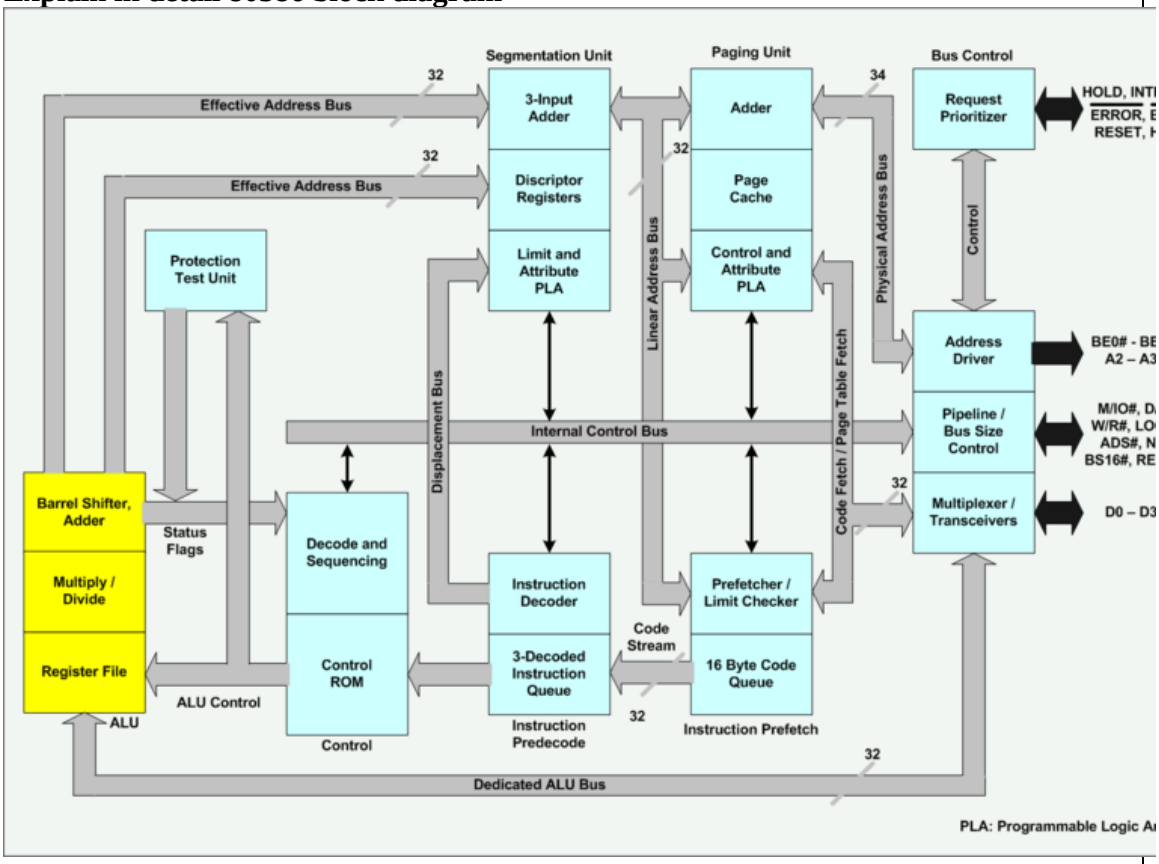
Sinhgad Institutes **SMT. KASHIBAI NAVALE COLLEGE OF ENGG.** **Department of Computer Engineering** **SEM I (2013-14)** **Oral Question bank**	
1.	Explain the difference between 8086 and 8386
2.	**What are the features of 8086?**
3.	**What are the features of 80386?** **Ans.** Features of 80386 are given below: 275,000 transistors Intel's first practical 32-bit microprocessor 32-bit data bus and memory address 4GB of memory Memory management unit Multitasking
4.	**What is an instruction queue? Explain?** **Ans.** This is introduced in 8086 processor. This queue is in the BIU and is used for storing the predecoded instructions. This will overlap the fetching and execution cycle. The E U will take the instructions from the queue for decoding and execution.
5.	**What is stack? Explain the use and operation of stack and stack pointer?** **Ans.** A stack is a portion of the memory used for the temporary storage. A stack is a last In first Out memory. A stack grows in the decreasing order. A stack will hold the temporary information's push and pop are the instructions used for storing and accessing data from the stack. Contents can be moved as 16 bit only using push and pop instructions.
6.	**What are the flags in 8086?** **Ans.** In 8086 Carry flag, Parity flag, Auxiliary carry flag, Zero flag, Overflow flag, Trap flag, Interrupt flag, Direction flag, and Sign flag.
7.	**What are the various interrupts in 8086? Explain.** **Ans.** Maskable interrupts, Non-Maskable interrupts. i) An interrupt that can be turned off by the programmer is known as Maskable interrupt. ii) An interrupt which can be never be turned off (ie.disabled) by the programmer is known as Non-Maskable interrupt.
8.	**What are the flags in 80386?** Explain each flags with its function
9.	**Which interrupts are generally used for critical events?** **Ans.** Non-Maskable interrupts are used in critical events. Such as Power failure, Emergency, Shut off etc.
10.	**Explain different types of registers in 8086 microprocessor arch.** **Ans.** Most of the registers contain data/instruction offsets within 64 KB memory segment. There are four different 64 KB segments for instructions, stack, data and extra data. To specify where in 1 MB of processor memory these 4 segments are located the processor uses four segment registers: **Code segment** (CS) is a 16-bit register containing address of 64 KB segment with processor instructions. The processor uses CS segment for all accesses to instructions referenced by instruction pointer (IP) register. CS register cannot be changed directly. The CS register is automatically updated during far jump, far call and far return instructions. **Stack segment** (SS) is a 16-bit register containing address of 64KB segment with program

	stack. By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment. SS register can be changed directly using POP instruction. **Data segment** (DS) is a 16-bit register containing address of 64KB segment with program data. By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment. DS register can be changed directly using POP and LDS instructions. **Extra segment** (ES) is a 16-bit register containing address of 64KB segment, usually with program data. By default, the processor assumes that the DI register references the ES segment in string manipulation instructions. ES register can be changed directly using POP and LES instructions. It is possible to change default segments used by general and index registers by prefixing instructions with a CS, SS, DS or ES prefix. All general registers of the 8086 microprocessor can be used for arithmetic and logic operations. The general registers are: **Accumulator** register consists of 2 8-bit registers AL and AH, which can be combined together and used as a 16-bit register AX. AL in this case contains the low-order byte of the word, and AH contains the high-order byte. Accumulator can be used for I/O operations and string manipulation. **Base** register consists of 2 8-bit registers BL and BH, which can be combined together and used as a 16-bit register BX. BL in this case contains the low-order byte of the word, and BH contains the high-order byte. BX register usually contains a data pointer used for based, based indexed or register indirect addressing. **Count** register consists of 2 8-bit registers CL and CH, which can be combined together and used as a 16-bit register CX. When combined, CL register contains the low-order byte of the word, and CH contains the high-order byte. Count register can be used as a counter in string manipulation and shift/rotate instructions. **Data** register consists of 2 8-bit registers DL and DH, which can be combined together and used as a 16-bit register DX. When combined, DL register contains the low-order byte of the word, and DH contains the high-order byte. Data register can be used as a port number in I/O operations. In integer 32-bit multiply and divide instruction the DX register contains high-order word of the initial or resulting number. The following registers are both general and index registers: **Stack Pointer** (SP) is a 16-bit register pointing to program stack. **Base Pointer** (BP) is a 16-bit register pointing to data in stack segment. BP register is usually used for based, based indexed or register indirect addressing. **Source Index** (SI) is a 16-bit register. SI is used for indexed, based indexed and register indirect addressing, as well as a source data address in string manipulation instructions. **Destination Index** (DI) is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data address in string manipulation instructions.
11.	**What are the different addressing modes of 80386 with examples.** **Ans.** Addressing modes of 80386: Register addressing: MOV ECX, EDX Immediate addressing: MOV EBX, 12345678H Direct addressing: MOV CX, LIST Register indirect addressing: MOV AL, [ECX] Base-plus-index addressing: MOV [EAX+EBX], CL Register relative addressing: MOV AX, [ECX+4] Base relative-plus-index addressing: MOV EAX, ARRAY [EBX+ECX] Scaled-index addressing: MOV EDX, [EAX+4*EBX]

12.	**What are the different addressing modes of 8086**
13.	**Explain in detail 80386 block diagram** 
14.	**Explain in detail 8086 block diagram**
15.	**Explain the difference between segmentation and paging**
16.	**Explain the terms** **1) Logical address** **2) Physical address** **3) Linear address**
17.	**Explain the function of EU of 8086**
18.	**Explain the function of BIU of 8086**
19.	**How address can be generated in 8086**
20.	**What are different operating modes in 8086 and 80386** Ans :- 8086 Min mode and Max mode And for 80386 Real Mode, Protected mode and Virtual mode
21.	**Discuss different register set of 8-386**
22.	**Explain the difference 8086 segmentation and 80386 segmentation**
23.	**What is segment?**
24.	**What do you mean by segment selector**
25.	**Define the term segment descriptor**

26.	**What are the different segment descriptor table** LDT, GDT, IDT
27.	**Explain descriptor format of LDT,GDT,IDT**
28.	**What do you mean by flat model** a "flat" model of memory organization, the applications programmer sees a single array of up to 2^{32} bytes (4 gigabytes). While the physical memory can contain up to 4 gigabytes, it is usually much smaller; the processor maps the 4 gigabyte flat space onto the physical address space by the address translation mechanisms
29.	**Explain in detail protection mechanism**
30.	**Explain the different privileged level.**
31.	**What are the different data types of 80386**
32.	**Discuss system descriptor and non-system descriptor**
33.	**What is descriptor cache?**
34.	**What do you mean by register operand and memory operand?**
35.	**How effective address can be calculate?**
36.	**Explain the algorithm of all assignments**
37.	**Explain difference between 16 bit , 32 bit and 64 bit programming.**
38.	**Explain int21 h and int80h w.r.t its function numbers.**
39.	**List different sys_call with its functions**
40.	**Explain the term page table, page table entry, page fault**
41.	**Discuss in detail 8086 and 80386 programing model**
42.	**How to run program using TASM and NASM**
43.	**How to run program of 32 bit and 64 bit program**
44.	**Explain in detail TLB**
45.	**Explain in detail Task state segment**
46.	**Explain in detail Task register**
47.	**What are the different gate descriptor** There are four kinds of gate descriptors: ● Call gates ● Trap gates ● Interrupt gates ● Task gates
48.	**Discuss stack switching**
49.	**Define the term clock, clock period, bus cycle, timing diagrams**
50.	**Explain in details instructions used in assignments**