Reverse Engineering

Utilising X-Analyser Advanced Features

Richard T. McLaughlin – Warwick Control

In this article, we will show you how to used advanced features of X-Analyser to assist with a couple of reverse engineering tasks. Often in the aftermarket industries, there is a need to ascertain where certain parameters are in CAN data frames. Here we will show you how to observe and experiment with a vehicle where the Engine Speed and Throttle position parameters are located in a CAN frame. This will utilise the X-Analyser CAN Signals Editor. We will also show you how to utilise the X-Analyser transmit function to compare the raw CAN data with Signals data.

The Test Scenario

The scenario here is a recording was taken observing Throttle Position and Engine Speed. One assumption that is made is that it is known that most cars of this manufacturer utilise the Motorola Backward bit format. Refer to the prior article to review this format.

On moving the throttle and observing engine revs, it is observed in ID 0x201 that the two leftmost Bytes (Bytes 0 and 1), and the second Byte from the right (Byte 6) change. It is safe to assume that Bytes 0 and 1 make up a 16-bit signal that would most likely be used for Engine Speed, and Byte 6 is probably Throttle Position, as it is only an 8-bit representation of percentage (0 to 100%) of the position. This is shown in Figure 1 – the Raw Data Trace display in X-Analyser.

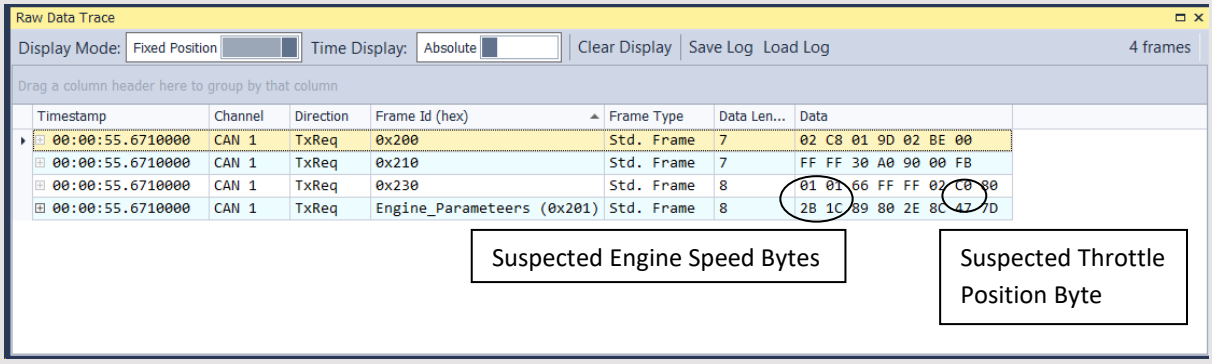

Figure 1. Illustrating possible Engine Speed and Throttle Position

Considering the 16-bit signal (Engine Speed), 2 to power of 16 will get a maximum of 65,535. Therefore, you need to ascertain a multiplier the will represent a sensible maximum RPM, e.g. 8,000 RPM or 16,000 for higher performance vehicles. On recording data from the vehicle, it was found that Bytes 0 and 1 give a reading of 2B 1C (Hex) at approx. 2700 RPM. This will allow you to play around with the multiplier in the Interactive Signal Editor to ascertain a correct multiplier. Utilising a multiplier of 0.25 (16K RPM max), the values match quite closely on testing with the Interactive Signal Editor used in conjunction with Transmit function.

Similarly, it is observed at the same time that Byte 6 is at a value of 47. The 8-bit signal (2 to the power of 8) will get a maximum of 255. As we are expecting a maximum of 100% throttle position, it safe to assume the multiplier will be somewhere around 0.5. On testing with the Interactive Signal Editor used in conjunction with Transmit function, the value of 47 relates to 35.5%.

Setting up X-Analyser for this test scenario – The Signals Editor

In X-Analyser, go to the analysis tab and click on Show Configuration. In here you will see the components in the current project. Select Add Component. In the Add New Component Window, click on the Signal Editor. Once you close these two windows, you will see in the main display that there is now a Signal Editor tab. In there, right click on Message Name, and then click on New Message. Here you can edit the name of the message, its ID number and Message Length. Also, you can add Signals by right clicking on the Message Name, then click on New Signal.

In there, you can edit the Signal name (e.g. Engine_Spd), its Bit Format (Motorola), its Start bit and Bit length (48 and 8 in this case), its Multiplier and units. Here we tried 0.25 to see if this is a sensible amount. The min and max will be automatically filled in. Figure 2 shows an example for illustration.

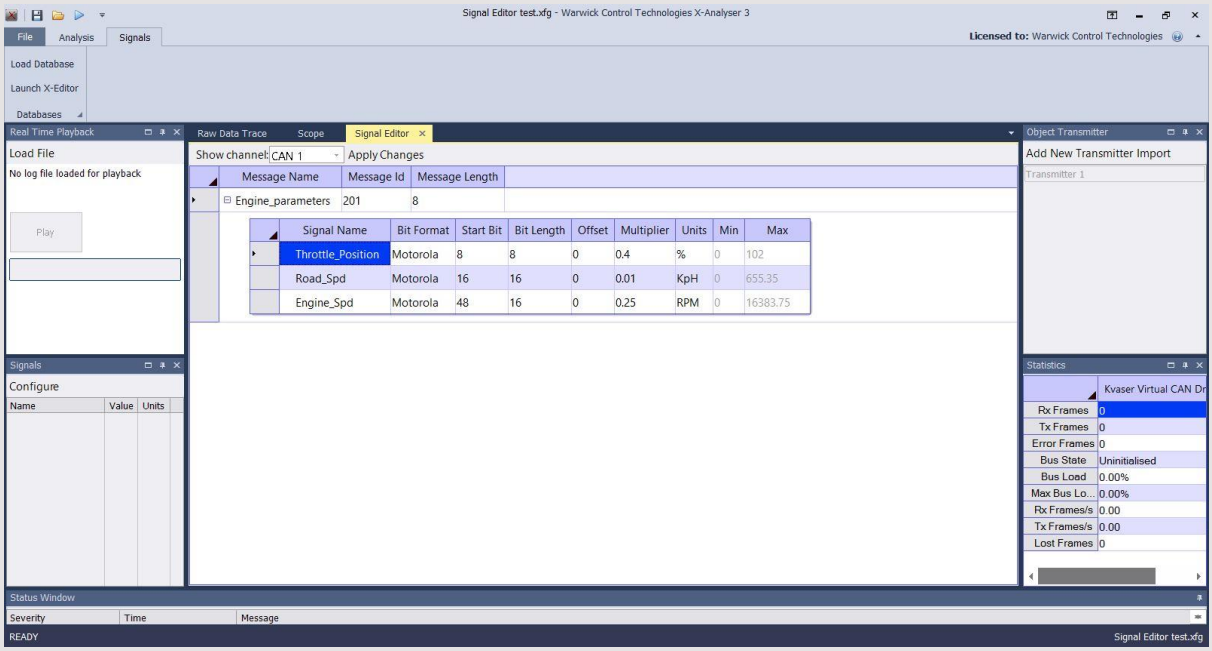

Figure 2. Configuring the Signals Editor

It is also possible to edit the Message and Signals by going to the Signals tab and click on Load Database. Here you will see a display that allows you to load in a pre-built CAN Signals Database file. Also, there is a working area at the top (Signals Editor) that allows you to build up a quick signals file as shown Figure 3.

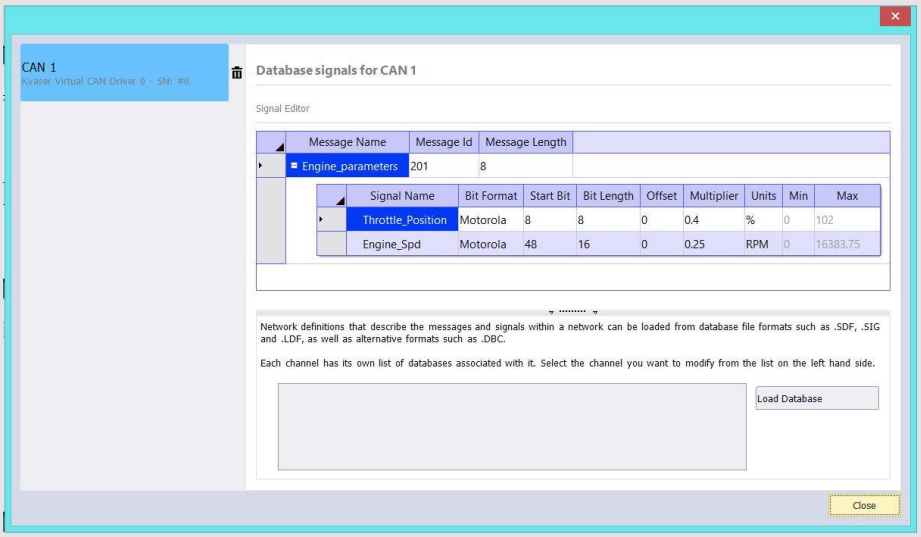

Figure 3. Short-cut to CAN Signals Editor

Once this is built up to your satisfaction, go to the Signals Panel on the left side of the main display, and click the Configure button. See Figure 4 for the resulting display. Here you will see a choice of the signals that you have built in the Editor. Select the two shown there. This will allow you to see the real values compared with the raw CAN data.

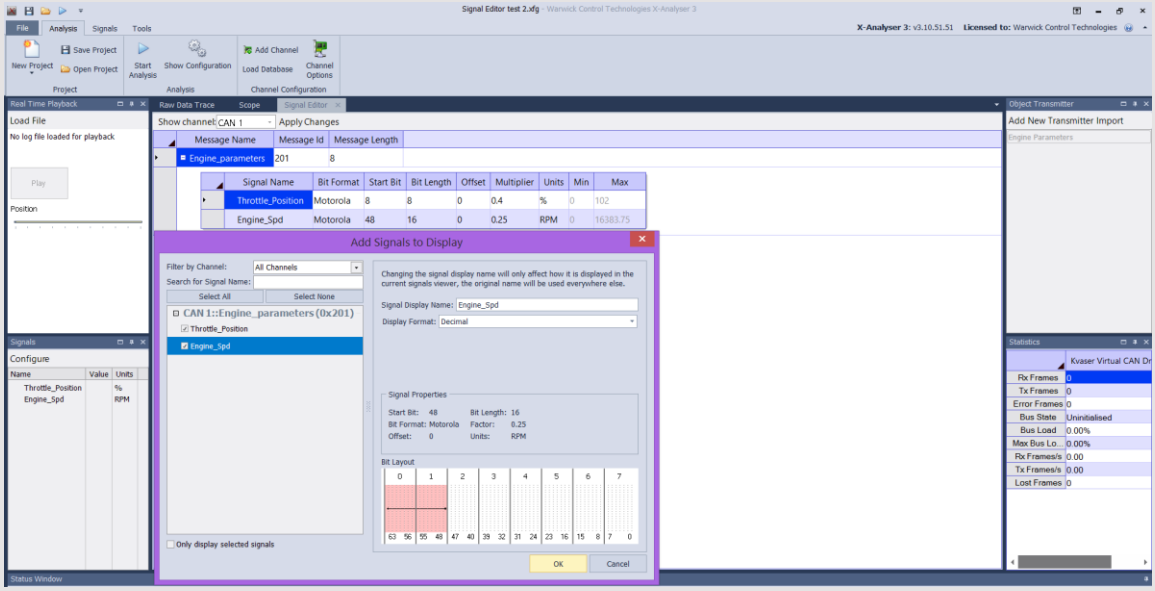

Figure 4. Configuring the Signals Panel

The CAN Transmitter

Now we need to set up a Transmit function to compare the raw CAN data with the Signals you have built up. Here you will be able to change the raw data, compare it with the Signals data, and change the Signals parameters (e.g. Multiplier) if necessary.

Go to the Object Transmitter box at top right of the X-Analyser main display, and click on Add New Transmitter. As shown in Figure 5, you can build up a message and change its data to simulate what occurred in the car. You can set a repetition rate (Delay), give it a name, designate the CAN ID and Byte structure. In the example in Figure 5, we have built it to simulate the scenario described above (D0 & D1 2B 1C, and D6 47). Ensure you tick the Enable auto-repeat box.

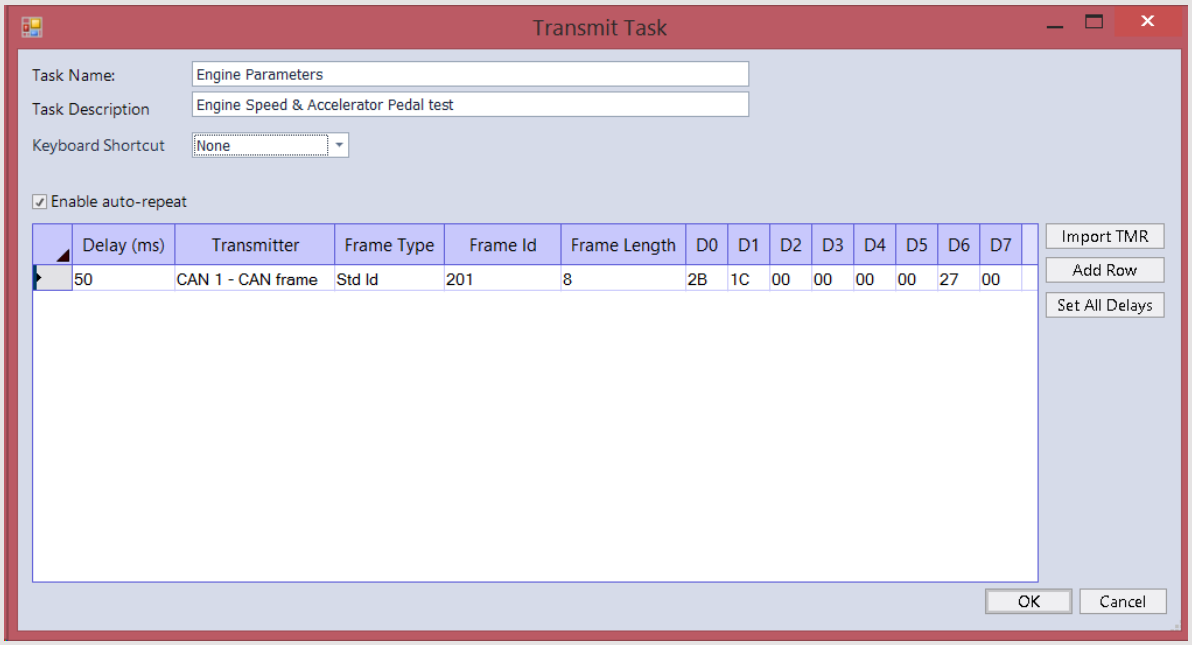

Figure 5. Object Transmitter setup

Click OK, and you are back to the main display. You will now see a transmit object in the Object Transmitter window top right.

You may want to move the Signal Editor to a split screen position as shown below. This will allow you to observe the raw CAN data while editing the Signal parameters. This is accomplished by clicking and grabbing the Signal Editor tab and dragging in to the bottom position of the main display as shown in Figure 6. Once this is done, click on the Start button and click on the Engine Parameters Transmit Object. You will have a display that appears as in Figure 7. Here the Object Transmitter is transmitting the CAN frame ID 201 with you set values to the X-Analyser to display the Raw Data (D0 & D1= 2B 1C, and D6 47) and the Signals data (Throttle Position = 15.6%, Engine Speed = 2759 RPM). This is good estimation of our theoretical concept of the multipliers of the data.

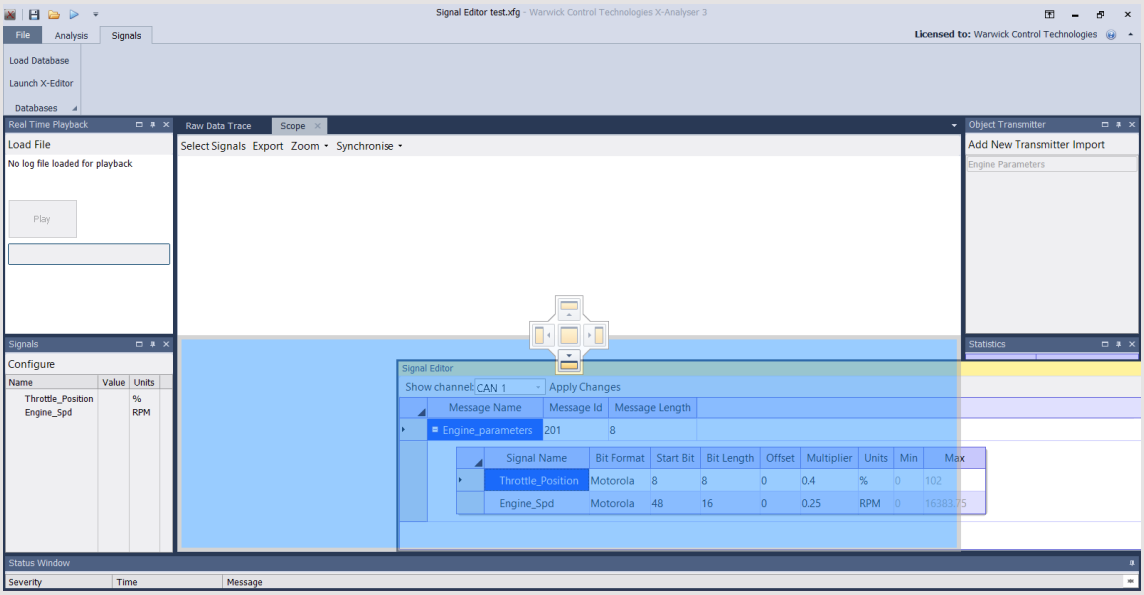

Figure 6. Repositioning the Signals Editor

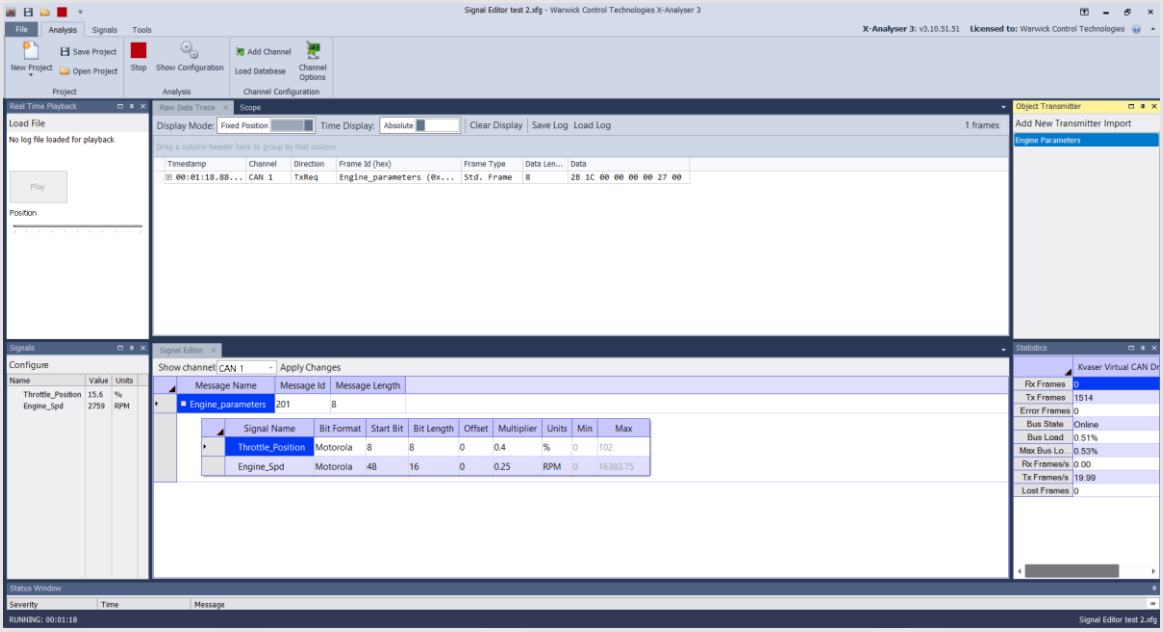

Figure 7. Observing Signals data compared with Raw data

From here you can interactively edit the multiplier in the Signal Editor, and you can change the data in the Object Transmitter (Engine Parameters in this example) – this is done by right clicking on the Transmitter object and selecting Edit Engine Parameters (in this example).