Dhirubhai Ambani
Institute of Information and Communication Technology

DA-IICT Road, Gandhinagar, Gujarat, India 382007.
Tel.: +91 79 6826 1700 I Fax: +91 79 6826 1710 I Web: www.daiict.ac.in

NAAC Accredited

ENGINEERS WITH SOCIAL RESPONSIBILITY

Recipient of Centre of Excellence Award by the Government of Gujarat
Recipient of '5 Star' in GSIRF Ranking by Government of Gujarat

Master of Technology in Information and Communication Technology – M.Tech. (ICT)

M.Tech. (ICT) is a full-time two-year (four semesters) program. The program has been specially designed to meet the increasing needs of professionals who would be able to respond to the convergence between computers and communication systems. The program aims to provide exposure to students who wish to build a professional career in ICT, working at the intersection of technology, research, and development in the areas of Machine Intelligence; Data Analytics; Intelligent Systems; Cyber Security; Distributed Computing; Software Engineering; Image Processing; Computer Vision; Speech Communication; RF and Antenna Theory; Signal Processing; Wireless Systems; Next Generation communication technology and MIMO channel; Embedded Systems; VLSI Subsystem Design; FPGA, and Nano electronics.

The Program curriculum includes four specializations tracks that provide a strong foundation and advanced courses in each track. The Program tries to leverage the strength and diversity of our faculty and currently offers the following specialization tracks:

- Machine Learning
- Signal Processing and Machine Learning
- Software Systems
- VLSI and Embedded Systems

Apart from courses, students are required to pursue one full year (two semesters) of research under the guidance of a faculty advisor and submit a master's thesis in order to obtain the degree of M.Tech. (ICT) specializing in the respective track. On successful completion of the program, the students will be able to acquire essential technical and practical knowledge for solving real-world problems in the ICT domain using modern technologies and tools, and will have the ability to demonstrate excellent analytical and logical problem-solving skills which may bridge the digital divide between urban and rural worlds. Apart from receiving rigorous exposure to various areas of scholastic study and research, students will be groomed to cultivate sound professional ethics.

Program Outcomes (POs)

As stated by NBA, POs represent the knowledge, skills and attitudes the students should have acquired at the end of the program.

PO1: **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs)

PSO1: To apply the theoretical concepts of computer engineering and practical knowledge in analysis, design and development of computing systems and interdisciplinary applications.

PSO2: To work as a socially responsible professional by applying ICT principles in real-world problems.

Programme Educational Objectives (PEOs)

PEO1: To prepare students with a strong foundation of core principles in ICT, wo will be able to solve and analyze real-world problems.

PEO2: To prepare students for their contributions in research and development by pursuing higher studies in the field of engineering, science, business, or administration.

PEO3: prepare students with the necessary theoretical background and technical skills to work professionally as software engineer, system analyst, research scientist, entrepreneur, software developer, and teaching professionals.

PEO4: To prepare students who will be socially responsible citizen with ethical and leadership qualities and effective interpersonal skills.

Curriculum Structure

The structure of the curriculum is broadly classified into 4 categories. The first category, referred to as *Program Core*, is a set of compulsory courses for every student in the program. The second category, referred to as *Specialization Core* courses impart domain knowledge, foundational as well as advanced, in respective specializations, and are compulsory for students of respective specializations. The third category is *Electives* in ICT, which allows one to go beyond his/her own specialization. The fourth one is the *Research Thesis,* spread over the third and fourth semesters. A student is required to carry out research under the supervision of a faculty member at DA-IICT.

Semester	**Courses**	**Credit Structure**
Semester 1	Program Core 1	1-0-4-3
	Program Core 2	2-0-0-2
	Specialization Core 1	3-0-0-3
	Specialization Core 2	3-0-2-4
	Specialization Core 3	3-0-2-4
Semester 2	Specialization Core 4	3-0-0-3
	Specialization Core 5	3-0-2-4
	Specialization Core 6	3-0-2-4
	Elective	3-0-0/2-3/4
Semester 3	Specialization Core 7	3-0-0-3
	Specialization Core 8	3-0-2-4
	Thesis	0-0-12-6
Semester 4	Thesis(continuation)	0-0-26-13
Total Credits		30-0-52/54-**56/57**

1 lecture hour contributes 1 credit; 1 tutorial hour contributes 1 credit; 2 laboratory hours contribute 1 credit

The curriculum mandates a minimum of 56 credits, 37 earned through coursework and 19 through research credits. Out of the 37 required coursework credits, 5 credits are allocated to compulsory courses (*Program core*), 29 credits are allocated to *Specialization core* courses, 3 credits are allocated to an *elective*.

The distribution of courses for M.Tech. (ICT) degree is as under:

Subject area	No. of credits
Program Core courses	5
Specialization Core courses	29
Elective courses	3
Thesis work	19
Total credits	**56**

Detailed Curriculum

The *Program Core* courses are targeted to build a set of common lab skills required in all specializations, along with imparting proper technical communication and writing skills. The *Specialization Core* courses impart foundational knowledge specific to each specialization to begin with, followed by advanced courses in the respective specialization. Some specializations may offer a **choice for specialization core courses**, especially in Semester 2 and 3. Topics covered in such advanced courses typically will introduce students to possible areas of research in the respective specialization. The *Research Thesis* gives a firsthand experience of contributing knowledge to the existing body of knowledge in each specialization, under the supervision of a faculty member at DA-IICT.

The Program Core and Specialization Core courses are described below:

Program Core Courses

Programming Lab

This course aims to provide hands-on practice in software tools and technologies used in ICT. The broad coverage of this course is as follows:

Module 1: Familiarity in Linux; Shell Programming; Perl (basic and process control); Programming tools (Makefiles, version control, debugger, profiler).

Module 2: Problem solving and programming using Python.

Module 3: Introduction to circuit modelling and analysis through LTSPICE; System design using FPGA.

Module 4: Understanding MATLAB; Lab on Sampling, Quantization and PCM; Bit error rate based performance analysis of PSK and QAM.

Communication Skills and Technical Writing

This course is especially designed for the post graduate students of engineering to train them in communicating effectively. Not only will the students develop skills required for everyday writing, but they will also be able to present their ideas in the professional scenario. Most of the classes will employ task based activities aimed at providing the students with the basic competencies in language viz., reading, writing, listening, speaking and thinking. This course will provide the learners with a practical framework and structural orientation toward language used in technical documents. The course will teach and train students to read, decipher and comprehend complex ideas which are indispensable to technical discourse. Besides preparing the students to construct documents (such as abstracts, research papers, proposals, memorandums and notes) the course will enable them to edit and proof read their own constructions. The communication aspect of the course will cater to honing the skills of speaking, expressing and adequately conveying one's ideas across. This will include teaching the students to successfully carry out presentations, interviews, impromptu speeches and discussions/deliberations. The writing aspect will focus on document construction and information processing.

Specialization Core Courses

Machine Learning

<u>Probability and Random Variables</u>

The objective of this course is to study the concepts of probability, random variables, and their applications in information and communication technology. Topics include: review of probability theory, Definitions, Set theory, Axioms of probability, Conditional probability, Bayes' theorem, Total probability, Concept of Random variable, Discrete and Continuous random variables, Commutative Distribution Function and its properties, Probability density function and its properties, Function of a random variable, Mean, variance and moments of a random variable, Characteristic functions, Bernoulli trials, Binomial distribution, Poisson distribution, Geometric distribution, Uniform distribution, Exponential distribution, Gaussian distribution, Rayleigh distribution, Joint random variables and their characterization and their application to real world problem solving, Convergence concepts, Law of large numbers, Central limit theorem, concept of mean square estimation.

<u>Linear Algebra and Optimization</u>

This course aims to strengthen the mathematical foundations in the areas of Linear Algebra and Optimization, necessary to understand and analyze problems and algorithms in Machine Learning. Machine Learning deals with modeling, analysis and processing of large amounts of data. A popular way of modeling data is using vector spaces, which are (pre-)processed using tools from Linear algebra and Optimization. The contents of the course are: (a) Linear Algebra - Importance of Vector spaces, Basis, Linear transformations, Matrices, Matrix Rank, Similarity transformations, Matrix Diagonalization, Eigenvectors & Eigenvalues, SVD, Norms, Inner products, Least squares, Projection and (b) Optimization-Unconstrained Optimization (First & Second Order Conditions, Gradient Descent and its variants), Constrained Optimization (KKT Conditions), Convex sets and functions.

Accelerated Computing

This course aims to cover the important topics of GPU programming such as design of parallel algorithms, GPU and CPU architecture, data parallelism, CUDA programming model, GPU memory model, memory performance and optimization, parallel complexity analysis, performance modeling, and case studies focused on important parallel patterns supported by lab assignments.

Pattern Recognition and Machine Learning

Machine learning concerns with designing and developing of algorithms that allow machines, essentially computers, to evolve realistic or human like behavior based on the empirical data available. This course aims to discuss the building blocks of pattern recognition problem and provide an overview of the machine learning and advanced topics. The focus would be to discuss various algorithms for pattern recognition and discuss the tools for pattern recognition.

Brain Cognitive Science

The objective of this course is to familiarize the student about brain science and cognition, and its entanglement with artificial intelligence and computer sciences. This course will introduce you about how various mental processes (information processing) like sensation, perception, attention, memory, learning, action and decision making etc. that are accomplished by the brain. In the later part of this course, we will study the various topics and techniques in the area of artificial intelligences that are partly or fully inspired from the principles of information processing in the brain. This course will also introduce you about brain-computer interaction (BCI) and neuromorphic computing.

DA-IICT GANDHINAGAR

Advanced Image Processing

The objective of this course is to introduce some basic to advanced levels of techniques for processing image. The techniques include various spatial domain, transformed domain techniques to enhance, to restore, and to segment image for higher level processing. This also includes 3D vision related perspective, and also discusses different metrics to measure the quality of image. The course will emphasize mathematical model of degradation, Noise models, Restoration in presence of noise, Periodic noise reduction, Linear position-invariant degradation, Estimating the degradation function, Inverse filtering, Wiener filtering, Constraint least square filtering. Image quality assessment metrics with Reference based metrics: MSE, PSNR, ISNR, SSIM, FSIM. No-Reference based metrics: FADE, UIQM, UCIQE will also be discussed in this course.

Information Retrieval

Unstructured information accounts for more than 70–80% of all data in organizations and is expected to grow even more in days ahead. The lion share of this is typically text. Apart from navigating the web (e.g. Google, Bing) a search engine is also useful for several other tasks like searching within intranet of an organization, looking for information in a database of documents that might not always be in public domain (e.g. enterprise search). The objective of this course is to study different components of an information retrieval system to access information from unstructured and semi-structured text. The three major components of this course are: Indexing, Retrieval and Evaluation. *Indexing* (Representation) includes vector space and embedded space representation. Implementation of Vector-Space Model, tokenizing, stop-word removal, and stemming. *Retrieval* models include vector space model, language model, probabilistic model, boolean model and a fair understanding of query processing. Query operations like Query expansion (Relevance feedback) in both supervised and unsupervised framework will be discussed. *Evaluation* will cover performance metrics: recall, precision, and F-measure; Evaluations on benchmark text collections. Given the above foundation this course will explore several domain Specific IR problems, like, IR form Legal, Medical, Financial and Social Domain.

Computer Vision

Past several decades researchers have endeavored hard to develop computers with the capacity to "see" and comprehend the world around them, as humans would do. This course aims to convey the nature of some of the fundamental problems in computer/machine/robot vision, and to explain a variety of techniques used to overcome them. Vision is a rapidly evolving area and new and emerging approaches to vision problems. We will cover some of them along with some classical techniques. Various vision problems are considered, including: feature detection in images, edge detection, and the accumulation of edge data to form lines; recovery of 3D shape from images, the use of a stereo image pair to derive 3D information; forming image mosaics; video surveillance techniques. Tracking objects in video; motion detection in video images; recognizing and classifying objects in images.

Topics in Deep Learning

Deep learning is a new area of Machine Learning research and a growing field in the area of pattern recognition, natural language processing, speech processing, image processing and vision. This course provides a broad introduction to deep learning architectures. The objectives include: (1) Formulate machine learning problems related to different applications and solve using deep learning approaches; and (2) Read current research papers and understand the issues in current research. The coverage of this course includes the following modules:

Module 1 – Machine learning (ML): Types of learning - Supervised, unsupervised, reinforcement. Linear Regression, Logistic Regression, Algorithms in ML–k-means, SVM, PCA. Concept of Over-fitting, Regularization.

Module 2 – Introduction to Neural Networks: Motivation, Simple Neural Network (shallow) and its learning (Intuition), Back propagation, Convergence Issues.

Module 3 – Deep learning Architectures: Deep Neural Network, Auto-encoder and Deep Auto-encoders, Sparse, De-noising, Convolutional Neural Networks (CNN) and Deep CNN. Recurrent Neural network, Generative Adversarial Networks (GANs), mode collapse, Variational Encoder Enhancement GAN (VEEGAN), Variational Autoencoder (VAE), and applications.

Natural Language Processing

This course presents an introduction to the computational modelling of natural language. Topics covered include: computational morphology, language modelling, syntactic parsing, lexical and compositional semantics, and discourse analysis. We will consider selected applications such as automatic summarization, machine translation, and speech processing. We will also study machine learning algorithms that are used in natural language processing.

Signal Processing and Machine Learning

Linear Algebra, Random Variables and Random Processes.

This course consists of two modules: Linear algebra and Random varibales and processes. The first module revisits a range of topics like Matrix and Vector operations, Matrix Rank, and Linear system of equations, and introduces students to concepts from vector spaces like linear independence subspaces and dimension of vector spaces and subspaces. Important factorizations like QR factorization, Matrix Diagonalization (Eigenvalue decomposition) and SVD are covered. A brief discussion on Quadratic forms and Positive Definiteness, Solution of Linear Least Squares Problems ends this module.

The next module begins with a quick overview of Probability and Random Variables (Discrete and Continuous). Functions of One Random Variable, Expectation, Mean, Variance and Moments, Characteristic Function, Joint Random Variables and Joint Distribution Functions, Discrete and Continuous, Conditional Distributions, Covariance and Correlation Coefficients are discussed next. Random processes are introduced next, and the course covers Characterization and Classification of Random Processes: Stationarity, Ergodicity, Power Spectrum (Wiener Khinchin Theorem). Random Walks, Discrete Markov Chains,White Gaussian Noise, and Response of Linear Systems to Noise.

Advanced Digital Signal Processing

This course aims to cover a few familiar topics to EC students in a mathematically rigorous manner, and introduced a few advanced topics as well.

Module 1: Basic Functional Analysis: Metric Space, Normed Spaces, Inner Product Space, Convergence of Sequences, Completeness, Banach and Hilbert Space. L_2 norm, inner product, $L_2(R)$.

Module 2: Fourier Analysis: LTI Operator (System) and Impulse Response, Fourier transform in L1(R) and its properties, Defining Fourier Transform in L2(R), via Density Extension, Global regularity of signal and spectrum decay, Gibbs Oscillations. Module 3: Sampling and Reconstruction, Shannon's three-step sampling paradigms and its limitations, Removal of Aliasing, Generalized Sampling Theorem in Hilbert Space Framework, Compressive sensing, Concept of frames (linearly dependent vectors).

Module 4: Time-Frequency analysis and wavelet: Spectrogram vs. Scalograms, Wavelets: CWT, Admissibility condition, real vs. analytic wavelets. Module 5: Multirate DSP: Upsampling, Downsampling, Quadrature Mirror Filterbank (QMF), Paraunitary Filterbanks, Wavelets and Filterbanks. Module 6: Inverse Problems, Homomorphic deconvolution, Blind deconvolution, Singular Value Decomposition (SVD), Weiner filter, All-pole models for inverse filtering. Module 7: Feature Engineering (Dimensionality Reduction):- Binning, Log-transform, Principal Components Analysis (PCA), Discrete Cosine Transform (DCT).

Introduction to Machine Learning

The course introduces students to the Basics of Machine learning. Since they are learning foundational and rerequisites like Linear algebra and Random variables along with this course, the course begins with a rather quick review of Linear algebra, with concepts like: Vector space, linear dependence and independence of vectors, matrix inverse, pseudo inverse, matrix diagonalization, singular value decomposition, Least squares (LS) estimation. Machine learning also requires optimization methods. Hence a few topics in optimization, like Convex/non convex functions, constrained/unconstrained optimization, gradient and steepest descent will be covered next. Topics like Supervised Vs Unsupervised learning, Linear and Polynomial regressions, logistic regression, entropy, cross entropy, soft max for multiclass classification, linear discriminant analysis and quadrature discriminant analysis, bias and variance in machine learning, overfitting and regularization, cross validation, Support vector machine (Soft margin and Hard margin classifiers), Use of Kernels, K means clustering,Principal component Analysis, independent component analysis, Gaussian mixture models. Introduction to Neural networks, Multilayer perceptron (MLP) will form the major component of this course.

Topics in Deep Learning

Deep learning is a new area of Machine Learning research and a growing field in the area of pattern recognition, natural language processing, speech processing, image processing and vision. This course provides a broad introduction to deep learning architectures. The objectives include: (1) Formulate machine learning problems related to different applications and solve using deep learning approaches; and (2) Read current research papers and understand the issues in current research. The coverage of this course includes the following modules:

Module 1 – Machine learning (ML): Types of learning - Supervised, unsupervised, reinforcement. Linear Regression, Logistic Regression, Algorithms in ML–k-means, SVM, PCA. Concept of Over-fitting, Regularization.

Module 2 – Introduction to Neural Networks: Motivation, Simple Neural Network (shallow) and its learning (Intuition), Back propagation, Convergence Issues.

Module 3 – Deep learning Architectures: Deep Neural Network, Auto-encoder and Deep Auto-encoders, Sparse, De-noising, Convolutional Neural Networks (CNN) and Deep CNN. Recurrent Neural network, Generative Adversarial Networks (GANs), mode collapse, Variational Encoder Enhancement GAN (VEEGAN), Variational Autoencoder (VAE), and applications.

<u>Detection and Estimation Theory</u>

Different problems in signal processing and communication involve detection and processing of the signals to make inference. In practical scenario, the signals could be noisy. The objective of this course is to provide fundamental and theoretical concepts to develop frameworks such that the inference problem can be addressed in those areas.

(a) Foundations: Probability- conditional probability, PDFs, Continuous random variable, Functions of random variables, Characteristic Functions, Expectation and Moments, Central Limit Theorem. Random processes - Ensemble Correlation Functions, Time averages, Power Spectral Density, Gaussian Process, Sampling and Random Sequences, Poisson Process. Linear Vector Spaces, Hilbert Spaces.Constrained and unconstrained optimization.

(b) Detection theory: Hypothesis testing - The Neyman-Pearson Criterion, Bayes Criterion, Minimum Error Probability Criterion, Minimax Criterion, Sequential Hypothesis Testing. Detection in the Presence of Unknowns: Random Parameters, Non-random parameters. Detection of Signals in Gaussian Noise: White Gaussian, Colored Gaussian, Spectral detection. Detection in the Presence of Uncertainties: Unknown signal and Noise parameters. Non-Gaussian Detection Theory: Robust Hypothesis Testing, Non-Parametric Model Evaluation, Partially Known Signals and Noise, Partially Known Signal Waveform, Partially Known Noise Amplitude Distribution, Non-Gaussian Observations.

(c) Estimation theory: Terminology in Estimation Theory. Minimum variance unbiased estimation: Unbiased estimators, Minimum variance criterion, Existence and search of the minimum variance unbiased estimator, Extension to a vector parameter. Cramer-Rao Lower Bound - Signals in white Gaussian noise, parameter transformation, vector parameter, general Gaussian case, and WSS Gaussian random process. Practical Estimation of Signal Parameter: Best Linear Unbiased Estimators, Maximum Likelihood Estimators, Least Squares estimation. Parameter Estimation via Bayesian: Bayesian linear model, nuisance parameter, Bayesian Estimation for Deterministic Parameters, Risk Functions, MMSE and MAP Estimator, Sequential Linear MMSE estimators, Wiener Filtering, Kalman Filtering.

Speech Technology

The objective of this course is to understand the potential of various speech technologies, such as, speech, speaker and language recognition, voice conversion, text-to-speech (TTS) synthesis, audio search, query-by-humming (QBH). Finally, course also discusses applications of speech technology in medical-domain, such as, infant cry classification, autism spectrum disorder (ASD), cleft speech, vocal fold pathology, hearing aids, voice cloning or voice banking, etc. Topics include: Overview of Speech Technology, Speech Production, Speech Analysis, Speaker and Language Identification, Voice Conversion, Automatic Speech Recognition, Audio Search, Query-by-Humming, Text to Speech Synthesis.

Wavelet Signal Processing

The course begins with a brief Review of Fourier analysis and its limitations: Global Regularity, Time -Frequency Resolution, Heisenberg's uncertainty principle in signal processing, Windowed Fourier transform. It then introduces Wavelet transform as Time-Frequency Analysis, Instantaneous frequency, Quadratic Time-Frequency Energy, Estimation of Instantaneous Frequency using STFT and Analytic Wavelet Transform. Advanced topics like Frame Theory, Windowed Fourier Frames, Wavelet Frames, Dyadic wavelet transform are discussed next. Other topics include: Lipschitz uniform vs. pointwise regularity, vanishing moments of wavelets, Detection of Singularities, Multiscale Edge Detection, Multifractals, fractal dimension, singularity spectrum, modeling of hydrodynamic turbulence. Wavelet bases, Classes of Wavelet bases, Multisacle Interpolations. Wavelet bases and filterbank, Design of Haar Wavelet, Spline Wavelets, Daubechies Wavelets. Wavelet Packets, Block Transforms, Lapped Orthogonal Transform. Linear/ Non-Linear Approximation, Adaptive Basis Selection, Basis Pursuit, Matching Pursuit. Applications of Wavelets to Estimation Theory and Deep learning will looked at towards the end.

Adaptive Signal Processing

This course aims to discuss algorithms to process the signals in an adaptive manner where it is assumed that the underline system/channel may vary over time. Its application areas include sonar, radar, biomedical and communication signal processing. This course will discuss all the classical adaptive algorithms like LMS, NLMS, RLS etc. The emphasis is on finding theoretical analyses of the adaptive algorithms and also on the recently proposed algorithms.

Computer Vision

Past several decades researchers have endeavored hard to develop computers with the capacity to "see" and comprehend the world around them, as humans would do. This course aims to convey the nature of some of the fundamental problems in computer/machine/robot vision, and to explain a variety of techniques used to overcome them. Vision is a rapidly evolving area and new and emerging approaches to vision problems. We will cover some of them along with some classical techniques. Various vision problems are considered, including: feature detection in images, edge detection, and the accumulation of edge data to form lines; recovery of 3D shape from images, the use of a stereo image pair to derive 3D information; forming image mosaics; video surveillance techniques. Tracking objects in video; motion detection in video images; recognizing and classifying objects in images.

Accelerated Computing

Know-how of principles of parallel algorithm design and ability to program on heterogeneous computing

systems using CUDA C will be explained in the beginning. Ability to express data and instruction level parallelism in applications using CUDA, and understanding of important parallel patterns will be covered next, along with Hands-on experience with fundamental tools and techniques for accelerating/ optimizing applications (C language) on GPUs with CUDA (taking into account processor architecture features).

Software Systems

Probability and Random Variables

The objective of this course is to study the concepts of probability, random variables, and their applications in information and communication technology. Topics include: review of probability theory, Definitions, Set theory, Axioms of probability, Conditional probability, Bayes' theorem, Total probability, Concept of Random variable, Discrete and Continuous random variables, Commutative Distribution Function and its properties, Probability density function and its properties, Function of a random variable, Mean, variance and moments of a random variable, Characteristic functions, Bernoulli trials, Binomial distribution, Poisson distribution, Geometric distribution, Uniform distribution, Exponential distribution, Gaussian distribution, Rayleigh distribution, Joint random variables and their characterization and their application to real world problem solving, Convergence concepts, Law of large numbers, Central limit theorem, concept of mean square estimation.

Advanced Algorithms

This course aims to cover the fundamentals of algorithm design and to enhance the problem-solving skills necessary for developing efficient software systems in various applications. The coverage of the course starts with the algorithm design technique including divide-and-conquer, greedy approaches, dynamic programming, heuristic algorithms, and approximation algorithms. Substantial emphasis will be given for discussion on relevant data structures while discussing these topics and the complexity analysis of algorithm design.

Advanced Software Engineering

The main objective of this course is to understand and learn how complexity and change are engineered during large software development. The course will focus on the methodologies (processes), techniques (methods), and tools that can be used to successfully design and validate large software systems. The course will focus on the state-of-the-art in applying quantitative assessment methods in Software Engineering and other related fields. The contents of this course to be covered are: (1) Software Requirements Modeling and Specifications, (2) Software Architecture and Design Patterns, Software Development Methodologies, (3) Software Measurement and Metrics, (4) Empirical Software Engineering, (5) Computer Aided Software Engineering and Tool Support (DevOps, Automation), (6) Applications of ML and AI in analyzing software products, and (7) Assessment and Evaluation in Software Engineering.

Network Security

In this course the student is exposed to different attacks and threats in computer networks including network mapping, port scanning, sniffing, DDoS, reflection attacks, attacks on DNS and leveraging P2P deployments for attacks. The course will also review some cryptographic primitives, such as the concepts of block ciphers, stream ciphers, pseudo-random functions, public key cryptography, digital signatures and key distribution, relevant to secure networking protocols. The course will discuss several secure networking protocols, including PGP, SSL, IPsec and wireless security protocols. The course is expected to examine operational security, including firewalls and intrusion-detection systems. Students would be expected to read recent research papers on network security and participate in an important lab component that includes packet sniffing, network mapping, firewalls, SSL and IPsec.

Computing and Complexity

The objective of the course is twofold. The first objective is to introduce the students with the exact notion of algorithms in terms of time and storage capacity. For this, different mathematical machine models are defined. The second objective of this course is to teach students how to divide problems in to various complexity classes based on the resources required to solve them. Topics include Formal definition of Turing machine, variants of Turing machine, decidable languages, semi decidable languages, Universal Turing machine, undecidable languages, Cantor's diagonalization, Halting problem, Post correspondence problem, Turing computable function, Rice's theorem, decidable logical theory, undecidable logical theory, Godel's incompleteness theorem, the class P, the class NP, NP-completeness, polynomial reducibility, Cook-Levin theorem, Savitch's theorem, the class PSPACE, PSPACE-completeness, the class L and NL, EXPSPACE, EXPSPACE completeness, Hierarchy theorem, relativization, introduction to circuit complexity, and introduction to communication complexity.

Distributed Databases

Collection of correlated databases distributed over a cluster of servers can facilitate scalable query processing. This course includes foundational work, recent developments and trends in scalable database management systems. Modern Data

Storage and Query processing for Parallel and distributed databases will be discussed. Students will be working on projects in the domain of modern distributed data storage and query processing during labs. Students will be using various SQL and NoSQL database management tools for implementing their projects. Topics include: Distributed architecture, distributed database design considerations and strategies, fragmentation and allocation, distribution and replication models, scalable query processing and optimization, query cost estimation, distributed transaction management, concurrency control, database interoperability, various data models, SQL and NoSQL databases, RDF and graph databases, static and streaming data, research issues in modern distributed database management.

Advanced Computer Networks

This course aims to provide an exposure to basic knowledge of networking concepts and background, such as basics of Internet protocols, link layer and framing concepts etc. However, an in-depth review of undergraduate level networks material will be done in this course. This course will emphasize the concepts and issues underlying the design and implementation of the Internet. We will also spend time learning to quantitatively analyze the performance of network protocols. Topics include: Datagram, circuit, and connection-oriented networks; Multiple access: contention and ordered techniques; TCP/IP group of protocols: routing and end-to-end reliability; Open and Closed loop flow control; CSMA/CD protocol analysis - efficiency, latency; Point-to-point routing: OSPF, BGP, convergence, Performance Analysis; Multicast protocols and Anycast protocols; Reliability Issues in Broadcast and Multicast; Performance Issues in TCP: Modeling and Analysis; Active Queue Management in TCP Networks; QoS definition, Mapping models; Label Switching, MPLS; Mobility support in Internet; Networking virtualization.

Distributed Systems

Distributed systems are systems comprising of computing elements that are spread across geographical locations. Examples include Internet, Google file systems etc. Contents of the course include (a) architectures of various distributed systems, (b) refresher on processes, clients and servers, (c) fundamentals of network communication, remote procedure calls, message oriented versus stream oriented communication, multicast communication, (d) domain name service and its architecture, (e) synchronization using Lamport logical clocks, mutual exclusion algorithms, (f) replication and consistency models, and (g) fault tolerance.

Blockchain and Crypto-Currency

This course explores the fundamentals of blockchains and cryptocurrencies. Topics covered include basic cryptographic tools, early digital cash, Bitcoin blockchain, Script language, Bitcoin wallets, applications of Bitcoin scripts, distributed consensus algorithms, proof of work, mining pools, mining attacks, altcoins, virtual mining, cross-chain transactions, Bitcoin exchanges, anonymity and privacy techniques, scaling blockchains, smart contracts, decentralized applications, and Ethereum blockchain.

Software Testing and Verification

Testing and verification of the software systems is not a "silver bullet" that can guarantee the production of high quality applications. While a "correct" correctness proof demonstrates that a software system (which exactly meets its specification) will always operate in a given manner, software testing that is not fully exhaustive can only suggest the presence of flaws and cannot prove their absence. Software systems are therefore always checked for correctness before they are deployed. Verification and testing are two predominant ways for checking the correctness of software systems. This course is aimed at learning various techniques of testing and verification. The coverage of this course includes the following modules:

Module 1: Software Testing: Basic concepts and preliminaries

Module 2: Model-based Testing

Module 3: Design and Code Inspections to Reduce Errors in Program Development

Module 4: Applications of AI in Software Quality Assurance

Module 5: Verification of Software Systems

Module 6: Testing of Web GUI and Mobile Apps

Module 7: AI and ML for Security Testing

VLSI and Embedded Systems

Basics of VLSI

This course is an overview to modern CMOS VLSI Design, mostly digital logic. It will start with a basic introduction of MOS transistors as switches and CMOS inverter characteristics. Following topics will be covered in this course: Introduction to VLSI Design; MOSFET Transistor Theory and characteristics; Scaling; CMOS Process Technology and Design Rules; Physical Layout of CMOS ICs, Stick Diagram; CMOS inverter characteristics, Power Dissipation; Pass Transistor Logic; Static Logic Gates; Logical effort and delay estimation of logic gates; Delay estimation in long wire: buffer insertion; Driving large capacitive load, delay minimization in an inverter cascade; Ratioed logic: Resistive load inverter, Pseudo-nMOS logic, inverter and other logic gates using Pseudo-nMOS; Transmission gate circuits; Dynamic logic circuits, Domino logic; Clocking strategies: one phase and two phase clocking. Clock Distribution; Physical Design Automation: Partitioning, Floor planning, Placement and Routing; VLSI testing.

Introduction to Embedded Systems

This course will give the students an introductory understanding of embedded systems which includes the programming the ARM based Microcontrollers, interfacing of external peripheral devices to microcontroller, and troubleshooting the embedded systems/products. Following topics will be covered in this course: Components of an Embedded System, HW/SW Building Blocks, Embedded C, Programming ARM/Cortex Microcontrollers, Keil Compiler, Interrupts and Interface with ARM/Cortex Microcontroller, UART-Serial Interface, I/O Synchronization, Interfacing Microcontroller with Keyboard, Display, Debugging Techniques, Debugging using Keil IDE; Introduction to Embedded OS: Overview of Embedded Operating System, Memory Management, Tasks and Scheduling, Embedded Linux, Embedded Android, Load OS on a

Fresh Board, Opening a Port, ssh etc.; Introduction to Various Sensors and Actuators: Embedding Sensors and Actuators in Microcontroller, Analog to Digital Conversion (ADC), Digital to Analog Conversion (DAC), Various Sensors and their Circuits, Motor and Motor Control Circuits, Pulse-Width-Modulation (PWM), Implementation of Proportional-Integral-Derivative (PID) Control in Microcontroller, Integration of Sensors and Motors with RaspberryPi/Arduino Boards; Project: Hands on project work based on real-world experience, build an embedded system with own, coding with numerous programs to make embedded

products useful.

Digital Design using HDL and FPGA

The course aims to impart digital logic concept realization using hardware description language (HDL) and its practical implementation on FPGA based hardware kits. The HDL will be discussed in detail with all its abstraction levels and modeling schemes. The focus of the course will be to develop practical realizable projects using HDL and FPGA. Topics include:

Introduction to digital design, concept of hierarchical and structured design, role of CAD tools in VLSI design process. Introduction to HDL, basics of Verilog language, synthesis, FPGA, familiarization with Xilinx Spartan-3E FPGA development board. Combinational systems realization using HDL and FPGA—arithmetic functions, multiplexer, demultiplexer, encoder and decoder, code converters. User-defined primitives, instantiation, parameterized module, realization of test benches, timing and delay models. Description and design of sequential circuits—latches, flip-flops, register, counter. Defining procedural blocks, procedural flow control, blocking and non-blocking assignments, implementation of synchronous and asynchronous designs. Data subsystems, storage modules, memory, stack, queue, functional modules, data paths, control subsystems, FSM and I/O subsystem, Mealy and Moore designs. System task and functions, operations related to I/O subsystems, compiler directives. VeSPA (Very Small Processor Architecture) processor - developing behavioral and structural modeling using HDL and FPGA. Designing of practical systems.

Digital System Architecture

This course covers the basics of uniprocessor, Multicore architecture and programming; Fundamentals of Computer Design; ARM architecture; Instruction Level Parallelism and its Dynamic Exploitation; Memory Hierarchy Design; Multiprocessors and TLP; Parallel Programs (Lab); CiLK (Lab); Programming for performance; Workload Driven Evaluation; Shared memory multiprocessor.

VLSI Subsystem Design

The *objective* of this *course* is to provide students with a sound knowledge of *VLSI subsystem design. The modules of this course include:*

Wires and Interconnect: Resistance, Capacitance, RC delay analysis, Cross-talk delay and noise effects, Repeaters, Logical Effort, Crosstalk control, etc.

Dynamic Logic, Domino logic, Keepers, etc.

Sequential Circuits: Latches, Flip-flops, RC delay analysis, Sequencing Methods: FF based, 2- phase Latch based, and Pulsed latch based. Timing analysis: Set-up time (Max-delay) constraint, Hold time (Min-delay) constraint, Clock-skew budgeting, Time borrowing, simple synchronizer, FSM, introduction to pipelined system/ALU, etc.

Datapath Subsystems:

- Adders: Full Adder using a variety of Logics styles, bit-serial Adder, Ripple Carry Adder, Carry-skip Adder, Carry Look-ahead Adder, Brent-Kung Adder, Kogge-Stone Adder, Carry-Save Adder (multi -operand addition), etc.
- Multipliers: Unsigned Array Multiplier, Booth Encoded Multiplier, Baugh-Wooley Multiplier, Wallace tree multiplier, etc.
- Division: Non-restoring method, and restoring method, etc.
- Standard Math function implementation: Cordic Algorithm, Newton-raphson mehod, etc.
- Comparators, shifter-registers, random number generator based on Linear Feedback Shift-Registers (LFSR), etc.
- Error Correcting Codes: LFSR based CRC, and Hamming7-4 codes.

Memory Array Subsystems: Register-file; Content-addressable memory; LIFO and FIFO; SRAM: decoders, column MUX, RC Analysis, etc.

Embedded System Design

This course aims to cover the concept of embedded systems design. Embedded microcontroller (ARM Architecture) cores, embedded memories. Examples of embedded systems.; Technological aspects of embedded systems: interfacing between analog and digital blocks, signal conditioning. Implementation of Low-power schemes, interfacing with external systems, user interfacing. Design trade-offs due to process compatibility, thermal considerations, temperature compensation methodologies. Software aspects of embedded systems: real time programming languages and operating systems for embedded systems.

Analog CMOS IC Design

This course will introduce advanced concepts in analog circuit design specifically relevant to CMOS IC design. The course will cover circuit noise and mismatch, their analysis, and their impact on CMOS and design. Topics include: Revision of Circuit Theory, Poles and Zeros, Dominant pole concept and stability; Passive elements in CMOS technology; MOSFET model, small signal parameter of the MOSFET; Current

Sources and Current Mirrors; Single ended single stage amps -CS,CD,CG; Differential Amps; Cascade and Cascode Amplifiers; Effect of feedback in amplifiers; Layout of CMOS analog circuits; Noise in CMOS analog circuits

Low Power VLSI Design

The aim of this course is to give a broad grounding in the principles and practice of Electronic Design Automation techniques for System on Chip Design. The course covers topics in Ultra Low Power VLSI digital circuits (Digital IC design, layout, simulation, synthesis, VLSI design techniques and system architecture; CAD tools and techniques, Low Power, ultra-low power circuit techniques and energy harvesting electronics). This course is a design intensive course that will cover moderate to advanced use of the following tools and languages:

- Magic or Cadence icfb, h/p/lt Spice

- VHDL simulator, synthesis tool

- (Circuits) MOS gate characteristics

- (HDL)VHDL or Verilog

- CMOS technology (0.18 CMOS).

Selected topics in VLSI

This course covers modules like Physical Design and Verification, and Nanoelectronics. The module on Physical Design and Verification introduces students to RTL and verification, Synthesis, Floor and Power plan, Clock and related issues, Routing and Physical verification and timing fixes. Nanoelectronics includes nanocircuit devices, memory devices and architecture.

Executive Registrar
DA-IICT, Gandhinagar

DA-IICT GANDHINAGAR