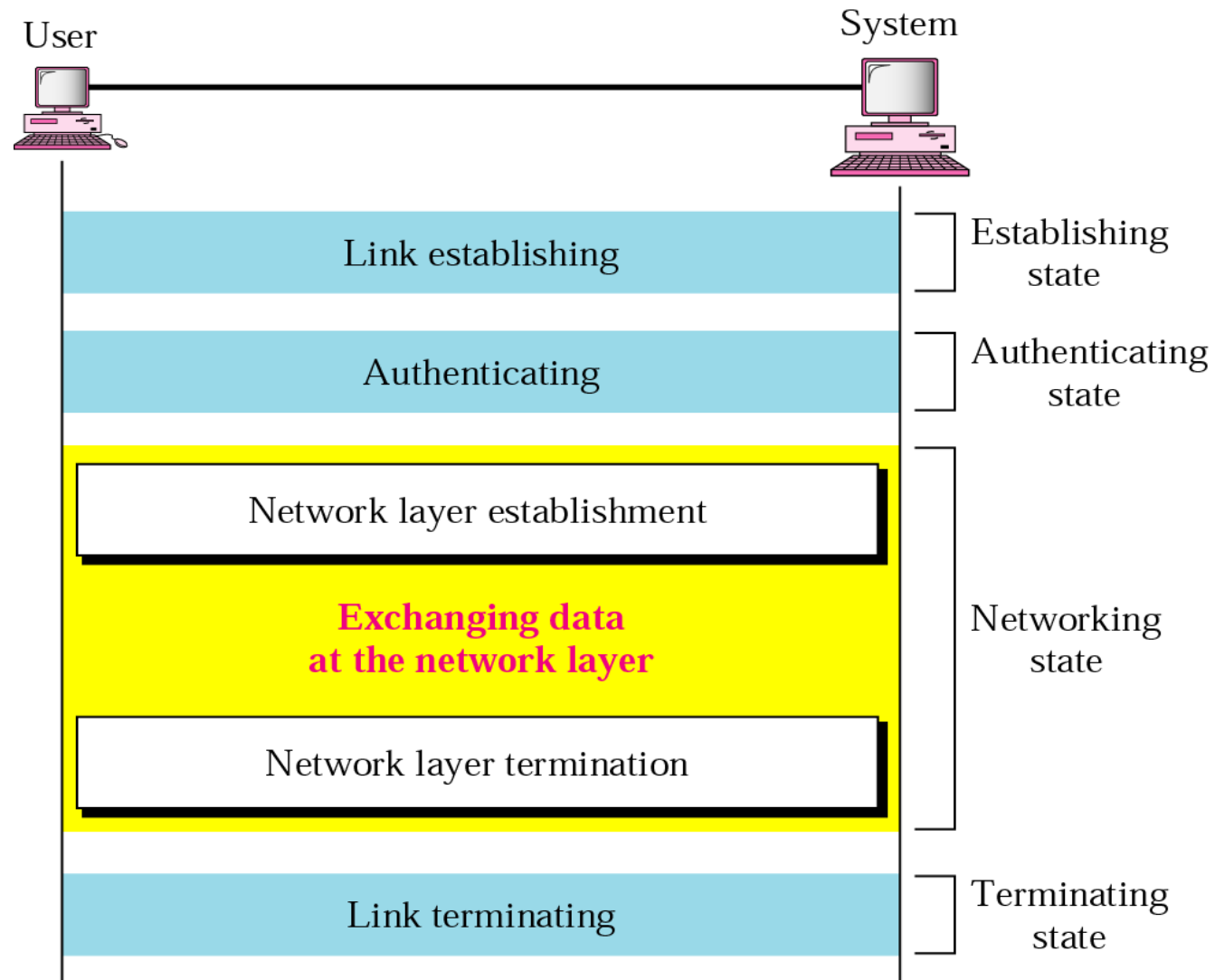
Code-reject (07)



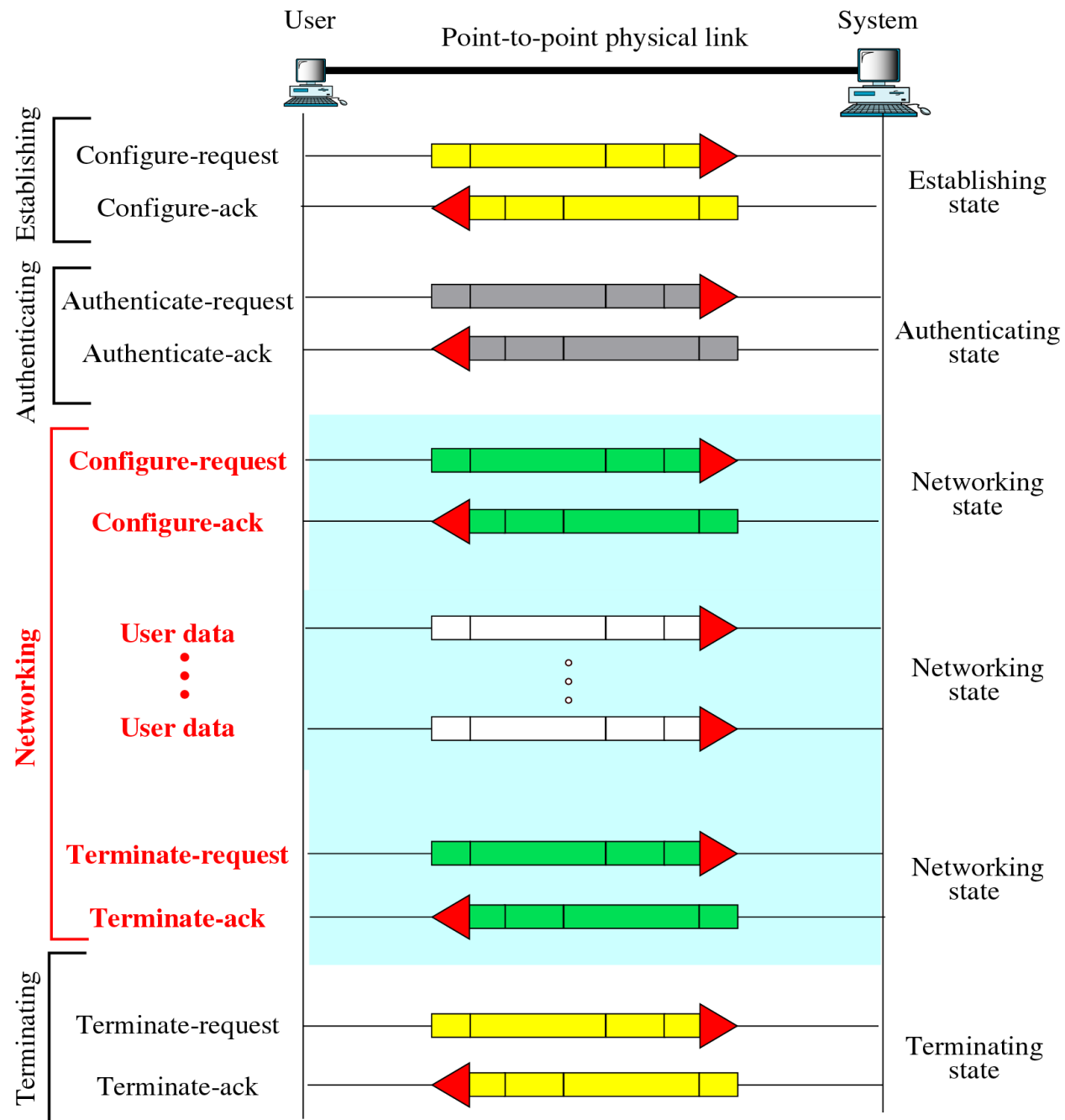
Enkapsulasi Paket IPCP dalam PPP Frame



Contoh



Contoh





References

- *Data Communications and Networking*, 5th Edition, Behrouz A. Forouzan, McGraw Hill, 2013
 - *Data and Computer Communications*, 10th Edition, William Stallings, Pearson Education, 2014
 - *Computer Networking: A Top Down Approach*, 7th Global Edition, James F. Kurose & Keith W. Ross, Pearson Education, 2017
- 