

## MODUL PRAKTIKUM 07

### TOPIK: SIMULASI TRANSLASI PAGING MEMORY & KALKULASI EAT TLB

Mata Kuliah: Sistem Operasi (INF-201) – Alokasi Waktu: 170 Menit

Program Studi Informatika – Fakultas Teknik – Universitas Bengkulu

#### 1. CAPAIAN PEMBELAJARAN MODUL (SUB-CPMK 7)

Setelah menyelesaikan Modul Praktikum 07 ini, mahasiswa diharapkan mampu:

1. Memisahkan alamat logis 32-bit menjadi Page Number ( $p$ ) dan Page Offset ( $d$ ).
2. Memprogram alur translasi Paging memori ke alamat fisik RAM  $(p, d) \rightarrow (f, d)$ .
3. Mensimulasikan perhitungan kuantitatif *Effective Access Time* (EAT) pada cache TLB.

#### 2. PRASYARAT TUGAS MEMBACA

Sebelum memulai percobaan laboratorium ini, mahasiswa **wajib** telah membaca bab referensi berikut:

- [OSTEP] Chapter 18: *Paging: Introduction* (hal. 189–200).
- [OSTEP] Chapter 19: *Paging: Faster Translations (TLBs)* (hal. 201–214).
- [xv6 RISC-V] Chapter 3: *Page Tables* (hal. 29–40).

#### 3. LANDASAN TEORI RINGKAS

##### 3.1 Translasi Alamat Paging

Ukuran Page 4 KB =  $2^{12}$  bytes  $\implies$  Offset  $d$  berukuran 12 bits (bit 0...11). Bit posisi paling atas (bit 12...31) bertindak sebagai Page Number  $p$ .

$$\text{Alamat Fisik} = (\text{Frame Number } f \ll 12) \text{ --- Offset } d$$

##### 3.2 Kalkulasi Effective Access Time (EAT)

$$\text{EAT} = \alpha(\epsilon + t_m) + (1 - \alpha)(\epsilon + 2t_m)$$

di mana  $\alpha$  = TLB Hit Ratio,  $\epsilon$  = TLB Search Time, dan  $t_m$  = Memory Access Time.

#### 4. KODE ACUAN UTUH (EXECUTABLE C LISTING: paging\_sim.c)

```
1 #include <stdio.h>
2 #include <stdint.h>
3
4 #define PAGE_SIZE 4096 // 4 KB
5 #define OFFSET_BITS 12
6 #define OFFSET_MASK 0xFFF // 12 bit terkecil (1)
7
8 int main() {
9     // Page Table Sederhana: Index = Page Number, Value = Frame Number
10     int page_table[] = {5, 8, 2, 12, 1};
11 }
```

```

12     uint32_t logical_addr = 0x00002018; // Alamat logis contoh
13
14     uint32_t page_num = logical_addr >> OFFSET_BITS;
15     uint32_t offset    = logical_addr & OFFSET_MASK;
16
17     int frame_num = page_table[page_num];
18     uint32_t physical_addr = (frame_num << OFFSET_BITS) | offset;
19
20     printf("Alamat Logis Hex: 0x%08X\n", logical_addr);
21     printf("Page Number (p): %u (Hex: 0x%X)\n", page_num, page_num);
22     printf("Offset (d): %u (Hex: 0x%X)\n", offset, offset);
23     printf("Frame Number (f): %d\n", frame_num);
24     printf("Alamat Fisik RAM: 0x%08X\n", physical_addr);
25
26     // Hitung EAT
27     float alpha = 0.80; // 80% Hit Ratio
28     float search = 20;  // 20 ns
29     float ram = 100;    // 100 ns
30     float eat = alpha * (search + ram) + (1 - alpha) * (search + 2 * ram);
31     printf("\nEffective Access Time (EAT) dengan TLB Hit 80%: %.2f ns\n",
32           eat);
33
34     return 0;
35 }

```

Listing 1: Program Simulasi Paging &amp; TLB EAT C (paging\_sim.c)

## 5. PROSEDUR LANGKAH KERJA PERCOBAAN

1. Kompilasi dan jalankan paging\_sim.c: `gcc paging_sim.c -o paging_sim && ./paging_sim.`
2. Ubah alamat logis menjadi 0x00001004 dan tentukan hasil Alamat Fisik RAM.

## 6. TUGAS PRAKTIKUM & PERTANYAAN ANALISIS LAPORAN

1. **Analisis EAT:** Berapa hit ratio  $\alpha$  minimum yang dibutuhkan agar EAT tidak melebihi 130 ns?