Course:
Python for Control System Laboratory Works - From Basic Control System to PID Control System

Course Description:
This course is a hands-on laboratory-based course that focuses on implementing control systems using Python programming. Students will gain practical experience in designing, simulating, and implementing control systems, starting from basic control system principles and progressing to the implementation of PID (Proportional-Integral-Derivative) control systems. By the end of the course, students will have acquired knowledge and skills in implementing various control systems using Python.

Prerequisites:
- Basic knowledge of control system theory
- Proficiency in Python programming language

Course Objectives:
Upon completion of this course, students will be able to:
1. Understand the principles and components of control systems.
2. Apply Python programming for system analysis and design.
3. Design and implement basic feedback control systems.
4. Implement and fine-tune PID control systems using Python.
5. Analyze and optimize the performance of control systems.
6. Develop control system applications using Python.

Course Outline:

Week 1: Introduction to Control Systems and Python
- Introduction to control systems and their applications
- Overview of Python programming language for control systems simulation
- Python environment setup and installation
- Introduction to Python libraries for control systems simulation

Week 2: Modeling of Dynamic Systems
- Review of system modeling concepts (differential equations, transfer functions)
- Mathematical modeling of electrical, mechanical, and thermal systems using Python
- Transfer function representation and block diagrams in Python
- Python tools for system identification and parameter estimation

Week 3: Time Response Analysis
- Time-domain analysis of control systems using Python
- Simulation and analysis of step, ramp, and impulse responses
- Performance metrics: settling time, rise time, peak time, and steady-state error
- Python libraries for time response analysis

Week 4: Frequency Response Analysis
- Frequency-domain analysis of control systems using Python

- Bode plots, Nyquist plots, and Nichols plots in Python
- Gain and phase margins
- Python libraries for frequency response analysis

Week 5: Controller Design
- Introduction to controller design techniques (PID, lead-lag, etc.)
- Design of controllers using Python libraries (such as SciPy, Control, or custom implementations)
- Controller tuning methods (Ziegler-Nichols, frequency response methods) in Python
- Simulation and evaluation of designed controllers using Python

Week 6: Stability Analysis and Design
- Stability analysis using root locus and Nyquist criteria in Python
- Stability margins and robustness analysis using Python
- Controller design for stability using Python libraries
- Simulation and analysis of stability using Python

Week 7: Simulation and Visualization
- Advanced simulation techniques for control systems using Python
- Modeling and simulation of complex systems in Python
- Simulation of control system behavior under different scenarios using Python
- Analysis, visualization, and interpretation of simulation results in Python

Week 8: Mid-Term Exam
- Assessment of theoretical knowledge and practical skills acquired during the course
- Exam covering topics from Weeks 1 to 7

Assessment Methods:
- Laboratory exercises and assignments
- Project work and presentations
- Midterm and final examinations