

CHƯƠNG 0: ÔN TẬP KỸ THUẬT SỐ

0.1 Hệ thống số đếm-Mã số

Biểu diễn tổng quát số đếm theo đa thức cơ số:

$$(N)_a = b_{n-1}a^{n-1} + b_{n-2}a^{n-2} + \dots + b_2a^2 + b_1a^1 + b_0$$

$(N)_a$: giá trị số đếm trong hệ cơ số a

a : cơ số

$b_i (i = 0, \dots, k-1)$: các ký tự biểu diễn giá trị trong hệ cơ số a

k : số ký tự biểu diễn giá trị $(N)_a$

0.1.1 Hệ thập phân(decimal)

Cơ số=10, biểu diễn 10 ký tự giá trị 0,1,2,3,4,5,6,7,8,9

Ví dụ 0.1: Biểu diễn số $(357809)_{10}$ dưới dạng đa thức cơ số

Giải:

$$(357809)_{10} = 3 \times 10^5 + 5 \times 10^4 + 7 \times 10^3 + 8 \times 10^2 + 0 \times 10^1 + 9$$

0.1.2 Hệ nhị phân(binary)

Cơ số=2, biểu diễn 2 ký tự giá trị 0,1

Ví dụ 0.2: Biểu diễn số $(1011)_2$ dưới dạng đa thức cơ số

Giải:

$$(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 = (11)_{10}$$

□ Các định nghĩa trong hệ nhị phân

- Bit: chỉ các ký tự giá trị 0 hoặc 1 → bit 0, bit 1
- MSB(Most Significant Bit): bit có trọng số cao nhất
- LSB(Least Significant Bit): bit có trọng số thấp nhất
- Nibble: 1 nibble=4 bit
- Byte: 1 byte=8 bit
- 1Kbyte=1KB=2¹⁰ Byte=1024 Byte
- 1Mbyte=1MB=2¹⁰ KB
- 1 word=2 byte=16 bit
- Double word=2 word=4 byte=32 bit

0.1.3 Hệ thập lục phân(hexadecimal)-hệ Hex

Cơ số=16, biểu diễn 16 ký tự giá trị 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F.

Số hex được sử dụng để biểu diễn rút gọn số nhị phân nhiều bit.

Quy ước biểu diễn hệ cơ số hex có chữ H đứng cuối.

Bảng 0.1: Biểu diễn số nhị phân-thập phân-hex

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

Ví dụ 0.3: Biểu diễn số nhị phân 01101011 sang số hex

Giải:

$$01101011 = (6D)_{16} = 6DH$$

0.1.4 Số BCD(Binary Code Decimal)

Số BCD là số thập phân được biểu diễn dưới dạng nhị phân

Bảng 0.2: Biểu diễn số BCD

Decimal	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Ví dụ 0.4: Biểu diễn số $(37)_{10}$ dưới dạng số BCD

Giải:

$$(37)_{10} = 0011 \ 0111$$

0.1.5 Chuyển đổi cơ số

- Chuyển từ nhị phân sang thập phân

Viết số nhị phân dưới dạng đa thức cơ số 2 và cộng kết quả đa thức.

Ví dụ 0.5: Chuyển số nhị phân 01011101 sang thập phân.

Giải:

$$01011101 = 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2 + 1 = (93)_{10}$$

- Chuyển từ thập phân sang nhị phân

Thực hiện liên tiếp các phép chia số thập phân và kết quả thương số sau đó cho 2, cho đến khi thương số bằng 0. Kết quả các dư số sau mỗi phép chia chính là các bit nhị phân, lần lượt dư số đầu tiên là LSB và dư số cuối cùng là MSB.

Ví dụ 0.6: chuyển số $(37)_{10}$ sang nhị phân

Giải:

□	Số bị chia	Số chia	Thương số	Dư số
	37	2	18	1
	18	2	9	0
	9	2	4	1
	4	2	2	0
	2	2	1	0
	1	2	0	1
	Kết quả: $(37)_{10} = 100101$			

- Chuyển từ hex sang thập phân

- Chuyển từ hex sang nhị phân sau đó chuyển từ nhị phân sang thập phân

- Chuyển trực tiếp từ hex sang thập phân qua biểu diễn đa thức cơ số, lưu ý chuyển ký tự từ A đến F sang 10 đến 15 tương ứng.

Ví dụ 0.7: Chuyển số hex $(3A4B)_{16}$ sang thập phân.

Giải:

$$(3A4B)_{16} = 3 \times 16^3 + 10 \times 16^2 + 4 \times 16^1 + 11 = (14923)_{10}$$

- Chuyển từ thập phân sang hex

- Chuyển từ thập phân sang nhị phân sau đó chuyển từ nhị phân sang hex

- Thực hiện tương tự như chuyển từ thập phân sang nhị phân với phép chia cho 16, lưu ý chuyển dư số từ 10 đến 15 sang ký tự A đến F tương ứng

Ví dụ 0.8: Chuyển số thập phân $(14923)_{10}$ sang số hex.

Giải:

Số bị chia	Số chia	Thương số	Dư số	Dư số(hex)
14923	16	932	11	B
932	16	58	4	4
58	16	3	10	A
3	16	0	3	3

Kết quả: $(14923)_{10} = (3A4B)_{16}$

0.1.6 Cộng trừ số nhị phân, hex

Ví dụ 0.9: Cộng 2 số nhị phân 8 bit

Nhị phân	Hex	TP
01111001	79	121
<u>10100111</u>	<u>A7</u>	<u>167</u>
100100000	1 20	288
↑ bit nhớ	↑ bit nhớ	

Phép cộng trên có kết quả bị tràn khỏi giá trị 8 bit(256) nên sử dụng thêm bit nhớ(bit carry)

Ví dụ 0.10: Trừ 2 số nhị phân 8 bit

Nhi phân	Hex	TP
01111001	79	121
<u>10100111</u>	<u>A7</u>	<u>167</u>
111010010	1 D2	-46
↑ bit nhớ	↑ bit nhớ	

Phép trừ trên có kết quả âm, tràn khỏi giá trị 8 bit nên sử dụng thêm bit nhớ (bit carry)

0.1.7: Số bù-1, số bù-2, số nhị phân có dấu

- Số bù-1 của số nhị phân chính là số đảo các bit của số nhị phân đó.
- Số bù-2 của số nhị phân chính là số bù-1 của số nhị phân đó cộng thêm 1
Số bù-2 thường được sử dụng biểu diễn số nhị phân có dấu
- Số nhị phân có dấu: quy ước MSB=0 là số dương, MSB=1 là số âm.

Biểu diễn số nhị phân 8 bit có dấu:

Bit 7=bit dấu

Từ 00000000 đến 01111111 (00H đến 7FH) = 0 đến +127 thập phân

Từ 10000000 đến 11111111 (80H đến FFH) = -128 đến -1 thập phân

Ví dụ 0.11: Biểu diễn số 00110111 dưới dạng số bù-2 của nó viết theo số nhị phân, hex

Giải:

Bù-1 00110111 → 11001000

Bù-2 11001000 + 1 = 11001001 = C9H

□ Thực chất máy tính chỉ làm phép toán cộng số nhị phân, nên ta quy ước biểu diễn số nhị phân dưới dạng bù-2, máy tính sẽ làm được các phép toán cộng số có dấu và chuyển phép trừ thành phép cộng.

Máy tính có sử dụng thêm các cờ báo trong phép toán cộng:

- Cờ carry C : báo kết quả phép tính tràn qua khỏi bit MSB
- Cờ over flow OV: báo kết quả phép tính tràn qua bit dấu (bit MSB)

Ví dụ 0.12: Xem các phép cộng 2 số bù-2 8 bit sau và lưu ý các cờ C và OV sau mỗi phép tính:

- Cờ C=1 khi phép cộng tràn qua khỏi bit 7
- Cờ OV=1 khi phép cộng tràn từ bit 6 qua bit 7 là bit dấu

00101001	29H	41
<u>00011101</u>	<u>1DH</u>	<u>29</u>
01000110	46H	70
C=0, OV=0, bit dấu=0 → kq +		

01111001	79H	121
<u>01100010</u>	<u>62H</u>	<u>+ 98</u>
11011011	DBH	219
C=0, OV=1, bit dấu=1 → kq +		

00101001	29H	41
<u>11100011</u>	<u>E3H</u>	<u>- 29</u>
1 00001100	1 0CH	12
C=1, OV=1, bit dấu=0 → kq +		

01100010	62H	98
<u>10000111</u>	<u>87H</u>	<u>- 121</u>
11101001	E9H	- 23
C=0, OV=0, bit dấu=1 → kq -		

10000111	87H	(-121)
<u>10011110</u>	<u>9EH</u>	<u>+ (- 98)</u>
1 00100101	1 25H	- 219
C=1, OV=0, bit dấu=0 → kq -		

11111110	FEH	(-2)
<u>11111111</u>	<u>FFH</u>	<u>+ (-1)</u>
1 11111101	1 FDH	- 3
C=1, OV=1, bit dấu=1 → kq -		

□ Từ tổ hợp cờ C, OV và bit dấu của các kết quả phép tính, máy tính sẽ hiệu đính kết quả theo bảng B0.3

Bảng B0.3: Kết quả hiệu đính phép cộng

C	OV	Bit dấu	Kết quả
0	0	0	+
0	0	1	-
0	1	1	+
1	0	0	-
1	1	0	+
1	1	1	-

0.1.8 Nhân số nhị phân

□ Phép nhân hai số nhị phân cũng tương tự cách nhân hai số thập phân

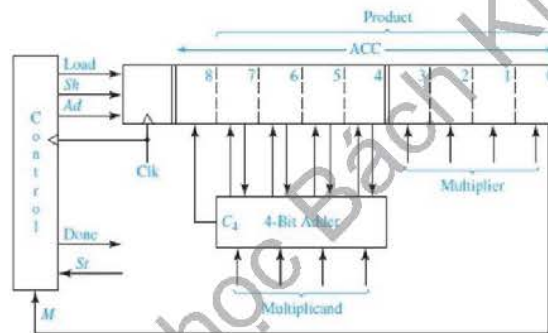
Ví dụ 0.13: Nhân 2 số nhị phân 1101 X 1011

Giải:

$$\begin{array}{r}
 \text{Số bị nhân:} \quad 1101 \quad (13) \\
 \text{Số nhân:} \quad 1011 \quad (11) \\
 \hline
 \text{Tích tạm thời} \rightarrow \begin{array}{r} 1101 \\ 1101 \\ 0000 \\ 1101 \end{array} \\
 \hline
 \text{Tích số:} \quad 1000111 \quad (143)
 \end{array}$$

Phép nhân nhị phân thực chất là thực hiện liên tiếp các phép cộng và dịch trái số bị nhân(multiplicand)tùy thuộc vào bit 1 hay 0 của số nhân(multiplier)trương ứng.

Thay vì dịch trái số bị nhân,ta dịch phải số nhân,nếu gặp bit 1 cộng số bị nhân với kết quả cộng tích lũy hay tích tạm thời(partial products).



Hình 0.1: Sơ đồ khối mạch nhân 4 bit

Hình 0.1 là sơ đồ khối mạch nhân 4 bit gồm bộ cộng FA 4 bit, thanh ghi 4 bit chứa số bị nhân, thanh ghi 4 bit chứa số nhân, thanh ghi kết quả tích lũy ACC 9 bit bao gồm luôn thanh ghi chứa số nhân. Bộ điều khiển gồm các tín hiệu điều khiển vào ra và xung CLK đồng bộ hoạt động.

- Tín hiệu Load nạp số nhân vào 4 bit thấp và xóa 5 bit cao thanh ghi ACC
- Tín hiệu St cho phép bộ điều khiển xuất tín hiệu Ad và Sh để thực hiện phép cộng và dịch phải ACC
- Nếu M=1 bộ điều khiển xuất tín hiệu Ad cho phép bộ FA 4 bit cộng số bị nhân vào 4 bit cao ACC với bit 8=C, sau đó xuất tín hiệu Sh dịch phải ACC
- Nếu M=0 bộ điều khiển không xuất tín hiệu Ad, không cộng chỉ dịch phải ACC
- Sau 4 xung CLK bộ điều khiển xuất tín hiệu Done báo hoàn thành phép nhân

□ Quy trình thực hiện các bước phép nhân theo sơ đồ khối hình 0.1 như sau:

Khởi động thanh ghi tích số	0 0 0 0 0 1 0 1 1 ←M	(11)
M=1 cộng số bị nhân	1 1 0 1	(13)
Sau khi cộng	0 1 1 0 1 1 0 1 1	
Sau khi dịch	0 0 1 1 0 1 1 0 1 ←M	
M=1 cộng số bị nhân	1 1 0 1	
Sau khi cộng	1 0 0 1 1 1 1 0 1	
Sau khi dịch	0 1 0 0 1 1 1 1 0 ←M	
M=0 bỏ qua cộng		
Sau khi dịch	0 0 1 0 0 1 1 1 1 ←M	
M=1 cộng số bị nhân	1 1 0 1	
Sau khi cộng	1 0 0 0 1 1 1 1 1	
Sau khi dịch	0 1 0 0 0 1 1 1 1	(143)

Đường phân biệt giữa tích số và số nhân

0.1.9 Chia số nhị phân

□ Phép chia 2 số nhị phân cũng tương tự phép chia 2 số thập phân

Ví dụ 0.14: Chia 2 số nhị phân 10000111 cho 1101

Giải:

$$\begin{array}{r}
 \text{Số chia } 1101 \overline{) 10000111} \\
 \underline{1101} \\
 0111 \\
 \underline{0000} \\
 1111 \\
 \underline{1101} \\
 0101 \\
 \underline{0000} \\
 0101
 \end{array}$$

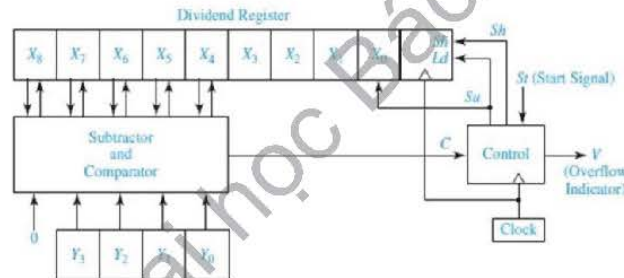
(135 ÷ 13 = 10 dư số 5)

Thương số
Số bị chia
Dư số

Phép chia số bị nhân thực chất là thực hiện liên tiếp các phép trừ và dịch phải số chia (divisor).

Thay vì dịch phải số chia, ta thực hiện phép trừ và dịch trái số bị chia nếu nó lớn hơn hoặc bằng số chia và điền thương số (quotient) bit 1, và không trừ chỉ dịch nếu số bị chia nhỏ hơn thương số, điền thương số bit 0

Sơ đồ khối mạch chia ví dụ 0.14 như hình 0.2, gồm thanh ghi số bị chia 9 bit bao gồm luôn thương số sau mỗi lần dịch trái, cần thêm bit thứ 9 để số bị chia không bị mất khi dịch trái, thanh ghi 4 bit chứa số chia, bộ so sánh và trừ 4 bit, bộ điều khiển gồm các tín hiệu CLK, Sh, Su, ...



Hình 0.2: Sơ đồ khối mạch nhân 4 bit

Các bước thực hiện phép chia ví dụ 0.14 theo sơ đồ hình 0.2 như sau:

1. Nạp số bị chia và số chia vào đúng vị trí

Số bị chia 0 1 0 0 0 1 1 1

Số chia 1 1 0 1

2. Số bị chia nhỏ hơn số chia, không trừ, dịch trái số chia, bit tận cùng bên phải bằng 0

↓ đường phân ranh số bị chia và thương số

Số bị chia 1 0 0 0 1 1 1 | 0

Số chia 1 1 0 1

3. Số bị chia lớn hơn số chia, thực hiện phép trừ, ghi thương số đầu tiên bit 1

Số bị chia 0 0 0 1 1 1 1 | 1

4. Dịch trái số bị chia

Số bị chia 0 0 1 1 1 1 1 | 1 0

Số chia 1 1 0 1

5. Số bị chia nhỏ hơn số chia, không trừ, ghi thương số thứ 2 bit 0, dịch trái số bị chia

Số bị chia 0 1 1 1 1 1 | 1 0 0

Số chia 1 1 0 1

6. Số bị chia lớn hơn số chia, thực hiện phép trừ, ghi thương số thứ 3 bit 1

Số bị chia 0 0 0 1 0 1 | 1 0 1

7. Dịch trái số bị chia lần cuối

Số bị chia 0 0 1 0 1 | 1 0 1 0

Số chia 1 1 0 1

8. Số bị chia nhỏ hơn số chia, không trừ, ghi thương số bằng 0

Kết quả 0 0 1 0 1 | 1 0 1 0

Dư số | Thương số

0.1.10 Chuyển đổi số hex/BCD trên các phép toán nhị phân

Trong các hệ vi xử lý thường hay thực hiện phép chuyển đổi Hex/BCD để tính toán/hiển thị dựa trên các lệnh thực hiện các phép tính nhị phân.

Chuyển đổi Hex sang BCD:

- Thực hiện liên tiếp các phép chia số hex và kết quả thương số sau đó cho 0AH cho đến khi thương số bằng 0.
- Số BCD tương ứng lần lượt là các dư số theo thứ tự dư số đầu là LSD và dư số cuối là MSD

Ví dụ 0.15: Chuyển số hex FAH sang BCD

Giải:

Số bị chia	Số chia	Thương số	Dư số
FA	0A	19	0
19	0A	2	5
2	0A	0	2
Kết quả FAH=(250) _{BCD}			

Chuyển đổi BCD sang Hex:

Khai triển theo đa thức, lấy tiêu biểu 4 chữ số:

$$(b_3b_2b_1b_0)_{BCD} = b_3 \times A^3 + b_2 \times A^2 + b_1 \times A^1 + b_0 = (b_3 \times 3E8 + b_2 \times 64 + b_1 \times A + b_0)_{16}$$

Ví dụ 0.16:

$$(2367)_{BCD} = (2 \times 3E8 + 3 \times 64 + 6 \times A + 7)_{16} = 93FH$$

0.1.11 Mã ASCII

ASCII(American Standard Code for Information Interchange) là bộ mã các ký tự được biểu diễn bằng mã nhị phân 7 bit, thường được sử dụng trong truyền dữ liệu.

- Mã ASCII các số thập phân từ 0 đến 9 là: 30H đến 39H
- Mã ASCII các ký tự chữ cái in hoa từ A đến Z là: 41H đến 5AH
- Mã ASCII các ký tự chữ cái thường từ a đến z là: 61H đến 7AH

Bảng 0.3: Bảng mã ASCII

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

0.1.12 Mã lái LED 7 đoạn

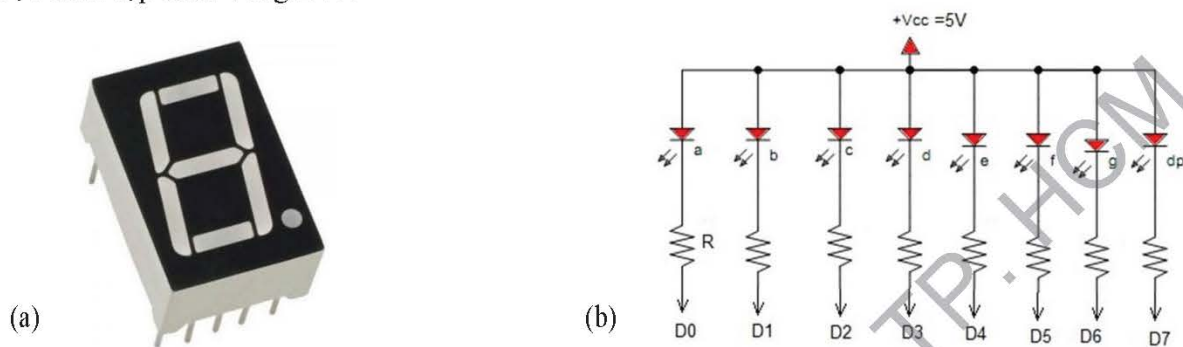
- Là mã nhị phân 8 bit lái LED 7 đoạn Anode chung hay Cathode chung.
- Hình 0.3 là LED 7 đoạn Anode chung(AC) và sơ đồ cấu trúc.

Anode nối với nguồn +5V, các Cathode(K) ứng với các chân đoạn ký hiệu từ a đến g và dp nối với R để hạn

dòng. Muốn đoạn nào sáng, chân đoạn tương ứng phải nối xuống mức 0 qua R.

Phân cực LED sáng, $V_D = 2V$, $I_D = 10mA \rightarrow R = (5-2)/10mA = 300\Omega$

□ Kết nối các chân đoạn từ a đến g, dp với 8 bit data ký hiệu D0 đến D7 tương ứng, mã lái LED 7 đoạn AC theo hình 0.3b được thiết lập theo bảng B0.4



Hình 0.3: Hình dạng và cấu trúc mạch LED 7 đoạn AC

Bảng B0.4: Bảng mã 7 đoạn lái LED AC theo hình 0.3b

Số	Số nhị phân								HEX
	7	6	5	4	3	2	1	0	
	dp	g	f	e	d	c	b	a	
0	1	1	0	0	0	0	0	0	C0
1	1	1	1	1	1	0	0	1	F9
2	1	0	1	0	0	1	0	0	A4
3	1	0	1	1	0	0	0	0	B0
4	1	0	0	1	1	0	0	1	99
5	1	0	0	1	0	0	1	0	92
6	1	0	0	0	0	0	1	0	82
7	1	1	1	1	1	0	0	0	8F
8	1	0	0	0	0	0	0	0	80
9	1	0	0	1	0	0	0	0	90
A	1	0	0	0	1	0	0	0	88
B	1	0	0	0	0	0	1	1	83
C	1	1	0	0	0	1	1	0	C6
D	1	0	1	0	0	0	0	1	A1
E	1	0	0	0	0	1	1	0	86
F	1	0	0	0	1	1	1	0	8E

□ **Ghi chú:** ký tự B,D hiển thị chữ thường lần lượt là b,d

0.2 Các mạch logic

0.2.1 Cổng 3 trạng thái

Hình 0.4 biểu diễn ký hiệu và bảng sự thật hoạt động của cổng 3 trạng thái.

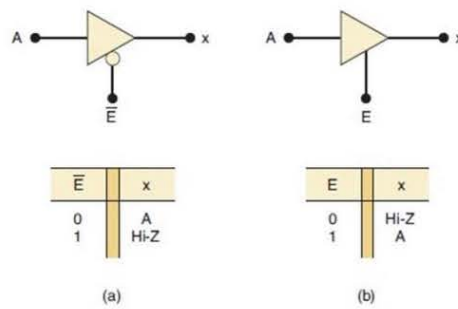
□ Cấu hình chung: ngõ vào=A, ngõ ra=x và ngõ điều khiển=E

- Ngõ điều khiển E tích cực mức 0 hình (a) hay mức 1 hình (b)

- Ngõ ra x=A khi E tích cực và x=hi-Z=hở mạch=trạng thái thứ 3 khi E không tích cực

□ Sử dụng trong các mạch giao tiếp vào/ra hệ vi xử lý thường kết nối chung bus data, tránh hiện tượng wired AND.

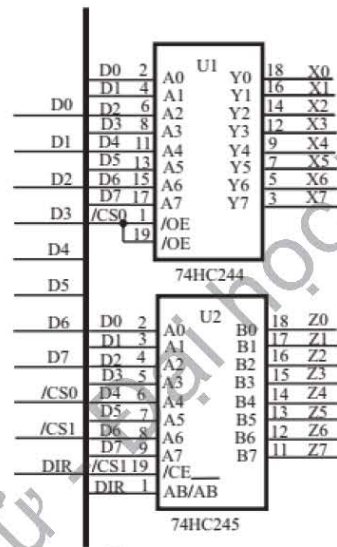
□ Các IC thường sử dụng 74HC240, 74HC244, 74HC245 (xem data sheet)



Hình 0.4: Ký hiệu và bảng sự thật cổng 3 trạng thái

Ví dụ 0.17: Hình 0.5 kết nối 2 thiết bị ngoại vi chung bus data với hệ vi xử lý:

- D0 – D7: bus data của hệ vi xử lý truyền 2 chiều
- X0 – X7: kết nối data từ ngõ ra ngoại vi 1
- Z0 – Z7: kết nối data từ ngõ vào/ra ngoại vi 2
- Các tín hiệu điều khiển: /CS0=0 chọn ngoại vi 1, /CS1=0, chọn ngoại vi 2, DIR chọn chiều vào/ra ngoại vi 2. Thiết lập bảng trạng thái hoạt động kết nối bus data giữa hệ vi xử lý và các ngoại vi trên.



Hình 0.5: Sơ đồ ví dụ 0.17

Giải:

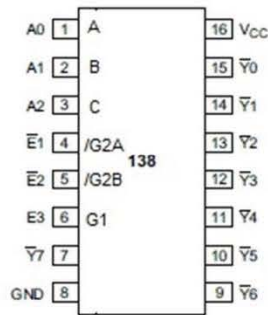
Theo sơ đồ hình 0.5 và data sheet IC 74HC244 và 74HC245, ta có bảng trạng thái sau:

/CS0	/CS1	DIR	NV1	I/O NV2
0	1	X	OUT	Hi-Z
1	0	1	Hi-Z	OUT
1	0	0	Hi-Z	IN

- Các ngoại vi nhìn từ hệ vi xử lý (bus data D0 – D7)

0.2.2 Mạch giải mã/phân kênh (Decoder/Demultiplexer)

- Thường sử dụng trong mạch giải mã địa chỉ chọn chip bộ nhớ/ngoại vi trong hệ vi xử lý
- Khảo sát IC 74HC138: sơ đồ chân như hình 0.6, bảng sự thật như bảng B0.5
 - Ngõ vào giải mã: A2A1A0=CBA
 - Ngõ vào điều khiển: /E1, /E2, E3=/G2A, /G2B, G1 cho phép/cấm giải mã
 - /Y0 - /Y7: ngõ ra giải mã tích cực mức 0



Hình 0.6: Sơ đồ chân IC 74HC138

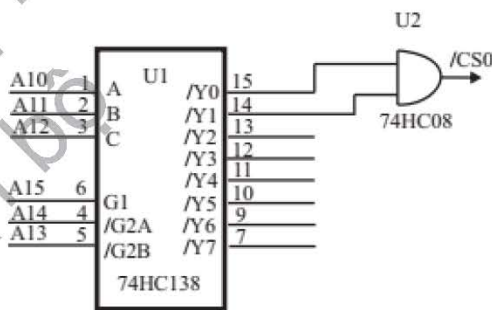
Bảng B0.5: Hoạt động IC 74HC138

Control			Input			Output							
E1	E2	E3	A2	A1	A0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
H	X	X	X	X	X	H	H	H	H	H	H	H	H
X	H	X											
X	X	L											
L	L	H	L	L	L	H	H	H	H	H	H	H	L
			L	L	H	H	H	H	H	H	H	L	H
			L	H	L	H	H	H	H	H	L	H	H
			L	H	H	H	H	H	L	H	H	H	H
			H	L	L	H	H	H	L	H	H	H	H
			H	L	H	H	H	L	H	H	H	H	H
			H	H	L	H	L	H	H	H	H	H	H
			H	H	H	L	H	H	H	H	H	H	H

[1] H = HIGH voltage level;
L = LOW voltage level;
X = don't care.

Ví dụ 0.18: Hình 0.7 là sơ đồ mạch giải mã địa chỉ tạo tín hiệu chọn chip /CS0 tích cực mức 0. Viết biểu thức Boolean ngõ ra /CS0 theo các ngõ vào A10 – A15.

Giải:



Hình 0.7

Từ sơ đồ hình 0.7 và bảng sự thật IC 74HC138 ta thiết lập bảng trạng thái /CS0 như sau:

A15	A14	A13	A12	A11	A10	Y0	Y1	/CS0
1	0	0	0	0	0	0	1	0
1	0	0	0	0	1	1	0	0

Từ bảng trên suy ra biểu thức Boolean:

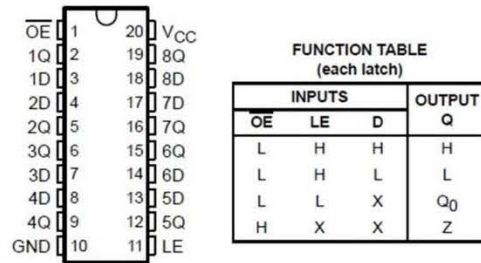
$$\overline{CS_0} = \overline{A_{15}}\overline{A_{14}}\overline{A_{13}}\overline{A_{12}}\overline{A_{11}}\overline{A_{10}} + \overline{A_{15}}\overline{A_{14}}\overline{A_{13}}\overline{A_{12}}\overline{A_{11}}A_{10}$$

$$\overline{CS_0} = \overline{A_{15}}\overline{A_{14}}\overline{A_{13}}\overline{A_{12}}\overline{A_{11}}$$

0.2.3 Mạch chốt D

- Thường được sử dụng chốt data trong các mạch tạo địa chỉ hoặc giao tiếp ngoại vi.

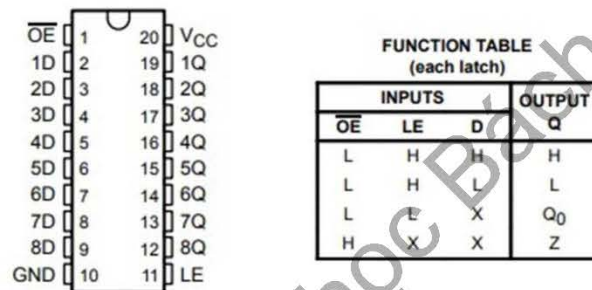
- IC 74HC373 sử dụng trong mạch chốt địa chỉ byte thấp từ ngõ ra MUX địa chỉ/data hệ vi xử lý.
 - Chốt D 8 bit ngõ ra 3 trạng thái
 - Ngõ vào cho phép xuất: /OE=0 :cho phép xuất , /OE=1: Qi=hi-Z
 - Ngõ vào điều khiển chốt:LE=1: mở chốt Qi=Di LE=0, khóa chốt Qi=Qio



Hình 0.8: Sơ đồ chân và bảng sự thật IC 74HC373

IC 74HC573 thường được sử dụng trong các mạch giao tiếp ngoại vi.

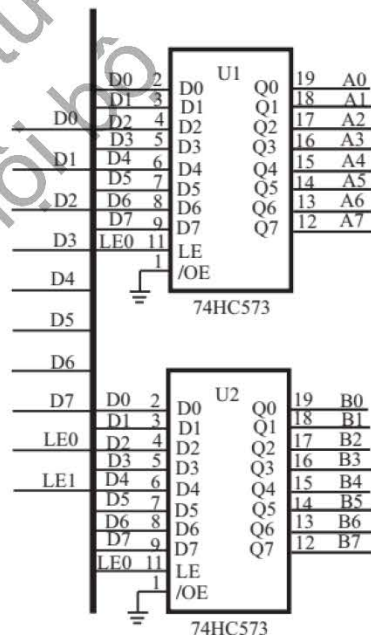
- Hoạt động tương tự như IC 74HC573, dòng điện ngõ ra cao $I_o = \pm 35\text{mA}$.



Hình 0.9: Sơ đồ chân và bảng sự thật IC74HC573

Ví dụ 0.19: Hình 0.10 là sơ đồ kết nối bus data D0 – D7 của hệ vi xử lý với 2 thiết bị ngoại vi qua 2 mạch chốt D 8 bit U5 và U6 74HC573. Giả sử hệ vi xử lý muốn truyền 2 byte data C0H và 3FH lần lượt ra U5 và U6, thiết lập bảng trạng thái logic minh họa cách truyền.

Giải:



Hình 0.10

Từ sơ đồ hình 0.10 và bảng sự thật IC74HC573, ta có bảng trạng thái logic như sau: $i=0 - 7$

Inputs										Outputs	
D7	D6	D5	D4	D3	D2	D1	D0	LE0	LE1	U5	U6
X	X	X	X	X	X	X	X	0	0	A _{i0}	B _{i0}
1	1	0	0	0	0	0	0	1	0	A _{i=D} i	B _{i0}
X	X	X	X	X	X	X	X	0	0	C0H	B _{i0}
0	0	1	1	1	1	1	1	0	1	C0H	B _{i=D_i}
X	X	X	X	X	X	X	X	0	0	C0H	3FH

A_{i0}, B_{i0}: trạng thái ổn định trước đó của ngõ ra U5, U6

0.3 Bộ nhớ(Memory)

- Các hệ vi xử lý thường sử dụng bộ nhớ ROM, RAM tĩnh giao tiếp song song dung lượng hàng MB
- Cấu trúc chung sơ đồ chân IC bộ nhớ song song:
 - Bus địa chỉ ký hiệu A_i tùy thuộc vào dung lượng bộ nhớ.
Ví dụ bộ nhớ 8KB=2¹³ byte có 13 đường địa chỉ ký hiệu A0 – A12
 - Bus data ký hiệu D_i, thường độ dài chuẩn là 8 bit: D0 – D7
- Ngõ data giao tiếp bus data với hệ vi xử lý nên luôn có ngõ ra 3 trạng thái
 - Các chân điều khiển: \overline{CE} (\overline{CS}) chọn chip, \overline{OE} cho phép xuất, \overline{PGM} , \overline{WE} ghi data....
- Khi thiết kế bộ nhớ, trước tiên phải lập bảng phân vùng địa chỉ bộ nhớ, từ đó thiết kế mạch giải mã địa chỉ chọn chip bộ nhớ (nếu sử dụng hơn 1 chip bộ nhớ)

Ví dụ 0.20: Cho biết vùng địa chỉ một bộ nhớ dung lượng 8KB nếu chọn địa chỉ đầu bộ nhớ là 0000H? 2000H?

Giải:

Bộ nhớ dung lượng 8KB=2¹³ cần 13 đường địa chỉ A0 – A12.

Từ ký hiệu địa chỉ đầu 16 bit A0 – A15, ta lập được bảng phân vùng địa chỉ như sau:

ĐC đầu	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Vùng ĐC
0000H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFH
2000H	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2000H
	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3FFFH

0.3.1 IC nhớ 27C64: bộ nhớ EPROM 8KB (ghi bằng điện, xóa bằng tia hồng ngoại)

- Thường được sử dụng làm bộ nhớ chương trình do vẫn lưu data khi cắt nguồn cấp điện.
- Hình 0.11 là sơ đồ chân IC 27C64 và bảng sự thật như bảng B0.6.

		Name	Function
A0-A12	Address Inputs		
\overline{CE}	Chip Enable		
\overline{OE}	Output Enable		
\overline{PGM}	Program Enable		
VPP	Programming Voltage		
O0 - O7	Data Output		
Vcc	+5V Power Supply		
Vss	Ground		
NC	No Connection; No Internal Connections		
NU	Not Used; No External Connection Is Allowed		

Hình 0.11: Sơ đồ chân IC 27C64

Bảng B0.6: Các mô thức làm việc IC27C64

Operation Mode	\overline{CE}	\overline{OE}	\overline{PGM}	V_{PP}	A9	O0 - O7
Read	V _{IL}	V _{IL}	V _{IH}	V _{CC}	X	DOUT
Program	V _{IL}	V _{IH}	V _{IL}	V _H	X	DIN
Program Verify	V _{IL}	V _{IL}	V _{IH}	V _H	X	DOUT
Program Inhibit	V _{IH}	X	X	V _H	X	High Z
Standby	V _{IH}	X	X	V _{CC}	X	High Z
Output Disable	V _{IL}	V _{IH}	V _{IH}	V _{CC}	X	High Z
Identity	V _{IL}	V _{IL}	V _{IH}	V _{CC}	V _H	Identity Code

X = Don't Care

 $V_{IH}=1, V_{IL}=0; V_{CC}=5V, V_H=12-13.5V$

❖ Các mô thức thường sử dụng:

- Mô thức lập trình(Program): $\overline{CE} = 0, \overline{OE} = 1, \overline{PGM} = 0, V_{pp} = 12V, O_0 - O_7 : data\ input$
- Mô thức đọc(Read): $\overline{CE} = 0, \overline{OE} = 0, \overline{PGM} = 1, V_{pp} = 5V, O_0 - O_7 : data\ output$
- Mô thức cấm xuất(Output disable): $\overline{CE} = 0, \overline{OE} = 1, \overline{PGM} = 1, V_{pp} = 5V, O_0 - O_7 : hi - Z$

Lập trình nạp data cho IC27C64 sử dụng bộ lập trình chuyên dụng.

Trong hệ vi xử lý thường sử dụng IC ROM mô thức đọc và cấm xuất.

0.3.2 IC HM6264 RAM tĩnh 8KB

NC	1	28	V _{CC}
A12	2	27	\overline{WE}
A7	3	26	CS2
A6	4	25	A8
A5	5	24	A9
A4	6	23	A11
A3	7	22	\overline{OE}
A2	8	21	A10
A1	9	20	CS1
A0	10	19	I/O8
I/O1	11	18	I/O7
I/O2	12	17	I/O6
I/O3	13	16	I/O5
VSS	14	15	I/O4

- A0 -A12(inputs) : 13 đường địa chỉ
- I/O1 - I/O8(in/outputs): 8 đường data vào/ra có 3 trạng thái
- /CS1,CS2(inputs): điều khiển chọn chip
- /OE(input): điều khiển cho phép xuất

Bảng 0.7: Các mode làm việc của ICHM6264

\overline{WE}	CS1	CS2	\overline{OE}	Mode	I/O pin	V _{CC} current	Note
x	H	x	x	Not selected (power down)	High Z	I _{SB} , I _{SB1}	
x	x	L	x		High Z	I _{SB} , I _{SB1}	
H	L	H	H	Output disabled	High Z	I _{CC}	
H	L	H	L	Read	Dout	I _{CC}	Read cycle
L	L	H	H	Write	Din	I _{CC}	Write cycle 1
L	L	H	L	Write	Din	I _{CC}	Write cycle 2

Note: x: Don't care.

Hình 0.12: Sơ đồ chân ICHM6264

❖ Các mô thức thường sử dụng:

- Mô thức đọc(Read): $\overline{WE} = 1, \overline{CS1} = 0, CS2 = 1, \overline{OE} = 0, I/O_1 - I/O_8 : data\ output$
- Mô thức ghi(Write): $\overline{WE} = 0, \overline{CS1} = 0, CS2 = 1, \overline{OE} = 1, I/O_1 - I/O_8 : data\ input$
- Mô thức cấm xuất(Output disable): $\overline{WE} = 1, \overline{CS1} = 0, CS2 = 1, \overline{OE} = 1, I/O_1 - I/O_8 : hi-Z$

Ví dụ 0.21: Giả sử hệ vi xử lý có bus địa chỉ từ A0 - A12, bus data từ D0 - D7 và bus điều khiển gồm các đường \overline{PSEN} truy xuất mã chương trình, \overline{RD} , \overline{WR} lần lượt là tín hiệu truy xuất đọc và ghi data.

Về sơ đồ giao tiếp hệ vi xử lý(VXL) với 1 chip ROM 27C64 chứa mã chương trình và 1 chip RAM HM6264 chứa data.

Giải:

Từ sơ đồ chân và bảng sự thật IC 27C64 và HM6264, kết nối như sau:

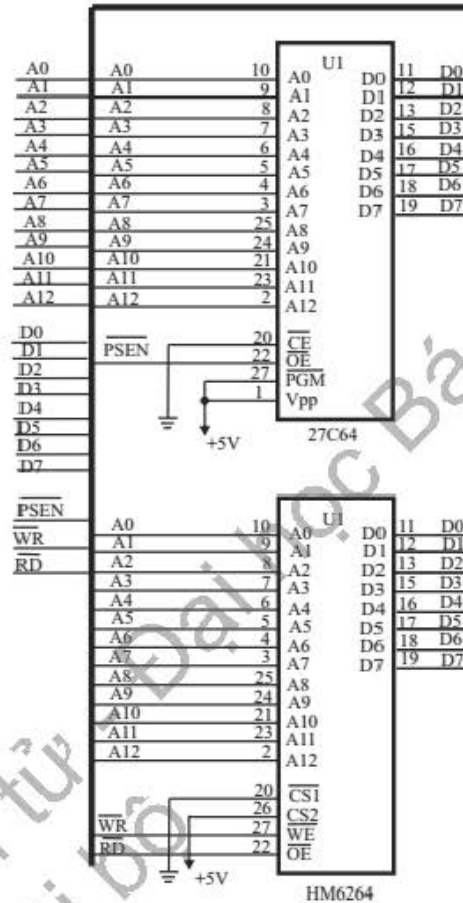
- IC27C64:
 - Các đường địa chỉ A0 -A12 kết nối đến A0 - A12 VXL

- Các đường data D0 – D7 kết nối đến D0 – D7 VXL
- \overline{OE} kết nối với \overline{PSEN} VXL do IC 27C64 chứa mã chương trình
- \overline{CE} nối GND luôn được xuất do chỉ có 1 chip
- $\overline{PGM} = V_{pp} = +5V$ mô thức đọc/cắm xuất

■ IC HM6264:

- Các đường địa chỉ A0 – A12 kết nối đến A0 – A12 VXL
- Các đường data D0 – D7 kết nối đến D0 – D7 VXL
- \overline{OE} kết nối với \overline{RD} VXL, \overline{WE} kết nối với \overline{WR} VXL do IC6264 có thể đọc/ghi data
- $\overline{CS1}$ nối GND $\overline{CS2} = 1$ nối +5V luôn được xuất do chỉ có 1 chip

Sơ đồ sau cùng như hình 0.13



Hình 0.13

0.3.3 Các bộ nhớ EEPROM nối tiếp

■ Bộ nhớ EEPROM 25AA1024

IC 25AA1024 là bộ nhớ EEPROM giao tiếp nối tiếp theo phương thức truyền SPI.

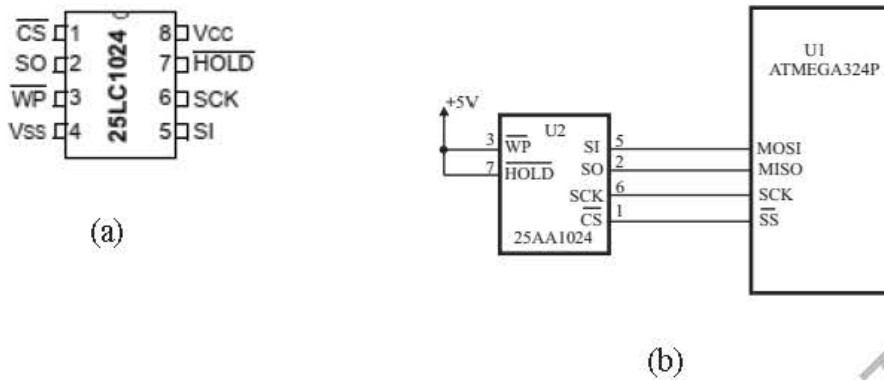
- Dung lượng 1024bit=128KB, truy xuất theo byte và định dạng trang =256 byte
- Tần số CKmax=20Mhz
- Chu kỳ ghi max=6ms
- Giới hạn xóa/ghi 1M chu kỳ

Hình 0.14a là sơ đồ chân IC 25AA1024 và hình 0.14b là sơ đồ kết nối IC25AA1024 với VXL.

Chức năng các chân:

- SI: Data in - ghi data vào bộ nhớ
- SO: Data out - đọc data từ bộ nhớ
- SCK: xung CK - nhận xung CK đồng bộ data
- /CS: chọn chip - tích cực mức 0 chọn chip
- /WP: Cắm ghi - tích cực mức 0 cấm ghi
- /HOLD: Giữ ngõ vào - tích cực mức 0 treo(cách ly) tạm thời ngõ vào

Trong phương thức truyền SPI,VXL đóng vai trò Master,bộ nhớ là Slaver.Master điều khiển việc phát xung SCK và mô thức đọc/ghi bộ nhớ.



Hình 0.14: (a) Sơ đồ chân 25AA1024,(b) Sơ đồ kết nối 25AA1024

■ Bộ nhớ EEPROM AT24C1024

IC AT24C1024 là bộ nhớ EEPROM giao tiếp nối tiếp theo phương thức truyền I2C.

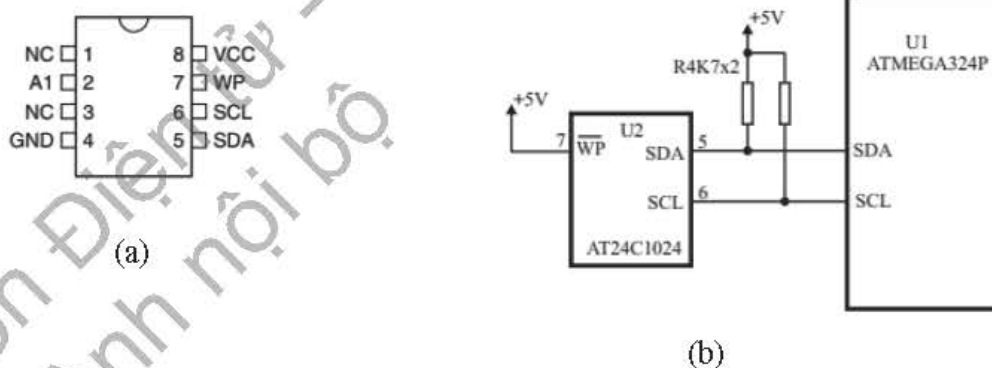
- Dung lượng 1024bit=128KB, truy xuất theo byte và định dạng trang =256 byte
- Tần số CKmax=1Mhz
- Chu kỳ ghi max=5ms
- Giới hạn xóa/ghi 100K chu kỳ

Hình 0.15a là sơ đồ chân IC AT24C1024 và hình 0.15b là sơ đồ kết nối IC AT24C1024 với VXL.

Chức năng các chân:

- SCL: Data in/out - ghi/đọc data vào/từ bộ nhớ
- SCL: xung CK - nhận xung CK đồng bộ data
- A1: chọn địa chỉ - kết nối mức 1/0 chọn địa chỉ chip mở rộng, hoặc thả nổi
- /WP: Cắm ghi - tích cực mức 0 cấm ghi

Trong phương thức truyền I2C,VXL đóng vai trò Master,bộ nhớ là Slaver.Master điều khiển việc phát xung SCL và mô thức đọc/ghi bộ nhớ.



Hình 0.15: (a) Sơ đồ chân AT24C1024,(b) Sơ đồ kết nối AT24C1024 với VXL