**19EE31P1-CONTROL SYSTEMS& SIMULATION LAB**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Category:** | Professional core | **Credits:** | 1.5 |
| **Course Type:** | Laboratory | **Lecture-Tutorial-Practical:** | 0-0-3 |
| **Pre-requisite:** | Linear control systems, Electrical Machines, Microprocessors and  MATLAB Software | **Sessional Evaluation:**  **External Exam Evaluation:**  **Total Marks:** | 40  60  100 |

|  |  |  |
| --- | --- | --- |
| **Course Objectives** | To make the student learn about: | |
| 1. The design and analysis of compensators.  2. The frequency & time domain specifications of network.  3. The Speed control of various DC & AC motors.  4.The characteristics of synchros  5.The design of controllers using MATLAB  6.To write the programme to find frequency & time domain  specifications of network using MATLAB | |
| **Course Outcomes** | After completing the coursethe student will be able to: | |
| CO1 | Apply appropriate compensator circuits experimentally. |
| CO2 | Analyse time and frequency specifications of network |
| CO3 | Examine the characteristics of various motors |
| CO4 | Enumerate the speed control of various motors using microprocessors. |
| CO5 | Demonstrate the usage of MATLAB in control system. |
| CO6 | Design the controllers. |
| **Course Content:** | Minimum of 10 experiments to be conducted out of the following:  **LIST OF EXPERIMENTS**   1. Characteristics of Lag - Lead & Lead - Lag compensator 2. Frequency response Specifications 3. Time response of first and second order System. 4. Characteristics of Synchros 5. Speed control of Stepper Motor 6. Speed control of DC Servo Motor 7. Root Locus & Bode plot for a given Transfer Function using MATLAB. 8. Simulation of P, PI and PID Controllers using MATLAB 9. AC Servo motor speed-torque characteristics 10. Polar & Nyquist plot for a given Transfer Function using MATLAB. 11. Testing of observability and controllability using MATLAB 12. Conversion of State Space Representation to Transfer function and vice-versa using MATLAB | |