Unit 7:
Advanced Microprocessors (9 Hrs.)

- 80286:
  - ✓ Salient Features of 80286,
  - ✓ Architecture (Block Diagram) ,
  - ✓ Registers,
  - ✓ (Real/Protected mode),
  - ✓ Privilege Levels,
  - ✓ Descriptor Cache,
  - ✓ Memory Access in GDT and LDT,
  - ✓ Multitasking,
  - ✓ Addressing Modes,
  - ✓ Flag Register
- 80386:
  - ✓ Architecture (Block Diagram),
  - ✓ Register organization,
  - ✓ Memory Access in Protected Mode,
  - ✓ Paging (Up to LA to PA)

Salient features of 80286

- It is **x86 family** and is of **16-bit microprocessor**, introduced in **1982**.
- It is housed in **68-pin package.**
- Available in **8 MHz, 10 MHz & 12.5 MHz** *clock frequencies*
- It has **non-multiplexed address and data bus** that *reduces operational speed.*
- It offers an **additional adder** *for address calculation*.
- The 80286 CPU, with its **24-bit address bus** is able to address **16MB** of *physical memory*.
- **286** includes *special instructions to support operating system*.
- **1GB** of **virtual memory** for each task.

Salient features of 80286 cont…

❖High performance microprocessor with **memory management and protection :** 80286 is the first member of the family of advanced microprocessors with **built-in/on-chip memory management** and **protection abilities** primarily designed for **multi-user/multitasking systems**.

❖ Intel 80286 has **2 operating modes:**

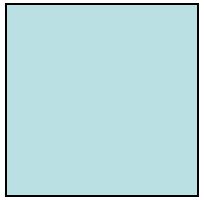
✓ **Real Address Mode :**
- ❑ 80286 is just a fast 8086 --- up **to 6 times faster**
- ❑ All memory management and protection mechanisms are disabled
- ❑ 286 is object code compatible with 8086

✓ **Protected Virtual Address Mode**
- ❑ 80286 works with all of its memory management and protection capabilities with the advanced instruction set.
- ❑ It is source code compatible with 8086

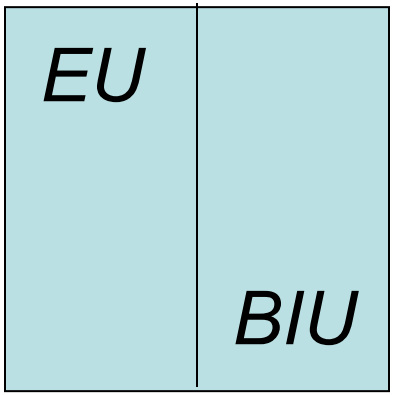
Salient features of 80286 cont…

- Unlike 8086, it has **4 separate processing units**: BIU(Bus Interface Unit),Instruction Unit(IU), Address Unit(AU) and EU(Execution Unit).
- It has **8 general purpose registers** and **4 segment registers**.
- **80286** have two operating modes namely real address mode and virtual address mode.
  - ✓In real address mode, it can address up to 1MB of physical memory address like 8086.
  - ✓In Protected virtual address mode(PVAM), it can address up to 16 MB of physical memory address space and 1 GB of virtual memory address space.
- The performance of 80286 is six times faster than 8086.

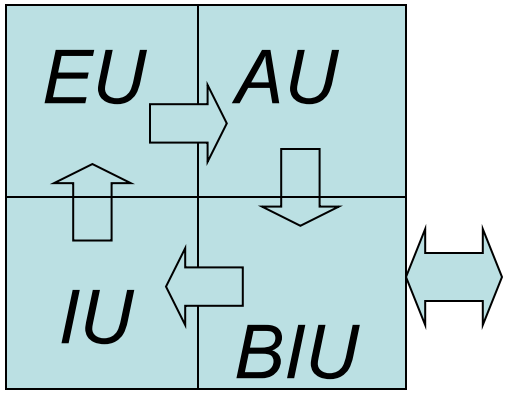
80286
INTERNAL ARCHITECTURE



8085



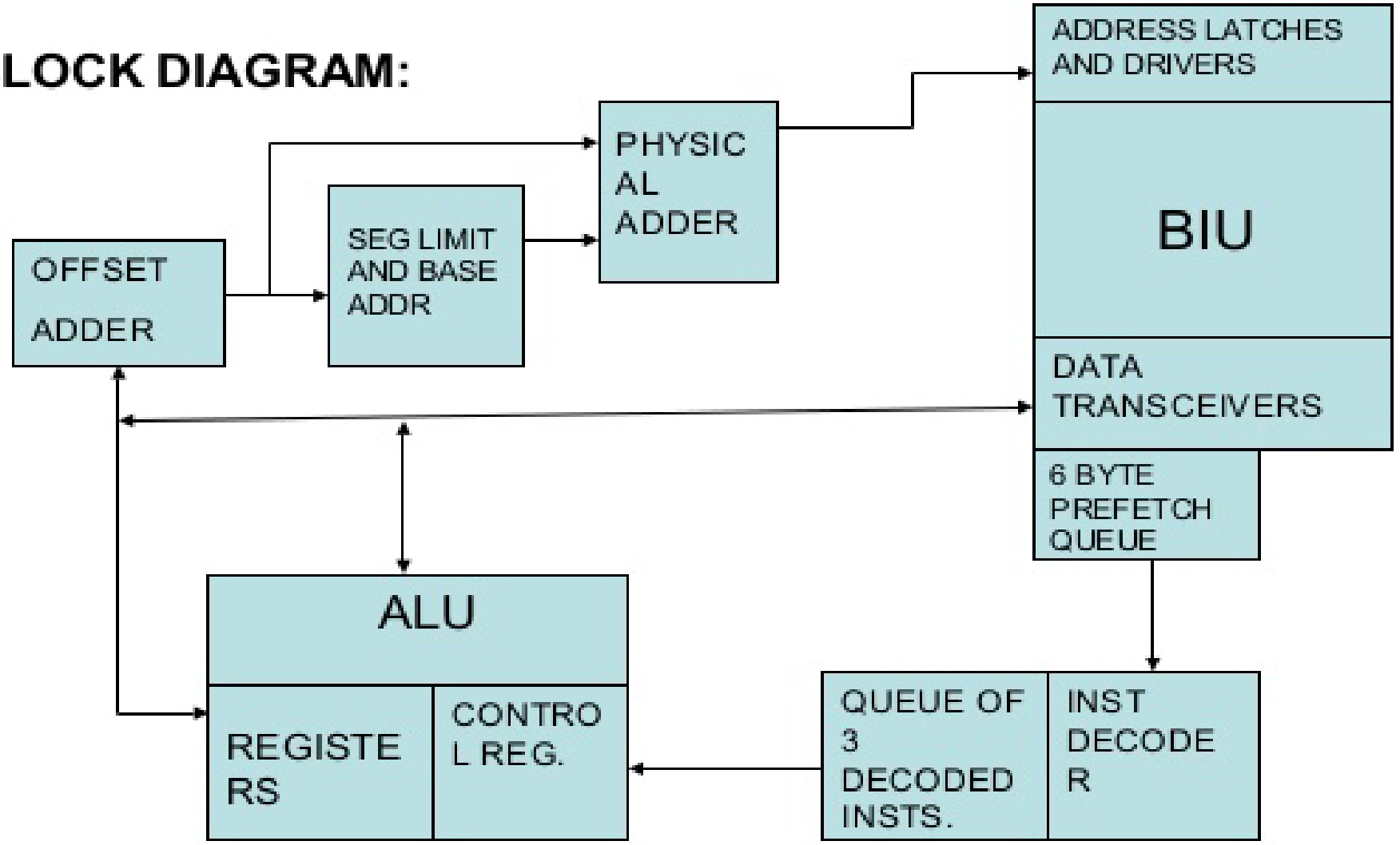
8086

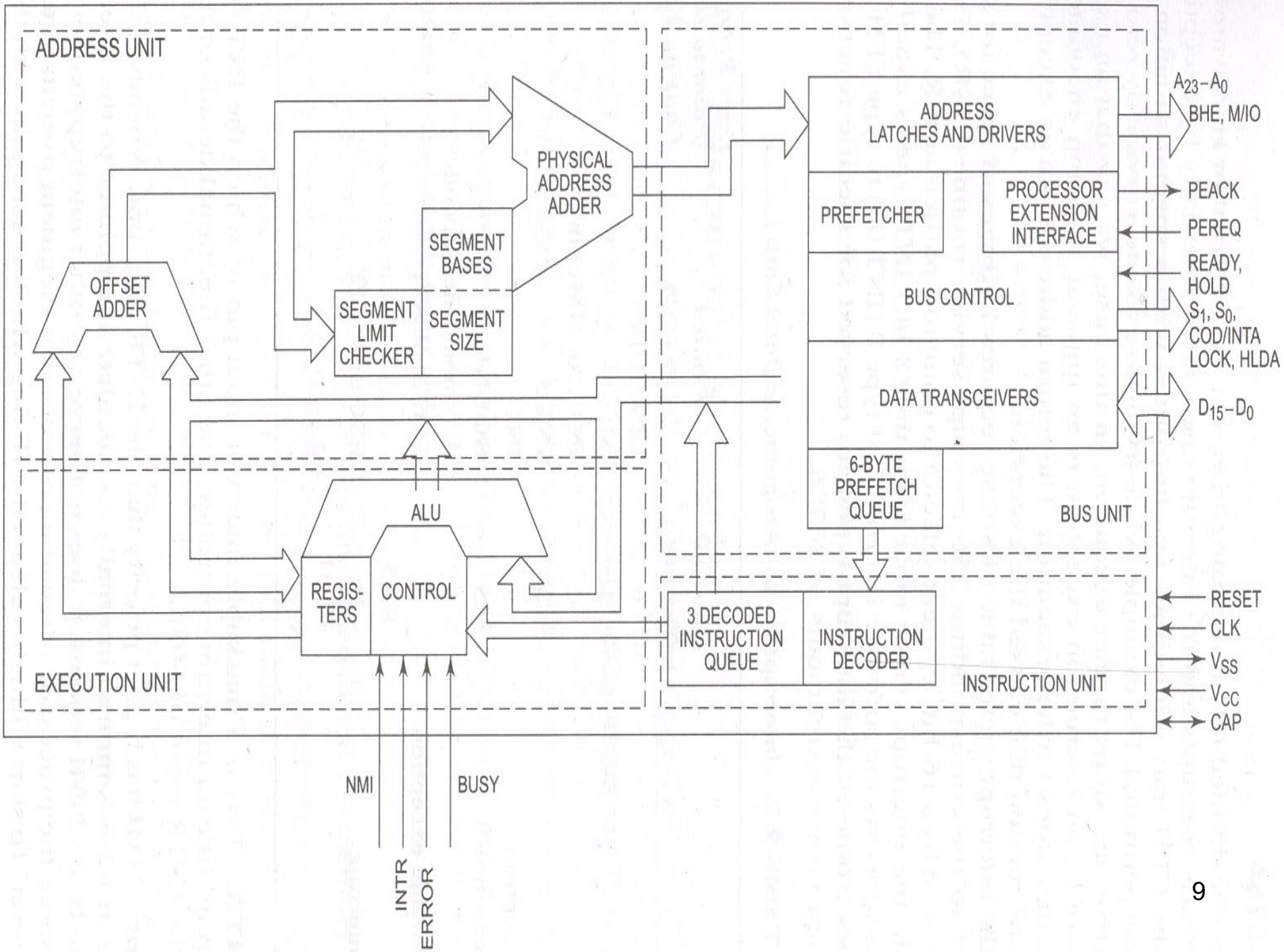


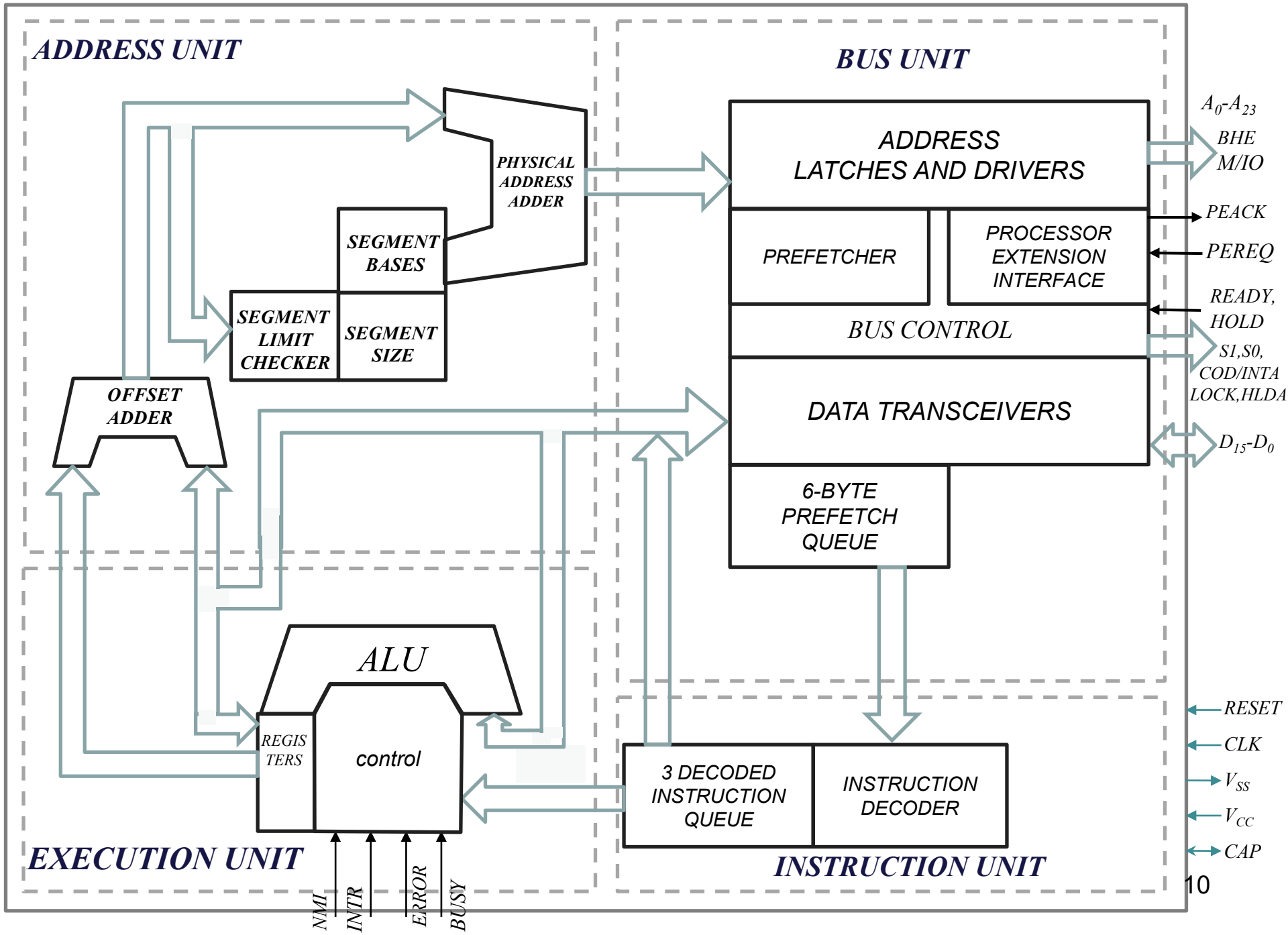
80286

80286

BLOCK DIAGRAM:




ADDRESS UNIT
PHYSICAL ADDRESS ADDER
SEGMENT BASES
SEGMENT LIMIT CHECKER
SEGMENT SIZE
OFFSET ADDER
ADDRESS LATCHES AND DRIVERS
PREFETCHER
PROCESSOR EXTENSION INTERFACE
BUS CONTROL
DATA TRANSCEIVERS
6-BYTE PREFETCH QUEUE
BUS UNIT
A23–A0
BHE, M/IO
PEACK
PEREQ
READY, HOLD
S1, S0, COD/INTA LOCK, HLDA
D15–D0
ALU
REGIS-TERS
CONTROL
3 DECODED INSTRUCTION QUEUE
INSTRUCTION DECODER
INSTRUCTION UNIT
EXECUTION UNIT
RESET
CLK
VSS
VCC
CAP
NMI
BUSY
INTR
ERROR


ADDRESS UNIT
BUS UNIT
PHYSICAL ADDRESS ADDER
SEGMENT BASES
SEGMENT LIMIT CHECKER
SEGMENT SIZE
OFFSET ADDER
ADDRESS LATCHES AND DRIVERS
PREFETCHER
PROCESSOR EXTENSION INTERFACE
BUS CONTROL
DATA TRANSCEIVERS
6-BYTE PREFETCH QUEUE
A0-A23
BHE
M/IO
PEACK
PEREQ
READY,
HOLD
S1,S0,
COD/INTA
LOCK,HLDA
D15-D0
ALU
REGIS TERS
control
3 DECODED INSTRUCTION QUEUE
INSTRUCTION DECODER
RESET
CLK
VSS
VCC
CAP
NMI
INTR
ERROR
BUSY
EXECUTION UNIT
INSTRUCTION UNIT

Functional Parts of 80286 are:

1. Bus Interface unit(BIU),
2. Instruction unit(IU),
3. Execution unit(EU), and
4. Address unit(AU)

1. Bus Interface Unit

- Performs all memory and I/O read and write operations.
- Take care of communication between CPU and a coprocessor.
- Transmit the physical address over address bus $A_0 - A_{23}$.
- **Prefetcher** module in the bus unit performs this task of prefetching.
- Bus controller controls the prefetcher module.
- Fetched instructions are arranged in a 6 – byte pre-fetch queue.

2. Instruction Unit

- Receive arranged instructions from **6 byte pre-fetch queue.**
- Instruction decoder decodes up to 3 pre-fetched instruction and are latched them onto a decoded instruction queue.
- Output of the decoding circuit drives a control circuit in the Execution unit.

3. Execution unit

- EU executes the instructions received from the decoded instruction queue sequentially.
- Contains Register Bank.
- contains one additional special register called **Machine status word (MSW)** register --- lower 4 bits are only used.
- ALU is the heart of execution unit.
- After execution ALU sends the result either over data bus or back to the register bank.

4. Address Unit

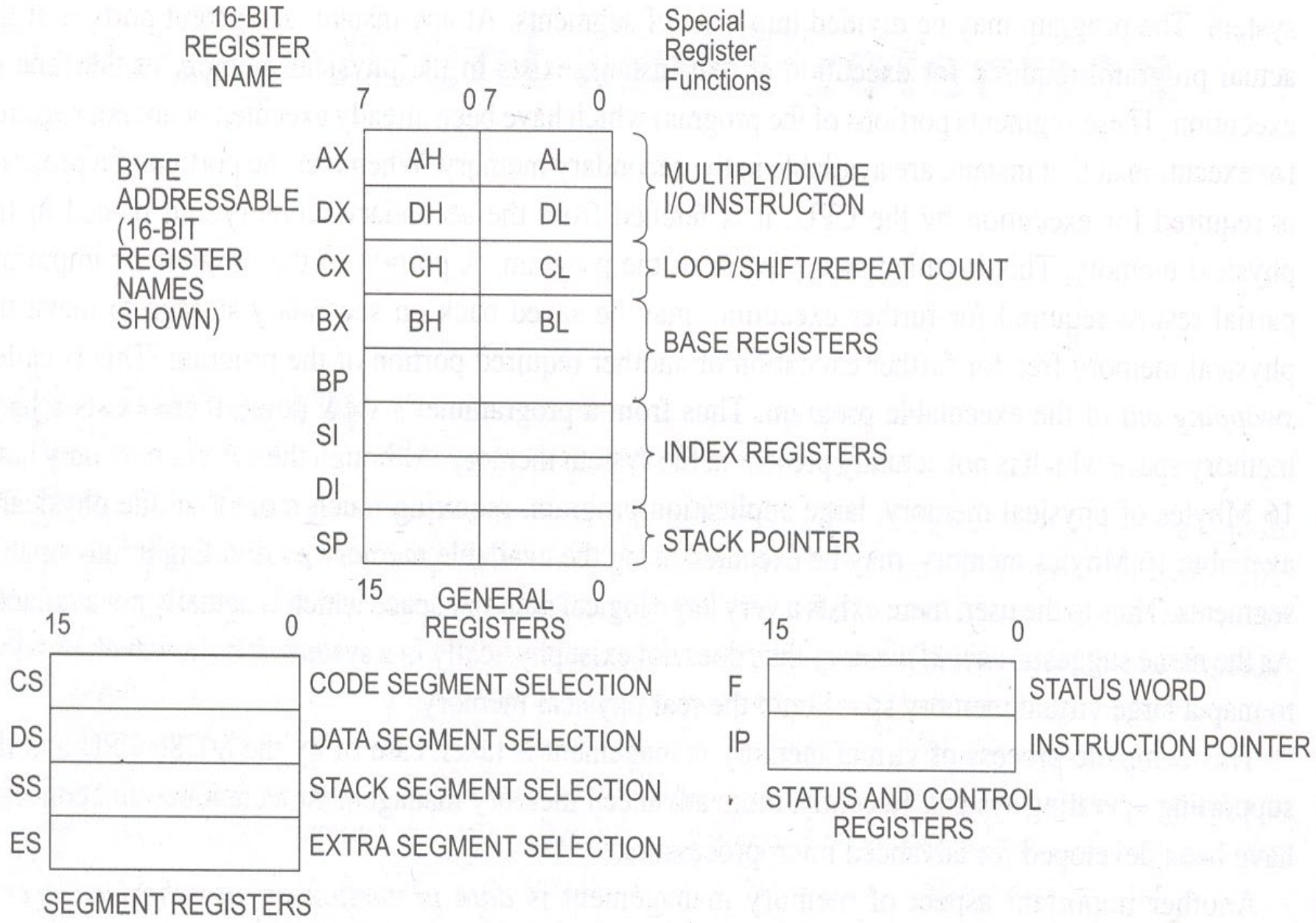
- Calculate the physical addresses of the instruction and data that the CPU want to access
- Address lines derived by this unit may be used to address different peripherals.
- Physical address computed by the address unit is handed over to the BUS unit.

Register organization of 80286

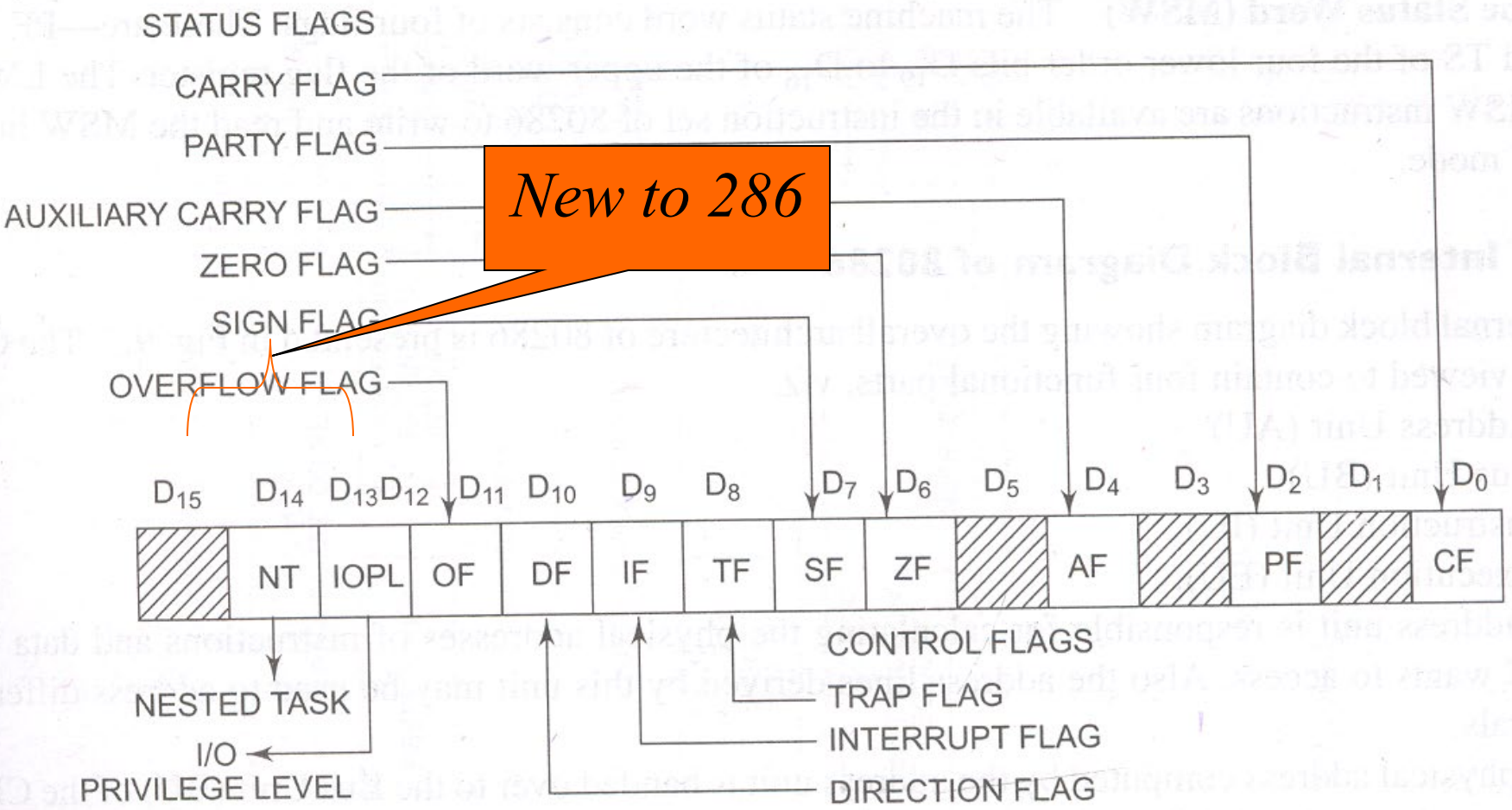
- The 80286 CPU contains the **same set of registers, as in 8086**.
  - Eight 16-bit general purpose registers.
  - Four 16 bit segment registers.
  - One 16-bit Flag register.
  - One 16-bit Instruction pointer.

plus

  - one new 16-bit machine status word (MSW) register


16-BIT REGISTER NAME
Special Register Functions
7 0 7 0
BYTE ADDRESSABLE (16-BIT REGISTER NAMES SHOWN)
AX AH AL
DX DH DL
CX CH CL
BX BH BL
BP
SI
DI
SP
MULTIPLY/DIVIDE I/O INSTRUCTON
LOOP/SHIFT/REPEAT COUNT
BASE REGISTERS
INDEX REGISTERS
STACK POINTER
15 0
GENERAL REGISTERS
15 0
CS CODE SEGMENT SELECTION
DS DATA SEGMENT SELECTION
SS STACK SEGMENT SELECTION
ES EXTRA SEGMENT SELECTION
SEGMENT REGISTERS
15 0
F STATUS WORD
IP INSTRUCTION POINTER
STATUS AND CONTROL REGISTERS

Flag Register



- IOPL – Input Output Privilege Level flags (bit D12 and D13)
  - IOPL is used in protected mode operation to select the privilege level for I/O devices. IF the current privilege level is higher or more trusted than the IOPL, I/O executed without hindrance.
  - If the IOPL is lower than the current privilege level, an interrupt occurs, causing execution to suspend.
  - Note that IPOL **00** is the **highest or more trusted;** and IOPL **11 is the lowest or least trusted.**

➢ NT – Nested task flag (bit D14)

- When set, it indicates that one system task has invoked another through a CALL instruction as opposed to a JMP.
- For multitasking this can be manipulated to our advantage

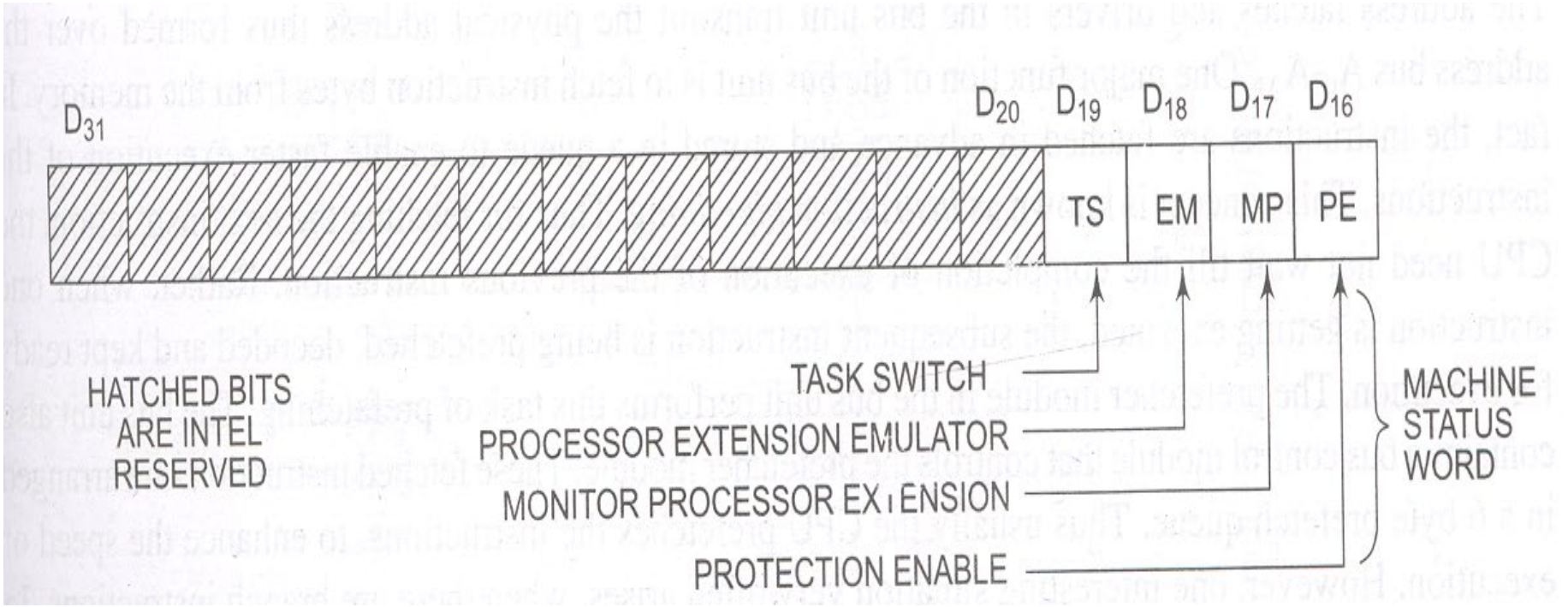
Machine Status Word(**MSW**) Register

- Consist of four flags
  - PE,
  - MP,
  - EM and
  - TS

are for the most part used to indicate whether a processor extension (co-processor) is present in the system or not

- LMSW & SMSW instruction are available in the instruction set of 80286 to write and read the MSW in real address mode
  - LMSW (**Load Machine Status Word)**
  - SMSW(**Store Machine Status Word)**

Machine Status Word...



- PE - Protection enable
  - Protection enable flag places the 80286 in protected mode, if set. this can only be cleared by resetting the CPU.

- MP – Monitor processor extension
  - flag allows WAIT instruction to generate a processor extension.

- EM – Emulate processor extension flag,
  - if set , causes a processor extension absent exception and permits the emulation of processor extension by CPU.

- TS – Task switch
  - if set, this flag indicates the next instruction using extension will generate exception 7, permitting the CPU to test whether the current processor extension is for current task.

80286 signals

- 68 pins are there, instead of 40 in 8086
- data and address bus are de-multiplexed
- mostly like 8086 with some differences

Privilege levels of 80286

- The privilege levels are numbered **0 through 3**.
- Level 0 is the most privileged level whereas Level 4 is the least privileged level.
- Privilege levels provide protection within a task.

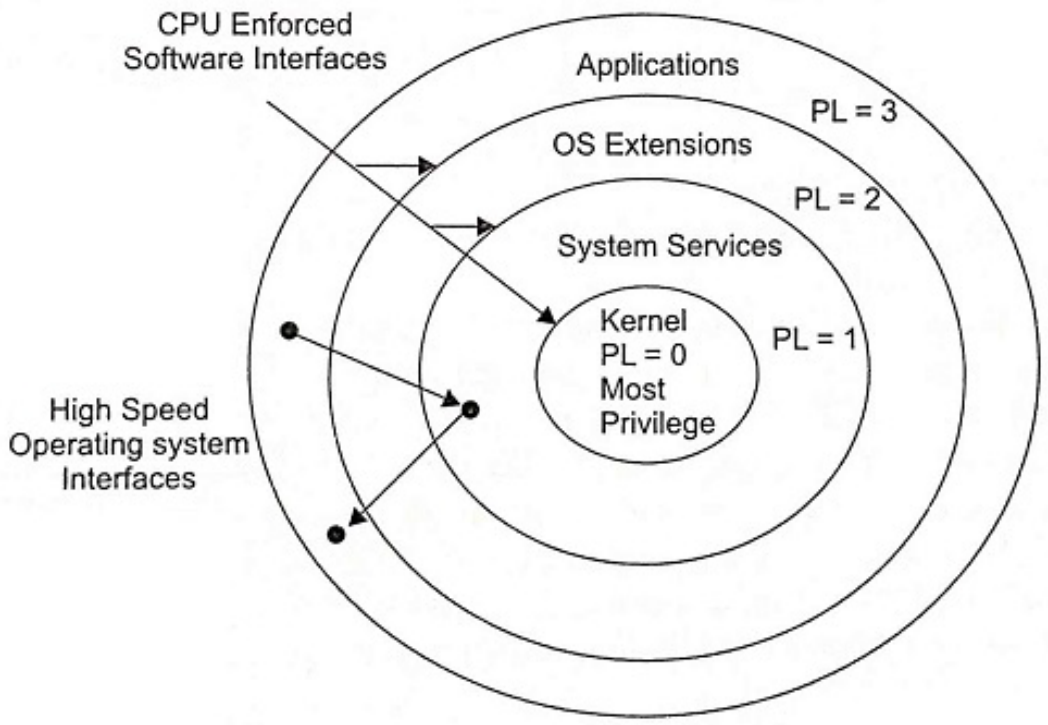
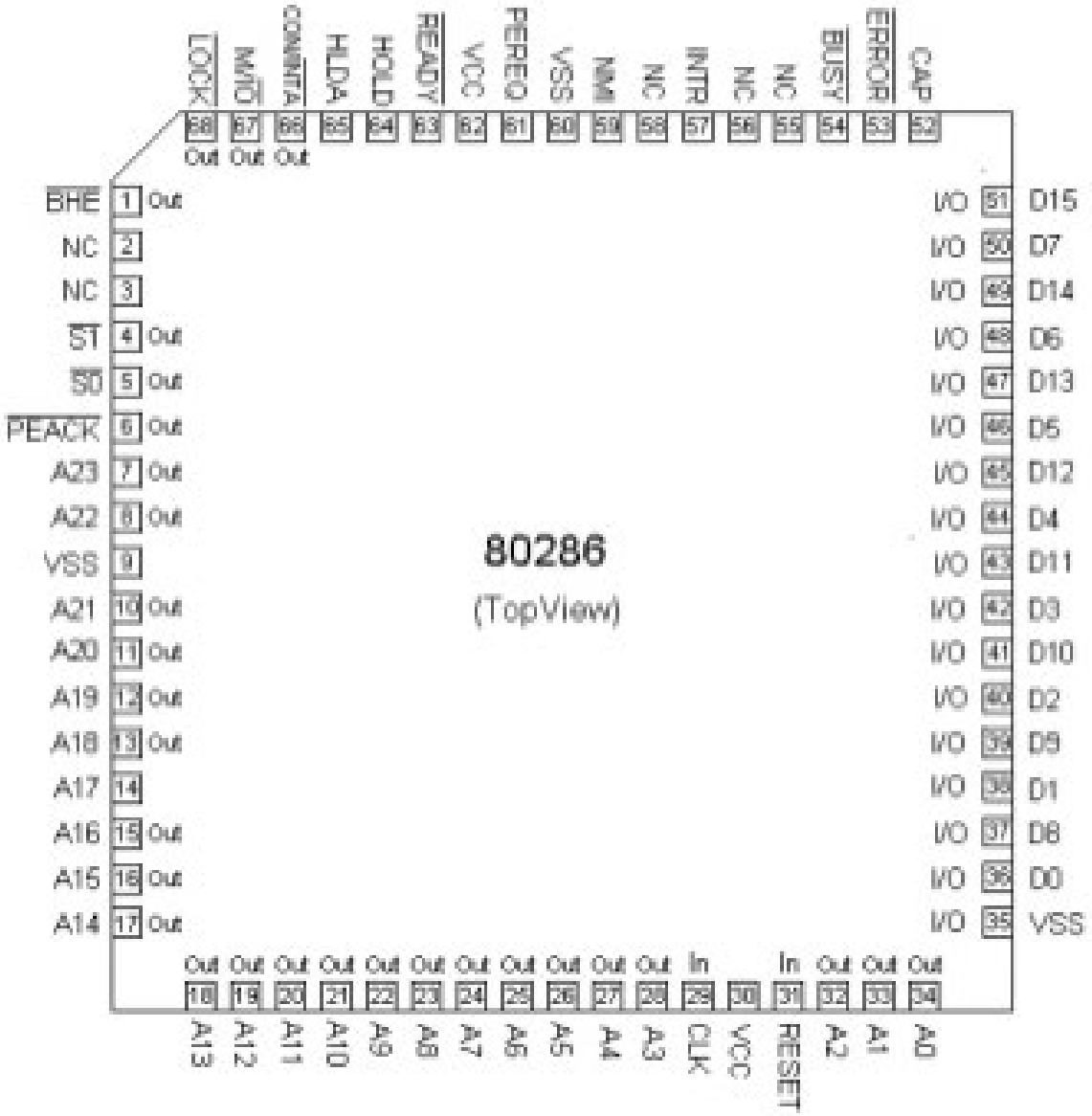


Fig. 11.30 *Four-level privilege*

Privilege levels of 80286

- Every segment is assigned a privilege level
- protection of important code/data such as OS routines and system data from being corrupted by user applications
- protect one application's code/data from other applications
- 4 privilege levels – 0, 1, 2 & 3
  - ✓Level 0 = highest privilege
    - usually assigned to OS Kernel routines, device driver routines and their associated data/stack
  - ✓Level 3 = lowest privilege
    - usually allotted to user applications
  - ✓Levels 1 & 2
    - usually allotted to OS Shell and Utilities

80286 pin diagram



80286 signals...

- some new signals are provided for special jobs like connecting co-processor with 286
  - 4 pins are provided to interface 286 with a co-processor
  - They are:
    - ✓$\overline{\text{PEREQ}}$ [**P**rocessor **E**xtension **REQ**uest]
    - ✓$\overline{\text{PEACK}}$ [**P**rocessor **E**xtension **ACK**nowledge]
    - ✓$\overline{\text{BUSY}}$
    - ✓$\overline{\text{ERROR}}$

80286 signals...

- $\overline{\text{PEREQ}}$ [**P**rocessor **E**xtension **REQ**uest]
  - Input signal to μP
  - Asserted by co-processor to tell μP to perform data transfer to or from memory for it

80286 signals…

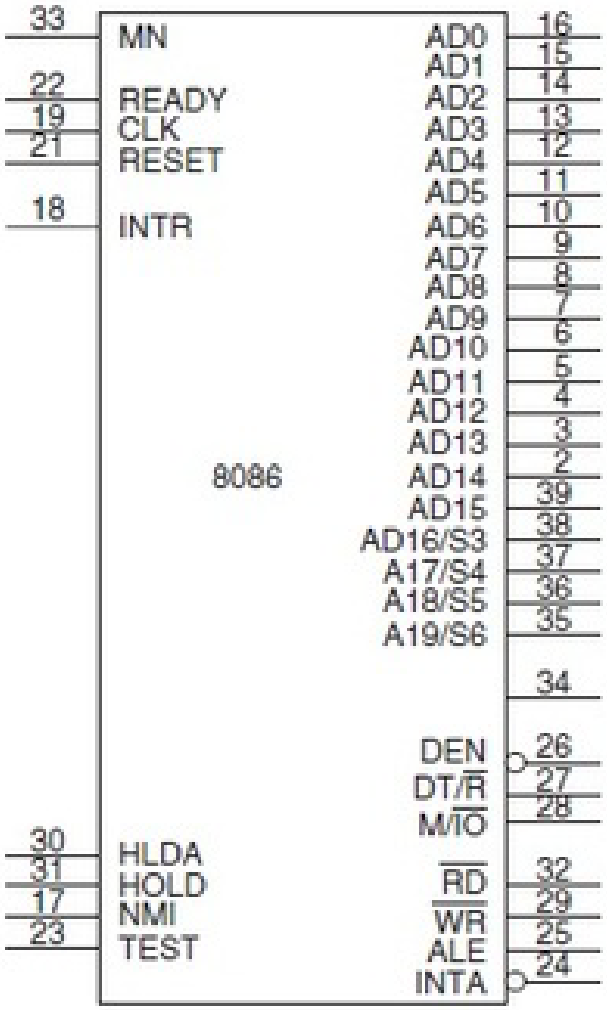
- $\overline{\text{PEACK}}$ [**P**rocessor **E**xtension **ACK**nowledge]
  - Output signal
  - Used to let the co-processor know that the data transfer has started
  - Data transfer are done through µP so that the co-processor can use the protection and virtual memory capability

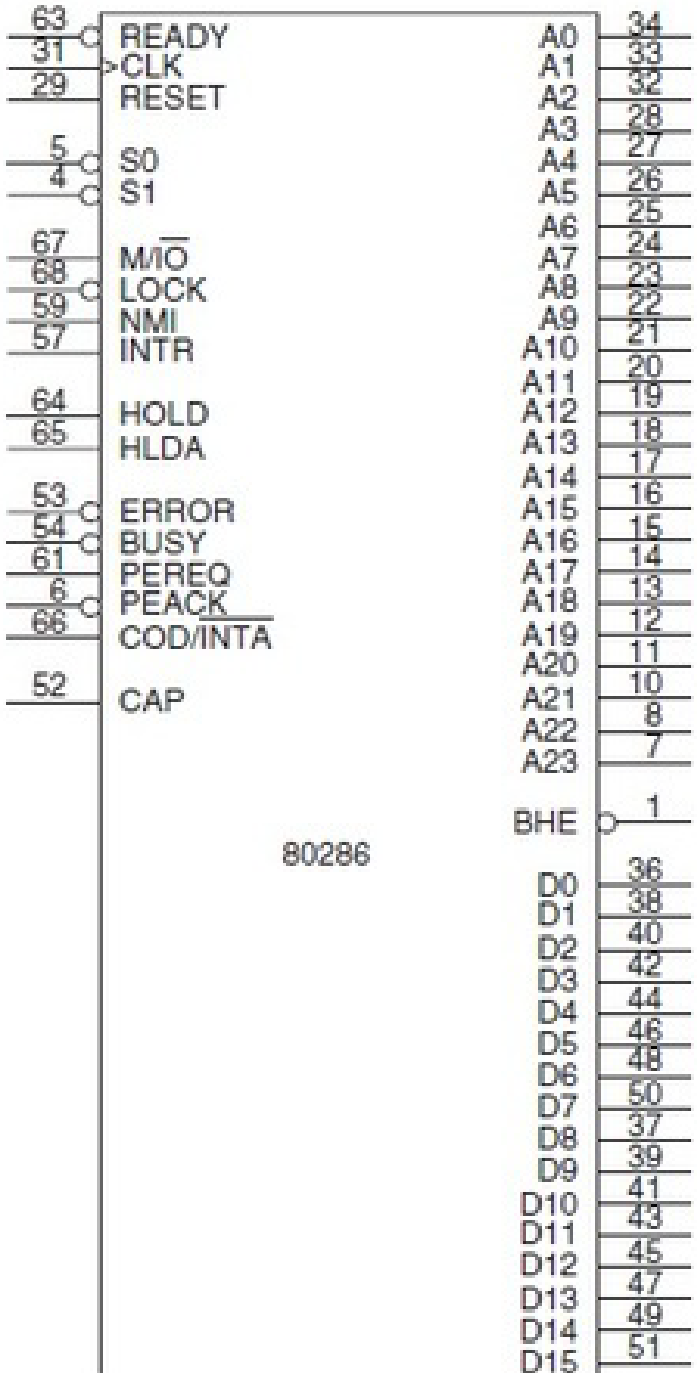
80286 signals…

- $\overline{\text{BUSY}}$
  - Input signal
  - Same as $\overline{\text{TEST}}$ of 8086
  - Enters in a WAIT loop till µP finds a *high* from co-processor

80286 signals...

- $\overline{\text{ERROR}}$
  - Input signal
  - If a co-processor finds some error during processing, it will let μP know through this line


33
MN
22
READY
19
CLK
21
RESET
18
INTR
30
HLDA
31
HOLD
17
NMI
23
TEST
8086
AD0 16
AD1 15
AD2 14
AD3 13
AD4 12
AD5 11
AD6 10
AD7 9
AD8 8
AD9 7
AD10 6
AD11 5
AD12 4
AD13 3
AD14 2
AD15 39
A16/S3 38
A17/S4 37
A18/S5 36
A19/S6 35
34
DEN 26
DT/R 27
M/IO 28
RD 32
WR 29
ALE 25
INTA 24


63
READY
31
CLK
29
RESET
5
S0
4
S1
67
M/IO
68
LOCK
59
NMI
57
INTR
64
HOLD
65
HLDA
53
ERROR
54
BUSY
61
PEREQ
6
PEACK
66
COD/INTA
52
CAP
80286
A0 34
A1 33
A2 32
A3 28
A4 27
A5 26
A6 25
A7 24
A8 23
A9 22
A10 21
A11 20
A12 19
A13 18
A14 17
A15 16
A16 15
A17 14
A18 13
A19 12
A20 11
A21 10
A22 8
A23 7
BHE 1
D0 36
D1 38
D2 40
D3 42
D4 44
D5 46
D6 48
D7 50
D8 37
D9 39
D10 41
D11 43
D12 45
D13 47
D14 49
D15 51

80286 Memory Organization

- Same as 8086
- Uses odd and even banks

Addressing Modes

- Same as 8086

INSTRUCTION SET

- Same as 8086 with some additional instructions

Additional Instructions of Intel 80286

Sl no	Instruction	Purpose
1.	CLTS	Clear the Task – Switched bit
2.	LDGT	Load Global Descriptor Table register
3.	SGDT	Store Global Descriptor Table register
4.	LIDT	Load Interrupt Descriptor Table register
5.	SIDT	Store Interrupt Descriptor Table register
6.	LLDT	Load Local Descriptor Table register
7.	SLDT	Store Local Descriptor Table register
8.	LMSW	Load Machine Status register
9.	SMSW	Store Machine Status register

Additional Instructions of Intel 80286

Sl no	Instruction	Purpose
10.	LAR	Load Access Rights
11.	LSL	Load Segment Limit
12.	SAR	Store Access Right
13.	ARPL	Adjust Requested Privilege Level
14.	VERR	Verify a Read Access
15.	VERW	Verify a Write Access

➢CLTS

- ✓The clear task – switched flag instruction clears the TS (Task - switched) flag bit to a logic 0.

➢LAR

- ✓The load access rights Instruction reads the segment descriptor and place a copy of the access rights byte into a 16 bit register.

➢LSL

- ✓The load segment limit instruction Loads a user – specified register with the segment limit.

- VERR
  - The verify for read access instruction verifies that a segment can de read.
- VERW
  - The verify for write access instruction is used to verify that a segment can be written.
- ARPL
  - The Adjust request privilege level instruction is used to test a selector so that the privilege level of the requested selector is not violated.

INTERRUPT

- Same as 8086 but defines more dedicated internal interrupts
  - Uses from type 5 to type 13 and type 16

80286
Memory Management

80286 Memory Addressing

- Memory management is supported by a hardware unit called Memory management unit.
- Intel's 80286 is the first CPU to incorporate the *Integrated memory management unit.*

80286 Memory Addressing

- Function of memory management unit :

1. To ensure smooth execution of the program by
   - SWAPPING IN data from secondary memory to physical memory
   - SWAPPING OUT data from physical memory to secondary memory

2. Important aspect of memory management is Data Protection or unauthorized access prevention done with the help of segmented memory

REAL MODE MEMORY ADDRESSING

- 80286 operates in either the
  - ✓**real** or
  - ✓**protected** mode.
- **Real mode operation** allows addressing of *only the first* **1M** byte of **memory space**—even in Pentium 4 or Core2 microprocessor.
  - the first 1M byte of memory is called the **real memory, conventional memory**, or **DOS memory** system

Real Address Mode

- Act as a fast 8086
- Instruction set is upwardly compatible
- It address only 1 M byte of physical memory using A0-A19.
- In real addressing mode of operation of 80286, it just acts as a fast 8086. The instruction set is upward compatible with that of 8086.
- *The 80286 addresses only 1Mbytes of physical memory using A0- A19. The lines A20-A23 are not used by the internal circuit of 80286 in this mode.*
- *In real address mode, while addressing the physical memory, the 80286 uses BHE along with A0- A19.*
- *The 20-bit physical address is again formed in the same way as that in 8086.*

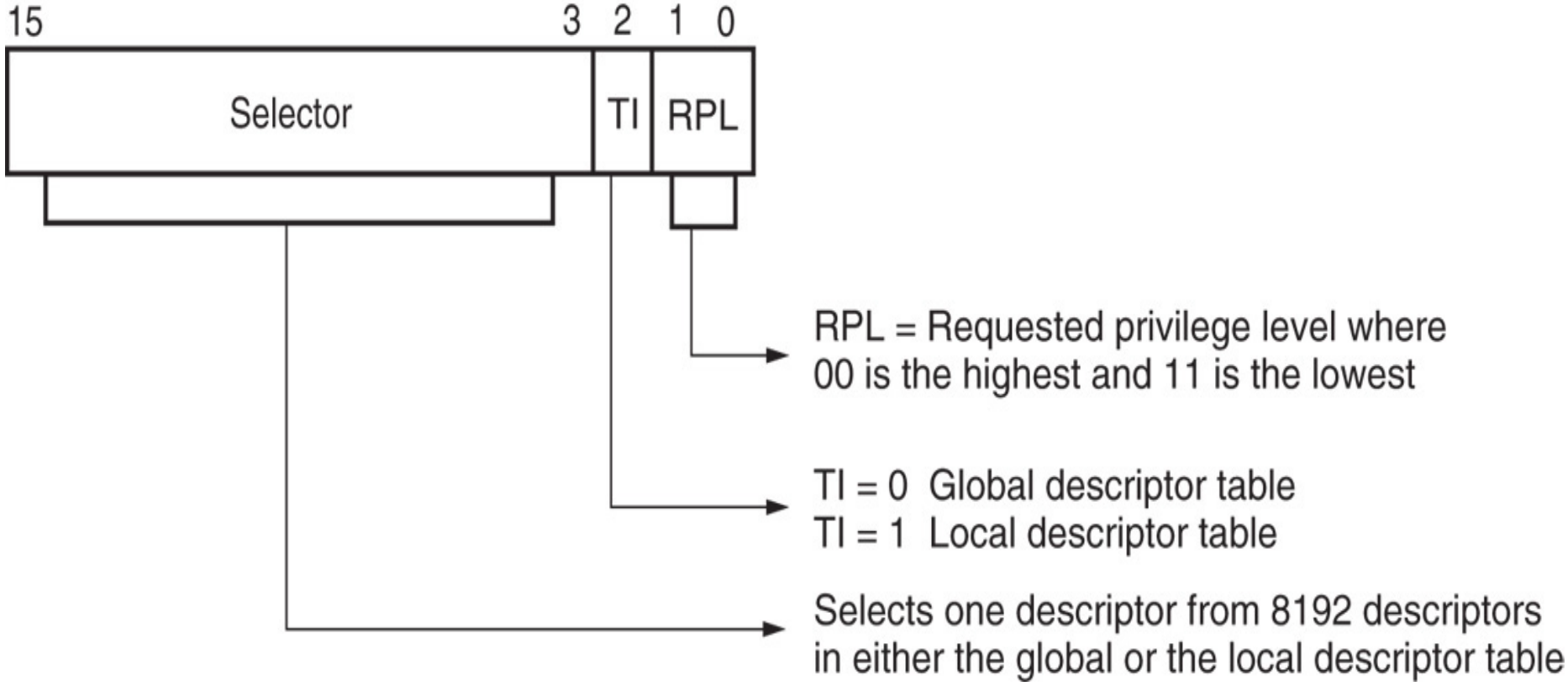
Segments and Offsets

- All real mode memory addresses must consist of a segment address plus an offset address.
  - ✓ **segment address** defines the beginning address of any **64K-byte** memory segment
  - ✓ **offset address** selects any location within the **64K byte** memory segment

PROTECTED MODE MEMORY ADDRESSING

- Allows access to data and programs located *within & above the first 1M byte of memory*.
- **Protected mode** is where Windows operates.
- In place of a segment address, the segment register contains a **selector** that selects a **descriptor** from a **descriptor table**.
- The **descriptor** describes the memory segment's location, length, and access rights.

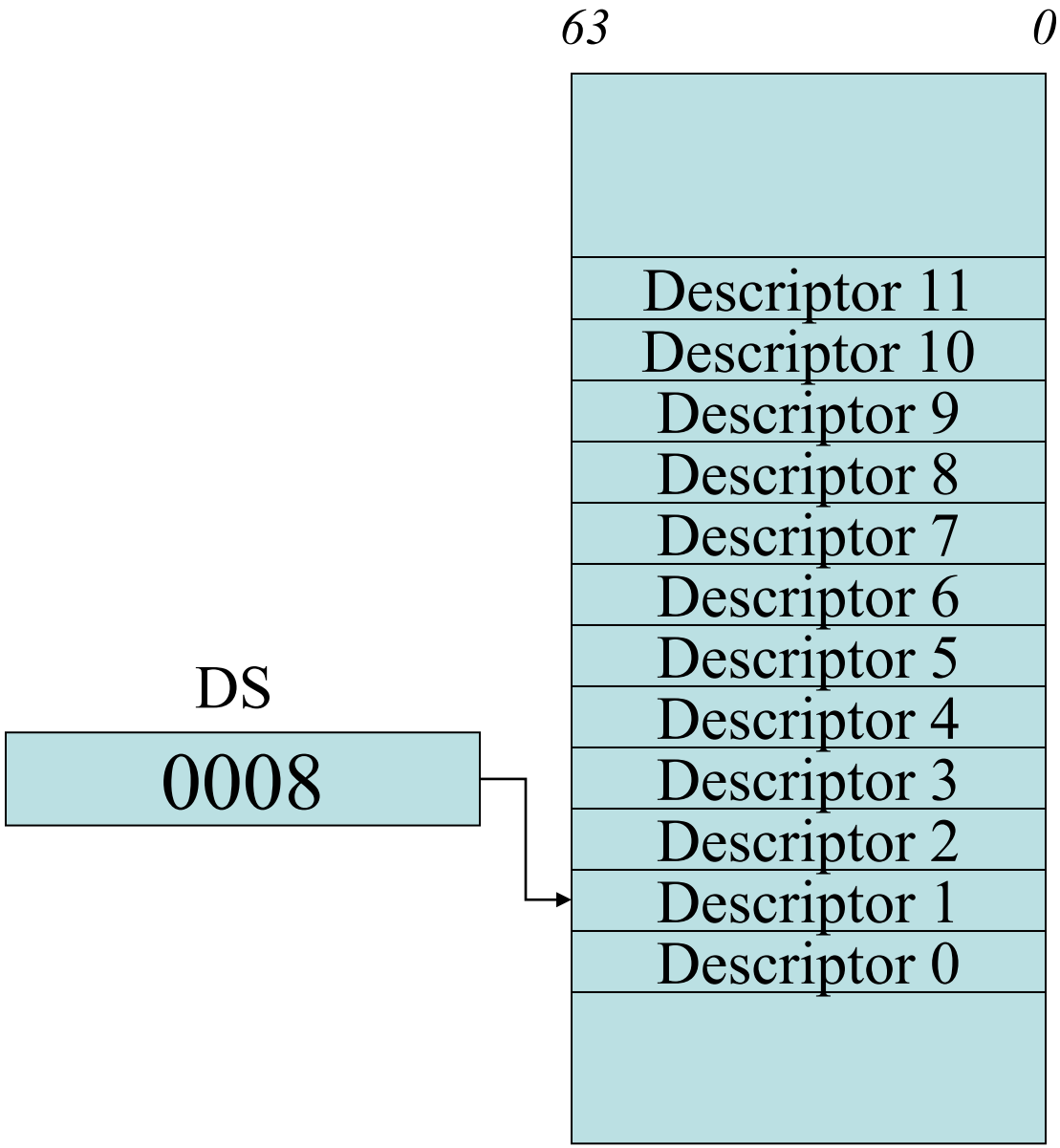
Contents of segment register



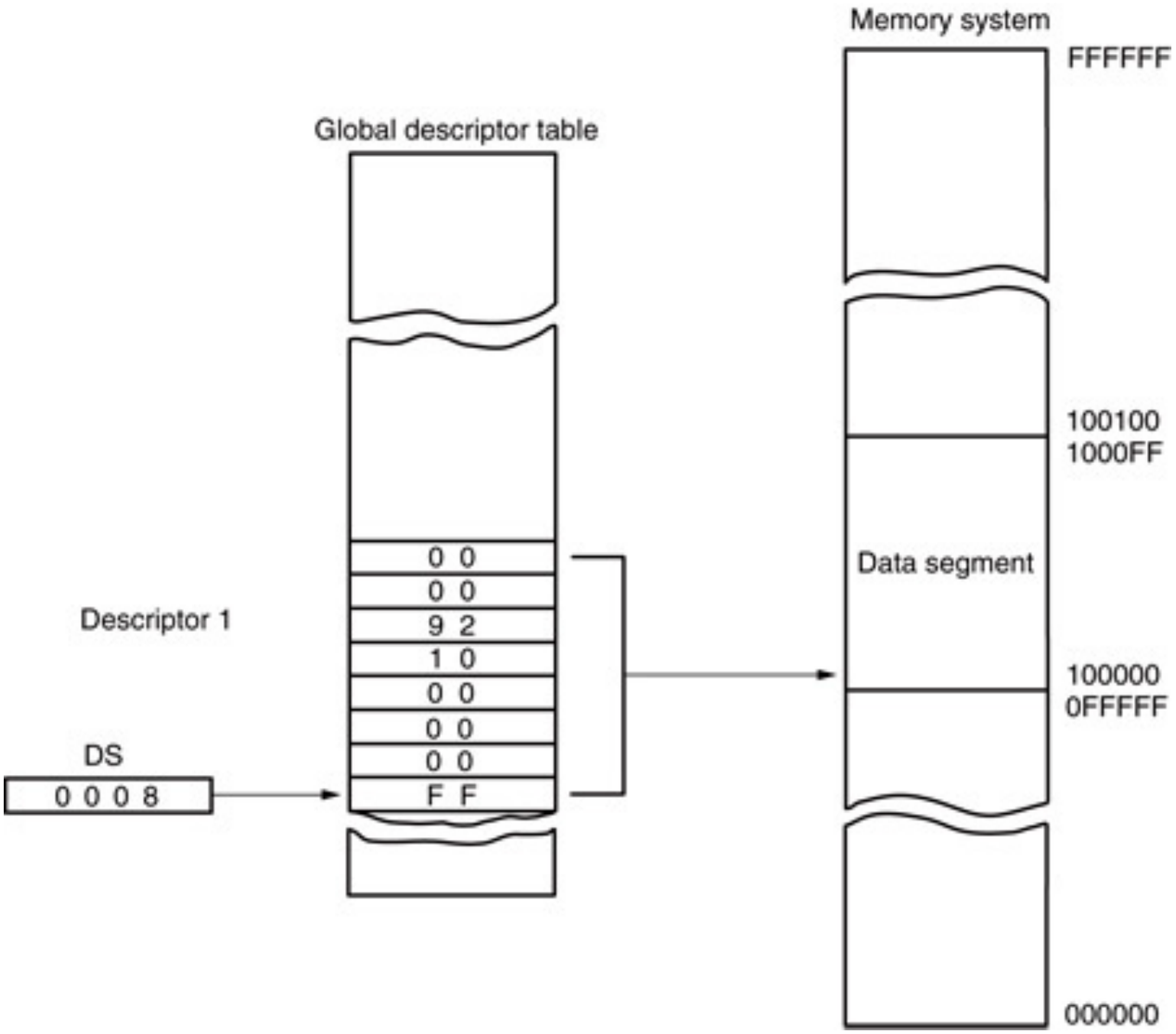
- Since segment descriptors are each 8 bytes, the last three bits of the selector is zero, in which one of them is used for LDT/GDT access.

- Descriptors are chosen from the descriptor table by the segment register.
  - ✓ register contains a 13-bit selector field, a table selector bit, and requested privilege level field
- The **TI bit** selects either the global or the local descriptor table.
- **Requested Privilege Level** (RPL) requests the access privilege level of a memory segment.
  - ✓ If privilege levels are violated, system normally indicates an application or privilege level violation

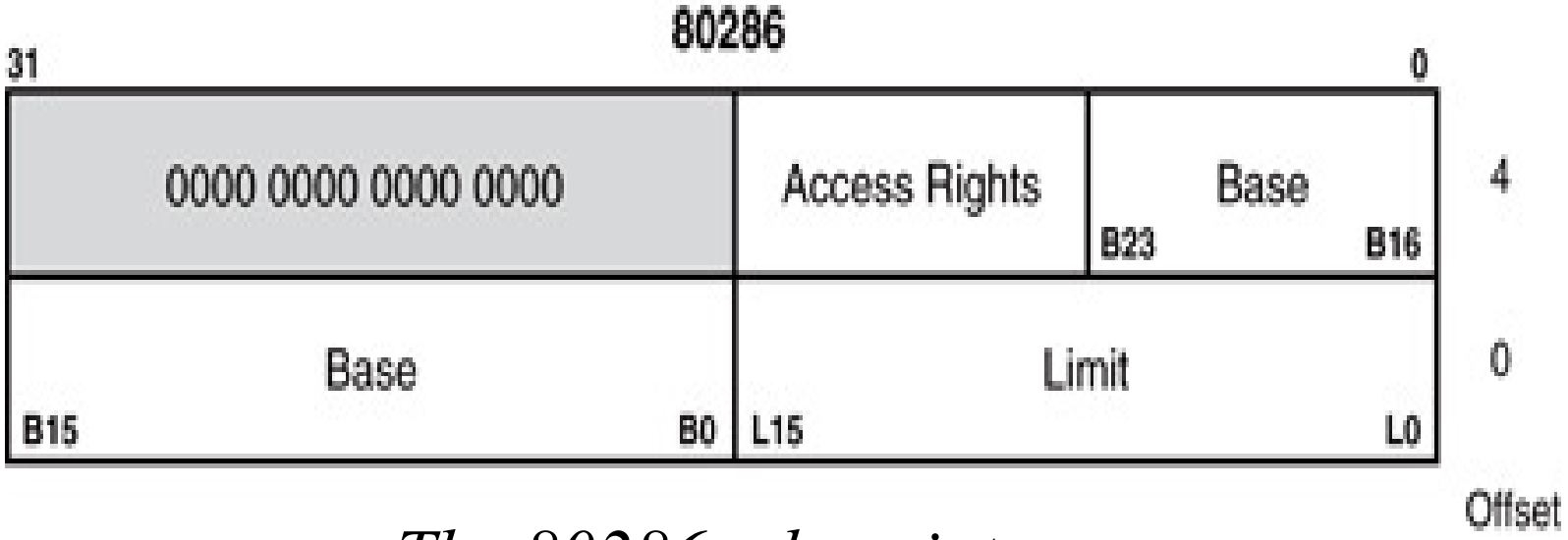
- Following slide shows how the segment register, containing a selector, chooses a descriptor from the global descriptor table.
- The entry in the global descriptor table selects a segment in the memory system.
- **Descriptor zero** is called the **null descriptor**, must contain all zeros, and may not be used for accessing memory.


63
0
Descriptor 11
Descriptor 10
Descriptor 9
Descriptor 8
Descriptor 7
Descriptor 6
Descriptor 5
Descriptor 4
Descriptor 3
Descriptor 2
Descriptor 1
Descriptor 0
DS
0008

- Using the DS register to select a description from the global descriptor table. In this example, the DS register accesses memory locations 00100000H–001000FFH as a data segment.



- Following slide shows the format of a descriptor for the 80286
  - ✓each descriptor is 8 bytes in length
  - ✓global and local descriptor tables are a maximum of 64K bytes in length



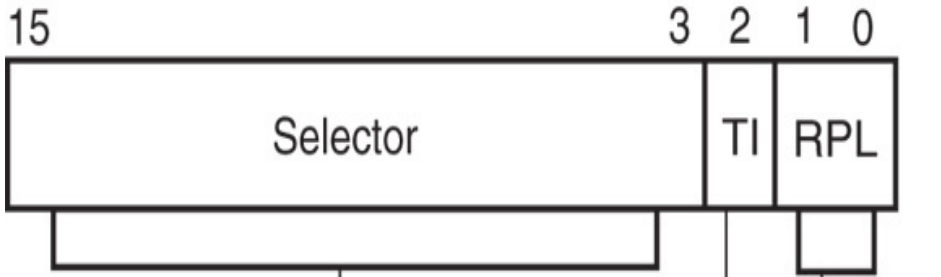
The 80286 descriptors

- 16-bit limit or size allows memory segment lengths of max^m 64K bytes.
- The **base address** of the descriptor indicates the starting location of the memory segment.
  - ✓ the paragraph boundary limitation is removed in protected mode
  - ✓ segments may begin at any address

Selectors &Descriptors

- *Selectors*: used to address a 64 kB memory segment in protected model; a segment register that holds the current descriptor number; 16 bits in length
- 2 bits for *requested privilege level*
- 1 bit for *table indicator* (local or global)
- 13 bits for index (8,192 possible descriptors)

Selectors &Descriptors



RPL = Requested privilege level where
00 is the highest and 11 is the lowest

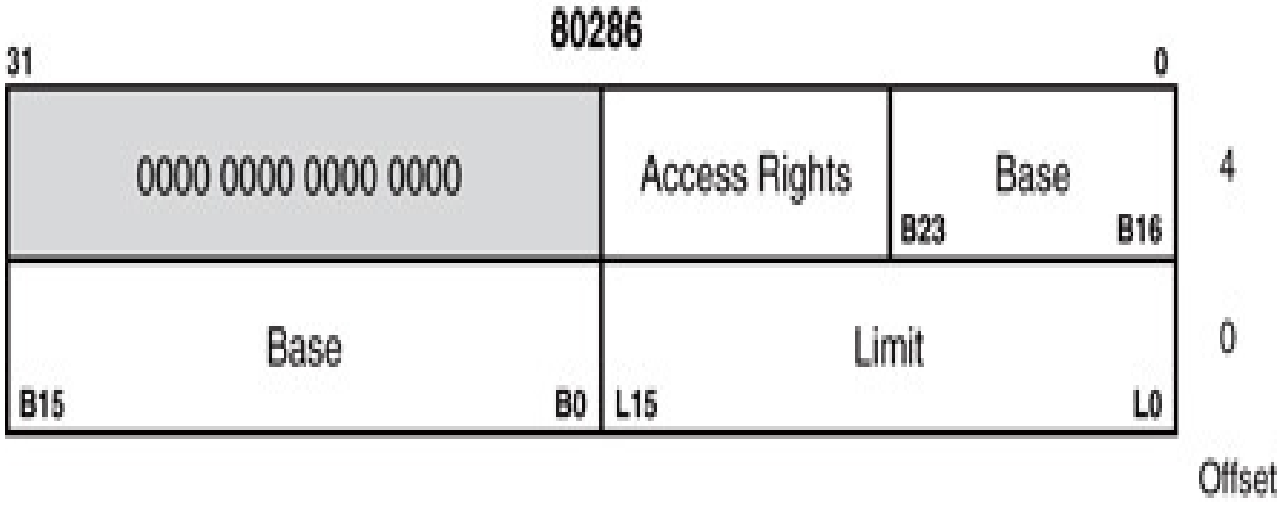
TI = 0 Global descriptor table
TI = 1 Local descriptor table

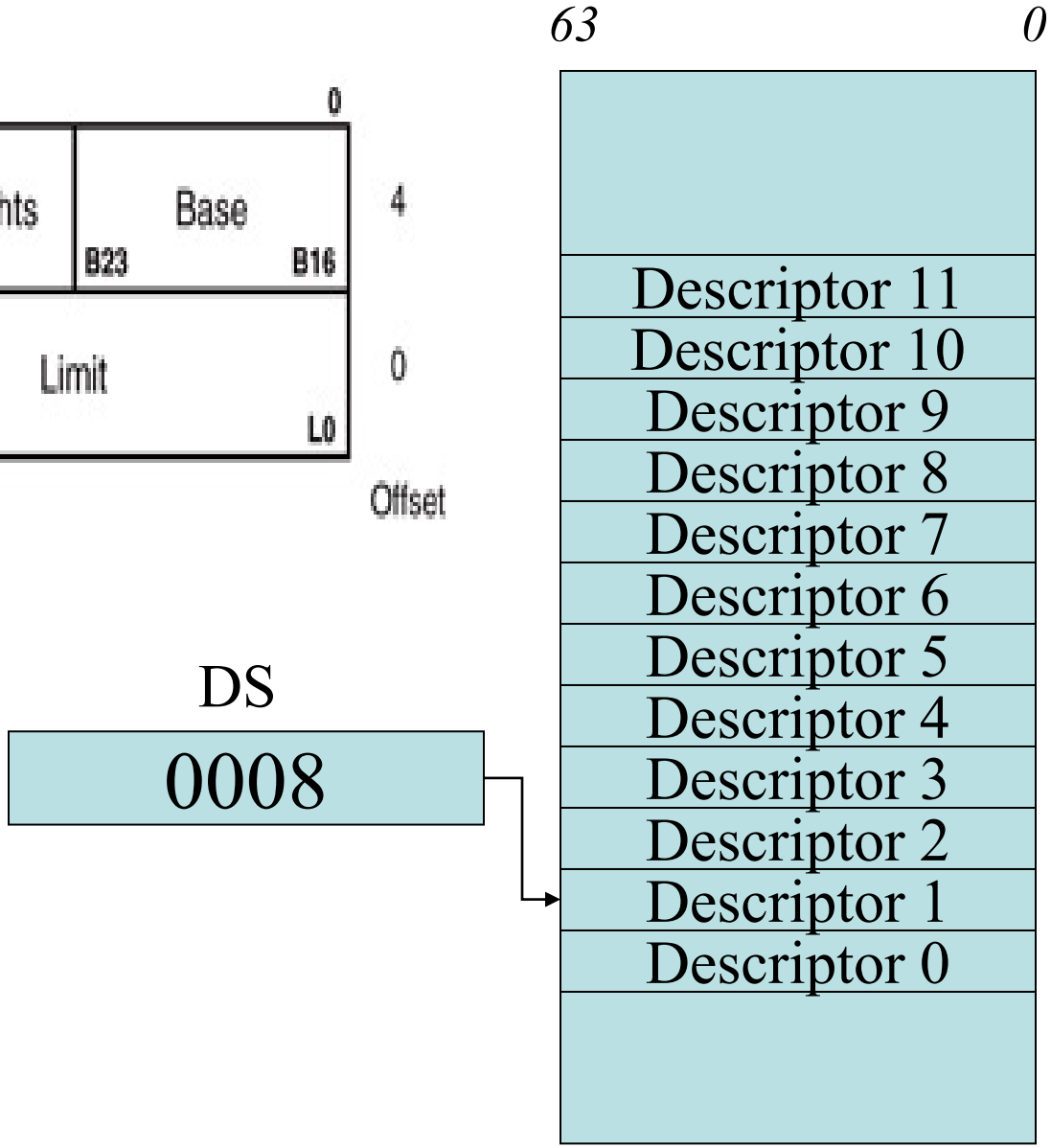
Selects one descriptor from 8192 descriptors
in either the global or the local descriptor table

Descriptors

- *Descriptors*: define the use of memory space (Global, Local, and Interrupt) – 64 bits in length
- 24 bits for *base address*(start of segment)
- 16 bits for *limit* (length of segment)
- 8 bits for *access bits*
- 16 bits unused (zero)

Descriptors





Descriptors

- **Types of Descriptors**
  - ✓Local Descriptor Table (LDT) Descriptor
  - ✓Task State Segment (TSS) Descriptor
  - ✓Gate Descriptors
    - Call Gate Descriptor
    - Task Gate Descriptor
    - Interrupt Gate Descriptor
    - Trap Gate Descriptor

Single-tasking & Multi-tasking

- Single-tasking system
  - ✓ only one program can be run at a time
  - ✓ a second program can be started only after the first program has completed/terminated
- Multi-tasking system
  - ✓ can start many more programs before the previously started program(s) have completed/terminated

Multitasking

- More than one task/program can be loaded in memory & they all of them can exist in memory at the same time
- System must maintain a queue of the programs started by the user
- execute them in some order, based on starting time, priority level, or some other criteria
- To be "fair" to all tasks, each task may be run for a short time, in turn, each task running at different intervals of time and waiting at different intervals of time.
- When one task is waiting for some I/O activity, another task can be executed
- It is also possible for one task to invoke execution of another task that is not part of its own "program“
  - Eg: from within Word or Notepad, we can print a file
    - print function is one of the utilities provided by the OS

Multitasking contd…

- Uniprocessor system
  - more than task may exist in memory at a time
  - only one program can be running at a time
  - other tasks remain suspended in partially-executed state & wait for their turn

- Multiprocessor system (Parallel Computer)
  - more than task may exist in memory at a time
  - many tasks can be running at a time
    - in n-processor system, n tasks can be running
    - at a time - simultaneously - concurrently
  - If >n tasks in memory, others remain suspended in partially-executed state & wait for their turn

Descriptor Cache

- **To allow for fast accesses to segmented memory, the x86 processor keeps a copy of each segment descriptor in a special descriptor cache**. This saves the processor from accessing the GDT for every memory access made. The workings of this cache allows for some interesting side effects.

- As early as the 80286, all Intel x86 processors have included an entity called the "**segment-descriptor cache**" which **works behind the scenes, hidden from you**. It is updated each time a segment register is loaded. It is used for all memory accesses by all Intel x86 processors since the 80286.

Descriptor Cache

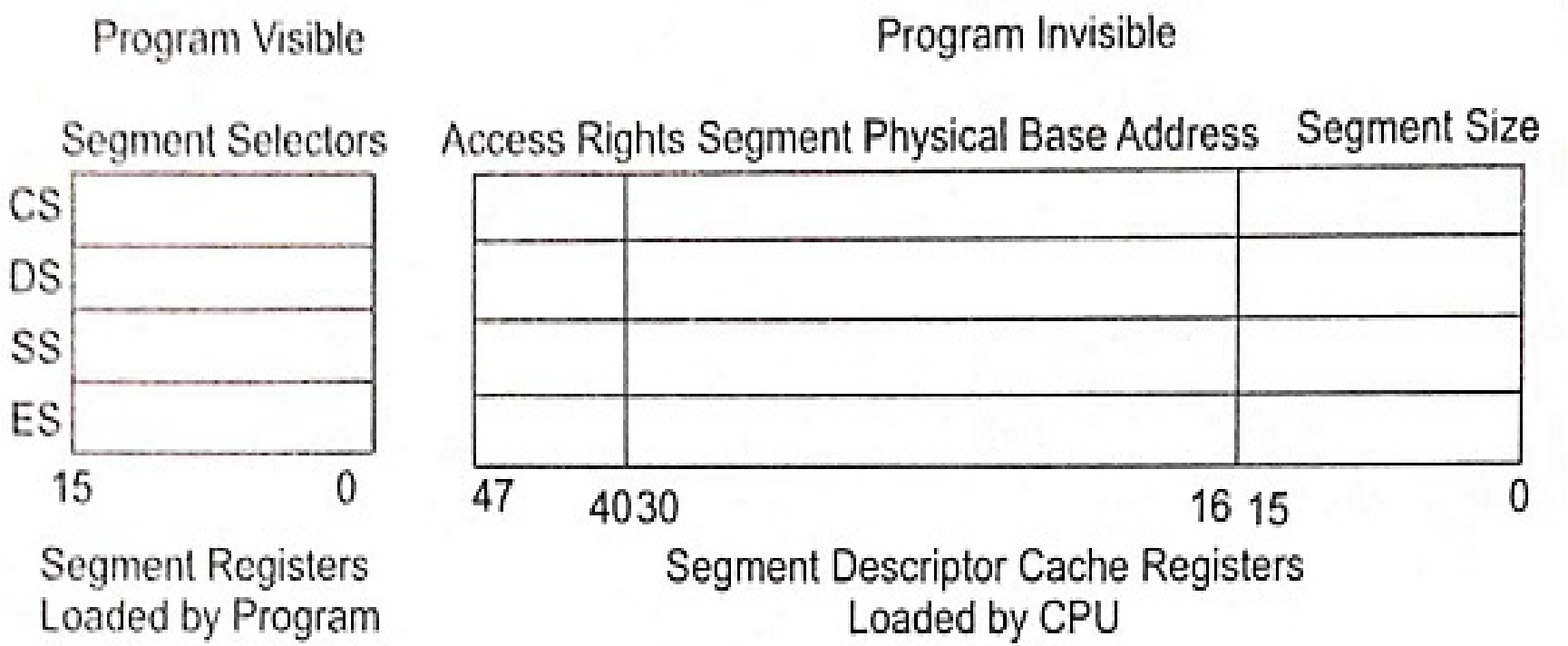


Fig. 11.25 *Descriptor cache registers*

Comparison between 8086 and 80286

	Aspect	8086	80286
1	Total No. of Pins	40	68
2	Clock Speed	3 to 6 MHz.	5 to 12 MHz.
3	Address Bus Width	20 bits	24 bits
4	Address & Data pins	Multiplexed	Not multiplexed
5	Max physical memory	1 Mb	16 Mb
6	Memory Segment Size	Fixed. 64 Kb.	Variable from 0 to 64 Kb.
7	Segment Start Address	Must be multiple of 16. Only upper 16 bits are stored in Segment Register.	Can be any location. All 24 bits are stored in Segment Descriptor.

Comparison - 8086 and 80286

8	Virtual Memory	Nil	1 Gb
9	Pipelining	6-byte Instruction Queue.	6-byte IQ; separate Queue to hold upto 3 Decoded Instructions.
10	Operating Modes	Two: Minimum & Maximum. Hardware configured. Cannot change from one mode to another while working.	Two: Real & Protected. Software configured. Can change from one mode to other while working.
11	Multitasking	No	Yes
12	Privilege Levels for Protection	No Privilege Levels; No built-in Protection.	Four Privilege Levels for Protection.
13	Action upon recognising an Interrupt	Saves only Return Address and Flags, before executing ISR.	Saves the entire Environment of the current Task, before executing ISR.