

BỘ CÔNG THƯƠNG
TRƯỜNG CAO ĐẲNG KỸ THUẬT CAO THẮNG
KHOA ĐIỆN TỬ - TIN HỌC

CHƯƠNG 3: GIAO TIẾP KẾT NỐI SỐ LIỆU

Môn Học
TRUYỀN SỐ LIỆU



NỘI DUNG

- 3.1 Các khái niệm cơ bản về truyền số liệu*
- 3.2 Thông tin nối tiếp không đồng bộ*
- 3.3 Thông tin nối tiếp đồng bộ*
- 3.4 Mạch điều khiển truyền số liệu*



NỘI DUNG

3.1 Các khái niệm cơ bản về truyền số liệu

3.2 Thông tin nối tiếp không đồng bộ

3.3 Thông tin nối tiếp đồng bộ

3.4 Mạch điều khiển truyền số liệu

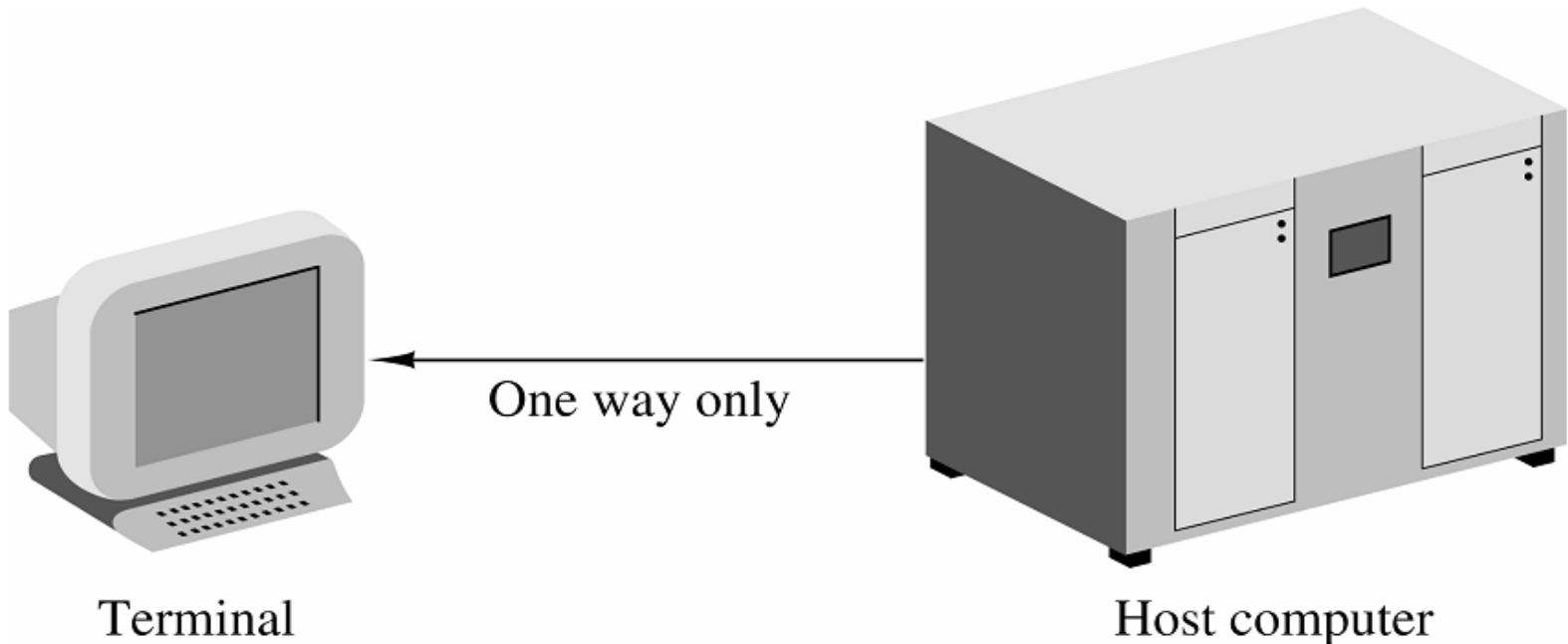


CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)

- Đơn công (one way hay simplex)
- Bán song công (either way hay half-duplex)
- Song công (both way hay full-duplex)

CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)

- Đơn công (one way hay simplex): dữ liệu truyền chỉ theo một hướng duy nhất (radio, TV)

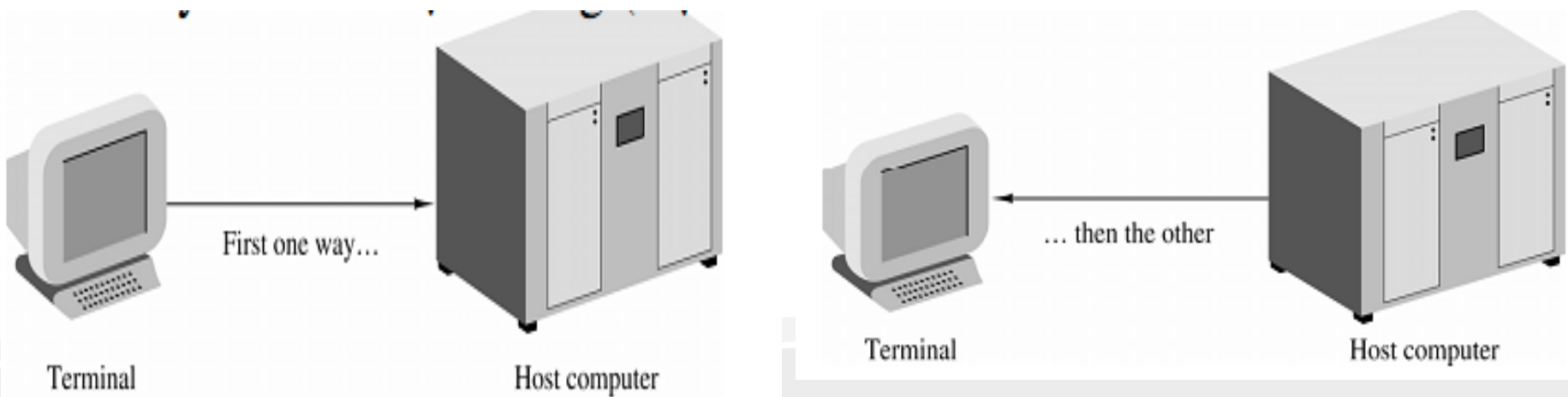


CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)

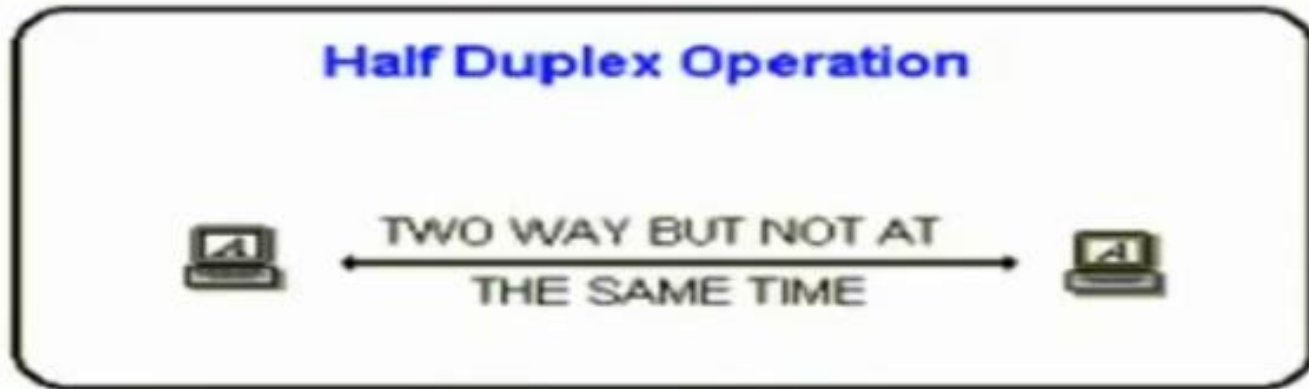


CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)

- Bán song công (either way hay half-duplex): thông tin được truyền theo 2 chiều nhưng không đồng thời, tại mỗi thời điểm thông tin chỉ có thể truyền theo một hướng (Bộ đàm)

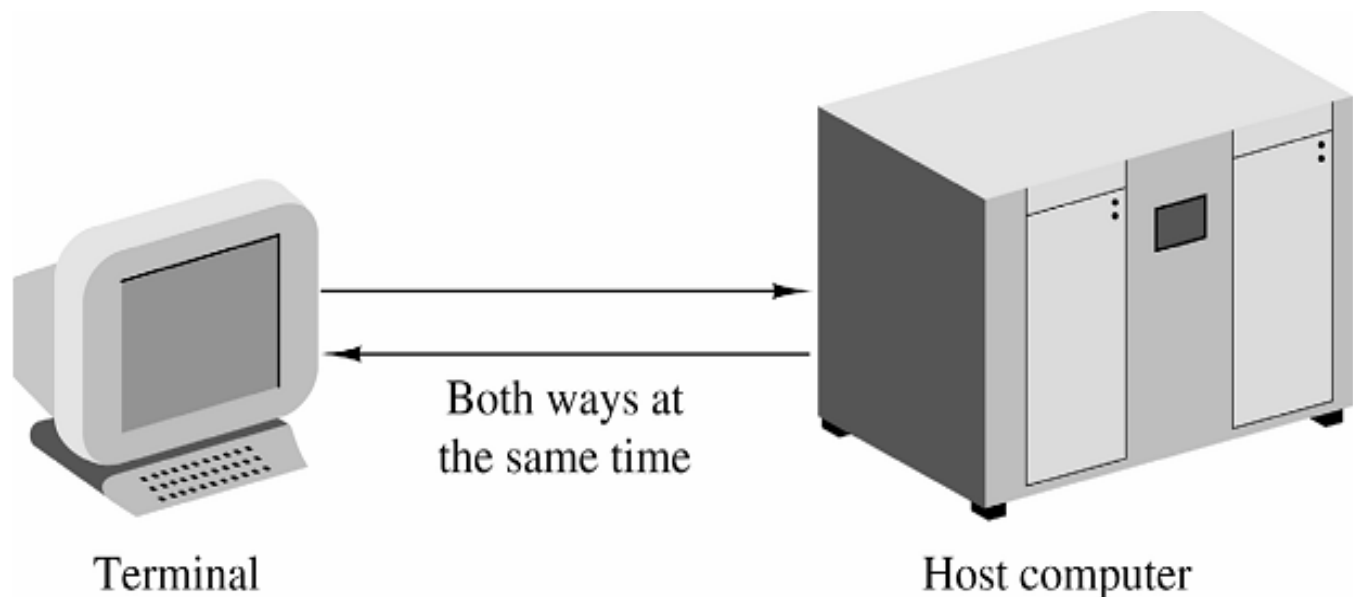


CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)



CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)

- Song công (both way hay full-duplex): thông tin có thể được truyền theo 2 chiều tại cùng một thời điểm trên tuyến dữ liệu (telephone)



CÁC CHẾ ĐỘ THÔNG TIN (COMMUNICATION MODES)



TRUYỀN BẤT ĐỒNG BỘ (Asynchronous Transmission)

- ❑ Là cách thức truyền mà các ký tự được truyền đi tại những thời điểm khác nhau mà khoảng thời gian nối tiếp giữa 2 ký tự không cần thiết phải là giá trị cố định

TRUYỀN BẤT ĐỒNG BỘ (Asynchronous Transmission)



- Máy phát và máy thu độc lập trong việc sử dụng đồng hồ
- Đồng hồ chính là bộ phát xung clock cho việc dịch bit dữ liệu
- Để nhận được dữ liệu máy thu phải đồng bộ theo từng ký tự một
- Sử dụng để truyền ký tự giữa một bàn phím và một máy tính, hay truyền các khối ký tự giữa 2 máy tính
- Ứng dụng khi truyền tốc độ trung bình và thấp

TRUYỀN ĐỒNG BỘ

(Synchronous Transmission)

- ❑ Là cách thức truyền trong đó khoảng thời gian cho mỗi bit như nhau
- ❑ Khoảng thời gian từ bit cuối của ký tự này đến bit đầu của ký tự kế tiếp bằng 0 hoặc bằng bội số tổng thời gian cần thiết truyền hoàn chỉnh một ký tự

TRUYỀN ĐỒNG BỘ

(Synchronous Transmission)

- ❑ Máy phát và máy thu sử dụng một đồng hồ chung
- ❑ Khối dữ liệu hoàn chỉnh được truyền thành một luồng bit liên tục không có bất cứ sự trễ nào giữa các phân tử 8 bit (ký tự)
- ❑ Ứng dụng khi truyền tốc độ cao

KIỂM SOÁT LỖI (Error Control)

□ Trong quá trình truyền luồng bit giữa 2 DTE rất thường xảy ra sai lạc thông tin

□ Ví dụ

➤ Truyền1 0 1 1 0.....

➤ Nhận1 0 1 0 0.....

□ Cần có phương tiện phát hiện lỗi và sửa lỗi

KIỂM SOÁT LỖI (Error Control)

- ❑ Sử dụng các lược đồ để phát hiện lỗi, sửa lỗi
- ❑ Việc chọn lược đồ tùy vào phương pháp truyền được dùng
 - **Truyền bất đồng bộ:** thêm 1 ký số nhị phân vào mỗi ký tự, ký số này còn gọi là bit chẵn lẻ (parity bit)
 - **Truyền đồng bộ:** xác định lỗi xảy ra trên một frame hoàn chỉnh, dùng tuần tự để kiểm tra lỗi phức tạp hơn
- ❑ Khi phát hiện lỗi truyền thì máy thu cần lấy một bản copy khác từ nguồn

ĐIỀU KHIỂN LƯỒNG (Flow Control)

- Khi 2 thiết bị truyền thông tin qua mạng số liệu hoạt động với tốc độ khác nhau thì phải điều khiển số liệu đầu ra của thiết bị tốc độ cao hơn để ngăn chặn trường hợp tắc nghẽn

CÁC GIAO THỨC LIÊN KẾT DỮ LIỆU

- ❑ Giao thức là một tập hợp các tiêu chuẩn hay quy định phải tuân theo bởi cả hai đối tác ở đầu
- ❑ Giao thức liên kết số liệu định nghĩa những chi tiết sau:
 - Khuôn dạng của mẫu số liệu đang trao đổi
 - Dạng và thứ tự các thông điệp được trao đổi để đạt được độ tin cậy giữa 2 đối tác truyền

MÃ TRUYỀN

(Transmission Code)

- ❑ Các bộ mã là tập hợp một số giới hạn các tổ hợp nhị phân
- ❑ Mỗi tổ hợp bit nhị phân mang ý nghĩa của của một ký tự nào đó theo quy định của từng bộ mã
- ❑ Số lượng bit nhị phân trong một tổ hợp bit nói lên quy mô của một bộ mã
- ❑ **Ví dụ:** gọi **n là số bit** trong một tổ hợp bit thì **số ký tự** có thể mã hóa là **2^n**
- ❑ Một số bộ mã thông dụng: Baudot, BCD, EBCDIC, ASCII

MÃ TRUYỀN

(Transmission Code)



Mã Baudot

- Năm 1874, Emil Baudot (người Pháp) đã phát triển mã chữ và số Baudot
- Dùng trong mạng telex
- Là mã chữ và số gồm **5 bits** cho phép biểu diễn **32 ký tự**
- Dùng 2 ký tự đặc biệt để mở rộng tập mã là Letters Shift (LS) hay Figures Shift (FS)
- Các ký tự FS/LS đi trước một mã ký tự sẽ cho ra các ý nghĩa khác nhau.

BẢNG MÃ BAUDOT

| Bin | Dec | Hex | Letter | Figure | Bin | Dec | Hex | Letter | Figure |
|-------|-----|-----|----------------|--------|-------|-----|-----|-----------|--------|
| 00000 | 0 | 0 | Blank | Blank | 10000 | 16 | 10 | T | 5 |
| 00001 | 1 | 1 | E | 3 | 10001 | 17 | 11 | Z | " |
| 00010 | 2 | 2 | LF (Line feed) | LF | 10010 | 18 | 12 | L |) |
| 00011 | 3 | 3 | A | - | 10011 | 19 | 13 | W | 2 |
| 00100 | 4 | 4 | Space | Space | 10100 | 20 | 14 | H | # |
| 00101 | 5 | 5 | S | ' | 10101 | 21 | 15 | Y | 6 |
| 00110 | 6 | 6 | I | 8 | 10110 | 22 | 16 | P | 0 |
| 00111 | 7 | 7 | U | 7 | 10111 | 23 | 17 | Q | 1 |
| 01000 | 8 | 8 | CR | CR | 11000 | 24 | 18 | O | 9 |
| 01001 | 9 | 9 | D | \$ | 11001 | 25 | 19 | B | ? |
| 01010 | 10 | A | R | 4 | 11010 | 26 | 1A | G | & |
| 01011 | 11 | B | J | BELL | 11011 | 27 | 1B | FS | FS |
| 01100 | 12 | C | N | , | 11100 | 28 | 1C | M | . |
| 01101 | 13 | D | F | ! | 11101 | 29 | 1D | X | / |
| 01110 | 14 | E | C | : | 11110 | 30 | 1E | V | ; |
| 01111 | 15 | F | K | (| 11111 | 31 | 1F | LS | LS |

Ví dụ: Chuỗi NO. 27 có dạng như sau :

LS N O FS . SPC 2 7
11111 01100 11000 11011 11100 00100 10011 00111

MÃ TRUYỀN

(Transmission Code)

☐ Mã BCD

- Binary Coded Decimal (BCD) là một cách khác để biểu diễn số thập phân (decimal numbers) ở dạng nhị phân.
- BCD được sử dụng rộng rãi và kết hợp các đặc tính của hệ thập phân và nhị phân.
- Mỗi chữ số thập phân được chuyển thành dạng nhị phân tương ứng.

MÃ TRUYỀN (Transmission Code)

□ Biến đổi số 874_{10} sang BCD:

$$\begin{array}{ccc} 8 & 7 & 4 \\ 1000 & 0111 & 0100 = 100001110100_{\text{BCD}} \end{array}$$

- Mỗi chữ số thập phân (decimal digit) là 4 bits.
- Mỗi nhóm 4-bit không bao giờ lớn hơn 9.
- Làm ngược lại để biến đổi từ BCD sang thập phân.

$$0110100000111001_{\text{BCD}} = 0110-1000-0011-1001_{\text{BCD}}$$

6 8 3 9



MÃ TRUYỀN (Transmission Code)

□ Mã EBCDIC

- Extended Binary Codes Decimal Interchange Code
- Phát triển bởi IBM năm 1962
- Bảng mã này dùng **8 bits** để biểu diễn **$2^8 = 256$** ký tự.

MÃ TRUYỀN

(Transmission Code)

| Char | EBCDIC | HEX | Char | EBCDIC | HEX | Char | EBCDIC | HEX |
|------|-----------|-----|------|-----------|-----|------|-----------|-----|
| A | 1100 0001 | C1 | P | 1101 0111 | D7 | 4 | 1111 0100 | F4 |
| B | 1100 0010 | C2 | Q | 1101 1000 | D8 | 5 | 1111 0101 | F5 |
| C | 1100 0011 | C3 | R | 1101 1001 | D9 | 6 | 1111 0110 | F6 |
| D | 1100 0100 | C4 | S | 1110 0010 | E2 | 7 | 1111 0111 | F7 |
| E | 1100 0101 | C5 | T | 1110 0011 | E3 | 8 | 1111 1000 | F8 |
| F | 1100 0110 | C6 | U | 1110 0100 | E4 | 9 | 1111 1001 | F9 |
| G | 1100 0111 | C7 | V | 1110 0101 | E5 | ... | ... | |
| H | 1100 1000 | C8 | W | 1110 0110 | E6 | ... | ... | |
| I | 1100 1001 | C9 | X | 1110 0111 | E7 | ... | ... | |
| J | 1101 0001 | D1 | Y | 1110 1000 | E8 | ... | ... | |
| K | 1101 0010 | D2 | Z | 1110 1001 | E9 | ... | ... | |
| L | 1101 0011 | D3 | 0 | 1111 0000 | F0 | ... | ... | |
| M | 1101 0100 | D4 | 1 | 1111 0001 | F1 | ... | ... | |
| N | 1101 0101 | D5 | 2 | 1111 0010 | F2 | ... | ... | |
| O | 1101 0110 | D6 | 3 | 1111 0011 | F3 | ... | ... | |

| binary | MSN | 0000 | | 0001 | | 0010 | | 0011 | | 0100 | | 0101 | | 0110 | | 0111 | |
|--------|-----|------------|----------|------------|----------|------------|----------|------------|----------|--------------------|----------|--------------|----------|----------|-----------|-----------|-----------|
| LSN | hex | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | |
| 0000 | 0 | NUL | 0 00 | DLE | 16 10 | DS | 32 20 | | 48 30 | SP | 64 40 | & | 80 50 | - | 96 60 | | 112 70 |
| 0001 | 1 | SOH | 1 01 | DC1 | 17 11 | SOS | 33 21 | | 49 31 | | 65 41 | | 81 51 | / | 97 61 | | 113 71 |
| 0010 | 2 | STX | 2 02 | DC2 | 18 12 | FS | 34 22 | SYN | 50 32 | | 66 42 | | 82 52 | | 98 62 | | 114 72 |
| 0011 | 3 | ETX | 3 03 | TM | 19 13 | | 35 23 | | 51 33 | | 67 43 | | 83 53 | | 99 63 | | 115 73 |
| 0100 | 4 | PF | 4 04 | RES | 20 14 | BYP | 36 24 | PN | 52 34 | | 68 44 | | 84 54 | | 100 64 | | 116 74 |
| 0101 | 5 | HT | 5 05 | NL | 21 15 | LF | 37 25 | RS | 53 35 | | 69 45 | | 85 55 | | 101 65 | | 117 75 |
| 0110 | 6 | LC | 6 06 | BS | 22 16 | ETB | 38 26 | UC | 54 36 | | 70 46 | | 86 56 | | 102 66 | | 118 76 |
| 0111 | 7 | DEL | 7 07 | IL | 23 17 | ESC | 39 27 | EOT | 55 37 | | 71 47 | | 87 57 | | 103 67 | | 119 77 |
| 1000 | 8 | | 8 08 | CAN | 24 18 | | 40 28 | | 56 38 | | 72 48 | | 88 58 | | 104 68 | | 120 78 |
| 1001 | 9 | | 9 09 | EM | 25 19 | | 41 29 | | 57 39 | | 73 49 | | 89 59 | | 105 69 | | 121 79 |
| 1010 | A | SMM | 10 0A | CC | 26 1A | SM | 42 2A | | 58 3A | ? (002d) | 74 4A | ! | 90 5A | 3 | 106 6A | : | 122 7A |
| 1011 | B | VT | 11 0B | CUI | 27 1B | CU2 | 43 2B | CU3 | 59 3B | . | 75 4B | \$ | 91 5B | , | 107 6B | # | 123 7B |
| 1100 | C | FF | 12 0C | IFS | 28 1C | | 44 2C | DC4 | 60 3C | < | 76 4C | * | 92 5C | % | 108 6C | @ | 124 7C |
| 1101 | D | CR | 13 0D | IGS | 29 1D | ENQ | 45 2D | NAK | 61 3D | (| 77 4D |) | 93 5D | - | 109 6D | . | 125 7D |
| 1110 | E | SO | 14 0E | IRS | 30 1E | ACK | 46 2E | | 62 3E | + | 78 4E | ; | 94 5E | > | 110 6E | = | 126 7E |
| 1111 | F | SI | 15 0F | IUS | 31 1F | BEL | 27 2F | SUB | 63 3F | _1 | 79 4F | ?_2 | 95 5F | ? | 111 6F | .. | 127 7F |

| binary | MSN | 1000 | | 1001 | | 1010 | | 1011 | | 1100 | | 1101 | | 1110 | | 1111 | |
|--------|-----|----------|-----------|----------|-----------|----------|-----------|----------|-----------|----------|-----------|----------|-----------|----------|-----------|----------|-----------|
| LSN | hex | <i>8</i> | | <i>9</i> | | <i>A</i> | | <i>B</i> | | <i>C</i> | | <i>D</i> | | <i>E</i> | | <i>F</i> | |
| 0000 | 0 | | 128 80 | | 144 90 | | 160 A0 | | 176 B0 | | 192 C0 | | 208 D0 | | 224 E0 | 0 | 240 F0 |
| 0001 | 1 | a | 129 81 | j | 145 91 | | 161 A1 | | 177 B1 | A | 193 C1 | J | 209 D1 | | 225 E1 | 1 | 241 F1 |
| 0010 | 2 | b | 130 82 | k | 146 92 | s | 162 A2 | | 178 B2 | B | 194 C2 | K | 210 D2 | S | 226 E2 | 2 | 242 F2 |
| 0011 | 3 | c | 131 83 | l | 147 93 | t | 163 A3 | | 179 B3 | C | 195 C3 | L | 211 D3 | T | 227 E3 | 3 | 243 F3 |
| 0100 | 4 | d | 132 84 | m | 148 94 | u | 164 A4 | | 180 B4 | D | 196 C4 | M | 212 D4 | U | 228 E4 | 4 | 244 F4 |
| 0101 | 5 | e | 133 85 | n | 149 95 | v | 165 A5 | | 181 B5 | E | 197 C5 | N | 213 D5 | V | 229 E5 | 5 | 245 F5 |
| 0110 | 6 | f | 134 86 | o | 150 96 | w | 166 A6 | | 182 B6 | F | 198 C6 | O | 214 D6 | W | 230 E6 | 6 | 246 F6 |
| 0111 | 7 | g | 135 87 | p | 151 97 | x | 167 A7 | | 183 B7 | G | 199 C7 | P | 215 D7 | X | 231 E7 | 7 | 247 F7 |
| 1000 | 8 | h | 136 88 | q | 152 98 | y | 168 A8 | | 184 B8 | H | 200 C8 | Q | 216 D8 | Y | 232 E8 | 8 | 248 F8 |
| 1001 | 9 | i | 137 89 | r | 153 99 | z | 169 A9 | 4 | 185 B9 | I | 201 C9 | R | 217 D9 | Z | 233 E9 | 9 | 249 F9 |
| 1010 | A | | 138 8A | | 154 9A | | 170 AA | | 186 BA | | 202 CA | | 218 DA | | 234 EA | | 250 FA |
| 1011 | B | | 139 8B | | 155 9B | | 171 AB | | 187 BB | | 203 CB | | 219 DB | | 235 EB | | 251 FB |
| 1100 | C | | 140 8C | | 156 9C | | 172 AC | | 188 BC | | 204 CC | | 220 DC | | 236 EC | | 252 FC |
| 1101 | D | | 141 8D | | 157 9D | | 173 AD | | 189 BD | | 205 CD | | 221 DD | | 237 ED | | 253 FD |
| 1110 | E | | 142 8E | | 158 9E | | 174 AE | | 190 BE | | 206 CE | | 222 DE | | 238 EE | | 254 FE |
| 1111 | F | | 143 8F | | 159 9F | | 175 AF | | 191 BF | | 207 CF | | 223 DF | | 111 6F | | 255 FF |



MÃ TRUYỀN (Transmission Code)

□ Mã ASCII

- American Standards Committee for Information Interchange
- Mã 7 bit để biểu diễn cho $2^7 = 128$ ký tự.
 - Thông thường bit thứ 8 được thêm vào như là bit kiểm tra

MÃ TRUYỀN

(Transmission Code)

| Binary | Dec | Hex | Abbr | Description | Binary | Dec | Hex | Abbr | Description |
|----------|-----|-----|------|---------------------|----------|-----|-----|------|------------------------------|
| 000 0000 | 0 | 0 | NUL | Null character | 001 0001 | 17 | 11 | DC1 | Device Control 1 (oft. XON) |
| 000 0001 | 1 | 1 | SOH | Start of Header | 001 0010 | 18 | 12 | DC2 | Device Control 2 |
| 000 0010 | 2 | 2 | STX | Start of Text | 001 0011 | 19 | 13 | DC3 | Device Control 3 (oft. XOFF) |
| 000 0011 | 3 | 3 | ETX | End of Text | 001 0100 | 20 | 14 | DC4 | Device Control 4 |
| 000 0100 | 4 | 4 | EOT | End of Transmission | 001 0101 | 21 | 15 | NAK | Negative Acknowledgement |
| 000 0101 | 5 | 5 | ENQ | Enquiry | 001 0110 | 22 | 16 | SYN | Synchronous Idle |
| 000 0110 | 6 | 6 | ACK | Acknowledgment | 001 0111 | 23 | 17 | ETB | End of Trans. Block |
| 000 0111 | 7 | 7 | BEL | Bell | 001 1000 | 24 | 18 | CAN | Cancel |
| 000 1000 | 8 | 8 | BS | Backspace[d][i] | 001 1001 | 25 | 19 | EM | End of Medium |
| 000 1001 | 9 | 9 | HT | Horizontal Tab | 001 1010 | 26 | 1A | SUB | Substitute |
| 000 1010 | 10 | 0A | LF | Line feed | 001 1011 | 27 | 1B | ESC | Escape[g] |
| 000 1011 | 11 | 0B | VT | Vertical Tab | 001 1100 | 28 | 1C | FS | File Separator |
| 000 1100 | 12 | 0C | FF | Form feed | 001 1101 | 29 | 1D | GS | Group Separator |
| 000 1101 | 13 | 0D | CR | Carriage return[h] | 001 1110 | 30 | 1E | RS | Record Separator |
| 000 1110 | 14 | 0E | SO | Shift Out | 001 1111 | 31 | 1F | US | Unit Separator |
| 000 1111 | 15 | 0F | SI | Shift In | 111 1111 | 127 | 7F | DEL | Delete[e][i] |
| 001 0000 | 16 | 10 | DLE | Data Link Escape | | | | | |

| binary | MSN | 0000 | | 0001 | | 0010 | | 0011 | | 0100 | | 0101 | | 0110 | | 0111 | |
|--------|-----|------------|----------|-----------------------|----------|--------------|----------|-------------|----------|----------|----------|----------|----------|----------|-----------|------------|-----------|
| LSN | hex | 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | |
| 0000 | 0 | NUL | 0 00 | DLE | 16 10 | SP | 32 20 | 0 | 48 30 | @ | 64 40 | P | 80 50 | , | 96 60 | P | 112 70 |
| 0001 | 1 | SOH | 1 01 | XON (DC1) | 17 11 | ! | 33 21 | 1 | 49 31 | A | 65 41 | Q | 81 51 | a | 97 61 | q | 113 71 |
| 0010 | 2 | STX | 2 02 | DC2 | 18 12 | " | 34 22 | 2 | 50 32 | B | 66 42 | R | 82 52 | b | 98 62 | r | 114 72 |
| 0011 | 3 | ETX | 3 03 | XOFF (DC2) | 19 13 | # | 35 23 | 3 | 51 33 | C | 67 43 | S | 83 53 | c | 99 63 | s | 115 73 |
| 0100 | 4 | EOT | 4 04 | DC4 | 20 14 | \$ | 36 24 | 4 | 52 34 | D | 68 44 | T | 84 54 | d | 100 64 | t | 116 74 |
| 0101 | 5 | ENQ | 5 05 | NAK | 21 15 | % | 37 25 | 5 | 53 35 | E | 69 45 | U | 85 55 | e | 101 65 | u | 117 75 |
| 0110 | 6 | ACK | 6 06 | SYN | 22 16 | & | 38 26 | 6 | 54 36 | F | 70 46 | V | 86 56 | f | 102 66 | v | 118 76 |
| 0111 | 7 | BEL | 7 07 | ETB | 23 17 | ' | 39 27 | 7 | 55 37 | G | 71 47 | W | 87 57 | g | 103 67 | w | 119 77 |
| 1000 | 8 | BS | 8 08 | CAN | 24 18 | (| 40 28 | 8 | 56 38 | H | 72 48 | X | 88 58 | h | 104 68 | x | 120 78 |
| 1001 | 9 | HT | 9 09 | EM | 25 19 |) | 41 29 | 9 | 57 39 | I | 73 49 | Y | 89 59 | i | 105 69 | y | 121 79 |
| 1010 | A | LF | 10 0A | SUB | 26 1A | * | 42 2A | : | 58 3A | J | 74 4A | Z | 90 5A | j | 106 6A | z | 122 7A |
| 1011 | B | VT | 11 0B | ESC | 27 1B | + | 43 2B | ; | 59 3B | K | 75 4B | I | 91 5B | k | 107 6B | { | 123 7B |
| 1100 | C | FF | 12 0C | FS | 28 1C | , | 44 2C | < | 60 3C | L | 76 4C | \ | 92 5C | l | 108 6C | | 124 7C |
| 1101 | D | CR | 13 0D | GS | 29 1D | - | 45 2D | = | 61 3D | M | 77 4D |] | 93 5D | m | 109 6D | } | 125 7D |
| 1110 | E | SO | 14 0E | RS | 30 1E | . | 46 2E | > | 62 3E | N | 78 4E | ^ | 94 5E | n | 110 6E | ~ | 126 7E |
| 1111 | F | SI | 15 0F | US | 31 1F | / | 47 2F | ? | 63 3F | O | 79 4F | _ | 95 5F | o | 111 6F | DEL | 127 7F |

MÃ TRUYỀN (Transmission Code)

| Binary | Dec | Hex | Glyph |
|-----------------|-----------|-----------|-----------|
| 010 0000 | 32 | 20 | SP |
| 010 0001 | 33 | 21 | ! |
| 010 0010 | 34 | 22 | " |
| 010 0011 | 35 | 23 | # |
| 010 0100 | 36 | 24 | \$ |
| 010 0101 | 37 | 25 | % |
| 010 0110 | 38 | 26 | & |
| 010 0111 | 39 | 27 | ' |
| 010 1000 | 40 | 28 | (|
| 010 1001 | 41 | 29 |) |
| 010 1010 | 42 | 2A | * |
| 010 1011 | 43 | 2B | + |
| 010 1100 | 44 | 2C | , |
| 010 1101 | 45 | 2D | - |
| 010 1110 | 46 | 2E | . |
| 010 1111 | 47 | 2F | / |

| Binary | Dec | Hex | Glyph |
|-----------------|-----------|-----------|----------|
| 011 0000 | 48 | 30 | 0 |
| 011 0001 | 49 | 31 | 1 |
| 011 0010 | 50 | 32 | 2 |
| 011 0011 | 51 | 33 | 3 |
| 011 0100 | 52 | 34 | 4 |
| 011 0101 | 53 | 35 | 5 |
| 011 0110 | 54 | 36 | 6 |
| 011 0111 | 55 | 37 | 7 |
| 011 1000 | 56 | 38 | 8 |
| 011 1001 | 57 | 39 | 9 |
| 011 1010 | 58 | 3A | : |
| 011 1011 | 59 | 3B | ; |
| 011 1100 | 60 | 3C | < |
| 011 1101 | 61 | 3D | = |
| 011 1110 | 62 | 3E | > |
| 011 1111 | 63 | 3F | ? |

MÃ TRUYỀN (Transmission Code)

| Dec | Hex | Glyph |
|-----------|-----------|----------|
| 64 | 40 | @ |
| 65 | 41 | A |
| 66 | 42 | B |
| 67 | 43 | C |
| 68 | 44 | D |
| 69 | 45 | E |
| 70 | 46 | F |
| 71 | 47 | G |
| 72 | 48 | H |
| 73 | 49 | I |
| 74 | 4A | J |
| 75 | 4B | K |
| 76 | 4C | L |
| 77 | 4D | M |
| 78 | 4E | N |
| 79 | 4F | O |

| Dec | Hex | Glyph |
|-----|-----|-------|
| 80 | 50 | P |
| 81 | 51 | Q |
| 82 | 52 | R |
| 83 | 53 | S |
| 84 | 54 | T |
| 85 | 55 | U |
| 86 | 56 | V |
| 87 | 57 | W |
| 88 | 58 | X |
| 89 | 59 | Y |
| 90 | 5A | Z |
| 91 | 5B | [|
| 92 | 5C | \ |
| 93 | 5D |] |
| 94 | 5E | ^ |
| 95 | 5F | _ |

| Dec | Hex | Glyph |
|-----------|-----------|----------|
| 96 | 60 | ` |
| 97 | 61 | a |
| 98 | 62 | b |
| 99 | 63 | c |
| 100 | 64 | d |
| 101 | 65 | e |
| 102 | 66 | f |
| 103 | 67 | g |
| 104 | 68 | h |
| 105 | 69 | i |
| 106 | 6A | j |
| 107 | 6B | k |
| 108 | 6C | l |
| 109 | 6D | m |
| 110 | 6E | n |
| 111 | 6F | o |

| Dec | Hex | Glyph |
|-----|-----|-------|
| 112 | 70 | p |
| 113 | 71 | q |
| 114 | 72 | r |
| 115 | 73 | s |
| 116 | 74 | t |
| 117 | 75 | u |
| 118 | 76 | v |
| 119 | 77 | w |
| 120 | 78 | x |
| 121 | 79 | y |
| 122 | 7A | z |
| 123 | 7B | { |
| 124 | 7C | |
| 125 | 7D | } |
| 126 | 7E | ~ |

CẤP ĐƠN VỊ DỮ LIỆU (Data Unit)

□ Đơn vị cơ bản là byte, một byte gồm 8 bits

➤ $1\text{Kb} = 2^{10} \text{ bytes} = 1024 \text{ bytes}$

➤ $1\text{Mb} = 2^{10} \text{ Kb} = 1024 \text{ Kb}$

➤ $1\text{Gb} = 2^{10} \text{ Mb} = 1024 \text{ Mb}$

➤ $1\text{Tb} = 2^{10} \text{ Gb} = 1024 \text{ Gb}$

□ Nhóm ký tự lại thành một khối gọi là đóng gói dữ liệu.

Một khối dữ liệu như vậy gọi là một packet hay một frame

GIAO THỨC (Protocol)

- ❑ Là tập hợp các quy định liên quan đến các yếu tố kỹ thuật truyền số liệu, cụ thể hoá các công tác cần thiết và quy trình thực hiện



HOẠT ĐỘNG KẾT NỐI

- ❑ Điểm nối điểm (point to point): một đầu cuối số liệu chỉ làm việc với một đầu cuối khác tại một thời điểm
- ❑ Đa điểm (multi point): một đầu cuối số liệu có thể thông tin với các đầu cuối khác một cách đồng thời



NỘI DUNG

3.1 Các khái niệm cơ bản về truyền số liệu

3.2 Thông tin nối tiếp không đồng bộ

3.3 Thông tin nối tiếp đồng bộ

3.4 Mạch điều khiển truyền số liệu

THÔNG TIN NỐI TIẾP KHÔNG ĐỒNG BỘ

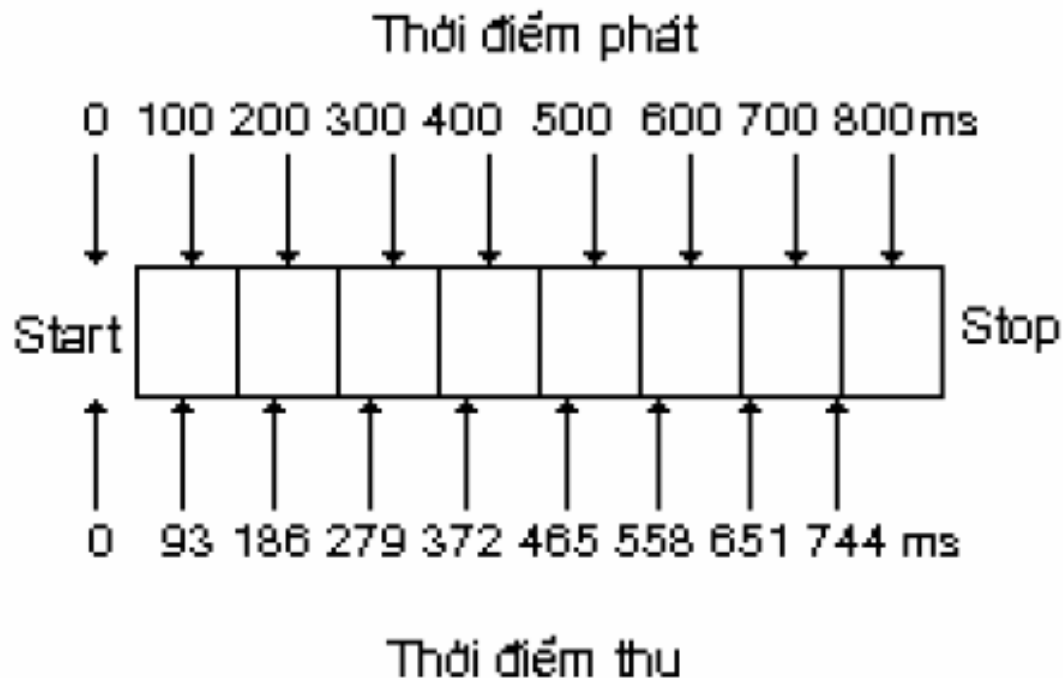
- ❑ Số liệu được truyền giữa 2 DTE dưới dạng chuỗi liên tiếp các bit gồm nhiều phần tử 8 bit gọi là byte hay ký tự dùng chế độ truyền đồng bộ hoặc bất đồng bộ
- ❑ Trong các DTE mỗi phần tử như vậy được lưu trữ và xử lý dưới dạng song song

THÔNG TIN NỐI TIẾP KHÔNG ĐỒNG BỘ

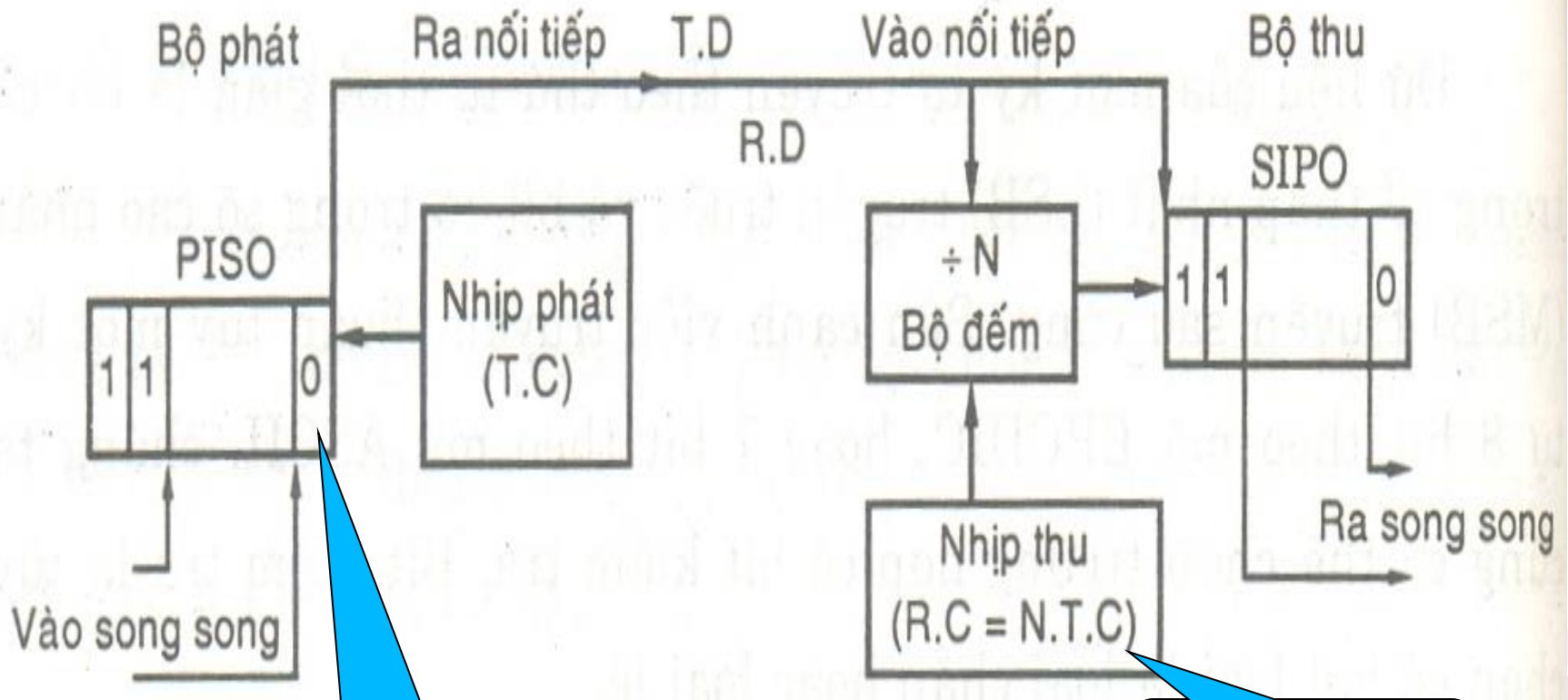
- ❑ Các mạch điều khiển trong DTE hình thành nên giao tiếp giữa thiết bị và liên kết dữ liệu nối tiếp phải thực thi các chức năng:
 - Chuyển từ song song → nối tiếp
 - Chuyển từ nối tiếp → song song
 - Tại máy thu phải đạt được sự đồng bộ bit, byte, frame
 - Cơ cấu phát sinh các ký số kiểm tra để phát hiện lỗi
- ❑ PISO (Parallel Input Serial Output)
- ❑ SIPO (Serial Input Parallel Output)

NGUYÊN TẮC ĐỒNG BỘ BIT

- Bộ thu lấy mẫu tại trung tâm của mỗi bit
- Mất đồng bộ bit, dữ liệu truyền và nhận sai lệch



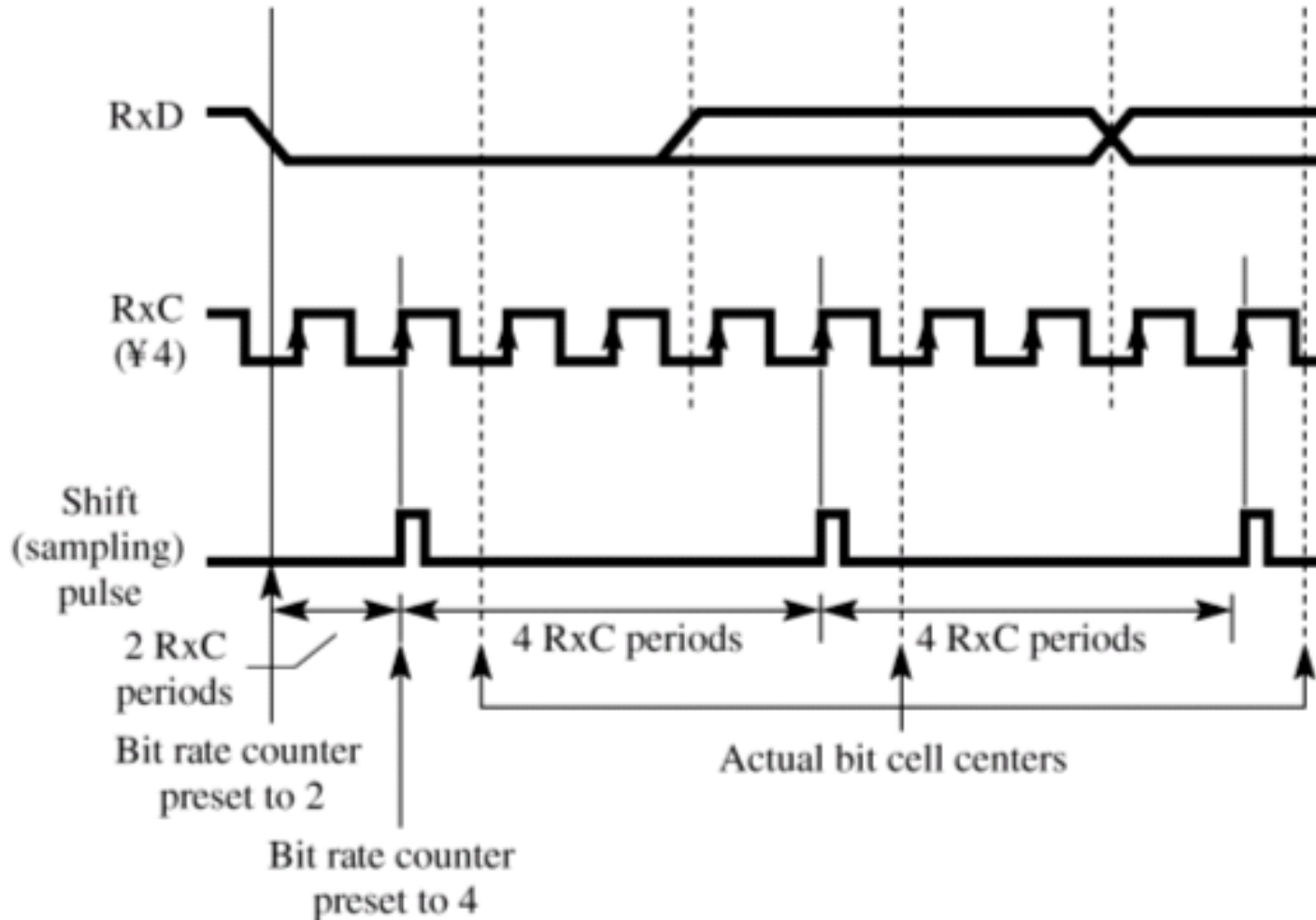
NGUYÊN TẮC ĐỒNG BỘ BIT



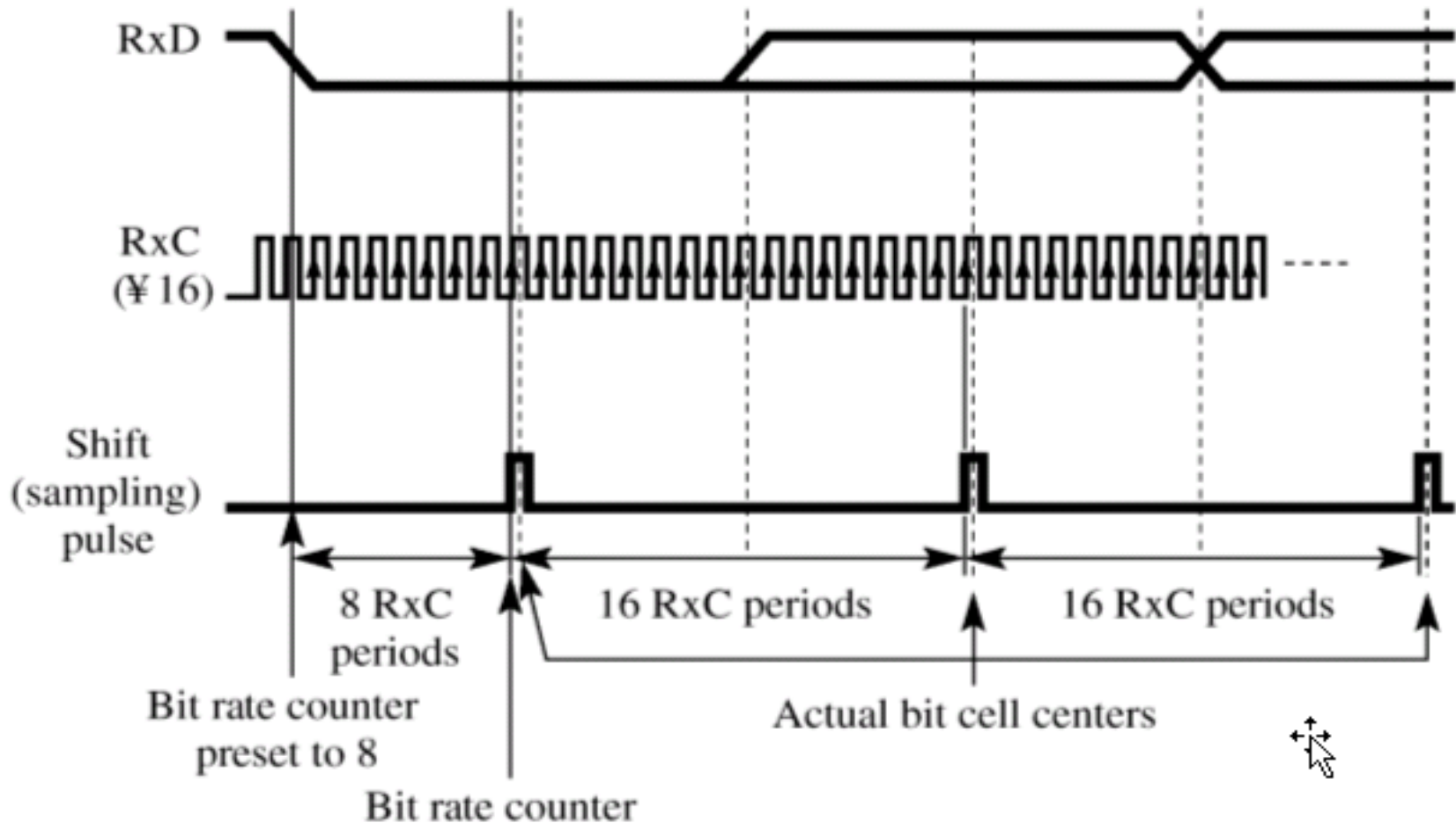
Parallel In
Serial Out

Nhịp thu gấp N
lần nhịp phát

ĐỒNG BỘ BIT – XUNG THU GẤP 4 LẦN XUNG PHÁT



ĐỒNG BỘ BIT – XUNG THU GẤP 16 LẦN XUNG PHÁT

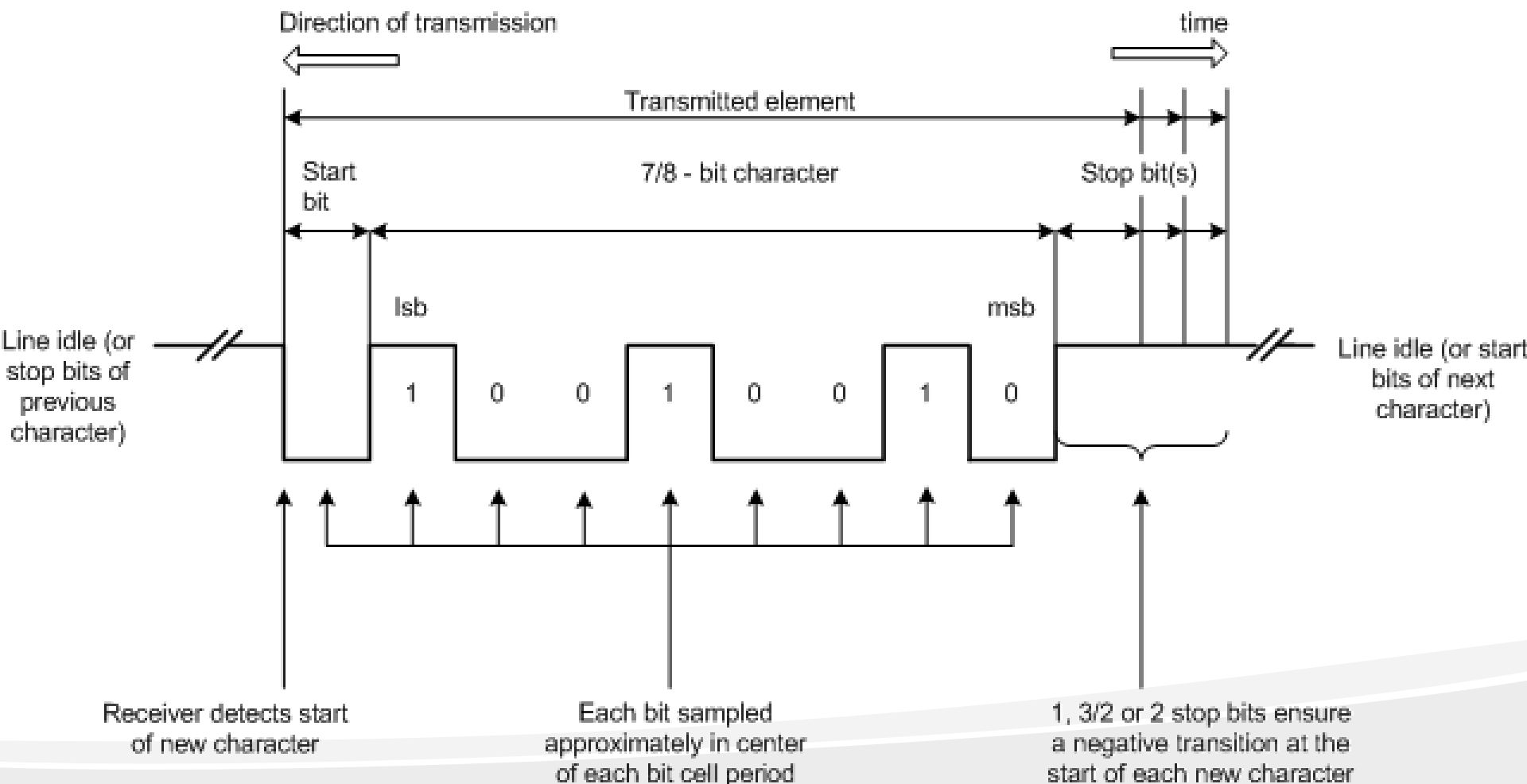




NGUYÊN TẮC ĐỒNG BỘ BYTE (KÝ TỰ)

- ❑ Mạch điều khiển truyền nhận được lập trình để hoạt động với số bit bằng nhau trong một ký tự
- ❑ Ký tự có thể 7 bits hoặc 8 bits được đồng bộ bằng cách thêm vào 1 bit biểu diễn sự bắt đầu của ký tự (start bit) và 1 hoặc 1.5 hoặc 2 bit biểu diễn sự kết thúc của 1 ký tự (stop bit)

NGUYÊN TẮC ĐỒNG BỘ BYTE (KÝ TỰ)





Start/stop bit trong truyền bất đồng bộ

- Phân biệt start bit của ký tự hiện hành và:
 - stop bit của ký tự trước
 - trạng thái rảnh (idle)
- Tối thiểu có một biến đổi ($1 \rightarrow 0 \rightarrow 1$) giữa các ký tự liên tiếp nhau
- Số stop bit nhiều hay ít tùy thuộc vào yêu cầu



MSB & LSB

- ❑ Bit có trọng số thấp nhất (LSB) được truyền trước, bit có trọng số cao nhất (MSB) được truyền sau cùng
 - LSB: Least Significant Bit
 - MSB: Most Significant Bit

NGUYÊN TẮC ĐỒNG BỘ KHUNG (FRAME)



- Các ký tự được truyền theo từng khối – khung tin (frame)
- Bộ thu cần biết lúc nào bắt đầu và lúc nào kết thúc một khung
- Đóng khung bằng ký tự STX (Start of Text) và ETX (End of Text)



- Nhận được STX: bắt đầu khung
- Tiếp tục nhận các ký tự cho đến khi nhận được ETX
 - Nếu nội dung của khối dữ liệu có chứa ký tự STX hay ETX???

Ký tự DLE

ASCII

- DLE (Data Link Escape) là ký tự thêm vào nhằm khắc phục vấn đề nêu trên
- Bắt đầu 1 khung là DLE STX
- Kết thúc 1 khung là DLE ETX
 - Nếu trong khối dữ liệu xuất hiện 2 ký tự liên tiếp DLE STX hay DLE ETX ???
- Nếu trong khối dữ liệu xuất hiện ký tự DLE thì thêm 1 ký tự DLE liền kề
 - Phía thu sẽ tự động loại bỏ 1 DLE



Ký tự DLE

ASCII



Chèn thêm



NỘI DUNG

3.1 Các khái niệm cơ bản về truyền số liệu

3.2 Thông tin nối tiếp không đồng bộ

3.3 Thông tin nối tiếp đồng bộ

3.4 Mạch điều khiển truyền số liệu



Khái quát

- Hiệu suất truyền bất đồng bộ thấp do dùng start và stop bit
- Đồng bộ bit của truyền bất đồng bộ trở nên thiếu tin cậy khi tăng tốc độ truyền
- => Sử dụng truyền đồng bộ
- Có 2 lược đồ truyền nối tiếp đồng bộ:
 - Truyền đồng bộ thiên hướng bit
 - Truyền đồng bộ thiên hướng ký tự

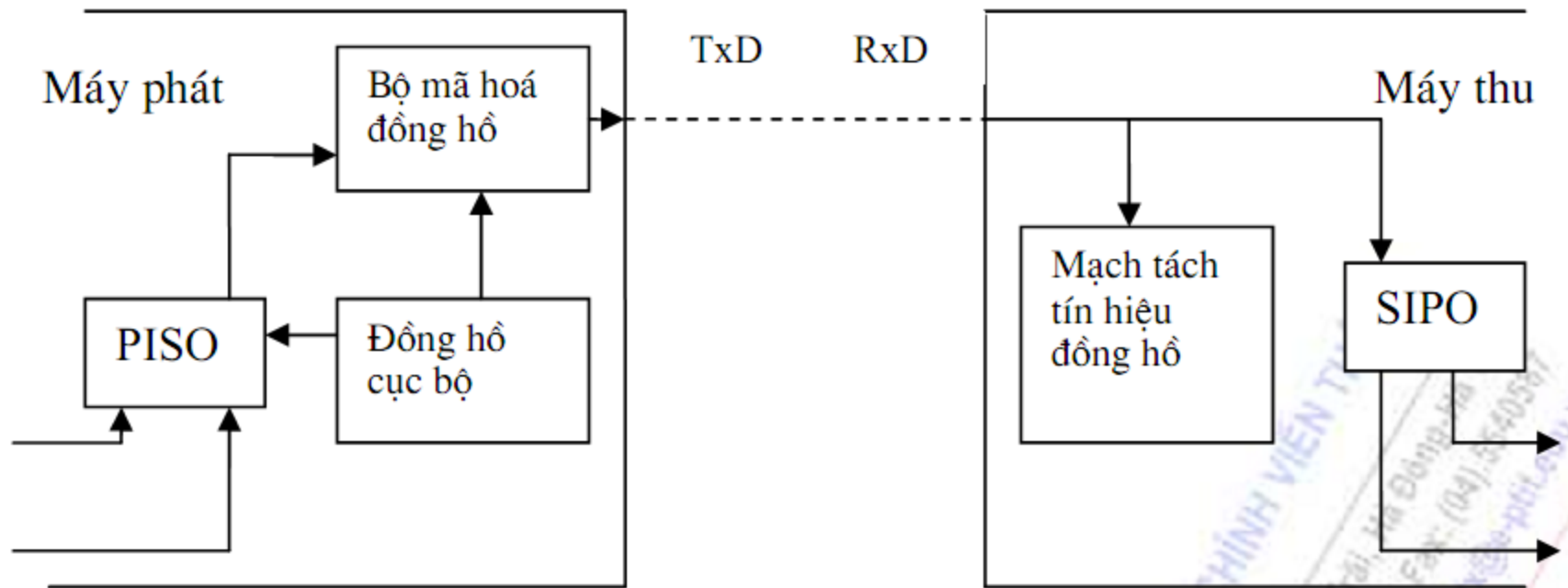


NGUYÊN TẮC ĐỒNG BỘ BIT

- Đồng hồ thu chạy đồng bộ với tín hiệu đến
- Không dùng start bit, stop bit
- Máy thu đồng bộ bit trong 2 cách
 - Nhúng thông tin định thời vào tín hiệu truyền (Sau đó máy thu sẽ tách tín hiệu định thời ra)
 - Máy thu có 1 đồng hồ cục bộ được giữ đồng bộ với tín hiệu thu nhờ vòng khoá pha số (Digital Phase Lock Loop)

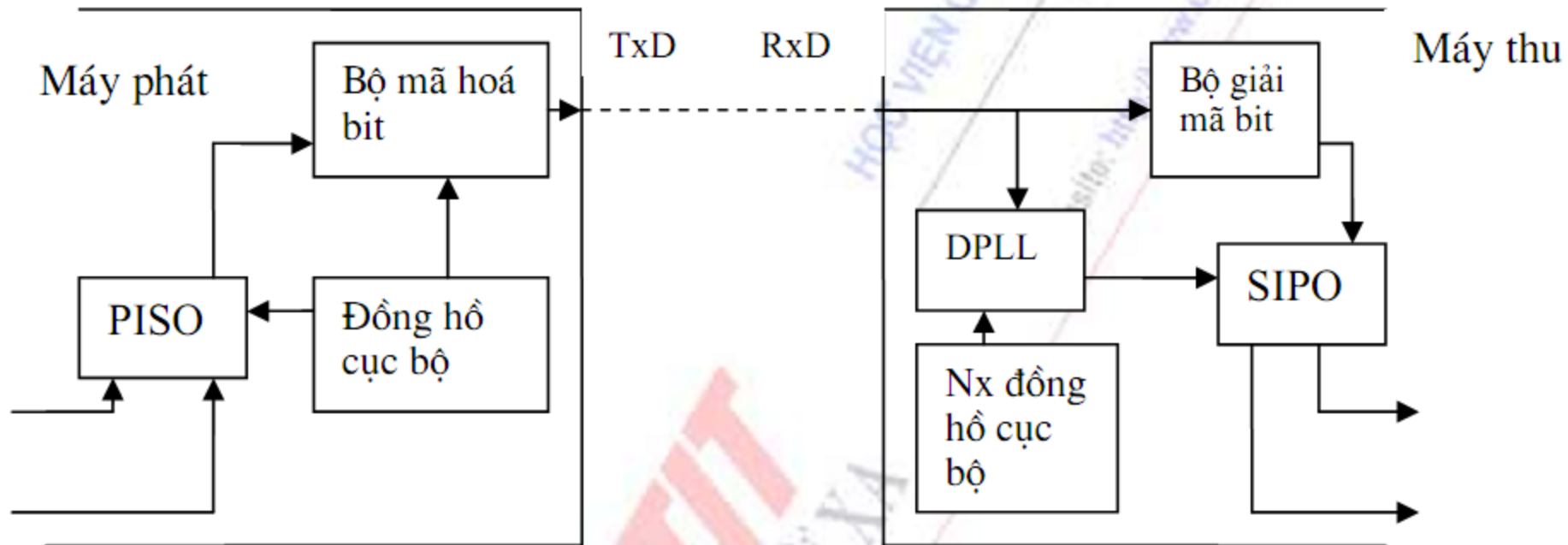
NGUYÊN TẮC ĐỒNG BỘ BIT

a) Mã hoá xung đồng hồ



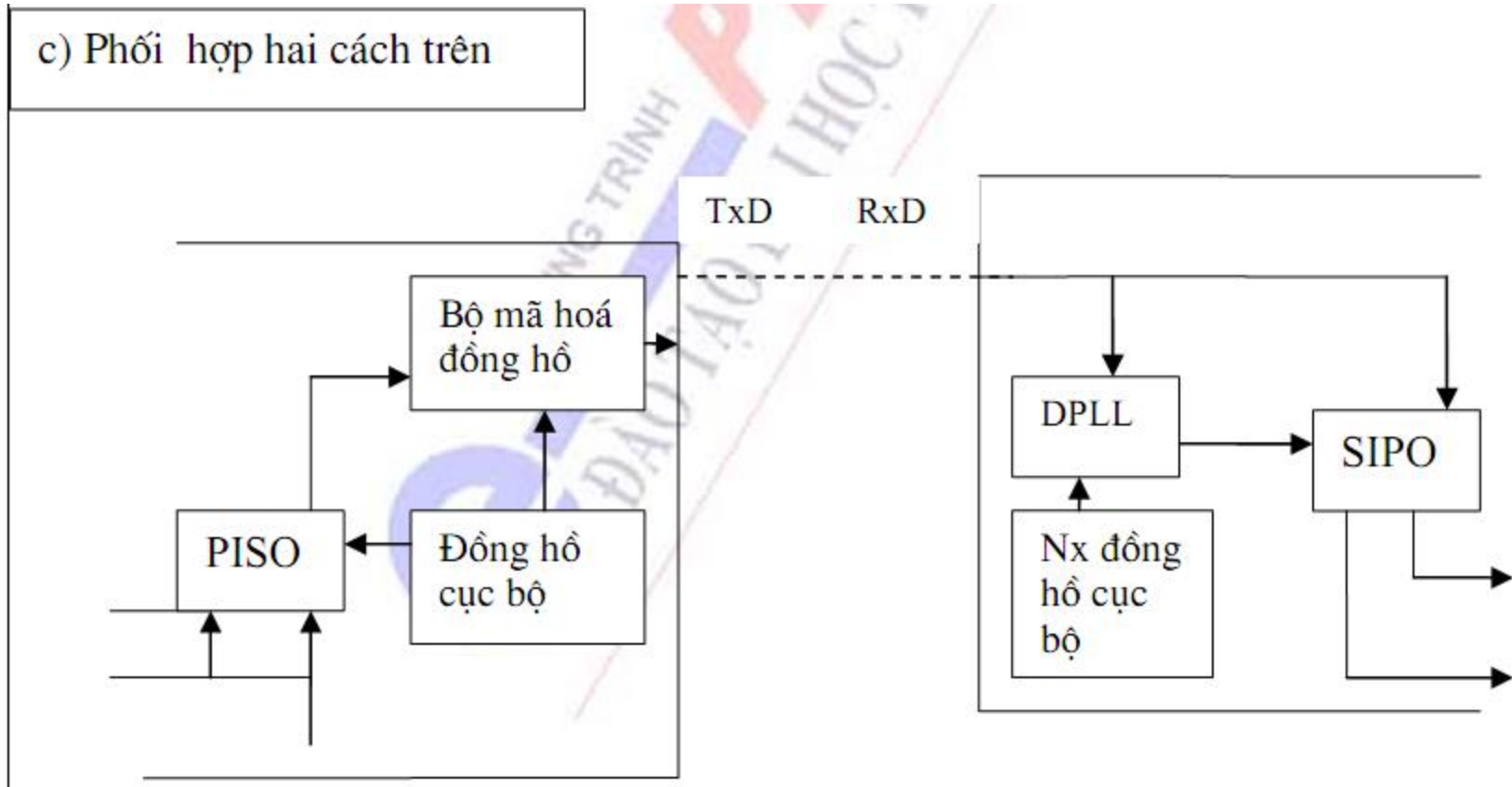
NGUYÊN TẮC ĐỒNG BỘ BIT

b) Dùng DPLL



NGUYÊN TẮC ĐỒNG BỘ BIT

c) Phối hợp hai cách trên

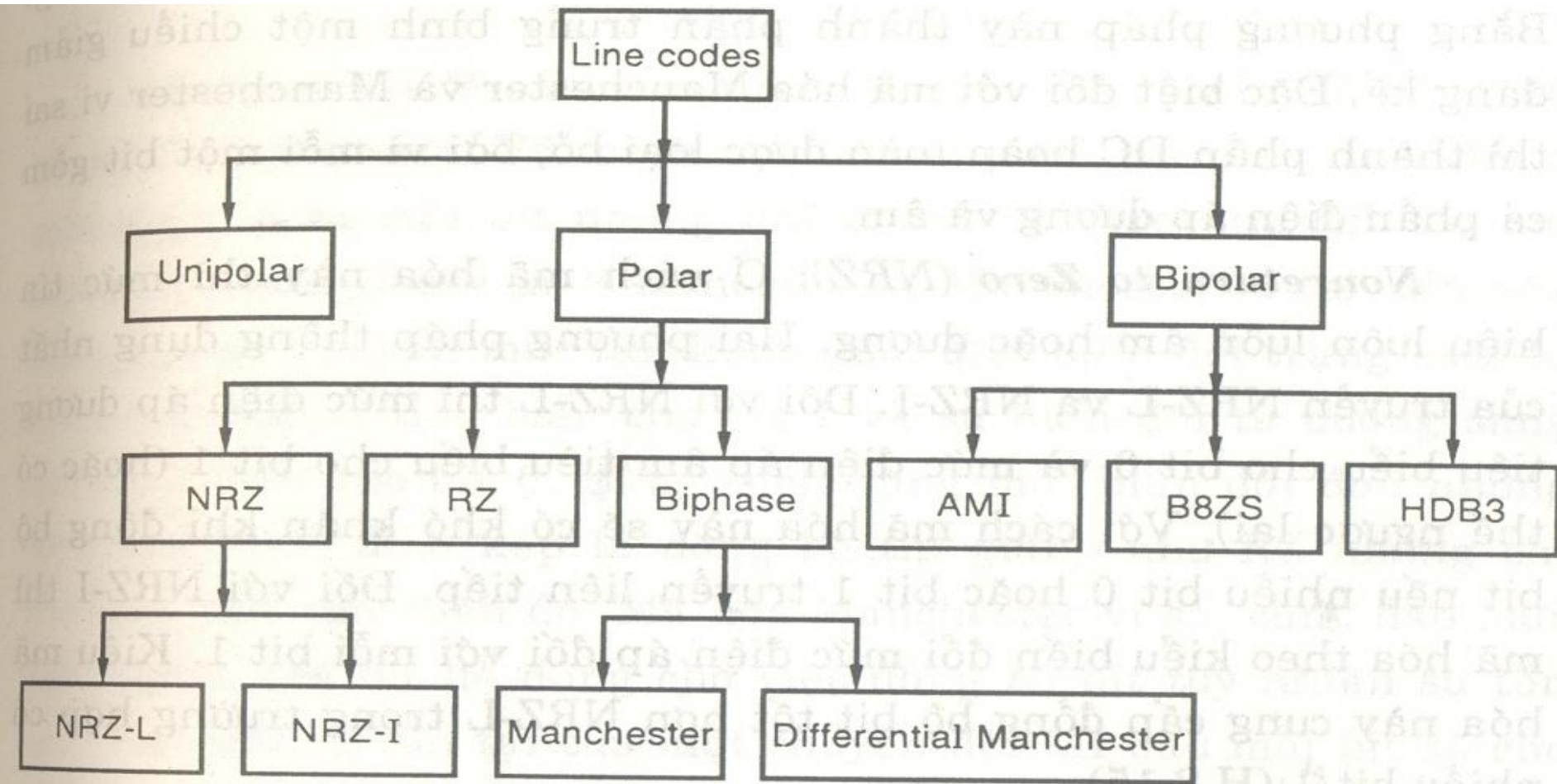




MÃ HOÁ XUNG ĐỒNG HỒ

- Mã hoá xung đồng hồ (mã hoá nhịp): clock encoding
- Nhịp được nhúng (mã hoá) vào trong tín hiệu phát và phía thu sẽ tách nhịp
- Cách mã hoá nhịp vào tín hiệu thường được thực hiện với mã đường dây hay còn gọi là biến đổi số - số

CÁC LOẠI MÃ ĐƯỜNG DÂY (LINE CODES)

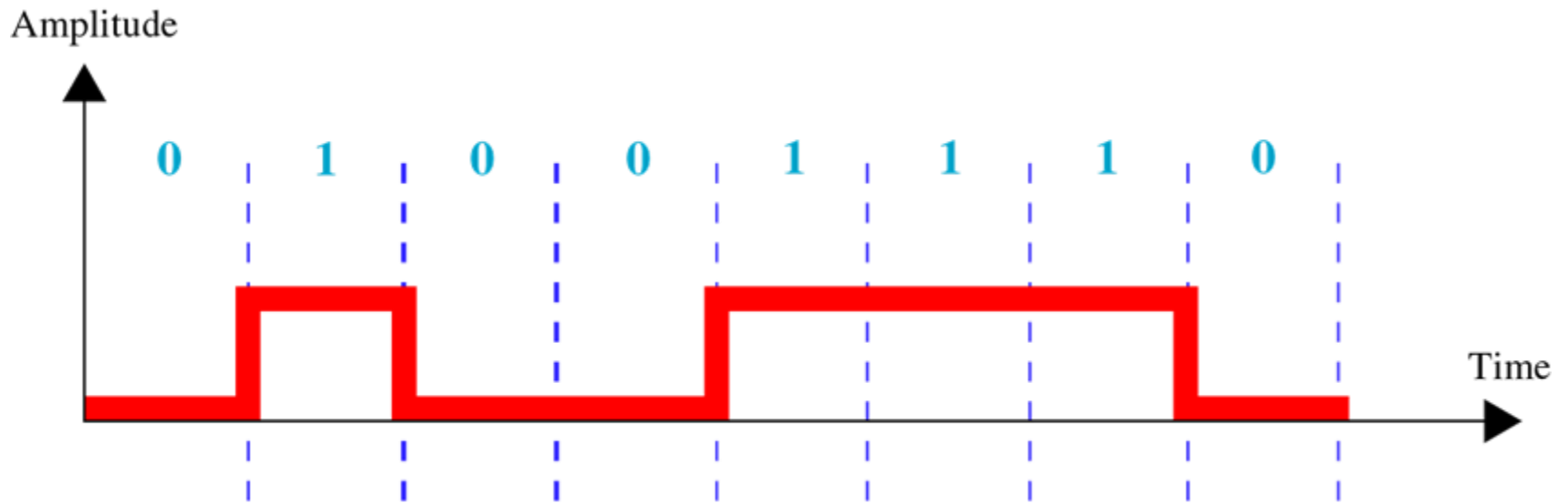




Unipolar

- Sử dụng các xung áp, gửi dọc theo dây dẫn
- Một mức điện áp cho bit 0 và 1 mức cho bit 1
 - Thông thường bit 1 có mức điện áp 1 cực tính (âm hoặc dương), bit 0 có mức điện áp 0
- Mức trung bình một chiều khác 0
- Khi tín hiệu phía thu không thay đổi, thì sẽ không xác định được thời điểm bắt đầu và kết thúc của 1 bit, dẫn đến sự đồng bộ bit kém

Unipolar



Polar

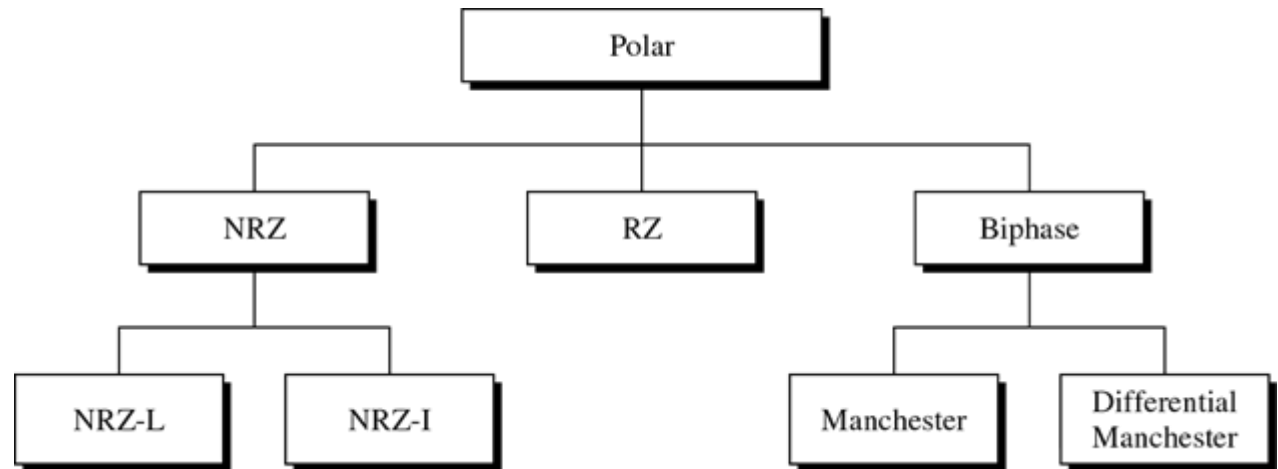
- Sử dụng 2 mức điện áp âm và dương
- Thành phần trung bình 1 chiều giảm đáng kể

- Bao gồm:

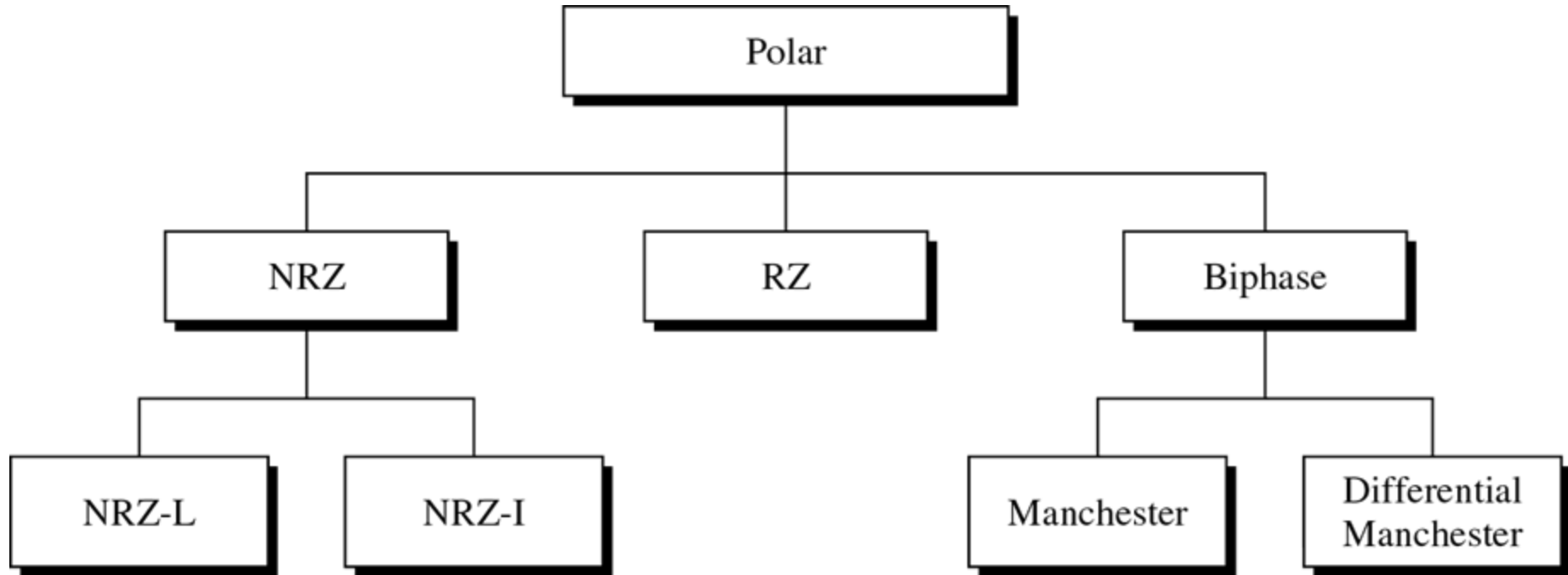
- NRZ

- RZ

- Biphase



Polar



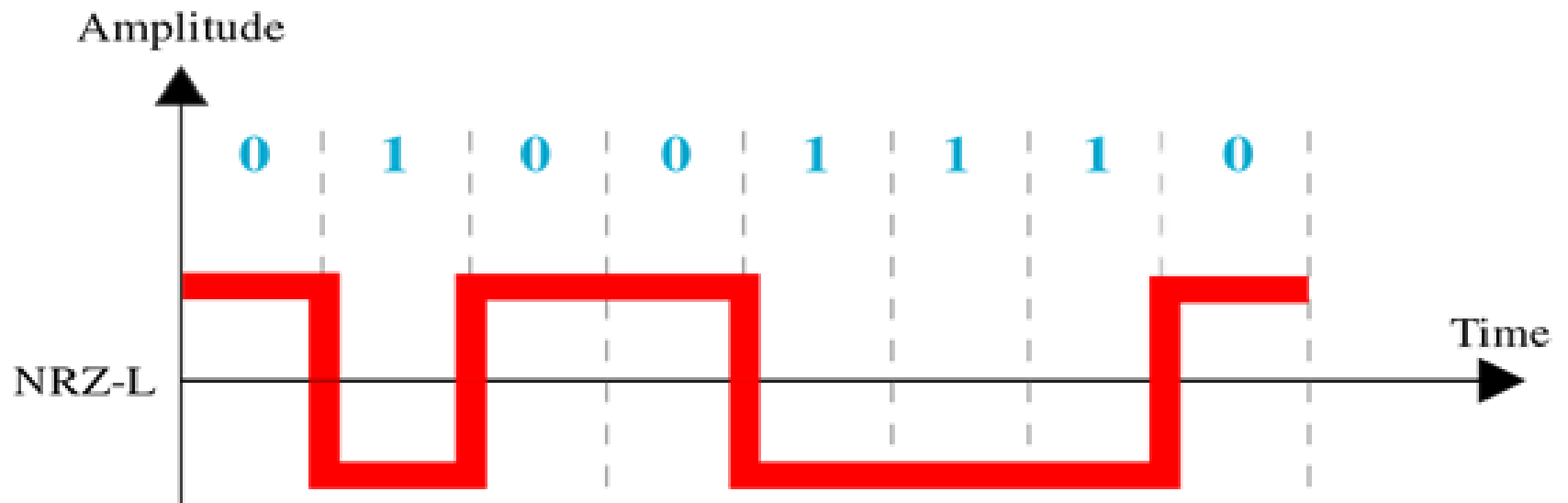


Polar NRZ

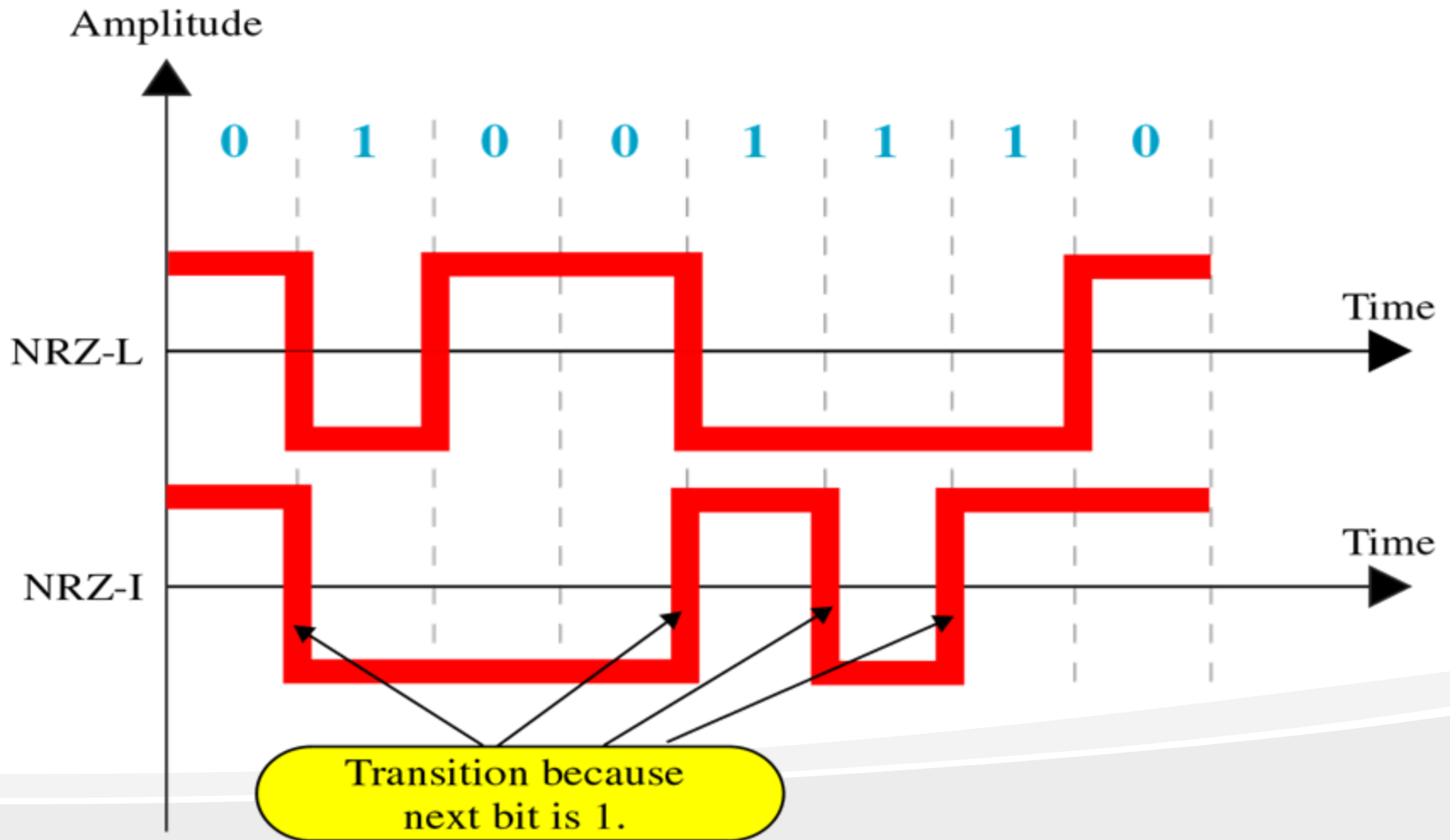
- ◆ Nonreturn to zero (NRZ): mức điện áp luôn âm hoặc dương
 - ◆ Nonreturn to zero – level (NRZ-L)
 - 2 mức điện áp khác nhau cho bit 1 và bit 0
 - Thông thường điện áp âm dùng cho bit 1, điện áp dương dùng cho bit 0 (hoặc có thể ngược lại)
 - ◆ Nonreturn to zero – Inverted (NRZ-I)
 - Bit 1 sẽ tạo một sự thay đổi mức điện áp
 - Bit 0 giữ nguyên mức điện áp



Polar NRZ



Polar NRZ





Ví dụ

◆ Vẽ giản đồ xung cho chuỗi
[LSB]0111111[MSB] theo mã
NRZ-L và NRZ-I.

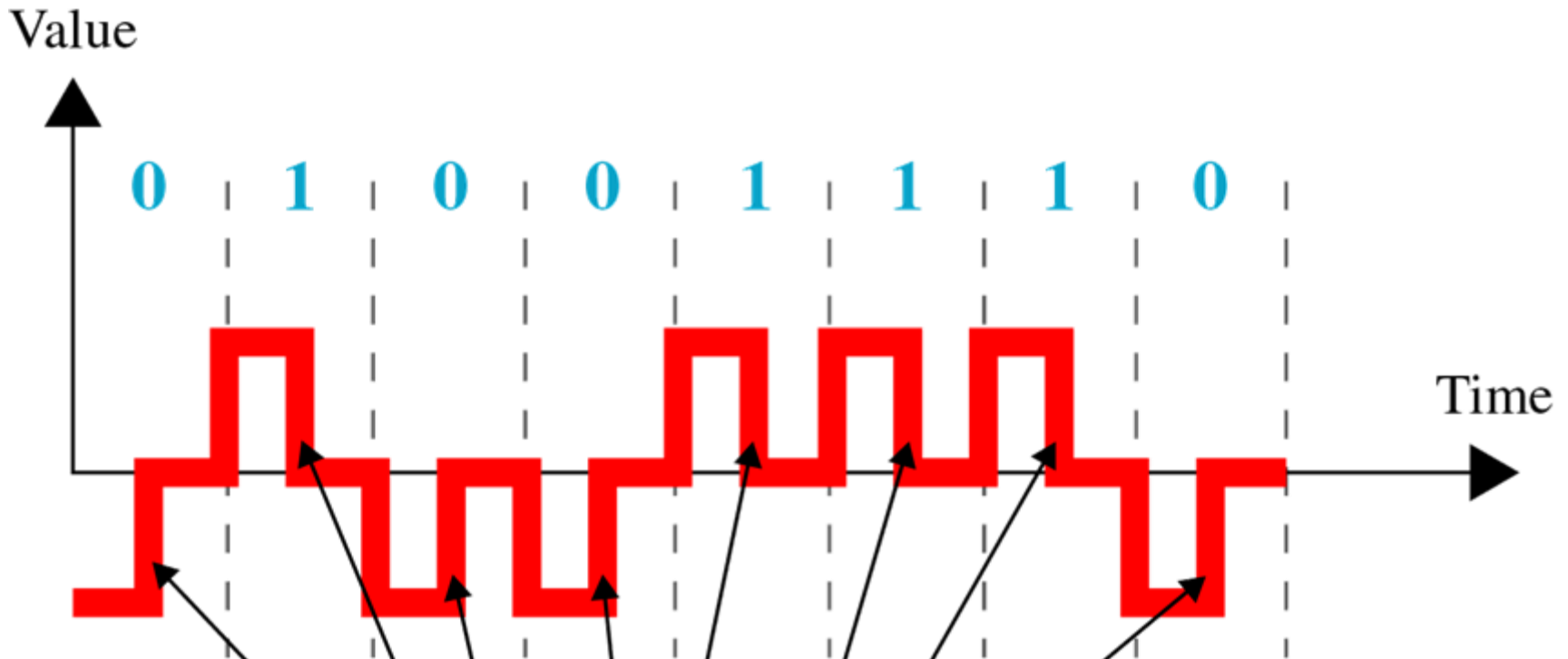


Return to zero (RZ):

- ◆ Mã hoá 3 mức: dương, âm, và zero
- ◆ Tín hiệu thay đổi trong mỗi khoảng bit
- ◆ Bit 1: thay đổi từ dương xuống zero
- ◆ Bit 0: thay đổi từ âm lên zero
- ◆ Khả năng đồng bộ bit rất hiệu quả tuy nhiên đòi hỏi một băng thông rộng

Return to zero (RZ)

◆ Return to zero (RZ):



These transitions can be used for synchronization.



Ví dụ

◆ Vẽ xung truyền chuỗi bit
[LSB]11100101[MSB]



Ví dụ

◆ Vẽ xung truyền chuỗi bit
[LSB]00111101[MSB]



Biphase

- Mã hóa giải quyết vấn đề đồng bộ tốt nhất
- Tín hiệu thay đổi ở điểm giữa nhưng không trở về zero như RZ
- Có 2 loại Biphase:
 - Manchester
 - Differential Manchester (Manchester vi sai)

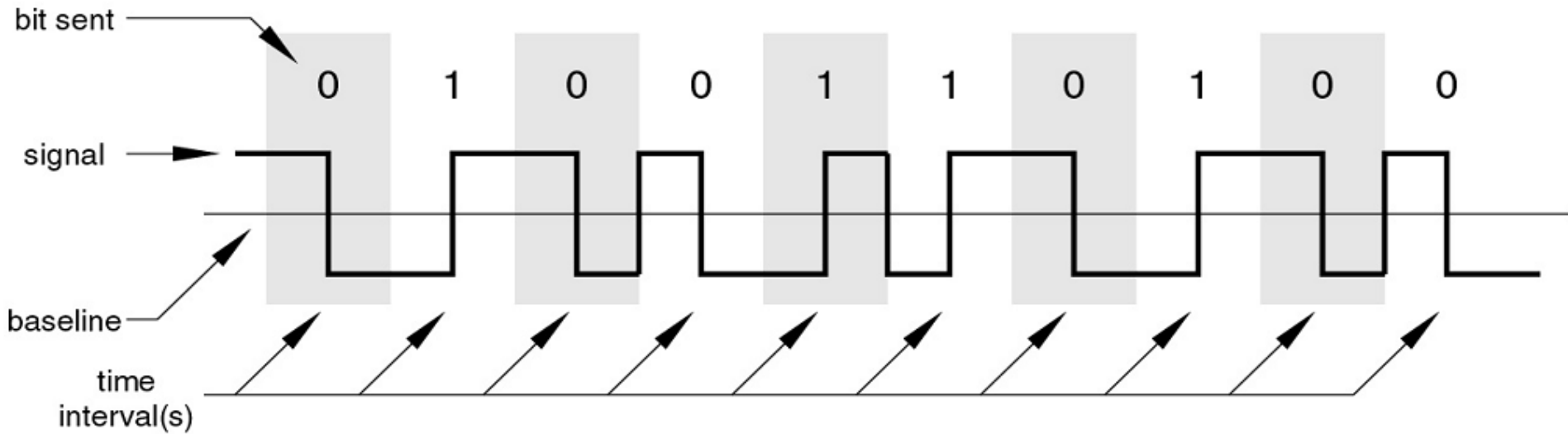


Manchester

- Mã hóa chuyển mức tại điểm giữa
- Bit 1 tương ứng với biến đổi trạng thái từ âm sang dương
- Bit 0 tương ứng với biến đổi từ dương sang âm



Manchester



Bit 1: - \rightarrow +

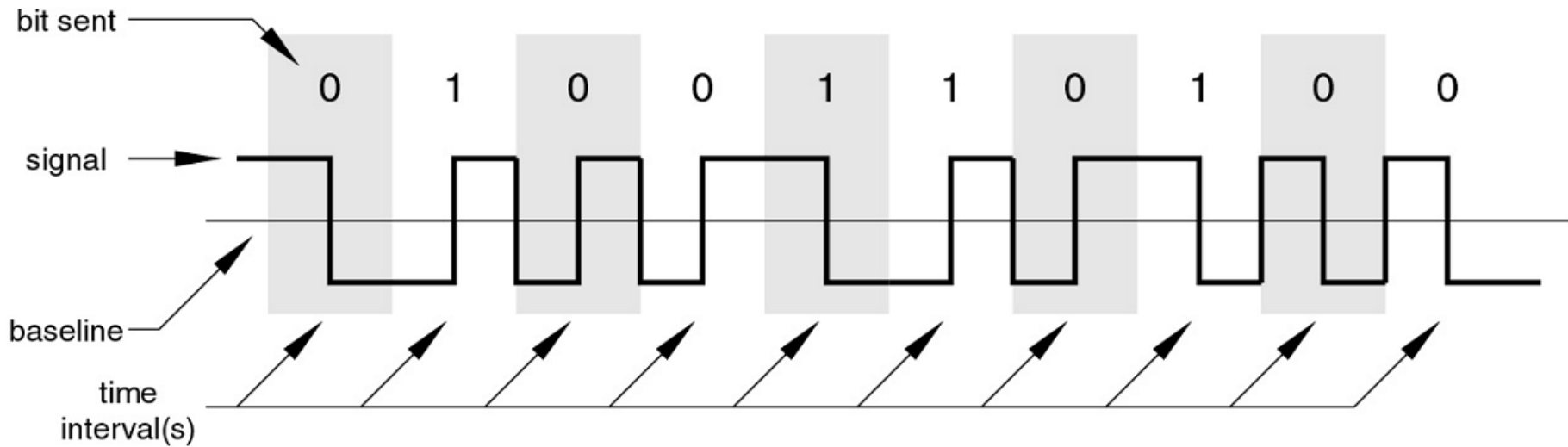
Bit 0: + \rightarrow -



Manchester vi sai

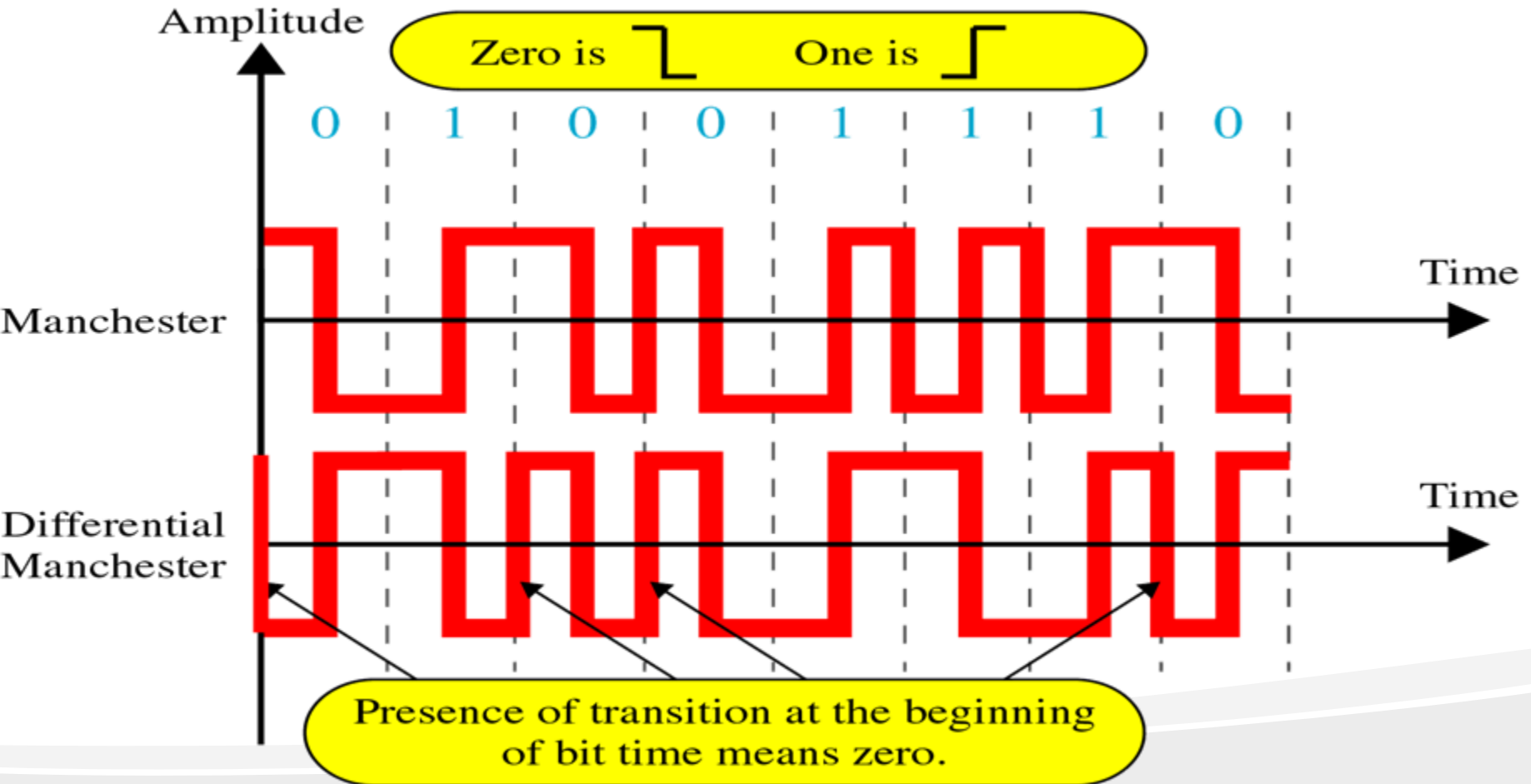
- Cũng sử dụng phương pháp đảo mức điện giữa của bit để dùng cho việc đồng bộ bit
- Phân biệt bit 0 / 1 dựa trên việc tồn tại hay không tồn tại chuyển đổi tại đầu mỗi bit
 - Bit 0: chuyển đổi
 - Bit 1: giữ nguyên

Manchester vi sai



Ví dụ

Manchester v.s Manchester vi sai





Bipolar

- Sử dụng 3 mức điện áp: dương, âm, zero
- Bit 0 tương ứng với mức zero
- Bit 1 tương ứng với thay đổi xen kẽ dương âm
- Ba loại thông dụng
 - AMI
 - B8ZS
 - HDB3



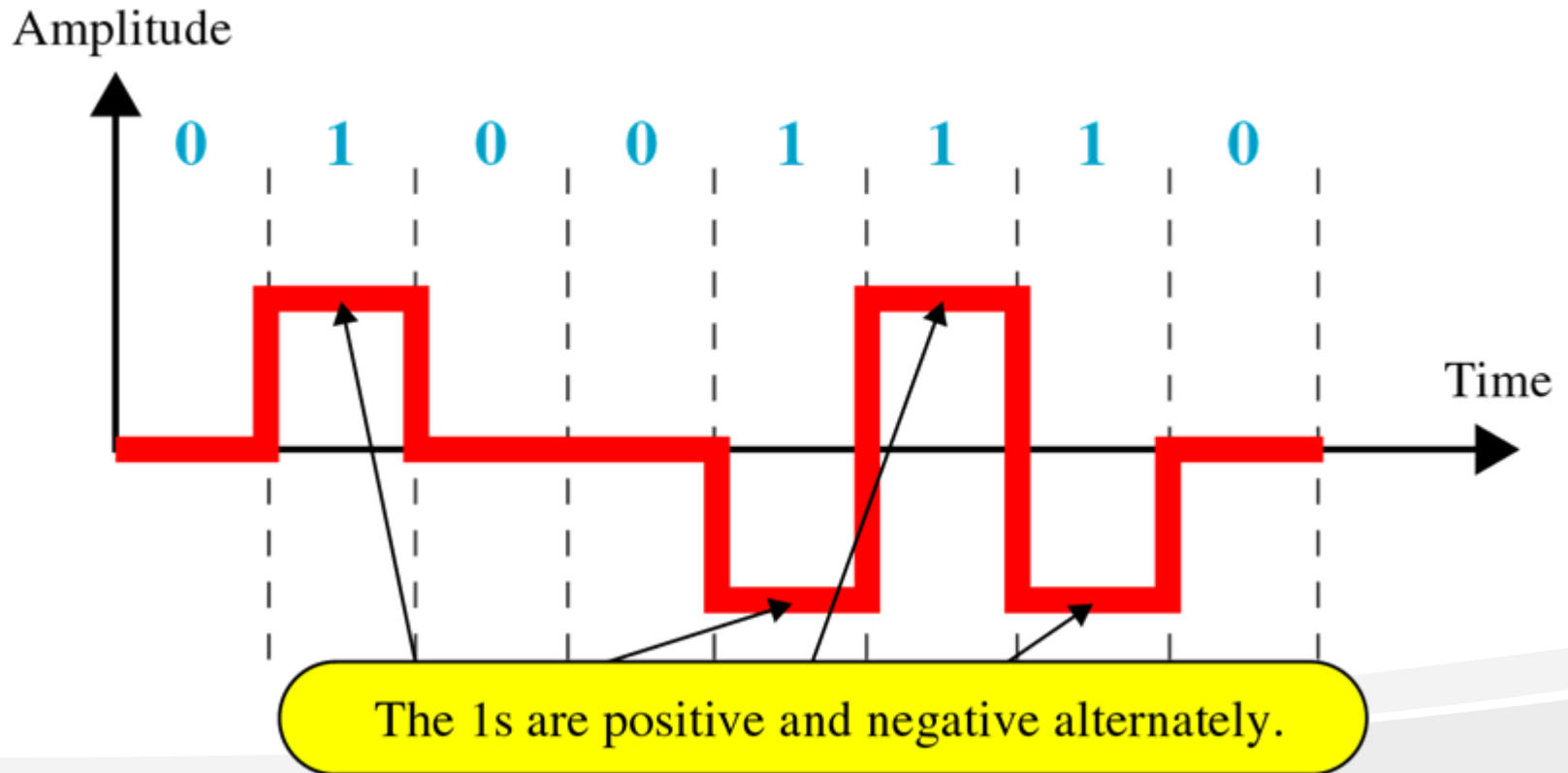
AMI

◆ AMI = Alternative Mark Inversion

- Bit 0 ở mức zero
- Bit 1 ở mức âm/dương: các bit 1 gần nhau nhận xen kẽ mức dương âm
- Đồng bộ bit tốt nếu chuỗi có nhiều bit 1, ngược lại không đảm bảo nếu gặp dãy bit 0 kéo dài

AMI

◆ AMI = Alternative Mark Inversion





Ví dụ

Vẽ xung truyền chuỗi bit
0010.0001.0010.1000

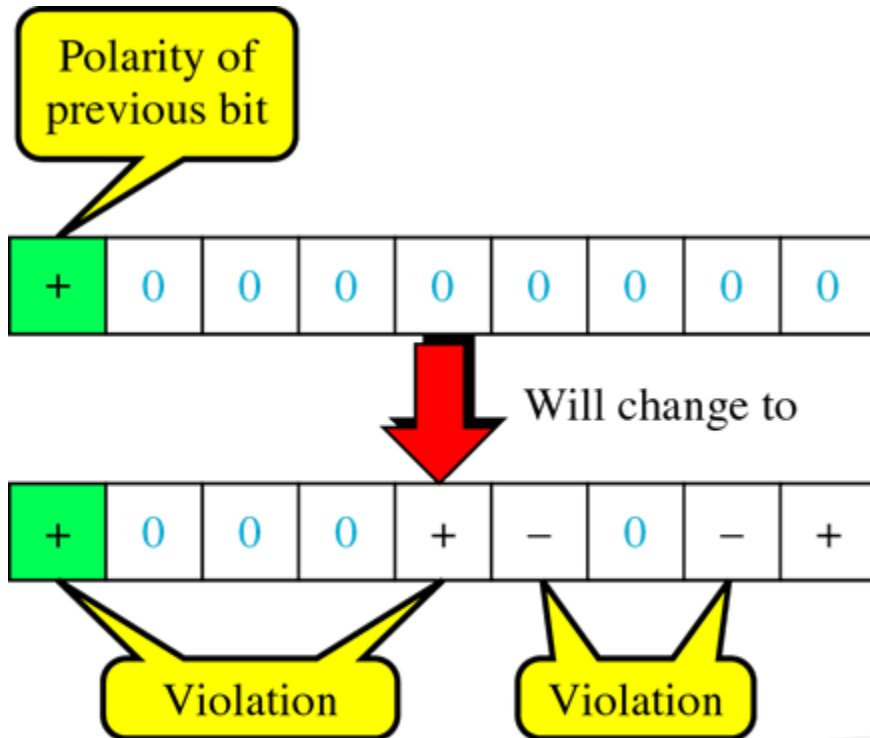
B8ZS

◆ B8ZS = Bipolar 8-zero Substitution

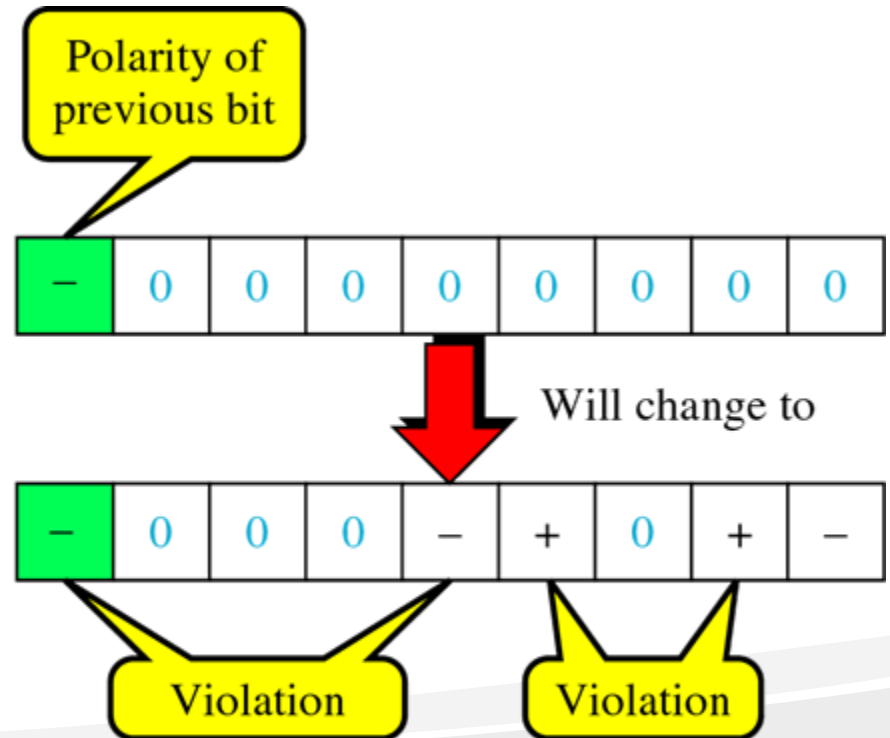
- Giải quyết vấn đề đồng bộ trong trường hợp có xuất hiện các chuỗi bit 0 kéo dài
- Tương tự AMI, có sự đảo cực tính mỗi khi gặp bit 1
- Mẫu 8 bit 0 liên tiếp được thay bằng mẫu 8 bit khác
- Tùy vào cực tính của bit nằm trước mẫu 8 bit 0 này mà sinh ra mẫu bit thay thế:
 - Nếu bit này có cực tính dương thì thay bằng dãy 0 0 0 + - 0 - +
 - Nếu bit này có cực tính âm thì thay bằng dãy 0 0 0 - + 0 + -

B8ZS

◆ B8ZS = Bipolar 8-zero Substitution



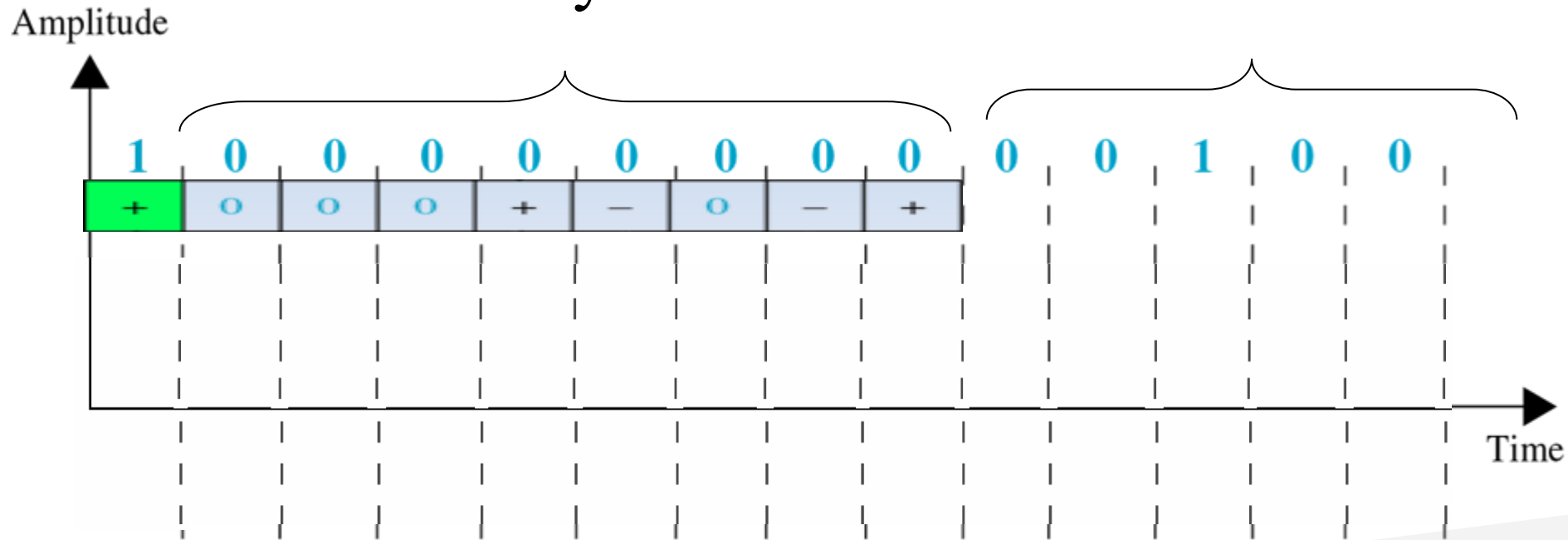
(a)



(b)

Ví dụ

- B8ZS
- Chuỗi bit truyền: 00.1000.0000.0001



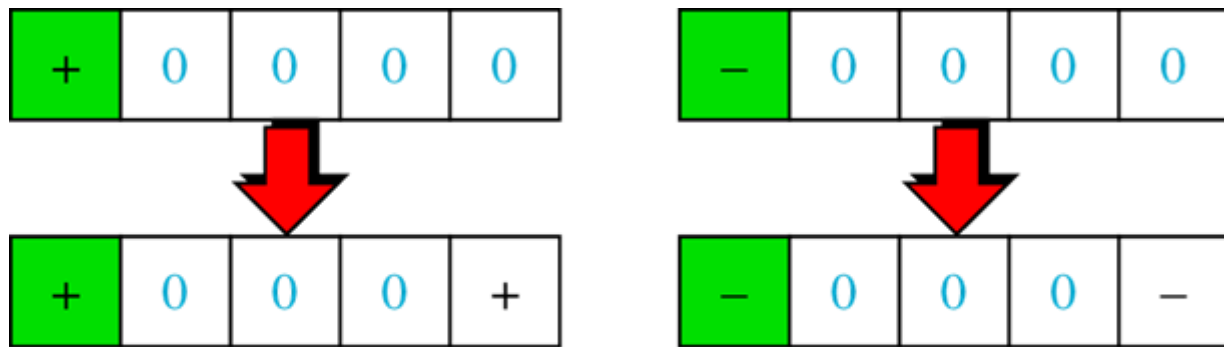


HDB3

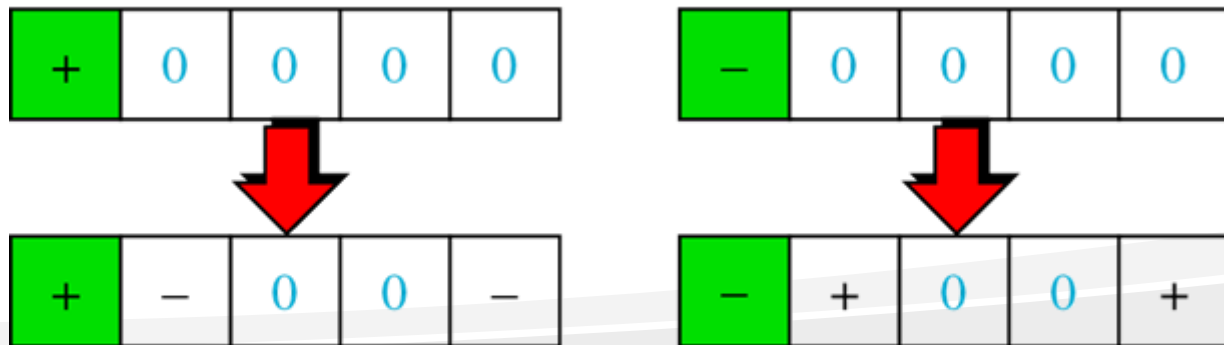
- ◆ HDB3 = High Density Bipolar 3
 - Mã hóa 4 bit 0 liên tiếp, dựa trên tổng số bit 1 kể từ lần thay thế sau cùng và cực tính của bit nằm liền trước
 - Nếu tổng số bit 1 trước đó là lẻ thì bit 0 thứ 4 sẽ chuyển thành bit vi phạm
 - Nếu tổng số bit 1 trước đó là chẵn thì bit 0 thứ nhất và thứ 4 sẽ chuyển thành bit vi phạm

HDB3

◆ HDB3 = High Density Bipolar 3



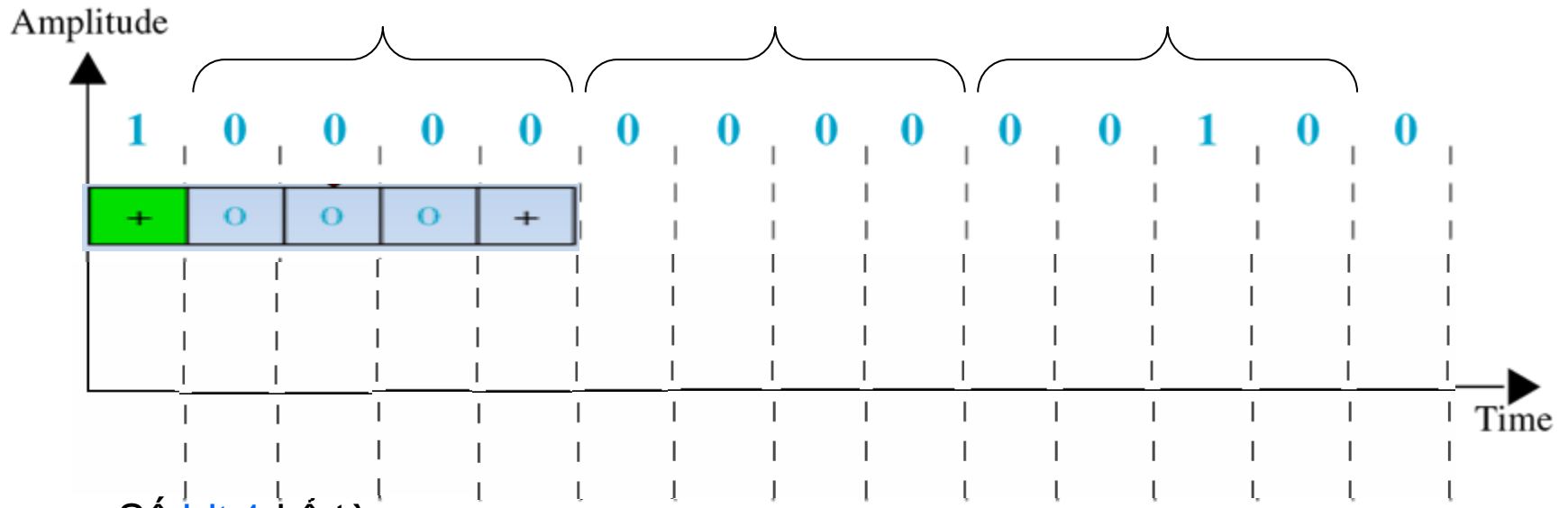
(a) If the number of 1s since the last substitution is odd



(b) If the number of 1s since the last substitution is even

Ví dụ

- HDB3
- Chuỗi bit truyền: 00.1000.0000.0001



Số bit 1 kể từ lần thay thế cuối cùng là **1**



Số bit 1 kể từ lần thay thế cuối cùng là **0**

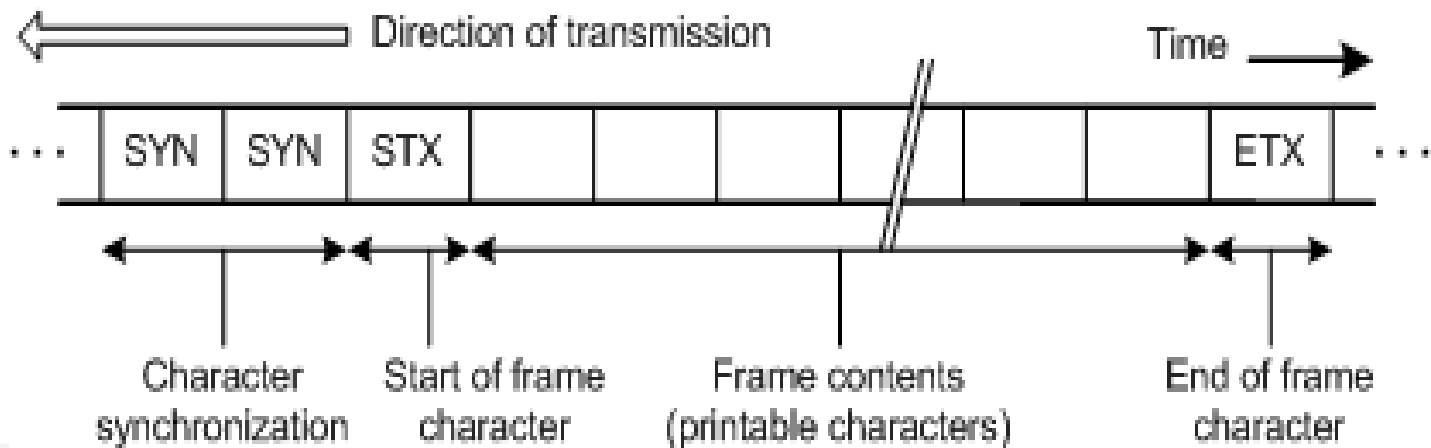


Nguyên lý kiểm soát đồng bộ

- Truyền đồng bộ định hướng ký tự
 - Character-oriented synchronous transmission
 - Dùng các ký tự điều khiển : SYN, STX, ETX, DLE.
- Truyền đồng bộ định hướng bit
 - Bit-oriented synchronous transmission
 - Dùng các mẫu bit điều khiển (flag byte or flag pattern)

Truyền đồng bộ định hướng ký tự

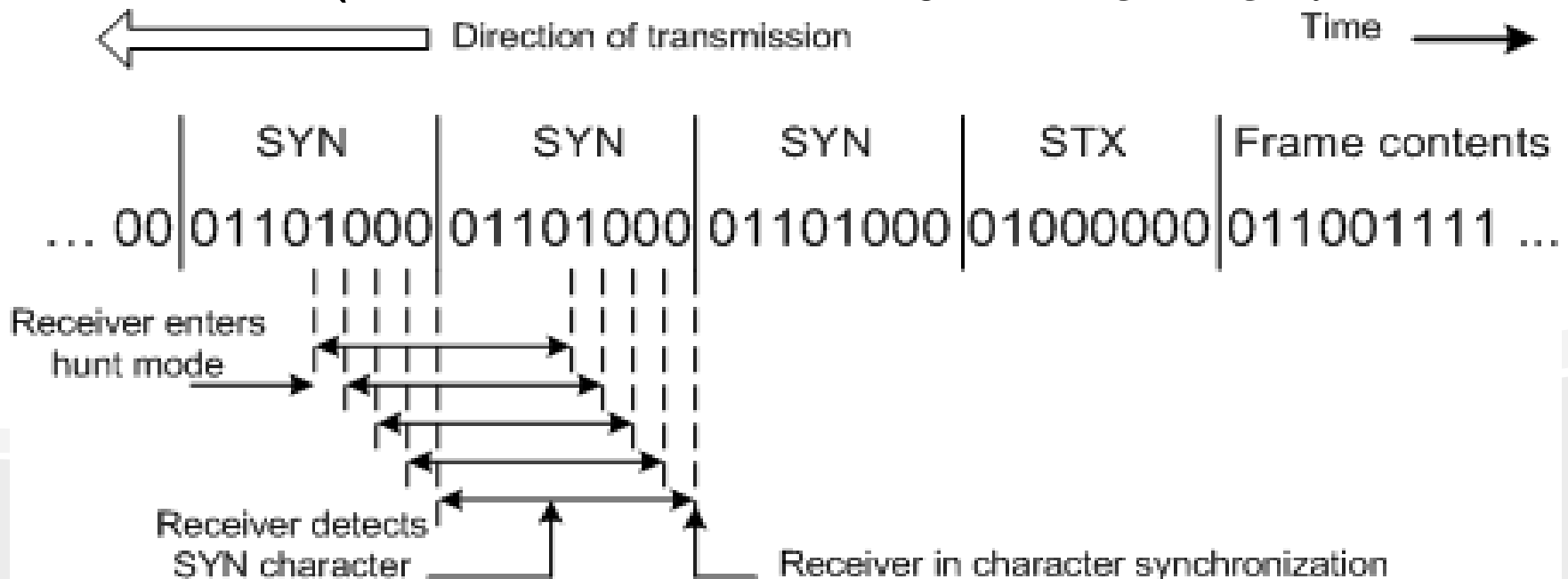
- Phía phát sẽ thêm vào 2 hoặc nhiều ký tự SYN trước và kết thúc mỗi khối ký tự
 - Nhằm duy trì đồng bộ bit
 - Đồng bộ ký tự



Dạng khung truyền định hướng ký tự

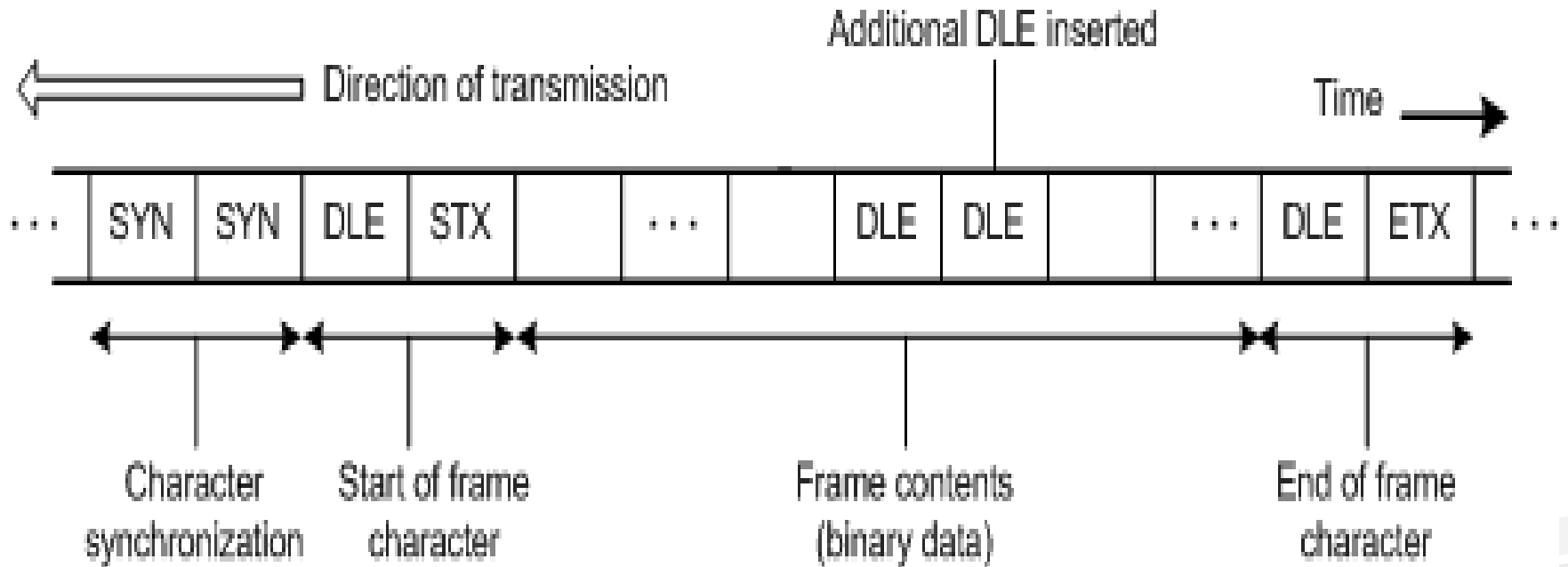
Truyền đồng bộ định hướng ký tự

- Khi máy thu đã được đồng bộ bit thì nó bắt đầu chế độ bắt số liệu
 - Dịch dòng bit trong một cửa sổ 8 bit khi tiếp nhận 1 bit mới
 - Kiểm tra xem 8 bit sau cùng có đúng bằng ký tự



Truyền đồng bộ định hướng ký tự

- Dữ liệu truyền được đóng gói bằng STX-ETX hoặc DLE STX – DLE ETX



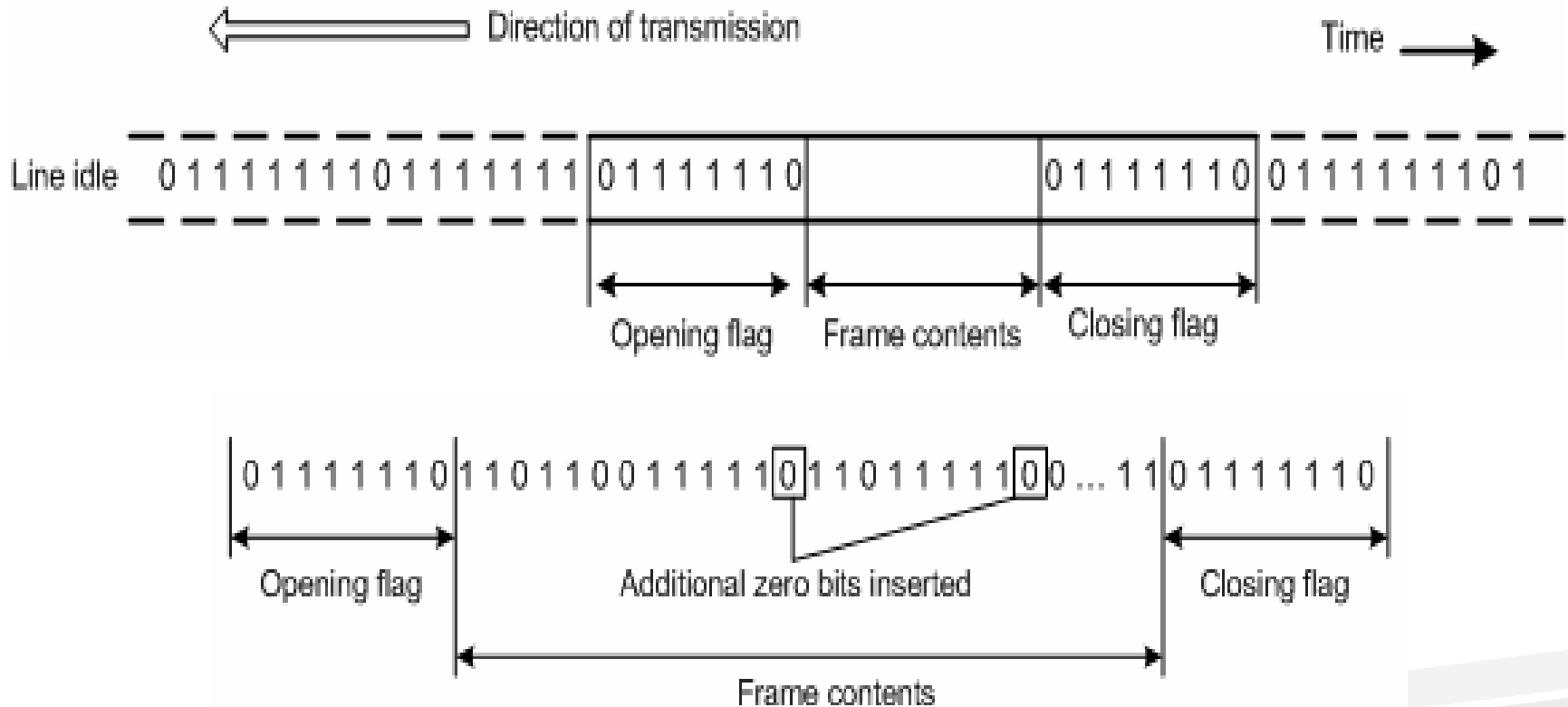
Định dạng khung dữ liệu trong suốt



Truyền đồng bộ định hướng bit

- Kiểu truyền định hướng ký tự với việc sử dụng ký tự SYN, STX, ETX, DLE có hiệu suất kém
 - Sử dụng định hướng bit
- Dùng chuỗi ký tự 8 bit 0111.1111 cho trạng thái đường dây rảnh
- Mẫu cờ 0111.1110 được dùng cho bắt đầu và kết thúc của một khung

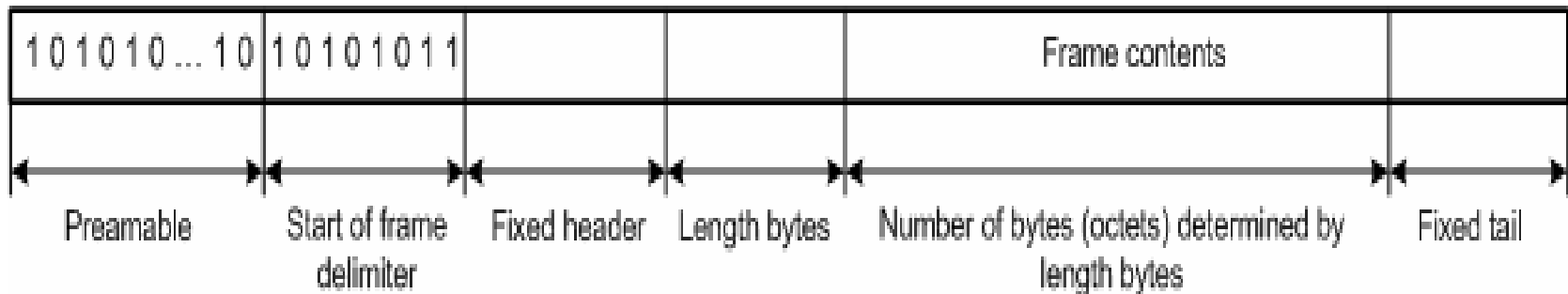
Truyền đồng bộ định hướng bit



Chuỗi 5 bit **1** liên tiếp -> chèn 1 bit **0**

Truyền đồng bộ định hướng bit

- Sử dụng mẫu bit preamble 10 bit 1010101010 để giúp các trạm có thể bám đồng bộ
- Kế đến là mẫu 8 bit 10101011 cho bắt đầu và kết thúc 1 khung

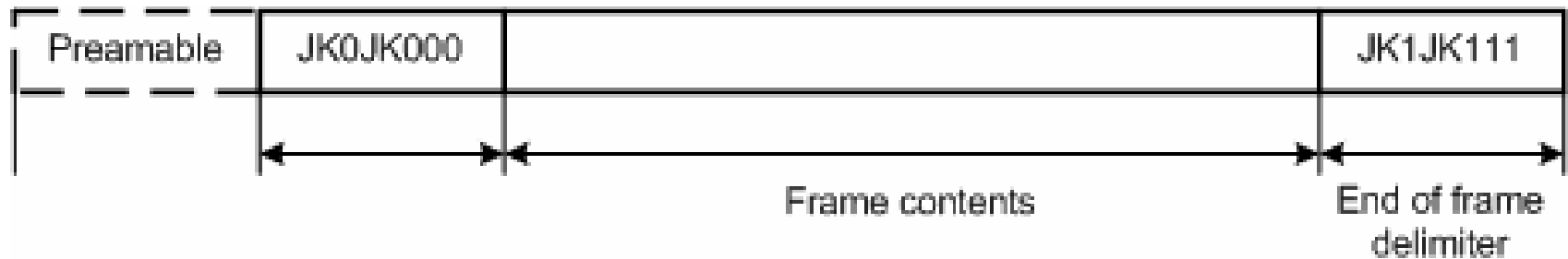


→ Dùng Flag + LEN

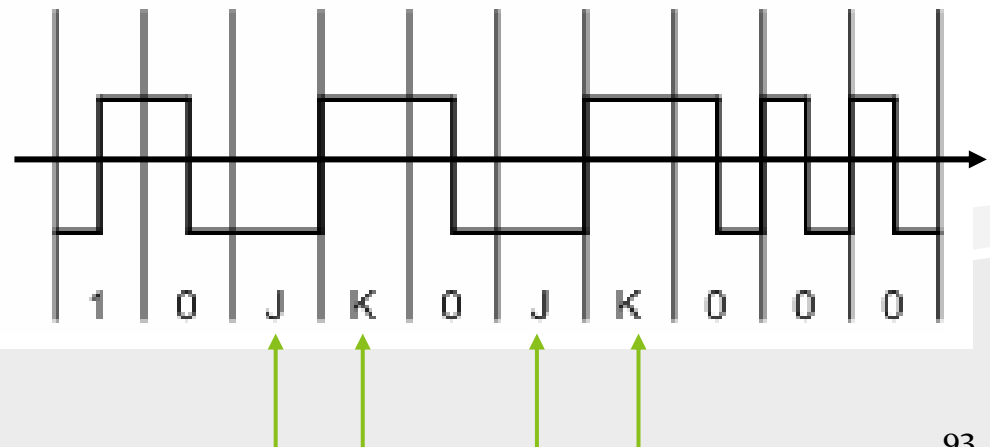
Truyền đồng bộ định hướng bit

→ Dùng vi phạm bit

- J bit: cùng mức điện áp bit trước đó
- K bit: Đảo mức điện áp bit trước đó



Manchester encoded bit stream with bit violations





NỘI DUNG

- 3.1 Các khái niệm cơ bản về truyền số liệu*
- 3.2 Thông tin nối tiếp không đồng bộ*
- 3.3 Thông tin nối tiếp đồng bộ*
- 3.4 Mạch điều khiển truyền số liệu*