Ch9（补充内容） Memory and FPGA

Main contents:

1、Memory types

- Read Only Memory (ROM)

  Mask ROM、PROM、EPROM、EEPROM、FLASH

- Random Access Memory (RAM)

  Static SRAM、Dynamic DRAM

- Sequential Access Memory (SAM)

  FIFO、FILO

2、Basic functions and internal structure of ROMs

3、Using ROMs for Combinational Logic

4、SRAM and the expanding of RAM

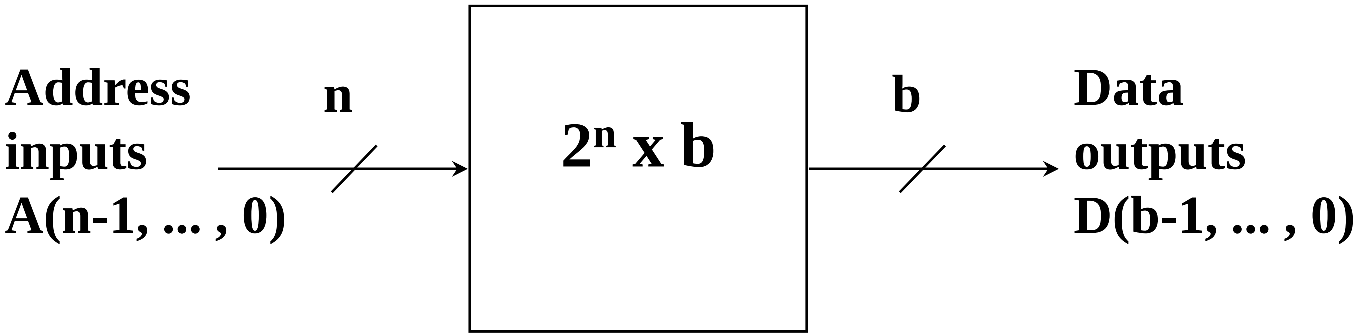
Types Of ROMs

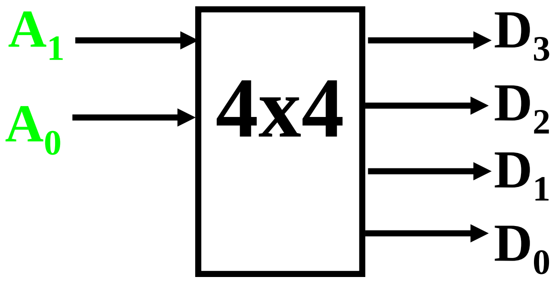
- Mask ROM
  - Connections made by the semiconductor vendor
  - Expensive setup cost, Several weeks for delivery. High volume only
  - Bipolar or MOS technology
- PROM
  - Programmable ROM
  - Vaporize (blow) fusible links with PROM programmer using high voltage/current pulses
  - Bipolar technology
  - One-time programmable
- EPROM
  - Erasable Programmable ROM
  - Charge trapped on extra “floating gate” of MOS transistors
  - Exposure to UV light removes charge.   Limited number of erasures (10-100)

- EEPROM (E^{2}ROM)
  - Electrically Erasable ROM
  - Not RAM (relatively slow charge/discharge)
  - limited number of charge/discharge cycles (10,000)
- Flash
  - Electronically erasable in blocks
  - 100,000 erase cycles
  - Simpler and denser than EEPROM

Basic functions of ROMs

- A combinational circuit with n inputs and b outputs:

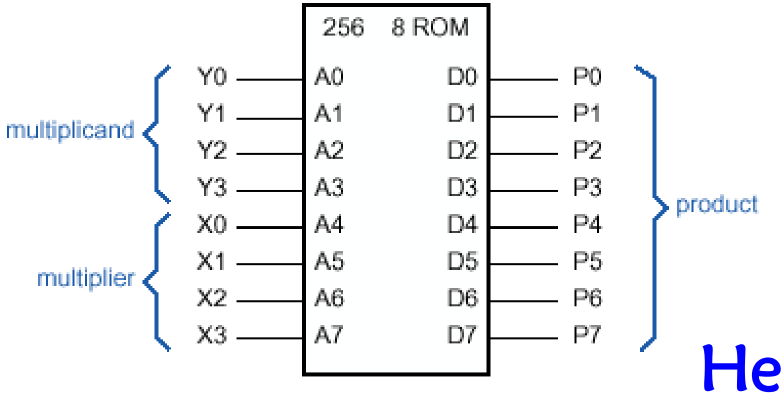




00	0101
01	1011
10	0100
11	1110

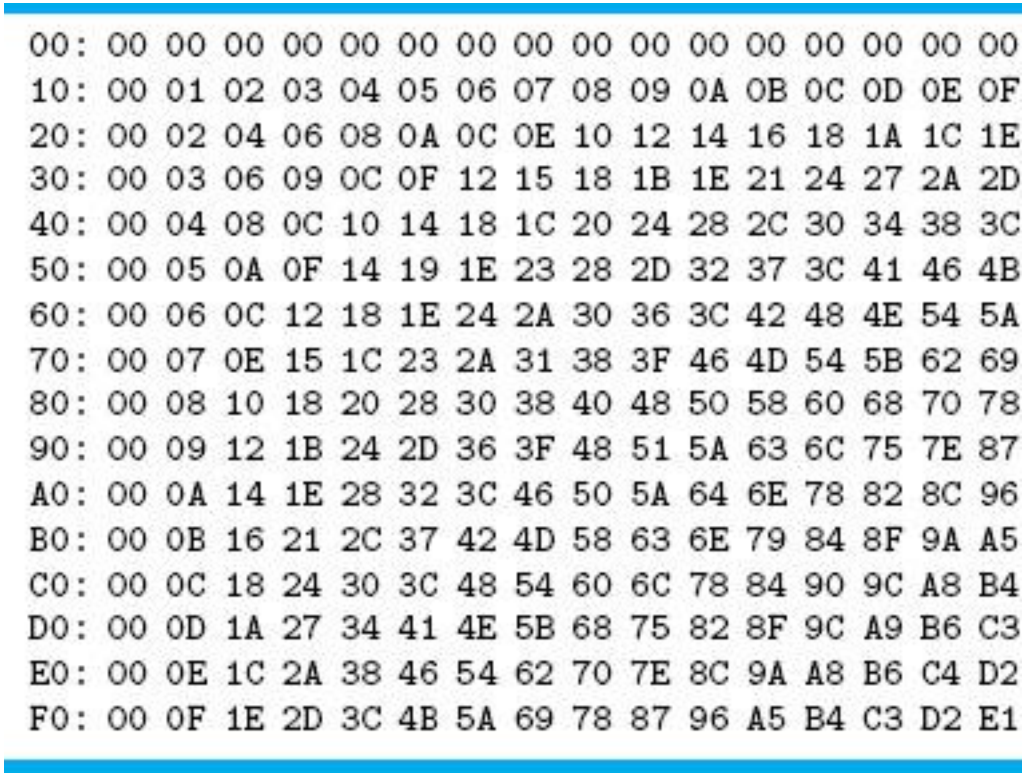
address		data			
A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	1	0	1
0	1	1	0	1	1
1	0	0	1	0	0
1	1	1	1	1	0

A 4x4 multiplier using a 256x8 ROM



Use a C-program to generate this table

Hexadecimal Text File Specifying the Contents of a 4 × 4 Multiplier ROM

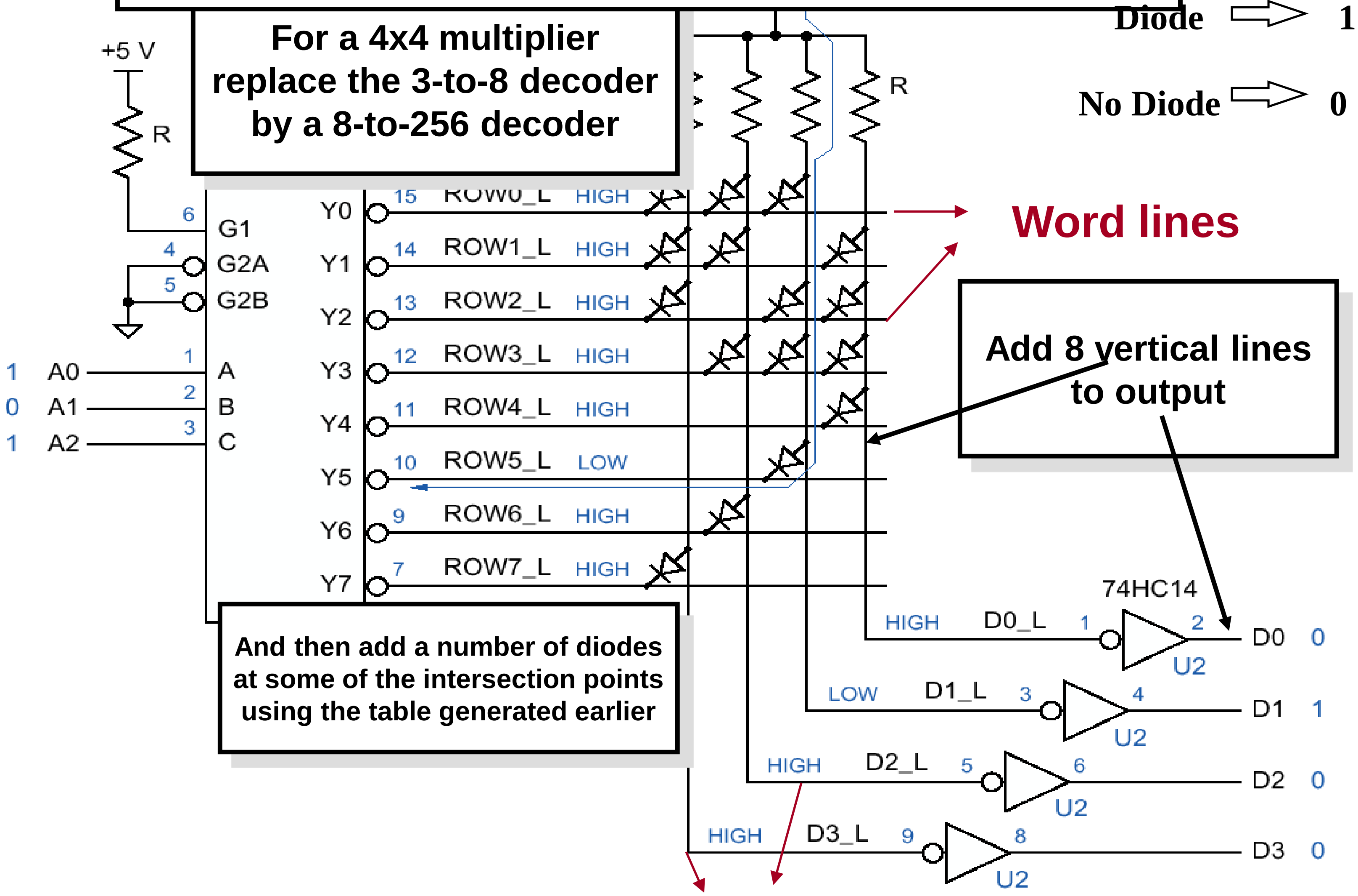


```
00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10: 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
20: 00 02 04 06 08 0A 0C 0E 10 12 14 16 18 1A 1C 1E
30: 00 03 06 09 0C 0F 12 15 18 1B 1E 21 24 27 2A 2D
40: 00 04 08 0C 10 14 18 1C 20 24 28 2C 30 34 38 3C
50: 00 05 0A 0F 14 19 1E 23 28 2D 32 37 3C 41 46 4B
60: 00 06 0C 12 18 1E 24 2A 30 36 3C 42 48 4E 54 5A
70: 00 07 0E 15 1C 23 2A 31 38 3F 46 4D 54 5B 62 69
80: 00 08 10 18 20 28 30 38 40 48 50 58 60 68 70 78
90: 00 09 12 1B 24 2D 36 3F 48 51 5A 63 6C 75 7E 87
A0: 00 0A 14 1E 28 32 3C 46 50 5A 64 6E 78 82 8C 96
B0: 00 0B 16 21 2C 37 42 4D 58 63 6E 79 84 8F 9A A5
C0: 00 0C 18 24 30 3C 48 54 60 6C 78 84 90 9C A8 B4
D0: 00 0D 1A 27 34 41 4E 5B 68 75 82 8F 9C A9 B6 C3
E0: 00 0E 1C 2A 38 46 54 62 70 7E 8C 9A A8 B6 C4 D2
F0: 00 0F 1E 2D 3C 4B 5A 69 78 87 96 A5 B4 C3 D2 E1
```

Logic diagram of a 8x4 diode ROM

And you have a 4x4 multiplier that will be real fast

For a 4x4 multiplier replace the 3-to-8 decoder by a 8-to-256 decoder



Diode ⇨ 1

No Diode ⇨ 0

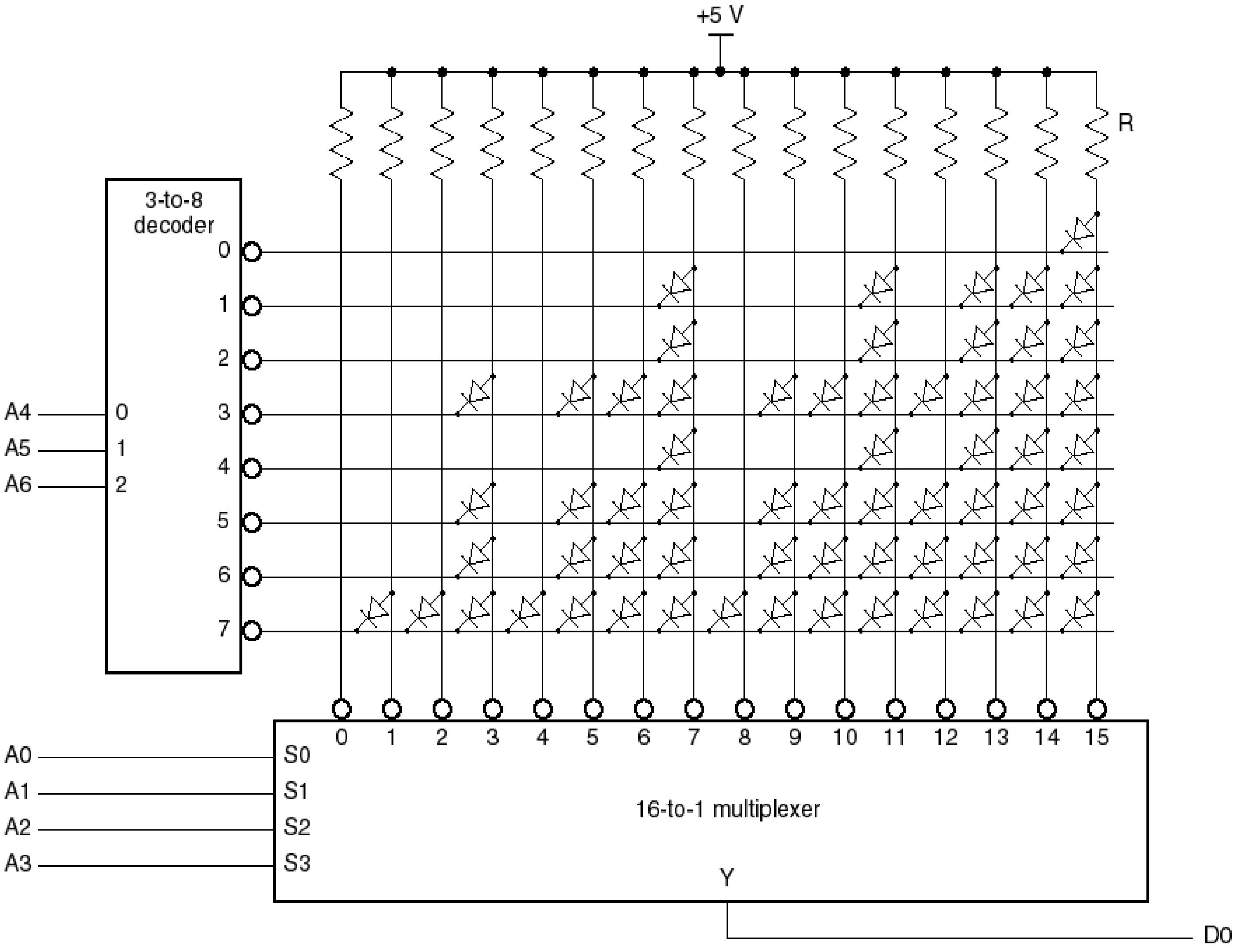
Word lines

Add 8 vertical lines to output

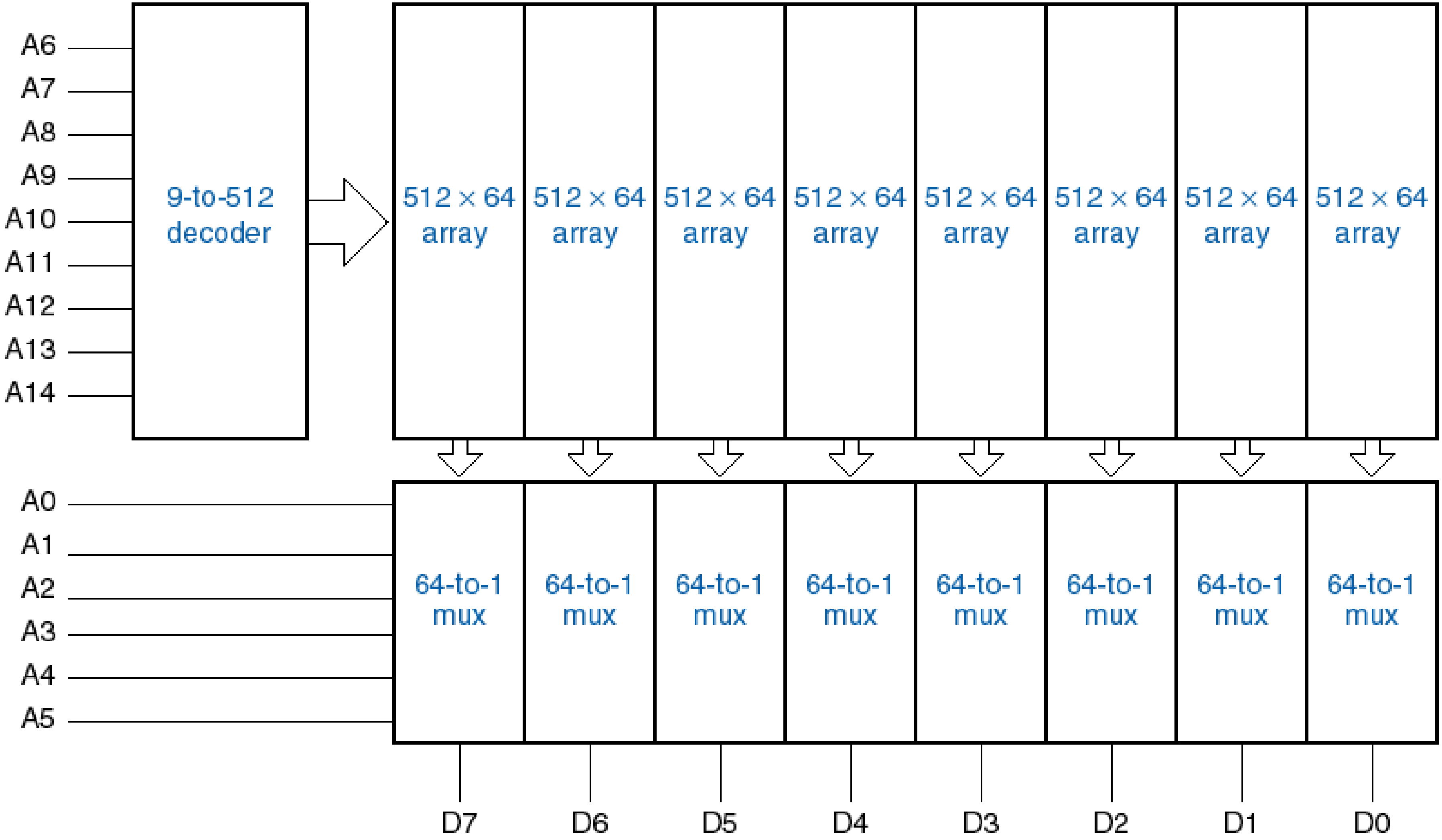
And then add a number of diodes at some of the intersection points using the table generated earlier

Bit lines

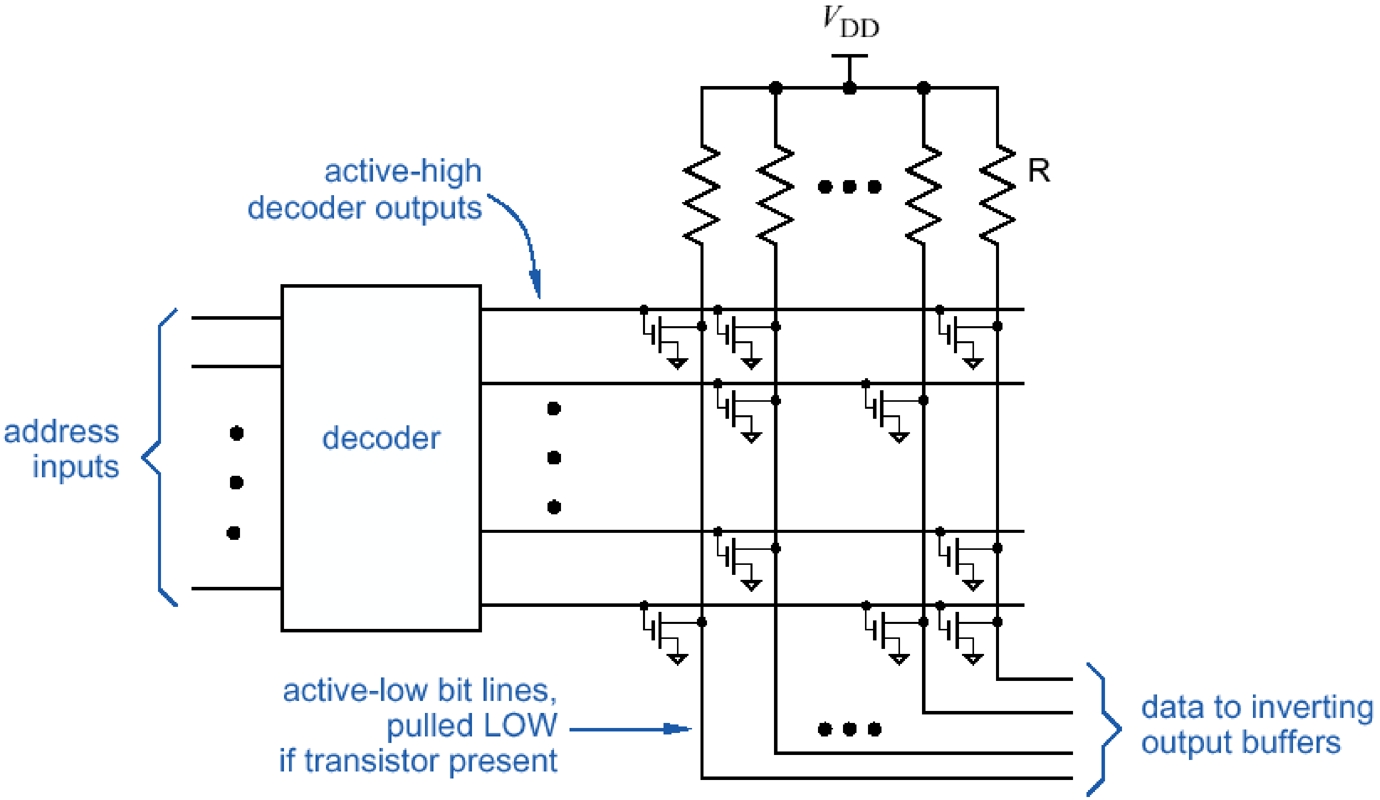
Two-dimensional decoding
128x1 ROM



Larger example, 32Kx8 ROM



MOS Transistors as Storage elements



V_{DD}
R
active-high decoder outputs
address inputs
decoder
active-low bit lines, pulled LOW if transistor present
data to inverting output buffers

Transistor ⇨ **1**

No transistor ⇨ **0**

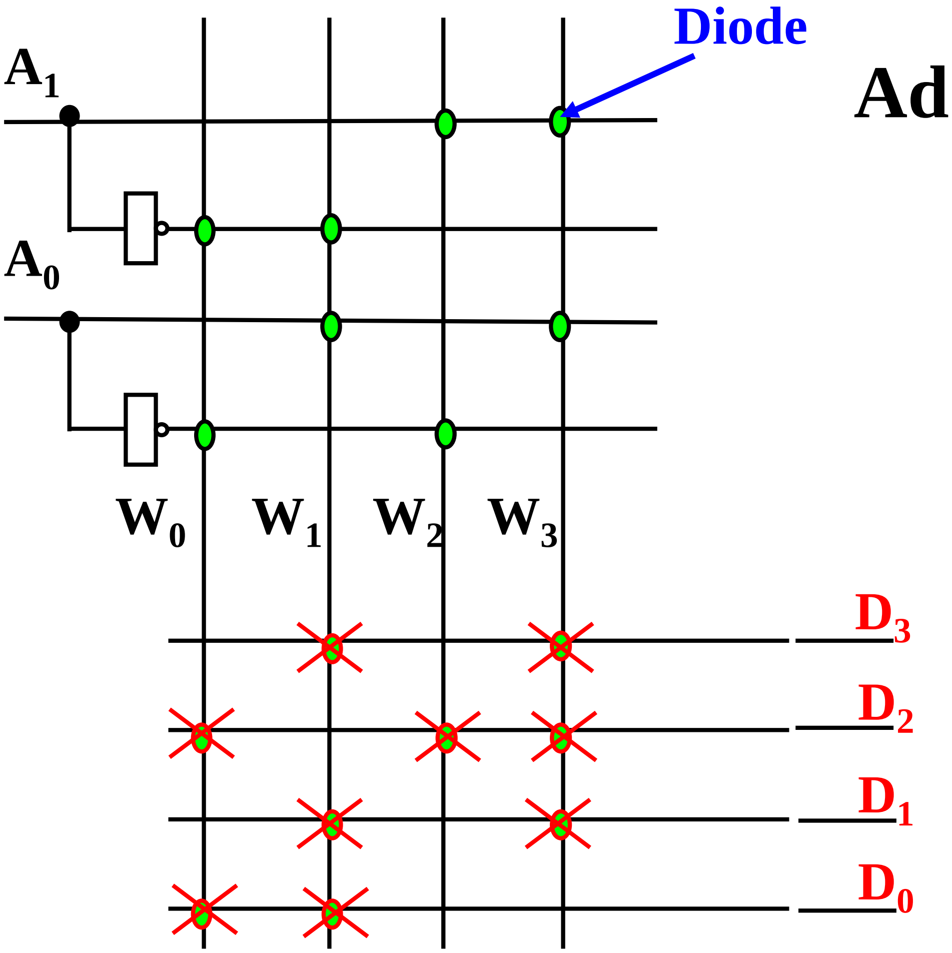
Using ROMs for Combinational Logic

- Two views of ROM:
  - stores 2^n words of b bits each, or
  - stores an n-input, b-output truth table

	n = 2		b = 4			
Example:	A1	A0	D3	D2	D1	D0
	0	0	0	1	0	1
	0	1	1	1	1	1
	1	0	0	0	0	1
	1	1	1	0	0	0

⇦ Stores 4 4-bit words, or
stores 4 functions of 2
input variables

Diode ROM simplified diagram



Address decode: AND array:

$$W_0 = \bar{A}_1 \bar{A}_0 \qquad W_1 = \bar{A}_1 A_0$$

$$W_2 = A_1 \bar{A}_0 \qquad W_3 = A_1 A_0$$

Store: OR array:

$$D_0 = W_0 + W_1 = \bar{A}_1 A_0 + \bar{A}_1 \bar{A}_0$$

$$D_1 = W_3 + W_1 = A_1 A_0 + \bar{A}_1 A_0$$

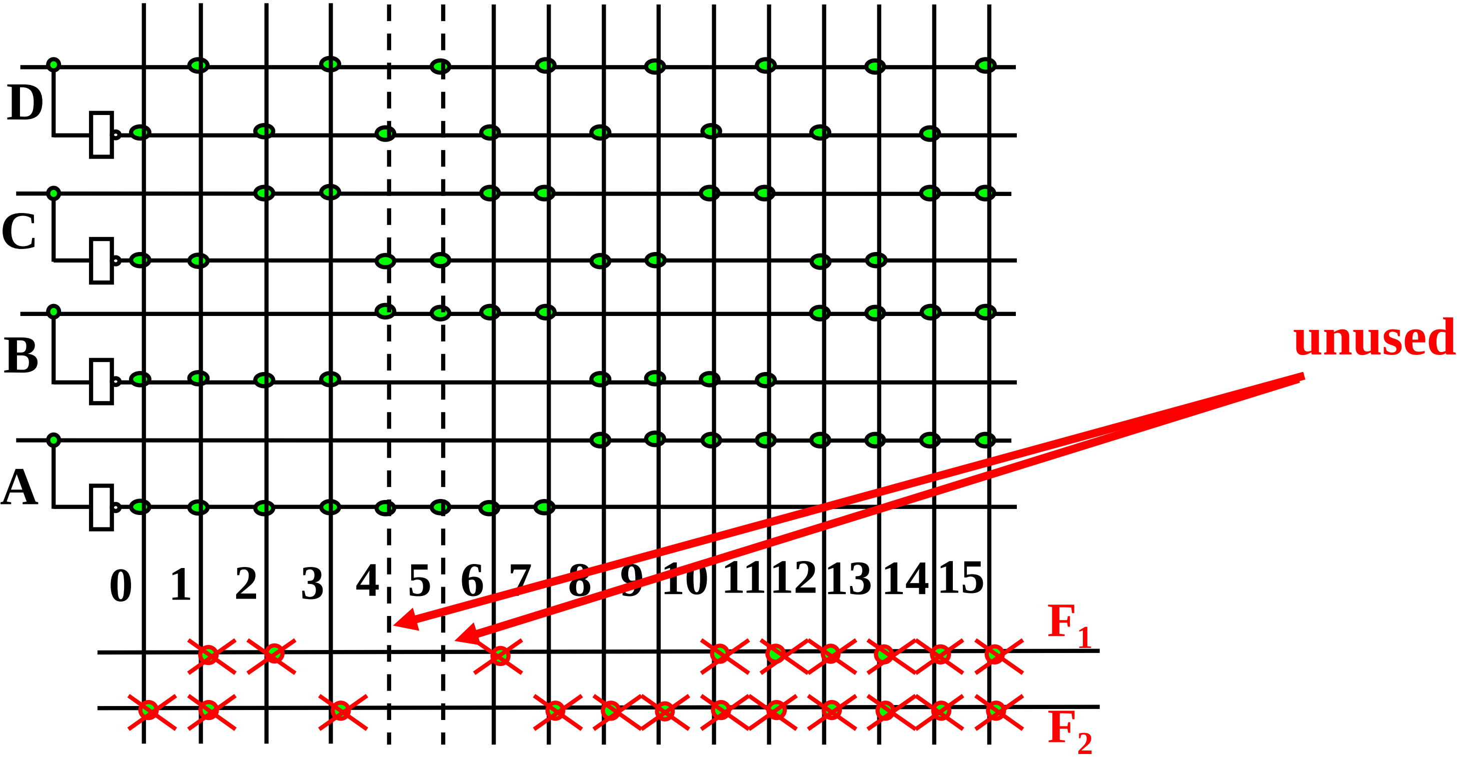
$$D_2 = W_3 + W_2 + W_0$$
$$= A_1 A_0 + A_1 \bar{A}_0 + \bar{A}_1 \cdot \bar{A}_0$$

$$D_3 = W_3 + W_1 = A_1 A_0 + \bar{A}_1 A_0$$

Example 1：using ROM to complete the function：

$$F_1(A,B,C,D)=m_1+m_2+m_6+m_{10}+m_{11}+m_{12}+m_{13}+m_{14}+m_{15}$$

$$F_2(A,B,C,D)=m_0+m_1+m_3+m_7+m_8+m_9+m_{10}+m_{11}+m_{12}+m_{13}+m_{14}+m_{15}$$



ROM store capacity：4×14×2

注：容量 = 输入数×乘积项数×输出数

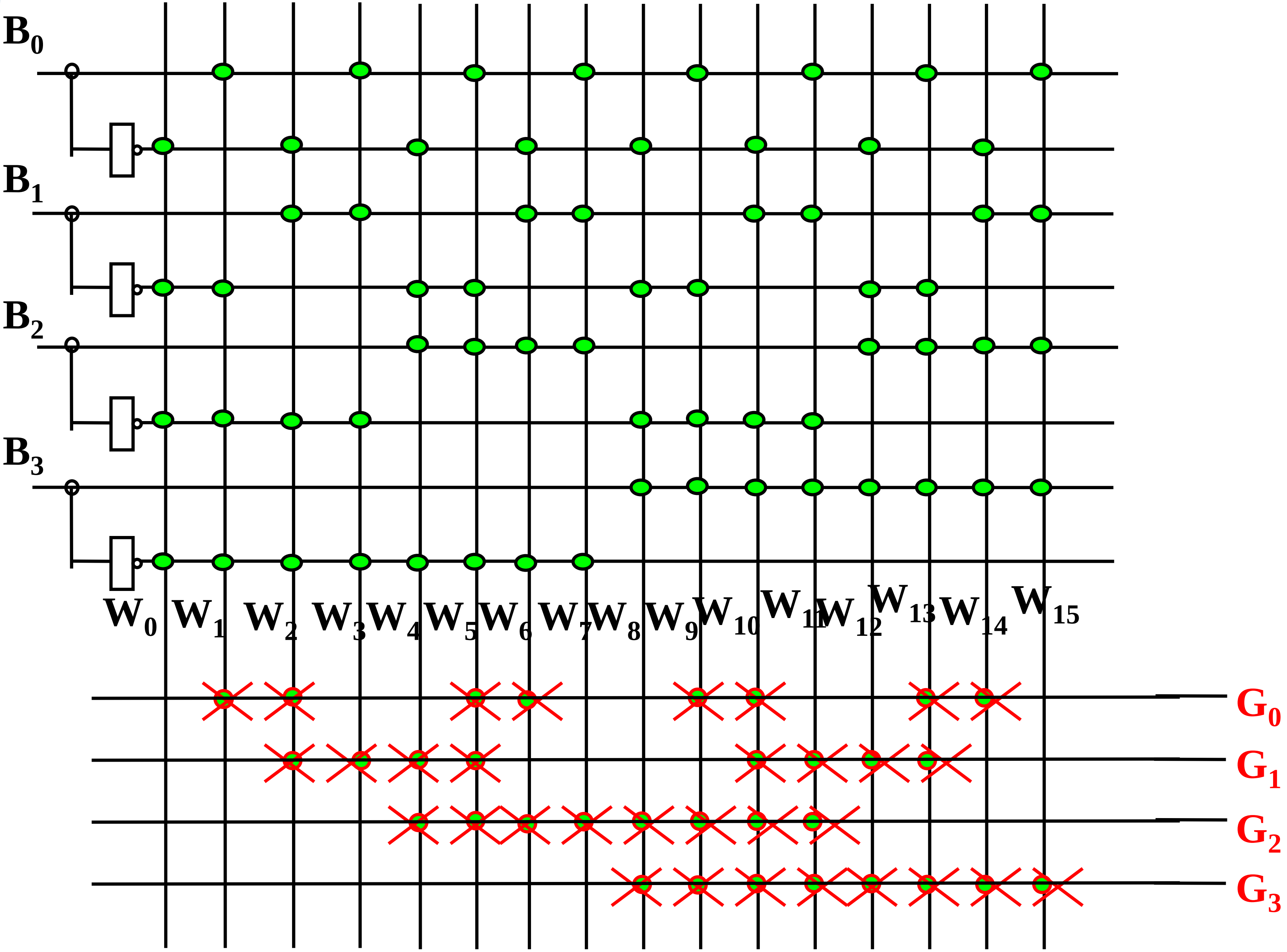
Example 2：using ROM to complete the conversion from binary code to gray code, draw ROM diagram。

B3 B2 B1 B0	G3 G2 G1 G0
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 1
0 0 1 1	0 0 1 0
0 1 0 0	0 1 1 0
0 1 0 1	0 1 1 1
0 1 1 0	0 1 0 1
0 1 1 1	0 1 0 0
1 0 0 0	1 1 0 0
1 0 0 1	1 1 0 1
1 0 1 0	1 1 1 1
1 0 1 1	1 1 1 0
1 1 0 0	1 0 1 0
1 1 0 1	1 0 1 1
1 1 1 0	1 0 0 1
1 1 1 1	1 0 0 0

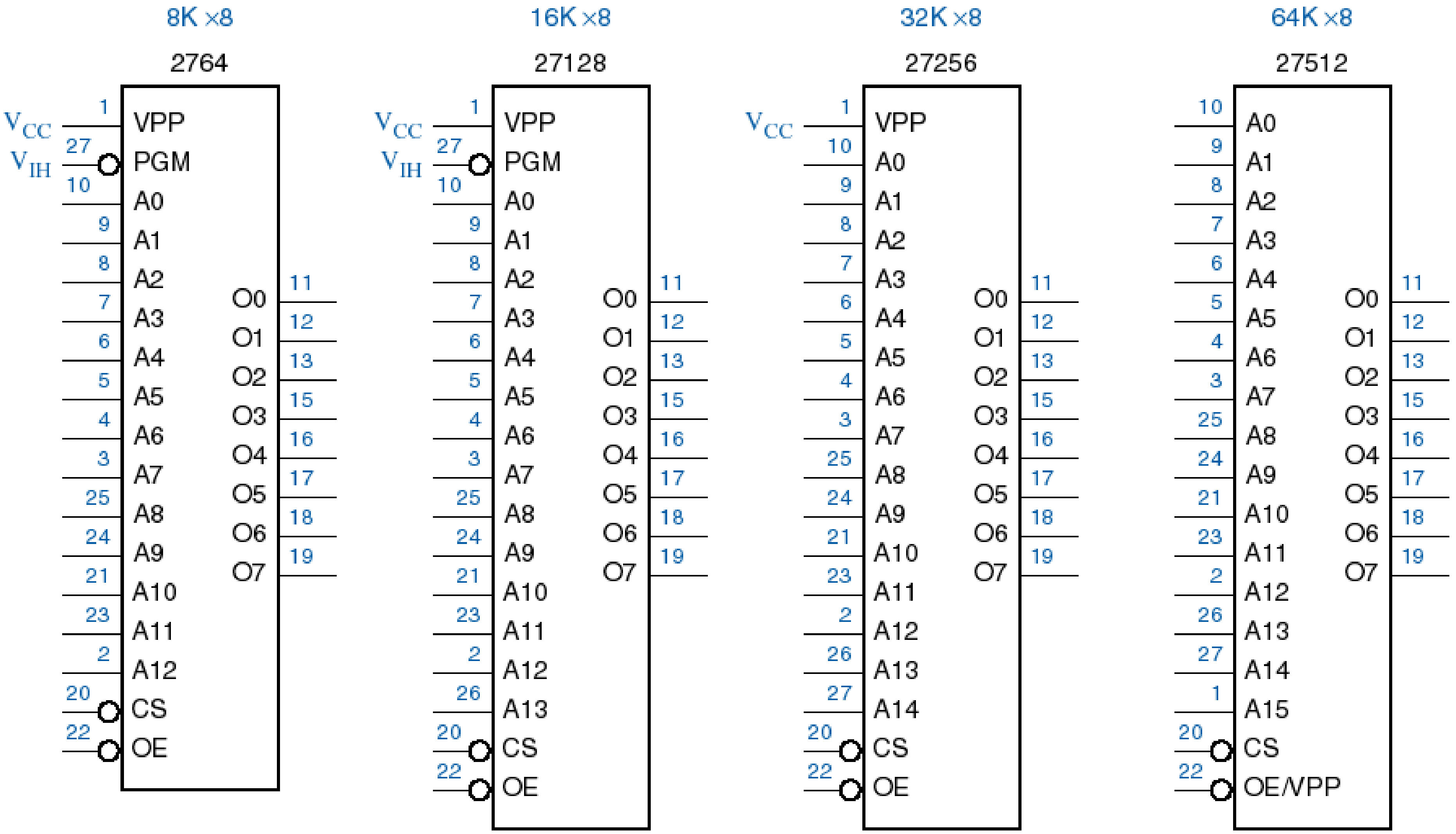
choose 2^4×4 ROM

input :B_3、B_2、B_1、B_0，address 4-bit;

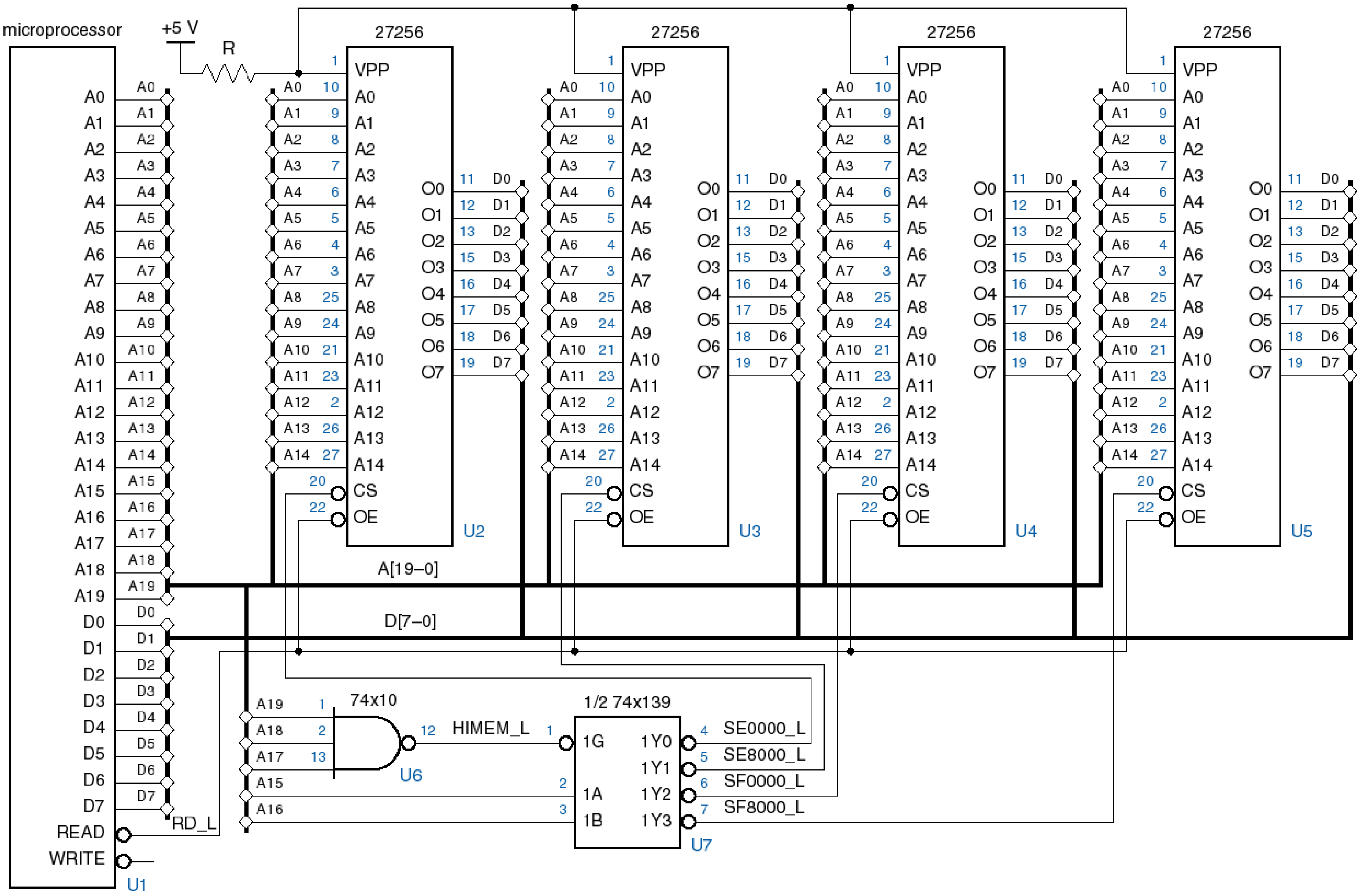
output: G_3、G_2、G_1、G_0，data 4-bit;


B_0
B_1
B_2
B_3
W_0
W_1
W_2
W_3
W_4
W_5
W_6
W_7
W_8
W_9
W_{10}
W_{11}
W_{12}
W_{13}
W_{14}
W_{15}
G_0
G_1
G_2
G_3

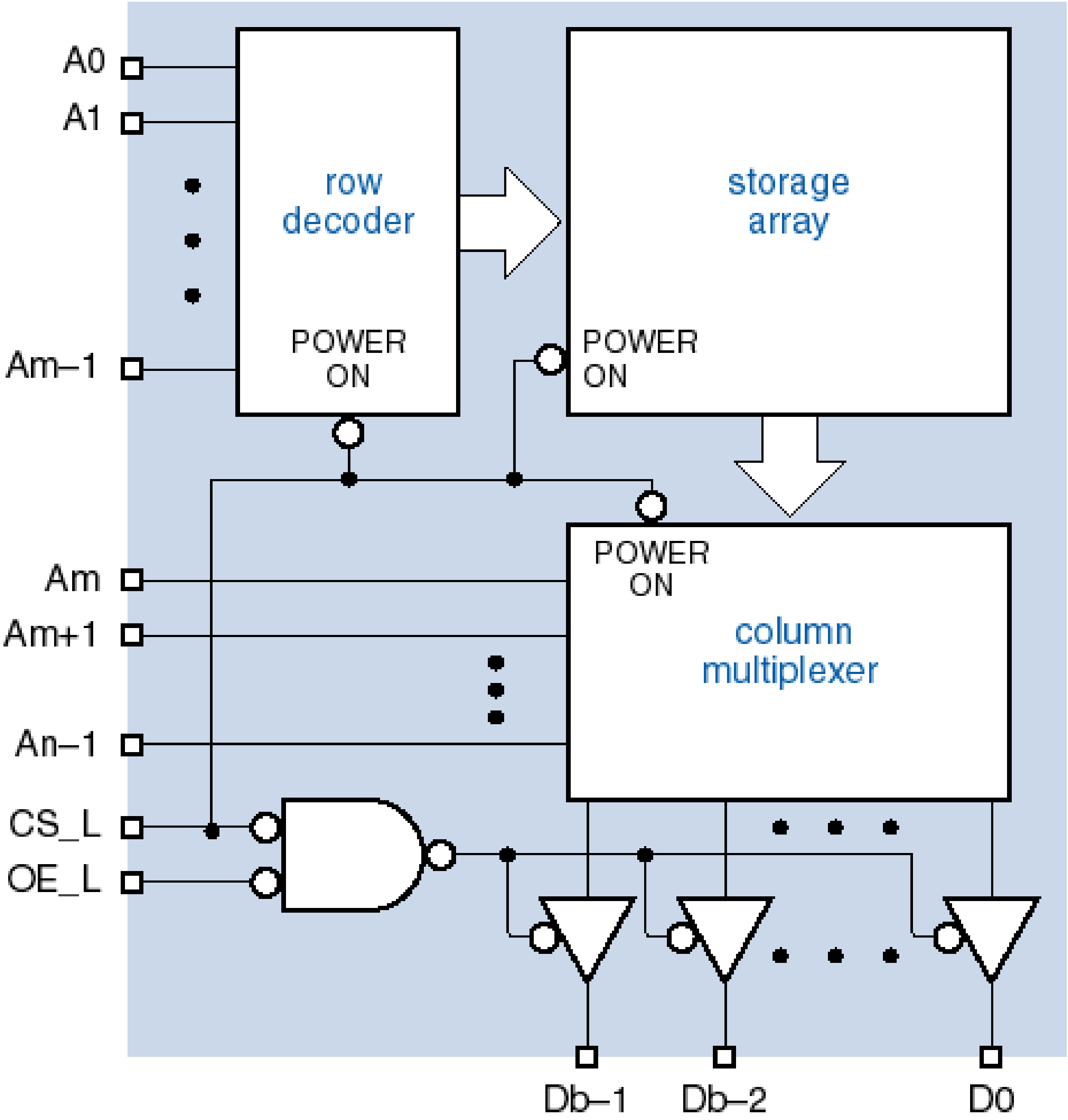
Typical commercial EEPROMs



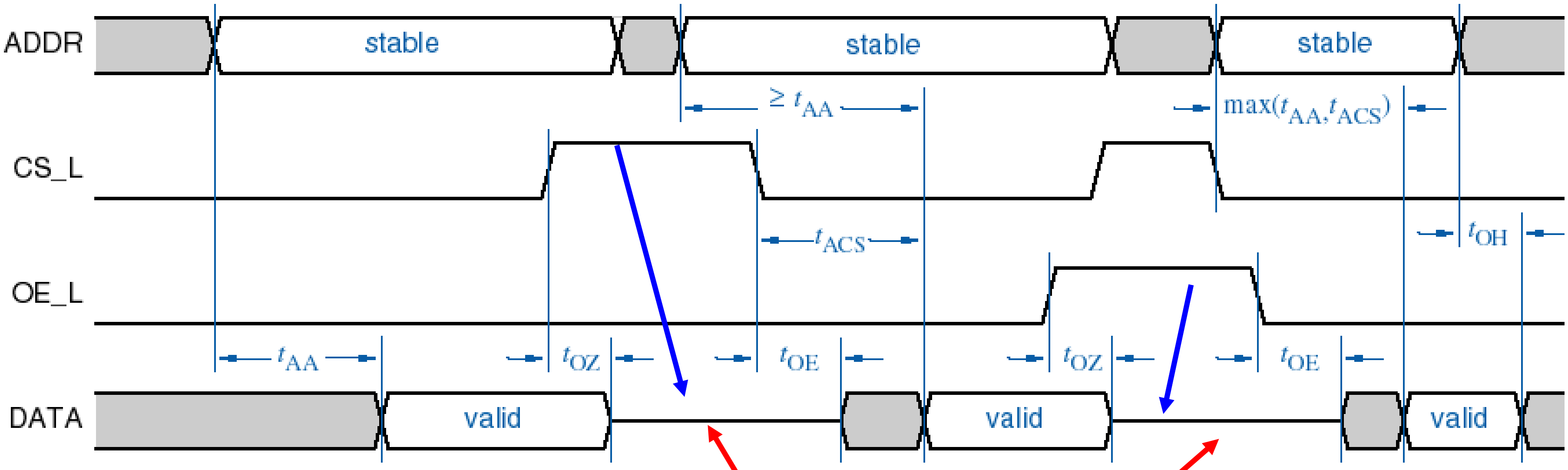
Microprocessor EPROM application



ROM control and I/O signals



ROM timing

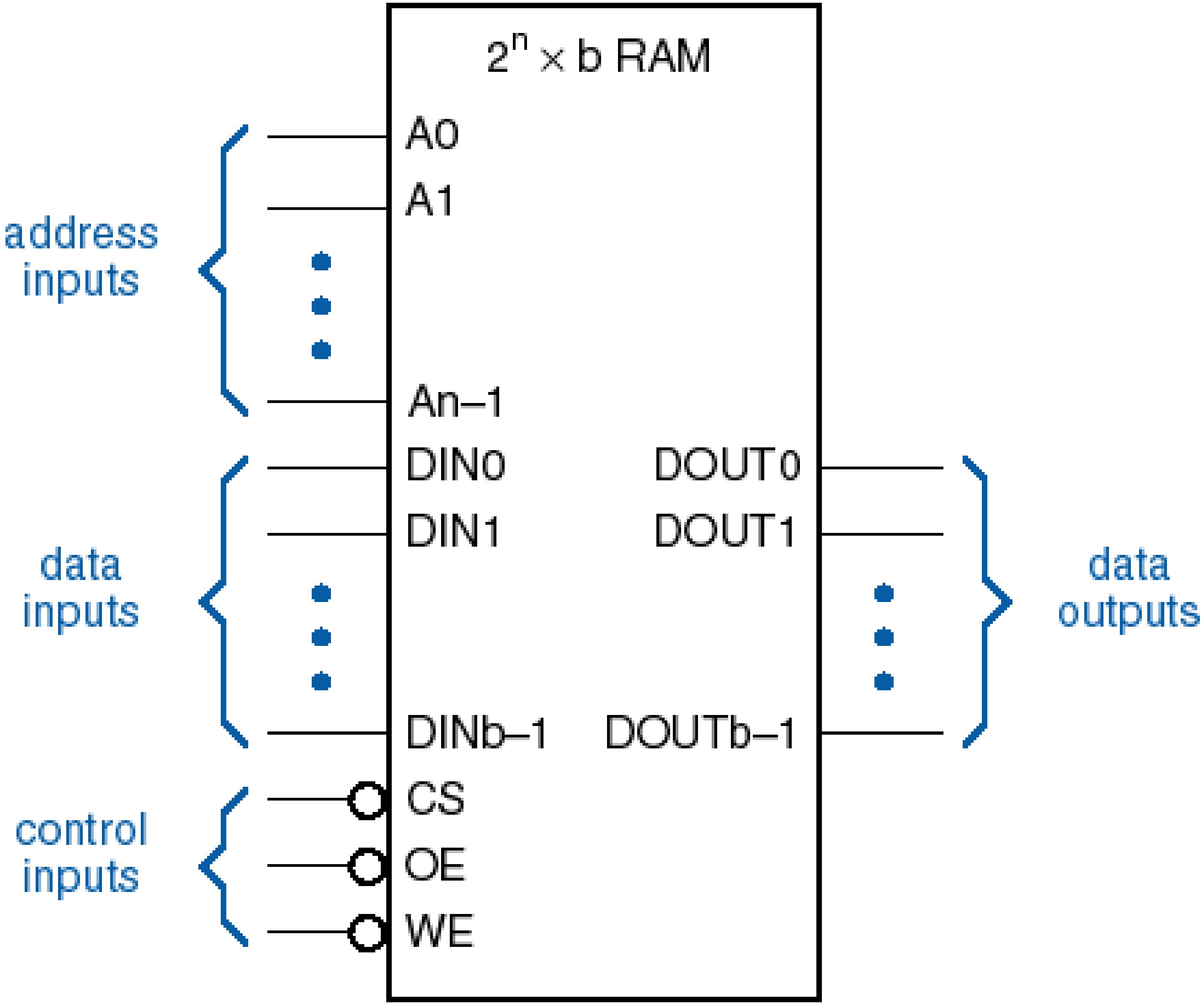


输出禁止时段

Read/Write Memories

- a.k.a. “RAM” (Random Access Memory)
- Volatility
  - Most RAMs lose their memory when power is removed
  - NVRAM = RAM + battery
  - Or use EEPROM
- SRAM (Static RAM)
  - Memory behaves like latches or flip-flops
- DRAM (Dynamic Memory)
  - Memory lasts only for a few milliseconds
  - Must “refresh” locations by reading or writing

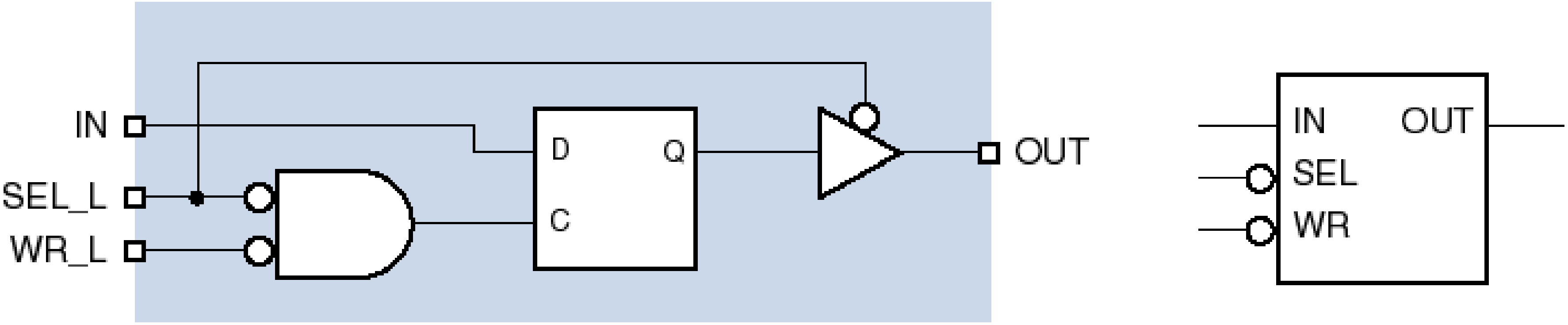
SRAM



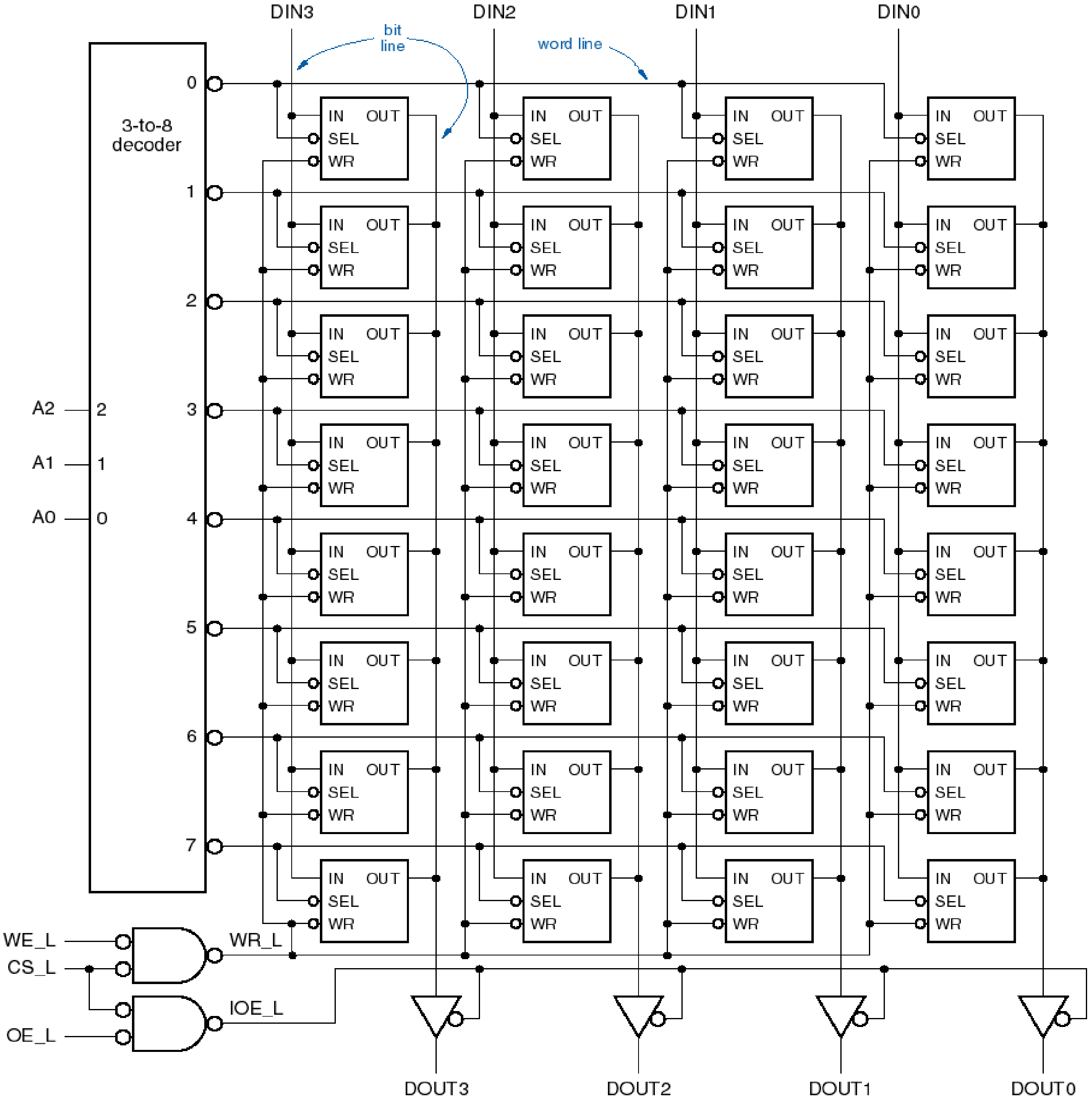
- Address/Control/Data Out lines (Reading)
- + Write Enable (WE) and Data In (DIN) (Writing)

SRAM operation

- Individual bits are D latches, *not* edge-triggered D flip-flops.
  - Fewer transistors per cell.
- Implications for write operations:
  - Address must be stable before writing cell.
  - Data must be stable before ending a write.

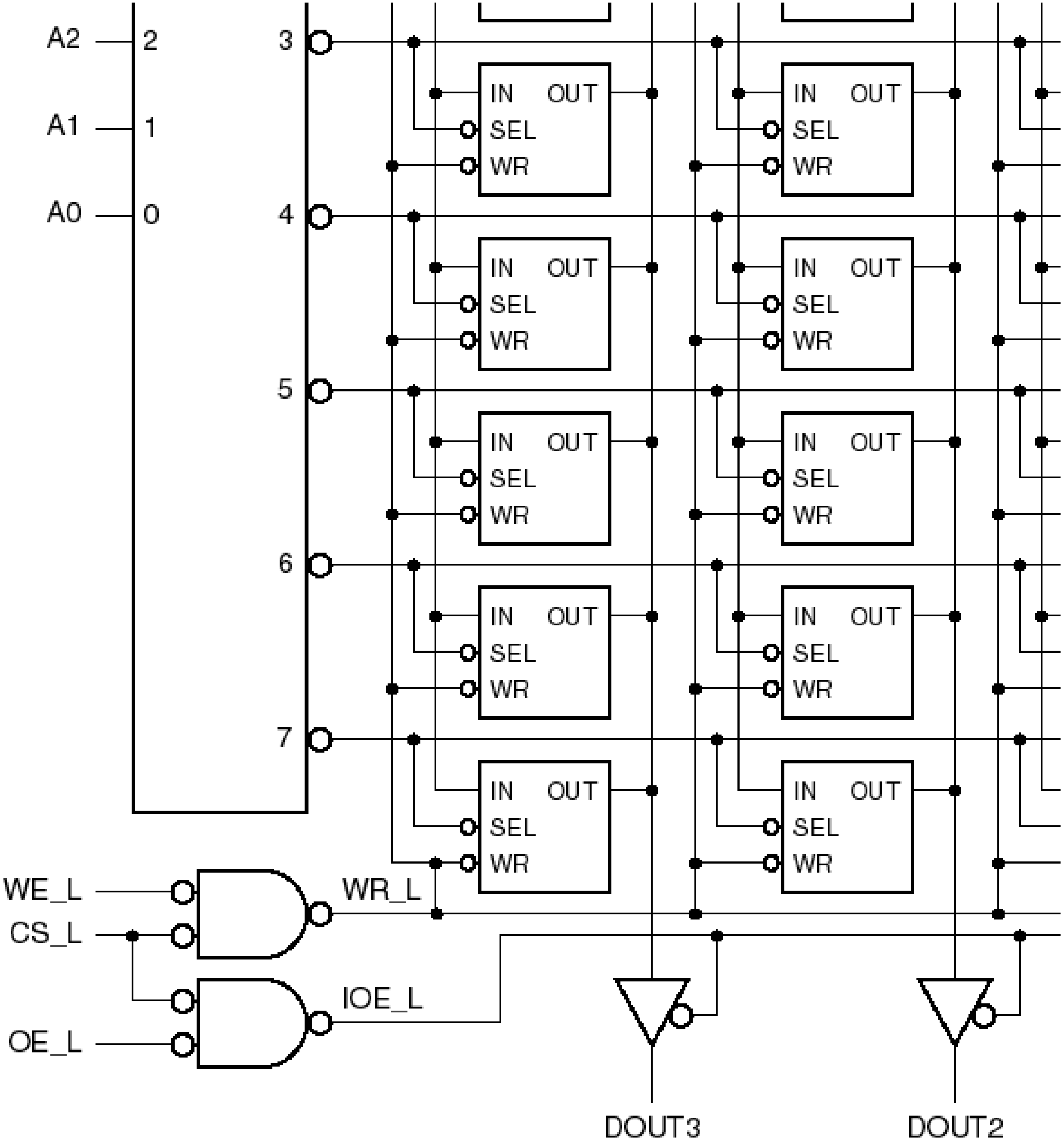


SRAM array


DIN3
DIN2
DIN1
DIN0
bit line
word line
3-to-8 decoder
0
1
2
3
4
5
6
7
A2
A1
A0
2
1
0
IN OUT
SEL
WR
WE_L
CS_L
OE_L
WR_L
IOE_L
DOUT3
DOUT2
DOUT1
DOUT0

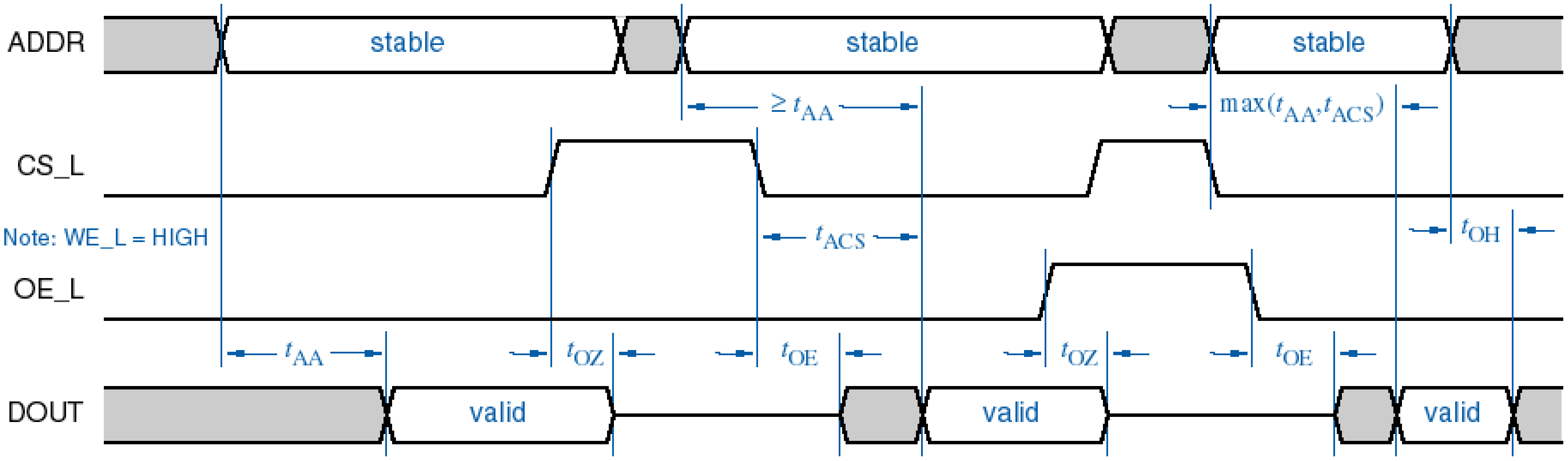
SRAM control lines

- Chip select
- Output enable
- Write enable

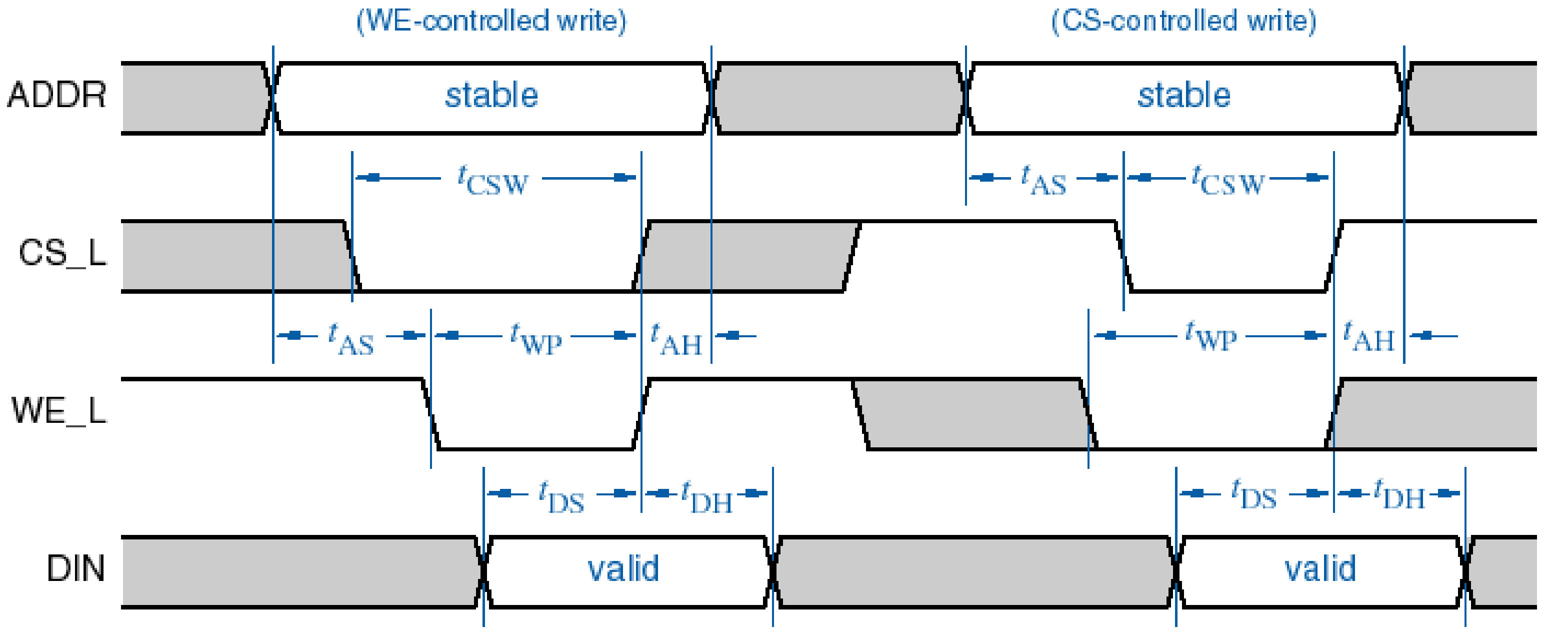


SRAM read timing

- Similar to ROM read timing

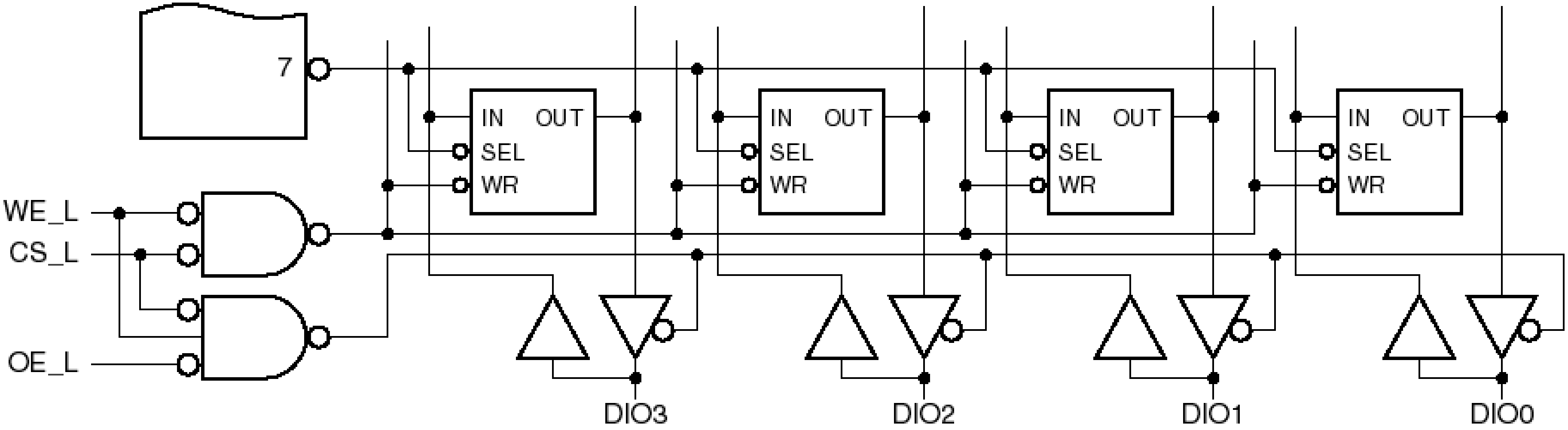


SRAM write timing



- **Address must be stable before and after write-enable is asserted.**
- **Data is latched on trailing edge of (WE & CS).**

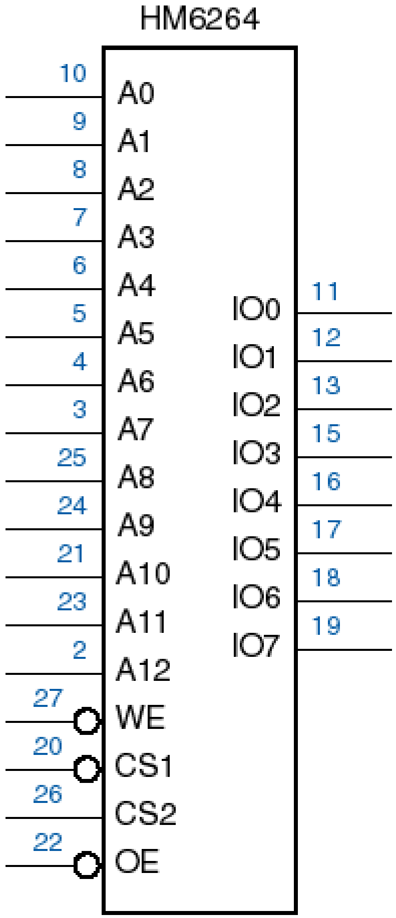
Bidirectional data in and out pins

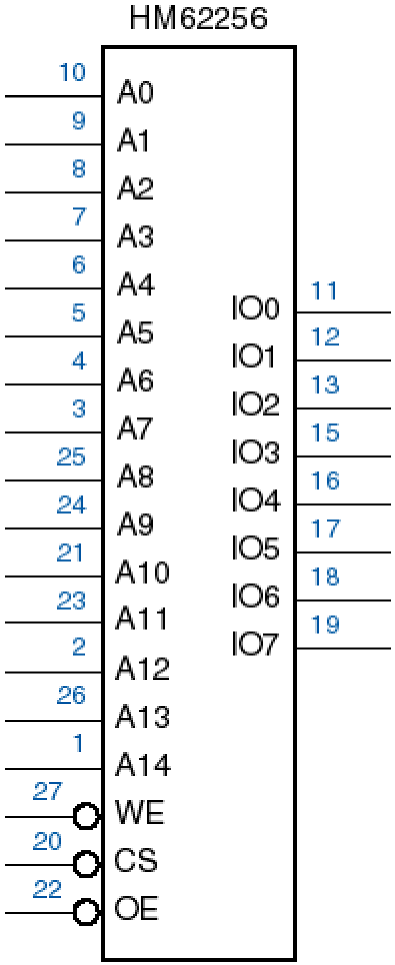


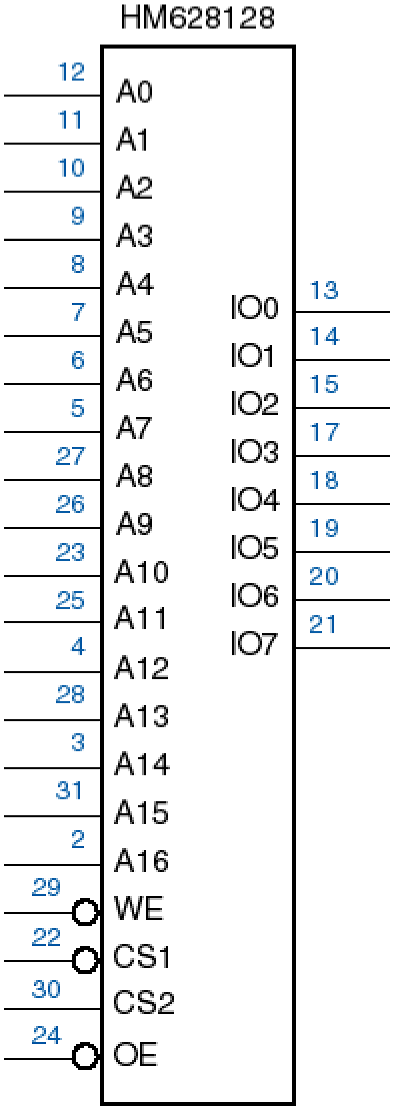
- Use the same data pins for reads and writes
  - Especially common on wide devices
  - Makes sense when used with microprocessor buses (also bidirectional)

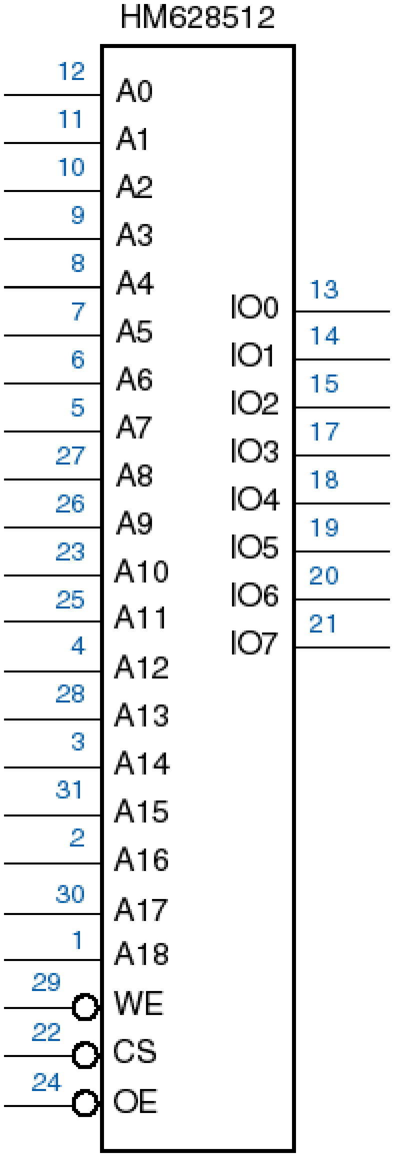
SRAM devices

- **Similar to ROM packages**



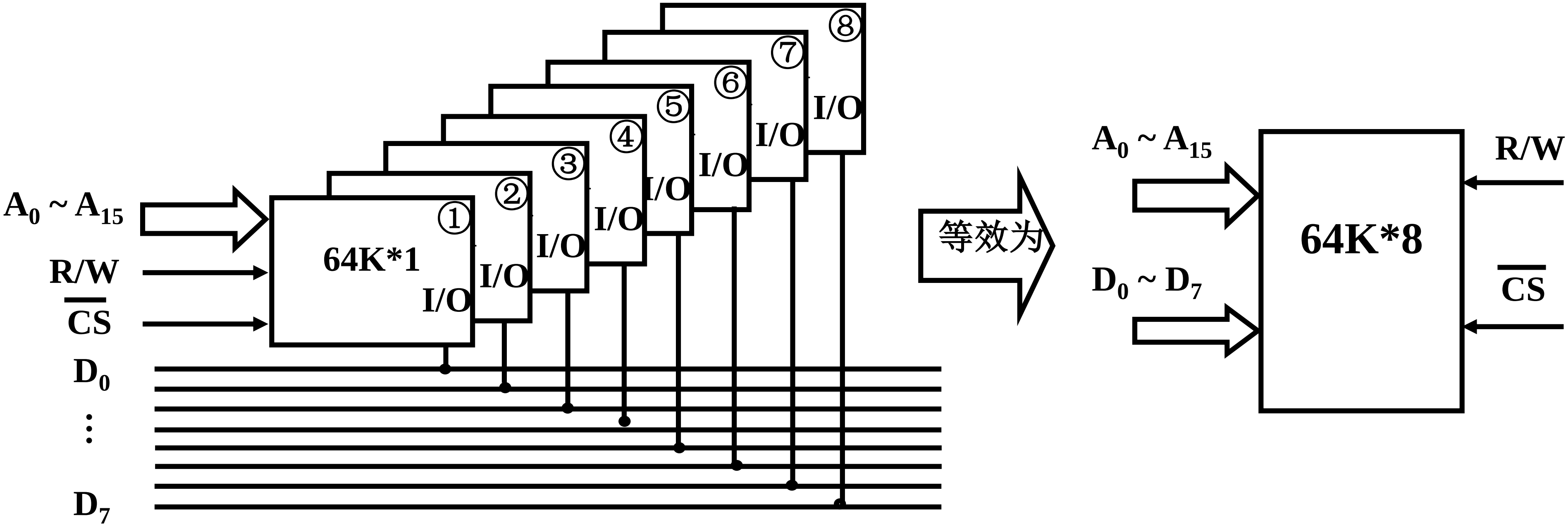






Expanding memory devices

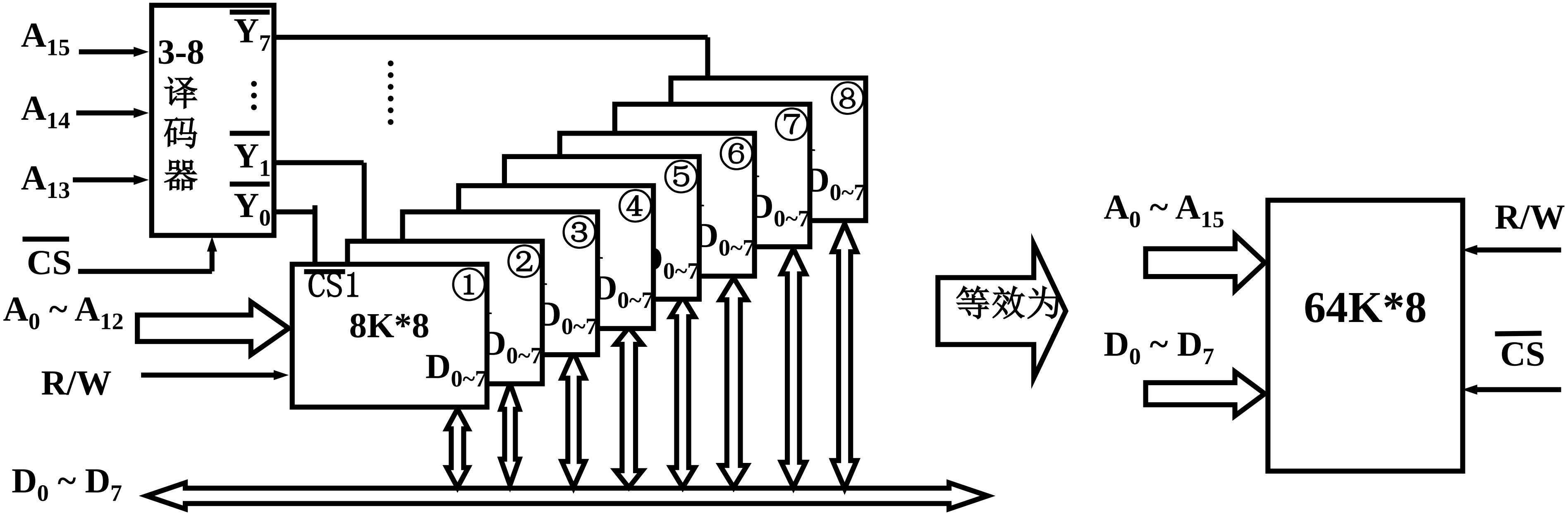
Examp.1：expand 64K×1 RAM into 64K×8RAM



bit expanding :need 8 RAM chips

位扩展时，模块中所有芯片的地址线和控制线互连形成整个模块的地址线和控制线，而各芯片的数据线并列形成整个模块的数据线。

Examp.2：expand 8K×8 RAM into 64K×8 RAM



word expanding: need 8 RAM chips

字扩展时，模块中所有芯片的地址线和读写信号线互连，扩展用的地址线可用来译码以形成对各个芯片的选择线（称为片选线）。

注：片选线的设计方法将确定每个芯片的地址范围。

1. A 4k × 8 ROM has (   ) bits of storage, (   ) address bits, and (   ) output data bits.

2. The capacity of a 4MB SRAM is expanded to (   ) if there is a 2-bit increase in its address.

3.To implement a logic circuit with 4 inputs and 3 outputs using a ROM, a (   ) ROM is required, at the very least.

解析：考查要点，ch9存储器的内容。

1. 32k;4k;8; 2. 2^2×4MB=16MB; 3. 2^4×3

4. A 1GB DRAM has (   )address lines.

5. A 4× 4 unsigned binary multiplication can be built with a (   ) ROM.
A. 128× 4   B. 256× 4
C. 128× 16   D. 256× 8

参考答案：
4. 30;
5. D 解析：4x4乘法运算，寻址线为8根，寻址能力为2^8=256，而数据线也为8根，故选D