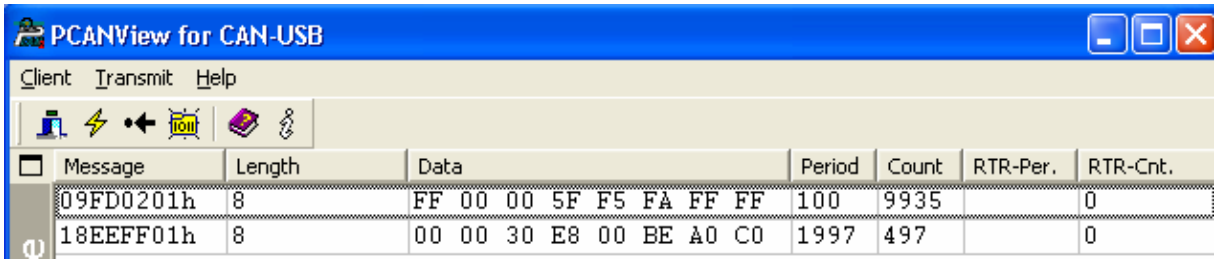
SimNet/NMEA2000 Can Data Sentences

SimNet/NMEA2000 can data sentences are made up of two parts transmitted as one long sequence
1. The message identifier
2. The data frame

The example below also show data length (8 byte), how often each sentence is transmitted, and how many sentences with the same message identifier have been received. The columns for 'Length', 'Period', 'Count', 'RTR-Per' and 'RTR-Cnt' are NOT part of the Can sentence, they are just for information purposes on the Can Viewer.



1. The Message Identifier

What is the Message Identifier ?
The 'message Identifier' is made up of 4 bytes (or 32 bits), only 29 of the 32 bits are used, the top 3 bits are always set to 0. This is also known as the '29 bit identifier'.

How is the 'Message Identifier' composed ?
(The following information refers to the first Message, 09FD0201h, as an example)
Byte 1 (09h)
This is essentially the type of message ID that is being transmitted, and is generally made up of flags. In the rest of this document we shall ignore this byte.
Bytes 2 & 3 (FD02h)
This is the actual 'Parameter Group Number', 'PGN', it is the reference number for the type of data that is being transmitted. FD02h is the PGN number for Wind Speed and Angle Information
Byte 4 (01h)
This is the 'source address', or 'SA' ; the address on the bus which the wind transducer has claimed. Each product (or active) node on the bus will claim its own source address. No two nodes will have the same address. The wind transducer is on address 01h

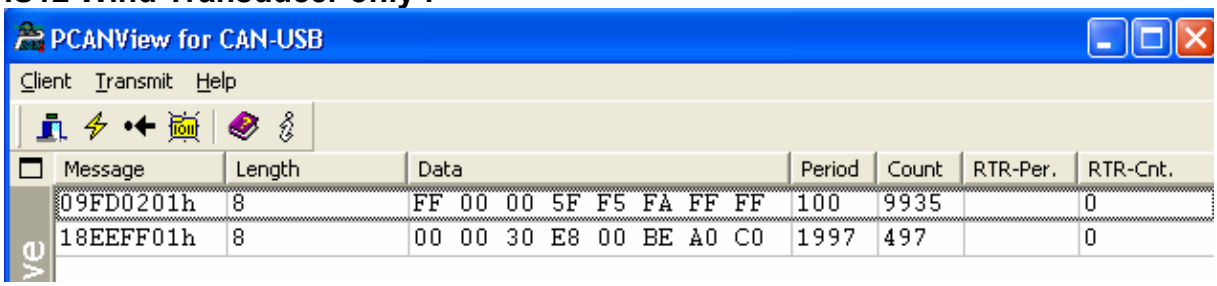
2. The Data Frame

What is in the Data Frame ?
The Data frame can be up to 8 bytes long. This is also known as a 'single data frame'. In general all the bytes are used, but some special PGN's can have as few as only 3 bytes used. Quite often the complete data frame is split into fields. The definitions of where the fields start and stop would be in the specification for each SimNet/NMEA2000 sentence. Some fields may be made up of one or two bits, some may be a few bytes long. It depends on the sentence being used

How is the Data Frame Composed ?
(The following information refers to the first Message, 09FD0201h, as an example)
Field 1 : Sequence Identifier (1 byte) - not used on this product, always set to FFh
Field 2 : Wind Speed (16 bit unsigned) - wind speed to a resolution of 0.01m/s
- (example shows wind speed transmitted as 0000h)
Field 3 : Wind Angle (16 bit unsigned) - Wind angle to a resolution of 10E-4 rad. Range 0-2PiRad
- (example shows wind angle transmitted as F55Fh)
Field 4 : Wind Reference flags (3 bits) - This is set to %xxxxx010
- which means 'this is apparent wind data'
Field 5 : Reserved (all left-over bits) - Rest of the unused bits are set to '1's

So, we have established the make-up of the Message Identifier and the data in the Wind Speed and Angle Sentence, It is now a case of using the same method to analyse each different type of data sentence sent by products on the SimNet/NMEA2000 bus.

IS12 Wind Transducer only :



1. Sentence '09FD0201h' – Wind Speed and Angle
This sentence has been discussed on the introduction page

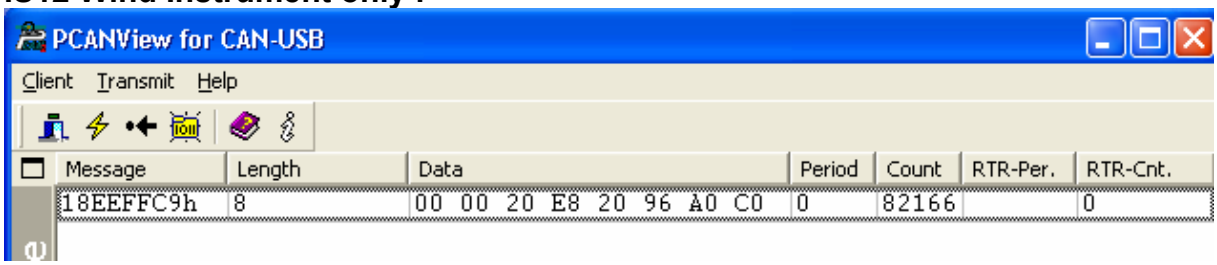
2. Sentence '18EEFF01h' – Address Claim / NAME
Bytes 2 & 3 (EEFFh)
This is the PGN number for 'Address Claim'
Byte 4 (01h)
This is the 'source address' 01h
Data Frame
The data is essentially made up of fields defining what type of product it is, who makes it, what classification it has, and most importantly, the first 21 bits are an identity number, a number which the factory has programmed in, each device of this type will have a unique identity number. Once all the fields in NAME are put together, it makes a unique number which no other product will have anywhere else in the world.

What is Address Claim/NAME used for ?
When each product is powered up on the bus, before anything else it sends its NAME and with it is it last known address on the bus (this is part of the message ID). The other products on the bus listen for every address claim that is sent on the bus, and if one product transmits a NAME with an address that another product has then they will fight for the address. The product with the lowest NAME/unique number (all 64 bits of the data frame) will win the fight. The one that loses will try to claim the next address, this is called 'address bumping' and so it goes on. At any time, the NAME of a product can be requested by a simple sentence, (this is not discussed here) and this can be used to keep track of where products are on the bus.

Who uses Address Claim/NAME ?
Every product must transmit it NAME on power-up and if requested to do so, this is a mandatory requirement of NMEA2000. Therefore you will see xxEEFFxxh regularly appearing in the following examples

Transmission frequency
In this example it shows that the Wind transducer is sending Wind Speed and Angle every 100ms (10 times per second) whereas the NAME is sent regularly every 2 seconds (a 'ping'). If another product is connected to the bus, the pinging will stop.

IS12 Wind Instrument only :



1. Sentence '18EEFFC9h' – Address Claim / NAME

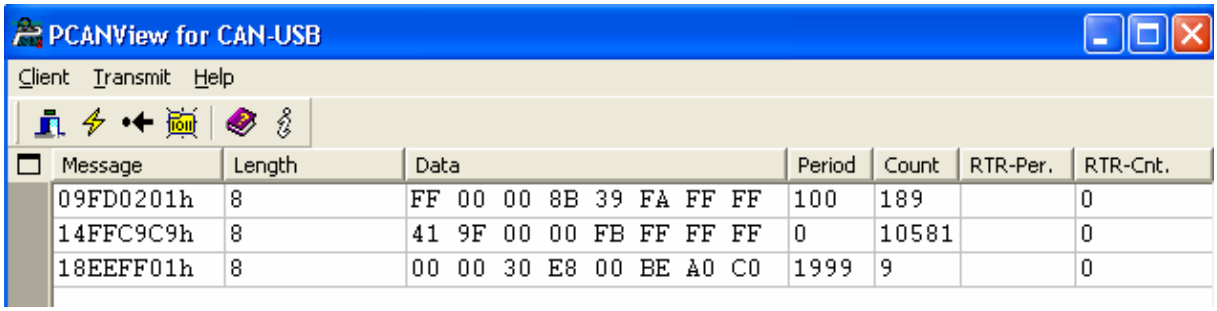
Bytes 2 & 3 (EEFFh)
This is the PGN number for 'Address Claim'
Byte 4 (C9h)
This is the 'source address' C9h
Data Frame
The products' unique 64-bit number

The Wind Instrument (display) is a data repeater type of instrument. It will only receive information that is being sent by the wind transducer and display it. It will generally only send out one PGN, that is address claim. Some of the other functions of the instrument may send out PGN's but this is not covered here.

Transmission frequency

In this example it shows that the Wind Instrument is sending NAME. If this was the only product on the bus from power-up, then this would be sending it regularly every 2 seconds. If two products were on the bus, the pinging would stop. If one product was then removed, the pinging would not restart.

IS12 Wind Transducer and Wind Instrument :



Message	Length	Data	Period	Count	RTR-Per.	RTR-Cnt.
09FD0201h	8	FF 00 00 8B 39 FA FF FF	100	189		0
14FFC9C9h	8	41 9F 00 00 FB FF FF FF	0	10581		0
18EEFF01h	8	00 00 30 E8 00 BE A0 C0	1999	9		0

It appears now that the Wind Transducer has been plugged back in. The Wind Instrument has started transmitting a 'Masthead Setting' PGN (14FFC9C9h), this is discussed below

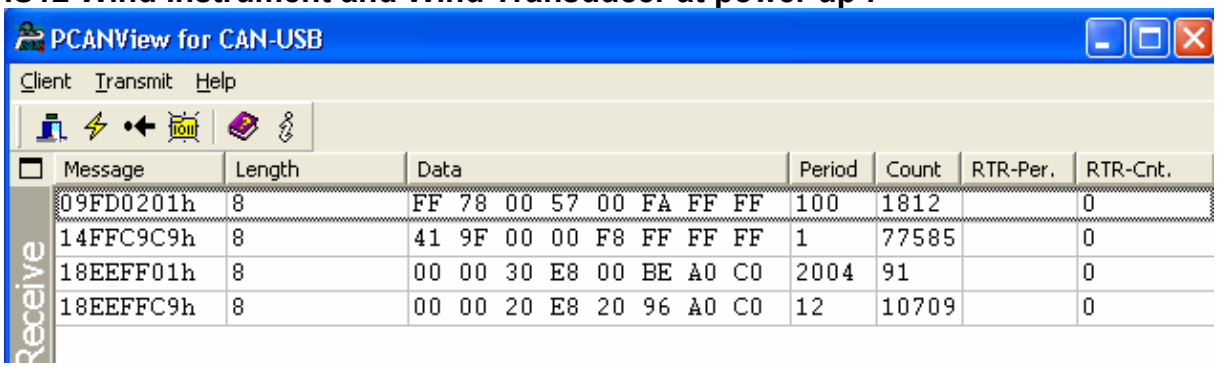
2. Sentence '14FFC9C9h' – Masthead Settings

Bytes 2 & 3 (FFC9h)
This is the PGN number for 'Masthead Settings'
Byte 4 (C9h)
This is the 'source address' C9h
Data Frame
Field 1 : Manufacturer ID (2 bytes) - Simrad Ltd manufacturing code - 9F41h
Field 2 : Wind Alarm limit (16 bit Unsi..) - Wind Speed alarm limit to a resolution of 0.01m/s
- (example shows 0000h, alarm not set)
Field 3 : Wind speed units (3 bit) - To define which units wind speed should be shown as
Field 4 : Reserved (all left-over bits) - Rest of the unused bits are set to '1's

Note : 'Masthead settings' is now an obsolete sentence on the current IS12 products range

Transmission frequency
In this example is it highly probable that the Masthead Settings PGN is sent once every 2 seconds

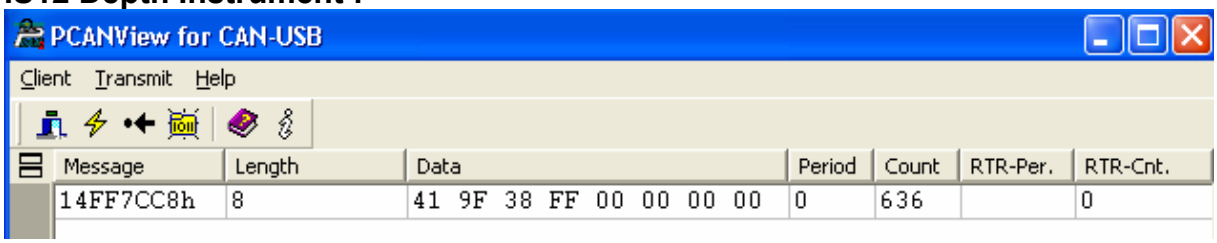
IS12 Wind Instrument and Wind Transducer at power up :



This is the same as 'Wind Instrument & Wind Transducer' except with the Address Claim/NAME for the Wind Instrument is shown

---

IS12 Depth Instrument :



1. Sentence '14FF7CC8h' – Supplimentary Depth Settings

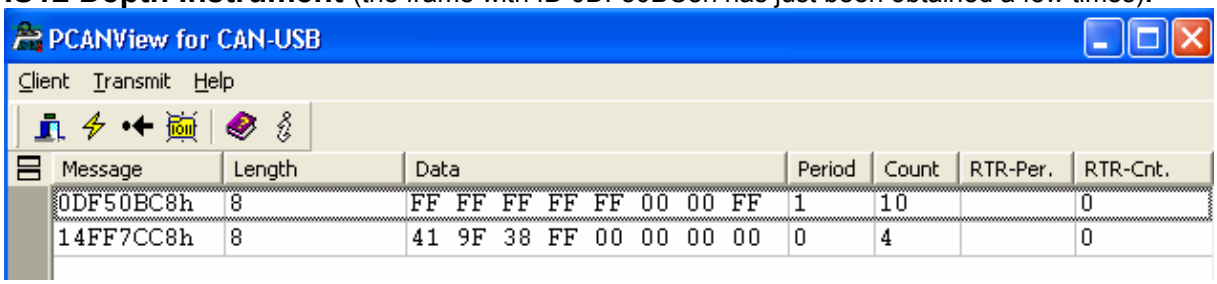
Bytes 2 & 3 (FF7Ch)
This is the PGN number for 'Supplementary Depth Settings'
Byte 4 (C8h)
This is the 'source address' C8h
Data Frame
Field 1 : Manufacturer ID (2 bytes) - Simrad Ltd manufacturing code - 9F41h
Field 2 : Depth Flags (8 bis) - for sending units, trend info & 'lost depth' alert
Field 3 : Signal Strength (1 byte) - Not used (set to FFh)
Field 4 : Deep Alarm Limit (16 bit unsi..) - Deep alarm limit to a resolution of 0.01 m
Field 5 : Shall Alarm Limit (16 bit unsi..) - Shall alarm limit to a resolution of 0.01 m

The actual depth value is not being transmitted, probably because the sensor is not connected or a reading has not been obtained, this would therefore give a display of ' - - - '

Transmission frequency
In this example, the Supplementary Depth Settings is sent once every two seconds

---

IS12 Depth Instrument (the frame with ID 0DF50BC8h has just been obtained a few times):



1. Sentence '0DF50BC8h' – Water Depth

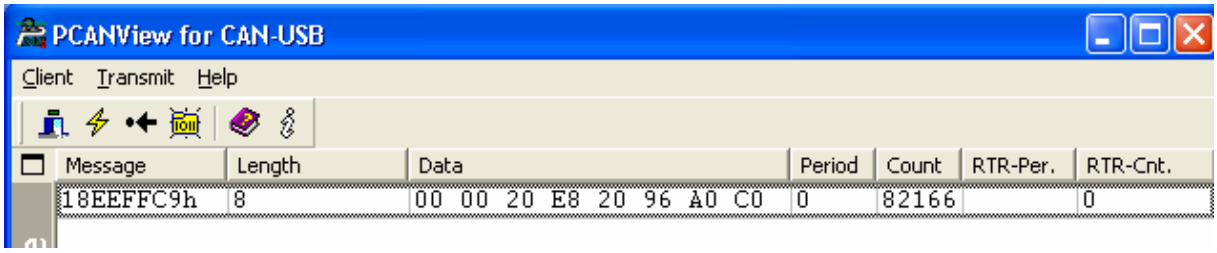
Bytes 2 & 3 (F50Bh)
This is the PGN number for 'Water Depth'
Byte 4 (C8h)
This is the 'source address' C8h
Data Frame

Field 1 : Sequence Identifier (1 byte)	- not used on this product, always set to FFh
Field 2 : Water Depth (32 bit unsigned)	- Water Depth to a resolution of 0.01m - (example shows 'Data not available' FFFFFFFFh)
Field 3 : Sensor Offset (16bit signed.)	- Position of Sensor relative to water line or keel - Resolution 0.001 meters
Field 4 : Reserved (all left-over bits)	- Rest of the unused bits are set to '1's

Transmission frequency
In this example, it will be likely that the water depth PGN is sent about 4 times a second

---

IS12 MEGA Instrument all alone :



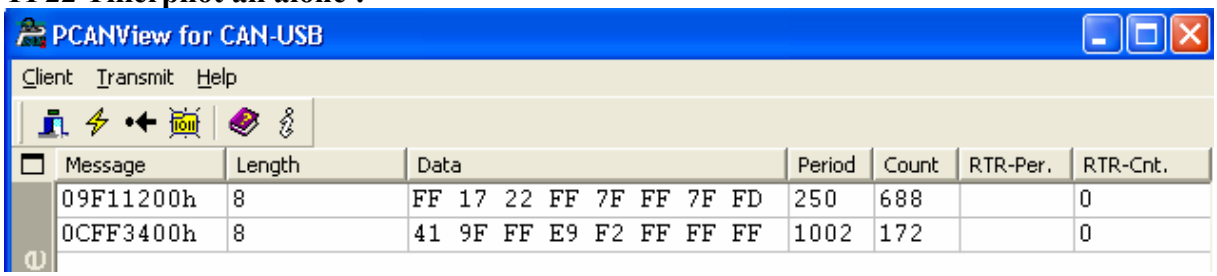
1. Sentence '18EEFFC9h' – Address Claim / NAME

Bytes 2 & 3 (EEFFh)
This is the PGN number for 'Address Claim'
Byte 4 (C9h)
This is the 'source address' C9h
Data Frame
The products' unique 64-bit number

The Mega Instrument (display) is a data repeater type of instrument. It will only receive information that is being sent by other data sources on the bus and display it. It will generally only send out one PGN, that is address claim.

Transmission frequency
In this example it shows that the Mega Instrument is sending NAME. If this was the only product on the bus from power-up, then this would be sending it regularly every 2 seconds. If two products were on the bus, the pinging would stop. If one product was then removed, the pinging would not restart.

TP22 Tillerpilot all alone :



1. Sentence '09F11200h' – Compass Heading

Bytes 2 & 3 (F112h)
This is the PGN number for 'Compass Heading'
Byte 4 (00h)
This is the 'source address' 00h
Data Frame

Field 1 : Sequence Identifier (1 byte) - not used on this product, always set to FFh
Field 2 : Compass angle (16 bit unsig..) - Compass angle to a resolution of 10E-4 rad.
- Range 0-2PiRad
Field 3 : Magnetic Dev. (16bit signed.) - Magnetic Deviation to a resolution of 10E-4 rad.
- Range 0-2PiRad
Field 3 : Magnetic Var. (16bit signed.) - Magnetic Variation to a resolution of 10E-4 rad.
- Range 0-2PiRad
Field 4 : Heading Ref (2 bits) - Reference (True /Magnetic etc)
- (example is set to 'magnetic')
Field 5 : Reserved (all left-over bits) - Rest of the unused bits are set to '1's

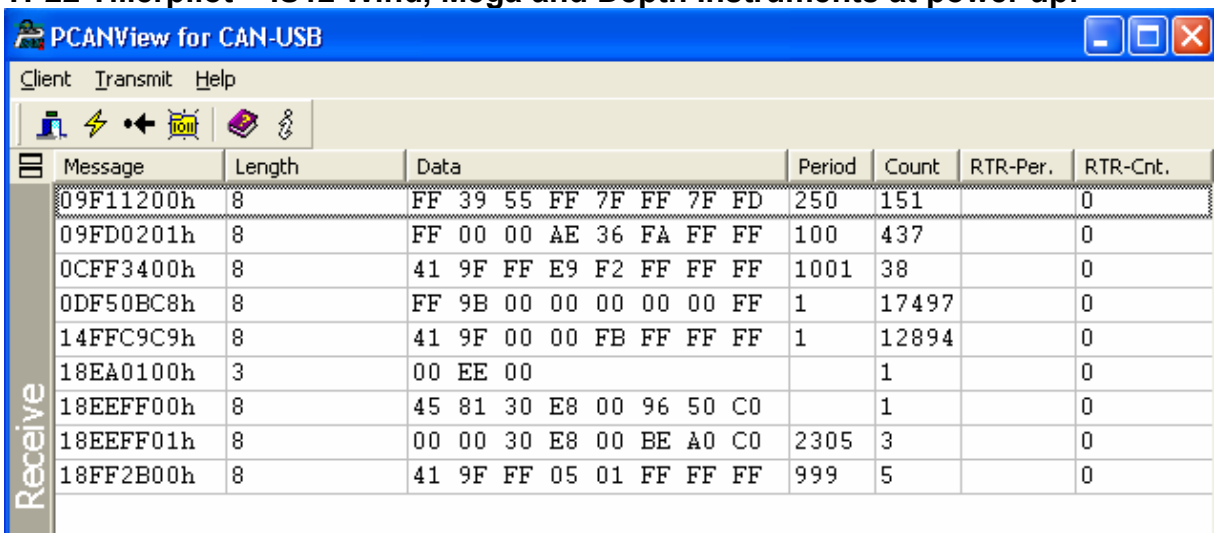
2. Sentence '0CFF3400h' – Remote Control

Bytes 2 & 3 (FF34h)
This is the PGN number for 'Remote Control'
Byte 4 (00h)
This is the 'source address' 00h
Data Frame

Field 1 : Manufacturer ID (2 bytes) - Simrad Ltd manufacturing code - 9F41h
Field 2 : Destination Address (1 byte) - Target Device Address
- (example shows FFh – pilot status message)
Field 3 : Remote Flags (8 bits) - Flags to select product, message type etc..
Field 4 : Keys / Lights (16 bits) - Shows which keys are pressed and lights are lit
Field 5 : Keys held (16 bits) - Shows which keys are being held
- (not used on this product so is set to FFFFh)

Transmission frequency
In this example it shows that the Compass heading is sent once every 250ms (4 times per second) and the remote control PGN sent once per second

TP22 Tillerpilot + IS12 Wind, Mega and Depth Instruments at power up:



Message	Length	Data	Period	Count	RTR-Per.	RTR-Cnt.
09F11200h	8	FF 39 55 FF 7F FF 7F FD	250	151		0
09FD0201h	8	FF 00 00 AE 36 FA FF FF	100	437		0
0CFF3400h	8	41 9F FF E9 F2 FF FF FF	1001	38		0
0DF50BC8h	8	FF 9B 00 00 00 00 00 FF	1	17497		0
14FFC9C9h	8	41 9F 00 00 FB FF FF FF	1	12894		0
18EA0100h	3	00 EE 00		1		0
18EEFF00h	8	45 81 30 E8 00 96 50 C0		1		0
18EEFF01h	8	00 00 30 E8 00 BE A0 C0	2305	3		0
18FF2B00h	8	41 9F FF 05 01 FF FF FF	999	5		0

The only new Sentence that appears in the example above is '18EA0100h'. This PGN is used to request another product/node on the bus to send a particular PGN. As long as the receiving node can send it, it should, but only once per request.

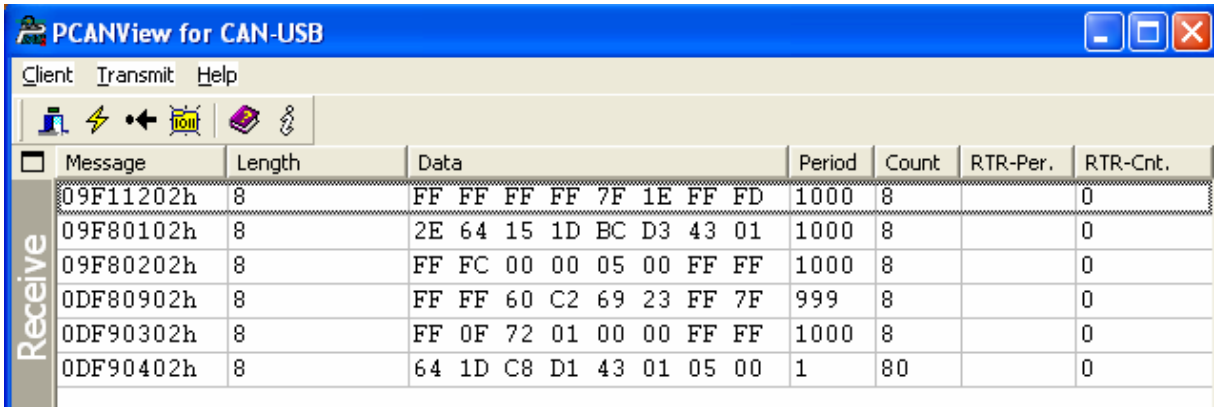
1. Sentence '18EA0100h' – ISO Request

Byte 2 (EAxxh)
This is the PGN number for 'ISO Request'
Byte 3 (01h)
This is the 'destination/target address' 01h
Byte 4 (00h)
This is the 'source address' 00h
Data Frame
Field 1 : PGN Requested (3 bytes) - The PGN that is required from the destination node
- (this example shows 00EE00h – which is NAME)

This is a short frame, there is only 3 byte in the data frame and then the frame ends.

As you can see from the example above, Node 00 requested NAME from Node 01, and Node 01 replied with its NAME (18EEFF00h). The reply that 01 made was a global one (where everyone can listen), but the original ISO request was specific (only destined for the one address, 01)

CP33 GPS all alone :



Message	Length	Data	Period	Count	RTR-Per.	RTR-Cnt.
09F11202h	8	FF FF FF FF 7F 1E FF FD	1000	8		0
09F80102h	8	2E 64 15 1D BC D3 43 01	1000	8		0
09F80202h	8	FF FC 00 00 05 00 FF FF	1000	8		0
0DF80902h	8	FF FF 60 C2 69 23 FF 7F	999	8		0
0DF90302h	8	FF 0F 72 01 00 00 FF FF	1000	8		0
0DF90402h	8	64 1D C8 D1 43 01 05 00	1	80		0

1. Sentence '09F11202h' – Compass Heading
This has been covered in a previous example

2. Sentence '09F80102h' – GPS Position Rapid Update
Data Frame
Field 1 : Latitude (32 bit signed) - Latitude of current position to a resolution of 1.1cm.
- Range -90 to +90 degrees
Field 2 : Longitude (32 bit signed) - Longitude of current position to a resolution of 1.1cm.
- Range -180 to +180 degrees

3. Sentence '09F80202h' – COG / SOG Rapid Update
Data Frame
Field 1 : Sequence Identifier (1 byte) - not used on this product, always set to FFh
Field 2 : Ref Flags (2 bits) - Defining true / magnetic COG
Field 3 : Reserved (6 bits) - Rest of the unused bits are set to '1's
Field 4 : COG (16bit signed.) - Course over ground to a resolution of 10E-4 rad.
- Range 0-2PiRad
Field 5 : SOG (16bit unsigned.) - Speed over ground to a resolution of 0.01 m/s
Field 6 : Reserved (all left-over bits) - Rest of the unused bits are set to '1's

4. Sentence '09F80202h' – UTC Date & Time
Data Frame
Field 1 : Date (16 bit unsigned) - Days since Jan1,1970, resolution of 1 day
Field 2 : Time (32 bit unsigned) - Time of Day, resolution 1x10E-4s, range 0 to 86,401s
Field 3 : Local Offset (16 bit signed) - Local time zone offset in minutes, resolution of 1 minute

5. Sentence '0DF90302h' – Cross Track Error (XTE)
Data Frame
Field 1 : Sequence Identifier (1 byte) - not used on this product, always set to FFh
Field 2 : Ref Flags (6 bits) - Definines XTE mode, direction ref (mag/true) etc
Field 3 : Reserved (2 bits) - Rest of the unused bits are set to '1's
Field 4 : XTE (32bit signed.) - Cross Track Error to a resolution of 0.01m
Field 5 : Reserved (all left-over bits) - Rest of the unused bits are set to '1's

6. Sentence '0DF90402h' – Navigation Data
This is a highly complex PGN, which uses 19 separate consecutive frames to get all the data across. Each frame is still only 8 bytes long. It is difficult to read the frames on the PCanView above as they will be layered one on top of another. Further more, it is unknown at what interval the combination of these frames will be sent.

Transmission frequency
Sentences 1-5 (above) are transmitted regularily at 1.0 second intervals. It is unknown at what frequency the Navigation Data is sent