MS3Pro Manual

April 11, 2025

MS3Pro Mini User Manual
Firmware version 1.6.1

Contents

1 Introduction **13**
1.1 Overview . . . 13
1.1.1 Warning labels . . . 13
1.1.2 Technical support . . . 13
1.1.3 Copyrights . . . 13
1.2 MS3Pro components . . . 13
1.2.1 MS3Pro Engine Control Unit Family . . . 13
1.2.1.1 MS3Pro Module . . . 13
1.2.1.2 MS3Pro Plug and Play . . . 13
1.2.1.3 MS3Pro EVO+ . . . 14
1.2.1.4 MS3Pro Mini . . . 14
1.2.2 Wiring harness . . . 14
1.2.3 Tuning cables . . . 15
1.3 MS3Pro accessories . . . 15
1.3.1 Sensors . . . 15
1.3.2 QuadSpark . . . 15
1.3.3 Ignition coils . . . 15
1.3.4 CAN-EGT thermocouple interface . . . 15
1.3.5 MicroSquirt . . . 15
1.3.6 HSD-4 High Side Driver module . . . 16
1.3.7 AXM AMP eXpansion Modules . . . 16
1.4 Tools . . . 16

2 Installing the software **18**
2.1 Registering TunerStudio . . . 18
2.2 TunerStudio . . . 18
2.2.1 Start screen . . . 18
2.2.2 Creating a project . . . 19
2.2.3 TunerStudio main screen . . . 21
2.2.4 Loading and saving tunes . . . 21
2.3 Tune Analyze Live . . . 22

3 MS3Pro hardware **23**
3.1 Overview . . . 23
3.2 Inputs . . . 24
3.2.1 Engine speed . . . 24
3.2.2 Temperature inputs . . . 24
3.2.3 Throttle position . . . 25
3.2.4 O_2 sensor input . . . 25
3.2.5 MAP sensor input . . . 25
3.2.6 General purpose analog inputs . . . 25
3.2.7 Knock input . . . 26
3.2.8 Digital input channels . . . 26
3.2.9 Barometric pressure input . . . 26
3.3 Outputs . . . 26
3.3.1 Injector outputs . . . 26
3.3.2 Ignition outputs . . . 27
3.3.3 High current outputs . . . 27
3.3.4 PWM medium current outputs . . . 27
3.3.5 Tach output . . . 27
3.3.6 Stepper motor control output . . . 28
3.4 Communications lines . . . 28
3.4.1 RS232 . . . 28
3.4.2 USB . . . 28
3.4.3 CANbus . . . 28

3.5 Maintenance inside the case . . . 28
3.5.1 Changing the fuses . . . 28
3.5.2 Replacing the SD card . . . 29
3.5.3 Changing the real time clock battery . . . 29
3.5.4 Water seal specifications . . . 29
3.5.5 Mounting footprint . . . 29

4 Wiring **30**
4.1 ECU Connectors . . . 31
4.2 Wiring diagram . . . 31
4.3 Engine position sensors . . . 35
4.3.1 Variable reluctor sensors . . . 35
4.3.2 Hall effect and optical sensors . . . 36
4.3.3 Points triggering and points replacement devices . . . 36
4.3.4 Fuel only installations . . . 37
4.4 Other sensors . . . 37
4.4.1 Temperature sensors . . . 37
4.4.2 MAP sensors . . . 37
4.4.3 Throttle position sensor . . . 38
4.4.4 Mass air flow sensors . . . 38
4.4.4.1 Ford 5.0 Mustang 4 pin oval connector MAF . . . 39
4.4.4.2 Ford 5 pin rectangular connector MAF . . . 39
4.4.4.3 Nissan / Infiniti Q45 MAF . . . 39
4.4.5 Oxygen sensors . . . 40
4.4.6 Knock sensors . . . 41
4.4.7 Speed and gear sensors . . . 41
4.4.8 Generic on/off switches . . . 41
4.4.9 Other sensors . . . 42
4.5 Fuel injectors . . . 42
4.6 Ignition outputs . . . 42
4.6.1 Conventional high-current / power transistor driven coils . . . 42
4.6.1.1 QuadSpark ignition module . . . 43
4.6.1.2 MSD and other single channel capacitive discharge ignitions . . . 43
4.6.2 Logic-level 'Smart' Coils (and coil packs) . . . 43
4.6.2.1 IGN-1A coil with built in ignition module . . . 43
4.6.2.2 LS series coils . . . 43
4.6.2.3 VW coils . . . 44
4.6.2.4 GM D599A 6-cylinder wasted-spark logic coil pack . . . 45
4.6.2.5 M&W capacitive discharge ignitions . . . 46
4.6.2.6 Common firing orders . . . 47
4.7 IAC valves . . . 48
4.7.1 Stepper IAC valves . . . 48
4.7.2 PWM IAC valves . . . 48
4.7.3 On/off IAC valves . . . 48
4.7.4 Thermal IAC valves . . . 48
4.7.5 DC servo IAC valves (currently not supported) . . . 48
4.7.6 Fuel pump relay . . . 49
4.7.7 Tach output . . . 49
4.7.8 High current outputs . . . 49
4.7.9 PWM medium current outputs . . . 49
4.7.10 Solid state relay control . . . 49
4.7.11 Grounding notes . . . 51
4.7.12 Output comparison chart . . . 51

5 Setting up a basic configuration **52**
5.1 Connecting the MS3Pro to your laptop . . . 52
5.2 Basic engine constants . . . 53
5.3 Ignition settings . . . 55
5.3.0.1 Distributor rotor phasing . . . 61
5.3.1 Basic Trigger . . . 61
5.3.1.1 Input phasing . . . 62
5.3.1.2 Ford TFI distributors . . . 63
5.3.1.3 GM HEI . . . 63
5.3.1.4 MSD distributor based ignitions . . . 64
5.3.2 Trigger Return . . . 64
5.3.3 Fuel only . . . 64
5.3.4 Toothed wheel . . . 64
5.3.4.1 Terminology notes . . . 66
5.3.4.2 Wheel naming . . . 66
5.3.4.3 Specific settings . . . 67
5.3.4.4 Existing . . . 67
5.3.4.5 Retrofit . . . 67
5.3.4.6 Missing tooth crank wheel . . . 68
A note about VR sensors and missing tooth wheels . . . 70
5.3.4.7 Missing tooth cam wheel . . . 70
5.3.4.8 Missing tooth crank wheel and single tooth cam wheel . . . 70
5.3.4.9 Missing tooth crank wheel and polled (50/50 or half moon) cam wheel . . . 72
5.3.4.10 Missing tooth crank wheel and irregular cam wheel . . . 76
5.3.4.11 Nippon Denso CAS . . . 78
5.3.4.12 Non-missing tooth cam wheel with single-tooth cam . . . 79
5.3.4.13 Non-missing tooth cam wheel with two opposite teeth on the cam . . . 82
5.3.4.14 Non-missing tooth cam wheel with one cam tooth per cylinder . . . 82
5.3.4.15 Non-missing tooth crank wheel with one cam tooth . . . 83
5.3.4.16 Example wheel decoder settings . . . 86
BMW inline sixes . . . 86
Buick C3I . . . 86
Chevrolet LS2, LS3 and other GM 58X V8s . . . 86
Chrysler 318 / 360 Magnum V8 . . . 87
Chrysler VVT Hemi . . . 87
Ford 36-1 . . . 87
Ford Coyote . . . 88
Ford Coyote GEN3 . . . 88
Toyota 2JZ-GTE . . . 88
Other wheel arrangements . . . 89
EDIS and EDIS Multispark . . . 89
Checking the timing . . . 90
Cam sensor and EDIS . . . 91
420A/Neon . . . 91
36-2+2 . . . 92
36-2-2-2 . . . 92
Subaru 6/7 . . . 93
Miata 99-05 . . . 93
6G72 . . . 93
IAW Weber . . . 94
CAS 4/1 . . . 94
4G63 . . . 95
Twin trigger . . . 95
Chrysler 2.2 / 2.5 . . . 96
Renix 44-2-2 . . . 96
Suzuki Swift . . . 96

Suzuki Vitara 2.0 . . . 97
Daihatsu 3cyl . . . 97
Daihatsu 4cyl . . . 97
VTR1000 . . . 98
Rover #1 . . . 98
Rover #2 . . . 98
Rover #3 . . . 99
GM 7X . . . 99
QR25DE . . . 99
Honda RC51 . . . 100
Fiat 1.8 16V . . . 100
Optispark . . . 100
Nissan SR20 . . . 102
Nissan RB25 . . . 102
LS1 . . . 103
YZF1000 . . . 103
Honda Acura (V6) . . . 104
VQ35DE . . . 104
Jeep 2000 . . . 104
Jeep 2002 . . . 105
Zetec VTC . . . 105
Flywheel tri-tach . . . 105
2JZ VVTi . . . 106
Honda TSX / D17 . . . 106
Viper V10 . . . 107
HD32-2 . . . 107
Miata 36-2 . . . 108
Daihatsu 12+1 . . . 108
Subaru 36-2-2-2 VVT . . . 108
Ski Doo PTEC . . . 109
Nissan QG15 . . . 109
Mazda MZR . . . 110
6G75 . . . 110
Idle valves . . . 110
On / Off settings . . . 111
Stepper valve settings . . . 111
PWM idle valve settings . . . 111
Sensor calibration . . . 111

6 Tuning the MS3Pro 114
6.1 Getting started . . . 114
6.1.1 Using output test mode . . . 114
6.1.1.1 Injector test mode . . . 114
6.1.1.2 Coil test mode . . . 115
6.1.1.3 Idle valve testing . . . 115
Testing a stepper IAC valve . . . 116
Testing a PWM IAC valve . . . 117
6.1.2 Checking RPM . . . 117
6.1.3 Setting cranking timing . . . 117
6.1.4 Starting the engine . . . 118
6.1.5 Setting running timing . . . 119
6.2 Basics of 3D tuning tables and loads . . . 119
6.3 Tuning fuel . . . 122
6.3.1 Dead time . . . 122
6.3.1.1 Injector settings . . . 123
6.3.2 Small pulse widths . . . 123
6.3.3 AFR table . . . 124

6.3.4 VE table . . . 125
6.3.4.1 VE table generator . . . 126
6.3.5 Basic acceleration enrichment (AE) tuning . . . 126
6.3.5.1 Main accel enrich settings menu . . . 127
6.3.5.2 Accelerator pump AE . . . 127
6.3.5.3 Time Based AE . . . 128
6.3.5.4 Getting it dialed in . . . 130
MAP based or TPS based AE? . . . 131
Setting the TPSdot/MAPdot Threshold Levels . . . 131
Tuning the tables . . . 131
6.3.6 Startup / warmup fueling . . . 132
6.3.6.1 Cranking / Startup settings . . . 132
6.3.6.2 Priming pulse . . . 133
6.3.6.3 After-start (ASE) percentage . . . 133
6.3.6.4 After-start (ASE) taper . . . 133
6.3.6.5 Warmup enrichment . . . 133
6.3.6.6 Putting it all together . . . 133
6.4 Tuning spark . . . 134
6.5 Getting a good idle . . . 134
6.5.0.1 Correct Fuel . . . 135
6.5.0.2 Correct Timing . . . 135
6.5.0.3 Correct Airflow . . . 136

7 Advanced Tuning 137
7.1 Basic / Load Sections . . . 137
7.1.1 General Settings . . . 137
7.1.1.1 Barometric settings . . . 137
7.1.1.2 MAP sensor settings . . . 138
7.1.1.3 General Sensor Settings . . . 138
7.1.1.4 Load Parameters . . . 139
7.1.2 Rev Limiter . . . 139
7.1.2.1 Hard rev limit . . . 140
7.1.2.2 Coolant temp limiter . . . 140
7.1.2.3 Spark retard . . . 140
7.1.2.4 Spark cut . . . 140
7.1.2.5 Fuel cut . . . 140
7.1.3 Shift light . . . 141
7.1.4 Engine state settings . . . 141
7.1.4.1 How the MS3Pro uses engine states . . . 142
7.1.4.2 Tuning engine states . . . 143
7.1.5 Tacho Output . . . 143
7.1.6 Fan control . . . 144
7.1.7 Torque converter lockup . . . 145
7.1.8 Alternator control . . . 146
7.1.8.1 Ford alternators . . . 148
7.1.8.2 Chrysler alternators . . . 149
7.1.8.3 Miata alternators . . . 149
Pull up method . . . 149
Spark output method . . . 149
7.1.8.4 Chevrolet 4 wire alternators . . . 149
Simple installation . . . 150
On/off control . . . 150
On/off control with load monitoring . . . 150
7.1.8.5 Chevrolet 2 wire alternators . . . 150
7.1.8.6 Honda alternators . . . 151
7.1.8.7 Other alternators . . . 151
7.1.9 Barometric correction . . . 151

7.1.10 MAP sample settings . . . 152
7.1.10.1 Event Average . . . 153
7.1.10.2 Timed Minimum . . . 153
7.1.11 ITB load settings . . . 155
7.1.11.1 ITB load VE table . . . 155
7.1.11.2 ITB load TPS switch point curve . . . 156
7.1.11.3 ITB Load at TPS Switchpoint Curve . . . 156
7.1.11.4 Putting it all Together - Calculating ITB Load . . . 156
7.1.11.5 ITB Load Calculation in Speed-Density Mode . . . 156
7.1.11.6 ITB Load Calculation in Alpha-N Mode . . . 157
7.1.11.7 Tuning For Idle Air Control . . . 158
7.1.12 MAF options . . . 158
7.1.12.1 More about frequency based MAFs . . . 160
7.1.13 MAF flow curve . . . 160
7.1.13.1 MAF Load . . . 160
7.1.13.2 Tuning . . . 161
7.1.13.3 Using VE1 as a trim table . . . 161
7.1.14 MAF correction table (old) . . . 161
7.1.15 MAF / MAT correction table . . . 162
7.1.16 MAT Air Density Table . . . 162
7.1.17 MAT / CLT Correction . . . 163
7.1.18 Realtime Display . . . 164
7.1.19 Feature List Showing I/O Pins . . . 164
7.1.20 I/O Pins Showing Usage . . . 165
7.1.21 CAN / Loop Usage and Status . . . 165
7.1.22 Gauge and setting limits . . . 165
7.2 Fuel Settings . . . 166
7.2.1 Injector timing table . . . 166
7.2.2 Secondary injector timing table . . . 166
7.2.3 Staged injection . . . 167
7.2.3.1 Tuning Staged Injection . . . 169
7.2.3.2 Tuning Table-based Staged Injection . . . 169
7.2.3.3 Tuning All other Staged Injection Modes . . . 169
7.2.4 Fuel sensor settings (Flex) . . . 170
7.2.5 Over Run Fuel Cut . . . 171
7.2.6 AFR / EGO control . . . 172
7.2.6.1 Basic EGO settings . . . 173
7.2.6.2 EGO ports . . . 174
7.2.6.3 AFR/EGO Sensor Mapping . . . 174
7.2.6.4 Tuning . . . 174
Simple algorithm with narrowband sensor . . . 174
Simple Algorithm with Wideband Sensor . . . 175
PID Algorithm with Narrowband Sensor . . . 175
PID Algorithm with Wideband Sensor . . . 175
7.2.7 Narrowband EGO targets . . . 175
7.2.8 EGO Authority Table . . . 175
7.2.9 EGO delay table . . . 175
7.2.10 AFR safety system . . . 176
7.2.11 Cylinder trim tables . . . 177
7.2.12 Sequenced Batch Fire . . . 177
7.2.13 Fuel pump and pressure . . . 178
7.2.14 Fuel pressure safety . . . 180
7.3 Ignition Settings . . . 180
7.3.1 Dwell battery correction . . . 180
7.3.2 Dwell vs RPM . . . 180
7.3.3 Dwell Table . . . 180

7.3.4 Cold advance . . . 181
7.3.5 MAT based timing retard . . . 181
7.3.6 Noise filtering . . . 181
7.3.6.1 Examples . . . 185
7.3.7 Knock sensor settings . . . 186
7.3.7.1 Knock sensor settings . . . 187
7.3.7.2 Detection . . . 188
7.3.7.3 Retarding . . . 188
7.3.7.4 Recovery . . . 189
7.3.7.5 Knock input threshold . . . 189
7.3.8 Knock sensor parameters . . . 189
7.3.9 Knock window settings . . . 190
7.3.10 Knock coolant scaling . . . 190
7.3.11 Rotary settings . . . 191
7.3.11.1 FC mode . . . 191
7.3.11.2 FD mode . . . 191
7.3.11.3 RX8 mode . . . 192
7.3.11.4 3 rotor . . . 192
7.3.11.5 4 rotor . . . 192
7.3.12 Rotary split table . . . 193
7.3.13 Cylinder trim tables . . . 193
7.4 Startup / Idle . . . 194
7.4.1 Cranking / startup settings . . . 195
7.4.2 Cranking Taper Curve . . . 196
7.4.3 Priming Pulse 2 through Warmup Enrichment 2 . . . 196
7.4.4 Idle control . . . 196
7.4.4.1 On/Off Valve . . . 197
7.4.4.2 Stepper valve settings . . . 197
7.4.4.3 PWM idle valve settings . . . 197
7.4.5 Idle Cranking Duty / Steps . . . 198
7.4.6 Idle Warmup Duty / Steps . . . 198
7.4.7 Closed loop idle settings . . . 198
7.4.7.1 Tuning closed loop idle . . . 200
7.4.8 Closed loop idle target curve . . . 203
7.4.9 Closed loop idle initial values . . . 203
7.4.10 PWM Idle voltage compensation . . . 204
7.4.11 Air conditioning idle up . . . 204
7.4.12 Idle advance settings . . . 205
7.4.12.1 Idle advance tuning . . . 206
7.4.12.2 Tuning Idle Advance Engagement Settings . . . 206
7.4.12.3 Tuning Idle Advance Timing . . . 207
7.4.13 Idle RPM timing correction curve . . . 207
7.4.14 Idle VE settings . . . 207
7.5 Accel Enrich . . . 208
7.5.1 TPS WOT curve . . . 209
7.5.2 Enhanced acceleration enrichment . . . 209
7.5.2.1 Adheres to Walls Coefficient . . . 210
7.5.2.2 EAE Sucked-from-walls coefficient . . . 210
7.5.2.3 EAE Adhere-to-walls RPM correction . . . 211
7.5.2.4 EAE Sucked-from-walls RPM correction . . . 211
7.5.2.5 EAE Adhere-to-walls CLT correction . . . 211
7.5.2.6 EAE Sucked-from-walls CLT correction . . . 212
7.5.2.7 Tuning EAE . . . 212
7.5.3 X-Tau acceleration enrichment . . . 213
7.5.3.1 Puddling Factors . . . 213
7.5.3.2 Time factors . . . 213

7.5.3.3 X (Puddling) Temp Correction Factors . . . 214
7.5.3.4 Tau (Time) temp corrections . . . 214
7.5.3.5 MAP . . . 214
7.5.3.6 X-Tau tuning tips . . . 215
7.6 Boost / VVT . . . 215
7.6.1 Boost control basics . . . 215
7.6.1.1 Boost control strategies . . . 215
7.6.1.2 Boost control hardware . . . 215
7.6.2 Boost control settings . . . 217
7.6.2.1 Boost control common settings . . . 217
7.6.2.2 Closed loop specific settings . . . 218
7.6.2.3 Overboost Protection . . . 219
7.6.2.4 Other Boost Control Settings . . . 220
7.6.3 Boost control duty table . . . 220
7.6.4 Boost control target table . . . 221
7.6.5 Boost control bias duty table . . . 221
7.6.6 Boost delay . . . 221
7.6.7 Boost control vs speed / time . . . 222
7.6.8 Boost 2 control settings . . . 222
7.6.9 Wastegate dome pressure control settings . . . 223
7.6.10 Dome Bias Table . . . 225
7.6.11 Turbo anti-lag (ALS) . . . 226
7.6.12 Turbo anti-lag tables . . . 228
7.6.13 ALS roving idle fuel cut % . . . 228
7.6.14 VVT settings . . . 229
7.6.15 VVT intake and exhaust tables . . . 232
7.6.16 VVT on/off curve . . . 232
7.7 Table Choices . . . 232
7.7.0.1 Example scenarios . . . 233
7.7.0.2 Fuel blending and switching types . . . 233
7.7.0.3 Fuel settings . . . 235
Single algorithm, single table . . . 235
Single algorithm with table switching . . . 236
Single algorithm with table blending . . . 236
Dual table . . . 236
7.7.0.4 Spark / ignition blending and switching types . . . 236
Single table . . . 238
Single algorithm with table switching . . . 238
Single algorithm with table blending . . . 238
Combined tables - secondary additive table . . . 239
Secondary algorithm blended table . . . 239
7.7.0.5 AFR blending and switching types . . . 239
7.7.0.6 Boost table switching . . . 240
7.7.1 Table Switch / Dual Fuel . . . 241
7.7.2 Temperature Adjustment . . . 243
7.7.3 Pressure Adjustment . . . 244
7.7.4 Alt Injector Dead-time 2 . . . 244
7.7.5 Alt Injector small pulsewidths 2 . . . 244
7.7.6 Blend curve menus . . . 244
7.8 Advanced Engine . . . 245
7.8.1 Speed and gear sensors . . . 245
7.8.1.1 Analogue linear input . . . 246
7.8.1.2 Digital pulse input . . . 246
Wheel mounted . . . 246
Driveline mounted . . . 246
7.8.1.3 Speed value from a remote CAN device . . . 247

7.8.1.4 Pulse data from a remote CAN device . . . 247
7.8.1.5 Menu settings . . . 247
7.8.1.6 VSS dot smoothing . . . 247
7.8.1.7 VSS sampling . . . 247
7.8.1.8 VSS output . . . 247
7.8.2 Shaft Speed Sensors . . . 248
7.8.3 EGT / Thermocouple settings . . . 248
7.8.4 Generic sensor inputs . . . 250
7.8.4.1 Source . . . 250
7.8.4.2 Field Name . . . 250
7.8.4.3 Transformation . . . 251
7.8.4.4 0V, 5V value . . . 251
7.8.4.5 Lag . . . 251
7.8.4.6 CLT/MAT temp units . . . 251
7.8.4.7 Allow Input Sharing . . . 251
7.8.5 Accelerometer parameters . . . 252
7.8.5.1 Collecting calibration data . . . 252
7.8.6 Traction control settings . . . 253
7.8.6.1 Traction control settings . . . 254
7.8.6.2 Three and four wheel speed sensor traction control . . . 254
7.8.7 Traction control - perfect run VSS . . . 255
7.8.8 Traction control - perfect run RPM . . . 256
7.8.9 Traction control - External % slip input . . . 256
7.8.10 Traction control reactions . . . 257
7.8.11 Launch / 2 step / 3 step / T-Brake . . . 257
7.8.11.1 Basic launch control / flat shift settings . . . 258
7.8.11.2 Variable launch settings . . . 260
7.8.11.3 Transbrake and throttle stop . . . 260
7.8.11.4 3 step / burnout limiter . . . 262
7.8.11.5 Line lock staging . . . 263
7.8.12 Timed retard after launch . . . 263
7.8.13 Speed based launch control . . . 263
7.8.14 Launch Limit vs. MAP . . . 264
7.8.15 Sequential shift cut . . . 265
7.8.16 Nitrous system . . . 266
7.8.16.1 Wet flow nitrous . . . 266
7.8.16.2 Dry nitrous . . . 266
7.8.16.3 On/Off . . . 266
7.8.16.4 Progressive control . . . 266
7.8.16.5 Multistage and progressive nitrous . . . 267
7.8.16.6 Tuning considerations . . . 267
7.8.16.7 Nitrous settings . . . 267
7.8.17 Nitrous - time based progressive . . . 270
7.8.18 Nitrous - RPM based progressive . . . 271
7.8.19 Nitrous - VSS based progressive . . . 271
7.8.20 Water injection . . . 271
7.8.21 High Power Time Enrichment . . . 272
7.8.22 Oil Pressure . . . 273
7.8.23 Pit lane limiter . . . 274
7.8.24 Programmable on/off outputs 1 . . . 275
7.8.24.1 Output port pane . . . 275
7.8.24.2 Port settings pane . . . 275
7.8.24.3 Conditions . . . 275
Output channel . . . 275
> = < . . . 276
Threshold . . . 276

Hysteresis . . . 276
Additional condition . . . 276
7.8.25 Programmable on/off outputs 2 . . . 276
7.8.25.1 Loop conditions . . . 277
7.8.26 Generic PWM outputs . . . 277
7.8.27 Generic Closed Loop outputs . . . 278
7.9 3D Tuning Maps . . . 279
7.10 CAN bus / Testmodes . . . 279
7.10.1 CAN parameters . . . 279
7.10.1.1 Base data . . . 280
7.10.1.2 PWM polling . . . 280
7.10.1.3 Digital I/O . . . 280
7.10.1.4 PWM outputs . . . 281
7.10.1.5 Analog (ADC) inputs over CAN . . . 281
7.10.2 CAN broadcasting . . . 281
7.10.3 CAN broadcast testing . . . 282
7.10.4 CAN realtime data broadcasting . . . 282
7.10.5 CAN receiving . . . 283
7.10.6 CAN VSS, gear . . . 283
7.10.7 CAN EGO, GPS . . . 283
7.10.8 Real time clock . . . 284
7.10.9 IO Box settings . . . 285
7.10.10 Dash broadcasting . . . 285
7.10.11 Check Engine Light . . . 286
7.10.11.1 Check Engine codes . . . 288
7.10.12 Limp Mode . . . 289
7.10.13 Fallback MAP table . . . 290
7.10.14 Output test mode - Inj / spark . . . 291
7.10.14.1 Fuel Pump . . . 291
7.10.14.2 Coils . . . 291
7.10.14.3 Injectors . . . 292
7.10.15 Output test mode - I/O 1 through 3 . . . 292
7.10.16 Output test mode - idle valve . . . 295
7.10.17 Injector Sequence Testing . . . 296
7.10.18 Inj / Spk Disabling Test Mode . . . 297
7.10.19 Engine controls . . . 297
7.10.20 Special options . . . 298
7.10.21 Long term fuel trim . . . 298
7.10.22 Throttle control (DBW) . . . 299

8 Troubleshooting and Data Logging 301
8.1 Resets . . . 301
8.2 Capturing data logs . . . 301
8.2.1 Data logging with a laptop . . . 301
8.2.1.1 GPS logging with a laptop . . . 302
8.2.2 Data logging with the SD card . . . 302
8.2.3 Downloading SD data logs . . . 304
8.2.4 SD card error codes . . . 304
8.3 Data log fields . . . 305
8.3.1 Understanding data log bit fields . . . 305
8.3.2 Complete list of fields . . . 306
8.4 Tooth/Trigger Logger . . . 313
8.4.1 Lost sync numbers . . . 314
8.4.2 Using the diagnostic logger . . . 315
8.4.3 Tooth logger . . . 315
8.4.4 Trigger logger . . . 316
8.4.5 Composite logger . . . 316

8.4.6 Sync error logger . . . 317
8.4.7 Example logs . . . 317
8.4.8 Engine logger . . . 320
8.5 Communication issues . . . 320
8.5.1 USB driver software . . . 320
8.5.1.1 Windows drivers . . . 320
8.5.1.2 Linux drivers . . . 322
8.5.1.3 MacOS drivers . . . 322
8.5.2 Additional diagnostics . . . 322
8.5.2.1 Portcheck . . . 322
8.5.2.2 Loading firmware . . . 322
8.6 Loading Firmware . . . 322
8.6.1 TunerStudio firmware loader . . . 323
8.6.2 Windows firmware loader . . . 325
8.6.3 Linux firmware loader . . . 327

1 Introduction

1.1 Overview

Congratulations on your purchase of an MS3Pro! The MS3Pro is a standalone engine management system capable of running sequential fuel injection on up to 8 cylinder engines. This manual covers the MS3Pro Mini. The MS3Pro is manufactured by AMP EFI, a branch of EFI Innovations, INC.

This manual is based on the 1.6.1 and later firmware. Documentation for earlier firmware is available at MS3Pro.com.

1.1.1 Warning labels

Everything comes with warning labels. Let's get these out of the way.

All parts are sold for OFF ROAD RACE-ONLY ground-vehicle use only, or vehicles that predate any federal and state emissions control requirements. Aftermarket EFI/EMS systems are not for sale or use on pollution controlled vehicles. Alteration of emission related components constitutes tampering under the US EPA guidelines and can lead to substantial fines and penalties. Your state/district may also have specific rules restricting your tampering with your vehicle's emissions system. In short, as stated before, our official policy has to be RACE or OFF-ROAD USE-ONLY in ground based vehicles ONLY.

Race parts are inherently dangerous and may cause injury or damage if improperly modified or altered before use. EFI Innovations will not be held liable for and will not pay you for any injuries or damage caused by misuse, modification, redesign, or alternation of any of our products. EFI Innovations will not be held in any way responsible for any incidental or consequential damages including direct or indirect labor, towing, lodging, garage, repair, medical, or legal expense in any way attributable to the use of any item in our catalog or to the delay or inconvenience caused by the necessity of replacing or repairing any such item.

1.1.2 Technical support

AMP EFI technical support can be reached by email at support@ampefi.com. If you are having difficulty with a particular issue, we recommend sending a data log of the problem and a copy of your tune file with the email, as these are often helpful for resolving issues. We can also be reached by phone at 678-261-8789.

The AMP EFI team continuously maintains this manual; please use the contact address above if you find any errors or sections that need to be cleared up.

1.1.3 Copyrights

This manual is copyright 2023 AMP EFI, and contains information used by permission of Bowling & Grippo Soflex and the MS/Extra Team. You may share this file in its unaltered form or print out copies for your own personal use. For permission for other uses, contact AMP EFI at the link above.

1.2 MS3Pro components

1.2.1 MS3Pro Engine Control Unit Family

There are currently four types of ECU in the MS3Pro family.

1.2.1.1 MS3Pro Module The MS3Pro Module uses the same basic design and feature set as the original MS3Pro, but the unit ships without a case and the AMSPEAL connectors have been replaced by 0.100" headers. This board is designed to mount to a carrier board incorporating additional circuitry and connectors, such as with the MSPNP Pro, or it can also fit inside most factory ECU housing for racing classes that allow any tuning device as long as it fits within the stock ECU housing.

1.2.1.2 MS3Pro Plug and Play The MS3Pro PNP is a line of vehicle specific plug and play ECUs using the MS3Pro at their core. The exact feature set and pin-out will depend on the application. All models offer peak and hold injector drivers for the main fuel outputs

1.2.1.3 MS3Pro EVO+ The MS3Pro EVO+ is a replacement for the first generation, incorporating several design improvements.

- 10 fuel outputs for high impedance injectors
- 8 logic level spark outputs (8 high-current spark outputs for EVO+ Blue version)
- 3 high current general purpose outputs (2 of which can be used as additional fuel channels)
- 7 PWM general purpose outputs
- 1 tach output (0-12 volts)
- 4 VR inputs for crank, cam, and vehicle speed sensor input
- 3 high frequency digital inputs (2 shared with VR inputs)
- 5 digital I/O outputs that can work as frequency inputs or 5 volt logic level signals
- 6 on/off digital inputs
- 8 general purpose analog inputs
- 2 knock sensor inputs
- Onboard 4 bar MAP and 1 bar barometric pressure sensors
- Dedicated inputs for CLT, MAT, TPS, and O_2 sensors
- CAN, RS232, and USB connectivity

1.2.1.4 MS3Pro Mini The MS3Pro Mini is a compact MS3Pro variant intended for engines with eight or fewer cylinders. It can run sequential fuel and ignition on engines with up to four cylinders, or sequential fuel with wasted spark on a six or eight cylinder engines.

(NOTE: MS3Pro Mini REQUIRES Firmware version 1.6.0 or later, and TunerStudio v3.2.0 or later!)

- 8 outputs for high impedance injectors or high current general purpose use
- 4 logic level spark outputs
- 1 dedicated fuel pump relay/controller output
- 1 PWM general purpose output
- 2 VR inputs for cam and crank sensors
- 2 high frequency digital inputs
- 1 on/off digital input
- 4 general purpose analog inputs
- Onboard 1 bar barometric and 4 bar MAP pressure sensors
- Dedicated inputs for CLT, MAT, and TPS, sensors
- CAN and RS232 connectivity

1.2.2 Wiring harness

The MS3Pro Mini uses a single wiring harness with a black AMPSEAL connector.

1.2.3 Tuning cables

We include tuning cables with the ECU . The MS3Pro Mini uses a RS232 cable with an M8 circular connector. You can use our USB-2920 adapter to connect this to USB if your laptop does not have an RS232 port.

Replacement cables are available from AMP EFI resellers.

1.3 MS3Pro accessories

1.3.1 Sensors

The MS3Pro can work with most OEM temperature, MAP, MAF, and throttle position sensors, as long as you're able to get the required information on the sensor's output curve. This is covered in greater detail in section 4, Wiring. If your installation requires new sensors, AMP EFI and its resellers carry GM temperature and MAP sensors.

1.3.2 QuadSpark

The MS3Pro uses logic level spark outputs. These are compatible with be used with AMP EFI IGN-1A race coils or OEM coils with integrated ignition modules, such as GM LS series coils, Toyota 1ZZ coil on plug coils, or the pencil coils on VW / Audi 1.8 Turbo motors. They also work with most OEM ignition modules. Some manufacturers, particularly Ford, Chrysler, and BMW, frequently build the ignition module into their ECUs. If you have coils which do not have then ignition modules built in, and do not have a separate ignition module, AMP EFI manufactures their own 4 channel ignition module, the QuadSpark. This module can drive up to four ignition coils using Bosch BIP373 power transistors.

1.3.3 Ignition coils

AMP EFI sells several ignition coils as well. The IGN-1A has a built in ignition module and can be wired directly to the MS3Pro. All other coils from AMP EFI require the QuadSpark or other external ignition module when used with MS3Pro. Coils that AMP EFI offers include the IGN-1CD for capacitive discharge ignitions, the IGN-4 four tower coil pack, and the IGN-6 six tower coil pack.

1.3.4 CAN-EGT thermocouple interface

Thermocouples put out a faint voltage that requires an amplifier circuit to produce an accurate reading. The CAN-EGT includes amplifiers for up to eight of these sensors, which can be used to measure exhaust gas or cylinder head temperature. The CAN-EGT module then transmits the temperature data to the MS3Pro over a CAN network to avoid the inaccuracies of using a digital to analog conversion circuit. The MS3Pro can use this both for data logging and for safety shutdown features if the exhaust gas temperatures go too high under load.

The CAN-EGT module can also gather data from the digital output on up to eight separate Innovate Motorsports LC-1, LM-2, or MTX-L wideband controllers and transmit this information to the MS3Pro without using any of the MS3Pro's analog inputs. Using this allows the MS3Pro to run separate wideband O_2 correction for all cylinders.

1.3.5 MicroSquirt

The MicroSquirt was originally intended as a fuel and ignition controller for powersports applications, but its CAN network capabilities and easily reprogrammed firmware allow it to be configured for other functions. Current release firmware allows it to be used as a transmission controller for many applications. The CAN bus allows you to view the MicroSquirt data through TunerStudio as if the MS3Pro and MicroSquirt are a single control unit with a single point of tuning. AMP EFI offers plug in harnesses for select transmissions.

Another use for the MicroSquirt is to run the IO Box firmware. This turns the MicroSquirt into an I/O expansion device, adding additional analog inputs as well as on/off or PWM outputs. The MS3Pro can accommodate up to three MicroSquirts running IO Box firmware, daisy chained together. The IO Box firmware adds the following inputs to the MS3Pro:

- 7 0-5V analog inputs
- 2 VR inputs

- 3 switched inputs
- 7 PWM or on/off outputs

1.3.6 HSD-4 High Side Driver module

While the standard MS3Pro outputs supply ground, a few types of solenoid ground through their housing or otherwise require the ECU to switch 12 volts. Examples include BMW's Double VANOS solenoids and the transmission shift solenoids on a Toyota A340. The HSD-4 accepts up to four ground triggered signals and uses this to supply high speed switched 12 volt signals. Maximum current is 8 amps per channel.

1.3.7 AXM AMP eXpansion Modules

The AXM add on modules allow for different sorts of input and output conditioning. Current products in this series include:

- AXM-100 high voltage tach adapter
  - This amplifies the 0-12 volt tach signal to control tachometers originally driven off the negative terminal of the coil. If you find your original tachometer does not work with the MS3Pro's output, this module can fix the problem.
- AXM-110 VR conditioner
  - Converts a VR signal, which looks like an AC sine wave, to a 0 to 5 volt square wave input. You can use this module with additional cam position sensors, vehicle speed sensors, and similar devices if you are out of VR inputs on the MS3Pro.
- AXM-120 high voltage isolator
  - Protects the MS3Pro switched inputs from high voltage spikes. Use this to read RPM off the negative terminal of an ignition coil, or when connecting a solenoid to a switched input (such as using an input to read if a line lock or trans brake solenoid is energized).

1.3.8 DD-EFI Digital Dashboard

The CAN-based digital dashboards from DD-EFI are custom made specifically to work with the full line of MS3Pro ECUs:

https://dd-efi.com/

1.4 Tools

As a general rule, you'll want to have the following tools and supplies on hand for a successful MS3Pro installation:

- Multimeter
- Laptop (the MS3Pro software supports Windows, Linux, or Mac OS)

- Screwdrivers
- Soldering iron, solder, and heat shrink tubing (if joining wires by soldering) or crimp connectors and crimp tool
- Wire cutters
- Wire stripping tool
- Timing light
- Wideband O_2 sensor and controller (not needed for permanent installation, but useful for tuning)
- Electrical tape
- Vacuum tubing

2 Installing the software

The MS3Pro utilizes the following programs, available from https://www.ampefi.com/downloads/:

- TunerStudio. This is the main program you'll use to connect to the MS3Pro. It is used to configure the MS3Pro's settings, tune the MS3Pro, record data logs on the laptop, and retrieve internal data logs.
- MegaLogViewer. This is an analysis program which can play back data logs and display the data in both line graph and scatter plot format. It can also modify tune files based on information recorded in data logs.
- Firmware loading utility. This is used to update firmware or for certain diagnostics.

Download the installers from the above link and follow the installation prompts, and the software should install in just a couple minutes.

The tuning software can run on most versions of Windows, as well as Linux and Mac OS.

2.1 Registering TunerStudio

Both TunerStudio and MegaLogViewer have registered versions. Purchases of new MSPNPPro, MS3Pro EVO+, and MS3Pro Mini ECUs include a registered copy of TunerStudio FOR THE ORIGINAL PURCHASER **ONLY**. To unlock the registered version, use the serial number code on your MS3Pro. This appears on a label on the underside of the box as well as on a tag on the back of the MS3Pro. You will enter the serial number at https://www.efianalytics.com/register/registerCoupon.jsp as a "Coupon code provided by dealer" to obtain a registration key.

2.2 TunerStudio

2.2.1 Start screen

Once TunerStudio opens, you will be presented with the start screen as shown below.

At this point, you'd normally use the screen to create a project or open one; this will set up TunerStudio with the configuration it needs to communicate with your ECU or allow you to view tune files offline. You can create or

open a new project under the **file menu (1)**, or use this menu to open a new tune for offline viewing. There are also command links to **create a new project (2)** or **open a recent project (3)**.

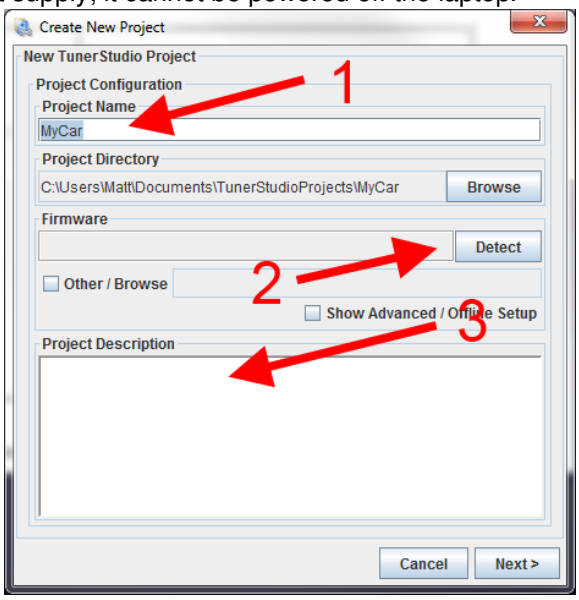
The Help and Support button brings up links to documentation and support forums for the MS3Pro as well as other systems in the MegaSquirt line.

Note that if you open a tune file from this screen, it will go into a temporary project and TunerStudio will not connect to the MS3Pro until you create or open a project for connecting to the MS3Pro.

2.2.2 Creating a project

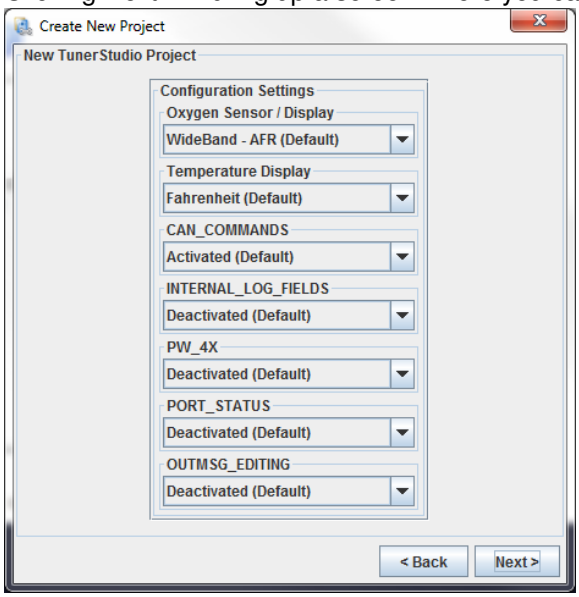
For a how-to video, see here: https://www.youtube.com/watch?v=aXDF5zcADU0

Selecting "New Project" from the File menu opens a wizard to create a new project. It will bring up a screen like the one below. At this point, you will want to have the MS3Pro powered up (see section 4 for wiring details) and connected to the laptop with either the USB or RS232 cable. The MS3Pro will need to be powered through the 12 volt supply; it cannot be powered off the laptop.



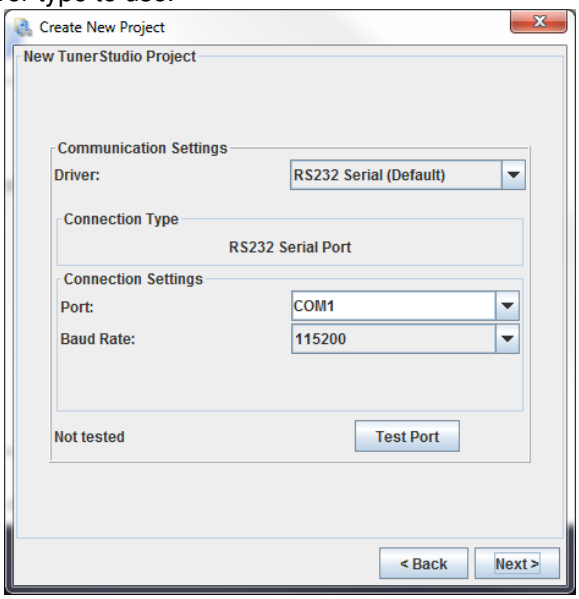
You can enter a name for the project in (1). Next, click the Detect button (2). TunerStudio will detect what firmware is on your MS3Pro. If TunerStudio does not have a definition file, it will prompt you to download one from the Internet, which TunerStudio will handle automatically for any standard release version of the MS3Pro firmware. You can also enter notes about this project in (3).

Clicking Next will bring up a screen where you can select project specific settings.



The exact settings shown will depend on your firmware. These settings are covered in more detail in section 5.1.

Clicking Next will bring up a communications test screen. This allows you to select what port is used and what driver type to use.

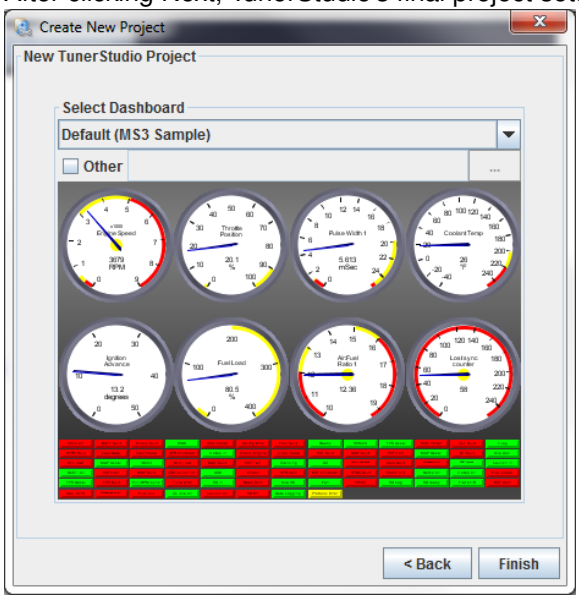


Driver options include:

1. **RS232 serial.** Use this for connecting to the MS3Pro's RS232 port without a USB adapter. This setting will also work with third party USB adapters that behave like a virtual COM port. The unregistered version of TunerStudio will use this with the MS3Pro's USB connection and AMP EFI's USB adapters as well.
2. **Bluetooth Direct.** Use when connecting a built in Bluetooth connection on the laptop to a Bluetooth to serial adapter, such as the one sold by EFI Analytics.
3. **FTDI - D2XX Driver.** This works with the MS3Pro's internal USB port, or AMP EFI's USB-2920 adapter on the MS3Pro's RS232 port. Provides somewhat faster and more reliable communication than using this adapter in RS232 Serial mode.
4. **TCP/IP - WiFi Driver.** Use with WiFi to serial adapters.

Specify which driver you will use at the top of this screen. MS3Pro normally communicates at a 112500 baud rate.

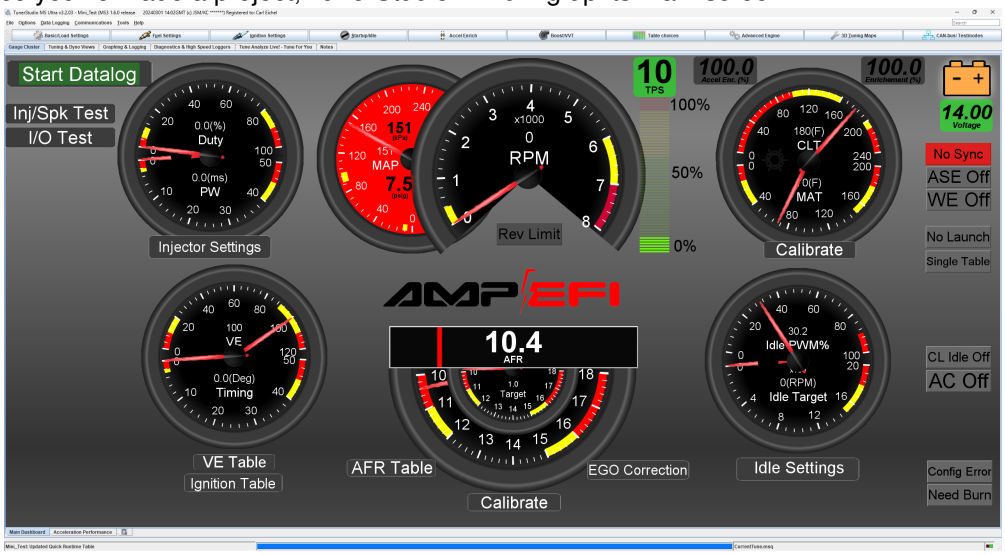
After clicking Next, TunerStudio's final project setup screen lets you select a gauge display.



This display will show the gauges and indicators cycling through their range of readings. There are several ready to run gauge options that you can select from the drop down menu at the top. Checking the “other” box allows you to load custom dash files. You can edit any gauge cluster once the project is opened, or load an alternate gauge cluster. Click “Finish” to confirm your gauge selection and launch the new project.

2.2.3 TunerStudio main screen

Once you’ve made a project, TunerStudio will bring up its main screen.



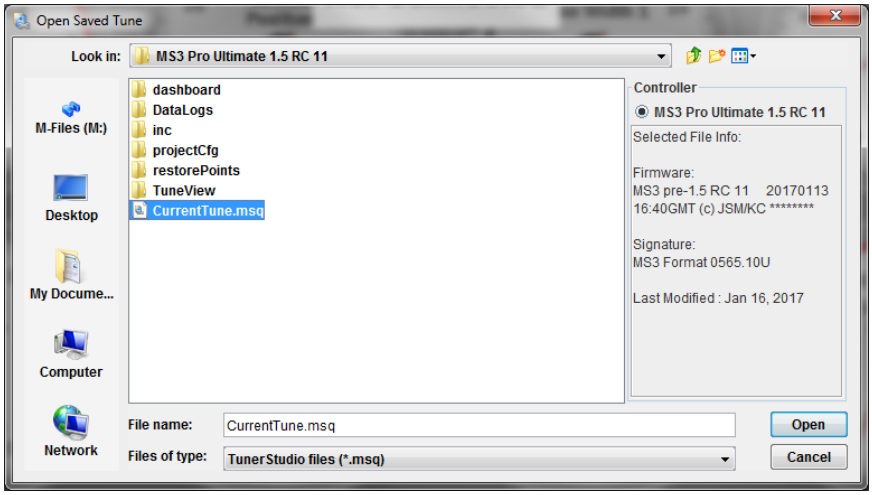
The screen is divided into several sections.

- The title bar at the top displays the current TunerStudio version, the project name, and when connected to the MS3Pro, the firmware version.
- The menu bar at the top deals mostly with TunerStudio functions, such as loading and saving files. They also allow for calibrating the sensors.
- The large row of buttons bring up menus to access the settings in the ECU.
- The row of tabs below the buttons lets you select four different screens:
  - Gauge Cluster: Displays basic information about the sensor readings, ECU calculations, and status.
  - Diagnostics & High Speed Loggers: These are used for various specialized troubleshooting functions; see section 8 for more detail.
  - Tune Analyze Live! Tune For You: This allows TunerStudio to recalculate new fuel tables based on oxygen sensor feedback.
  - Notes: A text editor for taking notes.
- You can change any gauge or indicator in the cluster by right-clicking on it and swapping it to a different function.
- There is a second set of tabs at the bottom of the gauge cluster that allows you to set up multiple, separate gauge clusters. Click on the rectangular icon to create a new cluster.

2.2.4 Loading and saving tunes

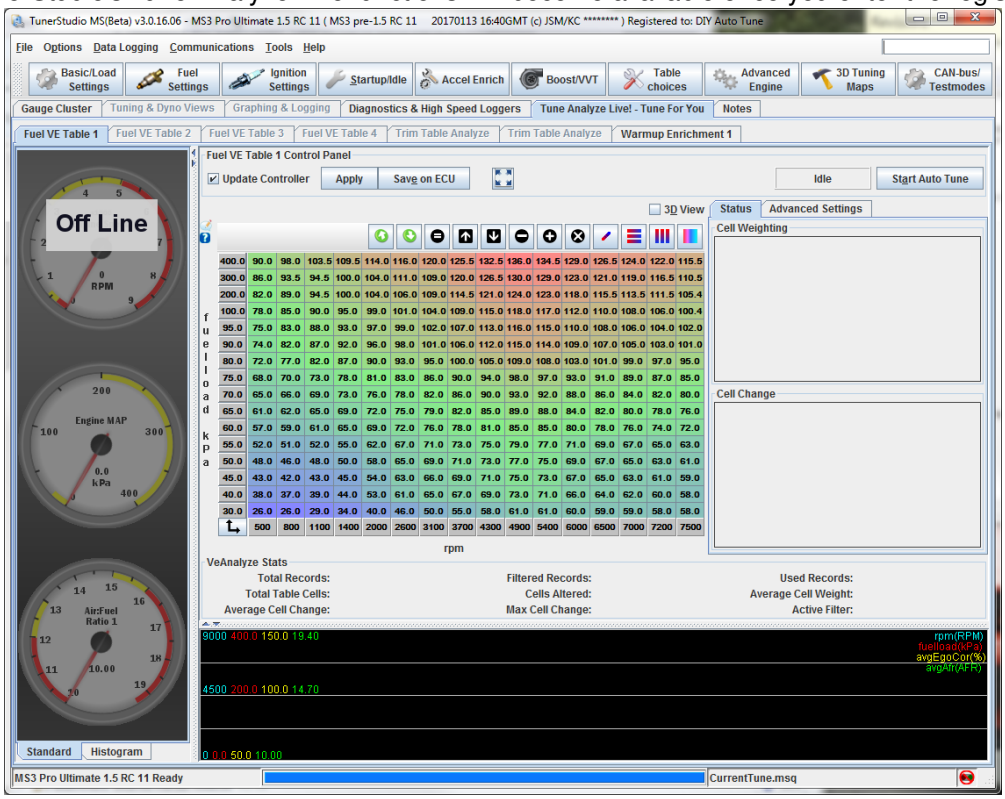
The File -> Save Tune As command will fetch the tune from the ECU and save it to a file. If you have just connected to an MS3Pro and do not have a copy of the tune, we strongly recommend saving a copy as soon as you connect. This will allow you to revert to the original tune if you run into difficulties after tuning.

The File -> Load Tune (MSQ) command will load a tune from a file into the ECU if you have the ECU connected and TunerStudio online. If you open a tune without the ECU connected, TunerStudio will display the tune so you can edit it.



2.3 Tune Analyze Live

TunerStudio's Tune Analyze Live functions will become available once you enter the registration code.



The Tune Analyze Live function requires a wideband O2 sensor installed and connected to the MS3Pro. It can tune several different tables:

- VE (fueling) tables, including secondary and switched tables
- Trim tables (requires wideband to be installed in individual header pipes).
- Warmup enrichment

TunerStudio's Help file has a detailed write up on how to use this feature.

3 MS3Pro hardware

3.1 Overview

The MS3Pro Mini is a full standalone ECU with everything you need to run sequential fuel injection on up to 8 cylinders. This ECU is based on the previous MegaSquirt-III design and runs similar code, but optimized for small engine use. Here are the MS3Pro Mini's specifications.

Hardware features:

- Freescale MC9S12X 16 bit, 50 MHz asymmetrical dual core processor
- 8 saturated injector / high current output drivers
- 4 logic level ignition outputs
- 1 medium current (3 amp), high frequency general purpose output
- 2 VR sensing differential inputs - supports VR, Hall effect, and optical input
- 12 volt tach output
- Fuel pump output
- Onboard 1 bar barometric and 4 bar manifold absolute pressure sensors
- 7 analog inputs (3 dedicated sensor inputs, 4 general purpose)
- 2 general purpose on/off or frequency inputs
- 1 digital I/O logic input
- RS232 and CAN communication
- Onboard microSD card for internal data logging
- Internal ECU temperature sensor
- Real time clock
- Temperature range: -30 to +80 degrees C (-40 to +85 without battery for real time clock)
- Minimum supply voltage: 6 volts
- Maximum supply voltage: 18.0 volts before triggering over-voltage protection circuits

Software features:

- Supports speed density, alpha-N, or MAF based fuel and spark tables with option to blend between any two load types
- 1 microsecond injector pulse width resolution
- Tables for nonlinear injector behavior at small pulse widths
- Allows blending multiple load types, including specialized mode for independent throttle bodies
- Supports a wide variety of OEM cam and crank position sensors
- Individual cylinder trim tables for fuel and ignition
- Accelerator pump or model based acceleration enrichment
- Closed or open loop idle speed control
- Closed or open loop boost control with gear or speed based tuning options
- CO_2 and dome pressure based boost control

- On/off or closed loop continuously variable valve timing control - supports up to 4 channels
- On/off or progressive nitrous control
- Traction control
- Rally anti-lag
- Rotary support - can run up to 4 rotor engines with separate leading and trailing spark tables
- Staged injection
- Table switching
- 16 x 16 fuel and spark tables - can be reconfigured to function as 30 x 16 or 16 x 30 by switching tables based on RPM or load
- 2 or 3 step rev limiter with no-lift shifting
- Wideband AFR target tables
- Flex fuel sensor input
- Allows re-purposing injector or ignition outputs as general purpose outputs, or high current outputs as injector drivers for 12 cylinder sequential applications
- Real time barometric correction
- A/C and cooling fan control with idle compensation
- Safety shutdown based on AFR or EGT input
- Motorcycle shift cut

Most of how to set these inputs up is covered in the wiring guide, but we have in depth information if you need to know everything about the circuits' capabilities. (Most users can skip straight to Section 4 and start wiring the unit up, but for those who want to know the full details, here they are!)

3.2 Inputs

3.2.1 Engine speed

The MS3Pro uses a Maxim MAX9926 adaptive differential conditioner for engine RPM. This circuit can accept input from variable reluctor, Hall effect, or optical sensors, and the input's absolute maximum rating is 400 volts. The MAX9926 input circuit cannot be triggered off the negative terminal of an ignition coil - this may damage the circuit. For fuel only installations, you will want to use a 12 volt square wave tach signal, like the one provided by most aftermarket ignition modules.

Some spark modes, discussed in the ignition section of the manual, use the digital I/O channels as well. These come into play for systems that either use very high frequency signals (such as the Nissan or Optispark optical systems), require more than two sensors for engine position (such as the Audi "tri-tach" system), or use more than one channel of variable valve timing.

3.2.2 Temperature inputs

The IAT and CLT inputs are designed to accept input from a negative temperature coefficient thermistor. Simply put, this is a sensor that is a type of resistor that changes resistance with temperature, and the resistance goes down as the temperature goes up. The default settings are for the sort of GM sensors that are something of an industry standard in the EFI world, but you can calibrate it for other sensor types through TunerStudio. The voltage on the sensor will decrease as the temperature goes up.

TunerStudio will ask for a bias resistor value when calibrating temperature sensors. This refers to a resistor inside the MS3Pro and should be set to 2490 unless directions specifically say to use a different value.

A third temperature input monitors ECU internal temperature. This sensor cannot detect individual hot spots, but if you are installing the MS3Pro under the hood or in other applications where you are concerned about exceeding its internal temperature limit, this will let you log and view how hot the MS3Pro is getting.

3.2.3 Throttle position

This input can accept signals in the 0 to 5 volt range and will is compatible with almost any potentiometer type TPS. If your engine does not have a suitable TPS, you should connect this input to the sensor return. Leaving this input free disconnected can give a false high TPS reading and cause the MS3Pro to trigger flood clear mode when starting.

3.2.4 O_2 sensor input

This input is also 0 to 5 volts. It can be used with either narrow band O_2 sensors or accept an analog output from a wideband O_2 sensor controller. It will not interface directly with a wideband O_2 sensor, but it is compatible with most external wideband controllers, such as Innovate, AEM, Zeitronix, PLX, and 14point7.

The general purpose analog inputs may also be configured as additional O_2 input channels and work in the same way when set up as O_2 inputs.

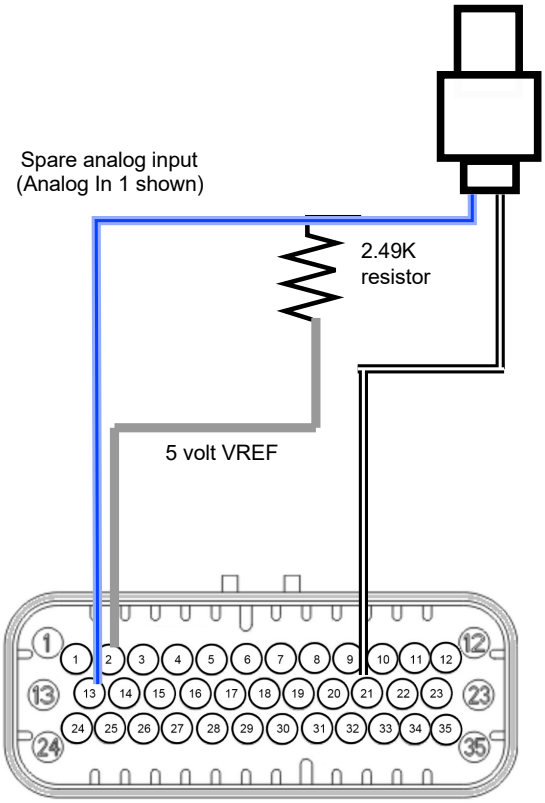
Some devices allow you to bring the O_2 sensor input over the CAN connection. If using a CAN-based wideband or not running any O_2 sensors, you can repurpose the O_2 input as a generic analog input to connect other devices.The MS3Pro Mini does not specifically designate any of its connections as O_2 inputs. You can wire the O_2 sensor to Analog In 1, 2, 3, or 4.

3.2.5 MAP sensor input

The MS3Pro Mini has a built in 4 bar MAP sensor. Connect the MAP sensor hose barb to the intake manifold downstream of the throttle body using a 1/8” ID hose. If you need to calibrate this sensor, it uses MPXH6400 calibration. If you prefer an external MAP sensor, you can use any unused analog input pin for MAP input.

3.2.6 General purpose analog inputs

These inputs can accept a 0 to 5 volt signal from a variety of sensors, which can be used for barometric correction, O_2 sensors, voltage based MAF sensors, or simply for data logging additional sensors such as oil or fuel pressure. If using these inputs with an external two wire temperature sensor, you will need to add a 2.49K resistor as shown below.



3.2.7 Knock input

The MS3Pro Mini does not have a built in knock sensor interface. Connecting a knock sensor requires using an external knock sensor conditioner. When using an external knock conditioner, the MS3Pro can accept an on/off input that switches state to report knock, or a 0-5 volt signal where the voltage increases with knock intensity.

3.2.8 Digital input channels

These use an on/off switch to ground to trigger the digital switched inputs. There are also two digital frequency inputs capable of receiving high frequency signals, which can be used for speed inputs, frequency based MAF or MAP sensors, or a number of other functions.

Digital frequency inputs have a cutoff frequency of 160 kHz. The cutoff frequency on digital switched inputs is 16 kHz.

3.2.9 Barometric pressure input

The MS3Pro Mini has a built in MPX4115 barometric pressure sensor. The “vent” is designed to allow pressure in the case to match the outside air pressure while maintaining a leak-proof seal.

3.3 Outputs

3.3.1 Injector outputs

The injector outputs are limited to 5 amps per channel, with thermal shutdown protection. These can drive one or two high impedance injectors on each output, but will not directly run low impedance injectors. We recommend

using an external peak and hold injector driver box with low impedance injectors. Unused injector outputs may be used for on/off or pulsed outputs.

3.3.2 Ignition outputs

The MS3Pro uses 5 volt, logic level ignition outputs with a high side driver circuit. These will drive most factory ignition modules and coils with built in ignitors. For use with coils that lack built in ignition modules, we recommend either the MS3Pro EVO+ HD (with built in high-current ignitors) or the QuadSpark external ignition module. Note that these outputs will not trigger an MSD ignition box or a Ford TFI.

In order to drive a Ford TFI, MSD CDI, or other similar ignition that requires a 12v square wave input with an MS3Pro Mini, please use the AXM-130 dual-channel spark output adapter that is available here: AXM-130 Driver Do not connect the MS3Pro ignition outputs directly to the negative terminal of an ignition coil; this can result in damage to the MS3Pro. Maximum current is 200 mA per ignition output channel, and pushing the output beyond this amount of current can damage the output channel.

The only exception to this is the EVO+ HD, which has built-in high-current ignition drivers and therefore the outputs CAN be connected directly to the negative side(s) of the ignition coil(s).

Unused ignition outputs may be used as on/off outputs, although their low current capacity means we do not recommend using them for anything other than logic level devices or LED lights. Do not use these outputs to drive relays directly or use them for anything that needs a ground for triggering.

3.3.3 High current outputs

You can use any unused injector outputs as general purpose ground switched outputs. These can be run in an on/off fashion or use pulse width modulation. Maximum frequency is 1021 Hz. Flyback voltage spikes are clamped at 36 volts to help the valve close rapidly. They typically work well with boost control solenoids and nitrous solenoids, which cycle fully off and fully on.

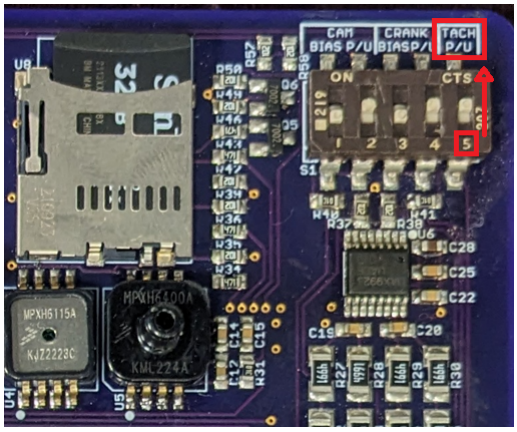
3.3.4 PWM medium current outputs

These are ground switched outputs, maximum current 3 amps. They are optimized for valves which use pulse width modulation to hold them in an intermediate position instead of pulsing on and off. Flyback spikes are clamped at battery voltage, which causes the valve to close more slowly. Common valves that would use these outputs include idle air control valves and variable valve timing oil control valves. They can also be used as an on/off signal. The fuel pump uses a similar circuit, although it does not use PWM. Maximum frequency on these outputs is 1021 Hz.

Note: Any device connected to these outputs must be set up so that it is powered off when the ECU is shut down. Relays or solenoids connected to these outputs can back feed power into the MS3Pro if the relay coil or solenoid winding receives power with the key on. Wire any device connected to these outputs to switched 12 volt power.

3.3.5 Tach output

Due to the limited pin count, the Mini has no dedicated 12 volt tach output. Instead, there is a set of internal DIP switches, the fifth of which can be switched 'ON' to provide a pull-up resistor to 12 volts on the output of the eighth fuel injector. Thus for applications using FEWER than all 8 injector outputs (and with this pull-up enabled), the INJ H output can be repurposed as a 12 volt tachometer output. Injector Out H should then be selected in the Tacho Output configuration in TunerStudio.



3.3.6 Stepper motor control output

Due to its small pin count, the MS3Pro Mini does not have a stepper IAC valve output. On engines originally equipped with a stepper IAC valve, we recommend retrofitting a two wire Bosch style valve equipped with hose barb fittings, such as a Bosch 0 280 140 551, or swapping to a throttle body with a built in 2 wire IAC valve.

3.4 Communications lines

3.4.1 RS232

RS232 is an industrial strength serial connection. RS232 is more noise resistant than USB, and we recommend using this connection if your laptop supports this. Do not attempt to plug the RS232 connection into any sort of 15 or 25 pin connector - these are usually for VGA displays or joysticks. Some third party devices such as Bluetooth adapters also use the RS232 connection.

3.4.2 USB

If you need to connect the MS3Pro Mini to a USB port, use our USB-2920 adapter.

3.4.3 CANbus

The CANbus wires are an automotive network that allow the MS3Pro to communicate with other CAN enabled modules. Note that not all CAN enabled devices will communicate with the MS3Pro; CAN is a type of hardware, and in addition to having the required hardware, the device you are attempting to connect to the MS3Pro must also be programmed to understand the type of messages the MS3Pro sends. Devices currently or soon on the market that can be connected to the MS3Pro include:

- DD-EFI Digital Dashboard: https://dd-efi.com/
- MicroSquirt transmission controller
- AIMSport MXL, Racepak IQ3S, Race Technology Dash2Pro, and other display dashes
- Third party CAN enabled wideband O_2 sensor controllers

3.5 Maintenance inside the case

3.5.1 Changing the fuses

All current MS3Pro ECUs use permanently installed Polyfuses, which increase their internal resistance as current flow (and thus their temperature) rises. These are not user-serviceable as they should never need replacing; simply fix any shorts or over-current issues, allow the polyfuses a minute to cool back down, and then the ECU will function properly again.

3.5.2 Replacing the SD card

The SD card stores the MS3Pro's data logs. This can be swapped out for a larger card if needed. Micro SD cards are supported up to 16GB micro SDHC, Class 10, formatted as FAT32. The card simply slides in until the retaining pin in the side of the SD holder moves out.

3.5.3 Changing the real time clock battery

The MS3Pro real time clock uses a lithium battery for maintaining its time and other data when the ECU is powered down. The battery slips in with the positive (flat) side up. The MS3Pro can use any of the following batteries. Part numbers starting with BR can be used for -30 to +80 °C (-22 to 176 °F) and we recommend these for their extended temperature range. CR series batteries fit and put out sufficient voltage, but they have a smaller temperature range of -30 to + 60 °C (-22 to 140 °F) and we do not recommend them.

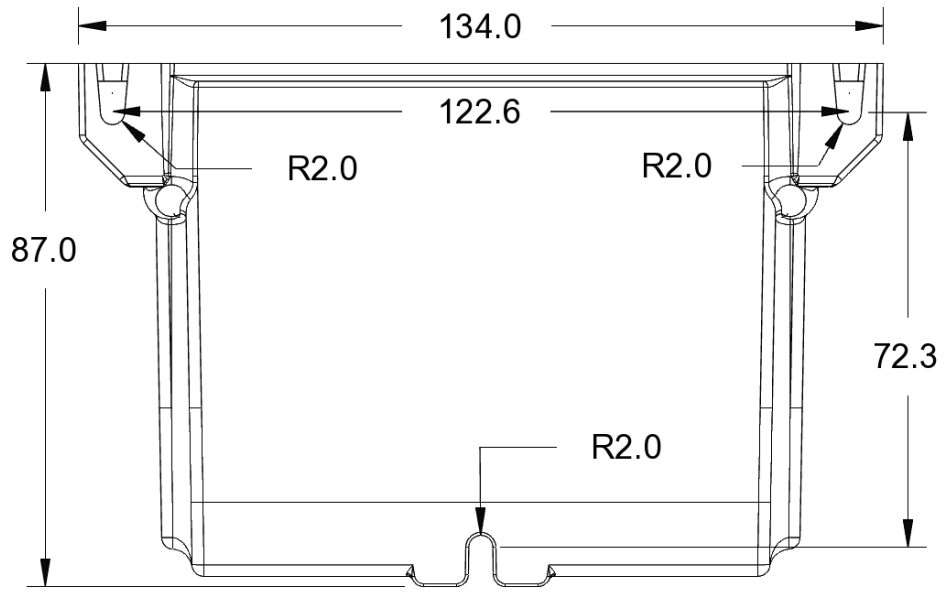
- BR1216
- BR1220
- BR1225

3.5.4 Water seal specifications

The case uses a rubber gasket and a sealed vent to protect against water intrusion.

3.5.5 Mounting footprint

Here is a dimensional drawing of the Mini case, in millimeters, for planning and electrical panel:



4 Wiring

First, some general advice about wiring:

1. We use automotive grade, high temperature TXL wiring in our harness (it's rated for up to 125℃), but you should still do your best to keep it away from heat sources. Also take any potential sources of mechanical damage into account - flip up headlight mechanisms, the side to side movements of the engine on its mounts, etc. Usually the biggest source of electrical noise isn't RFI or other radio noise - it's poor grounding or having noise injected into the power wires by another device. The MS3Pro works best when grounded to the battery negative terminal for maximum noise resistance. We do not recommend grounding the MS3Pro to the chassis or sheet metal under any circumstances. You may either crimp all ground wires to a single lug, or crimp the ground wires on the gray connector to one lug and the single ground on the white one to a separate lug. The sensors need to be grounded to the sensor return wire, which grounds them to the MS3Pro. Grounding sensors to the engine or body will create jitter from ground offsets.

2. You will also want to make sure the 12 volt power wire is clean. We recommend using one power wire from the battery for the MS3Pro EFI main relay, and a separate power wire to power the relay for the injectors, fuel pump, and ignition system for maximum noise reduction. Fuel pumps and ignition coils in particular tend to inject significant amounts of noise into their 12 volt supply wire. Also, installing a capacitor (the points condenser out of an old fashioned points distributor will work) on the supply line to the ignition coils can reduce the amount of noise they feed into the 12 volt supply.

3. You can start off a debate that goes on for hundreds of posts on an Internet forum over whether it's better to solder or crimp wires to join them. Either one can work very well if done right, and either one can be unreliable if done poorly. It's our experience that getting a good solder joint is more reliant on good technique, which can still be done with cheap tooling, while a good crimp joint is more reliant on good tooling. Either way, the joint needs to be sealed (such as with heat shrink tubing) to keep out moisture and contaminants. Heat shrink tubing also provides some measure of protection against vibration.

4. We designed the MS3Pro's power supply to run on a wide range of supply voltage. It can operate on voltages as low as 6 volts to cover voltage dips during cranking, although if your voltage drops that far, we're not sure your battery is going to crank it over. The MS3Pro has an input voltage limiting circuit that clamps input voltage at 22 volts, and feeding it 22 volts for any sustained length of time will blow the input voltage protection fuse.

4.1 ECU Connectors

AMPSEAL connector pinout

Pin	Function	Wire color	Stripe / shielding
1	12V+ switched power in	Red	None
2	+5V VREF out	Gray	None
3	PWM out 1 / Idle	Green	None
4	Fuel pump out	Purple	None
5	Injector out A	White	None
6	Injector out B	White	Orange
7	Injector out C	White	Green
8	Injector out D	White	Pink
9	Injector out E	White	Red
10	Injector out F	White	Brown
11	Injector out G	White	Blue
12	Injector out H	White	Purple
13	Analog in 1	Light blue	Blue
14	Analog in 2	Light blue	Red
15	Analog in 3	Light blue	Purple
16	Analog in 4	Light blue	Pink
17	Spark out A	Yellow	None
18	Spark out B	Yellow	Orange
19	Spark out C	Yellow	Green
20	Spark out D	Yellow	Pink
21	Sensor ground	Black	White
22	Power ground	Black	None
23	Power ground	Black	None
24	CAN Low	Tan	Red
25	CAN High	Tan	None
26	CLT in	Yellow	None
27	IAT in	Orange	None
28	TPS in	Light blue	None
29	Digital switched in 1 / Flex fuel	Gray	Orange
30	CKP+	White	Shielded
31	CKP-	Black	Shielded
32	CMP+	Yellow	Shielded
33	CMP-	Red	Shielded
34	Digital frequency in 1 (Hall)	Purple	White
35	Digital frequency in 2 (Hall)	Purple	Red

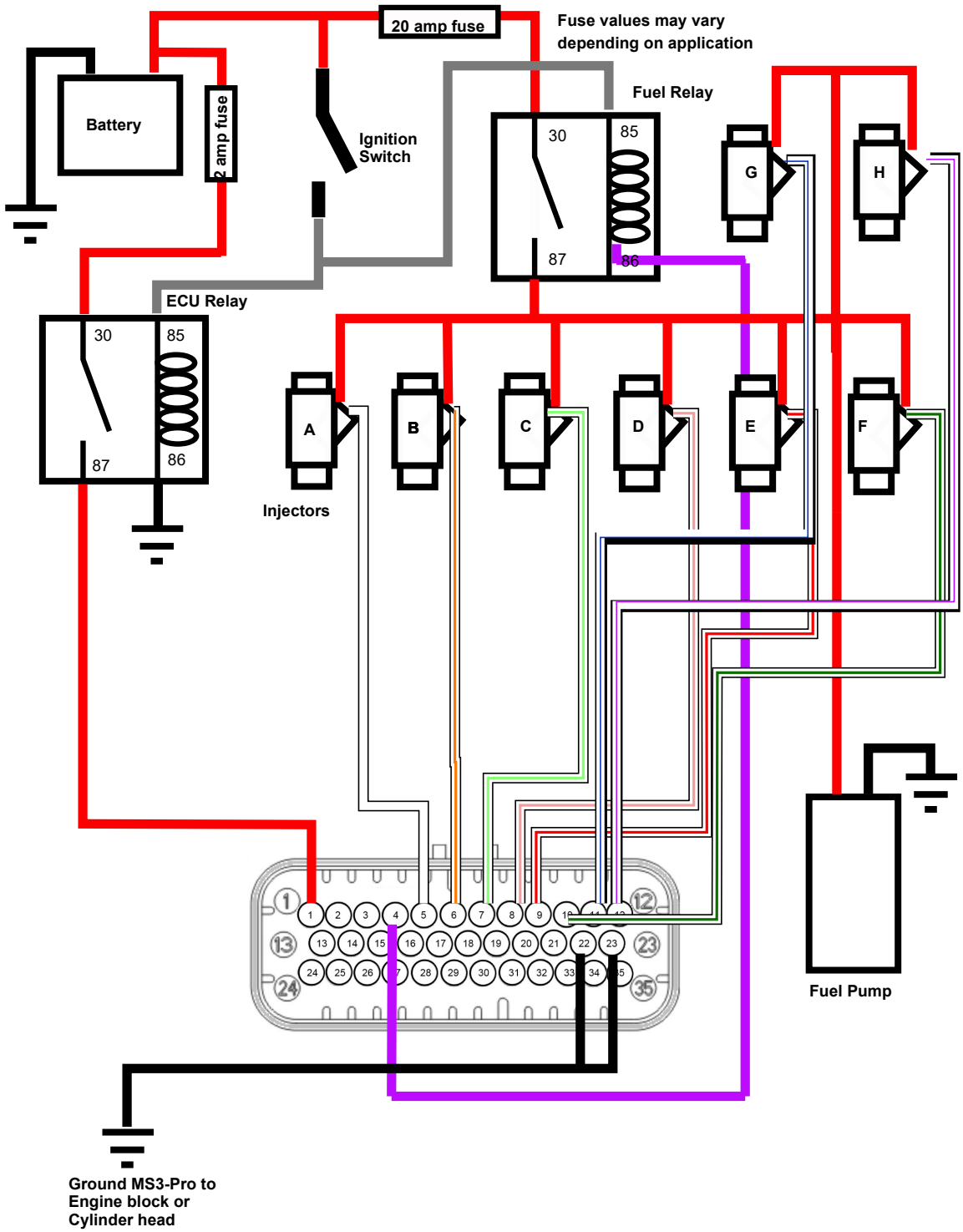
RS232 M8 serial cable

The pin names are marked on the connector.

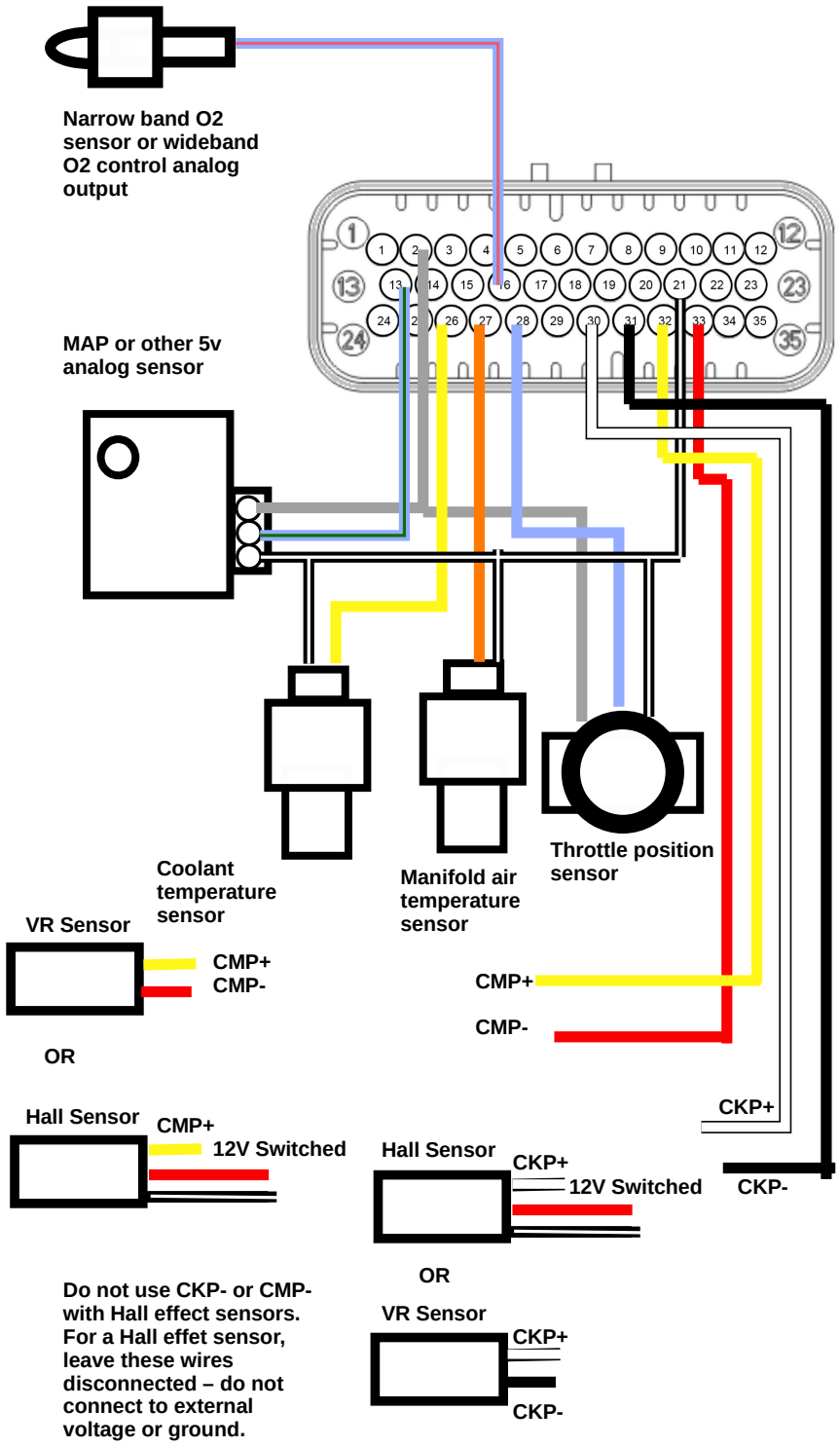
M8 pin	M8 pin
1	5 volt reference
2	Data
3	Data
4	Ground

4.2 Wiring diagram

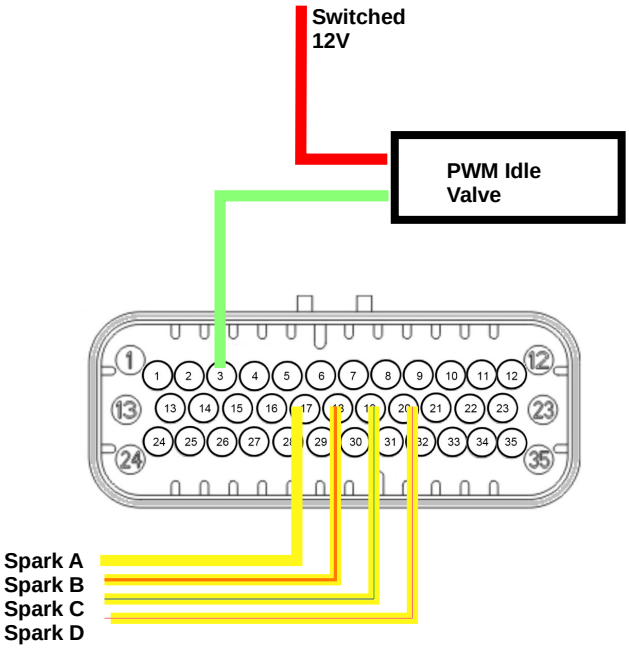
These show the basic connections - power, ground, fuel, ignition, idle control, and basic sensors. Other optional inputs and outputs are covered in their respective sections.



MS3-Pro Mini Wiring: Power, Ground, Fuel



MS3-Pro Mini Wiring: Basic Sensors



MS3-Pro Mini Wiring: Ignition and Idle

4.3 Engine position sensors

The MS3's CKP (crankshaft position) and CMP (camshaft position) inputs use differential signal conditioners that can accept input from VR (variable reluctor, also called a inductive pick-up or magnetic pick-up, although the latter is somewhat confusing because Hall effect sensors are also magnetic), Hall effect, or optical sensors. We use CKP to mean the primary trigger. In most spark modes, this is the signal the ECU relies on for most of its RPM and timing calculations. Although the abbreviation implies the sensor should be on the crankshaft, in some cases, it's actually in a distributor or on the camshaft.

The CMP signal is a secondary trigger. This is almost always on the camshaft or distributor, although there are one or two setups that put this on the crank. The CMP sensor generally provides supplemental information used to interpret the crank sensor readings, and its accuracy is usually not as critical. Here are the rules we go by to identify which sensor is the CKP and which is the CMP.

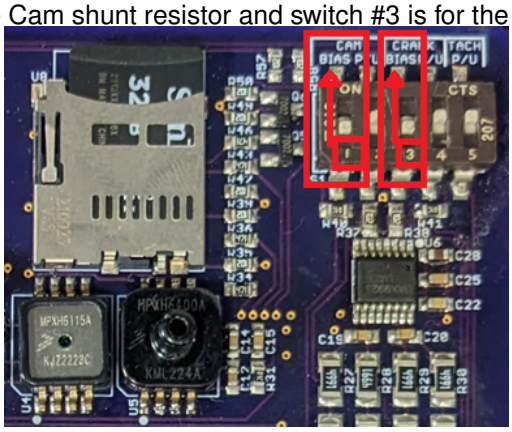
- If one sensor spins at crankshaft speed and the other spins at cam speed, the one that turns at crank speed is the CKP and the one at cam speed is the CMP.
- On a GM Optispark and most Nissan optical cam sensors, the MS3Pro normally uses a ring with one slot per cylinder for primary engine operation. There is a ring of 360 slits that is used for cranking and operation at very low RPM. Consequently, the one slot per cylinder sensor is the CKP and the 360 slot ring is CMP. (Note that these use Digital Frequency In 2 for the CMP sensor, as the regular CMP input is too heavily filtered to correctly handle this signal.)
- On almost all other setups with two sensors reading wheels spinning at cam speed, the sensor that reads the wheel with the lowest number of teeth is CMP and the one with the most teeth is CKP.
- Specific sensor setups that do not fall into any of these categories are covered in the ignition set-up section of the manual.

Note that the conditioner circuit will invert the signal. We have left the settings unchanged for backward compatibility with the original MegaSquirt line. So if you set the ignition input capture to "rising edge", that is the signal at the processor - it will actually be activated when the signal coming in to the MS3Pro transitions from high to low.

4.3.1 Variable reluctor sensors

Variable reluctor (VR) sensors have a magnet and a coil of wire which generates an A/C signal with a voltage that increases with RPM. VR sensors do not have power wires as they generate their own voltage. On a VR sensor, you will wire their two terminals to both of the CKP or CMP wires, as appropriate. Don't worry too much about getting it backwards - you can set the active edge in software.

If you are using a VR sensor on a high tooth count wheel with missing teeth (Ford 36-1 and Bosch 60-2 are two common examples) and lose sync as the RPM comes up, you may need to add a 'shunt' resistor in parallel across the VR+ and VR- signals. The Mini has internal DIP switches on the circuit board (labeled as BIAS) that can be switched 'ON' in order to enable built-in 10K shut resistors on both the Crank and Cam inputs. DIP switch #1 is for the Cam shunt resistor and switch #3 is for the Crank shunt resistor.



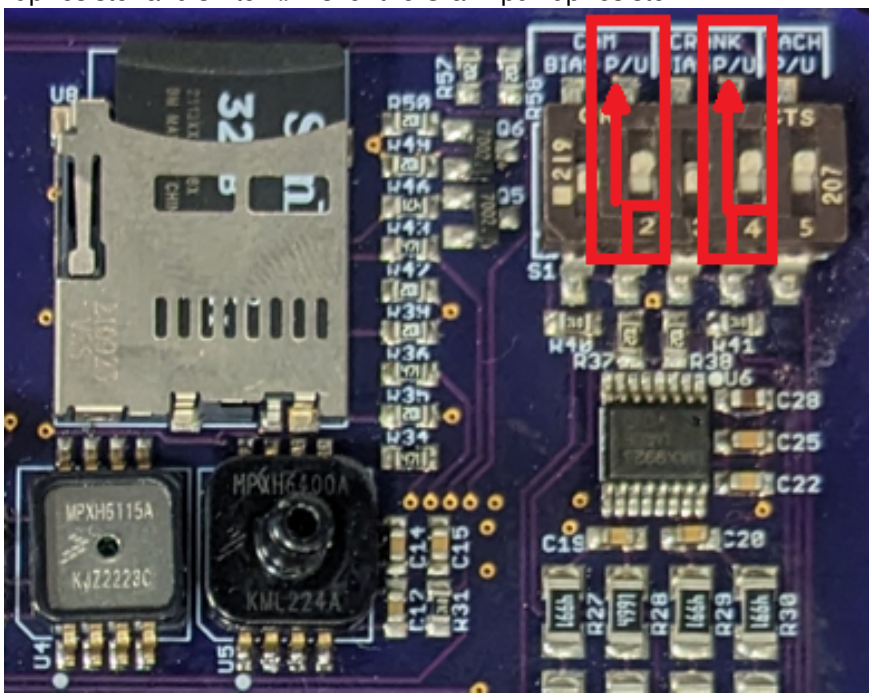
4.3.2 Hall effect and optical sensors

Hall effect and optical sensors work on different principles of physics, but these two types of sensors use the same wiring. These use a switching transistor and are powered from an external source (either 5 or 12 volts; if your sensor can run on either, we recommend using 12 volts). Your typical sensor has three pins - the power wire goes to a 12 or 5 volt feed, the ground wire goes to sensor ground, and the signal wire goes to CKP+ or CMP+ as appropriate. **Do not use CMP- or CKP- with Hall effect or optical sensors. Leave this wire disconnected and taped off; do not connect to ground.**

Some sensors put out a square wave signal on their own, like the tabbed body ones sold by AMP EFI. Others switch from a ground output to a floating (undefined) output. The ones that behave this way require running a 1K resistor from their power terminal to their signal terminal (called a pull-up resistor) to work with MS3Pro. If you're not sure which sensor type you have, supply it with power and ground, and check the voltage on its output wire when the sensor is both far away from any metal and touching a piece of steel (note - do not use stainless steel for this test; some grades of stainless steel aren't magnetic.) If the output voltage stays at zero the whole time, you'll need the pull up resistor; connect the 1K resistor from the signal wire to the power wire. If it is at zero volts in one state and bounces up to at least 5 volts in the other state, it doesn't need a pull up resistor.

In some cases where a Hall effect sensor puts out 12 volts and the wiring harness has severe noise issues, tying the CKP- or CMP- wires to the VREF output can reduce noise.

The Mini has internal DIP switches on the circuit board (labeled as P/U that can be switched 'ON' in order to enable built-in 470ohm pull-up resistors to 5v for both the Crank and Cam inputs. DIP switch #2 is for the CAM pull-up resistor and switch #4 is for the Crank pull-up resistor:



4.3.3 Points triggering and points replacement devices

If you wish to lock out a points distributor mechanism and let the MS3Pro control the ignition, you can treat the points output as a Hall sensor with a ground to floating output. You will need to switch ON the #4 DIP switch for the Crank Pull-Up resistor (see section 4.3.2 above), and also wire the points to the CKP+ wire. Do not use the CKP- wire; tape it out of the way safely.

Using a Pertronix Ignitor or equivalent is similar. With a Pertronix, connect the Ignitor's red wire to switched 12 volt power and connect the black wire to CKP+ (again with the #4 DIP crank pull-up switched ON) and leave CKP- disconnected. Other modules may use different wire colors.

4.3.4 Fuel only installations

CDI ignitions send a high voltage pulsed signal, from 300 to 600 volts, to the coil. You do not want to try to get a tach signal from the coil by any means when using this sort of ignition, including external adapter boxes. Instead, use the ignition box's tach output wire to the CKP+ instead if you are using a distributor advance mechanism to control timing and just want the MS3Pro to control fuel. Do not use the CKP- wire or the CMP+ and CMP- wires.

If you need to trigger off an inductive ignition, such as breaker points, a GM HEI, Ford Duraspark, or Chrysler's 1970s era electronic ignition, use an AXM-120 high voltage isolator to protect the CKP input circuit from the voltage from the coil.

4.4 Other sensors

4.4.1 Temperature sensors

The MS3Pro is designed to work with the type of sensor called a NTC thermistor. NTC stands for negative temperature coefficient, and simply means that the resistance goes down as the temperature goes up. This is how most automotive temperature sensors are set up, including the GM type sensors available from AMP EFI distributors. Normal automotive temperature sensors have two terminals. One connects to the coolant temperature sensor or intake air temperature sensor input of the MS3Pro. The other one connects to the sensor ground wire. These sensors have no polarity and can be wired either way.

There are two common variations on automotive temperature sensors. Closed element type sensors are resistant to water, oil, and other liquids. We recommend using this type for coolant. They are not a good idea to use with air temperature, as they react slowly and are prone to absorbing heat from the intake manifold. Open element sensors are better to use for air temperature. We recommend mounting them upstream of the throttle body to avoid absorbing heat from the intake manifold, unless you have a plastic intake manifold. If you are using a turbocharger or supercharger, the IAT sensor must be mounted downstream of the compressor and intercooler (if used).

Sometimes you will find cars with one wire temperature sensors. We do not recommend using this style as they are grounded through the engine block, and this will make for a noisy signal. A few cars employ a three wire coolant temperature sensor. This sensor is typically two sensors in one housing with a common ground, with one sensor driving a gauge and one sensor for the ECU.

TunerStudio has the calibration data for many common temperature sensors in its sensor calibration tool, including GM, Ford, and Bosch. You'll find those under Tools -> Calibrate Thermistor Tables.

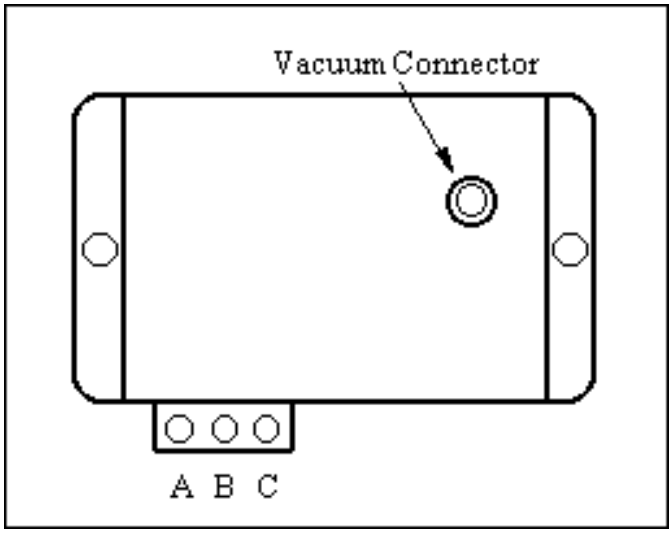
4.4.2 MAP sensors

Most MS3Pro installations use a MAP sensor for load measurement. The MS3Pro Mini has a built in 4 bar MAP sensor.

The MS3Pro supports both frequency and voltage based MAP sensors. We recommend GM MAP sensors if using an external MAP sensor, which are voltage based, because they are inexpensive and readily available in several different maximum pressures. Be sure to select a MAP sensor that can read as much pressure as you intend to run. On turbo cars, we recommend giving a slight amount of extra room to allow the ECU to detect if the boost control has malfunctioned.

MAP sensor type	Maximum boost level
1 bar	naturally aspirated
2 bar	14 psi
3 bar	29 psi
4 bar	44 psi

When using an external sensor, install the MAP sensor with the hose fitting pointed downward, or at most, 30 degrees from vertical. This will prevent oil or blow-by from accumulating in the sensor, which can damage its circuitry.



Here are what the pins on the GM MAP sensor do:

Pin	Function	Connection
A	Ground	Sensor return
B	Sensor output	MAP in
C	Reference voltage	5V+ VREF Out

MAP sensors will have three pins unless they have some sort of additional sensor (usually a temperature sensor) in the same package. The reference voltage connects to the MS3Pro's VREF pin, and the ground pin to the sensor ground wire. For a voltage based MAP sensor, connect the signal pin to the MS3Pro's MAP input wire. An additional MAP sensor may be used to measure barometric pressure on one of the spare analog inputs. Frequency based MAP sensors such as those used in Ford Mustangs must be wired to a Digital Frequency In wire instead of the MAP or analog inputs. (You can also use the CMP+ wire for a frequency based MAP sensor, if you are not using it for cam position.)

4.4.3 Throttle position sensor

The throttle position sensors used on most cars are potentiometer or variable resistance types, which put out a voltage that increases as the throttle opens. The easiest way to check a TPS is to unplug it and put an ohmmeter on its pins. Observe the resistance as the throttle opens and closes. Each pair of pins will behave differently:

The resistance between the VREF and ground pins will remain constant. The resistance between the ground and signal pins will be low with the throttle closed and high with the throttle wide open. The resistance between the VREF and signal pins will be high with the throttle closed and low with the throttle wide open.

If your sensor behaves like this, it will be pretty straightforward to connect it to the MS3Pro.

If the resistance jumps from infinite (or near infinite) to near zero, you have a switch type throttle position sensor, or possibly a seriously defective potentiometer type TPS. These do not provide very much information that MS3Pro needs, as it can tell if you are at idle or full throttle by the MAP sensor information. You can still use MAP based acceleration enrichment. If the TPS input is not used, this wire should be connected to sensor ground.

4.4.4 Mass air flow sensors

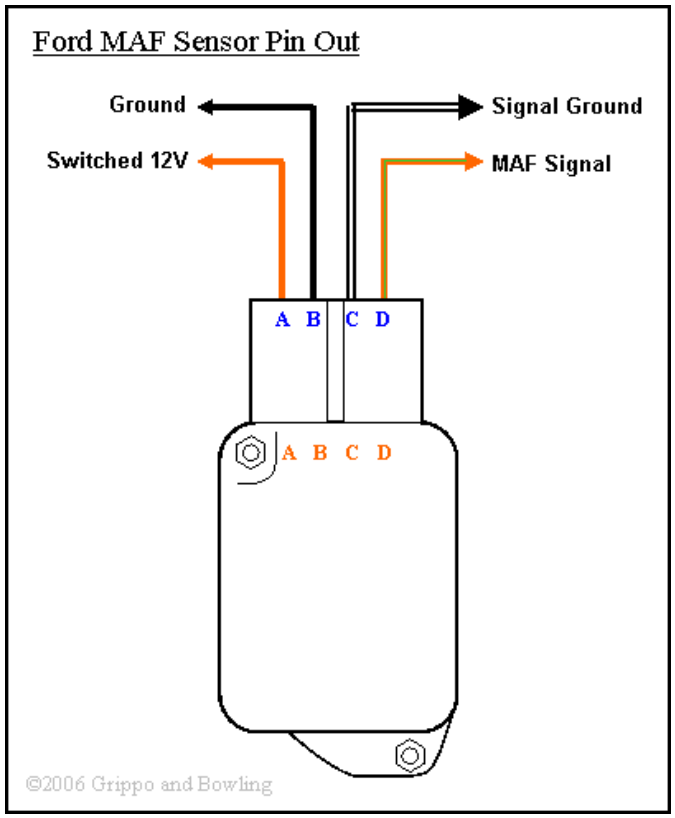
Mass air flow (MAF) sensors provide a measurement of actual airflow through the intake system of an engine. In its most basic form, the fuel required by the engine (in mass per second) is directly related to the air flowing into the engine (in mass per second) so the MAF sensor should give a good basis for the fuel equation. By comparison, Speed Density uses RPM, MAP, MAT and VE to estimate the mass air flow. In practice, like any other sensor input the MAF has its own quirks and can suffer from reversion effects (particularly on "cammy" engines) where the air flow gets measured twice.

MAF sensors have at least three wires: Ground, power supply, signal.

- The power supply is typically 12V and should be taken from a fused source from the main relay.
- Signal GND should connect to MS3Pro sensor ground. If the sensor has only one ground, it's sensor ground.

- If the sensor uses a terminal called Power GND, this should connect to the same point as the MS3Pro power grounds.
- With a voltage MAF, you may use the MAP signal wire or any of the extra analog input wires.
- With a frequency based MAF, you may use any of the Digital Frequency In inputs, or the CMP+ wire if you are not using it for cam position.

Here are some of the more commonly seen MAF pinouts.



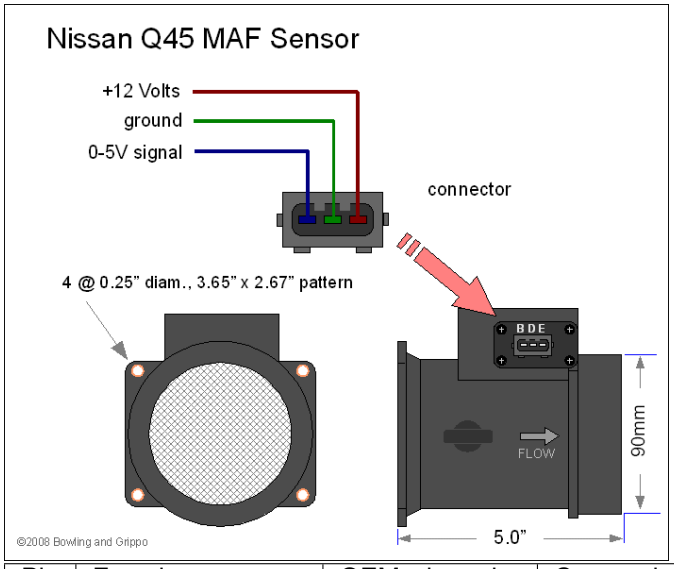
4.4.4.1 Ford 5.0 Mustang 4 pin oval connector MAF

Pin	Function	Connection
A	Switched 12 volts	12 volt fused supply from main relay
B	Ground	To engine block
C	MAF signal ground	MS3Pro sensor return
D	MAF output signal	MS3Pro analog input

4.4.4.2 Ford 5 pin rectangular connector MAF The rectangular connector MAF sensor has 5 pins denoted by A, C, D, and E, with the B pin missing:

Pin	Function	Connection
A	Switched 12 volts	12 volt fused supply from main relay
C	Ground	To engine block
D	MAF signal ground	MS3Pro sensor return
E	MAF output signal	MS3Pro analog input

4.4.4.3 Nissan / Infiniti Q45 MAF This is a popular MAF with many Japanese imports, although its huge diameter (90 mm) and ability to handle large air flow volumes (by OEM MAF standards) would work well on many small-block domestic V8s as well. Note that the picture below uses non-stock wire colors for clarity.



Pin	Function	OEM wire color	Connection
B	MAF output signal	White	MS3Pro analog input
D	Ground	Black	MS3Pro sensor return
E	Switched 12 volts	Black / white	12 volt fused supply from main relay

4.4.5 Oxygen sensors

The MS3Pro can accept input from a narrow band O_2 sensor directly; simply wire the narrow band sensor signal toAnalog In 1, 2, 3, or 4 the MS3Pro Mini. Narrow band O_2 sensors may have from one to four wires. Here is what to do with the wires depending on the type.

- Single wire: The wire is the sensor signal and it grounds through the body.
- Two wires: One wire is the signal and the other connects to the sensor ground.
- Three wires: This is a heated sensor. In addition to the signal wire, there is a heater wire that connects to a switched 12 volt source, and a ground wire which should be connected to the engine block as the heater draws more current than we recommend applying to the sensor ground wire.
- Four wires: These have a signal wire and a sensor ground wire which connects to the MS3Pro sensor ground wire. The two heater wires are normally the same color. One connects to a switched 12 volt source and the other can be grounded to the engine block or chassis.

As a general rule, if the sensor has two wires that are the same color, these two wires are for the heater, and may be hooked up either way. Bosch's wire coloring system is the most common, and looks like this:

Function	Color
Signal	Black
Sensor ground	Gray
Heater	White
Heater	White

However, there are several other color schemes out there; if you run into a different set of colors, consult a factory wiring diagram for your vehicle.

The MS3Pro can also work with most aftermarket wideband oxygen sensor controllers. You will need to install the controller according to the directions that came with it, and connect the controller's analog output wire to the MS3Pro's oxygen sensor input wire. If the wideband controller has a ground specifically for its analog output, it connects to the MS3Pro sensor ground wire. Do not connect any wires labeled "serial output," "digital output," "heater ground," "power ground," or similar to the MS3Pro. Here is a list of what wire to use with common wideband controllers.

Controller	Analog output	Analog ground
Innovate LC-1	Brown *	N/A
Innovate LC-2 or MTX series	Yellow *	N/A
Innovate LM-2	Lime green*	Yellow*
AEM UEGO **	White	Brown (X series only)
Zeitronix ZT-2, ZT-3	White	Brown (ZT-2 only)
PLX SM-AFR	Gray	N/A

*Innovate systems have multiple analog outputs; this is the recommended default.

** AEM X series can also provide data over CANbus instead of an analog signal.

The above list is not intended to be a complete list of what wideband controllers will work with the MS3Pro. As long as the wideband controller has a 0-5 volt analog output, and you can obtain the specifications for its output, you can make it work with the MS3Pro.

Additional narrow band O_2 sensors or wideband controllers may be wired to the spare analog input wires.

The MS3Pro can also accept wideband oxygen sensor data over CAN. The CAN-EGT or other devices can translate Innovate serial data into CAN data for the MS3Pro, and the MS3Pro can also receive CAN data from wideband O_2 sensor controllers from AEM, EcoTrons, and 14Point7.

4.4.6 Knock sensors

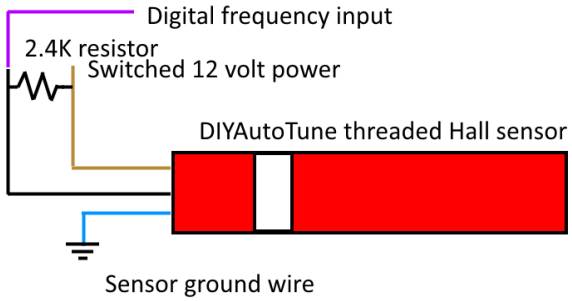
The MS3Pro Mini does not have a dedicated knock input pin and does not wire directly to knock sensors. It can accept on/off signals or 0-5 volt knock intensity signals from an external knock detection module.

4.4.7 Speed and gear sensors

The MS3Pro can use analog, VR, or Hall effect speed sensors.

Analog speed signals are rare, but show up in some aftermarket speedometers. These produce out 0 volts with the vehicle stopped, increasing to 5 volts at some top speed. These may be wired to any unused general purpose analog input.

Hall effect speed sensors are more common. These get 12 volts from the switched power or 5 volts from VREF, and have a ground wire that connects to the MS3Pro's sensor ground. Using the digital frequency inputs will give you the best response, but you can also use a digital switched input. No pull up resistor is required if using our tabbed Hall effect sensor; we recommend a pull up when using our threaded Hall effect sensor. The MS3Pro can use two separate speed sensors for driven and non-driven wheels.



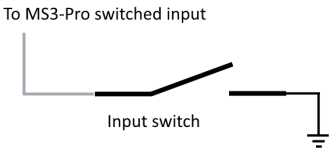
Some bikes are fitted with a Gear Position Sensor. This 0-5V analogue input can be used directly to determine gear, and can connect to any unused analog input. The MS3Pro can also determine the gear you are in by comparing the engine RPM and wheel speed.

When used with a MegaShift transmission controller, the MS3Pro can read speed and gear information through the MegaShift over CAN. No additional sensor wiring is needed, other than the wires for the MegaShift.

4.4.8 Generic on/off switches

There are several inputs the MS3Pro can use as switch inputs for triggering launch control, nitrous, table switching, or other functions. Valid inputs include the Digital Switched In inputs, the Digital Frequency In inputs, and Digital Switched 12V In. Digital Switched 12V In activates when connected by a switch to a 12 volt source. All other on/off switch inputs activate when connected through a switch to ground instead. The Digital Switched 12V In function is

set up this way to allow using it to detect when a nitrous system is energized and arm the system, but can be used for other functions as well. The illustration below shows typical switched input wiring.



Caution: Do not wire the Digital Switched 12V In wire directly to any input that is supplying power to a trans brake or other high powered solenoid. This may result in damage to the input circuit. Instead, use the 12 volt power to the solenoid to switch a relay to control an on/off input.

4.4.9 Other sensors

You can connect any 0 to 5 volt analog signal to the Analog In inputs. The MS3Pro software can use these for data logging or other functions.

4.5 Fuel injectors

Fuel injectors are pretty simple; one side gets switched, fused 12 volt power. The other side connects to the MS3Pro injector output. They are not polarized so it doesn't matter which is which. Current MS3Pro settings have the firing order hard-coded in, so they always fire in alphabetical order.

4.6 Ignition outputs

'Conventional' high-current coils, such as all of the old can-shaped coils that feed distributors, as well as early (1980s-2000s) multi-coil packs, in general have a primary winding resistance of about 3 ohms. This means that charging them with a standard 12~15v automotive electrical system they draw roughly 4~5 amps of current, and then they produce hundreds of volts of flyback voltage on the negative terminal when fired (ever seen points arc?). This combination of rapidly repeating cycles of high current and high voltage means that whatever devices is switching the ignition coil needs have some pretty serious capabilities, both in terms of heat and power management. It's pretty easy for mechanical points to handle this job, but it's a taller order to make solid-state electronics robust enough to tackle this task. The signal from the MS3Pro ECU is a much lower-power "logic-level" signal - the kind of signal seen inside computers, and is absolutely NOT capable of handling this directly. So some sort of device is needed to read the logic-level signal from the MS3Pro and then drive the coils accordingly. Fortunately there are specialized power-transistors designed specifically for this purpose, but it's critical that they are mounted on a GOOD HEAT SINK in order to keep their temperature in check during operation. This device is often called an "ignition module," "ignitor," or "coil driver." Such coils can often be identified as they only have two connection terminals, one positive terminal for +12v, and one negative terminal controlled by the driver. Since about 2010, most modern automotive coils have come with the driving transistor built into the coil assembly itself. We refer to these as "logic-level" or "smart" coils. Generally such smart coils have MORE than just 2 connection terminals because of the extra functions they contain, and these CAN be driven directly by the logic-level ECU output. If you're using a factory ignitor with matching factory coils, you can expect that the coils and ignitors are designed to work together. If your coils require an ignition module but you do not have one, our QuadSpark ignition module can drive most factory coils, or we also offer the MS3Pro EVO+ HD version, which has 8 built-in high current power transistors in place of the normal logic level circuits. This is the only ECU in the lineup that can therefore be directly wired to traditional two-terminal coils without the need for any additional ignition module.

4.6.1 Conventional high-current / power transistor driven coils

Some examples of where you would expect to find conventional coils include:

- Almost all older distributor-based ignition systems with a single coil
- Delphi coil packs with horizontal plug wire terminals used on many Mopar/Dodge and Subaru vehicles up until the mid-2000s

- Dodge V8 (Hemi) individual and twin-packaged COP coils.
- Ford EDIS-based coil packs
- Ford V8 COPs
- BMW COPs up to the mid-2000s. Note that unlike most other conventional coils, these tend to have THREE terminals opposed to the normal two (the extra is a supplemental ground).

IF YOU PLAN TO USE ONE OF THE ABOVE ENGINES FOR YOUR PROJECT - PARTICULARLY ANY OF THE ABOVE V8S - THEN YOU SHOULD STRONGLY CONSIDER THE MS3PRO EVO+ HD ECU WITH 8 BUILT IN COIL DRIVERS and heat sink!

Alternatively, you can still use any other MS3Pro ECU, but you'll additional need one (or two) QuadSpark modules to drive the coils.

4.6.1.1 QuadSpark ignition module The QuadSpark module is mounted in an aluminum enclosure which provides the necessary heat sinking for the transistors. Here are the pins to use on the QuadSpark:

Wire color	Thickness	Connection
Yellow	20 gauge	MS3Pro Spark Out A
White	16 gauge	Spark A to coil negative
Orange	20 gauge	MS3Pro Spark Out B
Pink	16 gauge	Spark B to coil negative
Dark green	20 gauge	MS3Pro Spark Out C
Light green	16 gauge	Spark C to coil negative
Blue	20 gauge	MS3Pro Spark Out D
Violet	16 gauge	Spark D to coil negative
Black	4 x 14 gauge	Ground to engine block or cylinder head

4.6.1.2 MSD and other single channel capacitive discharge ignitions In order to drive a Ford TFI, MSD CDI, or other similar ignition that requires a 12v square wave input with an MS3Pro Mini, please use the AXM-130 dual-channel spark output adapter that is available here: AXM-130 Driver

4.6.2 Logic-level 'Smart' Coils (and coil packs)

These are more modern coils with the power transistors built right into the coils themselves. These can be driven directly by the MS3Pro Module, Mini, or EVO+ ignition outputs without the need for an additional external driver.

Note however that the MS3 Pro EVO+ HD CANNOT be used with logic level coils; it is intended SOLELY for use with conventional high-current coils!

4.6.2.1 IGN-1A coil with built in ignition module The IGN-1A is a high powered coil with a built in ignition module, capable of handling engines that make over 250 hp per cylinder. These can be used both for coil per plug applications or with a distributor. These use a five wire connector. The connector pins are designated with letters molded into the connector:

Connection letter	Function
A	Spark output trigger from MS3Pro
B	ECU ground - connect to sensor return
C	Secondary ground - connect to engine block or cylinder head
D	Power ground - connect to battery negative terminal
E	Switched +12 volt power

4.6.2.2 LS series coils The coils on GM's LS series engines are a popular choice as they are readily available, inexpensive, powerful, and have a built in ignition module. They can be wired directly to the MS3Pro. LS1 and LS2 coils use different connector designs, but the connectors are always marked with small letters to indicate the function, and the function is the same on both:

Connection letter	Function
A	Power ground - connect to engine block or cylinder head
B	ECU ground - connect to sensor return
C	Spark output trigger from MS3Pro
D	Switched +12 volt power

4.6.2.3 VW coils Volkswagen Automotive Group (VAG) utilized quite a few different logic-triggered coils and coil packs in the 1990s and 2000s, making them a great source for ignition setups - especially 4-cylinder engines - that can be connected directly to an MS3 without the need for extra external ignition modules. These include:

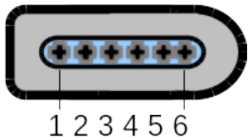
- 0 004-402 001 Single logic level coil for use with a distributor from Mk3 2.0L VW Golfs:

Pin number	Function
1	Power ground to cylinder head
2	SparkA trigger from MS3Pro
3	Switched +12 volt power

- 032 905 106B - Logic level 4-cylinder wasted-spark coil pack from Mk4. 1.6L VW Golfs, uses connector 1J0973724:

Pin number	Function
1	SparkA trigger from MS3Pro
2	Switched +12 volt power
3	SparkB trigger from MS3Pro
4	Power round to cylinder head

- 06A905097 4-cylinder sequential logic level coil from early 2000s VW Golfs, Beetles, and Jettas, uses connector 1J0973726:



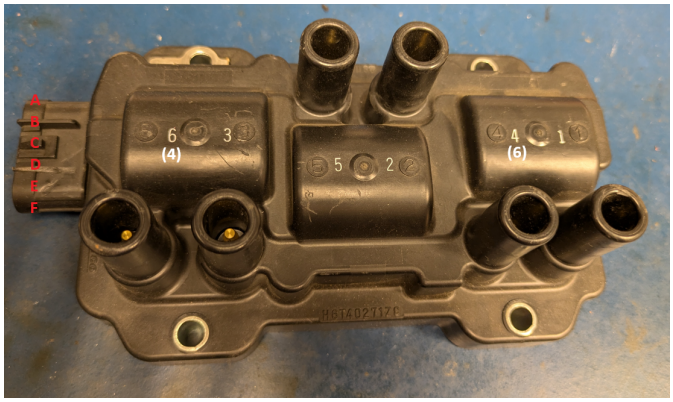
Pin number	Function
1	Power ground to cylinder head
2	SparkA trigger from MS3Pro
3	SparkB trigger from MS3Pro
4	SparkC trigger from MS3Pro
5	SparkD trigger from MS3Pro
6	Switched +12 volt power

- VW four pin pencil coils appeared on the VW / Audi 1.8 Turbo from 2001 and later, and many other VW-Audi Group products. These can be driven directly from the MS3Pro.

Pin number	Function
1	Switched +12 volt power
2	Ground to cylinder head
3	SparkA trigger from MS3Pro
4	Ground to cylinder head

4.6.2.4 GM D599A 6-cylinder wasted-spark logic coil pack

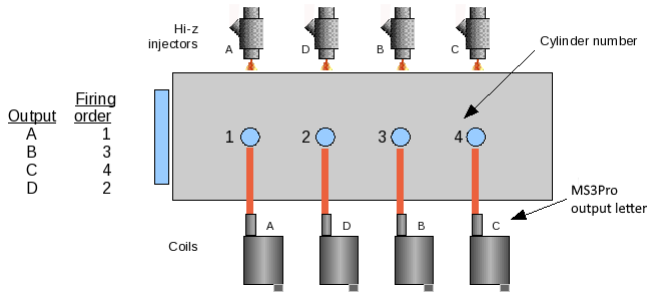
- For any 6-cylinder engine, a great wasted-spark option is the GM D599A logic-level coil pack. It comes from mid-2000s GM LZx 'High Value' V6 engine platforms (such as Chevy Malibus, Impalas, Monte Carlos, etc.), uses a Delphi/Packard 6-way GT 150 connector, and has a nice flat bottom for easy mounting. Note the cylinder numbering is for the GM V6 firing order, but for a standard I6 firing order just swap the #4 and #6 plug wires:



Pin number	Function
A	SparkB (center coil) trigger
B	SparkA (coil opposite connector) trigger
C	SparkC (coil next to connector) trigger
D	Signal ground (to ECU)
E	Power ground to cylinder head
F	+12 volt power

4.6.2.5 M&W capacitive discharge ignitions The MS3Pro standard spark outputs can supply power, but not ground, which creates a minor issue for M&W ignition boxes. You will need to add a 1K resistor teed from each spark output to ground when using an M&W multichannel CDI box. Set the spark output on the MS3Pro to “Going High”, and set the M&W ignition to trigger on the falling edge.

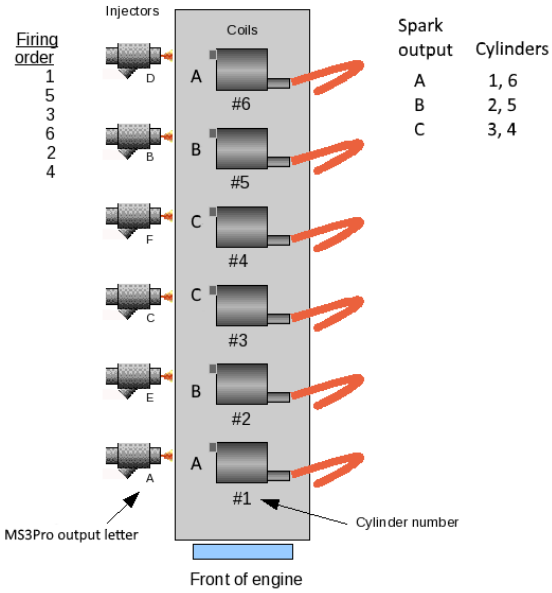
4cyl, even-fire, 1-3-4-2, Sequential fuel, Coil-on-plug logic coils



Output	Firing order
A	1
B	3
C	4
D	2

4.6.2.6 Common firing orders

6 cyl, even-fire, 1-5-3-6-2-4, Sequential fuel with wasted spark

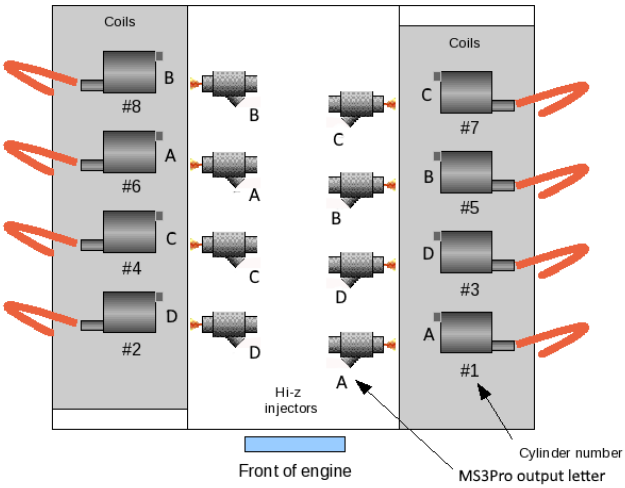


Fuel Output	Firing order
A	1
B	5
C	3
D	6
E	2
F	4

Spark output	Cylinders
A	1, 6
B	2, 5
C	3, 4

V8, even-fire, semi-sequential
Works for both 1-8-7-2-6-5-4-3
and 1-8-4-3-6-5-7-2 firing orders

Output	Cylinders
A	1, 6
B	5, 8
C	4, 7
D	2, 3



4.7 IAC valves

IAC stands for Idle Air Control. These valves work like a small throttle under ECU control to allow the MS3Pro to adjust idle speed. Sometimes you will see these called an ISC valve, for Idle Speed Control, or BAC, for Bypass Air Control. Idle air control valves come in several varieties.The MS3Pro Mini is set up to control PWM type IAC valves with either 2 wire or 3 wire connections. It does not have a stepper IAC controller.

4.7.1 Stepper IAC valves

The MS3Pro Mini does not have support for stepper IAC valves due to its small pin count. We may offer an external IAC module in the future for these applications.

There's no such thing as a stepper motor with less than four wires. If your IAC valve has two or three wires, it is probably a PWM style valve, described below.

4.7.2 PWM IAC valves

A PWM style valve uses a solenoid that opens or closes the valve, and the ECU controls its position by pulsing the solenoid coil off and on rapidly. These can sometimes be simpler to set up than a stepper IAC valve, as the position is just a function of how much current flows through the coil. These come in two varieties. One uses a single coil, which pushes the valve open (or, rarely, closed) when the coil energizes. Another design has two coils, one of which pushes the valve open and the other pushes it closed. The MS3Pro can drive both sorts.

A single coil IAC valve will have two wires. One connects to switched 12 volt power; the other connects to the MS3Pro PWM / Idle Out 1 wire. Sometimes these valves, like injectors, have no polarity. Other valves have a built in diode, and it is important with such a valve to make sure you do not wire it backwards. Consult the wiring diagram for the car the valve came on if you aren't sure which terminal should receive power.

Two coil IAC valves have three wires. One terminal is for 12 volt switched power; the other two connect to the MS3Pro PWM outputs, one of which should be PWM / Idle Out 1.The other should be controlled by a spare Injector output. If you are using all 8 injector outputs, then you should either use a single coil IAC or connect one of the two IAC coils through a 30~40ohm, 50Watt power resistor in series to ground. This will 'hold' the valve in either the open or closed position (depending on which coil is getting grounded) and give the PWM / Idle Out 1 output a force to work against for more stable control of the valve.If using two ECU outputs, you can specify in the software which output closes the valve and which opens it. Usually, the center wire on the connector gets 12 volt power. If you're not sure which is the power wire, check with an ohm meter. The resistance from the ECU connections to power will be one half the resistance from one ECU connection to the other.

Note that PWM IAC valves **must** be wired in such a way that they do not receive 12 volts with the key off.

4.7.3 On/off IAC valves

Rare but supported. The MS3Pro's PWM / Idle Out 1 can drive an on/off idle solenoid directly if the solenoid draws 3 amps or less.

4.7.4 Thermal IAC valves

These valves were used on some early Bosch L-Jetronic systems and are not controlled by the ECU. They use a thermal element that warms up and reduces idle speed as they heat up. If you have such a valve, power it from the fuel pump relay.

4.7.5 DC servo IAC valves (currently not supported)

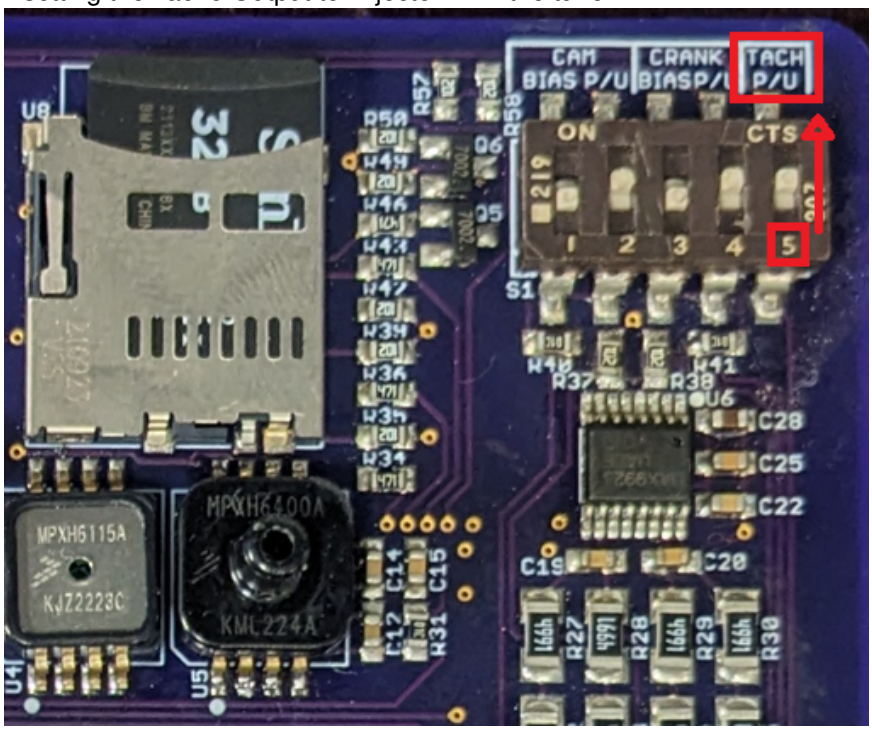
This design incorporates a two wire DC servo with a three wire position sensor. We've only seen them on a few Ford and Mitsubishi throttle body injected engines from the 1980s. The MS3Pro does not currently support this idle valve type.

4.7.6 Fuel pump relay

The MS3Pro fuel pump relay output is a low side driver that supplies ground, not power. It can only handle a maximum of 3 amps, so you will need to use this to turn on a relay to provide the fuel pump with enough power. We recommend supplying the fuel pump relay with a separate source of 12 volt power from the ECU main relay to minimize electrical noise. Do not wire a fuel pump directly to the MS3Pro. Also, do not wire the fuel pump relay coil in such a way that it receives 12 volts with the key off, as this power can back-feed into the MS3Pro and cause it to stay powered up when the ignition is turned off.

4.7.7 Tach output

There is no dedicated 12v square wave tach output on the Mini, however the Injector H output can be repurposed for this function by switching 'ON' the #5 micoswitch (labeled 'TACH P/U') to provide a pull-up resistor to 12v, and then setting the Tacho Output to 'Injector H' in the tune:



4.7.8 High current outputs

Any of the Mini's unused Injector outputs can be repurposed as generic/programmable low side drivers capable of controlling devices which draw up to 5 amps. Recommended uses besides on/off output include nitrous control and running boost control solenoids, which pulse open and shut.

4.7.9 PWM medium current outputs

The PWM/Idle output is a low side driver like the high current outputs, but can operate at a higher frequency and have a maximum current of 3 amps. Devices connected to these outputs absolutely must not receive power when the key is off, as the circuit used to clamp high frequency voltage spikes can feed power back into the MS3Pro's power supply and prevent the ECU from turning on. Maximum frequency is 1021 Hz.

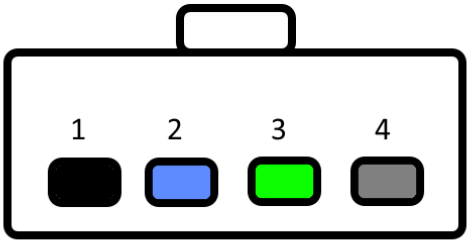
4.7.10 Solid state relay control

Use a solid state relay to control devices that require a pulsed signal with more current than the MS3Pro's outputs are rated for. Any high current, PWM, or injector output can be used to run a solid state relay. One readily available solid state relay is the Standard Motor Products RY330K (ask your local auto parts store for the radiator fan relay

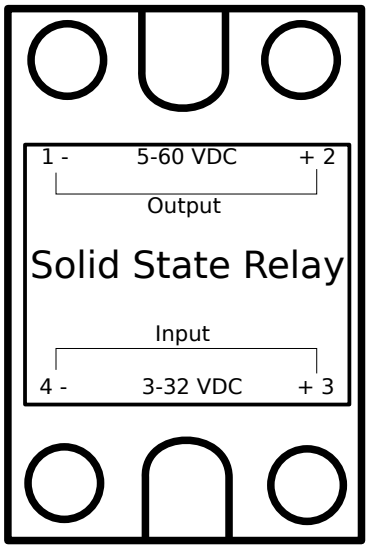
for a 2001 Jeep Grand Cherokee with a 4.0 inline six). The part number given above include a handy color-coded connector, and is fused for 40 amps in its factory application. Here is what the wires on this relay do:

1. Black: Power ground.
2. Blue / Pink: Trigger signal; connect to MS3Pro output.
3. Green: Power output, supplies 12 volts to device being controlled.
4. Gray: 12 volt switched power.

Note that while the Standard Motor Products item uses the OEM Chrysler wiring colors, some other manufacturers that make OEM replacement parts do not adhere to factory wiring colors and use other color schemes. If you get one of these - or a connector with uncrimped pins - see the picture below, which is looking into the pins on the relay with the locking tab facing up.



Many industrial solid state relays use an industrial standard pin numbering, shown below.



If using one of these relays, here is how to connect its terminals.

1. Output - supplies 12 volts to device being controlled. If switching an inductive load (pump, fan, motor, etc.) also install a flyback diode across the positive and negative leads of the load, with the banded end of the diode to the positive side.

2. Power input - connect to 12 volt power.
3. Positive trigger signal - connect to switched 12 volt source.
4. Negative trigger signal - connect to ECU output along with a 1k pull-up resistor to the +12v source.

Caution: Counterfeiting is very common in industrial solid state relays due to their simple design. And counterfeit units seldom have the quality of a genuine Omron or Fotek unit. Many counterfeit solid state relays are only capable of handling half their rated current, sometimes much less. Only buy industrial -type solid state relays from reputable sources.

4.7.11 Grounding notes

The MS3Pro uses multiple ground planes to reduce noise. It is important that you connect both the logic ground at pin 16 on the white connector and all the power grounds on the gray connector to the battery negative terminal. It is less than ideal, but still acceptable in most cases, to ground the MS3Pro to the engine block or cylinder head. Avoid grounding the MS3Pro to the frame rails or sheet metal; these points often have issues with rust and spot welds limiting how much current can flow through them.

Leaving the logic ground disconnected can result in improper operation.

The MS3Pro Mini has a sensor ground on pin# 21 of the connector. All sensors except VR sensors should be grounded to this pin. Do not ground the sensors to the chassis, engine block, or other external point, as this can create noise issues.

4.7.12 Output comparison chart

These charts show the current charts, clamped voltage, and other key information about the MS3Pro Mini's different output circuits.

Name	Output type	Max rated current	Clamping voltage
Injector	Low side	5 A	36 V
PWM	Low side	3 A	12 V
Spark	5V high side	100 mA*	N/A
Injector H (as Tach)	Low side w / 12 V pull up	1 A	12 V

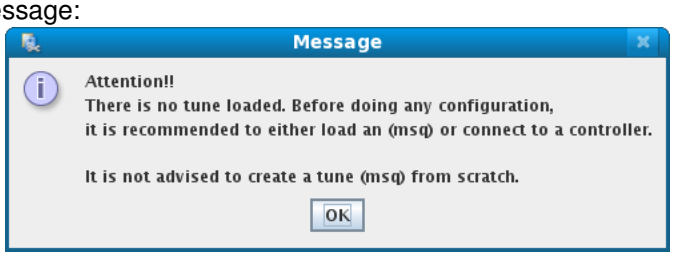
* Do not exceed 250 mA total for all spark outputs at one time.

5 Setting up a basic configuration

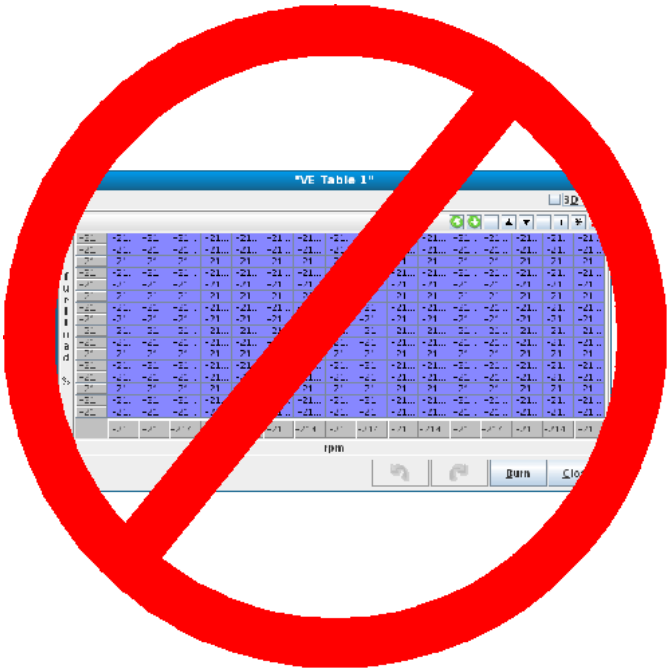
5.1 Connecting the MS3Pro to your laptop

1. Power up the MS3Pro and plug it in to your laptop with either the provided serial or USB cable. The MS3Pro is not powered through the serial cable; it will need to be connected to a good power and ground source to communicate.
2. Open TunerStudio. TunerStudio will check for available updates; if there is one, accept it and let it install.
3. From the File menu, select "New Project".
4. You may enter a name to keep track of the project and a detailed description in the Project Description field.
5. Click the “Detect” button. TunerStudio should automatically detect that you have an MS3Pro and what port it is using.
6. Clicking the Next button will bring up a menu of project settings.
7. Here are what the project settings do:
   - **Oxygen Sensor/Display** - selects the default gauges for oxygen or lambda sensor. If set to wideband, TunerStudio can display either lambda or air/fuel ratio. In narrow band mode, TunerStudio will display and log sensor voltage.
   - **Temperature Display** - preferred temperature units
   - **CAN_COMMANDS** - Allows tuning an MS3 over CAN when not using the USB or serial connection.
   - **Internal Log Fields** - normally deactivated. Allows developers to log special fields into SD card logs
   - **Port Status** - allows displaying the status of input and output ports directly as indicators, and recording port status in data logs. Useful for troubleshooting, but can slow down data logging rates.
   - **OutMSG Editing** - Enables a system for sending CAN data to other CAN devices. Only needed if the documentation for a CAN device you are using requires this.
   - **EXPANDED_CLT_TEMP** - Raises the range and limits of the CLT temperature readings so that the CLT channel can be used with a cylinder head or oil temperature sensor, rather than the default range/limits intended for the operating temperature range of liquid coolant.
8. When you are done with the project settings, click Next. This will bring up a page of communications settings.
9. The final screen for project setup is to select a dashboard view. It is safe to accept the default and Finish. TunerStudio allows advanced users to create their own custom dashboard layouts.
10. At this point, you’re ready to start setting up the MS3Pro to run your engine.

The first rule of tuning is that you do not want to tune from a blank file. You can either connect to the MS3Pro and retrieve the tune on it (this is done automatically) or open an existing tune file, but be careful if you see this message:

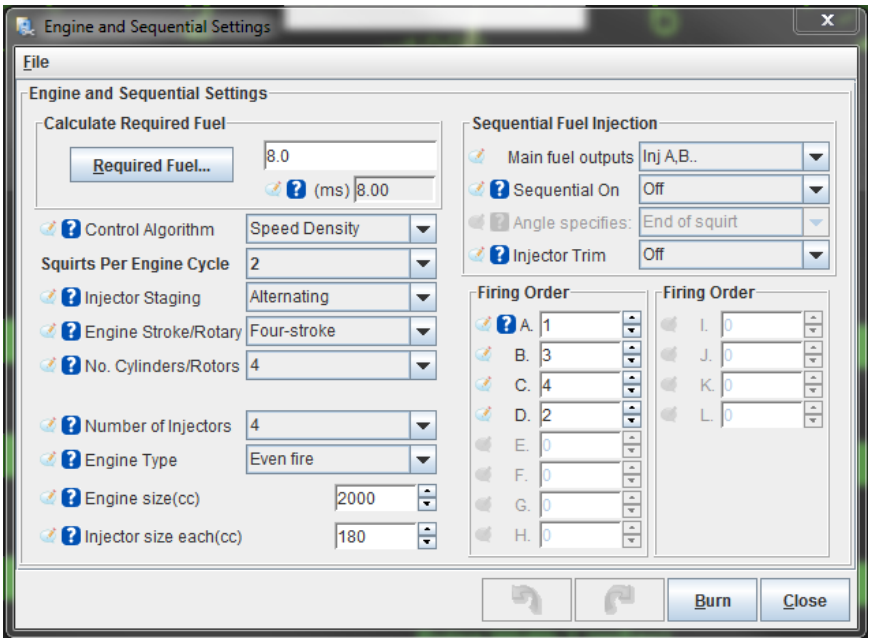


This means that you have no tune loaded. This is OK if you are just having a look, but **do not** try to start a tune like this. TunerStudio will default to a table full of zeroed out values - or in some cases, the numbers will be set to their lowest possible value. It’s not worth chasing down all the nonsense values to turn this “blank file” into a real tune. Start with either the tune in the ECU or one of the sample tunes provided in the software package.



We recommend that you do not have the ignition coils hooked up while doing a basic setup on the MS3Pro. During the setup stage, you will define what type of ignition hardware that you are using, but if the ignition you have doesn't work with the default settings, you may overheat your ignition module and damage it before you are done with your setup.

5.2 Basic engine constants



First, we'll go over the basic settings you need to go through for starting the engine. The following are the most essential settings. We'll come back and go over the other menu items later. The first stop is under the **Basic** / **Load Settings** menu. Select **Engine and Sequential Settings.** Here is what the settings do.

- **Required Fuel:** This is your base pulse width. It is the amount of pulse width the engine would theoretically need at 100 kPa and 100% volumetric efficiency. While you can enter it directly, we recommend using the calculator that the "Required Fuel..." button pulls up. Here, you will enter the engine displacement, number of cylinders, injector fuel rate, and air/fuel ratio. Usually the air/fuel ratio should be stoichiometric for the

fuel used, such as 14.7 for gasoline or 9.85 for E85 ethanol blends. Note that you will want to run this after entering the injector size and displacement numbers, below.

- **Control Algorithm** - This setting controls the method with which engine load is calculated. Engine load represents how hard the engine has to work and can be based on many factors such as manifold pressure, throttle position, air mass, or combinations of these. Note that this setting only sets the control algorithm for fuel injection and related settings; it does not set the control algorithm for ignition and related settings. You may use different algorithms for different tables. The following settings may be selected.
  - Speed Density - Use the MAP (Manifold Absolute Pressure) sensor to determine load. In this case, the vertical axis of any fuel table lookup is in kilopascals (kPa). The maximum value reported by the MAP sensor (in non-turbo applications) will be the same as the barometric pressure.
  - Percent Baro - This setting is similar to the Speed Density setting in that the MAP sensor is used to determine load. However, instead of directly using the manifold pressure, the manifold pressure is divided by barometric pressure to give a percentage of barometric pressure. This setting can be useful for those who regularly drive at high altitudes. It ensures that regardless of barometric pressure, all table lookups operate over 0-100%. For example, if barometric pressure is 80 kPa, and the engine is operating at 50 kPa, the actual value used for table lookups is 50kPa/80kPa or 62.5%.
  - Alpha-N - Use the throttle position to determine load on the engine. Be sure to calibrate the throttle range using Tools -> Calibrate TPS before using this setting.
  - MAF - Use the MAF sensor and related calculations to directly determine the amount of fuel to inject. In this mode, the VE table is not used to determine the amount of fuel to inject.
  - ITB - This mode was created specifically for naturally aspirated engines running with independent throttle bodies. It combines alpha-n (at high engine loads) with speed density (at low engine loads), using the load calculation that makes the most sense at each RPM. For example, most ITB setups do not have good vacuum at idle or low RPM, and slightly touching the throttle makes them lose all vacuum, but at higher RPM start to respond more like a traditional single throttle body engine. This mode allows the use of speed density at low engine loads and switches to alpha-n at high loads, with an adjustable switchpoint curve over RPM.
- **Squirts Per Engine Cycle** - Only used when running batch fired injection. Determines the number of times per engine cycle (two revolutions on a four-stroke engine) injectors are squirted when in a batch injection mode. This setting has no effect when injecting sequentially.
- **Injector Staging** - This setting is used only in batch injection modes. It determines whether the two injection channels are squirted at the same time, or in an alternating fashion.
- **Engine Stroke** / **Rotary** - Specify whether the engine is a four or two stroke piston engine, or a Wankel rotary engine. The rotary mode allows for trailing spark outputs on Mazda and other dual plug rotary designs.
- **Number of Cylinders** / **Rotors** - Sets the number of cylinders for the engine. The MS3Pro can run sequential injection on up to 12 cylinders.
- **Engine Type** - Determines whether the engine is an even-fire engine or odd-fire engine. An even fire engine is an engine where the crankshaft moves an equal number of degrees between each cylinder's top dead center (TDC). An odd fire engine may have a different number of degrees between TDC on some cylinders when compared with others. You are only likely to see odd fire settings on certain V6 models, "crossplane" inline fours and split crank parallel twins (found on some motorcycle engines), and V type engines having less than 6 cylinders. Normal V8, inline four, and inline six engines are all even fire.
- **Engine Size** and **Injector Size** are used for MAF fueling and the Required Fuel calculator.
- **Main fuel outputs** - This should be set to "Injector A, B..." Fuel for all normal MS3Pro installs. This is included to allow you to check tune files imported from standard MS3s. It can be set to use only injector I and J, but this is mostly to maintain compatibility with MS3 V3.0 / V3.57 tune files.
- **Sequential On** - The injector sequencing may be set to three different modes.

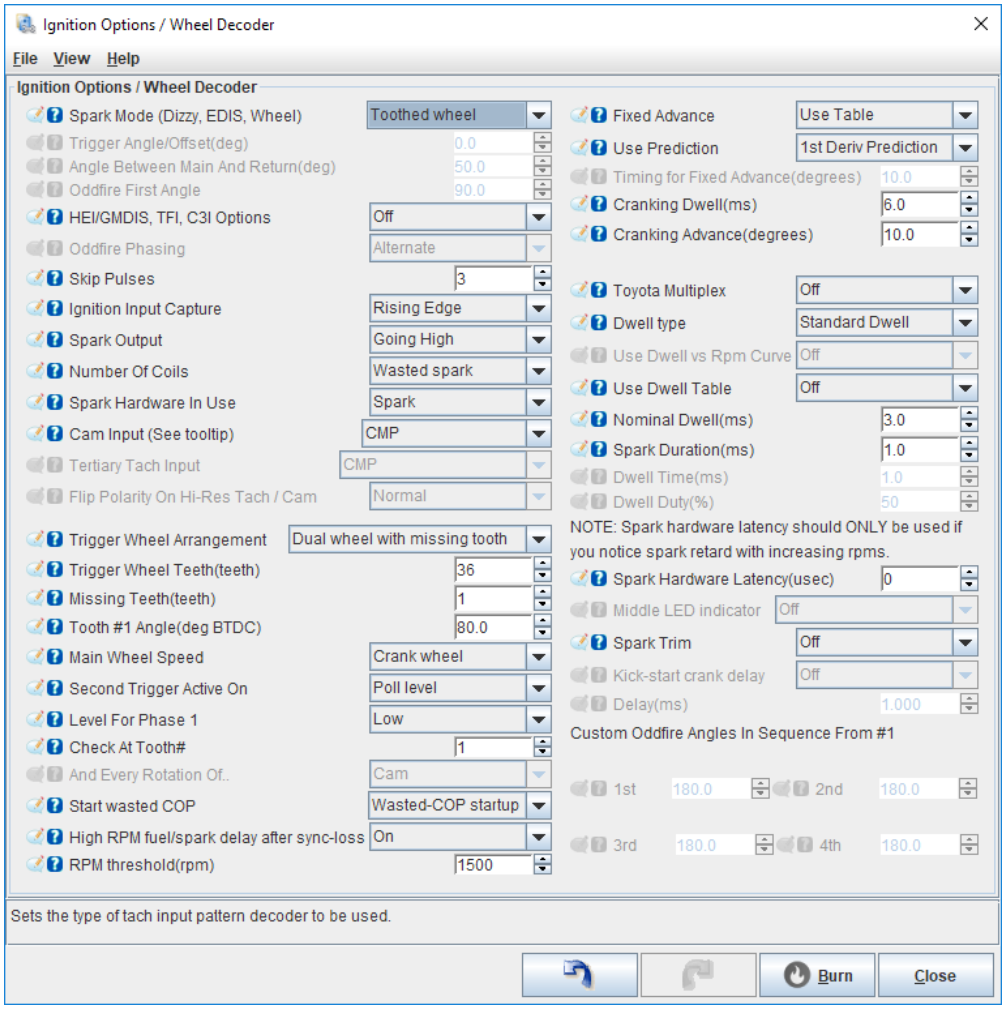
  - Off - Use batch injection. The channels are divided into two groups, and all injectors in a group are squirted simultaneously. This mode would typically be used if your engine has an ignition input like a HEI distributor that provides for no cylinder identification.
  - Semi-Sequential - Injects once every rotation, firing injectors in pairs. The injectors are timed, so you can set the angle at which they fire. Injectors should be wired in the same manner they would be wired for fully sequential injection.
  - Fully Sequential - Use fully sequential injection. In this mode, outputs A-H are squirted in sequence, so care must be taken to use the correct wiring for the engine's firing order. For example, for a four cylinder engine with firing order 1-3-4-2, the following wiring should be used:
    * Output A - Cylinder 1
    * Output B - Cylinder 3
    * Output C - Cylinder 4
    * Output D - Cylinder 2

Note that to use semi-sequential or sequential injection, you will need to select a spark mode that supports sequential injection. See ignition settings for more details.

- **Angle Specifies** - The MS3Pro has a tunable injector timing table, and this setting determines how the MS3Pro interprets that table.
  - End of Squirt - The angle specified in the injector timing table specifies the angle of the end of each squirt. This should be used by most people for most engines.
  - Beginning of Squirt - The angle specified in the injector timing table specifies the angle of the beginning of each squirt.
  - Middle of Squirt - You can also use the injector table to specify the mid-point of the injection event if you find this easier to tune.
- **Injector Trim** - Turning this on allows adjusting the injectors separately using injector trim tables.
- **Firing order** - The firing order should be set to the firing order that your engine uses. It does not affect the order in which the outputs are triggered. Instead it is used to associate an injector trim table with a particular physical output so that when injector trim is applied, it is applied to the correct cylinder. It is also used if closed-loop EGO is being done per-cylinder.

5.3 Ignition settings

Next, proceed to the **Ignition Settings** menu and select **Ignition Options** / **Wheel Decoder** to set up how the MS3Pro gets its RPM input and controls the ignition.



Here are what the settings under this menu do. We'll go over setting these up, then the next subsection will cover the settings for specific engine modes.

- **Spark mode** - this sets the type of tach input pattern decoder to be used. Four of these modes are generic, while the remainder of these are model specific ignitions. Here are the four generic modes. The chart below shows what the model specific modes cover.
  - Fuel Only: The ECU is not controlling ignition, and receives a tach signal of one pulse for every ignition event.
  - Basic Trigger: A simple distributor trigger setup where the ECU controls the ignition, but receives one pulse per ignition event. Common examples include Ford TFI, GM computer controlled HEI, and MSD's "flying magnet" crank trigger.
  - Trigger Return: Similar to Basic Trigger, but there is a Hall effect or optical distributor where the leading edge of the signal is used to calculate timing when the engine is running, and the trailing edge is used for cranking timing. Common in VW and other European Hall effect distributors.
  - Toothed Wheel: This is a user defined trigger wheel configuration. There are two basic types: "Dual Wheel" mode has one trigger wheel with equally spaced teeth and a one tooth sync signal, while "Missing tooth" mode has a wheel where most of the teeth are equally spaced, but there is a gap of the width of one or two teeth. This wheel may or may not be paired with a second sensor for cam sync use. Ford, Bosch, and Nippon Denso OEM systems often use this mode. So do the common aftermarket dual sync type distributors.

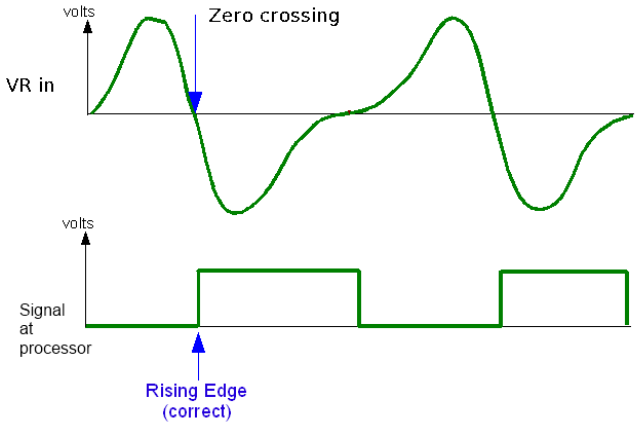
Note that some modes in the chart below can be run without a cam sensor, but if the cam sensor is omitted, the engine will not be able to run sequential injection or coil on plug.

Spark mode	Sequential capability	Common applications
Basic trigger	Only with Ford TFI signature PIP	Basic distributors - GM HEI, Ford TFI, MSD, some Bosch
Trigger return	No	1980s era VW distributors
Fuel only	No	Mechanical advance distributor
Toothed wheel	Yes, with appropriate cam sensor	Many Ford, BMW, Toyota
EDIS	Yes, with appropriate cam sensor	Fords with external EDIS ignition module
420A / Neon	Yes	1995-1999 Dodge Neon, Mitsubishi Eclipse 420A
36-2+2	Yes	Chrysler New Hemi and other 2003 and later engines
36-2-2-2	Yes	Mazda RX8, many Subarus after 2000 without VVT, some Suzuki engines
Subaru 6/7	Yes	Subarus from 1990s through mid 2000s
Miata 99-05	Yes	1994-2005 Mazda MX5 Miata
6G72	Yes	Many Mitsubishi and Chrysler V6s
IAW Weber	Yes	Ford Sierra Cosworth, many Fiat, Lancia
CAS 4/1	Yes	Mazda Protege optical distributor, Mitsubishi 4G91
4G63	Yes	Mitsubishi 4G63, 1990-1997 Mazda MX5 Miata
Twin trigger	No	Motorcycles
Chrysler 2.2 / 2.5	Yes	Chrysler K-car 2.2 / 2.5 multiport and turbo
Renix 44-2-2	Yes, if cam sensor is used	1987-1990 Jeep 4.0, many 1980s Renaults
Suzuki Swift	No	Suzuki Swift GTI
Suzuki Vitara 2.0	Yes	Suzuki Vitara, Sidekick
Daihatsu 3cyl	Yes	Daihatsu 3 cylinder
Daihatsu 4cyl	Yes	Daihatsu 4 cylinder
VTR1000	No	Many Honda carbureted V-twins
Rover #1	No	Rover K-series
Rover #2	No	Rover K-series
Rover #3	No	Rover K-series
GM 7X	Yes	GM 60 degree V6, Saturn, 4.2 Atlas
QR25DE	Yes	Nissan QR25DE
Honda RC51	Yes	Honda RC51, S2000, injected four cylinder motorcycles
Fiat 1.8 16V	Yes	Fiat 1.8 16V
Optispark	Yes	GM LT1 and variants, Nissan / Infiniti VH45DE
Nissan SR20	Yes	Nissan SR20DET, CA18DET with coil on plug
Nissan RB25	Yes	Nissan Skyline RB series, 300ZX VG30DE
LS1	Yes	GM LS1 and variants
YZF1000		Yamaha YZF1000
Honda Acura	Yes	Honda J series V6
VQ35DE	Yes	Nissan VQ35DE V6
Jeep 2000	Yes	1991-2000 Jeep 4.0
Jeep 2002	Yes	Jeep 4.0 and 3.8 with distributor-less ignition
Zetec VTC	Yes	Ford 2.0 Zetec with variable valve timing, Jaguar V8
Flywheel tri-tach	Yes	1980s era Porsche and Audi
2JZ VVTi	Yes	Most Toyotas with VVTi variable valve timing
Honda TSX / D17	Yes	Honda D17, K series, K24A2, 2006 and later S2000
Mazda6 2.3 VVT	Yes	Mazda 2.3 with variable valve timing
Viper V10	Yes	Dodge V10 with JTEC ECU
HD 32-2	Yes, using MAP sensor	Fuel injected Harley-Davidson V-twins
Miata 36-2	Yes	Flyin' Miata aftermarket trigger
Daihatsu 12+1	Yes	Daihatsu EF-SE and other coil on plug 3 cylinders

Subaru 36-2-2-2 VVT	Yes	Subaru engines with variable valve timing (AVCS)
Sea-Doo PTEC	Yes (Two stroke only)	Sea-Doo PTEC two stroke watercraft
Nissan QG16	Yes	Nissan QG16
Madza MZR	Yes	NC Miata and other Mazda versions of the MZR four cylinder
6G75	Yes	Mitsubishi 6G75 V6

- **Trigger Angle/Offset** - There are three different ways this setting is used:
  - In Basic Trigger and Trigger Return, it specifies the engine angle BTDC that each trigger pulse is received.
  - In Toothed Wheel, it should always be set to zero - use tooth#1 angle instead.
  - In other modes, this setting can make a +/- 20 degree adjustment to the preset timing alignment.
- **Angle between main and return** - This is only used in CAS 4/1 mode and specifies the angular length of each tooth.
- **Odd-fire first angle** - For odd-fire engines this specifies the angle the engine must rotate from spark A output to spark B output.
- **GM HEI/DIS, TFI, C3I options** - Some distributor and related setups have a couple of extra tricks that they use:
  - Off: No additional spark outputs or cylinder identifications are used:
  - GM Bypass on Spark B: On a computer controlled GM HEI or many GM distributor-less ignitions, this setting enables the "bypass" output on spark B. The bypass output allows the module to control the timing during cranking. The ECU applies voltage to the bypass wire after startup to take control of timing.
  - Buick C3I: This uses a distributor-less ignition which behaves like a GM HEI, but also includes a cam sync signal. Used with Toothed Wheel mode instead of Basic Trigger to enable the cam sensor.
  - TFI Non-Signature: Used with Ford TFI distributors where all trigger vanes are the same length.
  - TFI Signature PIP: This allows you use the cylinder identification feature on Ford Signature PIP distributor, where one vane is shorter than the others. Signature PIP mode is used with "Basic Trigger" and allows the MS3Pro to use these distributors for sequential fuel.
- **Odd-fire phasing** - Most odd-fire engines, such as the Chevy V6, use alternate long/short angles between cylinders. Yamaha Vmax and a couple other V-twins use a paired long/long/short/short pattern. You can also define a custom pattern with four separate odd fire angles, for engines where the odd fire sequence is **really** odd.
- **Skip Pulses** - This tells the code how many input pulses it should ignore before attempting to match the input pattern. The default value of 3 is suitable for most applications. Increasing this number can filter out RPM spikes right at the start of cranking, but setting this value too high will delay startup. This setting is only used for startup and is ignored after the engine is running.
- **Ignition Input Capture** - Determines if the active edge is when the voltage on the RPM input rises or falls. This must be matched to your tach input. If it is set incorrectly you can experience timing drift with rpm or misdirection of missing teeth on a missing tooth wheel. The tooth logger can be used to confirm polarity is correct. Note that for compatibility across several different products in the MegaSquirt family, this name refers to the signal at the processor, **not** the signal coming into the ECU. The MS3Pro's input conditioner will invert the signal. So if you were using a Hall effect sensor where the voltage dropped from 12 to 0 volts, the MS3Pro will see this as a rising edge instead of a falling one. If using the Digital Frequency In pins for cam input, the signal on the Digital Frequency In pins is not inverted.

When using a VR sensor, it's important to set the correct edge, as one is more stable than the other. Here is what a typical VR signal can look like:



The MS3Pro needs to be configured for rising edge in this instance as that matches the "sharp" well defined zero crossing of the input signal. Triggering on the falling edge will result in a more unstable signal.

- **Spark Output** - This setting is **critical** or you will damage your coils or ignition modules. Going High means that the output goes high (5 volts) to dwell the coil, and then drops low to discharge the coil. Going Low means that the output is held at 5 volts, drops to 0 volts to dwell the coil, and then goes back to 5 volts to make the coil discharge. **For 99% of installs using the MS3Pro, this should be set to "Going high".**
- **Number of coils** - Specifies the type of ignition coils in use. TheMS3Pro automatically chooses the number of outputs to use based on this setting and the number of cylinders.
  - Single coil - just one coil or spark output. e.g. single coil, TFI, HEI, EDIS. EDIS actually has multiple coils, but there is only one output from the MS3Pro to the EDIS module, so it uses the "Single Coil" setting.
  - Wasted spark - one double ended coil for a pair of cylinders. i.e. 2 spark outputs on a 4 cylinder.
  - COP - coil-on-plug - one coil per cylinder fired once per cycle (typically needs a cam sensor)
  - Wasted COP - one coil per cylinder fired twice per cycle (typically does not need a cam sensor.) Coils are wired the same as COP.
  - Dual dizzy - for single plug engines with two coils and two distributors, each operating half the cylinders. This appears on some Lexus, Porsche, Jaguar, and BMW engines.
-
- **Cam input** - Normally this should be the CMP input, but you can also use Digital Frequency In 2 for Hall sensors. The MS3Pro can also use MAP sampling combined with a missing tooth wheel on single or two cylinder motorcycle engines.
- **Tertiary tach input** - Only used on Audi tri-tach inputs.
- **Flip polarity on hi-res tach** - used on Optispark and Nissan distributors to invert the signal on the ring of 360 slots.
- **Start wasted COP** - In coil on plug modes that are able to start without a cam signal, the MS3Pro will normally start in wasted spark, then transition to running fully sequential coil on plug once cam sync is declared. On some engines with very high overlap cams, starting in wasted spark can cause a backfire. Setting this option to "Wait for cam signal" will force the engine to start in fully sequential coil on plug when cam signal is established and prevent the engine from starting in wasted spark. On engines where running wasted spark is not a problem, this will delay startup and prevent the engine from starting if the cam sensor malfunctions.

- **High RPM fuel / spark delay after sync-loss** - Normally after losing sync, the MS3Pro will attempt to restart the engine as soon as it receives enough crank pulses to determine engine position. In some cases, sensor issues under load (such as a false missing tooth) can cause the engine to re-sync at the wrong timing, resulting in a backfire. Turning this option on will cause the MS3Pro to apply a complete fuel and spark cut at sync loss and require the MS3Pro to wait at four full crank rotations without losing sync before it starts firing the injectors and coils after a lost sync event. Currently, this is not available in all spark modes.
  - **RPM threshold** - Below this RPM, the ECU will restart immediately after losing sync. Should be set just above idle speed to avoid having this feature engage while cranking, where it can significantly delay startup.

The remaining settings on the left hand side are for Toothed Wheel mode, which will be covered in the specific ignition settings section. Here is what the settings on the right hand side do.

- **Fixed advance** - When set to "Fixed Advance", the spark timing is locked to the value set below. This is used for confirming timing with a strobe timing light during setup. For normal operation, this **must** be set to "Use table" so that the spark advance table is used.
- **Use Prediction** - Normally set to "1st Deriv Prediction". Alters the way the internal spark timing calculation functions. With no prediction, spark retard may be observed during engine acceleration. With 1st Deriv, this is automatically compensated for.
- **Timing for Fixed Advance** - only applies when "Fixed Timing" is set above. Timing is then locked to this specified value.
- **Cranking dwell** - specifies the base coil dwell time during cranking before battery correction is applied.
- **Cranking advance** - The MS3Pro uses a fixed timing angle while cranking, specified here.
- **Toyota Multiplex** - Used for Toyota DLI distributor-less ignitions. Known applications include the 7M-GTE and Japanese market 4A-GZE. This uses the primary spark output as a timing signal, and secondary spark outputs to specify which coil is active.
- **Dwell type** - Can be "Standard Dwell", "Fixed Duty", "Time after Spark", "Charge at Trigger" to accommodate different ignition types.
  - Standard Dwell - this is the setting to use for most installs. i.e. regular coils that are fired by a module or with internal drivers or logic-level COPs. This is also used with GM-HEI7 and GM-DIS.
  - Fixed Duty - locks the spark output to a constant duty cycle instead of controlling dwell. This is typically used for ignition modules that control the dwell directly, including Ford's "Push Start" TFI modules, Bosch "smart" ignition modules, or MSD and other capacitive discharge boxes.
  - Time After Spark - only intended for Saab Tri-Ionic DI cassettes that require a backwards dwell signal.
  - Charge at trigger - This is a highly specialized mode intended for driving a CDI thyristor directly. Do not use with multi-tooth wheels. Do not use with direct coil control or COP type ignitions as the dwell varies with rpm and advance. Do not use if you are not certain what a thyristor is.
- **Use dwell vs rpm curve** - For Saab Tri-Ionic and time-after-spark, the dwell can be varied with rpm.
- **Use Dwell Table -** Enables a 3D table of dwell as a function of RPM and load. This lets you run a low dwell under vacuum and increase dwell only under boost, to reduce heat load on the coils.
- **Nominal dwell** - The base dwell time before battery voltage correction. (With the standard correction curve, this time is applied at 14V.) Typical values are 3-4 ms.
- **Spark duration** - The time required for the spark to complete, typically in the order of 1 millisecond. Note that in older single-coil systems with many cylinders, the dwell time can be constrained at higher rpm as there isn't enough time between cylinders for the desired dwell and spark. The dwell time is automatically reduced in this case (this is the reason for stock HEI ignition being weak above 5000 rpm on V8s.) With wasted-spark or COP, there is far greater time between ignition events and this dwell-reduction rarely occurs.
- **Dwell time** - the backwards dwell time for Time-After-Spark

- **Dwell duty** - for fixed duty, this specifies the percentage duty cycle of the output signal.
- **Spark Hardware Latency** - all electronic devices take a finite time to propagate signals and switch. This setting allows for that slight delay. If you notice a slight retarding of ignition timing with rpm while strobing with fixed timing, you can enter a small number in here to remedy it.
- **Middle LED indicator** - not used on MS3Pro.
- **Spark Trim** - enables the adjustment of timing per individual cylinder. This requires a suitable trigger wheel capable of cylinder identification; modes like Basic Trigger do not allow this.
- **Kick-start crank delay** - For small engines with a kick starter, it can help to force a longer wait time on cranking. This mode will delay starting if using basic trigger with a trigger angle of less than 20 degrees, or trigger return mode, for the amount of time set by “Delay (ms)” underneath. Most other modes require the ECU to detect a larger number of pulses, which results in a built-in delay.

Now, for the settings for specific ignition modes. Note that many of the model specific modes are named after one common application of the settings, but there are often other engines that will use the same settings. For example, the “4G63” mode is not just used on the Mitsubishi 4G63, but also on other Mitsubishi engines and Mazda Miatas.

A note about the trigger wheels: Most of the trigger wheels shown are drawn as a generic disc with squared-off teeth - a common arrangement for reading with a Hall effect / gear tooth sensor reading the sensor from the edge. In some cases, the factory trigger wheel may use slits, holes, or other tooth shapes, or have a notch trigger the sensor instead of a tooth, making for an “inside out” version of the pattern. In some cases we have tried to show the shape of the teeth when the design does not look very much like the generic trigger wheel representation; in other cases this would make it difficult to show the complete pattern. When in doubt, we’ve opted for depicting the pattern clearly. The diagrams do not necessarily show the factory trigger wheel alignment, just the arrangement of trigger teeth.

5.3.0.1 Distributor rotor phasing If using a distributor, you will need to make sure the rotor is in contact with the distributor cap through the entire timing advance range. Take the average of the most and least spark advance you will run, and rotate the engine to this angle as indicated with the timing marks on the balancer (you may need to use timing tape to find this range). The distributor rotor should be pointed directly at the connection for the #1 cylinder on the cap (or the cylinder 180 distributor degrees away from it). If it is not, you will need to adjust the distributor angle to match.

5.3.1 Basic Trigger

The distributor is the traditional method of timing spark and distributing the high-tension spark voltage to individual spark plugs. Originally, this used a set of breaker points, a condenser and a single ignition coil. Most distributors feature mechanical and vacuum advance systems to match spark timing somewhere close to optimal for different operating conditions. Later systems were "breakerless" and replaced the high-maintenance points with VR, hall or optical sensors, but there is still just one tooth for each cylinder. When combined with OEM fuel injection systems, the distributor may be "locked" in that there is no advance mechanism and the timing is controlled by the computer. Any engine predating fuel injection will have the tach pickup within the distributor. With most distributors there is no cylinder identification so it is not possible to run sequential fuel; Ford Signature PIP TFI distributors, which used one short vane and the others a different length, are the exception.

This mode also covers arrangements with a single crank wheel with one tooth for every two cylinders, where there is no cam sensor used. This looks exactly the same to the ECU - other than that the signal is more accurate.

Ford TFI and GM computer controlled are specialized versions of the distributor system and will be covered in their own sections. “Basic Trigger” does not cover “dual sync” distributors, which have two pickups and use “Toothed Wheel” mode, or setups which have a distributor but use a more complicated trigger pattern than the basic “one tooth, one cylinder.” There are covered in their own sections.

Examples of ignitions that use “Basic Trigger” include:

- Ford TFI
- GM HEI (both original and computer controlled)
- Chrysler Lean Burn

- MSD distributors and Flying Magnet

In all cases, you'd wire the Crankshaft Position In wire to the distributor pick-up, and Spark A Out to an ignition module.

5.3.1.1 Input phasing In most cases you will be controlling timing with the MS3Pro; for this you need a trigger that does not vary angle with rpm or load. This is called a "locked distributor".

Early distributors such as points, original HEI, Duraspark etc, all have advance mechanisms built in. Similar distributors can be converted for use with MS3Pro:

- Remove ignition module (if present)
- Connect pickup sensor (VR, hall, optical, points) to MS3Pro.
- Remove and weld up mechanical advance mechanism.
- Remove vacuum canister.
- Use remnants of vacuum advance mechanism to achieve correct input:output phasing. You may be able to set the rotor output phasing FIRST and then rotate the baseplate to achieve the correct input phasing.

Later engines may feature a distributor in conjunction with computer controlled timing - usually these distributor are "locked" from the factory and should already have good input and output phasing. (e.g. Ford TFI, GM computer controlled, Bosch hall effect.) Align as per the factory manuals and determine how it is phased before you modify anything!

The crank angle at which the tach input triggers is of importance and needs to be configured in the MS3Pro. It can be really helpful to install timing tape on your crank pulley or temporarily mark on a range of angles. For best spark control, there are some optimal and some disallowed crank angles. A typical engine will have an operating advance range of approximately 10-50° BTDC timing (depending on engine type.) If the trigger angle is within this range of angles, you will not be able to get a full range of timing advance.

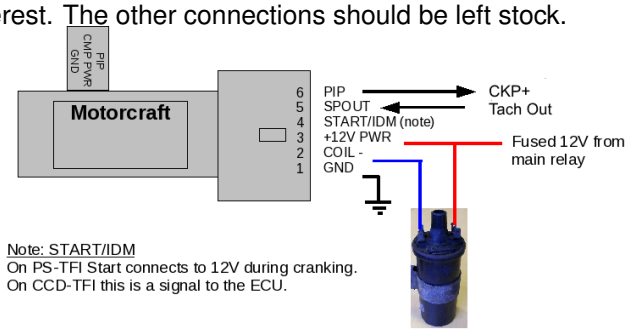
Some trigger angles work better than others. If you end up with an invalid trigger angle, adjust the distributor (the engine will often run with an invalid angle set, but will not be able to get a full range of advance).

Trigger angle range	Notes
Negative	Not allowed. Will create error message on power up.
0 - 15	Allowed. Timing cannot be retarded to less than trigger angle. Best cranking timing.
15-49	Not allowed. Engine will not be able to run on full timing range.
50-59	Allowed, but not ideal. Will not allow advance numbers past ~48 degrees.
60-90	Allowed. Provides best accuracy at high RPM and allows ATDC timing retard.
91-120	Allowed on engines with 6 cylinders or fewer. Not recommended, hard starting.
121-180	Allowed on engines with 4 cylinders or fewer. Not recommended, may be impossible to start.

For best accuracy at high revs or during transients, aim for the trigger to align at 60-90° BTDC. This also allows a full range of timing and including retarded timing should you need it for boosted conditions. This range of trigger angle is preferred for new installs. For slightly better starting, but not quite such good running accuracy, aim for a trigger ~10° BTDC or your desired cranking advance. This is the typical trigger angle for TFI and computer controlled HEI. You cannot retard timing later than the trigger angle. e.g. 9° BTDC and lower are not possible with a 10° BTDC trigger angle.

The VR sensor input presents a simple pulse as the reluctor passes the sensor, this gives a timing position easily identified by eye. Rotate the engine to 60° BTDC (or 10° BTDC if chosen) and then align the distributor so the reluctor aligns with the center of the sensor.

5.3.1.2 Ford TFI distributors Ford's TFI module was used throughout the 1980s and into the 1990s on many millions of vehicles in two main mounting positions - "distributor mount" and "remote mount". There are also two electrical versions: "Push Start" and "Computer Controlled Dwell" : documentation claims that these can be distinguished by color, but that appears unreliable. Checking the wiring on pin4 is likely best. The wiring of the modules is largely the same, just the distributor mount connects directly to a 3 wire hall sensor in the distributor. In most installations you do not need to concern yourself with that as only the "PIP" and "SPOUT" connections are of interest. The other connections should be left stock.



The MS3Pro Mini does not have a 12v ignition output needed to drive the TFI module directly and thus requires an AXM-130 adapter that can use a logic-level 5v spark signal as in input and provide the correct 12v square wave output to the TFI. The AXM-130 dual-channel adapter is available here: AXM-130 Driver

Be sure "Spark hardware in use" is set to Tacho. CKP- is not connected - tape this wire off; do not ground it.

The module described mainly here is the "Push Start" type that uses a 12V start signal. While some documentation claims that all "PS" type modules are gray, we've seen them in black as well. A more reliable indicator is that pin 4 will be connected to a 12 volt source when cranking. 50% dwell duty should be used. On the "CCD" type, pin 4 runs as a diagnostic signal to the original ECU. These modules need normal dwell control instead of a fixed duty. We recommend starting with 3.0 ms dwell and adjusting to match the coil used. Other wiring should be the same.

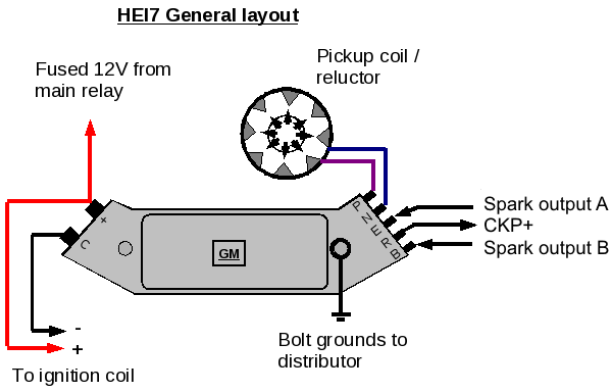
"Base Timing" on the distributor (with computer control "SPOUT" disconnected) is around 10BTDC. This is the number you should use as your initial Trigger Offset, and adjust as needed to obtain correct timing. As these distributors were designed for ECU control, the rotor arm phasing should already be correct.

The "Signature PIP" versions of the TFI distributor have a shorter vane used to identify the #1 cylinder. These appeared on factory sequential injected engines. These are still set up as basic trigger, but you will set "TFI Signature PIP" under Ford TFI Options.

Set "Ignition Input Capture" to "**Falling Edge**" and Spark Output to "**Going High**".

5.3.1.3 GM HEI The original "High Energy Ignition" (HEI) distributors used the 4 pin module from the early 1970s. That module has a decently hot spark, but doesn't support computer controlled timing. The later 7 and 8 pin modules and corresponding distributors are designed for computer control and should be an easy swap onto earlier engines - not only are those modules intended for computer control, but their distributors are already locked-out so no modifications are required. 7 and 8 pin HEI distributors use three control wires to/from the MS3Pro. The other wires connect to the coil or vehicle wiring.

- The 'Ref' signal from the module to the MS3Pro gives rpm and engine position information. It connects to the white wire on the CKP In line.
- The 'Est' signal from MS3Pro to the module controls the advance when running. It connects to Spark A Out.
- The 'Bypass' signal from MS3Pro to the module allows the module to beneficially control its own advance during cranking. Once the engine has been running for more than 5 seconds, the MS3Pro takes control of timing. It connects to Spark B Out.



To enable the bypass, set GM HEI / DIS options to "**GM Bypass on SpkB**". Set Ignition Input Capture to **"Rising Edge"** and Spark Output to **"Going High"**. Set trigger angle 10 and adjust as needed.

5.3.1.4 MSD distributor based ignitions The MS3Pro Mini does not have a 12v ignition output needed to drive the MSD module directly and thus requires an AXM-130 adapter that can use a logic-level 5v spark signal as in input and provide the correct 12v square wave output to the MSD. The AXM-130 dual-channel adapter is available here: AXM-130 TFI Driver

5.3.2 Trigger Return

Trigger Return is a special case of Basic Trigger which can be used if you have a Hall effect distributor or optical distributor where all the slots, shutters, or vanes are equally sized. The purpose of the "Trigger Return" mode is to have accurate cranking timing as well as accurate running timing. It achieves this by using the signal from both edges of a vane/slot. One edge is used for the timing calculations during running and will typically pass the sensor at 55° BTDC or more. The other edge is used for cranking timing and must pass the sensor at the desired cranking advance angle e.g. 10° BTDC.

Examples of ignitions that use Trigger Return include:

- Many '80s era water cooled VWs
- Volvos with Bosch LH2.2 ignition

5.3.3 Fuel only

When the MS3Pro is not controlling ignition, you will need to send it a tach signal on the CKP In white wire. Examples of acceptable signals that you CAN use to trigger the MS3Pro include:

- Electronic distributor pickups
- The tach output terminal of an MSD or other capacitive discharge ignition box
- **IMPORTANT NOTE: The negative terminal of the coil CANNOT be used, as the coil flyback voltage is too high and WILL DAMAGE the MS3Pro circuitry!!!**

Never connect any wires from the MS3Pro ECU to the coil if using an MSD or other capacitive ignition box, for any reason. When MSD says "These are the only wires that should be connected to the coil," they mean it.

5.3.4 Toothed wheel

The Trigger Wheel mode is designed to support most combinations of regular missing tooth wheels with or without a cam signal. Other ignition modes are used to support irregular or OEM specific wheel patterns.

The table below lists all of the valid combinations for trigger wheel. Some of these are very rare on production engines, but the code will allow using these if you encounter (or fabricate) such a trigger wheel arrangement.

Note - this table is for four-stroke piston engines. Two stroke or rotaries only need 360 degrees of information for full sequential and COP.

Trigger wheels		Sequential level			Settings		
Main wheel	Secondary wheel	Distributor only	Semi sequential	Full sequential	Trigger wheel arrangement	Main wheel speed	2nd trigger active on
Missing tooth on crank	None	Y	Y	N	Single wheel with missing tooth	Crank	n/a
Missing tooth on cam	None	Y	Y	Y	Single wheel with missing tooth	Cam	n/a
Missing tooth on crank	Single tooth on cam	Y	Y	Y	Dual wheel with missing tooth	Crank	n/a
Non missing tooth on crank	Single tooth on crank	Y	Y	N	Dual wheel	Crank	Crank
Non missing tooth on crank	Single tooth on cam	Y	Y	Y	Dual wheel	Crank	Cam
Non missing tooth on crank	Cam wheel with tooth per cylinder	Y	N	N	Dual wheel	Crank	Every Cylinder
Non missing tooth on cam	Single tooth on cam	Y	Y	Y	Dual wheel	Cam	Cam
Non missing tooth on cam	Single tooth on crank or two opposite teeth on cam	Y	Y	N	Dual wheel	Cam	Crank
Non missing tooth on cam	Cam wheel with tooth per cylinder	Y	N	N	Dual wheel	Cam	Every Cylinder

Distributor only: Can only use a distributor and batch fire fueling.

Semi-sequential: Supports wasted spark or wasted-COP ignition and semi-sequential fueling.

Full sequential: Supports sequential COP and sequential fueling.

For initial setup and determining tooth#1 angle on uncommon setups, having timing marks or tape on your crank pulley/damper covering the full 360 degrees will be greatly helpful. Speed shops sell timing tape for a variety of damper diameters. If your engine has no timing marks, you do need to add them. Just guessing at timing is a great way to damage an engine. **Running excessive timing under load will almost always cause severe engine damage, e.g. broken pistons. It is essential that timing is confirmed with a timing-light on EVERY install.**

The most common trigger wheels that work under the generic “Toothed Wheel” mode that you are likely to encounter on production engines are:

- 36-1 on crank with no cam sensor - early Ford distributor-less ignition
- 36-1 on crank plus single tooth cam sensor - many later Ford distributor-less ignitions
- 60-2 on crank with no cam sensor - many vehicles with early Bosch Motronic engine management: BMW, VW, Audi, Volvo, Vauxhall, Opel, Peugeot etc.
- 60-2 on crank plus single tooth cam sensor - Bosch Motronic systems with sequential injection or distributor-less ignition
- 60-2 on crank plus four tooth cam sensor - Bosch Motronic or GM 58X systems with variable valve timing

- 24 tooth on cam - many Japanese originated vehicles use the Nippon Denso 24 tooth CAS with differing numbers of 2nd trigger teeth and sensors.

Engines with a missing tooth crank wheel plus a cam signal, and an even number of cylinders, are able to start in semi-sequential mode until a cam signal is detected, at which point they switch to fully sequential operation. V10 and V12 sequential applications only sync once they have a cam signal detected.

5.3.4.1 Terminology notes

- **Missing tooth** - This is a regular wheel with a group of "missing" teeth e.g. 12-1, 36-1, 36-2, 60-2
- **on crank** - the wheel is rotating at crank speed, normally directly attached to the crank pulley or flywheel
- **on cam** - the wheel is rotating at camshaft or distributor speed
- **Single coil** - a single coil and distributor
- **Wasted spark** - double ended coils (or a pair of coils) that fire twice per cycle
- **Wasted-COP** - a single coil per cylinder, but firing twice per cycle. Each coil is controlled by a separate wire. Useful for testing out a dual wheel arrangement when you are uncertain of the cam phase.
- **COP** - a single coil per cylinder that fires once per cycle
- **Batch/bank fire** - groups of injector fired at once, not timed to a specific cylinder event
- **Semi-sequential** - injectors fired twice per cycle timed to cylinder events
- **Sequential** - each injector fires once per engine cycle timed to a specific cylinder event

5.3.4.2 Wheel naming There does not appear to be universal agreement on the way to name wheels among manufacturers. In the MS3Pro documentation, we use a standard where a minus sign indicates missing teeth, a slash indicates a separate wheel, and a plus indicates a tooth added to the same wheel. Here are some examples of how this naming convention works.

- 36-1. This means a single wheel with place for 36 teeth and a single tooth omitted. i.e. 35 teeth at 10 (360/36) degree spacing.
- 36-2. This means a single wheel with place for 36 teeth and two adjacent tooth omitted. i.e. 34 teeth at 10 (360/36) degree spacing.
- 36-1-1. This means a single wheel with place for 36 teeth and two non-adjacent single tooth omitted. This type of wheel is not supported by this generic wheel decoder. It requires a model specific decoder to specify the separate gaps
- 36-2-2-2. This means a single wheel with place for 36 teeth and three sets of double missing teeth. This type of wheel is not supported by this generic wheel decoder. It is supported as 36-2-2-2 with the specific OEM pattern required.
- 24/1. This means 24 teeth (non-missing) on one wheel and a single tooth on a second wheel.
- 36-1/1. This means a one 36-1 wheel and a single tooth on a second wheel.
- 3+1. This means one wheel with 3 equally spaced teeth and an additional tooth to indicate sync. Not supported by the generic wheel decoder.

5.3.4.3 Specific settings This covers just the settings used in Toothed Wheel mode; settings that are common to all wheel decoders are covered under “Ignition Settings.”

- Spark Mode - set to "Toothed Wheel"
- Trigger Angle/Offset - always zero in Toothed Wheel mode. You’ll use the Tooth #1 Angle instead.
- Trigger wheel arrangement - see table above for correct settings
- Trigger wheel teeth - the number of effective teeth, counting the missing teeth as if they existed. i.e. a 36-1 wheel has 35 physical teeth, but enter 36, as you’re including the teeth that would populate the gap as well.
- Missing Teeth - the number of missing teeth. Common are 1 for 36-1, or 2 for 60-2 or 36-2
- Tooth #1 angle - definition depends on whether main wheel is missing or non-missing type. See sections below.
- Main wheel speed - indicates whether the main wheel rotate at crankshaft speed or camshaft (distributor) speed.
- Second trigger active on - Like ignition input capture above, specifies which voltage level is considered "active"
- Level for phase 1 - only applies in "Poll level" mode. See Dual+Missing section.
- Check at tooth # - Determines which tooth on the crank trigger is used for polling. See Dual+Missing section.
- and every rotation of - how often are second trigger input pulses received. See Dual Wheel section

There are two main categories of install - Existing and Retrofit.

5.3.4.4 Existing In this cases where you are fitting MS3Pro to an engine already fitted with a trigger wheel, your main task is to wire up the sensor(s), determine the tooth #1 angle and wire up your coil(s). It should not normally be necessary to alter the engine.

5.3.4.5 Retrofit If you have an engine that did not originally come equipped with a trigger wheel (e.g. a distributor based, pre-EFI engine) then you have to mount a wheel and sensor and set the phasing correctly. For a typical car engine - go for a 36-1 wheel on the crank for non-sequential. or a 36-1 wheel on the crank and a 50/50 cam tooth with gear-tooth hall sensor for full sequential 60-2 works great on most engines too, but is not advised for very high rpm. For very high revving engines (such as motorcycle engines) due to the number of teeth per second, 36-1, 24-1 or 12-1 are preferred. There are several restrictions on base tooth count:

- No more than 60 base teeth.
- The angle between teeth in crank degrees must be an integer number or in increments of 0.1. So 36 teeth (10 degrees per tooth) is allowed; 32 teeth (11.25 degrees per tooth) is not.
- The ECU must read at least one real tooth per spark event. So cutting off one tooth of an 8 tooth distributor to get cylinder identification on a V8 is out - the ECU would need to schedule two spark events off the tooth before the missing tooth, which is allowed. Putting that modified distributor on a four cylinder, on the other hand, is OK. So is adding a second sensor to the distributor reading a separate, single tooth trigger wheel on a V8 (the common aftermarket dual sync distributor).
- The number of base teeth per spark event must be the same for each spark event. A four tooth crank wheel with a one tooth cam trigger can run a V8, but the same trigger wheel on a six cylinder would result in four cylinders having to use one tooth each and two cylinders having to use two teeth each, which is not allowed.

While the code can cope with any sensor/tooth phasing, during cranking the rpm will vary up and down greatly as the engine rotates. It is desirable to place the missing tooth such that it passes the sensor when the engine is somewhat stable. The OEMs have found that certain tooth #1 angles work well and it is worth following their lead.

We suggest to align your wheel and sensor to arrive at the following tooth #1 angles. (See later for explanation of tooth #1.) These will make it easier for the ECU to detect the missing tooth during cranking.

- 4 cylinders ~90-120°
- 6 cylinders ~50°
- 8 cylinders ~40°

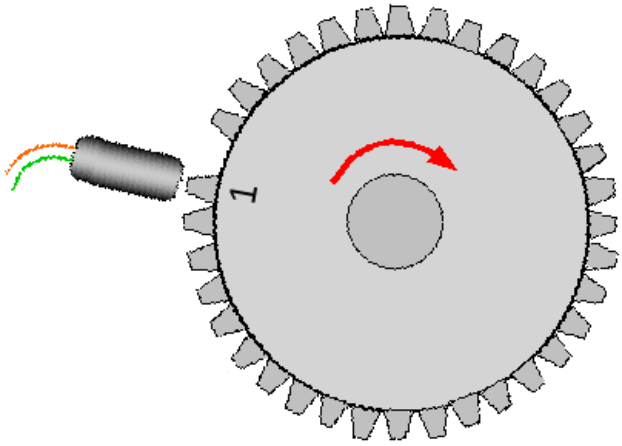
Mounting the wheel is quite critical in that it MUST be mounted so it rotates without moving up, down, left or right, as a typical sensor needs to see all of the teeth with a gap of 0.75 - 1.0mm. In other words, keep the runout to a minimum. The gap shouldn't get change size as the wheel rotates. It is also critical that the bracket is sturdy enough not to vibrate. When in doubt, make the bracket strong enough that you could lift the engine with it.

Having mounted the wheel and sensor, you can proceed for an existing install.

5.3.4.6 Missing tooth crank wheel This is a very common configuration for wasted spark with the most typical wheels being 36-1 (Ford) and 60-2 (Bosch.) Note that the missing teeth are in a single group - if your wheel has multiple groups then you need a special wheel decoder. We have special decoders for many of the multiple-gap trigger wheels, such as Chrysler's NGC or Subaru's 36-2-2-2 wheels. If you have one of these wheels, see the spark mode sections be

The MS3Pro code benefits from a reasonable number of teeth (hence 36 or 60) for best ignition timing accuracy. Low tooth count wheels such as 4-1 are not advised.

Make sure you understand the definition we're using of tooth #1. With the engine rotating in the normal direction, **tooth #1 is the first tooth to pass the sensor after the missing tooth gap.**

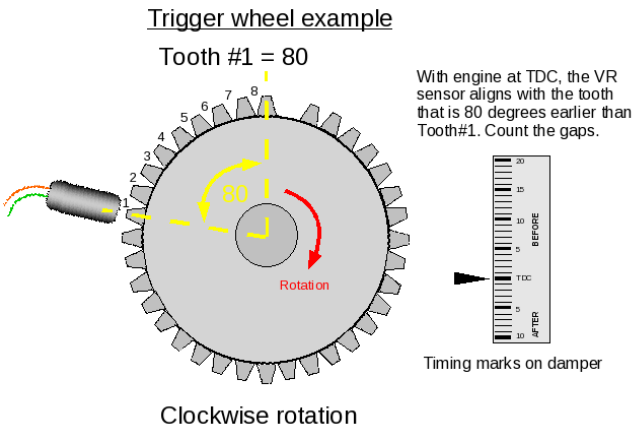


We use the term "tooth #1" as it is consistent across wheels with one, two, three or four missing teeth in the group. Once the code knows the tooth #1 angle it automatically calculates other needed information internally.

Clockwise rotation (normal) - method A

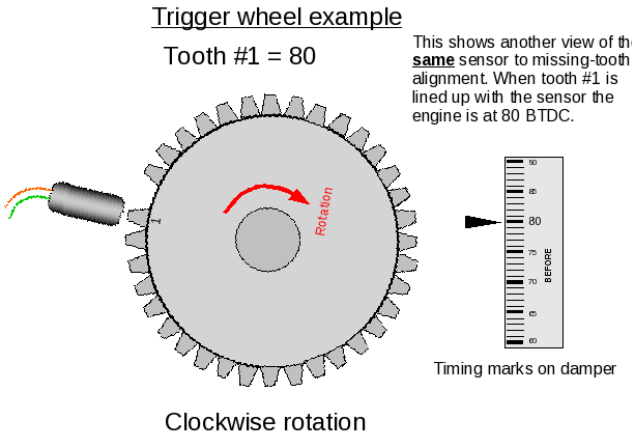
Set your engine at TDC, then count the number of **gaps** to tooth #1 in the direction of rotation (clockwise here), and multiply by the angular size of the tooth: e.g. 8 teeth * 10° per tooth = 80°

- 36-1 wheels are 10° per tooth (360° / 36 teeth)
- 60-2 wheels are 6° per tooth (360° / 60 teeth)
- 24-2 wheels are 15° per tooth (360° / 24 teeth)



Clockwise rotation (normal) - method B

A different way of looking at the **same** phasing. Turn your engine so that tooth #1 aligns with the sensor. Read off the tooth#1 angle from timing marks/tape on the crank pulley.



Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Single wheel with missing tooth
- Trigger wheel teeth = number of teeth including missing teeth
- Missing teeth = set according to your trigger wheel
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Crank wheel

Typical settings:

- Ford 4 cylinder = 36-1, 80° tooth #1
- Ford 6 cylinder = 36-1, 50° tooth #1
- Ford 8 cylinder = 36-1, 40° tooth #1
- Bosch 4 cylinder (Peugeot, Vauxhall) = 60-2, 114° tooth #1
- Bosch 6 cylinder (many BMWs) = 60-2, 89° tooth #1

A note about VR sensors and missing tooth wheels If using a variable reluctor type sensor with a high tooth count (usually 24 or more teeth), missing tooth wheel, you may experience a loss of sync at high RPM. If this happens, the first thing to try is to put a 10K resistor in parallel with the VR sensor; this can often improve operation with this particular input.

5.3.4.7 Missing tooth cam wheel This arrangement is not commonly used by OEMs, but does support full sequential with a single wheel and sensor. Cam triggering is less accurate than crank triggering due to timing belt or chain stretch.

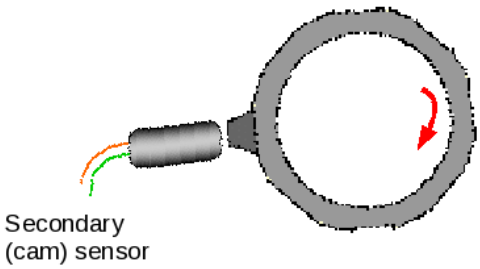
The MS3Pro code benefits from a reasonable number of teeth (hence 36 or 60) for best ignition timing accuracy. Low tooth count wheels such as 8-1 are not advised.

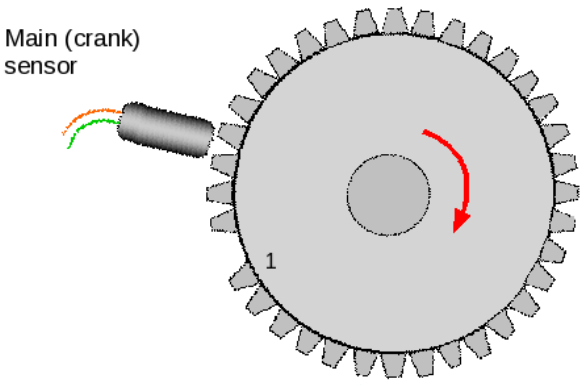
The previous section on missing tooth crank wheel generally applies when the wheel is mounted to the cam, but remember that one rotation of the cam is 720 crank degrees. The settings are in crank degrees. So a tooth #1 that is 8 gaps earlier than the sensor on a 36-1 wheel would give a 160° tooth #1 angle (8 * 10 * 2 [for cam]) .

Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Single wheel with missing tooth
- Trigger wheel teeth = number of teeth including missing teeth
- Missing teeth = set according to your trigger wheel
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Cam wheel

5.3.4.8 Missing tooth crank wheel and single tooth cam wheel This is an very common arrangement that supports full sequential and coil on plug. (For 50/50 type polled cam wheels, see the next sub-section.)

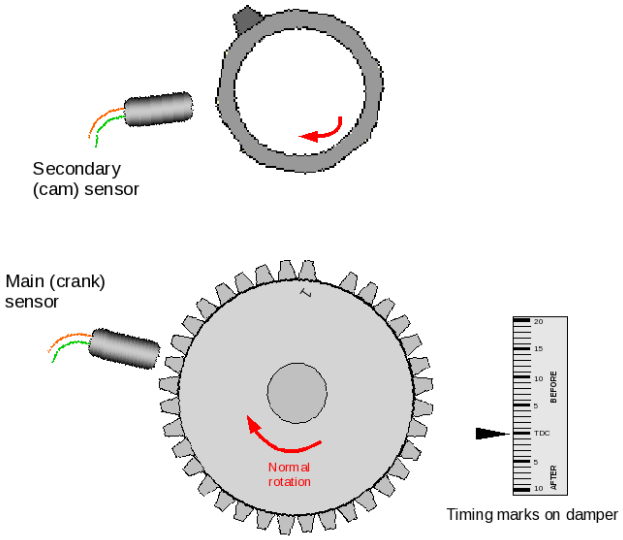




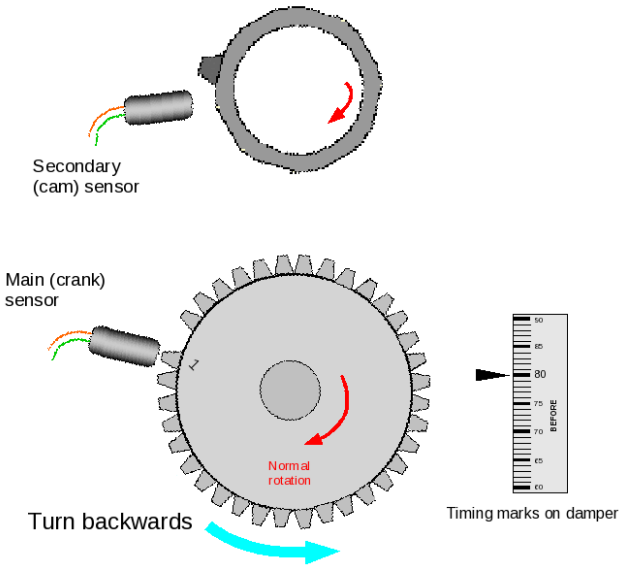
The definition of tooth#1 is the same as the basic missing tooth crank wheel and should be phased in the same way. Ensure you also read the section above. The cam input tells the code which engine cycle/phase it is on. From the crank wheel alone, the code knows when cylinder one is at TDC, but it cannot distinguish TDC compression or TDC exhaust. The cam sensor adds this information, which is why it needs to be one pulse only per engine cycle.

The cam signal is a single pulse, usually generated by a narrow tooth, vane, or window. Technically, it is edge triggered. For VR type sensors, the edge setting will depend on the wiring you use (typically Rising) as the signal is a very short pulse. For a hall type sensors you need to ensure that the edge you choose matches up with the phasing as below. To confirm correct cam sensor phasing proceed as follows.

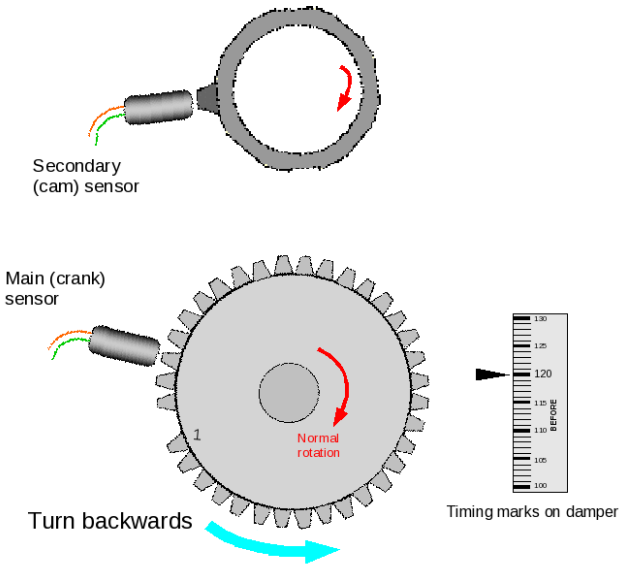
First, set your engine at TDC compression #1.



Now rotate the engine backwards to tooth #1. The angle read off the damper is the tooth#1 angle.



Now rotate the engine backwards some more, at least one or two teeth - this is the best place for the cam tooth to pass the sensor, if you are retrofitting a trigger setup. However, the cam sensor can be at any angle **except** having the active edge aligned in the tooth gap.



Critical settings include:

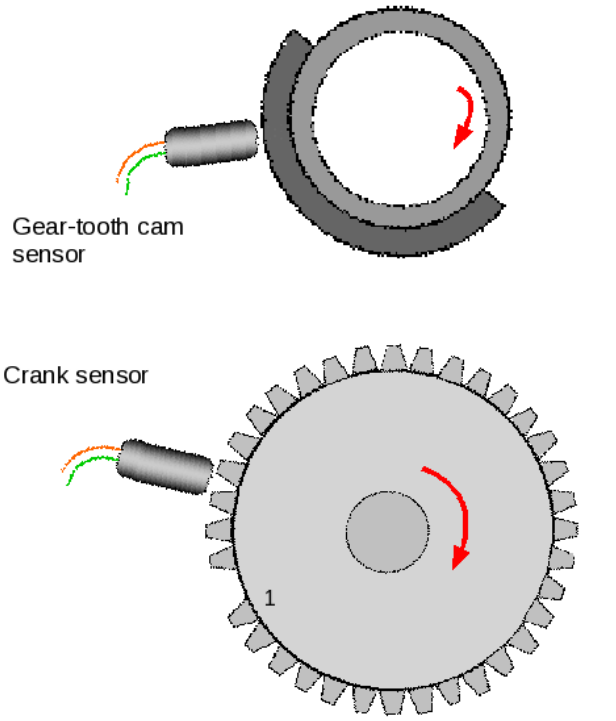
- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel with missing tooth
- Trigger wheel teeth = number of teeth including missing teeth
- Missing teeth = set according to your trigger wheel
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Crank wheel
- Second trigger active on = Set as above

The “and every rotation of” setting is grayed out as this mode only allows the second trigger to be on the cam wheel.

This wheel decoder mode has a limp home feature: If the camshaft position sensor works OK during startup, but the MS3Pro subsequently loses the cam wheel, it will “remember” what phase the cylinders are on, and keep running as if nothing happened.

5.3.4.9 Missing tooth crank wheel and polled (50/50 or half moon) cam wheel This is an fairly common arrangement that supports full sequential and coil on plug. Here, a missing tooth wheel is used on the crank in the common way, and a hall-effect or gear tooth sensor is used on the cam with a long tooth or window or vane. The MS3Pro first checks for a missing tooth in the crank wheel sensor, then checks the voltage coming from the cam signal to determine if the cylinder is on the compression or exhaust stroke, without waiting for the cam sensor to reach an edge. This gives you the ability to have full sequential, but the engine syncs up as fast as a regular missing tooth crank wheel without the cam sensor. Different OEM implementations exist - some engines use a 50/50 cam pattern, Vauxhall red-top engines use a window in the distributor rotor that spans the missing tooth region. As far as the code is concerned, these are equivalent because it only “looks at” (polls) the cam just after the missing tooth.

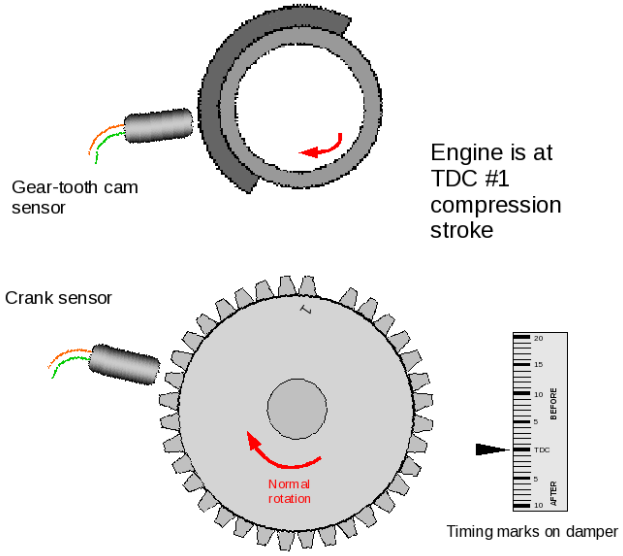
While this code normally checks at the missing teeth, the “Check at Tooth #” setting allows for it to delay this checking and look for a pulse on a later tooth. By default, this is set to tooth zero for the missing tooth. Some engines use multiple teeth on the cam wheel, with one tooth present on one rotation of the crank but not the next but no cam teeth over the missing tooth on the crank wheel. In this case, you can set “Check at Tooth #” to a crank tooth that lines up with a cam tooth on one rotation but not on the next rotation.



The definition of tooth #1 is the same as the basic missing tooth crank wheel and should be phased in the same way. The cam input tells the code which engine cycle/phase it is on. From the crank wheel alone the code knows when cylinder one is at TDC, but it cannot distinguish TDC compression or TDC exhaust. The cam sensor adds this information which is why it needs to be one "tooth" only per engine cycle.

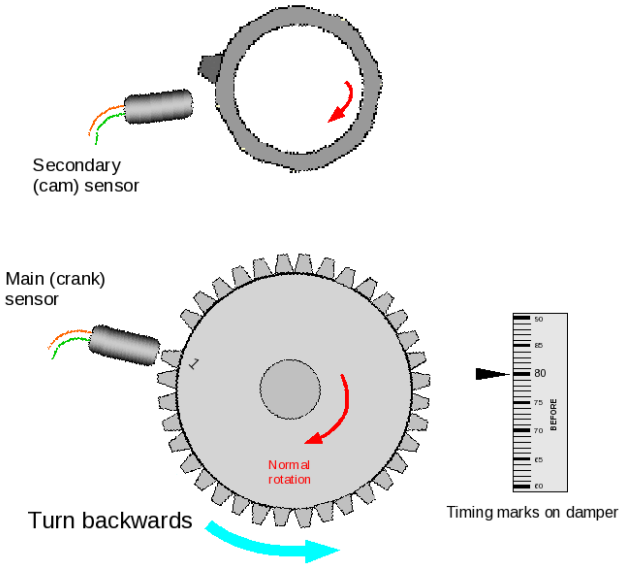
At close to tooth #1 the code examines the voltage level on the input to determine which phase it is on - the "tooth" should be normally start at least 20 crank degrees before tooth#1 and continue for another 20 crank degrees afterward. (The level is actually polled at tooth#2.)

To confirm correct cam sensor phasing proceed as follows. First, set your engine at TDC compression #1.

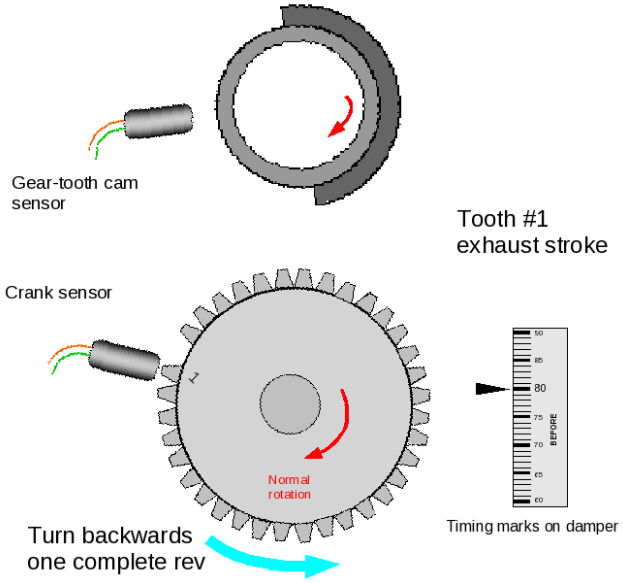


Now rotate the engine backwards to tooth #1. The cam sensor should be roughly in the middle of window/tooth/vane. With the cam sensor powered and connected to the MS3Pro, measure the output voltage. When

using the standard cam input, a voltage of ~0V here requires the HIGH setting and a voltage of ~5V here requires the LOW setting, as the MS3Pro input conditioner inverts the signal.



Now rotate the engine backwards a full revolution. The cam sensor will be opposite that previous window/tooth/vane. (If there was a window before it must be a vane now and vice versa.) You can use a volt meter to confirm the output has changed its state.



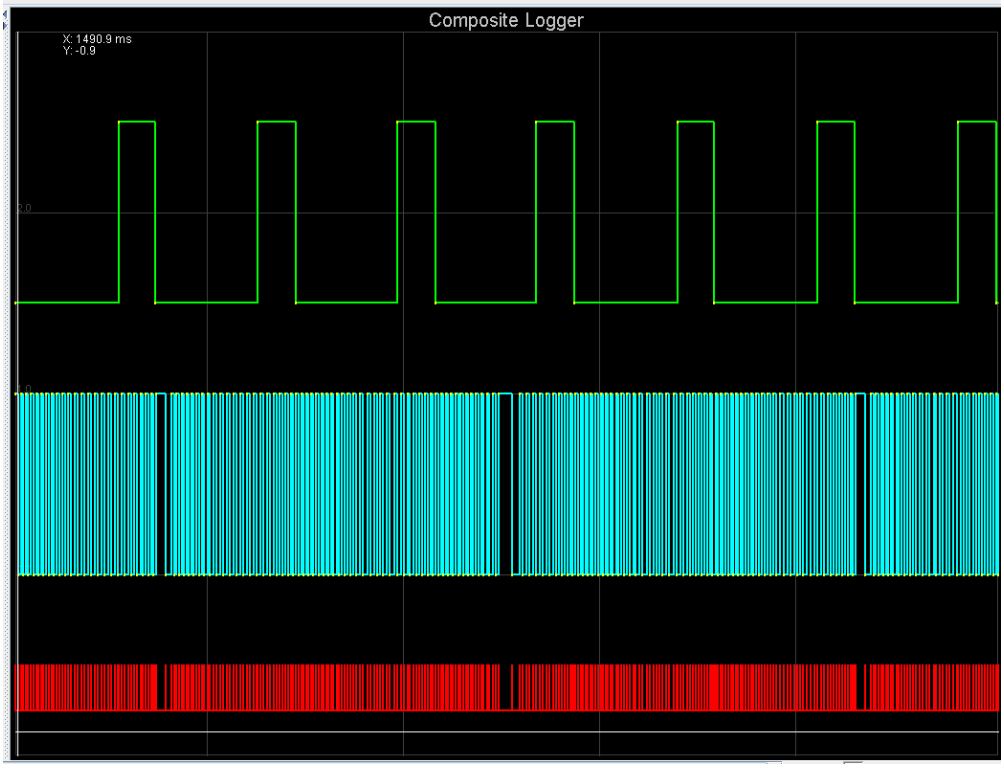
Example of a distributor converted to a cam sensor, using a commonly available gear-tooth sensor.

Critical settings include:

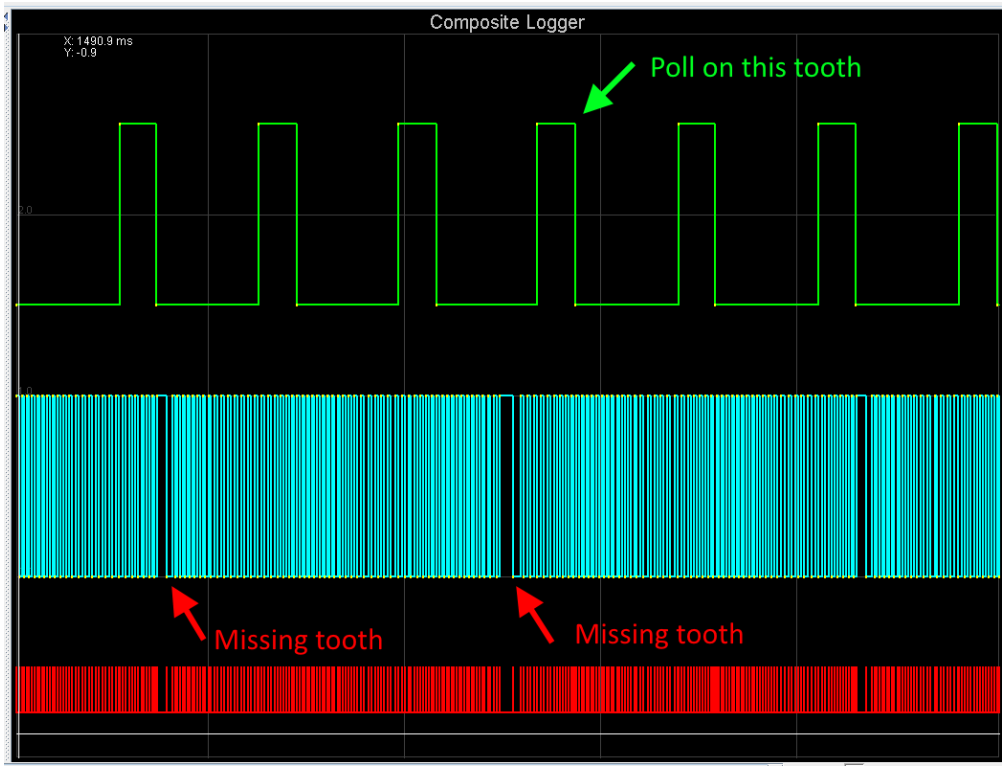
- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel with missing tooth
- Trigger wheel teeth = number of teeth including missing teeth
- Missing teeth = set according to your trigger wheel
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Crank wheel
- Second trigger active on = Poll Level
- Level for phase 1 = set as above
- Check at tooth # = which tooth after the missing tooth to use to poll the cam sensor. Leave at 0 to check when the missing tooth is present.

5.3.4.10 Missing tooth crank wheel and irregular cam wheel Some engines use a basic missing tooth crank wheel with a complex cam wheel. Some of these have a hard coded ignition mode (Toyota 2JZ-VVT and Zetec VCT), while others use the toothed wheel mode with a hard coded VVT decoder. If you do not have a hard coded decoder, you can use the Poll Level feature to identify engine phase by finding a tooth that triggers on one engine rotation but not on another.

To get the Poll Level settings figured out, start by setting the ignition mode to Log Crank and Cam, and make a composite log while cranking the engine (See section 8.4.5 for an explanation of how composite logs work). This will show both edges of both the cam and crank triggers. Here is an example from an inline 5 cylinder that originally had a distributor wheel with 5 teeth. The engine builder wished to add a 60-2 crank wheel to support sequential injection without making any modifications to the distributor trigger. The Poll Level mode makes this possible. Here is a composite log showing two rotations of the engine. The blue line represents the crank sensor, and the green line represents the cam (distributor) sensor. The red lines are not used in this test.

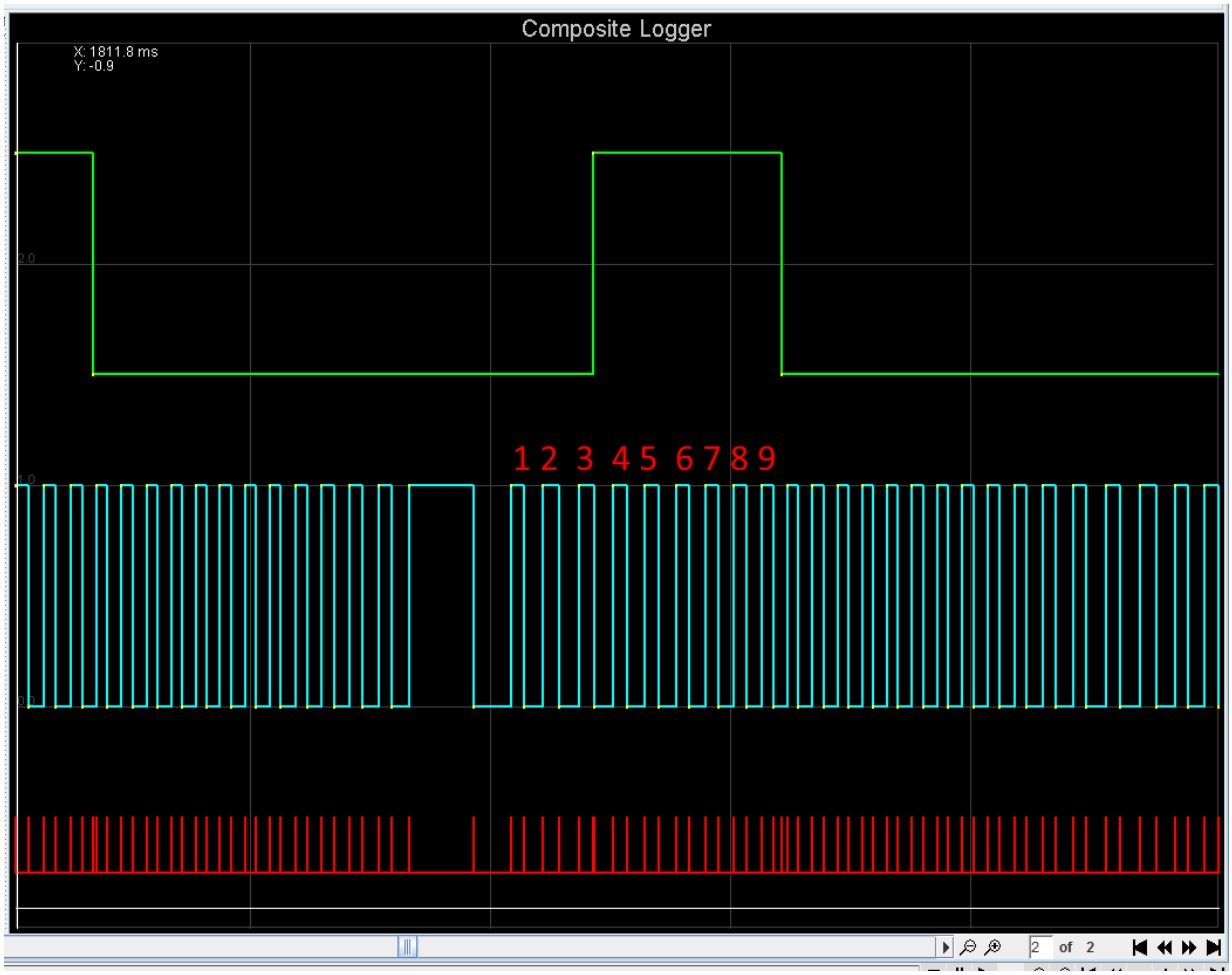


The first step is to identify the missing teeth, marked with red arrows in the image below. The next step is to find a tooth in the green line that triggers on one cylinder rotation but does not trigger on the rotation before it. The wider the tooth, the better, as it will span more crank degrees. In this example, we'll try to poll on the first cam pulse to arrive after the missing tooth (marked with a green arrow). Notice that the cam sensor signal is high for the entire length of this tooth, but low for the entire duration of the tooth on the rotation before it.



Next, we'll zoom in on that arrow and count how many crank teeth after the missing tooth there are to the center of the tooth that we're using for the cam sync.

minor changes to the decoder.



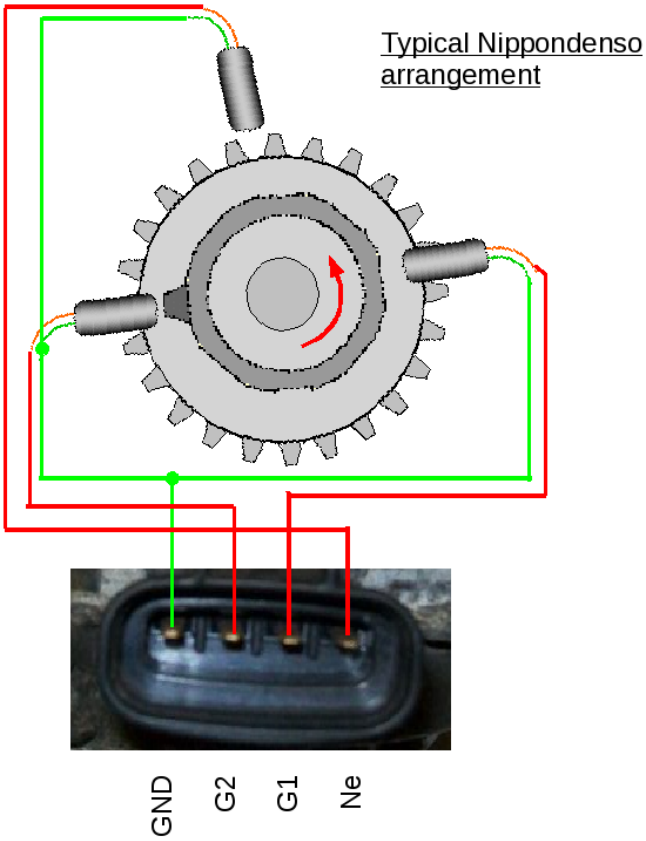
Here, we see that the cam pulse is high on teeth 4 through 9. We'll pick tooth 6, as it is closest to the center and will be least affected by any slop in the cam or distributor drive. So we would use the following settings:

- Second trigger active on: Poll Level
- Level for phase 1: High (because the cam signal is high in this log; selecting "Low" will move the tooth #1 angle by 360 degrees)
- Check at tooth #: 6

By checking the cam signal only on one specific tooth, we are able to make the MS3Pro run this engine sequentially without the need to modify the distributor trigger or write a custom wheel decoder.

5.3.4.11 Nippon Denso CAS The Nippon Denso CAS (crank angle sensor) comes in a number of versions, which all appear to use a 24 tooth main wheel and a second wheel with one, two, three or four teeth. There is a single sensor (called Ne) pointing at the 24 tooth wheel and one (G1) or two (G1 and G2) sensors pointing at the second wheel. This style of CAS is very common on Toyota and Mazda engine from the 1980s and 1990s. Honda also used a variant incorporating a 24 tooth, 4 tooth, and 1 tooth wheel into the distributor. The number of teeth on the second wheel determines whether it can be used (without modification) for single coil distributor, wasted spark or coil-on-plug (COP) and sequential. The version with a single tooth and two pickup sensors is intended for sequential. The two sensors are used by the OEM to allow the engine to synchronize within one engine revolution. Presently, the MS3Pro only supports using one of the "G" sensors.

Some Toyota JZ series and Honda H series engines used a variation of this method, with a 12 tooth crank wheel replacing the 24 tooth cam wheel. Other than the greater accuracy of a crank trigger, this makes little difference from the ECU's perspective.



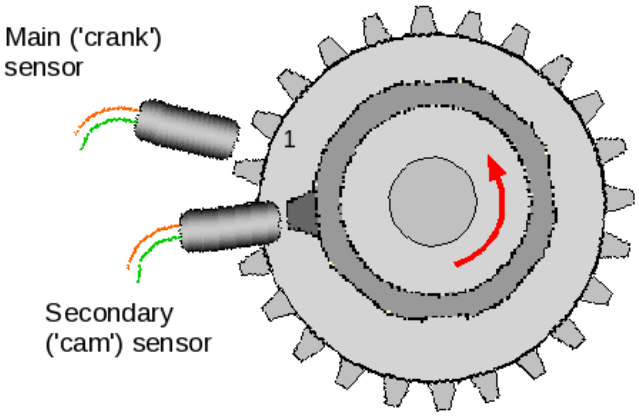
Here are the connections used.

CAS connection	MS3Pro wire
GND, NE-, or G-	CKP- and CMP-
NE	CKP+
G1 or G2	CMP+
Other G sensor	(not used)

Note that you **must** connect **both** the CKP- and CMP- to the ground wire if there is only one ground. The MS3Pro's input circuit uses a differential input, and both circuits must be at the same reference volts to use this mode.

5.3.4.12 Non-missing tooth cam wheel with single-tooth cam This arrangement is one of the variants used in Nippon Denso CAS. This also covers the common "dual sync" aftermarket distributors, AEM's Engine Position Module, and a few other aftermarket distributor or CAS setups.

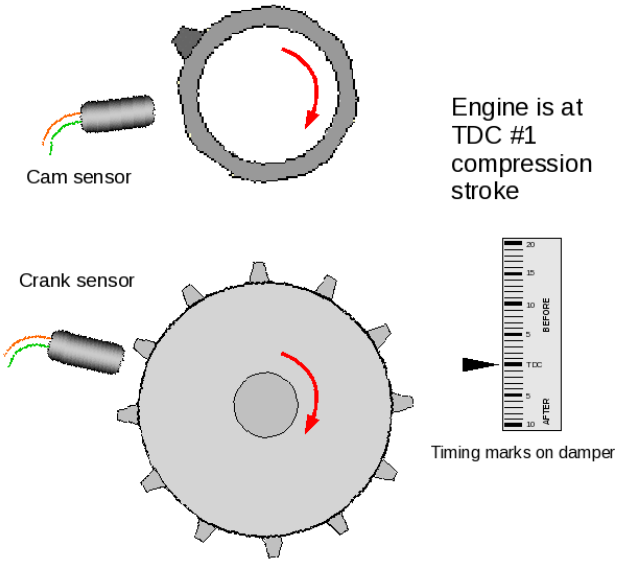
With the single tooth every 720 degrees this setup gives enough engine information for full sequential fuel and spark.



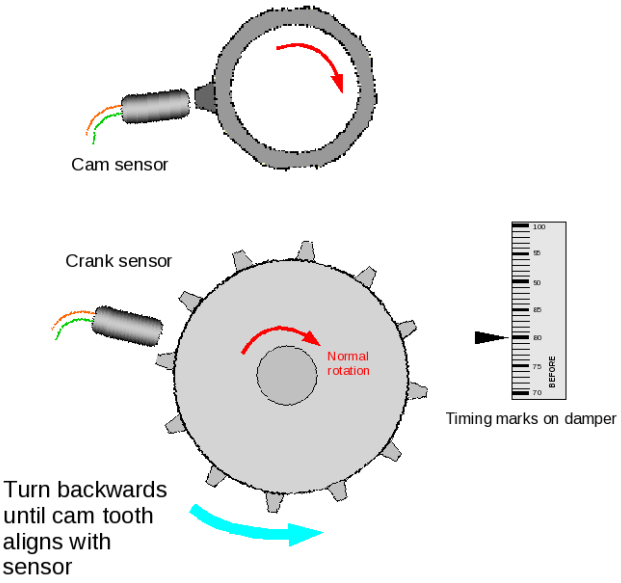
Make sure you understand the definition we're using of tooth #1. With the engine rotating in the normal direction, **tooth #1 is the first tooth to pass the main sensor after the single tooth has passed the second sensor.**

Make sure these do not happen at the same time - in the diagram you can see that the main sensor is over a gap when the secondary sensor is aligned with its tooth. If they do happen at the same time, you can probably chose the triggering edges so the MS3Pro does not "see" the two edges at the same time. Here is how to set this one up.

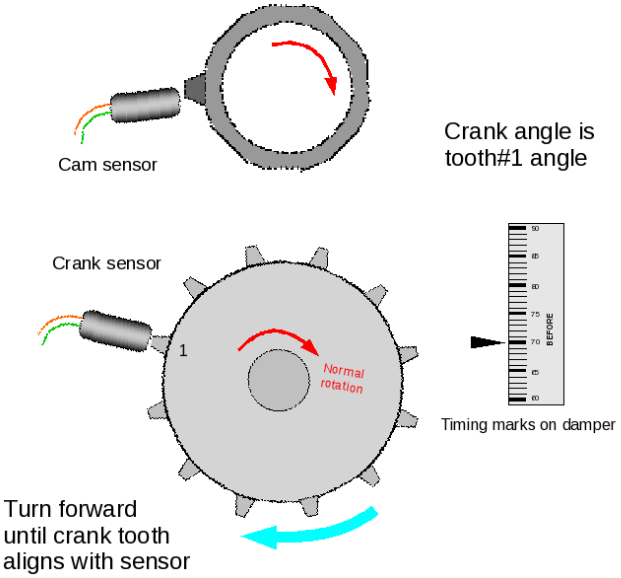
First, set your engine at TDC compression #1.



Now rotate the engine backwards until the "cam" sensor and tooth line up. If you rotated more than one turn, then add 360 to your tooth#1 angle.



Now rotate the engine forwards until the next "crank" tooth aligns with its sensor. The crank angle now is the tooth #1 angle. (Note that angles shown in diagram are examples only)

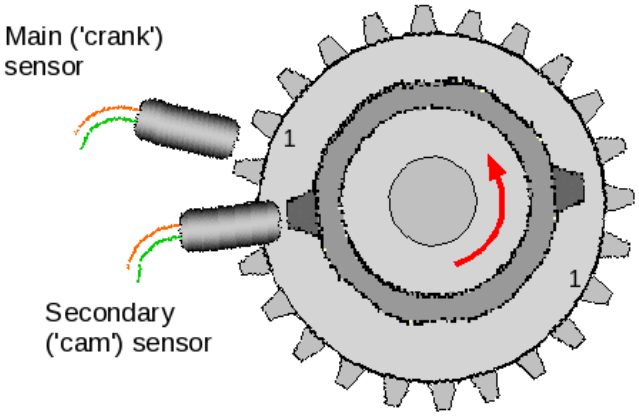


Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel
- Trigger wheel teeth = number of teeth
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Cam wheel
- Second trigger active on = Set to make sure this does not overlap with main wheel
- and every rotation of = Cam

5.3.4.13 Non-missing tooth cam wheel with two opposite teeth on the cam This arrangement is one of the variants used in Nippon Denso CAS.

With the cam tooth every 360 degrees this setup gives enough engine information for semi-sequential fuel and wasted spark. (On a rotary such as the RX7, or a two-stroke engine, full sequential fuel and spark is possible as the engine cycle spans 360 degrees.)



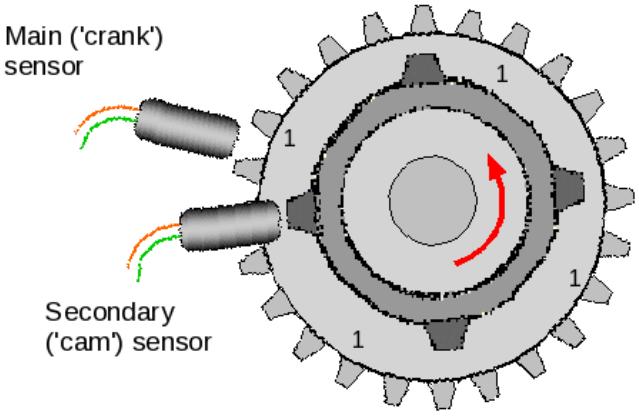
Make sure you understand the definition we're using of tooth #1. With the engine rotating in the normal direction, **tooth #1 is the first tooth to pass the main sensor after either cam tooth has passed the second sensor.**

Make sure these do not happen at the same time - in the diagram you can see that the main sensor is over a gap when the secondary sensor is aligned with its tooth. Use the instructions in the previous single cam tooth section to determine your tooth #1 angle. It will always be between 0 and 360 degrees.

Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel
- Trigger wheel teeth = number of teeth
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Cam wheel
- Second trigger active on = Set to make sure this does not overlap with main wheel
- and every rotation of = Crank

5.3.4.14 Non-missing tooth cam wheel with one cam tooth per cylinder This arrangement is one of the variants used in Nippon Denso CAS. Three and four cylinder versions are known to exist.



As there is one tooth per cylinder, there is only enough information to run a distributor and untimed injection. It is not strictly necessary to use both Ne and G wheels. Using both will give you the improved timing accuracy from the "every-tooth" wheel decoder system, but it is possible to use the "G" input only and configure as "Basic Trigger" instead. Timing will not be as accurate though.

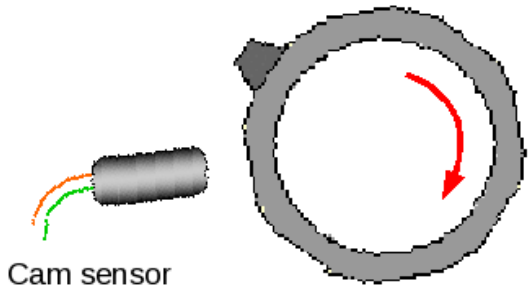
Make sure you understand the definition we're using of tooth #1. With the engine rotating in the normal direction, **tooth #1 is the first tooth to pass the main sensor after any cam tooth has passed the second sensor.**

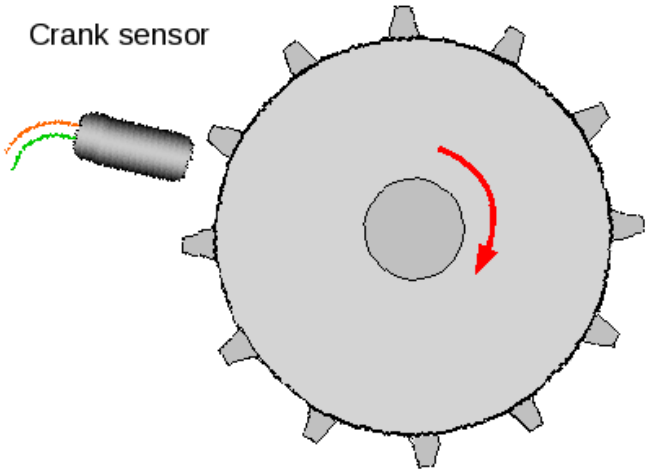
Make sure these do not happen at the same time - in the diagram you can see that the main sensor is over a gap when the secondary sensor is aligned with its tooth. Use the instructions in the previous single cam tooth section to determine your tooth#1 angle. It will always be between 0 and 360 degrees.

Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel
- Trigger wheel teeth = number of teeth
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Cam wheel
- Second trigger active on = Set to make sure this does not overlap with main wheel
- and every rotation of = Every cylinder

5.3.4.15 Non-missing tooth crank wheel with one cam tooth This arrangement appears on a few OEM applications, including the Chrysler 318 and 360 Magnum, the Mazda KLDE, and some Toyota JZ engines. Also, it could be used to extend a simple distributor crank trigger to support sequential. Generally MS3Pro benefits from many crank teeth to improve ignition timing accuracy. However, with this wheel arrangement, you need to beware of trying to use too many teeth on the crank as there is a risk of the trigger inputs overlapping as the cam belt or chain stretches. If this overlap occurs, it will cause sync-loss as the cam tooth moves from being seen "before" to "after" a crank tooth or vice versa. As a general rule, this mode does not work well with more than 12 teeth on the crank wheel.

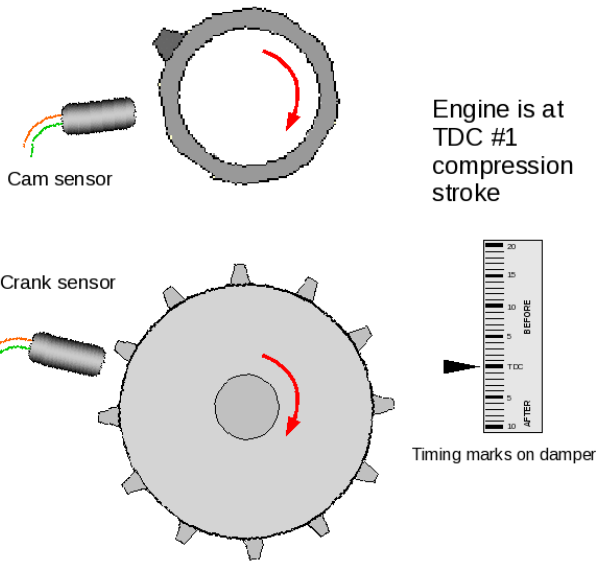




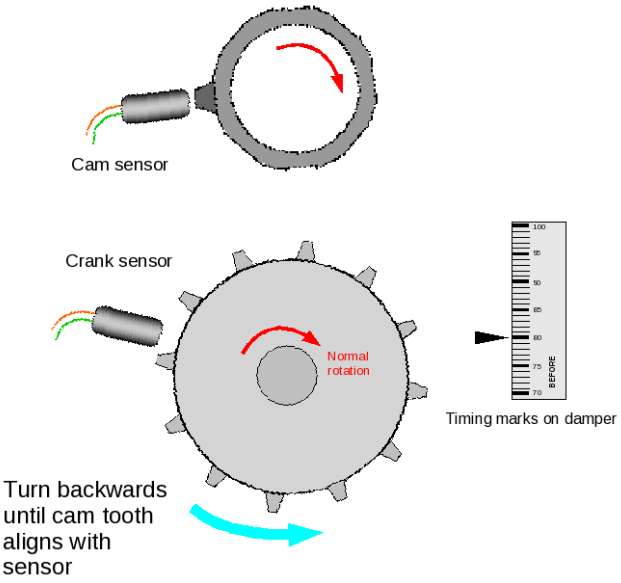
Make sure you understand the definition we're using of tooth #1. With the engine rotating in the normal direction, **tooth #1 is the first tooth to pass the main sensor after the cam tooth has passed the second sensor.**

Make sure these do not happen at the same time - in the diagrams below you can see that the main sensor is over a gap when the secondary sensor is aligned with its tooth. Here's how to determine the trigger angles.

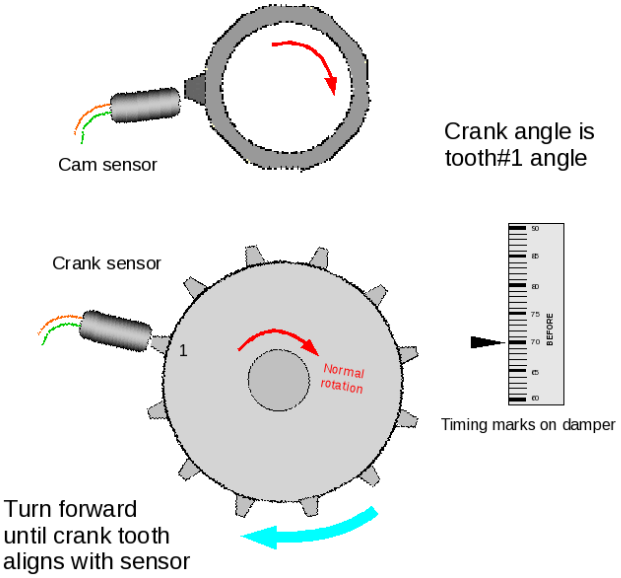
First, set your engine at TDC compression #1.



Now rotate the engine backwards until the cam sensor and tooth line up. If you rotated more than one turn, then add 360 to your tooth#1 angle.



Now rotate the engine forwards until the next crank tooth aligns with its sensor. The crank angle now is the tooth#1 angle. (Note that angles shown in diagram are examples only).



Critical settings include:

- Spark Mode = Toothed Wheel
- Trigger Angle / Offset = 0
- Trigger wheel arrangement = Dual wheel
- Trigger wheel teeth = number of teeth
- Tooth #1 Angle (deg BTDC) = set as above
- Main wheel speed = Crank wheel
- Second trigger active on = Set to make sure this does not overlap with main wheel
- and every rotation of = Cam

5.3.4.16 Example wheel decoder settings Here are settings from engines that use the Toothed Wheel mode. **You will want to confirm these with a timing light**, as trigger wheel tolerances will vary. Some of these are quite specific, while others are general starting points.

BMW inline sixes These engines typically use a 60-2 crank trigger starting with the Motronic 1.1 systems in the mid to late ’80s, with later models adding a one tooth cam sensor. Most use a tooth #1 angle in the 90 to 100 degree range (450 to 460 degrees with certain cam sensor versions), but there are a few exceptions. Check with a timing light to be sure.

Note that models with a VR sensor typically need a 10K resistor between the CKP+ and CKP- wires.

Buick C3I The Grand National (and the later front wheel drive 3800) used a distributor-less ignition with a module that partially decoded the trigger wheel. Although there is an 18 tooth wheel on the crank, it connects to the factory ignition module. The MS3Pro will use the 6X signal from the ignition module plus the 1X cam signal for timing control and sequential injection. You will use spark output A for the EST signal and spark output B for the Bypass signal. This includes the complete ignition module settings as well as the wheel decoder settings. **Note: Some aftermarket cam sensor caps are reported to change the timing significantly - recheck the timing if changing the cam sensor cap.**

- HEI / GMDIS, TFI, C3I Options: Buick C3I
- Ignition input capture: Rising Edge
- Spark output: Going High
- Number of coils: Single Coil
- Trigger wheel arrangement: Dual wheel
- Trigger wheel teeth: 3
- Tooth #1 angle: 70
- Main wheel speed: Crank wheel
- Second trigger active on: Rising Edge
- And every rotation of: Cam
- Dwell type: Standard dwell
- Nominal dwell: 3.0 (adjust as needed to match coils)

Chevrolet LS2, LS3 and other GM 58X V8s These use a 60-2 crank trigger with a 4 tooth cam wheel. The sensors run at 5 volts and use a 1K pull up resistor.

- Ignition input capture: Rising Edge
- Trigger wheel arrangement: Dual wheel with missing teeth
- Trigger wheel teeth: 60
- Missing teeth: 2
- Tooth #1 angle: 70
- Main wheel speed: Crank wheel
- Second trigger active on: Poll Level
- Level for phase 1: High
- Check at tooth #: 1

Chrysler 318 / 360 Magnum V8 These engines used an eight tooth flywheel trigger and a one tooth trigger in the distributor. This can be used to trigger a distributor-less ignition as well, as long as you keep the distributor body, sensor, and trigger wheel.

- Ignition input capture: Falling Edge
- Trigger wheel arrangement: Dual wheel
- Trigger wheel teeth: 8
- Tooth #1 angle: 549
- Main wheel speed: Crank wheel
- Second trigger active on: Falling edge
- And every rotation of: Cam

Chrysler VVT Hemi These used a 60-2 crank wheel with the same tone wheel used on the earlier (NGC) motors on the cam.

- Ignition input capture: Rising Edge
- Trigger wheel arrangement: Dual wheel with missing teeth
- Trigger wheel teeth: 60
- Missing teeth: 2
- Tooth #1 angle: 693
- Main wheel speed: Crank wheel
- Second trigger active on: Falling edge

To complete the cam wheel settings, you will need to set the cam decoder to “Hemi VVT” under the VVT Settings menu.

Ford 36-1 This pattern shows up in many four, six, and eight cylinder engines. Note that for VVT engines other than the Coyote, use the Zetec VCT pattern instead of Toothed Wheel. Many of these installs require a 10K resistor across CKP+ and CKP- to avoid losing sync at high RPM.

- Ignition input capture will depend on the crank sensor wiring.
- Trigger wheel arrangement: Dual wheel with missing teeth
- Trigger wheel teeth: 36
- Missing teeth: 1
- Tooth #1 angle depends on the number of cylinders and the cam sensor. If the tooth #1 angle is set to the version, the engine will be able to run in wasted spark or wasted-COP mode, but not with sequential coil on plug.

Cylinders	Possible Tooth # 1 angles
4	80 or 440
6	50 or 410
8	40 or 400

- Main wheel speed: Crank wheel

Some of these used Hall effect cam sensors and others used VR type ones, but engines without VVT all used a single tooth cam sensor. You may need to adjust the second trigger settings depending on the sensor type.

Ford Coyote This motor also uses a 36-1 crank wheel, but you set up the cam trigger under the VVT settings instead of directly. As with other Ford 36-1 wheels, you may need to use the 10K resistor across CKP+ and CKP- to avoid loss of sync at high RPM.

- Ignition input capture will depend on the crank sensor wiring.
- Trigger wheel arrangement: Dual wheel with missing teeth
- Trigger wheel teeth: 36
- Missing teeth: 1
- Tooth #1 angle: 400
- Main wheel speed: Crank wheel
- Second trigger active on: Rising edge

To complete the cam wheel settings, you will need to set the cam decoder to “Ford Coyote” under the VVT Settings menu.

Ford Coyote GEN3 This version of this engine will be setup just like the 58xLS

- Ignition input capture: Rising Edge
- Trigger wheel arrangement: Dual wheel with missing teeth
- Trigger wheel teeth: 60
- Missing teeth: 2
- Tooth #1 angle: 400
- Main wheel speed: Crank wheel
- Second trigger active on: Poll Level
- Level for phase 1: High
- Check at tooth #: 36

To complete the cam wheel settings, you will need to set the cam decoder to “Ford Coyote” under the VVT Settings menu.

Toyota 2JZ-GTE These settings are for the US market turbo motor that does not have VVT. These engines have a 12 tooth crank wheel called an NE sensor plus two one tooth cam sensors called G sensors. These settings use the “G1” sensor; for the “G2” sensor, add 360 degrees to the Tooth #1 Angle. The later VVT motors use the hard coded 2JZ-VVT spark mode. Pre-VVT versions of the non turbo motor often used a 24 tooth cam wheel in place of the 12 tooth crank wheel, but the settings are similar. The active edge settings assume you have the grounded (common) wires connected to CKP- and CMP-. Reverse wiring the sensors will reverse which edges are active.

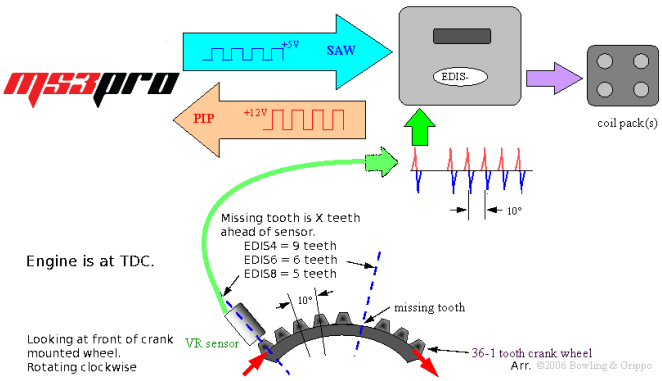
- Ignition input capture: Falling Edge
- Trigger wheel arrangement: Dual wheel
- Trigger wheel teeth: 12
- Tooth #1 angle: 16
- Main wheel speed: Crank wheel
- Second trigger active on: Rising edge
- And every rotation of: Cam

Other wheel arrangements The examples shown here are not an exhaustive list of all the combinations that are possible. For other arrangements of crank and cam wheels, you will need to apply the principles here to your install.

EDIS and EDIS Multispark Ford's Electronic Distributor-less Ignition System (EDIS) is an ignition system that does not require a cam position signal. It can function with just a variable reluctor (VR) sensor and a 36-1 tooth crank wheel (36-1 means "36 teeth minus one", and refers to 36 evenly spaced teeth, one of which has been removed), it will not work with other pattern wheels or hall sensors.

Because it doesn't need a camshaft position sensor, EDIS is a particularly easy way to replace distributor ignitions when retrofitting older engines with a modern computer programmable ignition. The EDIS modules are very reliable and the system works well. The EDIS module itself handles all the decoding of the toothed wheel and sends one pulse per cylinder to the MS3Pro. The code optionally supports the use of a cam sensor in addition to the EDIS system and then supports sequential fuel.

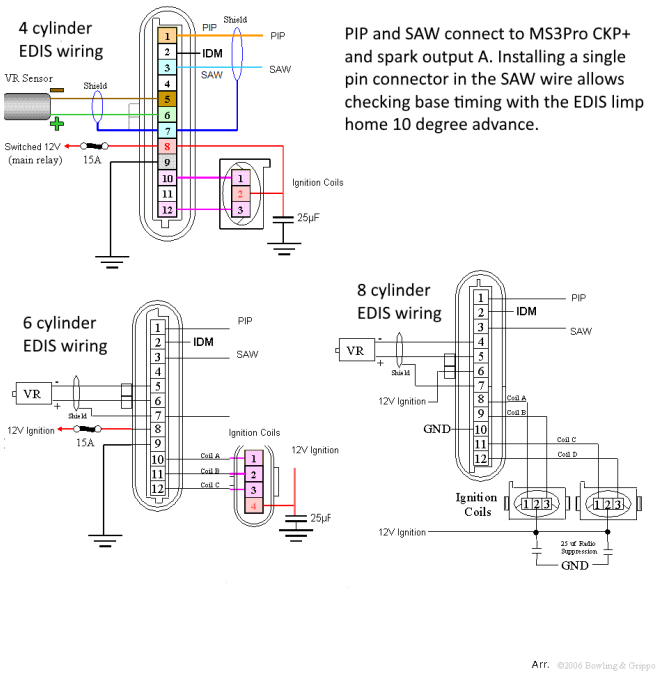
If your engine already has a different trigger wheel setup, consider using that before retro-fitting EDIS - many OEM tooth patterns are already supported in MS3Pro. We generally only recommend EDIS mode on the MS3Pro if the engine is already fitted with a complete, working EDIS setup.



The EDIS system is made up of:

- EDIS module
- crank wheel
- crank variable reluctor sensor (VRS)
- and one or more coil pack(s).

Note that in North America, Ford stopped using the external EDIS module in 1996, and integrated the ignition module into the ECU. Cars that do not have the external EDIS module can use the Toothed Wheel mode.



MS3Pro uses the following wire connections.

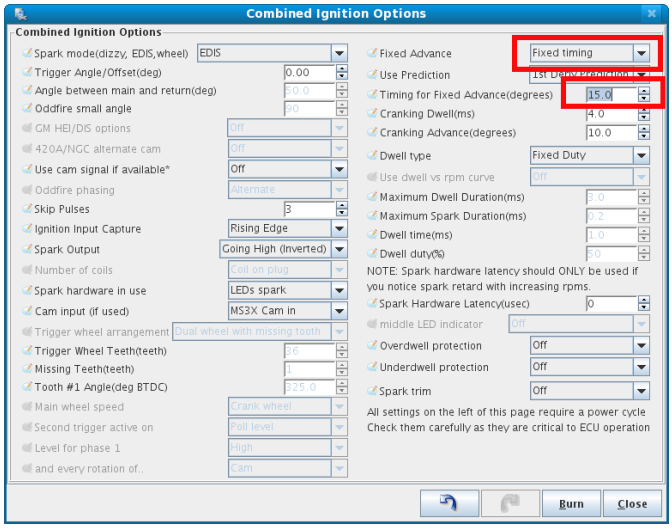
Function	Connection
PIP	CKP+
SAW	Spark A out
Cam (if used)	CMP+

Use the following settings:

- Spark mode: EDIS (EDIS Multispark can be used on some 4 cylinder modules, but this function only works if your module supports it.)
- Trigger angle: 0
- Ignition input capture: Rising edge
- Spark hardware in use: Spark
- Dwell type: Fixed Duty

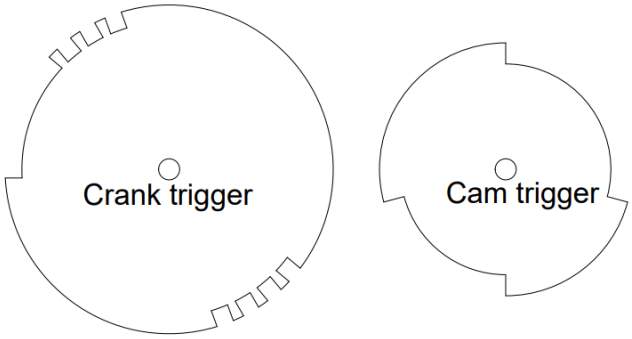
Checking the timing During install, it is important to confirm the timing is correct. To test this it is best to first run the EDIS in limp home mode. This can be achieved by disconnecting the SAW plug/socket or switching off/unplugging the ECU. Fit your strobe onto no.1 plug lead as normal (you may need to try the other tower of the pair). Ensure EDIS still has power and crank your engine, check that the timing is exactly 10°. If not, adjust your sensor until it is. It is safe to idle the engine with the SAW lead disconnected; timing should be rock solid at 10° BTDC. Don't forget to reconnect the plug when done!

Now that you have confirmed that the EDIS is correctly running at 10° BTDC base timing, you need to check that MS3Pro is correctly commanding timing on the EDIS. Start the engine and then on the Ignition settings menu, select Fixed Timing and enter 15 BTDC; check that you strobe 15° BTDC on the crank.



When done, reset Fixed Advance to “Use table.”

Cam sensor and EDIS The EDIS module itself does not need a cam sensor as it always runs in wasted spark, firing each coil pair every crank revolution. However for sequential fuel it is necessary to use a cam sensor in addition so the MS3Pro knows what phase the engine is on. Most original cam sensors are 2 wire VR type, so only require a ground and the signal connecting to the MS3Pro. The code works the same whether a hall or VR sensor is used.



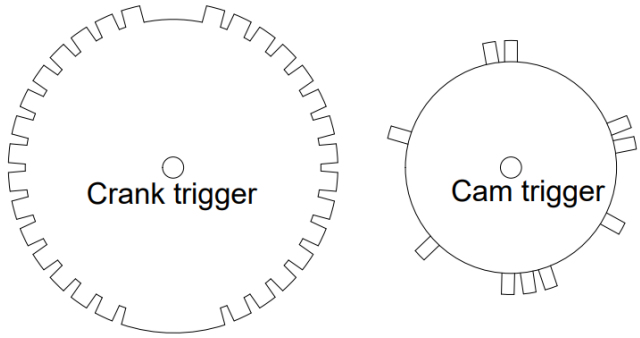
420A/Neon

Known applications include:

- First generation Chrysler / Dodge / Plymouth Neon (both DOHC and SOHC)
- Second generation Diamond Star cars (Mitsubishi Eclipse / Eagle Talon) with 420A
- 2001-2002 Chrysler PT Cruiser
- Many four cylinder “Cloud Cars” (Stratus, Breeze, Cirrus)

This mode uses an irregular crank wheel that can support wasted spark without a cam sensor. It also allows sequential injection if the cam sensor is used. Both sensors require pull-up resistors from the sensor power wire to the signal wire. Ignition input capture should be set to Rising Edge.

Factory ignition coils on these motors require an external ignition module such as the QuadSpark.



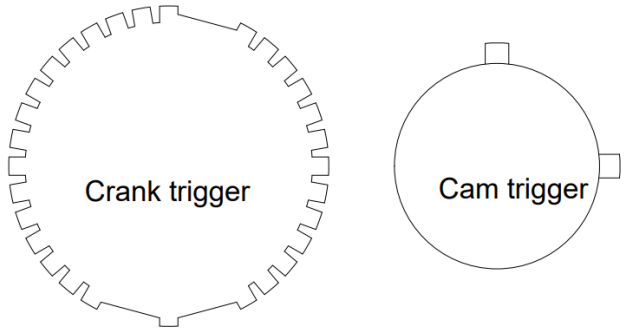
36-2+2
Known applications include:

- 5.7 and 6.1 Hemi without VVT
- Second generation Neon and SRT-4
- Most other Chryslers from 2003 and later

This mode uses a crank trigger (which Chrysler calls a “tone ring”) with 36 base teeth and two separate gaps 180 degrees apart. The number of teeth on the camshaft position sensor change with the number of cylinders; the MS3Pro will select the cam wheel based on the general setup information entered. The picture shown above is for an 8 cylinder application. You must use camshaft position sensor if you enable sequential injection or ignition, but the crank wheel can run the engine in semi-sequential and wasted spark. Both sensors require pull-up resistors. Ignition input capture should be set to Rising Edge.

Factory ignition coils on these motors require an external ignition module such as the QuadSpark.

VVT Hemi models used a 60-2 wheel that uses Toothed Wheel mode.

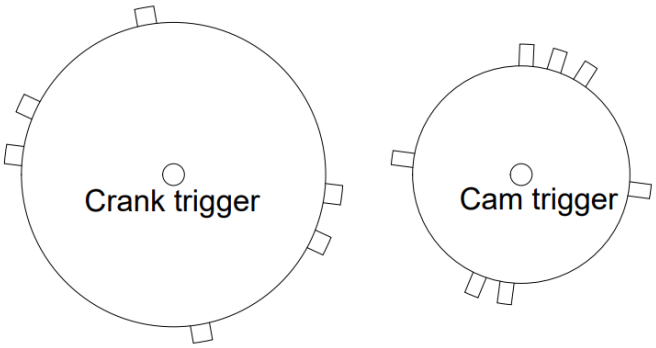


36-2-2-2
Known applications include:

- Many Subarus built after 2000 **without** variable valve timing (AVCS)
- Mazda RX8
- Suzuki J series

This mode uses a VR sensor and a crankshaft trigger wheel with two gaps on one side and one gap on the other. This mode can support sequential injection if cam input is used. Ignition input capture will depend on the way the sensor is wired.

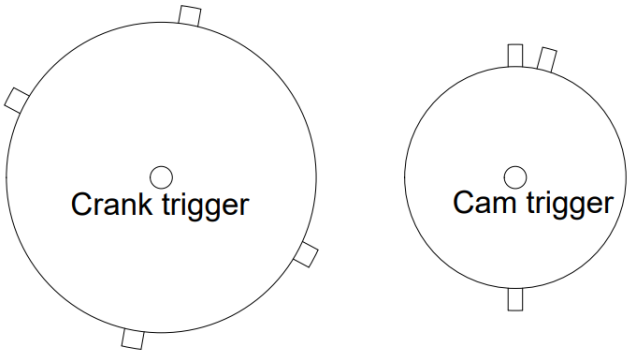
Note that Subarus with variable valve timing (AVCS) and Mazda MZR engines used a different cam pattern. For this mode, see the separate notes on Subaru 36-2-2-2 VVT and Mazda MZR modes.



Subaru 6/7
Known applications include:

- Many Subarus from the 1990s through mid 2000s

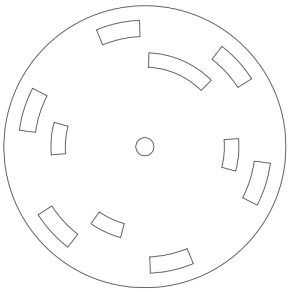
This mode uses a 6 tooth crankshaft trigger and a 7 tooth camshaft trigger, both with VR sensors. You **must** use both sensors with this spark mode. Sequential injection is supported. Because of the nature of the VR sensors, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.



Miata 99-05
Known applications include:

- Mazda MX5 Miata from 1999 to 2005

These cars have a four tooth crank trigger and a three tooth cam trigger with Hall effect sensors. This mode requires using both cam and crankshaft position sensors. This code allows for both sequential injection and control of the variable valve timing found on 2001-2005 engines. Ignition input capture should be set to Rising Edge.



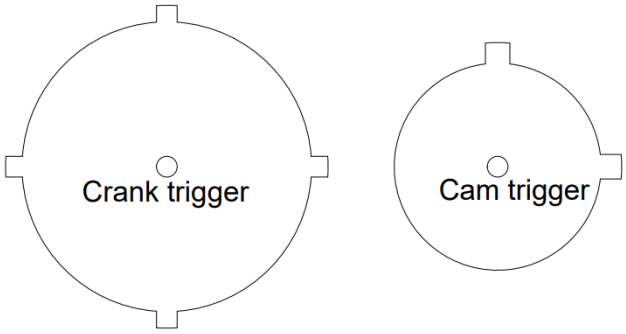
6G72
Known applications include:

- Mitsubishi 3000GT
- Mitsubishi Galant V6

- Some other Mitsubishi and Chrysler V6 models

This spark mode comes in two variations. The one pictured used an optical trigger with six slots for the CKP signal and four slots for the CMP signal. Another variation used Hall effect sensors with a three equally spaced tooth crank trigger and a four tooth cam trigger, with the cam trigger resembling the inner ring of slots in the disc shown above. Both require pull-up resistors, and from the ECU's perspective, they look alike, as in both cases the ECU will "see" three pulses on the CKP wire for every crank revolution.

Ignition input capture needs to be set to Rising Edge. This mode supports full sequential operation.

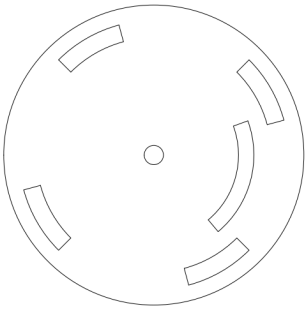


IAW Weber

Known applications include:

- Ford Sierra Cosworth
- Some Fiat and Lancia applications

This application uses a four tooth crank trigger with a VR sensor and a two tooth cam trigger with a Hall effect or VR sensor, depending on the year. Ignition input capture needs to be set to Falling Edge. This mode supports full sequential operation.

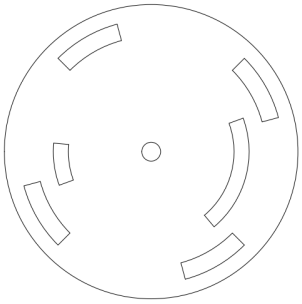


CAS 4/1

Known applications include:

- Mitsubishi 4G91
- Mazda Protege / Escort GT and 323 with optical distributor

This mode uses a Hall effect or optical sensor spinning at cam speed, with both sensors requiring a pull up. There are four teeth on the CKP signal and one on the CMP signal. You will need to set the "Angle between main and return" parameter to the distance between edges of the optical sensor. Note that if you are not able to get a stable signal off both edges, you should instead use Toothed Wheel mode. Full sequential operation is supported.



4G63

Known applications include:

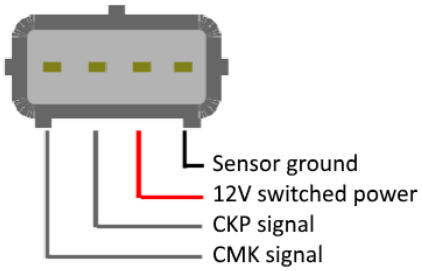
- Mitsubishi 4G63 with distributor-less ignition, as used in Eclipse, Galant VR4, and Lancer Evolution
- 1990-1997 Mazda MX5 Miata

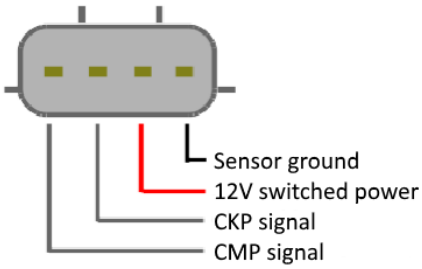
This spark mode comes in several variations. One used an optical trigger (shown above) or Hall effect sensor with four slots for the CKP signal and two unequal length slots for the CMP signal. Another variation used Hall effect sensors with a two tooth crank trigger and a two tooth cam trigger. All require pull-up resistors, and from the ECU's perspective, they look alike, as in both cases the ECU will "see" two pulses on the CKP wire for every crank revolution.

Ignition input capture needs to be set to Rising Edge. This mode supports full sequential operation.

Caution: Early Miatas used a version of this CAS where the timing is 360 (crank) degrees off normal. On these cars, you will need to swap cylinders A-B with cylinders C-D in the wiring.

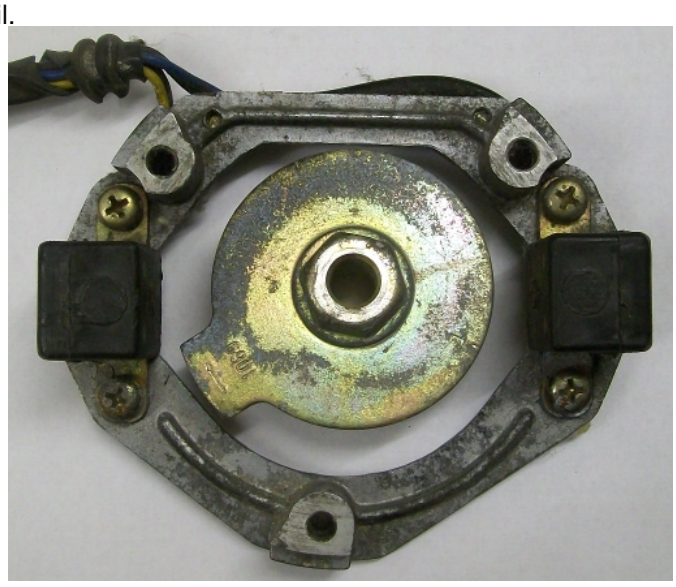
These pinouts show the connections when looking into the connector on the CAS on most common connector versions.





minor changes to the decoder.

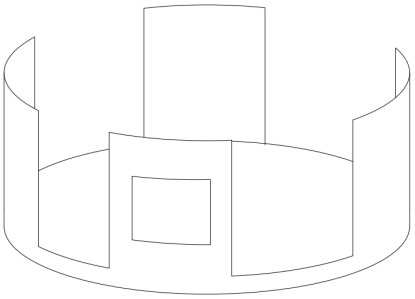
Twin trigger Twin trigger is a specific ignition mode designed for use on a range of motorcycle engines that use a crank or cam mounted reluctor and two discrete pickup sensors. Each sensor triggers a separate ignition coil.



Supported combinations include:

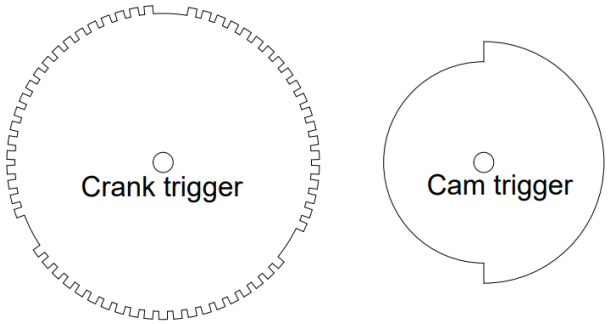
- Crank wheel. 4 cylinder, 4 stroke engines with wasted spark ignition, non sequential fuel.
- Crank wheel. 2 cylinder, 4 stroke engines with in wasted spark ignition, non sequential fuel.
- Cam wheel. 2 cylinder, 4 stroke engines with in coil-on-plug ignition, non sequential fuel.

This mode can be used on both even fire and odd fire engines.



Chrysler 2.2 / 2.5

This setup is unique to Chrysler 2.2 and 2.5 engines from the 1980s and early 1990s, equipped with multiport injection. (The TBI versions of this engine used Basic Trigger mode instead.) It uses a four tooth cam trigger with a “window” in the middle of one tooth, Hall effect sensors which require pull up resistors, and both sensors read the same trigger wheel, but from different angles. Currently this mode is experimental. With current firmware, only the primary Hall effect sensor is used, wired to CKP+.

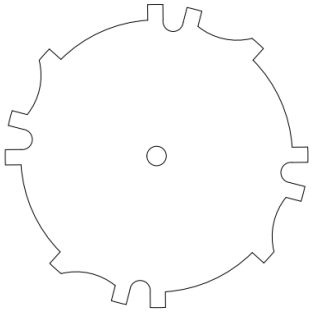


Renix 44-2-2

Known applications include:

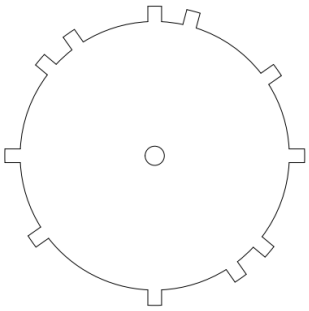
- 1987-1990 Jeep Cherokee 4.0
- Many 1980s era Renault products

This trigger mode came in a four cylinder variation which used 44 base teeth with two gaps 180 degrees apart, and a six cylinder version (shown) with 66 base teeth and three gaps 120 degrees apart. A cam signal is needed for sequential injection. Jeeps had a cam sensor from the factory, while most Renaults did not.



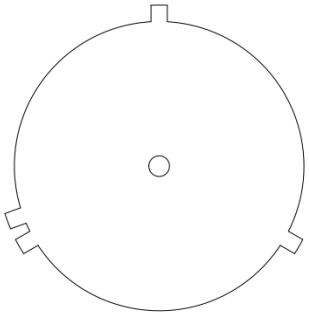
Suzuki Swift

This mode is considered experimental. It works for Suzuki Swift engines with a distributor with a VR sensor and 12 irregularly spaced teeth. This mode does not support sequential injection, and can only be used with a single coil. Because of the nature of the VR sensor, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.



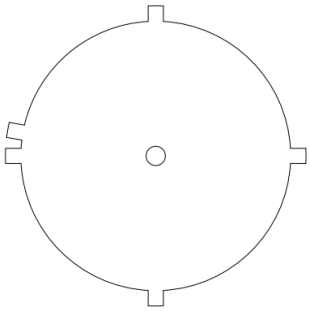
Suzuki Vitara 2.0

This mode is considered experimental. It works for a later modification of Suzuki's irregularly spaced teeth that has 11 teeth instead of 12, spinning at cam speed. Supports sequential injection. Because of the nature of the VR sensor, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.



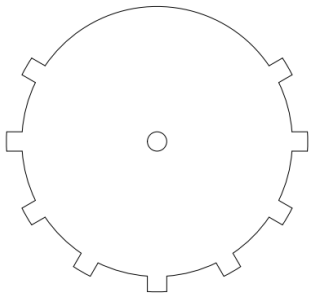
Daihatsu 3cyl

This mode is considered experimental. The Daihatsu three cylinder version has 3 equally spaced teeth in a distributor with a fourth tooth adjacent to one of the teeth and a VR sensor. Supports sequential injection.



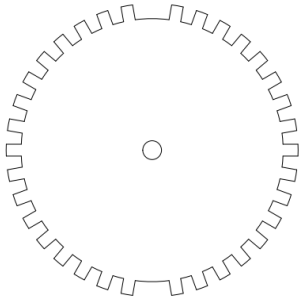
Daihatsu 4cyl

This mode is considered experimental. The Daihatsu four cylinder version has 4 equally spaced teeth in a distributor with a fifth tooth adjacent to one of the teeth and a VR sensor. Supports sequential injection.



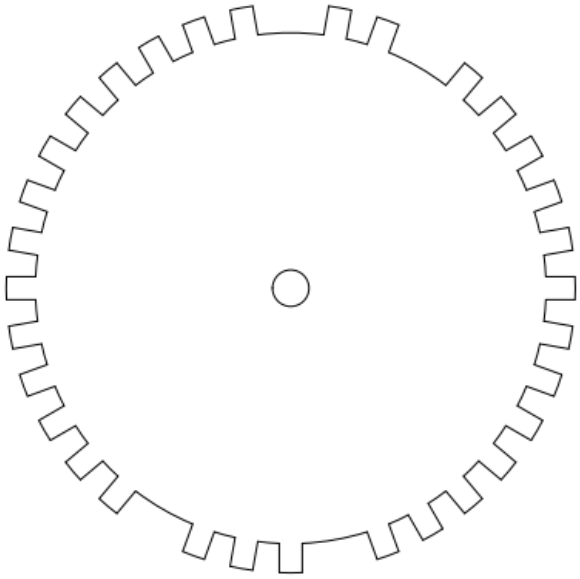
VTR1000

This ignition appears on some Honda V-twin motorcycles. It uses a 12-3 crank trigger with a VR sensor and no cam sensor. Sequential injection is not supported due to the lack of cam signal.



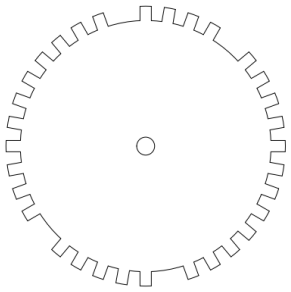
Rover #1

This mode is used on Rover K-series engines with a crank trigger wheel with 36 base teeth and two one tooth gaps, 180 degrees apart. This only allows a single coil and batch fire injection. Since the K series engines with this trigger pattern did not have a cam sensor, this decoder does not have any provisions for cam input.



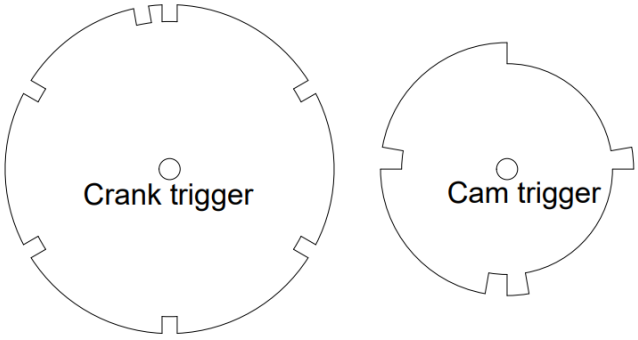
Rover #2

This mode is used on Rover K-series engines with a crank trigger wheel with 36 base teeth and four one tooth gaps, with two teeth between one pair of gaps and three between the other. This only allows a single coil or wasted spark ignition and batch fire or semi-sequential injection. Cam input is not supported.



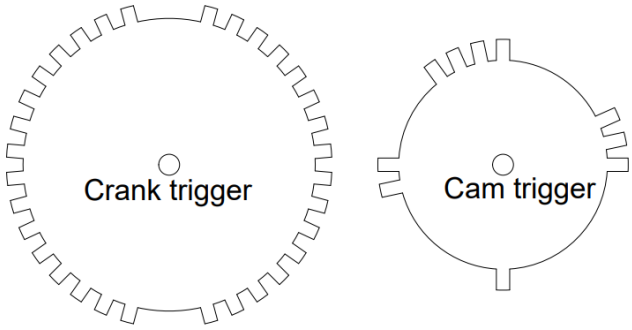
Rover #3

Similar to Rover #2, but the gaps are positioned differently: One pair of gaps has four teeth between them, and the other pair has five. As with Rover #2, supports wasted spark and semi-sequential injection, but does not support cam input.



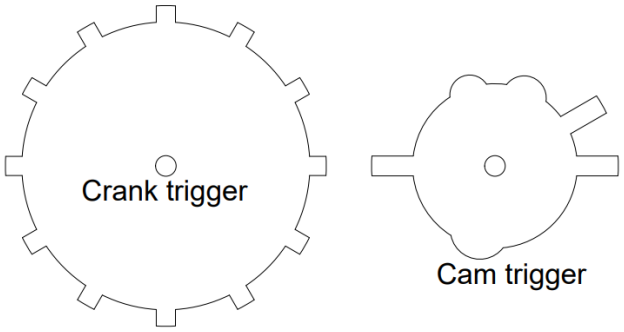
GM 7X

This appears on several GM four and six cylinder engines with distributor-less ignitions. GM refers to the crank wheel in their internal documentation as a 7X trigger wheel. It has six equally spaced teeth and a seventh tooth for cylinder identification. When using the cam sensor, the MS3Pro checks for a cam tooth at the same position as the first regular tooth after the added tooth. GM used several cam triggers with this engine - a 1 tooth wheel, a 4X design (shown above), and a 6X design. Currently the MS3Pro supports variable valve timing with the 6X trigger only.



QR25DE

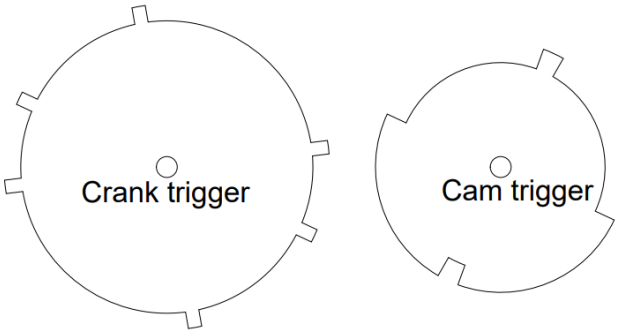
Used on the Nissan QR25DE and some other Nissan four cylinders with crank triggering. The crank wheel has 36 base teeth with two gaps two teeth wide, while the cam trigger has four clusters of teeth. Supports fully sequential injection.



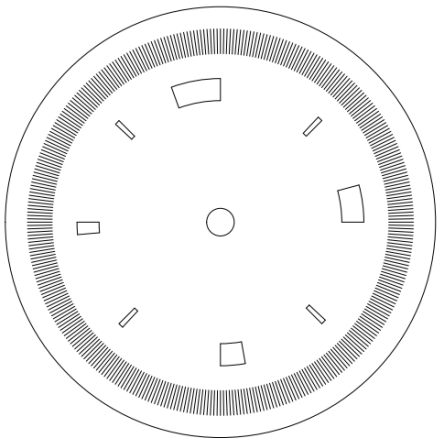
Honda RC51
Known applications include:

- Honda RC51
- Honda S2000
- Many injected Honda four cylinder applications

This one uses a 12 tooth crank trigger and 3 tooth cam trigger, with VR sensors on both. The cam trigger wheel may also include a set of bump as shown - the smaller bumps do not trigger the sensor, but help balance out cam vibrations. Supports fully sequential injection. The cam trigger pattern is different between odd fire V4 and even fire inline four applications; the even fire / odd fire setting under the Basic / Load Settings determines which cam decoder to use. The picture above shows the odd fire version of the cam pattern.



Fiat 1.8 16V
Uses a six tooth crank trigger and a three tooth cam trigger, with a VR crank trigger and Hall effect cam signal. Both wheels have irregular spacing.

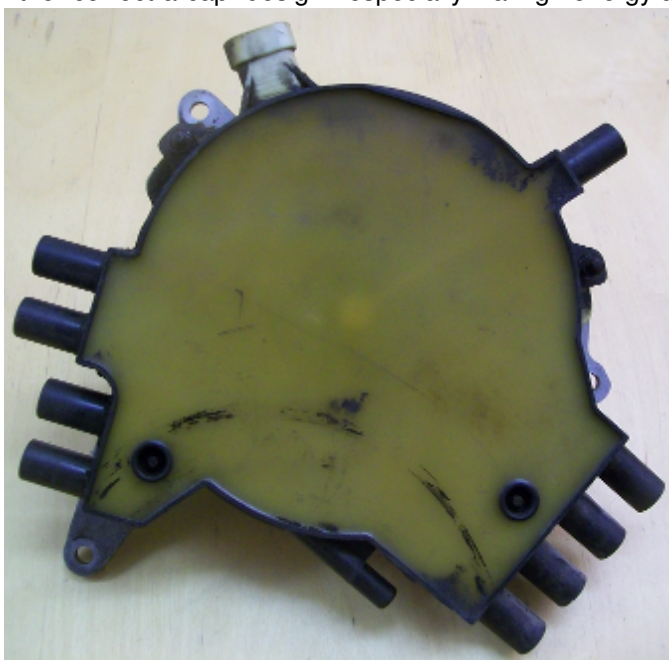


Optispark
Known applications include:

- Chevrolet LT1 and variations (e.g. LT4)

The Nissan VH45DE V8 uses a similar pattern, but the slots are slightly different lengths and in the reverse pattern - the Optispark decoder will not work with their trigger wheel.

The Optispark system was used on GM vehicles from 1993 to 1997 on LT1, LT4 and L99 applications. Internally, it uses a Mitsubishi / Nissan derived optical trigger arrangement. There is a "hi-res" track of 360 slits and a "low-res" track of 8 slots of varying length. The pickup design is sound, but the high-tension side can be problematic with the "correct-a-cap" design - especially if a high energy aftermarket ignition system is used.

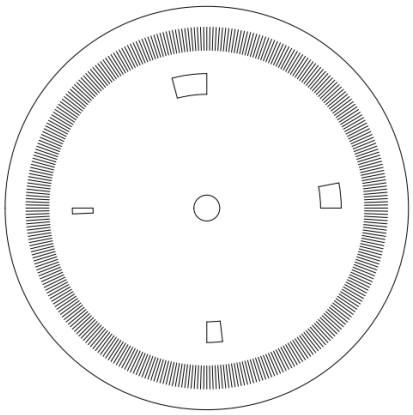


The MS3Pro Optispark decoder uses both low and high resolution tracks for improved ignition accuracy. (Most other aftermarket implementation only use the low resolution track.) The system allows for sequential fuel and the single coil as per the original install. However, as an enhancement, the single coil can be replaced by a wasted-spark or coil-on-plug setup to eliminate the troublesome high-tension cap. This lets you keep the stock Optispark trigger wheel with alternate ignitions.

The Optispark requires a fused 12V supply. This can be tapped into the same 12V supply as the MS3Pro. The Ground connection should be run to the sensor ground at the MS3Pro. The high and low resolution tach signals each require a 5 volt "pull up" resistor to operate correctly. The connector is marked with letters that can be used to identify the functions.

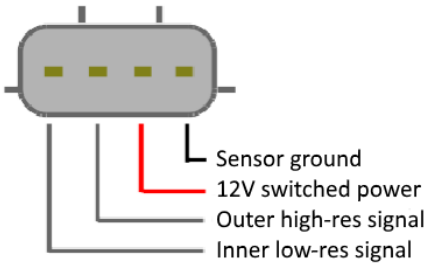
Optispark pin	Function	Connection
A	Low resolution signal	CKP+
B	High resolution signal	Digital frequency in 2
C	12 volt power	Shares power
D	Ground	Sensor ground

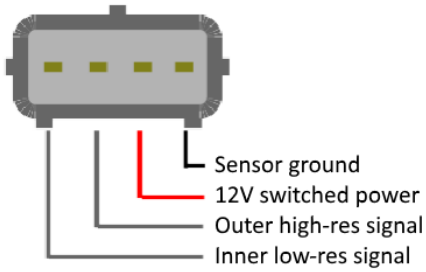
The low resolution signal runs to CKP. Because of the high frequency of the high resolution signal, it needs to be connected to the Digital Frequency In or Digital Frequency In 2 input instead of the CMP signal. You will need to specify which frequency input is used in the Ignition Options / Wheel Decoder menu. Ignition input capture needs to be set to Falling Edge, and the Spark Output needs to be set to Going High if using the stock ignition module.

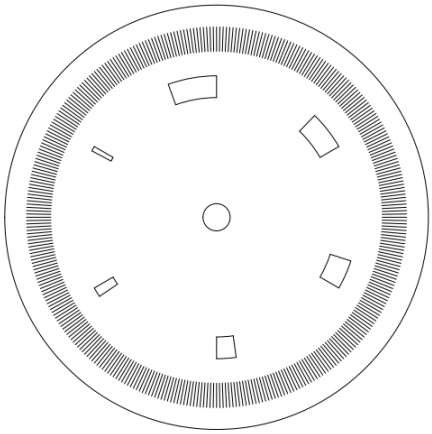


Nissan SR20

This mode is used for the RWD version of the SR20DET with coil on plug ignition and the CA18DET. These engines use an optical CAS with four unequally sized slots and a row of 360 slots. It uses the same pull ups as the Optispark, and like the Optispark, uses Digital Frequency In 2 instead of CMP for the cam signal. Nissan used a number of different CAS connectors depending on where they sourced their sensors. The diagrams below are shown looking into the connector on the CAS.





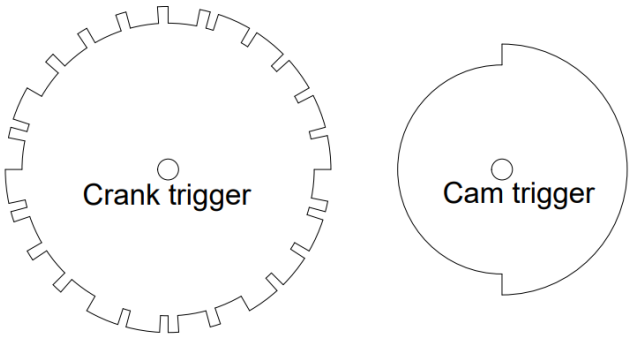


Nissan RB25

Known applications include:

- Nissan Skyline RB25DET, RB26DETT, RB20DET
- Nissan VG30DE and VG30DETT

This mode works for Nissan engines that used six unequally sized slots; it does not work for earlier VG30E or RB30E applications which used a distributor and one long slot with five short ones. Pull ups and CAS wiring are the same as for the SR20DET.

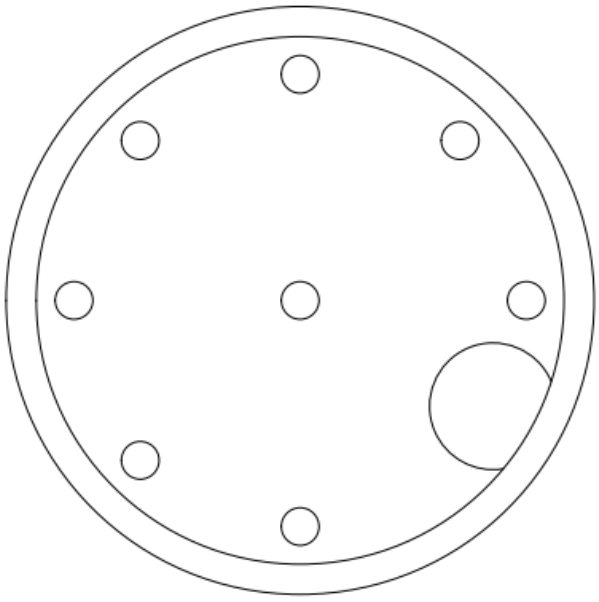


LS1

This mode is used on the Chevrolet LS1 and related Gen III small block V8 engines with the 24X wheel, and also appears on a few V6 applications from the same time frame. The CKP input wires to the 24X sensor and the CMP input wires to the 1X sensor. Both are Hall effect sensors, but do not require pull ups. This spark mode looks like it might be the “dual wheel” mode, but if you look closely at the crank trigger, you’ll see that the edges of the crank wheel are different lengths. The MS3Pro can use these different lengths to sync faster and avoid the sorts of problems caused by timing chain stretch causing the cam sensor to trigger on different teeth.

Ignition Input Capture needs to be set to Rising Edge for MS3Pro, and spark output needs to be set to Going High.

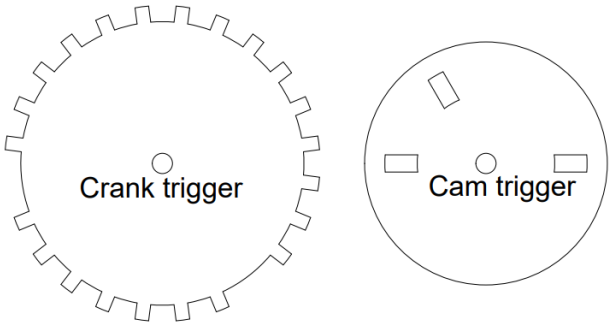
For LS2 and other engines in the Gen IV family with a 58X crank wheel, see the examples under Toothed Wheel.



YZF1000

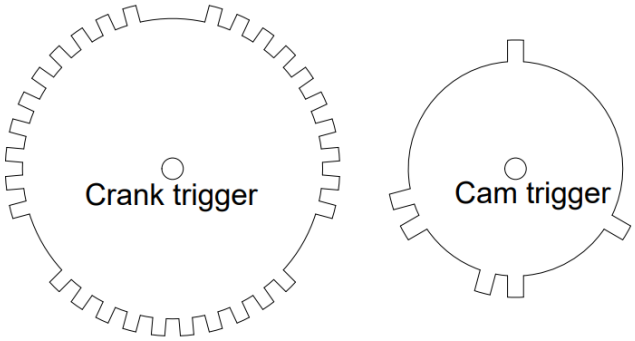
This is used on Yamaha motorcycle engines with 7 small holes and 1 long recess in the crank trigger. There is no cam sensor in this mode, so current firmware only supports wasted spark and semi-sequential fuel. Known applications include the following Yamahas:

- YZF1000 / Thunderace
- FZR1000
- FZR750
- FZ700



Honda Acura (V6)

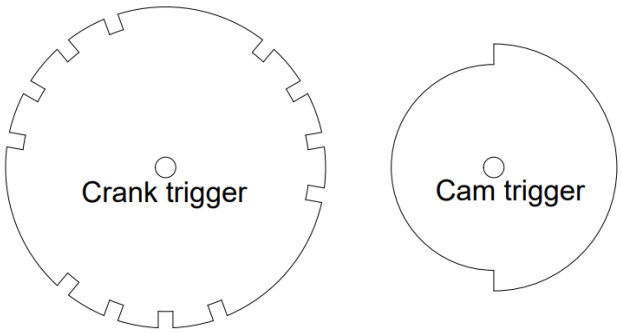
This mode uses a crank trigger with 24 base teeth and two separate missing teeth, along with a 3 tooth cam sensor. This is used on many Honda and Acura J series V6 motors. Note that some of these have a crank trigger with no missing teeth and a 4 tooth cam trigger - the MS3 currently does not support this pattern.



VQ35DE

Used on the Nissan 350Z and other VQ35DE applications. These use a crank trigger with 36 base teeth with 3 sets of 2 missing teeth, and a 6 tooth cam wheel. Supports fully sequential operation.

Note that the VQ30DE used in late ’90s Maximas has a different trigger pattern that the MS3Pro currently does not support.

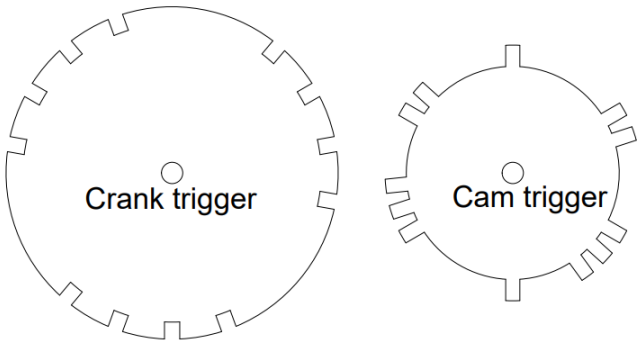


Jeep 2000

Known applications include:

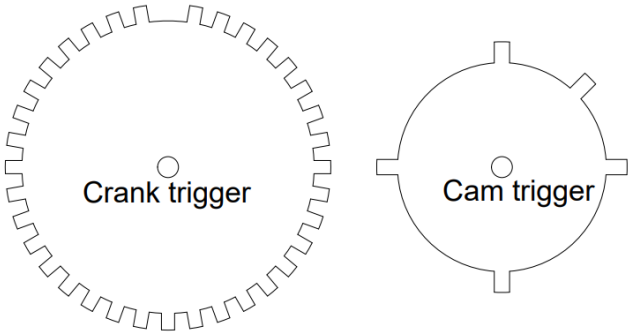
- 1991-2000 Jeep 4.0 inline six
- Dodge Avenger 2.5 V6
- Some Chrysler minivan V6s

This mode has thee sets of four notches on the crank trigger and a one tooth distributor trigger. Supports fully sequential operation.



Jeep 2002

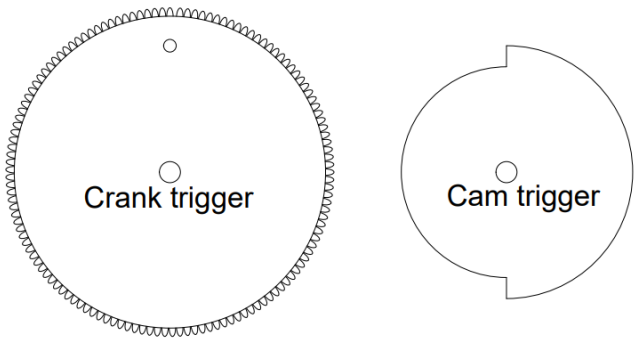
This mode appears on the last run of the Jeep 4.0 inline six, with coil packs instead of the distributor. Also appears on the Chrysler 3.7 V6. Uses the same crank trigger as the Jeep 2000 mode, but with a more complex cam pattern. This mode is considered experimental.



Zetec VTC

Used on Ford Zetec and other four cylinder engines with variable valve timing. It also supports Jaguar V8s with variable valve timing. Features a 36-1 crank trigger like many other Fords, but a five tooth cam wheel instead of a one tooth. Uses VR sensors on both, and supports full sequential operation and variable valve timing. Because of the nature of the VR sensor, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.

Note - the Ti-VCT system used on the Coyote motors is considered a subset of Toothed Wheel with a special cam decoder instead of Zetec VCT mode. See the Coyote example under Toothed Wheel.



Flywheel tri-tach

Known applications include:

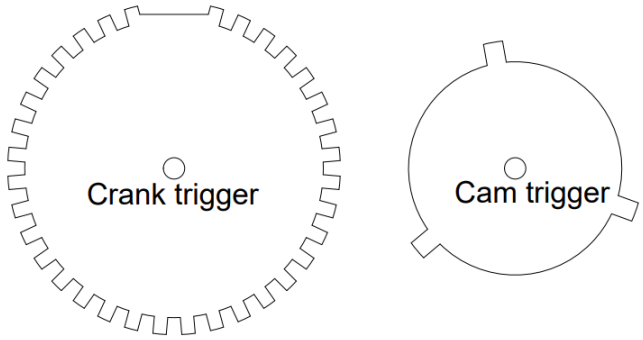
- Early 1980s Porsche 911
- Porsche 944 Turbo (951)
- 1986 and earlier BMW 325e
- E30 chassis BMW M3 with S14 motor
- Many 1980s and early 1990s Audis

This application uses a VR sensor that counts flywheel teeth, with a second flywheel sensor that reads a single post and, in most implementations, a cam sensor. Note that the number of teeth is hard coded for a specific number of cylinders. With a cam sensor, this will support full sequential. Current ones available include:

Number of teeth	Number of cylinders
135	5
135	6
136	8
132	4
130	4

For wiring, connect the CKP wires to the reference (1 tooth) sensor on the flywheel, Digital Frequency In 1 to the flywheel tooth sensor, and the cam signal (if present) to CMP. For most applications, the flywheel sensors are both VR type with a Hall effect cam sensor.

This mode is also considered experimental at this point.

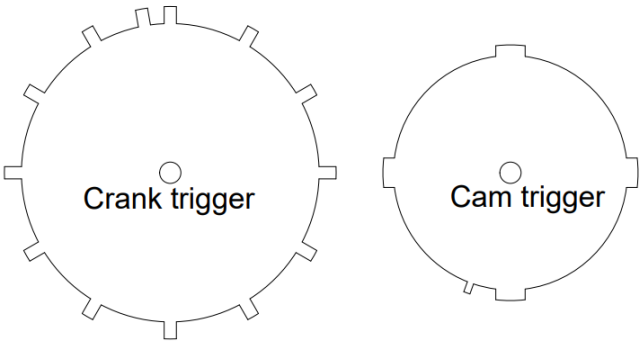


2JZ VVTi

Used on the Lexus IS300 and many 2000 and later Toyota six cylinder engines with VVTi. This uses a 36-2 crank trigger and a three tooth cam trigger, with VR sensors. Supports sequential injection and variable valve timing. Because of the nature of the VR sensor, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.

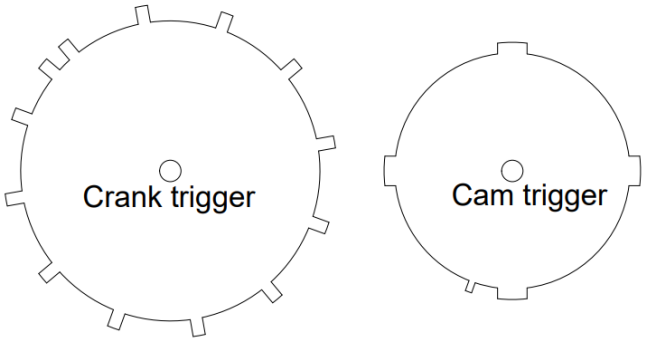
This mode now also supports VVTi applications with a 36-2 crank trigger in four, eight, and twelve cylinder variations. Twelve cylinder mode is considered experimental.

Note: Some installers have reported that they have found these engines have less sensor noise if you tie both CKP- and CMP- to sensor ground. Try this if you run into noise issues with an engine that uses this spark mode.

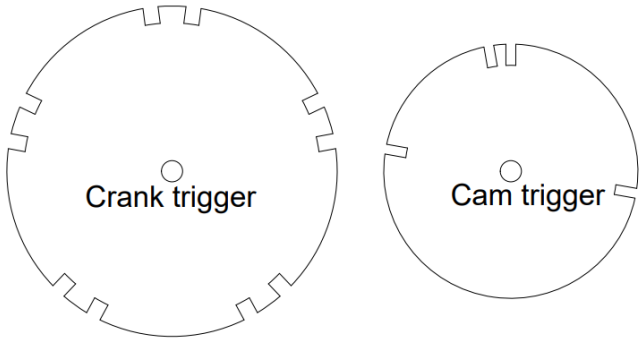


Honda TSX / D17

Uses a 12 tooth crank trigger with one tooth added for a total of 13 crank teeth, combined with a 5 tooth cam trigger. This allowed Honda to add continuously variable valve timing. Uses VR sensors.



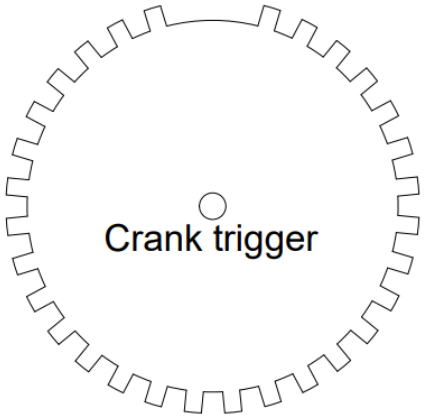
Note: Honda used a slightly different variant of this decoder mode on some K series engines such as the K24A2. If your timing is significantly off or you are not sure which one your engine uses, check with a timing light with the fuel injectors unplugged before attempting to start the motor. Note that the CMP input connects to the exhaust cam sensor. Connect the intake cam position sensor to a digital frequency input for VVT control.



Viper V10

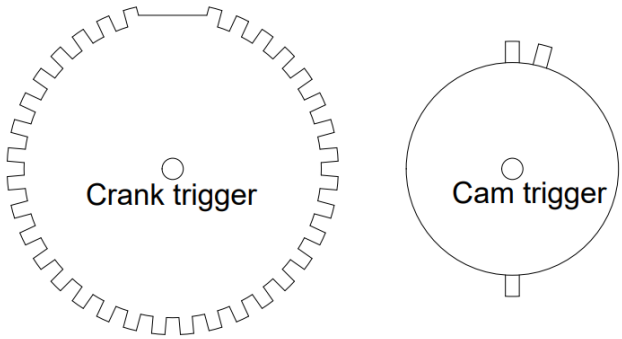
Used for Vipers and V10 Rams with JTEC ECU. This one has a crank trigger with five groups of two teeth. A cam sensor is also required, with a one tooth trigger wheel. Both are Hall effect and require pull ups. Supports sequential injection. You can use the main spark outputs for wasted spark, or use PWM Out 2 and 3 as extra spark outputs to use sequential coil on plug. Factory Chrysler coils require external ignition modules such as the QuadSpark. Gen 1 mode is used for the 1995 and earlier, which had several teeth on the cam trigger (above), while the others use regular Viper V10 mode and have a one tooth cam trigger (below).

Chrysler ran these sensors on an 8 volt regulator. We have found they will work if supplied with 12 volts, but do not give a reliable signal if connected to 5 volts.



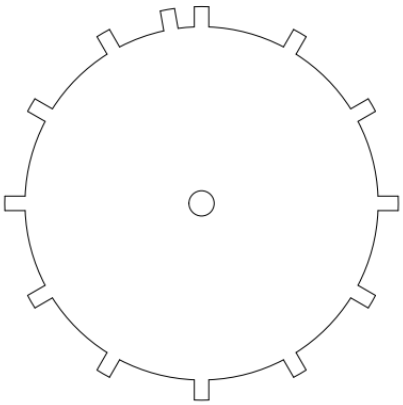
HD32-2

Used for Harley-Davidson engines that use a 32-2 crank wheel and a MAP sensor instead of a cam sensor. Cam sensor needs to be set to MAP. The front cylinder is considered the #1 cylinder.



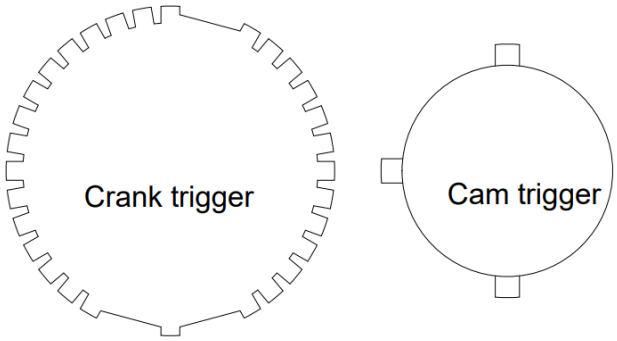
Miata 36-2

This mode is specifically written for ’99-’05 Mazda Miatas running the Flyin’ Miata crank trigger and the stock ’99-’05 cam sensor. Supports full sequential ignition and VVT. Ignition input capture needs to be set to Rising Edge. As with the 1999-2005 Miata spark mode, this uses Hall effect sensors.



Daihatsu 12+1

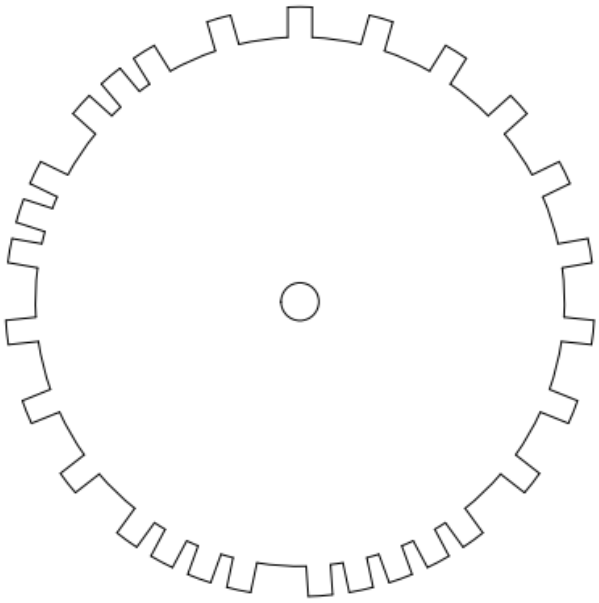
The Daihatsu EF-SE and other three cylinder engines with coil on plug ignition used a trigger wheel turning at cam speed with 12 equally spaced teeth and one additional tooth for cylinder identification. This single sensor is a VR type and wires to the CKP inputs. Because of the nature of the VR sensors, it can trigger off either edge, depending on the wiring. Ignition input capture should be tried both ways and set to which one gives you the most stable signal.



Subaru 36-2-2-2 VVT

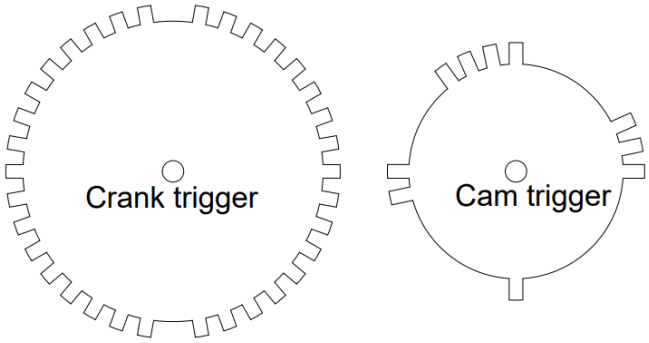
Subarus with variable valve timing (often called AVCS in Subaru’s documentation) changed their cam trigger pattern, requiring a different wheel decoder from Subarus with a 36-2-2-2 crank pattern and fixed cam position. These engines have a second cam sensor that can be wired to any unused digital frequency input. Settings are otherwise the same as for 36-2-2-2 mode. The cam sensors are Hall effect and some Subaru sensors need a pull up of a 1.0 to 2.49K resistor wired between their signal and supply voltage. If the pull-up is left out, the sensor may read correctly while cranking but give a cam fault once the engine RPM goes above cranking levels.

Note: The primary cam uses the cam sensor on the left hand side of the engine (as viewed from the driver's seat in a conventional installation), and should be wired to the CMP input. The right hand side cam sensor wires to a digital frequency input instead.



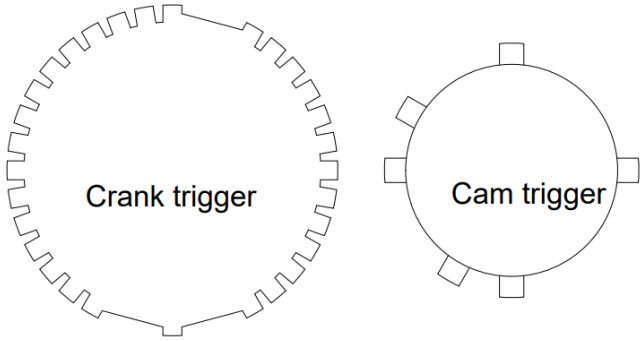
Ski Doo PTEC

This mode supports 2 cylinder, 2 stroke engines using a single crank sensor with 24 base teeth and four added teeth. As this engine is a two stroke, no cam wheel is needed or possible. Supports fully sequential injection for a two stroke.



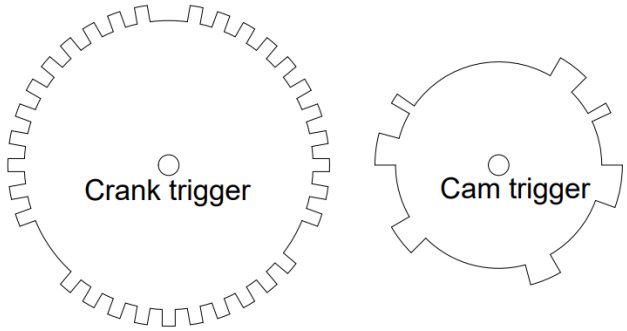
Nissan QG15

The QG15 used a similar pattern to the QR25, but instead of a crank wheel with 36 base teeth and two gaps two teeth wide, the crank wheel has two gaps **one** tooth wide. This required a different decoder from the QR25.



Mazda MZR

Yet another cam pattern paired with the 36-2-2-2 crank trigger; in this case, the cam wheel has six irregularly spaced teeth and a Hall effect sensor. This one was used in the third generation Mazda MX-5 Miata, along with the Mazda 3 and other applications. Note that versions of this motor branded as the Ford Duratec or Ecoboost used a 36-1 crank trigger and would use toothed wheel mode, combined with selecting the Ford Duratec cam decoder in the VVT menu.



6G75

The Mitsubishi V6 uses a crank trigger with 36 base teeth, multiple missing teeth and a seven tooth cam wheel.

Idle valves The idle valve can be selected under **Startup** / **Idle** and chose **Idle Control**. The first option allows you to select your type of idle valve:

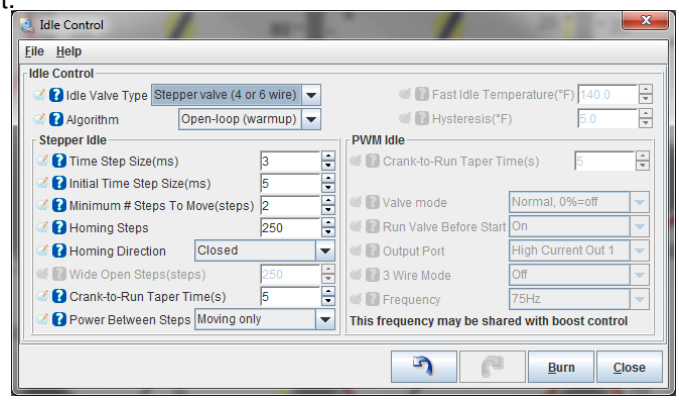
- **None** - The engine can run without an idle control valve if the throttle is adjusted to prevent stalling.
- **On** / **Off Valve** - A solenoid that only opens and closes for a very basic open loop operation.
- **PWM valve (2 or 3 wire)** - Uses a pulse width modulated solenoid where changing the duty cycle changes the valve position. 2 wire valves use a single solenoid and a return spring, while 3 wire valves use two solenoids in a push-pull configuration. Common applications include Ford, Mazda, Honda, some Toyotas, and most European cars that do not use drive by wire.
- **Stepper valve (4 or 6 wire)** - These valves use four separate coils to rotate a shaft, and have from 4 to 5 wires. Often used in GM, Chrysler, Mitsubishi, and Toyota vehicles.
- **DBW** - When using an external drive by wire controller, the MS3Pro can use the primary drive by wire actuator for idle control. Note that some drive by wire throttle bodies still use a PWM or stepper valve for idle control.

The next box allows you to chose between open loop and closed loop.

- **Open loop (warmup)** - The valve opening amount is set as a function of coolant temperature, starting at an open value with the engine cold and closing the valve as it warms up.
- **Closed loop** - The ECU moves the valve as needed to target a specified idle speed.

We recommend only selecting a closed loop idle mode after getting the engine running well in an open loop mode. Open loop commands how to open the valve as a function of temperature, while closed loop targets a specific idle speed.

Once you have selected a mode, you will need to set up the individual IAC modes. Settings not used are grayed out.



On / Off settings

- **Fast Idle Temperature** - the solenoid opens below this temperature and closes above it.
- **Hysteresis** - If the temperature drops below the opening temperature, it needs to drop by this additional amount before re-opening the valve to prevent the MS3Pro from pulsing the valve rapidly at the opening temperature.
- **Output port** - Select which output you wish to use to control the solenoid.

Stepper valve settings

PWM idle valve settings

- **Crank-to-Run Taper Time** - This is the amount of time the IAC valve uses to transition after starting from its crank position to its running position.
- **Valve Mode** - Normal mode is off at 0% duty cycle, inverted mode is off at 100% duty cycle. As the name implies, most valves use normal mode. If you've found you accidentally wired up a 3 wire IAC valve backwards, just flip this setting.
- **Output Port** - Allows you to pick which output wires are used. In 3 wire mode, this is for the coil that opens the valve.
- **3 wire mode** - Used with 3 wire rotary IAC valves, common in Denso and Bosch applications. Set to 'Off 'for a 2 wire valve. For a 3 wire valve, this specifies the connection used for the coil that closes the valves.
- **Valve Frequency Selection** - Select a frequency from the list. Different valves operate best on different frequencies.
- The PWM idle control uses two additional menus, **PWM Idle Cranking Duty** and **Idle PWM Duty Table**. These specify the IAC position in cranking and running as a function of temperature. A third menu, **PWM Idle Voltage Compensation**, lets you specify an amount of duty cycle to be added to or subtracted from the duty cycle as a function of battery voltage.

Sensor calibration The MS3Pro uses internal calibration tables to convert the resistance/voltage from the input sensors into a real world number. It is necessary to setup this calibration so the inputs read correctly. The built-in calibrations for coolant temperature (CLT) and air temperature (MAT) are for GM sensors, so if you have GM sensors you are safe to leave these alone. All installs should set the calibration for the AFR (air fuel ratio) table that is used for the oxygen sensor (lambda sensor) whether narrowband or wideband. Here's how to change the sensor calibrations.

1. Unlock the calibrations (This lock/unlock is to ensure the calibration isn't changed by accident.)

2. Open the Set Thermistor Tables dialogue, then set the CLT calibration (if needed) Pick the sensor type. The bias resistor value is hardwired at 2490. For a custom sensor, enter three measured temperature and resistance pairs. (Don't guess, measure.) For greatest accuracy the bias resistor should be close to the sensor resistance at a "warm" temperature. Write to controller.

3. Set the MAT calibration in the same way (if needed.) The bias resistor is 2490 ohms. Write to controller. Close the dialogue.

4. Calibrate the AFR table. Either pick a preset or look up the voltage/AFR pairs from your controller manual. Write to controller. Note! You should also set your Project Properties for narrowband/wideband and configure the settings on the "EGO control" page.

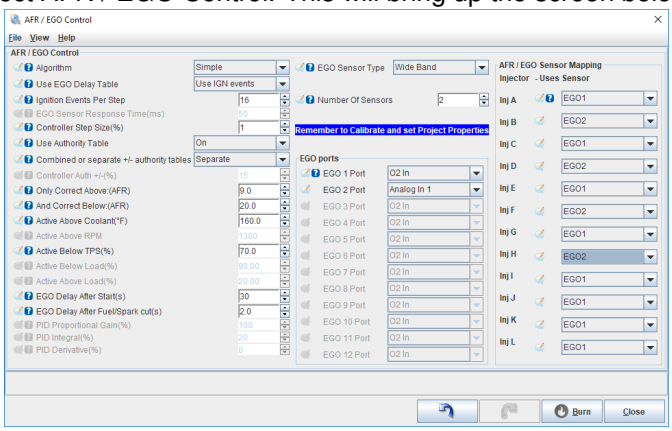
5. Re-lock the calibrations and Burn.

6. Calibrate TPS. With the throttle closed, "Get Current" on the closed throttle. Then fully open the throttle and "Get Current" on the open throttle. Close.

7. Calibrate the MAP/Baro. Typically chose from one of the preset values.

8. It is not usually needed to calibrate the battery voltage. Do not alter these values from the defaults unless you have confirmed the actual voltage reading is inaccurate.

You'll also want to set up a couple basics on how the MS3Pro uses the oxygen sensor. Go to Fuel Settings and select AFR / EGO Control. This will bring up the screen below.



At this point, you'll just want to set up a couple basic parameters. When first tuning, it helps to have the O_2 sensor correction off, so you'll set Algorithm to "No Correction". At the top of the middle column, specify if you have a wide band or narrow band sensor, and enter your number of sensors (note that setting the EGO sensor type to Disabled will not just disable correction; it will disable all O_2 input logging as well). The MS3Pro can read up to four sensors over its analog inputs, and even more over the CAN bus with appropriate expansion modules.

If you have more than one O_2 sensor, you will need to set what port (input) is being used. Select which analog input is associated with what sensor with the drop down boxes in the middle. The tab at the far right lets you pair each injector channel with a specific O_2 sensor. The MS3Pro can use any O_2 sensor to correct for any injector channel.

Once you're done calibrating the sensors, check them from the main dash. To change any gauges Right-Click on them and pick a different gauge. The ones we need here are under the Sensor Inputs categories. Check each of the following with the key on and engine off:

- Coolant temp - is it reasonable?
- Manifold air temp - is it reasonable and within a few degrees of CLT (for a cold engine)?
- Throttle - is it close to 0% with no throttle and close to 100% with full throttle
- Fuel load - if you are running speed density, this should be close to 100kPa at sea level, down to say 80kPa at high elevations.
- Air:Fuel ratio - this won't read anything sensible with the engine off.

- Engine speed - should be zero. (Note! If RPM ever reads anything like 65000 rpm, it means you have made an error in your configuration settings. Normally this will appear as an error when you first connect. In the unlikely case that it does not, open up Communications, MiniTerminal and turn the MS3Pro off then back on again. You'll get a message explaining what is wrong. Close MiniTerminal and fix the faulty settings. Then turn the MS3Pro off then back on again.)
- Fuel pump - the fuel pump output on the MS3Pro is ground switching for a relay. When the MS3Pro is first turned on it will run for a couple of seconds. As soon as tach input pulses are seen the pump will turn on again. If the engine stalls, the pump is turned off.

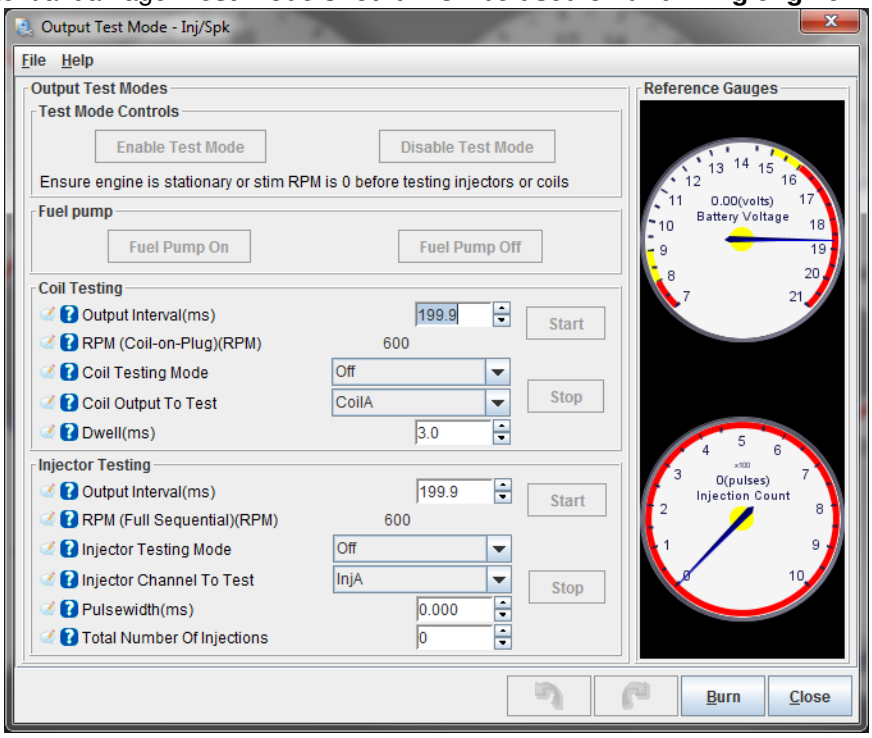
6 Tuning the MS3Pro

6.1 Getting started

OK, now that you have set up the basics of your tune file, it's time to make sure everything works and try to start the motor. First, make sure the ignition coils or ignition module do not get hot with the key on, ignition powered up, and the engine not running. If the ignition components get hot, turn them off **immediately.** You probably have the **spark output** setting backwards. Try switching this setting, click Burn, and turn the ECU off and back on again. Confirm that the problem is fixed.

6.1.1 Using output test mode

First, we'll use Output Test Mode (select it under the CAN bus / Test Modes menu) to check the outputs. The test mode allows direct control of the coils and injectors. We'll cover its full use here, although some of the settings won't be used until the engine is up and running. Used by accident, this could cause flooding of the engine and potential damage. **Test mode should NOT be used on a running engine.**



The Enable Test Mode button is locked out unless the RPM reading is zero. Click this button to enable test mode, and when you are done, click the Disable Test Mode to go back to normal operation.

6.1.1.1 Injector test mode The injector test mode can be used both to confirm the injectors are wired correctly and to use the MS3Pro as a controller for an injector test bench. If your injectors are fitted to the engine, you **must not** run the test mode with fuel pressure or you will fill your engine with fuel! Remove the fuse from your fuel pump and de-pressurize the line first. To enable injector testing:

- Select Injectors in the Test mode drop down.
- Choose your pulsewidth time in milliseconds. (To just make them click for this test, 3 to 5 ms is fine.)
- Choose which injector to test.
- Set Injector testing mode to One.
- If your injectors are powered via the fuel pump relay as is recommended, set the fuel pump to On.

- For functional testing, set the total number of injections to 65535 (the maximum) or for flow testing use your chosen number. (When performing repeated tests at the same number, you need to enter 0, TAB out, then enter the number you want and TAB out again.)

The injector testing mode called **Sequence** runs the injector in sequence up to the chosen injector. So if you have six injectors and selected Injector D and Sequence, then the test mode will repeatedly fire Injector A, then Injector B, then Injector C, then Injector D in turn, but will not fire injectors E and F. Set the chosen injector to the last one, and you can use this to confirm the injectors are following the right firing order.

The **Output Interval** determines how long to go between each test firing of the injectors. If you want the injectors to fire every 100 ms, you would enter 100. The screen shows the rpm this approximates to in full sequential.

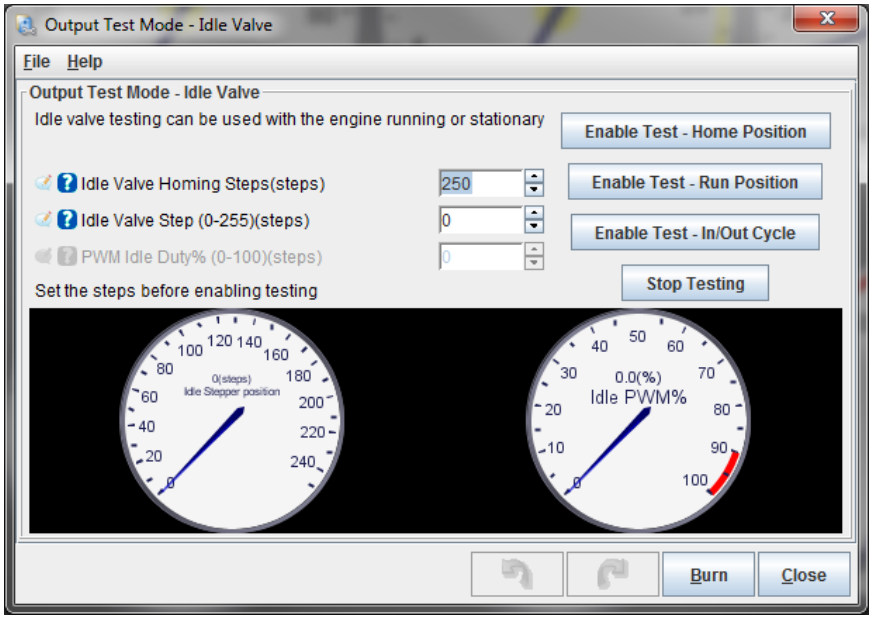
Testing the fuel pump is straightforward; just buttons under Fuel Pump to turn it off and on. You will notice that "Ready" lights up on the TunerStudio dash when the fuel pump is enabled. If you get no activity, try starting again. However, if Ready lights up but the pump doesn't run, you need to check you fuel pump relay and pump wiring. If this is a new fuel system, make sure that the fuel system doesn't leak and that it holds pressure. Note that some aftermarket fuel pressure regulators, including most from Aeromotive, will hold pressure for only about 30 seconds or so after shutdown even if the fuel system has no leaks. Most OEM fuel pressure regulators will hold pressure much longer.

6.1.1.2 Coil test mode The coil test mode is useful to confirm you have the coils wired correctly and to check dwell settings. (Do be careful as ignition coils put out a potential lethal voltage. Typically 30,000V or more.) To enable coil testing:

- Select coils in the Test mode drop down.
- Choose your dwell time in milliseconds. Start low. e.g. 2.0ms.
- Choose which coil to test
- Set Coil testing mode to One
- If your coils are powered via the fuel pump relay as is recommended, set the fuel pump to On.

The Sequence mode and Output Interval work the same for coils as for the injectors. For example, a 4 cylinder wasted spark engine at 6000rpm will fire a coil every 5ms, so each coil is fired every 10ms. Note that the test mode does not allow dwell periods to overlap in the sequence mode, unlike normal operation where the dwell periods overlap as required. (This is why traditional V8 distributor spark was weak at high revs - there was not enough time to charge the single coil between sparks.)

6.1.1.3 Idle valve testing The idle valve has a separate test mode, also under the CAN bus / Testmodes menu. The idle valve test mode can be used with the engine running or stationary, unlike the regular Output Test Mode. Note that with a PWM idle valve or a five or six wire stepper valve, you may need to run a temporary fused 12V supply to the valve if you normally take power from the fuel pump relay and are testing with the engine off.



- **Idle Valve Homing Steps** - This only applies to stepper idle valves. This setting determines how many steps the motor should move during homing to ensure it has reached the home position. The number needs to be large enough to close the valve from any unknown starting position. (This is because there is no position feedback on most automotive stepper IAC valves.)
- **Idle Valve Step** - the desired valve position if using a stepper IAC valve. Larger numbers should give a more open valve and higher rpm.
- **PWM idle duty%** - the desired valve position if using a PWM IAC valve. Larger numbers should give a more open valve and higher rpm.

(Note that some Bosch rotary PWM valves valves feature a failsafe mode so that zero flow is at a midpoint, such as 20% duty,; above that the valve operates as expected where more duty gives more airflow. Below this specific point is a failsafe mode that allows airflow even if there is no control signal. This is readily observed during test mode.)

There are three control buttons:

- **Enable Test - Home Position** - for stepper idle valves, this moves the valve through the number of homing steps back to the home position. (If it doesn't get all the way back to the stop, increase the number of homing steps.) The valve will move to the run position shortly afterward.
- **Enable Test - Run Position** - enable test mode where the steps or duty setting controls the valve position
- **Enable Test - In/Out Cycle** - Cycles the valve continuously between the homing and run positions.
- **Stop Testing** - disable test mode and allow normal idle control to operate.

Testing a stepper IAC valve First, click the Enable Test - Home Position button, and verify that the valve closes completely. The valve will move back to the run position on its own; if you are doing this test yourself and want to make sure the valve is closed, set the run position to zero. Stepper valves can be a bit more complicated to troubleshoot than PWM valves. Here are some common problems and what to do if you encounter them.

- Problem: Valve vibrates when commanded to move, but does not actually turn.
  - One coil is wired backwards; swap wire 1A for 1B, or 2A for 2B.
- Problem: Valve moves, but in opposite direction of intended.

  - Both coils are wired backwards for the way your tune is configured. With the current firmware, you can simply change the homing direction in the software. 1.2.4 and earlier code does not allow this. If you encounter this problem on older firmware, you can reverse the wiring. Switch wires 1A for 1B, and 2A for 2B.

- Problem: Valve does not move at all on initial power up.
  - One possibility is incorrect wiring. Make sure 1A and 1B are on one coil, and 2A and 2B are on the opposite coil.
  - The other usual problem is the time step size is set too small. Try increasing it and see if the valve moves.

- Problem: Idle speed is not consistent from one start to another at the same temperature.
  - Usually, this is caused by the homing steps being set too low. Increase this value until the valve retracts fully.
  - If the valve sometimes retracts fully, but not always, this may mean the steps are small enough that they do not always successfully move the motor. Try increasing the time step size.

- Problem: The valve retracts normally, but then appears to stop working at some point during operation.
  - Try increasing the time step size and/or minimum number of steps to move. Some older valves can have problems with stiction. Increasing the time step size will send a longer pulse, while increasing the minimum number of steps to move will send more pulses that can un-stick a stubborn valve.

- Problem: Idle speed hunts while the number of steps displayed is constant.
  - It's easy to assume the IAC valve is responsible here. After all, it's supposed to control the idle speed, and the idle speed is behaving wrong. However, this is likely to be an issue with the ignition timing or fueling instead.

Testing a PWM IAC valve PWM valves do not need a homing routine, as the PWM output directly controls their position. As noted above, be sure your valve has power when testing. The "PWM idle duty" box allows entry of 0-100% duty output. This could be used to confirm operation of the valve as well as determining what duty cycle gives what RPM with the engine running.

6.1.2 Checking RPM

Next, we'll disconnect the injectors and check to be sure the RPM input is working and your timing is correct. Pull the fuses to the injectors, or disconnect the connectors on the injectors, and crank the engine. With TunerStudio connected and running, turn the key on and crank the engine. If all is well you see "Synced" and a stable engine speed (rpm) number. If the pattern is noisy, incorrect or missing you might see "Synced" then "Not-synced" and/or rpm jumping up and down. The tooth logger (see section 8.3) should also be used to confirm that the pattern is stable if you do not have a steady RPM. Keep working on the crank/cam sensor inputs until you have stable rpm with no loss of sync.

While cranking, take a look at the fuel load (MAP sensor) reading in TunerStudio, if you are running speed density. You should expect to see it drop slightly as the engine pulls some vacuum. If it stays stuck on atmospheric pressure (near 100 kPa) then your map hose might be connected incorrectly.

6.1.3 Setting cranking timing

Having confirmed that the coils work as expected and with the fuel pump still disconnected, hook up a timing light to plug lead #1. Yes, this is really needed - correct timing is very important. If you are running wasted spark you need to ensure your strobe is compatible. "Dumb" timing lights are fine; some dial-back lights can be awkward. Use a "2-stroke" setting for wasted-spark or wasted-COP if you have a dial-back light. With a coil-on-plug pencil type coil you will need to remove the coil from the engine and install a short length of regular spark plug wire from

the COP to the spark plug, and possibly ground the COP if it was bolted on. Connect your timing light over this temporary plug lead.

If there are no timing marks on your engine, you will need to add some sort of marks. You can establish TDC with a piston stop and add timing marks with either paint or timing tape.

Crank the engine and confirm that #1 timing is somewhere close to the Cranking Advance number set in TunerStudio (this is set in section 5.3). If the timing doesn't match and you're in Fuel Only mode, you'll need to adjust whatever is controlling the timing. Otherwise, you'll need to adjust the Tooth #1 Angle setting (if running Toothed Wheel) or Trigger Angle / Offset (if running any other spark mode besides Fuel Only). If the timing is too far advanced, increase this number. If the timing is retarded, decrease this number. If adjustments of more than 10 degrees are required, making the change, click Burn, turn the key off and restart the engine. Very large changes don't take effect until you turn the MS3Pro off and back on again.

6.1.4 Starting the engine

Now that you've confirmed sensor inputs, coil and injector outputs, have good rpm and cranking timing is close you are ready to start the engine. Ensure the fuel pump is connected (since we disconnected it in previous tests), battery is charged, and all hoses are secure. We recommend starting a data log as well. While the engine will **usually** start if you've gone through all the previous steps for setting up MS3Pro, sometimes you'll need a couple more tweaks, and the data log can be very valuable if it comes to that. Go to the Data Logging menu, select Start Logging, and enter a name for your data log. The log starts when you click Save, and stops when you either close TunerStudio or go to the Data Logging menu and select Stop.

Now, hit the starter, and hope the moment for the big payoff has arrived.

It is common to need to give the engine a little throttle to keep it alive on the first start. If all sounds well, then keep the engine running and warm it up. Check for oil pressure, coolant temperature etc. If anything sounds wrong - stop and investigate.

Take your time! If things aren't working out, then take a breather and come back another day. Come to the www.msextra.com forum and ask for help. When posting, please include full details of your install: engine type, make, model, ignition setup. The forum is global and we might not be familiar with your engine, so just quoting an engine code is not sufficient. Don't use the forum to vent your frustration, keep it calm...

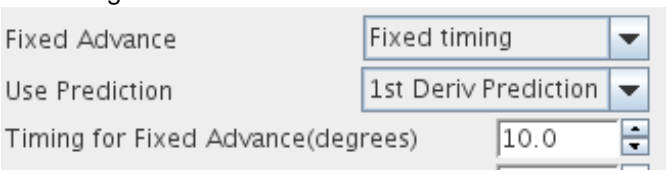
If the engine kicks back, your timing is too advanced. Disconnect the injectors again and check with a timing light. This can damage the starter motor if you ignore it.

If your engine does not spark, in spite of having a functioning fuel and ignition system and timing that appears to be correct, here are several things we recommend checking that commonly will cause a no start condition.

- If you are running full sequential fuel and / or coil on plug ignition, try changing these settings to semi-sequential fuel and wasted-COP ignition. This will rule out the possibility that the cam sensor hasn't been set up correctly, and you're firing on the exhaust stroke instead of the intake stroke. These changes will make it fire on both, which can be very useful for debugging startup problems.
- It may be refusing to start because the fuel isn't quite right. If the engine does not go above cranking RPM, try adjusting the cranking pulse width (the Cranking Pulse menu under Startup / Idle). If the engine catches but dies immediately, try adjusting the after-start enrichment (ASE Percentage under Startup / Idle). Both of these are temperature dependent, so be sure you're adjusting them at your actual coolant temperature. There's a couple rules of thumb you can use here.
  - Start at what you think may be a lean value, then work your way up in small increments.
  - Pull a spark plug after a failed start and check the electrode. If it's wet with fuel, decrease the amount of fueling. If it's dry, increase it. And if the electrode is covered with crud, your problem may just be that you need new spark plugs.
  - If the engine responds better if you give it a bit of throttle, you probably have too much fuel (because the extra air helps get your air/fuel ratio closer to what it should be). So you need less fueling. Conversely, if giving it the throttle makes things worse, you probably need more fuel.
- Check for various physical problems. Some examples I've seen include coil packs or distributors with one or two spark plug wires run to the wrong cylinders, a broken key way on the crankshaft pulley causing the timing marks to be 60 degrees off, and even a zip tie that got pinched in the intake manifold gasket next to a coolant passage, which nearly hydro-locked the motor.

6.1.5 Setting running timing

Now that the engine is running, we can come back and make the final timing adjustments. Hook up your timing light and start your engine. In TunerStudio, on the Ignition Options / Wheel Decoder menu, set Fixed timing to "Fixed Timing" and click Burn.



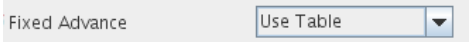
This locks the timing to the advance you specify in the "Timing for Fixed Advance" box. You do not need to alter your spark table.

- Enter say 15 degrees (or another angle that your engine will idle easily at.)
- Using your timing light, confirm that the timing on the crank matches the fixed advance you entered at idle.
- If it does not, then adjust your tooth#1 angle (trigger wheel) or trigger angle/offset (other spark modes.)
- If the actual timing is more advanced than MS3Pro thinks, then increase the trigger angle/offset. If adjustments of more than 10 degrees are required, making the change, click Burn, turn the key off, and restart the engine.
- Once you idle timing is matching up with your fixed angle, increase the engine revs and confirm that the timing does not drift.

If you get a large timing advance with increased rpm on a VR sensor distributor then it is likely that your ignition input capture is backwards. It is also possible that your **spark output** setting is wrong - however, exercise caution in changing this, as the wrong setting can damage your coils or ignition module. If the timing retards a little as rpm increase you can make a correction by adjusting the Spark Hardware Latency setting. The larger the number, the more compensation it will apply.

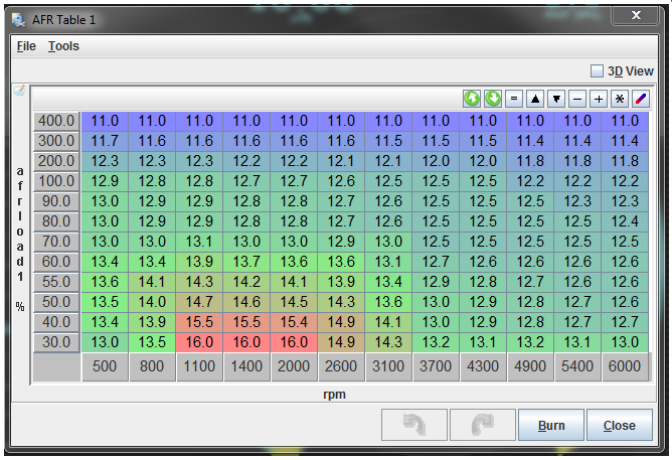
Do not proceed any further with your install if you cannot get stable timing. Note that “stable timing” is a matter of degree - a small block Chevy with a HEI distributor and a worn timing chain, for example, may have 2 or more degrees of timing jitter in normal operation. And many dial-back timing lights will have problems keeping up with rapid changes in RPM. But if you are seeing the timing bouncing around by 10 degrees or more, stop and investigate what’s wrong.

When finished, be sure to turn off the Fixed Timing and return to Use Table and click Burn.



6.2 Basics of 3D tuning tables and loads

At this point, a lot of your tuning from now on will involve 3D tuning tables, so this would be a good point to explain what these tables are and how to use them. Here’s an example, in this case, an AFR (air/fuel ratio) table.

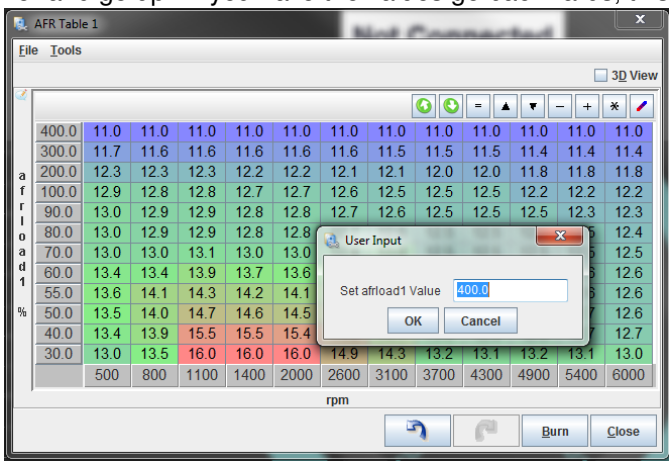


These tables are called 3D tables because they can be drawn as a shape in 3 dimensional space. The first dimension, the X axis, is pretty much always RPM for the 3D tables in MS3Pro. The Y axis is a load variable, usually expressed as a percent. It is possible to use different load methods for different tables; you don't have to use the same load type across the board.

- In speed density mode, the fuel load corresponds directly to the MAP sensor reading, and 100 kPa is taken to be 100% fuel load. Numbers over 100% are possible and indicate boost.
- In Percent Baro mode, the number is the MAP sensor reading divided by barometric pressure, so 100% is when the MAP sensor and barometric pressure readings are equal.
- In alpha-N mode, the load is the percent throttle opening.
- In ITB mode, this number is a blend between the MAP sensor and throttle position opening. We'll cover this in more detail in section 7.
- In MAF mode, the MAF load is a synthetic MAP load number that is a percentage of the air flow that the engine would draw in at 100% volumetric efficiency at its current RPM. For example, if you have a 5.0 liter Ford V8, it would theoretically draw in 2.5 liters of air every crank revolution (since only half the intake valves are open in a given rotation) at 100% volumetric efficiency. If this engine is turning 2000 RPM, it would theoretically pull in 5000 liters of air in a minute. If it's only pulling in 2500 liters a minute, you get a MAF load reading of 50% load. MAF load numbers over 100 would indicate either a race engine with a very efficient intake design or positive manifold pressure. The MAF load numbers do not directly match actual MAP sensor readings, but in the absence of a MAP sensor, they're a bit easier to get a simple 0 to 100% value for your load.
- Some 3D tables have numbers that are a direct sensor reading; for example, you can set the generic PWM outputs to work as a function of RPM and coolant or air temperature.

The third dimension is the number entered into the table.

TunerStudio lets you adjust the numbers on the load axis. Clicking on the gray boxes to the side will bring up a prompt to change the values, shown below. Note that the values start at their lowest point at the lower left hand corner and go up - if you have the values go backwards, this can cause an error.



The two green arrow buttons in the upper right hand corner allow you to save (up arrow) or load (down arrow) the table as its own separate file. An exported table will not include the rest of the tune, and can be useful if you're sending tune values from one MS3Pro variant to another.

The buttons to the right of the green arrows allow you to adjust several cells in the table at once. Simply drag the mouse to select several cells, and then click the button to adjust them. The buttons work as follows:

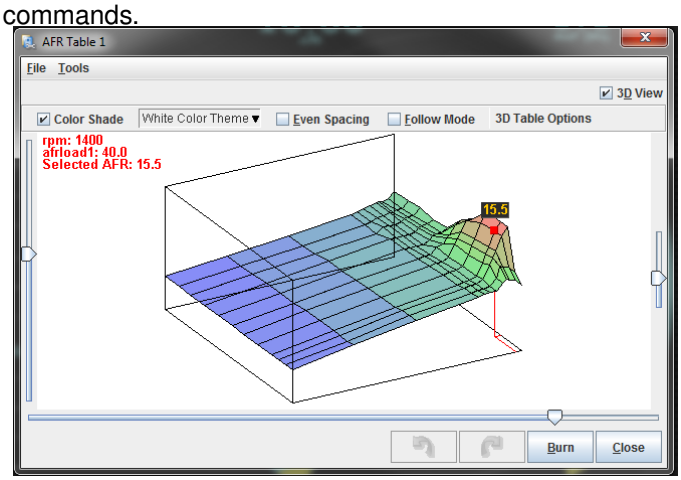
- The equal sign sets all selected cells equal to the value you enter.
- The up triangle increases all selected cells by one increment.
- The down triangle decreases all selected cells by one increment.
- The minus, plus, and multiplication signs use the math operation you requested and the number you enter on the selected cells.

- The last item may look like a divide sign, but it interpolates between selected rows instead.

The two blue arrows at the bottom of the cell stand for Undo and Redo. The Burn button writes (“burns”) the data to flash memory, and the close button closes the table. Note that changes to the cells will take effect as soon as you enter them, but if they are not burned to flash memory, the MS3Pro will not keep the changes after it shuts down.

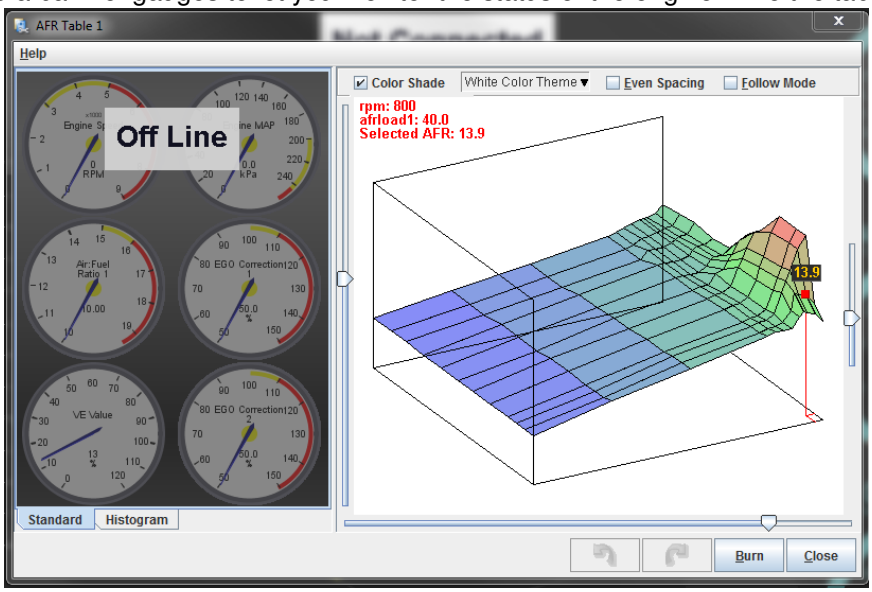
When the engine is running, a blue dot indicating the current RPM and load will appear on the table, and the nearest cells under the dot will be highlighted in yellow. The MS3Pro interpolates between the closest four cells if the engine is not directly on top of a load point.

The 3D view checkbox at the top of the table will change it to a three dimensional drawing with a different set of commands.



You can rotate the drawing with the mouse to get a better view, or press Z to toggle a flat overhead view. When the engine is running, a blue dot will appear showing what RPM and load point the engine is currently running at. The red dot shows which cell you are currently adjusting. Use the arrow keys to move the dot around the map, and shift-up arrow or shift-down arrow to move the map value up or down. Checking Follow Mode will cause the red adjusting dot to follow the blue dot around the table.

Many of the tuning maps can also be viewed under the 3D Tuning Maps menu. This brings up both a 3D table and a bank of gauges to let you monitor the status of the engine while the table is active.



You may change the gauge style to “Histogram” to change them to a line graph, or right click any gauge to swap it to a different gauge style if you want to see different variables.

6.3 Tuning fuel

Once you have the engine up and running, it's time to start tuning. There are three tools to use for tuning when the engine is in a steady state: the VE table, the AFR table, and the settings that define the injector behavior (Dead time and small pulse widths). We'll cover the last two first as they are applied across the board, so you will want these pinned down before you start tuning the whole table. After these are dialed in, you can come back and dial in the acceleration enrichment and other enrichments applied to the fueling.

6.3.1 Dead time

*****NOTE: Injector Dead-Time characterization has changed significantly as of firmware v1.6! The curves are now programmed directly in microseconds (uS) of dead-time, rather than as a percentage (%) of a 'nominal' dead-time value (often 1.0ms), as was done in all previous firmware versions. This means that when loading a tune from an earlier firmware version, you must update the dead-time curve in order to get it to 'match' your old dead-times, or else your fueling/VE table is likely to need substantial changes/re-tuning, especially at low pulse widths!*****

For a detailed step-by-step example of dead time tuning across the different firmware versions including a helpful dead-time calculator spreadsheet, please visit the MS3Pro Dead Time Tuning technical article on our website:

MS3Pro Injector Deadtime

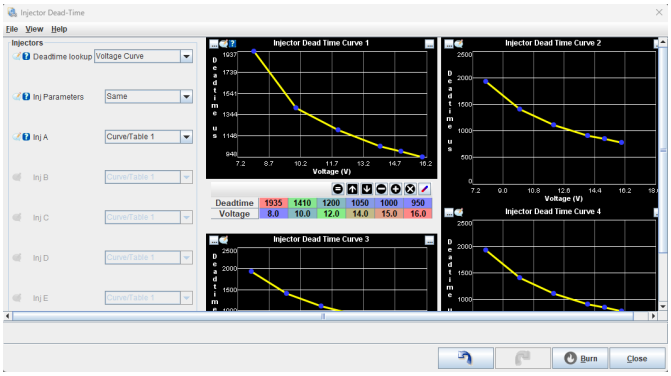
The dead time (or offset) of an injector is effectively the portion of the pulsewidth when no fuel is injected, this is because the injector is an electro-mechanical valve and takes a measurable time to open and close. (Note that documentation for some earlier MS variants referred to this as just "opening time".)

Setting the dead time correctly is an important setup step. If it is incorrectly set, you will find tuning light-load or idle very tricky and features such as air-temperature correction will malfunction. In worst cases your engine will be untuneable - especially if you have very large injectors.

The MS3Pro dead time system is found on the Fuel Settings -> Injector Dead-time. It allows for individual or grouped setting of these dead times. This will match up with the "Main fuel outputs setting" on the engine and sequential settings page.

There are three main areas on this page:

- The injector settings on the left side.
- The voltage correction curves on the right side. These allow fine tuning the battery voltage correction of the dead time.



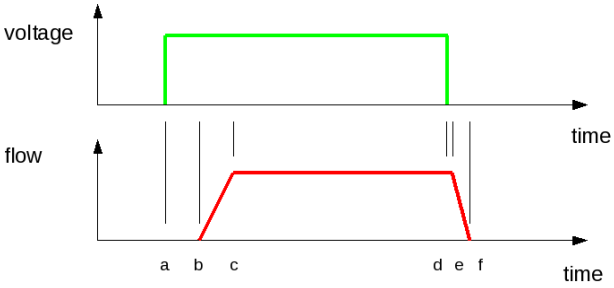
Additionally, as of firmware v1.6 you now also have the option of further characterizing injector dead-time not just on a curve as a function of voltage, but also as a two-dimensional table as a function of both voltage AND fuel pressure. Flow-tested injectors are often provided with just such a table, so if available, this provides an easy way to accurately input that measured data for a specific injector:

6.3.1.1 Injector settings As previously mentioned, the injector outputs directly support high-impedance injectors. Low impedance injectors may be used through an external peak-and-hold box or resistor pack.

The setting Same/Individual allows either one dead-time across all injectors, or if you have physically determined the precise dead time for each injector individually, you can use the Individual setting. Normally, use Same.

LPG injectors may require larger dead times.

Injector voltage / flow example



Fuel pulse on (a)
After a delay the injector starts to open (b) and the flow ramps up to maximum. (c)
When the fuel pulse turns off (d), the injector then begins to close (e)
Fuel ceases (f)

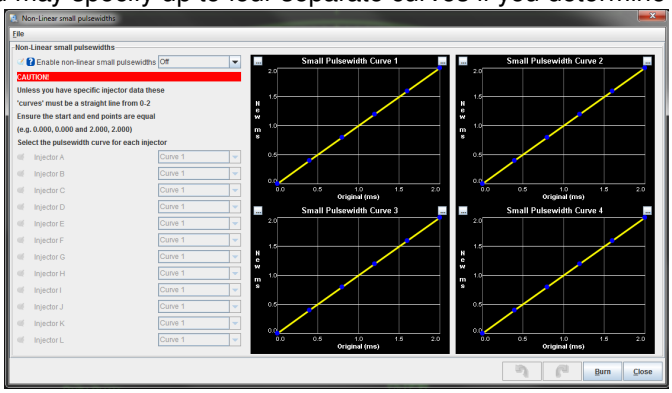
One manufacturer quoted a real opening time of 1.2ms and a closing time of 0.4ms.
This gave a dead time of 0.8ms.
It is the dead time that we care about for correct fuelling.

Say if the ECU wants to inject 7.000ms of fuel. Allowing for the dead time, the electrical pulsewidth in this example is acually 7.000 + 0.800 = 7.800ms.

We'll cover measuring injector dead time in more detail in the Advanced section.

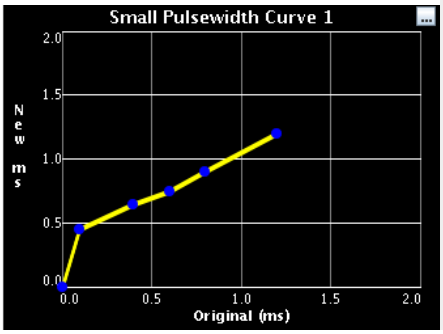
6.3.2 Small pulse widths

At most pulsewidths (say above 2ms), injectors behave in a linear manner, where 10% increase in effective pulsewidth gives a 10% increase in fuel flow. The MS3Pro normally relies on this linear behavior for fuel calculations. However, at small pulsewidths, injectors behave non-linearly. This table allows for you to correct for this. You may specify up to four separate curves if you determine that your injectors do not all work the same.



The Injector Small Pulsewidths feature allows the user to make a correction and make the non-linear region more linear in nature. Unless you have calibration data for your injectors measured on your MS3Pro, this feature should be left turned off. The X-axis of the curve is the MS3Pro pulsewidth (before dead-time is applied) and the Y-axis is the required injector pulsewidth (before dead-time) that the injector needs to approximate linearity. Most injectors observed exhibit an "S" shaped non-linear region. Nothing happens for the first few 0.1ms after dead time, so a curve that effectively adds on PW to the very low values will help linearize this somewhat. The non-linear region also frequently shows a "lump" which will make linearization tricky. See the dead time measurement section for real measured data.

Here are some example curves based on data from real injectors. Note that no allowance is made for the "lump" which will distort the results. The injectors were linear by 1.2ms (plus dead time) so the curve is arranged from 0-1.2ms.



Original (m	New ms
0.000	0.000
0.100	0.450
0.400	0.650
0.600	0.750
0.800	0.900
1.200	1.200

Ensure that the start and end points match. In this example, the curve starts at 0.000, 0.000 and ends at 1.200, 1.200.

However, the best approach is to avoid operating injectors in this non-linear region altogether. Size injectors appropriately and use a second set of staged injectors if a large dynamic range is needed.

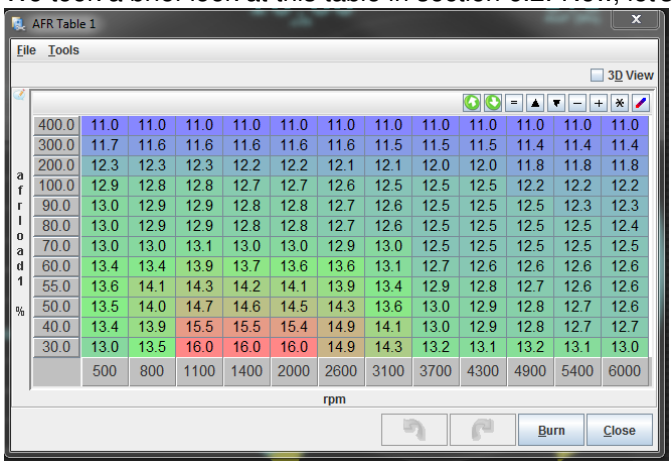
6.3.3 AFR table

The AFR table can be used in three different ways.

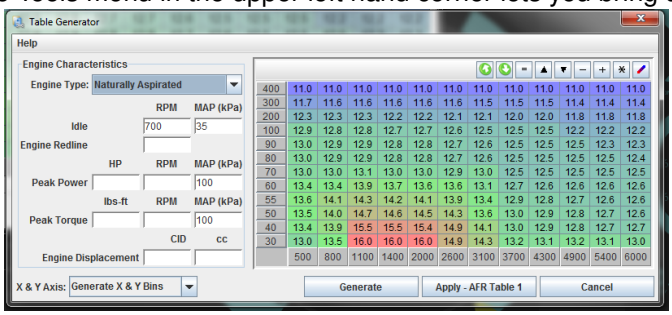
- In MAF mode, it is your primary fuel table.
- In all other fuel modes, it can be used purely for closed loop O_2 correction with a wideband O_2 sensor.
- If you set “Incorporate AFR Target” under the General Settings menu to “include AFRtarget,” it will scale the pulse widths based on the numbers entered.

We recommend using the “Incorporate AFR Target” option on most installs. This allows the VE table to closer reflect actual volumetric efficiency, and allows you to specify the air/fuel ratio by directly adjusting this table.

We took a brief look at this table in section 6.2. Now, let’s take a look at how you use this particular table.



You'll enter the AFR you want at each RPM and load point (see section 6.2 if you need a refresher on the load points.). Most sequential fire engines will run best at 14.7:1 at idle. At cruising RPM and low load, you can run 14.7:1 or sometimes leaner depending on the engine. Most naturally aspirated engines tend to run best at 13.2 to 13.5 at full throttle, while engines running forced induction may need to run as rich as the mid 11's under boost. The Tools menu in the upper left hand corner lets you bring up an AFR table generator.



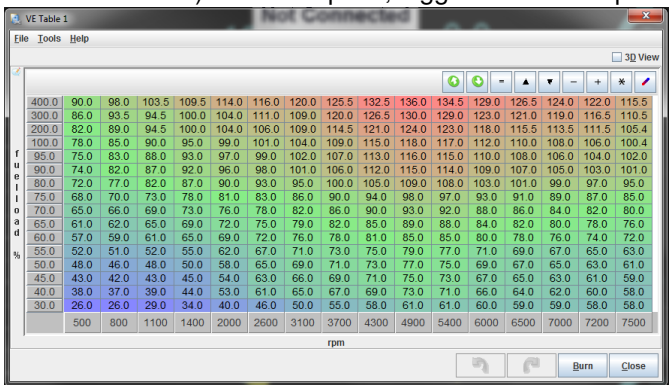
In this menu, you can enter information about the engine at left. The "X and Y Axis" option gives you the option of retaining the current load and RPM bins, or generating new ones based on the data entered. First, click Generate to create the table, then the Apply button to activate the new table or Cancel to close the generator and not use the generated table.

If you are running MAF mode, this is your main fuel tuning table. If you do not hit the actual AFR targets, you will need to adjust the MAF flow curve under the Basic / Load Settings menu.

If you set "Incorporate AFR Target" under the General Settings menu to "include AFRtarget," this table is used to scale the fueling. A table entry of 14.7:1 will mean no fuel scaling. Numbers less than that will increase the fuel, and numbers higher will decrease the fuel. When using this mode, you'll tune your VE table to hit your AFR targets, then adjusting the AFR target table will directly adjust the fuel to the desired AFR number.

6.3.4 VE table

The VE table is your primary tuning table unless running in MAF mode (and in MAF mode, it can be used as a corrective trim table). At its simplest, bigger numbers equal more fuel.



In any mode that uses this table, the numbers in the VE table are a percentage. The fueling equation takes the base pulse width from Required Fuel, scales it by the percentage in the VE table, and then applies any other corrections, enrichments, and the like, such as air density correction and warmup enrichment. If you have the tuning set to incorporate the AFR target, theoretically, the VE table will match the engine's actual volumetric efficiency. Don't be alarmed if you need to enter numbers above 100, particularly in boost. The maximum number is 255.

If you are running in MAF mode with the VE table enabled as a trim table, start by adjusting all cells to 100. The table will apply a percent trim to the fueling calculated from the MAF sensor curve and AFR table.

We recommend first setting up an appropriate AFR table, then adjust the numbers in the VE table (upwards to add fuel, downwards to take it away) until your actual air/fuel ratio hits the target table.

Before starting to tune the VE Table, it is recommended that you turn the Acceleration Enrichment (AE) off so you can see whats happening to the mixture as a direct result of the table rather than having fuel added from Accel Enrichment. See the next section on how to switch this off. You may find that you'll need to be light on the throttle, as the AE is needed to fill "holes" that occur when the throttle opens quickly. Remember to tune the AE when you have a tuned VE Table.

Another item to turn off during tuning is the Overrun settings as the PW may drop to 0.0mS causing lean spots. You'll find this by going to the Basic / Load Settings menu and selecting General Settings. Overrun fuel cut is in the lower right corner. The default tune has this turned off.

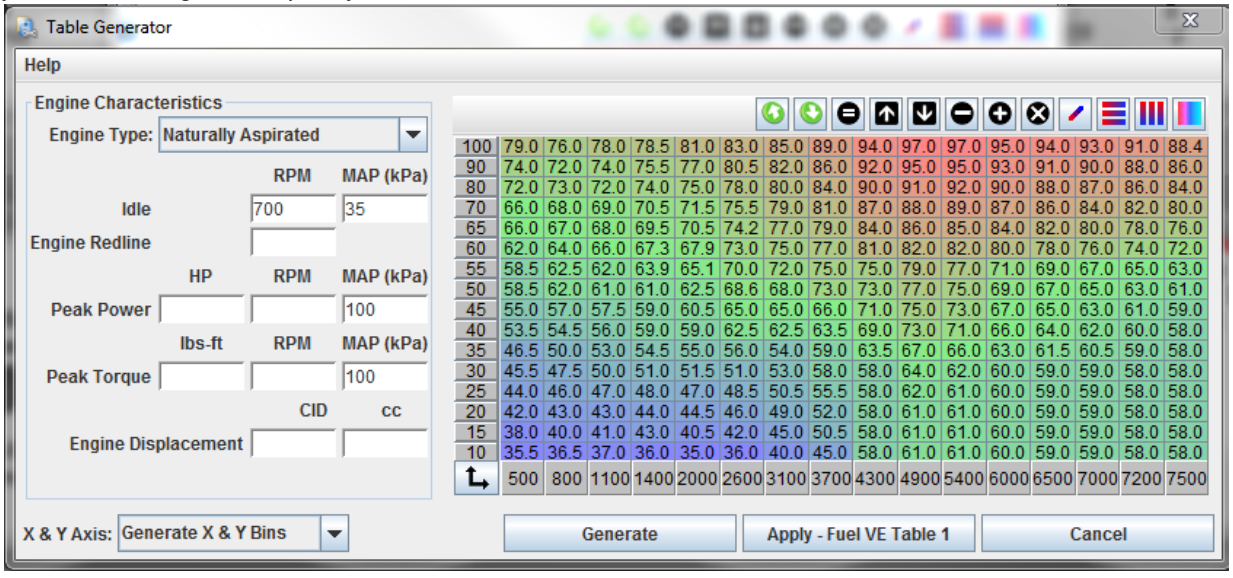
Tuning the VE table involves enriching (by increasing the VE) or leaning (be decreasing VE) at each point in the VE table. Most of your driving will occur in a diagonal strip of the VE table, from low rpm, low kPa (i.e. idle) to high rpm, high kPa (i.e. WOT). You can adjust these values using the O_2 sensor, data logs, VE Analyze Live in TunerStudio, and/or the seat of your pants. Low rpm and low kPa (say less than half of the max rpm and max kPa) might be able to use stoichiometric or leaner. Richer mixtures would be used at high rpm and high kPa.

However, the low rpm/high kPa and high rpm/low kPa are not seen as often driving your vehicle. Basically, if the engine never runs in certain parts of the MAP, then the numbers there should not matter. However, since you may not be able to guess where you will run under every possible set of conditions, you put estimated VE numbers that make sense into the little used areas.

From this frequently used diagonal strip of the VE table, you will be able to see how much the VE rises from one rpm bin to the next, and use these differences to estimate the low rpm, very high kPa numbers and the high rpm, very low kPa numbers. Since you rarely (if ever) run in these parts of the table, the actual numbers will not make much difference, but they will be there "just in case". You are looking to create a smooth VE map wherever possible.

It is a good idea to set either the second or third row of kPa bins in the VE Table to the idle MAP value. So, for example, if you idle at 35 kPa, set the first to say 20 kPa or even a little lower (this can be used for fuel saving during over run). The second can then either be 35 kPa or a value between 20 and 35, with the next row being 35. The reason is it is always easier to tune idle if the point of idle is on a cross hatch; then there's only really 2 points that influence it rather than 4. The engine may not always idle at the same MAP value when under load, but it's worth trying to set this up for unloaded idle. Most of the time, an engine will idle best if you flatten out the cells above and below where it idles; it's rare for an engine to need large changes in VE at idle.

6.3.4.1 VE table generator Clicking on the Tools menu under the VE table will bring up the VE table generator tool. This can create an approximate starting VE table based on the engine's size, induction type, and estimated torque curve. This table can be a good starting point, but will require some fine tuning to dial in. Note that entering a peak power and a peak torque that are only a few hundred RPM apart can result in a table with almost no useful data below the peak torque. The VE table generator works for fairly typical engines, but its algorithm does not work very well for an engine that peaky.

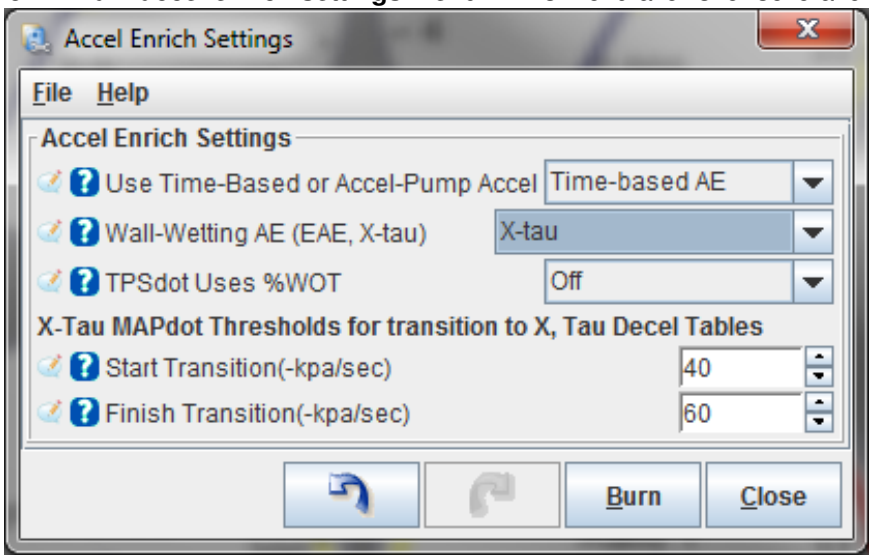


6.3.5 Basic acceleration enrichment (AE) tuning

MS3Pro has two options for basic acceleration enrichment, plus model based acceleration enrichment that we'll discuss later. One is the time based accel enrichment setup. The other one behaves similar to an accelerator pump on a carburetor. We recommend using the accelerator pump style in most cases, but we will cover both styles if you prefer the "classic" setup.

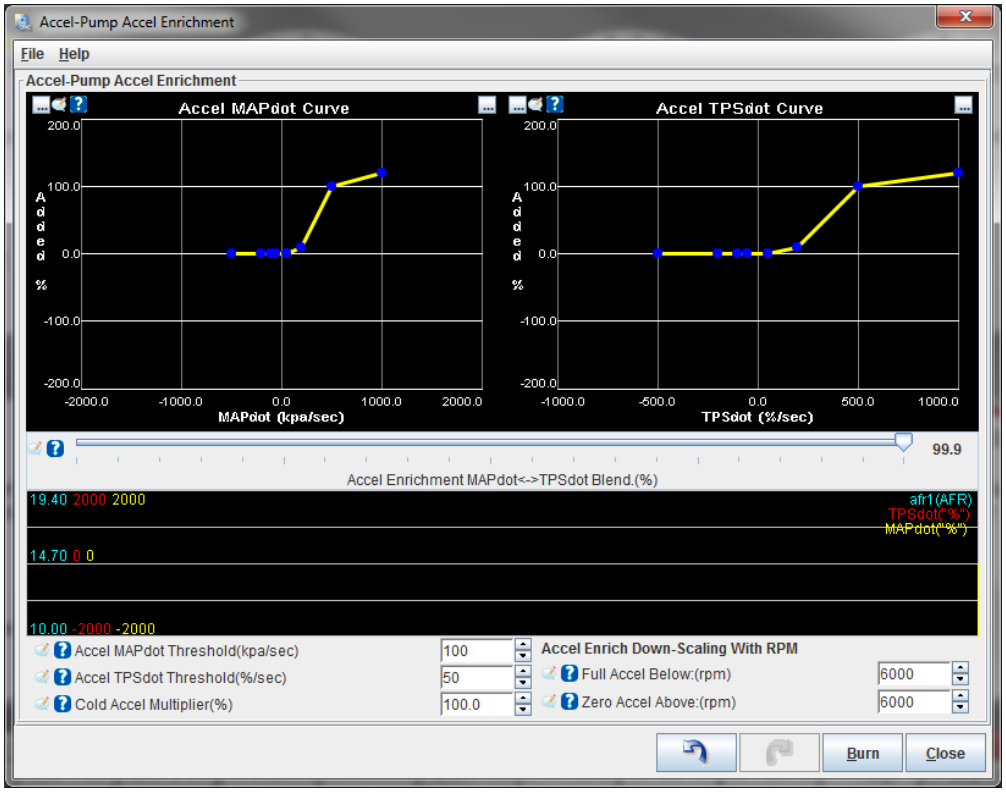
Both use variables called TPSdot and MAPdot. The “dot” is a calculus notation that indicates rate of change with respect to time. Don’t worry, you don’t need to know calculus to tune this. The amount is expressed as a percent of change over time. With TPSdot, it is percent opening versus time; a TPSdot rate of 100 %/sec would be opening from closed to wide open throttle in one second, or the equivalent speed, such as going from 25% to 75% throttle in 1/2 second. The MAPdot is measured in kPa per second. MS3Pro can use either one, or a combination of the two, to trigger acceleration enrichment.

6.3.5.1 Main accel enrich settings menu This menu allows for several changes that affect both AE modes.



- **Use Time-based or Accel-pump Accel**: This lets you select which of the two methods below will be used.
- **Wall-wetting AE (EAE, X-Tau):** Enables model based acceleration enrichment, discussed in section 7.
- **TPSdot uses %WOT**: This allows you to specify a curve that defines the upper limit of throttle position as a function of RPM. Moving the throttle beyond this point does not allow in any additional air, and therefor triggers no additional acceleration enrichment.
- **X-Tau MAPdot Thresholds**: The X-Tau acceleration enrichment, when enabled, will always trigger when the MAP readings increase. However, you can prevent this from automatically triggering when the MAP sensor decreases by specifying a minimum amount of MAPdot in kPa per second to trigger X-Tau.
  - Start Transition - A MAP decrease at a rate slower than this will not trigger X-Tau fuel cuts.
  - Finish Transition - When the MAP sensor reading drops faster than this, full X-Tau driven fuel cuts are in effect. Between the start and finish values, the MS3Pro will ramp in this feature.

6.3.5.2 Accelerator pump AE The accelerator pump AE uses one curve each for MAPdot and TPSdot driven acceleration enrichment. The enrichment starts as soon as the MAP or TPS reading changes, and stops when the reading stops changing. The % number is a multiplier which multiplies by Required Fuel to scale the pulse width up (or down; you can set negative multipliers when the MAP decreases or the throttle closes and pull fuel at this point) for acceleration enrichment. The curve lets you add a greater amount of pulse width at faster throttle openings or MAP rises. Zero on the curve means no enrichment is added at that point.



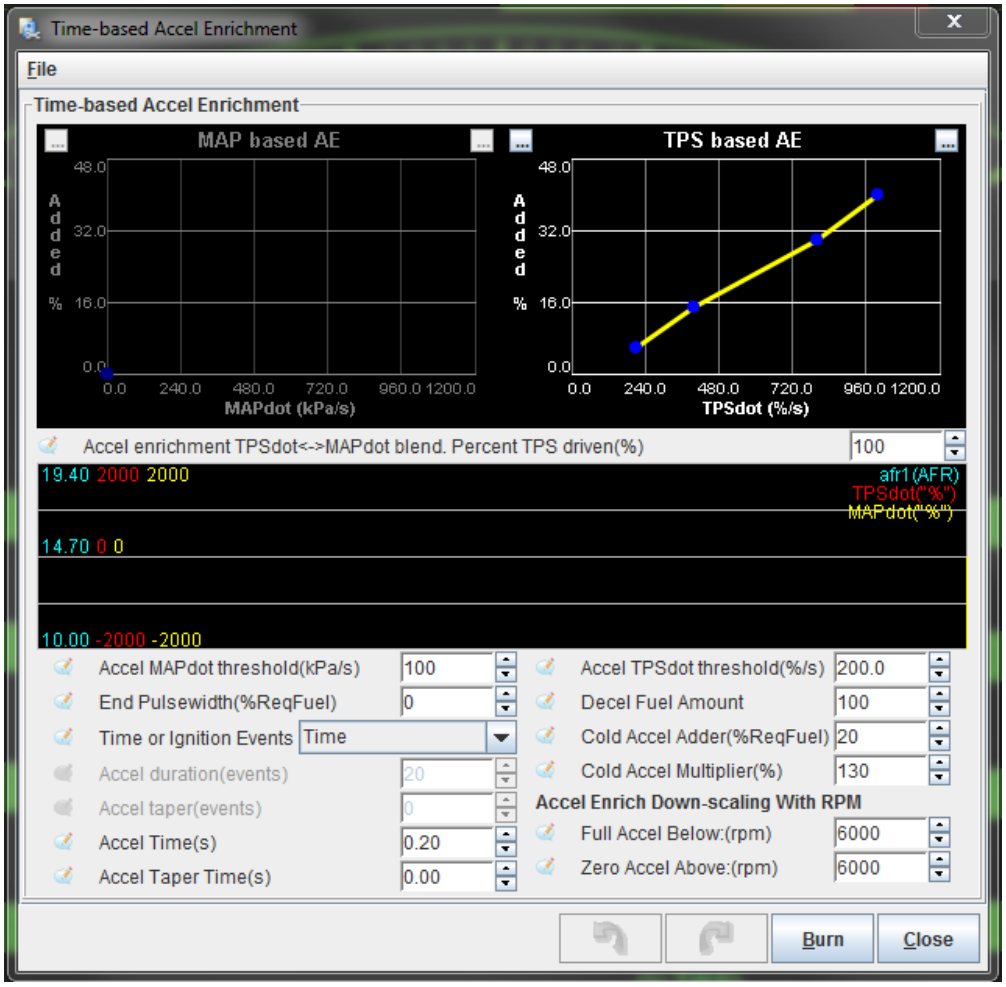
Underneath the curves, you have a line graph that appears when the engine is running. It tracks the air/fuel ratio, MAPdot, and TPSdot measurements to help you tune the acceleration enrichment.

Besides the curves, you have the following settings.

- **Accel enrichment TPS <-> MAPdot blend %:** This percentage determines what percentage of the acceleration enrichment comes from the TPS and what percentage comes from the MAP sensor. 100% is entirely TPS driven, and 0% is entirely MAP driven. In between, the percentage will scale the MAP driven pulse width adder down and the TPS driven pulse width adder up.
- **Accel MAPdot threshold:** The minimum MAPDot level needed to trigger MAP driven acceleration enrichment. Increasing this number will decrease false triggering from a noisy MAP signal, but too much will mean you miss triggering acceleration enrichment at lower amounts of tip in.
- **Accel TPSdot threshold:** The minimum TPS rate of change needed to trigger TPS driven acceleration enrichment. As with MAPdot threshold, this has similar trade-offs between setting it too sensitive and not sensitive enough.
- **Cold Accel multiplier %:** This is applied whenever warmup enrichment is active. The MS3Pro applies the full amount of multiplier when coolant temperature is at -40 degrees. It tapers off in a linear fashion to zero when you reach the point on the warmup enrichment curve where WUE equals 100% and there is no enrichment.
- **Accel Enrich Down-Scaling with RPM:** The acceleration enrichment will begin tapering off in a linear fashion once you reach the “Full Accel Below” RPM, and stop entirely once the RPM exceeds the “Zero Accel Above” RPM.

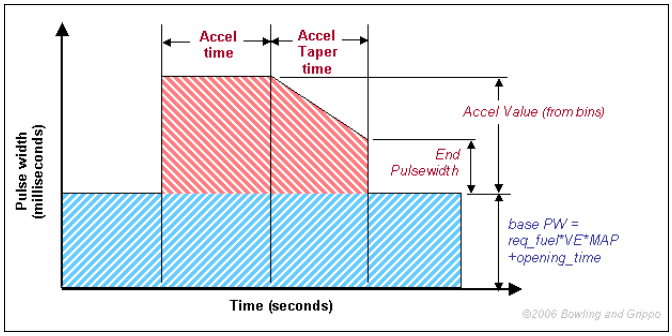
6.3.5.3 Time Based AE

This is tuned under the Time Based Accel menu.



The two curves are the primary tuning tools. They set the percentage of pulse width to add as a function of the rate of change of MAP and TPS readings. The adder % is a percentage of the main Required Fuel number. Under these charts, you can set a percentage to blend MAP and TPS driven acceleration enrichment. 100% is entirely TPS driven, and 0% is entirely MAP driven. In between, the percentage will scale the MAP driven pulse width adder down and the TPS driven pulse width adder up.

Unlike accelerator pump AE, which stops when the throttle position or MAP sensor stops moving, this mode will activate the acceleration enrichment for a defined time when the threshold rate of change occurs.



Below that, you have a line graph that appears when the engine is running. It tracks the air/fuel ratio, MAPdot, and TPSdot measurements to help you tune the acceleration enrichment.

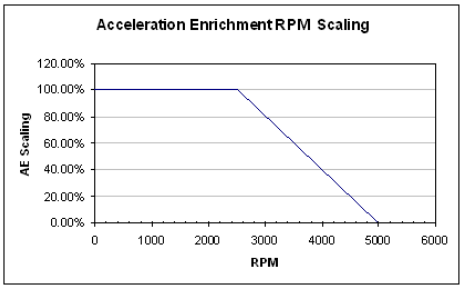
Here are what the settings do.

- **Accel MAPdot threshold**: The minimum MAPDot level needed to trigger MAP driven acceleration enrichment. Increasing this number will decrease false triggering from a noisy MAP signal, but too much will mean you miss triggering acceleration enrichment at lower amounts of tip in.
- **End pulse width:** The amount of pulse width at the end of the acceleration event, as a percentage of

Required Fuel.

- **Time or Ignition Events:** Allows you to specify the pulse width in either time or the number of ignition events.
- **Accel duration (events):** If using ignition events, the number of ignition events where you will receive full acceleration enrichment pulse width.
- **Accel taper (events):** If using ignition events, the acceleration enrichment will taper off to the specified end pulse width amount over this number of ignition events after the accel duration is complete.
- **Accel time:** The amount of time the full pulse width adder is applied, in seconds, when set to time mode.
- **Accel taper time:** The amount of time after the accel time over which the pulse width adder tapers off to the end pulse width number, if set to time mode. After this time, the amount of added pulse width drops back to zero.
- **Accel TPSdot threshold:** The minimum TPS rate of change needed to trigger TPS driven acceleration enrichment. As with MAPdot threshold, this has similar trade-offs between setting it too sensitive and not sensitive enough.
- **Decel Fuel Amount:** A multiplier that is applied whenever the TPSdot or MAPdot threshold is crossed down instead of up. 0% would trigger a total fuel cut, while 100% means no change would be applied. The firmware will allow using values over 100% to add fuel in this situation, although this is usually not a desirable effect. We recommend starting with 100% for a manual transmission or 90% for an automatic transmission.
- **Cold Accel Adder:** This is applied whenever warmup enrichment is active, and is added to the standard PW adder. The MS3Pro applies the full amount of added pulse width when coolant temperature is at the lowest temperature point in the warmup enrichment curve. It tapers off in a linear fashion to zero when you reach the point on the warmup enrichment curve where WUE equals 100% and there is no enrichment.
- **Cold Accel Multiplier:** This is applied whenever warmup enrichment is active, and is applied to scale the pulse width by a multiplied amount before the cold accel enrichment number is added. It tapers off in the same way as the Cold Accel Enrichment.
- **Full accel below:** Below this number, you will get full acceleration enrichment.
- **Zero accel above:** Above this number, the acceleration enrichment is completely off. In between these two numbers, the acceleration enrichment will taper off in a linear fashion.

The curve below illustrates how this works.



6.3.5.4 Getting it dialed in Regardless of which of the two settings you use, many of the aspects of acceleration enrichment tuning will be the same. Before you start tuning the acceleration enrichment, make sure your VE tables are dialed in. Adding a correctly tuned amount of acceleration enrichment to an incorrectly tuned fuel table is still going to result in an incorrect amount of fuel. Since the acceleration enrichment is stacked on top of the fuel the ECU is commanding based on the VE table, you'll want to tune AE last, after your VE table is fully tuned.

The standard AE can be thought of like the accelerator pump on a carb, which shoots in some extra fuel when you suddenly hit the throttle to compensate for the massive influx of air when you open the throttle blades. The goal is to compensate for this influx of air for an instant during that transient until the the transient is over and the VE tables are again providing the proper amount of fueling. If your background is tuning carburetors, you can think of this as a very powerful fully adjustable/tunable accelerator pump that lets you compensate for more than one rate of throttle position change. Slow "tip-in" can be adjusted with a certain PW adder, medium with another larger PW shot, mid-fast with another size PW squirt, and very fast throttle movements with another.

MAP based or TPS based AE? If you have a variable Throttle Position Sensor, we'd recommend 100% TPS based AE as a starting point. MAP based AE does the trick if you're running an engine without a usable TPS, but TPS based AE is generally easy to tune to the proper sensitivity level such that it activates when you want it to, and not when you don't.

Setting the TPSdot/MAPdot Threshold Levels First up, you need to determine what your TPSdot or MAPdot threshold will need to be. Below the line graph charts, you'll see a TPSdot threshold and a MAPdot threshold. These are the minimum amount of change of TPS position or MAP pressure reading, respectively, that are needed to trigger acceleration enrichment. Your goal is to set these high enough that the acceleration enrichment won't accidentally trigger at steady throttle, but low enough that it's sensitive enough to catch small or slow throttle movements as well. Setting it too high can mean the AE won't kick in when you need it.You want it as low as you can get it without it causing AE to trigger when it's not supposed to, such as during steady throttle.

Getting this dialed in can be done methodically, or by trial and error. Here's the methodical method. You'll need to have a safe way to accelerate from the bottom to the top of a gear, preferably your 1:1 ratio gear, and you'll want to datalog your engine idling, as well as these WOT pulls with your MS3Pro and TunerStudio. Ideally this is done on a dyno. Two to three pulls is better than one, but there's no need to get carried away. Then you can stop the datalogging and load this log up into MegaLogViewer.

In MegaLogViewer, on the left using the drop-down boxes that let you select what data you are viewing in your logs, set one of these to TPSdot, and/or MAPdot. You can set another to TPS to show you what throttle position you were at as well. Additionally, monitoring injector pulsewidth (select PW in the log to view that data) is helpful too, as you'll be able to see the commanded injector pulsewidth including the extra fuel added during an AE event. Lastly, near the bottom right corner of MegaLogViewer you'll see four indicators. Accel, Decel, Bit 7, and Bit 8. If either Accel or Bit 7 are active, that's a Accel event and AE is adding fuel. If Decel or Bit 8 are active, that's a Decel event and AE is pulling fuel if configured to do so. (Accel and Decel are used for TPS Driven AE, and Bit7/Bit8 are for MAP Driven AE).

Next, look at the datalog section of the car idling. The TPS signal should be rock solid, no movement. If it's jumping around, you have a noise issues on your TPS signal. This can be caused by ECU sensor signal wiring that's been routed too close to high voltage sources such as ignition coils, plug wires, the distributor, etc. Re-routing those wires away from those high-voltage sources is **always** a good idea. Additional possibilities for noise to get in the TPS signal could be a bad sensor ground (be sure you've grounded it the MS3Pro sensor return wire), or a faulty TPS sensor. Similarly noise on the MAP signal could be electrical, but is more likely to actually be real movement in the manifold pressure, at least at the source you're grabbing the vacuum signal from, meaning the port on your intake manifold that's connected to the MAP sensor.

If you're seeing AE events at idle, your TPSdot or MAPdot threshold is set too sensitive (too low of a number). You'll see the pulse width bounce around when this happens as well; AE is just doing what you told it to do.

Now look at the datalog sections for the WOT pulls, step through these in the logs and look for any time AE activated (Accel or Decel). It shouldn't be active at all on WOT pulls, because you're at Wide Open Throttle, you're not moving the throttle around, so AE is not needed. Again, if you are seeing AE activate during WOT pulls, your threshold is too sensitive.

In either case if you're triggering AE when you shouldn't be, you'll want to increase the threshold value a bit above the highest TPSdot (or MAPdot) value you see it reach in the logs. This is the lowest you can set your threshold in order to be a sensitive as possible while not triggering at steady state throttle (or idle, which is also steady state throttle, it's the steady state of NO throttle).

Note this setting is also the same setting you should use for your lowest (slowest rate of change) TPSdot or MAPdot field in the next section.

Tuning the tables Now that you have these, it's time to tune the PW adder line graphs. To tune these, start with the lowest speed throttle change position, and practice "tipping in" at that rate of speed. Use the real-time line graph at the bottom of the page to see how fast you're tipping in. Once you are able to consistently obtain a TPSdot reading equivalent to your lowest TPSdot on your acceleration enrichment curve, watch your wideband O_2 sensor readout. You want to be able to determine what's happening when you stab the throttle at that rate of speed. Is it going lean or rich? Note that it's hard to see this on a gauge, as often it will go one then the other really fast. You want the first response; if it goes lean then rich you'll usually see it hit 20:1 or maybe 22:1 for just an instant, and then as it goes back to running purely the table it may go to 13:1 or whatever... you want to watch that initial response, and if the first response is to go lean, then you want to add fuel; if the first response is to go rich, you want to reduce fuel (PW).

A wideband O_2 sensor is immensely helpful here, but don't get too caught up in seeing perfect AFR numbers through the whole transient event. Adjust it until it feels right, snappy and responsive with no hesitation. The feel is more important than looking for perfect numbers. If you want perfect numbers, you may want to try the model based acceleration enrichment. Setup basic AE first as described here. Next you'll adjust the other set points you've setup for faster throttle rate changes. Repeat the process for the medium speed tip in, then the mid-fast, then the fast.

6.3.6 Startup / warmup fueling

On the first couple starts, you'll want to use whatever combination of fuel and throttle can nurse the engine along until it is warmed up, and dial in the fuel at idle while hot. The reason is that this setting can affect startup behavior, and you will want to have good, solid values in the VE table (at least at idle) before you can get final values for the startup sequence. Here is the sequence that MS3Pro fueling goes through while starting.

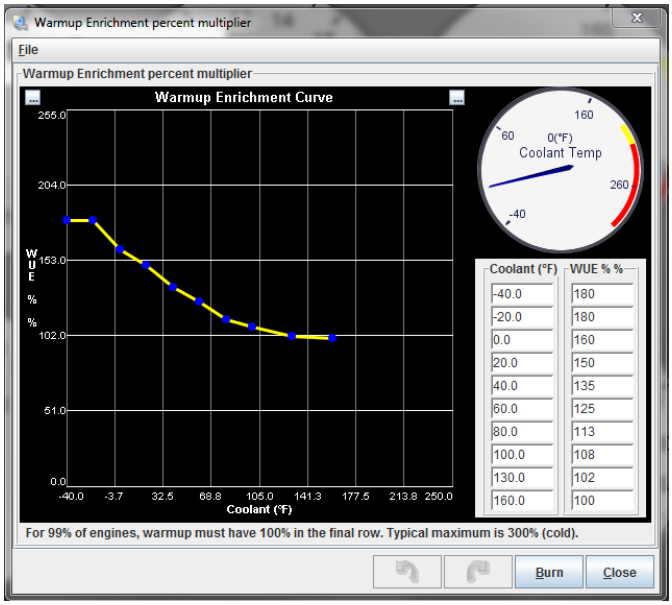
1. When the key is on, the MS3Pro will turn the fuel pump on for 2 seconds and fire one priming pulse to clear air for the lines. (You can disable this feature by setting the priming pulse to zero.)
2. When the engine starts turning, the MS3Pro will run off the cranking pulse width table in batch fire mode until the RPM rises above the cranking RPM threshold.
3. Once the RPM rises above the cranking threshold, the MS3Pro transitions to the main fuel table. It will initially multiply the pulse width from this table by the warmup enrichment (WUE) and after-start enrichment (ASE) combined.
4. The ASE tapers off based on a user defined curve. After the ASE tapers away, the MS3Pro will run on the main fuel table multiplied by WUE.
5. When the engine reaches normal operating temperature, as defined by the WUE curve, the engine will run on the main fuel table, and the warmup cycle is over.

Now, let's have a look at the menus in more detail.

6.3.6.1 Cranking / Startup settings

- Cranking RPM: This is the RPM where the MS3Pro uses cranking pulse width. You want this to be high enough that the engine does not exceed this if turning at its "cranking, but doesn't catch" speed, but less than the speed it will settle into as soon as the engine has started.
- Flood clear TPS: When the throttle is open above this percentage below cranking RPM, the MS3Pro will conclude you are trying to clear a flooded engine, and turn off the cranking pulse width.
- Cranking fuel pulse rate: The injectors are batch fired during cranking. This parameter lets you inject fuel every ignition event or every other ignition event at double the cranking pulse width number. Usually you would only use the "Alternate events" setting on engines with more than eight cylinders.
- Priming pulse delay: This allows the MS3Pro to wait a number of seconds after key on to fire a priming pulse.
- Ignore MAT correction during ASE: This can be used to compensate for MAT sensors that absorb heat from the engine. Turning this on will cause the air temperature correction to reset to 100% until the ASE taper is over.
- ASE count units: You can specify the length of the after-start enrichment in engine cycles (every other crank revolution for a four stroke) or tenths of a second.

Most of these use a tuning curve that specifies the amount as a function of RPM, like the chart below. This one shows warmup enrichment; the other screens work the same.



On the left side, you have a graphic representing the amount on the Y axis and the temperature on the X axis. The right side of the screen provides a handy temperature gauge and a table representation. (Click the "..." button in the upper right hand corner to hide or show the table.) You can either edit the numbers in the table or drag the nodes in the curve. Note that the entries **must** be set up so that the temperature data points on the table are in order, from the coldest at the top to warmest at the bottom. Accidentally setting the numbers in reverse order will result in incorrect operation.

6.3.6.2 Priming pulse This curve allows you to specify an initial pulse, in milliseconds, that is fired on the first turn of the key to clear air from the fuel lines. You can set the pulse width as a function of coolant temperature.

6.3.6.3 After-start (ASE) percentage This is a curve of percentage multiplier versus time. The MS3Pro will apply the full amount immediately after RPM climbs past cranking RPM and then taper it off linearly through the ASE taper time. This number is added to the WUE number and then the combined number is multiplied by the number the main fuel loop calculates. 0% is no after-start enrichment.

6.3.6.4 After-start (ASE) taper This curve is used to specify the amount of time that ASE is applied, in either cycles or tenths of a second. You specify which unit to use under Cranking / Startup settings.

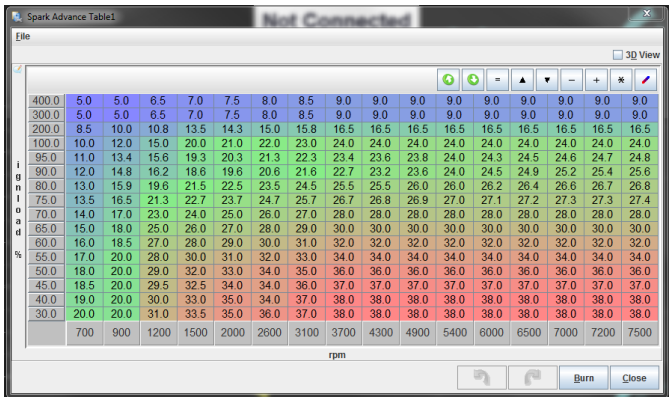
6.3.6.5 Warmup enrichment This curve is a percentage used to scale the fuel as a function of engine temperature. 100% is no enrichment; lower numbers will take away fuel. This is not quite the same behavior as ASE, which cannot be set to take away fuel. Just remember that ASE is an added number, while WUE is a multiplier. Note that if the top number on this curve is not 100%, the engine will not come out of warm up enrichment.

6.3.6.6 Putting it all together If the engine isn't starting, pay attention to the RPM it runs through. If the engine does not catch, but cranks at a steady RPM, you will need to adjust the cranking pulse width. If the engine fires up, but then stalls in a few seconds, you will need to adjust the after-start enrichment. You cannot rely on an O_2 sensor feedback for cranking, but here are some rules of thumb you can use.

- It's better to start off at a point you expect to be lean and add fuel in small increments, to avoid flooding the engine.
- If giving the engine a small bit of throttle helps, you probably have too much fuel. Conversely, if this makes it worse, you probably have too little.
- You can pull a spark plug and check if it's wet with fuel, in which case you need to reduce the fueling at the point where it stalled, or dry, in which case you need to add more fuel.

6.4 Tuning spark

Ignition is tuned on a 3D table like the AFR and VE tables.



	700	900	1200	1500	2000	2600	3100	3700	4300	4900	5400	6000	6500	7000	7200	7500
400.0	5.0	5.0	6.5	7.0	7.5	8.0	8.5	9.0	9.0	9.0	9.0	9.0	9.0	9.0	9.0	9.0
300.0	5.0	5.0	6.5	7.0	7.5	8.0	8.5	9.0	9.0	9.0	9.0	9.0	9.0	9.0	9.0	9.0
200.0	8.5	10.0	10.8	13.5	14.3	15.0	15.8	16.5	16.5	16.5	16.5	16.5	16.5	16.5	16.5	16.5
100.0	10.0	12.0	15.0	20.0	21.0	22.0	23.0	24.0	24.0	24.0	24.0	24.0	24.0	24.0	24.0	24.0
95.0	11.0	13.4	15.6	19.3	20.3	21.3	22.3	23.4	23.6	23.8	24.0	24.3	24.5	24.6	24.7	24.8
90.0	12.0	14.8	16.2	18.6	19.6	20.6	21.6	22.7	23.2	23.6	24.0	24.5	24.9	25.2	25.4	25.6
80.0	13.0	15.9	19.6	21.5	22.5	23.5	24.5	25.5	25.5	26.0	26.0	26.2	26.4	26.6	26.7	26.8
75.0	13.5	16.5	21.3	22.7	23.7	24.7	25.7	26.7	26.8	26.9	27.0	27.1	27.2	27.3	27.3	27.4
70.0	14.0	17.0	23.0	24.0	25.0	26.0	27.0	28.0	28.0	28.0	28.0	28.0	28.0	28.0	28.0	28.0
65.0	15.0	18.0	25.0	26.0	27.0	28.0	29.0	30.0	30.0	30.0	30.0	30.0	30.0	30.0	30.0	30.0
60.0	16.0	18.5	27.0	28.0	29.0	30.0	31.0	32.0	32.0	32.0	32.0	32.0	32.0	32.0	32.0	32.0
55.0	17.0	20.0	28.0	30.0	31.0	32.0	33.0	34.0	34.0	34.0	34.0	34.0	34.0	34.0	34.0	34.0
50.0	18.0	20.0	29.0	32.0	33.0	34.0	35.0	36.0	36.0	36.0	36.0	36.0	36.0	36.0	36.0	36.0
45.0	18.5	20.0	29.5	32.5	34.0	34.0	36.0	37.0	37.0	37.0	37.0	37.0	37.0	37.0	37.0	37.0
40.0	19.0	20.0	30.0	33.0	35.0	34.0	37.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0
30.0	20.0	20.0	31.0	33.5	35.0	36.0	37.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0

This table specifies the spark advance in absolute numbers in degrees BTDC (before top dead center) as a function of RPM and load. The timing you see with a timing light should match the timing table, unless there are spark trim tables in effect that make changes to it. We ordinarily recommend tuning the spark map on a dyno, but here are a few general pointers for an initial setup.

The idea is that the idle and low speed areas are set to around 8-20deg. Usually idle will be around 8-15°, but this depends on your engine's design. If you set the first row in the RPM range as a little under your usual idle (e.g. 600 if your engine idles at 800ish) and add some advance here this can help stabilize the idle, so if the engine stumbles into this area the slight increase in advance will help it to speed up a little so it doesn't stall. The cruising area of the map should have a reasonably high advance, (low to high 30's) as the mixture will be reasonably lean and therefore will give a slower burn. The overrun area can have an even greater advance, this will allow you to run lean in that section. At Wide Open Throttle (WOT) the spark map needs to be RPM based (analogous to centrifugal advance on an old style distributor) coming in at the right rate relative to engine RPM. Typically, you want it "all in" by about 2800-3200 RPM for a street performance motor, although many engines will benefit from even more advance after the torque starts decreasing, something you can't do with a mechanical advance.

Note the the optimum amount of total advance is not necessarily the most that doesn't cause detonation. For example, with a modern cylinder head design, you might get maximum power at 32°BTDC on a large V8, but might not experience any detonation until 38°- 40°. However, you will still want the advance to come in as quickly as possible (without knocking) up to 32°. This number won't be 32 degrees for all engines; for example, we've found that the new Chrysler Hemi can have problems at more than 22 degrees of timing at full throttle, and many small bore engines with pent-roof combustion chambers run numbers in the mid 20 degree range.

When going into boost the advance table, values will need to be lower than when out of boost. A common rule of thumb is that 1 degree of advance should be removed for every 2 psi of boost; this is simply a rough guide, and lots of things can depend on how much to remove. Advance below 100% load is often simply the flip side of retarding the timing under boost; as cylinder filling decreases, you will need more timing.

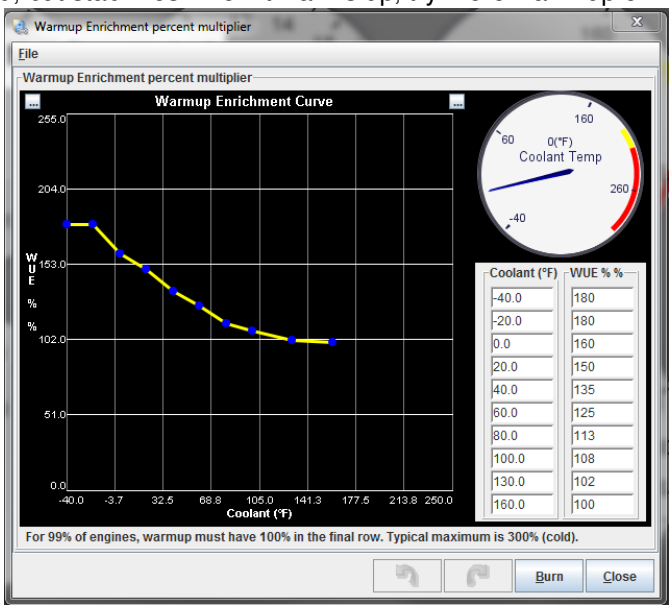
6.5 Getting a good idle

Idle tuning has several components. One is how far the throttle is open - this can be adjusted with the **Idle PWM Duty Table** or **Idle Steps** if you have an IAC valve, which lets you specify the valve opening amount in terms of temperature. If you don't have an IAC valve, you can use a set screw to adjust the throttle blade angle. There is also closed loop idle tuning, which we will cover later in the manual. Closed loop idle tuning targets a specific idle speed. But first, you need to get a stable idle without closed loop tuning. If your idle is already hunting because of a bad tune, putting closed loop on an idle that already has trouble will make it worse. We'll take some time here to explain how to tune your MS3Pro for a smooth idle.The first key may surprise many novice tuners: **The idle control settings are not the most important thing to adjust to get a smooth idle.**

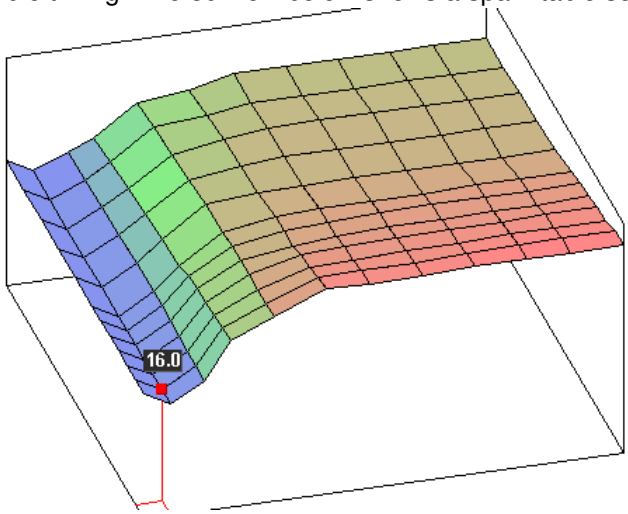
This may seem counter-intuitive, but consider this: If the engine is idling at constant load with the throttle held at a constant opening, shouldn't you have a constant idle speed? If your idle surges or hunts under these conditions, and your idle control valve is staying at one opening value, the idle control settings are not the problem. To get a steady idle, you need three things in place: correct fuel, correct timing, and correct airflow. These problems should be addressed individually.

6.5.0.1 Correct Fuel Engines don't like to idle lean; a lean air/fuel ratio is one of the biggest causes of a surging idle. On an engine with sequential injection, this is simple enough - get the engine idling at 14.7:1 for gasoline, and you're good to go. Batch fire engines, however, have issues at idle with intake pulse reversion sending some of the fuel into an adjacent cylinder. So, if your setup only allows batch fire, you'll need to add a little more fuel so all the cylinders can stay at 14.7:1 or richer. Batch fire engines typically idle best in the mid 13's, so if you're leaner than that, put in some more fuel and see if that stabilizes the idle.

You'll need to make sure the air/fuel ratio is maintained during cold start. If your idle hunts when the engine is cold, but stabilizes when it warms up, try more warm up enrichment.



6.5.0.2 Correct Timing Timing is a very useful tool for controlling your idle speed. Ideally, you'll want the timing to be set to slightly below the timing that would give it the most power - on most piston engines, this typically works out to somewhere in the 10 to 18 degree range. Then you can add a row around 300 RPM below your target idle speed if you find the idle to be unstable. The timing in this row can be bumped up by around 2 to 4 degrees over the idle timing. The 3d view below shows a spark table set up this way.



Setting up your timing like this has a stabilizing effect on the idle. When the idle speed drops, the timing advances, increasing power and moving the idle speed back up. The result is an automatic feedback loop that helps maintain a constant idle speed. Not every engine really needs this. However, you need to look out for timing curves that do the opposite. If you've specified less timing below the target idle speed than you have at the idle speed, this often makes for an unstable idle as the ECU pulls timing while the idle speed falls. This can make a slight disturbance in idle RPM amplify itself into an idle speed that cycles up and down rapidly.

Too much timing can be trouble. If the timing at your regular idle speed already puts you at the most power

output, you have nowhere to go but down. Over-advanced timing will hurt power as well - beyond a certain point, adding more timing will not bump up the idle, and taken to extremes, it can do the exact opposite. So you want to "hold back" a little timing at your idle RPM and run less than the amount of timing that would give it the most power.

6.5.0.3 Correct Airflow You'll need to get the right amount of air to maintain a good idle speed. How to do this depends on what sort of IAC (idle air control) valve you have.

No IAC valve at all? Yes, you can run without an IAC valve. Just use your idle speed set screw to adjust it open enough that the engine doesn't stall on cold starts. Sure, your idle speed with the engine warmed up will be higher than it could be, and you'll have no way to adjust for stuff like the A/C or electrical loads, but it'll work.

On/off IAC valves give you two choices: On fast idle, or off it. You'd set the idle speed set screw to your desired idle speed with the engine warm, and if your on/off valve is big enough, turning it on will keep the engine from stalling on a cold start. No closed loop control is possible. You can even add a second on/off IAC valve alongside an existing IAC to kick up the idle when the A/C turns on. Nissan used the on/off A/C valve on many early '90s cars, although they paired it with a PWM IAC valve.

PWM IAC valves or **stepper motor IAC valves** are continuously variable. This lets you adjust the amount of opening time as a function of temperature. There are two ways these can be run. Open loop mode (also called warmup mode) simply tells the valve to open a set amount as a function of temperature. Closed loop mode tries to maintain a target idle speed. We'll cover closed loop idle tuning and several of the MS3Pro's other idle control features, such as increasing the idle speed when the A/C or cooling fan come on, under the Additional Items.

7 Advanced Tuning

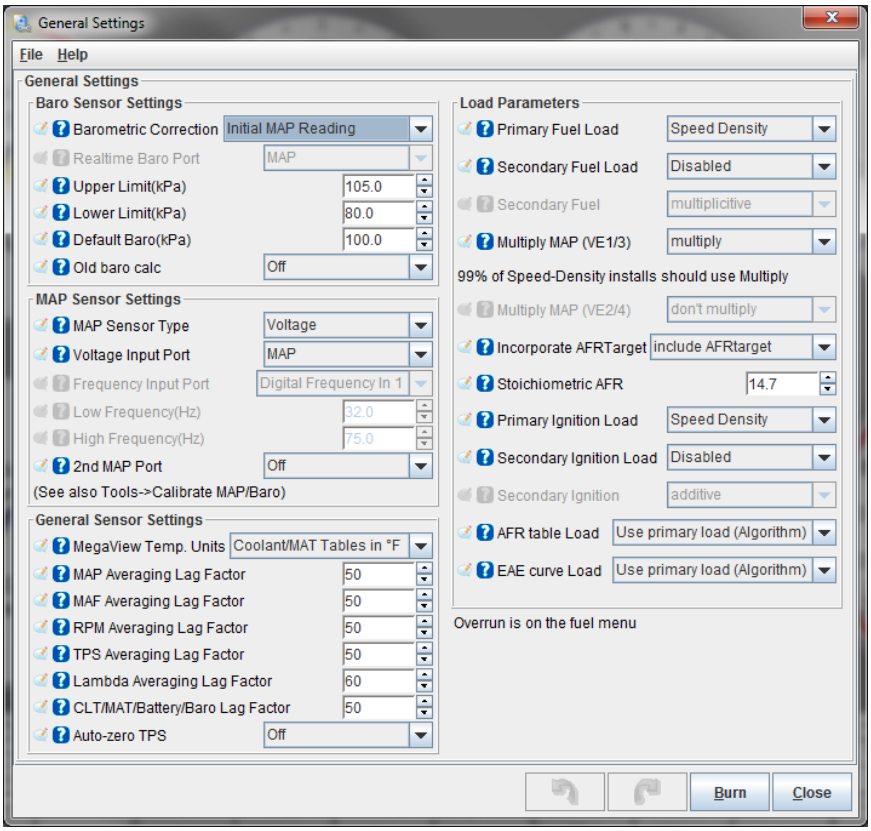
Going beyond basic fuel and ignition control!

The MS3Pro will control a lot more than basic fuel and ignition. In this section, we'll cover what else the MS3Pro can do, one menu screen at a time. We'll skip the menu screens that were set up in sections 5 and 6, and cover the rest here.

7.1 Basic / Load Sections

This section covers many basic aspects of the engine setup as well as very commonly used input and output features.

7.1.1 General Settings



7.1.1.1 Barometric settings

- **Barometric correction** For vehicles that experience altitude variations, this setting enables an automatic adjustment to fueling. See also the baro correction curve.
  - "None" - disables the feature, if adjustment is not required or desired.
  - "Initial MAP reading" - takes a reading from the primary MAP sensor at power on and uses that single value. If your altitude changes greatly during a single journey, you may need to stop and restart to take a fresh reading. Caution - engines that have severe voltage drops while cranking may cause the ECU to reset while cranking, resulting in a false low barometric pressure reading.
  - "Two independent sensors" - uses a second sensor which is sampled in real time to update the barometric correction continuously.
- **Realtime baro port** - when two sensors are enabled, this selects the input port you have the second sensor wired to.

The next two settings are used to limit false readings in case of sensor failure.

- **Upper limit** - the maximum barometric pressure allowed.
- **Lower limit** - the minimum barometric pressure allowed.
- **Default baro** - the typical baro pressure for your region. This is used as a the barometer value if you have not enabled a sensor.
- **Old Baro Calc** - Barometric correction has been improved for 1.4.0 and later firmware. If loading a tune from 1.3.4 firmware or older, turn this on to maintain backward comparability.

7.1.1.2 MAP sensor settings

- **MAP sensor type** - "Voltage" (by far the most common) or "Frequency" (used on a few Ford or Cosworth engines) are supported. The most common frequency based MAP sensor most enthusiasts may encounter is the one used on Fox body Mustangs.
- **Voltage input port** - where the primary MAP sensor is wired (when voltage is selected.)
- **Frequency input port** - where the primary MAP sensor is wired (when frequency is selected.)
- **Low frequency** - the frequency corresponding to the minimum pressure, if using a frequency based MAP sensor.
- **High frequency** - the frequency corresponding to the maximum pressure, if using a frequency based MAP sensor.
- **2nd MAP port** - enables a secondary voltage MAP sensor and selects where it is wired. If using two MAP sensors, the MS3Pro compares both MAP sensors and uses the lower one as the reading for the engine.

7.1.1.3 General Sensor Settings The MegaView was an add on display device from the DIY line of MegaSquirt products, although the availability of inexpensive Android tablets has eliminated most demand for it. MegaView displays built for the MegaSquirt-II will also work with MS3Pro.

- **MegaView temp units** - Choose degC or degF to match your MegaView.

Lag factors force the variables to change more slowly than the actual input value, smoothing them and reducing the effect of noise in the signals. Note that in all cases, 100% is no lag effect at all, and smaller numbers slow the input response speed. Setting a lag factor to zero will completely paralyze the input and prevent it from changing at all, normally not a desirable setting. Higher numbers can give faster throttle response in many cases, but setting them too high can cause problems if you have noise on the input.

The following inputs can have lag factors applied to them:

- MAP Averaging Lag Factor (also applies to MAF)
- RPM Averaging Lag Factor
- TPS Averaging Lag Factor
- Lambda Averaging Lag Factor
- CLT/MAT/Batt Averaging Lag Factor (these sensors run off a common number)

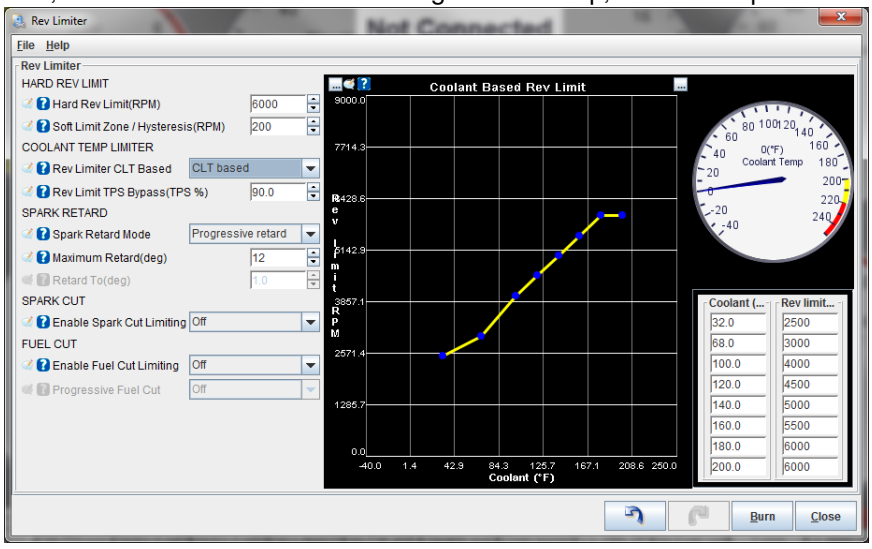
Auto-zero TPS - if TPS% is showing negative at power on, the current value will be taken as zero.

7.1.1.4 Load Parameters These critical settings determine the engine behavior and select your chosen load source. See section 5.2 for an explanation of the load types.

- **Primary Fuel load** - sets the primary fueling algorithm.
- **Secondary Fuel load** - enables an optional secondary blended fuel table. Most engines will not need a secondary fuel load.
- **Secondary Fuel** - "multiplicative" or "additive" - whether the values in VE table 1 and VE table 2 should be multiplied or added together when the secondary fuel table is in use. Selecting “Blend Curve” allows blending between two tables based on a curve (selected under the Table Choices menu).
- **Multiply MAP** - For Speed-Density installs this should normally be set to "Multiply". This number uses the air density, based on the MAP sensor reading, in the main fueling equation. If this is turned off on a speed density install, the VE table values at low loads can be very small, while any values under boost will head off to well over 100%. Turning this on will flatten out the table and make it a bit more manageable. About the only reason to turn this off when running speed density is on installs with a highly unstable MAP signal. Turning this on in alpha-N mode will result in a mode that is something of a hybrid between alpha-N and speed density, as the pulse width will be scaled for MAP readings.
- **Incorporate AFRTarget** - selects whether the target AFR value from the AFR table is used in the main fueling equation. See the tuning section before changing this setting.
- **Stoichiometric AFR** - this is used with the previous setting. 14.7 is typically used for gasoline. Many racing fuels may not follow this exact value; if you are changing from one race fuel to another and know the stoichiometric point for both, this can be used to change the fueling directly.
- **Primary Ignition load** - sets the load value for ignition table lookups.
- **Secondary Ignition load** - enables an optional secondary blended ignition table. The blended fuel and spark tables can be used to effectively enlarge the table size. This table can be additive or use a blend curve.
- **AFR table load** - sets the load value for AFR table lookups
- **EAE table load** - sets the load value for Enhanced Acceleration Enrichment curve lookups.

7.1.2 Rev Limiter

MS3Pro allows for fuel or spark based rev limiting. You can also allow the rev limit to change as a function of coolant, so as to set it low before the engine warms up, or even drop the rev limit if you overheat.



7.1.2.1 Hard rev limit

- Hard Rev Limit (RPM) - This is the point at which a fuel cut or complete spark cut is applied. The coolant temp limiter, if used, will override this setting.
- Soft Limit Zone / Hysteresis - The RPM needs to fall this far beyond the hard limit before the fuel cut is turned off. If you are using a spark cut or a progressive fuel cut, the cut will start to phase in when you reach this many RPM below the hard cut as well.

7.1.2.2 Coolant temp limiter The curve defines an alternate hard rev limit based on coolant temperature. You can use this to limit RPM with the engine cold or to create a shut-off if the engine is overheating, or both.

- **Rev limiter** coolant based -
  - Normal - The above settings are used independent of coolant.
  - CLT Based - the curve is used to select a rev limit based on coolant temperature, so that lower limits can be applied to a cold engine. When selected, the graph on the right hand side of the screen defines the rev limit.
- **Rev limit TPS bypass%** - above this TPS% the coolant curve is ignored, and it uses the standard rev limiter. This should be set to a high percentage number, so that only in an emergency when you really need full throttle on a cold engine can you override the cold limit.

7.1.2.3 Spark retard The MS3Pro has several different spark retard strategies. You specify the timing retard in an absolute angle which the MS3Pro will use as your timing; the timing specified here is not an amount of timing to pull but the total amount of timing to use. The spark retard begins at the soft rev limit - the hard limit minus the soft zone number.

- **Mode** - Select how
  - Off - no spark retard rev-limiting is used.
  - Progressive Retard - the timing is progressively retarded between the “retard above” threshold and hard rev limit. The retard starts just above the "retard above" rpm and reaches the maximum set retard by the "fuel cut" rpm.
  - Fixed angle - when the "retard above" rpm is reached the spark timing is immediately changed to the rev limiter timing.
- **Maximum retard** - the maximum amount of timing retard that will be applied
- **Retard to** - the fixed spark angle that will be used.

7.1.2.4 Spark cut Where MS3Pro is directly controlling spark, spark cut can be used as a rev limiter method. This cannot be used with EDIS and will not work as expected with TFI, GMDIS or C3I, as these systems have some level of independent spark control. Be aware that spark cut can be harsh and should not be used with catalytic converters due to the damage it causes. Frequently, there will be some level of "popping and banging" and exhaust system damage may occur. Use at your own risk!

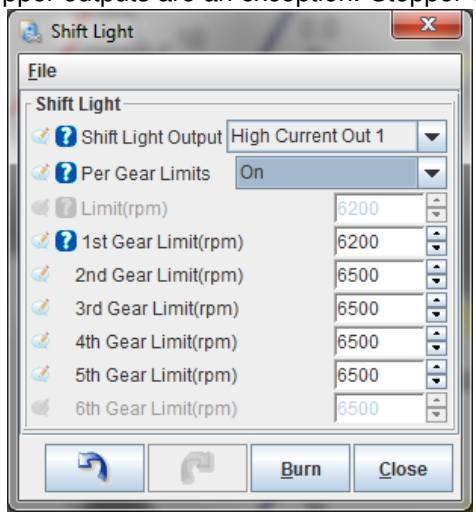
The spark cut algorithm cuts sparks on a "round-robin" basis to prevent individual cylinders loading up with unburned fuel. The cut begins at the soft limit value and the percent of sparks cut increases up to the hard limit, where the MS3Pro shuts off the spark entirely

7.1.2.5 Fuel cut This gives the option of an abrupt, total fuel cut, or cutting fuel to more and more cylinders until a total fuel cut at the hard limit.

- **Enable Fuel Cut Limiting -** Turns on fuel cut rev limiter.
- **Progressive Fuel Cut** - If this is turned off, the fuel cut applies abruptly when you reach the hard limit. If this setting is on, the MS3Pro will shut off fuel to one cylinder at the soft rev limit and ramp this up to cutting all cylinders at the hard limit. This does not rotate between cylinders like the spark cut - once the cylinder is off, it remains off until the RPM drops enough to turn it on. **Note that this cannot be used with throttle body injection or staged injection. This only works with one injector per cylinder.**

7.1.3 Shift light

While a programmable on/off output can be set to work as a shift light, the MS3Pro has a dedicated function for this that allows separate shift settings for each gear. The shift light will need switched 12 volt power on one end and the MS3Pro will supply ground. The light needs to be wired so it does not receive power with the key off. The stepper outputs are an exception. Stepper Out 1A and Stepper Out 2A supply 12 volt power when active.



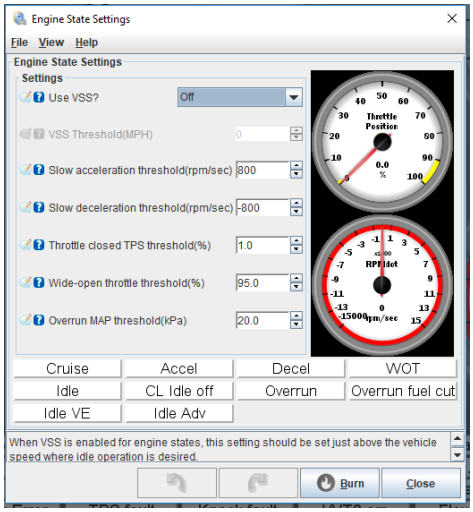
- Shift light output - Specifies which pin is used for shift light output.
- Per Gear Limits - Setting to On enables separate outputs per gear. Setting to Off uses a single limit. Transmission gear sensing needs to be set up in the Speed and Gear Sensors settings under Advanced Engine to use per gear limits.
- Limit - The point to turn on the shift light when not using per gear limits.
- 1st Gear Limit through 6th Gear Limit - Specifies when to turn on the shift light when using per gear limits.

7.1.4 Engine state settings

The Engine State Settings gives you a single point of tuning the criteria to activate several functions, to avoid the need to dial in nearly identical numbers on multiple screens. Currently, the following functions make use of engine states:

- Closed loop idle control
- Overrun fuel cut
- Idle timing control
- Idle VE table

If not using these advanced idle tuning settings, the engine states will have no effect on the tune, but will still show up in data logs. The indicators at the bottom of this menu toggle off and on as the engine states change.



Here are what the individual settings do and what functions they affect.

- **Use VSS** - Allows using the speed sensor to trigger closed loop idle and other idle related functions.
- **VSS threshold** - If using VSS is turned on, the MS3Pro will not use closed loop idle, idle advance, or idle VE above this speed.
- **Slow acceleration threshold** - This number should be set to the RPMdot range observed while slowly accelerating in high gear. It should be high enough to distinguish RPM jitter from intentional acceleration. If you see the slow acceleration status triggering at idle, this number needs to be raised.
- **Slow deceleration threshold** - To set this, cruise in a high gear, then see what RPMdot value you get by lifting off the throttle fully and decelerating. Set this value slightly above that RPMdot value. If you see the slow deceleration status triggering at idle, this number needs to be raised.
- **Throttle closed TPS threshold** - Defines the maximum throttle position for idle functions. Typical values range from 0.5% to 1.0% depending on how noisy your TPS signal is.
- **Wide open throttle threshold** - TPS value above which the engine is considered to be at wide-open throttle. Typical values range from 99.0% to 99.9%, depending on the amount of noise in the TPS signal.
- **Overrun MAP threshold** - Below this MAP value, the engine is considered to be decelerating in gear instead of idling.

7.1.4.1 How the MS3Pro uses engine states

The closed loop idle and other idle related settings use all of the settings:

- If using VSS, the idle functions are locked out above the specified vehicle speed.
- Closed loop idle will not engage if the RPMdot numbers exceed either the fast or slow acceleration / deceleration thresholds.
- The idle functions require the TPS to be below the throttle closed TPS threshold to function.
- If you select “Open valve on WOT” under closed loop idle settings, the MS3Pro will set the idle control valve to fully open if the TPS exceeds the wide open throttle threshold, and set the valve back to its closed position below this TPS reading.
- The closed loop idle will not operate when the MAP reading is below the overrun MAP threshold.

The overrun fuel cut uses several of the settings:

- The TPS reading needs to be below the throttle closed TPS threshold.
- The MAP sensor reading needs to be below the overrun MAP threshold.

- The RPMdot rate needs to be below the fast deceleration threshold (if the engine is slowing that fast, it is likely that the engine is in neutral and RPM is dropping rapidly, so the MS3Pro will turn on fueling to prevent the fuel cut from stalling the engine) and below the fast acceleration threshold (which clearly is not an overrun condition).

7.1.4.2 Tuning engine states Here are some general pointers for tuning the engine states.

Only use the VSS on vehicles capable of engine braking. The VSS should be turned off if your vehicle needs to idle while in motion. The most common example would be a car with an automatic transmission set up to freewheel on deceleration. If you have trouble with the engine stalling while the vehicle is moving, but it idles with no problem while stopped, turn this setting off.

If closed loop idle fails to engage properly, check to be sure the TPS reading does not go above the throttle closed TPS threshold and that neither the slow deceleration state nor the slow acceleration state are active while idling. If any of these states are false triggering, increase the threshold values. Also make sure the MAP sensor reading does not go below the overrun MAP threshold at idle.

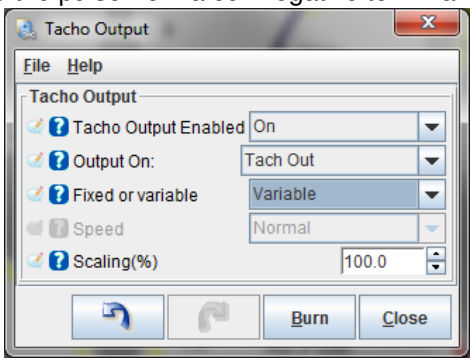
If the idle valve closes or opens abruptly when the engine is not idling, you may need to reduce the slow acceleration or slow deceleration threshold.

If overrun fuel cut fails to engage, check the throttle closed TPS threshold and MAP overrun MAP threshold to be sure both sensor readings are low enough.

If the engine stalls if you let off the throttle and disengage the clutch at high RPM, check if the overrun fuel cut engaged. If so, decrease the fast deceleration threshold to lock out the overrun fuel cut in these conditions.

7.1.5 Tacho Output

The tachometer feature provides a pulsed output suitable for connecting to a tachometer (rev counter.) The Tach Out pin will generate a 0-12V signal that will drive most aftermarket tachometers. The stepper output pins will also supply a 0-12V square wave. Some older tachometers may need a high voltage adapter if they are designed to use the pulse from a coil negative terminal or connect inline with the coil power feed wire.



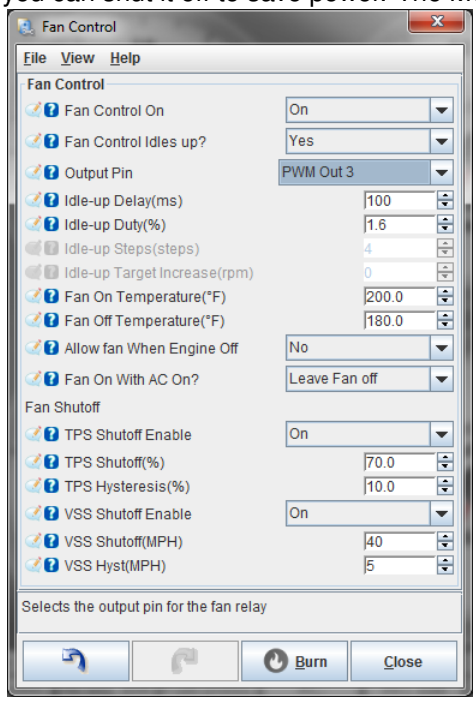
- **Tacho output enabled** - enables the feature
- **Output on** - selects the output pin
- **Fixed or Variable** - Fixed mode allows normal output or half speed. Variable tach output mode allows scaling the tach output by any percentage from 10% to 1000% of normal tach speed. Note that fixed mode may have a slight improvement in accuracy.
- **Speed** - This setting is used in fixed tach mode.
  - Normal - an output pulse is generated matching each cylinder event. i.e. 4 cylinder engines have 4 pulses per engine cycle.
  - Half speed - output runs at half speed. This can enable a 4-cyl tachometer to be used on an 8-cyl engine.
- **Scaling** - Scales the tach by a user selectable percentage instead of a fixed ratio. Numbers greater than 100 speed up the tach output (for example, if using a four cylinder engine with an eight cylinder tach, start with 200% scaling) while numbers less than 100 slow down the output (for example, if using an eight cylinder

engine and a four cylinder tach, start with 50% scaling). You can adjust this to correct for inaccuracies in your gauge.

Note that you can select other channels for tach output, but these pulse from floating to ground. To obtain a 0-12 volt signal from one of these channels, you can wire a 1K resistor from the output to a switched 12 volt source.

7.1.6 Fan control

While you can control a fan by simply stating it comes on at one temperature and off at another, MS3Pro gives you more options. You can adjust the idle speed to compensate for the increased electrical load, or factor in throttle opening or speed as well. Many cars have enough airflow at cruising speed to not need an electric fan at that point, so you can shut it off to save power. The MS3Pro outputs can drive a relay; don't wire them to the fan directly.

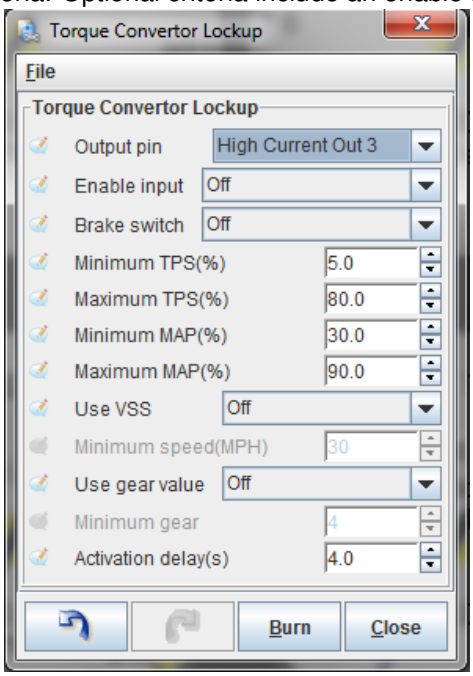


- **Fan Control On** - Enables the MS3Pro fan control
- **Fan Control Idles up** - Whether the fan activation should also raise the idle
- **Output Pin** - The output pin for the fan relay
- **Idleup delay** - how long to wait after raising idle before turning on fan
- **Idleup duty** - PWM open-loop duty cycle to increase by
- **Idleup steps** - additional stepper idle steps to open
- **Idleup target rpm increase** - target rpm to increase by (if using closed loop idle control)
- **Fan on temperature** - turn fan on above this temperature
- **Fan off temperature** - turn fan off below this temperature (must be lower than "on" temperature)
- **Allow fan when engine off** - allows fan to run in key-on, engine-off situation
- **Fan on with A/C on** - whether the fan should be automatically turned on to help cool the A/C radiator.
- **TPS shutoff enable** - turning this on will allow the MS3Pro to shut off the fan above a maximum throttle to reduce the alternator load on the engine.
- **TPS shutoff** - if TPS shutoff is enabled, the fan is disabled above this TPS%

- **TPS hysteresis** - the percent throttle opening that the TPS must drop below the TPS shutoff before the fan comes back on
- **VSS shutoff enable** - turning this on allows the MS3Pro to shut off the fan based on vehicle speed since at higher speeds the air flow through the radiator is often sufficient without the fan. Requires VSS input enabled under Speed and Gear Sensors.
- **VSS shutoff** - if VSS shutoff is enabled, the fan is disabled above this vehicle speed
- **VSS hyst** - the amount the speed must drop below the VSS shutoff number before the fan comes back on

7.1.7 Torque converter lockup

MS3Pro has the ability to control electric lockup on automatic transmissions. This is for part-electric transmissions (e.g. 700R4, A4LD) but not fully electric (e.g. 4L60E, 4L80E). As a minimum, TPS and MAP are used to set lockup criteria. Optional criteria include an enable switch, brake switch, speed sensor, or gear sensor.



- **Output port -** output port to connect to lockup solenoid
- **Enable input** - optional input port for enable switch. For example, some transmissions have a switch that turns on in fourth gear to enable lockup.
- **Brake switch** - optional input port for brake light switch. The torque converter unlocks if this input is active.
- **Minimum TPS** - minimum TPS to enable lockup
- **Maximum TPS** - maximum TPS to enable lockup
- **Minimum MAP** - minimum MAP to enable lockup
- **Maximum MAP** - maximum MAP to enable lockup
- **Use VSS** - optionally allow VSS1 or VSS2 speed input as criteria
- **Minimum speed** - sets minimum speed for lockup. (Requires VSS input in speed/gear sensors.)
- **Use gear value** - optionally allows gear number as criteria. (Requires gear to be enable in speed/gear sensors.)
- **Minimum gear** - minimum gear to enable lockup

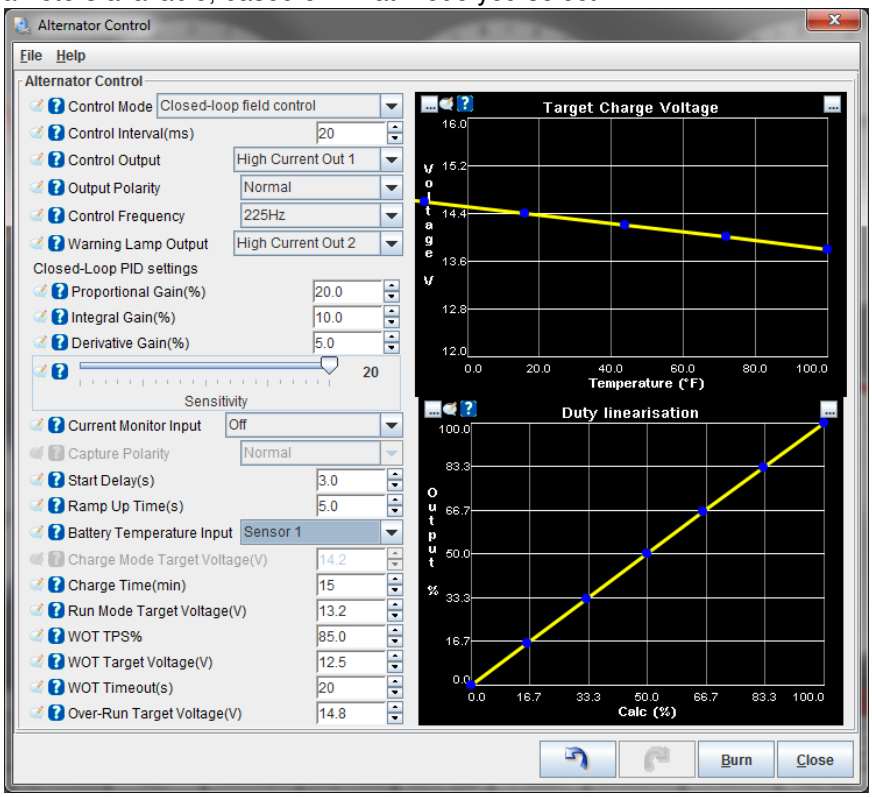
- **Activation delay** - wait this long after conditions are met before locking up.

The MS3Pro can drive a 5A solenoid with a high current output, or up to 3A with a PWM output. If you need more current, you will need to use a relay.

Note that these settings are not used if you have added on a MicroSquirt using the MicroSquirt transmission control firmware - the torque converter lock up will be tuned from the MicroSquirt's menu instead.

7.1.8 Alternator control

The MS3Pro can control alternator voltage if the alternator does not have an internal voltage regulator or if the voltage regulator supports allowing the ECU to adjust its voltage. Note that the menu will change some of the parameters available, based on what mode you select.



MS3Pro allows the following control modes.

- On-Off - Allows starting with the alternator off, then switching on after startup.
- Open Loop Frequency: Used for Ford alternators with internal voltage regulators that change target voltage in response to a frequency based signal.
- Open Loop Duty: Used for GM alternators with internal voltage regulators that change target voltage in response to the duty cycle of a signal from the ECU.
- Closed loop field control: Used for alternators that lack an internal voltage regulator, and the ECU controls field current. Uses a closed loop PID algorithm to target voltage.
- High speed feedback field control: A different method of closed loop alternator field control, similar to an electronic version of a points type regulator. The MS3Pro monitors the battery voltage at 20 kHz and switches the field off if it is above target voltage, and switches the field on if it is below target voltage. Often easier to set up than closed loop field control, as there are no PID parameters to tune - you just need to select this mode and set a target voltage.

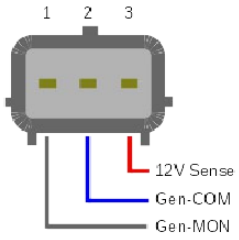
If the ECU is controlling the field coil directly, we recommend running the output through a solid state relay or external power transistor. Many alternator field coils may require 10 amps or more, which is more than the MS3Pro's recommended output current.

Here are what the other settings do.

- **Control interval -** How often, in milliseconds, the MS3Pro updates its calculations. Does not affect the speed at which the field switches in high speed feedback field control, but it does change other calculations such as target voltage.
- **Control output** - What pin is used for alternator control output.
- **Output polarity** - If set to normal, turning the output on sends the signal to increase voltage. If set to inverted, an on signal decreases voltage.
- **Control frequency** - What frequency is used to pulse the output in open loop duty or closed loop field control mode.
- **Warning lamp output** - If this output is enabled, the MS3Pro will trigger it if the voltage drops below the target voltage minus the **warn on voltage difference** setting.
- **Warn on voltage difference** - How far the battery voltage needs to drop below target voltage to trigger the warning lamp output, if used.
- **Load monitor input** - Allows using a PWM load signal from an alternator, such as the “F” terminal on many GM alternators. Not available in field control modes.
- **Capture polarity** - Determines if the PWM signal is based on sending voltage or being pulled low.
- **Warn on load -** Triggers the warning light output if the load signal goes above this value.
- **Closed loop PID settings** - When using closed loop field control, allows you to set the sensitivity of the closed loop response. See notes on closed loop idle tuning for how these settings work.
- **Current monitor input** - Allows using a PWM signal to monitor alternator output current. This number can be sent to a gauge or data logged. The MS3Pro does not adjust the alternator output based on this number - it just uses it for recording.
- **Capture polarity** - Determines if the PWM signal is based on sending voltage or being pulled low for the current monitor input.
- **Startup delay** - How many seconds to wait after engine start before turning on the alternator.
- **Ramp up time** - How many seconds to take the alternator from zero output to charging voltage after the startup delay is complete.
- **Battery temperature input** - Allows using a battery temperature sensor to adjust charging voltage based on battery temperature. This must first be defined as a generic sensor under Advanced Engine.
- **Charge mode target voltage** - If you are not using a battery temperature input, the MS3Pro will always target this voltage during the charge time. After the charge time is complete, it switches to run mode target voltage.
- **Charge time** - The time, in minutes, to apply the charging voltage.
- **Run mode target voltage** - The voltage to target after charge time is complete.
- **WOT TPS% -** Above this TPS reading, the MS3Pro can drop the target voltage to free up more power.
- **WOT target voltage** - The voltage to target above the WOT percentage.
- **WOT timeout** - After this many seconds at full throttle, target voltage returns to charge voltage (if during the charge time) or run voltage (if the charge time is complete).
- **Over-run target voltage** - The voltage to target when the overrun fuel cut is active. A higher voltage will create a minor engine braking effect.
- **Target charge voltage curve** - When you enable battery temperature input, this allows you to define the charging voltage target as a function of battery temperature.

- **Alternator control frequency** - When control mode is set to open loop frequency, this lets you specify what frequency commands what charging voltage.
- **Alternator control duty** - When control mode is set to open loop duty, this lets you specify what duty cycle commands what charging voltage.
- **Duty linearization** - When using closed loop field control, this curve translates what output percentage the ECU has calculated (calc% on the X axis) into output duty cycle (output% on the Y axis). This allows you to compensate for non-linear behavior or specify a minimum or maximum output duty cycle.

7.1.8.1 Ford alternators Ford added ECU controlled alternators starting in 1999. These use a three wire plug as shown below.



- SENSE (or A or S) - this connects to battery voltage, typically near or at the battery.
- GEN-COM (or GEN RC or SIG or RC) - this is the 'command' signal from the ECU to the alternator. Internally the alternator pulls this to battery voltage through a 1K resistor. Connect to
- GEN-MON (or GFS or FR or LI) - this is the 'monitor' signal from the alternator back to the ECU. The alternator switches this to ground; it connects to a 0-5V frequency input on the ECU with a pull-up resistor.

On the Ford Focus 98AB19399DF alternator tested, pin 3 (SENSE) was red, pin 2 (GEN-COM) was blue, and pin 1 (GEN-MON) was gray. Connect the GEN-COM to any unused high current or PWM output. Using the GEN-MON wire is optional. If used, connect it to an unused digital frequency or switched input, and use a 1K pull up wire between this input pin and VREF.

Ford uses a variable frequency system to set the alternator voltage. The 1.5.0 code allows setting a time period instead of frequency, which allows setting a wider voltage range. Setting a frequency sets the target for the internal voltage regulator. Turning off the signal or commanding a voltage of 12.0 V or lower will shut off alternator charging.

Voltage	Period
12.0 V	0.5 ms
12.8 V	3.4 ms
13.6 V	6.3 ms
14.4 V	9.3 ms
15.2 V	12.2 ms
16.0 V	5.1 ms

These alternators use the following settings:

- Control mode: Open loop frequency
- Control output: Use any available high current or PWM output.
- Output polarity: Normal
- Load monitor input: Can be set to off or an available digital switched or frequency input.
- Capture polarity: If using load monitor input, set this to Inverted.

7.1.8.2 Chrysler alternators Chrysler alternators have used an external field controller on their alternators since they started production of their own in-house alternators for the 1960 model year. Later Bosch and Nippondenso alternators used on their cars and trucks follow the same pattern of bringing out the field terminals for an external device instead of using internal regulation. Chrysler eventually integrated the voltage regulator control into the ECU, but did not change the alternator construction itself. The result is that you can control even 1960s era Chrysler alternators with the MS3Pro - or if you happen to be low on available pins, you can run a computer controlled Chrysler alternator with a 1970s era electronic voltage regulator.

Chrysler alternators have two different wiring types. The “single field” design first appeared in 1960 and ran through 1969, then resurfaced around 2007. The “dual field” design first appeared in small quantities in 1969, became standard in 1970, and started to be phased out after 2006. In spite of the name, both types have only one set of field **windings;** the difference is whether the alternator has one or two field **terminals**. The “single field” design brings out the 12 volt supply pin to the field windings, with the other end of the field winding grounded to the housing. The “dual field” alternators bring out both ends of the field wiring.

The field coil current draw can be 5 amps or more depending on the alternator. Consequently, we recommend using a solid state relay to drive the field coil. The solid state relay needs to be powered from switched 12 volts, and connects to the 12 volt field terminal. The other field terminal, if available, should be grounded to the alternator housing.

All Chrysler alternators use the same settings.

- Control mode: High speed field feedback control
- Control output: Use any available high current or PWM output.
- Output polarity: Normal

7.1.8.3 Miata alternators 1999 and later Mazda alternators used an ECU controlled field coil. However, instead of directly running the field coil, the ECU supplies 5 volts to a switching transistor inside the alternator to regulate the output voltage. You can control this alternator with the output from a high current or PWM output with a pull up resistor, or repurpose an unused spark output to provide a 0-5V signal without a pull up. Although they are marked as inputs, the digital frequency “inputs” can also supply a suitable 0-5V signal.

The alternator connector has two pins. The “P” pin is the 0-5V signal for field control; the “D” pin is a fault detection pin which the MS3Pro currently does not support.

Pull up method Settings:

- Control mode: High speed field feedback control
- Control output: Use any available high current or PWM output.
- Output polarity: Inverted

Spark output method

- Control mode: High speed field feedback control
- Control output: Use any available spark output or digital frequency input.
- Output polarity: Inverted for the spark outputs, normal for digital frequency input.

7.1.8.4 Chevrolet 4 wire alternators GM’s “CS” series alternators showed up in the late 1990s and were used through the mid 2000s. These used a four pin connector; not all applications used all the pins. The MS3Pro cannot fully control this alternator’s output voltage, but the alternator will support a delayed turn-on to reduce alternator load on startup.

- **P** - Phase output. Used for engine speed output on diesel applications. Not used with MS3Pro.
- **L** - Lamp. While this connection is used for an alternator warning lamp, it also doubles as a turn-on signal. When the alternator is supplied 12 volts through this pin - even if through a fairly high resistance - the alternator will switch on.

- **I/F** - Ignition or field. This pin can have two functions depending on the exact application the alternator is from - some alternators use this as an input and others use it as an output. The ignition terminal will trigger the alternator if supplied with 12 volts. On other alternators, however, this pin is called the field pin and is an output to indicate alternator load.
- **S** - Sense. Used in some applications to allow the alternator to monitor the voltage at a different point from its own output terminal. Not used with MS3Pro.

There are several ways you can run these alternators.

Simple installation This is the approach used on the MS3Pro drop on harness. Simply connect one side of a suitable alternator warning light to a switched 12 volt source, and the other side of the warning light to the “L” terminal. We recommend putting a resistor, at least a 1K, 1/4 watt unit, in parallel with the warning lamp so the alternator will still function if the warning light fails.

On/off control This allows the ECU to shut down the alternator during cranking and for a few seconds afterward. This can help with smoother startup, particularly if you are using one of these alternators on a small four cylinder engine or a wildly cammed engine that has difficulty idling. In this case, you would connect the “L” terminal to a 1K resistor, with the other end of the resistor wired to a switched 12 volt source. Then you also connect the “L” terminal to an unused high current or PWM output, allowing the MS3Pro to ground this pin and shut off the alternator.

Settings:

- Control mode: On/Off
- Control output: Use any available high current or PWM output.
- Output polarity: Inverted

On/off control with load monitoring Not all CS alternators support this feature. To find out which sort you have, unplug the four pin connector and measure the resistance between the “L” and “I/F” pins on the alternator side. If you measure approximately 400 ohms, the “I/F” pin is an ignition output and does not support load monitoring. If you measure 1000 ohms or more between these pins, the “I/F” pin is a field output pin and provides a pulsed output which the MS3Pro can use to monitor the alternator load.

The wiring is the same as for on/off control, except that you will wire the “I/F” pin directly to a digital frequency input pin.

Settings:

- Control mode: On/Off
- Control output: Use any available high current or PWM output.
- Output polarity: Inverted
- Load monitor: Use any digital frequency input pin.
- Capture polarity: Inverted

7.1.8.5 Chevrolet 2 wire alternators These alternators began to appear on General Motors products in the mid 2000s. These are sometimes called RVC, for remote voltage control. These alternators have true ECU control with a 128Hz 0-5V PWM signal to the "L" terminal. When the alternator is off or in fault conditions, the alternator pulls "L" to ground through a ~390R resistor.

The pins are as follows:

- **L** - Lamp. Used as an input to control alternator voltage.
- **F** - Field. This is a variable duty cycle output to indicate alternator load.

The signal uses a variable duty cycle to command a voltage to the internal regulator circuit.

Commanded voltage	Duty cycle
11.0 V	10%
11.8 V	26%
12.8 V	42%
13.7 V	58%
14.6 V	74%
15.5 V	90%

The easiest way to control this alternator is to wire a digital frequency input pin to the L terminal. If you wish to add load monitoring, connect the “F” terminal to another digital frequency input pin.

Settings:

- Control mode: Open loop duty
- Control output: any unused digital frequency input pin
- Output polarity: Normal
- Frequency: 128 Hz
- Load monitor input: any unused digital frequency input pin (optional)
- Capture polarity (if using load monitor input: Inverted

7.1.8.6 Honda alternators Most 1990s era and later Honda alternators used an on/off system that could switch between a charging voltage and a low output mode that produces around 12.5 to 12.7 volts to keep battery voltage from dropping. Grounding the “C” terminal on the alternator will put the alternator into low output mode. The MS3Pro supports this with on/off control, which allows the ECU to shut down the alternator during cranking and for a few seconds afterward. This can help with smoother startup. Simply wire the alternator “C” terminal to an unused high current or PWM output, allowing the MS3Pro to ground this pin and switch the alternator to low output mode.

Settings:

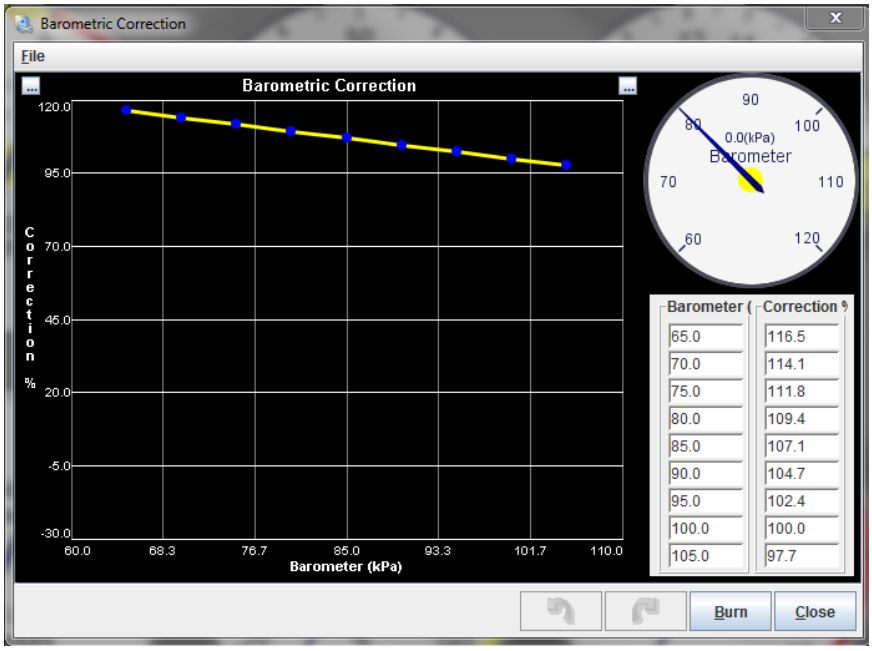
- Control mode: On/Off
- Control output: Use any available high current or PWM output.
- Output polarity: Inverted

7.1.8.7 Other alternators Many alternators do not have or need computer control. Internally regulated alternators, if not of the “one wire” type, typically only need an “exciter” signal (switched 12 volts) or a warning light signal to turn on. If the alternator uses an external voltage regulator connected to a field terminal, you can either use the regulator included with the alternator, or use the Chrysler directions.

If you encounter a computer controlled alternator not contacted here, consult a dealer service manual for the vehicle the alternator came from.

7.1.9 Barometric correction

This screen works similar to the warmup enrichment display, showing a percentage correction curve as a function of barometric pressure.



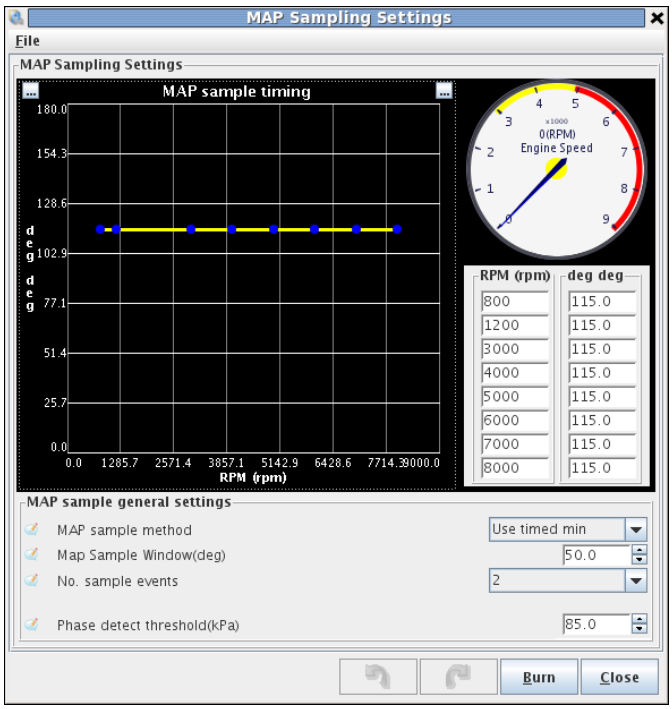
The number is a multiplier, so that 100% indicates to use normal fueling. The barometer reading is whatever the MS3Pro is currently using for correction. If you have enabled a barometric pressure sensor, the reading will change in real time. If you are using the initial MAP reading for barometric pressure, it will display the MAP reading it sampled when the MS3Pro initially powered up.

7.1.10 MAP sample settings

On an engine that is using the MAP sensor (most installs except pure MAF or pure Alpha-N) it is important to get a repeatable and stable MAP sensor signal. Note that repeatable is more important than "accurate". There are two schemes available:

- Event Average - the MAP signal is averaged between each ignition event. This is useful for most engines.
- Timed Minimum - the MAP signal is sampled in a defined sample window and the minimum reading is used. This is useful mainly on 1 or 2 cylinder engines.

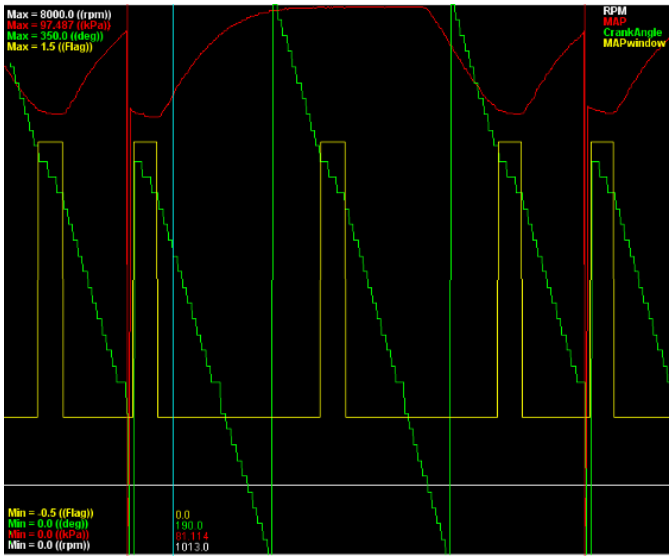
See also the MAP logger description.



- **MAP sample method** - whether to use the Event Average or Timed Minimum method. When Event Average is selected, no other settings on this page are required.
- **MAP Sample Window** - how wide the sampling window should be in degrees.
- **No. Sample Events** - how many windows to take the minimum across. For 1,2 cylinder 4-stroke engines this should typically be set to 2 - more details below. For 2-stokes or larger numbers of cylinders set to 1.
- **Phase detect threshold** - when using a MAP sensor in place of a Cam sensor (on 1cyl or V-twin), this is used to detect the intake stroke.
- **MAP sample timing** - this curve sets the start of the sample window in degrees BTDC for each ignition event. The number must always be less than the number of degrees per event. (720 divided by the number of cylinders: for a 4 stroke i.e. for 4cyl there are 180deg per event, for 8cyl there are 90deg.)

7.1.10.1 Event Average For the majority of engines, this scheme is proven to give the most consistent results. The multiple runners and overlapping intake events on a multi-cylinder engine lead to resonance and pulsations in MAP that vary across the RPM range and load. Taking an average over the cycle sidesteps these problems and gives a usable MAP reading for the fuel calculations and other table lookups. Once this mode is enabled, the other settings on the MAP sampling page are not needed or used.

7.1.10.2 Timed Minimum For some engines, particularly 1 and 2 cylinder four stroke engines, where there are fewer and more intermittent intake events it is necessary to specify where MAP sampling will occur in order to get a repeatable reading. Here is a MAP log from a 1 cylinder 4-stroke engine.



Observe the large variation in MAP signal during the 720 degree cycle. It can be seen that one MAP window falls on the intake stroke (where the MAP reaches a minimum) and the next MAP window is on the power stroke (MAP is near atmospheric.) On this engine it was important to set the appropriate MAP sample angle and window width to capture the dip in MAP, and set the number of events to 2, since the intake happens every two revolutions.

The ideal MAP sample angle may either be determined by:

- using the MAP logger (registered TunerStudio only) and opening the datalogs created in MegaLogViewer to ensure the MAP window aligns with the dip in MAP
- by watching the MAP gauge on the TunerStudio dash and adjusting the MAP sample angle in (say) 10 degree steps until the minimum is found.

Note that on most engines the best sample angle will vary with RPM. The Window Width should be set so that it covers the dip repeatably.

The sample data below shows a practical example from the above test engine with a short sample of the data.

First - with No. Events set to 1.

Event no.	Min MAP in window	MAP used for fueling
1	70	70
1	98	98
1	70.5	70.5
1	98	98

The MS3Pro is sampling the MAP reading on both the intake stroke and the power stroke; the intake strokes have a low MAP reading, but the pressure on the power stroke is nearly atmospheric. This results in the MAP bouncing around between 70 and 98 resulting in unpredictable tuning. Using Event Average would be equally poor.

Now - with No. Events set to 2.

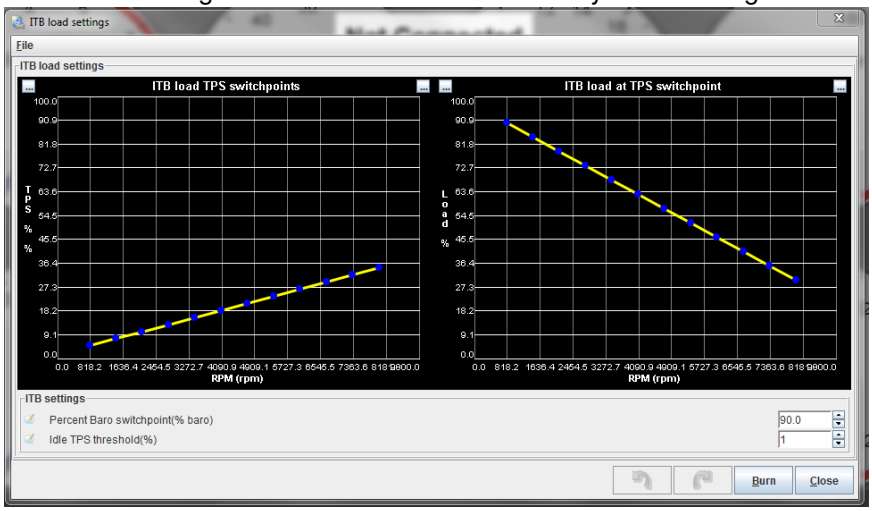
Event no.	Min MAP in window	MAP used for fueling
1	70	70
2	98	70
1	70.5	70.5
2	98	70.5

This gives a more realistic and more repeatable MAP signal and is a good start for tuning - the atmospheric readings of 98kPa on the "dead" windows are ignored.

The Phase detect threshold is in use on this particular engine. It allows 1cyl and certain 2cyl engines to run sequential fuel and spark without a cam sensor. The MS3Pro "looks" at the MAP signal during the reading and compares it to the threshold to determine whether this is an intake stroke.

7.1.11 ITB load settings

ITB tuning mode builds on the capabilities of the dual table blended tuning approach but solves one of the more significant drawbacks to that tuning mode: it provides the blended Speed-Density/Alpha-N behavior of the blended dual tables in just a single table. This single table approach is a significant improvement as all the automatic tuning tools available through TunerStudio now work correctly with the single table.



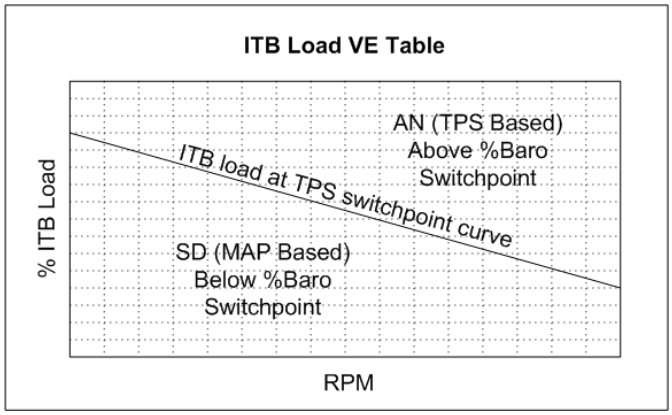
The ITB mode uses a hybrid of speed density and alpha-N for load. This load type is selected in the tuning configuration the same way that Alpha-N, Speed-Density, or blended tuning are selected. ITB load is also available for the other tuning settings including AFR, Ignition, and Enhanced Accel Enrichment.

The ITB Load is derived from a combination of MAP and TPS values as well as other ITB-related tuning curves that all work together to create a calculated value that is used as the "ITB Load" and applied to the Y axis of the tuning tables in the same way that MAP is used for Speed-Density tuning or TPS is used for Alpha-N tuning.

7.1.11.1 ITB load VE table ITB Load tuning uses a single VE table to control fuel but this VE table is partitioned into two regions, one for Speed-Density and one for Alpha-N tuning. There are two tuning curves and a couple of configuration values that must be set up to correctly partition the VE table for a given engine. The curves and values are:

- %Baro Switchpoint - The MAP value where tuning switches from TPS based to MAP based. Default is 90% and works for most engines.
- Idle TPS Threshold - A minimum TPS value that must be met to allow the switch to TPS based tuning to occur. This is sometimes useful in idle tuning.
- ITB Load TPS Switchpoint - A TPS curve that defines the TPS value across the RPM range where the MAP value reaches the pre-defined %b baro switch point.
- ITB Load at TPS Switchpoint - A curve that allocates the amount of each VE table column that will be allocated to MAP based or TPS based tuning.

The diagram below illustrates how the ITB load VE table is partitioned for dual use. The region of greatest MAP change (below %Baro switchpoint) is tuned using Speed-Density based tuning. The region of greatest TPS change (above %Baro switchpoint) will be tuned using Alpha-N based tuning. This is exactly the same technique used in the blended tuning, just applied within a single table.



7.1.11.2 ITB load TPS switch point curve This curve defines the TPS value where the MAP load reaches %Baro switchpoint. This curve will be different for each engine and should be set up using values obtained from log files from your engine. The curve tends to be fairly linear so you only need a few data points to plot the curve. A data point at low, medium, and high RPM from a log file is usually enough. A spreadsheet or just graph paper can then be used to establish enough data points to fill in the table for this curve.

ITB Load tuning requires that the MAP signal be above the %Baro switchpoint and that the TPS value be above the value defined on this curve to switch from Speed-Density tuning to Alpha-N tuning. Therefore, you want this curve to be relatively accurate and you may even want to set the values on the curve a few percent low to ensure that the TPS value has been met when the MAP reaches the %Baro switchpoint.

Besides defining the switch point to Alpha-N tuning, this curve also establishes the lower TPS value that will be used to interpret the range of VE bins allocated to Alpha-N tuning in the VE table.

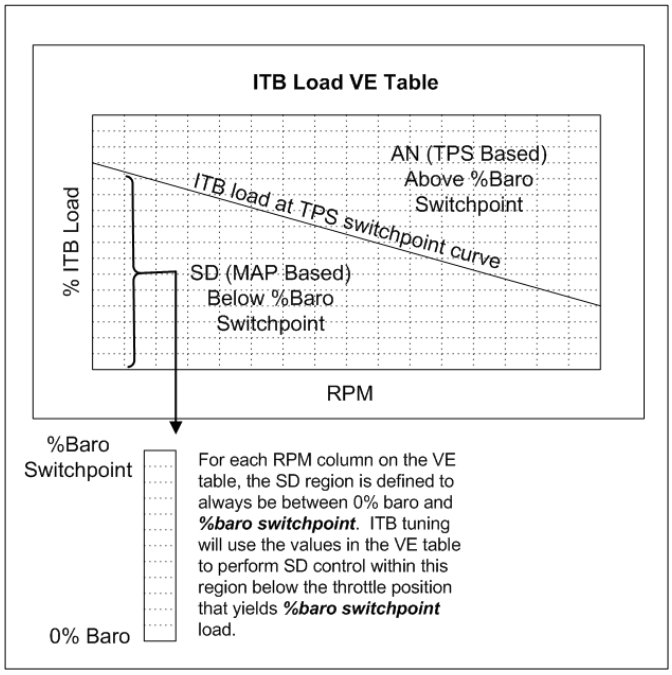
7.1.11.3 ITB Load at TPS Switchpoint Curve This curve is used to allocate the bins on the VE table to either Speed-Density or Alpha-N tuning. The area of the VE table below the curve will be used for Speed-Density tuning and the area above the curve will be used for Alpha-N tuning. The shape of this curve defines how much of the VE table will be allocated for use between Speed-Density and Alpha-N tuning for each RPM column. You want to allocate the largest portion of the VE table at each RPM to the tuning mode that has the most non-linear response. The lower RPM region typically requires a little more Speed-Density definition range than the upper RPMs.

7.1.11.4 Putting it all Together - Calculating ITB Load The ITB Load VE table is defined as ITB Load vs. RPM. The ITB Load should not be confused with MAP or TPS; it is neither. What the ITB Load tuning algorithm does is calculate a load value based on MAP, TPS, and the two ITB Load curves. This calculated load value is the Y axis of the ITB Load VE table; it can also be applied to the ignition advance and AFR tables as well.

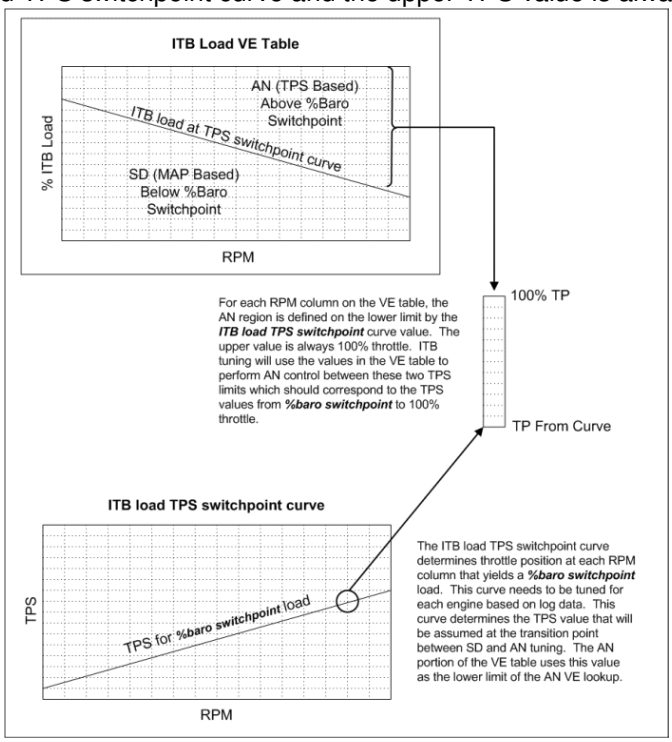
7.1.11.5 ITB Load Calculation in Speed-Density Mode When the throttle position is less than the value defined in the ITB load TPS switchpoint curve or the MAP value is less than the %Baro switchpoint, the tuning algorithm will take the array of cells from the VE table below the ITB load at TPS switchpoint curve and interpret this array within the context of 0% to %Baro switchpoint load.

A couple examples that assume %Baro switchpoint = 90:

If you have allocated the region between 0% ITB Load and 60% ITB Load on your VE table for use in Speed-Density tuning and your %baro is 50% then the VE value for 30% ITB load will be used. In this same example, a MAP value of 0kpa would use the 0% ITB load bin and a MAP value of just less than %Baro switchpoint would use the VE value just below the 60% ITB Load value on the VE table.



7.1.11.6 ITB Load Calculation in Alpha-N Mode When the throttle position is greater than or equal to the ITB load TPS switchpoint curve and the MAP value is greater than or equal to the %Baro switchpoint, the tuning algorithm will take the array of cells from the VE table above the ITB load at TPS switchpoint curve and interpret this array within the context of TPS position. The lower TPS value used for this interpretation is taken from the ITB load TPS switchpoint curve and the upper TPS value is always 100%.



For example, if you have allocated the region between 60% ITB Load and 100% ITB Load on your VE table for use in Alpha-N tuning and also assigned a value of 10% TPS on your ITB load TPS switchpoint curve then a TPS value of 55% would yield an ITB Load value of 80% and the VE bin for 80% ITB Load would be used. In this same example, a TPS value of 10% would use the 60% ITB Load bin on the VE table and 100% TPS would use the 100% ITB Load bin.

7.1.11.7 Tuning For Idle Air Control The Idle TPS Threshold % setting defines the minimum TPS value required to allow the transition to Alpha-N tuning. This feature allows the fast idle MAP to exceed the %Baro switchpoint without entering Alpha-N mode. On engines with aggressive cams with higher overlap, it is possible for the fast idle MAP to exceed the %Baro switchpoint. When this happens, you do not want to enter Alpha-N mode tuning since the throttle is still reading fully closed. The Idle TPS Threshold % prevents you entering Alpha-N mode and you can use the warm up enrichment curve to compensate for any tuning errors caused by being stuck at the maximum Speed-Density bin while the MAP is greater than the %Baro switchpoint.

7.1.12 MAF options

Mass air flow (MAF) sensors provide a measurement of actual airflow through the intake system of an engine. In its most basic form, the fuel required by the engine (in mass per second) is directly related to the air flowing into the engine (in mass per second), so the MAF sensor should give a good basis for the fuel equation. (By comparison, Speed Density uses RPM, MAP, MAT and VE to estimate the mass air flow.) In practice, like any other sensor input, the MAF has its own quirks and can suffer from reversion effects (particularly on "cammy" engines) where the air flow gets measured twice.

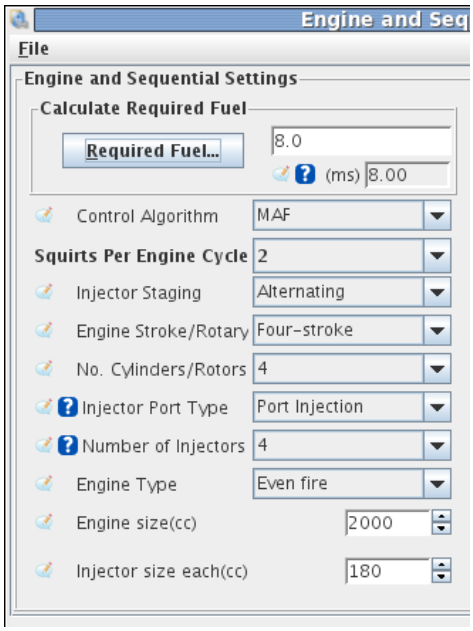
The MS3Pro implementation of MAF consists of the following main elements:

- Engine and injector parameters
- MAF calibration curve
- Optional trim table

Before setting up the MAF options, be sure you have the right parameters set under Engine and Sequential Settings. The critical settings to enter here are:

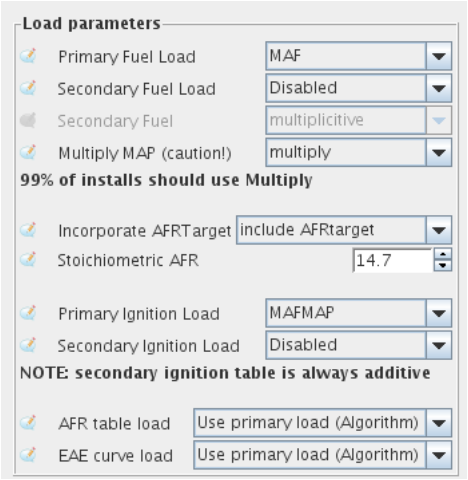
- Engine Stroke
- No. cylinders/rotors
- Number of Injectors
- Engine size
- Injector size each.

Having entered those details, click on the Required Fuel button to calculate the global fuel constant. You may need to re-enter the numbers if the calculator doesn't pick them up automatically. Make sure you set the correct units.



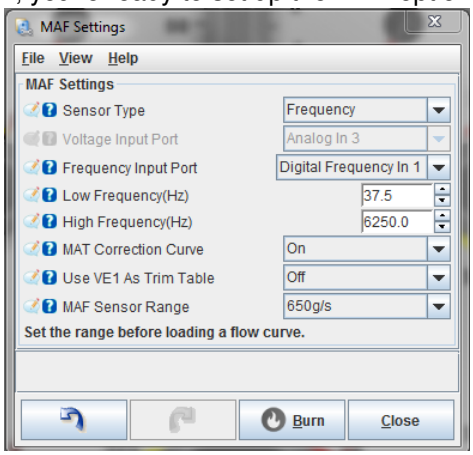
The MS3Pro will need all these numbers to be entered correctly, as it calculates the required pulse width to hit target air/fuel ratios, instead of using something along the lines of “send X pulse width at Y RPM and Z air flow meter volts.”

You will also need to set these settings in the General Settings page.



- **Primary fuel load** is set to MAF
- **Incorporate AFR target** is set to include AFR target
- **Stoichiometric AFR** is set to 14.7 for petrol/gasoline.
- **Ignition load** is set to MAF (If MAF is also being used for ignition control)

Now, you’re ready to set up the MAF options screen.



Note that you can configure the MAF sensor and datalog MAF without using MAF for fuel or ignition control. This could be useful to ensure it is operating as expected on an existing install before switching to MAF. Your Primary Fuel load would be set to something other than MAF.

- **Sensor type** - selects between 0-5V voltage input (e.g. most MAFs) or frequency input (e.g. GM)
- **Voltage input port** - selects the hardware input pin where the MAF is connected. (See wiring section above.) Or "off" to disable MAF readings.
- **Frequency input port** selects from four possible frequency input pins. (See wiring section above.)
  - Low frequency - the frequency the MAF outputs to indicate least air flow.
  - High frequency - the frequency the MAF outputs to indicate greatest air flow.
- **MAT correction curve** - optionally enables a trim curve based on MAT sensor input.
- **Use VE1 as trim table** - optionally enables VE table 1 as a trim table. Fill the table with 100% to begin with.

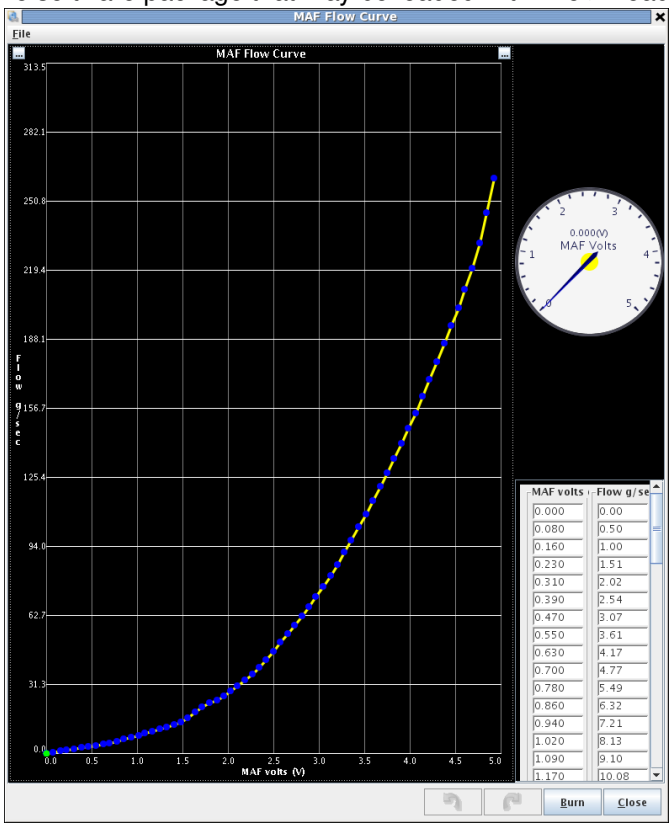
- **MAF sensor range** - sets the range to match your MAF sensor. Set this before loading a MAF calibration curve. On gasoline, 650g/s ~800hp; 1300g/s ~1600hp; 1950g/s ~2400hp and 2600g/s ~3200hp. Use the lowest range suitable for better low-end precision.
- **MAT correction table** - this curve allows adjustment of flow values from the loaded calibration curve.
- **Use old-style calibration curve** - select this if you want to use MAF in the “old way” as per 1.1.x and earlier. In this case, the calibration curve is not the actual flow curve, but a trim curve applied to the calibration curve loaded in the Tools menu. See section 7.1.11 for details.

7.1.12.1 More about frequency based MAFs Some GM MAF sensors give a frequency output rather than an analogue voltage. This can avoid ground offset issues. Due to the way that the frequency is measured, the reading becomes more coarse at higher frequencies. At 10kHz the measurement has 1% accuracy, 15kHz is 1.5%. For better repeatability, it is suggested to get a larger MAF and re-calibrate in preference to running above 10kHz.

Be sure to set the minimum and maximum frequencies before altering the flow curve. Pre-defined calibration curves are available for GM LT1, LS1, LSx MAFs. When using the 650g/s file (~800hp) set the min/max frequencies to 1000Hz and 11500Hz. When using the 1300g/s file (~1600hp) set the min/max frequencies to 1000Hz and 14125Hz. For larger flowing MAFs, a custom calibration will be required.

7.1.13 MAF flow curve

This will bring up the MAF calibration curve, showing airflow as grams per second as a function of voltage or frequency. Ideally, you should start with the curve that matches your actual sensor. Some examples are provided in the software package that may be loaded with File > Load Dialog Settings.



(This one curve replaces the Calibration and Correction Curve that were used in previous firmware releases.)

7.1.13.1 MAF Load As shown above, the primary fuel load is MAF because fuel required is directly related to mass air flow. However, ignition, AFR, and other load based tables are not directly related in the same way. While it would be possible to use MAF volts as the Y-axis on other tables, it would mean that only a very narrow band of the table was used, and tuning would suffer. So instead of that, the code calculates a synthesized load value called MAF Load. This behaves similarly to MAP in a speed density system and should give a repeatable load value and

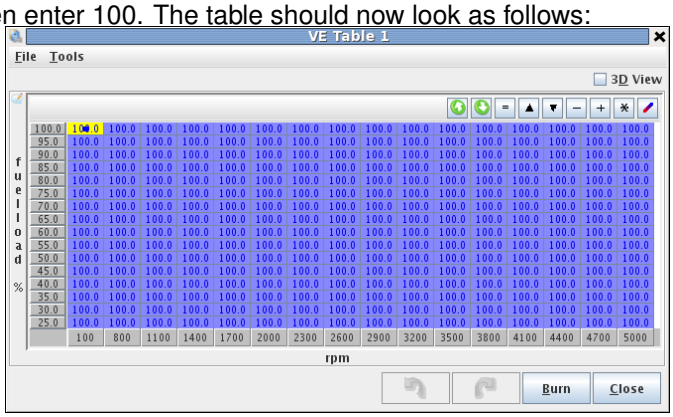
allow tuning of those tables in the conventional manner. Make sure the axes on the table are set to cover all MAF load values and RPMs if the engine is turbocharged; as a rule of thumb, the highest MAF load will probably be higher than the highest K Pa value the engine sees. When live tuning, the active cell will be highlighted.

A MAF only install will need to use MAF Load for spark and AFR table lookups. But, if you have a MAP sensor installed you can choose to set "Speed Density" for spark and/or AFR which will then use MAP as the Y-axis in the table. It is also possible to use alpha-N and tune the spark map based on TPS.

7.1.13.2 Tuning Initially it is suggested to leave the trim table turned off. Ensure you have configured the basic engine parameters: number of cylinders, number of injectors, engine size, injector flow and used the ReqFuel calculator to set ReqFuel. As noted above, set primary fuel load to MAF and spark and AFR to use MAF. Configure the AFR table for your chosen target AFR values. Ensure that the load and rpm axes cover your expected range; alter them to suit if needed. It is suggested that EGO correction be set to "No correction" at this time. Having started your engine, your primary tuning tool at this point is to make tweaks to the MAF Flow curve so that the reported AFR from your wideband matches the target table you configured. (You can enable the AFR error gauge in TunerStudio to see the difference between the two.)

Ignition tuning will be about the same, but do not try dialing in the final ignition table until you have the final MAF calibration curve.

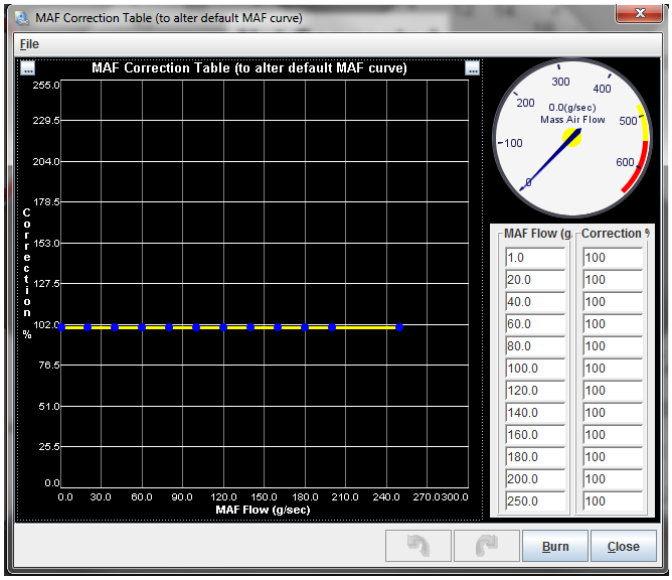
7.1.13.3 Using VE1 as a trim table If and only if you have exhausted tuning using the Flow curve alone, you can enable VE1 as a trim table. This gives you a full trim table of MAF Load vs RPM. (VE3 is also available with table switching.) When first enabling VE1 as a trim table you will need to fill it with 100%. Press CTRL-A, = and then enter 100. The table should now look as follows:



Adjust the table as needed to hit your AFR targets.

7.1.14 MAF correction table (old)

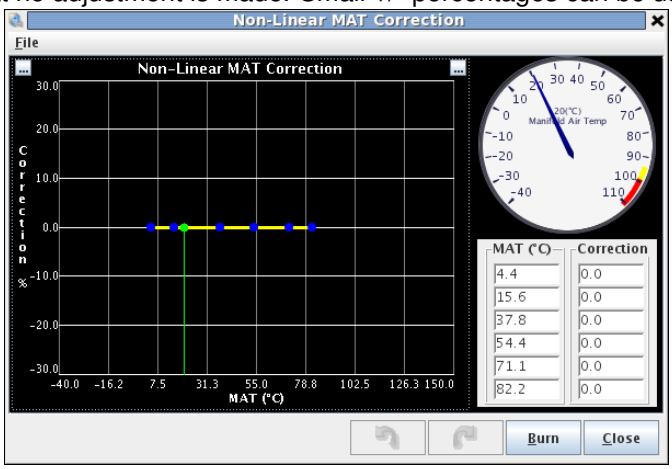
When using the older MAF implementation, you will enter the MAF calibration curve under Tools -> Calibrate MAF Table. This curve is a percentage trim to correct the curve loaded from the Tools menu.



The MAF flow number is the number on the original calibration curve, and you apply a % correction to adjust the curve.

7.1.15 MAF / MAT correction table

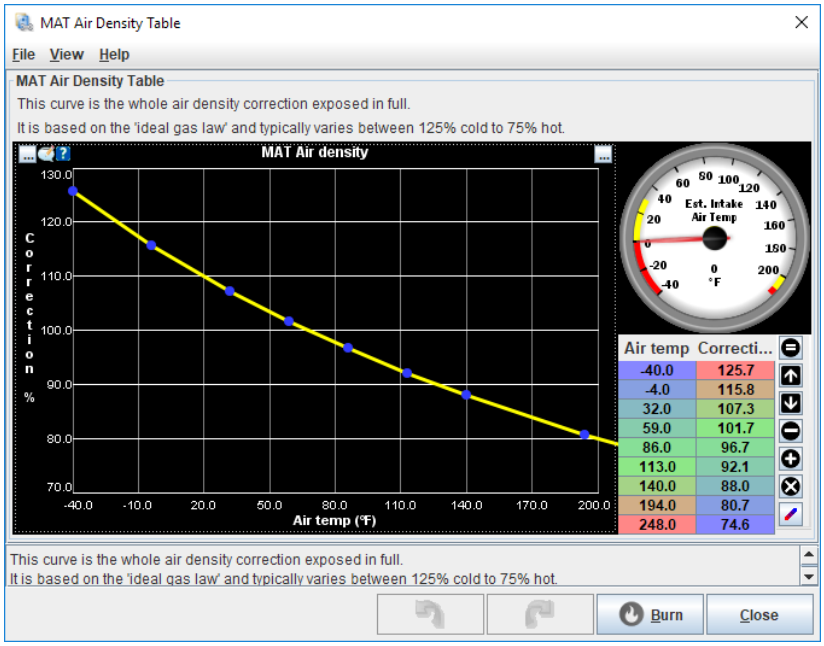
The MAF setup should automatically compensate for ambient air temperature as the mass flow is measured. If it is found necessary to make small adjustments for air temperature, then an optional correction curve may be enabled. Enabling this allows access to the MAT correction table on the Advanced menu. The default values of 0% mean that no adjustment is made. Small +/- percentages can be used to tweak the fueling at these air temperatures.



Use this table if you are using MAF based fueling and you find your car runs differently under different weather or temperature conditions.

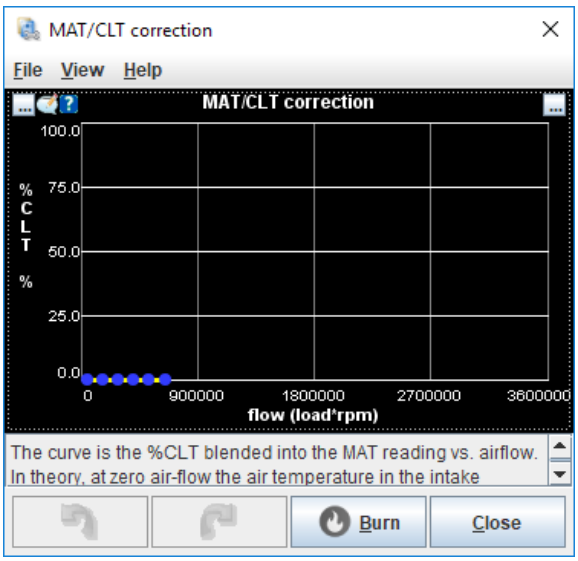
7.1.16 MAT Air Density Table

This table applies a percent correction to the fueling based on air temperature.



The default curve is based on how the air density changes based on the ideal gas law. However, if your find the engine is running lean at high indicated air temperatures, you may need to flatten out the curve around 100 degrees. You can even use this to add fuel at high air temperatures to reduce the risk of knock.

7.1.17 MAT / CLT Correction



This curve allows for correcting for additional heat transferred to the incoming air downstream of the MAT sensor.

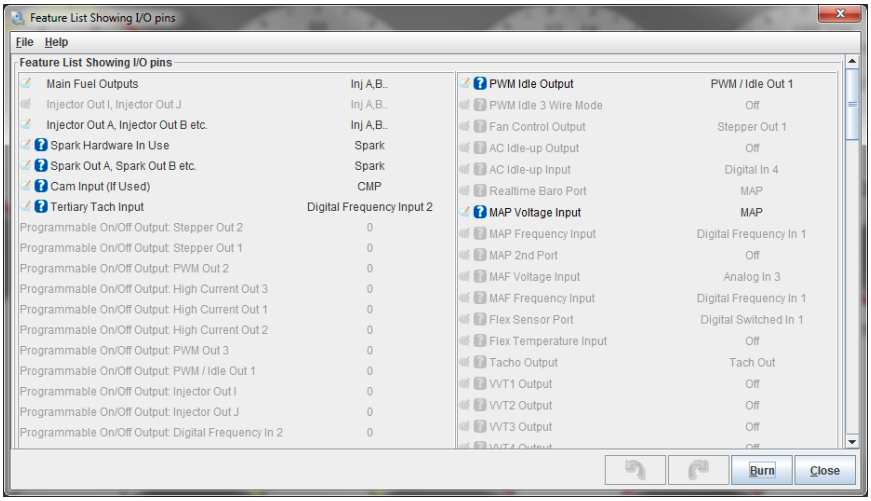
The numbers indicate the percentage of coolant temperature blended into the MAT reading vs. airflow. In theory, at zero airflow, the air temperature in the intake manifold will reach the manifold (coolant) temperature before it arrives at the intake ports. At infinite airflow, the air will be unaffected by the temperature of the manifold. The curve is setup to give the 'best estimate' of intake air temperature as part of the speed-density fuel calculation. When first enabling this curve it will be necessary to adjust the VE values at low RPMs, as they will be compensating for the air density change, and possibly the warmup enrichment as well. Use this table if you have problems with the engine running leaner at high RPM during warmup when it is idling correctly.

Note that this curve can only add to the air temperature - it is not able to subtract temperature to compensate for heat being conducted from the engine to the sensor. Use the MAT Air Density Table to add fuel at higher coolant temperatures instead.

7.1.18 Realtime Display

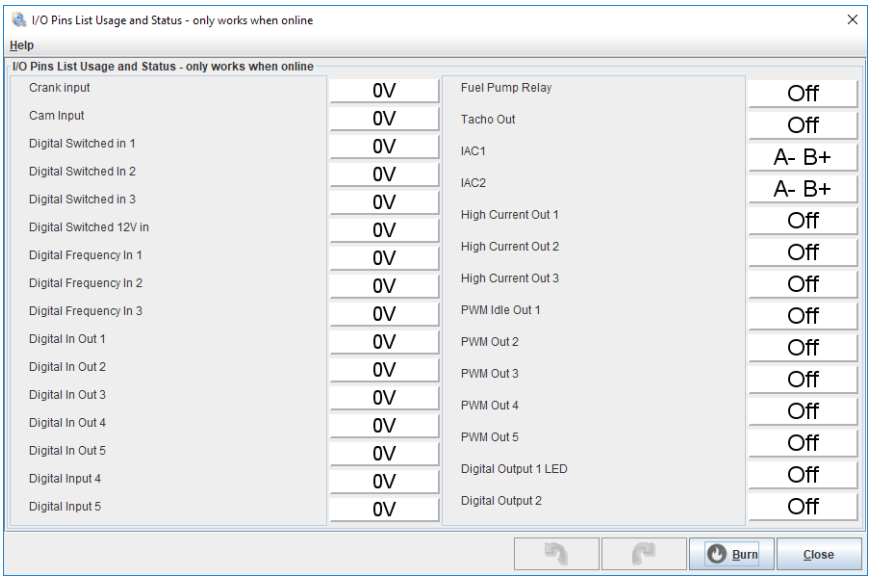
This tab allows you to monitor real time data from the ECU. The “Summary” tab gives line graphs over time of the more commonly used channels, while All Output Channels gives a numerical readout of more channels.

7.1.19 Feature List Showing I/O Pins



This allows you to display all the pins in use. You can use this to check what wire does what function, or check if you have accidentally assigned one pin to multiple functions. Features not in use are grayed out.

7.1.20 I/O Pins Showing Usage



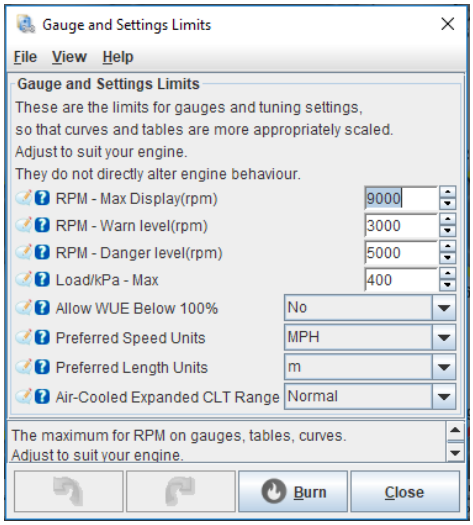
When TunerStudio is connected to the ECU, these two pages show what the state is on inputs and outputs. Note that the digital outputs can register 0 or 5 volts, but will not report actual voltage measurements in between.

7.1.21 CAN / Loop Usage and Status

This screen is similar to the I/O Pins Showing Usage screen, but does the same for devices connected to the MS3Pro via the CAN connection, such as a CAN-EGT or a MicroSquirt running IO Box or transmission control code, and for loop conditions configured under Programmable On / Off Outputs 2.

7.1.22 Gauge and setting limits

Allows you to specify RPM levels and maximum load values on the gauges, as well as the highest values you can enter into the load and VE tables. The other settings are explained below.



- **Allow WUE Below 100%** - The warmup enrichment is a multiplier, with 100% being no enrichment. Values under 100% will reduce fueling to less than the base fueling. This is ordinarily an undesirable thing to do, so the MS3Pro will lock you out of this. However, LPG and gaseous fuels often use an evaporator or expansion device heated by engine coolant. When the engine is cold, the evaporator delivers colder, denser fuel than normal, which may require reducing warmup enrichment below 100%. Some tuners also like to configure a WUE curve that has values slightly below 100% when the engine is above operating temperature but not into an overheat range.

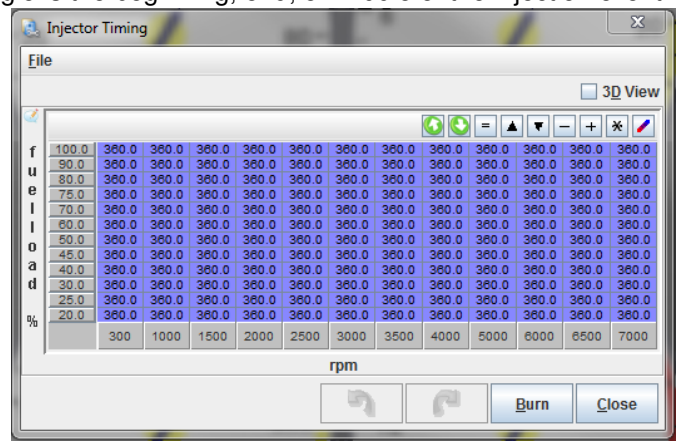
- **Preferred Speed Units -** This is used for speedometer gauges, boost by speed, and other speed related functions.
- **Preferred Length Units -** This is used for entering tire size and similar parameters. The MS3Pro allows mixing metric and English units.
- **Air Cooled Expanded CLT Range** - Adjusts the default range of the coolant temperature gauge. Liquid cooled engines seldom have a coolant temperature very far beyond the boiling point of water, but if you're measuring the cylinder head temperature with your CLT sensor, the temperatures can, and do, go much higher. When set to normal, maximum coolant temperature on the dash gauge is 250 degrees F. When set to expanded, maximum coolant temperature is set to 450 degrees F. Note that you will need to reload the dash for this to take effect.

7.2 Fuel Settings

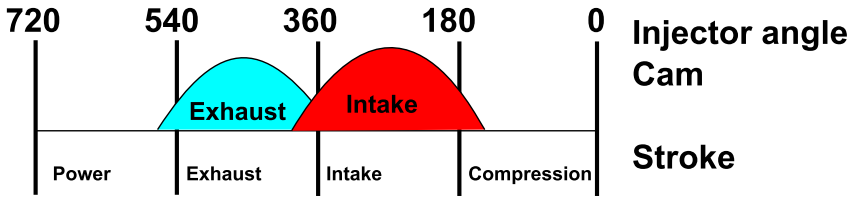
This contains all settings outside of the Basic / Load Settings related to fuel, other than nitrous control. Injector dead time, the injector small pulse widths, AFR table, and the VE table are discussed earlier.

7.2.1 Injector timing table

This is a 3D tuning table where you specify the injection angle for the cylinder receiving the injection. Like ignition timing, this is in degrees BTDC. The **Engine and Sequential Settings** menu allows you to define whether this angle is the beginning, end, or middle of the injection event.



Injector timing is specified as a function of RPM and fuel load. Most engines run best if the injection timing is completed before the valve opens, to allow the fuel to vaporize. So, if you have a cam card with an "Intake valve opens" spec, you can use this and the "end of squirt" setting to ensure the fuel injection finishes before the valve opens. An injector timing value of 360 degrees, when used with timing set to "end of squirt", would finish the injection event just at the intake valve is starting to open, as shown in the diagram below.



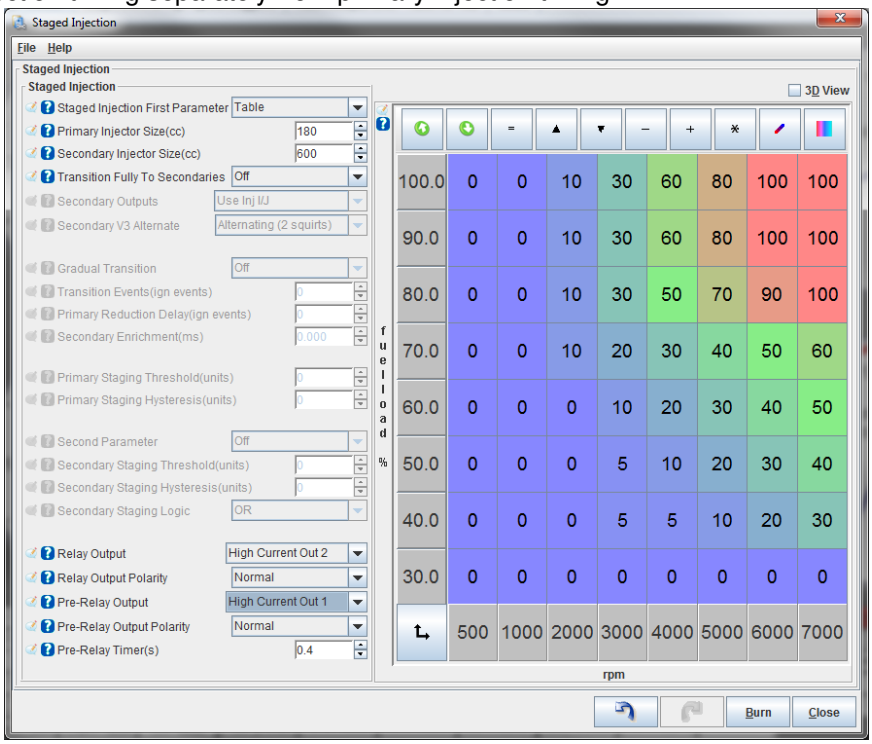
7.2.2 Secondary injector timing table

This timing table is used if you have enabled a secondary set of injectors, either with the Staged Injection or Dual Fuel menu. While these injectors may need the same timing as the primaries, secondary injectors that fire in an upstream direction or in an air horn often need different timing.

7.2.3 Staged injection

Staged injection allows the use of one set of small injectors for low-load and/or low RPM operation of the engine with the ability to engage a second set of injectors at higher load and RPM when the primary set of injectors would otherwise reach their maximum operational duty cycle. The MS3Pro's staged injection function supports staging equally between the primary injectors and secondary injectors during staged operation as well as staging completely to the secondary set of injectors. You may either use the standard injector outputs A through H on a four cylinder, or use injectors I and J in batch fire on engines with more than four cylinders. Sequential injection is not available for engines with more than eight cylinders.

If you have enough outputs to use timed injection on the secondary injection, the Secondary Injection Timing Table is enabled (in addition to the standard injection timing table), so the user must remember to tune secondary injection timing separately from primary injection timing.



- **Staged Injection First Parameter** - This setting enables staged injection and sets the primary staging parameter, which is used to determine when to engage the secondary injectors. This setting includes the following options:
  - Off - Staging disabled.
  - RPM Stage at a specific RPM.
  - MAP Stage at a specific MAP.
  - TPS Stage at a specific throttle position.
  - Duty Stage at a specific primary injector duty cycle.
  - Table Stage based on the table included in the dialog.

For table-based staging, 0% means not staged at all, and 100% means fully staged. Anywhere between 0% and 100% means partially staged. The firmware calculates the primary and secondary pulse-widths based on the injector sizes specified elsewhere in the dialog.

- **Primary Injector size** (cc) - Size of primary injectors in cubic centimeters.
- **Secondary Injector size** (cc) - Size of secondary injectors in cubic centimeters.
- **Transition fully to secondaries** - This setting is used to determine whether the fully staged state means that the primary and secondary pulse-widths are equal, or if the fully staged state means that primaries are shut down completely and staging transitions fully to the secondary injectors. The following settings are possible:

  - Off - The fully staged state means that the primary and secondary injector pulse-widths are equal.
  - On - The fully staged state means that the primaries are shut down, and staging transitions fully to the secondary injectors.

- **Secondary Outputs** - This setting controls whether the regular sequential outputs are used for secondaries or the I/J outputs on the gray connector are used. You can use sequential injectors on up to four cylinders. The following options are possible:

  - Use Inj A -H if possible - Use the standard outputs A through H. The second stage is fully sequential with this setting. Can be used for up to four cylinders.
  - Use Inj I/J - Use the I and J injector channels in batch fire. This is your only option for engines with four to eight cylinders.

- **Secondary V3 Alternate** - This setting controls whether the secondaries (when using the I/J outputs) alternate between injector channel I and injector channel J, or inject simultaneously on injector channel I and injector channel J. The following settings are available:

  - Alternating (2 squirts) - Alternate between injector channel I and injector channel J. Note that when in alternating mode, the firmware injects 2 squirts per cycle.
  - Simultaneous (1 squirt) - Squirt fuel on injector channel I and injector channel J simultaneously. Note that when in simultaneous mode, the firmware injects 1 squirt per cycle.

- **Gradual Transition** - This setting enables the ability to gradually reduce the pulse width of the primary injectors and gradually increase the pulse width of the secondary injectors as staged injection engages. This setting is most useful when the secondary injectors are located far away from the primaries on the intake tract. Gradual transition and associated settings are not available when using table-based staged injection. The following settings are available:

  - On - Gradual staging is enabled. The associated settings will be used to control how quickly the secondaries are engaged.
  - Off - Gradual staging is disabled. When staged injection engages, the pulse width will go immediately to the fully staged width.

- **Transition events** (ign events) - This setting is used when gradual staging is enabled to determine how many ignition events it will take to go from no staging to fully staged.

- **Primary reduction delay** (ign events) - This setting is used to delay the reduction of the primary pulse width by the number of ignition events after the secondaries start coming online. It is only enabled when the gradual transition feature is enabled.

- **Secondary Enrichment** (ms) - This setting is used to inject more fuel on the secondaries than the fully staged calculation determines. This is useful when there is a small lean spot just after staged injection fully engages. This setting is only available when the gradual transition feature is enabled.

- **Primary staging threshold** (units) - This setting is used to determine when staged injection engages when using RPM, MAP, TPS, or Duty. It is not available when using table-based staging.

- **Primary staging hysteresis** - This setting is used to determine when staged injection shuts off. For example, if RPM is selected as the staged injection first parameter, and the primary staging threshold is 4100, and the primary staging hysteresis is 300, then staged injection will shut off the secondaries and return the primaries to their normal pulse width at 3800 rpm. This setting is not available when using table-based staging.

- **Second parameter** - This setting is used to add a second method for determining when to stage. Note that this setting should not be set the same as the first parameter setting. This setting is not available with table-based staging. The following settings are available:

  - Off
  - RPM

  - MAP
  - TPS
  - Duty

- **Secondary staging threshold** (units) - This setting determines the RPM, MAP, TPS, or Duty at which the secondary staged parameter allows staged injection to engage. This setting is not available unless a second staging parameter is enabled.
- **Secondary staging hysteresis** (units) - This setting is used with the secondary staging threshold setting to determine when staged injection will shut off. This setting is not available unless a second staging parameter is enabled.
- **Secondary staging logic** - This setting is used to determine whether both the primary and secondary staging parameters must be met to engage staged injection, or if only one of the parameters must be met. This setting is not available in table-based staging mode.
- **Relay Output -** Allows using a relay to power secondary injectors in systems where the primary and secondary injectors are both wired to the same ECU output pin, with the secondary injectors receiving switched power from a relay. This tends to produce a jerky, abrupt transition, and is mainly in place to support plug and play applications where the factory wiring lacks separate outputs for primary and secondary injectors.
- **Relay Output Polarity** - If set to normal, turning the output on turns on the secondary injectors. Inverted turns this off to activate the secondary injectors.
- **Pre-Relay Output** - Allows activating a relay before the secondary injectors are active. This is usually to allow a secondary fuel pump to pressurize the second stage injector rail.
- **Pre-Relay Output Polarity -** If set to normal, the pre-relay is turned on before turning the injectors. If set to inverted, the pre-relay output is set to off before turning on the secondary injectors.
- **Pre-Relay Timer -** Specifies the time, in seconds, between turning on the pre-staging relay and the secondary injector stage.

7.2.3.1 Tuning Staged Injection Our recommendation is that on any setup with secondaries are further up the intake tract than the primaries, that you should use table-based staging. It is possible to achieve a much smoother transition to staged injection in all situations when tuning with this method.

7.2.3.2 Tuning Table-based Staged Injection The following tips should be followed when tuning table-based staged injection:

- RPM and Load transition bins - Make the two RPM bins and two Load bins where staged injection first engages close together. Also make the staging percent jump to 10-20% almost immediately (as shown in the dialog at the beginning of the staged injection section). This is so that a very small amount of time is spent with the secondary injectors at or near the injector opening time for those injectors. Spending a lot of time near the injector opening time can lead to inconsistent fueling, especially if the secondary injector opening time has not been determined and the default value is being used.
- Transition to 100% engaged - The transition to 100% engaged should be determined using experimentation. In general, the transition should be set so that the primaries stay close to their maximum duty cycle (80% is recommended) for as long as possible. This ensures that reduction to the primary pulse-width does not result in a lean situation. The table displayed at the beginning of the staged injection settings section is a good example of how to tune table-based staging for a smooth transition on a naturally aspirated engine.

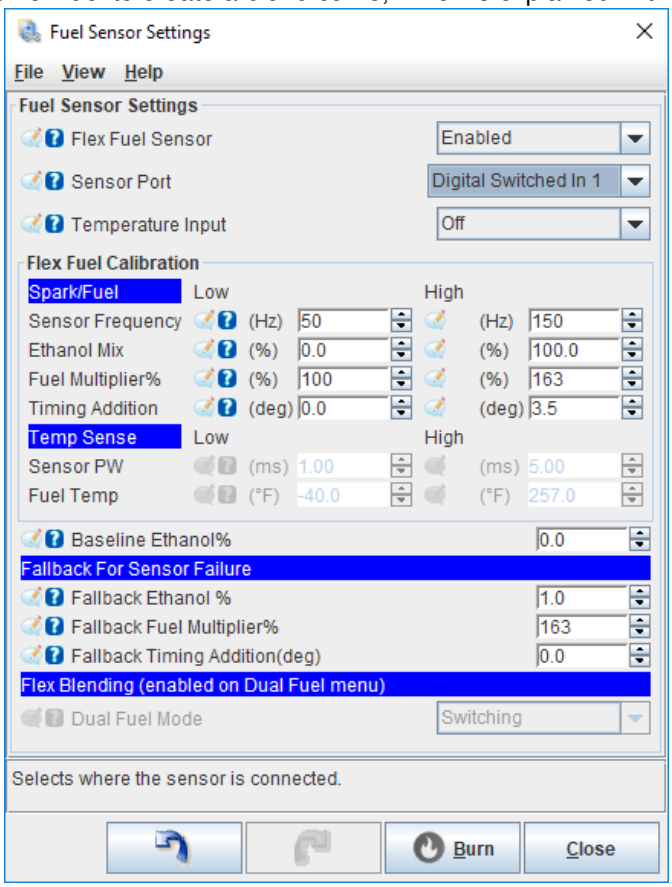
7.2.3.3 Tuning All other Staged Injection Modes All staged injection modes that do not use the table to determine the staging amount can be tuned similarly. The following tips should be used:

- Primary staging parameter - It is usually recommended that the primary staging parameter used is Duty. This ensures that staged injection engages when the duty cycle of the primary injectors warrants it instead of trying to guess what RPM or load will cause use of the secondaries to be necessary.

- Secondary staging parameter - It is recommended that this parameter only be used with forced induction engines. It should be used to ensure that staging is fully complete before going into boost so that any lean spots caused by staging are gone.
- Gradual transition - The gradual transition code was introduced to try to solve the same problems that table-based staging solves, notably the small lean spot in AFR briefly after staged injection engages. It should be set to transition over as many ignition events as possible for the smoothest transition. If doing this still causes a lean spot, the primary reduction delay can be used along with the secondary enrichment setting to make sure that slightly more fuel than calculated using the normal fuel calculations is injected. If enabling the gradual transition feature still does not get rid of the brief lean spot after staging is engaged, it is recommended that table-based staging is used.

7.2.4 Fuel sensor settings (Flex)

The flex fuel system allows the use of E85 or other ethanol and petroleum blended fuel mixes to be used when a special fuel composition. sensor is fitted. The support is specifically based around the use of a GM flex fuel sensor which outputs a varying frequency signal depending on the fuel composition. This information can then be used to add more fuel and advance timing. Ethanol blends need more fuel volume than unblended fuel and different advance. The basic system uses a two-point system (low/high) and interpolates between the values. You can also use flex fuel to create a blend curve, which is explained in the section on blend curves below.



- **Flex fuel sensor** - this enables or disables the Flex Fuel system
- **Sensor port** - where the sensor is connected. A GM flex fuel sensor typically has pin A as ground, B as 12 volt supply, and C to the selected input pin.
- **Temperature input** - GM's flex fuel sensors can output a fuel temperature signal based on their pulse width. You can also specify a separate analog sensor, enabled under Generic Sensors.
- **Spark** / **Fuel settings -** These have separate low and high settings to define the range of sensor outputs.

  - Frequency - Specifies the minimum and maximum frequencies the senor will send.
  - Ethanol mix - Specifies the percentage ethanol at minimum and maximum frequency.
  - Fuel multiplier - Ethanol requires more fuel. This is a percentage multiplier that scales how much fuel is added.
  - Timing addition - If not using a blended table, this simply adds a fixed amount of timing, interpolated between the low and high points.

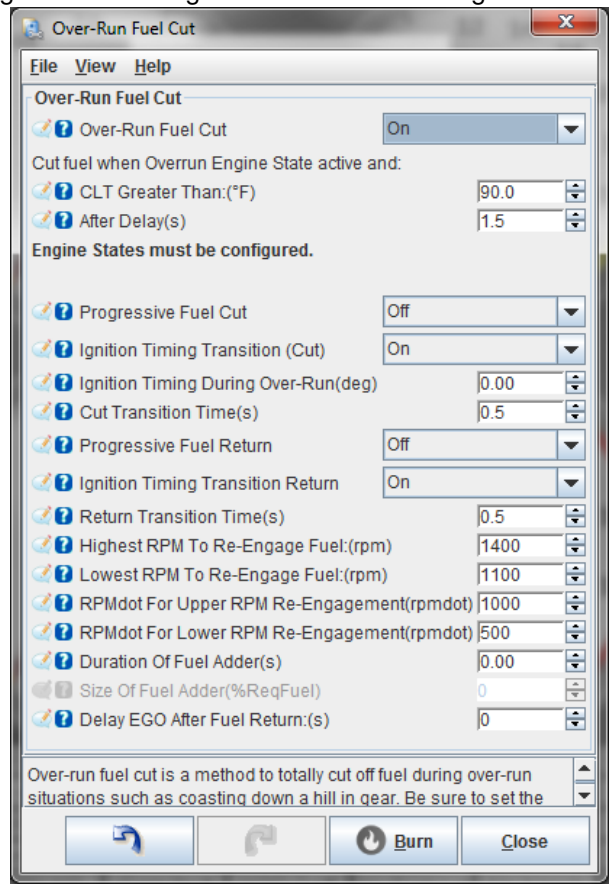
- **Temp Sense** - Allows setting up pulse width based fuel temperature output from a flex fuel sensor.
- **Baseline Ethanol %** - The fuel and ignition table blending will assume VE1 and Spark Table 1 were tuned at this percentage. Typically set to 10% if the engine is not tuned on known ethanol free fuel.
- **Fallback Ethanol %** - The MS3 will treat an out of range sensor signal as this ethanol percentage for purposes of table blending and other functions driven off ethanol percentage.
- **Fallback fuel multiplier** - If the sensor frequency goes out of range, the MS3 will apply this multiplier to the fuel to avoid a lean condition.
- **Fallback timing addition** - How much to adjust the timing in the event of a sensor failure.

Set the fallback results to give the safest results if the sensor fails. This is typically a low ethanol percentage if running boost table blending, the same fuel multiplier as on the high ethanol mix, and the same timing addition as the low ethanol mix.

- **Dual fuel mode** - The ability to blend tables is enabled under the Table Switch / Dual Fuel screen.

7.2.5 Over Run Fuel Cut

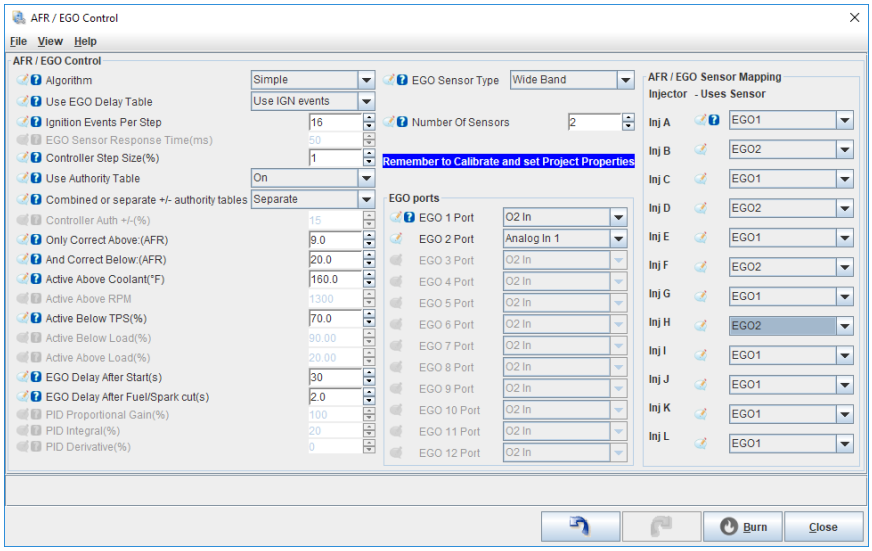
The MS3Pro not only allows you to cut off fuel while coasting; it also allows retarding spark timing at this point, and gradually phasing the fuel cut in and out to make for a smoother transition. Overrun fuel cut relies partially on engine state settings to determine if the engine is in overrun mode.



- **Over-run fuel cut** - Allows turning this feature off or on.
- **CLT greater than** - Allows locking out fuel cut on a cold engine.
- **After delay** - Specifies how many seconds the MS3Pro will wait after trigger conditions are met before engaging fuel cut.
- **Progressive fuel cut** - Allows cutting cylinders one at a time to smooth the transition.
- **Ignition timing transition** - Changes the ignition timing during overrun fuel cut.
- **Ignition timing during over-run** - The absolute timing during overrun fuel cut. Positive numbers are BTDC and negative numbers are ATDC.
- **Cut transition time** - The amount of time to transition in the spark retard and progressive fuel cut.
- **Progressive fuel return** - Allows gradually returning fuel. If using this setting, we recommend setting the fuel adder duration and size to zero.
- **Ignition timing transition return** - Allows gradually transitioning from spark retard to normal timing. This can prevent the fuel return from causing an abrupt increase in power and make for a “softer feeling” re-engagement.
- **Return transition time** - The time, in seconds, from re-engaging fuel to returning to full fuel (if progressive fuel return is used) and ignition timing (if ignition timing transition return is used).
- **Lowest RPM to re-engage fuel** - If the RPM falls below this value, fuel is turned back on. Typically a few hundred RPM lower than the minimum RPM for fuel cut.
- **RPMdot for upper RPM re-engagement** - When revs are falling more quickly than this RPMdot value the initial RPM is used for fuel return. In-between upper and lower RPMdot values, the return RPM is interpolated. This number must be larger than the ’lower’ RPMdot setting.
- **RPMdot for lower RPM re-engagement** - When revs are falling more slowly than this RPMdot value the lower RPM is used for fuel return. This will happen when coasting down. If revs are falling more quickly then any return transitions will be canceled and the return RPM will be raised. Set this so that clutch-in during fuel cut does not stall.
- **Duration of fuel adder** - Adds additional fuel when fuel cut disengages to avoid a lean spike. This is how long to apply the additional fuel, in seconds. This can be used with progressive fuel return, but is likely to give undesirable results.
- **Size of fuel adder -** A percentage multiplier to control how much additional fuel is added when fuel cut disengages.
- **Delay EGO after fuel return** - Turns off closed loop correction for this many seconds after fuel cut deactivates, to give the time for the air to clear out of the exhaust system.

7.2.6 AFR / EGO control

This screen allows you to set up closed loop O_2 sensor feedback.



7.2.6.1 Basic EGO settings These settings are used to control the behavior of the closed-loop EGO algorithm.

- **Algorithm** - There are three options here:
  - Simple - This method of closed-loop EGO control is well-suited to use with a narrowband O_2 sensor. If the current AFR (or EGO voltage for narrowband) incorrect, the amount of fuel being injected is adjusted by Controller Step Size every Ignition Events per Step ignition events. This method often results in the actual AFR oscillating above and below the target.
  - PID - This method incorporates a Proportional Integral Derivative control loop which tuned properly, adjusts the amount of fuel being injected to quickly get to the target, and then maintains the target without any oscillation (when tuned correctly). You tune the three proportional, integral, and derivative numbers below when this mode is active.
  - No correction: The O_2 sensor is used for data logging only and is not used for closed loop control.
- **Use EGO Delay Table** - You can select how often the EGO correction is updated based on either the number of ignition events or by using a 12 x 12 table that determines how often to update the correction as a function of RPM and engine load.
- **Ignition Events per Step** - While closed-loop EGO is active and the EGO delay table is off, how often the correction is run is determined by this setting. It is the number of ignition events per correction.
- **Controller Step Size** - This setting is only used with the "Simple" EGO algorithm. It controls how large each correction "step" is. So if the AFR does not match the desired AFR, and Controller Step Size is configured to be 1%, each time a correction is made, that correction will be 1%.
- **Use Authority Table** - Turning this on will allow using a table to determine how the maximum amount of correction that can be applied applied as a function of load and RPM. For example, you might want to give a large amount of control authority while cruising at light throttle, but reduce controller authority to zero under boost to protect against the consequences of a failing oxygen sensor leaning out the engine under load.
- **Combined or separate +/- authority tables** - If set to “combined”, the numbers in the EGO Authority Table set the maximum amount of fuel that the MS3Pro can add or subtract. If set to “separate”, the EGO Authority Table sets the maximum amount of fuel the correction can add, while the EGO Authority Table Negative sets the maximum amount of fuel the correction can subtract. You can use this feature if you are concerned a sensor failure or measurement inaccuracies might subtract too much fuel at full load.
- **Controller Auth** - This setting controls the maximum amount of adjustment performed by the closed-loop algorithm if not using the authority table.
- **Only correct above (AFR)** - AFR readings below this level are not used and correction is turned off when outside this reading. Used to turn off EGO correction if the sensor is out of its accurate range.

- **And correct below (AFR)** - This parameter allows setting a maximum AFR for correction in combination with the above setting.
- **Active Above Coolant** - Below the temperature defined by this setting, closed-loop EGO will not activate.
- **Active above RPM** - Below the RPM defined by this setting, closed-loop EGO will not activate.
- **Active Below TPS** - Above the throttle position defined by this setting, closed-loop EGO will not activate.
- **Active below Load** - Above the load defined by this setting, closed-loop EGO will not activate.
- **Active Above Load** - Below the load defined by this setting, closed-loop EGO will not activate.
- **EGO delay after start** - The time in seconds after engine-start before closed-loop EGO can be activated.
- **PID parameters** - These numbers are used to tune the correction when the algorithm is set to PID.
- **EGO Sensor Type** - This setting enables EGO control and allows the user to choose between using a wideband sensor or narrowband sensor. The following settings are supported:
  - Disabled - No EGO sensor enabled. **Note that this will also disable AFR logging.** If you wish to log your EGO sensor information, but do not want to use closed loop correction, enable your sensor type and set the correction algorithm to “No correction” instead.
  - Narrowband - Sensor in use is a narrowband sensor.
  - Wideband - Sensor in use is a wideband sensor.
- **Number of Sensors** - This setting is used to select the number of oxygen sensors in use by the MS3Pro. You can use up to 12 sensors.

7.2.6.2 EGO ports The EGO Ports settings allow the user to select the input port used to read the signal from the oxygen sensor. The number of EGO ports available depends on the number of sensors selected. The “Normal EGO” input is reserved for the primary O_2 sensor, but you can optionally use one of the three other analog inputs for the primary, or an O_2 sensor on a CAN network. The second and subsequent sensors may not use “Normal EGO”, but may use any spare analog input or a CAN device.

7.2.6.3 AFR/EGO Sensor Mapping Use this menu to assign which injector channel is associated with what O_2 sensor input.

7.2.6.4 Tuning

Simple algorithm with narrowband sensor A narrowband sensor is only accurate at exactly stoichiometric mixtures for the fuel being used (14.7:1 for gasoline). At around 0.5 volts, the mixture is stoichiometric. For leaner mixtures (above 14.7:1 for gasoline, above 1.0 lambda) the voltage dips slightly below 0.5 volts. For richer mixtures, the voltage goes above 0.5 volts. This behavior means that it is not possible to hold an exact mixture when running closed-loop with a narrowband sensor.

Because of this, the best algorithm to use with a narrowband sensor is the "simple" algorithm.

The simple algorithm adjusts the mixture richer if the sensor reads lean, and leaner if the sensor reads rich. It adjusts **Controller Step Size** percent every **Ignition Events per Step**. This can lead to a small oscillation in O_2-based correction once the AFR reaches close to stoichiometric.

The following steps are recommended when tuning the simple algorithm with a narrowband sensor:

1. Ignition Events per Step - When first tuning the engine, this should be set to a fairly low number (4-8) so that if the AFR is very far off, it is corrected quickly. Once the engine is better tuned, this number can be switched to a higher number to gain more stable correction behavior (8-16 or more).
2. Controller Step Size - When first tuning the engine, this should be set to 2% so that when correcting, the engine reaches stoichiometric quickly. Once the engine is well tuned, this should be reduced to 1% to gain more stable correction.

3. Controller Auth - This is the maximum percentage of fuel change the MS3Pro can make as a result of O_2 sensor feedback. When first tuning the engine, this should be set to 20% or higher. Care must be taken to watch how the algorithm is correcting. In some situations, it is possible for the sensor to read very lean when really the engine is running very rich. Once the engine is tuned, this should be set between 5% and 10%.
4. Engagement Settings - Most of the remaining settings control how and when the closed loop algorithm is engaged. Engagement with a narrowband sensor should happen when the engine is nearly fully warm, 500-1000 rpm above idle, below 80% throttle, below about 80% load, just above the lowest load seen when barely pressing the throttle, and at least 30 seconds after the engine starts. These settings are because the sensor must be hot to operate, must not be used at high load due to the fact that the engine should be operated rich of stoichiometric, and must not be used at very low load because the oscillations will cause the engine speed to oscillate.

Simple Algorithm with Wideband Sensor Tuning the simple algorithm with a wideband sensor is essentially the same as tuning it with a narrowband sensor with the caveat that the AFR target table is used to set the AFR target. It is still recommended that the EGO algorithm not be used at high throttle position/load due to the fact that the accuracy of the wideband sensor decreases dramatically with pressure and temperature changes caused by high load.

PID Algorithm with Narrowband Sensor When using a narrowband sensor with the PID algorithm, all the same recommendations for settings given in the section describing the Simple algorithm should be followed.

Additionally, since it is nearly impossible to keep the narrowband sensor from oscillating, it is recommended to start by tuning the "I" term until the target is reached with minimal oscillation. Once this point is reached. It is recommended that very little (if any) "P" term is used since the "P" part of the PID algorithm causes instantaneous reaction, and the response of the sensor is not proportional to the distance from stoichiometric.

PID Algorithm with Wideband Sensor When using a wideband sensor with the PID algorithm, the same steps as when using a narrowband sensor can be followed for tuning the "I" term.

Additionally, since the response of most wideband controllers and sensors is linear with AFR, a larger "P" term can be used to help correct for fast changes in AFR. Caution must still be used however since there is a significant delay between the amount of fuel being injected changing and MS3 registering an AFR change as a result.

Finally, a small amount of "D" term can be used to help slow response during very fast changes. This helps reduce overshoot of the target.

7.2.7 Narrowband EGO targets

This is a 3D table that behaves like an AFR table, but instead of specifying a target air/fuel ratio, it specifies what voltage from a narrow band the MS3Pro should target.

7.2.8 EGO Authority Table

This table allows you to specify the maximum correction that the MS3Pro can apply as a result of O_2 sensor feedback as a function of RPM and load, in percent. To enable using this table, set “Use Authority Table” in AFR / EGO Control to On. If using separate EGO authority tables, the EGO Authority Table sets the maximum fuel that the MS3Pro can add, while the EGO Authority Table Negative sets the maximum that can be removed.

7.2.9 EGO delay table

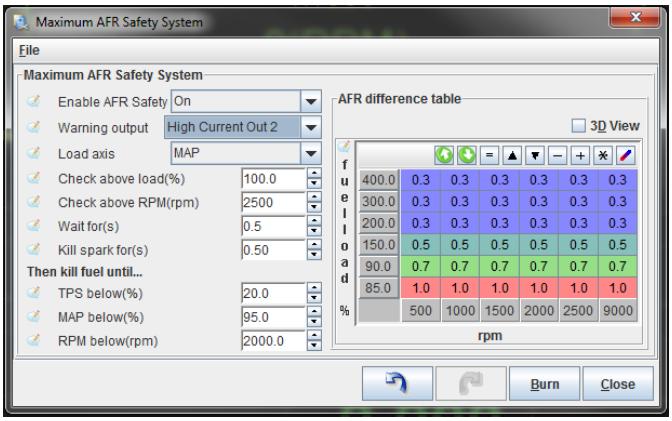
This table specifies how often to update the EGO correction as a function of RPM and load. The numbers in this table represent milliseconds. To enable using this table, set “Use EGO Delay Table” in AFR / EGO Control to “Use delay table”.

7.2.10 AFR safety system

The "AFR Safety" System is designed to help you protect your engine from unwanted dangerous lean conditions. Typically this will be used for wide open throttle and/or boosted conditions. It only makes sense to use this feature with a wideband lambda sensor on an already well tuned engine. **Do not use AFR Safety before the engine is fully tuned**. The system compares the wideband EGO input against your existing AFR target table and a safety limit difference table. If this situation persists beyond a time you specify, the engine will be shutdown as a protective measure until you get out of boost, off the throttle and down in revs. NOTE This system will only work if configured correctly and will not protect your engine against bad tuning or knock! However, it is likely it would detect a failing fuel pump or regulator. A single faulty injector could easily go unnoticed with a single wideband reading the average AFR.

This shutdown settings on this screen will also trigger off of other criteria for shutting down the engine. Other failsafe settings that can use the shutdown routine are:

- Fuel pressure safety
- EGT safety
- Low water injection fluid
- Oil pressure



Settings:

- **Enable AFR Safety** - This enables or disables the system
- **Warning output** - This can be used to turn on an output (such as a warning light or buzzer) as soon as an over-lean condition is detected.
- **Load axis** - Typically choose MAP if running speed-density or TPS if running Alpha-N
- **Check above load** - The system only checks the AFR above this load.
- **Check above RPM** - The system only checks the AFR above this rpm
- **Wait for** - This delay allows for a brief lean-spike without tripping the shutdown. Set as low as possible.
- **Kill spark for** - Initially spark is cut for this time and then fuel as well.
- **TPS below, MAP below, RPM below** - normal engine operation resumes when all three conditions are met. The table on the right is the "safety limit" difference table.

Each install is different and it is up to the installer and tuner to decide on safe limits! The example data in the screenshot above has the system only active above 95kPa and 2000rpm. Once in this region, the safety limit difference table is the controlling factor. For example, if at a certain point you have set your target AFR (AFR table) to be 11.9, and you then decide that a worst case of 12.3:1 AFR is allowable, the safety limit table would have 0.4 in that position. i.e. allowed AFR = target AFR + safety limit = 11.9 + 0.4 = 12.3

If then your wideband EGO sensor reports 12.4 or leaner, the warning output will be activated immediately. (Typically this would connect to a dash mounted warning light.) If this situation persists beyond the time limit

(shown as 0.5 seconds here) then the shutdown mode begins. If the AFR returns to a "safe" level before the time limit elapses, no shutdown takes place. Review datalogs before setting this time. The lower it is, then the sooner the shutdown kicks in. This could help protect your engine, but setting it too low may cause annoying false alarms.

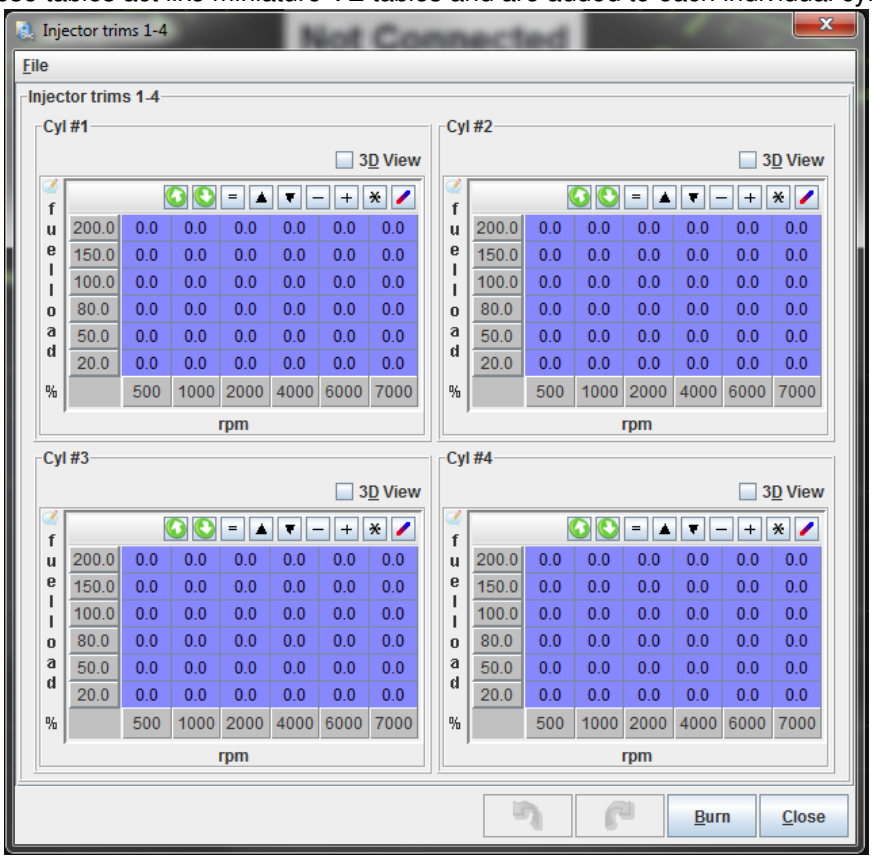
The first step in the shutdown procedure is to cut spark for a specified time (shown as 0.5 seconds here.) During this period fuel continues with the intention of cooling the engine internals, as the lean condition is likely to have generated excess heat. On vehicles fitted with a catalytic converter, 0.0 should normally be specified, as catalysts are not designed to cope with raw fuel.

After the "kill spark for" time, fuel is also cut.

Normal engine operation will resume only when your Throttle, MAP and RPM are less than the limits you set. In operation, if you trip the shutdown, the engine will suddenly die. While this will be irritating, hopefully it is less so than the engine rebuild you might have needed.

7.2.11 Cylinder trim tables

These tables act like miniature VE tables and are added to each individual cylinder.



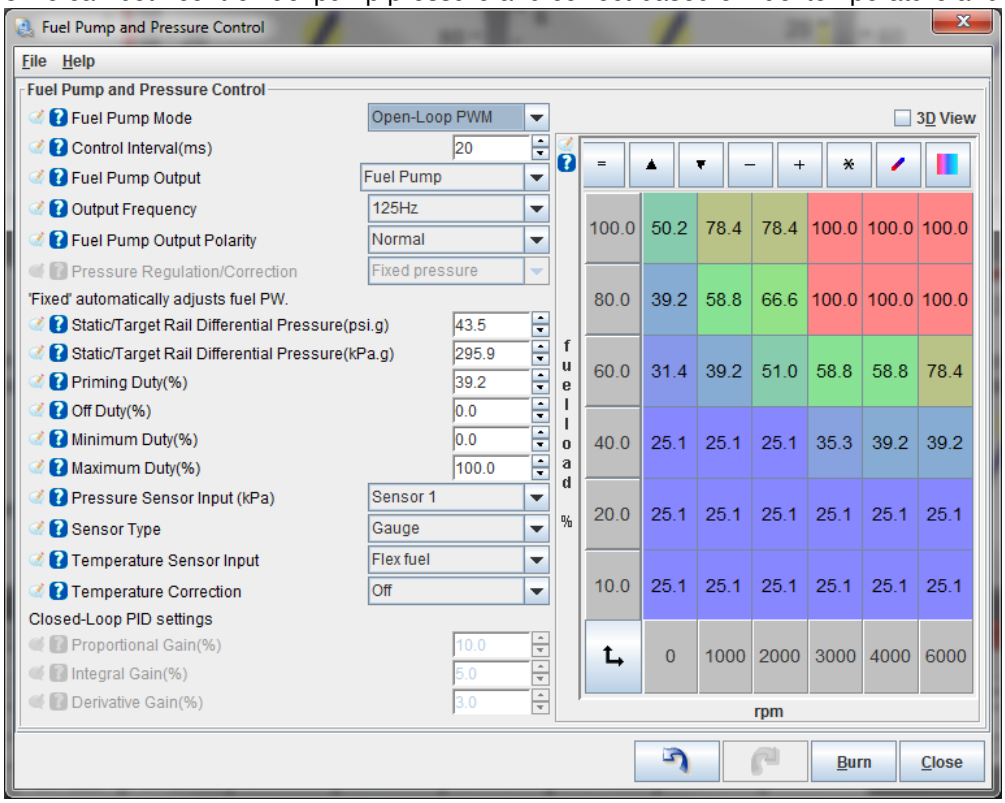
7.2.12 Sequenced Batch Fire

These settings allow batch firing to be sequenced from a specific trigger number for repeatable phasing on each start if you are running batch fire fuel with a sensor input mode that allows for wasted spark or coil on plug. As standard, batch fire sequencing is "random" at the change from cranking mode to run mode. This setting forces the sequence to always start on the same "trigger" on each startup. This can help if you are running batch fire injection and you find the engine runs differently from startup to startup; however, it also requires an ignition type that supports cylinder identification.

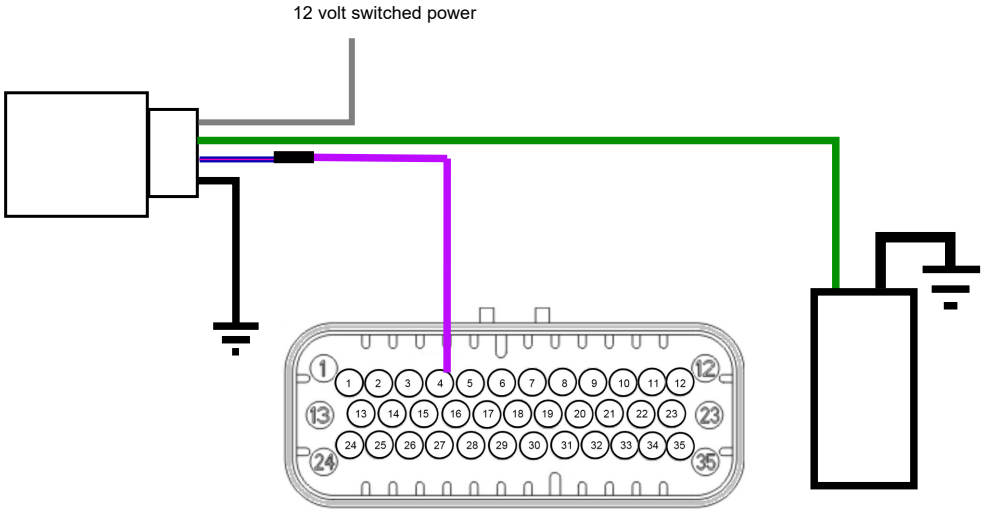
On an engine running wasted spark, the trigger numbers start at one and count up to half the number of cylinders, with trigger 1 corresponding to the number one cylinder firing event. On an engine running sequential coil on plug with batch fire injectors, the trigger numbers again start at 1 for the number one cylinder firing event and count up to the number of cylinders.

7.2.13 Fuel pump and pressure

MS3Pro can both control fuel pump pressure and correct based on fuel temperature and pressure.



PWM fuel pump control requires using a solid state relay or other fuel pump control module to supply power to the fuel pump. See section 4.11.1 for more information about solid state relays.The diagram below shows an example of running a fuel pump from the MS3Pro using a solid state relay.

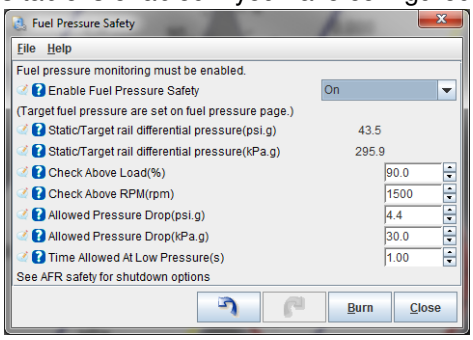


Here are what the settings do.

- **Fuel Pump Mode** - Specifies the following ways of controlling the fuel pump.
  - On/Off: The fuel pump is switched on and off. Pressure is regulated mechanically. The MS3Pro can monitor pressure and correct for it, but is not able to control the pressure directly.
  - Open-Loop PWM: Used for pump speed control in combination with a regulator. The table on the right hand side controls fuel pump duty as a function of RPM and fuel load.
  - Closed loop PWM: The MS3Pro uses a fuel pressure sensor to target a specific pressure. Can be used in an unregulated, return-less setup.
- **Control interval** - How often, in milliseconds, the MS3Pro updates its fuel pressure and temperature related calculations.
- **Fuel pump output** - Allows changing the fuel pump output pin in PWM mode. If using another output pin than the regular fuel pump output, the main fuel pump output pin will still operate in an on/off mode.
- **Output frequency** - The frequency used for PWM output.
- **Output polarity** - Normal polarity grounds the output to turn the fuel pump on and releases the output for the off signal. Inverted reverses this.
- **Fuel pressure regulation** / **correction** - By preference, fuel systems use a vacuum/boost referenced regulator that maintains line pressure a fixed pressure above manifold pressure. Some GM fuel systems use a fixed fuel pressure referenced to ambient air pressure instead. The vac referenced systems give even injector flow across all inlet pressure conditions and requires no compensation. The un-referenced systems give a varying flow and require a calculation within the ECU to compensate. Enabling 'Fixed' here turns on that automatic compensation. Setting this to a custom curve allows you to use a user-adjustable compensation curve based on a fuel pressure sensor.
- **Static** / **Target Rail Differential Pressure** - Allows you to specify the target fuel pressure in gauge pressure. You can specify the pressure in both psi and kPa. Adjusting one will automatically adjust the other.
- **Priming duty** - When using PWM fuel pump control, specifies the duty cycle to use during the initial priming pulse.
- **Off duty** - The duty cycle to command when the fuel pump is deactivated.
- **Minimum duty** - The lowest duty cycle that produces fuel flow. 0% for most applications, but Ford FPDM modules often need this set to 5%.
- **Maximum duty** - The highest duty cycle to command. Most applications will go to 100%, but Ford FPDM modules need this set to 50%.
- **Pressure sensor input** - enables a fuel pressure sensor signal. Required if using closed loop PWM. Used for correction on other modes.
- **Sensor type** - MS3Pro supports many different fuel pressure sensor types.
  - Gauge - The sensor is referenced to atmospheric pressure, and read zero with no fuel pressure. With a vacuum referenced regulator, the reported pressure will drop under vacuum and increase under boost.
  - Absolute - These sensors are referenced to absolute vacuum, and read about 14.7 psi when there is no fuel pressure.
  - Differential - Differential sensors are referenced to manifold pressure, and will read a constant value when the fuel pump is on if everything is working correctly.
- **Temperature sensor input** - Allows using a fuel temperature sensor.
- **Temperature correction** - If using a fuel temperature sensor, turns on a temperature correction curve.
- **Closed loop PID settings** - Changes the sensitivity of the closed loop PID control.

7.2.14 Fuel pressure safety

This table is enabled if you have configured a fuel pressure sensor under Fuel Pump and Pressure Control.



Fuel pressure safety is a failsafe feature that allows shutting down the engine in the same way as AFR Safety if the fuel pressure goes out of range. You can specify what RPM and load to use when checking, as well as the maximum allowable pressure drop and how long this can continue before triggering a shutdown.

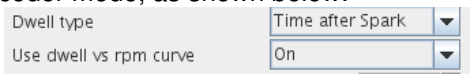
7.3 Ignition Settings

7.3.1 Dwell battery correction

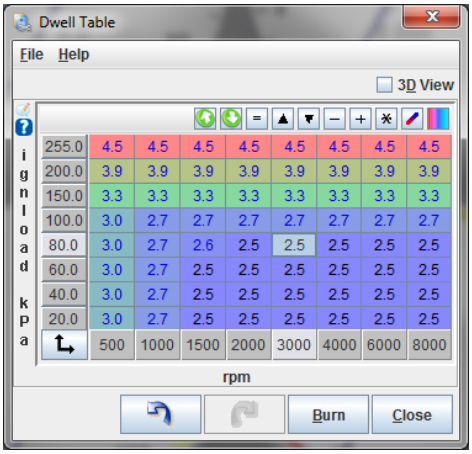
The dwell battery correction/adjustment curve is a built-in curve to automatically apply greater spark dwell at lower battery voltages. This is required for inductive ignition systems - more time is required to charge the coil with less voltage. The default curve as shown is suitable for most applications. Note that it is centered on 14.4V with 100% - at this voltage the actual dwell time matches the dwell entered on the ignition options page.

7.3.2 Dwell vs RPM

The Dwell vs. RPM curve is a feature intended to be used with Saab Tri-Ionic ignitions only. These require "Time After Spark " to be enabled. The dwell vs. RPM curve must also be turned on under the Ignition Options / Wheel Decoder mode, as shown below:



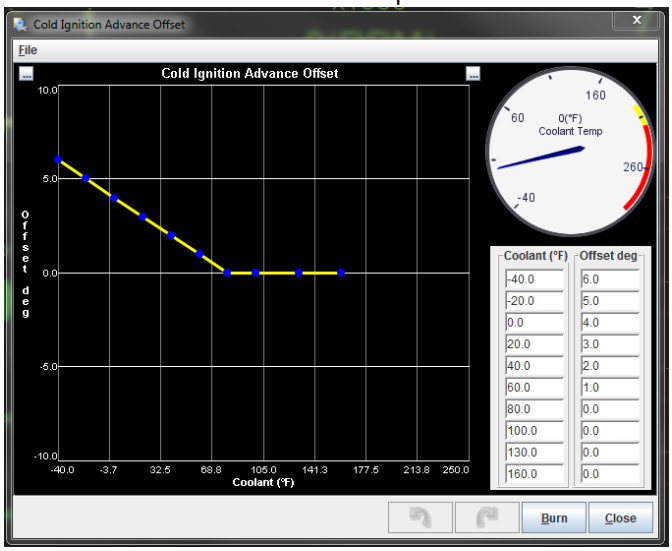
7.3.3 Dwell Table



This table allows you to specify dwell (before battery voltage correction) as a function of RPM and ignition load. This allows increasing dwell under boost, then backing it off at low load to reduce the amount of heat put into the ignition system.

7.3.4 Cold advance

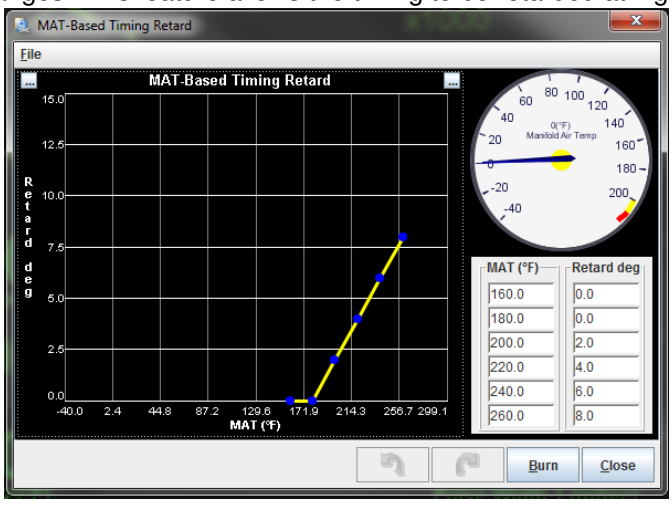
The cold advance feature can be used to run more advance on a cold engine. When cold, an engine typically requires more advance for normal operation. Although the function's original intended purpose is to add more timing advance with the engine cold, this can adjust timing in either direction at any temperature. Many OEMs are reported to retard timing during warmup to more quickly bring the catalyst up to operating temperature; you can also use the cold advance table to replicate this function.



Ensure that the advance figure in the highest temperature point is zero; this will allow the spark timing to follow the advance table when warmed up.

7.3.5 MAT based timing retard

High inlet air temperatures (MAT) can make an engine more prone to knock; this is especially the case on boosted engines where small changes in ambient air temperature can produce much larger post-compressor temperature changes. This feature allows the timing to be retarded at higher MAT readings for greater engine safety.



Ensure that the retard figure at normal temperatures is zero; this will allow the spark timing to follow the advance table during normal operation.

7.3.6 Noise filtering

The noise filtering menu allows the MS3Pro to detect and ignore pulses on the crank and cam position inputs caused by electrical noise.

In the simplest form of a fuel-only controller with a tach signal from the coil negative, the "ringing" of the coil may cause apparent rpm spikes. This will cause additional fuel to be injected on each of these noise pulses and

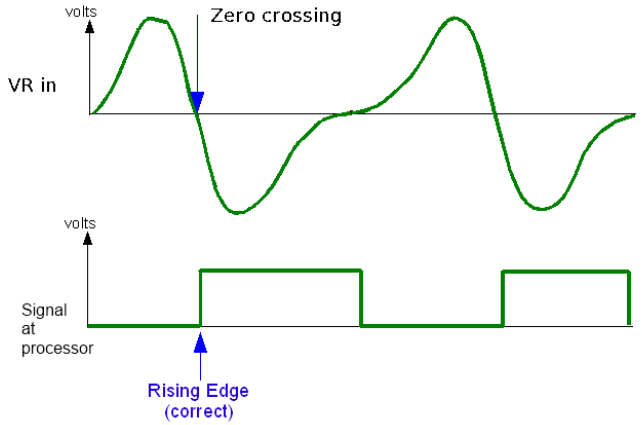
show up as a falsely high rpm. It will likely be irritating, but no major problems are likely.

On an install controlling ignition, it is a different matter. In the simple distributor install with a simple trigger arrangement (1 pulse per ignition event like in a normal distributor), the noise will cause a misfire probably on that one event. On a multi-toothed wheel, the noise is likely to cause a loss of sync with the wheel, and spark will be lost for 360 or even 720 degrees of engine rotation before the re-synchronization occurs.

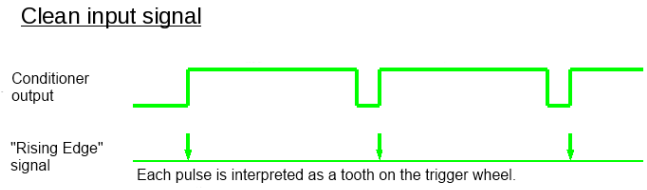
So, the tach input is critical to ECU operation, it is important to reduce or eliminate that noise. Hardware is your first line of defense. Here are some tips on setting up hardware to keep out noise:

- Set the sensor gap to an appropriate level. Most sensors work with a gap about the thickness of a normal business card.
- Use shielded wire, like in our standard wiring harness. Ground the shield at only one end - a shield grounded at both ends can carry current, and induce noise in the signal.
- Route the signal wire away from any electrical noise sources, such as ignition coils, spark plug wires, and the alternator or starter.
- Ensure that the MS3Pro has a clean ground to the engine block, cylinder head, or negative battery terminal.
- VR sensors need to be connected to the positive and negative signal wires. Hall effect or optical sensors should be grounded to the MS3Pro sensor ground.

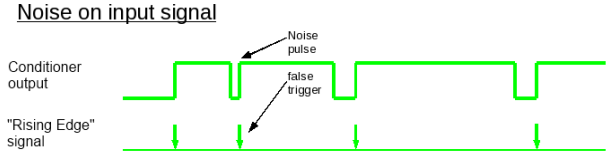
With the hardware well defended against noise, the second line of defense is to set up the ECU to read the sensors correctly and reject noise. The first setting is the ignition input capture is correct, as covered in section 5. For example, the VR signal below has one stable and one unstable edge. It needs to be set to triggering on Rising Edge.



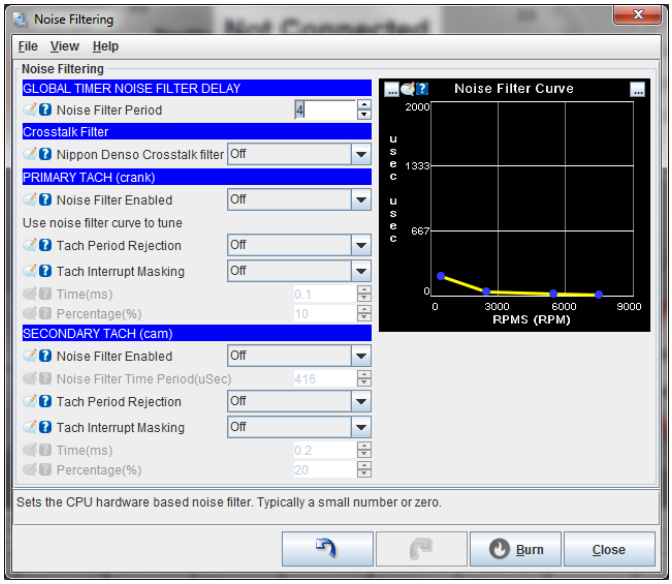
Setting to Falling Edge would result in an unstable signal - you'll see this as timing scatter and the RPM bouncing around. And this is with a fairly clean signal, which the processor would see like this if triggering on the correct edge:



If the wiring or sensor isn't perfect, and the signal becomes noisy, the CPU will see a signal like this instead:

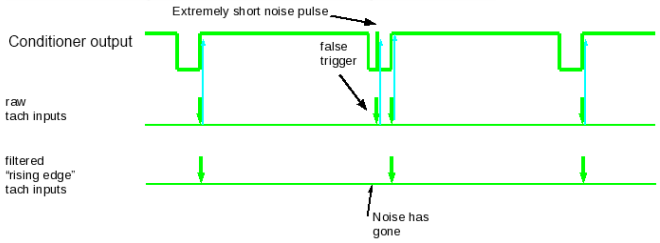


There are several strategies the MS3Pro has to filter this out, which you enable from the Noise Filtering screen. **Use caution; an overly aggressive noise filter can easily cause the MS3Pro to filter out real pulses.**



- **Noise filter period** - this sets a minimum pulse time to ignore, measured in clock cycles. Each clock cycle is 0.02 microseconds, so a value of 4 corresponds to 0.08 microseconds. It can filter out very short noise spikes, like this:

Noise filter period - minimum pulse time

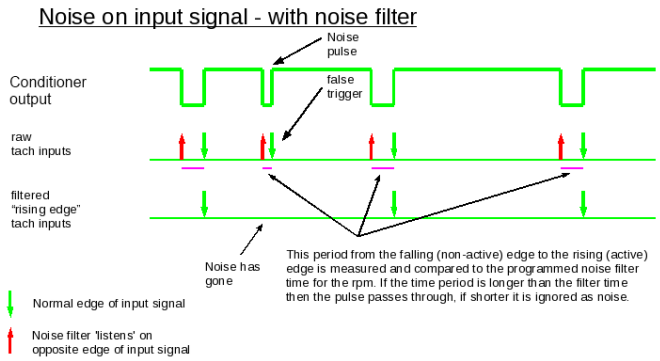


Any pulse shorter than this time is considered to be noise and thrown out.
Primary Tach settings apply to the crankshaft position (CKP) input.

- **Nippon Denso Cross-talk filter** - This setting is enabled if the spark mode is set to “Toothed wheel” and the trigger wheel arrangement set to “Dual wheel”. It is intended to handle Nippondenso cam angle sensors that incorporate two separate VR sensors with a common ground. This setting checks the status of the CKP input when it detects a signal on the CMP line. If the MS3Pro has received a pulse from the CKP signal and the CMP signal at the same time, it considers the CMP signal to be a piece of noise induced by the CKP signal, and ignores the CMP signal.

- **Noise filter enabled** - enables noise filter curve on primary/crank tach input. This sets a minimum pulse time versus RPM curve (the **noise filter curve** to the right)

The noise filter works on the principle that the noise is likely to be of short duration, far shorter than a genuine tooth. If all inputs conditioners are setup correctly, we should be able to detect this condition by measuring the width of the input pulse. A "long" pulse is real, a "short" pulse is noise. The width can vary with RPM, so the curve allows you to set a minimum time.



If using a VR sensor, you will need to measure the time between zero crossings with an oscilloscope to determine a good value.

With Hall effect or optical sensors sensors with equally spaced base teeth, you can calculate the number directly. First, find the time per pulse in microseconds:

Time per pulse = 60,000,000 / (Engine RPM * base teeth per crank revolution)

Next, multiply the time per pulse by the ratio of tooth length to total length between the start of each tooth. For example, if each tooth is 3 mm long followed by a 9 mm gap, you have a 1/4 ratio (3 mm / 12 mm total spacing from one rising edge to the next). Then multiply that by a safety factor of 0.6 to 0.8 in case a tooth measures a bit short.

For example, consider an engine with the following details:

- 36-1 crank trigger at crank speed
- Teeth are 6 mm long with 6 mm gaps
- Use a safety factor of 0.6 to avoid missing any real teeth

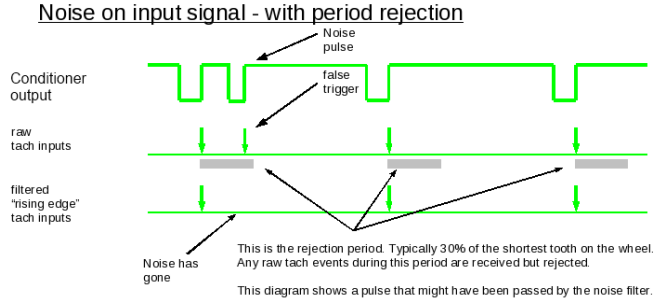
Working the math out at 3,000 RPM would give us the following noise filter threshold:

(60,000,000 / (3,000 * 36)) = 555.6 microseconds per tooth

555.6 microseconds per tooth * 1/2 tooth to gap ratio * 0.6 safety factor = 166.7 microsecond noise filter time

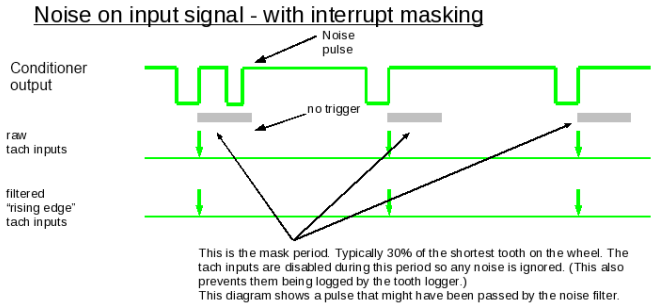
- **Tach period rejection** - ignores pulses shorter than time/percentage

This is a different approach: It ignores any pulse that comes in within either a timeframe or percent of the spacing between the tach pulses. This goes off the timing of the pulse, rather than the length. Note that the spacing used is between the smallest teeth, so this method works best on wheels where most of the pulses are a similar distance apart. Modes like the Subaru 6/7 have several closely spaced pulses (at least on the cam wheel), rendering this method less effective.



- **Tach interrupt masking** - turns off input pin so pulses are not seen at all for time/percentage

This setting is similar to tach period rejection. However, instead of sensing and rejecting pulses in the rejection time, this setting outright turns off the interrupt - it actively stops the ECU from listening during the rejection wheel.



- **Time(ms)** - time period to ignore for interrupt masking or period rejection. The tooth logger (see chapter 8) can determine the lowest time value for normal sensor pulses; make sure the value you set here is at least 25% less than the measured value at maximum RPM to avoid filtering out real pulses. Be careful when using this setting with high revving engines with high tooth count trigger wheels. If you see the MS3Pro failing to detect real teeth at high RPM, decrease this number. Even a value of 0.1 ms is likely to cause trouble with a 60-2 crank trigger.
- **Percentage** - percentage of "tach interval" to ignore. This needs to be smaller than the percentage variation between the largest and smallest gaps in the trigger wheel. For example, GM 7X mode has six notches 60 degrees apart and one notch offset 10 degrees from the next notch. A value of larger than 16% (10 degrees divided by 60 degrees) could cause it to miss the 10 degree notch.

Secondary Tach settings apply to the camshaft position (CMP) input.

- **Noise filter enabled** - enables noise filter time on secondary/cam tach input.
- **Noise filter time period** - pulses shorter than this are treated as noise. Used instead of the noise filter curve. If using a Hall effect or optical cam trigger, take the tooth width in crank degrees (or double the tooth width in cam degrees), multiply it by 50, and divide by 3 times the maximum engine RPM. This will give you the length of a real pulse. Then multiply this by 1.2 to 1.5 to give a safe value for the minimum noise filter period. If using a VR sensor, you will need to measure the time between zero crossings with an oscilloscope to determine a good value.
- **Tach period rejection** - ignores pulses shorter than time/percentage
- **Tach interrupt masking** - turns off input pin so pulses are not seen at all for time/percentage
- **Time(ms)** - time period to ignore.
- **Percentage** - percentage of "tach interval" to ignore.

7.3.6.1 Examples

These should be taken as a starting point and will often need install specific tuning.

- **Coil negative triggering** for a fuel-only install. This requires noise filtering to eliminate the effects of coil ringing. A typical setting would be to enable crank tach period masking at 2ms and 30%.
- **Distributor type pickup**, such as HEI, TFI, or EDIS (although the last is not a distributor, the MS3Pro will see a similar signal.). Noise filtering is often useful here. A typical setting would be crank tach period masking with 1ms and 30%.
- **Wheel decoder and other more advanced installs**. Noise filtering may be useful on some installs. Due to the multi-toothed input signal the rejection time periods need to be set far smaller. e.g. on a V8 with distributor at 6000rpm, the time between tach pulses is 5ms. A 36-1 wheel on the same engine gives a pulse every 0.6ms (600us). Using too large a filter time will filter out the real signal. As a general rule, time numbers should be set to 0.1 ms or less on trigger wheels with 36 or more teeth - and reduce this even further if you lost the crank signal at high RPM.

7.3.7 Knock sensor settings

Note: While this is a fairly good setup as far as knock sensing goes in a race ECU, getting OEM like levels of knock protection generally requires grenading a few motors on the dyno to get it dialed in perfectly. Knock sensing can provide an extra layer of protection, but never use knock sensing as your only spark tuning tool!

Spark knock is the sound of abnormal combustion in an engine. Once combustion in a spark-ignition internal combustion engine is initiated by a spark, the flame front is designed to spread from the spark plug and travel across the combustion chamber rapidly and smoothly. As the flame front propagates across the chamber, the remaining unburnt air-fuel mixture can ignite spontaneously (auto-ignites) before the flame front arrives, due to the increasing pressure and temperature in the combustion chamber. When this occurs, there is a sudden jump in the pressure in the cylinder. This causes in the characteristic knocking or pinging sound. It is most common at low-mid rpm and high load, such as ascending a hill in too high a gear.

Prolonged heavy knock is likely to cause severe and permanent engine damage and must be avoided.

It is a common misconception that engines make most torque just before knock. In reality, there isn't much of a connection between the knock threshold and the timing that makes best torque. A knock resistant engine may start losing power well before the onset of knock, while a knock limited engine may have the point where it makes best power past the knock threshold (and not safe to reach without higher octane fuel). Ideally, set timing on a dyno to achieve maximum brake torque (MBT) timing. Even with a well tuned engine, factors such as fuel octane, intake air temperature, coolant temperature, engine age and condition, air/fuel ratios, air density, altitude and humidity and others can push the engine from a safe condition to borderline knock or worse. The knock control system is a safety measure designed to retard timing under these conditions and safeguard the engine.

External modules use the digital input wire(s).

There are two main types of knock sensor: Resonant and wideband sensors. The resonant sensor is tuned to a particular frequency and is only likely to work on an engine with a very similar bore size. (Bore size determines "ping" frequency.) The wideband sensors give a lower voltage output but operate over a wider frequency range, and can usually be applied to different engines.

The GM resonant knock sensor pictured above is tuned to a specify frequency, like a tuning fork. When this frequency is applied to the sensor (through its connection to the engine), a piezoelectric crystal inside the sensor generates a small voltage (~1 volt), much like a microphone. As an example, some Corvette knock sensors (GM PN 1997562, 1997699, or Standard Motor Products KS45, KS46, KS49, or KS117) have a design frequency of 5200 Hz, and they produce a signal between 4800 Hz and 5600 Hz.

Most Bosch sensors with a hole through the center are wideband knock sensors, adaptable to many engine types.

The sensor should be mounted near the top of the engine block, as close to the center as practical. Do not mount it close to noisy components such as the fuel pump or cam shaft lifters. Mounting the knock sensor in the cylinder head is not a good idea because of valve train noise. Finding a suitable location of the sensor is crucial. Wherever possible, use the location specified by the manufacturer for that engine family.

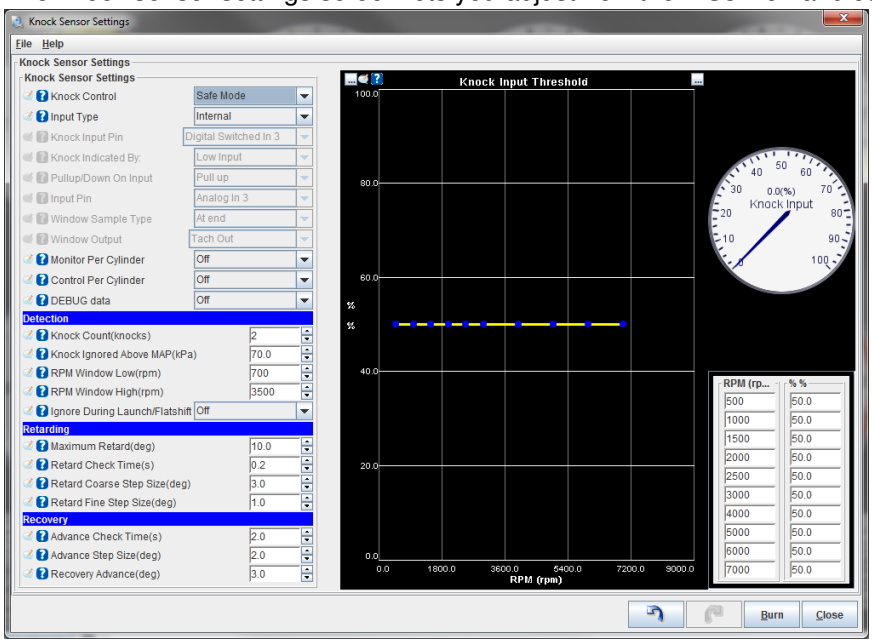
Ideally, you will be able to find a suitable threaded hole in your block to which you can mount the sensor. If not, an alternative is to drill and tap the block, or thread a steel adapter to accommodate the sensor on one end and a stub with the thread to match those in an existing pre-tapped boss in your block. Note that it may be necessary to change the sensor location if you cannot isolate engine noise while allowing MS3Pro to identify knock.

If you choose to drill and tap your block, choose a thick area of the block with a boss that is at least ¾" (19 mm) thick. Drill a ½" (13mm) hole. The hole should be 0.500" to 0.625" (13 mm to 16 mm) deep. Make absolutely sure that it is safe to drill a hole this size - **a poorly placed hole can trash the engine block.**

The GM knock sensors have a 3/8" NPT thread. Tap the hole with a 9/16" UNF starter tap. Go in 4 turns of the tap to begin with, clean out the chips and try the sensor for fit. Keep tapping one turn at a time until the sensor threads in 4 to 5 turns with hand pressure. Stop tapping when the sensor will screw into the hole 6 to 7 threads with a wrench. Note that the thread on the knock sensor is a tapered thread.

The Bosch sensor can be more simply attached by bolting it to the block. Two wire versions usually use a standard EV1 style injector connector.

The knock sensor settings screen lets you adjust how the MS3Pro handles knock input.



7.3.7.1 Knock sensor settings

- **Knock control:** Allows you to change the way the MS3Pro applies timing retard.
  - Disabled: do not use knock feedback for ignition advance control
  - Safe Mode: use knock retard, but keep the advance below that which caused knock. This backs the advance 1 small step back and leaves it at that until TPS or MAP changes - or knock comes back. This is "safe mode" scheme is the safest thing for a DIY set-up.
  - Aggressive Mode: use knock retard, but keep advance at threshold of knock occurring. That is, the program advances (up to the timing table value) if it doesn't see knock, and retards if does see knock. The difference from safe mode is that the timing can be advance all the way to the table value after knock, not just up to one step below knock. This may result in the knock returning, in which case the timing is retarded again, then advanced slowly, and so on. This approach is also called closed loop knock control.
- **Input type:**

  - On/Off: basic on/off "knock" or "no-knock" external input.
  - Analogue: external analogue input proportional to knock signal level
  -

- **Input pin** (only for on/off) Which input pin the external module is connected to. **Note: If using the Knock In 1 or Knock In 2 pins, set the input type to “internal” and this will automatically select the dedicated knock pins.**
- **Knock Indicated By** (only for On/off) The logic input level. i.e. when set to "low" then a 0V signal means there is knock. GM external knock modules fall into this categories.
- **Pull-up/down on input** (only for On/off) Whether an internal pull-resistor should be applied. When using the normal digital input modes, this should be set to “None.”
- **Input pin** (only for Analogue) Which analogue input pin the external module is connected to.
- **Window Sample Type** (only for Analogue) Either read single analogue value at end knock window or peak detect.
- **Window Output** (only for Analogue) Optional digital output for knock window signal.
- **Monitor Per Cylinder** Enables using different knock thresholds for each cylinder. This requires sequential fueling and/or coil on plug mode to be enabled.
- **Control Per Cylinder** This allows knock retard to only be applied to the cylinder that is knocking. If turned off, knock control is applied to all cylinders.
- **DEBUG data** Adds per-cylinder knock measurements to datalogging. Note that this will interfere with data logging analog sensor measurements - it uses these fields to log the per-cylinder data.

7.3.7.2 Detection

- **Knock Count** (knocks): number of knock events detected before control is started
- **Knock Ignored above MAP** (kPa): no knock retard is implemented above this MAP
- **RPM window low** (rpm): knock control is implemented above this rpm
- **RPM window high** (rpm): knock control is implemented below this rpm. This upper limit is desirable as valve train noise will typically mask out the knock sensor signal at higher rpm which could cause false reports of knock.
- **Ignore During Launch** / **Flat shift**: Flat shift or launch rev limiting with a spark cut can cause enough noise to cause false knock readings on some engines. This setting disables detection when the launch or flat shift rev limiter is active.

7.3.7.3 Retarding

- **Maximum Retard** (deg): maximum total retard when knock occurs. This can be useful to prevent timing from being excessively retarded (avoiding potential overheating issues) if the senor malfunctions or there are other problems with the knock sensing system.
- **Retard Check Time** (sec): this is the time between knock retard corrections, allows short time step to quickly retard.
- **Retard Coarse Step Size** (deg): ignition retard step size when 1st knock, make it large to quickly retard the timing and stop knock.
- **Retard Fine Step Size** (deg): ignition retard step size when knock restarts after it had stopped.

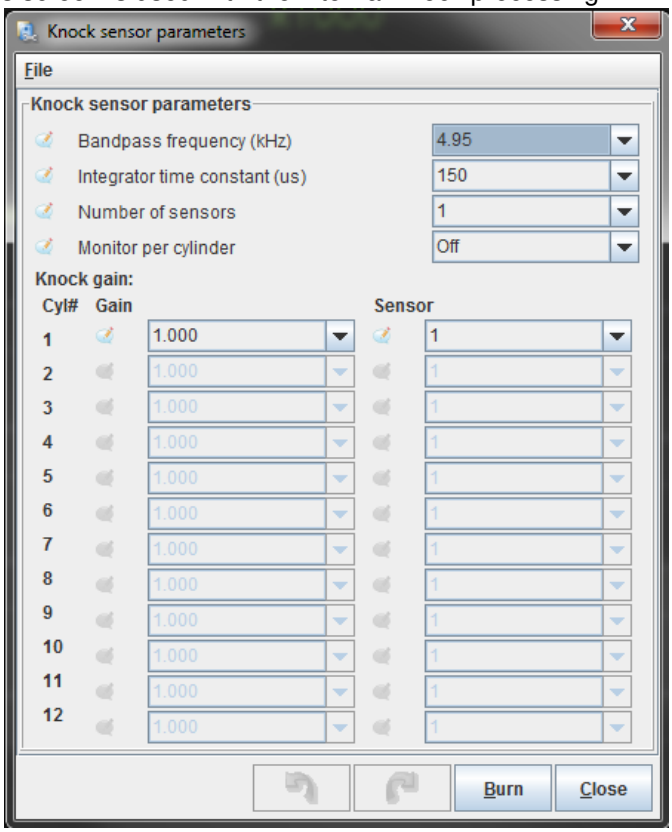
7.3.7.4 Recovery

- **Advance Check Time** (sec): this is the time between knock advance corrections (I.e., timing return to "normal")
- **Advance Step Size** (deg): ignition advance step size after knock has stopped.
- **Recovery Advance** (deg)): this is the change in table advance required to restart advance until knock or reach table value (0 knock retard) process. This only applies in Safe Mode.

7.3.7.5 Knock input threshold For analogue and internal modes, this sets a threshold level. If the measured knock signal exceeds the threshold, then the control system is started. This allows different thresholds to be applied to match actual engine behavior. The curve will need to be set empirically by examining datalogs of knock% when operating the engine under safe conditions.

7.3.8 Knock sensor parameters

This screen is used with the internal knock processing.

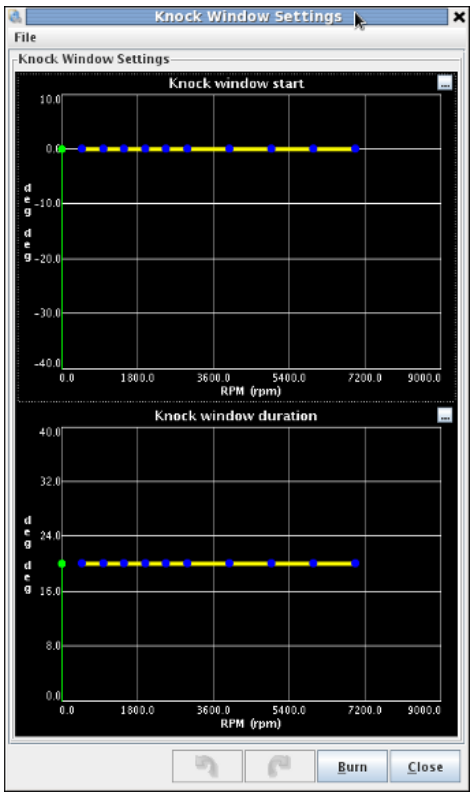


- **Bandpass frequency** - this should be adjusted to suit the resonant frequency of you engine. (Formulae exist on the web for estimating this frequency based on bore diameter.)
- **Integrator time constant** - internal setting to knock sensor chip. Default is 150us. A larger number will reduce the output level and creates more filtering.
- **Number of sensors** - how many knock sensors are connected (1 or 2.)
- **Monitor per cylinder** - when sequential fuel or spark are in operation the code can determine which cylinder the knock signal applies to. This enables the individual cylinder data to be recorded.
- **Gain** - compensate for sensor sensitivity and distance between cylinder and sensor.
- **Sensor** - pick which sensor to use for each cylinder.

Some engines have multiple knock sensors. In this case, you will usually want to pair the cylinder with whichever sensor is nearest. For example, LS1 style engines use a pair of sensors in the valley which are positioned to "listen" to four equally spaced cylinders each. i.e. the front sensor corresponds to cylinders 1,2,3,4 and the rear sensor to cylinders 5,6,7,8.

7.3.9 Knock window settings

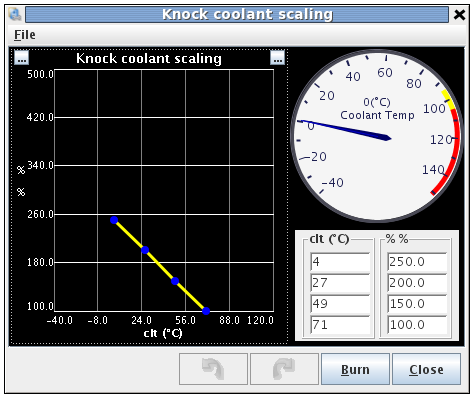
These are used with both internal and analog knock sensing modes.



The upper chart sets the crank angle to start the knock window on the cylinder being monitored. Positive is before TDC, negative is after TDC. The lower chart sets how many crank degrees the knock window stays open for. The defaults reflect that knock events start around top dead center on most engines.

7.3.10 Knock coolant scaling

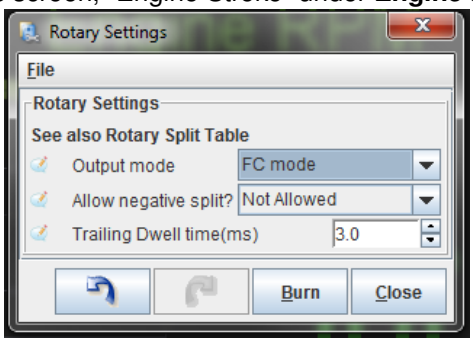
This screen is used with both internal and analog knock sensing modes.



This allows increasing the knock threshold at low coolant temperatures. Many engines are noisy during cold operating due to increased clearances. This would result in false knock reports if the threshold was not raised at low temperatures. Ensure that the final row is 100% for a warmed up engine.

7.3.11 Rotary settings

The MS3Pro supports Wankel rotary engines with a leading and trailing spark plug split. Versions supported include Mazda's RX-7 FC, FD, and RX8 ignition types, as well as 3 and 4 rotor engines with coil on plug. To enable this screen, "Engine Stroke" under **Engine and Sequential Settings** must be set to Rotary.



The "Allow negative split" setting lets you fire the trailing spark plugs before the leading (positive split numbers have the trailing spark plugs fired after the leading). You can also set a separate dwell time for the trailing coils. Note that all rotary modes have you set the number of injectors to the number of primary injectors; a 2 rotor engine would have the number set to 2, even though there are a total of four injectors, two primary and two secondary.

Here are the output modes available.

7.3.11.1 FC mode The second generation RX-7 used a logic level wasted spark coil for the leading plugs and a coil that had a "coil select" mode for the trailing coil. FC mode requires setting "Number of coils" to "wasted spark", and uses this wiring.

Ignition wiring:

- Spark A Leading coil (IGt-L)
- Spark B Trailing coil (IGt-T)
- Spark C Trailing coil select

Fuel wiring:

- Inj A Front rotor primary injector
- Inj B Rear rotor primary injector
- Inj C Front rotor secondary injector
- Inj D Rear rotor secondary injector

7.3.11.2 FD mode The third generation RX-7 retains a wasted spark coil for the leading plugs, but the trailing plugs are individual logic level coils. As with FC mode, "number of coils" needs to be set to "wasted spark."

Ignition wiring:

- Spark A Leading coil (IGt-L)
- Spark C Front trailing coil
- Spark D Rear trailing coil

Note that FD mode does not use spark B - this is done to match the RX8 mode. If you ever convert to coil on plug, this means you only have to change one wire instead of three.

Fuel wiring:

- Inj A Front rotor primary injector
- Inj B Rear rotor primary injector
- Inj C Front rotor secondary injector
- Inj D Rear rotor secondary injector

7.3.11.3 RX8 mode The RX8 mode is automatically enabled if you set the number of coils to “coil on plug” and the number of rotors to 2.

Ignition wiring:

- Spark A Front Leading coil
- Spark B Rear Leading coil
- Spark C Front trailing coil
- Spark D Rear trailing coil

Fuel wiring:

- Inj A Front rotor primary injector
- Inj B Rear rotor primary injector
- Inj C Front rotor secondary injector
- Inj D Rear rotor secondary injector

7.3.11.4 3 rotor Using three rotors requires coil on plug ignition to be set.

Ignition wiring:

- Spark A Leading coil A
- Spark B Leading coil B
- Spark C Leading coil C
- Spark D Trailing coil A
- Spark E Trailing coil B
- Spark F Trailing coil C

Fuel wiring:

- Inj A Primary injector A
- Inj B Primary injector B
- Inj C Primary injector C
- Inj D Secondary injector A
- Inj E Secondary injector B
- Inj F Secondary injector C

The coils and injectors need to be wired in firing-order sequence. The outputs fire A,B,C,D,A....

7.3.11.5 4 rotor As with 3 rotor mode, this requires enabling coil on plug ignition.

Ignition wiring:

- Spark A Leading coil A
- Spark B Leading coil B
- Spark C Leading coil C
- Spark D Leading coil D
- Spark E Trailing coil A

- Spark F Trailing coil B
- Spark G Trailing coil C
- Spark H Trailing coil D

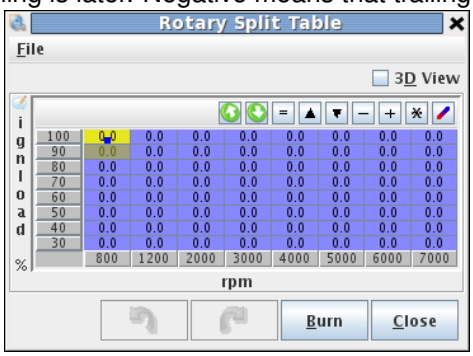
Fuel wiring:

- Inj A Primary injector A
- Inj B Primary injector B
- Inj C Primary injector C
- Inj D Primary injector D
- Inj E Secondary injector A
- Inj F Secondary injector B
- Inj G Secondary injector C
- Inj H Secondary injector D

The coils and injectors need to be wired in firing-order sequence. The outputs fire A,B,C,D,A....

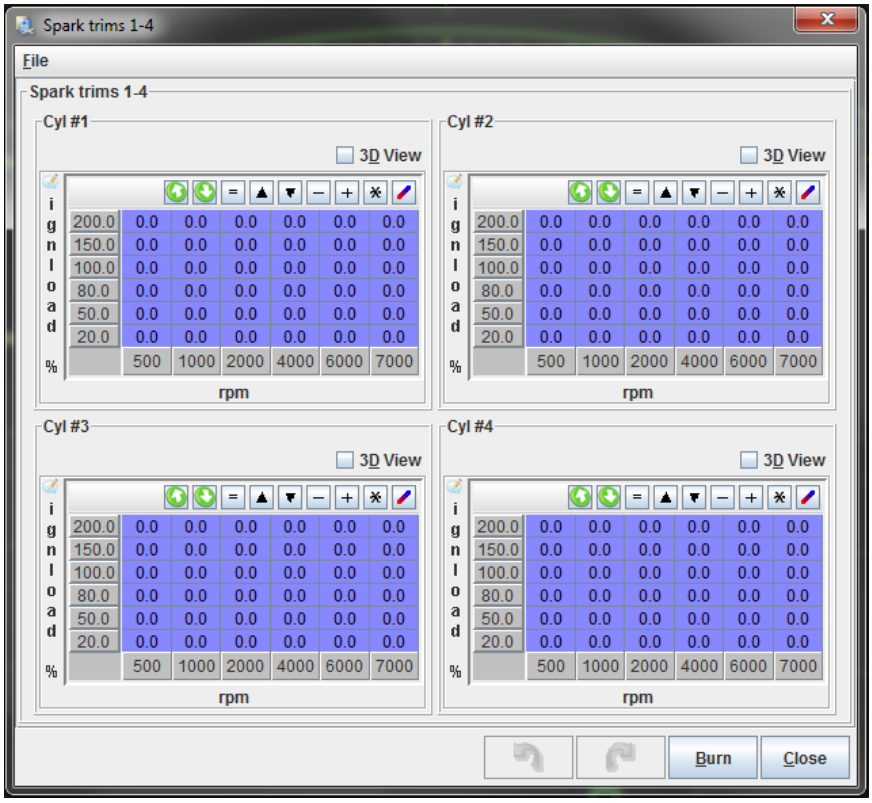
7.3.12 Rotary split table

This allows you to specify the split in degrees between leading and trailing sparks. Positive numbers mean the trailing is later. Negative means that trailing is fired before leading.



7.3.13 Cylinder trim tables

These tables are accessible if Spark Trim is set to On under Ignition Settings / Wheel Decoder.

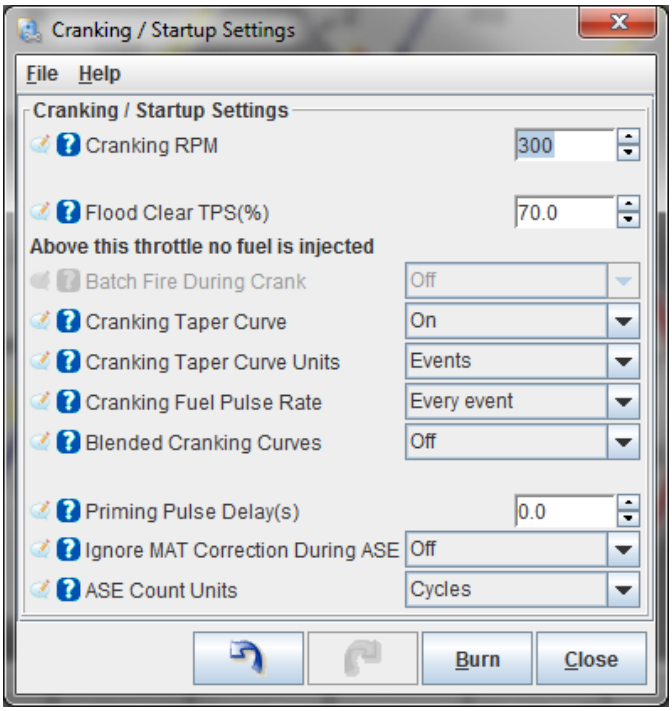


Each screen has four mini spark tables which are added to or subtracted from the main ignition table.

7.4 Startup / Idle

Section 5.4 covers the preliminary idle control setup. Section 6.3.6 covers many of the basics of the MS3/Pro's startup sequence, including the Cranking Pulse Width, Priming Pulse, After-start Enrichment, and Warmup Enrichment. Here, we'll cover the startup and idle control that was not covered there, and revisit a few settings in more detail.

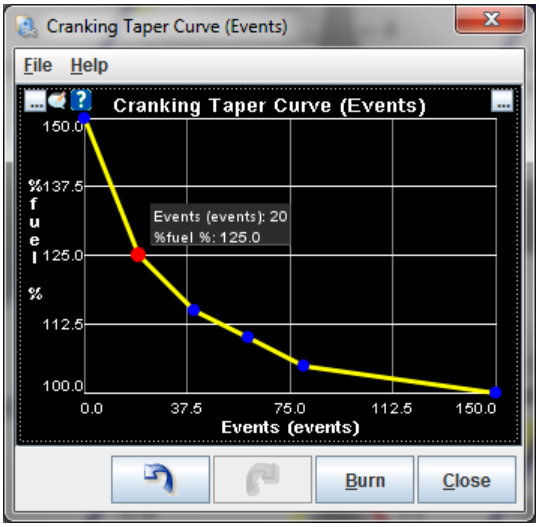
7.4.1 Cranking / startup settings



- **Cranking RPM** - On the initial start, the MS3Pro will run in batch fire on the cranking pulse width settings until RPM rises above this threshold, or if the RPM falls below this threshold within 5 seconds of starting. Set this value higher than the engine can reach while being just driven by the starter. The goal with this setting is to set it to a point where it will know the engine is up and running once RPM goes over this value.
- **Flood clear TPS** - If the throttle position exceeds this value below cranking RPM, the MS3Pro will shut off the injectors because it will take this as a signal to clear a flooded engine.
- **Batch fire during crank** - If set to on, the MS3Pro will fire all injectors every ignition event during cranking. If set to off, the MS3Pro will use sequential injection during cranking as soon as it has established cylinder identification. This setting is grayed out if the engine is set to batch fire as it will use batch firing on every ignition event in this case. Using batch fire during cranking can allow more fuel to be injected in cold climates - the maximum cranking pulse width is approximately 65 ms, so injecting fuel more often can allow for more fuel during startup.
- **Cranking taper curve** - This curve allows for the ECU to vary the pulse width while cranking, generally used to allow a large initial fuel shot that tapers off. This can make for easier startup but is a bit more complex to tune.
- **Cranking taper curve units** - The taper curve can either use time or ignition events to change how it adjusts fuel.
- **Cranking fuel pulse rate** - When using batch fired cranking, you can inject fuel every ignition event or every other ignition event. Every other ignition event can give better hot starting behavior on engines with more than 8 cylinders.
- **Blended cranking curves** - Allows using a blend curve to blend between two cranking pulse width tables. This is most commonly used with flex fuel engines.
- **Priming pulse delay** - Allows you to delay the priming pulse instead of firing it as soon as the key is turned on.
- **Ignore MAT correction during ASE** - This setting allows the MS3Pro to override the air temperature correction while after-start enrichment is active. If this is turned on, MAT correction will be set to 100% until the ASE time is complete. This is useful if the IAT sensor is in the intake manifold and absorbs heat from the manifold.

- **ASE count units** - You may set the after-start enrichment to be based on either time or engine cycles.

7.4.2 Cranking Taper Curve

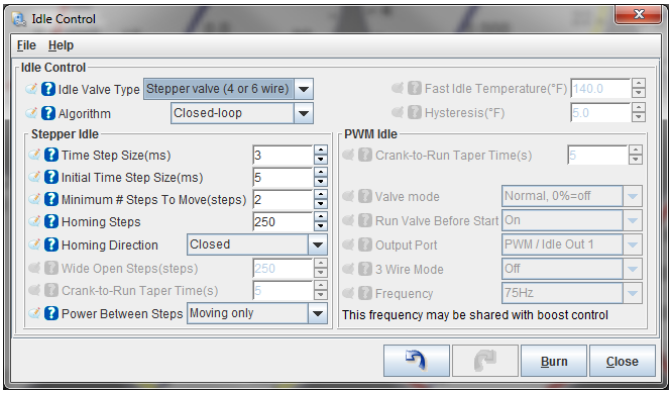


This curve provides a multiplier for the pulse width during cranking, with 100% being to use the cranking pulse width curve unchanged. The curve will repeat itself after reaching the end point. On some engines, an initial spike of fuel followed by pulling fuel can lead to better startup performance.

7.4.3 Priming Pulse 2 through Warmup Enrichment 2

These are used in dual fuel mode or a blend curve (see Table Choices) if running on an alternate fuel, as different fuels have different behavior when cold. An extreme case is CNG (compressed natural gas), which has almost no tendency to stick to the intake walls even on the coldest engine. This means an engine running CNG will require very little warmup or after-start enrichment. E85 often requires different cranking pulses from gasoline as well, both because it has a lower air/fuel ratio and because it is more likely to stick to a cold intake manifold.

7.4.4 Idle control



The MS3Pro firmware has several methods for controlling idle speed, besides the “not run any sort of ECU control, and just use a set screw to hold the throttle open” method. They were introduced in section 5.4, but we will cover them again in more detail.

The current firmware allows you to specify the valve type separately from open vs closed loop operation.

- **Idle valve type**
  - None: The MS3Pro does not control idle speed.
  - On/Off Valve: The valve is a two wire solenoid that can only turn on or off with no in between positions.
  - PWM valve: A solenoid type valve where the amount of opening is controlled by the output duty cycle.

  - Stepper valve: This uses a rotating motor to move a pintle. The motor takes a discrete step every time one of the wires is pulsed.
  - Drive by wire: Allows idle control through an external drive by wire controller connected over CAN.

- **Algorithm**
  - Open-loop (warmup): The MS3Pro commands how far to open the valve as a function of coolant temperature. Note that on/off valves are always open loop, so this setting is grayed out when an on/off valve is selected.
  - Closed-loop: The MS3Pro adjusts the idle valve opening to match an RPM target.

The different valve types use different settings, described in detail below.

7.4.4.1 On/Off Valve

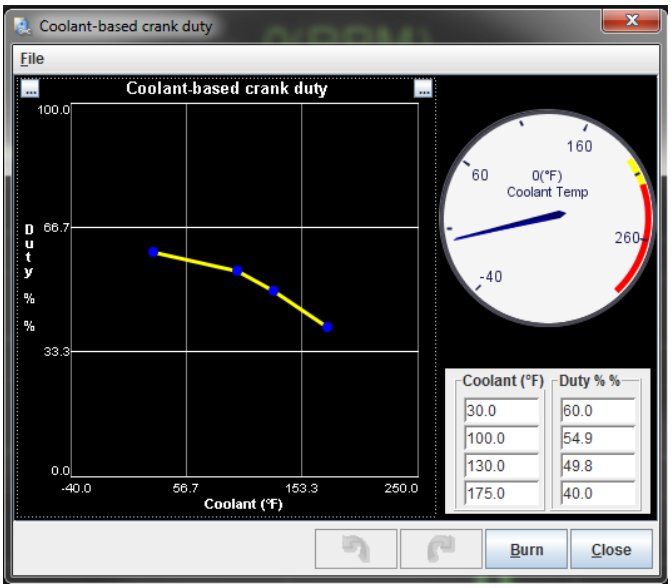
- **Fast Idle Temperature** - The MS3Pro will energize the on/off valve until the engine reaches this coolant temperature.
- **Hysteresis** - The amount of temperature drop below the fast idle temperature needed for the on/off valve to turn back on. This prevents the valve from toggling on and off rapidly when the engine is right at the fast idle temperature.

7.4.4.2 Stepper valve settings

7.4.4.3 PWM idle valve settings

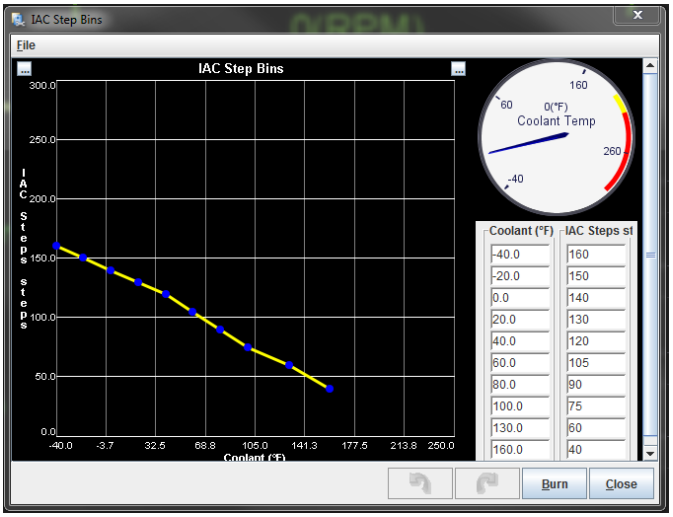
- **Crank-to-Run Taper Time** - This is the amount of time the IAC valve uses to transition after starting from its crank position to its running position.
- **Valve Mode** - Normal mode is off at 0% duty cycle, inverted mode is off at 100% duty cycle. As the name implies, most valves use normal mode. If you've found you accidentally wired up a 3 wire IAC valve backwards, just flip this setting.
- **Run valve before start** - If set to on, the valve is energized with the key on and engine off. This allows for slightly faster starting, but some PWM valves may make an objectionable amount of noise before starting. Turning this off will prevent the valve from making noise with the engine off, but may produce a small delay in starting.
- **Output Port** - Allows you to pick which output wires are used. In 3 wire mode, this is for the coil that opens the valve.
- **3 wire mode** - Used with 3 wire rotary IAC valves, common in Denso and Bosch applications. This may be left off or used to specify the connection used for the coil that closes the valves.
- **Valve Frequency Selection** - Select a frequency from the list. Different valves operate best on different frequencies.

7.4.5 Idle Cranking Duty / Steps



This screen gives how much IAC valve opening to apply when the engine is cranking as a function of temperature. A larger value means more opening. PWM valves go from zero to 100% duty, while stepper valves allow specifying how many steps to use. When using drive by wire, the number represents a percentage of the max idle addition number set under the electronic throttle control menu, with 0 being no addition and 100 being maximum addition.

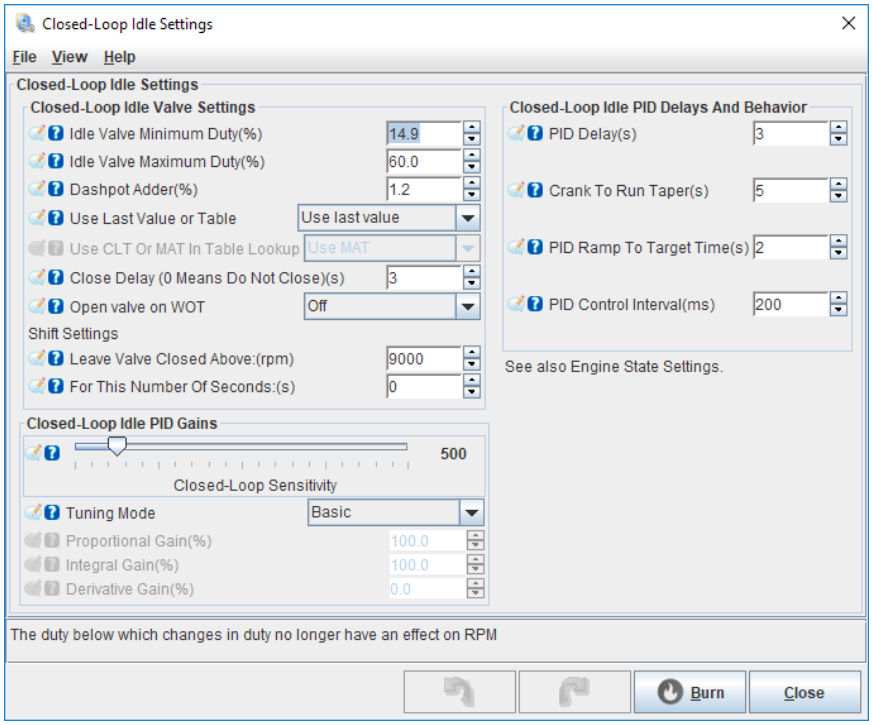
7.4.6 Idle Warmup Duty / Steps



This sets the position of the IAC valve as a function of temperature when the engine is running when using open loop mode.

7.4.7 Closed loop idle settings

Closed loop idle lets the MS3Pro target a set idle speed instead of opening the IAC valve a set amount. This is used for compensating for changing engine loads. It should not be used to fix a roving or hunting idle at steady load - you need to get the idle tuned so it is stable in warmup mode under constant load first, if you find the idle is hunting.



The MS3Pro uses the engine state settings to determine if the ECU should engage closed loop idle. The rate of RPM change needs to be below both the slow acceleration threshold and the slow deceleration threshold, and the TPS needs to be below the throttle closed TPS threshold.

The Closed Loop Idle Valve settings tell the MS3Pro what range of valve opening to use and where to set the valve when closed loop idle engages. Note that this screen will change between “steps” or “duty” depending on whether you have enabled a stepper or PWM idle control valve.

- **Idle Valve Minimum Duty** or **Minimum Steps** - This specifies the lowest duty or number of steps, below which changes in the opening position no longer affect RPM. The closed loop control will not set the IAC valve to a lower number.
- **Idle Valve Maximum Duty** or **Maximum Steps** - This specifies the highest IAC position beyond which further changes do not affect RPM. Note that intentionally setting this low can make the closed loop idle a bit less sensitive and more stable.
- **Dashpot Adder** - The MS3Pro will disengage the closed loop idle based on TPS or other criteria. Once the TPS reading falls below the value set by “throttle closed TPS threshold” in Engine State Settings, the MS3Pro will add this amount to the current IAC valve opening to prevent the RPM from falling too quickly and making the engine stall.
- **Use last value or table** - If “Use last value” is selected, the MS3Pro will start with the value it last used to obtain the target RPM when closed loop idle re-engages. If set to “use table,” when the closed loop idle engages, it will select the initial idle opening from the Closed Loop Idle Initial Value table.
- **Use CLT or MAT in table lookup** - This specifies if the initial value table uses coolant or manifold air temperature to determine the initial value.
- **Close delay** - This setting controls how long in seconds the valve takes to go to the closed position after the throttle is pressed. A setting of 0 seconds means do not close the valve.
- **Open valve on WOT** - This setting causes the idle valve to open if the TPS value goes above the wide open throttle threshold value specified in engine states. We’ve seen this produce measurable power gains on engines where the throttle body is slightly undersized and the motor needs all the air it can get.
- **Leave Valve Closed Above** (rpm) - On throttle lift, leave the idle valve closed above this RPM. Note: This setting is only necessary if a close delay is set since otherwise, the valve does not close when the throttle is pressed.

- **For this number of seconds** - This setting re-opens the idle valve after the number of seconds set on throttle lift/shift. After this timer runs out, the valve will open to the previous learned value plus the dashpot adder. This is used to allow the valve to closed while shifting gears, but to open the valve in cases where this may be necessary to prevent stalling.

Closed loop idle PID gains are used to tune the response of the valve. We'll go over the tuning process after covering the remaining settings.

- **Tuning Mode** - In Basic, the P and I terms are set to 100, the D term is set to 0, and the slider is the only adjustment. In Advanced, you can adjust the gains separately, and then they can all be scaled with the slider.
- **Proportional Gain** This setting controls the Proportional Gain of the PID algorithm. Setting this higher will result in a larger immediate response to changes in RPM. Setting this too high can result in unwanted oscillation of RPM.
- **Integral Gain** This setting controls the Integral Gain of the PID algorithm. This is the PID setting used to actually make the RPM meet the target RPM. Setting it too low will cause the RPM to never reach the target. Setting this value too high will result in unwanted oscillation of RPM.
- **Derivative Gain** This setting controls the Derivative Gain of the PID algorithm. It can be used to help dampen the effects of the Proportional and Integral settings. It is usually not necessary for good control of idle speed. It is recommended that this is left at 0%.

Closed Loop Idle PID Delays and Behavior - These settings cover more about how the engine reacts to a given amount of valve opening. The code has two adjustments, a sensitivity slider that affects response with a single adjustment, and individual PID gain adjustments for engines that need more detailed changes.

- **PID delay** - This is how long in seconds all other conditions for entering PID control must be met for before the MS3Pro will engage PID control. Set this as low as possible without being below what is normal RPMdot jitter with the engine RPM not changing. Typical values will be between 50 and 75 RPM/sec.
- **Crank to run taper** - How long after starting the MS3Pro will wait to engage PID control.
- **PID ramp to target time** - Once PID engages, the MS3Pro will gradually move the target value from the current RPM to the target speed.This can be used to help larger P-values be used, making it easier to tune PID to catch sudden drops in idle speed.
- **PID Control Interval** - This controls how often the PID control code runs. Setting this too high will make the code respond too slowly to sudden changes in load, possibly allowing the engine to stall. Setting it too low will make the engine unstable and likely to have surge issues.

7.4.7.1 Tuning closed loop idle Before trying to tune closed loop idle speed control, be sure to try tuning warmup only idle speed control. With warmup only control, a higher step-count or duty should yield higher RPM. Make sure that this is the case, and that smooth idle speed can be attained with warmup only before moving on to closed loop control.

You can also reference the support articles on our website for a step by step examples of how to tune closed loop idle with a PWM idle valve:

MS3Pro Closed Loop Idle –{} PWM Valve

and how to tune closed loop idle with a stepper motor idle valve:

MS3Pro Closed Loop Idle –{} Stepper Valve

There are two main things to tune when tuning closed-loop idle speed control:

- PID gains
- Conditions for entering PID control

We recommend tuning these in stages. For example, PID cannot be tuned if the code is never entering the PID loop. Because of this, it is a good idea to start by tuning the conditions for entering PID control, then come back and tune the PID gains.

This includes both closed loop settings and engine state settings:

- VSS threshold - Using VSS in engine states will make it so that the PID loop activates after the vehicle speed drops below the VSS threshold. This can prevent closed loop idle from activating while decelerating if using an automatic transmission that freewheels on deceleration or if coasting with your foot on the clutch.
- Throttle closed TPS threshold - Set as low as possible. If the TPS has a bit of noise, set it to around 1%; otherwise set it to 0.3%-0.5%.
- PID delay - This should be set so that the RPM dropping on throttle lift can come to a rest slightly higher than the target RPM, and become stable there. Between three and five seconds normally works the best.
- Crank to Run Taper - This setting controls how long after starting the code will delay before entering PID. Between three and five seconds works well for this setting.
- Slow deceleration threshold - This are the first of the PID lockout detection settings. Use this setting so that the code can tell the difference between decelerating with closed throttle (engine braking) and sitting at one RPM. This should be lower than the rate of deceleration in top gear.
- Overrun MAP threshold - This is the second PID lockout detection setting. The code assumes that if MAP is lower than this setting, the driver must be decelerating, and not "locked out" of the PID loop. Set this to a value lower than the MAP reading at idle.
- Slow acceleration threshold - A good value for this setting will typically be in the 200-400 RPM/sec range. If the engine speed suddenly accelerates with no throttle input (like if the clutch is engaged while the car is rolling and in gear), it must accelerate at a rate greater than this setting before the PID code will be disengaged. Setting this value too high can lead to stalls after engaging the clutch in this manner.

To tell whether the code is entering PID idle control, use the "CL Idle" indicator in TunerStudio. If the current gauge cluster in TunerStudio does not include this indicator, temporarily switch to a cluster that does.

Most modern OEM cars enter idle speed regulation in a very similar manner. The MS3 idle speed control algorithm was emulates this behavior. The sequence of events that the code was designed to follow are listed below:

1. Throttle Lift - On throttle lift, the code opens the valve to the value learned in the last iteration of the PID loop (or the initial value table) + the dashpot adder. The logic here is that the last learned value should result in an RPM close to the target RPM. The dashpot adder is added so that when RPM settles, it settles to an RPM slightly higher than the target. This is in case the air conditioning was turned on or IAT increased or anything else that might make RPM lower than the last time the PID code ran.
2. RPM settles - After throttle lift, eventually the clutch is pushed in and RPM drops to wherever it will settle given the learned value + the dashpot adder. Hopefully, the idle has settled to an RPM that is less than the commanded target + the Idle Activation RPM adder. If so, then the code will wait for the amount of time specified by the PID delay, and then enter PID control. If RPM settles above the commanded target + Idle Activation RPM adder, the code then starts checking the PID lockout detection conditions. Assuming those conditions are met, the code will still enter the PID loop after the amount of time specified by the PID delay.
3. PID control activates, RPM starts dropping to target - After the PID delay expires, the PID code will be activated. RPM will slowly drop to the target over the number of seconds specified by the PID ramp to target time.
4. Normal idle speed reached - RPM reaches the commanded target. PID continues regulating RPM until the throttle is pressed.

Once the code is reliably entering PID on every throttle lift, it is time to actually tune the PID code to reach and hold the RPM target. The settings that are associated with or affect the operation of the PID algorithm are listed below:

- Idle Open Duty/steps and Idle Valve closed duty/steps - These should be set to the minimum and maximum values that should be used during PID loop and driving operation. In addition, having these set further apart results in the PID loop being more sensitive (making changes to the output given much smaller changes in input).

- Min duty/steps for PID - This is the lowest duty/number of steps that the PID loop is allowed to command. Set this low enough to result in an RPM slightly lower than the lowest target RPM.
- RPM with valve open/closed - These should be set to the RPM with the valve closed and the RPM with the valve opened respectively. If using these settings makes the code unresponsive to changes in idle speed, the upper RPM value can be set lower.
- PID Control Interval - This controls how often the PID code runs. Setting this too high can result in sluggish response to sudden changes in load, such as the Air Conditioning being turned on. Setting it too low can result in the loop being overly sensitive to RPM changes. Typically 100ms works well.
- PID controller gains - These control the actual response of the code to changes in RPM, as well as how well the code will reach the target. Tips for tuning these are listed below.

In Basic mode, simply adjust the slider, starting with a low value and increasing it until the engine can react fast enough to any load placed on it to avoid stalling or an unacceptable RPM dip.

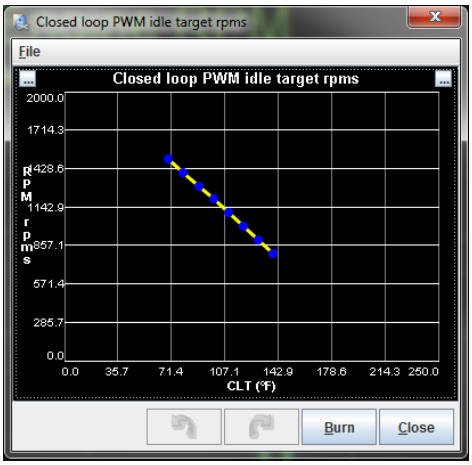
The following basic steps should be used for tuning the PID controller gains in advanced mode:

1. Zero all the gains - Set all the gains to 0%. This is so that the effects of tuning the I-term in the next step are not confused with the effects of any other setting.
2. Tune the Integral (I) gain - The Integral gain is the only term that controls whether the code actually reaches its target. Higher values for Integral gain will result in the code being able to get closer to the commanded target; however, a value that is too high will result in oscillation. The easiest way to determine a good value for the I term is to keep increasing it until oscillation occurs, then slightly lower it. If this value is increased to 200% without reaching a point where oscillation occurs, then the RPM with valve opened setting can be decreased as far as necessary, and the open duty/steps setting and closed duty/steps setting can be made further apart to make the PID loop more sensitive.
3. Tune the Proportional (P) gain - After tuning the I gain so that the RPM reaches the commanded target without oscillation, the P gain can be tuned. The best way to tune this is to set it as high as possible without getting any oscillation. After setting this, try turning on the air conditioning or other accessories that normally lower RPM or increase load. When these accessories are turned on, the RPM should dip a bit, then recover (the valve position should increase significantly). Using longer PID ramp to target times can also make it so that when the PID algorithm engages, a higher P gain can be set without causing oscillation.
4. Tune the Derivative (D) gain - For most users, use of the D gain should not be necessary. It substantially dampens the response of the loop.

Some final tips:

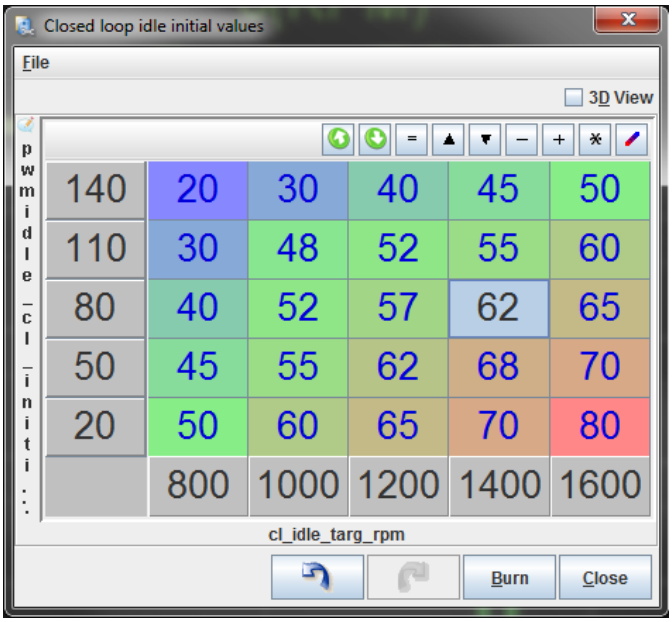
- Idle Fuel Tuning - Before even attempting to tune Closed-loop Idle speed control, tune the area around idle so that if RPM goes up or down or load goes up or down, the AFR stays close to the same value. Changing AFR can affect idle speed, which can then cause the PID code to try to correct, getting into an unrecoverable oscillation.
- Idle Advance - The idle advance feature can be used to help "catch" the idle in situations where heavy load is suddenly added while the engine is idling. It is recommended that the advance is increased with increasing load, and decreased with decreasing load. This way, when the air conditioning or electric fan are turned on, the sudden increase in load causes a corresponding increase in timing which generates more power. Also, this feature can be used so that on idle without load, slightly less than what would normally be considered "optimal" timing can be used. This causes the idle valve to need to open further to keep a particular idle speed. Then when sudden load is added, the timing increases and the valve position does not have to change as much to cope with the sudden load increase.

7.4.8 Closed loop idle target curve



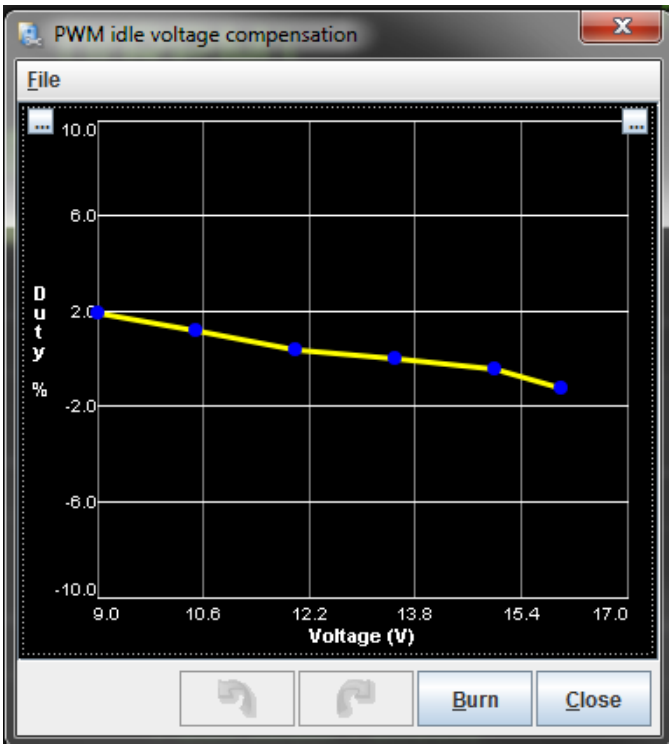
This screen is used when running closed loop idle. It lets you specify the target RPM as a function of coolant temperature.

7.4.9 Closed loop idle initial values



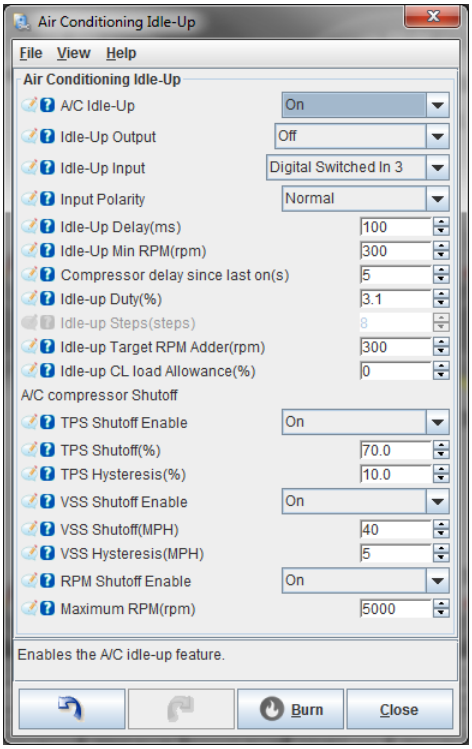
If you are using closed loop idle with the initial value table selected, the MS3Pro uses this to look up the initial duty cycle when closed loop idle control engages. The X-axis specifies what target RPM, and the Y-axis is either coolant or air temperature, as determined by the closed loop settings screen.

7.4.10 PWM Idle voltage compensation



This screen is available when using PWM warmup or closed loop idle. PWM valves, particularly the two wire sort, change how far they open when the battery voltage changes. This number is added to (or subtracted from, in the case of negative numbers) the duty cycle at the specified battery voltage. Three wire PWM valves are less likely to need this, and often need this table to be zeroed out.

7.4.11 Air conditioning idle up

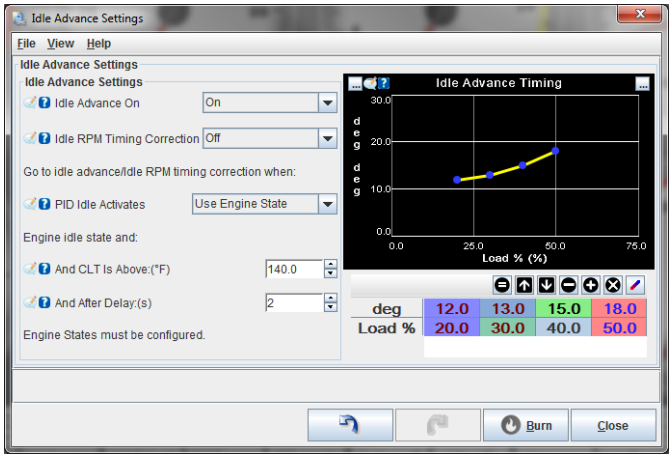


The MS3Pro can control the air conditioning and increase the idle speed when the air conditioning is active.

- **A/C idleup on** - Allows the MS3Pro to control the A/C or increase idle speed when the A/C is on.
- **Idle-up output** - This uses an output pin to activate the A/C compressor and/or condenser fan.
- **Idle-up input** - This is the input used to tell the MS3Pro that the air conditioning is on or needs to be turned on.
- **Input Polarity**
  - Normal: The A/C switch grounds the ECU input to trigger the A/C.
  - Inverted: The A/C switch sends 12 volts to the ECU to trigger the A/C.
- **Idle-up Delay** - The amount of time between receiving a signal on the idleup input and increasing the idle valve opening.
- **Idle-up min RPM** - The lowest RPM where the A/C can run. If the RPM falls below this point, the A/C output is turned off.
- **Compressor delay since last on** - The minimum amount of time between turning the idleup output off and turning it back on. Use this to prevent excessively rapid cycling of the compressor.
- **Idle-up duty** / **idle-up steps** - The amount to increase the idle valve opening when the A/C is on.
- **Idleup target RPM adder** - You can either set this to zero to make the MS3Pro hold a constant RPM when the A/C engages, or use this to increase the RPM to better drive the compressor. This feature requires closed loop idle to be on.
- **Idleup CL load allowance** - The engine may be under more load when decelerating with the A/C on. This number is added to the PID lockout max decel load when the idleup input is on.
- **TPS shutoff enable** - While most OEM ECUs shut the air conditioning off at full throttle to free up horsepower, the MS3Pro allows you to decide if you want to leave the A/C on full time, which can be useful for hot track days. Setting this to on will let the MS3Pro shut the A/C off above a set TPS value.
- **TPS shutoff %** - If using TPS shutoff, MS3Pro turns off the A/C output above this TPS opening value.
- **TPS hysteresis** - Once the TPS based A/C shutoff has engaged, the TPS needs to fall this amount below the TPS shutoff value to turn the A/C back on.
- **VSS shutoff enable** - Turning this on allows shutting off the A/C above a set speed.
- **VSS shutoff speed** - If VSS shutoff is enabled, the A/C will turn off above this speed.
- **VSS hysteresis** - Once the VSS based A/C shutoff has engaged, the speed needs to fall this amount below the VSS shutoff value to turn the A/C back on.
- **RPM shutoff enable** - allows shutting off the A/C above a maximum RPM to prevent damage to the compressor.
- **Maximum RPM** - If RPM shutoff is enabled, the A/C will turn off above this RPM.

7.4.12 Idle advance settings

The Idle Advance feature is useful to fine-tune ignition timing at idle. It is particularly useful to help catch sudden load increases on the engine at idle by increasing timing when load increases to help the engine generate more power, keeping RPM from dropping severely. Like many idle related features, this requires configuring the engine state settings.



- **Idle advance on** - Enables or disables this feature.
- **Idle RPM timing correction** - Setting this to on allows the MS3Pro to use the Idle RPM Timing Correction curve to add or subtract timing based on how far away the idle speed is from the target speed.
- **PID idle activates** - You can either trigger the idle advance settings when the conditions for closed loop idle are initially met, or after the PID delay causes the ECU to start actively correcting the idle speed. The engine state setting
- **and CLT is above (degrees)** - The engine coolant temperature must also be above this value before idle advance will engage, if using engine states.
- **and after delay (sec)** - If using engine states, all the other conditions must be met for this amount of time before idle advance will engage. If not using engine states, it will use the PID delay from the closed loop idle menu.
- **Idle Advance Timing curve** - This is a four-point curve with Load as the X-axis and timing as the Y-axis. This curve determines the actual timing once the idle advance feature has engaged.

7.4.12.1 Idle advance tuning

There are two main types of settings to tune for the Idle advance feature:

- Idle Advance engagement settings - These settings control the conditions under which Idle Advance will engage.
- Idle Advance Timing curve - This curve controls the actual ignition timing once all the Idle Advance engagement conditions have been met.

7.4.12.2 Tuning Idle Advance Engagement Settings

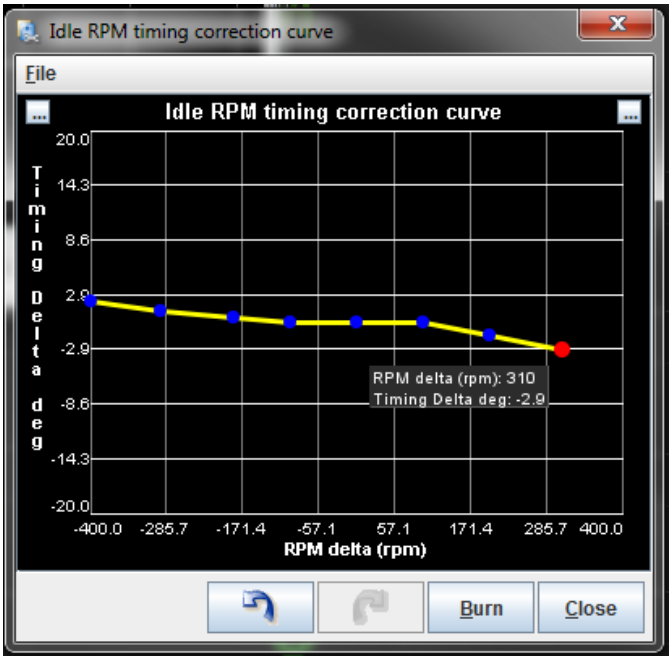
The Idle Advance engagement settings should be set so that idle advance will engage in roughly the same conditions that occur during normal, warmed-up idle.

Settings recommendations:

- Throttle closed TPS threshold (from engine state settings) - This setting should be set as low as possible. Typically settings between 0.5% and 1% should be used. If numbers that are too low are used, then idle advance may not engage if there is some play in the throttle body or there are minor electrical fluctuations that cause the closed TPS % to vary. If numbers that are too high are used, then idle advance may engage at undesirable times.
- Overrun MAP threshold (from engine state settings) - This setting should be set just below the load value seen during a normal idle with no load on the engine. It is used to prevent the idle advance from engaging while decelerating in gear.
- and CLT is above - This setting should be set to the temperature at which the engine idle characteristics no longer change. Generally this is when the engine is fully warm.
- and after delay - This setting should be set to a value that is long enough for the engine RPM and load to become stable before idle advance engages.

7.4.12.3 Tuning Idle Advance Timing In general, the most stable idle is reached by decreasing the idle timing, and increasing the amount of air entering the engine (using an idle air valve or similar). As such, the idle advance timing should be as low as possible while retaining a smooth idle. Since less timing is used during normal idle conditions, as load increase, the timing should also increase to counteract RPM decrease when the load increases.

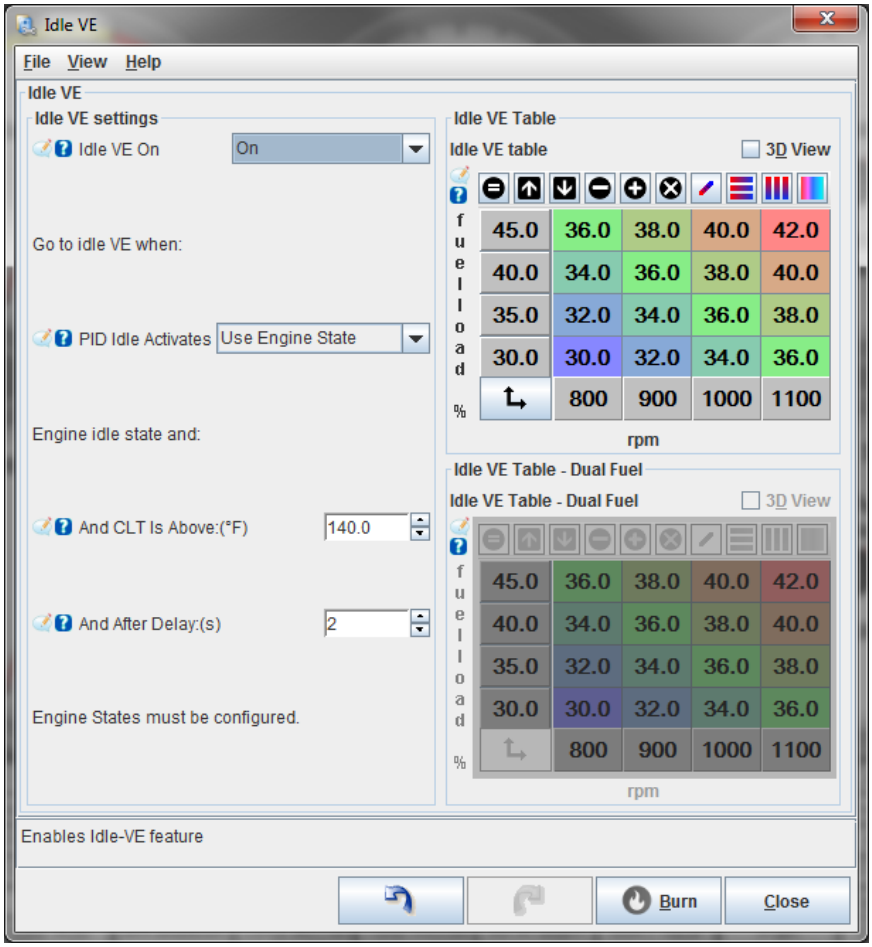
7.4.13 Idle RPM timing correction curve



This screen is available when you have enabled closed loop idle and Idle RPM Timing Correction. The RPM delta number is the difference between the target RPM and the actual RPM. This curve is added to the ignition advance, allowing you to add more timing if the RPM is below your target idle, or pull timing if the RPM goes over the target number.

7.4.14 Idle VE settings

The idle VE table is similar to the idle advance table, letting you specify a 4 x 4 VE table specifically for idle. This can avoid tying up multiple rows in the VE table strictly for idle tuning.



- **Idle VE on** - Setting this to On enables using the idle VE table, while setting this to off uses the regular VE table at idle.
- **PID idle activates** - You can either trigger the idle advance settings when the conditions for closed loop idle are initially met, or after the PID delay causes the ECU to start actively correcting the idle speed. The engine state setting
- **and CLT is above (degrees)** - The engine coolant temperature must also be above this value before idle advance will engage, if using engine states.
- **and after delay (sec)** - If using engine states, all the other conditions must be met for this amount of time before idle advance will engage. If not using engine states, it will use the PID delay from the closed loop idle menu.

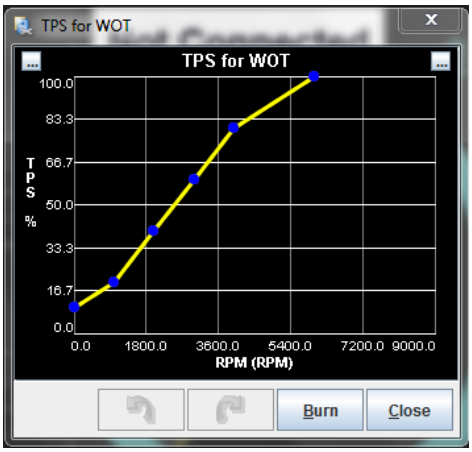
These should usually be tuned in the same manner as the settings for engaging the idle advance curve - see section 7.4.13.

The upper idle VE table is used for running off the main fuel table in place of VE table 1. The lower idle VE table is normally inactive, and is only used when using dual fuel mode with an alternate AFR table.

7.5 Accel Enrich

Section 6.3.5 covers basic acceleration enrichment tuning. Here, we'll go over more advanced ways to fine tune your transient fueling.

7.5.1 TPS WOT curve



If you have enabled “TPSdot enables %WOT” under Accel Enrich Settings, this curve defines what is considered wide open throttle for a given RPM. This curve should be set so that beyond this point, additional throttle opening has no effect on the MAP sensor reading or air flow. The acceleration enrichment will not add additional fueling once the throttle moves past the WOT level for your RPM, as the additional throttle will not make the engine need more fuel.

Setting the top of this curve below 100% will lock out acceleration enrichment at full throttle. However, setting too low a TPS percentage can make the acceleration enrichment very touchy, particularly at idle and low RPM driving. If acceleration enrichment dumps too much fuel or triggers too easily below 2000 RPM, raise the TPS percentage values at the low RPM area in this table.

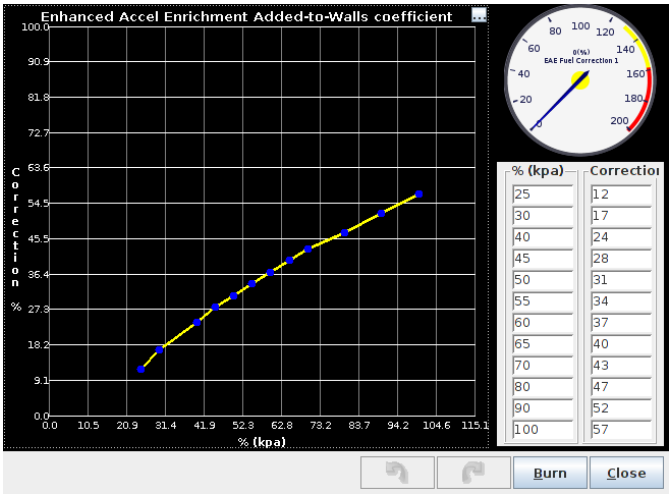
7.5.2 Enhanced acceleration enrichment

Enhanced Acceleration Enrichment (EAE) is a model base acceleration enrichment setup that uses a physics model to determine how much fuel the engine needs under changing load. This is more complicated to tune than regular acceleration enrichment, but can produce superior drivability. EAE is applied on top of regular acceleration enrichment, as the regular acceleration enrichment often reacts faster, but you’ll want to reduce the amount of regular acceleration enrichment you use when you turn EAE on.

The Enhanced Acceleration Enrichment feature is based on the concept that the fuel injected does not all enter the engine on every injector squirt. Instead, a portion of the fuel collects on the port and/or intake runner walls. The fuel collected there forms a puddle, from which some fuel enters the engine on every intake event.

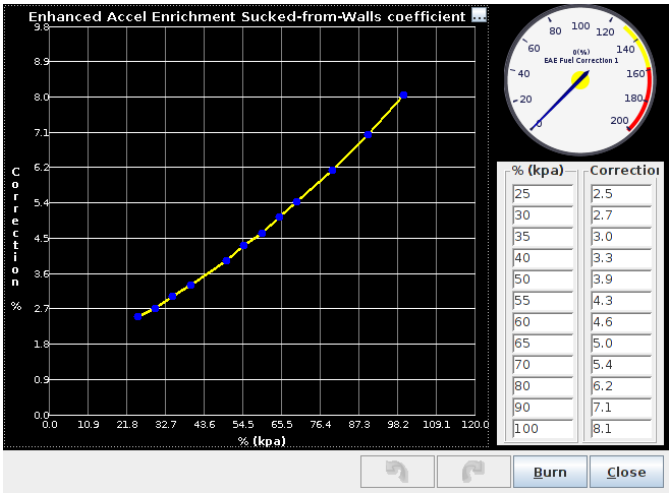
During steady-state conditions (such as cruise or idle), the amount of fuel entering the puddle, the amount of fuel leaving the puddle, and the amount of each injector squirt going directly into the engine reach a state of equilibrium. However, during throttle opening or throttle closing transient conditions, the amounts of fuel entering the puddle, leaving the puddle, and going directly into the engine change. Until equilibrium is reached again, the amount of injected fuel must change to ensure that the intended amount of fuel (from the normal fuel equations and VE table lookup) still enters the engine.

The EAE algorithm tracks the various quantities and adjusts the amount of fuel injected accordingly, but does so based on several adjustment curves and settings. The following curves are what you’ll use to tune it.



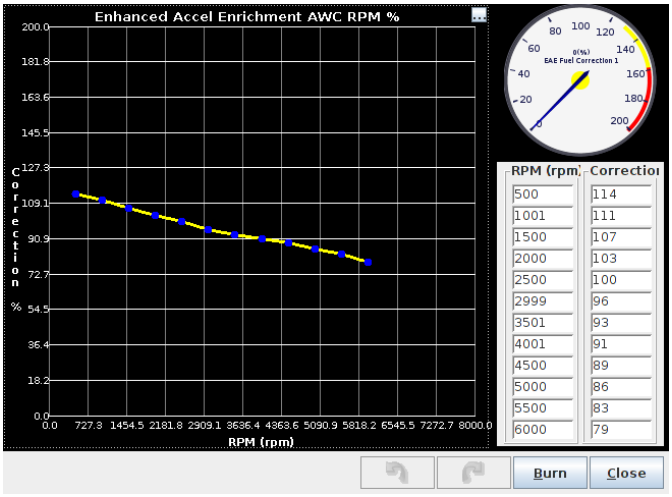
7.5.2.1 Adheres to Walls Coefficient

The X axis is the engine load. The number on the Y axis is what percent of the fuel sticks to the walls at that load level, instead of going on into the cylinders. As a general rule, the number is higher as the load goes up, and lower at low load since the vacuum helps vaporize the fuel. Engines with injectors pointed sharply downstream or aimed precisely at the intake valve tend to have lower coefficients, whereas engines with worse injector targeting can have higher numbers. An extreme example would be a carbureted manifold converted to EFI where the only available injector location has the injectors firing perpendicular to the intake runner - you're going to get a lot of your fuel puddling in the manifold.



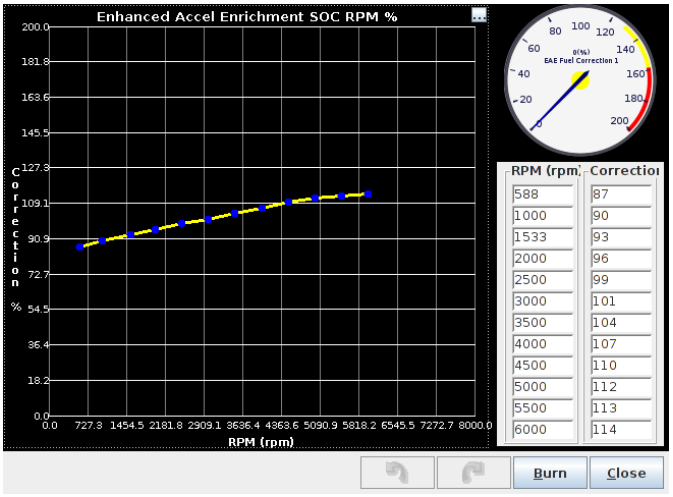
7.5.2.2 EAE Sucked-from-walls coefficient

This curve determines how much fuel is being sucked from the walls on each intake (valve open) event. It is a percentage (Y-axis) of the total amount of fuel that has adhered to the walls based on load (X-axis), so therefore the percentages are much smaller (about 10x smaller) than the Adhere-to-walls coefficients. It also typically increases with load - while the vacuum can help fuel vaporize, the added mass flow can help pull the fuel off.



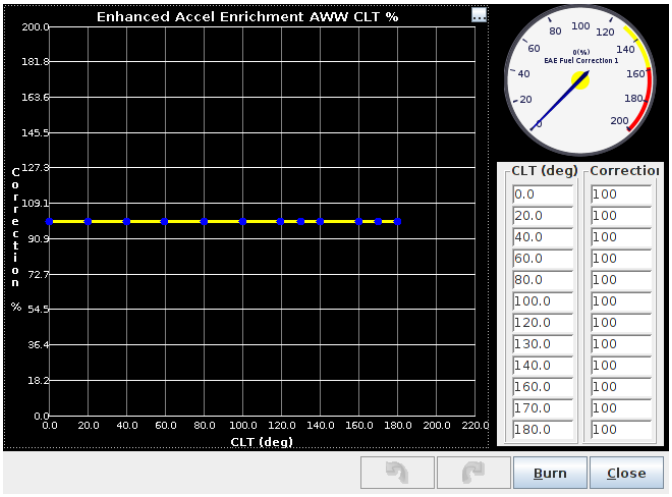
7.5.2.3 EAE Adhere-to-walls RPM correction

This curve modifies the EAE Adhere-to-walls curve. It allows the amount of correction specified by the EAE Adhere-to-walls curve to be increased or decreased based on RPM. As you may expect, the higher flow rates at higher RPM tend to carry more of the fuel into the cylinders and result in less on the walls.



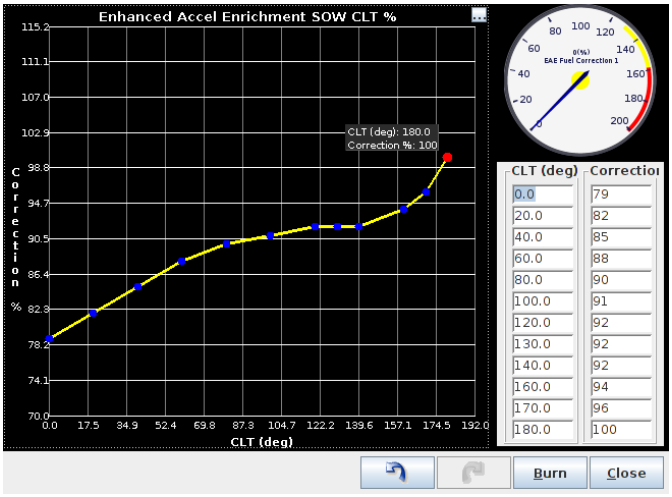
7.5.2.4 EAE Sucked-from-walls RPM correction

This curve modifies the EAE Sucked-from-walls curve. It allows the amount of correction specified by the EAE Sucked-from-walls curve to be increased or decreased based on RPM. The increased flow rate at higher RPM tends to pull more fuel off the walls as well as preventing as much of it from sticking in the first place.



7.5.2.5 EAE Adhere-to-walls CLT correction

This curve modifies the EAE Adhere-to-walls curve. It allows the amount of correction specified by the EAE Adhere-to-walls curve to be increased or decreased based on coolant temperature. The default curve is flat.



7.5.2.6 EAE Sucked-from-walls CLT correction

This curve modifies the EAE Sucked-from-walls curve. It allows the amount of correction specified by the EAE Sucked-from-walls curve to be increased or decreased based on coolant temperature. Increased temperature will make the fuel evaporate faster.

7.5.2.7 Tuning EAE Since EAE's main purpose is to ensure that the proper amount of fuel specified by the VE table (and other enrichments) actually gets into the engine, it is essential that the VE table as close as possible to correct before tuning EAE. Since EAE must track the amount of fuel collected on the port walls in order to function, it is also necessary for EAE to be enabled at all times and the tables to be tuned everywhere, including cells you rarely hit. This includes the high RPM, low load cells where you typically are running in an overrun fuel cut.

The following procedure should be followed to tune EAE:

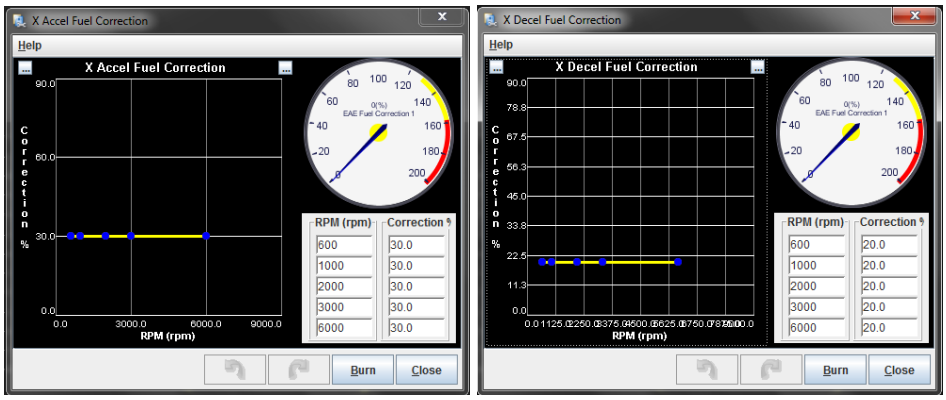
1. Tune VE, Warmup enrichments, and all other fuel-related features.
2. Turn off normal Acceleration enrichment by setting the TPSdot and MAPdot thresholds to extremely high numbers.
3. Make sure that the VE table covers all the way down to Cranking RPM and to extremely low kPa values.
4. Enable EAE in the Acceleration Enrichment dialog box.
5. With the engine at a steady RPM in a high gear, step on the throttle SLOWLY, and note the response feel and the AFR.
6. At the same RPM, lift off the throttle, and note the response feel and the AFR.
7. Tune the EAE Adhere-to-walls curve and EAE Sucked-from-walls curve until AFR and response are smooth and stable.
8. Make sure that the throttle movements used are small and slow, allowing the AFR to reach steady-state before moving the throttle again. Make sure that the whole load range is covered, and that every load seen during engine operation is covered by each of the curves.
9. Choose a few other RPM ranges, and slowly step on and release the throttle. Tune the EAE Adhere-to-walls RPM correction and EAE Sucked-from-walls RPM correction curves until the response and AFR are correct at the RPMs chosen. Typically, you should start at the idle and high cruise RPMs should be chosen. High cruise RPMs, for example are when speed is maintained but the gear selection is reduced by one or two gears.
10. Shut off the engine, and allow it to cool completely.
11. Start the engine; as the coolant temperature increases, adjust the EAE Adhere-to-walls CLT correction and EAE Sucked-from-walls CLT correction curves so that response and AFR are stable.

12. Once small, slow throttle movements are tuned, larger ones can be verified, as well as normal driving with gear shifts.
13. Finally, try to quickly blip the throttle while free-revving. If response is slower than desired, a very small amount of TPSdot or MAPdot acceleration enrichment may be re-enabled. Take care to only use it for high TPSdot values and use very little. Just enough to get EAE to respond is all that is required.

The second set of EAE curves under the main ones are for use with dual fuel mode, as alternate fuels often vaporize at different rates and need different tuning.

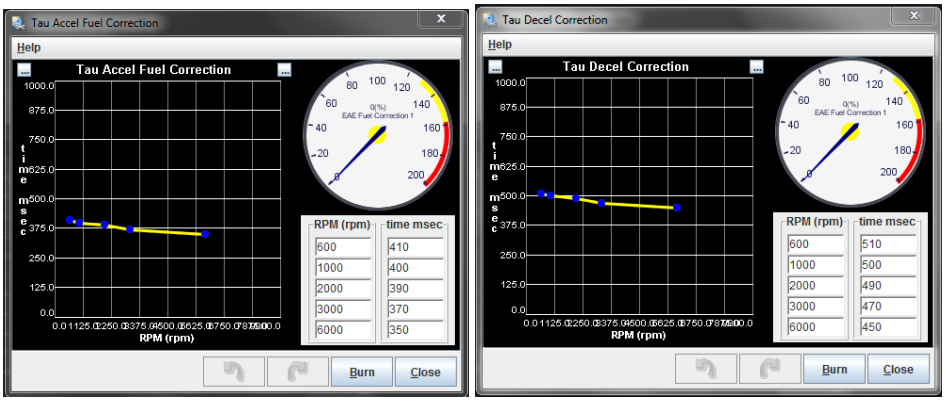
7.5.3 X-Tau acceleration enrichment

X-Tau is an alternate model based acceleration enrichment, with slightly different equations. This uses a number called X to represent the percentage of fuel sticking to intake walls, and tau to represent the amount of time it takes for the fuel that sticks to the walls to reach the cylinder. While EAE uses a load scalar curve, X-Tau uses two separate, RPM based curves for acceleration and deceleration. The X-Tau algorithm uses the rate of change of the MAP reading, not the absolute value of the MAP reading, to determine which of these two curves to use. At steady MAP, it blends between the two.



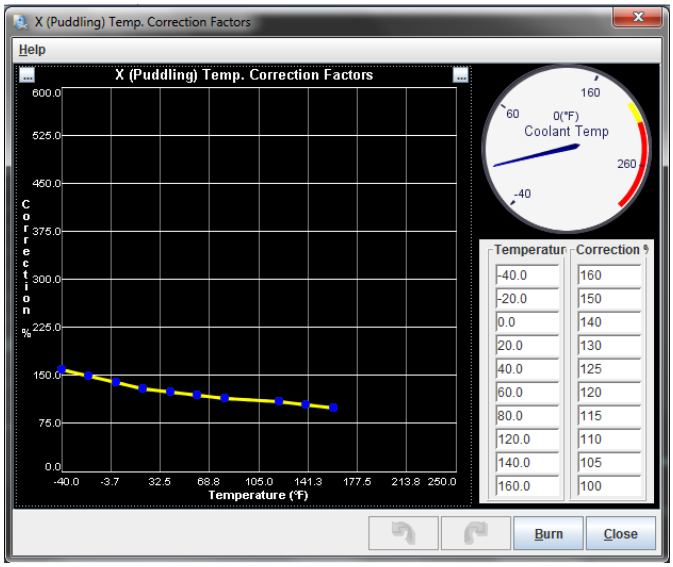
7.5.3.1 Puddling Factors

This is the X term. There are two tables, one for acceleration and one for deceleration, which represent what percent of the fuel puddles in the intake as a function of RPM. Note that in the way the X-Tau numbering works, setting either curve to values above 48% can cause the fueling to oscillate, particularly at low RPM, unless you use large tau values as well.



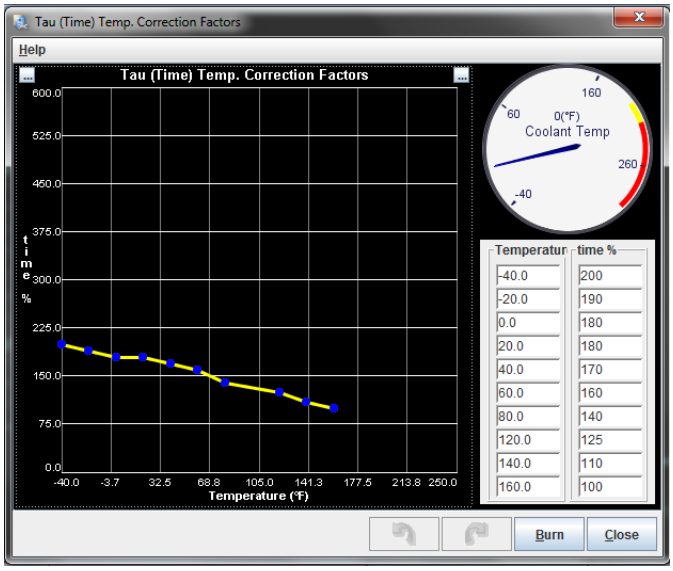
7.5.3.2 Time factors

This is the tau term. This number represents the time, in milliseconds, for the fuel to travel from the puddle to the cylinders. Valid numbers are from 10 msec to 2500 msec. The closer the fuel puddle is to the intake port and the faster the air is moving, the lower the number. A port injected engine with small intake runners and injectors targeted on the intake valves will be on the low side of the scale. A throttle body injection intake will be on the high side.



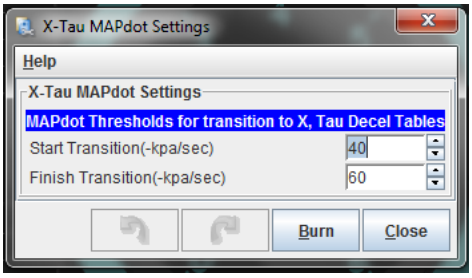
7.5.3.3 X (Puddling) Temp Correction Factors

The temperature correction factors are enabled if you enable X-Tau with CLT correction. This is a percent multiplier used to scale the accel and decel puddling factor curves as a function of coolant temperature.



7.5.3.4 Tau (Time) temp corrections

This is a percent multiplier used to scale the time factor accel and decel curves as a function of temperature.



7.5.3.5 MAP

This determines whether to use the accel or decel tables. If MAPdot is above the start transition threshold, the X-Tau accel tables are used. If MAPdot is less than (i.e., more negative than) the finish transition threshold, the the X-Tau Decel tables are used. If MAPdot is in between, a blend of the 2 tables is used. Note that these are in negative kPa (i.e., put 40 if you want the threshold to be at -40 kPa/sec).

7.5.3.6 X-Tau tuning tips You should start with low X and low Tau, such as the default values. You should then see if the X-Tau helps. Start by adjusting the X factor. If that doesn't help, try increasing the Tau time table entries in the areas where you are having troubles with lean spots (engine coughs on accel).

The Tau time table is deliberately conservative, so in most cases it will require increasing, by perhaps 50% to 100%.

You may need to try adjusting the lag filter values for the MAP sensor for less filtering. The X-Tau mode needs to see the rapid rate-of-change of MAP and the lag filters can reduce this if set too low. Generally you do not want to set it to less than 50%, and may need to set it to 70% to 80%.

As you dial-in the X-Tau parameters, the engine will likely become overly rich due to both normal and X-Tau enrichments being applied. You can try reducing the existing TPSdot or MAPdot based accel/decel enrichments, which do NOT go away when you specify the X-tau option and increasing the X-tau variables. (Note that you may have to increase the cold accel multiplier as you do this.)

7.6 Boost / VVT

The MS3Pro has multiple boost control strategies, and can control up to four channels of continuously variable valve timing.

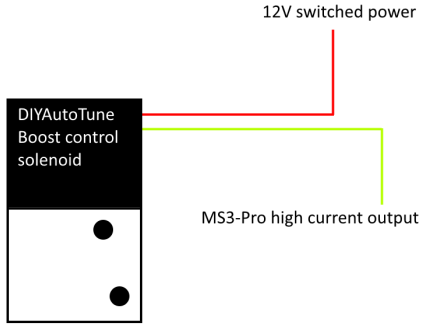
7.6.1 Boost control basics

7.6.1.1 Boost control strategies The MS3 Firmware has four primary algorithms for controlling boost:

- Open-Loop Single Solenoid - Solenoid duty comes from an 8x8 duty table of TPS vs RPM.
- Closed-Loop Single Solenoid - A Proportional-Integral-Derivative (PID) loop controls solenoid duty to match the actual boost to the boost target. Closed loop control starts by using a bias table to look up what duty cycle to start with for a particular RPM and target boost, and then uses the PID loop to apply a correction factor to hit your target boost. An 8x8 target table of TPS vs RPM or a 6-point boost vs vehicle speed curve is used to figure out the boost target.
- Open-Loop Dome Control - The MS3Pro uses a pressure sensor in the wastegate dome and a closed loop algorithm to specify a target pressure in the wastegate dome. The ECU does not attempt to target a particular MAP sensor reading. This uses a reservoir of CO_2 or compressed air, and two solenoids, one to let CO_2 into the dome and the other to vent it to the atmosphere.
- Closed-Loop Dome Control - This also uses a wastegate dome pressure sensor, CO_2, and two solenoids. However, this algorithm targets a MAP sensor reading by using a table showing what dome pressure should produce a target boost pressure at what RPM. The ECU then uses a PID loop to adjust the target dome pressure up or down if this does not hit the target boost table, while using a separate closed loop algorithm to make sure the dome pressure reaches its target.

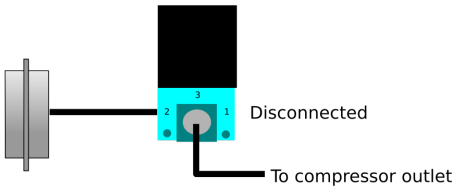
There are also several modifiers that allow you to change the boost level based on timing, gear, measured wheel speed, or other factors, on top of the primary boost control algorithm.

7.6.1.2 Boost control hardware You can wire a boost control solenoid to any unused high current or injector output, as shown below. Boost solenoids normally do not have a polarity; the two wires can be hooked up in either order.

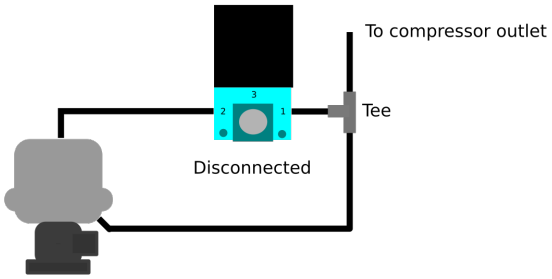


The AMP EFI boost solenoid can be used with either a one or a two port wastegate. This diagram shows how to plumb it in single solenoid mode. As a rule of thumb, the single solenoid mode can reach boost pressures approximately double the pressure allowed by the wastegate spring on its own.

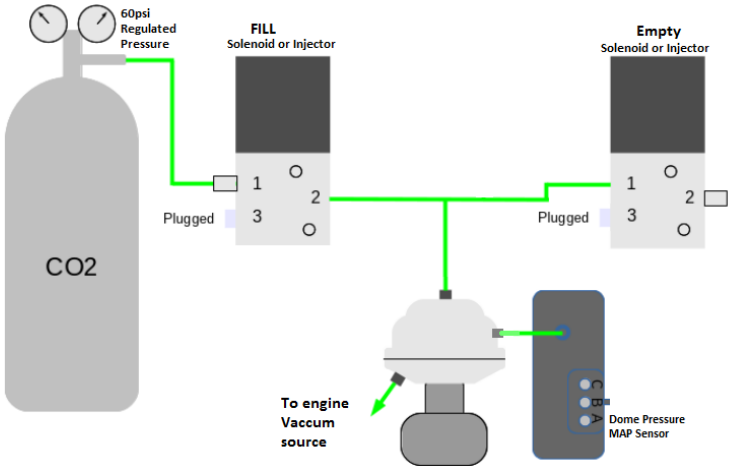
Single port wastegate



Dual port wastegate

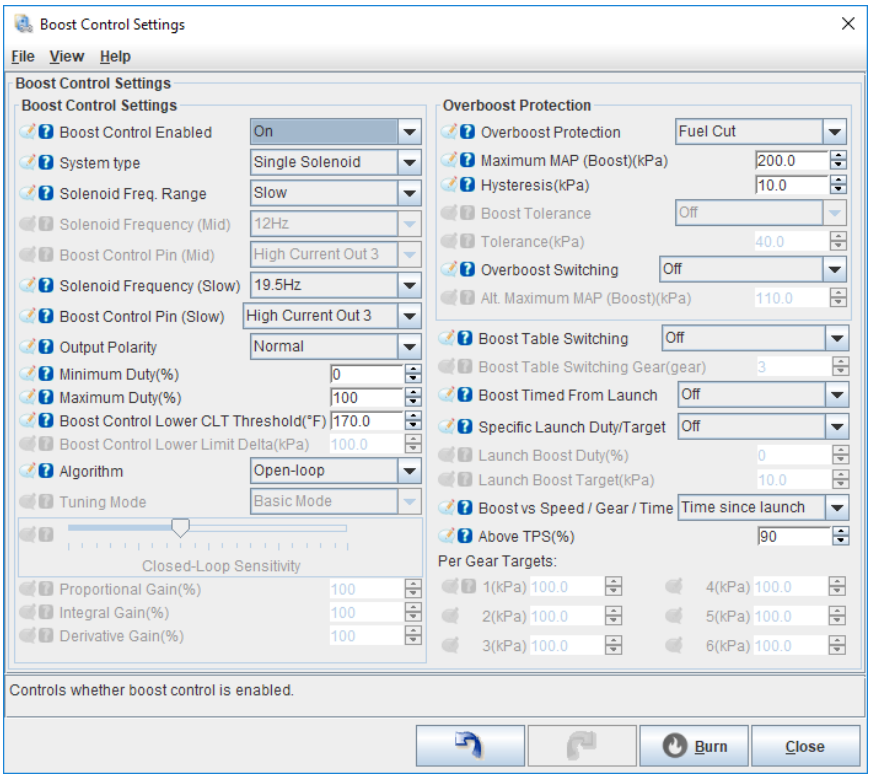


Dome pressure plumbing uses a compressed CO_2 supply to forcibly pressurize the top portion of the wastegate actuator to achieve boost levels well above spring pressure. Two three-ported solenoids are required as well as an additional pressure sensor to monitor the dome pressure. The 'vent' port on the 'fill' solenoid must be plugged. The compressed gas supply is typically CO_2 and needs to be regulated to a pressure a little above the maximum dome pressure required. For testing, compressed air works fine too.



Note that the MS3Pro will not energize the boost solenoid when the engine is in vacuum, to reduce wear on the solenoid.

7.6.2 Boost control settings



7.6.2.1 Boost control common settings This section covers settings that are used by both the open-loop and closed-loop control algorithms.

- **Boost Control Enabled** - This controls whether the boost control feature is enabled or disabled.
- **System Type** - Single solenoid uses one solenoid, plumbed to supply pressure from the turbine outlet to the wastegate. Dome control uses a CO_2 or compressed air reservoir, a sensor to monitor wastegate dome pressure, and two solenoids, one to fill the dome and one to empty it. Additional settings related to dome control will appear under Wastegate Dome Pressure Control Settings 1.
- **Solenoid Frequency Range** - This setting controls the frequency range of the valve. Mid range allows 12 to 1021 Hz, while Slow allows 11.1 to 78 Hz but allows you to use a wider variety of pins (such as more unused injector outputs) for boost control.
- **Solenoid Frequency** - Note that there are two drop down menus depending on whether you are using Mid or Slow mode. This sets how many times to pulse the boost control solenoid off or on. Many boost control solenoids work well from 19.5 to 78 Hz, depending on the valve - we typically recommend 19.5 to 39 Hz for the boost control solenoid we sell.
- **Boost Control Pin** - This sets which output pin is used to control the boost control solenoid.
- **Output Polarity** - This is used to set the polarity of the output. Normal is used when you have a conventionally set up boost control solenoid designed to go to minimum boost if it loses power as a fail-safe; with this setting, higher duty cycle means higher levels of boost. Inverted means that increasing the duty cycle decreases the boost.
- **Control Interval** - This setting is used to control how often the boost control algorithm runs.
- **Closed Duty** - This setting controls the lowest allowed reported duty. Normally, this should be 0%.
- **Open Duty** - This setting controls the highest allowed reported duty. Normally, this should be 100%.

- **Boost Control Lower CLT Threshold** - To prevent running maximum boost before the engine is warmed up, the MS3Pro will run minimum boost until you reach this temperature, at which point the boost control turns on.
- **Algorithm** - If using single solenoid mode, open loop

7.6.2.2 Closed loop specific settings Enabling the closed-loop boost control algorithm enables the following additional settings:

- **Boost control use initial value table** - If this is set to On, the boost control will start with an initial duty cycle table whenever it moves into a new load cell as a feed-forward mechanism. If it is set to off, the boost control will adjust from the previous duty cycle it was running at, and operate purely from the feedback loop.
- **Boost Control Lower Limit Delta** - This regulates how close the boost must be to your target to engage closed loop control. When the boost is more than this amount below your target, the controller holds the valve wide open to maximize spool up speed when using single solenoid control. When used with CO2 based control, the code will target the bias pressure value when boost is greater than ambient pressure but more than the boost delta below your target, and will then will then use PID control to move the dome pressure target when within the delta limit. (When MAP is below ambient pressure, the closed loop CO2 control will target ambient pressure in the dome.)
- **Tuning Mode:**
  - Setup Mode - PID correction is turned off to allow adjusting the bias tables. Set the bias tables so you reach your boost target levels.
  - Basic Mode - The PID parameters are all set to 100%, and you only tune with the slider.
  - Advanced Mode - All PID parameters are adjustable.
- **Closed Loop Sensitivity** - Use this setting to tune how aggressively the boost controller tries to reach the target boost. Adjust the slider up until the boost oscillates around the target, then back down until it stops.
- **Proportional Gain** - Proportional gain affects the strength with which changes in input immediately affect changes in output.
- **Integral Gain** - The Integral Gain setting affects the response to continued difference between the target boost and the actual boost.
- **Derivative Gain** - The Derivative Gain setting helps to slow down the response of the Proportional and Integral gain settings as the target is reached. This should be used sparingly as it can also completely dampen the other two Gains.

The first step for tuning Closed-loop boost control is to set the desired targets in the Boost Control Target table (section 7.6.3). Typically, lower throttle positions will have lower boost targets. This lets you modulate the boost with the throttle.

Once you have set the targets, set the tuning mode to Setup Mode. Then adjust the boost control duty bias table so that you hit your boost targets with the duty cycles used in this table.

Typically, the defaults for the following settings can be used:

- Closed Duty - A closed duty of 0% is the default. This should be tuned to the value that starts to open the wastegate, but typically 0% works well.
- Open Duty - An open duty value of 100% is the default. This should be tuned to the value that fully opens the wastegate, but typically 100% works well.
- Boost Control Lower Limit - This setting is used to set the pressure at which PID boost control is engaged. The default for this setting is 100. If a faster rise to target is desired, this setting can be set to a higher number, but the safest number is 100 since it gives the PID code the most time to react to climbing boost.

The next step after setting up the target table and supporting settings is to adjust the slider for best performance. If the slider on its own does not give the results you hope for, you can go to advanced mode and tune the PID gains:

1. Set Integral and Differential Gains to 0% - This will let you focus on tuning the proportional gain.
2. Set Proportional gain to 100% and slowly lower - While tuning Proportional gain, higher numbers mean slower boost climb and lower final boost. For safety, start with a very high gain (100% should be sufficient). Find the RPM that typically spools quickly, and fully and quickly depress the accelerator. Note how much boost is reached. If boost overshoots the target dramatically, increase the Proportional gain. Otherwise, reduce the Proportional gain and try again. Do this until boost reaches the target with a small amount of overshoot.
3. Tune the Integral Gain - The next step after the target is reached consistently is to tune the Integral gain. Starting from the RPM used to tune the Proportional gain, fully depress the accelerator and watch the boost as the engine climbs through the RPM range. As the engine accelerates through the rev range, the boost will probably creep away from the target. Keep increasing the Integral gain until the controller adequately maintains the target with minimal oscillation. It may be necessary to increase the Proportional gain a bit after tuning the Integral gain since the two gains tend to counteract each other.
4. Tune the Derivative Gain - Increase the Derivative gain until the overshoot is minimized. Care must be taken when increasing the Derivative gain as too much Derivative gain can over-dampen the effects of the Proportional and Integral gains.

7.6.2.3 Overboost Protection Overboost protection works similarly to a rev limiter, except that it can stop engine operation when boost exceeds a user-set limit. In addition, use of the overboost protection feature is required when using the closed-loop boost control algorithm. The following settings affect the operation of overboost protection:

- **Overboost Protection** - Activates overboost protection and lets you specify what action to take when boost exceeds the maximum boost number.
  - None - Disables overboost protection.
  - Fuel Cut - Stops the engine by cutting fuel.
  - Spark Cut - Stops the engine by cutting spark.
  - Both - Stops the engine using both fuel and spark cut.
- **Maximum Boost** - The maximum boost (in kPa) at which the engine should be operated.
- **Hysteresis** - The amount boost must drop by (in kPa) after hitting the maximum boost before fuel or spark are restored.
- **Boost Tolerance** - Turning this on in closed loop mode creates an alternate maximum boost value equal to your boost target plus the tolerance value set below. The MS3Pro will use either the maximum boost level or the number generated by the boost tolerance at your RPM and throttle, whichever number is lower. Allows the overboost protection to track with closed loop boost control for engines that can run more boost at some points of the RPM range than others.
- **Overboost switching** - Allows you to use an input to raise or lower the point at which overboost protection engages. For example, you might want one setting for pump gas and another for race gas, and can use a switch to let the MS3Pro know which you are running.
- **Alt Maximum Boost** - The boost level to allow when the overboost switch is active.

Setting this ups very similar to setting up a rev-limiter. Choose from Fuel Cut or Spark cut or Both. Do not use spark cut mode if the engine still has a catalytic converter. Spark cut can cause backfires that melt or break the catalyst.

The maximum boost should be set up higher than the maximum target boost will be in the closed-loop boost target table. Hysteresis should be set so that MAP jitter does not cause it to alternate rapidly between on and off. Remember, overboost protection is not telling the MS3Pro “I would prefer my boost level stays below this setting;” it is more like, **“Shut down my engine at once if the boost gets this high!”** And the MS3Pro will react with that same level of urgency.

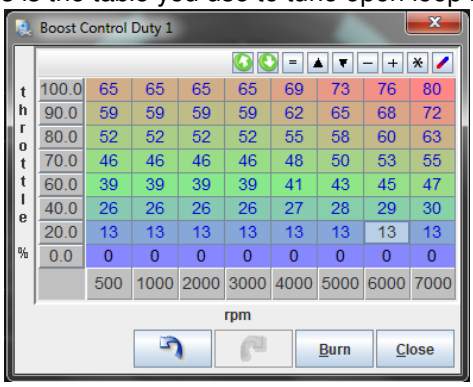
7.6.2.4 Other Boost Control Settings

The following settings remain:

- **Boost Table Switching** - This option when enabled allows a switch input to be configured which switches between two different boost duty or target tables. You can also allow it to switch tables based on gear by selecting “Gear-based”.
- **Boost Table Switching Gear** - If boost table switching is set to gear based, the MS3Pro will switch to the second boost table when you shift into this gear and above.
- **Boost Delay (Timed From Launch)** - If launch control is enabled, this setting applies a multiplier curve to the boost duty or target based on the time after launch, to ramp up boost once the launch control is released. If you would rather specify the boost duty or target directly as a function of time after launch, use the “Boost vs Speed / Gear / Time” setting instead.
- **Specific Launch Duty/Target** - If launch control is enabled, this setting controls what target (closed-loop) or duty (open-loop) is used for boost control.
- **Launch Boost Duty** - This is the duty used by Specific Launch Duty/Target, if using open loop boost control.
- **Launch boost target** - This is the boost target used by Specific Launch Duty/Target, if using closed loop boost control.
- **Boost vs speed / gear / time** - This item enables a curve that specifies boost duty or target directly as a function of VSS, gear position, or time after launch. This ignores the boost tables when active and uses its own curve, instead of modifying the tables as with the boost timed from launch option.
- **Above TPS %** - This setting is used in conjunction with the Boost vs Speed function. Below this TPS setting, the normal boost tables are used. Above this TPS setting, the boost vs speed curve is used.
- **Bias table per gear** - Enables a boost bias table per gear for single solenoid (not dome.) This could be useful if you are seeing different required solenoid duty cycles in each gear and can allow you to set the open-loop ’best guess’ bias to be closer to the most typical duty, if tuned correctly this gives the closed-loop less work to do.
- **Per Gear Targets** - Specifies the boost target in each gear if boost by gear is active.

7.6.3 Boost control duty table

This is the table you use to tune open loop boost control.

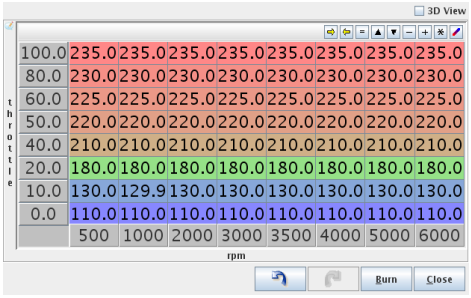


To tune this table, start with small values (the default table is all zeroes) and increase the numbers until you reach the desired amount of boost. Use open-loop boost control to try out different frequency settings and find the settings that work the best for the particular solenoid being used. We recommend using open loop boost control to establish that the boost control setup is working before you use closed loop control.

The second duty cycle table (Boost control duty table 2) can either be used for switched boost or a second wastegate.

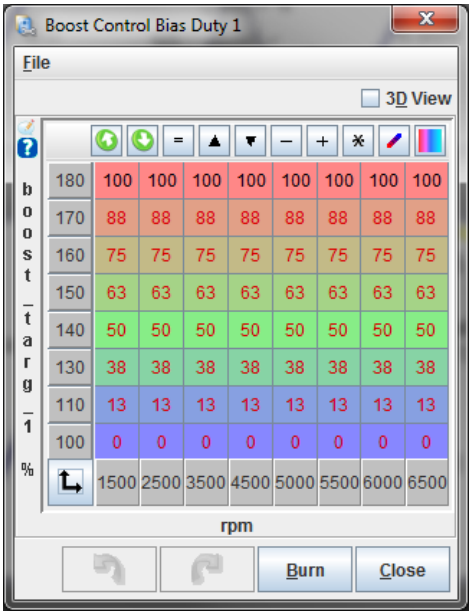
7.6.4 Boost control target table

When using closed loop control (either single solenoid or dome pressure), this table specifies what boost level to target in kPa as a function of throttle position and RPM.



The second table (boost control target table 2) can either be used for a second wastegate or for switched boost. If using open loop dome pressure control, this table is the target dome pressure.

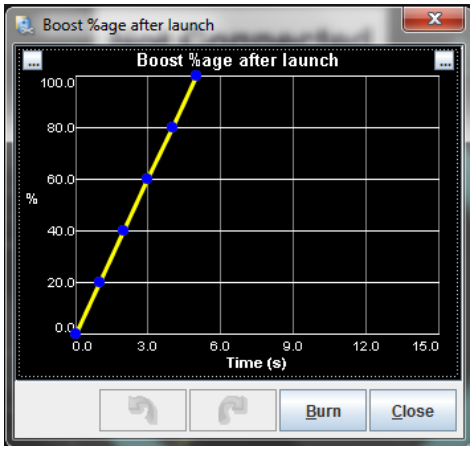
7.6.5 Boost control bias duty table



This is a feed-forward table. The MS3Pro first looks to the closed loop target table to see what boost target to use, then goes to this table to look up what duty cycle to start with at what RPM and target boost level. The X axis is the engine RPM, while the Y axis is the target boost level. As with the other tables, the second table (boost control initial duty table 2) can either be used for a second wastegate or for switched boost.

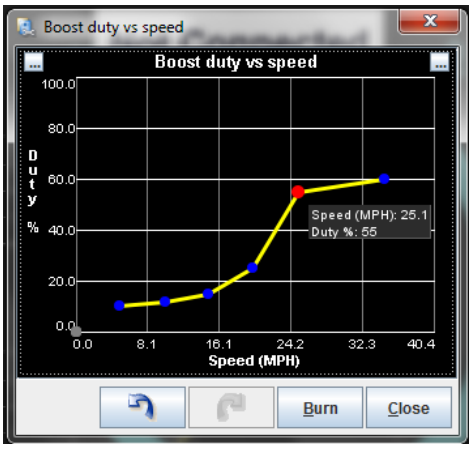
7.6.6 Boost delay

This screen is active if you enable “Boost timed from launch” and launch control.



This specifies how long to go from no boost (if the initial value is not used) or the specified initial value (if used) to full boost after you release the launch control switch. This is a percentage of the regular boost, so it should end at 100% boost.

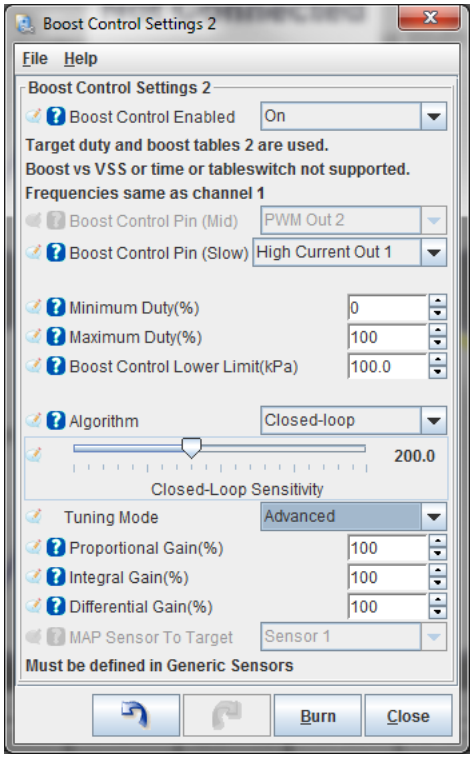
7.6.7 Boost control vs speed / time



This allows you to specify the amount of boost allowed (closed loop) or solenoid duty cycle (open loop) as a function of ground speed or time after launch.

7.6.8 Boost 2 control settings

The boost control 2 menu is for twin turbo setups, either sequential or compound. Use these settings if you have a second solenoid valve controlling a second wastegate independently of the primary wastegate. Setups that use a single solenoid to control two wastegates do not use these settings. As the screen notes, you cannot use this with timed, speed, or switched boost control. Leave this setting off if using dome control mode - this also uses two solenoids, but they both act on the same wastegate using the same tables.



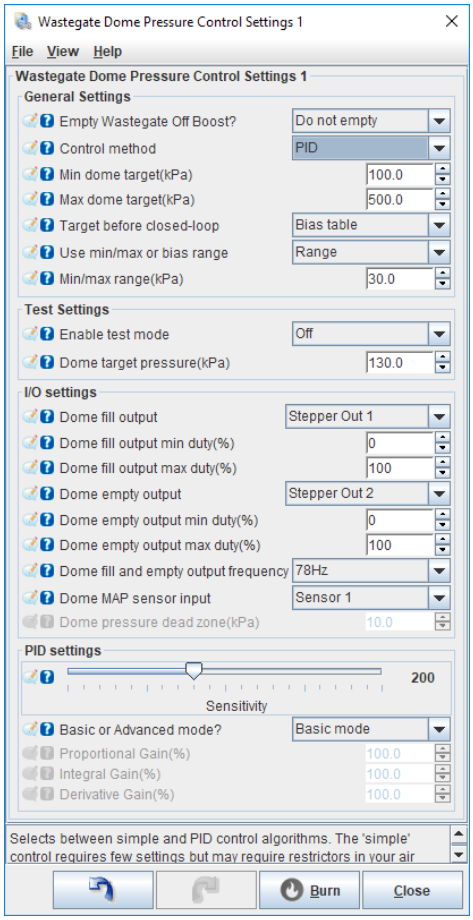
- **Boost control enabled** - allows a second solenoid output using the second set of boost tables.
- **Boost control pin** - specifies which output to use for the second solenoid.

The closed duty, open duty, lower limit, algorithm, and PID settings work in the same way as the primary boost control settings.

- **MAP sensor to target** - When running in closed loop, the second boost control solenoid will not target the main MAP sensor, as most setups that require different wastegate pressure from side to side also require different compressor outlet pressures. You will need to define what the second MAP sensor is on the Generic Sensor Inputs screen in the Advanced Engine menu.

7.6.9 Wastegate dome pressure control settings

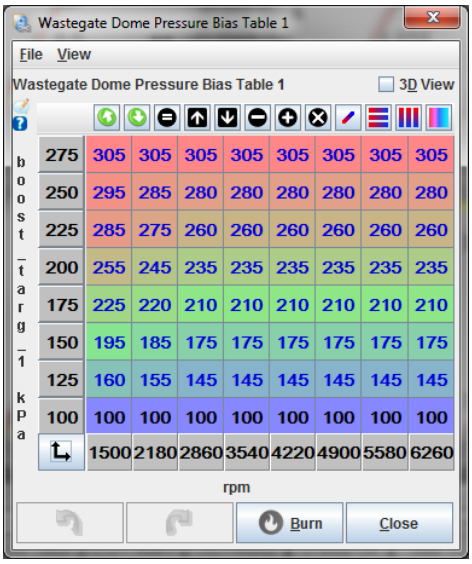
This screen includes additional boost control settings only used when running dome pressure control.



- **Empty wastegate off boost?** - This controls whether the dome remains pressurized when no boost is commanded, or if the outlet solenoid should vent the dome to atmosphere under these circumstances.
- **Control method** - Simple control pulses the solenoids in quick increments until the dome pressure reaches its target pressure. PID uses a proportional, integral, derivative loop to adjust duty cycles to reach dome pressure. This method is more complicated to tune, but can result in less overshoot if tuned correctly.
- **Min dome target** and **max dome target** - If using closed loop control, this sets upper and lower limits on what dome pressure targets the MS3Pro can use to reach a given boost target.
- **Target before closed-loop** - When the boost is less than the target boost minus the boost control delta (see the main boost control settings screen for these settings), you can either set the dome pressure to use the target taken from the bias table, or go to the maximum dome pressure. Using the maximum dome pressure can often build boost faster, but uses more CO_2.
- **Use min / max or bias range** - If this is set to “Range”, the MS3Pro will only adjust the dome target to the value from the bias table plus or minus the range number when in closed loop targeting. Setting this to “Range” can help if you find the MS3Pro makes abrupt changes when the engine changes RPM or boost targets directly and overshoots, but limits the maximum correction the MS3Pro can apply.
- **Min/max range (kPa)** - If the bias range is enabled, this is the most the closed loop correction can change the dome target from the value set by the bias table.
- **Enable test mode** - If this is set, the MS3Pro will activate the solenoids and attempt to fill the solenoid dome to the target pressure. The engine must not be running to use test mode.
- **Dome pressure target** - Specifies what pressure to target in test mode.
- **Dome fill output** - Selects what output pin is used to trigger the solenoid that delivers pressure to the wastegate dome.

- **Dome fill output min duty** and **dome fill output max duty** - If using PID control for the dome pressure, this sets upper and lower limits for what duty cycles to use when increasing pressure.
- **Dome empty output** - Selects what output pin is used to trigger the solenoid that vents pressure from the wastegate dome to atmosphere.
- **Dome empty output min duty** and **dome empty output max duty** - If using PID control for the dome pressure, this sets upper and lower limits for what duty cycles to use when decreasing pressure.
- **Dome fill and empty output frequency** - If using PID control, this setting controls how rapidly to pulse the solenoids.
- **Dome MAP sensor input** - Specifies which sensor channel to use for the dome pressure sensor. This must be set up under Generic Sensors.
- **Dome pressure dead zone** - When using simple mode, this sets up a range of pressures between where to fill the dome and where to empty the dome. If the dome pressure is within this range of the target pressure, the MS3Pro will consider the pressure level "good enough" and close both solenoids unless the pressure reading or target dome pressure changes. If you have trouble with the dome pressure overshooting, try increasing this number.
- **PID settings** - These adjustments control dome pressure response if using PID control.
  - **Closed Loop Sensitivity** - Use this setting to tune how aggressively the boost controller tries to reach the target boost. Adjust the slider up until the boost oscillates around the target, then back down until it stops.
  - **Basic or Advanced Mode** - Activating advanced mode allows adjusting the individual P, I, and D settings. In Basic Mode, these are ignored and the MS3Pro just uses the slider setting.
  - **Proportional Gain** - Proportional gain affects the strength with which changes in input immediately affect changes in output.
  - **Integral Gain** - The Integral Gain setting affects the response to continued difference between the target boost and the actual boost.
  - **Derivative Gain** - The Derivative Gain setting helps to slow down the response of the Proportional and Integral gain settings as the target is reached. This should be used sparingly as it can also completely dampen the other two Gains.

7.6.10 Dome Bias Table



This pressure lets you specify what dome pressure to target as a function of target boost and RPM when using closed loop dome pressure based boost control. For example, in the table shown above, the MS3Pro will start at a 235 kPa dome pressure if targeting 200 kPa boost at 4220 RPM.

7.6.11 Turbo anti-lag (ALS)

Anti-lag is a combination of features you can use to keep the turbo spooled while shifting, beyond the basic flat shift rev limiter. The MS3Pro has several strategies it can use to dump extra fuel into the exhaust and increase exhaust gas temperatures under these conditions. This increase in exhaust temperature will shorten turbo life and possibly result in increased wear and tear on other parts of the exhaust system as well.

All anti-lag variations require an input switch to arm the system. When active, the anti-lag system will add fuel and retard timing to a number specified by the ALS tables. You can also allow the MS3Pro to use the following additional strategies when the anti-lag is active:

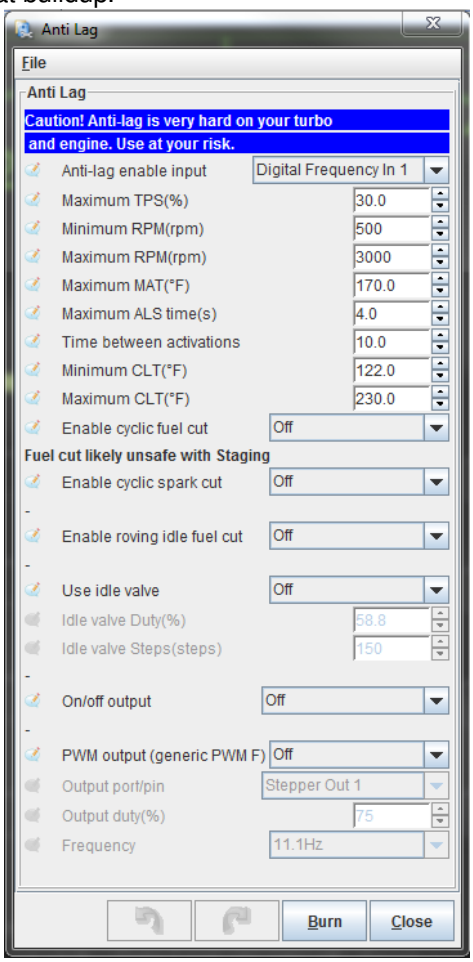
- Use a cyclic fuel cut
- Use a cyclic spark cut to dump unburned fuel into the exhaust
- Activate a solenoid or additional injector output to send fuel or air into the exhaust manifold directly
- Open the idle valve a set amount
- Open a solenoid to let additional air into the intake manifold

Note - use extreme caution if using any system that dumps fuel directly into the exhaust! This type of anti-lag can turn “danger to manifold” from an ironic “The Fast and the Furious” reference into a very real hazard.

As many anti-lag systems also have the throttle or idle control valve open by a large amount, the MS3Pro can also use a rotational idle fuel cut to reduce idle speed on engines using anti-lag. The rotational idle engages when anti-lag is off.

Anti-lag is only used when you release the throttle to shift; see the Launch Control section for how to set up the MS3Pro for no lift shifting.

Since anti-lag creates so much heat, the MS3Pro has several settings to limit the anti-lag system’s use to limit heat buildup.

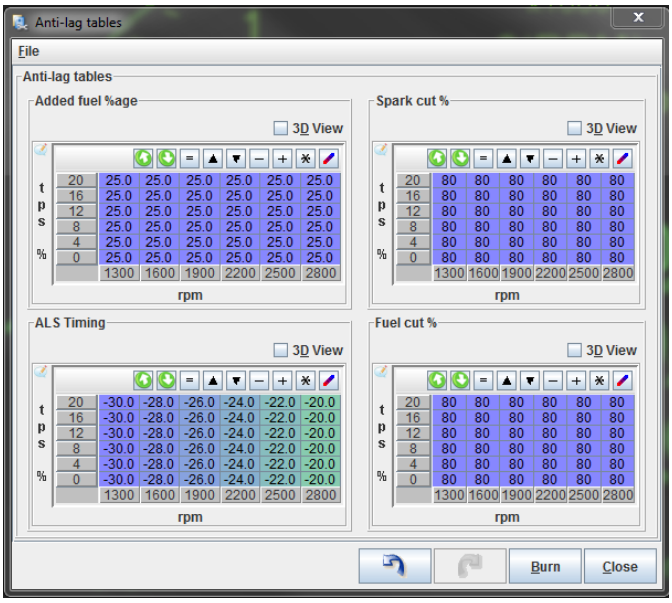


- **Anti-lag enable input** - Select what input pin is used to arm the anti-lag system.
- **Maximum TPS** - Specify the maximum throttle position for anti-lag to be active. Note: If using drive by wire, this setting uses the accelerator pedal position and not the throttle position, allowing the drive by wire to open the throttle beyond the maximum TPS as long as the accelerator pedal position is below this value.
- **Minimum RPM** - Anti-lag is shut down if the RPM drops below this point.
- **Maximum RPM** - Anti-lag is disabled above this RPM.
- **Maximum MAT** - Anti-lag shuts down if the air temperature goes over this point.
- **Maximum ALS time** - The longest allowable time, in seconds, for the anti-lag system to operate.
- **Time between activations** - This is used to prevent a brief sensor reading out of the anti-lag range from resetting the anti-lag timer. The MS3Pro treats any set of activation conditions in this time block as a single activation, even if it briefly comes out of the TPS or other conditions that would activate anti-lag.
- **Minimum CLT** - The lowest coolant temperature at which anti-lag can be used.
- **Maximum CLT** - Anti-lag is disabled above this temperature.
- **Enable cyclic fuel cut** - If this is turned on, the anti-lag system will use a rotating fuel cut when active. Do not use this with staged injection, as you may get a lean condition.
- **Enable cyclic spark cut** - Cuts spark when the anti-lag system is active.
- **Enable roving idle fuel cut** - This uses a secondary fuel cut to reduce idle speed. Roving idle fuel cut is active when anti-lag is off.

Note that cyclic fuel cut and roving idle fuel cut **only** work with full sequential fueling. They **do not** work in semi-sequential or batch fire fuel modes.

- **Use idle valve** - If set to on, the idle valve will open an additional amount when anti-lag is active.
- **Idle valve duty** - The amount of extra duty cycle to apply to a PWM valve if the idle valve is active.
- **Idle valve duty steps** - Like the above setting, but for a stepper IAC valve.
- **On/off output** - Allows you to activate an on/off output when anti-lag is active. Use this setting to specify which pin.
- **PWM output** (generic PWM F) - Allows you to turn on a PWM output when anti-lag is active.
- **Output port** / **pin** - Specifies the output pin for the PWM output.
- **Output duty** - The PWM output is driven at a fixed duty when anti-lag is active. Use this setting to specify what duty cycle.
- **Frequency** - Specifies the frequency to use with the PWM output.

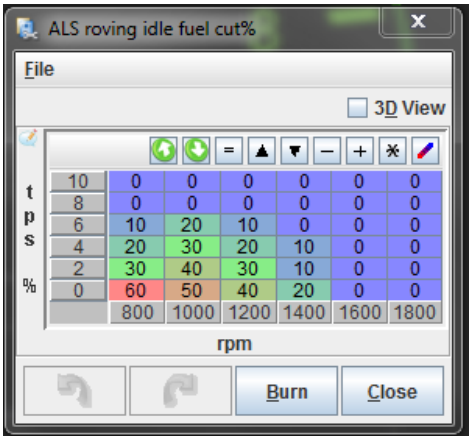
7.6.12 Turbo anti-lag tables



Anti-lag functions operate on a TPS vs RPM basis. The TPS range is fairly low as anti-lag is only active at part throttle. The tables available will depend on what anti-lag options have been enabled, although the Added Fuel %age and ALS Timing will always be available if anti-lag is on.

- Added fuel %age - The amount of extra fuel to add when anti-lag is active.
- ALS timing - This is a timing table in absolute numbers which the MS3Pro will use when anti-lag is active. As the MS3Pro specifies timing with BTDC numbers being positive, negative numbers indicate to retard the timing to after top dead center.
- Spark cut % - The percentage of spark events to cut if cyclic spark cut is enabled.
- Fuel cut % - The percentage of injection events to cut if cyclic fuel cut is enabled.

7.6.13 ALS roving idle fuel cut %



This screen is enabled when the roving (rotational) idle is enabled. It allows you to set the percent of fuel injection events to cut to limit idle speed on cars that do not allow added air bypass items and therefore have the throttle adjusted open to allow more air to flow into the turbo with the throttle “closed”. Note that the right hand column needs to be set to all zeroes.

7.6.14 VVT settings

Many engines now feature variable valve timing. This allows a balance between engine performance and economy, with the ECU automatically adjusting the valve timing to suit operating conditions. The particular implementation varies with engine and manufacturer. The first systems available are of the "on/off" variety where the cam is moved by a fixed angle only, and there is no feedback systems. Examples of this include:

- Honda first generation VTEC
- Toyota first generation VVT
- BMW single VANOS

More advanced systems use continuously variable cam controls - typically an oil solenoid valve controlled by a PWM signal from the ECU. The ECU uses one camshaft position sensor per cam to compare the cam advance to a crankshaft position sensor reading. Examples of continuously variable systems include:

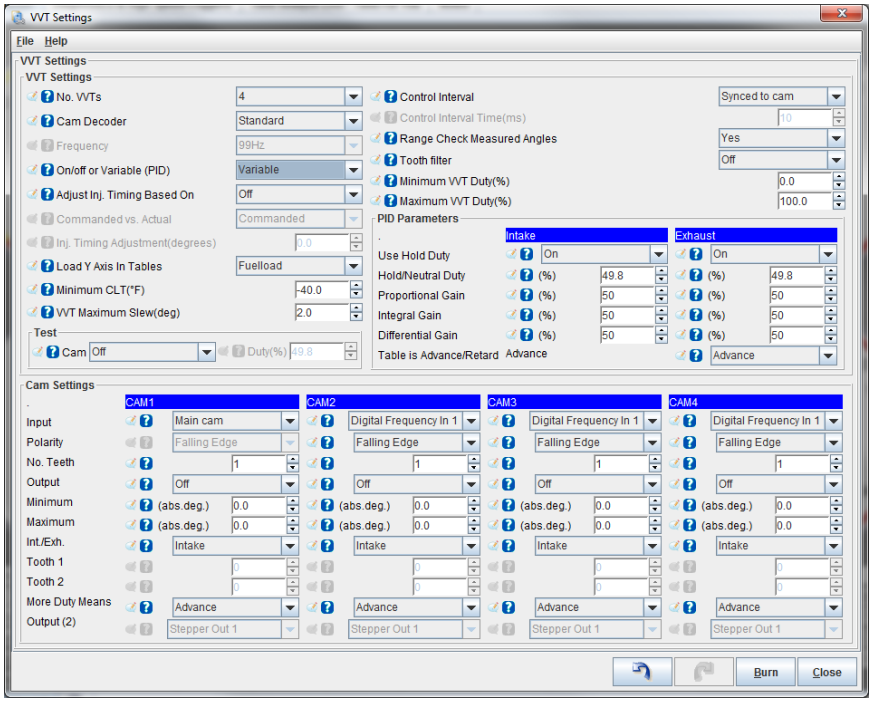
- Toyota VVTi
- BMW double VANOS
- Ford Zetec and Ti-VCT

Most continuously variable VVT systems use a single solenoid where a duty near 50% holds the cam in position. More than 50% causes it to advance or retard. Less than 50% moves it in the opposite direction. Since the duty cycle controls the direction the cam moves and not the absolute position, continuously variable valve timing requires closed loop control. The MS3Pro uses a missing tooth wheel (or otherwise irregular pattern) on the crankshaft compared to a sensor on the moving cam to determine camshaft position, and uses this to adjust the solenoid duty cycle.

BMW's Double VANOS is a bit of an oddity - it uses separate solenoids to advance and retard the cam, much like a 3 wire PWM idle control valve. The V10 version of Double VANOS is a more conventional system with one solenoid per cam.

If you are controlling more than one cam, you will need to use the digital frequency input circuits for cam input. These can read most Hall effect sensors with no modification. If your engine uses VR type camshaft position sensors, you will need external VR conditioner circuits to read the additional sensors if not using a dedicated VR input.

If you find you have run out of PWM outputs, you can use the high current outputs to control a VVT oil control valve. However, these outputs will cause a valve to close faster, making it harder to “park” the valve in an intermediate position. You can enable the same slow closing as on PWM outputs by connecting a 1N4001 diode across the VVT valve, with the banded end connected to the 12 volt supply.



- **No. VVTs** - the number of VVT actuators to be controlled
- **Cam Decoder** - Standard mode works with either cam teeth that are all equally spaced, or a hard coded spark mode. Nonstandard cam decoders work with engines that use Toothed Wheel mode with an unusual cam wheel. Currently supported modes include BMW V10s, the BMW S54's double VANOS setup, Chrysler VVT Hemi engines, Ford Coyote V8, and Ford Duratec 4 cylinder. BMW and Chrysler modes use a 60-2 crank trigger, while Ford modes use a 36-1.
- **Frequency** - the PWM output frequency. Adjust to suit your solenoids.
- **On/Off or PID** - the mode of operation. Full variable systems should use PID.
- **Adjust inj. timing based on** - whether the Injector Timing for sequential injection should be adjusted as the cams are moved.
- **Commanded vs Actual** - If the injection timing adjustment is on, this specifies whether the commanded or actual timing is used to make the adjustment.
- **Inj. Timing Adjustment** - for on/off mode, this specifies the Injector Timing adjustment when VVT is on.
- **Test output** - during initial setup the test mode is used to confirm solenoid operation and learn the cam range of movement. The output specifies which cam should be altered.
- **Test duty** - the PWM duty cycle to be applied to the chosen VVT output under test.
- **Control Interval** - Synced to Cam adjusts the duty cycle each time the MS3Pro updates its data on the cam position, which usually gives the best response. Timed mode adjusts the duty cycle once per specified timed interval.
- **Control Interval Time** - the time interval in milliseconds for adjustments in timed mode.
- **Range Check Measured Angles** - Checks to see if the measured range is within the minimum to maximum settings. If the cam moves out of range, this triggers a cam fault.
- **Tooth Filter** - Applies the cam noise filtering settings to the VVT decoder.
- **Minimum VVT Duty** - The lowest duty cycle to use for the solenoid.
- **Maximum VVT Duty** - The highest duty cycle to use for the solenoid.

PID Parameters - these are used to adjust the closed loop feedback. There are separate settings for intake and exhaust.

- **Use hold duty -** Specifies whether to default to a set value when no cam change is commanded.
- **Hold/neutral duty** - the duty at which the cam stays still.
- **Proportional gain** - Proportional gain affects the strength with which changes in input immediately affect changes in output.
- **Integral gain** - The Integral Gain setting affects the response to continued difference between the target cam angle and the actual angle.
- **Difference gain** - The Derivative Gain setting helps to slow down the response of the Proportional and Integral gain settings as the target is reached. Usually smaller than the other two numbers.
- **Table is Advance** / **Retard** - Controls whether the table reads the degrees advanced from the minimum advance, or degrees retarded from minimum retard. Intake cams always use advance, while exhaust cam tables may use a retard table instead.

Cam Settings - allow you to specify each cam's output, function, and range of movement.

- **Input** - the position sensor input for this cam. (Cam 1 always uses the main cam sensor input set in Ignition Settings.)
- **Polarity** - The capture polarity of the cam input. (Cam 1 typically follows the main settings.)
- **No. teeth** - How many capture teeth on the cam wheel.
  - Note: If you are using a wheel that can be described as a set of equally spaced teeth with one or more teeth missing, set this to the number of base teeth used. If no pulse is seen during the expected cam range where there should be a cam tooth, the MS3Pro simply "throws out" the reading. For example, if you have a cam trigger with teeth at 0, 60, 120, and 240 degrees, sett the number of teeth to 6. The MS3Pro will look for a tooth every 60 degrees, and simply will not update the cam position at the 180 and 300 degree points.
- **Output** - Which output pin the ECU uses to activate the cam control solenoid. For systems using two solenoids per cam, this is the solenoid that advances the cam.
- **Minimum** - the minimum absolute angle from the cam sensor. See setting up. This is the angle of the signal from the sensor - not necessarily the actual cam center angle.
- **Maximum** - the maximum absolute angle from the cam sensor.
- **Int./Exh.** - specifies if the cam is an intake or exhaust cam. This determines the PID parameters, direction and target table to be used.
- **Tooth 1** - only used by BMW V10 at present. Look for the cam tooth edge after this crank tooth no.
- **Tooth 2** - same again, but for second crank revolution.
- **More duty means** - whether more PWM duty on the solenoid causes the cam to advance or retard.
- **Output (2)** - For cams with two solenoids, this is the output pin used to activate the solenoid that retards the cam.

The most basic setup is to make the wiring connections for inputs and outputs. Most engines will run safely with VVT inactive; ensure this is the case on your engine. Having completed the wiring, set the basic configuration reflecting the wiring choices you made, the control type and the number of teeth on the cam wheels. Leave the min/max cam angles as zero for now.

On your TunerStudio dashboard, enable the gauges VVT_angle (1,2,3,4 as required). Start the engine and observe the VVT angle gauges. Use the test mode for each VVT with 0 and 100% duty and observe the swing of the cam angles. You will hear the engine tone change as the cam angle changes. It is important that the VVT

angles you observe are stable and vary evenly. With continuously variable cam timing, the VVT angle will normally swing from fully retarded to fully advanced (and vice versa) quite quickly with any duty beyond the hold/neutral duty, so it will be unlikely that you will be able to catch the cam at anything other than the limits of its travel. Double check that the cam min/max angles are presently set to zero.

For each cam, record the minimum and maximum angles you observe while varying the test duty. Enter these into the min/max fields. Note that the maximum MUST be larger than the minimum. (In the instance where the angles cross 720 degrees, add 720 to the maximum angle. e.g. if minimum was 700 and maximum was 40 degrees, enter 700 and 760.) At this stage you should also be able to determine if more duty is retarding or advancing each cam.

Having configured the inputs and outputs, proceed with PID tuning. Take datalogs and observe the VVT actual angle compared to the target angle. We recommend tuning in much the same way as boost control. Start with the I and D terms zeroed out, dial in the P term first, followed by the I. If you have problems with the cam overshooting its targets, gradually increase the D term until it damps out the overshoots.

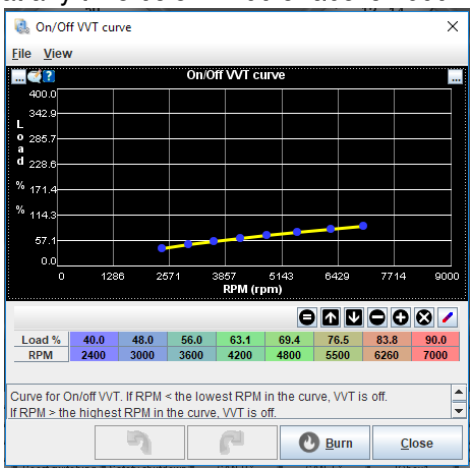
Once the cam control is working, you may dial in the VVT activation tables. These are usually best to dial in on a dyno.

7.6.15 VVT intake and exhaust tables

The intake and exhaust VVT tables show the cam position as determined by the angle of the cam sensor signal relative to the crankshaft position sensor. Usually, this one requires dyno tuning to determine what settings perform best.

7.6.16 VVT on/off curve

For on/off mode, no feedback of cam position is used and most settings on the VVT settings screen are unused and grayed out. The output value, frequency and injector timing adjustment are set through the main settings screen. You use this curve to set the minimum engine load to switch the VVT on. The VVT output will switch off if the RPM is below or above the range on the curve, regardless of load. In the example below, the VVT would be off at any time below 2400 or above 7000 RPM, and would need higher load values to switch on at higher RPM.



7.7 Table Choices

For many installs a single fuel table is all that is required. New users are strongly advised to start with a single table.

MS3Pro offers a number of methods of switching and blending between tables. These methods can be used to effectively extend the basic table size, for better tunability with different fuel algorithms (e.g. low vs high rpms), for dual fuels, or for different fuel mixes. Presently, fuel, ignition, boost and AFR targets may be blended or switched. Due to the complexity introduced with multiple tables, only experienced users should consider enabling these features. Additionally, the algorithm blending should only be enabled once each table is well tuned. Autotuning (VE Analyze Live) will not function on some combinations of blended tables.

Terms used:

- Table switching - the ability to swap from one table to another, based on RPM, TPS, MAP or a physical switch input
- Combined tables - two tables are combined to give a single VE (or spark) result
- Algorithm blending - two tables with different calculation algorithms are used to calculate fuel (or spark) and then the result is blended together
- Dual table - two tables operate independently to control injectors or injector banks
- Staged injection - one fuel calculation control two banks of injectors. e.g. small injectors for idling and larger injector for full power. (See Staged Injection page.)

Algorithms are different strategies for determining the "load" on an engine and then calculating fuel and spark etc. As a brief recap:

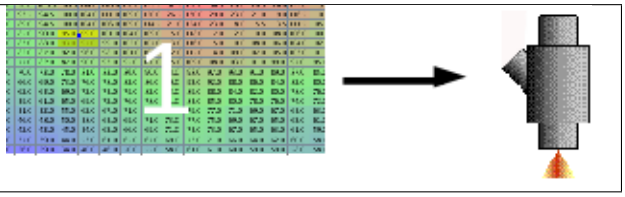
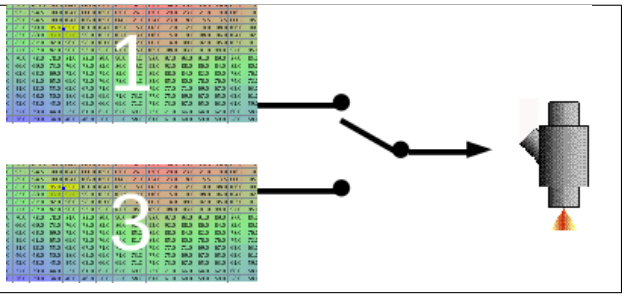
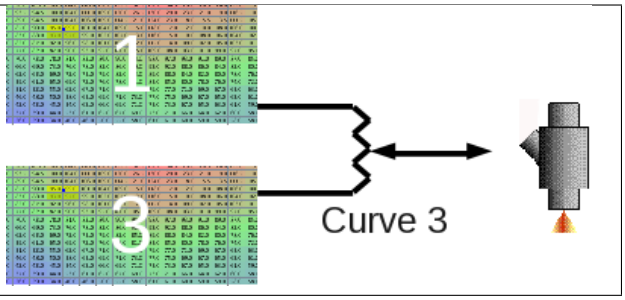
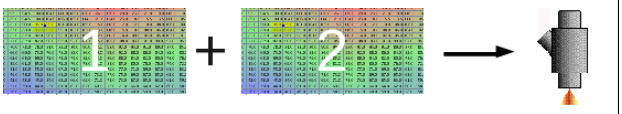
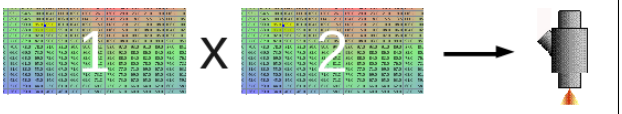
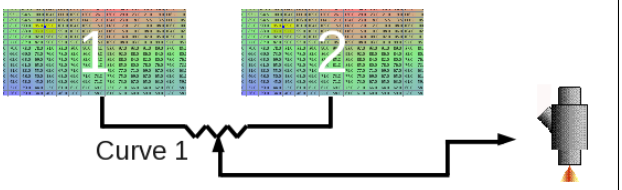
- Speed-Density - Uses MAP as load.
- Alpha-N - Uses TPS as load.
- MAF - Measures actual airflow. Spark load uses a calculated percentage of cylinder filling.
- %baro - Uses MAP divided by barometric pressure as load.
- ITB - Uses a mix of MAP and TPS as load.

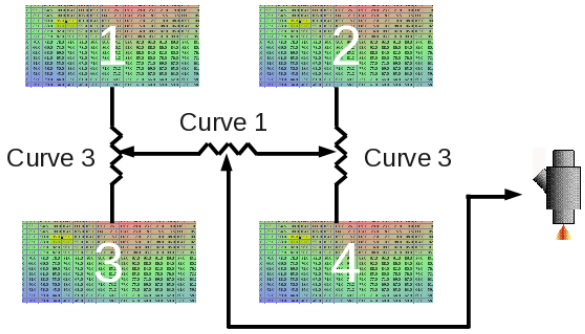
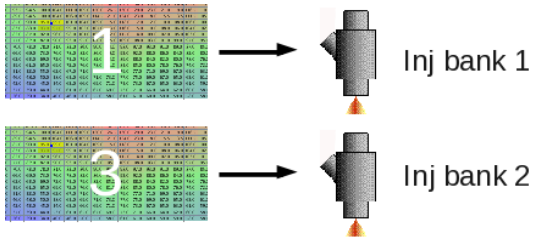
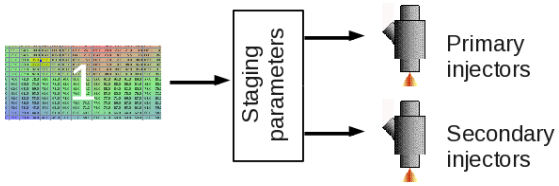
7.7.0.1 Example scenarios Why use table switching or blending at all? Actually, most users won't need to, and can get by just fine with a single table. Here are some examples of engines that could genuinely need table switching or blending.

- A big block Chevy has two different types of port designs, one half of which flow considerably better than the other half. Such an engine can use true dual table mode (although you can also use the individual cylinder trim to achieve the same effect).
- A large cammed motor with a poor MAP signal below 2,000 RPM that stabilizes at higher RPM could use blended alpha N and SD, with the motor running on alpha-N at low RPM and speed density at high RPM.
- A turbo motor with a very large MAF for good fuel metering that runs out of MAF resolution at low flow could use blended speed density and MAF fueling, with speed density at low RPM and the MAF at high RPM.
- A motor with a very wide power band could use two separate speed density tables, switching them based on RPM. This can effectively be used to create a 30 x 16 fuel table.
- A motor with switchable intake runners could use separate tables for when the long runners are open vs when the short runners are open.
- A race car that runs in classes with different fuel requirements could have separate maps for pump gas and race gas.
- A vehicle with a flex fuel sensor can blend in between separate gasoline and ethanol maps, interpolating between the two for the percent of fuel in the tank.

7.7.0.2 Fuel blending and switching types There are four VE tables, numbered 1 through 4.

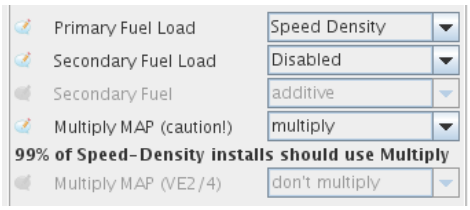
Here are some of the possible combinations of fuel table blending and switching. Electrical symbols for switches and variable resistors illustrate the switch or blend functions.

Method	Diagram
Single table. One fuel algorithm uses VE1. This is the most common setup.	
Single algorithm, with table-switching. One fuel algorithm uses VE1 or VE3 depending on table switch setting/input. Could be used with dual-fuel or also as a method to extend effective table size.	
Single algorithm with table-blending. One fuel algorithm uses a proportion of VE1 or VE3 depending on the blend input and curve (3). The tables could be blended based on ethanol percentage in the fuel from a flex-fuel sensor.	
Combined tables - secondary additive table. Fuel is calculated by adding the lookup from VE1 and VE2. VE2 can be used a tweak to VE1 to add a 4th dimension to the fuel table e.g. including a throttle component in a speed-density tune. 0% in the secondary table means no change, 2% means add 2%	
Combined tables - secondary multiplicative table. Fuel is calculated by multiplying the lookup from VE1 and VE2. VE2 can be used a tweak to VE1 to add a 4th dimension to the fuel table e.g. including a throttle component in a speed-density tune. 100% in the secondary table means no change, 102% means add 2% etc.	
Secondary algorithm blended table. Fuel is calculated from VE1 using the primary fuel algorithm and also from VE2 using the secondary fuel algorithm and then the result is blended. This could be used to have a blend between a Speed-Density in VE1 and MAF tune in VE2. The blend curve (1) allows you to alter where each algorithm is in use. Each table should be tuned individually i.e. set the curve to 0% and tune VE1, then set to 100% and tune VE2, afterwards enable blending.	

Two way blend. Fuel is calculated from VE1 and VE3 using the primary fuel algorithm. These are blended. Then VE2 and VE4 are calculated using the secondary fuel algorithm and blended. The VE1/3 result is then blended with the VE2/4 result. This could be considered for E85 (VE3/4) vs regular gasoline (VE1/2) where algorithm blending is also used between left and right, such as a blend of Speed Density and MAF. This is incompatible with 'Dual Table' as that uses VE3/4 for the second bank. Two way blending is complex ! Do not enable it without fully tuning each individual table.	
Dual table. Fuel for injector bank 1 is calculated from VE1 and fuel for injector bank 2 is calculated from VE3. This could be used where an engine has radically different fuel requirements between banks (or cylinders in a V twin.) or as part of a dual-fuel install where the second bank is used for LPG or methanol etc.	
Injector staging. Fuel is calculated from VE1 and the staging parameters determine the split between primary and secondary injectors. This is typically used on engines with a large dynamic range so that small injectors may be used at low loads for good tunability with additional (staged) large injectors coming online at full power. This concept is used from the factory on Mazda rotary engines.	
More variations. There are many more combinations possible to include double blending, switching between blended tables etc.	

These are partially set in the Basic / Load Settings and partially set in the Table Choices menu.

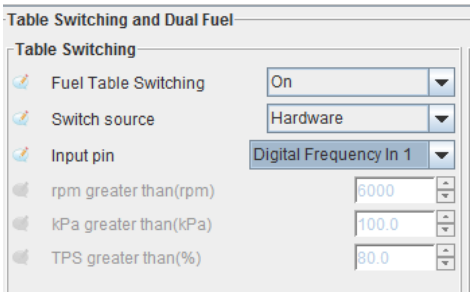
7.7.0.3 Fuel settings



Single algorithm, single table

By far the most common install is to use a single Speed-Density fuel table. This really should be your starting point before considering other load methods and certainly before multiple-table configurations are enabled.

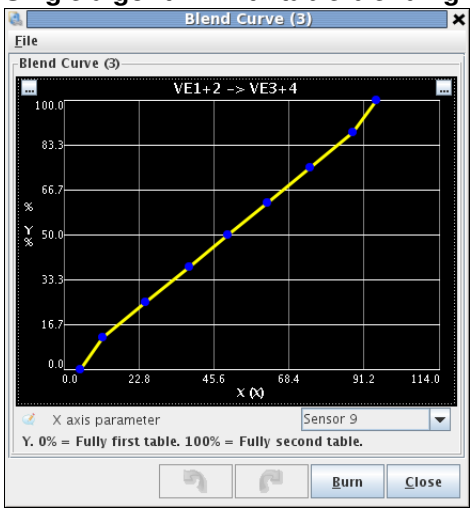
Multiply MAP controls whether MAP is included in the calculation for VE1 (and VE3). For regular Speed-Density this should almost always be enabled. For "Pure" Alpha-N, leave it disabled. For "Hybrid" Alpha-N, enable it. For MAF, it is ignored.



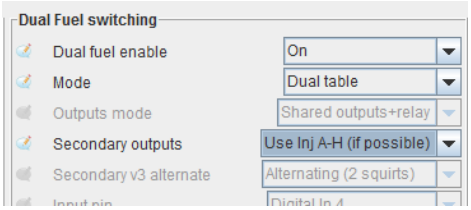
Single algorithm with table switching

Here we have table switching enabled in the upper left hand corner of the Table Switch / Dual Fuel menu. A physical switch input (shown here connected to “Digital Frequency In 1”) is used to swap between VE1 and VE3 for fueling. The same algorithm is used for both tables. The RPM, TPS, kPa switch points can be used as a form of table extension where say VE1 is 0-3500rpm and VE3 is 3500-7000rpm.

Single algorithm with table blending



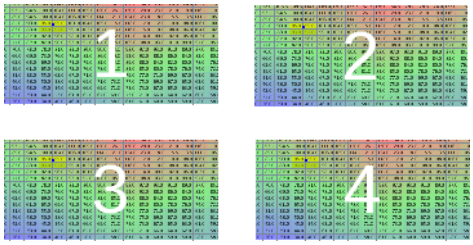
Blend curve (3) is used to define the blend between VE1 and VE3 which use the same algorithm. At 0% of ’sensor9’ (could be a trim pot) the fuel is calculated from VE1. At 100% of ’sensor9’ the fuel is calculated from VE3. In-between, proportions of table will be used. It is very important to tune each table individually - trying to retune with a partial blend is likely to result in confusion or failure.



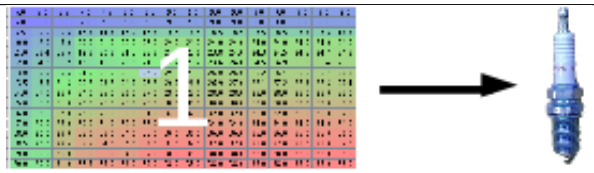
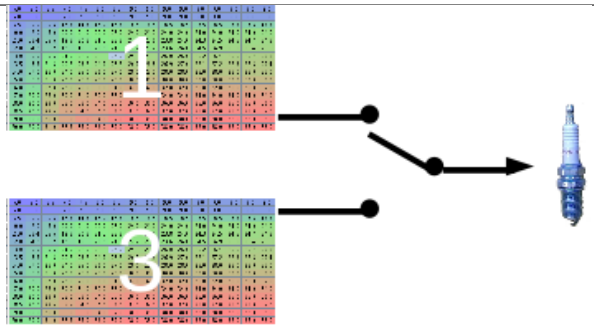
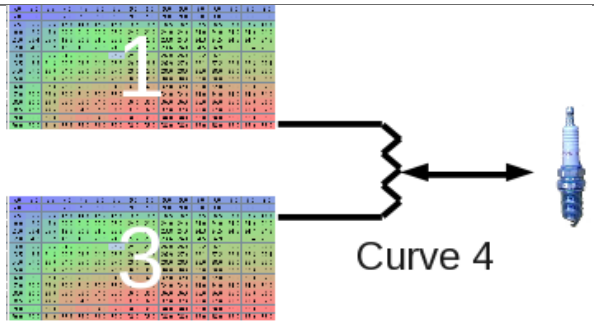
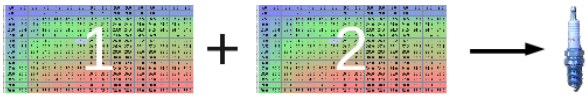
Dual table

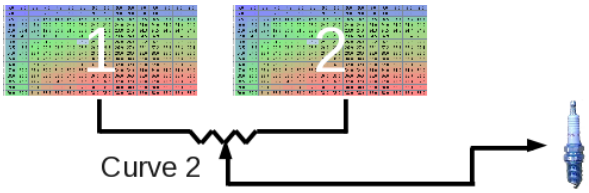
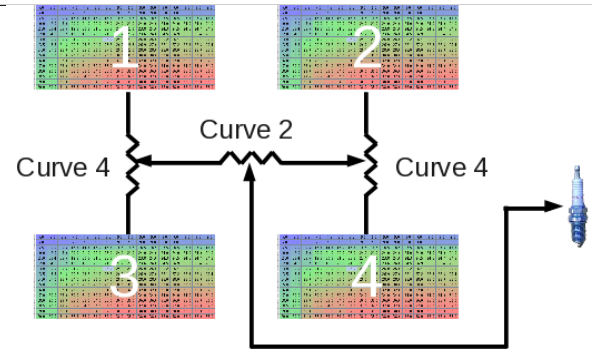
Fuel for injector bank 1 is calculated from VE1 and fuel for injector bank 2 is calculated from VE3. This can also be used in conjunction with VE1/2 (and VE3/4) combined tables and algorithm blending. It is not compatible with table switching or blending from VE1 to VE3.

7.7.0.4 Spark / ignition blending and switching types There are also four spark tables, with switching and blending options similar to fuel.



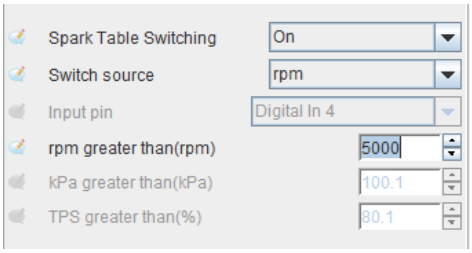
Some options are not present as they do not apply to ignition; you cannot multiply spark tables, for instance, or stage two sets of spark plugs (although the rotary split tables are a similar idea).

Method	Diagram
Single table. One ignition algorithm uses Spk1. This is the most common setup.	
Single algorithm with table-switching. One ignition algorithm uses Spk1 or Spk3 depending on table switch setting/input. Could be used with dual-fuel or also as a method to extend effective table size.	
Single algorithm with table-blending. One ignition algorithm uses a proportion of Spk1 or Spk3 depending on the blend input and curve (2). The tables could be blended based on ethanol percentage in the fuel from a flex-fuel sensor.	
Combined tables - secondary additive table. Ignition is calculated by adding the lookup from Spk1 and Spk2. Spk2 can be used a tweak to Spk1 to add a 4th dimension to the fuel table e.g. including a throttle component in a speed-density tune.	

Secondary algorithm blended table. Ignition is calculated from Spk1 using the primary ignition algorithm and also from Spk2 using the secondary ignition algorithm and then the result is blended. This could be used to have a blend between a Speed-Density in Spk1 and MAF tune in Spk3. The blend curve (2) allows you to alter where each algorithm is in use. Each table should be tuned individually i.e. set the curve to 0% and tune Spk1, then set to 100% and tune Spk2, and finally enable blending.	
Two way blend. Ignition is calculated from Spk1 and Spk3 using the primary ignition algorithm. These are blended. Then Spk2 and Spk4 are calculated using the secondary ignition algorithm and blended. The Spk1/3 result is then blended with the Spk2/4 result. This could be considered for high-octane (Spk3/4) vs low octane (Spk1/2) where algorithm blending is also used between left and right.	

Single table

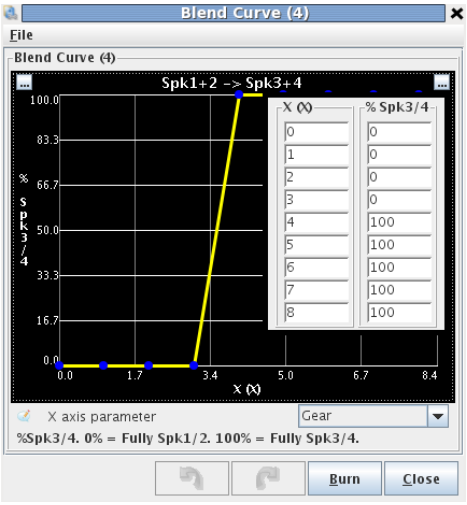
By far the most common install is to use a single Speed-Density ignition table. This really should be your starting point before considering other load methods and certainly before multiple-table configurations are enabled.



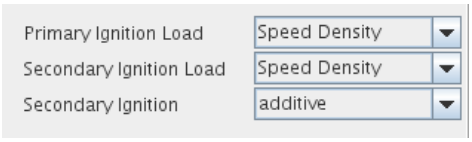
Single algorithm with table switching

In the above screen shot, the MS3Pro is set to switch the spark tables at 5000 RPM. You can also set the switching to kPa or throttle position, or use a physical switch.

Single algorithm with table blending

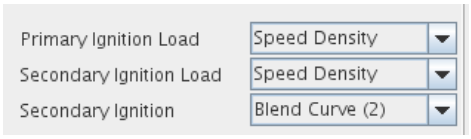


Blend curve (4) is used to define the blend between Spk1 and Spk3 which use the same algorithm. Here it is used for gear based spark table switching. Spk1 is used for gears 1,2,3. Spk3 is used for gears 4,5. It is very important to tune each table individually - trying to retune with a partial blend is likely to result in confusion or failure.

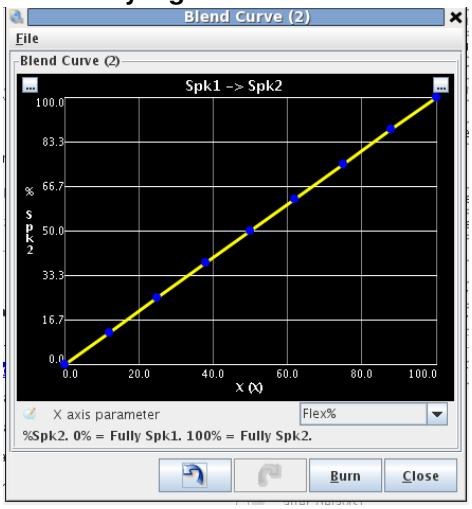


Combined tables - secondary additive table

Spk1 is combined with Spk2 using addition (the advance numbers are added together.) This can allow a secondary algorithm to act as a modifier or as a means to extend table size e.g. Spk1 = 0-100kPa and Spk2 = 100-400kPa with the same primary and secondary algorithm selected. In the above screen shot, it's being used to extend tables.



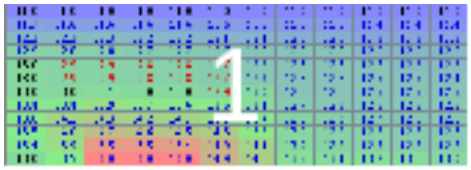
Secondary algorithm blended table

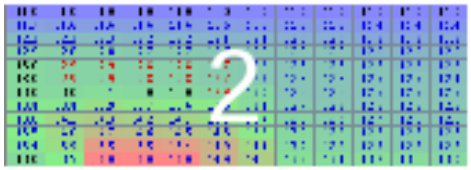


Blend curve (2) is used to define the blend between Spk1 and Spk2 which may use different algorithms. Here it is used to blend between the spark tables based on flex fuel percentage. It is very important to tune each table individually - trying to retune with a partial blend is likely to result in confusion or failure.

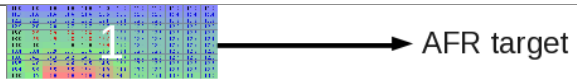
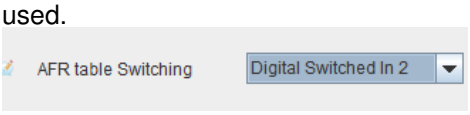
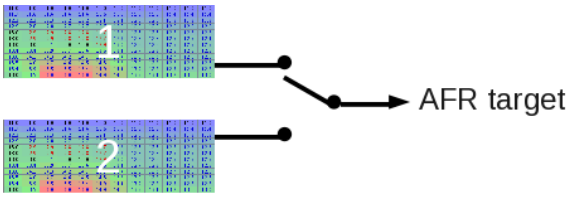
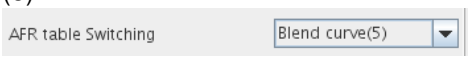
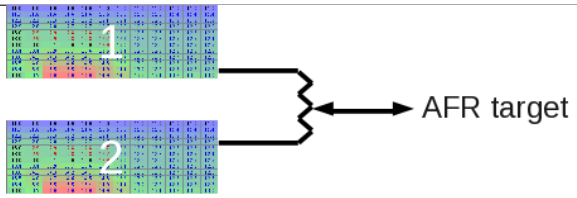
7.7.0.5 AFR blending and switching types

There are two AFR tables, AFR1 and AFR2.





The MS3Pro has three options for choosing these.

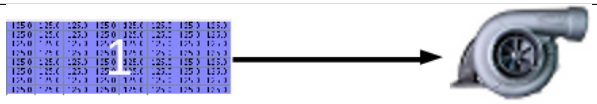
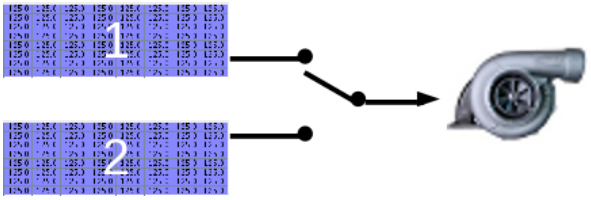
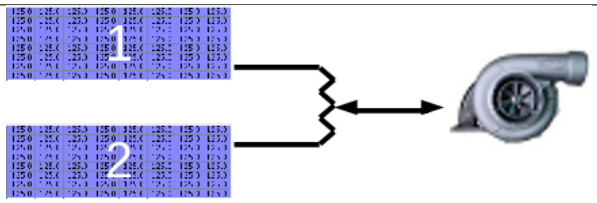
Method	Diagram
Single table. Target AFR is determined from AFR1. This is the most common setup.	
Switched tables.Target AFR is determined from AFR1 or AFR2. A physical switch input determines which table is used. 	
Blended tables. Target AFR is determined as a blend between AFR1 and AFR2 using Blend Curve (5) . 	

The target AFR is then be used for closed loop EGO or for Incorporate AFR. Note that even in "Dual Table" mode, only a single AFR target number is in use at one time. Both channels will target the same number if closed loop is enabled.

7.7.0.6 Boost table switching Like with the AFR tables, there are two boost tables, and you can either use one table, switch tables, or blend them. These work the same way as AFR table switching, except that you have the option of setting the boost table switching to use a gear input instead of a hardware switch, so you can use a different boost map in lower gears if you are traction limited. The boost table blending uses blend curve 6.

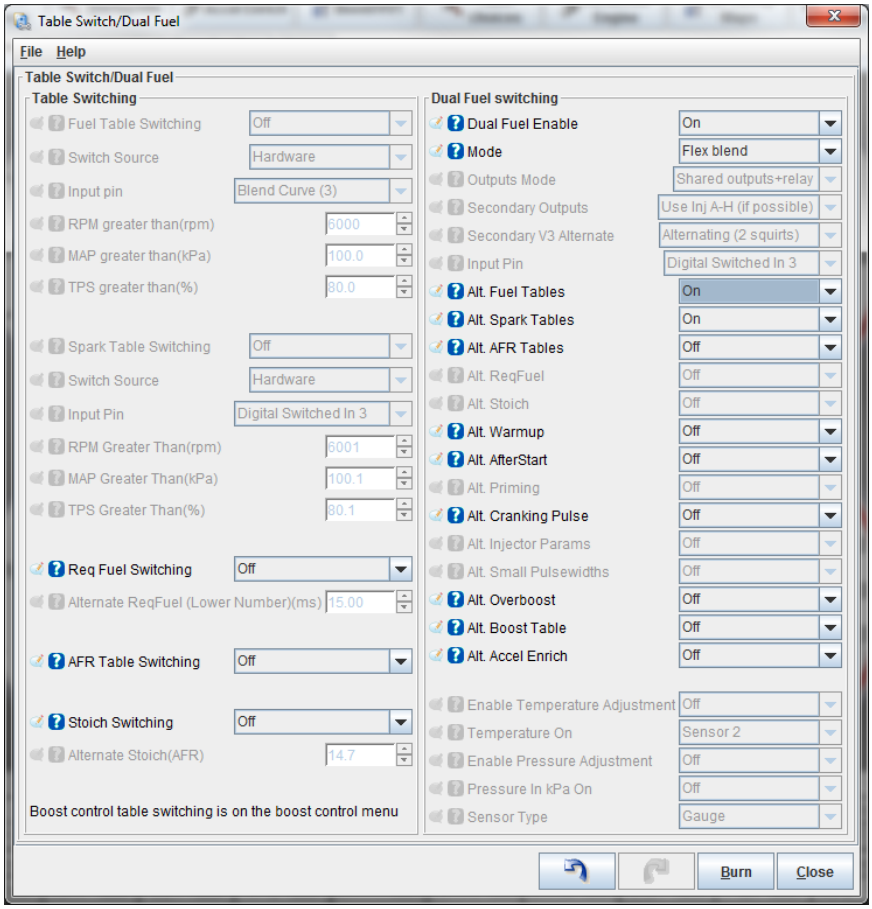
Note that you could effectively create a third boost setting by using a switch to shut off power to a boost control solenoid. The result would give you the lowest boost level permitted by the wastegate spring.

In Closed Loop mode these two tables are target boost. In Open Loop they are raw boost solenoid duty cycle.

Method	Diagram
Single table. Boost is determined from boost table 1. This is the most common setup.	
Switched tables. Boost is determined from boost table 1 or boost table 2. Either a physical switch or a gear option determines which table is used. 	
Blended tables. Boost is determined as a blend between boost table 1 and boost table 2 using Blend Curve (6). 	

7.7.1 Table Switch / Dual Fuel

This screen sets up many of the table switching options, as well as settings for running an engine on multiple fuel types. You can switch a wide variety of tables, not just VE and spark. Since fuels have different rates of evaporation and different amounts of sticking to walls, you can also change cranking settings, warmup and after-start enrichment, and acceleration enrichment as well.

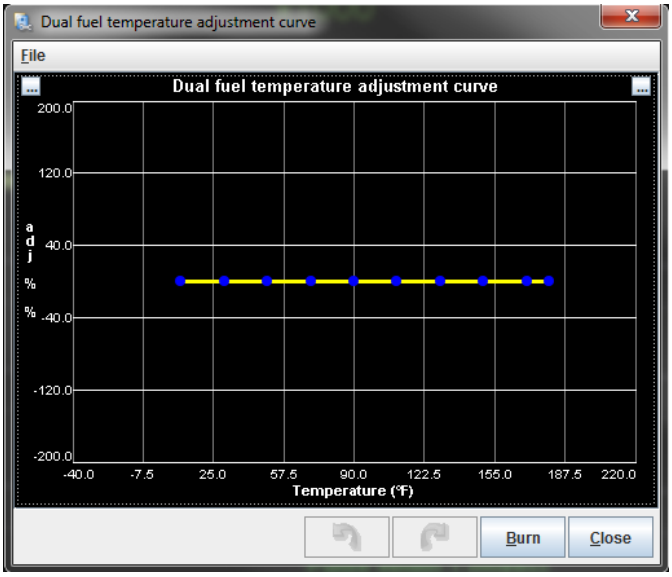


Several of the settings have been covered in the previous section; here, we'll explain the settings we have not yet covered.

- **Req Fuel Switching** - Allows you to use a hardware input to change the base pulse width number. The MS3Pro will use the lower number when the switch is active and the higher number when the switch is off.
- **Alternate ReqFuel** - The number to be entered when the switch is active.
- **Stoich Switching** - Switch to the alternate stoichiometric ratio number (for purposes of using the "incorporate AFR" into the fuel calculations) when the switch is active.
- **Alternate Stoich** - The air/fuel ratio used when the switch is active.
- **Dual fuel enable** - Enables modes that use multiple fuel types.
- **Mode** - MS3Pro allows several multi-fuel modes.
  - Switching: Switches between separate fuel systems and separate fuel (and ignition) tables at the same time. One fuel or the other is active.
  - Dual table: Both fuel systems are active at the same time, each running off their own fuel table. For example, a race car might have one set of injectors that spray gasoline and a second set that supplies methanol. The system uses a single spark table and uses the primary fuel outputs for starting.
  - Flex blend: The fuel system delivers a mixture of two separate fuels through one line and one set of injectors. The MS3Pro blends between two sets of settings based on the percentage of separate fuels, using a flex-fuel sensor input.
- **Outputs mode** - You can either use two sets of injectors wired to the same outputs and a relay to switch between them, or a second set of outputs for the secondary fuel injectors. Note that the code assumes that the same switch triggers the relay and the input pin - the MS3Pro does not trigger the relay.

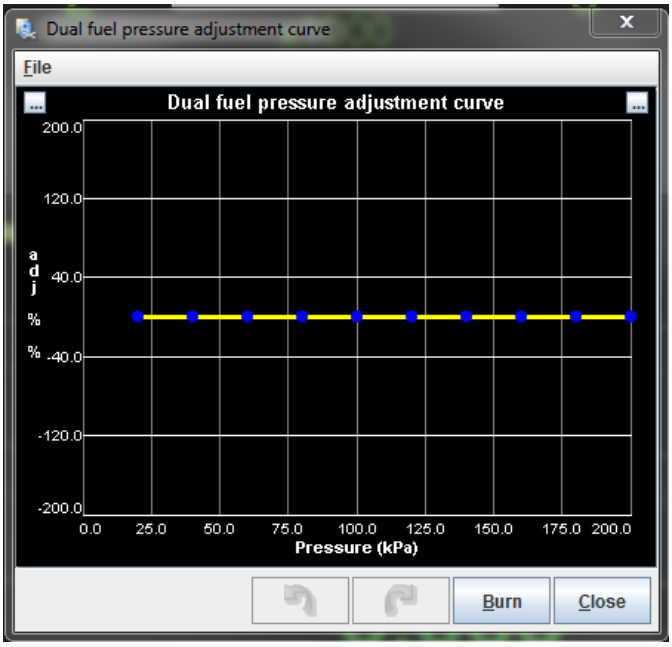
- **Secondary outputs** - You can either use the injector I and J in batch fire, or use the remaining outputs in the injector A through H series.
- **Secondary v3 alternate** - If using injectors I and J in batch fire, this determines if they fire at the same time or alternate.
- **Input pin** - Specifies what input is used to signal to switch to the secondary fuel.
- **Alt** parameters - allows switching the specified numbers when the dual fuel input pin is active. Note that this will override other table switching options.
- **Enable temperature adjustment -** Allows you to turn on a curve to compensate for fuel temperature for secondary fuel.
- **Temperature on** - Allows you to select the fuel temperature sensor. This must be defined in Generic Sensor Inputs.
- **Enable pressure adjustment** - Allows you to use a curve to compensate for fuel pressure on the secondary fuel.
- **Gauge pressure in kPa on** - Allows you to specify a pressure sensor for the secondary fuel. This also must be defined in Generic Sensor Inputs.

7.7.2 Temperature Adjustment



This screen is enabled for the secondary fuel in the dual fuel menu. This curve gives a percentage multiplier that is applied to the fueling as a function of temperature.

7.7.3 Pressure Adjustment



This screen is enabled for the secondary fuel in the dual fuel menu, and is intended for gaseous fuels, which often can have the tank pressure drop below regulator set pressure. It allows adjusting the fuel amount in percent as a function of fuel pressure.

7.7.4 Alt Injector Dead-time 2

If you are using a second set of injectors for dual fuel mode, this screen allows you to specify separate dead time values for these. It works just like the primary dead time screen.

7.7.5 Alt Injector small pulsewidths 2

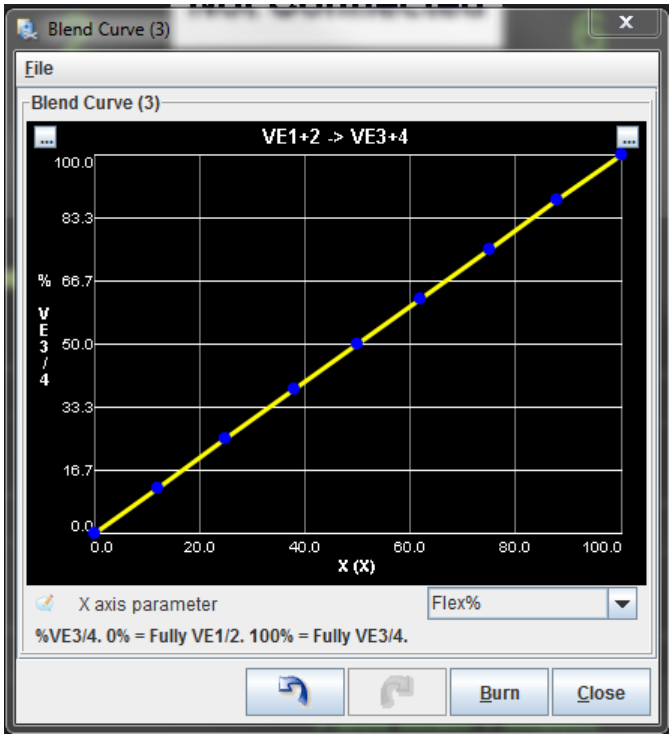
This works like the primary injector small pulse widths, but applies to a second set of injectors when dual fuel mode is enabled.

7.7.6 Blend curve menus

The MS3Pro has six blend curves for blending multiple tables. Each curve has a specific function.

1. Blend curve 1 blends between a primary and secondary fuel algorithm (VE1 and VE2).
2. Blend curve 2 blends between a primary and secondary spark algorithm (Ignition 1 and Ignition 2).
3. Blend curve 3 blends between two fuel tables on the same algorithm (VE1 and VE3). If using a two way blend, blends between VE2 and VE4 as well.
4. Blend curve 4 blends between two ignition tables on the same algorithm (Ignition 1 and Ignition 3). If using a two way blend, blends between Ignition 2 and Ignition 4 as well.
5. Blend curve 5 blends between AFR tables 1 and 2.
6. Blend curve 6 blends between boost tables 1 and 2.
7. Blend curve 7 blends cranking pulse widths. Typically used for E85 flex fuel applications.

The curves all work in the same manner.



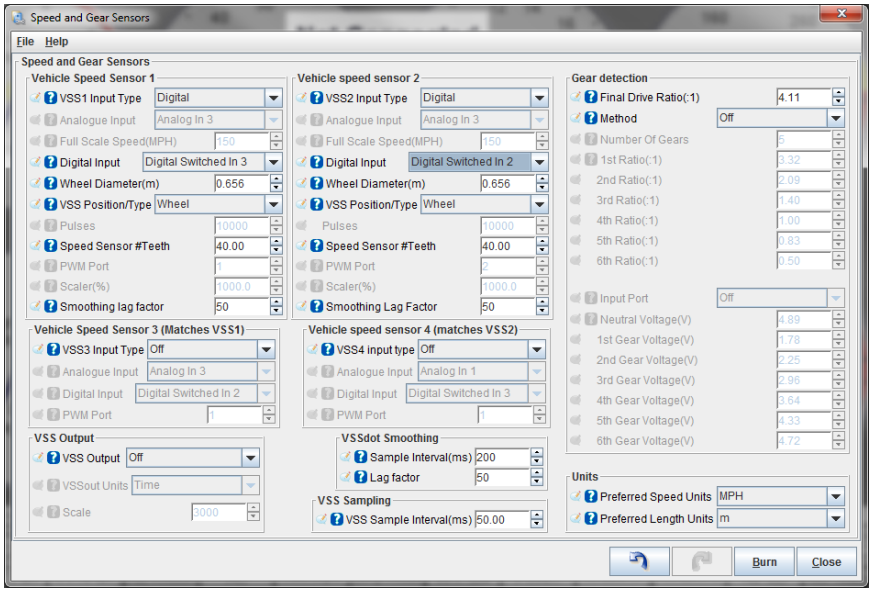
The % is the amount of the second (higher numbered) table to use, with the remaining percent being on the lower numbered table. The X axis can be set to several different types of inputs:

- TPS - Throttle position
- MAP - MAP sensor reading
- RPM
- MAF load
- Flex% (percent ethanol in fuel, as measured by a flex fuel sensor)
- VSS1 - driven wheel speed
- Gear - transmission gear position
- RPM * Load / 100
- Generic sensor inputs

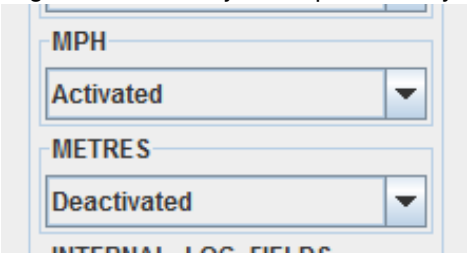
7.8 Advanced Engine

7.8.1 Speed and gear sensors

The vehicle speed sensors (VSS) allow one or two wheel speeds to be monitored. The MS3Pro allows one or two speed sensors, with the second one being used to measure wheel slip for traction control. The inputs read from either a wheel/axle mounted pickup or a perhaps pickup that is part of the transmission. The input can be magnetic (with suitable interface) or more directly a hall type input. This input is converted to a linear speed (mph or kph). The shaft speed inputs work in a similar way to the vehicle speed but calculates to rpms. It could be used as gearbox input shaft speed to monitor converter or clutch slip. The VSS output feature provides a pulsed output in proportion to the VSS for driving a digital speedometer. The gear detection calculates the current gear selected in the gearbox by a number of methods.



You can use the Project Properties screen to set the units, both mph vs kph and meters vs inches, in the Settings tab under Project Properties, or you can change them with the settings in the lower corner of this table.



There are four options for speed input.

1. Analogue linear input where 0V = zero speed and 5V = max speed.
2. Digital pulse input to MS3Pro
3. MS3Pro collects speed value from remote CAN device
4. MS3Pro collects pulse data from remote CAN device

7.8.1.1 Analogue linear input A 0-5V signal is applied to one of the analogue inputs. Some aftermarket speedometers have an analog output that can be used for this purpose. This can also apply to Pitot tube type speed indicators. Set the analogue sensor input channel, the full scale speed, and the smoothing lag factor.

7.8.1.2 Digital pulse input This is most easily achieved with a hall-type sensor applied to one of the frequency or switched digital inputs. Analogue sensor should be set to Off and the 0-5V input pin specified as the Input. Set wheel diameter, VSS position, number of teeth and lag factor. There are many options for mounting a reluctor and sensor on the vehicle; two basic categories exist - wheel mounted and driveline mounted.

Wheel mounted Select this when your sensor is picking up on a shaft spinning at the same speed as the wheels. Possibly the simplest option here is a gear tooth sensor (such as our Hall effect crankshaft position sensors) in close proximity to the back of the wheel studs, or a pre-existing ABS reluctor.

Driveline mounted Select this when your sensor is picking up on a shaft before the final drive gear. e.g. gearbox output shaft, driveshaft. (This is most likely the setting to use for RWD transmission mounted sensors.) Many transmissions already offer a VSS output. If this is suitable, you may be able to just tap into this. Alternatively, you could add a trigger wheel on the output flange and one of our Hall effect crankshaft position sensors.

7.8.1.3 Speed value from a remote CAN device In the case where you have, say, a CAN-connected transmission controller that is already calculating vehicle speed, you can simply configure MS3Pro to collect this number. Analogue sensor should be set to Off and CANVSS specified as the Input. Set scaler% and lag factor. Internally, MS3Pro stores speed in units of 0.1 ms-1 (meters per second). To convert from a remote device using mph, a scaler of 46% would apply. Configure the CANVSS offset/table/id through the CAN menus.

7.8.1.4 Pulse data from a remote CAN device This is similar to option 2, but the VSS connects to a remote CAN device, the time between pulses is collected and MS3 calculates the speed. Analogue sensor should be set to Off and CANPWMin specified as the Input. Set wheel diameter, VSS position, number of teeth, PWM port and lag factor. Configure the CANPWM offset/table/id through the CAN menus.

7.8.1.5 Menu settings

- **Analogue sensor** - If using an analogue sensor (method 1) this is set to the analogue input. Otherwise, set it to Off
- **Full scale speed** - only used with an analogue sensor. Sets the speed reported on a 5V input. Assumes 0V = 0 speed.
- **Input** - specifies either the digital pulse input or CAN input. Not used with analog sensors.
- **Wheel diameter** - only applies to methods pulsed input modes.
- **VSS position** - either wheel or driveline - applies to methods pulsed input. If using driveline, be sure to set your final drive ratio.
- **Speed sensor #teeth** - how many teeth the reluctor has - used with pulsed input
- **PWM port** - the CAN PWM "port" being used - only used with pulsed input over CAN.
- **Scaler** - applies a scaling to the CAN collected speed value, for units conversion. Applies if using CAN.
- **Smoothing lag factor** - Smoothing on the speed value. Lower numbers mean more delay and smoothing. Sensible range is 15-100.

7.8.1.6 VSS dot smoothing These settings control how the VSSdot (acceleration) calculation is performed. The lower the sample interval, the more responsive the calculation but the noisier it will be. Experimentation will be required to get the best settings for your application. Calculating launch acceleration is particularly difficult to get right. Very low tooth counts and wheel speed collected via CAN will be the most difficult due to the low data rate and the communication delays.

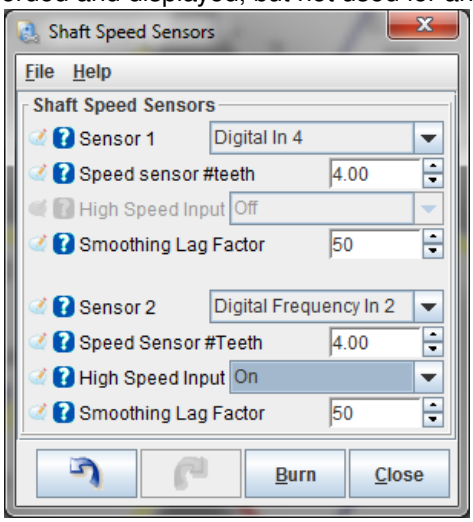
7.8.1.7 VSS sampling The MS3Pro calculates vehicle speed by counting the number of pulses in a sample interval time, separate from (and shorter than) the VSSdot sample interval. Normally 50ms is OK and gives excellent smoothing. Hard accelerating vehicles will need to lower this number for improved zero-speed response.

7.8.1.8 VSS output This allows the MS3Pro to provide a scaled output based on speed sensor #1 to control a speedometer.

- **VSS output -** specifies which output channel to use.
- **VSSout units** - valid units are as follows:
  - Time
  - Pulses per mile
  - Pulses per kilometer
- **Scale -** If “pulses per mile” or “pulses per kilometer” is selected, this is how many pulses to put out per unit of distance. If “time” is selected, the time between output pulses is calculated as time factor / speed in meters per second.

7.8.2 Shaft Speed Sensors

These use digital pulsed inputs to determine speed of other components in the car for data logging. For example, you can log the converter output shaft speed in an automatic transmission. Currently, shaft speeds are just recorded and displayed, but not used for any other calculations.



High speed input mode allows more precision, but is not available on all inputs. If it is available on your input, it is best to turn this on.

This mode allows fractional teeth in case the sensor is driven by a gear or other mechanism that results in a non-integer number of outputs per shaft revolution.

Tip: Want to measure a generic frequency-based sensor? Setting the sensor speed number of teeth to 60 will make the indicated “shaft RPM” equal the sensor output frequency, in hertz.

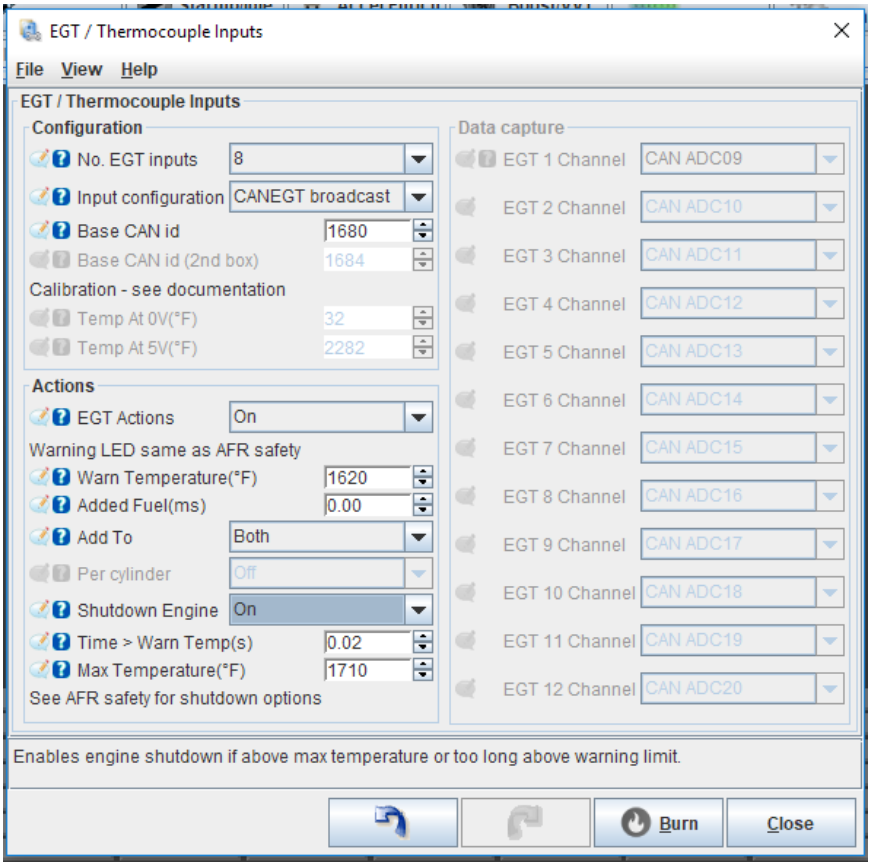
7.8.3 EGT / Thermocouple settings

The EGT system in MS3Pro allows inputs from EGT sensors to be datalogged by serial or SD card. It optionally allows the engine to be shut down if EGT exceeds a set temperature. If an EGT probe is fitted to each exhaust port, it can be used in conjunction with per-cylinder fuel trim to balance cylinders to account for air flow and injector flow differences. EGT is used extensively in piston driven aircraft and much useful information can be found from aircraft literature and then applied to your non-aero engine. (EGT differences between cylinders can be significant at idle. Evidence shows that V8s often run cold on the four corner cylinders and hot on the inner four. One engine required over 10% fuel trim to balance these air-flow differences.)

To use the EGT system an EGT probe (usually a K-type thermocouple) is installed in the exhaust manifold. When installing an EGT per cylinder, it is typical to install ~50mm from the head.

The thermocouple output is a small voltage that the MS3Pro will not read directly. There are two ways you can connect a thermocouple to the MS3Pro:

- An amplified that converts the signal to a 0-5 volt reading which you can connect to an MS3Pro analog input
- A CAN enabled thermocouple interface like the CAN-EGT input module that transmits the data over the CAN bus



- **No. EGT inputs** - specify the number of channels you have connected
- **Input configuration**
  - Manual - set the EGT input channels manually either using CAN receiving or analog inputs.
  - CANEGT broadcast - automatically configures the unit for an AMP EFI CAN-EGT module. The module must be set to broadcast mode.
- **Base CAN ID:** The CAN ID used for a CAN-EGT module in broadcast mode. If using two CAN-EGT modules, the first box will control channels 1 through 8, and the second box will control the remaining channels.
- **Calibration** - the calibration depends on the thermocouple amplifier in use. Consult the documentation that came with your amplifier.
- **Data Capture** - here you specify the local or remote analogue channels that the EGT inputs are connected to. For remote CAN inputs, see the CAN section for configuration. These are automatically configured when using CANEGT broadcast mode.

The Actions section allows you to specify action if EGT gets too high.

- **Warn temperature** - above this temperature the code will take action. The warning output selected on the AFR Safety system is turned on.
- **Add fuel** - specifies a raw pulsewidth in milliseconds of additional fuel. This could be used to help protect the engine if EGT is too high.
- **Add to** - Normally this is set to "Both". When using staged fuel outputs or dual table, you may want to specify a particular channel.
- **Shutdown engine** - if you want, the engine to be shutdown if EGT is too high.
- **Time > warn temp** - if any of the EGT stays above the warn temperature for more that this time, the engine is shutdown.

- **Max temperature** - if any of the EGTs exceeds this temperature, then the engine is shutdown immediately.

See the AFR Safety system for shutdown and resume settings.

The temperatures, pulsewidth and time presented are purely examples. It is very important that you review your datalogs and think carefully before specifying these settings. With correct settings, EGT over-temp is likely to detect a failing injector causing a lean mixture on one cylinder or an overall problem such as a faulty pump or regulator causing everything to run lean.

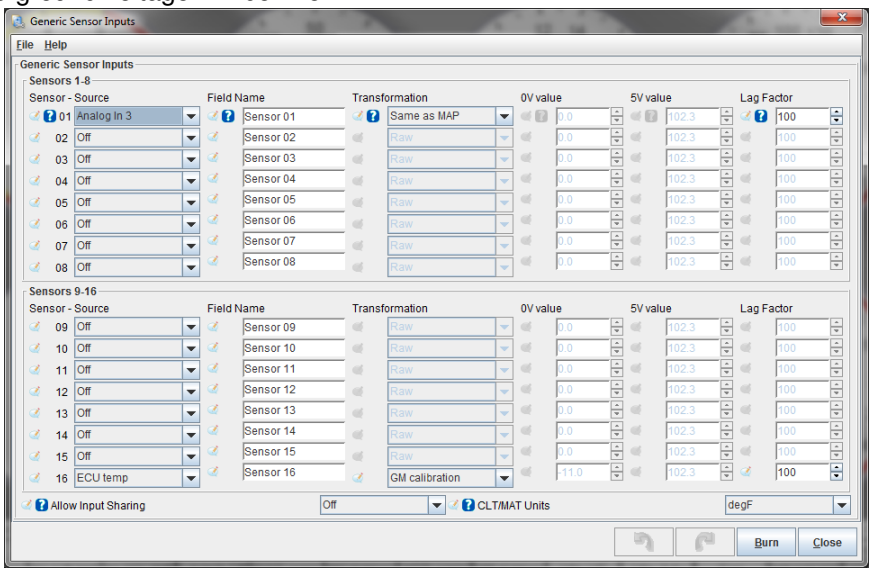
7.8.4 Generic sensor inputs

The Generic Sensors allows optional analogue inputs to be connected and easily recorded in datalogs or used as parameters in the Output Ports. The analogue inputs are in the 0-5V range and are internally converted with a 10bit ADC (Analogue to Digital Converter), giving 1024 steps. Typical uses include:

- Pre- and post- intercooler pressure and temperature
- Fuel pressure
- Nitrous pressure
- Exhaust backpressure
- Pedal position
- Suspension travel

The MS3Pro Mini has three built in 0-5 volt inputs. You can also bring analog sensors in over the CAN bus.

If you're using a potentiometer input, the exact resistance range of the potentiometer is not important as long as it does not draw too much current. Any potentiometer with a 500 ohm resistance or greater between the supply and ground voltage will be fine.



Each sensor channel is turned on individually. Once it is enabled it will appear in datalogs. The channels are named sensor01, sensor02....up to sensor16.

7.8.4.1 Source To enable a channel, change the source from "off" to a specific input. The Analog In 1 through Analog In 3 are built into the MS3Pro. CAN ADC01-24 are analogue inputs collected from a remote CAN connected board(s) and must be configured in the CAN section before use.

7.8.4.2 Field Name You can set a description of the field, to improve readability.

7.8.4.3 Transformation This specifies how the raw 0-1023 number the MS3Pro sees internally should be turned into something meaningful to you. These are your options.

- **Raw** - just the raw 0-1023 number. This is mostly of use for troubleshooting.
- **Linear** - apply a linear conversion from 0V to 5V values. This is appropriate for many sensors
- **Same as MAP** - if you have an identical pressure sensor as the main MAP sensor
- **Same as CLT** - if you have an identical temperature sensor and bias resistor as the main CLT sensor
- **Same as MAT** - if you have an identical temperature sensor and bias resistor as the main MAT sensor
- **Same as EGO** - if you have an identical EGO calibration as the main EGO sensor
- **Same as MAF** - if you have an identical Mass Air Flow sensor as the main MAF sensor calibration
- **GM calibration** - if you have a GM temperature sensor and standard (2490 ohm) bias resistor
- **Production calibration** and **beta calibration** - Used for MS3Pro internal temperature. A few early units use the "beta" setting; all others should use "production".

7.8.4.4 0V, 5V value These are only used with the Linear transformation.

- **0V value** - specifies the number you want to see when 0V is applied
- **5V value** - specifies the number you want to see when 5V is applied

Many sensors specify output at other voltages, e.g. some data sheets advise pressure at 0.5V and 4.5V. Firmware version 1.6 and higher has an option to directly input 0.5-4.5v sensor data, but for earlier firmware versions his needs to be converted to the 0V and 5V values using a calculator.

Take the following numbers:
"A" = low voltage
"B" = high voltage
"C" = low pressure
"D" = high pressure
Calculate E = (D - C) / (B - A)
0V value = C - (E * A)
5V value = (E * 5) + 0V value

Let's work these out with an example sensor: Honeywell's 2000 psi sensor, part number MLH02KPSB06A.

This sensor has a linear (ratiometric) output voltage range of 0.5V to 4.5V from zero to full pressure. It is a 2000psi gauge sensor that would be suitable for measuring nitrous line pressure 0.5V is 0psi 4.5V is 2000psi

Calculate the 0V and 5V values as above. Take A = 0.5V, B = 4.5V, C = 0psi, D = 2000psi

E = (D - C) / (B - A) = (2000 - 0) / (4.5 - 0.5) = 2000 / 4 = 500

0V value = C - (E * A) = 0 - (500 * 0.5) = 0 - 250 = -250 (yes, this is negative) 5V value = (E * 5) + 0V value = (500 * 5) - 250 = 2500 - 250 = 2250

For this particular sensor, you would enter -250 in the 0V box and 2250 in the 5V box. Those values are never actually reached, but this gives the correct calibration.

7.8.4.5 Lag The smoothing factor to apply. This uses the same number system as lags elsewhere. 100% means no smoothing and lower numbers give increased smoothing and lag to the input. 15% is a sensible lower limit.

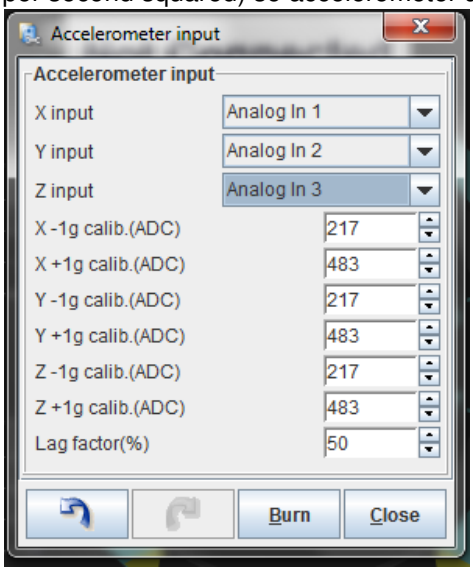
7.8.4.6 CLT/MAT temp units Specify the temperature unit system you want.

7.8.4.7 Allow Input Sharing Normally each input can only be used by one feature. This setting allows inputs to be shared between generic sensors and other features. (Obviously you can still only connect one sensor to each input!) For example, you may want to enable a sensor as an EGT input, but also have TunerStudio display the raw voltage from the sensor.

7.8.5 Accelerometer parameters

The accelerometer input allows for a user-supplied one, two or three axis accelerometer to be connected to the MS3Pro and datalogged. It may be connected to any of the spare analogue inputs or remotely via a CAN.

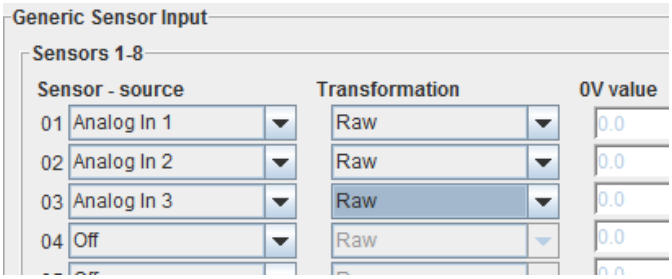
The accelerometer is calibration to +/-1g for convenience, but internally the code uses SI measurement units (m per second squared) so accelerometer acceleration can be compared to vss measured acceleration.



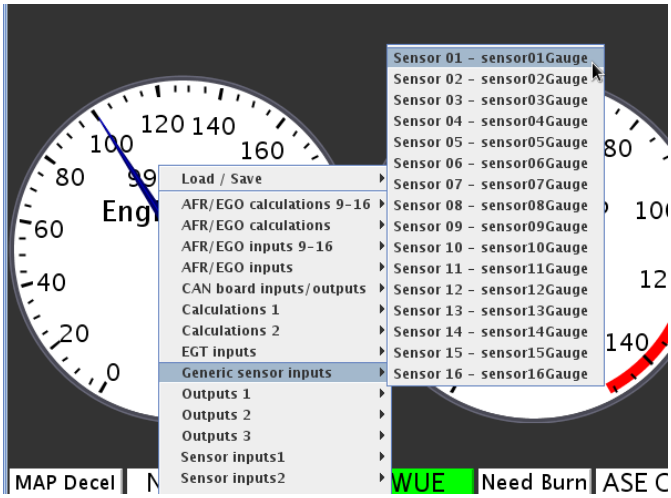
However, **BEFORE** enabling the accelerometer, your need to collect the calibration data.

7.8.5.1 Collecting calibration data In this example we will assume that the three axes are connected to Analog In 1 through 3.

- Ensure that the accelerometer is wired in and powered on.
- Open up Generic Sensors and define three spare "sensors" with the three accelerometer inputs. Set the transformation to “Raw”.



- Next, define a gauge on your TunerStudio dash to match one of these inputs. Right click on a gauge and then re-define it to match one of the sensor inputs you just defined.



- Physically move and rotate the accelerometer until it gives the minimum reading on that gauge. Write it down.
- Repeat for the maximum reading on that gauge. Write it down. The minimum value will be for -9.81ms-2 (-1g) and the maximum will be for +9.81ms-2 (+1g).
- Repeat for all three axes.
- Return to the Generic Sensors screen and turn off the three inputs you defined.
- Go to the Accelerometer dialogue and enable the three inputs and enter the pairs of calibration data you captured.
- The process is now complete, and in future the acceleration will be recorded in datalogs as accelX, accelY, accelZ.

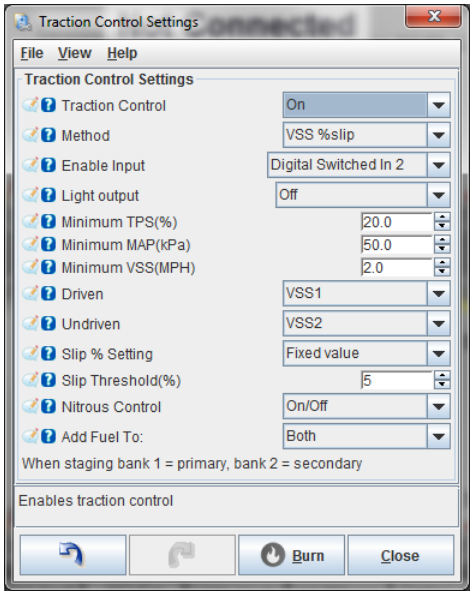
7.8.6 Traction control settings

The MS3Pro offers several strategies for traction control:

- **VSS Slip %:** The MS3Pro compares a speed sensor on the driving wheels to one on the non-driven wheels, and reduces power if the driving wheel speed exceeds the non-driven wheel speed by a specified amount.
- **Perfect Run:** The MS3Pro monitors a single speed sensor. You specify the maximum speed the vehicle can reach in a given amount of time, and the MS3Pro considers any acceleration faster than this to be wheel spin and engages traction control. As this strategy is intended for drag racing, Perfect Run traction control requires starting with launch control active to engage.
- **Perfect Run RPM:** A variant of traction control for racing events which do not allow the ECU to monitor vehicle speed. In this mode, you specify the engine RPM as a function of time. The MS3Pro will engage the traction control if the RPM increases faster than the specified curve. Like regular Perfect Run, this mode requires starting with launch control active before it engages.
- **Switch Input:** The MS3Pro engages traction control in response to a signal from an external traction control module. Used on cars with OEM traction control modules which send an on/off signal, such as some Corvettes.

Traction control allows several different responses to reduce power, depending on what features you are using.

- Retarding the spark timing.
- Using a rolling spark cut.
- Adding fuel.
- Reducing or shutting off a nitrous oxide injection system.
- Reducing boost on a turbocharged motor.



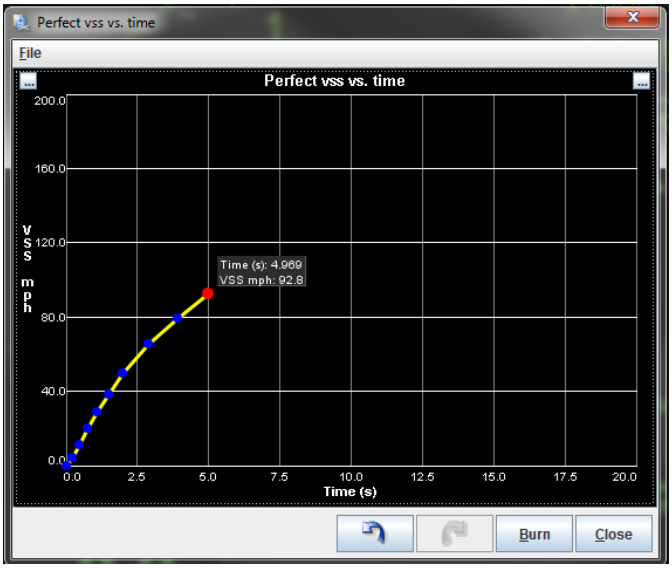
7.8.6.1 Traction control settings

- **Traction control** - Allows turning traction control off or on.
- **Method** - Specifies what method is used to determine when to apply traction control.
- **Enable input** - If set to “off,” traction control is active full time (for VSS %slip) or active any time the launch control system is used (for Perfect Run). This can specify an input pin used to arm traction control. If using switched input, this is the pin used to trigger traction control.
- **Light output** - The MS3Pro can trigger an output pin to switch on a light when traction control is active.
- **Minimum TPS** - Specifies the minimum throttle position below which traction control is not applied.
- **Minimum MAP** - Specifies the minimum MAP reading below which traction control is not applied. (Not used for Perfect Run)
- **Minimum VSS** - Traction control is not used below this speed. (Not used for Perfect Run)
- **Driven** - Specifies which speed sensor or combination of speed sensors to used for the driven wheels.
- **Undriven** - Specifies which speed sensor or combination of speed sensors to use for the non-driven wheels.
- **Slip % setting** - If set to “Fixed value,” you specify the maximum percent the drive wheel speed can exceed the non driven wheel speed with the Slip Threshold % setting. “Adjustable” allows you to use an external dial to change the slip threshold.
- **Slip threshold** - The maximum amount of wheel slip permitted in VSS % slip mode.
- **Nitrous control** - Current firmware only supports on/off. In on/off mode, nitrous is left on if the Nitrous Cut curve is less than 40%, and cut if nitrous cut % is over 60%.
- **Add fuel to** - If using added fuel, specifies if this is applied to the first stage (Bank 1), secondary injectors if used (Bank 2), or both.

7.8.6.2 Three and four wheel speed sensor traction control The VSS % slip requires at least two speed sensors. On a motorcycle, any sensor beyond the first two would be redundant. On a four wheeled vehicle, a typical rear wheel drive application would use a driveshaft speed sensor for the rear wheels compared to a sensor on the right front wheel (since this one is less likely to lift off the ground with normal engine rotation). And a typical front wheel drive drag racing application would use whichever front tire is more prone to spin, compared with either rear tire. However, the MS3Pro offers other options, using three or four sensors.

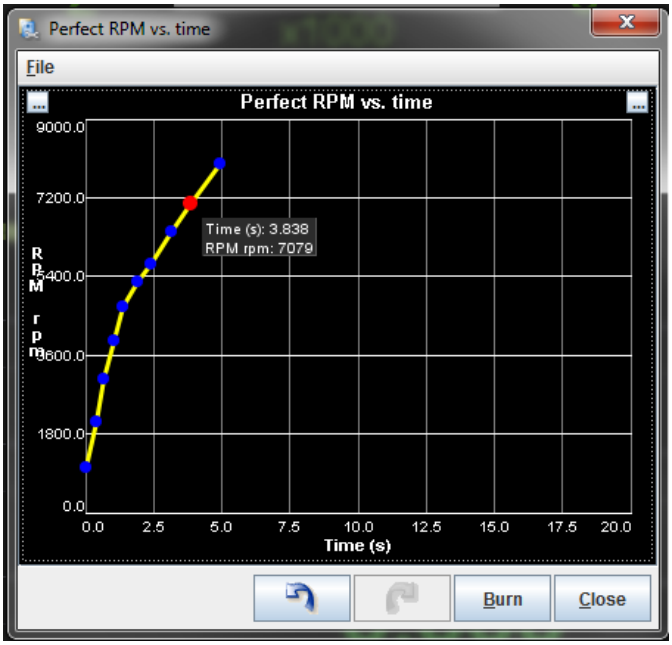
- The MS3Pro can use two wheels on one axle for either driven or non driven wheel speed. The MS3Pro allows you to either average the two sensor readings or to compare both sides and select the higher value. The maximum function can be particularly useful on applications where one wheel is likely to leave the ground, such as many front wheel drive VW vehicles in road race or autocross applications. Note that a driveshaft speed sensor will also create an average value between both wheels on the driven axle, but do so mechanically.
- The MS3Pro can use the average or highest value across all four wheels. This can be used for perfect run VSS mode on all wheel drive vehicles, or compared to a high speed GPS speedometer input for % slip based control.
- Side2Side mode requires both the driven and undriven wheels to use this setting. It compares the wheel-speed between driven and non driven wheels on each side, using the highest value as the percent slip. When configuring speed sensors for side2side mode, VSS1 and VSS3 need to be on the driven wheels. VSS2 needs to be on the non-driven wheel on the same side as VSS1, and VSS4 needs to be on the non-driven wheel on the same side as VSS3. This mode is intended for autocross use and racetracks with sharp corners, where the difference between wheel speed on each side can be considerable.

7.8.7 Traction control - perfect run VSS



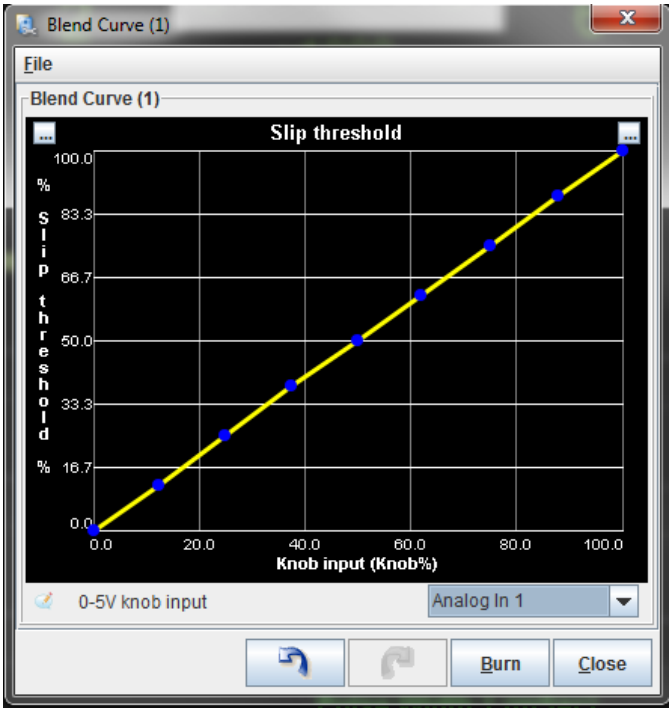
This screen lets you specify the maximum speed for the traction control to allow as a function of the time after launch control deactivates. After passing the top time point, the traction control turns off.

7.8.8 Traction control - perfect run RPM



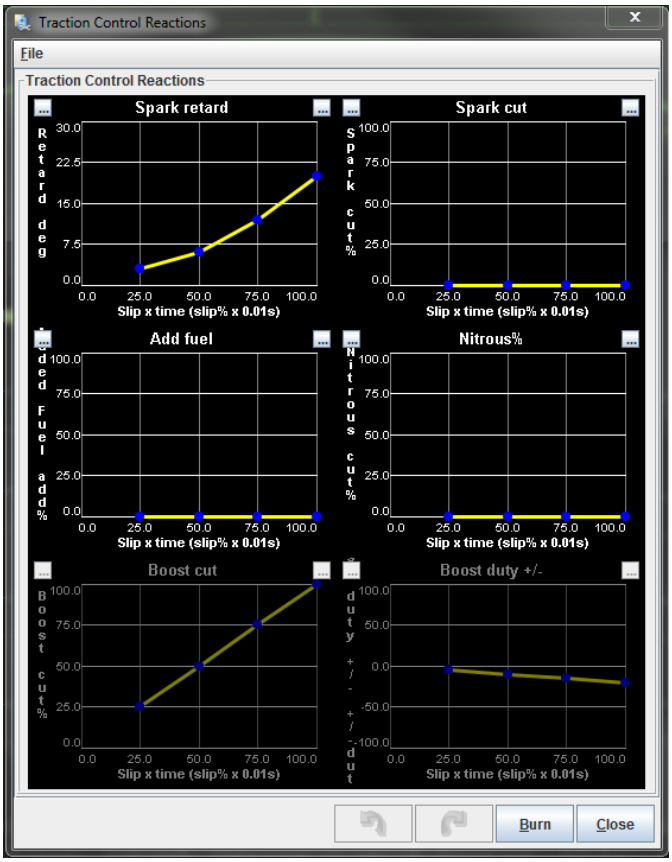
This works in the same manner as the perfect run VSS screen, except you specify the maximum engine RPM to allow as a function of time. This provides a slew control function instead of true traction control, to meet various sanctioning body rules.

7.8.9 Traction control - External % slip input



Allows you to specify the maximum slip percent to allow as function of the amount of input from an external adjustment knob.

7.8.10 Traction control reactions



This screen lets you specify what actions to take if traction control is active. The MS3Pro can be set to take more action the longer the traction control is active. The slip number is different based on the method used.

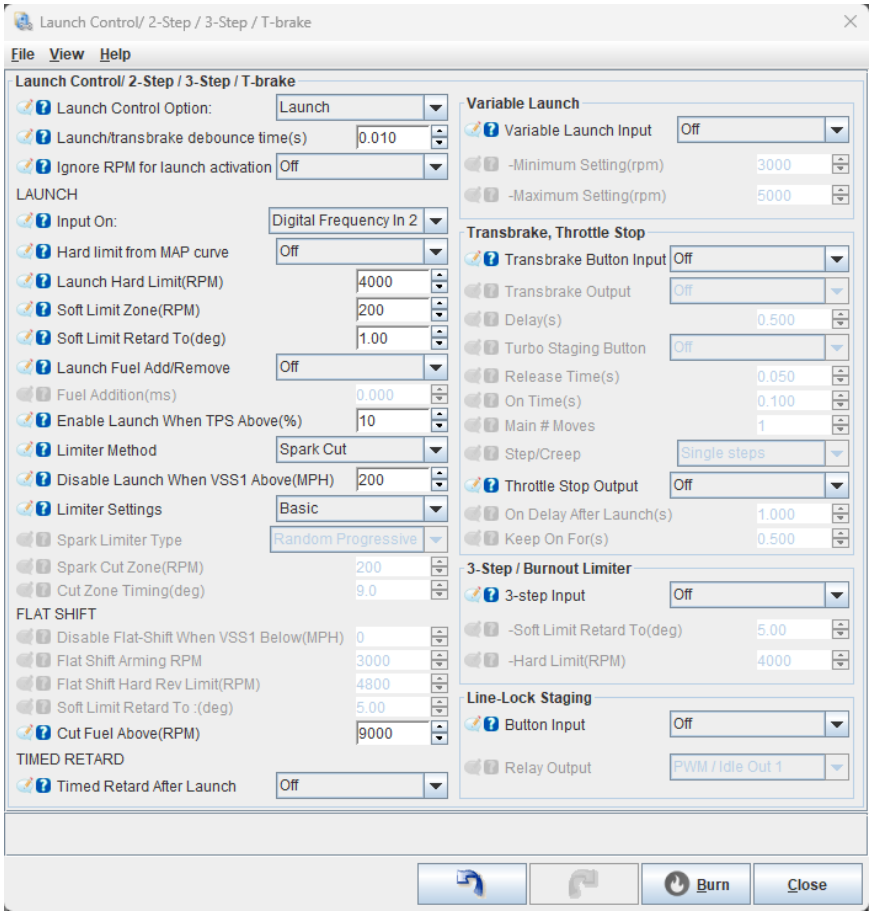
- For VSS% slip, it is the percent by which the driving wheel speed exceeds the non-driven wheel speed.
- For perfect run, it is the percentage difference in acceleration or time vs the target.
- For switched mode, off is 0% slip and on is 100% slip.

With on/off nitrous cut, we recommend setting the nitrous cut at 0 for leaving the nitrous off, and 100% to cut off the nitrous.

7.8.11 Launch / 2 step / 3 step / T-Brake

The MS3Pro launch control system is a 2-step limiter, typically used in drag racing applications for consistent hard launches and/or to build boost with a turbocharged engine. The flat shift settings allow full throttle shifts on a clutched vehicle. A switch on the clutch triggers the flat-shift rev limiting (for bikes, see also the sequential shift cut, which may be more suitable.) It is important to install the clutch switch with adjustability, as the exact height has an effect on how well flat-shifting works. Flat shift is unlikely to be useful when running an automatic transmission. The 3-step is typically used as a burnout limiter and will often be linked to a line-lock.

A typical drag racing scenario would go like this: You pull forward into the burnout box, stand on the brakes and flick on the line-lock. This also enables the 3-step. You now do your burnout and the 3-step limits are applied. After your burnout, you pull into stage, depress the clutch (activating the launch switch) and then push the throttle to the floor. The launch settings will hold the engine close to your chosen rpm. When you change gear, keep the throttle planted, and only use the clutch. Now the flat shift settings come into play, and the revs will be limited to those settings. After changing gear and lifting your foot off the clutch, the limiter is disabled. Be sure to also set the normal rev limiter should you miss shift.



7.8.11.1 Basic launch control / flat shift settings

- **Launch control option**
  - Off - all of launch if turned off
  - Launch - launch only, no flat shift
  - Launch/flatshift - launch and flat shift
- **Launch / transbrake debounce time** - Sometimes a button can “bounce”, acting like it has been pressed several times when it has only been pressed or released once. When the MS3Pro detects a launch or trans brake switch has activated, it waits for this time and then checks the status again to confirm the signal is real and not noise or false triggering. Note: If you have previously tuned the trans brake release time or throttle stop on earlier firmwares that lack this feature, you may need to subtract this number from the trans brake delay time and throttle stop delay.
- **Ignore RPM for launch activation** - If off, then the engine RPM must reach the lower RPM limiter before launch is considered ’active’ allowing timers to start once the button or switch is released. When on RPM is ignored, if button is pressed and other conditions are met, then timers will begin when button or switch is released. This can be useful if you stage incorrectly and don’t bring the engine up to the limiter.
- **Input on** - where the launch input switch is connected
- **Hard limit from MAP curve** - Off allows for a single, static RPM rev limit for launching (see next item below). When on, the launch rev limit is a variable RPM limit based on current MAP (see 7.8.14 - Launch Limit vs. MAP table).
- **Launch hard limit** - when the launch conditions are met, above this rpm the hard limiter methods are used

- **Soft Limit Zone** - When the RPM is this far below the launch hard limit, the MS3Pro will apply spark retard, and if spark cut is used, it will begin to phase in a partial spark cut.
- **Soft Limit Retard To** - This is the timing (in absolute degrees BTDC) that the MS3Pro will command while in the soft limit zone.
- **Launch retard above** - when the launch conditions are met, the timing is retarded above this rpm
- **Retard to** - the absolute timing to retard to. For example, if you enter 5 in here, the timing will be 5 BTDC, -3 means 3 ATDC. Using ATDC timing can be useful for building boost, but beware of the potentially harmful heat it generates in the turbo too.
- **Launch fuel add/remove** - Allows fueling to be changed while the launch limiter is on.
- **Fuel addition** - the raw millisecond pulsewidth change to make (+/-)
- **Enable launch when TPS above** - launch is only active when the throttle is open a certain percentage. Most users can safely set this to a very low number. Setting it too high can cause launch not to activate as expected.
- **Limiter method**
  - None - Launch control itself is not activated. However, this will allow you to use timed features that activate after a launch, such as timed boost control or perfect run traction control.
  - Spark cut - drop sparks; this is the most common limiter used. (Will not work with EDIS ignition or some other semi-autonomous ignition controls.)
  - Fuel cut - cuts fuel. Seen by many as risky due to the possibility of partly fueling, but if your ignition goes into a limp home mode if you try to cut it off, this may be your only option.
  - Spark and fuel - both
- **Disable launch when VSS above** - If you have set up a speed input, this allows you to lock out the launch control when the vehicle is in motion.
- **Limiter settings** - Basic automatically sets the spark limit behavior. Advanced allows you to fine tune this setting.
- **Spark limiter type** - In advanced mode, you can fine tune the spark cut behavior.
  - Random progressive: Applies a random spark cut that cuts more sparks in between the soft zone and hard limit. This is what the MS3Pro uses when the limiter setting is set to basic. This generally applies the smoothest rev limiter.
  - Random ramped: Random cut that ramps up across the spark cut zone.
  - Fixed 5/7: Repeats a pattern of 5 sparks cut and 2 sparks fired. Use the fixed 5/7 or 4/5 if you want a violent, popping and banging sort of rev limiter.
  - Fixed 4/5: Repeats a pattern of 4 sparks cut and 1 spark fired. Do not use on 5 cylinder engines!
- **Spark cut zone** - The range of RPM below the hard limit to apply a spark cut.
- **Cut zone timing** - The absolute timing to run when in the spark cut zone.

Flat shift settings:

- **Disable Flat-Shift when VSS Below** - Prevents flat-shift from activating at low speeds, such as just after launch.
- **Flat shift arming rpm** - this rpm set point is used to differentiate between a flat shift and launch condition. Typically as you are preparing to start a launch you will be at lower rpms, while for flat shift you operate the clutch near your rev limit. So set the flat shift arming rpm to say 3/4 or your max rpms. i.e. consider what rpms are you doing when you press the clutch and make sure the code will always know if you want the launch limits or flat shift limits.

- **Flat shift hard limit** - The maximum RPM to use when flat shift is active. Uses the same soft limit zone as launch control.
- **Soft Limit Retard To** - This is the timing (in absolute degrees BTDC) that the MS3Pro will command while in the soft limit zone during a flat shift.
- **Cut fuel above** - this is intended to be used during flat shifting. During the up-shift as the revs are falling, the fuel is cut off until the revs have fallen below this set point. This reduces the raw fuel in the exhaust system that can otherwise produce a loud backfire.
- **Timed retard after launch** - Allows you to set a spark retard that is active immediately after launch. The MS3Pro will then use a time based curve to put the timing back in. Use this to reduce power off the line to prevent wheel spin without resorting to traction control.

7.8.11.2 Variable launch settings The variable launch mode allows you to adjust your launch control RPM limit with a 0-5 volt input, allowing you to adjust your launch control for changing track conditions.

- **Variable launch input**
  - Off - disables the feature and uses the single limits
  - One of the analogue inputs - indicates what input pin is being used for the variable launch
- **minimum setting** - With a 0V input, the launch hard limit is set to this rpm.
- **maximum setting** - With a 5V input, the launch hard limit is set to this rpm

7.8.11.3 Transbrake and throttle stop The MS3Pro can control many timed delay functions for bracket or drag racing. The trans brake control applies the trans brake when you press the trans brake input button. When you release the button, the trans brake will remain on for the delay time, then release. Launch control is enabled while the trans brake is on, and releases at the same time as the transbrake. The throttle stop output switches on after launch, stays on for a set time, then releases.

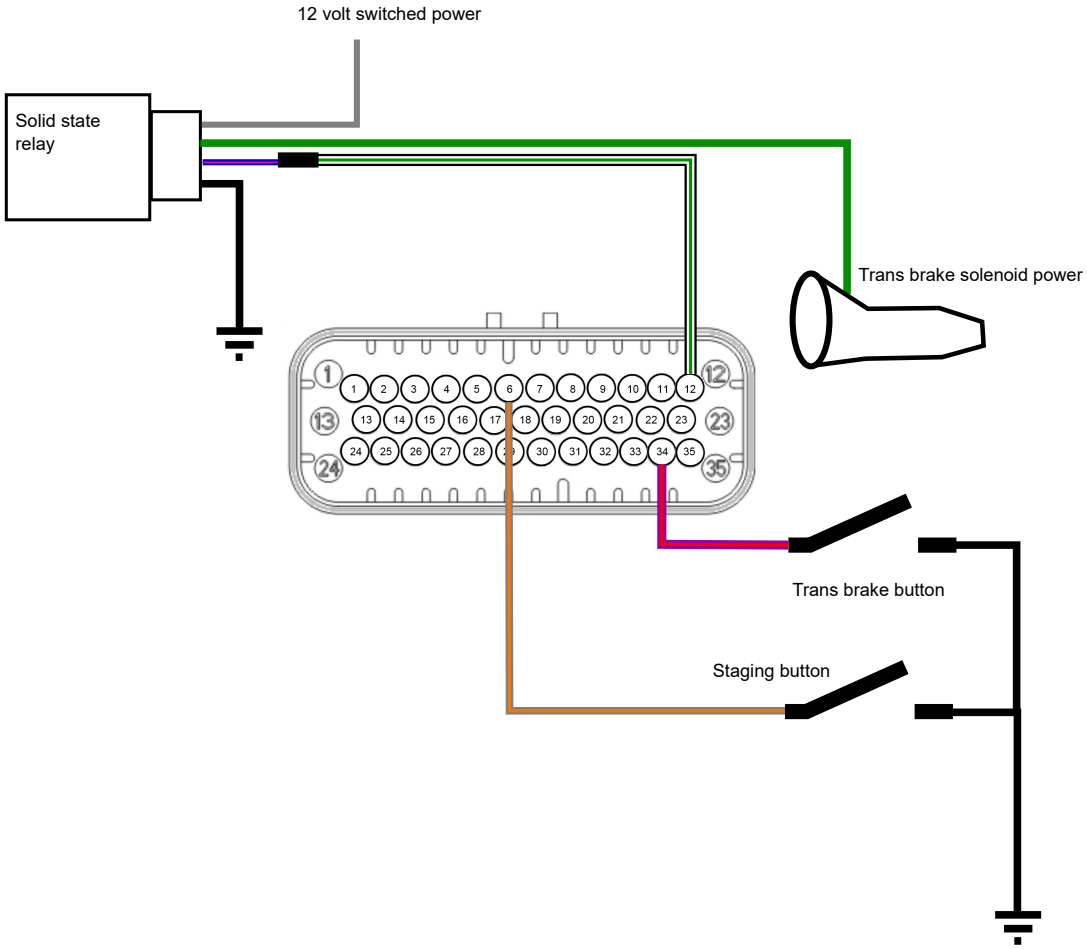
Note: As most trans brakes will not operate without transmission line pressure, the trans brake **will not engage unless the engine is running.** If you are testing your trans brake wiring without starting the engine, use output test mode instead of testing it using the trans brake switch and throttle.

- **Transbrake button input** - Specify what input pin is used for the transbrake switch. Set to “Same as launch” to share this input with the launch control.
- **Transbrake output** - Specify what output pin is used to control the transbrake solenoid.
- **Delay** - The time, in seconds, between releasing the transbrake input and releasing the transbrake.
- **Turbo Staging Button** - Press this button, and the MS3Pro will pulse the trans brake off and on, allowing the car to creep forward while building boost.
- **Release Time** - How long to release trans-brake for on each move step. This should be set long enough to move the car a little without gaining any speed.
- **On Time** - How long the trans-brake should be re-activated for between each step. This should be set long enough to fully grab again.
- **Main # Moves** - How many moves (trans-brake release steps) on the first press of the button. Each subsequent press of the button gives a single move in step mode.
- **Step** / **Creep** - In step mode, each press of the button causes the trans brake to release and then stop. In creep mode, the trans brake will continue its cycle of releasing and re-engaging as long as you hold down the staging button, allowing a continuous creep forward.
- **Throttle stop output** - Specify what output pin is used for a throttle stop.
- **On delay after launch** - The time, in seconds, between releasing launch control and the throttle stop turning on.

- **Keep on for** - The time, in seconds, to keep the throttle stop active.

If using the trans brake without the staging feature, you can drive it with a conventional relay. A solid state relay will give more precise release timing, and you will also need a solid state relay if using the turbo staging button feature in order to pulse the trans brake off and on rapidly enough.

The wiring example on the next page shows Digital Frequency In 2 as the trans brake button, Digital Switched In 1 as the staging button, and Injector Out F as the solenoid control output. This is just one example. You can use any unused digital switched or frequency input for the button inputs and any available PWM or injector output for the solenoid control.



MS3-Pro Mini Wiring: Sample trans brake

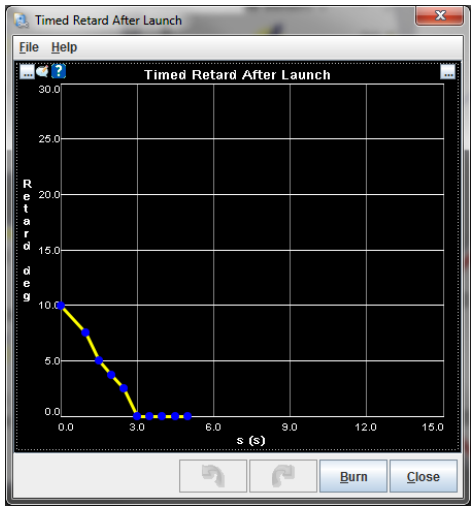
7.8.11.4 3 step / burnout limiter This is another switch operated limiter that works similarly to launch. It could be used as a burnout limiter and the input switched along with your line-lock.

- **3 step input** - Allows you to turn this feature off or specify an input pin to activate the 3 step.
- **Soft Limit Retard To** - the absolute timing to retard to.

- **Hard limit** - when the 3-step is on, above this rpm the hard limiter methods are used

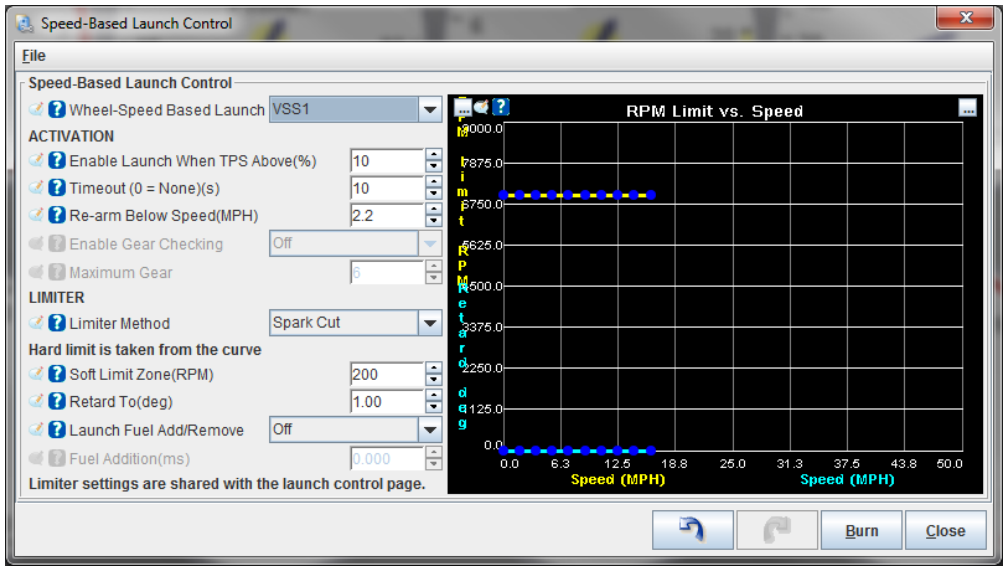
7.8.11.5 Line lock staging When drag racing a car with a manual transmission, sometimes it seems like it would be easier if you had three feet and three hands. Line lock staging automates brake release on a car with a line lock. The MS3Pro will engage the line lock using the selected output when you trigger the button input with a momentary switch. The line lock releases if you push the button a second time or when the launch control disengages.

7.8.12 Timed retard after launch



If you have enabled “Timed retard after launch” in the Launch Control screen, this screen allows you to adjust the timing immediately after launch. This curve specifies how much timing to subtract from the regular timing curve, in degrees, as a function of the seconds after the launch control has been released.

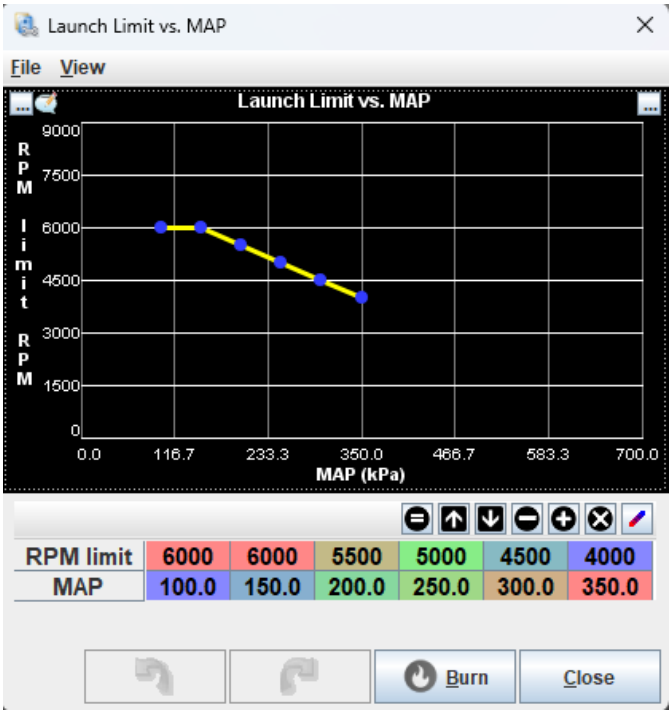
7.8.13 Speed based launch control



Speed based launch control allows for using launch control on vehicles without a clutch switch and also allows for a gradual release of the launch control as the vehicle starts moving. In this case, the MS3Pro releases the launch control when it detects vehicle speed instead of at the point of releasing the clutch or trans brake. Once the vehicle starts moving, the speed based launch control follows a programmable curve for the rev limiter (useful for reducing power in case of wheel spin) and a timing retard vs mph curve to allow for power management off the line. The VSS based launch control deactivates once you reach the highest vehicle speed defined by the curve shown on the right-hand side of the screen.

- **Wheel speed based launch** - Turns this feature on and specifies which VSS input to use. This speed input should be on the non-driven wheels.
- **Enable launch when TPS above** - Specifies the minimum throttle position to use this feature.
- **Timeout** - Once you have been moving for this number of seconds, VSS based launch control is locked out until next stop. Set to zero to make it permanently active (no timeout.)
- **Re-arm below speed** - Below this speed the wheel speed launch is re-armed. Typically set to a very low number so as to re-arm the feature for a standing start.
- **Enable gear checking** - Allows speed based launch control to only engage in certain gears.
- **Maximum gear** - If gear checking is enabled, this is the highest gear you can use this setting.
- **Limiter method** - Allows using a fuel and/or spark cut for speed based launch. The RPM limit and amount of spark retard is taken from the curve at the right.
- **RPM limit** - Set the RPM limit at zero mph to the rev limit you want to use at launch, and typically would hold that same limit until you get to the vehicle speed where the clutch is fully engaged and no longer slipping or torque converter slip is minimal. Past that point, the RPM limit should be only slightly higher than what the engine would reach at that speed without wheel slip.
- **Retard to** - The absolute timing to use when at the rev limit. Below the rev limit, the MS3Pro uses the green curve for timing retard.
- **Launch fuel add** / **remove** - Allows adding pulse width (specified by Fuel Addition) when on the launch limiter. Entering a negative value in Fuel Addition removes fuel.

7.8.14 Launch Limit vs. MAP

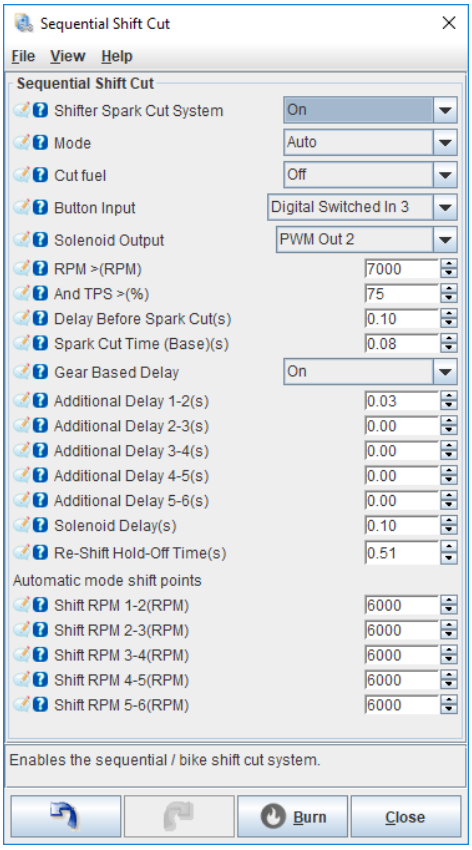


MAP based launch control limit allows for a variable rev limit when launch control is active based on how much boost (MAP) is built up prior to launching.

7.8.15 Sequential shift cut

Another drag racing feature, this one is intended for use with air shifters and sequentially shifted transmissions. The MS3Pro can cut the ignition during shifting and also control the air shifter solenoid valve. Note that automatic shift control and gear based delays require setting up the gear input under Speed and Gear Sensors. The shifting will happen in this sequence:

1. As soon as the button is pressed or automatic shift is required, the air shifter solenoid is turned on.
2. The MS3Pro pauses for a time defined by the shift cut delay before cutting spark.
3. The MS3Pro waits for the specified spark cut time plus any gear based delay, if applied. During this point, the solenoid turns off before the ignition resumes.
4. The ignition resumes firing.
5. The MS3Pro starts the re-shift hold off timer. The MS3Pro will not shift again until this timer runs out.



- **Shifter Spark cut system** - Toggles this feature off or on.
- **Mode**
  - Button - The shift cut is activated by an external button.
  - Auto - The MS3Pro controls the shifting using preset RPM points. Pressing the button will force an upshift.
- **Button input** - Select the channel for the switch to activate the shift cut. Usually, this button is in the shifter mechanism itself.
- **Cut fuel** - Turning this on cuts fuel as well as ignition during upshifts.
- **Solenoid output** - Select the output pin for an air shift solenoid.
- **RPM > (RPM)** - The shift cut will only operate above this RPM.

- **And TPS > (%)** - The minimum TPS reading to activate the shift cut.
- **Delay before spark cut** - The time, in seconds, between the input shift and cutting the spark. This is to allow the solenoid to start moving.
- **Spark cut time (base)** - The amount of time, in seconds, for the spark cut to be applied.
- **Gear based delay** - Turning this on allows you to add an extra amount of spark cut time in one or more gears.
- **Additional delay settings** - These add a small number of extra spark cut for the specified shift
- **Solenoid delay** - the time to allow the solenoid to deactivate before re-applying spark
- **Re-shift hold off time** - Disables the shift cut for the specified time after shifting, to avoid accidentally triggering.
- **Shift RPM** - These five points let you specify the RPM for each shift if running in automatic mode.

7.8.16 Nitrous system

Nitrous oxide injection is primarily used in racing applications where nitrous oxide and additional fuel are injected into the intake to give a horsepower boost. This is particularly popular in drag racing. Used correctly, nitrous can give perhaps 100% additional power. Used incorrectly, it can melt pistons and destroy engines in seconds. The nitrous system within MS3Pro allows control over the nitrous and optional fuel solenoid to enable the nitrous injection when desired and also retard the ignition timing. The system supports two stages of control with on/off or progressive.

There are two main categories of nitrous systems - wet and dry. You can further subdivide this into on/off, progressive, and multistage nitrous systems.

7.8.16.1 Wet flow nitrous A wet system injects fuel into the manifold ahead of the injectors. This can result in fuel distribution problems and mixture varying between cylinders. The simplest example of a wet system is a "plate" system that installs under the throttle body (or carb) above the intake manifold and delivers nitrous and fuel through spray bars. Other variations exist which use single or multiple nozzles. A standalone wet nitrous system has the advantage that when used with the suppliers suggested jetting it will typically install and work. More hardware, less tuning.

7.8.16.2 Dry nitrous With a dry system, the fuel injectors supply additional fuel when the nitrous is active. Systems designed to work with OEM engine management sometimes do this by increasing fuel pressure, which does not give accurate fuel increase across the rev range, but with MS3Pro, we can directly increase the injector pulse width instead. The nitrous enters through a nozzle typically located upstream of the throttle body.

In a port-injection system, a dry nitrous system minimizes the fuel distribution problems. A dry system can also have fewer system components, as it uses the existing fuel system components. However, because the fuel and nitrous can be controlled independently, the software settings are critical to getting the mixture correct. Less hardware, more tuning.

7.8.16.3 On/Off With On/Off control, the nitrous/fuel solenoids are either on or off. This is the standard operating mode.

7.8.16.4 Progressive control The Progressive system allows you to pulse you solenoids with output duty varying with rpm or time during drag pass. It is more complicated to setup than the On/Off system.

7.8.16.5 Multistage and progressive nitrous **Note!** - you are advised to always start with a simple system using small jets. Big hits with multiple stages or progressive are significantly more difficult to install and tune correctly.

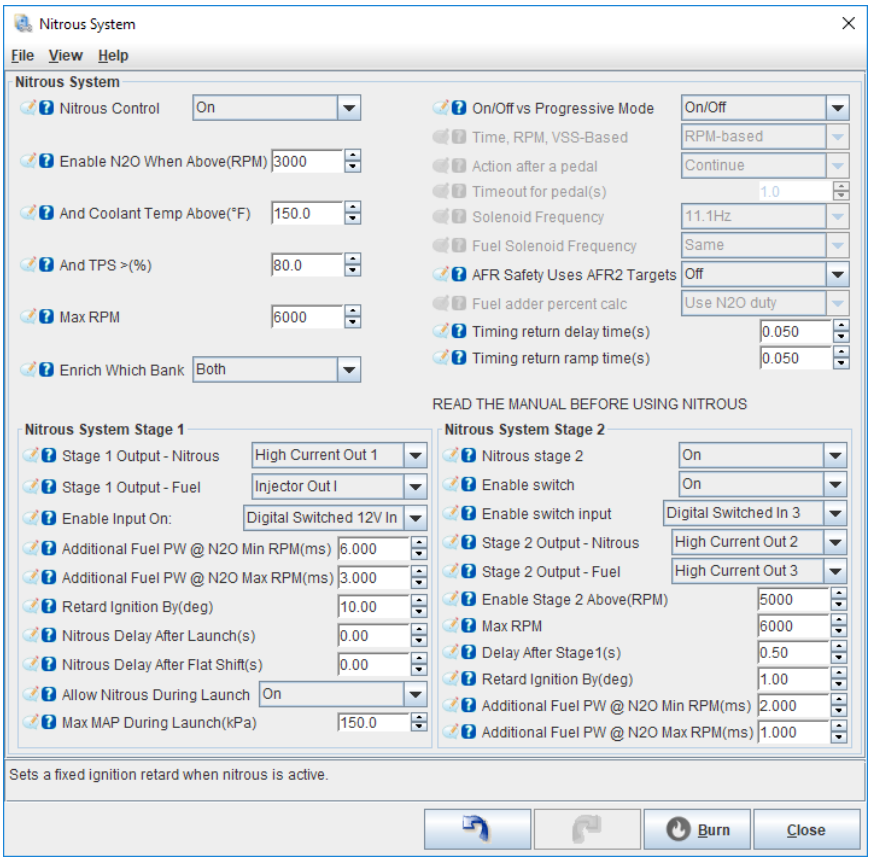
With a single on/off stage using fixed jetting, the flow of nitrous (and fuel) is approximately constant under all conditions. This means that if you are getting a claimed 100hp at 3000rpm when you activate the system, by 6000rpm the engine is turning twice as fast and each intake event will only receive half the nitrous/fuel, so you will get half the torque benefit. One answer might be to use two stages based on rpm. The first stage is still jetted at 100hp, but at 4500rpm you bring in a second stage of say 25hp. This will make up for some of the "lost" torque. Another approach would be a single stage progressive system using "200hp" jets. You could start at ~50% duty at 3000rpm and ramp up to 100% duty at 6000rpm. This could keep a similar torque gain across the rev range.

7.8.16.6 Tuning considerations In general, applying up to 50% extra hp to your engine with a kit manufacturer's jetting and suggested retard works well and will not cause problems to your engine. However, it is very important to take plug readings and check for any signs of leanness or detonation. As mentioned already, it is quite easy to destroy your engine with nitrous given the wrong combination.

Easy mistakes to make with bad consequences:

- Mixed up fuel/nitrous jets.
- Fuel pressure too low (mismatched to fuel jet size)
- Incorrect dry fuel pulsewidth
- Insufficient timing retard
- Too hot a spark plug
- Faulty fuel solenoid or wiring
- Blocked fuel lines or jets
- System activated at too low an rpm

7.8.16.7 Nitrous settings



- **Nitrous Control** - Turns the system on or off
- **Enable N2O when above** - sets the minimum RPM to enable nitrous.
- **And Coolant Temp above** - sets the minimum coolant temperature to enable nitrous.
- **And TPS >** - sets the minimum throttle position to enable nitrous (Typically, nitrous should only be enabled on a warmed up engine at wide open throttle at an RPM point where the engine is making at least as much horsepower as the nitrous shot provides.)
- **Max RPM** - sets the maximum RPM to enable nitrous. You will want this off before the rev limiter engages.
- **Enrich which bank** - Set to "Both" for most installs. If "dual table" or Staging are in use it, selects whether the primary injectors (Bank 1), secondary (Bank 2), or both banks receive additional fuel.
- **On/Off vs Progressive mode** - Selects between On/Off or Progressive control mode.
- **Time-based, RPM, VSS based** - For progressive, chooses between time, RPM, or wheel speed as control factor.
- **Action after a pedal** - If you lift off the throttle far enough to fall below the minimum TPS for nitrous, this determines if the progressive nitrous timer should restart from zero or resume at the point when you lifted.
  - **Timeout for pedal (s)** - If you return to above the minimum TPS before this many seconds, the progressive nitrous output will continue using the original timer. If you do not return to full throttle before this time, the MS3Pro will reset the nitrous timer.
- **Solenoid frequency** - For progressive, selects nitrous solenoid frequency.
- **Fuel solenoid frequency** - Allows specifying a separate frequency for the fuel solenoid.
- **AFR safety uses AFR2 targets** - Allows optional AFR safety system to use alternate AFR targets when nitrous is active.
- **Fuel adder percent calc** - When using progressive nitrous in a time based or VSS based mode, the added fuel also needs to be scaled based on RPM. The MS3Pro will scale the “additional fuel PW” settings based on either the nitrous duty cycle table or a separate fuel percentage table. For example, if the additional fuel pulse width at minimum RPM is 6 ms, and the percentage table gives 75%, the added fuel pulse width will be cut back to 4.5 ms.

Stage 1 settings:

- **Stage 1 output - nitrous** - Selects output pin to use for stage 1 nitrous solenoid
- **Stage 1 output - fuel** - Selects output pin to use for stage 1 fuel solenoid (These will normally be the same for an on/off system, but may be different for progressive.)
- **Enable input on:** - Selects the input pin to use to enable the nitrous system. We recommend using the digital 12 volt input pin for nitrous. If you’re using a conventional arming switch, this allows you to wire the input to the nitrous solenoid power source.
- **Additional Fuel PW @ N2O min rpm** - In on/off mode with a dry system the fuel PW to add at the min rpm. In progressive mode, this value is scaled down by either the nitrous duty curve or the fuel percentage curve, depending on how you have set the “fuel adder percent calc” setting.
- **Additional Fuel PW @ N2O max rpm** - In on/off mode with a dry system the fuel PW to add at the max rpm. When using progressive nitrous, this is scaled in the same way as the minimum RPM fueling. (In between these two rpm points, the MS3Pro calculates the required pulsewidth using linear interpolation.)
- **Retard Ignition by** - In on/off mode sets a fixed ignition retard.
- **Nitrous delay after launch** - Sets a minimum time after launch control before nitrous can activate.
- **Nitrous delay after flat shift** - Likewise for flat shift.

- **Allow nitrous during launch** - Allows turning nitrous stage 1 on while launch control is activated to use nitrous to build boost on a turbo car.
- **Max MAP during launch** - The nitrous system will turn off if MAP goes over this value and launch is active.

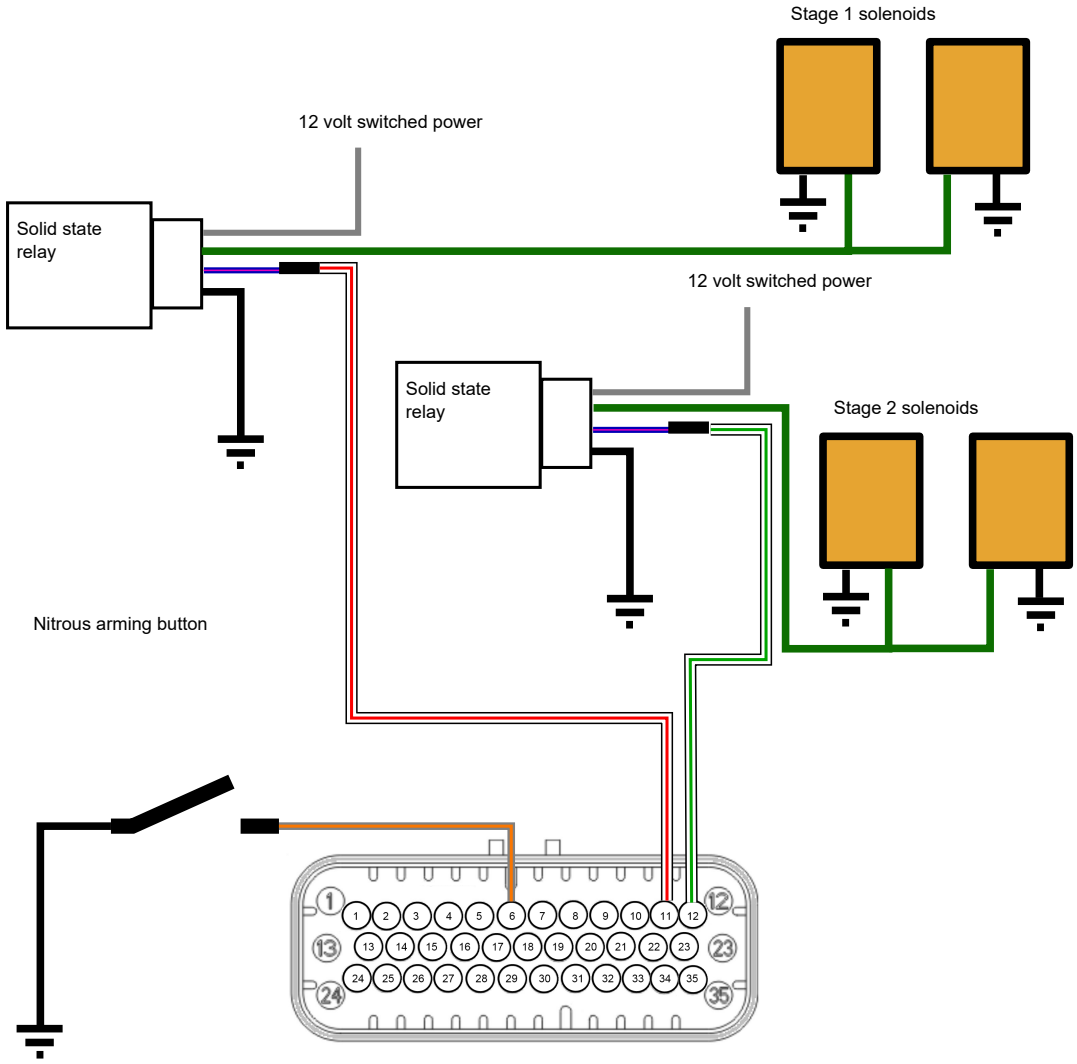
Stage 2 settings:

- **Nitrous stage 2** - Turns the second nitrous stage off or on.
- **Enable switch** - Allows using a separate switch to enable or disable the second nitrous stage. If this is set to off and the nitrous stage 2 is set to on, the second stage is armed any time the first stage is armed.
- **Enable switch input** - Selects the input pin to enable the second nitrous stage.
- **Stage 2 output - nitrous** - Selects output pin to use for stage 2 nitrous solenoid
- **Stage 2 output - fuel** - Selects output pin to use for stage 2 fuel solenoid
- **Enable stage 2 above** - In on/off mode, sets the activation rpm for stage 2.
- **Max rpm** - In on/off mode, sets the maximum rpm for stage 2.
- **Delay after stage 1** - Sets a minimum time delay after stage 1 activates before stage 2 can activate. (To prevent a massive hit.)
- **Retard Ignition by** - In on/off mode sets a fixed ignition retard for stage 2.
- **Additional Fuel PW @ N2O min rpm** - In on/off mode with a dry system the fuel PW to add at the min rpm.
- **Additional Fuel PW @ N2O max rpm** - In on/off mode with a dry system the fuel PW to add at the max rpm.

The progressive feature adds another level of control and complexity. The solenoid duty, added fuel and timing retard for one or two stages can be plotted out against rpm or time giving you fine control over nitrous operation. You'll need to experiment to determine the best duty cycles. The way most solenoids work, 50% duty is rarely equal to 50% flow. You will likely find that fuel and nitrous solenoids vary in behavior too (due to different coils and liquid pressures, among other factors.)

As always, experimentation and careful testing, datalogging and refinement will be required to make the most of this feature.

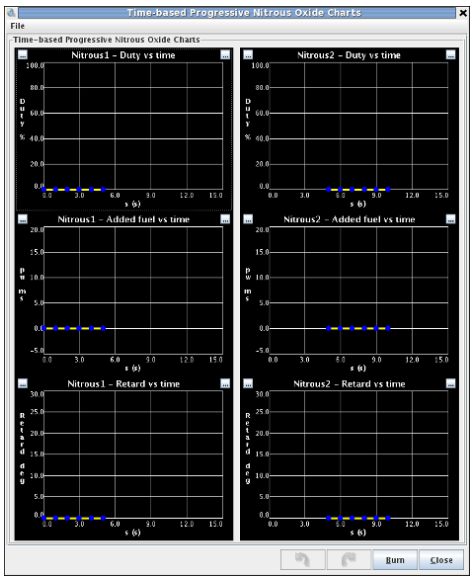
Progressive nitrous control will require using solid state relays to control the solenoids (see section 4.11.1 for more information on solid state relays). The diagram below shows a nitrous system armed with a button connected to Digital Switched In 1, using Injector Out E for the first stage and Injector Out F for the second stage.



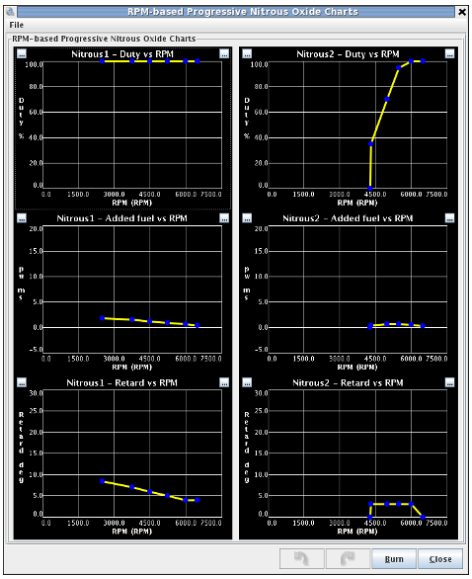
MS3-Pro Mini Wiring: Progressive nitrous

7.8.17 Nitrous - time based progressive

This requires the use of the MS3Pro launch control system and the time is measured from the release of launch. This is only really of use in drag racing. This allows you to specify the amount of nitrous duty cycle, fuel (in raw pulse width numbers), and timing retard (in degrees) for both stages of a progressive nitrous system.



7.8.18 Nitrous - RPM based progressive



These curves work in the same manner as the time based progressive curves, but use RPM for the X axis instead of time.

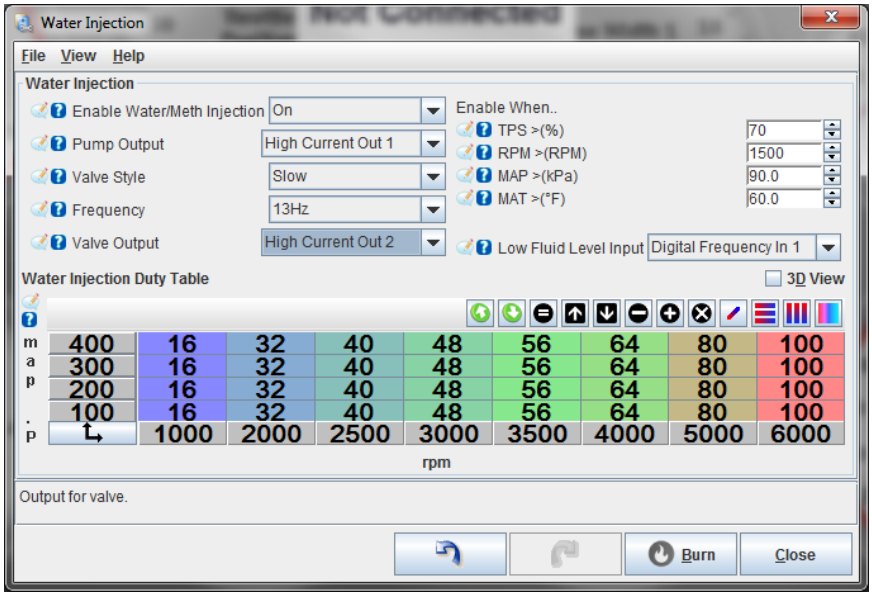
7.8.19 Nitrous - VSS based progressive

These curves work in the same way as the above two curves, but use speed instead of RPM or time.

7.8.20 Water injection

Water/meth injection is primarily used in boosted applications for charge cooling to reduce the tendency to knock, often allowing boost to be increased. Due to limited capacity of water/meth reservoirs this will mainly be of use for short duration sprints and less of use for endurance racing. (Make sure the reservoir does not run empty! The MS3Pro can use a level switch to check this.)

The MS3Pro system can control a pump via a relay when conditions are met and map the duty cycle of a water valve on a 3D map of duty vs MAP vs RPM. The valve may either be fast acting (runs at injector speed) or a slow speed similar to a boost solenoid.



- **Enable water/meth injection** - enables or disables the feature.
- **Pump Output** - output pin to control pump relay.
- **Valve style** - select
  - "Off" - no valve output (used fixed nozzle)
  - "Slow" - slow speed valve output with chosen PWM duty cycle
  - "Fast" - valve switches at injector speed with chosen PWM duty cycle
- **Valve Output** - output pin to control water valve.
- **Frequency** - selects the output frequency to the valve. (Slow mode only.)

Enable when...

- **TPS >** - enable when TPS is above this %age
- **RPM >** - enable when RPM is above this value
- **MAP >** - enable when MAP is above this value
- **MAT >** - enable when MAT is above this value
- **Low fluid level input** - This allows using a level switch on an input pin connected to the MS3Pro. The switch must be wired so it grounds the input when the fluid level is OK and un-grounds the input when the fluid falls below an unsafe level. If the level switch sends a low level signal when the MS3Pro needs to enable water injection, the MS3Pro will behave as if it has been shut down from the AFR Safety feature.

The 3D table controls the valve duty cycle as a function of RPM and MAP sensor readings.

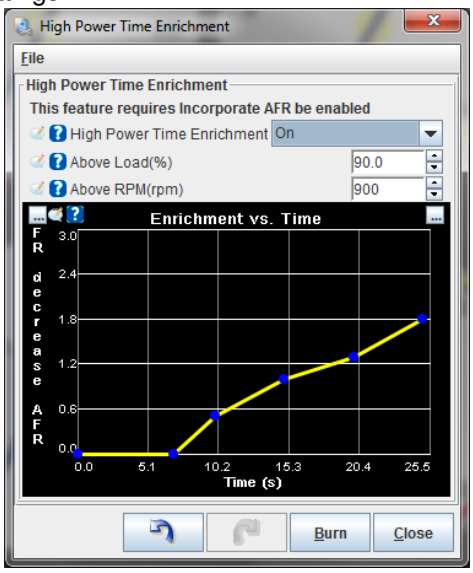
Water injection will almost certainly alter your reported AFRs, and you should therefore ensure that closed-loop EGO control is not operating when water injection is active. Set the EGO control so that open-loop mode is used at the boost levels where you plan to enable water injection.

7.8.21 High Power Time Enrichment

Engines that run flat out for long periods of time often require a richer air/fuel ratio than what would make best power. This is needed to cool down the combustion chambers and reduce the vulnerability to detonation. MS3Pro allows ramping up the air/fuel ratio during sustained full throttle operation so that you can tune for a maximum power air/fuel ratio, then add fueling if the engine stays under power. Applications that can benefit from this include:

- Road race vehicles
- Boats
- Air cooled engines

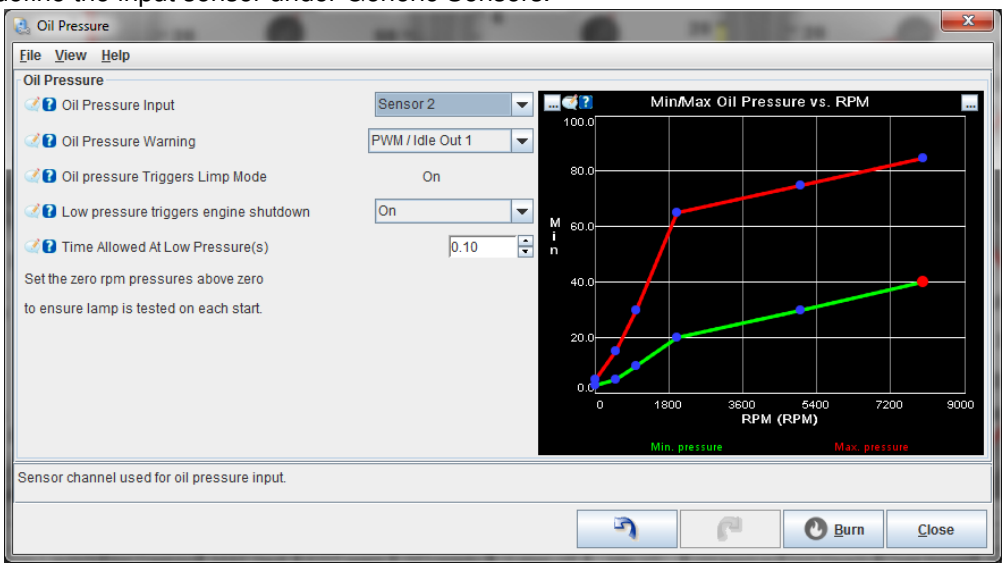
To use this feature, you will need to turn on Incorporate AFR Target under Basic / Load Settings -> General Settings.



You can specify the minimum load and RPM to apply this feature. Once the engine is running in this range and this feature is active, it will adjust the air/fuel ratio as a function of time until the engine drops below the active load or RPM range. The curve specifies AFR decrease (richer) vs. time. It is normal for the first row to be 0, 0. The second row typically specifies when the enrichment should begin. e.g. 7, 0. This means that there is no change from 0-7 seconds as both are set to a 0 decrease in AFR. The third row onwards specify the richening.

7.8.22 Oil Pressure

The MS3Pro can monitor oil pressure and trigger a smart warning light. This feature can also be used to trigger the check engine light. Setting the minimum pressure to more than zero on startup will ensure that the oil pressure warning light turns on when you turn the key on, so you can make sure it is working. The light will come on if the oil pressure goes above the maximum pressure vs RPM curve or below the minimum pressure curve. You will need to define the input sensor under Generic Sensors.

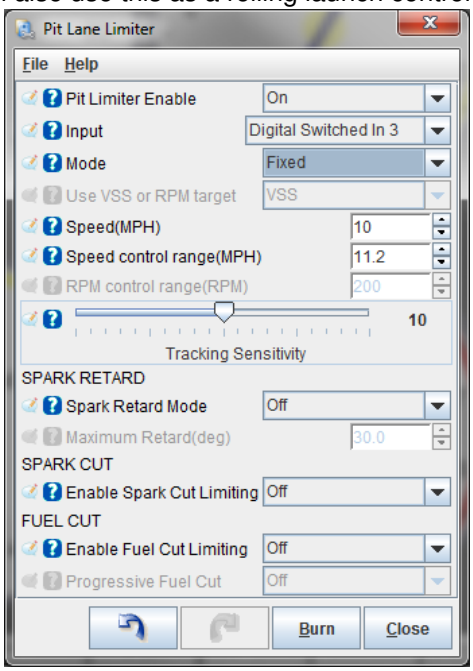


- **Oil pressure input** - Specifies which sensor configured under Generic Sensors is used for oil pressure.

- **Oil pressure warning** - Allows you to use an output pin to trigger a warning light if pressure is either too high or too low.
- **Oil pressure triggers limp mode** - This is actually set on the limp mode settings under CAN bus / test modes, but displayed on this page as well.
- **Low pressure triggers engine shutdown** and **Time allowed at low pressure** - If the oil pressure falls below the minimum pressure for longer than the allowed maximum time in seconds, the MS3Pro will shut down the engine in the same way as specified on the AFR Safety screen.

7.8.23 Pit lane limiter

This allows you to apply a speed limiter in the pit lane of a road course as long as an input switch is activated. You can also use this as a rolling launch control for racing events that use a rolling start.



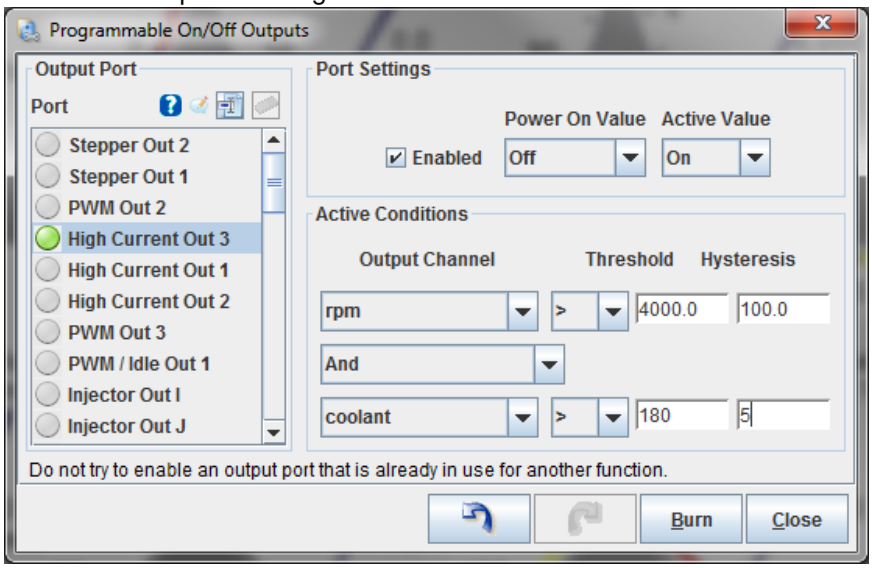
- **Pit limiter enable** - Toggles this feature on or off.
- **Input** - Specifies what switch to use to the speed limiter.
- **Mode** - Fixed mode uses a single speed. Holding mode will hold your speed limit at the speed when the switch is activated.
- **Use VSS or RPM target** - In holding mode, this allows you to use either the vehicle speed sensor or RPM to determine speed. Fixed mode always uses VSS.
- **Speed** - The maximum speed in fixed mode.
- **Speed control range** - Control range of speed to apply limiters across. 5mph recommended as a starting point. Setting the range too small will cause oscillation.
- **RPM control range** - Control range of RPM to apply limiters across. Setting the range too small will cause oscillation.
- **Tracking sensitivity** - Adjust tracking sensitivity to adjust limiter window. Initially the 0% fuel/spark cut is applied at the limit and full cut is applied at the limit+range. If the speed is above the limit, the lower limiting point is lowered slowly.
- **Spark retard mode** - Allows using spark retard while the speed limiter is activated.
- **Maximum retard** - The most spark retard allowed, in degrees from the regular timing table.

- **Enable spark cut limiting** - Allows using a spark cut as pit speed rev limiter.
- **Enable fuel cut limiting** - Allows using a fuel cut as pit speed rev limiter.
- **Progressive fuel cut** - Ramps in the fuel cut instead of applying it all at once. Requires sequential injection to be enabled.

7.8.24 Programmable on/off outputs 1

This screen is for setting up generic on/off outputs for features like variable intake control solenoids, secondary cooling fans, and other items that don't need complicated control settings.

Note that if you're using one of these outputs for another function, you don't want to enable it here. For example, if you're using PWM / Idle Out 1 for idle control, don't enable it as a programmable on/off output. Enabling it here will tell the MS3Pro that you're trying to use PWM / Idle Out 1 as some sort of on/off output at the same time, and the MS3Pro will report a configuration error.



7.8.24.1 Output port pane In the left hand pane, ports currently configured through this system are shown with a green bulb. To configure a port, click on the name to highlight it and then view or alter the settings in the right hand pane.

7.8.24.2 Port settings pane

- **Enabled** This enables or disables the selected port. (Toggles green bulb on/off) Do not try to configure a port already used for another feature.
- **Power On Value** Whether the output should be on or off when MS3Pro first starts. Ensure that this matches up with the conditions the MS3Pro is likely to see on startup.
- **Trigger Value** The output state when your conditions are met. Typical settings would have the power on value as Off and the trigger value as On.

7.8.24.3 Conditions This pane is where you specify the logic that activates the output.

Output channel This is a list of MS3Pro internal variables - much the same as the various gauges you can select in TunerStudio.

> = < This “logical operator” forms part of the equation.

- > is greater than
- = is equals. Usually, you don’t want to use the = sign here. This will turn on the output only when the variable is **exactly** equal to the threshold number, with **zero** difference. There are a few variables, such as SD card error conditions, where this can be useful. Most of the time, a variable won’t hit your threshold right on the nose, and the output will stay inactive.
- < is less than

Threshold The value compared against.

Hysteresis It is very common for a value to hover around the threshold, which would cause the output to turn on and off rapidly. This setting helps to prevent that by creating a dead band below the threshold before the output turns back off again.

- & is a bit-wise AND function. This is normally used with Status, CELStatus, or Port fields. Both the Threshold and Hysteresis values work completely differently when using this mode. This output is triggered when the result of performing a bit-wise AND function between the parameter and the threshold is equal to the hysteresis value.

For example, suppose you want a value to activate when launch control is armed, but not active. Looking at the Status2 field, we see we have the following bits in the field.

128	64	32	16	8	4	2	1
Closed loop idle	Overboost protection	Spark cut	Flat shift active	Launch control active	Launch control armed	Nitrous stage 2 active	Nitrous stage 1 active

We see launch control active has a value of 8, and launch control armed has a value of 4. So to look at both these bits, we add these values together and set the “threshold” (bitmask) to 12. The bitmask “trims down” the Status2 field like this:

128	64	32	16	8	4	2	1
Blocked by bitmask	Blocked by bitmask	Blocked by bitmask	Blocked by bitmask	Launch control active	Launch control armed	Blocked by bitmask	Blocked by bitmask

So, the possible values you can get through the bitmask are 0 (launch neither armed nor active), 4 (launch active but not armed), 8 (launch active but not armed - not really possible) and 12 (launch both armed and active).

We then want to check that the value that makes it through the bitmask has launch armed (4) active, but NOT launch 8. So we set the “hysteresis” (match) value to 4.

Additional condition

- No additional condition - only a single condition is used
- Or - output triggers if either condition 1 OR condition 2 is met
- And - output triggers only when condition 1 AND condition 2 are both met

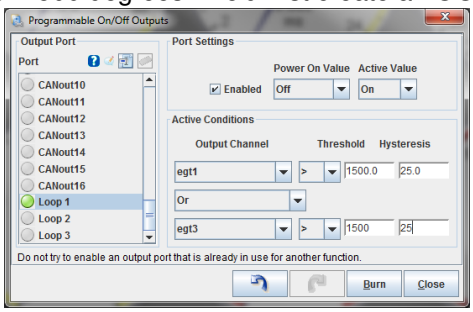
7.8.25 Programmable on/off outputs 2

This screen works in the same way as the first programmable on/off outputs, but is used to trigger outputs on remote CAN devices and loop conditions. Loop outputs behave as if you have “looped” a wire from an MS3Pro output pin back into one of its own input pins to use it to trigger an external device such as a table switching input or three step rev limiter - except you don’t have to use up any real pins.

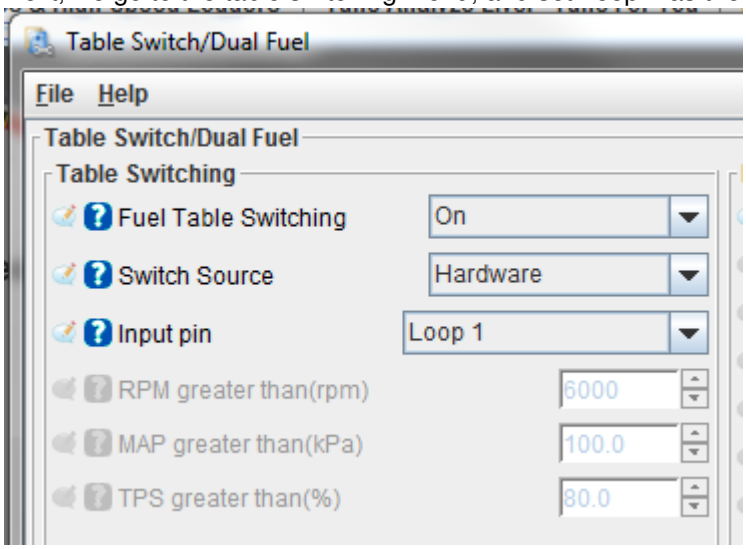
7.8.25.1 Loop conditions There are eight outputs called loops. These do not toggle physical outputs, but instead represent a virtual input switch that can be used to activate other features that normally use an on/off input, such as table switching or launch control. The loop also creates a bitfield that can be used with the AND operation on other on/off outputs:

128	64	32	16	8	4	2	1
-	-	-	-	-	Loop 3	Loop 2	Loop 1

For example, suppose you want to use an alternate fuel table when the EGT on cylinder 1 or cylinder 3 goes over 1500 degrees. We'd first create a LOOP condition triggered by EGT:

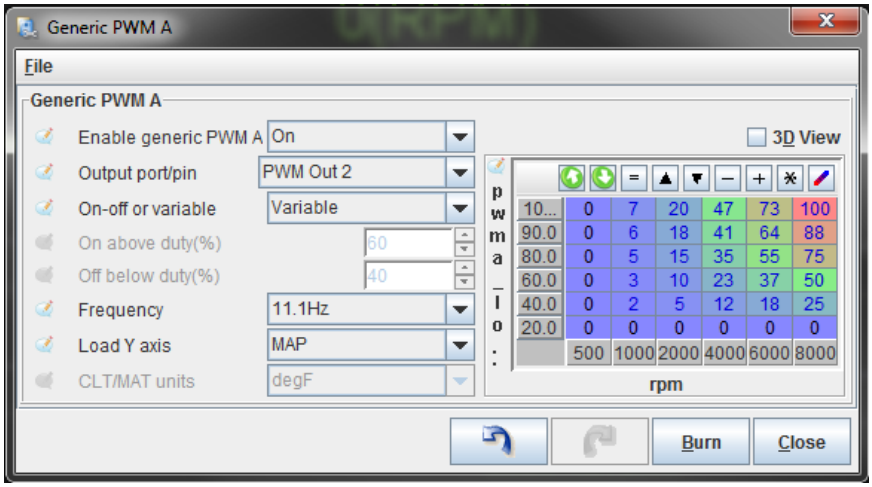


Next, we go to the table switching menu, and set Loop 1 as the source.



7.8.26 Generic PWM outputs

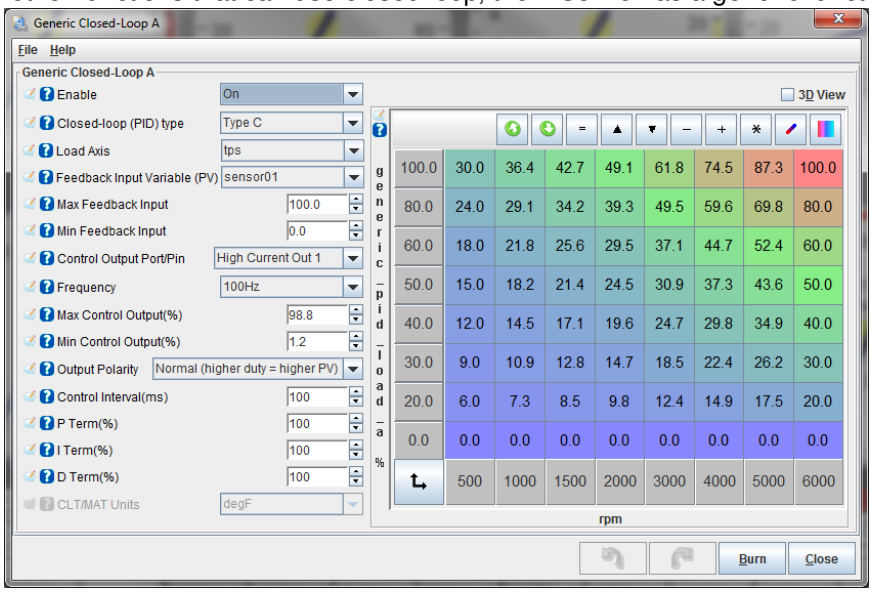
The Generic PWM system allows up to six independent open-loop PWM (pulse width modulated) or on/off outputs. The duties are programmed through an 6x6 table of rpm vs. Y-axis. For the Y-axis you can choose from MAP, %baro, TPS, MAF load, battery voltage, CLT, or MAT. The output is typically used to drive a solenoid valve (PWM mode) or relay (on/off.) PWM is a method that pulses an output on and off, the on-time percentage (duty) is variable, this allows a valve to "float" in-between closed and open for variable flow rates.



- **Enable generic PWM** - Turns this PWM channel on or off
- **Output port/pin** - Selects the output pin to use
- **On-off or variable** - On/off or variable PWM output
- **On above duty** - For on/off mode, duties in the table above this value turn the output on.
- **Off below duty** - For on/off mode, duties in the table below this value turn the output off. (Leave a gap to prevent chatter.)
- **Frequency** - For variable outputs, this sets the frequency to pulse the output channel.
- **Load Y axis** - Choose from MAP, %baro, TPS, MAF load, battery voltage, CLT, MAT for the Y-axis of the table.
- **Table or Curve -** You can use a 3D table of load and RPM or a 2D curve with just the load to determine the output.
- **CLT/MAT units** - This settings chooses the temperature scale in use. It can only be set through the "Generic Sensors" menu.

7.8.27 Generic Closed Loop outputs

The MS3Pro has several predefined closed loop functions, such as idle speed, fuel pressure, and boost control. For other functions that can use closed loop, the MS3Pro has a generic function.



- **Enable** - Turns the generic closed loop function on or off.
- **Closed loop PID type** - This allows you to change which PID loop type is used. A Type B loop has the P term react to changes in error, while a Type C load has the P term respond to changes in the feedback input variable instead.
- **Load Axis** - Sets what variable is used for the load in the 3D table.
- **Feedback Input Variable** - This is the sensor input to target with the PID loop.
- **Max Feedback Input** - The highest value the loop can target.
- **Min Feedback Input** - The lowest value the loop can target.
- **Control Output Port** / **Pin** - The PWM output channel used to control the device.
- **Frequency** - The frequency used to pulse the output.
- **Max Control Output** - The highest duty cycle the PID loop can command.
- **Min Control Output** - The lowest duty cycle the PID loop can command.
- **Control Interval** - The time between updating the control output duty.
- **P Term, I term, and D term** - The gain terms used to control the PID loop.
- **CLT** / **MAT units** - If using a temperature input as the load or feedback input variable, you can set this screen to display Celsius or Fahrenheit.
- **Target Table** - This is the target value which the closed-loop algorithm aims to match to the feedback input by controlling the PWM output.

7.9 3D Tuning Maps

This menu allows alternate 3D views of tuning maps from other settings: Fuel, ignition, boost control, and sequential injection timing. Each view has a set of mini gauges to help tune the map. You may right click on the gauges and swap them to other values if you prefer to use a different gauge type.

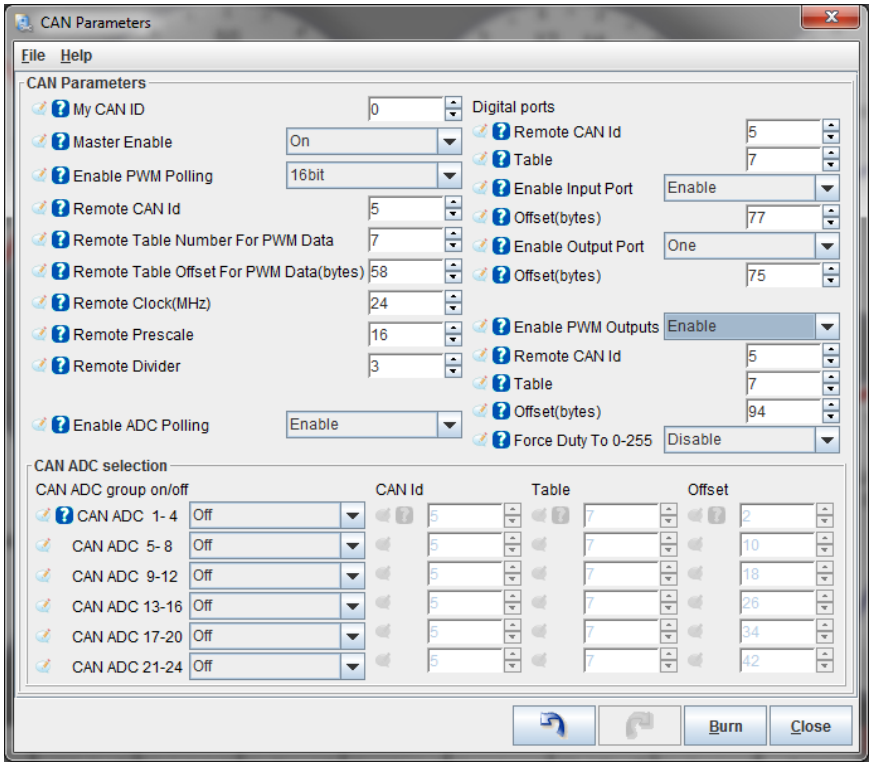
7.10 CAN bus / Testmodes

This menu allows you to connect the MS3Pro to external CAN based devices, use diagnostic functions including check engine light and limp home modes, and enable a few highly specialized options.

7.10.1 CAN parameters

The MS3Pro's CAN wires allow it to connect to other devices over a network that uses a simple two wire connection. These devices can add inputs and outputs, or display information from other devices on the CAN network. Here are some general notes; we plan to offer more detailed notes on exactly how to use these settings with specific devices. You will only need to set this up if the MS3Pro is using the CAN network for data acquisition or to control remote outputs. You do not need to do anything other then set the CAN ID to zero if using the MS3Pro CAN network for an external dashboard or other CAN device that is set up to automatically fetch data from the MS3Pro.

The CAN network typically works by passing along sections of memory tables. So many of the settings will deal with telling the MS3Pro where the table is in memory and how to retrieve it.



7.10.1.1 Base data

- **My CAN ID** - the CAN ID of this MS3Pro, normally left at zero.

Each device on the CAN network has its own unique ID, with the number zero reserved for the master device. You may connect up to 14 extra devices to the MS3Pro, so remote CAN ID numbers can be from 1 to 14.

- **Master Enable** - Enables fetching data from expansion board via CAN. Setting to "Off" still allows CAN-passthrough for tuning but turns off all other messages. This can be useful for performing a firmware upgrade on a remote expansion board.

7.10.1.2 PWM polling The MS3Pro can collect remote PWM or frequency based type input data, from a VSS for example. Turning “Enable PWM polling” on enables capturing PWM (pulse/frequency) from expansion board and selects either 16bit or 32bit data. Refer to your expansion board documentation for the appropriate setting.

- **Remote CAN ID** - the CAN ID of the remote board
- **Enable PWM polling** - enable or disable this feature
- **Remote table number for PWM data** - varies by application
- **Remote table offset for PWM data** - varies by application.
- **Remote clock** - The clock on the remote device
- **Remote prescale** - If used a prescale divider
- **Remote divider** - If used a clock divider

7.10.1.3 Digital I/O This allows the MS3 to use up to eight remote on/off inputs (CANIN1-8) and up to sixteen remote on/off outputs (CANOUT1-16). CANIN1 is bit 0 from the remote input port and CANIN8 is bit7. CANOUT1 is bit 0 from the first remote output port and CANOUT8 is bit7. CANOUT9 is bit 0 from the second remote output port and CANIN16 is bit7.

- **Remote CAN Id** - the remote CANid to fetch from

- **Table** - the remote table number
- **Enable input port** - enable remote input port
- **Offset** - the offset number in the remote table
- **Enable output port** - enable remote output port
- **Offset** - the offset number in the remote table

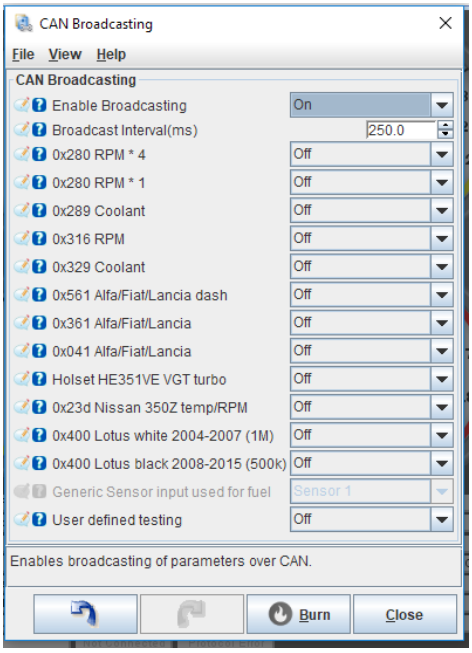
7.10.1.4 PWM outputs MS3Pro can command PWM outputs on remote CAN devices. You will need to refer to your CAN device's documentation for the appropriate settings to use.

- **Enable PWM outputs** - Enables sending PWM/pulsed outputs to expansion board.
- **Remote CAN ID** - Enables sending PWM/pulsed outputs to expansion board.
- **Table** - Enables sending PWM/pulsed outputs to expansion board.
- **Offset** - Data offset on the expansion board used for PWM outputs.
- **Force duty to 0-255** - Normally, PWM duty cycles read on a 0-100% scale. However, some remote devices require an 8 bit, 0 to 255 scale. This setting enables the 8 bit scaling.

7.10.1.5 Analog (ADC) inputs over CAN This allows additional analogue inputs to be captured from remote CAN devices and used with most MS3 features. The code allows up to 24 remote CAN ADCs. The dropdown menus (e.g. EGT) allow you to select either a local ADC or a remote CANADC. Before attempting to use the chosen CANADC, you need to enable it on this CAN settings menu. Due to data sizes, the CANADCs are configured in groups of four. The exact table and offset numbers are specific to the configuration of the remote device.

- **Enable ADC polling** - enable or disable the feature
- **CAN ADC 1-4** - enable or disable this group.
- **CAN Id** - the remote CANid to fetch from
- **Table** - the remote table number
- **Offset** - the offset number in the remote table

7.10.2 CAN broadcasting

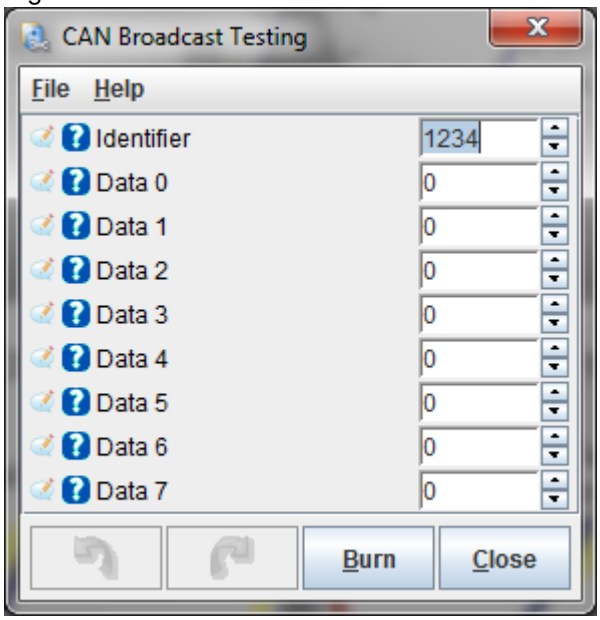


This is for interfacing with simple gauge circuits that need the MS3Pro to actively send out information at regular intervals. You can specify how often to transmit this information. Currently, the MS3Pro can broadcast RPM or coolant temperature data, and supports a handful of OEM dashes. Here are some notes about known applications:

- **0x280 RPM * 4, 0x290 RPM * 1, 0x289 Coolant:** Used in Porsche instrument clusters (the 996 is one known application) and some Volkswagens. You will need to test if you need the *4 or *1 multiplier on RPM output, depending on the vehicle.
- **0x316 RPM and 0x329 Coolant:** Used on E46 chassis BMW instrument clusters.
- **0x561, 0x361, 0x041 Alfa / Fiat / Lancia:** Used in some mid 2000s cars from these manufacturers.
- **Holset HE351VE VGT turbo**:Allows you to send a commanded boost duty cycle to a Holset VGT turbo controller, as used on the HE351VE.
- **0x23d Nissan 350Z temp / RPM:** Used on Z33 chassis Nissan 350Z.
- **0x400 Lotus** modes: Used on Lotus Elise / Exige. Sends RPM, speed, coolant temperature, shift light, check engine light, oil pressure warning, and fuel level. The latter two require enable an oil pressure sensor and fuel level sensor under the generic sensor menu, and all warning lights must be configured as light outputs to work. You will need to set the CAN baud rate to 1M for the 2004-2007 mode and 500K for the 2008-2015 mode.

7.10.3 CAN broadcast testing

This screen allows you to send a user defined CAN message with a 10 bit identifier and eight data bytes. Used for testing remote CAN devices.

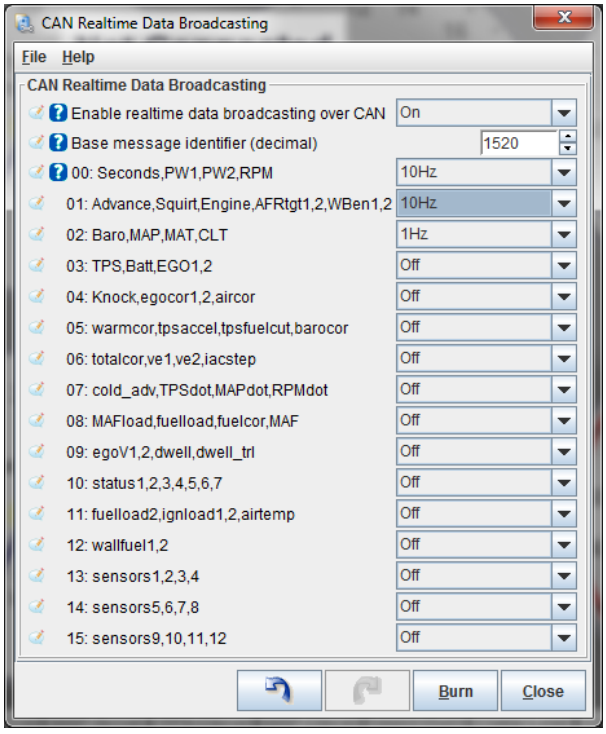


7.10.4 CAN realtime data broadcasting

This mode allows broadcasting messages over CAN to display devices or data loggers.

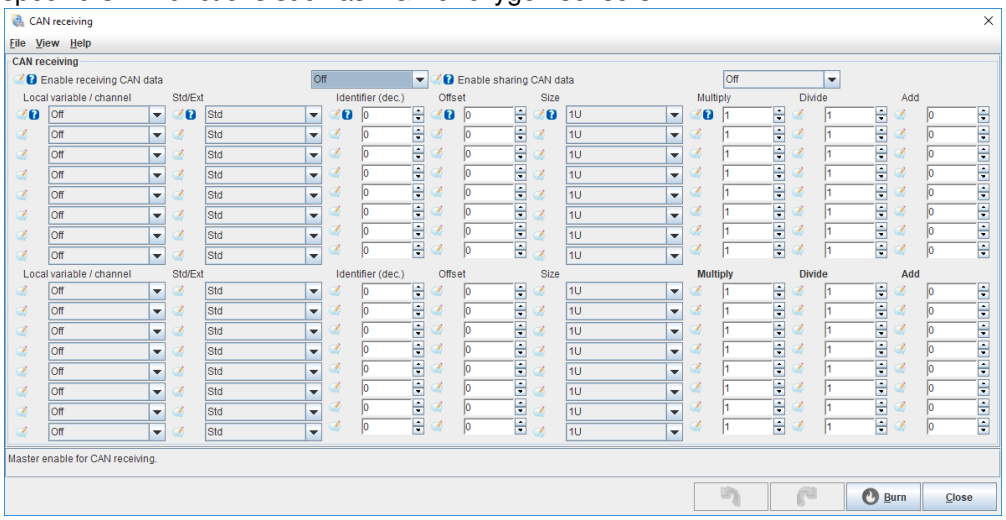
The base message identifier in decimal of the sequential 11-bit CAN addresses used for realtime data (outpc) broadcasting. The ECU assigns the next identifier for each data group whether enabled or not. i.e. group 0 is at base identifier, group 17 is at base identifier+17

This means that by using the default base address you can use a predefined template for your dash, etc. The data is divided into separate groups, and each can be broadcast at a separate frequency. Pick an appropriate rate for the data within the group; there is no benefit in transmitting slow changing data at a high rate.



7.10.5 CAN receiving

Allows MS3Pro to receive certain types of CAN data. Consult the documentation for your CAN device as to how to configure this. “Enable sharing CAN data” allows you to use data packets both as generic CAN receiving and also for specific CAN functions such as EGT or oxygen sensors.



7.10.6 CAN VSS, gear

These settings are for third party CAN enabled devices that provide the MS3Pro with speed, or gear information. Refer to your device’s documentation for how to configure it with the MS3Pro.

7.10.7 CAN EGO, GPS

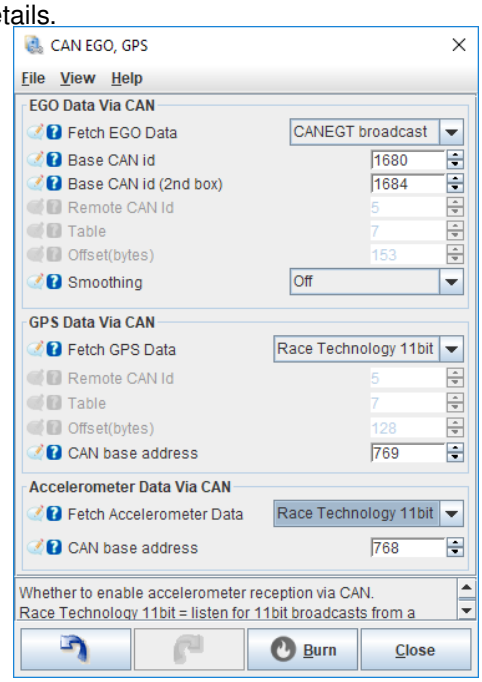
The MS3Pro can also receive O_2 sensor, accelerometer, and GPS data over CAN. It supports the following devices for O2 sensor input.

- Innovate Motorsports serial data is not supported directly, but you can connect an Innovate Motorsports serial data chain to a CAN-EGT module or third party devices sold by JBPerf that translate this data into MS3Pro 29 bit CAN.
- AEM 4 channel and X series UEGO devices use their own AEMNet CAN format. If using an AEM external wideband over CAN, you must disable all other 29 bit CAN functions such as interfacing with a MicroSquirt transmission control or CAN-EGT module running in 29 bit mode. 11 bit functions such as dash broadcasting will still function.
- ECOTRONS widebands also use their own proprietary 29 bit CAN format and require disabling all other 29 bit CAN functions. You will need to set the CAN data rate to 250K instead of the usual 500K under CAN Parameters to use an ECOTRONS wideband.
- Generic wideband receiving allows for receiving user defined messages over 11 bit CAN. You will need to configure the device under the CAN Receiving screen to provide a value of lambda * 10000.
- The CAN-EGT mode uses the CAN-EGT module's 11 bit broadcasting.

The MS3Pro supports the following devices for GPS data:

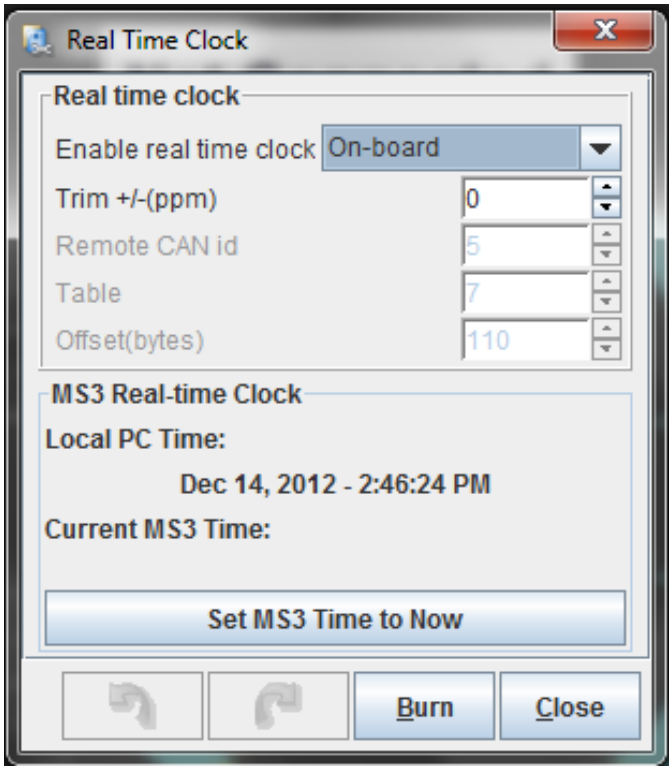
- JBPerf GPS modules
- Race Technology devices that transmit GPS and accelerometer data.

JBPerf devices and other devices that use 29 bit MS-CAN require configuring the remote CAN ID, table, and offset values. Race Technology devices require setting a base address. Consult your CAN device's documentation for details.



7.10.8 Real time clock

The MS3Pro has a built in clock that lets you apply a date stamp to data logs.



- **Enable real time clock** - You may set this to “On-board” to use the MS3Pro’s internal clock, or “CAN” to use an external, CAN enabled device such as a GPS module.
- **Trim** - This is measured in parts per million, and lets you adjust the clock if you find it is running fast or slow.

The Remote CAN ID, Table, and Offset devices are only used if you are using a clock on a CAN device instead of the MS3Pro’s internal clock. If you are using a remote clock, refer to your CAN device’s documentation for how to configure this,

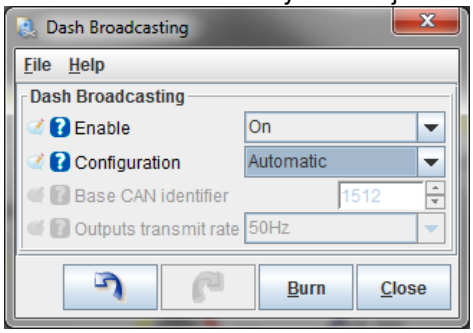
The “Set MS3 Time to Now” button will synchronize the MS3Pro internal clock to your computer’s clock.

7.10.9 IO Box settings

The MicroSquirt IO Box firmware allows you to connect up to four MicroSquirt controllers to the MS3Pro to provide the MS3Pro with additional input and output channels with greatly simplified CAN settings. Consult the MicroSquirt IO Box firmware documentation for details.

7.10.10 Dash broadcasting

A simplified CAN broadcasting mode that allows the MS3Pro to communicate with aftermarket dashes and data logs such as Racepak V-net. Automatic broadcasting uses a base CAN identifier of 1512 and broadcasts at 50 Hz.Advanced mode allows you to adjust the base CAN ID and the transmission rate.

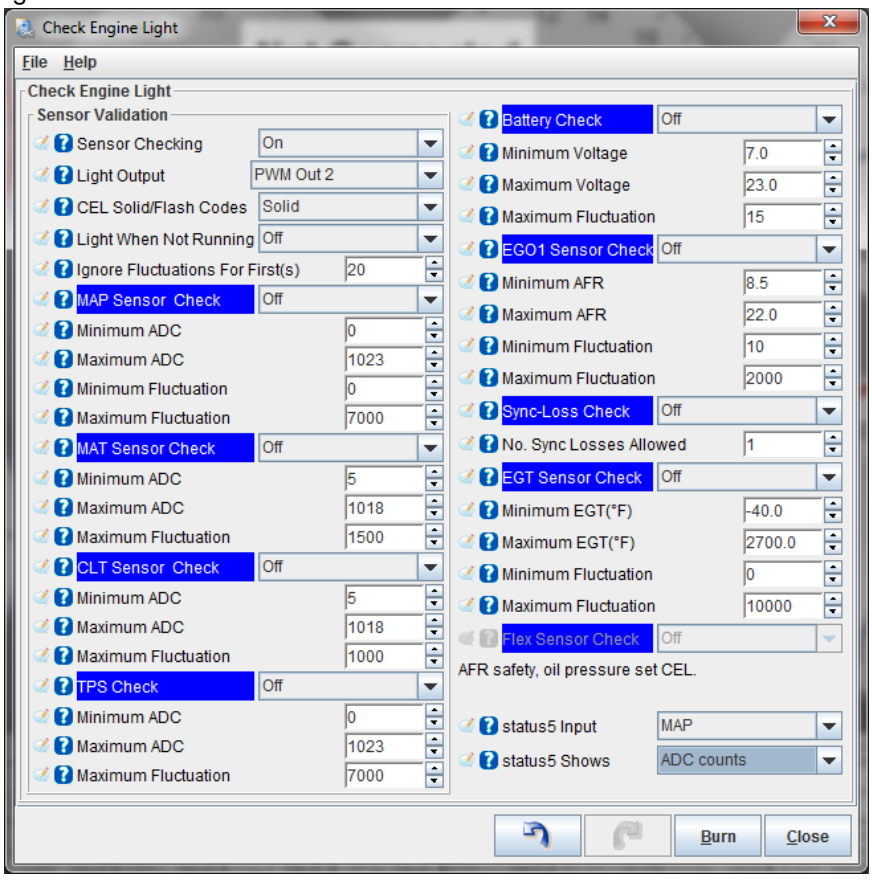


7.10.11 Check Engine Light

The MS3Pro has a sensor validation system that can sanity check the sensor inputs. If the sensor appears to be malfunctioning, then it can switch on a Check Engine light. Additionally, limited operating strategies (limp modes) can be enabled. The aim of these settings is firstly to warn you of a failed sensor that could cause poor or damaging engine operation and secondly, through the limp modes, to allow some engine operation even with the failed sensor. The checking scheme varies slightly for each sensor. Most sensors are checked against minimum and maximum possible values that can occur if the wire breaks or is shorted to ground or 5V. The other criteria is minimum and maximum fluctuation - this is to detect a stuck or wildly flailing sensor. All of these parameters are engine specific, and you will need to dial them in for your installation.

The sensor validation often works with ADC counts. ADC stands for analog to digital converter, and refers to a circuit in the MS3Pro's processor that converts an analog voltage to a digital readout. The MS3Pro converts sensor readings into a 0 to 5 volt signal that goes to the processor, and the processor uses a 10 bit converter to turn this into a value that ranges from 0 at 0 voltage to 1023 at 5 volts. Usually, these ADC counts are hidden from the user - you're more interested in what the sensor reading actually means than a raw number. However, with sensor diagnostics, you'll be looking at the ADC count directly, since we're looking at establishing when a signal doesn't mean anything other than that your sensor isn't working. For starting settings, try a minimum ADC value of 5, and a maximum ADC value of 1018. You can use Status 5 to data log the actual ADC readings.

The fluctuation units are a bit arbitrary; we recommend data logging the values to see what is normal, and using that to set the minimum and maximum values.



- **Sensor checking** - Enables or disables this system
- **Light output** - Selects an optional output for the Check Engine Light. Recommended. Note that this can share a channel with the AFR Safety warning light.
- **CEL Solid** / **Flash Codes** - You can have the Check Engine light stay on, or transmit flash codes.
- **Light When Not Running** - Specify if the Check Engine light should be off or on when the engine RPM is at zero.

- **Ignore fluctuations for first** - The number of seconds that sensor fluctuations are ignored after startup. This is to allow conditions to settle down and avoid false alarms.
- **MAP sensor check** - Enable or disable checking of the MAP sensor input.
- **Minimum ADC** - minimum valid ADC reading for the MAP sensor
- **Maximum ADC** - maximum valid ADC reading for the MAP sensor
- **Minimum Fluctuation** - minimum fluctuation during smooth running
- **Maximum Fluctuation** - maximum fluctuation during operation

The minimum fluctuation setting is likely the most useful setting here, as it can be used to detect a failed or "stuck" sensor, or if a hose fell off. A correctly operating MAP sensor will not produce a perfectly flat value on a running engine.

- **MAT sensor check** - Enable or disable checking of the MAT sensor input.
- **Minimum ADC** - minimum valid ADC reading for the MAT sensor
- **Maximum ADC** - maximum valid ADC reading for the MAT sensor
- **Maximum Fluctuation** - maximum fluctuation during operation
- **CLT sensor check** - Enable or disable checking of the CLT sensor input.
- **Minimum ADC** - minimum valid ADC reading for the CLT sensor
- **Maximum ADC** - maximum valid ADC reading for the CLT sensor
- **Maximum Fluctuation** - maximum fluctuation during operation

The CLT sensor should be smooth in operation; the max fluctuations can be used to detect a fault. You're likely to see a lower maximum fluctuation value here than on most other sensors. Not only do you have a sensor with a fairly large thermal mass, but the coolant and engine block aren't going to change temperature very quickly.

- **TPS sensor check** - Enable or disable checking of the TPS sensor input.
- **Minimum ADC** - minimum valid ADC reading for the TPS sensor
- **Maximum ADC** - maximum valid ADC reading for the TPS sensor
- **Maximum Fluctuation** - maximum fluctuation during operation

For the TPS, the minimum and maximum ADC values are the more likely settings to detect a fault. You can view the ADC counts at closed and wide open throttle under the Tools menu when you select Calibrate TPS. A good starting point is a minimum ADC that is 10 counts below the closed throttle calibration and a maximum ADC 10 counts above the full throttle calibration.

- **Battery check** - Enable or disable checking of the battery voltage input.
- **Minimum voltage** - minimum acceptable voltage reading
- **Maximum voltage** - maximum acceptable voltage reading
- **Maximum Fluctuation** - maximum fluctuation during operation

This can be used to warn of charging system faults or ground issues causing excessive voltage noise.

- **EGO1 sensor check** - Enable or disable checking of the EGO1 sensor input.
- **Minimum ADC** - minimum valid ADC reading for the EGO1 input
- **Maximum ADC** - maximum valid ADC reading for the EGO1 input
- **Minimum Fluctuation** - minimum fluctuation during smooth running

- **Maximum Fluctuation** - maximum fluctuation during operation

The fluctuation settings are key to detecting a failed O_2 sensor. Many wideband controllers will get "stuck" if the sensor fails, although appearing to show a valid AFR reading. Combined with closed-loop EGO or autotuning, this can be troublesome without this detection scheme. Other controllers allow an “out of range” voltage output - match the min/max ADC readings to capture this.

- **Sync-loss check** - Enable or disable checking of sync losses to troubleshoot crank or cam sensors.
- **No. sync losses allowed** - how many recorded losses of tach-in synchronization are allowed before setting the CEL. (During normal operation, you may see one or two sync-losses during startup. Beyond that, there is a problem with your RPM input wiring or configuration.)
- **EGT sensor check** - Enable or disable checking of the EGT sensor inputs.
- **Minimum EGT** - minimum valid EGT while engine is running
- **Maximum EGT** - maximum valid EGT while engine is running
- **Minimum Fluctuation** - minimum fluctuation during smooth running
- **Maximum Fluctuation** - maximum fluctuation during operation ()

The exhaust gas temperatures are not checked immediately after startup, so the minimum temperature can be set to 200C to detect a dead cylinder or faulty sensor. The maximum will depend on your engine. N/A could be around 900C, boosted perhaps 1000C.

- **Flex sensor check** - Enable or disable checking of the Flex sensor input. (An error condition will be triggered if the frequency on the flex input is outside of the set range.)
- **status5 input** - Specifies which sensor details are logged in status5 in the data log fields.
- **status5 shows -** If set the fluctuation, the value of status5 is the fluctuation number the code calculates and compares to the minimum or maximum for each chosen sensor. e.g. Set this to MAP and then enable ’status5’ on your dash or view in a datalog. Allow the engine to idle and observe the lowest number. You can then set the MAP minimum fluctuation to some value LOWER than this number. If the MAP input becomes too stable (e.g. the hose falls off), then the alarm will be raised. If set to ADC counts, this shows the raw analog to digital conversion reading of the sensor.

7.10.11.1 Check Engine codes The Check Engine light can be set to flash codes. These codes also show up as CELStatus in the data logs. The codes are as follows:

- 2: MAP sensor fault
- 3: MAT sensor fault
- 4: CLT sensor fault
- 5: TPS fault
- 6: Battery voltage fault
- 7: O_2 sensor fault
- 8: Loss of sync (crank or cam sensor)
- 9: EGT sensor fault
- 10: Flex fuel sensor fault
- 11: MAF sensor fault
- 12: Knock sensor fault
- 13: Cam sensor fault, engine still has sync but is running in semi-sequential

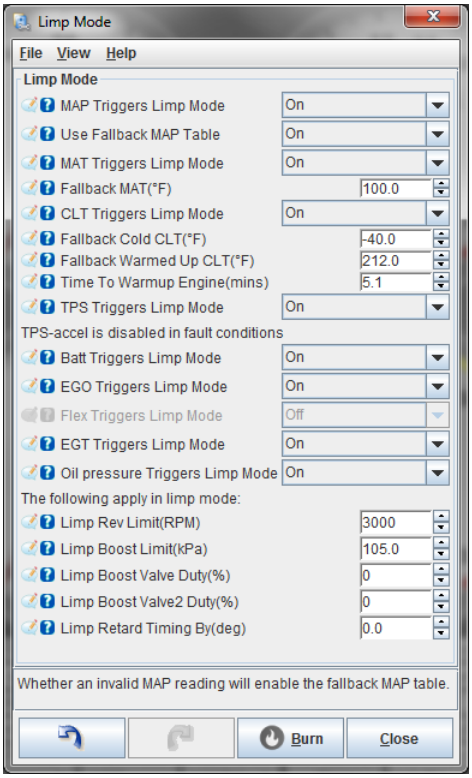
- 14: Oil pressure out of range
- 15: Fuel pressure sensor out of range
- 16: EGT safety shutdown triggered
- 17: AFR safety shutdown triggered

7.10.12 Limp Mode

After detecting a sensor fault, the MS3Pro can take action based on what it has detected - either substitute a replacement sensor reading, ignore the sensor, or use an alternate sensor. The limp mode can also impose a low rev limit and limit boost. The purpose of the limp mode is to get you home or off the race-track etc. without damaging the engine. Certain faults will cause poor operation; the aim is to allow some operation in preference to no operation.

Here is what action the MS3Pro can take in response to sensor failures.

- MAP sensor failure: The MS3Pro can either continue running on the MAP sensor but apply the rev and boost limits, or it can substitute a fallback MAP table that guesses the MAP reading based on throttle position and RPM.
- MAT sensor failure: Substitutes a fixed sensor reading.
- CLT sensor failure: Approximates a coolant sensor reading based on time after start. Note - It's a good idea to set the warmed up value to a point high enough to turn your cooling fan on, if the MS3Pro is controlling the fans.
- TPS failure: Disables TPS based acceleration enrichment and assumes a zero TPS value.
- EGO sensor failure: Turns off all closed loop O_2 correction.
- Flex fuel sensor failure: Uses the fallback timing addition and fuel multiplier defined on the flex fuel screen.



- **MAP triggers limp mode** - whether a MAP fault triggers limp mode

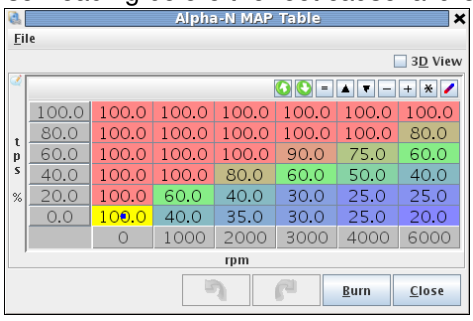
- **Use fallback MAP table** - if this is enabled, the MS3Pro uses a table to approximate the likely engine MAP reading if the MAP sensor fails.
- **MAT triggers limp mode** - whether a MAT fault triggers limp mode
- **Fallback MAT** - replacement MAT value
- **CLT triggers limp mode** - whether a CLT fault triggers limp mode
- **Fallback cold CLT** - replacement initial CLT value
- **Fallback warmed up CLT** - replacement final CLT value
- **Time to warmup engine** - time taken to blend from cold to warm value (simulating a warmup cycle)
- **TPS triggers limp mode** - whether a TPS fault triggers limp mode. TPS based acceleration is disabled if TPS is faulty.
- **Batt triggers limp mode** - whether an out of range battery voltage reading triggers limp mode
- **EGO triggers limp mode** - whether an EGO fault triggers limp mode
- **Flex triggers limp mode** - whether a Flex fault triggers limp mode
- **EGT triggers limp mode** - whether an EGT fault triggers limp mode
- **Oil pressure triggers limp mode** - controls if the oil pressure sensor reading triggers limp home mode. Note that this is configured under the oil pressure screen, not the check engine light screen.

The settings below are applied when any sensor triggers the limp mode.

- **Limp rev limit** - the hard fuel cut rev limiter when in limp mode
- **Limp boost limit** - the boost limit when in limp mode. Typically, this will be no boost to prevent engine damage in unknown conditions.
- **Limp boost valve duty** - forces boost valve1 to a set duty in limp mode.
- **Limp boost valve2 duty** - forces boost valve2 to a set duty in limp mode.
- **Limp retard timing by** - retards timing by the set amount in limp mode.

7.10.13 Fallback MAP table

In the event of a MAP sensor failure, the MS3Pro can use a synthesized MAP value based on TPS and Rpm. This will be less accurate than the real MAP reading, but the aim is to allow some driveability in the absence of a real sensor reading before the root cause failure is fixed.

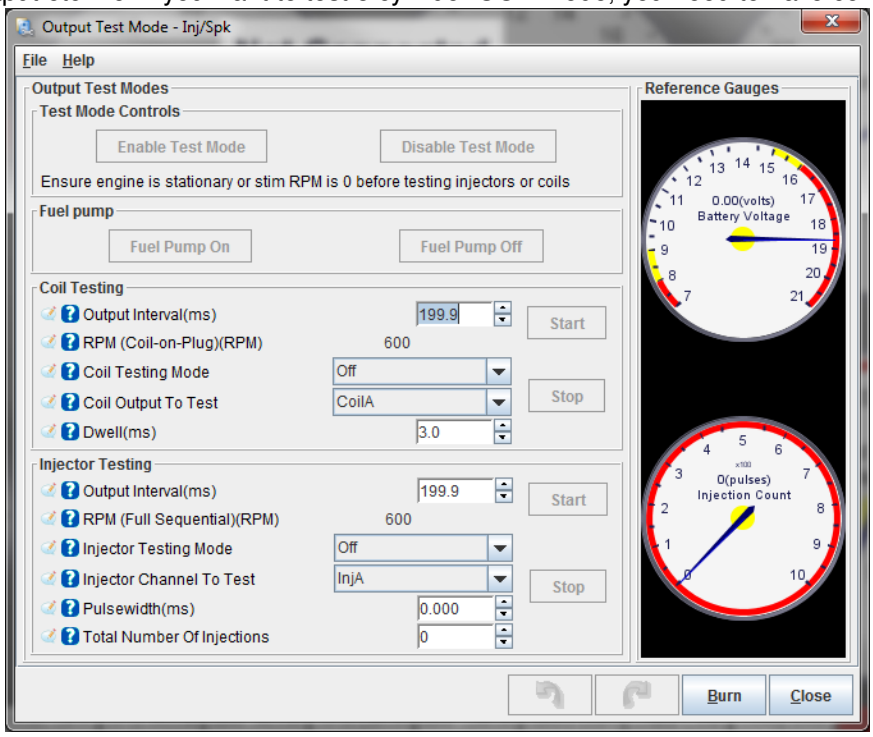


The initial vales for this table can be derived from datalogs comparing MAP, RPM, TPS while driving. You can use MegaLogViewer's scatter plot feature to see what MAP reading is common for the throttle position and RPM range in a data log; the shape of the scatter plot will be close to the table required.

To test and tune this table, unhook the MAP hose and wait until "Limp Mode" engages. Then this table will be operative and you can tune.

7.10.14 Output test mode - Inj / spark

The test mode allows direct control of the coils and injectors. Used by accident this could cause flooding of the engine and potential damage. Therefore a special initialization method is required to enable it. Do not attempt to test injectors or coils on a running engine. Before testing injectors on a engine be sure to depressurize the fuel rail - this can be achieved by unplugging the fuel pump while the engine is idling. Unless you are flow testing injectors on a flow-bench, you may well want to remove the fuel pump fuse during all of this testing. Before using the test mode, ensure that you have already configured the correct number of cylinders and type of injector and spark output etc. i.e. if you want to test 8 cylinder COP mode, you need to have configured that already.

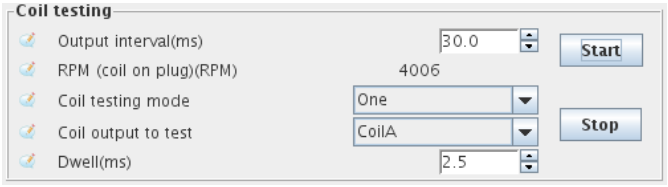


First ensure that you are key-on, engine-off. (Or that RPM is set to zero on the stim if bench testing.) In this condition the “Enable Test Mode” button will be available. Click the button to enable the test mode.

7.10.14.1 Fuel Pump Clicking Fuel Pump On will enable the fuel pump output and light the "Ready" indicator on TunerStudio.

Clicking Fuel Pump Off will disable the fuel pump output and turn off the "Ready" indicator.

7.10.14.2 Coils The coil test mode is useful to confirm you have the coils wired correctly and to check dwell settings. (Do be careful as ignition coils put out a potential lethal voltage. Typically 30,000V or more.)



To enable coil testing:

1. Choose the desired output interval (the estimated RPM equivalent is shown below.)
2. Set Coil testing mode to One
3. Choose which coil to test e.g. Coil A
4. Choose your dwell time in milliseconds. Start low. e.g. 2.5ms.
5. If your coils are powered via the fuel pump relay as is recommended, set the fuel pump to On.

6. With valid settings entered, the Start button should be active - click it.
7. When finished, click Stop.

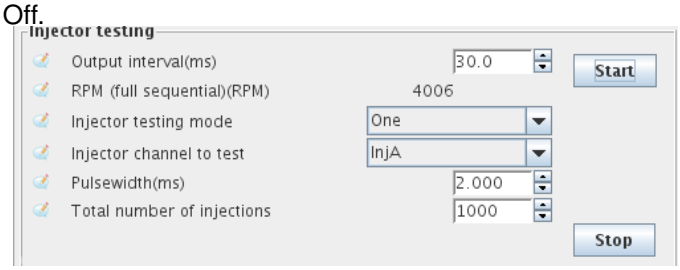
If the coils get hot or your coil fuses pop, then you almost certainly have a basic settings mistake with output polarity. Turn off power immediately and re-visit those settings.

The Coil testing mode called Sequence runs the coils in sequence up to the chosen coil. So if you have four coils and selected Coil D and Sequence then the test mode will repeatedly fire Coil A, then Coil B, then Coil C, then Coil D in turn. Note that the test mode does not allow dwell periods to overlap in the sequence mode, unlike normal operation where the dwell periods overlap as required. (This is why traditional V8 distributor spark was weak at high revs - there was not enough time to charge the single coil between sparks.)

7.10.14.3 Injectors The coil test mode is useful to confirm you have the injectors wired correctly and to experiment with opening times, PWM settings or to measure flow rate if you have an injector test bench setup. (Using flammable liquids while testing is hazardous and can lead to serious fires. Non-flammable injector test liquid is available.)

If your injectors are fitted to the engine you MUST NOT run the test mode with fuel pressure, or you will fill your engine with fuel! Remove the fuse from your fuel pump and depressurize the line first.

When semi-sequential is enabled, the injectors will fire in pairs. To test individually, temporarily set Sequential to Off.



To enable injector testing:

1. Choose the desired output interval (the estimated RPM equivalent is shown below.)
2. Set Injector testing mode to One
3. Choose which injector to test e.g. Injector A
4. Choose your pulsewidth time in milliseconds.
5. If your injectors are powered via the fuel pump relay as is recommended, set the fuel pump to On.
6. For functional testing, set the total number of injections to 65535 (the maximum), or for flow testing use your chosen number.
7. With valid settings entered, the Start button should be active - click it.
8. The number of injections gauge will count down and stop at zero.
9. To stop testing sooner, click Stop.

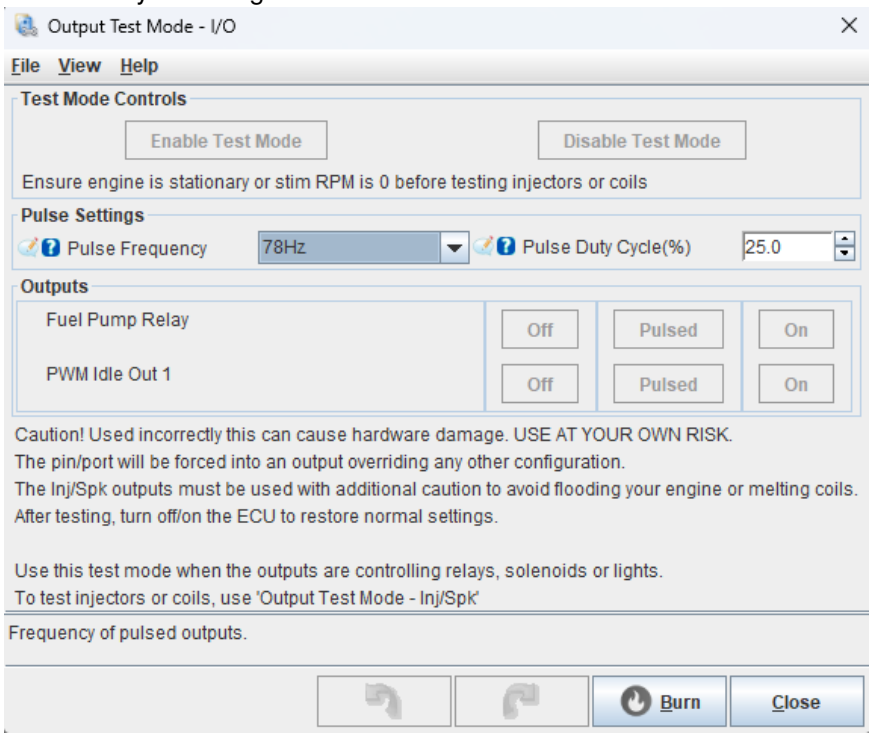
When you have finished using the test mode, close the test mode window and power off the MS3Pro. If you disconnected you fuel pump, reconnect it now.

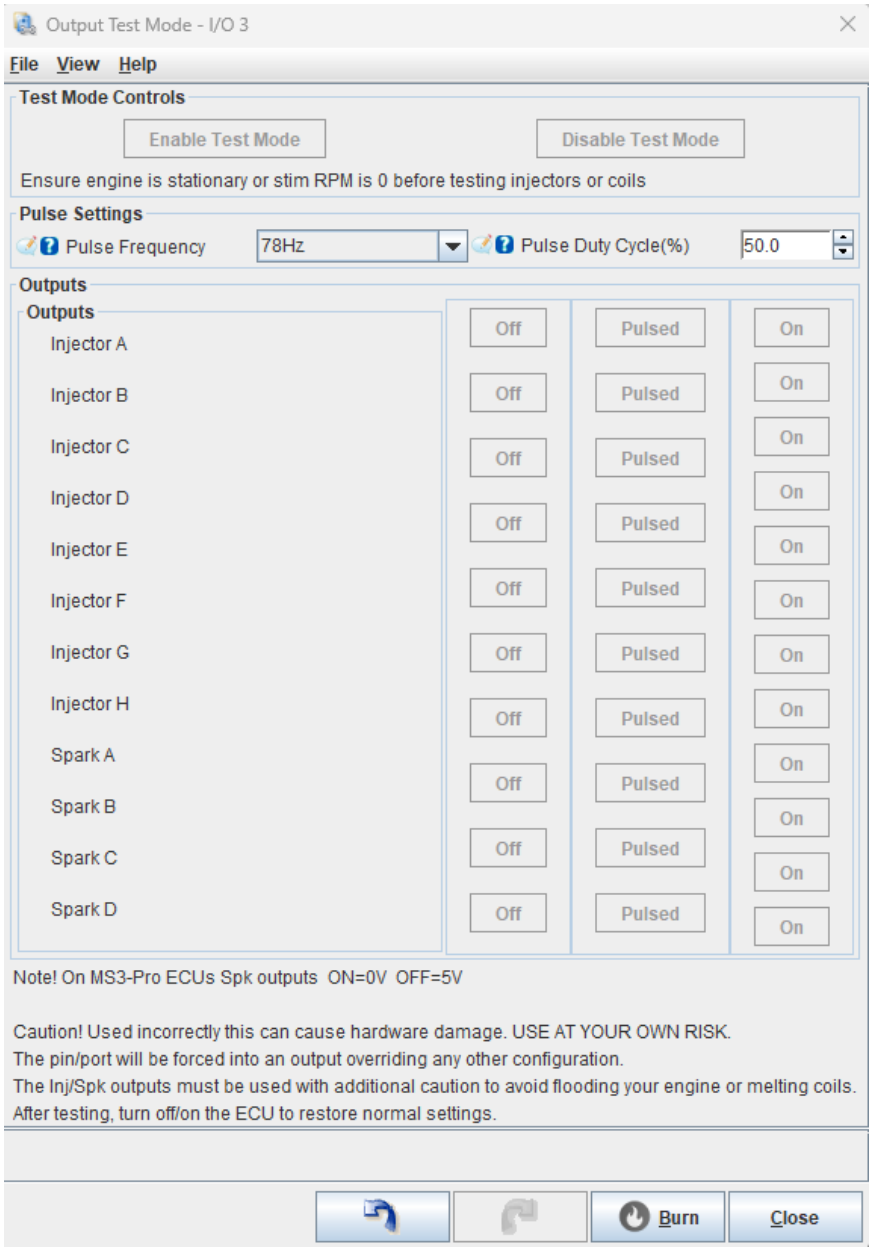
7.10.15 Output test mode - I/O 1 through 3

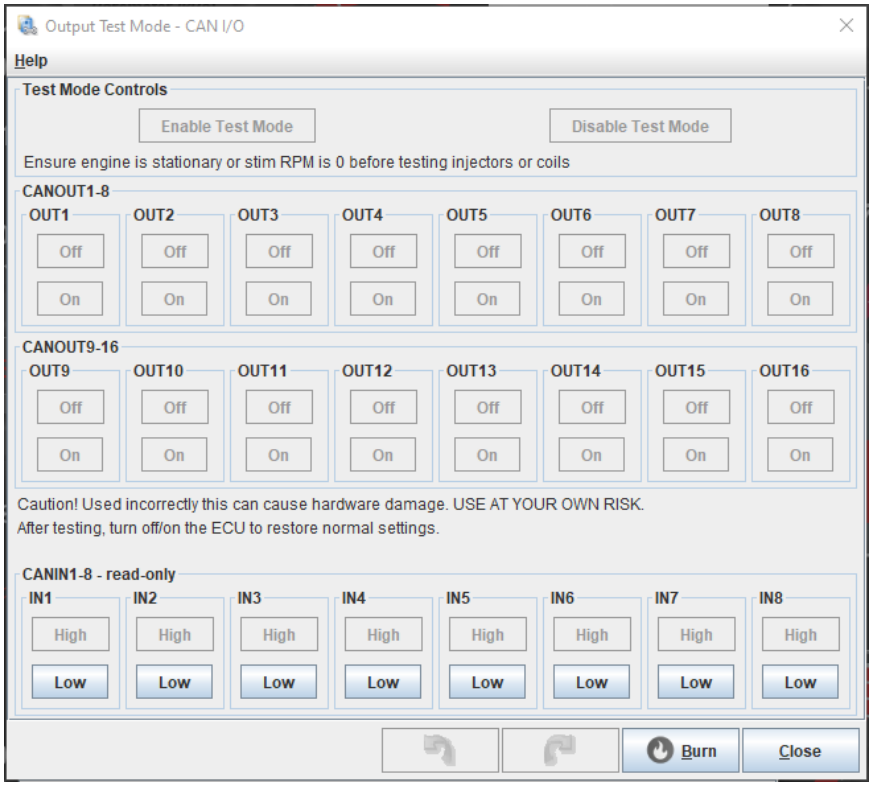
This allows you to test some of the on/off and pulsed outputs in a controlled manner. As with the other tests, be cautious - this mode allows you complete control over the outputs; it is up to you to ensure that doesn't cause any damage. These are split into three screens. Screen 1 tests regular on/off and stepper IAC outputs, as well as injector channels I and J. Screen 2 tests spark and injector outputs. The final screen tests CAN based I/O.

Be careful when using this output screen for testing spark outputs. Spark outputs behave "backwards" in this test mode, supplying 5 volts when off and 0 volts when on. They will often remain at 5 volts after exiting test mode until the ECU is switched off and back on again. Do not use this test mode with the ignition system

connected. It is, however, very useful for testing outputs with a Multimeter if you are not able to fire the coils and need to check your wiring.




Output Test Mode - I/O 3
File View Help
Test Mode Controls
Enable Test Mode
Disable Test Mode
Ensure engine is stationary or stim RPM is 0 before testing injectors or coils
Pulse Settings
Pulse Frequency
78Hz
Pulse Duty Cycle(%)
50.0
Outputs
Outputs
Injector A
Injector B
Injector C
Injector D
Injector E
Injector F
Injector G
Injector H
Spark A
Spark B
Spark C
Spark D
Off
Pulsed
On
Note! On MS3-Pro ECUs Spk outputs ON=0V OFF=5V
Caution! Used incorrectly this can cause hardware damage. USE AT YOUR OWN RISK.
The pin/port will be forced into an output overriding any other configuration.
The Inj/Spk outputs must be used with additional caution to avoid flooding your engine or melting coils.
After testing, turn off/on the ECU to restore normal settings.
Burn
Close



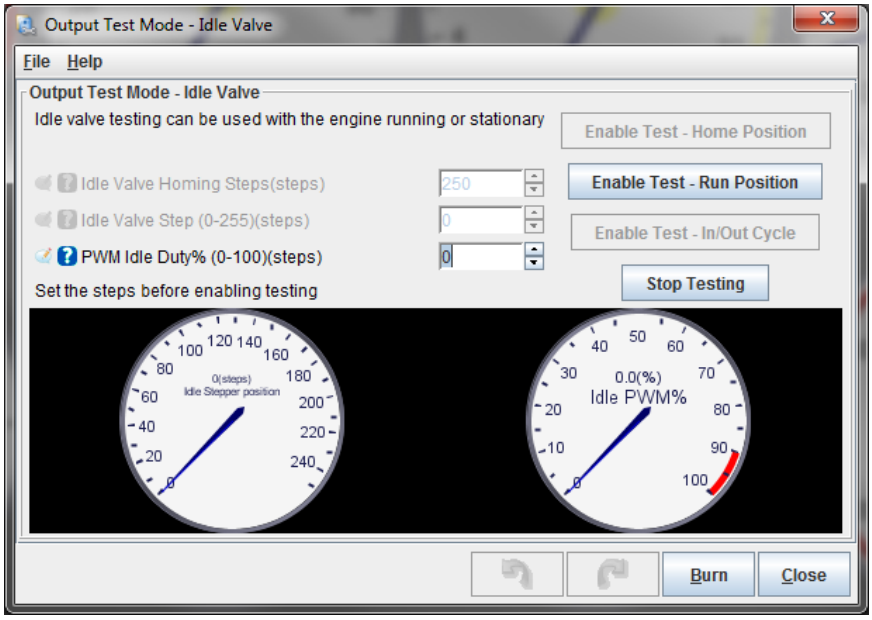
First ensure that you are key-on, engine-off. In this condition, the “Enable Test Mode” button will be available. Click the button to enable the test mode. For each of the outputs you can select the following:

- **Off** - turns the selected output off.
- **On** - turns the selected output on.
- **Pulsed** - pulses the output at the selected frequency and duty. (Not available on the fuel pump.)

Except for the Tach channel and stepper IAC outputs, all of these outputs are ground switched, so you will not be able to see their status with a volt meter. Use a small test lamp or test LED connected between 12V and the output, if you are not using them with a solenoid or the actual device you will be driving.

7.10.16 Output test mode - idle valve

The idle valve test mode can be used with the engine running or stationary. Note that with a PWM idle valve you may need to run a temporary fused 12V supply to the valve if you normally take power from the fuel pump relay and are testing with the engine off. You need to have enabled idle control before this is available.



- **Idle Valve Homing Steps** - This only applies to stepper idle valves. This setting determines how many steps the motor should move during homing to ensure it has reached the home position. The number needs to be large enough to close the valve from any unknown starting position. (This is because there is no position on most automotive stepper IAC valves.)
- **Idle Valve Step** - the desired valve position if using a stepper IAC valve. Larger numbers should give a more open valve and higher rpm.
- **PWM idle duty%** - the desired valve position if using a PWM IAC valve. Larger numbers should give a more open valve and higher rpm.

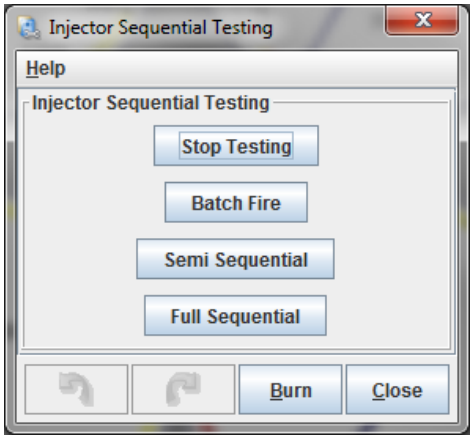
(Note that some Bosch rotary PWM valves valves feature a failsafe mode so that zero flow is at a midpoint, such as 20% duty,; above that the valve operates as expected where more duty gives more airflow. Below this specific point is a failsafe mode that allows airflow even if there is no control signal. This is readily observed during test mode.)

There are three control buttons:

- **Enable Test - Home Position** - for stepper idle valves, this moves the valve through the number of homing steps back to the home position. (If it doesn't get all the way back to the stop, increase the number of homing steps.) The valve will move to the run position shortly afterward.
- **Enable Test - Run Position** - enable test mode where the steps or duty setting controls the valve position
- **Enable Test - In / Out Cycle** - Moves the valve continuously between the home and run position.
- **Stop Testing** - disable test mode and allow normal idle control to operate.

7.10.17 Injector Sequence Testing

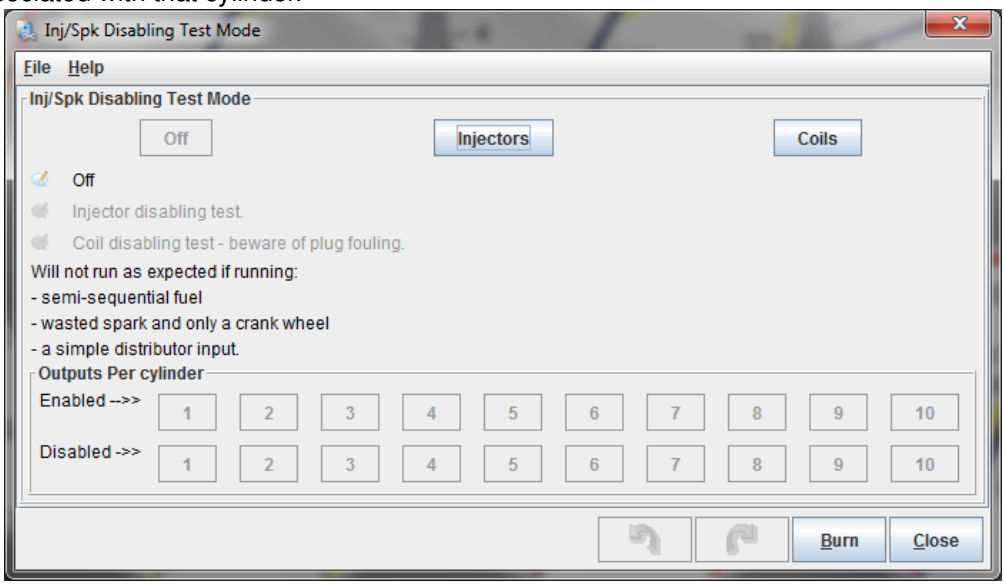
This screen allows you to switch in and out of sequential injection on the fly to evaluate how the engine runs in different modes. You can make an engine running full sequential drop down to semi sequential or batch fire, or make an engine running semi sequential swap to batch fire.



Note that Full Sequential and Semi Sequential modes only work if the engine is already configured to run in these modes. Clicking Stop Testing reverts to the existing settings.

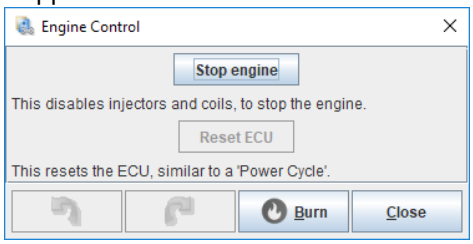
7.10.18 Inj / Spk Disabling Test Mode

Looking for a dead cylinder? Instead of unplugging the injectors or coils to check for a cylinder that's not working, the MS3Pro lets you shut down individual cylinders through TunerStudio. Note that you need enough inputs for full sequential operation to allow for disabling cylinders. Click the cylinder buttons to disable the coil or injector associated with that cylinder.



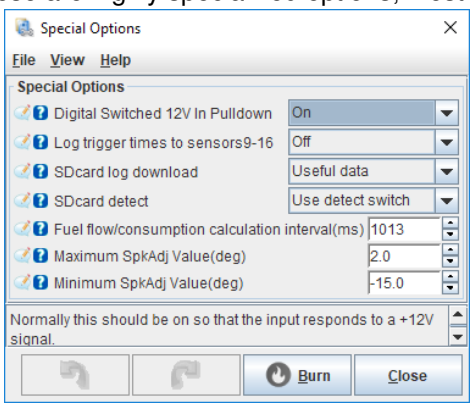
7.10.19 Engine controls

This allows you to command an engine shutdown through TunerStudio. Resetting the ECU requires the engine to be stopped and will cause the ECU to reboot itself, checking all variables that only take effect on startup.



7.10.20 Special options

These are highly specialized options; most users will not need to ever touch them.



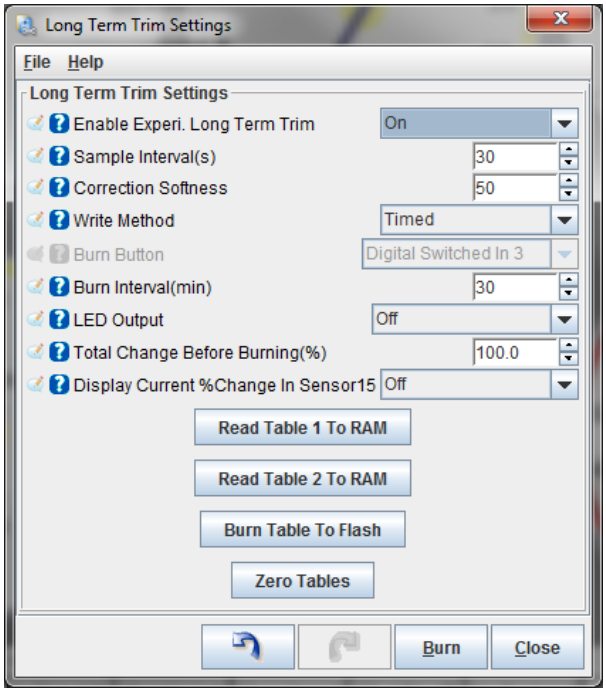
- **Digital switched 12V In Pull-down** - Causes the voltage on this pin to default to 0 volts. This ordinarily should not be changed unless you have modified the MS3Pro internally or connected a highly unusual circuit to this input.
- **Log Trigger Times to Sensors 9-16** - This enables an experimental feature to record engine trigger times to sensors 9-16 to monitor per-cylinder effort. Only for developers.
- **SDcard log download** - SD card logs use a fixed size block of data. Setting this to “Useful data” stops the download once the MS3Pro reaches a point where the data stops. “Whole file” continues the download until the end of the space the card has assigned for the file.
- **SDcard detect** - If set to “Use detect switch”, the MS3Pro will only attempt to access the card if the switch in the SD card socket shows there is a card present. “Always in” overrides this and will attempt to connect to the SD card even if the switch indicates there is not a card present. Use this if you need to bypass a damaged SD card detection switch.
- **Fuel flow / consumption calculation interval** - How often the MS3Pro updates fuel consumption calculations, in milliseconds.

The SpkAdj values can place limits on a remote device modifying the spark timing over CAN. Currently, the most likely device to do this is a transmission controller; the MicroSquirt transmission control firmware can retard spark while shifting to reduce wear and tear on the transmission. Other devices may also adjust timing in the future.

- **Maximum SpkAdj Value** - Sets a limit on the maximum adjustment a remote device can apply to the ignition timing. This is a positive number, meaning advance.
- **Minimum SpkAdj Value** - Sets a limit on the minimum adjustment a remote device can apply to the ignition timing. This is a negative number, meaning retard.

7.10.21 Long term fuel trim

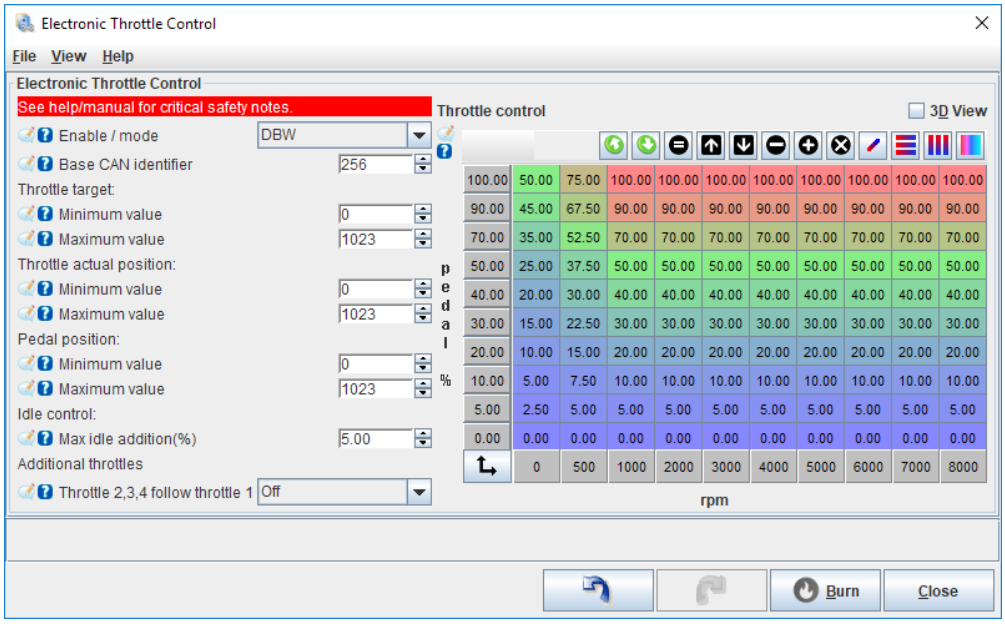
The MS3Pro can store a long term trim table to flash memory, based on oxygen sensor feedback. Enables the long-term fuel trim. This is intended to automatically make minor corrections in fueling **after** the engine is fully tuned. It requires EGO control to be enabled and operational.



- **Enable experimental long term trim** - Turns on the ability to store and write long term fuel trim tables.
- **Sample interval** - How often, in seconds, the code checks the amount of EGO correction being applied.
- **Correction softness** - Controls how aggressively the trim is applied. Larger numbers mean a softer amount of correction.
- **Write method** - Allows you to select writing the trim table to flash based on timed updates or triggered with an external button.
- **Burn button** - If using button input, specifies which input triggers writing the long term trim table.
- **Burn interval** - Minimum time, in minutes, before updating the long term trim to flash.
- **LED output** - Turns on an output when the amount of trim change exceeds the total change threshold, to let you know the ECU needs the trim to be burned to flash.
- **Total change before burning** - In timed write mode, the MS3Pro will automatically burn the trim table to flash if it changes by this large a percentage. In button mode, this will turn on the LED indicator.
- **Display current % change in sensor 15** - Allows using the Sensor 15 generic sensor to read the percent of change to long term trim.
- **Read Table 1 to RAM** - This takes the saved table #1 and copies it to the working RAM area.
- **Read Table 2 to RAM** - This takes the saved table #2 and copies it to the working RAM area.
- **Burn Table to Flash** - This takes the saved table #1 and copies it to the working RAM area.
- **Zero Tables** - This resets the working table and two saved tables to zero trim. It should be used after tuning the VE table.

7.10.22 Throttle control (DBW)

The MS3Pro does not operate a drive by wire throttle directly, but it can interface with external control units over the CAN network. Currently, AMP EFI does not offer a drive by wire controller, but there are third party products that support the MS3 DBW CAN messages. AMP EFI has not evaluated these modules for safety.



- **Enable / mode** - Idle control mode uses the device entirely for idle control, with the primary throttle controlled by a cable. DBW uses the device as a primary throttle.
- **Base CAN identifier** - The number the CAN device uses to identify messages for throttle control.

Throttle target, throttle position, and pedal position are normally handled as 10-bit values, giving a scale of 0 to 1023. You can adjust the minimum and maximum values if the device does not use the entire range.

- **Max idle addition** - The most additional throttle opening that can be handled as an added value to the throttle opening.
- **Throttle 2, 3, 4 follow throttle 1** - If set to off, the throttle targets for additional throttles are set to zero. If this is on, they follow the main throttle.
- **Throttle control table** - Sets a throttle opening percentage as a function of accelerator pedal position and engine RPM. All numbers are compared to the minimum and maximum values for throttle target, position, and pedal position. This can be used to limit throttle at low RPM. If the throttle responds like an on/off switch, you can tame its response by setting the throttle percentage numbers to be less than the pedal position below 20-30% pedal travel. If the you want faster throttle response to small pedal travel or for the car to “feel snappier”, you can set the throttle opening to a larger value than the pedal opening at lower pedal travel percentages. You can also set the throttle to close when you hit your intended maximum engine RPM as a throttle based rev limiting.

8 Troubleshooting and Data Logging

8.1 Resets

A reset is when a disruption in power or a spike in voltage causes the processor to briefly shut down. This will cause the engine to hiccup while running, and will display a couple other symptoms. The MS3Pro will reset when it is powered up, and this is the right time for it to reset. Having it reset on a running engine, however, is something you'll want to fix right away.

- The MS3Pro will briefly disconnect from the laptop or other devices on USB or RS232.
- Data logs will indicate a "Mark (number) - RESET".
- The SecL count will drop to zero.

The last item is a key part for spotting true resets in a data log. If the engine hiccups, but the SecL count does not drop back to zero, you've got something else, such as a fuel cut, loss of RPM signal, or other issue. If it's a real reset, here are the items we recommend checking first.

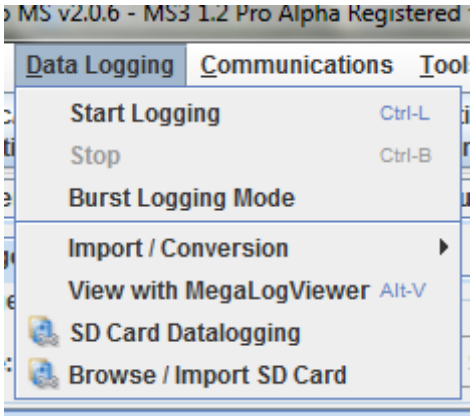
First, make sure the MS3Pro has all ground wire used, and grounded to the engine block, cylinder head, or negative terminal of the battery. Chassis grounds can have issues caused by rust or points where current runs through nothing but a thin spot weld.

Check where the MS3Pro is getting power from a clean source - ideally, you'll want it to be powered off the battery, with no devices that draw significant current getting power from the same feed. Many resets come not from EMI / RFI interference, but from something like an ignition coil or fuel pump injecting voltage spikes into the power feed.

Make sure that you are using resistor type spark plugs, good spark plug wires (if you're not running coil on plug), and that the spark plug wires are not within one inch of any wires going to or from the MS3Pro.

8.2 Capturing data logs

Data logs allow recording information from the engine for later analysis. These can help you troubleshoot your fuel tables, sensor readings, and more. You can record data logs from a laptop or with the MS3Pro's built in SD card. All of these are accessible from the Data Logging menu.



Use MegaLogViewer to play back the data logs.

8.2.1 Data logging with a laptop

To start a data log, go to the Data Logging menu and select Start Logging. TunerStudio will prompt you for a file name. You can go with the default date stamp, or enter a more descriptive name. The data log starts when you save the file. It stops when you either select Stop or close TunerStudio.

Burst logging mode shuts down gauge updates and several other features. We don't recommend using this except for logging on extremely slow devices.

Automatic logging enables TunerStudio to start or stop logging based on ECU variables or math expressions. For example, you can set TunerStudio to start logging when the RPM goes over a certain value, or when the SD card logging reports an error.

The engine does not need to be able to start successfully to record a data log. As long as TunerStudio is connected to the ECU, you can record a data log. Data logs can record failed attempted at starting, or even record sensor data with the key on and engine off.

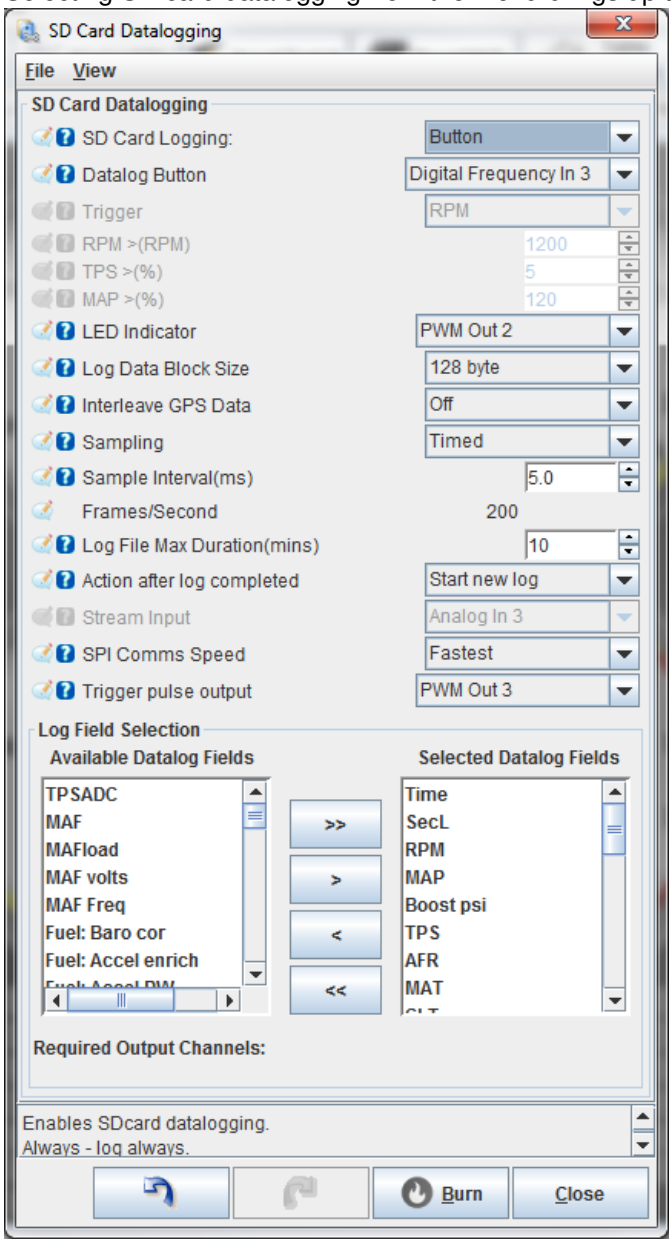
8.2.1.1 GPS logging with a laptop

TunerStudio can add GPS data from an external device such as the one sold by EFI Analytics. To enable this, go to Communications and select GPS Configuration. Check “GPS Enabled” and select how your GPS device is connected to your laptop.

8.2.2 Data logging with the SD card

The MS3Pro comes with an SD card built in for internal logging. The standard 8 gigabyte SD card can hold several days’ worth of continuous running data even at the fastest logging speeds. If you do manage to fill up the SD card, it will overwrite the lowest numbered files first.

Selecting SD card datalogging from the menu brings up this screen:



- **SD card logging**
  - Disabled - turns off the SD logging feature

  - Insertion - Logs at any time the MS3Pro contains an SD card. We do not recommend inserting or removing the card with the MS3Pro powered up.
  - Button - logging is controlled by a push button
  - Trigger - Logging is triggered by RPM or throttle position.

- **Datalog button** - selects the input pin used for the button if used.
- **Trigger** - Specify whether the log is to be triggered by RPM, MAP, or throttle position.
- **LED indicator** - selects the output used for the optional LED indicator. This output is on when SD card data logging is active.
- **Log data block size**
  - 64 byte is the most common setting
  - 64 byte + stream allows the simultaneous recording of data and a 8kHz mono sound input. This could be used as a commentary during a datalog such as position on a racetrack or notable incidents in the log. Presently the tuning software does not support extracting this audio data.
  - 128 byte allows logging twice as many settings, at half the maximum logging speed.
- **Interleave GPS Data** - Enables the recording of interleaved GPS data. Requires a CAN connected GPS module.
- **Sampling**
  - Timed - a data sample is taken at predefined time interval - this is the most common option
  - every tooth, every cycle - for special logging and at low engine speeds, these options take a data sample at these events
- **Sample interval** - how often to take a data sample. 3ms is the fastest supported, but many cards will only sustain 5ms before you start getting small gaps in the logs. See the utilities section for speed testing. The frames per second that this sample interval works out to is displayed below this setting.
- **Log file max duration** - how long each datalog should be. The files are always created full size and fill up the allocated space during logging. If the log file is filled, a new file is started. If you turn off the power to the MS3Pro (which is safe to do during a log) the log will be there and will show full size. When the log file is extracted, the unused part of the allocated file will be ignored. (Note, it is impossible for the MS3Pro to set the file length or otherwise "clean up" the log file in this case because there is no power and the processor is off !) A result of this is that all log files on the SD card will appear the same size.
- **Stream input** - where the 0-5V audio input is connected, if using the stream option.
- **SPI Comms Speed** - Allows you to adjust the speed if you have swapped the SD card for one that does not run at normal logging speed.
- **Trigger pulse output** - The MS3Pro will pulse this output at the start of recording, if using triggered or button mode. Use this to sync an SD card data log to a video recording or other external data acquisition.

Log field selection

The field selector allows you to choose which data fields you want to log. Because of the compact datalog format, only 64 or 128 bytes are available total, so you need to pick the data that you find most critical. To add a field, highlight it on the left and then click on > to move it to the right. To remove a field, highlight it on the right and then click on < to move it to the left.

Button usage

When MS3Pro powers up, it will create an empty datalog file ready for logging. Press and hold the button for over half a second to start a log. A quick press of the button will end the log. When extracting the log files you will likely find a blank log file on the card after your real logs, this is normal.

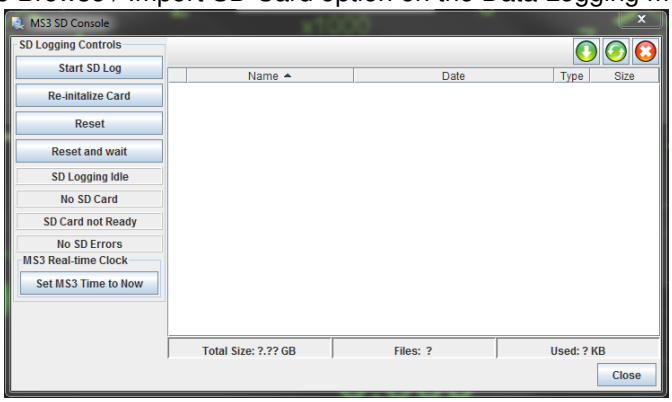
LED flash codes

- Off - SD logging not active

- On (solid) - Ready but not logging
- Fast flashing - Initializing card
- Slow flashing - Logging
- Pause, sequence of flashes - error code (See Error codes section)

8.2.3 Downloading SD data logs

The Browse / Import SD Card option on the Data Logging menu brings up the following screen.



The screen shot above shows the menu without a MS3Pro connected. If you have data logs on the SD card, they will appear in the list at the right hand side. Click the green “down arrow” button to save the selected logs to your computer. The middle button will reload the list of files, and the red X will delete the selected files.

8.2.4 SD card error codes

You can view SD card error codes by right-clicking on a gauge and swapping it to an “SD error” gauge. Some error codes are not used.

Error number	Error name	Notes
0	No error	SD logger is working correctly.
1	Sending clock error	Possible card fault. Try replacing SD card.
2	Idle error	Possible card fault. Try replacing SD card.
3	Init error	Possible card fault. Try replacing SD card.
4	Set block size	Possible card fault. Try replacing SD card.
5	Request CSD	The SD card being used is an unsupported type.
6	Error reading MBR	Erase files and reformat SD card.
7	MBR end marker not found	Erase files and reformat SD card.
8	No partition 1 defined	Erase files and reformat SD card.
9	Error reading partition boot sector	Possible card fault. Try replacing SD card.
10	Error reading directory	Possible card fault. Try replacing SD card.
12	Error reading file allocation table (FAT)	Possible card fault. Try replacing SD card.
13	Error reading FAT continuation sector	Possible card fault. Try replacing SD card.
14	Error writing FAT	Possible card fault. Try replacing SD card.
15	Error reading directory	Possible card fault. Try replacing SD card.
16	Error writing directory	Possible card fault. Try replacing SD card.
17	Error writing log sector	Possible card fault. Try replacing SD card.
18	Unsupported FAT type (not FAT16)	Erase files and reformat SD card.
20	SDHC detection error	Try reformatting SD card.
21	VCA check pattern failed	
22	VCA voltage rejected	
23	Operation Condition Register (OCR) error 1	Possible hardware fault. Try replacing SD card.
24	OCR2	Possible hardware fault. Try replacing SD card.
25	OCR3	Possible hardware fault. Try replacing SD card.
26	OCR4	Possible hardware fault. Try replacing SD card.
28	Error processing CSD	Try reducing data logging speed.
29	Write failed	Possible card fault. Try replacing SD card.
30	Directory full	Full SD card - delete data logs.
31	Can't find space in FAT	Full SD card - delete data logs.
32	Looks like VBR but not valid	Erase files and reformat SD card.

8.3 Data log fields

The data log fields represent not only the main sensor inputs but also a large number of the MS3Pro's internal calculations. Here is a complete list of what data you can view.

8.3.1 Understanding data log bit fields

Several of the data log fields are a bit code. To decipher a bit field, you need to determine what combination of numbers adds up to the final number. Do this by taking the first code number that is smaller than the value displayed, subtract that, and repeat the process until you've used up all the numbers. For example, suppose you have a CEL Status code of 15. Here's what you would do.

- The highest number on the chart that is less than 15 is 8. TPS fault. So we have a TPS fault present.
- Subtract 8 from 15, giving us 7. We'll need to see what codes make up the 7.
- The next lowest value is 4. CLT sensor fault. So there is also a CLT sensor fault.
- Subtracting 4 from 7 gives us 3. So we have more codes to find.
- The next lowest number on the chart is 2. MAT sensor fault.
- Subtracting 2 from 3 gives us 1. This is the last code, a MAP sensor fault.

The above combination of faults is likely to be a problem with the VREF voltage or sensor return wiring, if you run into this status in the real world.

8.3.2 Complete list of fields

These are the fields you can read with an MS3Pro on its own. Some of these fields are only logged if the appropriate settings requiring its use are enabled. CAN expansion devices may add more fields; see the documentation for your CAN device if you are using one.

- **Accel X, Y, Z:** These record the accelerometer data in the X, Y, and Z direction in meters per second.
- **AFR (1-8):** Air/fuel ratio readings. The MS3Pro can connect to four widebands on its own, and can read up to eight if using CAN based expansion devices.
- **AFR load:** The load variable being used for the AFR target table.
- **AFR Target 1:** The current target value for AFR, looked up from the active AFR target table.
- **AFR target 1 error:** The difference between actual and target AFR.
- **ALS added fuel:** The fuel pulsewidth (time) adder due to the anti-lag system.
- **Alternator control duty:** The duty cycle of the output being used to control the alternator.
- **Alternator control frequency:** Output frequency to control alternator, if using frequency based alternator control mode.
- **Alternator load:** Measured load on the alternator.
- **Alternator target voltage:** The voltage being targeted by the alternator control system.
- **Barometer:** The barometer reading. This will show a current value if a dedicated baro sensor is installed, the initial MAP reading, or 100 kPa if barometric correction is disabled.
- **Battery Voltage:** The measured battery voltage. If this is radically different from your actual battery voltage, check for wiring faults.
- **Battery Current:** The measured battery current measured through the alternator control system.
- **Boost Duty 1,2:** The current duty cycle output to the boost solenoids 1 and 2.
- **Boost psi:** Boost in psi, calculated by subtracting the Barometer field from the MAP field.
- **Boost Target 1,2:** The target boost pressure on boost outputs 1 and 2 (in absolute kPa) for closed-loop mode.
- **CANin1_8:** Records on/off inputs from a remote CAN device. See “Data log bit fields” for how to calculate which outputs are off and on.

128	64	32	16	8	4	2	1
Channel 8 on	Channel 7 on	Channel 6 on	Channel 5 on	Channel 4 on	Channel 3 on	Channel 2 on	Channel 1 on

- **CANout1_8, CANout9_16:** Records on/off outputs from a remote CAN device. CAN channels work as above for 1-8. For 9-16, add 8 to the channel number.
- **CANpwmin0-3:** PWM inputs from a remote CAN device.
- **CEL error code:** Current error code if the check engine light is triggered. See section 7.10.10.1 for details.
- **CEL status:** This bit field contains information about what has triggered the check engine fault. The first status is the only 16 bit bitfield currently in use.

128	64	32	16	8	4	2	1
EGT fault	RPM loss of sync	AFR input fault	Battery voltage fault	TPS fault	CLT sensor fault	MAT sensor fault	MAP sensor fault

32768	16384	8192	4096	2048	1024	512	256
AFR shutdown	EGT shutdown	Fuel pressure fault	Oil pressure warning	Cam fault	Knock fault	MAF fault	Flex fuel sensor fault

- **CEL status 2:** Additional data on what triggered the check engine light.

128	64	32	16	8	4	2	1
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Safety shutdown	Water injection fluid low

- **Closed loop idle target RPM:** The target engine speed for closed-loop idle control.
- **Closed-loop idle RPM error:** The difference between actual rpm and target rpm for closed-loop idle control.
- **CLT:** The measured coolant temperature (CLT).
- **Duty Cycle 1:** The calculated injector duty cycle on channel 1. This is made by taking the calculated pulse width and dividing by the available time to fire the injector. It will display a value over 100% if the calculated pulse width exceeds the available time. We recommend upgrading your injector size if this value goes above 85%.
- **Duty Cycle 2:** The calculated injector duty cycle on channel 2.
- **Dwell:** The ignition dwell (charge) time on the main spark outputs. Note that if using an ignition module that controls dwell (such as a Ford EDIS, MSD ignition, or some Ford or Bosch ignition modules), the MS3Pro will not report actual dwell time; it will instead report the length of the pulse it is sending to the module.
- **E85 Fuel Correction:** The percentage multiplier to fueling due to the flex fuel sensor.
- **EAE1:** The percentage multiplier to fueling due to Enhanced Acceleration Enrichment channel 1.
- **EAE2:** The percentage multiplier to fueling due to Enhanced Acceleration Enrichment channel 2.
- **EAE load:** The load variable used on the Y-axis of the EAE curves.
- **EGO cor 1-8:** The percentage fuel correction from AFR1-8 when closed-loop EGO control is used (100% means no change.)
- **EgoV 1 Target:** The current target value for EGO voltage (1,2), looked up from the active target table when using a narrow band O_2 sensor.
- **EgoV 1 Error:** The difference between actual and target EGO voltage.
- **EGT 1-16:** The reading from an external EGT sensor.
- **Engine:** “Engine” is a single status field with many basic information items. See the notes on bit fields for how to interpret these numbers.

128	64	32	16	8	4	2	1
MAP decel fuel cut	MAP accel enrichment	TPS decel fuel cut	TPS accel enrichment	Warmup enrichment	After-start enrichment	Cranking	Ready (RPM or fuel pump on)

- **Ethanol Percentage:** The calculated ethanol percentage in the fuel reported from the flex fuel sensor.
- **Fuel: Air cor:** The percentage multiplier to fueling due to air density.
- **Fuel: Accel enrich:** The current fuel pulsewidth percentage multiplier due to acceleration enrichment.
- **Fuel: Accel PW:** The current fuel pulsewidth (time) adder due to acceleration enrichment.
- **Fuel: Baro cor:** The percentage multiplier to fueling due to barometric correction.
- **Fuel: Total cor:** The total fuel percentage multiplier obtained by multiplying the other factors. If this is outside of the range 80%- 120% then there is likely something wrong with your tuning settings unless you are running in extremely cold weather or very high altitude.

- **Fuel: Warmup cor:** The percentage multiplier to fueling due to Warmup and After-start. Once the engine is fully warmed up, this should be 100%.
- **Fuel flow cc/min:** Calculated averaged fuel flow.
- **Fuel Pressure1_xxx:** Primary differential fuel pressure in bar, kPa or bar. This is the differential fuel pressure above intake manifold pressure.
- **Fuel Pressure2_xxx:** Secondary differential fuel pressure in bar, kPa or bar. This is the differential fuel pressure above intake manifold pressure.
- **Fuel pump control duty:** The output duty cycle to control the fuel pump, if using variable speed fuel pump control.
- **Fuel Temp 1,2:** Fuel temperature. 1 = primary fuel, 2 = secondary fuel if using dual fuel mode.
- **Fuel temperature cor:** Correction to fuel due to fuel temperature.
- **Fuel pressure cor:** Correction to fuel due to fuel pressure.
- **Gear:** The current transmission gear.
- **Generic Closed-Loop A,B duty:** Output duty cycle from Generic Closed-Loop system.
- **GPS Altitude:** Altitude in meters from external GPS. The GPS device is normally plugged into the laptop doing the data logging, not the MS3Pro.
- **GPS Course:** The course/heading from external GPS.
- **GPS Latitude:** Latitude degrees from external GPS. (Combination of degrees, minutes, seconds as one decimal number.)
- **GPS Lat. deg:** Latitude degrees component from external GPS.
- **GPS Lat. min:** Latitude minutes component from external GPS.
- **GPS Lat. sec:** Latitude seconds component from external GPS.
- **GPS Longitude:** Longitude degrees from external GPS. (Combination of degrees, minutes, seconds as one decimal number.)
- **GPS Lon. deg:** Longitude degrees component from external GPS.
- **GPS Lon. min:** Longitude minutes component from external GPS.
- **GPS Lon. sec:** Longitude seconds component from external GPS.
- **GPS Lon. E/W:** Longitude East/West indicator from external GPS.
- **GPS Speed:** The speed from external GPS in your chosen speed units.
- **GPS Speed ms-1:**The speed from external GPS in meters per second.
- **Ign Load:** The load variable used on the Y-axis of the ignition tables.
- **Injector Timing pri:** The injection timing used for primary injector channels.
- **Injector Timing sec:** The injection timing for secondary injector channels.
- **Knock cyl# 1-12:** The signal value per-cylinder from the internal knock sensor interface.
- **Knock In:** The signal value from an optional add-on knock sensor interface.
- **l/100km:** Fuel economy in liters per 100 kilometers.
- **Lambda (1-8):** Lambda reading from wideband EGO sensors 1-8.

- **Lambda 1 Target:** The current target value for Lambda (1,2), looked up from the active target table.
- **Lambda 1 Error:** The difference between actual and target Lambda.
- **Launch timer:** The time after launch control has been released, if using timed retard or timed boost after launch.
- **Launch timing:** The retardation to the timing due to the timed-retard after launch.
- **Load:** The primary load variable for fuel calculations. This is equal to MAP on typical speed-density installations.
- **Long term fuel trim cor:** Fuel correction due to long term trim.
- **Loop:** This bitfield is used with the Programmable on/off outputs feature and allows other features to be controlled by user-defined combinations. The number will tell you which of the three Loop settings has been triggered.

128	64	32	16	8	4	2	1
Loop 8	Loop 7	Loop 6	Loop 5	Loop 4	Loop 3	Loop 2	Loop 1

- **Lost sync counter:** The count of "lost-sync" events. If the engine stalls the counter will increase by one. During startup it is acceptable to have a few lost syncs. If you see this counting up under other conditions while the engine is running, there is a problem with your tach input.
- **Lost sync reason:** A technical reason code for why the ECU lost synchronization with the tach input. See section 8.4.1 for an explanation of the numbers.
- **MAF:** The air-flow in g/sec calculated from the MAF sensor.
- **MAFload:** A calculated load value that works similarly to MAP in a speed-density system. MAFload is equal to the MAF sensor measurement divided by the theoretical air flow at the current engine RPM if the engine had 100% volumetric efficiency.
- **MAF_volts:** The raw voltage input from a voltage MAF sensor.
- **MAF_freq:** The raw frequency input from a frequency MAF sensor.
- **MAP:**The measured MAP sensor reading. On typical speed-density installs this should read from 90 to 100kPa during key-on, engine-off (depending on altitude and weather conditions) and between 30kPa and 60kPa at idle.
- **MAP accel:** MAP based component of acceleration enrichment.
- **MAPdot:** The rate of change of MAP in kPa per second. Positive numbers indicate the MAP is increasing and negative numbers indicate it is decreasing.
- **MAT:** The measured intake manifold air temperature (MAT). Also known as intake air temperature.
- **MPG(UK):** Fuel economy in miles per UK gallon (4.546 liters).
- **MPG(USA):** Fuel economy in miles per US gallon (3.8 liters.)
- **Nitrous 1,2 duty:** The duty cycle to the nitrous solenoids in progressive mode.
- **Nitrous added fuel:** The fuel pulsewidth (time) adder due to the nitrous system.
- **Nitrous timer:** The delay timer for the nitrous system.
- **O2 (#1-8):** Input voltage from narrowband EGO sensors 1-8.
- **Odometer Miles:** The calculated mileage based on VSS input.
- **Port:** These fields read the actual status of the input and output pins at the processor. Note that the ports that drive the fuel and ignition outputs, cam / crank inputs, and PWM signals, may not be sampled fast enough to tell for certain if these outputs are active. This chart lists both the input or output and the processor port that corresponds to each number.

Port bitfields Port A

128	64	32	16	8	4	2	1
Injector H	Injector G	Injector F	Injector E	Injector D	Injector C	Injector B	Injector A
A7	A6	A5	A4	A3	A2	A1	A0

Port B

128	64	32	16	8	4	2	1
Spark H	Spark G	Spark F	Spark E	Spark D	Spark C	Spark B	Spark A
B7	B6	B5	B4	B3	B2	B1	B0

Port EH

128	64	32	16	8	4	2	1
Digital switched 12V in	Digital switched in 3	-	Fuel pump	-	Digital switched in 1	Digital Switched In 5	Digital Switched In 4
H7	H6		E4		E2	E1	E0

Port K

128	64	32	16	8	4	2	1
-	-	-	-	Internal	Digital switched in 2	Internal	Tach out
K7	K6	K5	K4	K3	K2	K1	K0

Port MJ

128	64	32	16	8	4	2	1
Knock interface	Stepper enable	Knock interface	Knock interface	Knock interface	Knock interface	Stepper out 2	Stepper out 1
J7	J6	M5	M4	M3	M2	J1	J0

The stepper enable output is on whenever either stepper output is active.

Port P

128	64	32	16	8	4	2	1
PWM / Idle Out 1	PWM Out 3	High Current Out 2	High Current Out 1	High Current Out 3	PWM Out 2	-	-
P7	P6	P5	P4	P3	P2	P1	P0

Port T

128	64	32	16	8	4	2	1
	Digital I/O 5	Digital frequency in 2	Digital frequency in 1	Injector J	CMP	Injector I	CKP
T7	T6	T5	T4	T3	T2	T1	T0

- **PW1:** The base fuel pulsewidth for injector channel 1. This is the actual electrical pulsewidth including dead-time, but not trim. See also SeqPW1.
- **PW2:** The base fuel pulsewidth for injector channel 2. This is the actual electrical pulsewidth including dead-time, but not trim. Primarily for dual table or dual fuel.
- **PWM A-F duty:** Output duty cycle from Generic PWM system.
- **PWM Idle Duty:** The current duty cycle output to the idle valve.
- **RPM:** The tachometer / rev counter (if this is dropping to zero or bouncing around, you probably have a problem with the tach-in signal.)
- **RPMdot:** The rate of change of RPM (the increase in RPM per second.)
- **SD card error:** Error code for SD card datalogging system - see the SD logging section for a list.
- **SD card file number:** The current active datalog file number on the SD card.
- **SD card phase:** Technical code relating to SD card datalogging system.

- **SD card status:** Bitfield indicating current status of SD card datalogging system.
- **SecL:** A built in clock that counts up seconds since the MS3Pro was last turned on. If this suddenly resets to zero, you probably have a power connection problem.
- **Secondary Ign Load:** Like “Ign Load” but used for the secondary lookup table.
- **Secondary Load:** Like “Load” but used for the secondary fuel lookup table.
- **Sensor01-16:** The reading from a sensor converted into your chosen units through the Generic Sensors system. You can also give the sensors a name of your choice which will then be used in the datalog.
- **Seq PW1-8:** The base fuel pulsewidth for the injector on cylinder #1. This is the actual output electrical pulsewidth including dead-time, all corrections and trim. (The "Firing Order" setting is used to determine which cylinder is which as the injectors always fire in the same order. A,B,C...)
- **SPK: 3-step Timing:** The spark timing/advance taken from the 3-step system.
- **SPK: ALS Timing:** The spark timing/advance taken from the anti-lag system.
- **SPK: Base Spark Advance:** The spark timing/advance looked up from the ignition table before other adjustments are made.
- **SPK: CEL retard:** The retardation to the timing due to check engine light and limp mode.
- **SPK: Cold advance:** The advancement to the timing due to the cold advance curve.
- **SPK: External Advance:** The external advancement to the timing, e.g., from a transmission controller.
- **SPK: Flex Advance:** The advancement to the timing due to flex fuel.
- **SPK: Fuel cut retard:** The retardation to the timing due to over-run fuel cut.
- **SPK: Idle Correction Advance:** The advancement to the timing due to idle correction.
- **SPK: Knock Retard:** The retardation to the timing due to knock control.
- **SPK: Launch Timing:** The spark timing/advance taken from the launch (2-step) system.
- **SPK: Launch VSS Retard:** The retardation to the timing due to the speed based launch system.
- **SPK: MAT Retard:** The retardation to the timing due to the MAT retard curve.
- **SPK: Nitrous Retard:** The retardation to the timing due to the nitrous system.
- **SPK: Revlim Retard:** The retardation to the timing due to the rev limiter.
- **SPK: Spark Advance:** The overall ignition advance.This should match the actual timing you see with a timing light - if not, check the spark settings.
- **SPK: Spark Table 1:** The looked up spark advance from ignition table 1.
- **SPK: Spark Table 2:** The looked up spark advance from ignition table 2.
- **SPK: Spark Table 3:** The looked up spark advance from ignition table 3.
- **SPK: Spark Table 4:** The looked up spark advance from ignition table 4.
- **SPK: Traction retard:** The retardation to the timing due to traction control.
- **State:XX:** Whether the ECU sees the engine as currently in state XX or not.
- **Shaft rpm 1,2:** Shaft speed in RPM.
- **Status codes:** These describe which features are currently active. See “Data log bit fields” for how to see which bits are active.

Status code bitfields Status 1

128	64	32	16	8	4	2	1
Full RPM sync	Spark table switching	Fuel table switching	Sync latch	RPM sync	Config error	Lost data	Need to burn settings

"Sync latch" indicates the ECU has completed a page of data for a composite or tooth log.

Status 2

128	64	32	16	8	4	2	1
Closed loop idle	Overboost protection	Spark cut	Flat shift active	Launch control active	Launch control armed	Nitrous stage 2 active	Nitrous stage 1 active

Status 3

128	64	32	16	8	4	2	1
Launch on	Sequential shift cut	Soft rev limit	3 step armed	Test mode active	3 step active	T log	Fuel cut

Status 4 and 5 are development fields. You can set the Check Engine functionality to use Status 5 to record sensor ADC counts or input fluctuation levels.

Status 6

128	64	32	16	8	4	2	1
MAP sample error	Fan on	Idle advance mode	Idle VE table mode	AFR shutdown	AFR warning	EGT shutdown	EGT warning

Status 7

128	64	32	16	8	4	2	1
Limp home mode	Check engine light	A/C on	Knock detected	VVT4 error	VVT3 error	VVT2 error	VVT1 error

Status 8

128	64	32	16	8	4	2	1
Drive by wire fault	Reserved	VSS based launch on	VSS launch locked out	Reserved	Water injection low	Spark disable test mode	Injector disable test mode

Status 9

128	64	32	16	8	4	2	1
AFR or EGT safety shutdown	Boost table switching triggered	Shutdown warning	RequFuel switching triggered	Stoich ratio switching triggered	Overrun fuel cut	Fuel pressure triggered shutdown	Oil pressure triggered shutdown

Status 10

128	64	32	16	8	4	2	1
Long term fuel trim activated	CAN EGT 2 active	CAN EGT 1 active	IO Box 3 active	IO Box 2 active	IO Box 1 active	CAN TX	CAN RX

SD Status

128	64	32	16	8	4	2	1
Reserved	Reserved	Reserved	SD card error	SD logging active	SD card ready	Reserved	SD card in

- **Status: XX:** Bitwise indication of whether the state of parameter XX is currently met or not.
- **Stepper Idle position:** The current stepper valve position (0 means fully closed.)
- **Time:** TunerStudio maintains a time since it was started and records this is the datalog. The time is shown at the bottom of the graph area. This can be helpful if discussing a data log with someone online; you can note what the time reading was when a problem occurs.
- **Timing err:** A calculation of how accurately the position of the next tach tooth was predicted. As the data log sampling rate is very low compared to the frequency of the tach input, this number tends to bounce around and is not nearly as useful as a tooth or composite log for analyzing RPM input problems.

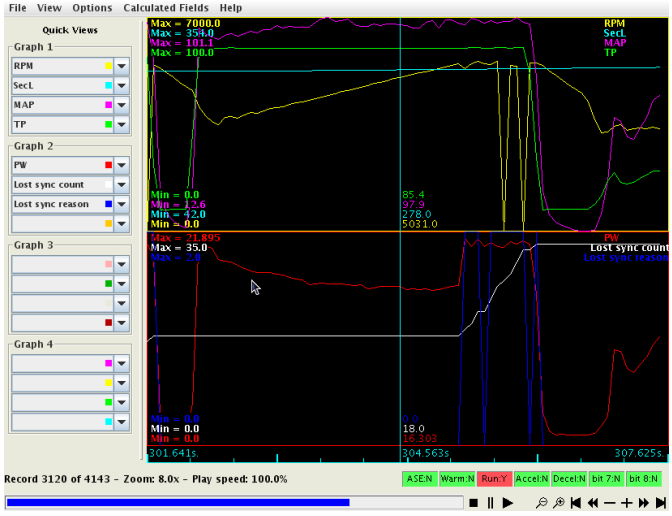
- **Total accel:** Combination of TPS and MAP based components of acceleration enrichment.
- **TPS:** The 0-100% throttle (pedal) position (check the calibration settings if you do not get 0-100%)
- **TPS accel:** TPS based component of acceleration enrichment.
- **TPSADC:** The raw ADC input of the throttle (pedal) position.
- **TPSdot:** The rate of change of TPS, in percent throttle opening per second.
- **VE1,2** The looked up VE value for channel 1,2. Blend tables are factored in.
- **VSS1-4:** Vehicle (wheel) speed sensor inputs in your chosen speed units.
- **VSS1-4 ms-1:** Vehicle (wheel) speed sensor inputs in meters per second.
- **VSS1dot, VSS2dot:** The rate of change of VSS (acceleration) measured in meters per second squared.
- **VVT angle 1,2,3,4:** The current relative angle of VVT cam 1,2,3,4
- **VVT duty 1,2,3,4:** The solenoid output duty cycle for VVT cam 1,2,3,4
- **VVT target 1,2,3,4:** The target angle of VVT cam 1,2,3,4
- **WallFuel1:** The calculated fuel puddle volume on the intake port walls, channel 1. Wall fuel numbers are used for model based acceleration enrichment.
- **WallFuel2:** The calculated fuel puddle volume on the intake port walls, channel 2.
- **Water Duty:** The duty cycle to the water injection solenoid (or pump).

8.4 Tooth/Trigger Logger

Certain conditions such as electrical noise can cause the MS3Pro to lose sync with the engine position. This is called a sync loss event or a sync error. Common symptoms of this include:

- Feels like a misfire or a false rev limiter
- RPM in datalog suddenly drops to zero
- RPM in datalog suddenly spikes up
- "Lost sync count" increments while "SecL" counts up normally. (NB. If SecL suddenly drops to zero this is a "reset"; see section 8.1.)

Taking and reviewing a datalog is a required step for troubleshooting this. This datalog extract below (viewed in Megalogviewer) shows the rpm (yellow trace) dropping to zero and at the same time the "Lost sync count" increases. The "Lost sync reason" shows reason code 2.



8.4.1 Lost sync numbers

The MS3Pro can report several different lost sync reasons in the data logs. Most of these simply show that there's a problem and don't particularly pinpoint where it is, but some are more specific than others. In particular, if you get Lost Sync Reason 11 or 17 in a specific RPM band, try changing the ignition input capture and /or second trigger active edge. Lost sync reason 2 with a VR sensor often means you will need a resistor across the CKP sensor.

0 no problem
1 init error
2 missing tooth at wrong time
3 too many teeth before missing tooth (last)
4 too few teeth before missing tooth (last)
5 1st tooth failed test
6 nonsense input (last)
7 nonsense input (mid)
8 too many teeth before missing tooth (mid)
9 too few teeth before missing tooth (mid)
10 too many teeth before end of sequence
11 too few teeth before second trigger
12 too many sync errors
13 dizzy wrong edge
14 trigger return vane size
15 and 16 - EDIS SAW signal error
17 second trigger not found when expected
20 Subaru 6/7 tooth 6 error
21 Subaru 6/7 tooth 3 error
22 Rover #2 missing tooth error
23 420A long tooth not found
24 420A cam phase wrong
25, 26, and 27 pattern does not otherwise match 420A
28 Pattern does not match 36-1+1
29 36-2-2-2 semi sync failed
30 36-2-2-2 tooth 14 error
31 Miata 99-00 - 2 cams not seen
32 Miata 99-00 - 0 cams seen
33 6G72 - tooth 2 error
34 6G72 - tooth 4 error
35 Pattern does not match Weber-Marelli
36 CAS 4/1 error
37, 38, and 39 Pattern does not match 4G63
40 and 41 Twin trigger error
42 Pattern does not match Chrysler 2.2/2.5
43 Pattern does not match Renix
44 Pattern does not match Suzuki Swift
45 and 46 Pattern does not match Vitara
47 Pattern does not match Daihatsu 3
48 Pattern does not match Daihatsu 4
49 Pattern does not match VTR1000
50 Pattern does not match Rover #3
51 Pattern does not match GM 7X
52 36-2-2-2 tooth 30 error
53 rc51 semi error
54 rc51 re-sync error tooth 6
55 rc51 re-sync error tooth 16
56 rc51 re-sync error tooth 18
57 fiat 1.8 tooth 0
58 fiat 1.8 tooth 3
59 fiat 1.8 tooth 6

60 fiat 1.8 tooth 9
61 36-1+1 first
62 36-1+1 second
63 QR25DE semi failed
64 QR25DE lost running sync
65 CAS360 running
66 NGC8 semi failed
67 LS1 semi failed
68 LS1 re-sync failed
69 YZF1000 re-sync failed
70 36-2+2 cam signal not present

Having identified that you have a sync loss problem, you need to attempt to eliminate it. Check for any basic problems:

- Loose sensor
- Loose cabling
- Sensor wires running close to spark plug wires or coils
- Slop in timing belt
- VR sensor wired backwards
- Fouled plugs causing misfire
- Non-resistor plugs
- Excessively rich mixture causing misfire
- Excessively lean mixture causing misfire

Having ruled out all those possible causes, you should enable some noise filtering. See the noise filtering settings, section 7.3.5.

8.4.2 Using the diagnostic logger

The tooth / trigger logger, accessible under TunerStudio's Diagnostic tab, lets you visualize crank and cam signals. It does not give you a voltage reading directly, but it will show you the state of the crank or cam signal, either on or off, as it reaches the MS3Pro's processor. Use this to troubleshoot issues with the MS3Pro's RPM input. There are several different modes. They work in a similar way:

The Logger Type setting selects which log type to use.

The Start, Stop, and Clear buttons work as you'd expect.

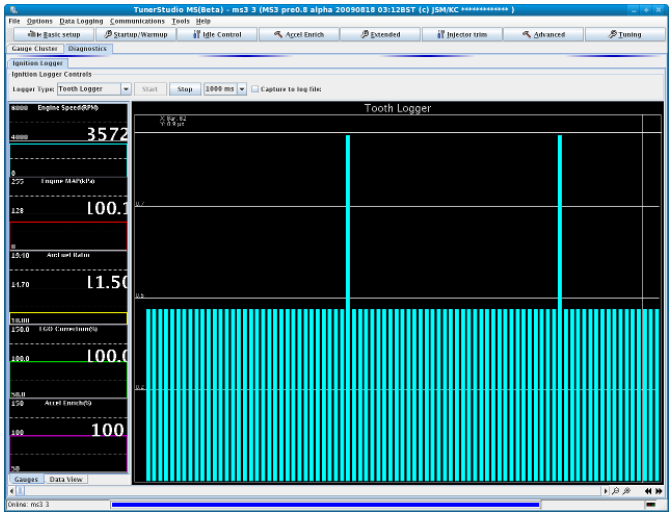
The Open Log button lets you pull up previous logs.

To capture a log to a file, first click the Capture Log to File button. You will be prompted to save a file. After you save the file, you may then start the log and collect data.

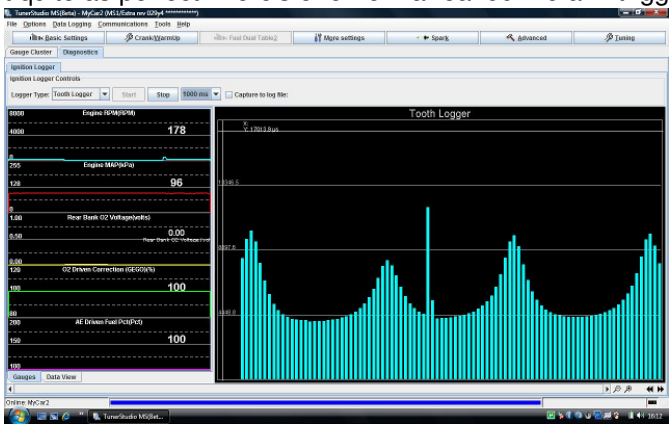
8.4.3 Tooth logger

This records the tach events that are received by the main tach input (crank) after any noise filtering. The height of the bar on the screen represents pulse length. The taller the bar, the longer the pulse. This one is useful for interpreting sync issues with a missing tooth type wheel.

Here's one with a simulated 36-1 wheel.



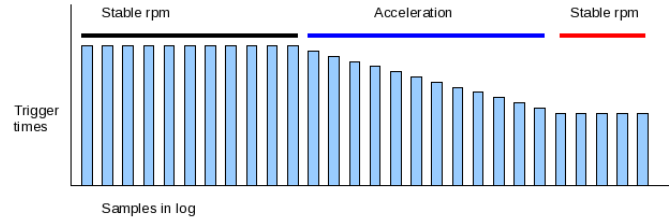
In the above 36-1 screenshot you can see 34 normal teeth and the single long "missing" tooth. Usually, the log isn't quite as perfect. Here's one from a real 60-2 crank trigger while cranking a motor.



In the above 60-2 screenshot you can see 57 normal teeth and the single extra long "missing" tooth. The cyclic nature of the teeth is due to the engine speeding up and slowing down during cranking. The OEM has place the missing tooth in a good place to avoid confusion with the compression effect. With poor choice it can be difficult for the code to reliably detect the tooth.

8.4.4 Trigger logger

This records the "decoded" and filtered tach events and is one pulse per normal ignition event. Since trigger logging is after the wheel decoding, this information is not very useful for diagnosing most sensor faults. It is only useful in spark modes that use an external device for wheel decoding, such as EDIS.



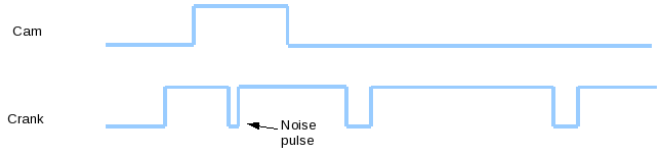
8.4.5 Composite logger

The basic tooth logger records enough information to analyze a typical missing tooth wheel on the crank. However, if the engine uses both cam and crank sensors, then a new logging method is required. This third composite logger records both crank and cam signals and under some circumstances it will record both edges of the input signal. It could be used as a crude two channel oscilloscope. The composite logger shows three signals.

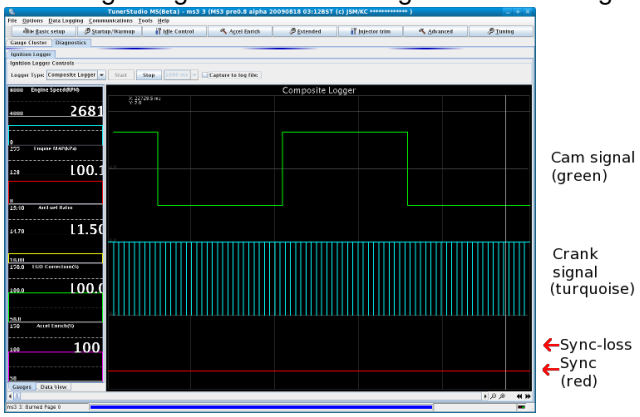
- The green line at the top represents the cam signal.
- The blue line in the middle represents the crank signal.
- The red line at the bottom represents error pulses that do not fit the expected pattern.

This log shows Subaru 6/7. This is a single edged signal, so the visualization can only show a point/line.

This log shows an example double edged signal with some noise.



Here is a screen shot of a simulated signal with a 24 tooth crank wheel and a 1 tooth cam wheel. The crank signal is single edged and the cam signal is double edged.

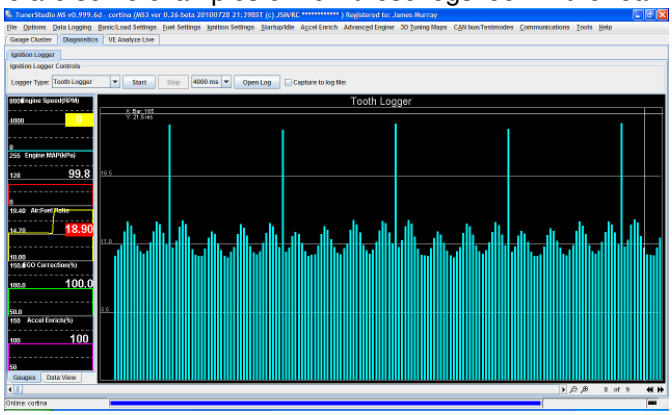


8.4.6 Sync error logger

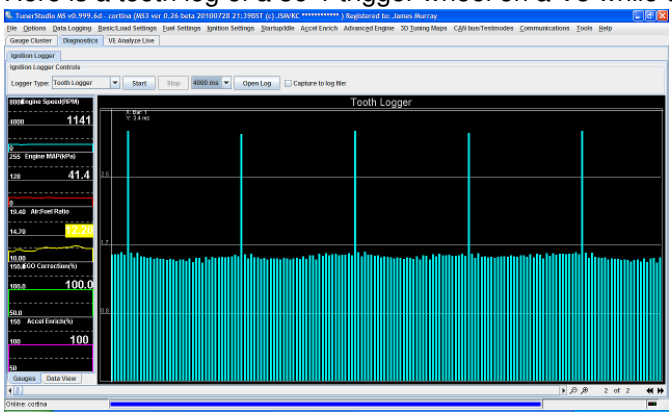
This is a variant of the composite logger. During normal engine operation it will show nothing at all, but if a sync-loss event occurs (due to noise) it shows the tooth pattern that led up to the loss of synchronization. This allows you to pinpoint the problem without searching for the needle in the haystack as would be the case trawling through large composite logs. To use this logger, it is usually useful to capture to a log file. Tick the "Capture to log file" box and save the file name. Then start the sync error logger and drive or ride around. (You can datalog at the same time.) When you have finished, Stop the logger and un-tick the "Capture to log file".

8.4.7 Example logs

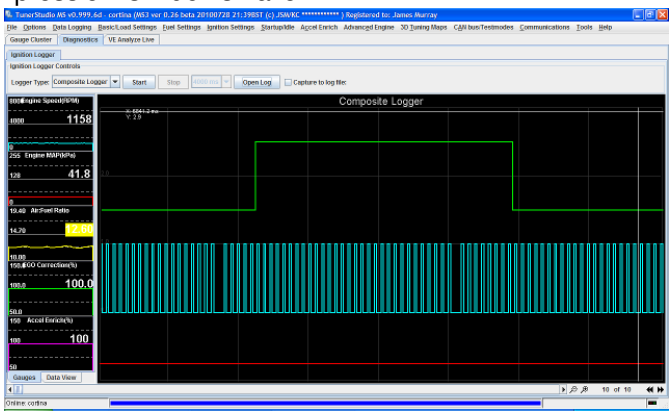
Here are some examples of how these logs look in the real world.



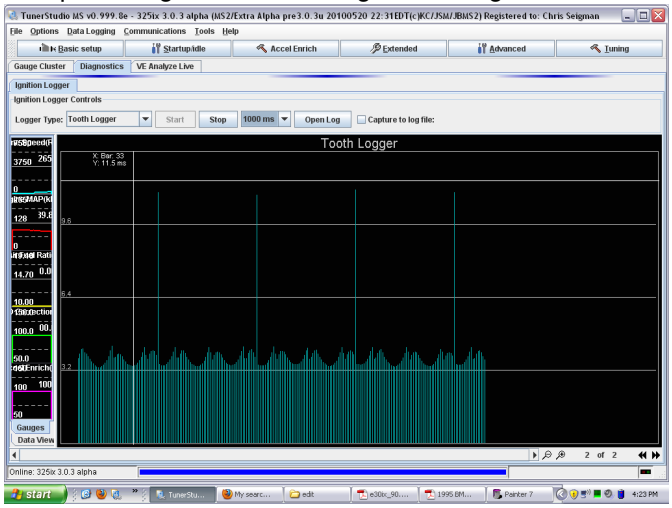
Here is a tooth log of a 36-1 trigger wheel on a V8 while cranking.



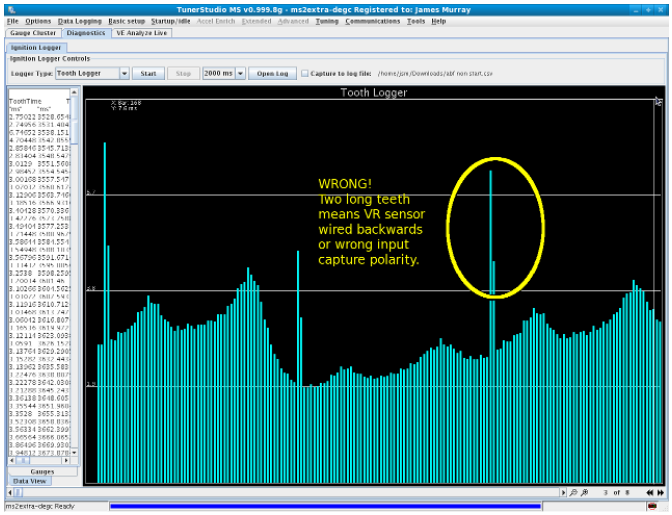
The same trigger setup, but at running speed. Note that the time on the teeth smooths out and the effect of the compression is much smaller.



Composite log of the same engine running.

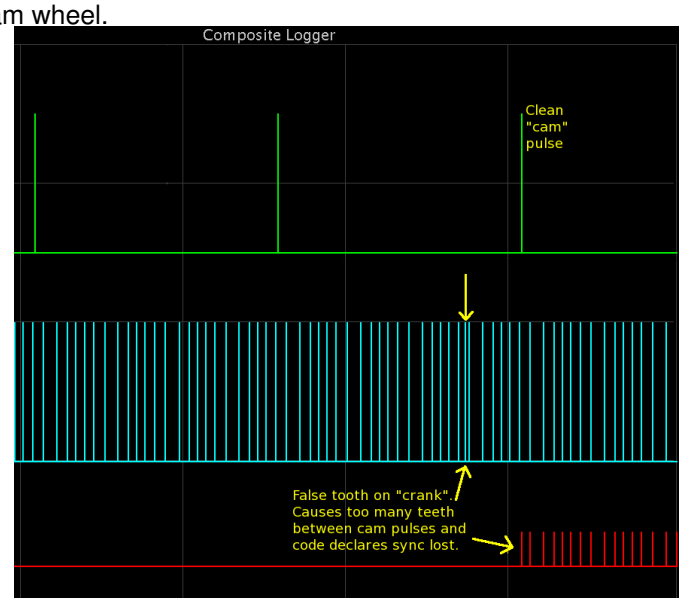


The above picture shows a 60-2 crank wheel during cranking. Note that the missing tooth is three times the height of the other tooth marks, instead of twice the height as on a 36-1.



Tooth log of 60-2 showing incorrect VR sensor wiring. Two somewhat long teeth instead of one very long tooth. In this situation the engine might get sync some of the time. Test it by change input capture polarity to Falling, if that helps then swap VR sensor wires and return input capture setting to Rising.

Composite log of a Nippon Denso ignition with a 24 tooth cam wheel (on the crank input channel) and a 1 tooth cam wheel.



Noise issues on the same ignition type. More noise filtering can reduce this problem.

8.4.8 Engine logger

While the tooth, trigger, and composite logs deal primarily with engine **inputs**, the engine logger provides high speed information about engine **outputs**. This records the crank signal like a composite log, but also records the status of the processor pins used to drive ignition and injector outputs. The engine logger can be used to detect abnormal behavior on fuel and ignition outputs, such as an unexpected spark output.

Tip: You can run the engine logger, as well as other high speed logs, at the same time as a data log. Capturing both logs simultaneously will let you verify that your logs recorded whatever problem you are trying to examine with the conventional data log, then let you take a closer look at the inputs and outputs in the high speed log. You can also look back to the standard data log to see if anything unusual, such as a temperature spike, may have triggered the unexpected input or output behavior.

8.5 Communication issues

Most reported user problems are actually a computer setup problem, so follow these steps to get your settings right. The MS3Pro package comes with a serial (RS232) and a USB cable; you may use either one to connect, but not (currently) both at one time. If your tuning computer has a built in RS232 port, that should “just work.” The USB port will work correctly with most versions of Windows, but sometimes you will need to set up driver software. Note that the MS3Pro cannot be powered through the laptop; it needs to have 12 volt power through its wiring harness before it will communicate. The USB connection will still show up on a computer, but it will not be able to send or receive data to or from the MS3Pro.

8.5.1 USB driver software

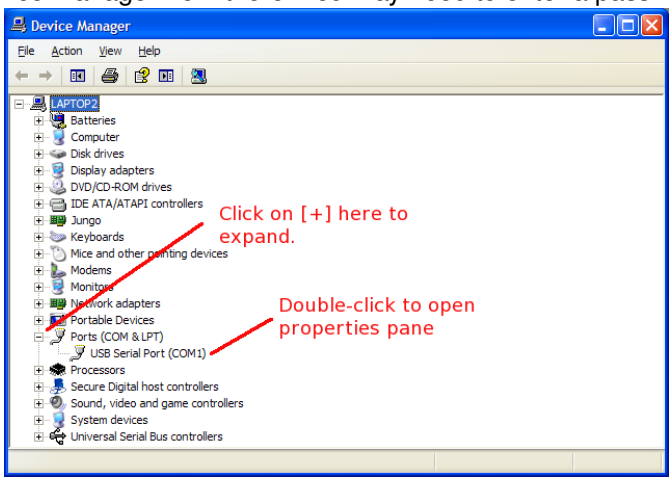
The MS3Pro USB connection uses an FTDI chipset with a Virtual COM Port (VCP) driver.

8.5.1.1 Windows drivers You may download the driver from this page on the FTDI website:

FTDI virtual COM port drivers

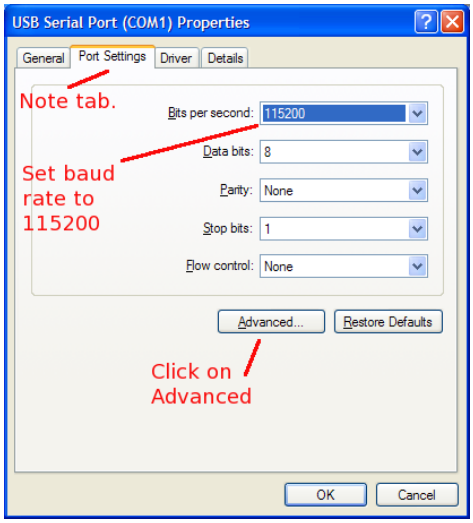
Having downloaded the driver, you should run it and the drivers should be installed. Then insert the USB cable and when Windows asks for a driver, you should be able to follow the automatic install process. Having installed the drivers, proceed with Device manager. There are several ways to do this, and you may have a preferred way. If you don’t know how to do this, here are directions for getting into the Device Manager for Windows 7 and newer:

Click the Start button and go to the Control Panel. From there, click System and Maintenance, then click on the Device Manager from there. You may need to enter a password.

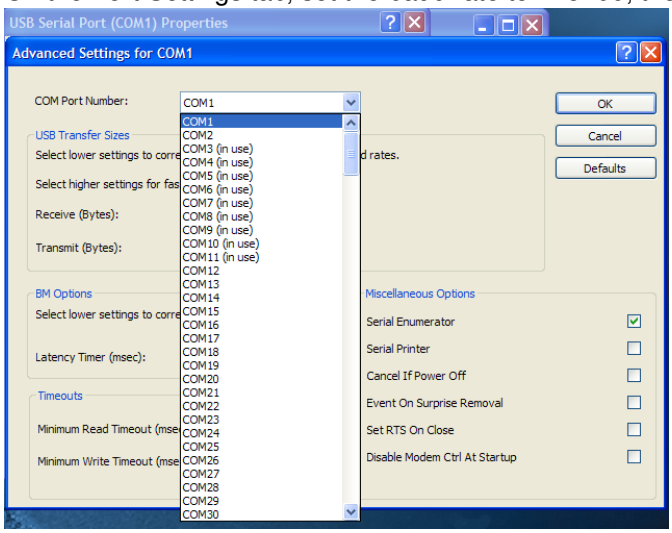


If you DO NOT see "Ports" listed in the devices, then you do not have any ports setup. This would usually indicate either your forgot to plug in your USB-serial cable to the MS3Pro, or there is a driver problem. Usually the best solution to the driver problem is to unplug the cable, uninstall the driver, and then re-install the driver.

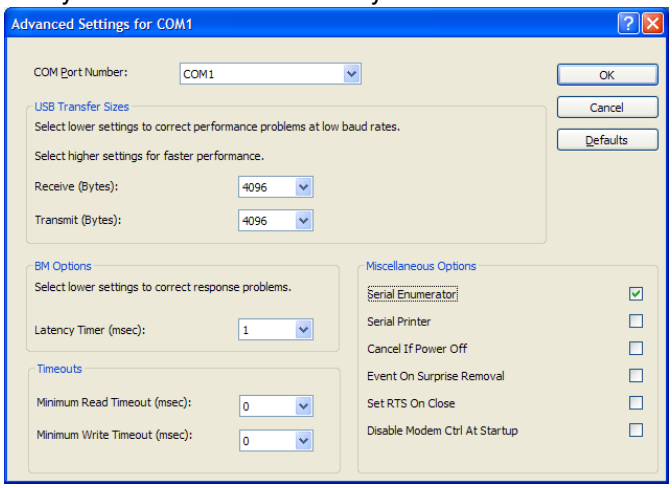
Find the Ports section and click the [+] to expand. Then double-click on the port number.



On the Port Settings tab, set the baud rate to 115200, then click on Advanced.



Ensure that the COM port number is 9 or less. If not, change it - TunerStudio can find higher numbered ports, but it will take longer to detect an MS3Pro on a high numbered port. It is usually safe to ignore any Windows warning about ports being in use unless you really do have a physical serial port or built-in modem. The port number you select here is the one you need to use in TunerStudio.



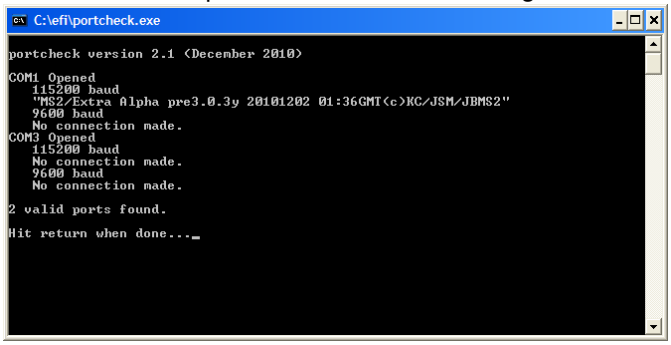
While on this page reduce the Latency Timer. 1 is normally OK. The default value gives very slow serial.
Now [OK] all windows to close them and then close the Device Manager.

8.5.1.2 Linux drivers USB serial adapters are typically used as /dev/ttyUSB0. All recent Linux kernels include support for the built-in FT232RL USB-serial adapter. You might want to use the lsusb command to check that the adapter has been recognized. dmesg can be used to see what port number has been assigned.

8.5.1.3 MacOS drivers Install the latest MacOS driver from the link below:
FTDI virtual COM port drivers
Then plug in the cable and power on the MS3. Either use Autodetect in TunerStudio or check the /dev directory for a port such as /dev/cu.usbserial-XXXXX .

8.5.2 Additional diagnostics

8.5.2.1 Portcheck The Port Check program is a small file, portcheck.exe, included for Windows. Port Check scans the available COM ports and sends them commands that will cause a MS3Pro or other ECU to display its signature. The example below shows it detecting an MS2 on COM port 1.



Note that to run Port Check, you will want any other programs that use the serial connection, like TunerStudio, closed. Sometimes it will report '"S"ing "S"' on a port at both speeds. This usually happens if it's trying to check a mouse port or other port that sends back any character it receives instead of sending new information.

8.5.2.2 Loading firmware One other way to test an MS3Pro that is not communicating and does not pass the port check program is to try reloading firmware. It is possible to corrupt the firmware in such a way that it does not respond to normal communications, but still responds to a command to load firmware. In such a case, loading firmware can reset the MS3Pro's settings. If the MS3Pro has power and ground, and is putting out a 5 volt signal on VREF, but does not respond to other communications tests, try this.

8.6 Loading Firmware

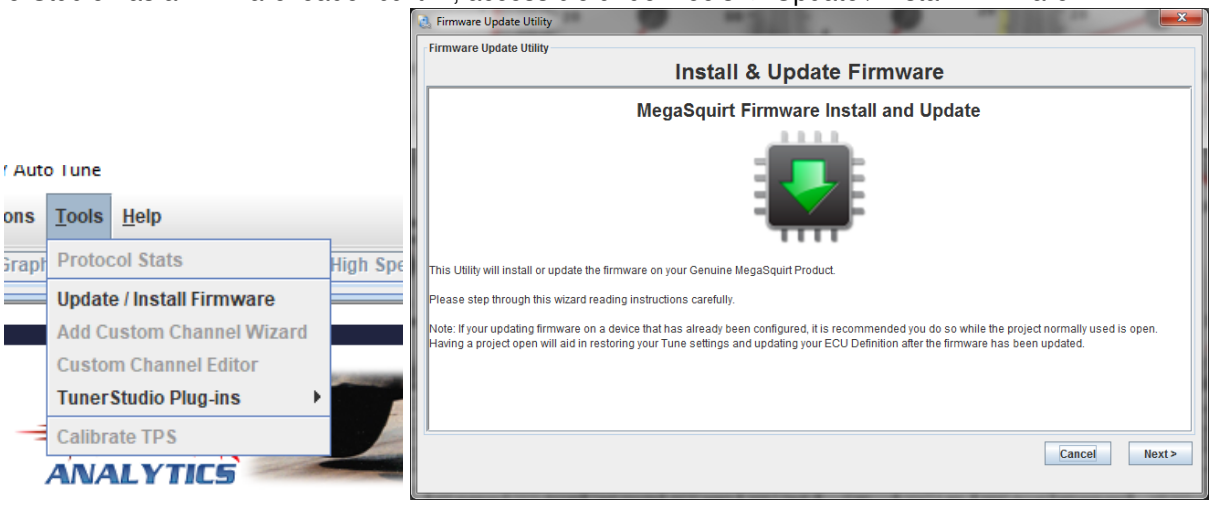
MS3Pros ship with the latest firmware on them, but we anticipate upgraded firmware being available. The firmware can also be reloaded if it has somehow been accidentally erased or loaded with settings that appear to have "bricked" the MS3Pro. The MS3Pro must be powered up when loading firmware, and you need to close any other programs that could use serial communication, such as TunerStudio (if using the external firmware loader), Palm HotSync, or modem software. The firmware loader installs with the CD.

While the firmware loader utility has an option for preserving sensor calibration and tune data, this option will not work if the existing tune is corrupted or if the firmware being loaded is too different from the existing firmware. You will want to save a back-up copy of the tune before loading firmware. The flip side of this is that if you load firmware without commanding the MS3Pro to preserve the current tune, loading the firmware will serve as a "hard reset" - it will wipe out all settings currently in the MS3Pro and restore them to factory defaults.

If the firmware crashes due to an error such as an impossible configuration that does not have a suitable error handling (one example on previous codes was setting a trigger wheel with two base teeth and two missing teeth, making a zero tooth trigger wheel), using the boot jumper will cause the firmware to look for a firmware update before checking any settings. This can often enable loading firmware to an MS3Pro when nothing else seems to work. To use the boot jumper, open the MS3Pro case and insert a paper clip across the two holes marked Boot on the back side of the MS3Pro Module board:

8.6.1 TunerStudio firmware loader

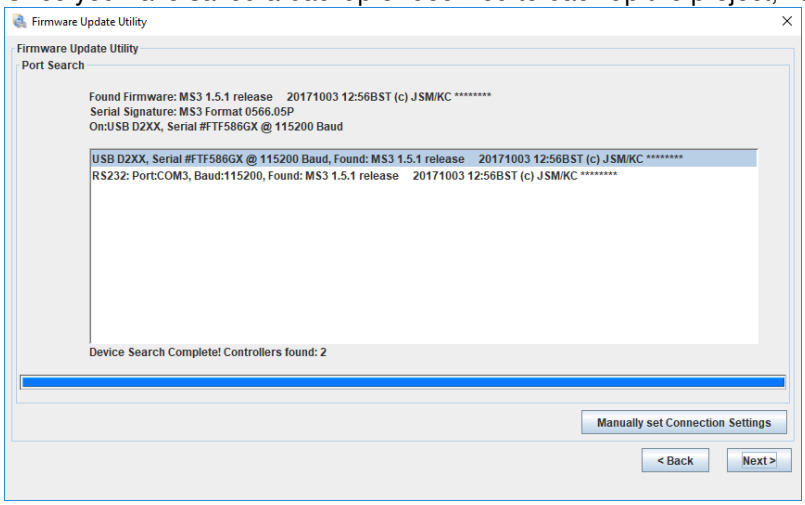
TunerStudio has a firmware loader built in, accessible under Tools -> Update / Install Firmware.



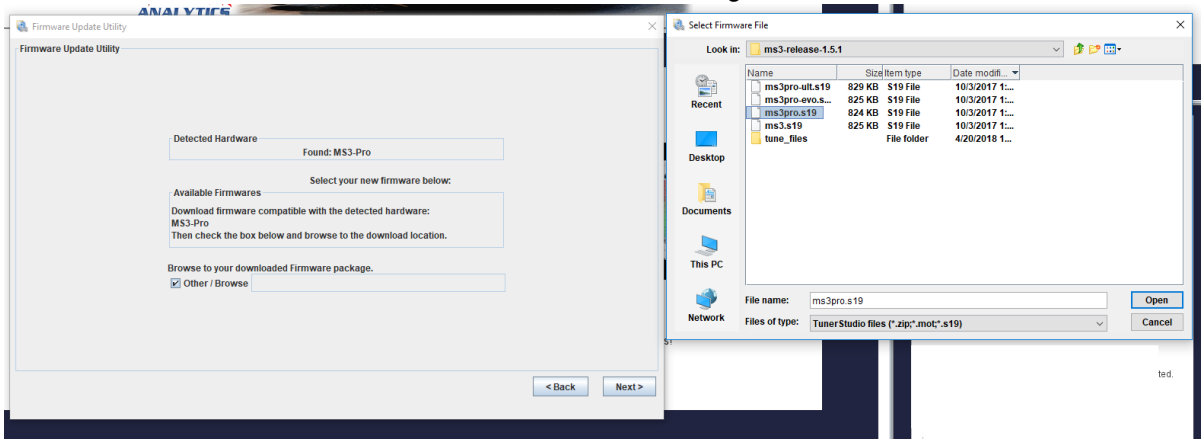
Following the onscreen prompts will allow you to load firmware to your MS3Pro. Note that in some cases TunerStudio can load firmware even if it is otherwise unable to communicate with the MS3Pro by any other means.

If you start the firmware loader with a project open,TunerStudio will prompt you to back up your project. This is optional, but it will make sure you have a backup of your tune before you reload firmware.

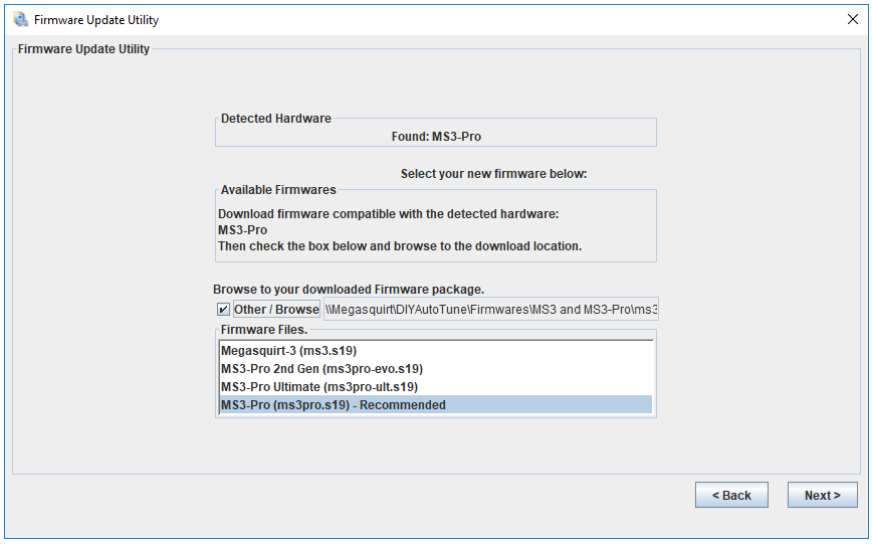
Once you have saved a backup or declined to back up the project, TunerStudio will search for your MS3Pro.



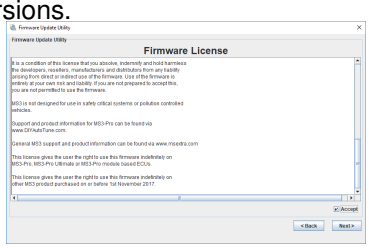
You will then need to browse for the correct .S19 file containing the firmware to load to the ECU.

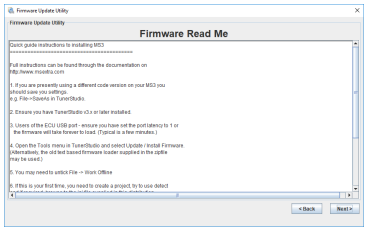


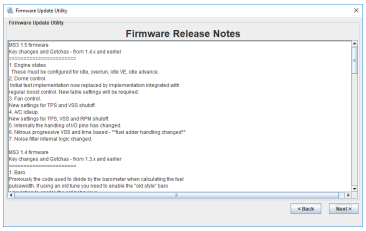
TunerStudio will then confirm the firmware is for your MS3Pro version (first generation MS3Pro shown below).



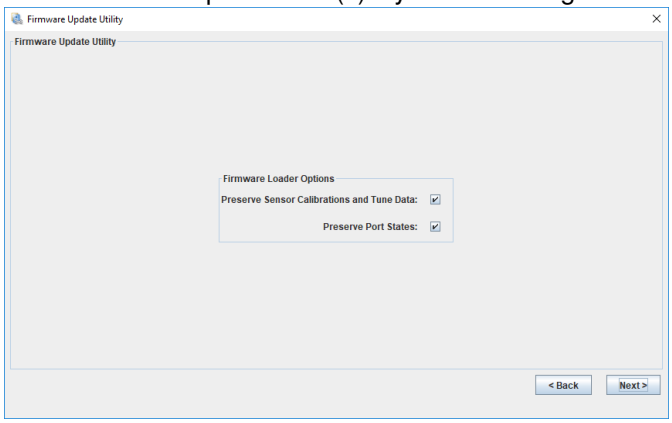
TunerStudio will then display the firmware licensing information, a readme file with general notes on loading the firmware, and release notes which will cover differences between the firmware you are loading and previous versions.



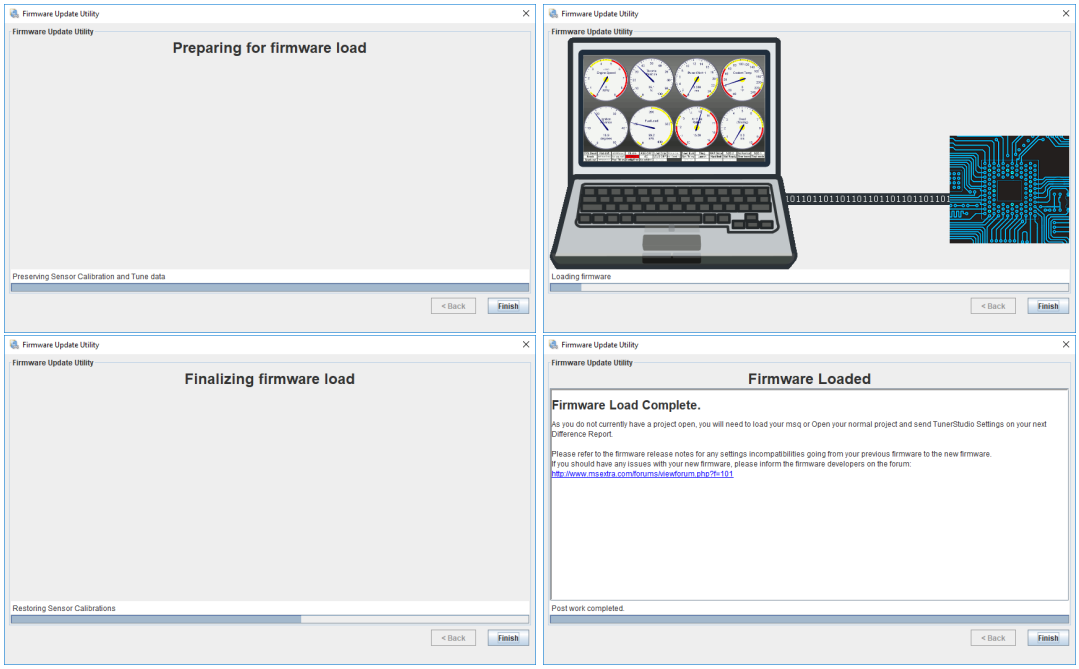




TunerStudio will then check if you want to retain settings and keep the output ports in their current states. If you are reloading the firmware due to communications issues, you should not attempt to preserve the current tune or port states. Examples of port states would be holding spark outputs in an off state. Note that you can still cause damage to coils if they are plugged in if an error occurs during the process; we recommend disconnecting the ignition coils or their power fuse(s) if you are loading firmware with the ECU in the car.

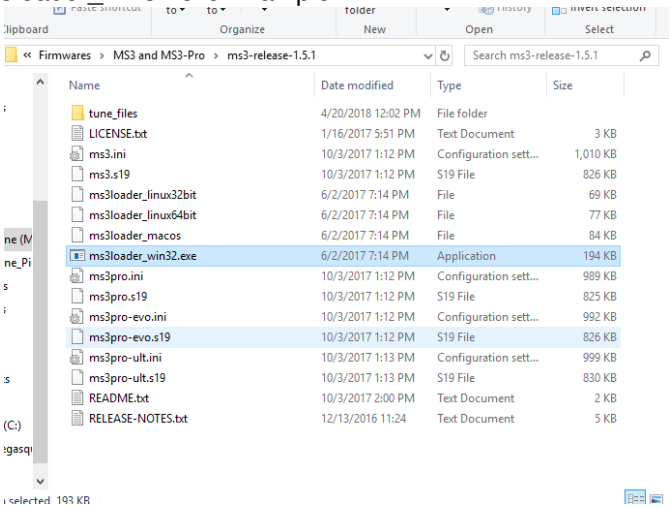


At this point, TunerStudio will reload the firmware. Once this is complete, TunerStudio can reload the project settings. If you have changed the firmware, TunerStudio will attempt to adapt the tune for the new firmware, but it may produce warning messages or errors. Check the settings mentioned in the warning messages; these parameters may need to be set manually.

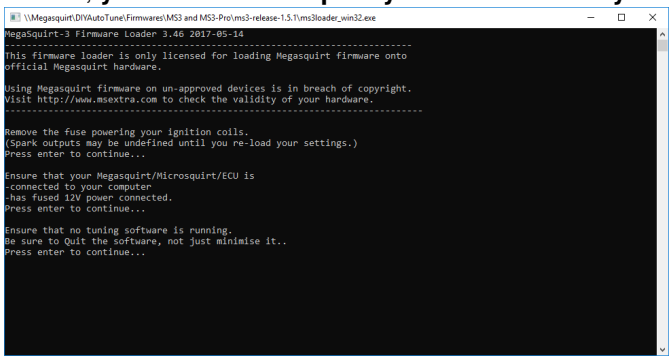


8.6.2 Windows firmware loader

This download utility is text based. Browse to the directory where you extracted the zip and double-click on ms3loader_win32.exe Example:

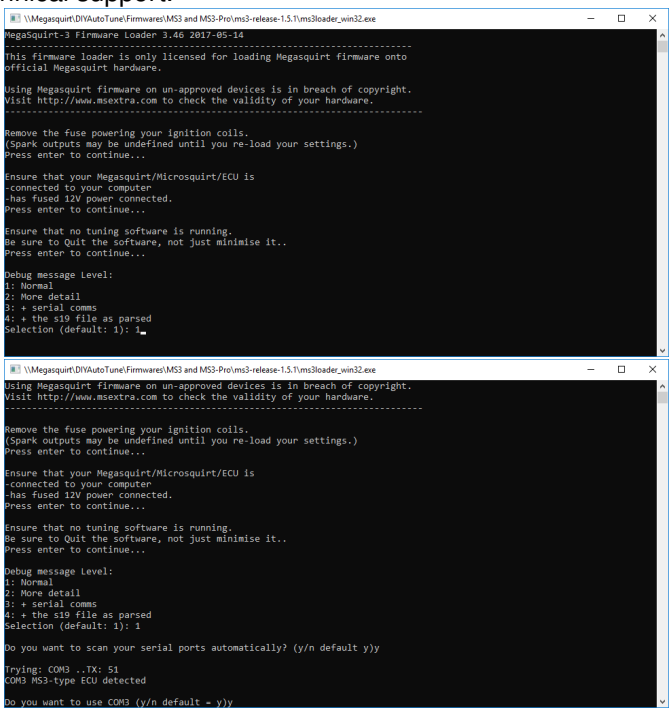


The firmware loading program will launch. Follow the onscreen prompts. Note that when it prompts you for the file to load, **you will need to specify the firmware for your MS3Pro version.**

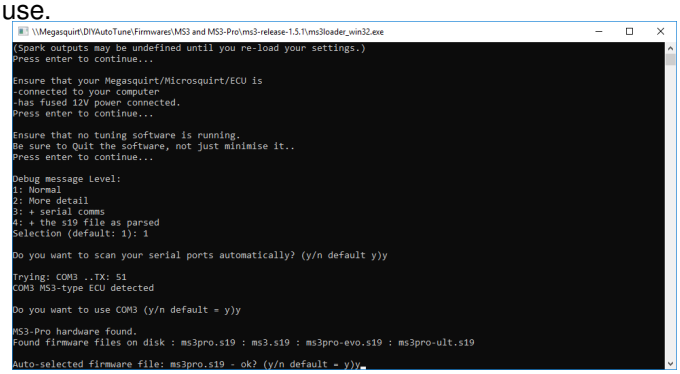


Debug message levels allow you to select how much information is displayed. These screen shots show loading firmware with debug level 1. Occasionally you may need to enable a higher debug level if requested by MS3Pro

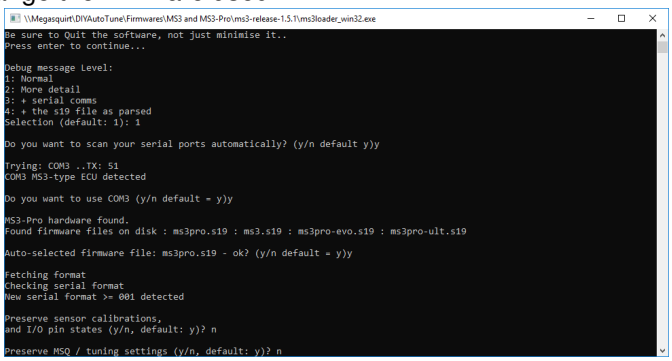
technical support.



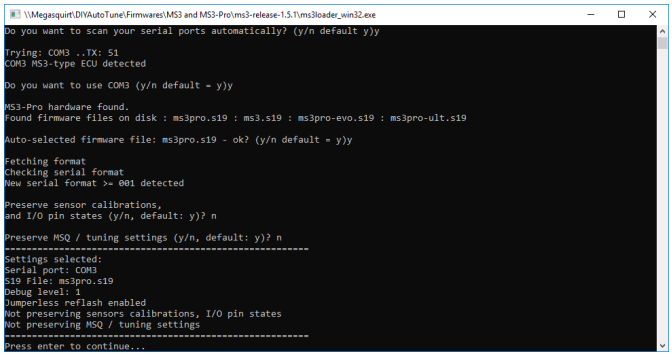
Scanning ports automatically is usually your easiest option, but it is slightly faster if you specify what COM port to use.



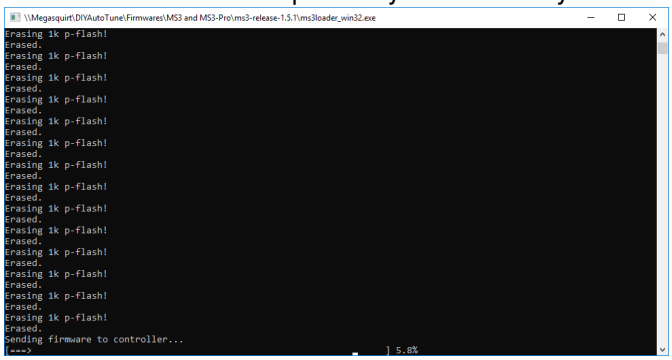
Normally, the loader will detect which MS3Pro version you are using. The above screen shot is for a first generation unit. If the loader selects the correct firmware, confirm this by pressing Y. Enter “N” if you need to change the firmware used.



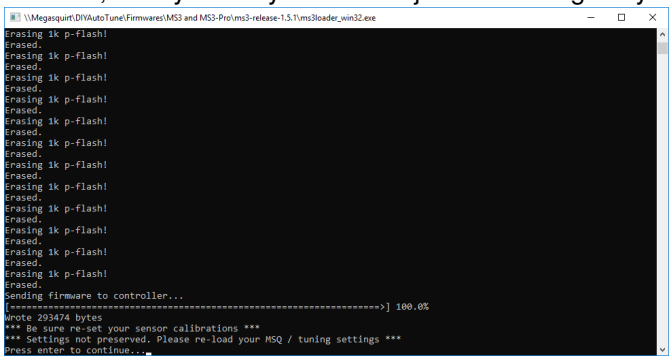
. If you are reloading the firmware due to communications issues, you should not attempt to preserve the current tune or port states. Examples of port states would be holding spark outputs in an off state. Note that you can still cause damage to coils if they are plugged in if an error occurs during the process; we recommend disconnecting the ignition coils or their power fuse(s) if you are loading firmware with the ECU in the car.



The firmware loader will present you a summary of the settings before loading firmware.



Do not interrupt the program or switch to another window while loading. If this takes long than approximately 45 seconds, then you likely need to adjust the settings of your USB to serial adapter to reduce the latency setting.



The loader will display a confirmation message once firmware is loaded successfully.

8.6.3 Linux firmware loader

It is assumed you'll be using the command line here.

1. Create a directory for MS3. e.g. mkdir ms3
2. A 32bit compiled command-line firmware loader for Linux is included in firmware zip.
3. Fetch the latest s19 and ini files in a zip from the Code file area and save into the ms3 directory. Extract the zip.
4. cd to the correct directory and key in the following: ./ms3loader_linux32bit

The program runs the same as the Windows version, so see above for an example of typical output. Typically, you only need to check you serial comm port. You can usually just press Enter on the other selection. When loading code for the first time be sure to answer "Y" to the First time loading question. If this takes long than ~45 seconds then you likely need to adjust the port settings of your USB-serial adapter to reduce the latency setting. For the FTDI built in device, try the following:

echo 1 > /sys/bus/usb-serial/devices/ttyUSB0/latency

If you get a permissions problem trying to run the file, you likely need to make it executable (e.g. chmod a+x ms3loader_linux32bit or through your graphical interface.)