

HD AGNSS 应用指南

Version 1.3

Contents

1. HD AGNSS 技术支持流程	3
2. HD AGNSS 系统组成	4
2.1. 系统功能模块	4
2.2. 系统数据流	4
3. AGNSS 方案流程	5
4. HD 星历访问地址	7
5. 通信协议说明	8
6. 服务器上的星历格式	9
6.1. HD_GPS.hdb	9
6.2. HD_BD.hdb	9
6.3. HD_GPS_BD.hdb	9
7. 数据帧格式	10
7.1. AID-PEPH-GPS	10
7.2. AID-PEPH-BDS	12
7.3. AID-POS	14
7.4. AID-TIME	15
7.5. ACK-ACK	17
附录 1 : GPS 星历数据帧实例	18
附录 2 : BD 星历数据帧实例	19
附录 3 : 截取星历信息并发送程序 Demo	20

1. HD AGNSS 技术支持流程

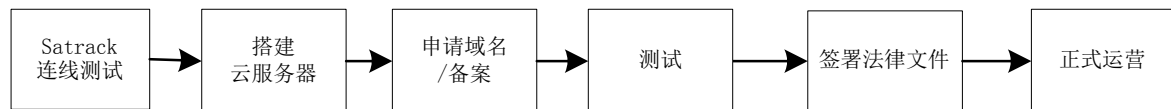


图 2：AGNSS 支持流程

- 阶段 1
 - ✧ 华大北斗提供 EVK 板卡样品和 Satrack 软件工具，供客户进行 AGNSS 的功能测试验证。
具体操作参考《AGNSS 功能测试方法》文档
- 阶段 2
 - ✧ 客户搭建 AGNSS 服务器（云端/本地），实现与 HD 服务器的 FTP 连接
 - ✧ 华大北斗提供星历服务器测试账户，有效期 3 个月；同时对客户服务器域名进行备案
搭建云服务器可以参考《云服务器申请》文档
- 阶段 3
 - ✧ 客户进行 AGNSS 功能及性能的测试及评估
 - ✧ 客户提供测试报告给华大北斗（测试报告格式由双方协商确定）
- 阶段 4
 - ✧ 客户 AGNSS 功能测试通过，提出正式账号申请
 - ✧ 签订 AGNSS 使用相关协议文件，华大北斗提供正式 AGNSS 账户给客户

注：客户服务器要使用域名访问，不能使用 IP 访问，以防止服务器变更 IP 地址时需升级终端主控程序。

2. HD AGNSS 系统组成

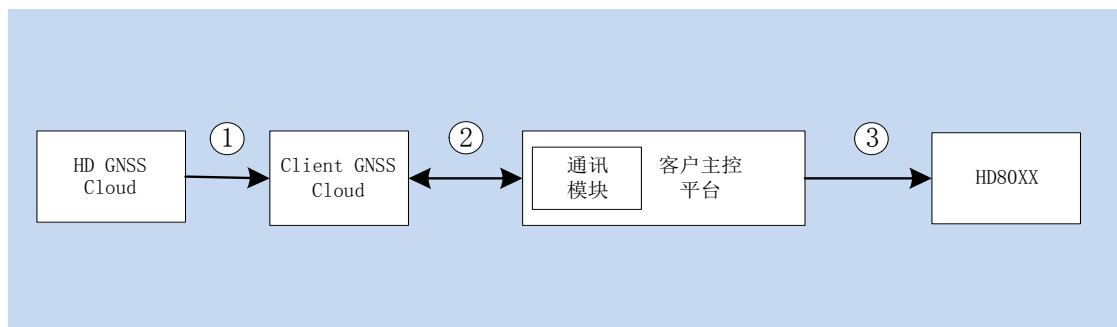


图 1：AGNSS 系统框图

2.1. 系统功能模块

《图 1：HD AGNSS 系统框图》中各功能模块说明如下：

对应框图模块	模块功能说明
HD GNSS Cloud	华大星历服务器，用于存储 GNSS 星历信息（按照 HD-EPH 格式进行存储）
Client GNSS Cloud	客户星历服务器，支持 FTP 连接到 HD GNSS Cloud 服务器下载 GNSS 星历信息
客户主控平台	客户主控平台通过通讯模块建立连接至 Client GNSS Cloud 服务器，下载星历信息到本地
HD80XX	支持 AGNSS 功能的华大北斗 HD80XX 系列芯片，通过接收注入的星历数据，实现快速冷启动，加快定位速度

2.2. 系统数据流

《图 1：HD AGNSS 系统框图》中数据流说明如下：

对应流程	数据流说明
1	Client GNSS Cloud 服务器通过 FTP 连接，从 HD GNSS Cloud 服务器上获取到最新的卫星星历数据
2	客户主控平台从 Client GNSS Cloud 服务器上，通过 HTTP/FTP 协议请求交互获取最新星历数据
3	客户主控平台将星历数据（或第三方获取到的时间或概率位置辅助信息）下发至 HD80XX 系列接收机

3. AGNSS 方案流程

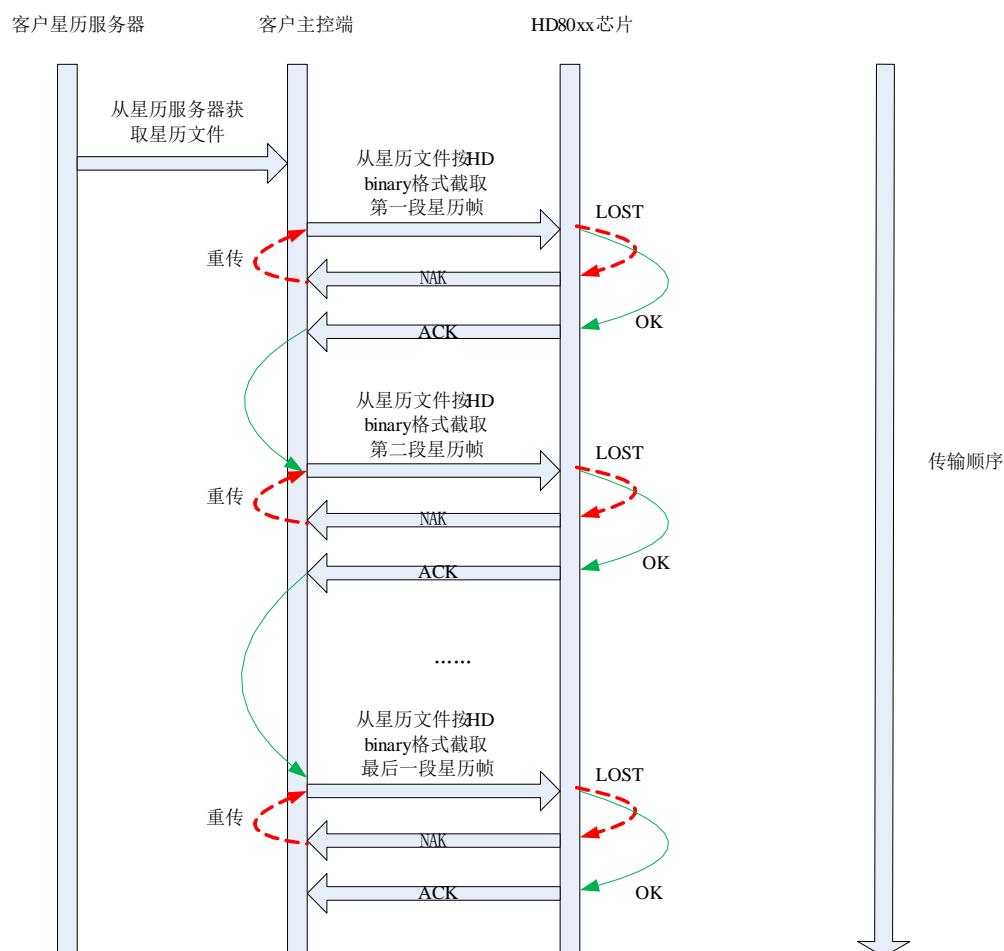


图 2: HD AGNSS 方案流程

1. 客户星历服务器通过 FTP 连接（需访问账号），从华大星历服务器上获取到最新的卫星星历数据（GPS 星历大约 2k 字节，BD 星历大约 1k 字节）；
2. 客户主控端从客户星历服务器上，通过 HTTP/FTP 协议请求和获取最新星历数据,下载到控制端本地（如能从第三方获取到时间信息和概略位置信息，也可一并下载到控制端本地）；
3. 主控端首先给 HD80XX 芯片下发冷启动指令（可去除原过期星历、时间及历书信息，防止对辅助定位有干扰）；
 - 冷启动指令：F1 D9 06 40 01 00 01 48 22
 - HD80xx 芯片接收到冷启动指令后，NMEA 语句中会输出两条标识语句（可用来调试）

```
$GNTXT,02,01,02,HED*19
$GNTXT,02,02,02,ANT_OPEN*43
```

4. 设备的主控单元按照 HD Binary 协议格式发送辅助信息：时间（可选项）、位置（可选项）和星历数据（必选项）。这三个辅助信息需按顺序发送，发送顺序依次为：AID-TIME（可选项）、AID-POS（可选项）、AID-PEPH-GPS/AID-PEPH-BDS 语句。每发送一条语句等待 ACK 后再进入下一条。

注：AID-TIME 时间误差需在 ± 3 秒内，AID-POS 输入位置误差需小于 75km，若超出此范围，该项辅助信息对 AGNSS 辅助定位将无实际帮助意义。

5. 由于星历数据文件比较长，我们需要将星历数据按照 HD Binary 协议格式截取为多帧，客户端主控将截取的每一个星历数据帧依次发送给 HD80XX，HD80XX 接收到星历数据帧后会回复 ACK，主控收到 ACK 后接着发送下一条星历数据帧；
6. 发送完毕后，HD80XX 自动进入辅助定位模式。

4. HD 星历访问地址

华大北斗云服务器会储存最新的星历数据文件（星历分单 GPS、单 BD 和 GPS&BD 双模三种数据文件），供客户服务器下载。访问地址如下：

- ◆ 单 GPS 星历：

ftp://agnss.allystar.com/ephemeris/HD_GPS.hdb

- ◆ 单 BD 星历：

ftp://agnss.allystar.com/ephemeris/HD_BDS.hdb

- ◆ GNSS 双模星历：

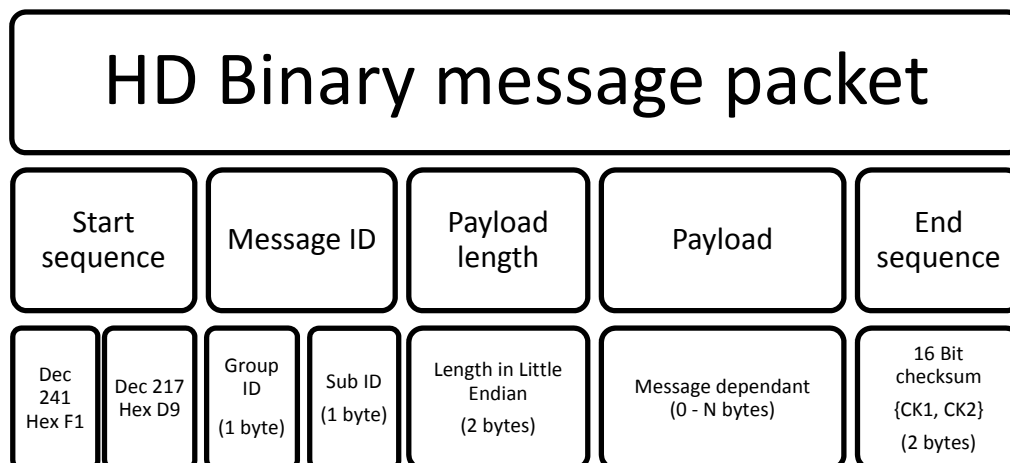
ftp://agnss.allystar.com/ephemeris/HD_GPS_BDS.hdb

- ◆ 调试用http服务器获取星历（无需账号密码）：

http://aclientt.allystar.com/ephemeris/HD_GPS_BDS.hdb

5. 通信协议说明

从星历服务器上下载的星历文件是以如下 HD-Binary 的格式封装好后组成的二进制文件，HD80XX 和主控平台之间的数据传输都使用华大定义的 HD-Binary 格式，其格式定义如下：



字段说明：

start sequence: 同步字节，固定为 0xF1 0xD9

Message ID: 固定占两个字节，由 class ID 和 sub ID 各占一个字节组成

Payload length: payload 的长度，固定占两个字节，最小值为 0

Payload: 可变负载，其长度由具体命令决定

End sequence: 16bit 校验和，计算范围包括从 Message ID 到 payload 的所有数据

主控模块在发送 HD-Binary 格式星历数据时，其通信流程如下：

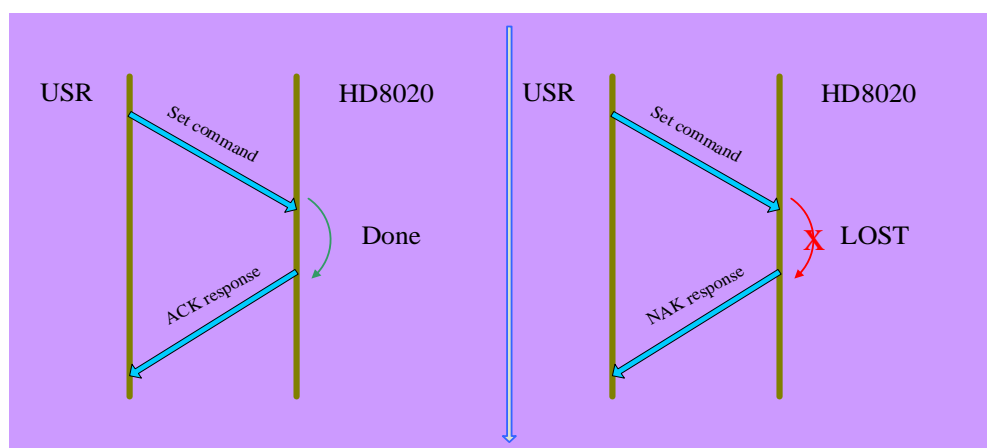
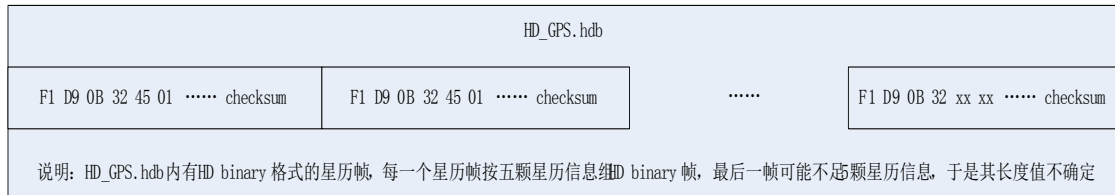


图 3 HD80XX 指令交互

6. 服务器上的星历格式

服务器上的星历文件是以标准的 HD-Binary 的格式封装并组装好放于对应的星历文件中，每一种星历文件其包含的星历数据帧不同，因而文件大小也不一致。

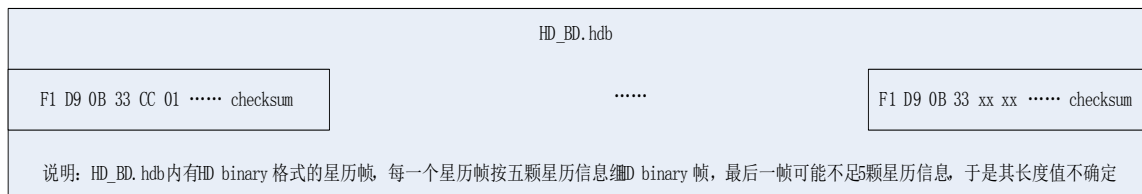
6.1. HD_GPS.hdb



GPS 星历文件的格式内容如下：即其中一段 F1 D9 0B 32 45 01 checksum 为一个星历数据帧，具体格式分析如下：

帧头	Class ID	Msg ID	Length	Payload	Check Sum
F1 D9	0B	32	45 01 (小端模式, payload 中的星历信息长度为 0x0145)	星历信息	2 个字节

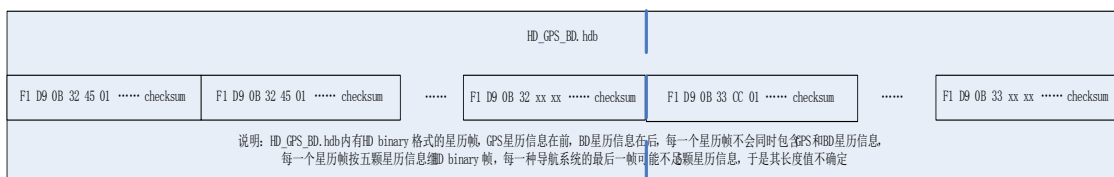
6.2. HD_BD.hdb



BD 星历文件的格式内容如下：即其中一段 F1 D9 0B 33 CC 01 checksum 为一个星历数据帧，具体格式分析如下：

帧头	Class ID	Msg ID	Length	Payload	Check Sum
F1 D9	0B	33	CC 01 (小端模式, payload 中的星历信息长度为 0x01CC)	星历信息	2 个字节

6.3. HD_GPS_BD.hdb



GPS_BD 星历文件的格式包含 GPS 星历文件和 BD 星历文件，文件中前面部分的星历数据帧为 GPS 星历数据帧，后面部分为 BD 星历数据帧。

7. 数据帧格式

本章正文中所引用到的符号意义如下表。

Symbol	Name	Size in bytes
U1	Unsigned char	1
S1	Signed char	1
U2	Unsigned short	2
S2	Signed short	2
U4	Unsigned integer	4
S4	Signed integer	4
R4	float	4

7.1. AID-PEPH-GPS

AID-PEPH-GPS 为 GPS 星历数据帧，注入 GPS 星历时，该长度信息可以随着星历的条数而变，最多五条星历；目前在 AGNSS 服务器上的星历都是按最大 5 个星历信息进行打包，即其 Length=325，最后不足 5 个的则以实际卫星个数进行打包。

例如：若一次注入包含五个卫星的 GPS 星历帧，则该帧长度信息为 $65 \times 5 = 325 = 0x0145$

MESSAGE	AID-PEPH-GPS				
Description	Proprietary Ephemeris Data for GPS				
Type	Polled/Set				
Comment	cynosure II				
Message Structure	Header	ID	Length (Bytes)	Payload	Checksum
	0xF1 0xD9	0x0B 0x32	65	See below	CK_1 CK_2
Payload Contents:					
Byte Offset	Data Type	Scale	Name	Unit	Description
0	U1		reserved		
1	U1		svid		SV ID defined in GPS Satellite system
2	U4	2^{19}	sqrtA		Semi-major axis

6	U4	2^{33}	e		eccentricity
10	S4	$2^{31} \pi$	M0		Mean Anomaly (radian)
14	S2	$2^{43} \pi$	Delta_n		Mean motion correction (radian/sec)
16	U2	2^4	toe		Ref time of Ephemeris
18	S4	$2^{31} \pi$	i0		Inclination angle (radian)
22	S2	$2^{43} \pi$	iDot		Inclination rate (radian/sec)
24	S4	$2^{31} \pi$	Omega0		Longitude of ascending node at weekly epoch (radian)
28	S4	$2^{43} \pi$	OmegaDot		Right Ascension Rate (radian/sec)
32	S4	$2^{31} \pi$	w		Argument of Perigee (radian)
36	S2	2^{29}	Cuc		correction coefficients in ICD
38	S2	2^{29}	Cus		correction coefficients in ICD
40	S2	2^{-5}	Crc		correction coefficients in ICD
42	S2	2^{-5}	Crs		correction coefficients in ICD
44	S2	2^{-29}	Cic		correction coefficients in ICD
46	S2	2^{29}	Cis		correction coefficients in ICD
48	U2	2^4	toc		Ref time of clock
50	S4	2^{31}	af0		SV clock correction term 0
54	S2	2^{43}	af1		SV clock correction term 1
56	S1	2^{55}	af2		SV clock correction term 2
57	S1	2^{31}	tGD		Group Delay
58	i2		weeknum		Ref. week number
60	U2		IODC		Issue of data, clock
62	U1		IODE		Issue of data Ephemeris
63	U1		ura		User range accuracy
64	U1		health		Usage status

7.2. AID-PEPH-BDS

AID-PEPH-BDS 为北斗星历数据帧。注入北斗星历时，该长度信息可以随着星历的条数而变，最多五条星历；目前在 AGNSS 服务器上的星历都是按最大 5 个星历信息进行打包，即其 Length=460，最后不足 5 个的则以实际卫星个数进行打包。

例如：若一次注入包含五个卫星的北斗星历帧，则该长度信息为 $92 \times 5 = 460 = 0x01CC$

MESSAGE	AID-PEPH-BDS				
Description	Proprietary Ephemeris Data for Beidou				
Type	Polled/Set				
Comment	cynosure II				
Message Structure	Header	ID	Length (Bytes)	Payload	Checksum
	0xF1 0xD9	0x0B 0x33	92	See below	CK_1 CK_2
Payload Contents:					
Byte Offset	Data Type	Scale	Name	Unit	Description
0	U1		reserved		
1	U1		svid		SV ID defined in each BDS system
2	U4	2^{19}	sqrtA		Semi-major axis
6	U4	2^{33}	e		eccentricity
10	S4	$2^{31} \pi$	M0		Mean Anomaly (radian)
14	S2	$2^{43} \pi$	Delta_n		Mean motion correction (radian/sec)
16	U4	2^3	toe		Ref time of Ephemeris
20	S4	$2^{31} \pi$	i0		Inclination angle (radian)
24	S2	$2^{43} \pi$	iDot		Inclination rate (radian/sec)
26	S4	$2^{31} \pi$	Omega0		Longitude of ascending node at weekly epoch (radian)
30	S4	$2^{43} \pi$	OmegaDot		Right Ascension Rate (radian/sec)
34	S4	$2^{31} \pi$	w		Argument of Perigee (radian)
38	S4	2^{31}	Cuc		correction coefficients in ICD

42	S4	2^{31}	Cus		correction coefficients in ICD
46	S4	2^6	Crc		correction coefficients in ICD
50	S4	2^6	Crs		correction coefficients in ICD
54	S4	2^{31}	Cic		correction coefficients in ICD
58	S4	2^{31}	Cis		correction coefficients in ICD
62	U4	2^3	toc		Ref time of clock
66	S4	2^{33}	af0		SV clock correction term 0
70	S4	2^{50}	af1		SV clock correction term 1
74	S2	2^{66}	af2		SV clock correction term 2
76	S2	$0.1e^{-9}$	tGD		Group Delay
78	S1	2^{30}	Alpha0		coef. for the amplitude of the vertical delay
79	S1	2^{27}	Alpha1		coef. for the amplitude of the vertical delay
80	S1	2^{24}	Alpha2		coef. for the amplitude of the vertical delay
81	S1	2^{24}	Alpha3		coef. for the amplitude of the vertical delay
82	S1	2^{11}	Beta0		coef. for the period of the model
83	S1	2^{14}	Beta1		coef. for the period of the model
84	S1	2^{16}	Beta2		coef. for the period of the model
85	S1	2^{16}	Beta3		coef. for the period of the model
86	U2		weeknum		Ref. week number
88	U1		IODC		Issue of data, clock
89	U1		IODE		Issue of data, Ephemeris
90	U1		ura		User range Accuracy
91	U1		health		Usage status

7.3. AID-POS

MESSAGE	AID-POS				
:Description	Initial Aiding Data (position)				
Type	Set				
Comment	Cynosure II, LLA or ECEF could be selected				
Message Structure	Header	ID	Length (Bytes)	Payload	Checksum
	0xF1 0xD9	0x0B 0x10	17	See below	CK_1 CK_2
Payload Contents:					
Byte Offset	Data Type	Scale	Name	Unit	Description
0	U1	-	type	-	Position type (1: LLA, 0: ECEF)
LLA					
1	S4	10^{-7}	Lat	degree	Latitude
5	S4	10^{-7}	Lon	degree	Longitude
9	S4		Alt	cm	Altitude
13	U4		Pos_acc	-	Position accuracy
ECEF					
1	S4		ECEF.x	cm	Position x
5	S4		ECEF.y	cm	Position y
9	S4		ECEF.z	cm	Position z
13	U4		Pos_acc	-	Position accuracy

注：AID-POS 为位置辅助信息数据帧，要求提供大概位置的经纬高信息

例如：设置位置信息：lat = 22.5006727, lon = 114.2424747, alt = -882.55

- Command: F1 D9 0B 10 11 00 01 87 54 69 0D AB 04 18 44 41 A7 FE FF 00 00 00 00 6E 4A

7.4. AID-TIME

MESSAGE	AID-TIME				
Description	Initial Aiding Data (time)				
Type	Set				
Comment	Cynosure II, TOW or UTC could be selected				
Message Structure	Header	ID	Length (Bytes)	Payload	Checksum
	0xF1 0xD9	0x0B 0x11	20	See below	CK_1 CK_2
Payload Contents:					
Byte Offset	Data Type	Scale	Name	Unit	Description
0	U1		Type	-	Time type (1: TOW 0: UTC)
1	U1	-	-	-	Reserved
Tow:					
2	U1		Gnss_id	-	Source of time information 0: GPS time 1: BD time 2: GLONASS time
3	U2		Week_no	-	Week number
5	U4		Tow_s	s	Time of week
9	U4		Tow_ns	ns	Nanoseconds time of week, from 0 to 999,999,999
13	U2		Tacc_s	s	Seconds part of time accuracy
15	U4		Tacc_ns	ns	Sub-millisecond part of time accuracy
19	U1		Reserved	-	Reserved
UTC					
2	U1		Leap_sec	s	Number of leap seconds since 1980

					(or <0 if unkown)
3	U2		Year		
5	U1		Month		
6	U1		Day		
7	U1		Hour		
8	U1		Minute		
9	U1		Second	s	
10	U4		Sec_ns	ns	Nanoseconds, from 0 to 999,999,999
14	U2		Tacc_s	s	Seconds part of time accuracy
16	U4		Tacc_ns	ns	Sub-millisecond part of time accuracy

注：AID-TIME 为时间辅助信息数据帧，要求提供大概时间的 UTC 信息

例如：设置时间信息：2016-6-22 15:56:03.288393，tacc = 0.600796，该时刻 leapsecond = 17

➤ Command: F1 D9 0B 11 14 00 00 00 11 E0 07 06 16 0F 38 03 28 87 30 11 00 00 60 6B CF 23 3B A3

7.5. ACK-ACK

MESSAGE	ACK-ACK				
Description	Message acknowledge				
Type	Response				
Comment	The response message for a message ID which is valid and recognized				
Message Structure	Header	ID	Length (Bytes)	Payload	Checksum
	0xF1 0xD9	0x05 0x01	2	See below	CK_1 CK_2
Payload Contents:					
Byte Offset	Data Type	Scale	Name	Unit	Description
0	U1		groupID		Group ID of the acknowledge signal
1	U1		subID		Sub ID of the acknowledge signal

注：当 HD80XX 接收到正确的数据帧时会返回 ACK

例如：

- GPS 星历返回 ACK : F1 D9 05 01 02 00 0B 32 45 73
- BDS 星历返回 ACK : F1 D9 05 01 02 00 0B 33 46 74

附录 1：GPS 星历数据帧实例

例如：一个 GPS 星历数据帧包含五颗 GPS 卫星的星历信息，Payload 长度为 $65 \times 5 = 325 = 0x0145$

```
F1 D9 0B 32 45 01 0B 01 3F 32 0D A1 9F 66 DD 02 BD 5B 69 F9 20 2E A6 59 58 54 50
27 D7 02 AE 74 91 93 EE A9 FF FF 66 9A 73 13 A0 F8 0E 16 BA 16 5F F7 21 00 16 00
A6 59 EA CD 00 00 0B 00 00 0B 6E 03 56 00 56 02 00 0B 02 D9 B9 0C A1 CB B4 15 08
8E 00 92 08 C4 33 A6 59 DB 31 73 26 1D 02 24 C5 B0 91 93 A5 FF FF AB C9 1D AA
4A F7 1E 18 38 13 5E F6 98 FF 79 00 A6 59 C0 1E 13 00 E9 FF 00 D6 6E 03 36 00 36
02 00 0B 03 F8 62 0D A1 93 C2 1C 00 62 1F C6 21 64 36 A6 59 D9 3D 11 27 9A 05 90
3D 00 BE E5 A6 FF FF 30 4A AF BC 1B 03 54 0F EA 1D 82 03 ED FF 10 00 A6 59 3E
B7 FD FF E3 FF 00 03 6E 03 50 00 50 02 00 0B 04 83 3B 0D A1 AE B7 5B 04 5F 81 3E
7B E3 38 A6 59 A8 73 0D 28 08 03 6E BE EF 3E 79 A9 FF FF 60 62 9B 24 15 0A 08 08
90 27 A8 0B 09 00 D6 FF A6 59 74 8A F7 FF FD FF 00 D7 6E 03 4C 00 4C 02 3F 0B 05
F2 7F 0E A1 69 EF 5F 02 B4 B2 7A 98 DC 38 68 5B 30 65 8A 26 6F 06 22 B5 7F BD 43
A6 FF FF 68 0C 90 12 34 03 8C 0F FE 1C 3A 04 F6 FF DA FF 68 5B 3F 5F FC FF 19 00
00 E9 6E 03 13 00 13 02 00 F0 A5
```

附录 2：BD 星历数据帧实例

例如：一个 BD 星历数据帧包含五颗北斗卫星的星历信息，Payload 长度为 $92 \times 5 = 460 = 0x01CC$

```
F1 D9 0B 33 CC 01 0B 01 80 08 EC CA 62 17 2E 00 E6 D7 97 E1 45 2F BA DB 00 00
58 55 94 02 AA 03 B1 76 11 B3 AF DB FF FF E6 59 E2 07 51 FD FF FF 9F 5D 00 00 33
AC FF FF 01 FF FF FF 42 FF FF FF B3 FF FF FF BA DB 00 00 E7 15 10 00 EF C0 00
00 00 00 8E 00 0A 0E EA 38 3E CB 7F 81 55 02 00 01 00 00 0B 02 95 9B EB CA 85 90
2D 00 9B D2 4A 8B 74 2A BA DB 00 00 58 4E EE 03 6C 03 D0 F9 C2 AC 97 E1 FF FF
78 17 A3 3C C3 E8 FF FF 0F 6E 00 00 D4 9B FF FF 97 EC FF FF 81 FF FF FF CC FF
FF FF BA DB 00 00 78 06 4F 00 09 8D FF FF 00 00 34 00 0A 0E EA 38 3E CB 7F 81 55
02 00 01 00 00 0B 03 D2 2A EA CA E9 80 49 00 60 41 2B 4A 99 2A BA DB 00 00 42 11
E9 02 D2 03 ED 6C D1 AF CD E1 FF FF 8A BA 83 8D 61 00 00 00 F2 6D 00 00 FB 9C
FF FF 5B 02 00 00 21 FF FF FF 4B FF FF FF BA DB 00 00 7D FD D7 FF E0 EB 00 00
00 00 44 00 0A 0E EA 38 3E CB 7F 81 55 02 00 01 00 00 0B 04 9B 52 EB CA 27 66 2B
00 DA F9 5A D0 DD 28 BA DB 00 00 06 C8 56 03 96 03 FD A1 52 B3 46 E1 FF FF 6A
1A 0D 27 C8 FC FF FF A6 4F 00 00 3D B9 FF FF 01 FE FF FF F0 FF FF FF 30 FF FF
FF BA DB 00 00 19 A9 DA FF E1 72 00 00 00 00 39 00 0A 0E EA 38 3E CB 7F 81 55 02
00 01 00 00 0B 06 EB 18 EA CA 0C F1 FC 02 22 A2 F7 FA 52 0F 74 D6 00 00 F8 E6 7E
26 A5 05 A6 32 11 D1 D5 E9 FF FF 32 65 D8 A1 E7 40 00 00 21 81 00 00 6E C2 FF FF
A6 3B 00 00 A2 00 00 00 FB 01 00 00 74 D6 00 00 0C 4B 71 00 2E 1D 06 00 00 00 65
00 0A 0D EA 38 3F C9 7F 81 55 02 00 01 00 00 B0 1F
```

附录 3：截取星历信息并发送程序 DEMO

```
/**
 * @brief get the eph frame and send to HD80xx
 * @param data: the pointer of the eph data file
 * @param len : the total length of eph data file(eg:the length of HD_GPS_BD.hdb )
 * @retval 0: successs -1 error
 */

static int gnss_eph_inject_data(char *data, int length)
{
    LOG("data length is %d\n",length);
    LOG("data:d[0,1,2,3,4] is %x,%x,%x,%x\n",data[0],data[1],data[2],data[3],data[4]);
    unsigned int f1_flag = 0; //f1 位置标识
    unsigned int frame_len = 0; //每一个 HD binary 中的 payload(即星历信息)的长度
    char *pdata;
    int nwrite=0;
    pdata = data;
    while(f1_flag<length-1)
    {
        if((pdata[f1_flag]==0xF1) && (pdata[f1_flag+1]==0xD9))
        {
            // get each eph frame length
            frame_len = (unsigned short)(pdata[f1_flag + 4] | (pdata[f1_flag + 5]<<8));
            nwrite = uart_send(pdata+f1_flag,frame_len+8);
            if(nwrite<0)
            {
                return (-1);
                LOG("gnss_eph_inject_data Error!\n");
            }
        }
        else
        {
            LOG("the pdata is not 0xF1:%x,%x\n",pdata[f1_flag],pdata[f1_flag+1]);
            break;
        }
        sleep(3); // just example, use wait time 3ms instead of ACK
        f1_flag = f1_flag + frame_len + 8;
    }
    return 0;
};
```