Z-80 Microprocessor

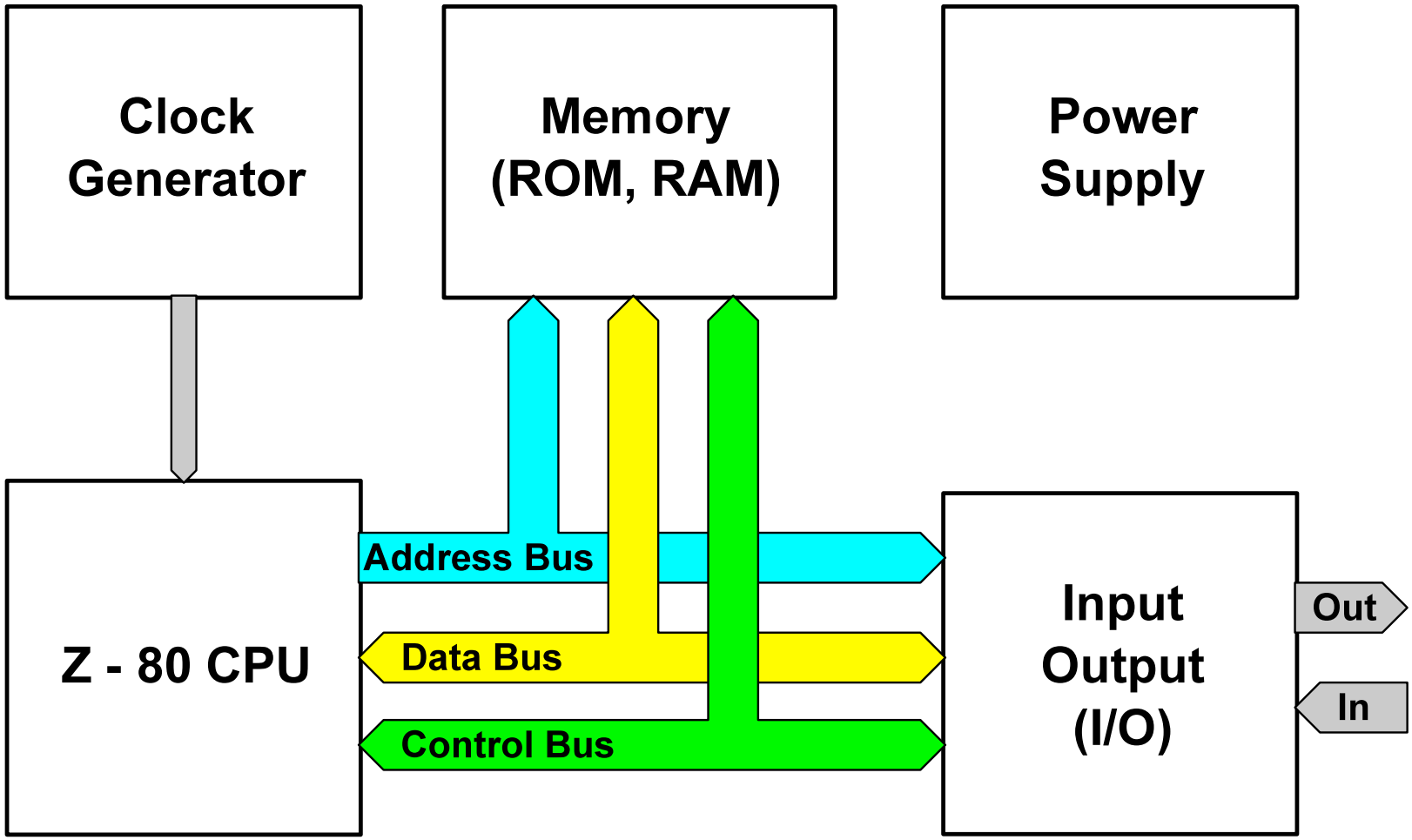
Addressing Modes

- **Immediate**
- **Immediate Extended**
- **Modified Page Zero Addressing (rst p)**
- **Relative Addressing**
  - Jump Relative (2 byte)
    - One Byte Op Code
    - 8-Bit Two's Complement Displacement (A+2)
- **Extended Addressing**
  - Absolute jump
    - One byte opcode
    - 2 byte address
- **Indexed Addressing**
  - (Index Register + Displacement) (IX+d)
  - 2 byte opcode
  - 1 byte displacement

Addressing Modes(cont.)

- **Register Addressing**
  - LD C,B
- **Implied Addressing**
  - Op Code implies other operand(s)
  - ADD E
- **Register Indirect Addressing**
  - 16-bit CPU register pair as pointer (such as HL)
  - ADD (HL)
- **Bit Addressing**
  - set, reset, and test instructions.
  - SET 3,A
  - RES 7,B

Minimal Configuration of a Z80 Microcomputer

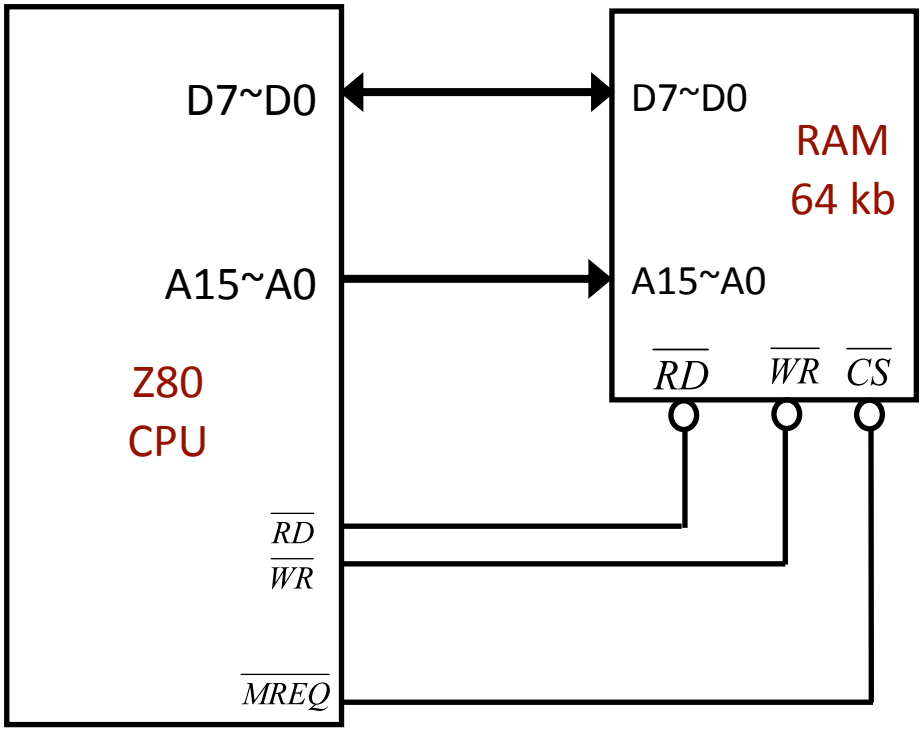


Z80 Memory connection

- CPU 16 bit address bus → 64 k memory(max)
- CPU 8 bit data bus → 8 bit data width
- Generally should be connected
  - Data to data
  - Address to address
  - Wr to wr
  - Rd to rd
  - Mreq to cs

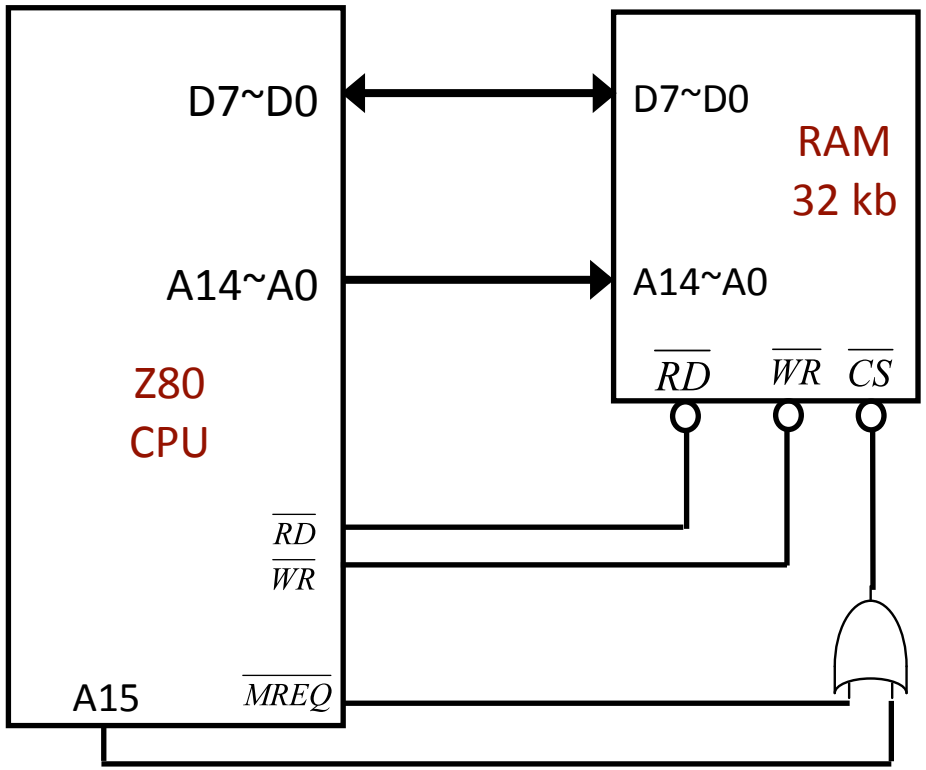
Memory connection (cont.)

- If only one RAM chip Full size (64 kb capacity)



Memory connection (cont.)

- If RAM capacity was 32 kb
- A15 composed with MREQ
- RAM area is from 0000h to 7FFFh

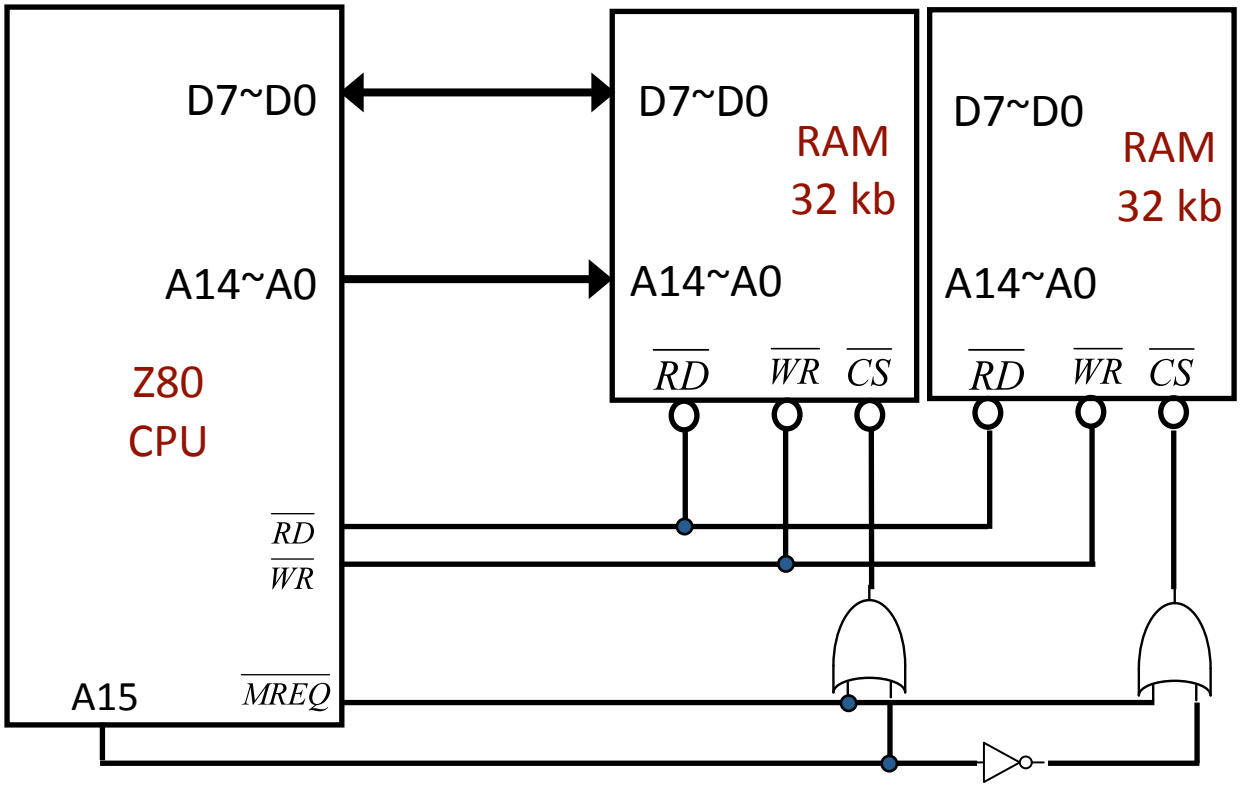


Memory connection (cont.)

- There is two 32 kb RAM
- Problem: Bus Conflict. The two memory chips will provide data at the same time when microprocessor performs a memory read.
- Solution: Use address line A15 as an “arbiter”. If A15 outputs a logic “1” the upper memory is enabled (and the lower memory is disabled) and vice-versa.

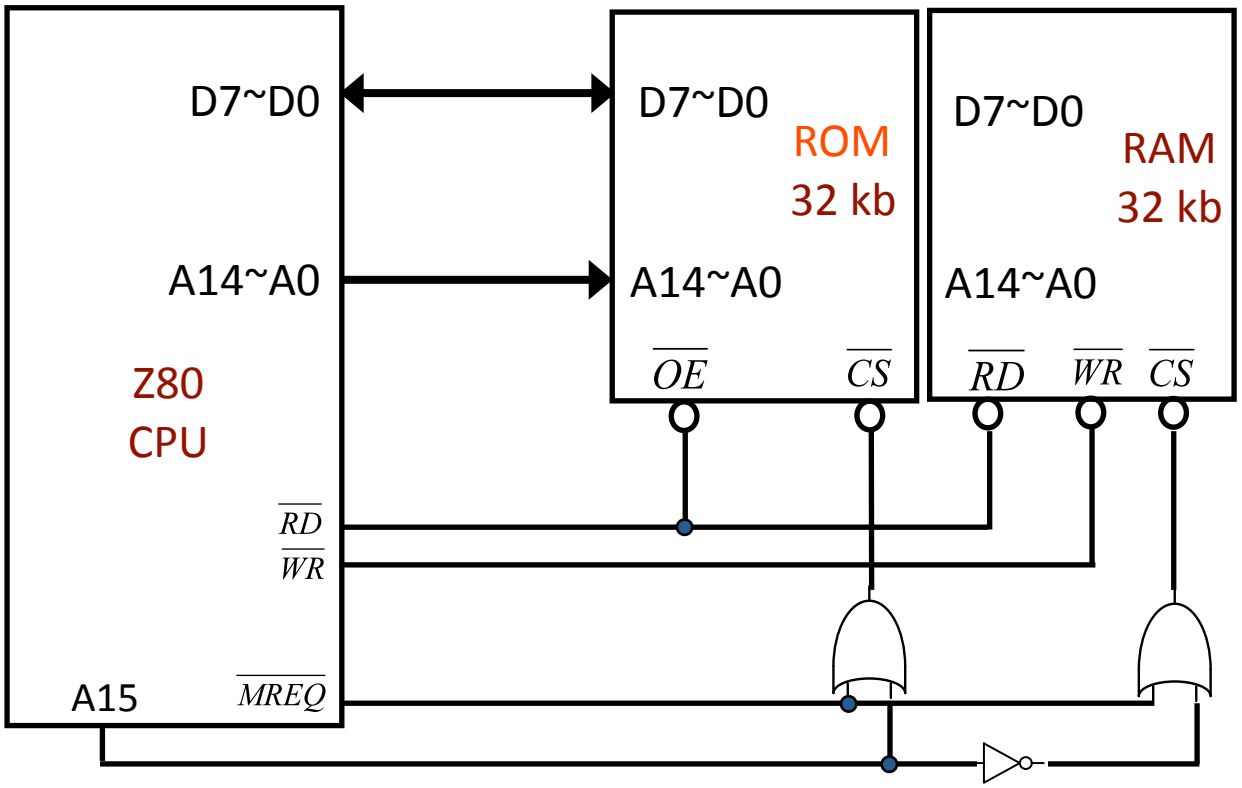
Memory connection (cont.)

- There is two 32 kb RAM
- A15 applied to select one RAM chip
- Two RAM area is from 0000h to 7FFFh (RAM1)
  and 8000h to FFFFh (RAM1)



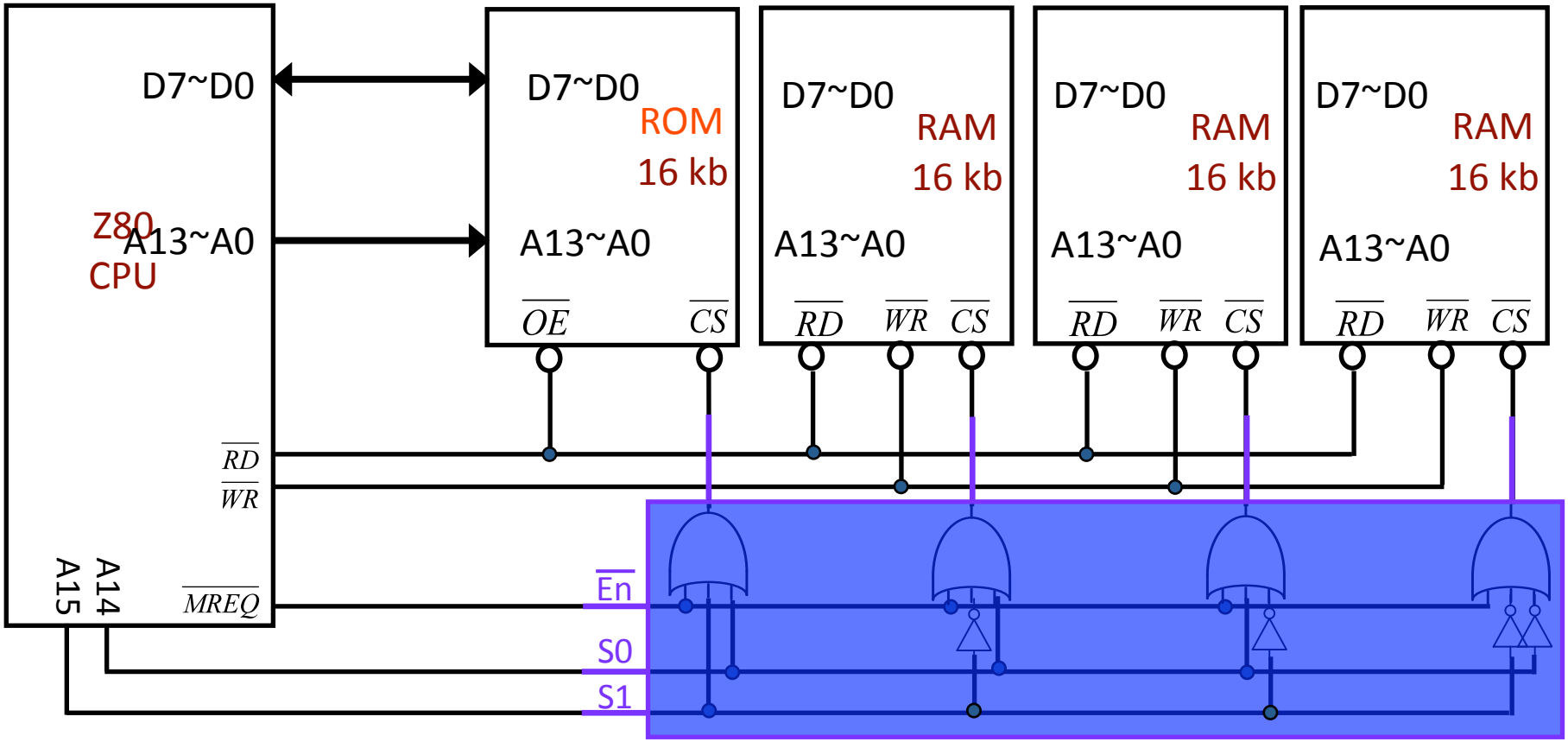
Memory connection (cont.)

- 32 kb ROM and 32 kb RAM
- ROM doesn't have wr signal



Memory connection (cont.)

There is 4 memory chip
A14 and A15 applied to chip selection



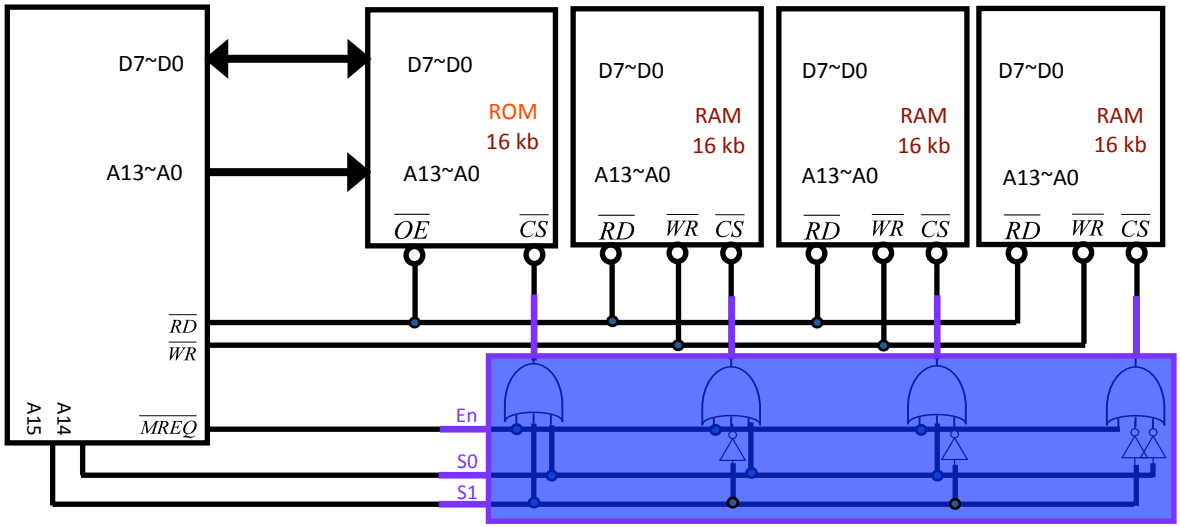
Address Bit Map

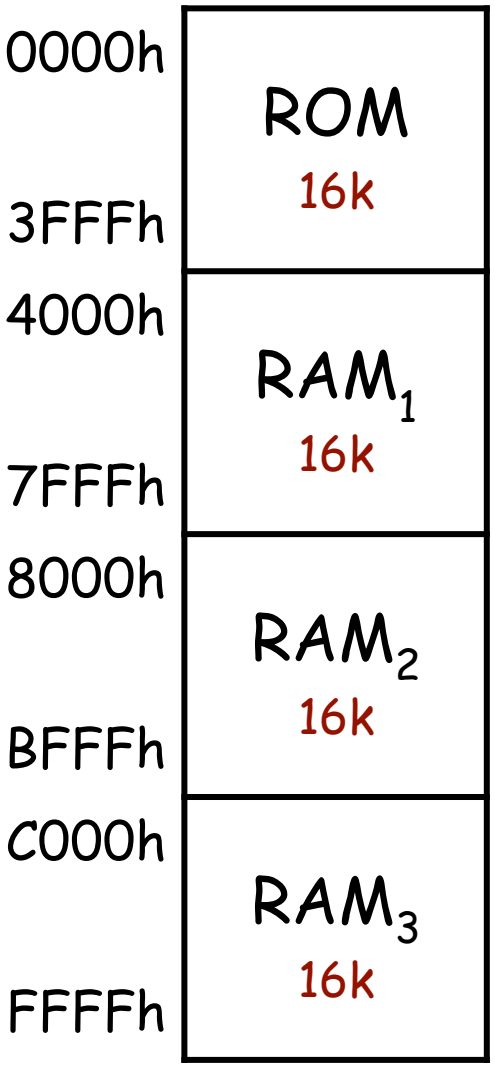
Selects chip Selects location within chips

A15 to A0 (HEX)	AA 11 54	AA 11 32	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0000h	00	00	0000	0000	0000	ROM
3FFFh	00	11	1111	1111	1111	
4000h	01	00	0000	0000	0000	RAM_1
7FFFh	01	11	1111	1111	1111	
8000h	10	00	0000	0000	0000	RAM_2
BFFFh	10	11	1111	1111	1111	
C000h	11	00	0000	0000	0000	RAM_3
FFFFh	11	11	1111	1111	1111	

Memory Map

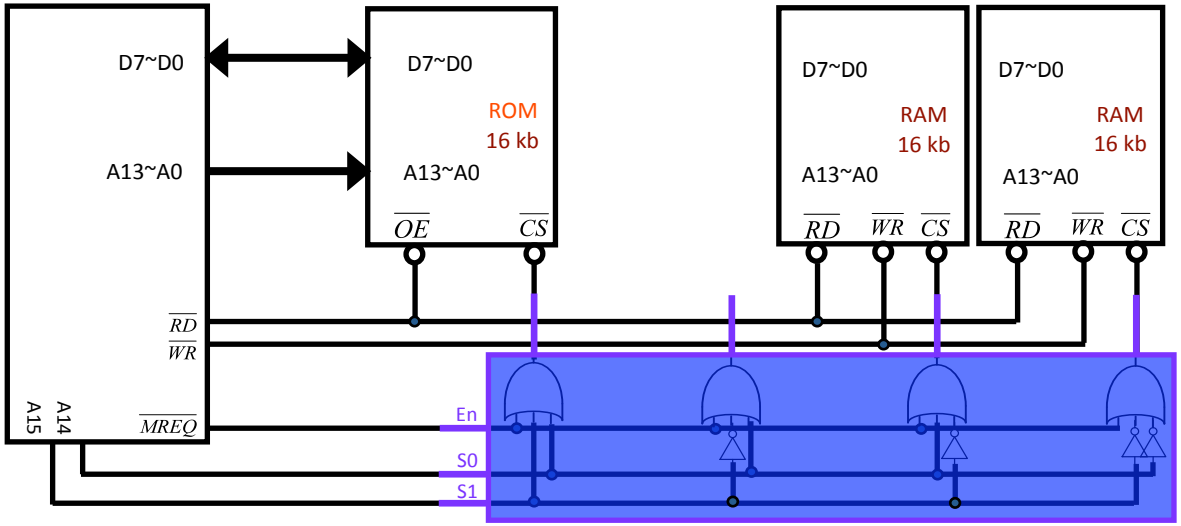
- Represents the memory type
- Address area of each memory chip
- Empty area

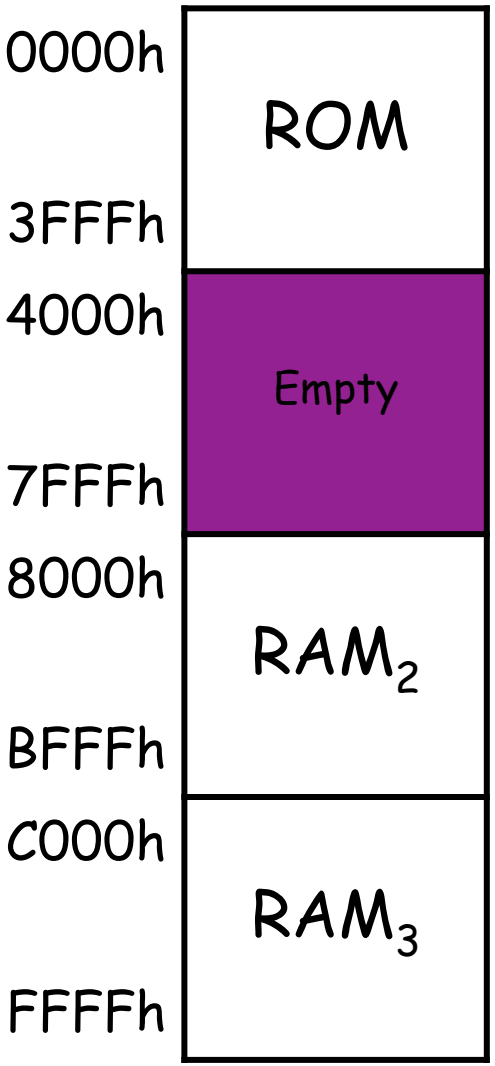




Memory Map

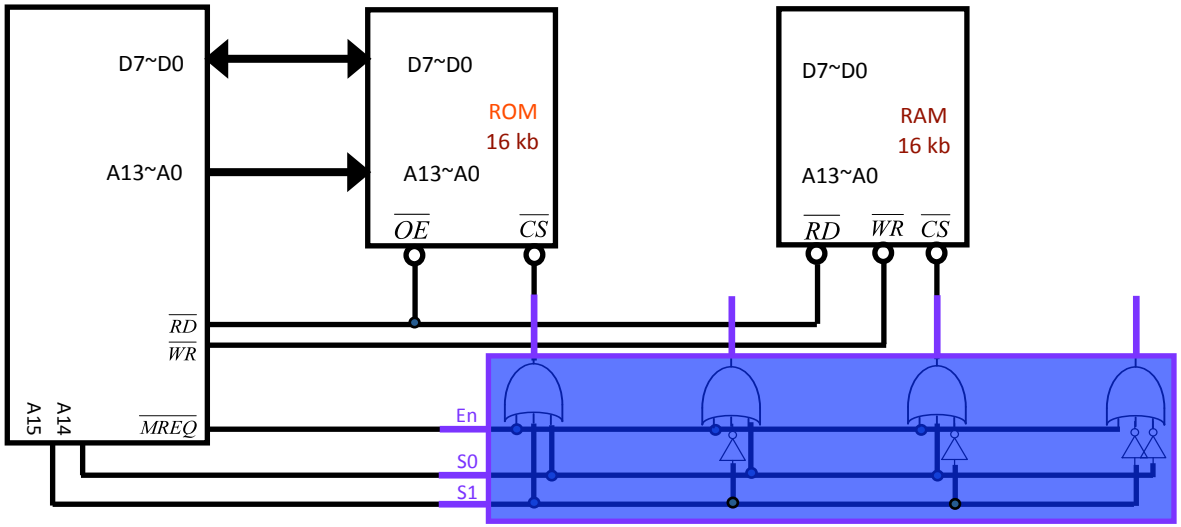
- Empty Area cann't write and read
- Read op. returns FFh value (usualy)
- Write op. cann't store any value on it

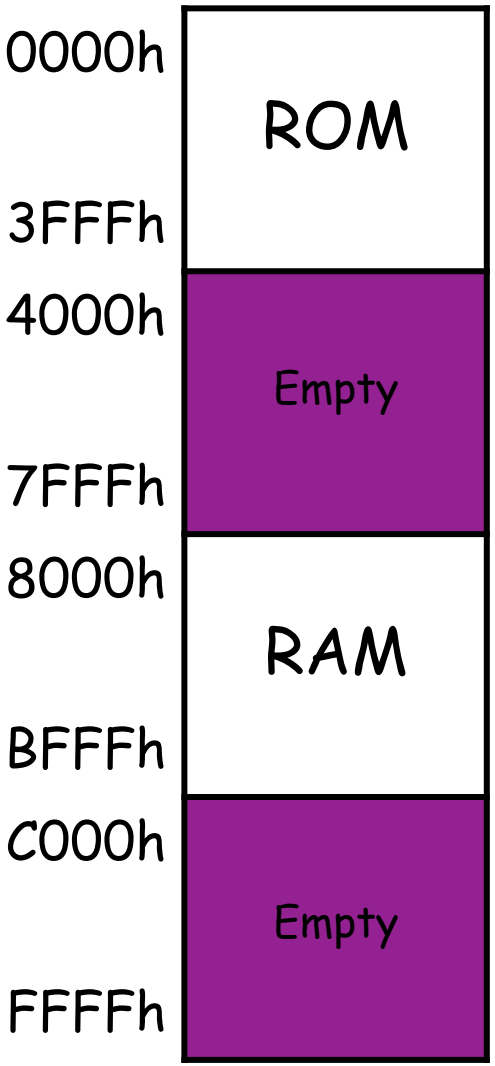




Memory Map

- Empty Area cann't write and read
- Read op. returns FFh value (usualy)
- Write op. cann't store any value on it





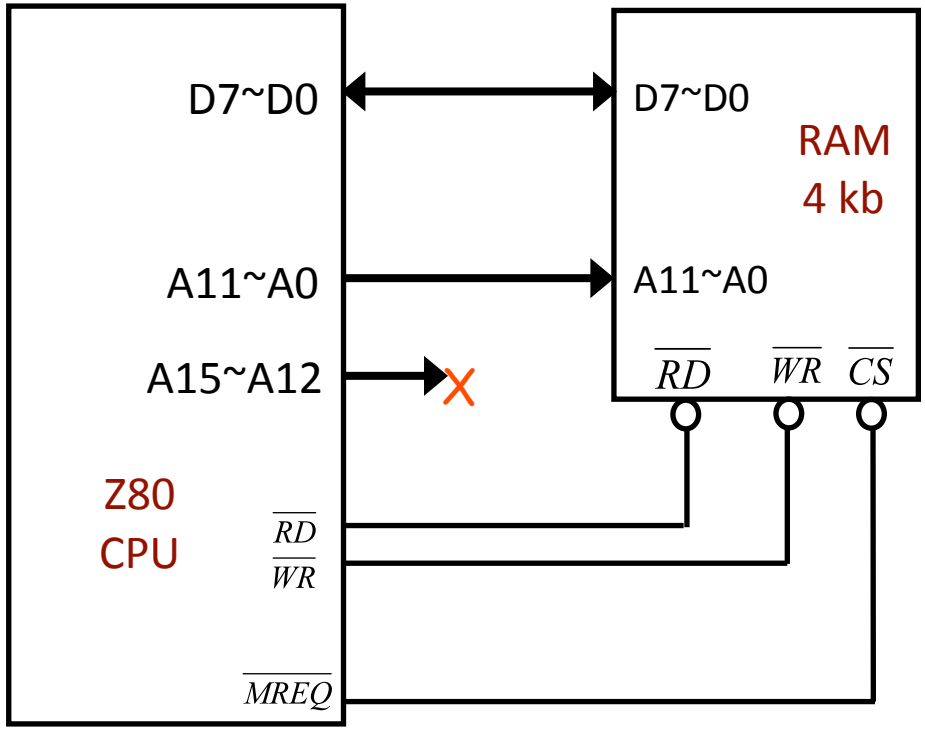
Full and Partial Decoding

- Full (exhaust) Decoding
  - All of the address lines are connected to any memory/device to perform selection
  - Absolute address : any memory location has one address

- Partial Decoding
  - When some of the address lines are connected the memory/device to perform selection
  - Using this type of decoding results into roll-over addresses (fold back or shading).
  - roll-over address : any memory location has more than one address

Partial Decoding

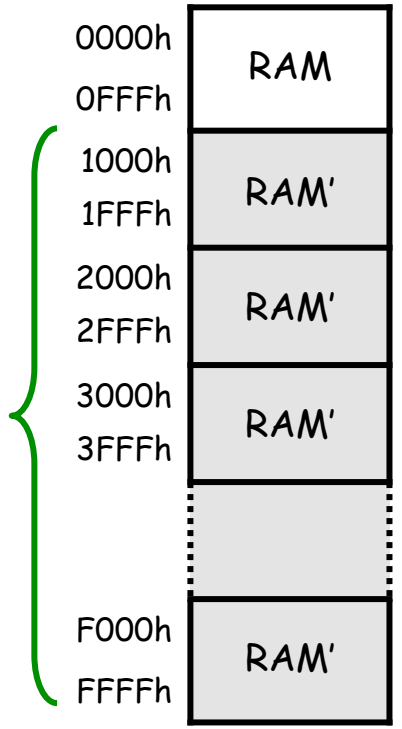
- A15~A12 has no connection
- Then doesn't play any role in addressing
- What is the Memory and Address Bit map?



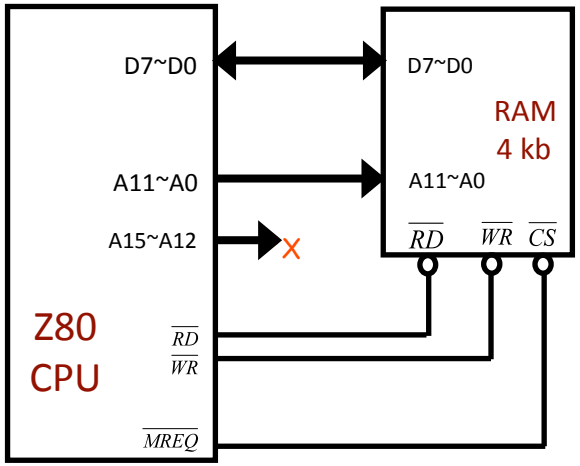
Partial Decoding

- Every memory location has more than one address
- For example first RAM location has addresses:
  - 0000h
  - 1000h
  - 2000h
  - 3000h
  - .....
  - .....
  - F000h

Roll-over Address

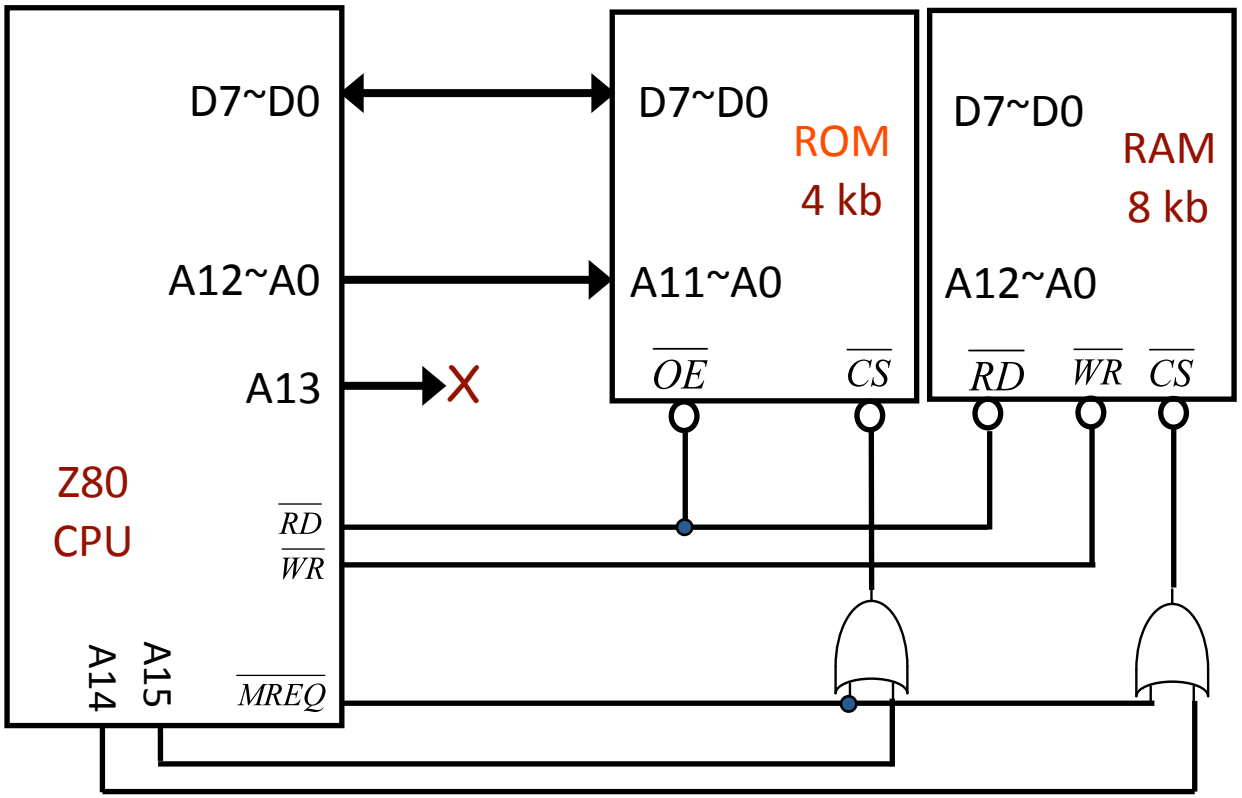


A15 to A0 (HEX)	AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
X000h	xxxx	0000	0000	0000	RAM
XFFFh	xxxx	1111	1111	1111	



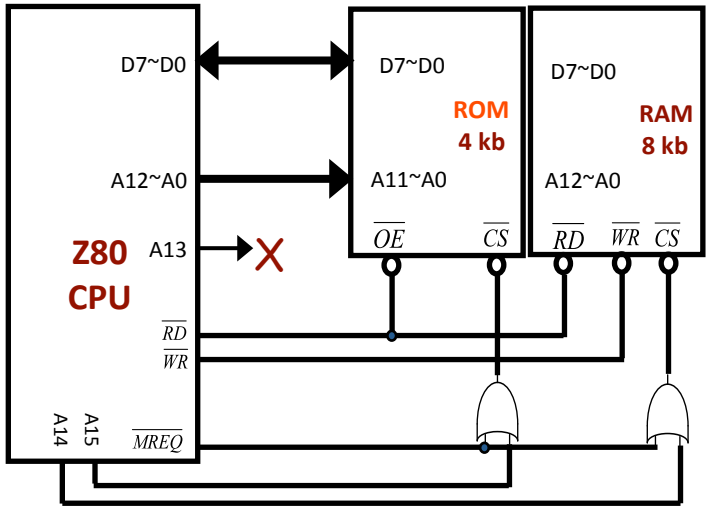
Partial Decoding

- A12 only connected to RAM
- A13 has no connection
- What is the memory map?



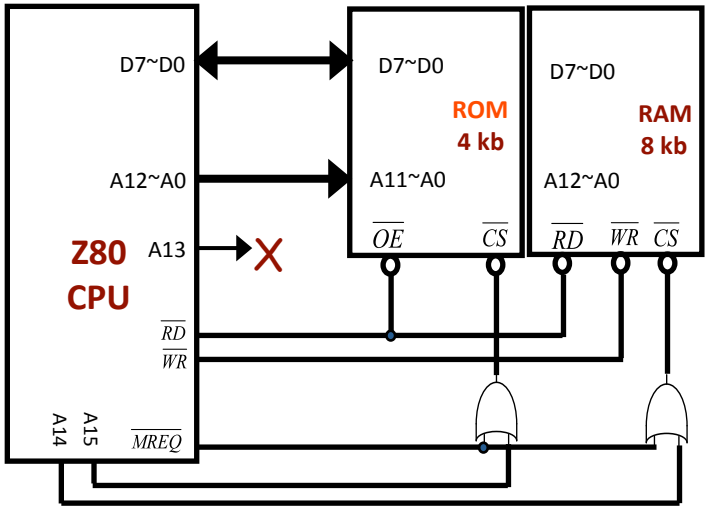
Partial Decoding

- 8 roll-over address for ROM
- 4 roll-over address for RAM

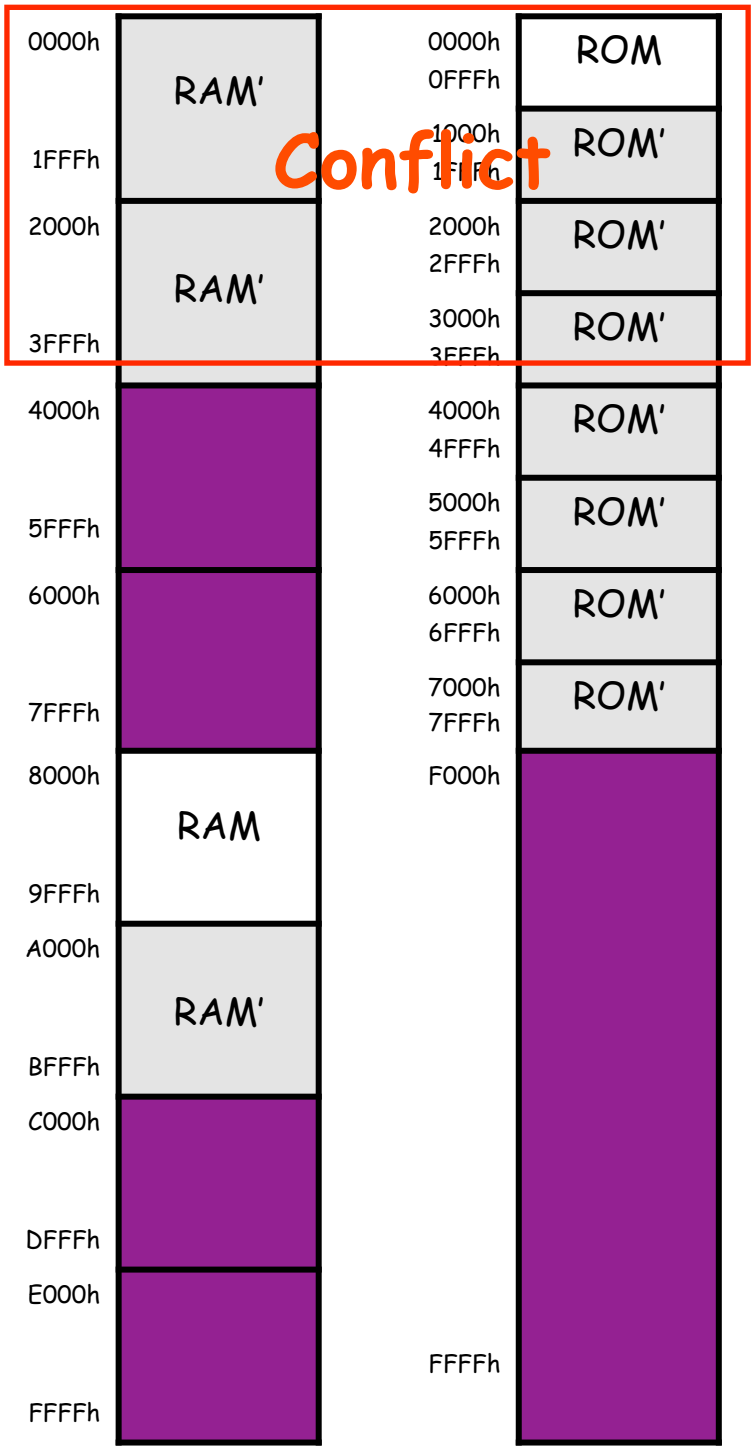


AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0xxx	0000	0000	0000	ROM
0xxx	1111	1111	1111	
X0x0	0000	0000	0000	RAM
X0x1	1111	1111	1111	

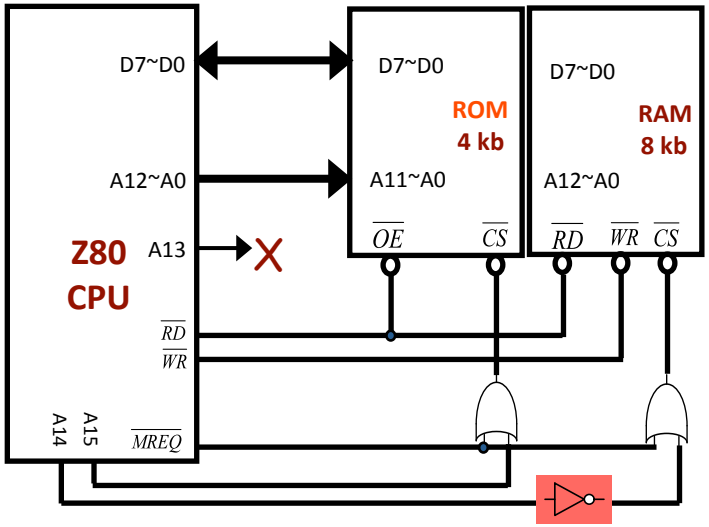
Partial Decoding



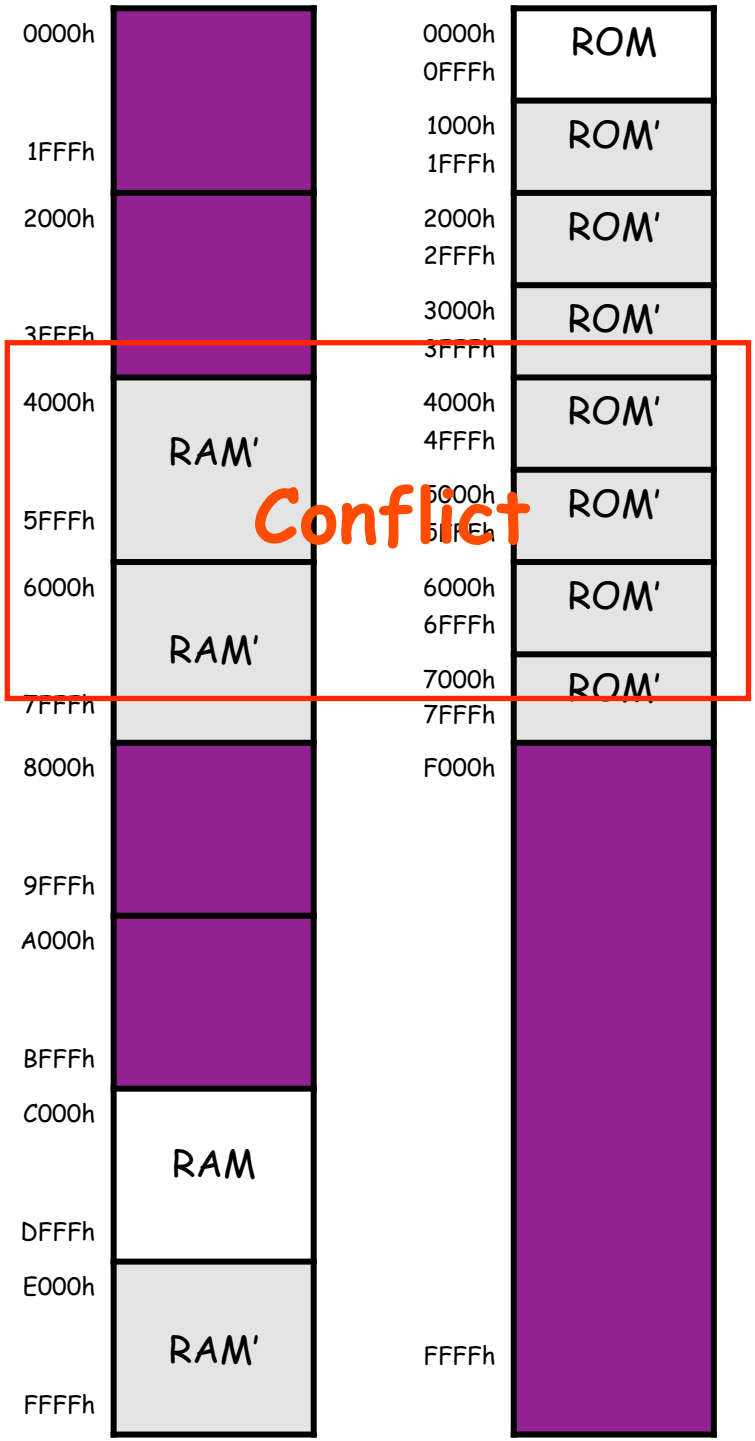
AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0xxx	0000	0000	0000	4k ROM
0xxx	1111	1111	1111	
X0x0	0000	0000	0000	8k RAM
X0x1	1111	1111	1111	



Partial Decoding

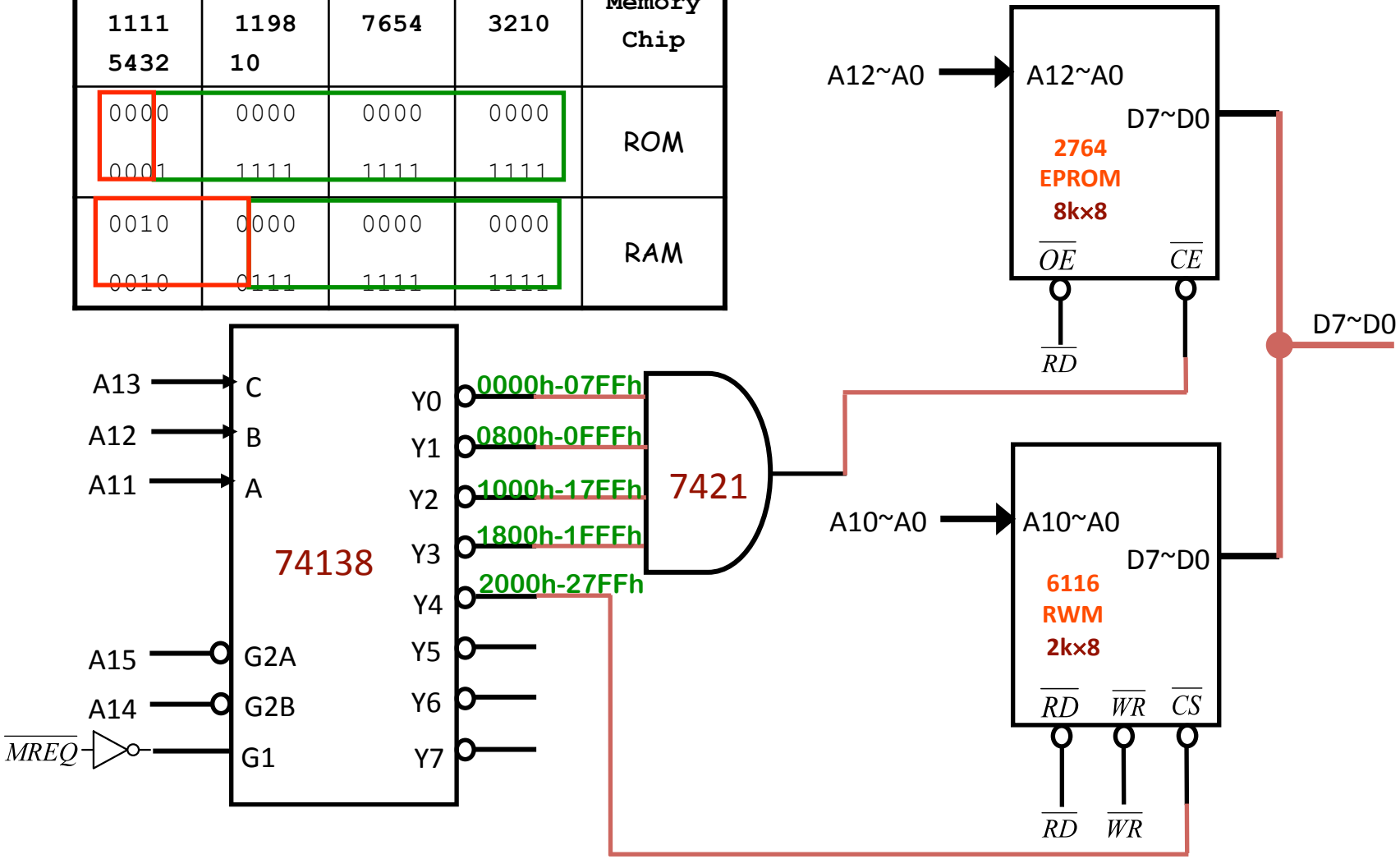


AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0xxx	0000	0000	0000	4k ROM
0xxx	1111	1111	1111	
X1x0	0000	0000	0000	8k RAM
X1x1	1111	1111	1111	



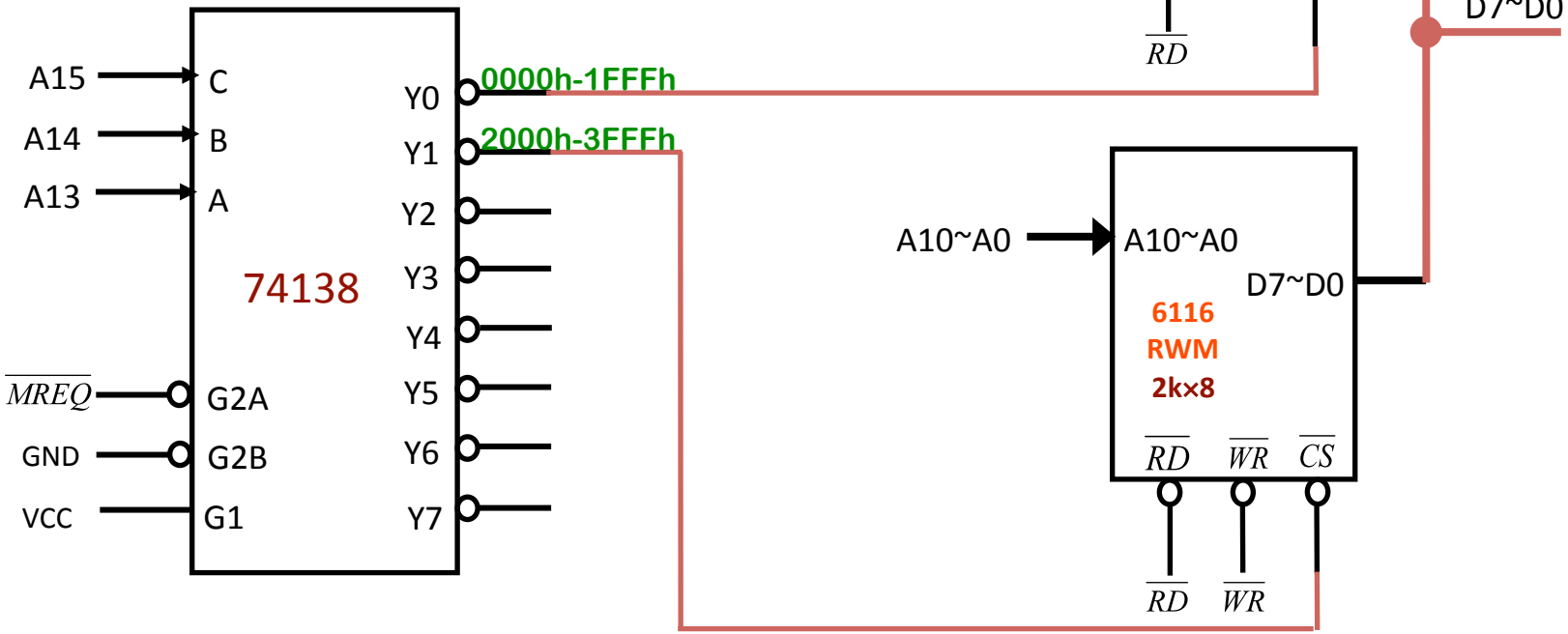
Full (exhaustive) decoding

AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0000 0001	0000 1111	0000 1111	0000 1111	ROM
0010 0010	0000 0111	0000 1111	0000 1111	RAM

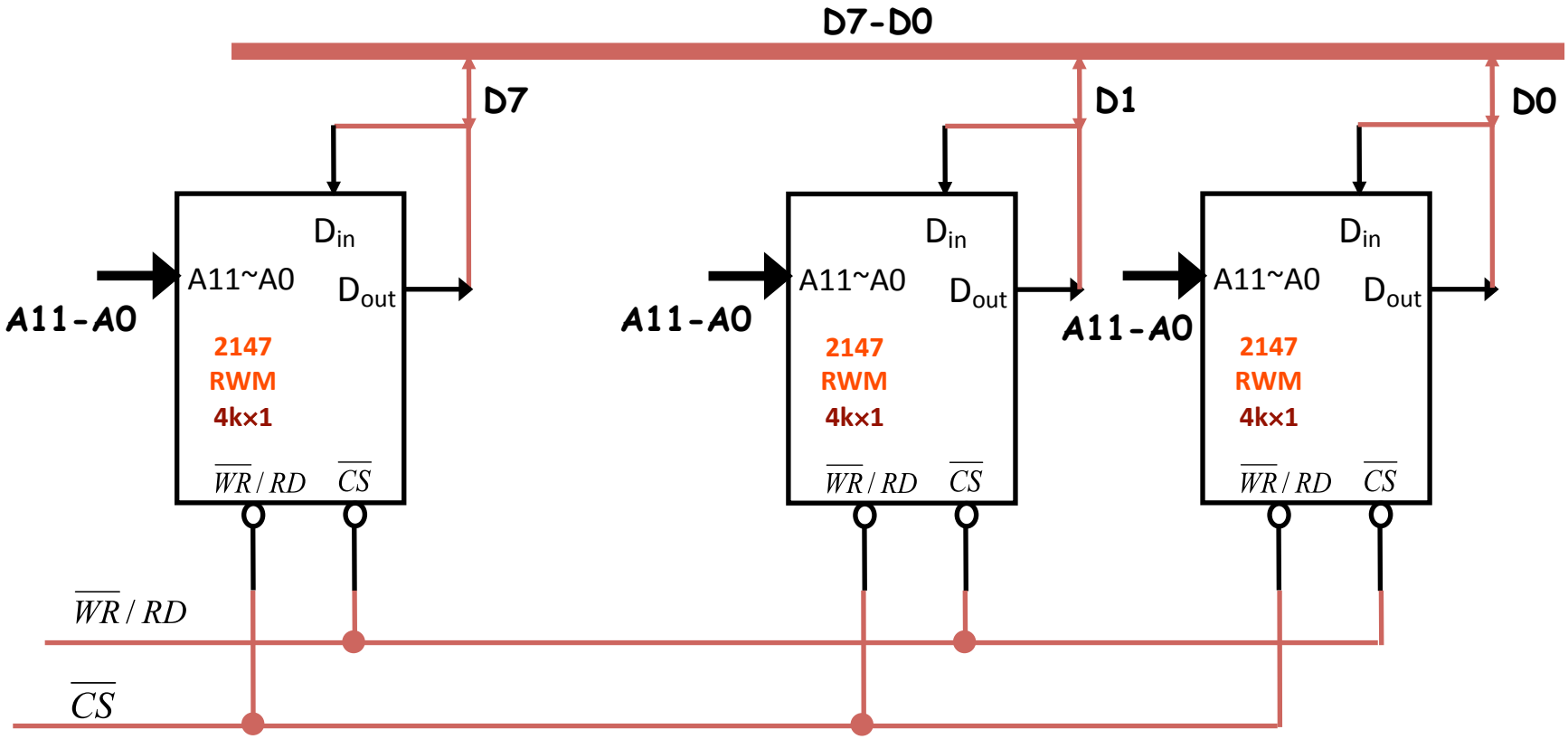


Partial decoding

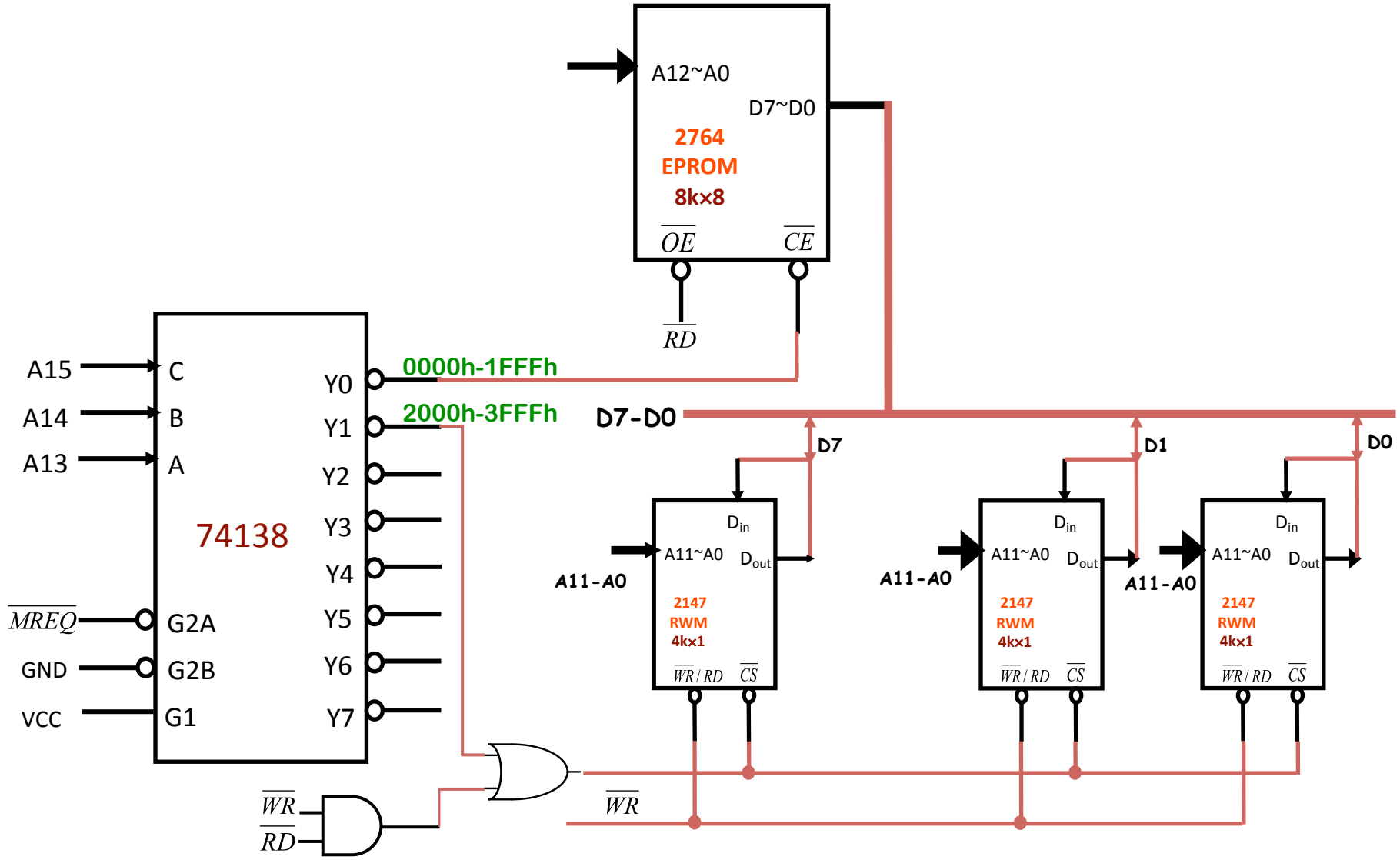
AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0000 0001	0000 1111	0000 1111	0000 1111	ROM
001x 001x	x000 x111	0000 1111	0000 1111	RAM



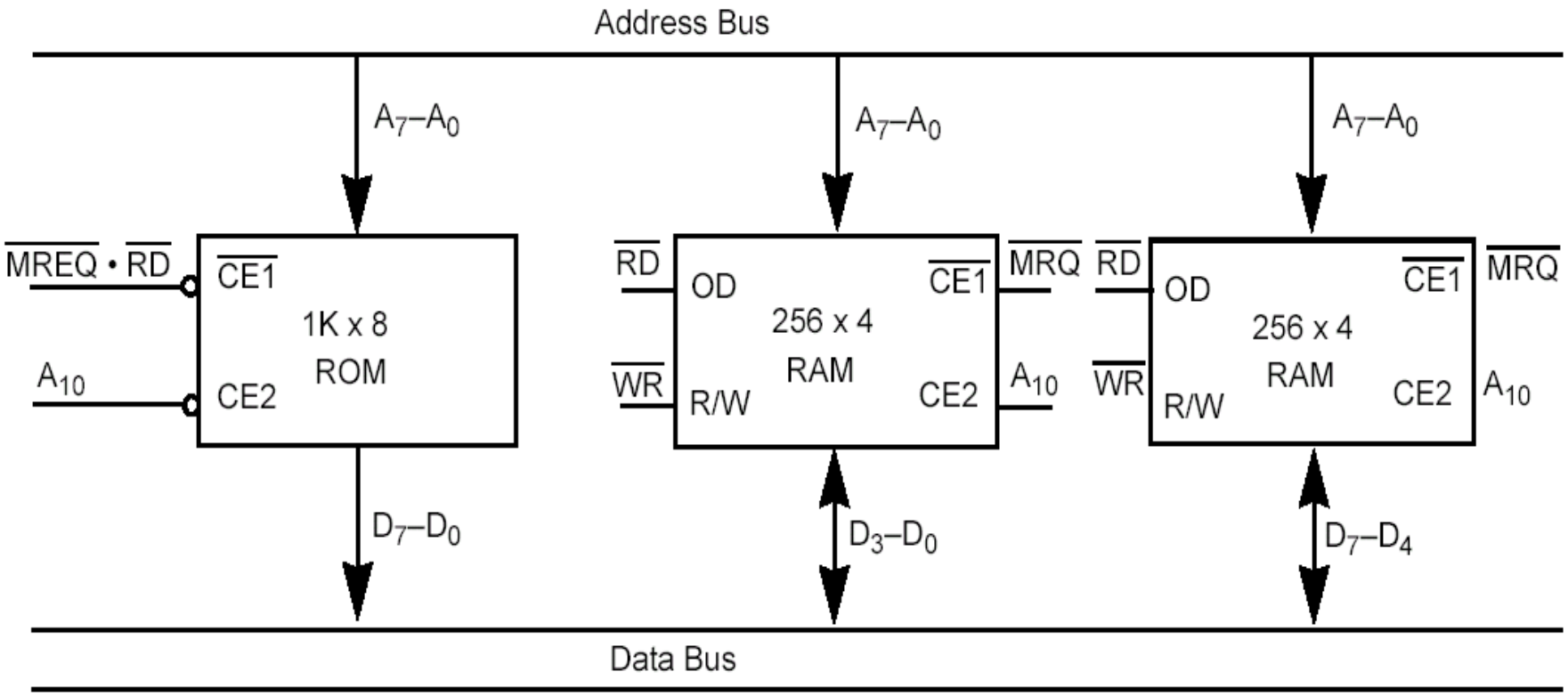
1 Bit Memory With Separated I/O



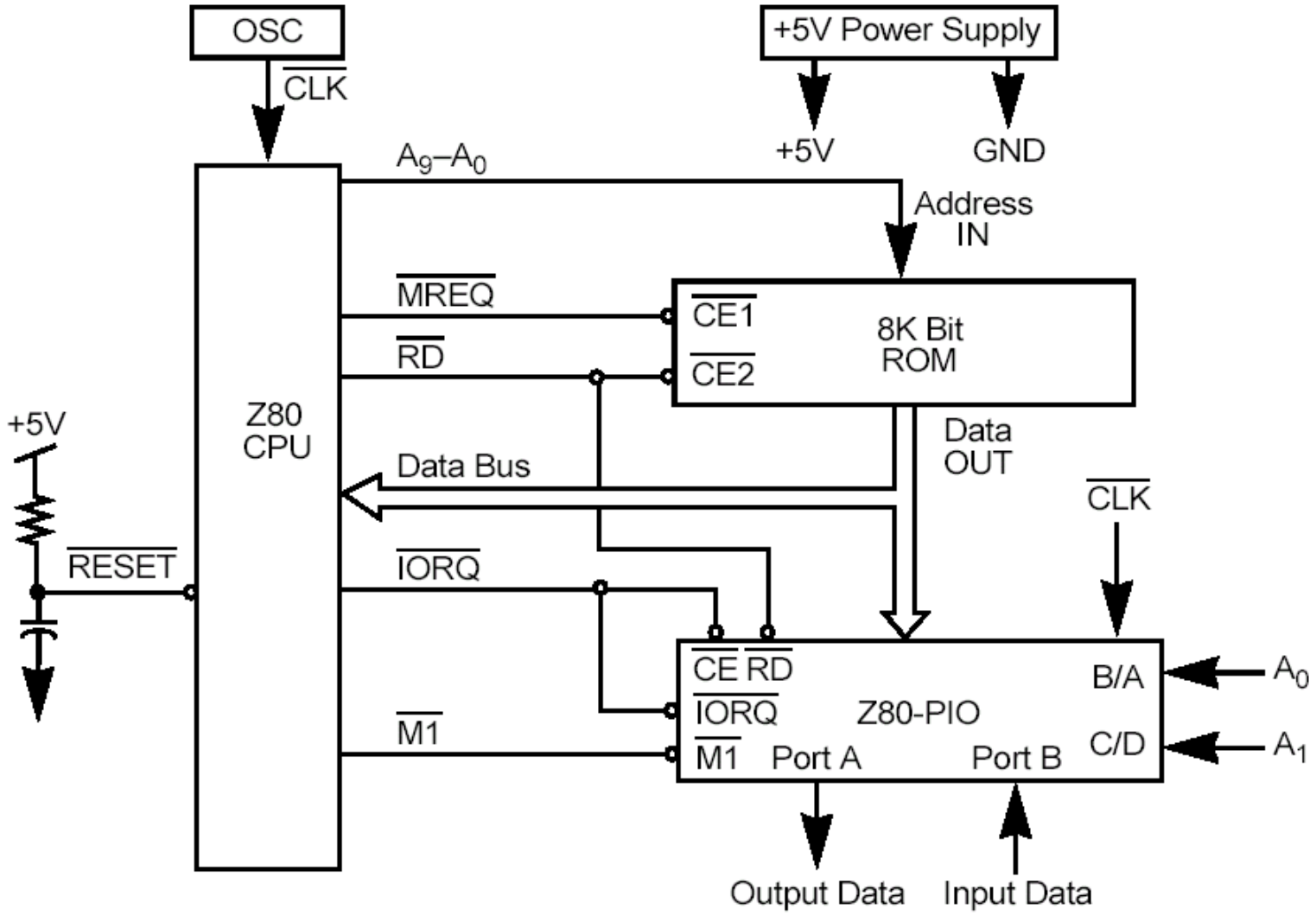
What is the memory(addr. bit) map



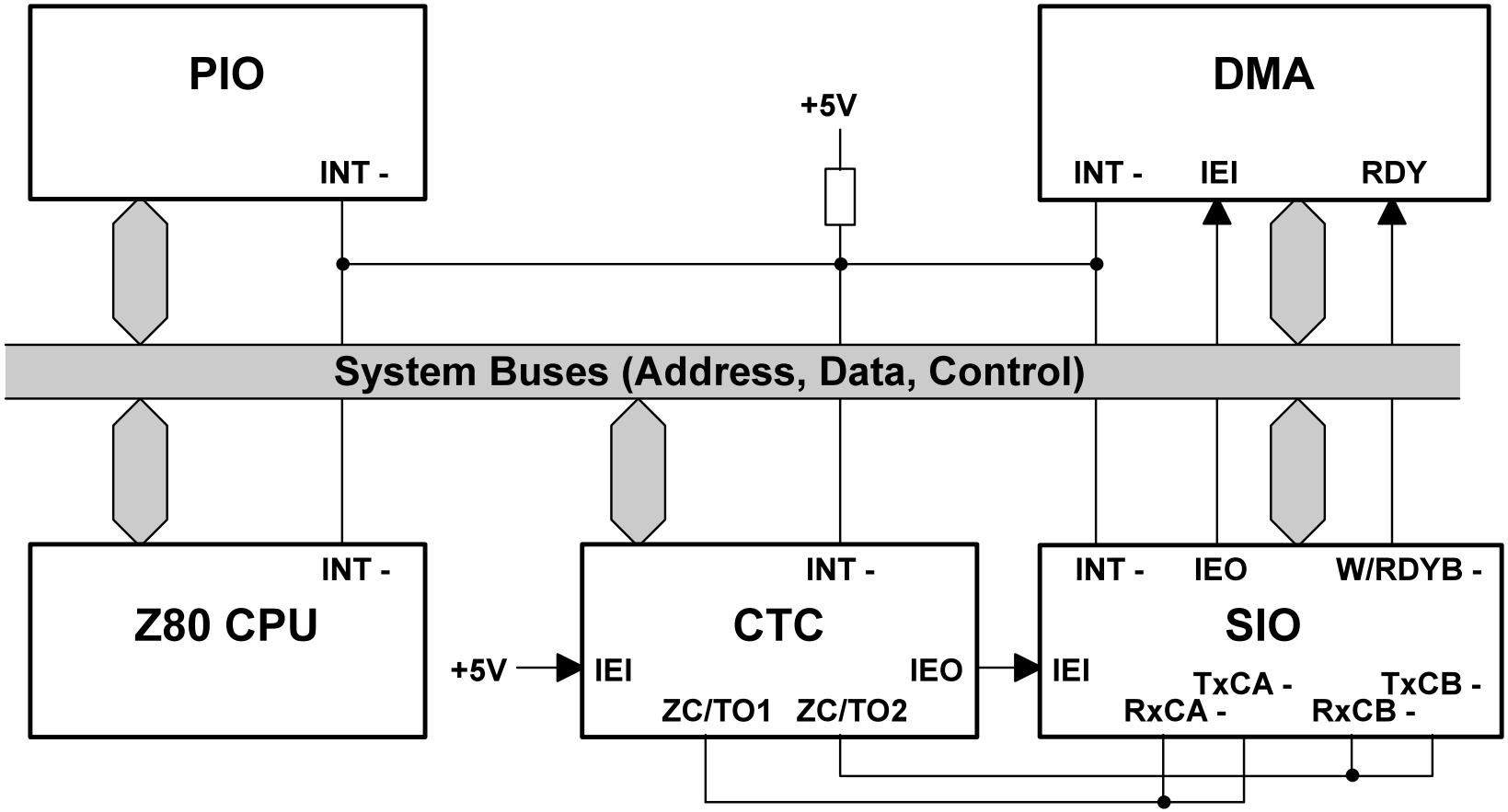
Adding RAM & ROM



Minimum Z80 Computer System



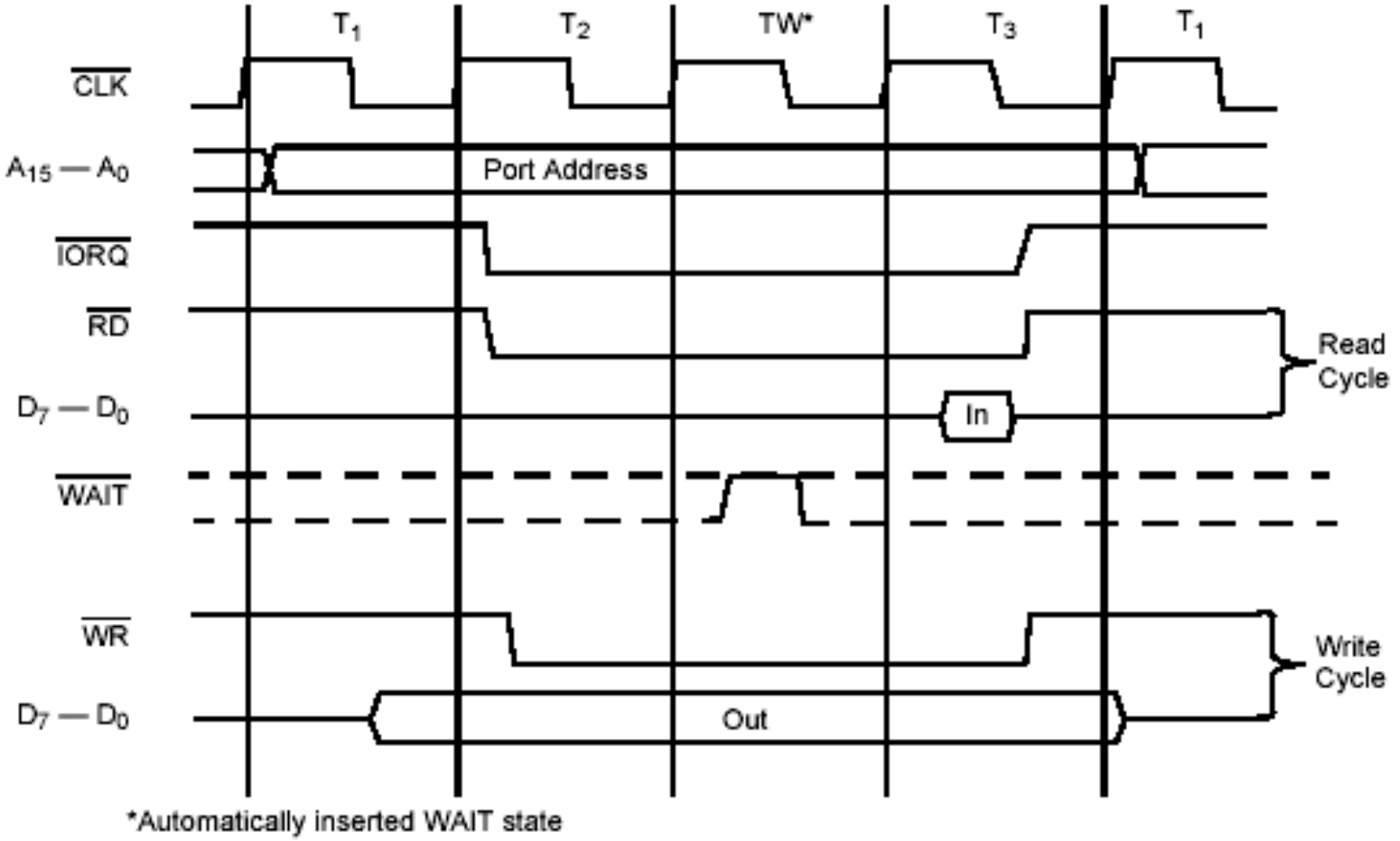
Z80-µP-Family (Typical Environment)



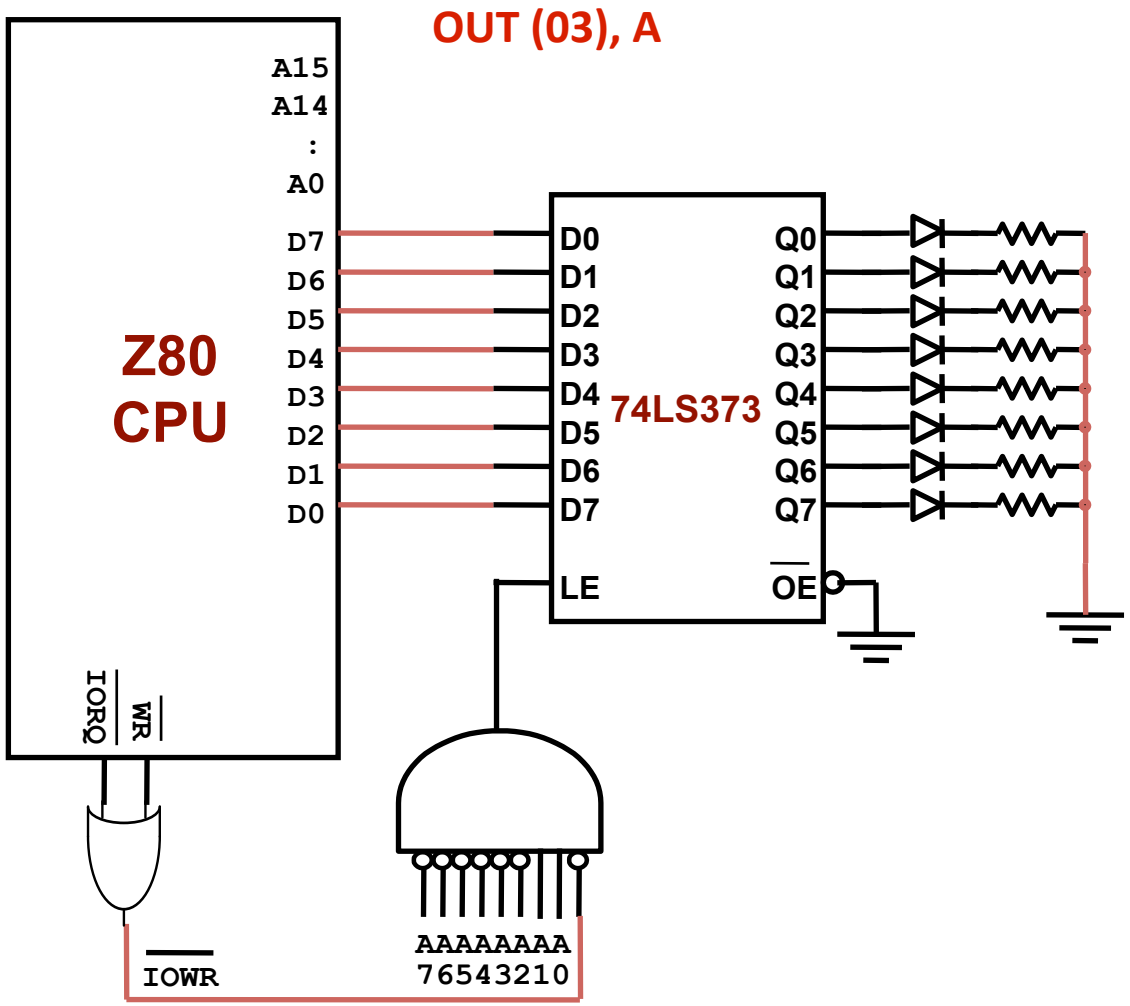
Z80 Input Output

- Z80 at most could have 256 input port and 256 output
- 8 bit port address is placed on A7-A0 pin to select the I/O device
- **OUT (n), A**
  - n is 8 bit port address
  - Content of A is data
- **OUT (C), r**
  - Content of C is a port address
  - r is a data register
- **IN A, (n)**
  - n is 8 bit port address
  - Data is transfered to A
- **IN r (C)**
  - Content of Reg C is a port address
  - Input data is transfered to r (data reg)

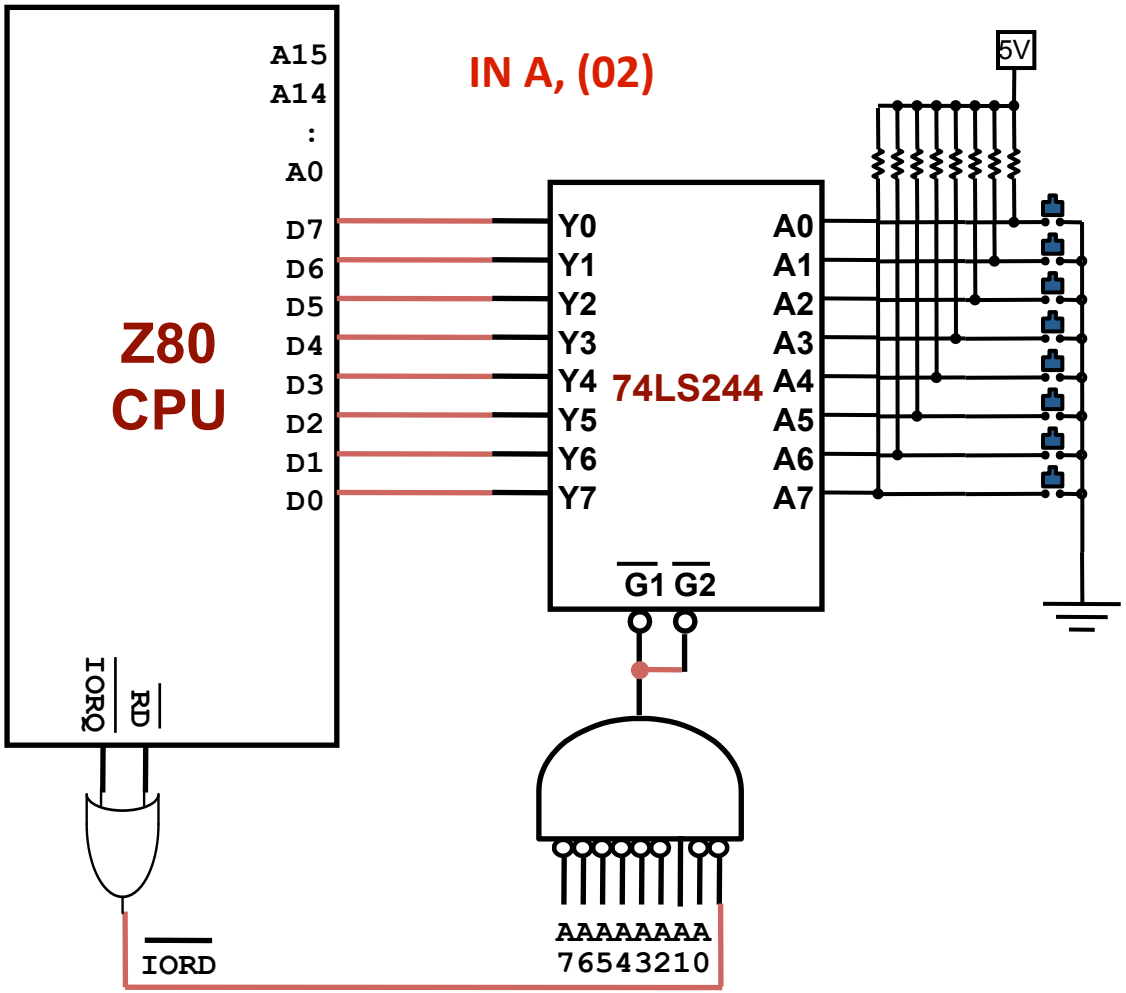
Remember IO read/write cycle



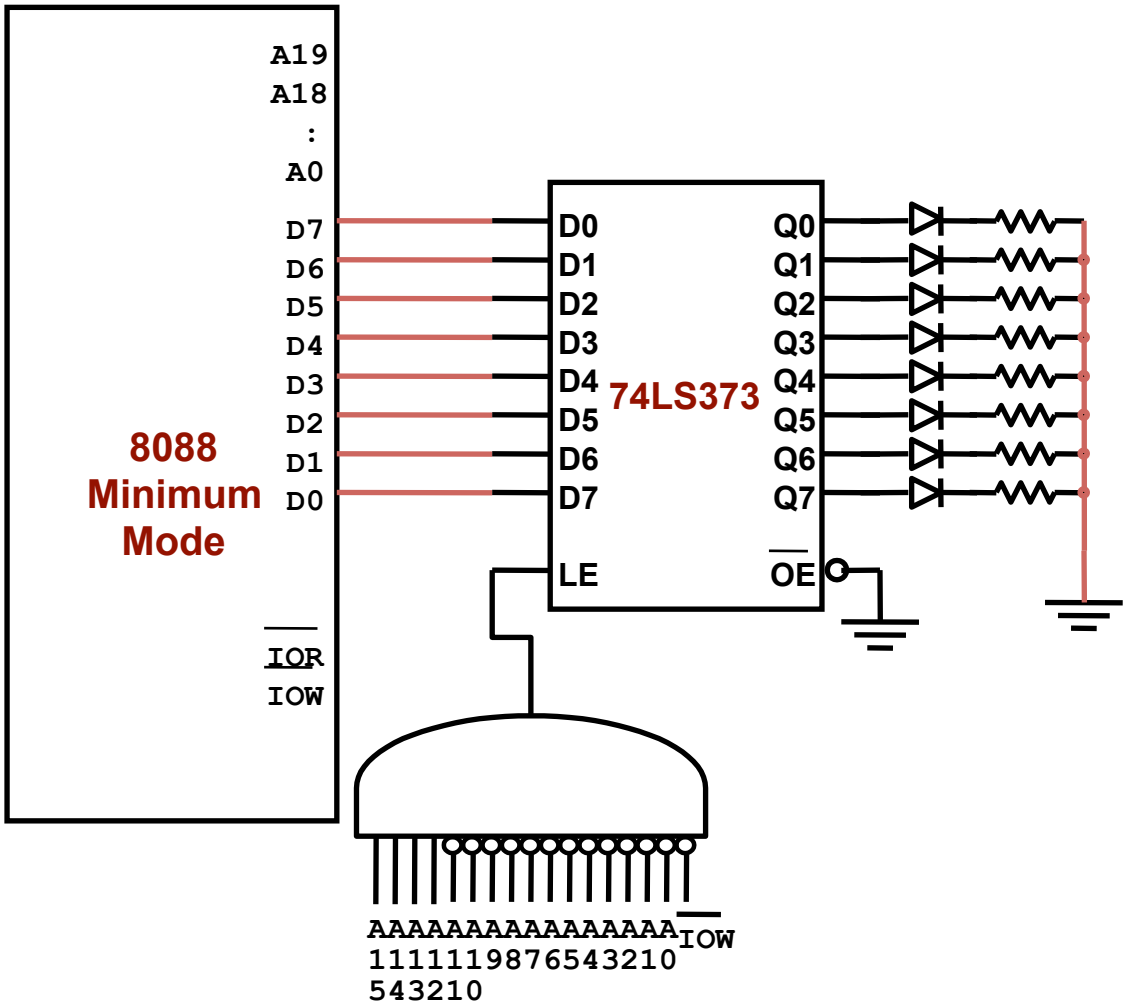
Z80 and simple output port



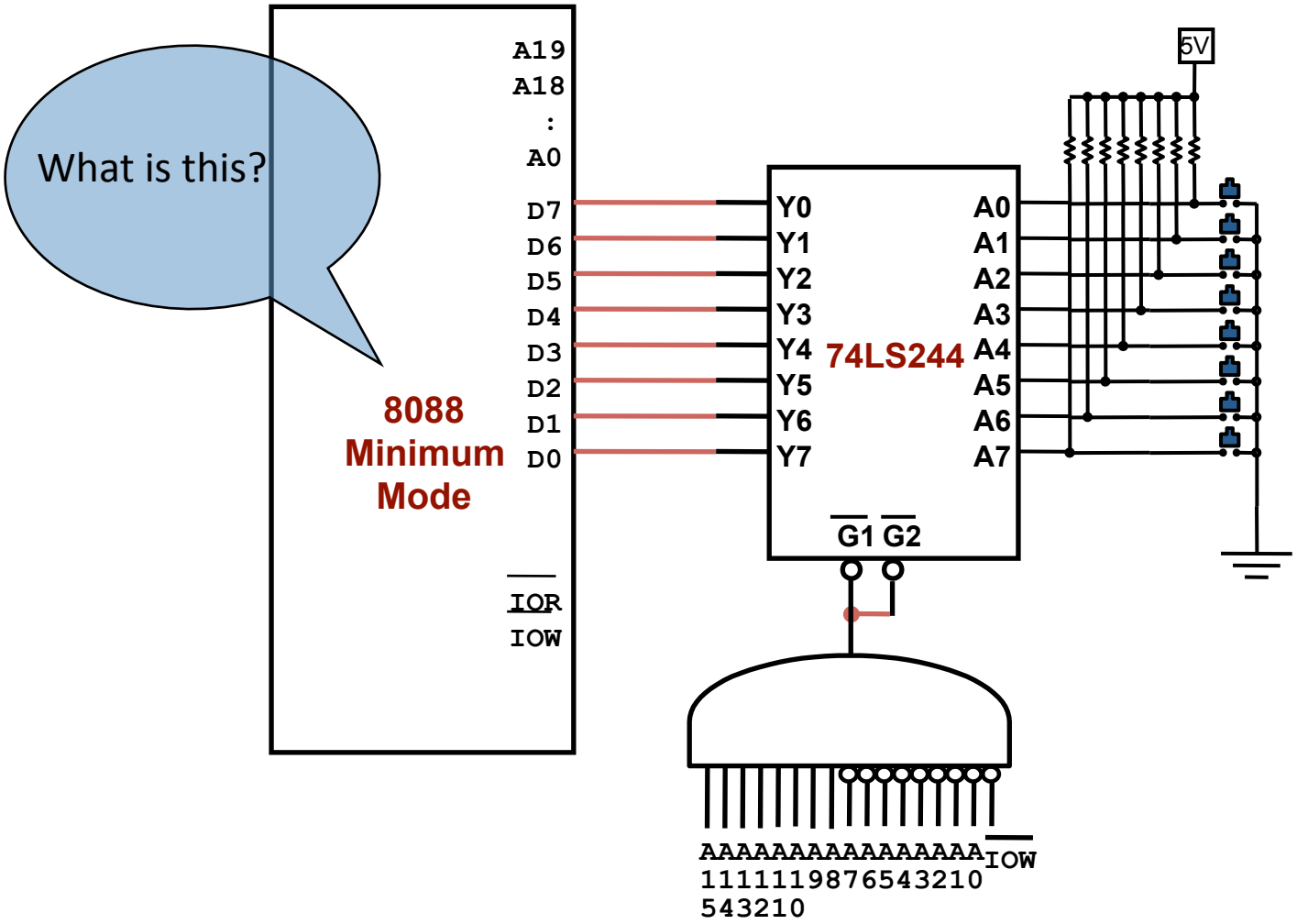
Z80 and simple input port



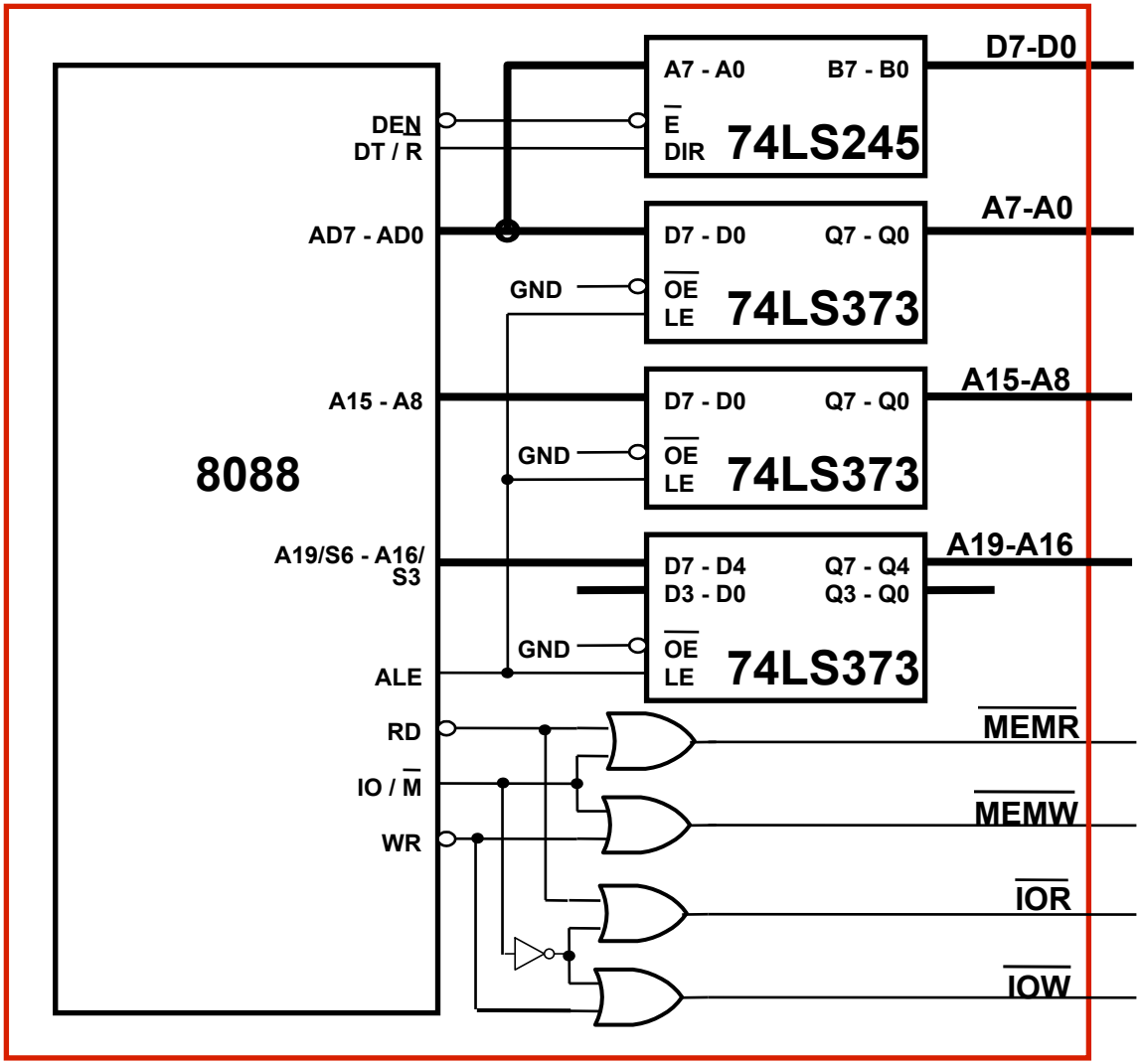
8088 and simple output port



8088 and simple input port

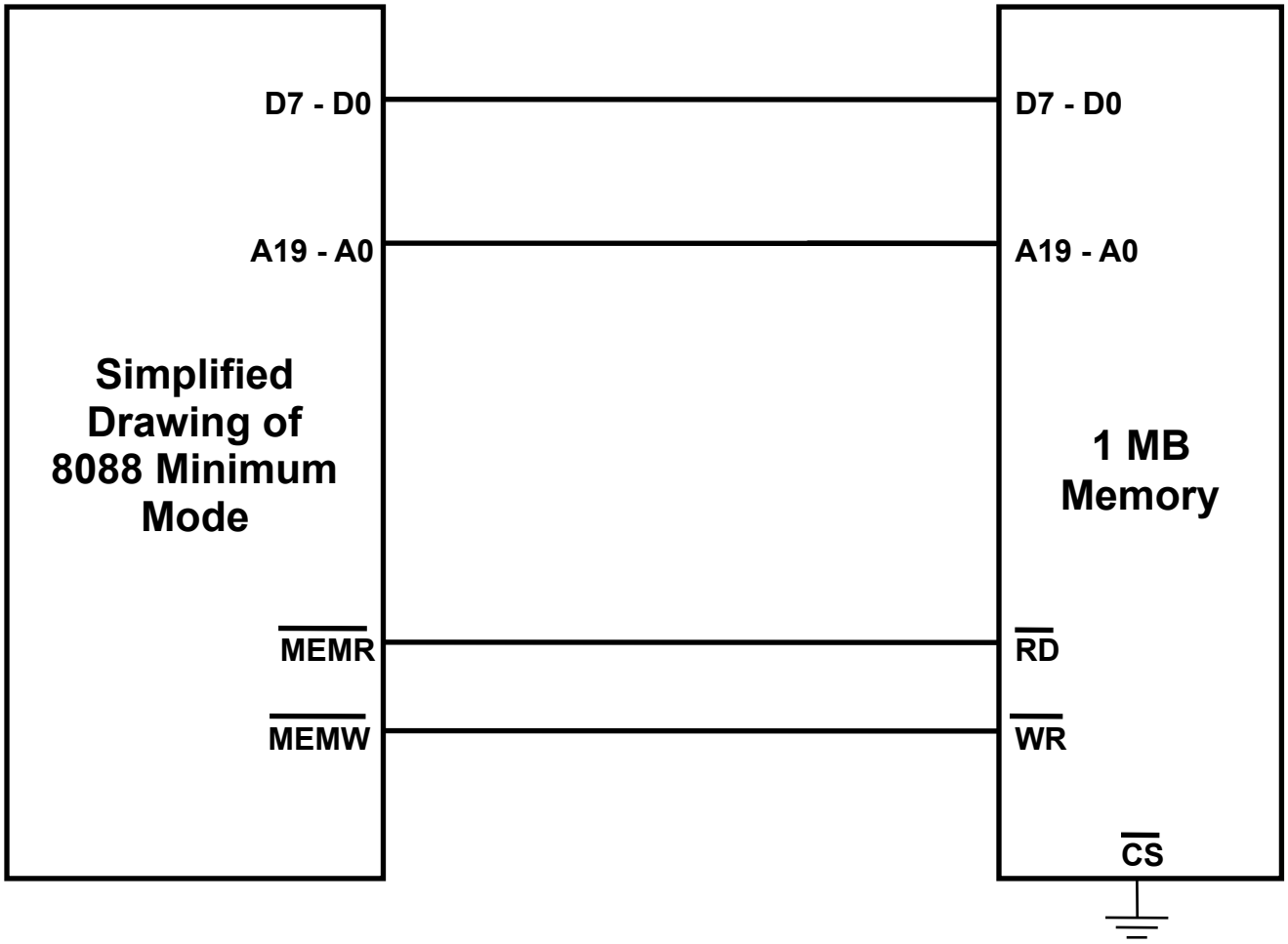


Simplified Drawing of 8088 Minimum Mode



Minimum Mode

2^{20} bytes or 1MB memory



What are the memory locations of a 1MB (2^{20} bytes) Memory?

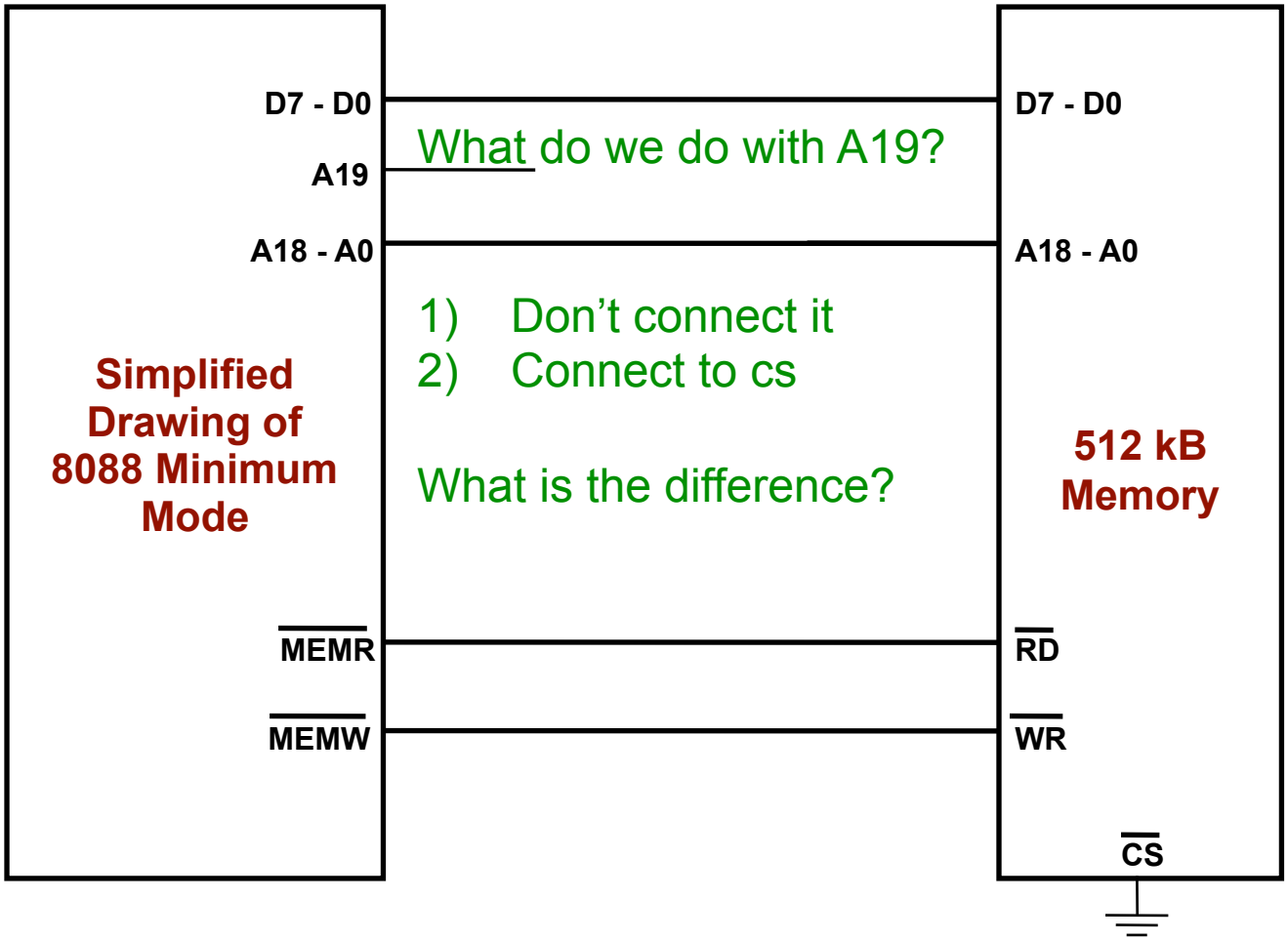
A19 to A0 (HEX)	AAAA 1111 9876	AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210
00000	0000	0000	0000	0000	0000
FFFFF	1111	1111	1111	1111	1111

Example: 34FD0

0011 0100 11111 1101 0000

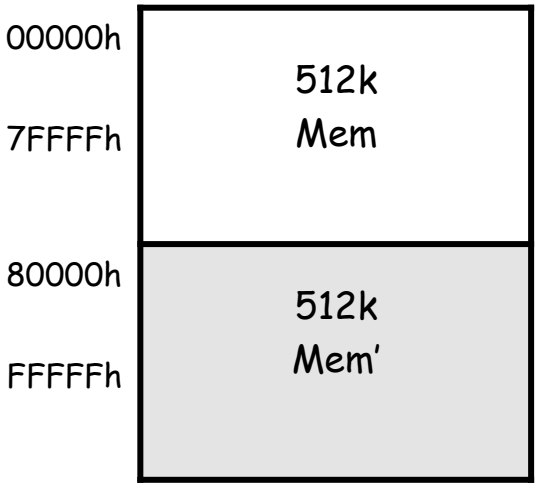
Minimum Mode

512 kB memory

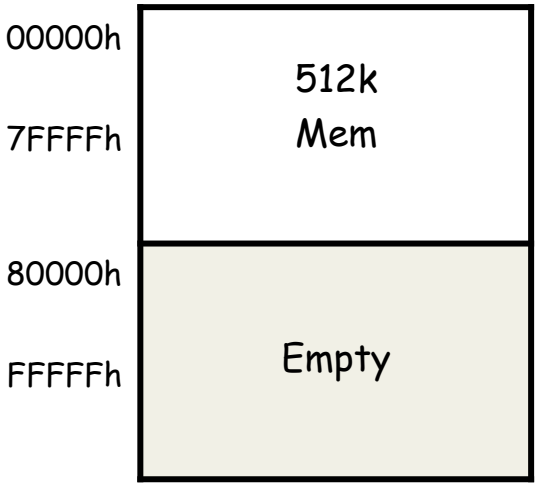


512 kB Memory Map

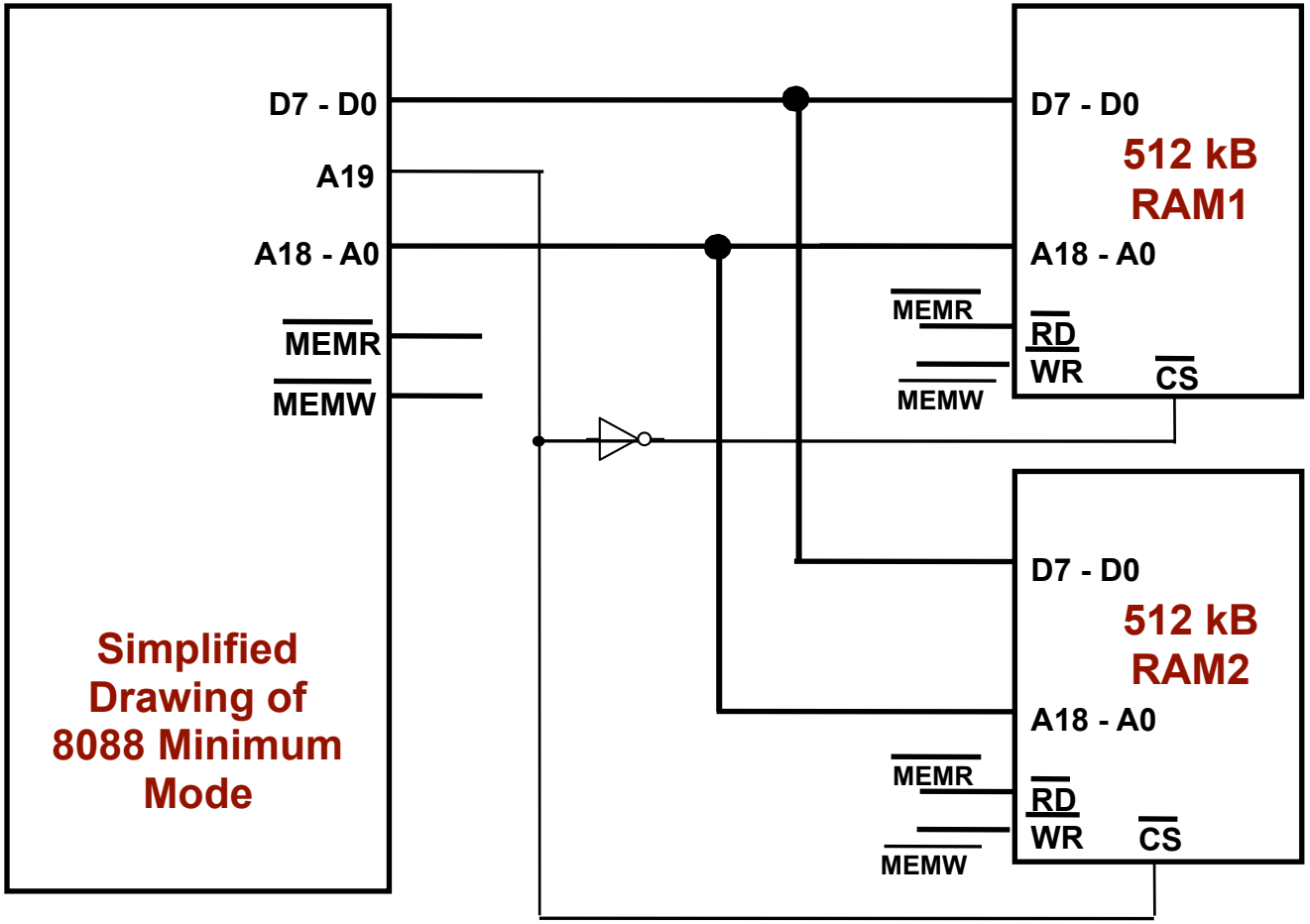
- Don’t connect it
  - A19 is not connected to the memory so even if the 8088 microprocessor outputs a logic “1”,the memory cannot “see” it.
  - A19=0 is the same as A19=1 for Memory



- Connect to cs
  - If A19=0 Memory chip act normal fanction



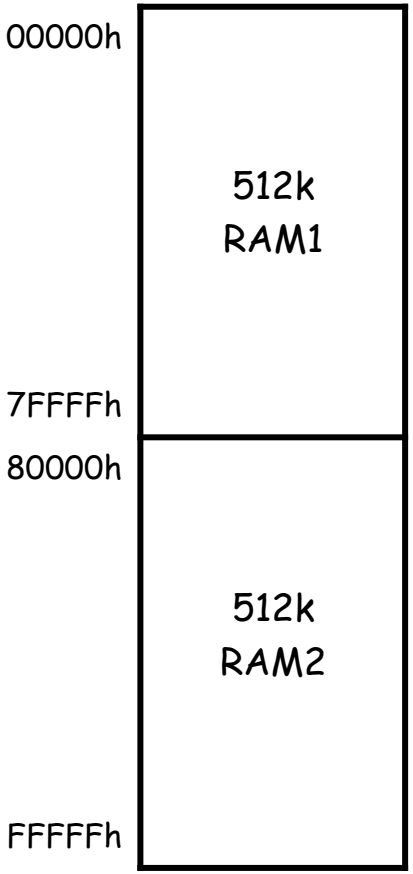
2 × 512 kB memory



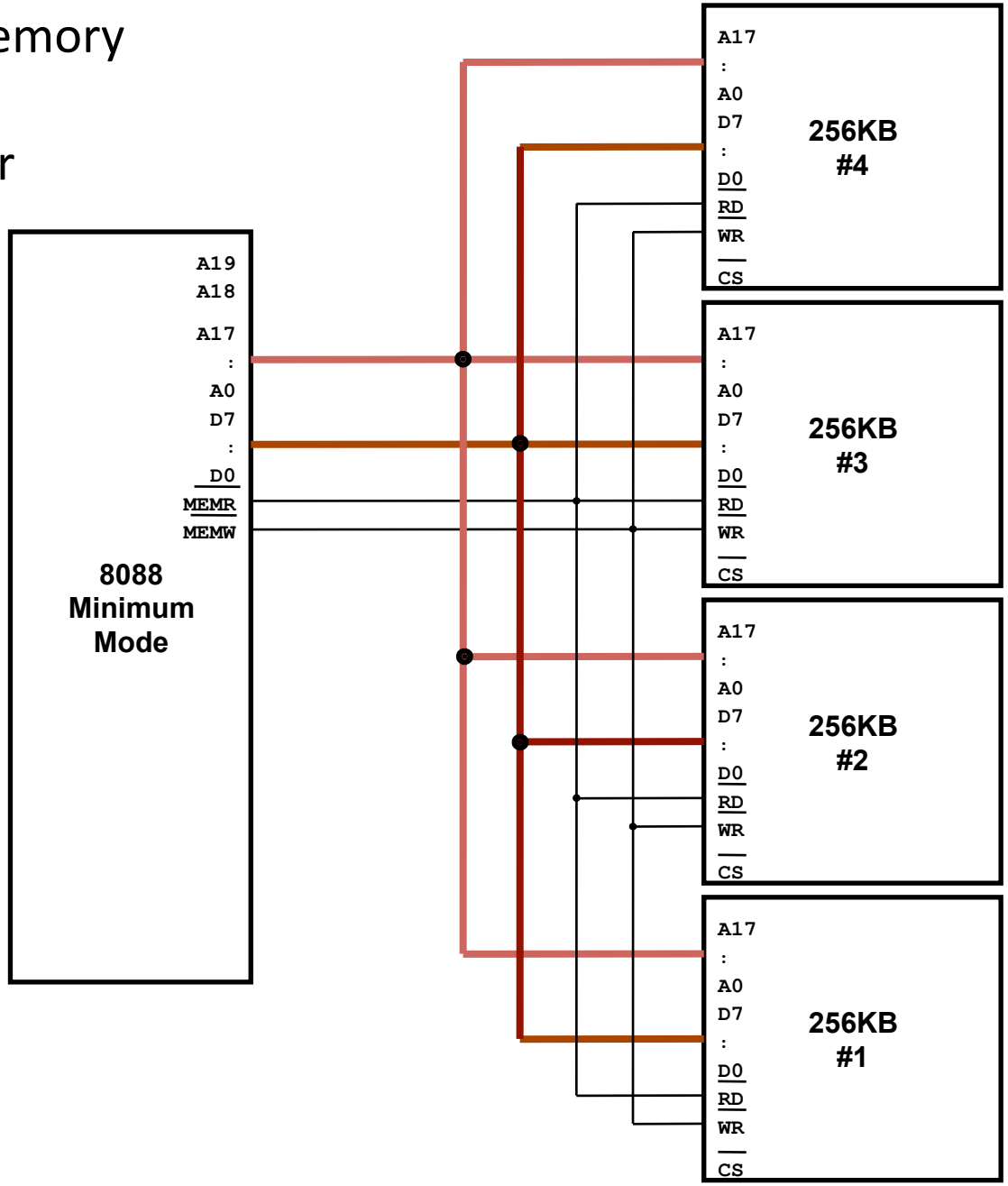
2 × 512 kB memory

What are the memory locations of two consecutive 512KB (2^{19} bytes) Memory?

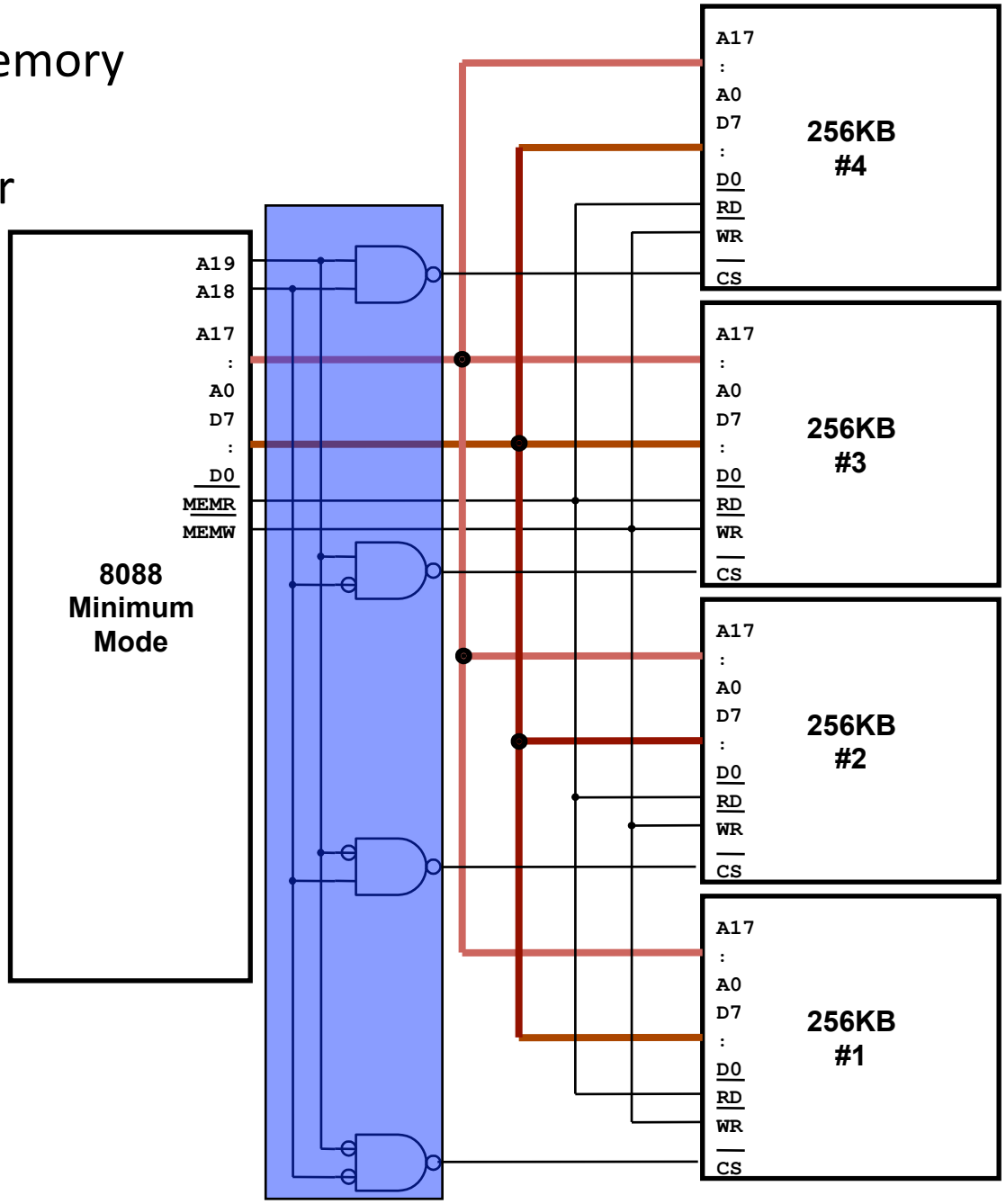
AAAA 1111 9876	AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
0000 0111	0000 1111	0000 1111	0000 1111	0000 1111	ROM
1000 1111	0000 1111	0000 1111	0000 1111	0000 1111	RAM



Interfacing four 256K Memory Chips to the 8088 Microprocessor



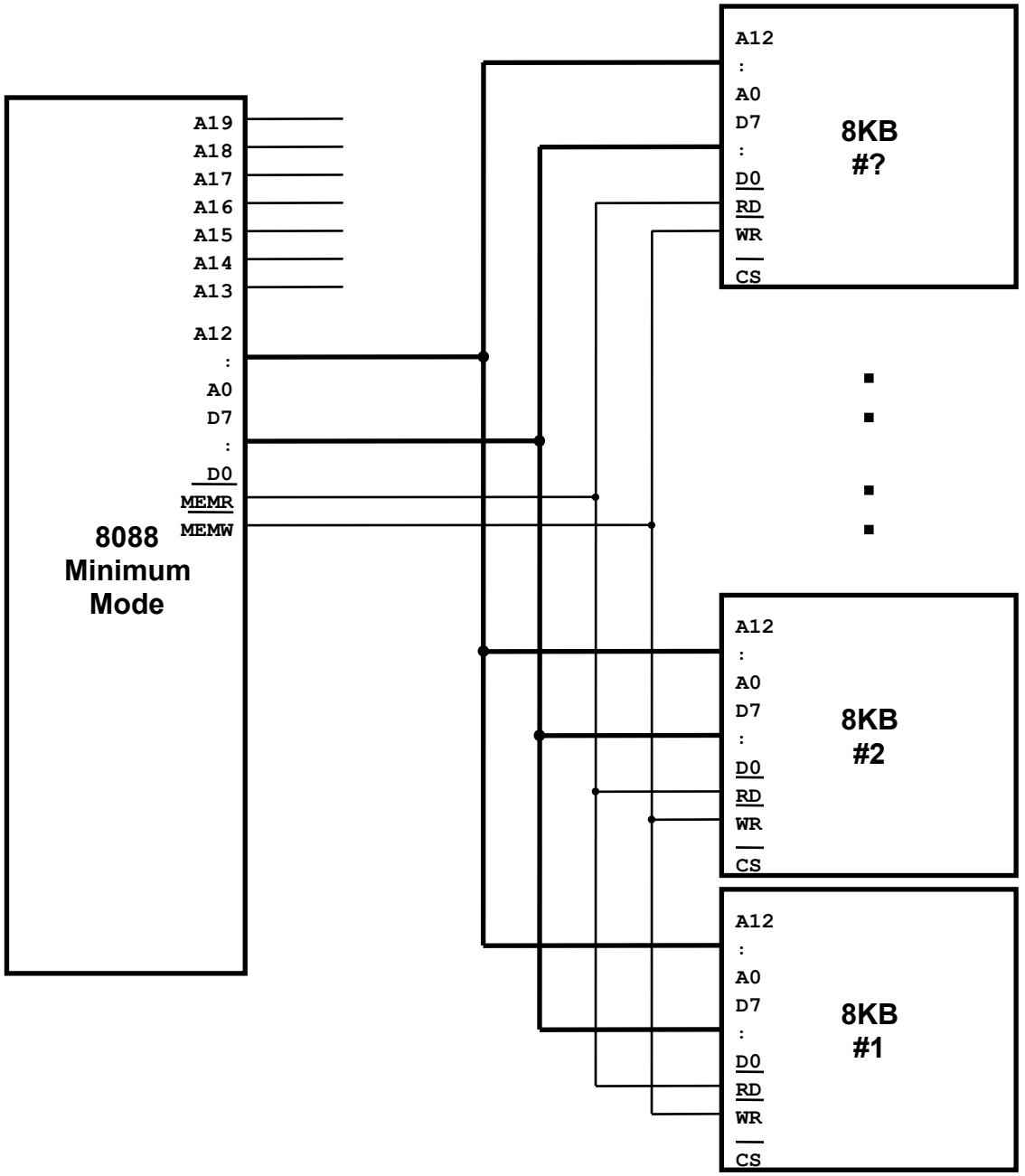
Interfacing four 256K Memory Chips to the 8088 Microprocessor



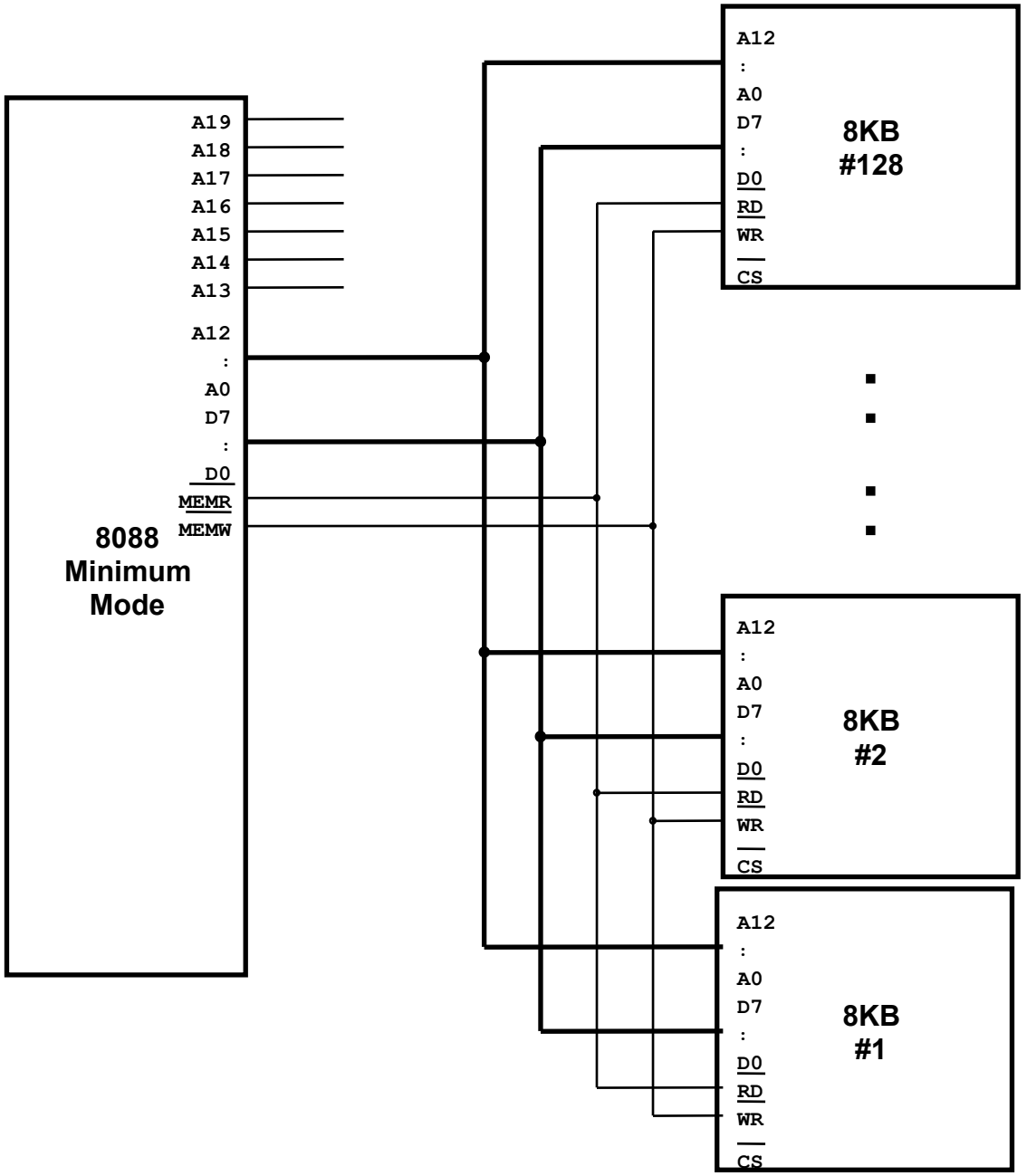
Memory chip#__ is mapped to:

AAAA 1111 9876	AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
					RAM#1
					RAM#2
					RAM#3
					RAM#4

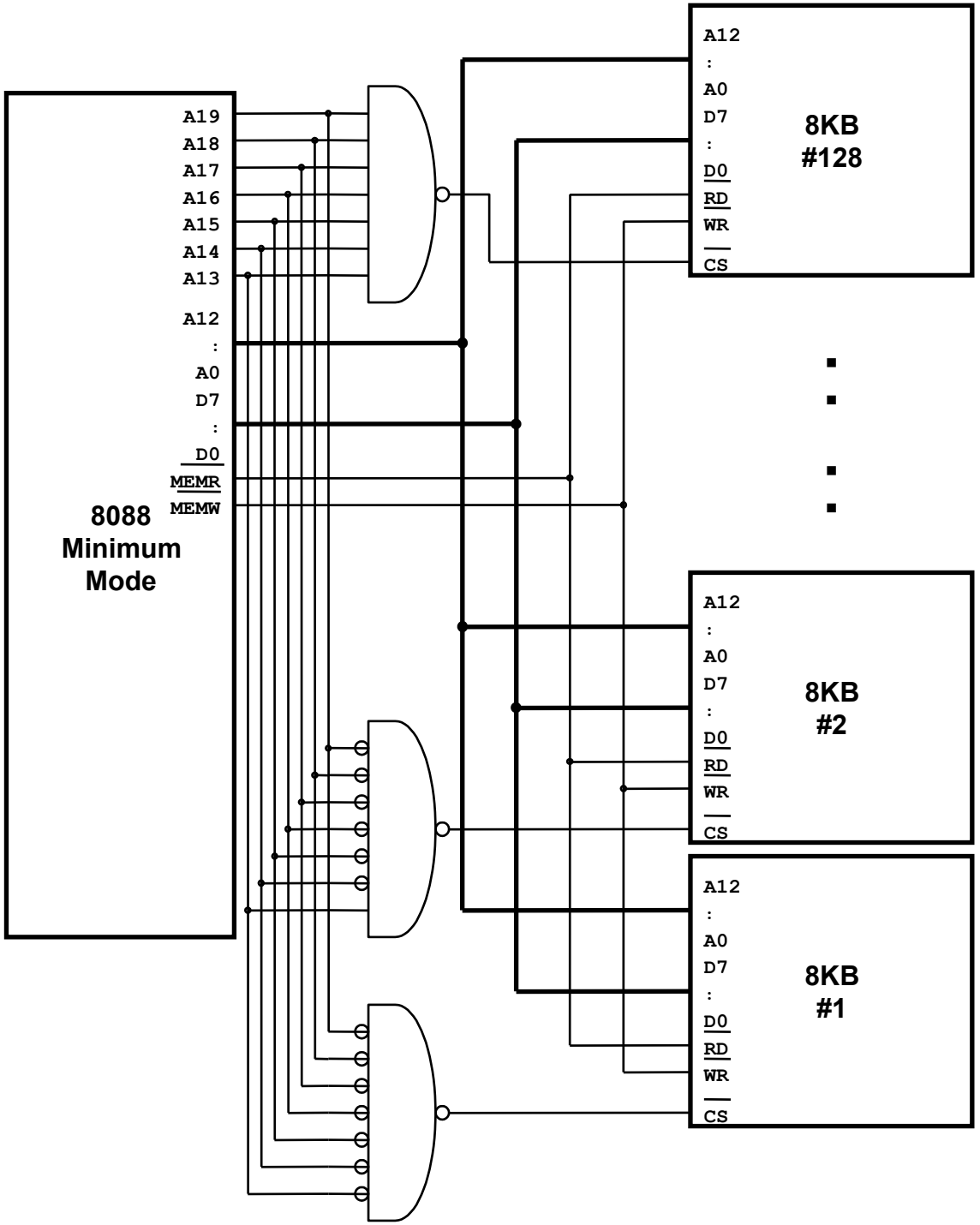
Interfacing several 8K Memory Chips to the 8088 μP



Interfacing 128 8K Memory Chips to the 8088 μP



Interfacing 128 8K Memory Chips to the 8088 μP



Memory chip#__ is mapped to:

AAAA 1111 9876	AAAA 1111 5432	AAAA 1198 10	AAAA 7654	AAAA 3210	Memory Chip
					RAM#1
					RAM#2
					RAM#126
					RAM#127
					RAM#128

What is the Memory and Address Bit map?