Memory Chips

Microprocessor systems use memory ICs to store programs and data. This lecture describes common semiconductor memory devices and how they are organized in microprocessor memory systems.
After this lecture you should be able to select the appropriate type of memory device for different applications, combine memory ICs to form memory arrays, and design address decoders using SSI decoders and PLDs.

Introduction

ROM

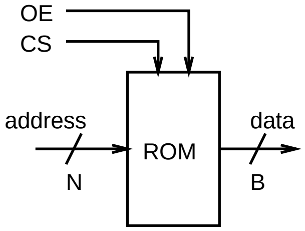
The simplest memory IC is a ROM (read-only memory). A ROM can be described as a combinational logic circuit that implements an arbitrary B-bit function of N bits. The N input bits are known as the *address* and the B output bits are the *data*. Such a device is referred to as a 2^N *by* B memory. For example, a 4096 by 8 (4kx8) ROM would have 12 address inputs and 8 data pins.

Like any combinational logic circuit, a ROM can be described using a lookup table. The following table shows some of the contents (in hexadecimal) of a hypothetical byte-wide device:

address	data
0000	2E
0001	A3
0002	73
....	..
FFFF	D9

Exercise: What are the values of N and B for this device? When the values of the address inputs are 0002 (hex) what will be the values (in binary) of the outputs D_0 to D_7?

The following diagram shows the input and output pins on a typical read-only memory (ROM):

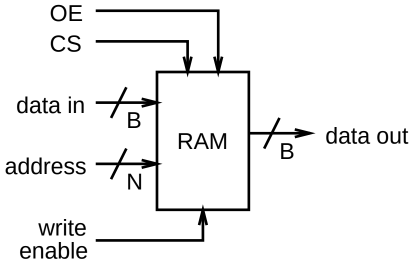


The address inputs are typically labeled A_0 to A_{N-1} and the data outputs D_0 to D_{B-1}. The CS (*chip select*) and OE (*output enable*) pins must be active for data to appear on the output.

Exercise: If we wanted to be able to connect the outputs of several memory chips in parallel, what state would the outputs have to be in when CS or OE were not asserted?

RAM

A RAM (random-access memory) chip is a memory that can be written as well as read. A RAM can be described as a sequential circuit in which N address inputs select one of 2^N sets of B flip-flops. The following diagram shows the essential pins on a (RAM):



During a *write* operation one set of B flip-flops is selected by the address pins and the data to be latched (stored) in the flip-flops in placed on the data input pins. The write enable pin is used to latch the data into the flip-flops in the same way that a clock is used in a D flip-flop.

During a *read* operation a particular set of flip-flops is again selected by the address pins and the data previously stored in the flip-flops appears on the data output pins. In many cases a bidirectional data pin is used for both data input and data output.

As with a ROM, CS must be asserted for either operation and the OE pin must be asserted during a read operation.

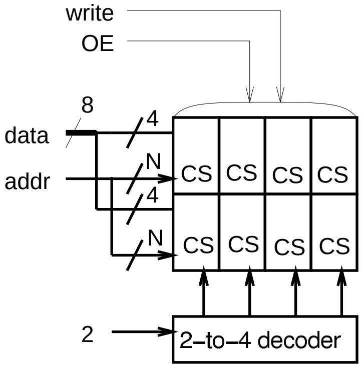
Internally, the address input is used to enable the clock input of a particular set of flip-flops during a write operation or to connect a particular set of flip-flop outputs (Q) to the data output during a read operation.

Exercise: Show how a 4-to-1 demultiplexer and a 4-to-1 multiplexer could be used to build a 4x1 RAM from four D flip-flops (ignore OE and CS).

Combining Memory Chips

If the width (number of bits available in parallel) of the individual memory devices is less than what is required by the CPU, devices can be combined in parallel by connecting their data bits to successive bits of the data bus. In addition, these B-bit wide blocks of devices can be combined to make available more memory than can be provided by one such set of ICs.

The following diagram shows an example of how memory chips can be combined to increase both the word size and the number of words available. In this example two four-bit-wide memories are combined to form an 8-bit-wide memory and four banks of chips are combined to form a 2^{N+2}-word array. The 2-to-4 decoder selects one of four banks (using their CS inputs) according to the value of the 2 extra address bits.



Exercise: Eight (8) 1Mx4 devices are to be connected to a CPU with a 16-bit data bus. How many address and data bits does each IC require? What is the total memory size in MBytes? Draw a block diagram showing the address and data bus connections to the different ICs. How many enables (used to drive CS pins) will be required from the address decoder?

Address Decoding

The memory space of a computer system can be considered to be an array of $2^{N_{cpu}}$ bytes where N_{cpu} is the number of CPU address bits. The memory (and possibly the i/o devices) must fit within this address space.

Typically N_{cpu} is larger than the N for a memory device. For example, the 8088 has $N_{cpu} = 20$ address pins but we might want to use 32kB ($N = 15$) RAMs.

Combinational circuits called *address decoders* are used to enable the appropriate set of memory devices when the address on the CPU bus lies in the desired range of 2^N addresses. The decoders' inputs are $N_{cpu} - N$ most-significant CPU bus address bits and its outputs are logic signals that enable the chip(s) when the address falls within the desired address range. The two most common ways to implement address decoders are SSI (small scale integration) decoders and PLDs.

Address decoders invariably decode regions that start on an address that is a power-of-two. The decoded region is invariably also a power-of-two. Therefore the address range can be written as a bit pattern that the decoder responds to. For example, a decoder for addresses from 2 0000H to 2 7FFFH ($N = 15$, 32 kBytes) would respond to addresses of the form 0010 0XXX XXXX XXXX XXXX where the X's are "don't cares."

Exercise: If a decoded region spans 4 kBytes starting at address 1 0000H, what pattern of addresses will the memory respond to?

Decoder designs can be simplified by allowing the decoder to respond to multiple addresses. This is called *partial decoding*. In this case the decoder uses fewer than $N_{cpu} - N$ bits. For example, if the above decoder responded to addresses of the form 000X XXXX XXXX XXXX XXXX the decoded region would extend from 0 0000H to 1 FFFFH (128kB) and the 32kB region would appear replicated in all four 32k blocks in that region. This partial decoding "wastes" part of the processor's address space because it is now unavailable for other devices. The advantage is that the extra "don't-care" bits leads to a simpler implementation for the decoder.

SSI Decoders

A decoder such as the 74LS138 3-to-8 decoder can be used to divide up an address range.

Exercise: How could a '138 be used to divide up the 8088's 20-bit (1 MByte) address space into eight 128-kByte regions?

PLD Decoders

Since PLDs can generate complex combinational functions they can be used to divide up a memory space into regions of different sizes.

Exercise: We want to design a PAL decoder that selects one 64kB region from 0 0000H to 0 FFFFH and one 256 kB region from C 0000 to F FFFFH out of the 8088's address space. How may input bits are required? How many outputs? Write the VHDL expressions for signals `sel1` and `sel2` assuming the address bits are declared as `a : in bit_vector (19 downto 0) ;`.

Memory Technologies

A wide variety of IC memory devices are available. They vary in terms of data permanence, power consumption, cost, capacity, and access time.

SRAM

Static RAM is volatile read/write memory. Data is stored as the state of a flip-flop. The contents are lost when power is removed. CMOS devices have very low power consumption when not being accessed and can be used with battery or capacitor backup. Bipolar devices have higher power consumption but feature the shortest access times.

DRAM

In dynamic RAM the data is stored as the charge on a capacitor. These devices have the lowest cost per bit for RAM. The contents are lost if memory locations are not accessed (refreshed) every few milliseconds. Capacity about 4 times that of same-generation SRAM. 16 Mbit 60ns devices currently in volume production.

Mask-Programmed ROM

Non-volatile read-only memory. Data stored as connections between gates. Memory contents determined at time of manufacture. Lowest cost per bit of any memory but have large NRE costs so only suitable for large volume applications.

EPROM

Field-programmable non-volatile ROM that can be erased by exposure to UV light. Data stored as charge on "floating gates." Typically have byte-wide organization. Not as fast or dense as RAM. OTP devices are less expensive since the packages don't have windows (and can't be erased).

EEPROM

Non-volatile memory that can be written like RAMs. Write cycles are relatively long (hundreds of microseconds). Relatively small capacity. Limited to several thousand write cycles. Mostly used to store infrequently-changed configuration information.

Flash EEPROMs are similar to EEPROMs but have larger capacity. Whole sections of the chip must be erased and then re-written.

Specialized Memories

Video RAM is used in video display circuits. They have a second independent data output used that sequentially reads out a complete row of data from the memory array. This "dual ported" arrangement allows both the CPU and the video signal generator to access the RAM at the same time. This reduces contention and improves performance.

Serial EEPROMs are EEPROMs whose contents must be read or written one bit at a time from start to end. The serial interface reduces chip size, pin count and cost (e.g. a 128x8 EEPROM in an 8-pin DIP for $1). Typical application is storing configuration information. The slow serial access is not a drawback since the device is typically used only when the product is turned on (or off).

Exercise: What type of memory device(s) would likely be used in a popular PC for the following purposes: storage for the power-on boot code? the "CMOS" configuration memory? the main data/program RAM? the main memory cache? the boot program in a prototype of this PC? the video display memory?

What type of memory device(s) would likely be used for the following applications: non-volatile memory for user programs in a calculator? a font card for a laser printer? user-upgradeable storage for the firmware in modem? the mileage reading in a digital car odometer? a video game cartridge?