

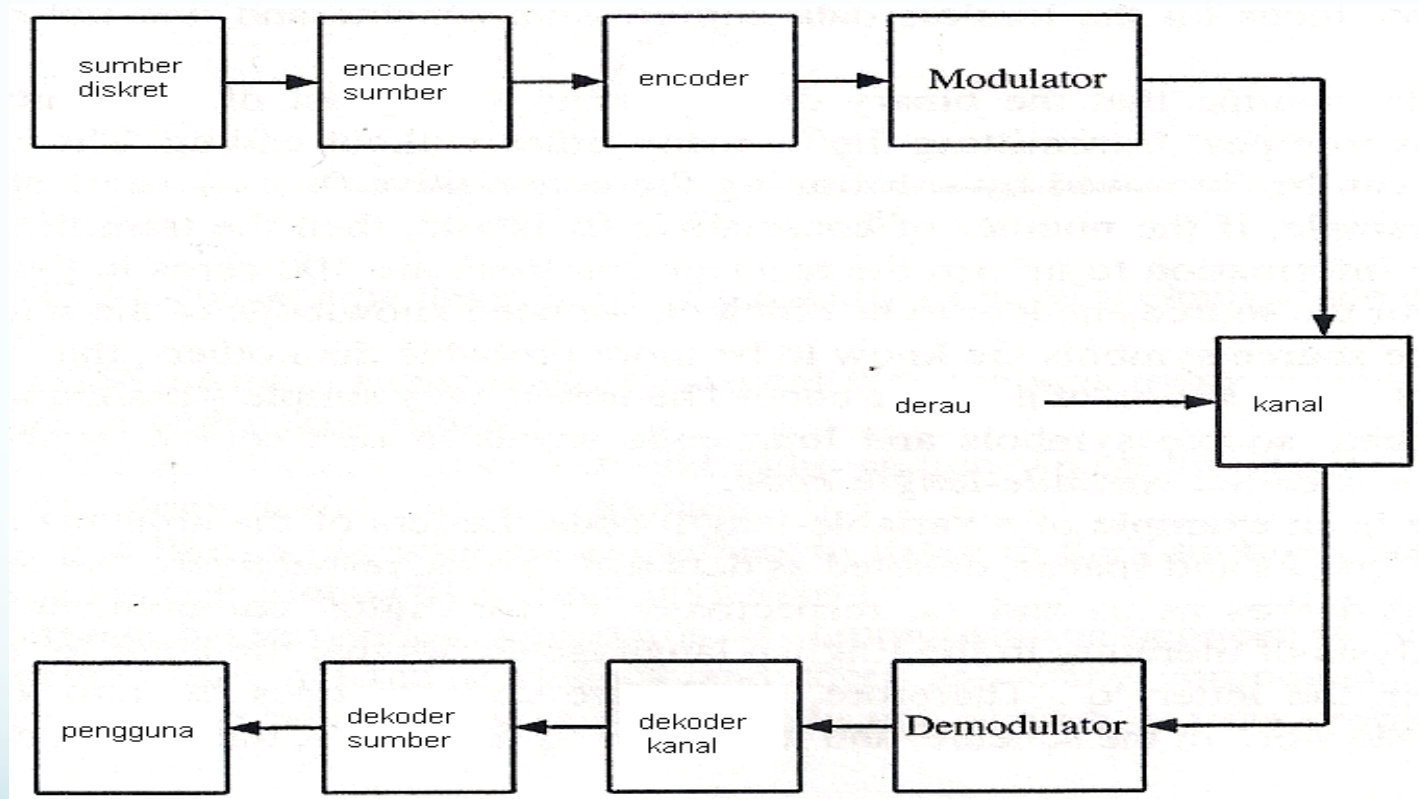
PENYANDIAN SUMBER DAN PENYANDIAN KANAL

Risanuri Hidayat

Penyandian sumber

- Penyandian yang dilakukan oleh sumber informasi.
- Isyarat dikirim/diterima kadang-kadang/sering dikirimkan dengan sumber daya yang tidak efisien. Penyandian sumber dimaksudkan untuk mengatasi hal ini.
- terdapat dua skema:
 1. Operasi tanpa kehilangan informasi, disebut sebagai *data compaction* atau kompresi data tanpa rugi-rugi, dimana data asli dapat direkonstruksi dari data tersandinya tanpa kehilangan informasi.
 2. Operasi dengan hilangnya informasi, disebut sebagai kompresi data, dimana data yang direkonstruksi mungkin memiliki perbedaan dengan data aslinya.

- Gambar dari suatu sistem komunikasi digital yang menggabungkan antara penyandian sumber dan penyandian kanal



PANJANG KATA SANDI RATA-RATA DARI PENYANDIAN SUMBER

- Bit informasi sumber pertama-tama dikelompokkan ke dalam blok dengan panjang N , hasilnya dalam semua 2^N kemungkinan kata-kata biner yang dinyatakan sebagai $A^{(N)}$.
- Sebagai contoh, jika $N=2$ dan kita memiliki semua kemungkinan kata-kata biner dari $A^{(2)}=\{00,01,10,11\}$ dengan himpunan statistic $\{p_0^2, p_0p_1, p_1p_0, p_1^2\}$. Kita bisa menuliskan kata-kata biner $s_0=\{00\}$, $s_1=\{01\}$, $s_2=\{10\}$, dan $s_3=\{11\}$.
- Probabilitas terjadinya element $A^{(2)}$ sering tidak sama Misal $\{ 9/16, 3/16, 3/16, 1/16 \}$.
- Dengan penyandian sumber, kita bisa rancang sebuah himpunan kata sandi dengan panjang yang berbeda untuk menyatakan elemen dari $A^{(2)}$

- Panjang kata sandi dari penyandi sumber didefinisikan sebagai berikut:

$$\bar{L} = \sum_{k=0}^{2^N-1} p_k l_k$$

- p_k = probabilitas
- l_k = panjang penyandian
- Panjang bit informasi rata-rata adalah

$$\bar{L}_b = \frac{1}{N} \sum_{k=0}^{2^N-1} p_k l_k = \frac{\bar{L}}{N}$$

- $\mathcal{L}$ = jumlah bit

Simbol Sumber	Peluang kemunculan	Sandi I	Sandi II	Sandi III
$s_0 = (00)$	9/16	0	1	0
$s_1 = (01)$	3/16	00	10	10
$s_2 = (10)$	3/16	1	100	110
$s_3 = (11)$	1/16	11	1000	111

- jika tidak ada penyandian sumber, digit dari sumber secara langsung ditransmisikan tanpa proses apapun. panjang sandi rata-rata adalah sebagai berikut

$$\bar{L} = 2 \times \left(\frac{9}{16}\right) + 2 \times \left(\frac{3}{16}\right) + 2 \times \left(\frac{3}{16}\right) + 2 \times \left(\frac{1}{16}\right) = 2$$

- Panjang bit informasi rata-rata dapat dihitung sebagai berikut (N=2)

$$\bar{L}_b = \frac{\bar{L}}{N} = \frac{2}{2} = 1$$

- Jika menggunakan data yaitu '00 10 11 01 00 00', disandikan dengan sandi III maka menghasilkan rangkaian '0 110 111 10 00

Simbol Sumber	Peluang kemunculan	Sandi III
$s_0 = (00)$	9/16	0
$s_1 = (01)$	3/16	10
$s_2 = (10)$	3/16	110
$s_3 = (11)$	1/16	111

● 00 10 11 01 00 00

↓ ↓ ↓ ↓ ↓ ↓

0 110 111 10 0 0

- Panjang kode sandi rata-rata dari sandi III dapat dihitung sebagai berikut

$$\bar{L} = 1 \times \left(\frac{9}{16}\right) + 2 \times \left(\frac{3}{16}\right) + 3 \times \left(\frac{3}{16}\right) + 3 \times \left(\frac{1}{16}\right) = \frac{27}{16} = 1.6875$$


0 10 110 111

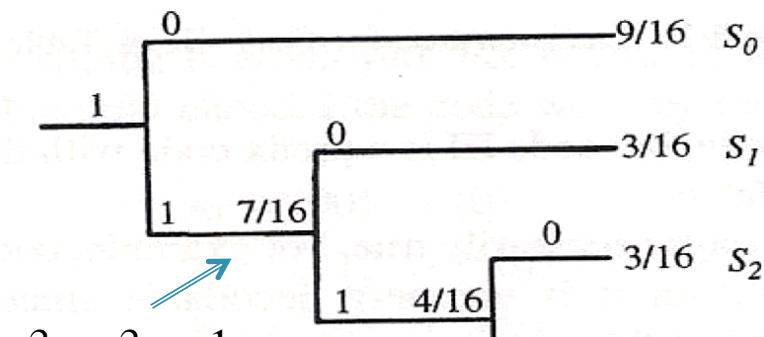
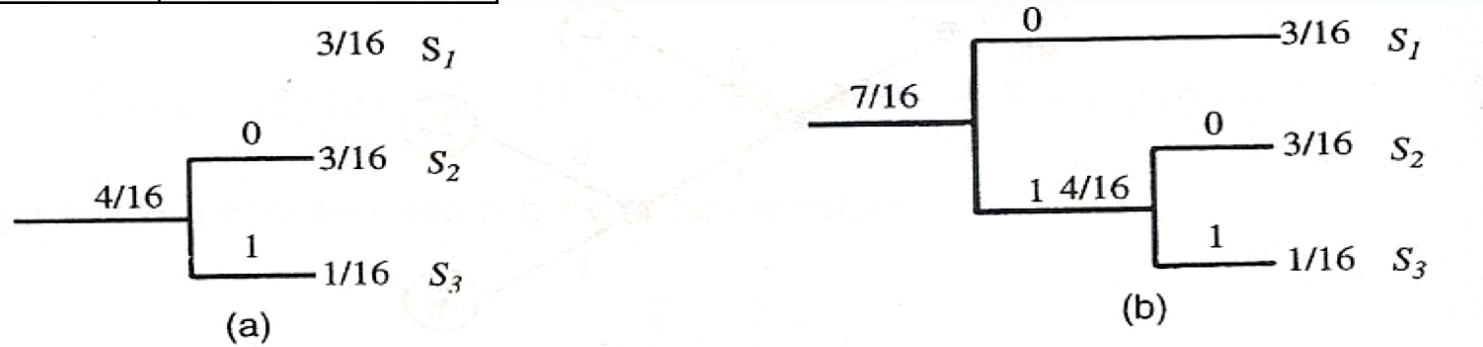
- Panjang rata-rata per bit informasi untuk sandi III dapat dihitung sebagai berikut (N=2)

$$\overline{L_b} = \frac{\bar{L}}{N} = 0.8438$$

Kode Huffman

- Sandi Huffman berprinsip memendekkan sandi untuk pesan-pesan dengan peluang muncul besar (sering muncul), memanjangkan sandi untuk pesan yang peluang munculnya kecil (jarang muncul).
- Gagasan dari proses kode Huffman sebagai berikut:
 1. Simbol dua sumber dari peluang terendah disimbolkan sebagai 0 dan 1.
 2. Dua simbol sumber peluang terendah kemudian dikombinasi ke dalam simbol sumber baru yang kemungkinan sama dengan jumlah dari peluang dua simbol asli. Bilangan dari simbol sumber berkurang oleh kombinasi ini.

Simbol Sumber	Peluang kemunculan
$s_0 = (00)$	$9/16$
$s_1 = (01)$	$3/16$
$s_2 = (10)$	$3/16$
$s_3 = (11)$	$1/16$



$$\frac{3}{16} + \frac{3}{16} + \frac{1}{16}$$

- Sebagai contoh, susunan dari empat simbol kode Huffman dengan peluang sumber $9/16, 3/16, 3/16$ dan $1/16$ digambarkan pada Gambar 8.3.
- Tahap pertama ditunjukkan pada 8-3(a), ada dua peluang terendah dengan peluang $3/16$ dan $1/16$ dikombinasikan sehingga keluar peluang baru $3/16 + 1/16 = 4/16$.
- Dua simbol peluang terendah disimbolkan dengan 0 dan 1, berturut-turut. mengurangi susunan pohon simbol kode Huffman dengan peluang sumber $9/16, 3/16$ dan $4/16$. ambil dua peluang terendah $3/16$ dan $4/16$.
- Gambar 8-3(b) di kombinasikan kembali dari peluang $3/16$ dan $4/16$ dan hasilnya dalam simbol baru $7/16$.

- Sekarang tertinggal dua simbol dengan peluang $9/16$ dan $7/16$.
- Tahap akhir adalah untuk mengkombinasi dua simbol terakhir dan disimbolkan 0 dan 1, berturut-turut, untuk mereka.
- Sekarang, kami mempunyai susunan sebuah pohon. *Code word* untuk setiap simbol sumber dapat dibaca dari simbol paling kiri ke kanan mengikuti pohon.
- Sebagai contoh, untuk simbol s_3 , kata sandi seharusnya 111 dibaca dari kiri ke kanan. Untuk simbol s_2 , *code word* seharusnya 110, dan seterusnya.
- Susunan kode persisnya kode sama sebagai kode III pada Tabel 8.1. Dengan menggunakan persamaan (8.1), panjang rata-rata *code word* adalah $L=1.6875$. Panjang rata-rata tiap bit informasi dapat dihitung sebagai $L_b=L/2=0.8438$. Hasil ini menunjukkan efisiensi dari bit sumber dapat di tingkatkan secara signifikan dengan kode Huffman.

Penyandian Kanal

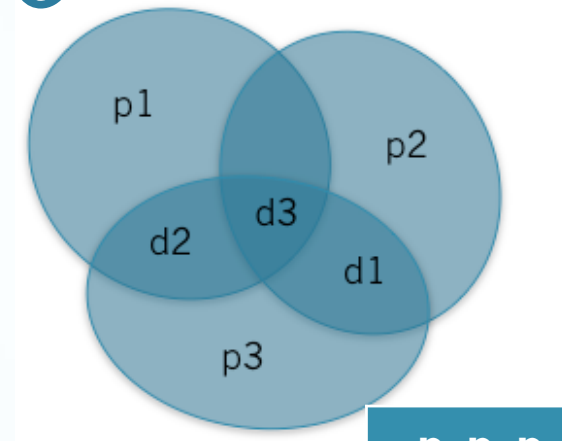
- Penyandian kanal dipakai untuk keamanan informasi yang dikirimkan.
- Sandi kanal berfungsi menambah redundansi untuk meningkatkan keandalan. Sandi kanal disebut juga sebagai *error correcting code (CRC)*
- Sandi Kanal dibagi menjadi 2 klas:
 - *block codes* dan
 - *convolutional codes*.
- *Convolutional codes* memiliki memori pada penyandi sedangkan pada *block codes* tidak mempunyai memori.

Block Codes: Kode Hamming

- Panjang n dan jumlah bit informasi k dari suatu kode Hamming mempunyai bentuk $n = 2^m - 1$ dan $k = 2^m - 1 - m$ untuk integer positif $m \geq 3$.
- Kode Hamming mempunyai kemampuan mengkoreksi satu bit *error* dalam n bit yang terkodekan

Example

- $k = 3, P = 3, n = ..6$
- Code $(n,k) = \text{Code}(6,3)$
- Coret salah satu baris dari P
- Tambahkan matriks Identitas $k \times k$ (3x3) di depannya



p_1	p_2	p_3
0	1	1
1	0	1
1	1	0
1	1	1

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$P_1 = D_2 + D_3$$

$$P_2 = D_1 + D_3$$

$$P_3 = D_1 + D_2 + D_3$$

H Generation

- Tambahkan matriks Identitas $P \times P$ (3x3) di bawahnya

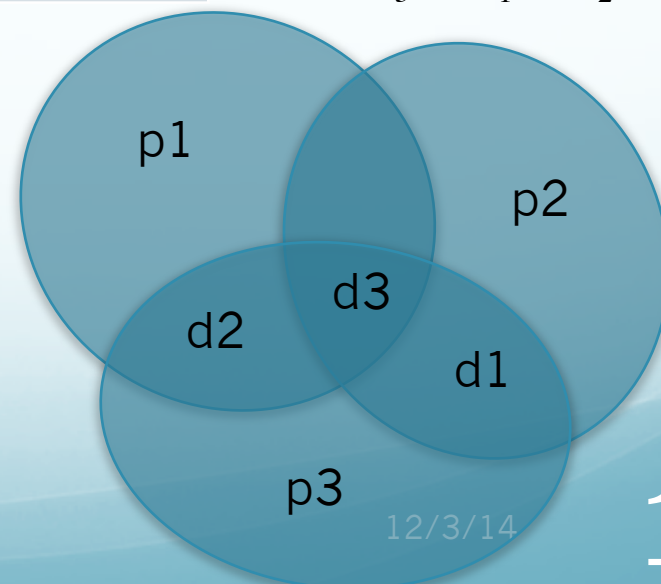
$$H = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$p_1 p_2 p_3$
0 1 1
1 0 1
1 1 1

$$P_1 = D_2 + D_3$$

$$P_2 = D_1 + D_3$$

$$P_3 = D_1 + D_2 + D_3$$

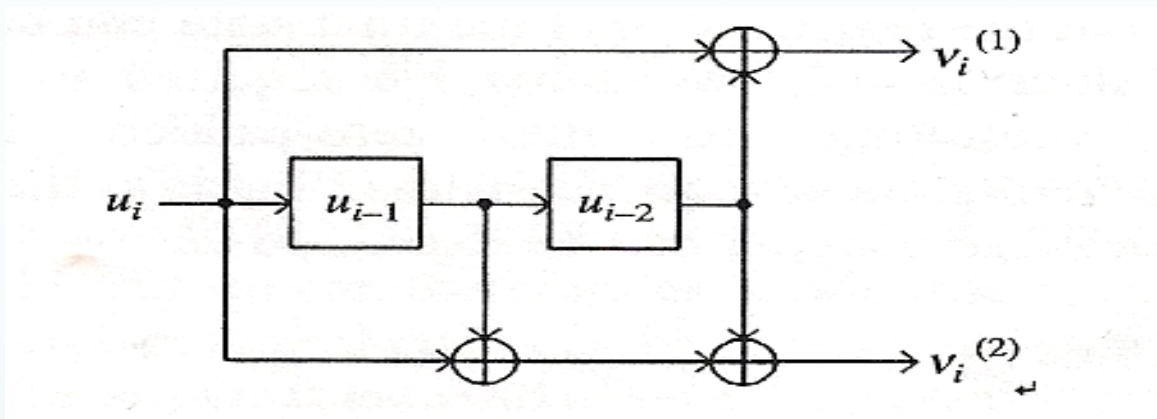


Sisi Penerima

- Hence, for any received word r without errors,
 - $c = r$
 - $r.H = \mathbf{0}$
- Now suppose a received word r has some errors in it.
 - $r = c + e$
- c is some valid codeword and e is an *error vector*,
 - $r.H = (c+e).H = 0 + e.H$

Sandi Konvolusional

- Sandi konvolusional berbeda dengan sandi blok yang di dalam encodernya dilengkapi *memory*
- sandi konvolusional disajikan dalam gambar



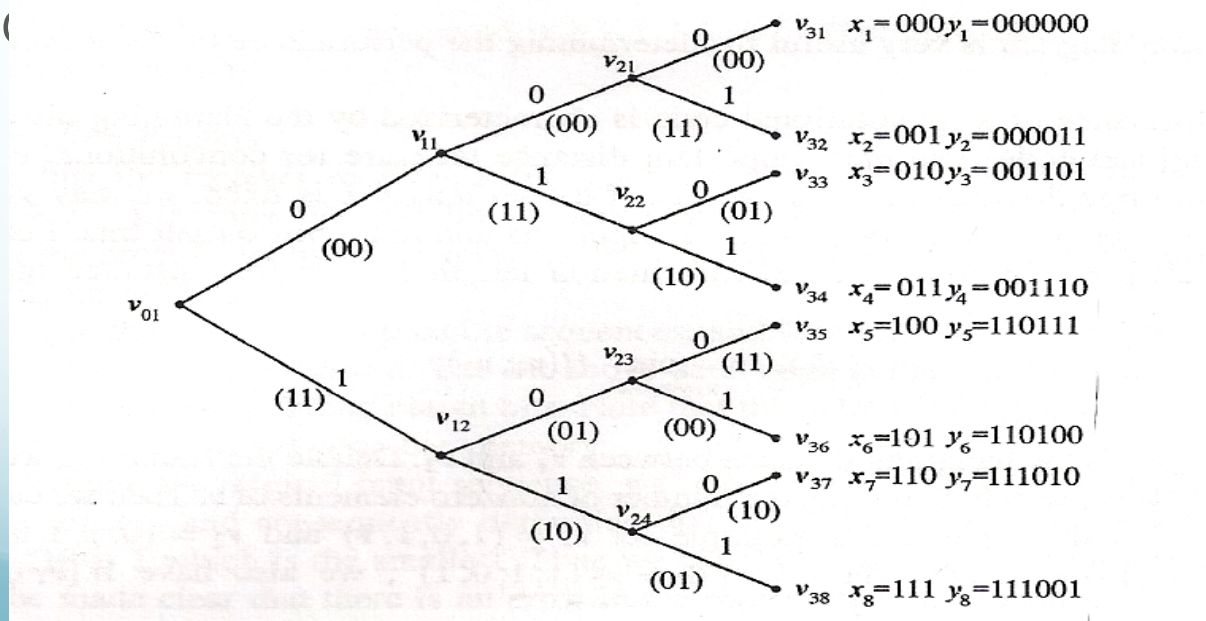
- Dimana notasi $\oplus$ menunjukkan operator logika eksklusif OR (XOR). Karena encoder merupakan sebuah rangkaian sekuensial maka merupakan mesin keadaan berhingga (FSM=Finite State Machine)

$$v_i^{(1)} = u_i \oplus u_{i-2}$$

$$v_i^{(2)} = u_i \oplus u_{i-1} \oplus u_{i-2}$$

Pohon sandi konvolusional

- masing-masing bit di atas tepi merupakan bit input dan dua bit di bawah tepi merupakan bit-bit keluaran. Catatan bit keluaran tidak berhubungan dengan bit masukan secara tepat, hal ini merupakan karakteristik khusus dari sandi konv



- dengan $L=5$, maka $(u_1, u_2, u_3, u_4, u_5) = (1,0,1,1,1)$. Rangkaian ditambahi zero menjadi $(u_1, u_2, u_3, u_4, u_5, 0, 0) = (1,0,1,1,1,0,0)$. persamaan dari gambar 8-11 dengan keadaan awal $(0,0)$, kita dapatkan rangkaian tersandi sebagai $(1,1,0,1,0,0,1,0,0,1,1,0,1,1)$.

