Commands

I have tried to come up with enough commands and settings that will cover most scenarios.
- Commands start with a Capital Letter immediately followed by a number, there are a little like Geometric (G) Code.
- Some commands may also have a value, this also starts with a Capital Letter immediately followed by a number.
- A space will separate the two above.
- Mostly values are the number of ticks from an Encoder or 1 or 0 to represent a Boolean.
- As these commands are sent over I2C, each Motor will have it's own Address that the Commands are sent to.

'C' Commands (settings):

- C0 = Reset things to 0 (zero). To Do. Not used at the moment.
- C1 = Sets the Current Position to 0 (zero). Just use C1.
- C2 B<value> = Set Has a Maximum Position. Boolean = 1 or 0 (true or false).
- C3 P<value> = Set the Maximum Position value. Value = a 24+ bit value, Sets maximum Forward position.
- C4 B<value> = Set Has a Minimum Position. Boolean = 1 or 0 (true or false).
- C5 P<value> = Set the Minimum Position value. Value = a 24+ bit value, Sets maximum Reverse Position.
- C6 B<value> = Set Has a Station Position. Boolean = 1 or 0 (true or false).
- C7 P<value> = Set the Forward Station Position value. Value = a 16 bit value, Sets the Station Position while moving Forward.
- C8 P<value> = Set the Reverse Station Position value. Value = a 16 bit value, Sets the Station Position while moving in Reverse.
- C10 A<value> = Set I2C Address. Value = 1 (0x00) to 127 (0x7F). This is to change the I2C Address of the device.
- C11 P<value> = Motor Slowdown Point. Value = a 16 bit value, Sets the number of tick before at target stop point, the motor will reduce speed.
- C12 P<value> = Motor Speed Reduction. A Percentage, 0 to 100% reduction of current speed.
- C13 B<value> = Set I2C Speed. This is a Boolean, Is it 400k. Value = 1 or 0 (true or false). 1 = 400000, 0 = 100000.

All settings are saved in non-volatile memory.
There is no need to apply settings every time power it connected.

'S' Command (Speed):

- S<Value> = Set the **S**peed 0 to 1000. This is a PWM value.

So 500 would be 50% on 50% off square wave.

'F' Commands (Forward):

- F0 = **Stop**. Can just use 'F' as the buffer will be full of zeros.
- F1 = **F**orward at a current **S**peed.
- F1 S<value> = **F**orward at a set **S**peed.

'R' Commands (Reverse):

- R0 = **Stop**. Can just use 'R' as the buffer will be full of zeros.
- R1 = **R**everse at a current **S**peed.
- R1 S<value> = **R**everse at a set **S**peed.

'G' Commands (Go to):

- G0 = **Stop**. Can just use 'G' as the buffer will be full of zeros.
- G1 P<value> S<value> = Move to an absolute **P**osition Forward (Positive) of zero at a **S**peed.
- G2 P<value> S<value> = Move to an absolute **P**osition Reverse (Negative) of zero at a **S**peed.

*The **'G'** code **P**<value> can be negative values. So why have G1 and G2?*

- *To keep options open.*
- *All positions are relative to a zero.*
- *When writing your own code to control the Motors you may only have the option of Unsigned Integers.*
- *You may want to only use Unsigned Integers to save memory.*