EE499 End-of-Term Project Guidelines
Software-Oriented Design Project

To conclude the EE499 Three-Phase AC Motor Control course, students must complete a final project which has emphasis on either the hardware or software development of a power electronics inverter.

For the software-oriented project, students must develop and demonstrate a motor control algorithm written on an embedded microcontroller system. The student is free to choose which development system they would like to use, although the Arduino Uno + Robot Power MegaMoto platform is recommended. The system will control a DC motor using an H-Bridge circuit topology, where the motor current is regulated. The software control algorithm must include the following elements or features…

- Initialize the PWM modules of the microcontroller to operate in the kHz frequency range, and configure the PWM signals to operate in a center-aligned mode.
- Configure the appropriate ADC channels and their sample / hold acquisition times to appropriate values for use in a kHz frequency control loop.
- Use the low-priority main-loop of the processor to periodically send and receive data messages to the Human-Machine-Interface (HMI). This interface should be used to configure the microcontroller system to operate in different modes or control states. The communication protocol used should be RS232 or similar. Student's must write communication code which can do the following:
    - Change the operating state of the controller between one of three modes: off, open-loop control, and closed-loop control.
    - Change the duty cycle output of the controller manually when in open-loop mode.
    - Change the current reference set-point of the controller when in closed-loop mode.
    - Adjust the PI controller k_p and k_i parameters for on-line tuning of the controller.
- Utilize a CPU timer interrupt or PWM timer interrupt to periodically perform the following tasks:
    - Collect ADC data from on-board current sensors and convert the raw data into a fixed-point equivalent real-world value.
    - Compare the sensed current value to a reference current to generate an error signal. Feed the error signal to a PI controller for voltage regulation.
    - Implement a PI control function in code. Use the Forward Euler integration method for the controller integrator portion. The integrator and the PI output should have saturation limits.
    - Utilize the PI controller output to directly drive the PWM duty cycle value, which in-turn will control the average output voltage of the H-Bridge circuit.