ORION Protocol

For FW 970817

Version：3.02.03

Date：2011/3/15

Table of Contents

0、Document History

1、Properties

2、Actions

3、Event Reports

Appendix-1

Appendix-2

Appendix-3

0. Document history

0.1 Revision history

Date	Version	Description	Hardware	Firmware
2006/03/01	V2.06.03	First version of ORION Protocol	Easytrac Speedtrac	960723 970123
2008/06/01	V3.01.01	Revised all event types to 4 bytes Add optional Cell Info report	Easytrac basic Easytrac pro Speedtrac	970609 970710 970804
2009/02/01	V3.01.03	Add 1.14 COM to congigure COM port baud rate Add 1.15 IDControl to enable RFID and Immobilizer integration Add 1.16 ValidID to add RFID in the tracker Add 2.6 ActUpgradeFW to update firmware OTA by GPRS Add 3.25 LGSK, 3.26 LMSK for Speedtrac Pro, Choco and Suntrac Revised 3.23 LUIN and add 3.24 LUOT to support iButton reading	Easytrac basic Easytrac pro Speedtrac basic Speedtrac Pro Chco Suntrac	970806 970807
2009/10/01	V3.01.04	Modify 1.16 ValidID to max. 200 IDs and $GetvalidID command syntax Modify 2.26 ActSetGeoFence to max 199 sets of GeoFence Add 2.28 ActComOut to send strings to Com port Add vibration driven journey stat / end events 3.25 LGSK, 3.26 LGEK, 3.27 LMSK, 3.28 LMEK Add schedule timer events 3.29 RGRK, 3.30 LGRK, 3.31 RMRK, 3.32 LMRK	Easytrac Speedtrac	970815

		Add ORION iButton Reader events 3.33 LIBT		
2011/01/03	V3.02.01	Add command EventMask Add command OBDLimit Add function VACCON Add function ID Control WARN Added command ATR Added command TempLimit Added command PowerConfig	Easytrac Speedtrac OBDtrac	970817
2011/03/15	V3.02.03	Added DataMode Added AI mode Added ActSendSMS	Easytrac Speedtrac OBDtrac	970817

0.2 Related Documents

Item	Document	Description
1	Easytrac Getting Started V3.01.04	Hardware introduction Configuration steps by steps
2	Easytrac Basic Getting Started V1.01.01	Hardware introduction Configuration steps by steps
3	Speedtrac Getting Started V3.02.01	Hardware introduction Configuration steps by steps
4	OBDtrac Basic Getting Started V1.01.03	Hardware introduction Configuration steps by steps

1. Properties

1.1 SerialNo (Device Serial Number)

Command	**GetSerialNo**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the Serial Number of the Device. It's fixed and can not be changed.	
Syntax	**$GetSerialNo,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number". Range from "0000" to "9999".
	Password	Default Value : "0000". Range from "0000" to "9999".
Return Syntax	**$GetSerialNo,[Command S/N],[Serial No],[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	Serial No	Device Serial Number. 12 Digit in total. Range from "EZ0000000000" to "EZ4294967296".
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the serial number of device and return "EZ1234567890". →$GetSerialNo,1234,0000& ←$GetSerialNo,1234,EZ1234567890,080603154025&	

1.2 UnitID (Device Unit ID, IMEI, IMSI, HW Version, FW Version)

Command	**SetUnitID**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	The Device Unit ID can be set by the System Integrator. All 'Event Reports' are identified by 'Unit ID'.	
Syntax	**$SetUnitID,[Command S/N],[Password],[UnitID]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000".

		Range from “0000” to “9999”.
	UnitID	Device Unit ID. Default value is the same as the serial number. Range from ‘0000000000’ to ‘4294967296’, Other characters are not allowed. Range: 1 ~ 10 digits
Return Syntax	**$SetUnitID,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the device ID as “1234567890”. →$SetUnitID,1234,0000,1234567890& ←$SetUnitID,1234,OK,080603154025&	

Command	**GetUnitID**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the device ID, IMEI, IMSI, HW Version and FW Version	
Syntax	**$GetUnitID,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetUnitID,[Command S/N],[Unit ID],[IMEI],[IMSI],[HW Version], [FW Version],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	UnitID	Device ID. Default value is the same as the serial number. Range from ‘0000000000’ to ‘4294967296’
	IMEI	International Mobile Equipment Identity (GSM Module unique ID)
	IMSI	International Mobile Subscriber Identity

		(SIM Card unitque ID)
	HW Version	Hardware Version
	FW Version	Firmware Version
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the tracker ID, IMEI, IMSI, HW Version, FW Version →$GetUnitID,1234,0000& ←$GetUnitID,1234,1234567890,352021005322607,466922002606092,V6,EZ979693,080603154025&	

1.3 Password (Device Password)

Command	**SetPwd**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the password of the device.	
Syntax	**$SetPwd,[Command S/N],[Password],[NewPassword]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	NewPassword	New Device Password.
Return Syntax	**$SetPwd,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Change the password of the device from ”0000” to ”9999”. →$SetPwd,1234,0000,9999& ←$SetPwd,1234,OK,080603154025&	

1.4 SIMPINx (SIM PIN1 and PIN2)

Command	**SetSIMPINx**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the PINs of the SIM card to be used in the device.	
Syntax	**$SetSIMPINx,[Command S/N],[Password],[SIM PIN1],[SIM PIN2]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	SIM PIN1	Default Value = “”. Range from “0000” to “99999999”
	SIM PIN2	Default Value = “”. Range from “0000” to “99999999”
Return Syntax	**$SetSIMPINx,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the SIM PIN1 and PIN2 as ”0123” and “4567”. →$SetSIMPINx,1234,0000,0123,4567& ←$SetSIMPINx,1234,OK,080603154025&	

Command	**GetSIMPINx**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the SIM PIN1 and PIN2.	
Syntax	**$GetSIMPINx,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.

Return Syntax	**$GetSIMPINx,[Command S/N],[SIM PIN1],[SIM PIN2],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	SIM PIN1	Range from “0000” to “99999999”
	SIM PIN2	Range from “0000” to “99999999”
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the SIM PIN1 and PIN2 and return SIM PIN1 and PIN2 as “0123” and “4567”. →$GetSIMPINx,1234,0000& ←$GetSIMPINx,1234,0123,4567,080603154030&	

1.5 SwitchMode (GPRS mode, SMS mode or Auto switch Mode)

Command	**SetSwitchMode**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the reporting mode of the unit.	
Syntax	**$SetSwitchMode,[Command S/N],[Password],[Switch Mode],[GPRS Reconnect],[SMS Duration]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Switch Mode	GPRS : Set the tracker to send reports by GPRS connection only. But the tracker can receive commands over GPRS from the server and to receive commands by SMS from the mobile phone as well. SMS : Set the tracker to send reports and receive commands by SMS only. All event reports are supported in SMS mode. It helps to use the tracker in the area without GPRS coverage. Multiple protocol commands and event reports are can be in one SMS as long as the size less than 140 Bytes. AUTO : Set the tracker to send reports by GPRS mainly and to send reports by SMS as backup when the GPRS connection fails. LOG: Reserved

	GPRS Reconnect	The reconnecting time duration to open GPRS connection. In GPRS Mode, After retrying N minutes with error, the tracker will reboot and do it over. In AUTO Mode, the tracker will switch to SMS mode and send '$RMTK' SMS as a backup solution. Range from "2" to "65535". The unit is 'minute' Default Value="2" Tip: Enlarge this number properly according to the GSM/GPRS environment can reduce the rebooting. In some countries, it takes longer to open a GPRS connection.
	SMS Duration	The time duration to stay in SMS mode when the SwitchMode is set to 'AUTO' mode. After the duration, the tracker will switch back to GPRS mode automatically. Default Value = "180". The unit is 'second'. Range from "180" to "65535" Tip: Set this number smaller will let the tracker back to GPRS mode as soon as possible. Enlarge the 'SMS Report Interval' in the meantime (Ex. 2 hrs = 3600 seconds). It can reduce SMS cost accordingly.
Return Syntax	**$SetSwitchMode,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the Switch Mode as AUTO. →$SetSwitchMode,1234,0000,AUTO,2,180& ←$SetSwitchMode,1234,OK,080603154030&	

Command	**GetSwitchMode**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the Switch Mode setting	
Syntax	**$GetSwitchMode,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetSwitchMode,[Command S/N],[Switch Mode],[GPRS Reconnect],[SMS Duration],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Switch Mode	GPRS : Set the tracker to send reports by GPRS connection only. But the tracker can receive commands over GPRS from the server and to receive commands by SMS from the mobile phone as well. SMS : Set the tracker to send reports and receive commands by SMS only. All event reports are supported in SMS mode. It helps to use the tracker in the area without GPRS coverage. Multiple protocol commands and event reports are can be in one SMS as long as the size less than 140 Bytes. AUTO : Set the tracker to send reports by GPRS mainly and to send reports by SMS as backup when the GPRS connection fails. LOG: Reserved
	GPRS Reconnect	The reconnecting time duration to open GPRS connection. In GPRS Mode, After retrying N minutes with error, the tracker will reboot and do it over. In AUTO Mode, the tracker will switch to SMS mode and send ‘$RMTK’ SMS as a backup solution. Range from “2” to “65535”. The unit is ‘minute’ Default Value=”2” Tip: Enlarge this number properly according to the GSM/GPRS environment can reduce the rebooting. Because in some countries, it takes longer to open a GPRS connection.

	SMS Duration	The time duration to stay in SMS mode when the Switch Mode is set to 'AUTO' mode. After the duration, the tracker will switch back to GPRS mode automatically. Range from "180" to "65535" Default Value = "180". The unit is 'second'. Tip: Set this number smaller will let the tracker back to GPRS mode as soon as possible. Enlarge the 'SMS Report Interval' (Ex. 2 hrs = 3600 seconds) in the meantime. It can reduce SMS cost accordingly.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the Switch Mode of the unit. →$GetSwitchMode,1234,0000& ←$GetSwitchMode,1234,AUTO,2,180,080603153040&	

1.6 GPRSConfig (GPRS Connection Settings)

Command	**SetGPRSConfig**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the GPRS connection Settings.	
Syntax	**$SetGPRSConfig,[Command S/N],[Password],[Dial Up Number],[APN],[GPRS User Name],[GPRS Password],[DNS Server IP address],[Track Mode],[Report Interval],[SYNC Interval],[Resync Interval],[Protocol],[Require Server ACK],[ACK Timeout],[ACK Error Count],[Include Cell ID Info]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value : "0000". Range from "0000" to "9999".
	Dial Up Number	GPRS dialup phone number. It is "*99#" or "*99***1#" in most countries.
	APN	Access Point Name
	GPRS User Name	GPRS Login User Name
	GPRS Password	GPRS Login Password
	DNS Server IP adress	Provided by the operator

	Track Mode	STOP : Stop the GPRS tracking action. The tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. START : The tracker sends GPS data over GPRS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data over GPRS only when the ACC is ON. When the ACC is OFF, the tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK vibration events.
	Report Interval	GPS data Report Interval. The unit is 'second'. Default Value = "30". Range from [10-65535].
	SYNC Interval	SYNC packet Interval: The unit is 'second' Range from [0(Disable)|300-65535]. "0" means to disable SYNC packet. Default = 300 Seconds. Set longer Sync Interval to save GPRS cost. Note: The SYNC packet is now only applicable in 'STOP' mode or when ACC=OFF in 'ACCON' mode or 'VACCON' mode.
	Resync Interval	Compulsory SYNC packet Interval: The unit is 'minute' Range from [0(Disable)|1(Once)|5-65535]. "0" means to disable SYNC packet. "1" means the tracker only sends a SYNC packet every time it open a new GPRS session. Default value = 1 (Once) The Resync packet is now applicable in 'START' mode or when ACC=ON in 'ACCON' more or 'VACCON' mode.
	Protocol	TCP: To use TCP/IP in GPRS communication UDP: To use UDP/IP in GPRS communication Default = TCP

	Require Server ACK	YES: To require the Gateway Server send back an ACK for every event report for confirmation to send next one. NO: Disable requiring Server ACK of every event report Default = NO
	ACK Timeout	Time to wait for the Sever ACK. Everytime when timeout, the tracker will send the event report again.The unit is ‘second’. Range from [5-65535]
	Retry	How many times to retry to send the event report. If it keeps failing, the tracker will re-open the GPRS connection and do it over. The unit is ‘count’. Range from [1-65535]
	Include Cell ID Info	YES: To include Cell ID information in every event report. If enabled, two new columns [PLMN] (Network ID), [Cell ID] (Base station ID) will be added into every event report. NO: Not to include Cell ID information in all event reports.
Return Syntax	**$SetGPRSConfig,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Configure GPRS Parameters as following. →$SetGPRSConfig,1234,0000,*99***1#,internet,web,web,168.95.1.1,START,30,300,60,TCP,NO,0,0,NO& ←$SetGPRSConfig,1234,OK,080603154030&	

Command	**GetGPRSConfig**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the GPRS Connection Settings.	
Syntax	**$GetGPRSConfig,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetGPRSConfig,[Command S/N],[Dial Up Number],[APN],[GPRS User Name],[GPRS Password],[DNS Server IP address],[Enable],[Report Interval],[SYNC Interval],[Resync Interval],[Protocol],[Require Server ACK],[ACK Timeout],[ACK Error Count],[Include Cell ID Info],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Dial Up Number	GPRS dialup phone number.
	APN	Access Point Name
	GPRS User Name	GPRS Login User Name
	GPRS Password	GPRS Login Password
	DNS Server IP adress	Provided by the operator
	Enable	STOP : Stop the GPRS tracking action. The tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. START : The tracker sends GPS data over GPRS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data over GPRS only when the ACC is ON. When the ACC is OFF, the tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK

		vibration events.
	Report Interval	GPS data Report Interval. The unit is ‘second’. Default Value = “30”. Range from [10-65535].
	SYNC Interval	SYNC packet Interval: The unit is ‘second’ Range from [0(Disable)|300-65535]. “0” means to disable SYNC packet. Default = 300 Seconds. Set longer Sync Interval to save GPRS cost. Note: The SYNC packet is now only applicable in ‘STOP’ mode or when ACC=OFF in ‘ACCON’ mode or ‘VACCON’ mode.
	Resync Interval	Compulsory SYNC packet Interval: The unit is ‘minute’ Range from [0(Disable)|1(Once)|5-65535]. “0” means to disable SYNC packet. “1” means the tracker only sends a SYNC packet every time it open a new GPRS session. Default value = 1 (Once) The Resync packet is now applicable in ‘START’ mode or when ACC=ON in ‘ACCON’ more or ‘VACCON’ mode.
	Protocol	TCP: To use TCP/IP in GPRS communication UDP: To use UDP/IP in GPRS communication Default = TCP
	Require Server ACK	YES: To require the Gateway Server send back an ACK for every event report for confirmation to send next one. NO: Disable requiring Server ACK of every event report Default = NO
	ACK Timeout	Time to wait for the Sever ACK. Everytime when timeout, the tracker will send the event report again.The unit is ‘second’. Range from [5-65535]
	Retry	How many times to retry to send the event report. If it keeps failing, the tracker will re-open the GPRS connection and do it over. The unit is ‘count’.

		Range from [1-65535]
	Include Cell ID Info	YES: To include Cell ID information in every event report. If enabled, two new columns [PLMN] (Network ID), [Cell ID] (Base station ID) will be added into every event report. NO: Not to include Cell ID information in all event reports.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the GPRS Parameters as following. →$GetGPRSConfig,1234,0000& ←$GetGPRSConfig,1234,*99***1#,internet,web,web,168.95.1.1,START,30,300,60,TCP,NO,0,0,NO,080603153040&	

1.7 Connection (GSM/GPRS Time-out Redial and Roaming Control)

Command	**SetConnection**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the GSM/GPRS network registration Time-out to redial and Enable/Disable roaming. Caution: Please refer to the 'Optimized Configuration Guideline.pdf' for optimization. If the related connection settings are optimized, the tracker performs better connectivity and shorter disconnect time.	
Syntax	**$SetConnection,[Command S/N],[Password],[Roaming Enable],[GSM Not Ready Re-Connect Time],[GPRS Not Ready Re-Connect Time],[Battery Status Report Interval]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value : "0000". Range from "0000" to "9999".
	Roaming Enable	YES : Enable roaming to any network as long as the SIM card support it. LIST : Enable roaming, and only allow roaming in the operator networks listed in the tracker. NO : Disable roaming. Default Value = "YES"
	GSM Not Ready Re-Connect	If the device can't register to GSM network, the tracker will wait 'GSM Not Ready Re-Connect Time' to reboot. The unit is 'second'.

	Time	Default Value = “210”. Range from “210” to 65535”.
	GPRS Not Ready Re-connect Time	If the device register to GSM network but can not open GPRS connection successfully, the tracker will wait ‘GPRS Not Ready Re-Connect Time’ to reboot. The unit is ‘second’. Default Value = “300”. Range from “300” to 65535”.
	Battery Status Report Interval	Time duration to report Main battery voltages and Backup battery voltages by an event report $LBAT to server. This feature is only applicable in GPRS mode. The unit is ‘minute’ Range from [0(Disable)|30-65535], “0” means not to send $LBAT event report.
Return Syntax	**$SetConnection,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Configure GPRS Parameters as following. →$SetConnection,1234,0000,YES,210,300,1& ←$SetConnection,1234,OK,080603153040&	

Command	**GetConnection**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the GPRS Connection Settings.	
Syntax	**$GetConnection,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetConnection,[Command S/N],[Roaming Enable],[GSM Not Ready Re-Connect Time],[GPRS Not Ready Re-Connect Time],[Battery Status Report Interval],[RTC Time]&**	

Return Values	Command S/N	Return string must include the “Command S/N”.
	Roaming Enable	YES : Enable roaming to any network as long as the SIM card support it. LIST : Enable roaming, and only allow roaming in the operator networks listed in the tracker. NO : Disable roaming. Default Value = “YES”
	GSM Not Ready Re-Connect Time	If the device can’t register to GSM network, the tracker will wait ‘GSM Not Ready Re-Connect Time’ to reboot. The unit is ‘second’. Default Value = “210”. Range from “210” to 65535”.
	GPRS Not Ready Re-connect Time	If the device register to GSM network but can not open GPRS connection successfully, the tracker will wait ‘GPRS Not Ready Re-Connect Time’ to reboot. The unit is ‘second’. Default Value = “300”. Range from “300” to 65535”.
	Battery Status Report Interval	Time duration to report Main battery voltages and Backup battery voltages by an event report $LBAT to server. This feature is only applicable in GPRS mode. The unit is ‘minute’ Range from [0(Disable)|30-65535], “0” means not to send $LBAT event report.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the Connection settings as following. →$GetConnection,1234,0000& ←$GetConnection,1234,YES,210,300,1,080603153040&	

1.8 PriServer (Primary Gateway Server Settings)

Command	**SetPriServer**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the Primary Gateway Server Settings.	
Syntax	**$SetPriServer,[Command S/N],[Password],[Enable],[IP/Domain Name],[Port]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Enable	Default Value : “YES” and can not be changed.
	IP or Domain Name	Primary Gateway Server IP address or Domain Name, if using Domain Name. DNS Server IP address of GPRS connection settings is needed.
	Port	The port number of the Primary Gateway Server to receive GPS data
Return Syntax	**$SetPriServer,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the Primary Gateway Server →$SetPriServer,1234,0000,YES,220.128.135.215,9998& ←$SetPriServer,1234,OK,080603153040&	

Command	**GetPriServer**
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	Get the Primary Gateway Server Settings.
Syntax	**$GetPriServer,[Command S/N],[Password]&**

Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetPriServer,[Command S/N],[Enable],[IP or Domain Name],[Port],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Enable	Default Value : “YES”
	IP or Domain Name	Primary Gateway Server IP address or Domain Name, if using Domain Name. DNS Server IP address of GPRS connection settings is needed.
	Port	The port number of the Primary Gateway Server to receive data
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the Primary Gateway Server →$GetPriServer,1234,0000& ←$GetPriServer,1234,Y, 220.128.135.215,9998,080603153040&	

1.9 GSMConfig (SMS Connection Settings)

Command	**SetGSMConfig**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the SMS connection Settings.	
Syntax	**$SetGSMConfig,[Command S/N],[Password],[Track Mode],[Selection],[Report Interval],[Filter],[Include Cell ID Info]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Track Mode	STOP : Stop the SMS tracking action. The tracker only registers to the GSM network and sends nothing. It stays in stand-by mode and waits to receive SMS commands. START : The tracker sends GPS data by SMS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data by SMS only when the ACC is ON. When the ACC is OFF, the tracker sends nothing.

		VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK vibration events.
	Selection	Range from '0' to '9' : to select the mobile phone number in SMS Tel list to send reports by SMS. All: Sends reports to every phone numbers in the SMS Tel list
	Report Interval	GPS data Report Interval. The unit is 'second'. Default Value = "60". Range from "60" to 65535".
	Filter	YES : Filter the invalid GPS data. Only send the valid GPS data NO : Not to filter GPS data. Send all GPS data, no matter the GPS data is valid or invalid.
	Include Cell ID Info	YES: To include Cell ID information in every event report. If enabled, two new columns [PLMN] (Network ID), [Cell ID] (Base station ID) will be added into every event report. NO: Not to include Cell ID information in all event reports.
Return Syntax	**$SetGSMConfig,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Configure SMS Parameters as following. →$SetGSMConfig,1234,0000,ACCON,0,300,YES,NO& ←$SetGSMConfig,1234,OK,080603153040&	

Command	**GetGSMConfig**
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac

Description	Get the SMS Connection Settings.	
Syntax	**$GetGSMConfig,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetGSMConfig,[Command S/N],[Track Mode],[Selection],[Report Interval],[Filter],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Track Mode	STOP : Stop the SMS tracking action. The tracker only registers to the GSM network and sends nothing. It stays in stand-by mode and waits to receive SMS commands. START : The tracker sends GPS data by SMS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data by SMS only when the ACC is ON. When the ACC is OFF, the tracker sends nothing. VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK vibration events.
	Selection	Range from ‘0’ to ‘9’ : to select the mobile phone number in SMS Tel list to send reports by SMS. All: Sends reports to every phone numbers in the SMS Tel list
	Report Interval	GPS data Report Interval. The unit is ‘second’. Default Value = “60”. Range from “60” to 65535”.
	Filter	YES: Filter the invalid GPS data. Only send the valid GPS data NO: Not to filter GPS data. Send all GPS data, no matter the GPS data is valid or invalid.
	Include Cell ID	YES: To include Cell ID information in every event report. If

	Info	enabled, two new columns [PLMN] (Network ID), [Cell ID] (Base station ID) will be added into every event report. NO: Not to include Cell ID information in all event reports.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get the SMS Parameters as following. →$GetGSMConfig,1234,0000& ←$GetGSMConfig,1234,ACCON,0,300,YES,NO,080603153040&	

1.10 TimeZone (RTC Time Zone offset of the device)

Command	**SetTimeZone**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the TimeZone offset of the tracker and allow it to synchronize RTC Time with GPS UTC Time automatically. Meanwhile you are still able to disable this function and use $SetTimeStamp to synchronize RTC Time with Server Time.	
Syntax	**$SetTimeZone,[Command S/N],[Password],[Enable],[Offset]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Enable	YES: To enable the tracker to synchronize RTC Time with GPS UTC Time according to the Time Zone Offset. NO: To disable the tracker to synchronize RTC Time with GPS UTC Time.
	Offset	Format is +hh:mm or -hh: mm
Return Syntax	**$SetTimeZone,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set TimeZone as GMT+8 →$SetTimeZone,1234,0000,YES,+08:00&	

	←$SetTimeZone,1234,OK,080603153040&

Command	**GetTimeZone**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the TimeZone setting of the device	
Syntax	**$GetTimeZone,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetTimeZone,[Command S/N],[Enable],[Offset],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Enable	YES: To enable the tracker to synchronize RTC Time with GPS UTC Time according to the Time Zone Offset. NO: To disable the tracker to synchronize RTC Time with GPS UTC Time.
	Offset	Format is +hh:mm or -hh: mm
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetTimeZone →$GetTimeZone,1234,0000& ←$GetTimeZone,1234,YES,+8,080603153040&	

1.11 TimeStamp (RTC Time of the device)

Command	**SetTimeStamp**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set RTC Time (Real Time Clock) of the device by the Gateway Server instead of synchronization with GPS UTC Time	
Syntax	**$SetTimeStamp,[Command S/N],[Password],[RTC Time]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Return Syntax	**$SetTimeStamp,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
Examples	SetTimeStamp →$SetTimeStamp,1234,0000,070501143000& ←$SetTimeStamp,1234,OK,080603153040&	

Command	**GetTimeStamp**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the RTCTime of the device	
Syntax	**$GetTimeStamp,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetTimeStamp,[Command S/N],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss

Examples	GetTimeStamp →$GetTimeStamp,1234,0000& ←$GetTimeStamp,1234,070501143000&

1.12 SMSTel (Telephone Number to receive SMS reports)

Command	**SetSMSTel**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the telephone numbers to receive SMS report sent from the tracker in SMS mode.	
Syntax	**$SetSMSTel,[Command S/N],[Password],[Index],[Symbol],[Telephone Number],[TON]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Index	The telephone number index Range from “0” to “9”
	Symbol	”+” is add or modify, and ”-“ is delete.
	Telephone Number	The telephone number to receive SMS event reports. Only the numbers in the list are allowed to receive SMS event reports.
	TON	**T**ype **O**f **N**umber 0: TON=0 (Unknown. This is used when the user or network has no a priori information about the numbering plan. In this case, the Address-Value field is organized according to the network dialling plan, e.g. prefix or escape digits might be present.) 1: TON=1 (International number, majority of GSM networks and cell phones support TON=1. A phone number in international format looks like 46718251358, where the country code is 46 or 886937123456, where the country code is 886)
Return Syntax	**$SetSMSTel,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd |	Command is done. | Invalid Command. |

	Pwd | Param	Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the SMSTel 0 as 0921987654, TON=1 →$SetSMSTel,1234,0000,0,+,0921987654,1& ←$SetSMSTel,1234,OK,080603153040&	

Command	**GetSMSTel**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the telephone numbers to receive SMS report sent from the tracker in SMS mode.	
Syntax	**$GetSMSTel,[Command S/N],[Password],[Index]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Index	The telephone number index Range from “0” to “9”
Return Syntax	**$GetSMSTel,[Command S/N],[Telephone Number],[TON],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Telephone Number	The telephone number to receive SMS event reports. Only the numbers in the list are allowed to receive SMS event reports.
	TON	**T**ype **O**f **N**umber 0: TON=0 (Unknown. This is used when the user or network has no a priori information about the numbering plan. In this case, the Address-Value field is organized according to the network dialling plan, e.g. prefix or escape digits might be present.) 1: TON=1 (International number, majority of GSM networks and cell phones support TON=1. A phone number in international format looks like 46718251358, where the country code is 46 or 886937123456, where the country code is 886)
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetSMSTel Index 1 →$GetSMSTel,1234,0000,1&	

	←$GetSMSTel,1234,0912987654,1,080603153040&

1.13 RoamList (Roaming operator list)

Command	**SetRoamList**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Set the roaming network list. The list will be applied when the 'Roaming Enable' of 'SetConnection' command is set to 'LIST'. The tracker can only register to those roaming networks in the list.	
Syntax	**$SetRoamList,[Command S/N],[Password],[Group],[ID0],[IDI],[ID2],[ID3],[ID4],[ID5],[ID6],[ID7],[ID8],[ID9]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value : "0000". Range from "0000" to "9999".
	Group	Group number. Every group can have 10 sets of network ID. Range from '0' to '4'.
	IDn	Numeric format of operator network. This is the GSM Location Area Identification (LAI) number, which consists of the 3-digit Mobile Country Code (MCC) plus the 2- or 3-digit Mobile Network Code (MNC). Ex. UK Vodafone: 23415, UK O2: 23410. You may ignore the rest of comma symbols after the preferred network ID. Ex: $SetRoamList,1234,0000,0,23415& or $SetRoamList,1234,0000,0,23415,23410&
Return Syntax	**$SetRoamList,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the Roaming List →$SetRoamList,1234,0000,0,23415,23410& ←$SetRoamList,1234,OK,080603153040&	

Command	**GetRoamList**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the telephone numbers to receive SMS report sent from the tracker in SMS mode.	
Syntax	**$GetRoamList,[Command S/N],[Password],[Group]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Group	Group number. Every group can have 10 sets of network ID.
Return Syntax	**$GetRoamList,[Command S/N],[Group],[ID0],[IDI],[ID2],[ID3],[ID4],[ID5],[ID6],[ID7],[ID8],[ID9],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Group	Group number. Every group can have 10 sets of network ID.
	IDn	Numeric format of operator netowork. This is the GSM Location Area Identification (LAI) number, which consists of the 3-digit Mobile Country Code (MCC) plus the 2- or 3-digit Mobile Network Code (MNC). Ex. UK Vodafone: 23415, UK O2: 23410.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetRoamList →$GetRoamList,1234,0000& ←$GetRoamList,1234,0,23415,23410,080603153040&	

1.14 Com (COM port parameters)

Command	**SetCom**	
Models	☑Easytrac ☑Speedtrac ☐OBDtrac	
Description	Set the parameters of two COM ports of Easytrac. It can configure the baudrate of the COM ports in order to let the extended accessories work with Easytrac.	
Syntax	**$SetCom,[Command S/N],[Password],[COM1 Baudrate],[COM2 Baudrate] &**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	COM1 Baudrate	Baudrate of Easytrac COM1. Range from ‘4800’, ’9600’, ‘19200’ to ‘38400’.
	COM2 Baudrate	Baudrate of Easytrac COM2. Range from ‘4800’, ’9600’, ‘19200’ to ‘38400’.
Return Syntax	**$SetCom,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the baudrate of COM ports →$SetCom,1234,0000,9600,38400& ←$SetCom,1234,OK,080603153040&	

Command	**GetCom**	
Models	☑Easytrac ☑Speedtrac ☐OBDtrac	
Description	Get the parameters of two COM ports of Easytrac.	
Syntax	**$GetCom,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.

Return Syntax	**$GetCom,[Command S/N],[COM1 Baudrate],[COM2 Baudrate],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	COM1 Baudrate	Baudrate of Easytrac COM1. Range from ‘4800’, ’9600’, ‘19200’ to ‘38400’.
	COM2 Baudrate	Baudrate of Easytrac COM2. Range from ‘4800’, ’9600’, ‘19200’ to ‘38400’.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetCom →$GetCom,1234,0000& ←$GetCom,1234,9600,38400,080603153040&	

1.15 IDControl (Enable RFID and Immobilizer integration)

Command	**SetIDControl**	
Models	☑Easytrac ☑Speedtrac ☐OBDtrac	
Description	Set the RFID reading to co-work with the immobilizer connecting to DO_4 of Easytrac. Only when there is a valid RFID swiped by the driver, the driver is able to start the vehicle. Use the other command $SetValidID to add or delete valid IDs in the tracker. The tracker will verify a valid ID or an invalid ID by itself. It means either SMS or GPRS communication with the server is not necessary to verify an ID. The valid ID can be read from a RFID reader or an iButton reader. The maximum valid IDs can be up to 200 sets.	
Syntax	**$SetIDControl,[Command S/N],[Password],[Enable],[AccOff Timer],[Immobilizer Reset Timer]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Enable	YES : Enable the valid ID control by RFID reading and the immobilizer connecting to DO_4 of Easytrac. Without a valid RFID or iButton, the DO_4 will be ON to enable the immobilizer to cut the starter circuit. Note: The driver has to keep the ignition key in AccOff status. Then he can swipe the RFID for verification. If the driver swipes

		the RFID in AccOn status, the tracker will send an $LUIN event report but it won't check the ID validity. NO : Disable the valid ID control. WARN: Enable ID Control Warning mode. When ID Control Enable is switched to WARN, whenever a valid ID event (either LUIN/LIBT) appears 120 sec. before and after the appearance of LACN event, the warning buzzer output DO4 will not be activated. If no such ID event appears 120 sec. before or after LACN, warning buzzer DO4 will be activated 120 sec. after LACN event appears. DO4 will be deactivated immediately either by shutting off ACC or a valid ID event has been inputted after its activation.
	AccOff Timer	The timer to set DO_4=ON automatically to cut the starter circuit after the ignition OFF. The unit is 'second'. Range from [0-65535].
	Immobilizer Reset Timer	The timer to set DO_4=ON automatically to cut the starter circuit after a valid RFID detected. The driver has to start the vehicle within this duration after he swipes a valid RFID. Otherwise, he has to swipe the RFID again. The unit is 'second'. Range from [10-65535].
Return Syntax	**$SetIDControl,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Enable the RFID and immobilizer integration →$SetIDControl,1234,0000,YES,30,15& ←$SetIDControl,1234,OK,080603153040&	

Command	**GetIDControl**
Models	☑Easytrac ☑Speedtrac ☐OBDtrac
Description	Get the configuration of IDControl
Syntax	**$GetIDControl,[Command S/N],[Password]&**

Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetIDControl,[Command S/N],[Enable],[AccOff Timer],[Immobilizer Reset Timer],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Enable	YES : Enable the valid ID control by RFID reading and the immobilizer connecting to DO_4 of Easytrac. Without a valid RFID or iButton, the DO_4 will be ON to enable the immobilizer to cut the starter circuit. Note: The driver has to keep the ignition key in AccOff status. Then he can swipe the RFID for verification. If the driver swipes the RFID in AccOn status, the tracker will send an $LUIN event report but it won’t check the ID validity. NO : Disable the valid ID control.
	AccOff Timer	The timer to set DO_4=ON automatically to cut the starter circuit after the ignition OFF. The unit is ‘second’. Range from [0-65535].
	Immobilizer Reset Timer	The timer to set DO_4=ON automatically to cut the starter circuit after a valid RFID detected. The driver has to start the vehicle within this duration after he swipes a valid RFID. Otherwise, he has to swipe the RFID again. The unit is ‘second’. Range from [10-65535].
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetIDControl →$GetIDControl,1234,0000& ←$GetIDControl,1234,YES,30,15,080603153040&	

1.16 ValidID (Valid Driver ID List)

Command	**SetValidID**
Models	☑Easytrac ☑Speedtrac ☐OBDtrac
Description	Set the valid RFID numbers in the tracker. The list will be applied to check the

	Driver ID read by the RFID reader. The administrator can use the command to add valid IDs to the tracker or to remove them from the tracker. It can store up to max. 200 IDs. The ID can be added or deleted one by one or by batch. Note that the IDs are not stored in the tracker in sequence. The tracker will deal with the insert and delete by itself as long as the ID is unique.	
Syntax	**$SetValidID,[Command S/N],[Password],[Symbol],[ID0],[IDl],[ID2],[ID3],[ID4],[ID5],[ID6],[ID7],[ID8],[ID9]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Symbol	”+” is add and ”-“ is delete. Note: $SetValidID,[Command S/N],[Password],-,All& is the command to delete all IDs in the tracker.
	IDn	The ID is in ASCII format and is composed of alphabet letters and numbers. When the driver swipes hi RFID or iButton, the tracker will check if the ID is in the list. If yes, the DO_4 will be OFF to enable the starter circuit in order to let the driver ignition ON the vehicle. ID of RFID tag or RFID card: 8 bytes ID of iButton: 12 bytes
Return Syntax	**$SetValidID,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the valid ID list →$SetValidID,1234,0000,+,12345678,5678ABCD& ←$SetValidID,1234,OK,080603153040&	

Command	**GetValidID**
Models	☑Easytrac ☑Speedtrac ☐OBDtrac
Description	Get the valid ID list from the tracker. The IDs are not stored in the tracker in sequence. The tracker will reply several times of $GetValidID return strings due to there are up to max. 200 sets of ID in a tracker.

Syntax	**$GetValidID,[Command S/N],[Password] &**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetValidID,[Command S/N],[ID],[ID],…, [ID],[RTC Time]&** **$GetValidID,[Command S/N],OK, [RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	IDn	The ID is in ASCII format and is composed of alphabet letters and numbers. When the driver swipes hi RFID or iButton, the tracker will check if the ID is in the list. If yes, the DO_4 will be OFF to enable the starter circuit in order to let the driver ignition ON the vehicle. ID of RFID tag or RFID card: 8 bytes ID of iButton: 12 bytes
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	GetValidID →$GetValidID,1234,0000& ←$GetValidID,1234,0,12345678,5678ABCD,080603153040& ←$GetValidID,1234,OK,080603153041&	

1.17 EventMask

Command	**SetEventMask**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set event mask for users to customize event on the Tracker. In the Main Setting Menu, two more columns (H&I) are added under the 1. Basic Settings menu for event mask configuration.	
Syntax	**$SetEventMask,[Command S/N],[Password], [MASK1], [MASK2]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	MASK1	Event Mask bit value 1

	MASK2	Event Mask bit value 2
Return Syntax	**$SetEventMask,[sequence number],OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set EventMask →$SetEventMask,1234,0000,FFFFFFFF,FFFFFFFF& Setting is complete ←$SetEventMask,1234,OK,080603153040&	

Command	**GetEventMask**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get Event Mask bit values stored in the tracker.	
Syntax	**$GetEventMask,[sequence number],[password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetEventMask,[sequence number],[MASK1],[MASK2]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	MASK 1	Event Mask bit value 1
	MASK 2	Event Mask bit value 2
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get EventMask →$GetEventMask,1234,0000& ←$GetEventMask,1234,FFFFFFFF,FFFFFFFF,080603153040&	

1.18 OBDlimit

Command	**SetOBDlimit**	
Models	☐Easytrac ☐Speedtrac ☑OBDtrac	
Description	This is for setting up maximum RPM and coolant temperature under the config menu. Both original report interval settings and original LOBD events are removed. The tracker will keep the record when LRPN and LRPF data are sent.	
Syntax	**$SetOBDLimit,<SEQ>,<PW>,<RPM LIMIT>,<COOLANT TEMP. LIMIT>&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	RPM Limit	Set the maximum RPM value before sending out warning events.
	Coolant Temperature Limit	Set the maximum coolant temperature limit value before sending out warning events.
Return Syntax	**$SetOBDLimit,<SEQ>,OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Setting RPM limit to 5 and Coolant Temp. limit to 35 degrees celsius →$SetOBDLimit,1234,0000,2500,120& Setting complete ←$SetOBDLimit,1234,OK,080603153040&	
Command	**GetOBDlimit**	
Models	☐Easytrac ☐Speedtrac ☑OBDtrac	
Description	Get the RPM and Cooland temperature limit values stored in the tracker.	
Syntax	**$GetOBDLimit,<SEQ>,<PW>&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”.

		Range from “0000” to “9999”.
Return Syntax	**$GetOBDLimit,<SEQ>,<RPM LIMIT>,<COOLANT TEMP. LIMIT>&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	RPM Limit	Set the maximum RPM value before sending out warning events.
	Coolant Temperature Limit	Set the maximum coolant temperature limit value before sending out warning events.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Get OBDlimit →$GetOBDLimit,1234,0000& ←$GetOBDLimit,1234,OK,2500,120,080603153040&	

1.19 ATR

Command	**ATR**	
Models	☐Easytrac ☐Speedtrac ☑OBDtrac	
Description	For use with OBDtrac only. Command to search vehicle OBDII protocol as well as setting mileage value into the tracker.	
Syntax	**$ATR,<SEQ>,<PW>,< MILEAGE>&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Mileage	Set the total mileage value displayed on the odometer of vehicle.
Return Syntax	**$ATR,<SEQ>,OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param | MIL | FAIL	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters. | Mileage value not 0 | Fail to input mileage.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss

Examples	Setting the total mileage value to be 15000. →$ATR,1234,0000,15000& Setting complete ←$ATR,1234,OK,080603153040&

1.20 TempLimit

Command	**$SetTempLimit**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Command to set upper and lower limit warnings for temperature sensors.	
Syntax	**$SetTempLimit,<SEQ>,<PW>,<ID [1|2]>,<Upper Limit Enable [YES|NO]>,<Upper Limit>,<Lower Limit Enable [YES|NO]>,<Lower Limit>&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	ID	Temperature sensor selection, 1 or 2.
	Upper Limit Enable	Enable/Disable temperature upper limit warning.
	Upper Limit	Temperature upper limit value.
	Lower Limit Enable	Enable/Disable temperature lower limit warning.
	Lower Limit	Temperature lower limit value.
Return Syntax	**$SetTempLimit,<SEQ>,OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param |	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters. |
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Setting upper and lower temperature limit to be 30C and -20C. →$SetTempLimit,1234,0000,1,YES,30,YES,-20&	

	Setting complete ←$SetTempLimit,1234,OK,080603153040&	
Command	**$GetTempLimit**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get upper and lower limit warning parameters for temperature sensors.	
Syntax	**$GetTempLimit,<SEQ>,<PW>&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetTempLimit,<SEQ>,<PW>,<ID [1|2]>,<Upper Limit Enable [YES|NO]>,<Upper Limit>,<Lower Limit Enable [YES|NO]>,<Lower Limit>&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	ID	Temperature sensor selection, 1 or 2.
	Upper Limit Enable	Enable/Disable temperature upper limit warning.
	Upper Limit	Temperature upper limit value.
	Lower Limit Enable	Enable/Disable temperature lower limit warning.
	Lower Limit	Temperature lower limit value.
Examples	Get temperature limit Setting upper and lower temperature limit to be 30C and -20C. →$GetTempLimit,1234,0000& ←$GetTempLimit,1234,1,YES,30,YES,-20&	

1.21 PowerConfig

Command	**SetPowerConfig**
Models	☑Easytrac ☑Speedtrac ☑OBDtrac

Description	Set parameters to configure GPS fix timeout, vibration sensor time out, and sleep mode functions.	
Syntax	**$SetPowerConfig,[sequence number],[password],[GPS Fix Timeout],[Vibration Timeout],[Power Control]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	GPS Fix Timeout	Time interval of GPS signal fixing before tracker reboots
	Vibration Timeout	Time interval for vibration sensor to determine the tracker is still and enters sleep mode
	Power Control	Power control parameters when vibration sensor determines the tracker is still and not in use. 0. Disable: tracker will not enter any power off mode. 1. Power Off: cuts off tracker power completely. 2. Standby: GSM module remains registered, will receive SMS texts, GPRS connection shuts down, GPS module remains online. 3. Sleep: GSM module remains registered, will receive SMS texts, GPRS connection shuts down, GPS module shuts down. 4. Deep Sleep: GSM module enters sleep mode, cannot receive SMS texts, GPS module shuts down.
Return Syntax	**$SetPowerConfig,[sequence number],OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Power Config →$SetPowerConfig,1234,0000,60,30,0& Setting is complete	

	←$SetPowerConfig,1234,OK,080603153040&	
Command	**GetPowerConfig**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get power configuration parameters from the tracker	
Syntax	**$GetPowerConfig,[sequence number],[password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetPowerConfig,[sequence number],[password],[GPS Fix Timeout],[Vibration Timeout],[Power Control]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	GPS Fix Timeout	Time interval of GPS signal fixing before tracker reboots
	Vibration Timeout	Time interval for vibration sensor to determine the tracker is still and enters sleep mode
	Power Control	Power control parameters when vibration sensor determines the tracker is still and not in use. 0. Disable: tracker will not enter any power off mode. 1. Power Off: cuts off tracker power completely. 2. Standby: GSM module remains registered, will receive SMS texts, GPRS connection shuts down, GPS module remains online. 3. Sleep: GSM module remains registered, will receive SMS texts, GPRS connection shuts down, GPS module shuts down. 4. Deep Sleep: GSM module enters sleep mode, cannot receive SMS texts, GPS module shuts down.
Examples	Get Power Config →$GetPowerConfig,1234,0000& ←$GetPowerConfig,1234,60,30,0,080603153040&	

1.22 DataMode

Command	**SetDataMode**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set parameters to switch between binary and text data mode	
Syntax	**$SetDataMode,[sequence number],[password],[Data Mode]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	Data Mode	0. Text Mode 1. Binary Mode
Return Syntax	**$SetDataMode,[sequence number],OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Data Mode →$SetDataMode,1234,0000,1& Setting is complete ←$SetDataMode,1234,OK,080603153040&	
Command	**GetDataMode**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get Data Mode configuration.	
Syntax	**$GetDataMode,[sequence number],[password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetDataMode,[sequence number],[password],[Data Mode]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.

	Data Mode	0. Text Mode 1. Binary Mode
Examples	Get Data Mode →$GetDataMode,1234,0000& ←$GetDataMode,1234,1,080603153040&	

1.23 AI Mode

Command	**SetAIMode**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set modes to configure analog input.	
Syntax	**$SetAIMode,[sequence number],[password],[AI1 Mode],[AI2 Mode]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	AI1 Mode	Set analog input mode for AI1. N= None, V= Voltage, T= Temperature, B= Backup battery power, P= Main power.
	AI2 Mode	Set analog input mode for AI2. N= None, V= Voltage, T= Temperature, B= Backup battery power, P= Main power.
Return Syntax	**$SetAIMode,[sequence number],OK&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set AI Mode →$SetAIMode,1234,0000,M,B& Setting is complete	

	←$SetAIMode,1234,OK,080603153040&	
Command	**GetAIMode**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get AI Mode configuration from tracker.	
Syntax	**$GetAIMode,[sequence number],[password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$GetAIMode,[sequence number],[password],[AI1 Mode],[AI2 Mode]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	AI1 Mode	Set analog input mode for AI1. N= None, V= Voltage, T= Temperature, B= Backup battery power, P= Main power.
	AI2 Mode	Set analog input mode for AI2. N= None, V= Voltage, T= Temperature, B= Backup battery power, P= Main power.
Examples	Get AI Mode →$GetAIMode,1234,0000& ←$GetAIMode,1234,M,B,080603153040&	

2、Action

2.1 ActGPRSTrack（GPRS Tracking Action）

Command	**ActGPRSTrack**
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	Command the tracker to continuously report GPS data to the Gateway Server. The tracker sends GPS data and all I/O status over GPRS to the Gateway Server. Send an ActSaveCFG command right after the ActGPRSTrack command to save new setting to the Flash Memory to apply it.

Syntax	**$ActGPRSTrack,[Command S/N],[Password],[Track Mode],[Report Interval],[Sync Interval],[Filter]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Track Mode	STOP : Stop the GPRS tracking action. The tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. START : The tracker sends GPS data over GPRS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data over GPRS only when the ACC is ON. When the ACC is OFF, the tracker sends SYNC packet by SYNC interval to the Gateway server to keep alive. VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK vibration events.
	Report Interval	GPS Transmission Interval. The unit is ‘second’. Default Value = “30”. Range from “15” to 65535”.
	SYNC Interval	SYNC packet Interval: The unit is ‘second’ Range from [0(Disable)|300-65535]. “0” means to disable SYNC packet. Default = 300 Seconds. Set longer SYNC Interval to save GPRS cost. Note: The SYNC packet is now only applicable in ‘STOP’ mode or when ACC=OFF in ‘ACCON’ mode or ‘VACCON’ mode.
	Filter	YES : Filter the invalid GPS data. Only send the valid GPS data NO : Not to filter GPS data. Send all GPS data, no matter the GPS data is valid or invalid.
Return Syntax	**$ActGPRSTrack,[Command S/N],OK,[RTC Time]&**	

Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Data Structure	Real time GPS Data Report format: **$RGTK,[UnitID],[UTCTime],[A\|V],[Longitude],[E\|W],[Latitude],[N\|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&** Resent format: (When the GPRS is recovered, the tracker will resend all GPS data logs to the Gateway Server automatically.) **$LGTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**	
	UnitID	UnitID
	UTC Time	GPS Date and Time. The format is DDMMYYHHmmss
	A|V	GPS Data is Valid or Invalid. “A” = valid, “V” = invalid.
	Longitude	GPS Longitude
	E/W	“E”= East, “W” = West.
	Latitude	GPS Latitude
	N/S	“N” = North, “S” =South
	Satellite Numbers	Satellite Numbers acquired
	GSM Signal	GSM Signal strength, From 0 to 31, Unknown: 99 or other numbers
	Angle	GPS Angle, unit = degree
	Speed	GPS Speed, unit = knot
	Mileage	Accumulated Mileage of the tracker. It is calculated by GPS data. Unit = ‘mile’. Maximum value: 999999.9
	AI1	Analog Input#1 Voltages. Range from 0~12 or 0~24. Or RPM
	AI2	Analog Input#2 Voltages. Range from 0~12 or 0~24. Or Coolant Temperature

	DI	Summary reading of Digital Input status. Value of DI ON reading: DI0=1, DI1=2, DI2=4, DI3=8, DI4=16, DI5=32, DI6=64 If the DI reading = 6, it means DI1 and DI2 are ON DI0=0 means the tracker using the vehicle power. DI0=1 means the tracker using the backup battery power. Thus the DI reading in every report is an even number if the tracker uses vehicle power. And it becomes an odd number if the tracker uses backup battery power.
	DO	Summary reading of Digital Output status. Value of DO ON reading: DO1=2, DO2=4, DO3=8, DO4=16 If the DI reading = 10, it means DO1 and DO3 are ON
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Command the tracker to send GPS data to the Gateway Server over GPRS every 30 seconds only when the ACC (Ignition) is ON. When the ACC is OFF, send SYNC packet to the Gateway Server every 600 seconds. →$ActGPRSTrack,1234,0000,ACCON,30,600,NO& ←$ActGPRSTrack,1234,OK,080603153040& →$ActSaveCFG,1234,0000& ←$ActSaveCFG,1234,OK,080603153041&	

2.2 ActGSMTrack（SMS Tracking Action）

Command	**ActGSMTrack**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Command the tracker to report GPS data by SMS to the telephone numbers in SMS Tel list.	
Syntax	**$ActGSMTrack,[Command S/N],[Password],[Track Mode],[Selection],[Report Interval],[Filter]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.

	Track Mode	STOP : Stop the SMS tracking action. The tracker only registers to the GSM network and sends nothing. It stays in stand-by mode and waits to receive SMS commands. START : The tracker sends GPS data by SMS as long as it is powered ON. No matter the ACC (Ignition) is ON or OFF. ACCON : The tracker sends GPS data by SMS only when the ACC is ON. When the ACC is OFF, the tracker sends nothing. VACCON : Report RGTK at either LGSK (vibration on) or LACN (engine on), and stop reporting RGTK when both LGEK (vibration off) and LACF (engine off) events appear. If actual ACC is not connected, report RGTK between LGSK and LGEK vibration events.
	Selection	Range from '0' to '9' : to select the mobile phone number in SMS Tel list to send reports by SMS. All: Sends reports to every phone numbers in the SMS Tel list
	Report Interval	GPS data Report Interval. The unit is 'second'. Default Value = "60". Range from "60" to 65535".
	Filter	YES : Filter the invalid GPS data. Only send the valid GPS data NO : Not to filter GPS data. Send all GPS data, no matter the GPS data is valid or invalid.
Return Syntax	**$ActGSMTrack,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Data Structure (SMS)	**$RMTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&** Please refert to 2.1 ActGPRSTrack for data details	
Examples	Command the tracker to send GPS data by SMS.	

	→$ActGSMTrack,1234,0000,START,0,60,YES& ←$RMTK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

2.3 ActGetPos（Get the current GPS postion Action）

Command	**ActGetPos**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Command the tracker to report a latest GPS position to the Gateway Server over GPRS or by SMS. When send this command by GPRS, the tracker will reply position by GPRS. When send it by SMS, the tracker will reply position by SMS accordingly.	
Syntax	**$ActGetPos,[Command S/N],[Password],[Include Cell ID Info]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Include Cell ID Info	YES: To include Cell ID information in every event report. If enabled, two new columns [PLMN] (Network ID), [Cell ID] (Base station ID) will be added into every event report. NO: Not to include Cell ID information in all event reports. Note: This parameter of $ActGetPos is optional to have Cell ID information besides the regular event reports configured by either $SetGPRSConfig or $SetGSMConfig. It helps the user to get Cell ID only in emergency.
Return Syntax	**$ActGetPos,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Data Structure	**$LGPS,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&** Please refert to 2.1 ActGPRSTrack for data details	

Examples	Command the tracker to send current GPS data to the Gateway Server over GPRS. →$ActGetPos,1234,0000& ←$LGPS,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

2.4 ActSaveCFG（Save Basic Configuration to Flash memory）

Command	**ActSaveCFG**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Save the Basic Configuration (All Properties in Ch.1) values to Flash memory. All Basic Configuration changes must follow an ActSaveCFG command to save to the Flash Memory to apply it.	
Syntax	**$ActSaveCFG,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActSaveCFG,[Command S/N],OK,[RTC Time]&** Note: The tracker reboots with the new configuration right away when it receives ActSaveCFG command and send return string after rebooting.	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Save the basic configuration and reboot the tracker. →$ActSaveCFG,1234,0000& ←System reboot ←$ActSaveCFG,1234,OK,080603153040&	

2.5 ActSaveVarCFG（Save Function Variable Configuration to Flash memory）

Command	**ActSaveVarCFG**
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac

Description	Save the Variable Configuration values to the flash memory. Ex: Digital output status or Mileage Report Settings. If the digital output is changed by ActSetDigitalOut command and follow by an ActSaveVarCFG command. No matter the tracker is rebooted by other commands or by watchdog later. The tracker will remain the new digital output status according to the new Variable Configuration.	
Syntax	**$ActSaveVarCFG,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActSaveVarCFG,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Save the Variable Configuration. →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153040&	

2.6 ActReboot（Reboot Action）

Command	**ActReboot**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Reboot the tracker	
Syntax	**$ActReboot,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActReboot,[Command S/N],OK,[RTC Time]&** Note: The tracker reboots right away when it receives ActReboot command and send return string after rebooting.	
Return Values	Command S/N	Return string must include the “Command S/N”.

	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Reboot the tracker →$ActReboot,1234,0000& ←$ActReboot,1234,OK,080603153040&	

2.7 ActUpgradeFW（Upgrade Firmware OTA）

Command	**ActUpgradeFW**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Use the command to send the URL of the new firmware image file to the tracker in order to update the firmware remotely by GPRS	
Syntax	**$ActUpgradeFW,[Command S/N],[Password],[URL]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	URL	Put the new firmware on a web server and come out the URL. (Ex. http://211.79.38.93/970709.img or http://211.79.38.93:8080/970709.img). Or you may use the URL provided by ORION. Please note that you may have to rename the .img to .zip or other file name in order to pass the firewall. (Ex. http://211.79.38.93/970709.zip). It supports http protocol only, but ftp.
Return Syntax	When the tracker receives the command without problem, it will reply the following acknowledgement first. **$ActUpgradeFW,[Command S/N],OK,[RTC Time]&** If the firmware update is successful, it replies the following information after booting by the new firmware. All configuration will be remained the same and won’t effected by the new firmware. **$ActUpgradeFW,[Command S/N],DONE,<UnitID>,<IMEI>,<IMSI>,<HW Vsion >,<FW Version>,<RTC Time>&** If the firmware download fails, it replies the following message. In this case, you	

	have to send $ActUpgradeFW again to do it over. **$ActUpgradeFW,[Command S/N],FAIL,[RTC Time]&** If the firmware download is in progress or parameters error, it replies **$ActUpgradeFW,[Command S/N],Param,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Upgrade the firmware of the tracker →$ActUpgradeFW,1234,0000,http://211.79.38.93/970806E.img& ←$ActUpgradeFW,1234,OK,080603153040& ←$ActUpgradeFW,1234,DONE,1234567890,352021005322607,466922002606092,V6,970805E,080603154025&	

2.8 ActGetLog（Get System Logs Action）

Command	**ActGetLog**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Get the system logs in the tracker. This feature is only for advanced diagnosis.	
Syntax	**$ActGetLog,[Command S/N],[Password],[Log Type],[StartTimeStamp],[EndTimeStamp]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Log Type	S: System Logs
	StartTimeStamp	All system logs are recorded according to device RTCTime. The format is YYMMDDHHmmss
	EndTimeStamp	All system logs are recorded according to device RTCTime. The format is YYMMDDHHmmss

Return Syntax	**$ActGetLog,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Return Logs	**$log,[TimeStamp],[EventType]&**	
Return Log Values	TimeStamp	All system logs are recorded according to device RTCTime. The format is YYMMDDHHmmss
	EventType	Types of system events (For advanced diagnosis only): evtGSMOnline, evtGSMOffline, evtGPRSOnline, evtGPRSOffline, evtSessionOnline, evtSessionOffline, evtGPRSSyncCmdOK, evtGPSFix, evtGPSFail, evtWatchdogReset, evtGSMLACChanged,etc
Examples	Get all of system logs in the tracker →$ActGetLog,1234,0000,S,070628180000,070628183500& ←$ActGetLog,1234,OK,080603153040& ←$log,080603181054,evtGPSFix&	

2.9 ActClearLog（Clear System Logs and Data Logs Action）

Command	**ActClearLog**	
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac	
Description	Clear all logs including system logs and data logs in the tracker. Before using this command, make sure all logs are useless to be deleted. This protocol command corresponds to the console command ‘CLEARLOG’	
Syntax	**$ActClearLog,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActClearLog,[Command S/N],OK,[RTC Time]&**	

Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Clear all of logs in the tracker →$ActClearLog,1234,0000& ←$ActClearLog,1234,OK,080603153040&	

2.10 ActSetDigitalOut（Change Digital Output Status Action）

Command	**ActSetDigitalOut**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Use this command to change the output status as ON or OFF of the tracker. The Digital Output port can be connected to an Immobilizer or a Door Lock/Unlock controller. Send an ActSaveVarCFG command right after the ActSetDigitalOut command to save new setting to the Flash Memory to apply it.	
Syntax	**$ActSetDigitalOut,[Command S/N],[Password],[Digital Output Port No.],[Status]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Digital Output Port No.	“1” or “4”
	Status	The Status of the Digital Output Port. Values: ON: to close the Digital Output Port circuit OFF: to open the Digital Output Port circuit
Return Syntax	**$ActSetDigitalOut,[Command S/N],OK,[Digital Output Port No.],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	Digital Output	“1” or “4”

	Port No.	
	Status	The Status of the Digital Output Port. Values: ON: to close the Digital Output Port circuit OFF: to open the Digital Output Port circuit
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Digital Out 1 to ON →$ActSetDigitalOut,1234,0000,1,ON& ←$ActSetDigitalOut,1234,OK,1,ON,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.11 ActPulseOut（Send Digital Pulses via Output Ports Action）

Command	**ActPulseOut**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Use this command to send digital pulses through the Digital Output port of the tracker. It can control some types of Door Lock/Unlock controller.	
Syntax	**$ActPulseOut,[Command S/N],[Password],[Digital Output Port No.],[State],[Duration],[Pulse Count]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Digital Output Port No.	“1” or “4”
	State	ON/OFF. It has to be the opposite state of the digital output state at the moment. Otherwise, the command will be ignored.
	Duration	Duration of the pulse, the unit is second. When set to 1 sec, it will generate a 1sec ON and 1sec OFF pulse. When set to 2 sec, it will generate a 2sec ON and 2sec OFF pulse.
	Pulse	To generate how many pulses from the original state and go back to it.
Return Syntax	**$ActPulseOut,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd |	Command is done. | Invalid Command. |

	Pwd | Param	Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Send three 1-second ON pulses through Digital Output Port 1 →$ActPulseOut,1234,0000,1,ON,1,3& ←$ActPulseOut,1234,OK,080603153040&	

2.12 ActSetTotalMile（Set the Total Mileage）

Command	**ActSetTotalMile**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the Total Mileages (Accumulated Mileages) to the tracker. Send an ActSaveVarCFG command right after the ActSetTotalMile command to save new setting to the Flash Memory and to apply it.	
Syntax	**$ActSetTotalMile,[Command S/N],[Password],[Total Mileage]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Total Mileage	The accumulated mileages of the tracker.
Return Syntax	**$ActSetTotalMile,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Total Mileage as ’50000’ →$ActSetTotalMile,1234,0000,50000& ←$ActSetTotalMile,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.13 ActGetTotalMile（Get the Total Mileage）

Command	**ActGetTotalMile**

Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the Total Mileages (Accumulated Mileages) from the tracker.	
Syntax	**$ActGetTotalMile,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetTotalMile,[Command S/N],[Total Mileage],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Total Mileage	The accumulated mileages of the tracker.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetTotalMile,1234,0000& ←$ActGetTotalMile,1234,1200.3245,080603153040&	

2.14 ActSetMileLimit（Set the Mileage Limitation）

Command	ActSetMileLimit	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the Mileage Limit to the tracker. Send an ActSaveVarCFG command right after the ActSetMileLimit command to save new setting to the Flash Memory and to apply it. When the Total Mileage is more than ‘Mileage Limitation’, it triggers an ‘evtOverMile’ event and send a ‘$LVML’ Event Report the server. Then the Mileage Limitation is reset to 0. You can use this trigger to send an alert for periodic maintainance reminding.	
Syntax	**$ActSetMileLimit,[Command S/N],[Password],[Mileage Limitation]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Mileage Limitation	A mileage limitation to send a ‘$LVML’ Event Report when Total Mileage reaches this limitation. Then it will be reset to 0.

Return Syntax	**$ActSetMileLimit,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Mileage Limitation as ’50500’ →$ActSetMileLimit,1234,0000,50500& ←$ActSetMileLimit,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.15 ActGetMileLimit（Get the Mileage Limitation）

Command	**ActGetMileLimit**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the Mileage Limitation form the tracker	
Syntax	**$ ActGetMileLimit,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetMileLimit,[Command S/N],[Mileage Limitation],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Mileage Limitation	A mileage limitation to send a ‘$LVML’ Event Report when Total Mileage reaches this limitation. Then it will be reset to 0.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetMileLimit,1234,0000& ←$ActGetMileLimit,1234,50500,080603153040&	

2.16 ActSetDistance（Set the Distance Tracking Interval）

Command	**ActSetDistance**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the Distance Tracking Interval (mile) to the tracker. Send an ActSaveVarCFG command right after the ActSetDistance command to save new setting to the Flash Memory and to apply it. When the 'Distance Track Interval' =0, the tracker disables the 'Track by Distance' function. When the 'Distance Tracker Interval' >0, the tracker enables the 'Track by Distance' function and it triggers an 'evtDistance' event and send a '$LDIS' event report the server by every 'Distance Track Interval'.	
Syntax	**$ActSetDistance,[Command S/N],[Password],[Distance Track Interval]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000". Range from "0000" to "9999".
	Distance Track Interval	A distance interval to send a '$LDIS' event report. The unit is 'mile'.
Return Syntax	**$ActSetDistance,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Distance Track Interval as '10' to track the vehicle every 10 miles. →$ActSetDistance,1234,0000,10& ←$ActSetDistnace,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.17 ActGetDistance（Get the Distance Tracking Interval）

Command	**ActGetDistance**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the Distance Tracking Interval form the tracker	
Syntax	**$ ActGetDistance,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetDistance,[Command S/N],[Distance Track Interval],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Distance Track Interval	A distance interval to send a ‘$LDIS’ event report. The unit is ‘mile’
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetDistance,1234,0000& ←$ActGetDistance,1234,10,080603153040&	

2.18 ActSetDevAng（Set the Deviation Angle setting of the tracker）

Command	**ActSetDevAng**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the Diviation Angle to the tracker. Send an ActSaveVarCFG command right after the ActSetDevAng command to save new setting to the Flash Memory and to apply it. When the ‘Dev Angle’ =0, the tracker disables the ‘Track by Deviation Angle’ function. When the ‘Dev Angle’ >0, the tracker enables the ‘Track by Deviation Angle’ function and it triggers an ‘evtDevAng’ event and send a ‘$LANG’ event report the server everytime the tracker changes it’s direction more than the ‘Deviation Angle’.	
Syntax	**$ActSetDevAng,[Command S/N],[Password],[Dev Angle]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”.

		Range from “0000” to “9999”.
	Dev Angle	A deviation Angle of the tracker to send a ‘$LANG’ event report. “0” means disable to trigger it. The unit is ‘degree’ Default = “0”
Return Syntax	**$ActSetDevAng,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Deviation Angle as ’45 degree’ to track the vehicle. →$ActSetDevAng,1234,0000,45& ←$ActSetDevAng,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.19 ActGetDevAng（Get the Deviation Angle setting）

Command	**ActGetDevAng**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the Deviation Angle setting form the tracker	
Syntax	**$ ActGetDevAng,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetDevAng,[Command S/N],[Dev Angle],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Dev Angle	A deviation Angle of the tracker to send a ‘$LANG’ event report. “0” means disable to trigger it. The unit is ‘degree’ Default = “0”
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetDevAng,1234,0000& ←$ActGetDevAng,1234,45,080603153040&	

2.20 ActSetIdle（Set the Idle settings）

Command	**ActSetIdle**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the tracker to generate Idle event and Move event. Send an ActSaveVarCFG command right after the ActSetIdle command to save new setting to the Flash Memory and to apply it. When the 'Enable' =YES, the tracker enables the idling and moving detection by GPS speed. It triggers an 'evtIdle' event and sends a '$LIDL' event report the server by every 'Report Time Interval' when the GPS speed < 5mph for more than 'Idle Duration'. Then it triggers an 'evtMove' event and sends a '$LMOV' when the GPS speed >5mph and moving > 0.02 miles (30 meters)	
Syntax	**$ActSetIdle,[Command S/N],[Password],[Enable],[Idle Duration],[Report Time Interval]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000". Range from "0000" to "9999".
	Enable	YES : Enable the tracker to generate 'logIdle' and 'logMove' event reports. NO : Disable the tracker to generate 'logIdle' and 'logMove' event reports. Default Value = "NO"
	Idle Duration	If the GPS speed < 5mph for more than 'Idle Duration'. It sends a '$LIDL' event report by every 'Report Time Interval'. The unit is 'second'. Range from "10" to 65535".
	Report Time Interval	A time interval to keep sending '$LIDL' event reports if the tracker keep idling. The unit is 'second'.
Return Syntax	**$ActSetIdle,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss

Examples	Set the tracker to generate Idle reports when the tracker idles more than 30 seconds and keep sending it every 60 seconds. →$ActSetIdle,1234,0000,YES,30,60& ←$ActSetIdle,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,0806153041&

2.21 ActGetIdle（Get the Idle settings）

Command	**ActGetIdle**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the Idle settings form the tracker	
Syntax	**$ ActGetIdle,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetIdle,[Command S/N],[Enable],[Idle Duration],[Report Time Interval],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Enable	YES : Enable the tracker to generate ‘$LIDL’ and ‘$LMOV’ event reports. NO : Disable the tracker to generate ‘$LIDL’ and ‘$LMOV’ event reports. Default Value = “NO”
	Idle Duration	If the GPS speed < 5mph for more than ‘Idle Duration’. It sends a ‘$LIDL’ event report by every ‘Report Time Interval’. The unit is ‘second’. Range from “10” to 65535”.
	Report Time Interval	A time interval to keep sending ‘$LIDL’ event reports if the tracker keep idling. The unit is ‘second’.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetIdle,1234,0000& ←$ActGetIdle,1234,YES,30,60,080603153040&	

2.22 ActSetSpeed（Set the Speeding settings）

Command	**ActSetSpeed**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Set the tracker to generate SpeedIn event and SpeedOut event. Send an ActSaveVarCFG command right after the ActSetSpeed command to save new setting to the Flash Memory and to apply it. When the 'Enable' =YES, the tracker enables the speeding detection by GPS speed. It triggers an 'evtSpeedIn' event and sends a '$LSIN' event report the server by every 'Report Time Interval' when the GPS speed > 'Speed Limitation' for more than 'Speed Limitation Duration'. Then it triggers an 'evtSpeedOut' event and sends a '$LSUT' when the GPS speed < 'Speed Limitation'.	
Syntax	**$ActSetSpeed,[Command S/N],[Password],[Enable],[Speed Limitation],[Speed Limitation Duration],[Report Time Interval]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000". Range from "0000" to "9999".
	Enable	YES : Enable the tracker to generate '$LSIN' and '$LSUT' event reports NO : Disable the tracker to generate '$LSIN' and '$LSUT' event reports. Default Value = "NO"
	Speed Limitation	A speed threshold to trigger '$LSIN' event reports by every 'Report Time Interval' if the GPS speed > 'Speed Limitation' for more than 'Speed Limitation Duration'. Then it sends a '$LSUT' when the GPS speed < 'Speed Limitation'.The unit is 'mph'
	Speed Limitation Duration	A time threshold to trigger '$LSIN' event reports if the GPS speed > 'Speed Limitation' for more than 'Speed Limitation Duration'. It keeps sending '$LSIN' event report by every 'Report Time Interval'. The unit is 'second'. Range from "10" to 65535".
	Report Time Interval	A time interval to keep sending '$LSIN' event reports if the tracker keep speeding. The unit is 'second'.

Return Syntax	**$ActSetSpeed,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set the tracker to generate Speeding reports when the tracker move faster than 70 mph for more than 30 seconds and keep sending it every 60 seconds. →$ActSetSpeed,1234,0000,YES,70,30,60& ←$ActSetSpeed,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.23 ActGetSpeed（Get the Speeding settings）

Command	**ActGetSpeed**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the speeding settings form the tracker	
Syntax	**$ ActGetSpeed,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
Return Syntax	**$ActGetSpeed,[Command S/N],[Enable],[Speed Limitation],[Speed Limitation Duration],[Report Time Interval],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	Enable	YES : Enable the tracker to generate ‘$LSIN’ and ‘$LSUT’ event reports NO : Disable the tracker to generate ‘$LSIN’ and ‘$LSUT’ event reports. Default Value = “NO”
	Speed Limitation	A speed threshold to trigger ‘$LSIN’ event reports by every ‘Report Time Interval’ if the GPS speed > ‘Speed Limitation’ for

		more than ‘Speed Limitation Duration’. Then it sends a ‘$LSUT’ when the GPS speed < ‘Speed Limitation’.The unit is ‘mph’
	Speed Limitation Duration	A time threshold to trigger ‘$LSIN’ event reports if the GPS speed > ‘Speed Limitation’ for more than ‘Speed Limitation Duration’. It keeps sending ‘$LSIN’ event report by every ‘Report Time Interval’. The unit is ‘second’. Range from “10” to 65535”.
	Report Time Interval	A time interval to keep sending ‘$LSIN’ event reports if the tracker keep speeding. The unit is ‘second’.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetSpeed,1234,0000& ←$ActGetSpeed,1234,YES,70,30,60,080603153040&	

2.24 ActSetParkFence（Set the ParkFence Settings）

Command	**ActSetParkFence**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Enable or Disable the ParkFence function of the tracker. Send an ActSaveVarCFG command right after the ActSetParkFence command to save new setting to the Flash Memory and to apply it. When the ‘ParkFence Enable’ =YES, the tracker build up a GeoFence automatically where the ACC is OFF. The tracker will be guarded by the ‘ParkFence Radius’. If there is an unauthorized moving or towing, a ‘$LGUT’ event Report (GeoFence Group Number =0) will be triggered. Note: When the ACC is ON again, the ParkFence will be deactivated.	
Syntax	**$ActSetParkFence,[Command S/N],[Password],[ParkFence Enable],[Radius],[Alarm Time Interval]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	ParkFence Enable	YES : Enable ParkFence feature when the driver parks the vehicle. NO : Disable ParkFence feature.

	Radius	A Radius distance of the ParkFence to trigger a '$LGUT' event report. The unit is 'mile'. Min. 0.3 miles
	Alarm Time Interval	A Time Interval to continuously trigger a '$LGUT' event report when the vehicle keeps staying outside of the ParkFence. The unit is 'second'. Range from [0(Once)|20-65535]
Return Syntax	**$ActSetParkFence,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Set Distance Track Interval as '10' to track the vehicle every 10 miles. →$ActSetParkFence,1234,0000,YES,0.1,30& ←$ActSetParkFence,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153041&	

2.25 ActGetParkFence（Get the ParkFence Settings）

Command	**ActGetParkFence**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the ParkFence setting form the tracker	
Syntax	**$ ActGetParkFence,[Command S/N],[Password]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value : "0000". Range from "0000" to "9999".
Return Syntax	**$ActGetParkFence,[Command S/N],[ParkFence Enable],[Radius],[Alarm Time Interval],[Moved Distance],[Park Longitude],[Park Latitude],[ParkFence Status],[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".
	ParkFence Enable	YES : Enable ParkFence feature when the driver parks the vehicle.

		NO : Disable ParkFence feature.
	Radius	A Radius distance of the ParkFence to trigger a '$LGUT' event report. The unit is 'mile'. Min. 0.3 miles
	Alarm Time Interval	A Time Interval to continuous trigger a '$LGUT' event report when the vehicle keeps staying outside of the ParkFence. The unit is 'second'. Range from [0(Once)|20-65535]
	Moved Distance	Reserved. For debugging only.
	Park Longitude	Reserved. For debugging only.
	Park Latitude	Reserved. For debugging only.
	ParkFence Status	Reserved. For debugging only.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetParkFence,1234,0000& ←$ActGetParkFence,1234,YES,0.100000,0,30.248614,0.000000,0.000000,1,080603153040&	

2.26 ActSetGeoFence（Set the GeoFence Settings）

Command	**ActSetGeoFence**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Enable or Disable the GeoFence function of the tracker. Send an ActSaveVarCFG command right after the ActSetGeoFence command to save new setting to the Flash Memory and to apply it. When the 'GeoFence Enable' =YES and the tracker enters the boundary of the GeoFence, it triggers a '$LGIN' event report. Whe it leaves the boundary, it triggers a '$LGUT' event Report. The current version firmware supports 199 sets of GeoFence.	
Syntax	**$ActSetGeoFence,[Command S/N],[Password],[GeoFence Group Number],[Enable],[E|W],[Longitude degree],[Longitude minute],[Longitude second],[N|S],[Latitude degree],[Latitude minute],[Latitude second],[GeoFence Radius],[Inside Time Interval],[Outside Time Interval]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000".

		Range from “0000” to “9999”.
	GeoFence Group Number	Range from “1” to “199”.
	Enable	YES : Enable the GeoFence Group. NO : Disable the GeoFence Group.
	E|W	E: East, W: West
	Longitude degree	Central Longitude degree of the GeoFence
	Longitude minute	Central Longitude minute of the GeoFence
	Longitude second	Central Longitude second of the GeoFence
	N|S	N: North, S: South
	Latitude degree	Central Latitude degree of the GeoFence
	Latitude minute	Central Latitude minute of the GeoFence
	Latitude second	Central Latitude second of the GeoFence
	GeoFence Radius	A Radius distance of the GeoFence to trigger a ‘$LGUT’ or a ‘$LGUT’ event report. The unit is ‘mile’. Min. 0.3 miles
	Inside Time Interval	A Time Interval to continuously trigger a ‘$LGIN’ event report when the vehicle keeps staying inside of the GeoFence. The unit is ‘second’. Range from [0(Once)|20-65535]
	Outside Time Interval	A Time Interval to continuously trigger a ‘$LGUT’ event report when the vehicle keeps staying outside of the GeoFence. The unit is ‘second’. Range from [0(Once)|20-65535]
Return Syntax	**$ActSetGeoFence,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss

Examples	Set the GeoFence at E 121°31′24″, N 25°2′57″, radius =0.5 mile, →$ActSetGeoFence,1234,0000,1,YES,E,121,31,24,N,25,2,57,0.5,0,0& ←$ActSetGeoFence,1234,OK,080603153040& →$ActSaveVarCFG,1234,0000& ←$ActSaveVarCFG,1234,OK,080603153040&

2.27 ActGetGeoFence（Get the GeoFence Settings）

Command	**ActGetGeoFence**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Get the GeoFence setting form the tracker	
Syntax	**$ActGetGeoFence,[Command S/N],[Password],[GeoFence Group Number]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value : ”0000”. Range from “0000” to “9999”.
	GeoFence Group Number	Range from “1” to “199”.
Return Syntax	**$ActGetGeoFence,[Command S/N],[GeoFence Group Number],[Enable],[E/W],[Longitude degree],[Longitude minute],[Longitude second],[N/S],[Latitude degree],[Latitude minute],[Latitude second],[GeoFence Radius],[Inside Time Interval],[Outside Time Interval],[Moved Distance],[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	GeoFence Group Number	Range from “1” to “24”.
	Enable	YES : Enable the GeoFence Group. NO : Disable the GeoFence Group.
	E|W	E: East, W: West
	Longitude degree	Central Longitude degree of the GeoFence
	Longitude minute	Central Longitude minute of the GeoFence
	Longitude second	Central Longitude second of the GeoFence
	N|S	N: North, S: South
	Latitude degree	Central Latitude degree of the GeoFence

	Latitude minute	Central Latitude minute of the GeoFence
	Latitude second	Central Latitude second of the GeoFence
	GeoFence Radius	A Radius distance of the GeoFence to trigger a '$LGUT' or a '$LGUT' event report. The unit is 'mile'. Min. 0.3 miles
	Inside Time Interval	A Time Interval to continuously trigger a '$LGIN' event report when the vehicle keeps staying inside of the GeoFence. Range from [0(Once)|20-65535]
	Outside Time Interval	A Time Interval to continuously trigger a '$LGUT' event report when the vehicle keeps staying outside of the GeoFence. Range from [0(Once)|20-65535]
	Moved Distance	Reserved. For debugging only.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	→$ActGetGeoFence,1234,0000,1& ←$ActGetGeoFence,1234,1,YES,E,121,31,24,N,25,2,57,0.5,0,0,0.000,080603153040&	

2.28 ActComOut（Send a string to COM port）

Command	**ActComOut**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Use this command to send a string to COM1 or COM 2 port of Easytrac. If an external device connected to Easytrac or Speedtrac by RS-232 connection, it can receive the string. The server can send a command string to the external device like this way.	
Syntax	**$ActComOut,[Command S/N],[Password],[COM Port No.],[String]&**	
Parameters	Command S/N	Command must include the "Command Sequence Number".
	Password	Default Value: "0000". Range from "0000" to "9999".
	COM Port No.	"1" or "2"
	String	ASCII string, maximum
Return Syntax	**$ActComOut,[Command S/N],OK,[Com Port No.],[RTC Time]&**	
Return Values	Command S/N	Return string must include the "Command S/N".

	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	Com Port No.	“1” or “2”
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Send ‘Hello World’ to COM port 1 →$ActComOut,1234,0000,1,Hello World& ←$ActComOut,1234,1,OK,080603153040&	

2.29 ActSendSMS（Send a SMS message from the tracker）

Command	**ActSendSMS**	
Models	☑Easytrac ☑Speedtrac ☑OBDtrac	
Description	Command to send a SMS text message from the tracker.	
Syntax	**$ActSendSMS,[Command S/N],[Password],[Phone number],[TON],[Message]&**	
Parameters	Command S/N	Command must include the “Command Sequence Number”.
	Password	Default Value: ”0000”. Range from “0000” to “9999”.
	Phone number	Destination phone number.
	TON	Type of number, 0 or 1.
	Message	Text message body.
Return Syntax	**$ActSendSMS,[Command S/N],OK,[RTC Time]&**	
Return Values	Command S/N	Return string must include the “Command S/N”.
	OK | Cmd | Pwd | Param	Command is done. | Invalid Command. | Invalid Password. | Invalid Parameters.
	RTC Time	Device RTCTime. The format is YYMMDDHHmmss
Examples	Send ‘Hello World’ SMS message →$ActSendSMS,1234,0000,123456789,1,Hello World& ←$ActSendSMS,1234,1,OK,080603153040&	

3、Event Reports

3.1 RGTK (evtGPRSTrack, Real Time GPRS Tracking Event Report）

Trigger event	evtGPRSTrack
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	This is the time based real time GPS position report over GPRS according to 'Report Interval' parameter in either $SetGPRSConfig or $ActGPRSTrack.
Data Structure	**$RGTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When it is in START mode or ACC is switched ON in ACCON mode, the tracker sends the event report as ←$RGTK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

3.2 LGTK (logevtGPRSTrack, Resent GPRS Tracking Event Report）

Trigger event	logevtGPRSTrack
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	When the real time event report '$RGTK' can't be sent out, the tracker stores those logs in the flash memory and resends them when the GPRS connection is available.
Data Structure	**$LGTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker resends the event report as ←$LGTK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.3 RMTK (evtGSMTrack, Real Time SMS Tracking Event Report）

Trigger event	evtGSMTrack
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	This is the time based real time GPS position report by SMS according to 'Report Interval' parameter in either $SetGSMConfig or $ActGSMTrack.
Data Structure	**$RMTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When it is in START mode or ACC is switched ON in ACCON mode, the tracker sends the event report as ←$RMTK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

3.4 LMTK (logevtGSMTrack, Resent SMS Tracking Event Report）

Trigger event	logevtSMSTrack
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	When the real time event report '$RGTK' can't be sent out, the tracker stores those logs in the flash memory and resends them when the GPRS connection is available.
Data Structure	**$LMTK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker resends the event report as ←$LMTK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.5 LGPS (logGetPos, Current GPS position Event Report）

Trigger Command	ActGetPos
Models	☑Easytrac ☑ Speedtrac ☑OBDtrac
Description	When the tracker receives an ‘ActGetPos’ command, it sends this report to the server accordingly.
Data Structure	**$LGPS,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker receives an ‘ActGetPos’ command and reply the following event report. ←$LGPS,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

3.6 LACN (logAccOn, ACC ON Event Report）

Trigger event	evtAccOn
Models	☑Easytrac ☑Speedtrac ☐OBDtrac
Description	When the ACC (Ignition) is switched ON, the tracker sends this report to the server.
Data Structure	**$LACN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	ACC is switched ON, the tracker sends the event report as ←$LACN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,2,0,070623182844&

3.7 LACF (logAccOff, ACC OFF Event Report）

Trigger event	evtAccOff
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the ACC (Ignition switch) is switched OFF, the tracker sends this report to the server.
Data Structure	**$LACF,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	ACC is switched OFF, the tracker sends the event report as ←$LACF,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.8 LMPN (logMainOn, Main Power ON Event Report）

Trigger event	evtMainOn
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Main power (vehicle power) is back, the tracker sends this report to the server. The DI reading in every report is an even number which means the tracker uses vehicle power. Ex. 0,2,6,14… The AI1 and AI2 readings in logMainOn report indicate the voltages of vehicle power and backup battery.
Data Structure	**$LMPN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When the tracker’s power souce switchs from Backup battery to Vehicle power, the tracker sends the event report as ←$LMPN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,12.3,3.8,2,0,070623182844&

3.9 LMPF (logMainOff, Main Power OFF Event Report）

Trigger event	evtMainOff
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Main power (vehicle power) is cut off, the tracker sends this report to the server. The DI reading in every report becomes an odd number which means the tracker uses backup battery power. Ex. 1,3,7,15… The AI1 and AI2 readings in 'logMainOff' report indicate the voltages of vehicle power and backup battery.
Data Structure	**$LMPF,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When the tracker's power souce switchs from Vehicle power to Backup battery, the tracker sends the event report as ←$LMPF,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,0,4.8,0,0,070623182844&

3.10 LBAT (logBatVolt, Battery Voltages Event Report）

Trigger Event	evtBatVolt
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	The tracker sends this report according to 'Battery Report Interval' of $SetConnection command. The AI1 and AI2 readings in the report indicate the voltages of vehicle power and backup battery.
Data Structure	**$LBAT,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker receives a key pressed signal from the remote controller, it sends the following event report. ←$LBAT,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,12.3,3.8,2,0,070623182844&

3.11 LVML (logOverMile, Over Mileage Limitation Event Report）

Trigger event	evtOverMile
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Total Mileage reaches Mileage Limitation, the tracker sends this report to the server.
Data Structure	**$LMVL,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When the Total Mileage reaches Mileage Limitation, the tracker sends the event report as ←$LMVL,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 0.00, 500.75,8.8,9.4,2,6,070623182844&

3.12 LDIS (logDistance, Distance Tracking Event Report）

Trigger event	evtDistance
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Distance Track Interval >0, the tracker sends this report to the server by every ‘Distance Track Interval’.
Data Structure	**$LDIS,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends the event report by every ‘Distance Track Interval’. ←$LDIS,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 0.00, 500.75,8.8,9.4,2,6,070623182844&

3.13 LANG (logDevAng, Deviation Angle Tracking Event Report）

Trigger event	evtDevAngle
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Deviation Angle > ‘Dev Angle’ of $ActSetDevAngle command, the tracker sends this report.
Data Structure	**$LANG,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends the event report by every ‘Distance Track Interval’. ←$LANG,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 0.00, 500.75,8.8,9.4,2,6,070623182844&

3.14 LIDL (logIdle, Idling Event Report）

Trigger event	evtIdle
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the GPS speed < 5mph for more than ‘Idle Duration’. The tracker sends this report to the server by every ‘Report Time Interval’.
Data Structure	**$LIDL,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends the event report by every ‘Distance Track Interval’. ←$LIDL,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00, 500.75,8.8,9.4,2,6,070623182844&

3.15 LMOV (logMove, Moving Event Report）

Trigger event	evtMove
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	The tracker sends this report when it moves again from its idling. It sends the report when the GPS speed >5mph and moving > 0.02 miles (30 meters).
Data Structure	**$LMOV,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends the event report by every ‘Distance Track Interval’. ←$LMOV,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 10.00, 500.75,8.8,9.4,2,6,070623182844&

3.16 LSIN (logSpeedIn, Speeding Event Report）

Trigger event	evtSpeedIn
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the GPS speed > ‘Speed Limitation’ for more than ‘Speed Limitation Duration’. The tracker sends this report to the server by every ‘Report Time Interval’.
Data Structure	**$LSIN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends the event report by every ‘Report Time Interval’ of $ActSetSpeed command ←$LSIN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 72.00, 500.75,8.8,9.4,2,6,070623182844&

3.17 LSUT (logSpeedOut, SpeedOut Event Report）

Trigger event	evtSpeedOut
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	The tracker sends this report when its speed is lower than ‘Speed Limitation’ again.
Data Structure	**$LSUT,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	The tracker sends this report when its speed is lower than ‘Speed Limitation’ again. ←$LSUT,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 66.00, 500.75,8.8,9.4,2,6,070623182844&

3.18 LGIN (logGeoIn, Enter GeoFence Event Report）

Trigger event	evtGeoIn
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the tracker enters the GeoFence boundary, it sends this report with GeoFence Group Number and the distance between the center of the GeoFence and the tracker.
Data Structure	**$LGIN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[GeoFence Group No.],[Moved Distance],[Empty],[Empty],[RTCTime]&**
Examples	The tracker enters a GeoFence, it sends the following event report. ←$LGIN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00, 0.00,500.75,2,0.4621,,,070623182844&

3.19 LGUT (logGeoOut, Exit GeoFence Event Report）

Trigger event	evtGeoOut
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the tracker exits the GeoFence boundary, it sends this report with GeoFence Group Number and the distance between the center of the GeoFence and the tracker.
Data Structure	**$LGUT,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[GeoFence Group No.],[Moved Distance],[Empty],[Empty],[RTCTime]&**
Examples	The tracker exits a GeoFence, it sends the following event report. ←$LGUT,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,2,0.5621,,,070623182844&

3.20 LDxN (logDIxOn, Digital Input Change to ON Event Report）

Trigger Event	evtDIxOn
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Digital Input 2 ~ 6 are switched ON, the tracker sends this report to the server. DI1 is reserved of ACC (Ignition) detection.
Data Structure	**$LDxN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	DI2 is switched ON, the tracker send the event report as ←$LD2N,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,0.1605,8.8,9.4,4,0,070623182844&

3.21 LDxF (logDIxOff, Digital Input OFF Event Report）

Trigger Event	evtDIxOff
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the Digital Input 2 ~ 6 are switched OFF, the tracker sends this report to the server. DI1 is reserved of ACC (Ignition) detection.
Data Structure	**$LDxF,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	DI2 is switched OFF, the tracker send the event report as ←$LD2F,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,0.1605,8.8,9.4,0,0,070623182844&

3.22 LKIN (logKeyIn, User Keyin by Keypad Event Report）

Trigger Event	evtKeyIn
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the user enters some number by the Keypad that connectes to the tracker and press 'Enter', the tracker sends this report to the server. The available numbers are 0~9, A (F1 key of keypad), B(F2 key), C (F3 key) and D (F4 key). The maximum data entry is 27 digits.
Data Structure	**$LKIN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[Keyin Data],[Empty],[Empty],[Empty],[RTCTime]&**
Examples	The user enter [F1]1234567890 on the Keypad then press 'Enter', the tracker sends the event report as ←$LKIN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,0,A1234567890,,,,070623182844&

3.23 LUIN (logUserLogin, User Login by RFID card or iButton Event Report）

Trigger Event	evtUserLogin
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the user swipes a RFID card to a RFID reader or when the user uses his iButton to touch the iButton reader that connects to the tracker, the tracker sends this report to the server.
Data Structure	**$LUIN,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[RFID No.], [Empty],[Empty],[Empty],[RTCTime]&**
Examples	The user login by a RFID card with No. 0A0080A980, the tracker send the event report as ←$LUIN,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,0,0A0080A980,,,,070623182844&

3.24 LUOT (User Logout by removing iButton Event Report）

Trigger Event	evtUserLogout
Models	☑Easytrac ☑Speedtrac ☑OBDtrac
Description	When the user removes his iButton from the iButton reader that connects to the tracker, the tracker sends this report to the server.
Data Structure	**$LUOT,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[RFID No.], [Empty],[Empty],[Empty],[RTCTime]&**
Examples	The user login by a RFID card with No. 0A0080A980, the tracker send the event report as ←$LUOT,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,0,0A0080A980,,,,070623182844&

3.25 LGSK (Start journey event by vibration detection in GPRS Mode Event Report）

Trigger Event	evtStartMoving
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a OBDtrac or a Speedtrac PRO is waked up by a schedule timer or a vibration tetection, the tracker sends this report to the server by GPRS.
Data Structure	**$LGSK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac starts moving, the tracker send the event report as ←$LGSK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.26 LGEK (End journey event by vibration detection in GPRS Mode Event Report）

Trigger Event	evtStopMoving
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac stops without vibration detected for more than certain time period (ex. 60 sec), it sends this report to the server by GPRS then shutting down.
Data Structure	**$LGEK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac starts moving, the tracker send the event report as ←$LGEK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.27 LMSK (Start journey event by vibration detection in SMS Mode Event Report）

Trigger Event	evtStartMoving
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac is waked up by a schedule timer or a vibration tetection, the tracker sends this report to the server by SMS.
Data Structure	**$LMSK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac starts moving, the tracker send the event report as ←$LMSK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.28 LMEK (End journey event by vibration detection in SMS Mode Event Report）

Trigger Event	evtStopMoving
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac stops without vibration detected for more than certain time period (ex. 60 sec), it sends this report to the server by SMS then shutting down.
Data Structure	**$LMEK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac starts moving, the tracker send the event report as ←$LMEK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.29 RGRK (Real Time Schedule Timer Event Report in GPRS Mode）

Trigger Event	evtScheduleTimer
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac is waked up by the schedule timer, the tracker sends this report to the server by GPRS.
Data Structure	**$RGRK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac is waked up by the schedule timer, the tracker send the event report as ←$RGRK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.30 LGRK (Logged Schedule Timer Event Report in GPRS Mode）

Trigger Event	evtScheduleTimer
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac is waked up by the schedule timer, and the tracker can not send the report to the server by GPRS immediately. The tracker will send it as LGRK next time it get GPRS connected.
Data Structure	**$LGRK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac is waked up by the schedule timer, the tracker send the event report as ←$LGRK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.31 RMRK (Schedule Timer Event Report in SMS Mode）

Trigger Event	evtScheduleTimer
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac is waked up by the schedule timer, the tracker sends this report to the server by SMS.
Data Structure	**$RMRK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedrac or OBDtrac is waked up by the schedule timer, the tracker send the event report as ←$RMRK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.32 LMRK (Logged Schedule Timer Event Report in SMS Mode）

Trigger Event	evtScheduleTimer
Models	☐Easytrac ☑Speedtrac ☑OBDtrac
Description	When a Speedtrac or OBDtrac is waked up by the schedule timer, and the tracker can not send the report to the server by SMS immediately. The tracker will send it as LMRK next time it gets GSM connected.
Data Structure	**$LMRK,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[AI1],[AI2],[DI],[DO],[RTCTime]&**
Examples	When a Speedtrac or OBDtrac is waked up by the schedule timer, the tracker send the event report as ←$LMRK,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,8.8,9.4,0,0,070623182844&

3.33 LIBT (ORION iButton Reader Event Reports）

Trigger Event	evtiButton
Models	☑Easytrac ☑Speedtrac ☑OBDtrac

Description	When an ORION iButton reader is connected to Easytrac or Speedtrac, all iButton reader events will be send by the tracker as $LBAT event reports. In $LIBT reports, it uses AI-1 column for the iButton reader 'Event' and AI-2 column for the 'iButton ID'.
Data Structure	**$LIBT,[UnitID],[UTCTime],[A|V],[Longitude],[E|W],[Latitude],[N|S],[Satellite Numbers],[GSM Signal],[Angle],[Speed],[Mileage],[iButton Event],[iButton ID],[DI],[DO],[RTCTime]&**
Examples	When the B/P switch of the iButton Reader is pushed or an iButton ID is detected, the tracker sends the event report as ←$LIBT,3000000004,230607102747,A,12131.3775,E,2502.8874,N,10,25,0.00,0.00,500.75,DN,123456789,0,0,070623182844&

iButton Events	Examples	Description
DN	DN,C50008014DEA8E10	Driver Log-in : ID: C50008014DEA8E10
DF	DF,C50008014DEA8E10	Driver Log-out : ID: C50008014DEA8E10
1N	1N,C50008014DEA8E11	1st Passenger Log-in: ID: C50008014DEA8E11
1F	1F,C50008014DEA8E11	1st Passenger Log-out: ID: C50008014DEA8E11
2N	2N,C50008014DEA8E12	2nd Passenger Log-in: ID: C50008014DEA8E12
2F	2F,C50008014DEA8E12	2nd Passenger Log-out: ID: C50008014DEA8E12
3N	3N,C50008014DEA8E13	3rd Passenger Log-in: ID: C50008014DEA8E13
3F	3F,C50008014DEA8E13	3rd Passenger Log-out: ID: C50008014DEA8E13
4N	4N,C50008014DEA8E14	4th Passenger Log-in: ID: C50008014DEA8E14
4F	4F,C50008014DEA8E14	4th Passenger Log-out: ID: C50008014DEA8E14
5N	5N,C50008014DEA8E15	5th Passenger Log-in: ID: C50008014DEA8E15
5F	5F,C50008014DEA8E15	5th Passenger Log-out: ID: C50008014DEA8E15
BT	BT,	Button pushed: Business Trip
PT	PT,	Button pushed: Private Trip
OT	OT,	Button pushed: Other Trip
PB	PB,	Button pushed for more than 3 sec as a Panic Button: SOS

Tel: +886-2-2521-2610
http:// www.oriontech.com.tw
Email: support@oriontech.com.tw

Appendix-1 : Event Reports Syntax

S/N	Item	Length (bytes)	Description
1	Start Sign	1	$
2	Event Type	4	4 characters
3	,	1	
4	UnitID	10	10 digits
5	,	1	
6	UTCTime	12	ddMMyyHHmmss
7	,	1	
8	GPS Valid	1	V/A
9	,	1	
10	Longitude	10	xxxxx.xxxx
11	,	1	
12	E/W	1	E/W
13	,	1	
14	Latitude	9	xxxx.xxxx
15	,	1	
16	N/S	1	N/S
17	,	1	
18	Satellite No.	2	0~12
19	,	1	
20	GSM Signal	2	0~31
21	,	1	
22	Angle	3	0~360
23	,	1	
24	Speed	5	999.9
25	,	1	
26	Mileage	8	Max: 999999.9
27	,	1	
28	AI-1	4	Temperature or RPM
29	,	1	
30	AI-2	4	Temperature or Coolant Temperature
31	,	1	
32	DI	3	
33	,	1	
34	DO	2	
35	,	1	
36	RTCTime	12	yyMMddHHmmss
37	,	1	
38	PLMN	6	Network Info
39	,	1	
40	Cell & LAC	8	Cell Info
41	End Sign	1	&
42	package chechsum	2	Optional, when use ACK
43	<CR><LF>	2	0xD 0xA
	Total:	132	bytes

Appendix-2 : All Event Reports (Cell ID Info Excluded)

Note: All event reports are all the same with 20 sectors. For data structure description, please refer to Section 2.1 ActGPRSTrack (P.21)

Description	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	Start	EventType	UnitID	UTCTime	Valid	Longitude	E/W	Latitude	N/S	Satellite No.	GSM Signal	Angle	Speed	Mileage	AI-1	AI-2	DI	DO	RTCTime YYMMDDHHmmss	End
GPRSTrack	$	RGTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logGPRSTrack	$	LGTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
GSMTrack	$	RMTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logGSMTrack	$	LMTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logGetPos	$	LGPS	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logAccOn	$	LACN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logAccOff	$	LACF	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logMainOn	$	LMPN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logMainOff	$	LMPF	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logBatteryVoltage	$	LBAT	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logOverMile	$	LVML	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logDistance	$	LDIS	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logDevAng	$	LANG	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logIdle	$	LIDL	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logMove	$	LMOV	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logSpeedIn	$	LSIN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logSpeedOut	$	LSUT	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logGeoIn	$	LGIN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	1(GeoFence Group No.)	0.4521(Moved Distance)	0	0	70622181946	&
logGeoOut	$	LGUT	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	1(GeoFence Group No.)	0.5613 (Moved Distance)	0	0	70622181946	&

logDIChange	$	LD2N LD2F LD3N LD3F LD4N LD4F LD5N LD5F LD6N LD6F	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	4	0	70622181946	&
logKeyIn	$	LKIN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	1234567890 (Keyin Data)	Empty	0	0	70622181946	&
logUserLogin	$	LUIN	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0A0080A980 (RFID No.)	Empty	0	0	70622181946	&
logUserLogout	$	LUOT	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0A0080A980 (RFID No.)	Empty	0	0	70622181946	&
logStartJourney	$	LGSK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logEndJourney	$	LGEK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logStartJourney	$	LMSK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logEndJourney	$	LMEK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logScheduleTimer	$	RGRK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logScheduleTimer	$	LGRK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&

logScheduleTimer	$	RMRK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logScheduleTimer	$	LMRK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	&
logiButton	$	LIBT	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	DN (Event)	0A0080A980 (iButton ID)	0	0	70622181946	&

Appendix-3 : Event Report with Cell ID Examples

Note: All event reports are all the same with 22 sectors. For data structure description, please refer to Section 2.1 ActGPRSTrack (P.21)

Description	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	Start	EventType	UnitID	UTCTime	Valid	Longitude	E/W	Latitude	N/S	Satellite No.	GSM Signal	Angle	Speed	Mileage	AI-1	AI-2	DI	DO	RTCTime YYMMDDHH mmss	PLMN (Network ID)	Cell Info	End
GPRS Track	$	RGTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	46692	0B5565CA	&
logGPRSTrack	$	LGTK	2000000956	220607101948	A	12131.4	E	2502.917	N	6	17	0	0	500.75	0	0	0	0	70622181946	46692	0B5565CA	&