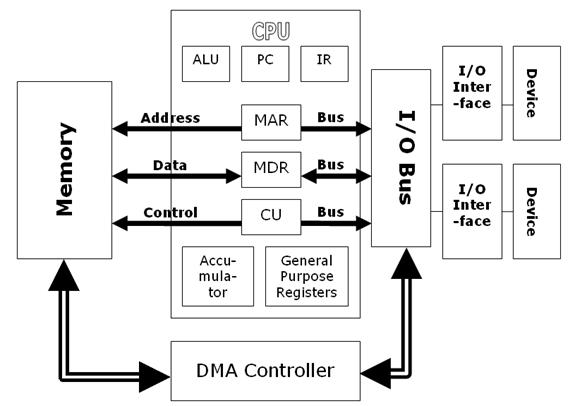
**M2 - Compare the roles played by different types of memory**

**Introduction**

In this report, I will be comparing the different types of memories that is used within a computer. I will say for each one how they work. The types of memory that I will name is the following:

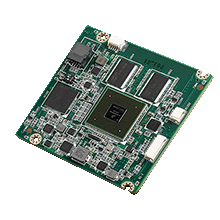
* Direct Memory Access (DMA)
* ROM (Read Only Memory)
* Cache
* RAM (Random Access Memory) e.g. static, dynamic, flash
* I/O Maps

**Direct Memory Access (DMA)**

DMA is a feature that allows the user to gain access to the main memory, and by being independently, the CPU cannot do it all itself. DMA allows the user to interpret, and send any attached device from the peripheral (output) to the memory on the computer’s motherboard. These files can be large, and using DMA, it is beneficial in this sense. As you can see on the picture (left), it shows how the DMA communicates. It communicates with the I/O bus. Each of the I/O bus is connected with a device. This enables communication between the two.

**How it works?**

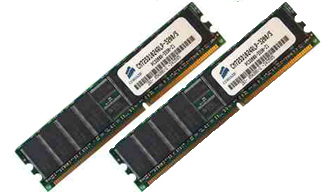
DMA works by allowing the user to send data from one location to another. They are two controllers for the DMA controller. The first one uses from 0 to 3 channels and this transfers only one byte in each transfer. The other one is 4 to 7 and it transfers 16-bits in each transfer.

**ROM**

ROM is known as read only memory. This is where data that is stored on this memory, but this cannot be changed. This is much different to RAM. ROM is where data is stored permanently on to this chip. As well as storing data, this gives out instructions.

**How it works?**

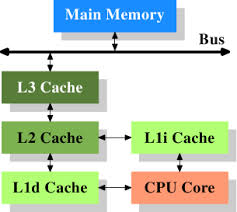
It is quite simple. It works by using a diode to connect only a value of 1. The lines are used to connect the value of 1, but it will not connect with the value of 0, as there is no lines for it to connect. A diode normally allows the data to flow in to one direction. This gets passed onto the binary system, where only “on” (1 value) can be passed on. This is why a 0 value cannot be passed through it. If the data needs updating, the process has to be started over again.

**RAM**

RAM is known as random access memory is used to store data. This is inserted on the motherboard, which it is located there. When the computer is switched on, the files that are stored on the RAM automatically be operating as long as the computer is running. All the documents that you create is stored on the RAM.

**How it works?**

RAM uses transistor and capacity for this process to work. A transistor acts like a switch for the capacity to change state. Changing state would be from either 0 or 1. This is the information that the capacitor holds. They both help each other out for RAM to work. This is important for the RAM to work. However, one disadvantage is that once the system is completely off, which means the power is turned off, the data is lost. Therefore, it is important to save all the work before the system is shut down.

[](http://www.google.co.uk/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=0CAcQjRw&url=http://lwn.net/Articles/252125/&ei=WMpgVNz3O7OKsQSX4oF4&bvm=bv.79189006,d.ZGU&psig=AFQjCNH5aVgkPLaBA5JB3gkkZvBzePOm8A&ust=1415715794084888)

**Cache**

A cache is a content that enables the request of the specific site to be located faster. It gets saved, but the computer does not know. This is to improve the input/output of the system. All these saved cache can be cleared, but when the I/O is requested, it will be slow.

**How it works?**

Referring to the image, each cache has different levels. Most computers uses L1 and L2, but nowadays, they use L3 as well. This is very important part as the data gets thrown around from then CPU to the memory bus.

**I/O maps**

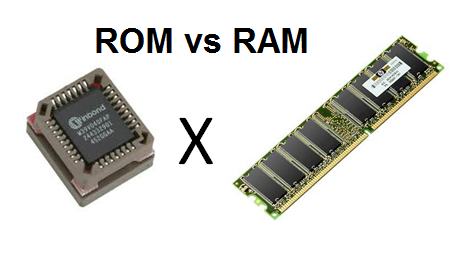
An ‘I/O’ stands for Input/Output. I/O maps is a process between a processor and a peripheral. An I/O mapped is assigned to one or more of the processor’s I/O maps and they transfer information between the processor and peripheral using the input and output instructions. This means, they send it when a user sends an input. E.g. clicks the program to close, the expectation is to close the program. They do this by using the I/O maps and communicating between the processor and the peripheral.

**How it works?**

This works by the device that is used. For example, if I were to be using a mouse, the CPU would send instructions to move the mouse. The CPU gives commands for the I/O to work.

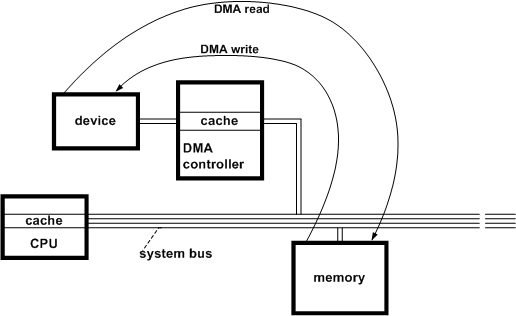
**Comparing RAM vs. ROM**

* ROM’s content is not lost when the power is off, and can be retained. However, RAM’s content is lost, when the power is off. It is advised to save it before the power is off.
* ROM’s information is permanent, and cannot be changed unless the whole procedure of it repeats. However, RAM’s memory is used to store information that can be changed, but it is only temporary i.e. it would be deleted.
* The similarity for both of them is that they are both contained within the system, and they both allow the user to store data.
* RAM is expensive and ROM is cheap. RAM costs around £40, and ROM is around £0 to £10.
* Both of them allow the user to read the information that they have stored
* Both can be accessed, but the ROM is slightly harder to get.
* They both have different types. RAM has Dynamic and Static RAM, whereas ROM has EEPROM, EPROM, and PROM. Each of them is different.



**Comparing Cache vs. DMA**

* DMA is a complex procedure, and at times, it could be slow. However, cache is fast
* DMA transfers from one location to another. Cache only gets the data from various locations around the computer. Each request is datum (piece of data).
* The similarity is that they both either get data from one location to another, or they transfer it. It works the same, but the procedure (the way it happens) is different.



**Reference**

* <http://en.wikipedia.org/wiki/Cache_(computing)>
* <http://docs.freebsd.org/doc/2.1.7-RELEASE/usr/share/doc/handbook/handbook245.html>
* <http://computer.howstuffworks.com/ram.htm>
* <http://computer.howstuffworks.com/cache.htm>
* <http://www.computermemoryupgrade.net/how-computer-memory-works.html>
* <http://stackoverflow.com/questions/9544399/dma-vs-cache-difference>
* <http://www.diffen.com/difference/RAM_vs_ROM>