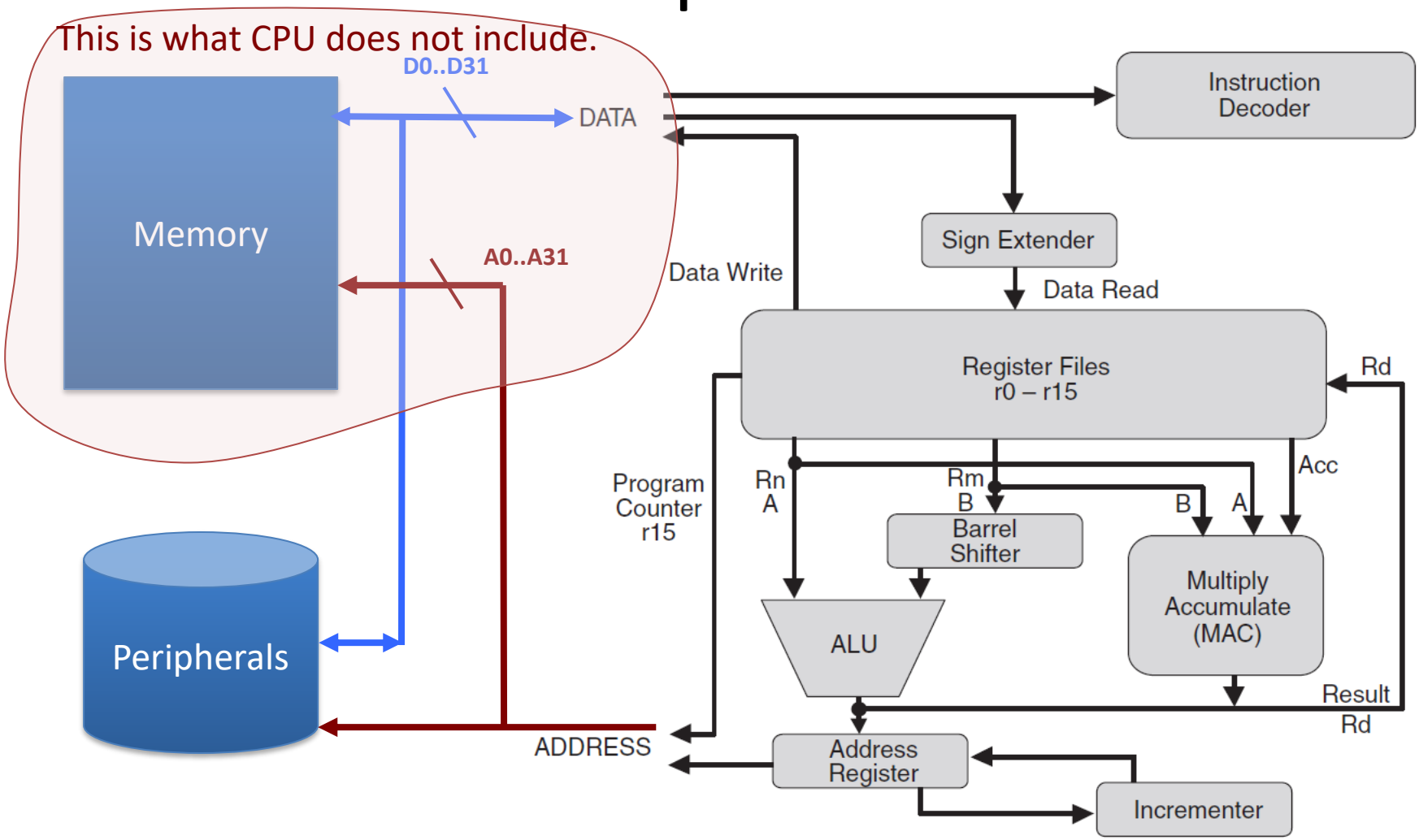
Bus and Memory Design

Ver. 2

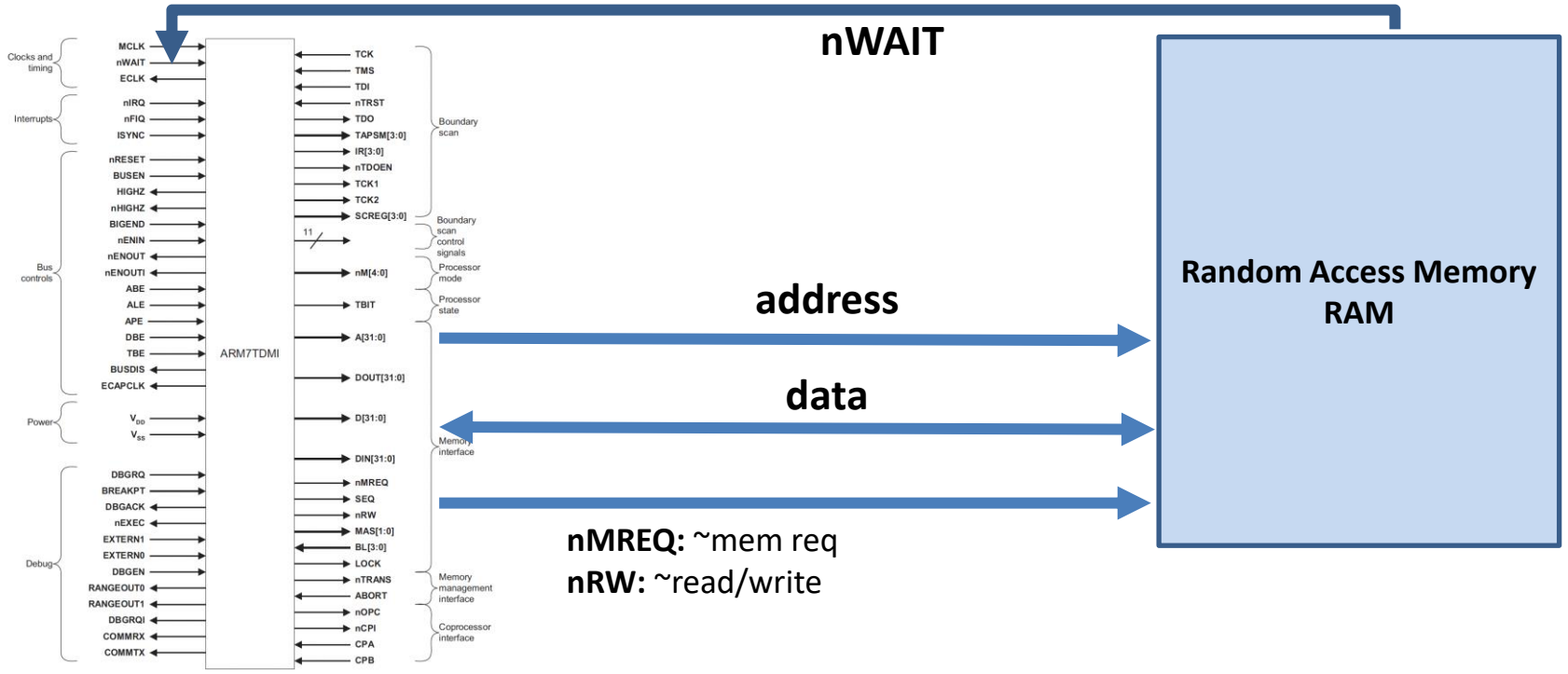
Professor: Yang Peng

Microprocessor

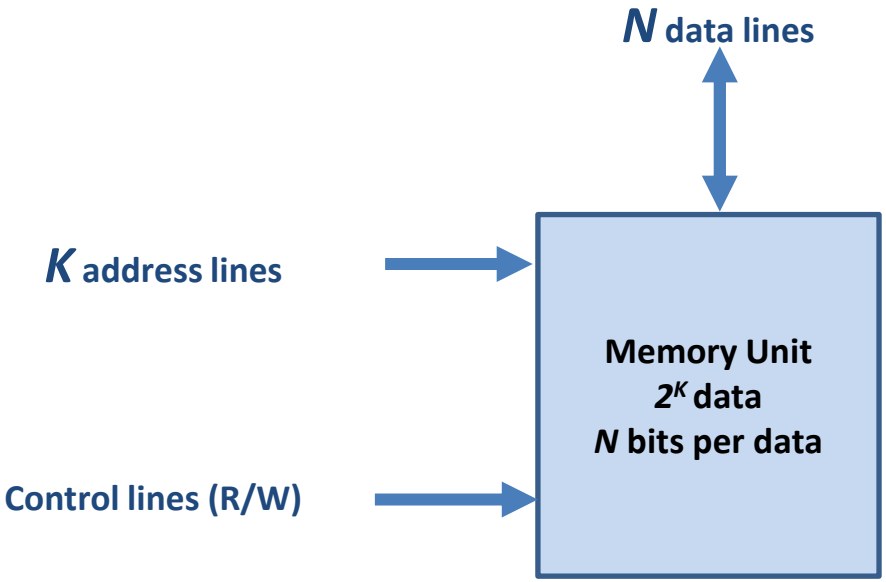


CPU and Memory

- CPU has only temporary storage, the registers, for computation
- Most data should be maintained in external memory
  - ROM (Read-Only Memory): code, const data
  - RAM (Random Access Memory): can store (after loading) your programs and data
- CPU-memory communication takes place with **address**, **data**, and **control** signals.



Memory Capacity



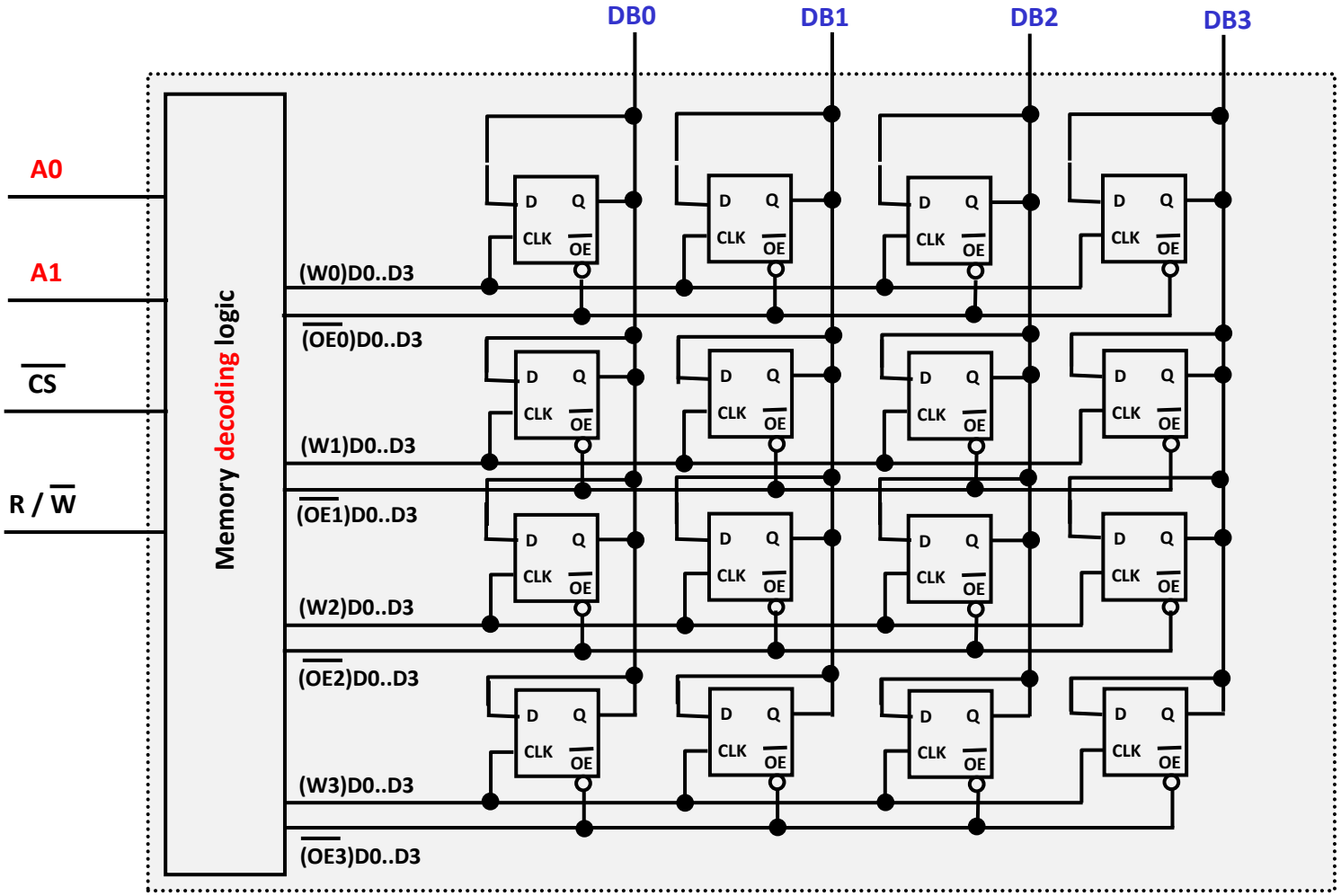
Memory capacity: $2^k \times N$

- Suppose a memory chip has 16 address lines and 8 data lines

  What is the capacity of the chip?
  - _____ bits
  - _____ bytes (8 bits)
  - _____ words (16 bits)

- Suppose a chip has a capacity of $2^K \times 16$, how many address bits and data bits?
  - _____ address bits
  - _____ data bits

Memory Organization

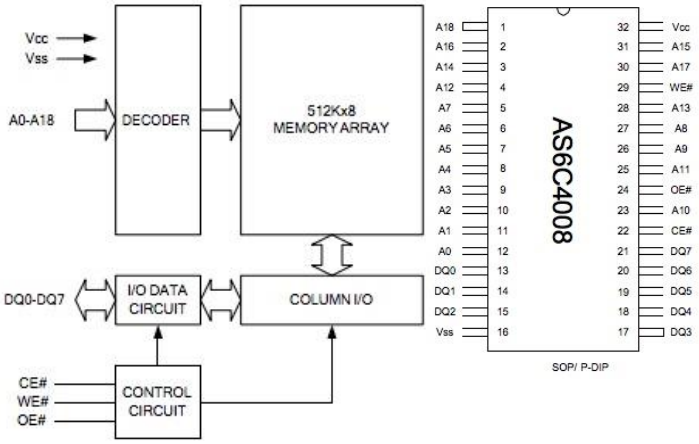


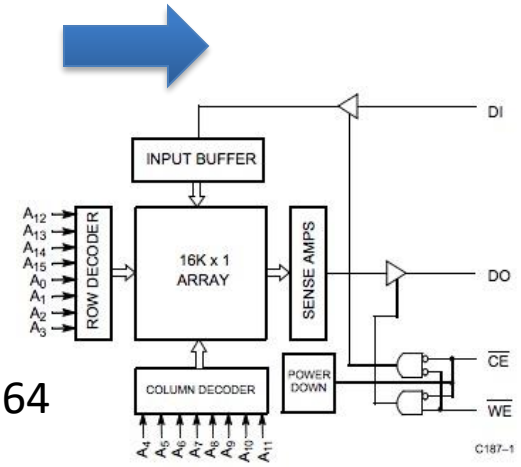
4 addresses x 4 bits/address = 16 bits = 2 bytes

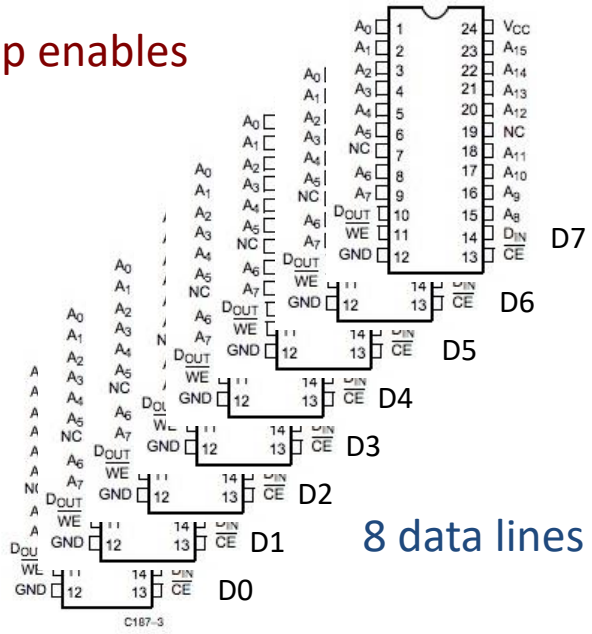
Memory Extension

- Large memory chip can be constructed with several smaller size of memory chips
- AS6C4008 is an SRAM with the capacity of 19 x 8.
- CY7C187 is an SRAM with the capacity of 16 x 1.
- How to arrange CY7C187s to replace an AS6C4008 with them?

3 more address lines
= decoded to 2^3 different chip enables



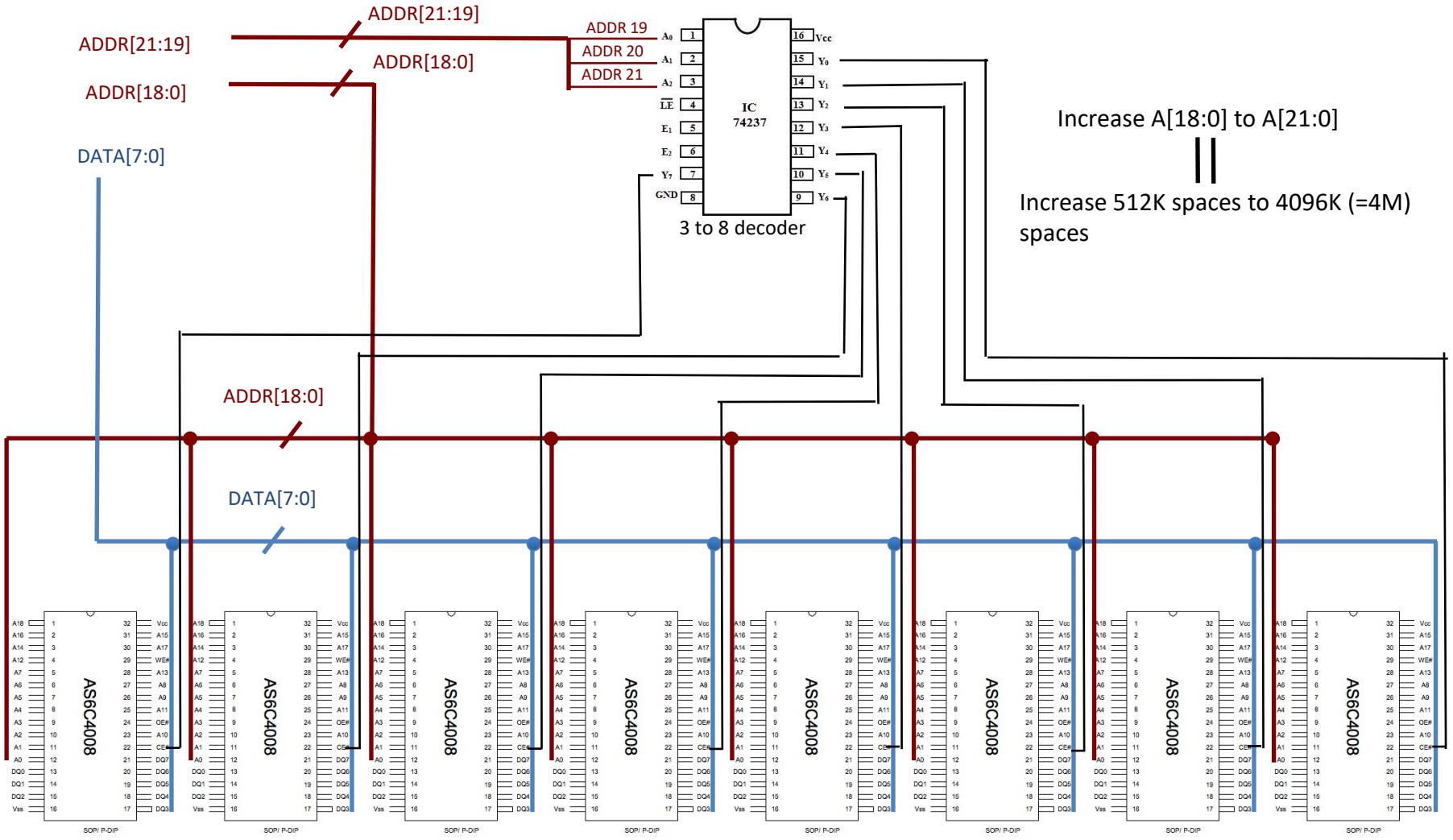




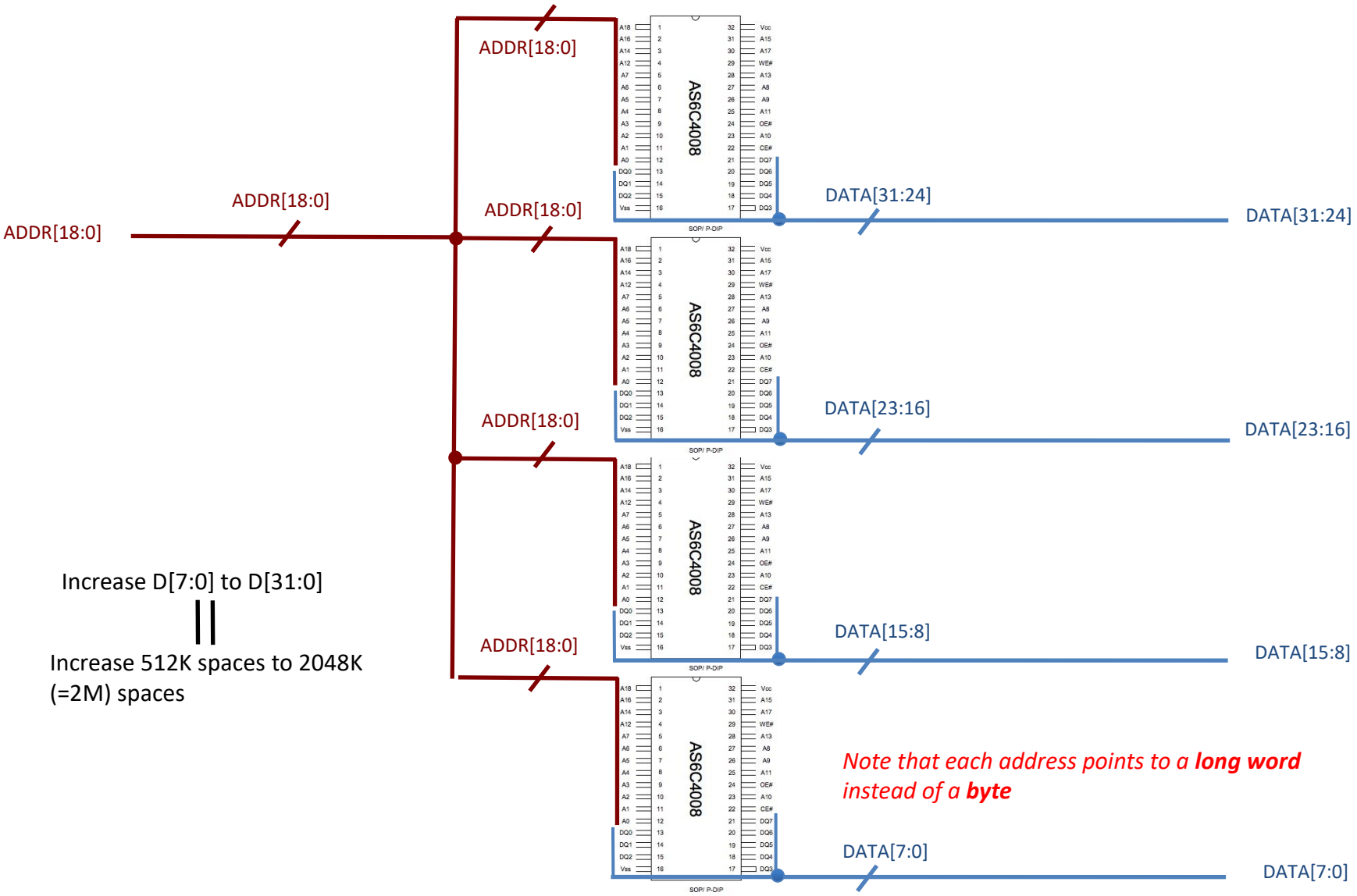
8 data lines

In total, $(2^{19} \times 8)/(2^{16} \times 1) = 2^3 \times 8 = 64$

Increase Address Spaces



Increase Data Width



Increase D[7:0] to D[31:0]

||

Increase 512K spaces to 2048K (=2M) spaces

*Note that each address points to a **long word** instead of a **byte***

Increase Both

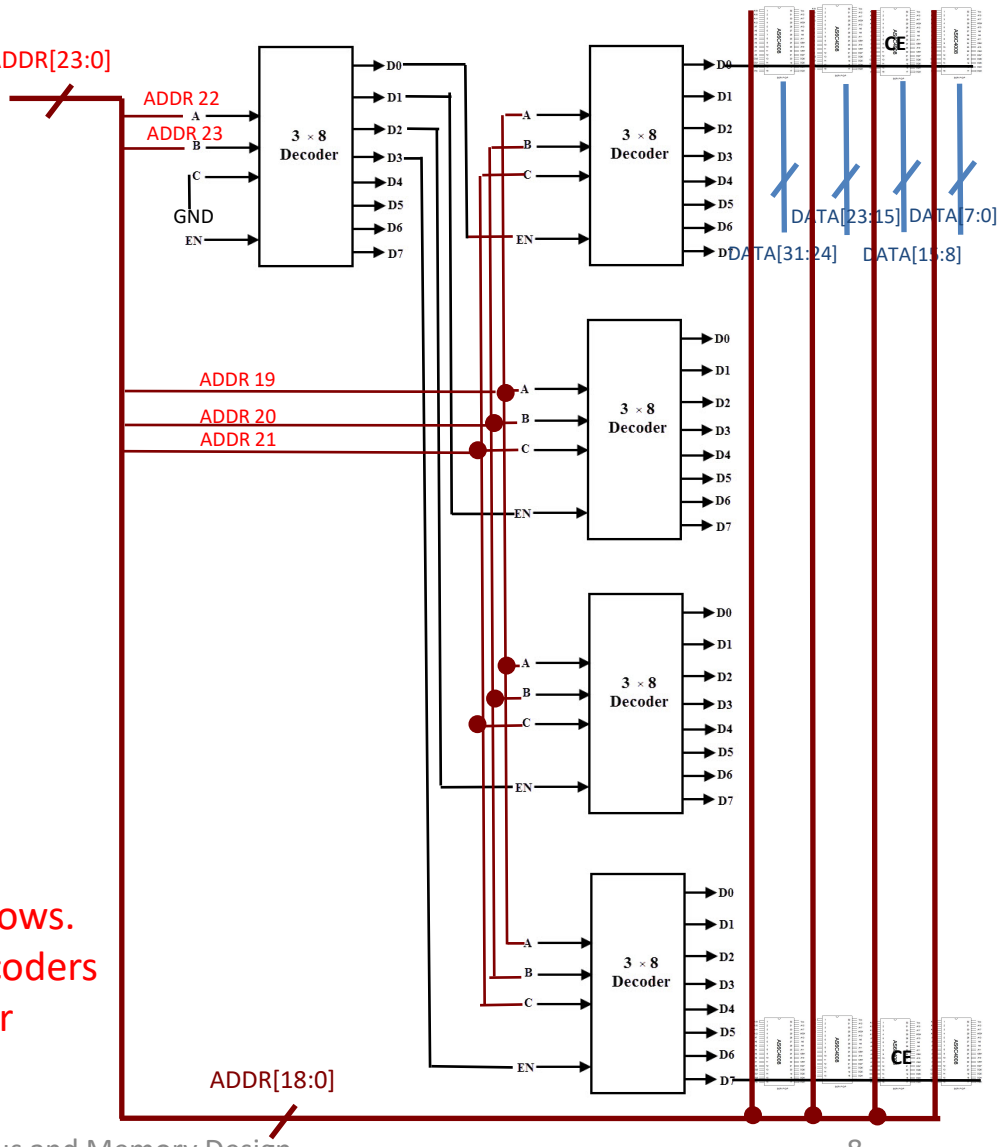
- Increase the address length
- Increase the data width

- Exercise:
  - Build a 2^{24} x 32 memory using 2^{19} x 8 memory chips
  - How many 2^{19} x 8 memory chips should you have?

  $(2^{24} \times 32) / (2^{19} \times 8) = 2^5 \times 4 = 128$ chips

  24-19 = 5 additional address lines
  - How many 3-to-8 decoders do you need to activate 128 CE lines?

four 8-bit chips (32 bits) in a row, 128/4 = 32 rows.
Select 1 out of 32 rows, need 32/8 = 4 3-8 decoders
Select 1 out of 4 decoders, need 1 3-8 decoder
5 decoders in total!

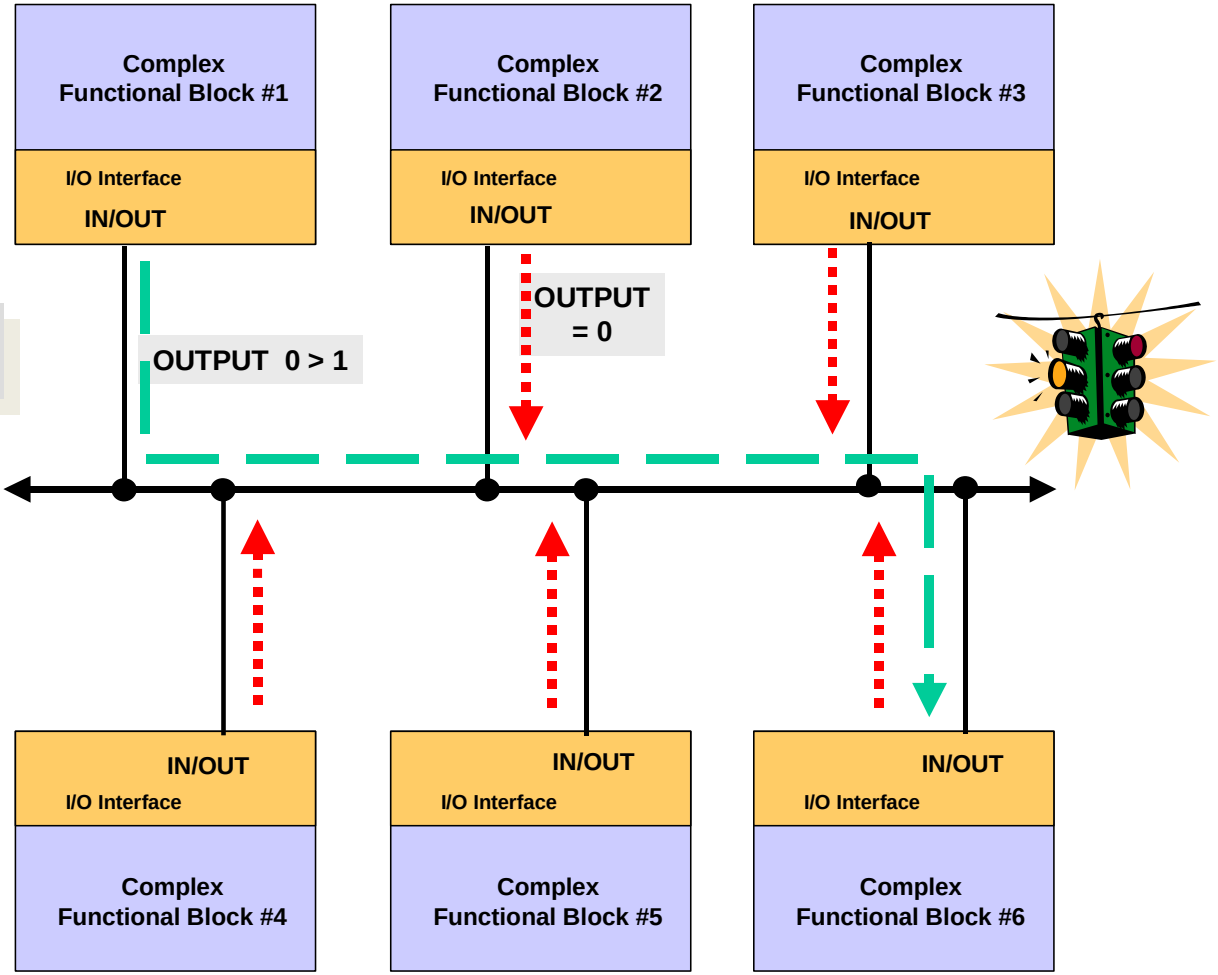


Bus and I/O

Ver. 2

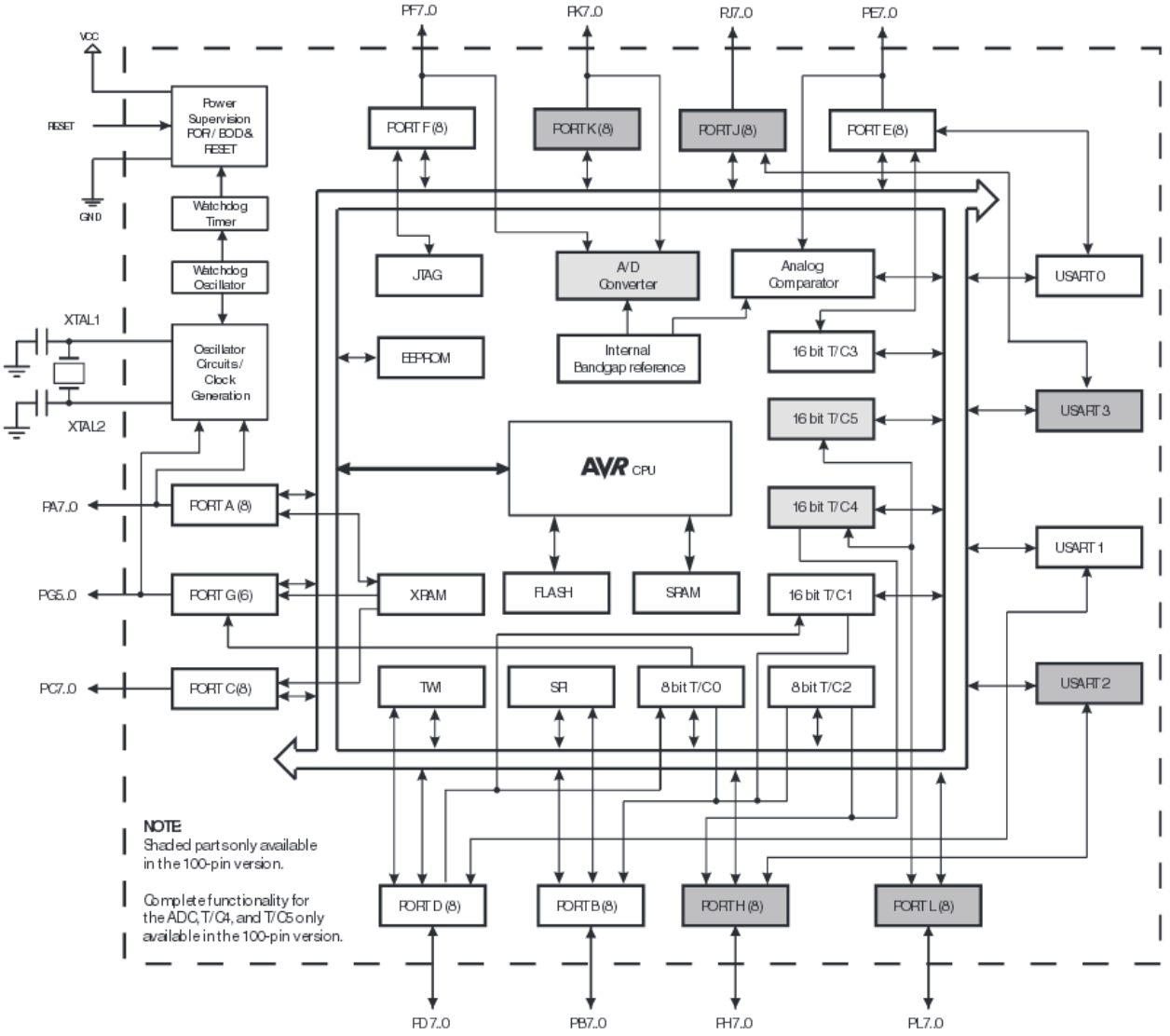
Bus Organization

How do we manage the flow of information?



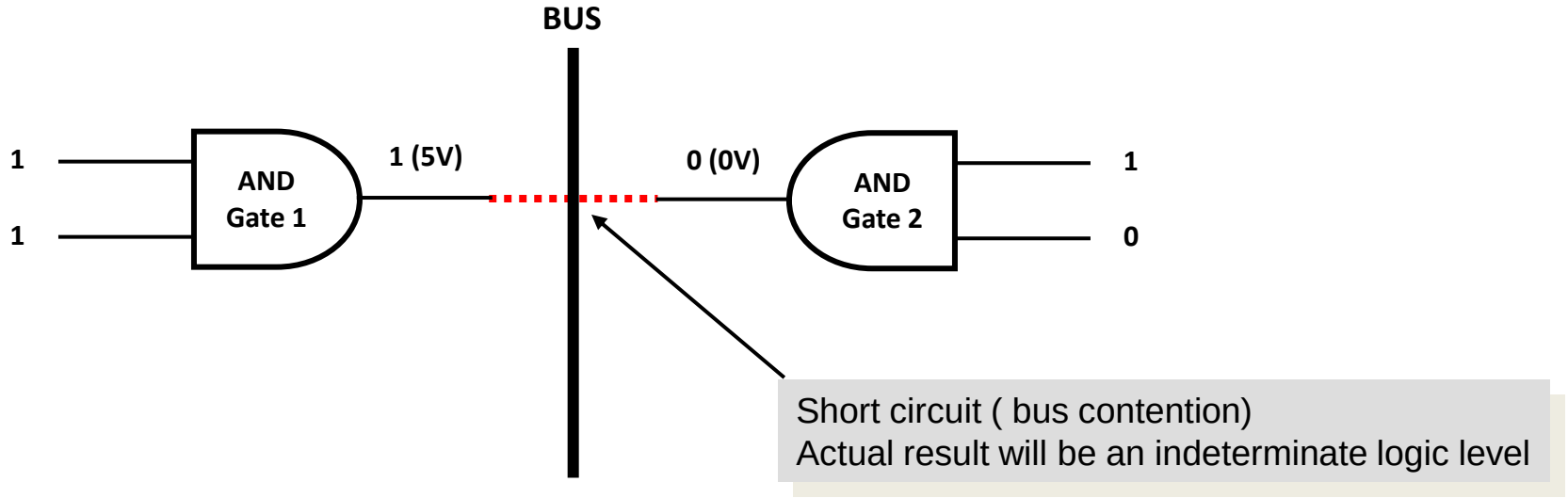
Example Bus Connection

- Busses were invented in order to simplify the organization and flow of data within computer systems
  - Busses allow many devices to connect to the same data path
  - Allow for efficient exchange of data between devices

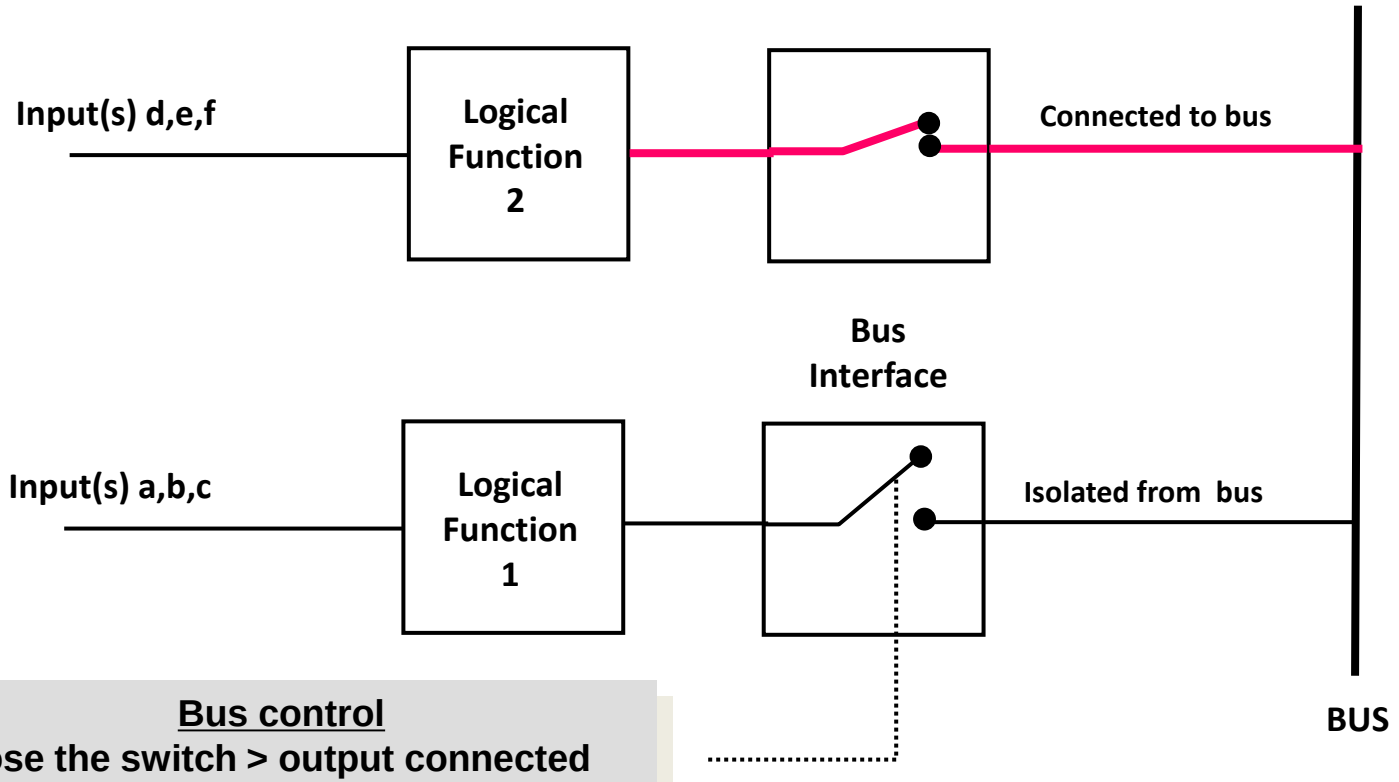


Bus Collision

- Question: How do I connect outputs together and not get a short circuit?



Bus Arbitration



Bus control

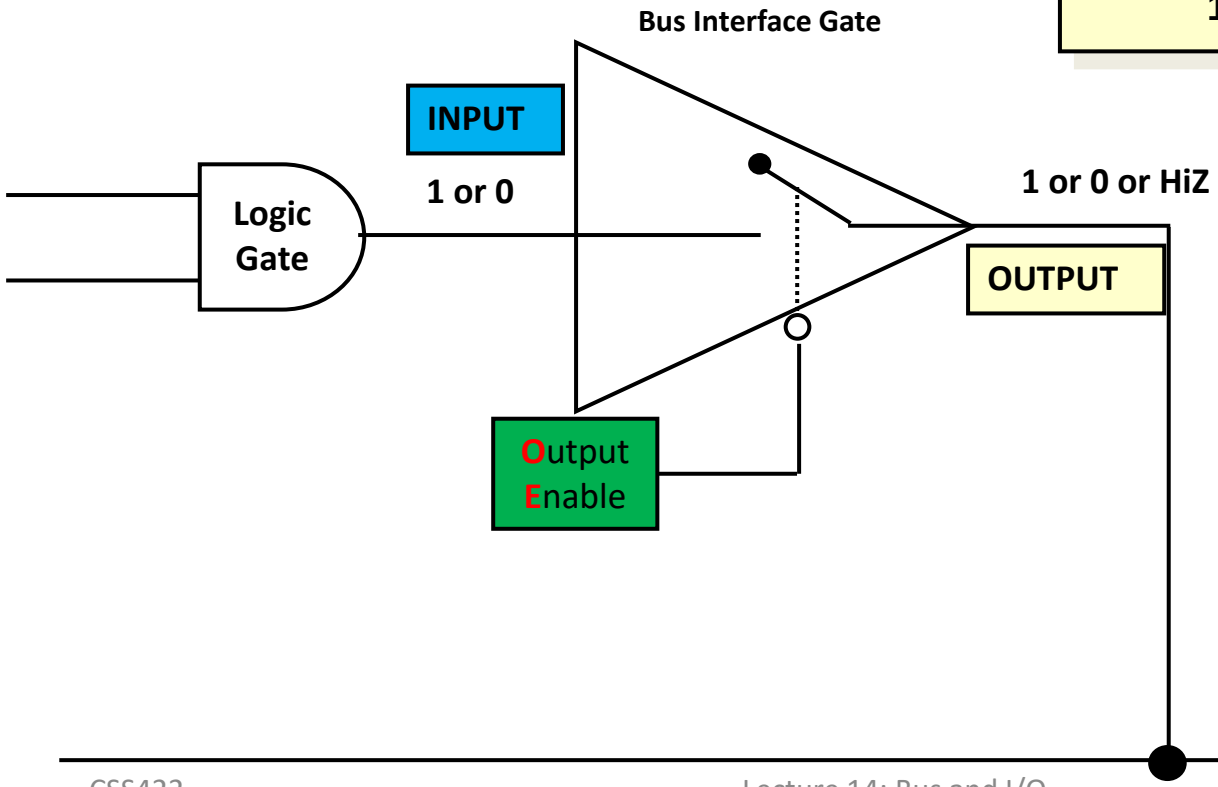
- Close the switch > output connected
- Open the switch > output disconnected

Bus Interface Structure: Tristate

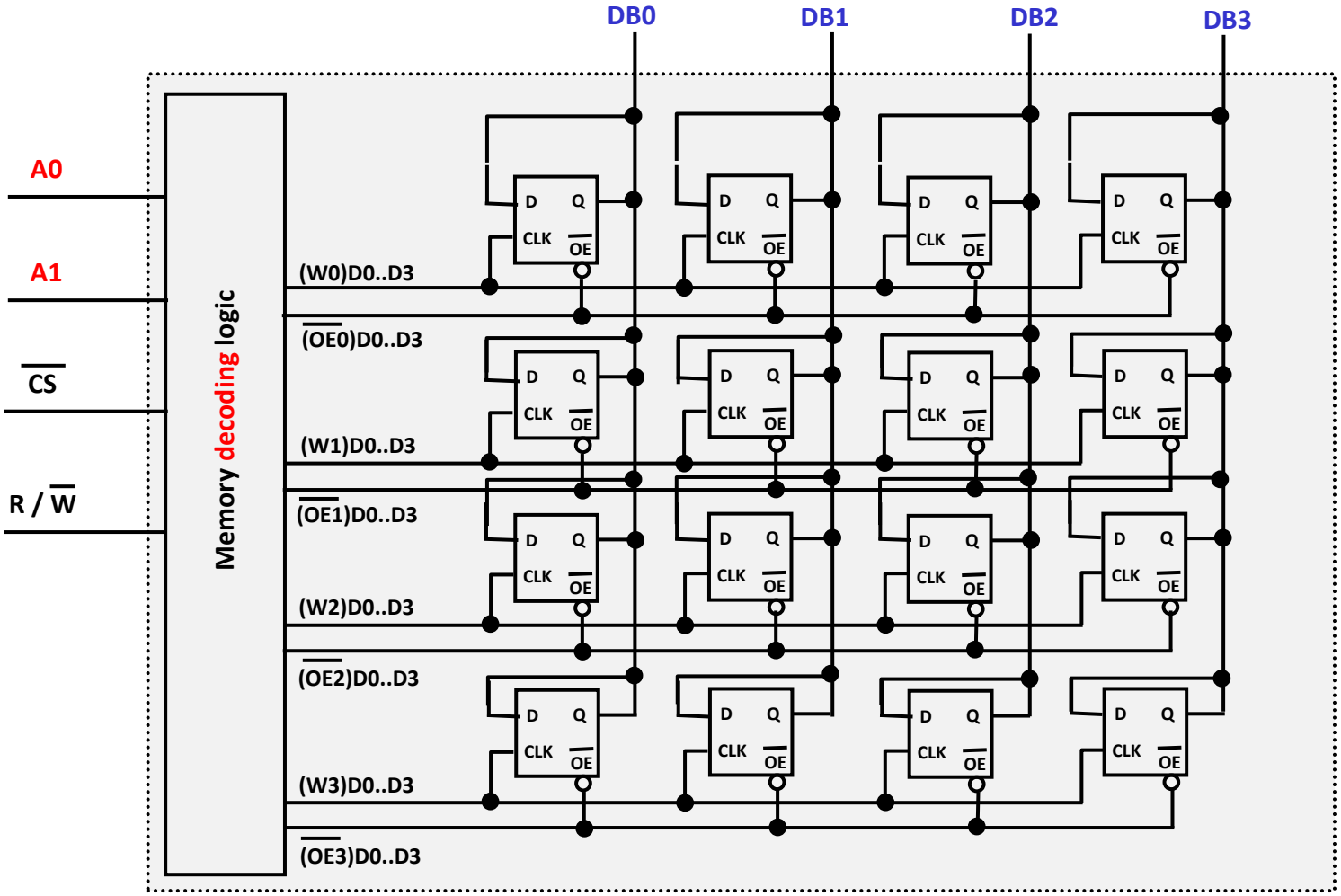
Truth table for Bus Interface Gate →

Output Enable	INPUT	OUTPUT
0	1	1
0	0	0
1	1	Hi Z
1	0	Hi Z

Hi Z: removing the gate from the circuit



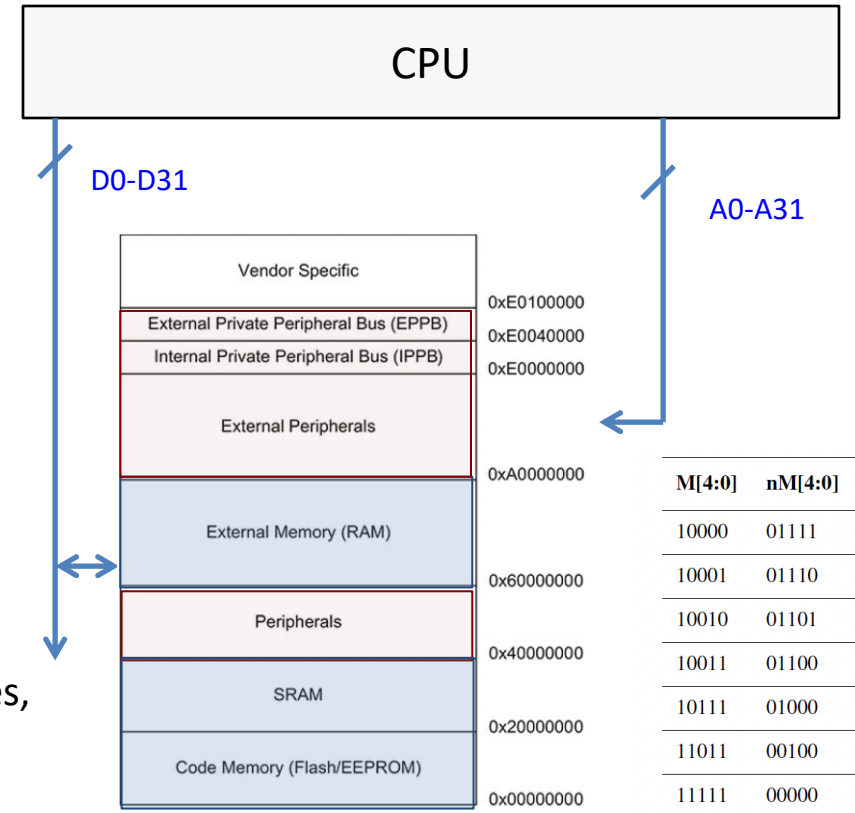
Memory Organization



4 addresses x 4 bits/address = 16 bits = 2 bytes

Memory-Mapped I/O

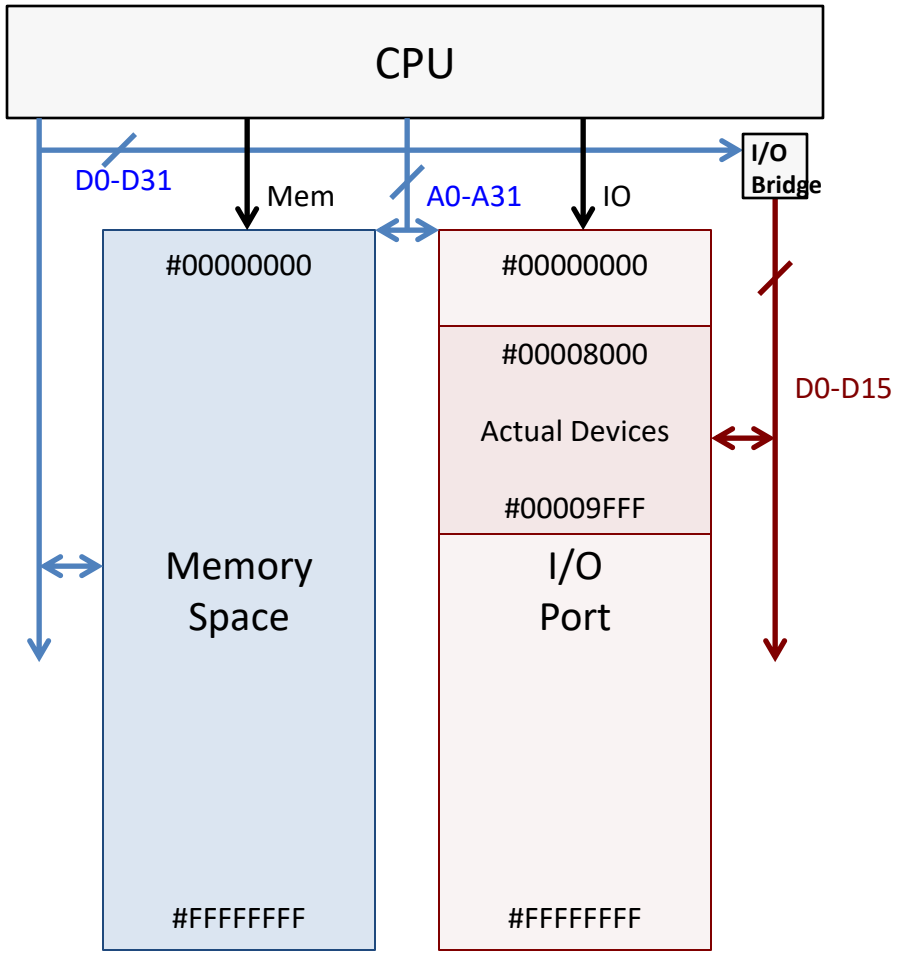
- I/O ports are mapped on the memory address space
- Processor
  - Easy implementation
- Address decoding
  - **More complicated**
  - nWAIT is used for slow I/O devices
- OS viewpoint
  - Can't cause a trap when a user program accesses I/O ports.
  - Yet reserve I/O ports for OS accesses, using M[4:0]



M[4:0]	nM[4:0]	Mode
10000	01111	User
10001	01110	FIQ
10010	01101	IRQ
10011	01100	Supervisor
10111	01000	Abort
11011	00100	Undefined
11111	00000	System

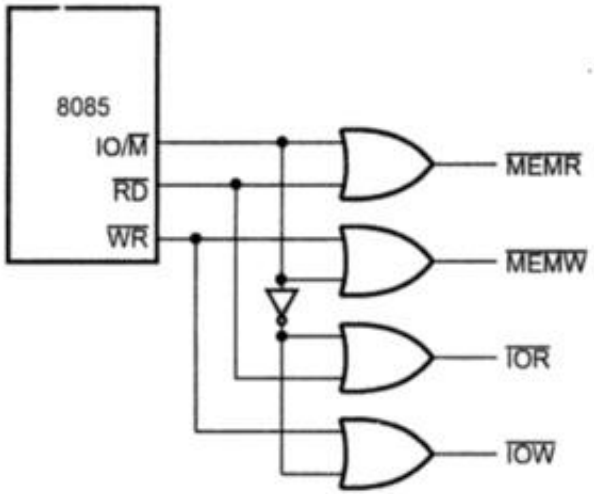
Port-Mapped I/O

- **I/O ports have a separate space** from the memory address space.
- Processor
  - Distinguishes data access and I/O operations
  - Needs a pin distinguish I/O and data memory access, such as IO/~M
- Address decoding
  - Easy
- OS viewpoint
  - Can cause a trap when a user program executes IN and OUT



Switching between Memory and I/O Address Spaces

- IO/~M distinguishes memory and I/O address spaces.
- ~RD and ~WR are shared among system memory and I/O advices



From Microprocessors and Microcontroller 1-15 8085 Microprocessor Architecture