https://copperhilltech.com

NMEA 2000/SAE J1939 Scanner – WCOM Protocol

The NMEA 2000 / SAE J1939 series of data scanners transmits CAN Bus data per Bluetooth or BLE (Bluetooth Low Energy) for display on a phone, tablet, or computer.

The NMEA 2000 / SAE J1939 Scanner firmware uses an ASCII protocol (in the following referred to as *WCOM – Wireless Communication*) as described below, i.e., all data is transmitted using Hex-ASCII bytes for better readability when using a terminal software.

The WCOM protocol is designed primarily for data monitoring (e.g., for Electronic Logging Devices – ELD) with focus on minimum wireless communication. If your application requires the simulation of SAE J1939 data traffic, have a look at SAE J1939 to Bluetooth Gateway Module - Copperhill (copperhilltech.com).

The following refers to the *scanner* and *host*, where the scanner can be either, SAE J1939 or NMEA 2000. The host is typically an iOS/Android phone/tablet but can also be a PC. Copperhill Technologies provides a free-of-charge Windows software to program application parameters, such as:

- PGN Filters
- CAN Baud Rate Mode (250k/500k/Automatic)
- Device ID
- UUIDs (BLE only)

These parameters are permanently stored within the scanner module but can be overwritten at any time.

WCOM Protocol

The WCOM Protocol comprises of a set of wireless messages distinguished by a token that is placed at the beginning of each data frame.

The WCOM protocol operates in two modes:

1. Operational (marked in blue below)
   Operational mode is for exchange of data, such as PGNs and CAN Bus baud rate detection (if activated).

2. Setup (marked in green below)
   Setup mode supports the setting of operational parameters, such as baud rate mode, PGN filters, and more.

The messages and tokens are:

		Source		Message Length	
Token	**Description**	**Host**	**Scanner**	**Bluetooth**	**BLE**
“$”	PGN Data Frame	x	x	30	15 + 20
“.”	CAN Baud Rate Detection		x	4	4
“#”	CAN Baud Rate Info		x	4	4
“@”	Setup Mode Start/Stop	x		6	6
“R”	CAN Baud Rate Mode	x		4	4
“I”	Device ID	x		20	20
“U”	UUIDs (BLE Only)	x		-	17 + 17 + 17
“P”	PGN Filter	x		19	19
“?”	Error Detection		x	4	4

Notes:

- All tokens have been assigned outside the ranges of “0” to “9” and “A” to “F”, which are assigned to data only.
- All messages are at least 4 bytes long to allow sufficient verification.
- Constant messages, i.e., those messages whose content will not change, do not require a checksum.

Error Detection

Each message is terminated by a two-character checksum. A checksum is a technique to check data for transmission errors or tampering.

The checksum is a 2’s complement checksum over the length of the message but not including the checksum itself. The checksum is always 8 bits long (0 to 255) and is converted into a two-character string.

A one’s complement is just a complement: ~x
A two’s complement is a complement and increment, ignoring carry: (~x)+1

The scanner will ignore faulty messages but will send a “?ERR” string to report an error.

Data Frame: Total Length = 4 bytes

Byte Position	Length	Description
0	1	“?” Token
1…3	3	“ERR”

Notes:

- The “?” token applies to both, operational and setup mode.
- The “?” token works only in one direction, i.e., from scanner to host.
- No checksum required.

PGN Data Frame

The PGN Data Frame is a bi-directional message frame, i.e., it is being transmitted in both directions, from scanner to host as well as from host to scanner.

The scanner transmits all received PGNs per default, unless PGN filters are active, which reduces the wireless data traffic.

The host should transmit only occasional protocol PGNs when necessary, such as *Message Request*. A constant data stream from host to scanner is supported but not recommended. It would not only unnecessarily increase data traffic but also defeat the purpose as a scanning device.

The PGN Data Frame format differs between Bluetooth and BLE, since BLE is limited to 20 bytes per data frame.

Bluetooth Data Frame

Data Frame: Total Length = 29 bytes

Byte Position	Length	Description
0	1	“$” Token
1	1	Data Page (DP) = 0/1
2…5	4	PGN
6	1	Priority
7…8	2	Destination Address (DA)
9…10	2	Source Address (SA)
11	1	DLC = Data Length Code
12…27	16	0…8 data bytes; unused data = 0x00
28…29	2	Checksum

BLE Data Frame

BLE data frames are limited to 20 bytes, thus we use segmentation.

Data Frame 1: Total Length = 15 bytes

Byte Position	Length	Description
0	1	“$” Token
1	1	“1” indicates frame #1
1	1	Data Page (DP) = 0/1
2…5	4	PGN
6	1	Priority
7…8	2	Destination Address (DA)
9…10	2	Source Address (SA)
11	1	DLC = Data Length Code
12…13	2	Checksum

Data Frame 2: Total Length = 20 bytes

Byte Position	Length	Description
0	1	“$” Token
1	1	“2” indicates frame #2
2…17	16	0…8 data bytes; unused data = 0x00
18…19	2	Checksum

CAN Baud Rate Detection & Info

The scanner supports automatic CAN Bus baud rate detection according to SAE J1939-16 Automatic Baud Rate Detection Process. However, the default setting is 250 kbps, while 500 kbps and automatic baud rate detection must be set through the host (PC) connection. The baud rate mode is stored permanently and will be used after each power-up cycle.

According to SAE J1939-16, the device must stay in detection mode until it detects a valid CAN Bus data frame. The time required to detect a valid data frame depends primarily on the SAE J1939 busload, i.e., higher busload will result in faster detection times.

The scanner transmits a “.BRD” string every 500 msec to indicate the baud rate detection process. As soon as it detects a valid data frame it reports the result to the host by sending the “#” token followed by the detected baud rate (250/500).

Data Frame during detection process: Total Length = 4 bytes

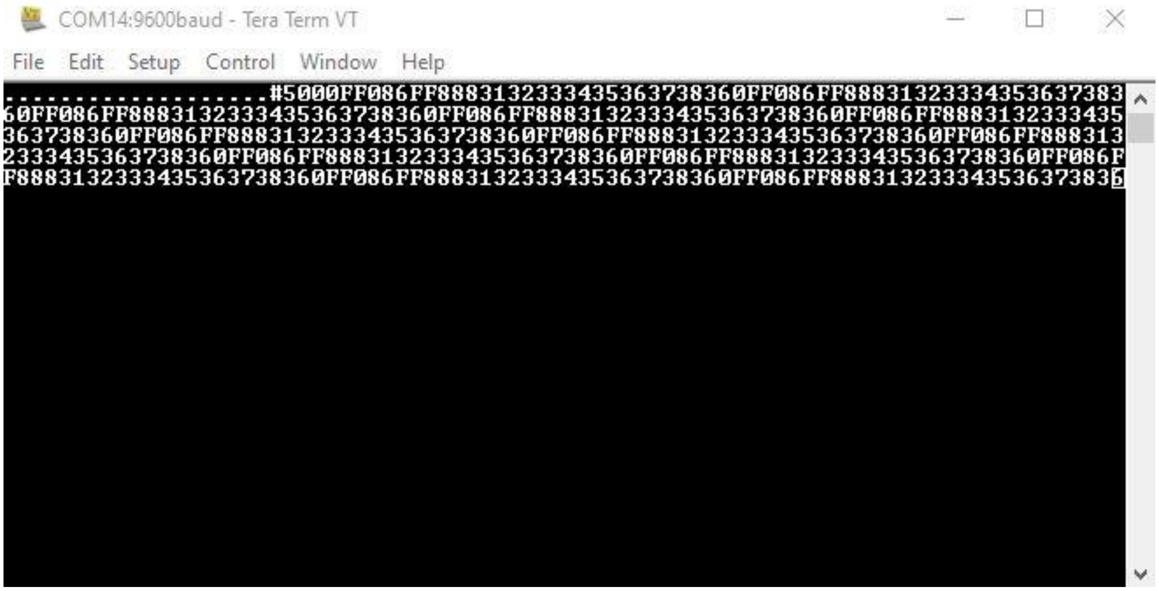
Byte Position	Length	Description

0	1	“.” Token
1…3	3	“BRD” (Baud Rate Detection)

Data Frame after successful detection process: Total Length = 4 bytes

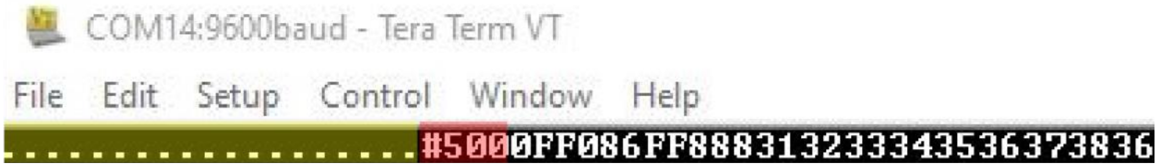
Byte Position	Length	Description
0	1	“#” Token
1…3	3	“250” / “500”

The following image shows the sequence as recorded under a Windows PC running a terminal software. We purposely connected to a “quiet” network to demonstrate the functionality and initiated data traffic a few seconds later.



```
...................#5000FF086FF888313233343536373836 0FF086FF88831323334353637383
60FF086FF888313233343536373836 0FF086FF888313233343536373836 0FF086FF8883132333435
363738360FF086FF888313233343536373836 0FF086FF888313233343536373836 0FF086FF888313
233343536373836 0FF086FF888313233343536373836 0FF086FF888313233343536373836 0FF086F
F888313233343536373836 0FF086FF888313233343536373836 0FF086FF888313233343536373836
```

In more detail:



```
...................#5000FF086FF888313233343536373836
```

- The yellow section indicates the baud rate detection process.
- In the red section, the scanner reports a detected baud rate of 500 kbps.
- This is followed by a PGN data frame:
  - Data Page = 0
  - PGN = FF08
  - Priority = 6
  - Destination Address (DA) = FF
  - Source Address (SA) = 88

  - Data bytes = 8
  - Data = 31 32 33 34 35 36 37 38
  - Checksum = 36

Notes:

- The above image shows the data frame format for a Bluetooth connection. However, regarding the baud rate detection, the sequence will be the same when using BLE.
- If you connect the scanner to a running network and have automatic baud rate detection activated, you might not see the "." token. The scanner uses a 10 msec scan time, and it may take that short of a time to detect the baud rate.
- No checksum required.

Setup Mode Start/Stop

The setup string "@SETUP", received during operation mode, will switch the scanner module into setup mode.

A second "@SETUP" string, received during setup mode, will reset the scanner module, and cause it to operate using the newly assigned parameters.

Data Frame during detection process: Total Length = 6 bytes

Byte Position	**Length**	**Description**
0	1	"@" Token
1…5	5	"SETUP"

Notes:

- During **setup mode**, the scanner will not transmit any PGNs to the host, and it will ignore any PGN transmissions from the host.
- During **operational mode**, the scanner will ignore any tokens other than "$" or "@".
- No checksum required.

CAN Baud Rate Mode

The "R" token sets the CAN Baud Rate Mode, which is stored permanently in the scanner module (but can be overwritten at any time afterwards).

The scanner will use 250 kbps per default unless it is overwritten by the "R" token.

There are three modes:

Mode	**Host Transmission**

250 kbps	“R250”
500 kbps	“R500”
Auto Detection	“RAUT”

Data Frame during detection process: Total Length = 4 bytes

Byte Position	**Length**	**Description**
0	1	“R” Token
1…3	3	“250” / “500” / “AUT”

Notes:

- No checksum required.

Device ID

Each Bluetooth or BLE device should support a unique Device ID to clearly identify them within a network. Both, the SAE J1939 and NMEA 2000 scanners, come with a default Device ID that can be modified by the end user.

Scanner	**Wireless Technology**	**Default Device ID**
SAE J1939	Bluetooth	“J1939-BT-1001”
SAE J1939	BLE	“J1939-BLE-1001”
NMEA 2000	Bluetooth	“N2K-BT-1001”
NMEA 2000	BLE	“N2K-BLE-1001”

Data Frame: Total Length = 20 bytes

Byte Position	**Length**	**Description**
0	1	“I” Token
1…2	2	Device ID String Length
3…17	15	Device ID String
18…19	2	Checksum

To comply with BLE data frame length limitations, the data frame is limited to 20 characters with a maximum of 15 characters used for the Device ID. Any unused characters (i.e., those beyond the reported string length) are filled with blank characters.

UUIDs (BLE Only)

Each BLE device must support a set of three UUIDs, Service UUID, Receive UUID, and Transmit UUID. Each UUID is 36 characters long. The defaults are:

UUID	**Setting**
Service	"6E400001-B5A3-F393-E0A9-E50E24DCCA9E"

Receive	"6E400002-B5A3-F393-E0A9-E50E24DCCA9E"
Transmit	"6E400003-B5A3-F393-E0A9-E50E24DCCA9E"

There is no mandatory need to modify these settings.

BLE data frames are limited to 20 bytes, thus we use segmentation, i.e., three segments per 12 UUID characters.

Data Frame 1: Total Length = 17 bytes

Byte Position	Length	Description
0	1	“U” Token
1	1	Service = “S”, “R”, “T”
2	1	Segment Number: “1”…”3”
3…14	12	UUID characters
15…16	2	Checksum

Notes:

- The UUID will be stored when all three data frames have been received.

PGN Filter

The data frame length must be constant (18 bytes), thus allowing to set up to 3 Filter PGNs per data frame. All unused bytes should be filled with “0”. The data frame can be transmitted multiple times to continuously add PGNs to the filter array. There is a maximum of 80 filter entries. Any entry above 80 will be ignored.

All current filters will be deleted when the number of PGN filters is transmitted as zero.

Data Frame: Total Length = 19 bytes

Byte Position	Length	Description
0	1	“F” Token
1	1	Number of Filter PGNs = 1…3
2	1	Data Page (DP) = 0/1
3…6	4	PGN
7	1	Data Page (DP) = 0/1
8…11	4	PGN
12	1	Data Page (DP) = 0/1
13…16	4	PGN
17…18	2	Checksum