

UBX_Speed Command for ADR

GADN/ANNA Series with u-blox NEO-M8L, ZED-F9R

V1.0

Oct 3, 2020



ANNA-FxxLx

Full-sized mPCIe



ANNA-F9xRx Series

M.2 2242 B key



GADN-FxxLx Series

Full-sized mPCIe

ANTZER TECH CO., LTD.

Website : www.antzer-tech.com

Facebook : www.facebook.com/ANTZERTECH/

Tel : +886-2-7729-9223

Email : sales@antzer-tech.com

Office : 7F-7, No.237, Sec. 1, Datong Rd., Xizhi Dist., New Taipei City 221 Taiwan

Important User Information

Liability

Every care has been taken in the preparation of this document. Please inform ANTZER TECH CO., LTD. any inaccuracies or omissions. The data and illustrations found in this document are not binding. We, ANTZER TECH CO., LTD. reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and therefore should not be considered as a binding description of the range of functions (neither for future product versions). ANTZER TECH CO., LTD. assumes no responsibility for any errors that may appear in this document.

There are many applications of the described product. Those responsible for the use of this device must ensure that all the necessary steps have been taken to verify that the applications meet all performance and safety requirements including any applicable laws, regulations, codes, and standards.

ANTZER TECH CO., LTD. will under no circumstances assume liability or responsibility for any problems that may arise as a result from improper use or use that is not in accordance with the documented features of this product.

The examples and illustrations in this document are included solely for illustrative purposes.

Intellectual Property Rights

ANTZER TECH CO., LTD. has intellectual property rights relating to technology embodied in the product described in this document. These intellectual property rights may include patents and pending patent applications in Taiwan and other countries.

Copyright © 2020 ANTZER TECH CO., LTD. All rights reserved.

Antzer Tech In-Vehicle Module User Manual

Table of Contents

1. UBX Command for Sending Speed	4
2. Example	6
3. Sample Code	7
4. References	8

Change History

Version	Date	Author	Description
1.0	2020/10/30	Vincent Cheng	First version release

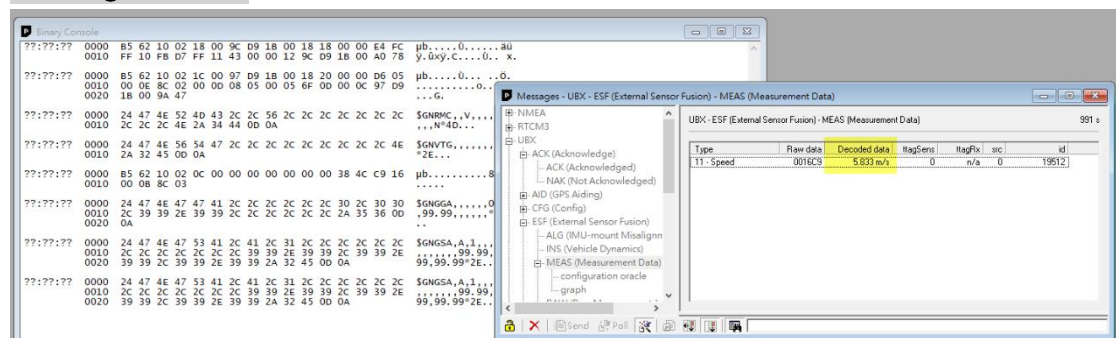
1. UBX Command for Sending Speed

- The number of the **timeTag** should be added up as the counter of the data

Header	Class	ID	Length (Bytes)	timeTag	flags	id	speed (signed)	Data Type	checksum
0xB5 0x62	0x10	0x02	8+N*4	0xFFFFFFFF	0x0000	0xFFFF	0xFFFFFFFF	0x0B	CK_A CK_B

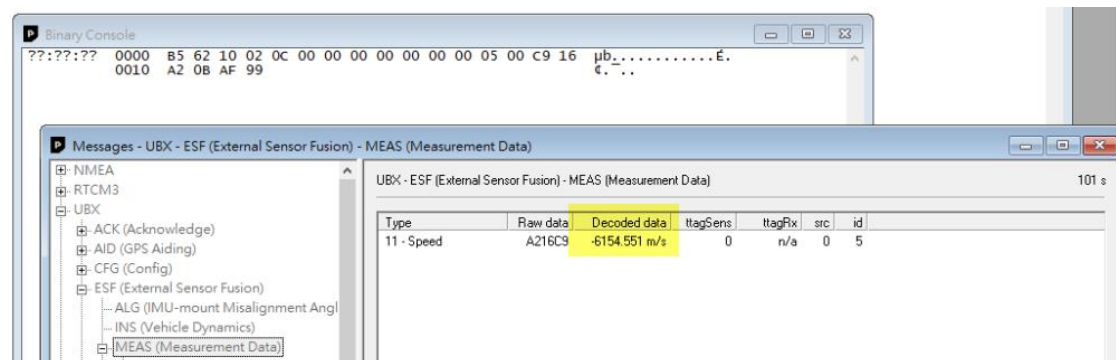
- If the car is moving backward, you should use negative number (e.g. -10 m/s).

[Moving Forward]



The screenshot shows the Binary Console window with a hex dump and ASCII representation of a UBX message. The Messages window displays a tree view of the message structure, including NMEA, RTCM3, UBX, and MEAS (Measurement Data). The MEAS window shows a table with columns: Type, Raw data, Decoded data, ttagSens, ttagFix, src, and id. The data row shows '11 - Speed' with a raw data value of '0016C9' and a decoded data value of '5.633 m/s'.

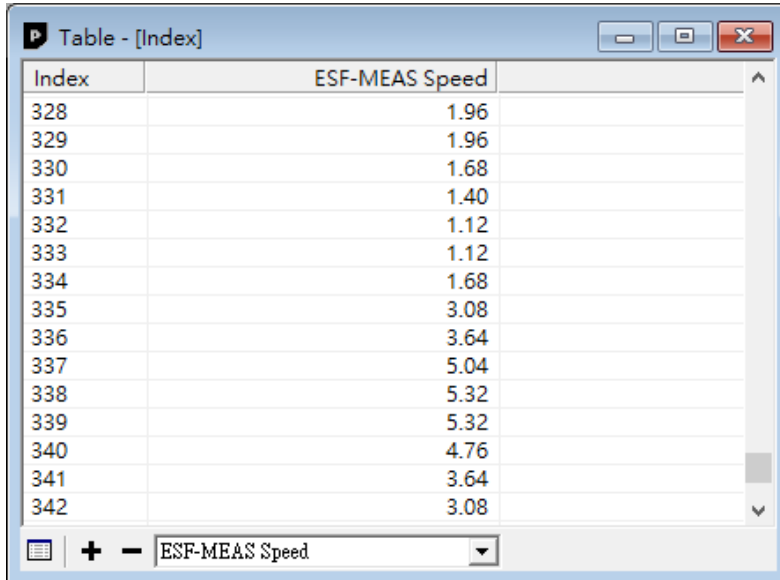
[Moving Backward]



The screenshot shows the Binary Console window with a hex dump and ASCII representation of a UBX message. The Messages window displays a tree view of the message structure, including NMEA, RTCM3, UBX, and MEAS (Measurement Data). The MEAS window shows a table with columns: Type, Raw data, Decoded data, ttagSens, ttagFix, src, and id. The data row shows '11 - Speed' with a raw data value of 'A216C9' and a decoded data value of '-6154.551 m/s'.

- The time interval between speed commands must be less than 100ms.

- You could also check the speed information in the Table View of the u-center v20.06.1 or above.



Index	ESF-MEAS Speed
328	1.96
329	1.96
330	1.68
331	1.40
332	1.12
333	1.12
334	1.68
335	3.08
336	3.64
337	5.04
338	5.32
339	5.32
340	4.76
341	3.64
342	3.08

2. Example

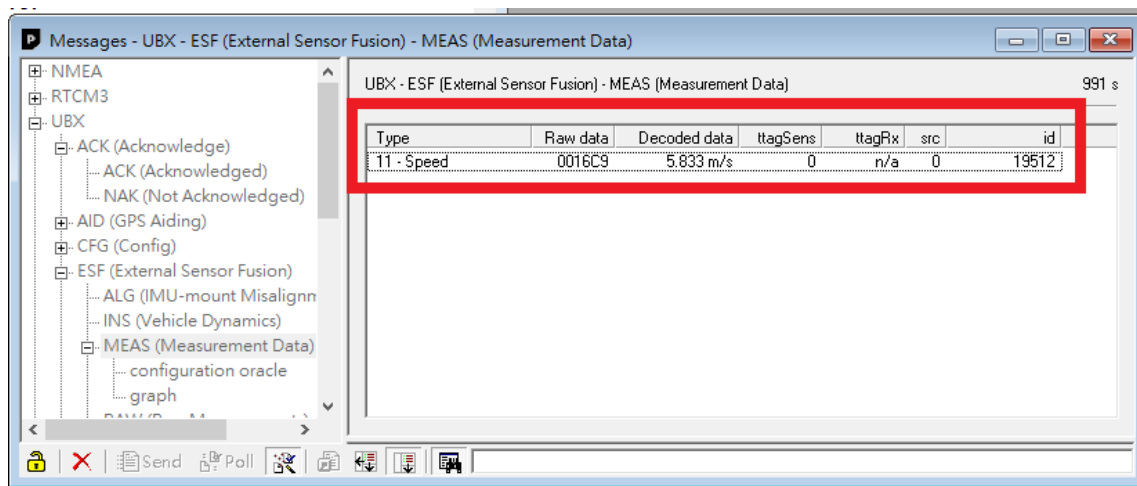
Speed : 5.833m/s

Speed = 5833 = 16C9

Command:

B5 62 10 02 0C 00 00 00 00 00 00 00 38 4C C9 16 00 0B 8C 03

B5 62 10 02 0C 00 01 00 00 00 00 00 38 4C C9 16 00 0B 8D 0F



3. Sample Code

```
void Send_UBX_ESF_MEAS_Datatype11(void)
{
    unsigned char ubx[30] = {0};
    TYPE_BIT32 temp;

    ubx[0] = 0xB5; // Preamble
    ubx[1] = 0x62; // Preamble
    ubx[2] = 0x10; // Class
    ubx[3] = 0x02; // ID
    ubx[4] = 0x0C; // Length (low) Length=8+N*4 = 8+4 = 12
    ubx[5] = 0; // Length (high)

    temp.dword = Count_TimeTick;
    ubx[6] = temp.byte.B0; //Byte 0
    ubx[7] = temp.byte.B1; //Byte 1
    ubx[8] = temp.byte.B2; //Byte 2
    ubx[9] = temp.byte.B3; //Byte 3

    ubx[10] = 0; // flags (low)
    ubx[11] = 0; // flags (high)
    ubx[12] = 0x05; // id (low)
    ubx[13] = 0; // id (high)

    temp.dword = Speed //unit: m/s / 1000
    ubx[14] = temp.byte.B0; //Byte 0
    ubx[15] = temp.byte.B1; //Byte 1
    ubx[16] = temp.byte.B2; //Byte 2
    ubx[17] = 0x0B; // Data Type
    ubx[18] = CK_A //checksum
    ubx[19] = CK_B //checksum
}
```



```
checksum:
//=====

void ubx_checksum(uint8_t* ubxcmd, uint16_t Length)
{
    int i;

    uint8_t CK_A = 0;

    uint8_t CK_B = 0;

    for(i = 2; i < (Length - 2); i++)
    {
        CK_A = CK_A + ubxcmd[i];

        CK_B = CK_B + CK_A;
    }

    ubxcmd[Length - 2] = CK_A;

    ubxcmd[Length - 1] = CK_B;
}
```

4. References

- u-blox8-M8_ReceiverDescrProtSpec_(UBX-13003221)_Public.pdf

<https://www.u-blox.com/en/product/neo-m8-series#tab-documentation-resources>



Interface Manual

Products

Support

Beyond

Investors



u-blox 8 / u-blox M8 Receiver Description, Protocol Specification

28-May-2019

3.49 MB

 PDF Download

System/software design

- 30.3.4 Sensor Data Types
- 33.11.2 UBX-ESF-MEAS (0x10 0x02)