**HARDWARE**

Top Priorities

* Automatic sample time adjustment per chosen input frequency (partially tested, added something to allow stoptime to be calculated by either frequency1 or frequency2 to primary code)
* Input Mode - Oscillatory vs. Step - can test this on non-RPi code (done)
* Software RESET at end of program rather than entering manual mode (done)
* Emergency Stop Hardware connection RPi to RESET (done - note to wait 5 seconds to resubmit commands from RPi)
* Add buttons/controls to control second motor positioning (tested, add to primary code)
* Redo control equation (done, add frequency band control equations to primary code)
* TEST ALL WITH RPI

In Primary Code, test these things:

* Is default fs for STEP 50Hz?
* STEP/OSCILLATION modes
* Software reset
* stopTime calculated by motor 1 or motor 2 input

Lowest Priorities - only if time

* Enable Motor 2
* Improve control equation, proper design
* Limit switches - mechanical hard stops (don’t have 4, so could only do 1 DOF anyway)
* Emergency Stop in code due to maximum position exceeded (need to test actual max positions on platform) - this avoids having to use the ugly limit switches (this is only good if encoder is working, worthwhile?)

**DOCUMENTATION**

* Clean up/Document all code
* How To document for RPi interface/Xmega combo
* How To to program Xmega without RPi (including documented code for manual entry via Xmega)
* Clean up Dynamic Model
* Clean up Encoder/Sample Time work
* Data Collection/Figure - “amplitude ratio vs input amplitude from 1-7 degrees” to show high friction response over range

**Current code for Documentation**

* “A1U Manual Position Mode R04”, has all features for one motor plus buttons for second motor, for direct programming of Xmega
* “A1 Platform Controller with SPI R04” has all up until adding any second motor features (so has step/input mode, software reset, test change of SPI transactions outside of if statement inside interrupt)
* “A1U Platform Ctrl w Motor 2 Position” has motor 2 buttons only enabled