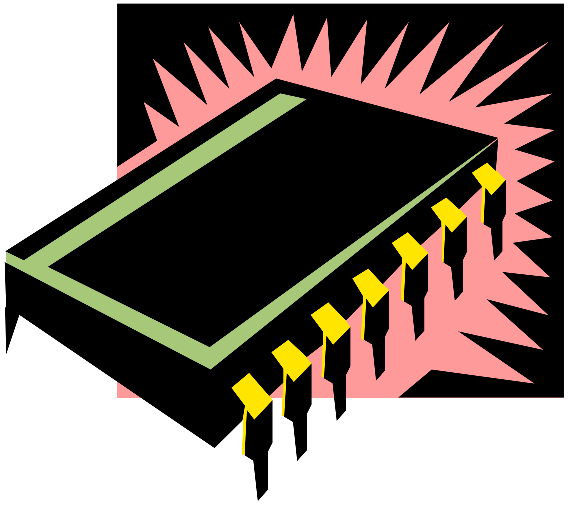
Random access memory

- Random access memory, or RAM, allows us to store even larger amounts of data than flip-flops or registers.
- Today we'll see the external and internal aspects of static RAM.
  - All memories share the same basic interface.
  - You can implement static RAM chips hierarchically.
- We'll also talk a bit about different kinds of dynamic memory, which also appear in all modern computer systems.
- This gives us all the pieces we need to put together a computer!



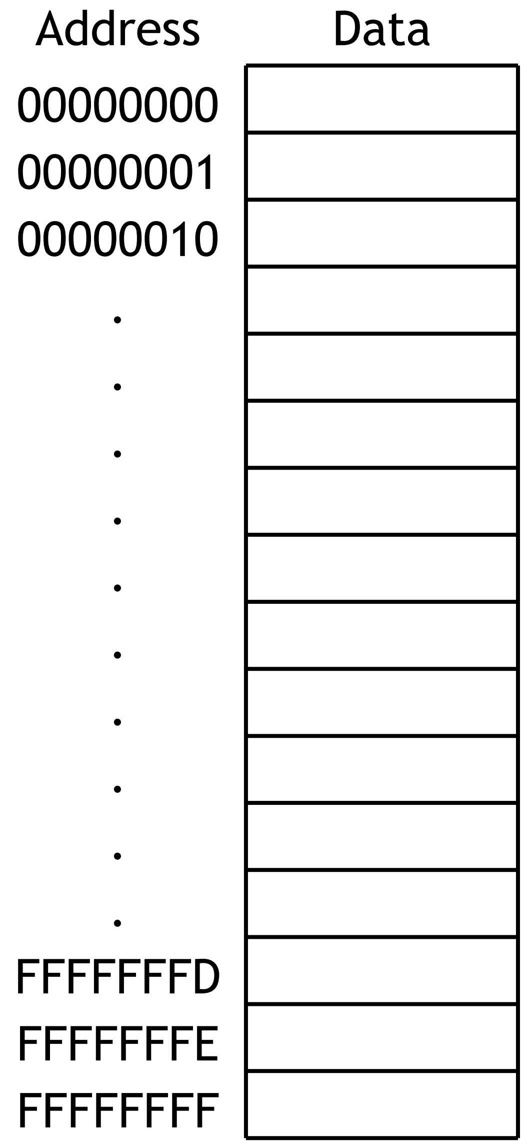
Introduction to RAM

- Random-access memory, or RAM, provides large quantities of temporary storage in a computer system.
- Remember the basic capabilities of a memory.
  - It should be able to store a value.
  - You should be able to read the value that was saved.
  - You should be able to change the stored value.
- A RAM is similar, except that it can store *many* values.
  - An address will specify which memory value we're interested in.
  - Each value can be a multiple-bit word (e.g., 32 bits).
- We'll refine the memory properties as follows.

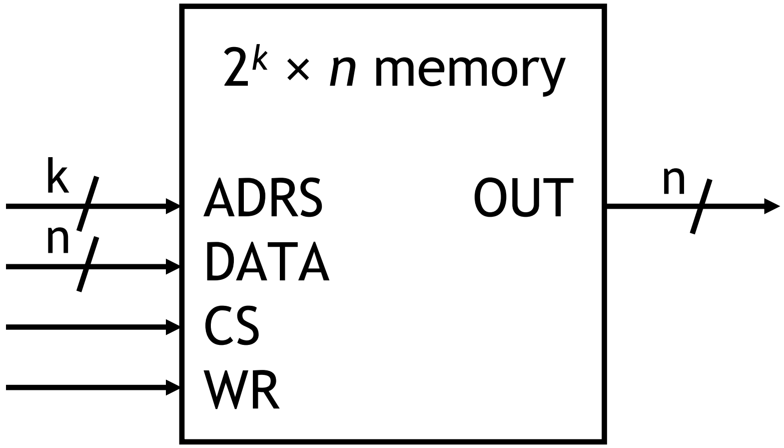
> A RAM should be able to:
> 1. Store many words, one per address
> 2. Read the word that was saved at a particular address
> 3. Change the word that's saved at a particular address

Picture of memory

- You can think of computer memory as being one big array of data.
  - The address serves as an array index.
  - Each address refers to one word of data.
- You can read or modify the data at any given memory address, just like you can read or modify the contents of an array at any given index.
- If you’ve worked with pointers in C or C++, then you’ve already worked with memory addresses.



Block diagram of RAM

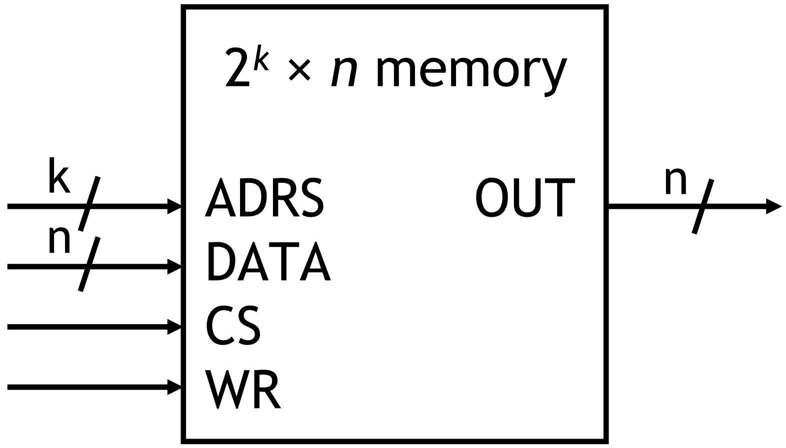


CS	WR	Memory operation
0	x	None
1	0	Read from ADRS
1	1	Write DATA to ADRS

- This block diagram introduces the main interface to RAM.
  - A Chip Select, CS, enables or disables the RAM.
  - ADRS specifies the address or location to read from or write to.
  - WR selects between reading from or writing to the memory.
    - To read from memory, WR should be set to 0.
      OUT will be the *n*-bit value stored at ADRS.
    - To write to memory, we set WR = 1.
      DATA is the *n*-bit value to save in memory.
- This interface makes it easy to combine RAMs together, as we'll see.

Memory sizes

- We refer to this as a $2^k \times n$ memory.
  - There are k address lines, which can specify one of 2^k addresses.
  - Each address contains an n-bit word.



- For example, a $2^{24} \times 16$ RAM contains 2^{24} = 16M words, each 16 bits long.
  - The RAM would need 24 address lines.
  - The total storage capacity is $2^{24} \times 16 = 2^{28}$ bits.

Size matters!

- Memory sizes are usually specified in numbers of bytes (8 bits).
- The 2^{28}-bit memory on the previous page has a capacity of 2^{25} bytes.

$$2^{28} \text{ bits} / 8 \text{ bits per byte} = 2^{25} \text{ bytes}$$

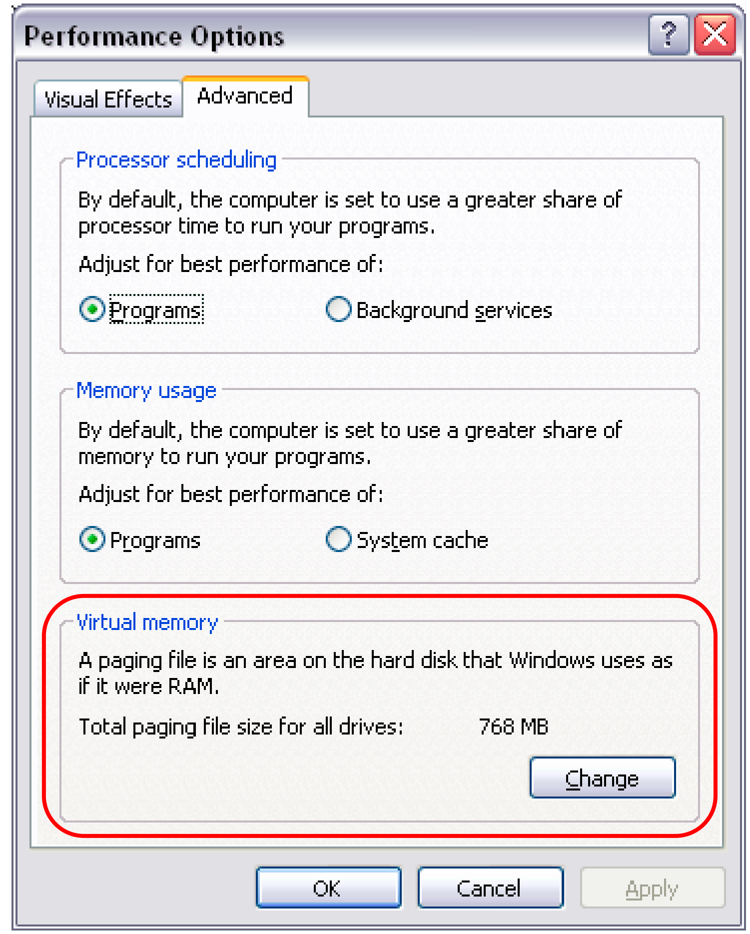
- With the abbreviations below, this is equivalent to 32 megabytes.

$$2^{25} \text{ bytes} = 2^{5} \times 2^{20} \text{ bytes} = 32 \text{ MB}$$

	Prefix	Base 2	Base 10
K	Kilo	2^{10} = 1,024	10^{3} = 1,000
M	Mega	2^{20} = 1,048,576	10^{6} = 1,000,000
G	Giga	2^{30} = 1,073,741,824	10^{9} = 1,000,000,000

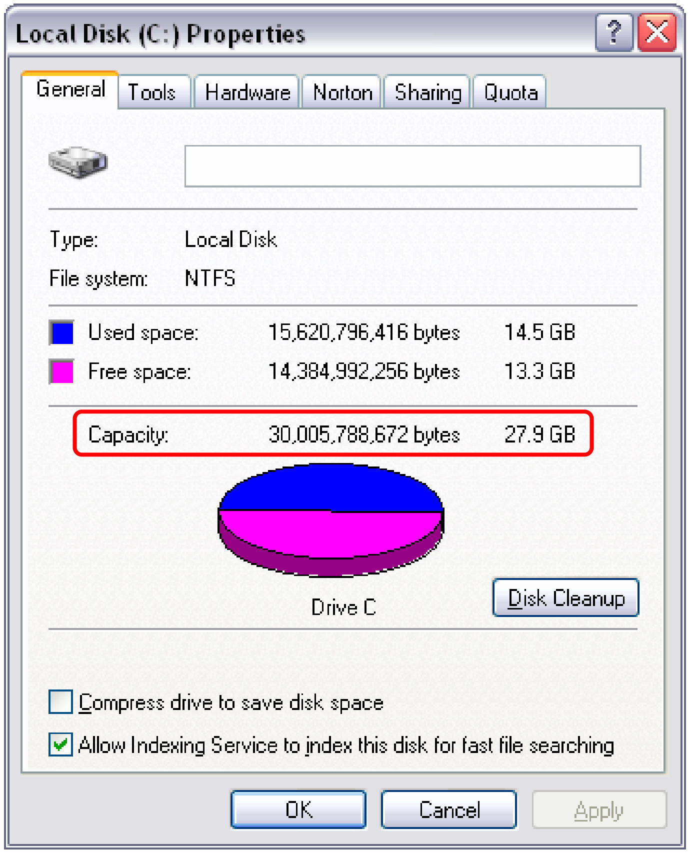
Typical memory sizes

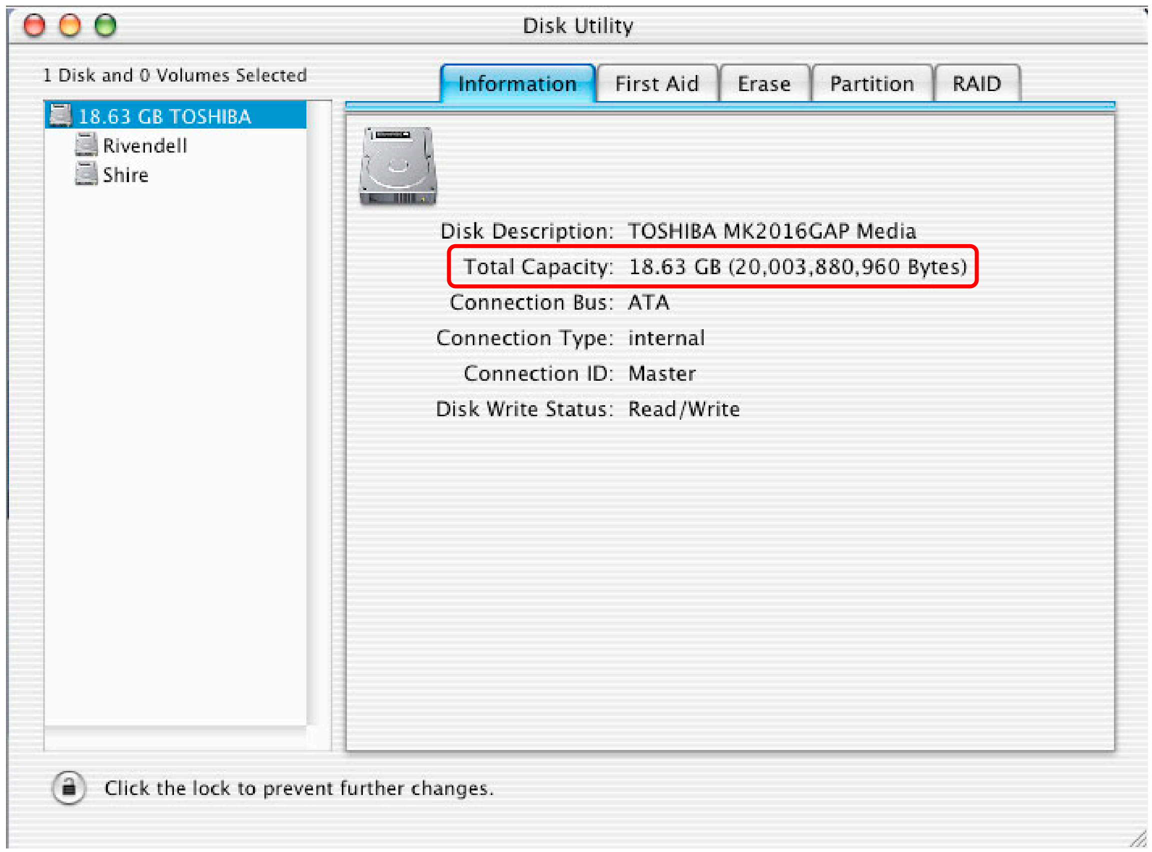
- Current PCs work with 32-bit addresses. With one byte of data at each address, this results in a maximum memory of 4GB.
- Most consumer devices have far less RAM.
  - PCs usually come with 128-256MB.
  - PDAs have 8-64MB of memory.
  - Digital cameras and MP3 players can have 32MB or more of storage.
- Many operating systems implement virtual memory, which uses hard drive space as an extension of the physical memory.
  - Hard drives are much slower, but also much cheaper, than main memory.
  - The hard drive space used for virtual memory may be a separate partition, or it may appear as a normal file like `PAGEFILE.SYS` or `swapfile0`.



Be careful what you buy

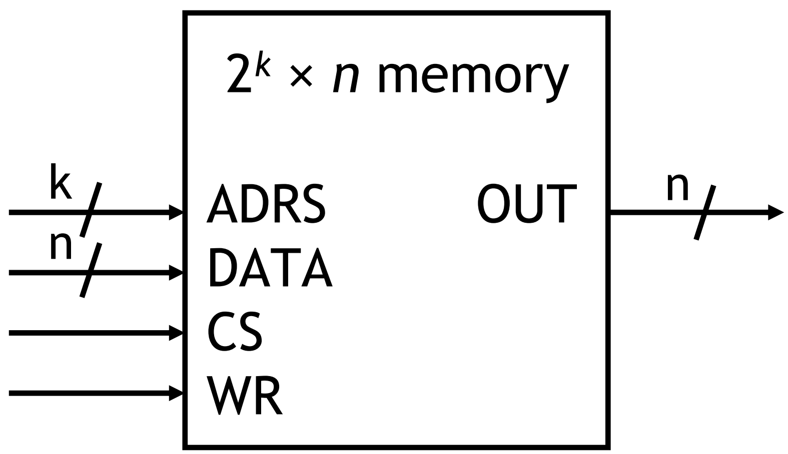
- To confuse you, RAM size is measured in base 2 units, while hard drive size is measured in base 10 units.
- To confuse you more, operating systems often report hard drive sizes in base 2 units. This is why a "30GB" hard drive shows up with only 27.9GB of free space, or why a "20GB" drive has only 18.63GB available.





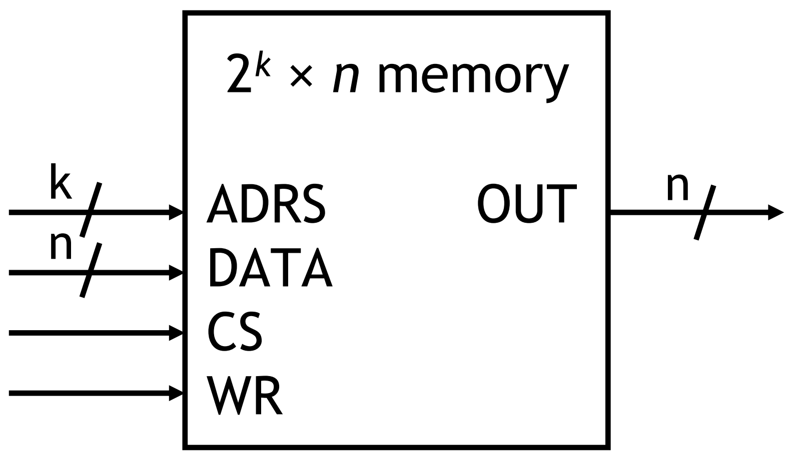
Reading RAM

- To *read* from a RAM, the controlling circuit must take several steps.
  1. Enable the chip by ensuring CS = 1.
  2. Select the read operation, by setting WR = 0.
  3. Send the desired address to the ADRS input.
  4. The contents of that address appear on OUT after a little while.
- Notice that the DATA input is unused for read operations.



Writing RAM

- To *write* to this RAM, you need to do the following tasks.
  1. Enable the chip by setting CS = 1.
  2. Select the write operation, by setting WR = 1.
  3. Send the desired address to the ADRS input.
  4. Send the word to store to the DATA input.
- The output OUT is not needed for memory write operations.

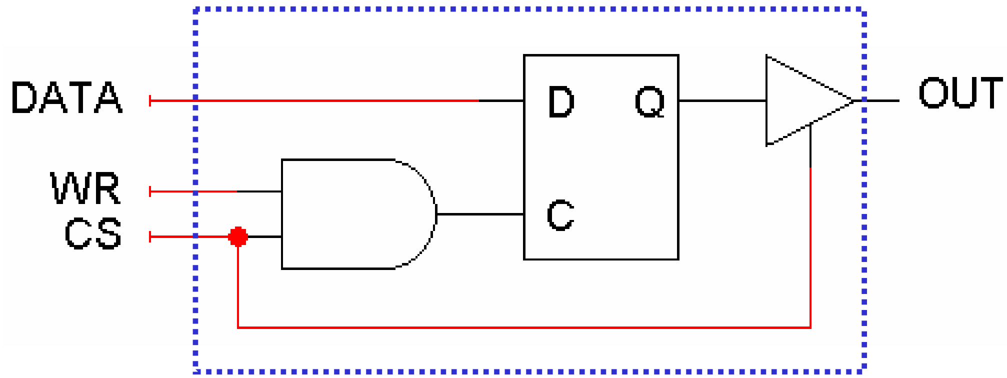


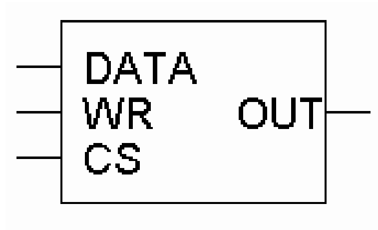
Static memory

- What's inside that memory block symbol?
- There are many different kinds of RAM.
  - We'll start off discussing static memory, which is most commonly used in caches and video cards.
  - Later we'll mention a little about dynamic memory, which forms the bulk of a computer's main memory.
- Static memory is modelled using one *latch* for each bit of storage.
- Why use latches instead of flip flops?
  - A latch can be made with only two NAND or two NOR gates, but a flip-flop requires at least twice that much hardware.
  - In general, smaller is faster, cheaper and requires less power.
  - The tradeoff is that getting the timing exactly right is a pain.

Starting with latches

- To start, a one-bit RAM cell is shown here. Since this stores just one bit, an ADRS input is not needed.

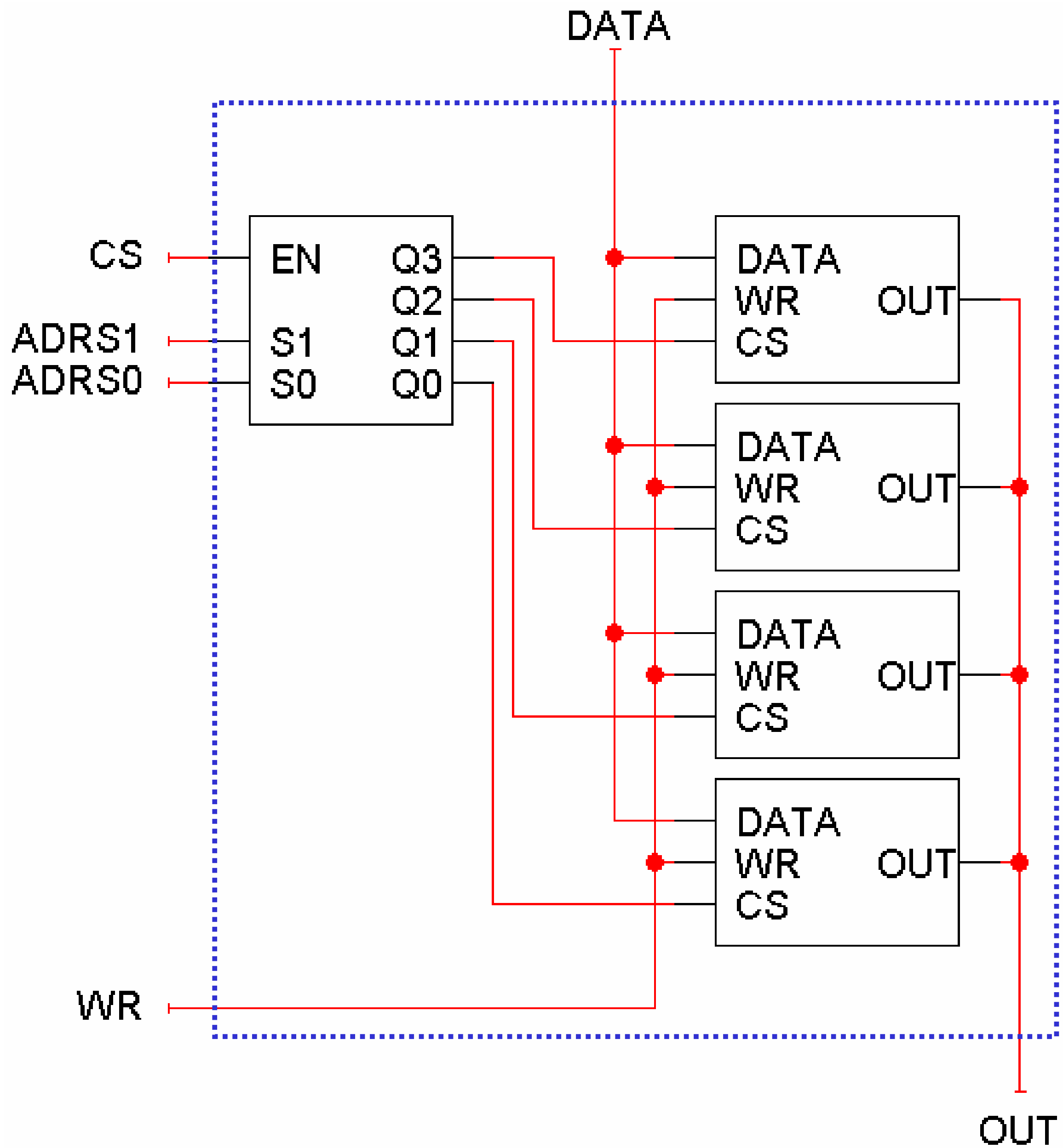




- You can ignore the funny triangle symbol for now.
- We can write to this RAM cell.
  - When CS = 1 and WR = 1, the latch control input will be 1.
  - The DATA input is thus saved in the D latch.
- We can also read from it and maintain the current contents.
  - When CS = 0 or when WR = 0, the latch control input is also 0, so the latch just maintains its present state.
  - The current latch contents will appear on OUT.

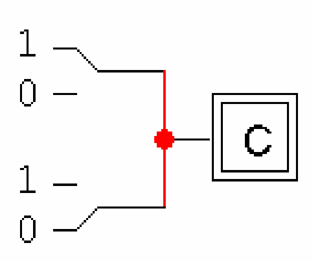
My first RAM

- We can use these cells to make a 4 x 1 RAM.
- Since there are four words, ADRS is two bits.
- Each word is only one bit, so DATA and OUT are one bit each.
- Word selection is done with a decoder attached to the CS inputs of the RAM cells. Only one cell can be read or written at a time.
- Notice that the outputs are connected together with a *single* line!



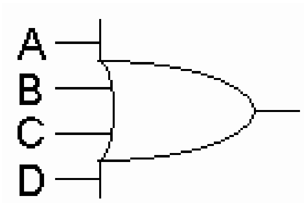
Connecting outputs together

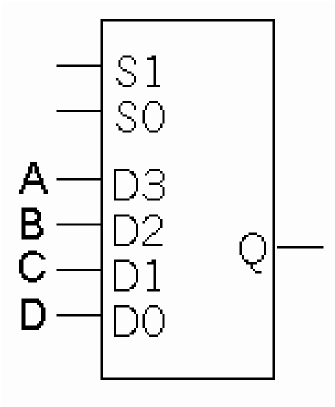
- In normal practice, it's bad to connect outputs together. If the outputs have different values, then a conflict arises.



The "C" in LogicWorks means "conflict."

- The standard way to "combine" outputs is to use OR gates or muxes.

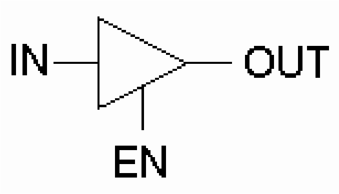




- This can get expensive, with many wires and gates with large fan-ins.

Those funny triangles

- The triangle represents a three-state buffer.
- Unlike regular logic gates, the output can be one of *three* possible values, as shown in the table.

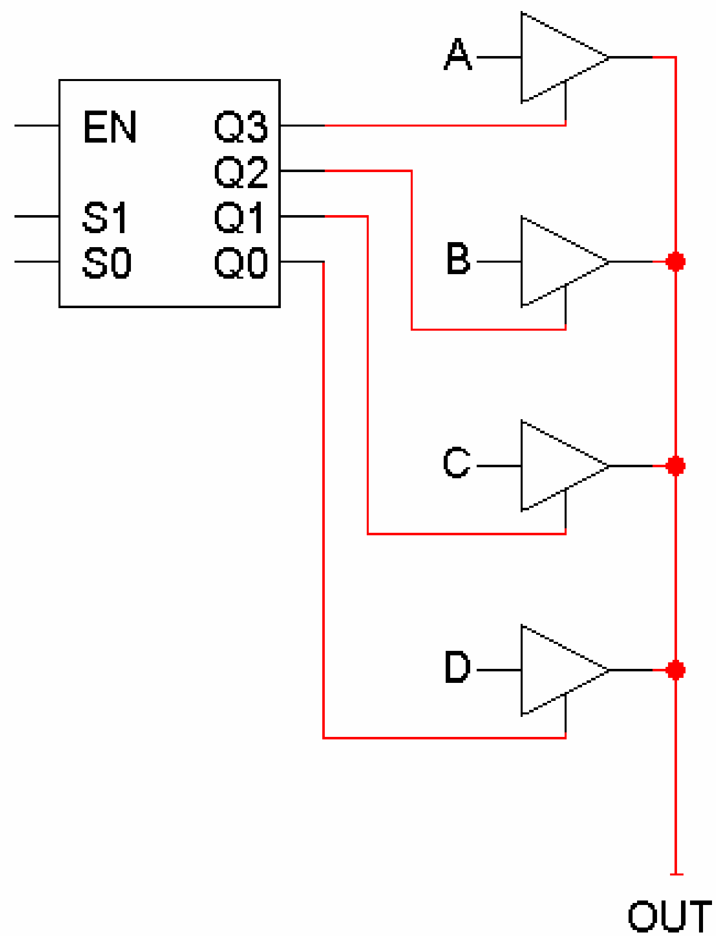


EN	IN	OUT
0	x	Disconnected
1	0	0
1	1	1

- "Disconnected" means no output appears at all, in which case it's safe to connect OUT to another output signal.

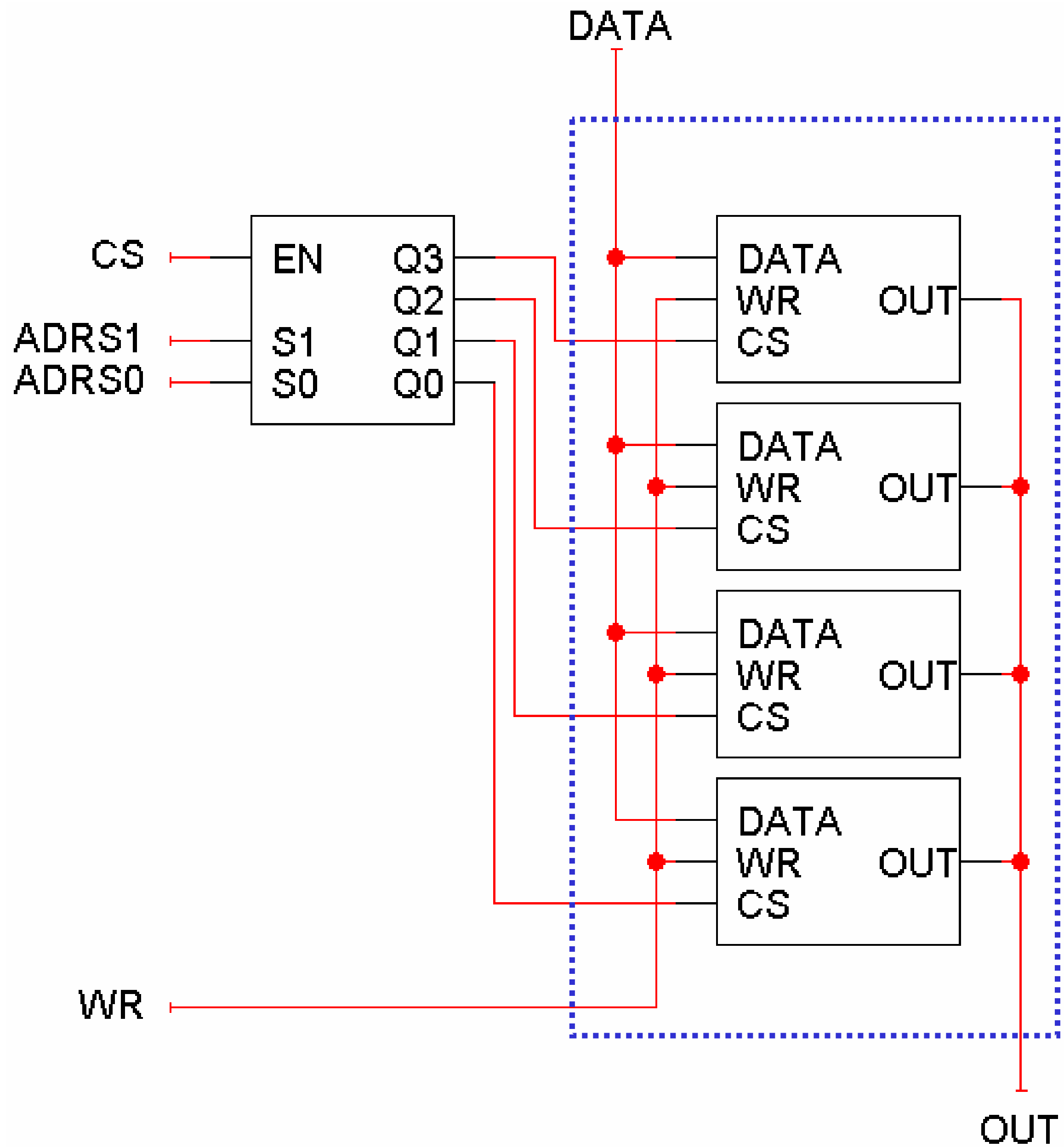
Connecting three-state buffers together

- You can connect the outputs of several three-state buffers together if you can guarantee that at most one of them is enabled at any time.
- The easiest way to do this is to use a decoder!
  - If the decoder is disabled, then all of the three-state buffers will appear to be disconnected, and OUT will also appear disconnected.
  - If the decoder is enabled, exactly one of its outputs will be true, so only one of the tri-state buffers will produce an output.
- The net result is we can save some wire and gate costs. We also get a little more flexibility in putting circuits together.



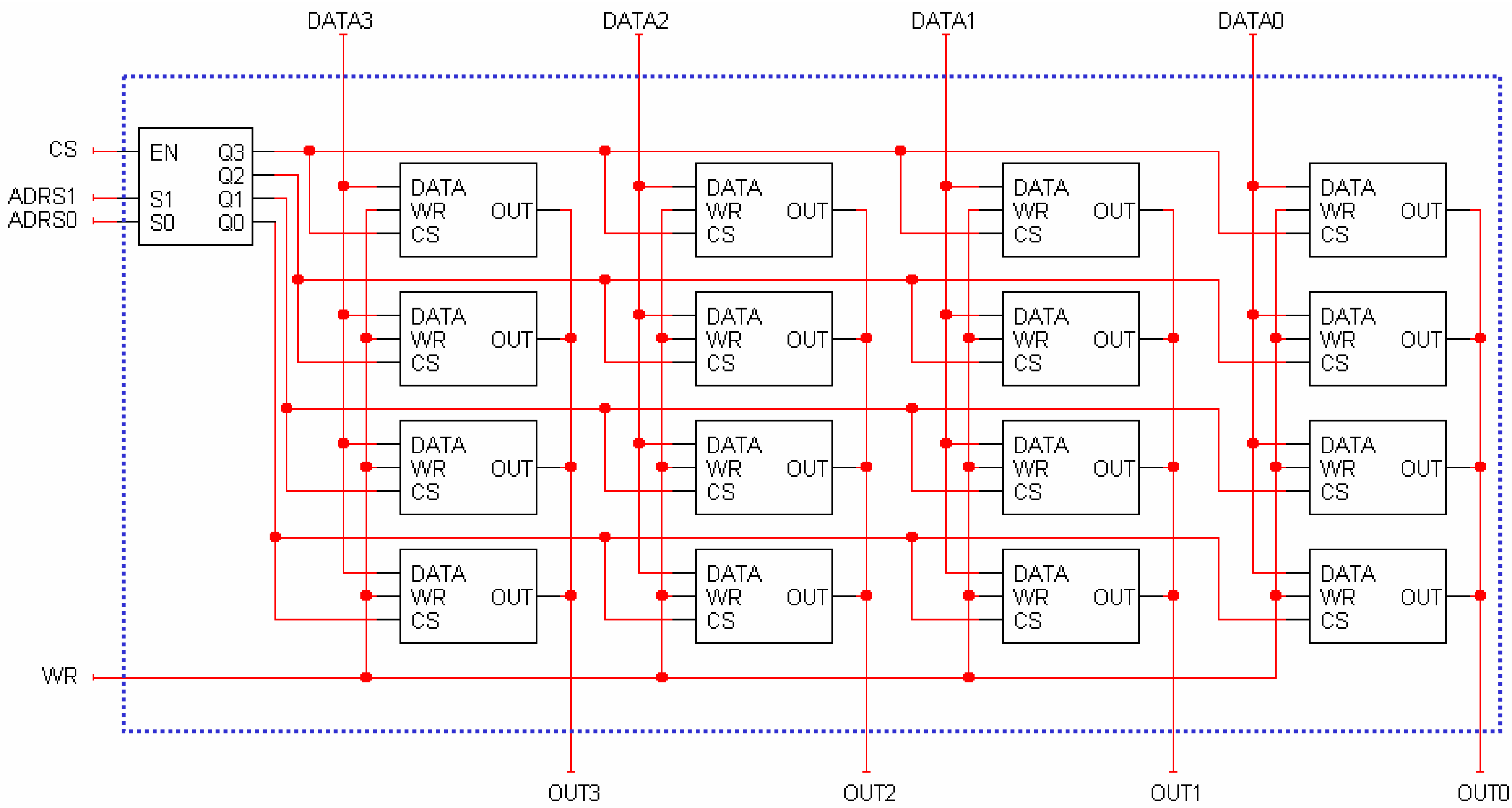
Bigger and better

- Here is the 4 x 1 RAM once again.
- Can we make a wider memory with more bits per word, like perhaps a 4 × 4 RAM?
- Duplicate the stuff in the blue box!



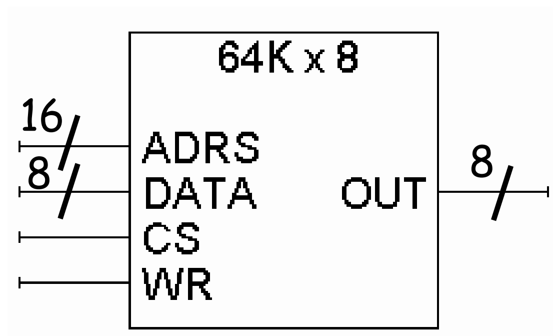
A 4 × 4 RAM

- DATA and OUT are now each *four* bits long, so you can read and write four-bit words.



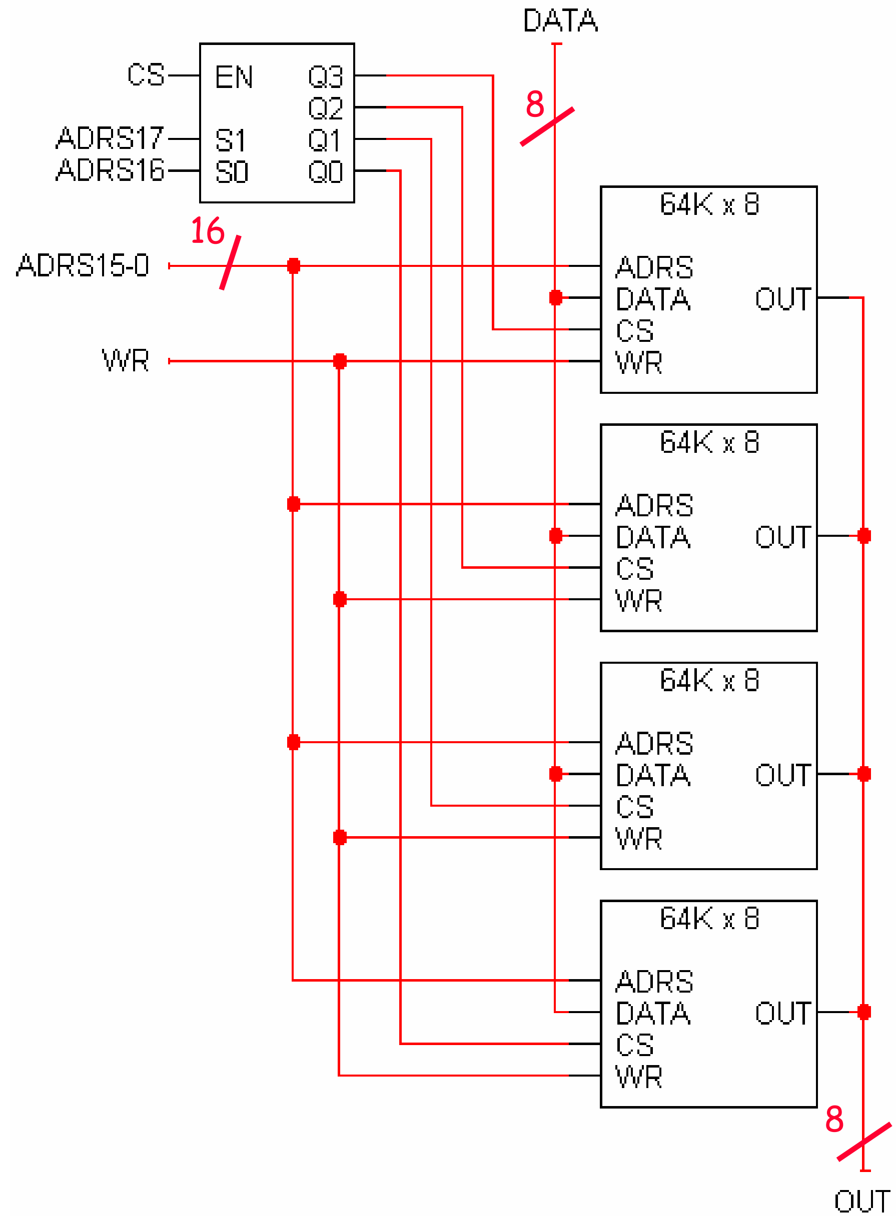
Bigger RAMs from smaller RAMs

- We can use small RAMs as building blocks for making larger memories, by following the same principles as in the previous examples.
- As an example, suppose we have some 64K × 8 RAMs to start with.
  - $64K = 2^6 \times 2^{10} = 2^{16}$, so there are 16 address lines.
  - There are 8 data lines.



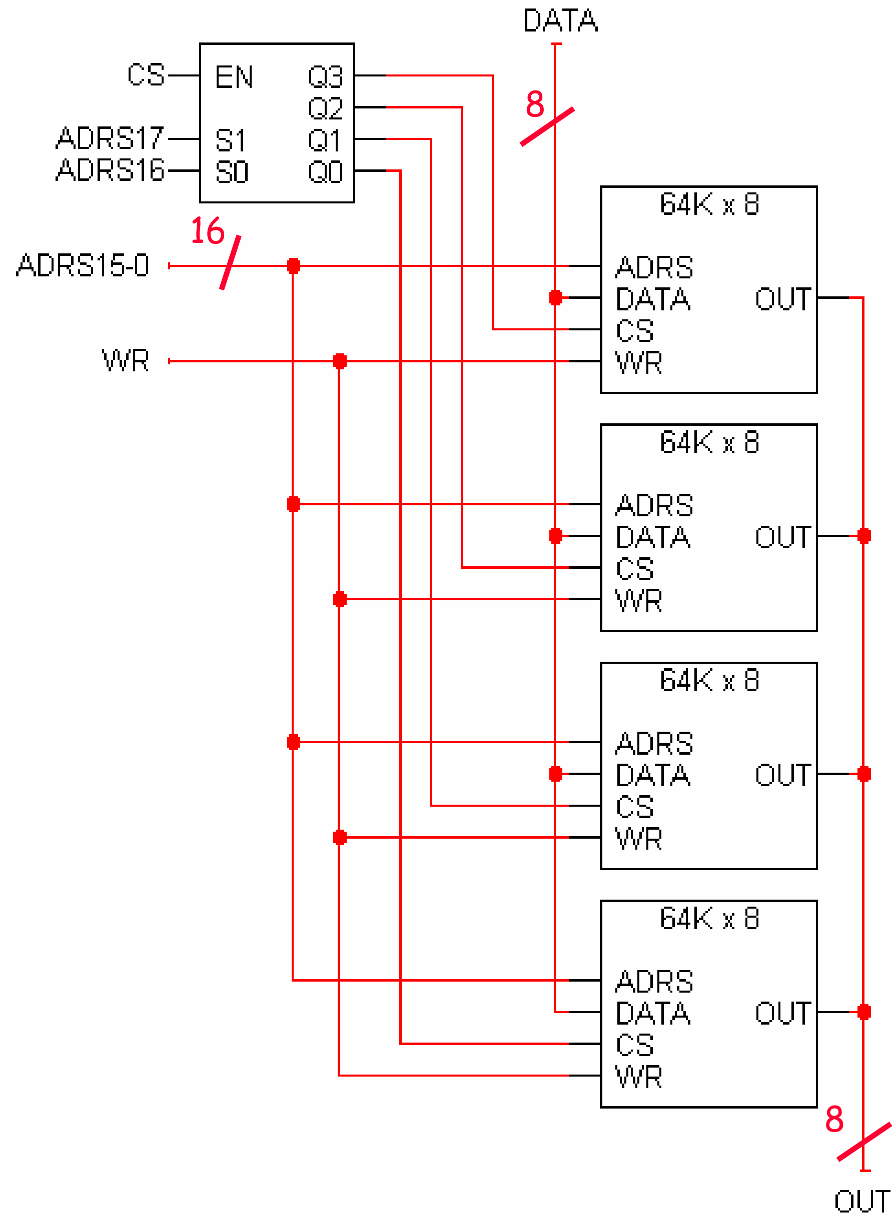
Making a larger memory

- Four 64K × 8 chips together form a 256K × 8 memory.
- There are 18 address lines in a RAM with 256K words.
  - The two most significant address lines go to the decoder, which selects one of the four 64K × 8 RAM chips.
  - The other 16 address lines are shared by the 64K × 8 chips.
- The 64K × 8 chips also share WR and DATA inputs.
- This assumes the 64K × 8 chip has three-state outputs.

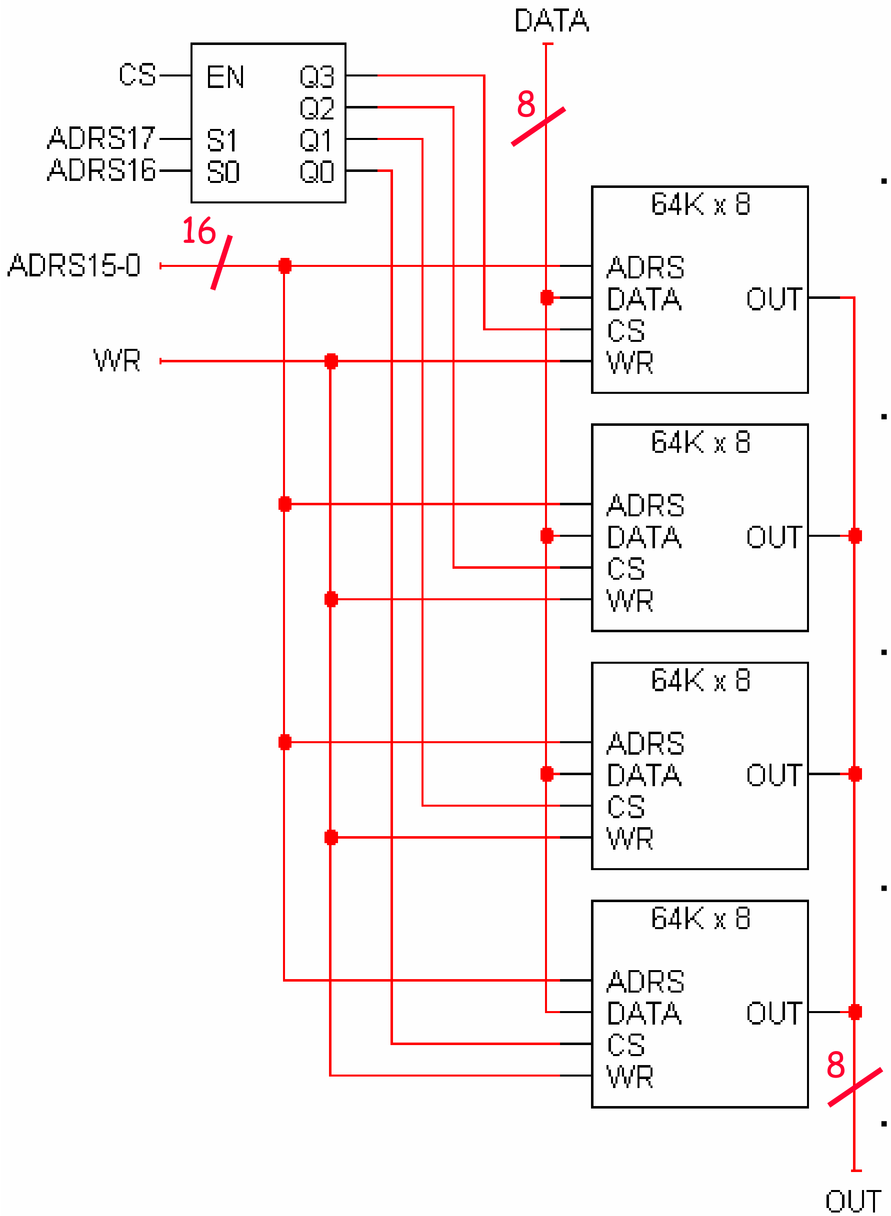


Analyzing the 256K × 8 RAM

- There are 256K words of memory, spread out among the four smaller 64K × 8 RAM chips.
- When the two most significant bits of the address are 00, the bottom RAM chip is selected. It holds data for the first 64K addresses.
- The next chip up is enabled when the address starts with 01. It holds data for the second 64K addresses.
- The third chip up holds data for the next 64K addresses.
- The final chip contains the data of the final 64K addresses.



Address ranges



11 1111 1111 1111 1111 (0x3ffff)
to
11 0000 0000 0000 0000 (0x30000)

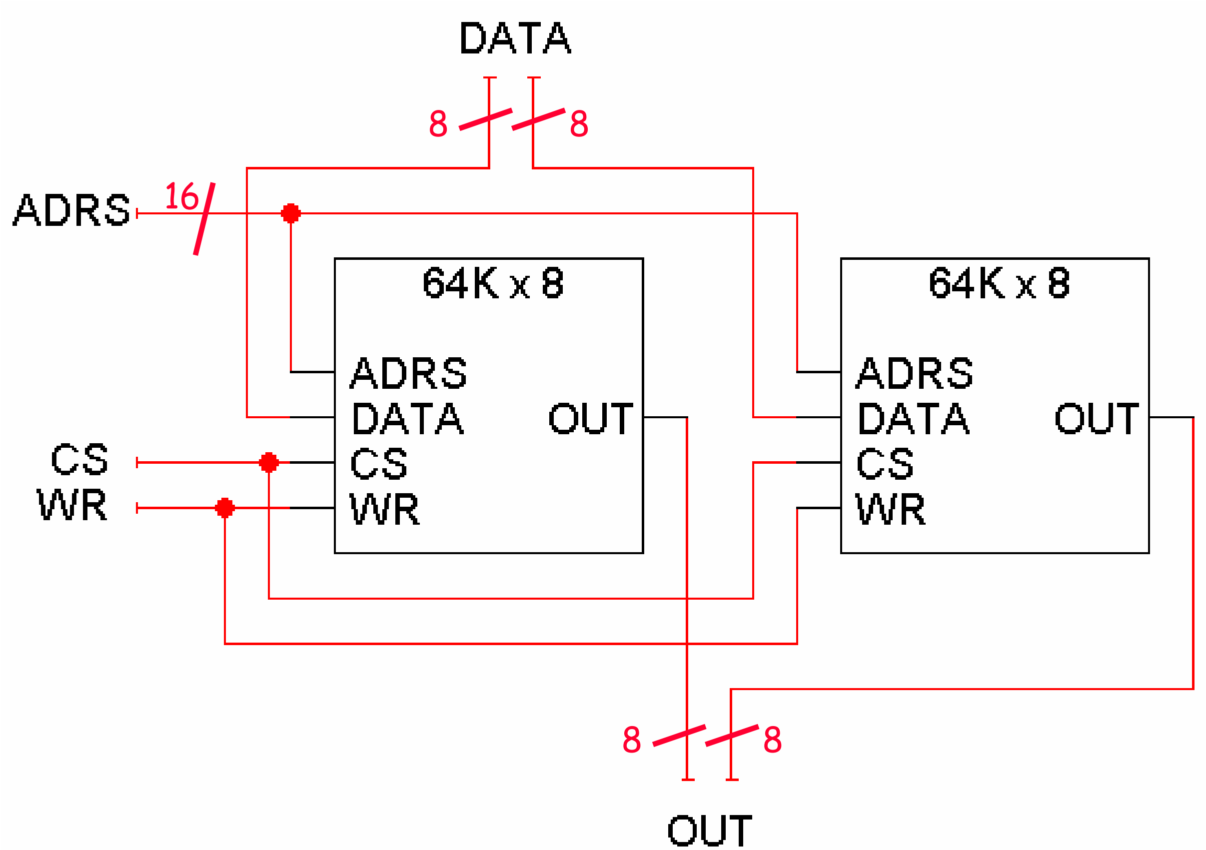
10 1111 1111 1111 1111 (0x2ffff)
to
10 0000 0000 0000 0000 (0x20000)

01 1111 1111 1111 1111 (0x1ffff)
to
01 0000 0000 0000 0000 (0x10000)

00 1111 1111 1111 1111 (0x0ffff)
to
00 0000 0000 0000 0000 (0x00000)

Making a wider memory

- You can also combine smaller chips to make wider memories, with the same number of addresses but more bits per word.
- Here is a 64K × 16 RAM, created from two 64K × 8 chips.
  - The left chip contains the most significant 8 bits of the data.
  - The right chip contains the lower 8 bits of the data.



Dynamic memory in a nutshell

- Dynamic memory is built with capacitors.
  - A stored charge on the capacitor represents a logical 1.
  - No charge represents a logic 0.
- However, all capacitors lose their charge after a few milliseconds. The memory requires constant refreshing to recharge the capacitors. (That's what's "dynamic" about it.)
- Dynamic RAMs tend to be physically smaller than static RAMs.
  - A single bit can be stored with just one capacitor and one transistor, while static RAM cells typically require 4-6 transistors.
  - So dynamic RAM is cheaper and denser—more bits can be stored in the same physical area.

SDRAM

- Synchronous DRAM, or SDRAM, is one of the more common types of PC memory now.
- Memory chips are organized into modules that are connected to the CPU via a 64-bit (8-byte) bus.
- Speeds are rated in megahertz: PC66, PC100 and PC133 memory run at 66MHz, 100MHz and 133MHz respectively.
- The maximum memory bandwidth is found by multiplying the number of transfers per second by the size of each transfer.
  - PC100 memory can transfer up to 800MB per second (100MHz × 8 bytes/cycle).
  - PC133 can get over 1 GB per second.
- Unfortunately, systems never achieve these maximum data rates in practice.

DDR-RAM

- A newer type of memory is Double Data Rate, or DDR-RAM.
- It's very similar to regular SDRAM, except data is transferred on both the positive *and* negative clock edges. So on 100-133MHz buses, the memory speeds appear to be 200-266MHz.
- This is known as DDR200 or DDR266 RAM, where the number now refers to the effective memory speed instead of the actual frequency.
- It's also sometimes confusingly called PC1600 and PC2100 RAM because of its bandwidth, and because larger numbers look better.
  - 200MHz × 8 bytes/cycle = 1600 MB/s
  - 266MHz × 8 bytes/cycle = 2100 MB/s.
- Newer systems can support DDR333 or even DDR400 memory.
- DDR-RAM has lower power consumption, using 2.5V instead of 3.3V like SDRAM. This makes it good for notebooks and other mobile devices.

RDRAM

- A type of memory called RDRAM is used in the Playstation 2 as well as some Pentium 4 computers.
- The data bus is only 16 bits wide, but the memory runs at 400-533MHz and data is transferred on both positive and negative clock edges.
- This works out to maximum transfer rates of 1.6 to 2.1GB per second.
- RDRAM-based machines also often use two "channels" of memory, or two modules in parallel, resulting in a 4.2 GB/s maximum bandwidth.

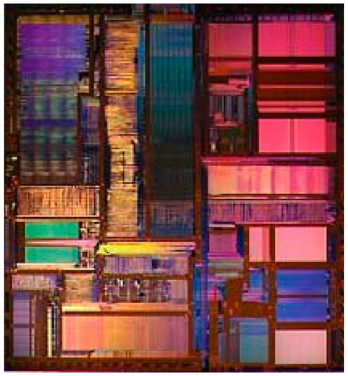
Dynamic vs. static memory

- In practice, dynamic RAM is used for a computer's main memory, since it's cheap and you can pack a lot of storage into a small space.
  - These days you can buy 512MB of memory for $60 or less.
  - You can also load a system with 1.5GB or more of memory.
- The disadvantage of dynamic RAM is its speed.
  - Memory transfer rates are still much slower than the processor itself, which can run at up to 3 GHz, or 3 billion cycles per second.
  - You also have to consider latency, or the time it takes data to travel from the memory modules to the processor.
- Real systems augment dynamic memory with small but fast sections of static memory called caches.
  - Typical processor caches range in size from 128 to 512 *kilobytes*.
  - That's small compared to a 128 *megabyte* main memory, but it's big enough to significantly increase a computer's overall speed.
  - We'll talk about caches a little more at the end of the summer, and you'll study them a lot more if you take CS232 or CS333.

ROMs vs. RAMs

- How do the ROMs that we mentioned earlier compare with RAMs?
  - ROMs are non-volatile and data is preserved even without power. On the other hand, RAM contents disappear once power is lost.
  - ROMs require special (and slower) techniques for writing, which is why they're considered to be read-only devices.
- Many newer types of ROMs do permit easier writing, although the speeds are still much slower than RAMs.
  - MP3 players, digital cameras and other toys use CompactFlash, Secure Digital, or MemoryStick cards for non-volatile storage.
  - Many devices also allow you to upgrade data stored in "flash ROM."

Summary



- Today we talked about static and dynamic random-access memory.
- Static RAM is just a bunch of latches connected together, allowing users to select a particular address to read or write.
  - Much of the hardware in memory chips supports the address selection process: chip select inputs, decoders and tri-state buffers.
  - By providing a uniform interface, RAM chips are easily joined to form longer and wider memories.
- Dynamic RAM is built from capacitors instead of latches.
  - They cost less and require less physical space, making them ideal for larger-capacity memories.
  - However, dynamic memories are much slower than static RAMs.