

# How ROM Works

by [Jeff Tyson](#)

**Read-only memory** (ROM), also known as **firmware**, is an integrated circuit programmed with specific data when it is manufactured. ROM chips are used not only in [computers](#), but in most other electronic items as well. In this edition of [HowStuffWorks](#), you will learn about the different types of ROM and how each works. This article is one in a series of articles dealing with computer memory, including:

- [How Computer Memory Works](#)
- [How RAM Works](#)
- [How Virtual Memory Works](#)
- [How Flash Memory Works](#)
- [How BIOS Works](#)

Let's start by identifying the different types of ROM.

## ROM Types

There are five basic ROM types:

- **ROM**
- **PROM**
- **EPROM**
- **EEPROM**
- **Flash memory**

Each type has unique characteristics, which you'll learn about in this article, but they are all types of [memory](#) with two things in common:

- Data stored in these chips is **nonvolatile** -- it is not lost when power is removed.
- Data stored in these chips is either **unchangeable** or requires a special operation to change (unlike [RAM](#), which can be changed as easily as it is read).

This means that removing the power source from the chip will not cause it to lose any data.

## ROM at Work

Similar to [RAM](#), ROM chips (Figure 1) contain a grid of columns and rows. But where the columns and rows intersect, ROM chips are fundamentally different from RAM chips. While RAM uses [transistors](#) to turn on or off access to a [capacitor](#) at each intersection, ROM uses a **diode** to connect the lines if the value is 1. If the value is 0, then the lines are not connected at all.



Figure 1. BIOS uses Flash memory, a type of ROM.

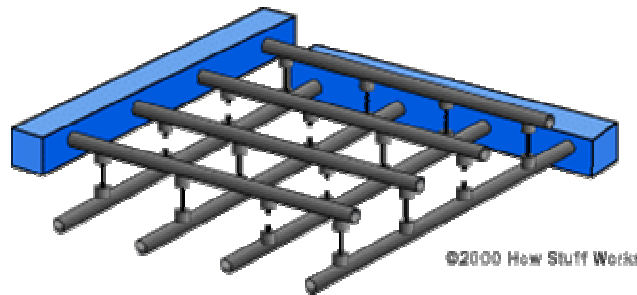
A [diode](#) normally allows current to flow in only one direction and has a certain threshold, known as the **forward breakover**, that determines how much current is required before the diode will pass it on. In silicon-based items such as [processors](#) and memory chips, the forward breakover voltage is approximately 0.6 volts. By taking advantage of the unique properties of a diode, a ROM chip can send a charge that is above the forward breakover down the appropriate column with the selected row grounded to connect at a specific cell. If a diode is present at that cell, the charge will be conducted through to the ground, and, under the [binary system](#), the cell will be read as being "on" (a value of 1). The neat part of ROM is that if the cell's value is 0, there is no diode at that intersection to connect the column and row. So the charge on the column does not get transferred to the row.

As you can see, the way a ROM chip works necessitates the programming of perfect and complete data when the chip is created. You cannot reprogram or rewrite a standard ROM chip. If it is incorrect, or the data needs to be updated, you have to throw it away and start over. Creating the original template for a ROM chip is often a laborious process full of trial and error. But the benefits of ROM chips outweigh the drawbacks. Once the template is completed, the actual chips can cost as little as a few cents each. They use very little power, are extremely reliable and, in the case of most small electronic devices, contain all the necessary programming to control the device. A great example is the small chip in the [singing fish toy](#). This chip, about the size of your fingernail, contains the 30-second song clips in ROM and the control codes to synchronize the [motors](#) to the music.

## PROM

Creating ROM chips totally from scratch is time-consuming and very expensive in small quantities. For this reason, mainly, developers created a type of ROM known as **programmable read-only memory** (PROM). Blank PROM chips can be bought inexpensively and coded by anyone with a special tool called a **programmer**.

PROM chips (Figure 2) have a grid of columns and rows just as ordinary ROMs do. The difference is that every intersection of a column and row in a PROM chip has a **fuse** connecting them. A charge sent through a column will pass through the fuse in a cell to a grounded row indicating a value of 1. Since all the cells have a fuse, the initial (**blank**) state of a PROM chip is all 1s. To change the value of a cell to 0, you use a programmer to send a specific amount of current to the cell. The higher voltage breaks the connection between the column and row by **burning** out the fuse. This process is known as **burning the PROM**.



## PROM

Figure 2

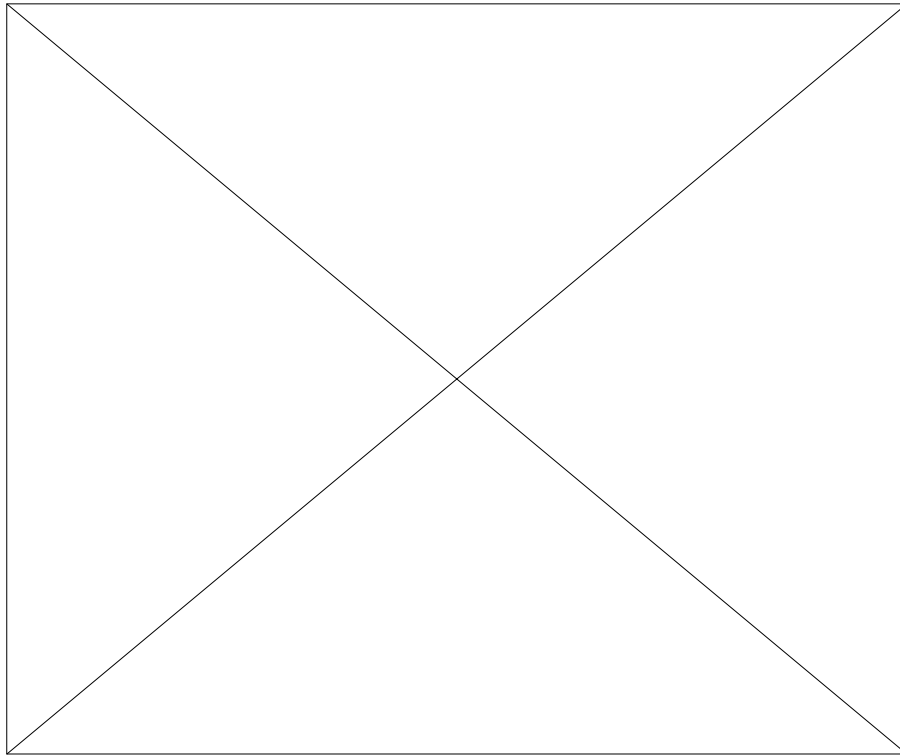
PROMs can only be programmed once. They are more fragile than ROMs. A jolt of static electricity can easily cause fuses in the PROM to burn out, changing essential [bits](#) from 1 to 0. But blank PROMs are inexpensive and are great for prototyping the data for a ROM before committing to the costly ROM fabrication process.

## EPROM

Working with ROMs and PROMs can be a wasteful business. Even though they are inexpensive per chip, the cost can add up over time. **Erasable programmable read-only memory** (EPROM) addresses this issue. EPROM chips can be rewritten many times. Erasing an EPROM requires a special tool that emits a certain frequency of [ultraviolet \(UV\) light](#). EPROMs are configured using an EPROM programmer that provides voltage at specified levels depending on the type of EPROM used.

Once again we have a grid of columns and rows. In an EPROM, the cell at each intersection has two transistors. The two transistors are separated from each other by a thin oxide layer. One of the transistors is known as the **floating gate** and the other as the **control gate**. The floating gate's only link to the row (**wordline**) is through the control gate. As long as this link is in place, the cell has a value of 1. To change the value to 0 requires a curious process called **Fowler-Nordheim tunneling**. **Tunneling** is used to alter the placement of electrons in the floating [gate](#). An electrical charge, usually 10 to 13 volts, is applied to the floating gate. The charge comes from the column (**bitline**), enters the floating gate and drains to a ground.

This charge causes the floating-gate transistor to act like an [electron gun](#). The excited electrons are pushed through and trapped on the other side of the thin oxide layer, giving it a negative charge. These negatively charged electrons act as a barrier between the control gate and the floating gate. A device called a **cell sensor** monitors the level of the charge passing through the floating gate. If the flow through the gate is greater than 50 percent of the charge, it has a value of 1. When the charge passing through drops below the 50-percent threshold, the value changes to 0. A blank EPROM has all of the gates fully open, giving each cell a value of 1.



**Figure 3**

To rewrite an EPROM, you must erase it first. To erase it, you must supply a level of energy strong enough to break through the negative electrons blocking the floating gate. In a standard EPROM, this is best accomplished with [UV light](#) at a frequency of 253.7. Because this particular frequency will not penetrate most plastics or glasses, each EPROM chip has a quartz window on top of it. The EPROM must be very close to the eraser's light source, within an inch or two, to work properly.

An EPROM eraser is not selective, it will erase the entire EPROM. The EPROM must be removed from the device it is in and placed under the UV light of the EPROM eraser for several minutes. An EPROM that is left under too long can become **over-erased**. In such a case, the EPROM's floating gates are charged to the point that they are unable to hold the electrons at all.

## EEPROMs and Flash Memory

Though EPROMs are a big step up from PROMs in terms of reusability, they still require dedicated equipment and a labor-intensive process to remove and reinstall them each time a change is necessary. Also, changes cannot be made incrementally to an EPROM; the whole chip must be erased. **Electrically erasable programmable read-only memory** (EEPROM) chips remove the biggest drawbacks of EPROMs.

In EEPROMs:

- The chip does not have to be removed to be rewritten.
- The entire chip does not have to be completely erased to change a specific portion of it.
- Changing the contents does not require additional dedicated equipment.

Instead of using UV light, you can return the electrons in the cells of an EEPROM to normal with the localized application of an **electric field** to each cell. This erases the targeted cells of the EEPROM, which can then be rewritten. EEPROMs are changed 1 [byte](#) at a time, which makes them versatile but slow. In fact, EEPROM chips are too slow to use in many products that make quick changes to the data stored on the chip.

Manufacturers responded to this limitation with **Flash memory**, a type of EEPROM that uses **in-circuit wiring** to erase by applying an electrical field to the entire chip or to predetermined sections of the chip called **blocks**. Flash memory works much faster than traditional EEPROMs because it writes data in chunks, usually 512 bytes in size, instead of 1 byte at a time. See [How Flash Memory Works](#) to learn more about this type of ROM and its applications.