Advanced Microprocessor

Chettinad
College of Engineering & Technology

NH – 67, Karur – Trichy Highways, Puliyr C.F, 639 114 Karur District

B-E- ELECTRONICS AND COMMUNICATION ENGG

SEMESTER VII

080290062-ADVANCED MICROPROCESSOR

COURSE MATERIAL

(TWO MARK QUESTIONS WITH ANSWERS & UNIVERSITY QB)
UNIT I STRUCTURE OF ADVANCED MICROPROCESSORS

General Structure of Microprocessors - Microprocessor Architecture – Instruction Set - Data Formats - Instruction Formats – Addressing Modes – Memory Hierarchy- Register File –Cache- Virtual Memory- Paging-Segmentation-Pipelining- The Instruction Pipeline- Pipeline Hazards- Instruction Parallelism –RISC versus CISC – RISC Properties- RISC Evaluation

UNIT II THE 80386 AND 80486 MICROPROCESSOR

80386 Microprocessors – Special 80386 Registers – 80386 Memory Management – Moving to Protected Mode – Virtual 8086 Mode – The Memory Paging Mechanism – 80486 Microprocessor - 80386 Addressing Modes – Instruction Set

UNIT III THE PENTIUM MICROPROCESSOR

The Software model of the Pentium-Pentium processor registers, Data organisation, instruction types, Addressing modes–Pentium instructions - Interrupt processing-Programming the Pentium.

UNIT IV PENTIUM HARDWARE

CPU pin description-RISC concepts-Bus operation-The Pentium’s superscalar architecture-pipelining-Branch prediction-The instruction and data codes-The floating point unit-Protected mode operation-Segmentation-Paging-Protectionmultitasking-Exception-Input/output-Virtual 8086 mode-Pentium II and Beyond.

UNIT V THE MOTOROLA M68000 FAMILY

The MC680X0 Architecture – CPU Registers – Data Formats – Addressing Modes – Instruction Set and Assembly Directives – Memory Management – Instruction and Data Cache – Exception Processing
TEXT BOOKS:

REFERENCES:
BASICS OF MICROPROCESSORS

**Microprocessor:**
The Central Processing Unit (CPU) of a microcomputer. CPU on a single chip.

**Microprocessor Development System:**
A tool for designing and debugging both hardware and software for microcomputer-based system.

**Microprocessor-Halt_DMA:** Data transfer is performed between the microcomputer’s memory and a peripheral device either by completely stopping the microprocessor or by a technique called cycle stealing.

**Microprogramming:** The microprocessor can use microprogramming to design the instruction set. Each instruction in the Instruction register initiates execution of a micro program stored typically in ROM inside the control unit to perform the required operation.

**Monitor:** Consists of a number of subroutines grouped together to provide “intelligence” to a microcomputer system. This intelligence gives the microcomputer system the capabilities for debugging a user program, system design, and displays.

**Multiprocessing:** The process of executing two or more programs in parallel, handled by multiple processors all under common control. Typically each processor will be assigned specific processing tasks.

**Multitasking:** Operating system software that permits more than one program to run on a single microprocessor. Even though each program is given a small time slice in which to execute, the user has the impression that all tasks (different programs) are executing at the same time.

**Multiuser:** Describes a computer operating system that permits a number of users to access the system on a time-sharing basis

**Nanomemory:** Two-level ROM used in designing the control unit.

**Nested Subroutine:** A commonly used programming technique in which one subroutine calls another subroutine

**Nibble:** A 4-bit word.

**Non-inverting Buffer:** Microprocessor Theory and Applications with 68000/68020 and Pentium. Input is same as output. Current amplifier.
Nonmaskable Interrupt: Occurrence of this type of interrupt cannot be ignored by microcomputer and even though interrupt capability of the microprocessor is disabled. Its effect cannot be disabled by instruction.

Non-Multiplexed: A non-multiplexed microprocessor pin that assigns a unique function as opposed to a multiplexed microprocessor pin defining two functions on timeshared basis.

Object Code: The binary (machine) code into which a source program is translated by a compiler, assembler, or interpreter.

Ones Complement:
Obtained by changing 1’s to 0’s, and 0’s to 1’s of a binary number.

One-Pass Assembler:
This assembler goes through the assembly language program once and translates the assembly language program into a machine language program. This assembler has the problem of defining forward references. See Two-Pass Assembler.

Op Code (Operation Code): Part of an instruction defining the operation to be performed.

Operand: A datum or information item involved in an operation from which the result is obtained as a consequence of defined addressing modes. Various operand types contain information, such as source address, destination address, or immediate data.

Operating System:
Typical resources include microprocessors, disks, and printers. Consists of a number of program modules to provide resource management.

Page:
Some microprocessors, divide the memory locations into equal blocks. Each of these blocks is called a page and contains several addresses.

Parallel Operation:
Any operation carried out simultaneously with a related operation.

Parallel Transmission:
Each bit of binary data is transmitted over a separate wire.

Parity: The number of 1’s in a word is odd for odd parity and even for even parity

Peripheral: An I/O device capable of being operated under the control of a CPU. Examples include disk drives, keyboards, CRT’s, printers, and modems. through communication channels
Personal Computer: Low-cost, affordable microcomputer normally used by an individual for word processing and Internet applications.

Physical Address Space: Address space is defined by the address pins of the microprocessor.

Absolute Addressing: This addressing mode specifies the address of data with the instruction.

Accumulator: Available with 8-bit microprocessors. Register used for storing the result after most ALU operations.

Address: A unique identification number (or locator) for source or destination of data. An address specifies the register or memory location of an operand involved in the instruction.

Addressing Mode: Address of source and destination operands in an instruction. The manner in which a microprocessor determines the effective.

Address Register: A register used to store the address (memory location) of data.

Address Space: The number of storage location in a microcomputer’s memory that can be directly addressed by the microprocessor. The addressing range is determined by the number of address pins provided with the microprocessor chip.

American Standard Code for Information Interchange (ASCII): It is commonly used with microprocessors for representing alphanumeric codes.

Analog-to-Digital Converter (ADC): An 8-bit code Transforms an analog voltage into its digital equivalent.

AND gate: The output is 1, if all inputs are 1; otherwise the output is 0.

ASIC: Normally reduces the total manufacturing cost of a product by reducing chip count. Application Specific IC. Chips designed for a specific, limited application.

Assembler: A program that translates an assembly language program into a machine language program.

Assembly Language: A type of microprocessor programming language that uses a semi-English-language statement Microprocessor Theory and Applications with 68000/68020 and Pentium

Asynchronous Operation: The execution of a sequence of steps such that each step is initiated upon completion of the previous step.
**Asynchronous Serial Data Transmission:** The transmitting device does not need to be synchronized with the receiving device.

**Autodecrement Addressing Mode:** The contents of the specified microprocessor register are first decremented by n (1 for byte, 2 for 16-bit, and 4 for 32-bit) and then the resulting value is used as the address of the operand.

**Autoincrement Addressing Mode:**

The contents of a specified microprocessor register are used as the address of the operand first and then the register contents are automatically incremented by n (1 for byte, 2 for 16-bit, and 4 for 32-bit).

**Base address:**

An address that is used to convert all relative addresses in a program to absolute (machine) addresses

**Baud Rate:** Rate of data transmission in bits per second.

**Binary-Coded Decimal (BCD):**

The representation of 10 decimal digits, 0 through 9, by their corresponding 4-bit binary number.

**Bit:**

A unit of information equal to one of two possible states (one or zero, on or off, true or false). An abbreviation for a binary digit

**Block Transfer DMA:**

A peripheral device requests the DMA transfer via the DMA request line, which is connected directly or through a DMA controller chip to the microprocessor. The DMA controller chip completes the DMA transfer and transfers the control of the bus to the microprocessor.

**Branch:**

The branch instruction allows the computer to skip or jump out of program sequence to a designated instruction either unconditionally or conditionally (based on conditions such as carry or sign).

**Breakpoint:**

Allows the user to execute the section of a program until one of the breakpoint conditions is met. It is then halted. The designer may then single step or examine memory and registers. Typically breakpoint conditions are program counter address or data references. Breakpoints are used in debugging assembly language programs.
Buffer:
A temporary memory storage device designed to compensate for the different data rates between a transmitting device and a receiving device (for example, between a CPU and a peripheral). Current amplifiers are also referred to as buffers.

Bus:
A collection of wires that interconnects computer modules. The typical microcomputer interface includes separate buses for address, data, control, and power functions.

Bus Arbitration:
Bus operation protocols (rules) that guarantee conflict-free access to a bus. Arbitration is the process of selecting one respondent from a collection of several candidates that concurrently request service.

Bus Cycle:
The period of time in which a microprocessor carries out read or write operations.

Cache Memory:
A high speed, directly accessible, relatively small, semiconductor readwrite memory block used to store data instructions that the microcomputer may need in the immediate future. Increases speed by reducing the number of external memory reads required by the processor. Typical 32 and 64-bit microprocessors are normally provided with on-chip cache memory.

Compact Disc (CD) Memory: Optical memory. Uses laser and stores audio information.

Central Processing Unit (CPU): Have register section, and control unit. CPU in a single chip is called microprocessor. The brains of a computer containing the ALU.

Chip: An Integrated Circuit (IC) package containing digital circuits.

CISC: Complex Instruction Set Computer. The Control unit is designed using Microprogramming. Contains a large instruction set. Difficult to pipeline compared to RISC.

Clock:
Timing signals providing synchronization among the various components in a microcomputer system. Analogous to heart beats of a human being.

CMOS: Complementary MOS. Dissipates low power, offers high density and speed compared to TTL.

Combinational Circuit: Output is provided upon application of inputs; contains no memory.
Compiler:
A program which translates the source code written in a high-level Programming language into machine language that is understandable to the processor.

Condition Code Register:
It contains information like carry, sign, zero, and overflow based on ALU operations.

Control Unit:
Microprocessor Theory and Applications with 68000/68020 and Pentium Part of the CPU; its purpose is to translate or decode instructions read (fetched) from the main memory into the Instruction Register.

Coprocessor:
A companion microprocessor that performs specific functions such as floating-point operations independently from the microprocessor to speed up overall operations.

Cycle Stealing DMA:
The DMA controller transfers a byte of data between the microcomputer's memory and a peripheral device such as the disk by stealing a clock cycle of the microprocessor.

Data:
Basic elements of information represented in binary form (that is, digits consisting of bits) that can be processed or produced by a microcomputer. Data represents any group of operands made up of numbers, letters, or symbols denoting any condition, value, or state. Typical microcomputer operand sizes include: a word, which typically contains 2 bytes or 16-bits; a long word, which contains 4 bytes or 32 bits; a quad word, which contains 8 bytes or 64 bits.

Data Register:
A register used to temporarily hold operational data being sent to and from a peripheral device.

Debugger:
A program that executes and debugs the object program generated by the assembler or compiler. The debugger provides a single stepping, breakpoints, and program tracing.

Decoder:
A chip, when enabled, selects one of $2^n$ output lines based on n inputs.

Digital to Analog (D/A) Converter:
Converts binary number to analog signal.
**Diode:** Two terminal electronic switch.

**Direct Memory Access (DMA):**

A type of input/output technique in which data can be transferred between the microcomputer memory and external devices without the microprocessor's involvement.

**Directly Addressable Memory:**

The memory address space in which the microprocessor can directly execute programs. The maximum directly addressable memory is determined by the number of the microprocessor's address pins.

**DRAM (Dynamic RAM):**

Stores data as charges in capacitors and therefore, must be refreshed milliseconds. Hence, requires refresh since capacitors can hold charges for a few circuitry.

**EAROM (Electrically Alterable Read-only Memory):**

Same as EEPROM or EPROM. Can be programmed one line at a time without removing the memory from its sockets. This memory is also called read-mostly memory since it has much slower write times than read times.

**Editor:** A program that produces an error-free source program, written in assembly or high-level languages.

**Emulator:** A hardware device that allows a microcomputer system to emulate (that is, mimic) another microcomputer system.

**Encoder:** Performs reverse operation of a decoder. Contains a maximum of $2^n$ inputs and $n$ outputs.

**EPROM (Erasable Programmable Read-only Memory):**

Can be programmed and erased all programs in an EPROM chip using ultraviolet light. The chip must be removed from the microcomputer system for programming.

**Exception Processing:** Includes the microprocessor's processing states associated with interrupts, trap instructions, tracing, and other exceptional conditions, whether they are initiated internally or externally.

**Exclusive-OR:** The output is 0, if inputs are same; otherwise; the output is 1.

**Exclusive-NOR:** The output is 1, if inputs are same; otherwise, the output is 0.

**Extended Binary-Coded Decimal Interchange Code (EBCDIC):** An 8-bit code commonly used with microprocessors for representing alphanumeric codes. Normally used by IBM.
Firmware:

Microprogram is sometimes referred to as firmware to distinguish it from hardwired control (purely hardware method).

Flag(s): An indicator, often a single bit, to indicate some conditions such as trace, carry, zero, and overflow.

Flash Memory:

Utilizes a combination of EPROM and EEPROM technologies.

Flip-Flop: One-bit memory. Used in cellular phones and digital cameras.

FPGA:

Field Programmable Gate Arrays. This chip contains several smaller individual logic blocks along with all interconnections.

Gate: Digital circuits which perform logic operations.

Handshaking: Data transfer via exchange of control signals between the microprocessor and an external device.

Hardware: The physical electronic circuits (chips) that make up the microcomputer system.

Hardwired Control: Used for designing the control unit using all hardware.

HCMOS: High speed CMOS. Provides high density and consumes low power.

Hexadecimal Number System: Base-16 number system.

High-Level Language: A type of programming language that uses a more understandable human-oriented language such as C.

HMOS: High-density MOS reduces the channel length of the NMOS transistor and provides increased density and speed in VLSI circuits.

Immediate Address: An address that is used as an operand by the instruction itself.

Implied Address: An address is not specified, but is contained implicitly in the instruction.

In-Circuit Emulation:

The most powerful hardware debugging technique; Especially valuable when hardware and software are being debugged simultaneously.
Index:
A number (typically 8-bit signed or 16-bit unsigned) is used to identify a particular element in an array (string). The index value typically contained in a register is utilized by the indexed addressing mode.

Indexed Addressing: The effective address of the instruction is determined by the Sum of the address and the contents of the index register. Used to access arrays.

Index Register:
A register used to hold a value used in indexing data, such as when a value is used in indexed addressing to increment a base address contained within an instruction.

Indirect Address: A register holding a memory address to be accessed.

Instruction: Causes the microprocessor to carry out an operation on data. A program contains instructions and data.

Instruction Cycle: The sequence of operations that a microprocessor has to carry out while executing an instruction.

Instruction Register (IR):
A register storing instructions; typically 32 bits long for a 32-bit microprocessor.

Instruction Set: Lists all the instructions that the microcomputer can execute.

Interleaved DMA: Using this technique, the DMA controller takes over the system bus when the microprocessor is not using it.

Internal Interrupt: Activated internally by exceptional conditions such as overflow and division by zero.

Interpreter: A program that executes a set of machine language instructions in response to each high-level statement in order to carry out the function.

Interrupt I/O: An external device can force the microcomputer system to stop executing the current program temporarily so that it can execute another program known as the interrupt service routine.

Interrupts:
A temporary break in a sequence of a program, initiated externally or internally, causing control to jump to a routine, which performs some action while the program is stopped.

I/O (Input/Output): Describes that portion of a microcomputer system that exchanges data between the microcomputer system and an external device.
**I/O Port:** A register that contains control logic and data storage used to connect a microcomputer to external peripherals.

**Inverting Buffer:** Current amplifier. Performs NOT operation.

**Keyboard:** Has a number of push button-type switches configured in a matrix form (rows x columns).

**Keybounce:** When a mechanical switch opens or closes, it bounces (vibrates) for a small period of time (about 10-20 ms) before settling down.

**Large-Scale Integration (LSI):** An LSI chip contains 100 to 1000 gates.

**LED:** Light Emitting Diode. Typically, a current of 10 ma to 20 ma flows at 1.7 to 2.4 drop across it.

**Logic Analyzer:**
A hardware development aid for microprocessor-based design; gathers data on the fly and displays it.

**Logical Address Space:** All storage locations with a programmer’s addressing range.

**Loops:** A programming control structure where a sequence of microcomputer instructions are executed repeatedly (looped) until a terminating condition (result) is satisfied.

**Machine Code:** Microprocessor Theory and Applications with 68000/68020 and Pentium A binary code (composed of 1’s and 0’s) that a microcomputer understands.

**Machine Language:** A type of microprocessor programming language that uses binary or hexadecimal numbers.

**Macroinstruction:** Commonly known as an instruction; initiates execution of a complete micro program. Example includes assembly language instructions.

**Macroprogram:** The assembly language program.

**Mask:** A pattern of bits used to specify (or mask) which bit parts of another bit pattern are to be operated on and which bits are to be ignored or “masked” out. Uses logical AND operation.

**Mask ROM:**
Programmed by a masking operation performed on the chip during the manufacturing process; its contents cannot be changed by user.

**Maskable Interrupt:**
Can be enabled or disabled by executing typically the interrupt instructions.
**Memory:** Any storage device which can accept, retain, and read back data.

**Memory Access Time:** Average time taken to read a unit of information from the memory.

**Memory Address Register (MAR):** Stores the address of the data.

**Memory Cycle Time:** Average time lapse between two successive read operations.

**Memory Management Unit (MMU):** Hardware that performs address translation and protection functions.

**Memory Map:**
A representation of the physical locations within a microcomputer’s addressable main memory.

**Memory-Mapped I/O:**
I/O ports are mapped as memory locations, with every connected device treated as if it were a memory location with a specific address. Manipulation of I/O data occurs in “interface registers” (as opposed to memory locations); hence there are no input (read) or output (write) instructions used in memory-mapped I/O.

**Microcode:**
A set of instructions called “microinstructions” usually stored in a ROM in the control unit of a microprocessor to translate instructions of a higher-level programming language such as assembly language programming.

**Microcomputer:** Consists of a microprocessor, a memory unit, and an input/output unit.

**Microcontroller:**
Typically includes a microcomputer, timer, A/D (Analog to Digital) and D/A (Digital to Analog) converters in the same chip.

**Microinstruction:**
Most microprocessors have an internal memory called control memory. This memory is used to store a number of codes called microinstructions. These microinstructions are combined to design the instruction set of the microprocessor.

**Pipeline:**
A technique that allows a microcomputer processing operation to be broken down into several steps (dictated by the number of pipeline levels or stages) so that the individual step outputs can be handled by the microcomputer in parallel. Often used to fetch the processor’s next instruction while executing the current instruction, which considerably speeds up the overall operation of the microcomputer. Overlaps instruction fetch with execution.
**Pointer:**
A storage location (usually a register within a microprocessor) that contains the address of (or points to) a required item of data or subroutine.

**Polled Interrupt:**
A software approach for determining the source of interrupt in a multiple interrupt system.

**POP Operation:**
Reading from the top or bottom of stack.

**Port:**
A register through which the microcomputers communicate with peripheral devices.

**Primary or Main Memory:**
Storage that is considered as part of the microcomputer. The microcomputer can directly execute all instructions in the main memory. The maximum size of the main memory is defined by the number of address pins in the microprocessor.

**Privileged Instructions:**
An instruction which can only be executed by the microprocessor in the supervisor (operating system) mode.

**Processor Memory:**
A set of microprocessor registers for holding temporary results when a computation is in progress.

**Program:**
A self-contained sequence of computer software instructions (sourcecode) that, when converted into machine code, directs the computer to perform specific operations for the purpose of accomplishing some processing task. Contains instructions and data.

**Program Counter (PC):**
A register that normally contains the address of the next instruction to be executed in a program.

**Programmed I/O:**
The microprocessor executes a program to perform all data transfers between the microcomputer system and external devices.

**PROM (Programmable Read-only Memory):**
Can be programmed by the user by using proper equipment. Once programmed, its contents cannot be altered.

**Protocol:**
A list of data transmission rules or procedures that encompass the timing, control, formatting, and data representations by which two devices are to communicate. Also known as hardware “handshaking”, which is used to permit asynchronous communication.
**PUSH Operation:**

Microprocessor Theory and Applications with 68000/68020 and Pentium Writing to the top or bottom of stack.

**Random Access Memory (RAM):**

Are volatile in nature (in other words, information is lost when power is removed). A read/write memory. RAMS (static or dynamic)

**Read-Only-Memory (ROM):**

A memory in which any addressable operand can be read from, but not written to, after initial programming. ROM storage is non volatile (information is not lost after removal of power).

**Reduced Instruction Set Computer (RISC):**

A simple instruction set is included. The RISC architecture maximizes speed by reducing clock cycles per instruction. The control unit is designed using hardwired control. Easier to implement pipelining.

**Register:**

A high-speed memory usually constructed from flip-flops that are directly accessible to the microprocessor. It can contain either data or a specific location in memory that stores word(s) used during arithmetic, logic, and transfer operations.

**Register Indirect:** Uses a register which contains the address of data.

**Relative Address:** An address used to designate the position of a memory location

**RISC:** See Reduced Instruction Set Computer. in a routine or program.

**Routine:**

A group of instructions for carrying out a specific processing operation. Usually refers to part of a larger program. A routine and subroutine have essentially the same meaning, but a subroutine could be interpreted as a self-contained routine nested within a routine or program.

**Scalar Microprocessor:**

The 80486 is a scalar microprocessor. Provided with one pipeline. Allows execution rate of one clock cycle per instruction for most instructions.

**Scaling:** Multiplying an index register by 1,2,4 or 8. Used by the addressing modes of typical 32- and 64-bit microprocessors.

**Schmitt Trigger:** An analog circuit that provides high noise immunity.
**SDRAM:**

Control signals and address inputs are sampled by the SDRAM by a common clock. Synchronous DRAM. This chip contains several DRAMS internally. The

**Secondary Memory Storage:**

An auxiliary data storage device that supplements the main (primary) memory of a microcomputer. It is used to hold programs and data that would otherwise exceed the capacity of the main memory. Although it has a much slower access time, secondary storage is less expensive. Examples include floppy and hard disks.

**Sequential Circuit:** Combinational circuit with memory

**PUSH Operation:** Microprocessor Theory and Applications with 68000/68020 and Pentium Writing to the top or bottom of stack.

**Random Access Memory (RAM):**

A readwrite memory. RAMS (static or dynamic) are volatile in nature (in other words, information is lost when power is removed).

**Read-Only-Memory (ROM):**

A memory in which any addressable operand can be read from, but not written to, after initial programming. ROM storage is non volatile information is not lost after removal of power).

**Reduced Instruction Set Computer (RISC):**

A simple instruction set is included. The RISC architecture maximizes speed by reducing clock cycles per instruction. The control unit is designed using hardwired control. Easier to implement pipelining.

**Register:**

A high-speed memory usually constructed from flip-flops that are directly accessible to the microprocessor. It can contain either data or a specific location in memory that stores word(s) used during arithmetic, logic, and transfer operations.

**Register Indirect:** Uses a register which contains the address of data.

**Relative Address:** An address used to designate the position of a memory location in a routine or program.

**RISC:** See Reduced Instruction Set Computer.

**Routine:** A group of instructions for carrying out a specific processing operation. Usually refers to part of a larger program. A routine and subroutine have essentially the same
meaning, but a subroutine could be interpreted as a self-contained routine nested within a routine or program.

**Scalar Microprocessor:** The 80486 is a scalar microprocessor. Provided with one pipeline. Allows execution rate of one clock cycle per instruction for most instructions

**Scaling:**

Multiplying an index register by 1, 2, 4 or 8. Used by the addressing modes of typical 32- and 64-bit microprocessors

**Schmitt Trigger:** An analog circuit that provides high noise immunity.

**SDRAM:**

Control signals and address inputs are sampled by the SDRAM by a common clock. Synchronous DRAM. This chip contains several DRAMS internally. The

**Secondary Memory Storage:**

An auxiliary data storage device that supplements the main (primary) memory of a microcomputer. It is used to hold programs and data that would otherwise exceed the capacity of the main memory. Although it has a much slower access time, secondary storage is less expensive. Examples include floppy and hard disks.

**Sequential Circuit:** Combinational circuit with memory

**Superscalar Microprocessor:**

The Pentium is a superscalar microprocessor. Microprocessor Theory and Applications with 68000/68020 and Pentium Provided with more than one pipeline and executes More than one instruction per clock cycle.

**Supervisor State:**

When the microprocessor processing operations are conducted at a higher privilege level, it is usually in the supervisor state. An operating system typically executes in the supervisor state to protect the integrity of “basic” system operations from user influences.

**Synchronous Operation:**

Operations that occur at intervals directly related to a clock period.

**Synchronous Sequential Circuit:**

The present outputs depend on the present inputs and the previous states stored in flip-flops

**Synchronous Serial Data Transmission:**

Data is transmitted or received based on a clock signal.
Tracing: (Allows single stepping.) A dynamic diagnostic technique permits analysis debugging of the program’s execution.

Transistor: Electronic switch; performs NOT; current amplifier.

Tristate Buffer: 
This chip is typically enabled by a control signal to provide logic 0 or 1 outputs. This type of buffer can also be disabled by the control signal to place it in a high-impedance state. Has three output states: logic 0, 1, and a high-impedance state.

Two’s Complement: 
The two’s complement of a binary number is obtained by replacing each 0 with a 1 and each 1 with a 0 and adding one to the resulting number.

Vectored Interrupts: A device identification technique in which the highest Priority device with a pending interrupt request forces program execution to branch to an interrupt routine to handle exception processing for the device.

Very Large Scale Integration (VLSI): a VLSI chip contains more than 1000 gates. More commonly, a VLSI chip is identified by the number of transistors rather than the gate count.

Virtual Memory: An operating system technique that allows programs or data to exceed the physical size of the main, internal, directly accessible memory of the microcomputer. Program or data segment pages are swapped from external disk storage as needed. The swapping is invisible (transparent) to the programmer. Therefore, the programmer does need not to be concerned with the actual physical size of internal memory while writing the code.

Word: The bit size of a microprocessor refers to the number of bits that can be Processed simultaneously by the basic arithmetic and logic circuits of the microprocessor. A number of bits taken as a group in this manner is called a word.
UNIT 1
GENERAL STRUCTURE OF MICROPROCESSOR

PART A (TWO MARKS)

1. What is Microprocessor? Give the power supply & clock frequency of 8085
   A microprocessor is a multipurpose, programmable logic device that reads binary
   instructions from a storage device called memory accepts binary data as input and processes
   data according to those instructions and provides result as output. The power supply of 8085
   is +5V and clock frequency in 3MHz.

2. List few applications of microprocessor-based system.
   It is used:
   - For measurements, display and control of current, voltage,
   - For temperature, pressure, etc.
   - For traffic control and industrial tool control.
   - For speed control of machines.

3. What is an Opcode?
   The part of the instruction that specifies the operation to be performed is called the
   operation code or opcode.

4. What is an Operand?
   The data on which the operation is to be performed is called as an Operand.

5. What is meant by Wait State?
   This state is used by slow peripheral devices. The peripheral devices can transfer the
   data to or from the microprocessor by using READY input line. The microprocessor remains
   in wait state as long as READY line is low. During the wait state, the contents of the address,
   address/data and control buses are held constant.

6. What is meant by polling?
   Polling or device polling is a process which identifies the device that has interrupted
   the microprocessor.

7. What is meant by interrupt?
   Interrupt is an external signal that causes a microprocessor to jump to a specific
   subroutine.

8. Basic concepts in memory interfacing
   The primary function of memory interfacing is that the microprocessor should be able to
   read from and write into a given register of a memory chip.
   - To perform these operations the microprocessor should
   - Be able to select the chip
   - Identify the register
9. What is an instruction?
   An instruction is a binary pattern entered through an input device to Command the microprocessor to perform that specific function.

10. Explain the different instruction formats with examples
   The instruction set is grouped into the following formats
   - One byte instruction MOV C,A
   - Two byte instruction MVI A,39H
   - Three byte instruction JMP 2345H

11. What is the use of addressing modes, mention the different types
   The various formats of specifying the operands are called addressing modes, it is used to access the operands or data. The different types are as follows
   - Immediate addressing
   - Register addressing
   - Direct addressing
   - Indirect addressing
   - Implicit addressing

12. What is Microcontroller and Micro-computer?
    Microcontroller is a device that includes microprocessor; memory and I/O signal lines on a single chip, fabricated using VLSI technology. Microcomputer is a computer that is designed using microprocessor as its CPU. It includes microprocessor, memory and I/O.

13. What is assembler?
    The assembler translates the assembly language program text which is given as input to the assembler to their binary equivalents known as object code. The time required to translate the assembly code to object code is called access time. The assembler checks for syntax errors & displays them before giving the object code.

14. What are procedures?
    Procedures are a group of instructions stored as a separate program in memory and it is called from the main program whenever required. The type of procedure depends on where the procedures are stored in memory. If it is in the same code segment as that of the main program then it is a near procedure otherwise it is a far procedure.

15. What are Macros?
    Macro is a group of instruction. The macro assembler generates the code in the program each time where the macro is called.
    Macros are defined by MACRO & ENDM directives. Creating macro is similar to creating new opcodes that can be used in the program.
    INIT MACRO
    MOV AX, data
    MOV DS
    MOV ES, AX
    ENDM
16. What is interrupt service routine?
   Interrupt means to break the sequence of operation. While the CPU is executing a program an interrupt breaks the normal sequence of execution of instructions & diverts its execution to some other program. This program to which the control is transferred is called the interrupt service routine.

17. Define BIOS
   The IBM PC has in its ROM a collection of routines, each of which performs some specific function such as reading a character from keyboard, writing character to CRT. This collection of routines is referred to as Basic Input Output System or BIOS.

18. What is memory mapping?
   The assignment of memory addresses to various registers in a memory chip is called as memory mapping.

19. What is I/O mapping?
   The assignment of addresses to various I/O devices in the memory chip is called as I/O mapping.

20. List the major components of the keyboard/Display interface.
   - Keyboard section
   - Scan section
   - Display section
   - CPU interface section

21. What is Key bouncing?
   Mechanical switches are used as keys in most of the keyboards. When a key is pressed the contact bounce back and forth and settle down only after a small time delay (about 20ms). Even though a key is actuated once, it will appear to have been actuated several times. This problem is called Key Bouncing.

22. Define swapping in?
   The portion of a program is required for execution by the CPU, it is fetched from the secondary memory and placed in the physical memory. This is called ‘swapping in’ of the program.

23. What are the CPU contents used in 80286?
   The 80286 CPU contains almost the same set of registers, as in 8086
   - Eight 16-bit general purpose register
   - Four 16-bit segment registers
   - Status and control register
   - Instruction pointer.
24. Define Microprocessor?
A microprocessor is a multipurpose, programmable logic device that reads binary instructions from a storage device called memory accepts binary data as input and processes data according to those instructions and provides result as output. The power supply of 8085 is +5V and clock frequency in 3MHz.

25. What is an instruction set and enlist the types of instructions?
The instruction set is indeed one of the key features of computer architecture, defining and describing the capabilities of any computing system, including microprocessors.
Types as following,
- Data movement instruction
- Integer arithmetic and logic instructions
- Shift and rotate instructions
- Control transfer instructions
- Bit manipulation instructions
- System control instruction
- Floating-point instruction
- Special function unit instruction.

26. Define the following terms:
(a) Two-operand instructions
(b) Three-operand instructions

**Two-operand instructions:**
- ADD arc, dst; (dst) + (src) -> dst
- SUB src, dst; (dst) - (src) -> dst
- MUL src, dst; (dst) * (src) -> dst
- DIV src, dst; (dst) / (src) -> dst

**Eg:** In case of the divide instruction (DIV) a double destination is usually specified to store the quotient and the remainder.

**Three-operand instructions:**
- ADD src1, scr2, dst; (src2) + (src1) -> dst
- SUB src1, scr2, dst; (src2) - (src1) -> dst
- MUL src1, scr2, dst; (src2) * (src1) -> dst
- DIV src1, scr2, dst; (src2) / (src1) -> dst

**Eg:** The use of three-operand instructions results in a more compact code. This can be seen from the following simple example.
Take a HLL expression: C: = A + B; C <- (A) + (B)
27. Enlist the different data-types supported by a microprocessor?
   A typical set of integer data formats as practiced in modern microprocessor,
   - Signed and unsigned
   - ASCII characters, 8 bits each
   - Packed and unpacked binary coded decimal (BCD) numbers, 4 bits each
   There are two ways of ordering the byte addresses within a word:
   - Little-endian
   - Big-endian

28. Enlist the different instruction formats in a microprocessor?
   Instruction formats vary considerably among different microprocessors. In some systems, instructions are formed in units of bytes, in some in units of 16-bit half words (however, in some systems they are called “words”), and in others in units of 32-bit words. The basic information included in an instruction format consists primarily of:
   - Instruction opcode
   - Addresses of operands

29. What do you mean by addressing mode? Enlist the different types of addressing mode.
   The address of the operand in memory is stored in one of registers that is addressing modes.
   - Register addressing modes
   - Immediate addressing modes
   - Direct addressing modes
   - Register indirect addressing modes
   - Register indirect with post increment
   - Register indirect with pre decrement
   - Register indirect with displacement

30. What are the advantages & disadvantages of segmentation?
    **Advantages:**
    - Segmentation supports well-structured software, since segments can contains meaningful information units.
    - Supports more compact code, since reference within a segment can be shorter.
    - Segmentation supports efficient typing of data, thus contribute g to the narrowing of the semantic gap.
    
    **Disadvantages:**
    - Increased hardware complexity
    - Execution time overhead in handling segmentation
    - Extra memory need for segmentation tables
    - Added memory fragmentation for small segments.
31. What is the different between a page & a segment?

<table>
<thead>
<tr>
<th></th>
<th>Page</th>
<th>Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fixed size, or a finite number of fixed</td>
<td>Variable size, from a single byte to the</td>
</tr>
<tr>
<td></td>
<td>sizes</td>
<td>whole physical memory</td>
</tr>
<tr>
<td></td>
<td>Page frames allocated contiguously in</td>
<td>May be placed anywhere in memory and may</td>
</tr>
<tr>
<td></td>
<td>memory, no overlapping</td>
<td>overlap</td>
</tr>
<tr>
<td></td>
<td>Not characterized by type of information</td>
<td>Each segment defined by the type of</td>
</tr>
<tr>
<td></td>
<td></td>
<td>information in it.</td>
</tr>
</tbody>
</table>

32. What do mean by instruction pipeline?
   Where different stages of instruction fetch and execution are in a pipeline.

33. What do mean by arithmetic pipeline?
   Where different stages of an arithmetic operation are handled along the stages of a pipeline.

34. What are the three types of pipeline hazards?
   We can define three types of pipeline hazards.
   - Structural hazards
   - Data hazards
   - Control hazards.

35. What are the four types of pipeline stages?
   The four stages of pipelines are:
   - Fetch
   - Decode
   - Execute
   - Write back.

36. What are RISC processors?
   Opposed to the traditional CISC design, in the early eighties there emerged a new trend of computer design called RISC –reduced instruction set computer.

37. What are CISC processors?
   The control units of such microprocessors are naturally complex, since they have to distinguish between a larger number of opcodes, addressing modes, and formats. type of system belongs to the category called complex instruction set computer(CISC).
Although many CISC microprocessors are pipeline there exist an inherent difficulty in managing a pipe in a system with a variety of instruction sizes and different instruction execution lengths.

38. Basic principles of RISC architecture? give some examples of the RISC & CISC processors.
   The some basic principles of RISC are:
   - HLL support
   - Implementation of register windows
   - Pipelining
   - Delayed branch
   - Score boarding
   - Dual cache
   - ILP

39. Give some advantages of the RISC processors?
   The advantages of RISC will be discussed from a number of points of view:
   - VLSI realization
   - Computing speed
   - Design cost and reliability
   - HLL support.

40. Give some advantages of the CISC processors?
   The advantages of CISC are:
   - It has encountered opposition right from its inception, in the same issue where it has been first from publicly announced.
   - The reason for the success of the RISC idea, despite its criticism, is the proven performance of RISC-type system.

41. Give different between the RISC & the CISC processors?

   **CISC**
   - Instruction Set: large set of instruction with variable size (16 to 64)
   - Addressing Modes: 12-24
   - General Purpose registers: 8-24
   - Clock rate: 33-50MHz in 1992

   **RISC**
   - Instruction Set: Small set of instruction with fixed size (32-bit)
   - Addressing Modes: 3-5
   - General Purpose registers: 32-192
   - Clock rate: 50-150MHz in 1993
42. Different between cache & CPU register file?

<table>
<thead>
<tr>
<th>Cache</th>
<th>CPU register file</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addressed as locations in memory-long addresses</td>
<td>Separate register addressing – short addresses</td>
</tr>
<tr>
<td>Has to be tens of Kbytes to be effective</td>
<td>About 128 register will have significant effect on performance</td>
</tr>
<tr>
<td>Information loaded in units of lines</td>
<td>Information can loaded individually to each register</td>
</tr>
<tr>
<td>Usually inaccessible by the user</td>
<td>Fully accessible by the user</td>
</tr>
<tr>
<td>Possibility of a miss</td>
<td>No miss</td>
</tr>
</tbody>
</table>

43. Mention the Properties of RISC Processor

- The RISC microprocessor is designed using hardwired control with little or no microcode. Note that variable-length instruction formats generally require microcode design. All RISC instructions have fixed formats, so microcode design is not necessary.

- A RISC microprocessor executes most instructions in a single cycle.

- The instruction set of a RISC microprocessor typically includes only register, load, and store instructions. All instructions involving arithmetic operations use registers, and load and store operations are utilized to access memory.

- The instructions have a simple fixed format with few addressing modes.

- A RISC microprocessor has several general-purpose registers.

44. Mention the properties of CISC processor?

- CISC microprocessors, on the other hand, contain a large number of instructions and many addressing modes, while RISC microprocessors include a simple instruction set with a few addressing modes. Almost all computations can be obtained from a few simple operations.

- RISC basically supports a small set of commonly used instructions that are executed at a fast clock rate compared to CISC, which contains a large instruction set (some of which are rarely used) executed at a slower clock rate. To implement the fetch/
execute cycle for supporting a large instruction set for CISC, the clock is typically slower.

In CISC, most instructions can access memory while RISC contains mostly load store instructions.

The complex instruction set of CISC requires a complex control unit, thus requiring micro programmed implementation. RISC utilizes hardwired control which is faster.

CISC is more difficult to pipeline; RISC provides more efficient pipelining.

An advantage of CISC over RISC is that complex programs require fewer instructions in CISC with fewer fetch cycles, while RISC requires a large number of instructions to accomplish the same task with several fetch cycles.

45. What do mean by cache memory?
Cache operation is based on the principle of locality. We can distinguish between two types of locality:

Temporal locality: If an information item is accessed by the CPU, there is a high probability that it will be accessed again in the near future.

Spatial locality: If an information item is accessed, there is a high probability that other items nearby in the program will be accessed in the near future.

46. Mention the types of instruction level parallelism?
The main types of instruction level parallelism are:

- Superscalar
- Super pipeline
- VLIW: Very Long Instruction Word.

47. Comparison of ILP architecture?

<table>
<thead>
<tr>
<th>Feature</th>
<th>Sequential</th>
<th>Dependence</th>
<th>Independence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional information</td>
<td>None</td>
<td>Complete specification of dependence between operations</td>
<td>A partial list of independence</td>
</tr>
<tr>
<td>required in the program</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type of ILP</td>
<td>Superscalar</td>
<td>Dataflow</td>
<td>VLIW</td>
</tr>
</tbody>
</table>
### Analysis of dependences by Hardware

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Compiler</th>
<th>Compiler</th>
</tr>
</thead>
</table>

### Analysis of independences operations by Hardware

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Hardware</th>
<th>Compiler</th>
</tr>
</thead>
</table>

### Operation scheduling by Hardware

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Hardware</th>
<th>Compiler</th>
</tr>
</thead>
</table>

### Role of compiler

<table>
<thead>
<tr>
<th>Rearranges the code</th>
<th>Replaces some analysis hardware</th>
<th>Replaces all analysis and scheduling hardware</th>
</tr>
</thead>
</table>

### 48. What is meant by little-endian & big-endian?

**Little-endian:** Where the LSB has the lower address. In other words, the bytes are addressed 3,2,1,0, from the MSB down.

**Big-endian:** Where the LSB has the upper address. In other words, the are addressed 0,1,2,3, from the MSB down.

### 49. What is the primary function of MMU?

- Provide for the paging mechanism involved in the virtual memory organization. This is done by the paging unit.
- Provide for the segmentation mechanism by the segmentation unit.
- Provide for memory protection. This is usually done within the paging or segmentation unit, or both.
- Inclusion and management of a fast-access translation look side buffer (TLB), or address translation cache (ATC), for virtual to physical page number translation.

### 50. Define register file in processor?

A register file is a small, fast access memory. It is really a part of the CPU. It can be accessed within a fraction of a CPU cycle. It has been denoted here as level 0 in the memory hierarchy.

### 51. What is meant by replacement algorithm?

Method according to which it is decided which line will be replaced is called the replacement algorithm.
There are several replacement algorithms
- Random
- First-in first-out (FIFO)
- Least recently used (LRU)

52. Make a note on virtual memory?
Virtual memory is a hierarchical storage system of at least two levels, managed by an operating system (OS) to appear to a user as a single, large, directly addressable main memory.

53. Mention the parameter of virtual memory?

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page size</td>
<td>0.5 Kbytes - 4 Mbytes</td>
</tr>
<tr>
<td>Main memory size (Mbytes)</td>
<td>4 - 4096</td>
</tr>
<tr>
<td>Hit time (cycles)</td>
<td>1 - 20</td>
</tr>
<tr>
<td>Miss penalty (cycles*10^5)</td>
<td>1 - 6</td>
</tr>
<tr>
<td>Miss rate (%)</td>
<td>0.000001 - 0.0001</td>
</tr>
</tbody>
</table>

54. Make a note on page table mechanism?
Each page in memory has a page table entry (PTE) corresponding to it.
The PTE contains:
- The upper bits of the page base address
- Status and protection bits related to the page.

55. Make a note on segmentation?
The concept of a segment can be defined as a set of logically related contiguous words generated by a compiler or a programmer. Thus the segmentation is a memory management mechanism that allocates main memory by segments and supervises any segment-related activities.
PART B

BIG QUESTIONS

1. Draw and discuss the general block diagram of microprocessor.
   - Block diagram
   - Registers Available
   - Functions of Accumulator
   - Explanation about all blocks in the block diagram

2. Explain the different types of addressing modes supported by a microprocessor.
   - The address of the operand in memory is stored in one of registers that is addressing modes.
   - Register addressing modes
   - Immediate addressing modes
   - Direct addressing modes
   - Register indirect addressing modes
   - Register indirect with post increment
   - Register indirect with pre decrement
   - Register indirect with displacement
   - Explanation about each addressing modes with an example

3. Explain the different types of instruction set supported by a microprocessor.
   - The instruction set is indeed one of the key features of computer architecture, defining and describing the capabilities of any computing system, including microprocessors.
   - Types as following,
     - Data movement instruction
     - Integer arithmetic and logic instructions
     - Shift and rotate instructions
     - Control transfer instructions
     - Bit manipulation instructions
     - System control instruction
     - Floating-point instruction
     - Special function unit instruction
   - Explanation about each instruction with an example
4. Explain the mechanism of paging with the help of a diagram. Also give an example
   - Explanation about paging operation
   - Explanation about page table mechanism
     - Page table entry
     - Page directory entry
   - Page design consideration

5. Explain the pipelining hazards in detail
   - Definition Pipeline hazards
   - Three types of pipeline hazards
     - Structural hazards
     - Data hazards
     - Control hazards
   - Explanation about each hazards

6. Explain the concept of instruction level parallelism in detail
   - Instruction level parallelism (ILP)
   - Types of ILP
     - Superscalar
     - Super pipelined
     - Very long instruction word
   - Explanation about each type
   - Comparisons of ILP Architectures
7. Explain the following terms
   a. Pipeline
   b. Paging unit
   c. Segmentation unit
   d. Floating point unit

8. Explain the following terms
   a. Associatively
   b. Average access time
   c. Second level cache
   d. Line size
   f. Cache size

9. Give all the difference between the RISC and CISC processors.

CISC vs. RISC. CISC emphasizes hardware complexity. RISC emphasizes compiler complexity.
**CISC**

- Instruction Set: large set of instruction with variable size (16 to 64)
- Addressing Modes: 12-24
- General Purpose registers: 8-24
- Clock rate: 33-50MHz in 1992

**RISC**

- Instruction Set: Small set of instruction with fixed size (32-bit)
- Addressing Modes: 3-5
- General Purpose registers: 32-192
- Clock rate: 50-150MHz in 1993

10. (a) **List out the properties of RISC & CISC Processors**

- The RISC microprocessor is designed using hardwired control with little or no microcode. Note that variable-length instruction formats generally require microcode design. All RISC instructions have fixed formats, so microcode design is not necessary.

- A RISC microprocessor executes most instructions in a single cycle.

- The instruction set of a RISC microprocessor typically includes only register, load, and store instructions. All instructions involving arithmetic operations use registers, and load and store operations are utilized to access memory.

- The instructions have a simple fixed format with few addressing modes.

- A RISC microprocessor has several general-purpose registers.

- CISC microprocessors, on the other hand, contain a large number of instructions and many addressing modes, while RISC microprocessors include a simple instruction set with a few addressing modes. Almost all computations can be obtained from a few simple operations.

- RISC basically supports a small set of commonly used instructions that are executed at a fast clock rate compared to CISC, which contains a large instruction set (some of which are rarely used) executed at a slower clock rate. To implement the fetch/execute cycle for supporting a large instruction set for CISC, the clock is typically slower.

- However, RISC can significantly improve its performance with a faster clock, more efficient pipelining, and compiler optimization.
Advanced Microprocessor

- In CISC, most instructions can access memory while RISC contains mostly load store instructions.

- The complex instruction set of CISC requires a complex control unit, thus requiring micro programmed implementation. RISC utilizes hardwired control which is faster.

- CISC is more difficult to pipeline; RISC provides more efficient pipelining.

- An advantage of CISC over RISC is that complex programs require fewer instructions in CISC with fewer fetch cycles, while RISC requires a large number of instructions to accomplish the same task with several fetch cycles.

10. (b) Make a note on RISC Evaluation

- Advantages of RISC
  - VLSI Realization
  - Computing speed
  - Design cost and reliability
  - HLL Support

- Explanation about each advantages
## UNIT II
THE 80386 AND 80486 MICROPROCESSOR

### PART A (TWO MARKS)

1. List out the difference between the 8086, 80286, 80386 & 80486 Processor.

<table>
<thead>
<tr>
<th>Features</th>
<th>8086</th>
<th>80286</th>
<th>80386</th>
<th>80486</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address lines</td>
<td>20 bits</td>
<td>24 bits</td>
<td>32 bits</td>
<td>32 bits</td>
</tr>
<tr>
<td>Data lines</td>
<td>16 bits</td>
<td>16 bits</td>
<td>32 bits</td>
<td>32 bits</td>
</tr>
<tr>
<td>No of pins</td>
<td>20</td>
<td>68</td>
<td>132</td>
<td>168</td>
</tr>
<tr>
<td>Memory capacity</td>
<td>1 MB</td>
<td>Physical memory16 MB</td>
<td>Physical memory16 MB</td>
<td>Physical memory16 MB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cache memory1GB</td>
<td>Cache memory1GB</td>
<td>Cache memory1GB</td>
</tr>
<tr>
<td>Clock frequency</td>
<td>4,6,8MHz</td>
<td>4,6,8MHz</td>
<td>4,6,8MHz</td>
<td>4,6,8MHz</td>
</tr>
<tr>
<td>Modes of Operation</td>
<td>Real</td>
<td>Real, Protected, virtual</td>
<td>Real, Protected, virtual</td>
<td>Real, Protected, virtual</td>
</tr>
</tbody>
</table>

2. List out the features of 80286 Processor

- Physical memory-16 MB
- Cache memory-1GB
- Memory management section supports
  - Virtual memory
  - Paging
  - 4 levels of protection
3. What are the base architecture register (or) Application register set in x86?

The architecture registers are classified into following types

- General purpose registers
- Instruction pointer
- Flag register
- Segment registers

4. What are the system register in x86?

The system registers are classified into following types

- Memory management registers
- Control registers
- Code segment
- Data segment
- Stack segment

5. List out the floating point registers in x86.

The floating point registers are classified into following types

- Data registers
- Tag word
- Status word
- Control word
- Instruction and data pointers

6. Define Debug registers

The base architecture and floating-point registers are accessible by applications programs. The system and debug registers are accessible only by system programs (such as OS), running on the highest privilege level.
7. **Enlist data formats of 80286 Processor**

   The different types of data formats are
   - Signed integer data
   - Binary coded decimal data
   - Bit, byte, word, strings

8. **List out the features of 80386 Processor**

   **Features**
   - 32-bit microprocessor
   - Physical memory space is 4Gbytes\(2^{32}\)
   - Virtual memory space is 64 terabytes.
   - The memory management section of 80386 supports
     - Virtual memory
     - Paging
     - Four levels of protection
   - Operates in two mode
     - Real address mode
     - Protected mode
   - The concept of paging is introduced in 80386

9. **List out the Memory address of 8086, 8286, 80386 Processor**

   - 8086/8088=1MB
   - 80286=16MB
   - 80386=4GB

10. **Mention the two versions of 80386 Processor**

    Two versions are
    - 80386 DX(in 1985)
    - 80386 SX (in 1988)
11. Comparison between 80386 DX & 80386 SX

<table>
<thead>
<tr>
<th></th>
<th>80386 DX</th>
<th>80386 SX</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-Bit address bus</td>
<td>32-Bit address bus and 32-bit data bus</td>
<td>24 Bit address bus and 16 bit data bus</td>
</tr>
<tr>
<td></td>
<td>Packed in 132 pin ceramic pin grid array(PGA)</td>
<td>100 Pin flat package</td>
</tr>
<tr>
<td>Address</td>
<td>Address 4GB of memory</td>
<td>16 MB of memory</td>
</tr>
</tbody>
</table>

12. List out the functional units of 80386 Processor

The functional units are:
- Bus interface unit
- Code pre-fetch unit
- Decode unit
- Execution unit
- Segmentation unit

13. Define Addressing modes & mention the types of 80386

Address of source and destination operands in an instruction. The manner in which a microprocessor determines the effective.

The different types of the 80386 processor are:
- Register operand mode
- Immediate operand mode
- Direct mode
- Register mode
- Based mode
- Indexed mode
- Based indexed mode
- Based indexed mode with displacement
- Scaled indexed
- Based Scaled indexed
- Based Scaled indexed mode with displacement
14. List out the instruction set of 80386 Processor

The instruction extends the 8086/80286 instruction set in two ways. 32-bit formats of all 16-bit instructions are included to support the 32 bit data types and 32-bit addressing modes are provided for all memory reference instructions.

The 80386 instruction set is divided into nine types:
- Data transfer
- Arithmetic
- String
- Logical
- Bit manipulation
- Program control
- High-level language
- Protection model
- Processor control

15. Define interrupt

An interrupt is usually understood to be an event when an external signal stops the execution of a program. While the program is interrupted, the processor runs an interrupt service routine.

16. What are types of operating modes present in x86 Processor?

The types of operating modes are:
- Real mode
- Protected mode
- Virtual 8086 mode

17. Make a note on Real mode operation

This has the same base architecture as the 8086 but allows access to the 32-bit register set. When the processor is reset or powered up, it is initialized in real mode.

18. Make a note on Virtual mode operation

This allows the execution of 8086 applications, while still allowing the system designer to take full advantage of the protection mechanism.

19. Enlist data types of 80386 Processor

The different data types are:
- Bit
- Bit string
- Bit field
20. **Enlist the additional instruction set of 80386 Processor.**

   The newly added instructions may be categorized in to the following groups:
   - Bit scan instructions
   - Bit test instructions
   - Conditional instructions
   - Shift double instructions
   - Control transfer instructions

21. **PIPELINE HAZARDS**

    Hazards are situations in pipelining where one instruction cannot immediately follow another.

    Hazards reduce the ideal speedup gained from pipelining and are classified into three classes:
    - **Structural hazards:** Arise from hardware resource conflicts when the available hardware cannot support all possible combinations of instructions.
    - **Data hazards:** Arise when an instruction depends on the results of a previous instruction in a way that is exposed by the overlapping of instructions in the pipeline.
    - **Control hazards:** Arise from the pipelining of conditional branches and other instructions that change the PC.

    **Can always resolve hazards by waiting**
22. List out the different signals provided by 80486 Processor

**Signals**
- Address bus (A31-A2)
- Data bus (D0-D31)
- Lock
- Read /write
- Reset
- NMI
- HOLD
- HLDA
- BOFF
- D/C

23. List out the addressing modes present in 80486 Processor

Address of source and destination operands in an instruction. The manner in which a microprocessor determines the effective.

The different types of the 80486 processor are
- Register operand mode
- Immediate operand mode
- Direct mode
- Register mode
- Based mode
- Indexed mode
- Based indexed mode
- Based indexed mode with displacement
- Scaled indexed
- Based Scaled indexed
- Based Scaled indexed mode with displacement
24. List out the difference between the 800386 & 80486 Processor

<table>
<thead>
<tr>
<th>Features</th>
<th>80386</th>
<th>80486</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year</td>
<td>1985</td>
<td>1989</td>
</tr>
<tr>
<td>Address lines</td>
<td>32 bits</td>
<td>32 bits</td>
</tr>
<tr>
<td>Data lines</td>
<td>32 bits</td>
<td>64 bits</td>
</tr>
<tr>
<td>No of pins</td>
<td>132</td>
<td>168</td>
</tr>
<tr>
<td>Memory capacity</td>
<td>Physical memory-16 MB</td>
<td>Physical memory -16MB</td>
</tr>
<tr>
<td></td>
<td>Cache memory-1GB</td>
<td>Cache memory-1GB</td>
</tr>
<tr>
<td>Clock frequency</td>
<td>4,6,8MHZ</td>
<td>4,6,8MHZ</td>
</tr>
<tr>
<td>Modes of Operation</td>
<td>Real, Protected, virtual</td>
<td>Real, Protected, virtual</td>
</tr>
</tbody>
</table>

25. Enlist data types of 80486 Processor

- Signed/Unsigned data types
- Floating point
- BCD
- String
- ASCII, Pointer data types

26. Make a note on Virtual memory

An operating system technique that allows programs or data to exceed the physical size of the main, internal, directly accessible memory of the microcomputer. Program or data segment pages are swapped from external disk storage as needed. The swapping is invisible (transparent) to the programmer. Therefore, the programmer does not need to be concerned with the actual physical size of internal memory while writing the code.
27. **What are the 5 types of descriptors tables in 80386?**

   The 5 types of descriptors that the 80386 has are as follows:

   - Code or data segment descriptors
   - System descriptors
   - Local descriptors
   - Task state descriptors
   - Gate descriptors

28. **Segment Descriptor tables**

   Segmentation scheme is a way of offering protection to different types of data and code. In 80386 Processor consist of following segment descriptor tables:

   - Global Descriptor table (GDT)
   - Local descriptor table (LDT)
   - Interrupt Descriptor table (IDT)

   Their respective significances are also similar to the corresponding descriptor table significances in 80286.

29. **Make a note on Paging**

   **PAGING OPERATION**: Paging is one of the memory management techniques used for virtual memory multitasking operating system.

   The segmentation scheme may divide the physical memory into a variable size segments but the paging divides the memory into a fixed size pages.

   The segments are supposed to be the logical segments of the program, but the pages do not have any logical relation with the program.

   The pages are just fixed size portions of the program module or data.
30. Make a note a note Virtual 8086 Mode

- In its protected mode of operation, 80386DX provides a virtual 8086 operating environment to execute the 8086 programs.

- The real mode can also be used to execute the 8086 programs along with the capabilities of 80386, like protection and a few additional instructions.

- Once the 80386 enters the protected mode from the real mode, it cannot return back to the real mode without a reset operation.

- Thus, the virtual 8086 mode of operation of 80386, offers an advantage of executing 8086 programs while in protected mode.
PART B

BIGQUESTIONS

1. Explain the architecture of Intel 80386 the help of a block diagram
   - Block diagram
   - Registers Available
   - Functions of Accumulator
   - Explanation about all blocks in the block diagram

2. Explain the operating modes of 80386 Processor
   The 80386 Processor operates in following types.
   - Real mode operation
   - Protected mode operation
   - Virtual mode of 8086
   - Explanation about each operating modes

3. what are the addressing modes present in 80386 processor, explain with examples
   The addressing modes are classified into the following types
   - Immediate operand mode
   - Direct mode
   - Register mode
   - Based mode
   - Indexed mode
   - Based indexed mode
   - Based indexed mode with displacement
   - Scaled indexed
   - Based Scaled indexed
   - Based Scaled indexed mode with displacement

   - Explanation about each addressing modes with examples
4. Draw and discuss the register set of 80386 and explain a typical function of each of the register in brief.

- Diagram of register set format
- Registers Available
- Explanation about all register in the register set format

5. Explain the architecture of Intel 80486 the help of a block diagram

- Block diagram
- Registers Available
- Functions of Accumulator
- Explanation about all blocks in the block diagram

6. List the various instructions set available in 80286 and 80386 processor

   The 80286 & 80386 instruction set is divided into nine types
   - Data transfer
   - Arithmetic
   - String
   - Logical
   - Bit manipulation
   - Program control
   - High-level language
   - Protection model
   - Processor control

   - Explanation about each instruction set with example

7. Explain the following

   (i) 80386 memory management
   - Features of 80386 processor
   - Explanation about memory management techniques

   (ii) Virtual 8086 mode
   - Explanation about virtual mode of 8086 processor
(iii) Enlist the different functional groups of signals provided by 80386 & 80486 Processor.

- Pin diagram (80386 & 80486)
- Explanation about different signals provided by 80386 processor
- Explanation about different signals provided by 80486 processor

8. What are the addressing modes present in 80486 processor, explain with examples

The addressing modes are classified into the following types

- Immediate Direct mode
- Register mode
- Based mode
- Indexed mode
- Based indexed mode
- Based indexed mode with displacement
- Scaled indexed
- Based Scaled indexed
- Based Scaled indexed mode with displacement
- operand mode

- Explanation about each addressing modes with examples
UNIT III
PENTIUM MICROPROCESSOR

PART A (TWO MARKS)

1. What are the segments registers available in Pentium processor?

There are six 16bit segment registers available in Pentium processor.

- Code segment register
- Data segment register
- Stack segment register
- Extra segment register
- FS
- GS

2. What is the purpose of operand size prefix?

- When the Pentium operates in real mode, it defaults to the original 16bit 8086 register sizes.
- It is possible to take advantage of the 32bit extended registers while running in real mode.
- A special one byte code called the operand size prefix is inserted before each instruction that uses a 16bit register. For these instructions, the Pentium will use 32-bit register length.

3. Draw the flag format in Pentium.

Flag register is 16 bit wide in Pentium. It is divided into two main groups

- Control flags
- Status flags

<table>
<thead>
<tr>
<th>15</th>
<th>14</th>
<th>13</th>
<th>12</th>
<th>11</th>
<th>10</th>
<th>9</th>
<th>8</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT</td>
<td>IOPL</td>
<td>OF</td>
<td>DF</td>
<td>IF</td>
<td>TF</td>
<td>SF</td>
<td>ZF</td>
<td>-</td>
<td>AF</td>
<td>PF</td>
<td>-</td>
<td>CF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. What is byte swapping?

Byte swapping is a common method for all Intel microprocessors.
For example a 16bit hexadecimal number 2055 are written into location 15000 and 15001, with the low order 8bits(55) going into first location and higher order 8bit(20) is written as next location. i.e the lower byte is always written first, followed by higher byte.

```
   15000:   15001:
--------:--------:
   20     55
```

5. Write assembler directives in Pentium.

- DB(define byte)
- DW(define word)
- DUP(duplicate)
- EQU(equate)

6. List the various instructions available in Pentium processor

The instructions set are classified into the following types.

- Data transfer
- Arithmetic
- Bit manipulation
- String
- Program transfer
- Processor control

7. Write data transfer instructions in Pentium processor.

Data transfer instructions are used to move data among registers, memory, and outside world.
Also some instructions directly manipulate the stack, while others may be used to alter the flags.
Example

IN Input byte or word from port
OUT Output byte or word to port
XLAT Translate byte
MOV Move to/from register/memory

8. Write any four arithmetic instructions in Pentium processor.

AAA ASCII adjust for addition
AAD ASCII adjust for division
CBW Convert byte or word
CMPXCHG8B Compare and exchange 8 bytes

9. Write about bit manipulation instructions.

The instructions capable of performing logical, shift, and rotate operations are called bit manipulation instructions.

Example

AND Logical AND of byte or word
NOT Logical NOT of byte or word
ROR Rotate right byte or word
TEST Test byte or word

10. What is the purpose of string instructions?

String instructions simplify programming whenever a program must interact with a user.

User commands and responses are usually saved as ASCII strings of characters, which may be processed by the proper choice of string instruction.

Example

CMPS Compare byte or word string
LODS Load byte or word string
REP Repeat
SCAS Scan byte or word string

11. Write about program transfer instructions.

A program transfer instruction contains all jumps, loops, and subroutine and interrupt operations.

The great majority of jumps are conditional, testing the processor flags before execution

Example

CALL Call procedure
12. Write processor control instructions in Pentium.

- INT: Interrupt
- IRET: Return from interrupt
- INTO: interrupt if overflow

13. How physical address is generated?

   - Physical address is 20-bit. It is formed by the sum of two 16-bit address values.
   - One of the four segments registers will always supply the first 16-bit address. The second 16-bit address is formed by specific addressing mode operation.
   - The resulting 20-bit address points to one specific location in the Pentium’s 1MB real mode addressing space.

14. Write addressing modes in Pentium.

   Addressing modes in Pentium is basically divided into following types
   - Immediate addressing mode
   - Direct addressing mode
   - Register indirect addressing mode
   - Index addressing mode
   - Based addressing mode
   - Based-indexed addressing mode
   - Based-indexed with displacement addressing mode

15. How 32 bit effective address is generated in Pentium?

   - In protected mode, addresses are 32 bits wide, spanning a 4-gigabyte range.
   - The segment register contents are added with base register contents. Then index register is multiplied with scale factor, which is again added with the result obtained from the previous step. The scale factor may be 1, 2, 4 or 8.
   - Final result is again added with some 8-bit or 16-bit displacement value, which is the effective address.
   - Example: MOV EAX, [EBX], [ECX*4+6].
16. **What is the interrupt?**
   - An interrupt is an event that occurs while the processor is executing an instruction. The instruction might be part of a group of instructions in a main program, such as a word processing application.
   - The interrupt temporarily suspends execution of the main program in favor of a special routine that services the interrupt.
   - When interrupt processing is complete, the processor is returned back to the exact place in the main program where it left off.

17. **What are the types of interrupts in Pentium?**
   - The Pentium microprocessor is capable of responding to 256 different types of interrupts. These interrupts are generated in a number of different ways.
   - External hardware interrupts are caused by activating the processors NMI and INTR signals.
   - Internal interrupts are caused by execution of an INT instruction.
   - Divide and error is generated internally by processor itself.

18. **What is interrupt vector table?**
   - All interrupts use a dedicated table in memory for storage of their interrupt service routine (ISR) addresses. The table is called an interrupt pointer table or interrupt vector table.
   - This is 1024 bytes long, enough storage space for 256 4-byte entries.

19. **Give brief introduction about DOS (disk operating system) interrupt 21H.**
   - One of the most useful DOS interrupts is number 21H. This interrupt was chosen as the entry point into DOS for the programmers writing their own DOS application.
   - Although there are many other interrupts assigned to specific functions by DOS, INT 21H is loaded with so many different functions we need to use others.

20. **What is the effective address generated by MOV [BP+SI], AH? Assume the SS (stack segment) register contains 3000H, register BP contains 5000H, and register SI contains 800H.**
   - Step1: shifting the SS register contents 4 bits to the left.
   - Step2: adding contents of BP and SI with result obtained from previous step.
     
     SS: 30000
     + BP: 5000
     + SI: 0800
     
     Effective address: 35800
   - This is one of the example of based indexed addressing.
21. Write the priority of interrupts.

```
<table>
<thead>
<tr>
<th>Interrupt</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Divide and error</td>
<td>highest</td>
</tr>
<tr>
<td>INT,INT0</td>
<td></td>
</tr>
<tr>
<td>NMI</td>
<td></td>
</tr>
<tr>
<td>INTR</td>
<td></td>
</tr>
<tr>
<td>Single step</td>
<td>lowest</td>
</tr>
</tbody>
</table>
```

22. Write about single step interrupt.

This interrupt relies on the setting of the trace flag in the flag register. When the trace flag is set, the Pentium will generate a type-1 interrupt after each instruction executes.

23. Give brief about breakpoint interrupt.

- This is the type-3 interrupt, but is coded as a single byte for reasons of efficiency.
- Breakpoint aids in debugging in the following way. A program being debugged will have the first byte of one of its instructions replaced by the code for breakpoint (CC) when the processor gets to this instruction it will generate a type-3 interrupt.

24. Write software driver program for solving quadratic equation \( Y=5X^2-2X+6 \)?

```
QUAD PROC NEAR
  MOV BL,AL
  MUL BL
  MOV CX,5
  MUL CX
  XCHG DX,AX
  MOV AL,2
  MUL BL
  SUB DX,AX
  XCHG DX,AX
  ADD AX,6
  RET
QUAD ENDP
```
25. Differentiate 80386 and Pentium microprocessor.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>80386</th>
<th>Pentium</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Specification</td>
<td>32 bit processor</td>
<td>32 bit processor</td>
</tr>
<tr>
<td>2. Number of pins</td>
<td>132</td>
<td>296</td>
</tr>
<tr>
<td>3. Memory capacity</td>
<td>Physical memory-4GB</td>
<td>Total of 4GB</td>
</tr>
<tr>
<td></td>
<td>Virtual memory 64TB</td>
<td></td>
</tr>
<tr>
<td>4. Number of transistors</td>
<td>275,000</td>
<td>3.1 million</td>
</tr>
<tr>
<td>5. Clock frequency</td>
<td>12.5, 16, 20, 25, 33MHz</td>
<td>60-233MHz</td>
</tr>
</tbody>
</table>
PART B

BIG QUESTIONS

1. Explain the architecture of Intel Pentium processor the help of a block diagram
   - Block diagram
   - Registers Available
   - Functions of Accumulator
   - Explanation about all blocks in the block diagram

2. List the various instructions available in Pentium processor
   - The instructions set are classified into the following types.
     - Data transfer
     - Arithmetic
     - Bit manipulation
     - String
     - Program transfer
     - Processor control
     - Explanation about each instruction with examples

3. Explain the interrupt processing of Pentium processor
   - *Explanation becomes*
   - Types of interrupt
     - H/W & S/W interrupts
     - Interrupt vector table
     - Interrupt service routine
     - Vector address table
     - Multiple interrupts and interrupt priorities
4. What are the addressing modes present in Pentium processor, Explain with example

The addressing modes are classified into the following types

- Immediate addressing mode
- Direct addressing mode
- Register indirect addressing mode
- Index addressing mode
- Based addressing mode
- Based-indexed addressing mode
- Based-indexed with displacement addressing mode

- Explanation about each addressing modes with examples

5. (i) Explain the similarities differences between 80386, 80486 & Pentium

- Compare the features of 80386, 80486 & Pentium processor
- Enlist the different similarities of each processor

(ii) Explain the data transfer & logical instruction with examples

- Data transfer instructions are used to move data among registers, memory, and outside world.
- Also some instructions directly manipulate the stack, while others may be used to alter the flags.

**Example**
- `IN` Input byte or word from port
- `OUT` Output byte or word to port
- `XLAT` Translate byte
- `MOV` Move to/from register/memory

- Explanation about logical instruction with example

6. Explain the following

- **Software model of the Pentium processor**
  - Block diagram
  - Explanation about each parts in the block diagram
- **What are the data types supported by Pentium processor**
  - Bits, Byte and Words
UNIT IV

PENTIUM HARDWARE

PART A(TWO MARKS)

1. Write the goals of RISC machine.
   - Reduce accesses to main memory
   - Keep instructions and addressing modes simple
   - Make good use of registers
   - Pipeline everything
   - Utilize the compiler extensively

2. What are the operations of Pentium processor?
   - Data transfer(both single cycle and burst transfer)
   - Interrupt acknowledge cycles
   - Inquire cycles for examining the internal code and data cache
   - I/O operations

3. What is pipelining?
   Pipelining is a technique used to enable one instruction to complete with each clock cycle.

   On a non pipelined machine, nine clock cycles are needed for individual fetch, decode, and execute cycles for executing three instructions.

   On a pipelined machine, where fetch, decode, and execute operations are performed in a parallel. Only five cycles are needed to execute the same three instructions.

4. What are types of pipelines in Pentium processor?
   The Pentium employs two types of pipelines.

   These are the instruction pipelines (U and V pipelines) and a pipeline that performs special types of bus cycles.

5. What is the purpose of BOFF input?
   The BOFF input provides a way for others processors in a multiprocessor system to instantly take over the Pentium’s busses.

   The Pentium samples BOFF every clock cycle, and if low, puts its busses into a high impedance state, beginning with the next clock cycle.

   Execution resumes with the interrupted cycle when BOFF is returned high.
6. Write burst transfer order?

```
<table>
<thead>
<tr>
<th>1st address</th>
<th>2nd address</th>
<th>3rd address</th>
<th>4th address</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>6</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>18</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>18</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>18</td>
<td>10</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>
```

7. Justify Pentium is superscalar machines?

- Processors capable of parallel instruction execution of multiple instructions are known as superscalar machine.
- The Pentium is capable of executing two integer or two floating point instructions simultaneously with the help of Pentium’s twin U and V pipelines. Therefore, Pentium is superscalar machine.

8. What are the restrictions occurred when a pair of integer instructions attempting parallel execution?

- Both must be simple instructions
- No data dependencies may exist between them
- Neither instruction may contain both immediate data and a displacement value (example: MOV TABLE[SI],8)
- Prefixed instructions (example: MOV ES:[DI],AL) may only execute in the U pipeline.

9. Write U and V pipeline instruction stages?

- PF  Prefetch
- D1  Instruction Decode
- D2  Address Generate
- EX  Execute, Cache, and ALU Access
- WB  Writeback

10. Define dynamic branch prediction?

- The Pentium employs a technique called branch prediction that helps identify possible interruptions to the normal flow of instructions through U and V pipelines.
- By predicting which instructions might branch and change program flow, it is possible to keep a steady stream of instructions flowing into the pipelines.
11. What is the purpose of BTB(Branch target buffer)?

The Pentium accomplishes branch prediction through the use of a branch target buffer (BTB), a special cache that stores the instruction and target addresses of any branch instructions that have been encountered in the instruction stream.

Along with the addresses for each instruction, the BTB also stores two history bits that indicate the execution history of the last two branch instruction.

12. Define cache?

Cache is a special type of high-speed RAM that is used to help speed up accesses to memory and reduce traffic on the processor’s busses. It has the access time of 10ns or less.

Internal cache memory has instruction and data cache.

13. Define hit ratio?

If the required instruction or data operand is found in the cache, it is called as hit.

A hit ratio specifies the percentage of hits to total cache accesses. For example, a hit ratio of 0.9 means that nine times out of ten the cache contains the requested information.

\[
\text{hit ratio} = \frac{\text{Number of hits}}{\text{total cache accesses}}
\]

14. Write the types of cache organization?

The address and data that make up each cache entry may be organized in different hardware configurations. This is called cache organization.

Cache organization is classified into three types:
- Direct mapped
- Full associative
- Set associative

15. Define bus snooping?

The MESI (modified/exclusive/shared/invalid) protocol requires the Pentium to monitor all accesses to main memory in a multiprocessor system. This is called bus snooping.

The code and data cache tags are triple ported, with one port dedicated to bus snooping. The Pentium’s address lines are used as inputs during an inquire cycle to accomplish bus snooping.
16. What is the TLBs(Translate Lookaside Buffers)?

- The instruction and data caches contain TLBs(Translate Lookaside Buffers) that translate virtual addresses into physical addresses.
- Physical addresses are used to access the cache because the same address is used to access main memory.

17. How cache coherency achieved in Pentium?

- The Pentium’s mechanism for maintaining cache coherency in its data cache is called MESI(modified/exclusive/shared/invalid).
- MESI is a cache-consistence protocol that uses two bits stored with each line of data to keep track of the state of the cache line. the four states are modified, exclusive, shared and invalid.

18. Write the pipeline stages in floating point unit(FPU) of Pentium?

The FPU achieves its quick speed through a pipeline containing eight stages. the stages and their functions, are as follows

- PF                      Pre-fetch
- D1                     Instruction decode
- D2                     Address generation
- EX                     Memory and register read, floating –point data converted into memory format, memory Write
- X1                     Floating-point execute, stage one. Memory data converted into floating-point format, Write operand to floating-point register file, bypass 1.
- X2                     Floating-point execute stage two
- WF                     'Round floating-point result and write to floating-point register file, bypass 2
- ER                     Error reporting, update status word

19. Define null selector?

A selector that has an index value of zero and points to the GDT (global descriptor table) is called a null selector.

20. What are the different types of checks performed by Pentium when protecting access to segment memory?

Pentium performs five different checks

- Type check
- Limit check
- Addressable domain check
- Procedure entry point check
- Privileged instruction check
21. Define multitasking?
   One of the most important features of protected mode is its ability to support execution of multiple programs or tasks simultaneously. This is called multitasking.

22. Write the ways for switching from one task to another task?
   Task switching is accomplished in four different ways:
   - The current task JMPs or CALLs a TSS descriptor.
   - The current task JMPs or CALLs a task gate.
   - The current task executes an IRET when the NT flag is set.
   - An interrupt or exception selects a task gate.

23. Draw the task register format?

   ![Task Register Format]

<table>
<thead>
<tr>
<th>TR</th>
<th>Selector</th>
<th>Base address</th>
<th>Segment limit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   visible to programmer

   Invisible to programmer

24. Give brief about virtual 8086 mode?
   - Virtual-8086 mode is the last of the three main operating modes of the Pentium processor.
   - Virtual-8086 mode is entered from protected mode when the VM bit of the flag’s register is set, and executes programs written for the 8086 (and 8088) microprocessor.
   - Multiple virtual-8086 programs may execute simultaneously on virtual machines.

25. Draw the format of segment selector?

   ![Segment Selector Format]
PART-B

BIG QUESTIONS

1. Describe the CPU pin configuration of Pentium processor
   - Pin diagram
   - Explanation about all signals

2. Explain in detail about FPU of Pentium processor with necessary diagram
   - List out the coprocessor
   - FMUL instruction performance
   - Explanation about the FPU pipeline stages
   - Draw FPU Register file and explain

3. List out the maskable & non-maskable interrupts available in Pentium processor
   - Explanation about interrupt
   - List out the maskable interrupts
   - List out the non-maskable interrupts
   - Explanation about each interrupts with example

4. Explain the operation of Real and Protected mode of Pentium processor
   - Explanation about Real mode
   - Explanation about Protected mode

5. Draw and Explanation the superscalar architecture of Pentium processor
   - Block diagram
   - Parallel instruction execution of multiple instruction is called superscalar machines
   - Explain about all blocks in the block diagram
6. Explain the following

(i) Branch prediction

- Definition of Branch prediction
- Explanation about dynamic prediction operation

(ii) Segmentation, Page protection, multitasking

Segmentation

- Explanation about Selectors
- Explanation about segment descriptors
- Segment descriptor table
- System descriptor types

Page protection

- Define protection
- Protecting segmented accesses
- Explanation about page-level protection

Multitasking

- Define the multitasking
- Explanation about running multiple tasks simultaneously

(iv) Exceptions and interrupts

- Explanation about the following topics
  - Exceptions
  - The interrupt descriptor table (IDTR)
  - Reading and displaying the IDTR
  - IDT descriptors
  - Interrupt and exception descriptions
  - Protected mode interrupts and exceptions
  - Interrupt/exception classifications

(v) Virtual 8086 mode

- Explanation about virtual mode 8086
- Explanation about the concept of a virtual machine.
UNIT V
THE MOTOROLA M68000 FAMILY

PART A (TWO MARKS)

1. Make a note on M68000 Processor

The Motorola M68000 family is one of the most widely used microprocessor families. Its applications include:

- Apple computers (Macintosh series)
- Sun workstations
- Multiprocessors
- Many PCs, Computing systems

2. Comparison of Motorola M68000 family

<table>
<thead>
<tr>
<th>M68000 CPU</th>
<th>68000</th>
<th>68020</th>
<th>68030</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIPS</td>
<td>2.4</td>
<td>6.5</td>
<td>12</td>
</tr>
<tr>
<td>MFLOPS</td>
<td>-</td>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>Address range</td>
<td>16 Mbytes</td>
<td>4Gbytes</td>
<td>4Gbytes</td>
</tr>
<tr>
<td>Data bus</td>
<td>16-bit</td>
<td>32-bit</td>
<td>32-bit</td>
</tr>
</tbody>
</table>

3. What are the two parts of register file in M68000 Architecture?

The M68000 architecture register file is subdivide into two main parts:

- The user programming model
- The supervisor programming model

The M68000 architecture microprocessor operate in one of two privilege modes:

- User mode
- Supervisor mode
4. Mention the features of M68000 processor.

The features of the M68000 processor are,

The programming model consists of two parts

- Integer part: identical in all M68000 family Microprocessors
  - 16 general-purpose 32-bit registers
  - 32-bit program counter
  - 8-bit condition code register

- Floating –point part: identical to that of the MC68881 or MC68882 floating-point coprocessor
  - Eight 80-bit floating –point data registers (FP0 to FP7)
  - 16-bit floating-point control registers (FPCR)
  - 32-bit floating-point status registers (FPSR)
  - 32-bit floating-point instruction address registers (FPIAR)

5. What are the two privilege modes of M68000 Processor?

The two privilege modes of M68000 are

- User mode
- Supervisor mode

6. Define Memory-Mapped I/O?

I/O ports are mapped as memory locations, with every connected device treated as if it were a memory location with a specific address. Manipulation of I/O data occurs in “interface registers” (as opposed to memory locations); hence there are no input (read) or output (write) instructions used in memory-mapped I/O.

7. Define Microinstruction?

Most microprocessors have an internal memory called control memory. This memory is used to store a number of codes called microinstructions. These microinstructions are combined to design the instruction set of the microprocessor.

8. Multitasking?

Operating system software that permits more than one program to run on a single microprocessor. Even though each program is given a small time slice in which to execute, the user has the impression that all tasks (different programs) are executing at the same time.

9. Superscalar Microprocessor?

The Pentium is a superscalar microprocessor. Microprocessor Theory and Applications with 68000/68020 and Pentium Provided with more than one pipeline and executes More than one instruction per clock cycle.
10. Define User State
   Typical microprocessor operations processing conducted at the user Level. The user state is usually at lower privilege level than the supervisor state. In the user mode, the microprocessor can execute a subset of its instruction set, and allows protection of basic system resources by providing use of the operating system in the supervisor state. This is very useful in multiuser/multitasking systems.

11. Define Supervisor State
   When the microprocessor processing operations are conducted at a higher privilege level, it is usually in the supervisor state. An operating system typically executes in the supervisor state to protect the integrity of “basic” system operations from user influences.

12. Tracing?
   A dynamic diagnostic technique permits analysis debugging of the program’s execution. (Allows single stepping.)

13. Make a note on Virtual Memory
   An operating system technique that allows programs or data to exceed the physical size of the main, internal, directly accessible memory of the microcomputer. Program or data segment pages are swapped from external disk storage as needed. The swapping is invisible (transparent) to the programmer. Therefore, the programmer does not need to be concerned with the actual physical size of internal memory while writing the code.

14. Define Vectored Interrupts
   A device identification technique in which the highest Priority device with a pending interrupt request forces program execution to branch to an interrupt routine to handle exception processing for the device.

15. Mention the data types in M68000 processor
   The data types in M68000 processor are,
   - Bit
   - Bit field
   - BCD
   - Byte integer
   - Word integer
   - Long word integer
   - Quad word integer
   - Single-precision real
   - Double-precision real
   - Extended-precision real
16. List out the categories of addressing modes in MC68000?

- Data
- Memory
- Alterable
- Control

17. Mention different types of addressing modes in M68000 processor.
The different types of addressing modes in M68000 processor are
- Register direct modes
- Register indirect modes
- Register indirect with index modes
- Memory indirect modes
- PC indirect modes
- PC memory indirect modes
- Absolute modes

18. Define Register direct mode
Data register direct, Dn: the operand is in the specified data register D0 to D7
Address register direct, An: the operand is in the specified address register A0 to A7

19. Define indirect mode
Address register indirect, An: the specified address register An contains the address of
the operand in memory.

20. List out the control registers in supervisor programming model

- Two 32-bit supervisor stack pointers ISP&MSP
- A 16-bit SR
- A 32-bit vector base registers (VBR)
- A 32-bit cache control register (CACR)
- A 32-bit user root pointer (URP)
- A 32-bit MMU status register
- A 16-bit translation control register

21. Define FPCR
Floating Point Control Register-Contains an exception enable byte, that enables
or disables traps for each class of floating-point exceptions, and a mode control byte that
sets the user-selectable modes of rounding and rounding precision. FPCR can be read or
written to by the user. It is cleared by the reset function.
22. FPCR exception enable byte format

<table>
<thead>
<tr>
<th>15</th>
<th>14</th>
<th>13</th>
<th>12</th>
<th>11</th>
<th>10</th>
<th>9</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSUN</td>
<td>SNAN</td>
<td>OPERR</td>
<td>OVFL</td>
<td>UNFL</td>
<td>DZ</td>
<td>INE X2</td>
<td>INE X1</td>
</tr>
</tbody>
</table>

- **BSUN** - BRANCH SET ON UNORDERED
- **SCAN** - SIGNALLING NOT A NUMBER
- **OPEAR** - OPERAND ERROR
- **OVFL** - OVERFLOW
- **UNFL** - UNDERFLOW
- **DZ** - DIVIDE BY ZERO
- **INE X2** - INEXACT OPERATION
- **INE X1** - INEXACT DECIMAL INPUT

23. FPCR mode control byte format

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREC</td>
<td>RND</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **PREC** - Rounding Precision
- **RND** - Rounding Mode
24. FPSR Exception status byte format

<table>
<thead>
<tr>
<th></th>
<th>BSUN</th>
<th>SCAN</th>
<th>OPEAR</th>
<th>OVFL</th>
<th>UNFL</th>
<th>DZ</th>
<th>INE X2</th>
<th>INE X1</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSUN - BRANCH SET ON UNORDERED
SCAN - SIGNALLING NOT A NUMBER
OPEAR - OPERAND ERROR
OVFL - OVERFLOW
UNFL - UNDERFLOW
DZ - DIVIDE BY ZERO
INE X2 - INEXACT OPERATION
INE X1 - INEXACT DECIMAL INPUT

25. FPSR condition code byte format

<table>
<thead>
<tr>
<th></th>
<th>O</th>
<th>N</th>
<th>Z</th>
<th>I</th>
<th>NAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NAN - NOT A NUMBER
I - INFINITY
Z - ZERO
N - NEGATIVE
26. Mention the different types of operations included in the instruction set of M68000

The instruction set includes the following types of operation:

- Data movement
- Integer arithmetic
- Floating-point arithmetic
- Logical
- Shift and rotate
- Bit manipulation
- Bit-field manipulation
- BCD arithmetic
- Program control
- System control
- Memory management
- Cache maintenance
- Multiprocessor communication

27. Define memory management of M68000 processor

One of the almost unique features of the M68000 architecture is that starting with the MC68040 it has an on-chip dual MMU, one for code and the other for data. Each MMU contains an address translation cache (ATC), called TLB in other systems, in which recently used virtual to physical address translations are stored.

28. MMU status register format

<table>
<thead>
<tr>
<th>PHYSICAL ADDRESS</th>
<th>B</th>
<th>G</th>
<th>U1</th>
<th>U0</th>
<th>S</th>
<th>CM</th>
<th>M</th>
<th>0</th>
<th>W</th>
<th>T</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHYSICAL ADDRESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **B**: BUS ERROR
- **G**: GLOBAL
- **U1, U0**: USER PAGE ATTRIBUTES
- **S**: SUPERVISOR PROTECTION
- **CM**: CACHE MODE
- **M**: MODIFIED
- **W**: WRITE PROTECTED
- **T**: TRANSPARENT TRANSLATIONREGISTER
- **R**: RESIDENT
29. **Define exception processing**
Exception processing is defined as the set of activities performed by the processor in preparing to execute a handler routine for any condition that causes an exception.

30. **What are the steps involved in M68000 exception processing**

The exception processing steps of M68000 are:

- Copy SR internally; set the S bit in SR, changing to supervisor mode. Inhibit tracing by clearing the T1 and T0 bits in SR. For and interrupt exceptions, the interrupt priority mask in SR should updated.

- Determine the vector number of the exception, provided by internal logic. For interrupts, an interrupt acknowledge cycle to obtain the vector number should be performed.

- Save the current processor context by creating an exception stack frame on the active supervisor stack. If it is an interrupt and the M bit of SR is set, the M bit should be cleared and a second stack frame on the interrupt stack should be built.

- Initiate execution of the exception handler. Multiply the vector number by 4 to determine the exception vector offset. Add the offset to the value stored in the VBR to obtain the memory address of the exception vector. Load the PC from the exception vector table entry.
PART B

BIG QUESTIONS

1. Explain the M6800O architecture register file in detail

   The register file of M6800 Processor consist of
   - user programming model
   - Supervisor programming model
   - Diagram of user programming model
   - Diagram of Supervisor programming model
   - Explanation of all registers in the register file(both user & Supervisor)

2. Explain with examples, all the addressing modes of the MC68000

   The addressing modes are classified into the following types
   - Register direct modes
   - Register indirect modes
   - Register indirect with index modes
   - Memory indirect modes
   - PC indirect modes
   - PC memory indirect modes
   - Absolute modes
   - Explanation about each addressing modes with examples

3. What are the different types of operations included in the instruction set of the M68000 family, Explain each one of them in detail

   The instruction set includes the following types of operation:
   - Data movement
   - Integer arithmetic
   - Floating-point arithmetic
   - Logical
   - Shift and rotate
   - Bit manipulation
   - Bit-field manipulation
   - BCD arithmetic
   - Program control
   - System control
   - Memory management
   - Cache maintenance
   - Multiprocessor communication

   * Explanation about each instruction set with example
4. Explain with the help of a diagram, the instruction and the data caches

- Explanation about instruction cache and data cache
- Block diagram of instruction cache and data cache
- Describe the operations

5. Explain the memory management features of the MC68000 architecture in detail.

- Explain the concept of memory management
  - Address translation cache (ATC)
  - Translation lookaside Buffer (TLB)
- Explanation about Translation table tree
- Explanation about Translation tree layout in memory
- Explanation about ATC organization
- Format of MMU status register

6. Explain the following

(i) Exception processing

Exception processing is defined as the set of activities performed by the processor in preparing to execute a handler routine for any condition that causes an exception

- List out the Exception processing steps

(ii) Assembly directives

Definition of constants is done using the define constant (DC) directive.

```
LABLE:DC.SIZE LIST
```

- The Size can be byte (B), Word(W) or Length word(L)
- Describe some examples
UNIVERSITY QUESTION BANKS

ANNA UNIVERSITY COIMBATORE

(2010, 2011&2012 YEAR QUESTIONS)
1. Write differences between instruction pipeline and arithmetic pipeline?

**Instruction pipeline**
Where different stages of instruction fetch and execution are in a pipeline.

**Arithmetic pipeline**
Where different stages of an arithmetic operation are handled along the stages of a pipeline.

2. List the various pipeline hazards.

Hazards are situations in pipelining where one instruction cannot immediately follow another.

We can define three types of pipeline hazards.

- Structural hazards
- Data hazards
- Control hazards.

3. Differentiate RISC and CISC designs.

**CISC**
- Compiler
- Greater Complexity
- Code Generation
- Processor

**RISC**
- Compiler
- Code Generation
- Processor

CISC vs. RISC. CISC emphasizes hardware complexity. RISC emphasizes compiler complexity.
4. List any two properties of RISC.

- The RISC microprocessor is designed using hardwired control with little or no microcode. Note that variable-length instruction formats generally require microcode design. All RISC instructions have fixed formats, so microcode design is not necessary.
- A RISC microprocessor executes most instructions in a single cycle.
- The instructions have a simple fixed format with few addressing modes.
- A RISC microprocessor has several general-purpose registers.

5. What is descriptor table?

Segmentation scheme is a way of offering protection to different types of data and code.

The 5 types of descriptors that the 80386 has are as follows:

- Code or data segment descriptors
- System descriptors
- Local descriptors
- Task state descriptors
- Gate descriptors

6. Define task register.

<table>
<thead>
<tr>
<th>TR</th>
<th>Selector</th>
<th>Base address</th>
<th>Segment limit</th>
</tr>
</thead>
</table>

visible to programmer

Invisible to programmer
7. What is translation look aside buffer?

- The instruction and data caches contain TLBs (Translate Lookaside Buffers) that translate virtual addresses into physical addresses.
- Physical addresses are used to access the cache because the same address is used to access main memory.

8. Write the features of 80486 microprocessor?

- 32-bit microprocessor
- Physical memory space is 4Gbytes ($2^{32}$)
- Virtual memory space is 64 terabytes.
- The memory management section of 80486 supports
  - Virtual memory
  - Paging
  - Four levels of protection
- Operates in two mode
  - Real address mode
  - Protected mode

The concept of paging is introduced.

9. Define NMI.

Occurrence of this type of interrupt cannot be ignored by microcomputer and even though interrupt capability of the microprocessor is disabled. Its effect cannot be disabled by instruction.

10. What are the special interrupts of Pentium processor?

- The Pentium microprocessor is capable of responding to 256 different types of interrupts. These interrupts are generated in a number of different ways.
- External hardware interrupts are caused by activating the processors NMI and INTR signals.
- Internal interrupts are caused by execution of an INT instruction.
- Divide and error is generated internally by processor itself.
11. How INTR is disabled?

No interrupt is generated by INTR unless the interrupt enable flag is set. This can easily be accomplished with the STI instruction.

12. What interrupt number has a vector table address range of 00280H to 00283H?

Vector interrupts number 0 to 255

13. What do you mean by branch prediction?

- The Pentium employs a technique called branch prediction that helps identify possible interruptions to the normal flow of instructions through U and V pipelines.
- By predicting which instructions might branch and change program flow, it is possible to keep a steady stream of instructions flowing into the pipelines.

14. Define BOFF.

This I/p causes the processor to terminate any bus cycle currently in progress and tri-state its busses. Execution of interrupted bus cycle is restated when BOFF goes high.

15. Differentiate instruction and data cache.

**Instruction cache:**

The internal code cache keeps copies of most frequently used instructions. The internal code cache instructions is an 8KB cache dedicated to supplying to each of the processor’s execution pipelines.

**Data cache:**

The internal data cache keeps copies of the most frequently used data requested by the two integer pipelines and the floating point unit. The internal data cache is an 8KB write-back cache, organized as two way set associative with 32-byte lines.

16. What do you mean by multitasking?

One of the most important features of protected mode is its ability to support execution of multiple programs or tasks simultaneously; this is called multitasking.
17. What do you mean by exception processing?

Exception processing is defined as the set of activities performed by the processor in preparing to execute a handler routine for any condition that causes an exception.

18. What is meant by assembly directives?

Definition of constants is done using the define constant (DC) directive.

```
LABEL:DC.SIZE LIST
```

The Size can be byte (B), Word(W) or Length word(L)

19. How the bit manipulation operations are accomplished by M68000?

The instructions capable of performing logical, shift, and rotate operations are called bit manipulation instructions

```
Example
AND Logical AND of byte or word
NOT Logical NOT of byte or word
ROR Rotate right byte or word
TEST Test byte or word
```

20. What are the data formats supported by M68000?

The data types in M68000 processor are,

- Bit
- Bit field
- BCD
- Byte integer
- Word integer
- Long word integer
- Quad word integer
- Single-precision real
PART-B

ANSWER ANY FIVE QUESTIONS

21. With the neat diagram explain the various blocks of a microprocessor?
   - Block diagram
   - Registers Available
   - Functions of Accumulator
   - Explanation about all blocks in the block diagram

22. Discuss in detail about instruction level parallelism with suitable examples.
    Definition of instruction level parallelism (ILP)
    - Types of ILP
    - Superscalar
    - Super pipelined
    - Very long instruction word
    - Explanation about each type
    - Comparisons of ILP Architectures

23. With neat pin diagram explain the functions of various pins of 80486 microprocessor?
    - Block diagram
    - Registers Available
    - Functions of Accumulator
    - Explanation about all blocks in the block diagram

24. a) Explain in detail about 80386 memory system?
    - Features of 80386 processor
    - Explanation about memory management techniques

   b) Explain about protected mode.
    - Explanation about protected mode with diagram
25. (a) Describe in brief about software model of the Pentium processor with neat diagram?

- Block diagram
- Explanation about each parts in the block diagram

25. (b) Write short notes on Pentium processor registers?

There are six 16bit segment registers available in Pentium processor.

- Code segment register
- Data segment register
- Stack segment register
- Extra segment register
- FS
- GS
- Explanation about all registers in the Pentium processor

26. Discuss in detail about paging with respect to Pentium processor?

- Paging operation
- Paging unit
- Define page protection
- Protecting segmented accesses
- Explanation about page-level protection

27. (a) Explain in detail about floating point unit of Pentium processor with necessary diagram?

- List out the coprocessor
- FMUL instruction performance
- Explanation about the FPU pipeline stages
- Draw FPU Register file and explain
27. (b) Explain virtual 8086 mode?

- Virtual-8086 mode is the last of the three main operating modes of the Pentium processor.
- Virtual-8086 mode is entered from protected mode when the VM bit of the flag’s register is set, and executes programs written for the 8086(and 8088) microprocessor.
- Multiple virtual-8086 programs may execute simultaneously on virtual machines.

Explanation about the virtual 8086 mode with diagram

28. Describe in detail about the addressing modes of M68000 architecture?

The different types of addressing modes in M68000 processor are:

- Register direct modes
- Register indirect modes
- Register indirect with index modes
- Memory indirect modes
- PC indirect modes
- PC memory indirect modes
- Absolute modes
  - Explanation about each addressing modes with an examples
1. Define instruction pipelining.

*Instruction pipeline*
Where different stages of instruction fetch and execution are in a pipeline

2. Differentiate RISC and CISC processors.

**CISC**
- Instruction Set: large set of instruction with variable size (16 to 64)
- Addressing Modes: 12-24
- General Purpose registers: 8-24
- Clock rate: 33-50MHz in 1992

**RISC**
- Instruction Set: Small set of instruction with fixed size (32-bit)
- Addressing Modes: 3-5
- General Purpose registers: 32-192
- Clock rate: 50-150MHz in 1993

2. Give two examples for implicit addressing modes in 80386 microprocessor.
   - ADD START [EDI],EBX
   - MOV EDX,MOVECX

4. What is translation look aside buffer?

   - The instruction and data caches contain TLBs(Translate Lookaside Buffers) that translate virtual addresses into physical addresses.
   - Physical addresses are used to access the cache because the same address is used to access main memory

5. What is superscalar execution?

   - Processors capable of parallel instruction execution of multiple instructions are known as superscalar machine.
The Pentium is capable of executing two integer or two floating point instructions simultaneously with the help of pentium’s twin U and V pipelines. Therefore, Pentium is a superscalar machine.

6. Write any two MMX instructions in Pentium processor.
- Multimedia extension
- Multiple math extension
- Matrix math extension
- Single instruction multiple data

7. Differentiate exceptions and interrupts.

**Exception**
Exception processing is defined as the set of activities performed by the processor in preparing to execute a handler routine for any condition that causes an exception.

**Interrupts**
A temporary break in a sequence of a program, initiated externally or internally, causing control to jump to a routine, which performs some action while the program is stopped.

8. What is branch prediction in Pentium processor?
- The Pentium employs a technique called branch prediction that helps identify possible interruptions to the normal flow of instructions through U and V pipelines.
- By predicting which instructions might branch and change program flow, it is possible to keep a steady stream of instructions flowing into the pipelines.

9. List the addressing modes of MOTOROLA M68000

The different types of addressing modes in M68000 processor are
- Register direct modes
- Register indirect modes
- Register indirect with index modes
- Memory indirect modes
- PC indirect modes
- PC memory indirect modes
- Absolute modes

10. What is exception processing?
Exception processing is defined as the set of activities performed by the processor in preparing to execute a handler routine for any condition that causes an exception.
11. (a) Explain various hazards in RISC processor. (16)

Hazards are situations in pipelining where one instruction cannot immediately follow another.

We can define three types of pipeline hazards.

- Structural hazards
- Data hazards
- Control hazards.

- Explanation about each hazards with an example

11. (b) Draw and explain the general structure of microprocessor. (16)

- Block diagram
- Registers Available
- Functions of Accumulator
- Explanation about all blocks in the block diagram

12. (a) Write down the steps to enter and leave the virtual 8086 mode. (16)

- Virtual-806 mode is the last of the three main operating modes of the Pentium processor.

- Virtual-806 mode is entered from protected mode when the VM bit of the flag’s register is set, and executes programs written for the 8068(68088) microprocessor

- Multiple virtual-8086 programs may execute simultaneously on virtual machines.

- Explanation about the virtual 8086 mode with diagram

12. (b) With the neat block diagram explain the architecture of 80386 Processor. (16)

- Block diagram
- Registers Available
- Functions of Accumulator
- Explanation about all blocks in the block diagram
13. (a) Define addressing modes. Explain the different types of addressing modes in Pentium processor With example.(16)

The addressing modes are classified into the following types

- Immediate addressing mode
- Direct addressing mode
- Register indirect addressing mode
- Index addressing mode
- Based addressing mode
- Based-indexed addressing mode
- Based-indexed with displacement addressing mode

- Explanation about each addressing modes with examples

13. (b) List the architectural features of PENTIUM processor. Explain interrupt processing mechanism available in Pentium processor.(16)

- Explanation about interrupt
- List out the maskable interrupts
- List out the non-maskable interrupts
- Explanation about each interrupts with example

14. (a) With neat block diagram explain the superscalar architecture of PENTIUM processor.(16)

- Block diagram
- Parallel instruction execution of multiple instruction is called superscalar machines
- Explain about all blocks in the block diagram

(b) Explain exception processing mechanism available in PENTIUM processor. (16)

- Explanation about the following topics
  - Exceptions
  - The interrupt descriptor table(IDTR)
  - Reading and displaying the IDTR
  - IDT descriptors
  - Interrupt and exception descriptions
15. (a) List the CPU registers of Motorola 68000 processor. Explain the requirements of each registers. (16)

The register file of M68000 Processor consist of

- user programming model
- Supervisor programming model
- Diagram of user programming model
- Diagram of Supervisor programming model
- Explanation of all registers in the register file(both user & Supervisor)

(b) What is instruction set? How it is resides in the memory? Explain the instruction set of Motorola 68000 processor. (16)

The instruction set includes the following types of operation:

- Data movement
- Integer arithmetic
- Floating-point arithmetic
- Logical
- Shift and rotate
- Bit manipulation
- Bit-field manipulation
- BCD arithmetic
- Program control
- System control
- Memory management
- Cache maintenance
- Multiprocessor communication

- Explanation about each instruction set with example
1. What are CISC Processors?

The control units of such microprocessors are naturally complex, since they have to distinguish between a larger number of opcodes, addressing modes, and formats. Type of system belongs to the category called complex instruction set computer (CISC). Although many CISC microprocessors are pipeline there exist an inherent difficulty in managing a pipe in a system with a variety of instruction sizes and different instruction execution lengths.

2. What is a page technique?

Each page in memory has a page table entry (PTE) corresponding to it. The PTE contains:
- The upper bits of the page base address
- Status and protection bits related to the page.

3. Specify any four Registers of 80486 Microprocessor

- General data and address registers
- Segment selector register
- Instruction pointer
- Flag register
4. Mention the important features of 80486 microprocessor.

**Features**
- 32-bit microprocessor
- Physical memory space is 4Gbytes ($2^{32}$)
- Virtual memory space is 64 terabytes.
- The memory management section of 80386 supports
  - Paging
  - Virtual memory
  - Four levels of protection
  - Operates in two mode
  - Real address mode
  - Protected mode

5. What is the purpose of the DP0-DP7 pins on the Pentium?

**DP0-DP7 (Data parity):** These bidirectional signals are used to indicate the even parity of each data byte on the data bus. DP0 represents the parity of the lower byte.

6. How many caches are found in the Pentium and what are their sizes?

- The address and data that make up each cache entry may be organized in different hardware configurations. This is called cache organization.

- Cache organization is classified into three types
  - Direct mapped
  - Full associative
  - Set associative

7. What is multitasking?

Operating system software that permits more than one program to run on a single microprocessor. Even though each program is given a small time slice in which to execute, the user has the impression that all tasks (different programs) are executing at the same time.

8. What are limitations of Pentium hardware?

- Data transfer (both single cycle and burst transfer)
- Interrupt acknowledge cycles
- Inquire cycles for examining the internal code and data cache
- I/O operations
9. Compare Motorola and Pentium family

<table>
<thead>
<tr>
<th>M68000 CPU</th>
<th>68000</th>
<th>68020</th>
<th>68030</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIPS</td>
<td>2.4</td>
<td>6.5</td>
<td>12</td>
</tr>
<tr>
<td>MFLOPS</td>
<td>-</td>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>Address range</td>
<td>16 Mbytes</td>
<td>4Gbytes</td>
<td>4Gbytes</td>
</tr>
<tr>
<td>Data bus</td>
<td>16-bit</td>
<td>32-bit</td>
<td>32-bit</td>
</tr>
</tbody>
</table>

10. Define data cache

The internal data cache keeps copies of the most frequently used data requested by the two integer pipelines and the floating point unit. The internal data cache is an 8KB write-back cache, organized as two way set associative with 32-byte lines.

PART-B (5x16=80 marks)

11. (a).(i) Describe what happens when a number is loaded into a segment register when the microprocessor is operated in the protected mode. (8)

- Explanation about segment registers
- List out the segment registers
- Explanation about protected mode operation

11. (a).(ii) Explain the operation of the memory-paging mechanism (8)

- Explanation about memory
- Paging operation
- Paging unit
- Page table entry

11. (b) (i) Describe the sequence of events that place data onto the stack or remove data stack from stack (8)

- Describe the stack operation with example
11.(b) (ii) Elucidate on RISC properties and its evaluation (8)

**RISC Properties**

- The RISC microprocessor is designed using hardwired control with little or no microcode. Note that variable-length instruction formats generally require microcode design. All RISC instructions have fixed formats, so microcode design is not necessary.
- A RISC microprocessor executes most instructions in a single cycle.
- The instruction set of a RISC microprocessor typically includes only register, load, and store instructions. All instructions involving arithmetic operations use registers, and load and store operations are utilized to access memory.
- The instructions have a simple fixed format with few addressing modes.
- A RISC microprocessor has several general-purpose registers.

**RISC evaluation**

- Advantages of RISC
  - VLSI Realization
  - Computing speed
  - Design cost and reliability
  - HLL Support
- Explanation about each advantages

12. (a) (i) Explain how the 80386 switches from the real mode to the protected mode and vice versa (10)

The 80386 Processor operates in following types.

- Real mode operation
- Protected mode operation
- Explanation about each operating modes
12. (a)(ii) What is the concept of DPL? Explain the function of descriptor Table (6)

- Explanation about DPL
- Explanation about the descriptor table

12. (b) Explain the instruction set with addressing modes of 80386 processor (16)

The 80386 instruction set is divided into nine types

- Data transfer
- Arithmetic
- String
- Logical
- Bit manipulation
- Program control
- High-level language
- Protection model
- Processor control

- Explanation about each instruction set with example

The addressing modes are classified into the following types

- Immediate operand mode
- Direct mode
- Register mode
- Based mode
- Indexed mode
- Based indexed mode
- Based indexed mode with displacement
- Scaled indexed
- Based Scaled indexed
- Based Scaled indexed mode with displacement

- Explanation about each addressing modes with examples

13. (a) Draw the block diagram of the internal structure of the Pentium microprocessor and explain its memory system, system timing, input/output system and its special features. (16)

- Block diagram
- Registers Available
Advanced Microprocessor

- Functions of Accumulator
- Explanation about all blocks in the block diagram

13.(b)(i) Explain the function of interrupt processing in Pentium processor(8)

Explanation becomes
- Types of interrupt
- H/W & S/W interrupts
- Interrupt vector table
- Interrupt service routine
- Vector address table
- Multiple interrupts and interrupt priorities

(ii) What are types of addressing modes of Pentium processor? Explain(8)

The addressing modes are classified into the following types
- Immediate addressing mode
- Direct addressing mode
- Register indirect addressing mode
- Index addressing mode
- Based addressing mode
- Based-indexed addressing mode
- Based-indexed with displacement addressing mode

- Explanation about each addressing modes with examples

14. (a) Explain the operation of Pentium’s superscalar architecture with neat diagram.(16)

- Block diagram
- Parallel instruction execution of multiple instruction is called superscalar machines
- Explain about all blocks in the block diagram
14. (b) (i) Describe the CPU pin description of Pentium hardware (8)

- Pin diagram
- Explanation about all signals

14. (b) (ii) With neat diagram, explain the virtual 8086 mode operation (8)

- Virtual-8086 mode is the last of the three main operating modes of the Pentium processor.
- Virtual-8086 mode is entered from protected mode when the VM bit of the flag’s register is set, and executes programs written for the 8086 (and 8088) microprocessor.
- Multiple virtual-8086 programs may execute simultaneously on virtual machines.
- Explanation about the virtual 8086 mode with diagram

15. (a) With a block diagram explain the MC680X0 architecture (16)

The register file of M68000 Processor consist of

- User programming model
- Supervisor programming model
- Diagram of user programming model
- Diagram of Supervisor programming model
- Explanation of all registers in the register file (both user & Supervisor)

15. (b) (i) Explain the instruction set and assembly directives of MC680X0 (12)

The instruction set includes the following types of operation:

- Data movement
- Integer arithmetic
- Floating-point arithmetic
- Logical
- Shift and rotate
- Bit manipulation
- Bit-field manipulation
- BCD arithmetic
- Program control
- System control
- Memory management
- Cache maintenance
Advanced Microprocessor

- Multiprocessor communication
  - Explanation about each instruction set with example

15.(b)(ii) What is memory management (4)
- Explain the concept of memory management
  - Address translation cache (ATC)
  - Translation lookaside Buffer (TLB)
- Explanation about Translation table tree
- Explanation about Translation tree layout in memory
- Explanation about ATC organization
ALL THE BEST