Goke AGPS User Manual

Legal Notices

If you receive this document from Hunan Goke Microelectronics Co., Ltd. (hereinafter referred to as "Goke Micro"), you have agreed to the following terms. If you do not agree to the following terms, please stop using this document.

The copyright of this document is reserved by Hunan Goke Microelectronics Co., Ltd., and any rights not expressly granted in this document are reserved. Without the prior written permission of Goke Micro, no unit or individual may copy, transmit, distribute, use and disclose this document and any pictures, forms, data and other information contained in this document.

The company reserves the right to modify and improve the products described in this manual without prior notice;

reserves the right to revise or withdraw this manual at any time.

If there are any unclear words in this user manual, please consult the company, agents, and sellers in time.

1 Introduction to **GPD**

GPD is an implementation method defined by Goke to realize AGPS assisted positioning. Mainly used from

The IGS website obtains the current navigation data Rinex file, and then converts it into the current ephemeris through the serial port

Send it to the chip, thereby realizing the acceleration of the positioning of the GPS chip.

2 How to get the **GPD** file

Download the GPD file corresponding to the current time by visiting Goke's GPD server website (http://www.goke-agps.com:7777/brdcGPD.dat). because The real-time ephemeris published on the IGS website is updated every 2 hours, so the relative GPD file will also be updated every 2 hours.

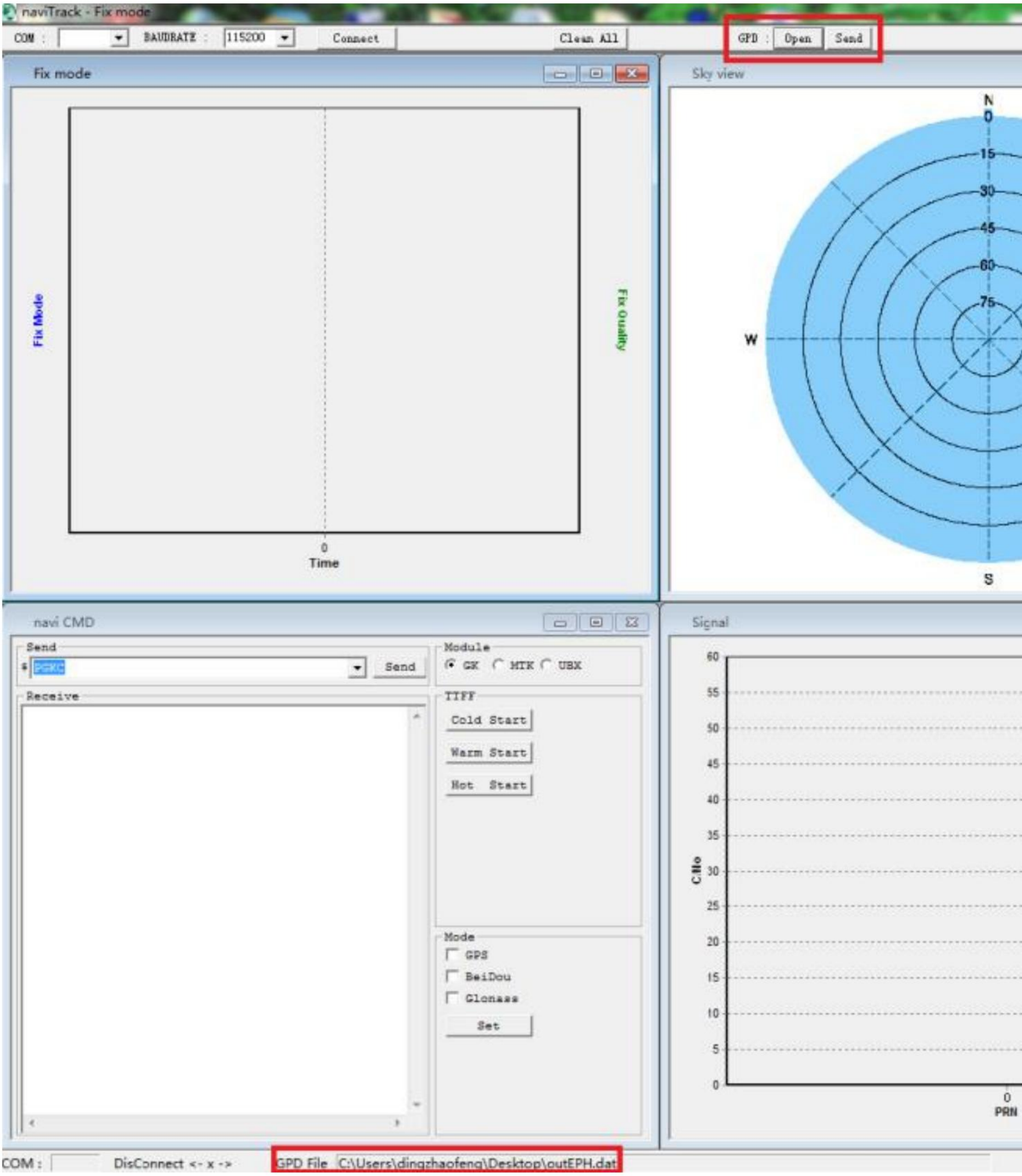
3 How to use **GPD** files

The naviTrack tool provided by GOKE is uploaded to the chip through the serial port. 1. After the chip

is powered on, click the "open" button next to the GPD icon above, and select the GPD file downloaded from the Internet. The selected file

information is displayed below the tool. 2. After successfully selecting the file, click the "send" button, and the tool will start uploading. 3. After waiting for a

while, a completion prompt box will appear, indicating that the upload is successful, otherwise it will fail and upload again.



BAUDRATE : 115200

$ PGKC

GPD File C:\Users\dingzhaofeng\Desktop\outEPH.dat

4How to delete the **GPD** data in the chip

Since the GPD data is only valid for 6 hours, if it exceeds the time limit, the GPD in the chip will

Data will not work. Users can clear the GPD data in the chip by sending serial port commands. when

However, when uploading new GPD data each time, the original old GPD data will be cleared first.

Delete GPD data command: enter in the command input box of the naviTrack navigation tool

“PGKC047”.

5 Effects after using **GPD**

Since the ephemeris data of the currently visible satellites has been obtained, the positioning time can be effectively reduced. After using GPD, the cold start positioning time can be improved by about 10~15 seconds. Especially in the case of weak signal, it is possible to improve more positioning speed.

6 Auxiliary positioning method to accelerate **GPD**

Since the pure GPD assistance still needs to rely on the GPS time information of searching the sky, sometimes it will consume too much time when the signal is poor. Setting the current time information and rough coordinate information through the PGKC639 command can achieve faster positioning time. Note: When setting through the 639 command, the latitude and longitude range should

deviate from the actual position by 20km

Within , the time deviation should not exceed 5 minutes.

Command:639

Set approximate location information and time information to speed up positioning.

Arguments: Arg1:

latitude, for example: 28.166450 Arg2: longitude, for

example: 120.389700

Arg3: altitude, for example:

0 Arg4: year Arg5: month

Arg6: day Arg7: hour, time

is UTC time Arg8: minute

Arg9: second Example:

```
$PGKC639,28.166450,120.389700,0,2017,3,15,12,0
          ,0*33<CR><LF>
```

After the 639 command is successfully executed, GK9501 will return the following format:

```
$PGKC001,639,3*21
```

7 GPD communication transmission

The GPD data is sent to the chip in blocks mainly through serial communication. The main process is as

follows: a) Switch NMEA reception to BINARY reception mode (for details about the command format, see GK9501 Input and Output

format.pdf)

Send: message type + switching mode + baud rate +

CheckSum data: $PGKC149,1,115200*15 (the message type

transmitted by GPD is 149)

Receive: packet header (2B) + packet length (2B) + ACK type (2B) + message type (2B) +

valid flag (1B) + CheckSum (1B) + packet tail (2B) Data: 0xaa, 0xf0, 0x0c, 0x00

,0x01,0x00,0x95,0x00,0x03,(chk),0x0d,0x0a

(checksum is byte-by-byte XOR from the packet length field to the field before checksum)

b) Send the first data block of GPD, wait for ACK response to

send: packet header (2B) + packet length (2B) + transmission type (2B) + GPD packet number (2B) + data payload (512B) + CheckSum (1B) + packet Tail (2B) Data: 0xaa, 0xf0, 0x0b, 0x02, 0x066, 0x02, 0x00, 0x00,..., (chk), 0x0d, 0a

Receive: packet header (2B) + packet length (2B) + ACK type (2B) + GPD packet sequence number (2B) + valid flag (1B) + CheckSum (1B) + packet tail (2B) Data: 0xaa, 0xf0, 0x0c, 0x00,0x03,0x00,0x00,0x00,0x01, (chk),0x0d,0x0a

c) Send the remaining data blocks in sequence and wait for the

ACK response to be sent: packet header (2B) + packet length (2B) + transmission type (2B) + GPD packet number (2B) + data payload (512B) + CheckSum (1B) + packet Tail (2B) Data: 0xaa, 0xf0, 0x0b, 0x02, 0x066, 0x02, 0x01, 0x00,..., (chk), 0x0d, 0a (GPD file is divided into 512-byte data blocks for transmission, and the last block is insufficient 512 bytes padded with 0)

Receive: packet header (2B) + packet length (2B) + ACK type (2B) + GPD packet sequence number (2B) + valid flag (1B) + CheckSum (1B) + packet tail (2B) Data: 0xaa, 0xf0, 0x0c, 0x00,0x03,0x00,0x01,0x00,0x01, (chk),0x0d,0x0a

d) Send the GPD transmission end statement, wait for the

response to be sent: packet header (2B) + packet length (2B) + transmission type (2B) + GPD terminator (2B) +CheckSum (1B) + end of packet (2B) Data:

0xaa, 0xf0, 0x0b, 0x00, 0x066, 0x02, 0xff, 0xff, (chk), 0x0d, 0a

Receive: packet header (2B) + packet length (2B) + ACK type (2B) + GPD terminator (2B)

+ Valid Flag (1B) + CheckSum (1B) + End of Packet (2B)

Data: 0xaa, 0xf0, 0x0c, 0x00, 0x03, 0x00, 0xff, 0xff, 0x01, (chk), 0x0d, 0x0a

e) Switch BINARY reception to NMEA reception

mode and send: packet header (2B) + packet length (2B) + message type (2B) + transmission type (1B) + baud rate (4B) + CheckSum (1B) + packet end (2B)) Data: 0xaa, 0xf0, 0x0e, 0x00, 0x95, 0x00, 0x00, 0x00, 0xc2, 0x01, 0x00, (chk),

0x0d,0x0a

Receive: packet header (2B) + packet length (2B) + ACK type (2B) + message type (2B) + Valid Flag (1B) + CheckSum (1B) + Packet Trailer (2B)

Data: 0xaa, 0xf0, 0x0c, 0x00, 0x01, 0x00, 0x95, 0x00, 0x03, (chk), 0x0d, 0x0a (status flag: 0 means invalid, 1 means not supported, 2 means failed, 3 means successful)