WIKIPEDIA

Data buffer

In computer science, a **data buffer** (or just **buffer**) is a region of a physical memory storage used to temporarily store data while it is being moved from one place to another. Typically, the data is stored in a buffer as it is retrieved from an input device (such as a microphone) or just before it is sent to an output device (such as speakers). However, a buffer may be used when moving data between processes within a computer. This is comparable to buffers in telecommunication. Buffers can be implemented in a fixed memory location in hardware—or by using a virtual data buffer in software, pointing at a location in the physical memory. In all cases, the data stored in a data buffer are stored on a physical storage medium. A majority of buffers are implemented in software, which typically use the faster RAM to store temporary data, due to the much faster access time compared with hard disk drives. Buffers are typically used when there is a difference between the rate at which data is received and the rate at which it can be processed, or in the case that these rates are variable, for example in a printer spooler or in online video streaming.

A buffer often adjusts timing by implementing a queue (or FIFO) algorithm in memory, simultaneously writing data into the queue at one rate and reading it at another rate.

Contents

Applications

Buffers are often used in conjunction with I/O to hardware, such as disk drives, sending or receiving data to or from a network, or playing sound on a speaker. A line to a rollercoaster in an amusement park shares many similarities. People who ride the coaster come in at an unknown and often variable pace, but the roller coaster will be able to load people in bursts (as a coaster arrives and is loaded). The queue area acts as a buffer—a temporary space where those wishing to ride wait until the ride is available. Buffers are usually used in a FIFO (first in, first out) method, outputting data in the order it arrived.

Buffers can increase application performance by allowing synchronous operations such as file reads or writes to complete quickly instead of blocking while waiting for hardware interrupts to access a physical disk subsystem; instead, an operating system can immediately return a successful result from an API call, allowing an application to continue processing while the kernel completes the disk operation in the background. Further benefits can be achieved if the application is reading or writing small blocks of data that do not correspond to the block size of the disk subsystem, allowing a buffer to be used to aggregate many smaller read or write operations into block sizes that are more efficient for the disk subsystem, or in the case of a read, sometimes to completely avoid having to physically access a disk.

Telecommunication buffer

A buffer routine or storage medium used in telecommunications compensates for a difference in rate of flow of data, or time of occurrence of events, when transferring data from one device to another.

Buffers are used for many purposes, including:

- Interconnecting two digital circuits operating at different rates,
- Holding data for later use,
- Allowing timing corrections to be made on a data stream,
- Collecting binary data bits into groups that can then be operated on as a unit,
- Delaying the transit time of a signal in order to allow other operations to occur.

Buffer versus cache

A cache often also acts as a buffer, and vice versa. However, caches operate on the premise that the same data will be read from them multiple times, that written data will soon be read, or that there is a good chance of multiple reads or writes to combine to form a single larger block. Their sole purpose is to reduce accesses to the underlying slower storage. A cache is also usually an abstraction layer that is designed to be invisible.

A "Disk Cache" or "File Cache" keeps statistics on the data contained within it and commits data within a time-out period in write-back modes. A buffer does none of this.

A buffer is primarily used for input, output, and sometimes *very temporary* storage of data that is either en route between other media or data that may be modified in a non-sequential manner before it is written (or read) in a sequential manner.

Examples

- The BUFFERS command/statement in CONFIG.SYS of DOS.
- The buffer between a serial port (UART) and a MODEM. The COM port speed may be 38400 bit/s while the MODEM may have only a 14400 bit/s carrier.
- The integrated buffer on a Hard Disk Drive, Printer or other piece of hardware.
- The Framebuffer on a video card.

History

An early mention of a print buffer is the Outscriber devised by image processing pioneer Russel A. Kirsch for the SEAC computer in 1952:[1]

> One of the most important problems in the design of automatic digital computers is that of getting the calculated results out of the machine rapidly enough to avoid delaying the further progress of the calculations. In many of the problems to which a general-purpose computer is applied the amount of output data is relatively big —so big that serious inefficiency would result from forcing the computer to wait for these data to be typed on existing printing devices. This difficulty has been solved in the SEAC by providing magnetic recording devices as output units. These devices are able to receive information from the machine at rates up to 100 times as fast as an electric typewriter can be operated. Thus, better efficiency is achieved in recording the output data; transcription can be made later from the magnetic recording device to a printing device without tying up the main computer.

See also

- Bucket (computing)
- Buffer overflow
- Buffer underrun
- Circular buffer
- Disk buffer
- Streaming media
- Frame buffer for use in graphical display
- Double buffering and Triple buffering for techniques mainly in graphics
- Depth buffer, Stencil buffer, for different parts of image information
- Variable length buffer
- Optical buffer
- MissingNo., the result of buffer data not being cleared properly in Pokémon Red and Blue
- UART buffer
- ENOBUFS, POSIX error caused by lack of memory in buffers
- Write buffer, a type of memory buffer
- 512k day

References

1. "SEAC Maintenance Manual: The Outscriber" (https://www.nist.gov/msidlibrary/doc/kirsch_1953_nbs2794.pdf) (PDF). *National Bureau of Standards Report*. **2794**. July 1953.

Retrieved from "https://en.wikipedia.org/w/index.php?title=Data_buffer&oldid=830751215"

This page was last edited on 16 March 2018, at 18:36.