KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
*COMPUTER ENGINEERING DEPARTMENT*

COE 205 Computer Organization & Assembly Language

**Term 082 Lecture Breakdown**

|  |  |  |  |
| --- | --- | --- | --- |
| **Lec#** | **Date** | **Topics** | **Ref.** |
| 1 | S 28/2 | Syllabus. Introduction. |  |
| 2 | M 2/3 | Assembly and machine language, Compiler and assembler. Instruction and machine language, Instruction fields: opcode, operands, Assembly vs. Machine code, Translating languages. Advantages of High-Level languages, Why learn Assembly language, Assembly vs. High-level languages. | Chapter1 &Int. Computer Org. & Assembly Lang. Prog.(Online) |
| 3 | W 4/3 | Assembly language programming tools: Assembler, Linker, Debugger, Editor. Programmer’s view of a computer system. Instruction set architecture. I-8086 instruction set architecture. | Chapter1 &Int. Computer Org. & Assembly Lang. Prog.(Online) |
| 4 | Th. 5/5 | No Class. |  |
| 5 | S 7/3 | Basic Computer Organization, Processor. Processor, Clock, Memory, Address Space,Address, Data, and Control Bus, Memory Read and Write Cycles, Memory Devices: RAM. Static vs. Dynamic RAM. Processor-Memory Performance Gap, Memory Hierarchy: Registers, RAM, Cache, Hard disk. | Chapter1 & 2 & Int. Computer Org. & Assembly Lang. Prog.(Online) |
|  | U 8/3 (Makeup) | Magnetic Disk Storage access time. Data Representation: Binary Numbers, Hexadecimal Numbers, Base Conversions, Integer Storage Sizes. | Chapter1 &Data Representation(Online) |
| 6 | M 9/3 | Binary and, Hexadecimal Addition, Signed Integers. Two’s complement representation, Sign extension, 16’s complement. | Chapter1 &Data Representation (Online) |
| 7 | W 11/3 | Binary and hexadecimal subtraction, Ranges of signed integers, carry and overflow. Character representation: ASCII code, parity bit. | Chapter1 &Data Representation (Online) |
| 8 | S 14/3 | History of Intel Processors. **(Quiz#1)** | Chapter 2 & Int. Computer Org. and Assembly Lang. Prog. (Online) |
| 9 | M 16/3 | Intel Core MicroArchitecture, CISC vs. RISC processors. IA-32 Registers, Status Flags. | Chapter 2 & Int. Computer Org. and Assembly Lang. Prog. (Online) |
| 10 | W 18/3 | Status Flags. Fetch-Execute cycle. Pipelined Execution.  | Chapter 2 & Int. Computer Org. and Assembly Lang. Prog. (Online) |
| 11 | S 21/3 | Pipelined Execution, Superscalar Architecture, IA-32 Memory Management, Modes of Operation, Real Address Mode: Logical to Linear Address Translation. Flat Memory Model. | Chapter 2 & Int. Computer Org. andAssembly Lang. Prog. (Online) |
| 12 | M 23/3 | Protected Mode Architecture and address translation, Segment Descriptor Tables, Paging. **(Quiz#2)** | Chapter 2 & 3Assembly Language Syntax and Program Structure  (Online) |
| 13 | W 25/3 | Basic Elements of Assembly Language, Flat Memory Program Template, Assembling, Linking, and Debugging Programs. | Chapter 3 &Variable, Constant, and Array Declaration (Online) |
| 14 | S 28/3 | **Variable Declaration**: DB, BYTE, SBYTE, DW, WORD, SWORD, DD, DWORD, SDWORD, DQ, QWORD, DT, TBYTE. Defining strings. DUP Operator. | Chapter 3 &Variable, Constant, and Array Declaration (Online) |
| 15 | M 30/3 | Byte Ordering, Defining Symbolic Constants: =, EQU & TEXTEQU Directives. Offset Operator, Align Directive. | Chapter 3 &Variable, Constant, and Array Declaration (Online) |
| 16 | W 1/4 | Type, Lengthof, Sizeof, PTR operators. Label directive. Operand Types. **Basic Pentium Instructions**: MOV, MOVZX, MOVSX. | Chapter 4 &(Assembly)Basic Pentium Instructions (Online) |
| 17 | S 4/4 | **Basic Pentium Instructions**: XCHG. Direct Memory Operands, Direct-Offset Operands.ADD and SUB instructions and their effect on flags. | Chapter 4 &(Assembly)Basic Pentium Instructions (Online) |
|  | U 5/4(Makeup) | ADD and SUB instructions and their effect on flags. INC, DEC, NEG, ADC, SBB. | Chapter 4 &(Assembly)Basic Pentium Instructions (Online) |
| 18 | M 6/4 | Addressing Modes: Immediate, Register, Direct, Register Indirect. **(Quiz#3)** | Chapter 4 &(Assembly)Basic Pentium Instructions (Online) |
| 19 | W 8/4 | Addressing modes: Based, Indexed, Based- Indexed. 8086 Addressing modes. Two dimensional array processing. | Chapter 4(Assembly)&Addressing Modes(Online) |
| 20 | S 11/4 | Default segments and Segment Override.LEA instruction. JMP and Loop instruction.Example Programs: Copying a string. | Chapter 4(Assembly)&Addressing Modes(Online) |
|  | U 12/4(Makeup) | Example Programs: Summing an Array. PCrelative addressing. Link Library Overview,Calling a Library Procedure, Linking to aLibrary, Input/Output procedures usingbook’s library (Irvine32.lib). | Chapter 4 & 5(Assembly)&Addressing Modes(Online) |
|  | U 12/4 | Last Day for Dropping with W |  |
| 21 | M 13/4 | Input/Output procedures using book’slibrary (Irvine32.lib), **Introduction to****Stack.** | Chapter 5(Assembly) &Stack & Procedures(Online) |
| 22 | W 15/4 | **Stack instructions**: Push and Pop. Uses ofthe Runtime Stack: Temporary Storage ofRegisters. | Chapter 5(Assembly) &Stack & Procedures(Online) |
|  | W 15/4 | 1st Major Exam |  |
| 23 | S 18/4 | Solution of Major Exam I. |  |
| 24 | M 20/4 | No Class. |  |
| 25 | W 22/4 | No Class. |  |
|  | Apr. 25-30 | Midterm Vacation |  |
| 26 | S 2/5 | Nested Loop, Saving Return Address. Stack Instructions: pushad, pusha, popad, popa, pushfd, pushf, popfd, popf. **Defining and Using Procedures**, Call & RET Instructions. Parameter Passing, Passing Parameters in Registers. | Chapter 5(Assembly) &Stack & Procedures (Online) |
| 27 | M 4/5 | Preserving Registers, USES Operator, Parameter Passing Through Stack, Call & Return Instructions, Freeing Passed Parameters from Stack, Local Variables, Stack Frame. | Chapter 5(Assembly) &Stack & Procedures (Online) |
| 28 | W 6/5 | Program Design using Procedures, Integer Summation Program, Example of a Recursive Procedure: Factorial.  | Chapter 5(Assembly) &Stack & Procedures (Online) |
| 29 | S 9/5 | **Logical Instructions**: AND, OR, XOR**,** Test, NOT. Applications of logic instructions. **Flow Control Instructions**: CMP Instruction. Conditional Jump instructions: Signed and Unsigned. | Chapter 6(Assembly)& Flow Control Instructions(Online) |
|  | U 10/5 | Last Day for Dropping all Courses with W |  |
| 30 | M 11/5 | No Class. |  |
|  | T 12/5(Makeup) | Conditional Jump instructions: Single-flag jump instructions. Jumps Based on Equality, Jumps Based on Unsigned Comparison, Jumps Based on Signed Comparisons, BT Instruction | Chapter 6(Assembly)& Flow Control Instructions(Online) |
| 31 | W 13/5 | Conditional Loop Instructions: Loope/Loopz, Loopne/Loopnz.High-Level Decision Control Structures: IF-Then-Else. | Chapter 6(Assembly)& Flow Control Instructions(Online) |
| 32 | S 16/5 | Compound Expression with AND. Compound Expression with OR, WHILE Loops, Indirect Jump, Switch Statement. | Chapter 6(Assembly)& Flow Control Instructions(Online) |
| 33 | M 18/5 | Bubble Sort Algorithm. **Shift Instructions**: SHL, SAL, SHR, SAR. Applications of using shift instructions in performing multiplication & division.  | Chapter7(Assembly)& Bit Manipulation Instructions(Online) |
| 34 | W 20/5 | **Rotate Instructions**: ROL, ROR, RCL, RCR. SHLD instruction. | Chapter7(Assembly)& Bit Manipulation Instructions(Online) |
| 35 | S 23/5 | SHRD instructions. Shift & Rotate Applications. **(Quiz#4)** | Chapter7(Assembly)& Advanced Arithmetic(Online) |
| 36 | M 25/5 | Shift & Rotate Applications. **Multiplication Instructions**: MUL, IMUL. Pentium IMUL instructions. **Division Instructions**: DIV, IDIV. | Chapter7(Assembly)& Advanced Arithmetic(Online) |
| 37 | W 27/5 | **Division Instructions**: DIV, IDIV. Sign Extension Instructions: CBW, CWD, CDQ. Translating Arithmetic Expressions, Decimal string to number conversions. Number to decimal string conversion. **Introduction to Interrupts**. | Chapter7(Assembly)& Advanced Arithmetic(Online) |
| 38 | S 30/5 | No Class. |  |
| 39 | M 1/6 | Interrupts vs. procedures.Types of Interrupts: Hardware. Software, Exceptions. Maskable and non-maskable Interrupts, Interrupt Flag. **(Quiz#5)** | Interrupts (Online) |
|  | M 1/6 | 2nd Major Exam |  |
| 40 | W 3/6 | Interrupt Processing, IDT, Trap Flag.**CPU Design**: Register Transfer. **Data-Path Design**. Connecting Registers using Muxs. | Interrupts (Online) & CPU Design (Online)  |
| 41 | S 6/6 | **Data-Path Design**. Connecting Registers using a tri-state bus. Register Transfer Timing. | CPU Design (Online) |
|  | U 7/6 (Makeup) | **Single-Bus CPU Data path design**: Fetch Control Sequence. Synchronous vs. Asynchronous Memory Transfer, Execution control sequence for ADD with register indirect addressing mode. | CPU Design (Online) |
| 42 | M 8/6 | **Single-Bus CPU**: Execution control sequence for unconditional & Conditional JMP, INC [R1], CMP, Loop instruction. Performance Considerations.  | CPU Design (Online) |
|  | M 8/6 | Dropping all Courses with WP/WF |  |
| 43 | W 10/6 | Performance Considerations. **Two-Bus CPU**: Fetch Control Sequence, Execution Control Sequence for ADD with register indirect addressing mode, unconditional and conditional Jump. **Three-Bus CPU.** Fetch Control Sequence, Execution Control Sequence for ADD. | CPU Design (Online) |
| 44 | S 13/6 | **Three-Bus CPU**: unconditional and conditional Jump Instructions. **Control Unit Design: Hardwired Control Unit,** General Hardwired Control Unit Organization, Generation of Control Signals. Deriving Rout & Rin Signals. **CPU-Memory Interface Circuit.** | CPU Design (Online) |
| 45 | M 15/6 | **CPU-Memory Interface Circuit. Microprogrammed Control Unit Design*:*** Control Word, Control Store, Micro Program Counter, Micro Instruction Register, Microinstruction, Microroutine. General Organization of Micro-programmed Control Unit, Branching Address, Sequencer. | CPU Design (Online) |