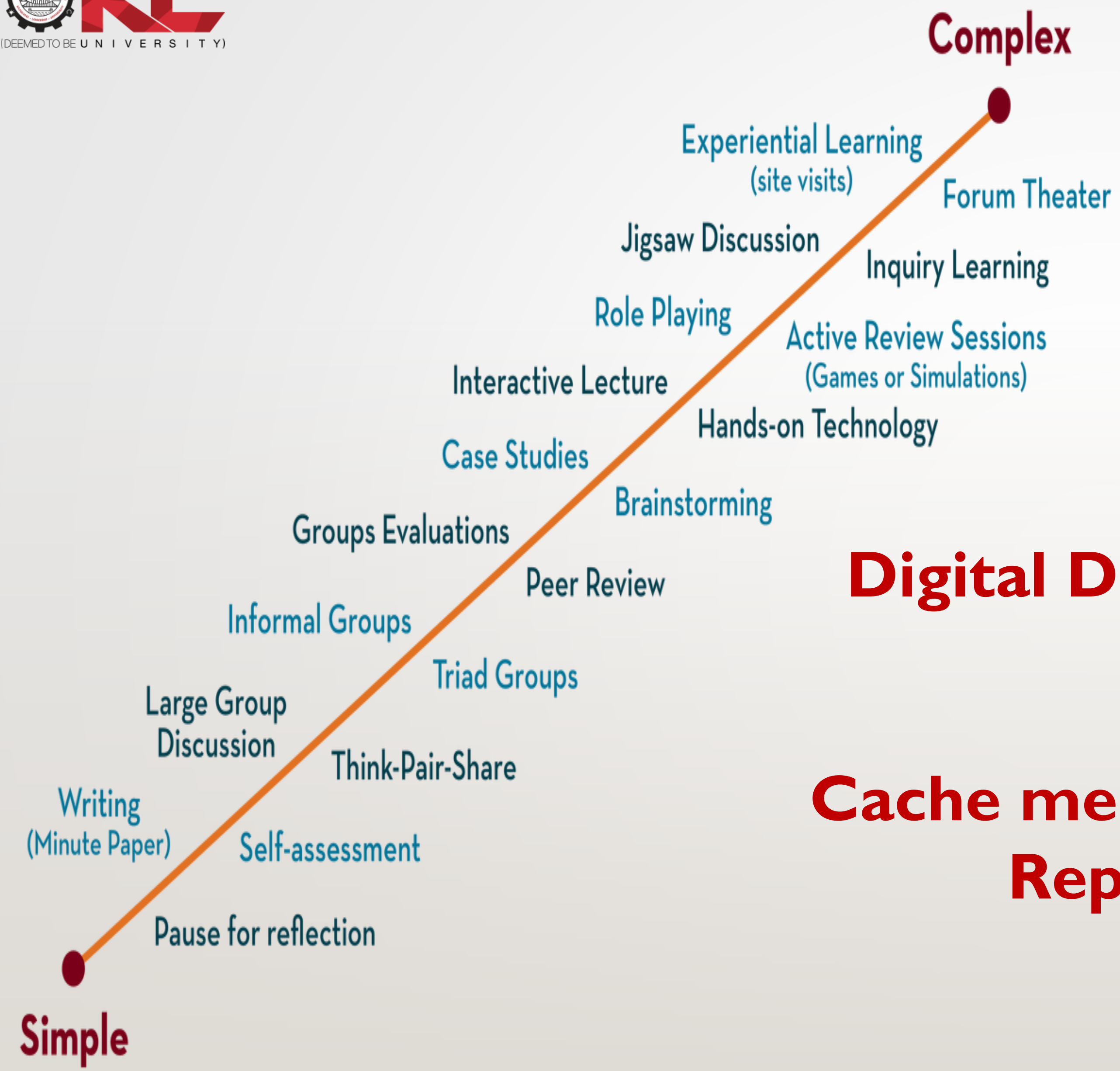


Department of BES-II

Digital Design and Computer Architecture
23EC1202

Topic:

Cache memory: Address mapping, Block size, Replacement, and Store policies

Session No: 34 & 35

AIM OF THE SESSION

To familiarize students with the basic concept of Cache memory and its importance in computer system

INSTRUCTIONAL OBJECTIVES

This Session is designed to:

1. Demonstrate the concept of cache memory & its types
2. Describe about various procedures of cache memory mapping
3. List out the replacement policies in cache memory
4. Describe the benefits, roles of cache memory

LEARNING OUTCOMES

At the end of this session, you should be able to:

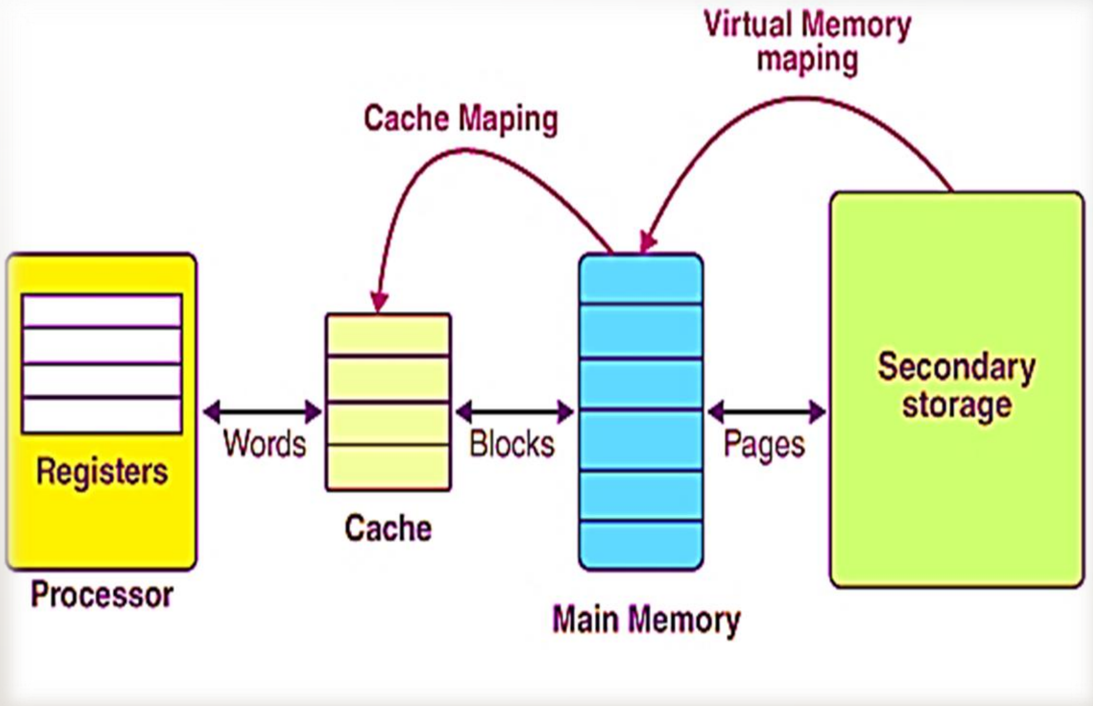
1. Define cache memory, types
2. Describe about various cache memory mapping techniques
3. Summarize the benefits & role of cache memory in computer system

Cache Memory

- Cache memory is a speedier, smaller section of memory with an access time that is comparable to registers. Cache memory has a shorter access time than primary memory in a memory hierarchy. Since cache memory is typically relatively little, it serves as a buffer.
- Primary memory access times are typically in the order of a few microseconds, while CPU operations can be completed in nanoseconds. Because of the delay between retrieving data and acting upon it, the system's performance degrades, and the CPU may sit idle for extended periods of time. A new memory section called cache memory is introduced to reduce this time gap.

Cache Memory (Cont..)

- Cache provides faster access.
- It acts as buffer between CPU and main memory (RAM).
- Cache primary role is to reduce the average time taken to access data, thereby improving overall system performance.



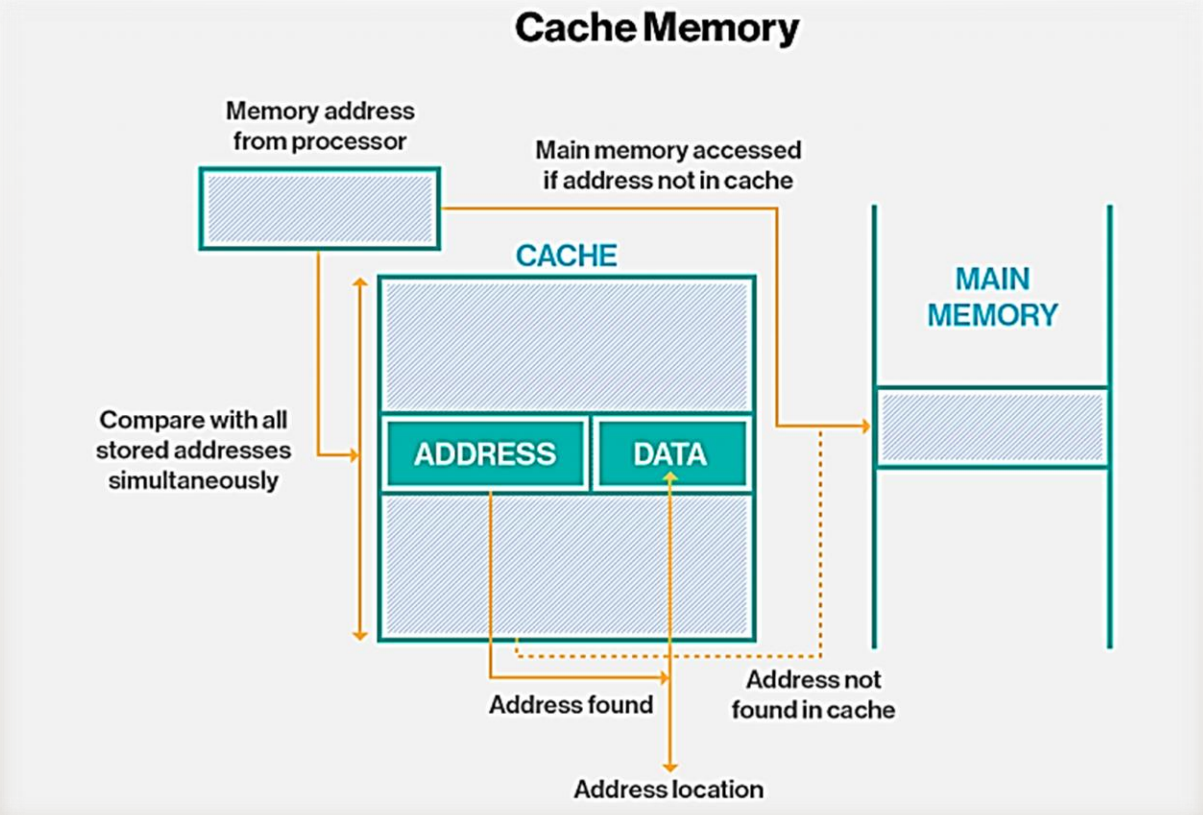
Benefits of Cache Memory

- **Faster access:** Faster than main memory. It resides closer to CPU, typically on same chip or in proximity.
- **Reducing memory latency:** Memory access latency refers to time taken for processes to retrieve data from memory.
- **Lowering bus traffic:** By utilizing cache memory, processor can reduce frequency of accessing main memory resulting in less bus traffic and improves system efficiency.
- **Increasing effective CPU utilization:** Cache memory allows CPU to operate at a higher effective speed. CPU can spend more time executing instruction rather than waiting for memory access.
- **Enhancing system scalability:** Cache memory helps improve system scalability by reducing impact of memory latency on overall system performance.

Working of cache

- Cache memory is faster but smaller capacity, a large amount of data cannot be stored.
- Whenever CPU needs any data, it searches for corresponding data in the cache (fast process) if data is found, it processes the data according to instructions, however, if data is not found in the cache CPU search for that data in primary memory (slower process) and loads it into the cache.
- On searching in the cache if data is found, a cache hit has occurred.
- On searching in the cache if data is not found, a cache miss has occurred.

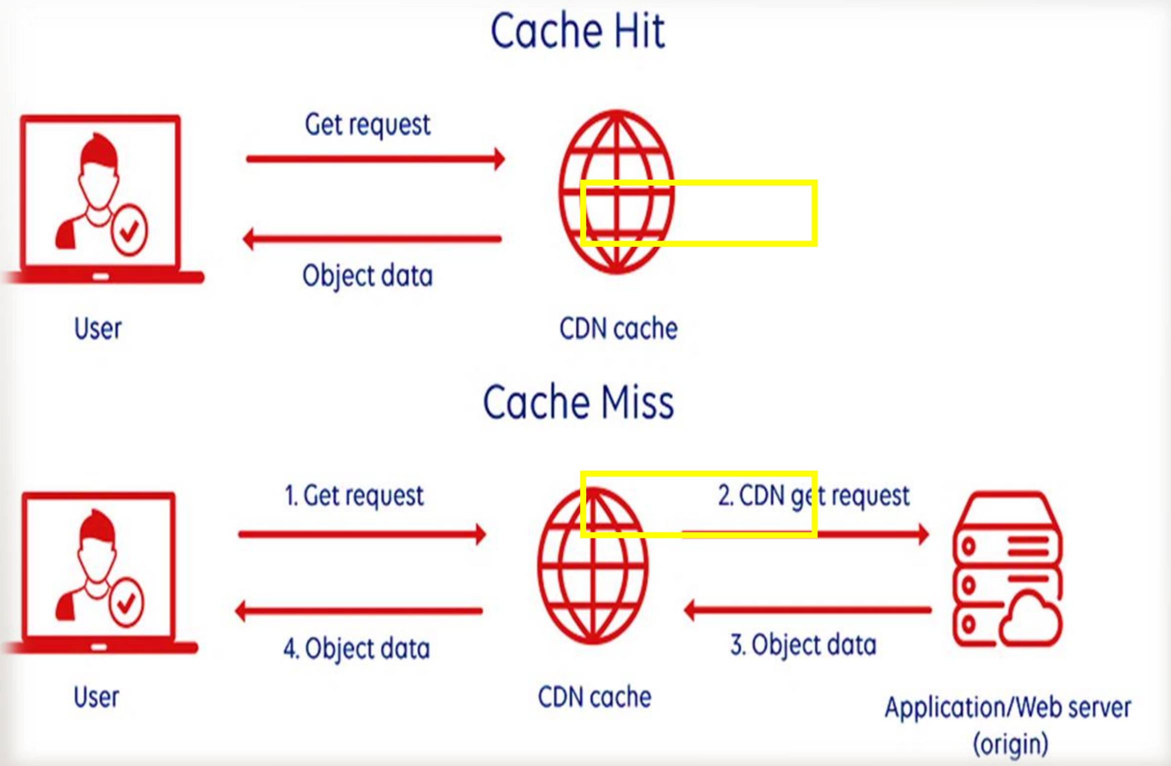
Working of cache


Cache Memory
Memory address from processor
Main memory accessed if address not in cache
CACHE
MAIN MEMORY
Compare with all stored addresses simultaneously
ADDRESS
DATA
Address found
Address not found in cache
Address location

Cache Performance

Performance of cache is measured by the number of cache hits to the number of searches. This parameter of measuring performance is known as the **Hit Ratio**.

Hit ratio = (Number of cache hits)/ (Number of searches)

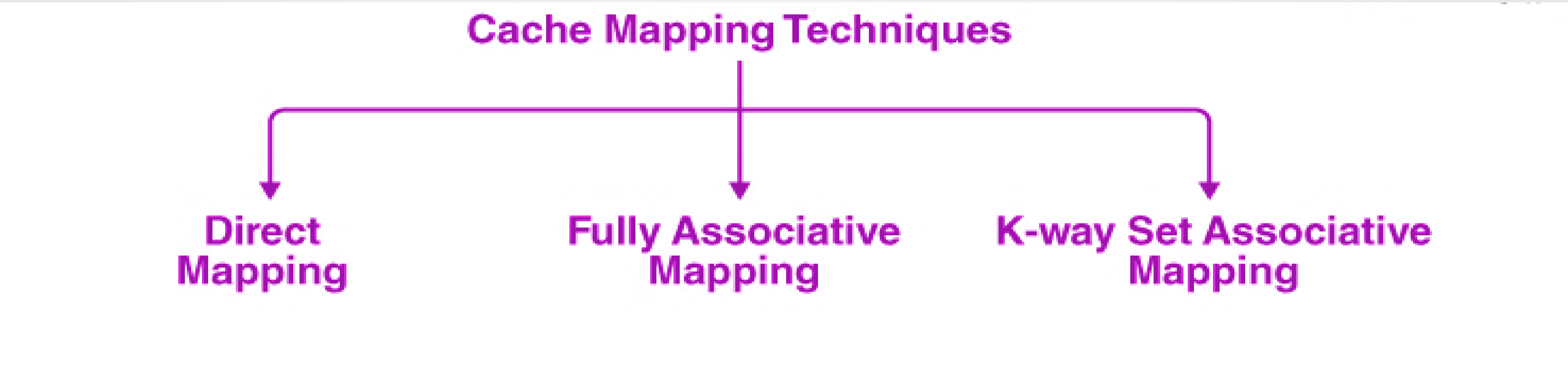


RAM Vs Cache

RAM	Cache
RAM is larger in size compared to cache. Memory ranges from 1MB to 16GB	The cache is smaller in size. Memory ranges from 2KB to a few MB generally.
It stores data that is currently processed by the processor.	It holds frequently accessed data.
OS interacts with secondary memory to get data to be stored in Primary Memory or RAM	OS interacts with primary memory to get data to be stored in Cache.
It is ensured that data in RAM are loaded before access to the CPU. This eliminates RAM miss.	CPU searches for data in Cache, if not found cache miss occur.

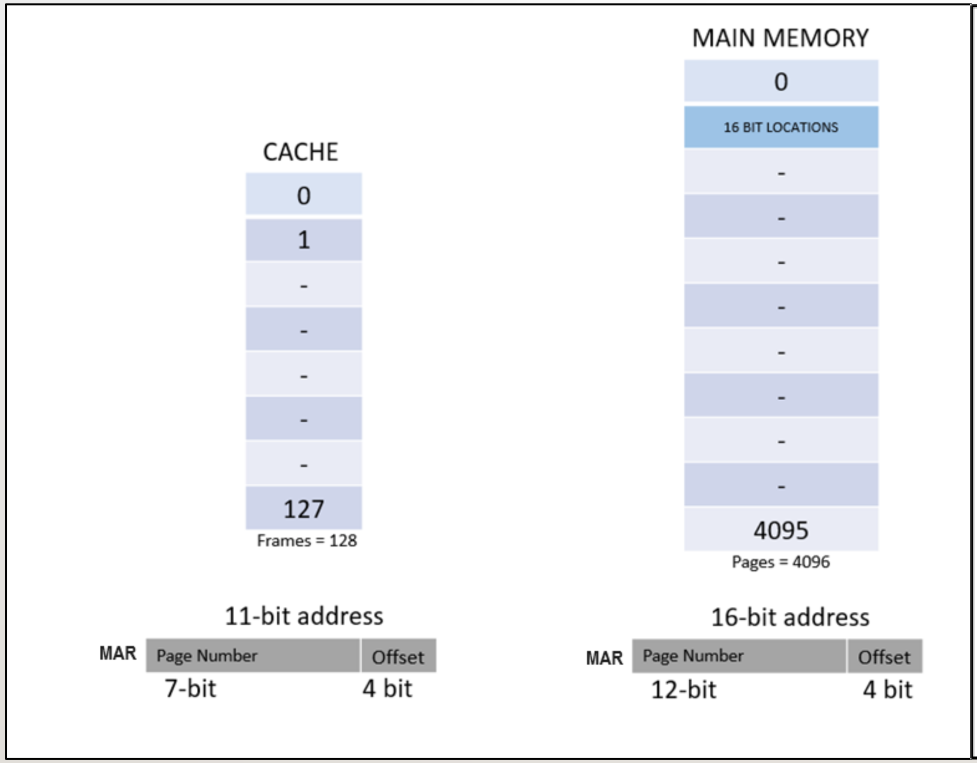
Cache Mapping

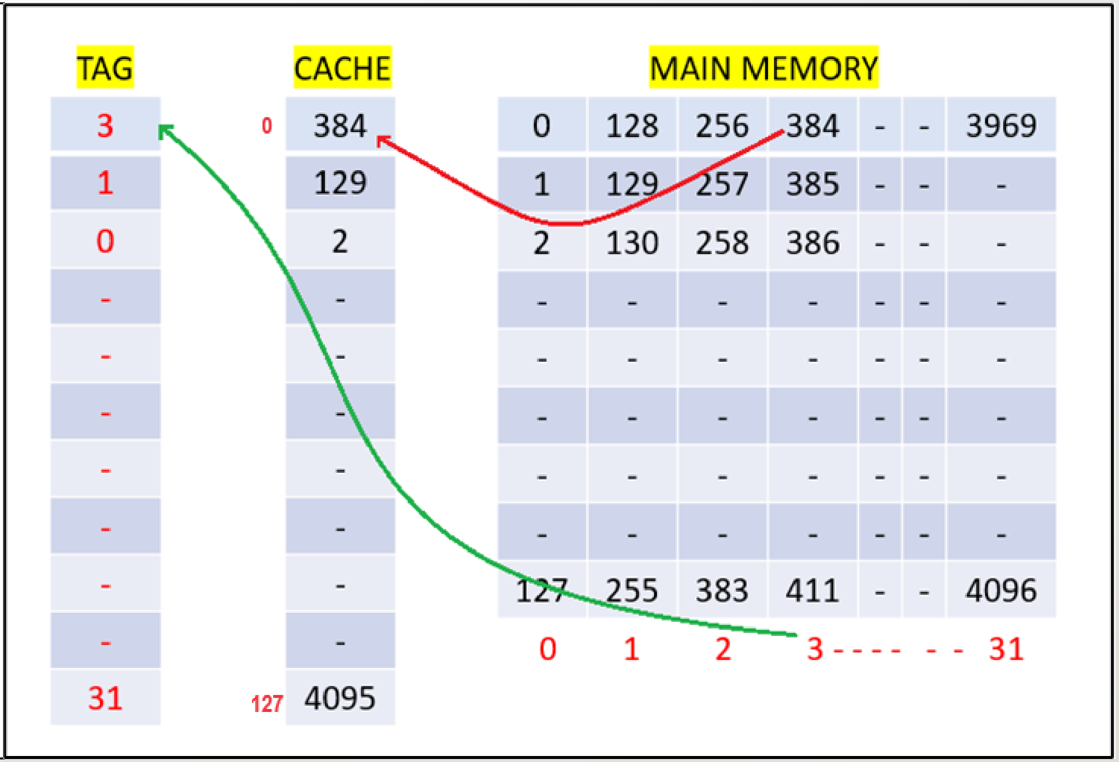
- **Cache mapping** refers to a technique using which the content present in the main memory is brought into the memory of the cache.
- Three distinct types of mapping are used for cache memory mapping



Direct Mapping

In direct mapping the main memory data is directly mapped to cache frame without any replacement algorithm.





TAG		CACHE
3	0	384
1		129
0		2
-		-
-		-
-		-
-		-
-		-
-		-
-		-
31	127	4095

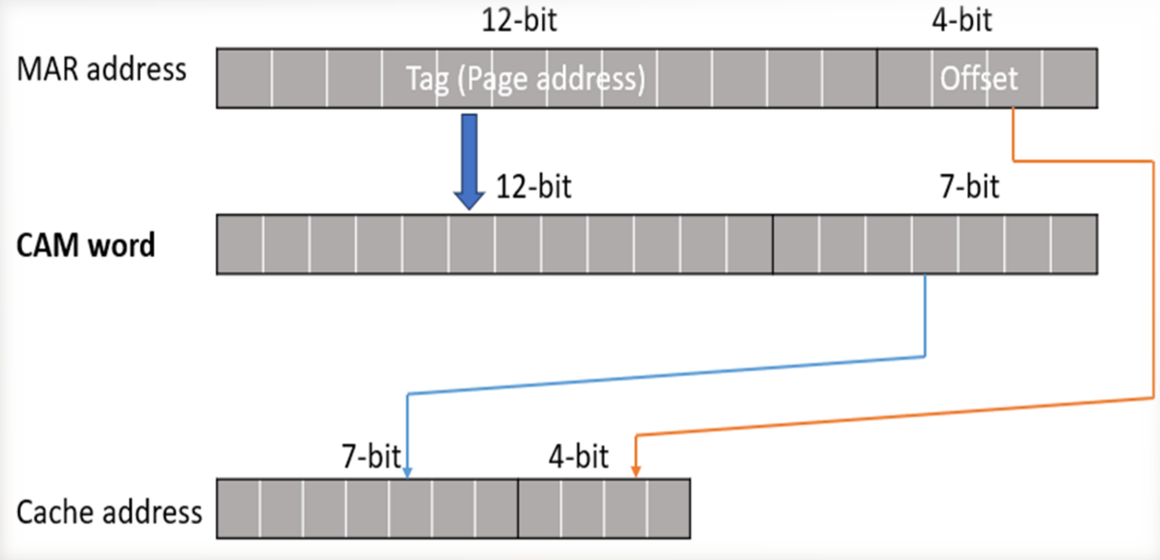
MAIN MEMORY

0	128	256	384	-	-	3969
1	129	257	385	-	-	-
2	130	258	386	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
127	255	383	411	-	-	4096
0	1	2	3 - - - -		- -	31

The main disadvantage is that multiple page data with the same tag number cannot be loaded in cache memory either of them only loaded.

Fully Associative Mapping

In associative mapping content of any page in main memory can be placed in any frame of cache memory. This is the main flexibility mapping over direct mapping.



- Every time the MAR 12-bit address will be taken and is searched in CAM (Content Addressable Memory) word space.
- In CAM word there are so many locations containing the page number and associated cache frame number.
- Whenever the data found, it is placed in the corresponding frame number associated by generating the cache address.

Disadvantage:

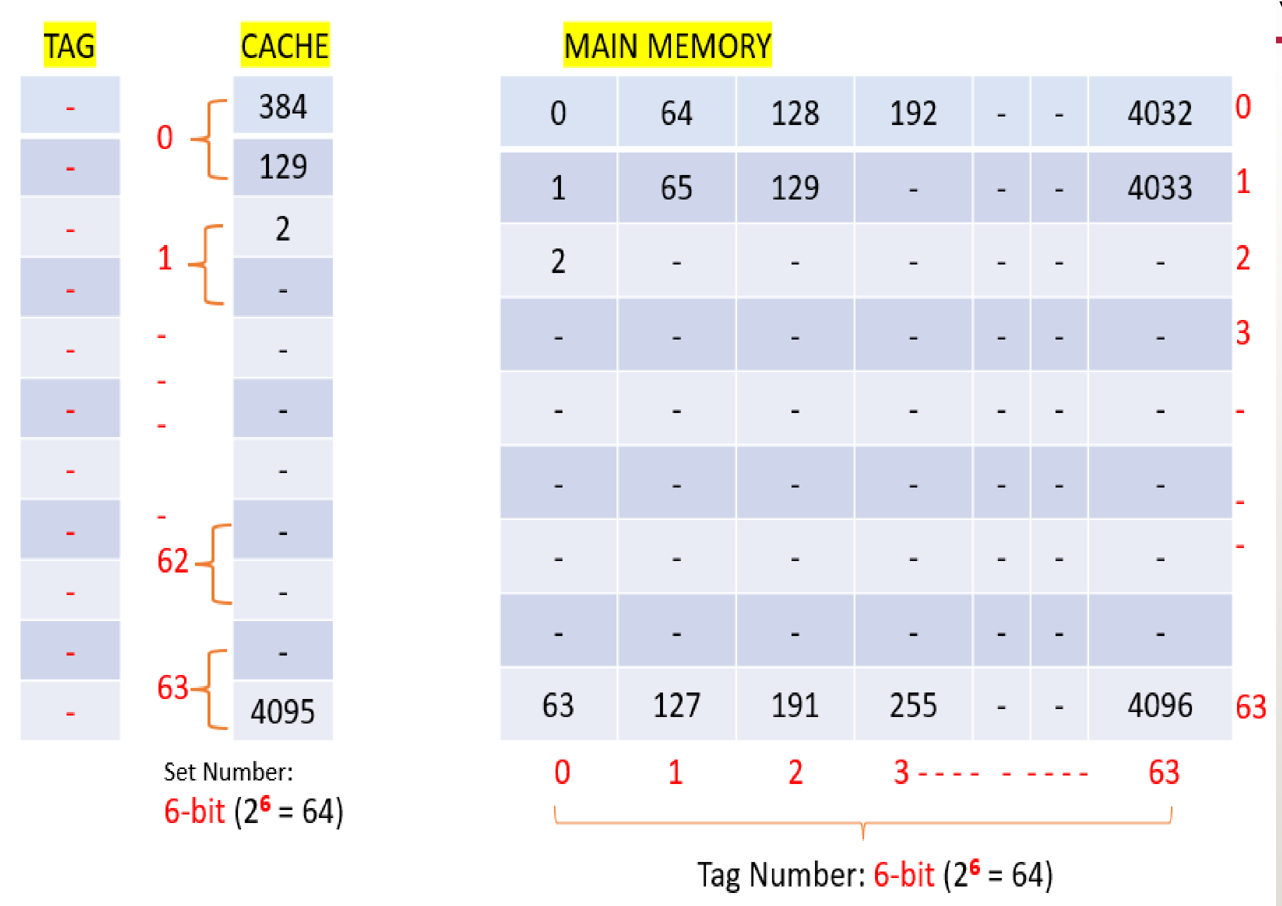
CAM word hardware is very costly, and each word can use 19-bit is not an affordable design mostly.

Advantages:

1. Main memory any page can be placed in Cache memory in any frame.
2. Due to usage of CAM word space, parallel searching will be done and very fastest data placing is possible.

K-Way Set (Block set) Associate mapping

If K = 2, means every two cache frames are forming a SET



TAG	Set	CACHE
-	0	384
-		129
-	1	2
-		-
-	-	-
-	-	-
-	-	-
-	-	-
-	62	-
-		-
-	63	-
-		4095

Set Number:
6-bit (2^6 = 64)

MAIN MEMORY

0	1	2	3 - - - -		- - - -	63	
0	64	128	192	-	-	4032	0
1	65	129	-	-	-	4033	1
2	-	-	-	-	-	-	2
-	-	-	-	-	-	-	3
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	
63	127	191	255	-	-	4096	63

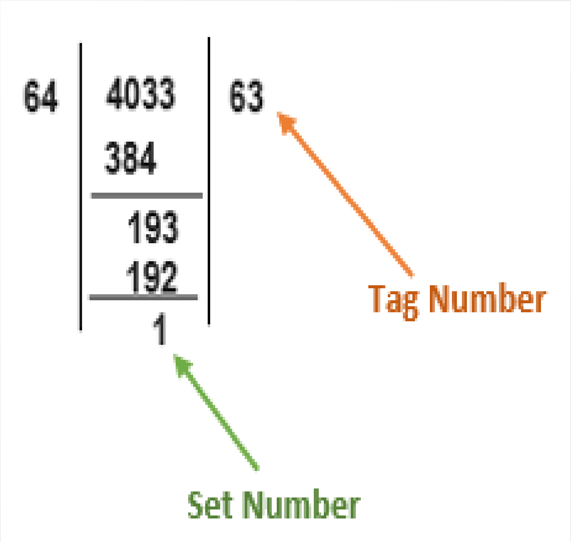
Tag Number: 6-bit (2^6 = 64)

For example, the page data '4033' we can identify the cache address where it can be placed in frames as follows.

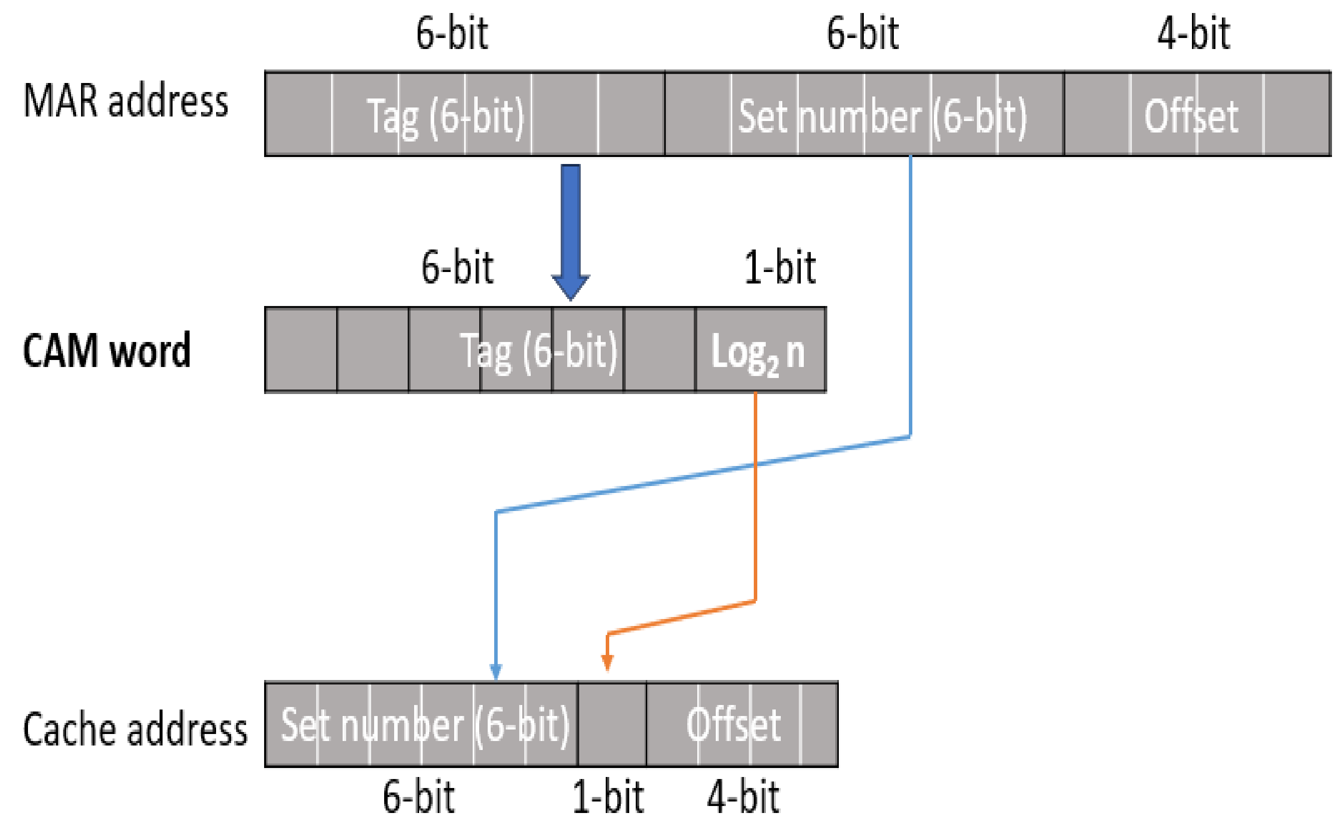
Let divide the 4033 by 64:

Here quotient 63 represents the Tag number and remainder represents the Set number.

Hence the data 4033 can be placed in either of the two frames in Set 1.



Cache Address Generation



- Here the CAM word uses 7-bit only compared to Associative mapping.
- Hence cost will greatly reduce and is the best mapping technique among all types.
- It is also called the Block set mapping.
- For different values of 'K' we can deduce the Cache address accordingly for the mapping of main memory data into the cache.

CACHE REPLACEMENT POLICY

- **First-in-first-out (FIFO) policy:** The earliest inserted item in the cache will be evicted when a new item needs to be inserted.
- **Last-in-first-out (LIFO) policy:** The last item inserted in the cache will be evicted first.
- **Least-recently used (LRU) policy:** The item which is least recently used will be evicted first. This is one of the most simple and common cache replacement policies.
- **Least-frequently-used (LFU) policy:** The cache algorithm maintains a counter on the number of times an item in the cache is accessed. It will evict the least frequently accessed item to add a new item.
- **Most-recently used (MRU) policy:** The item which is most recently used will be evicted first. This policy is useful, when the chance of repeating the same request soon is unlikely (like scrolling through a social media feed or flipping through a photo album).
- **Time-to-live (TTL) policy:** If an item remains in the cache beyond a given period without being accessed, the cache algorithm would discard it, to make room for a new item.
- **Random replacement (RR) policy:** The cache algorithm randomly selects an item to evict.

SUMMARY

A faster and smaller segment of memory whose access time is as close as registers are known as Cache memory. In a hierarchy of memory, cache memory has access time lesser than primary memory. Generally, cache memory is very small and hence is used as a buffer.

- Cache memory is faster than main memory.
- It consumes less access time as compared to main memory.
- It stores the program that can be executed within a short period of time.
- It stores data for temporary use.
- Cache memory has limited capacity.
- It is very expensive.

SELF-ASSESSMENT QUESTIONS

1. What is the high-speed memory between the main memory and the CPU called?

a) Register Memory
b) Cache Memory
c) Storage Memory
d) Virtual Memory

2. Whenever the data is found in the cache memory it is called as

A) HIT
b) MISS
c) FOUND
d) ERROR

3. LRU stands for

a) Low Rate Usage
b) Least Rate Usage
c) Least Recently Used
d) Low Required Usage

SELF-ASSESSMENT QUESTIONS

4. In _______________ mapping, the data can be mapped anywhere in the Cache Memory

a) Associative
b) Direct
c) Set Associative
d) Indirect

5. The transfer between CPU and Cache is _______________

a) Block transfer
b) Word transfer
c) Set transfer
d) Associative transfer

6. Which of the following is true about cache memory?

a) It is a small and fast memory
b) It is a type of volatile memory
c) It is primarily used for long-term storage
d) It has the largest storage capacity among all memory types

SELF-ASSESSMENT QUESTIONS

7. Which level of cache memory is closest to the CPU?

a) L1 cache
b) L2 cache
c) L3 cache
d) L4 cache

8. Which cache mapping technique allows a data item to be stored in any cache location?

a) Direct mapping
b) Set-associative mapping
c) Fully associative mapping
d) Segmented mapping

9.Which cache replacement algorithm aims to minimize the number of cache misses?

a) FIFO (First-In, First-Out)
b) LRU (Least Recently Used)
c) LFU (Least Frequently Used)
d) Random replacement

TERMINAL QUESTIONS

Short answer questions:

1. Formulate cache performance and its purpose.
2. Highlight various policies of cache data replacement.

Long answer questions:

1. Investigate the significance and impact of "Hit" and "Miss" events in cache memory, detailing how these occurrences influence system performance.
2. Designing a gaming console with a focus on **cache memory** efficiency, outline and exemplify **three mapping procedures**, each tailored to specific scenarios to optimize performance in gaming console architecture.
3. Interpret the **operation of cache memory** contribute to improving the performance of a processor, and what **advantages does it offer in terms of speed and efficiency?**

REFERENCES FOR FURTHER LEARNING OF THE SESSION

Reference Books:

1. Computer Organization by Carl Hamacher, Zvonko Vranesic and Saftwat Zaky.
2. Computer System Architecture by M. Morris Mano
3. Computer Organization and Architecture by William Stallings

Sites and Web links:

1. https://www.geeksforgeeks.org/cache-memory-in-computer-organization/
2. https://www.javatpoint.com/cache-memory

THANK YOU

(DEEMED TO BE UNIVERSITY)

Team – Digital Design & Computer Architecture