
# EFILive FlashScan Serial Protocol

Paul Blackmore

# EFILive FlashScan Serial Protocol



First published
16 May 2009

Revised
7 February 2022

# Introduction

EFILive's FlashScan device may be configured to output OBDII PID data via its onboard RS232 serial port. External systems, such as dynamometers, can gain access to that PID data.

The RS232 serial port may also be used to transmit PID data to FlashScan for inclusion in the EFILive data logs.

This document describes how to configure FlashScan to output the serial PID data and the format of that data and how to transmit data to FlashScan and the format of the data to be transmitted.

# Prerequisites

For FlashScan V2 devices:

- You must be using software version: EFILive V8.1.2 Build 81 or later.
- You must be using firmware version: V2.06.21 or later.

For FlashScan V3 devices:

- You must be using software version: EFILive V8.3.17 or later.
- You must be using firmware version: V3.00.87 or later.

# Configuring FlashScan

Power up FlashScan, and select the following option:

For FlashScan V2:

1. F4: Options -> F1 Setup -> F1 Edit Settings.
2. Scroll down to the ***COM Port*** setting using the down arrow key.
3. Change the value to ***Dyno*** using the left or right arrow key.
4. Press Ok to save the changes.

   Do not remove power from FlashScan while the blue LED is illuminated. The blue LED indicates that FlashScan is busy writing data to flash memory.

For FlashScan V3:

1. Options -> F1 Setup -> F1 Configuration.
2. Scroll down to the ***COM Port Type*** setting using the down arrow key.
3. Change the value to ***External*** using the left or right arrow key.
4. Press "Tick" to save the changes.

# The Serial Protocol

### COM Port Settings

5. Baudrate 57600.
6. Data Bits: 8.
7. Parity: None.
8. Stop Bits: 1.

### Data Frames

There are three types of data frames that FlashScan can transmit:

- F Frames, which list the names and units of the PIDs that return floating point values.
- S Frames, which list the names and units of the PIDs that return string values.
- V Frames, which contains the data values of all PIDs', listed in the F, B and S frames.

  Previously a fourth frame type (B which was used for binary data) was supported. The binary frame type is now implemented via the S Frame.

There is one type of data frame that FlashScan can receive:

- D Frames, which are further subdivided by message IDs, each with different formats. Those formats are described later in this document.

### Data Format

Each frame transmitted/received by FlashScan follows this format:

**X : ~ |** <length><frame data><checksum>

Where **X** is one of:

- ***F***, ***S*** or ***V*** for frames transmitted by FlashScan and;
- ***D*** for frames received by FlashScan.

**: ~ |** is a unique, static value that:

a) Pads the frame's data to a long word boundary and;
b) Provides a searchable ASCII marker for the start of a valid frame.

<length> is a 16 bit word indicating how many bytes (including the checksum) make up the frame.

<frame data> is an array of data bytes with format determined by the frame type.

<checksum> is a modulo 0x100 sum of all bytes in the frame (excluding the checksum byte).

### Transmitting Data to FlashScan

Data that is transmitted to FlashScan must conform to one of EFILive's pre-configured D Frames. Currently only three D Frames are available for transmitting PID data to FlashScan.

All numeric data in D frames are stored in Intel (least significant byte first) byte format.

**Message ID 0x7F01** is used to request PID names and units from FlashScan. See *Data Content Request* in the next section.

**Message ID 0x0100** is predefined to contain the following PIDs.

```
00000000 443A 7E7C 3F00 0001 0000 7A43 0000 1643 D:~|?.....zC...C
00000010 52B8 5E3F 0000 C841 3400 CA42 0000 8C42 R.^?...A4..B...B
00000020 24B9 FC3D 0000 0C44 0000 7041 0000 7041 $..=...D..pA..pA
00000030 0000 A040 0000 7A43 0000 7A43 0000 803F ...@..zC..zC...?
00000040 3333 4B41 5F                          33KA_
```

| Field | Description | Type | Hex Data | Factor | Offset | Data | Units |
|---|---|---|---|---|---|---|---|
| Frame Marker | | | 443A7E7C | | | D:~\| | |
| Length | Number of data bytes | Word | 3F00 | | | 63 | Bytes |
| ID | Message ID | Word | 0001 | | | 0x0100 | |
| DYNTRQ | Torque | Float | 00007A43 | 1 | 0 | 250.0 | Nm |
| DYNPWR | Power | Float | 00001643 | 1 | 0 | 150.0 | kW |
| DYNLAMDA | Lambda | Float | 52B85E3F | 1 | 0 | 0.870 | Lambda |
| DYNAAT | Ambient Air Temp | Float | 0000C841 | 1 | 0 | 25.0 | °C |
| DYNBARO | Barometric Pressure | Float | 3400CA42 | 1 | 0 | 101.0 | kPa |
| DYNRELHUM | Relative Humidity | Float | 00008C42 | 1 | 0 | 70.0 | % |
| DYNATMCOR | Atmospheric Correction | Float | 24B9FC3D | 1 | 0 | 0.1234 | |
| DYNEGT1 | EGT1 | Float | 00000C44 | 1 | 0 | 560.0 | °C |
| DYNO2 | O2 | Float | 00007041 | 1 | 0 | 15.0 | % |
| DYNCO | CO | Float | 00007041 | 1 | 0 | 15.0 | % |
| DYNCO2 | CO2 | Float | 0000A040 | 1 | 0 | 5.0 | % |
| DYNHC | HC | Float | 00007A43 | 1 | 0 | 250 | ppm |
| DYNNOX | NOx | Float | 00007A43 | 1 | 0 | 250 | ppm |
| DYNCOCOR | COcor | Float | 0000803F | 1 | 0 | 1.00 | |
| DYNAFR | Brettsneider (Gas calc'd) | Float | 33334B41 | 1 | 0 | 12.70 | AFR |
| Check sum | | Byte | 5F | | | 0x5F | |

**Message ID 0x0200** is predefined to contain the following PIDs.

```
00000000 443A 7E7C 2D00 0002 EA14 B900 2C03 E902 D:~|-.......(...
00000010 6500 1803 4A03 2303 4E03 3F03 2703 3F03 e...J.#.N.?.'.?.
00000020 5203 4303 4303 0800 1E00 1D00 3700 2C00 R.C.C.......7.,.
00000030 4A00 3B                                 J.5
```

| Field | Description | Type | Hex Data | Factor | Offset | Data | Units |
|---|---|---|---|---|---|---|---|
| Frame Marker | | | 443A7E7C | | | D:~\| | |
| Length | Number of bytes | Word | 2D00 | | | 45 | Bytes |
| ID | Message ID | Word | 0002 | | | 0x0200 | |
| FPERPM | RPM | Word | EA14 | 1 | 0 | 5354 | rpm |
| FPEGPSSPD | Ground Speed | Word | B900 | 1 | 0 | 185 | Km/h |
| FPEACCL | Acceleration | Word | 2C03 | 0.0078125 | -4 | 2.345 | g-force |
| FPETURBO1 | Turbo Speed 1 | Word | E902 | 5.0 | 0 | 3728 | rpm |
| FPETEMP1 | Temperature 1 | Word | 6500 | 1 | 0 | 101 | °C |
| FPETEMP2 | Temperature 2 | Word | 1803 | 1 | 0 | 792 | °C |
| FPETEMP3 | Temperature 3 | Word | 4A03 | 1 | 0 | 842 | °C |
| FPETEMP4 | Temperature 4 | Word | 2303 | 1 | 0 | 803 | °C |
| FPETEMP5 | Temperature 5 | Word | 4E03 | 1 | 0 | 846 | °C |
| FPETEMP6 | Temperature 6 | Word | 3F03 | 1 | 0 | 831 | °C |
| FPETEMP7 | Temperature 7 | Word | 2703 | 1 | 0 | 807 | °C |
| FPETEMP8 | Temperature 8 | Word | 3F03 | 1 | 0 | 831 | °C |
| FPETEMP9 | Temperature 9 | Word | 5203 | 1 | 0 | 850 | °C |
| FPETEMP10 | Temperature 10 | Word | 4303 | 1 | 0 | 835 | °C |
| FPETEMP11 | Temperature 11 | Word | 4303 | 1 | 0 | 835 | °C |
| FPEPRSR1 | Pressure 1 | Word | 0800 | 6.894757 | 0 | 56 | kPa |
| FPEPRSR2 | Pressure 2 | Word | 1E00 | 6.894757 | 0 | 213 | kPa |
| FPEPRSR3 | Pressure 3 | Word | 1D00 | 6.894757 | 0 | 201 | kPa |
| FPEPRSR4 | Pressure 4 | Word | 3700 | 6.894757 | 0 | 382 | kPa |
| FPEPRSR5 | Pressure 5 | Word | 2C00 | 6.894757 | 0 | 307 | kPa |
| FPEPRSR6 | Pressure 6 | Word | 4A00 | 6.894757 | 0 | 514 | kPa |
| Check sum | | Byte | 3B | | | 0x3B | |

**Message ID 0x0201** is predefined to contain the following PIDs.

```
00000000 443A 7E7C 2300 0102 0400 4148 0000 AD0A D:~|#.....AH....
00000010 3E03 6500 1803 4A03 2303 4E03 8400 8C00 >.e...J.#.N.....
00000020 8800 8500 8E00 8800 9A                  .........
```

| Field | Description | Type | Hex Data | Factor | Offset | Data | Units |
|---|---|---|---|---|---|---|---|
| Frame Marker | | | 443A7E7C | | | D:~\| | |
| Length | Number of bytes | Word | 2300 | | | 35 | Bytes |
| ID | Message ID | Word | 0102 | | | 0x0201 | |
| FPESATS | Satellites in View | Word | 0400 | 1 | 0 | 4 | count |
| FPEELEVATION | Elevation | DWord | 41480000 | 0.01 | 0 | 185 | m |
| FPERPM2 | RPM 2 | Word | AD0A | 1 | 0 | 2733 | Rpm |
| FPETURBO2 | Turbo Speed 2 | Word | 3E03 | 5.0 | 0 | 4152 | Rpm |
| FPETEMP12 | Temperature 12 | Word | 6500 | 1 | 0 | 101 | °C |
| FPETEMP13 | Temperature 13 | Word | 1803 | 1 | 0 | 792 | °C |
| FPETEMP14 | Temperature 14 | Word | 4A03 | 1 | 0 | 842 | °C |
| FPETEMP15 | Temperature 15 | Word | 2303 | 1 | 0 | 803 | °C |
| FPETEMP16 | Temperature 16 | Word | 4E03 | 1 | 0 | 846 | °C |
| FPEPRSR7 | Pressure 7 | Word | 8400 | 6.894757 | 0 | 911 | kPa |
| FPEPRSR8 | Pressure 8 | Word | 8C00 | 6.894757 | 0 | 967 | kPa |
| FPEPRSR9 | Pressure 9 | Word | 8800 | 6.894757 | 0 | 941 | kPa |
| FPEPRSR10 | Pressure 10 | Word | 8500 | 6.894757 | 0 | 920 | kPa |
| FPEPRSR11 | Pressure 11 | Word | 8E00 | 6.894757 | 0 | 985 | kPa |
| FPEPRSR12 | Pressure 12 | Word | 8800 | 6.894757 | 0 | 944 | kPa |
| Check sum | | Byte | 9A | | | 0x9A | |

**Message ID 0x0202** is predefined to contain the following PIDs.

```
00000000 443A 7E7C 1D00 0202 9500 5401 2C01 3E00 D:~|......T.,.>.
00000010 0100 1A0B D717 2E00 0B00 4300 2900 7901 ..........C.).y.
00000020 9301 B5                                 ...
```

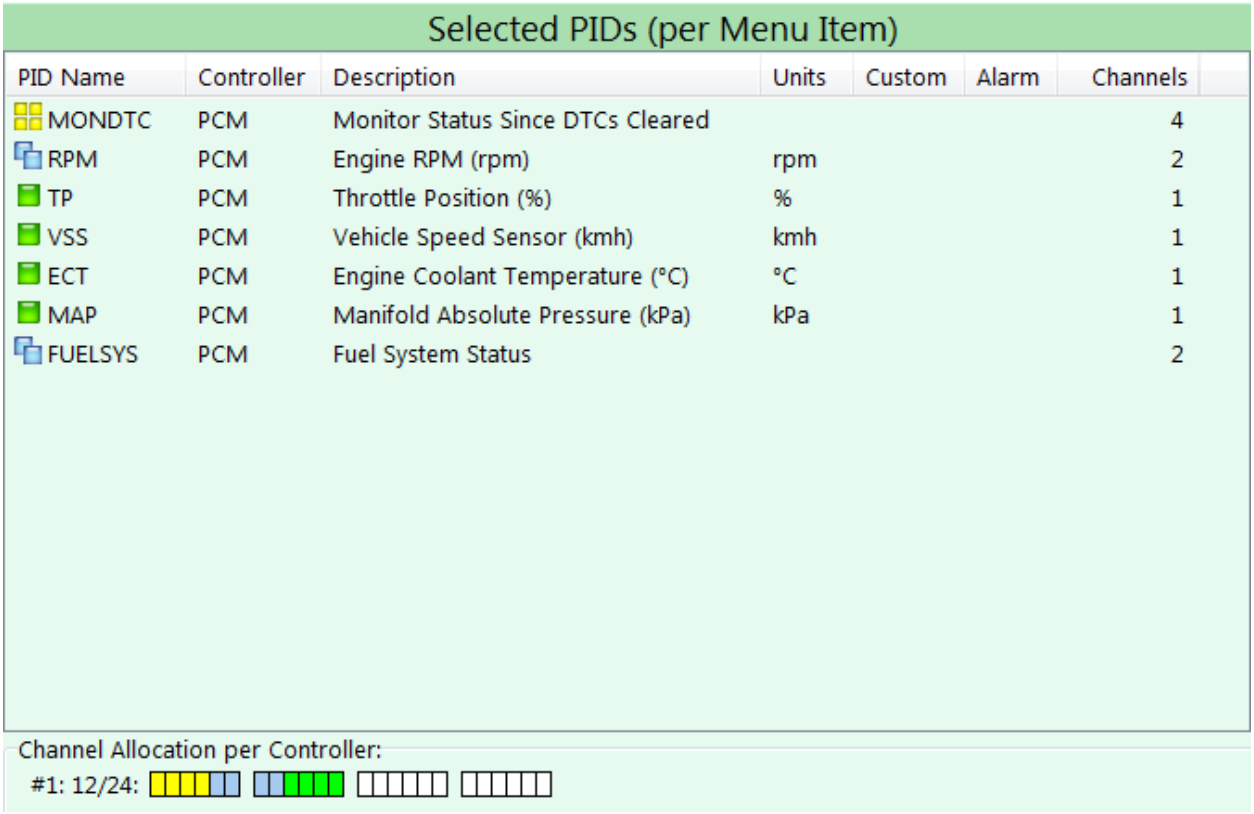
| Field | Description | Type | Hex Data | Factor | Offset | Data | Units |
|---|---|---|---|---|---|---|---|
| Frame Marker | | | 443A7E7C | | | D:~\| | |
| Length | Number of bytes | Word | 1D00 | | | 29 | Bytes |
| ID | Message ID | Word | 0202 | | | 0x0202 | |
| FPEANLGIN1 | Analog Input 1 Voltage | Word | 9500 | 2.44 | 0 | 365 | mV |
| FPEANLGIN2 | Analog Input 2 Voltage | Word | 5401 | 2.44 | 0 | 832 | mV |
| FPEANLGIN3 | Analog Input 3 Voltage | Word | 2C01 | 2.44 | 0 | 733 | mV |
| FPEANLGIN4 | Analog Input 4 Voltage | Word | 3E00 | 2.44 | 0 | 153 | mV |
| FPEDGTLIN1 | Digital Input 1 State | Byte | 01 | 0=Off 1=On | 0 | On | |
| FPEDGTLIN2 | Digital Input 2 State | Byte | 00 | 0=Off 1=On | 0 | Off | |
| FPEFREQIN1 | Frequency Input 1 | Word | 1A0B | 1 | 0 | 2842 | Hz |
| FPEFREQIN2 | Frequency Input 2 | Word | D717 | 1 | 0 | 6103 | Hz |
| FPEOUTPDC1 | Output Duty Cycle 1 | Word | 2E00 | 1 | 0 | 46 | % |
| FPEOUTPDC2 | Output Duty Cycle 2 | Word | 0B00 | 1 | 0 | 11 | % |
| FPEOUTPDC3 | Output Duty Cycle 3 | Word | 4300 | 1 | 0 | 67 | % |
| FPEOUTPDC4 | Output Duty Cycle 4 | Word | 2900 | 1 | 0 | 41 | % |
| FPEANLGOUT1 | Analog Output 1 Voltage | Word | 7901 | 2.44 | 0 | 920 | mV |
| FPEANLGOUT2 | Analog Output 2 Voltage | Word | 9301 | 2.44 | 0 | 985 | mV |
| Check sum | | Byte | 9A | | | 0x9A | |

## Unit Conversions

| Convert from | Convert To | Function |
|---|---|---|
| **Temperature** | | |
| °F | °C | C = (F-32)*5/9 |
| °C | °F | F = C*9/5+32 |
| **Pressure** | | |
| psi | kPa | kPa = psi*6.894757 |
| kPa | Psi | psi = kPa/6.894757 |
| **Speed** | | |
| mph | kmh | kmh=mph*1.609344 |
| kmh | mph | mph=kmh/1.609344 |
| **Torque** | | |
| ftlb | Nm | Nm=ftlb*1.355818 |
| Nm | ftlb | ftlb=Nm/1.355818 |
| **Power** | | |
| hp | kW | kW=hp*1.340483 |
| kW | hp | hp=kW/1.340483 |

### Receiving Data from FlashScan

For the remainder of this document, the examples shown are based on this LS1B PID selection.




The image above is taken from the EFILive_ScanAndTune.exe program in the Black Box Logging setup window.

More detailed information about each PID in the list above (including any sub-PIDs) can be obtained by right-clicking on the PID and selecting “More info...” from the “F5: Black Box Logging” option in the EFILive V8 Scan and Tune software.

### Data Content Request

FlashScan will always transmit the complete set of PID values selected by the user. Because it is not possible for the receiving application to know the layout of the data frame in advance, FlashScan provides a method to query the names, formats and units of each of the PIDs that are being transmitted.

To query the PIDs' names, formats and units, you must transmit a D frame with the message ID 0x7F01 and one data byte (which is the checksum byte). The D frame may be transmitted at any time. The reply (a pair of F and S frames) will be transmitted after the current V frame or current pair of F and S frames has completed.

D Frame for requesting PID names:

| | | |
|---|---|---|
| 00000000 | 443A 7E7C 0300 017F F9 | D:~\|..... |

| Field | Description | Hex Data | Formatted Data | Units |
|---|---|---|---|---|
| Frame Marker | | 0x443A7E7C | D:~\| | |
| Length | Number of data bytes | 0x0300 | 3 | Bytes |
| ID | Message ID | 0x017F | 0x7F01 | |
| Check sum | | 0xFB | 0xFB | |

### Data Content Frames

Frames F and S are returned in that order, in response to the Data Content request as described above.

Each Data Content frame is terminated with a null character 0x00 followed by the check sum byte.

The F frame:

```
00000000 463A 7E7C 3200 4454 4343 4E54 3D43 6F75 F:~|2.DTCCNT=Cou
00000010 6E74 2C52 504D 3D72 706D 2C54 503D 252C nt,RPM=rpm,TP=%,
00000020 5653 533D 6B6D 682C 4543 543D B043 2C4D VSS=kmh,ECT=.C,M
00000030 4150 3D6B 5061 00CB                    AP=kPa..
```

The S frame:

```
00000000 533A 7E7C CF00 4D49 4C2C 4343 4D5F 5244 S:~|..MIL,CCM_RD
00000010 592C 4655 454C 5F52 4459 2C4D 4953 5F52 Y,FUEL_RDY,MIS_R
00000020 4459 2C43 434D 5F53 5550 2C46 5545 4C5F DY,CCM_SUP,FUEL_
00000030 5355 502C 4D49 535F 5355 502C 4547 525F SUP,MIS_SUP,EGR_
00000040 5355 502C 4854 525F 5355 502C 4F32 535F SUP,HTR_SUP,O2S_
00000050 5355 502C 4143 5246 5F53 5550 2C41 4952 SUP,ACRF_SUP,AIR
00000060 5F53 5550 2C45 5641 505F 5355 502C 4843 _SUP,EVAP_SUP,HC
00000070 4154 5F53 5550 2C43 4154 5F53 5550 2C45 AT_SUP,CAT_SUP,E
00000080 4752 5F52 4459 2C48 5452 5F52 4459 2C4F GR_RDY,HTR_RDY,O
00000090 3253 5F52 4459 2C41 4352 465F 5244 592C 2S_RDY,ACRF_RDY,
000000A0 4149 525F 5244 592C 4556 4150 5F52 4459 AIR_RDY,EVAP_RDY
000000B0 2C48 4341 545F 5244 592C 4341 545F 5244 ,HCAT_RDY,CAT_RD
000000C0 592C 4655 454C 5359 5331 2C46 5545 4C53 Y,FUELSYS1,FUELS
000000D0 5953 3200 70                           YS2.p
```

### The Data Value Frame

The V frame contains a non-delimited list of all the floating-point PIDs' values and a comma separated list of all the string PIDs' values determined by the F and S frames.

The length of the frame is determined by the number and format of the PIDs listed in the F and S frames.

For FlashScan V2: V Frames are transmitted 10 times per second.

For FlashScan V3: V Frames are transmitted at the same rate that frames are logged on the V3 device. Many CAN based vehicles log data at 40 frames per second (or sometimes faster) which can saturate the RS232 serial link when a large number of PIDs are being logged. In that case frames will be skipped.

At 57600 baud, the RS232 serial link can transmit about 5,000 bytes per second, which is about 125 bytes at 40 frames per second. So, a good rule of thumb to prevent frame skipping is to reduce the PID selection so that the total length of the V frame is 125 bytes or less.

The format of the V frame is:

- A 4 byte unique frame marker: **V:~|**
- A 2 byte count in Intel (least significant byte first) byte format of the number of bytes left in the frame, including the checksum byte.
- A 4 byte timestamp in Intel (least significant byte first) byte format. The timestamp represents the number of milliseconds since the log was started.
- A 4 byte (32 bit) flag value.
    - For FlashScan V2 devices, only the first 8 bits are defined and those 8 bits correspond to the F1..F4 and Ctrl+F1..Ctrl+F4 keys that may have been pressed by the user to mark some significant event during data logging.
    - For FlashScan V3 devices, only the first 4 bits are defined and those 4 bits correspond to the Left, Right, Fn+Left and Fn+Right keys that may have been pressed by the user to mark some significant event during data logging.
- IEEE floating point values in Intel (least significant byte first) byte format. There will be as many 4 byte floating point values as are defined in the F frame.
- Comma separated strings. There will be as many comma separated strings as are defined in the S frame. A string value will never exceed 12 bytes in length. The last string is followed by a null terminator.
- A checksum byte which contains a "modulo 256" checksum of all the previous bytes in the V frame.

Example V Frame:

```
00000000 563A 7E7C 7C00 2A15 0000 0000 0000 0000 V:~||.*.........
00000010 0000 0000 0000 0000 0000 0000 0000 0000 ................
00000020 1CC2 0000 2041 4F66 662C 5965 732C 5965 .... AOff,Yes,Ye
00000030 732C 5965 732C 5965 732C 5965 732C 5965 s,Yes,Yes,Yes,Ye
00000040 732C 5965 732C 5965 732C 5965 732C 4E6F s,Yes,Yes,Yes,No
00000050 2C4E 6F2C 5965 732C 4E6F 2C59 6573 2C4E ,No,Yes,No,Yes,N
00000060 6F2C 4E6F 2C4E 6F2C 5965 732C 5965 732C o,No,No,Yes,Yes,
00000070 4E6F 2C59 6573 2C59 6573 2C4F 4C2C 4F4C No,Yes,Yes,OL,OL
00000080 00FF                                     .f
```

| Field | Hex Data | Formatted Data | Units |
|---|---|---|---|
| Frame Marker | 0x563A7E7C | V:~\| | |
| Length | 0x7C00 | \| | |
| Timestamp | 0x2A150000 | 00:00:05.418 | |
| Flags | 0x00000000 | | |
| DTCCNT | 0x00000000 | 0 | Count |
| RPM | 0x00000000 | 0 | Rpm |
| TP | 0x00000000 | 0 | % |
| VSS | 0x00000000 | 0 | kmh |
| ECT | 0x00001CC2 | -39 | °C |
| MAP | 0x00002041 | 10 | kPa |
| MIL | 0x4F6666 | Off | |
| CCM_RDY | 0x596573 | Yes | |
| FUEL_RDY | 0x596573 | Yes | |
| MIS_RDY | 0x596573 | Yes | |
| CCM_SUP | 0x596573 | Yes | |
| FUEL_SUP | 0x596573 | Yes | |
| MIS_SUP | 0x596573 | Yes | |
| EGR_SUP | 0x596573 | Yes | |
| HTR_SUP | 0x596573 | Yes | |
| O2S_SUP | 0x596573 | Yes | |
| ACRF_SUP | 0x4E6F | No | |
| AIR_SUP | 0x4E6F | No | |
| EVAP_SUP | 0x596573 | Yes | |
| HCAT_SUP | 0x4E6F | No | |
| CAT_SUP | 0x596573 | Yes | |
| EGR_RDY | 0x4E6F | No | |
| HTR_RDY | 0x4E6F | No | |
| O2S_RDY | 0x4E6F | No | |
| ACRF_RDY | 0x596573 | Yes | |
| AIR_RDY | 0x596573 | Yes | |
| EVAP_RDY | 0x4E6F | No | |
| HCAT_RDY | 0x596573 | Yes | |
| CAT_RDY | 0x596573 | Yes | |
| FUELSYS1 | 0x4F4C | OL | |

| | | | |
|---|---|---|---|
| FUELSYS2 | 0x4F4C | OL | |
| Check sum | 0xFF | 0xFF | |

**Future developments**

- Allow data logging to be started/stopped using commands sent on the serial link.
- Allow PID selection to be changed using commands sent on the serial link.
- If/when PID selection is allowed to be changed “on the fly”; embedded F and S frames would indicate the new PID selection without being requested by the receiving application.
- Additional frames for storing Meta data such as customer and vehicle details may be added in future. Those frames will start with the standard **X:~!** sequence where “X” is a letter other than the existing F, S or V frame identifiers. For forward compatibility, if you encounter a frame with an identifier other than F, S or V, you may simply ignore it.