Protocol

BWS NB2

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

Date	Version	Changes	Author	Reviewer
25/05/2024	1.0.0	Start	Samuel Dhemerson	Flávio Fernandes

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

1. Purpose

This document aims to describe the BWS NB2 device protocol for approving device communication on monitoring platforms, and to describe and detail the packets sent by the device.

2. Description

Connection	NB Iot
Location	GPS
Protocol type	UDP
Buffer	FIFO

3. Recommendations

- Read this document carefully
- Always keep your integration updated with the latest version.
- Attention to the responses required for the device.
  Consider viewing the device's configuration manual for additional information.

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

4. Protocol format

The device will send binary packets.

The device send package binary.

All packets will wait for ACK return from the gateway to proceed to the next event.

All packages wait ACK return of gateway to next event

Event	37 bytes
KeepAlive	8 bytes
ACK	8 bytes
NACK	8 bytes
Command	Check command table

Caption	
TBD	To be defined
TS	Time Stamp
Hex.	Hexadecimal
Dec.	Decimal
Bin.	Binary
Max.	Maximum
Min.	Minimum
Serial	Serial Number
CRC	Check digit
→ ←	Send Receive

Serial number

Each package is identified by a serial number, represented in hexadecimal format. The value of this number, consisting of 8 hexadecimal characters, is available on the device label as a unique identification.

Example:

Labelo of device ID: FFFFFFFF

Package send : 00FFFFFFFF05...

5. Events

Event packets are data sent periodically by the device to the server, containing crucial information for updating and monitoring the platform. Each packet consists of data collected directly from the device, reflecting both its usage and current configuration. These packages provide detailed visibility into the state of the device and any events or actions that occur during its operation.

Each event packet is generated according to the device configuration, with adjustable intervals for sending data.

The package contains,

Header			
Device type	Serial number	message ID	sequency ID

Payload													
TS GPS	Latitude	Longitude	Fix+satnum	Speed	Heading	External battery	Internal Battery	Input/ output	Hodometer	Horimeter	internal temperature	Network	CRC

Package Shipped:

Device type + Serial number+Mensagem ID+ Sequency ID + TS GPS + Latitude + Longitude + Fix + Speed +Heading +External Battery +Internal Battery + Input/output+ Hodometer + Horimeter + internal temperature + Network +CRC

The event will be sent from the device to the server, and will wait for confirmation from the server to proceed to the next message. If the response is not returned, the device will send the same package, except if the connection is lost, in which case the package will be stored in memory.

Platform Device →

Item	Size (bytes)	Description
Device type	1	Technology/Version Bit 0 – Technology Bit 1 - Technology Bit 2 – Technology Bit 3 – Technology Bit 4 – Version Bit 5 – Version Bit 6 – Version Bit 7- Version (Check table "Device type")
Serial Number	4	Value in hexadecimal, the value presented on the device label will be used for registration, being 8 digits. Example: 05E82A1C.
ID messages	1	Packet generator type (Check table "Message ID").
Sequential ID	1	The HEX packet counter is used to determine the position of the message if the limit is reached or the counter is reset in cases where the limit is reached or in cases where the unit is restarted. Example: 0x0A converting to December. We obtain the value 10, the sequential ID has a minimum value of 1 and a maximum of 255, resetting when the limit is reached.
TS GPS	4	GPS Date and Time in Timestamp (Epoch) to GMT (Human View Date and Time) without Zone Adjustment. Example: 66ED71F7 = 1726837239 20 September 2024 13:00:39 GMT
Latitude	4	Latitude signed int 32, the first bit corresponds to the determination of positive or negative value. Example: 0xFF7EECCB Dec. = -8459061 Calculation: = (-8459061/360000) = -23.49739167
Longitude	4	Longitude signed int 32, the first bit corresponds to the determination of positive or negative value.

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

		Example: 0xFEFEA520 Dec. = -16866016 Calculation: (-16866016 / 360000) = -46.85004444
Fix+sat	1	GPS Validation + Number of satellites, bit 7 is the satellite fix, being, 0 Fix OK 1 Fix NOK. The sequent bits correspond to the number of satellites that must be converted to dec. Example: 0x07 = 0000 0111 (Binary). Fix = 0 = OK (Valid Position). Satellites = 111 = 7.
Speed	1	Value in hex. The conversion to decimal must be performed. Speed (Max.: 255 km/h). Example: 0x25 In decimal = 37 km/h
Heading	2	Value in hex. The conversion to decimal must be performed. Bow steering (0 to 360). Example: 0x02 0x5D In decimal: 60.5º
External power	2	External power voltage to one decimal place in hex. Example: 0x00 0x99 In decimal: 153/10 = 15.3 V
Internal Battery	1	Internal battery voltage to one decimal place in hex. Example: 0x2A In decimal: 42 /10 = 4.2 V
Input/Output	2	Inputs and outputs, Outputs = 4 bits Inputs = 12 bits (Check the status table of entries and exits). Example: 0x00 0x05 = 0000 0000 0000 0101 = Ignition On + Accelerometer

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

Hodometer	3	Odometer in hundreds meters (Max.: 16777215) presented in hex. Example: 0x00 0x30 0xFF (12543). In decimal: 1254.3 km
Horimeter	3	Hourimeter in minutes displayed in hex. Example: 0x00 0x0B 0x40 In decimal: 2880 minutes.
Internal temperature	1	Internal temperature in hex being used in unit celsius. Example: 0x1D In decimal: 29 °C
Network info	1	Available signal displayed in hex. 99 – No signal0- Low signal. 31 - High Signal. Example: 0x16 In decimal: 22
CRC	1	I type package integrity checker (Check item CRC).

Example:
Device sends→:

0005E82A1C011E66ED71F7FF813AE0FEFFB34C0702025D00992A000101C2C800001C1D1617

Item	Value	
deviceType	00	NB Technology Initial Release
SerialNumber	05E82A1C	ID: 05E82A1C
Message ID	01	Update by time.
SequencyID	1E	Sequential position: 30
TSGPS	66ED71F7	Converting time stamp to UTC, we get 2024-09-20 10:00:39.
Latitude	FF813AE0	Int32 FF813AE0 = -8308000 -8308000/360000 Result: -23.07777777
Longitude	FEFFB34C	Int32 FEFFB34C = -16796852 -16796852/360000 Result: -46.65792222
Sat_Fix	07	Converting to binary we get, 00000111, we consider the seventh bit as fix, in this example, FIX =OK Satellites = 7
Speed	20	Converting to decimal = 32 km/h.
Angle	025D	Converting to decimal we get 605 dividing by 10, we get the value of 60.5º azimuth.
Ex_Batery	0099	Converting to decimal, we get 153, dividing by 10, we get the result of 15.3V.
In_Batery	2A	Converting to decimal, we get 42, dividing by 10, we get the result of 4.2V.
Status In/out	0001	In binary corresponds to 0000 0000 0000 0001 = Ignition on.
Hodometer	01C2C8	Converting to dec, 01C2C8 = 115400 meters
Horimeter	00001C	Converting to decimal, 28 minutes.
internal temperature	1D	Converting to decimal, 29º C.
Network info	16	Converting to decimal we arrive at 22 (Good sign).
CRC	17	17

Platform responds ACK:	3505E82A1C401E51

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

6. Device type

Matches device model and version.

Model	Version	Device type
BWS NB2	Initial	0x00 = 0000 0000

7. Message ID

The Message ID is a key field present in the header of every data packet sent from the device to the server and from the server to the device. It serves as a unique identifier that determines the type of message generator being transmitted.

Below is the ID description table.

Item	ID (Hex)	Description
Position by time	0x01	Sent upon reaching the preset time for location update
TBD	0x02	TBD
Position by angle	0x03	Made available when bow changes occur.
Event Position	0x04	Sent in case of events.
Ignition On Position	0x05	Sent when ignition is switched on
Ignition off position	0x06	Sent when ignition is switched off
Main power disconnected	0x07	Shipped when main power is cut off.
Main power reconnected	0x08	Sent when the device is repowered.
Interference Detected (Jammer)	0x0B	Jammer Alarm Detected
Signal resumed (Jammer presence ended).	0x0C	Jammer alarm reset.
Active Output 1	0x1D	Active Output 1 (Lockout)
Output 1 disabled	0x1E	Output 1 Disabled (Unlock)
Sleep onset	0x11	Sent whenever sleep starts
Wake start	0x12	Sent when Wake is started
Speed exceeded	0x27	Sent when the set speed limit is reached.
Tracker breached	0x34	Movement with disconnected power
Speed restored	0x35	Speed limit resumed
Keep alive	0x3f	Keep alive
ACK	0X40	Sent when the message is correct
NACK	0x41	Sent when the message is incorrect

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

8. Input/Output Table

LSB = least significant bit.MSB = most significant bit.

Consider the table below to be 1st bit LSB and 16th bit MSB.

Status In/out		
1st bit	Ignition	0 = Off
		1 = On
2nd bit	Interrupted power	0 = Off
		1 = On
3rd bit	Accelerometer	0 = Off
		1 = On
4th bit	Jammer	0 = Off
		1 = On
5th bit	IN 2	0 = Off
		1 = On
6th bit	TBD	
7th bit	TBD	
8th bit	TBD	
9th bit	TBD	
10th bit	TBD	
11th bit	TBD	
12th bit	TBD	
13th bit	Cut fuel	0-Off
		1-ON
14th bit	TBD	
15th bit	TBD	
16th bit	Buffer	0 = Online
		1 = Memory

9. Keep alive

The **Keep Alive** package is essential to ensure that the connection between the device and the server remains active and stable throughout the communication period. It is periodically sent by the device to prevent the connection from being terminated due to inactivity. This packet serves as a "live" or "presence" signal, ensuring that the server recognizes that the device is still connected and operating properly.

Like other data packets sent by the device, Keep Alive requires a return from the server, which validates that communication continues to work bidirectionally.

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

The device waits 30 seconds for the server's response to the packet being sent.

is in sleep it will respect the economy mode, being able to send packages in sleep in the least economical mode and not send keep packages in the most economical mode.

Item	Size (bytes)	Description
Device type	1	Device model + device version
Serial Number	4	Device identifier
Message ID	1	Type of packet sent, (see Message ID)
Sequency ID	1	Packet Counter
CRC	XX	Digit Checker

Example:
Device Send →, 0005E82A1C3F3BE3

Device →		
Item	Value (0x)	Description
Device type	00	Model NB2 + Initial Version
serial number	05E82A1C	Identifier: 05E82A1C
message ID	3F	Message Type, Keep Alive
sequency ID	3B	Sequential position number, 59
CRC	E3	E3 (Calculated)

Platform Answer←: 0005E82A1C40881C

Platform ←		
Item	Value (0x)	Description
Device type	00	Model NB2 + Initial Version
serial number	05E82A1C	Identifier: 05E82A1C
message ID	40	Message type, ACK.
sequency ID	3B	Sequential position number, 59
CRC	1C	

10. CRC

The CRC is responsible for analyzing the integrity of the packet sent, that is, it monitors whether all bytes arrived correctly filled, for this calculation the polynomial 8 is used, below examples of the CRC calculation.

C	Int8_t dallas_crc8(Int8_t*date, int32_t size)
	{
	Int8_t crcr = 0;
	
	for (int32_t i = 0; i < size; ++i)
	{
	Int8_t inbyte = date[i];
	for (Int8_t j = 0; j < 8; ++j)

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

	{
	Int8_t mix = (crcr^inbyte) & 0x01;
	crcr = (crcr >> 1);
	if (mix) crcr = (crcr ^ 0x8C);
	inbyte = (inbyte >> 1);
	}
	}
	return crcr;
	}

PHP	function dallas_crc8($data) {
	$crcr = 0;
	
	for ($i = 0; $i < count($data); ++$i) {
	$inbyte = $data[$i];
	for ($j = 0, $j < 8, ++$j) {
	$mix = ($crcr^$inbyte) & 0x01;
	$crcr = ($crcr >> 1);
	if ($mix) {
	$crcr = ($crcr^0x8C);
	}
	$inbyte = ($inbyte >> 1);
	}
	}
	return $crcr;
	}

Python	def dallas_crc8(date):
	crcr = 0
	
	for byte in data:
	inbyte = byte
	for _ in range(8):
	mix = (crcr^inbyte) & 0x01
	crcr = crcr >> 1
	if mix:
	crcr = crcr^ 0x8C
	inbyte = inbyte >> 1
	return crcr

TypeScript	function dallasCrc8(data: number[]): number {
	let crcr = 0;
	
	for (let i = 0; i < data.length; ++i) {
	let inbyte = data[i];
	for (let j = 0; j < 8; ++j) {
	const mix = (crcr^inbyte) & 0x01;
	crcr = crcr >> 1;

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

	if (mix) {
	crcr = crcr^0x8C;
	}
	inbyte = inbyte >> 1;
	}
	}
	return crcr;
	}

JavaScript	function dallasCrc8(data) {
	let crcr = 0;
	
	for (let i = 0; i < data.length; ++i) {
	let inbyte = data[i];
	for (let j = 0; j < 8; ++j) {
	const mix = (crcr^inbyte) & 0x01;
	crcr = crcr >> 1;
	if (mix) {
	crcr = crcr^0x8C;
	}
	inbyte = inbyte >> 1;
	}
	}
	return crcr;
	}

11. ACK/NACK

ACK and NACK are response packets that indicate whether the received packet is correct or not.

These responses must be sent by the server whenever the device sends packet.

ACK and NACK are part of the communication and can occur on both sides.

Platform Device →

Platform Device ←

The ACK contains ID **0x40** indicating that the message was successfully processed.

The ACK packet must be sent each time a packet is sent by the device (except command response from the device) and is correct. To do this, validate the CRC, it indicates if the contents of the packet are correct, if the calculation is successful the ACK must be sent.

ACK		
Item	Size (bytes)	Description
Device type	1	Device Model
Serial Number	4	Device identifier
Message ID	1	Type of package shipped
Sequency ID	1	Packet Counter
Comand type	1	Command Type
CRC	1	Digit Checker

The NACK contains ID **0x41** indicating that the message was not successfully processed.

Use the CRC to validate the packet, if it arrives a different value than expected return NACK to the device.

NACK		
Item	Size (bytes)	Description
Device type	1	Device Model
Serial Number	4	Device identifier
Message ID	1	Type of package shipped
Sequency ID	1	Packet Counter
Comand type	1	Command Type
CRC	1	Digit Checker

The device will also return NACK in cases where the packet received is not as expected or the ACK is incorrect.

12. Buffer

The device will hold temporary storage for up to 4 recent packages, which will have priority sending. If it is not possible to send these packets, they will be transferred to Flash memory, with the capacity to store up to 10000 positions. In Flash memory, the packets will be organized and unloaded following FIFO (First-In-First-Out). Packets stored in Flash will be marked as 'buffer' in the status and will wait for confirmation from the server via an ACK.

13. Commands

The formation of commands is defined by type of command, namely, action, consultation and definition

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

The commands below follow the direction, Platform → Device.

Example Speed limit:

Device Type + Device ID + Command ID +Counter+ Table + Subtable + Value + CRC

3500BC614E440006025A56

Command ID		
Command	ID (x0)	Description
Action	42	Every command of which has direct action.
Check	43	Any command that aims to change parameters.
Define	44	Check status and/or configuration.

Table		Sub-table	
Connection	0x02	All network parameters	0x00
Time	0x03	All time	0x00
		Moving Time	0x01
		Update time in event	0X02
		Downtime	0x03
		Time to keep alive	0x04
TBD	0X04	TBD	-
		TBD	-
Odometer	0X05	Odometer	0x00
Features	0x06	All sensitivities	0x00
		Angle	0x01
		Speed	0x02
		Sensitivity on	0x03
		Sensitivity Off	0x04
IMEI/ICCID Reading	0x07	Read IMEI	0x00
Reading service	0x08	Read Service Information	0x00
Reading Settings	0x09	Read all settings	0x00
Read firmware version	0x0A	Read firmware version	-

type	Item		Payload(bytes)	Command	Description
Action	Cut fuel ON	→	9	00_Serial_420000_CRC_	Enable Output 1 Example command confirmation response←: 00__Serial__420500__CRC__ 0x42-Command ID 0x05 -Counter 0x00 - Table
	Cut fuel OFF	→	9	00_Serial_420001_CRC_	Disable Output 1

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

					Example command confirmation response←: (9 bytes) 00__Serial__420601__CRC__ 0x42-Command ID 0x06 - Counter 0x01 - Table
	Restart	→	9	00_Serial_420002_CRC_	Restart the device. Example command confirmation response←: (9 bytes) 00__Serial__420702__CRC__ 0x42-Command ID 0x07 - Counter 0x02 - Table
	Clear buffer	→	9	00_Serial_420030_CRC_	Clear External Flash. Example command confirmation response←: (9 bytes) 00__Serial__420103__CRC__ 0x42-Command ID 0x01 - Counter 0x03 – Table Sample response: 00__SERIAL__420030__CRC__
	Odometer	→	13	00__SERIAL__42000500(Value in hex 3 bytes)__CRC__	Initial odometer, where 3 bytes (0 - 16777215 hundreds of meters) in hex. Example command: 1254.3 km = 12543 metres 00__SERIAL__420005000030FF__CRC__ Example command confirmation response←: 00__SERIAL__4201__CRC__
	Hourimeter	→	13	00__SERIAL__42000600(Value in hex.)__CRC__	Initial hourimeter, being 3 bytes (0 - 16777215 minutes) in hex. Example command confirmation response←:

					00__SERIAL__4202__CRC__

Kind	Item		Payload(bytes)	Command	Description
Definition	All update times	→	14	00__SERIAL__44000300(TN,TS,TW,TK)__CRC__	TN = Moving update time. TS = Sinister Update time TW= No movement. TK = Keepalive Update Time. All values must be sent in hex. Example command confirmation response←: 00__Serial__441300__CRC__ 0x44-Command ID 0x13 - Counter 0x00 - Table
	Moving Time	→	12	00__SERIAL__44000301(value in hex)__CRC__	Update time on the move, being 2 bytes (5 -65535 seconds) in hex. Example command confirmation response←: 00__Serial__441303__CRC__ 0x44-Command ID 0x13 -Counter 0x03 - Table
	Set update time on emergency	→	12	00__SERIAL__44000302(value in hex)__CRC__	Sets the update time when the device is in emergency. Value in seconds. Example command confirmation response←: 00__Serial__441303__CRC__ 0x44-Command ID 0x13 -Counter 0x03 - Table

	Time no moviment	→	12	00__SERIAL__44000303(value in hex)__CRC__	Update time when static, being 2 byte (0-65535 minutes) in hex. Example command confirmation response←: 00__Serial__441403__CRC__ 0x44-Command ID 0x14 -Accountant 0x03 - Table
	Keepalive time	→	12	00__SERIAL__44000304(value in hex.)__CRC__	Keep alive update time, being 2 byte (0-65535 seconds) in hex. Example command confirmation response←: 00__Serial__441403__CRC__ 0x44-Command ID 0x14 -Accountant 0x03 - Table
	Angle	→	12	00__SERIAL__44000601(Value in hex.)__CRC__	Refresh angle, being 2 bytes (0-360º) in hex. Example command confirmation response←: 00__Serial__441706__CRC__ 0x44-Command ID 0x17 -Accountant 0x06 - Table
	Speed	→	12	00__SERIAL__44000602(Value in hex.)__CRC__	Speed limit, where 1 byte (0 – 255) in hex. Example command confirmation response←: 00__Serial__441606__CRC__

					0x44-Command ID 0x16 -Accountant 0x06 - Table

type	Item		Payload (bytes)	Command	Description
Check	Consult IMEI/ICCID	→	10	00__SERIAL__43000700__CRC__	Checks the IMEI and ICCID of the device in use, Hex text. and must be converted to ASCII. Example answer, 35__Serial__431207 ICCID,IMEI__CRC__ 43 – Command ID 12 – Counter 07 – Table Content = The content is in comma-separated hex text. Example 89554000000353211568,860197070012169
	Check all times	→	10	00__SERIAL__43000300__CRC__	Checks the times set on the device, being, 2 bytes - Time in motion. 2 bytes -Time in emergency. 2 bytes -No moviment. 2 bytes – Keep alive time. Return example: (17 bytes) 00__Serial__43010300001e001e001e003c__CRC__ 43 – Command ID 01 – Counter 03 – Table 00 – Subtable 001E- Moving Time (Sec.) 001E-Time in emergency (Sec.) 001E-Stopped Time (Min.) 003C-Keepalive Time (Sec.)
	Check Movement Time	→	10	00__SERIAL__43000301__CRC__	Queries the update time of the device on the go, being 2 bytes. Sample response: (12 bytes)

					00__Serial__43020301001e_CRC__ 0x43- Command ID0x02-Counter0x03-Table0x01-Subtable0x001E- Movement Time (Seconds)
	Check Time no moviment	→	10	00__SERIAL__43000302__CRC__	Checks the update time of the device when stopped. Sample response←: (12 bytes) 00__Serial__43030302001e__CRC__ 0x43 – Command ID 0x03 – Accountant 0x03- Table 0x02- Sub-table 0x001E- Downtime (min.)
	Check emergency time		10	00__SERIAL__43000303__CRC__	Checks the update time when the device is emergency. Example answer ←: (12 bytes) 00__Serial__43040303001e__CRC 0x43 – Command ID 0x04 – Counter 0x03- Table 0x02- Sub-table 0x001E- Emergency time (Min.).
	Check keep alive time		10	00__SERIAL__43000304__CRC__	Check the device's keep alive time. (12 bytes) Example answer ←: 00__Serial__43050304003c__CRC 0x43- Command ID 0x05- Accountant 0x03- Table 0x04- Sub-table 0x003C- Keepalive time (Sec.).
	Check sensitivity settings	→	10	00__SERIAL__43000600__CRC__	Checks the values set for the sensitivities, being, 2-bytes-Angle 2 bytes – Speed limit

					1 bytes- Sensitivity when awake 1 bytes – Sensitivity when in sleep Sample response←: (12 bytes) 00__Serial__430a0600001e00640 20300960032002d0201__CRC__ 0x43 – Command ID 0x0A- Counter 0x06- Table 0x00- Sub-table 0x001E- Angle of new package 0x0064- Speed limit 0x02- Ignition sensitivity on 0x03- Ignition Off Sensitivity
	Check Upgrade Angle	→	10	00__SERIAL__43000601__CRC__	Queries the update angle set on the device getting response as in the example below. Sample response←: (12 bytes) 00__Serial__43060601002D__CRC 0x43 – Command ID 0x06- Counter 0x06- Table 0x01- Sub-table 0x002D- Refresh Angle
	Check Speed limit	→	10	00__SERIAL__43000602__CRC__	Query the speed limit to generate alerts. Sample response←: (12 bytes) 00__Serial__430706020064__CRC 0x43 – Command ID 0x07- Counter 0x06- Table 0x02- Sub-table 0x0064- Speed limit (KM/h)

	Check Accelerometer with Ignition On	→	10	00__SERIAL__43000603__CRC__	Check sensitivity with ignition on. Sample response←: (11 bytes) 00__Serial__4308060302__CRC 0x43 – Command ID 0x08- Counter 0x06- Table 0x03- Sub-table 0x02- Ignition On Sensitivity
	Check accelerometer with ignition off	→	10	00__SERIAL__43000604__CRC__	Check the sensitivity of the accelerometer with ignition off. Sample response←: (11 bytes) 00__Serial__4309060403__CRC 0x43 – Command ID 0x09- Counter 0x06- Table 0x04- Sub-table 0x03- Ignition Off Sensitivity
	Check connection parameters	→	10	00__SERIAL__43000800__CRC__	Connection parameters being, hex text. Needing to convert them into ASCII being, IP1,DNS1:port,IP2,DNS2:port,APN,User,Pass Sample response←: (bytes) 00__Serial__4309080403__Conteudo__CRC 0x43 – Command ID 0x0b- Counter 0x08- Table Content = IP1,DNS1:PORT (HEX),IP2,DNS2:PORT(HEX),APN, USER,PASSWORD

BWS IoT
Calçada das Acácias, 31 - Alphaville Comercial
Centro comercial, Barueri – SP
Tel.: +55 11 4191 – 7482 / 4193 – 1475/ 94017-4266
E-mail: suporte@bwsiot.com
Site: www.bwsiot.com

	Check firmware version	→	10	00__SERIAL__43000A__CRC__	Check the firmware version of the device. Sample response←: 00_Serial_43030a425753496f745f4e42325f565f302e305f2831302f30332f323529_CRC_ 0x43-Command ID0x03 – Counter 0x0A- Table Contents – Firmware Version