### SCHEME OF TEACHING AND EXAMINATION

**M.TECH. - VLSI DESIGN & EMBEDDED SYSTEMS**

#### II SEMESTER

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Subject Code</th>
<th>Name of the Subject</th>
<th>Teaching hours/week</th>
<th>Duration of Exam in Hours</th>
<th>Marks for</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>10LVS21</td>
<td>10EC025</td>
<td>Design of analog &amp; mixed mode VLSI Circuits</td>
<td>Lecture 4, Practical 2, Tutorial -</td>
<td>3</td>
<td>50*</td>
<td>100</td>
</tr>
<tr>
<td>10LVS22</td>
<td>10EC126</td>
<td>Real Time Operating Systems</td>
<td>Lecture 4, Practical 2, Tutorial -</td>
<td>3</td>
<td>50*</td>
<td>100</td>
</tr>
<tr>
<td>10LVS23</td>
<td>10EC116</td>
<td>Advanced Microcontrollers</td>
<td>Lecture 4, Practical - 2, Tutorial 2</td>
<td>3</td>
<td>50*</td>
<td>100</td>
</tr>
<tr>
<td>10LVS24</td>
<td>10EC047</td>
<td>Low Power VLSI Design</td>
<td>Lecture 4, Practical - 2, Tutorial 2</td>
<td>3</td>
<td>50*</td>
<td>100</td>
</tr>
<tr>
<td>10LVS25</td>
<td>10ECxxx</td>
<td>Elective-II (10LVS25x)</td>
<td>Lecture 4, Practical - 2, Tutorial 2</td>
<td>3</td>
<td>50*</td>
<td>100</td>
</tr>
<tr>
<td>10LVS26</td>
<td>10EC921</td>
<td>Mini – Project &amp; Seminar</td>
<td>Lecture - 3, Practical - Tutorial -</td>
<td>-</td>
<td>50*</td>
<td>-</td>
</tr>
</tbody>
</table>

*Practical will be evaluated 25 marks and internal assessment for 25 marks. Lab journals should be maintained.

* Assignments/seminar will be evaluated for 25 marks and internal assessment for 25 marks. Record of Assignments/seminar should be maintained.

$ Mini project should be done individually and is assessed for 25 marks. Seminar on Miniproject will be assessed for 25 marks.

#### ELECTIVE – II

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Subject Code</th>
<th>Name of the Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>10LVS251</td>
<td>10EC027</td>
<td>Design of VLSI system</td>
</tr>
<tr>
<td>10LVS252</td>
<td>10EC010</td>
<td>VLSI Design Automation</td>
</tr>
</tbody>
</table>

#### DESIGN OF ANALOG & MIXED MODE VLSI CIRCUITS

**Subject Code**: 10EC025  
**IA Marks**: 50  
**No. of Lecture Hours /week**: 04  
**Exam Hours**: 03  
**Total no. of Lecture Hours**: 52  
**Exam Marks**: 100

**Basic MOS Device Physics**: General considerations, MOS I/V Characteristics, second order effects, MOS device models.

**Single stage Amplifier**: CS stage with resistance load, divide connected load, current source load, triode load, CS stage with source degeneration, source follower, common-gate stage, cascade stage, choice of device models.

**Differential Amplifiers**: Basic difference pair, common mode response, Differential pair with MOS loads, Gilbert cell.

**Passive and active Current mirrors**: Basic current mirrors, Cascade mirrors, active current mirrors.

**Frequency response of CS stage**: source follower, Common gate stage, Cascade stage and Difference pair. Noise in CS stage, C- G stage, source follower, cascade stage, differential pair.

**Operational Amplifiers**: One Stage OP-Amp. Two Stage OP-Amp, Gain boosting, Common Mode Feedback, Slew rate, PSRR. Compensation of 2stage OP-Amp, Other compensation techniques.
Oscillators: Ring Oscillators, LC Oscillators, VCO, Mathematical Model of VCO.

PLL: Simple PLL, Charge pump PLL, Non-ideal effects in PLL, Delay locked loops and applications.

Bandgap References and Switched capacitor filetrs.

REFERENCE BOOK:


LABORATORY EXPERIMENTS:

ALL EXPERIMENTS MUST BE IMPLEMENTED USING VLSI TOOLS LIKE CADANCE/SYNOPSIS/MENTAGRAPHCICS.

1. Design the MOS transistor circuits for DC & AC small signal parameters, completing the design flow mentioned below:
   a. Draw the schematic and verify the following
      i) DC Analysis
      ii) AC Analysis
      iii) Transient Analysis
   b. Draw the Layout and verify the DRC, ERC
   c. Check for LVS

2. Design a TWO stage op-amp with given specification* using given differential amplifier Common source and Common Drain amplifier in library** and completing the design flow mentioned below:
   a. Draw the schematic and verify the following
      i) DC Analysis
      ii) AC Analysis
      iii) Transient Analysis
   b. Draw the Layout and verify the DRC, ERC
   c. Check for frequency response, slew rate, offset effects and Noise.

3. Design a simple sample and hold circuit and measure the switching times.

4. Design a PLL and measure all the parameters.

5. Design a simple ADC/DAC and measure the data conversion time.
   Assume the 95 nanometer technology.

6. Design 3-8 decoder using MOS technology.

ANY EXPERIMETNS CAN BE ADDED TO SUPPLEMENT THE THEORY. ABOVE IS THE ONLY GUIDE LINES.

REAL TIME OPERATING SYSTEMS

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>IA Marks</th>
<th>No. of Lecture Hours /week</th>
<th>Exam Hours</th>
<th>Total no. of Lecture Hours</th>
<th>Exam Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>10EC126</td>
<td>50</td>
<td>04</td>
<td>03</td>
<td>52</td>
<td>100</td>
</tr>
</tbody>
</table>


**Processing:** Preemptive Fixed-Priority Policy, Feasibility, Rate Monotonic least upper bound, Necessary and Sufficient feasibility, Deadline – Monotonic Policy, Dynamic priority policies.

**I/O Resources:**

**Memory:**
Physical hierarchy, Capacity and allocation, Shared Memory, ECC Memory, Flash filesystems.

**Multi-resource Services:**
Blocking, Deadlock and livestock, Critical sections to protect shared resources, priority inversion.

**Soft Real-Time Services:**
Missed Deadlines, QoS, Alternatives to rate monotonic policy, Mixed hard and soft real-time services.

**Embedded System Components:**
Firmware components, RTOS system software mechanisms, Software application components.

**Debugging Components:**

**Performance Tuning:**
Basic concepts of drill-down tuning, hardware – supported profiling and tracing, Building performance monitoring into software, Path length, Efficiency, and Call frequency, Fundamental optimizations.

**High availability and Reliability Design:**
Reliability and Availability, Similarities and differences, Reliability, Reliable software, Available software, Design trade offs, Hierarchical applications for Fail-safe design.

**Design of RTOS – PIC microcontroller. (Chap 13 of book Myke Predko)**

**Reference Books:**

**Real Time Operating Systems:**
**Laboratory Experiments**
(Reference Book 3 can be used to give Lab. Assignments)

USE LINUX/SOLARIS/QNX OS ONLY.
1. Implement simple IPC protocol.
2. Implement Semaphore and Mutex for any given applications.
3. Communicate between 2 PCs using Socket programming or message passing techniques (ie., MPI).
4. Create ‘n’ number of child threads. Each thread prints the message “I’m in thread number …” and sleeps for 50 ms and then quits. The main thread waits for complete execution of all the child threads and then quits. Compile and execute in Linux.

5. Implement the multithread application satisfying the following:
   i. Four child threads are cratered with normal priority.
   ii. Thread 1 & 2 receives and prints its priority and sleeps for 50ms and then quits.
iii. Thread 3&4 prints the priority of the thread 1 & 2 and raises its priority to above normal and retrieves the new priority of thread 1, prints it and then quits.

iv. The main thread waits for the child thread to complete its job and quits.

6. Implement the usage of send and receive primitives with 512 bytes for data sharing between parent and child processes using handle inheritance mechanism.

7. Test the program below using multithread application-
   1. The main thread creates a child thread with default stack size and name ‘Child_Thread’.
   2. The main thread sends user defined messages and the message ‘WM_QUIT’ randomly to the child thread.
   3. The child thread processes the message posted by the main thread and quits when it receives the ‘WM_QUIT’ message.
   4. The main thread checks the termination of the child thread and quits when the child thread completes its execution.
   5. The main thread continues sending the random messages to the child thread till the ‘WM_QUIT’ message is sent to the child thread.
   6. The messaging mechanism between the main thread and child thread is synchronous.

8. Test the program application for creating an anonymous pipe with 512 bytes of size and pass the ‘Read Handle’ of the pipe to a second process using memory mapped object. The first process writes a message ‘Hi from Pipe Server’. The 2nd process reads the data written by the pipe server to the pipe and displays it on the console. Use event object for indicating the availability of data on the pipe and mutex objects for synchronizing the access in the pipe. For synchronization semaphore/mutex can be used.

9. Create a POSIX based message queue for communicating between several tasks as per the requirements given below:
   i. Use a named message queue with name ‘MyQueue’. 
   ii. Create N tasks with stack size 4000 & priorities (n-1) & n respectively. N can be any number but more than 4.
   iii. Tasks create the specified message queue as Read Write and reads the message present, if any, from the message queue and prints it on the console.
   iv. Tasks open the message queue and posts the message ‘Hi from Task(n-1)’.
   i. Handle all possible error scenarios appropriately.

MINI PROJECTS: (optional)
1. Implement protocol converter (refer book 3 given in the RTOS theory)
2. Implement System Calls for the RTOS using RTLinux.
3. Implement an IP phone.
4. Implement Device Driver.

ANY EXPERIMENTS CAN BE ADDED TO SUPPLEMENT THE THEORY. ABOVE IS THE ONLY GUIDE LINES.

ADVANCED MICROCONTROLLERS (16-bit/32-bit)

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>: 10EC116</th>
<th>IA Marks : 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Lecture Hours /week</td>
<td>: 04</td>
<td>Exam Hours : 03</td>
</tr>
<tr>
<td>Total no. of Lecture Hours</td>
<td>: 52</td>
<td>Exam Marks : 100</td>
</tr>
</tbody>
</table>

Microcontrollers have become prevalent in a number of applications such as instrumentation, industrial electronics, automotive electronics, robotics, etc. Advances in VLSI technology permit the integration of not only the processor but also the analog electronics, memory and peripherals necessary for system implementation; this allows low-cost system implementation. Some microcontrollers used in industrial electronics also provide some digital signal processing capability to further reduce the system cost. Power dissipation is often a consideration in many systems and modern microcontrollers address it through the support of several low-power modes of operation. The aim of the course is to introduce advanced microcontrollers (16-bit and 32-bit).
**Motivation for advanced microcontrollers** – Low Power embedded systems, On-chip peripherals, low-power RF capabilities. Examples of applications.

**MSP430 – 16-bit Microcontroller family.** CPU architecture, Instruction set, Interrupt mechanism, Clock system, Memory subsystem, bus architecture. The assembly language and ‘C’ programming for MSP-430 microcontrollers. On-chip peripherals. WDT, Comparator, Op-Amp, Timer, Basic Timer, Real Time Clock (RTC), ADC, DAC, Digital I/O. Using the low-power features of MSP430. Clock system, low-power modes, Clock request feature, Low-power programming and interrupts.


**Applications** – Wireless Sensor Networking with MSP430 and Low-Power RF circuits; Pulse Width Modulation (PWM) in Power Supplies.

**References Books:**

4. Sample Programs for MSP430 downloadable from msp430.com

**LOW POWER VLSI DESIGN**

Subject Code : 10EC047     IA Marks : 50
No. of Lecture Hours /week : 04     Exam Hours : 03
Total no. of Lecture Hours : 52     Exam Marks : 100

**Introduction** : Need for low power VLSI chips, Sources of power dissipation on Digital Integrated circuits. Emerging Low power approaches, Physics of power dissipation in CMOS devices.

**Device & Technology Impact on Low Power**: Dynamic dissipation in CMOS, Transistor sizing & gate oxide thickness, Impact of technology Scaling, Technology & Device innovation

**Power estimation, Simulation Power analysis**: SPICE circuit simulators, gate level logic simulation, capacitive power estimation, static state power, gate level capacitance estimation, architecture level analysis, data correlation analysis in DSP systems, Monte Carlo simulation.

**Probabilistic power analysis**: Random logic signals, probability & frequency, probabilistic power analysis techniques, signal entropy.

**Low Power Design Circuit level**: Power consumption in circuits. Flip Flops & Latches design, high capacitance nodes, low power digital cells library

**Logic level**: Gate reorganization, signal gating, logic encoding, state machine encoding, pre-computation logic

**Low power Architecture & Systems**: Power & performance management, switching activity reduction, parallel architecture with voltage reduction, flow graph transformation, low power arithmetic components, low power memory design.

**Low power Clock Distribution**: Power dissipation in clock distribution, single driver Vs distributed buffers, Zero skew Vs tolerable skew, chip & package co design of clock network

REFERENCE BOOKS:


ELECTIVE –II

DESIGN OF VLSI SYSTEMS

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>IA Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>10EC027</td>
<td>50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No. of Lecture Hours /week</th>
<th>Exam Hours</th>
<th>Total no. of Lecture Hours</th>
<th>Exam Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>03</td>
<td>52</td>
<td>100</td>
</tr>
</tbody>
</table>


Chip Design Methods: Behavioral synthesis, RTL synthesis, Logic optimization and structural tools layout synthesis, layout synthesis, EDA Tools for System


Data Path Sub System Design: Introduction, Addition, Subtraction, Comparators, Counters, Boolean logical operations, coding, shifters, Multiplication, Parallel Prefix computations

Array Subsystem Design: SRAM, Special purpose RAMs, DRAM, Read only memory, Content Addressable memory, Programmable logic arrays.

Control Unit Design: Finite State Machine (FSM) Design, Control Logic Implementation: PLA control implementation, ROM control implementation.

Special Purpose Subsystems: Packaging, power distribution, I/O, Clock, Transconductance amplifier, follower integrated circuits, etc

Design Economics: Nonrecurring and recurring engineering Costs, Fixed Costs, Schedule, Person power, example


VLSI Applications: Case Study: RISC microcontroller, ATM Switch, etc.

REFERENCE BOOKS:


### VLSI DESIGN AUTOMATION

**Subject Code**: 10EC010  
**IA Marks**: 50  
**No. of Lecture Hours /week**: 04  
**Exam Hours**: 03  
**Exam Marks**: 100

**Logic Synthesis & Verification**: Introduction to combinational logic synthesis, Binary Decision Diagram, Hardware models for High-level synthesis.

**VLSI Automation Algorithms:**

**Partitioning**: problem formulation, classification of partitioning algorithms, Group migration algorithms, simulated annealing & evolution, other partitioning algorithms

**Placement, Floor Planning & Pin Assignment**: problem formulation, simulation base placement algorithms, other placement algorithms, constraint based floor planning, floor planning algorithms for mixed block & cell design. General & channel pin assignment

**Global Routing**: Problem formulation, classification of global routing algorithms, Maze routing algorithm, line probe algorithm, Steiner Tree based algorithms, ILP based approaches

**Detailed Routing**: problem formulation, classification of routing algorithms, single layer routing algorithms, two layer channel routing algorithms, three layer channel routing algorithms, and switchbox routing algorithms

**Over The Cell Routing & Via Minimization**: two layers over the cell routers, constrained & unconstrained via minimization

**Compaction**: problem formulation, one-dimensional compaction, two dimension based compaction, hierarchical compaction

**REFERENCE BOOKS:**


### MODERN DSP

**Subject Code**: 10EC123  
**IA Marks**: 50  
**No. of Lecture Hours/Week**: 04  
**Exam Marks**: 03  
**Exam Hours**: 100

**Goal of the course** – Advances in Digital Signal Processing involve variable sampling rates and thus the multirate signal processing and hence their applications in communication systems and signal processing. It is intended to introduce a basic course in multirate signal processing especially meant for students of branches eligible for M Tech courses in EC related disciplines.
Review of Signals and Systems – Discrete time processing of continuous signals - Structure of a digital filter; Frequency domain analysis of a digital filter; Quantization error; Sigma and Sigma Delta Modulation. Fourier Analysis – DFT, DTFT, DFT as an estimate of the DTFT for Spectral estimation. DFT for convolution, DFT/DCT for compression, FFT. Ideal Vs non ideal filters, FIR and IIR Filters Digital Filter Implementation; Elementary Operations.

Digital Filters –, State Space realization, Robust implementation of Digital Filters, Robust implementation of equi – ripple FIR digital filters

Multirate Systems and Signal Processing. Fundamentals – Problems and definitions; Upsampling and downsampling; Sampling rate conversion by a rational factor;

Multistage implementation of digital filters; Efficient implementation of multirate systems.


Maximally Decimated Filter banks – Vector spaces, Two Channel Perfect Reconstruction conditions; Design of PR filters Lattice Implementations of Orthonormal Filter Banks, Applications of Maximally Decimated filter banks to an audio signal.

Introduction to Time Frequency Expansion; The STFT; The Gabor Transform, The Wavelet Transform; The Wavelet transform; Recursive Multiresolution Decomposition.

References: