APRS® and D-STAR® = D-PRS™

Peter Loveall AE5PL

Abstract

D-STAR®i is a digital over-the-air protocol developed by the Japan Amateur Radio League, Inc. (JARL) which supports Ethernet at 128 kbps (DD) or digital voice at 4800 bps (DV). DV uses 3600 bps for voice (2400 AMBE encoding, 1200 bps FEC) and 1200 bps for synchronization and multiuse (approximately 900 bps is available for general use). APRS®ii is a protocol designed by Robert Bruninga WB4APR to communicate information such as positions, weather, etc. using AX.25 as a transport protocol. It has been adapted to use any clear-text protocol such as Telnet. This paper explores the methods used to bridge Icom D-STAR® radios running GPS with APRS-IS (APRS® Internet Service) and other APRS clients. We also explore using the two D-STAR® protocols to carry APRS® information over them.

Keywords

APRS, D-STAR, D-PRS

Introduction

Icom Incorporated Corporation has introduced a number of radios with D-STAR® capability. All support the digital voice (DV) protocol and the ID-1 (1.2 GHz) also supports the digital data (DD) protocol. D-STAR® was developed by the Japan Amateur Radio League, Inc. to support fully digital modes of communication in the VHF, UHF, and higher Amateur bands. The protocol is an open protocol which uses the proprietary voice codec AMBE. (Japan Amateur Radio League, Inc., 2005)

APRS® is a protocol developed by Robert Bruninga WB4APR as a standardized method for communicating automated information such as position, weather, and telemetry. The protocol is based on AX.25 but has been adapted for most any transport protocol by using the TNC-2 monitor format for packets. (TAPR APRS Working Group, 2000)

Icom placed a GPS reporting capability in all of the VHF and UHF D-STAR® radios (not included in the 1.2 GHz ID-1) which makes use of the available general-use low-speed portion of the DV data stream. This functionality was added to the Icom radios as an effective, complimentary use of the digital voice channel. This usage is complimentary because it is non-interfering to the primary purpose of the DV data stream which is voice. The basic GPS mode is simply a pass-through of the GPS NMEA strings with a station identification line added for simple decoding at the remote end. With the IC-2820, Icom

introduced a second GPS mode called GPS-A which sends a TNC-2 format APRS packet with a CCITT-CRC wrapper for error detection.

This paper demonstrates the ability to use the high-speed digital data (128 kbps) to support APRS-IS and the ability to bridge the GPS information from the Icom D-STAR® radios to APRS-IS. As an added benefit of the latter, the bridge can also pass APRS packets via the D-STAR® DV low-speed channel in TNC-2 format.

APRS-IS Via DD

The 1.2 GHz radios support a 128 kbps digital data transport which is, essentially, an Ethernet bridge. Because this is a bridge, it is recommended to place a firewalled NAT router at each end. This is automatically done at a "repeater" when used in conjunction with the gateway. The function of the firewalled NAT router is to remove all of the unnecessary broadcast packets such as those used by NETBIOS. If one end of the communications is connected to the Internet, then it becomes a simple matter to use the 128 kbps DD for APRS.

The user of an ID-1 (currently the only end-user radio which supports DD) connects their PC to their router which is then connected to the ID-1 via appropriate Ethernet cables. The ID-1 is set to DD mode on a frequency with another ID-1 or a DD "repeater". The word "repeater" is in quotations because this is really an access point and does not repeat packets. DD is a half-duplex simplex protocol. The end-user then starts their favorite APRS client configured to talk to a **filtered** port on an APRS-IS server. Because of the half-duplex nature of DD and the shared nature of the channel, it is very important to limit the amount of data being received by the client. The filtered ports available from javAPRSSrvr servers (the majority of servers on APRS-IS) make this very easy to do with most APRS clients that have Internet capability.

No other special configuration of the APRS client or the ID-1 is required. As far as the APRS client and the PC are concerned, they have a direct pipe to the Internet.

D-PRS™ Description

D-PRS™[iii] was developed by Peter Loveall AE5PL to present the Icom GPS information to APRS clients and to APRS-IS. The D-STAR® protocol defines a 4800 bps digital voice (DV) mode. The DV mode contains an RF header, 3600 bps for voice, and 1200 bps for data and synchronization. The RF header is sent using forward error correction and also contains a 16 bit CCITT CRC for further error detection. The AMBE codec also includes forward error correction. The 1200 bps reserved for general purpose data and D-STAR® synchronization carries no error detection or correction. It is up to the users of the general purpose data stream to incorporate their own error detection and/or correction.

Icom defined a mechanism for sending current GPS positions for a station while that station is transmitting voice. As an extension, Icom also gave the user the ability to periodically transmit their position regardless of whether they are using the radio for voice. It is important to note here that all discussions in the D-PRS™ sections of this paper are referencing D-STAR® radios in digital voice mode, not analog mode. The format for these GPS positions is very basic and prone to bit errors when any level of noise is present. The <u>D-PRS™ Specification</u> section describes this format in detail.

The initial iteration of D-PRS™ was to simply convert these GPS position reports to APRS position packets in TNC-2 format. This required interpretation of the GPS NMEA strings being passed unchanged from the remote radio and an "identification" line containing the remote radio's callsign and ID, and the text of the radio's GPS message (message C1 on many radios). The GPS NMEA strings all carry a fundamental XOR checksum which can be used for error checking but the identification line did not and it became apparent that this could cause issues unless a work-around was discovered. Simply using a character count was not sufficient for single bit errors. Also, it was discovered that the radios transmit the GPS/identification sequence continuously while the remote station is transmitting. This would be unacceptable if someone would try to gate these reports to the APRS analog channel so an acceptable work-around would need to be defined to limit these reports.

I used javAPRSIGate (an adjunct to javAPRSSrvr) as the foundation for D-PRS™. The modular architecture of javAPRSIGate and javAPRSSrvr allowed me to concentrate solely on the D-STAR® to APRS® conversion while javAPRSIGate and javAPRSSrvr handle all of the IGate functions. Because of this, this paper also concentrates solely on the D-PRS™ functionality leaving the APRS-IS IGate description to another paper. The module described in the rest of this paper is called DSTARInterface.

The only GPS strings of interest to forming an APRS position packet are $GPRMC and $GPGGA strings. Both of these strings provide position while $GPRMC provides speed and direction, and $GPGGA provides altitude. The Icom radios can be configured to only pass these strings. Therefore, the remote radios should be configured to only pass these strings.

To overcome the lack of error detection in the low speed data stream, I created a simple web page using JavaScript which calculates a simple XOR checksum for the identification line. The end-user goes to http://www.aprs-is.net/dprscalc.htm and inserts their own callsign, ID, chooses the symbol they want for their radio, and inserts any text they want in their GPS message. The page then automatically calculates the proper checksum for them to append to their GPS message. Users of radio programming software can simply copy/paste to their radio's GPS message. This work-around required limiting the character set to the least common denominator (most limited radio). Using the GPSxyz format for the symbol choice kept everything to upper case letters or numbers and the page limits what can be entered for a message.

I inserted a timer on the receive algorithm which basically says "If this is a new station, create and gate an APRS® position. If not and is less than 10 seconds from the last position information from this

station, ignore and reset the timer." This is the work-around for continuous position transmission while a person as the PTT button depressed.

The local radio which is connected to the computer running javAPRSSrvr is placed in data mode, not GPS mode. GPS mode blocks the received information from being passed to the radio's serial port. Because the local radio is in data mode, bidirectional communication is possible. Initially, DSTARInterface simply appended an XOR checksum to the end of a gated TNC-2 line. However, when Icom introduced the IC-2820, they also introduced a new mode called GPS-A mode. This mode sends an APRS position packet in TNC-2 format with a 16 bit CCITT-CRC prepended to the APRS line. DSTARInterface now uses that method for sending and receiving APRS® TNC-2 format packets via the serial data port. DSTARInterface removes the prepended CRC when gating to APRS-IS. Note that even the ID-1 can be used with D-PRS™ when used with an external PC running D-PRS™ software.

A need was also determined for a implementing a stand-alone version of javAPRSSrvr with DSTARInterface so a user could have an APRS® client like UI-View see all of the remote stations and potentially be able to communicate to other APRS® clients via the low speed D-STAR® serial port. This was initially done using Java and a limited javAPRSSrvr. However, for ease of implementation, this is now done using D-PRS Interface. D-PRS Interface is a Windows .NET application with minimal configuration parameters allowing for relatively easy installation on a Windows XP or later platform.

Finally, D-PRS™ is now being implemented on most D-STAR® gateway computers. Another software called DStarMonitor presents all of the low speed data seen at the gateway to TCP ports. javAPRSSrvr then connects to these ports providing a unidirectional (receive-only) IGate for all stations using any of the repeaters connected via a controller to the gateway computer.

D-PRS™ Specification

The Icom VHF and UHF radios transmit GPS information (when properly configured and attached to a GPS) via the general-use data portion of the digital voice (DV) signal. This has about 900 bps capacity and is non-interfering to the voice portion of the signal. When the radio is set to see only $GPRMC and $GPGGA strings, it sends the following sequence when transmitting:

$GPRMC,…*CS
$GPGGA,…*CS
MYCALL I,GPS MESSAGE

The last line (I will call it the identification line) is always 29 characters long with a comma in the ninth position. All lines are terminated by at least a carriage return although all carriage returns and line feeds should be discarded. This sequence is sent once when part of a timed beacon. This sequence is repeated continuously when sent as part of a voice transmission. It is up to the receiving station (software) to check the XOR checksum on the GPS sentences shown here as *CS. It is also up to the receiving station

(software) to verify that the GPS sentences are valid. This means that the receiving station (software) should not show a position for a GPS sentence that indicates the GPS is not in lock.

To provide error detection for the identification line, the GPS message must contain a valid checksum. This checksum is computed using a straight XOR like the GPS NMEA sentences. The checksum is computed from the first character on the line up to but not including the asterisk preceding the checksum. If the checksum is less than 16, it will consist of a single character.

To provide the ability to choose any symbol from the APRS® symbol set including overlays, the xyz portion of the GPSxyz destination address format is used. The first 4 positions in the GPS message are reserved for the xyz portion of the GPSxyz definition and a space (the fourth character is always a space). The symbol table can be found in the APRS specification. To aid in calculating the GPS message, I established http://www.aprs-is.net/dprscalc.htm

The GPS sequence is converted to a TNC-2 format APRS line terminated by a carriage return and line feed. The following is a converted APRS line:

KE5C>APDPRS,DSTAR*:!3104.33N/09723.58W>220/001 IC-91AD/A=000518

The source callsign (KE5C) is the callsign of the radio in GPS mode. If the user has defined a non-space 8th character (ID), this character will be appended with a hyphen to the callsign emulating an AX.25 SSID. If the callsign is seven characters then the ID will be added without a hyphen. For instance, if the KE5C had defined an ID of 'A' then the source callsign would have been KE5C-A. For seven character callsigns, the ID is added without the hyphen.

The destination is always APDPRS. This indicates that this is a D-PRS™ translation.

The path of DSTAR* indicates that this "packet" originated in the D-STAR® network.

The data type '!' is the data type for a standard position packet with no messaging capability.

The symbol codes are derived from the first 3 characters of the GPS message using the table in the APRS 1.01 specification.

The dir/spd extension is derived from the $GPRMC sentence. If a valid $GPRMC sentence is not received, this extension is not added.

The comment area always starts with a space if there is any text in the message or an altitude is appended. This prevents misinterpretations by certain software. Note that the checksum (asterisk followed by one or two characters) is stripped from the comment.

The /A= altitude extension is only appended if a valid $GPGGA sentence is received.

With the IC-2820, Icom introduced a new GPS mode called GPS-A. Like the standard GPS mode, this mode uses the general-purpose portion of the DV signal to send position information. However, Icom greatly simplified the translation that must occur by basing GPS-A on standard APRS® formats. The APRS® line is prepended with a CRC and only has a carriage return for termination.

$$CRCCE3E,AE5PL-T>API282,DSTAR*:!3302.39N/09644.66W>/

The CRC will always be 4 characters terminated by a comma. The CRC is calculated starting with the first character of the callsign and terminating with the carriage return (including the carriage return). In a similar fashion to the D-PRS™ callsign translation described above, Icom also appends the ID to the callsign with a hyphen for callsigns less than seven characters long. For seven character callsigns, the ID is added without the hyphen.

The destination of API282 and the path of DSTAR* is manually programmed into the radio by the user setting the UNPROTO setting in the radio. This is a requirement for proper recognition by APRS clients.

When gating to APRS-IS, this line is passed (after CRC checking) with all after the comma and terminated with a carriage return and line feed. The CRC field is not passed.

The D-PRS™ gateway can use this format to send APRS[®] packets to the D-STAR[®] low speed serial channel. This provides relatively reliable reception with the higher quality CRC than was achieved with a simple XOR checksum.

The GPS-A format is also repeated continuously when the PTT button is depressed. Therefore, the same timer used above to prevent an uncontrolled stream of position reports is used when receiving GPS-A packets as well.

The CRC calculation is (Java):

```java
public final static int calcCCITTCRC(byte[] buffer, int startpos, int length)
{
        int icomcrc = 0xffff;
        for (int j = startpos; j < startpos+length; j++)
        {
                int ch = buffer[j] & 0xff;
                for (int i = 0; i < 8; i++)
                {
                        boolean xorflag = (((icomcrc ^ ch) & 0x01) == 0x01);
                        icomcrc >>>= 1;
                        if (xorflag)
                                icomcrc ^= 0x8408;
                        ch >>>= 1;
                }
        }
        return (~icomcrc) & 0xffff;
}
```

Conclusion

Anytime a bridge is designed between two disparate protocols, limitations and features of both protocols must be fully considered. In the case of D-PRS™, the follow factors were considered:

- No error detection or correction on D-STAR® low speed serial data stream.
- D-STAR® low speed data stream always carried with voice data stream.
- Icom GPS mode transmits unaltered NMEA sentences.
- Icom GPS mode transmits a 20 character message with the callsign and ID.
- Icom radios continuously send GPS information while voice is being transmitted.
- Icom GPS mode radios do not have a method of defining an APRS® symbol.
- Icom GPS-A mode uses APRS® format lines with the CCITT-CRC added to the beginning of the line.
- There is no direct indication of transmitting station identification on the Icom serial port.
- APRS® frequencies can be at or near capacity requiring limiting packet gating to RF.
- APRS® networks rely on duplicate checking to prevent loops and excessive retransmissions.
- Icom GPS modes are manufacturer specific and not part of the D-STAR® specification.

By using a modular development method, D-PRS™ was designed without regard to APRS-IS requirements such as the q algorithm. Total focus was on the definition of a reliable, constant bridge between the Icom GPS modes and APRS®. D-PRS™ has been implemented in software with javAPRSSrvr and the D-PRS™-specific version of javAPRSSrvr, D-PRS Interface. D-PRS™ has been implemented in hardware with the µSmartDigi manufactured by Rich Painter ABØVO. As others look to implementing D-PRS™, it is imperative that the format of the translation be kept constant to prevent issues seen on APRS-IS when different APRS® clients converted Mic-E packets using many different formats.

References

Japan Amateur Radio League, Inc. (2005). *Translated D-STAR System Specification.* Retrieved 8 23, 2005, from JARL English: http://www.jarl.com/d-star/shogen.pdf

TAPR APRS Working Group. (2000, 31 8). *APRS Protocol Specification Version 1.0.* Retrieved from TAPR: http://www.tapr.org/aprs_working_group.html

---

[i] D-STAR® is a registered trademark of the Icom Incorporated Corporation Japan
[ii] APRS® is a registered trademark of Robert Bruninga WB4APR
[iii] D-PRS™ is a trademark of Icom Incorporated Corporation Japan