

Đại học Sư Phạm Kỹ Thuật Tp.HCM

Khoa Cơ Khí Chế Tạo Máy

Bộ môn Cơ Điện tử



BÀI GIẢNG

VI XỬ LÝ (MICO236929)

01/2023

1-2	Chương 1: Tổng quan về máy tính và điện tử <ul style="list-style-type: none">- Lưu trữ thông tin số trong bộ nhớ, Hệ thống số đếm, bảng mã- Cấu trúc máy tính và vi xử lý- Lưu đồ giải thuật và cấu trúc chương trình, Cấu trúc chương trình tuyến tính và có cấu trúc
3	Chương 2: Giới thiệu vi xử lý, vi điều khiển và hệ thống nhúng <ul style="list-style-type: none">- Tổng quan về vi điều khiển, vi xử lý, hệ thống nhúng- Vòng đời sản phẩm, Chất lượng thiết kế, Phương pháp debug- Ôn tập về ngôn ngữ C và ứng dụng ngôn ngữ C trong lập trình vi điều khiển
4	Chương 3: Giới thiệu vi điều khiển dựa trên ARM cortex <ul style="list-style-type: none">- Cấu trúc ARM- Phần mềm lập trình Keil uVision- Ngôn ngữ Assembly cho lập trình ARM

5-9

Chương 4: Giao tiếp vi điều khiển ARM và ngoại vi

- Giới thiệu vi điều khiển STM32F103 và cấu trúc Input/output port
- Các phương pháp mở rộng I/O port các IC hỗ trợ mở rộng PORT
- Giao tiếp nút nhấn đơn và chương trình ứng dụng và debug
- Giao tiếp keypad và chương trình ứng dụng và debug
- Giao tiếp led 7 đoạn và chương trình ứng dụng và debug
- Giao tiếp LCD và chương trình ứng dụng và debug
- Giao tiếp relay và chương trình ứng dụng và debug
- Tương quan giữa ngôn ngữ C và ngôn ngữ Assembly
- Địa chỉ Indexed và con trỏ và chương trình ứng dụng
- Strings và chương trình ứng dụng
- Structure và chương trình ứng dụng
- Lập trình theo phương pháp máy trạng thái và chương trình ứng dụng
- Ma trận và đồ họa chương trình ứng dụng

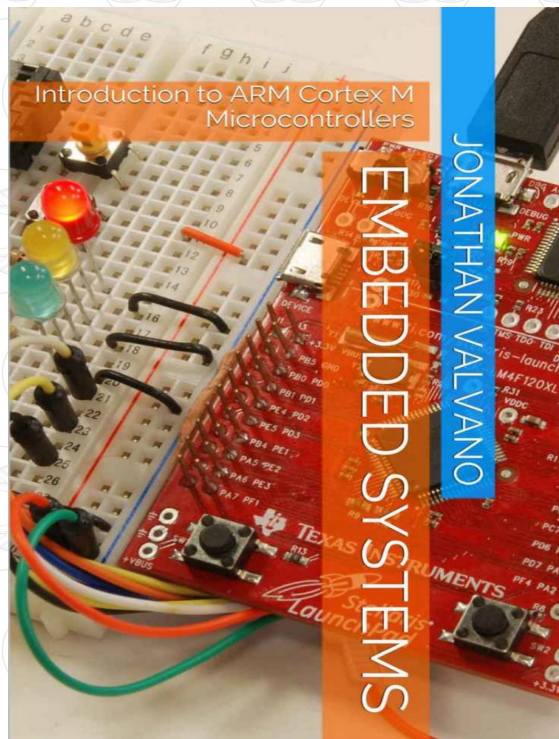
5-9 (tt)	Chương 4: Giao tiếp vi điều khiển ARM và ngoại vi (tt) <ul style="list-style-type: none">- Bảng tra và chương trình ứng dụng- Stack và chương trình ứng dụng- Các lệnh số học và chương trình ứng dụng
10	Chương 5: Giao tiếp nối tiếp và giao tiếp song song 5.1 Khái niệm về truyền dữ liệu nối tiếp và truyền dữ liệu song song 5.2 UART và ứng dụng 5.3 SSI và ứng dụng
11-12	Chương 6: Ngắt và hệ thống thời gian thực <ul style="list-style-type: none">- Khái niệm ngắt (interrupt) và thời gian thực (real-time)- Ứng dụng của interrupt giải quyết một số bài toán điều khiển- Giới thiệu về Timer/Counter- Ứng dụng timer giải quyết một số bài toán điều khiển



13-14	Chương 7: ADC, DAC và ứng dụng <ul style="list-style-type: none">- Khái niệm, nguyên tắt hoạt động, cách sử dụng bộ ADC- Ứng dụng của ADC- Khái niệm, nguyên tắt hoạt động, cách sử dụng bộ ADC- Ứng dụng của ADC
15	Ôn tập
Thời gian	Kiểm tra - đánh giá môn học
Tuần 5	Kiểm tra quá trình lần 1: Nội dung chương 1+2
Tuần 9	Kiểm tra quá trình lần 2: Nội dung chương 4
Tuần 10-11	Bài tập lớn: Thiết kế và lập trình mạch điều khiển sử dụng vi điều khiển (mô phỏng)
(theo lịch phòng đào tạo)	Thi cuối kỳ 90 phút. Trắc nghiệm và tự luận (tất cả nội dung đã học)

[1] JONATHAN W. VALVANO, **EMBEDDED SYSTEMS: INTRODUCTION TO ARM® CORTEX-M MICROCONTROLLERS** (VOLUME 1,2) – ([link](#))

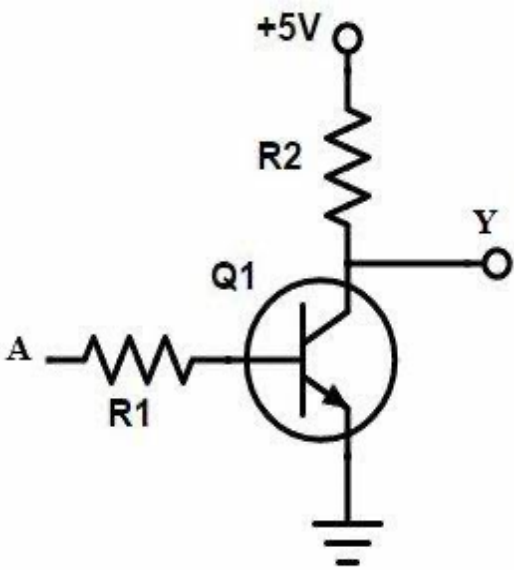
[2] Nguyễn Quang Nam, **Hệ thống giao tiếp thời gian thực với vi điều khiển ARM Cortex-M**



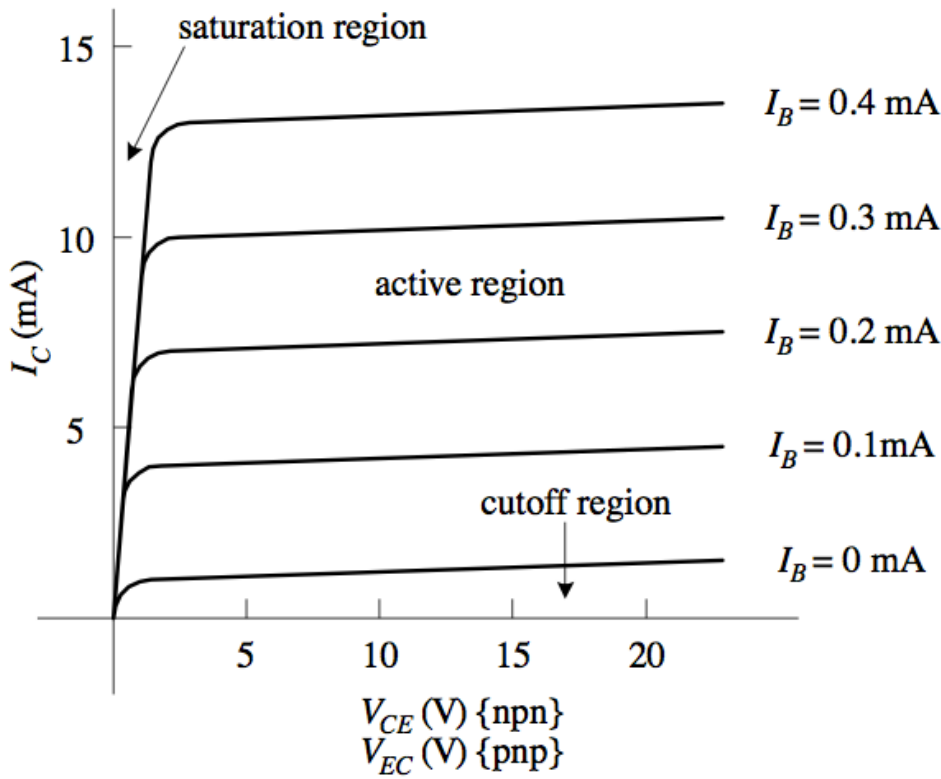
Nội dung chương 1

- Tổng quan về điện tử
- Công nghệ CMOS và số nhị phân
- Mạch logic
- Lưu trữ thông tin số trong bộ nhớ
- Hệ thống số đếm, bảng mã
- Cấu trúc máy tính và vi xử lý
- Lưu đồ giải thuật và cấu trúc chương trình
- Cấu trúc chương trình tuyến tính và có cấu trúc

1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic



Hình 1. BJT NOT gate

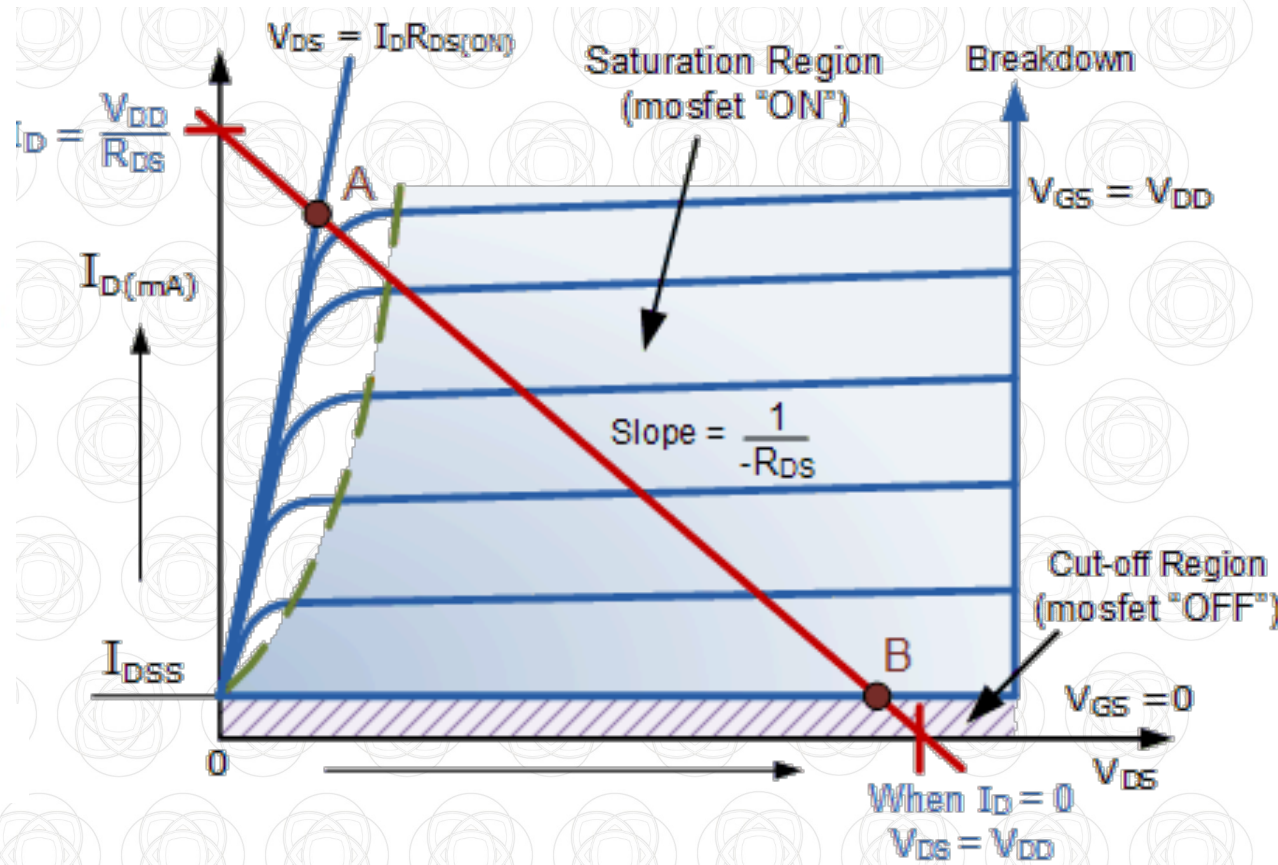
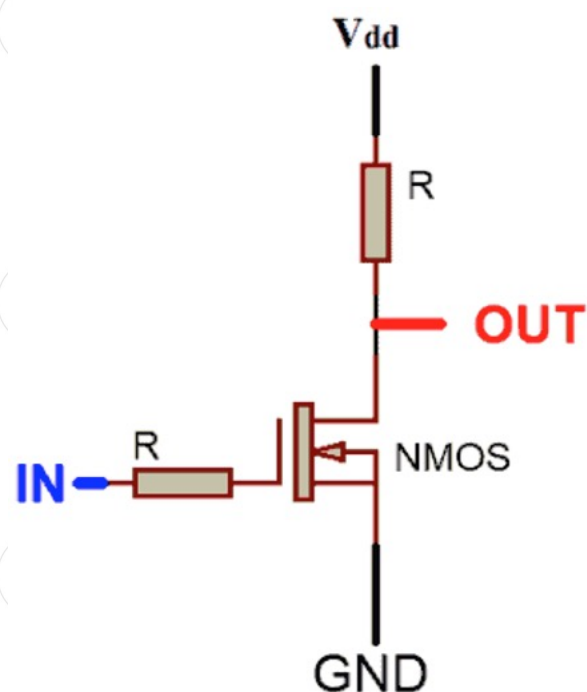


Câu hỏi:

1.1. Nguyên lý hoạt động mạch điện hình 1?

1.2. Sử dụng BJT 2N2222 $R1=R2=10k$, phân tích công suất tiêu thụ và dòng điện qua BJT? (tham khảo datasheet [2N2222](#))

1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

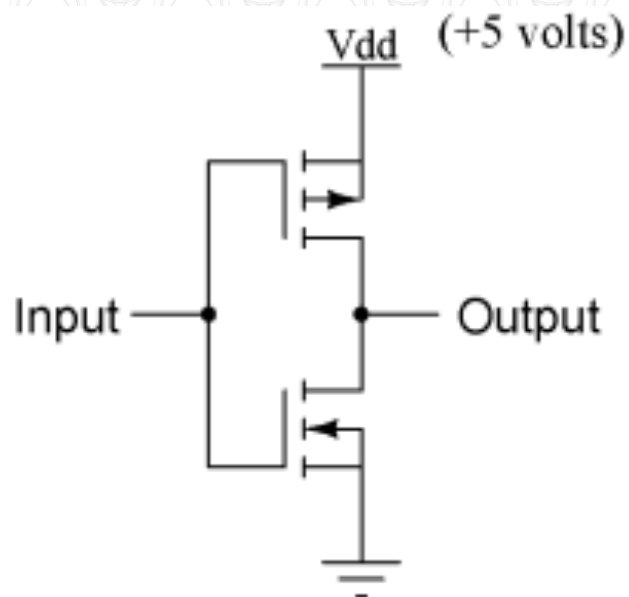


Hình 2. FET NOT gate

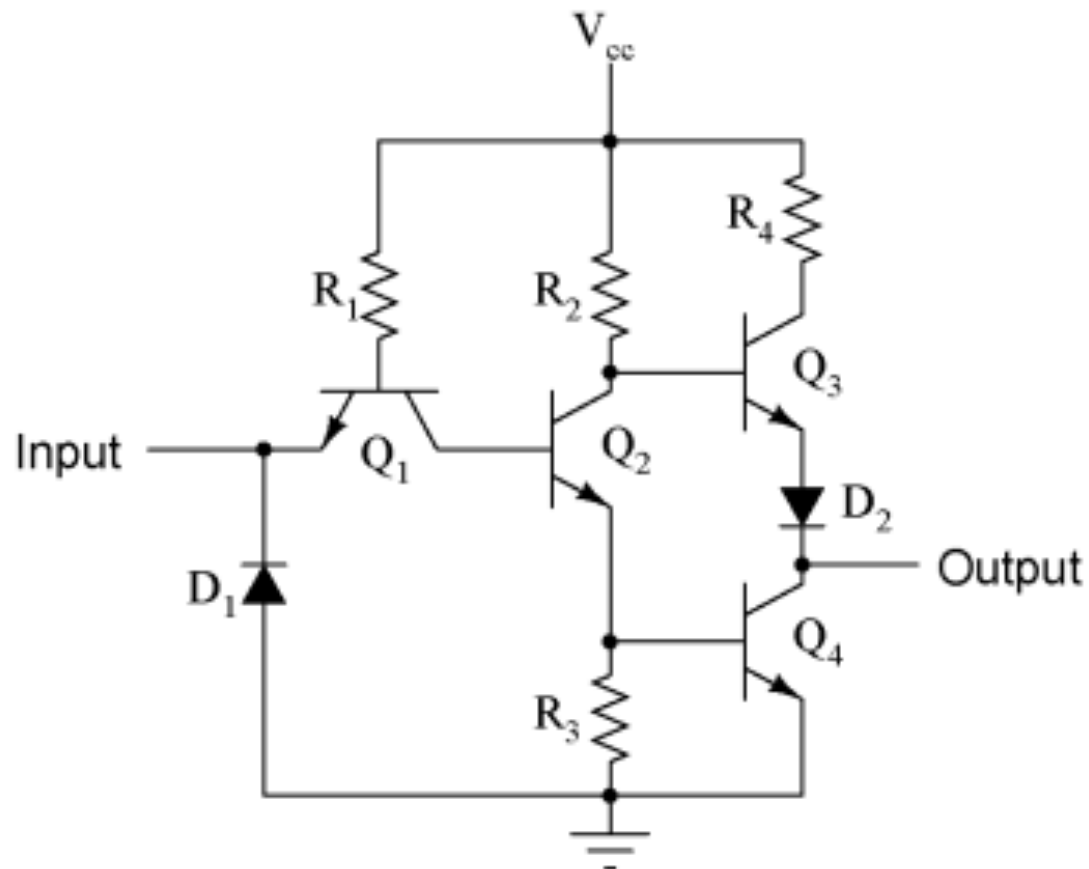
1.3. Nguyên lý hoạt động mạch điện hình 2?

1.4. Sử dụng FET SI2301 R=10k, VDD=5V, phân tích công suất tiêu thụ và dòng điện qua FET? (tham khảo datasheet [SI2301](#))

1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic



Hình 3. CMOS NOT gate



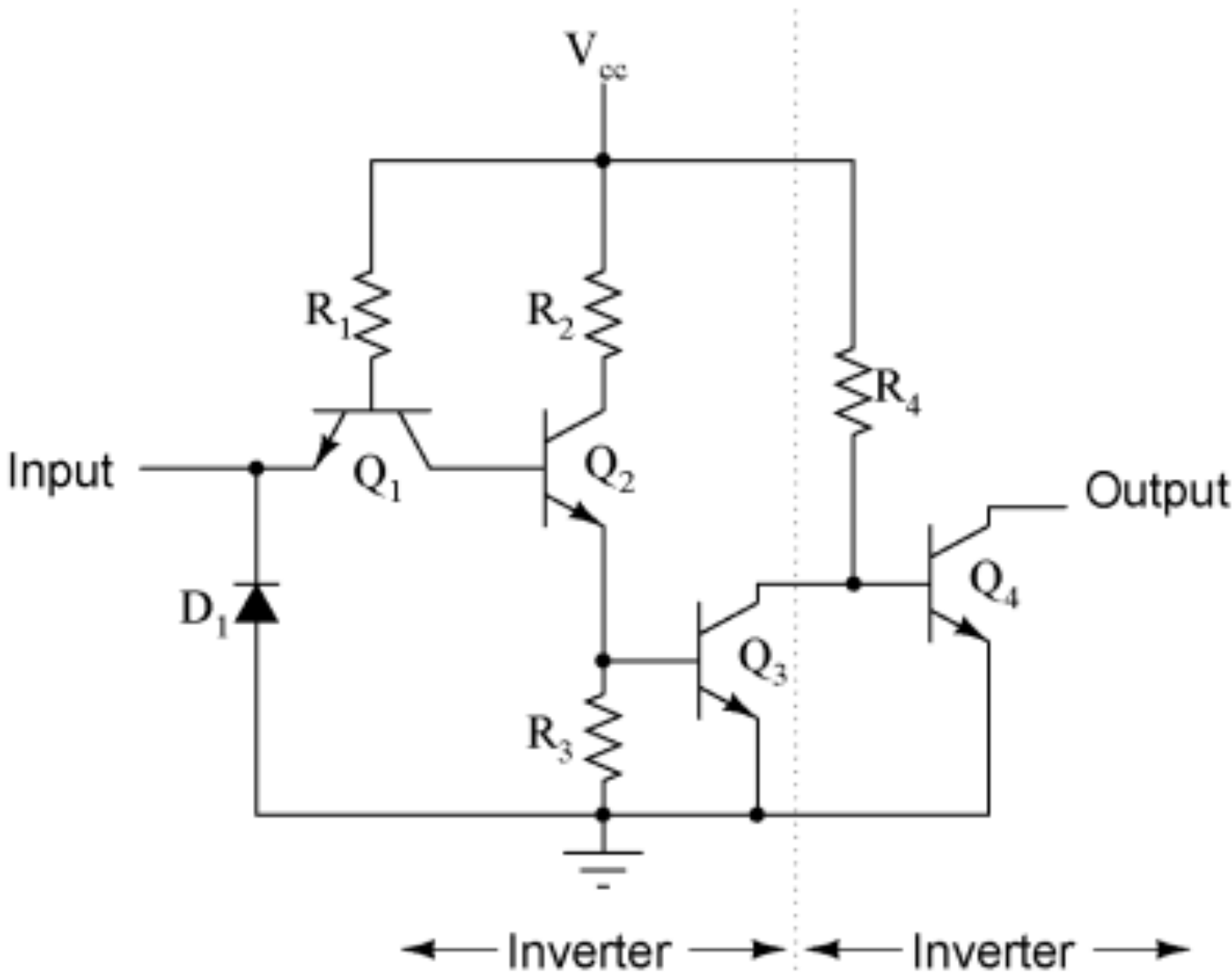
Hình 4. TTL NOT gate

1.5. Vì sao cần mạch buffer?

1.6. Có mấy loại mạch buffer? Vẽ sơ đồ nguyên lý và giải thích?

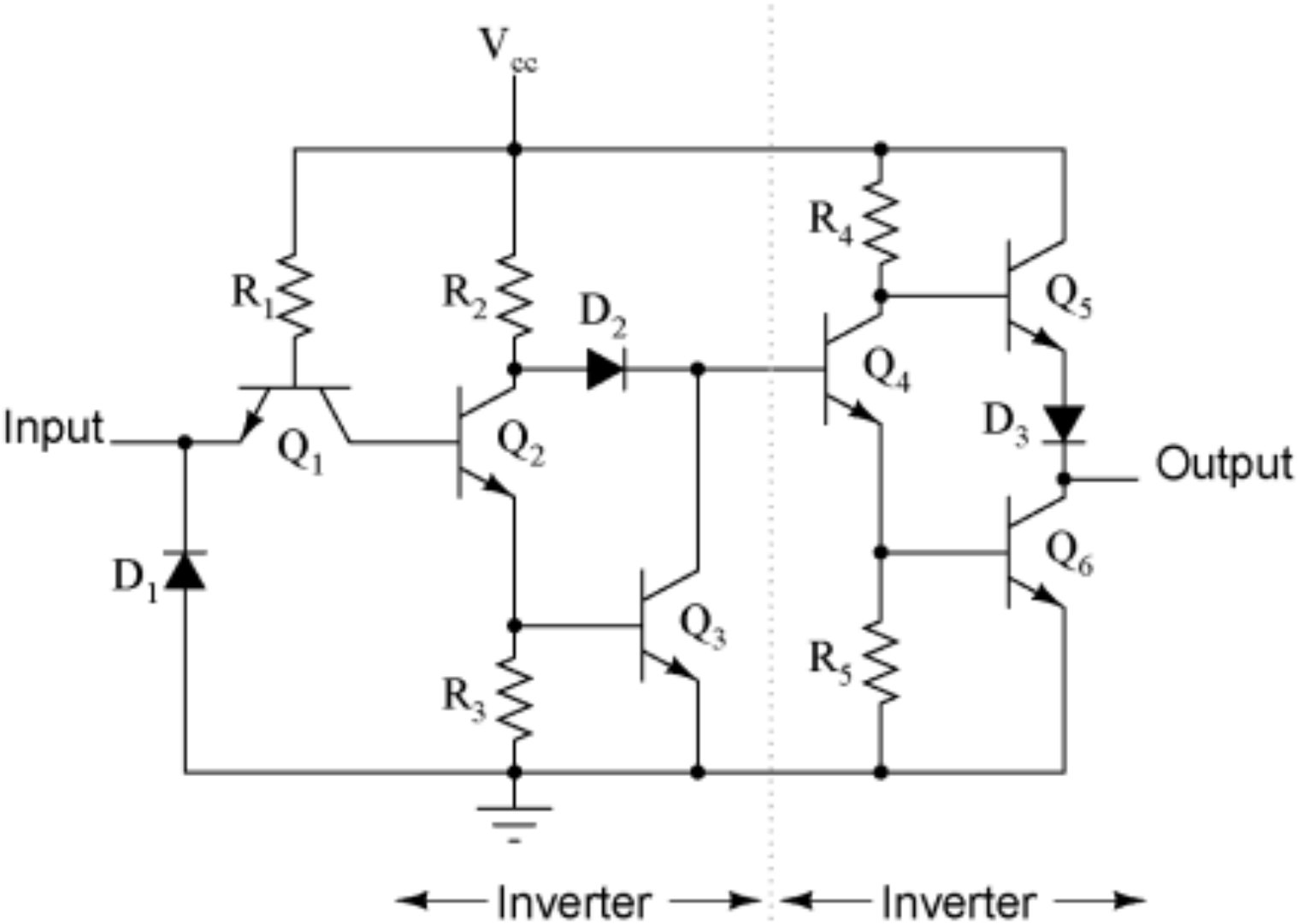
1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

Buffer circuit with open-collector output



1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

Buffer circuit with totem pole output



1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

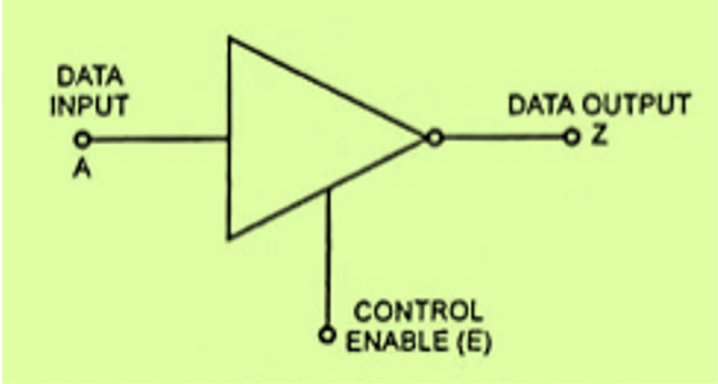
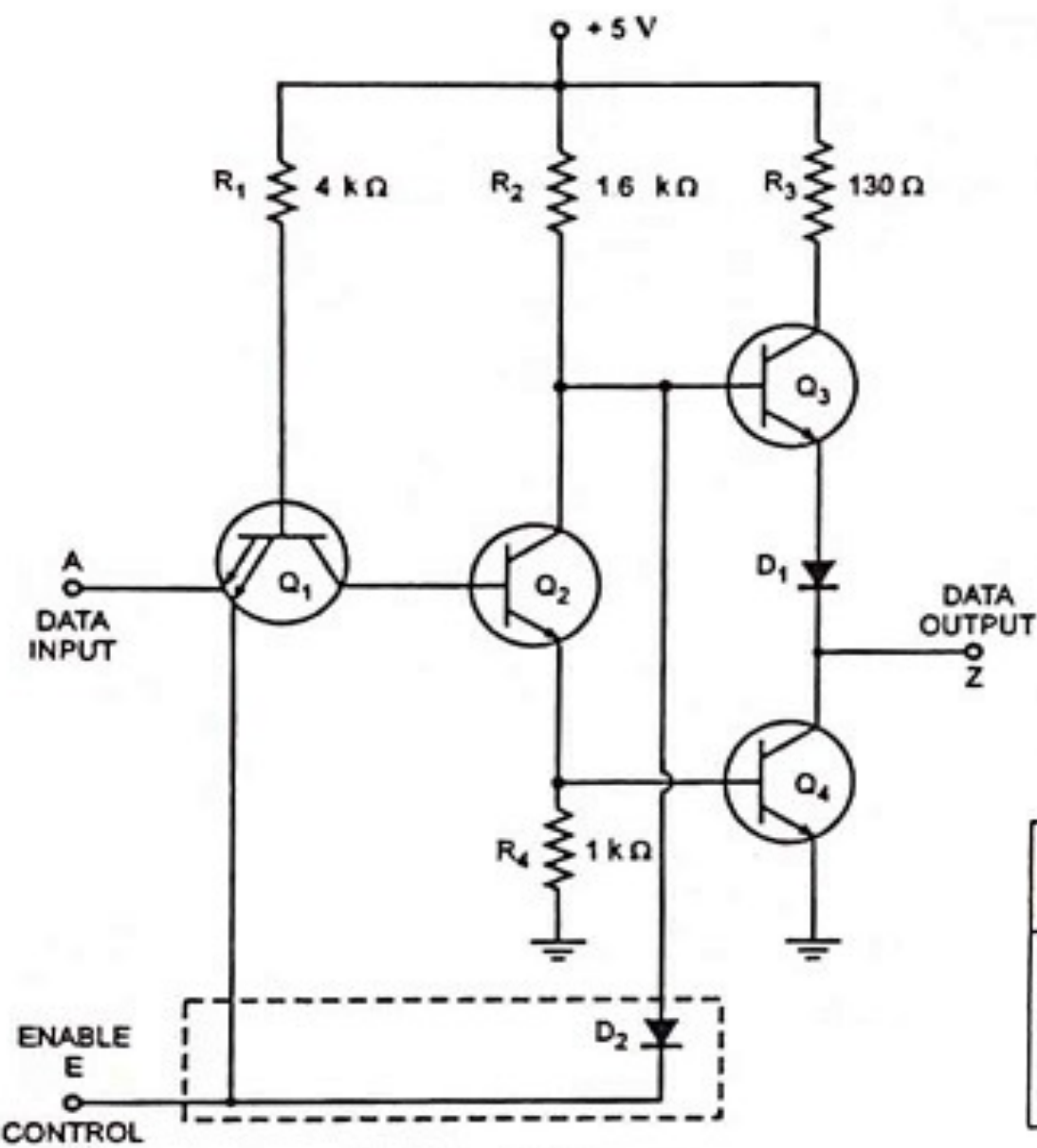
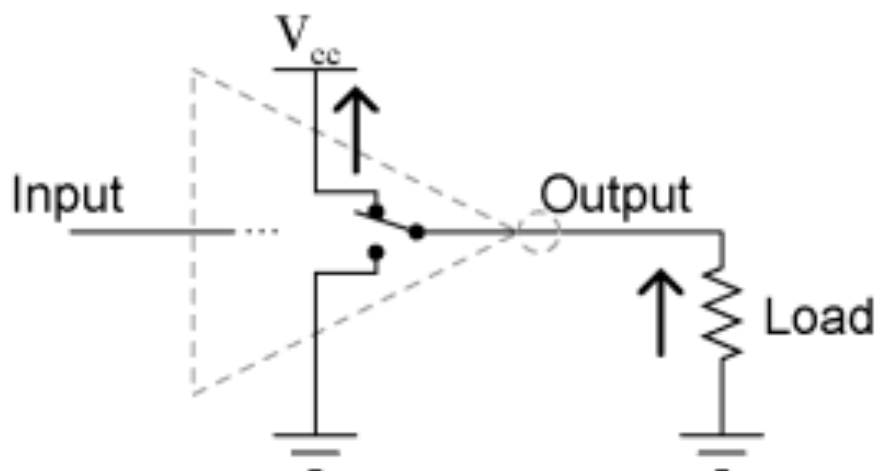


TABLE 47.2 Truth Table For TTL Inverter

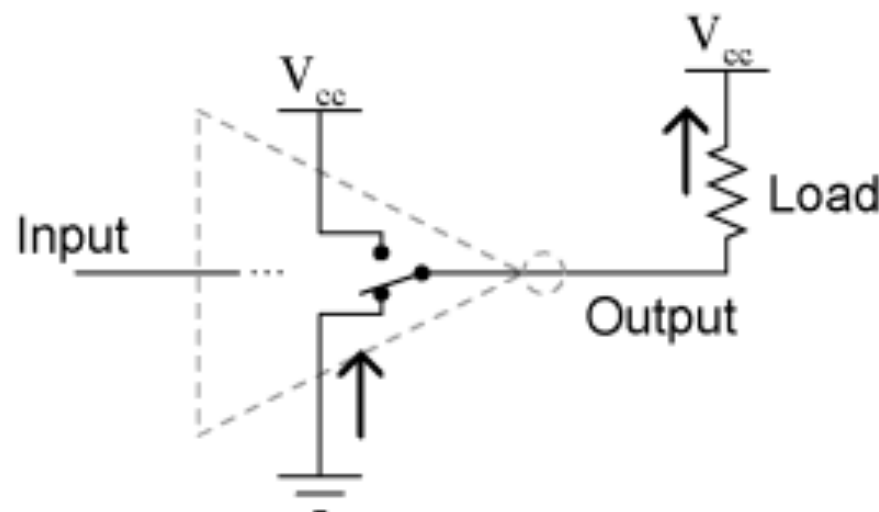
Data Input, A	Control E	Data Output
0	0	high Z
1	0	high Z
0	1	1
1	1	0

1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

*Simplified gate circuit **sourcing** current*

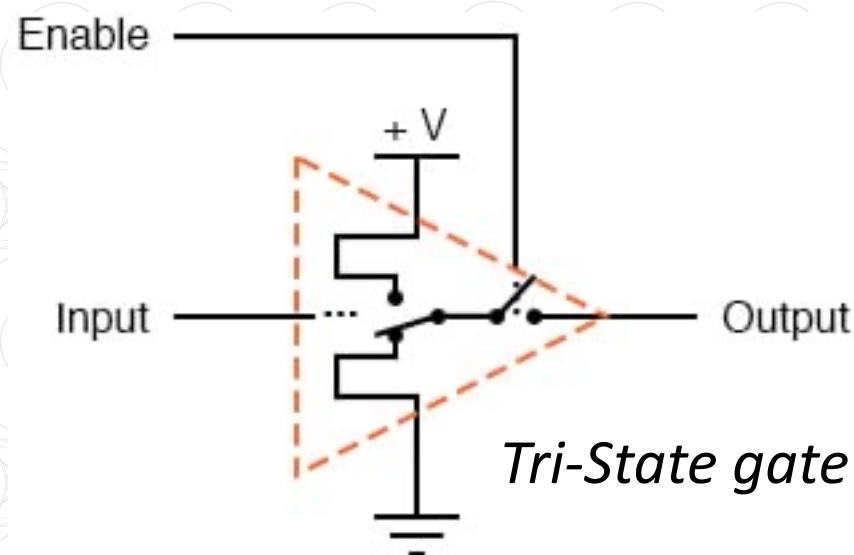


*Simplified gate circuit **sinking** current*



Câu hỏi ôn tập:

1. Giải thích hoạt động ngõ ra cực thu hở, ngõ ra đẩy kéo (totem-pole), ngõ ra 3 trạng thái
2. Giải thích khái niệm cấp dòng, rút dòng và trạng thái tổng trở cao.



1.1. Tổng quan về điện tử: Từ linh kiện rời rạc đến cổng logic

Table 7. Current characteristics

ST Datasheet: DS5319

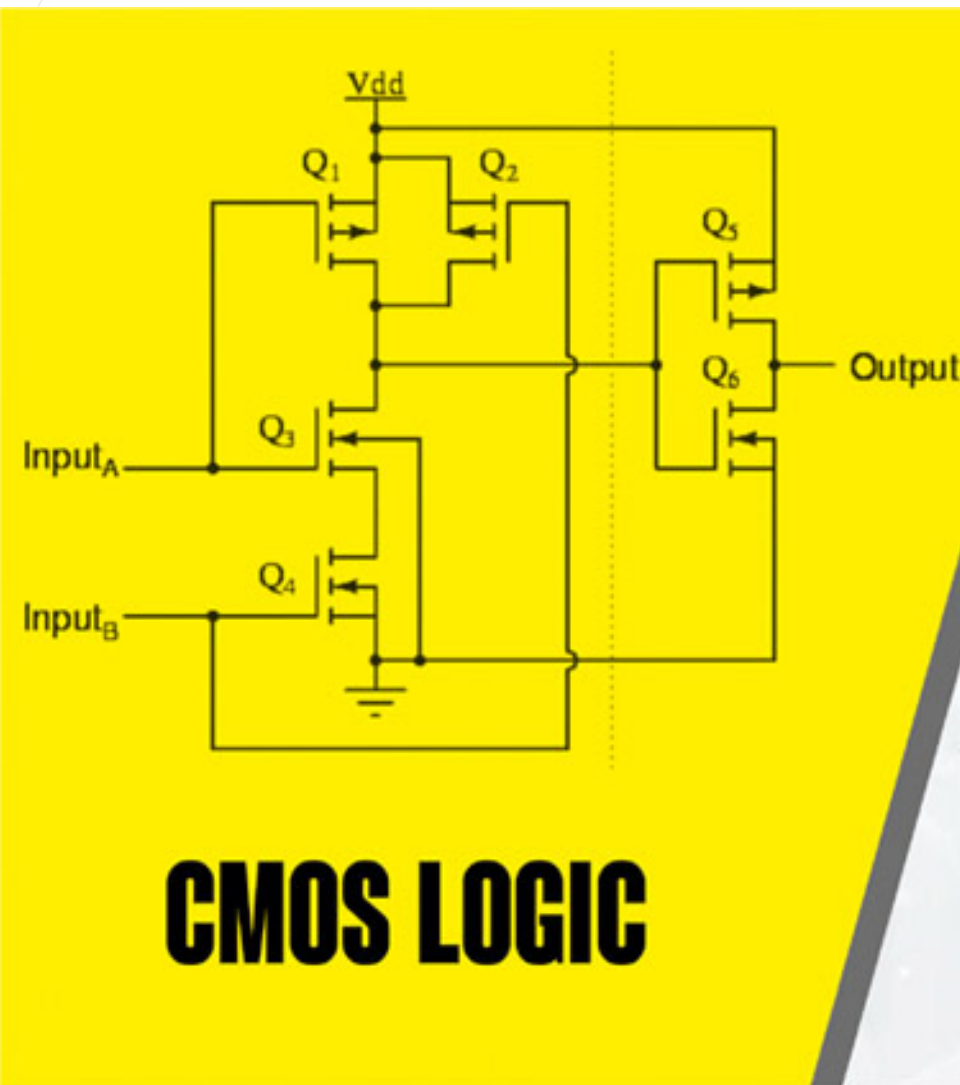
Symbol	Ratings	Max.	Unit
I_{VDD}	Total current into V_{DD}/V_{DDA} power lines (source) ⁽¹⁾	150	mA
I_{VSS}	Total current out of V_{SS} ground lines (sink) ⁽¹⁾	150	
I_{IO}	Output current sunk by any I/O and control pin	25	
	Output current source by any I/Os and control pin	− 25	
$I_{INJ(PIN)}^{(2)}$	Injected current on five volt tolerant pins ⁽³⁾	−5/+0	
	Injected current on any other pin ⁽⁴⁾	± 5	
$\Sigma I_{INJ(PIN)}$	Total injected current (sum of all I/O and control pins) ⁽⁵⁾	± 25	

Câu hỏi:

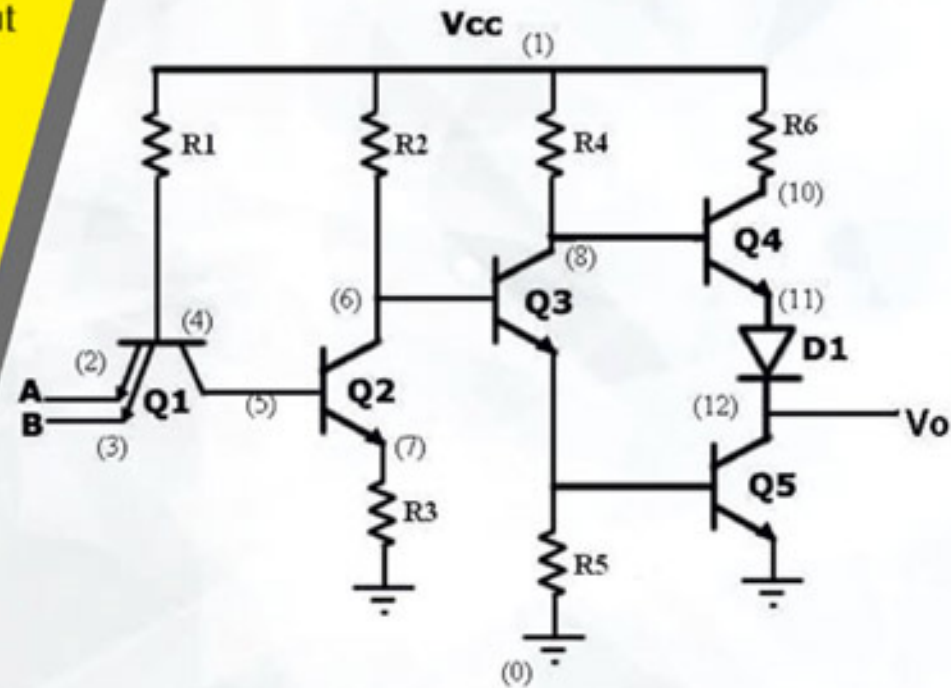
1.7. Nếu STM32F103C8 gắn tải là đèn LED có rơi áp 1,5V thì điện trở hạn dòng cho LED là bao nhiêu?

1.8. Nếu tải là LED 7 đoạn có dòng mỗi LED 5mA, STM32F103C8 điều khiển tối đa được bao nhiêu LED 7 đoạn?

1.2 Tổng quan về điện tử: Công nghệ CMOS



TTL LOGIC

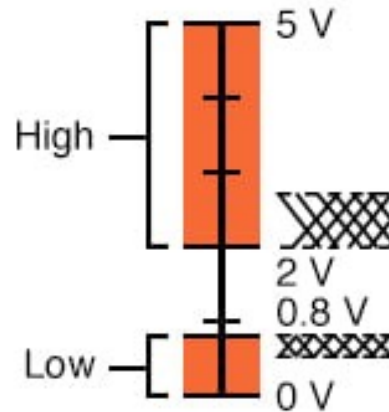


[Comparison between CMOS and TTL Logic - Which is Better and Why? \(circuitdigest.com\)](http://circuitdigest.com)

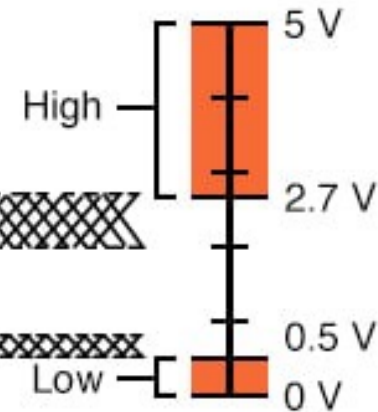
[TTL vs. CMOS: Integrated Circuit Logic Families \(ourpcb.com\)](http://ourpcb.com)

1.2 Tổng quan về điện tử: Công nghệ CMOS

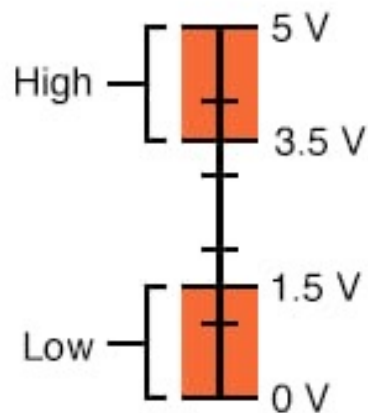
Acceptable TTL Gate
Input Signal Levels



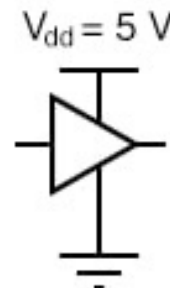
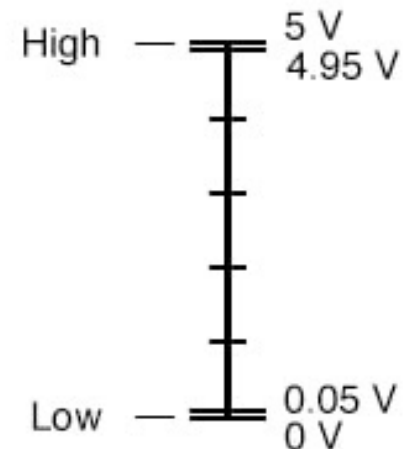
Acceptable TTL Gate
Output Signal Levels



Acceptable CMOS Gate
Input Signal Levels



Acceptable CMOS Gate
Output Signal Levels



1.2 Tổng quan về điện tử: Công nghệ CMOS

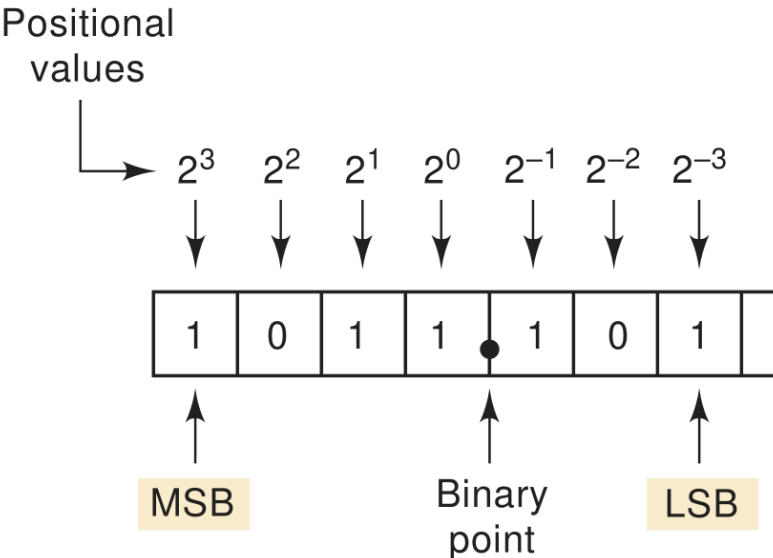
	CMOS	TTL
Cost	More expensive than TTL but are less costly at a system level. Also, they are smaller.	Less expensive but not as economical as CMOS.
Propagation Delays	Between 20 and 50 nS,	Around 10 nS.
Rise and Fall Times	Longer rise and fall times, resulting in simpler, less costly digital signals.	Shorter rise and fall times.
Voltage Levels	Part of being more energy efficient includes having low voltage levels, usually between 0 to 1/3 VDD at a low level and 2/3VDD to VDD at high levels.	TTL chips have a high operating voltage level, typically ranging from 4.75 V to 5.25 V.
Current Draw	CMOS chips require less current, which limits power consumption. Therefore, they make it easier to design circuits with better power management.	Like voltage, TTL chips require more current.

1.2 Tổng quan về điện tử: Công nghệ CMOS

Electromagnetic Disruptions	CMOS chip components are more sensitive to electromagnetic disruptions.	TTL logic components are less susceptible to electromagnetic disruptions.
Standard Load Connections (Fan-out)	The number of loads connectable to the gate output under normal operation is 50.	The number here is significantly lower. Only 10.
Standard Input Connections (Fan-in)	The number of connectable inputs to the gate is only 10.	TTL has the upper hand because the number is slightly higher (12 - 14).
Noise Immunity	Better noise immunity.	Not as immune as CMOS chips to noise.
Basic Logic Gates	Feature NAND and NOR gates to carry out logic functions.	It contains only NAND gates
Design	Simpler design with n-type and p-type metal oxide field-effect transistors.	More complex design with bipolar junction transistors.

1.3. Hệ thống đếm

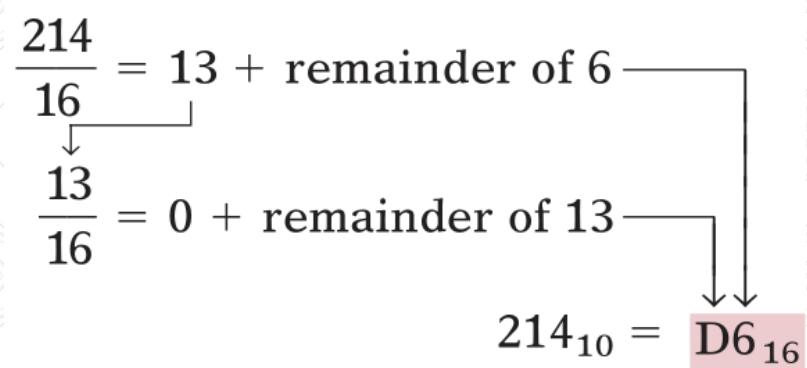
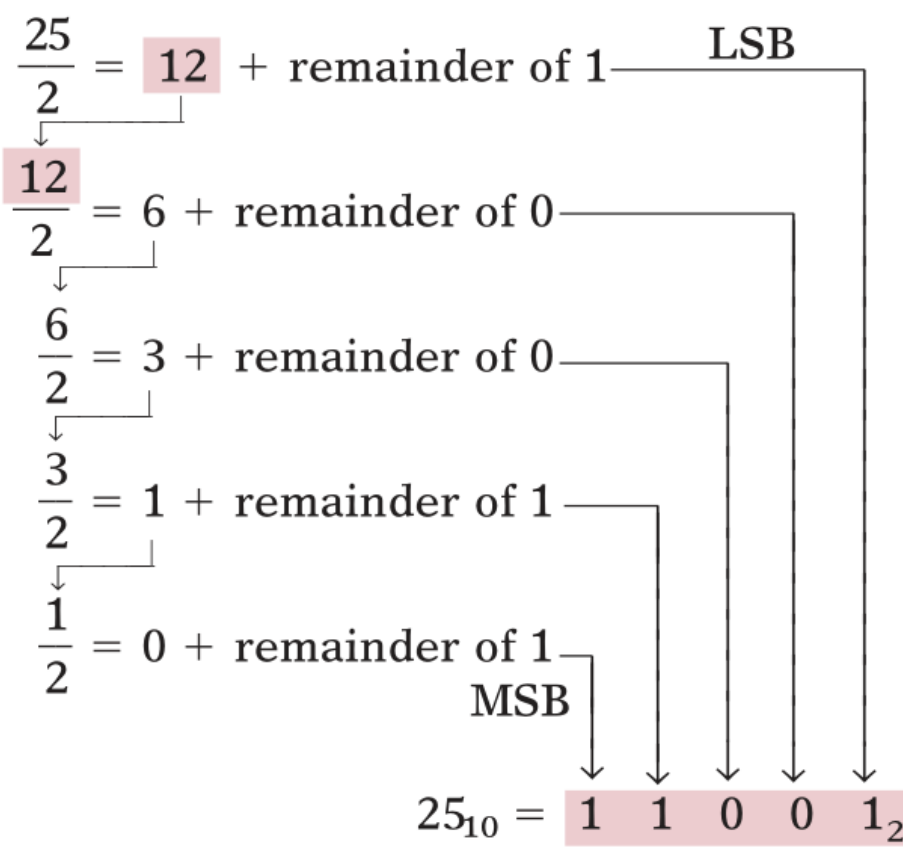
Decimal	Binary	Hexadecimal	BCD	GRAY
0	0	0	0000	0000
1	1	1	0001	0001
2	10	2	0010	0011
3	11	3	0011	0010
4	100	4	0100	0110
5	101	5	0101	0111
6	110	6	0110	0101
7	111	7	0111	0100
8	1000	8	1000	1100
9	1001	9	1001	1101
10	1010	A	0001 0000	1111
11	1011	B	0001 0001	1110
12	1100	C	0001 0010	1010
13	1101	D	0001 0011	1011
14	1110	E	0001 0100	1001
15	1111	F	0001 0101	1000



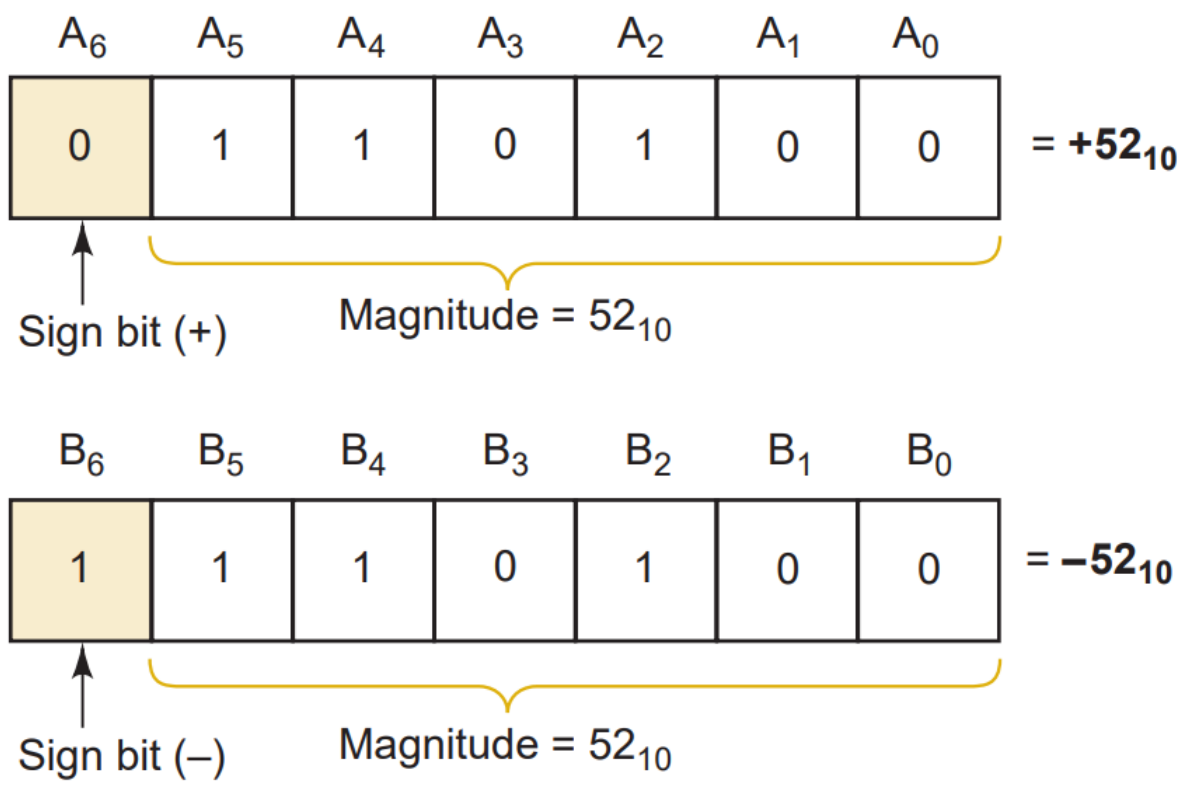
Câu hỏi:

- 1.9. Giải thuật chuyển đổi từ các hệ đếm sang hệ thập phân?
- 1.10. Giải thuật chuyển đổi từ hệ thập phân sang các hệ khác?

1.3. Hệ thống đếm



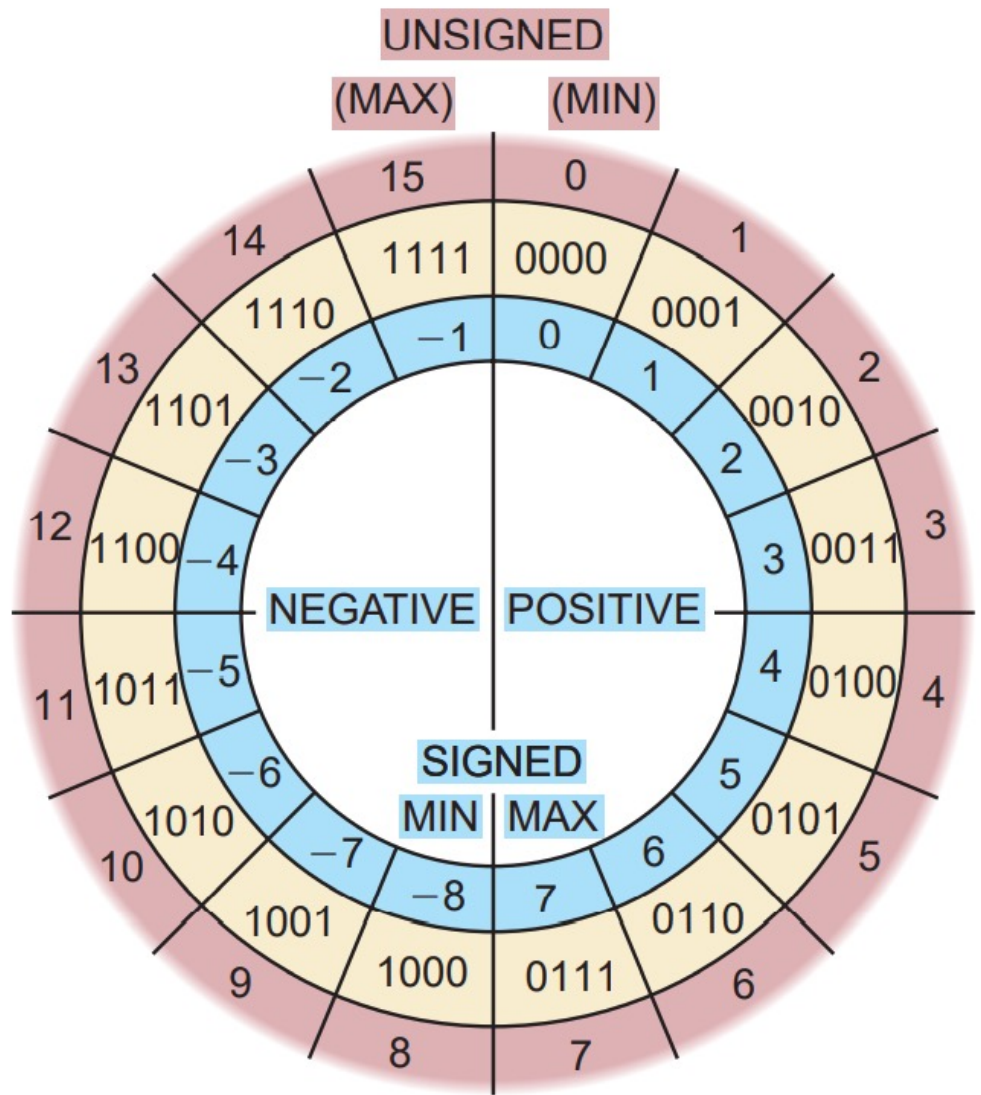
1.4. Lưu trữ thông tin số trong bộ nhớ



Câu hỏi:

- 1.11. Giải thích data bank? Cách đánh địa chỉ ô nhớ trong máy tính?
- 1.12. Cách lưu trữ số âm? Cách lưu trữ số thực?

1.4. Lưu trữ thông tin số trong bộ nhớ



011 (3)
+ 110 (6)

1001 (9)

$9_{10} = 01001_2$
sign bit \updownarrow
 $-9_{10} = 11001_2$

100010

Câu hỏi:

1.13. Cách máy tính xử lý tính toán với số âm?

1.4. Lưu trữ thông tin số trong bộ nhớ

1's-Complement Form

1 0 1 1 0 1 original binary number

↓ ↓ ↓ ↓ ↓ ↓

0 1 0 0 1 0 complement each bit to form 1's complement

2's Complement Form

1 0 1 1 0 1 binary equivalent of 45

0 1 0 0 1 0 complement each bit to form 1's complement

+ 1 add 1 to form 2's complement

0 1 0 0 1 1 2's complement of original binary number

-9 → 10111

+9 → 01001

0 1 00000

Disregard; the result is 00000 (sum = +0).

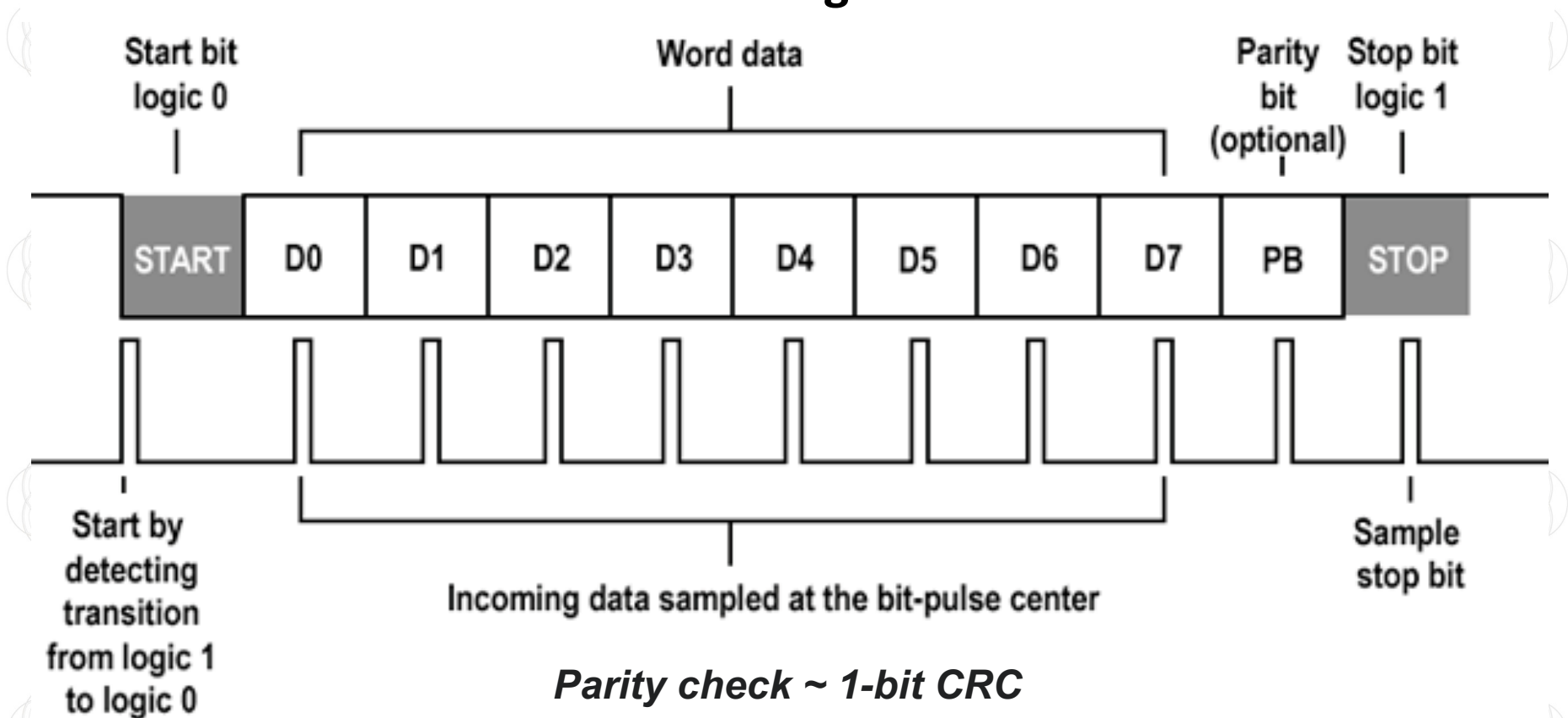
1.5. Bảng mã

Character	HEX	Decimal	Character	HEX	Decimal	Character	HEX	Decimal	Character	HEX	Decimal
NUL (null)	0	0	Space	20	32	@	40	64	.	60	96
Start Heading	1	1	!	21	33	A	41	65	a	61	97
Start Text	2	2	"	22	34	B	42	66	b	62	98
End Text	3	3	#	23	35	C	43	67	c	63	99
End Transmit.	4	4	\$	24	36	D	44	68	d	64	100
Enquiry	5	5	%	25	37	E	45	69	e	65	101
Acknowledge	6	6	&	26	38	F	46	70	f	66	102
Bell	7	7	`	27	39	G	47	71	g	67	103
Backspace	8	8	(28	40	H	48	72	h	68	104
Horiz. Tab	9	9)	29	41	I	49	73	i	69	105
Line Feed	A	10	*	2A	42	J	4A	74	j	6A	106
Vert. Tab	B	11	+	2B	43	K	4B	75	k	6B	107
Form Feed	C	12	,	2C	44	L	4C	76	l	6C	108
Carriage Return	D	13	-	2D	45	M	4D	77	m	6D	109
Shift Out	E	14	.	2E	46	N	4E	78	n	6E	110
Shift In	F	15	/	2F	47	O	4F	79	o	6F	111
Data Link Esc	10	16	0	30	48	P	50	80	p	70	112
Direct Control 1	11	17	1	31	49	Q	51	81	q	71	113
Direct Control 2	12	18	2	32	50	R	52	82	r	72	114
Direct Control 3	13	19	3	33	51	S	53	83	s	73	115
Direct Control 4	14	20	4	34	52	T	54	84	t	74	116
Negative ACK	15	21	5	35	53	U	55	85	u	75	117
Synch Idle	16	22	6	36	54	V	56	86	v	76	118
End Trans Block	17	23	7	37	55	W	57	87	w	77	119
Cancel	18	24	8	38	56	X	58	88	x	78	120
End of Medium	19	25	9	39	57	Y	59	89	y	79	121
Substitutue	1A	26	:	3A	58	Z	5A	90	z	7A	122
Escape	1B	27	;	3B	59	[5B	91	{	7B	123
Form separator	1C	28	<	3C	60	\	5C	92		7C	124
Group separator	1D	29	=	3D	61]	5D	93	}	7D	125
Record Separator	1E	30	>	3E	62	^	5E	94	~	7E	126
Unit Separator	1F	31	?	3F	63	_	5F	95	Delete	7F	127

Character	HEX	Decimal
@	40	64
A	41	65
B	42	66
C	43	67

1. How do we command to print a character on the screen?
2. Find the original document for ASCII table ?

1.5. Bảng mã

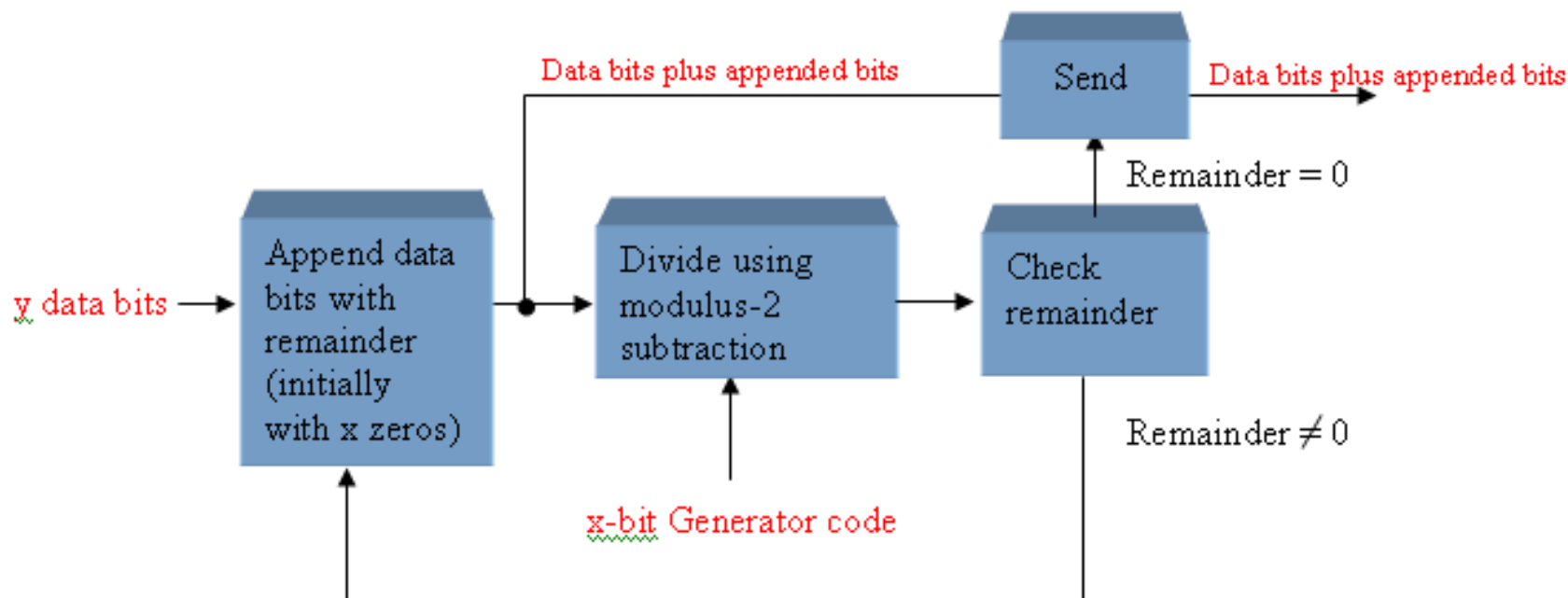


1.14. Cách máy tính truyền dữ liệu nối tiếp theo mã ASCII chuỗi “HI” với frame truyền 8 bit, không kiểm tra parity?

1.15. Kiểm tra parity là gì? Vẽ giản đồ xung chuỗi truyền “HI” có kiểm tra parity even.

1.5. Bảng mã

Cyclic Redundancy Check



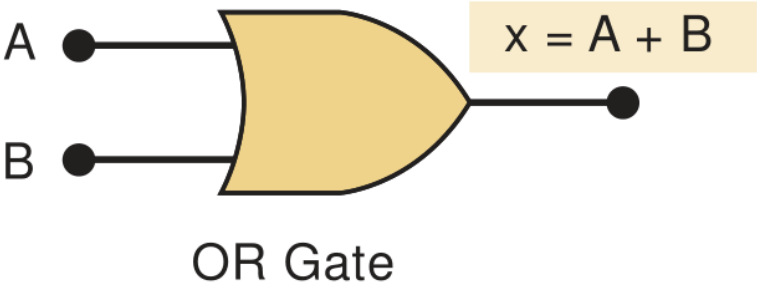
1.16 (*) Cách tính byte checksum CRC-8?

(tham khảo: [CRC Implementation Code in C and C++ | Barr Group](#))

1.6. Mạch logic

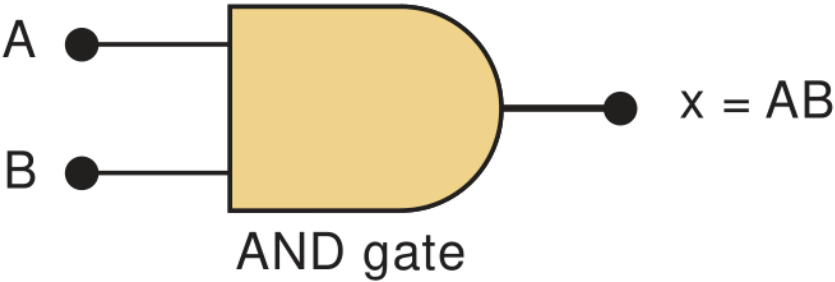
OR

A	B	$x = A + B$
0	0	0
0	1	1
1	0	1
1	1	1



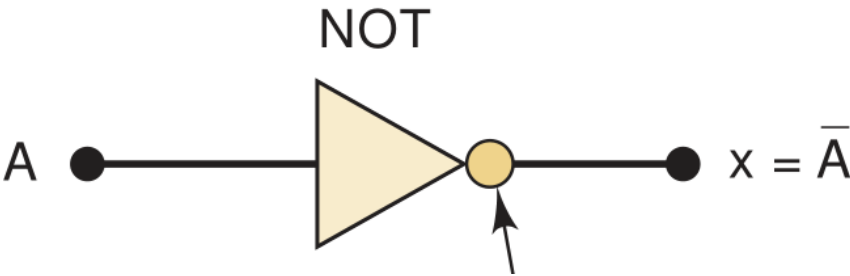
AND

A	B	$x = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

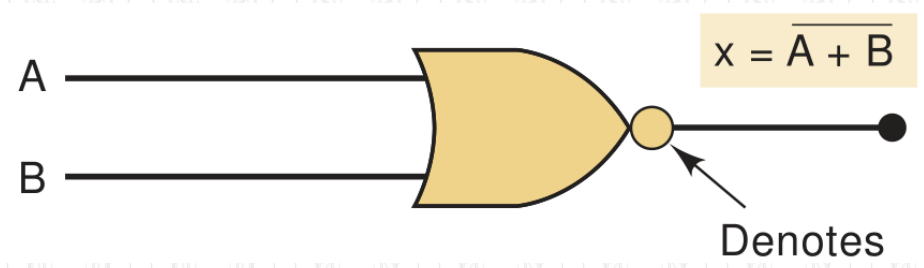


NOT

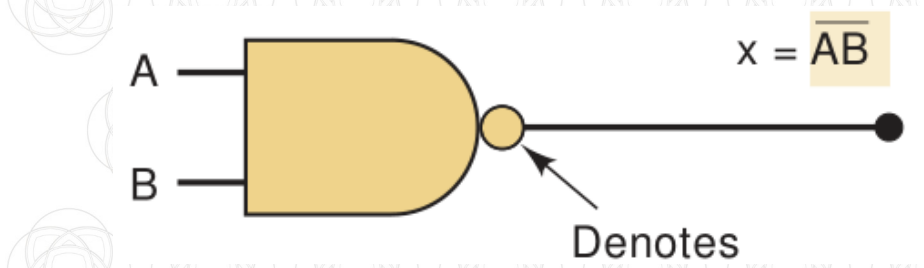
A	$x = \bar{A}$
0	1
1	0



1.6. Mạch logic



		OR		NOR	
A	B	$A + B$		$\overline{A + B}$	
0	0	0		1	
0	1	1		0	
1	0	1		0	
1	1	1		0	



		AND		NAND	
A	B	AB		\overline{AB}	
0	0	0		1	
0	1	0		1	
1	0	0		1	
1	1	1		0	

1.17. Vẽ mạch logic cho biểu thức $x = (A + B)(\overline{B} + C).$

1.6. Mạch logic

$$(16) \quad \overline{(x + y)} = \bar{x} \cdot \bar{y}$$

DEMORGAN'S THEOREMS

the OR sum of two variables is inverted, this is the same as inverting each variable individually and then ANDing these inverted variables.

$$(17) \quad \overline{(x \cdot y)} = \bar{x} + \bar{y}$$

the AND product of two variables is inverted, this is the same as inverting each variable individually and then ORing them

Bài tập

1.18 Đơn giản biểu thức:

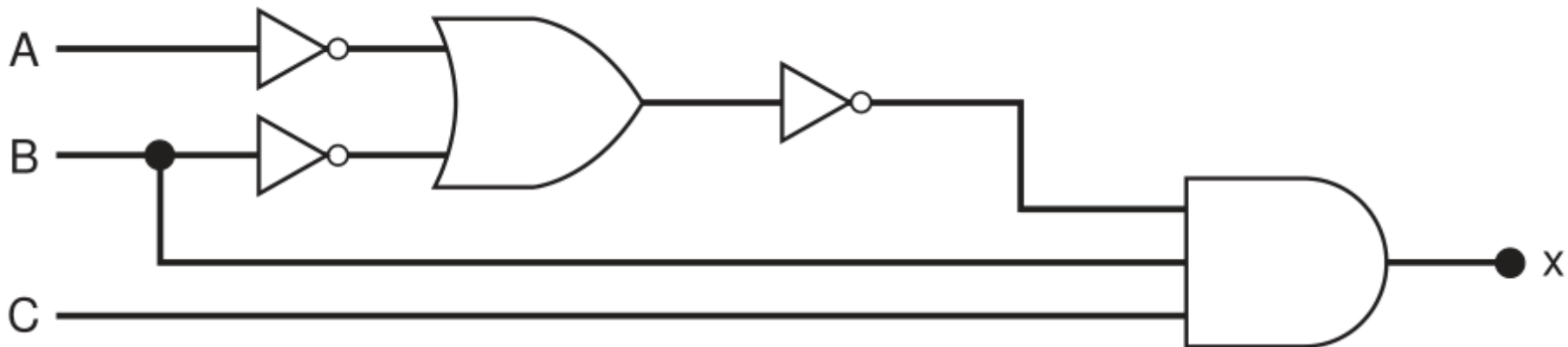
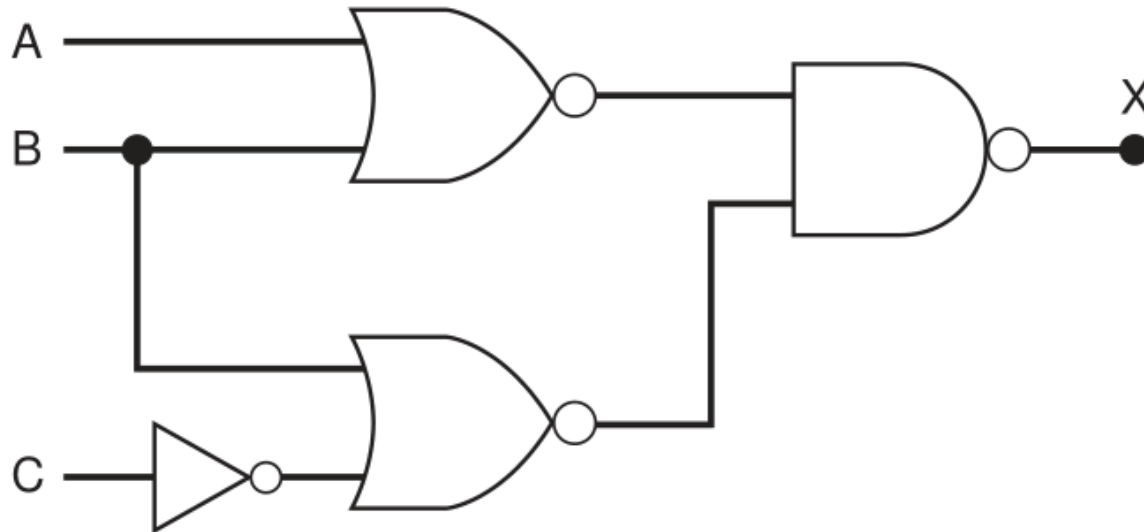
$$z = \overline{(\bar{A} + C) \cdot (B + \bar{D})}$$

1.19. Đổi sang biểu thức chỉ sử dụng cổng NOR (hoặc NAND)

$$x = \overline{AB \cdot (\bar{C} + \bar{D})}$$

1.6. Mạch logic

1.20. Use DeMorgan's theorems to simplify the expression for the output of Figure. Then rewrite it in only NOR (or only NAND) form



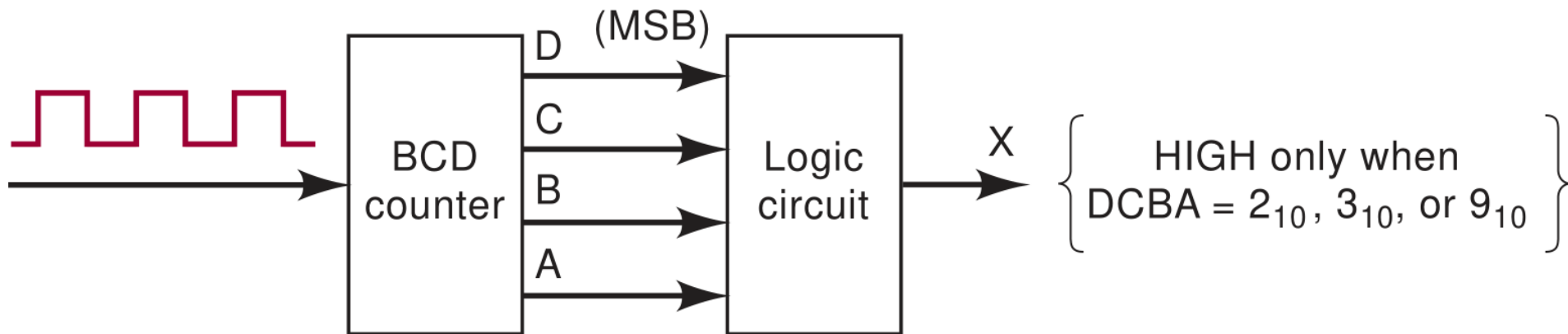
1.6. Mạch logic

KARNAUGH MAP METHOD:

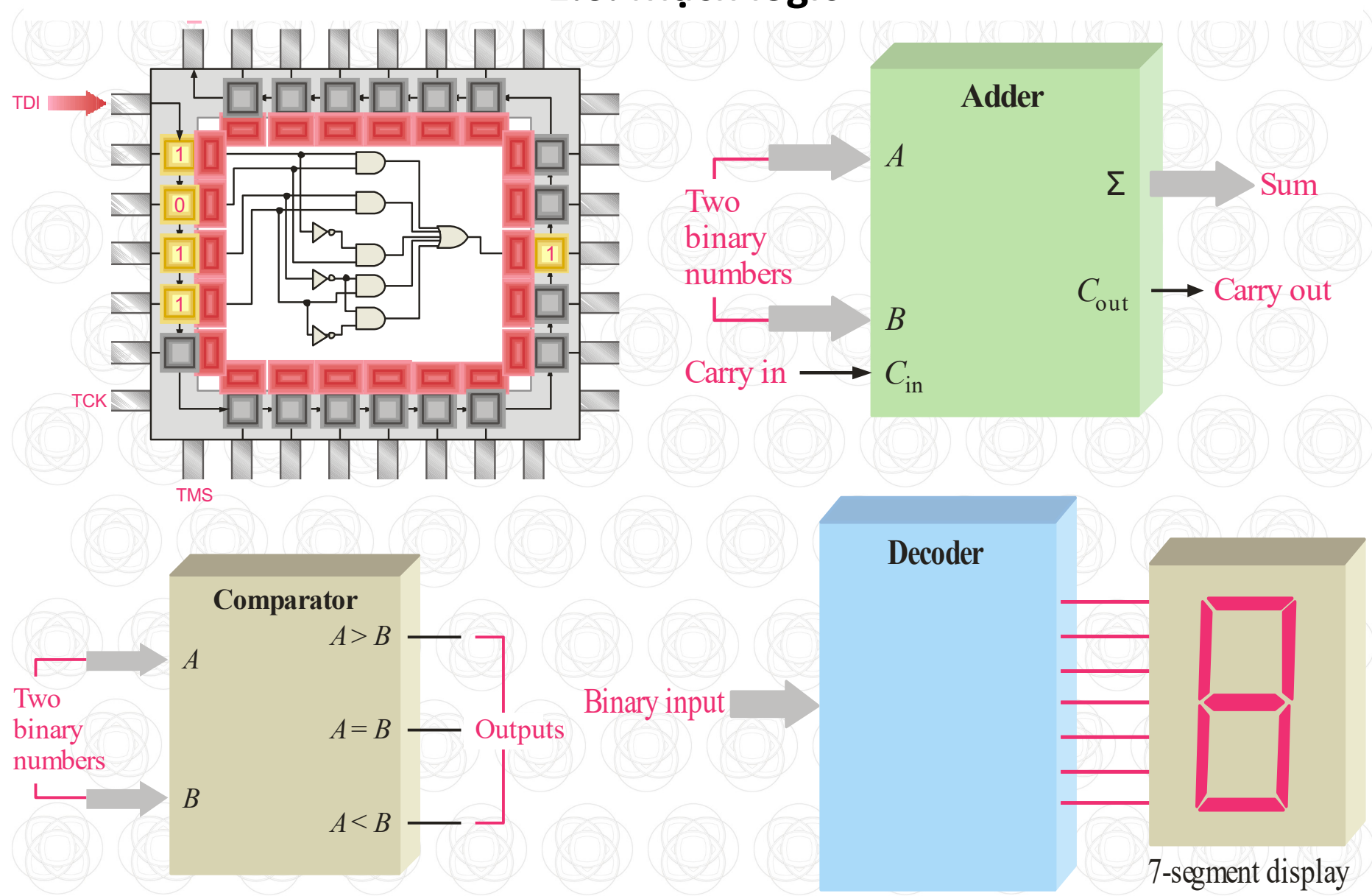
Câu hỏi

1.21. Sử dụng phương pháp bìa Karnaugh để thiết kế mạch theo yêu cầu sau đây:

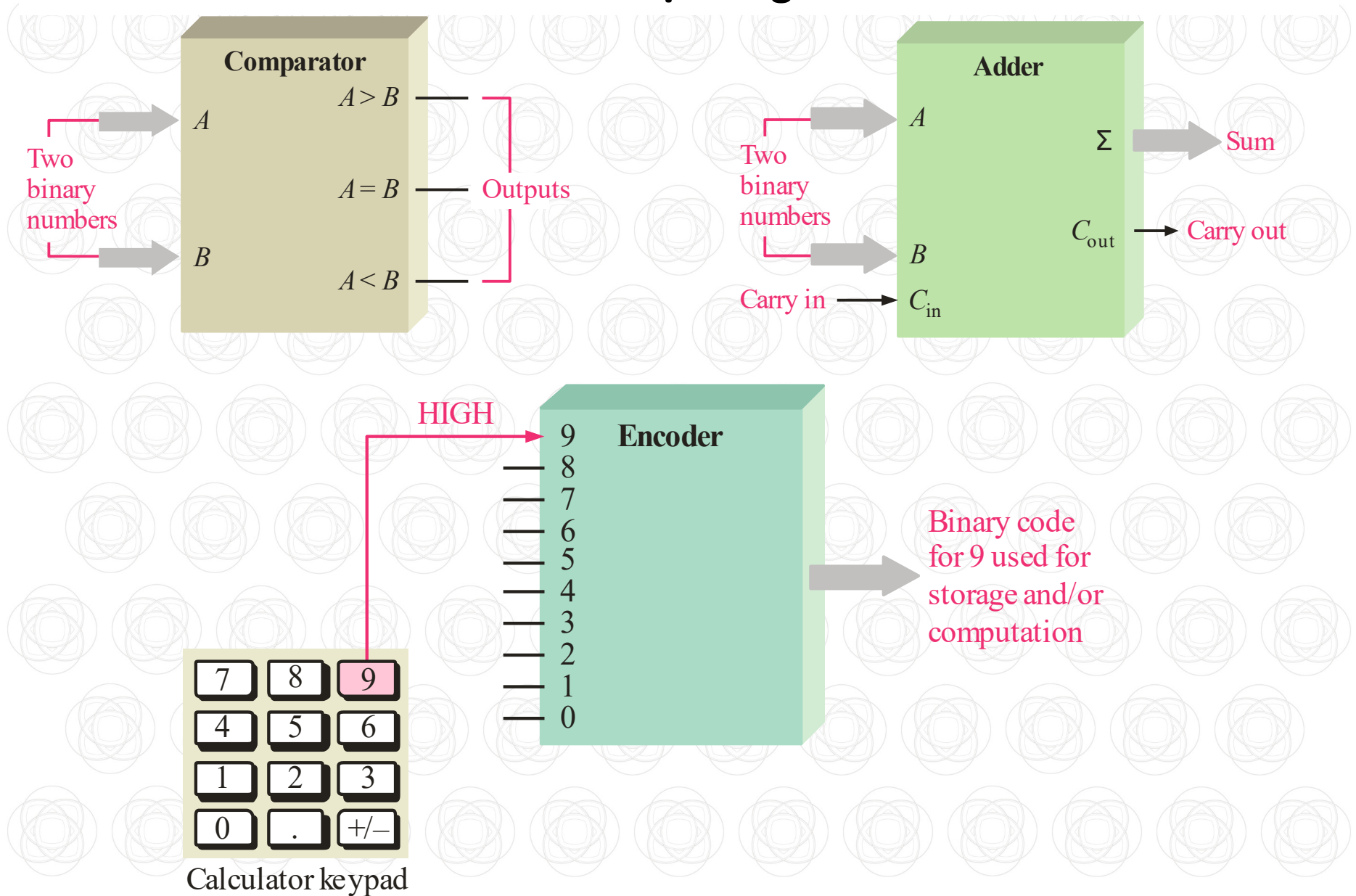
- Ngõ ra X xuất tín hiệu mức cao (1) khi mạch đếm BCD (với D là MSB) có giá trị 2, 3, 9.



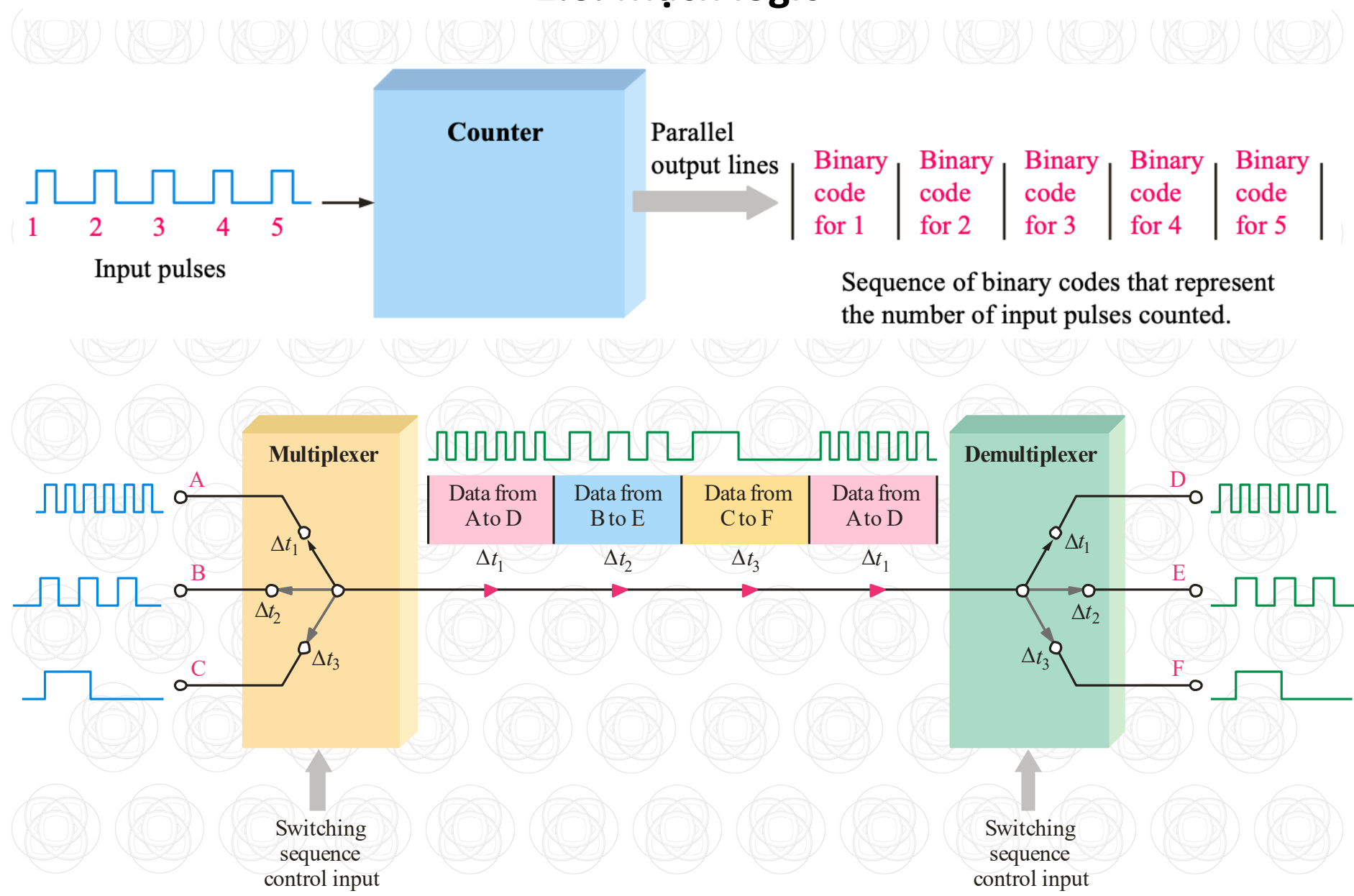
1.6. Mạch logic



1.6. Mạch logic



1.6. Mạch logic



1.6. Mạch logic

FF triggers on positive transition

Inputs			Output
S	R	CLK	Q
0	0	↑	Q_0 (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	Ambiguous

FF triggers on positive transition

J	K	CLK	Q
0	0	↑	Q_0 (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	$\overline{Q_0}$ (toggles)

FF triggers on positive transition

D	CLK	Q
0	↑	0
1	↑	1

1.6. Mạch logic

1.22. Trong mạch điện bên dưới, tín hiệu OPEN LOCK sẽ xuất mức 1 khi nào?

