NOVA COLLEGE-WIDE COURSE CONTENT SUMMARY ELE 233 – PROGRAMMABLE LOGIC CONTROL SYSTEM I (3 CR.)

Course Description

Teaches operating and programming of programmable logic controllers. Covers analog and digital interfacing and communication schemes as they apply to system. Part I of II. Lecture 2 hours. Laboratory 3 hours. - Total 5 hours per week.

General Course Purpose

This course is designed to teach students the basic concepts of Programmable Logic Controllers. This course will teach students Ladder logic programming using Allen Bradley Compact Logix Hardware and Studio 5000 Software. Students will learn to program basic input and outputs, PLC timers and counters, Sequencers, Programmable controls, and Data manipulation instructions. This course will also teach students the basics of HMI and HMI programming using Allen Bradley Hardware and software.

Course Prerequisites/Corequisites

Prerequisites: ETR 156 and ETR 211, or equivalent.

Course Objectives

After the completion of this course, students will be able to:

- Operate different types of PLCs, including Allen Bradley PLC
- Write basic PLC programs
- Evaluate the role of PLCs in modern automation systems
- Compose PLC instructions for PLC Communication
- Design and write Ladder Logic Programs for PLC applications
- Use PLC for motor control applications
- Design analog modules based on analog input and output
- Employ relay type instructions (XIC, XIO, OTE, OTL, and OTU), timing and counting instructions, and sequencer instructions to solve PLC problems
- Use Allen Bradley FactoryTalk View to design and organize HMI tags

Major Topics to be Included

- Review of electrical quantities, Electrical Safety
- Introduction To PLC
- Compact Logix
- Ladder Logic programming
- XIC, XIO, OTE
- Relays, Contactors, Overloads, Motor Starters, Seal in
- Control Devices, Manual, Magnetic and automatic switches, Sensors
- Reversing Motors using PLC
- Timing, Counting
- MOV, Math and program control
- Analog Control
- PLC Motor Control
- HMI Programming
- Advance Topics