AMATEUR SATELLITE TELEMETRY: PAST, PRESENT and FUTURE

Joe Kasser W3/G3ZCZ
Loral AeroSys, 4600 Powder Mill Rd., Beltsville, MD., 20705

Contents

Abstract

Starting with the first Orbiting Satellite Carrying Amateur Radio (OSCAR) in 1961, each time an OSCAR has been launched the spacecraft designers have developed their own telemetry systems. The result has been a different downlink data format for each spacecraft. As a consequence, the developers had to re-invent the data transfer process and the users were faced with the problem of capturing and processing data in different formats. With the anticipated launch of between five and ten more spacecraft within the next five years, should they also use different formats, life will get very complicated for ground stations planning to acquire and process telemetry.

This paper examines the telemetry from the existing satellites, discusses telemetry from a systems perspective and looks at requirements for all parts of the system including spacecraft builders, telemetry users and ground station software developers. The paper then introduces two proposed amateur telemetry standards, the first for downlinking of data, the second for interchanging data between different computers and long term archival of the data.

Spacecraft in Orbit

The earliest of the amateur spacecraft designed to downlink digital telemetry was the first Phase 3A satellite, built by the Radio Amateur Satellite Corporation (AMSAT), which was designed in the middle 1970's. At that time personal computers were not on the market. Anyone who wanted one had to build it from scratch or from the few kits such as the AMSAT-GOLEM-80 Project[1,2] that did exist. Most radio amateurs who were interested in digital communications used mechanical (noisy) teleprinters; a very few lucky ones had cathode ray tube (CRT) terminals.

The AMSAT Phase 3A satellite failed to achieve orbit because the Ariane launch vehicle malfunctioned a minute or so after leaving the ground and the range safety officer activated the self destruct procedure[3]. Subsequent Phase 3 satellites, AMSAT-OSCAR (AO) 10 and AO-13, launched in 1983 and 1988 respectively, used the same data formats and downlinked 50 baud BAUDOT and 400 baud ASCII telemetry. The downlink format was somewhat similar to the fixed frames of morse code telemetry generated by AO-6, AO-7 and AO-8. The 50 baud telemetry used standard amateur radio teletypewriter (RTTY) frequency shift keyed (FSK) formats so that any radio amateur suitably equipped could receive the data. The received data display shown in Figure 1 is simple and reception errors can be readily identified. The 400 baud telemetry was mainly intended for use by the command stations and used phase shift keying (PSK) to provide what was then (1975-1980) high speed data over long distance radio links[4]. The 400 baud data downlink format of blocks (similar to packets) was designed to be displayed on home made personal computers (PC) using crt displays having 16 lines of 64 characters per line.

Figure 1. Sample AO-13 RTTY Z Block Transfer Frames

```
Z HI. THIS IS AMSAT OSCAR 13

05.02.54 8661
.0086 .0000 .07B9
64 6 0 1 16 218 1

193 170 158 143 181 144 147 140 200 7
147 7 7 7 165 29 100 7 149 7
10 7 145 115 34 7 153 129 122 180
152 73 7 145 137 55 7 183 136 151
7 154 137 169 211 142 127 100 9 140
161 7 173 149 150 154 14 131 127 210

HI THIS IS AMSAT OSCAR 13 08SEP90
NEW AO13 SCHEDULE FROM 17OCT90 AFTER MOVE TO LON 180 LAT 0
MODE B MA 000 TO 095
MODE JL MA 095 TO 125
MODE LS MA 125 TO 130
MODE S MA 130 TO 135
MODE BS MA 135 TO MA 140
MODE B MA 140 TO 256
```

The satellites built at the University of Surrey, England increased the downlink telemetry rate to 1200 baud. UoSAT-1 (UO-1), launched on 1981 and UoSAT-2 (UO-2) launched in 1983 sent back 1200 baud ASCII telemetry in a number of formats. A basic received real-time ASCII telemetry frame is shown in Figure 2. This format is more complex than the Phase 3 BAUDOT format and reception errors are not as readily identified. Consequently, the UoSATs were the first to introduce a checksum into the telemetry data to allow the receiver to verify that the data was good.

Figure 2 UoSAT-2 ASCII Telemetry Data Transfer Frame

```
00519D01413702676503614004046605;4 6019E07045608040C08036C
10519C11298312000313056114069A15529A!6188;175452185905195058
20519F21220322662223000124001725000726093E27541528564D294681
30519E31041732287C33568B34007035217226B39455E
40649F41117242647343061044162545000146000247444748454949422x
50456251108D52634653284p54663215000056p00357451258447A59460E
60826A615FC1625F4A63334164440265160466174267700668000E69000F
UOSAT-2 9101281004625
```

Note : Errors due to noise.

Terrestrial amateur packet radio experiments began in 1980 and an international network grew rapidly once an amateur adaptation of the X.25 standard (AX.25) was adopted[5]. The Japanese AMSAT built Fuji-OSCAR 12 (FO-12) (launched in 1986) was the first amateur satellite to incorporate packet radio. FO-12 and FO-20 (launched in 1990), while using packet radio also used a fixed format as shown in Figure 3. No checksum was needed in the data, because the error detection was performed by the link protocol. All good packets received, were error free by definition.

Figure 3 FO-20 Telemetry Data Transfer Frame

```
05-Jun-91 09:43:35 8J1JBS*>BEACON:
JAS1b RA 91/06/05 09:39:58
493 481 688 691 854 839 850 833 002 746
615 000 418 453 457 448 451 454 651 000
683 681 745 713 999 643 874 385 1BE 000
010 111 011 000 111 100 001 100 111 000
```

In 1990 AMSAT achieved an impressive milestone with the launch of six spacecraft as secondary payloads on an Ariane vehicle. With one stroke, two more UoSATs and four digital satellites were launched (UO-3 and UO-4, OSCARs 16, 17, 18 and 19). Each downlinked telemetry using packet radio. UO-4 malfunctioned soon after launch and has not been heard from since. UO-3 swiftly changed its downlink format to 9600 baud binary.

AMSAT's microsats, AO-16, DOVE- OSCAR 17 (DO-17), Weber-OSCAR 18 (WO-18) and Lusat-OSCAR 19 (LO-19), downlinked telemetry at 1200 baud in an ASCII format using packet radio. DO-17's signals can be received by anyone who has a 2 meter packet radio system. A typical example is shown in Figure 4. This format uses a number of transfer frames to downlink a source frame of data and mixes ASCII text and hexadecimal data.

Figure 4 DO-17 Telemetry Data Transfer Frames

```
23-Jan-91 02:49:23 DOVE-1*>TIME-1:
PHT: uptime is 173/00:36:26. Time is Wed Jan 23 02:47:30 1991

23-Jan-91 02:49:26 DOVE-1*>TLM:
00:59 01:59 02:87 03:31 04:59 05:5A 06:6E 07:52 08:6D 09:72 0A:A2
0B:DC 0C:E9 0D:D8 0E:02 0F:26 10:CC 11:A8 12:01 13:04 14:AD 15:94
16:98 17:94 18:96 19:98 1A:94 1B:91 1C:9B 1D:98 1E:25 1F:5F 20:BA

23-Jan-91 02:49:27 DOVE-1*>TLM:
21:95 22:82 23:24 24:1E 25:2A 26:01 27:02 28:02 29:01 2A:02 2B:02
2C:01 2D:29 2E:02 2F:9E 30:CA 31:9E 32:11 33:CE 34:C4 35:9A 36:A8
37:A6 38:B6

23-Jan-91 02:49:28 DOVE-1*>STATUS:
80 00 00 8F 00 18 CC 02 00 B0 00 00 0C 0E 3C 05 0B 00 04 04

23-Jan-91 02:49:28 DOVE-1*>LSTAT:
I P:0x3000 o:0 l:13081 f:13081, d:0

23-Jan-91 02:49:28 DOVE-1*>WASH:
wash addr:0680:0000, edac=0xd6
```

The major improvement introduced by the microsats is twofold. All four satellites used the same downlink data format allowing a single table driven data acquisition, decode and display program to be used in the ground station. Their telemetry channel information is transmitted together with the data, and this format provides a great deal of flexibility. For the first time, the potential to change the order of data on the downlink came into existence. For example, the same channel or a particular group of channels could be sampled more often. The fact would be obvious to anyone watching the downlink.

AO-16, WO-18 and LO-19, the microsats with downlinks in the 437 MHz band switched to an unpublished binary downlink format a month or so after UO-4 with little prior notice. More than a year after the event, AMSAT still have not published this binary downlink format.

AMSAT's Philosophy

In its earliest days, AMSAT had a policy of advancing the state-of-the-art while at the same time trying not to obsolete ground station equipment[6]. Each of the satellites provided a downlink using technology compatible to that existing in amateur radio stations at that time, and also introduced something new. Thus the Phase 3 satellites carried 50 baud BAUDOT RTTY even as late as 1988 to allow reception by amateurs equipped for RTTY.

The UoSAT downlinks were designed to be compatible with a terrestrial audio cassette data storage standard[7]. This use of a standard allowed radio amateurs having suitable devices to connect them to their radio receivers and display the data received from the spacecraft. The advent of low cost floppy disk drives provided random access off-line data storage and effectively killed the tape standard as a storage approach before it came into very widespread use. SARA-OSCAR 23's (O-23) digital downlink telemetry modulation complies with the same audio standard. A similar approach of using a USSR tape data storage standard was adopted for the digital PSK telemetry of AO-21 built in the Soviet Union and launched as an attached secondary payload to a USSR geological research satellite in 1992[8].

FO-12 and FO-20 carried packet radio, but required the prospective user to obtain a PSK modem to be able to receive the digital signals. The user however was still able to use the basic AX.25 software in the terminal node controller (TNC). The AMSAT microsats also used 1200 baud PSK compatible to FO-12 and FO-20.

Historically, the lack of standards has resulted in the builders of each spacecraft having to develop a unique set of ground processing equipment to perform telemetry processing functions, though the functions themselves are basically identical. A major benefit of implementing standards is the development of ground data-handling facilities that can be rapidly reconfigured (using look-up tables) to meet the requirements of additional terrestrial and orbiting missions.

The Future

Table 1 contains a list of announced future OSCARs. Some of these spacecraft, notably ARSENE, are finished and in the testing stage, others such as the SEDSAT[9] and the World Space Foundation's Lunar Solar Sail project[10] are under development and time may still permit programming their OBCs to make their telemetry conform to a downlink telemetry standard.

Table 1 Amateur Radio Spacecraft Currently Under Development

SPACECRAFT	COUNTRY	MISSION	LAUNCH DATE
ARSENE	France	Long Life Intercontinental AX.25 Communications	1992
KITSAT	Korea	Educational Construction Project	1992
SEDSAT	USA	Educational Construction Project	1993
Solar Sail	USA	World Space Foundation Project	1993
TECHSAT	Israel	Educational Construction Project	1993
IT-AMSAT	Italy	Similar to AO-16 with science experiment.	1994
SUNSAT	S. Africa	Educational Construction Project	1994
MARS	Germany and an International Team	Interplanetary probe and long range communications relay	1995
AMSAT-PHASE	Germany and an	Long Life Intercontinental Communications	1995

3D	International Team		

Telemetry Terminology

The spacecraft on-board computer (OBC) sequentially measures spacecraft housekeeping and payload voltages, currents and temperatures at various points in the spacecraft and arranges the data in a frame in a predefined format. The housekeeping information is used to maintain the health and safety of the spacecraft. This information pertains to the battery voltage, solar array currents and internal temperatures. The payload information may be the science data obtained from on-board experiments or communications transponder and other payload data.

The data are transmitted to earth through the weak signal, noisy space channel as a serial, synchronous symbol stream on the telemetry beacon downlink. Fixed length data frames are used to allow for simple, robust and reliable frame synchronization processes at the receiving ends. The frame boundaries are indicated using Synchronization Markers. These synchronization techniques have performed reliably in virtually hundreds of professional as well as all amateur space mission applications. The received telemetry data frames are then stored in ground station computers. Telemetry that is downlinked within seconds of being collected is known as real-time telemetry.

Source Data Frames

When the OBC performs a set of measurements, it stores the results in source data frames. A source data frame may be a telemetry collection sequence, a set of WOD, a data file or a picture taken by an onboard camera.

Transfer Frames

The source data frame is transmitted to earth in transfer frames on the downlink. There does not have to be a 1:1 correspondence between the source data frame and the transfer frame. For example, if the 'Q' Blocks of PSK telemetry from AO-13 are thought of as being transfer frames containing source data frames with a 1:1 correspondence, the 'Z' blocks of RTTY telemetry are transfer frames containing a subset of the source data frame. Each UO-2 and FO-20 transfer frame contain a single source data frame. Large source data frames can be split up into a number of segments and downlinked in transfer frames. DO-17 uses several transfer frames to downlink segments of the source data frame. WO-18 downlinks WOD and pictures in a number of transfer frames.

Acquisition Sessions

An individual ground station receives telemetry during an acquisition session. Once the data is captured by the ground station it is stored in the computer. An acquisition session begins when the first frame of data is captured and ends following the capturing of the last frame of data. These times correspond somewhat to acquisition of signals (AOS) and loss of signals (LOS).

Types of Telemetry

There are two types of amateur satellite telemetry; public and private. Public telemetry is available to anyone who has the equipment to either capture the data during an acquisition session, or read data files produced by other ground stations. Public telemetry consists of all the data that the spacecraft builders make the decoding information readily available. It is most of the housekeeping and much of the science data.

Private telemetry on the other hand consists of command acknowledgements, temporary telemetry data only downlinked during specific tests and any other information not available to the general public. Private telemetry is normally used by the command stations, spacecraft support personnel and other authorized people.

Whole Orbit Data (WOD)

Not all telemetry is downlinked in real-time. Low earth orbiting satellites are out of range of any single command station for most of the time. In order to obtain data collected over a whole orbit, the command station has to instruct the spacecraft to record data measurements as well as downlinking them on the telemetry beacon. The recorded data are then played back when the spacecraft passes in range of the command station. The National Aeronautics and Space Administration's (NASA) spacecraft record data measurements using magnetic tape recorders. When a tape playback is commanded, the machine is rewound and the data are read from the tape and downlinked at the same time. This playback data is thus both downlinked at a higher speed than the real-time data and is reversed. The UOSATs pioneered an alternative approach in the OSCAR program by storing measured data (for a few selected channels) in the solid state memory in the OBC. The stored data now named WOD, are later downlinked in forward order using a special WOD format.

Virtual Channels

Packetized telemetry shares the physical downlink channel with other types of packets. For example the FO-20 telemetry shares the channel with the Bulletin Board System (BBS) downlink. The AO-16 and LO-19 telemetry share the channel with files, messages and announcements. This physical channel sharing concept can be expressed as providing each service with a virtual channel on the same physical channel. Ground station software distinguishes between the virtual channels by means of the packet headers. In the future, it is possible that several spacecraft may share the same downlink frequency for packet telemetry and the virtual channel concept will come into its own. Note that at present, UO-2 (ASCII), DO-17 (AX.25) and AO-21 (CW) share 145.825 MHz.

If a telemetry beacon uses a virtual channel on a physical channel shared with the communications payload (the primary mission), as in AO-16, LO-19 and FO-20, then intuitively, the spacecraft should maximize the amount of time that it is downlinking communications and minimize the amount of time spent transmitting telemetry. This factor also tends to suggest the use of binary telemetry since more data can be transmitted in a fixed time.

Data Products

Telemetry data can be considered in three categories, Real-time, Quick Look and Archive. Real-time data are produced by a spacecraft and received by a ground station during an acquisition session. The purpose of real-time data is mainly to check the spacecraft; verify the health and welfare of the on-board equipment and monitor trends. Ground stations produce Quick Look and Archive data. Quick look data are ground station playbacks of the real-time data, used shortly after an acquisition session to reexamine something. Archive data are used days, weeks or even years later in some long term correlation or other kind of analysis. Today's radio amateurs do not normally exchange data after an acquisition session, so there is no perceived difference between Quick Look and Archive data.

Goals of the Telemetry System

The telemetry system is more than just the spacecraft. It contains the space-craft, the communications link and the ground station. The goals of the telemetry system are twofold. The first is to provide the information for ensuring the health and welfare of the spacecraft. The second is to provide information that can be used by anyone in an educational manner. To incorporate these spacecraft into educational curricula, information about their capabilities and telemetry must be available well before launch.

The system needs to provide the public with data in a manner in which they can use it, while at the same time, provide the software development people with a means to have temporary, permanent or special data elements. This should also be done with minimum changes to software on the ground and in orbit, and minimal needs for documentation.

Microprocessor and amateur radio capabilities are continually getting better. While a fully automated telemetry capturing system might be expensive today, it will get cheaper as time goes by. Software to be developed in the next few years should look ahead to what can be expected to exist and provide for those capabilities. They may not be implemented initially in every ground station but the capability to upgrade performance and function-ality in an incremental manner should be provided.

Today's amateur telemetry decoding and display software are written as a labor of love by interested enthusiasts who want to know what's up in the satellite or as an educational exercise[11]. The easier it is for them to write good software, the more people will be attracted to the spacecraft. The development process should make the work of the ground station software developer as easy as possible.

To meet these goals, a number of requirements have been developed as discussed in the following paragraphs. The requirements have been categorized as discussed below and are summarized in Appendix 1.

Requirements for Spacecraft Builders

Spacecraft builders have to provide users with the capability to capture data automatically store the data and perform analysis on that data using computerized tools. The user community needs time to prepare for the spacecraft. People should be able to build and test ground station hardware and software a long time before the actual flight takes place. If people are "ready to go" at launch, there is a greater incentive to monitor the spacecraft and a larger population of potential sources for data during the critical early hours of the flight.

The groups that build the spacecraft are small. They should be able to use already developed and proven processes as much as possible. They should be able to offload the task of developing and providing ground station software (at least for public use).

Telemetry should be encoded to allow ground station software to use a Table Driven Approach. This is a very powerful technique which allows the users to change the coefficients of an equation, or customize displays without having to change the software itself. Typical entries in such tables are considered below.

- The channel number of the telemetry data in the frame.
- The segment identifier.
- The description of the telemetry channel that will be displayed on the screen page. (e.g. '+Z Array Grad.')
- The type of equation to use to decode the telemetry. Typical examples are shown below.

Published Format	Spacecraft	"Type"
Y = A*N^2 + B*N + C	AO-16	1
Y = B*(A+N) + C	AO-13	2
Y = B*(A-N) + C	FO-20	3
Y = B*(N+A)^2 + C	AO-13	4
Y = B*(A-N)^2 + C	AO-13	5

Where A, B and C are coefficients and may be 0. The Type values have been assigned arbitrarily. Note the Type 2 equation "B*(A+N)+C" can be expanded to "(B*A)+(B*N)+C" which, since A, B and C are constants, can also be written as "0*N^2+B*N+E" where "E = (B*N)+C", which is the same form as the Type 1 equation. Since the computer is doing the computations, people can use the format that looks simpler.

- The equation Coefficient C.
- The equation Coefficient B.
- The equation Coefficient A.
- The Units text string (e.g. '.C') in the screen display.
- The display page.
- The row in the screen the data element will be displayed.
- The column in the screen the first character of the data will be displayed.
- The characters width of the display. For example, voltage can be displayed as '1.3' or '1.28567'.
- The number of digits after the decimal point in the display.
- The kind of limit checking to perform on the telemetry channel data.
- The Low limit value (e.g. -4.00).
- The High limit value (e.g. +10.6).
- The Blanking parameter. This item indicates that computed negative results are to be displayed as a zero. It is used for example, in Solar Cell voltage computations, when negative values are produced because the equation used to convert the data is not valid at low or zero values of light.

ASCII messages should also be seen on the downlink. These messages tend to attract new people who happen to tune into the downlink, provide bulletins on schedules and other events of interest.

There needs to be some parameter in the system to enable the groundstation to detect (and correct) errors induced into the transfer frames due to the effects of noise (QRM and QRN). While noise can readily be seen in ASCII telemetry, it is difficult to detect in binary telemetry. Today's optimal approach to meeting the need to detect errors is to use packet radio AX.25 techniques as the downlink transfer protocol. Error correction schemes are not currently in use.

Requirements For Ground Stations

Ground stations should be optimized for real-time data capture. This requirement is intuitive. The better the antenna and the lower the receiver front end noise, the greater the amount of telemetry that will be captured.

Data captured should be saved for later processing and analysis in a format that can be exchanged between different computers, even if someone doesn't want to do anything with it at the time it is captured. There is no requirement for the people who analyze the data to be the same people as those who acquired it directly from the satellite.

Requirements for Ground Station Software

Users should be able to use the same computer capture, decode and display program with each satellite. The remaining requirements developed for ground station software fall into two categories, acquisition and archiving.

Software should provide real-time customizable engineering unit displays for use during acquisition sessions. It is much more interesting to see voltages, temperatures and colors change, than to look a stream of ASCII numbers or binary gibberish. The user should also be able to customize the display to either change the location of a parameter display on the screen, or set up different screens for different sets of channels. Raw telemetry display capability should also be provided to allow the user to validate the decoding configurations by performing calculations by hand to validate the displayed numbers. Audio and visual alarms should be provided for conditions when telemetry channels contain values that exceed, fall below or fall outside preset limit value(s).

Users should be able to configure the software for their particular systems. The user should be able to view the raw data to see non telemetry information such as messages.

The user should have the capability to extract telemetry data to a spreadsheet for further analysis. Each user will want to see different data displays and/or relationships. With this capability, the software developer does not have to anticipate them all.

Requirements for Tracking Capabilities

Users should be able to see when acquisition sessions can be expected. The system should configure the receiver and TNC to the spacecraft beacon frequency and modulation at AOS. The software should provide antenna control capability and compensate for Doppler frequency changes. This requirement is for automated data capture.

Telemetry Standards

Designing amateur satellites requires in-depth knowledge of several fields. In the past spacecraft developers knowledgeable in one or more areas of activity, when needing something from another area have had to develop items from first principles. When these things are viewed by professionals in those fields, the 'sloppiness' is readily apparent. **The word "amateur" is not a synonym for "sloppiness"**. Defining and using standards makes life simpler for everyone and eliminates the need to continuously reinvent the same application due to different non- compatible implementations.

Two standards are herein proposed, one for transfer frames, the second for data archive and interchange. Many of the ideas herein are adapted from the Consultative Committee for Space Data Systems (CCSDS) recommendations for standards [12,13,14,15,16].

Since future radio amateur telemetry links can be expected to use packet radio as the downlink, at least for the next few years, and binary is suitable for computer compatible data, a binary standard is recommended for the downlink telemetry transfer frames. An ASCII format is recommended for the data archive and exchange standard to allow users with minimal knowledge of data access techniques to roll their own software to get at the archived data.

Binary Format Transfer Frame Standard

The proposed standard for Radio Amateur Binary transfer frames is attached as Appendix 2. The basic assumption is that the transfer frame is encapsulated in an AX.25 packet. The AX.25 link process guarantees that delivered data will be error free. It does not guarantee delivery of every packet. The AX.25 packet contains a header which identifies the source and destination (broadcast address) of the packet. The transfer frame standard inserts a secondary header into the data area prior to the actual data itself. This secondary header provides information about the data that may be used by ground station software both during an acquisition session and for archive data processing.

The term 'octet' represents 8 bits. The rationale for the contents of the secondary header is given in the following paragraphs.

- The secondary header should contain the number of bytes ground station software should look for. **There is no need for people writing simple telemetry decoding software to have to reinvent AX.25 handlers**. The AX.25 firmware in the TNC is debugged and stable. If the first two octets in the secondary header are the number of octets in the data part of the AX.25 packet, life gets simple.
- The secondary header should contain the number of channels ground station software should decode and display. Ground station software for public use would not display the private telemetry octets, yet could archive them so they would be available for later analysis. If the ground station software only displays the public octets, and is told to ignore anything after that, most people will never see the engineering octets. Most of those who look at the raw data using a dumb program will probably never count the octets, and those who do, well, we cover that by stating that they are engineering octets, subject to unannounced changes so will not be documented.
- The secondary header should contain a packet sequence count that increments for each downlinked telemetry packet. As most ground stations miss a few packets during acquisition sessions, these octets allow data from several ground stations to be merged. For example, AMSAT-NA requested telemetry captured during the 1991 solar eclipse to be sent in for processing. A packet sequence count in the telemetry would have made merging the data produced by different ground stations a relatively simple task. This information also serves as an "up-time" indicator. Rollover of the sequence count should not be a problem due to both the low data rates and the inclusion of the sample time in the secondary header.
- The secondary header should contain the time the data sample was made. The format should be UTC time to simplify ground station software as in YYYYMMDDHHMMSS.

- The secondary header should contain one octet containing the version of the flight software in the OBC. It is updated as needed, but in any event any time a public change is made.

When data are transmitted in more than one octet, the data should be transmitted least significant octet first to comply with the rationale used in defining the Pacsat Protocols[17].

There are several options for the telemetry data formats, a straight run and a tagged run. In the Straight Run, each channel is transmitted in a predefined sequence of data (D) value octets, i.e. D1D2D3. The advantage is fewer octets, the disadvantage is the fixed format, so the OBC software revision must be updated each time a change is made, not forgetting the public announcements (ahead of time).

In the Tagged run, each channel is transmitted as two octets; channel number (C), and then data values (D), i.e. C1D1C2D2C3D3 etc. This is somewhat like the DO-17 ASCII format. The disadvantage is more octets. The big advantage is the flexible format so that the order can be changed without affecting (properly written) ground station software, updating the OBC software revision number, or needing advance public warning. This format allows the OBC to change the downlink packet contents without the world complaining.

The public telemetry data is the mixture of analog and status data, the order should be as published in the radio amateur press. When allocating status data channels, some thought should be given to facilitating manual reading of the data. Status telemetry does not need to use all 8 bits of an octet for different items. For example, if digipeater status was allocated one octet at the start of the data part of the frame with the hex code of 01H for "ON" and 20H for "OFF", anyone looking at the raw binary as displayed on a PC would see a smiling face character when the digipeater was on, and a blank space character when it was off.

The private telemetry data octets contain whatever data are needed for spacecraft command and control. If these octets use the same format as the public octets, command stations don't need special software to set up special screen displays of this 'private' data.

The Amateur Standard Formatted Data Unit (SFDU)

The purpose of the Amateur Standard Formatted Data Unit (SFDU) is to provide data in a format that can be readily exchanged between users of different types of computers. Data compression is not incorporated, because many different techniques exist for different machines and each file can be separately compressed for storage. The proposed SFDU is not the most efficient for any specific machine, but does allow for data exchange between different home computers. It also allows people with all levels of programming skills to access the data using simple text file read/write software. The proposed SFDU Standard is attached as Appendix 3.

The SFDU header should identify the Spacecraft and the Ground Station that acquired the data. It should identify the times associated with the first data frame in the unit and the last data frame in the unit to simplify automated searches of multiple SFDUs for data between specific times. Requiring this item in the header does mean that generating the SFDU may not be a one pass process.

The SFDU header should contain identification about the data format. Since the existing spacecraft use different modulation schemes and data formats. The time should also be identified as either spacecraft or ground station generated. UO-2, AO-13 and FO-20 each downlink time in the transfer frame. DO-17 downlinks time in one segment of the transfer frames. Since each ground station may not capture all transfer frames, there will be some acquired data that will consist of incomplete source frames and may not contain accurate spacecraft times.

The SFDU header should identify the number of channels of data to assist processing software to quickly scan the stored data. The SFDU header should contain information representing the OBC Software Release value for all the source data frames stored in the unit. This parameter will provide the capability to allow the spacecraft programmers to change channel assignments or inhibit channels from the downlink. For example the assignments of some of the status channels of DO-17 have been changed since its launch.

File Naming Conventions

There is not much point in having standards for the contents of files without giving some consideration to naming the files. There are two types of files to consider, the one containing the data captured during the acquisition session, and the SFDU. The standard adopted in WHATS-UP[11] for the PC is to name the capture-to-disk files with the date of capture (YYMMDD) and give them a file type corresponding to the spacecraft designator. There is little point in adding minutes and seconds, since most stations do not have clocks that are that accurate. As an example, data captured on 12 September 1991 for different spacecraft are stored in files named as shown below.

AO-10	910912.O10
UO-2	910912.U11
AO-13	910912.O13
AO-16	910912.O16
DO-17	910912.D17
WO-18	910912.W18
LO-19	910912.L19
FO-20	910912.F20
AO-21	910912.A21
SARA	910912.O23
MIR	910912.MIR

The suggestion for the SFDU naming convention is to use a name containing the spacecraft designator describing the spacecraft and the day of year of the first frame of data. The filetype should be SFD for standard format data. As an example, the name for an SFDU containing DO-17 data from January 1, 1991 would be D1791001.SFD.

Summary

This paper has covered a lot of ground in providing an introduction to telemetry. After discussing the telemetry from satellites in orbit, it discussed telemetry concepts and terminology. The telemetry process was described from a Systems point of view looking at the requirements for developers and users. Lastly the rationale for standards was covered. Draft proposed standards are appended to this paper.

In this century, building crystal sets introduced thousands of people to amateur radio and electronics even though the signals they received were not from amateur radio stations. Capturing, decoding, displaying and analyzing telemetry from space has the potential to do the same in both the last decade of this century and in the 21st Century.

Glossary

AMSAT	The Radio Amateur Satellite Corporation
AO	AMSAT-OSCAR
AOS	Acquisition of Signals
ARRL	American Radio Relay League
BBS	Bulletin Board System
CCSDS	Consultative Committee for Space Data Systems
crt	cathode ray tube
CW	continuous wave (morse code)
DO	DOVE-OSCAR
DOVE	Digital Orbiting Voice Encoder
FM	Frequency Modulation
FO	Fuji-OSCAR
FSK	Frequency Shift Keying
HTU	hundreds-tens-units
LO	LUSAT-OSCAR
LOS	Loss of Signals
OSCAR	Orbiting Satellite Carrying Amateur Radio
PC	Personal Computer
PSK	Phase Shift Keying
RTTY	Radio Teletypewriter
SFDU	Standard Formatted Data Unit
THTU	thousands-hundreds-tens-units
TLM	telemetry

TNC	Terminal Node Controller
UO	UoSAT-OSCAR
WO	WEBER-OSCAR
WOD	Whole Orbit Data
YYYYMMDDHHMMSS	Year-month-day-hour-minute-second (2 digits for each)

References

1. Richard C. Allen, W5SXD, Joe Kasser, G3ZCZ, AMSAT 8080 Standard Debug Monitor: AMS80 Version 2, Byte, Volume 2, Number 1, September 1976.
2. Joe Kasser, The AMSAT-GOLEM 80, Byte, Volume 4, Number 9, September 1979.
3. Tom Clark, W3IWI, Joe Kasser, G3ZCZ, Ariane Launch Vehicle Malfunctions Phase IIIA Spacecraft Lost, Orbit, Volume 1, Number 2, June/July 1980.
4. Karl Meinzer, DJ4ZC, Digital Communications Techniques and the AMSAT Phase III Satellites, The AMSAT Newsletter, Volume 12, Number 2, June 1979.
5. Terry Fox, WB4JFI, AX.25 Amateur Packet-Radio Link-Layer Protocol. Version 2.0 October 1984, ARRL.
6. Joe Kasser, G3ZCZ, A New Era in Amateur Radio, The AMSAT News-letter, Volume 9, Number 2, June 1977.
7. Manfred & Virginia Peschke, Report: BYTE's Audio Cassette Standards Symposium, Byte, Volume 1, Number 6, February, 1976.
8. Peter Guelzow, DB2OS, Joe Kasser, G3ZCZ, Leo Labutin, UA3CR, Radio M1/RUDAK-2, The AMSAT Journal, Volume 14, Number 2, March 1991.
9. Dennis Wingo, Ed Stluke, W4QAU, Chris Rupp, W4MIY, Amateur Satellite Communications and the SEDS AMSEP Project, Proceedings of the AMSAT Space Symposium, October 1990.
10. John M. Garvey, N6VHP, Sailing to the Moon, The AMSAT Journal, Volume 12, Number 2, August 1989.
11. Joe Kasser, G3ZCZ, WHATS-UP, Software for Amateur Radio, POB 3419, Silver Spring, Md. 20918.
12. CCSDS 100.0-G-0, Report Concerning Space Data System Standards. Telemetry: Summary of Concept and Rationale, Green Book, December I987.
13. CCSDS 700.0-G-2, Report Concerning Space Data System Standards. Advanced Orbiting Systems, Networks and Data Links: Summary of Concept Rationale and Performance, Green Book, October I989.
14. CCSDS 102.0-B-2, Recommendation for Space Data System Standards. Packet Telemetry, Blue Book, January 1987.
15. CCSDS 620.0-B-1, Recommendation for Space Data System Standards. Standard Formatted Data Units -- Structure and Construction Rules, Blue Book, February 1988.
16. CCSDS 610.0-G-5, Report Concerning Space Data System Standards. Space Data System Operations with Standard Formatted Data Units: System and Implementation Aspects, Green Book, February 1987.

17. Harold Price, NK6K, Jeff Ward, G0/K8KA, PACSAT Data Specification Standards, Proceedings of the 9th ARRL Computer Networking Conference, 1990.

Appendix 1. Requirements for Spacecraft Telemetry Systems

A.1 Requirements for Spacecraft Builders

The following requirements have been developed for spacecraft builders.

A.1.1.1 Provide computer compatible telemetry conforming to the standard for transfer frames.

A.1.1.2 Provide/publish preliminary telemetry decoding information at least 6 months before launch.

A.1.1.3 Provide tape cassettes of sample data from spacecraft ground testing or a telemetry simulator to national AMSAT organizations for distribution to potential users at least 6 months before launch.

A.1.1.4 Update the preliminary telemetry decoding information as soon as possible after launch (or even before).

A.1.1.5 Provide telemetry for use in real-time.

A.1.1.6 Formulate the telemetry to use generic decoding equations.

A.1.1.7 Provide short ASCII messages as well as computer compatible telemetry.

A.1.1.8 Provide a mechanism for allowing ground stations to detect errors in the received telemetry data induced by the radio transmission link.

A.1.2 Requirements For Ground Stations

The following requirements have been developed for spacecraft ground stations.

A.1.2.1 Optimize the receiving system for real-time data capture.

A.1.2.2 Provide capability for processing archive data.

A.1.2.3 Provide capability to read data archived by other ground stations.

A.1.3 Requirements for Ground Station Software

The requirements developed for ground station software fall into two categories, acquisition and archiving.

A.1.3.1 Provide ONE generic user friendly program for all spacecraft.

A.1.3.2 Provide the capability to decode and display the raw telemetry while it is being captured.

A.1.3.3 Perform automatic capture-to-disk of raw telemetry.

A.1.3.4 Provide link quality measurement capability.

A.1.3.5 Provide several user configurable display screens.

A.1.3.6 Provide for wild card screens (channel shows up on all screens).

A.1.3.7 Provide selectable display of Engineering units or Raw Telemetry data for each display screen.

A.1.3.8 Provide the capability to display raw (unprocessed) data.

A.1.3.9 Provide color change capabilities if a parameter value changed between successive frames.

A.1.3.10 Provide audio and visual alarms if telemetry values exceed, fall below or fall outside preset limit value(s).

A.1.3.11 Provide customizable colors, PC to TNC baud rate, data parity and stop bits.

A.1.3.12 Provide the capability to extract telemetry data to a spreadsheet for further analysis.

A.1.3.13 Provide default configuration files for different spacecraft.

A.1.4 Requirements for Tracking Capabilities

A.1.4.1 Display spacecraft orbital elements and tracking data.

A.1.4.2 Set receiver to spacecraft beacon frequency and modulation at acquisition of signal (AOS).

A.1.4.3 Set the TNC/TU to the correct modulation mode at AOS.

A.1.4.4 Provide audio warning of spacecraft AOS and loss of signal (LOS).

A.1.4.5 Provide antenna control capability.

A.1.4.6 Provide capability to compensate for Doppler frequency changes.

Appendix 2 Transfer Frame Standard

A.2 Introduction

The transfer frame standard describes the downlink standard for BINARY telemetry. The basic assumption is that the transfer frame is encapsulated in an AX.25 packet. The AX.25 packet contains a header which identifies the source and destination (broadcast address) of the packet. The transfer frame standard inserts a secondary header into the data area prior to the data itself as shown in Figure A1. This secondary header provides information about the data that may be used by ground station software both during an acquisition session and for archive data production.

Figure A1 The Transfer Frame

Secondary Header	Data

A.2.1 The Secondary Header.

The contents of the secondary header shall be as listed below in the order listed below.

Octets	Item	Octets
0,1	Total Octet Count	2
2,3	Public Octet Count	2
4,5	Packet Sequence Count	2
6-12	Time of Sample	6
13	Segment Identifier	1
14	Type of Frame	1
15	OBC Software Release	1

A.2.1.1 Total Octet Count: The secondary header shall contain two binary octets containing the total number of octets in the data section of the transfer frame (public and private).

A.2.1.2 Public Octet Count: The secondary header shall contain two binary octets containing the number of public data octets in the data section of the transfer frame.

A.2.1.3 Packet Sequence Count: The secondary header shall contain a packet sequence count in the form of a 16 bit binary number that increments monotonically for each downlinked transfer frame.

A.2.1.4 Time of Sample: The secondary header shall contain seven octets in which the OBC shall tag the data. The format is YYYYMMDDHHMMSS (UTC). The octets shall each contain two BCD digits. The first octet shall contain the YY information, the last octet the SS information.

A.2.1.5 Segment Identifier: The secondary header shall contain one binary octet containing the segment identifier. If the transfer frame contains the whole source frame, the segment identifier shall be 0.

A.2.1.6 Type of Frame: The secondary header shall contain one binary octet identifying the format of the data in the transfer frame. Two options for the telemetry data format are specified herein.

A.2.1.6.1 Straight run:- Telemetry data shall be transmitted in a predefined sequence of data (D) value octets, i.e., D1D2D3.

A.2.1.6.2 Tagged run:- Telemetry data shall be transmitted as two octets; channel number (C), and then data (D) value, i.e., C1D1C2D2C3D3 etc.

A straight run shall be identified as a type 0, a tagged run as a type 1.

A.2.1.7 OBC Software Release: The secondary header shall contain one binary octet containing the version of the flight software in the OBC. This octet shall be updated as needed, but in any event any time a change noticeable by the public is made.

A.2.2 The Data

The data section of the transfer frame shall contain the public telemetry octets followed by the private telemetry octets. Data shall be digital status bits or algebraic analog data according to the formats specified below.

Type	Description
1	One octet (8 bit value) Y = A*N^2 + B*N + C
2	One octet (8 bit value) Y = B*(A+N) + C
3	One octet (8 bit value) Y = B*(A-N) + C
4	One octet (8 bit value) Y = B*(A+N)^2 + C
5	One octet (8 bit value) Y = B*(A-N)^2 + C
11	Two octets (16 bit value) Y = A*N^2 + B*N + C
12	Two octets (16 bit value) Y = B*(A+N) + C
13	Two octets (16 bit value) Y = B*(A-N) + C
14	Two octets (16 bit value) Y = B*(A+N)^2 + C
15	Two octets (16 bit value) Y = B*(A-N)^2 + C
21	Two octets (12 bit value) Y = A*N^2 + B*N + C

22	Two octets (12 bit value) Y = B*(A+N) + C
23	Two octets (12 bit value) Y = B*(A-N) + C
24	Two octets (12 bit value) Y = B*(A+N)^2 + C
25	Two octets (12 bit value) Y = B*(A-N)^2 + C
30	Octet contains 8 bit Channel Number (used in tagged formats).
31	one octet of 8 individual status bits.
32	Octet contains two 4 bit data elements.

Where:

- A, B and C are coefficients and may be set to 0.
- Y is the arithmetic result.
- N is the decimal value of the telemetry data.

Data requiring more than one octet shall be inserted in the transfer frame least significant octet first.

Appendix 3 The Draft Standard Formatted Data Unit Standard

A.3 Introduction

The purpose of the SFDU storage unit is to facilitate the exchange of data between users of different types of computers. Data compression is not specified, because many different techniques exist for different machines and each file can be separately compressed for storage.

A.3.1 Basic SFDU Requirements

A.3.1.1 Each SFDU shall only contain data from one spacecraft.

A.3.1.2 If a spacecraft source data frame contains an OBC Software Release parameter, SFDUs shall not contain data from more than one release of the OBC Software.

A.3.1.3 The SFDU shall be an ASCII file format comprising a storage header followed by lines of data.

A.3.1.4 Each SFDU shall only contain one header. This header shall provide information about the data and the ground station that captured the data.

A.3.1.5 The SFDU header shall be a stand alone line.

A.3.2 The SFDU Header

The contents of the SFDU header shall be as listed below in the order listed below.

Octets	Item	Octets
0-4	Spacecraft ID	5
5-15	Ground Station ID	10
16-27	Start Time of Data	6
28-40	End Time of Data	6
41	Data Format ID	1
42	Time ID	1
43-45	Frame length	3
46	OBC Software Release	1

A.3.2.1 Spacecraft ID: The SFDU header shall contain five octets to identify the spacecraft in ASCII. The format shall be a two letter identifier, a hyphen and a two number identifier (AA-NN), as illustrated by the following examples UO-02, AO-13, AO-16, DO-17, and FO-20. When OSCAR 100 is launched, the hyphen shall be replaced by the digit '1'.

A.3.2.2 Ground Station ID: The SFDU header shall contain ten ASCII octets which shall contain the callsign or other identification of the ground station that captured the data. The first octet shall contain the first character of the ground station ID. Blank characters shall be used following the callsign to build the string up to 10 characters.

A.3.2.3 Start Time of Data: The SFDU header shall contain twelve octets which shall contain the ASCII string of the time associated with the first data frame in the unit. The format is UTC time as in YYMMDDHHMMSS. The first octet shall contain the higher digit of the YY information, the last octet the lower half of the SS information.

A.3.2.4 End Time of Data: The SFDU unit header shall contain twelve octets which shall contain the ASCII string of the time associated with the last data frame in the unit. The format is UTC time as in YYMMDDHHMMSS. The first octet shall contain the higher digit of the YY information, the last octet the lower half of the SS information.

A.3.2.5 Data Format ID: The SFDU header shall contain a one digit ASCII identification about the data format. Since the existing spacecraft use different modulation schemes and data formats, several storage unit data formats are specified, each optimized for specific spacecraft. Two data formats are currently specified.

A.3.2.5.1 Type "D" Storage Format :- ASCII Decimal/Octal Format. The data elements are stored as three digits (hundreds-tens-units [HTU]) in a sequential line. There are no spaces between data elements. Channel numbers are not included. If the transfer frame was not AX.25 packet, or a segment was lost, erroneous or missing, that data shall be replaced by space characters.

A.3.2.5.2 Type "H" Storage Format :- ASCII Hexadecimal Format. The data are stored in a sequential line as two hexadecimal digits. There are no spaces between data elements. Channel numbers are not incorporated. If a segment was lost, erroneous or missing, that data shall be replaced by space characters.

As a goal, if the all data elements in a frame never contain a value greater than 255, the Type H format should be used to minimize the length of the SFDU.

A.3.2.6 Time ID: The SFDU header shall contain one octet which shall identify the times as spacecraft or ground station generated. The octet shall contain a single ASCII character. An "S" shall define spacecraft time, a "G" shall define ground station time.

A.3.2.7 Frame Length: The SFDU header shall contain three octets which shall contain a three digit ASCII text string (HTU) representing the count of the number of channels of data (not the number of octets) in a single frame of data.

A.3.2.8 OBC Software Release: The SFDU header shall contain two octets which shall contain a two digit ASCII text string representing the OBC Software Release value for all the source data frames stored in the unit. This item shall only be present on the header if the spacecraft telemetry contains the parameter.

A.3.3 The Data Lines

A.3.3.1 Each frame of data shall make up one line of data in the storage unit.

A.3.3.2 Data frames shall be stored in ascending time order.

A.3.3.3 A carriage return/line feed combination shall separate the header and each of the frames.

A.3.3.4 The first 12 characters of the data frame shall be the time identification associated with the frame in YYMMDDHHMMSS format.

A.3.3.5 The next four digits shall be the packet sequence count as a hexadecimal ASCII text string (thousands-hundreds-tens-units [THTU]). If the spacecraft telemetry does not contain a packet sequence count, these four digits shall be filled by space characters.

A.3.3.6 The remainder of the frame shall be the data stored according to the specification for the type parameter.