GOES/AxSys Data Decoding

The following information is regarding the decoding of GOES transmissions from the Stevens AxSys MPU.

AxSys satellite messages are compacted messages consisting of 8 bit data characters. Because of the transmission limitations of GOES, these characters must be converted to 6 bit for transmission. Thus, the AxSys converts three 8 bit characters into four 6 bit characters for transmission. If there are not enough characters to come out even, additional, arbitrary characters are padded onto the end of the data. Each resulting 6 bit character has 32 added to it, resulting in all transmitted characters being between 32 and 96 for ASCII values. When data is received, the process must be reversed. The 32 must be subtracted from each character, and then four 6 bit characters need to be reconverted to three 8 bit characters.

When a data string is received, it first contains the GOES ID, and the string lead characters "VGTM".. The "VGTM" string is the key to where the AxSys Data is to be found. There is a single space after the "VGTM" string. What follows after that is the AXSys 6 bit encoded data. At the end of the data string are some additional characters generated by the transmitter. These are unimportant with regard to the AxSys data.

Once the AxSys data is reconverted to 8 bit characters, it can be interpreted. AxSys data is transmitted in a series of packets, linked together. Each packet represents the data from a single channel of the Axsys. The format for each channel packet begins with 2 bytes, indicating the length of the packet, a "D" character, indicating the packet contains data, and an "A", "B", "C", ... indicating the channel number (A = 1, B = 2, C = 3, and so on) The two bytes that indicate the length of the packet represent the length including the "D" character until the end of data for that packet. Immediately following that data packet will be 2 bytes indicating the length of the next channel data packet, again followed by a "D" and a letter indicating the channel number. This pattern will continue for all channels transmitted. There will be additional characters after the last channel packet, but they will not decode properly according to the method above, and can be ignored.

The two bytes indicating the channel packet length consist of the lower 6 bits of the first byte, and the lower 2 bits of the second byte, making a single 8 byte character, which will represent a length of from 14 to 255. There should be no packet smaller than 14 bytes, and none larger than 255.

The data packet decodes as follows:

Byte	Indication
1	"D", indicating a data packet
2	Letter indicating channel number (A=1, B=2, etc)
3,4,5,6	4 bytes indicating date and time (see below for breakdown) This is the date and time for the first reading transmitted, or the reference reading.

Subsequent readings are at time intervals after the first reading, as indicated by the following byte.

7 Indicates interval of recorded data, and scale factor(placement of decimal)

8,9,10 A signed 19 bit number, indicating offest for the reference reading

11,12,13 An unsigned 16 bit number, the reference reading

14 Battery voltage

The above information is considered the channel "header" information. All information that follows would relate to additional readings from that channel. Normally, additional readings are shown simply as deltas of the previous reading. This allows for deltas in a range of +/- 120 counts using a single 8 bit ASCII character. If the delta exceeds this amount, a new reference reading must be entered. This is indicated by a an ASCII character with a value of 125. Immediately after the 125 follows 3 bytes of reference reading, as previously described. then the deltas continue. If a similar delta is repeated more than 4 times (for example, a very stable water level that seldom changes, so the delta is zero for many time intervals), it can be further compacted. this is indicated by an ASCII character of value 127. Immediately following the 127 is an ASCII value indicating the number of times the delta repeats, followed by the value of the delta. Following this, normal deltas continue.

Sometimes the reading changes so much as to require not only a new reference reading, but a new offset. Under these circumstances, a whole new header would be written. This is indicated by an ASCII character value of 126. Immediately following the 126 would be a normal header, without the initial "D" character, and without he battery voltage byte. So this header would be 12 characters in length. In addition, because fo data storage issues within the AxSys, a header may be randomly generated, even though there appears no need for it.

Header Decoding Explained

byte	description	byte format	
1	Data msg code	D	
2	channel number	A thru Z	(A = chan 1, B = chan 2, etc...)
3	month/hour	xmmh hhhh	
	x:	undefined	
	mm:	2 LSB of month	
	hhhhh:	hour	
4	month/day	xmmd dddd	
	x:	undefined	
	mm	2 MSB of month	
	ddddd:	day	
5	minutes	0 - 59	
6	year	0 - 99	
7	interval & scale	xiii iiss	
	x:	undefined	

```
                iiiii:            logging interval (0: 1 sec, ... 18: 24 hrs)
                ss:               logging scale (0: 0.00, 1: 0.000, 2: 0.0)
8          high bits of data offset      bits 20 thru 14
9          middle bits of offset         bits 13 thru 7
10         lo bits of offset                   bits 6 thru 0
11   high bits of ref. reading           bits 15 thru 9
12   middle bits of ref. reading         bits 8 thru 2
13   low bits of ref reading    xooo oorr
                x:                undefined
                ooooo:            option bits -> 1: Negative offset
                rr:          bits 1 thru 0 of reference reading
14   battery voltage         convert hex value to decimal (ie, 127 is 12.7 volts)
```

Example Data
Channel 1 data, ASCII characters, and resultant data

```
 68
 65
 21
 54
 30
 0
 40
 0
 0
 0
 0
 2
 0
 126
 127
 9
 0
 -1
 0
 1
 0
 0
 0
```

```
"ID ORB 01  DATE 04/22/00  TIME 21:30:00  INTERVAL 00:15:00"
"Power: 12.6v"

00000.08   00000.08   00000.08   00000.08   00000.08   00000.08
00000.08   00000.08   00000.08   00000.08   00000.07   00000.07
00000.08   00000.08   00000.08   00000.08
```

Example Data
Channel 2 data, ASCII characters, and resultant data

68
66
21
54
30
0
40
126
60
78
52
111
6
126
-10
-20
-10
-20
-10
-20
127
4
-10
-20
0
-10
-10
0

"ID ORB 02 DATE 04/22/00 TIME 21:30:00 INTERVAL 00:15:00"
"Power: 12.6v"

00020.60 00020.50 00020.30 00020.20 00020.00 00019.90
00019.70 00019.60 00019.50 00019.40 00019.30 00019.10
00019.10 00019.00 00018.90 00018.90

AxSys/GOES Operation

Introduction

The Stevens AxSys MPU, when enabled for GOES transmission, will continue to function as it normally does, with the following exception:

On a predefined transmit interval and timing, the AxSys will be awakened by the transmitter to queue a message for transmission. This message will consist of a single, compacted packet of data for all enabled channels, containing the recorded data within the AxSys since the last transmission. Transmitted data will retain full data scaling as is present in the channel setup in the AxSys without any need for special configuration by the user. The only exception is that channel IDs are not transmitted, only reference to channel numbers (ie, channel IDs would be GOES 1, GOES 2, GOES 3, etc.). Once a data packet is received by the user and unpacked, the data will have the same format as might be obtained from the AxSys using a Data Card, or using a PC at the site and performing a "Data Dump".

Data packets are sent in a similar format to that used by the AxSys for ORBCOMM satellite transmissions. This is a proprietary 8-bit data compaction scheme. However, GOES transmissions do not support 8-bit data, nor do they support non-printable characters. Therefore, the GOES data packet is uu- encoded (3 8-bit characters are converted to 4 6-bit characters, and 32 is added to the number; thus all transmitted characters are in the ASCII range of 32 to 96). Received data must be re-converted to 8- bit data, and then unpacked using Stevens STDV, AXREAD, or the users own software routine. Also, the Stevens utility program GX.EXE will unpack AxSys/GOES messages in a batch process together with STDV.

Specific channels can be enabled or disabled for transmission in the data packet. GOES timing is currently programmed directly in the Transmitter. Future capabilities should allow for this to be accomplished through the AxSys keypad and display.

Using GOES requires authorization and time/channel assignments from the United States Government. Typical transmission windows are 1 minute every 4 hours. Transmissions are only one way, from the remote site to the satellite. If a user has their own receiving station, they can receive data directly from the satellite. If not, data must be obtained by a dial-up or Internet connection to the US Government facility at Wallops Island, Maryland.

1.0 GOES Operational Overview. Because of the critical timing needed for GOES Satellite operation, the transmission timing is all accomplished by the GOES transmitter module. It has its own clock, and is programmed through the RS-232 port for the following parameters:

Time
Date
First transmission time
Transmission time interval
Identification

The setting of these parameters is described in section 2.0.

One minute prior to the transmission time interval, the transmitter wakes up from low power standby, and in turn wakes up the attached AxSys MPU. The AxSys then transfers a data packet to the transmitter for transmission. There is a one minute time window for this to occur. At the end of that time window, the transmitter takes the received data packet and sends it off over the appropriate GOES data channel. The entire system then returns to the low power mode until the next time interval.

Data is sent from all channels of the attached AxSys MPU which have a "+" character as the last character in their Channel ID. This allows the user to select which channels are appropriate for transmission over the GOES network. To enable GOES transmission, the user selects "GOES" for the AUTO-TELEMETRY setting under the System Menu, and also set the BAUD to 8-N-1.

Data is sent in one compacted data packet, which contains data from all channels which are enabled for transmission. Data sent includes all recorded readings since the previous transmission. While the format is compacted, it is easily unpacked into standard AxSys data formats once received by the receiving station. Information on unpacking the data packets is contained in Section 1.1.

1.1 Data Packet Format. Data transmitted over the GOES system must be in printable ASCII characters. Since the AxSys compacts data into an eight bit format, an encoding scheme must be used. This is achieved by converting 3 eight bit words into 4 six bit words, and adding 32 to each character. This results in all characters transmitted having an ASCII value between 32 and 96 (a six bit character will give an ASCII value between 0 and 64), which are all printable characters. Thus, when the data is received, it must first be converted back to the appropriate eight bit format. Each data packet begins with the lead characters "VGTM" followed by a space.

The next step is to separate the AxSys Channels. Each sub-packet of channel data begins with the letter "D", followed by a channel designator. "A" is channel 1, "B" is channel 2, "C" is channel 3, and so on. The character directly in front of each "D" character is a single character whose ASCII value determines the length of the channel sub-packet. For example, a "x", which is an ASCII value 120, would indicate that the channel sub-packet, including the "D" start character, is 120 characters long. This alerts the user as to where to expect the start of another channel sub-packet, if there is one.

Once subpackets are separated, data can be read using the Stevens Utility STDV, or the Windows version AXREAD. Alternatively, GOES packets can be processed automatically using the DOS based program GX.EXE. This results in a data packet designated GOESDATA.TXT. The format for invoking GX.EXE is:

C:\>**GX xxxxxx.yyy**

Where xxxxxx.yyy is the file designation for the raw GOES data packet.

2.0 GOES Transmitter Setup. Various setup parameters are available through the serial port connection to the GOES transmitter. These can be accessed with a computer running a standard terminal program such as WINDOWS TERMINAL.

Connect to the serial port using the following setup parameters:

Baud rate	9600
Data bits	8
Stop bits	1
Parity	None

To access the menu, first press the ESCAPE key to wake up the transmitter, then press "M" to display the main menu. The Transmitter main menu is shown below:

TRANSMITTER Version G2.01-E Main Menu

T	Time
D	Date
L	Location Name
S	Sensor
N	Self-Timed GOES
R	Random GOES
Gn	Group (1-4) Setup
E	Retrieve Data
·	Ops Tests
·	PCMCIA
·	Activate

Selection:

Each function is executed by pressing the proper letter and then pressing <ENTER>. The only parameters used by the AxSys GOES System are:

Time
Date
Self-timed GOES

TIME SELECTION

Current time: 09:48:12
Enter new time: HH:MM:SS <ESC> to exit

The user may enter a new time in the 24 hour format shown above. Seconds are optional. The time is written into the real time clock of the TRANSMITTER when the user presses <ENTER>. At any point the user may press the <ESC> and abandon the new time entry; the old time is retained.

DATE SELECTION

Current Date: 11-10-95
Enter new date: MM-DD-YY <ESC> to exit

The user may enter one or two digit numbers for the day, month and year, with each field separated by a hyphen(-) or a slash (/), followed by <ENTER>. At any point hitting the <ESC> key will abandon the new date entry and the old date will be retained.

SELF-TIMED GOES SELECTION

From the main menu, select IN' <ENTER> to get to the self-timed GOES setup menu. The following menu will appear:

Self-Timed GOES

A	Transmissions:	ON
I	ST ID:	010lF3DC
C	ST Channel:	151
H	First Tx:	01:25:15
T	Tx Interval:	0240 min
P	Preamble:	SHORT
F	Format	ST GOES

<ESC> Return to previous menu

Selection:

Enable/Disable Self-Timed Transmissions (A)

Enables or disables self-timed GOES transmissions. Press "A" <ENTER> to toggle the setting. If transmissions are disabled, the TRANSMITTER will function either in a random only mode or as a data logger.

ID (I)

This selection prompts the user for the 31-bit BCH-encoded platform ID assigned by the administrator to the DCP. The ID string must be entered as an even 8-digit hex number.

Channel (C)

This selection prompts the user for the GOES transmit channel to be used by the TRANSMITTER.

First Transmit Time (H)

The user must specify the time of the first GOES transmission of the 24hour day. This time is usually between midnight and 04:00 AM. All subsequent transmissions will occur at the intervals specified by the Transmit Interval selection. It is good practice to include a "guard" time in the seconds field of this time (5 to 15 seconds). This prevents your DCP from drifting out of the proper time slot; it also prevents your transmissions from being interfered with by another DCP with a drifting clock.

Transmit Interval (T)

Specifies the interval in minutes between self-timed GOES transmissions. Valid entries can be between 3 and 1440 minutes, although intervals are usually 180 or 240 minutes.

Preamble Type (P)

Both the SHORT (1 second) and the LONG (7.5 seconds) preamble types may be selected. Random transmissions automatically revert to the SHORT preamble. If any channel greater than 200 (international channels) is selected then the LONG preamble is sent regardless of the setting of this parameter. Older demodulators require the LONG preamble; the SHORT preamble can almost always be used. Press "P" <ENTER> to toggle the setting of this parameter.

Format Transmission (F)

This option causes the TRANSMITTER to format a self-timed GOES transmission and to print it to the serial port. This is useful to verify that the transmission format is correct and to help program the decoding software. The format is exactly as is transmitted except the battery voltage character (the last character in the transmission) is always defaulted to a 'B'.

MaxBytes

This is the only setup parameter required for using the TRANSMITTER GTM function. This selection allows the user to specify the maximum number of bytes that the TRANSMITTER will accept for transmission. Setting this number to the proper value insures that the TRANSMITTER will not transmit outside its assigned time window in the event of a malfunctioning data logger. For example, if the data logger was supposed to transfer 500 bytes of data to the TRANSMITTER, but instead transferred 1500 bytes, it is likely that the transmission would take too long and overrun into the next time window. Setting the maximum number of bytes to 500 or 550 would prevent this error. Each byte takes 80mS to transmit over the satellite system.

The transmission time is determined by the number of data bytes and the preamble type. A rough estimate of the transmission time can be calculated as follows:

Tx Time = Preamble + (number of bytes x 80mS)

Examples:

Long Format, Max bytes set to 500;
Tx Time = 7.5 seconds + (500 x .08) = 47.5 seconds.

Short Format, Max bytes set to 600;
Tx Time = 0.75 seconds + (600 x .08) = 48.75 seconds.

Long Format, Max bytes set to 1000;
Tx Time = 7.5 seconds + (1000 x .08) = 87.5 seconds,
too long for most time slot assignments.

The TRANSMITTER will calculate the exact time value for you. See the last part of the Serial Port section.

Max Bytes is a maximum value. The TRANSMITTER only transmits the number of data bytes transferred from the data logger., If the Max Bytes is set to
500, and only 100 bytes are transferred from the data logger, then the
TRANSMITTER transmits only the 100 data bytes. The TRANSMITTER can buffer up to 2000 bytes.

Activatel (A)

This selection is the final entry to be made. This sets the transmitter "Active" and ready to begin transmissions per the system program.

DCP-TRANSMITTER Data Transfer

This section defines the serial interface between the TRANSMITTER GTM and the external data logger.

Electrical Interface:	RS-232
Baud Rate:	1200, 2400, 4800, 9600
Data:	8 bits/char, No parity, 1 stop bit

Message Format

Data Header -- The TRANSMITTER inserts the following data header at the beginning of every transmission: < Space >VGTM< Space >-

No Data -- The TRANSMITTER sends a prompt to the data logger requesting data. If no data is received within 50 seconds, the TRANSMITTER assumes that the data logger is not functioning and transmits a "No Data" message.

Data Characters -- All characters transmitted over the GOES (or other satellite) system must be valid ASCII characters. The data transferred from the data logger to the TRANSMITTER must be 8-bit characters between 0 and 127 decimal. The most significant bit of every byte will be zero. The TRANSMITTER will set the most significant bit to the proper parity for transmission.

Forbidden Data Characters -- The following ASCII characters are not allowed to appear in the transmitted data: SOH, STX, ETX, ENQ, ACK, DLE, EOT, NAK, SYN, ETB, CAN, GS, and RS. If any of these characters appear in the received data, they will be changed to the ASCII "/" character (2F Hex).

Special Characters -- Three ASCII characters are used for signaling from the data logger to the TRANSMITTER. The ASCII EOT character is sent from the data logger to the TRANSMITTER to mark the end of the data field. When the TRANSMITTER detects the EOT, it responds by sending an "OK" response to the data logger. The ASCII CAN and DLE characters are used in error checking (see below).

Error Checking and Correction -- The TRANSMITTER echoes each character back to the data logger as it is received. If the data logger receives an echo byte that is not correct, or does not receive the echo byte within a reasonable time, the data logger sends the ASCII DLE character to the TRANSMITTER. The TRANSMITTER echoes the DLE character, and moves backs one byte in its receive buffer. The DLE is used, in effect, to erase a one byte error. The DLE erase procedure may be used any number of times within a data transfer session.

If for some reason, the data logger is unsure that the TRANSMITTER has received and buffered the data correctly, the data logger may send the ASCII CAN ("cancel") character to the TRANSMITTER. When the TRANSMITTER detects the CAN character, it abandons all data received so far and restarts the procedure with the "send data" prompt.

3.0 AxSys Settings. To set the AxSys to work with the GOES transmitter, make the following setup changes:

1. For each channel that is to be transmitted, set the last character of the Channel ID to "+". This will enable the channel for transmission.

2. Under System Setup, set the Baud Rate to 8-N-1.

3. Under System Setup, set Auto-Telemetry to "GOES" or "GOEX". GOES is the standard mode and will transmit only new data each interval. GOEX is a special mode which will transmit new data as well as retransmit data from the previous interval.

This completes the setup for the AxSys to use the GOES Transmitter.

Troubleshooting

Problem	Possible Cause
No Display	Poor power connection Fuse is blown Battery is low
No "GOES" transmission	Incorrect time Incorrect time of first transmission "GOES" enable not set Antenna connection bad Antenna not properly aligned Wrong "GOES" Channel Wrong "GOES" ID Fail-safe condition
No Transmitter Power Under Ops Test	Antenna connection bad Fail-safe condition
"GOES" transmission but no data	No power to AxSys Fuse is blown Channel ID does not end in "+" "GOES" or "GOEX" not set in Auto-Telemetry "MAXBYTES" not set to 500 on transmitter "BAUD RATE" not set to 9600-8-N-1 on AxSys