# Arduino Dyno

## Overview

Using an Arduino Uno to control the braking force to the retarder and passing data to the PC to be displayed and logged. In order to achieve this The Arduino must send the retarder RPM and load sensor reading via usb to the PC, which will calculate the power and display it and/or log it. There will be two setting. Tuning: where the Arduino will hold the RPM set by a pot, which will enable ECU tuning at all throttle positions. Power run: where the Arduino will create a braking force to enable a full throttle power run lasting approximately 10 seconds which can then be logged by the PC.

## Arduino

### Start Up

Arduino will listen to PC for what settings to run and report back to PC.

1. Tuning or Power Run mode .Whilst running in either mode, It will need to listen for this setting to change.
2. Expected Max engine RPM converted by the PC, using the gear ratio, to retarder RPM.
3. Expected maximum power, with which can be used to calculate a braking force for the power run mode

### RPM

An optical sensor from a printer and a home made disc will be used to measure the retarders RPM. The retarders max RPM is 4500. As the sensor comprises of an LED, a 10K ohm resistor will have to be used. The disc will have two slots in it for the Arduinos hardware interrupt (pins 2 and 3 I believe) to count. A resolution of 100ms is hoped for therefore the Arduino will have to count the number of interrupts every 10th of a second and divide by the number of slots in the disc(2) to give revs per 10th of second. Multiply this by 600 to give RPM. Send the result to the PC via USB.

* RPM=(Count/2)\*600

### Load Cell

Details of the load cell are to be listed.

The load cell reading must be converted to pounds and as the distance from the Retarder centre to the load cell is 1ft, the result will be ft/lbs and sent to the PC for Power calculation.,

### Pot

The Arduino will be told what RPM to hold using a pot. The pot can be programmed using the Map function, mapping to RPM values of 500 to 4500. The pot will be turned whilst reading the ECU output until the required RPM is met.

### Power output

When the pot changes value, Arduino will calculate the difference between required RPM and actual RPM measured by the optical sensor. Using a formula (to be determined), Arduino will calculate how much more or less power to send to the retarder. 0 - 5v can be sent via an analogue pin using a value between 0 and 255 0 = 0v and 255 = 5v. I am not sure if this resolution will be accurate enough as 5v sent to the retarder power supply will equate to 196v at the retarder.

* At 4500 RPM and 196v supply, the retarder will produce a braking force of 800 ft/lb. Therefore 800/196 = 4 ft/lb per volt.
* 196/255 = 0.77v per analogue point.
* 4\*0.77 = 3.08 ft/lb per analogue point