Navigation sensor data exchange format
Version 0.0

© 2020 Fedor Baklanov, Vishwanath Malipatil. All rights reserved.

Table of contents

1 General information 3

1.1 About format 3

1.2 Copyright statement 3

1.3 List of modifications 3

2 Protocol description 3

2.1 Byte order (endianness) 3

2.2 Frame format 3

2.3 Supported CRC types and checksums 3

2.4 CRC/checksum computation 4

3 Description of messages 4

3.1 Service message group (0x00) 4

3.1.1 Protocol version (0x00) 4

3.2 Sensor message group (0x01) 4

3.3 Camera message group (0x03) 4

3.3.1 Image in JPEG format (0x00) 4

3.4 GNSS message group (0x02) 4

3.4.1 GNSS raw measurements (0x00) 4

3.4.2 GNSS navigation message (0x01) 5

3.4.3 GNSS and space vehicle numbering 5

4 Protocol versions 5

1 General information

1.1 About format

Navigation sensor data exchange format is an open format/protocol for measurement data exchange and/or transmission. Any person or organization is granted rights, free of charge, to use this format/protocol for commercial and non-commercial purposes.

Readers are welcome to propose improvements and to contribute to further development of the format. You may get in touch with the authors by sending an email to contact@navigation-expert.com.

1.2 Copyright statement

This document and all the information contained in this document are the property of Fedor Baklanov and Vishwanath Malipatil. This document may not be modified or altered. Any person or organization is granted the right to copy, republish, and redistribute this document on a non-profit basis and in each case identifying https://navigation-expert.com as the original source of the document.

1.3 List of modifications

Date	Author	Change summary
22.05.2020	Fedor Baklanov	Initial version of the document

2 Protocol description

2.1 Byte order (endianness)

Least significant byte first, i. e. little-endian.

2.2 Frame format

Byte position	Data type	Description
1	uint8	Message marker 1, value 0x4E (ASCII 'N')
2	uint8	Message marker 2, value 0x45 (ASCII 'E')
3	uint8	Message group
4	uint8	Message type
5	uint8	CRC/checksum type
6 to 9	uint32	Length of payload in bytes (LEN)
10 to (9 + LEN)	uint8[LEN]	Payload
(LEN + 10) to (LEN + 10 + sizeof(CRC))	uint8[sizeof(CRC)]	CRC/checksum (can be empty, i. e. have zero length if CRC type is CRC_NONE, binary value 0x00)

2.3 Supported CRC types and checksums

CRC/checksum type	8-bit binary code	Description
CRC_NONE	0x00	Used when a message is transmitted without a checksum/CRC
CRC_24Q	0x01	24-bit cyclic redundancy check used by Qualcomm, SBAS, and others. Division by a polynomial 0x1864CFB.

CKS_FLET16	0x02	Fletcher's 16-bit checksum

2.4 CRC/checksum computation

Bytes 1 through (9 + length of payload) shall be used in CRC/checksum computation. A message may be transmitted without a checksum. In such a case CRC/checksum type shall be set to CRC_NONE.

3 Description of messages

3.1 Service message group (0x00)

3.1.1 Protocol version (0x00)

Byte position	Data type	Description
1	uint16	Protocol version, see Section 4

This message SHALL be present in recorded file at least once.

3.2 Sensor message group (0x01)

Tbd

3.3 Camera message group (0x03)

3.3.1 Image in JPEG format (0x00)

Byte position	Data type	Description
1 to 8	uint64	Image timestamp in nanoseconds
9 to (8 + ImageLen)	uint8[ImageLen]	Image in JPEG format. Image length (ImageLen) can be derived from payload length as (Length of payload - 8)

3.4 GNSS message group (0x02)

3.4.1 GNSS raw measurements (0x00)

Byte position	Data type	Description
1	uint8	Number of measurements (MeasNum)
2 to 9	uint64	GNSS receiver clock time in nanoseconds
10 to 17	uint64	Microcontroller clock time in nanoseconds
18	bitfield: uint8	Bit 0: availability of GNSS receiver clock time, 1 if available, 0 otherwise Bit 1: availability of microcontroller clock time, 1 if available, 0 otherwise Bit 2: GNSS receiver clock reset or realignment flag, 1 if clock were reset/realigned this epoch, 0 otherwise Bit 3: microcontroller clock reset or realignment flag, 1 if clock were reset/realigned this epoch, 0 otherwise Bits 4 to 7: reserved
Repeating blocks		
Position in block		
1	uint8	GNSS ID, see Section 3.4.3
2	uint8	Space vehicle ID, see Section 3.4.3

3 to 10	float64	Pseudorange in meters
11 to 18	float64	Pseudorange rate in meters per second
19 to 26	float64	Carrier phase in cycles
27 to 30	float32	Pseudorange accuracy (1 sigma) in meters
31 to 34	float32	Pseudorange rate accuracy (1 sigma) in meters per second
35 to 38	float32	Carrier phase accuracy (1 sigma) in cycles
39	uint8	C/N0 value in dB-Hz
40 to 43	float32	Carrier frequency in Hz
44	bitfield: uint8	Bit 0: pseudorange validity, 1 if valid, 0 otherwise. Pseudorange accuracy is valid only if this bit is 1. Bit 1: pseudorange rate validity, 1 if valid, 0 otherwise. Pseudorange rate accuracy is valid only if this bit is 1. Bit 2: carrier phase validity, 1 if valid, 0 otherwise. Carrier phase accuracy is valid only if this bit is 1. Bit 3: availability of carrier frequency, 1 if available, 0 otherwise. Bits 4 to 7: reserved

3.4.2 GNSS navigation message (0x01)

Byte position	Data type	Description
1	uint8	Satellite vehicle ID
2	uint8	Navigation message type
3 to (MsgLen + 2)	uint8[MsgLen]	Navigation message. Length of the message (MsgLen) depends on navigation message type.

3.4.3 GNSS and space vehicle numbering

GNSS identifier	Satellite navigation system
0	Unknown
1	GPS
2	SBAS
3	GLONASS
4	QZSS
5	Beidou
6	Galileo

Space vehicle IDs:

- GPS: 1 to 32
- SBAS: tbd
- GLONASS: 1 to 24
- QZSS: tbd
- Beidou: 1 to 37
- Galileo: 1 to 36
- Invalid/unsupported space vehicle: 0

4 Protocol versions

Protocol version	16-bit integer identifier

Version 0.0	0x0000