

AUTOMATIC POSITION REPORTING SYSTEM



APRS PROTOCOL REFERENCE

Protocol Version 1.0

<i>Authors</i>	The APRS Working Group
<i>Document Version</i>	Draft Version 1.0.1g
<i>Filename</i>	aprs101g.pdf
<i>Date of Issue</i>	3 December 1999
<i>Copyright</i>	©1999 APRS Working Group All rights reserved
<i>Technical Editor</i>	Ian Wade, G3NRW

APRS Protocol Reference
Protocol Version 1.0

by the APRS Working Group

Edited by Ian Wade

Published by
Tuscon Amateur Packet Radio Corp
8987-309 East Tanque Verde Road, #337
Tuscon
AZ 85749-9399
United States of America

<http://www.tapr.org>

ISBN 0-9644707-6-4

TAPR Publication Number: 99-4

Copyright ©1999 APRS Working Group
All rights reserved

APRS[®] is a registered trademark of Bob Bruninga.

WinAPRS[™], MacAPRS[™], X-APRS[™], PalmAPRS[™] and APRS/CE[™]
are trademarks using the APRS[®] name, licensed from Bob Bruninga.

This document may be copied for non-commercial purposes only, and must
include the above copyright statement and trademark statements in full.

APRS PROTOCOL REFERENCE

TABLE OF CONTENTS

PREAMBLE 1

- This Document is a DRAFT 1**
- APRS Working Group 1
- Acknowledgements 2
- Document Version Number 2
- Document Release History 3
- Document Conventions 3

AUTHORS' FOREWORD..... 4

THE STRUCTURE OF THIS SPECIFICATION 5

1 INTRODUCTION TO APRS 7

- What is APRS? 7
- APRS Features 8

2 THE APRS DESIGN PHILOSOPHY..... 9

- Net Cycle Time..... 9
- Packet Timing 10
- Generic Digipeating..... 11

3 APRS AND AX.25 12

- Protocols 12
- The AX.25 Frame..... 12

4 APRS DATA IN THE AX.25 DESTINATION AND SOURCE ADDRESS FIELDS..... 13

- The AX.25 Destination Address Field..... 13
- Generic APRS Callsigns 13
- Generic APRS Callsigns with Symbols..... 14
- APRS Software Version Number..... 14
- Mic-E Encoded Data 15
- Alternate Net Callsign..... 15
- The AX.25 Source Address Field 15

5 APRS DATA IN THE AX.25 INFORMATION FIELD 16

- Generic Data Format..... 16
- APRS Data Type Identifier 16
- APRS Data and Data Extension 17
- Comment Field..... 19



6 TIME AND POSITION FORMATS.....	20
Time Formats.....	20
Use of Timestamps.....	21
Latitude Format.....	21
Longitude Format.....	21
Position Ambiguity.....	22
Position Coordinates.....	22
Default Null Position.....	22
Maidenhead Locator (Grid Square).....	23
NMEA Data.....	23
Altitude.....	23
7 APRS DATA EXTENSIONS.....	24
Course and Speed.....	24
Wind Direction and Wind Speed.....	24
Power, Effective Antenna Height/Gain/ Directivity.....	25
Range Circle Plot.....	26
Pre-Calculated Radio Range.....	26
Omni-DF Signal Strength.....	26
Bearing and Number/Range/ Quality.....	27
Area Object Descriptor.....	27
8 POSITION AND DF REPORT DATA FORMATS.....	28
9 COMPRESSED POSITION REPORT DATA FORMATS.....	30
The Advantages of Data Compression.....	30
Compressed Data Format.....	31
Symbol.....	31
Lat/Long Encoding.....	32
Lat/Long Decoding.....	32
Course/Speed and Pre-Calculated Radio Range.....	32
The Compression Type (T) Byte.....	33
Altitude.....	34
New Trackers.....	34
Old Trackers.....	35
10 MIC-E DATA FORMAT.....	36
Mic-E Data Format.....	36
Mic-E Data Payload.....	36
Mic-E Destination Address Field.....	37
Destination Address Field Encoding.....	37
Destination Address SSID Field.....	39
Mic-E Information Field.....	39
Information Field Data.....	39
Mic-E Telemetry Data.....	40
Mic-E Status Text.....	41
Maidenhead Locator in Comment Field.....	41
Altitude in the Comment Field.....	42



Mic-E Data in Non-APRS Networks.....42
 Example of Mic-E Packet Decoding43

11 OBJECT AND ITEM REPORTS.....45

 Objects.....45
 Items45
 Replacement Object / Item.....46
 Killed Object.....46
 Area Objects46
 Signpost Objects48
 Obsolete Object Format48

12 WEATHER REPORTS49

 Data Type Identifiers49
 Time Stamps.....49
 Positionless Weather Reports49
 Weather Data.....50
 APRS Software Type50
 Weather Unit Type51
 Weather Reports with Timestamp and Position.....51
 Weather Reports without Position Information52
 Storm Data.....53

13 TELEMETRY DATA54

 Telemetry Report Format54
 On-Air Definition of Telemetry Parameters.....54
 Parameter Name Beacon55
 Unit/Label Beacon55
 Equation Coefficients Beacon56
 Bit Sense/ Project Name Beacon56

14 MESSAGES, BULLETINS AND ANNOUNCEMENTS57

 Messages.....57
 Message Acknowledgements.....58
 Message Groups.....58
 General Bulletins.....58
 Announcements59
 Group Bulletins.....59
 National Weather Service Bulletins60
 Obsolete Bulletin and Announcement Format60

15 STATION CAPABILITIES, QUERIES AND RESPONSES61

 Station Capabilities.....61
 Queries and Responses61
 General Queries.....62
 Directed Station Queries63



16 STATUS REPORTS	64
Status Report with Beam Heading and Power.....	64
Status Report with Maidenhead Grid Locator	65
Transmitting Status Reports.....	65
17 NETWORK TUNNELING AND THIRD-PARTY DIGIPEATING	66
Third-Party Networks.....	66
Source Path Header.....	66
Third-Party Header.....	68
Action on Receiving a Third-Party packet.....	69
An Example of Sending a Message through the Internet.....	69
18 USER-DEFINED DATA FORMAT	70
19 OTHER PACKETS	72
Invalid Data or Test Data Packets	72
All Other Packets	72
20 APRS SYMBOLS.....	73
Symbol Table Identifier and Symbol Code.....	73
Symbol Overlays	73
Symbols with Stand-Alone GPS Trackers	74
APPENDIX 1: APRS DATA FORMATS	75
APPENDIX 2: THE APRS SYMBOL TABLES	83
APPENDIX 3: ASCII CODE TABLE.....	86
APPENDIX 4: GLOSSARY	89
APPENDIX 5: REFERENCES.....	93



PREAMBLE

**This Document is a
DRAFT**

THIS DOCUMENT IS STILL A DRAFT.

This version of the document is a draft for comment on the *current* protocol *only* (i.e. APRS Protocol Version 1.0), and not for further distribution until approved by the APRS Working Group. Enhancements and improvements to the protocol will be made later, after this draft has gone final.

This document is a complete rewrite of the original draft, and although the APRS Working Group have checked it thoroughly, it is probable there are still some errors, omissions or inconsistencies.

Please address your comments on this draft to the *aprsspec* mail list. To join the list, start at <http://www.tapr.org> and follow the path Special Interest Groups > APRS Specification > Join APRS Spec Discussion List.

To help the Working Group categorize the comments into groups, please prefix the subject line with one of the following:

ERROR:

OMISSION:

INCONSISTENCY:

UNCLEAR:

SUGGESTION:

GENERAL:

Our aim is to make this specification as accurate, complete and understandable as possible, so your feedback is very important.

Please send your comments to the *aprsspec* list by no later than midnight your time on Sunday 19 December 1999. Thank you.

APRS Working Group

The APRS Working Group is an unincorporated association whose members undertake to further the use and enhance the value of the APRS protocols by (a) publishing and maintaining a formal APRS Protocol Specification; (b) publishing validation tests and other tools to enable compliance with the Specification; (c) supporting an APRS Certification program; and (d) generally working to improve the capabilities of APRS within the amateur radio community.

Although the Working Group may receive support from TAPR and other



organizations, it is an independent body and is not affiliated with any organization. The Group has no budget, collects no dues, and owns no assets.

The current members of the APRS Working Group are:

John Ackermann, N8UR	Administrative Chair
Bob Bruninga, WB4APR	Technical Chair, founder of APRS
Steve Dimse, K4HG	Author of javAPRS/APRServe/XMLserve
Brent Hildebrand, KH2Z	Author of APRS+SA
Stan Horzepa, WA1LOU	Secretary
Greg Jones, WD5IVD	Representing TAPR
Mike Musick, N0QBF	Author of pocketAPRS
Keith Sproul, WU2Z	Co-Author of WinAPRS/MacAPRS/X-APRS
Mark Sproul, KB2ICI	Co-Author of WinAPRS/MacAPRS/X-APRS

Acknowledgements

This document is the result of contributions from many people. It includes much of the material produced by individual members of the Working Group.

In addition, the paper on the Mic-E data format by Alan Crosswell, N2YGK, and Ron Parsons, W5RKN was a useful starting point for explaining the complications of this format.

Document Version Number

Except for the very first public draft release of the APRS Protocol Reference, the document version number is a 3-part number “P.p.D” (for an approved document release) or a 4-part number “P.p.Dd” (for a draft release):

<i>Document Version Number</i>			
<i>APRS Protocol Version</i>		<i>Document Release</i>	<i>Draft</i>
<i>Major Release</i>	<i>Minor Release</i>		
P.	p.	D	d

Thus, for example:

- Document version number “1.2.3” refers to document release 3 covering APRS Protocol Version 1.2.
- Document version number “1.2.3c” is draft “c” of that document.



**Document
Release History**

<i>Date</i>	<i>Doc Version</i>	<i>Status / Major Changes</i>
10 Oct 1999	1.0 (Draft)	Protocol Version 1.0. First public draft release.
3 Dec 1999	1.0.1g	Protocol Version 1.0. Second public draft release. Much extended, incorporating packet format layouts, APRS symbol tables, compressed data format, Mic-E format, telemetry format, many more packet examples.

**Document
Conventions**

This document uses the following conventions:

- `Courier font` ASCII characters in APRS data.
- `␣` ASCII space character.
- `...` (ellipsis) zero or more characters.
- `/s` Symbol from Primary Symbol Table.
- `\s` Symbol from Alternate Symbol Table.
- `0x` hexadecimal (e.g. `0x1d`).
- All callsigns are assumed to have SSID –0 unless otherwise specified.
- **Yellow marker** (appears as light gray background in hard copy). Marks text of interest — especially useful for highlighting single literal ASCII characters (e.g. `"`) where they appear in APRS data.
- Shaded areas in packet format diagrams are optional fields.

AUTHORS' FOREWORD

This reference document describes what is known as *APRS Protocol Version 1.0*, and is essentially a description of how APRS operates today.

It is intended primarily for the programmer who wishes to develop APRS-compliant applications, but will also be of interest to the ordinary user who wants to know more about what goes on “under the hood”.

It is *not* intended, however, to be a dry-as-dust, pedantic, RFC-style programming specification, to be read and understood only by the Mr Spocks of this world. We have included many items of general information which, although strictly not part of the formal protocol description, provide a useful background on how APRS is actually used on the air, and how it is implemented in APRS software. We hope this will put APRS into perspective, will make the document more readable, and will not offend the purists too much.

It is important to realize how APRS originated, and to understand the design philosophy behind it. In particular, we feel strongly that APRS is, and should remain, a light-weight tactical system — almost anyone should be able to use it in temporary situations (such as emergencies or mobile work or weather watching) with the minimum of training and equipment.

This document is the result of inputs from many people, and collated and massaged by the APRS Working Group. Our sincere thanks go to everyone who has contributed in putting it together and getting it on to the street. If you discover any errors or omissions or misleading statements, please let us know — the best way to do this is via the TAPR *aprsspec* mailing list at www.tapr.org.

Finally, users throughout the world are continually coming up with new ideas and suggestions for extending and improving APRS. We welcome them. Again, the best way to discuss these is via the *aprsspec* list.

The APRS Working Group

December 1999



THE STRUCTURE OF THIS SPECIFICATION

This specification describes the overall requirements for developing software that complies with APRS Protocol Version 1.0. The information flow starts with the standard AX.25 UI-frame, and progresses downwards into more and more detail as the use of each field in the frame is explored.

A key feature of the specification is the inclusion of dozens of detailed examples of typical APRS packets and related math computations.

Here is an outline of the chapters:

Introduction to APRS — A brief background to APRS and a summary of its main features.

The APRS Design Philosophy — The fundamentals of APRS, highlighting its use as a real-time tactical communications tool, the timing of APRS transmissions and the use of generic digipeating.

APRS and AX.25 — A brief refresher on the structure of the AX.25 UI-frame, with particular reference to the special ways in which APRS uses the Destination and Source Address fields and the Information field.

APRS Data in the AX.25 Destination and Source Address Fields — Details of generic APRS callsigns and callsigns that specify display symbols and APRS software version numbers. Also a summary of how Mic-E encoded data is stored in the Destination Address field, and how the Source Address SSID can specify a display icon.

APRS Data in the AX.25 Information Field — Details of the principal constituents of APRS data that are stored in the Information field. Contains the APRS Data Type Identifiers table, and a summary of all the different types of data that the Information field can hold.

Time and Position Formats — Information on formats for timestamps, latitude, longitude, position ambiguity, Maidenhead locators, NMEA data and altitude.

APRS Data Extensions — Details of optional data extensions for station course/speed, wind speed/direction, power/height/gain, pre-calculated radio range, DF signal strength and Area Object descriptor.

Position and DF Report Data Formats — Full details of these report formats.

Compressed Position Report Data Formats — Full details of how station position and APRS data extensions are compressed into very short packets.

Mic-E Data Format — Mic-E encoding of station lat/long position, altitude, course, speed, Mic-E message code, telemetry data and APRS digipeater path into the AX.25 Destination Address and Information fields.

Object and Item Reports — Full information on how to set up APRS Objects and Items, and details of the encoding of Area Objects (circles, lines, ellipses etc).

Weather Reports — Full format details for weather reports from stand-alone (positionless) weather stations and for reports containing position information. Also details of storm data format.

Telemetry Data — A description of the MIM/KPC-3+ telemetry data format, with supporting information on how to tailor the interpretation of the raw data to individual circumstances.

Messages, Bulletins and Announcements — Full format information.

Station Capabilities, Queries and Responses — Details of the ten different types of query and expected responses.

Status Reports — The format of general status messages, plus the special cases of using a status report to contain meteor scatter beam heading/power and Maidenhead locator.

Network Tunneling — The use of the Source Path Header to allow tunneling of APRS packets through third-party networks that do not understand AX.25 addresses, and the use of the third-party Data Type Identifier.

User-Defined Data Format — APRS allows users to define their own data formats for special purposes. This chapter describes how to do this.

Other Packets — A general statement on how APRS is to handle any other packet types that are not covered by this specification.

APRS Symbols — How to specify APRS symbols and symbol overlays, in position reports and in generic GPS destination callsigns.

APRS Data Formats — An appendix containing all the APRS data formats collected together for easy reference.

The APRS Symbol Tables — A complete listing of all the symbols in the Primary and Alternate Symbol Tables.

ASCII Code Table — The full ASCII code, including decimal and hex codes for each character (the decimal code is needed for compressed lat/long and altitude computations), together with the hex codes for bit-shifted ASCII characters in AX.25 addresses (useful for Mic-E decoding and general on-air packet monitoring).

Glossary — A handy one-stop reference for the many APRS-specific terms used in this specification.

References — Pointers to other documents that are relevant to this specification.



1 INTRODUCTION TO APRS

What is APRS?

APRS is short for *Automatic Position Reporting System*, which was introduced by Bob Bruninga, WB4APR, at the 1992 TAPR/ARRL Digital Communications Conference.

Fundamentally, APRS is a packet communications protocol for disseminating live data to everyone on a network in real time. Its most visual feature is the combination of packet radio with the Global Positioning System (GPS) satellite network, enabling radio amateurs to automatically display the positions of radio stations and other objects on maps on a PC. Other features not directly related to position reporting are supported, such as weather station reporting, direction finding and messaging.

APRS is different from regular packet in several ways:

- It provides maps and other data displays, for vehicle/personnel location and weather reporting in real time.
- It performs all communications using a one-to-many protocol, so that everyone is updated immediately.
- It uses generic digipeating, with well-known callsign aliases, so that prior knowledge of network topology is not required.
- It supports intelligent digipeating, with callsign substitution to reduce network flooding.
- Using AX.25 UI-frames, it supports two-way messaging and distribution of bulletins and announcements, leading to fast dissemination of text information.
- It supports communications with the Kenwood TH-D7 and TM-D700 radios, which have built-in TNC and APRS firmware.

Conventional packet radio is really only useful for passing bulk message traffic from point to point, and has traditionally been difficult to apply to real-time events where information has a very short lifetime. APRS turns packet radio into a real-time tactical communications and display system for emergencies and public service applications.

APRS provides universal connectivity to all stations, but avoids the complexity, time delays and limitations of a connected network. It permits any number of stations to exchange data just like voice users would on a voice net. Any station that has information to contribute simply sends it, and all stations receive it and log it.

APRS recognizes that one of the greatest real-time needs at any special event or emergency is the tracking of key assets. Where is the marathon leader? Where are the emergency vehicles? What's the weather at various points in the county? Where are the power lines down? Where is the head of the

parade? Where is the mobile ATV camera? Where is the storm?

To address these questions, APRS provides a fully featured automatic vehicle location and status reporting system. It can be used over any two-way radio system including amateur radio, marine band, and cellular phone. There is even an international live APRS tracking network on the Internet.

APRS Features

APRS runs on most platforms, including DOS, Windows 3.x, Windows 95/98, MacOS, Linux and Palm. Most implementations on these platforms support the main features of APRS:

- **Maps** — APRS station positions can be plotted in real-time on maps, with coverage from a few hundred yards to worldwide. Stations reporting a course and speed are dead-reckoned to their present position. Overlay databases of the locations of APRS digipeaters, US National Weather Service sites and even amateur radio stores are available. It is possible to zoom in to any point on the globe.
- **Weather Station Reporting** — APRS supports the automatic display of remote weather station information on the screen.
- **DX Cluster Reporting** — APRS an ideal tool for the DX cluster user. Not only is it possible to see all DX spots on the map, but by operating in the monitor-only mode, the overall packet load on the DX cluster is reduced. This is a benefit to everyone on the channel.
- **Internet Access** — The Internet can be used transparently to cross-link local radio nets anywhere on the globe. It is possible to telnet into Internet APRS servers and see hundreds of stations from all over the world live. Everyone connected can feed their locally heard packets into the APRS server system and everyone everywhere can see them.
- **Messages** — Messages are two-way messages with acknowledgment. All incoming messages alert the user on arrival and are held on the message screen until killed.
- **Bulletins and Announcements** —Bulletins and announcements are addressed to everyone. Bulletins are sent a few times an hour for a few hours, and announcements less frequently but possibly over a few days.
- **Fixed Station Tracking** — In addition to automatically tracking mobile GPS/LORAN-equipped stations, APRS also tracks from manual reports or grid squares.
- **Objects** — Any user can place an APRS Object on his own map, and within seconds that object appears on all other station displays. This is particularly useful for tracking assets or people that are not equipped with trackers. Only one packet operator needs to know where things are (e.g. by monitoring voice traffic), and as he maintains the positions and movements of assets on his screen, all other stations running APRS will display the same information.



2 THE APRS DESIGN PHILOSOPHY

Net Cycle Time

It is important to note that APRS is primarily a *real-time, tactical* communications tool, to help the flow of information for things like special events, emergencies, Skywarn, the Emergency Operations Center and just plain in-the-field use under stress. But like the real world, for 99% of the time it is operating routinely, waiting for the unlikely serious event to happen.

Anything which is done to enhance APRS must not undermine its ability to operate in local areas under stress. Here are the details of that philosophy:

1. APRS uses the concept of a “net cycle time”. This is the time within which a user should be able to hear (at least once) all APRS stations within range, to obtain a more or less complete picture of APRS activity. The net cycle time will vary according to local conditions and with the number of digipeaters through which APRS data travels.
2. The objective is to have a net cycle time of 10 minutes for local use. This means that within 10 minutes of arrival on the scene, it is possible to capture the entire tactical picture.
3. All stations, even fixed stations, should beacon their position at the net cycle time rate. In a stress situation, stations are coming and going all the time. The position reports show not only where stations are without asking, but also that they are still active.
4. It is not reasonable to assume that all APRS users responding to a stress event understand the ramifications of APRS and the statistics of the channel — user settings cannot be relied on to avoid killing a stressed net. Thus, to try to anticipate when the channel is under stress, APRS automatically adjusts its net cycle time according to the number of digipeaters in the UNPROTO path:
 - Direct operation (no digipeaters): 10 minutes (probably an event).
 - Via one digipeater: 10 minutes (probably an event).
 - Via two digipeaters: 20 minutes.
 - Via three or more digipeaters: 30 minutes.
5. Since almost all home stations set their paths to three or more digipeaters, the default net cycle time for routine daily operation is 30 minutes. This should be a universal standard that everyone can bank on — if you routinely turn on your radio and APRS and do nothing else, then in 30 minutes you should have virtually the total picture of all APRS stations within range.
6. Since knowing where the digipeaters are located is fundamental to APRS



connectivity, digipeaters should use multiple beacon commands to transmit position reports at different rates over different paths; i.e. every 10 minutes for sending position reports locally, and every 30 minutes for sending them via three digipeaters (plus others rates and distances as needed).

7. If the net cycle time is too long, users will be tempted to send queries for APRS stations. This will increase the traffic on the channel unnecessarily. Thus the recommended extremes for net cycle time are 10 and 30 minutes — this gives network designers the fundamental assumptions for channel loading necessary for good engineering design.

Packet Timing

Since APRS packets are error-free, but are not guaranteed delivery, APRS transmits information redundantly. To assure rapid delivery of new or changing data, and to preserve channel capacity by reducing interference from old data, APRS should transmit new information more frequently than old information.

There are several algorithms in use to achieve this:

- **Decay Algorithm** — Transmit a new packet once and n seconds later. Double the value of n for each new transmission. When n reaches the net cycle time, continue at that rate. Other factors besides “doubling” may be appropriate, such as for new message lines.
- **Fixed Rate** — Transmit a new packet once and n seconds later. Transmit it x times and stop.
- **Message-on-Heard** — Transmit a *new* packet according to either algorithm above. If the packet is still valid, and has not been acknowledged, and the net cycle time has been reached, then the recipient is probably not available. However, if a packet is then subsequently heard from the recipient, try once again to transmit the packet.
- **Time-Out** — This term is used to describe a time period beyond which it is reasonable to assume that a station no longer exists or is off the air if no packets have been heard from it. A period of 2 hours is suggested as the nominal default timeout. This time-out is not used in any transmitting algorithms, but is useful in some programs to decide when to cease displaying stations as “active”. Note that on HF, signals come and go, so decisions about activity may need to be more flexible.



Generic Digipeating

The power of APRS in the field derives from the use of *generic* digipeating, in that packets are propagated without a priori knowledge of the network. There are six powerful techniques which have evolved since APRS was introduced in 1992:

1. **RELAY** — Every VHF APRS TNC is assumed to have an alias of RELAY, so that anyone can use it as a digipeater at any time.
2. **ECHO** — HF stations use the alias of ECHO as an alternative to RELAY. (However, bearing in mind the nature of HF propagation, this has the potential of causing interference over a wide area, and should only be used sparingly by mobile stations).
3. **WIDE** — Every high-site digipeater is assumed to have an alias of WIDE for longer distance communications.
4. **TRACE** — Every high-site digipeater that is using callsign substitution is assumed to have the alias of TRACE. These digipeaters self-identify packets they digipeat by inserting their own call in place of RELAY, WIDE or TRACE.
5. **WIDEn-N** — A digipeater that supports WIDEn-N digipeating will digipeat any WIDEn-N packet that is “new” and will subtract 1 from the SSID until the SSID reaches -0. The digipeater keeps a copy or a checksum of the packet and will not digipeat that packet again within (typically) 28 seconds. This considerably reduces the number of superfluous digipeats in areas with many digipeaters in radio range of each other.
6. **GATE** — This generic callsign is used by HF-to-VHF Gateway digipeaters. Any packet heard on HF via GATE will be digipeated locally on VHF. This permits local networks to keep an eye on the national and international picture.

3 APRS AND AX.25

Protocols At the link level, APRS uses the AX.25 protocol, as defined in *Amateur Packet-Radio Link-Layer Protocol* (see Appendix 5 for details), utilizing Unnumbered Information (UI) frames exclusively. This means that APRS runs in *connectionless* mode, whereby AX.25 frames are transmitted without expecting any response, and reception at the other end is not guaranteed.

At a higher level, APRS supports a messaging protocol that allows users to send short messages (one line of text) to nominated stations, and expects to receive acknowledgements from those stations.

The AX.25 Frame All APRS transmissions use AX.25 UI-frames, with 9 fields of data:

AX.25 UI-FRAME FORMAT								
Flag	Destination Address	Source Address	Digipeater Addresses (0-8)	Control Field (UI)	Protocol ID	INFORMATION FIELD	FCS	Flag
Bytes: 1	7	7	0-56	1	1	1-256	2	2

- **Flag** — The flag field at each end of the frame is the bit sequence 0x7f that separates each frame.
- **Destination Address** — This field can contain an APRS destination callsign or APRS data. APRS data is encoded to ensure that the field conforms to the standard AX.25 callsign format (i.e. 6 alphanumeric characters plus SSID). If the SSID is non-zero, it specifies a generic APRS digipeater path.
- **Source Address** — This field contains the callsign and SSID of the transmitting station. In some cases, if the SSID is non-zero, the SSID may specify an APRS display Symbol Code.
- **Digipeater Addresses** — From zero to 8 digipeater callsigns may be included in this field. (N.B. In the special case of a frame containing Mic-E data, these digipeater addresses may be overridden by a separately specified APRS digipeater path).
- **Control Field** — This field is set to 0x03 (UI-frame).
- **Protocol ID** — This field is set to 0xf0 (no layer 3 protocol).
- **Information Field** — This field contains more APRS data. The first character of this field is the APRS Data Type Identifier that specifies the nature of the data that follows.
- **Frame Check Sequence** — The FCS is a sequence of 16 bits used for checking the integrity of a received frame.



4 APRS DATA IN THE AX.25 DESTINATION AND SOURCE ADDRESS FIELDS

The AX.25 Destination Address Field

The AX.25 Destination Address field can contain five different types of APRS information:

- A generic APRS callsign.
- A generic APRS callsign with a symbol.
- An APRS software version number.
- Mic-E encoded data.
- An Alternate Net (ALTNET) callsign.

Generic APRS Callsigns

APRS uses the following generic beacon-style destination callsigns:

AIR* †	ALL*	AP*	BEACON	CQ*	GPS*	DF*
DGPS*	DRILL*	DX*	ID*	JAVA*	MAIL*	MICE*
QST*	QTH*	RTCM*	SKY*	SPACE*	SPC*	SYM*
TEST*	TLM*	WX*	ZIP* †			

The asterisk is a wildcard, allowing the callsign to be extended (up to a total of 6 alphanumeric characters). Thus, for example, WX1, WX12 and WX12CD are all valid APRS destination callsigns.

† The **AIR*** and **ZIP*** callsigns are being phased out, but are needed at present for backward compatibility.

All of these callsigns have an SSID of -0. Non-zero SSIDs are reserved for generic APRS digipeating (see Chapter 10: Mic-E Data Format).

These calls are copied by everyone. All APRS software must accept packets addressed to these calls.

Packets addressed to **SPCL** are intended for special events. APRS software can display such packets to the exclusion of all others, to minimize clutter on the screen from other stations not involved in the special event.

The call **DGPS** is used by differential GPS correction stations. Most software will not make use of packets using this call, other than to pass them on to an attached GPS unit — the **DGPS** call is required to enable this function.

Generic APRS Callsigns with Symbols

APRS uses several of the above-listed generic callsigns in a special way to specify not only a callsign but also a display symbol:

GPS _{xyz}	Raw GPS NMEA packets
GPSC _{nn}	Raw GPS NMEA packets
GPSE _{nn}	Raw GPS NMEA packets
SPC _{xyz}	Raw GPS NMEA packets (special event)
SYM _{xyz}	(to be decided)

The characters *xyz* and *nn* refer to entries in the APRS Symbol Tables (see Appendix 2). For example, from the Primary Symbol Table, a tracker could use the destination callsign **GPS**PR, indicating that it is transmitting raw GPS NMEA packets and its position is to be displayed with a “Recreational Vehicle” icon.

APRS Software Version Number

The AX.25 Destination Address field can contain the version number of the APRS software that is running at the station. Knowledge of the version number can be useful when debugging.

The following software version types are reserved (*xx* and *xxx* indicate a version number):

APC _{xxx}	APRS/CE, Windows CE
APE _{xxx}	PIC-Encoder
API _{xxx}	Icom radios (future)
APIC _{xx}	ICQ messaging
APK _{xxx}	Kenwood radios
APM _{xxx}	MacAPRS
APP _{xxx}	pocketAPRS
APR _{xxx}	APRSdos
APRS	older versions of APRSdos
APRSM	older versions of MacAPRS
APRSW	older versions of WinAPRS
APS _{xxx}	APRS+SA
APW _{xxx}	WinAPRS
APX _{xxx}	X-APRS
APY _{xxx}	Yaesu radios (future)
APZ _{xxx}	Experimental

This table will be added to by the APRS Working Group.

For example, a station using version 3.2.6 of MacAPRS could use the destination callsign **APM**326.

The Experimental destination is designated for *temporary* use only while a product is being developed, before a special APRS Software Version call is assigned to it.



Mic-E Encoded Data

Another alternative use of the AX.25 Destination Address field is to contain Mic-E encoded data. This data includes:

- The latitude of the station.
- A West/East Indicator and a Longitude Offset Indicator (used in longitude computations).
- A Message Code.
- The APRS digipeater path.

This data is used with associated data in the AX.25 Information field to provide a complete Position Report and other information about the station (see Chapter 10: Mic-E Data Format).

Alternate Net Callsign

Any other callsign not included in the specific generic list or the other categories mentioned above may be used in Alternate Nets (ALTNETs) by groups of individuals for special purposes. Thus they can use the APRS infrastructure for a variety of experiments without cluttering up the maps and lists of other APRS stations. Only stations using the same ALTNET callsign should see their data.

The AX.25 Source Address Field

The AX.25 Source Address field contains the callsign and SSID of the originating station. If the SSID is -0, APRS does not treat it in any special way.

If, however, the SSID is non-zero, APRS interprets it as a display icon. This is intended for use only with stand-alone trackers where there is no other method of specifying a display symbol or a destination address (e.g. MIM trackers or NMEA trackers).

SSID-Specified Icons in the AX.25 Source Address Field

SSID	Icon	SSID	Icon
-0	[no icon]	-8	Ship (power boat)
-1	Ambulance	-9	Car
-2	Bus	-10	Motorcycle
-3	Fire Truck	-11	Balloon
-4	Bike	-12	Jeep
-5	Yacht	-13	Recreational Vehicle
-6	Helicopter	-14	Truck
-7	Small Aircraft	-15	Van

Note: The icon specified by the SSID in the Source Address is ignored if there is a Symbol Code in the APRS data in the AX.25 Information field.

5 APRS DATA IN THE AX.25 INFORMATION FIELD

Generic Data Format

In general, the AX.25 Information field can contain some or all of the following information:

- APRS Data Type Identifier
- APRS Data
- APRS Data Extension
- Comment

Generic APRS Information Field			
Data Type ID	APRS Data	APRS Data Extension	Comment
Bytes: 1	n	7	n

APRS Data Type Identifier

Every APRS packet contains an APRS Data Type Identifier. This determines the format of the remainder of the data in the Information field, as follows:

APRS Data Type Identifiers

Ident	Data Type
0x1c	Current Mic-E Data (Rev 0 beta)
0x1d	Old Mic-E Data (Rev 0 beta)
!	Position without timestamp (no APRS messaging), or Ultimeter 2000 WX Station
"	<i>[Unused]</i>
#	Peet Bros U-II Weather Station
\$	Raw GPS data or Ultimeter 2000
%	Agrelo DFJr / MicroFinder
&	<i>[Reserved — Map Feature]</i>
'	Old Mic-E Data
(<i>[Unused]</i>
)	Item
*	Peet Bros U-II Weather Station
+	Shelter data with time
,	Invalid data or test data
-	Killed APRS Object
.	Space weather
/	Position with timestamp (no APRS messaging)
0-9	<i>[Do not use]</i>
:	Message
;	Object

Ident	Data Type
<	Station Capabilities
=	Position without timestamp (with APRS messaging)
>	Status
?	Query
@	Position with timestamp (with APRS messaging)
A-S	<i>[Do not use]</i>
T	Telemetry data
U-Z	<i>[Do not use]</i>
[<i>[Unused]</i>
\	<i>[Unused]</i>
]	<i>[Unused]</i>
^	<i>[Unused]</i>
_	Weather Report (without position)
`	Current Mic-E Data
a-z	<i>[Do not use]</i>
{	User-Defined APRS packet format
 	<i>[Do not use — TNC stream switch character]</i>
}	Third-party traffic
~	<i>[Do not use — TNC stream switch character]</i>



Note: There is one exception to the requirement for the Data Type Identifier to be the *first* character in the Information field — this is the *Position without Timestamp* (indicated by the **!** Data Type Identifier). The **!** character may occur *anywhere up to and including the 40th character position* in the Information field. This variability is required to support X1J TNC digipeaters which have a string of unmodifiable text at the beginning of the field.

APRS Data and Data Extension

There are 10 main types of APRS Data:

- Position
- Direction Finding
- Objects and Items
- Weather
- Telemetry
- Messages, Bulletins and Announcements
- Queries
- Responses
- Status
- Other

Some of this data may also have an APRS Data Extension that provides additional information.

The APRS Data and optional Data Extension follow the Data Type Identifier.

The table on the next page shows a complete list of all the different possible types of APRS Data and APRS Data Extension.

	<i>Possible APRS Data</i>	<i>Possible APRS Data Extension</i>
Position	Time (DHM or HMS) Lat/long coordinates Compressed lat/long/course/speed/radio range/altitude Symbol Table ID and Symbol Code Mic-E longitude, speed and course, telemetry or status Raw GPS NMEA sentence	Course and Speed Power, Effective Antenna Height/Gain/Directivity Pre-Calculated Radio Range Omni DF Signal Strength Storm Data (in Comment field)
Direction Finding	Time (DHM or HMS) Lat/long coordinates Compressed lat/long/course/speed/radio range/altitude Symbol Table ID and Symbol Code	Course and Speed Power, Effective Antenna Height/Gain/Directivity Pre-Calculated Radio Range Omni DF Signal Strength Bearing and Number/Range/Quality (in Comment field)
Objects and Items	Object name Item name Time (DHM or HMS) Lat/long coordinates Compressed lat/long/course/speed/radio range/altitude Symbol Table ID and Symbol Code	Course and Speed Power, Effective Antenna Height/Gain/Directivity Pre-Calculated Radio Range Omni DF Signal Strength Area Object Storm Data (in Comment field)
Weather	Time (MDHM) Lat/long coordinates Compressed lat/long/course/speed/radio range/altitude Symbol Table ID and Symbol Code Raw weather station data	Wind Direction and Wind Speed Storm Data (in Comment field)
Telemetry	Telemetry (non Mic-E)	
Messages, Bulletins and Announcements	Addressee Message Text Message Identifier Message Acknowledgement Bulletin ID, Announcement ID Group Bulletin ID	
Queries	Query Type Query Target Footprint Addressee (Directed Query)	
Responses	Position Object/Item Weather Status Message Digipeater Trace Stations Heard Heard Statistics Station Capabilities	Course and Speed Power, Effective Antenna Height/Gain/Directivity Pre-Calculated Radio Range Omni DF Signal Strength Area Object Wind Direction and Speed
Status	Time (DHM zulu) Status text Meteor Scatter Beam Heading/Power Maidenhead Locator (Grid Square) Altitude (Mic-E) E-mail message	
Other	Third-Party forwarding Invalid Data/Test Data	



Comment Field

In general, any APRS packet can contain a plain text comment (such as a beacon message) in the Information field, immediately following the APRS Data or APRS Data Extension.

There is no separator between the APRS data and the comment unless otherwise stated.

The comment may contain any printable ASCII characters (except **|** and **~**, which are reserved for TNC channel switching).

The maximum length of the comment field depends on the report — details are included in the description of each report.

As special cases, the Comment field can also contain:

- The bearing and number/range/quality parameters in a DF report.
- Storm data.

This information immediately follows the APRS Data Extension.

6 TIME AND POSITION FORMATS

Time Formats APRS timestamps are expressed in three different ways:

- Day/Hours/Minutes format
- Hours/Minutes/Seconds format
- Month/Day/Hours/Minutes format

In all three formats, the 24-hour clock is used.

Day/Hours/Minutes (DHM) format is a fixed 7-character field, consisting of a 6-digit *day/time* group followed by a single *time indicator* character (**z** or **l**). The *day/time* group consists of a two-digit day-of-the-month (01–31) and a four-digit time in hours and minutes.

Times can be expressed in *zulu* (UTC/GMT) or *local* time. For example:

092345**z** is 2345 hours *zulu* time on the 9th day of the month.
 092345**l** is 2345 hours *local* time on the 9th day of the month.

It is recommended that future APRS implementations only transmit zulu format on the air.

Note: The time in Status Reports may *only* be in zulu format.

Hours/Minutes/Seconds (HMS) format is a fixed 7-character field, consisting of a 6-digit time in hours, minutes and seconds, followed by the **h** time-indicator character. For example:

234517**h** is 23 hours 45 minutes and 17 seconds *zulu*.

Note: This format may *not* be used in Status Reports.

Month/Day/Hours/Minutes (MDHM) format is a fixed 8-character field, consisting of the month (01–12) and day-of-the-month (01–31), followed by the time in hours and minutes zulu. For example:

10092345 is 23 hours 45 minutes zulu on October 9th.

This format is only used in reports from stand-alone “positionless” weather stations (i.e. reports that do not contain station position information).



Use of Timestamps

When a station transmits a report *without* a timestamp, an APRS receiving station can make an internal record of the time it was received, if required. This record is the *receiving* station's notion of the time the report was created.

On the other hand, when a station transmits a report *with* a timestamp, that timestamp represents the *transmitting* station's notion of the time the report was created.

In other words, reports sent *without* a timestamp can be regarded as real-time, "current" reports (and the *receiving* station has to record the time they were received), whereas reports sent *with* a timestamp may or may not be real-time, and may possibly be (very) "old".

Four APRS Data Type Identifiers specify whether or not a report contains a timestamp, depending on whether the station has APRS messaging capability or not:

	No APRS Messaging	With APRS Messaging
(Current/real-time) Report without timestamp	!	=
(Old/non-real-time) Report with timestamp	/	@

Stations without APRS messaging capability are typically stand-alone trackers or digipeaters. Stations reporting without a timestamp are generally (but not necessarily) fixed stations.

Latitude Format

Latitude is expressed as a fixed 8-character field, in degrees and decimal minutes (to two decimal places), followed by the letter **N** for north or **S** for south.

Latitude degrees are in the range 00 to 90. Latitude minutes are expressed as whole minutes and hundredths of a minute, separated by a decimal point.

For example:

4903.50**N** is 49 degrees 3 minutes 30 seconds north.

In generic format examples, the latitude is shown as the 8-character string `ddmm.hhN` (i.e. degrees, minutes and hundredths of a minute north).

Longitude Format

Longitude is expressed as a fixed 9-character field, in degrees and decimal minutes (to two decimal places), followed by the letter **E** for east or **W** for west.

Longitude degrees are in the range 000 to 180. Longitude minutes are expressed as whole minutes and hundredths of a minute, separated by a

decimal point.

For example:

07201.75W is 72 degrees 1 minute 45 seconds west.

In generic format examples, the latitude is shown as the 9-character string `dddmm.hhW` (i.e. degrees, minutes and hundredths of a minute west).

Position Ambiguity

Where the exact position is not known, the `mm` and `hh` digits in the latitude and longitude may be progressively replaced by a (space) character as the amount of imprecision increases. For example:

07201.7 W represents longitude to nearest 1/10th of a minute.

07201. W represents longitude to nearest minute.

0720. W represents longitude to nearest 10 minutes.

072. W represents longitude to nearest degree.

Position Coordinates

Position coordinates are a combination of latitude and longitude, separated by a display Symbol Table Identifier, and followed by a Symbol Code. For example:

4903.50N/07201.75W-

The `/` character between latitude and longitude is the Symbol Table Identifier (in this case indicating use of the Primary Symbol Table), and the `-` character at the end is the Symbol Code from that table (in this case, indicating a “house” icon).

A description of display symbols is included in Chapter 20: APRS Symbols. The full Symbol Table listing is in Appendix 2.

Default Null Position

Where a station does not have *any* specific position information to transmit (for example, a Mic-E unit without a GPS receiver connected to it), the station must transmit a default null position in the location field.

The null position corresponds to 0° 0' 0" north, 0° 0' 0" west.

The null position should include the `\.` symbol (unknown/indeterminate position). For example, a Position Report for a station with unknown position will contain the coordinates `...0000.00N\00000.00W...`

Implementation Recommendation: As the default null position is in the Atlantic Ocean off the west coast of Africa, consideration should be given as to whether it is useful to display a station reporting this position.



Maidenhead Locator (Grid Square)

An alternative method of expressing a station's location is to provide a Maidenhead locator (grid square), together with a Symbol Table Identifier and Symbol Code, in a Status Report or in Mic-E status text.

For example:

```
IO91sx/-
```

indicates a "house" at grid square IO91sx.

Note: The locator must be the *first* word in the Status Report or status text.

NMEA Data

APRS recognizes raw ASCII data strings conforming to the NMEA 0183 Version 2.0 specification, originating from navigation equipment such as GPS and LORAN receivers. It is recommended that APRS interprets at least the following NMEA Received Sentence types:

GGA	Global Positioning System Fix Data
GLL	Geographic Position, Latitude/Longitude Data
RMC	Recommended Minimum Specific GPS/Transit Data
VTG	Velocity and Track Data
WPT	Way Point Location

Altitude

Altitude may be expressed in two ways:

- In the comment text.
- In Mic-E format.

Altitude in Comment Text — The comment may contain an altitude value, in the form **/A=**aaaaaa, where aaaaaa is the altitude in feet. For example: /A=001234. The altitude may appear anywhere in the comment.

Altitude in Mic-E format — The optional Mic-E status field can contain altitude data. See Chapter 10: Mic-E Data Format.

7 APRS DATA EXTENSIONS

A fixed-length 7-byte field may follow APRS position data. This field is an APRS Data Extension. The extension may be one of the following:

- CSE/SPD Course and Speed (this may be followed by a further 8 bytes containing DF bearing and Number/Range/Quality parameters)
- DIR/SPD Wind Direction and Wind Speed
- PHGphgd Station Power and Effective Antenna Height/Gain/Directivity
- RNGrrrr Pre-Calculated Radio Range
- DFSshgd DF Signal Strength and Effective Antenna Height/Gain
- Tyy/Cxx Area Object Descriptor

Course and Speed

The 7-byte CSE/SPD Data Extension can be used to represent the course and speed of a vehicle or APRS Object.

The course is expressed in degrees (001-360), clockwise from due north. The speed is expressed in knots. A slash / character separates the two.

For example:

088/036 represents a course 88 degrees, traveling at 36 knots.

If the course and speed are unknown or not relevant, they are set to 000/000.

Wind Direction and Wind Speed

The 7-byte DIR/SPD Data Extension can be used to represent the wind direction and sustained one-minute wind speed in a Weather Report.

The wind direction is expressed in degrees (001-360), clockwise from due north. The speed is expressed in knots. A slash / character separates the two.

For example:

220/004 represents a wind direction of 220 degrees and a speed of 4 knots.

If the direction and speed are unknown or not relevant, they are set to 000/000.



**Power,
Effective Antenna
Height/Gain/
Directivity**

The 7-byte **PHG** phgd Data Extension specifies the transmitter power, effective antenna height-above-average-terrain, antenna gain and antenna directivity. APRS uses this information to plot radio range circles around stations.

The 7 characters of this Data Extension are encoded as follows:

Characters 1–3: **PHG** (fixed)
 Character 4: p Power code
 Character 5: h Height code
 Character 6: g Antenna gain code
 Character 7: d Directivity code

The PHG codes are listed in the table below:

PHG Codes

phgd Code:	0	1	2	3	4	5	6	7	8	9	Units
Power	0	1	4	9	16	25	36	49	64	81	watts
Height	10	20	40	80	160	320	640	1280	2560	5120	feet
Gain	0	1	2	3	4	5	6	7	8	9	dB
Directivity	omni	45 NE	90 E	135 SE	180 S	225 SW	270 W	315 NW	360 N		deg

The height code represents the effective height of the antenna *above average local terrain*, not above ground or sea level — this is to provide a rough indication of the antenna's effectiveness in the local area .

The height code may in fact be any ASCII character 0–9 and above. This is so that larger heights for balloons, aircraft or satellites may be specified.

For example:

; is the height code for 10240 feet (approximately 1.9 miles).
; is the height code for 20480 feet (approximately 3.9 miles), and so on.

The Directivity code offsets the PHG circle by one third in the indicated direction. This means a front-to-back ratio of 2 to 1. Most often this is used to indicate a favored direction or a null, even if an omni antenna is at the site.

An example of the PHG Data Extension:

PHG5132 means a power of 25 watts,
 an antenna height of 20 feet above the average local terrain,
 an antenna gain of 3 dB,
 and maximum gain due east.

Range Circle Plot

On receipt, APRS uses the *p*, *h*, *g* and *d* codes to calculate the usable radio range, for plotting a range circle representing the local radio horizon around the station. The radio range is calculated as follows:

$$\text{POWER} = p^2$$

$$\text{Height-above-average-terrain (HAAT)} = 10 * 2^h$$

$$\text{GAIN} = 10^{(g/10)}$$

$$\text{RANGE} = \text{SQR} (2 * \text{HAAT} * \text{SQR} ((\text{POWER}/10) * (\text{GAIN}/2)))$$

Thus, for PHG5132:

$$\text{POWER} = 5^2 = 25 \text{ watts}$$

$$\text{HAAT} = 10 * 2^1 = 20 \text{ feet}$$

$$\text{GAIN} = 10^{(3/10)} = 1.995262$$

$$\text{RANGE} = \text{SQR} (2 * 20 * \text{SQR} ((25/10) * (1.995262/2)))$$

~ 8 miles

As the direction of maximum gain is due east, APRS will draw a range circle of radius 8 miles around the station, offset by 2.7 miles (i.e. one third of 8 miles) in an easterly direction.

Note: In the absence of any PHG data, stations are assumed to be running 10 watts to a 3dB omni antenna at 20 feet, resulting in a 4½-mile radius range circle, centered on the station.

Pre-Calculated Radio Range

The 7-byte **RNG**rrrr Data Extension allows users to transmit a pre-calculated omni-directional radio range, where rrrr is the range in miles (with leading zeros).

For example, RNG0050 indicates a radio range of 50 miles.

APRS can use this value to plot a range circle around the station.

Omni-DF Signal Strength

The 7-byte **DFS**shgd Data Extension lets APRS localize jammers by plotting the overlapping signal strength contours of all stations hearing the signal. This Omni-DF format replaces the PHG format to indicate DF signal strength, in that the transmitter power field is replaced with the relative signal strength (*s*) from 0 to 9.

For example, DFS2360 represents a weak signal (around strength S2) heard on an omni antenna with 6 dB gain at 80 feet.

A signal strength of zero (0) is particularly significant, because APRS uses these 0 signal reports to draw (usually black) circles where the jammer is *not*



heard. These black circles are extremely valuable since there will be a lot more reports from stations that do not hear the jammer than from those that do. This quickly eliminates a lot of territory.

Bearing and Number/Range/Quality

DF reports contain an 8-byte field `/BRG/NRQ` that follows the `CRS/SPD` Data Extension, specifying the course, speed, bearing and NRQ (Number/Range/Quality) value of the report. NRQ indicates the Number of hits, the approximate Range and the Quality of the report.

For example, in:

```
...088/036/270/729...      course = 88 degrees, speed = 36 knots,
                           bearing = 270 degrees, N = 7, R = 2, Q = 9
```

If N is 0, then it means nothing. Values from 1 to 8 give an indication of the number of hits per period relative to the length of the time period — thus a value of 8 means 100% of all samples possible got a hit. A value of 9 for N indicates to other users that the report is manual.

The N value is not processed, but is just another indicator from the automatic DF units.

The range limits the length of the line to the original map's scale of the sending station. The range is 2^R so, for $R=4$, would be 16 miles.

Q is a single digit in the range 0–9.

If the course and speed parameters are not appropriate, they should have the value 000/000.

Area Object Descriptor

The 7-byte `TYY/Cxx` Data Extension is an Area Object Descriptor. The T parameter specifies the type of object (square, circle, triangle, etc) and the /C parameter specifies its fill color.

Area Objects are described in Chapter 11: Object and Item Reports.

8 POSITION AND DF REPORT DATA FORMATS

Lat/long Position Reports and DF reports are contained in the Information field of an APRS AX.25 frame.

The following diagrams show the permissible formats of these reports, together with some examples. The gray areas indicate optional fields, and the shaded (yellow) characters are literal ASCII characters. In all cases there is a maximum of 43 characters after the Symbol Code.

Lat/Long Position Report Format							
! = / @	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Comment (max 43 chars)	
Bytes:	1	7	8	1	9	1	0-43
<p><u>Examples</u></p> <p>!4903.50N/07201.75W-Test 001234 no timestamp, no APRS messaging, with comment.</p> <p>!4903.50N/07201.75W-Test /A=001234 no timestamp, no APRS messaging, altitude = 1234 ft.</p> <p>!49...N/072...W- no timestamp, no APRS messaging, location to nearest degree.</p> <p>...!4903.50N/07201.75W# no timestamp, no APRS messaging, X1J digipeater.</p> <p>/092345z4903.50N/07201.75W>Test1234 with timestamp, no APRS messaging, zulu time, with comment.</p> <p>@092345/4903.50N/07201.75W>Test1234 with timestamp, with APRS messaging, local time, with comment.</p>							

Lat/Long Position Report Format — with Data Extension								
! = / @	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	Comment (max 36 chars)	
						Power/Height/Gain/Dir		
						Radio Range		
						DF Signal Strength		
Bytes:	1	7	8	1	9	1	7	0-36
<p><u>Examples</u></p> <p>@092345/4903.50N/07201.75W>088/036 with timestamp, with APRS messaging, local time, course/speed.</p> <p>=4903.50N/07201.75W#PHG5132 no timestamp, with APRS messaging, with PHG.</p> <p>@234517h4903.50N/07201.75W>PHG5132 with timestamp, APRS messaging, hours/mins/secs time, PHG.</p> <p>@092345z4903.50N/07201.75W>RNG0050 with timestamp, APRS messaging, zulu time, radio range.</p> <p>@234517h4903.50N/07201.75W>DFS2360 with timestamp, APRS messaging, hours/mins/secs time, DF.</p>								



DF Report Format									
! or = or / or @	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	BRG/NRQ	Comment (max 28 chars)	
						Power/Height/Gain/Dir			
						Radio Range			
						DF Signal Strength			
Bytes:	1	7	8	1	9	1	7	8	0-28
<p><u>Example</u></p> <p>=4903.50N/07201.75W>088/036/270/729 no timestamp, course/speed/ bearing/NRQ, with APRS messaging.</p> <p>@092345z4903.50N/07201.75W>088/036/270/729 with timestamp, course/speed/ bearing/NRQ, with APRS messaging.</p> <p>/092345z4903.50N/07201.75W>000/000/270/729 with timestamp, bearing/NRQ, no course/speed, no APRS messaging.</p>									

Raw NMEA Position Report Format	
NMEA Received Sentence	
\$... / ... / ... / ... / ... / ...
Bytes:	1 25-209

<p><u>Examples</u></p> <p>\$GPGGA,102705,5157.9762,N,00029.3256,W,1,04,2.0,75.7,M,47.6,M,,*62</p> <p>\$GPGLL,2554.459,N,08020.187,W,154027.281,A</p> <p>\$GPRMC,063909,A,3349.4302,N,11700.3721,W,43.022,89.3,291099,13.6,E*52</p> <p>\$GPVTG,318.7,T,,M,35.1,N,65.0,K*69</p>	
---	--

9 COMPRESSED POSITION REPORT DATA FORMATS

In compressed data format, the Information field contains the station's latitude and longitude, together with course and speed or pre-calculated radio range or altitude.

This information is compressed to minimize the length of the transmitted packet (and therefore improve its chances of being received correctly under less than ideal conditions).

The Information field also contains a display Symbol Code, and there may optionally be a plain text comment (uncompressed) as well.

The Advantages of Data Compression

Compressed data format may be used in place of the numeric lat/long coordinates already described, such as in the **!**, **/**, **@** and **=** formats.

Data compression has several important benefits:

- Fully backwards compatible with all existing formats.
- Fully supports any comment string.
- Speed is accurate to +/-1 mph up to about 40 mph and within 3% at 600 mph.
- Altitude in feet is accurate to +/- 0.4% from 1 foot to 3000 miles.
- Consistent one-algorithm processing of compressed latitude and longitude.
- Improved position to 1 foot worldwide.
- Pre-calculated radio range, compressed to one byte.
- Potential 50% compression of every position format on the air.
- Potential 400% reduction of raw GPS NMEA data length.
- Additional 7-byte reduction for NEMA GGA altitudes.
- Support for TNC compression at the NMEA source (from the GPS receiver).
- Digipeater compression of old NMEA trackers on the fly.
- Usage is optional in all cases.

The only minor disadvantages are that the course only resolves to +/- 2 degrees, and this format does not support PHG.



Compressed Data Format

Compressed data may be generated in several ways:

- by APRS software.
- pre-entered manually into a digipeater's beacon text.
- by a digipeater converting raw tracker NMEA packets to compressed.

[In future, there is the possibility that a Kantronics KPC-3 or other tracker TNC will be able to compress data directly from an attached GPS receiver].

In all cases the compressed format is a fixed 13-character field:

/YYYYXXXX\$csT

where / is the Symbol Table Identifier
 YYYY is the compressed latitude
 XXXX is the compressed longitude
 \$ is the Symbol Code
 cs is the compressed course/speed or compressed pre-calculated radio range or compressed altitude
 T is the compression type indicator

Compressed Position Data					
Sym Table ID	Compressed Lat YYYY	Compressed Long XXXX	Symbol Code	Compressed Course/Speed	Comp Type T
				Compressed Radio Range	
				Compressed Altitude	
Bytes: 1	4	4	1	2	1

Compressed format can be used in place of lat/long position format anywhere that ...ddmm.hhN/dddmm.hhW\$xxxxxxx... occurs.

All bytes except for the / and \$ are base-91 printable ASCII characters (!..f). These are converted to numeric values by subtracting 33 from the decimal ASCII character code. For example, # has an ASCII code of 35, and represents a numeric value of 2 (i.e. 35-33).

Symbol

The presence of the leading Symbol Table Identifier instead of a digit indicates that this is a compressed Position Report and not a normal lat/long report.

Lat/Long Encoding

The values of `YYYY` and `XXXX` are computed as follows:

`YYYY` is $380972 * (90 - \text{latitude})$ [base 91]
latitude is positive for north, negative for south, in degrees.

`XXXX` is $190486 * (180 + \text{longitude})$ [base 91]
longitude is positive for east, negative for west, in degrees.

For example, for a longitude of $72^\circ 45' 00''$ west (i.e. -72.75 degrees), the math is $190486 * (180 - 72.75) = 20429623$. Because this is to base 91, it is then necessary to progressively divide this value by reducing powers of 91, to obtain the numerical values of `x`:

$$\begin{aligned} 20429623 / 91^3 &= 27, \text{ remainder } 83206 \\ 83206 / 91^2 &= 10, \text{ remainder } 396 \\ 396 / 91 &= 4, \text{ remainder } 32 \end{aligned}$$

To obtain the corresponding ASCII characters, 33 is added to each of these values, yielding 60 (i.e. $27+33$), 43, 37 and 65. From the ASCII Code Table (in Appendix 3), this corresponds to `<+%A` for `XXXX`.

Lat/Long Decoding

To decode a compressed lat/long, the reverse process is needed. That is, if `YYYY` is represented as `y1,y2,y3,y4` and `XXXX` as `x1,x2,x3,x4`, then:

$$\text{Lat} = 90 - ((y1-33)*91^3 + (y2-33)*91^2 + (y3-33)*91 + y4-33) / 380972$$

$$\text{Long} = -180 + ((x1-33)*91^3 + (x2-33)*91^2 + (x3-33)*91 + x4-33) / 190486$$

For example, if the compressed value of the longitude is `<+%A` (as computed above), the calculation becomes:

$$\begin{aligned} \text{Long} &= -180 + (27*91^3 + 10*91^2 + 4*91 + 32) / 190486 \\ &= -180 + (20346417 + 82810 + 364 + 32) / 190486 \\ &= -180 + 107.25 \\ &= -72.75 \text{ degrees} \end{aligned}$$

Course/Speed and Pre-Calculated Radio Range

The two `cs` bytes following the Symbol Code character can contain either the compressed course and speed or the compressed pre-calculated radio range. These two bytes are in base 91 format.

In the special case of `c = [space]`, there is no course, speed or range data, in which case the `csT` bytes are ignored.

Course/Speed — If the ASCII code for `c` is in the range `!` to `z` inclusive — corresponding to numeric values in the range 0–90 decimal (i.e. after subtracting 33 from the ASCII code) — then `cs` represents a compressed course/speed value:



$$\text{Course} = c * 4$$

$$\text{Speed} = 1.08^s - 1$$

For example, if the *cs* characters are **7P**, the corresponding values of *c* and *s* (after subtracting 33 from the ASCII character code) are 22 and 47 respectively. Substituting these values in the above equations:

$$\text{Course} = 22 * 4 = 88 \text{ degrees}$$

$$\text{Speed} = 1.08^{47} - 1 = 36.2 \text{ knots}$$

Pre-Calculated Radio Range — If *c* = **{**, then *cs* represents a compressed pre-calculated radio range value:

$$\text{Range} = 2 * 1.08^s$$

For example, if the *cs* bytes are **{?**, the ASCII code for **?** is 63, so the value of *s* is 30 (i.e. 63-33). Thus:

$$\text{Range} = 2 * 1.08^{30}$$

$$\sim 20 \text{ miles}$$

So APRS will draw a circle of radius 20 miles around the station plot on the screen.

The Compression Type (T) Byte

The *T* byte follows the *cs* bytes. The *T* byte contains several bit fields showing the GPS fix status, the NMEA source of the position data and the origin of the compression.

The *T* byte is not meaningful if the *c* byte is (space).

Compression Type (T) Byte Format								
Bit:	7	6	5	4	3	2	1	0
	<i>Not used</i>	<i>Not used</i>	GPS Fix	NMEA Source		Compression Origin		
Value:	0	0	0 = old (last) 1 = current	0 0 = other 0 1 = GLL 1 0 = GGA 1 1 = RMC	0 0 0 = Compressed 0 0 1 = TNC BText 0 1 0 = Software (DOS/Mac/Win/+SA) 0 1 1 = [tbd] 1 0 0 = KPC3 1 0 1 = Pico 1 1 0 = Other tracker [tbd] 1 1 1 = Digipeater conversion			

For example, if the compressed position was derived from an RMC sentence, the fix is current, and the compression was performed by APRSdos software, then the value of *T* in binary is 0 0 1 1 010, which equates to 58 decimal.

Adding 33 to this value gives the ASCII code for the *T* byte (i.e. 91), which corresponds to the **[** character.

Thus, using data from all the earlier examples, if the RMC sentence contains (among other parameters) the following data:

Latitude = 49° 30' 00" north
 Longitude = 72° 45' 00" west
 Speed = 36.2 knots
 Course = 88°

and: the fix is current
 compression is performed by APRSdos software
 the display symbol is a “car”

then the complete 13-character compressed location field is transmitted as:

/	YYYY	XXXX	\$	csT
/	5L5L	<+%A	>	7P[

Altitude

If the T byte indicates that the raw data originates from a GGA sentence (i.e. bits 4 and 3 of the T byte are 10), then the sentence contains an altitude value, in feet. After compression, the compressed altitude data is placed in the cs bytes, such that:

$$\text{Altitude} = 1.002^{\text{cs}} \text{ feet}$$

For example, if the received cs bytes are **SJ**, the computation is as follows:

- Subtract 33 from the ASCII code for each character:
 $c = 83 - 33 = 50$
 $s = 93 - 33 = 60$
- Multiply c by 91 and add s to obtain cs:
 $cs = 50 * 91 + 60$
 $= 4610$
- Then Altitude = 1.002^{4610}
 $= 10003 \text{ feet}$

New Trackers

Tracker firmware may compress GPS data directly to APRS compressed format. They would use the **!** Data Type Indicator, showing that the position is real-time and that the tracker is not APRS-capable.

If the Position Report is not real-time, then the **/** Data Type Indicator can be used instead, so that the latest fix time may be included.



Old Trackers Some digipeaters have the ability to convert raw NMEA strings from existing trackers to compressed data format for further forwarding.

These digipeaters will compress the data if the tracker Destination Address is GPS. (**Note:** This is the 3-letter callsign GPS, not GPS*).

Trackers desiring for their packets to not be modified by the APRS network will use any other valid generic APRS callsign.

Compressed data is contained in the AX.25 Information field, in this format:

Compressed Lat/Long Position Report Format									
! or = or / or @	Time DHM / HMS	Sym Table ID	Comp Lat YYYY	Comp Long XXXX	Symbol Code	Compressed Course/Speed	Comp Type T	Comment (max 40 chars)	
						Compressed Radio Range			
						Compressed Altitude			
Bytes:	1	1	1	4	4	1	2	1	0-40
<p>Examples</p> <p>=/5L5L<+%A >_sTComment with APRS messaging. Note the _ space character following the > Symbol Code, indicating that there is no course/speed, radio range or altitude. The sT characters are fillers and have no significance here.</p> <p>=/5L5L<+%A >7P[with APRS messaging, RMC sentence, with course/speed.</p> <p>=/5L5L<+%A >{?! with APRS messaging, with radio range.</p> <p>=/5L5L<+%A >S]S with APRS messaging, GGA sentence, altitude.</p> <p>@092345z/5L5L<+%A >{?! with APRS messaging, timestamp, radio range.</p>									

10 MIC-E DATA FORMAT

Mic-E Data Format

In Mic-E data format, the station's position, course, speed and display symbol, together with an APRS digipeater path and Mic-E Message Code, are packed into the AX.25 Destination Address and Information fields.

The Information field can also optionally contain either Mic-E telemetry data or Mic-E status.

Mic-E packets can be very short. At the minimum, with no callsigns in the Digipeater Addresses field and no optional telemetry data or Mic-E status text, a complete Mic-E packet is just 25 bytes long (excluding FCS and flags).

Mic-E data format is not only used in the Microphone Encoder unit, it is also used in the PIC-E encoder and in the Kenwood TH-D7 and TM-D700 radios.

Mic-E Data Payload

The Mic-E data format allows a large amount of data to be carried in a very short packet. The data is split between the Destination Address field and the Information field of a standard AX.25 UI-frame.

Destination Address Field — The 7-byte Destination Address field contains the following encoded information:

- The 6 latitude digits.
- A 3-bit Mic-E message identifier, specifying one of 8 Standard Mic-E Message Codes or one of 8 Custom Message Codes.
- The North/South and West/East Indicators.
- The Longitude Offset Indicator.
- The generic APRS digipeater path code.

Although the destination address appears to be quite unconventional, it is still a valid AX.25 address, consisting only of printable 7-bit ASCII values (shifted one bit left) — see the *Amateur Packet-Radio Link-Layer Protocol* specification for full details of the format of standard AX.25 addresses.

Information Field — This field contains the following data:

- The encoded longitude.
- The encoded course and speed.
- The display Symbol Code and Symbol Table Identifier.
- An optional field, containing either Mic-E telemetry data or a Mic-E status text string. The status string can contain plain text, Maidenhead locator or the station's altitude.



Mic-E Destination Address Field

The standard AX.25 Destination Address field consists of 7 bytes, containing 6 callsign characters and the SSID (plus a number of other bits that are not of interest here). When used to carry Mic-E data, however, this field has a quite different format:

Mic-E Data — DESTINATION ADDRESS FIELD Format						
<i>Latitude Digits + Message Identifier + N/S + Longitude Offset + W/E</i>						<i>Digi Code</i>
A rxdddd0	B rxddd0	C rxmmmm0	N rxmmmm0	L rxhhhh0	W rxhhhh0	C rrSSID0
Bytes: 1	1	1	1	1	1	1

The Destination Address field contains:

- Six binary-coded-decimal latitude 4-bit nibbles giving degrees, minutes, and hundredths of minutes: **ddd ddd mmm mmm hhh hhh**
- **A, B and C:** 3-bit Mic-E message identifier.
- **N:** North/South latitude bit (N = 1, S = 0).
- **L:** Longitude offset (L = 1 means add 100 degrees to the longitude computation in the Information field).
- **W:** West/East longitude bit (W = 1, E = 0).
- **SSID:** Generic APRS digipeater path code.
- **C:** Command/Response flag (see AX.25 specification).
- **r:** Reserved for future use (currently 0).
- **x:** 1 or 0.

Destination Address Field Encoding

The following table shows the encoding of the first 6 bytes of the Destination Address field, for all combinations of latitude digit, the 3-bit Mic-E message identifier (A/B/C) and the N/L/W bits.

The encoding allows for position ambiguity, whereby spaces are used in the lesser-significant digits of the latitude and longitude data.

Where a Mic-E message identifier bit is a **1**, it can represent a **1** in a Standard Message identifier or a **1** in a Custom Message identifier.

The ASCII character is left-shifted one bit position prior to transmission.

Mic-E Destination Address Field Encoding

ASCII Char	Lat Digit	Message A/B/C	N/S	Long Offset	W/E
0	0	0	South	+0	East
1	1	0	South	+0	East
2	2	0	South	+0	East
3	3	0	South	+0	East
4	4	0	South	+0	East
5	5	0	South	+0	East
6	6	0	South	+0	East
7	7	0	South	+0	East
8	8	0	South	+0	East
9	9	0	South	+0	East
A	0	1 (Custom)			
B	1	1 (Custom)			
C	2	1 (Custom)			
D	3	1 (Custom)			
E	4	1 (Custom)			
F	5	1 (Custom)			
G	6	1 (Custom)			

ASCII Char	Lat Digit	Message A/B/C	N/S	Long Offset	W/E
H	7	1 (Custom)			
I	8	1 (Custom)			
J	9	1 (Custom)			
K	space	1 (Custom)			
L	space	0	South	+0	East
P	0	1 (Std)	North	+100	West
Q	1	1 (Std)	North	+100	West
R	2	1 (Std)	North	+100	West
S	3	1 (Std)	North	+100	West
T	4	1 (Std)	North	+100	West
U	5	1 (Std)	North	+100	West
V	6	1 (Std)	North	+100	West
W	7	1 (Std)	North	+100	West
X	8	1 (Std)	North	+100	West
Y	9	1 (Std)	North	+100	West
Z	space	1 (Std)	North	+100	West

The following table shows the encoding of the 8 Standard and the 8 Custom message types, for all combinations of the A/B/C message identifier bits:

Standard and Custom Message Types

A	B	C	Standard Mic-E Message Type	Custom Mic-E Message Type
1	1	1	M0: Off Duty	C0: Custom-0
1	1	0	M1: En Route	C1: Custom-1
1	0	1	M2: In Service	C2: Custom-2
1	0	0	M3: Returning	C3: Custom-3
0	1	1	M4: Committed	C4: Custom-4
0	1	0	M5: Special	C5: Custom-5
0	0	1	M6: Priority	C6: Custom-6
0	0	0	M7: Emergency	C7: Emergency



Destination Address SSID Field

The SSID in the Destination Address field of a Mic-E packet is coded to handle either a conventional digipeater VIA path (as specified in the Digipeater Addresses field of the AX.25 frame), or one of 15 generic APRS-compliant digipeater paths.

The SSID field in the Destination Address is encoded as follows:

SSID	Path	SSID	Path
-0	Use VIA path	-8	North path
-1	WIDE1-1	-9	South path
-2	WIDE2-2	-10	East path
-3	WIDE3-3	-11	West path
-4	WIDE4-4	-12	North path + WIDE
-5	WIDE5-5	-13	South path + WIDE
-6	WIDE6-6	-14	East path + WIDE
-7	WIDE7-7	-15	West path + WIDE

Mic-E Information Field

The Information field is used to complete the Position Report that was begun in the Destination Address field. The encoding used is different from the destination address since the content is not constrained to be printable, shifted 7-bit ASCII, as it is in the address. However, full 8-bit binary is not used — all values are offset by 28 and further operations (described below) are performed on some of the values to make almost all of the data printable ASCII.

The format of the Information field is as follows:

<i>Mic-E Data — INFORMATION FIELD Format</i>									
Data Type ID	Longitude			Speed and Course			Symbol Code	Sym Table ID	Mic-E Telemetry Data
	d+28	m+28	h+28	SP+28	DC+28	SE+28			Mic-E Status Text
Bytes:	1	1	1	1	1	1	1	1	n

Information Field Data

The Information field is encoded as follows:

- APRS Data Type Identifier. One of:
 - \ Current GPS data.
 - ' Old GPS data.
 - 0x1c Current GPS data (Rev. 0 beta units only).
 - 0x1d Old GPS data (Rev. 0 beta units only).
- d+28**: Binary degrees of longitude. To decode:
 - subtract 28
 - if the **L** bit is set, add 100 to get final value of longitude.
 - subtract 80 if $180 \leq D \leq 189$.

- 4. or, subtract 190 if $190 \leq D \leq 199$.
- **m+28**: Binary minutes of longitude. To decode:
 1. subtract 28.
 2. subtract 60 if $M \geq 60$.
- **h+28**: Binary hundredths of minutes of longitude. Note that 00...03 hundredths will be unprintable.
- **SP+28**: First part of speed in knots. Subtract 28 and multiply by 10.
- **DC+28**: Second part of speed and first part of course. Subtract 28 and divide by ten. Quotient is units of speed. Remainder is hundreds of degrees.
- **SE+28**: Second part of course in degrees. Subtract 28, the add to remainder, above.

Finally, make these course and speed adjustments:

1. if speed ≥ 800 knots, subtract 800.
2. if course ≥ 400 , subtract 400.

Note: A valid course is in the range 001–360 degrees, clockwise from north. A value of 000 means that the course is unknown or undefined.

- **\$**: Symbol Code.
- **/**: Symbol Table Identifier

Mic-E Telemetry Data

The Information field may optionally contain either Mic-E telemetry data values or Mic-E status text.

If the byte following the Symbol Table Identifier is one of the Telemetry Flag characters (****, **'** or 0x1d), then telemetry data follows:

Optional Mic-E Telemetry Data					
Telemetry Flag	Telemetry Data Channels				
	Ch 1	Ch 2	Ch 3	Ch 4	Ch 5
F					
Bytes:	1	2	2	2	2

The Telemetry Flag **F** is one of:

- ** 2 printable hex telemetry values follow (channels 1 and 3).
- '** 5 printable hex telemetry values follow.
- 0x1d 5 binary telemetry values follow (Rev. 0 beta units only)

If **F** is **** or **'**, the value is a 2-digit printable hexadecimal representation of a binary value ranging from 0–255. For example, 254 is represented as **FE**.



Mic-E Status Text

As an alternative to telemetry data, the packet may include Mic-E status text. The status text may be any length that fits in the rest of the Information field.

The Mic-E status text must not start with `\,` or `0x1d`, otherwise it will be confused with telemetry data.

It is possible to include a standard APRS-formatted position in the Mic-E status text field. A suitable position will cause the APRS display software to override any position data the Mic-E has encoded. This is useful if using a Mic-E without a GPS receiver.

Note: The Kenwood radios automatically insert a special type code at the front of the status text string:

Kenwood TH-D7: `>`
 Kenwood TM-D700: `]`

These characters should not be confused with the APRS Data Type Identifier that appears at the start of reports.

It is envisaged that other Mic-E-compatible devices will be allocated their own type codes in future.

Note: When Kenwood radios receive the status, they can only display a small number of text characters:

Kenwood TH-D7: 20 characters
 Kenwood TM-D700: 28 characters

**Maidenhead
Locator in
Comment Field**

The Mic-E status text field can contain a Maidenhead locator.

If the locator is followed by a plain text comment, the first character of the text *must* be a space. For example:

`IO91sx/G_Hello_world` (from a Mic-E or PIC-E)
`>IO91sx/G_Hello_world` (from a Kenwood TH-D7)
`]IO91sx/G_Hello_world` (from a Kenwood TM-D700)

(`/G` is the grid locator symbol).

Altitude in the Comment Field

The Mic-E status text field can contain the station's altitude. The altitude is expressed in the form `xxxj`, where `xxx` is in meters relative to 10km below mean sea level (the deepest ocean), to base 91.

For example, to compute the `xxx` characters for an altitude of 200 feet:

$$\begin{aligned} 200 \text{ feet} &= 61 \text{ meters} = 10061 \text{ meters relative to the datum} \\ 10061 / 91^2 &= \mathbf{1}, \text{ remainder } 1780 \\ 1780 / 91 &= \mathbf{19}, \text{ remainder } \mathbf{51} \end{aligned}$$

Adding 33 to each of the highlighted values gives 34, 52 and 84 for the ASCII codes of `xxx`.

Thus the 4-character altitude string is `"4Tj`

If the altitude is followed by a plain text comment, the first character of the text *must* be a space. Some examples:

```
>"4Tj
j"4Tj
>"4Tj _My_house
```

Important Note: The status text field should only contain altitude data if the data is derived from a DGPS-corrected source or if it is manually entered. If the altitude is derived from an *uncorrected* source (such as a raw NMEA GPS sentence), it will probably fluctuate wildly because of GPS Selective Availability — this information will be useless, so it is pointless to transmit it.

Mic-E Data in Non-APRS Networks

Some parts of the Mic-E AX.25 Information field may contain binary data (i.e. non-printable ASCII characters). If such a packet is constrained to the APRS network, this should not cause any difficulties.

If, however, the packet is to be forwarded via a network that does not reliably preserve binary data (e.g. the Internet), then it is necessary to convert the data to a format that will preserve it.

Further, if the packet subsequently re-emerges back onto the APRS network, it will then be necessary to re-convert the data back to its original format.



Example of Mic-E Packet Decoding

To illustrate how data is packed into a Mic-E packet, here is an example showing two packets received on-air from the same station. The first is a regular (non Mic-E) status packet, showing the station's location, course and speed:

```
K1ATV-9>APRS,TCPIP,KB7WFO*:@091929z3325.63N/11207.73Wj251/000/Mic-E/M3/Returning
```

Decoding the data in detail:

- It is a Position Report with time.
- It is timed at 1929z on the 9th of the month.
- The latitude is 33 degrees 25.63 minutes north.
- The longitude is 112 degrees 7.73 minutes West.
- The symbol **j** is from the Primary Symbol Table **J**, representing a jeep.
- The course is 251 degrees.
- The speed is zero knots.
- The comment string says that the Mic-E unit is sending message # 3 (“Returning”).

The corresponding Mic-E packet is as follows:

```
K1ATV-9>S32UVT,W7MOT-1,KD7DR-3*,WIDE/V: `(_fl"Oj/'7200007100Mon-146.84 Earl ATV
```

Decoding this packet, the destination address is **S32UVT**, which represents:

Dest Address:	S	3	2	U	V	T
Latitude Digit	3	3	2	5	6	4
Status Bit	A=1 (Std)	B=0	C=0	North	+100	West

- It has message bits A/B/C = 100, a standard message meaning “Returning”.
- The latitude is 33 degrees 25.64 minutes north (close enough to the posit in the first packet).
- The longitude is in the western hemisphere, and 100 degrees has to be added to the longitude computation in the Information field.

The Information field contains ``(_fℓ"Oj/'7200007100`. Breaking this down into pieces:

- ``` is the APRS Data Type Identifier for a Mic-E packet.
- `(` is the d+28 byte. The `(` character has the value 40 decimal. Subtracting 28 gives 12. The longitude offset (in the destination address) is +100 degrees, so the longitude is $100 + 12 = 112$ degrees.
- `"` is the m+28 byte. The `"` character has the value 95 decimal. Subtracting 28 gives 67. This is ≥ 60 , so subtracting 60 gives a value of 7 minutes longitude.
- `f` is the h+28 byte. The `f` character has the value 102 decimal. Subtracting 28 gives 74 hundredths of a minute.

Thus the longitude is 112 degrees 7.74 minutes West (again close enough to the posit in the first packet).

The speed and course are calculated as follows:

- `ℓ` (lower-case “L”) is the SP+28 byte. The `ℓ` character has the value 108 decimal. Subtracting 28 and multiplying by 10 gives the first part of the speed computation in knots, 800.
- `"` is the DC+28 byte. The `"` character has the value 34 decimal. Subtracting 28 gives 6. Dividing this by 10 gives a quotient of 0 (units of speed) and a remainder of 6 (course bearing in hundreds of degrees). Adding the first part of the speed (800) to the quotient (0) gives a computed speed of 800 knots. This is ≥ 800 , so subtracting 800 gives a real speed of zero knots.
- `O` (upper-case letter “O”) is the SE+28 byte. The `O` character has the value 79 decimal. Subtracting 28 gives 51. Adding this to the remainder calculated above multiplied by 100 (i.e. 600) gives 651. This is ≥ 400 , so subtracting 400 gives the final value of 251 degrees for the course.

The next two characters (`j/`) represent the jeep symbol from the Primary Symbol Table.

The data continues with the characters `'7200007100`. The `'` indicates that that 5 bytes of telemetry follow, coded in hexadecimal. These convert to:

0x72 = 114 decimal
 0x00 = 0 decimal
 0x00 = 0 decimal
 0x71 = 113 decimal
 0x00 = 0 decimal



11 OBJECT AND ITEM REPORTS

Objects Any APRS station can manually report the position of an APRS Object (e.g. another station or a weather phenomenon). Object reports are identical to posit reports, except that the Position Report is preceded by a fixed 9-character Object name. Object names are case-sensitive.

The **;** is the Data Type Identifier for an Object Report, and a ***** separates the Object name and the posit. The position may be in lat/long or compressed lat/long format, it may be time stamped, and the report may also contain Extended Data.

Implementation Recommendation: When an APRS Object is displayed on the screen, the callsign of the station sending the object should be associated with the Object.

Object Report Format — with Lat/Long position										
; or -	Object Name	*	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	Comment (max 36 chars with Data Extension, or 43 without)	
								Power/Height/Gain/Dir		
								Radio Range		
								DF Signal Strength		
Bytes:	1	9	1	7	8	1	9	1	7	0-36/43
<p><u>Example</u> ;LEADER_*** *092345z4903.50N/07201.75W>088/036 At 2345 hours zulu on the 9th of the month, the "Leader" was in the car at 49°3'30"N/72°1'45"W, heading 88 deg at 36 knots.</p>										

Items Items are the same as APRS Objects, except that the Data Type Identifier is a **)**, there is no time stamp, and the Item name is variable length from 3 to 9 bytes followed by the standard **!** format position.

Item Report Format — with Lat/Long position									
)	Item Name	!	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	Comment (max 36 chars with Data Extension, or 43 without)	
							Power/Height/Gain/Dir		
							Radio Range		
							DF Signal Strength		
Bytes:	1	3-9	1	8	1	9	1	7	0-36/43
<p><u>Example</u>)BRENDA!4903.50N)07201.75W@088/036 Hurricane Brenda is at 49°3'30"N/72°1'45"W, heading 88 degrees at 36 knots. (The) and @ characters represent the hurricane display symbol).</p>									

- Replacement Object / Item** A fundamental precept in APRS is that any station may take over reporting responsibility of an APRS Object or Item by simply transmitting the Object/Item in a new location (or the same location).
The original station will cease transmitting that Object when it sees an incoming same-named object from another station. All receiving software must attach the call of the sending station to the Object.
- Killed Object** The station reporting an APRS Object may kill the Object from everyone else's screen by continuing to transmit it but with the ; Data Type Identifier replaced by a minus sign -.
The format is otherwise the same.
For example, to kill the LEADER_ Object:
-LEADER_*234517h4903.50N307201.75W>088/036
- Area Objects** Using the L symbol (i.e. the lower-case letter "L" symbol from the Alternate Symbol Table) it is possible to define circle, line, ellipse, triangle and box objects in all colors, either open or filled in, any size from 60 feet to 100 miles.
These Objects are useful for real-time events such as for a search-and-rescue, or adding a special road or route for a special event.
The Object format is specified as a 7-character APRS Data Extension Tyy/Cxx immediately following the L Symbol Code. For example:
;OBJECT_*ddmm.hhN\dddmm.hhWLTyy/Cxx
where:
T is the type of object shape.
/C is the color of the object.
yy is the square root of the latitude offset in 1/100ths of a degree.
xx is the square root of the longitude offset in 1/100ths of a degree.



The object type and color codes are as follows:

T	Object Type	/C	Object Color
0	Open circle	/0	Black
1	Line	/1	Blue
2	Open ellipse	/2	Green
3	Open triangle	/3	Cyan
4	Open box	/4	Red
5	Color-filled circle	/5	Violet
6	Colored Line	/6	Yellow
7	Color-filled ellipse	/7	Gray
8	Color-filled triangle	/8	Dim Black
9	Color-filled box	/9	Dim Blue
		10	Dim Green
		11	Dim Cyan
		12	Dim Red
		13	Dim Violet
		14	Dim Yellow
		15	Dim Gray

The latitude/longitude position is the upper left corner of the object, and the offsets are relative to this position — the yy offset is *down* from this position and the xx offset is to the *right* of this position. (An exception is the special case of a Type 6 colored line which is drawn down and to the *left*).

Here are some examples of Object Position Reports. The latitude and longitude offsets are each one degree (i.e. 100/100ths of a degree), so yy=xx= SQR (100)=10.

```
;SEARCH_***4903.50N\07201.75W/710/310
```

A cyan filled ellipse, yy=10, xx=10

```
;SEARCH_***4903.50N\07201.75W/8101310
```

A dim violet filled triangle, yy=10, xx=10

Further, with the line option it is possible to specify a “corridor” either side of the central line. The width of the corridor (in miles) either side of the line is specified in the comment text, enclosed by {}.

For example:

```
;FLIGHTPTH*4903.50N\07201.75W/610/310{100}
```

A cyan line, with a 100-mile corridor either side

N.B. The color fill option should be used with care, since a color-filled object will obscure information displayed underneath it.

Signpost Objects

Signpost Objects (with the symbol **\m**) display as a yellow box with a 1- to 3-character overlay on them. The overlay is specified by enclosing the 1–3 characters in braces in the comment field. Thus a signpost with {55} would appear as a sign with “55” on it.

For example:

```
; I91_3N_...*4903.50N\07201.75Wm { 5 5 }
```

This was originally designed for posting the speed of traffic past speed measuring devices, but can be used for any purpose.

Implementation Recommendation: Signposts should not display any call sign or name, and to avoid clutter should only be displayed at close range.

Obsolete Object Format

Some stations transmit Object reports without the **;** APRS Data Type Identifier. This format is obsolete. Some software may still decode it, but this is not required.



12 WEATHER REPORTS

APRS is an ideal tool for reporting weather conditions via packet. APRS supports serial data transmissions from the Peet Brothers, Ultimeter and Davis home weather stations. It is even possible to mount an Ultimeter remotely with only a TNC and radio to report and plot conditions. APRS is also ideally suited for the Skywarn weather observer initiative.

Data Type Identifiers

The following APRS Data Type Identifiers are used in Weather Reports containing *raw* data:

- ! Generic raw weather data
- ! Ultimeter 2000
- # Peet Bros U-II
- \$ Ultimeter 2000
- * Peet Bros U-II

In addition, where the raw data has been post-processed (for example, by the insertion of station location information), the four position Data Type Identifiers !, =, / and @ may be used instead. In this case, the Weather Report is identified with the weather symbol / in the APRS Data.

Time Stamps

Some Weather Reports can contain timestamps. Depending on the weather data format, timestamps can be in any of the 3 standard timestamp formats.

Positionless Weather Reports

Generic raw weather data from a stand-alone weather station is contained in the Information Field of an APRS AX.25 frame:

Generic Raw Weather Report Format				
Data Type ID	Time MDHM	Weather Data	APRS Software S	WX Unit uuuu
1	8	n	1	2-4
<p><u>Examples</u></p> <pre>*7007600000000 Peet Bros U-II #50B7500820082 Peet Bros U-II ! !006B005803500000----03E9-----002105140000005D Ultimeter 2000 \$ULTW0031003702CE0069----000086A00001----011901CC00000005 Ultimeter 2000 _10090556c220s004g005t077r000p000P000h50b09900wRSW report derived from Radio Shack WX station data.</pre>				



Weather Data The format of weather data within a Weather Report differs according to the type of weather station unit, but generically consists of some or all of the following elements:

Generic Weather Data within a Weather Report								
Wind Direction c c c c	Wind Speed s s s s	Gust g g g g	Temp t t t t	Rain Last Hr r r r r	Rain Last 24 Hrs p p p p	Rain Since Midnight P P P P	Humidity h h h	Barometric Pressure b b b b b b
Bytes:	4	4	4	4	4	4	3	5

where:

- c** = wind direction (in degrees).
- s** = sustained one-minute wind speed (in mph).
- g** = gust (peak wind speed in mph in the last 5 minutes).
- t** = temperature (in degrees Fahrenheit).
- r** = rainfall (in hundredths of an inch) in the last hour.
- p** = rainfall (in hundredths of an inch) in the last 24 hours.
- P** = rainfall (in hundredths of an inch) since midnight.
- h** = humidity (in %. 00 = 100%).
- b** = barometric pressure (in tenths of millibars/tenths of hPascal).

Other parameters that are available on some weather station units include:

- L** = luminosity (in watts per square meter) 999 and below.
- ℓ** (lower-case letter “L”) = luminosity (in watts per square meter) 1000 and above.
(L is inserted in place of one of the rain values).
- s** = snowfall (in inches) in the last 24 hours.
- #** = raw rain counter

Note: The weather data always starts with wind direction, wind speed, gust and temperature, but the remaining parameters may be in a different order (or may not even exist).

APRS Software Type A Weather Report may contain a single-character code S for the type of APRS software that is running at the weather station:

- d** = APRSdos
- M** = MacAPRS
- P** = pocketAPRS
- S** = APRS+SA
- W** = WinAPRS
- X** = X-APRS (Linux)



Weather Unit Type

A Weather Report may contain a 2–4 character code `uuuu` for the type of weather station unit:

- Dvs** = Davis
- HKT** = Heathkit
- PIC** = PIC device
- RSW** = Radio Shack
- U-II** = Original Ultimeter U-II (auto mode)
- U2R** = Original Ultimeter U-II (remote mode)
- U2k** = Ultimeter 500/2000
- U2kr** = Remote Ultimeter logger
- U5** = Ultimeter 500
- Upkm** = Remote Ultimeter packet mode

Weather Reports with Timestamp and Position

A Weather Report can contain a timestamp and location information, using any of the legal lat/long and compressed lat/long position formats described earlier. An APRS Object may also have weather information associated with it.

Examples of report formats are shown below. Note that the Symbol Code in every case is the `_` (underscore). Also, the 7-byte Wind Direction and Wind Speed Data Extension replace the `cccc` and `ssss` fields of a raw weather report.

Weather Report Format — with Lat/Long position										
<code>!</code> or <code>=</code> or <code>/</code> or <code>@</code>	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code <code>_</code>	Wind Directn/ Speed	Weather Data	APRS Software S	WX Unit uuuu	
Bytes:	1	7	8	1	9	1	7	n	1	2-4
<p><u>Examples</u> !4903.50N/07201.75W_220/004g005t077r000p000P000h50b09900wRSW @092345z4903.50N/07201.75W_220/004g005t077r000p000P000h50b09900wRSW</p>										

Weather Report Format — with Compressed Lat/Long position											
! or = or / or @	Time DHM / HMS	Sym Table ID	Comp Lat YYYY	Comp Long XXXX	Symbol Code ■	Comp Wind Directn/ Speed	Comp Type T	WX Data	APRS Software S	WX Unit uuuu	
Bytes:	1	7	1	4	4	1	2	1	n	1	2-4
<u>Examples</u> =/5L5L<+%A_7P!g005t077r000p000P000h50b09900wRSW @092345z/5L5L<+%A_7P!g005t077r000p000P000h50b09900wRSW											

Weather Report Format — with Object and Lat/Long position												
! or =	Object Name	*	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code ■	Wind Directn/ Speed	WX Data	APRS Software S	WX Unit uuuu	
Bytes:	1	9	1	7	8	1	9	1	7	n	1	2-4
<u>Examples</u> ;BRENDA_***4903.50N/07201.75W_220/004g005t077r000p000P000h50b09900wRSW ;BRENDA_***092345z4903.50N/07201.75W_220/004g005t077r000p000P000h50b09...												

Weather Reports without Position Information

APRS cannot display weather data on a map until it knows the location of the sending station. If the Weather Report does not include location information, the station has to occasionally send an additional packet containing its position.



Storm Data APRS reports can contain data relating to tropical storms, hurricanes and tropical depressions. The format of the data is as follows:

Storm Data													
Course/ Speed	/	Storm Type ST	/	Sustained Wind Speed www	^	Peak Wind Gusts ggg	>	Radius Hurricane Winds RRR	&	Radius Trop Storm Winds rrr	/	Central Pressure ppp...	/
Bytes: 7	1	2	1	3	1	3	1	3	1	3	1	n	1

where: ST = **TS** (Tropical Storm)
HC (Hurricane)
TD (Tropical Depression).

www = sustained wind speed (in mph).

ggg = gust (peak wind speed in mph in the last 5 minutes).

RRR = radius of hurricane winds (in miles).

rrr = radius of tropical storm winds (in miles).

ppp... = central pressure (in millibars/hectoPascal).

The final slash is optional, required if the data is followed by a comment.

Storm data will usually be included in an Object Report, but may also be included in a Position Report or an Item Report.

The display symbol will be either:

/@ Hurricane Future Prediction

\@ Hurricane/Tropical Storm (current position)

For example, the progress of Hurricane Brenda could be expressed in Object Reports like these:

```
;BRENDA_...*092345z4903.50N/07201.75W@088/036/HC/104^123>037&065/0980/comment
;BRENDA_...*092345z4903.50N\07201.75W@088/036/HC/104^123>037&065/0980/comment
```

13 TELEMETRY DATA

Telemetry Report Format

The AX.25 Information field can contain telemetry data. The APRS Data Type Identifier is **T**.

There are five analog data values (expressed as 3-digit decimal numbers in the range 000–255), followed by a single 8-bit digital data value (expressed as 8 bytes, each containing **1** or **0**).

The Kantronics KPC-3+ TNC and APRS Micro Interface Module (MIM) use this format.

Telemetry Report Format							
T	Sequence No #nnn,	Analog Value 1 aaa,	Analog Value 2 aaa,	Analog Value 3 aaa,	Analog Value 4 aaa,	Analog Value 5 aaa,	Digital Value bbbbbbbb
Bytes:	1	5	4	4	4	4	8
<u>Example</u> T#005,199,000,255,073,123,01101001							

On-Air Definition of Telemetry Parameters

In principle, received telemetry data may be interpreted in any appropriate way. In practice, however, an APRS user can define the telemetry parameters (such as quadratic coefficients for the analog values, or the meaning of the binary data) at any time, and then send these definitions as beacons to other stations, so that they too know how to interpret the data.

This is achieved by sending four UI beacons:

- A Parameter Name Beacon.
- A Unit/Label Beacon.
- An Equation Coefficients Beacon.
- A Bit Sense/Project Name Beacon.

Note: These beacons are *not* APRS Bulletins — they are ordinary UI beacon text frames, and do not have any APRS Data Type Identifier.

The beacons are addressed to the callsign of the station transmitting the telemetry data. For example, if N0QBF launches a balloon with the callsign N0QBF-11, then the four beacons are addressed to N0QBF-11.



Parameter Name Beacon The Parameter Name Beacon contains the names (N) associated with the five analog channels and the 8 digital channels. Its format is as follows:

Telemetry Parameter Name Beacon (sent as ordinary UI Beacon Text)													
Note the different byte counts, which include commas where shown. The list may stop at any field.													
	A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	B6	B7	B8
	PARM.	N...	,N...										
Bytes:	5	1-7	1-7	1-6	1-6	1-5	1-6	1-5	1-4	1-4	1-4	1-3	1-3
<u>Example of a PARM beacon addressed to N0QBF-11 (as an ordinary non-APRS UI Beacon):</u> PARM.Battery,Btemp,ATemp,Pres,Alt,Camra,Chut,Sun,10m,ATV													

Note: The field widths are not all the same (this is a legacy arising from earlier limitations in display screen width). Note also that the byte counts *include* the comma separators where shown.

The list can terminate after any field.

Unit/Label Beacon The Unit/Label Beacon specifies the units (U) for the analog values, and the labels (L) associated with the digital channels:

Telemetry Unit/Label Beacon (sent as ordinary UI Beacon Text)													
Note the different byte counts, which include commas where shown. The list may stop at any field.													
	A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	B6	B7	B8
	UNIT.	U...	,U...	,U...	,U...	,L...							
Bytes:	5	1-7	1-7	1-6	1-6	1-5	1-6	1-5	1-4	1-4	1-4	1-3	1-3
<u>Example of a UNIT beacon addressed to N0QBF-11 (as an ordinary non-APRS UI Beacon):</u> UNIT.v/100,deg.F,deg.F,Mbar,Kft,Click,OPEN,on,on,hi													

Note: Again, the field widths are not all the same, and the byte counts *include* the comma separators where shown.

The list can terminate after any field.

Equation Coefficients Beacon The Equation Coefficients Beacon contains three coefficients (a, b and c) for each of the five analog channels.

Telemetry Equation Coefficients Beacon (sent as ordinary UI Beacon Text)															
The list may stop at any field. Value = a * x^2 + b*x + c															
	A1			A2			A3			A4			A5		
EQNS.	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c
Bytes:	5	n	n	n	n	n	n	n	n	n	n	n	n	n	n
<u>Example of an EQNS beacon addressed to N0QBF-11 (as an ordinary non-APRS UI Beacon):</u> EQNS.0,5.2,0,0,.53,-32,3,4.39,49,-32,3,18,1,2,3															

To obtain the final value of an analog channel, these coefficients are substituted into the equation:

$$a * v^2 + b * v + c$$

where v is the raw received analog value.

For example, analog channel A1 in the above beacon examples relates to the battery voltage, expressed in hundredths of volts, and a = 0, b = 5.2, c = 0. If the raw received value v is 199, then the voltage is calculated as:

$$\begin{aligned} \text{voltage} &= 0 * 128^2 + 5.2 * 199 + 0 \\ &= 1034.8 \text{ hundredths of a volt} \\ &= 10.348 \text{ volts} \end{aligned}$$

**Bit Sense/
Project Name
Beacon**

The Bit Sense/Project Name Beacon contains two types of information:

- A bit pattern of 8 ones and zeros, specifying the sense of each digital channel that matches the corresponding label.
- The name of the project associated with the telemetry station.

Telemetry Bit Sense/Project Name Beacon (sent as ordinary UI Beacon Text)									
BITS.	B1	B2	B3	B4	B5	B6	B7	B8	Project Title
Bytes:	5	1	1	1	1	1	1	1	0-23
<u>Example of a BITS beacon addressed to N0QBF-11 (as an ordinary non-APRS UI Beacon):</u> BITS.10110000,N0QBF's Big Balloon									

Thus in the above beacon examples, if digital channel B1 is 1, this indicates the camera has clicked. If channel B2 is 0, the parachute has opened, and so on.



14 MESSAGES, BULLETINS AND ANNOUNCEMENTS

APRS messages, bulletins and announcements are packets containing free format text strings, and are intended to convey human-readable information. A message is intended for reception by a single specified recipient, and an acknowledgement is usually expected. Bulletins and announcements are intended for reception by multiple recipients, and are not acknowledged.

Messages

An APRS message is a text string with a specified addressee. The addressee is a fixed 9-character field (padded with spaces if necessary) following the **:** Data Type Identifier. The addressee field is followed by another **:**, then the text of the message.

The message text may be up to 67 characters long.

A message may also have an optional message identifier, which is appended to the message text. The message identifier consists of the character **{** followed by a message number (up to 5 alphanumeric characters, no spaces) to identify the message.

Messages *without* a message identifier are not to be acknowledged.

Messages *with* a message identifier are intended to be acknowledged by the addressee. The sending station will repeatedly send the message until it receives an acknowledgement, or it is canceled, or it times out.

Message Format						
	:	Addressee	:	Message Text (max 67 chars)	Message ID	
					{	Message No xxxxxx
Bytes:	1	9	1	0-67	1	1-5
<p><u>Examples</u></p> <p>:WU2Z_____:Testing A message for WU2Z, containing the text "Testing", no acknowledgement expected. (Note the filler spaces in the 9-character addressee field).</p> <p>:WU2Z_____:Testing{003 The same message, Message No=003, acknowledgement expected.</p> <p>:EMAIL_____:msproul@ap.org Test email An e-mail message (N.B. This is an example of how such a message could be constructed. APRS itself does not support e-mail delivery)</p>						

Message Acknowledgements

A message acknowledgement is similar to a message, except that the message text field contains just the letters **ack**, and this is followed by the Message Number being acknowledged.

Message Acknowledgement Format					
	Addresssee		ack	Message No xxxxxx	
Bytes:	1	9	1	3	1-5
<u>Example</u> :KB2ICI-14:ack003					

Message Groups

An APRS receiving station can specify special Message Groups, containing lists of callsigns that the station will read messages from (in addition to messages addressed to itself). Such Message Groups are defined internally by the user at the receiving station, and are used to filter received message traffic.

The receiving station will read all messages addressed to ALL, QST or CQ.

The receiving station will only acknowledge messages addressed to itself, and not any messages received which were addressed to any group callsign.

General Bulletins

General bulletins are messages where the addressee consists of the letters **BLN** followed by a single-*digit* bulletin identifier, followed by 5 filler spaces. General bulletins are generally transmitted a few times an hour for a few hours, and typically contain time sensitive information (such as weather status).

Bulletin text may be up to 67 characters long.

General Bulletin Format						
	BLN	Bulletin ID n	_____	:	Bulletin Text (max 67 characters)	
Bytes:	1	3	1	5	1	0-67
<u>Example</u> :BLN3_____:Snow expected in Tampa RSN						



Announcements

Announcements are similar to general bulletins, except that the letters **BLN** are followed by a single upper-case *letter* announcement identifier. Announcements are transmitted much less frequently than bulletins (but perhaps for several days), and although possibly timely in nature they are usually not time critical.

Users must be alerted on arrival of a new bulletin or announcement.

Announcement Format						
	:	BLN	Announcement Identifier x	_____	:	Announcement Text (max 67 characters)
Bytes:	1	3	1	5	1	0-67
<u>Example</u> :BLNQ_____ :Mt St Helen digi will be QRT this weekend						

Group Bulletins

Bulletins may be sent to *bulletin groups*. A bulletin group address consists of the letters **BLN**, followed by a single-*digit* group bulletin identifier, followed in turn by the name of the group (up to 5 characters long, with filler spaces to pad the name to 5 characters).

Group Bulletin Format						
	:	BLN	Group Bulletin ID n	Group Name	:	Group Bulletin Text (max 67 characters)
Bytes:	1	3	1	5	1	0-67
<u>Example</u> :BLN4WX___ :Stand by your snowplows Group bulletin number 4 to the WX group. (Note the filler spaces in the group name).						

A receiving station can specify a list of bulletin groups of interest. The list is defined internally by the user at the receiving station. If a group is selected from the list, the station will only copy bulletins for that group, plus any general bulletins. If the list is empty, all bulletins are received and generate alerts.

National Weather Service Bulletins

APRS recognizes special National Weather Service bulletins with addresses of the form **NWS-xxxxxx**.

<i>National Weather Service Bulletin Format</i>			
	: NWS-xxxxxx :		<i>NWS Bulletin Text</i>
Bytes:	1	9	1 n
<p><u>Example</u></p> <p>:NWS-WARN_:092010z, THUNDER_STORM, AR_ASHLEY, {S9JbA (Note: The “message identifier” at the end is for reference only, as receiving stations do not acknowledge bulletins).</p>			

Obsolete Bulletin and Announcement Format

Some stations transmit bulletins and announcements without the **:** APRS Data Type Identifier. This format is obsolete. Some software may still decode it, but this is not required.



15 STATION CAPABILITIES, QUERIES AND RESPONSES

Station Capabilities

A station may set up a file containing a list of one or more Station Capabilities. These define attributes for the station. When queried for its capabilities, the station responds by transmitting the contents of the file, preceded by the < Data Type Identifier.

Each item in the file is a TOKEN or a TOKEN=VALUE pair. More than one item may be on a line, with each item separated by a comma.

Currently defined items include:

```
IGATE,MSG_CNT=n,LOC_CNT=n
```

where IGATE defines the station as an IGate, MSG_CNT is the number of messages transmitted, and LOC_CNT is the number of “local” stations (those to which the IGate will pass messages in the local RF network).

Queries and Responses

There are two types of APRS queries. One is general to all stations and the other is in a message format directed to a single individual station.

Queries always begin with a ?, are one-time transmissions, do not have a message identifier and should not be acknowledged. Similarly the responses to queries are one-time transmissions that also do not have a message identifier, so that they too are not acknowledged.

Each query contains a Query Type. The following Query Types and expected responses are supported:

Query Type	Query	Response
APRS	General — All stations query	Station's position and status
APRSD	Directed — Query an individual station for stations heard direct	List of stations heard direct
APRSH	Directed — Query if an individual station has heard a particular station	Position of heard station as an APRS Object, plus heard statistics for the last 8 hours
APRSM	Directed — Query an individual station for outstanding unacknowledged or undelivered messages	All outstanding messages for the querying station
APRSO	Directed — Query an individual station for its Objects	Station's Objects
APRSP	Directed — Query an individual station for its position	Station's position
APRSS	Directed — Query an individual station for its status	Station's status
APRST	Directed — Query an individual station for a trace (i.e. path by which the packet was heard)	Route trace
IGATE	General — Query all Internet Gateways	IGate station capabilities
WX	General — Query all weather stations	Weather report (and the station's position if it is not included in the Weather Report)

General Queries The format of a general query is as follows:

General Query Format								
?	Query Type	?	Target Footprint					
			Lat	,	Long	,	Radius	
Bytes:	1	n	1	n	1	n	1	4

Examples	
Query	Typical Response
?APRS? General query, with standard posit and status reply.	/092345z4903.50N/07201.75W> >092345zNet Control Center
?APRS? 34.02,-117.15,0200 General query for stations within a target footprint of radius 200 miles centered on 34.02 degrees north, 117.15 degrees west, with standard posit and status reply. (Note the leading space in the latitude, as its value is positive, see below).	/3402.78N11714.02W- >Digi on low power
?IGATE? General query for IGate stations, with a Station Capabilities reply.	<IGATE,MSG_CNT=43,LOC_CNT=14
?WX? Query for weather stations, with a standard Weather Report reply (without a position), followed by a standard posit.	_10090556c220s004g005t077... >090556z4903.50N/07201.75W>

In the case of an ?APRS? query for stations within a particular target footprint, the latitude and longitude parameters are in *floating point* degrees (*not* in APRS lat/long position format).

- North and east coordinates are positive values, indicated by a leading (space).
- South and west coordinates are negative values.
- The radius of the footprint is in miles, expressed as a fixed 4-digit number in whole miles.

All stations inside the specified coverage circle should respond with a Position Report and a Status Report.



Directed Station Queries

Queries addressed to individual stations are in APRS message format (except that they never include a message identifier). The addressee is the callsign of the station being queried.

The message text is the Query Type. This is followed optionally by another callsign — this callsign does not need filler spaces as it is at the end of the data.

<i>Directed Station Query Format</i>						
	<i>Addressee</i>			<i>Query Type</i>	<i>Callsign of Heard Station</i>	
	:	:	?			
Bytes:	1	9	1	1	5	0-9

Examples

<u>Query</u>	<u>Typical Response</u>
:KH2Z_:_: ?APRSD A query asking KH2Z what stations he has heard direct.	:N8UR_:_: Directs=_WALLOU_WD5IVD...
:KH2Z_:_: ?APRSH NOQBF A query asking for the number of times NOQBF was heard in each of the last 8 hours.	:N8UR_:_: NOQBF_HEARD: _1_3_2_..._4_5_6
:KH2Z_:_: ?APRSM A query asking KH2Z for any unacknowledged or undelivered messages for him. KH2Z responds with all such messages.	:N8UR_:_: Testing{003
:KH2Z_:_: ?APRSO A query asking for KH2Z's APRS Objects.	; LEADER_:_*092345z4903.50N/07201.75W>
:KH2Z_:_: ?APRSP A query asking for KH2Z's position.	/092345z4903.50N/07201.75W>
:KH2Z_:_: ?APRSS A query asking for KH2Z's status.	>092345zNet Control Center
:KH2Z_:_: ?APRST A query asking KH2Z for a trace of the route taken to reach him.	:N8UR_:_: KH2Z>APRS, DIGI1, WIDE*:



16 STATUS REPORTS

A Status Report announces the station's current mission or any other single line status to everyone. The report is contained in the AX.25 Information field, and starts with the **>** APRS Data Type Identifier.

The report may optionally contain a timestamp.

Note: The timestamp can *only* be in DHM *zulu* format.

The status text occupies the rest of the Information field, and may be up to 62 characters long (if there is no timestamp in the report) or 55 characters (if there is a timestamp).

Status Report Format		
>	<i>Time DHM z</i>	<i>Status Text (max 62 chars if no timestamp, or 55 chars if there is a timestamp)</i>
Bytes: 1	7	0-62 or 0-55
<p><u>Examples</u></p> <p>>Net Control Center without timestamp.</p> <p>>092345zNet Control Center with timestamp.</p>		

Although the status will usually be plain language text, there are two cases where the report can contain special information which can be decoded:

- Beam Heading and Power
- Maidenhead grid locator

Status Report with Beam Heading and Power

It is useful to include beam heading (H) and power (P) in packets in meteor scatter work. To keep packets as short as possible, these parameters are encoded into two characters, as follows:

H = beam heading / 10
(H=0–9 for 0–90 degrees, and A–Z for 100–350 degrees).

P = SQR (power / 10)
(P=2 for 40 watts, 3 for 90 watts, 9 for 810 watts).

The HP value appears as the *last* two characters of the status text, preceded by the **^** character — for example, **^B7** means a beam heading of 110 degrees and a power of 490 watts.

The HP value may be combined with the Maidenhead grid locator (as described below), or with any other plain language status text.



Status Report with Maidenhead Grid Locator

The Maidenhead grid locator may be 4 or 6 characters long, and must immediately follow the > Data Type Identifier.

In 6-character form, the last two letters may be upper case or lower case.

The Symbol Table Identifier and Symbol Code follow the locator.

If the report also contains status text, the first character of the text *must* be a space.

A Status Report with Maidenhead locator can not have a timestamp.

Status Report Format — with Maidenhead Grid Locator							
>	Maidenhead Locator			Sym Table ID	Symbol Code	Status Text (<i>starting with a space</i>) (max 54 chars)	
	GG	nn	gg				
Bytes:	1	2	2	2	1	1	1-54
<p><u>Examples</u></p> <p>>IO91sx/G</p> <p>>IO91/G</p> <p>>IO91sx/- My house (Note the space at the start of the status text).</p> <p>>IO91sx/- ^B7 Meteor Scatter beam heading = 110 degrees, power = 490 watts.</p>							

Transmitting Status Reports

Each station should only transmit a Status Report once every net cycle time (i.e. once every 10, 20 or 30 minutes), or in response to a query.

17 NETWORK TUNNELING AND THIRD-PARTY DIGIPEATING

Third-Party Networks

APRS provides a mechanism for formatting packets that are to be transported through third-party (i.e. non AX.25) networks, such as the Internet, an Ethernet LAN or a direct wire connection.

These networks do not understand APRS source, destination and digipeater addresses, so it is necessary to send them as data, along with the original data being transmitted.

Source Path Header

Prior to sending an APRS packet into the third-party network, the APRS address path is prepended to the Data Type Identifier and the rest of the original data.

The prepended address path is known as the Source Path Header. It consists of the source, destination and digipeater callsigns, with associated SSIDs.

The main purpose of introducing the Source Path Header is to allow receiving stations on the far side of the third-party network to identify the sender — this is needed when acknowledging receipt of a message, for example. Knowledge of the source path is also useful in diagnosing network problems.

<i>Data with Source Path Header</i>		
<i>Source Path Header</i>	<i>Data Type ID</i>	<i>Rest of the original data</i>
Bytes: n	1	n

The Source Path Header may be in either of two formats, known as the “TNC-2” format and the “AEA” format (so called because when TNC-2 or AEA-compatible TNCs are operating in terminal MONitor mode they automatically produce headers in these formats).

The APRS Working Group has agreed to move towards standardization on the “TNC-2” format in future implementations.

In most cases, AEA TNCs will produce Source Path Headers in “TNC-2” format when BBSMSGs is set to ON.



The formats of these headers are as follows:

Source Path Header — “TNC-2” Format					
An asterisk follows the digipeater callsign heard.					
Source Callsign (-SSID)	>	Destination Callsign (-SSID)	0-8 Digipeaters		:
			,	Digipeater Callsign (-SSID)(*)	
Bytes: 1-9	1	1-9	0-81		1
<u>Example</u> WB4APR-14>APRS,RELAY*,WIDE: (WIDE digipeater “unused”)					

Source Path Header — “AEA” Format						
An asterisk follows the source or digipeater callsign heard.						
Source Callsign (-SSID)(*)	>	0-8 Digipeaters		>	Destination Callsign (-SSID)	:
		Digipeater Callsign (-SSID)(*)				
Bytes: 1-10		0-81		1	1-9	1
<u>Example</u> WB4APR-14>RELAY*>WIDE>APRS: (WIDE digipeater “unused”)						

In both formats, the SSID may be omitted if it is -0.

In both formats, the callsign of the digipeater from which the incoming packet was heard is indicated with an asterisk. (Alternatively, for “AEA” format only, the asterisk will follow the source callsign if the packet was heard direct from there).

Any digipeaters following the callsign of the station from which the packet was heard are termed “unused”. These unused digipeaters are stripped out when building a Third-Party Header (see below).

Third-Party Header

After a packet emerges from a third-party network, the receiving gateway station modifies it (by inserting a } Third-Party Data Type Identifier and modifying the Source Path Header) before transmitting it on the local APRS network.

The modified Source Path Header is called the Third-Party Header.

Third-party Format		
}	Third-Party Header	Rest of the original data
Bytes: 1	n	n

In a similar way to the Source Path Header, The Third-Party Header can be in either of two formats: “TNC-2” or “AEA” format.

Third Party Header — “TNC-2” format							
Source Path Header (without “unused” digipeaters, * or :)	/	Third-Party Network Identifier (“callsign”)	/	Callsign of Receiving Gateway Station (-SSID)	*	:	
Bytes: 3-99	1	1-9	1	1-9	1	1	
<u>Example</u> WB4APR-14>APRS, RELAY, TCPIP, G9RXG* :							

Third Party Header — “AEA” format									
Source Path Header (without “unused” digipeaters, destination, * or :)		Third-Party Network Identifier (“callsign”)	>	Callsign of Receiving Gateway Station (-SSID)	*	>	Destination Callsign from Source Path Header (-SSID)	:	
Bytes: 2-90		1-9	1	1-9	1	1	1-9	1	
<u>Example</u> WB4APR-14>RELAY>TCPIP>G9RXG*>APRS :									

In both cases, the “unused” digipeater callsigns (i.e. those digipeater callsigns after the asterisk) in the original Source Path Header are stripped out. The asterisk itself is also stripped out of the Source Path Header.

Then two additional callsigns are inserted:

- The Third-Party Network Identifier (e.g. TCPIP). This is a dummy “callsign” that identifies the nature of the third-party network.



Action on Receiving a Third-Party packet

- The callsign of the receiving gateway station, followed by an asterisk.

When another station receives a third-party packet, it can extract the callsign of the original sending station from the Third-Party Header, if it is needed to acknowledge receipt of a message.

The other addresses in the Third-Party Header may be useful for network diagnostic purposes.

An Example of Sending a Message through the Internet

The Scenario:

- WB4APR-14 wants to send a message via the Internet to G3NRW.
- The nearest Internet gateway to WB4APR-14 is K4HG, reachable via a RELAY, WIDE path.
- The nearest Internet gateway to G3NRW is G9RXG.

The Process:

- In the normal way, WB4APR-14 builds a message packet that contains:
`:G3NRW_ :Hi Ian{001`
- WB4APR-14 transmits the packet via his UNPROTO path RELAY, WIDE.
- The Internet gateway K4HG happens to receive this packet from the RELAY digipeater in the path.
- K4HG builds a new packet that contains the source path and the original message:
`WB4APR-14>APRS, RELAY*, WIDE : :G3NRW_ :Hi Ian{001`
- K4HG sends this packet (using telnet) to an APRServer on the Internet.
- All Internet gateways throughout the world that are connected to the APRServe network (including G9RXG) receive the packet.
- G9RXG converts the packet into a Third-Party packet:
`}WB4APR-14>APRS, RELAY, TCPIP, G9RXG* : :G3NRW_ :Hi Ian{001`

Note that the WIDE digipeater was stripped out of the header because it was unused.

- G9RXG transmits the packet over the local APRS network.
- G3NRW receives the packet, strips out the Third-Party Header, and discovers that the packet contains a message for him. From the header, G3NRW then establishes that the acknowledgement is to go back to WB4APR-14.

18 USER-DEFINED DATA FORMAT

The APRS protocol defines many different data formats, but it cannot anticipate every possible data type that programmers may wish to send. The User-Defined data format is designed to fill these gaps. Under this system, program authors are free to send data in any format they choose.

The data in the AX.25 Information field consists of a three-character header:

- { APRS Data Type Identifier.
- U A one-character User ID.
- X A one-character user-defined packet type.

The APRS Working Group will issue User IDs to program authors who express a need.

[Keep in mind there is a limited number of available User IDs, so please do not request one unless you have a true need. The Working Group may require an explanation of your need prior to issuing a character. If only one or two data formats are needed, those may be issued from a User ID pool].

For experimentation, or prior to being issued a User ID, anyone may utilize the User ID character of { without prior notification or approval (i.e. packets beginning with {{ are experimental, and may be sent by anyone).

Important Note: Although there is no restriction on the nature of user-defined data, it is highly recommended that it is represented in printable 7-bit ASCII character form.

<i>User-Defined Data Format</i>			
	<i>User ID</i>	<i>User-Defined Packet Type</i>	<i>User-defined data (printable ASCII recommended)</i>
	{ U	X	
Bytes:	1	1	n
<p><u>Examples</u></p> <p>{Q1qwerty User ID = Q, User-defined packet type = 1.</p> <p>{{zasdfg User ID undefined (experimental), User-defined packet type = z.</p>			

This is envisioned as a way for authors to experiment and build in features specific to their programs, without the danger of a non-standard packet crashing other authors' programs. In keeping with the spirit of the APRS protocol, authors are encouraged to make these formats public. The APRS Working Group will maintain a web site defining all of the assigned User IDs, and either the packet formats provided by the author, or links to their



own web sites which define their formats.

Generally, all formats using this method will be considered optional. No program is required to decode any of these packets, and must ignore any it does not decode. However, it is possible that in the future some of these formats may prove to be of sufficient utility and interest to the entire APRS community that they will be specifically included in future versions of the APRS protocol.



19 OTHER PACKETS

Invalid Data or Test Data Packets

To indicate that a packet contains invalid data, or test data that does not conform to any standard APRS format, the **v** Data Type Identifier is used.

For example, the Mic-E unit will generate such a packet if it detects that a received GPS sentence is not valid.

<i>Invalid Data / Test Data Format</i>	
v	<i>Invalid Data or Test Data</i>
Bytes: 1	n
<p><u>Example</u></p> <p>,191146,v,4214.2466,N,07303.5181,W,417.238,114.5,091099,14.7,W/GPS FIX Invalid GPS data from a Mic-E unit. The unit has interpreted the v character in the received sentence to mean the data is invalid, and has stripped out the \$GPRMC header.</p>	

All Other Packets

Any packet that does not meet any of the formats in this document are assumed to be a status beacon and will show up as status as long as no other properly formatted status has been received. This allows APRS to accept any UI packet addressed to the typical beacon address to be captured as a status message. Typical TNC ID packets fall into this category.

Once a proper status **>** packet has been received from a station it will not be overwritten by other non-status non-identified packets from that station.

Programs can decide to handle these, or ignore them, but they must be able to process them without ill effects.



20 APRS SYMBOLS

Symbol Table Identifier and Symbol Code

A symbol (corresponding to a display icon) is a combination of a one-character Symbol Table Identifier and a one-character Symbol Code.

For example, in the uncompressed Position Report:

```
@092345z4903.50N/07201.75W>088/036...
```

the forward slash **/** is the Symbol Table Identifier and the **>** character is the Symbol Code (in this case representing a “car” icon) from the selected table.

The Symbol Table Identifier character selects one of two Symbol Tables, or it may be used as single-character (alpha or numeric) overlay, as follows:

<i>Symbol Table Identifier</i>	<i>Selected Table or Overlay Symbol</i>
/	Primary Symbol Table (mostly stations)
\	Alternate Symbol Table (mostly Objects)
0-9	Overlay symbol from Alternate Symbol Table with 0-9 (lat/long data format)
A-Z	Overlay symbol from Alternate Symbol Table with A-Z (lat/long data format)
a-j	Overlay symbol from Alternate Symbol Table with 0-9 (compressed lat/long data format) i.e. a-j maps to 0-9.

In the generic case, a symbol from the Primary Symbol Table is represented as the character-pair **/**\$, and a symbol from the Alternate Symbol Table as ****\$.

Symbol Overlays

The Primary and Alternate Symbol Tables are shown in full in Appendix 2.

Where the Symbol Table Identifier is 0-9 or A-Z (with uncompressed data) or a-j (with compressed data), the symbol comes from the *Alternate* Symbol Table, and is overlaid with the identifier (as a single digit or a capital letter).

For example, in the *uncompressed* Position Report:

```
@092345z4903.50N307201.75W>...
```

the digit **3** following the latitude will cause the number “3” to be overlaid on top of the “car” icon (N.B. Because the symbol is overlaid, the **>** Symbol Code here comes from the *Alternate* Symbol Table).

In *compressed* Position Reports, the overlay character is in the range **a-j**, which maps to the digits **0-9** (i.e. a=0, b=1, c=2, d=3 etc).

Thus, in the compressed Position Report:

```
=d5L5L<+%A >7P [
```

the letter **d** is the Symbol Table Identifier corresponding to overlay character “3”. Again, the Symbol Code is taken from the *Alternate* Symbol Table.

Not all symbols from the Alternate Symbol Table may be overlaid in this way — those that can be overlaid are marked as *[with overlay]* in Appendix 2. This means that they are *capable* of taking an overlay, but they do not necessarily need to have one. Thus, for example, the following report uses the car symbol from the Alternate Symbol Table, but does not display an overlay:

```
@092345z4903.50N\07201.75W>...
```

Symbols with Stand-Alone GPS Trackers

Stand-alone TNC trackers which transmit raw GPS have two methods of selecting their symbol.

One method is to set the callsign in the Destination Address field of the AX.25 frame to the form **GPS**_{xyz}. The characters _{xy} point to an entry in the Symbol Table, and the _z is either a (space) or an overlay character.

For example, for a “car” icon without an overlay, the destination address is **GPSMV_** (from the Primary Symbol Table).

If the “car” icon is to be overlaid with a digit “3”, the destination address is **GPSNV3** (from the Alternate Symbol Table).

However, even if a GPS callsign is overlay-capable, it is not actually necessary to specify an overlay. For example, **GPSNV_**.

There are five destination calls of this type that will specify a symbol:

- **GPS**_{xyz} is for stand-alone trackers, taking the symbol from either Symbol Table. Certain symbols from the Alternate Symbol Table may be overlaid.
- **GPSC**_{nn} (e.g. GPSC30) is for symbols from the Primary Symbol Table.
- **GPSE**_{nn} (e.g. GPSE30) is for symbols from the Alternate Symbol Table. These callsigns can not use overlays.
- **SPC**_{xyz} is for stand-alone trackers at special events in SPCL mode.
- **SYM**_{xyz} is reserved (usage to be defined).



APPENDIX 1: APRS DATA FORMATS

This Appendix contains format diagrams for all APRS data formats. The gray fields are optional. Shaded (yellow) characters are literal ASCII characters.

AX.25 UI-FRAME FORMAT									
<i>Flag</i>	<i>Destination Address</i>	<i>Source Address</i>	<i>Digipeater Addresses (0-8)</i>	<i>Control Field (UI)</i>	<i>Protocol ID</i>	<i>INFORMATION FIELD</i>	<i>FCS</i>	<i>Flag</i>	
Bytes:	1	7	7	0-56	1	1	1-256	2	2

Generic APRS Information Field				
<i>Data Type ID</i>	<i>APRS Data</i>	<i>APRS Data Extension</i>	<i>Comment</i>	
Bytes:	1	n	7	n

Lat/Long Position Report Format							
<i>! or = or / or @</i>	<i>Time DHM / HMS</i>	<i>Lat</i>	<i>Sym Table ID</i>	<i>Long</i>	<i>Symbol Code</i>	<i>Comment (max 43 chars)</i>	
Bytes:	1	7	8	1	9	1	0-43

Lat/Long Position Report Format — with Data Extension								
<i>! or = or / or @</i>	<i>Time DHM / HMS</i>	<i>Lat</i>	<i>Sym Table ID</i>	<i>Long</i>	<i>Symbol Code</i>	<i>Course/Speed</i>	<i>Comment (max 36 chars)</i>	
						<i>Power/Height/Gain/Dir</i>		
						<i>Radio Range</i>		
						<i>DF Signal Strength</i>		
Bytes:	1	7	8	1	9	1	7	0-36

DF Report Format									
<i>! or = or / or @</i>	<i>Time DHM / HMS</i>	<i>Lat</i>	<i>Sym Table ID</i>	<i>Long</i>	<i>Symbol Code</i>	<i>Course/Speed</i>	<i>BRG/NRQ</i>	<i>Comment (max 28 chars)</i>	
						<i>Power/Height/Gain/Dir</i>			
						<i>Radio Range</i>			
						<i>DF Signal Strength</i>			
Bytes:	1	7	8	1	9	1	7	8	0-28

Raw NMEA Position Report Format	
NMEA Received Sentence	
\$...E...E...E...E...E...E...E...E...E...
Bytes:	1 25-209

Compressed Lat/Long Position Report Format									
! = / @	Time DHM / HMS	Sym Table ID	Comp Lat YYYY	Comp Long XXXX	Symbol Code	Compressed Course/Speed	Comp Type T	Comment (max 40 chars)	
						Compressed Radio Range			
						Compressed Altitude			
Bytes:	1	1	1	4	4	1	2	1	0-40

Compression Type (T) Byte Format								
Bit:	7	6	5	4	3	2	1	0
Value:	Not used	Not used	GPS Fix	NMEA Source		Compression Origin		
	0	0	0 = old (last) 1 = current	0 0 = other 0 1 = GLL 1 0 = GGA 1 1 = RMC	0 0 0 = Compressed 0 0 1 = TNC BText 0 1 0 = Software (DOS/Mac/Win/+SA) 0 1 1 = [tbd] 1 0 0 = KPC3 1 0 1 = Pico 1 1 0 = Other tracker [tbd] 1 1 1 = Digipeater conversion			

Mic-E Data — DESTINATION ADDRESS FIELD Format						
Latitude Digits + Message Identifier + N/S + Longitude Offset + W/E						Digi Code
A r x d d d d 0	B r x d d d d 0	C r x m m m m m 0	N r x m m m m m 0	L r x h h h h 0	W r x h h h h 0	C r r S S I D 0
Bytes:	1	1	1	1	1	1

Mic-E Data — INFORMATION FIELD Format									
Data Type ID	Longitude			Speed and Course			Symbol Code	Sym Table ID	Mic-E Telemetry Data
	d+28	m+28	h+28	SP+28	DC+28	SE+28			Mic-E Status Text
Bytes:	1	1	1	1	1	1	1	1	n



Object Report Format — with Lat/Long position										
; or -	Object Name	*	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	Comment (max 36 chars with Data Extension, or 43 without)	
								Power/Height/Gain/Dir		
								Radio Range		
								DF Signal Strength		
Bytes:	1	9	1	7	8	1	9	1	7	0-36/43

Item Report Format — with Lat/Long position									
)	Item Name	!	Lat	Sym Table ID	Long	Symbol Code	Course/Speed	Comment (max 36 chars with Data Extension, or 43 without)	
							Power/Height/Gain/Dir		
							Radio Range		
							DF Signal Strength		
Bytes:	1	3-9	1	8	1	9	1	7	0-36/43

Generic Raw Weather Report Format							
Data Type ID	Time MDHM	Weather Data			APRS Software S	WX Unit uuuu	
Bytes:	1	8	n			1	2-4

Generic Weather Data in a Weather Report								
Wind Direction cccc	Wind Speed ssss	Gust gggg	Temp tttt	Rain Last Hr rrrr	Rain Last 24 Hrs pppp	Rain Since Midnight pppp	Humidity hhh	Barometric Pressure bbbbbbb
Bytes:	4	4	4	4	4	4	3	5

Weather Report Format — with Lat/Long position										
! or = or / or @	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code	Wind Directn/ Speed	Weather Data	APRS Software S	WX Unit uuuu	
Bytes:	1	7	8	1	9	1	7	n	1	2-4

Weather Report Format — with Compressed Lat/Long position											
! or = / / @	Time DHM / HMS	Sym Table ID	Comp Lat YYYY	Comp Long XXXX	Symbol Code ■	Comp Wind Directn/ Speed	Comp Type T	WX Data	APRS Software S	WX Unit uuuu	
Bytes:	1	7	1	4	4	1	2	1	n	1	2-4

Weather Report Format — with Object and Lat/Long position												
? or -	Object Name	*	Time DHM / HMS	Lat	Sym Table ID	Long	Symbol Code ■	Wind Directn/ Speed	WX Data	APRS Software S	WX Unit uuuu	
Bytes:	1	9	1	7	8	1	9	1	7	n	1	2-4

Storm Data														
Course/ Speed	/	Storm Type ST	/	Sustained Wind Speed www	^	Peak Wind Gusts ggg	>	Radius Hurricane Winds RRR	&	Radius Trop Storm Winds rrr	/	Central Pressure ppp...	/	
Bytes:	7	1	2	1	3	1	3	1	3	1	3	1	n	1

Telemetry Data Format							
T	Sequence No #nnn,	Analog Value 1 aaa,	Analog Value 2 aaa,	Analog Value 3 aaa,	Analog Value 4 aaa,	Analog Value 5 aaa,	Digital Value bbbbbbbbb
Bytes:	1	5	4	4	4	4	8

Telemetry Parameter Name Beacon (sent as ordinary UI Beacon Text)														
Note the different byte counts, which include commas where shown. The list may stop at any field.														
PARM.	A1 N...	A2 ,N...	A3 ,N...	A4 ,N...	A5 ,N...	B1 ,N...	B2 ,N...	B3 ,N...	B4 ,N...	B5 ,N...	B6 ,N...	B7 ,N...	B8 ,N...	
Bytes:	5	1-7	1-7	1-6	1-6	1-5	1-6	1-5	1-4	1-4	1-4	1-3	1-3	1-3



Telemetry Unit/Label Beacon (sent as ordinary UI Beacon Text)														
Note the different byte counts, which include commas where shown. The list may stop at any field.														
UNIT.	A1 U...	A2 ,U...	A3 ,U...	A4 ,U...	A5 ,U...	B1 ,L...	B2 ,L...	B3 ,L...	B4 ,L...	B5 ,L...	B6 ,L...	B7 ,L...	B8 ,L...	
Bytes:	5	1-7	1-7	1-6	1-6	1-5	1-6	1-5	1-4	1-4	1-4	1-3	1-3	1-3

Telemetry Equation Coefficients Beacon (sent as ordinary UI Beacon Text)															
The list may stop at any field. Value = a * x^2 + b*x + c															
EQNS.	A1			A2			A3			A4			A5		
	a	,b	,c	,a	,b	,c									
Bytes:	5	n	n	n	n	n	n	n	n	n	n	n	n	n	n

Telemetry Bit Sense/Project Name Beacon (sent as ordinary UI Beacon Text)										
BITS.	B1 x	B2 x	B3 x	B4 x	B5 x	B6 x	B7 x	B8 x	Project Title	
Bytes:	5	1	1	1	1	1	1	1	1	0-23

Message Format								
	Addressee		Message Text (max 67 chars)			Message ID		
	:	:				{ xxxxxx		
Bytes:	1	9	1	0-67			1	1-5

Message Acknowledgement Format					
	Addressee		ack	Message No xxxxxx	
Bytes:	1	9	1	3	1-5

General Bulletin Format						
	BLN	Bulletin ID n	xxxxxx		Bulletin Text (max 67 characters)	
Bytes:	1	3	1	5	1	0-67

Announcement Format						
	:	BLN	Announcement Identifier x	:	Announcement Text (max 67 characters)
Bytes:	1	3	1	5	1	0-67

Group Bulletin Format						
	:	BLN	Group Bulletin ID n	Group Name	:	Group Bulletin Text (max 67 characters)
Bytes:	1	3	1	5	1	0-67

National Weather Service Bulletin Format						
	:	NWS-xxxxxx	:	NWS Bulletin Text		
Bytes:	1	9	1	n		

General Query Format								
	?	Query Type	?	Target Footprint				
				Lat	r	Long	r	Radius
Bytes:	1	n	1	n	1	n	1	4

Directed Station Query Format						
	:	Addressee	:	?	Query Type	Callsign of Heard Station
Bytes:	1	9	1	1	5	0-9

Status Report Format			
	>	Time DHM z	Status Text (max 62 chars if no timestamp, or 55 chars if there is a timestamp)
Bytes:	1	7	0-62 or 0-55



Status Report Format — with Maidenhead Grid Locator						
Bytes:	Maidenhead Locator			Sym Table ID	Symbol Code	Status Text (<i>starting with a space</i>) (max 54 chars)
	GG	nn	gg			
1	2	2	2	1	1	1-54

Data with Source Path Header		
Source Path Header	Data Type ID	Rest of the original data
n	1	n

Source Path Header — “TNC-2” Format				
An asterisk follows the digipeater callsign heard.				
Source Callsign (-SSID)	>	Destination Callsign (-SSID)	0-8 Digipeaters Digipeater Callsign (-SSID)(*)	:
1-9	1	1-9	0-80	1

Source Path Header — “AEA” Format					
An asterisk follows the source or digipeater callsign heard.					
Source Callsign (-SSID)(*)	>	0-8 Digipeaters Digipeater Callsign (-SSID)(*)	>	Destination Callsign (-SSID)	:
1-10		0-80	1	1-9	1

Third-party Format		
}	Third-Party Header	Rest of the original data
Bytes:	1	n

Third Party Header — “TNC-2” format						
Source Path Header (without “unused” digipeaters, * or :)	/	Third-Party Network Identifier (“callsign”)	/	Callsign of Receiving Gateway Station (-SSID)	*	:
Bytes:	n	1	1-9	1	1-9	1 1

Third Party Header — “AEA” format								
Source Path Header (without “unused” digipeaters, destination, * or :)	/	Third-Party Network Identifier (“callsign”)	/	Callsign of Receiving Gateway Station (-SSID)	*	/	Destination Callsign from Source Path Header (-SSID)	:
Bytes:	2-90	1-9	1	1-9	1	1	1-9	1

User-Defined Data Format			
{	User ID U	User-Defined Packet Type X	User-defined data (printable ASCII recommended)
Bytes:	1	1	n

Invalid Data / Test Data Format	
/	Invalid Data or Test Data
Bytes:	1 n

Agrelo Format			
%	Bearing nnn	%	Quality n
Bytes:	1	3	1 1



APPENDIX 2: THE APRS SYMBOL TABLES

(Each highlighted character in the Alternate Symbol Table may be replaced with an overlay character).

PRIMARY SYMBOL TABLE			
/ \$	GPS xyz	GPS Cnn	Icon
/!	BB_	01	Police, Sheriff
/"	BC_	02	[reserved]
/#	BD_	03	Digipeater (white center)
/\$	BE_	04	Phone
/%	BF_	05	DX Cluster
/&	BG_	06	HF Gateway
/'	BH_	07	Small Aircraft (SSID -7)
/ (BI_	08	Cloudy
/)	BJ_	09	
/*	BK_	10	Snowmobile
/+	BL_	11	Red Cross
/,	BM_	12	Boy Scouts
/-	BN_	13	House QTH (VHF)
./	BO_	14	X
//	BP_	15	Dot
/0	P0_	16	Numerical Circle ①
/1	P1_	17	Numerical Circle ①
/2	P2_	18	Numerical Circle ②
/3	P3_	19	Numerical Circle ③
/4	P4_	20	Numerical Circle ④
/5	P5_	21	Numerical Circle ⑤
/6	P6_	22	Numerical Circle ⑥
/7	P7_	23	Numerical Circle ⑦
/8	P8_	24	Numerical Circle ⑧
/9	P9_	25	Numerical Circle ⑨
/:	MR_	26	Fire
/;	MS_	27	Campground
/<<	MT_	28	Motorcycle (SSID -10)
/=	MU_	29	Railroad Engine
/>	MV_	30	Car (SSID -9)
/?	MW_	31	Server for Files
/@	MX_	32	Hurricane Future Prediction (dot)
/A	PA_	33	Aid Station
/B	PB_	34	BBS
/C	PC_	35	Canoe

Obsolete. Use the "Circle with overlay" symbol instead (code \0).

ALTERNATE SYMBOL TABLE			
\ \$	GPS xyz	GPS Enn	Icon
\!	OB_	01	Emergency
\"	OC_	02	[reserved]
\#	ODz	03	Star (green) [with overlay]
\\$	OE_	04	Bank or ATM (green box)
\%	OF_	05	
\&	OGz	06	Diamond [with overlay]
\'	OH_	07	Crash Site
\ (OI_	08	Cloudy
\)	OJ_	09	
*	OK_	10	Snow
\+	OL_	11	Church
\,	OM_	12	Girl Scouts
\-	ON_	13	House (HF)
\.	OO_	14	Unknown/indeterminate position
\/	OP_	15	
\0	A0z	16	Circle [with overlay]
\1	A1_	17	
\2	A2_	18	
\3	A3_	19	
\4	A4_	20	
\5	A5_	21	
\6	A6_	22	
\7	A7_	23	
\8	A8_	24	
\9	A9_	25	Gas Station (blue pump)
\:	NR_	26	Hail
\;	NS_	27	Park/Picnic Area
\<	NT_	28	Advisory
\=	NU_	29	
\>	NVz	30	Car [with overlay]
\?	NW_	31	Information Kiosk (blue box with ?)
\@	NX_	32	Hurricane/Tropical Storm
\A	AAz	33	Box [with overlay]
\B	AB_	34	Blowing Snow
\C	AC_	35	Coastguard

APRS SYMBOL TABLES (continued)

(Each **highlighted** character in the Alternate Symbol Table may be replaced with an overlay character).

PRIMARY SYMBOL TABLE			
/ \$	GPS xyz	GPS Cnn	Icon
/D	PD_	36	
/E	PE_	37	Eyeball (eye catcher)
/F	PF_	38	
/G	PG_	39	Grid Square (6-character)
/H	PH_	40	Hotel (blue bed icon)
/I	PI_	41	TCP/IP
/J	PJ_	42	
/K	PK_	43	School
/L	PL_	44	
/M	PM_	45	MacAPRS
/N	PN_	46	NTS Station
/O	PO_	47	Balloon (SSID -11)
/P	PP_	48	Police
/Q	PQ_	49	
/R	PR_	50	Recreational Vehicle (SSID -13)
/S	PS_	51	Space Shuttle
/T	PT_	52	SSTV
/U	PU_	53	Bus (SSID -2)
/V	PV_	54	ATV
/W	PW_	55	National Weather Service Site
/X	PX_	56	Helicopter (SSID -6)
/Y	PY_	57	Yacht (sail boat) (SSID -5)
/Z	PZ_	58	WinAPRS
/[HS_	59	Jogger
/\	HT_	60	Triangle (DF)
/]	HU_	61	PBBS
/^	HV_	62	Large Aircraft
/_	HW_	63	Weather Station (blue)
/`	HX_	64	Dish Antenna
/a	LA_	65	Ambulance (SSID -1)
/b	LB_	66	Bike (SSID -4)
/c	LC_	67	
/d	LD_	68	Dual Garage (Fire Department)
/e	LE_	69	Horse (equestrian)
/f	LF_	70	Fire Truck (SSID -3)

ALTERNATE SYMBOL TABLE			
\ \$	GPS xyz	GPS Enn	Icon
\D	AD_	36	Drizzle
\E	AE_	37	Smoke
\F	AF_	38	Freezing Rain
\G	AG_	39	Snow Shower
\H	AH_	40	Haze
\I	AI_	41	Rain Shower
\J	AJ_	42	Lightning
\K	AK_	43	Kenwood
\L	AL_	44	Lighthouse
\M	AM_	45	
\N	AN_	46	Navigation Buoy
\O	AO_	47	
\P	AP_	48	Parking
\Q	AQ_	49	Earthquake
\R	AR_	50	Restaurant
\S	AS_	51	Satellite/PACsat
\T	AT_	52	Thunderstorm
\U	AU_	53	Sunny
\V	AV_	54	VORTAC Nav Aid
\W	AWz	55	NWS Site (Options) [with overlay]
\X	AX_	56	Pharmacy Rx
\Y	AY_	57	
\Z	AZ_	58	
\[DS_	59	Wall Cloud
\]	DT_	60	
\^	DU_	61	
\^	DVz	62	Aircraft [with overlay]
_	DWz	63	Weather Site (green digi) [w/ ov'lay]
\`	DX_	64	Rain
\a	SA_	65	ARRL ARES etc
\b	SB_	66	Blowing Dust/Sand
\c	SCz	67	Civil Defense (RACES) [w/ overlay]
\d	SD_	68	DX Spot by Callsign
\e	SE_	69	Sleet
\f	SF_	70	Funnel Cloud



APRS SYMBOL TABLES (continued)

(Each **highlighted** character in the Alternate Symbol Table may be replaced with an overlay character).

PRIMARY SYMBOL TABLE			
/s	GPS xyz	GPS Cnn	Icon
/g	LG_	71	Glider
/h	LH_	72	Hospital
/i	LI_	73	IOTA (Island on the Air)
/j	LJ_	74	Jeep (SSID -12)
/k	LK_	75	Truck (SSID -14)
/l	LL_	76	
/m	LM_	77	Mic-repeater
/n	LN_	78	Node
/o	LO_	79	Emergency Operations Center
/p	LP_	80	Rover (puppy dog)
/q	LQ_	81	Grid Square shown above 128m
/r	LR_	82	Antenna
/s	LS_	83	Ship (power boat) (SSID -8)
/t	LT_	84	Truck Stop
/u	LU_	85	Truck (18-wheeler)
/v	LV_	86	Van (SSID -15)
/w	LW_	87	Water Station
/x	LX_	88	X-APRS (Unix)
/y	LY_	89	Yagi at QTH
/z	LZ_	90	
{	J1_	91	
	J2_	92	[Reserved — TNC Stream Switch]
}	J3_	93	
~	J4_	94	[Reserved — TNC Stream Switch]

ALTERNATE SYMBOL TABLE			
\s	GPS xyz	GPS Enn	Icon
\g	SG_	71	Gale Flags
\h	SH_	72	Ham Store
\i	SI _z	73	Indoor short range digi [w/ overlay]
\j	SJ_	74	Work Zone (steam shovel)
\k	SK_	75	
\l	SL_	76	Area Symbols (box, circle, etc)
\m	SM_	77	Value Signpost {3-char display}
\n	SN _z	78	Triangle [with overlay]
\o	SO_	79	Small Circle
\p	SP_	80	Partly Cloudy
\q	SQ_	81	
\r	SR_	82	Restrooms
\s	SS _z	83	Ship/Boat (top view) [with overlay]
\t	ST_	84	Tornado
\u	SU _z	85	Truck [with overlay]
\v	SV _z	86	Van [with overlay]
\w	SW_	87	Flooding
\x	SX_	88	
\y	SY_	89	
\z	SZ_	90	
\{	Q1_	91	Fog
\	Q2_	92	[Reserved — TNC Stream Switch]
\}	Q3_	93	
\~	Q4_	94	[Reserved — TNC Stream Switch]

APPENDIX 3: ASCII CODE TABLE

In addition to listing the ASCII character codes in their usual form, this table also expresses the hexadecimal codes for the ASCII digits 0–9 and the upper-case letters A–Z in *shifted* form; i.e. shifted one bit left. This is particularly useful for decoding callsigns and Mic-E position information contained in the address fields of AX.25 frames.

Part 1: Codes 0–31 decimal (00–1f hexadecimal)

<i>Dec</i>	<i>Hex</i>	<i>Char</i>		
0	00	NUL	CTRL-@	
1	01	SOH	CTRL-A	Start of Header
2	02	STX	CTRL-B	Start of Text
3	03	ETX	CTRL-C	End of Text
4	04	EOT	CTRL-D	End of Transmission
5	05	ENQ	CTRL-E	Enquiry (Poll)
6	06	ACK	CTRL-F	Acknowledge
7	07	BEL	CTRL-G	Bell
8	08	BS	CTRL-H	Backspace
9	09	HT	CTRL-I	Horizontal Tab
10	0a	LF	CTRL-J	Line Feed
11	0b	VT	CTRL-K	Vertical Tab
12	0c	FF	CTRL-L	Form Feed
13	0d	CR	CTRL-M	Carriage Return
14	0e	SO	CTRL-N	Shift Out
15	0f	SI	CTRL-O	Shift In
16	10	DLE	CTRL-P	Data Link Escape
17	11	DC1/XON	CTRL-Q	Device Control 1
18	12	DC2	CTRL-R	Device Control 2
19	13	DC3/XOFF	CTRL-S	Device Control 3
20	14	DC4	CTRL-T	Device Control 4
21	15	NAK	CTRL-U	Negative Acknowledge
22	16	SYN	CTRL-V	Synchronous Idle
23	17	ETB	CTRL-W	End of Transmission Block
24	18	CAN	CTRL-X	Cancel
25	19	EM	CTRL-Y	End of Medium
26	1a	SUB	CTRL-Z	Substitute
27	1b	ESC	CTRL-[Escape
28	1c	FS	CTRL-\	File Separator
29	1d	GS	CTRL-]	Group Separator
30	1e	RS	CTRL-^	Record Separator
31	1f	US	CTRL-_ CTRL-^	Unit Separator



Part 2: Codes 32–127 decimal (20–7f hexadecimal), including hex codes for shifted 0–9/A–Z

Dec	Hex	Char	Shifted
32	20	␣	40/41 (space)
33	21	!	
34	22	"	(inv commas)
35	23	#	
36	24	\$	
37	25	%	
38	26	&	
39	27	'	(apostrophe)
40	28	(
41	29)	
42	2a	*	
43	2b	+	
44	2c	,	(comma)
45	2d	-	(minus)
46	2e	.	(dot)
47	2f	/	
48	30	0	60/61
49	31	1	62/63
50	32	2	64/65
51	33	3	66/67
52	34	4	68/69
53	35	5	6a/6b
54	36	6	6c/6d
55	37	7	6e/6f
56	38	8	70/71
57	39	9	72/73
58	3a	:	
59	3b	;	
60	3c	<	
61	3d	=	
62	3e	>	
63	3f	?	
64	40	@	
65	41	A	82/83
66	42	B	84/85
67	43	C	86/87
68	44	D	88/89
69	45	E	8a/8b
70	46	F	8c/8d
71	47	G	8e/8f
72	48	H	90/91
73	49	I	92/93
74	4a	J	94/95
75	4b	K	96/97
76	4c	L	98/99
77	4d	M	9a/9b
78	4e	N	9c/9d
79	4f	O	9e/9f

Dec	Hex	Char	Shifted
80	50	P	a0/a1
81	51	Q	a2/a3
82	52	R	a4/a5
83	53	S	a6/a7
84	54	T	a8/a9
85	55	U	aa/ab
86	56	V	ac/ad
87	57	W	ae/af
88	58	X	b0/b1
89	59	Y	b2/b3
90	5a	Z	b4/b5
91	5b	[
92	5c	\	
93	5d]	
94	5e	^	
95	5f	_	(underscore)
96	60	`	(grave accent)
97	61	a	
98	62	b	
99	63	c	
100	64	d	
101	65	e	
102	66	f	
103	67	g	
104	68	h	
105	69	i	
106	6a	j	
107	6b	k	
108	6c	l	
109	6d	m	
110	6e	n	
111	6f	o	
112	70	p	
113	71	q	
114	72	r	
115	73	s	
116	74	t	
117	75	u	
118	76	v	
119	77	w	
120	78	x	
121	79	y	
122	7a	z	
123	7b	{	
124	7c		
125	7d	}	
126	7e	~	
127	7f	DEL	



Part 3: Codes 128–255 decimal (80–ff hexadecimal)

<i>Dec</i>	<i>Hex</i>
128	80
129	81
130	82
131	83
132	84
133	85
134	86
135	87
136	88
137	89
138	8a
139	8b
140	8c
141	8d
142	8e
143	8f
144	90
145	91
146	92
147	93
148	94
149	95
150	96
151	97
152	98
153	99
154	9a
155	9b
156	9c
157	9d
158	9e
159	9f

<i>Dec</i>	<i>Hex</i>
160	a0
161	a1
162	a2
163	a3
164	a4
165	a5
166	a6
167	a7
168	a8
169	a9
170	aa
171	ab
172	ac
173	ad
174	ae
175	af
176	b0
177	b1
178	b2
179	b3
180	b4
181	b5
182	b6
183	b7
184	b8
185	b9
186	ba
187	bb
188	bc
189	bd
190	be
191	bf

<i>Dec</i>	<i>Hex</i>
192	c0
193	c1
194	c2
195	c3
196	c4
197	c5
198	c6
199	c7
200	c8
201	c9
202	ca
203	cb
204	cc
205	cd
206	ce
207	cf
208	d0
209	d1
210	d2
211	d3
212	d4
213	d5
214	d6
215	d7
216	d8
217	d9
218	da
219	db
220	dc
221	dd
222	de
223	df

<i>Dec</i>	<i>Hex</i>
224	e0
225	e1
226	e2
227	e3
228	e4
229	e5
230	e6
231	e7
232	e8
233	e9
234	ea
235	eb
236	ec
237	ed
238	ee
239	ef
240	f0
241	f1
242	f2
243	f3
244	f4
245	f5
246	f6
247	f7
248	f8
249	f9
250	fa
251	fb
252	fc
253	fd
254	fe
255	ff



APPENDIX 4: GLOSSARY

Altitude	1. In Mic-E format, the altitude in meters relative to 10km below mean sea level. 2. In Comment text, the altitude in feet above mean sea level.
Announcement	An APRS message that is repeated a few times an hour, perhaps for several days.
Announcement Identifier	A single letter A-Z that identifies a particular announcement.
Antenna Height	In NMEA sentences, the height of the antenna in meters relative to mean sea level. (The antenna height in GPS NMEA sentences fluctuates wildly because of Selective Availability, and should only be used if DGPS correction is applied).
APRS	Automatic Position Reporting System.
APRS Data	The data that follows the APRS Data Type Identifier in the AX.25 Information field and precedes the APRS Data Extension.
APRS Data Extension	A 7-byte extension to APRS Data. The Data Extension includes one of Course/Speed, Wind Direction/Wind Speed, Station Power/Antenna Effective Height/Gain/Directivity, Pre-Calculated Radio Range, DF Signal Strength/Effective Antenna Height/Gain, Area Object Descriptor.
APRS Digipeater Path	A digipeater path via repeaters with RELAY, WIDE and related aliases. Used in Mic-E compressed location format.
APRS Data Type Identifier	The single-byte identifier that specifies what kind of APRS information is contained in the AX.25 Information field.
Area Object	A user-defined graphic object (circle, ellipse, triangle, box and line).
AX.25	Amateur Packet-Radio Link-Layer Protocol.
Base 91	Number base used to ensure that numeric values are transmitted as printable ASCII characters. To obtain the character string corresponding to a numeric value, divide the value progressively by decreasing powers of 91, and add 33 decimal to the result at each step. Printable characters are in the range !..f. Used in compressed lat/long and altitude computation.
Bulletin	An APRS message that is repeated several times an hour, for a small number of hours. A General Bulletin is addressed to no-one in particular. A Group Bulletin is addressed to a named group (e.g. WX).
Bulletin Identifier	A single digit 0-9 that identifies a particular bulletin.
Destination Address field	The AX.25 Destination Address field, which can contain an APRS destination callsign or Mic-E encoded data.
DF Report	A report containing DF bearing and range.
DGPS	Differential GPS. Used to overcome the errors arising from Selective Availability.
DHM	7-character timestamp: day-of-the-month, hour, minute, zulu or local time.
DHMz	7-character timestamp: day-of-the-month, hour, minute, zulu only.
Digipeater	A station that relays AX.25 packets. A chain of up to 8 digipeaters may be specified.
Digipeater Addresses field	The AX.25 field containing 0-8 digipeater callsigns (or aliases).
Directivity	The favored direction of an antenna. Used in the PHG Data Extension.
DX Cluster	A network host that collects and disseminates user reports of DX activity.
ECHO	A generic APRS digipeater callsign alias, for an HF digipeater.
Effective Antenna Height	The height of an antenna above the local terrain (not above sea level). A first-order indicator of the antenna's effectiveness in the local area. Used in the PHG Data Extension.



FCS	Frame Check Sequence. A sequence of 16 bits that follows the AX.25 Information field, used to verify the integrity of the packet.
GATE	A gateway between HF and VHF APRS networks. Used primarily to relay long-distance HF APRS traffic onto local VHF networks.
GGA Received Sentence	A standard NMEA sentence, containing the receiving station's lat/long position and antenna height relative to mean sea level, and other data.
GLL Received Sentence	A standard NMEA sentence, containing the receiving station's lat/long position and other data.
GMT	Greenwich Mean Time (=UTC=zulu).
GPS	Global Positioning System. A global network of 24 satellites that provide lat/long and antenna height of a receiving station.
GPSxyz	An APRS destination callsign that specifies a display symbol from either the Primary Symbol Table or the Alternate Symbol Table. Some symbols from the Alternate Symbol Table can be overlaid with a digit or a letter. Used by trackers that cannot specify the symbol in the AX.25 Information field.
GPSCnn	An APRS destination callsign that specifies a display symbol from the Primary Symbol Table. The symbol can not be overlaid. Used by trackers that cannot specify the symbol in the AX.25 Information field.
GPSEnn	An APRS destination callsign that specifies a display symbol from the Alternate Symbol Table. The symbol can not be overlaid. Used by trackers that cannot specify the symbol in the AX.25 Information field.
HMS	1. In NMEA sentences, a 6-character timestamp: hour, minute, second UTC. 2. In APRS Data, a 7-character timestamp: hour, minute, second, zulu or local.
ICQ	International CQ chat.
IGate	A gateway between a VHF and/or HF APRS network and the Internet.
Information field	The AX.25 Information field containing APRS information.
Item	A type of display object.
Item Report	A report containing the location of an APRS Item.
Killed Object	An Object that an APRS user has assumed control of.
knots	International nautical miles per hour (mph = knots * 1.151).
KPC-3	A Terminal Node Controller from Kantronics Co Inc.
Longitude Offset	An offset of +100 degrees longitude (used in Mic-E longitude computation).
LORAN	Long Range Navigation System (a terrestrial precursor to GPS).
Maidenhead Locator	A 4- or 6-character grid locator specifying a station's position.
MDHM	8-byte timestamp: month, day, hour, minute (used in stand-alone weather station reports).
Message	A one-line text message addressed to a particular station.
Message Acknowledgement	An optional acknowledgement of receipt of a message.
Message Group	A user-defined group to receive messages.
Message Identifier	A 1–5 character message identifier (typically a line number).
Mic-E	Originally Microphone Encoder, a unit that encodes location, course and speed information into a very short packet, for transmission when releasing the microphone PTT button. The Mic-E encoding algorithm is now used in other devices (e.g. the Kenwood TH-D7).
Mic-E Message Identifier	A 3-bit identifier (A/B/C) specifying a standard Mic-E message or custom message code.



Mic-E Message Code	A 3-bit code specifying a standard Mic-E message.
MIM	Micro Interface Module. A complete telemetry TNC transmitter on a chip.
mph	miles per hour (km/hour = mph * 1.60935. m/sec = mph * 0.447).
(International) Nautical Mile	6076.103 feet / 1852 meters.
Net Cycle Time	The time within which it should be possible to gain the complete picture of APRS activity (typically 10, 20 or 30 minutes, depending on the number of digipeaters traversed and local conditions). Stations should not transmit status or position information more frequently unless mobile, or in response to a Query.
NMEA	National Marine Electronic Association (United States). Producer of the <i>NMEA 0183 Version 2.0</i> specification that governs the format of Received Sentences from navigation equipment (such as GPS and LORAN receivers).
NMEA (Received) Sentence	The ASCII data stream received from navigation equipment conforming to the NMEA 0182 Version 2.0 specification. APRS supports four Received Sentences: GGA, GLL, RMC and VTG.
NRQ	Number/Rate/Quality. A measure of confidence in DF Bearing reports.
Null Position	Default position to be reported if the actual position is unknown or indeterminate. The null position is 0° 0' 0" north, 0° 0' 0" west.
NWS	National Weather Service (United States).
Object	A display object that is (usually) not a station. For example, a weather front or a marathon runner.
Object Report	A report containing the position of an object, with optional timestamp and APRS Data Extension.
PHG	APRS Data Extension specifying Power, Effective Antenna Height/Gain/Directivity.
Position Ambiguity	A reduction in the accuracy of APRS position information (implemented by replacing low-order lat/long digits with spaces). Used when the exact position is not known.
Position Report	A report containing lat/long position, optionally with timestamp and Data Extension.
Pre-Calculated Radio Range	A station's estimate of omni-directional radio range (in miles). Used in compressed lat/long format.
Query	A request for information. Queries may be addressed to stations in general or to specific stations.
Range Circle	Usable radio range (in miles), computed from PHG data.
RELAY	A generic APRS digipeater callsign alias, for a VHF/UHF digipeater with limited local coverage.
Response	A reply to a query.
RMC Received Sentence	A standard NMEA sentence, containing the receiving station's lat/long position, course and speed, and other data.
Selective Availability	Deliberate GPS position dithering, introducing significant received position errors in latitude, longitude <i>and</i> antenna height. Errors can be greatly reduced with differential GPS.
Sentence	See NMEA (Received) Sentence.
Signpost	A special signpost icon that displays user-defined variable information (such as a speed limit or mileage) as an overlay.
Skywarn	A weather spotter program coordinated by the United States National Weather Service.
Source Address Field	The AX.25 Source Address field, containing the callsign of the originating station. A non-zero SSID specifies a display symbol.
Source Path Header	The digipeater path followed prior to a packet entering a Third-Party Network.
SPCL	A generic APRS destination callsign used for special stations.



SSID	Secondary Station Identifier. A number in the range 0-15, as an adjunct to an AX.25 address. If the SSID in a source address is non-zero, it specifies a display symbol. (This is used when the station is unable to specify the symbol in the AX.25 Destination Address field or Information field).
Station Capabilities	A list of station characteristics that is sent in reply to a query.
Status Report	A report containing station status information (and optionally a Maidenhead locator).
Switch Stream Character	A character normally used for switching TNC channels.
Symbol	A display icon. Consists of a Symbol Table Identifier/Symbol Code pair. Generically, /S represents a symbol from the Primary Symbol Table, and \S represents a symbol from the Alternate Symbol Table.
Symbol Code	A code for a symbol within a Symbol Table.
Symbol Table Identifier	An ASCII code specifying the Primary Symbol Table (/) or Alternate Symbol Table (\). The Symbol Table Identifier is also implicit in GPSCnn and GPSEnn destination callsigns.
Target Footprint	A target area for queries. The querying station asks for responses from stations within a specified number of miles of a lat/long position.
TH-D7	A combined VHF/UHF handheld radio and APRS-compatible TNC from Kenwood.
Third Party Network	A non-APRS network that does not understand AX.25 addresses (e.g. the Internet).
Third-Party Header	A Path Header with the Third-Party Network Identifier and the callsign of the receiving gateway inserted.
TNC	Terminal Node Controller. A combined AX.25 packet assembler/disassembler and modem.
Trace	An APRS query that asks for the route taken by a packet to a specified station.
TRACE	A generic digipeater callsign alias, for digipeaters that performs callsign substitution. These digipeaters self-identify packets they digipeat, by inserting their own callsign in place of RELAY,WIDE or TRACE.
Tunneling	Passing APRS AX.25 traffic through a third-party network that does not understand AX.25 addressing. The AX.25 addresses are carried as data (in the Source Path Header) through the tunneled network.
UI-Frame	AX.25 Unnumbered Information frame. APRS uses only UI-frames — that is, it operates entirely in connectionless (UNPROTO) mode.
UNPROTO Path	The digipeater path to the destination callsign.
UTC	Coordinated Universal Time (=zulu=GMT).
VTG Received Sentence	A standard NMEA sentence, containing the receiving station's course and speed.
WIDE	A generic APRS digipeater callsign alias, for a digipeater with wide area coverage.
WIDEn-N	A generic APRS digipeater callsign alias, for a digipeater with wide area coverage (N=0-7). As a packet passes through a digipeater, the value of N is decremented by 1 until it reaches zero. The digipeater keeps a record of each packet (or its FCS) as it passes through, and will not digipeat the packet again if it has digipeated it already within the last 28 seconds.
WX	Weather.
Ziplan	A cheap twisted-pair LAN connecting PCs via their serial I/O ports. Designed for use in emergency situations.
Zulu	UTC/GMT.



APPENDIX 5: REFERENCES

AX.25 Amateur Packet-Radio Link-Layer Protocol Version 2.0, October 1984, at <http://www.tapr.org/tapr/html/ax25.html>.

NMEA 0183 ASCII Interface Specification, at <http://www.nmea.org/0183.htm>.

END OF DOCUMENT

