**Chapter 2 Exercise**

1. **What are program-visible registers?**  
   **Answer:** Program-visible registers are the registers that are directly used/accessible by instructions (i.e., registers visible to the programmer and referenced in instruction operands).
2. **The 80286 addresses registers that are 8 and \_\_\_\_\_\_\_\_\_ bits wide.**  
   **Answer:** **16** bits.
3. **The extended registers are addressable by which microprocessors?**  
   **Answer:** The extended (32-bit) registers are addressable by the **80386 and later** (i.e., 80386 → Pentium 4, etc.).
4. **The extended BX register is addressed as \_\_\_\_\_\_\_\_\_.**  
   **Answer:** **EBX**.
5. **Which register holds a count for some instructions?**  
   **Answer:** **CL / CX / ECX / RCX** (CX is the 16-bit count; ECX is the 32-bit count; RCX the 64-bit in x86-64).
6. **What is the purpose of the IP/EIP register?**  
   **Answer:** IP (EIP/RIP) is the **instruction pointer** — it holds the offset (address) of the next instruction to be fetched/executed.
7. **The carry flag bit is not modified by which arithmetic operations?**  
   **Answer:** The **INC** and **DEC** instructions do **not** modify the Carry Flag (CF).
8. **Will an overflow occur if a signed FFH is added to a signed 01H?**  
   **Answer:** **No.** Signed 0xFF = −1 (8-bit), plus +1 → 0x00. No overflow (result = 00H).

### Step 3: Overflow (OF) condition

Signed overflow হয় যদি:

1. দুটো **positive** সংখ্যা যোগ করলে result negative হয়
2. দুটো **negative** সংখ্যা যোগ করলে result positive হয়
3. **A number that contains 3 one bits is said to have \_\_\_\_\_\_\_\_\_ parity.**  
   **Answer:** **Odd parity.**
4. **Which flag bit controls the INTR pin on the microprocessor?**  
   **Answer:** The **Interrupt Enable Flag (IF)** controls whether the INTR pin is recognized.

A screenshot of a computer program

AI-generated content may be incorrect.

1. **Which microprocessors contain an FS segment register?**  
   **Answer:** The **80386 and later** processors (80386 introduced FS and GS).

A screenshot of a black screen

AI-generated content may be incorrect.

1. **What is the purpose of a segment register in the real mode operation of the microprocessor?**  
   **Answer:** In real mode a segment register provides a **segment base**. In real mode, a segment register holds the base address of a memory segment, which is used to calculate the physical memory address.

The physical address = **(segment × 10h) + offset**. Each segment register selects a 64-KB window (base .. base+0FFFFH).

1. **In the real mode, show the starting and ending addresses of each segment located by the following segment register values:**

• (a) 1000H: Starting address = 10000H, Ending address = 1FFFFH

• (b) 1234H: Starting address = 12340H, Ending address = 2233FH

• (c) 2300H: Starting address = 23000H, Ending address = 32FFFH

• (d) E000H: Starting address = E0000H, Ending address = EFFFH

• (e) AB00H: Starting address = AB000H, Ending address = BAFFFH

* + **Starting address** = Segment × 10H + 0000H

**Ending address** = Segment × 10H + FFFFH

1. **Find the memory address of the next instruction executed by the microprocessor, when operated in the real mode, for the following CS:IP combinations:**  
   (1) CS = 1000H and IP = 2000H → address = 1000H×10h+ 2000H = **12000H**.  
   (2) CS = 2000H and IP = 1000H → address = 2000H× 10h + 1000H = **21000H**.  
   (3) CS = 2300H and IP = 1A00H → address = 2300H× 10h + 1A00H = **24A00H**.  
   (4) CS = 1A00H and IP = B000H → address = 1A00H×10h+ B000H = **25000H**.  
   (5) CS = 3456H and IP = ABCDH → address = 3456H×10h+ ABCDH = **3F12DH**.
2. **Real mode memory addresses allow access to memory below which memory address?**  
   **Answer:** Real mode (20-bit) can address up to **FFFFFH**, i.e. below **1 MB (0x100000)**.
3. **Which register or registers are used as an offset address for the string instruction destination in the microprocessor?**  
   **Answer:** **DI** (or **EDI/RDI** for 32/64-bit modes) is used as the destination offset for string instructions (SI/ESI/RSI is source).
4. **Which 32-bit register or registers are used to hold an offset address for data segment data in the Pentium 4 microprocessor?**  
   **Answer:** The 32-bit base/index registers such as **ESI, EDI,**  (and in general EAX/ECX/EDX can be used) can hold offsets for data; commonly **EBX/ESI/EDI/EBP** are used as base/index registers for data addressing.
5. **The stack memory is addressed by a combination of the \_\_\_\_\_\_\_\_\_ segment plus \_\_\_\_\_\_\_\_\_ offset.**  
   **Answer:** **SS** segment plus **SP** (stack pointer) offset
6. **If the base pointer (BP) addresses memory, the \_\_SS,SP,BP\_\_\_\_\_\_\_ segment contains the data.**  
   **Answer:** **SS** (stack segment is the default when BP is used).
7. **Determine the memory location addressed by the following real mode 80286 register combinations:**  
   (1) DS = 1000H and DI = 2000H → address = 1000H×10h + 2000H = **12000H**.  
   (2) DS = 2000H and SI = 1002H → address = 2000H× 10h + 1002H = **21002H**.  
   (3) SS = 2300H and BP = 3200H → address = 2300H×10h+ 3200H = **26200H**.  
   (4) DS = A000H and BX = 1000H → address = A000H× 10h + 1000H = **A1000H**.  
   (5) SS = 2900H and SP = 3A00H → address = 2900H×10h+ 3A00H = **2CA00H**.

**21. Determine the memory location addressed by the following real mode Core2 register combinations:**  
(a) **DS = 2000H and EAX = 00003000H**  
**Answer:** Segment base = 2000H × 10h = **20000H**. Address = 20000H + 3000H = **23000H**.

(b) **DS = 1A00H and ECX = 00002000H**  
**Answer:** Segment base = 1A00H × 10h= **1A000H**. Address = 1A000H + 2000H = **1C000H**.

(c) **DS = C000H and ESI = 0000A000H**  
**Answer:** Segment base = C000H × 10h = **C0000H**. Address = C0000H + A000H = **CA000H**.

(d) **SS = 8000H and ESP = 00009000H**  
**Answer:** Segment base = 8000H × 10h = **80000H**. Address = 80000H + 9000H = **89000H**.

(e) **DS = 1239H and EDX = 0000A900H**  
**Answer:** Segment base = 1239H × 10h= **12390H**. Address = 12390H + A900H = **1CC90H**.

**22. Protected mode memory addressing allows access to which area of the memory in the 80286 microprocessor?**  
**Answer:** Protected mode allows access to memory above the 1MB limit.

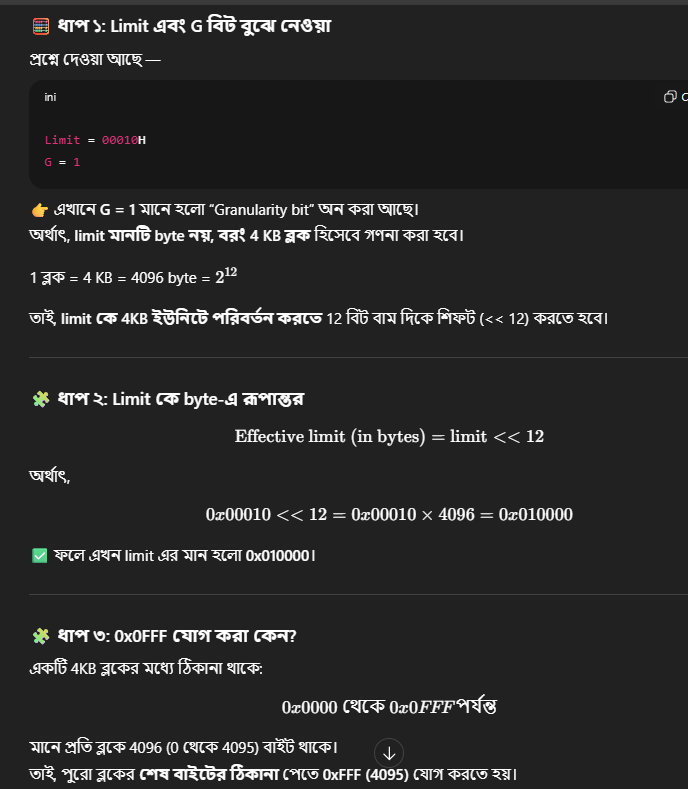
**23. Protected mode memory addressing allows access to which area of the memory in the Pentium 4 microprocessor?**  
**Answer:** Protected mode allows access to the entire 4GB addressable memory.

**24. What is the purpose of the segment register in protected mode memory addressing?**  
**Answer:** In protected mode a segment register holds a **segment selector** (not a plain base). The selector indexes into the **GDT/LDT** to fetch a **descriptor** (which contains base address, limit, and access/control information). Thus segment registers provide **base/limit translation and protection (access rights)** via descriptors.

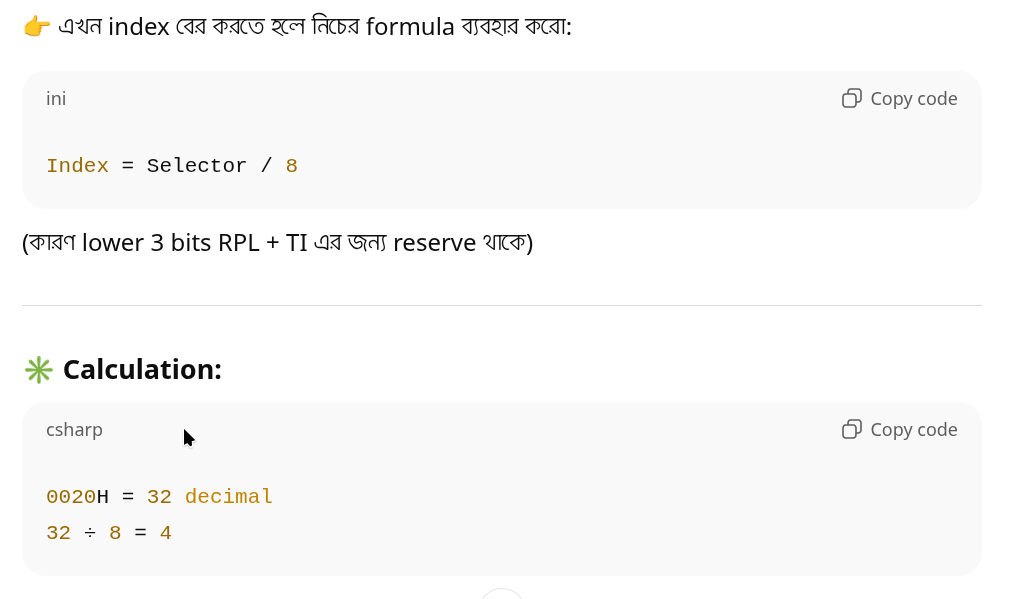
**25. How many descriptors are accessible in the global descriptor table in the protected mode?**  
**Answer:** The GDT limit is a 16-bit field (max 0xFFFF = 65535 bytes). Each descriptor is 8 bytes, so maximum descriptors = 65536 / 8 = **8192 descriptors** (indexes 0 through 8191).

**26. For an 80286 descriptor that contains a base address of A00000H and a limit of 1000H, what starting and ending locations are addressed by this descriptor?**  
**Answer:** Starting = **A00000H**. Ending = base + limit = A00000H + 1000H = **A01000H**.  
(অর্থাৎ শ্রেণীটি A00000H ... A01000H পর্যন্ত।)

**27. For a Core2 descriptor that contains a base address of 01000000H, a limit of 0FFFFH, and G = 0, what starting and ending locations are addressed by this descriptor?**  
**Answer:** G = 0 → limit is in **bytes**.  
Starting = **01000000H**. Ending = 01000000H + 0FFFFH = **0100FFFFH**.

**28. For a Core2 descriptor that contains a base address of 00280000H, a limit of 00010H, and G = 1, what starting and ending locations are addressed by this descriptor?**  
**Answer:** G = 1 → limit is in **4-KB units**; effective byte-limit = (limit << 12) + 0xFFF.  
Here limit = 00010H → (00010H << 12) = 0x010000, + 0x0FFF = **0x010FFF**.  
Starting = **00280000H**. Ending = 00280000H + 010FFFH = **00290FFFH**. 

**29. If the DS register contains 0020H in a protected mode system, which global descriptor table entry is accessed?**  
**Answer:** Selector 0020H → index = selector >> 3 = 0020H >> 3 = **0004H**. TI bit = 0 → **GDT**.  
সুতরাং এটি **GDT entry (descriptor) number 4** (index 4) অ্যাক্সেস করে।



1. **If DS = 0103H in a protected mode system, the requested privilege level is \_\_\_\_\_\_\_\_\_.**

Protected mode–এ segment selector (যেমন DS register) এর গঠন হলো:

15 .......... 3 2 1 0

[ Index ][TI][ RPL ]

* **Index** = bits 15–3
* **TI (Table Indicator)** = bit 2 → 0 = GDT, 1 = LDT
* **RPL (Requested Privilege Level)** = bits 1–0

**এখন DS = 0103H**

Hex → Binary:

0103H = 0000 0001 0000 0011b

* RPL = bits 1–0 = **11b = 3**
* TI = bit 2 = 0 → GDT ব্যবহার হচ্ছে
* Index = bits 15–3 = 0000 0001 0000b = 0x20 = decimal 32

✅ **Answer:** The **Requested Privilege Level (RPL) = 3**

1. **If DS = 0105H in a protected mode system, which entry, table, and requested privilege level are selected?**

**Like 30**

**✅ চূড়ান্ত উত্তর**

* **Entry selected** = 32nd descriptor
* **Table** = LDT
* **Requested Privilege Level (RPL)** = 1
* The system selects a specific descriptor from the global descriptor table based on the requested privilege level.

1. **What is the maximum length of the global descriptor table in the Pentium 4 microprocessor?**

* The maximum length is 8GB (8192 entries, each 8 bytes).

1. **Code a descriptor that describes a memory segment that begins at location 210000H and ends at location 21001FH. This memory segment is a code segment that can be read. The descriptor is for an 80286 microprocessor.**

* Base address = 210000H, Limit = 1FH, Type = Code Segment, Access Rights = Read.

1. **Code a descriptor that describes a memory segment that begins at location 03000000H and ends at location 05FFFFFFH. This memory segment is a data segment that grows upward in the memory system and can be written. The descriptor is for a Pentium 4 microprocessor.**

* Base address = 03000000H, Limit = 02FFFFFFH, Type = Data Segment, Access Rights = Read/Write.

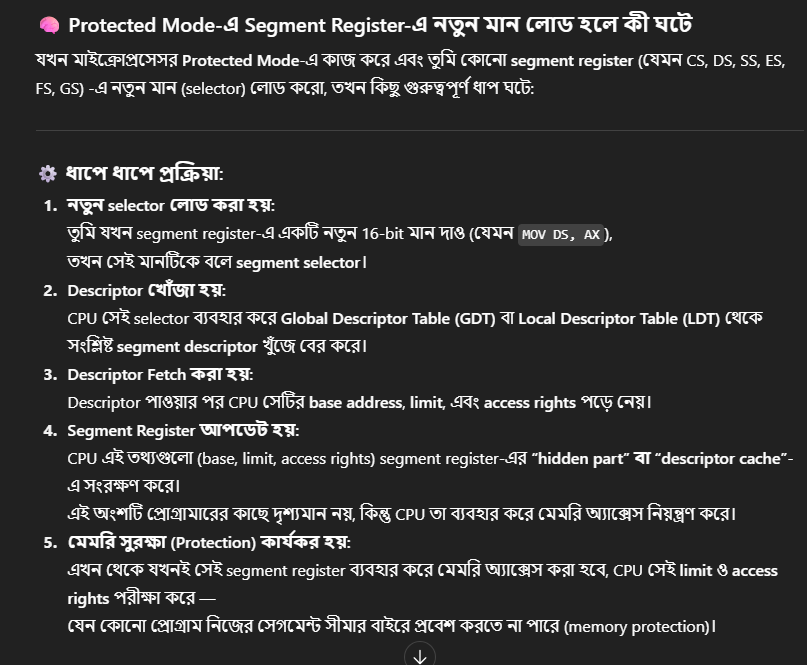
1. **Which register locates the global descriptor table?**

* The GDTR (Global Descriptor Table Register).

1. **How is the local descriptor table addressed in the memory system?**

* The LDTR (Local Descriptor Table Register) addresses the local descriptor table.

1. **Describe what happens when a new number is loaded into a segment register when the microprocessor is operated in the protected mode.**

* A new number is loaded into the segment register, which causes the descriptor for the segment to be fetched, and the microprocessor updates the segment's base, limit, and access rights from the descriptor.
* 

1. **What are the program-invisible registers?**

* Program-invisible registers are internal registers used by the microprocessor for various purposes but are not directly accessible by programs, such as the CR0(CR0 হলো একটি **“program-invisible register”**, কারণ প্রোগ্রামার এটি সরাসরি ব্যবহার করতে পারে না — শুধুমাত্র অপারেটিং সিস্টেম বা বিশেষ অনুমতি-প্রাপ্ত কোড এটি পরিবর্তন করতে পারে।) register for control.
* Program-Invisible Registers হলো segment descriptor cache (base, limit, access rights), যা CPU descriptor table থেকে লোড করে, কিন্তু প্রোগ্রামার সরাসরি দেখতে বা পরিবর্তন করতে পারে না।

1. **What is the purpose of the GDTR?**

* The GDTR (Global Descriptor Table Register) holds the base address and limit of the global descriptor table.

1. **How many bytes are found in a memory page?**

* A memory page is typically 4KB.

1. **What register is used to enable the paging mechanism in the 80386, 80486, Pentium, Pentium Pro, Pentium 4, and Core2 microprocessors?**

The CR0 register is used to enable the paging mechanism.

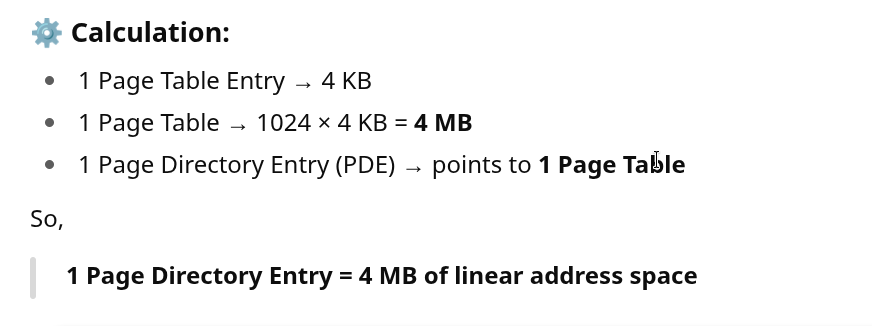
1. **How many 32-bit addresses are stored in the page directory?**

* 1024 addresses.

1. **Each entry in the page directory translates how much linear memory into physical memory?**

* Each entry in the page directory translates 4MB of linear memory to physical memory.

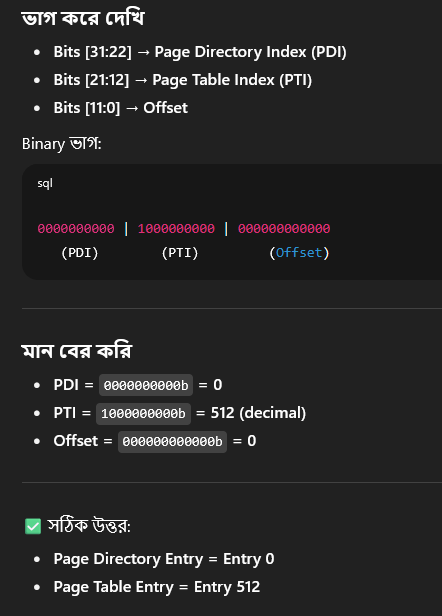
👉 **Linear Address** হলো সেই 32-bit (বা 64-bit) ঠিকানা যেটা **Segmentation এর পরে পাওয়া যায়, কিন্তু Paging-এর আগে থাকে।**



1. **If the microprocessor sends linear address 00200000H to the paging mechanism, which paging directory entry is accessed, and which page table entry is accessed?**

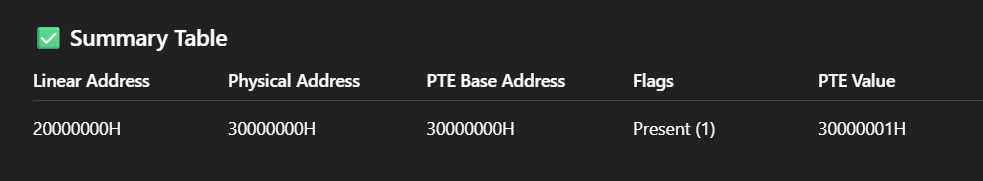
A screenshot of a computer

AI-generated content may be incorrect.

*  A black background with white text

  AI-generated content may be incorrect.

1. **What value is placed in the page table to redirect linear address 20000000H to physical address 30000000H?**

* The page table entry would contain the base address of 30000000H.­
* 

1. **What is the purpose of the TLB located within the Pentium class microprocessor?**

* The TLB (Translation Lookaside Buffer) caches recent page table entries to speed up address translation.

1. **Using the Internet, write a short report that details the TLB. Hint: You might want to go to the Intel Website and search for information.**

* The TLB speeds up memory access by storing recent translations of virtual addresses to physical addresses. It reduces the need to access the page table, improving system performance.

1. **Locate articles about paging on the Internet and write a report detailing how paging is used in a variety of systems.**

* Paging is a memory management scheme that eliminates the need for contiguous allocation of physical memory, making it easier to manage memory efficiently.

1. **What is the flat mode memory system?**

* The flat mode memory system provides a linear address space that allows programs to access memory without segmentation, typically used in modern 64-bit systems.

1. **A flat mode memory system in the current version of the 64-bit Pentium 4 and Core2 allow these microprocessors to access \_\_\_\_\_\_\_\_\_ bytes of memory.**

* 1TB (Terabyte) of memory.