ASSISTED GNSS

INSTRUCTIONS

WWW.UNICORECOMM.COM

AGNSS Instructions

Revision History

Version	Revision History	Date
R1	Initial release	2019-10
R1.1	Add AGNSS instructions of ROM version	2020-04
R1.2	Add dld restrictions	2020-12
R1.3	Section 3.1: update the sample code Chapter 5: elaborate the information of HTTP header	2021-04-20

Disclaimer

This manual provides information about the products of Unicore Communications, Inc. This document does not transfer the patent, trademark, copyright of the company or any third party, or any right or permission under it by implication, estoppel or otherwise.

Except as stated in the sales terms and conditions of the products, the company shall not assume any other responsibilities. Furthermore, Unicore Communications, Inc. makes no warranty, express or implied, for the sale and/or use of its products, including the suitability of a particular purpose and marketability of products, or liability for infringement of any patent, copyright or other intellectual property rights. If the connection or operation is not in accordance with the manual, the company is not liable. Unicore Communications, Inc. may make changes to product specifications and product descriptions at any time without prior notice.

The company's products may contain certain design defects or errors, which will be included in the corrigendum once found, and may therefore result in differences between the products' actual specifications and the published ones. Updated corrigendum is available upon request.

Before placing an order, please contact our company or local distributors for the latest specifications.

* Unicore Communications, UNICORECOMM, UFirebird and its logo have been applied for trademark registration by Unicore Communications, Inc.

Other names and brands are the property of their respective owners.

Foreword

This manual introduces the instructions on the use of Unicore AGNSS functions.

Audience

This manual is created for the technical personnel, who possess the expertise of GNSS receivers.

Contents

1 Overview

Unicore AGNSS provides real-time ephemeris of global GPS, BeiDou, Galileo, GLONASS and QZSS and offers the ephemeris filtering function based on approximate location. The AGNSS service transmits ephemeris data via HTTP protocol.

2 Prerequisites

Since AGNSS function requires to access the server through the Internet, the target device needs to have the ability to access the Internet to obtain ephemeris data.

The account provided by Unicore is required to access AGNSS server to obtain ephemeris data.

Unicore AGNSS account example:

Host: unicore-api1.rx-networks.cn

cId: TA453Jxdkk7z

mId: Unicore

pw:cUvnnJvwDXNBvM51YUbIdA==

2.1 Server Address

Unicore offers a second-level domain (URL) such as unicore-api1.rx-networks.cn to each customer using AGNSS function. When accessing the server, it's required to resolve the IP address through DNS and then connect the server through this IP address.

It should be noted that, in order to ensure the global service quality, the dynamic load balancing technology is used for the AGNSS server. The IP address that the domain name points to is not fixed, so the server cannot be directly accessed through the IP address. It's required to get the IP address through DNS resolution, and then use the IP address to access the server.

2.2 Authentication

The AGNSS service provides ephemeris data over the HTTP protocol and the HTTP request must contain the account information. The format of the authentication fields in the HTTP request is as follows:

Authorization:RXN-BASIC cId=<cId>,mId=<mId>,dId=<dId>,pw=<pw>

The format of each parameter and the customer's corresponding account are as follows:

Table 2- 1 Parameter Description

Parameter name	Account example	Description	Remarks
cId	ID	User name, provided by Unicore	
mId	mId	Customer ID, no longer than 25 bytes, provided by Unicore	
dId	-	Device ID. Each device is required to have a unique device ID, which can be MAC address, IMEI, etc. The length shall be no more than 50 bytes, customized by the customer	To ensure the quality of service and prevent the frequent request of ephemeris data after the device goes wrong, the device ID access times are limited. Each device ID can request ephemeris data 12 times per day. If the number of requests is greater than 12 times, the server may deny service.
pw	Base64 Password	Password no longer than 50 bytes, provided by Unicore	

Note:

If ***dId*** is available, the request must be sent in strict compliance with the above-mentioned field format, which is:

Authorization:RXN-BASIC cId=<cId>,mId=<mId>,dId=<dId>,pw=<pw>

If ***dId*** is not available, you can simply remove the ***dId*** field, which is:

Authorization:RXN-BASIC cId=<cId>,mId=<mId>, pw=<pw>

The ***dId*** field can not be null, or the request will not be received properly.

3 Request

3.1 Request Real-time Ephemeris

The format of the request complies with the HTTP protocol. A complete request contains two parts: the HTTP header and the request Content. According to the HTTP protocol, there must be a blank line between the two parts. The Content conforms to JSON data format. The following is an example of requesting the ephemeris data of GPS and Beidou from the server to explain the request format. A complete request is as follows:

```
POST /rxn-api/locationApi/rtcm HTTP/1.1
Host:XXXXXXX
Authorization:RXN-BASIC cId=XXXXXX,mId=XXXXX,dId=XXXXXXXXXX,pw=XXXXXX
Content-type:application/json
Accept:application/octet-stream
Content-length: XX

[{"rtAssistance":{"format":"rtcm","msgs":["GPS:1NAF","BDS:2NAF"]}}]
```

The Host field in the HTTP header is the domain name of the server (URL).

The length of the Content must match the Content-length field in the HTTP header, as described below:

Table 3- 1 Field Description

Field	Description
rtAssistance	The real-time ephemeris data is requested. This field is fixed and cannot be changed.
"format":"rtcm"	The requested ephemeris data is in RTCM format. This field is fixed and cannot be changed.
"msgs":["GPS:1NAF","BDS:2NAF"]	The ephemeris data of GPS and BDS is requested. This field can be freely combined by the user as needed. The identification of each constellation supported by the server is as follows: GPS: 1NAF represents GPS ephemeris BDS: 2NAF represents Beidou ephemeris GLO: 2NAF represents GLONASS ephemeris GAL: 2NAF represents Glileo ephemeris QZS: 2NAF represents QZSS ephemeris Example: to request the ephemeris of three constellations GPS+GAL+BDS: "msgs":["GPS:1NAF","GAL:2NAF","BDS:2NAF"]

3.2 Approximate Location Filtering Ephemeris Request

Approximate location filtering is that the server is allowed to filter out the invisible satellites in this location and only returns the ephemeris of the visible satellites, thus reducing the ephemeris data size. The ephemeris data can be reduced by at least half by using approximate location filtering.

The following is an example of the request for GPS+BDS:

```
POST /rxn-api/locationApi/rtcm HTTP/1.1
Host:XXXXXXX
Authorization:RXN-BASIC cId=XXXXXX,mId=XXXXX,dId=XXXXXXXXXX,pw=XXXXXX
Content-type:application/json
Accept:application/octet-stream
Content-length: XX

[{"rtAssistance":{"format":"rtcm","msgs":["GPS:1NAF","BDS:2NAF"],
"mask":15,"lat":40.232925,"lon":116.467036}}]
```

This request indicates that the server is required to return the visible satellite ephemeris with an elevation of more than 15° at the latitude 40.232925 and longitude 116.467036.

The three parameters are described as follows:

Table 3- 2 Parameter Description

Parameter	Value Range	Description
mask	0 <= mask <90	Visible satellite elevation
lat	-90 <= lat <=90	Latitude, positive for north latitude and negative for south latitude
lon	-180 <= lon <= 180	Longitude, positive for east longitude and negative for west longitude

4 Echo

The server will return data that conforms to the format of HTTP protocol after receiving a legal request. The content returned contains HTTP header and ephemeris data, with a blank line between them as specified by the HTTP protocol. The example is as follows:

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
Date: Tue, 06 Nov 2018 08:49:10 GMT
```

Content-Length: XXXX
Connection: keep-alive

data…

"200" in the first line of HTTP protocol header is the error code of HTTP, defined as follows:

Table 4- 1 Error Code Definition

HTTP Error Code	Meaning	Explanation
200	Normal	Return this value if no exception occurs
204	No data	Return this value when the server receives a valid request but has no valid ephemeris data
400	Illegal request	Return this value when the request is not valid
401	Authentication failed	Return this value when the server detects an exception in the user name or password
500	Server exception	Return this value when an exception occurs on the server
501	Temporarily out of service	Return this value when the HTTP service is temporarily unavailable

The Content-Length line indicates the length of the ephemeris data and the client program may receive ephemeris data based on this length.

5 Injection

After receiving the data returned by the server, you only need to remove the HTTP header (there is a blank line between the HTTP header and the data, which can be used as a basis to remove the HTTP header) and send the remaining auxiliary data directly to the positioning chip or module through the serial port.

After the positioning chip or module is started, the earlier the injection is, the greater the help for the TTFF will be. However, the injection cannot be done until the positioning chip is started. If the injection is done too early, part of ephemeris data will be lost. The sign for completing the positioning chip starting is that the positioning chip starts outputting the data.

6 Verification

After the ephemeris data injection, the AIDINFO or CFGAID command can be used to verify whether the ephemeris data is injected successfully.

There are two commands to query the auxiliary ephemeris status:

1. AIDINFO command, which is supported only in the new version of firmware and is used as follows:

Table 6- 1 AIDINFO Command Format

Syntax	$AIDINFO
Example	$AIDINFO
Description	Query the status of secondary data. After receiving this command, the receiver outputs $AIDINFO message
Type	Input
No parameter	

Output the auxiliary data information		
Syntax	$AIDINFO,GPSRS,GPSUS,BDSRS,BDSUS,GALRS,GALUS,GLORS,GLOUS,AType	
Example	$AIDINFO,003FFFFFF7,000000FA00,0000003F7F,0000001A3F,0000000000,0000000000,,,7	
Description	Output the status and auxiliary type of auxiliary data	
Type	Output	
Parameter Definition		
Parameter	Type	Description
GPSRS	UINT64	Receive status of GPS ephemeris. As long as the received data is verified, the corresponding bit is set to 1. If the GPS system is not enabled, this field is empty
GPSUS	UINT64	If the GPS ephemeris is valid and can be used for positioning, the corresponding bit is set to 1. If the GPS system is not enabled, this field is empty
BDSRS	UINT64	Receive status of BDS ephemeris. As long as the received data is verified, the corresponding bit is set to 1. If the BDS system is not enabled, this field is empty
BDSUS	UINT64	If the BDS ephemeris is valid and can be used for positioning, the corresponding bit is set to 1. If the BDS system is not enabled, this field is empty
GALRS	UINT64	Receive status of GAL ephemeris. As long as the received data is verified, the corresponding bit is set to 1. If the GAL system is not enabled, this field is empty
GALUS	UINT64	If the GAL ephemeris is valid and can be used for positioning, the corresponding bit is set to 1. If the GAL system is not enabled, this field is empty
GLORS	UINT64	Receive status of GLO ephemeris. As long as the received data is verified, the corresponding bit is

		set to 1. If the GLO system is not enabled, this field is empty
GLOUS	UINT64	If the GLO ephemeris is valid and can be used for positioning, the corresponding bit is set to 1. If the GLO system is not enabled, this field is empty
Atype	UINT	Auxiliary type bit0-3: GPS/BDS/GAL/GLO ephemeris assisted bit4: effective auxiliary position Bit5: use auxiliary position Bit6-7: reserve Bit8: effective auxiliary time Bit9: use auxiliary time Bit10-15: reserve

2. CFGAID command, which is an old version of command also supported in new version of firmware, but may be canceled in later versions and the new command should be used whenever possible. This command is used as follows:

Table 6- 2 CFGAID Command Format

Syntax	$CFGAID,sys	
Example	$CFGAID,0	
Description	Read the received secondary data input	
Type	Input	
Parameter Definition		
Parameter	Type	Description
sys	UINT	System: GPS: 0 BDS: 1

Output the auxiliary data information		
Syntax	$CFGAID,sys,ephmap,almmap,aidflag	
Example	$CFGAID,0,C1B26E01,00000000,2E*39	
Description	Output the auxiliary data status of the system	
Type	Output	
Parameter Definition		
Parameter	Type	Description
sys	UINT	System, as defined by the $CFGAID input
ephmap	UINT	Ephemeris bitmap, hexadecimal, a total of 32 bits, with the low bit as the low satellite number Invalid ephemeris (default) Valid ephemeris
almmap	UINT	Almanac bitmap, hexadecimal, a total of 32 bits, with the low bit as the low satellite number Invalid almanac (default)

		Valid almanac
aidflag	UINT	Auxiliary data type bitmap, hexadecimal bit0~reserved bit1~ ionospheric parameter bit2~UTC parameter bit3~ephemeris parameter bit4~almanac parameter bit5~position time parameter bit6~reserved bit7~reserved setting the corresponding bit to 1 represents that the parameter has been received and resolved

7 Ephemeris Data Form at and Size

The ephemeris data from the server follows the RTCM protocol standard.

Taking GPS+BDS ephemeris for example, the structure of the auxiliary ephemeris downloaded from the server is shown in the figure below:

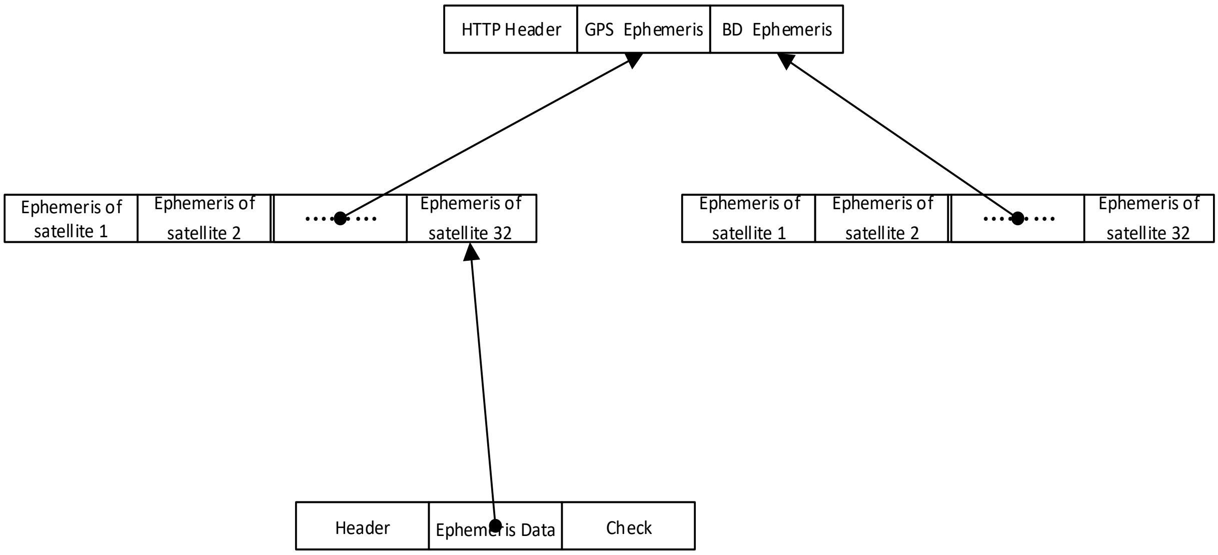

Figure 7- 1 Auxiliary Ephemeris Structure

The ephemeris of each satellite is independent and is a separate unit that contains a separate protocol header and check. Then the ephemeris of each satellite is arranged together to form a complete set of ephemeris

The length of ephemeris of a single GPS satellite is 67 bytes;

The length of ephemeris of a single BDS satellite is 70 bytes;

The length of ephemeris of a single GLO satellite is 51 bytes;

The length of ephemeris of a single GAL satellite is 69 bytes;

The length of ephemeris of a single QZSS satellite is 67 bytes;

The size of the real-time ephemeris data downloaded will vary with the number of available satellites. Currently, there are 32 GPS on-orbit service satellites, 37 BDS on-orbit service satellites, 28 GLO on-orbit service satellites and 30 GAL on-orbit service satellites.

The number of BDS satellites was increased from 14 to 37 by the end of 2018 and will be increased to 64.

The on-orbit service satellite may also be unavailable for a short time due to debugging and other reasons, and the server will not issue the ephemeris data of the satellite.

The method to calculate the size of ephemeris data is as follows:

Single ephemeris length * number of satellites

Example: ephemeris data size of 32 GPS satellites: 67*32 =2144 bytes

8 Instructions on ROM Version of AGNSS

When using AGNSS, it's necessary to ensure that the module has been activated before the module can be infused with AGNSS data; otherwise the data will be lost and the function of AGNSS will fail. It is recommended to use either "COM" or "UM220" keyword to determine that the module has started successfully. "COM" and "Unicore" characters are only printed when the module is restarted.

When the ROM version performs the AGNSS test, the specified command ($reset,0, H01) shall be sent before the above infusion rule is executed. When the command is sent, the module shall be in working state; otherwise, the command will not be executed. Similarly, the keyword shall also be used to determine that the module is in working state. It is recommended to use any keyword of “GGA”, “RMC”, “GSV”, “GSA”, “COM” and “Unicore”.

To sum up, the operation process when the ROM version performs AGNSS operation is as follows:

1. Report the information through the module, continuously test any keyword of “GGA”, “RMC”, “GSV”, “GSA”, “COM” and “Unicore” and confirm that the module is in the working state;
2. Send the command “$reset,0,h01” to the module;
3. Test whether the information reported by the module contains either keyword “COM” or “Unicore” to confirm that the module starts successfully;
4. Issue AGNSS data to the module and stop the tests.

9 Q&A

9.1 About DNS

DNS analytic function

Gethostbyname() function declaration

Gethostbyname() function declaration —— get an IP address with a domain name or hostname

Include header file

```
#include <netdb.h>
#include <sys/socket.h>
```

Function prototype

```
struct hostent *gethostbyname(const char *name);
```

The incoming value of this function is domain name or hostname, such as "www.google.cn". The outgoing value is of hostent structure. If the function call fails, NULL is returned.

Return hostent struct pointer

```
struct hostent
{
char       *h_name;
char       **h_aliases;
int   h_addrtype;
int   h_length;
char**h_addr_list;
#define h_addr h_addr_list[0]
};
```

hostent->h_name

Represent the canonical name of the host. For example, the canonical name of www.google.com is actually www.l.google.com.

hostent->h_aliases

Represent the alias name of the host. www.google.com is the alias name of google.

Sometimes, some hosts may have several alias names, which are in fact entitled by the users for their sites for easy to remember.

hostent->h_addrtype

Represent the type of the host ip address, either ipv4 (AF_INET) or pv6 (AF_INET6)

hostent->h_length

Represent the length of the host ip address

hostent->h_addr_lisst

Represent the host ip address. Note, this address is stored in the network byte order. It is prohibited to print the host IP directly using printf plus %s parameter, which is problematic. To print this IP, call inet_ntop().

const char *inet_ntop(int af, const void *src, char *dst, socklen_t cnt): this function converts the network address structure src of type af to a host-order string, stores it in a cnt string, and returns a pointer to dst. If the wrong function is called, NULL is returned.

Sample code:

```
#include <netdb.h>
#include <sys/socket.h>
#include <stdio.h>

int main(int argc, char **argv)
{
        char        *ptr, **pptr;
        struct hostent *hptr;
        char        str[32];
        ptr = argv[1];

        if((hptr = gethostbyname(ptr)) == NULL)
        {
                printf(" gethostbyname error for host:%s\n", ptr);
        return 0;
                }

        printf("official hostname:%s\n",hptr->h_name);
        for(pptr = hptr->h_aliases; *pptr != NULL; pptr++)
```

```
            printf(" alias:%s\n",*pptr);

    switch(hptr->h_addrtype)
    {
        case AF_INET:
        case AF_INET6:
            pptr=hptr->h_addr_list;
            for(; *pptr!=NULL; pptr++)
                printf(" address:%s\n",
                        inet_ntop(hptr->h_addrtype, *pptr, str, sizeof(str
)));
            printf(" first address: %s\n",
                        inet_ntop(hptr->h_addrtype, hptr->h_addr, str, siz eof(str)));
        break;
        default:
            printf("unknown address type\n");
        break;
    }

    return  0;
}
```

Returned value:

```
# gcc test.c
# ./a.out www.baidu.com
official hostname: www.a.shifen.com
alias:www.baidu.com
address:121.14.88.11
address:121.14.89.11
first address: 121.14.88.11
```

9.2 What If the Ephemeris Data Is Too Large?

If it is considered that the ephemeris data size issued by the server is too large, the data size can be reduced through the ephemeris filtering function.

9.3 Why is Ping Unavailable?

In terms of the network security, the server disables ping, so ping of the ping AGNSS server is unavailable.

9.4 Validity Period of the Real-time Ephemeris

The GPS satellite ephemeris is valid for 4 hours;

The BDS satellite ephemeris is valid for 2 hours;

The GLO satellite ephemeris is valid for 0.5 hours;

The GAL satellite ephemeris is valid for 4 hours

9.5 Does the Injection of Ephemeris Have to Be Continuous?

If the ephemeris data size is 5K but the serial port buffer is only 1K, there is no need to input all the data into the serial port buffer at once. The data can be input in 5 times, with 1K each time. It should be noted that other data cannot be sent to the positioning module in the middle; otherwise part of ephemeris data will be lost.

和芯星通科技（北京）有限公司

Unicore Communications, Inc.

北京市海淀区丰贤东路7号北斗星通大厦三层
F3, No.7, Fengxian East Road, Haidian, Beijing, P.R.China,
100094

www.unicorecomm.com

Phone: 86-10-69939800

Fax: 86-10-69939888

info@unicorecomm.com