FLOW CONTROL & A VARIABLE

Telkom University

Budhi Irawan, S.Si, M.T

Kendali Aliran (Flow Control)

- Fungsi lain yang diperlukan dalam mentransmisikan data di suatu ***link*** adalah **Kendali Aliran**
- Dibutuhkan terutama jika aliran data dari ***yang cepat ke yang lambat***, dimana aliran data harus diatur agar penerima tidak ***overflow***

Kendali Aliran (Flow Control)

Model Kendali Aliran

Jenis Flow Control

1. Start-stop
2. Mengatur Besarnya Aliran

1. START - STOP

- Aliran data diatur sesuai dengan ***permintaan pihak penerima***, jika ***penerima*** merasa ***buffer penerimaannya penuh***, maka ia akan mengirim ***sinyal stop*** *ke* ***pengirim***, dan jika buffer penerimaannya ***kosong***, ia akan mengirim ***sinyal start*** ke pengirim.

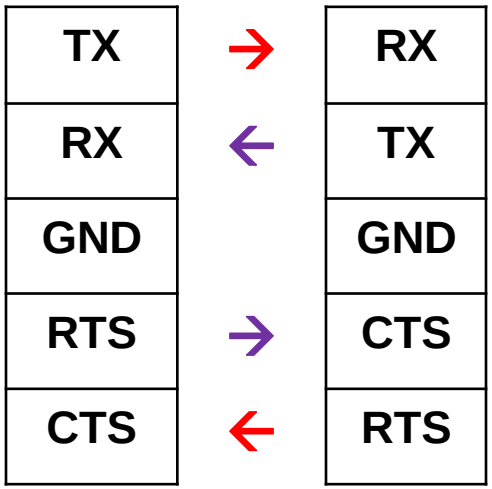
1. START - STOP

- Teknik ini sederhana, relatif mudah di implementasikan
- Contoh dari teknik ***start-stop*** umum :
  - ***RTS, CTS*** (***Hardware Flow Control***)
  - ***X-on, X-off*** (***Software Flow Control***)

Start – Stop (RTS-CTS)

- **RTS – CTS** (***Request to Send – Clear to Send***), digunakan saluran tambahan untuk mengkomunikasikan informasi kendali aliran selain Tx, Rx & GND
- Dirancang untuk berkomunikasi dengan ***modem*** yang ***lebih lambat*** dari interface ***RS-232.***

Koneksi fisik

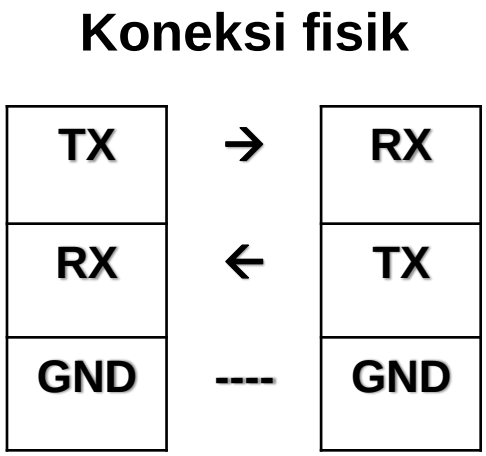


Pertukaran sinyal

- **RTS** dikirim pada saat akan melakukan **pengiriman data**
- Jika dijawab ***CTS*** maka ***TX aktif***

Start – Stop (X-On – X-Off)

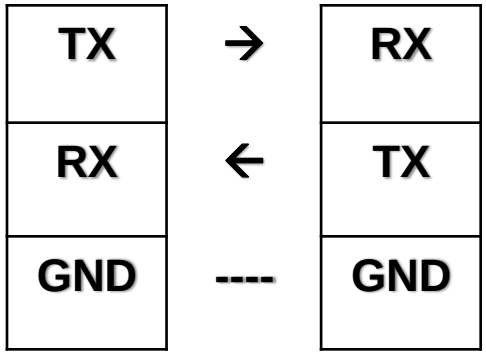
- ***Software (X-on, X-off)***, digunakan ***karakter-karakter tertentu*** dalam bertukar informasi ***kendali aliran***
- Lebih sedikit membutuhkan ***koneksi fisik*** (***2 kabel*** untuk ***satu arah komunikasi***, ***3 kabel*** untuk ***dua arah komunikasi***)



Start – Stop (X-On – X-Off)

- **Algoritma kerja disisi pengirim**
  - Tunggu ***X-ON***
  - Kirim TX
  - Jika mendapat ***X-OFF***, berhenti kirim
- **Algoritma kerja disisi penerima**
  - Periksa buffer penerimaan
  - Jika kosong kirim ***X-ON***, jika penuh kirim ***X-OFF***

Koneksi fisik



Kekurangan Flow Control Start-Stop

- Teknik kendali aliran **Start-Stop** mempunyai kelemahan ***trafik yang terjadi menjadi padat*** yang menyebabkan naiknya peluang ***kongesti*** di jaringan, tidak cocok untuk komunikasi jarak jauh (melalui banyak link).

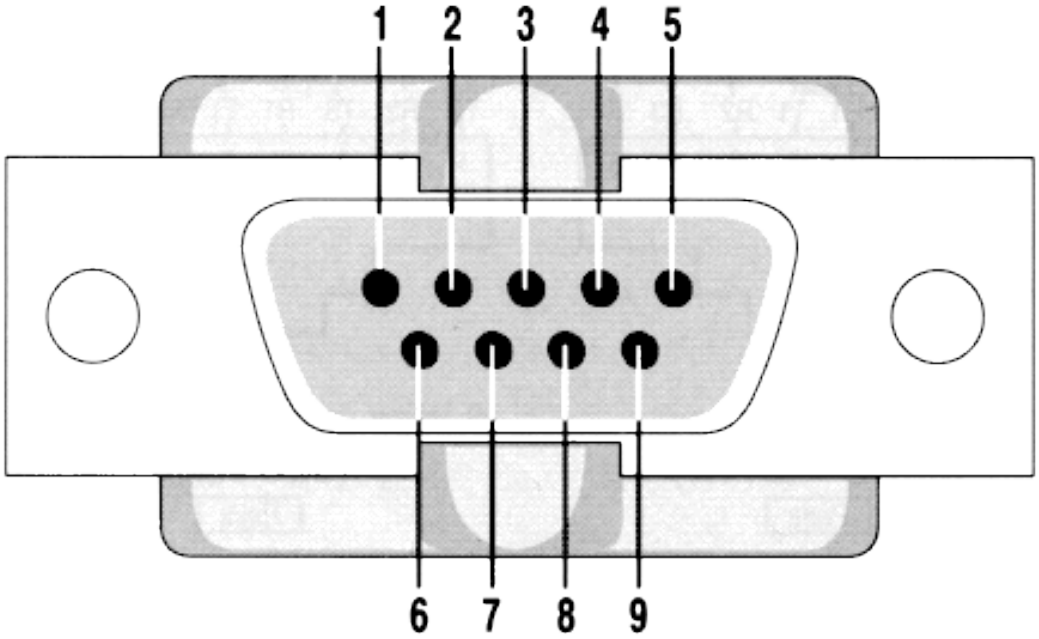
Flow Control Adaptif

- Dikembangkan teknik pengendalian aliran yang ***lebih adaptif*** sesuai dengan kondisi jalur transmisi yang dilewati, sehingga data dapat ditransmisikan dengan jumlah yang ***'cukup'***, **"tidak berlebih"** dan ***"tidak kurang"***.
- Teknik ini meningkatkan ***efisiensi bandwidth*** yang pada ujungnya akan mengurangi terjadinya ***kongesti jaringan***.

2. MENGATUR BESAR ALIRAN

- Aliran data diatur berdasarkan ***besar bandwitdh saluran*** saat itu, teknik ini bekerja berdasarkan ***feedback dari penerima*** yang '***mengukur' laju data*** yang mampu ***diterima.***
- ***Relatif lebih rumit*** dari teknik start-stop
  Contoh : ***(sliding) window***

Flow Control pada RS 232



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

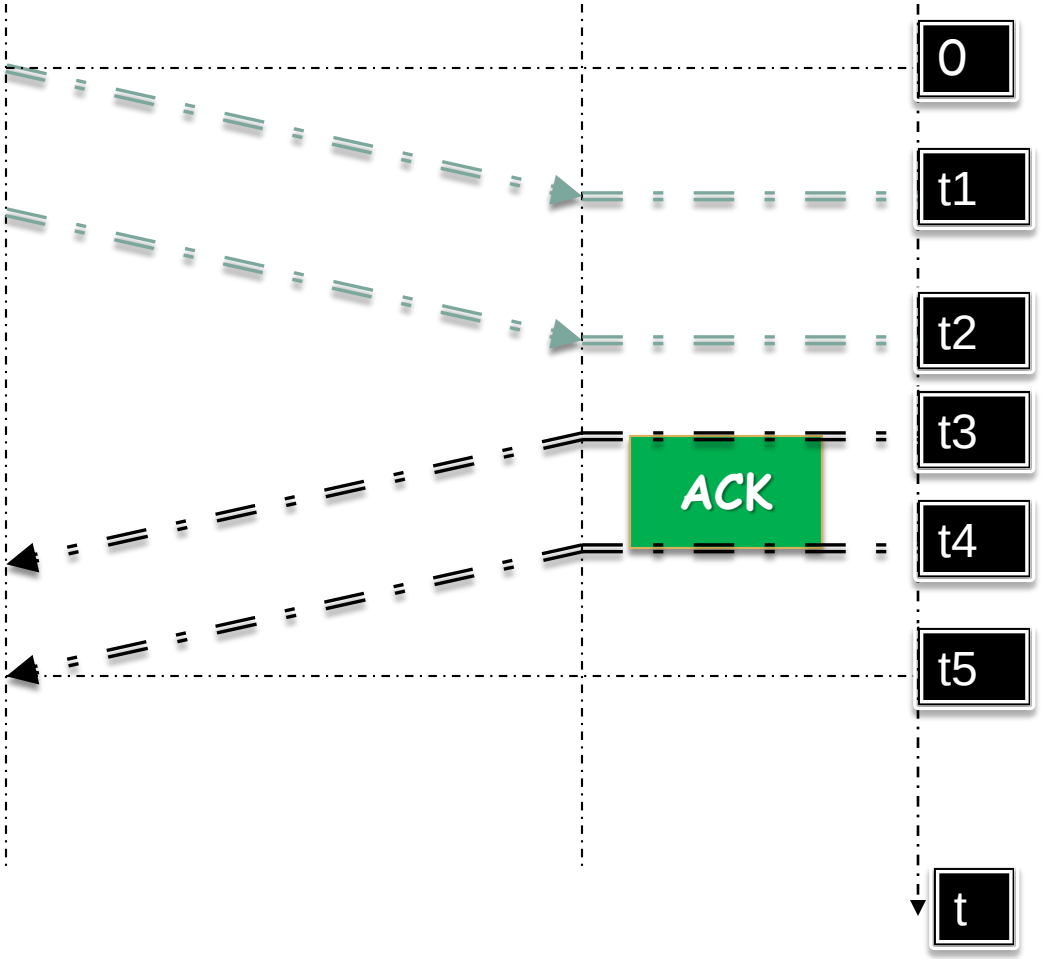
Perhitungan Waktu Transmisi Paket

- Penggunaan ***ARQ*** (***Automatic Repeat Request***) menyebabkan ***waktu transmisi*** sama dengan waktu dari ***mulai paket dikirim*** sampai dengan ***waktu diterimanya ACK*** (***Acknowledge***) oleh pengirim
- Komponen waktu transmisi bisa dihitung dengan penyederhanaan pada skema sebagai berikut :

Perhitungan Waktu Transmisi Paket

Paket

0-t1 = waktu propagasi
t1-t2 = waktu paket (waktu pengeluaran bit 1 sampai terakhir)
t2-t3 = waktu deteksi
t3-t4 = waktu paket ack
t4-t5 = waktu propagasi



Perhitungan Waktu Transmisi Paket

- ***Waktu Propagasi***
  Waktu yang diperlukan untuk ***1 bit menempuh*** jarak pengirim-penerima

tpro = jarak/kecepatan

- ***Waktu Paket***
  Waktu yang diperlukan untuk ***mengeluarkan semua bit*** pada paket tersebut

tpak = panjang paket (bit)/bitrate

Perhitungan Waktu Transmisi Paket

- ***Waktu deteksi***
  Waktu yang dibutuhkan oleh penerima untuk ***menentukan*** paket yang diterima ***benar*** atau ***salah***

  tdet = Tergantung Kecepatan Komputasi

- ***Waktu Paket Acknowledge (Ack)***
  Waktu yang dibutuhkan oleh pengirim untuk menerima acknowledge : **tack = panjang paket ack/bitrate**

ttotal = 2tpro + tpak + tdet + tack

Rumus Penyederhanaan

- Didapatkan dari beberapa kasus percobaan, **ttotal** transmisi paket didominasi oleh **tpak** atau **tpro** tergantung dari jarak transmisi,
- **tdet** sangat bergantung pada kecepatan perhitungan penerima yang cenderung semakin kesini semakin cepat,

Rumus Penyederhanaan

- **tack** relatif dapat diabaikan karena panjang paket ack jauh lebih kecil dari panjang paket data.

  Sehingga : $t_{total} \approx 2\ t_{pro} + t_{pak}$

Variabel a

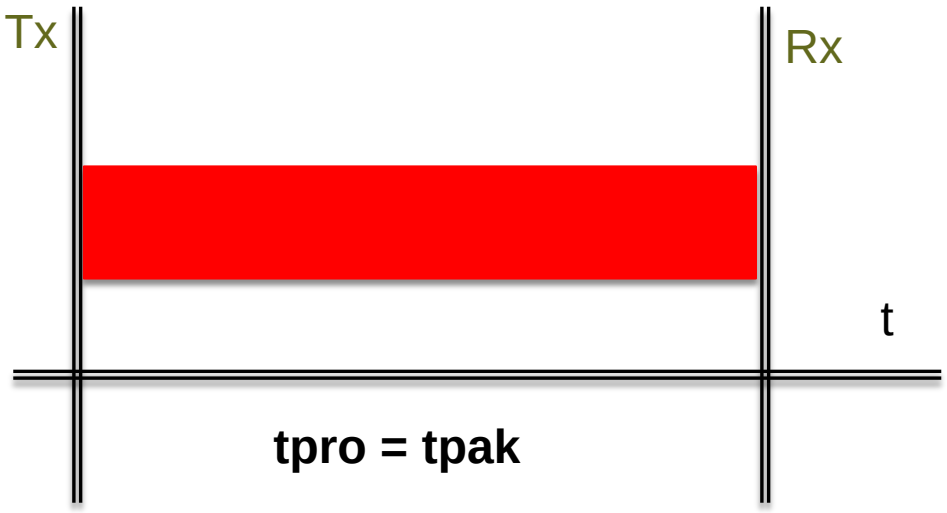
- Untuk memudahkan penulisan rumus komponen waktu transmisi paket dan memperlihatkan suatu variabel penentu hasil perhitungan ***utilitas link***, maka dibuatlah **variabel a**

Dengan :

$$a = tpro/tpak$$

Kasus a = 1

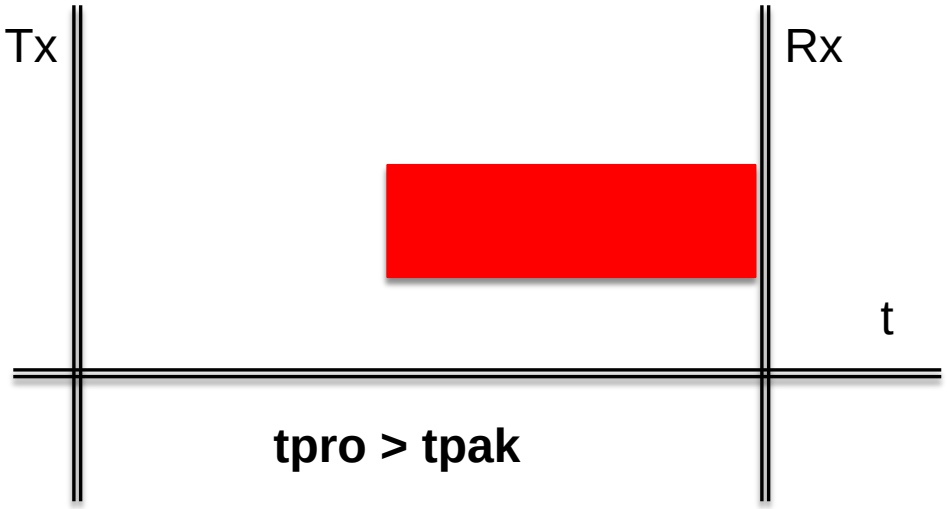
- **a = 1**, menyatakan ***gejala fisik saluran akan dipenuhi oleh paket***, dalam arti ***bit pertama mulai diterima saat bit terakhir dikirim***



- Terjadi jika ***waktu untuk menghasilkan paket*** sama persis dengan ***waktu propagasi***

Kasus a > 1

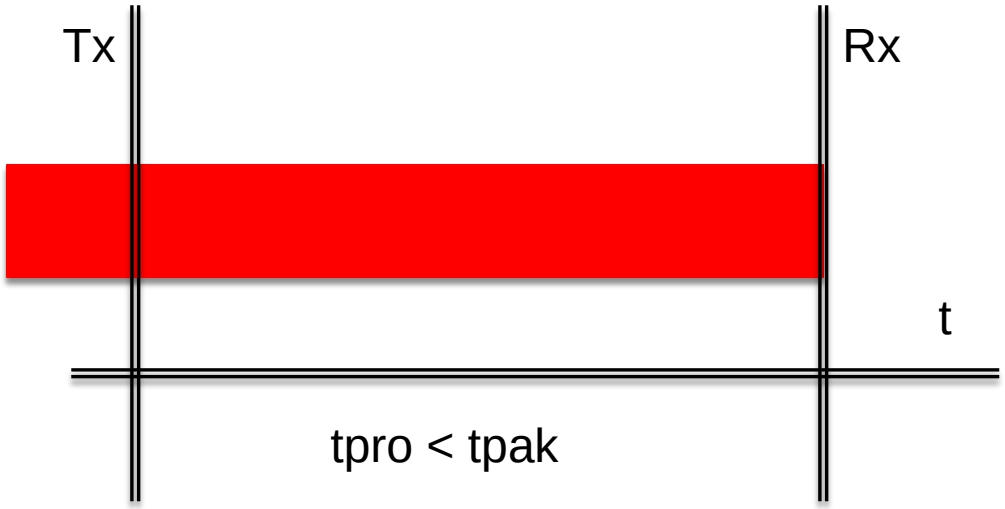
- **a > 1**, menyatakan ***gejala fisik saluran akan sebagian kosong***, dalam arti ***paket telah selesai dihasilkan saat bit pertama diterima***



- Terjadi jika ***waktu untuk menghasilkan paket*** lebih kecil dari ***waktu propagasi***

Kasus a < 1

- **a < 1**, menyatakan ***gejala fisik saluran akan penuh oleh paket lebih lama dari waktu propagasinya***



- Terjadi jika ***waktu untuk menghasilkan paket lebih lama*** dari ***waktu propagasi***

Backward Error Control (BEC)

- Metode BEC merupakan suatu ***metode kendali kesalahan*** dimana penerima akan melakukan ***pengecekan*** terhadap data yang datang.
- Jika data yang diterima diketahui telah mengalami ***error***, maka penerima akan meminta pengirim untuk ***mengirimkan kembali*** data tersebut sampai data yang diterimanya benar.

Backward Error Control (BEC)

- Pada metode ini, keputusan untuk mengirimkan ulang atau tidak tergantung ***feedback dari penerima***.
- Saat pengirim mengirimkan data, maka pengirim akan ***menyimpan salinan data*** tersebut sampai ada ***feedback dari penerima*** bahwa telah tiba dengan ***benar*** ke penerima.

Backward Error Control

Kemampuan ***deteksi kesalahan*** digunakan untuk melakukan perbaikan kesalahan ***(error control)*** dengan cara meminta ***pengiriman ulang*** jika paket yang diterima ***salah/error***

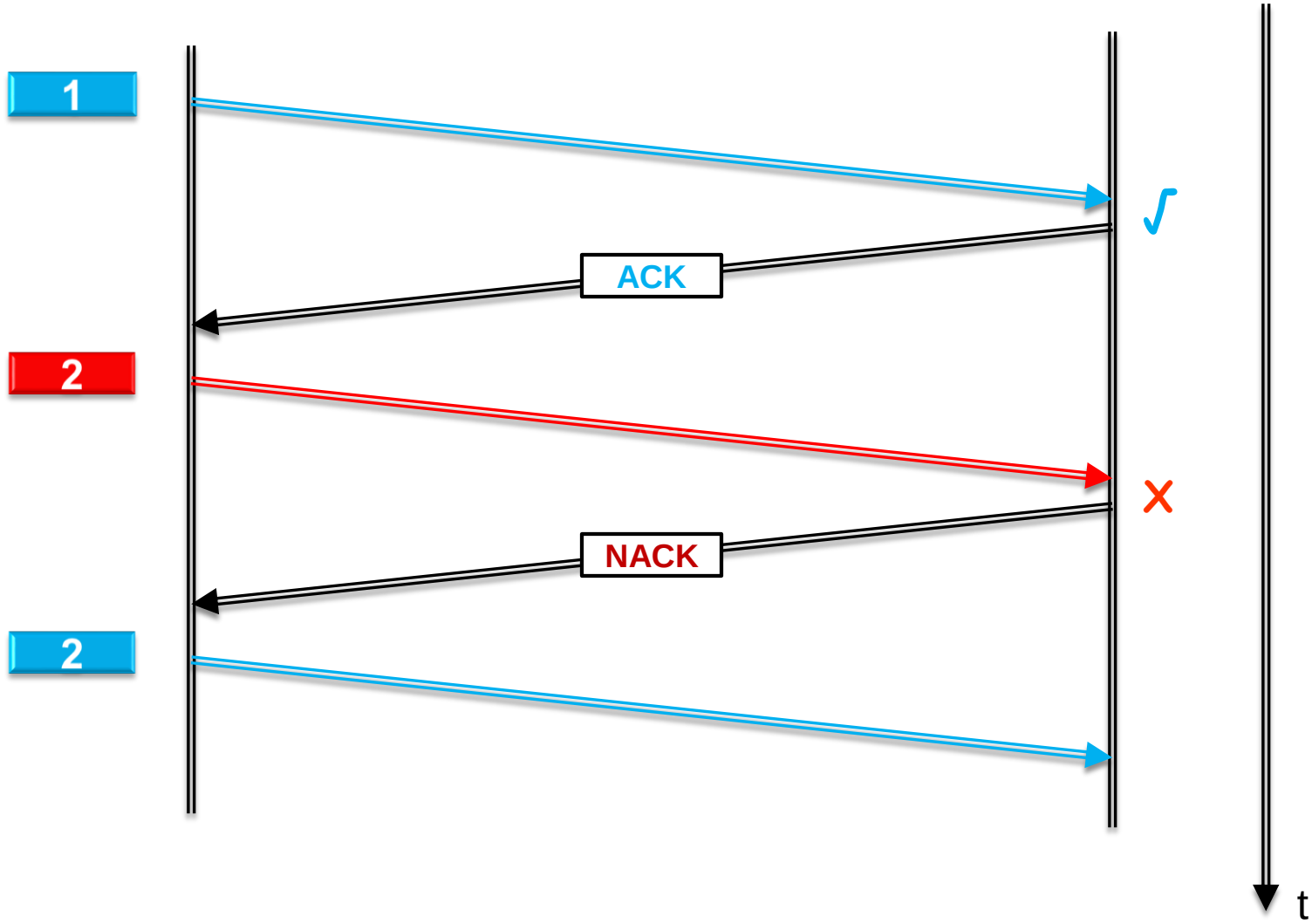
Backward Error Control (BEC)

1. Idle Automatic Request
2. Go Back End
3. Selective Repeat

1. Idle ARQ

- Pada metode ***Idle ARQ***, paket yang sampai ke penerima akan diperiksa terlebih dahulu apakah mengalami ***error atau tidak***.
- ***Feedback*** ini akan dikirimkan oleh penerima ke pengirim.
- Apabila paket sampai dengan ***benar*** maka penerima memberikan ***feedback ACK (Acknowledge)*** tetapi apabila error maka penerima memberikan ***feedback NACK (Not Acknowledge)***

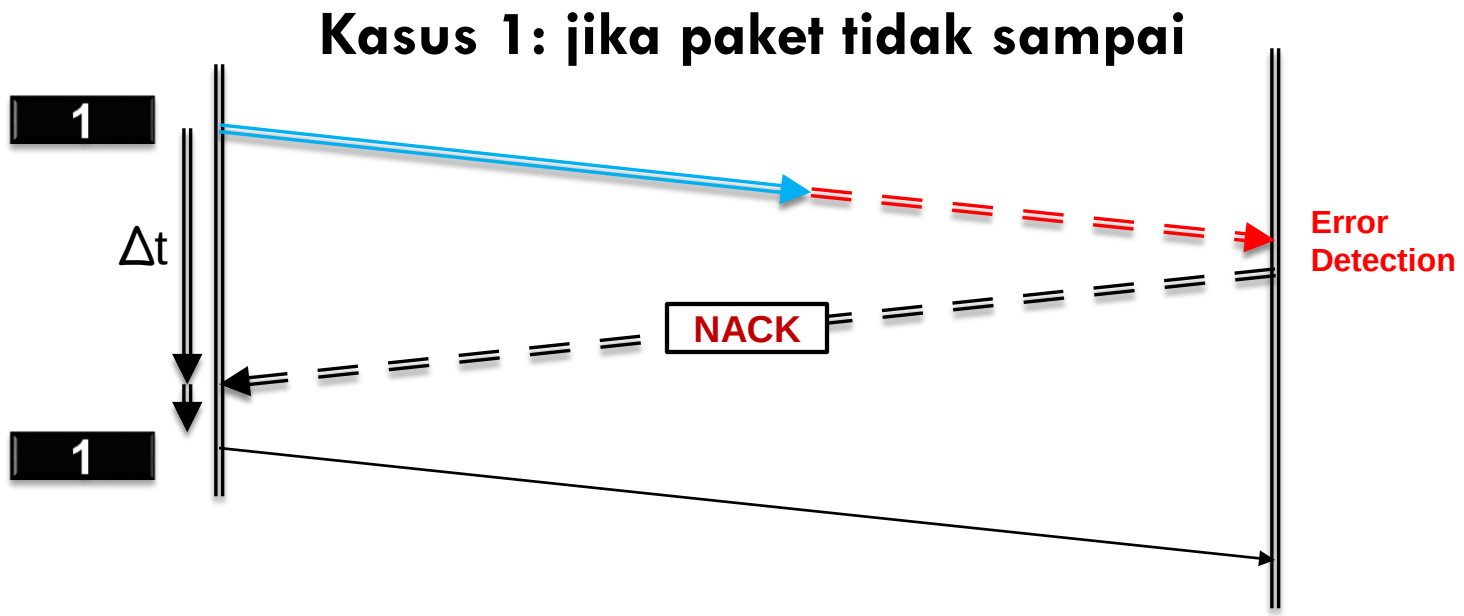
1. Idle ARQ



1. Idle ARQ

- Apabila suatu saat paket ***tidak pernah sampai ke penerima*** , maka penerima tidak akan mengirim ***feedback***.
- Pada kondisi ini pengirim akan menunggu ***selama waktu tertentu*** yang disebut dengan ***waktu timeout*** yang kemudian mengirimkan paket yang sama.

1. Idle ARQ



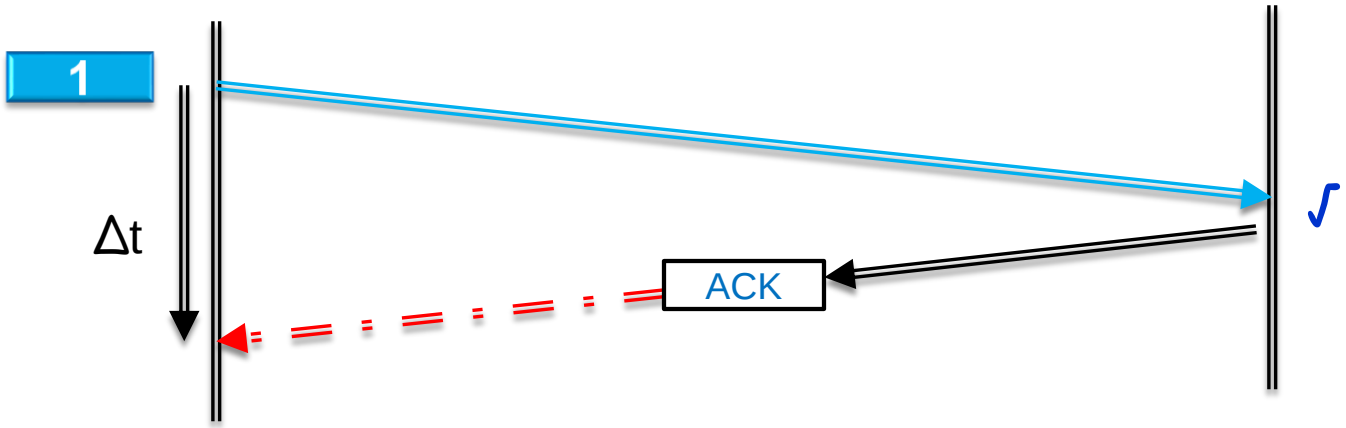
Pengirim menunggu feedback sampai $\Delta t + \delta t$, jika tidak ada respon maka pengirim harus ***mengirimkan kembali*** paket tersebut.

1. Idle ARQ

- Pada metode ini paket akan diperiksa satu per satu dan paket berikutnya akan dikirimkan jika pengirim telah menerima ***feedback*** berupa ***ACK*** bahwa paket sebelumnya telah sampai dengan benar pada penerimanya.
- Metode ini cocok digunakan untuk ***saluran yang tingkat errornya tinggi***.

1. Idle ARQ

Kasus 2: feedback tidak sampai



Diperlakukan sama dengan kondisi kasus 1 (**time-out**)

1. Idle ARQ

Kapankah pengirim mengirim ulang paket ??

- Jika mendapat feedback **NAK**
- Jika *timeout*
- Jika mendapat *feedback yang tidak dimengerti*

Kesimpulan : pengirim mengirim ulang paket Jika ***tidak*** mendapat **ACK**

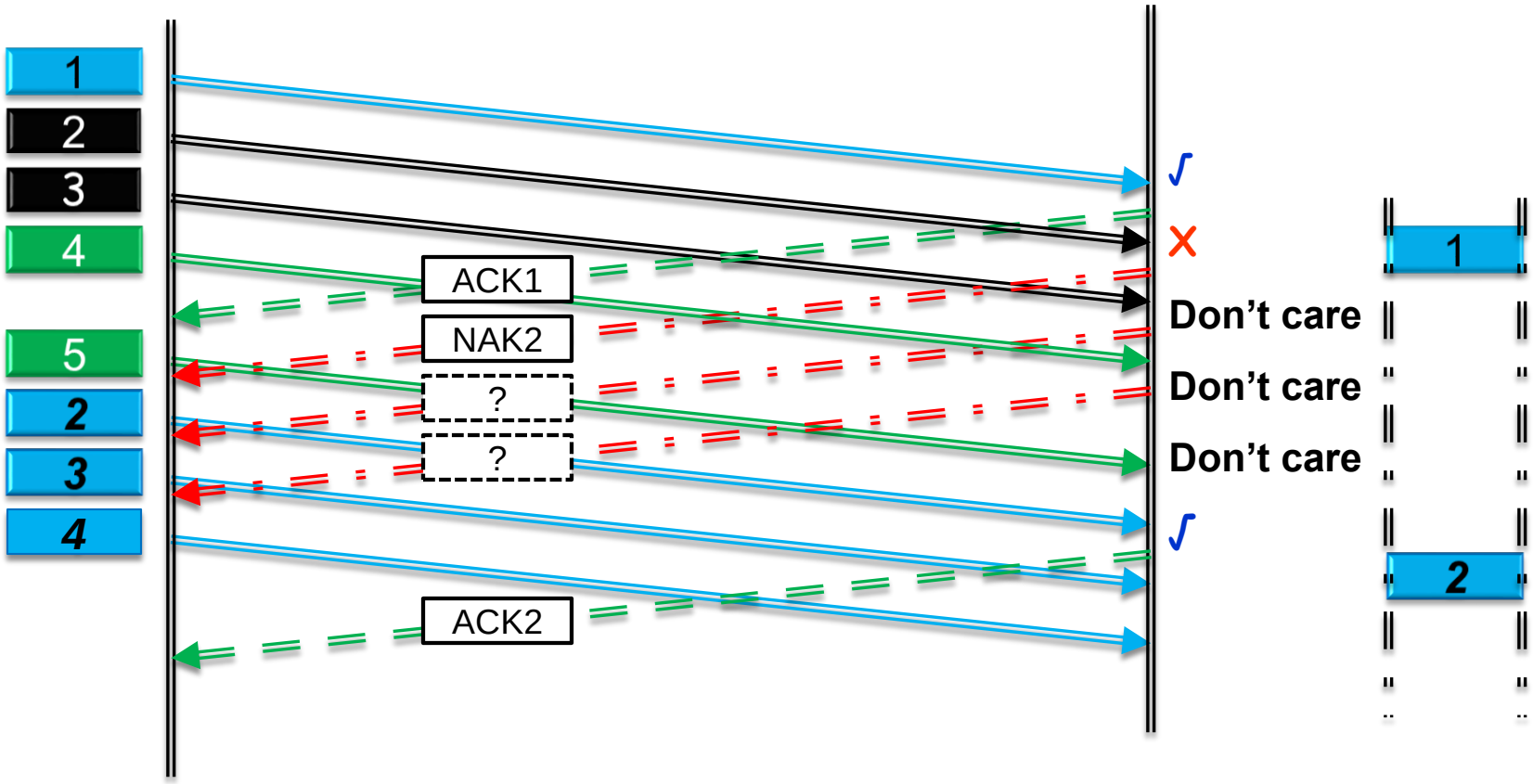
2. Go Back N

- Pada metoda ***Go Back End***, pengirim akan mengirimkan beberapa paket secara berurutan tanpa menunggu paket pertama mendapat ***feedback*** dari penerima.
- Pengecekan paket yang sampai dilakukan oleh penerima
- Setelah penerima ***mengidentifikasi paket*** tersebut, penerima akan mengirimkan ***feedback*** kepada pengirim

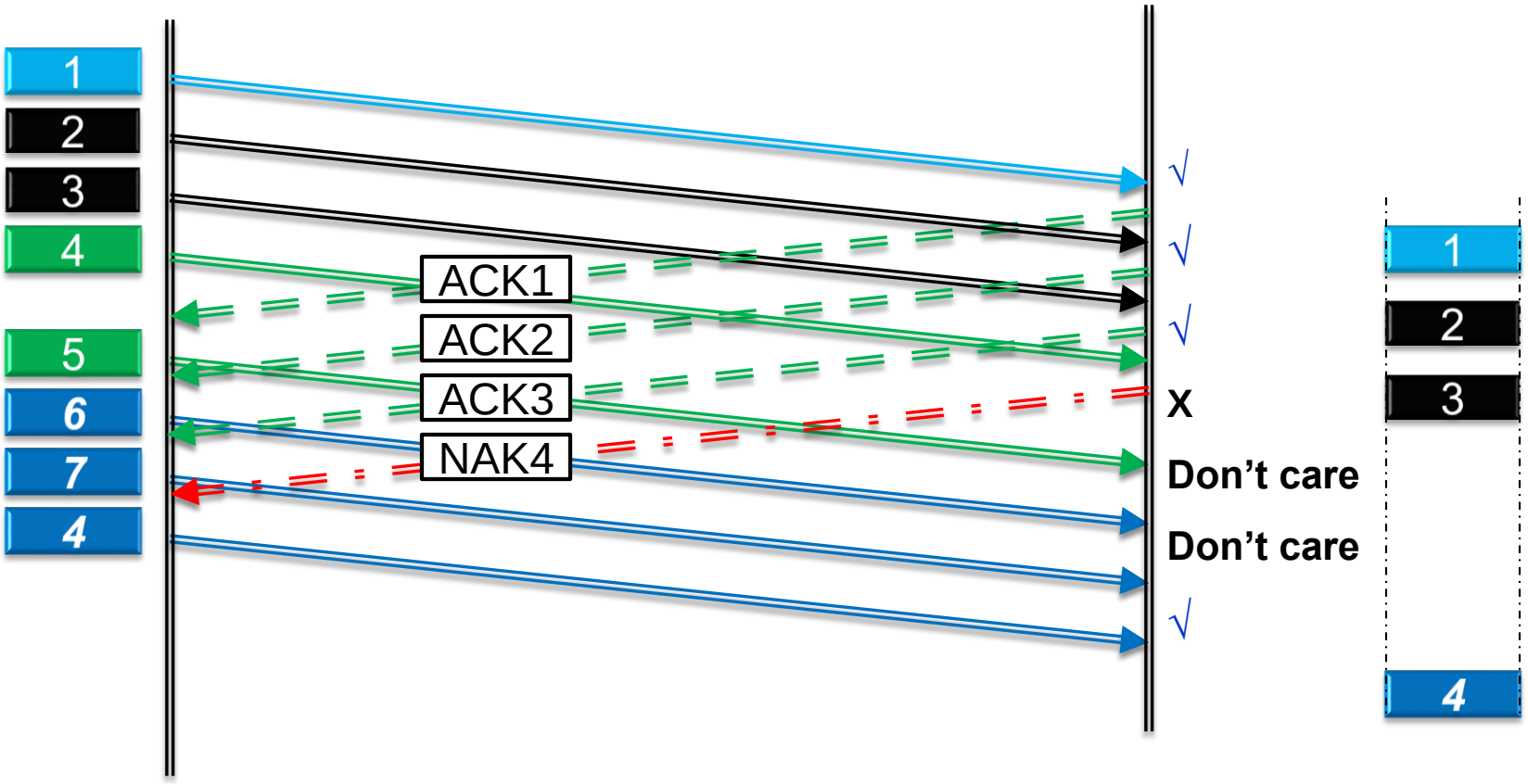
2. Go Back N

- Apabila mendapat ***feedback ACK*** dari penerima , maka pengirim akan mengirimkan paket selanjutnya.
- Apabila mendapat ***feedback NACK*** dari penerima, maka pengirim akan mengirimkan paket mulai dari yang salah tersebut
- ***Keterurutan*** data di sisi penerima akan selalu tetap ***terjaga***.

2. Go Back N



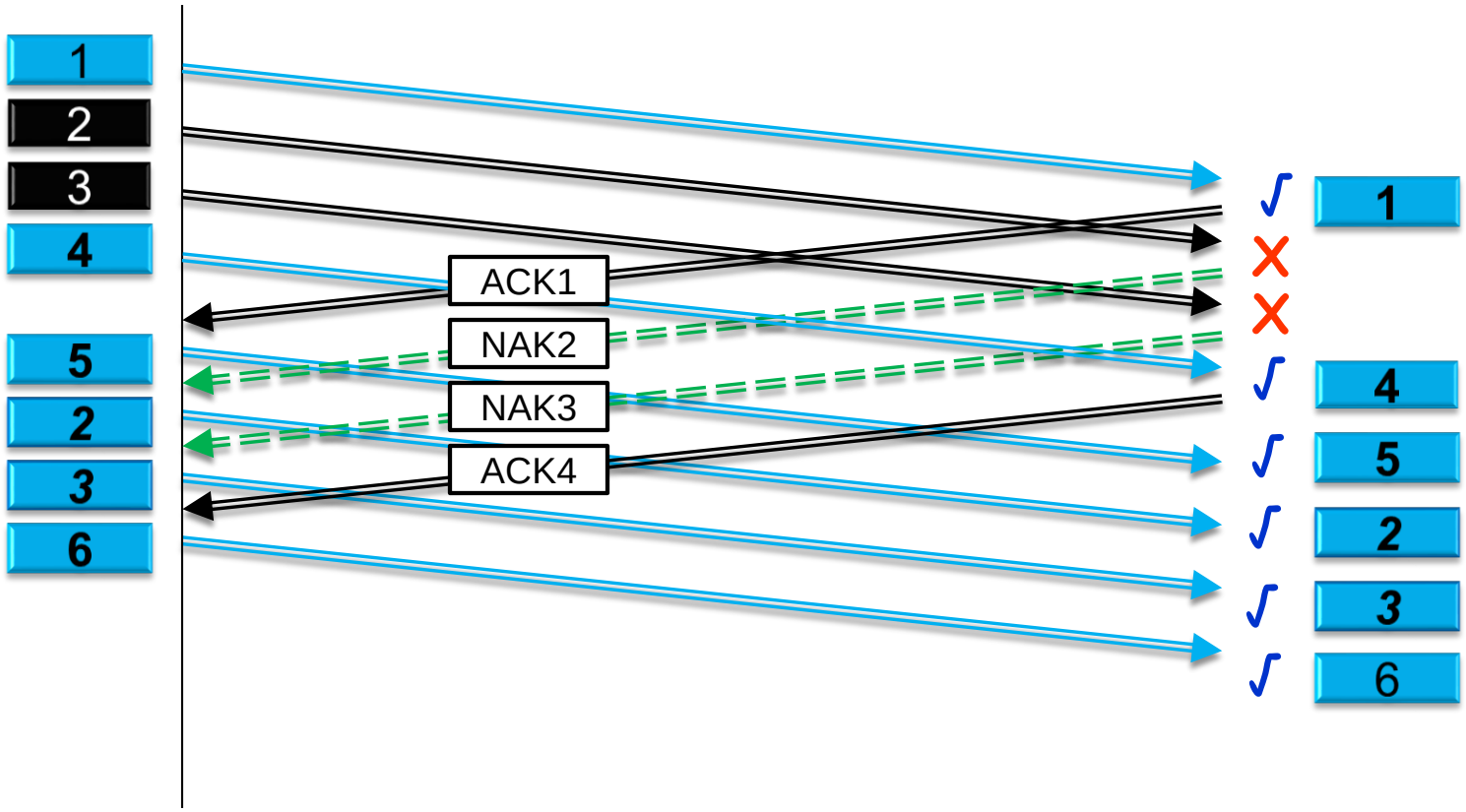
2. Go Back N (Other Case)



3. Selective Repeat

- Metode ini mirip dengan ***Go Back End*** hanya saja pengiriman dan pengecekan paket tetap dilakukan walaupun ada paket yang ***error*** saat diterima.
- Pada ***selective repeat*** pengiriman ulang paket hanya dilalukan untuk ***paket yang salah*** saja.
- Paket data bisa saja untuk diterima dengan ***tidak berurutan***

3. Selective Repeat



Sliding Window

- Salah satu teknik yang sejak awal dibuatnya protokol internet adalah ***teknik sliding window***
- **Window** = angka jumlah pengiriman paket saat ini
- ***Window* = *3*** artinya **satu kali kirim maksimum 3 paket**

Cara Kerja Sliding Window

- Penerima akan menetapkan jumlah **window** yang diterimanya berdasarkan ***tingkat keberhasilan penerimaan paket***, kebijakan yang ditetapkan oleh lapis aplikasi, dll
- Pengirim kemudian akan mengirim paket sesuai dengan jumlah **window** yang ditetapkan ***penerima***

Besarnya Window

Untuk setiap ***algoritma ARQ***, ukuran window yang sesuai adalah :

ARQ	**Window Kirim**	**Window Terima**
Idle RQ	1	1
Selective Repeat	N	N
Go Back N	N	1

Pengaruh Window pada Proses Pengiriman Paket