RIFA-M Multifunctional Vehicle Tracking Unit

USER MANUAL

Version 2.1.0

June 22th, 2019

ANTZER TECH CO., LTD.
11F-1, No.475, Guangfu S. Rd., Xinyi Dist., Taipei City 11074,
Taiwan (R.O.C.)
Tel: +886-2-2723-3131
www.antzer-tech.com

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.

Antzer Tech RIFA Series User Manual

Table of Contents

1. Product Description

1.1. Introduction

Antzer-Tech Automotive-Grade RIFA-M Multifunctional vehicle tracking unit provides a compact, economic, easily installed solution for fleet management. RIFA's self-designed firmware not only supports OBDII & J1939 protocols, but integrates superior cellular modem, highly sensitive GPS, accelerometer (G-sensor) in one end-to-end solution. With built-in antennas for both GPS and Cellular modules, RIFA engineers a truly robust connectivity. RIFA-M is combining with the most comprehensive and economical vehicle diagnostics technology and multi-functional I/Os, which provides real-time engine monitoring and GPS location. The engine diagnostic data is collected through the vehicle's OBD-II/J1939 communication port and is transmitted via cellular network to the back-end center. In this way, potential engine problems can be identified earlier before the vehicle breaks down at an inopportune time. Furthermore, you may configure other advanced driving behavior events such as harsh braking, sudden acceleration, speeding, cornering, and much more in order to reduce the risks of vehicle damage and drive down the costs of fuel. This user manual is intended to guide you through the installation and configuration process. Also, you can select different accessories such as TPMS (tire pressure monitoring sensor), NFC card reader, temperature and humility sensor, and emergency button which is compatible with RIFA-M.

1.2. Document History

Version	Date	Author	Description
1.3	22-Nov-18	Leopold Chen	First version of this document
2.1	22-Jun-19	Leopold Chen	Second version

1.3. Hardware Specification

General	Vehicle Interface	20pin MicroFit Connector
	Messages	12,000 Buffered Messages
	Geofencing	Geofences Zones (Polygon, Rectangular or Circle Setting)
	Configuration	AT command or configurator under Microsoft Windows
Vehicle Network	Protocol Support	ISO15765-4 On-Board Diagnostic, J1939 and Mobileye® Protocol (reserved)
Cellular Network	Frequency Band (WWAN Module Selected)	GSM/GPRS: 850,900,1800,1900 Mhz
		HSPA/UMTS : 800,850,900,1700,1900,2100 Mhz
		LTE Cat 1 : Band 4, 13 or Band 3,7,20
		LTE Cat M1/NB 1 : Band 2,3,4,5,8,12,13,20,28
		LoRa[1]: US915,EU868,AS923
	Data Protocol	TCP, UDP, MQTT(reserved) or LoRaWAN(reserved)
Wireless Network	Bluetooth	2.4GHz Low Energy Bluetooth Class 2
GPS	Chipset	Ublox Neo M8 Engine, 72Channels support GPS, Galileo, GLONASS, BeiDou
	Dead Reckoning	Optionally Support UDR, Tracking with GPS Signal Loss
Sensor		3-Axis G-sensor with Auto-Calibrating Function
LED Indicator		x2 LEDS (2 Colors each for GPS, Cellular Network, and System Status)
SIM Card	Form Factor	Mini SIM 2FF (25 x 15 mm)
Comprehensive I/O	MicroFit Connector	Dual CAN Ports for OBDII, J1939 or Mobileye® Signal Input
		x1 RS-485 and x1 RS-232
		x2 Digital Input and x2 Digital Output (100mA)
		x1 Analog Input (under 32V) and x1 1-wire Interface
		x1 Ignition Input Detect
Audio		x1 Mic-in and x1 Headphone-out (resvered)
Configuration Port		Micro USB with 5V Power Input (without battery)
Power	Power Input Range	9~32V DC Power Input from Micro Fit Connector
	Power Mode	Operating Mode, Sleep Mode, and Backup Battery Mode
	Min. Power Consumption	<4.5 mA @ 12V (Sleep mode), Support CAN Wakeup Function
Buzzer		Built-in Buzzer for System Status
Environment	Operating Temperature	-30 to 70 °C (Without Battery); -20 to 70 °C (With Battery)

	Storage Temperature	-40 to 85 °C
	Certificate and Vibration	CE, FCC, RoHS and MIL-STD-810G 514.6
	Humidity	10% to 90% R.H. (Non-Condensing) Compliant
Dimensions		86 x 56 x 28 mm
Battery	Lithium Ion Battery	Optionally support Built-in 3.7V 130mAh Battery (standard for emergency service)

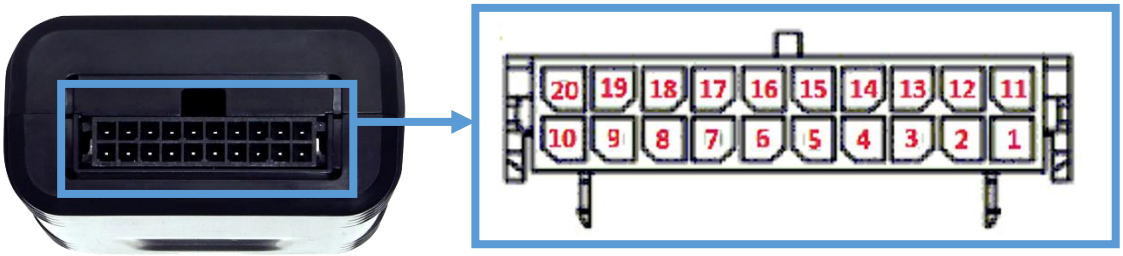
1.External antenna is needed

2. Hardware Feature

2.1. MicroFit Connector

For the standard configuration, Antzer-Tech RIFA-M can support ISO 15765-4, SAE J2284 and SAE J1939 on MicroFit by pin 2/12 and pin 3/13 with CAN High/Low signal. Also, RIFA-M supports wide range power input from 9V to 32V by pin 1/11 and ignition input by pin 4. Furthermore, RIFA-M provides 2xDI, 2xDO, 1xRS485, 1xRS232, 1x Analog in, 1x one-wire, 1x buzzer out and 1xVDC(5V) out.The pin definition is as the below table.

RIFA-M



Pin No.	Define	Pin No.	Define
1	VDC In	11	VDC Ground
2	CAN1 Low	12	CAN1 High
3	CAN2 Low	13	CAN2 High
4	Ignition In	14	DI 1
5	DI 2	15	Buzzer Out
6	DO 1	16	DO 2
7	RS-485 D+	17	RS-485 D-
8	RS-232 RX	18	RS-232 TX
9	VDC Out (5V)	19	VDC Ground
10	Analog In	20	1-Wire

2.2. Micro-USB Connector

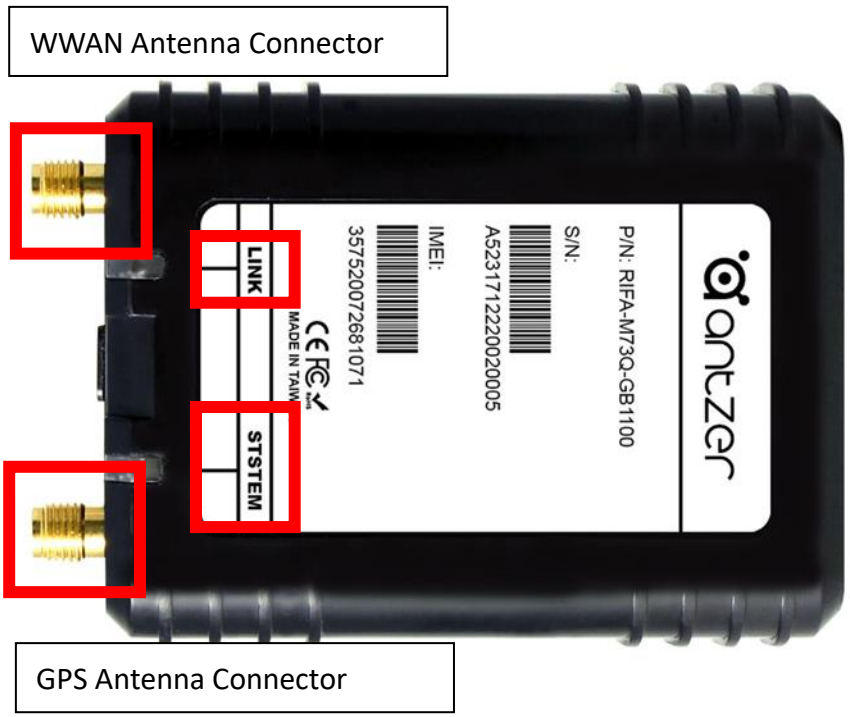
There is a micro-USB port on the RIFA-M for configuration usage. Connecting it with your PC or notebook USB port can power up RIFA-M for configurator.

Run the RIFA-M Configuration Tool or AT command to execute the settings on RIFA-M.

Warning : RIFA-M is powered up by USB (0.8A@5V) , for this mode , RIFA-M is only permitted for configuring. For real application , please connect power supply via Micro-fit connector over 12V.

2.3. LED Indication and External Antenna Connector

There is a LED indicator of RIFA-M to show the status of WWAN(cellular network),GPS and System power.



LINK(LED1) Yellow Light: to show RIFA's WWAN module works fine.

LINK(LED1) Green Light: to show RIFA's GPS module works fine.

SYSTEM(LED2) Red Light: to show RIFA's power-input works fine.

There are one external WWAN antenna connector nearby LINK LED(RP-SMA Female) and one external GPS antenna connector nearby SYSTEM LED (RP-SMA Female) equipped in RIFA-M

2.4. Buzzer Operation

There is a buzzer equipped on RIFA to indicate (1) System reboot (2) Waring on unsafe driver behavior (3) Waring on driving to restricted area (Geofence) (4) GPS signal detection.

Beep * 1 : Power / System On

Beep * 3 : WWAN Access Point Connection

Beep * 4 : GPS Satellite Connect

2.5. SIM Card Slot

User can insert the SIM card (MiniSIM,2FF) to RIFA's SIM card slot showing by below photo.

2.6. Backup Battery

The RIFA-M is optionally equipped a backup battery which allows RIFA-M keep operating for about 5~8 min emergency service. and sending the alarm to central management center at a power loss. However, you may not want to enable the battery after testing. You can turn off the battery power by removing the jumper and turn the battery power (for charge or discharge) by connecting the jumper.

Below photo shows the jumper's location on RIFA

3. Configuration

Antzer Tech supply a simplicity and clarity configuration tool or AT command which is so user-friendly. RIFA collects the common setting by default that customer only need to define the impactive setting.

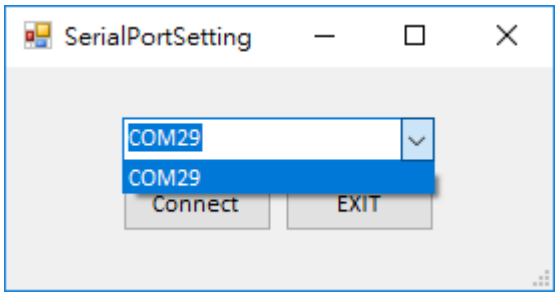
3.1. USB Driver Installation

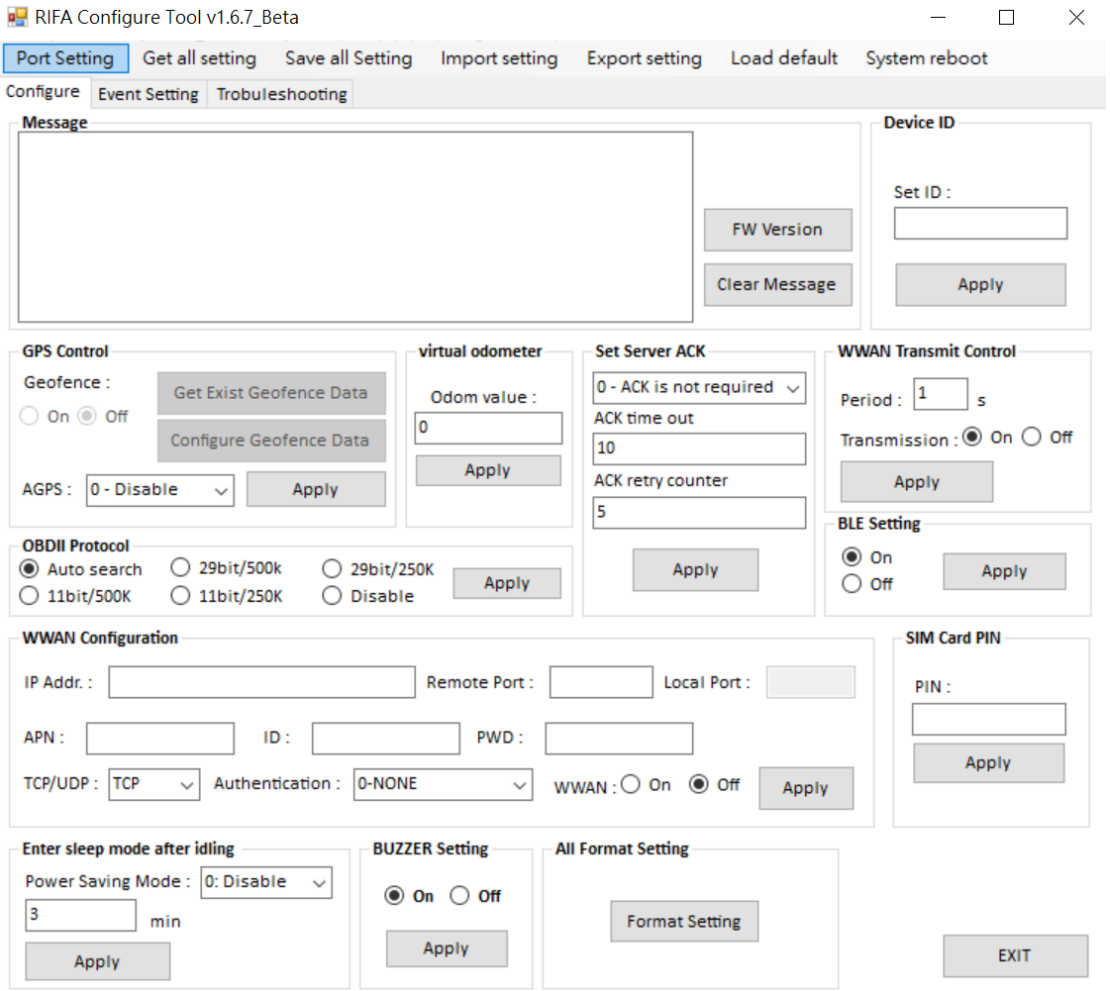
Following section 2.2. Checking the driver of USB cable is already installed on your laptop or notebook PC which you want to use it to configure the RIFA. (in Windows 10 , the USB driver is auto detect and install)

3.2. Configuring RIFA-M

Checking the hardware set-up is finished by section2.2. And then executing the "RIFA configure Tool" to configure RIFA.

First, you need to select the COM port correctly with your host device. And then click "Connect" button to start your RIFA's configuration.





Message

You can check the system status in message box.

FW Version: Click this button to check the firmware version in message box.

You can clean all the messages by clicking “Clear Message”.

Device ID

You can set up the ID of the RIFAs from 00000~99999, it can help you to track the device by ID and show IDs in your backend server.

GPS Control

(1) AGPS: Set “Enable” to execute AGPS function (Optional). It can enhance the performance of GPS locating under poor signal circumstance.

(2) Geofence: Set “On” to execute Geofence function

(3) Get Exist Geofence Data: Check now the geofence region (coordinates) which is setting.

(4) Configure Geofence Data: Set up a new geofence region (coordinates).

(5) Apply: Click this button to save all the "GPS control" settings.

Virtual Odometer

You can set up the initial value of virtual odometer by input value (unit=100 meters) and apply the setting.

Set Server ACK

By this function, you can set server ack mechanism to RIFA (data check) . but if your server doesn't send ack checking to client or node device , please set "0 – ACK is not required" to RIFA for avoiding data traffic jam.

WWAN Transmit Control

(1) Period: Type the period value (milliseconds) you want to send data via cellular network. (System will show the network type such as 3G , 4G or NBIoT)

(2) Transmission: Set "On" to request WWAN module to send data by defined time interval.

(3) Apply: Click this button to save all the "WWAN Transmit control" settings.

Set OBDII Protocol

Set baud rate which is corresponding to your vehicle ECU or you can set "Auto search" if you don't know what the baud rate is , but it may take a longer time for detecting the correct baud rate.

BLE Setting (optional)

Set on/off to trigger BLE function to RIFA .With this function, you can check RIFA's collected data through Android phone.

WWAN configuration

(1) IP Addr/Remote Port/Local Port : Type the IP address, remote port and local port(if needed) of your server uploading by RIFA

(2) APN/ID/PWD: Type the APN,PIN,ID and password provided by your WWAN SIM card carrier

(3) TCP/UDP: Select the data transmission protocol by RIFA's data uploading

(4) SMS phone No.: Type the phone number used which you want to remotely configure (dial into) RIFA by SMS

(5) Authentication: Select the authentication type by data transmission

(6) WWAN: Set "On" to wake up WWAN module

(7) Apply: Click this button to save all the "WWAN configuration" settings.

SIM Card PIN

Set SIM card pin number to RIFA which need to be corresponding to the telecom carrier's setting

Enter sleep mode after idling

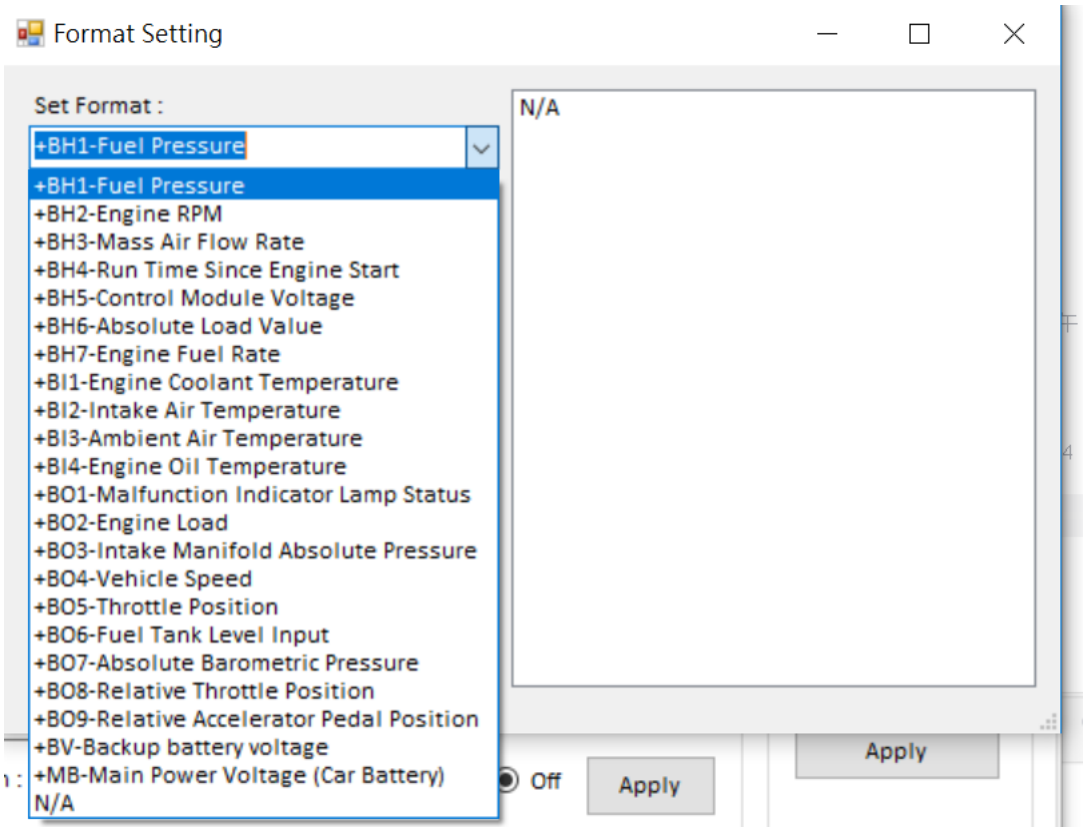
You can set the power saving mode with idling time , it can help RIFA enter to power saving mode to save the power of vehicle power. **Note**: RIFA is auto-configured to wake up by vibration or receiving data.

Buzzer Setting (optional)

You can trigger RIFA's internal buzzer by setting "on". Buzzer is designing for GPS starting and the waring of geofencing.

All Format Setting

You set what data you want to transmit to the server from RIFA. Add the data by ">" button and the click "OK" . The content includes OBDII data , Vehicle battery voltage and RIFA's backup battery voltage.



Get all setting : Click this button to check the previous settings of your RIFA.

Save all setting : Click this button to save all of your settings.

Noted:Please don't forget to click "Save all setting" button, it will keep all your settings after finished configuration procedure.

Import setting: you can import previous configuration by this button.

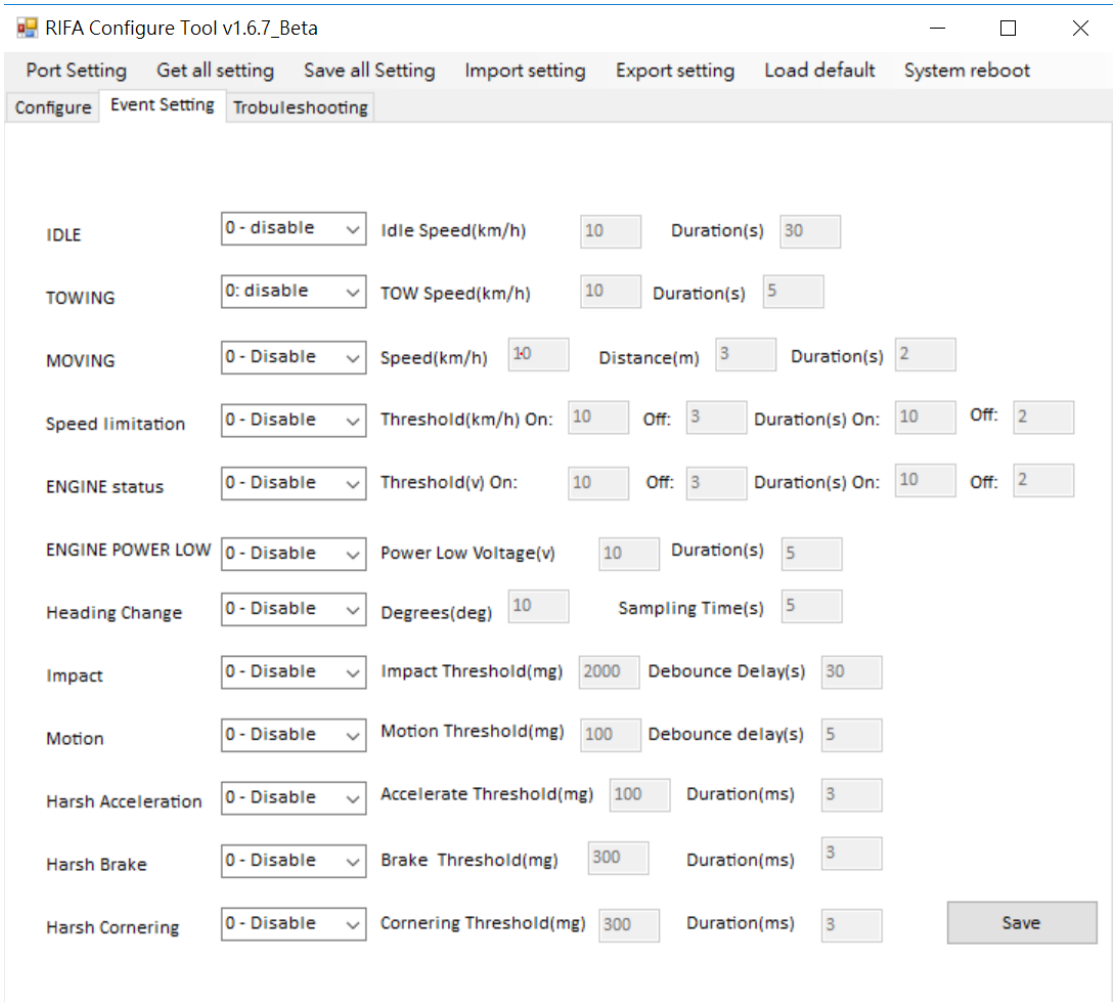
Export setting : you can export existing configuration by this button.

Load Default: Click this button to set factory settings to the RIFA.

System reboot: Click this button to reboot your device without any configuration changed.

EXIT: Click this button to close configuration tool.

Event setting



RIFA can integrate algorithm by event setting to detect harsh driving behavior, car collision, and vehicle's status. You can enable or disable the separate functions and choose the parameter to fulfill the situation for best performance.

3.3. BLE Function for Android App Application (Optional)

3.3.1 System Requirement

Software version needed is Android 6.0.0 with BLE 4.0
(or above version)

3.3.2 APK File

Please install Antzer BLE software APK : Antzer_BLE_vX_X_X.apk

3.3.3 Installation

1.Copy above .apk file into Android cell phone.
2.Click the apk and following the installing steps to install the BLE application.
3. It appears an ANTZER-BLE icon on Android cell phone Apps list as Figure 1.

Figure 1

3.3.4 Connect Procedures

1.Plug in the power adapter and power on RIFA BLE dongle.
2.Wait for booting ready buzzer beep (1 high and 1 low beep).
3.Launching ANTZER BLE ANTZER-BLE app as Figure 2

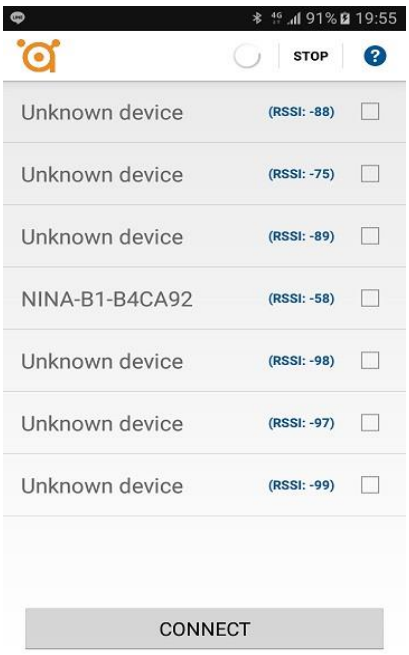


Figure 2

4. Select RIFA BLE device on the list (check BLE mac address) and click the "CONNECT" button as Figure 3.

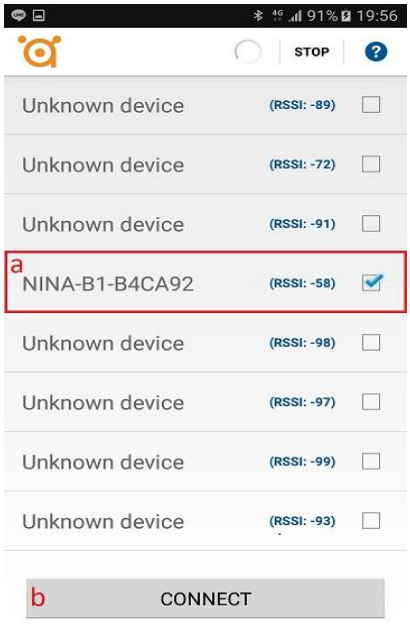


Figure 3

5. Waiting for RSSI value appearance (BLE signal checking) and then click "CHAT" tab button to read CAN bus data as Figure 4.

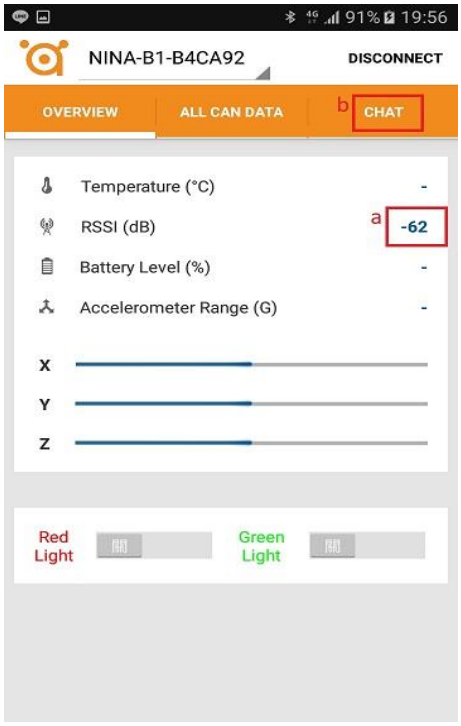


Figure 4

6. CAN bus data appears by hex data format and in a row by data scrolling as Figure 5.

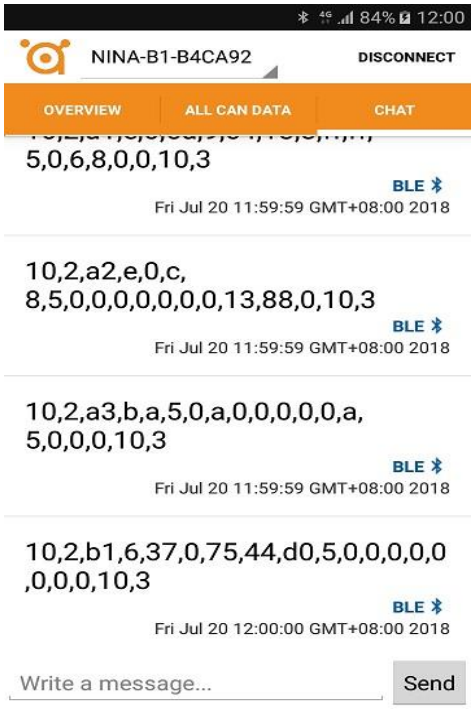


Figure 5

For OBDII/J1939 Data Collection, please follow below steps

7. Waiting for RSSI value appearance (BLE signal checking) and then click

“ALL CAN DATA” sheet to read CAN(J1939) data as Figure 6.

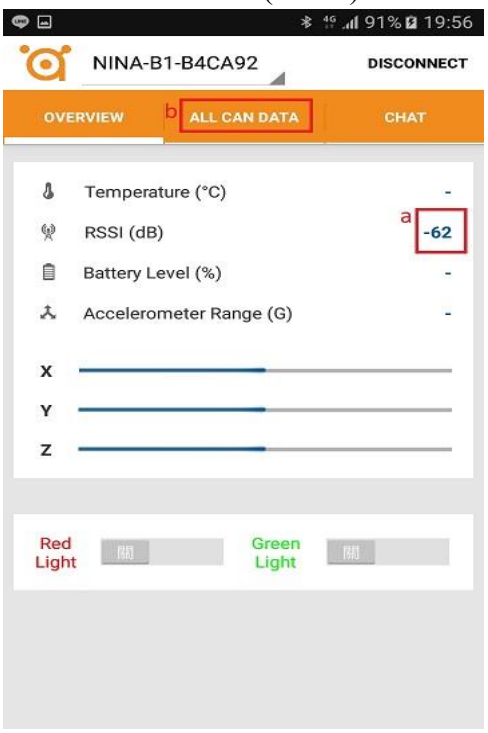


Figure 6

8. CAN (OBDII/J1939) bus data will appear by readable format as Figure 7.

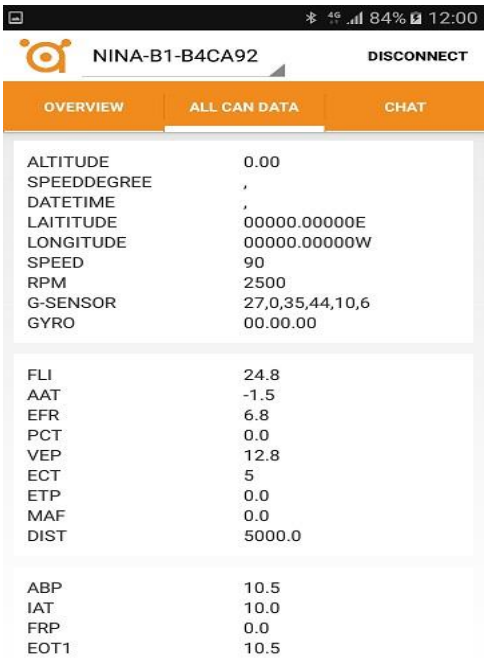


Figure 7

3.3.5 Bluetooth Disconnection Procedures

1. 1.Click “DISCONNECT” button as Figure 9(a).

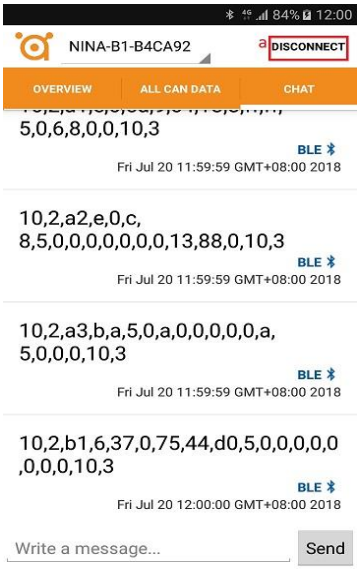


Figure 9

2. The App will stop data appearance scrolling, and RIFA BLE dongle will have a short beep buzzer after completing BLE disconnection between smart phone and RIFA.

3. 3.The “DISCONNECT” button will change to “CONNECT” button around 30 seconds as Figure 10(a).

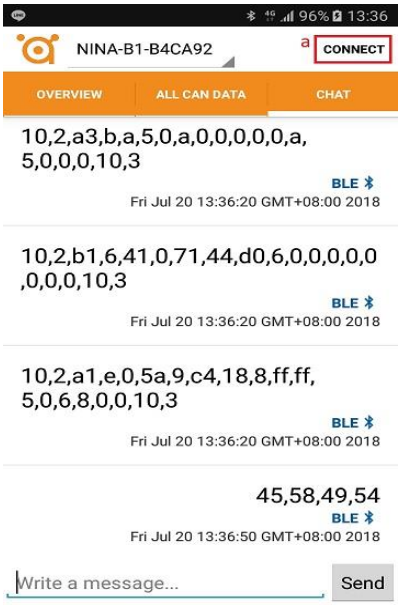


Figure 10

4. For re-connection of BLE device, click Android device button ( ) to initialize connection(as Figure 2).

3.3.6 Known Issues and troubleshooting (depends on Android Version)

1. sometimes in disconnection procedure, the “DISCONNECT” button may not change to “CONNECT” button after 30 seconds waiting.
Troubleshooting – click Android device button (◁) to forcing disconnection.

2. Sometimes in BLE re-connection, Android app may not read CAN bus data.

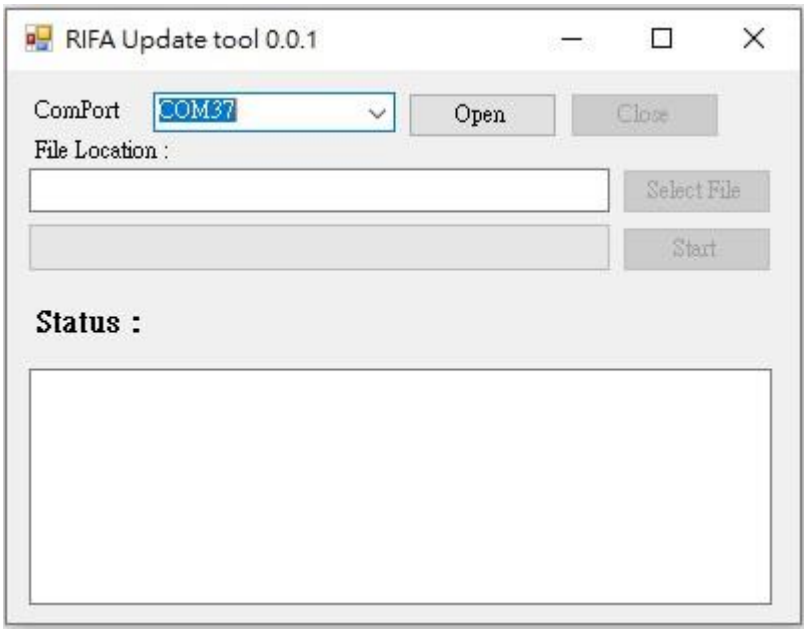
Troubleshooting – click Android device button (◁) to return to initial connection screen and doing connection procedures again.

3. In disconnection procedure, it could not beep the buzzer of disconnection and can’t be re-connected anymore. (It may happen in some specific Android devices with low possibility).

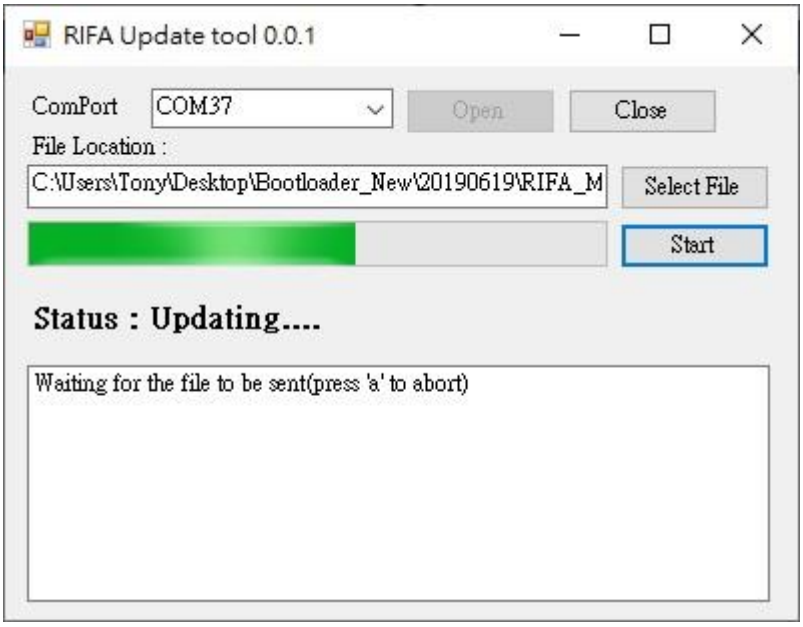
Troubleshooting – Power off RIFA dongle (re-connect from vehicle) for forcingly disconnect BLE connection and then re-connect it.

4. Firmware Update

1. Launch RIFA_Update_tool.exe (with the firmware file-RIFA_V2_2x.bin at the same folder) on your PC or NB which is connecting to RIFA-M
2. (1) Select the correct COM port which is connecting to RIFA-M
   (2) Click “Open” button
   (3) Click “Select File “ button and choose the right firmware file.

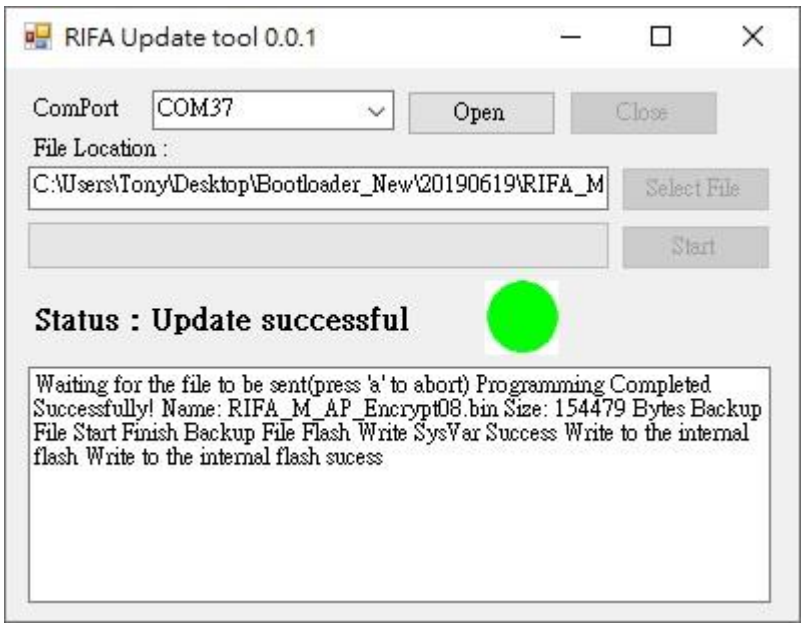


(4) Click “Start” button and start the update procedure.



Note:Keep connect RIFA with your PC while updating procedure.

(5) You will see the message as below photo while finishing the update.



5. Dimension Drawing

6.Reliability Specifications

6.1Environmental

Environmental specifications of RIFA-M follow MIL-STD-810G, as indicated in the following table.

Environment	Specifications
Operating Temperature	-40°C to 85°C (without Battery) -20°C to 60°C (with Battery)
Vibration	Operating: Random, 7.69(Grms), 20~2000(Hz) Compliant with MIL-STD-810G

6.2Certification and Compliance

Antzer-Tech RIFA-M complies with the following standards:

- CE
- FCC Class B
- RoHS
- MIL-STD-810G Vibration Compliant

7 Appendix A - RIFA-M Series Transmission Protocol

Tracking Data by ASCII Format

*Single Packet:

For the single packet (we use ASCII data format), each field is delimited by a single comma " , " for example :

Header						Data1	Trailer		
Prefix	,	Len	,	Seq_Id	,	Tracking Data	*	Checksum	<CR><LF>

|←--------Calculated Length ------→|

|←------Included for Checksum calculation--------→|

*Multiple-Packets:

When it comes to the multiple-packets, each tracking data is delimited by <CR><LF> character (0x0D 0x0A). for example, if there are three packets , the data format is as blow:

Header					Data1	Trailer	Data2	Trailer	Data3	Trailer		
Prefix	,	Len	,	Seq_Id	Tracking Data (Packet 1)	<CR><LF>	Tracking Data (Packet 2)	<CR> <LF>	Tracking Data (Packet 3)	*	Checksum	<CR><LF>

|←---------- Calculated Length ---------- →|

|←----------Included for Checksum calculation----------→|

Device Message Formats:

Syntax:

<Prefix>,<Len>,<Seq_ID>,<Tacking Data>*<Checksum><CRLF>

@@,<Len>,<Seq_ID>,<UD> ,<EC>,<EP>,<LT>,<LN>,<AL>,<SP>,<HD>,<GD>,<GT>,<PF>,<DS>,<AI1>,<IN>,<OT>…..*XX<CR><LF>

Field	Descriptions	Unit
Prefix	Default "@@"	
Len	Package Length	
Seq_ID	Sequence number, cyclic accumulation from 1 to 65535	
UD	Unit ID, Default IMEI	
EC	Event Code	See Table A

EP	Event Params	See Table A
LT	Latitude (NMEA format), e.g. DDMM.mmmm	0.00001 unit
LN	Longitude (NMEA format), e.g. DDMM.mmmm	0.00001 unit
AL	Altitude in meters above sea level, e.g. AA.a	0.1 unit
SP	Speed over the ground in knots, e.g. SSS.sss	0.001 unit
HD	Heading over ground in degrees, e.g. HHH.hh	0.01 unit
GD	UTC Date, e.g. YYMMDD	
GT	UTC TIME, e.g. hhmmss	
PF	1 if a GPS valid fix. 0 if not a valid fix.	
DS	Odometer	0.1 Km
AI1	Analog Input Value	0.001 Volt
IN	All Input Status	
OT	All Output Status	
.....	<Custom Info> For appending additional data field to the tail of the standard **Tacking Data.**	Refer to AT command AT+FORMAT part
*XX	Checksum	

Table A. Event Code and Event Parameters

Event Code	Event Params	Descriptions
2	NULL	GPRS Time based tracking event
21	Input Index (0-2)	Input on Alert
22	Input Index (0-2)	Input off Alert
23	Output Index (0-1)	Output on Alert
24	Output Index (0-1)	Output off Alert
25	NULL	Motion on alert
26	NULL	Motion off alert
27	NULL	Impact on alert
28	NULL	Impact off alert
29	NULL	Idle on alert
30	NULL	Idle off alert
31	NULL	Towing detected alert (define as Appendix B)
32	NULL	Towing stopped alert (define as Appendix B)
33	NULL	Moving detected
34	NULL	Moving stopped

35	NULL	Harsh acceleration on event
36	NULL	Harsh acceleration off event
37	NULL	Harsh brake on event
38	NULL	Harsh brake off event
39	NULL	Harsh cornering on event
40	NULL	Harsh cornering off event
41	NULL	Direction change alert
43	NULL	Main power on event
44	NULL	Main power off event
45	NULL	Main power low voltage on event
46	NULL	Main power low voltage off event
47	NULL	Backup battery low voltage on event
48	NULL	Backup battery low voltage off event
49	NULL	Over speed on event
50	NULL	Over speed off event
51	Geofence Index (Max: 255)	Geofence in event
52	Geofence Index (Max: 255)	Geofence out event
53	NULL	Wake up by G-sensor
54	NULL	Motion off during 60 sec

8 Appendix B – AT command list of RIFA-M

Syntax:

Write cmd:

AT+<Command >,<Parameter1>,<Parameter2>,<Parameter3>….<CR><LF>

Response:

OK+<Command ><CR><LF> or FAIL+<Command ><CR><LF>

Read cmd:

AT+<Command >?<CR><LF>

Response:

OK+<Command >,<Parameter1>,<Parameter2>,<Parameter3>….<CR><LF>

//WWAN: set parameters of your WWAN module (SIM card information)

AT+WWAN,<WWAN_Enable>,<APN>,< Host_IP >,< Socket_Type >,< Host_Port >,<
UDP_LocalPort >,< UserName >,< Password >,< PDP_Authentication >

<WWAN_Enable>: 0:Disable WWAN, 1:Enable WWAN

<APN>: Max Len 32

< Host_IP >: Max Len 32

< Socket_Type >: 0: TCP, 1:UDP

< Host_Port >: 0-65535

< UDP_LocalPort >: 0-65535

< UserName >: Max Len 32

< Password >: Max Len 32

< PDP_Authentication >: 0:None, 1:PAP, 2:CHAP, 3:AUTO

//SERVER: ACK Message "$AZACK" ,set the server ACK mechanism

AT+SERVER,<SERVER_ACK>,<ACK_TIMEOUT>,<ACK_Retry_Counter>

<SERVER_ACK>: 0: ACK is not required, 1: ACK is required(reserved)

<ACK_TIMEOUT>: Minimum unit is 1 sec, 0-255, default 10 sec,

<ACK_Retry_Counter>: 0-255, default 5,

//INPUT:set the debounce time of ACC input and digital input.

Write cmd:

AT+INPUT,<Index>,<On Debounce Time>,<Off Debounce Time>

Read cmd:

AT+INPUT,<Index>,?

<Index>: 0-2, 0 is ACC PIN, 1 is DI0 PIN, 2 is DI1 PIN

<On Debounce Time>: Minimum unit is 0.1sec, 0-65535, default 10 (1sec)

<Off Debounce Time>: Minimum unit is 0.1sec, 0-65535, default 10 (1sec)

//IDLE: set the parameters of idle event code

AT+IDLE,<Condition>,<Idle Speed>,<Duration>

<Condition>:

0: disable, 1: only used speed, 2: ACC ON, 3:ENGINE ON, 4: ACC ON & ENGINE ON

<Idle Speed>: default 10km/h, 0-255

<Duration>: minimum unit is 1 sec, default 30 (30 sec), 0-65535

//TOWING:set the parameters of towing event code

AT+TOW, <Condition>,<TOW Speed>,<Duration>

<Condition>:

0: disable, 1: ACC OFF, 2:ENGINE OFF, 3: ACC OFF & ENGINE OFF

< TOW Speed>: default 10km/h, 0-255

<Duration>: base on 1 sec, default 5 (5 sec), 0-65535

//MOVING: set the parameters of moving event code

AT+MOVE,<Enable>,<Speed>,<Distance>,< Duration >

<Enable>: 0 disable, 1 enable

<Speed>: 10 (Km/h)

<Distance>: 3 (minimum unit is 10m)

< Duration >: 2 sec(minimum unit is 1sec)

//Limit Speed: set the parameters of speed limitation event code

AT+LSPD, <Enable>,<OnThreshold>,< OnDuration >,< OffThreshold >,< OffDuration >

<Enable>: 0 disable, 1 enable

<OnThreshold>: 70 (km/h)

< OnDuration >: 30 (seconds, minimum unit is 1sec)

< OffThreshold >: 60 (km/h)

< OffDuration >: 30 (seconds, minimum unit is 1sec)

//Car(Engine) Power ON and OFF Event: set the parameters of car power on and off event code

AT+ENGE, <Enable>, <OnThreshold>,< OnDuration >,< OffThreshold >,< OffDuration >

<Enable>: 0 disable, 1 enable

<OnThreshold>:132 (volt, minimum unit is 0.1 volt), ex: parameter 132 = 13.2 volt

<OnDuration>: 30 (seconds, minimum unit is 1sec)

<OffThreshold>:124 (volt, minimum unit is 0.1 volt), ex: parameter 124 = 12.4 volt

<OffDuration>:180 (seconds, minimum unit is 1sec)

//Car(Engine) Power Low Event: set the parameters of car power low event code

AT+MPWLW, <Enable>,< Power Low Voltage>,<Duration>

<Enable>: 0 disable, 1 enable

< Power Low Voltage>: 110 volt (Minimum unit is 0.1 volt) 110 = 11.0 volt

<Duration>: 2 sec (minimum unit is 1sec)

//Backup Battery Low Event (hardcoded) :the parameters of RIFA-M battery power low event code

Default low volt is 3.5 volt, duration is 20 sec,

This function Enable/Disable is following AT+MPWLW

//Heading Change Event: set the parameters of car heading change event code

AT+HDCG, <Enable>,< Degrees>,<Sampling Time>

<Enable>: 0 disable, 1 enable

< Degrees>: Default 45 deg. (Range is from 0 to 359)

<Sampling Time>: Default is 2 sec (minimum unit is 1sec)

//Impact Event: set the parameters of impact event code

AT+IMPT, <Enable>,<Impact Threshold>,<Debounce Delay>

<Enable>: 0 disable, 1 enable

<Impact Threshold>: Default 2000mg (Range is from 1000 to 2000mg)

<Debounce Delay>: Debounce time delay in second for the first impact detection to the next detection. Default 2 sec

//Motion Event: set the parameters of motion event code

AT+ MTON, <Enable>,<Motion Threshold>,<Debounce delay>

<Enable>: 0 disable, 1 enable

<Motion Threshold>: Default is 100 mg, Range is from 30 to 2000mg.

<Debounce delay>: Default is 5 sec (minimum unit is 1 sec)

Example:

Write:AT+MTON,1,100,5

Means: enable motion event of threshold 100mg and debounce delay 5 secs

//Power Saving Mode: set the parameters of how to enter power saving code

AT+PWSV,< Power OFF Detect >,<Duration>

< Power OFF Detect >: 0: Disable, Bit 0: Motion OFF, Bit 1: Engine OFF, Bit 2: ACC OFF

<Duration>: Default 3min(minimum unit is 1 min), wait after <Power OFF Detect> conditions are all detected, then enter into power saving.

//System Reset (only Write Cmd): reset and save parameters by this command

AT+RST,<Action>

<Mode>:

1: System Reboot

2: Save all parameters

3: Reset all parameters to factory default

4: Clear external flash memory

// Query firmware Version: check the firmware version by this command

AT+VER?

//Tracking Report Data: set up the frequency of tracking data

AT+TRCK,<Mode>,< Tracking interval >

<Mode>: 0: Disable Tracking Report Data, 1: Time Mode

< Tracking interval >: Default “30” seconds (minimum unit is 1sec)

//Set SIMCard PIN Code: key in the pin code of SIM card if needed

AT+PIN,<PIN Code>

<PIN Code>: Max Len 10 characters

//Set the vehicles virtual odometer: define the initial value of virtual odometer

AT+ODOM,<odom_value>

<odom_value>: set the virtual odometer initial value (Default value is 0 Kilometers)

The unit of virtual odometer value is in 0.1 kilometer (ex: value 12350 is equal to set 1235 km)

//Set the vehicles identification number : customer can set up the ID of the RIFA from 00000~99999, it can help you to track the device by ID and show IDs in your backend server.

AT+VIN,< ID_Number >

< ID_Number >: vehicles identification number,

Max Len 17 characters (Default setting is 24688)

//Set Harsh Acceleration event: set command to activate harsh acceleration detecting

AT+HRAC,<Enable>,< Accelerate _Threshold>,< Duration >

<Enable>: 0:Disable, 1:Enable

< Accelerate_Threshold > Default 100mg, Range is from 30 to 2000mg.

< Duration > Duration in 0.1 seconds, Max Range is 255, Default: 3 (300ms)

//Set Harsh Brake event : set command to activate harsh brake detecting

AT+HRBK,<Enable>,< Brake _Threshold>,< Duration >

<Enable>: 0:Disable, 1:Enable

< Brake _Threshold > Default 300mg, Range is from 30 to 2000mg.

< Duration > Duration in 0.1 seconds, Max Range is 255, Default: 3 (300ms)

//Set Harsh Cornering event : set command to activate harsh corneting detecting

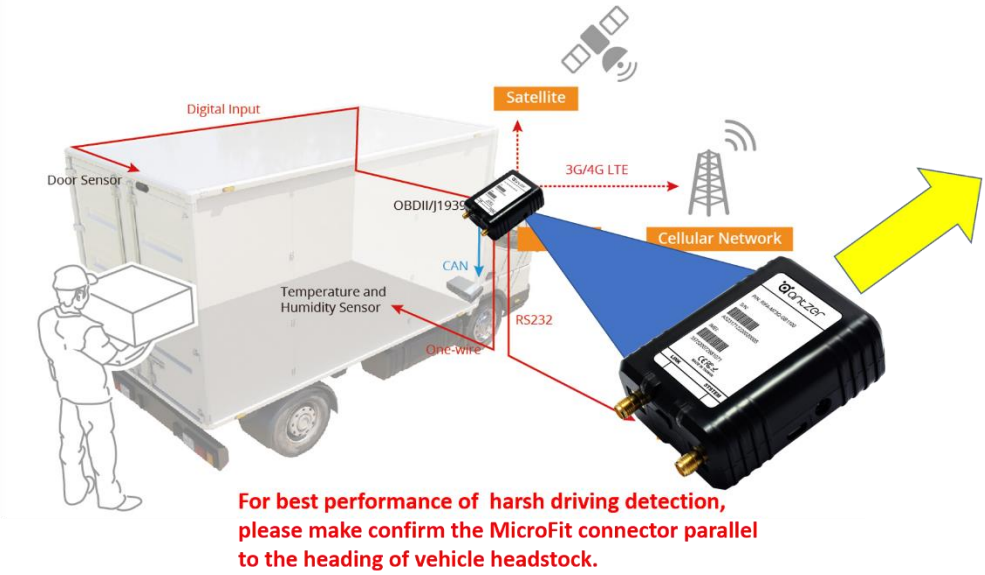
AT+HRCR,<Enable>,< Cornering _Threshold>,< Duration >

<Enable>: 0:Disable, 1:Enable

< Cornering _Threshold > Default 300mg, Range is from 30 to 2000mg.

< Duration > Duration in 0.1 seconds, Max Range is 255, Default: 3 (300ms)

Note: For best performance of harsh driving detection, please make confirm the MicroFit connector parallel to the heading of vehicle headstock. And keep flat on installation.



//Geofencing Set-up : RIFA-M support geofence function which can help user to define the allowed or prohibited region. When vehicle is crossing to the geofence function, RIFA-M will send the event code 51 and 52 to backend server.

//Digital output: set the parameters of digital output

AT+OCTL, <Output ID>, <New State>, <New State Duration>, <Reverse Duration>,

<Repeat Times>

<Output ID>: Digital output ID, 1: DO0, 2:DO1

<New State>: 0: OFF, 1: ON

<New State Duration>: New state duration in 0.1 seconds

<Reverse Duration>: Reverse state duration in 0.1 seconds.

<Repeat Times>: repeat times. (0 ~ 255, 0 mean output control once,255 means continuous repeat)

//FORMAT : set the data format which transmit from RIFA-M

AT+FORMAT,<Data Format>,<Format Header Prefix>,<Format Custom Info>

<Data Format>: 0: ASCII, 1:Binary (reserved)

<Format Header Prefix>: @@ (default)

<Format Custom Info>: Null (default)

// Custom Info Table:

Field	Descriptions	Data Size	Unit (value range)
+MB	Main Power Voltage (Car Battery)	uint16_t	0.1 volt
+BV	Backup battery voltage	uint16_t	0.1 volt
+BO1	Malfunction Indicator Lamp Status	uint8_t	0: OFF, 1: ON
+BO2	Engine Load	uint8_t	% (0 ~ 100)
+BO3	Intake Manifold Absolute Pressure	uint8_t	kPa (0 ~ 255)
+BO4	Vehicle Speed	uint8_t	km/h (0 ~ 255)
+BO5	Throttle Position	uint8_t	% (0 ~ 100)
+BO6	Fuel Tank Level Input	uint8_t	% (0 ~ 100)
+BO7	Absolute Barometric Pressure	uint8_t	kPa (0 ~ 255)
+BO8	Relative Throttle Position	uint8_t	% (0 ~ 100)
+BO9	Relative Accelerator Pedal Position	uint8_t	% (0 ~ 100)
+BH1	Fuel Pressure	uint16_t	kPa (0 ~ 765)
+BH2	Engine RPM	uint16_t	rpm (0 ~ 16383)
+BH3	Mass Air Flow Rate	uint16_t	0.01 grams/sec (0 ~ 65535)
+BH4	Run Time Since Engine Start	uint16_t	Seconds (0 ~ 65535)
+BH5	Control Module Voltage	uint16_t	0.001 volt (0 ~ 65535)
+BH6	Absolute Load Value	uint16_t	% (0 ~ 25700)
+BH7	Engine Fuel Rate	uint16_t	L/h (0 ~ 3212)
+BI1	Engine Coolant Temperature	int16_t	℃(-40 ~ 215)
+BI2	Intake Air Temperature	int16_t	℃ (-40 ~ 215)
+BI3	Ambient Air Temperature	int16_t	℃ (-40 ~ 215)
+BI4	Engine Oil Temperature	int16_t	℃ (-40 ~ 210)
+BS1	Vehicle Identification Number	17 char	

Example:

Attach custom data (like OBDII data) to standard protocol , please set AT command “AT+FORMAT” , the data format and sequence are defined by your AT command. ◦

Command Set:

AT+FORMAT,0,@@,**+BV+BO4+BH2+BO1+BI4+BH5+BI1**

- The data structure sent to server is as below sequence:

Syntax:

@@,<Len>,<Seq_ID>,<UD> ,<EC>,<EP>,<LT>,<LN>,<AL>,<SP>,<HD>,<GD>,<GT>,<PF>,<DS>,<AI1>,<IN>,<OT>,**<BV>,<BO4>,<BH2>,<BO1>,<BI4>,<BH5>,<BI1>***XX<CR><LF>

Example:

@@,114,73,352953080326071,2,0,250200935,1213344184,363,45,26726,190116,102603,1,3,0,1,1,**37,125,1800,1,85,12000,70***04

37 => Backup battery voltage is 37 * 0.1 = 3.7 volts

125 => Vehicle Speed is 125 km/h

1800 => Engine RPM is 1800 rpm

1 => MIL ON

85 => Engine Oil Temperature is 85 °C

12000 => Control Module Voltage is 12000 * 0.001 = 12.000 volts

70 => Engine Coolant Temperature is 70 °C

9 Appendix C – RIFA BLE Transmit Protocol Format (Optional)

RIFA BLE Transmission Protocol Format
GPS 、 OBD2 、 GSensor Data

Framing: Total = 20 bytes

DLE	STX	CMD	DLC	Data	DLE	ETX
1 byte	1 byte	1 byte	1 byte	14 bytes	1 byte	1 byte
0x10	0x02				0x10	0x03
0	1	2	3	4 ~ 17	18	19

Response = 0x91: Receive GPS Latitude + Hemisphere Data (ASCII)

DLE	STX	CMD	DLC	Latitude	Hemi	0x00	DLE	ETX
0x10	0x02	0x91	11	10 bytes	N or S	3 bytes	0x10	0x03

Response = 0x92: Receive GPS Longitude + Hemisphere Data (ASCII)

DLE	STX	CMD	DLC	Longitude	Hemi	0x00	DLE	ETX
0x10	0x02	0x92	12	11 bytes	E or W	2 bytes	0x10	0x03

Response = 0x93: Receive GPS Altitude Data (ASCII)

DLE	STX	CMD	DLC	Altitude	0x00	DLE	ETX
0x10	0x02	0x93	7	7 bytes	7 bytes	0x10	0x03

Response = 0x94: Receive GPS Speed + Degree Data (ASCII)

DLE	STX	CMD	DLC	Speed	Degree	0x00	DLE	ETX
0x10	0x02	0x94	11	6 bytes	5 bytes	3 bytes	0x10	0x03

Response = 0x96: Receive GPS Date + Time (ASCII)

DLE	STX	CMD	DLC	Date	Time (UTC)	DLE	ETX
0x10	0x02	0x96	6	6 bytes (DDMMYY)	8 bytes (hhmmss.d)	0x10	0x03

Response = 0xB1: Receive G-Sensor Data (Binary)
(**available data length = 12bit, lowest 4bit is unavailable data, negative value represent by 2's complement number**)

DLE	STX	CMD	DLC	XX YY ZZ	0x00	DLE	ETX
0x10	0x02	0xB1	6	6 bytes	8 bytes	0x10	0x03
				XX_L/H(4,5), YY_L/H(6,7), ZZ_L/H(8,9)			

Response = 0xA1: Receive MIL+ELD+ECT+FLP+IMAP+RPM+SPEED+IAT+MAF Data (Binary)

DLE	STX	CMD	DLC	MIL (unsign)	ELD (unsign)	ECT (sign)	FLP (unsign)	IMAP (unsign)	RPM (unsign)	SPEED (unsign)	IAT (sign)	MAF (unsign)	DLE	ETX
0x10	0x02	0xA1	14	1 byte	1 byte	2 byte	2 byte	1 byte	2 byte	1 byte	2 byte	2 byte	0x10	0x03
						ECT_L ECT_H	FLP_L FLP_H		RPM_L RPM_H		IAT_L IAT_H	MAF_L MAF_H		

Response = 0xA2: Receive THROTTLE+ENG_RUNTIME+FTLI+ABP+Volt+AVL+RTP+AAT+RAPP Data (Binary)

DLE	STX	CMD	DLC	THROTTLE	ENG_RUNTIME	FTLI (unsign)	ABP (unsign)	Volt (unsign)	AVL (unsign)	RTP (unsign)	AAT (sign)	RAPP (unsign)	DLE	ETX
0x10	0x02	0xA2	13	1 byte	2 byte	1 byte	1 byte	2 byte	2 byte	1 Byte	2 Byte	1 Byte	0x10	0x03
					ERT_L ERT_H			Volt_L Volt_H			AAT_L AAT_H			

Response = 0xA3: Receive EOT+EFR Data (Binary)

DLE	STX	CMD	DLC	EOT (sign)	EFR (unsign)	DLE	ETX
0x10	0x02	0xA3	4	2 byte	2 byte	0x10	0x03
				EOT_L EOT_H	EFR_L EFR_H		

10. Appendix D - Server information setting

//==========//
// Command //
//==========//
//Set the backend server information include Acknowledgement and Heartbeat.

Syntax:

AT+SERVER,<SERVER_ACK>,<ACK_TIMEOUT>,<ACK_Retry_Counter>,<Heartbeat_ interval>

<SERVER_ACK>:

If the ACK is required, device will wait for the ACK and then send next report. If no ACK is received within <ACK_TIMEOUT> value for <ACK_Retry_Counter> times, the device will reset wwan module and try the steps again.

0: ACK is not required.

1: ACK is required.

<ACK_TIMEOUT>:

Minimum unit is 1 sec, 0-255, default 10 sec.

<ACK_Retry_Counter>:

0-255, default 5 times.

<Heartbeat_ interval>:

Period in minutes for the device to send heartbeat package message to the backend server.

0-255, default 0 minute(Disable).

//==========//
// Heartbeat Message Format //
//==========//
Heartbeat is recommended to ensure TCP connection when the time interval of tracking is set too long. Whenever tha backend server receives a heartbeat package, it should reply an acknowledgement to device

Syntax:

<Header>,<Seq_Num>,<UD><CR><LF>

<Header>: 2 Bytes, it is in ASCII "@Z" (0x40 0x5A).

Header of the package from tracker to server.

<Seq_Num>:
The backend server uses the <Seq_Num> extracted from the heartbet package from the device as the <Seq_Num> in the server acknowledgement of the heartbeat.

<UD>:
Unit ID, Default IMEI.
Package size: 22~26 Bytes(ASCII Format).

Example:
@Z,65535,352753092064383<CR><LF>

//==//
//   Server Acknowledgement Message Format   //
//==//
Server response message format in ACK protocol for acknowledging receipt of device message.
Syntax:
<Header>,<Seq_Num>,<UD><CR><LF>

<Header>: 2 Bytes, it is in ASCII "$$" (0x24 0x24).
Header of the package from server to tracker.

<Seq_Num>:
The backend server uses the <Seq_Num> extracted from the received message as the <Seq_Num> in the server acknowledgement.
<UD>:
Unit ID, Default IMEI.
Package size: 22~26 Bytes(ASCII Format).

Example:
$$,5,352753092064383<CR><LF>