# Purpose and Scope

* 1. Define a standardized method for versioning hardware, firmware, and software projects. This is not an attempt to re-invent the wheel, only a definition of versioning guidelines to promote consistencies and remove ambiguity.

# Hardware Versioning

* 1. Physical parts all begin in the numeric form: ***Rev. 01***
  2. When prototypes move into production the form will change to alpha: ***Rev. A***
  3. Avoid the following alpha revision letters: *I, O, Z, S, X, & Z* per ASME standards.
  4. Rolling over alpha revision letters will go like this: ***Rev. Y*** 🡪 ***Rev. AA***
  5. If a version of hardware is sent outside for review, the revision must be upped if changes are made: ***Rev. B*** 🡪 ***Rev. B1***

# Firmware & Software Versioning

* 1. Follow the same basic idea as semantic versioning
  2. There are four levels: vA.B.C.D
     + A = Major
     + B = Minor
     + C = Bug Fix
     + D = Build
  3. The version shall be retrievable by a user via a command or displayed on the GUI
  4. Major: Increment when the code is no longer backwards compatible with previous revisions or APIs.
  5. Minor: Increment when new features or improvements are added such as performance updates.
  6. Bug Fix: Increment when small portions of the code were changed to address bugs or issues.
  7. Build: Increments each time the code successfully compiles. This should increment automatically.

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Description** | **Author** | **Date** |
| 01 | Initial Release | J. Petrilli | 10/30/2022 |
|  |  |  |  |