**Sending Byte Packages to the Arduino**

We will send packages of eight bits to the Arduino board via the serial port, representing a **command package**. The eight bit code is as follows:

X X X X X X X X

Blue (Command): This represents the command code. Controlling the pins will serve as a command, while causing a delay or something like that will be a different code.

Red (Pin Mapping/Command Mapping): When commanding a pin, this will serve as the pin number. Of course, if the command is to cause a delay, this code might represent the delay time.

Green (On/Off, State): When relevant, such as when controlling a pin, this is the on/off state.

The board will read a collection of bytes from the serial port. Each of these bytes will represent the eight bit code specified (each byte is a command package).

**Receiving Byte Packages from the Arduino**

The Arduino board will reply with a package via the serial port. The reply package will change depending on the context (if the sent package was controlling a pin, it will have a different meaning than another package). For pin package replies, this is the basic code I've come up with.

X X X X X X X X

Red (Pin Mapping/Command Mapping): This is the pin mapping.

Green (State): This is the state. 0 means the pin is off and 1 means it's on. The code 111, or 7, means there's an error with the pin. We can designate a few other state codes in here if we want.