REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan	:	EC00201983168, 20 November 2019
Pencipta		
Nama	:	**Dr. Ratnadewi, S.T., M.T., Novie Theresia Br. Pasaribu, S.T., M.T.,**
Alamat	:	Gg.Silih Asih I No. 20, Bandung, Jawa Barat, 40253
Kewarganegaraan	:	Indonesia
Pemegang Hak Cipta		
Nama	:	**Universitas Kristen Maranatha**
Alamat	:	Jl. Prof.drg. Suria Sumantri No. 65, Bandung, Jawa Barat, 40164
Kewarganegaraan	:	Indonesia
Jenis Ciptaan	:	**Karya Tulis**
Judul Ciptaan	:	**Teori Informasi**
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	:	10 November 2019, di Bandung
Jangka waktu pelindungan	:	Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	:	000165432

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL

Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Dr. Ratnadewi, S.T., M.T.	Gg.Silih Asih I No. 20
2	Novie Theresia Br. Pasaribu, S.T., M.T.	Taman Cibaduyut Indah E-167

TEORI INFORMASI

Ratnadewi dan Novie Theresia Br. Pasaribu

2019

TEORI INFORMASI

Disusun oleh:

Ratnadewi
Novie Theresia Br. Pasaribu

Jurusan Teknik Elektro
Fakultas Teknik
Universitas Kristen Maranatha

2019

KATA PENGANTAR

Diktat yang berjudul "Teori Informasi", ini merupakan diktat kami yang menyajikan tentang informasi pada suatu sistem pengiriman dan penerimaan data digital.

Teori Informasi merupakan ilmu yang banyak dibutuhkan di bidang teknik komunikasi digital di Perguruan Tinggi. Kerahasian data pada proses pengiriman menjadi hal penting yang menyebabkan diperlukan suatu metode mengkodekan suatu data, sehingga data yang dikirim tidak dimengerti oleh orang lain yang tidak berkepentingan, selain penerima data. Tidak dapat dihindari pada proses pengiriman data seringkali data terkena noise sehingga data yang diterima oleh penerima menjadi berubah, tidak sama dengan data yang dikirimkan, oleh sebab itu diperlukan suatu teknik memperbaiki data yang diterima. Pada diktat ini dibahas beberapa metode untuk memperbaiki data yang diterima.

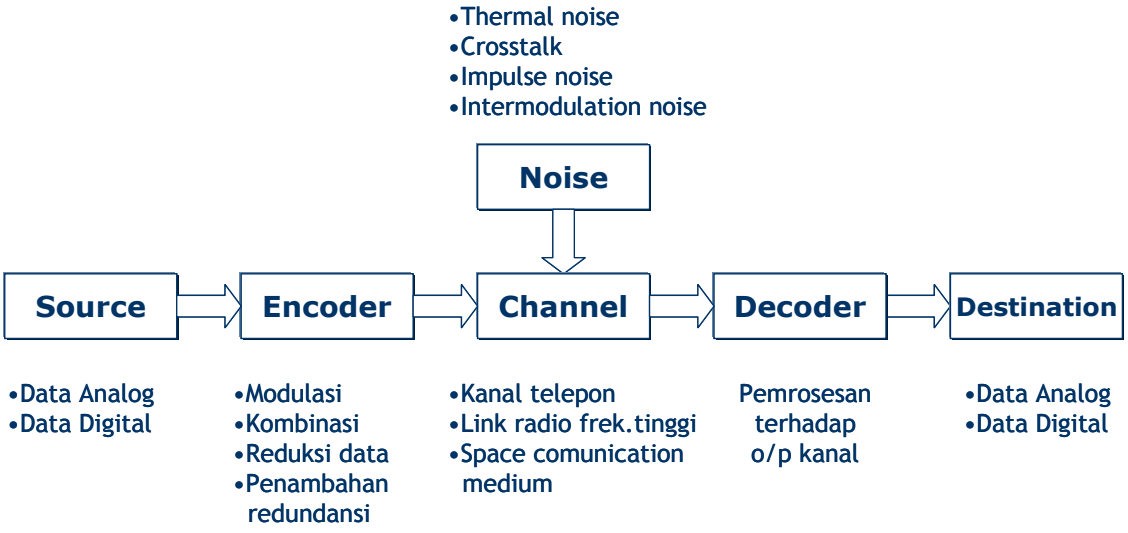
Seringkali banyak mahasiswa merasa kesulitan dalam mempelajarinya. Oleh karena itu, pada diktat ini diberikan topik-topik dari teori informasi dengan cara penyajian yang diharapkan mudah dipahami oleh para pembacanya.

Buku ini menyajikan teori-teori secara singkat yang berhubungan dengan: konsep ketidakpastian, informasi dan entropi, kanal transmisi, Huffman algorithm, teknik pendeteksi error, elemen aljabar untuk pengkodean, linier binary block code, cyclic code, kode BCH, dan kode convolutional.

DAFTAR ISI

I. PENDAHULUAN

BLOK DIAGRAM SISTEM KOMUNIKASI



SHANNON(MACKAY, 2003)

- 1940-an, mengembangkan teori matematika, disebut :
  TEORI INFORMASI → aspek dasar sistem komunikasi

- 2 Karakteristik dasar teori informasi :
  - Memberikan perhatian besar pada teori PROBABILITAS
  - Fokus utama pada : ENKODER & DEKODER

Teori dan praktek error-correction coding adalah mengenai perlindungan informasi digital terhadap error yang terjadi selama transmisi data atau penyimpanan.

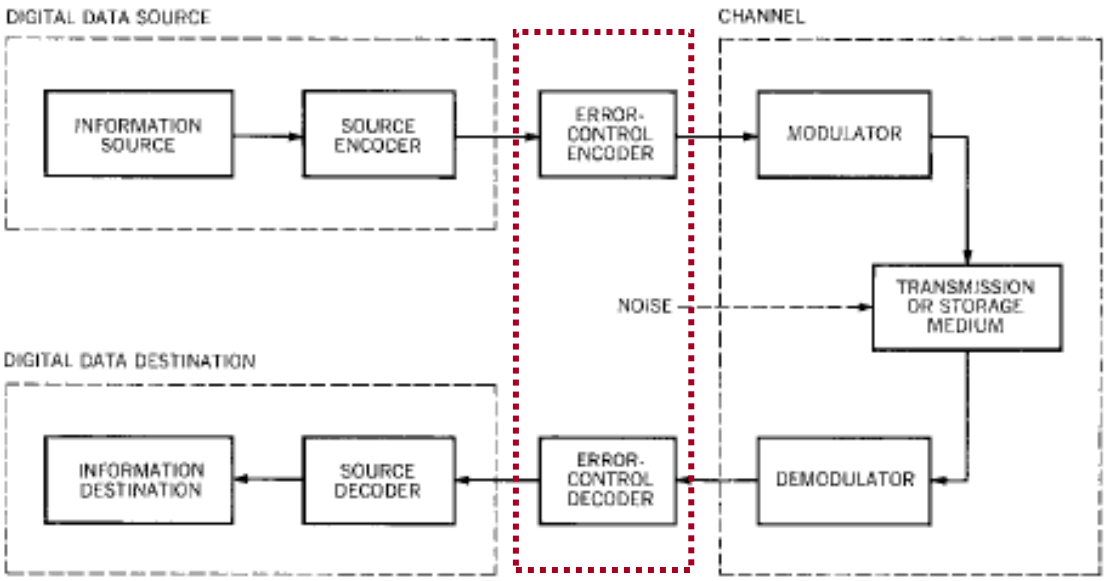
PERKEMBANGAN SETELAH SHANNON

Metoda praktis yang telah diciptakan dan diimplementasikan :

- Source coding:
  - Huffman codes (compact)
  - Lempel-Ziv (compress, gzip)

- Channel coding:
  - Error-correcting codes
  - Hamming
  - Reed-Solomon
  - Convolutional
  - Trellis
  - turbo

SISTEM KOMUNIKASI DIGITAL



II. KONSEP KETIDAKPASTIAN, INFORMASI & ENTROPI

KETIDAKPASTIAN(MACKAY, 2003)

Defenisi Ketidakpastian :

Kondisi kurangnya informasi yang diperlukan untuk mengambil suatu keputusan.

→ keputusan yang dihasilkan bukan merupakan yang terbaik, & mungkin merupakan keputusan yang buruk.

Teori mengatasi Ketidakpastian :

- Classical probability
- Bayesian probability
- Hartley teory berdasarkan pada classical sets
- Shannon theory berdasarkan probability
- Dempster-Shafer theory

PROBABILITAS

Probabilitas adalah peluang suatu kejadian

Manfaat Probabilitas :

Membantu pengambilan keputusan yang tepat, karena kehidupan di dunia tidak ada kepastian, dan informasi yang tidak sempurna.

Probablitas → mekanisme penanggulangan ketidakpastian secara kuantitatif.

Formula probabilitas klasik → **a priori probability :**

- Probabilitas yang ada dapat diperhitungkan sebelumnya
- Asumsi : semua kejadian yang mungkin diketahui dan setiap kejadian tersebut memiliki peluang yang sama untuk terjadi

PENDEKATAN KLASIK

Definisi: Setiap peristiwa mempunyai kesempatan yang sama untuk terjadi.

Rumus:

$$probabilitas\ suatu\ peristiwa = \frac{jumlah\ kemungkinan\ hasil}{jumlah\ total\ kemungkinan\ hasil}$$

Contoh :

Sebuah dadu memiliki 6 permukaan yang tertulis angka 1,2,3,4,5,6 . Jika dadu dilemparkan sekali , maka terdapat 6 kejadian yang mungkin dari munculnya 1 angka. Maka probabilitas munculnya :

- angka 1 : P(1)=1/6
- angka 2 : P(2)=1/6
- angka 3 : P(3)=1/6
- angka 4 : P(4)=1/6
- angka 5 : P(5)=1/6
- angka 6 : P(6)=1/6

TEORI PROBABILITAS

1. Aksioma 1 : $0 \le P(E) \le 1$

   Range probabilitas yang bernilai dari 0 (kejadian yang tidak mungkin terjadi) hingga 1 (kejadian yang pasti).

- Aksioma 2 : $\sum_i 0 \le P(E_i) = 1$

  Jumlah dari semua kejadian yang tidak saling mempengaruhi satu dengan lainnya (**mutualy exclusive**) adalah 1 → $P(E) + P(E') = 1$

- Aksioma 3 : $P(E_1 \cup E_2) = P(E_1) + P(E_2)$

  Jika E1 dan E2 tidak dapat terjadi secara simultan/ bersamaan (**mutually exclusive**) → probabilitas satu atau lainnya terjadi adalah penjumlahan dari probabilitas semuanya.

CONDITIONAL PROBABILITY

Kejadian yang tidak **mutually exclusive** (sebuah kejadian yang tidak memiliki sample point yang sama) mempengaruhi satu dengan lainnya.

Mengetahui bahwa satu kejadian telah terjadi mengakibatkan kita harus memperbaiki probabilitas terjadinya kejadian yang lain.

Multiplicative Law : Probabilitas kejadian A jika kejadian B telah terjadi

→ Probabilitas conditional

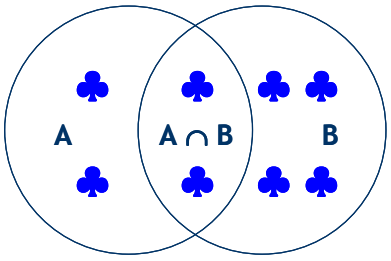
$$P(A \mid B)$$

Baca :

Probabilitas kejadian A, jika kejadian B telah terjadi

Contoh : Ruang sampel 2 kejadian yang beririsan

Jawab :



$$P(A) = \frac{n(A)}{n(S)} = \frac{4}{8} \qquad P(B) = \frac{n(B)}{n(S)} = \frac{6}{8} \qquad P(A \mid B) = \frac{n(A \cap B)}{n(B)} = \frac{2}{6}$$

- Kondisional probabilitas dapat diekspresikan dengan :

$$P(A \mid B) = \frac{\frac{n(A \cap B)}{n(S)}}{\frac{n(B)}{n(S)}} = \frac{P(A \cap B)}{P(B)} \quad \text{untuk } P(B) \neq 0$$

- Multiplicative Law :
  - 2 Kejadian

    $P(A \cap B) = P(A|B)\,P(B)$ ekuivalen dengan $P(A \cap B) = P(B|A)\,P(A)$
  - 3 Kejadian

    $P(A \cap B \cap C) = P(A|B \cap C)\,P(B|C)\,P(C)$
  - N Kejadian

$$P(A_1 \cap A_2 \cap \ldots \cap A_N) = P(A_1 | A_2 \cap \ldots \cap A_N) \cdot P(A_2 | A_3 \cap \ldots \cap A_N) \cdot \ldots \cdot P(A_{N-1} | A_N)\,P(A_N)$$

TEOREMA BAYES

Conditional probability P(A|B) menyatakan probabilitas kejadian A jika kejadian B telah terjadi. Masalah lain adalah mencari inverse probability.

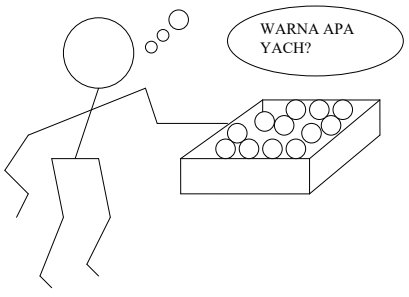
$$P(H_i|E) = \frac{P(E \cap H_i)}{\sum_j P(E \cap H_j)}$$

$$= \frac{P(E|Hi)\,P(Hi)}{\sum_j P(E|H_j)\,P(Hj)}$$

$$= \frac{P(E|Hi)\,P(Hi)}{P(E)}$$

Kasus ini disebut **posteriori probability**.

Contoh Kasus:



Suatu kotak berisi :

- 1 bola putih, 1 bola hitam.
- 1 bola putih, 9 bola hitam.
- 1000 bola putih, 999000 bola hitam.

Ketidakpastian
→ Terbesar
→ Lebih kecil
→ Paling kecil

Pada kasus (c), Jika kotak berisi :

- (c') 1001 bola putih, 999999 bola hitam
- (c'') 1002 bola putih, 999998 bola hitam
- Ketidakpastian (c') dan (c'') ≈ (c)

Pada kasus (b), Jika kotak berisi :

- (b') 9 bola putih, 1 bola hitam
- Ketidakpastian (b') terhadap (b) ?

KONSEP INFORMASI

Kasus a), jika 1 bola diambil dari kotak, akan membawa banyak informasi. Kasus b) dan c), jika bola yang ' diambil ' adalah hitam, maka informasi yang kita dapatkan adalah sedikit; tetapi jika bola yang ' diambil ' adalah putih, ini mempunyai informasi yang banyak.

Ketidakpastian $\rightarrow$ Konsep Apriori

Informasi $\rightarrow$ Konsep Aposteriori

Contoh:

- KONSEP APRIORI
  - P [putih] = 2/100 = 0.02
  - P [hitam] = 98/100 = 0.98

- KONSEP APOSTERIORI
  - Terambil bola hitam
    - P [putih] = 2/99 = 0.0202 $\Delta = 0.0002$
    - P [hitam] = 97/99 = 0.9797 $\Delta = 0.0003$
  - Terambil bola putih
    - P [putih] = 1/99 = 0.0101 $\Delta = 0.0099$
    - P [hitam] = 98/99 = 0.9898 $\Delta = 0.0098$

- Penarikan Bola Putih Mengurangi Tingkat Ketidakpastian Dari Sisa Lebih Besar Dari Penarikan Bola Hitam
- Putih memuat lebih banyak informasi dibandingkan hitam

SUMBER INFORMASI

Sumber Informasi (*Information Source*) : sebuah pasangan teratur S=(S,P)

- S = {S1, S2, …, Sn} , S: alphabet sumber (*source alphabet*)
- P : suatu aturan peluang yang mengkaitkan elemen Si dari S dengan sebuah peluang P(Si)
- Deretan P(S1), P(S2),…, P(Sq) → Distribusi peluang (*Probability distribution*) untuk S

Suatu sumber memancarkan/ mengirim simbol sumber (*source symbols*), membentuk → message

PENCUPLIKASI SUATU INFORMASI

Jika suatu sumber informasi **S**=(S,P) dicuplik, maka: Peluang dari elemen Si diambil adalah P(Si).

- Sebelum pencuplikan terjadi → ada suatu ketidakpastian (uncertainty) tentang keluaran pencuplikan (simbol apa yang akan muncul)
- Setelah ada keluaran (simbol) → didapat sejumlah informasi mengenai sumber informasi tersebut.

INFORMASI I(p)

INFORMASI yang diperoleh dari sebuah simbol sumber, bukan fungsi itu sendiri, tetapi merupakan fungsi peluang terjadinya suatu simbol : P(Si)=pi

Jika I(p) → informasi yang didapat dari sebuah simbol sumber s dengan peluang terjadinya p.

Dengan I(p) yang terdefenisi untuk $0<p\leq1$, maka dibuat asumsi sebagai berikut :

- $I(p)\geq0$
- Fungsi I(p) adalah fungsi kontinu dengan variabel p
- Event Si dan Sj terjadi independent I(pipj)=I(pi)+I(pj)

Fungsi yang memenuhi → **I(p)= C log 1/p**

- I(p) : Self information (satuan : bits)
- C : konstanta positif
- Log : logaritma base 2

KONSEP ENTROPI

ENTROPI suatu sumber → suatu ukuran jumlah informasi suatu sumber.

Jika :

- **S**=(S,P) → suatu sumber informasi
- P={p1,p2,…,pq} → disribusi peluang

Maka average information didapat dari sebuah sampel/simbol dari S adalah :

$$H(S) = \Sigma\ pi\ I(pi) = \Sigma\ pi\ ^2\log\ 1/pi = -\ \Sigma\ pi\ ^2\log(pi)\ \text{bits/simbol}$$

- JOINT ENTROPY : Diberikan 2-dimensi ruang sampel (X,Y) dengan joint probability distribution R={rij}. Joint entropy antara X & Y :

$$H(X,Y) = -\sum_{i=1}^{n}\sum_{j=1}^{m} r_{ij} \log r_{ij}$$

- CONDITIONAL ENTROPY : Kondisional entropi Y karena X

$$H(Y \mid X) = -\sum_{i=1}^{n}\sum_{j=1}^{m} r_{ij} \log q(y_j \mid x_i)$$

<u>dibaca:</u> q_{ji} adalah conditional probability dari y_j yang diberikan x_i

<u>Sifat-sifat Entropi :</u>

- H(X) maksimal, jika seluruh event probabilitasnya sama $p(x_i) = k$
- Entropi r.v (X,Y) adalah sama dengan jumlah entropi X dan Y|X

  H(X,Y) = H(X) + H(Y|X)

  = H(Y) + H(X|Y)
- H(X,Y) ≤ H(X) + H(Y)
- H(X,Y) = H(X) + H(Y) jika dan hanya jika *random variable* X dan Y saling bebas (tidak ada hubungan)

III. KANAL TRANSMISI

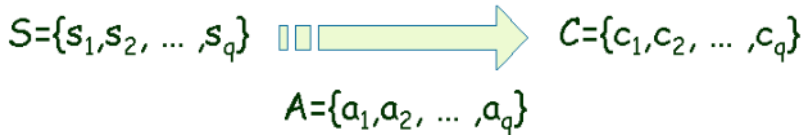
SUMBER INFORMASI(Wicker, 1995)

Suatu sumber informasi **S**=(S) memancarkan/ mengirimkan simbol sumber (source symbols) untuk membentuk sebuah message.

Diandaikan bahwa pengiriman simbol sumber tidak tergantung waktu

- $S = \{S_1, S_2, \ldots, S_q\}$, alphabet sumber, elemen s_i disebut symbol/ letter
- $A = \{a_1, a_2, \ldots, a_q\}$, alphabet kode, elemen a_i disebut simbol kode
- $C = \{c_1, c_2, \ldots, c_q\}$, kode dengan memakai alphabet A

SKEMA PENGKODEAN



Bila C adalah suatu kode, dan fungsi pengkodean (*encoding function*) *f*:S→C, maka pasangan teratur (C,*f*) disebut skema pengkodeaan (*enchoding scheme*) untuk S

Untuk mengukur efesiensi skema pengkodean → average codeword length (acl)

$$acl = \sum_{i=1}^{q} len(f(s_i))P(s_i)$$

Contoh :

1. Bila aphabet sumber : S={a,b,c,…z}, dan alphabet kode : A={0,1,2,…,9}, serta kode: C={00,01,…,25}

maka :

f:S→C dapat didefinisikan sebagai :

f(a)=00, *f*(b)=01, *f*(c)=02, *f*(d)=03, *f*(e)=04, *f*(f)=05, *f*(g)=06, *f*(h)=07, *f*(i)=08, *f*(j)=09, *f*(k)=10, *f*(l)=11, *f*(m)=12,*f*(n)=13, *f*(o)=14, *f*(p)=15, *f*(q)=16, *f*(r)=17, *f*(s)=18, *f*(t)=19, *f*(u)=20, *f*(v)=21, *f*(w)=22, *f*(x)=23, *f*(y)=24, *f*(z)=25

sebagai ilustrasi : Berita "saya baca" → 1800240001000200

2. Bila alphabet sumber : S={a,b,c,d}, dengan peluang P(a)=2/17, P(b)=2/17, P(c)=8/17, P(d)=5/17.

Ada 2 set skema encoding sebagai berikut :

a →Ca=11	a →Ca=01010
b →Cb=0	b →Cb=00
c →Cc=100	c →Cc=10
d →Cd=1010	d →Cd=11

maka panjang codeword rata-rata :

skema 1 : acl=2.2/17+1.2/17+3.8/17+4.5/17=50/17

skema 2 : acl=5.2/17+2.2/17+2.8/17+2.5/17=40/17

panjang codeword rata-rata skema 2 lebih pendek dari skema 1 → panjang codeword rata-rata skema 2 lebih efisien.

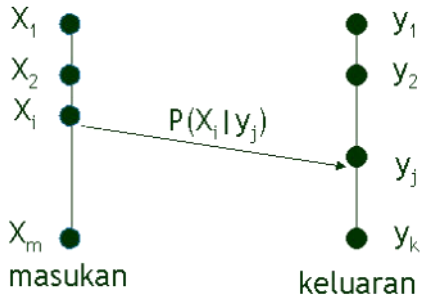
KANAL TRANSMISI

Sebuah kanal komunikasi diskrit dapat dinyatakan dalam besaran-besaran :

- Himpunan alphabet masukan : $X=\{x_1,x_2, \dots ,x_m\}$
- Himpunan alphabet keluaran : $Y=\{y_1,y_2, \dots ,y_k\}$
- Untuk setiap alphabet masukan, peluang terjadinya alphabet keluaran bersyarat terhadap alphabet masukan tersebut : **transition probability p(yk|xm)**

KANAL DISKRIT TANPA MEMORI (DISCRETE MEMORYLESS CHANNEL)

- DISKRIT : mengacu pada kenyataan bahwa alphabet masukan dan keluaran terbatas (finite)
- MEMORYLESS : menunjukkan probabilitas xi diterima hanya bergantung pada masukan sekarang yj , dan tidak bergantung pada masukan sebelumnya



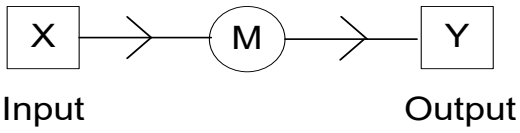
DISCRETE MEMORYLES CHANNEL :

Adalah sebuah kanal dimana masukan dan keluarannya masing-masing adalah deretan simbol-simbol yang berasal dari alphabet terbatas. Simbol keluaran pada waktu tertentu secara statistik hanya tergantung pada symbol masukan pada waktu yang sama.

Sebuah kanal diskrit tanpa memory memiliki tiga elemen:

1.) Input ke kanal (X): X = { x_1 , x_2 , …. , x_m }
2.) Output dari kanal (Y): Y = { y_1 , y_2 , …. , x_k }
3.) Matriks transisi [M] = m_{ij} didefinisikan $m_{ij} = P_{j|i}$

Gambaran kanal diskrit tanpa memory :



Sebuah kanal binary mempunyai karakteristik :

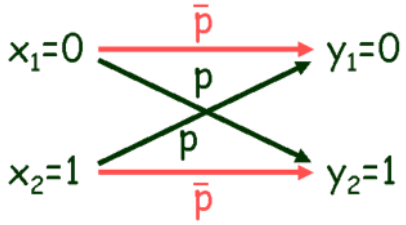
X = Y , $x_1 = y_1 = 0$, $x_2 = y_2 = 1$

NOISELESS BINARY CHANNEL

- Masukan ke kanal (X) : X={0,1}
- Keluaran dari kanal (Y) : Y={0,1}

BINARY SYMMETRIC CHANNEL (BSC)

- Masukan ke kanal (X) : X={0,1}
- Keluaran dari kanal (Y) : Y={0,1}



$$\left.\begin{array}{l} P[\,Y=0 \mid X=1\,] = P[\,0|1\,] \\ P[\,Y=1 \mid X=0\,] = P[\,1|0\,] \end{array}\right\} \mathbf{p}$$

$$\left.\begin{array}{l} P[\,Y=0 \mid X=1\,] = P[\,0|0\,] \\ P[\,Y=1 \mid X=0\,] = P[\,1|1\,] \end{array}\right\} \mathbf{1-p=\bar{p}}$$

Matriks Transisi :

$$[\,M\,] = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix} \approx \begin{matrix} \\ x_1 \\ x_2 \end{matrix} \begin{matrix} \begin{matrix} y_1 & y_2 \end{matrix} \\ \begin{pmatrix} .... & .... \\ .... & .... \end{pmatrix} \end{matrix}$$

Contoh :

Sebuah kanal binary dengan input yang mempunyai kemungkinan :

$$p_1 = \frac{1}{4} \qquad p_2 = \frac{3}{4}$$

Matriks transisi : $[\,M\,] = \begin{matrix} \\ x_1 \\ x_2 \end{matrix} \begin{matrix} \begin{matrix} y_1 & y_2 \end{matrix} \\ \begin{pmatrix} 2/3 & 1/3 \\ 2/5 & 3/5 \end{pmatrix} \end{matrix}$

Sehingga diperoleh Y :

$$q_1 = p_1 \,.\, P\{ Y = y_1 \mid X = x_1 \} + p_2 \,.\, P\{ Y = y_1 \mid X = x_2 \} = \frac{7}{15}$$

$$q_2 = p_1 \,.\, P\{ Y = y_2 \mid X = x_1 \} + p_2 \,.\, P\{ Y = y_2 \mid X = x_2 \} = \frac{8}{15}$$

Distribusi (X ,Y) adalah :

$$p_{11} = p_1 \cdot p_{1|1} = \frac{1}{6} \qquad p_{12} = p_1 \cdot p_{2|1} = \frac{1}{12}$$

$$p_{21} = p_2 \cdot p_{1|2} = \frac{3}{10} \qquad p_{22} = p_2 \cdot p_{2|2} = \frac{9}{20}$$

Entropi dari X , Y , (X,Y) dan (X|Y):

$$H(X) = -\frac{1}{4}\log\frac{1}{4} - \frac{3}{4}\log\frac{3}{4} = 0.811 \text{ bits}$$

$$H(Y) = -\frac{7}{15}\log\frac{7}{15} - \frac{8}{15}\log\frac{8}{15} = 0.997 \text{ bits}$$

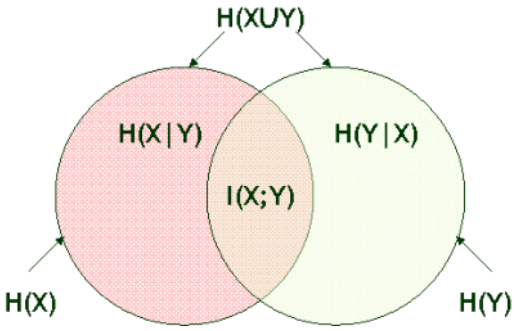
$$H(X,Y) = -\frac{1}{6}\log\frac{1}{6} - \frac{1}{12}\log\frac{1}{12} - \frac{3}{10}\log\frac{3}{10} - \frac{9}{20}\log\frac{9}{20} = 1.769 \text{ bits}$$

$$H(Y|X = x_1) = -\frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3} = 0.918 \text{ bits}$$

$$H(Y|X = x_2) = -\frac{2}{5}\log\frac{2}{5} - \frac{3}{5}\log\frac{3}{5} = 0.971 \text{ bits}$$

$$H(Y|X) = \frac{1}{4} \text{ x } 0.918 + \frac{3}{4} \text{ x } 0.971 = 0.958 \text{ bits}$$

MUTUAL INFORMATION { Dari Kanal }



- $I(X;Y) = H(X) - H(X|Y)$
  $= H(Y) - H(Y|X)$
- $H(Y)$: Apriori Entropy
- $H(Y|X)$: Aposteriori Entropy

$\therefore$ Selisih = Informasi aliran dari $X \rightarrow Y$

KAPASITAS KANAL

$$C = \max_{p(x_i)} I(X;Y)$$

{ Satu kanal tertentu, memiliki probabilitas awal yang tetap }

Dapat diduga bahwa C juga fungsi dari probabilitas awal dari kanal

Kanal berbeda memiliki kapasitas berbeda

Contoh :

Probabilitas random variable Y adalah :

$$q_1 = p_1\, q + (1 - p_1)\, p$$

$$q_2 = p_1\, p + (1 - p_1)\, p$$

$$H(Y|X) = p_1 \times H(Y|X = x_1)\ (1 - p_1) \times H(Y|X = x_2)$$
$$= p_1\,[-(1-p)\log(1-p) - p\log p] + (1 - p_1)[-p\log p - (1-p)\log(1-p)$$
$$= -p\log p - q\log q$$

$$H(Y) = -[p_1 q + (1 - p_1) p] \log [p_1 q + (1 - p_1) p]$$

$$= -[p_1 p + (1 - p_1) q] \log [p_1 p + (1 - p_1) q]$$

Dari definisi C adalah nilai maksimal dari [H (Y) – H (Y|X)] sebagai fungsi dari input probabilitas p_i.

Terlihat H (Y|X) tidak tergantung probabilitas input. Sehingga C diperoleh dari maksimum H (Y).

Dari awal didefinisikan entropy maks, jika $p_1 q + (1 - p_1) p$ sama dengan $p_1 p + (1 - p_1) q$ (sama dengan $\frac{1}{2}$)maka ketika $p_i = \frac{1}{2}$ diperoleh :

$$C = 1 + p \log p + q \log q$$
$$= 1 - H (X)$$

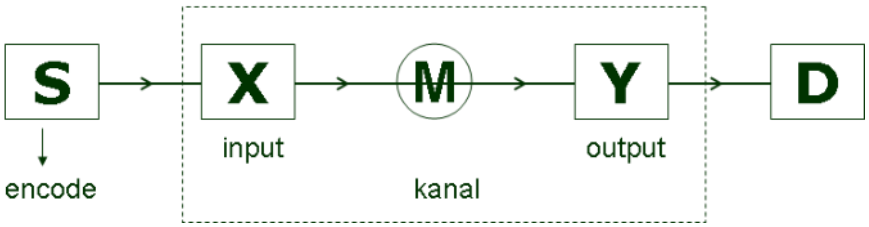
KANAL TRANSMISI TANPA NOISE

Terjadi jika :

$$H (X,Y) = H(X) = H(Y)$$

Dan

$$I (X;Y) = H(X)$$



IV. HUFFMAN ALGORITHM

LANGKAH-LANGKAH HUFFMAN ALGORITHM :(Moon, 2005)

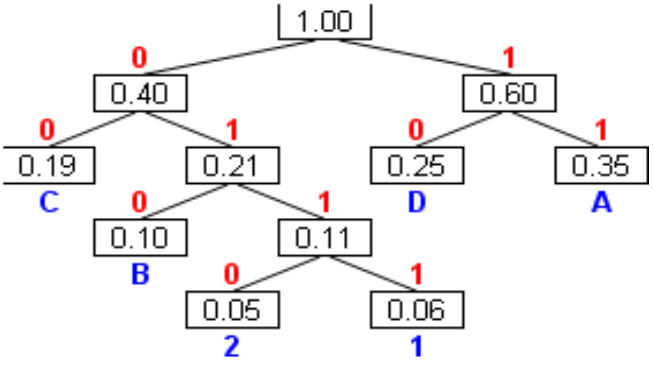
1.) Simpan simbol sumber dalam daftar dengan probabilitas menurun
2.) Gabungkan sejumlah maks k simbol dengan peluang terendah kedalam satu kelompok { k : Banyak Alfabet Pengkoda }
3.) Ulangi (1) dan (2) sehingga tinggal k simbol
4.) Beri satu alphabet pengkoda kepada setiap simbol terakhir
5.) Lanjutkan ke awal dengan memberi tambahan satu symbol pengkoda setiap kali keluar dari gabungan

Contoh : { Pengkode Binari ; x=(0,1)}

Sumber S_8 Untuk sumber ini k=2

Simbol	Probability	Probability								
A	0.35	0.35	0.35	0.35	0.35	0.35	0.40	0.40	0.60	1.00
D	0.25	0.25	0.25	0.25	0.25	0.25	0.35	0.60	0.40	
C	0.19	0.19	0.19	0.19	0.21	0.40	0.25			
B	0.10	0.10	0.11	0.21	0.19					
1	0.06	0.11	0.10							
2	0.05									

Kita bangun kodenya :



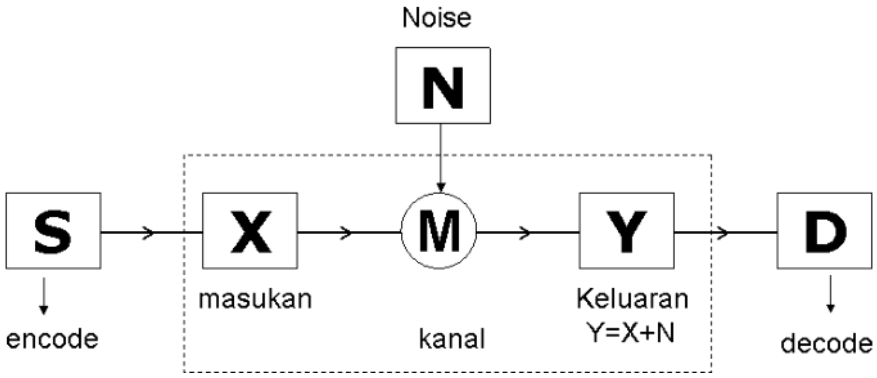
Simbol	Kode Biner
A	11
D	10
C	00
B	010
1	0111
2	0110

Keterangan : 0,02 + 0,02 = 0,04

Acl = 2(0.35)+2(0.25)+2(0.19)+3(0.10)+4(0.06)+4(0.05)=2.32 Bit/simbol

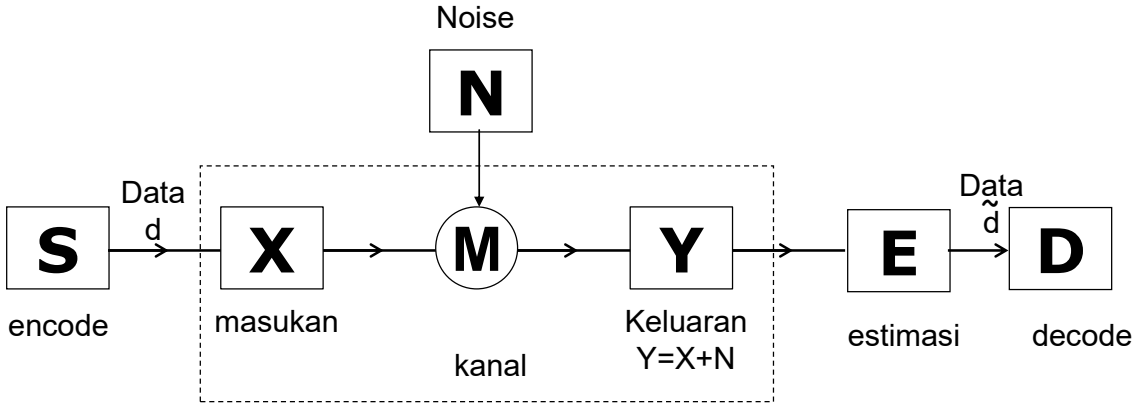
V. TEKNIK PENDETEKSIAN ERROR

KANAL TRANSMISI DENGAN NOISE



- Noise berupa derau Gaussian yang memiliki mean nol dan variansi tertentu ($\sigma 2$)
- Informasi I(X,Y)=H(Y)-H(N)

KESALAHAN TRANSMISI DAN PERBAIKANNYA



Asumsi : data biner d={0,1}

Misal : d = (10110101)

$\tilde{d}$ = (11110001)

---------- +

$d \oplus \tilde{d}$ = (01000100) = e (error pattern)

KESALAHAN TRANSMISI DAN PERBAIKANNYA

Konsep error correcting di kode biner : $\tilde{d} \oplus e = d$

Masalahnya :

- Bagaimana mengenal ada salah bit
- Bagaimana mencari error pattern

Metode :

- Mengulang tiap bit (2n+1) kali
- Tambah bit pariti
- Kode Hamming(n,k)

MENGULANG TIAP BIT (2N+1) KALI

Misal :

- Asal (101)
- Kirim (111000111)
- Terima (111010111)
  $\underbrace{\quad}$ tidak terulang 3x, berarti ada salah

Koreksi dengan mengikuti mayoritas

TAMBAH BIT PARITI

Tambah bit pariti :p

Misal : d = (d_1, d_2, d_3, … d_k)

s = (d_1, d_2, d_3, … d_k, p)

Dengan

$p = \sum_{\oplus} d_i$ untuk pariti genap (s=0)

$p = \sum_{\oplus} d_i + 1$ untuk pariti ganjil (s=1)

contoh :

- d=(1011) → $\sum_{\oplus} d_i = 1$
  - Pariti genap: 1 ⇒ Se=(10111) → $\sum_{\oplus} S_i = 0$
  - Pariti ganjil : 0 ⇒ So=(10110) → $\sum_{\oplus} S_i = 1$

- Misal: e=(01000)
  - Maka: r_e=(11111) → $\sum_{\oplus} r_i = 1$
    Berarti ada salah karena pariti genap seharusnya bernilai 0
  - r_o=(11110) → $\sum_{\oplus} r_i = 0$
    Berarti ada salah karena pariti ganjil seharusnya bernilai 1

BIT PARITI

Jika error satu bit, maka:

- Akan terdeteksi
- Tidak dapat dikoreksi

Jika error dua bit, maka:

- Tidak terdeteksi
- Tidak terkoreksi

Perlu kode yang lebih kuat

KODE HAMMING(n,k)

- k:banyak bit (panjang) data : d
- n:panjang kode : c
- n-k:panjang bit pariti : p

Misal: H(7,4)

- d=(d_1, d_2, d_3, d_4)
- c=(d_1, d_2, d_3, d_4, p_1, p_2, p_3)
- p= (p_1, p_2, p_3)

Jika:

- $p_1 = d1 \oplus d_2 \oplus d_3$
- $p_2 = d_2 \oplus d_3 \oplus d_4$
- $p_3 = d_1 \oplus d_2 \oplus d_4$

Kode ini mampu memperbaiki satu bit error

Bukti dengan menghitung: q_1, q_2, q_3

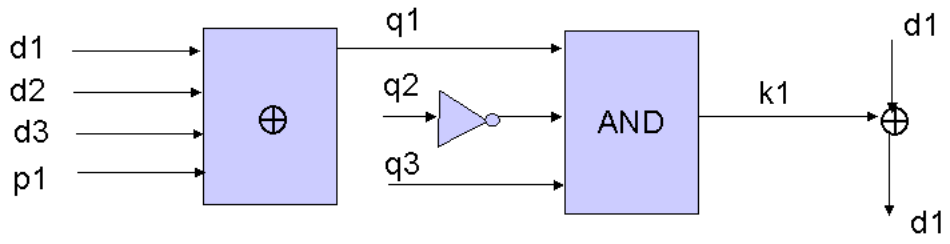
- $q_1 = p_1 \oplus d_1 \oplus d_2 \oplus d_3$
- $q_2 = p_2 \oplus d_2 \oplus d_3 \oplus d_4$
- $q_3 = p_3 \oplus d_1 \oplus d_2 \oplus d_4$

Maka jika:

- Tidak ada salah, maka nilai $q_1 = q_2 = q_3 = 0$
- Ada salah satu bit :

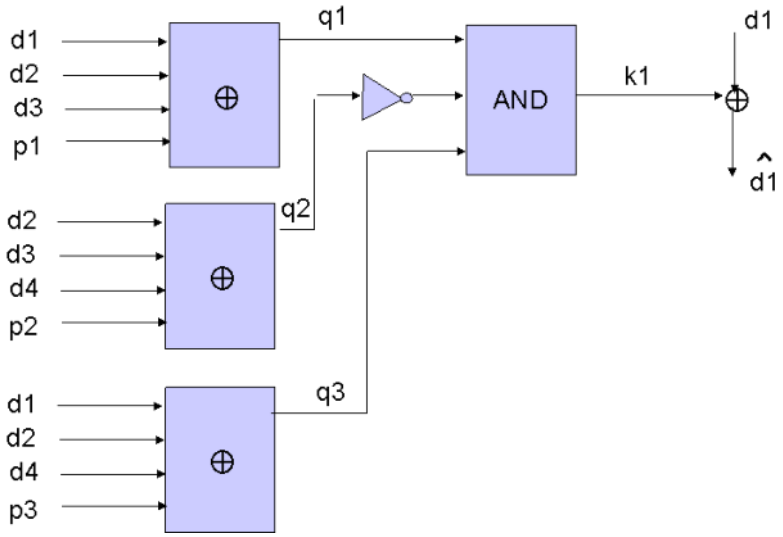
		Bit salah						
		d_1	d_2	d_3	d_4	p_1	p_2	p_3
nilai	q_1	1	1	1	0	1	0	0
	q_2	0	1	1	1	0	1	0
	q_3	1	1	0	1	0	0	1

- Terlihat pola $q=(q_1, q_2, q_3)$ unik untuk setiap posisi bit salah di kode yang diterima
- Kesimpulan : dapat dibuat rangkaian logika untuk mengenali posisi salah bit dan mengkoreksi data yang diterima



- $q=(101) \Rightarrow k1=1 \Rightarrow d1' = d1$
- $q \neq (101) \Rightarrow k1=0 \Rightarrow d1' = d1$

KODE HAMMING(7,4) →data d1



X	Y	XOR X⊕Y
0	0	0
0	1	1
1	0	1
1	1	0

Misal 1 : d=(1010)

p1=d1⊕ d2 ⊕ d3 =0

p2=d2⊕ d3 ⊕ d4 =1

p3=d1⊕ d2 ⊕ d4 =1

- Jika tanpa noise r=(1010011)
  - q1=p1 ⊕ d1⊕ d2 ⊕ d3 =0
  - q2=p2 ⊕ d2⊕ d3 ⊕ d4 =0
  - q3= p3 ⊕ d1⊕ d2 ⊕ d4 =0
  - **q=(000) ⇒ k1=0 ⇒ $\hat{d1}$ = d1**

- Jika ada noise e=(1000000), r=(0010011)
  - q1=p1 ⊕ d1⊕ d2 ⊕ d3 =1
  - q2=p2 ⊕ d2⊕ d3 ⊕ d4 =0
  - q3= p3 ⊕ d1⊕ d2 ⊕ d4 =1
  - **q=(101) ⇒ k1=1 ⇒ $\hat{d1}$ = $\overline{d1}$**

Catatan:

- Di penerima pi tidak dibutuhkan, jadi tidak perlu dikoreksi
- Karena menggunakan rangkaian logika, perbaikan terjadi seketika (sangat cepat)
- Operasi harus per blok kode yang benar (tidak ada salah di bit awal)
- Kode hamming ∈ blok kode

VI. ELEMEN ALJABAR UNTUK PENGKODEAN

(S) set adalah kumpulan object / elemen (Judson & Austin, 2009)(Lint, 1973)(Spence, 2008)

- a : satu dari elemen
  - a $\in$ S : a anggota S
  - a $\notin$ S : a bukan anggota S
- $\varnothing$: set tanpa elemen (set yang kosong)
- B : subset dari set A, jika dan hanya jika setiap elemen dari B ada di dalam A → B $\subset$ A
- A : subsetnya set A dan $\varnothing$
- $*$: operasi binari: -, X, :, +

GROUPS

Suatu set elemen G, berdasarkan operasi [$*$]

Memenuhi :

- Tertutup/closure
  Untuk setiap a, b Є G; $c = a * b$ ada dalam G
- Assosiatif
  Untuk setiap a, b, c Є G; $a*(b*c) = (a*b)*c$
- Memiliki Identity : I
  Untuk setiap a Є G; $a*i = i*a = a$
- Memiliki invers a^{-1}
  Untuk setiap a Є G; maka ada a^{-1} Є G; dimana $a*a^{-1} = a^{-1}*a = i$

Jika : Comutative / Abelian Group

Untuk setiap a , b $\in$ G; a • b = b • a

Teorema :

Pada setiap group G, elemen identity dan inverse adalah unique

Bukti :

a. Anggap ada $i_1 \neq i_2 \Rightarrow \quad \underbrace{i_1 \bullet i_2}_{i_2} = \underbrace{i_2 \bullet i_1}_{i_1}$

b. Anggap $[a]^{-1}$: b & c $\Rightarrow$

$$b = b \bullet i = b \bullet (a \bullet c) = (b \bullet a) \bullet c = c$$

Teorema :

Untuk setiap a , b , c $\in$ G , jika :

(i) $a \bullet b = a \bullet c \Rightarrow b = c$ (kiri)

(ii) $a \bullet b = c \bullet b \Rightarrow a = c$ (kanan)

Definisi :

Elemen pembangkit : Ep

$a \in G$; Ep , jika : $a \bullet \underline{\Delta}\ i$, $a^{(1)} = a_j$

$a \bullet a \underline{\Delta}\ a^{(2)}$, $\underbrace{\ldots. a \bullet a \bullet \ldots. \bullet a}_{k}\ \underline{\Delta}\ a^{(k)}$ adalah semua elemen dari G, G demikian disebut Cyclic Group

Definisi :

Orde dari elemen $a \in G$: n terkecil sehingga $a^{(n)} = i$

Definisi :

Subgroup : H

$H \subset G$:Subgroup jika H memenuhi semua persyaratan untuk suatu G

RING : R

- G-Abelian di operasi (+)
- Di (x) :
  - Closure
  - Associative
  - Distributive terhadap (+)
- Memiliki identity multiplikatip

  Catatan :
  - Sebut $I_{(+)} = 0$; $i_{(x)} = 1$
  - Jika jelas, notasi (x) ditiadakan

Definisi :

R disebut Commutative Ring , Rc : Jika untuk semua a , b $\in$ R

Berlaku a x b = b x a

Teorema :

Untuk sebarang a dan b di R :

- a x 0 = 0 x a = 0
- (-a) x b = - (a x b) = a x (-b)

Bukti :

- $a \bullet 0 = a (0 + 0) = a0 + a0$

  $\underbrace{a0 + (-a0)}_{0} = a0 + \underbrace{a0 + (-a0)}_{0}$

  $= a0 + 0$

  $0 = a0$

- $0 = a0 = a (b+(-b)) = ab + a(-b)$

  Maka $a(-b) = -(ab)$

  $(-a)b + ab = (-a+a)b = 0b = 0$

  Maka $(-a)b = -(ab)$

  $(-a)(-b) = -([a(-b)]) = -(-ab) = ab$

Definisi :

Ideal ; I

S⊂R : ideal , jika :

- S : Subroup dari R di (+)
- Untuk semua s∈S dan untuk semua r∈R, maka s x r ∈ S, R = Ring

Definisi :

Integer Ring : Z

= Integer (popsitif, negative, zero) dibawah operasi (+) dan (x)

Jika r, s , a integer, dan ra = s maka disebut :

- s dapat dibagi r
- r pembagi s
- r adalah faktor s

- Integer prima : p ≥ 1 dapat dibagi hanya oleh ± p atau ± 1
- Composite : Integer positif, bukan prima
- Greatest Common Divisor dari integer r dan s , GCD (r,s) adalah integer positif terbesar antara pembagian r dan s
- Least Common Multiple dari dua integer r dan s, LCM (r,s) adalah integer positif terkecil antara pembagian r dan s

- Teorema :

  Algoritma Pembagian
  - s = dq + r , 0 ≤ r < | d |
  - s = bilangan integer yang dibagi
  - d = bilangan integer pembagi
  - q = quotient
  - r = remainder

- Euclidean Division Algorithm : GCD (r,s), r < s

  Tahap iterative:

  $s = q_1 r + r_1$

  $r = q_2 r_1 + r_2$

  $r_1 = q_3 r_2 + r_3$

  :

  :

  $r_{n-2} = q_n r_{n-2} + r_n$

  $r_{n-1} = q_{n+1} r_n + r_{n+1}$

  Proses berhenti jika remainder nol. Remainder terakhir r_n adalah GCD

  r_n = GCD (r,s)

  GCD (r,s) = ar + bs

Contoh :

Cari GCD dari dua integer 22471 dan 3266

22471= (3266)6+2875
= (2875)1+391
= (391)7+138
= (138)2+115
= (115)1+23
= (23)5+0

GCD (3266,22471)= 23

23 =138–(115)1
=138–(391–138x2)=(138)3–391
=(2875–391x7)3–391= (2875)3–(391)22
=(2875)3–(3266–2875x1)22
=(2875)25–(3266)22
=(22471–3266x6)25–(3266)22
=(22471)25–(3266)172

r ≡s(mod d)

FIELD : F

≡ Comutative ring + memiliki :

- Multiplicative identity
- Multiplicative inverse { bagi non zero element }

Catatan :

Untuk semua field F memiliki 2 – struktur:

- Semua elemen adalah group abelian pada penjumlahan (+)
- Non zero elemen termasuk multiplicative group

Dua kelompok (banyak elemen)

- tak hingga
- Berhingga : bermanfaat untuk coding
  Sebutan : Galois Field , GF (q)
  (q : banyak elemen : harus bilangan prima)

POLINOMIAL RING : f (x)

Bentuk :

$$f(x) = f_0 + f_1x + \ldots + f_{n-2}x^{n-2} + f_{n-1}x^{n-1}$$

Derajat f (x) : pangkat terbesar dari x dan koefisien $f_{n-1} \neq 0$

Monic = koefisien pangkat terbesar x adalah 1

Operasi :

- Penjumlahan
  - h (x)=f (x)+g(x)= Σ $(f_i + g_i)x^i$
  - derajat g (x) ≤ derajat f (x)

contoh:

$$f(x) = 1 + 2x + \quad + x^3$$
$$g(x) = 2 + 2x + 2x^2$$
$$h(x) = 3 + 4x + 2x^2 + x^3$$

- Perkalian

$$h(x) = f(x).g(x) = \sum_i \left(\sum_{j=0}^{i} f_i g_{i-j} \right) .x^i$$

  - Derajat h (x)=jumlah derajat f (x)+g(x)

Definisi :

- Irreducible polynomial = non zero polynomial p(x) yang dapat dibagi hanya oleh p (x)
- Primitive polynomial = sebuah monic irreducible polynomial
- Divisible/Factor b(x) dikatakan terbagi oleh d(x). Atau d(x) adalah factor dari b(x). Jika ada polynomial q(x), sehingga b(x)=q(x)•d(x)
- GCD (a(x),b(x))
- Polynomial monic berderajat tertinggi yang adalah factor dari a(x) dan b(x)
- Relatively Prime
- a(x) dan b(x) adalah relatively prime,
- jika GCD [a(x),b(x)]=1

Teorema :

- Jika GCD [a(x),b(x)]=d(x) maka terdapat polynomial r(x) dan s(x), sehingga : a(x)r(x)+b(x)s(x)=d(x)
- Untuk setiap b(x) dan m(x)≠0, terdapat q(x) dan r(x), sehingga:
  b(x)=q(x)m(x)+r(x)
  DEG[r(x)]<DEG[m(x)]
  (Division Algorithm)
- b(x)MOD m(x)=r(x)
  b(x)≡r(x) (mod m(x))

Definisi :

Akar / Root

Anggap f(x) adalah polynomial dengan DEG m > 2 OVER GF(g)

Jika (x–a), a ∈ GF(g) adalah factor dari f(x), maka a disebut "AKAR" dari f(x) dan f(a)=0

GALOIS FIELD : GF(p)

Asumsi : $a(\neq 0,\neq 1) \in GF(p)$

Bentuk : $a ; a \bullet a = a^2 ; a \bullet a \bullet a = a^3 ; \bullet\bullet\bullet$

Bahasan :

- $a, a^2, a^3, \ldots, a^k \in GF(p)$ closure
- p berhingga $\Rightarrow$ pembentukkan dilanjutkan, maka terjadi : pengulangan, ada :$a^j = 1 (a^j \bullet a = a)$

Definisi :

- j terkecil sehingga $a^j = 1$ orde dari a
- a : primitive jika : j = p-1

Contoh :

GF (5)=(0,1,2,3,4)

a \ j	a			
	1	2	3	4
2	2	4	8 = 3	16 = 1
3	3	9 = 2	27 = 2	81 = 1
4	4	16 = 1	64 = 4	256 = 1

Teorema :

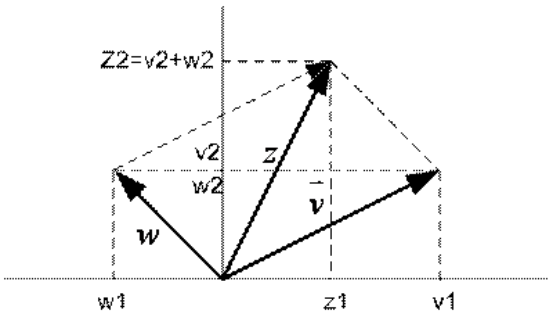
- Untuk setiap $a \in GF(q), a \neq 0, a^{q-1} = 1$
- Orde dari $a \in GF(q), a \neq 0$, adalah pembagi dari (q-1)

VEKTOR / MATRIK

Vektor 2 dimensi

Notasi : $\vec{v} = (v_1, v_2)$

$\vec{w} = (w_1, w_2)$



Operasi :

- Penjumlahan

  $z=v+w=(v_1+w_1, v_2+w_2)=(z_1,z_2)$

- Perkalian (Inner/Skalar Product)

  $y=v \bullet w=v_1w_1+v_2w_2$

- Matrix 1 – 2

  $A=[a_1\ a_2]$

  $B=[b_1\ b_2]$

- Penjumlahan

  $$\mathbf{C}=\mathbf{A}+\mathbf{B}=[\mathbf{a_1}\quad \mathbf{a}_2]+[\mathbf{b_1}\quad \mathbf{b_2}]$$

  $$\mathbf{C}=[(\mathbf{a_1}+\mathbf{b}_1)\quad (\mathbf{a_2}+\mathbf{b_2})]$$

- Perkalian

  $$\mathbf{D}=\mathbf{A}\cdot\mathbf{B}^{\mathrm{T}}=\begin{bmatrix}\mathbf{a_1} & \mathbf{a_2}\end{bmatrix}\begin{bmatrix}\mathbf{b_1}\\ \mathbf{b_2}\end{bmatrix}=\mathbf{a_1b_1}+\mathbf{a_2b_2}$$

- Perkalian dengan satu skalar : *k*

  (1) Vektor : $\vec{v} \Leftrightarrow (v_1, v_2)$

  $$\vec{w} = k\vec{v} \Leftrightarrow (kv_1, kv_2)$$

  (2) Matrix Baris : A = $\begin{bmatrix}a_1 & a_2\end{bmatrix}$

  $$B = k\begin{bmatrix}a_1 & a_2\end{bmatrix} = \begin{bmatrix}ka_1 & ka_2\end{bmatrix}$$

Catatan:

- Ada kesamaan dalam operasi vektor dengan operasi matrix
- Struktur : $(g_1, g_2, \cdots, g_n)$ disebut n –Tuple
- Hitung ^{2} an diatas berlaku untuk vector dimensi –n
- Jika $\vec{v} \in (v_1, v_2, \ldots, v_n)$ dan $\vec{v}_i \in F$ untuk semua i, maka $\vec{v}$ disebut vector "over suatu field F

VEKTOR SPACE / RUANG VEKTOR : $\vec{V}$

Definisi : $\vec{v}$ adalah set vector " over " medan F , yang memenuhi:

(1) Group Abelian pada penjumlahan.

(2) Untuk setiap $\vec{v} \in \vec{V}$ dan $a \in F$

$a\vec{v} \in \vec{V}$

(3) Untuk setiap $\vec{u}, \vec{v} \in \vec{V}$ dan setiap $a, b \in F$,berlaku:

* $a(\vec{u} + \vec{v}) = a\vec{u} + a\vec{v}$
* $(a + b)\vec{u} = a\vec{u} + b\vec{u}$
* $(a\,b)\,\vec{u} = a(b\vec{u})$
* $1 \cdot \vec{u} = \vec{u}$
* $0 \cdot \vec{u} = 0$

SUBSPACE : $\vec{S}$

Definisi : Suatu subset $\vec{S}$ dari suatu ruang $\vec{V}$ adalah subspace jika $\vec{S}$ memenuhi sifat suatu vector space.

Teorema :

Set $\vec{S}$ yang elemennya adalah semua kombinasi linier dari sebanyak m - vektor : $\vec{v}_{1,}$, $\vec{v}_2$ … $\vec{v}_m \in \vec{V}$ adalah suatu **SUB-SPACE**.

$$\vec{u} = k_1\vec{v}_1 + k_2\vec{v}_2 + k_3\vec{v}_3 + \ldots + k_m\vec{v}_m$$

Contoh :

$\vec{V}$: Set dari semua S -TUPLE over GF(2)

AMBIL $\vec{v}_1$ = (0 0 1 1 1)

$\vec{v}_2$ = (1 1 1 01)

Maka

(1) 0 0 → 0(00111) + 0(11101) = (00000)

(2) 0 1 → 0(00111) + 0(11101) = (111O1)

(3) 1 0 → 1(00111) + 0(11101) = (00111)

(4) 1 1 → 1(00111) + 1(11101) = (11010)

Terlihat

(1)+ (3) = (4)

SALING BEBAS

Suatu set $\vec{v}_1, \vec{v}_2, \ldots \vec{v}_K \in \vec{V}$ over F, ADALAH

(i) Saling bergantungan Linier jika :

$$\sum_{i=1}^{k} a_i \vec{v}_i = 0; \quad a_i \in F, \text{tak semua } a_i = 0$$

$$a_1\vec{v}_1 + a_2\vec{v}_2 + \ldots . a_k\vec{v}_k = 0$$

(ii) Saling Bebas Linier, jika :

$$\sum_{j=1}^{k} a_j \vec{v}_j = 0; \quad \text{hanya jika semua } a_j \in F = 0$$

$$a_1\vec{v}_1 + a_2\vec{v}_2 + \ldots . a_k\vec{v}_k = 0$$

Contoh :

5 – Tuple over GF(2)

vektor : (1 0 0 0 0)
(0 1 0 0 0)
(0 0 1 0 0)
(0 0 0 1 0)
(0 0 0 0 1)

Adalah Saling Bebas

Bukan satu-satunya : Salah satu atau semua dapat diganti dengan kombinasi linier dari vektor diatas asal tetap berbeda satu sama lain

VEKTOR BASIS

Definisi :

Suatu set $\vec{B} = (\vec{v}_1, \vec{v}_2, \cdots \vec{v}_n) \in \vec{V}$ disebut basis dari $\vec{V}$ jika :

(1) $\vec{v}_i$ saling bebas dan

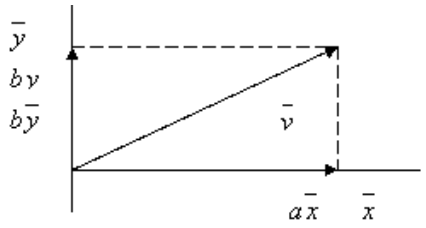
(2) Semua $\vec{w}_k \in \vec{V}$ dapat ditulis sebagai $\vec{w}_k = \sum_{i=1}^{n} a_i \vec{v}_i$

Banyak vector di basis disebut : Dimensi dari $\vec{V}$. Atau $\vec{V}$ adalah
Ruang Vektor n –Dimensional jika banyak vector dibasisnya adalah : n

Teorema :

Setiap ruang vektor $\vec{V}$ memiliki sedikitnya satu vektor basis

Contoh : $\vec{V}$ = 2-Dimensi



Basis : $(\vec{x}, \vec{y})$

$$\vec{V} = a\vec{x} + b\vec{y}$$

VII. LINIER BINARY BLOCK CODE

Linear binary code (Moreira & Farrell, 2006) $\rightarrow$ GF(2) $\rightarrow$ $\{0,1\}$ $\rightarrow$ Modulo 2 Arithmetic

Nonbinary code $\rightarrow$ c : Reed-salomon code $\rightarrow$ GF(2^m)

m = Jumlah BIT per simbol

Linear Block codes = Group Codes

= Parity Check Codes

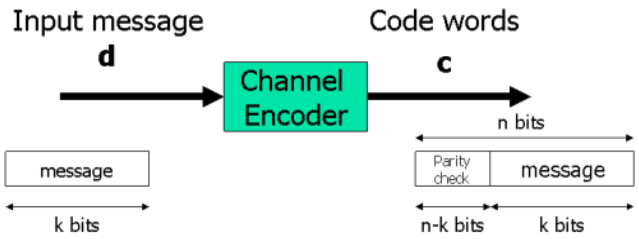
Definisi :

- Panjang blok = n $\rightarrow$ 2^n n-Tuple
- Panjang data = k $\rightarrow$ 2^k data words/code word (k<n)
    - Linear code (n,k)
- Encoding oleh generator matriks & parity_check matriks
- Syndrome untuk error correction
- Standard array untuk table – lookup decoding
- Weight enumerator & probability of decoding error

Deskripsi Matriks dari Kode Linier

- Set dari semua n-Tuple dimana masukannya dipilih dari GF(2) adalah suatu ruang vector.
- Sebuah kode linier adalah sebuah ruang vector, dimana masing- masing kode word adalah sebuah vector.
- Set dari vector dengan panjang n disebut blok kode linier jika dan hanya jika termasuk subspace dari vector space n-Tuple

Kode linier (n,k) dilambangkan dengan sebuah k$\times$n matriks generator G (terdiri dari k-baris,n-kolom)



Dimana : $d_i \quad , 0 \le i \le k-1$

$g_i \quad , 0 \le i \le k-1$

d =Data informasi yang dikodekan

$d = (d_0, d_1, \cdots, d_{k-1})$

C : Code word

Adalah kombinasi linier data & M.Gen

C= $d_0 g_0 + d_1 g_1 + \cdots + d_{k-1} g_{k-1} = (C_0, C_1, \cdots, C_{n-1})$

Sebuah code word terdiri dari :

- k-bit Informasi
- n-k bit Parity – Check

Persamaan Parity-Check :

$$\gamma_j = \sum_{i=0}^{k-1} P_{i,j} di$$

$$= P_{0,j} d_0 + P_{1,j} d_1 + \cdots + P_{k-2,j} d_{k-2+} P_{k-1,j} d_{k-1}$$

Dimana $0 \le j \le n-k-1$

Dan $P_{i,j} = 0$ atau 1

$$C = (C_0, C_1, \cdots, C_{n-k-1}, C_{n-k}, \cdots, C_{n-1})$$

$$= (\underbrace{\gamma_0, \gamma_1, \cdots, \gamma_{n-k-1}}_{ParityChedk}, \underbrace{d_0, d_1, \cdots, d_{k-1}}_{Informasi}$$

Parity Check :

$$C_0 = \gamma_0 = P_{0,0} d_0 + P_{1,0} d_1 + \cdots + P_{k-2,o} d_{k-2} + P_{k-1,0} d_{k-1}$$

$$C_1 = \gamma_1 = P_{0,1} d_0 + P_{1,1} d_1 + \cdots + P_{k-2,1} d_{k-2} + P_{k-1,1} d_{k-1}$$

$$C_2 = \gamma_2 = P_{0,2} d_0 + P_{1,2} d_1 + \cdots + P_{k-2,2} d_{k-2} + P_{k-1,2} d_{k-1}$$

$$C_{n-k-1} = \gamma_{n-k-1} = P_{0,n-k-1} d_0 + P_{1,n-k-1} d_1 + \cdots + P_{k-2,n-k-1} d_{k-2} + P_{k-1,n-k-1} d_{k-1}$$

$\vdots$

Dengan bentuk matriks

$$Parity \rightarrow (C_0, C_1, C_2, \cdots C_{n-k-1}) = (d_0, d_1, \cdots, d_{k-1}) \begin{bmatrix} P_{0,0} & P_{0,1} & \cdots & P_{0,n-k-1} \\ P_{1,0} & P_{1,1} & \cdots & P_{1,n-k-1} \\ P_{2,0} & P_{2,1} & \cdots & P_{2,n-k-1} \\ \vdots & & & \\ P_{k-1,0} & P_{k-1,1} & \cdots & P_{k-1,n-k-1} \end{bmatrix}$$

$$Informasi \rightarrow (C_{n-k}, C_{n-k+1}, \cdots, C_{n-1}) = (d_0, d_1, \cdots d_{k-1}) \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

$$C = (C_0, C_1, \cdots C_{n-k-1}, C_{n-k}, \cdots, C_{n-1}) = (d_0, d_1, \cdots, d_{k-1}) \left\lfloor P_{k\times(n-k)} \quad \vdots \quad I_k \right\rfloor$$

$P_{k\times(n-k)} = k \times (n-k)$ *Matrik*

I_k = $k \times k$ *Matriks Identiti*

MATRIKS GENERATOR : G

C = d . G

$$G = \left\lfloor P_{k\times(n-k)} \quad \vdots \quad I_k \right\rfloor$$

$$= \begin{bmatrix} P_{0,0} & P_{0,1} & \cdots & P_{0,n-k-1} & 1 & 0 & 0 & \cdots & 0 \\ P_{1,0} & P_{1,1} & \cdots & P_{1,n-k-1} & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & & & & & \\ P_{k-1,0} & P_{k-1,1} & \cdots & P_{k-1,n-k-1} & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix}$$

Bentuk Sistematik ↵

$$G = \begin{bmatrix} g_{00} & g_{01} & \cdots & g_{0,n-1} \\ g_{10} & g_{11} & \cdots & g_{1,n-1} \\ \vdots & & & \\ g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,n-1} \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \\ \vdots \\ r_{k-1} \end{bmatrix}$$

$$= \begin{bmatrix} C_0 & C_1 & \cdots & C_{n-1} \end{bmatrix}$$

Dengan :

$g_{ij} \in GF(q)$

$r_j = \lfloor g_{j0}, g_{j1}, \cdots, g_{j,n-1} \rfloor$

$= Vektor \quad baris \quad dari G \in \vec{V}_m$

$C_m = \lfloor g_{0m} \quad g_{1m} \quad \cdots \quad g_{k-1,m} \rfloor$

$= Vektor \quad kolom \quad dari \quad G \in \vec{V}_k$

Bentuk :

- Sistematik
- Non sistematik, akan = sistematik setelah dioperasikan baris dan kolom.

OPERASI ELEMENTER

- Terhadap baris
  - Menukar dua baris
  - Mengalikan suatu baris dengan satu scalar
  - Menjumlahkan beberapa $\{skalar \times baris\}$

- Terhadap kolom
  - Idem Baris
  - Echelon Canonical Form
  - Non zero leading elemen dari satu baris,adalah : 0 (nol)
  - Elemen lain dari kolom yang memuat leading elemen : 0 (nol)
  - Leading elemen dari satu baris terletak lebih kanan dari posisi leading elemen baris sebelumnya

Dapat disusun lewat operasi elementer

Contoh 1:

Diketahui : Over GF(3) $\left(\begin{array}{cc|ccc} 2 & 2 & 2 & 0 & 2 \\ 1 & 0 & 1 & 2 & 2 \\ 2 & 2 & 0 & 1 & 2 \end{array}\right)$

Langkah-langkah :

- (baris 1) x 2 $\left(\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 2 & 2 \\ 2 & 2 & 0 & 1 & 2 \end{array}\right)$
- (baris 1 x 2)+ (baris 2) $\left(\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 1 \\ 0 & 2 & 0 & 2 & 1 \\ 2 & 2 & 0 & 1 & 2 \end{array}\right)$
- (baris 2) x 2 $\left(\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 2 \\ 2 & 2 & 0 & 1 & 2 \end{array}\right)$
- (baris 2 x 2) + (baris 3) $\left(\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 0 & 0 \end{array}\right)$
- (baris 3 x 2) $\left(\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 2 \\ 1 & 2 & 0 & 0 & 0 \end{array}\right)$
- (baris 3 x 2)+ (baris 1) $\left(\begin{array}{cc|ccc} 0 & 2 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 2 \\ 1 & 2 & 0 & 0 & 0 \end{array}\right)$

Bentuk Sistematik

- kolom 1 ditukar kolom $\left(\begin{array}{cc|ccc} 1 & 2 & 1 & 0 & 0 \\ 2 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{array}\right) = \left(\begin{array}{cc|c} 1 & 2 & \\ 2 & 1 & I_3 \\ 0 & 2 & \end{array}\right)$

Contoh 2:

- Perhatikan kode linier (6,3) dengan matriks generator dalam bentuk nonsistematik

$$G = \begin{pmatrix} 1\,1\,0\,1\,1\,0 \\ 1\,1\,0\,0\,0\,1 \\ 0\,1\,1\,0\,1\,0 \end{pmatrix} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \end{pmatrix}$$

- Dengan operasi baris elementer diperoleh matriks bentuk sistematik :

$$G = \begin{pmatrix} 1\,0\,1\,\mathbf{1}\,0\,0 \\ 0\,1\,1\,0\,\mathbf{1}\,0 \\ 1\,1\,0\,0\,0\,\mathbf{1} \end{pmatrix} = \begin{pmatrix} g_0+g_2 \\ g_2 \\ g_1 \end{pmatrix}$$

TABEL CODEWORDS

Kode blok pada tabel dibawah adalah kode linier (6,3) yang dibangkitkan oleh 2 matriks generator terdahulu.

Message Informasi			Codewords											
			Nonsistematik						Sistematik					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1	0	1	1	0	0	0	1
0	1	0			.						.			
0	1	1			.						.			
1	0	0			.						.			
1	0	1	1	0	1	1	0	0	0	1	1	1	0	1
1	1	0			.						.			
1	1	1			.						.			

Contoh :

d=(1 0 1) dikodekan dengan bentuk :

- Nonsistematik

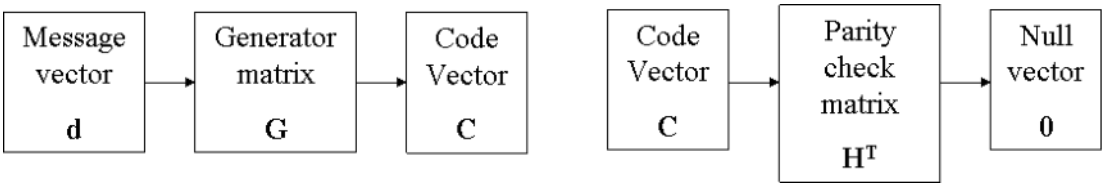
$$C = (101)\left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{array}\right) = (101100)$$

- Sistematik

$$C = (101)\left(\begin{array}{ccc|ccc} 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right) = (011101)$$

MATRIKS GENERATOR & MATRIKS PARITY CHECK

Operasi : Matriks generator & matriks parity check



MATRIKS PARITY CHECK

Jika kode linier (n,k)maka ada dual code (n,n-k) dengan 2^{n-k} code word,yang merupakan null-space dari kode linier(n,k)

H :

- Generator matriks untuk dual code
- Ortogonal dari G
- Matriks parity check dari code linier
- (n-k) x n matriks

$C.H^T = \mathbf{0}$ **0** = vector baris semua nol dengan (n-k) elemen

$G.H^T = \mathbf{0}$ **0** = matriks k x (n-k) dengan semua elemen nol

$$\text{Jika } G = [P_{kx(n-k)} \vdots I_K]$$

$$\text{Maka } H = [I_{n-k} \vdots P^T_{(n-k)xk}]$$

$$= \begin{bmatrix} 1 & 0 & 0 & \ldots 0 & P_{0,0} & \ldots & P_{k-1,0} \\ 0 & 1 & 0 & \ldots 0 & P_{0,1} & \ldots & P_{k-1,1} \\ & \vdots & & & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots 1 & P_{0,n-k-1} & \ldots & P_{k-1,n-k-1} \end{bmatrix}$$

Contoh :

- Sebuah kode linier (7,4) dengan matriks generator :

$$G = \left(\begin{array}{cccc|cccc} 1 & 0 & 1 & & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & & 0 & 0 & 0 & 1 \end{array}\right) = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix}$$

- Vektor informasi d=(0 1 1 0) dikodekan ke codeward sbb:

$$\mathbf{c = d \,.\, G}$$

$$c = (0\ 1\ 1\ 0) \left(\begin{array}{cccc|cccc} 1 & 0 & 1 & & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & & 0 & 0 & 0 & 1 \end{array}\right)$$

$$= (1\ 0\ 1\ 0\ 1\ 1\ 0)$$

$$c = \sum_{i=0}^{3} d_i g_i = 0.g_0 + 1.g_1 + 1.g_2 + 0.g_3$$

$$= (0110100) + (1100010)$$

$$= (1010110)$$

Table berikut ada 16 code word dari 2^4=16 information word

Informasi bits				**Codeword**						
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	0	1
0	0	1	0	1	1	0	0	0	1	0
0	0	1	1	0	0	1	0	0	1	1
0	1	0	0	0	1	1	0	1	0	0
0	1	0	1	1	0	0	0	1	0	1
0	1	1	0	1	0	1	0	1	1	0
0	1	1	1	0	1	0	0	1	1	1
1	0	0	0	1	0	1	1	0	0	0
1	0	0	1	0	1	0	1	0	0	1
1	0	1	0	0	1	1	1	0	1	0
1	0	1	1	1	0	0	1	0	1	1
1	1	0	0	1	1	0	1	1	0	0
1	1	0	1	0	0	1	1	1	0	1
1	1	1	0	0	0	0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1

IMPLEMENTASI ENKODER DARI BLOK KODE LINIER

Dari persamaan :

$$C = d \cdot G$$

Dimana :

- d : k-tuple information : $(d_0,d_1,\ldots d_{k-1})$
- c : n-tuple code word
- G : generator matriks

Contoh :

Kode linier (7,4) dengan matriks generator :

$$G=\begin{array}{c} \begin{matrix} P & & & I & & & \end{matrix} \\ \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right) \end{array} \qquad d=(d_0, d_1, d_2, d_3)$$

Parity check bits r_i , i=0,1,2

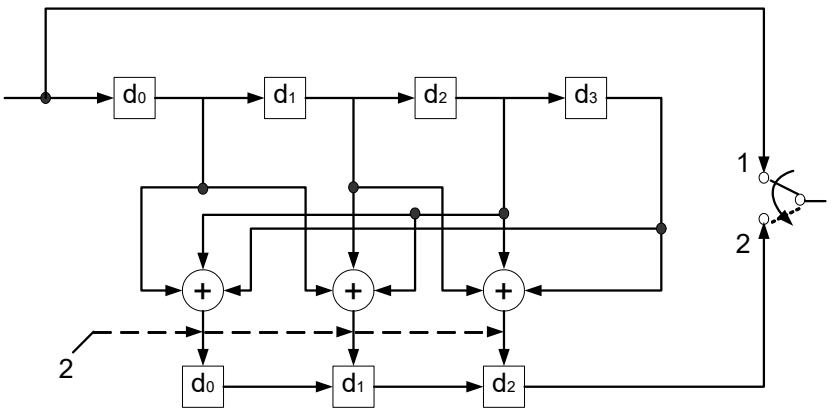
$$(r_0, r_1, r_2)=(d_0,d_1,d_2,d_3)\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

$$=(d_0+d_2+d_3,\ d_0+d_1+d_2,\ d_1+d_2+d_3)$$

$$\mathbf{c=(r_0, r_1, r_2, d_0, d_1, d_2, d_3)}$$

Contoh :

- d = (0 1 1 0)
- Digeser ke shift register
- Maka n-k parity-check digits dibentuk = 3 → $r_0 = 1$, $r_1 = 0$, $r_2 = 0$
- Sehingga : c = (1 0 0 0 1 1 0)



VIII. CYCLIC CODE

DESKRIPSI CYCLIC CODE (MOREIRA & FARRELL, 2006)

Cyclic Code banyak dipakai pada sistem komunikasi yang memiliki blok kode yang panjang dengan jumlah Code Word yang besar. Cyclic Code juga efektif pada pencarian Error & Perbaikannya.

Definisi:

Suatu kode linier (n, k) C over GF(q) disebut 'Cyclic Code', jika :

Bila c = (c_0 , c_1 ,……. c_{n-1}) adalah satu kode di C, dan

$c^{(1)}$ = (c_{n-1} , c_0 , c_1 , …. , c_{n-2}) juga ada di C.

Asosiasi:

Code word C dengan code polynomial c(x)

C = (c_0 , c_1 , … , c_{n-1}) ditulis c(x) = $c_0 + c_1x + c_2x^2 + \dots + c_{n-1}x^{n-1}$

Contoh:

Code (7,4) dengan $g(x) = 1 + x + x^3$

$\therefore c(x) = d(x).g(x) \text{ MOD } (1 + x^7)$; GF(2)

Data				d(x)	c(x)=d(x).g(x)	codeword C						
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	$1 + x + x^3$	1	1	0	1	0	0	0
0	**1**	0	0	x	$x + x^2 + x^4$	0	1	1	0	1	0	0
1	**1**	0	0	$1 + x$	$1 + x^2 + x^3 + x^4$	1	0	1	1	1	0	0
0	0	**1**	0	x^2	$x^2 + x^3 + x^5$	0	0	1	1	0	1	0
1	0	**1**	0	$1 + x^2$	$1 + x + x^2 + x^5$	1	1	1	0	0	1	0
0	**1**	**1**	0	$x + x^2$	$x + x^3 + x^4 + x^5$	0	1	0	1	1	1	0
1	**1**	**1**	0	$1 + x + x^2$	$1 + x^4 + x^5$	1	0	0	0	1	1	0
0	0	0	**1**	x^3	$x^3 + x^4 + x^6$	0	0	0	1	1	0	1
1	0	0	**1**	$1 + x^3$	$1 + x + x^4 + x^6$	1	1	0	0	1	0	1
0	**1**	0	**1**	$x + x^3$	$x + x^2 + x^3 + x^6$	0	1	1	1	0	0	1
1	**1**	0	**1**	$1 + x + x^3$	$1 + x^2 + x^6$	1	0	1	0	0	0	1
0	0	**1**	**1**	$x^2 + x^3$	$x^2 + x^4 + x^5 + x^6$	0	0	1	0	1	1	1
1	0	**1**	**1**	$1 + x^2 + x^3$	$1 + x + x^2 + x^3 + x^4 + x^5 + x^6$	1	1	1	1	1	1	1
0	**1**	**1**	**1**	$x + x^2 + x^3$	$x + x^5 + x^6$	0	1	0	0	0	1	1
1	**1**	**1**	**1**	$1 + x + x^2 + x^3$	$1 + x^3 + x^5 + x^6$	1	0	0	1	0	1	1
x^0	x^1	x^2	x^3			x^0	x^1	x^2	x^3	x^4	x^5	x^6

$x^0\ x^1\ x^2\ x^3$

$$\begin{array}{r} 1\ 0\ 0\ 0 \\ 0\ 1\ 0\ 0 \\ \hline 1\ 1\ 0\ 0 \\ 0\ 1\ 1\ 0 \end{array} + \quad \Rightarrow \quad \begin{array}{r} 1\ 1\ 0\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 1\ 0\ 0 \\ \hline 1\ 0\ 1\ 1\ 1\ 0\ 0 \\ 0\ 1\ 0\ 1\ 1\ 1\ 0 \end{array} + \begin{array}{l} = c \\ = c^{(1)} \end{array}$$

Teorema :

Generator polinomial g(x) dari kode siklis (n, k) adlah faktor dari ($1+x^n$)

Test : $\dfrac{1+x^7}{1+x+x^3} = \begin{array}{l} x^0 \\ 1\ 1\ 0\ 1 \leftarrow x^3 \end{array} \overset{x^7}{\overbrace{\begin{array}{r} 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1 \end{array}}}$

$$\begin{array}{l} L\quad H \\ 1\ 1\ 0\ 1 \end{array} \quad 1\ 1\ 1\ 0\ 1 \leftrightarrow \overset{h(x)=1+x+x^2+x^4}{\overbrace{\big)\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1}} \rightarrow 1+x^7$$

$$\begin{array}{r} 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 0\ 1\ 1\ 0 \\ 1\ 1\ 0\ 1\ 0 \\ \hline 1\ 0\ 1\ 1\ 1 \\ 1\ 1\ 0\ 1 \\ \hline 1\ 1\ 0\ 1 \\ 1\ 1\ 0\ 1 \\ \hline 0\ 0\ 0\ 0 \end{array}$$

$(1+x^7) = (1+x+x^3)\ (1+x+x^2+x^4)$

Bukti :

Kode (n, k) $\rightarrow c(x) = c_0 + c_1x + \ldots\ldots + c_{n-1}x^{n-1}$

$d(x) = d_0 + d_1x + \ldots\ldots + d_{k-1}x^{k-1}$

$$\left.\begin{array}{l} \therefore DEG[c(x)] \le n-1 \\ DEG[d(x)] \le k-1 \end{array}\right\} \rightarrow \begin{array}{l} DEG[g(x)] = n-k \\ DEG[x^k.g(x)] = n \end{array}$$

$$\therefore x^k g(x) = (x^n + 1) + g^{(k)}(x) \qquad (*)$$

g(x) suatu POLINOM KODE → juga $g^{(k)}(x)$

atau $g^{(k)}(x) = a(x).g(x)$ (**)

(*) & (**)

$x^n+1 = \{ x^k + a(x) \}$

dengan kata lain: g(x) : faktor dari $x^n + 1$

Teorema :

Jika g(x) adalah polynomial berderajat (n–k) dan adalah faktor dari $(1+x^n)$, maka g(x) membangkitkan kode siklis (n, k)

STRUKTUR KODE SISTEMATIK

Diketahui: c(n, k)

- $d = (d_0, d_1, \dots , d_{k-1})$
  $d(x) = d_0 + d_1x + \dots + d_{k-1}x^{k-1}$
- $g(x) = 1 + g_1x + \dots + g_{n-k}x^{n-k}$

Dicari:

$c = (p_0 , p_1 , \dots p_{n-k-1} , d_0 , d_1 , \dots d_{k-1})$

$c = (c_0 , c_1 , \dots\dots\dots\dots\dots , \dots c_{n-1})$

Solusi:

$x^{n-k}d(x) = d_0x^{n-k} + d_1x^{n-k+1} + \dots + d_{k-1}x^{n-1}$

DEG [g(x)] = n - k

DEG [x^{n-k} d(x)] = n

Jadi $x^{n-k}d(x)$ dapat dibagi g(x)

$x^{n-k}d(x) = a(x)\, g(x) + b(x)$

Dengan : $b(x) = b_0 + b_1x + \dots + b_{n-k-1}x^{n-k-1}$

Atau $b(x) + x^{n-k-1}d(x) = a(x)\, g(x) \quad \Delta = c(x)$

$b_0 + b_1x + \dots + b_{n-k-1}x_n^{-k-1} + d_0x_{n-k} + \dots + d_{k-}1x^{n-1}$

Contoh : 1 1 1 0 0 0 0

- $d = (1\ 1\ 1\ 0) \rightarrow d(x) = 1 + x + x^2 + 0x^3$
  $g(x) = 1 + x + x^3 = (1\ 1\ 0\ 1)$

- Kode sistematik (7 , 4) = ?

Jawab :

n = 7

k = 4

$x^3 d(x) = x^3 + x^4 + x^5 + x^6 = (0\ 0\ 0\ 1\ 1\ 1\ 0)$

Mencari b(x) : { sisa hasil bagi $\frac{x^{n-k} d(x)}{g(x)}$ }

g(x)

$$
\begin{array}{r|l}
 & x^3 d(x) \\
1\ 1\ 0\ 1 & 0\ 0\ 0\ 1\ 1\ 1\ 0 \qquad 0\ 1\ 1\ 0 \\
 & \vdots\ \vdots\ \vdots 0\ 0\ 0\ 0 \\
\hline
 & \vdots\ \vdots\ 0\ 1\ 1\ 1 \\
 & \vdots\ \vdots\ 1\ 1\ 0\ 1 \\
\hline
 & \vdots\ 0\ 1\ 0\ 1 \\
 & \vdots\ \ 1\ 1\ 0\ 1 \\
\hline
 & 0\ 1\ 0\ 0 \\
 & 0\ 0\ 0\ 0 \\
\hline
 & 0\ 1\ 0
\end{array}
$$

$\rightarrow b(x) = 0x^0 + 1x^1 + 0x^2$

Code :

$d^{(3)}$: 0 0 0 1 1 1 0

b : 0 1 0 0 0 0 0

c = b+$d^{(3)}$: 0 1 0 1 1 1 0

MATRIKS GENERATOR

Catatan :

- (n , k) code : subspace dari $\overline{v_n}$
- Matriks generator : vektor basis

Satu pilihan : Code dari vektor basis data

Contoh 1:

- Kode (7, 4) dengan $g(x) = 1 + x^2 + x^3$

d	b	G
1 0 0 0	1 0 1	1 0 1 1 0 0 0
0 1 0 0	1 1 1	1 1 1 0 1 0 0
0 0 1 0	1 1 0	1 1 0 0 0 1 0
0 0 0 1	0 1 1	0 1 1 0 0 0 1

$$\rightarrow G = \begin{bmatrix} 1\,0\,1\,1\,0\,0\,0 \\ 1\,1\,1\,0\,1\,0\,0 \\ 1\,1\,0\,0\,0\,1\,0 \\ 0\,1\,1\,0\,0\,0\,1 \end{bmatrix}$$

- Null-Space/ Parity Check Matrix

  Seperti pada Block Code : GF(2) $\quad G = \left[P_{k,n-k} \vdots I_k\right] \Rightarrow \left[I_{n-k} \vdots P^T{}_{n-k,k}\right]$

Contoh 2:

$$G = \left[\begin{array}{c:c} 1\,0\,1 & 1\,0\,0\,0 \\ 1\,1\,1 & 0\,1\,0\,0 \\ 1\,1\,0 & 0\,0\,1\,0 \\ 0\,1\,1 & 0\,0\,0\,1 \end{array}\right] \Rightarrow H = \left[\begin{array}{c:c} 1\,0\,0 & 1\,1\,1\,0 \\ 0\,1\,0 & 0\,1\,1\,1 \\ 0\,0\,1 & 1\,1\,0\,1 \end{array}\right]$$

d = (1 0 1 1) $\rightarrow$ c = d.G

$$= \left[1\,0\,1\,1\right]_{1*4} \begin{bmatrix} 1\,0\,1\,1\,0\,0\,0 \\ 1\,1\,1\,0\,1\,0\,0 \\ 1\,1\,0\,0\,0\,1\,0 \\ 0\,1\,1\,0\,0\,0\,1 \end{bmatrix}_{4*7}$$

$$= \left[0\,0\,0\,1\,0\,1\,1\right]_{1*7}$$

C.H^T : { $\sum$ baris 4, 6& 7 dari H^T }

$$\begin{array}{r} 4: \; 1\,0\,1 \\ 6: \; 1\,1\,0 \\ \underline{7: \; 0\,1\,1} \\ 0\,0\,0 \end{array} +$$

Jika : ε = (0 0 0 0 1 0 0)

r = c′ = c+ε = (0 0 0 1 1 1 1)

C′.H^T : { $\sum$ baris 4, 5, 6, 7 dari H^T }

$$\begin{array}{l} 4: 1\,0\,1 \\ 5: 1\,1\,1 \\ 6: 1\,1\,0 \\ \underline{7: 0\,1\,1} \\ 1\,1\,1 \end{array} +$$

r. H^T = S

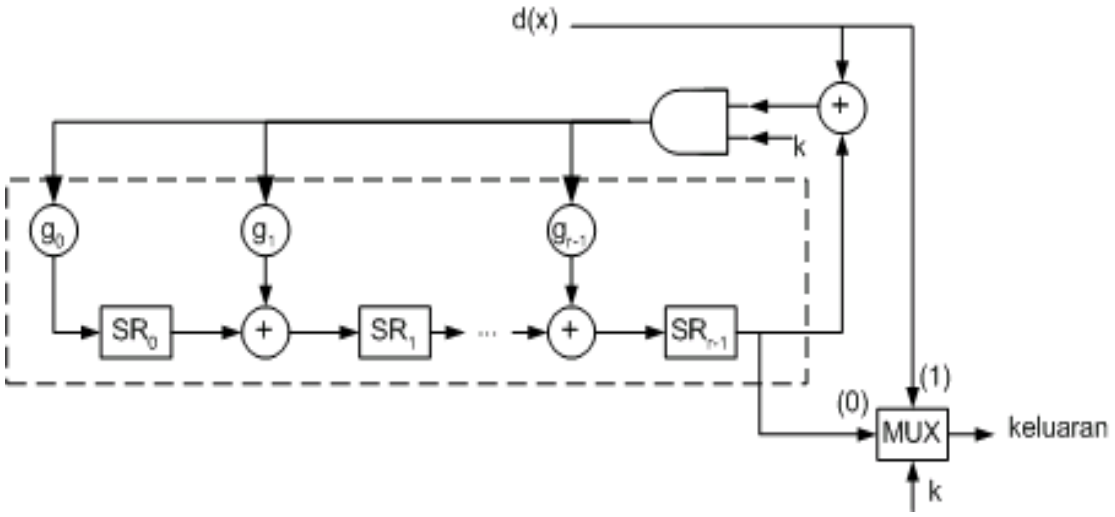
$$\begin{bmatrix}0&0&0&1&1&1&1\end{bmatrix}\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\\1&0&1\\1&1&1\\1&1&0\\0&1&1\end{bmatrix} = \begin{bmatrix}1&1&1\end{bmatrix}$$

ENKODER UNTUK CYCLIC CODE

<u>Kode Sistematik</u> :

X^{n-k} d(x) = a(x) g(x) + b(x)

c(x) = b(x) + x^{n-k} d(x) → b(x) : sisa hasil bagi x^{n-k} d(x) / g(x)



Penghasil sisa x^r d9x) / g(x)

(OP MOD g(x))

1. k=1 : d(x) dimasukan satu-satu, High order first, ke sistem & ke output (lewat MUX)
2. Setelah data habis : SR Sindrom dari x^r d(x) { sisa hasil bagi }
3. k=0 : isi SR dikeluarkan lewat MUX

Catatan :

Sebelum langkah 1 dimulai SR harus dikosongkan terlebih dahulu

Contoh 1:

- Kode sikllis (7,4) dengan $g(x) = 1 + x + x^3$

  $d(x) = x + x^2 + x^3$
- Gambar sistem Enkoder, serta Hitung Isi SR pada setiap langkah
- Tulis keluaran sistem dan teliti apakah keluaran tsb benar[2] satu kode
- g(x) : satu faktor dari : x^{n-1}

  atau : $x^{n-1} = g(x)h(x)$

  → kode (n,k) : DEG [g(x)] = n-k

  $\therefore$ DEG [h(x)] = n-(n-k) = k

? Dapatkah C(x) dibangkitkan oleh h(x)

$C(x)= c_0 + c_1x +...+ c_{n-1}x^{n-1} = a(x)g(x)$

$C(x)h(x) = a(x)g(x)h(x) = 0 \text{ MOD } (x^n+1)$

DKL : Koefisien dari suku x^j di $C(x)h(x) =0 \rightarrow h_0 c_j + h_1 c_{j-1} + \dots + h_k c_{j-k} = 0$

Tapi : $h_k = 1 \rightarrow$ maka :

$$C_{j-k} = h_0 c_j + h_1 c_{j-1} + \dots + h_{k-1} c_{j-k-1} \quad (j = 0, 1,\dots , n-1)$$

Uintuk j = n-1 → $c_{n-k-1} = h_0c_{n-1} + h_1 c_{n-2} + \dots + h_{k-1} c_{n-k}$

Pada kode sistem :

$$(c_{n-k}, \dots c_{n-1}) \equiv (d_0 , \dots , d_k)$$

Jawaban :

Kode siklis (7,4) dengan g(x) = 1 + x + x^3 = 1 1 0 1 ←x^3

d(x) = x +x^2 = 0 1 1 0

Cara 1 :

d(x)	b	c
1) 1 0 0 0	(∗) 1 1 0	1 1 0 1 0 0 0
2) 0 1 0 0	(#) 0 1 1	0 1 1 0 1 0 0
3) 0 0 1 0	(Δ) 1 1 1	1 1 1 0 0 1 0
4) 0 0 0 1	(•) 1 0 1	1 0 1 0 0 0 1

$$\Rightarrow G = \begin{bmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0 \\ 1\,1\,1\,0\,0\,1\,0 \\ 1\,0\,1\,0\,0\,0\,1 \end{bmatrix}_{4*7}$$

1) $x^{n-k}d(x)$

$$\begin{array}{r|l} 0\,0\,0\,1\,0\,0\,0 & 1\,1\,0\,1 \\ \hline \vdots\vdots\vdots 0\,0\,0\,0 & 1\,0\,0\,0 \\ \hline \vdots\vdots 0\,1\,0\,0 & \\ \vdots\vdots 0\,0\,0\,0 & \\ \hline \vdots 0\,0\,1\,0 & \\ \vdots 0\,0\,0\,0 & \\ \hline 0\,0\,0\,1 & \\ 1\,1\,0\,1 & \\ \hline 1\,1\,0\,(*) & \end{array}$$

$g(x)$ $b(x)$ $a(x)$

$x^{n-k}d(x) = a(x).g(x) + b(x)$

2)

$$\begin{array}{r|l} 0\,0\,0\,0\,1\,0\,0 & 1\,1\,0\,1 \\ \hline \vdots\vdots\vdots 0\,0\,0\,0 & 0\,1\,0\,0 \\ \hline \vdots\vdots 0\,0\,1\,0 & \\ \vdots\vdots 0\,0\,0\,0 & \\ \hline \vdots 0\,0\,0\,1 & \\ \vdots 1\,1\,0\,1 & \\ \hline 0\,1\,1\,0 & \\ 0\,0\,0\,0 & \\ \hline 0\,1\,1\,(\#) & \end{array}$$

3)

$$\begin{array}{r|l} 0\,0\,0\,0\,0\,1\,0 & 1\,1\,0\,1 \\ \hline \vdots\vdots\vdots 0\,0\,0\,0 & 1\,0\,1\,0 \\ \hline \vdots\vdots 0\,0\,0\,1 & \\ \vdots\vdots 1\,1\,0\,1 & \\ \hline \vdots 0\,1\,1\,0 & \\ \vdots 0\,0\,0\,0 & \\ \hline 0\,0\,1\,1 & \\ 1\,1\,0\,1 & \\ \hline 1\,1\,1\,(\Delta) & \end{array}$$

4)

$$\begin{array}{r|l} 0\,0\,0\,0\,0\,0\,1 & 1\,1\,0\,1 \\ \hline \vdots\vdots\vdots 1\,1\,0\,1 & 1\,1\,0\,1 \\ \hline \vdots\vdots 0\,1\,1\,0 & \\ \vdots\vdots 0\,0\,0\,0 & \\ \hline \vdots 0\,0\,1\,1 & \\ \vdots 1\,1\,0\,1 & \\ \hline 0\,1\,1\,1 & \\ 1\,1\,0\,1 & \\ \hline 0\,1\,1\,(\bullet) & \end{array}$$

Gambar sistem Enkoder, serta hitung isi SR pada setiap langkah.

Cara 2 :

Tinjau rumus:

$$b(x)+x^{n-k}d(x)=a(x)g(x)\triangleq c(x).....(1)$$

$$d(x)= \quad x+x^2 \quad =0110$$

$$g(x)= \ 1+x+x^3 \ =1101$$

$$\left.\begin{array}{l} n=7 \\ k=4 \end{array}\right\} \text{ kode siklis } (7,4)$$

$$x^{n-k}d(x)=x^3d(x)$$
$$=x^3(x+x^2)$$
$$=x^4+x^5=0000110$$

Mencari b(x) {sisa hasil bagi $x^{n-k}d(x)/g(x)$}

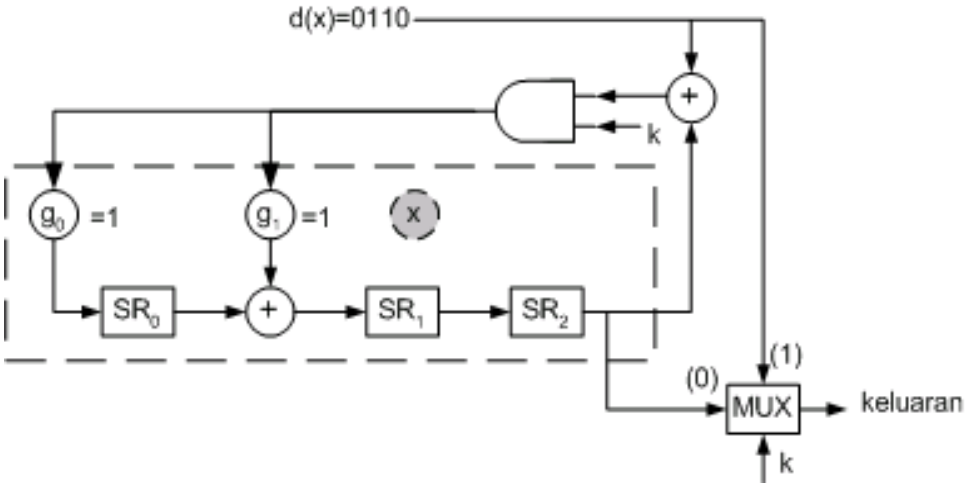
$$\begin{array}{r|l} 1101\overline{)\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ \ } & (\ 1110 \\ \vdots\ \vdots\ \vdots\ 0\ 0\ 0\ 0 & \\ \hline \vdots\ \vdots\ 0\ 0\ 1\ 1 & \\ \vdots\ \vdots\ 1\ 1\ 0\ 1 & \\ \hline \vdots\ 0\ 1\ 1\ 1 & \\ \vdots\ 1\ 1\ 0\ 1 & \\ \hline 0\ 1\ 0\ 1 & \\ 1\ 1\ 0\ 1 & \\ \hline 1\ 0\ 0 & \\ \uparrow & \end{array}$$

$$=b(x)=1000000$$

$$\text{code} \quad d^3=0000110$$
$$\underline{b \ =1000000}$$
$$c=d^3+b=1000110$$

Cara 3 :

Sistem Encoder g(x) = 1101



Tabel SR

Langkah	k	d(x)	SR_0	SR_1	SR_2	Keluaran						
0	0	-	0	0	0	-						
1	1	0	0	0	0	0						
2	1	1	1	1	0	1	0					
3	1	1	1	0	1	1	1	0				
4	1	0	1	0	0	0	1	1	0			
5	0	-	0	1	0	0	0	1	1	0		
6	0	-	0	0	1	0	0	0	1	1	0	
7	0	-	0	0	0	1	0	0	0	1	1	0

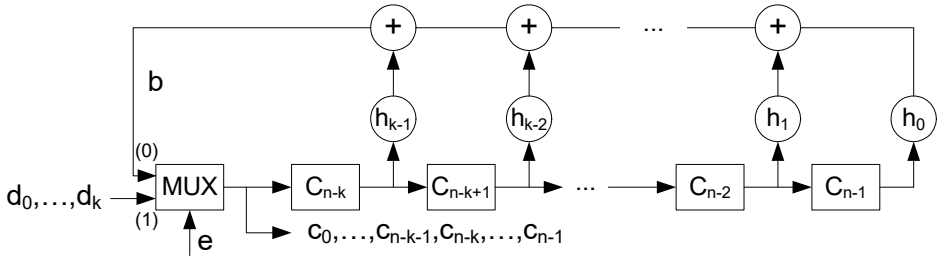
Lanjutan Cara 1 :

$$\Rightarrow \quad G = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$d = (0\,1\,1\,0) \rightarrow c = dG$$

$$= (0\,1\,1\,0) \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= (1\,0\,0\,0\,1\,1\,0)$$



- Enkoder DG h(x)
- Dipilih bila banyak SR < DG g(x)

Langkah:

1. DG l=1 Data: (d_0, ... d_k) dimasukan ke sistem, output & Shift register.
2. Setelah semua data masuk (k-kali Right Shift): l=0
   Kini: $b= h_0C_{n-1} + ... + h_{k-1}C_{n-k} = C_{n-k-1}$
3. SR digeser hingga akhirnya diperoleh C_0 (sebanyak n-k-1 kali)

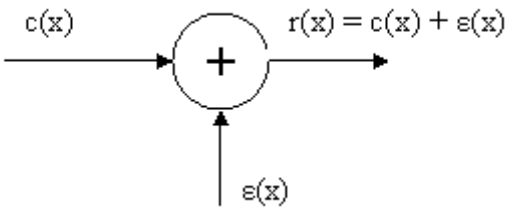
Catatan:

- Sistem digambar untuk kode binari.
- Rangkaian dapat digunakan untuk pengkoda non binari DG:

  $g_j \rightarrow -g_j$

  $h_j \rightarrow -h_j$

SYNDROME



Div. Algorithm:

- r(x) = q(x).g(x) + s(x)
- DEG [s(x)] < DEG [g(x)]

Kasus-1:

$\varepsilon(x) = 0$

$r(x) = c(x) = a(x)g(x)$ atau $S(x) = 0$

Kasus-2:

$\varepsilon(x)$: Suatu Code-Word = $C_1(x)$

$r(x) = C(x) + C_1(x) = C_2(x) = a_2(x)g(x)$ atau $S(x) = 0$

Catatan:

Untuk selanjutnya: Kasus-2 tak diperhatikan (anggap tak ada)

Kasus-3

$\varepsilon(x) = \varepsilon_0 + \varepsilon_1 x + \ldots + \varepsilon_{n-1}x^{n-1} \neq 0$

$r(x) = C(x) + \varepsilon(x)$

$= a(x)g(x) + \{l(x)g(x) + k(x)\}$ .....(**)

$= \{a(x) + l(x)\}g(x) + k(x)$ .....(***)

(*) $\rightarrow$ $= q(x)g(x) + S(x)$

Kesimpulan:

1. $\varepsilon(x) = 0 \rightarrow$ SYN $[r(x)] \stackrel{\Delta}{=} S(x) = 0$
2. $\varepsilon(x) \neq 0 \rightarrow$ SYN $[r(x)]$ = SYN $[\varepsilon(x)] \neq 0$
3. Jika DEG $[\varepsilon(x)]$ < DEG $[g(x)] \rightarrow \varepsilon(x) = 0$

   $\therefore k(x) = S(x) = \varepsilon(x)$

Catatan:

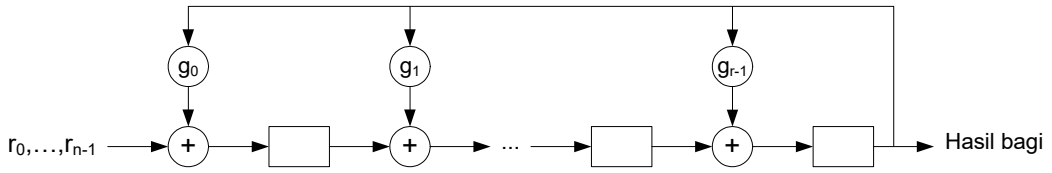
Sindrom S(x) = sisa hasil bagi r(x) oleh g(x)

$\therefore$ Rangkaian pembangkit / penghitung sindrom $\equiv$ Pengkoda (r(x) masuk dari kiri)

Contoh:

Sindrom GEN. untuk kode siklik (7,4)

DG $g(x) = 1 + x + x^3$. r(x) masuk dari kiri



Dan High Order First. Kondisi awal: 00...0 setelah semua data (kode) masuk, isi shift Register = Sindrom

CYCLIC SHIFT

Catatan:

$$DEG[w(x)] \le n \rightarrow x^j w(x) = u(x)(1+x^n) + x^j(x)$$
$$(1+x^n) = g(x)z(x)$$

Pertanyaan:

s(x) adalah sindrom dari r(x) = c(x) + ε(x). Apakah sindrom dari: $r^j(x)$?

Jawab :

Sebut $SYN[r^j(x)] = \delta(x)$

$$\therefore r^j(x) = a(x)g(x) + \delta(x)$$

TETAPI:

$$\begin{aligned} r^j(x) &= v(x)(1+x^n) + x^j r(x) \\ &= v(x)[g(x)z(x)] + x^j[w(x)g(x) + s(x)] \\ &= g(x)[v(x)z(x) + x^j w(x)] + x^j s(x) \\ &= g(x)p(x) + x^j s(x) \end{aligned}$$

JADI: $a(x)g(x) + \gamma(x) = p(x)g(x) + x^j s(x)$

ATAU:

$$x^j s(x) = [a(x) + p(x)]g(x) + \delta(x)$$
$$\delta(x) : \text{sindrom dari } x^j s(x)$$

JUGA:

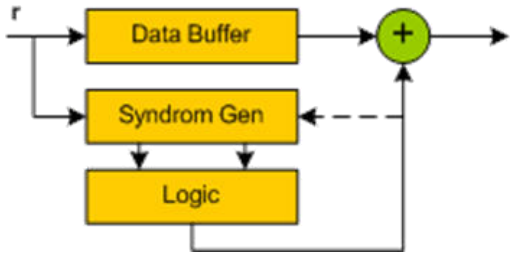
$$x^j s(x) = q(x)[1 + x^n] + s^j(x)$$
$$= q(x)z(x)g(x) + s^j(x)$$

MAKA:

$$s^j(x) = m(x)g(x) + \delta(x)$$
$$\delta(x) : \text{sindrom dari } s^j(x)$$

APLIKASI: SINGLE ERROR CORRECTING

Konsep:



- r(x) digeser siklis $\rightarrow$ posisi bit error
  - Terbawa serta
  - Setelah m langkah terletak di posisi "tertinggi"
- Rangkaian logic didesain mengenali syndrom dari
- Output: '1' jika sesuai, '0' jika tak sesuai $\varepsilon(x) = x^{n-1}$
- Karena SYN $[r^j(x)]$=SYN$[s^j(x)]$, yang perlu digeser siklis hanya isi SYN-GEN.
- Begitu bit error tiba di 'ujung kanan' data buffer, keluaran SYN-GEN dikenali oleh rangkaian logic $\rightarrow$ data di koreksi.
- Keluaran '1' logic CCT, digunakan untuk meng- "clear" SYN-GEN

OPERASI MOD [M(X): IRR.POL]

Teorema:

Set polynomial modulo m(x) adalah suatu ring commutative, dan adalah suatu FIELD, jika dan hanya jika m(x) adalah IRREDUCIBLE.

Contoh:

- $m(x)=1+x+x^2$ over GF(2) $\leftrightarrow$ irreducible
  $A(x)=\{0,1,x,1+x\}$
  Jadi ada sebanyak $4=2^2$ elemen

- $p(x)=2+2x+x^2$ over GF(3) $\leftrightarrow$ irreducible
  $B(x)=\{0,1,2,x,1+x,2+x,2x,1+2x,2+2x\}$
  Jadi ada sebanyak $9=3^2$ elemen

- Akar dari f(x) : irreducible polynomial over GF(p)
  - Tak ada di GF(p)
  - Mungkin ada di field lain {sebut:α} → $f(\alpha)=0$

- Perhatikan :
  $m(x)=1+x+x^2$ over GF(2)
  α satu akar → $m(\alpha)=1+\alpha+\alpha^2=0$
  $$\alpha^2=\alpha+1$$

- Jadi: (over GF(2))

(over GF(2))			A(x)
$\alpha+\alpha=2\alpha$		$=0$	0
α^0		$=1$	1
$\alpha^1=1\alpha$		$=\alpha$	x
$\alpha^2=\alpha+\alpha$		$=\alpha+1$	x+1
$\alpha^3=\alpha^2.\alpha$	$=(\alpha+1).\alpha=\alpha^2+\alpha$ $=(\alpha+1)+\alpha$ $=2\alpha+1=0+1$ $=1$	$=1$	1
$\alpha^4=\alpha^3.\alpha$	$=1.\alpha=\alpha$	$=\alpha$	x
⋮			

- Kesimpulan :

  α : akar dari m(x) membangkitkan semua elemen dari field A(x).

  Sebutan : GF(2) {pangkat: orde dari m(x)}

- Defenisi :

  α : Primitive elemen

  m(x) : Primitive Polinomial

Catatan:

Elemen A(x)/ GF(22) dalam notasi 2 tupple :

- {00,10,01,11}
- {0 ,1 , x,1+x}
- {0 ,1 , α,1+α}

} Low order first

Teorema:

Polinomial 'Irreducible' p(x) dengan DEG.m over GF(2) adalah:

- Faktor dari $x^{2^m-1}+1$
- Polinomial primitive jika DIA faktor dari x^n+1 untuk $n \geq 2^m-1$

Definisi :

- CHARACTERISTIC, dari GF(p): λ terkecil, hingga $\sum_{1=1}^{\lambda} 1 = 0$
- ORDER, n, dari suatu elemen a $\in$ GF(p): integer positif terkecil hingga $a^n = 1$

Beberapa Sifat:

- Jika $\alpha, \beta \in GF(p^m)$ DG. Characteristic p, maka $(\alpha+\beta)^p = \alpha^p + \beta^p$

  Bukti:

  $$(\alpha+\beta)^p = \sum_{i=0}^{p} \binom{p}{i} \alpha^i \beta^{p-1} \; dengan \binom{p}{i} = \frac{p!}{t!(p-i)!} = \frac{p(p-1)!}{i!(p-i)!}$$

  $$(i) \binom{p}{p} = \binom{p}{0} = 1 \quad (sebarang\ p)$$

  $$(ii)\, i\ lain = \binom{p}{i} = \frac{p(p-1)!}{i!(p-i)!} \quad tetapi\ over\ GF(p): p = 0 \rightarrow \binom{p}{i} = 0$$

- $\forall \beta \neq 0 \in GF(p^m)$ berlaku: $\beta^{p^m-1} = 1$

  Jika orde dari β adalah n: n pembagi p^m-1

- Jika $c(x) = c_0 + c_1 x + \ldots + c_n x^n : c_i \in GF(2)$, maka $[c(x)]^{2^\lambda} = c(x^{2^\lambda}) \quad \lambda \neq 0$

  Dengan kata lain :

  Jika: $\beta \in GF(2^m)$ adalah akar dari c(x) over GF(2)

  Maka: $\beta^{2^\lambda}, \lambda \neq 0$, adalah juga akar dari c(x)

  Bukti:

$$c^2(x) = [c_0 + \{c_1 x + \ldots + c_n x^n\}]^2$$
$$= {c_0}^2 + \{c_1 x + \ldots + c_n x^n\}^2 + 2c_0\{c_1 x + \ldots + c_n x_n\}$$
$$\{c_1 x + \ldots + c_n x^n\}^2 = [c_1 x + \{c_2 x^2 + \ldots + c_n x^2\}]^2$$
$$= c_1 x^2 + \{c_2 x^2 + \ldots + c_n x^n\}]^2$$
$$dan\ seterusnya : c^4(x) = [c^2(x)]^2 \rightarrow dst$$

- Ke 2^m-1 elemen tak nol dari GF(2^m) adalah seluruh akar dari $\{x^{2^m-1} + 1\}$

MINIMAL POLINOMIAL

Definisi:

m(x) over GF(2) disebut minimal polinmial dari $\beta \in GF(2^m)$, jika: m(x) adalah monic berpangkat terendah yang memenuhi $m(\beta) = 0$

Teorema:

Min. polinomial dari β adalah Irreducible

Bukti:

Jika m(x) reducible $\rightarrow$ m(x) = $m_1(x) m_2(x)$

DEG[$m_1(x) m_2(x)$]>0

$$\therefore 0 = m(\beta) = m_1(\beta) m_2(\beta)$$

Hanya jika sedikitnya salah satu dari $m_1(x)$ dan $m_2(x)$ = 0

Bertentangan $\rightarrow$ m(x) Irreducible

Teorema:

Andaikan m(x) over GF(2) adalah min. pol. Dari $\beta \in GF(2^m)$ dan f(x) over GF(2) suatu polinom sembarang $\rightarrow$ jika $f(\beta)=0$ maka m(x)|f(x)

Bukti:

fx)=q(x)m(x)+r(x)

$\{m(x): DIFISOR\} \rightarrow DEG[r(x)] < DEG[m(x)]$.....(*)

q(x): hasil bagi

$f(\beta)=0=q(\beta)m(\beta)+r(\beta)$

$m(\beta)=0 \rightarrow r(\beta) harus = 0$.....(**)

(*) dan (**) $\rightarrow$ r(x) harus $\equiv$ 0

m(x) min.pol

IX. KODE BCH

(BOSE, CHAUDURI, HOCQUENGHEM)

- Batasan binari, primitive code
- Cyclic code
- Multiple error correcting

Parameter :(Moreira & Farrell, 2006)

Panjang kode : $n=2^m-1 \quad (m \geq 3)$

Panjang data : k

Panjang pariti : $n-k \leq mt$

Jarak minimum : $d_{min} \geq 2t + 1$

$t < \frac{m}{2}$: Jumlah Error Terkoreksi

Contoh 1 :

- Koreksi 1 error → t=1

$$t < \frac{m}{2} \rightarrow 1 < \frac{m}{2} \rightarrow m = 3$$

$$n = 2^m - 1 \quad (m \geq 3)$$
$$= 2^3 - 1$$
$$n = 7$$

$$n - k \leq m.t$$
$$7 - k \leq 3.1$$
$$7 - 3 \leq k \rightarrow k \geq 4$$

Jadi Kode BCH (n,k)=BCH(7,4)

GENERATOR POLINOMIAL: g(x)

Polinomial over GF(2)

- Berderajat terendah
- Akar: $\alpha, \alpha^2, \alpha^3 .. \alpha^{2t}$

Dengan α: elemen primitive dari GF(2^m)

Jadi: $g(x) = LCM[m_1(x), m_2(x), \ldots m_{2t}(x)]$

Dengan LCM: Least Common Multiple dan $M_i(x)$: Min. Polinomial dari α^i

Catatan:

- β suatu akar $\rightarrow$ β^{2^λ} juga akar
- α^i akar dari $m_i(x)$ $\rightarrow$ $(\alpha^i)^2, (\alpha^i)^i, \ldots$ juga akar dari $m_i(x)$

Contoh 2:

- Single error correcting : BCH(7,3)
- Double error correcting : BCH(31,21)
- Panjang kode : 31

Jawab :

t< $\frac{m}{2}$ → m=5

N= $2^m - 1$ →31=2^m-1 → m=5

α Elemen primitf dari GF(2^5)

TABEL :

p(x)= $1 + x^2 + x^5$: polinominal Primitif over GF(2) dengan degre –5

0 = $1 + \alpha^2 + \alpha^5$ → $\alpha^5 = 1 + \alpha^2$

elemen GF(2^5) : Pembangkit p(x)= $1 + x^2 + x^5$

Elemen Field	5 tupple
$\alpha^{-}=0$	0 0 0 0 0
$\alpha^{31}=1$	1 0 0 0 0
$\alpha^{32}=\alpha^{1}$	0 1 0 0 0
α^{2}	0 0 1 0 0
α^{3}	0 0 0 1 0
α^{4}	0 0 0 0 1
$\alpha^{5}=1+\alpha^{2}$	1 0 1 0 0
$\alpha^{6}=\alpha+\alpha^{3}$	0 1 0 1 0
$\alpha^{7}=\alpha^{2}+\alpha^{4}$	0 0 1 0 1
$\alpha^{8}=1+\alpha^{2}+\alpha^{3}$	1 0 1 1 0
$\alpha^{9}=\alpha+\alpha^{3}+\alpha^{4}$	0 1 0 1 1
$\alpha^{10}=1+\alpha^{4}$	1 0 0 0 1
$\alpha^{11}=1+\alpha+\alpha^{2}$	1 1 1 0 0
$\alpha^{12}=\alpha+\alpha^{2}+\alpha^{3}$	0 1 1 1 0
α^{13}	0 0 1 1 1
α^{14}	1 0 1 1 1

Elemen Field	5 tupple
α^{15}	1 1 1 1 1
α^{16}	1 1 0 1 1
α^{17}	1 1 0 0 1
α^{18}	1 1 0 0 0
α^{19}	0 1 1 0 0
α^{20}	0 0 1 1 0
α^{21}	0 0 0 1 1
α^{22}	1 0 1 0 1
α^{23}	1 1 1 1 0
α^{24}	0 1 1 1 1
α^{25}	1 0 0 1 1
α^{26}	1 1 1 0 1
α^{27}	1 1 0 1 0
α^{28}	0 1 1 0 1
α^{29}	1 0 0 1 0
α^{30}	0 1 0 0 1

DOUBLE ERROR CORRETING: t=2

$g(x) = LCM [m_1(x), m_2(x), m_3(x), m_4(x) \}$

Tetapi :$m_1(x)=m_2(x)=m_4(x)$

$m_2(x) : \alpha^2,\alpha^4,\alpha^8,\alpha^{16},\alpha^{32}= \alpha$

$m_4(x) : \alpha^4,\alpha^8,\alpha^{16},\alpha^{32}= \alpha \rightarrow \alpha^{64}= \alpha^2$

$g(x)=m_1(x).\ m_3(x)$

Mencari / menghitung $m_1(x)$

1. $m_1(x)$: min polinominal dari α^1

Akar dari $m_1(x)$: $\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}, \alpha^{32} = \alpha$

$m_1(x)=(x+\alpha)(x+\alpha^2)(x+\alpha^4)(x+\alpha^8)(x+\alpha^{16})(x+\alpha^{32})(x^2+\alpha^{19}+\alpha^3)(x+\alpha^4)(x+\alpha^8)$
$(x+\alpha^{16})$

Contoh :

$(x+\alpha)(x+\alpha^2)=x^2+(\alpha+\alpha^2)x+\alpha^3$

Dari tabel α= 0 1 0 0 0

GF(2^5) α^2= 0 0 1 0 0 +

$\alpha + \alpha^2$ = 0 1 1 0 0 → α^{19} → tabel

dst

$m_1(x) = 1 + x^2 + x^5$ → BUKTIKAN

$m_1=(x+\alpha)(x+\alpha^2)(x+\alpha^4)(x+\alpha^8)(x+\alpha^{16})(x+\alpha^{32})(x^2+\alpha^{19}+\alpha^3)(x+\alpha^4)(x+\alpha^8)(x+\alpha^{16})$

$= (x^3+\alpha^4x^2+\alpha^{19}x^2+\alpha^{23}x+\alpha^3x+\alpha^7)(x+\alpha^8)(x+\alpha^{16})$

$= [x^3+x^2(\alpha^4+\alpha^{19})+x(\alpha^{23}+\alpha^3)+\alpha^7](x+\alpha^8)(x+\alpha^{16})$

α^4 = 0 0 0 0 1

α^{19}= 0 1 1 0 0 +

= 0 1 1 0 1 = α^{28}

α^3 = 0 0 0 1 0

α^{23}=1 1 1 1 0 +

1 1 1 0 0 = α^{11}

$= (x^3+\alpha^{28}x^2+\alpha^{11}x+\alpha^7)(x+\alpha^8)(x+\alpha^{16})$

$= [x^4+x^3(\alpha^8+\alpha^{28})+x^2(\alpha^{36}+\alpha^{11})+x(\alpha^{19}+\alpha^7)+\alpha^{15}](x+\alpha^{16})$

α^8 = 1 0 1 1 0

α^{28}= 0 1 1 0 1 +

= 1 1 0 1 1 = α^{16}

α^{11} = 1 1 1 0 0

$\alpha^{36}=\alpha^5$=1 1 1 1 0 +

0 1 0 0 0 = α^1

α^8 = 0 0 1 0 1

α^{19}=0 1 1 0 0 +

= 0 1 0 0 1 = α^{30}

$m_1 = (x^4+\alpha^{16}x^3+\alpha x^2+x\alpha^{30}+\alpha^{15})(x+\alpha^{16})$

$= x^5+(\alpha^{16}+\alpha^{16})x^4+x^3(\alpha^{32}+\alpha)+x^2(\alpha^{17}+\alpha^{30})+x(\alpha^{46}+\alpha^{15})+\alpha^{31}$

α^{16} = 1 1 0 1 1

α^{16} = 1 1 0 1 1 +

= 0 0 0 0 0 = $\alpha^{-\infty}$ = 0

α = 0 1 0 0 0

$\alpha^{32}=\alpha$ = 0 1 0 0 0 +

0 0 0 0 0 = $\alpha^{-\infty}$ = 0

α^{17} = 1 1 0 01

α^{30} = 0 1 0 0 1 +

= 1 0 0 0 0 = α^0 = 1

α^{15} = 1 1 1 1 1

$\alpha^{32}=\alpha$ = 1 1 1 1 1 +

0 0 0 0 0 = $\alpha^{-\infty}$ = 0

$m_1(X) = x^5 + x^2 + 1$

Elemen Field		5 tupple
α^0	$=1$	1 0 0 0 0
α^1	$=\alpha$	0 1 0 0 0
α^2	$=\alpha^2$	0 0 1 0 0
α^3	$=\alpha^3$	0 0 0 1 0
α^4	$=\alpha^4$	0 0 0 0 1
α^5	$=1+\alpha^2$	1 0 1 0 0
α^6	$=\alpha+\alpha^3$	0 1 0 1 0
α^7	$=\alpha^2+\alpha^4$	0 0 1 0 1
α^8	$=\alpha^3+\alpha^5$	1 0 1 1 0
α^9	$=\alpha+\alpha^3+\alpha^4$	0 1 0 1 1
α^{10}	$=1+\alpha^4$	1 0 0 0 1
α^{11}	$=1+\alpha+\alpha^2$	1 1 1 0 0
α^{12}	$=\alpha+\alpha^2+\alpha^3$	0 1 1 1 0
α^{13}	$=\alpha^2+\alpha^3+\alpha^4$	0 0 1 1 1
α^{14}	$=1+\alpha^2+\alpha^3+\alpha^4$	1 0 1 1 1
α^{15}	$=1+\alpha+\alpha^2+\alpha^3+\alpha^4$	1 1 1 1 1
α^{16}	$=1+\alpha+\alpha^3+\alpha^4$	1 1 0 1 1

Elemen Field		5 tupple
α^{17}	$=1+\alpha+\alpha^4$	1 1 0 0 1
α^{18}	$=1+\alpha$	1 1 0 0 0
α^{19}	$=\alpha+\alpha^2$	0 1 1 0 0
α^{20}	$=\alpha^2+\alpha^3$	0 0 1 1 0
α^{21}	$=\alpha^3+\alpha^4$	0 0 0 1 1
α^{22}	$=1+\alpha^2+\alpha^4$	1 0 1 0 1
α^{23}	$=1+\alpha+\alpha^2+\alpha^3$	1 1 1 1 0
α^{24}	$=\alpha+\alpha^2+\alpha^3+\alpha^4$	0 1 1 1 1
α^{25}	$=1+\alpha^3+\alpha^4$	1 0 0 1 1
α^{26}	$=1+\alpha+\alpha^2+\alpha^4$	1 1 1 0 1
α^{27}	$=1+\alpha+\alpha^3$	1 1 0 1 0
α^{28}	$=\alpha+\alpha^2+\alpha^4$	0 1 1 0 1
α^{29}	$=1+\alpha^3$	1 0 0 1 0
α^{30}	$=\alpha+\alpha^4$	0 1 0 0 1
$\alpha^{31}=\alpha^0$	$=1$	1 0 0 0 0
$\alpha^{32}=\alpha^1$	$=\alpha$	0 1 0 0 0

2. $m_3(x)$ = min Pol dari α^3

Akar : α^3, $(\alpha^3)^2 = \alpha^6$, $(\alpha^3)^4 = \alpha^{12}$, α^{24} , α^{48}, $\alpha^{34} = \alpha^3$

$m_3(x) = (x+\alpha^3)(x+\alpha^6)(x+\alpha^{12})(x+\alpha^{24})(x+\alpha^{19})$

$= 1+x^2+x^3+x^4+x^5$

jadi : $g(x) = (1+x^2+x^5)(1+x^2+x^3+x^4+x^5)$

$= 1+x^3+x^5+x^6+x^8+x^9+x^{10}$

catatan :

g(x) : untuk keperluan praktis, ditabelkan dalam kode / notasi oktal

(i) : tulis g(x) dalam bentuk binari high order first { $\frac{11}{3}\frac{101}{5}\frac{101}{5}\frac{001}{1}$ }

(ii) : ubah kode binari → notasi oktal

contoh :

Kode contol diatas :

n=31, DEG[g(x)] = 10, panjang data = 21

Kode yang diperoleh : BCH (31,21)

maka : G[31,21] = 3551

PARITY CHECK MATRIX : H

Catatan :

- t-error correcting code → g(x)=LCM[$m_1(x)$, $m_2(x)$,…,$m_{2t}(x)$]
- c(x)=suatu koda=d(x).g(x) $\therefore c(\alpha i)=0$, i=1,2,…,2t
- β suatu akar → $\beta^{2\lambda}$ juga suatu akar $\therefore c(\alpha i)=0 \lozenge c\{(\alpha i)2\lambda\}=0$

Tulis :

$c(x)=c_0+c_1x+c_2x^2+\ldots+c_{n-1}x^{n-1}$

$\therefore$ $c(\alpha i)=c0+c1\alpha i+c2(\alpha i)2+\ldots+cn-1(\alpha i)n-1=0$

Ditulis dalam bentuk notasi matriks :

$$[c_0\ c_1 \dots c_{n-1}]\begin{bmatrix}(\alpha^i)^0\\(\alpha^i)^1\\\vdots\\(\alpha^i)^{n-1}\end{bmatrix}=0 \quad \text{atau} \quad [c_0\ c_1 \dots c_{n-1}]\begin{bmatrix}(\alpha^1)^0 & (\alpha^2)^0 & \dots & (\alpha^{2t})^0\\(\alpha^1)^1 & (\alpha^2)^1 & \dots & (\alpha^{2t})^1\\\vdots\\(\alpha^1)^{n-1} & (\alpha^2)^{n-1} & \dots & (\alpha^{2t})^{n-1}\end{bmatrix}=0$$

$1 \le i \le 2t$

$$C.H^T = 0$$

t=error correcting ➔tak ada sebanyak (2t atau lebih kecil) baris dari H^T yang berjumlah nol.

Jadi :

$$H=\begin{bmatrix}1 & (\alpha^1)^1 & (\alpha^1)^2 \dots & (\alpha^1)^{n-1}\\1 & (\alpha^2)^1 & (\alpha^2)^2 \dots & (\alpha^2)^{n-1}\\1 & (\alpha^3)^1 & (\alpha^3)^2 \dots & (\alpha^3)^{n-1}\\\vdots\\1 & (\alpha^{2t})^1 & (\alpha^{2t})^2 \dots & (\alpha^{2t})^{n-1}\end{bmatrix}=0$$

Catatan :

Tiap elemen dari H (dan HT) adalah satu m-tuple [lihat contoh elemen GF(25)]

$$S \triangleq r.H^T$$

$$[s_0\ s_1 \dots s_{2t-1}]=[r_0\ r_1 \dots r_{n-1}]\begin{bmatrix}(\alpha^1)^0 & (\alpha^2)^0 & \dots & (\alpha^{2t})^0\\(\alpha^1)^1 & (\alpha^2)^1 & \dots & (\alpha^{2t})^1\\\vdots\\(\alpha^1)^{n-1} & (\alpha^2)^{n-1} & \dots & (\alpha^{2t})^{n-1}\end{bmatrix}$$

$$s_{k-1} = r_0(\alpha^k)^0 + r_1(\alpha^k)^1 + \dots + r_{n-1}(\alpha^k)^{n-1}$$

atau

$$= r(x)\,|\,x=\alpha^k$$

$$= c(x)\,|\,x=\alpha^k + \varepsilon(x)\,|\,x=\alpha^k$$

$$= \cancel{c(\alpha^k)}^{=0} + \varepsilon(\alpha^k)$$

$$s_{k-1} = \varepsilon(\alpha^k) = 0 \quad \textbf{jika, } \varepsilon(x) = 0$$

$$(1 \le k \le 2t)$$

Asumsi :

ada sebanyak $v(1 \le v \le t)$ salah

error terjadi dan tidak diketahui lokasinya $j_1, j_2..j_v$

atau : $\varepsilon(x) = \sum_{\lambda=1}^{v} x^{j\lambda}$ ($j\lambda$: posisi bit salah)

$(0 \le j\lambda \le n\text{-}1)$

$$\therefore \varepsilon(\alpha^k) = \sum_{\lambda=1}^{v} (\alpha^k)^{j\lambda} = \sum_{\lambda=1}^{v} (\alpha^{j\lambda})^k = s_{k-1}$$

jadi :

$$s_0 = \alpha^{j1} + \alpha^{j2} + \ldots + \alpha^{jv}$$
$$s_1 = (\alpha^{j1})^2 + (\alpha^{j2})^2 + \ldots + (\alpha^{jv})^2$$
$$\vdots$$
$$s_{2t-1} = (\alpha^{j1})^{2t} + (\alpha^{j2})^{2t} + \ldots + (\alpha^{jv})^{2t}$$

Bahasan :

- r : diketahui → S_k dapat dihitung
- Ada sebanyak : 2t persamaan untuk α^{ji} (error location number)
  - Pada dasarnya : α^{ji} dapat dihitung
  - Dari α^{ji} dihasilkan ji (diharapkan)

Tulis :

$$r(x) = q(x).m_k(x) + y_k(x) \quad m_k(x) \text{ pol. min dari } \alpha^k$$
$$\therefore r(x) = q(\alpha^k).m_k(\alpha^k) + y_k(\alpha^k)$$
$$c(\alpha^k) + \varepsilon(\alpha^k) = y_k(\alpha^k) = s_{k-1}$$

α : elemen dari $GF(2^m)$ diperlukan cara khusus (ump. look up tabel untuk menghitung ji dari α^{ji})

Contoh :

BCH Code (31,21), mampu mengkoreksi 2 error, hitung syndrome dari :

r(x)=x3+x11 dengan :

- Perhitungan matriks
- Algoritma pembagian

Jawab :

n=31 k=21 t=2

- S = r . HT

  1x4 1x31 31x4
- g(x)=LCM($m_1(x)$,$m_2(x)$,$m_3(x)$,$m_4(x)$)
  - $g(x)=m_1(x).m_3(x)$
  - $r(x)=q(x).m_1(x)+y_1(x)$
  - $r(x)=q(x).m_3(x)+y_3(x)$
  - $S_{k-1}=y_k(\alpha^k)$

ERROR LOCATOR POLYNOMIAL : $\sigma(x)$

{ $\beta_\lambda \underline{\underline{\Delta}} \alpha^{J\lambda}$: ERROR LOCATION NUMBER }

Definisi:

$$\sigma(x) \underline{\underline{\Delta}} (1+\beta_1 x)(1+\beta_2 x) \quad \dots \quad (1+\beta_v x)$$
$$= \sigma_0 + \sigma_1 x + \sigma_2 x^2 + \quad \cdots \quad + \sigma_v x^v$$

DG :

$$\sigma_0 = 1$$
$$\sigma_1 = \beta_1 + \beta_2 + \quad \cdots \quad + \beta_v$$
$$\sigma_2 = \beta_1\beta_2 + \beta_2\beta_3 + \quad \cdots \quad + \beta_{v-1}\beta_v$$
$$\vdots$$
$$\sigma v = \beta_1\beta_2 \quad \cdots \quad \beta_v$$

{ ELEMENTARY SYMMETRIC FCT }

Bahasan :

Jika $\sigma(x)$ dapat disusun :

- Akar dari $\sigma(x)$ adalah $(\beta_k)^{-1} = \alpha^{-Jk}$
- Maka α^{-Jk} dan Jk dapat dihitung (Jk: posisi bit salah)

Khusus

$t \le 2$

Umpama : $\varepsilon(x) = x^u + x^v$

$$\therefore \quad S_0 = \varepsilon(\alpha') = \alpha^u + \alpha^v = \beta_1 + \beta_2$$
$$S_2 = \varepsilon(\alpha^3) = \alpha^{3u} + \alpha^{3v} = \beta_1^{\,3} + \beta_2^{\,3}$$

Tetapi:

$$\begin{aligned}\sigma(x) &= (1+\beta_1 x)(1+\beta_2 x)\\ &= 1+(\beta_1+\beta_2)x+\beta_1\beta_2 x^2\\ &= 1+S_0 x+?\,x^2\end{aligned}$$

Catat :

* $\beta_1^{\,3}+\beta_2^{\,3}=(\beta_1+\beta_2)(\beta_1^{\,2}+\beta_1\beta_2+\beta_2^{\,2})$

* $\beta_1^{\,2}+\beta_2^{\,2}=(\beta_1+\beta_2)^2=S_0^{\,2}$

$\therefore S_2 = S_0(\beta_1\beta_2+S_0^{\,2})$

Jika $S_0 \neq 0 \rightarrow \beta_1\beta_2 = S_0^{\,2}+\dfrac{S_2}{S_0}$ (asalkan $S_2 \neq S_0^{\,3}$)

Maka :

$$\sigma(x)=1+S_0 x+\left(S_0^{\,2}+\frac{S_2}{S_0}\right)x^2$$

Contoh :

DEC (31, 21) BCH ; $\varepsilon(x)=x^9+x^{15}$

{ $m_1(x) = 1 + x^2 + x^5$; $m_3(x) = 1 + x^2 + x^3 + x^4 + x^5$ }

Diperoleh :

$$\begin{aligned}S_0 &= 1+\alpha^2=\alpha^5\\ S_2 &= \alpha^6+\alpha^9+\alpha^{12}=\alpha^{28}\\ \therefore \sigma(x) &= 1+\alpha^5 x+\left(\alpha^{10}+\frac{\alpha^{28}}{\alpha^5}\right)x^2\\ &= 1+\alpha^5 x+\alpha^{24}x^2\\ &= (1+\beta_1 x)(1+\beta_2 x)\end{aligned}$$

$$\left.\begin{aligned}\therefore \beta_1 &= \alpha^{15}\\ \beta_2 &= \alpha^9\end{aligned}\right\} \; try \;\&\; error$$

{ error di bit 10 dan bit 16 }

Maka :

$$e(x)=x^9+x^{15}$$

X. KODE CONVOLUTIONAL

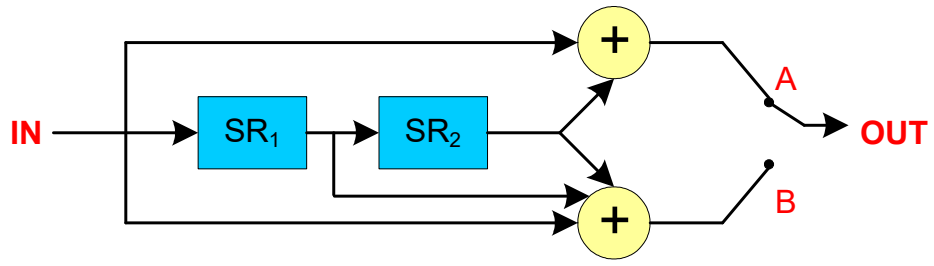
Sifat Convolutional Code(Moreira & Farrell, 2006) Ditulis dengan 3 parameter (n,k,m) :

- n = jumlah bit codeword/ output enkoder/ banyaknya modulo 2 adder
- k = jumlah bit input enkoder
- m = jumlah memory register/ shift register

- Contoh penulisan convolutional code (2,1,3)
- Ada 3 cara mendesain operasi encoder pada convolutional codes :
  - State diagram
  - Trellis diagram
  - Tree diagram (tidak dibahas)

STATE DIAGRAM

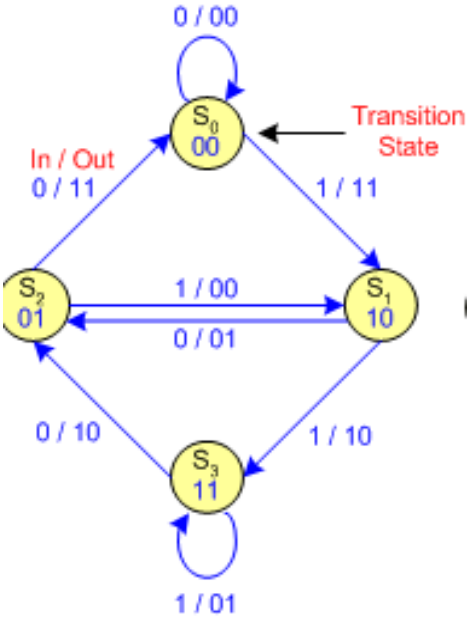
Diagram yang menampakkan perubahan isi memori (state) karena data masukan.



Tabel dari STATE

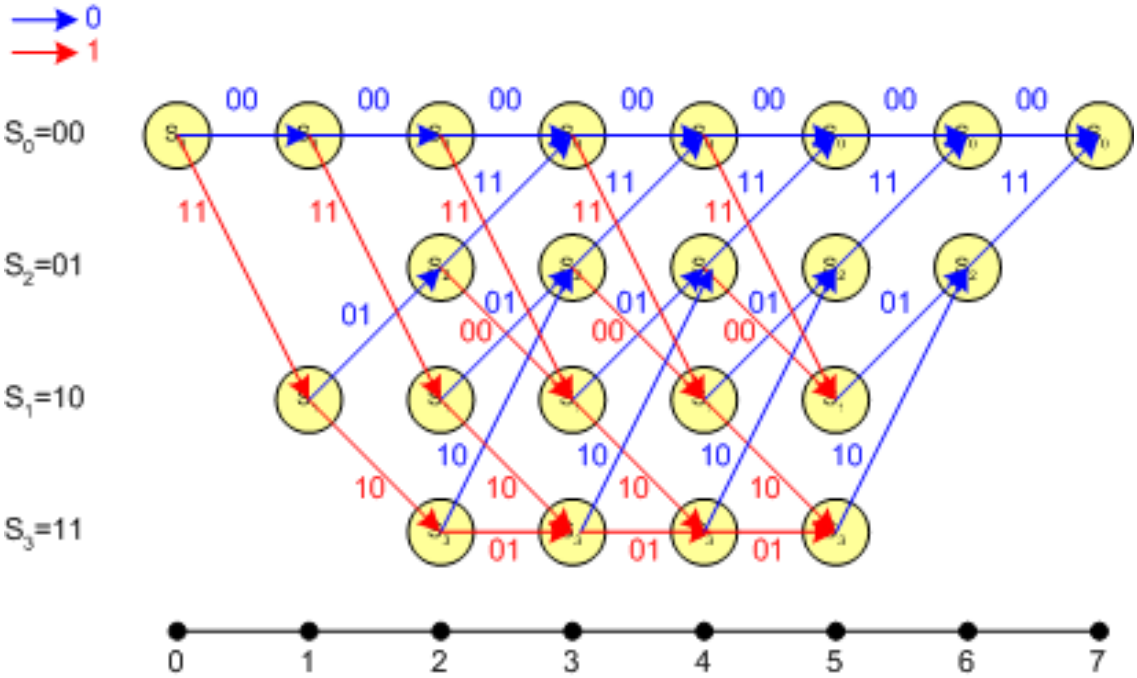
State Sekarang	Input			
	0		1	
	Out	Transition State	Out	Transition State
0 0	0 0	0 0	1 1	1 0
0 1	1 1	0 0	0 0	1 0
1 1	1 0	0 1	0 1	1 1
1 0	0 1	0 1	1 0	1 1

STATE – Diagram



TRELLIS DIAGRAM

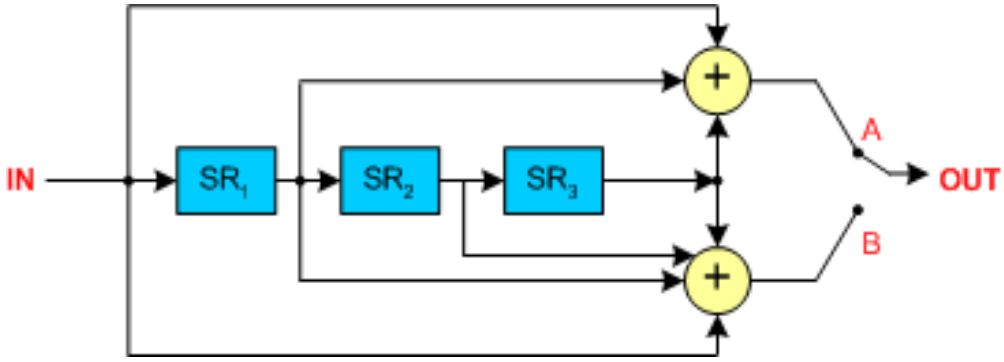
State diagram yang dibentangkan sehingga memperlihatkan perjalanan state dari enkoder sebagai fungsi waktu.



Input : d = (101100)

c = (11, 01, 00, 10, 10, 11)

r = (11, 10, 01,10,10,11)

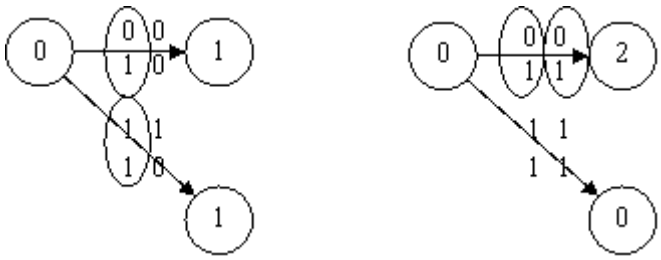


ALGORITMA VITERBI

1. Titik awal S_0 : beri nilai <u>nol</u>
2. Ambil set r

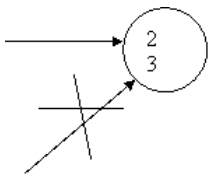
   Bandingkan dengan set disemua lintasan keluar dari titik awal.

   Beri nilai yang sama dengan beda set lintasan dengan set yang diterima.
3. nilai node yang kini dicapai = nilai node asal + nilai lintasan



4. Ambil set r berikutnya. Ulang proses 2 dan 3.

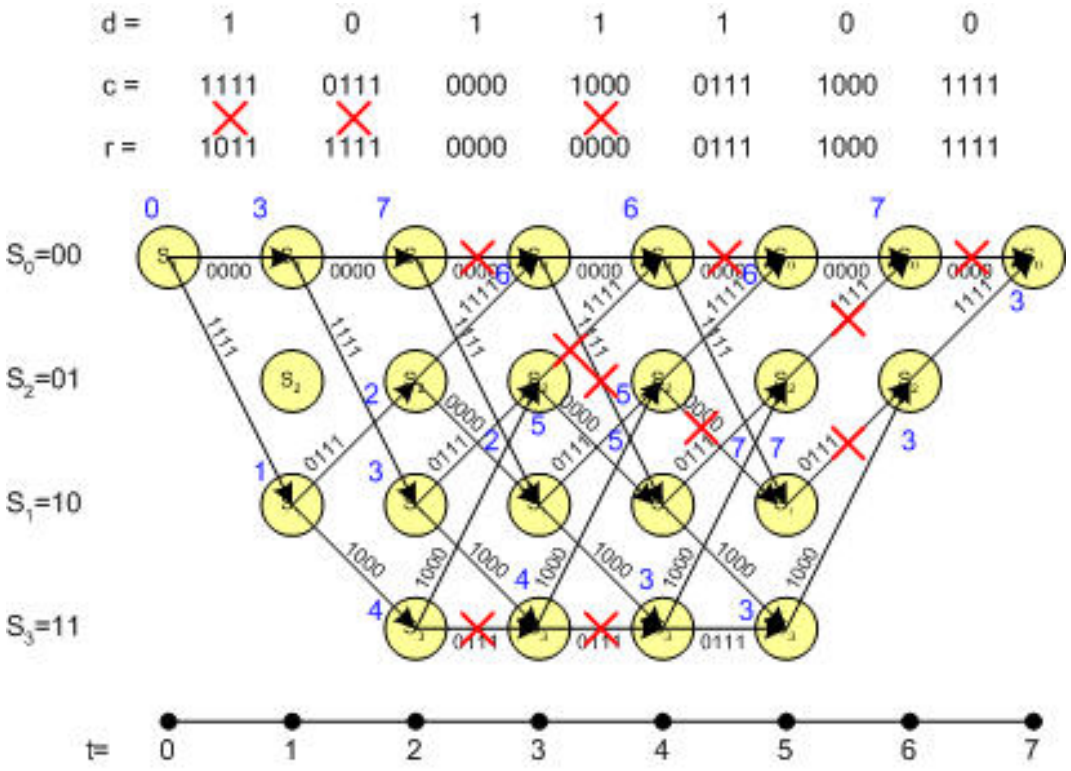
   Jika satu node dicapai oleh lebih dari satu arah, sisakan lintasan yang menghasilkan node minimum.

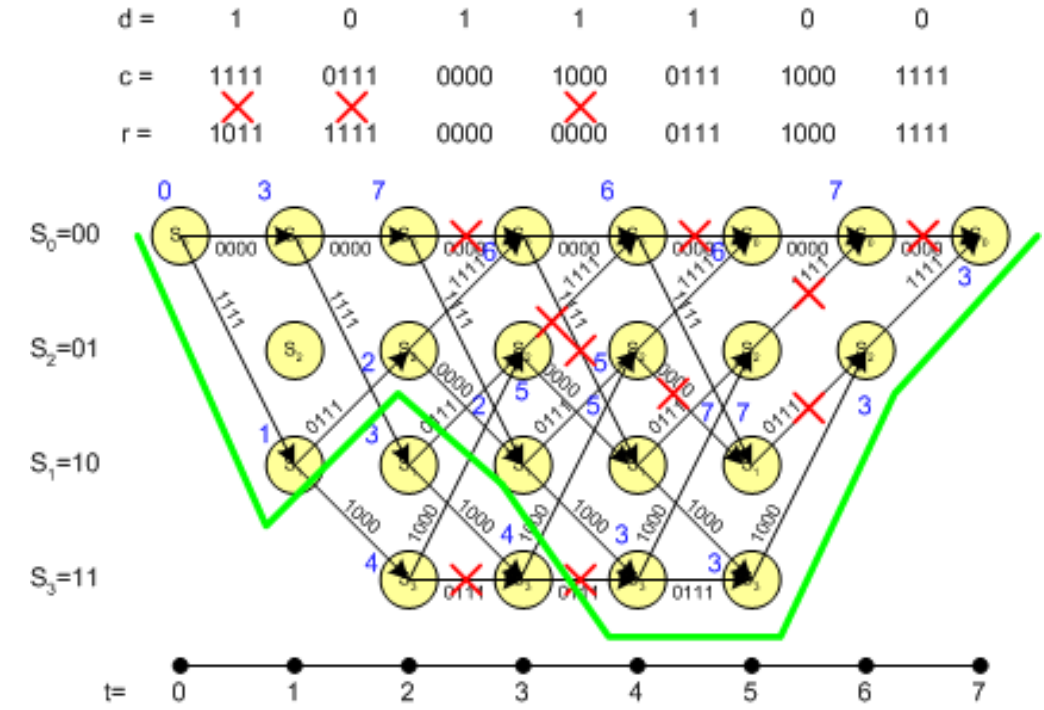


5. Selesai
6. Lintasan yang benar adalah yang nilai nodenya minimum.

Contoh:

- d = (1011100)
- c=(1111 0111 0000 1000 0111 1000 1111)
- r=(1011 1111 0000 0000 0111 1000 1111)





DAFTAR PUSTAKA

Judson, T. W., & Austin, S. F. (2009). *Abstract Algebra Theory and Applications*.

Lint, J. H. Van. (1973). *Lecture Notes in Mathematics Coding Theory*. Springer-Verlag.

Mackay, D. J. C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Moon, T. K. (2005). *Error Correction Coding Mathematical Methods and Algorithms*. A John Wiley & Sons, Inc.

Moreira, J. C., & Farrell, P. G. (2006). *Essentials of Error-Control Coding*. John Wiley & Sons.

Spence, S. A. (2008). *Introduction to Algebraic Coding Theory*.

Wicker, S. B. (1995). *Error Control Systems for Digital Communication and Storage*. Prentice-Hall.