

## **CS-601 – Micro Processor and Interfacing**

### **RATIONALE:**

The purpose of this subject is to cover the underlying concepts and techniques used in Micro Processor and Interfacing. In this subject we cover the unique issues associated with designing, testing, integrating, and implementing microcontroller/microprocessor-based embedded systems.

### **PREREQUISITE**

The students should have acquired fundamental microcontroller-associated programming skills using both the C programming language and assembly language

### **Unit-I**

**Microprocessor and Microprocessor Development Systems:** Evolution of Microprocessor, Microprocessor architecture and its operations, memory, inputs-outputs (I/Os), data transfer schemes interfacing devices, architecture advancements of microprocessors, Typical microprocessor development system.

### **Unit-II**

**8085 Microprocessor :** Architecture of 8085 microprocessor ,Instruction set and Addressing modes of 8085 microprocessor,Assembly language programs of 8085 microprocessor, Stack, Subroutines, Time-Delay loops, Modular programming, Macro .

### **Unit-III**

**8086 Microprocessor :** Architecture , Registers ,Memory Segmentation ,8086 Memory Addressing , memory Read and Write Bus Cycle of 8086, Demultiplexing of the system Bus in 8086 and 8088 microprocessors, Instruction set and Addressing modes of 8086 microprocessor,Assembly language programs of 8086 microprocessor.

### **Unit-IV**

**I/O and Memory Interfacing Using 8085/8086:** memory interfacing, Interrupts of 8085/8086 Microprocessors, 8259A Programmable Interrupt Controller, Programmable peripheral Interface, 8253 Programmable Counter/Interval Timer.

**Communication and Bus Interfacing with 8085/8086 Microprocessor :**Serial Communication Interface, DMA Controller 8257, 8279-Programmable Keyboard and Display I/O Interface, Bus Interface,8089 I/O processor

### **Unit-V**

**8051 Microcontroller:** Architecture of 8051 microcontroller, Memory organization, Timers/Counters, Interrupts, Addressing modes, 8051 Instruction set , Assembly language Programs, Applications of microcontrollers.

### **Suggested Reading:**

1. Douglas V Hall, “Microprocessors and interfacing – Programming & Hardware” TMH
2. Gaonkar, “Microprocessor Architecture, Programming & Applications with 8085”, TMH
3. Rafiquzzaman, “Microprocessors-Theory & Applications”, PHI
4. Savaliya, “8086 Programming & Advance Processor Architecture”, Wiley India
5. Ray, Bhurchandi, “Advanced Microprocessor and peripherals” TMH Pub
6. Soumitra Kumar Mandal, “Microprocessors and Microcontroller” TMH Pub

## **List of Experiments**

1. To study 8085 based microprocessor system
2. To study 8086 based microprocessor system
3. To develop and run a program for finding out the largest/smallest number from a given set of numbers.
4. To develop and run a program for arranging in ascending/descending order of a set of numbers
5. To perform multiplication/division of given numbers
6. To perform conversion of temperature from 0F to 0C and vice-versa
7. To perform computation of square root of a given number
8. To perform floating point mathematical operations (Addition, Subtraction, Multiplication and Division)
9. To obtain interfacing of RAM chip to 8085/8086 based system
10. To obtain interfacing of keyboard controller
11. To obtain interfacing of DMA controller
12. To obtain interfacing of PPI
13. To perform microprocessor based temperature control of hot water

## **CS-602 – Principles Of Programming Languages**

### **RATIONALE:-**

The purpose of this subject is to cover the underlying concepts and techniques used in Programming Languages. It provides general idea related to operating & Programming environment.

### **PREREQUISITE:-**

The students should have general idea about programming language . In addition, a familiarity with Elementary and Structured Data Types is needed for better understanding.

### **UNIT-I**

Language Evaluation Criteria, influences on Language design, Language categories, Programming Paradigms – Imperative, Object Oriented, functional Programming , Logic Programming. Programming Language Implementation – Compilation and Virtual Machines, programming environments. Issues in Language Translation: Syntax, Semantics, Stages, analysis and synthesis, Parse Tree, CFG and BNF grammar.

### **UNIT-II**

Data types: Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types, design and implementation uses related to these types. Names , Variable, concept of binding, type checking, strong typing, type compatibility, named constants, variable initialization. Sequence control with Expressions, Conditional Statements, Loops, Exception handling.

### **UNIT-III**

Subprograms and Blocks: Fundamentals of sub-programs, Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, design issues for functions overloaded operators, co routines.

### **UNIT-IV**

Abstract Data types: Abstractions and encapsulation, introductions to data abstraction, Static and Stack-Based Storage management. heap based storage management. Garbage Collection. object oriented programming in small talk, C++, Java, C#, PHP, Perl . Concurrency: Subprogram level concurrency, semaphores, monitors, message passing, Java threads, C# threads.

### **UNIT – V**

Exception handling, Exceptions, exception Propagation, Exception handler in C++ and Java. Logic Programming Language : Introduction and overview of logic programming, basic elements of prolog, application of logic programming. Functional Programming Languages: Introduction, fundamentals. Introduction to 4GL.

### **Suggested Reading:**

1. Sebesta, "Concept of programming Language", Pearson Edu.
2. Louden, "Programming Languages: Principles & Practices" , Cengage Learning
3. Tucker, " Programming Languages: Principles and paradigms ", Tata McGraw –Hill
4. Terrance W Pratt, "Programming Languages: Design and Implementation" Pearson Edu.
5. Cavlo Ghezzi & Mehdi Jazayeri " Programming Languages Concepts", Willey India
- 6 E Horowitz, "Programming Languages", 2nd Edition, Addison Wesley

## **CS-603 – Software Engineering & Project Management**

### **RATIONALE:**

The purpose of this subject is to cover the underlying concepts and techniques used in Software Engineering & Project Management. Some of these techniques can be used in software design & its implementation.

### **PREREQUISITE:-**

The students should have at least one year of experience in programming a high-level language and databases. In addition, a familiarity with software development life cycle will be useful in studying this subject..

### **Unit I: The Software Product and Software Process:**

Software Product and Process Characteristics, Software Process Models: Linear Sequential Model, Prototyping Model, RAD Model, Evolutionary Process Models like Incremental Model, Spiral Model, Component Assembly Model, RUP and Agile processes. Software Process customization and improvement, CMM, Product and Process Metrics

### **Unit II: Requirement Elicitation, Analysis, and Specification**

Functional and Non-functional requirements, Requirement Sources and Elicitation Techniques, Analysis Modeling for Function-oriented and Object-oriented software development, Use case Modeling, System and Software Requirement Specifications, Requirement Validation, Traceability

### **Unit III: Software Design**

The Software Design Process, Design Concepts and Principles, Software Modeling and UML, Architectural Design, Architectural Views and Styles, User Interface Design, Function-oriented Design, SA/SD Component Based Design, Design Metrics

### **Unit IV: Software Analysis and Testing**

Software Static and Dynamic analysis, Code inspections, Software Testing Fundamentals, Software Test Process, Testing Levels, Test Criteria, Test Case Design, Test Oracles, Test Techniques, Black-Box Testing, White-Box Unit Testing and Unit Testing Frameworks, Integration Testing, System Testing and other Specialized Testing, Test Plan, Test Metrics, Testing Tools. Introduction to Object-oriented analysis, design and comparison with structured software engg.

### **Unit V: Software Maintenance & Software Project Measurement**

Need and Types of Maintenance, Software Configuration Management (SCM), Software Change Management, Version Control, Change control and Reporting, Program Comprehension Techniques, Re-engineering, Reverse Engineering, Tool Support.

Project Management Concepts, Feasibility Analysis, Project and Process Planning, Resources Allocations, Software efforts, Schedule, and Cost estimations, Project Scheduling and Tracking, Risk Assessment and Mitigation, Software Quality Assurance (SQA). Project Plan, Project Metrics.

### **Practical and Lab work**

Lab work should include a running case study problem for which different deliverables at the end of each phase of a software development life cycle are to be developed. This will include modeling the requirements, architecture and detailed design. Subsequently the design models will be coded and tested. For modeling, tools like Rational Rose products. For coding and testing, IDE like Eclipse, NetBeans, and Visual Studio can be used.

### **Suggested Reading:**

1. Pankaj Jalote, "An Integrated Approach to Software Engineering", Narosa Pub, 2005
2. Rajib Mall, "Fundamentals of Software Engineering" Second Edition, PHI Learning
3. R. S. Pressman, "Software Engineering: A Practitioner's Approach", Sixth edition 2006, McGraw-Hill.
4. Sommerville, "Software Engineering", Pearson Education.
5. Richard H. Thayer, "Software Engineering & Project Management", Wiley India
6. Waman S. Jawadekar, "Software Engineering", TMH
7. Schwalbe, "IT Project Management", Cengage Learning.

## **CS-604 – Computer Networking**

**RATIONALE:-** The purpose of this subject is to cover the underlying concepts and techniques used in Computer Networking. This syllabus provides a comprehensive introduction to computer network, network architecture and protocols.

**PREREQUISITE:-** The students should have thorough exposure in Analog and Digital Communication and Data Communications. Knowledge of Topology and protocol will help in better understanding

### **Unit –I**

Computer Network: Definitions, goals, components, Architecture, Classifications & Types. Layered Architecture: Protocol hierarchy, Design Issues, Interfaces and Services, Connection Oriented & Connectionless Services, Service primitives, Design issues & its functionality. ISO- OSI Reference Model: Principle, Model, Descriptions of various layers and its comparison with TCP/IP. Network standardization.

Queueing Models: Little's Theorem, Queueing System: M/M/1, M/M/m, M/M/∞, M/M/m/m, M/G/1

### **Unit-II**

Data Link Layer: Need, Services Provided, Framing, Flow Control, Error control. Data Link Layer Protocol: Elementary & Sliding Window protocol: 1-bit, Go-Back-N, Selective Repeat, Hybrid ARQ. Bit oriented protocols: SDLC, HDLC, BISYNC, LAP and LAPB. Protocol verification: Finite State Machine Models & Petri net models.

### **Unit-III**

MAC Sublayer: MAC Addressing, Binary Exponential Back-off (BEB) Algorithm, Distributed Random Access Schemes/Contention Schemes: for Data Services (ALOHA and Slotted- ALOHA), for Local-Area Networks (CSMA, CSMA/CD, CSMA/CA), Collision Free Protocols: Basic Bit Map, BRAP, Binary Count Down, MLMA Limited Contention Protocols: Adaptive Tree Walk, URN Protocol, High Speed LAN: Fast Ethernet, Gigabit Ethernet, FDDI, Performance Measuring Metrics. IEEE Standards 802 series & their variant.

### **Unit-IV**

Network Layer: Need, Services Provided, Design issues, Routing algorithms: Least Cost Routing algorithm, Dijkstra's algorithm, Bellman-ford algorithm, Hierarchical Routing, Broadcast Routing, Multicast Routing, Congestion Control Algorithms: General Principles of Congestion control, Prevention Policies, Congestion Control in Virtual-Circuit Subnets, Congestion Control in Datagram subnets. IP protocol, IP Addresses, Comparative study of IPv4 & IPv6, Mobile IP.

### **Unit-V**

Transport Layer: Design Issues, UDP: Header Format, Per-Segment Checksum, Carrying Unicast/Multicast Real-Time Traffic, TCP: Connection Management, Reliability of Data Transfers, TCP Flow Control, TCP Congestion Control, TCP Header Format, TCP Timer Management. Session layer: Authentication, Authorisation, Session layer protocol (PAP, SCP, H.245). Presentation layer: Data conversion, Character code translation, Compression, Encryption and Decryption, Presentation layer protocol (LPP, Telnet, X.25 packet Assembler/Disassembler). Application Layer: WWW and HTTP, FTP, SSH, Email (SMTP, MIME, IMAP), DNS, Network Management (SNMP).

### **References:**

1. Andrew S. Tanenbaum, David J. Wetherall, "Computer Networks" Pearson Education.
2. Dimitri Bertsekas, Robert Gallager, "Data Networks", PHI Publication, Second Edition.
3. Kaveh Pahlavan, Prashant Krishnamurthy, "Networking Fundamentals", Wiley Publication.
4. Uyless Black, "Computer Networks", PHI Publication, Second Edition.
5. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, "Computer Networks: An Open Source Approach", McGraw Hill.

**List of Experiments:**

1. Study of Different Type of LAN& Network Equipments.
2. Study and Verification of standard Network topologies i.e. Star, Bus, Ring etc.
3. LAN installations and Configurations.
4. Write a program to implement various types of error correcting techniques.
5. Write a program to Implement various types of framing methods.
6. Study of Tool Command Language (TCL).
7. Study and Installation of Standard Network Simulator: N.S-2, N.S-3.OpNet,QualNet etc .
8. Study & Installation of ONE (Opportunistic Network Environment) Simulator for High Mobility Networks .
9. Configure 802.11 WLAN.
10. Implement & Simulate various types of routing algorithm.
11. Study & Simulation of MAC Protocols like Aloha, CSMA, CSMA/CD and CSMA/CA using Standard Network Simulators.
12. Study of Application layer protocols- DNS, HTTP, HTTPS, FTP and TelNet.

## **CS-605 – Advance Computer Architecture (ACA)**

### **RATIONALE:**

The purpose of this subject is to cover the underlying concepts and techniques used in Advance Computer Architecture. The Syllabus discusses principles of parallel algorithms design and different parallel programming models

### **PREREQUISITE**

The students should have general Idea of Computer Organization. In addition, a familiarity with Memory organization, Computational models is required.

### **Unit-I**

Flynn's Classification, System Attributes to Performance, Parallel computer models - Multiprocessors and multicomputers, Multivector and SIMD Computers. Data and resource dependences, Hardware and software parallelism, Program partitioning and scheduling, Grain size and latency, Control flow, data flow and Demand driven mechanisms. Static interconnection networks, Dynamic interconnection Networks: Bus Systems, Crossbar Switch, Multiport Memory, Multistage and Combining Networks

### **Unit- II**

Instruction set architecture, CISC Scalar Processors , RISC Scalar Processors, VLIW architecture, Memory Hierarchy, Inclusion, Coherence and Locality, Memory capacity planning. Interleaved memory organization- memory interleaving, pipelined memory access, Bandwidth and Fault Tolerance. Backplane Bus System :Backplane bus specification, Addressing and timing protocols, Arbitration transaction and interrupt.

### **Unit-III**

Linear pipeline processor, Nonlinear pipeline processor, Instruction pipeline design, Mechanisms for instruction pipelining, pipeline hazards, Dynamic instruction scheduling - score boarding and Tomosulo's algorithm, Branch handling techniques, Arithmetic Pipeline Design, Static arithmetic pipeline, Multifunctional arithmetic pipelines. Superscaler pipeline design, Super pipeline processor design.

### **Unit-IV**

Cache coherence, Snoopy protocols, Directory based protocols. Message routing schemes in multicomputer network, deadlock and virtual channel. Vector Processing Principles, Vector instruction types, Vector-access memory schemes. Vector supercomputer architecture, SIMD organization: distributed memory model and shared memory model. Principles of Multithreading: Multithreading Issues and Solutions, Multiple-Context Processors

### **Unit-V**

Parallel Programming Models, Shared-Variable Model, Message-Passing Model, Data-Parallel Model, Object-Oriented Model, Functional and Logic Models, Parallel Languages and Compilers, Language Features for Parallelism, Parallel Programming Environment, Software Tools and Environments.

### **Suggested Reading:**

1. Kai Hwang, "Advanced computer architecture", TMH.
2. J.P.Hayes, "computer Architecture and organization"; MGH.
3. V.Rajaraman & C.S.R.Murthy, "Parallel computer"; PHI Learning.
4. Kain, "Advance Computer Architecture: - A System Design Approach", PHI Learning
5. M.J Flynn, "Computer Architecture, Pipelined and Parallel Processor Design"; Narosa Publishing.
6. Hwang and Briggs, "Computer Architecture and Parallel Processing"; MGH.
7. David E. Callav & Jaswinder Pal Singh Marge Kaufmann Advance Computer Architecture, EIS India.
8. Sajjan G. Shiva, Taylor & Francis, "Advance Computer Architecture