LINEAR BLOCK CODE

TTI3J3 SISTEM KOMUNIKASI II

TUJUAN PEMBELAJARAN

- Mahasiswa mampu memahami Linear Block Code dengan menentukan codeword, matriks parity check, syndrome, deteksi dan koreksi error serta menggambarkan rangkaian encoder dan rangkaian syndrome

CHANNEL CODING

- Channel coding berhubungan dengan teknik pengontrol error.
- Channel coding pada dasarnya membolehkan adanya peningkatan rate transmisi informasi pada rate error yang tetap atau
- Mengurangi rate error pada rate transmisi yang tetap.
- Contoh:
  - **Linear Block Code**
  - **Cyclic Codes**
  - BCH (The Bose Chaudhuri & Hocquenghem) Codes
  - **Convolutional Codes**
  - Turbo Codes
  - LDPC

2 METODE ERROR CONTROL

- **Automatic Repeat Request (ARQ)**,
  - ketika penerima mendeteksi adanya error pada data yang diterima, maka penerima akan meminta untuk retransmit
- **Forward Error Correction (FEC)**
  - Data yang ditransmisikan telah dikodekan sedemikian sehingga bila terdeteksi adanya error pada data akan dapat dikoreksi.

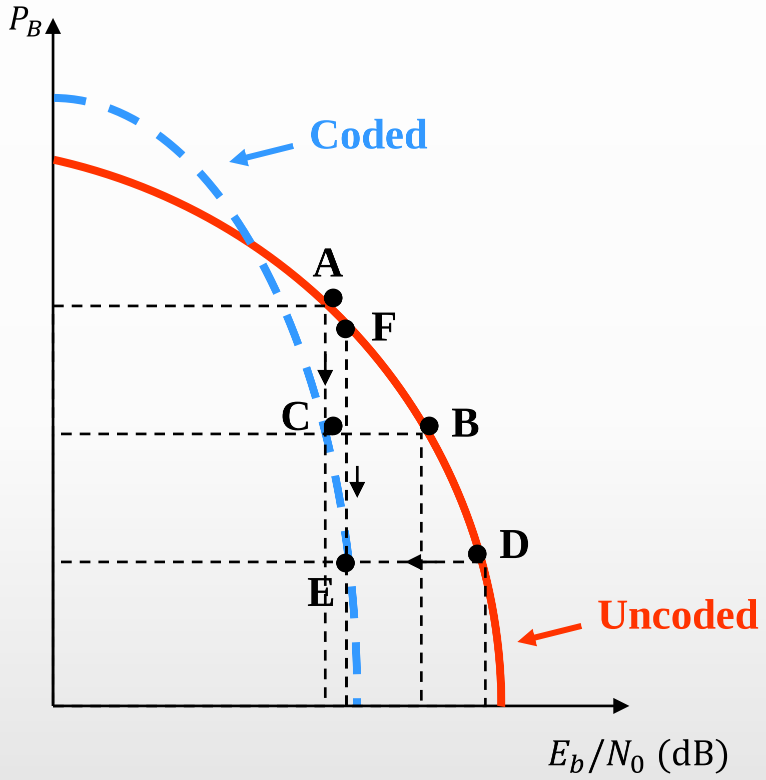
2 TIPE CHANNEL CODING

- **Block codes** merupakan tipe pengkodean yang mengkodekan data yang terdiri dari blok k bit informasi yang akan dikodekan menjadi n bit codeword (n > k)
  - Contoh: LBC, cycic code,LDPC

- **Convolutional Codes** dimana n bit codeword yang dihasilkan tidak hanya tergantung dari k bit data saat ini tapi tergantung pula dari bit data sebelumnya

DAMPAK PENGGUNAAN CHANNEL CODING

- Kinerja vs bandwidth
- Power vs. bandwidth
- Data rate vs. bandwidth
- Capacity vs. bandwidth



Coding gain:

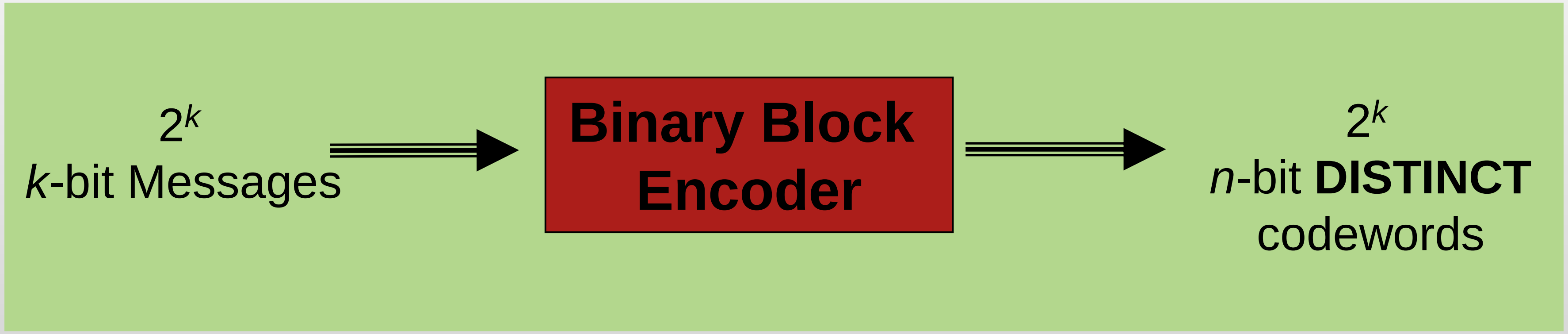
Reduksi Eb/N0 jika menggunakan skema channel coding untuk mencapai kinerja tertentu

$$G\ [\mathrm{dB}] = \left(\frac{E_b}{N_0}\right)_{\mathrm{u}} [\mathrm{dB}] - \left(\frac{E_b}{N_0}\right)_{\mathrm{c}} [\mathrm{dB}]$$

DEFINISI

Linear Block Codes

- Rangkaian bit informasi disegmentasi menjadi blok-blok message dengan **ukuran yang tetap.**
- Setiap *k*-bit *information **message*** dikodekan menjadi *n*-bit ***codeword*** (*n*>*k*)



DEFINISI

Linear Block Codes

- Penjumlahan Modulo-2 dari 2 buah ***codeword*** akan menghasilkan ***codeword***
- Setiap codeword **v** yang termasuk dalam block code **C** merupakan kombinasi linear dari *k* buah codeword yang bebas linear di dalam block code **C**, contoh:

$$U = m_0 . g_0 + m_1 . g_1 + \ldots + m_{k-1} . g_{k-1}$$

$$g_i = [g_{i0}\ g_{i1}\ \ldots.\ g_{i,n-1}]$$

BEBERAPA DEFINISI PENTING

- **Binary field** :
  - The set {0,1}, under modulo 2 binary addition and multiplication forms a field.

Addition	Multiplication
$0 \oplus 0 = 0$	$0 \cdot 0 = 0$
$0 \oplus 1 = 1$	$0 \cdot 1 = 0$
$1 \oplus 0 = 1$	$1 \cdot 0 = 0$
$1 \oplus 1 = 0$	$1 \cdot 1 = 1$

  - Binary field is also called Galois field, GF(2).

BEBERAPA DEFINISI

- **Fields** :
  - Let F be a set of objects on which two operations '+' and '.' are defined.
  - F is said to be a field if and only if
    1. F forms a commutative group under + operation. The additive identity element is labeled "0".

       $$\forall a, b \in F \Rightarrow a + b = b + a \in F$$

    2. F-{0} forms a commutative group under . Operation. The multiplicative identity element is labeled "1".

       $$\forall a, b \in F \Rightarrow a \cdot b = b \cdot a \in F$$

    3. The operations "+" and "." distribute:

       $$a \cdot (b + c) = (a \cdot b) + (a \cdot c)$$

BEBERAPA DEFINISI

- Vector space:
  - Let V be a set of **vectors** and F a fields of elements called **scalars**. V forms a vector space over F if:
    1. Commutative:

       $$\forall \mathbf{u},\mathbf{v} \in V \Rightarrow \mathbf{u}+\mathbf{v} = \mathbf{v}+\mathbf{u} \in F$$

       $$\forall a \in F, \forall \mathbf{v} \in \mathbf{V} \Rightarrow a \cdot \mathbf{v} = \mathbf{u} \in \mathbf{V}$$
    2. Distibutive

       $$(a+b)\cdot\mathbf{v} = a\cdot\mathbf{v} + b\cdot\mathbf{v} \ \text{ and } \ a\cdot(\mathbf{u}+\mathbf{v}) = a\cdot\mathbf{u} + a\cdot\mathbf{v}$$
    3. Associative

       $$\forall a,b \in F, \forall \mathbf{v} \in V \Rightarrow (a\cdot b)\cdot\mathbf{v} = a\cdot(b\cdot\mathbf{v})$$

       $$\forall \mathbf{v} \in \mathbf{V}, \qquad 1\cdot\mathbf{v} = \mathbf{v}$$

BEBERAPA DEFINISI

- Examples of vector spaces
  - The set of binary n-tuples, denoted by V_n

$$V_4 = \{(0000), (0001), (0010), (0011), (0100), (0101), (0111), (1000), (1001), (1010), (1011), (1100), (1101), (1111)\}$$

- Vector subspace:
  - A subset S of the vector space V_n is called a subspace if:
    - The all-zero vector is in S.
    - The sum of any two vectors in S is also in S.

    Example:

$$\{(0000), (0101), (1010), (1111)\} \text{ is a subspace of } V_4.$$

BEBERAPA DEFINISI

- Spanning set:
  - A collection of vectors $\boldsymbol{G} = \{\boldsymbol{g_1}, \boldsymbol{g_2}, \ldots, \boldsymbol{g_n}\}\{\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_n}\}$ the linear combinations of which include all vectors in a vector space V, is said to be a <u>spanning set</u> for V or to <u>span</u> V.
    - Example:

      $\{(1000), (0110), (1100), (0011), (1001)\}$ spans V_4.

- Bases:
  - A spanning set for V that has minimal cardinality is called a basis for V.
    - Cardinality of a set is the number of objects in the set.
    - Example:

      $\{(1000), (0100), (0010), (0001)\}$ is a basis for V_4.

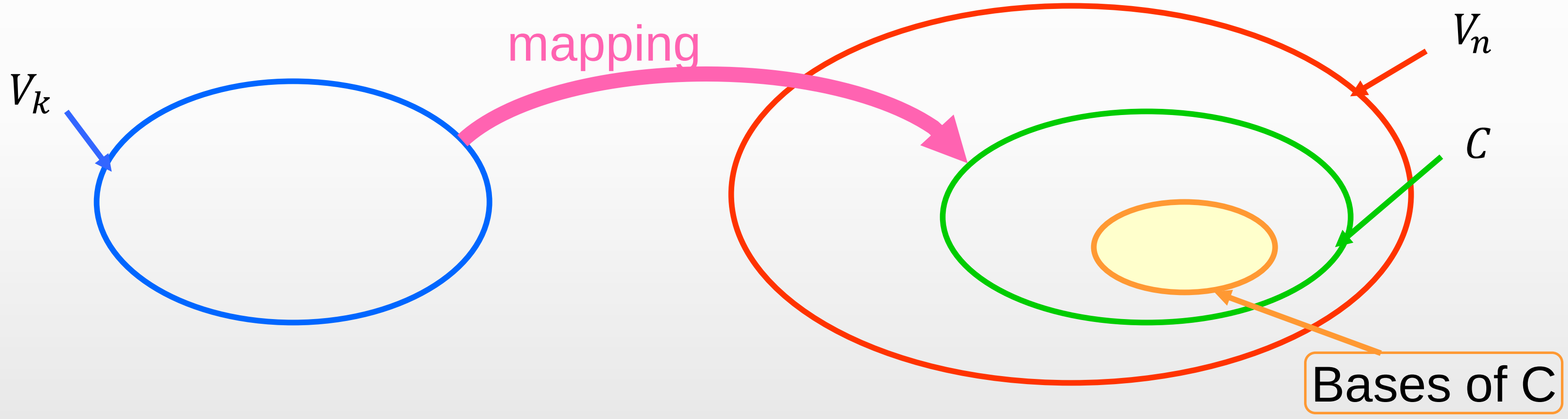
LINEAR BLOCK CODE

- Linear block code (n,k)
  - A set $\boldsymbol{C} \subset \boldsymbol{V_n}$ with cardinality $\boldsymbol{2^k}$ is called a linear block code if, and only if, it is a subspace of the vector space $\boldsymbol{V_n}$.

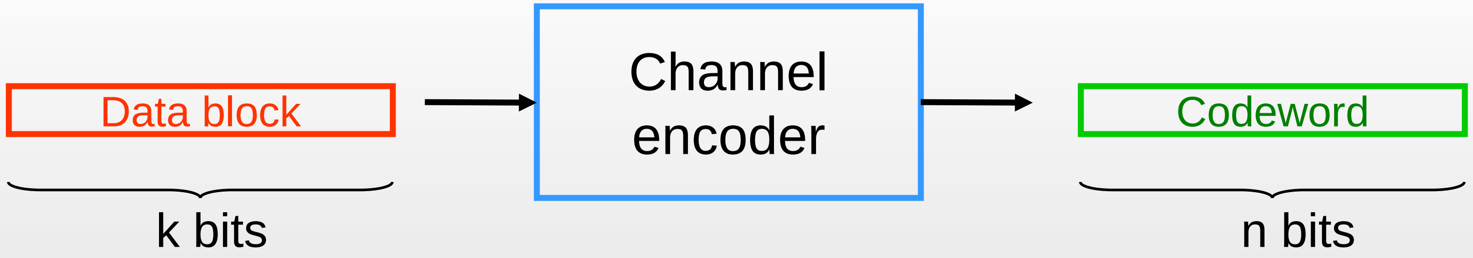
$$\boxed{\boldsymbol{V_k} \rightarrow \boldsymbol{C} \subset \boldsymbol{V_n}}$$

    - Members of C are called code-words.
    - The all-zero codeword is a codeword.
    - Any linear combination of code-words is a codeword.

LINEAR BLOCK CODE



- Bit-bit informasi dipotong-potong menjadi blok yang terdiri dari **k** bits.
- Setiap blok message di encode menjadi blok kde dengn ukuran yang lebih besar **n** bits.
- Bit-bit yang telah dikodekan dimodulasi dan ditransmisikan melalui kanal
- Reverse procedure dilakukan di sisi receiver.



$n - k$ Redundant bits

$R_c = \dfrac{k}{n}$ Code rate

- Bobot Hamming (Hamming weight) dari vektor **U**, dinyatakan dengan w(**U**), jumlah elemen non-zero pada **U**.
- Jarak Hamming atau (Hamming distance) antara 2 vector **U** dan **V**, adalah jumlah elemen yang berbeda pada keduanya.

$$d(\mathbf{U},\mathbf{V}) = w(\mathbf{U} \oplus \mathbf{V})$$

- Minimum distance dari sebuah block code adalah

$$d \min_{i \neq j} {}_{i} \quad {}_{j} \min_{i} {}_{i} \quad {}_{\min}$$

- Kemampuan deteksi error :

$$e = d_{\min}$$

- Kemampuan error koreksi **t** dari sebuah kode, didefinisikan sebagai jumlah bit maksimum yang dijaminkan untuk dapat dikoreksi pada setiap codewordnya

$$t = \left\lfloor \frac{d_{\min}}{2} \right\rfloor$$

LINEAR BLOCK CODE

$$\mathbf{G} = \begin{bmatrix} \mathbf{V}_1 \\ \vdots \\ \mathbf{V}_k \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & & \ddots & \vdots \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{bmatrix}$$

- EncodingProses pengkodean pada (n,k) block code

$$\boxed{\mathbf{U} = \mathbf{mG}}$$

$$(u_1, u_2, \ldots, u_n) = (m_1, m_2, \ldots, m_k) \cdot \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_k \end{bmatrix}$$

$$(u_1, u_2, \ldots, u_n) = m_1 \cdot \mathbf{V}_1 + m_2 \cdot \mathbf{V}_2 + \cdots + m_2 \cdot \mathbf{V}_k$$

- Nilai baris pada G, merupakan *linearly independent*.

CONTOH : **LBC (7,4)**

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \mathbf{g}_2 \\ \mathbf{g}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

m= [0 1 1 0]

$\Downarrow$

Linear Block Encoder (**U=m.G**)

$\Downarrow$

U= $\mathbf{g}_1 + \mathbf{g}_2$

U= [1 0 0 0 1 1 0]

Message (m)	Codeword	
0000	0000000	
0001	**1010001**	$\mathbf{g}_3$
0010	**1110010**	$\mathbf{g}_2$
0011	0100011	
0100	**0110100**	$\mathbf{g}_1$
0101	1100101	
0110	**1000110**	
0111	0010111	
1000	**1101000**	$\mathbf{g}_0$
1001	0111001	
1010	0011010	
1011	1001011	
1100	1011100	
1101	0001101	
1110	0101110	
1111	1111111	

$$\mathbf{G}=\begin{bmatrix}\mathbf{g}_0\\ \mathbf{g}_1\\ \mathbf{g}_2\\ \mathbf{g}_3\end{bmatrix}=\begin{bmatrix}1&1&0&1&0&0&0\\0&0&1&0&1&1&1\\1&1&1&1&1&1&1\\1&0&1&0&0&0&1\end{bmatrix}$$

Linearly Dependent

m= [0 1 1 1] → Block Encoder (**U=m.G**) → **U**= $\mathbf{g}_1+\mathbf{g}_2+\mathbf{g}_3$, **U**= [0 1 1 1 0 0 1]

m= [1 0 0 1] → Block Encoder (**U=m.G**) → **U**= $\mathbf{g}_0+\mathbf{g}_3$, **U**= [0 1 1 1 0 0 1]

NOT DISTINCT

LINEAR SYSTEMATIC BLOCK CODE

n-k bits	*k* bits
Redundant Checking Part	Message Part

p-matrix | *kxk- identity matrix*

$$\mathbf{G}=[\mathbf{P}\ \mathbf{I}_k]=\left[\begin{array}{cccc|cccc} p_{00} & p_{01} & \ldots & p_{0,n-k-1} & 1 & 0 & \ldots & 0 \\ p_{10} & p_{11} & \ldots & p_{1,n-k-1} & 0 & 1 & \ldots & 0 \\ . & . & & . & . & . & & . \\ . & . & & . & . & . & & . \\ . & . & & . & . & . & & . \\ p_{k-1,0} & p_{k-1,1} & \ldots & p_{k-1,n-k-1} & 0 & 0 & \ldots & 1 \end{array}\right]$$

PARITY CHECK MATRIX

- Untuk matriks **G (***k x n) dengan k baris yang bersifat* linearly independent, maka terdapat matriks **H** (Parity Check Matrix) (*n*-*k*) x *n* , dimana:
  - **G.H^T=0**

$$\mathbf{H}=\left[\mathbf{I}_k \ \mathbf{P}^T\right]=\begin{bmatrix} 1 & 0 & \dots & 0 & p_{00} & p_{01} & \dots & p_{k-1,0} \\ 0 & 1 & \dots & 0 & p_{01} & p_{11} & \dots & p_{k-1,1} \\ . & . & & . & . & . & & . \\ . & . & & . & . & . & & . \\ . & . & & . & . & . & & . \\ 0 & 0 & \dots & 1 & p_{0,n-k-1} & p_{1,n-k-1} & \dots & p_{k-1,n-k-1} \end{bmatrix}$$

CONTOH:

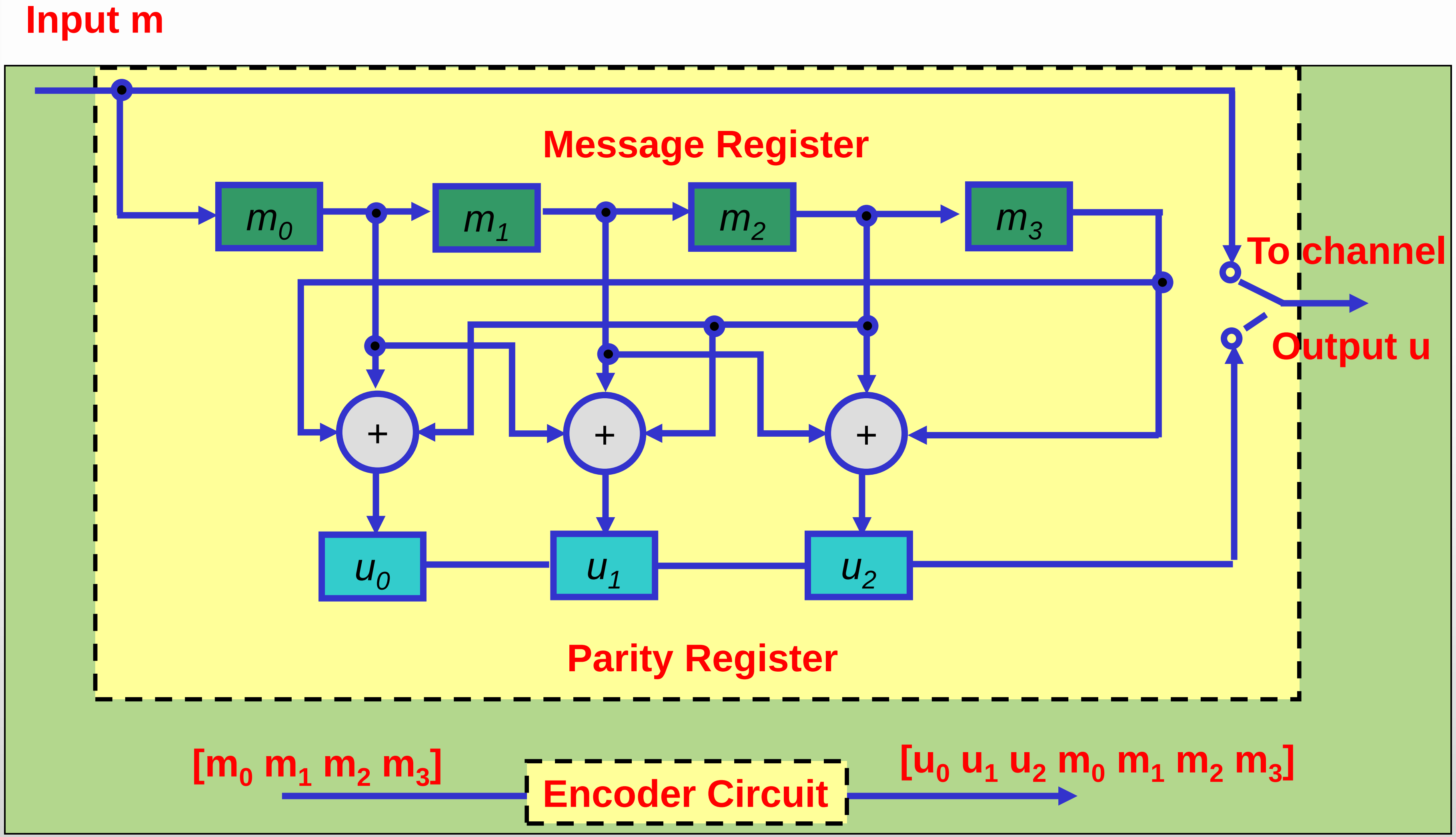
$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\downarrow$$

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

RANGKAIAN ENCODER

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

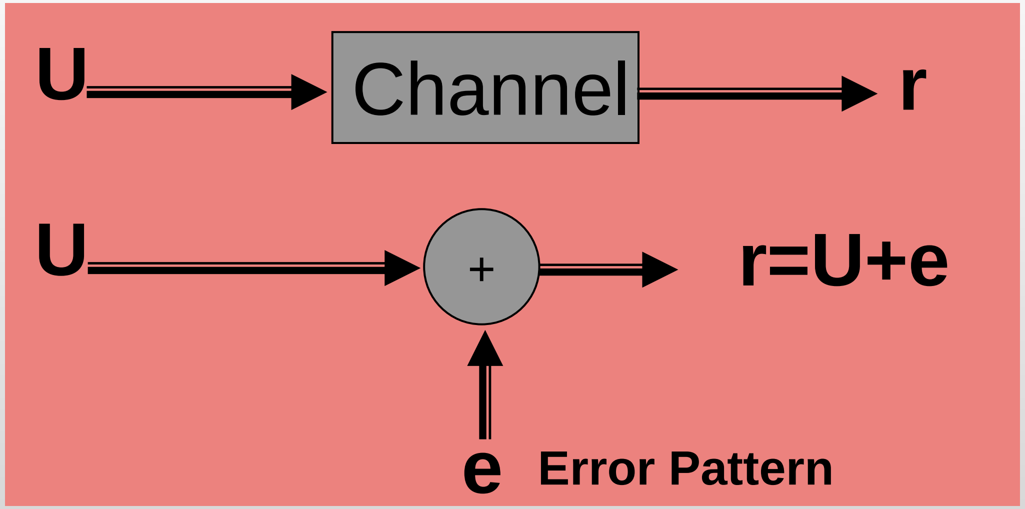


SYNDROME

- Characteristic of parity check matrix (**H**)

$$r.H^T = 0 \Rightarrow r \in C$$

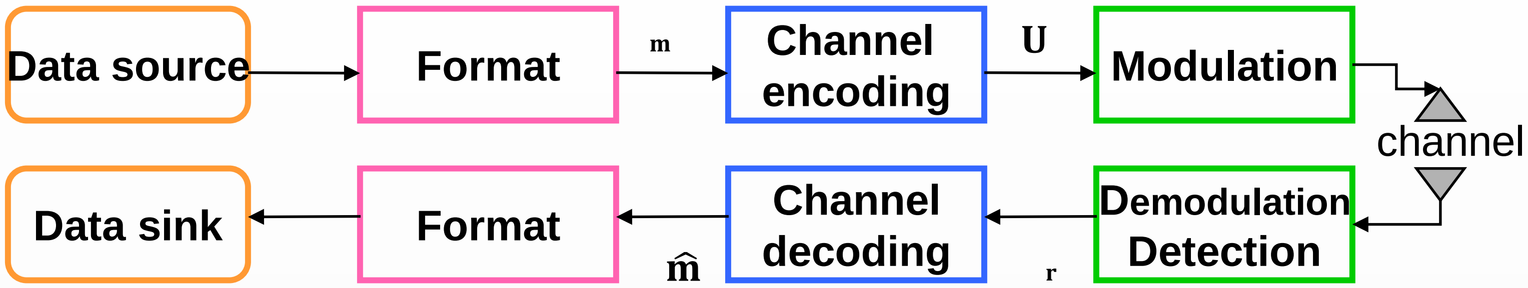
$$r.H^T \neq 0 \Rightarrow r \notin C$$



Syndrome

$$\mathbf{s = r.H^T}$$

Telkom University



$$\mathbf{r} = \mathbf{U} + \mathbf{e}$$

$\mathbf{r} = (r_1, r_2, \ldots., r_n)$ received codeword or vector

$\mathbf{e} = (e_1, e_2, \ldots., e_n)$ error pattern or vector

- Syndrome testing:
  - **S** adalah syndrome dari **r**, terhubung dengan pola error **e**.

$$\mathbf{S} = \mathbf{r}\mathbf{H}^T = \mathbf{e}\mathbf{H}^T$$

Standard array

- Untuk baris ke $i = 2,3,\dots,2^{n-k}$, temukan vector $\boldsymbol{V}_n$ dengan bobot minimum yang tidak tercantum pada list array.
- Nyatakan sebagai pola error $\mathbf{e}_i$ dan bentuk setiap baris ke i sebagai corresponding coset

zero codeword

$$\begin{array}{cccc}
\mathbf{U}_1 & \mathbf{U}_2 & \cdots & \mathbf{U}_{2^k} \\
\mathbf{e}_2 & \mathbf{e}_2 \oplus \mathbf{U}_2 & \cdots & \mathbf{e}_2 \oplus \mathbf{U}_{2^k} \\
\vdots & \cdots & \ddots & \vdots \\
\mathbf{e}_{2^{n-k}} & \mathbf{e}_{2^{n-k}} \oplus \mathbf{U}_2 & \cdots & \mathbf{e}_{2^{n-k}} \oplus \mathbf{U}_{2^k}
\end{array}$$

coset

coset leaders

- ## Standard array and syndrome table decoding

1. Hitung $\mathbf{S} = \mathbf{r}\mathbf{H}^T$
2. Temukan coset leader, $\hat{\mathbf{e}} = \mathbf{e}_i$, hubungkan ke syndrome $\mathbf{S}$.
3. Hitung $\hat{\mathbf{U}} = \mathbf{r} + \hat{\mathbf{e}}$ dan tentukan $\hat{\mathbf{m}}$.

- Note : $\hat{\mathbf{U}} = \mathbf{r} + \hat{\mathbf{e}} = (\mathbf{U} + \mathbf{e}) + \hat{\mathbf{e}} = \mathbf{U} + (\mathbf{e} + \hat{\mathbf{e}})$
  - jika $\hat{\mathbf{e}} = \mathbf{e}$, error di koreksi.
  - Jika $\hat{\mathbf{e}} \neq \mathbf{e}$, terjadi error yang tidak terdeteksi.

- Contoh : **Standard array untuk LBC(6,3)**

codewords

000000	110100	011010	101110	101001	011101	110011	000111
000001	110101	011011	101111	101000	011100	110010	000110
000010	110111	011000	101100	101011	011111	110001	000101
000100	110011	011100	101010	101101	011010	110111	000110
001000	111100	⋮			⋮		⋮
010000	100100						
100000	010100				⋮		
010001	100101		⋯			⋯	010110

coset

Coset leaders

Error pattern	Syndrome
000000	000
000001	101
000010	011
000100	110
001000	001
010000	010
100000	100
010001	111

$\mathbf{U} = (101110)$ transmitted.

$\mathbf{r} = (001110)$ is received.

- The syndrome of $\mathbf{r}$ is computed :

$$\mathbf{S} = \mathbf{r}\mathbf{H}^T = (001110)\mathbf{H}^T = (100)$$

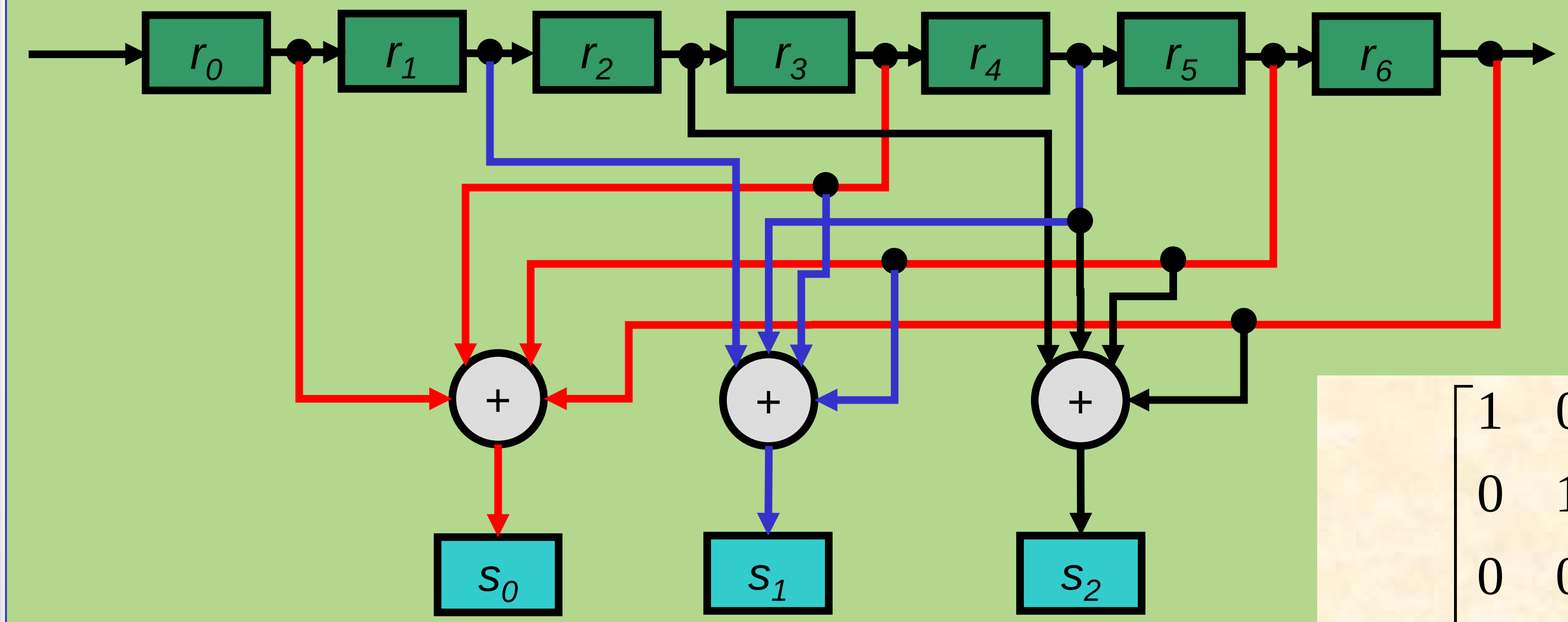
- Error pattern corresponding to this syndrome is

$$\hat{\mathbf{e}} = (100000)$$

- The corrected vector is estimated

$$\hat{\mathbf{U}} = \mathbf{r} + \hat{\mathbf{e}} = (001110) + (100000) = (101110)$$

RANGKAIAN SYNDROME



$$s_0 = r_0 + r_3 + r_5 + r_6$$
$$s_1 = r_1 + r_3 + r_4 + r_5$$
$$s_2 = r_2 + r_4 + r_5 + r_6$$

$$\mathbf{H}^{\mathbf{T}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

HAMMING CODE

- Hamming codes
  - Hamming codes merupakan subclass dari linear block codes dan dikategorikan sebagai *perfect codes*.
  - Hamming codes diekspresikan sebagai sebuah fungsi single integer $m \geq 2$

Code length :	$n = 2^m - 1$
Number of information bits :	$k = 2^m - m - 1$
Number of parity bits :	$n\text{-}k = m$
Error correction capability :	$t = 1$

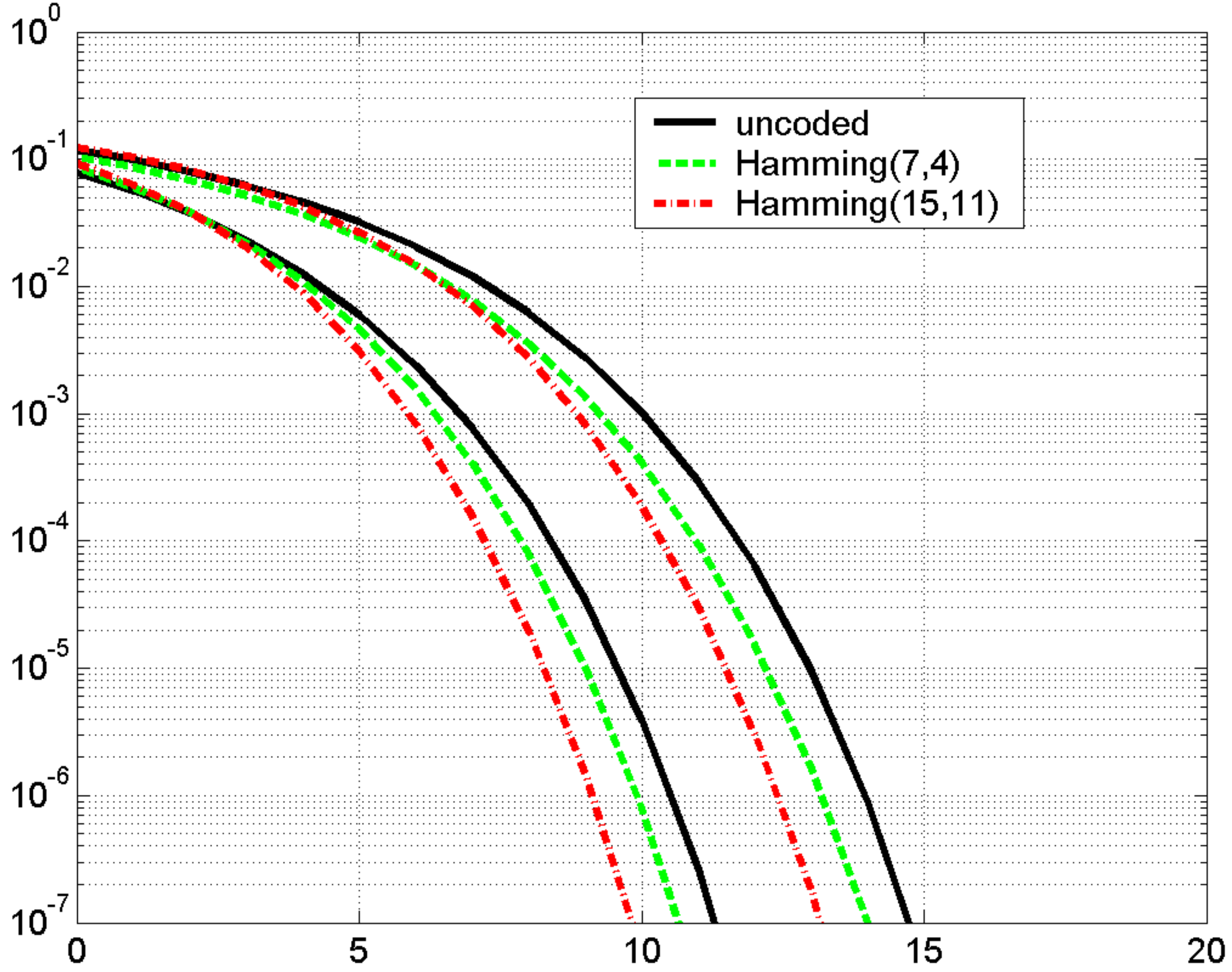

Richard Hamming

  - Kolom pada parity-check matrix, **H**, semuanya terdiri atas non-zero binary m-tuples.

- Example: Systematic Hamming code (7,4)

$$\mathbf{H} = \left[\begin{array}{ccc:cccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{array}\right] = [\mathbf{I}_{3\times 3} \;\vdots\; \mathbf{P}^T]$$

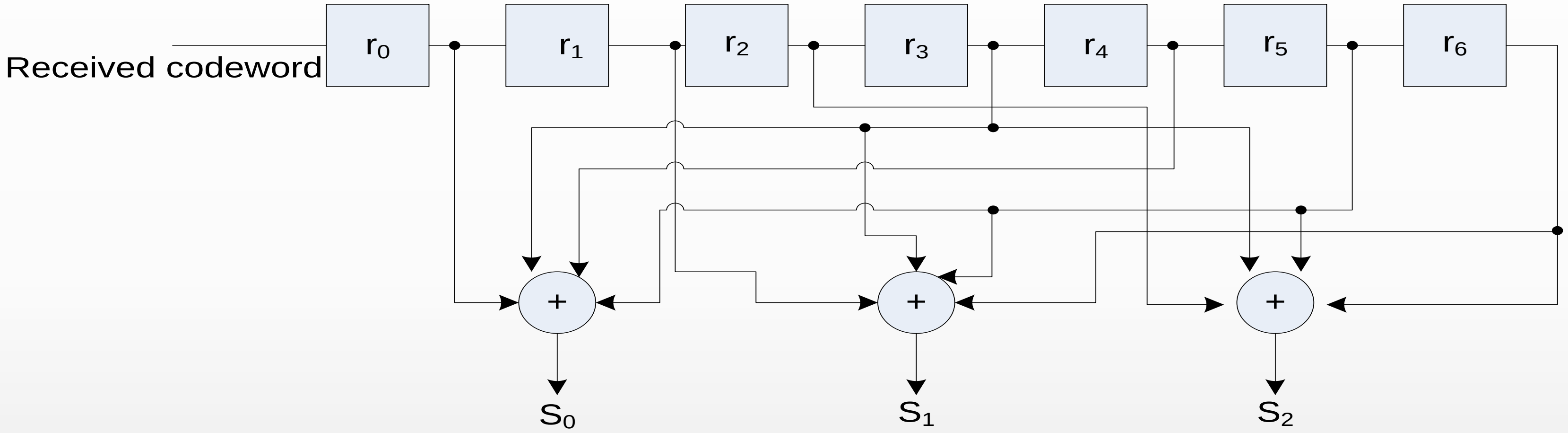
$$\mathbf{G} = \left[\begin{array}{ccc:cccc} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right] = [\mathbf{P} \;\vdots\; \mathbf{I}_{4\times 4}]$$



10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
0
5
10
15
20
uncoded
Hamming(7,4)
Hamming(15,11)

TUGAS

- Rangkaian di bawah ini merupakan rangkaian syndrome pada LBC (7,4).



- Tentukan:
  a. Matriks generator G dan matrik Parity check H.
  b. Tentukan kemampuan deteksi dan koreksi pada LBC tersebut
  c. Bila kode yang diterima adalah 1011001 (t_1 dari kanan) tentukan apakah kode tersebut valid, dan message yang dikirmkan