

LPMS-IG1 Series

User Manual ver. 1.2



LPMS-IG1-CAN



LPMS-IG1P-CAN



LPMS-IG1-RS232



LPMS-IG1P-RS232

Version History

Date	Version	Details
2019-05-23	Ver. 1.0	First draft
2019-08-28	Ver. 1.1	English version first draft
2019-09-05	Ver. 1.2	<ul style="list-style-type: none">- Rename document from reference manual to user manual.- Compatible with firmware IG1-3.0.3-20190508 or above.- Remove the hardware contents to hardware manual

Table of Contents

1. INTRODUCTION	- 1 -
2. REFERENCE SYSTEMS	- 2 -
2.1 COORDINATES	- 2 -
2.2 EULER ANGLES	- 2 -
2.3 ORIENTATION ALIGNMENT MODES	- 3 -
<i>Heading Reset</i>	- 3 -
<i>Alignment Reset</i>	- 3 -
<i>Object Reset</i>	- 4 -
3. COMMUNICATION	- 5 -
3.1 COMMUNICATION MODES	- 5 -
3.2 LPBUS PROTOCOL	- 6 -
<i>GET Commands</i>	- 6 -
<i>SET Commands</i>	- 6 -
3.3 LPBUS PACKET FORMAT	- 6 -
<i>Data Format in a Packet Data Field</i>	- 8 -
<i>Sensor Measurement Data</i>	- 8 -
3.4 LPBUS EXAMPLE COMMUNICATION	- 12 -
<i>Goto Command Mode</i>	- 12 -
<i>Goto Steaming Mode</i>	- 13 -
<i>Request Gyroscope Range</i>	- 15 -
<i>Set Accelerometer Range</i>	- 16 -
<i>Save Sensor Parameters</i>	- 17 -
<i>Read Sensor Status</i>	- 18 -
<i>Set UART / RS232 Baudrate</i>	- 19 -
3.5 CANOPEN AND SEQUENTIAL CAN PROTOCOLS	- 20 -
3.6 CAN DATA OUTPUT FORMAT	- 20 -
3.7 CAN MAPPING	- 21 -
3.8 ASCII OUTPUT	- 23 -
4. APPENDIX	- 24 -
4.1 FIRMWARE FUNCTION / COMMAND LIST	- 24 -
<i>Summary</i>	- 24 -

<i>Acknowledged and Not-acknowledged Identifiers</i>	- 27 -
<i>Register Value Save and Reset Command</i>	- 27 -
<i>Mode Switching Commands</i>	- 28 -
<i>Sensor Status Command</i>	- 28 -
<i>Get Data Commands</i>	- 28 -
<i>Device Info Commands</i>	- 29 -
<i>Data Transmission Commands</i>	- 29 -
<i>IMU ID Setting Command</i>	- 31 -
<i>Stream Frequency Commands</i>	- 31 -
<i>Deg/Rad Output Commands</i>	- 32 -
<i>Reference Setting and Offset Reset Command</i>	- 32 -
<i>Accelerometer Settings Command</i>	- 33 -
<i>Gyroscope Settings Command</i>	- 33 -
<i>Magnetometer Settings Command</i>	- 34 -
<i>Filter Settings Command</i>	- 36 -
<i>CAN Settings Command</i>	- 36 -
<i>UART / RS232 Settings Command</i>	- 39 -
<i>Sensor Data Timestamp Manipulation</i>	- 41 -
<i>GPS Data Transmission Commands</i>	- 42 -





1. Introduction

LPMS-IG1 series sensors are high precision multi-purpose inertial measurement units specially developed for industrial applications. LPMS-IG1 sensors embed a powerful processor, 3 high precision single axis gyroscopes, a general purpose 3 axis gyroscope, accelerometer and magnetometer. The unique dual-gyroscope (Gyro I & II) setup enables accurate dynamics measurement in both low and high-speed applications. Gyro I is suitable for applications where the accuracy requirements are high and the detection range is not critical (<400dps). Gyro II is suitable for general applications where expected measurement range exceeds 400dps.

LPMS-IG1 series include two models, LPMS-IG1 (without GPS) and LPMS-IG1P (with GPS). Both models offer the following communication methods: USB+RS232 or USB+CAN. For details on how to use the communication interface, please refer to the relevant sections below. LPMS-IG1 series come in a waterproof metal housing with a flat bottom profile and mounting holes for easy installation.

Main features:

- High precision and stability
- Dual 3-axis general purpose gyroscope, 3-axis accelerometer, 3-axis magnetometer
- GyroI: High precision 3-axis gyroscope
- GyroII: 3-axis general purpose gyroscope
- Realtime output: Raw and calibrated sensor data, quaternion, Euler angles, temperature, GPS (LPMS-IG1P)
- Communication method : USB+RS232 or USB+CAN

Applications:

- Robot navigation
- Automotive navigation
- Remote control and monitoring for industrial robots
- Automated guided vehicle navigation



2. Reference Systems

2.1 Coordinates

The LPMS sensor calculates the orientation difference between a fixed sensor coordinate system (**S**) and a global reference coordinate system (**G**). Both coordinate systems are defined as right-handed Cartesian coordinate systems. The sensor coordinate system (**S**) can be referred to the sticker information on the sensor enclosure.

The global reference coordinate system (**G**) can be divided into two different cases. While the orientation calculation is using all acceleration, gyroscope and magnetic data (sensor filter mode set at acc+gyr+mag), (**G**) system is defined as following:

- X positive when pointing to the magnetic north
- Y positive when pointing to the magnetic west
- Z positive when pointing up (gravity points vertically down with -1g)

While the orientation calculation is using only acceleration and gyroscope data (sensor filter mode set at acc+gyr), (**G**) system is defined as following:

- X positive aligned to ground plane horizontal projection of x axis of (**S**) when sensor powered on
- Y positive based on right-handed Cartesian coordinate definition
- Z positive when pointing up (gravity points vertically down with -1g)

2.2 Euler Angles

A positive rotation is always right-handed, i.e. defined according to the right-hand rule (corkscrew rule). This means a positive rotation is defined as clockwise in the direction of the axis of rotation.

The definition used for Euler angles in this document is equivalent to roll, pitch, yaw/heading. The Euler angles are of **ZYX global type** (subsequent rotation around global Z, Y and X axis, also known as aerospace sequence):

- Roll: Rotation around global X, defined from -180° ~ 180°
- Pitch: Rotation around Y, defined from -90° ~ 90°
- Yaw: Rotation around Z, defined from -180° ~ 180°

NOTE: Due to the definition of Euler angles there is a mathematical singularity



when the sensor-fixed X-axis is pointing up or down in the global reference frame (i.e. pitch approaches ± 90). This singularity is not present in quaternion output.

2.3 Orientation Alignment Modes

Heading Reset

Often it is important that the global Z-axis remains along the vertical (defined by local gravity vector), but the global X-axis has to be pointed in a particular direction on horizontal plane. In this case a heading reset may be used. When performing a heading reset, the new global reference frame is chosen such that the global X-axis points in the direction of the sensor's x direction while keeping the global Z-axis vertical (along gravity, pointing upwards). In other words: The global Z-axis point upwards along gravity, where the X and Y axis orthogonally form a perpendicular plane.

The operation of heading reset comes out a result on Euler angles data is that yaw angle becomes zero, roll and pitch angles keep same.

NOTE: After a heading reset, the yaw may not be exactly zero, this occurs especially when the X-axis is close to the vertical. This is caused by the definition of the yaw when using Euler angles, which becomes unstable when the pitch approaches ± 90 deg.

Alignment Reset

The alignment reset function aims to facilitate in aligning the sensor coordinate system (**S**) with the ground plane. After an alignment reset, the (**S**) coordinate system is changed to (**S'**) as follows:

- The (**S'**) Z-axis is the vertical (up) at time of reset
- The (**S'**) X-axis equals the (**S**) X-axis but projected on the new horizontal plane.
- The (**S'**) Y-axis is chosen as to obtain a right-handed coordinate frame.

The operation of alignment reset comes out a result on Euler angles data is that yaw angle keeps same, roll and pitch angles become zero.

NOTE: Once this alignment reset is done, orientation will be output in the new

coordinate frame (S').

Object Reset

The object reset aligns the sensor coordinate frame to that of the object to which it is attached (see Fig. 1). The sensor must be attached in such a way that the X-axis is in the XZ-plane of the object coordinate frame, i.e. the sensor can be used to identify the X-axis of the object. To preserve the global vertical, the object must be oriented such that the object Z-axis is vertical. The object reset causes the new (S') coordinate frame and the object coordinate frame (O) to be aligned.

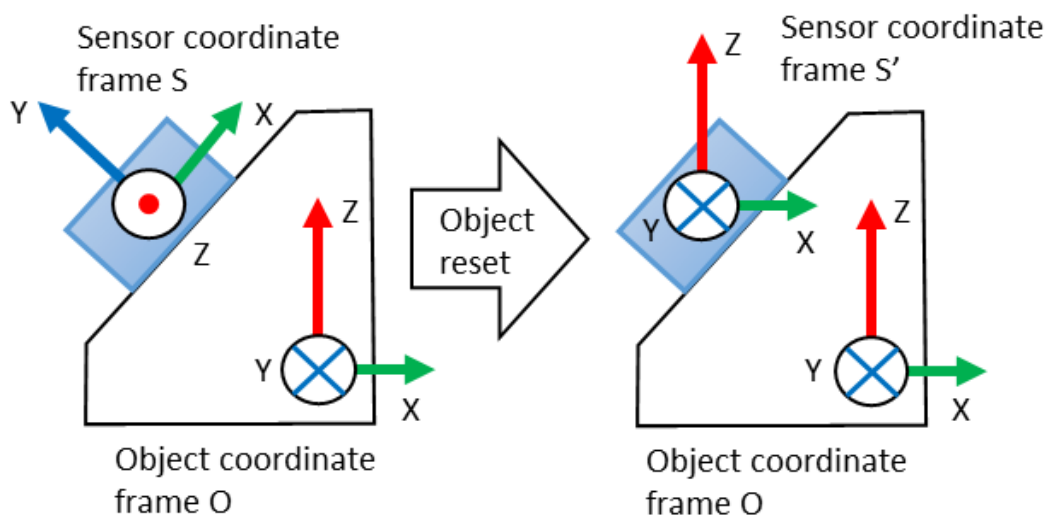


Fig. 1. Illustration of object reset

The object reset simply combines alignment reset and the heading reset at a single instant in time. This has the advantage that all coordinate systems can be aligned with a single action. Keep in mind that the new global reference X-axis (heading) is defined by the object X-axis (to which XZ-plane you have aligned the sensor).

NOTE: Since the sensor X-axis is used to describe the direction of the object X-axis, the reset will not work if the sensor X-axis is aligned along the Z-axis of the object.

3. Communication

LPMS-IG1 series offer two types of communication methods:

- USB+RS232
- USB+CAN

Sensor data is streamed to both USB and RS232/CAN terminals simultaneously. Communication protocol for different terminals are summarized as below:

Table 1. Protocols for different terminals

Terminal	Protocol
USB	LPBUS Protocol
RS232	LPBUS Protocol/ASCII
CAN	CANOpen / CAN sequential

3.1 Communication Modes

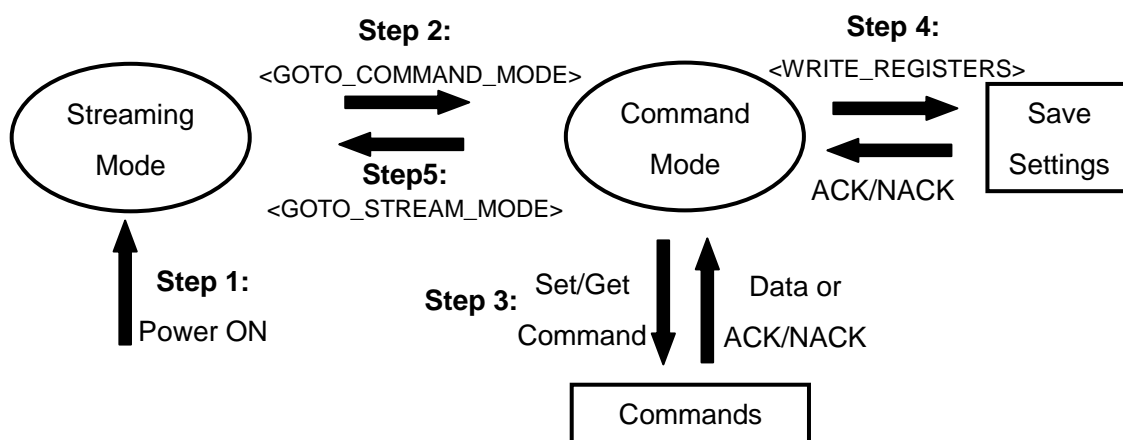


Fig. 2. State diagram of sensor communication modes

There are two operational modes, Streaming Mode and Command mode in LPMS-IG1 sensors. By default, the sensor will start in Streaming Mode on power up. In Streaming Mode, the sensor will continuously stream out sensor data via USB and RS232/CAN terminals simultaneously. Streaming frequency for both USB and RS232/CAN terminals is determined by the data streaming rate settings (default at 100Hz). The sensor will stop data streaming in Command Mode. User can change the sensor internal parameters in both Streaming and Command mode, but **it is highly**



recommended to put the sensor in Command Mode before making any parameter changes. The state diagram of the sensor modes is summarized in Fig. 2.**Error! Reference source not found.**

NOTE: User must issue a Save Parameters command to retain any parameter changes to the sensor before next power cycle.

3.2 LPBUS Protocol

LPBUS is a communication protocol based on the industry standard MODBUS protocol. It is the default communication format used by LPMS sensors. .An LPBUS communication packet has two basic command types, GET and SET, that are sent from a host (PC, mobile data logging unit etc.) to a client (LPMS sensor). Later in this manual we will show a description of all supported commands to the sensor, their types and transported data.

GET Commands

Data from the client is read using GET requests. A GET request usually contains no data. The answer from the client to a GET request contains the requested data.

SET Commands

Data registers of the client are written using SET requests. A SET command from the host contains the data to be set. The answer from the client is either ACK (acknowledged) for a successful write, or NACK (not acknowledged) for a failure to set the register occurred.

3.3 LPBUS Packet Format

Each packet sent during the communication is based on the following structure:

Table 2. LPBUS protocol

Flag Name	Start	Sensor ID	Command No.	Data length	Data field	LRC	End
Bytes	1	2	2	2	n	2	2

- **Start:** 1 byte. data packet start flag, which is fixed to 3Ah.



- **Sensor ID:** 2 bytes transmitted at LSB. It contains ID of the sensor to be communicated with. The default value of this ID is 01h. The host sends out a GET / SET request to a specific LPMS sensor by using this ID, and the client answers to request also with the same ID. This ID can be adjusted by sending a SET command to the sensor firmware.
- **Command No.:** 2 bytes transmitted at LSB. It contains command number information to be performed by the data transmission.
- **Data length:** 2 bytes transmitted at LSB, It contains the length information of packet data field.
- **Data field:** n bytes transmitted at LSB, where n is not a fixed number which depends on the command types. It contains all the data needed to be transmitted under a specific command.
- **LRC:** 2 bytes transmitted at LSB. It contains the packet checksum information. To ensure the integrity of the transmitted data the LRC checksum is used. It is calculated in the following way:
$$LRC = \text{sum}(\text{Sensor ID}, \text{Command no.}, \text{Data length}, \text{and Data field}).$$
The calculated LRC is usually compared with the LRC transmitted from the remote device. If the two LRCs are not equal, an error is reported.
- **End:** 2 bytes transmitted at LSB, which is fixed to 0D0Ah.



Data Format in a Packet Data Field

Generally, data is sent in little-endian format, low order byte first, high order byte last. Data in the data fields of a packet can be encoded in several ways, depending on the type of information to be transmitted. In the following we list the most common data types. Other command-specific data types are explained in the command reference.

Table 3. Data format in a packet data field

Identifier	Description
Int32	32-bit signed integer value
UInt32	32-bit unsigned integer value
Int16	16-bit signed integer value
UInt16	16-bit unsigned integer value
Int8	8-bit signed integer value
UInt8	8-bit unsigned integer value
Float32	32-bit float value
Vector3f	3 element 32-bit float vector
Vector3i16	3 element 16-bit signed integer vector
Vector4f	4 element 32-bit float vector
Vector4i16	4 element 16-bit signed integer vector
Matrix3x3f	3x3 element 32-bit float value matrix

Sensor Measurement Data

IMU data

There are two precision modes for sensor data: **32-bit float or 16-bit integer**. Users can switch between these modes based on the requested data sampling rate and volume. While sensor is in streaming operational mode, LPBUS transports measurement data in the data field of a packet in the following orders showed in Table 4 and Table 5 respectively for the cases of 32-bit float and 16-bit integer precision modes. The order of the sensor data chunks depends on which sensor data is enabled.

NOTICE: Timestamp data is always 32-bit unsigned integer in both data precision modes.



Table 4. Data order in 32-bit float precision mode

Order	Identifier	Description	Unit
1	UInt32	Timestamp	multiply by factor 0.002 to convert to seconds
2	Vector3f	Raw accelerometer	g
3	Vector3f	Calibrated accelerometer	g
4	Vector3f	Raw Gyro I	dps (default) or rad/s
5	Vector3f	Raw Gyroll	dps (default) or rad/s
6	Vector3f	Static bias calibrated Gyrol	dps (default) or rad/s
7	Vector3f	Static bias calibrated Gyroll	dps (default) or rad/s
8	Vector3f	Alignment calibrated Gyrol	dps (default) or rad/s
9	Vector3f	Alignment calibrated Gyroll	dps (default) or rad/s
10	Vector3f	Raw magnetometer	uT
11	Vector3f	Calibrated magnetometer	uT
12	Vector3f	Reserved	
13	Vector4f	Quaternion	
14	Vector3f	Euler	deg (default) or rad
15	Vector3f	Linear acceleration (g)	g
16	Float32	Reserved	
17	Float32	Reserved	
18	Float32	Temperature	°C

In 16-bit data precision mode values are transmitted to the host with a multiplication factor applied to increase precision:

Table 5. Data order in 16-bit integer precision mode

Order	Format	Sensor data	Scale factor
1	UInt32	Timestamp	multiply by factor 0.002 to convert to seconds
2	Vector3i16	Raw accelerometer (g)	1000
3	Vector3i16	Calibrated accelerometer (g)	1000
4	Vector3i16	Raw Gyrol (dps or rad/s)	dps: 10 rad/s: 100
5	Vector3i16	Raw Gyroll (dps or rad/s)	10/100



6	Vector3i16	Static bias calibrated Gyrol (dps or rad/s)	10/100
7	Vector3i16	Static bias calibrated Gyroll (dps or rad/s)	10/100
8	Vector3i16	Alignment calibrated Gyrol (dps or rad/s)	10/100
9	Vector3i16	Alignment calibrated Gyroll (dps or rad/s)	dps: 10 rad/s: 100
10	Vector3i16	Raw magnetometer (uT)	100
11	Vector3i16	Calibrated magnetometer (uT)	100
12	Vector3i16	Reserved	
13	Vector4i16	Quaternion	10000
14	Vector3i16	Euler(degree or rad)	dps: 100 rad/s:10000
15	Vector3i16	Linear acceleration (g)	1000
16	Int16	Reserved	
17	Int16	Reserved	
18	Int16	Temperature (°C)	100

GPS Data

In addition to IMU data, LPMS-IG1P (with GPS) will output additional 1Hz GPS data packet with the following format:

Table 6. GPS data

Order	Format	Sensor data	Scale factor
1	UInt32	Timestamp	
2	UInt32	PVT iTOW (ms) - GPS time of week of the navigation epoch.	
3	UInt16	PVT year (UTC)	
4	UInt8	PVT month (UTC)	
5	UInt8	PVT day (UTC)	
6	UInt8	PVT hour (UTC)	
7	UInt8	PVT min (UTC)	
8	UInt8	PVT sec (UTC)	
9	UInt8	PVT valid - Validity flags	
10	UInt32	PVT tAcc - Time accuracy estimate	



		(UTC)	
11	Int32	PVT nano (ns) - Fraction of second (UTC)	
12	UInt8	PVT fixType	
13	UInt8	PVT flags - Fix status flags	
14	UInt8	PVT flags2 - Additional flags	
15	UInt8	PVT numSV - Number of satellites used in Nav Solution	
16	Int32	PVT longitude (deg)	10000000
17	Int32	PVT latitude (deg)	10000000
18	Int32	PVT height (mm) - Height above ellipsoid	
19	Int32	PVT hMSL (mm) - Height above mean sea level	
20	UInt32	PVT hAcc (mm)- Horizontal accuracy estimate	
21	UInt32	PVT vAcc (mm) - Vertical accuracy estimate	
22	Int32	PVT veIN (mm/s) - NED north velocity	
23	Int32	PVT veIE (mm/s) - NED east velocity	
24	Int32	PVT veID (mm/s) - NED down velocity	
25	Int32	PVT gSpeed (mm/s) - Ground Speed (2-D)	
26	Int32	PVT headMot (deg) - Heading of motion (2-D)	100000
27	UInt32	PVT sAcc (mm/s) - Speed accuracy estimate	
28	UInt32	PVT headAcc (deg) - Heading accuracy estimate	100000
29	UInt16	PVT pDOP - Position DOP	100
30	Int32	PVT headVeh (deg) - Heading of vehicle (2-D)	100000



31	UInt32	ATT iTOW (ms) - GPS time of week of the navigation epoch	
32	UInt8	ATT version - Message version (0 for this version)	
33	Int32	ATT roll (deg) - Vehicle roll	100000
34	Int32	ATT pitch (deg) - Vehicle pitch	100000
35	Int32	ATT heading (deg) - Vehicle heading	100000
36	UInt32	ATT accRoll (deg) - Vehicle roll accuracy	100000
37	UInt32	ATT accPitch (deg) - Vehicle pitch accuracy	100000
38	UInt32	ATT accHeading (deg) - Vehicle heading accuracy	100000
39	UInt32	ESF iTOW (ms) - GPS time of week of the navigation epoch	
40	UInt8	ESF version - Message version (2 for this version)	
41	UInt8	ESF initState1	
42	UInt8	ESF initState2	
43	UInt8	ESF fusionMode	
44	UInt8	ESF numSens - Number of sensors	
45	UInt32[n]	ESF sensStatus	

3.4 LPBUS Example Communication

In this section we will show a few practical examples of communication using the LPBUS protocol.

Goto Command Mode

GET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start



1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	06h	Command no. LSB (06h = GOTO_COMMAND_MODE)
4	00h	Command no. MSB
5	00h	Data length LSB (GET command = no data)
6	00h	Data length MSB
7	07h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	00h	Command no. LSB (00h = REPLY_ACK)
4	00h	Command no. MSB
5	00h	Data length LSB (Zero length data for ACK reply)
6	00h	Data length MSB
7	01h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Goto Steaming Mode

GET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	07h	Command no. LSB(07h = GOTO_STREAMING_MODE)



4	00h	Command no. MSB
5	00h	Data length LSB (GET command = no data)
6	00h	Data length MSB
7	08h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	00h	Command no. LSB (00h = REPLY_ACK)
4	00h	Command no. MSB
5	00h	Data length LSB (Zero length data for ACK reply)
6	00h	Data length MSB
7	01h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2



Request Gyroscope Range

GET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	3Dh	Command no. LSB(3Dh = GET_GYRO_RANGE)
4	00h	Command no. MSB
5	00h	Data length LSB (GET command = no data)
6	00h	Data length MSB
7	3Eh	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	3Dh	Command no. LSB(3Dh = GET_GYRO_RANGE)
4	00h	Command no. MSB
5	04h	Data length LSB (32-bit integer = 4 bytes)
6	00h	Data length MSB
7	xxh	Range data byte 1 (LSB)
8	xxh	Range data byte 2
9	xxh	Range data byte 3
10	xxh	Range data byte 4 (MSB)
11	xxh	Check sum LSB
12	xxh	Check sum MSB
13	0Dh	Packet end 1
14	0Ah	Packet end 2

xx = Value depends on the current sensor configuration.



Set Accelerometer Range

SET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	32h	Command no. LSB (32h = SET_ACC_RANGE)
4	00h	Command no. MSB
5	04h	Data length LSB (32-bit integer = 4 bytes)
6	00h	Data length MSB
7	08h	Range data byte 1 (Range indicator 8g = 8d)
8	00h	Range data byte 2
9	00h	Range data byte 3
10	00h	Range data byte 4
11	3Fh	Check sum LSB
12	00h	Check sum MSB
13	0Dh	Packet end 1
14	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	00h	Command no. LSB (00h = REPLY_ACK)
4	00h	Command no. MSB
5	00h	Data length LSB (Zero length data for ACK reply)
6	00h	Data length MSB
7	01h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2



Save Sensor Parameters

WRITE_REGISTER request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	04h	Command no. LSB(04h = WRITE_REGISTER)
4	00h	Command no. MSB
5	00h	Data length LSB (WRITE_REGISTER command = no data)
6	00h	Data length MSB
7	05h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	00h	Command no. LSB (00h = REPLY_ACK)
4	00h	Command no. MSB
5	00h	Data length LSB (Zero length data for ACK reply)
6	00h	Data length MSB
7	01h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

NOTE: WRITE_REGISTER command involves flash operation, which might result in delay ACK response



Read Sensor Status

GET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	08h	Command no. LSB(08h = GET_SENSOR_STATUS)
4	00h	Command no. MSB
5	00h	Data length LSB (GET command = no data)
6	00h	Data length MSB
7	09h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	08h	Command no. LSB (08h = GET_SENSOR_STATUS)
4	00h	Command no. MSB
5	04h	Data length LSB (32-bit integer = 4 bytes)
6	00h	Data length MSB
7-10	xxxxxxxh	Sensor status data
11	xxh	Check sum LSB
12	xxh	Check sum MSB
13	0Dh	Packet end 1
14	0Ah	Packet end 2

NOTE: Please refer to Appendix for details of reply data mapping



Set UART / RS232 Baudrate

SET request (HOST -> SENSOR)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	82h	Command no. LSB (82h = SET_UART_BAUDRATE)
4	00h	Command no. MSB
5	04h	Data length LSB (32-bit integer = 4 bytes)
6	00h	Data length MSB
7	00h	921600 = 0x000E1000
8	10h	
9	0Eh	
10	00h	
11	A5h	Check sum LSB
12	00h	Check sum MSB
13	0Dh	Packet end 1
14	0Ah	Packet end 2

Reply data (SENSOR -> HOST)

Packet byte no.	Content	Description
0	3Ah	Packet start
1	01h	Sensor ID LSB (ID = 1)
2	00h	Sensor ID MSB
3	00h	Command no. LSB (00h = REPLY_ACK)
4	00h	Command no. MSB
5	00h	Data length LSB (Zero length data for ACK reply)
6	00h	Data length MSB
7	01h	Check sum LSB
8	00h	Check sum MSB
9	0Dh	Packet end 1
10	0Ah	Packet end 2

NOTE: Power cycle is required for baudrate settings to take effect



3.5 CANOpen and Sequential CAN Protocols

In CANOpen and sequential CAN transmission modes, two or more output words of measurement data can be assigned to a CAN channel. In sequential CAN mode the channel addressing can be individually controlled. In CANOpen mode, 4 TPDO (Transmission Data Process Object) messages and a heartbeat message are transmitted.

NOTE: In CANOpen mode a heartbeat message is transmitted with a frequency between 0.1 Hz and 2 Hz.

In CANOpen mode, the message base address is calculated in the following way:

$$\text{CAN ID1} = \text{Base CAN ID} = \text{Start ID} + \text{IMU ID}$$

$$\text{CAN ID2} = \text{CAN ID1} + 100\text{h}$$

$$\text{CAN ID3} = \text{CAN ID2} + 100\text{h}$$

$$\text{CAN ID4} = \text{CAN ID3} + 100\text{h}$$

$$\text{CAN ID5} = 700\text{h} + \text{IMU ID}$$

Note: In CANOpen mode, Start ID is fixed at 180h

In sequential CAN mode, the message base address is calculated in the following way:

$$\text{CAN ID1} = \text{Base CAN ID} = \text{Start ID} + \text{IMU ID}$$

$$\text{CAN ID2} = \text{CAN ID1} + 1\text{h}$$

$$\text{CAN ID3} = \text{CAN ID2} + 1\text{h}$$

$$\text{CAN ID4} = \text{CAN ID3} + 1\text{h}$$

NOTE: In sequential CAN mode, Start ID is set to 514h (1300d) by default. This value can be changed via IG1Control interface

3.6 CAN Data Output Format

Each CAN message can be assigned multiple channels representing the sensor data. The number of assignable sensor data will depend on the data output precision, i.e. 32bit data or 16bit data output. By utilizing 4 CAN message with 16bit sensor data precision, the sensor can output a maximum of 16 different sensor data for a given instance. Table 7 and Table 8 summarizes the channel mapping for 16bit and 32bit data output.



Table 7. CAN channel mapping (16bit data output)

CANOpen ID	Sequential CAN	Output data
181h	515h	channel 1 channel 2 channel 3 channel 4
281h	516h	channel 5 channel 6 channel 7 channel 8
381h	517h	channel 9 channel 10 channel 11 channel 12
481h	518h	channel 13 channel 14 channel 15 channel 16
701h	-	- (heartbeat)

Table 8. CAN channel mapping (32bit data output)

CANOpen ID	Sequential CAN	Output data
181h	515h	channel 1 channel 2
281h	516h	channel 3 channel 4
381h	517h	channel 5 channel 6
481h	518h	channel 7 channel 8
701h	-	- (heartbeat)

3.7 CAN Mapping

Each channel can be assigned different sensor data by changing the CAN mapping via IG1Control. **Error! Reference source not found.** summarizes the available sensor output data.

Table 9. CAN data mapping

Mapping index	Data	Unit
0	Not assigned	
1	Raw Accelerometer X	g
2	Raw Accelerometer Y	g
3	Raw Accelerometer Z	g
4	Calibrated Accelerometer X	g
5	Calibrated Accelerometer Y	g
6	Calibrated Accelerometer Z	g
7	Raw Gyrol X	dps or rad/s
8	Raw Gyrol Y	dps or rad/s
9	Raw Gyrol Z	dps or rad/s



10	Raw Gyroll X	dps or rad/s
11	Raw Gyroll Y	dps or rad/s
12	Raw Gyroll Z	dps or rad/s
13	Bias calibrated Gyrol X	dps or rad/s
14	Bias calibrated Gyrol Y	dps or rad/s
15	Bias calibrated Gyrol Z	dps or rad/s
16	Bias calibrated Gyroll X	dps or rad/s
17	Bias calibrated Gyroll Y	dps or rad/s
18	Bias calibrated Gyroll Z	dps or rad/s
19	Alignment calibrated Gyrol X	dps or rad/s
20	Alignment calibrated Gyrol Y	dps or rad/s
21	Alignment calibrated Gyrol Z	dps or rad/s
22	Alignment calibrated Gyroll X	dps or rad/s
23	Alignment calibrated Gyroll Y	dps or rad/s
24	Alignment calibrated Gyroll Z	dps or rad/s
25	Raw magnetometer X	uT
26	Raw magnetometer Y	uT
27	Raw magnetometer Z	uT
28	Calibrated magnetometer X	uT
29	Calibrated magnetometer Y	uT
30	Calibrated magnetometer Z	uT
31	Reserved	
32	Reserved	
33	Reserved	
34	Quaternion W	
35	Quaternion X	
36	Quaternion Y	
37	Quaternion Z	
38	Euler X	deg or rad
39	Euler X	deg or rad
40	Euler Z	deg or rad
41	Reserved	
42	Reserved	
43	Reserved	
44	Reserved	
45	Temperature	°C





3.8 ASCII Output

LPMS-IG1 RS232 supports both LPBUS protocol and ASCII output via RS232 terminal. The output format can be set via SET_UART_FORMAT command. In ASCII output format, every data packet has a prefix Start character and End character at the end. Default start character is '\$' and end character is '\n'. User can define both Start and End character via SET_UART_ASCII_CHARACTER command. Sensor data is streamed out in comma separated format. Sensor data is in 16bit integer format. ASCII data should be scaled according Table 10 to obtain correct data.

Table 10. ASCII output format

Start character	Data 1	,	Data 2	,	,	Data n	End character
-----------------	--------	---	--------	---	-------	---	--------	---------------



4. Appendix

4.1 Firmware Function / Command List

Summary

Acknowledged / Not-acknowledged Identifiers				
Identifier	Name	Parameter	Response	Default
0(00h)	REPLY_ACK			
1(01h)	REPLY_NACK			

Register Value Save and Reset Command				
Identifier	Name	Parameter	Response	Default
4 (04h)	WRITE_REGISTERS	NONE	ACK/NACK	
5 (05h)	RESTORE_FACTORY_VALUE	NONE	ACK/NACK	

Mode Switching Commands				
Identifier	Name	Parameter	Response	Default
6 (06h)	GOTO_COMMAND_MODE	NONE	ACK/NACK	
7 (07h)	GOTO_STREAM_MODE	NONE	ACK/NACK	

Sensor Status Command				
Identifier	Name	Parameter	Response	Default
8 (08h)	GET_SENSOR_STATUS	NONE	UInt32	1

Get Data Commands				
Identifier	Name	Parameter	Response	Default
9 (09h)	GET_IMU_DATA	NONE		
10 (0Ah)	GET_GPS_DATA	NONE		

Device Info				
Identifier	Name	Parameter	Response	Default
20 (14h)	GET_SENSOR_MODEL	NONE	Char[24]	
21 (15h)	GET_FIRMWARE_INFO	NONE	Char[24]	
22 (16h)	GET_SERIAL_NUMBER	NONE	Char[24]	
23(17h)	GET_FILTER_VERSION	NONE	Char[24]	



Data Transmission Commands				
Identifier	Name	Parameter	Response	Default
30 (1Eh)	SET_IMU_TRANSMIT_DATA	UInt32	ACK/NACK	
31 (1Fh)	GET_IMU_TRANSMIT_DATA	NONE	UInt32	

IMU ID Settings Commands				
Identifier	Name	Parameter	Response	Default
32 (20h)	SET_IMU_ID	Int32	ACK/NACK	
33 (21h)	GET_IMU_ID	NONE	Int32	1

Stream Frequency Commands				
Identifier	Name	Parameter	Response	Default
34 (22h)	SET_STREAM_FREQ	Int32	ACK/NACK	
35 (23h)	GET_STREAM_FREQ	NONE	Int32	100

Deg/Rad Output Commands				
Identifier	Name	Parameter	Response	Default
36 (24h)	SET_DEGRAD_OUTPUT	Int32	ACK/NACK	
37 (25h)	GET_DEGRAD_OUTPUT	NONE	Int32	0

Reference Setting and Offset Reset Commands				
Identifier	Name	Parameter	Response	Default
38 (26h)	SET_ORIENTATION_OFFSET	Int32	ACK/NACK	
39 (27h)	RESET_ORIENTATION_OFFSET	NONE	ACK/NACK	

Accelerometer Settings Commands				
Identifier	Name	Parameter	Response	Default
50 (32h)	SET_ACC_RANGE	Int32	ACK/NACK	
51 (33h)	GET_ACC_RANGE	NONE	Int32	4g

Gyroscope Settings Commands				
Identifier	Name	Parameter	Response	Default
60 (3Ch)	SET_GYR_RANGE	Int32	ACK/NACK	
61 (3Dh)	GET_GYR_RANGE	NONE	Int32	500dps
62 (3Eh)	START_GYR_CALIBRATION	NONE	ACK/NACK	
64 (40h)	SET_ENABLE_GYR_AUTOCALIBRATION	Int32	ACK/NACK	
65 (41h)	GET_ENABLE_GYR_AUTOCALIBRATION	NONE	Int32	1
66(42h)	SET_GYR_THRESHOLD	Float32	ACK/NACK	



67 (43h)	GET_GYR_THRESHOLD	NONE	Float32	0
----------	-------------------	------	---------	---

Magnetometer Settings Commands				
Identifier	Name	Parameter	Response	Default
70 (46h)	SET_MAG_RANGE	Int32	ACK/NACK	
71 (47h)	GET_MAG_RANGE	NONE	Int32	8Gauss
84 (54h)	START_MAG_CALIBRATION	NONE	ACK/NACK	
85 (55h)	STOP_MAG_CALIBRATION	NONE	ACK/NACK	
86 (56h)	SET_MAG_CALIBRATION_TIMEOUT	Int32	ACK/NACK	
87 (57h)	GET_MAG_CALIBRATION_TIMEOUT	NONE	Int32	20s

Filter Settings Commands				
Identifier	Name	Parameter	Response	Default
90 (5Ah)	SET_FILTER_MODE	Int32	ACK/NACK	
91 (5Bh)	GET_FILTER_MODE	NONE	Int32	1

CAN Settings Command				
Identifier	Name	Parameter	Response	Default
110 (6Eh)	SET_CAN_START_ID	Int32	ACK/NACK	
111 (6Fh)	GET_CAN_START_ID	NONE	Int32	0x514
112 (70h)	SET_CAN_BAUDRATE	Int32	ACK/NACK	
113 (71h)	GET_CAN_BAUDRATE	NONE	Int32	500
114 (72h)	SET_CAN_DATA_PRECISION	Int32	ACK/NACK	
115 (73h)	GET_CAN_DATA_PRECISION	NONE	Int32	0
116 (74h)	SET_CAN_MODE	Int32	ACK/NACK	
117 (75h)	GET_CAN_MODE	NONE	Int32	0
118 (76h)	SET_CAN_MAPPING	Int32[16]	ACK/NACK	
119 (77h)	GET_CAN_MAPPING	NONE	Int32[16]	
120 (78h)	SET_CAN_HEARTBEAT	Int32	ACK/NACK	
121 (79h)	GET_CAN_HEARTBEAT	NONE	Int32	1

UART / RS232 Settings Command				
Identifier	Name	Parameter	Response	Default
130 (82h)	SET_UART_BAUDRATE	Int32	ACK/NACK	
131 (83h)	GET_UART_BAUDRATE	NONE	Int32	921600
132 (84h)	SET_UART_FORMAT	Int32	ACK/NACK	
133 (85h)	GET_UART_FORMAT	NONE	Int32	0
134 (86h)	SET_UART_ASCII_CHARACTER	Int8[4]	ACK/NACK	
135 (87h)	GET_UART_ASCII_CHARACTER	NONE	Int8[4]	h24 h0D



136 (88h)	SET_LPBUS_DATA_PRECISION	Int32	ACK/NACK	
137 (89h)	GET_LPBUS_DATA_PRECISION	NONE	Int32	1

Sensor Data Timestamp Manipulation				
Identifier	Name	Parameter	Response	Default
152 (98h)	SET_TIMESTAMP	Int32	ACK/NACK	

GPS Data Transmission Commands				
Identifier	Name	Parameter	Response	Default
160 (A0h)	SET_GPS_TRANSMIT_DATA	Int32[2]	ACK/NACK	
161 (A1h)	GET_GPS_TRANSMIT_DATA	NONE	Int32[2]	
162 (A2h)	SAVE_GPS_STATE	NONE	ACK/NACK	
163 (A3h)	CLEAR_GPS_STATE	NONE	ACK/NACK	

Acknowledged and Not-acknowledged Identifiers

Identifier	0(0x00)
Name	REPLY_ACK
Description	Confirms a successful SET command.

Identifier	1(0x01)
Name	REPLY_NACK
Description	Reports an error during processing a SET command.

Register Value Save and Reset Command

Identifier	4 (0x04)
Name	WRITE_REGISTERS
Description	Write the currently set parameters to flash memory.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	5 (0x05)
Name	RESTORE_FACTORY_VALUE
Description	Reset the LPMS parameters to factory default values. Please note that upon issuing this command your currently set parameters will be erased.
Parameter	NONE
Response:	ACK (success) or NACK (error)



Mode Switching Commands

Identifier	6 (0x06)
Name	GOTO_COMMAND_MODE
Description	Switch to command mode. In command mode the user can issue commands to the firmware to perform calibration, set parameters etc.
Parameter	NONE
Response:	ACK (success) or NACK (error)
Identifier	7 (0x07)
Name	GOTO_STREAM_MODE
Description	Switch to streaming mode. In this mode data is continuously streamed from the sensor, and some commands cannot be performed until the sensor receives the GOTO_COMMAND_MODE command.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Sensor Status Command

Identifier	8 (0x08)						
Name	GET_SENSOR_STATUS						
Description	Get the current sensor status						
Parameter	NONE						
Response:	Int32 <table border="1"> <thead> <tr> <th>Sensor status</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>Command Mode</td> <td>0</td> </tr> <tr> <td>Streaming Mode</td> <td>1</td> </tr> </tbody> </table>	Sensor status	Identifier	Command Mode	0	Streaming Mode	1
Sensor status	Identifier						
Command Mode	0						
Streaming Mode	1						

Get Data Commands

Identifier	9(0x09)
Name	GET_IMU_DATA
Description	Get the sensor data
Parameter	NONE
Response	Please see Chapter 5.3 for details

Identifier	10(0x0A)
Name	GET_GPS_DATA
Description	Get the GPS data
Parameter	NONE
Response	Please see Chapter 5.3 for details



Device Info Commands

Identifier	20 (0x14)
Name	GET_SENSOR_MODEL
Description	Get the sensor model information
Parameter	NONE
Response	Char[24]

Identifier	21 (0x15)
Name	GET_FIRMWARE_INFO
Description	Get the firmware information
Parameter	NONE
Response	Char[24]

Identifier	22 (0x16)
Name	GET_SERIAL_NUMBER
Description	Get the serial number information
Parameter	NONE
Response	Char[24]

Identifier	23 (0x17)
Name	GET_FILTER_VERSION
Description	Get the internal filter version information
Parameter	NONE
Response	Char[24]

Data Transmission Commands

Identifier	30 (0x1E)
Name	SET_IMU_TRANSMIT_DATA
Description	Set the current transmitted data of sensor



Parameter	Int32	
	Bit	Reported State / Parameter
	0	Accelerometer raw data transmission enabled
	1	Accelerometer calibrated data transmission enabled
	2	Gyrol raw data transmission enabled
	3	Gyroll raw data transmission enabled
	4	Gyrol bias calibrated data transmission enabled
	5	Gyroll bias calibrated data transmission enabled
	6	Gyrol alignment calibrated data transmission enabled
	7	Gyroll alignment calibrated data transmission enabled
	8	Magnetometer raw data transmission enabled
	9	Magnetometer calibrated data transmission enabled
	10	Reserved
	11	Quaternion orientation transmission enabled
	12	Euler angle data transmission enabled
	13	Reserved
	14	Reserved
	15	Reserved
16	Temperature data transmission enabled	
17-31	Reserved	
Response:	ACK (success) or NACK (error)	

Identifier	31 (0x1F)	
Name	GET_IMU_TRANSMIT_DATA	
Description	Get the current transmitted data from sensor	
Parameter	NONE	
Response:	Int32	
	Bit	Reported State / Parameter
	0	Accelerometer raw data transmission enabled
	1	Accelerometer calibrated data transmission enabled
	2	Gyrol raw data transmission enabled
	3	Gyroll raw data transmission enabled
	4	Gyrol bias calibrated data transmission enabled
	5	Gyroll bias calibrated data transmission enabled
	6	Gyrol alignment calibrated data transmission enabled
	7	Gyroll alignment calibrated data transmission enabled
	8	Magnetometer raw data transmission enabled
	9	Magnetometer calibrated data transmission enabled
	10	Reserved
	11	Quaternion orientation transmission enabled
	12	Euler angle data transmission enabled
	13	Reserved
	14	Reserved
	15	Reserved
16	Temperature data transmission enabled	
17-31	Reserved	



IMU ID Setting Command

Identifier	32 (0x20)
Name	SET_IMU_ID
Description	Set sensor ID
Parameter	Int32
Response:	ACK (success) or NACK (error)

Identifier	33 (0x21)
Name	GET_IMU_ID
Description	Get sensor ID
Parameter	None
Response:	Int32

Stream Frequency Commands

Identifier	34 (0x22)												
Name	SET_STREAM_FREQ												
Description	Set the current streaming frequency												
Parameter	Int32 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Frequency (Hz)</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>5</td> </tr> <tr> <td>10</td> <td>10</td> </tr> <tr> <td>50</td> <td>50</td> </tr> <tr> <td>100</td> <td>100</td> </tr> <tr> <td>500</td> <td>500</td> </tr> </tbody> </table>	Frequency (Hz)	Identifier	5	5	10	10	50	50	100	100	500	500
Frequency (Hz)	Identifier												
5	5												
10	10												
50	50												
100	100												
500	500												
Response:	ACK (success) or NACK (error)												

Identifier	35 (0x23)												
Name	GET_STREAM_FREQ												
Description	Get the current streaming frequency												
Parameter	NONE												
Response:	Int32 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Frequency (Hz)</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>5</td> </tr> <tr> <td>10</td> <td>10</td> </tr> <tr> <td>50</td> <td>50</td> </tr> <tr> <td>100</td> <td>100</td> </tr> <tr> <td>500</td> <td>500</td> </tr> </tbody> </table>	Frequency (Hz)	Identifier	5	5	10	10	50	50	100	100	500	500
Frequency (Hz)	Identifier												
5	5												
10	10												
50	50												
100	100												
500	500												



Deg/Rad Output Commands

Identifier	36 (0x24)	
Name	SET_DEGRAD_OUTPUT	
Description	Set the current output unit of angle and rate	
Parameter	Int32	
	Output unit	Identifier
	degree or degree per second	0
	radian or radian per second	1
Response:	ACK (success) or NACK (error)	

Identifier	37 (0x25)	
Name	GET_DEGRAD_OUTPUT	
Description	Get the current output unit of angle and rate	
Parameter	NONE	
Response:	Int32	
	Output unit	Identifier
	degree or degree per second	0
	radian or radian per second	1

Reference Setting and Offset Reset Command

Identifier	38 (0x26)	
Name	SET_ORIENTATION_OFFSET	
Description	Set the orientation offset (unity quaternion).	
Parameter	Int32	
	Offset Mode	Identifier
	Object	0
	Heading	1
	Alignment	2
Response:	ACK (success) or NACK (error)	

Identifier	39 (0x27)	
Name	RESET_ORIENTATION_OFFSET	
Description	Reset the orientation offset to 0 (unity quaternion).	
Parameter	NONE	
Response:	ACK (success) or NACK (error)	



Accelerometer Settings Command

Identifier	50 (0x32)	
Name	SET_ACC_RANGE	
Description	Set the current range of the accelerometer	
Parameter	Int32	
	Range	Identifier
	2g	2
	4g	4
	8g	8
Response:	ACK (success) or NACK (error)	

Identifier	51 (0x33)	
Name	GET_ACC_RANGE	
Description	Get the current range of the accelerometer	
Parameter	NONE	
Response:	Int32	
	Range	Identifier
	2g	2
	4g	4
	8g	8

Gyroscope Settings Command

Identifier	60 (0x3C)	
Name	SET_GYR_RANGE	
Description	Set the current range of the gyroscope	
Parameter	Int32	
	Range (deg/s)	Identifier
	400	400
	1000	1000
Response:	ACK (success) or NACK (error)	

Identifier	61 (0x3D)	
Name	GET_GYR_RANGE	
Description	Get current gyroscope range.	
Parameter	NONE	
Response:	Int32	
	Range (deg/s)	Identifier
	400	400
	1000	1000



Identifier	62 (0x3E)
Name	START_GYR_CALIBRATION
Description	Start gyro static bias calibration.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	64 (0x40)						
Name	SET_ENABLE_GYR_AUTOCALIBRATION						
Description	Enable / Disable gyro autocalibration.						
Parameter	Int32						
	<table border="1"> <thead> <tr> <th>Function</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>ENABLE_GYR_AUTOCAL</td> <td>1</td> </tr> <tr> <td>DISABLE_GYR_AUTOCAL</td> <td>0</td> </tr> </tbody> </table>	Function	Identifier	ENABLE_GYR_AUTOCAL	1	DISABLE_GYR_AUTOCAL	0
	Function	Identifier					
ENABLE_GYR_AUTOCAL	1						
DISABLE_GYR_AUTOCAL	0						
Response:	ACK (success) or NACK (error)						

Identifier	65 (0x41)						
Name	GET_ENABLE_GYR_AUTOCALIBRATION						
Description	Get gyro autocalibration status.						
Parameter	NONE						
Response:	Int32						
	<table border="1"> <thead> <tr> <th>Function</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>ENABLE_GYR_AUTOCAL</td> <td>1</td> </tr> <tr> <td>DISABLE_GYR_AUTOCAL</td> <td>0</td> </tr> </tbody> </table>	Function	Identifier	ENABLE_GYR_AUTOCAL	1	DISABLE_GYR_AUTOCAL	0
	Function	Identifier					
ENABLE_GYR_AUTOCAL	1						
DISABLE_GYR_AUTOCAL	0						

Identifier	66 (0x42)
Name	SET_GYR_THRESHOLD
Description	Set the gyroscope threshold which can be used to suppress noise or vibrations that might impact the sensor measurements.
Parameter	Float32
Response:	ACK (success) or NACK (error)

Identifier	67 (0x43)
Name	GET_GYR_THRESHOLD
Description	Get the gyroscope threshold.
Parameter	NONE
Response:	Float32

Magnetometer Settings Command

Identifier	70 (0x46)
Name	SET_MAG_RANGE



Description	Set the current range of the gyroscope	
Parameter	Int32	
	Range	Identifier
	2 Gauss	2
	8 Gauss	8
Response:	ACK (success) or NACK (error)	

Identifier	71 (0x47)	
Name	GET_MAG_RANGE	
Description	Get current magnetometer range.	
Parameter	NONE	
Response:	Int32	
	Range	Identifier
	2 Gauss	2
	8 Gauss	8

Identifier	84 (0x54)	
Name	START_MAG_CALIBRATION	
Description	Start calibration of magnetometer.	
Parameter	NONE	
Response:	ACK (success) or NACK (error)	

Identifier	85 (0x55)	
Name	STOP_MAG_CALIBRATION	
Description	Stop calibration of magnetometer.	
Parameter	NONE	
Response:	ACK (success) or NACK (error)	

Identifier	86 (0x56)	
Name	SET_MAG_CALBRATION_TIMEOUT	
Description	Set the time of the magnetometer calibration.	
Parameter	Int32	
Response:	ACK (success) or NACK (error)	

Identifier	87 (0x57)	
Name	GET_MAG_CALBRATION_TIMEOUT	
Description	Get the time of the magnetometer calibration.	
Parameter	NONE	
Response:	Int32	



Filter Settings Command

Identifier	90 (0x5A)	
Name	SET_FILTER_MODE	
Description	Set the sensor filter mode	
Parameter	Int32	
	Mode	Value
	Gyroscope(Only)	0
	Accelerometer + gyroscope (Kalman filter)	1
	Accelerometer + gyroscope + Magnetometer(Kalman filter)	2
	Accelerometer + gyroscope (DCM filter)	3
Response:	ACK (success) or NACK (error)	

Identifier	91 (0x5B)	
Name	GET_FILTER_MODE	
Description	Get the sensor filter mode	
Parameter	NONE	
Response:	Int32	
	Mode	Value
	Gyroscope(Only)	0
	Accelerometer + gyroscope (Kalman filter)	1
	Accelerometer + gyroscope + Magnetometer(Kalman filter)	2
	Accelerometer + gyroscope (DCM filter)	3
	Accelerometer + gyroscope + Magnetometer(DCM filter)	4

CAN Settings Command

Identifier	110 (0x6E)	
Name	SET_CAN_START_ID	
Description	Set the CAN sequential start ID	
Parameter	Int32	
Response:	ACK (success) or NACK (error)	

Identifier	111 (0x6F)	
Name	GET_CAN_START_ID	
Description	Get the CAN sequential start ID	
Parameter	NONE	
Response:	Int32	



Identifier	112 (0x70)	
Name	SET_CAN_BAUDRATE	
Description	Set the current can baudrate	
Parameter	Int32	
	Baud rate	Identifier
	125K	125
	250K	250
	500K	500
	800K	800
Response:	ACK (success) or NACK (error)	

Identifier	113 (0x71)	
Name	GET_CAN_BAUDRATE	
Description	Get the current can baudrate	
Parameter	NONE	
Response:	Int32	
	Baud rate	Identifier
	125K	125
	250K	250
	500K	500
	800K	800
Response:	ACK (success) or NACK (error)	

Identifier	114 (0x72)	
Name	SET_CAN_DATA_PRECISION	
Description	Set the CAN output data precision	
Parameter	Int32	
	Data Precision	Identifier
	16bit Fixed point	0
	32bit Float point	1
Response:	ACK (success) or NACK (error)	

Identifier	115 (0x73)	
Name	GET_CAN_DATA_PRECISION	
Description	Get the CAN output data precision	
Parameter	NONE	
Response:	Int32	
	Data Precision	Identifier
	16bit Fixed point	0
	32bit Float point	1



Identifier	117 (0x75)	
Name	GET_CAN_MODE	
Description	Get the current can mode	
Parameter	NONE	
Response:	Int32	
	Mode	Identifier
	CANopen	0
	Sequential can	1

Identifier	118 (0x76)	
Name	SET_CAN_MAPPING	
Description	Set the current transmitted data of each can channel	
Parameter	Int32[16]	
	Int32[]	CAN Channel
	Int32[0]	Channel1
	Int32[1]	Channel2
	Int32[2]	Channel3

	Int32[15]	Channel16
	Mapping index please refer to Table 5-8	
Response:	ACK (success) or NACK (error)	

Identifier	119 (0x77)	
Name	GET_CAN_MAPPING	
Description	Get the current transmitted data of each can channel	
Parameter	NONE	
Response:	Int32[16]	
	Int32[]	CAN Channel
	Int32[0]	Channel1
	Int32[1]	Channel2
	Int32[2]	Channel3

	Int32[15]	Channel16
	Mapping index please refer to Table 5-8	



Identifier	120 (0x78)	
Name	SET_CAN_HEARTBEAT	
Description	Set the CANopen heartbeat	
Parameter	Int32	
	period	Identifier
	0.5s	0
	1s	1
	2s	2
	5s	5
Response:	ACK (success) or NACK (error)	

Identifier	121 (0x79)	
Name	GET_CAN_HEARTBEAT	
Description	Get the CANopen heartbeat	
Parameter	NONE	
Response:	Int32	
	period	Identifier
	0.5s	0
	1s	1
	2s	2
	5s	5
Response:	ACK (success) or NACK (error)	

UART / RS232 Settings Command

Identifier	130 (0x82)	
Name	SET_UART_BAUDRATE	
Description	Set the current UART / RS232 baudrate	
Parameter	Int32	
	Baud rate	Identifier
	115200	115200
	230400	230400
	256000	256000
	460800	460800
Response:	ACK (success) or NACK (error)	



Identifier	131 (0x83)	
Name	GET_UART_BAUDRATE	
Description	Get the current UART / RS232 baudrate	
Parameter	NONE	
Response:	Int32	
	Baud rate	Identifier
	115200	115200
	230400	230400
	256000	256000
	460800	460800
	921600	921600

Identifier	132 (0x84)	
Name	SET_UART_FORMAT	
Description	Set the UART / RS232 output format	
Parameter	Int32	
	Format	Identifier
	LPBUS	0
	ASCII	1
Response:	ACK (success) or NACK (error)	

Identifier	133 (0x85)	
Name	GET_UART_FORMAT	
Description	Get the UART / RS232 output format	
Parameter	NONE	
Response:	Int32	
	Format	Identifier
	LPBUS	0
	ASCII	1

Identifier	134 (0x86)	
Name	SET_UART_ASCII_CHARACTER	
Description	Set the ASCII start/stop character	
Parameter	Int8[4]	
	Byte	Parameter
	0	Start character
	1	stop character
	2	Reserved
	3	Reserved
Response:	ACK (success) or NACK (error)	



Identifier	135 (0x87)	
Name	GET_UART_ASCII_CHARACTER	
Description	Get the ASCII start/stop character	
Parameter	NONE	
Response:	Int8[4]	
	Byte	Parameter
	0	Start character
	1	stop character
	2	Reserved
	3	Reserved

Identifier	136 (0x88)	
Name	SET_LPBUS_DATA_PRECISION	
Description	Set the current UART / RS232 output data precision	
Parameter	Int32	
	Data Precision	Identifier
	16bit Fixed point	0
	32bit Float point	1
Response:	ACK (success) or NACK (error)	

Identifier	137 (0x89)	
Name	GET_LPBUS_DATA_PRECISION	
Description	Get the current UART / RS232 output data precision	
Parameter	NONE	
Response:	Int32	
	Data Precision	Identifier
	16bit Fixed point	0
	32bit Float point	1

Sensor Data Timestamp Manipulation

Identifier	152 (0x98)	
Name	SET_TIMESTAMP	
Description	Set the sensor data timestamp	
Parameter	Int32	
Response:	ACK (success) or NACK (error)	



GPS Data Transmission Commands

Identifier	160 (0xA0)		
Name	SET_GPS_TRANSMIT_DATA		
Description	Set the current transmitted data of GPS		
Parameter	Int32[2]		
	Int32[0]	Bit	Reported State / Parameter
		0	GPS NAV-PVT iTOW transmission enabled
		1	GPS NAV-PVT year transmission enabled
		2	GPS NAV-PVT month transmission enabled
		3	GPS NAV-PVT day transmission enabled
		4	GPS NAV-PVT hour transmission enabled
		5	GPS NAV-PVT min transmission enabled
		6	GPS NAV-PVT sec transmission enabled
		7	GPS NAV-PVT valid transmission enabled
		8	GPS NAV-PVT tAcc transmission enabled
		9	GPS NAV-PVT nano transmission enabled
		10	GPS NAV-PVT fixType transmission enabled
		11	GPS NAV-PVT flags transmission enabled
		12	GPS NAV-PVT flags2 transmission enabled
		13	GPS NAV-PVT numSV transmission enabled
		14	GPS NAV-PVT longitude transmission enabled
		15	GPS NAV-PVT latitude transmission enabled
		16	GPS NAV-PVT height transmission enabled
		17	GPS NAV-PVT hMSL transmission enabled
		18	GPS NAV-PVT hAcc transmission enabled
		19	GPS NAV-PVT vAcc transmission enabled
		20	GPS NAV-PVT veN transmission enabled
		21	GPS NAV-PVT veE transmission enabled
		22	GPS NAV-PVT veD transmission enabled
		23	GPS NAV-PVT gSpeed transmission enabled
		24	GPS NAV-PVT headMot transmission enabled
		25	GPS NAV-PVT sAcc transmission enabled
		26	GPS NAV-PVT headAcc transmission enabled
		27	GPS NAV-PVT pDOP transmission enabled
		28	GPS NAV-PVT headVeh transmission enabled
	29-31	Reserved	
	Int32[1]	Bit	Reported State / Parameter
		0	GPS NAV-ATT iTOW transmission enabled
		1	GPS NAV-ATT version transmission enabled
		2	GPS NAV-ATT roll transmission enabled
		3	GPS NAV-ATT pitch transmission enabled
		4	GPS NAV-ATT heading transmission enabled
		5	GPS NAV-ATT accRoll transmission enabled
		6	GPS NAV-ATT accPitch transmission enabled
		7	GPS NAV-ATT accHeading transmission enabled
		8	GPS ESF-STATUS iTOW transmission enabled
		9	GPS ESF-STATUS version transmission enabled
		10	GPS ESF-STATUS initState1 transmission enabled
		11	GPS ESF-STATUS initState2 transmission enabled
		12	GPS ESF-STATUS fusionMode transmission enabled
		13	GPS ESF-STATUS numSens transmission enabled
14		GPS ESF-STATUS sensStatus transmission enabled	
15-31		Reserved	



Response:	ACK (success) or NACK (error)
------------------	-------------------------------

Identifier	161 (0xA1)
Name	GET_GPS_TRANSMIT_DATA
Description	Get the current transmitted data of GPS
Parameter	NONE



Response:	Int32[2]																																																															
	Int32[0]	<table border="1"> <thead> <tr> <th>Bit</th> <th>Reported State / Parameter</th> </tr> </thead> <tbody> <tr><td>0</td><td>GPS NAV-PVT iTOW transmission enabled</td></tr> <tr><td>1</td><td>GPS NAV-PVT year transmission enabled</td></tr> <tr><td>2</td><td>GPS NAV-PVT month transmission enabled</td></tr> <tr><td>3</td><td>GPS NAV-PVT day transmission enabled</td></tr> <tr><td>4</td><td>GPS NAV-PVT hour transmission enabled</td></tr> <tr><td>5</td><td>GPS NAV-PVT min transmission enabled</td></tr> <tr><td>6</td><td>GPS NAV-PVT sec transmission enabled</td></tr> <tr><td>7</td><td>GPS NAV-PVT valid transmission enabled</td></tr> <tr><td>8</td><td>GPS NAV-PVT tAcc transmission enabled</td></tr> <tr><td>9</td><td>GPS NAV-PVT nano transmission enabled</td></tr> <tr><td>10</td><td>GPS NAV-PVT fixType transmission enabled</td></tr> <tr><td>11</td><td>GPS NAV-PVT flags transmission enabled</td></tr> <tr><td>12</td><td>GPS NAV-PVT flags2 transmission enabled</td></tr> <tr><td>13</td><td>GPS NAV-PVT numSV transmission enabled</td></tr> <tr><td>14</td><td>GPS NAV-PVT longitude transmission enabled</td></tr> <tr><td>15</td><td>GPS NAV-PVT latitude transmission enabled</td></tr> <tr><td>16</td><td>GPS NAV-PVT height transmission enabled</td></tr> <tr><td>17</td><td>GPS NAV-PVT hMSL transmission enabled</td></tr> <tr><td>18</td><td>GPS NAV-PVT hAcc transmission enabled</td></tr> <tr><td>19</td><td>GPS NAV-PVT vAcc transmission enabled</td></tr> <tr><td>20</td><td>GPS NAV-PVT velN transmission enabled</td></tr> <tr><td>21</td><td>GPS NAV-PVT velE transmission enabled</td></tr> <tr><td>22</td><td>GPS NAV-PVT velD transmission enabled</td></tr> <tr><td>23</td><td>GPS NAV-PVT gSpeed transmission enabled</td></tr> <tr><td>24</td><td>GPS NAV-PVT headMot transmission enabled</td></tr> <tr><td>25</td><td>GPS NAV-PVT sAcc transmission enabled</td></tr> <tr><td>26</td><td>GPS NAV-PVT headAcc transmission enabled</td></tr> <tr><td>27</td><td>GPS NAV-PVT pDOP transmission enabled</td></tr> <tr><td>28</td><td>GPS NAV-PVT headVeh transmission enabled</td></tr> <tr><td>29-31</td><td>Reserved</td></tr> </tbody> </table>	Bit	Reported State / Parameter	0	GPS NAV-PVT iTOW transmission enabled	1	GPS NAV-PVT year transmission enabled	2	GPS NAV-PVT month transmission enabled	3	GPS NAV-PVT day transmission enabled	4	GPS NAV-PVT hour transmission enabled	5	GPS NAV-PVT min transmission enabled	6	GPS NAV-PVT sec transmission enabled	7	GPS NAV-PVT valid transmission enabled	8	GPS NAV-PVT tAcc transmission enabled	9	GPS NAV-PVT nano transmission enabled	10	GPS NAV-PVT fixType transmission enabled	11	GPS NAV-PVT flags transmission enabled	12	GPS NAV-PVT flags2 transmission enabled	13	GPS NAV-PVT numSV transmission enabled	14	GPS NAV-PVT longitude transmission enabled	15	GPS NAV-PVT latitude transmission enabled	16	GPS NAV-PVT height transmission enabled	17	GPS NAV-PVT hMSL transmission enabled	18	GPS NAV-PVT hAcc transmission enabled	19	GPS NAV-PVT vAcc transmission enabled	20	GPS NAV-PVT velN transmission enabled	21	GPS NAV-PVT velE transmission enabled	22	GPS NAV-PVT velD transmission enabled	23	GPS NAV-PVT gSpeed transmission enabled	24	GPS NAV-PVT headMot transmission enabled	25	GPS NAV-PVT sAcc transmission enabled	26	GPS NAV-PVT headAcc transmission enabled	27	GPS NAV-PVT pDOP transmission enabled	28	GPS NAV-PVT headVeh transmission enabled	29-31	Reserved
		Bit	Reported State / Parameter																																																													
		0	GPS NAV-PVT iTOW transmission enabled																																																													
		1	GPS NAV-PVT year transmission enabled																																																													
		2	GPS NAV-PVT month transmission enabled																																																													
		3	GPS NAV-PVT day transmission enabled																																																													
		4	GPS NAV-PVT hour transmission enabled																																																													
		5	GPS NAV-PVT min transmission enabled																																																													
		6	GPS NAV-PVT sec transmission enabled																																																													
		7	GPS NAV-PVT valid transmission enabled																																																													
		8	GPS NAV-PVT tAcc transmission enabled																																																													
		9	GPS NAV-PVT nano transmission enabled																																																													
		10	GPS NAV-PVT fixType transmission enabled																																																													
		11	GPS NAV-PVT flags transmission enabled																																																													
		12	GPS NAV-PVT flags2 transmission enabled																																																													
		13	GPS NAV-PVT numSV transmission enabled																																																													
		14	GPS NAV-PVT longitude transmission enabled																																																													
		15	GPS NAV-PVT latitude transmission enabled																																																													
		16	GPS NAV-PVT height transmission enabled																																																													
		17	GPS NAV-PVT hMSL transmission enabled																																																													
		18	GPS NAV-PVT hAcc transmission enabled																																																													
		19	GPS NAV-PVT vAcc transmission enabled																																																													
		20	GPS NAV-PVT velN transmission enabled																																																													
		21	GPS NAV-PVT velE transmission enabled																																																													
		22	GPS NAV-PVT velD transmission enabled																																																													
		23	GPS NAV-PVT gSpeed transmission enabled																																																													
		24	GPS NAV-PVT headMot transmission enabled																																																													
		25	GPS NAV-PVT sAcc transmission enabled																																																													
		26	GPS NAV-PVT headAcc transmission enabled																																																													
	27	GPS NAV-PVT pDOP transmission enabled																																																														
	28	GPS NAV-PVT headVeh transmission enabled																																																														
	29-31	Reserved																																																														
Int32[1]	<table border="1"> <thead> <tr> <th>Bit</th> <th>Reported State / Parameter</th> </tr> </thead> <tbody> <tr><td>0</td><td>GPS NAV-ATT iTOW transmission enabled</td></tr> <tr><td>1</td><td>GPS NAV-ATT version transmission enabled</td></tr> <tr><td>2</td><td>GPS NAV-ATT roll transmission enabled</td></tr> <tr><td>3</td><td>GPS NAV-ATT pitch transmission enabled</td></tr> <tr><td>4</td><td>GPS NAV-ATT heading transmission enabled</td></tr> <tr><td>5</td><td>GPS NAV-ATT accRoll transmission enabled</td></tr> <tr><td>6</td><td>GPS NAV-ATT accPitch transmission enabled</td></tr> <tr><td>7</td><td>GPS NAV-ATT accHeading transmission enabled</td></tr> <tr><td>8</td><td>GPS ESF-STATUS iTOW transmission enabled</td></tr> <tr><td>9</td><td>GPS ESF-STATUS version transmission enabled</td></tr> <tr><td>10</td><td>GPS ESF-STATUS initState1 transmission enabled</td></tr> <tr><td>11</td><td>GPS ESF-STATUS initState2 transmission enabled</td></tr> <tr><td>12</td><td>GPS ESF-STATUS fusionMode transmission enabled</td></tr> <tr><td>13</td><td>GPS ESF-STATUS numSens transmission enabled</td></tr> <tr><td>14</td><td>GPS ESF-STATUS sensStatus transmission enabled</td></tr> <tr><td>15-31</td><td>Reserved</td></tr> </tbody> </table>	Bit	Reported State / Parameter	0	GPS NAV-ATT iTOW transmission enabled	1	GPS NAV-ATT version transmission enabled	2	GPS NAV-ATT roll transmission enabled	3	GPS NAV-ATT pitch transmission enabled	4	GPS NAV-ATT heading transmission enabled	5	GPS NAV-ATT accRoll transmission enabled	6	GPS NAV-ATT accPitch transmission enabled	7	GPS NAV-ATT accHeading transmission enabled	8	GPS ESF-STATUS iTOW transmission enabled	9	GPS ESF-STATUS version transmission enabled	10	GPS ESF-STATUS initState1 transmission enabled	11	GPS ESF-STATUS initState2 transmission enabled	12	GPS ESF-STATUS fusionMode transmission enabled	13	GPS ESF-STATUS numSens transmission enabled	14	GPS ESF-STATUS sensStatus transmission enabled	15-31	Reserved																													
	Bit	Reported State / Parameter																																																														
	0	GPS NAV-ATT iTOW transmission enabled																																																														
	1	GPS NAV-ATT version transmission enabled																																																														
	2	GPS NAV-ATT roll transmission enabled																																																														
	3	GPS NAV-ATT pitch transmission enabled																																																														
	4	GPS NAV-ATT heading transmission enabled																																																														
	5	GPS NAV-ATT accRoll transmission enabled																																																														
	6	GPS NAV-ATT accPitch transmission enabled																																																														
	7	GPS NAV-ATT accHeading transmission enabled																																																														
	8	GPS ESF-STATUS iTOW transmission enabled																																																														
	9	GPS ESF-STATUS version transmission enabled																																																														
	10	GPS ESF-STATUS initState1 transmission enabled																																																														
	11	GPS ESF-STATUS initState2 transmission enabled																																																														
	12	GPS ESF-STATUS fusionMode transmission enabled																																																														
13	GPS ESF-STATUS numSens transmission enabled																																																															
14	GPS ESF-STATUS sensStatus transmission enabled																																																															
15-31	Reserved																																																															



Identifier	162 (0xA2)
Name	SAVE_GPS_STATE
Description	Save current GPS state to the flash of GPS module
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	163 (0xA3)
Name	CLEAR_GPS_STATE
Description	Clear flash of GPS module
Parameter	NONE
Response:	ACK (success) or NACK (error)