www.mientayvn.com

Dịch tiếng anh chuyên nghành khoa học tự nhiên và kĩ thuật.

Dịch các bài giảng trong chương trình học liệu mở của học viện MIT, Yale.

Tìm và dịch tài liệu phục vụ cho sinh viên làm seminer, luận văn.

Tại sao mọi thứ đều miễn phí và chuyên nghiệp ???

Trao đổi trực tuyến tại:

http://www.mientayvn.com/chat_box_li.html

CHƯƠNG 3

TRUYỀN DỮ LIỆU

NỘI DUNG CHÍNH

- Cấu trúc của hai bảng mã ký tự EBCDIC & ASCII/IA5.
- So sánh 2 sơ đồ điều khiển truyền dẫn: SYN & ASYN.
- Mô tả các phương thức đồng bộ bit (clock), character (byte) của mode ASYN và các phương pháp mã hoá tương ứng.
- Mô tả kỹ thuật đồng bộ khung trong truyền dẫn ASYN & SYN hướng ký tự.
- Sử dụng các ký tự và bit chèn thêm phục vụ đồng bộ.
- Các kỹ thuật đồng bộ bit trong truyền dẫn đồng bộ.
- Các kỹ thuật phát hiện lỗi, nén dữ liệu.
- Nguyên lý hoạt động của các loại thiết bị ghép kênh.

GIỚI THIỆU

- Data được dùng để chỉ 1 (vài) khối tài liệu đã số hoá.
- Thông tin được hiểu là tin tức chứa trong data đang được truyền đi.
- Mất dữ liệu, ngay cả 1 bit cũng gây lỗi nên cần phải điều khiển lỗi.
- Điều khiển luồng là cần thiết.
- Mã hoá là công cụ phục vụ điều khiển lỗi.
- Môi trường truyền dẫn bit nối tiếp được sử dụng để kết nối các DTE.

I. CƠ SỞ CỦA TRUYỀN SỐ LIỆU

- Có một số chuẩn mã hóa cho dữ liệu ký tự (text):

1. EBCDIC – Exteded Binary Code Decimal Interchange Code

2. ASCII – American Standard Code for Information Interchange (IA5 - International Alphabet Number 5)

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	? (cond)	74 4A	!	90 5A	3	106 6A	:	122 7A
1011	B	VT	11 0B	CU1	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

Bảng mã EBCDIC

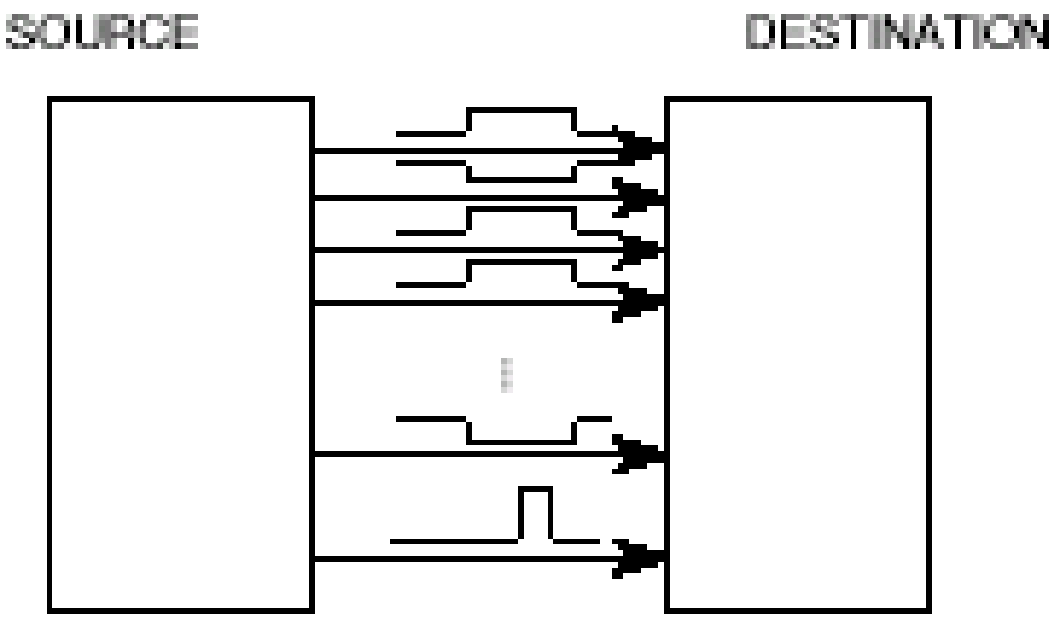
binary	MSN	1000		1001		1010		1011		1100		1101		1110		1111	
LSN	hex	8		9		A		B		C		D		E		F	
0000	0		128 80		144 90		160 A0		176 B0		192 C0		208 D0		224 E0	0	240 F0
0001	1	a	129 81	j	145 91		161 A1		177 B1	A	193 C1	J	209 D1		225 E1	1	241 F1
0010	2	b	130 82	k	146 92	s	162 A2		178 B2	B	194 C2	K	210 D2	S	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	`	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

Bảng mã EBCDIC (tiếp theo)

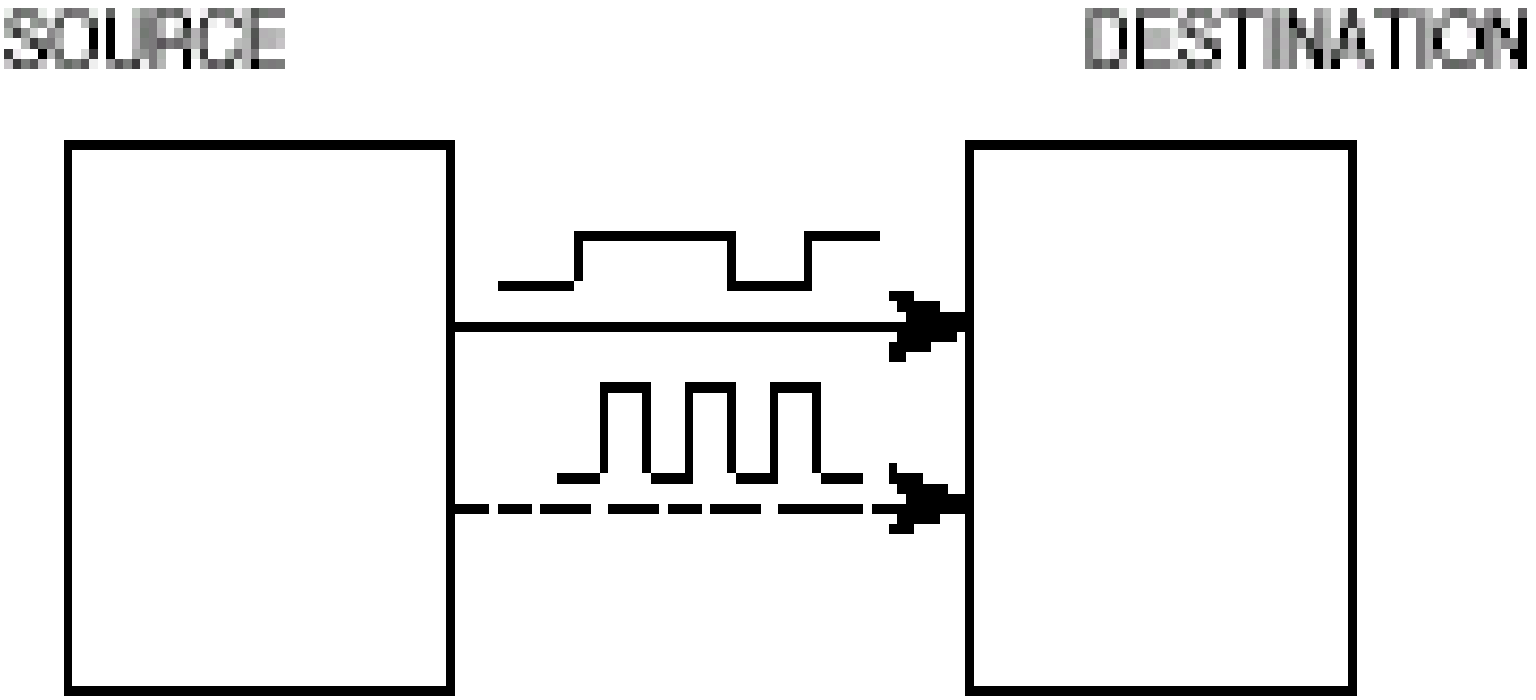
binary	MSN	0000		0001		0010		0011		0100		0101		0110		0111	
LSN	hex	0		1		2		3		4		5		6		7	
0000	0	NUL	0 00	DLE	16 10	SP	32 20	0	48 30	@	64 40	P	80 50	`	96 60	p	112 70
0001	1	SOH	1 01	XON (DC1)	17 11	!	33 21	1	49 31	A	65 41	Q	81 51	a	97 61	q	113 71
0010	2	STX	2 02	DC2	18 12	"	34 22	2	50 32	B	66 42	R	82 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	[	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

Bảng mã ASCII

TRUYỀN SONG SONG

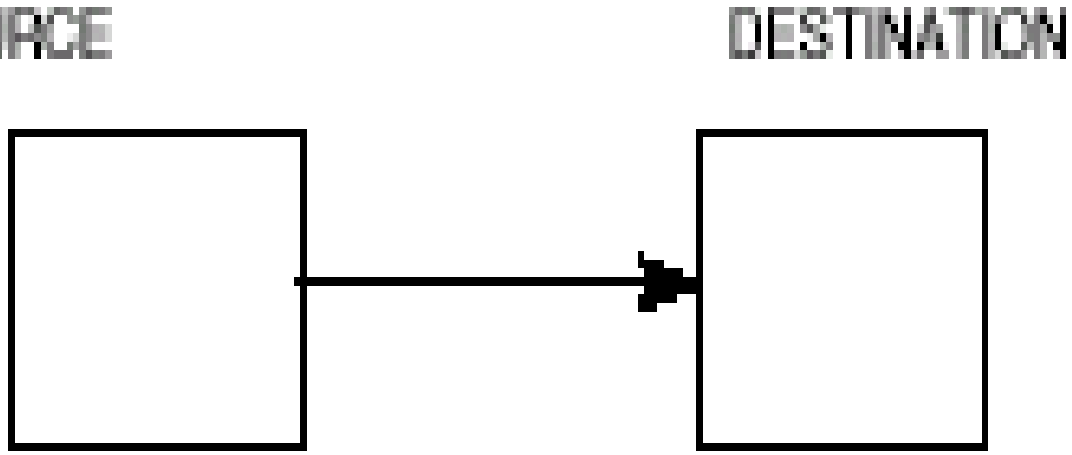


TRUYỀN NỐI TIẾP

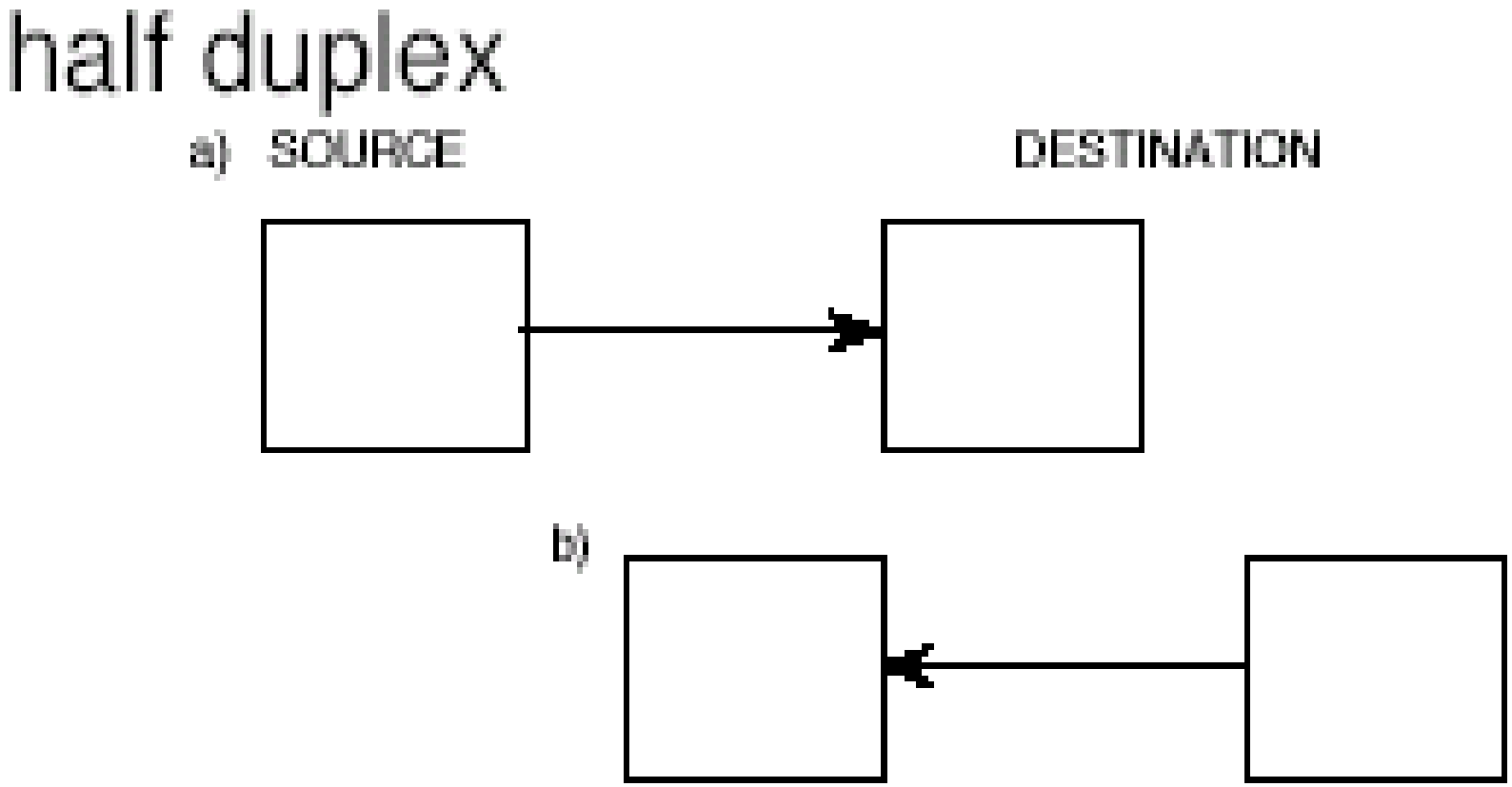


CHẾ ĐỘ ĐƠN CÔNG

simplex

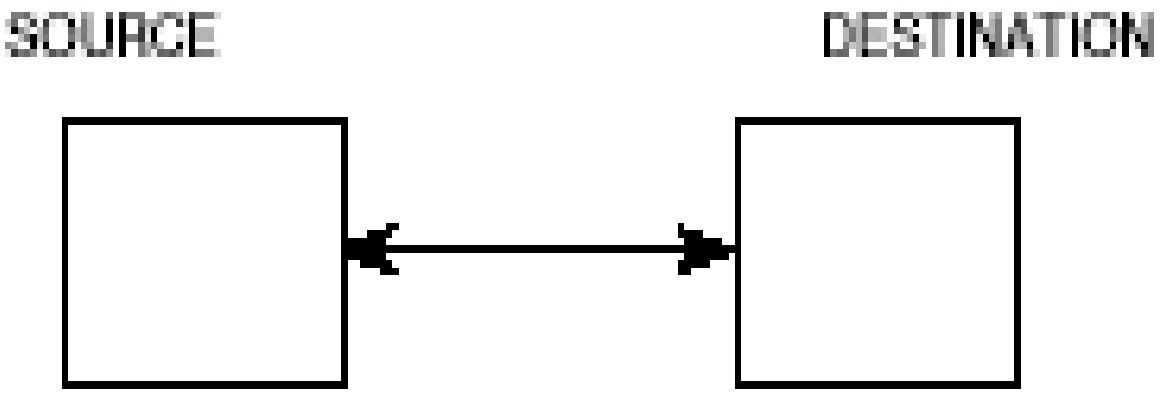


CHẾ ĐỘ BÁN SONG CÔNG



CHẾ ĐỘ SONG CÔNG

full duplex



CÁC MODE ĐỒNG BỘ PHIÊN TRUYỀN

- Phần tử nhỏ nhất trong truyền dữ liệu là *bit*.
- Từng 8 bit nhóm thành các *byte* hoặc *ký tự (character)*.
- Các byte hoặc ký tự được tổ chức thành các *khung(frame)*.

Để đồng bộ phiên truyền, ta cần xác định:

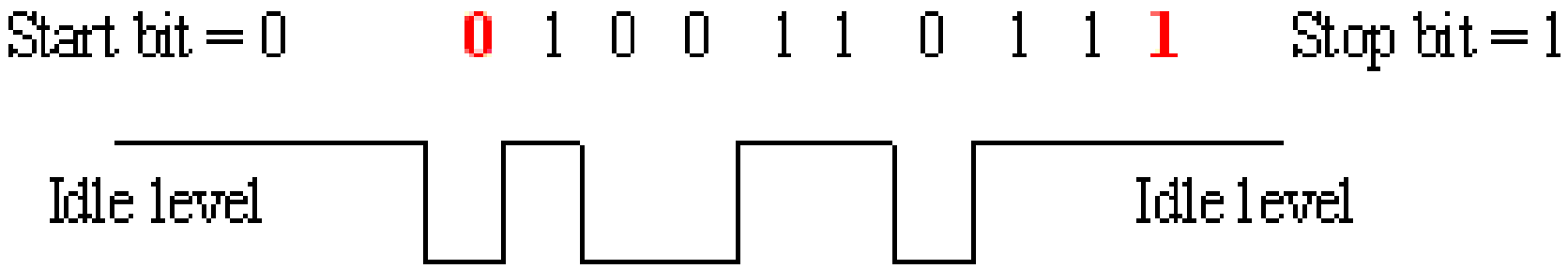
- Điểm bắt đầu mỗi chu kỳ bit = *đồng bộ bit (đồng hồ)*
- Điểm bắt đầu mỗi byte hoặc ký tự = *đồng bộ byte or ký tự*
- Điểm bắt đầu mỗi khung = *đồng bộ khung*

CÁC MODE ĐỒNG BỘ PHIÊN TRUYỀN

Để đồng bộ được 3 cấp trên, có 2 mode đồng bộ được sử dụng:

- Mode *không đồng bộ (Asynchronous transmission)*
- Mode *đồng bộ (Synchronous transmission)*

MODE TRUYỀN DẪN KHÔNG ĐỒNG BỘ



- Sự đồng bộ giữa 2 DTE ***không*** được duy trì trong suốt phiên truyền mà chỉ được thiết lập mỗi khi có dữ liệu truyền.
- *Ví dụ: có ký tự cần truyền đi thì đồng bộ được thiết lập.*

- Sử dụng cho dữ liệu ký tự, đồng bộ được thiết lập cho từng ký tự ở Star bit và kết thúc ở Stop bit.

MODE TRUYỀN DẪN ĐỒNG BỘ

- Sự đồng bộ giữa 2 DTE *được thiết lập từ đầu và duy trì* trong suốt phiên truyền.
- Sử dụng cho dữ liệu là các khối lớn, truyền với tốc độ cao.
- Đầu thu giữ được đồng bộ với đầu phát bằng cách:

- Luồng bit phát đi phải được mã hoá để đồng bộ bit.

- Tất cả các khung phải các các byte hoặc ký tự dự trữ để đồng bộ byte (hoặc ký tự).

- Từng khung phải được "đóng gói" trong một cặp ký tự (byte) đặc biệt để đồng bộ khung.

ĐIỀU KHIỂN LỖI

- Điều khiển lỗi là chức năng cần thiết để khắc phục các lỗi do môi trường vật lý gây ra.
- Điều khiển lỗi là sửa lỗi hoặc ***yêu cầu phát lại*** khung lỗi.
- Sửa lỗi sử dụng ***mã sửa sai***.
- Yêu cầu phát lại sử dụng ***mã phát hiện lỗi***.

  *Ví dụ: Phát hiện lỗi bằng bit **Parity (P)**.*

ĐIỀU KHIỂN LUỒNG

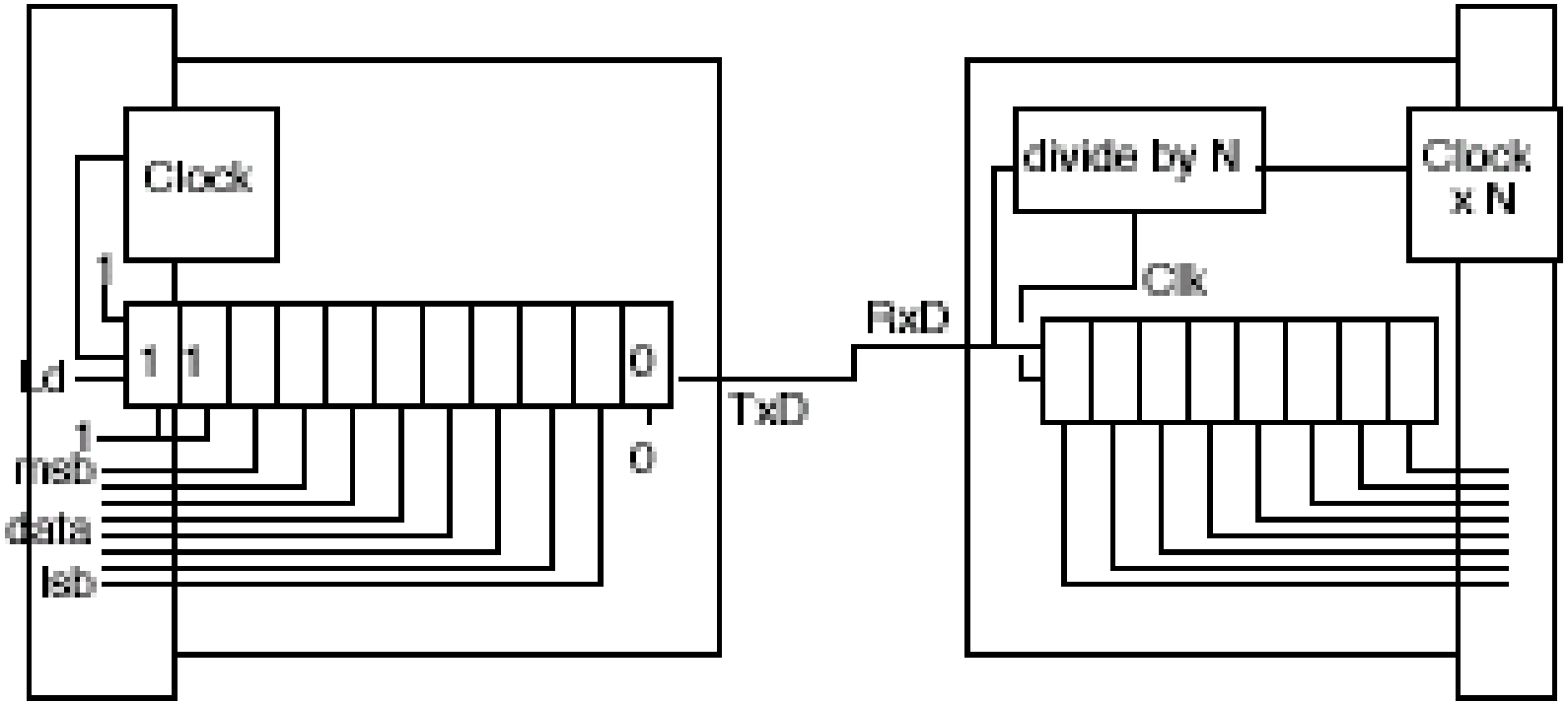
- Điều khiển luồng là điều chỉnh tốc độ phát của DTE nguồn để không gây tràn ở DTE đích.

II. TRUYỀN KHÔNG ĐỒNG BỘ

Mạch điều khiển truyền dẫn ở DTE bao gồm các khối chức năng sau:

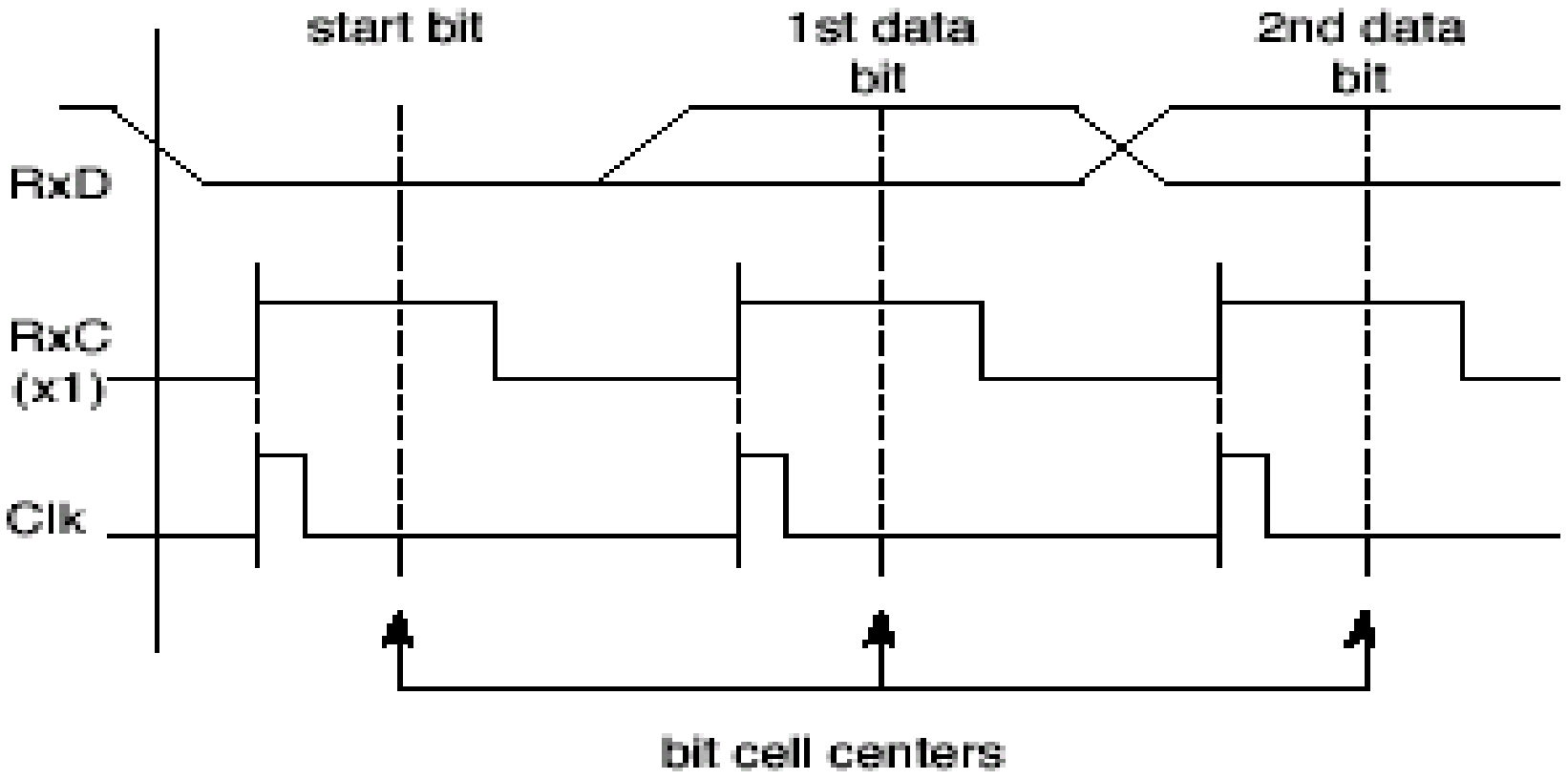
- P/S khi phát đi.
- S/P khi thu về.
- Khối chức năng thực hiện các cấp đồng bộ.
- Khối chức năng thực hiện phát hiện lỗi.

2.1 ĐỒNG BỘ BIT



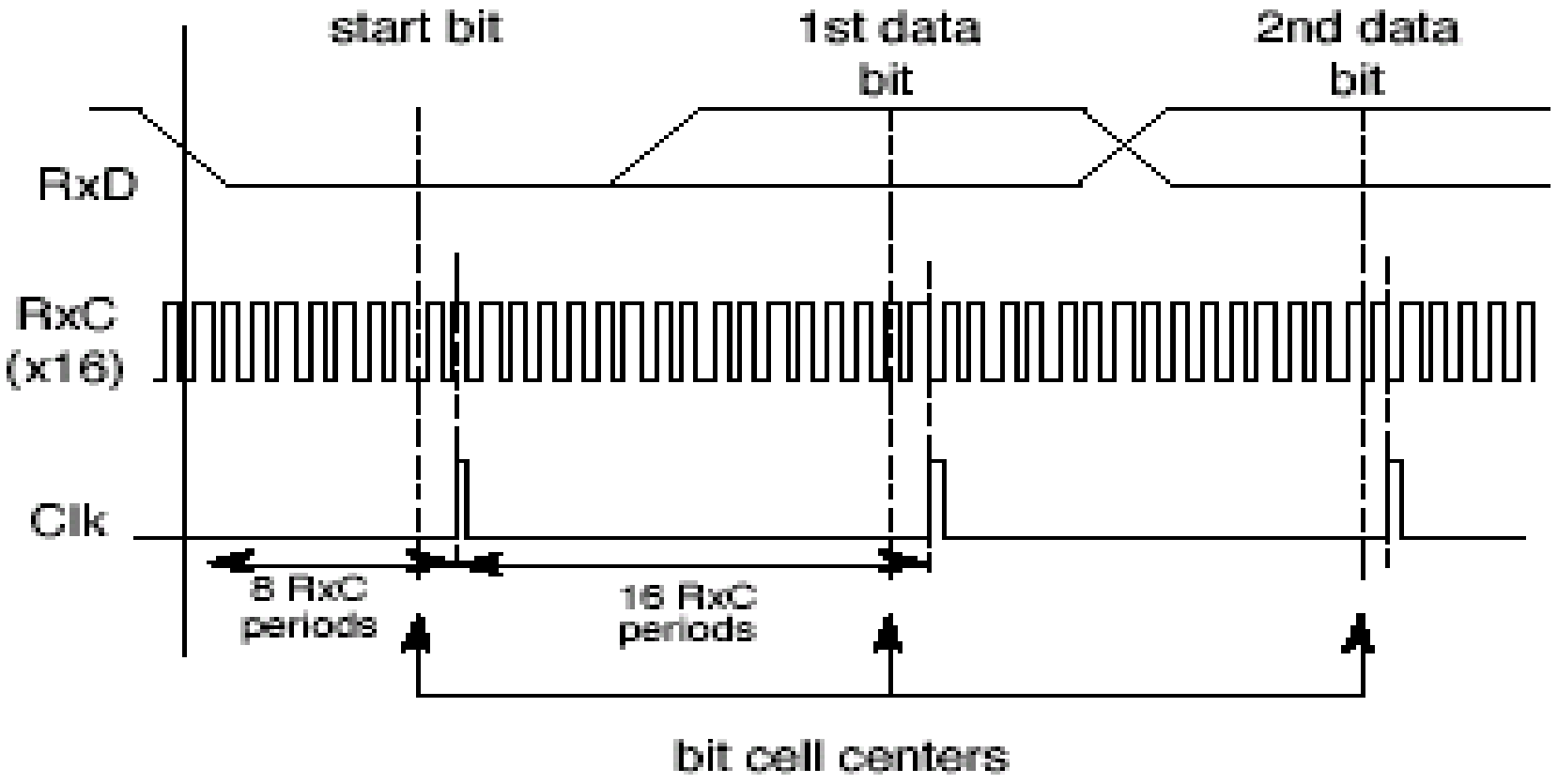
Nguyên lý hoạt động

2.1 ĐỒNG BỘ BIT



N=1 (RxC = TxD)

2.1 ĐỒNG BỘ BIT



N=16 (RxC = 16 x TxD)

2.2 ĐỒNG BỘ KÝ TỰ

- Đồng bộ ký tự được thực hiện sau khi có đồng bộ bit.

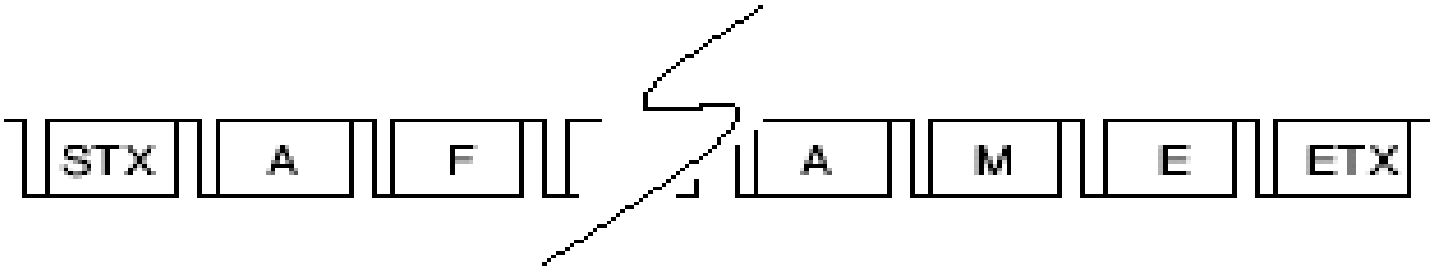
- Các bit nằm giữa start-bit và stop-bit là của một ký tự.

2.3 ĐỒNG BỘ KHUNG

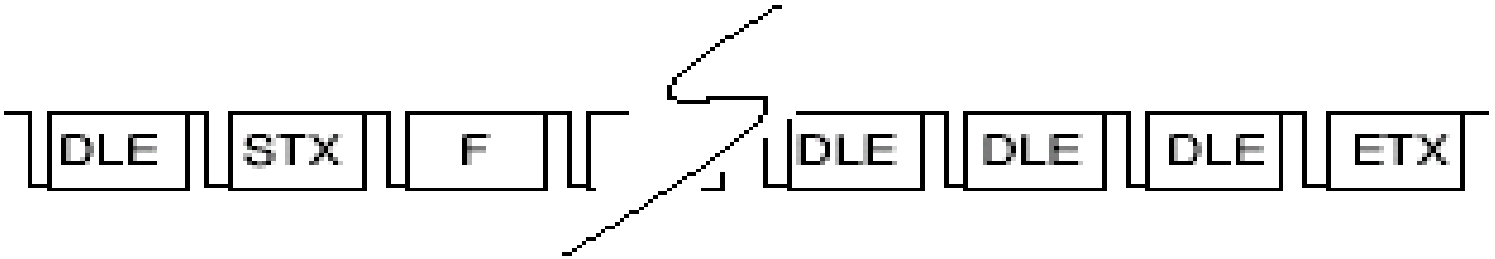
Có hai loại dữ liệu ký tự:

- Ký tự printable: không có các ký tự điều khiển.
- Ký tự Non-printable: bao gồm cả các ký tự điều khiển.
- Ví dụ: một số ký tự điều khiển như: SOH, DLE. STX. . .

2.3 ĐỒNG BỘ KHUNG



Cấu trúc khung với dữ liệu Printable



Cấu trúc khung với dữ liệu Non-Printable

III. TRUYỀN ĐỒNG BỘ

Truyền đồng bộ phân biệt hai loại dữ liệu:

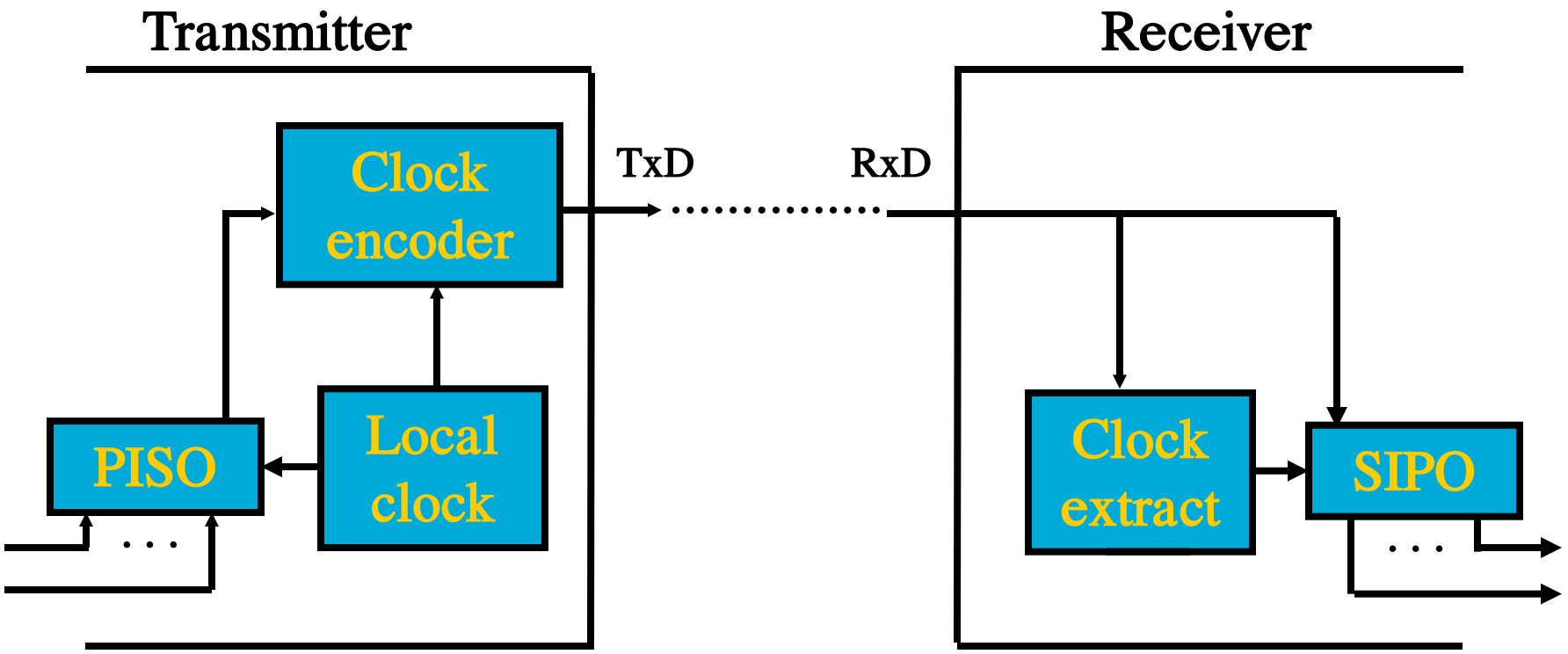
- Dữ liệu là ký tự: có thực hiện đồng bộ ký tự.
- Dữ liệu là bit nhị phân: không thực hiện đồng bộ ký tự.
- Vì vậy có hai loại truyền đồng bộ: *hướng ký tự* & *hướng bit*.
- Cả hai loại trên đều có chung chức năng đồng *bộ bit*.

3.1 ĐỒNG BỘ BIT

Trong truyền đồng bộ có hai phương pháp đồng bộ bit:

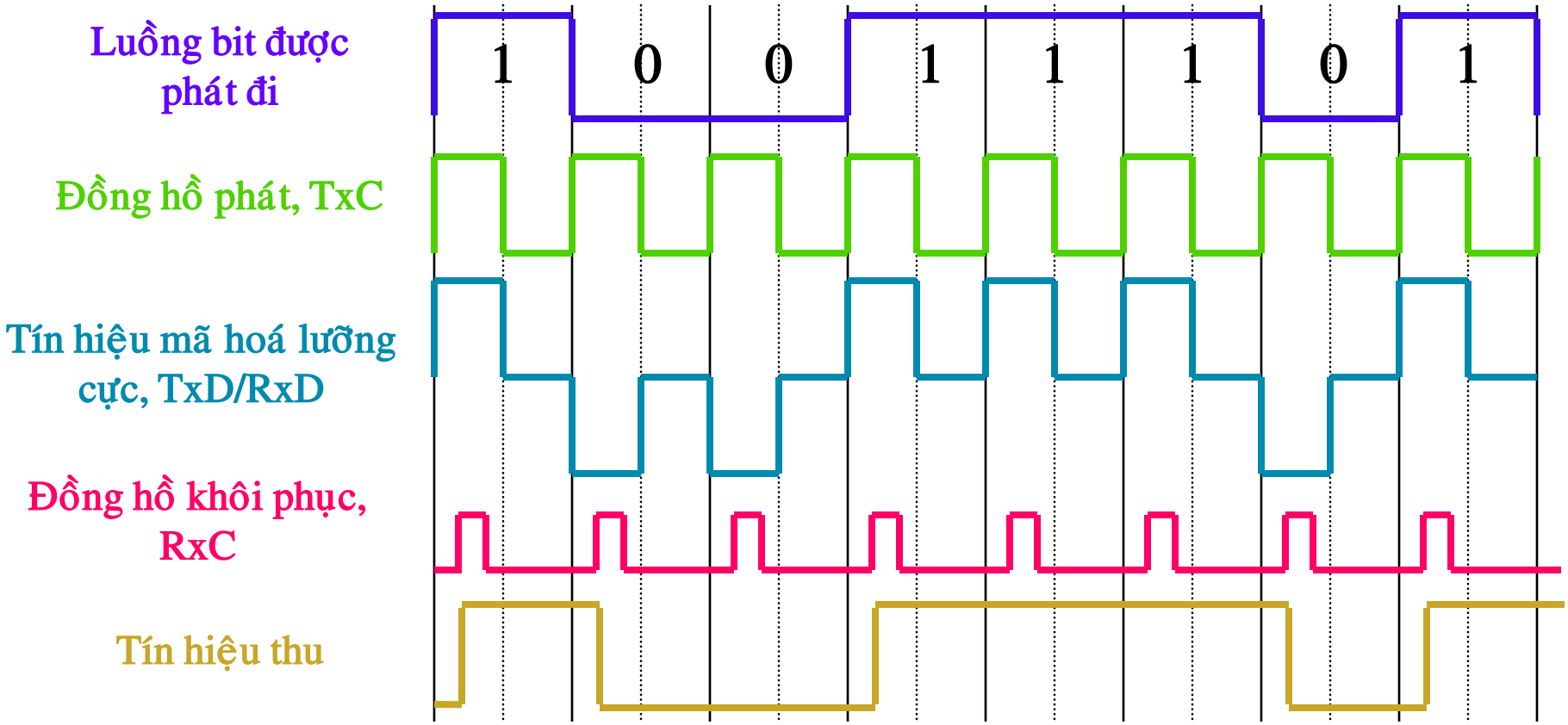
- Đồng bộ bit bằng *mã hoá* và *khôi phục đồng hồ*.
- Đồng bộ bit bằng *vòng khoá pha số*(Phase lock loop-DPLL).

a) Đồng bộ bit bằng mã hoá và khôi phục đồng hồ

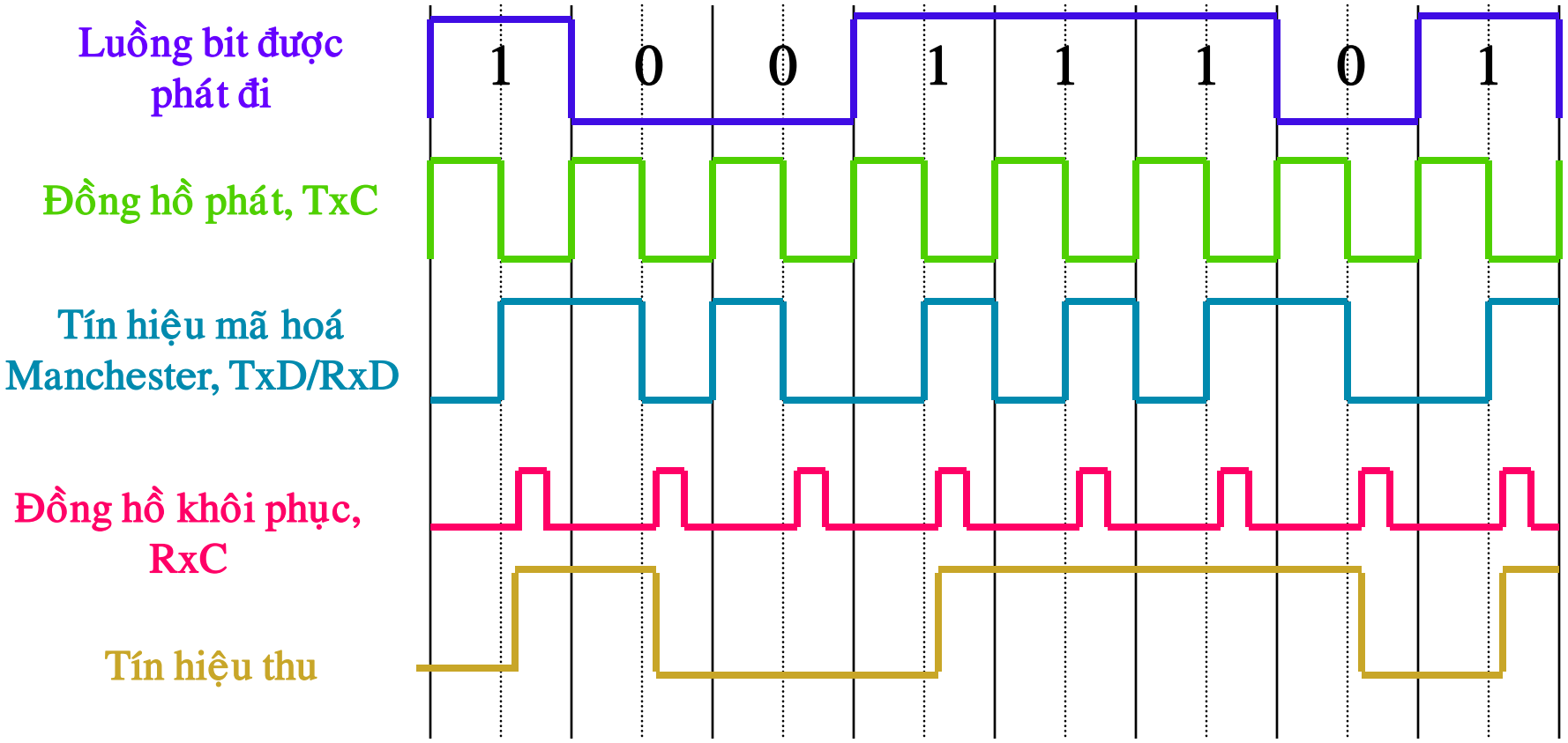


- Các loại mã: lưỡng cực, Manchester, và Manchester vi sai.
- Việc khôi phục đồng hồ giựa vào các chuyển tiếp của mã.

Mã hóa lưỡng cực (Bipolar)

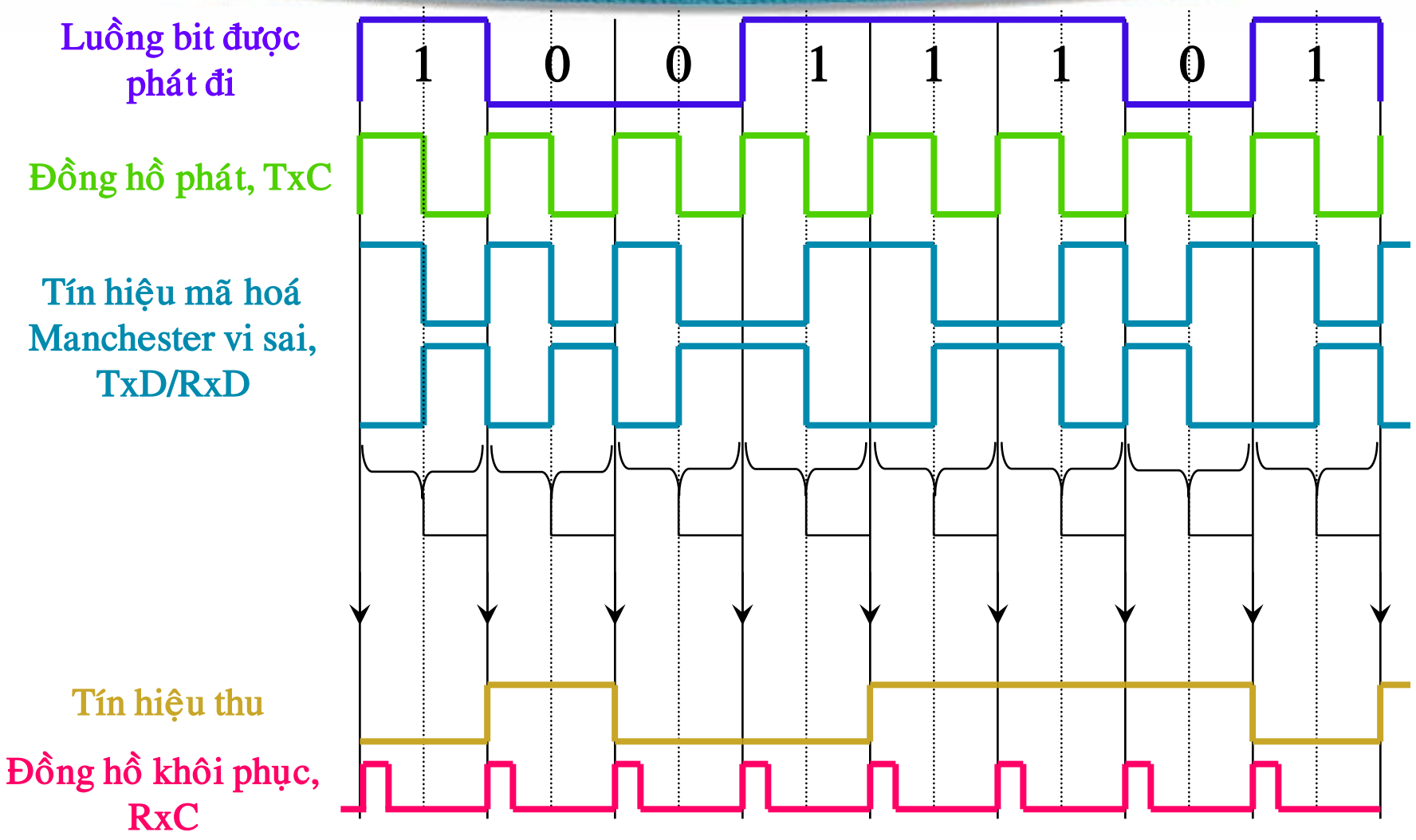


Mã hóa Manchester (Mã hoá Phase)

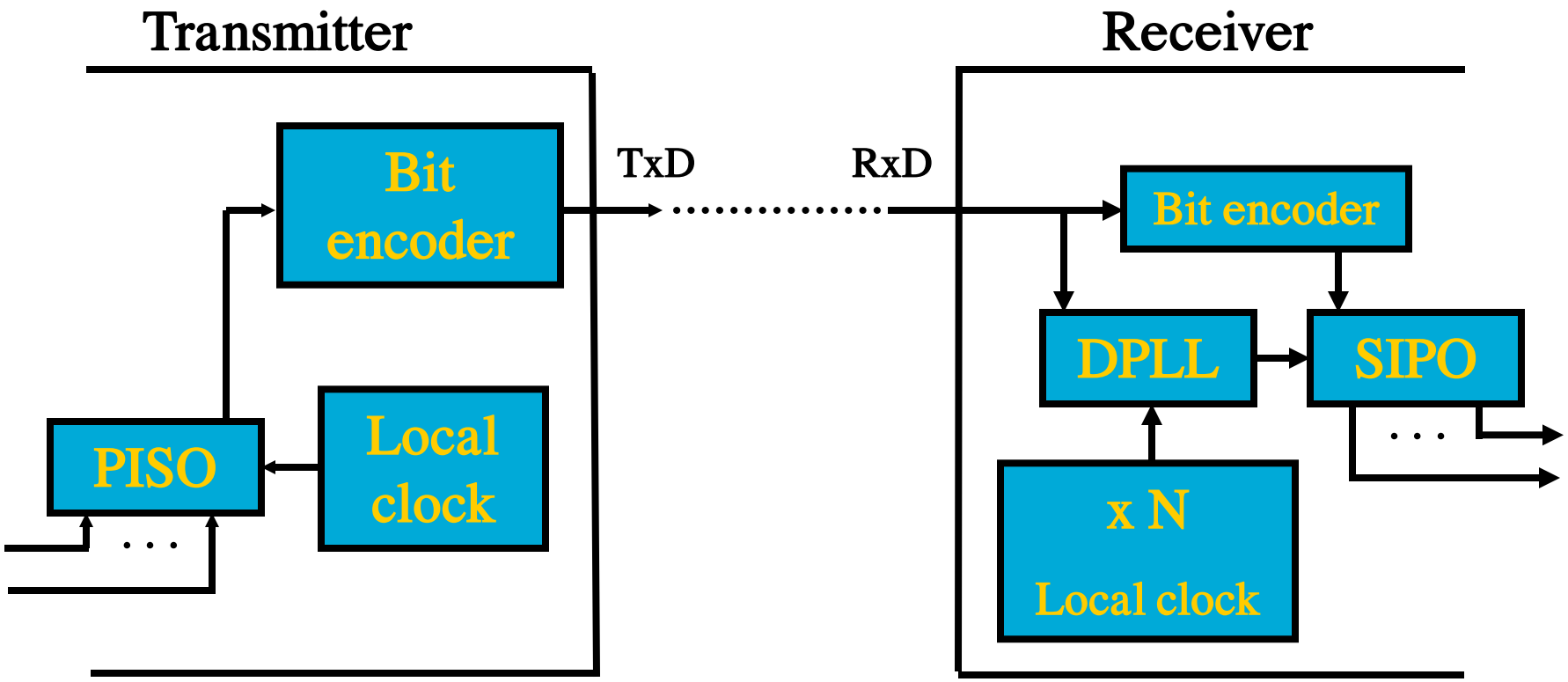


Mã hoá Manchester (Mã hóa Phase)

Mã hóa Manchester vi sai

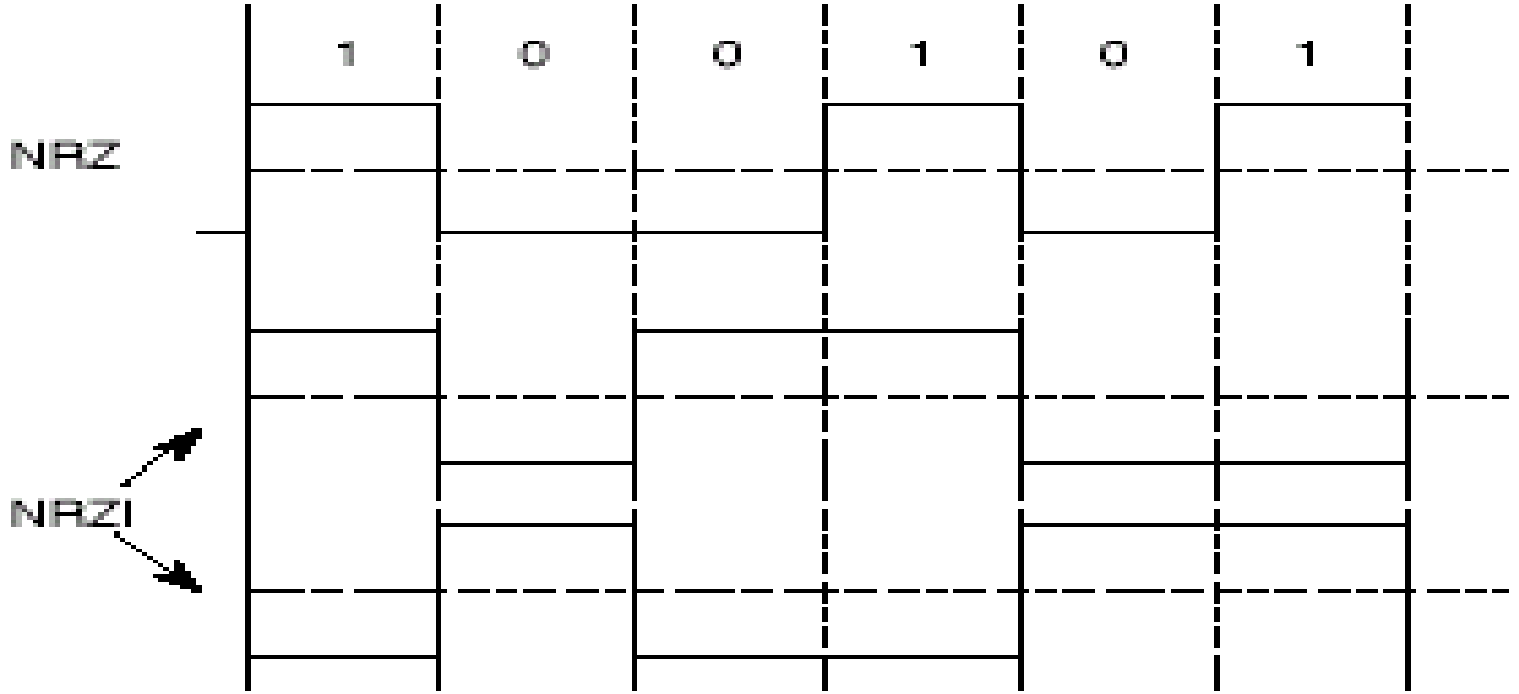


b) Đồng bộ bit bằng DPLL



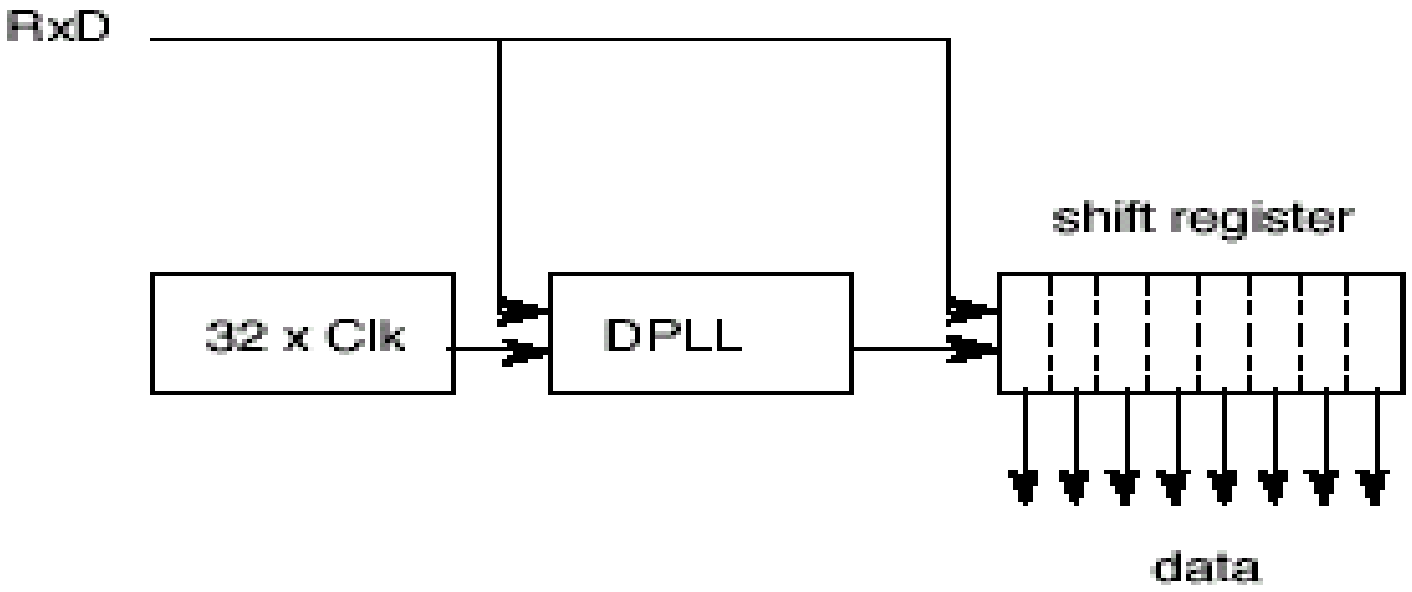
- Các bit được mãhóa: *NRZI*.

b) Đồng bộ bit bằng DPLL

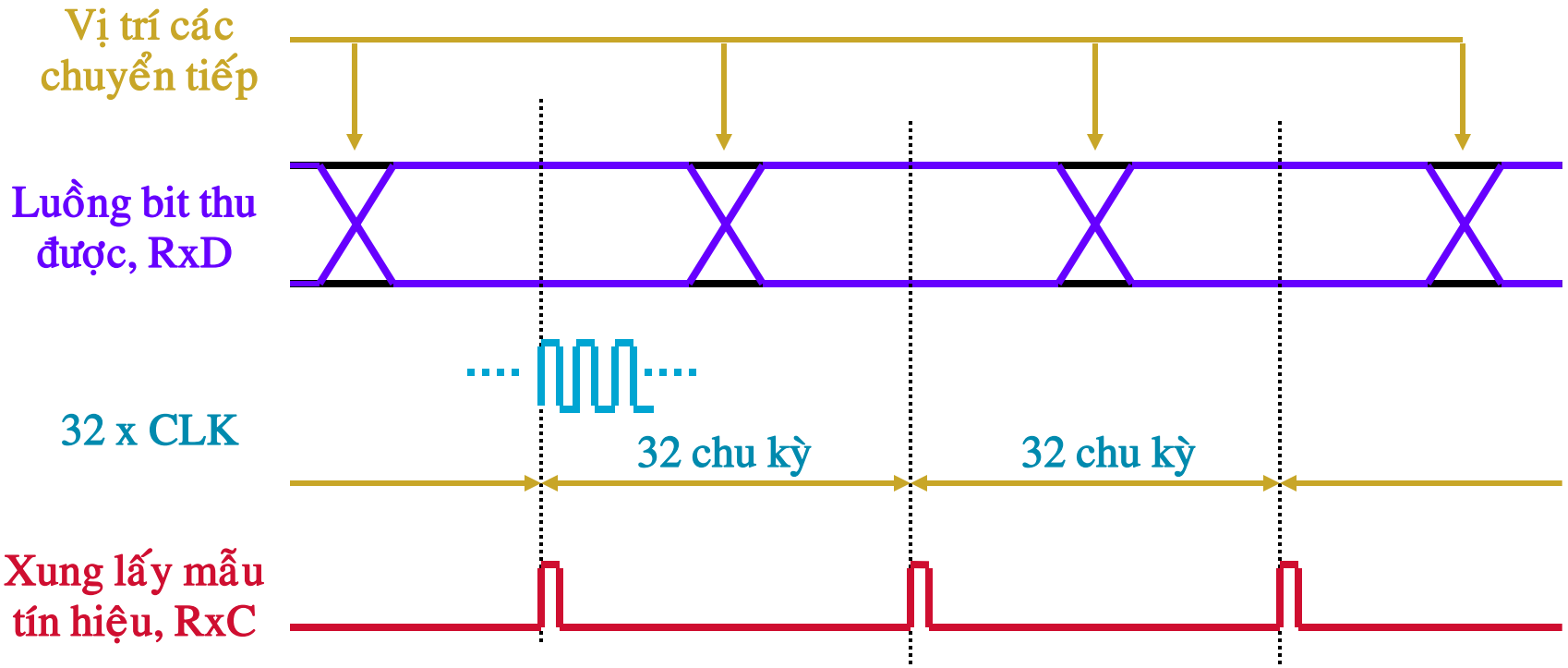


- NRZI: bit 0 - đổi trạng thái, bit 1 - không đổi trạng thái.

b) Đồng bộ bit bằng DPLL

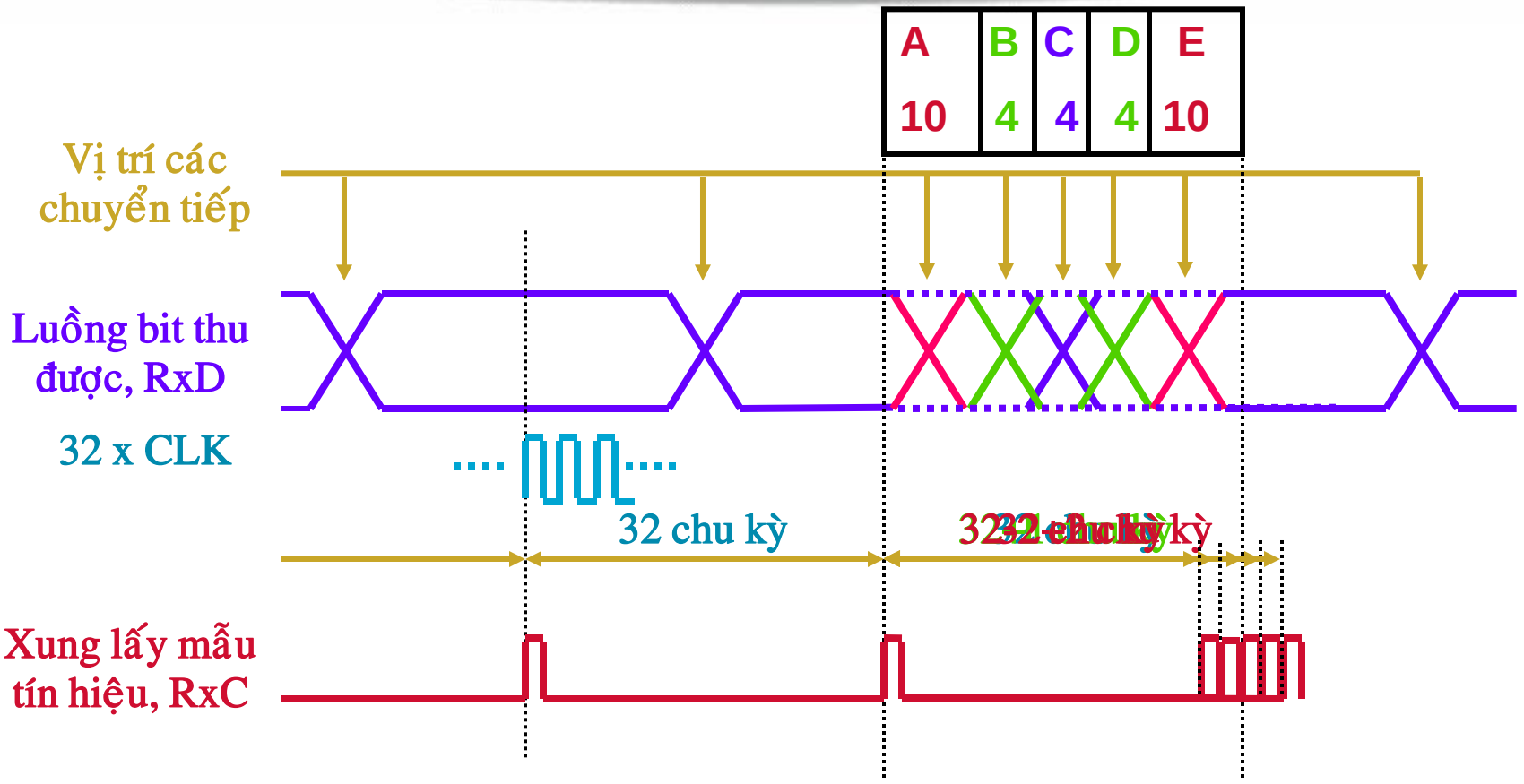


b) Đồng bộ bit bằng DPLL



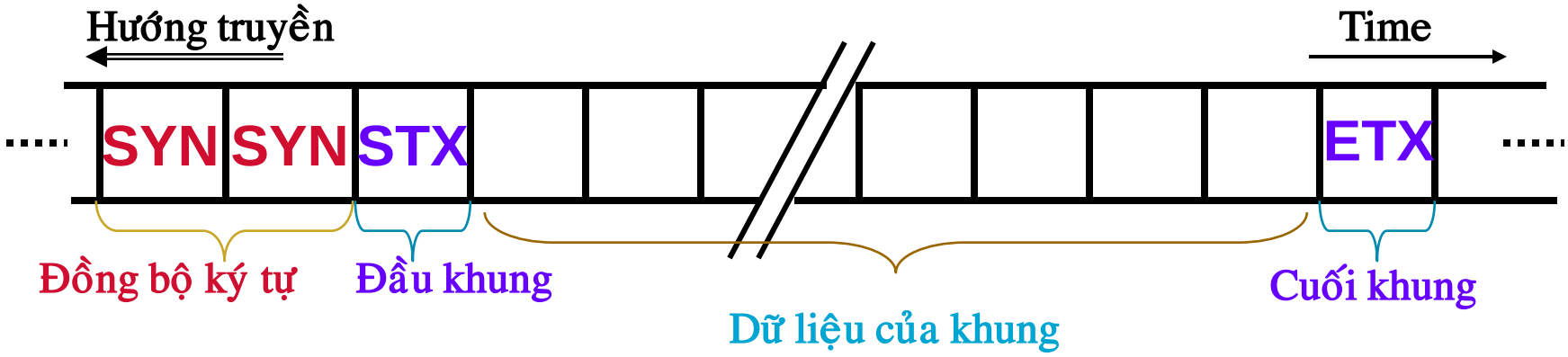
Trường hợp lý tưởng.

b) Đồng bộ bit bằng DPLL



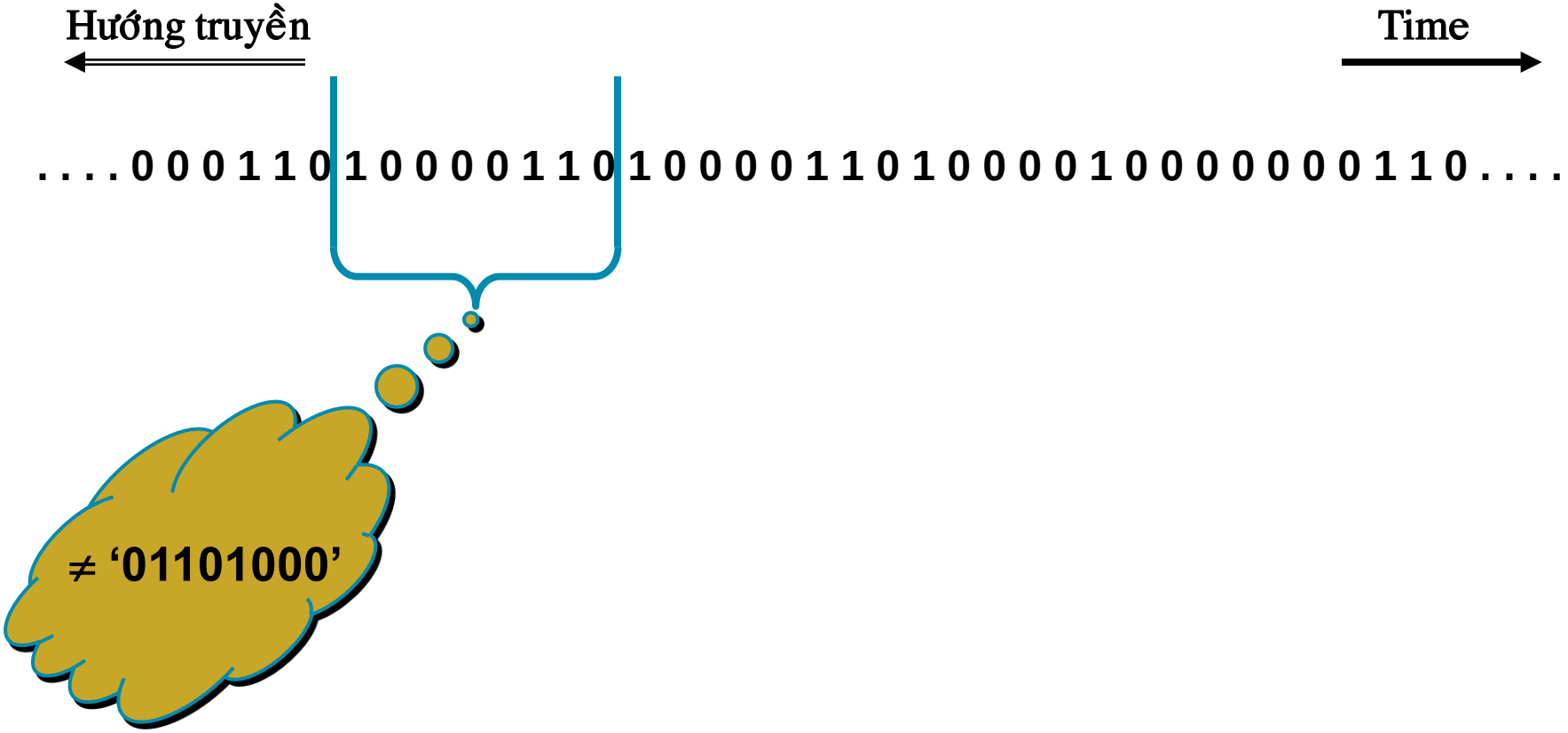
Trường hợp điều chỉnh pha (DPLL).

3.2 ĐỒNG BỘ HƯỚNG KÝ TỰ



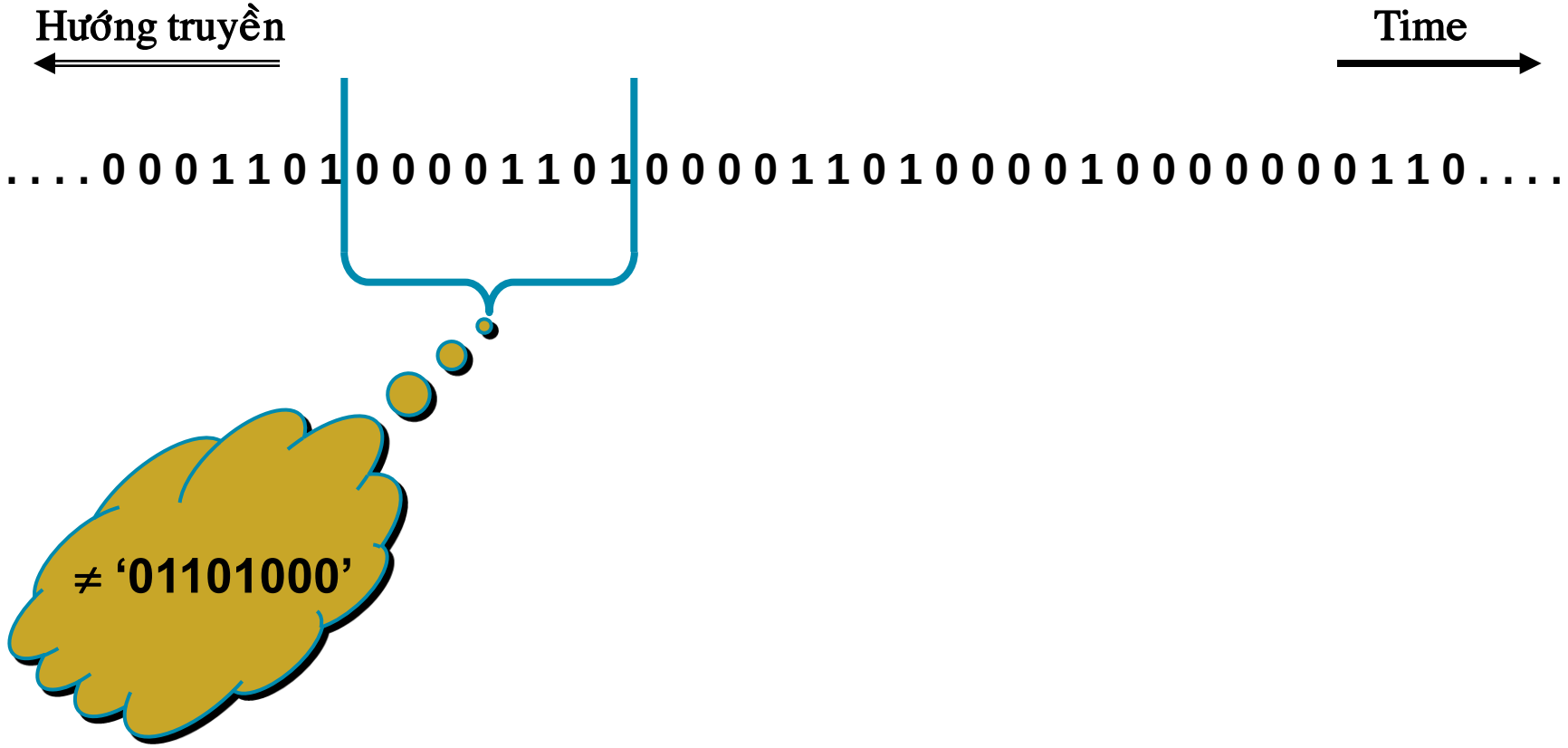
Cấu trúc khung.

ĐỒNG BỘ HƯỚNG KÝ TỰ



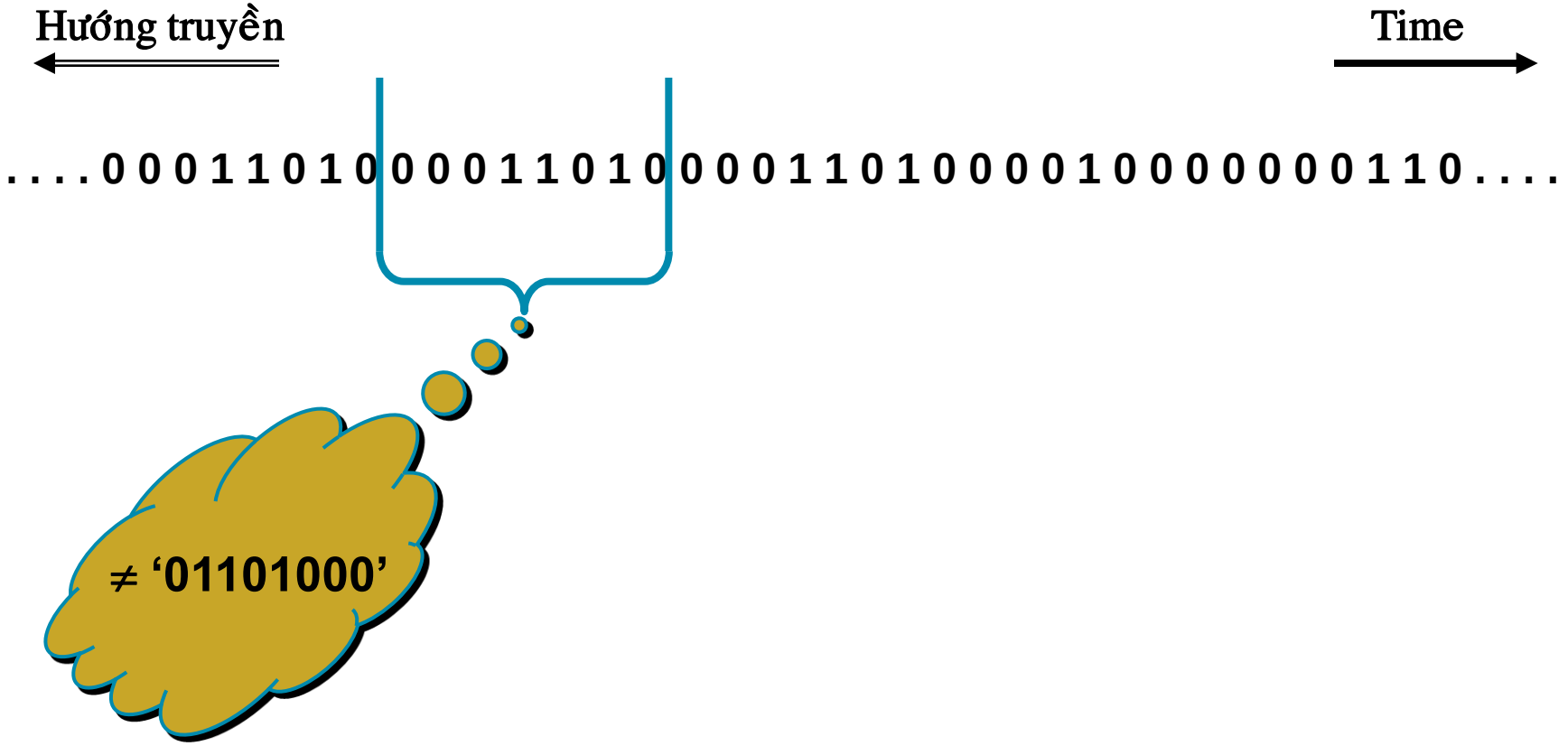
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



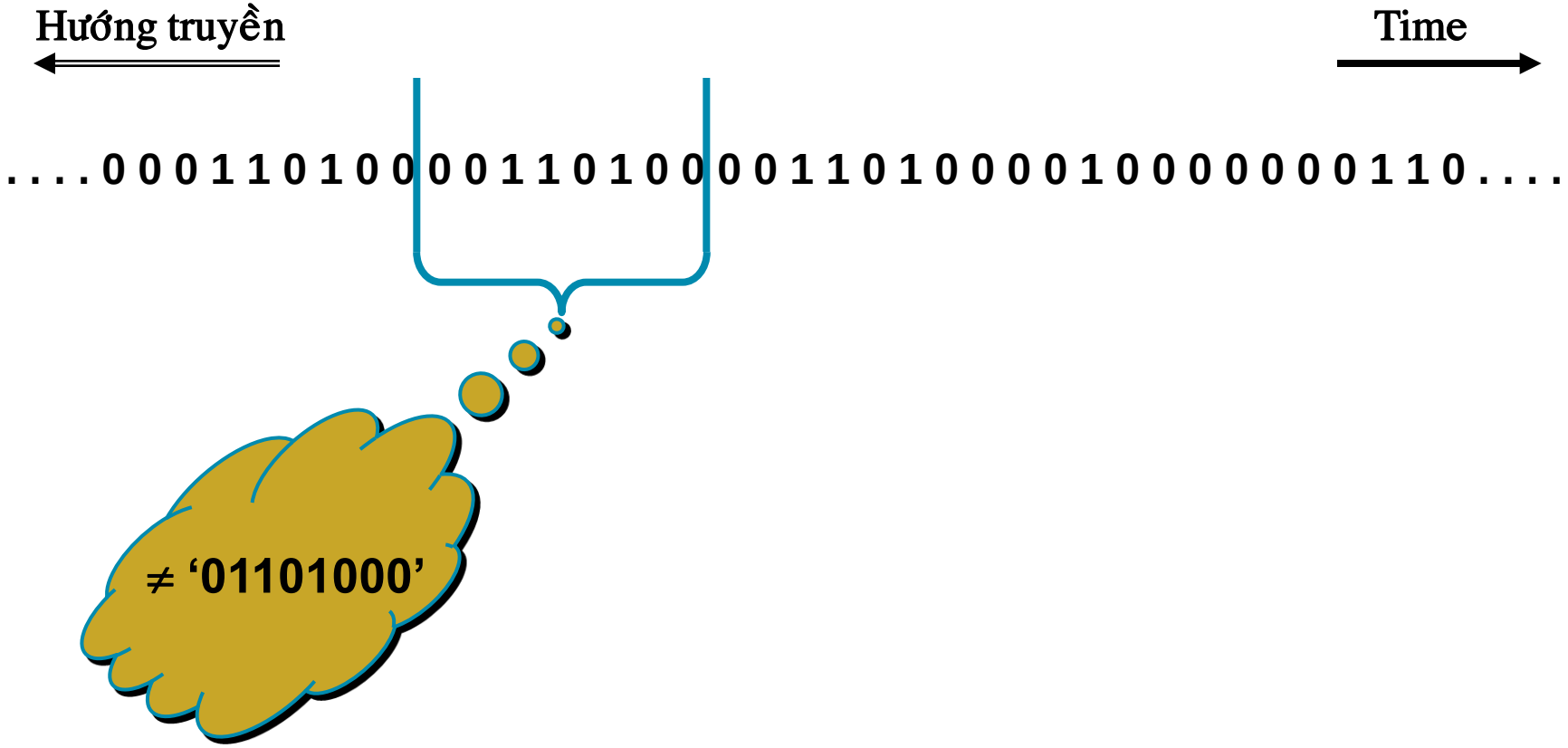
Quá trình đồng bộ ký tự.

3.2 ĐỒNG BỘ HƯỚNG KÝ TỰ



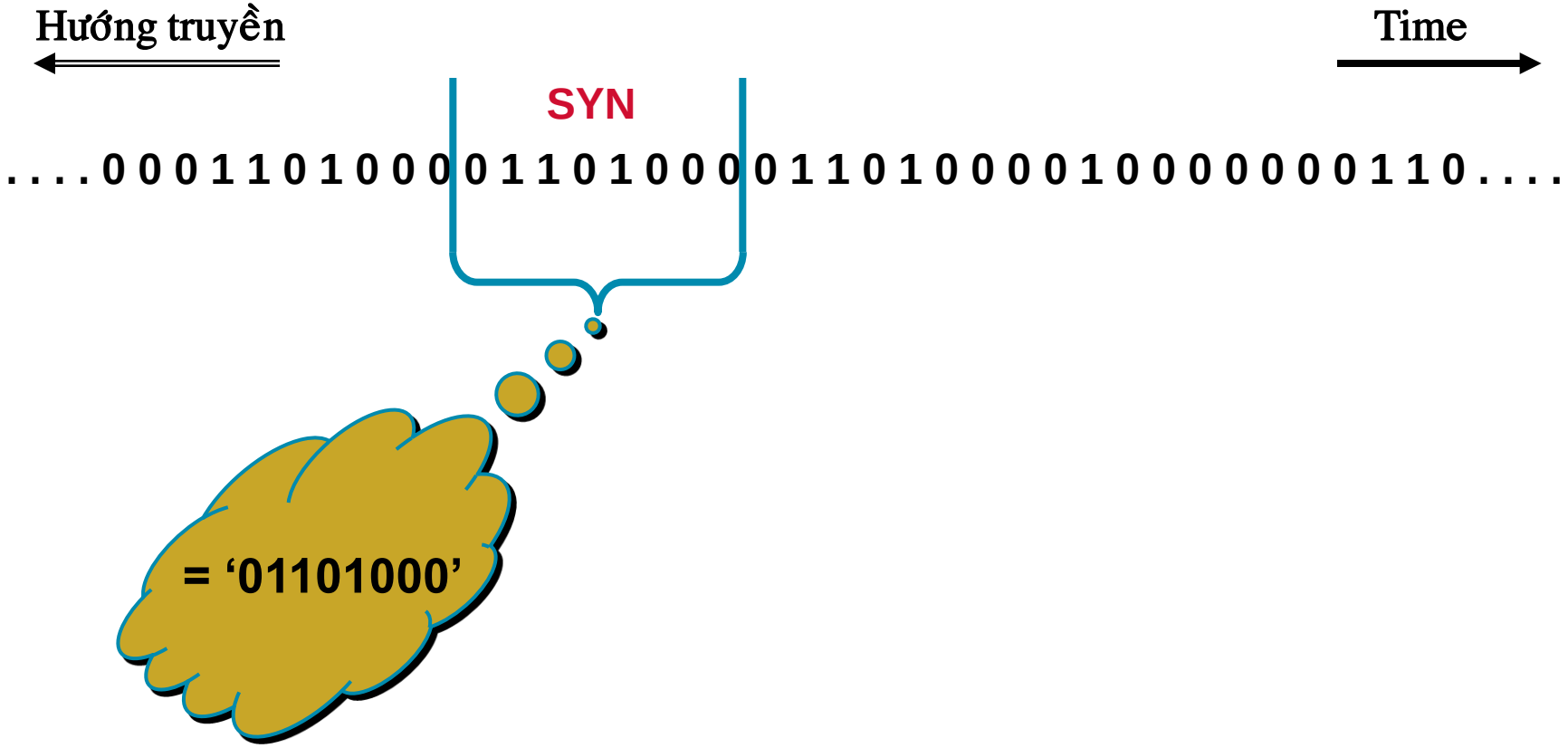
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



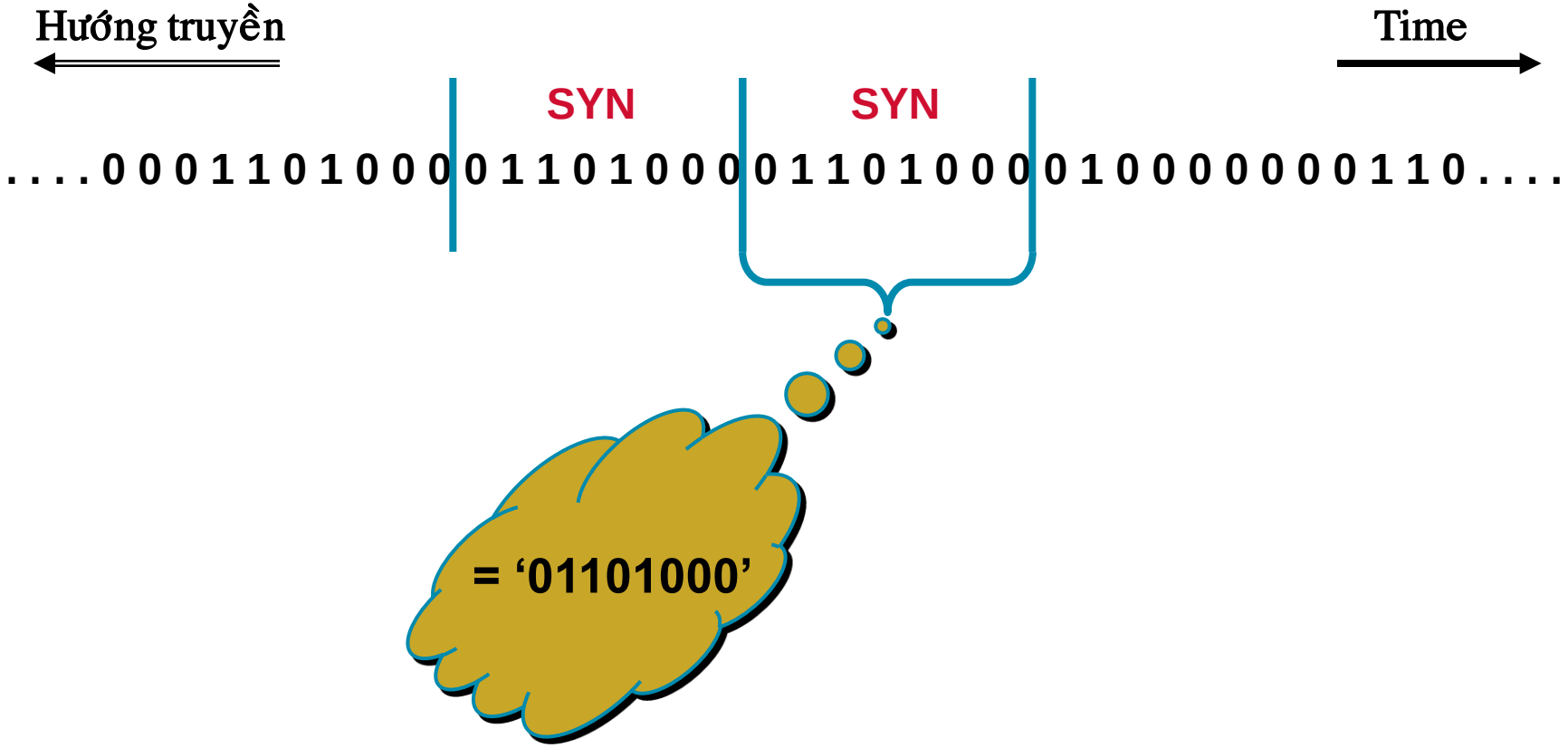
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



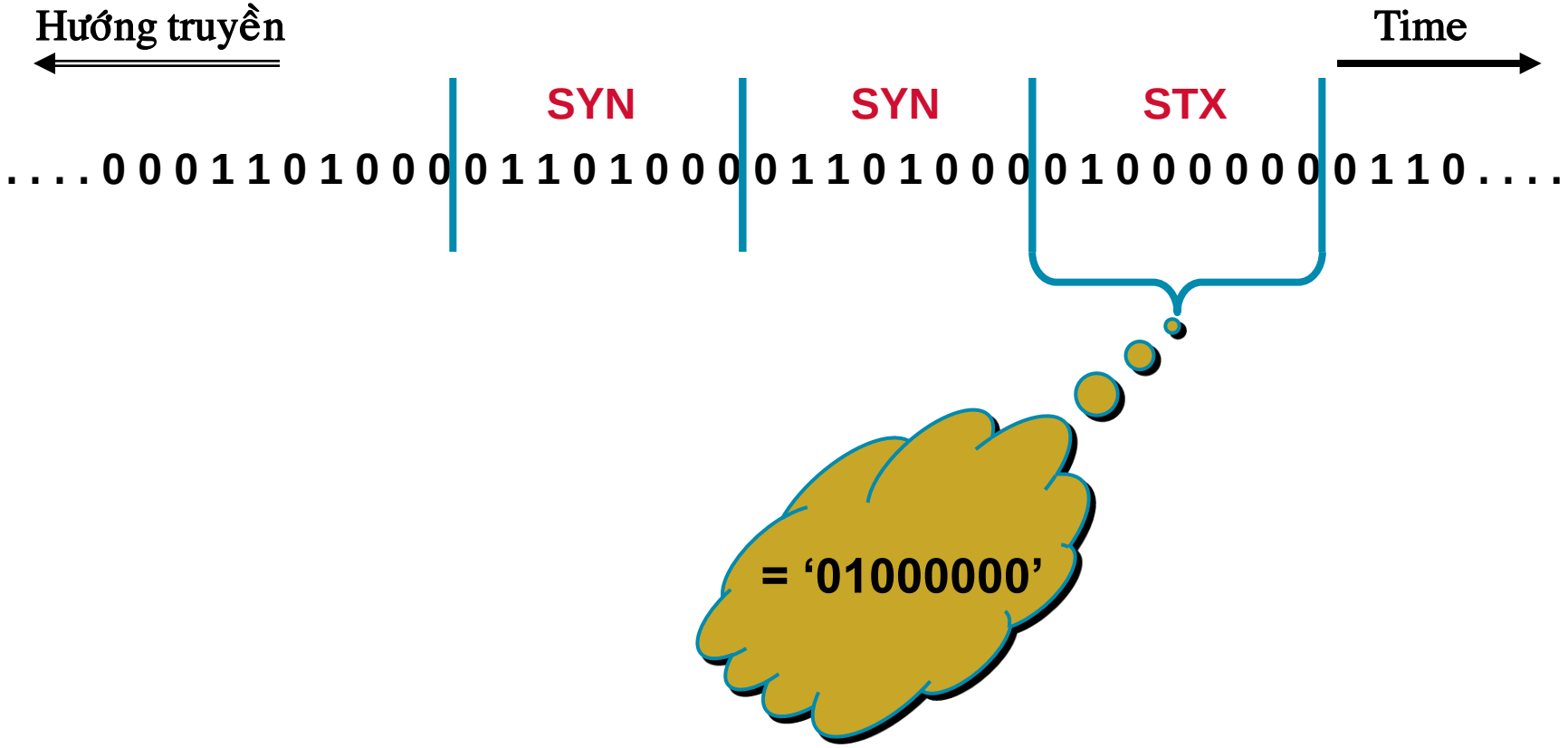
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



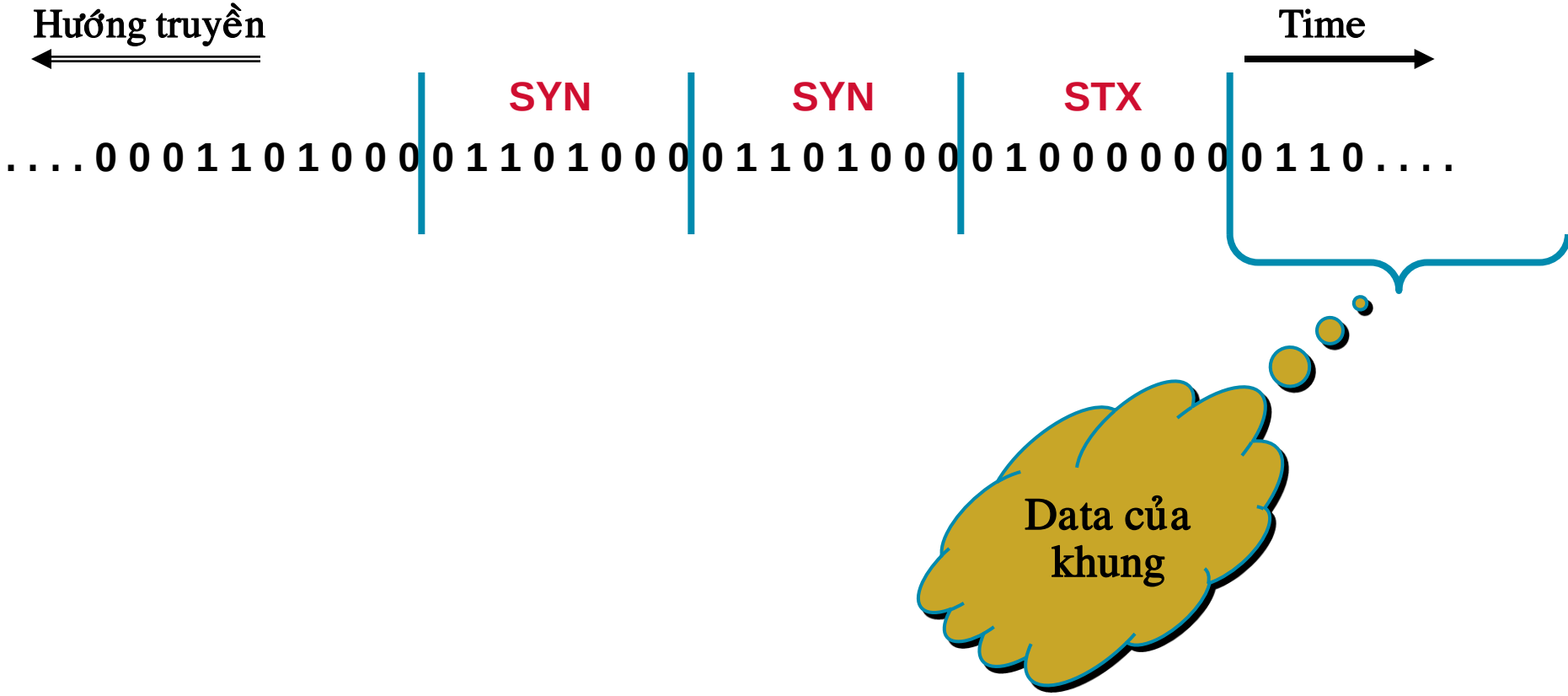
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



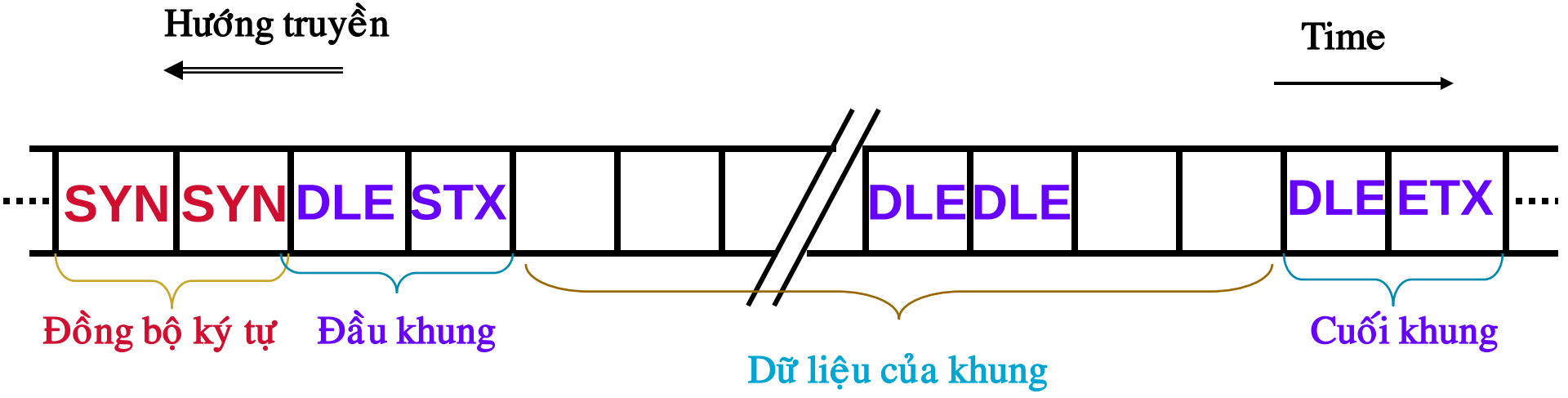
Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



Quá trình đồng bộ ký tự.

ĐỒNG BỘ HƯỚNG KÝ TỰ



Chèn thêm DLE cho dữ liệu Non-printable.

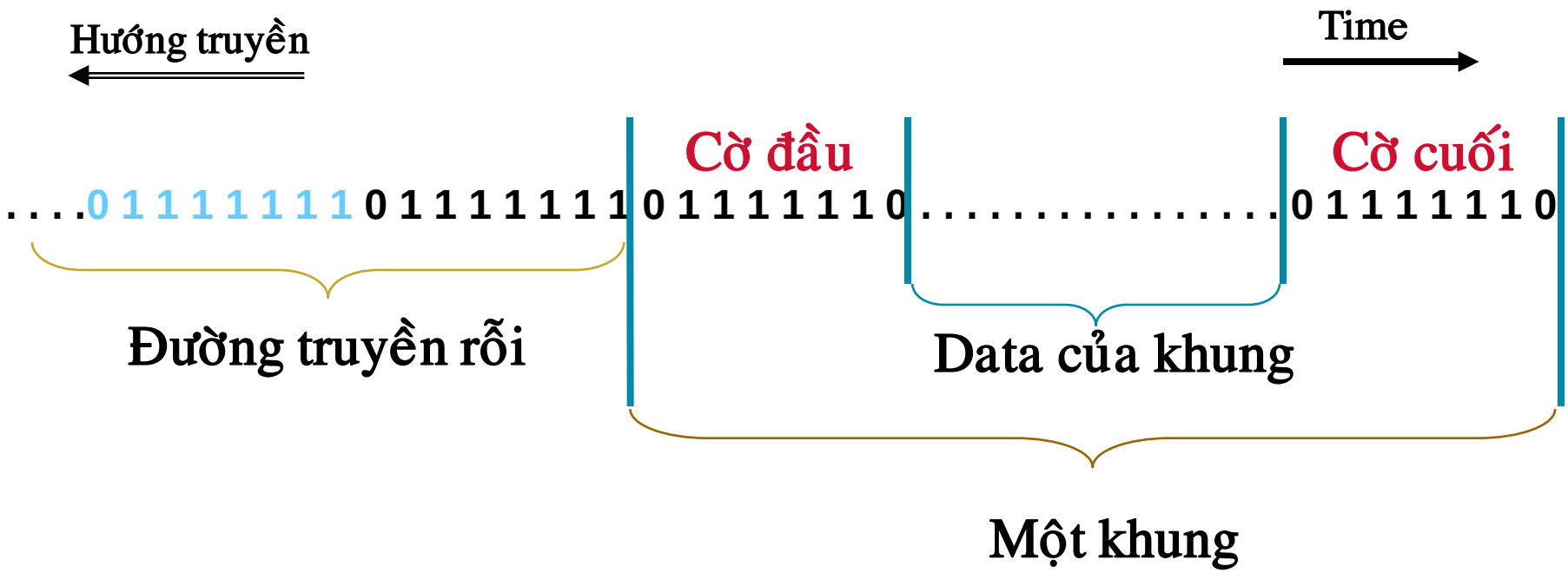
3.3 ĐỒNG BỘ HƯỚNG BIT

Truyền đồng bộ hướng bit không có cấp đồng bộ ký tự:

Có ba phương pháp đồng bộ khung:

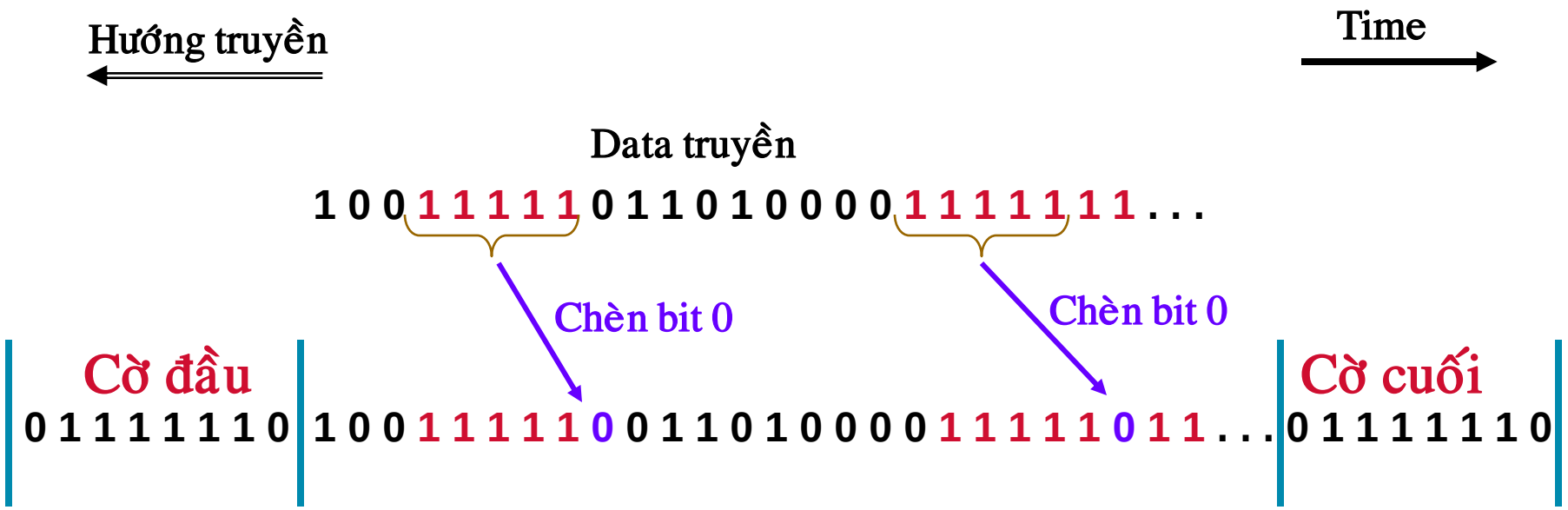
- Sử dụng *cờ đầu khung* và *cờ cuối khung (01111110)*.
- Sử dụng *cờ đầu khung (10101011)* và *độ dài khung (Length)* .
- Sử dụng *các bit vi phạm (JK0JK000, JK1JK111)* .

a) Sử dụng cờ đầu & cuối khung

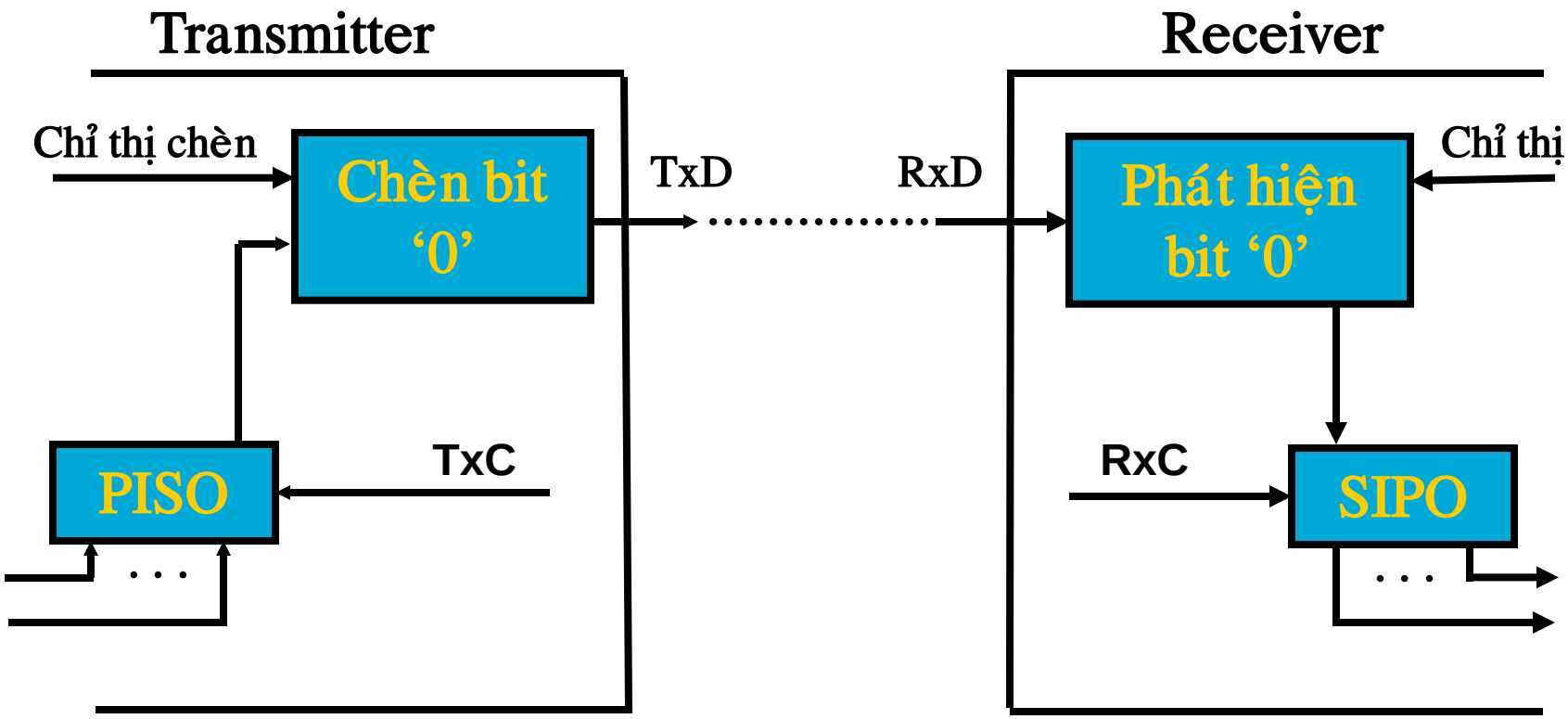


- Không được có nhóm '111111' trong thành phần của khung.

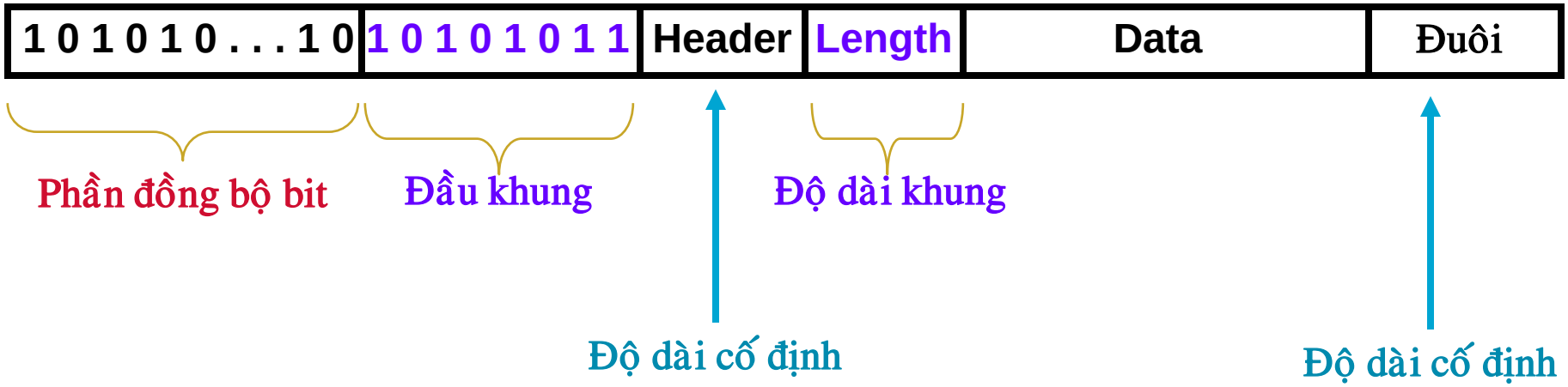
a) Sử dụng cờ đầu & cuối khung



a) Sử dụng cờ đầu & cuối khung



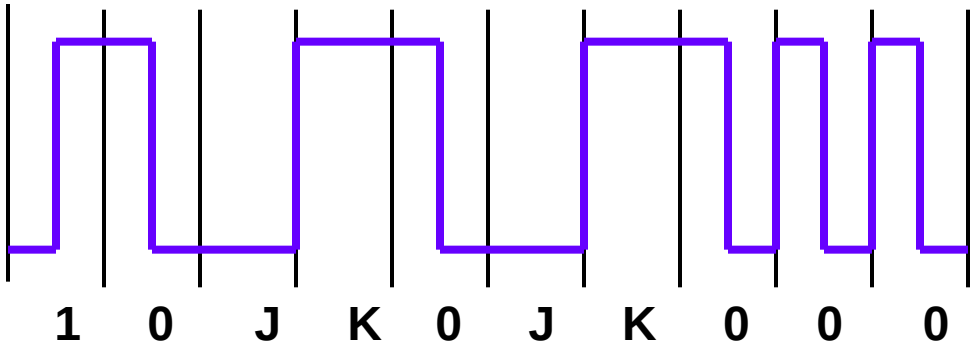
b) Sử dụng cờ đầu & độ dài khung



Sơ đồ này thường được sử dụng trong các LAN.

c) Sử dụng các bit vi phạm

Sử dụng mã Manchester, cờ đầu và cuối



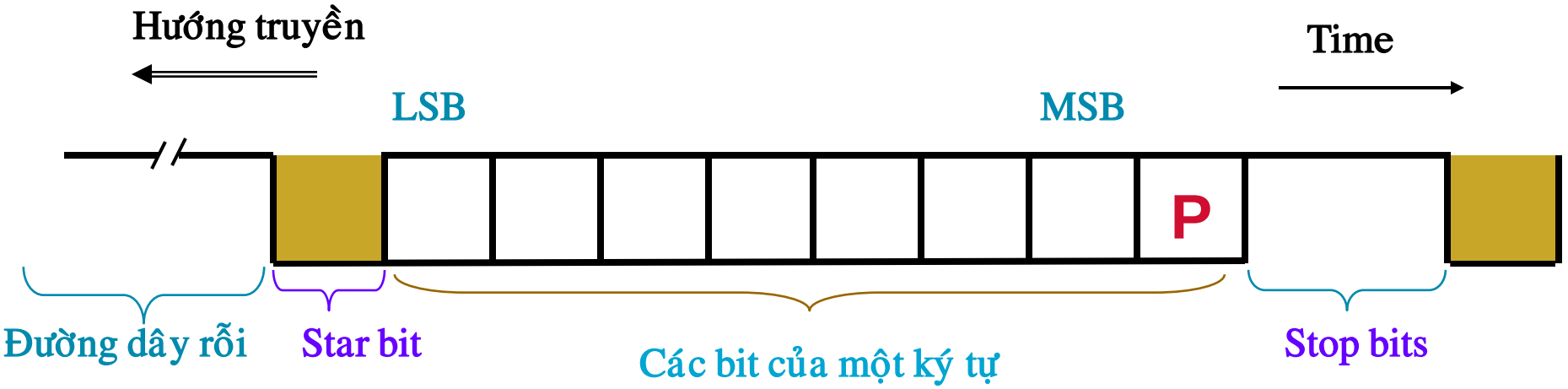
Sơ đồ này thường được sử dụng trong các LAN.

III. CÁC PHƯƠNG THỨC PHÁT HIỆN LỖI

Có hai phương pháp điều khiển lỗi:

- Điều khiển lỗi bằng phương pháp *Sửa lỗi tại đầu thu*.
- Điều khiển lỗi bằng phương pháp *yêu cầu phát lại*.

4.1 PARITY

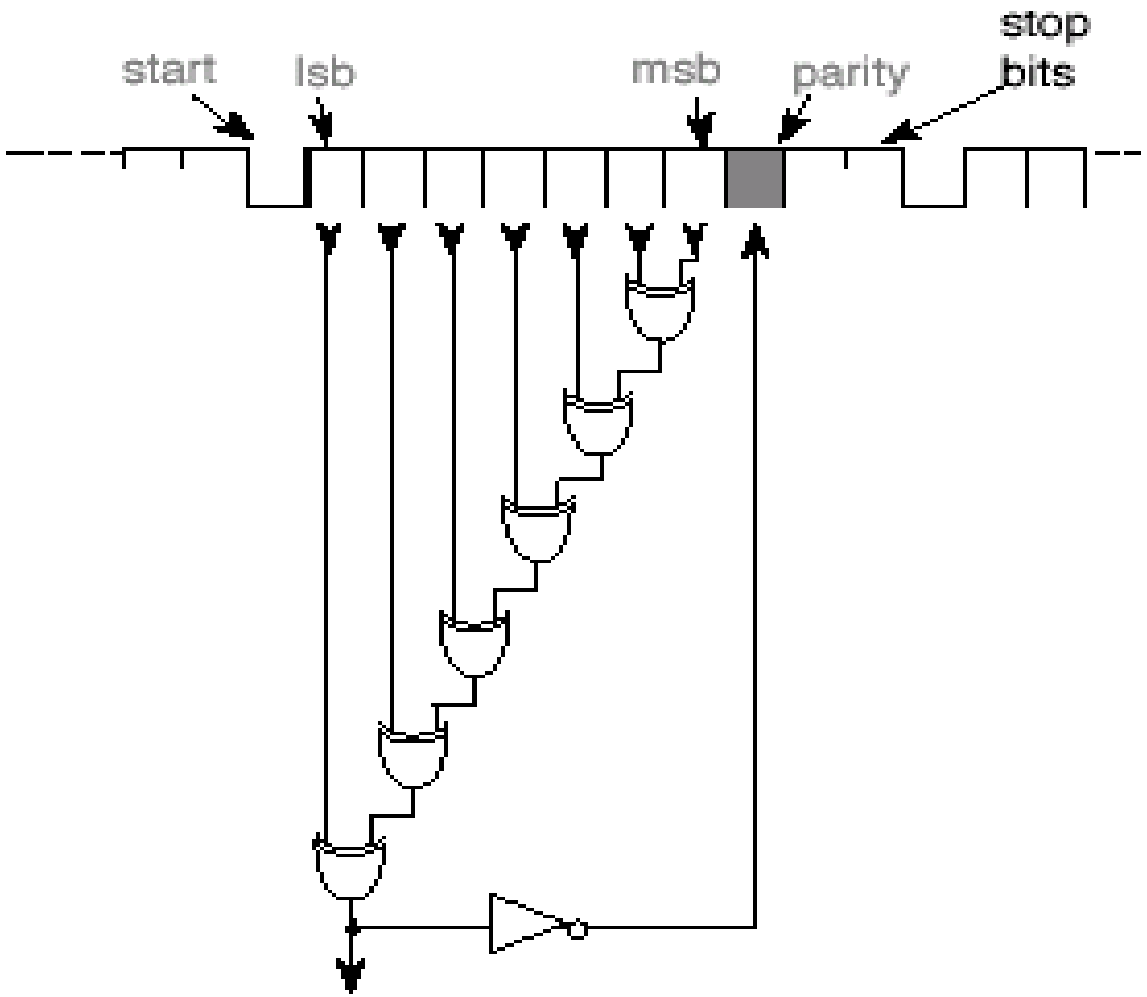


1001001 1 Parity chẵn

1001001 0 Parity lẻ

- *Chỉ phát hiện được lỗi lẻ bit.*
- *Overhead lớn: 1/8 = 12.5%*

PARITY



Bit 1	Bit 2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

4.2 BLOCK SUM CHECK

P_R	B_6	B_5	B_4	B_3	B_2	B_1	B_0	
0	0	0	0	0	0	1	0	STX
1	0	1	0	1	0	0	0	
0	1	0	0	0	1	1	0	
0	0	1	0	0	0	0	0	
1	0	1	0	1	1	0	1	
0	1	0	0	0	0	0	0	
1	1	1	0	0	0	1	1	
1	0	0	0	0	0	1	1	ETX
1	1	0	0	0	0	0	1	BCC (chẵn)

Parity lẻ

Các ký tự của khung

SỬA LỖI MỘT BIT

P_R	B_6	B_5	B_4	B_3	B_2	B_1	B_0	
0	0	0	0	0	0	1	0	STX
1	0	1	0	1	0	0	0	
0	1	0	0	0	0	1	0	
0	0	1	0	0	0	0	0	
1	0	1	0	1	1	0	1	
0	1	0	0	0	0	0	0	
1	1	1	0	0	0	1	1	
1	0	0	0	0	0	1	1	ETX
1	1	0	0	0	0	0	1	BCC (chẵn)

Parity lẻ

Các ký tự của khung

PHÁT HIỆN LỖI 2 BIT

P_R	B_6	B_5	B_4	B_3	B_2	B_1	B_0	
0	0	0	0	0	0	1	0	STX
1	0	1	0	1	0	0	0	
0	1	1	0	0	0	1	0	
0	0	1	0	0	0	0	0	
1	0	1	0	1	1	0	1	
0	1	0	0	0	0	0	0	
1	1	1	0	0	0	1	1	
1	0	0	0	0	0	1	1	ETX
1	1	0	0	0	0	0	1	BCC (chẵn)

Parity lẻ

Các ký tự của khung

Không phát hiện được lỗi

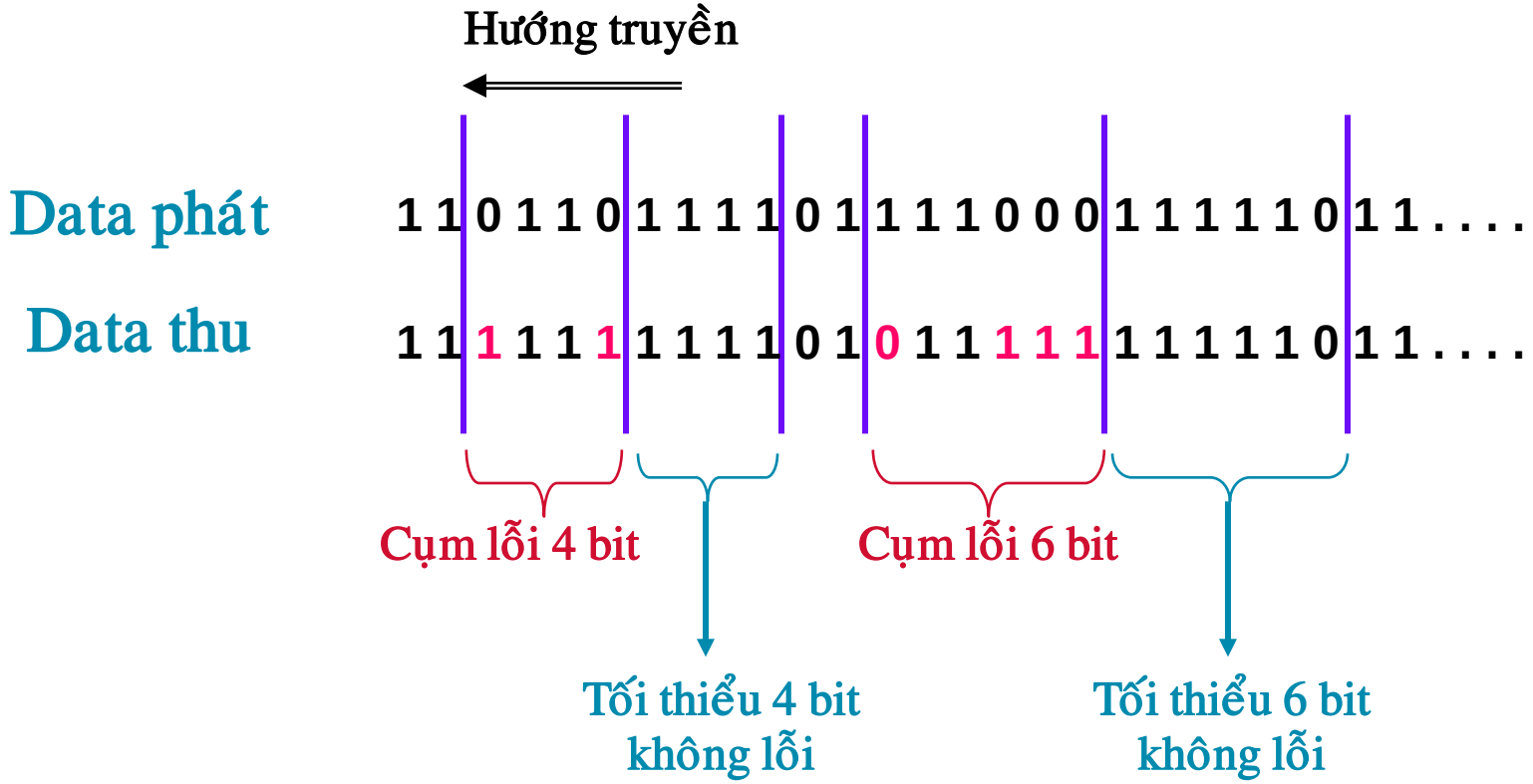
P_R	B_6	B_5	B_4	B_3	B_2	B_1	B_0	
0	0	0	0	0	0	1	0	STX
1	0	1	0	1	0	0	0	
0	1	1	0	0	0	1	0	
0	0	1	0	0	0	0	0	
1	0	1	0	1	1	0	1	
0	1	0	0	0	0	0	0	
1	1	0	0	0	1	1	1	
1	0	0	0	0	0	1	1	ETX
1	1	0	0	0	0	0	1	BCC (chẵn)

Parity lẻ

Các ký tự của khung

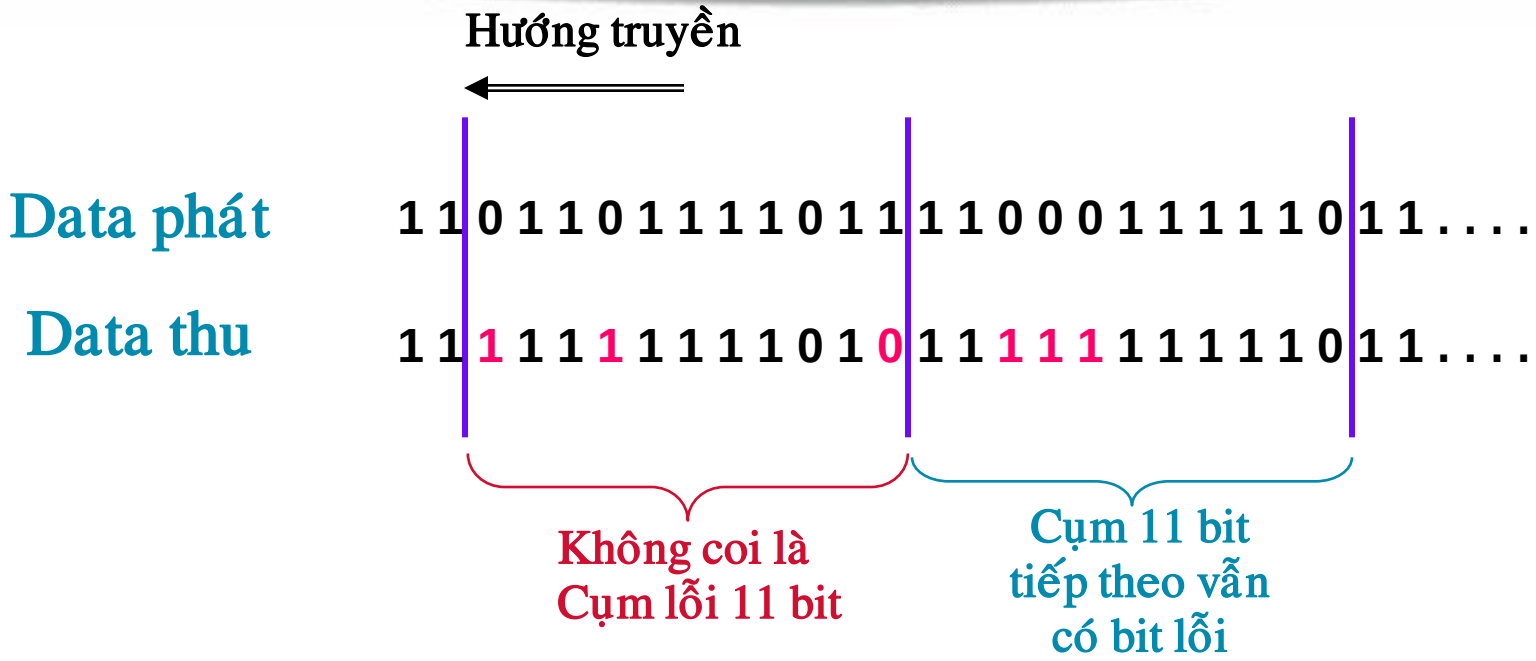
- Độ tin cậy của BCC khoảng 98 %

4.3 Cyclic redundancy check (CRC)



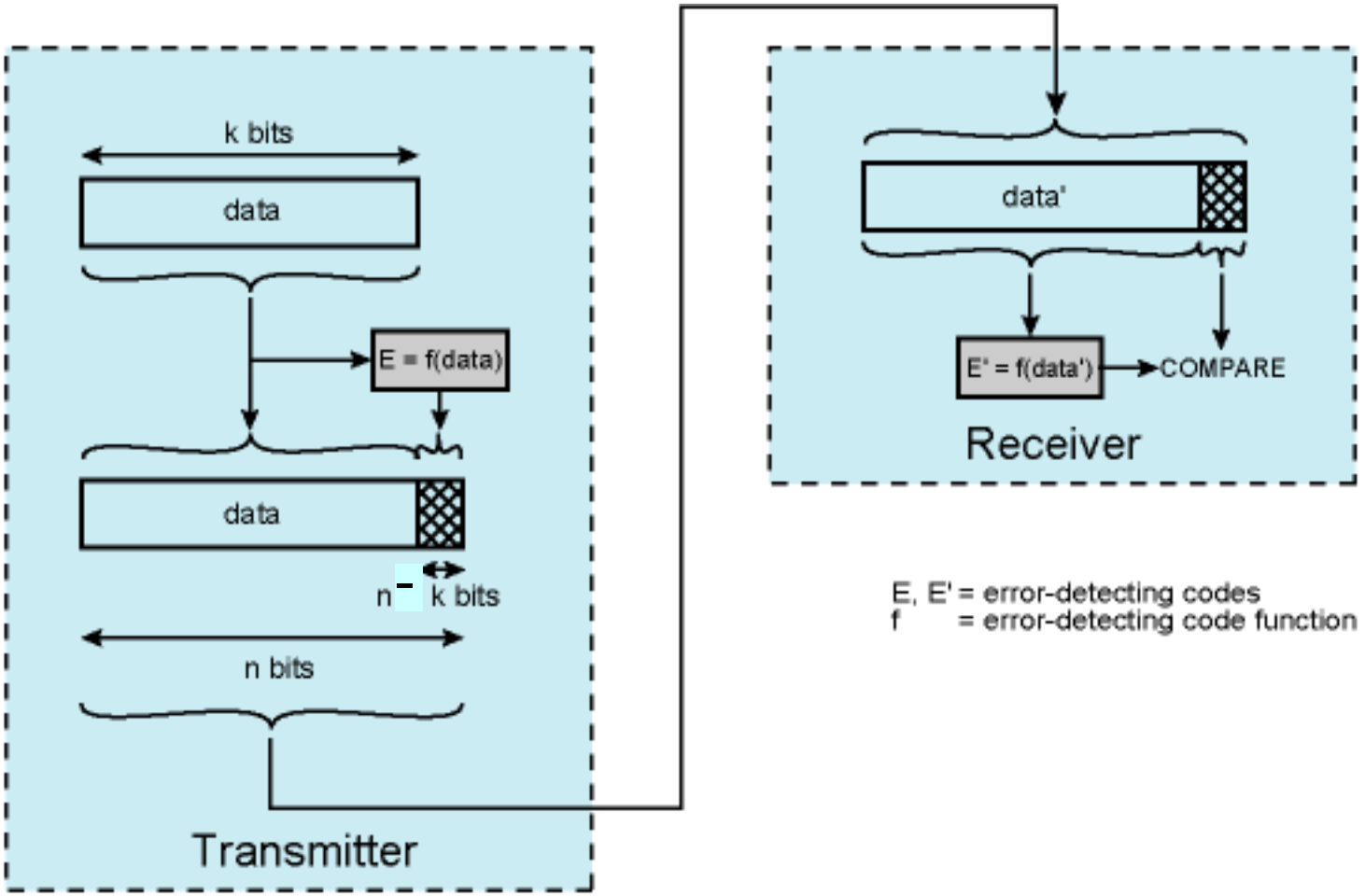
Lỗi cụm

CRC



Parity & BCC không phát hiện được lỗi cụm

CRC


k bits
data
E = f(data)
data
n - k bits
n bits
Transmitter
data'
E' = f(data')
COMPARE
Receiver
E, E' = error-detecting codes
f = error-detecting code function

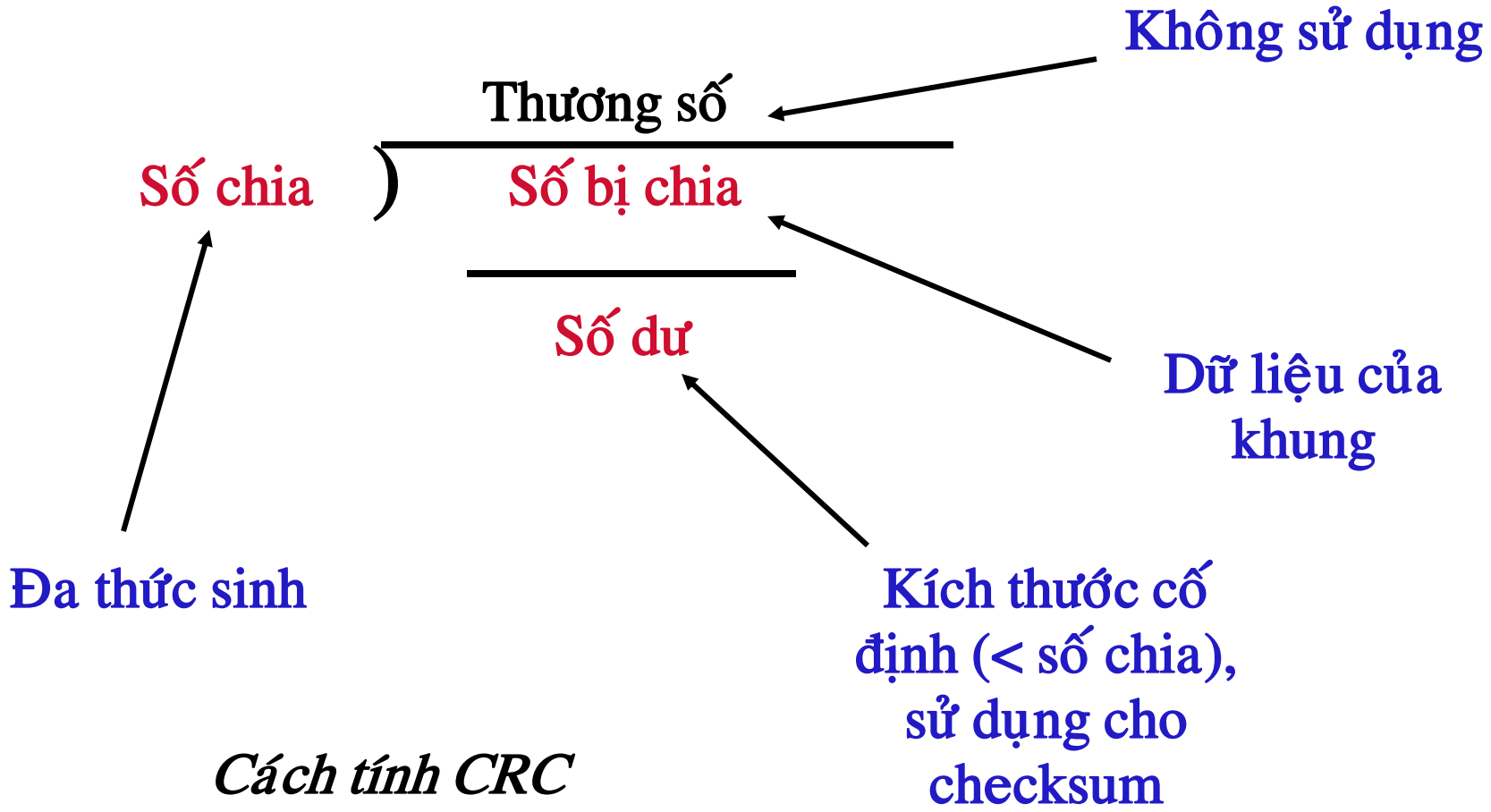
4.3 Cyclic redundancy check (CRC)

CRC sử dụng phép chia toàn bộ dãy bit data của khung cho dãy bit của đa thức sinh (n). Phần dư là CRC(n-1).

Số chia	Số bị chia	Kết quả
0	0	0
0	1	1
1	0	1
1	1	0

Phép chia Modulo-2 (XOR)

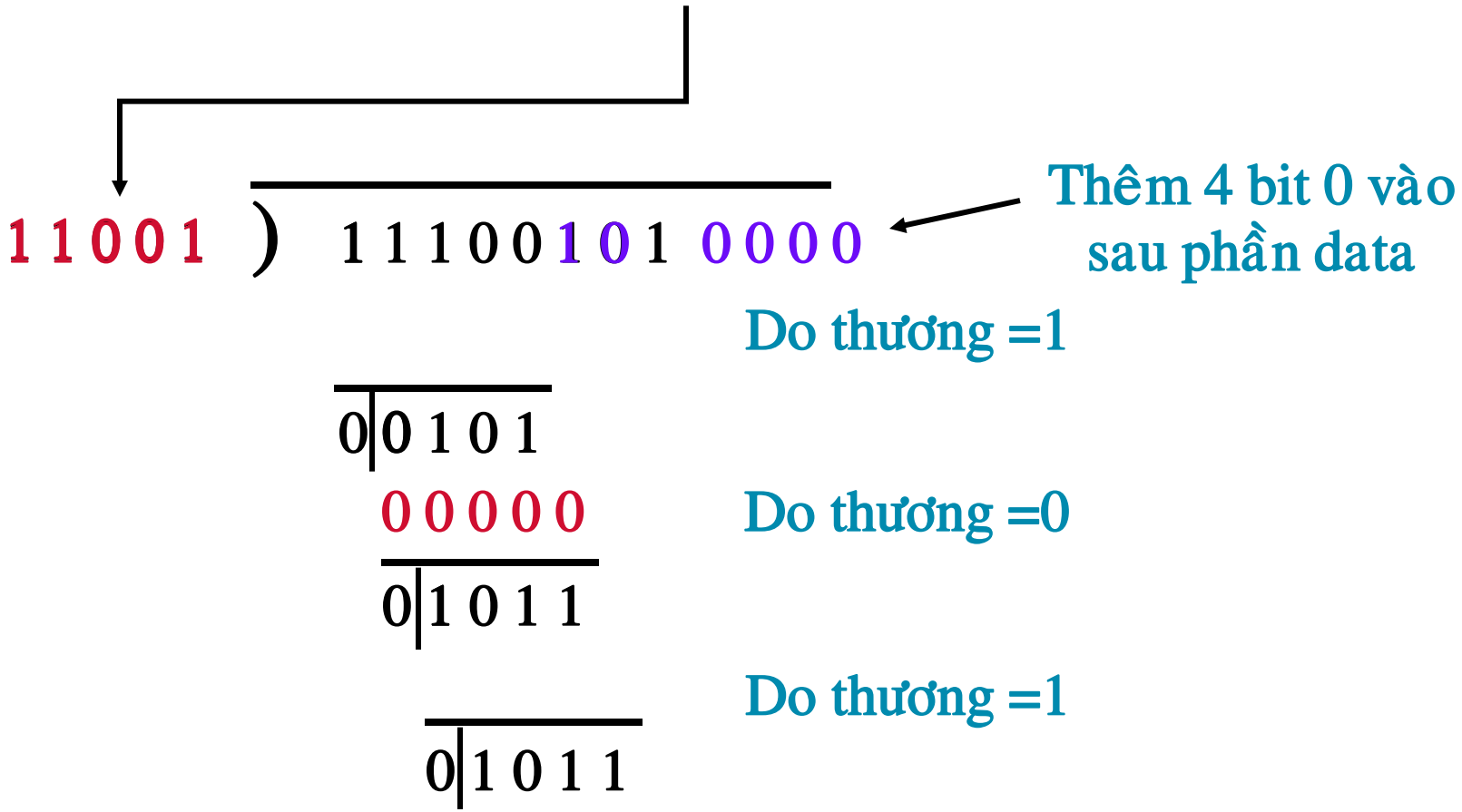
4.3 Cyclic redundancy check (CRC)



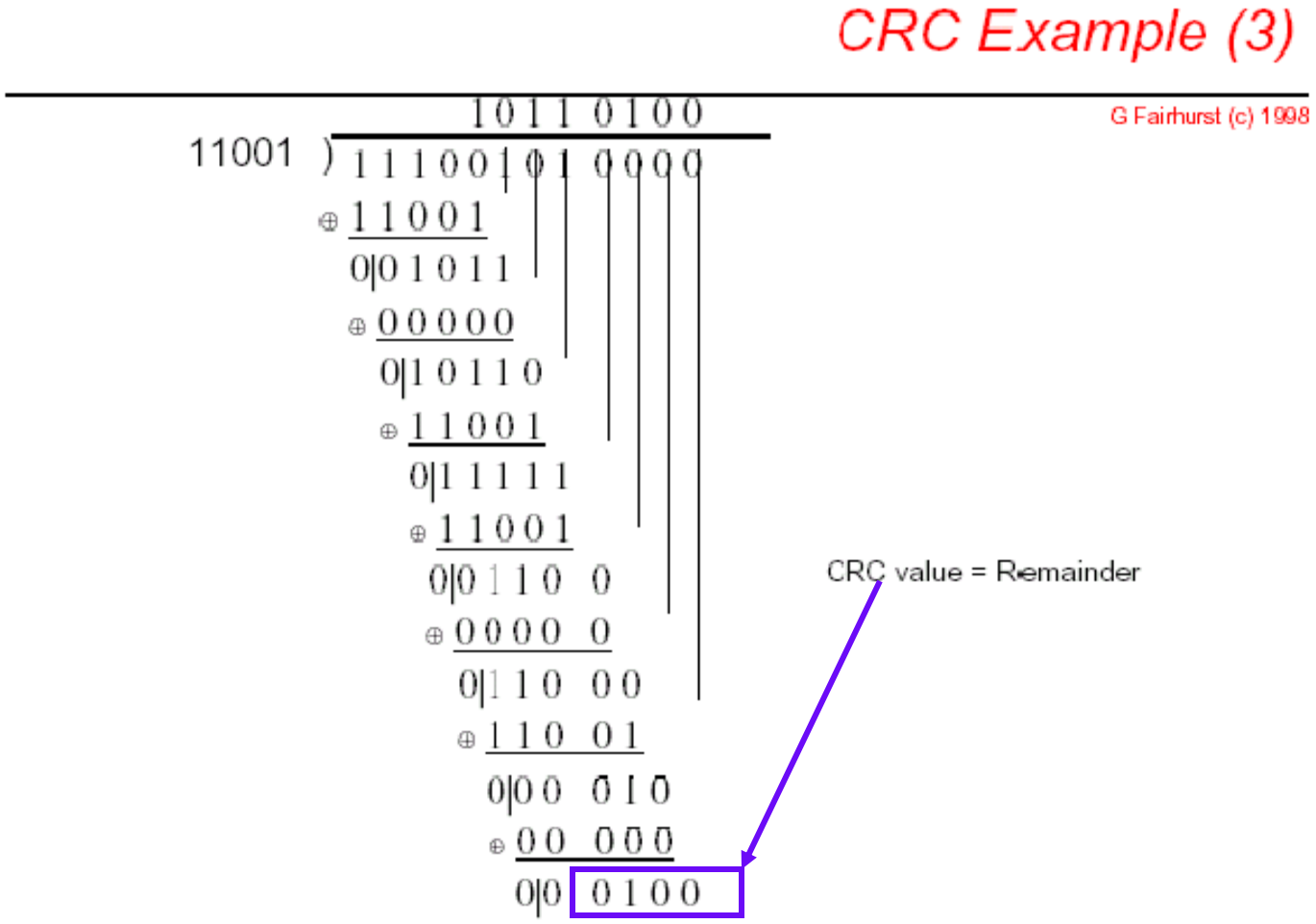
Cách tính CRC

4.3 Cyclic redundancy check (CRC)

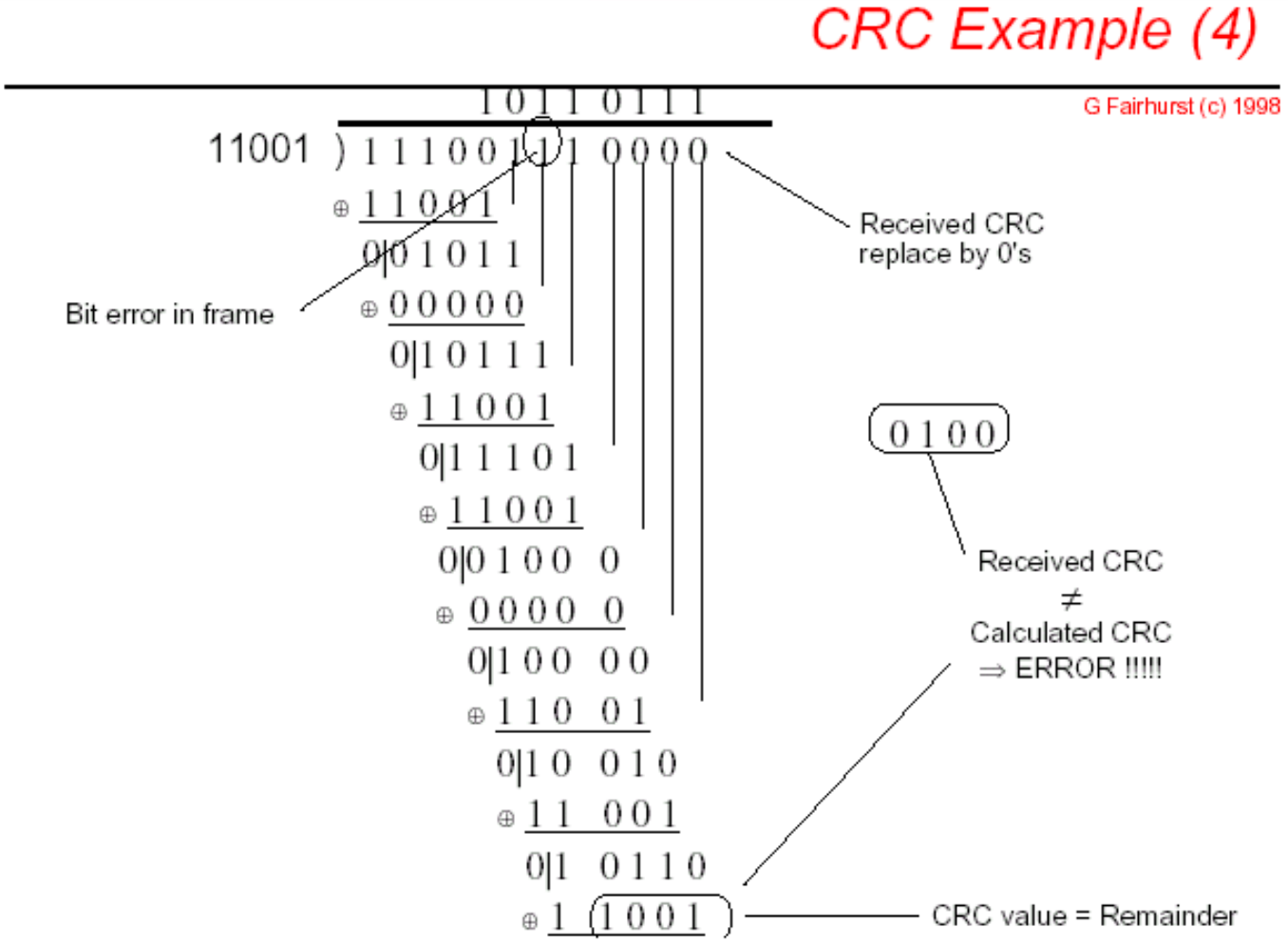
Ví dụ 1: dãy bit là *11100101*, đa thức sinh là $x^4 + x^3 + 1$



4.3 Cyclic redundancy check (CRC)



4.3 Cyclic redundancy check (CRC)



4.3 Cyclic redundancy check (CRC)

CRC Generator Polynomials

G Fairhurst (c) 1998

CRC-16 = $x^{16} + x^{12} + x^5 + x$

CRC-CCITT = $x^{16} + x^{12} + x^5 + 1$

CRC-32 = $x^{32} + x^{26} + x^{16} + x^{12} + x^{11} + x^{10} + x^5 + x^4 + x^2 + x + 1$

4.3 Cyclic redundancy check (CRC)

Cyclical Redundancy Check

G Fairhurst (c) 1998

Properties of a CRC-16	Synchronous Block/Frame Protocols Implemented in Hardware
1-bit error	100% Detected
2-bit errors	100% Detected
Odd errors	100% Detected
Burst errors < 16 bits	100% Detected
Burst errors exactly 17 bits	99.9969% Detected
All other error bursts	99.9984 % Detected

IV. NÉN DỮ LIỆU

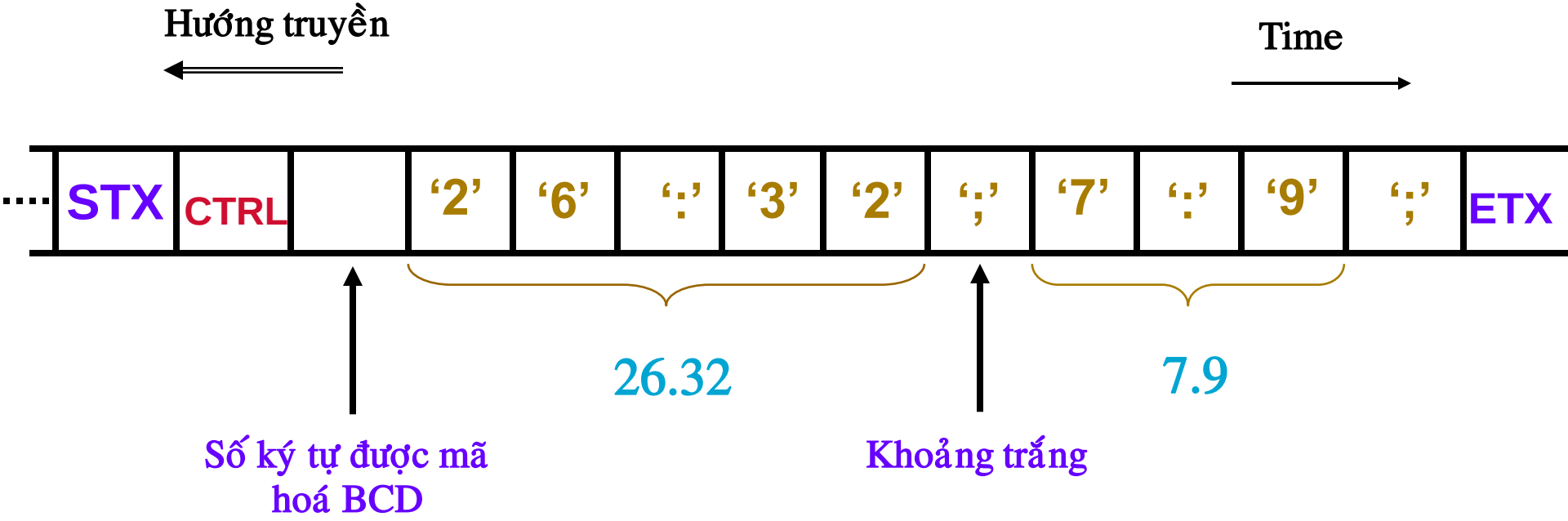
- Packed decimal.
- Mã hoá vi sai.
- Mã hoá Huffman.

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	[	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

Bảng mã ASCII

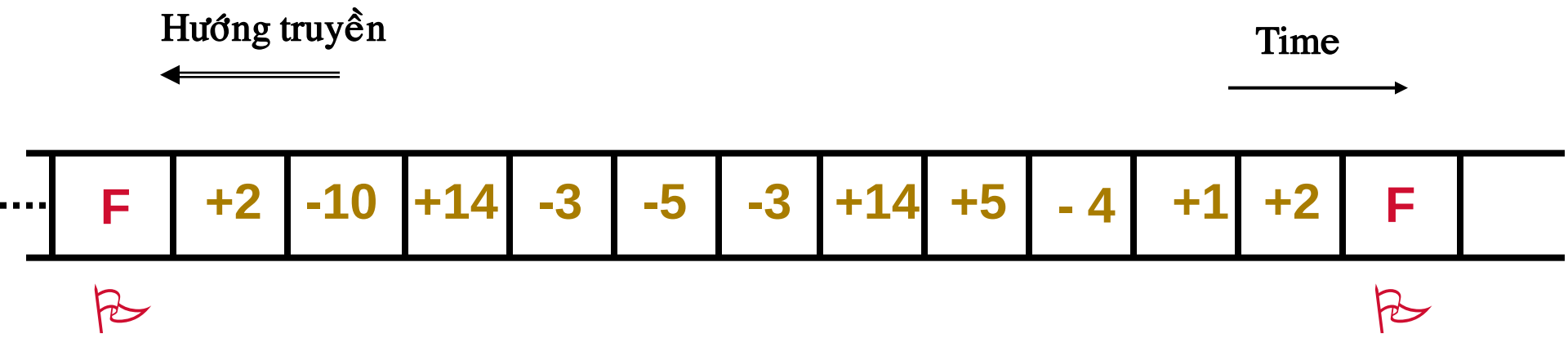
PACKED DECIMAL

- Nếu data chỉ là các số thì sử dụng mã hoá BCD (4 bit) hiệu quả hơn sử dụng ASCII (7 bit).



MÃ HOÁ VI SAI

- Một số loại dữ liệu thực hiện mã hoá độ lệch của phần tử sau so với phần tử trước sẽ hiệu quả hơn mã hóa trị tuyệt đối.



MÃ HOÁ HUFFMAN

- Sinh viên tự nghiên cứu.
- Xem lại mã thống kê tối ưu – môn lý thuyết truyền tin.
- Đọc thêm mã Fano.