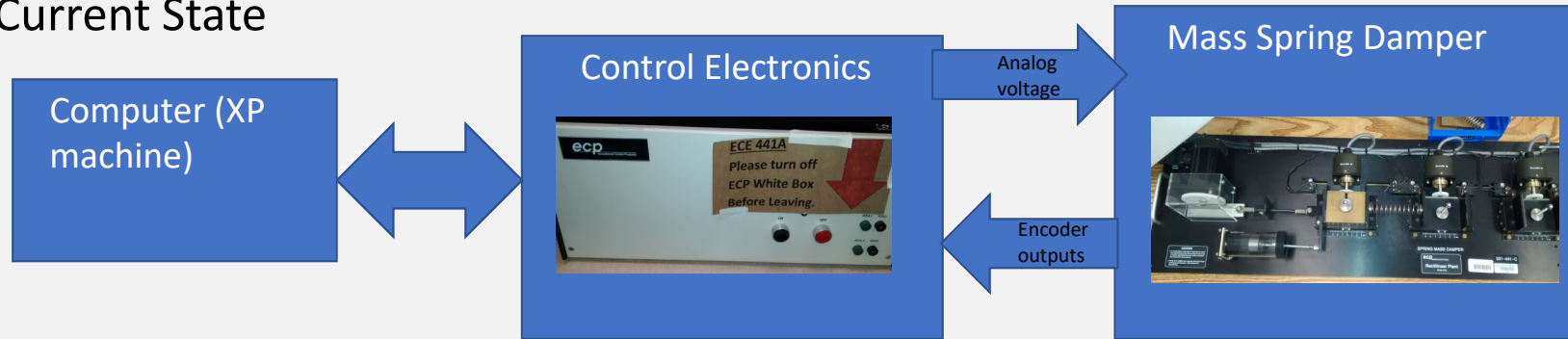


ECE 542 Project Proposal

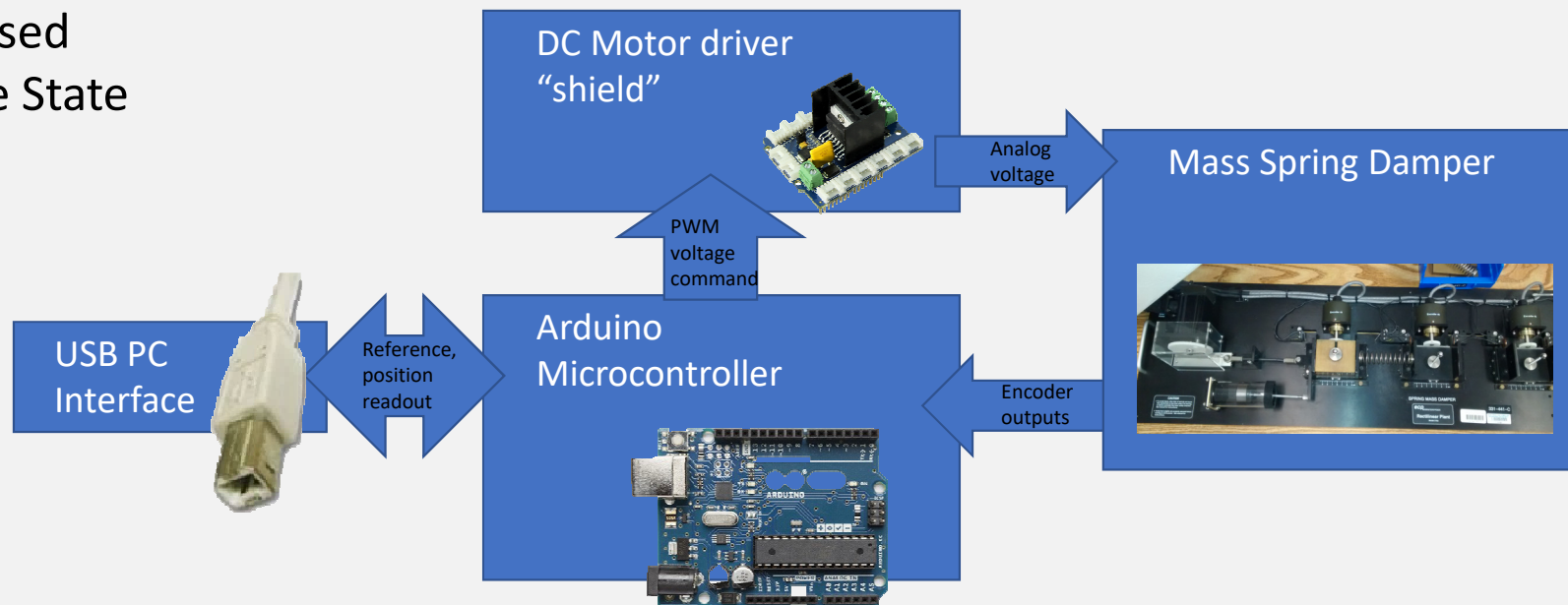
Control mass-spring damper with Arduino

- Mass Spring Damper Setup on 3rd floor in the senior capstone room has a few problems
 - Older computer
 - No Network connection
 - Proprietary parallel interface leaves little room for PC upgrade
- Arduino could connect to existing hardware interface (motor/encoder) to bypass proprietary electronics box
 - Familiar Arduino programming interface for students
 - USB connection works with modern computers

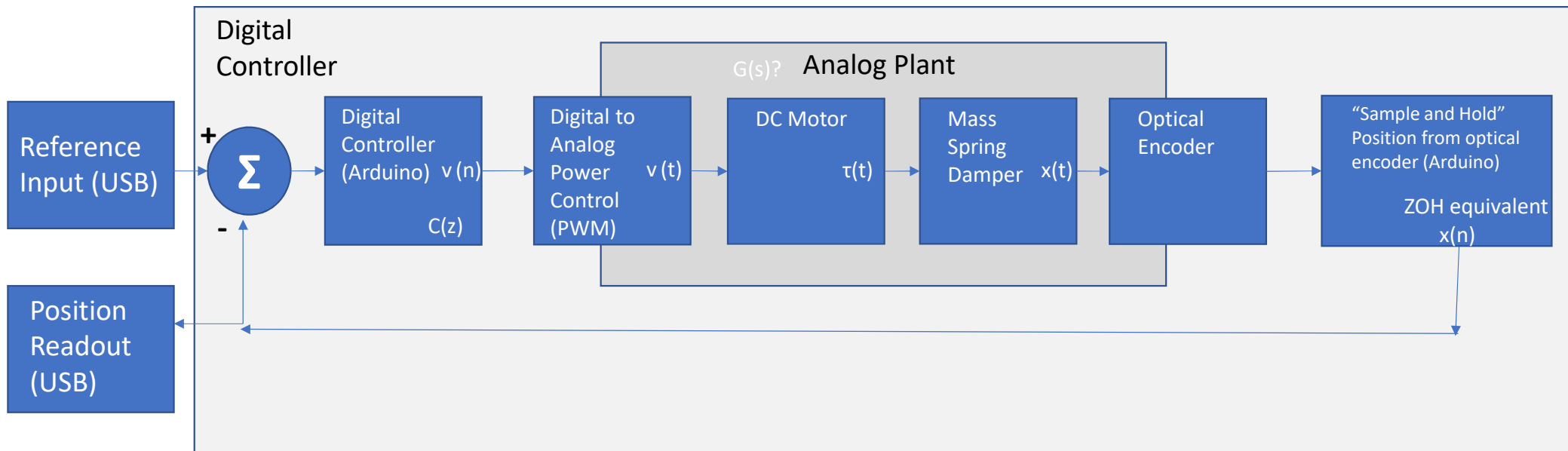
Current State



Proposed Future State



Proposed control scheme

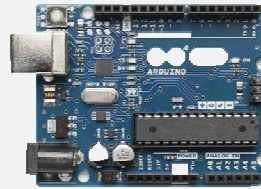


Hardware Needed

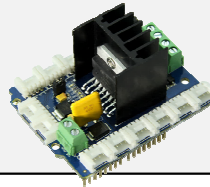
2x 16-pin circular connectors



Arduino Uno



Seed studio motor shield



Hardware enclosure,
wiring, etc.

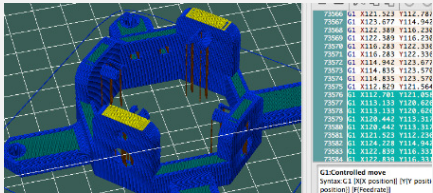


**Most of the hardware work
will be wiring, soldering,
integration, debugging**

Software Needed

USB Interface

Plan to use open-source G-CODE interface used on RepRap 3d printers. Arduino module for this terminal exists, but may have to be adapted to fit into the rest of my Arduino code.



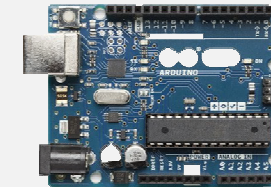
PC Terminal

Existing 3d printer software will allow sending of G-CODE commands, but not data capture and plotting. This may be deferred for a future project to try to keep the scope manageable

Arduino Code

The majority of the project will be writing a C++ program to do the following tasks at a particular sample rate:

- Count pulses from optical encoder and translate into distance
- Keep track of distance vs reference input, subtract these to produce error signal.
- Apply recursive difference equation on error signal using pre-programmed coefficients to generate commanded output to motor
- Update PWM duty cycle for motor shield output



A simple example

A simple PID controller will be used to prove the system. I will generate a ZOH equivalent of the mass spring damper and motor system, and design controller coefficients using techniques from class.

P I D

I will develop these two pieces as part of this project

Risks, Assumptions, and Other Concerns

- PWM output from Arduino may not generate linear torque response on motor
 - May need to generate a lookup table to translate between PWM duty cycle and expected motor torque.
- Will be able to connect to Mass-Spring-Damper hardware with an Arduino
 - I have DC motors and potentiometers at home to complete the project if I can't figure out how to connect to the Mass Spring Damper hardware in the ECE lab
- Using open source software as part of project
 - I will segregate any open source code into a "libraries" folder identified by a readme file as not my own work.
 - I will include a header identifying myself as the author for the source code that is my own work.