Part 2. Channel Coding
What’s Channel Coding?

- **Channel coding:**
  - To add redundancy in the information sequence so that the sequence can be recovered at the receiver even in the presence of noise and interference.
  - *Transmission reliability* is improved.
  - Examples:
    - **Block code** ↔ memoryless
      - Repetition code, Hamming code, Maximum-length code, BCH code, Reed-Solomon code
      - Cyclic redundancy check (CRC) code
    - **Convolutional code** ↔ with memory
      - Turbo code
      - LDPC code
Error Control Techniques

- **Forward error-control (FEC)**
  - Using the redundancy in the transmitted code word for both the *detection and correction* of errors incurred during the transmission.
  - Simplex connection

- **Automatic-repeat request (ARQ)**
  - Using the redundancy merely for *error detection*.
  - The receiver sends a feedback to the transmitter, saying that if any error is detected in the received packet or not (Not-Acknowledgement (NACK) and Acknowledgement (ACK), respectively).
  - The transmitter retransmits the previously sent packet if it receives NACK.
  - Full-duplex connection

- **Hybrid ARQ (ARQ+FEC)**
  - Full-duplex connection
  - *Error detection and correction*
FEC Historical Pedigree (1)

1950
- Shannon’s Paper 1948
- Hamming defines basic binary codes
- BCH codes Proposed
- Reed and Solomon define ECC Technique

1960
- Gallager’s Thesis On LDPCs
- Viterbi’s Paper On Decoding Convolutional Codes
- Berlekamp and Massey rediscover Euclid’s polynomial technique and enable practical algebraic decoding
- Forney suggests concatenated codes

1970
- Early practical implementations of RS codes for tape and disk drives

For additional information on Digital Communications III, please refer to ELEC 7073, Department of E.E.E., HKU.
FEC Historical Pedigree (2)

- **1980**
  - Ungerboeck’s TCM Paper - 1982
  - RS codes appear in CD players
  - First integrated Viterbi decoders (late 1980s)
  - TCM Heavily Adopted into Standards

- **1990**
  - Berrou’s Turbo Code Paper - 1993
  - Turbo Codes Adopted into Standards (DVB-RCS, 3GPP, etc.)

- **2000**
  - LDPC beats Turbo Codes For DVB-S2 Standard - 2003
  - Renewed interest in LDPCs due to TC Research

- **1980s**
  - First integrated Viterbi decoders (late 1980s)

- **1990s**
  - Berrou’s Turbo Code Paper - 1993
  - Turbo Codes Adopted into Standards (DVB-RCS, 3GPP, etc.)

- **2000s**
  - LDPC beats Turbo Codes For DVB-S2 Standard - 2003
  - Renewed interest in LDPCs due to TC Research
Part 2.1 Block codes
Some Definitions (1)

Field

- Let $F$ be a set of elements on which two operations *addition* and *multiplication* are defined.

- $F$ is said to be a *field* if and only if
  1. $F$ forms a commutative group under addition operation.
     \[
     \forall a, b \in F \Rightarrow a + b = b + a \in F
     \]
  2. $F$ contains an element called *zero* that satisfies $a + 0 = a$
  3. $F$ forms a commutative group under multiplication operation.
     \[
     \forall a, b \in F \Rightarrow a \cdot b = b \cdot a \in F
     \]
  4. $F$ contains an element called the *identity* satisfying $a(1) = a$
  5. The operations “+” and “.” are distributive:
     \[
     a \cdot (b + c) = (a \cdot b) + (a \cdot c)
     \]
Some Definitions (2)

Galois field: $GF(q)$

- A finite field with $q$ elements
- Example: when $q$ is a prime, $GF(q)$ has the elements \{0, 1, ..., q-1\}. The addition and multiplication operations are defined modulo $q$.

Binary field: $GF(2)$, \{0, 1\}

<table>
<thead>
<tr>
<th>Addition</th>
<th>Multiplication</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 ⊕ 0 = 0</td>
<td>0 · 0 = 0</td>
</tr>
<tr>
<td>0 ⊕ 1 = 1</td>
<td>0 · 1 = 0</td>
</tr>
<tr>
<td>1 ⊕ 0 = 1</td>
<td>1 · 0 = 0</td>
</tr>
<tr>
<td>1 ⊕ 1 = 0</td>
<td>1 · 1 = 1</td>
</tr>
</tbody>
</table>
**Example of Galois Field**

- **GF(5)** is a set of elements \{0, 1, 2, 3, 4\}

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

---

ELEC 7073 Digital Communications III, Dept. of E.E.E., HKU
Some Definitions (3)

Vector space

- Let $V$ be a set of vectors and $F$ a field of elements (*called scalars*). $V$ forms a vector space over $F$ if:

1. **Vector addition commutative:** $\forall u, v \in V \Rightarrow u + v = v + u \in V$

2. **Scalar multiplication:** $\forall a \in F, \forall v \in V \Rightarrow a \cdot v = u \in V$

3. **Distributive:**
   
   $(a + b) \cdot v = a \cdot v + b \cdot v$ and $a \cdot (u + v) = a \cdot u + a \cdot v$

4. **Associative:** $\forall a, b \in F, \forall v \in V \Rightarrow (a \cdot b) \cdot v = a \cdot (b \cdot v)$

5. $\forall v \in V, \ 1 \cdot v = v$

6. **Zero vector:** $0 \in V, \forall v \in V \Rightarrow v + 0 = v$

7. $\forall v \in V$, there exists an element $w \in V$, such that $v + w = 0$
Some Definitions (4)

- Vector space (cont’d):
  - Example
    - The set of binary n-tuples, denoted by $V_n$
    
    $V_4 = \{(0000), (0001), (0010), (0011), (0100), (0101), (0110), (0111), (1000), (1001), (1010), (1011), (1100), (1101), (1110), (1111)\}$

- 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 $S$.
    - The scalar multiplication of the vectors in $S$ is also in $S$.
    - Example: $\{(0000), (0101), (1010), (1111)\}$ is a subspace of $V_4$. 
Some Definitions (5)

Spanning set

- A collection of vectors \( G = \{v_1, v_2, \ldots, v_n\} \)
  the linear combinations of which include all vectors in a vector space \( V \), is said to be a spanning set for \( V \) or to span \( V \).
- Example:
  \( \{(1000), (0110), (1100), (0011), (1001)\} \) spans \( V_4 \).

Basis:

- A spanning set for \( V \) that has minimal cardinality is called a basis for \( V \).
- Cardinality of a set is the number of vectors in the set.
- Example: \( \{(1000), (0100), (0010), (0001)\} \) is a basis for \( V_4 \).
Block Codes (1)

- **(n, k) block code:**
  - The information stream is partitioned into blocks of \( k \) elements.
  - Each block is mapped to a larger block of \( n \) elements, called code word. \( \rightarrow \) memoryless
  - The elements of a code/information block word are selected from an alphabet of \( q \) symbols.
    - Binary code: \( q=2 \); the alphabet = \{0, 1\}
    - Non-binary code: \( q>2 \)

\[ R_c = \frac{k}{n} \] Code rate

\[ n - k \] Redundant elements

\[ \text{Data block} \rightarrow \text{Channel encoder} \rightarrow \text{Code word} \]
Block Codes (2)

(n, k) binary block code

k bits:
2^k information blocks

V_k

mapping

n bits:
2^n possible codes

V_n

n=6, k=3

3-bit blocks are mapped to 6-bit codewords.

(100) \rightarrow (110100); (010) \rightarrow (011010)

Example: (n=6, k=3)

code rate=\frac{3}{6}, 3 redundant elements

Basis of C

k code words
(constructing a generator matrix)
Block Codes (3)

- **Linear block codes:**
  - Suppose \( c_i \) and \( c_j \) are two code words in an \((n,k)\) block code and \( a_1 \) and \( a_2 \) be any two elements selected from the alphabet. Then the code is linear if and only if \( a_1 c_i + a_2 c_j \) is also a code word.
  - A linear code must contain the all-zero code word
  - Easy implementation and analysis

- **Non-linear block codes:**
  - *Example:* \((4, 2)\) code
    - \((0 \ 0) \rightarrow (1 \ 0 \ 0 \ 0)\)
    - \((1 \ 0) \rightarrow (0 \ 1 \ 0 \ 0)\)
    - \((0 \ 1) \rightarrow (0 \ 0 \ 1 \ 0)\)
    - \((1 \ 1) \rightarrow (0 \ 0 \ 0 \ 1)\)
Linear Block Codes (1)

- **Important parameters (1)**
  - **Hamming weight** of a code word \( c_i \), denoted by \( w_i = w(c_i) \), is the number of non-zero elements in \( c_i \).
  - **Hamming distance** between two code words \( c_i \) and \( c_j \), is the number of elements in which they differ.
    \[
    d_{ij} = d(c_i, c_j) = w(c_i \oplus c_j)
    \]
  - **Minimum distance** is \( d_{\text{min}} = \min_{i \neq j} d(c_i, c_j) = \min_{i \neq j} w(c_i) \)

*Examples*
1. \( c_L = [1,0,1] \) \( \Rightarrow w_L = 2 \)
2. \( c_L = [0,0,0] \) \( \Rightarrow d_{LJ} = 3 \)
3. \( c_L = [0,0,1] \) \( \Rightarrow d_{LJ} = 1 \)
Linear Block Codes (2)

Important parameters (2)

- **Error detection capability** is given by \( e = d_{\text{min}} - 1 \)
- **Error correcting-capability** \( t \) of a code, which is defined as the maximum number of guaranteed correctable errors per codeword, is

\[
t = \left\lfloor \frac{(d_{\text{min}} - 1)}{2} \right\rfloor
\]

Copied from *Proakis’s Digital Communications*
Linear Block Codes (3)

Encoding in \((n,k)\) linear block code:

- The generator matrix

\[ c = mG \]

\[ (c_1, c_2, \ldots, c_n) = (m_1, m_2, \ldots, m_k) \cdot \]

\[ (c_1, c_2, \ldots, c_n) = m_1 \cdot g_1 + m_2 \cdot g_2 + \ldots + m_k \cdot g_k \]

Generally, the rows of \(G, \{g_i\}\), are linearly independent. They form a basis for the \((n, k)\) codes.
Example of Linear Block Code (6, 3)

Generator matrix:
Basis of the codewords

$$G = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$c = mG$$

Message vector | Codeword | Hamming weight
--- | --- | ---
000 | 000000 | 0
100 | 110100 | 3
010 | 011010 | 3
110 | 101001 | 3
001 | 101110 | 4
111 | 000111 | 3
101 | 011101 | 4
011 | 110011 | 4
111 | 000111 | 3

$$d_{\text{min}} = 3$$
Linear Block Codes (4)

- **Systematic linear block code (n,k)**
  - For a systematic code, the first (or last) k elements in the codeword are information bits.

\[
G = [I_k \mid P]
\]

\[
I_k = k \times k \quad \text{identity matrix}
\]

\[
P_k = k \times (n - k) \quad \text{matrix}
\]

\[
c = (c_1, c_2, \ldots, c_n) = (m_1, m_2, \ldots, m_k, p_1, p_2, \ldots, p_{n-k})
\]

- message bits
- parity check bits
### Example of Systematic Linear Block Code

A systematic linear block code is a special type of linear block code where the original message is placed at the beginning of the codeword and the remaining positions are filled with parity bits.

#### (7,4) systematic code

- **Generator Matrix** $G$

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

- **Message** and **codeword** table:

<table>
<thead>
<tr>
<th>Message</th>
<th>codeword</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0000 000</td>
</tr>
<tr>
<td>0001</td>
<td>0001 011</td>
</tr>
<tr>
<td>0010</td>
<td>0010 110</td>
</tr>
<tr>
<td>0011</td>
<td>0011 101</td>
</tr>
<tr>
<td>0100</td>
<td>0100 111</td>
</tr>
<tr>
<td>0101</td>
<td>0101 100</td>
</tr>
<tr>
<td>0110</td>
<td>0110 001</td>
</tr>
<tr>
<td>0111</td>
<td>0111 010</td>
</tr>
<tr>
<td>1000</td>
<td>1000 100</td>
</tr>
<tr>
<td>1001</td>
<td>1001 110</td>
</tr>
<tr>
<td>1010</td>
<td>1010 010</td>
</tr>
<tr>
<td>1011</td>
<td>1011 000</td>
</tr>
<tr>
<td>1100</td>
<td>1100 011</td>
</tr>
<tr>
<td>1101</td>
<td>1101 001</td>
</tr>
<tr>
<td>1110</td>
<td>1110 100</td>
</tr>
<tr>
<td>1111</td>
<td>1111 111</td>
</tr>
</tbody>
</table>

---

ELEC 7073 Digital Communications III, Dept. of E.E., HKU
Linear Block Codes (5)

- For any linear code we can find a matrix $H_{(n-k)\times n}$, whose rows are orthogonal to rows of $G$:
  \[
  GH^T = 0
  \]

- $H$ is called the parity check matrix and its rows are linearly independent.

- The relation between any non-zero code word $c$ and the parity check matrix is:
  \[
  cH^T = mGH^T = 0
  \]

- Any linear code $c$ with generator matrix as $H$, is called the dual code of the linear code $c$ with generator matrix as $G$. They are orthogonal.
  \[
  cc^T = mG(mH)^T = mGH^Tm^T = 0
  \]
Systematic linear block code:

\[ G = \begin{bmatrix} I_k \mid P \end{bmatrix}; \quad H = \begin{bmatrix} -P^T \mid I_{n-k} \end{bmatrix} = \begin{bmatrix} P^T \mid I_{n-k} \end{bmatrix} \]

binary codes

\[ G = \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 0
\end{bmatrix} \]

\[ H = \begin{bmatrix}
1 & 1 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1
\end{bmatrix} \]

\[ GH^T = \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 0
\end{bmatrix} \]

\[ = 0 \]
Decoding over AWGN Channel

- **Maximum likelihood decoding**: In the decoding of a $n$-bit block code, the distances between the received code word and the $2^k$ possible transmitted code words are computed. *The code word that is closest in distance to the received code word is selected as the decoder output.*
  - Optimum in the sense that it results in a minimum probability of a code word error
  - *Soft decision decoding* $\leftarrow$ Euclidean distance
  - *Hard decision decoding* $\leftarrow$ Hamming distance
- Also known as minimum-distance decoding
Decoding over AWGN Channel (2)

- **Soft decision decoding using Euclidean distance**
  
  Example:
  
  $(n,k)$ binary block code $\rightarrow$ computing $2^k$ metrics (high complexity)

  The transmitted code word $C_m = [c_{m1}, c_{m2}, \ldots, c_{mn}]$, $c_{mj} = 0$ or $1$,
  
  the transmitted signal: $(2c_{mj} - 1)\sqrt{\epsilon_c} \left( 0 \rightarrow -\sqrt{\epsilon_c}; \ 1 \rightarrow \sqrt{\epsilon_c} \right)$
  
  the received signals: $r = [r_1, r_2, \ldots, r_n] \leftarrow$ unquantized
  
  BPSK modulation+AWGN channel: $r_j = (2c_{mj} - 1)\sqrt{\epsilon_c} + n_j, \quad j = 1, 2, \ldots, n$
  
  Euclidean distance:
  
  $$\sqrt{\sum_{j=1}^{n} (r_j - (2c_{ij} - 1)\sqrt{\epsilon_c})^2} = \sqrt{\sum_{j=1}^{n} \left( r_j^2 + \left( (2c_{ij} - 1)\sqrt{\epsilon_c} \right)^2 \right) - 2r_j (2c_{ij} - 1)\sqrt{\epsilon_c}}$$
  
  Select the code word corresponding to minimal Euclidean distance, mathematically:
  
  $$\max_i CM_i = \max_i C(r, c_i) = \max_i \sum_{j=1}^{n} (2c_{ij} - 1)r_j, \quad i = 1, 2, \ldots, 2^k$$
Decoding over AWGN Channel (3)

- **Hard decision decoding using Hamming distance**
  - Received signal is quantized/detected.
  - Select the code word with minimum Hamming distance to the received coded word
  - Low complexity
  - *Syndrome decoding*

- **Comparison**

Soft decision decoding: better performance with higher complexity

Probability of a decoding error for a binary (3, 1) code over AWGN channel

Copied from *The Art of Error Correcting Coding*, pp. 17.
Syndrome Decoding

Message vector \( \mathbf{m} \) → Generator matrix \( \mathbf{G} \) → Code vector \( \mathbf{C} \) → Received vector \( \mathbf{r} = \mathbf{C} + \mathbf{e} \) → Parity check matrix \( \mathbf{H}^T \) → \( \hat{\mathbf{e}} \leftarrow \mathbf{S} = \mathbf{rH}^T \), \( \hat{\mathbf{C}} = \mathbf{r} + \hat{\mathbf{e}} \)

\[ \mathbf{r} = \mathbf{C} + \mathbf{e} \]

\( \mathbf{r} = (r_1, r_2, \ldots, r_n) \) received codeword or vector
\( \mathbf{e} = (e_1, e_2, \ldots, e_n) \) error pattern or vector

■ Syndrome testing:
  - \( \mathbf{S} \) is syndrome of \( \mathbf{r} \), corresponding to an error pattern \( \mathbf{e} \).

\[ \mathbf{S} = \mathbf{rH}^T = (\mathbf{C} + \mathbf{e})\mathbf{H}^T = \mathbf{eH}^T \]
Syndrome Decoding (2)

How to decode?

1. Calculate $S = rH^T$
2. Find the error pattern, $\hat{e} = e_i$, corresponding to $S$, based on syndrome table.
3. Calculate $\hat{C} = r + \hat{e}$ and the corresponding $\hat{m}$.

- Note that $\hat{C} = r + \hat{e} = (C + e) + \hat{e} = C + (e + \hat{e})$
  - If $\hat{e} = e$, error is corrected.
  - If $\hat{e} \neq e$, undetectable decoding error occurs.

Achieve minimum Hamming distance $\iff$ find $\hat{e}$ with minimum weight 😊
How to Generate Syndrome Table?

- **Standard array**
  1. List all the $2^k$ possible code words in the first row, with $C_1 = \emptyset$ at the first column.
  2. For row $i=2,3,\ldots,2^{n-k}$, find a vector in $V_n$ with **minimum weight** which is not already listed in the array.
  3. Call this pattern $e_i$ and form the $i$th row as the corresponding coset.

The syndromes corresponding to the coset leaders form the syndrome table: $S_i = e_i H^T$. 

The diagram shows a standard array with zero codeword, coset leaders, and cosets.
Example: Standard Array for (6, 3) Code

\[
\begin{array}{cccccccccc}
000000 & 110100 & 011010 & 101110 & 101001 & 011101 & 110011 & 000111 \\
000001 & 110101 & 011011 & 101111 & 101000 & 011100 & 110010 & 000110 \\
000010 & 110111 & 011000 & 101100 & 101011 & 011111 & 110001 & 000101 \\
000100 & 110011 & 011100 & 101010 & 101101 & 011010 & 110111 & 000110 \\
001000 & 111100 & \vdots & \vdots & \vdots \\
010000 & 100100 \\
100000 & 010100 & \vdots \\
010001 & 100101 & \vdots & \vdots & 010110 \\
\end{array}
\]

\[
G = \begin{bmatrix}
110100 \\
011010 \\
101001
\end{bmatrix}, \quad H = \begin{bmatrix}
100101 \\
010110 \\
001011
\end{bmatrix}
\]

\[
d_{\min} = 3 \Rightarrow t = 1
\]
Example of Error Detection and Correction

(6, 3) linear block code : Syndrome decoding

\[ S_i = e_i H^T \]

<table>
<thead>
<tr>
<th>Error pattern</th>
<th>Syndrome</th>
</tr>
</thead>
<tbody>
<tr>
<td>000000</td>
<td>000</td>
</tr>
<tr>
<td>000001</td>
<td>101</td>
</tr>
<tr>
<td>000010</td>
<td>011</td>
</tr>
<tr>
<td>000100</td>
<td>110</td>
</tr>
<tr>
<td>001000</td>
<td>001</td>
</tr>
<tr>
<td>010000</td>
<td>010</td>
</tr>
<tr>
<td>100000</td>
<td>100</td>
</tr>
<tr>
<td>010001</td>
<td>111</td>
</tr>
</tbody>
</table>

\[ C = [101110] \] is transmitted.
\[ r = [001110] \] is received.

The syndrome of \( r \) is computed:
\[ S = rH^T = (001110)H^T = (100) \]

Error pattern corresponding to this syndrome is \( \hat{e} = (100000) \)

The received vector is corrected as
\[ \hat{C} = r + \hat{e} = (001110) + (100000) = (101110) \]

\[ \hat{e} : \text{minimum weight} \]
\[ \downarrow \]
\[ \text{minimum distance between } \hat{C} \text{ and } r \]
Example of Error Detection and Correction (2)

- **(6, 3) linear block code**: Syndrome decoding

  - Syndrome computation:
    \[ S_i = e_i H^T \]
    | Error pattern | Syndrome |
    |---------------|----------|
    | 000000        | 000      |
    | 000001        | 101      |
    | 000010        | 011      |
    | 000100        | 110      |
    | 001000        | 001      |
    | 010000        | 010      |
    | 100000        | 100      |
    | **010001**    | **111**  |

  - Ciphertext: \( C = \begin{bmatrix} 101110 \end{bmatrix} \) is transmitted.
  - Received vector: \( r = \begin{bmatrix} 001100 \end{bmatrix} \) is received.

  - Syndrome of \( r \) is computed:
    \[ S = rH^T = (001100)H^T = (111) \]

  - Two errors are uncorrectable.
  
  - Error pattern corresponding to this syndrome is \( \hat{e} = (010001) \).
  
  - The decoded vector is
    \[ \hat{C} = r + \hat{e} = (001100) + (010001) = (011101) \]
Hamming codes

- Hamming codes are a subclass of linear block codes and belong to the category of perfect codes.
- Hamming codes are expressed as a function of a single integer $m \geq 2$.

<table>
<thead>
<tr>
<th>Code length:</th>
<th>$n = 2^m - 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of information bits:</td>
<td>$k = 2^m - m - 1$</td>
</tr>
<tr>
<td>Number of parity bits:</td>
<td>$n - k = m$</td>
</tr>
<tr>
<td>Error correction capability:</td>
<td>$t = 1$</td>
</tr>
<tr>
<td>Minimum distance:</td>
<td>$d_{\text{min}} = 3$</td>
</tr>
</tbody>
</table>

- The columns of the parity-check matrix, $H$, consist of all non-zero binary $m$-tuples.
Example of Hamming Code (1)

➢ Systematic Hamming code (7, 4)

\[ m=3, \quad n=2^3-1=7, \quad k=2^3-m-1=4 \]

\[
H = \begin{bmatrix}
1 & 1 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1
\end{bmatrix} = [P^T \quad I_{3\times3}]
\]

\[
G = \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 0
\end{bmatrix} = [I_{4\times4} \quad P]
\]
Example of Hamming Code (2)

- **Systematic Hamming code (7, 4)**

The Hamming (7, 4) code is constructed as follows: the information sequence \([u_1, u_2, u_3, u_4]\) is encoded into the codeword

\[
[u_1, u_2, u_3, u_4, u_1 \oplus u_2 \oplus u_4, u_1 \oplus u_3 \oplus u_4, u_1 \oplus u_2 \oplus u_3]
\]

\[= \mathbf{p}_1 \quad = \mathbf{p}_2 \quad = \mathbf{p}_3\]

\[\mathbf{x}_i = [u_1 u_2 u_3 u_4] \rightarrow \mathbf{c}_i = [u_1 u_2 u_3 u_4 p_1 p_2 p_3]\]

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

\[d_{\text{min}} = 3\]
Example of Hamming Code (3)

Coding gain:
For a given bit-error probability, the reduction in \( E_b/N_0 \) that can be realized through the use of coding:

\[
G \text{ [dB]} = \left( \frac{E_b}{N_0} \right)_u \text{ [dB]} - \left( \frac{E_b}{N_0} \right)_c \text{ [dB]}
\]
Cyclic Block Codes

- Cyclic block codes are a subclass of linear block codes. BCH and Reed-Solomon are cyclic codes.
- A linear (n,k) code is called a cyclic block code if all cyclic shifts of a codeword are also codewords.

\[
C = (c_0, c_1, c_2, \ldots, c_{n-1})
\]

\[
C^{(1)} = (c_{n-i}, c_{n-i+1}, \ldots, c_{n-1}, c_0, c_1, c_2, \ldots, c_{n-i-1})
\]

Example:

\[
C = (1101) \\
C^{(1)} = (1110) \\
C^{(2)} = (0111) \\
C^{(3)} = (1011) \\
C^{(4)} = (1101) = C
\]

Cyclic property enables efficient implementations of encoding and decoding.
Cyclic Block Codes (2)

- Cyclic block codes in polynomial form
  \[ C(X) = c_0 + c_1X + c_2X^2 + ... + c_{n-1}X^{n-1} \quad \text{degree} \ (n-1) \]

- Relationship between a codeword and its cyclic shifts:
  \[ X \cdot C(X) = c_{n-1}X + c_0X + c_1X^2 + ... + c_{n-2}X^{n-1} + c_{n-1}X^n \]
  \[ = \underbrace{c_{n-1} + c_0X + c_1X^2 + ... + c_{n-2}X^{n-1}}_{C^{(i)}(X)} + \underbrace{c_{n-1}X^n + c_{n-1}}_{c_{n-1}(X^n + 1)} \]
  \[ = C^{(i)}(X) + c_{n-1}(X^n + 1) \]

By extension
\[ C^{(i)}(X) = X^i C(X) \mod (X^n + 1) \]
**Cyclic Block Codes (3)**

- **Basic properties**

  Let C be a binary (n,k) cyclic code

  1. There is a polynomial \( g(X) \) with degree \( r = n - k \). \( g(X) \) is called the generator polynomial.

     \[
     g(X) = g_0 + g_1X + \ldots + g_rX^r
     \]

  2. Every code polynomial \( C(X) \) in C, can be expressed uniquely as

     \[
     C(X) = m(X)g(X)
     \]

     \[
     m(X) = m_0 + m_1X + m_2X^2 + \ldots + m_{k-1}X^{k-1} \quad \text{degree (k - 1)}
     \]

  3. The generator polynomial \( g(X) \) is a factor of \( X^n + 1 \)
Cyclic Block Codes (4)

- Basic properties (cont’d)

4. The row $i, i = 1, ..., k$, of the generator matrix is formed by the coefficients of the "$i-1" cyclic shift of the generator polynomial.

$$G = \begin{bmatrix} g(X) \\ Xg(X) \\ \vdots \\ X^{k-1}g(X) \end{bmatrix} = \begin{bmatrix} g_0 & g_1 & \cdots & g_r & 0 \\ g_0 & g_1 & \cdots & g_r & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & g_0 & g_1 & \cdots & g_r \end{bmatrix}$$

5. The orthogonality of $G$ and $H$ in polynomial form is expressed as $g(X)h(X) = X^n + 1$. This means $h(X)$ is also a factor of $X^n + 1$. $h(X)$ denotes the parity polynomial that has degree of $k$.

$$\left[ C(X)h(X) \bmod (X^n + 1) \right] = \left[ m(X)g(X)h(X) \bmod (X^n + 1) \right] = 0$$
Example of Cyclic Block Code (1)

$(n=7, k=4)$ cyclic code

$$X^7 + 1 = (1 + X)(1 + X + X^3)(1 + X^2 + X^3)$$

$$g(X) = (1 + X^2 + X^3) \rightarrow (1 0 1 1)$$

$$h(X) = (1 + X)(1 + X + X^3) = 1 + X^2 + X^3 + X^4$$

$$G = \begin{bmatrix} g(X) \\ Xg(X) \\ \vdots \\ X^{k-1}g(X) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Generator matrix
Example of Cyclic Block Code (2)

- **(n=7, k=4) cyclic code (Cont’d)**

$$m(X) = m_0 + m_1 X + m_2 X^2 + m_3 X^3$$

$$g(X) = (1 + X^2 + X^3)$$

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

<table>
<thead>
<tr>
<th>m_0</th>
<th>m_1</th>
<th>m_2</th>
<th>m_3</th>
<th>c_0</th>
<th>c_1</th>
<th>c_2</th>
<th>c_3</th>
<th>c_4</th>
<th>c_5</th>
<th>c_6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Cyclic property:

$$G \cdot \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix} \equiv \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} \pmod{2}$$
Cyclic Block Codes (5)

- **(n, k) systematic cyclic code generation**

Multiply the message polynomial $m(X)$ by $X^{n-k}$

Divide the result of Step 1 by the generator polynomial $g(X)$ to obtain the remainder $p(X)$

$$p(X) = X^{n-k}m(X) \mod g(X)$$

$$X^{n-k}m(X) = q(X)g(X) + p(X)$$

Add $p(X)$ to $X^{n-k}m(X)$ to form the codeword $C(X)$

$$C(X) = X^{n-k}m(X) + p(X) = q(X)g(X) + p(X)$$

- Shifting the message signal
For the systematic (7,4) cyclic code with generator polynomial \( g(X) = 1 + X + X^3 \)

Find the codeword for the message \( m = (1011) \)

\( n = 7, \quad k = 4, \quad n - k = 3 \)

\[ m = (1011) \Rightarrow m(X) = 1 + X^2 + X^3 \]

\[ X^{n-k} m(X) = X^3 m(X) = X^3 (1 + X^2 + X^3) = X^3 + X^5 + X^6 \]

Divide \( X^{n-k} m(X) \) by \( g(X) \):

\[ X^3 + X^5 + X^6 = (1 + X + X^2 + X^3) \bigg( 1 + X + X^3 \bigg) + \frac{1}{g(X)} \]

Form the codeword polynomial:

\[ C(X) = p(X) + X^3 m(X) = 1 + X^3 + X^5 + X^6 \]

\[ C = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \]

parity bits message bits
Example of Systematic Cyclic Block Code (2)

- For the systematic (7,4) cyclic code with generator polynomial $g(X) = 1 + X + X^3$

✓ Find the generator and parity check matrices, $G$ and $H$, respectively.

$$g(X) = 1 + 1 \cdot X + 0 \cdot X^2 + 1 \cdot X^3 \Rightarrow (g_0, g_1, g_2, g_3) = (1101)$$

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

Not in systematic form. We do the following:
- row(1) + row(3) → row(3)
- row(1) + row(2) + row(4) → row(4)
Example of Systematic Cyclic Block Code (3)

- For the systematic (7,4) cyclic code with generator polynomial \( g(X) = 1 + X + X^3 \)

<table>
<thead>
<tr>
<th>( m_0 )</th>
<th>( m_1 )</th>
<th>( m_2 )</th>
<th>( m_3 )</th>
<th>( c_0 )</th>
<th>( c_1 )</th>
<th>( c_2 )</th>
<th>( c_3 )</th>
<th>( c_4 )</th>
<th>( c_5 )</th>
<th>( c_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\( d_{\text{min}} = 3 \)
Syndrome decoding in polynomial form

- Received codeword in polynomial form is given by
  \[ r(X) = C(X) + e(X) = m(X)g(X) + e(X) \]

- The syndrome is the reminder obtained by dividing the received polynomial by the generator polynomial.
  \[ r(X) = q(X)g(X) + S(X) \]

- With syndrome and error patterns, error is estimated. The corrected codeword in polynomial form is obtained as
  \[ \hat{C}(X) = r(X) + \hat{e}(X) \]
Example of Error Detection and Correction

 Syndrome decoding: cyclic code (7, 4) \( g(X) = 1 + X + X^3 \)

\[ \mathbf{C} = (1010001) \iff \mathbf{C}(X) = 1 + X^2 + X^6 \]
\[ \mathbf{r} = (1011001) \text{ is received.} \iff \mathbf{r}(X) = 1 + X^2 + X^3 + X^6 \]

\[ \mathbf{r}(X) = 1 + X^2 + X^3 + X^6 = (X + X^3)(1 + X + X^3) + (1 + X) \]

The syndrome of \( \mathbf{r} \) is computed as the remainder of \( \mathbf{r}(X)/\mathbf{g}(X) \):

\[ \mathbf{S}(X) = (1 + X) \iff \mathbf{S} = (110) \]

Error pattern corresponding to this syndrome is

\[ \hat{\mathbf{e}} = (0001000) \iff \hat{\mathbf{e}}(X) = X^3 \]

The corrected codeword in polynomial is estimated as

\[ \hat{\mathbf{C}}(X) = \mathbf{r}(X) + \hat{\mathbf{e}}(X) = 1 + X^2 + X^3 + X^6 + X^3 = 1 + X^2 + X^6 \]

\[ \hat{\mathbf{C}} = \mathbf{r} + \hat{\mathbf{e}} = (1011001) + (0001000) = (1010001) \]