

NASA BTLE Navtex Engine Interface Specification.

INTRODUCTION

The NASA BTLE Navtex Engine uses a Microchip RN4020 Bluetooth Low Energy (BTLE) device for external communications, operated in the MLDP mode. Its default name is BTnavtex, which can be changed as described in commands, below.

The Navtex Engine is normally always connected to its antenna and stores incoming Navtex messages in its non-volatile EEPROM. The Engine does not communicate via the MLDP interface until commanded to do so by the App. The App. is at all times the client for data from the Navtex Engine.

There are two types of message between the Navtex Engine and the App. which controls the traffic: a command string from the App. to the Navtex Engine to define the data requirement, and data from the Navtex Engine to the App. in response to a command. The data returned are listed with the commands below.

A PIN is required to validate all commands which change operating system parameters such as the clock, switching times, A/B setting, etc., in the commands listed below. The requirement for the PIN in such commands is shown by the presence of **,PPPP** after the command letter.

New Engines are delivered from the factory with a PIN number of 0000, and it is wise to change this number as soon as possible after starting to use the unit.

COMMAND STRING FORMAT

The command string sent to the Navtex Engine defines the data or the action requested.

All commands start with a '\$'. For some commands, only a single letter is required. The terminator for all commands is a '~' (tilde) symbol. The Engine responds with data as specified and all commands, whether returning data or not, get a response of "ok" These commands are as follows:

1. **\$C,PPPP,hh:mm~** Set the clock, where hh:mm is the time in UTC. All four digits must be present. If valid, the Engine responds with the time sent in the format hh:mm UTC. Times outside the range 00:00 to 23:59 are not accepted by the Engine.

It is advisable to send the App's system clock time (which is local time), and corrected to UTC, to the BTLE Navtex Engine whenever the device is accessed because its internal clock is less accurate than the system clock.

2. **\$A,PPPP,aa:aa,cc:cc~** Set up the times (which are stored) to switch to channel A (518kHz), where aa:aa and cc:cc are the two UTC times to effect the switches. The four digits must be present in both cases. The Engine responds with an ordered list of all A and B switching times on separate lines.

3. **\$B,PPPP,bb:bb,dd:dd~** Set up the times (which are stored) to switch to channel B (490kHz), where bb:bb and dd:dd are the two UTC times to effect the switches. The four digits must be present in both cases. The Engine responds with an ordered list of all A and B switching times on separate lines.

4. **\$A,PPPP~** Immediately switch to channel A (518kHz) and stop timed switching.

5. **\$B,PPPP~** Immediately switch to channel B (490kHz) and stop timed switching.

6. **\$E,PPPP~** Switch on timed A/B switching using the stored times. The timed switching state at power-on is off. The switching times are listed in order by the \$T~ and \$#~ commands.

7. **\$D~** Switch ON display of duplicated header IDs for all the following commands.

8. **\$H~** Switch OFF display of duplicated header IDs for all the following commands.

9. **\$F,abcd~** any of a,b,c, or d may be a "?" acting as a wild card. The Engine responds with a comma-delimited list of only the matching stored message header IDs, terminated by the word "end" and a CR/LF couplet. The first ID is the most recently received matching ID and the last is the oldest matching message ID.

10. **\$L~** The Engine responds with a comma-delimited list of all the stored message header IDs, terminated by the word "end" and a CR/LF couplet.

(The command is the same as issuing \$F,????~ as above). The first ID is the most recently received and the last is the oldest available ID (unlike the original RS232 Navtex Engine, which sent oldest first and newest last).

11. **\$R,AB26~** Request a download of all stored messages (AB26, for example) defined by its header ID. The most recent is sent first. If no matching header ID is found, "Message ID NNNN not found." is returned.

12. **\$S,AB26~** Request a download of a single message (AB26, for example) defined by its header ID. Only the most recent is sent. If no matching header ID is found, "Message ID NNNN not found." is returned.

13. **\$N,PPPP,New_name~** change the Engine's name to New_name. Only 8 characters are allowed. After sending this command, the Engine waits for 15 seconds during which interval the LOAD DATA button must be pressed to validate the change. If the button is not pressed within the interval, the command is ignored.

14. **\$P,nnnn~** Change the 4-character PIN to nnnn, where nnnn are printable ASCII characters. After sending this command, the Engine waits for 15 seconds during which interval the LOAD DATA button must be pressed to validate the change. If the button is not pressed within the interval, the command is ignored.

15. **\$T~** Show a snapshot the Engine's clock with the message "Time now is hh:mm UTC". Note that the clock remains stopped on 00:00 until it has been set. If A/B switching has been programmed, the switching times are also listed in order.

16. **\$W,PPPP~** This command wipes all messages older than the most recent 20 and cannot be undone.

17. **\$V~** The Engine responds with its firmware version.

18. **\$#~** The Engine responds with all its settings: Name, PIN, time now, A/B switching status, switching times, present channel selection, and duplicate header ID display status. A typical example is as follows, where btNAVTEX is the unit's name:

> btNAVTEX

PIN is 0000
Time now is 11:35 UTC
A/B_switching is OFF
switch to A at 00:00 UTC
switch to B at 06:00 UTC
switch to A at 12:00 UTC
switch to B at 18:00 UTC
Unit is now on Channel A
Duplicated IDs are hidden.
<

MESSAGE STRING FORMAT

A typical message, for example, following a \$R,SB06~ command is displayed thus:

> SB06
300711 NAVTEX-HAMBURG (NCC)
WARNING NO*M QQP EPPUPTUTC SEP
FOR GERMAN BIGHT:

a13
<

The '>' character is the start-of-received message marker. The 'a' character is the end-of-received message marker. Everything between these markers is the total message content, as received, including all the embedded formatting. All the message body is always in Upper Case or numbers and punctuation.

The letter 'a' shows that the message was received using the 518kHz (A) setting. Letter 'b' at this position would show that the message was received using the 490kHz (B) setting.

The number after the letter 'a' is the number of Forward Error Corrections (FEC) performed on the message during reception. In this example, FEC did not achieve a great deal of success, because it is almost certain that the number shift character was corrupted after NO*).

Smaller values denote better reception quality. '*' characters denote characters which could not be corrected using FEC (and do not add to the FEC count after the 'a' character).

The '<' character is the end-of-stored message marker followed by a CR.