

Format of files and UDP-streams used on D-STAR

http://villazeebries.krbonne.net/hamstuff/?page_id=12

The goal of this document is to describe the format of file-formats and UDP network-streams used on D-STAR.

*** D-STAR STREAM-FORMAT**

Sending a voice-message to a repeater is achieved by sending a UDP-stream to the “dsgwd” process on the repeater gateway-PC. This application listens by default on UDP port 40000. Packets needs to be send every 20 ms.

The format of this UDP-stream to be send is described below. This stream-format is very similar to the file-format of the .dvtool files. The differences between these two formats are described further down in this text.

1/ The configuration-frame:

The first frame of the UDP-stream is a start-frame and is 56 bytes in length. It contains addressing information about the stream:

- octets 0 to 3: “DSVT”
- octet 4: 0×10 (= frame is configuration-frame)
- octets 5 to 7: 0×00, 0×00, 0×00
- octet 8: 0×20 (= stream is voice-stream)
- octets 9 to 11: 0×00, 0×01, 0×01
- octets 12 to 13: streamid (random, 16 bits)
- octet 14: 0×80
- octets 15 to 17: 0×00, 0×00, 0×00 (flag1, flag2, flag3) (*)
- octets 18 to 25: destination = repeatername + destination module
- octets 26 to 33: departure = repeatername + ‘G’
- octets 34 to 41: companion = “CQCQCQ ”
- octets 42 to 49: own1 = repeatername
- octets 50 to 53: own2 = “RPTR”
- octets 54 to 55: packet frame checksum

(*) The exact description of the flags can be found in the “shogen” DSTAR documentation. Some D-STAR applications set flag1 to “0×40”. This sets the “repeater” bit (bit 6) to 1.

2/ A voice-packet

The concequative frames in the UDP-stream are 27 byte voice-packet. They contains 20 ms of voice-data. The frame-format is described below:

- octets 0 to 3 : “DSVT”
- octet 4: 0×20 (= frame is voice-frame)
- octets 5 to 7 : 0×00, 0×00, 0×00
- octet 8: 0×20 (= stream is voice-stream)
- octets 9 to 11: 0×00, 0×01, 0×01
- octets 12 to 13: streamid (random, 16 bits, same as in header)
- octet 14: framecounter (goes from 0 to 20)
- octets 15 to 23: ambe-data

- octets 24 to 26: slow-speed data

In the last frame of a stream, the frame-counter is increased by 0×40 .

The .dvtool file format

As already mentioned, the format of the .dvtool files (binary files containing AMBE-encoded audio) is very similar to the network stream-format used to transport D-STAR DV-streams.

In fact, the only difference between these two formats, are 10 bytes which are added in front of the configuration-frame. They contain the following information:

- 6 octets contain "DVTOOL"
- A 4 octet length-indication (in little-endian format), indicating the number of frames present in frame (startframe + voiceframes)
-
- In front of each frame, a 2 byte length field is added, formatted in small-endian format. It contains $\{0 \times 38 \ 0 \times 00\}$ for a configuraton-frame or $\{0 \times 1b \ 0 \times 00\}$ for a voice-frame.

Note that, when converting a .dvtool to an actual UDP-stream send to the repeater, some parts of the information in the .dvtool file is overwritten:

In the configuration-frame

- all repeater-information (like the name of the repeater and the module) is replaced by the actual name and module that is used.
- A new streamid is generated
- A new packet checksum is calculated.

In the voice-frames:

- the streamid is replaced to match the one chosen in the configuration-frame.

The .AMBE file format

The .ambe files are designed to hold AMBE pre-encoded audio-fragments; like individual words or parts of sentences.

These files can be easily concatenated together to make up a complete voice-announcement. This is possible as .ambe-files are just plain text-files.

The format-description of the .ambe format is as follows:

- .ambe files are plain ascii
- any line beginning with '#' is considered to be comment and is ignored
- Althou comments-lines are ignored, some lines are reserved for future expansion:
 - Lines beginning with "#C Version: " contain the version of the protocol.
 - The current protocol-version is 1.0
 - Lines beginning with "#C Name: " contain the name of the ambe-file
 - Lines beginning with "#C Info: " contain any additional information

Currently, all these three lines are optional

- Lines containing actual AMBE-data use this format: 00000 00 AABBCCDDEEFF001122
- The 1st and 2nd are purely informational. They field contain sequencing-information of ambe frame. The 1st field contain the seconds (digits from 00000 to 99999) the 2nd field contains hundreds of seconds (digits from 00 to 99) since the beginning the encoding-process.

As DSTAR sends and receives a AMBE-frame every 1/50th of a second, the 2nd field will go up by 2 for every frame.

As .ambe files are intended to be used to build voice-messages containing multiple smaller .ambe files; this timing-information is not used in the actual process of producing the resulting .dvtool file. They are only there for informational reason, so give some idea of the length of the resulting audio-message.

- The 3th field contains the actual AMBE data.

As D-STAR uses AMBE voice-encoding at 3600 bps (2400 bps audio + 1200 bps FEC), any .ambe frame of 1/50 of a second contains 9 bytes of .ambe audio. (3600 bps = 450 bytes per second -> divide that by 50 frame per second and you get 9 bytes per frame)

The AMBE-data is encoded as 18 hex-characters.

- .ambe-files do NOT contain slow-speed data information.

The slow-speed-data part

The slow speed data is stored in the 3 last octets of every DV-frame (hereafter called a “packet”) just behind the 9 AMBE octets of voice-data. The “shogen” DSTAR documentation only specifies two things:

- slow-speed data is organised in a “superframe” structure of 21 packets of 3 bytes (the 3 octets at the end of 21 consecutive DV-frame).

The 1st packets contains a 24 bit synchronisation-pattern:

- twice the “7 bit-sequence maximum-length sequence” 1101000 (read from LEFT to RIGHT, so 0001011)
- A 10 bit synchronisation-pattern “1010101010” (read from LEFT to RIGHT, so 0101010101)

In addition to that, the order of the 3 octets is reversed.

- Before sending, the default D-STAR scrambler is applied to the slow-speed data. This can be emulated by doing a xor operation on the 3 bytes of the slow-speed data in every DV-frame with the values 0×70, 0×4f and 0×93 (for octets 1 to 3 of the slow-speed packet)

Note that the D-STAR scrambler is NOT applied to the synchronisation-packet (i.e. the slow-speed data present of the first packet of a superframe)

ICOM has implemented an extension to the slow-speed data protocol, which has been reverse-engineered by Jonathan G4KLX and Denis Bederov DL3OCK. See this URL for more info: <http://www.qsl.net/kb9mwr/projects/voip/dstar/Slow%20Data.pdf>

In the current version of the voice-announcement package, only the ”20 bytes free-text” extension is implemented. This extensions allows for a 20-byte text to be send along in a DV-stream. That text will be shown on the display of the transceiver.

The specifications are as follows:

- The 20 bytes of data are divided in 4 “groups” of 5 octets each.
- Each group is distributed over two 3-bytes slow-speed packets in a DV-frame
- The 5 bytes of text-data are preceded by 1 octet containing 0×40 up to 0×43.

- If the message is less than 20 bytes, it is filled up with spaces
- The remaining part of the slow-data superframe is filled up with a filler (containing "0x66").

73

Kristoff ON1ARF