

CRC (Cyclic Redundancy Check)

# Pendahuluan

- Metode ini merupakan metode lain untuk menghitung checksum, dengan panjang pesan berkisar 8, 16, atau 32 bit.
- Merupakan kode pemeriksaan error yang umum digunakan pada sistem komunikasi data dan sistem transmisi data serial lainnya.
- Metode ini berdasarkan manipulasi polinomial menggunakan aritmatika modulo. Standar CRC yang umum antara lain CRC-8, CRC-16, CRC-32, dan CRC-CCIT

# Metode CRC

- Data diperlakukan sebagai bilangan biner (0 & 1)
- Bilangan ini dibagi dengan bilangan biner lainnya yang disebut polinomial.
- Hasil sisa pembagian ini merupakan checksum CRC, yang akan ditambahkan pada pesan yang akan ditransmisikan.

# Metode CRC (lanjutan)

- Receiver akan membagi pesan (termasuk CRC yang dihitung) dengan polinomial yang sama dengan polinomial yang digunakan oleh transmitter.
- Jika sisa pembagian yang dilakukan oleh receiver ini sama dengan sisa pembagian yang dilakukan oleh transmitter, maka transmisi dapat dikatakan berhasil.

# Perhitungan untuk mendapatkan CRC

## Contoh

*Pesan* = 110101 (yang akan ditransmisikan)

*Polinomial* = 101 (divisor)

- Pesan akan ditambahkan dengan bit nol sebanyak lebar bit polinomial. Dalam hal ini, lebar bit polinomial adalah 2, maka pesan akan ditambahkan dengan 00 menjadi 11010100, dan akan dibagi dengan polinomial.
- Pembagian ini sama saja dengan men-XOR-kan semua bit yang dibagi dengan bit pembagi.

# Perhitungan P(x)

11010100 ÷ 101

$$\begin{array}{r} 11010100 \\ 101 \\ \hline 111 \\ 101 \\ \hline 100 \\ 101 \\ \hline 110 \\ 101 \\ \hline 110 \\ 101 \\ \hline \underline{11} \end{array}$$

SISA = CRC checksum

Jadi, didapat pesan dengan CRC (P(x)) = 1 1 0 1 0 1 1 1

# Memeriksa pesan pada Receiver

Hal ini dapat dilakukan dengan dua cara, yaitu :

- **Cara I :**

- Pertama receiver akan memisahkan pesan dan checksum, kemudian akan menghitung checksum untuk pesan (setelah menambahkan bit nol sebanyak lebar bit polinomial).
- Lalu receiver akan membandingkan kedua checksum tersebut (yang diterima dan yang dihitung).
- Jika kedua checksum tersebut sama besar, maka tidak terjadi error selama transmisi

# Memeriksa pesan pada Receiver

*Pesan* = 11010111 (Pesan diterima)

Checksum pesan yang diterima = 11

*Polinomial* = 101

11010100 ÷ 101

```
11010100
101
-----
111
101
-----
100
101
-----
110
101
-----
110
101
-----
11
```

→ SISA = CRC checksum



# Memeriksa pesan pada Receiver

- **Cara II :**
  - receiver akan menghitung checksum untuk keseluruhan pesan.
  - kemudian memeriksa apakah hasilnya sama dengan nol (berarti tidak terjadi error selama transmisi).

# Memeriksa pesan pada Receiver

*Pesan* = 11010111

*Polinomial* = 101

11010111 ÷ 101

$$\begin{array}{r} 11010111 \\ 101 \\ \hline 111 \\ 101 \\ \hline 100 \\ 101 \\ \hline 111 \\ 101 \\ \hline 101 \\ 101 \\ \hline \underline{00} \end{array}$$

Checksum = 0  
Tidak ada error transmisi

- Pesan : 100110001101
- Divisor : 11001