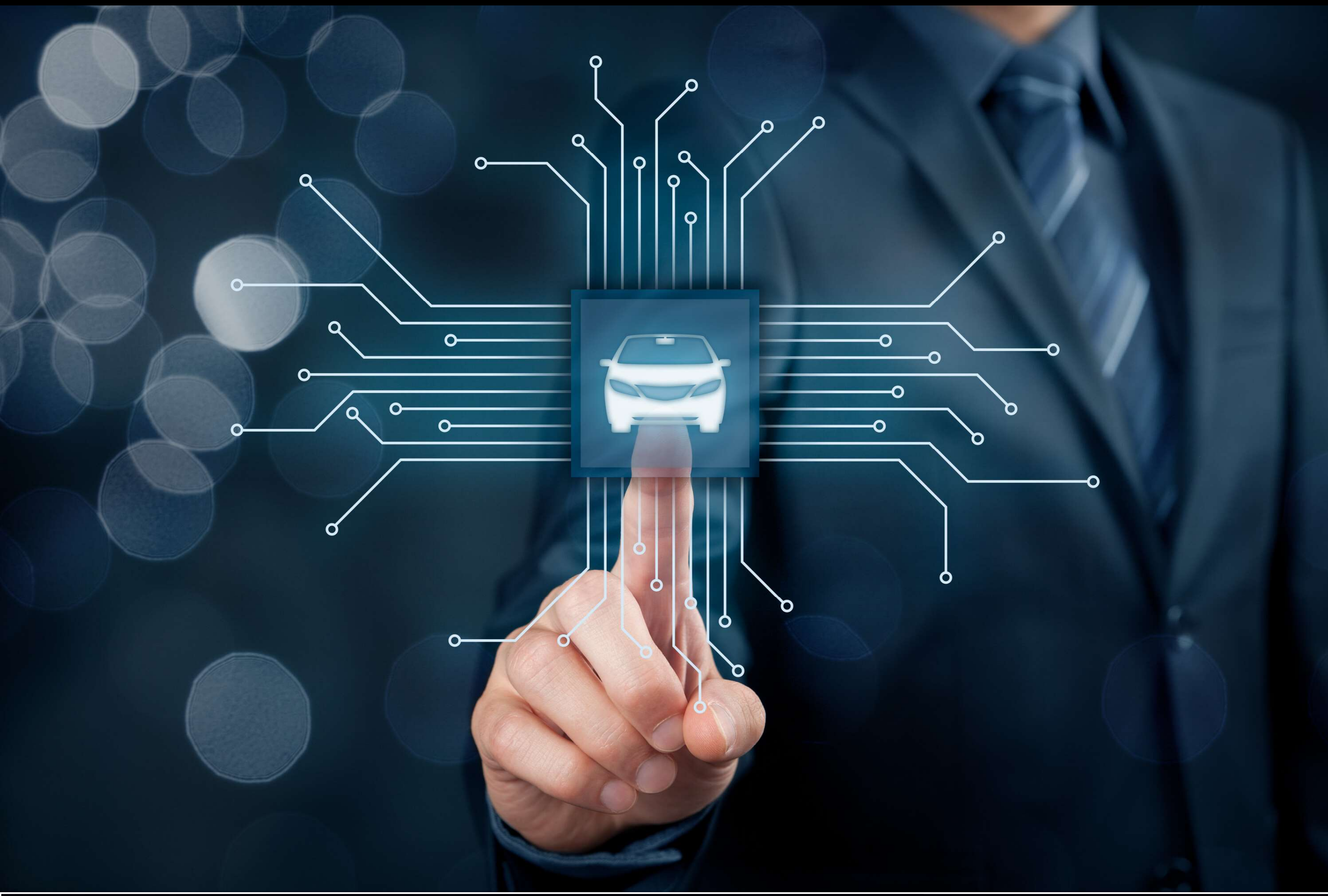
locktronics™

Simplifying Electricity

CAN bus: under the hood



MATRIX

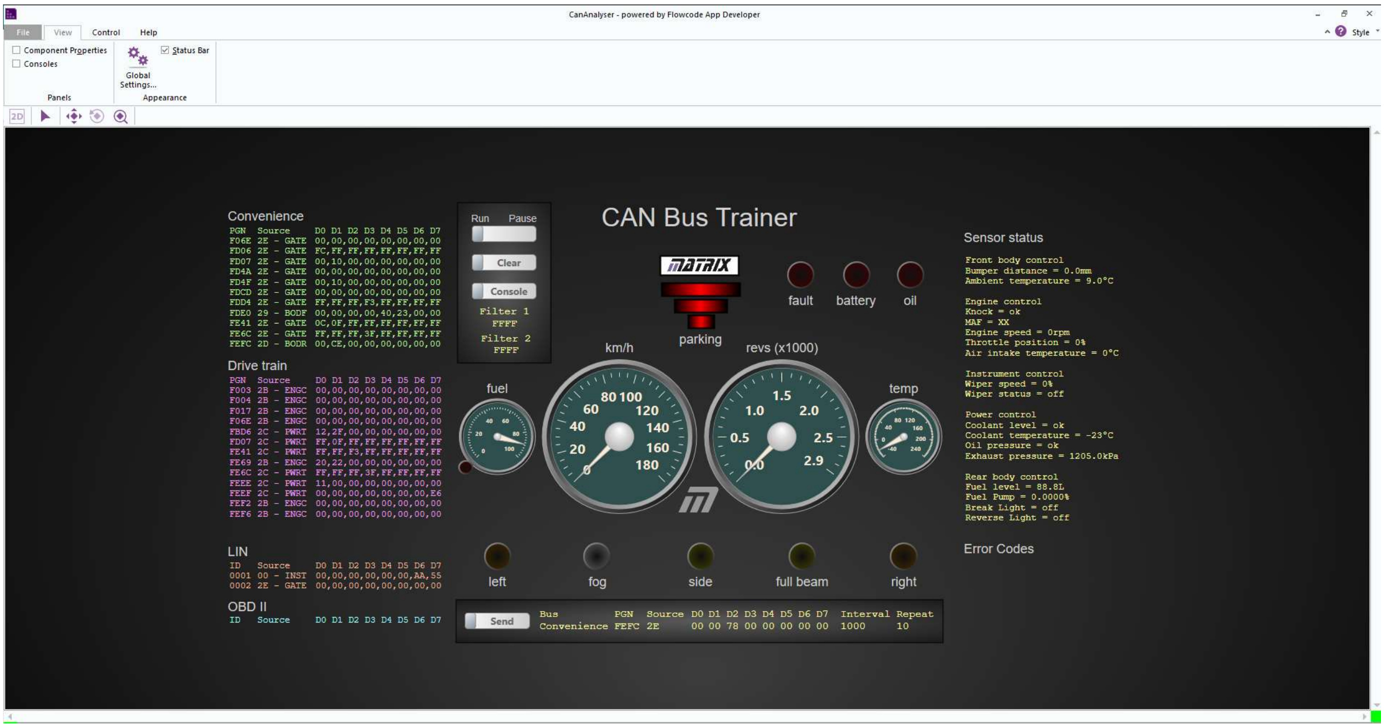
CP0411

www.matrixtsl.com

Contents

Worksheet 1

Adding a diagnostic PC to the system



Over to you:

1. Using a standard USB cable connect the Gateway MIAC to your PC.
   No drivers are needed for the MIAC. (Number removed - not an instruction.)
2. Open the “Open CANAnalyser.bat” application.
   (See the ‘Can and LIN bus reference’ section for how to find this program.)
3. Press the ‘GO’ button on the top left hand side of the screen
4. The on-screen instruments should now reflect what is happening on the CAN bus.
5. Make sure that you are familiar with the information in the document ‘CAN and LIN bus system reference’. This contains information on how to use the diagnostic software as well as information on the J1939 CAN bus scheme, the hardware and software etc.
   You don’t need to memorise this information. Refer to it as needed as you work through the worksheets.

Worksheet 2

Decoding a value in a CAN bus message

ID1	ID2	ID3	D0	D1	D2	D3	D4	D5	D6	D7
FE	EE	29	23	16	00	12	00	29	20	01

CAN bus messages in J1939 have two parts:

- ID 1,2,3 - the Parameter Group Number (PGN), identifying the type of message;
- D0 to D7 - the message data - eight bytes of data.

For example PGN FEEE is the PGN for engine temperature data. ID3 is the unique reference for the ECU that sent the message, in this case, 29.

Other codes starting with 'FEEE', include :

D0 - the PGN for coolant temperature, D1 for fuel temperature,

D2 and D3 for engine oil temperature, D4 and D5 for turbocharger oil temperature,

D6 for engine intercooler temperature, and D7 for intercooler thermostat opening status.

Over to you:

1. Follow instructions in 'CAN bus and LIN bus fundamentals' to start your system.
2. Make sure all switches are off and clear any faults with the scan tool.
3. Open the "Open CANAnalyser.bat" application and press the 'GO' button.
   The on-screen fields show messages on the CAN and LIN buses in the system. Activating switches and potentiometers allows you to monitor the buses and decode these messages.
4. Click on the temperature dial. It will be highlighted by a red square. Another red square appears in the Drivetrain bus text field, highlighting byte D0 of PGN FEEE to indicate that this is the byte that gives temperature information.
5. Adjust the coolant temperature potentiometer on the Powertrain board.
   On the Diagnostic software screen, the temperature dial and byte D0 of PGN FEEE alter correspondingly.
6. Using the CAN bus look-up table in the Reference section, you see that the scaling used is 1^0C per bit. The offset is -40^0C.
7. For example, suppose that byte D0 contains the hex number 82. The decimal equivalent is (8 x 16) + (2 x 1) = 130. Subtracting the offset gives a temperature reading of 130 - 40 = 90^0C. Adjust the coolant temperature potentiometer until the temperature dial reads 90^0C and look at the contents of PGN FEEE byte D0. What do you expect the contents of byte D0 to be when the coolant temperature is 100^0C?
   Test out your answer!

Worksheet 2

Decoding a value in a CAN bus message

So what?

The process you have just witnessed is:

- Powertrain MIAC:
    - samples the value of the potentiometer voltage;
    - converts the potentiometer voltage into a temperature value;
    - sends this value in a FEEE message on the CAN bus.
- Gateway MIAC:
    - receives the FEEE message from the CAN bus;
    - strips the temperature data from the FEEE message;
    - passes this temperature value to the PC software.
- PC diagnostic software:
    - displays the value of temperature on the screen.

The J1939 standards committee dictated that, for ‘Parameter Group Number FEEE Suspect Parameter 110 - Engine Coolant Temperature’, just one byte will be used (D0) and, in this byte, the hex number represents the temperature value offset by -40^{0}C.

Putting analogue values in the data bytes is easy - it’s just the number with an offset. Putting digital states in the data bytes is a little harder.

Over to you:

In the Reference section:

1. Read through the section on ‘CAN messages in close up’ .
   This explains how information is encoded onto a CAN message.
2. Read the following sections:
    - CAN bus messages;
    - Binary and hex refresher;
    - CAN bus look up table;
    - CAN messages in close up;
    - Full CAN bus look up table.

Worksheet 3

Encoding a state in a CAN bus message

`FE6C 2C - PWRT FF,FF,FF,3F,FF,FF,FF,FF`

'FE6C' is the Tachograph Parameter Group Number (PGN).

It is used to convey information on the vehicle direction and triggers the reversing light.

Over to you:

1. On the diagnostic software, focus on message FE 6C.
2. Press the reverse switch on the Locktronics Powertrain panel.

   The D3 byte should change value from 3F to 7F.

So what?

The table on the right shows how the information on the state of the reverse switch is encoded.

Referring to the 'Full CAN bus look-up table' in the Reference section:

- the offset byte number is 3, so byte D3 of PGN FE6C is used.

You can see that byte number 3 contains the number 6. So bits 6 and 7 are used (remember that for binary we count right to left and start at 0). The effect of the reverse switch is to toggle bit 6 off (0) and on (1).

Reverse switch in D3 of FE6C

Off value in **h**ex:	3	F
Off value in binary:	**00**11	1111
On value in **h**ex:	7	F
On value in binary:	**01**11	1111
Bits used:	XX00	0000

Worksheet 3

Encoding a state in a CAN bus message

Over to you:

1. On the diagnostic software focus on message FE41.
2. Press the brake pedal switch on the Locktronics Powertrain panel. Byte D2 should change its value.
3. Using the information in the 'Full CAN bus look-up table' in the Reference section, fill in the fields opposite for the brake switch.

So what?

- Interpreting the information in the CAN bus look-up table is not intuitive, but ultimately it is simple - two bits are used to convey the relevant information.

Over to you:

- On the diagnostic software screen, click on each of 'Left', 'Side', 'Full beam', and 'Right' in turn.
- Notice that the appropriate parts of the Drivetrain CAN messages are highlight-ed as you do so.
- Use this information and the information in the 'Full CAN bus look-up table' in the Reference section to complete the tables opposite.

Brake switch in D2 of FE41

Off value in **h**ex: ____ ____

Off value in binary: ____ ____

On value in **h**ex: ____ ____

On value in binary: ____ ____

Bits used: ____ ____

Left switch in ____ of FE41

Off value in **h**ex: ____ ____

Off value in binary: ____ ____

On value in **h**ex: ____ ____

On value in binary: ____ ____

Bits used: ____ ____

Right switch in ____ of FE41

Off value in **h**ex: ____ ____

Off value in binary: ____ ____

On value in **h**ex: ____ ____

On value in binary: ____ ____

Bits used: ____ ____

Side switch in ____ of FE41

Off value in **h**ex: ____ ____

Off value in binary: ____ ____

On value in **h**ex: ____ ____

On value in binary: ____ ____

Bits used: ____ ____

Full beam switch in ____ of FE41

Off value in hex: ____ ____

Off value in binary: ____ ____

On value in Hex: ____ ____

On value in binary: ____ ____

Bits used: ____ ____

Worksheet 3
Encoding a state in a CAN bus message

So what?

To encode a state in a CAN bus message J1939 dictates that two bits are needed. In practice, only a single bit is needed.

So why does J1939 dictate that we use two?

The specification states that the following commands correspond to the four possible values of those two bits:

Binary	**command**
00	off
01	on
10	no change
11	no change

There is a very good reason for this:

In the J1939 standard, the same Parameter Group Number can be used for several Suspect Parameter Numbers (SPNs). For example, the PGN FE41 is used for seven different lighting commands (i.e. seven different SPNs).

This means that there has to be a way for an ECU to send a command relevant for just one SPN without affecting the other SPNs.

The 'no change' values allow an ECU to do this. It just sends the desired value for the two bits of the relevant SPN and sends '11' for all others.

`Example:`

The CAN bus command for turn left:

PGN FE41, SPN 2367

If we send this command:

`FE41 2C 00 40 00 00 00 00 00 00`

this sends a command for 'turn left state on' and all other FE41-based SPN states to 'off'.

(Notice:

- **FE41** is the PGN;
- **2C,** the Powertrain node**,** is the source of the command;
- the next eight bytes are the message.)

whereas, in the command:

`FE41 2C FF 4F FF FF FF FF FF`

the Powertrain sends a command for 'turn left state on' and all other FE41-based SPN states to 'no change'.

Correspondingly, the command:

`FE41 2C FF 0F FF FF FF FF FF`

sends a command for 'turn left state off' and all other FE41-based SPN states to 'no change'.

Worksheet 4

Sending a CAN bus message

The diagnostic software allows you to send CAN messages on either the Convenience bus or the Drivetrain bus. Click on ‘Bus’ in the ‘Send’ panel to switch between them.

You can set up a message by just clicking on the different parts of the message and entering the required value. (‘Source’ is fixed at 2E, the Gateway.) Equally, you can send the same message a set number of times at a defined interval - for example ten messages at an interval of 1000ms as shown in the image above. Once the message is configured, press ‘Send’.

Over to you:

1. Control the reversing light on by sending the messages detailed opposite.
2. Send fifty repeated messages at an interval of 100ms to turn on the reversing light.

So what?

Each ECU on the CAN bus looks for messages with a particular ID.

The ECUs can’t tell where the messages come from. They don’t care.

You are sending fifty messages 100ms apart. However, at the same time, the Powertrain ECU is sending FE6C PGN messages at regular intervals with a ‘reverse light off’ message.

So, after five seconds, (= 50 x 100ms), the system reverts to turning the reverse light off.

Over to you:

1. You now know how to fool the system into turning the Left, Right, Side lights and Full beam lights on.
   Construct and ‘Send’ messages into the Drivetrain bus to control these lights to check your understanding.

Reverse switch in D3 of FE6C

Drivetrain bus

```
Bits used:              XX00 0000
On value in binary:     0111 1111
On value in hex:         7    F
Off value in binary:    0011 1111
Off value in hex:        3    F
```

Reverse on:

```
 PGN ID3 D0 D1 D2 D3 D4 D5 D6 D7
FE6C 2E  FF FF FF 4F FF FF FF FF
```

Reverse off:

```
 PGN ID3 D0 D1 D2 D3 D4 D5 D6 D7
FE6C 2E  FF FF FF 0F FF FF FF FF
```

Worksheet 4

Sending a CAN bus message

Over to you:

1. Here are further commands that affect the CAN bus system and diagnostic software.

   Explore each, using the 'Send' function in the diagnostic software.

So what?

With a little understanding of binary and hexadecimal notation, commands can be constructed from information given in the CAN bus look up table in the Reference section.

Engine speed:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
F004 2E  FF FF FF 4E 20 FF FF FF
```

Interval: 200 ms   Repeat 100 times

Displays 2500 rpm on the rev counter.

Fuel level:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
FEFC  2E  FF 7D FF FF FF FF FF FF
```

Interval: 200 ms   Repeat 40 times

Displays fuel level of 50 litres.

(50 / 0.4 = 125 decimal = 7D hex.)

Left indicator on:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
FE41  2E  FF 40 FF FF FF FF FF FF
```

Interval: 200 ms   Repeat 40 times

Vehicle speed:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
FEF1  2E  FF 64 29 FF FF FF FF FF
```

Interval: 200 ms   Repeat 40 times

Displays around 60mph.

(6429 is 100Kmph - around 60mph.)

Vehicle horn on:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
FDD4  2E  FF FF FF F4 FF FF FF FF
```

Interval: 200 ms   Repeat 40 times

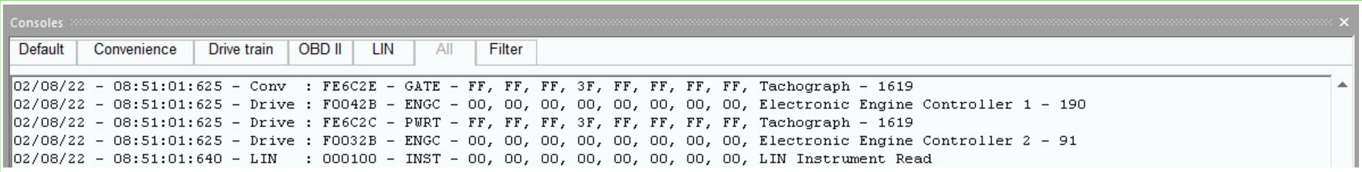
Vehicle horn off:

```
PGN  ID3  D0 D1 D2 D3 D4 D5 D6 D7
FDD4  2E  FF FF FF F0 FF FF FF FF
```

Interval: 200 ms   Repeat 40 times

Worksheet 5

Using the console



The PC diagnostic software collates the messages on each of the CAN and LIN buses and shows the latest message on each Parameter Group Number.

What it does not show you is a historic record of the messages on the system. This is what the console does. The console allows you to track the history of messages and where they came from.

Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Click on the 'Console' button to activate the console.
4. You can move the console window by undocking it - use the mouse 'left click and hold' on the dotted top part of the console window and drag it free from its position. You can also resize it.
5. Click on the Drivetrain tab to see all messages on the Drivetrain bus.
6. Click on the Convenience tab to see all messages on the Convenience bus.
7. You can pause and restart the console at any time by clicking on 'Pause/Run' button.

So what?

- The messages on the Drivetrain and Convenience buses never stop.
- The CAN bus message table in the Reference section shows that some messages are transmitted when there is a change in a particular parameter while others are transmitted every 100ms.
- Every message has a date and time stamp.

Over to you:

1. Click on the 'Console' filter.
2. Enter 'FE41', the Lighting Parameter Group Number.
3. In the console, click on the Convenience tab.

So what?

- You now see only messages containing the PGN FE41 on the Convenience bus. This makes it much easier to track sequential messages in the system .
- The filter tab identifies which bus the message is on.

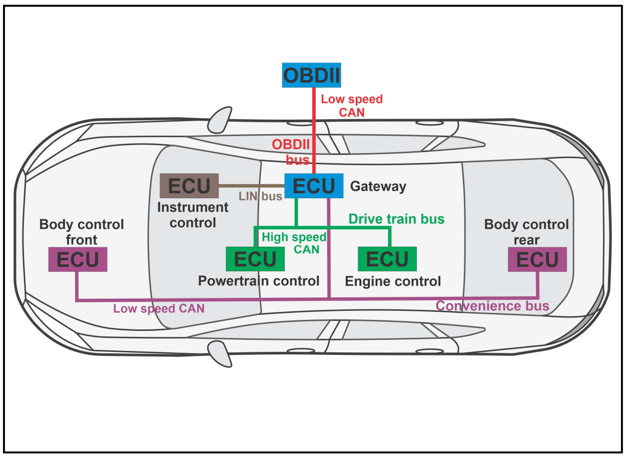
Worksheet 6

Startup routines

Safety is at the heart of much of the design of automotive systems.

One of the benefits of a computer-based network in a vehicle is that, on startup, the vehicle can carry out basic diagnostic checks to make sure that the main parts of the vehicle are functioning properly.

The startup routine is a good example of this.



Over to you:

1. Referring to the CAN bus system graphic in the Reference section, notice that the ignition switch on the Power and fuse panel switches +12V to the ECUs in the system, but that the Gateway ECU is always powered directly from the battery.
2. Turn off the ignition switch.
3. Start up the PC diagnostic software and press ‘GO’. There is no traffic on the CAN buses so the display will contain little information.
4. Click on the ‘Filter text’ field and enter ‘FD50’, the Programme Group Number for Startup messages. This instructs the console to display only messages with the FD50 PGN.
5. Turn on the ignition switch.
6. Make a note of the messages recorded in the console.

So what?

The CAN system graphic shows that the Gateway is permanently powered. The ignition switch also feeds into input 1 of the Gateway so that it can detect when power is applied to the rest of the vehicle.

So what .....

When the Gateway senses power, it pauses and then sends a ‘Who is there’ message to the other ECUs.
This is PGN FD50, D0 = FF.

The Gateway does this twice - once on the Convenience bus and once on the Drivetrain bus. It then waits for a reply from all ECUs to make sure all ECUs are in place.

Over to you:

1. Clear any faults in the system using the OBDII tool.
2. Turn off the ignition switch
3. Disconnect the CAN bus from the Body Control rear ECU.
4. Turn on the ignition switch .
5. What do you see on the console filter? How is this different to when the Body Control rear ECU is connected?

So what?

Startup routines, and any resulting actions, vary from one vehicle to another.

In practice, the Gateway may check for the presence of ECUs regularly.

Worksheet 7

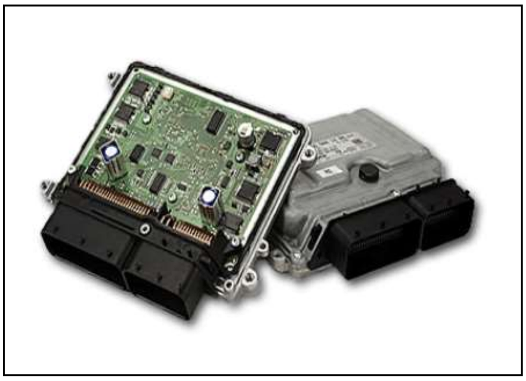
Gateway functions

Our Gateway ECU several functions:

- it acts as a 'go between' for the Convenience and Drivetrain CAN buses and Instrument LIN bus;
- it collates and passes messages to the OB-DII analyser;
- it monitors battery voltage.

The function of the gateway in other vehicles may be different.

Photograph shows a typical car ECU.



Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Click on the 'Console' button to activate the console.
4. Click on the 'Console filter' text.
5. Enter 'F06E' - the fuel pump PGN.
6. Press 'Pause' to stop the console from gathering data.
7. The 'Filter' tab shows you the messages and their source. Notice that a 'F06E' message starts on the Convenience bus and is mirrored on the Drivetrain bus a few milliseconds later.
8. Choose a message and note the time of the original message on the Drivetrain bus and the repeated message on the Convenience bus.
9. Click in turn on the Drivetrain tab and the Convenience tab and make sure that you can find the messages with the same time stamps.

So what?

This exercise shows how the Gateway mirrors messages on the system using the F06E message. The Engine Control ECU sends this message on the Drivetrain bus. The Gateway picks it up and mirrors it onto the Convenience bus. The Body Control rear ECU picks up the F06E message on the Convenience bus and alters the speed of the fuel pump motor according to the information in bytes D0 and D1 of the F06E message.

Over to you:

1. Repeat the exercise with PGN FEEE - the engine temperature PGN.

So what?

- FEEE engine temperature messages originate from the Powertrain ECU, 2C, and are sent on the Drivetrain bus.
- However the information is needed by the Instrument cluster and so it is mirrored by the Gateway onto the Convenience bus.

Worksheet 8

OBDII CAN bus

The OBD II scanner communicates with the Gateway via a CAN bus. It uses a protocol defined by a number of different specifications, governing the physical connection, the CAN bus protocol and the fault codes, ensuring a level of interoperability between scanners and vehicles. The Matrix CAN bus system has a full OBDII protocol built into it.

The photograph shows the standard OBDII socket.

Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Disconnect the Body Control rear ECU.
   A fault should be registered.
4. Ensure that the scan tool is connected to CAN 3 on the Gateway ECU using the OBD II socket. (Refer to the CAN bus system graphic if necessary.)
5. Click on the 'Console' button on the PC diagnostic software to open the console. Reposition it so that you can see the screen and the console.
6. Click on the OBDII tab to see all the traffic on the ODBII CAN bus.
7. Using the OBDII scan tool, request the fault code and then reset the fault.

So what?

- You should be able to see some CAN communication on the OBDII CAN bus. to and from the Gateway and the OBDII.
- The OBDII CAN bus protocol, governed by J1979 and other standards, is not discussed in this document.
- Vehicles support the OBDII standard to give fault codes and selected data on the status of the vehicle through ECU functionality.
- The CAN bus system supports the following fault codes:

P0090 - Fuel level circuit error
B1318 - Battery Voltage Low
U0141 - Lost Communication Body Control Front
U0142 - Lost Communication Engine Control
U0143 - Lost Communication Instrument Control
U0144 - Lost Communication Powertrain Control
U0145 - Lost Communication Body Control Rear
C1223 - Lamp Brake Warning Failure

Over to you:

1. Use the potentiometer on the Fuse panel to simulate a battery low fault.
2. Use the scan tool to get the fault code and fault description
3. Put the potentiometer back up to 12V
4. Use the scan tool to reset the fault.

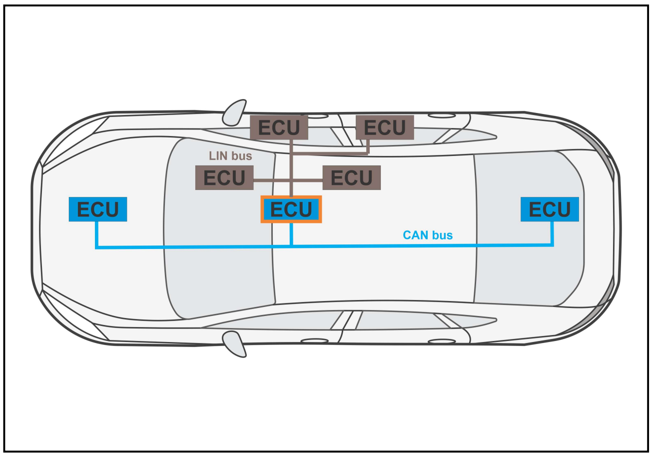
Worksheet 9

LIN bus

The LIN bus is a low data rate, low reliability bus. Its advantage over CAN bus is that it uses only one wire rather than two so it is cheaper to implement.

It is used for non-critical vehicle functions, like electric windows, instrument control etc. Most car systems contain several LIN buses, linked to different ECUs to give the car its functionality.

Image shows LIN bus topology.



Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Click on the 'Console' button to start the console and reposition it so that you can see both the screen and the console.
4. Click on the LIN bus tab to allow you to see all of the messages on the LIN bus.
5. On the Body Control rear node adjust the fuel level potentiometer so that it is fully clockwise, i.e. maximum fuel level.
6. Slowly adjust the potentiometer turning it anticlockwise.
7. On the PC diagnostic software watch the LIN bus data as you do so.

So what?

- The LIN bus is simpler than the CAN bus but has a similar data structure, e.g. eight data bytes in a LIN packet.
- The LIN protocol is 'master/slave'. The master node requests information and the slave node provides it.
- Nodes control the information on the bus, and so it does not need the complex encoding required by the open access system used in the CAN bus.
- The first byte transmits the status of the fuel level warning lamp. '01' means 'on'. '00' means 'off'. No other message accesses data byte D0. Only the Gateway will transmit this message.

Over to you:

1. Activate the switches on the Instrument panel in turn.
2. Verify the information given in the LIN bus look-up table below.

LIN ID	ID Label	Data Label	Transmission Rate	Byte(s)	Bit(s) in the byte	Bit Length	Data Mask HEX	Data Byte(s) BIN	Scaling or scheme
1	Switch Poll Request	Flag Wiper %	80ms	1	0	8	FF 00 00 00 00 00 00 00	xxxxxxxx	0.4% per bit
1	Switch Poll Request	Lamp Low	80ms	1	0	1	00 01 00 00 00 00 00 00	0000000x	On or Off
1	Switch Poll Request	Lamp High	80ms	1	0	1	00 00 01 00 00 00 00 00	0000000x	On or Off
1	Switch Poll Request	Lamp Left	80ms	1	0	1	00 00 00 01 00 00 00 00	0000000x	On or Off
1	Switch Poll Request	Lamp Right	80ms	1	0	1	00 00 00 00 01 00 00 00	0000000x	On or Off
1	Switch Poll Request	Flag Horn	80ms	1	0	1	00 00 00 00 00 01 00 00	0000000x	On or Off
1	Switch Poll Request	Alive (AA)	80ms	1	0	8	00 00 00 00 00 00 FF 00	xxxxxxxx	
1	Switch Poll Request	Alive (55)	80ms	1	0	8	00 00 00 00 00 00 00 FF	xxxxxxxx	
2	Lamp Status Send	Lamp Fuel	350ms	1	0	1	01 00 00 00 00 00 00 00	0000000x	On or Off
2	Lamp Status Send	Fault Count	350ms	1	0	8	00 FF 00 00 00 00 00 00	xxxxxxxx	>0 = Lamp On

Worksheet 10

LIN to CAN, CAN to LIN

The LIN bus interfaces to the two CAN buses via the Gateway.

Messages are sent from the Instrument ECU to the Gateway. The Gateway then uses the Gateway look-up table to decide whether to put them onto the Convenience bus or the Drivetrain bus.

Photograph shows an OBDII scan tool.

Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Click on the 'Console' button to start the console and reposition it so that you can see both the screen and the console.
4. Add a CAN bus message filter by clicking on the text below the 'Console' button. Enter 'FEFC' and press 'Return'. This displays only CAN messages starting with 'FEFC' - the fuel level message.
5. Rotate the fuel level potentiometer from empty to full and look at the resulting messages as you do so.

So what?

- Parameter Group FEFC is used to convey fuel level data in the vehicle.
- In the FEFC message, byte D0 gives the level of fuel in the system.
- The filter function is useful in getting rid of a lot of messages on the system so that it is easier to see what is going on.

Over to you:

1. Rotate the fuel level potentiometer clockwise so that the fuel tank is 'full'.
2. Slowly rotate it anticlockwise until the fuel warning level light comes on.
3. Pause the console by pressing the 'Run/ pause' button on the diagnostic software.
4. Click on the LIN bus tab on the console to see LIN bus messages.
5. Locate the message where the fuel warning level light is first activated by the Gateway, i.e. where byte D0 first reads '01' and note the exact time when it was sent by the Gateway.
6. Click on the 'Filter' tab.
7. Locate the FEFC message that has the same time of transmission.
   In this message, byte D1 gives a value indicating the fuel level in the 'tank'.
8. At what value of fuel level does the Gateway send an 'activate low fuel lamp' command to the Instrument ECU?

Worksheet 10

LIN to CAN, CAN to LIN

locktronics

So what?

There is a train of messages here:

- The Body Control rear ECU sends fuel level data every second or on change;
- The Gateway intercepts this fuel level CAN bus message;
- It decides whether fuel level is low;
- If it is, Gateway notifies the Instrument panel ECU that fuel level is low.
- The Instrument panel ECU then turns the low fuel warning light on.

Over to you:

1. Familiarise yourself with the LIN bus control for main beam lights on and off.
2. Familiarise yourself with the CAN bus command for main beam lights on and off.
3. Put a filter of 'FE41' on the console..
4. Turn the main beam lights on and off using the instrument panel switch and then pause the PC software.
5. Make a note of the time stamp in the LIN bus tab where the lights go on.
6. In the filter tab track down the FE41 message at the same time.

So what?

- Parameter Group FEFC is used to convey the fuel level in the vehicle.
- In FEFC messages, byte D0 is used to convey the level of fuel in the system.
- The 'Filter' function is useful in getting rid of a lot of messages on the system so that it is easier to see what is going on.

Over to you:

1. Activate the CAN bus.
2. Start the PC diagnostic software.
3. Click on the 'Console' button to start the console and reposition it so that you can see both the screen and the console.
4. Add a CAN bus message filter by clicking on the text below the 'Console' button. Enter 'FEFC' and press 'Return'. This displays only CAN messages that start with 'FEFC' - fuel level messages.
5. Rotate the fuel level potentiometer from empty to full and look at the resulting messages as you do so.

Worksheet 11

Some test codes

locktronics

The following commands can be issued to show features of the system.

To do so, use the SEND feature of the PC diagnostic software.

You need to set the 'Interval,' (time between messages) and 'Repeat' (number of times you want to repeat it).

Remember that the CAN bus nodes will also be transmitting similar messages and you need to overwrite this information.

Command	PGN	D0 D1 D2 D3 D4 D5 D6 D7	Bit pattern
Knock	F017	F0 FF FF FF FF FF FF FF	00000000
Fuel pump	F06E	FF 20 FF FF FF FF FF FF	00000000
Exhaust pressure	FBD6	AA FF FF FF FF FF FF FF	00000000
Battery low lamp	FD06	01 FF FF FF FF FF FF FF	000000xx
Engine oil low	FD07	FF 10 FF FF FF FF FF FF	00xx0000
Wiper on	FDCD	F0 FF FF FF FF FF FF FF	xxxx0000
Horn on	FDD4	FF FF FF 04 FF FF FF FF	00000100
Ambient temp	FDE0	FF FF FF FF 11 30 FF FF	xxxxxxxx xxxxxxxx
Left indicator on should be	FE41	FF 40 FF FF FF FF FF FF	01000000
Right indicator on should be	FE41	FF 10 FF FF FF FF FF FF	00xx0000
Stop light	FE41	FF FF 04 FF FF FF FF FF	0000xx00
Fog light	FE41	FF FF FF FF 01 FF FF FF	000000xx
Running lights on	FE41	0D FF FF FF FF FF FF FF	000000xx
Running lights off	FE41	0C FF FF FF FF FF FF FF	000000xx
Full beam on	FE41	41 FF FF FF FF FF FF FF	xx000000
Full beam off	FE41	0C FF FF FF FF FF FF FF	xx000000
Air intake temp	FE69	XX XX FF FF FF FF FF FF	xxxxxxxx xxxxxxxx
Vehicle direction	FE6C	FF FF FF 40 FF FF FF FF	xx000000
Vehicle speed	FEF1	FF 30 D4 FF FF FF FF FF	xxxxxxxx xxxxxxxx
Startup	FD4F	FF FF FF FF FF FF FF FF	

Notes for the Instructor

Teacher's notes
About this course

Introduction

The first of our automotive network courses, 'CAN and LIN bus fundamentals,' gives an introduction to CAN and LIN networks in vehicles and a guide on how to use multimeters, oscilloscopes and OBDII scan tools to diagnose faults in them.

This course takes the student a step further into CAN and LIN bus protocols and explains how they are used to pass information between the ECUs in a vehicle. It also gives insight into how engineers design a system of ECUs and circuits to provide the functionality that modern vehicles require.

The course makes use of the Matrix CAN bus system, consisting of six educational ECU's and accompanying Locktronics boards which mimic the electrical system in a vehicle. The system uses three CAN bus networks at different speeds and a LIN bus. A Gateway ECU provides communication between the system and OBDII scan tools.

Prior Knowledge

Students need an understanding of electricity in a vehicular context.

Students should have completed the 'CAN and LIN bus fundamentals' course.

Learning Objectives

On successful completion of this course the pupil will have learned about:

- PC-based diagnostics;
- Sensors and actuators in CAN and LIN bus systems;
- CAN bus message encoding;
- LIN bus message encoding;
- ECU and vehicle circuit function;
- Gateway function: CAN to LIN, LIN to CAN;
- Start up routines;
- J1939 CAN bus protocol;
- OBDII functions.

Bill Of Materials

locktronics

The basic CAN bus can be set up with the following parts;

CAN bus II		
2	HP2045	Shallow tray
5	HP2666	Adjustable DC power supply
1	HP2876	12V Power supply
1	HP2705	Poster
7	HP4039	Tray Lid
4	HP5540	Deep tray
5	HP7750	Daughter tray foam cutout
1	HP8681	OBDII bracket
4	HP9564	62mm daughter tray
1	LK3246	Buzzer, 12V, 15mA
6	LK5202	Resistor, 1k, 1/4W, 5% (DIN)
6	LK5206	Resistor, 120R, 1/4W, 5% (DIN)
1	LK5208	250 ohm potentiometer
8	LK5214	Potentiometer, 10K ohm (DIN)
72	LK5250	Connecting Link
17	LK5291	Lampholder, MES
10	LK5603	Red 4mm to 4mm lead, 500mm
10	LK5604	Black 4mm to 4mm lead, 500mm
32	LK5607	Blue 4mm to 4mm lead, 500mm
30	LK5609	Yellow 4mm to 4mm lead, 500mm
2	LK6207	Switch, push to make, metal strip
8	LK6209	Switch, on/off, metal strip
6	LK6574	Red 4mm to 4mm lead, 2m
2	LK6635	LED, red
2	LK6706	Motor, 12V
4	LK6749	12V LED bulb red
4	LK6822	12V LED bulb orange
13	LK6841	12V LED bulb white
1	LK8275	Power supply carrier with battery symbol
6	LK8623	Circuit breaker
6	LK8900	7 x 5 metric baseboard with 4mm pillars
6	MI5550	MIAC NXT
6	LK8654	Black 4mm to 4mm lead, 2m
4	LK7466	Blue 4mm to 4mm lead, 2m
4	LK7155	Yellow 4mm to 4mm lead, 2m
1	COM5825	UK head for HP2876
1	COM5826	EU head for HP2876
1	COM5827	USA head for HP2876

11 03 24

It can be extended to work with the following real car parts, mounted on brackets:

1	HP4548	Cam and crank(Hall Effect) sensor
1	HP5111	Ultrasonic parking sensor
1	HP5785	Intake Air Temperature (Thermistor) sensor
1	HP3181	Throttle position sensor
1	HP7575	Knock Sensor
1	HP0713	Mass Air Flow Sensor
1	HP1409	Coolant level (magnetic reed) switch
1	HP7910	Coolant temperature (Thermistor) sensor
1	HP8738	Exhaust (differential) pressure sensor
2	HP3961	rear light cluster
1	HP3141	throttle valve
1	COM00170	Pipette filler
1	COM4177	4mm diameter tubing, 300mm

Notes for the instructor

Worksheet	Notes
1 Adding a diagnostic PC to the system	Once the hardware is set up, students will need to learn how to use the PC diagnostic software. They should be given a copy of the CAN and LIN bus Reference document that shows how the system works. (You may decide to spend time going through this information with them - it is a little tricky in parts and may be best delivered in small chunks!) They also need access to the poster provided with the kits. This is available in PDF format from the Matrix website and can be reprinted. Before students start, you may need to spend time explaining the concepts of PGN and SPN and going through the basic CAN bus message table.
2 Decoding a value into a CAN bus message	Decoding data in a message is the most straightforward transaction in CAN!
3 Encoding a state into a CAN bus message	Encoding states can be tricky to understand. Knowledge of binary and hexadecimal number systems is needed here. The reference document includes is a refresher page on these.
4 Sending a CAN bus message	Great fun! The 'Send' feature allows you to send CAN messages and see their effect on the system and on the PC screen. Just let students explore this feature for a while.
5 Using the console and console filter	The PC diagnostic software provides a summary of what is happening in the system. However, when we want to track the sequence of information on the various buses, we need a display that shows messages with a time stamp attached. The console provides this.
6 Startup routines	A startup routine is implemented in the system. This checks that all nodes in the system are present, flashes the PC diagnostic software display and powers up the fuel pump for a few seconds. The startup routine can be implemented in software by sending command 'FD4F'.

Notes for the instructor

Worksheet	Notes
7 Gateway functions	In this worksheet, students explore the function of the Gateway and how it transmits selected messages from one bus to another.
8 OBDII CAN bus	This exercise allows students to appreciate that errors are generated internally in the system and are then sent to the scan tool when requested. The system implements the J1979 OBDII standard - although only a small number of commands are implemented. Most modern scan tools can be used with the system. No scan tool is shipped with the kit as these are common workshop pieces of equipment.
9 LIN bus	This worksheet allows students to explore some relatively simple LIN bus protocols.
10 LIN to CAN, CAN to LIN	Here, students explore the function of the Gateway more deeply, looking at its function in conveying LIN messages to and from the Instrument panel. This requires use of the console again. We advise that at least two students work on this together - one to operate the software and one to operate the CAN bus hardware.
11 Some test codes	The CAN bus message specification will be hard to understand for any not well-versed in binary, hex and the encoding of messages on the CAN bus. Extended practice here can resolve those issues.

Version control

Frist release	18 10 22
Update	08 11 22 several changes.
26 04 23	small changes
01 08 23	Reformatted to new style
11 03 24	BOM on page 21 updated