[Updated: 11/Sep/2020]

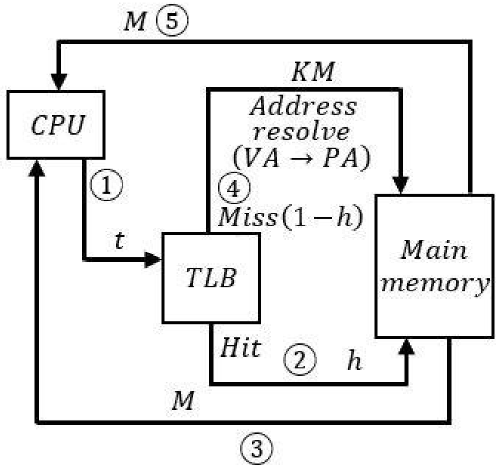
NOTE :

1) **We have considered average access time between the cache and the main memory, by default. However, one can consider the access time per instruction as well. This depends on the question and the assumptions made in the question.**
2) **Here, we have assumed no cache updation mechanism (i.e, a cache miss will lead to main memory access, no updation of cache is involved). In the GATE Exam, it is recommended that the students should read the question carefully, and note down the assumptions made in the question, and derive the formula accordingly.**
3) **We have also assumed that TLB hit implies no page fault.**
4) **Memorizing these formulas is highly discouraged.**

Calculate EMAT given the following scenarios:

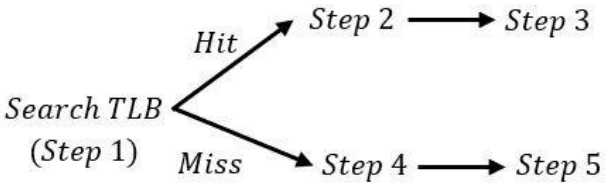
1.
K-level paging, tlb, main memory.
M = memory access time
t = tlb access time
h = tlb hit ratio, (1 - h) = tlb miss ratio
K = no. of page table levels

EMAT = h(t + M) + (1 - h) [KM + M + t]



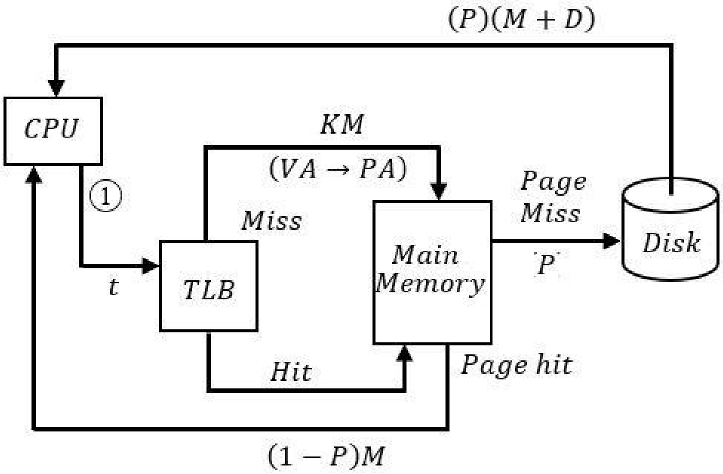
(1) Search TLB for address

(2) Address found in TLB (TLB hit)
(3) Fetch the page, memory access (M)
(4) TLB miss, resolve address (VA→PA), for K-level, KM accesses
(5) Step3: Fetch the page, Memory access (M)



2.

K-level paging, TLB, Main memory, page fault, Disk access given



P = Page fault rate, (1 - P) = Page hit rate
D = Disk access time

EMAT = h(t + M) + (1 - h) [t + KM + (1-P)(M) + P(M+D)]
Case(1) Given page is dirty with probability (d)
EMAT = h(t+M) + (1-h) [t + KM + (1-P)(M) + P[(1-d)(M+D) + d(M+2D)]]

3.

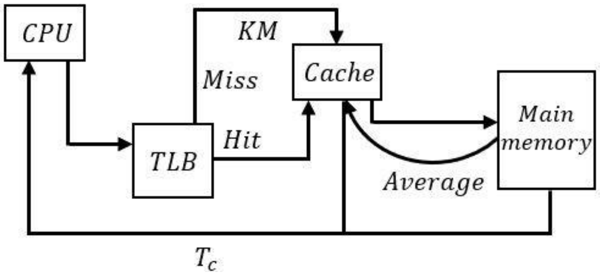
K-level paging, TLB, Main memory, Page fault, Page fault service time given (PFST)

EMAT = h(t+M) + (1-h)[t + KM + (1-P)(M) + P(PFST)]

4.

TLB, Cache, Main memory, K-level paging.

Assume page tables are not cached.



C = cache access time
a = cache hit ratio, (1-a) = cache miss ratio

EMAT = h[t + aC + (1-a)(M)] + (1-h)[t + KM + aC + (1-a)M]
Let T_c = aC + (1-a)M
EMAT = h(t + T_c) + (1-h)[t + KM + T_c]

5.
TLB, Cache, Main memory, K-level paging, Page fault, Disk access.
EMAT = h(t+T_c) + (1-h)[t + KM + (1-P)(T_c) + P(M+D)]

6.
TLB, Cache, Main memory, K-levels, Page fault, PFST.
EMAT = h(t+T_c) + (1-h)[t + KM + (1-P)(T_c) + P(PFST)]

7.
Case 5, with dirty (d)
EMAT = h(t+T_c) + (1-h)[t + KM + (1-P)(T_c) + P[(1-d)(M+D) + d(M+2D)]]

From now on assume page tables are also cached.8.
TLB, Cache, Main memory, K-level
EMAT = h[t+T_c] + (1-h)[t + KT_c + T_c]

9.
TLB, Cache, Main memory, K-level, Page fault, Disk access
EMAT = h[t+T_c] + (1-h)[t + KT_c + (1-P)(T_c) + P(M+D)]

10.
TLB, Cache, Main memory, K-levels, Page fault, PSFT
EMAT = h[t+T_c] + (1-h)[t + KT_c + (1-P)T_c + P(PFST)]

11.

Case 9, with Page dirty

EMAT = $h[t+T_c] + (1-h)[t + KT_c + (1-P)T_c + P[(1-d)(M+D) + d(M+2D)]]$

Note: $T_c = aC + (1-a)M$ = average access time between Memory & Cache. We assume that on cache miss, the page is fetched from main memory, and the cache update time is negligible. Similarly, in the case of page fault and TLB miss.

****