



# Jaringan Komputer (KP041)

Edisi kerjasama Univ. Budi Luhur dengan  
Univ Tribuana Kalabahi

## Pertemuan 5

TRANSMISI ASINKRON, SINKRON,  
DETEKSI KESALAHAN

# Sinkronisasi

- ❑ **Synchronisasi** adalah salah satu tugas utama dari komunikasi data
- ❑ Suatu transmitter mengirim message 1 bit pada suatu waktu melalui suatu medium ke receiver
- ❑ Receiver harus mengenal awal dan akhir dari blok-blok bit dan juga harus mengetahui durasi dari tiap bit sehingga dapat men-sampel line tersebut dengan timing yang tepat untuk membaca tiap bit.

- ❑ Misalkan pengirim (sender) mentransmisi sejumlah bit-bit data.
- ❑ Pengirim mempunyai suatu clock yang mempengaruhi timing dari transmisi bit-bit.
- ❑ Sebagai contoh, jika data ditransmisi dengan 10000 bits per second (bps), kemudian 1 bit akan ditransmisi setiap  $1/10000 = 0,1$  millisecond (ms), sebagai yang diukur oleh clock pengirim.
- ❑ Maka, receiver akan menentukan waktu yang cocok untuk sampel-sampelnya pada interval dari 1 bit time.

- ❑ Pada contoh ini, pen-sampling-an akan terjadi sekali setiap 0,1 ms
- ❑ Jika waktu pen-sampling-an berdasarkan pada clocknya sendiri, maka akan timbul masalah jika clock-transmitter dan receiver tidak disamakan dengan tepat
- ❑ Jika ada perbedaan 1 persen (clock receiver 1 persen lebih cepat atau lebih lambat daripada clock transmitter), maka pen-sampling-an pertama 0,001 ms meleset dari tengah bit (tengah bit adalah 0,05 ms dari awal dan akhir bit).

- ❑ Setelah sampel-sampel mencapai 50 atau lebih, receiver akan error karena pen-sampling-annya dalam bit time yang salah ( $50 \times 0,001 = 0,05 \text{ ms}$ )
- ❑ Untuk perbedaan timing yang kecil, error akan terjadi kemudian, tetapi kemudian receiver akan keluar dari step transmitter jika transmitter mengirim aliran bit yang panjang dan jika tidak ada langkah-langkah yang men-synchron-kan transmitter dan receiver

# Ada 2 mode transmisi data :

- Paralel transmission

Antara pengirim dan penerima dihubungkan oleh lebih dari 1 jalur transmisi



- ▶ contoh : RAM ke Prosesor, HD ke RAM, Peripheral internal komputer

## ❑ Serial Transmission

Antara pengirim dan penerima dihubungkan hanya oleh 1 jalur transmisi untuk pengiriman jarak jauh



Contoh : Keyboard ke Prosesor, Prosesor ke Printer, LAN, MAN, WAN

## **Masing-masing punya keunggulan**

Paralel :

Cepat, karena beberapa bit dapat dikirim bersamaan melalui jalur yang berbeda. Tetapi sangat mahal, sehingga hanya untuk pengiriman jarak pendek

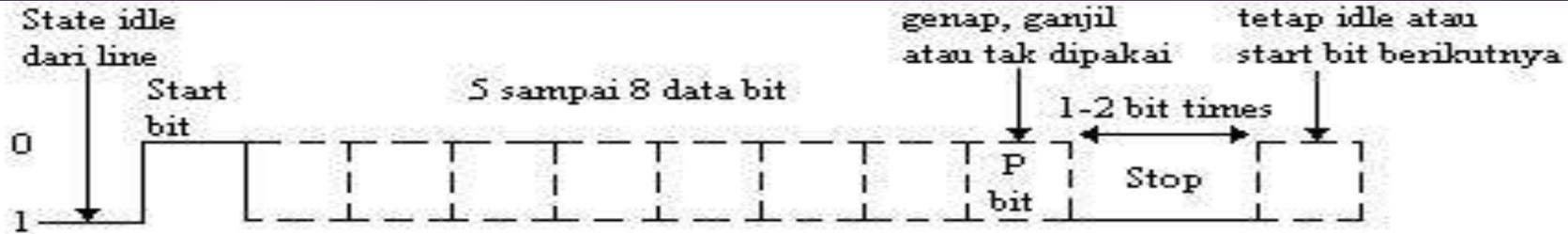
Serial :

Lambat tetapi murah, sehingga dikembangkan lebih lanjut

# Transmisi Asinkron

- ❑ Tidak perlu sinkronisasi clock antara pengirim dan penerima
- ❑ Strategi dari metode ini yaitu mencegah problem timing dengan tidak mengirim aliran bit panjang yang tidak putus-putusnya.
- ❑ Melainkan data ditransmisi per karakter pada suatu waktu, dimana tiap karakter adalah 5 sampai 8 bit panjangnya.
- ❑ Timing atau synchronisasi harus dipertahankan antara tiap karakter; receiver mempunyai kesempatan untuk mensynchronkan awal dari tiap karakter baru.

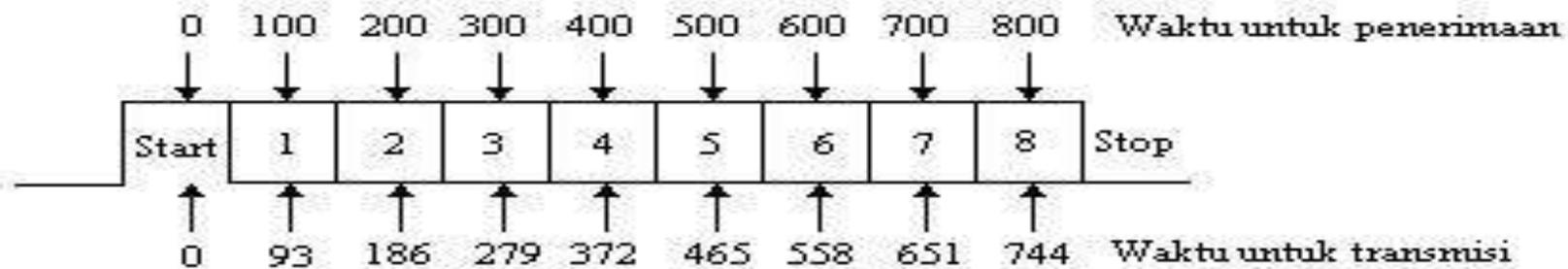
Gambar berikut : menjelaskan suatu contoh untuk teknik Asynchronous :



(a) Format karakter



(b) 8 bit aliran karakter asynchronous



(c) Efek dari timing error

- ❑ Ketika tidak ada transmisi karakter, line antara transmitter dan receiver dalam keadaan "idle". **Idle** adalah ekuivalen untuk elemen pensinyalan bagi binary '1'
- ❑ Awal dari suatu karakter diisyaratkan oleh suatu *start bit* dengan binary '0'. Kemudian diikuti oleh 5 sampai 8 bit yang membentuk karakter tersebut. Bit-bit dari karakter itu ditransmisi dengan diawali *least significant bit (LSB)*
- ❑ Biasanya, bit-bit karakter ini diikuti oleh suatu *parity bit* yang berada pada posisi *most-significant-bit (MSB)*

- ❑ Parity bit tersebut diset oleh transmitter sedemikian seperti total jumlah binary '1' dalam karakter; termasuk parity bit-nya, adalah genap (even parity) atau ganjil (odd parity), tergantung pada konversi yang dipakai
- ❑ Elemen terakhir yaitu *stop*, yang merupakan suatu binary '1'
- ❑ Panjang minimum dari stop biasanya 1;1,5 atau 2 kali durasi dari bit.
- ❑ Sedangkan maksimumnya tidak dispesifikasikan. Karena stop sama dengan kondisi idle, maka transmitter akan melanjutkan transmisi sinyal stop sampai siap untuk mengirim karakter berikutnya
- ❑ Contohnya adalah : modem, mesin fax, TCP/IP, mail, buletin board, USB

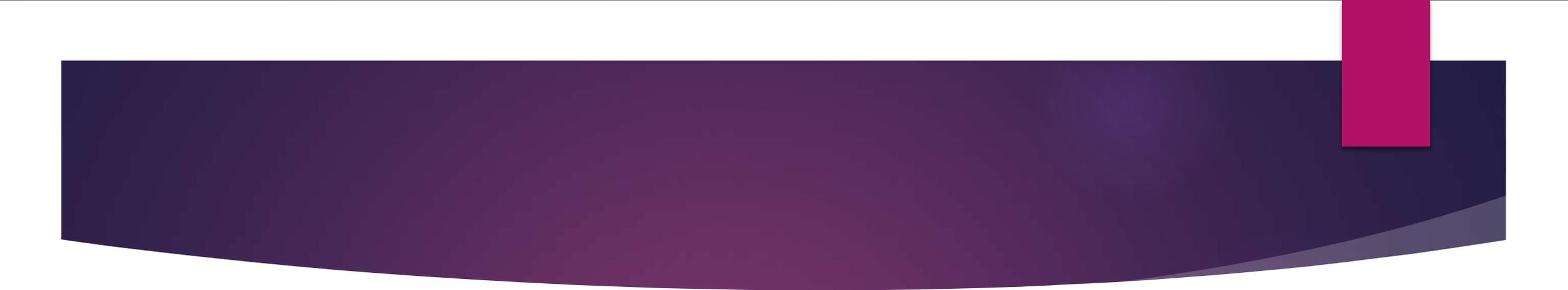
- 
- Keuntungan Komunikasi asynchronous adalah:
    - ▶ Sederhana dan murah tetapi memerlukan tambahan 2 sampai 3 bit per karakter untuk synchronisasi.
    - ▶ Persentase tambahan dapat dikurang dengan mengirim blok-blok bit yang besar antara start dan stop bit, tetapi akan memperbesar kumulatif timing error.

# *Transmisi Synchronous*

- ▶ Proses pengirim dan penerima diatur sedemikian rupa sehingga memiliki pengaturan yang sama, sehingga dapat diterima dan dikirim dengan baik.
- ▶ umumnya pengaturan ini didasarkan pada waktu dalam mengirimkan sinyal.
- ▶ waktu ini diatur oleh denyut listrik secara periodik yang disebut *clock* .
- ▶ dengan kata lain synchronous adalah sistem operasi untuk kejadian yang terjadi pada waktu bersamaan, berkelanjutan dan dapat diprediksi.
- ▶ contoh: chatting , Ethernet

- ❑ Dengan transmisi synchronous, ada level lain dari synchronisasi yang perlu agar receiver dapat menentukan awal dan akhir dari suatu blok data. Untuk itu, tiap blok dimulai dengan suatu pola *preamble* bit dan diakhiri dengan pola *postamble* bit. Pola-pola ini adalah kontrol informasi.
  - ✓ **Frame** adalah data plus kontrol informasi. Format yang tepat dari frame tergantung dari metode transmisinya, yaitu :
- ❑ Transmisi *character-oriented* :
  - ✓ Blok data diperlakukan sebagai rangkaian karakter-karakter (biasanya 8 bit karakter). Semua kontrol informasi dalam bentuk karakter

- ▶ Frame dimulai dengan 1 atau lebih 'karakter synchronisasi' yang disebut SYN, yaitu pola bit khusus yang memberi sinyal ke receiver bahwa ini adalah awal dari suatu blok.
- ▶ Sedangkan untuk postamble-nya juga dipakai karakter khusus yang lain. Jadi receiver diberitahu bahwa suatu blok data sedang masuk, oleh karakter SYN, dan menerima data tersebut sampai terlihat karakter postamble.
- ▶ Kemudian menunggu pola SYN yang berikutnya. Alternatif lain yaitu dengan panjang frame sebagai bagian dari kontrol informasi; receiver menunggu karakter SYN, menentukan panjang frame, membaca tanda sejumlah karakter dan kemudian menunggu karakter SYN berikutnya untuk memulai frame berikutnya.



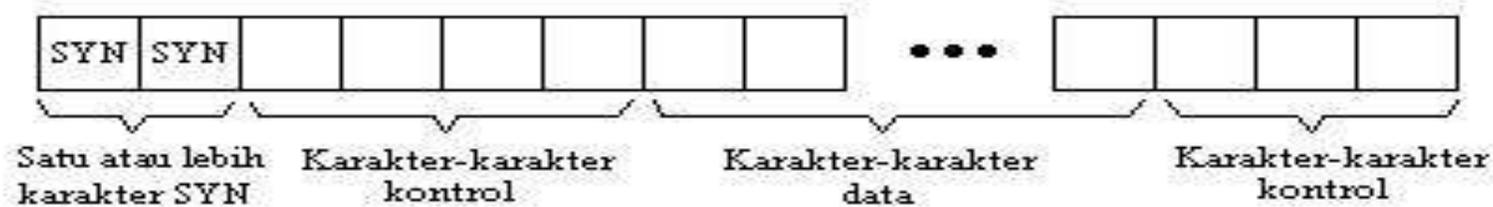
□ Keuntungan transmisi synchronous :

- Efisien dalam ukuran blok data, sedangkan transmisi asynchronous memerlukan 20% atau lebih tambahan ukuran.
- Kontrol informasi kurang dari 100 bit.

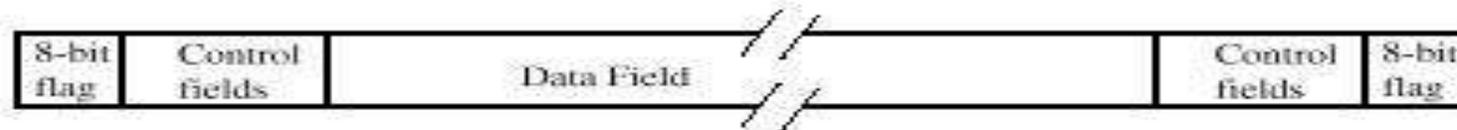
## Transmisi *bit-oriented*

- ❑ Blok data diperlakukan sebagai serangkaian bit-bit
- ❑ Kontrol informasi dalam bentuk 8 bit karakter
  - ✓ Pada transmisi ini, preamble bit yang panjangnya 8 bit dan dinyatakan sebagai suatu flag sedangkan postamble-nya memakai flag yang sama pula.
  - ✓ Receiver mencari pola flag terhadap sinyal start dari frame. Yang diikuti oleh sejumlah kontrol field. Kemudian sejumlah data field, kontrol field dan akhirnya flag-nya diulangi.

- ▶ Perbedaan dari kedua metode diatas terletak pada format defilnya dan kontrol informasinya.
- ▶ Keuntungan transmisi synchronous :
  - Efisien dalam ukuran blok data; transmisi asynchronous memerlukan 20% atau lebih tambahan ukuran.
  - Kontrol informasi kurang dari 100 bit.



(a) Frame Character-oriented



(b) Frame Bit-oriented

# Teknik Deteksi Error

Ketika suatu frame ditransmisikan, tiga kelas probabilitas yang dapat muncul pada akhir penerimaan :

- ❑ Klas 1 ( $P_1$ ) : frame tiba tanpa bit-bit error.
- ❑ Klas 2 ( $P_2$ ) : frame tiba dengan satu atau lebih bit-bit error yang tidak terdeteksi.
- ❑ Klas 3 ( $P_3$ ) : frame tiba dengan satu atau lebih bit-bit error yang terdeteksi dan tidak ada bit-bit error yang tidak terdeteksi.

- ❑ Teknik deteksi error menggunakan error-detecting-code, yaitu tambahan bit yang ditambah oleh transmitter. Dihitung sebagai suatu fungsi dari transmisi bit-bit lain. Pada receiver dilakukan perhitungan yang sama dan membandingkan kedua hasil tersebut, dan bila tidak cocok maka berarti terjadi deteksi error :
- ❑ Tiga teknik yang umum dipakai sebagai deteksi error :
  - Parity bit
  - Longitudinal Redudancy Check
  - Cyclic Redudancy Check

# *Parity Bit*

- ▶ Dalam lingkup masalah detection and correction error terdapat salah satu metode yang dapat digunakan, metode tersebut adalah bit parity.
- ▶ Bit paritas (bit pemeriksa) merupakan salah satu metode untuk mendeteksi kesalahan pada level bit. Bit paritas adalah bit **tambahan** yang diberikan pada akhir byte atau baris terakhir dari kumpulan bit untuk digunakan dalam proses pengecekan kebenaran data pada saat penyimpanan atau proses transmisi.

- ❑ Paritas ganjil (odd parity)

Bit paritas di set menjadi 1, jika jumlah angka 1 dalam kumpulan bit e.g (10001) adalah genap. Sehingga jumlah bit dalam kumpulan bit tersebut menjadi ganjil (10001-1) setelah ditambah bit paritas.

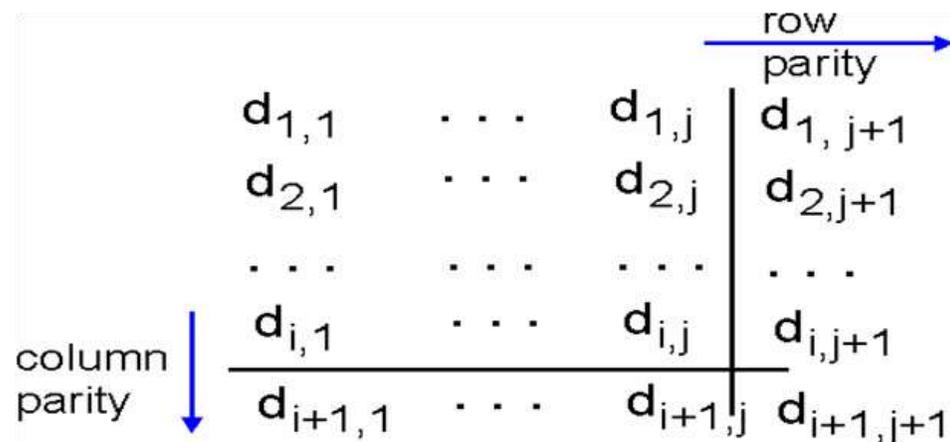
- ❑ Paritas genap (even parity)

Bit paritas di set menjadi 1, jika jumlah angka 1 dalam kumpulan bit e.g (10000) adalah ganjil. Sehingga jumlah bit dalam kumpulan bit tersebut menjadi genap (10000-1) setelah ditambah bit paritas.

- ▶ Nilai bit paritas genap adalah hasil dari XOR sedangkan nilai bit paritas ganjil hasil komplemen atau kebalikan dari hasil XOR.

# Longitudinal Redundancy Check

- Teknik Longitudinal Redundancy Check (LRC) ini merupakan pengembangan teknik parity check. Pada LRC, **data** (payload) disusun menjadi sejumlah baris yang ditentukan (blok), kemudian dilakukan perhitungan bit paritas untuk setiap baris dan setiap kolom. Bit paritas baris ditaruh di ujung kanan, sedangkan bit paritas kolom diletakkan dibagian bawah. Sedangkan urutan transmisi dimulai dari kolom paling kiri ke arah bawah.



# Cyclic redundancy check (CRC)

- ▶ *Cyclic redundancy check (CRC)* merupakan metode verifikasi data yang digunakan oleh komputer untuk memeriksa data pada cakram, misalnya cakram keras (*hard disk*) dan cakram optik (seperti DVD dan CD). Kesalahan (*error*) CRC bisa disebabkan oleh beberapa masalah yang berbeda: *registry* yang rusak, cakram keras yang kacau, kegagalan dalam memasang program, atau berkas yang tidak diatur dengan benar. Apa pun penyebabnya, kesalahan CRC merupakan masalah yang serius dan harus ditangani agar terhindar dari hilangnya data atau bahkan kegagalan sistem secara total.