

DETEKSI DAN KOREKSI KESALAHAN

1. DETEKSI KESALAHAN

Pengiriman informasi yang menggunakan sinyal digital atau analog selalu mengalami perubahan yang dialami oleh informasi tersebut. Perubahan tersebut bias disebabkan oleh:

- Media pengiriman itu sendiri
- Gangguan terhadap media tersebut
- Sinyal informasi itu sendiri yang melemah karena jarak tempuh
- Peralatan perantara lain yang digunakan dalam pengiriman informasi

Media pengiriman data sangat dipengaruhi oleh gangguan gejala listrik seperti :

- Kilat
- Pengaruh medan listrik motor atau peralatan elektronika lain
- Pengaruh media lain yang membawa sinyal listrik yang berdekatan dengannya

Semua gejala ini disebut derau yang dapat menyebabkan informasi mengalami perubahan atau kesalahan

Oleh karena itu terdapat usaha untuk mencegah, mendeteksi, bahkan memperbaiki kesalahan yang terjadi pada data yang dikirimkan.

Cara mencegah terjadinya kesalahan dilakukan dengan memperbaiki peralatan pengiriman dan penerimaan, serta media pengiriman datanya. Selain itu, system yang dirancang harus dapat melacak kesalahan dan memperbaikinya.

Salah satu deteksi kesalahan dalam komunikasi data adalah: menggunakan tambahan informasi yang tidak ada kaitannya dengan isi informasi yang dikirimkan. Data tambahan inilah yang menunjukkan ada atau tidaknya kesalahan pada data yang dikirimkan tadi.

Data tambahan ini disebut dengan pariti: yaitu penambahan 1 atau beberapa bit "non-information carrying bit", sehingga penerima dapat melakukan perhitungan matematis untuk memeriksa kevalidan data yang diterimanya.

2. Cara Deteksi Kesalahan

Ada beberapa metode untuk mengetahui adanya suatu kesalahan

- a. Metode Eho
- b. Metode deteksi error otomatis
- c. Framming check

a. Metode Echo

Metode yang paling sederhana dan digunakan secara interaktif. Operator memasukkan data melalui terminal yang kemudian mengirimkannya ke computer. Komputer kemudian mengirimkannya kembali ke terminal dan ditampilkan ke monitor. Operator dapat melihat apakah data yang dikirimkannya benar.

b. Metode Deteksi error otomatis

Sistem computer lebih menghendaki sedikit mungkin melibatkan manusia. Oleh karena itu digunakan system bit pariti, yaitu bit tambahan yang digunakan untuk mendeteksi kesalahan. Terdapat 2 macam cara penambahan bit pariti:

Pariti ganjil(Odd parity)

Bit pariti tambahan, supaya banyaknya bit "1" tiap karakter/data ganjil

Pariti genap (even parity)

Bit pariti tambahan, supaya banyaknya bit "1" tiap karakter/data genap

Ada 3 macam teknik deteksi kesalahan dengan menggunakan bit pariti:

- Vertical Redudancy Check(VRC)
- Longitudinal Redudancy Check(LRC)
- Cyclic Redudancy Check (CRC)
- **Character Parity (Vertical Redudancy Check/VRC)**
Merupakan metode pemeriksaan kesalahan per-karakter dan digunakan pada system yang berorientasi karakter, misalnya terminal.
Dengan cara ini, setiap karakter yang dikirimkan (terdiri 7 bit) diberi tambahan 1 bit pariti yang akan diperiksa oleh penerima untuk mengetahui kebenaran karakter yang diterima tersebut.

Cara ini hanya dapat melacak kesalahan 1 bit dan hanya digunakan untuk melacak kesalahan yang terjadi pada pengiriman data berkecepatan menengah.

Misalnya ASCII huruf A, kodenya adalah hex 41:

1000001 ASCII 7 bit (terdapat 2 bit 1)
 1000001 1 tambahkan 1, jumlah bit 1 jadi ganjil(odd parity)
 1000001 0 tambahkan 0, jumlah bit 1 jadi genap(even parity)

Contoh “Even parity”

Pengirim:

Data : 1 1 0 0 0 0 1
 b1 b2 b3 b4 b5 b6 b7

Ada 3 bit “1” (ganjil), tambahkan bit 1, jumlah bit “1” jadi genap

Kirim : data dan parity = 11000011

Penerima:

Proses (algoritma) even parity:

Hitung jumlah bit 1 == > x

Jika x = genap disimpulkan tidak ada error

Jika x = ganjil, terjadi error

Terima : data dan Parity = 11100011

Error?



Gambar : Vertical Redudancy Check dengan Even Parity

Penerima memeriksa pariti dari karakter yang diterima, bila tidak sesuai dengan ketentuan maka akan diketahui adanya kesalahan pada waktu penyaluran data.

VRC mempunyai kekurangan, yaitu bila ada 2 bit yang terganggu maka tidak dapat terlacak, sehingga dianggap apritinya akan benar.

- **LRC**

Untuk memperbaiki kinerja VRC, digunakan LRC untuk data yang dikirim secara blok. Cara ini mirip dengan VRC, hanya saja penambahan bit dilakukan pada akhir setiap blok karakter yang dikirimkan.

Dengan cara ini kesalahan lebih dari 1 bit juga dapat ditemukan, sehingga kecepatan pengiriman data dapat dipertinggi



Gambar Longitudinal Redundancy Check dengan Even Parity

- **CRC**

CRC digunakan untuk pengiriman data berkecepatan tinggi. CRC disebut sebagai pengujian berorientasi bit, karena dasar pemeriksaan kemungkinan kesalahan adalah bit atau karakter dan menggunakan rumus matematika khusus.

Dalam metode ini 1 blok dilihat sebagai deretan bit yang ditransmisikan. Bit yang akan disalurkan dimasukkan ke dalam register geser siklis yang disebut generator CRC. Operasi matematik dikerjakan atas deretan bit tersebut.

Operasi CRC ini didasarkan atas pembagian deretan bit dengan sebuah fungsi khusus. Hasil bagi pembagian diabaikan. Sisa disalurkan sebagai Block Check Sequence (BCS) yaitu akhir dari deretan bit isi register geser.

Berdasarkan pemeriksaan perbandingan hasil perhitungan rumus matematika dari pada saat dikirim dan setelah diterima akan dapat ditentukan adanya kesalahan atau tidak. Pada penerima, deretan bit data termasuk BCS juga dimasukkan ke dalam register geser siklis yang disebut penguji CRC. Hasil operasi matematik ini berupa isi register yang dapat diperkirakan ada tidaknya kesalahan transmisi.

c. Framming check

Digunakan pada transmisi asinkron dengan adanya bit awal dan bit akhir. Dengan memeriksa ke-2 bit ini dapat diketahui apakah data diterima dengan baik.

Transmisi sinkron mempunyai berbagai bentuk bingkai sesuai dengan ketentuan yang digunakan.