UltraGauge MX V1.4

The OBDII standard describes certain parameters which can be accessed via the OBDII. Many of the parameters are left to the manufacturer's discretion to support or not. This is why the gauges available through UltraGauge are vehicle dependent. All Manufacturers have the ability to access additional vehicle parameters beyond those specified in the OBDII. For example, there is no OBDII parameter for Transmission Temperature; however, some manufacturers can access this parameter.
Fundamentally there are two types of parameters:

1. **Standardized OBDII parameters**; roughly 135. Many are not very useful and of the 135, manufacturers typically provide 1/3 or less. Ultragauge inherently offers around 60 of these parameters. Standardized OBDII parameters are typically related to emissions.
2. **Manufacturer specific parameters**: These are parameters that the manufacturer has inserted for their own purposes. Many are duplicates of the Standardized gauges, while others are wholly separate and not included in the standardized OBDII parameters. These parameters are not focused on emissions and can be any useful parameter the manufacturer has seen fit to insert.

Many manufactures access these parameters through the OBDII. Others access the parameters through proprietary, non-standard pins inserted into the OBDII connector. Only manufacturers which use the same interface as OBDII have parameters that can be accessed by the UltraGauge MX. Typically Ford, GM, and vehicles with CAN interfaces can be accessed. All vehicles sold in the USA since 2008 are required to support CAN, and many supported CAN prior to 2008. This is not necessarily true for vehicles sold outside the USA market.

Manufacturer parameters typically cannot be accessed on Vehicles with 9141 and KWP2000 protocols/interfaces. In general foreign vehicles prior to 2008 will use the 9141 or KWP2000 protocol. Our vehicle gauge estimator can also be used to determine the protocol used by most vehicles. The UltraGauge MX can be used to access all of the ~135 standardized OBDII parameters, if supported, even on vehicles with 9141 or KWP2000 protocols. It's important to understand that while the OBDII defines ~135 parameters, manufacturers only support a fraction of this; the older the vehicle, typically the fewer parameters supported.

The ability to access manufacturer specific parameters is very similar to the X-gauge feature found in ScanGauge™. Many codes which function in ScanGauge™ will function in UltraGauge with minor modifications. However, if the code has been obfuscated, so that the actual manufacturer code is concealed, then it is not a valid code and will not directly work on UltraGauge and effort will be needed to convert the code to its true form.

In order to access parameters, beyond the provided pre-programmed parameters, it is necessary to configure Ultragauge with codes and it is for this reason the Ultragauge MX is only recommended for users which are very comfortable with technology and have the time to devote to tracking down codes for their vehicle. Information regarding parameters and their access and interpretation is not made public by manufacturers. This information over time leaks out or is found via reverse engineering.

To simplify use, the UltraGauge MX is offered pre-programmed with ~8 common manufacturer parameters for various vehicles. There is no guarantee that all the pre-programmed parameters will work with your year, make and model vehicle. Those that are not supported by your vehicle will show "Err". It is best to search vehicle specific forums to understand if such parameters are accessible in your specific vehicle. Ultragauge.com does not have any additional gauges/codes beyond those provided as part of the pre-programming options, other than those contained below in the tables.

The UltraGauge MX is simply a tool that provides the potential to access these parameters and provides no assurance that such parameters are actually available or accessible. The pre-programmed parameters can be fully reprogrammed by the user.

The UG MX supports 8 manufacturer programmable parameters. The following tables represent the various pre-programming options. However, not all parameters may work for your particular year and model. If your vehicle does not support the code for the parameter, "Err" will be displayed. Also, the preprogrammed codes may not be correct but are accurate to the best of our knowledge.

Early FORD (non-CAN) PreProgrammed Parameters	Description
Kilometers Per Hour	Much more accurate version of the OBDII standardized version (see Special Bit below)
Fuel Level %	Fuel Tank Level percent of full. Allows Auto Fill function for vehicles which previously did not support it., See "Special Bit" below
Transmission Temperature °F	Temperature of the transmission fluid °F
Barometric Pressure	Atmospheric Pressure in Inches of Mercury
Converter Torque	ft.lbs
Fuel Rail Pressure	PSI
Transmission output shaft RPM	RPM
Engine Coolant Temperature °F	Same as standardized OBDII coolant temperature

Ford 7.3L Diesel* PreProgrammed Parameters	Description
Engine Oil Temperature °F	Engine Oil Temperature in °F
Fuel Injector PSI	The pressure of the fuel at the injectors in PSI
IAT °F	Intake Air Temperature °F
Exhaust Pressure	Exhaust pressure in PSI
Boost Pressure	Turbo Boost Pressure in PSI
Glow Plug %	Glow Plug %
Barometric Pressure	Inches of Mercury
Engine Coolant Temperature F	Same as standardized OBDII coolant temperature
* See discussion on Ford Diesels here: http://www.ultra-gauge.com/customer_support/knowledgebase.php?article=28	

If any of these 7.3L gauges show "Err" please contact support

Ford 6.4L Diesel* PreProgrammed Parameters	Description
Barometric Pressure	Inches of Mercury
Fuel Level %	Fuel Tank Level percent of full. Allows Auto Fill function for vehicles which previously did not support it. See "Special Bit" below
Transmission Temperature °F	Temperature of the transmission fluid °F
Engine Oil Temperature °F	Much more accurate version of the OBDII standardized version
Exhaust Pressure	Exhaust pressure in PSI
Exhaust Temperature °F	Temperature of the Exhaust prior to Catalyst °F
Boost Pressure	Turbo Boost Pressure in PSI
RPM	Same as standardized OBDII RPM
* See discussion on Ford Diesels here: http://www.ultra-gauge.com/customer_support/knowledgebase.php?article=28	

Ford 6.0L Diesel* PreProgrammed Parameters	Description
Barometric Pressure	Inches of Mercury
Fuel Level %	Fuel Tank Level percent of full. Allows Auto Fill function for vehicles which previously did not support it. See "Special Bit" below
Transmission Temperature °F	Temperature of the transmission fluid °F
Engine Oil Temperature °F	Much more accurate version of the OBDII standardized version
Exhaust Pressure	Exhaust pressure in PSI
Boost Pressure	Turbo Boost Pressure in PSI
Misfire Count Bank 1	Total accumulative Misfire Count for Bank1
RPM	Same as standardized OBDII RPM
* See discussion on Ford Diesels here: http://www.ultra-gauge.com/customer_support/knowledgebase.php?article=28	

Ford (with CAN) PreProgrammed Parameters	Description
Barometric Pressure	Inches of Mercury
Fuel Level %	Fuel Tank Level percent of full. Allows Auto Fill function for vehicles which previously did not support it. See "Special Bit" below
Transmission Temperature 1 °F	Temperature of the transmission fluid °F
Transmission Temperature 2 °F	Temperature of the transmission fluid °F (Using alternate code)
Transmission Torque	Transmission Torque in Ft.lbs
Cylinder Head Temperature	Cylinder Head Temperature in °F
EGR Duty Cycle	EGR Duty Cycle %
RPM	Same as standardized OBDII RPM
This pre-programming is Ford CAN other than those specific engine sized vehicles above	

Early GM (non-CAN) PreProgrammed Parameters	Description
Run Time	Elapsed seconds since engine start
Knock Retard	degrees
Barometric Pressure	Inches of Mercury
Air Fuel ratio	
Oil Life remaining	Percentage of oil life remaining
Engine Torque	ft,lbs
Transmission Temperature	Temperature of the transmission fluid °F
Engine Coolant Temperature	Same as standardized OBDII coolant temperature

GM with CAN (~2008+) PreProgrammed Parameters	Description
Run Time	Elapsed seconds since engine start
Knock Retard	degrees
Barometric Pressure	Inches of Mercury
Air Fuel ratio	
Oil Life remaining	Percentage of oil life remaining
Engine Coolant Temperature	Same as standardized OBDII coolant temperature
Engine Torque	ft,lbs
Transmission Temperature	Temperature of the transmission fluid °F (Some GM vehicles use an alternate Code: TDATA: 07E222194001)

Toyota Tundra/Tacoma Not CAN (≤2004)	Description
Transmission Temperature	Temperature of the transmission fluid °F. The pr-programmed code will need to be changed to: TDATA: 686AF101B40000 TCTRL: 85 RCTRL: 00 RPOS: 2808 MTCH: 41B400 X: 0001 /: 0001 +: FFD8 AVE: 00 Output Format 00 for °C Output Format 10 for °F Left/Right: 31

Toyota Tundra/Tacoma with CAN (2005-2011)	Description
Kilometers Per Hour	Same as the Standardized KPH. Note that the pre-programming is incorrect and the width of this must be set to 8bits. RPos=1808, not 1810
Barometric Pressure	Inches of Mercury
Air Fuel ratio	
Transmission Temperature °F	Temperature of the transmission fluid °F. Math should be changed to X:0001 /:0100 +:FFD8 Output Format: 10
Transmission Temperature °C	Temperature of the transmission fluid °C . Math should be changed to X:0001 /:0100 +:FFD8
Injector Time	Injector on time in milliseconds
Lock up Converter Status	1=Locked up, 0=Not locked
Evaporative System PSI	Pressure of the Evaporative system in PSI

Toyota Tundra/Tacoma with CAN (2012-2015)	Codes
Transmission Torque Converter Temperature (The pre-programming provided is for 2005-2011. It will be necessary to replace the provided codes with these codes)	TDATA: 07E02182 Mtch: 6182 RPos: 2808 RCtrl:21 TCtrl:92 X: 0001 /: 0001 +: FFD8 Output format: 10 (for °F) Output format: 00 (for °C) Left/Right is 31
Transmission Pan Temperature (The pre-programming provided is for 2005-2011. It will be necessary to replace the provided codes with these 2012-2015 codes)	TDATA: 07E02182 Mtch: 6182 RPos: 1808 RCtrl:21 TCtrl:92 X: 0001 /: 0001 +: FFD8 Output format: 10 (for °F) Output format: 00 (for °C) Left/Right is 31

Toyota Tundra/Tacoma with CAN (2016+)	Codes
Transmission post Torque Converter Temperature (The pre-programming provided is for 2005-2011. It will be necessary to replace the provided codes with these 2016+ codes)	TDATA: 0701221628 MATCH: 621628 TCtrl: 93 RPos 2010 RCtrl: 31 X: 0001 /: 0100 +: FFD8 Output format: 10 (for °F) Output format: 00 (for °C) Left/Right is 31
Transmission Pan Temperature (The pre-programming provided is for 2005-2011. It will be necessary to replace the provided codes with these 2016+ codes)	TDATA: 0701221627 MATCH: 621627 TCtrl: 93 RPos 2010 RCtrl: 31 X: 0001 /: 0100 +: FFD8 Output format: 10 (for °F) Output format: 00 (for °C) Left/Right is 31

Toyota LandCruiser 200	Description
Transmission Torque Converter Temperature	Temperature of the transmission Torque Converter fluid °C
Transmission Pan Temperature	Temperature of the transmission Pan fluid °C
Alternator Duty Ratio	Alternatory % Duty Ratio
Gear Position	Current Gear Position
Torque Converter Lock Up Status	1=Locked up, 0=Not locked
Actual Fuel Temperature	Tank Fuel Temperature in °C
Return Fuel Temperature	Return Fuel Temperature in °C
Engine Coolant Temperature	Engine Coolant Temperature in °C

Honda with CAN	Description
Transmission Temperature	Temperature of the transmission fluid °C
Transmission Temperature	Temperature of the transmission fluid °F
Misfire Count	Total Engine Misfire Count
A/C PSI	Pressure of the Air-conditioning System in PSI
ECM Volts	Voltage at the ECM
Oil Life remaining	Percent
Air Fuel ratio	
Alternator Amps	Alternator output Amps

Prius Gen II	Description
Kilometers Per Hour	Much more accurate version of the OBDII standardized version (see Special Bit below)
Fuel Level %	Fuel Tank Level percent of full. Allows Auto Fill function for vehicles which previously did not support it. See “Special Bit” below
Battery Voltage 1	
Battery Voltage 2	
Battery Amps	
SOC %	Battery State of Charge %
Upper Battery Temp °F	
Lower Battery Temp °F	

Prius Gen III (2010-2015)	Description
Miles Per Hour	Much more accurate version of the OBDII standardized version (see Special Bit below)
SOC %	Battery State of Charge %
Battery Amps	
Friction Brake Sensor	0.00-4.50 (Shows actual friction braking amount)
Battery Air Intake Temp °F	
RPM	
A/C Watts	A/C Power usage in Watts
ICT °F	Intake Coolant temp (Unconfirmed)

Since the creation of the Gen III pre-programming option, there is much better information and more available gauges. Please do an internet search for Gen III codes.

Mgauge	2007-2013 Sprinter/Mercedes RV	Abbreviation	Description
1	**T**ransmission **F**luid **T**emperature	TFT °F	Transmission Fluid Temperature
2	Start Wait Indicator	Glw sts	Glow Plug Status. This single bit when "1" means "wait". When "0" means that it is okay to start the engine.
3	Torque Demanded %	TQ% Dmd	Displays the requested torque output of the engine by the driver. Potential sources: Operator via pedal, Cruise Control, Trans governor.
4	Torque Actual %	TQ% Act	Actual output torque of engine as a percent of max torque.
5	**M**ass **A**ir **F**low	MAF g/s	Mass of Air flowing into the engine as measured by the MAF sensor. Units are grams/sec and can be changed to lbs/min or lbs/hour via the output format field.
6	Engine Coolant Temperature	ECT °F	Engine Coolant Temperature as reported by Sensor 1. Set the output format field to "00" for Celsius.
7	Turbo Inlet Pressure	TI PSI	Turbo Inlet Pressure in PSI. Can be changed from PSI to kPa by setting the output format field to "00".
8	Intake **M**anifold **A**ir **P**ressure	MAP PSI	Intake Manifold Air Pressure in PSI.

Mazda with CAN	Description
Transmission Temperature	Three different codes are provided for Mazda CAN vehicles. Only one will work for your vehicle and the others can be ignored or replaced and should be unassigned from the Gauge Page

The units for the above can be changed by reprogramming. For example, Degrees F can be set to Degrees C, PSI to kPA, etc.
The UltraGauge MX can be used on vehicles other than those listed above, but there are no preprogrammed parameters available at this time.

Mgauge Programming

(This information is subject to change)

MENU → Gauge/Page Menu ... → Select Gauge/Page → M Gauge Setup
There are 8 Manufacturer gauges. Each is fully programmed via the screen below. When the cursor is positioned at "Mx", pressing the UP/Down Key will proceed to the next gauge.

Mgauge DISPLAY VIEW:

0	1	2	3	4	5	6	7	8	9	0	11	12	13	14	15	16	17	18	19	20	
M		G	a	u	g	e		N								E	X	I	T		Mgauge N
A	b	b	r	1	:	X	X	X			A	b	b	r	2	:	Y	Y	Y		Abbreviation fields
T	D	a	t	a	:		A	A	B	B	C	C	D	D	E	E	F	F	G	G	up to 7 bytes
T	C	t	r	l	:	A	A					R	C	t	r	l	:		A	A	
R	P	O	S		A	A	B	B		M	T	C	H		A	A	B	B	C	C	Targeting and Match fields
X	:	X	X	X	X		/	:	Y	Y	Y	Y			+	:	Z	Z	Z	Z	Math
O	u	t		F	o	r	m	a	t	:	X	X			A	v	e	:	X	X	Output Formatting and smoothing(averaging)
L	e	f	t	/	R	i	g	h	t	:	X	X									Left right digits

Depending on the vehicle, entries above may or may not be required. Entries not required are ignored

Abbr1/Abbr2: are simple text fields that identify the function of the gauge on the display page when in use.

TData: It is the unique command sent by UG MX to the vehicle that requests a particular parameter. When TCtrl is set to "listen only", the last byte of TData (GG above) controls how many page refreshes UltraGauge waits without receiving a response, before "Err" is displayed. By default this value is "2", but can be as large as 0xFF or 255. This is useful when UltraGauge is capturing infrequent data such as TPMS data.

TCtrl:

[7] - always 1
[6] std gauge – Identifies the entry to be a Standard OBDII gauge
[5] Request/response, or listen only, Listen=1 (Can only)
[4] functional=0, physical=1 (Can only)
[3] – Special Bit – Used for to flag to UG that a Mgauge should be substituted for a std gauge (see below)
[2:0] how many bytes to send; 0-7 bytes. The TData area is only 7 bytes deep. For CAN, this count does not include the address

TCtrl	CAN	FORD	GM	9141/KWP
7	1	1	1	1
6	N/A	STD	STD	STD
5	Listen Only	-	-	-
4	Functional/Physical	-	-	-
3	Special Bit	Special Bit	Special Bit	Reserved
2:0	TX Size	TX Size	TX Size	TX Size

Special Bit

Mgauge 1: If the special bit is set, then UG assumes Mgauge 1 contains parameters to access the KPH parameter. It replaces accesses to the standardized KPH parameter. This bit should not be set if Mgauge(1) is used for a parameter other than KPH. This replacement is provided because the standardize value only has a resolution of 8 bits . The Manufacture parameter typically has a resolution of 16 bits, potentially providing 256 times more accuracy.

Mgauge 2: If the special bit is set, then UG assumes Mgauge 2 contains parameters to access the Fuel Level percentage. It replaces accesses to the standardized parameter. This bit should not be set if Mgauge(2) is used for a parameter other than Fuel Level %. This replacement is provided because many early OBDII vehicles do not support the Fuel Level % via the OBDII and as a result auto fuel fill-up detection is not possible.

No other M gauges have special bits at this time

Mtch: A match value used to confirm that the response from the ECM is the correct response matching the request

RCtrl (For Mtch positioning) (Not used for Ford or GM)

[7] Reserved, set to 0
[6:4] Number of consecutive bytes to match
[3] reserved, set to 0
[2:0] byte position within the response

RPos (position of data)

AA[7:0] Bit position
BB[7] = signed result =1, 0 = unsigned (means the ECM supplied data is signed)
BB[6] = reserved
BB[5:0] number of bits . A single bit or multiples of 8 bits, i.e., 0x1, 0x8,0x10,0x18 etc.

AVE:

This parameter averages the value shown. A value of 0 results in no averaging. With increasing values the results are further averaged. This is useful for values which are relatively not constant. For example, the fuel level value can change rapidly as fuel sloshes around in the fuel tank. Increasing the "AVE" value will eliminate this rapid variation.

MATH:

The data read from the vehicles computer can be altered with basic Math.

Result = Value * XXXX / YYYY + ZZZZ
Both XXXX and ZZZZ are signed values…which means that if the upper most bit is 1, the value is interpreted as a negative number

Out Format:

After the math function is applied, the value can be further altered using the Out format field.
For example, if the Mgauge parameter is provided in kPa, and the desired gauge is PSI, the OUT field would be set to "17"
Note that many of the "Out" formats can be achieved with the MATH fields. This Format field is offered for cases when the MATH parameters have been used to shape the value and there is no additional flexibility to achieve the conversion.

Out Format	Function
0	No change
1	/10K
2	/1000
3	/100
4	/10
5	No Change
6	x 10
7	x 100
8	x 1000
9	x 10k
A	/60 convert seconds to minutes
B	/3600 convert seconds to hours
C	/(3600*24) convert seconds to days
D	Single digit 0 or 1 (Use when RPos=xx01)
E	/3600 and display as Time Format
F	Display as raw 2 or 4 digit Hex value
10	Celsius to Fahrenheit
11	kPa to PSI
12	kPa to inHg
13	Nm to ft.lbs
14	Liters to US Gallons
15	grams/sec to lbs/min
16	grams/sec to lbs/hour

LEFT / RIGHT: Sets the number of digits to be displayed to the Left and Right of the decimal place.

GM Vehicles

If the Mgauges all show "Err", then try the following change:

Non CAN GM: Change TCtrl from 87 to 86

If a particular pre-programmed Mgauge continues to show "Err", the cause may likely be that you vehicle does not support this Mgauge.

CAN Vehicles

To Translate CAN ScanGauge™ codes:

TXD: Use as is as TData
RXF -> Mtch - Ignore bytes 1,3 & 5. For example 046205090691 -->620991
RXD -> RPos = RXD – 0x1000. For example hex value 0x3010-0x1000 becomes 0x2010
TCtrl = 93 (typically) For CAN, the 2nd nibble indicates the number of bytes in TData after the address bytes
RCtrl = 31 (typically) ("3" means compare 3 bytes. And the "1" means compare starting at position 1.
Math -> First 2 bytes = x:, Next 2 bytes = /:, Next 2 = +:
Output format: 0

UG has additional flexibility. In general leave the other fields as is unless you have a need to change them

CAN CODE DEBUGGING

If you have a code that you believe should work for your vehicle, but receive "Err" instead, then either the code is not supported by your vehicle or the code has been entered incorrectly or needs tweaking to work. There are three primary functions of the codes:

1. **TData & TCtrl** represent the parameter request to the vehicle. UltraGauge sends these bytes to the vehicle's ECU. The ECU receives the bytes and if the code is valid and valid for the vehicle, the ECU will generate a response. If the code is invalid, then it will either be ignored or a Negative Response will be generated. A Negative Response is simply a response were the ECU specifically communicates that it does not support the TData code or that the code was incorrectly entered. A Negative response is characterized by a value of "7F" appearing as the 2nd byte of the response. RCtrl and RPos can be used to examine a response, if any, to determine if it is a negative response.
2. **Mtch & RCtrl** are used to capture the correct response. The CAN bus is busy with traffic as the ECU and other vehicle modules communicate. The Mtch and RCtrl fields are used by UltraGauge to examine each information packet on the CAN bus until a matching packet is received or the requests times out. If no matching response is received, UltraGauge will display "Err". Mtch contains the bytes which will be compared against the packets. RCtrl controls which byte position within the packet the comparison begins and how many bytes are compared.
   When RCtrl is 00, UltraGauge performs no examination and captures the first packet with the correct response Address. RCtrl =00 can be used along with RPos to capture and examine packets to determine if the response is a Negative response or a response containing data. For Example, to capture the raw unfiltered data the following settings would be used
   RCtrl = 00 (compare no bytes)
   Format = 0F (Displays raw hexadecimal data)
   RPos = 0010, 1010, 2010, 3010. (For each of these the data is displayed. Together they form the response)

   For example, if the results are:
   RPos = 0010, Data = 0462
   RPos = 1010, Data = 0991
   RPos = 2010, Data = 3200
   RPos = 3010, Data = 0000
   Then the complete information packet response is 0462099132

   Let's say, the data which represents the parameter (gauge) of interest is a single byte and is the "32" and the 04620991 is a preceding constant. Then to correctly capture this specific packet, we need to capture only packets with that preceding constant. For Example RCtrl = 40, and Mtch = 04620991. This means compare 4 bytes starting at byte position 0, and the 4

bytes must equal 04620991. Likewise, this could be shortened to RCtrl=21, Mtch=6209. This means compare 2 bytes starting at byte position 1.

3. The remaining fields are used to target the data within the response and convert the data to an understandable value.

   RPos - targets the data within the response packet. The First byte of RPos determines the bit position within the response where the data bits start and is the hexadecimal bit location. The first position is position 0.
   The 2nd byte of RPos determines how many bits represent the data. For example RPos=1008, indicates the data is 8 bits in length. Whereas RPos=1010, indicates the data is 16 bits in length

   MATH – The Math is applied directly to the data bits targeted by RPos.

   Result = Data * XXXX / YYYY + ZZZZ

   Where XXXX is the multiplication field x:
   YYYY is the division field /:
   ZZZZ is the addition field +:

   For example, if the data was temperature and the data was 1/10th of a degree and the data was offset by -40, the math would be:
   result = Data * 1/10 –40
   Since all the fields are hexadecimal,
   result = Data * 0001/000A+FFD8
   x:0001, /:000A, +:FFD8
   Note that the -40 is converted to signed 32-bit value; FFD8, which is equivalent to -40.

   This basic equation can be further altered with the **Output Format** Field. The Output Format functions, except for 0F, are applied to the Math results.
   For example if Output Format = 03, the equation becomes: Result = (Data * X/Y + Z)/100

CODES FOR MY VEHICLE?

Why doesn't UltraGauge support my vehicle? Codes are created by the manufacturer and are proprietary and are not released to the public. The codes that are available have leaked out or were reverse engineered by an automotive enthusiast. UltraGauge can access parameters for which codes are known. To learn more: http://ultra-gauge.com/ultragauge/support/ultragauge-mx-FAQ.htm