Appendix B

PSM-4900/4900H/4900L Remote Control Command Protocol

This Appendix applies to Datum Systems' PSM-4900, M5 Class Modem series including the PSM-4900 (70 MHz IF), the PSM4900L (L-Band IF) and the PSM-4900H (Hybrid 70 Mhz/L-Band). The M5 Class Modem is distinguished from M4 Class Modems which include the PSM-512, PSM-2100 and PSM-2100L. The remainder of this document refers only to these modems as the M5, PSM-4900 or the modem.

Revision History

Rev. 1.0, July 5, 2000 – Initial Release.
Rev. 1.1, July 28, 2000 –Corrections.
Rev. 1.2, December 10, 2000 – Additions for Reed-Solomon and IBS Multiplexer.
Rev. 1.3, December 20, 2000 – Additions for SDMS (Ethernet) optional Interface.
Rev. 1.4b, November 5, 2001 – Additions for Hybrid unit and TPC and corrections.
Rev. 1.5, November 20, 2001 – Rewrite with added information.
Rev. 1.6, August 20, 2002 – Additions for L-Band (PSM-4900L) unit.
Rev. 1.7, December 12, 2002 – Additions for new TPC and TCT capabilities.
Rev. 1.8, January 12,2003 – Additions for new Mod Clock Auto Mode

Differences From Previous Datum Systems' Command Protocols

The command protocol described here and used in the M5 Class Modems uses the same basic packet structure as previous versions, but the contents, methods and scope of commands differ significantly. In short the M4 Class Modems used a single packet for each possible function command, while this new M5 structure uses a single packet for to control multiple related functions. This was done for two reasons; first, the M5 contains significantly more programmability and second, this new structure improves efficiency in typical applications.
Some other differences between the M4 and M5 protocols are.

- There is no ASCII Packet Protocol in the M5 Modem.
- The M5 Modems are capable of accepting commands at the near end for monitor/control of the far end modem. This feature requires that the modem be equipped with the "Multiplexer" option card.
- There is no read, write and write to EEPROM function in the M5 Modem. This is because the non-volatile memory is a different type and is always written to when power is removed from the modem. This results in the "Mode" Byte having a different function. It is now used to determine if a command is for the near end or far end modem. All "Read" commands are designated by setting the data byte count to zero.

M5 Modem Control Overview

The PSM-4900 modem can be controlled by the front panel or from an external device. External devices may be a "dumb terminal" or terminal emulation program, a specialized controller connected via an IrDA optical connection or via the rear "Command" Port. The use of an external "terminal" for control of the PSM-4900 is covered in the main manual. This Appendix describes the protocol for control of the modem by an external computer or controller connecting to either the rear panel Command Port or the front panel IrDA port.

The computer may take many forms ranging from an IrDA capable device such as a PDA or dedicated stand-alone processor to a personal computer or a larger mini or main-frame computer, and is referred to here simply as a controller. The PSM-4900 contains full software allowing it to be externally controlled. Note however that no software is provided for the external controller, which is the responsibility of the user.

The protocol described here contains "place holders" within the data structure in two types, "Reserved" and "Future Expansion". The reserved data structure elements are used by the factory for specialized testing

and calibration. The future expansion elements are spares and data needed for yet to be implemented features. The expanded frequency control needed in L-Band IF type modems were “future” items.

Setup For Remote Control

Before the PSM-4900 can be externally controlled it must be set to operate in the proper remote control mode via the front panel. Several parameters must be set as dictated by the control system to be used:

1. Modem Address (all control methods)
2. Control Interface as either RS-232 or RS-485 (Control Port, J6)
3. IrDA Enable (IrDA port)
4. Bit Rate, Format (all control methods, separate entries for IrDA and Control Port)
5. Remote Protocol (Control Port)

Front Panel setting of these parameters is available in the <Unit: IrDA> and <Unit: Remote> columns. These parameters may also be set via the remote control port itself, but this is dangerous, as it will probably result in loss of communications.

The RS-232 interface is only useful in a point to point control with one controller and one modem because of the nature of RS-232. The 485 type interface allows multiple modems and controllers to be tied to the same serial bus. The modem address insures that the modem only responds to messages intended for it.

The RS-485 interface on the modem is configured as a “4 wire” interface. That means that the transmit and receive wire pairs are separate. This allows a controller to both talk and listen at the same time. If a “2 wire” configuration is desired, the transmit and receive pairs may be simply tied together external to the modem. Care should be taken here to insure that the “A” or “-“ side of the transmit is tied to the the “A” or “-“ of the receive, and the same for the “B” or “+” side.

Packet Protocol Basics

All remote control communications are formatted as “packets” of information including addresses, commands, responses and data. The modem never initiates transmission of a packet on its own, it only responds to a request or command packet from the controller. The sequence of events in this protocol is for the controller to send a command packet to a particular addressed modem. The addressed modem reads the command packet and if valid executes the command and sends back a response packet. A response is always returned unless:

a) The unit is improperly addressed, which causes the modem to never see the message packet, or
b) The message is globally addressed to all modems, or
c) The message flags or checksum are incorrect causing the modem to reject the message.

If the message packet address is accepted by the modem but the packet format is incorrect then an invalid message response is returned. The response may take one of several formats depending on the command type, but the response format for any particular command is fixed.

Example of Binary Packet Control System

An example Binary control system might consist of a single PC type computer communicating with one to 10 or more modems using an RS-485 interface card installed in the PC as one of the “Com” channels. This setup might be used to monitor and control a small station. The PC could in turn be communicating with a central computer system via a telephone line and modem. A program written in “C” or “BASIC” could periodically request status of each modem to insure that nothing has changed, and upon command from the central computer would change the parameters of any individual modem. For an example of the message format similar to this, see the Binary Packet Command and Response Message sections below.

Binary Packet Command Message Format

The remainder of the appendix describes the Binary packet formats.

Each control entity has its own set of change flags to determine if the changed data has been read or not. i.e. the IrDA, front panel and remote control.

The Binary Packet from the controller to the PSM-4900 Modem adheres to the following message format.

Byte 1 Pad Byte FF hex	Byte 2 Opening Flag A5 hex	Byte 3 Destination Address 8 bits	Byte 4 Source Address 8 bits	Byte 5 Binary Command 8 bits	Byte 6 Mode Byte 8 bits
Byte 7 Data Byte Count	Byte 8 - (n-3) Data Bytes 128 maximum	Byte n-2 Closing Flag 96 hex	Byte n -1 Checksum	Byte n Pad Byte FF hex	

Address Field

The modem is assigned an address via the front panel control or via the remote control line itself. Modems are normally shipped with the address preset to "1". When multiple modems are connected to the same RS-485 control line each must have an unique address to avoid conflicts. A modem may have the same address as any controller device on a shared bus (not recommended), but no two controller devices may have the same address. Modems respond only to incoming messages containing their unique address in the destination address position of the control message. A destination address of 255 (0xFF) is a global address received by all modems.
The Source address may be any value from 0 to 254 that is not assigned to a modem and becomes the destination address of the response message. This allows for multiple controllers in one system. The convention of using 255 as the global address is assumed here also for controllers. The modem makes no use of the source address other than to place it in the response packet directing the response to the originating command source.

Mode Byte Field

The Mode Byte is "01" for a local (near end) command, "02" hex for commands intended for the far end (remote) modem. Access to remote modem requires that each modem be equipped with the Datum Multiplexer option and that it be enabled. Note that the mode byte is not returned in the response packet.

Command Byte Field

The Binary Command Byte is taken from the Command Tables below. Note that there may be multiple command byte tables depending on the modem software version number. The software revision is read from the front panel LCD display.

Data Byte Count Field

The Data Byte Count field includes the total number of Data Bytes only, and should be zero (00) for read mode. Note that this determines if the command is a read (request for information) or write (command to change parameters).

Data Byte Field

Only two general data formats are used: 1 byte entries are a single character or unsigned byte type or containers for bit flags; and 4 byte numbers are in signed format (commonly called "long integer"). An extended 6 byte long integer is used for L-Band and RF Frequencies. No floating point numbers are used, although the incremental value of an entry may allow a decimal point value. For instance the transmit power level is entered as an integer in increments of 0.1 dB, so an entry of -176 represents -17.6 dB. The incremental value (represented by 1 least significant bit change) is determined from the Write Bytes section of the Command Tables by ignoring any decimal point and using the number of displayed digits. Thus frequencies are entered in 1 Hz increments, data rates in 1 bps increments, and times in increments as shown in the tables. No offsets are used in any of the number entries.

Checksum Field

The checksum is computed as 256 minus the sum of all bytes excluding opening and closing pad bytes, and the checksum itself. The checksum is modulo 256, that is the checksum never exceeds 255 in value but rolls over at 256. The sum of all bytes (modulo 256) including the checksum itself is always zero.

Pad Bytes

The Pad Bytes are not checked by the processor, and multiple pad bytes may be used. Pad bytes are all 1's or "FF" hex. The pad bytes serve several functions in an RS-485 configured system, indicating clean transitions from idle to active states. The modem only sends a starting pad byte in RS-485 responses.

Binary Packet Response Message Format

The Binary Response Packet from the PSM-4900 Modem to the controller adheres to the following message format. The response from a modem will occur within approximately ½ second. Note that a modem set to 485 control port mode mutes its receive while sending the response message, so if the 485 bus is configured as 4 wire or 2 wire a modem will not receive a message while responding to a previous message.

Byte 1 Pad Byte FF hex	Byte 2 Opening Flag 5A hex	Byte 3 Destination Address 8 bits	Byte 4 Source Address 8 bits	Byte 5 Binary Command 8 bits	Byte 6 Status Byte 8 bits
Byte 7 Error Byte	Byte 8 Data Byte Count	Byte 9 - (n-3) Data Bytes 128 maximum	Byte n-2 Closing Flag 96 hex	Byte n -1 Checksum	Byte n Pad Byte Not Sent

The Destination address is taken from the Source address of the incoming packet to which this is a response. The Status and Error Bytes are defined later in this Appendix. The Data Byte Count field includes the total number of Data Bytes only, and should be zero for read mode. The Pad Bytes are not checked by the processor, and multiple pad bytes may be used. The checksum is 256 minus the sum of all bytes excluding opening and closing pad bytes, and the checksum itself. The sum of all bytes including the checksum itself is always zero.

Example Binary Packet Command and Response Messages

As an example of using the remote control assume that the controller should check the transmit frequency, and if not set correctly change it to 74.652 MHz. The modem address is set to 12 decimal and the controller is address 200 decimal. Setting the transmit frequency in this example will consist of three operation; 1) Checking the current frequency, 2) Setting the transmit frequency and 3) Checking the response to insure that the setting was correctly accomplished. What is not shown in any of these steps are the necessary background work such as building and receiving packets, calculating checksums and testing for errors.

1. **Check the transmit frequency:**

To read the transmit IF frequency we send a command request to the modem with the Mod IF command number (41 hex) and set the data byte count to zero.

Command packet from controller to read the transmit frequency

Byte 1 Pad Byte "FF hex"	Byte 2 Opening Flag "A5 hex"	Byte 3 Destination Address- 8 bits "0C hex"	Byte 4 Source Address- 8 bits "C8 hex"	Byte 5 Binary Command 8 bits "41 hex"	Byte 6 Mode Byte 8 bits "01"
Byte 7 Data Byte Count "00 hex"	Byte 8 Closing Flag "96 hex"	Byte 9 Checksum "AF hex"	Byte 10 Pad Byte "FF hex"		

Response packet from modem

Byte 1 Pad Byte "FF hex"	Byte 2 Opening Flag "5A hex"	Byte 3 Destination Address- 8 bits "C8 hex"	Byte 4 Source Address- 8 bits "0C hex"	Byte 5 Binary Command 8 bits "41 hex"	Byte 6 Status Byte 8 bits "18"
Byte 7 Error Byte "00 hex	Byte 8 Data Byte Count "24 hex"	Byte 9 - (n-3) Data Bytes 128 maximum See Below	Byte n-2 Closing Flag "96 hex"	Byte n -1 Checksum "52 hex"	Byte n Pad Byte Not Sent

The full data response is:
Byte 0 – 00 hex – No Changes since last read
Byte 1 – 00 hex – "
Byte 2 – 00 hex – '
Byte 3 – 00 hex – '
Byte 4 – 05 hex – Carrier enabled, QPSK mode
Byte 5 – 80 hex – No Preamble, No AUPC, Mute Automatic, 75 Ohm
Byte 6 – 00 hex - Spares
Byte 7 – 00 hex - Spares
Bytes 8-11 – 80 1D 2C 04 hex – Frequency = 70.000000 MHz (Hex = 04 2C 1D 80).
Bytes 12-13 – 00 00 hex – Used for L-Band
Bytes 14-17 – 00 00 00 00 hex – No Carrier Offset
Bytes 18-19 – 9C FF hex – Minus 10.0 dBm output level (FF 9C Hex = -100 in 0.1dB steps)
Bytes 20-21 – 00 00 hex – AUPC
Bytes 22-23 – 00 00 hex – AUPC
Bytes 24-25 – 00 00 hex – AUPC
Bytes 26-31 – 00 00 00 00 00 00 hex – Used for L-Band
Bytes 32-35 – 00 00 00 00 hex – Spare

This response said that the modem was set to 70.000000 MHz, so the IF frequency parameter must be changed. Note that only the packet byte sequence is given below.

2. **Program the modem transmit frequency.**
Command packet from controller to set the transmit frequency to 74.652 MHz:

Packet Start - "FF A5 0C C8 41 01 24"
Data Bytes:
Byte 0 – 01 hex – Change the Transmit Frequency Only
Byte 1 – 00 hex – "
Byte 2 – 00 hex – '
Byte 3 – 00 hex – '
Byte 4 – 05 hex – Carrier enabled, QPSK mode
Byte 5 – 80 hex – No Preamble, No AUPC, Mute Automatic, 75 Ohm (Shown but ignored)
Byte 6 – 00 hex - Spares
Byte 7 – 00 hex - Spares
Bytes 8-11 – 60 19 73 04 hex – Frequency = 74.652000 MHz
Bytes 12-13 – 00 00 hex – Used for L-Band
Bytes 14-17 – 00 00 00 00 hex – No Carrier Offset
Bytes 18-19 – 9C FF hex – Minus 10 dBm output level (Shown but ignored)
Bytes 20-21 – 00 00 hex – AUPC
Bytes 22-23 – 00 00 hex – AUPC
Bytes 24-25 – 00 00 hex – AUPC
Bytes 26-31 – 00 00 00 00 00 00 hex – Used for L-Band
Bytes 32-35 – 00 00 00 00 hex – Spare

Packet Ending – "96 7A FF"

Several things should be noticed here. First, the Write Enable flags do not always cover the enable change for all of the Write bit flags themselves, therefore the Carrier Enable must be repeated from the previous response or set for this specific case. Second, a value for the transmit output level was used although it was not necessary. This could have been all zeros since the enable flag did not enable setting this parameter.

3. **Read the response from the modem showing the correctly set transmit frequency.**

The response will look the same as the original request command except that the frequency change flag should be set and the data bytes for the frequency should now read “60 19 73 04” hex – meaning the Frequency = 74.652000 MHz

The response status byte is shown as “1A” hex, and the error byte was taken as “00”. The “1A” status byte means “Alarm B” is active, and the Demod is in alarm. These status bits may change depending on other factors in the modem.

Following is an abbreviated list of currently available commands and the parameters monitored and/or controlled by that command. Click on the underlined “Command Byte” to jump to that command table.

Command Number Index		
Command Byte (hex)	**Name**	**Description**
00	Unit Status	Current Unit Status
01	Unit Config	Unit Configuration
02	Unit Keybrd	Unit Keyboard Setup – Click, Backlight
03	Unit IrDA	Unit Infrared Control Setup
04	Unit Remote	Unit Remote Control Port Setup – Address, rate, format
05	Unit Ref	Unit Reference Oscillator Setup – Source and Calibration
06	Unit Redundancy	Unit Redundancy Setup – Enable and monitored alarms
07	Unit Monitor	Unit auxiliary analog voltage monitor output for AGC or Eb/No or Mod CXR Level
08	Unit Alarm	Unit Alarm and Relay Setup
09	Unit Test	Unit Tests – Self Test
40	Mod Status	Current Modulator Status
41	Mod IF	Transmit IF Control – Frequency, Offset, Level, AUPC, Carrier Enable
42	Mod Data	Modulator Data Setup - Bit rate, FEC
43	Mod Alarm	Modulator Alarm Setup
44	Mod Test	Modulator Test Modes -
45	Mod RS Fec	Reed-Solomon Option Control
46	Mod Mux	IBS Multiplexer Control
47	Mod BUC	Control of Transmit Block Up Converter for PSM-4900L
80	Demod Status	Current Demodulator Status
81	Demod IF	Receive IF Control – Frequency, Offset, Acquisition
82	Demod Data	Demodulator Data Setup – Bit Rate, FEC
83	Demod Alarm	Demodulator Alarm Setup
84	Demod Test	Demodulator Test Modes
85	Demod RS Fec	Reed-Solomon Option Control
86	Demod Mux	IBS Multiplexer Control
87	Demod LNB	Control of Receive LNB Down Converter for PSM-4900H&L
C0	Intf Status	Interface Status
C1	Intf I/O	Interface Type and Control Line use.
C2	Intf Alarm	Interface Alarm Status
C3	Intf Test	Interface Test Setup – Data Loop-backs, BER Control
C4	Intf SDMS	SDMS Option (Ethernet) Setup/Control – IP Address & Mask

About the Command Tables:

Each command number begins on a new page. The command number is shown on the first line. There are two main groups of bit/byte sequences included in each command; Read (from Modem) and Write (to Modem).

The “Read” sequence represents the response from a request using that command number and includes “Change” flags, “Read” [Data] flags and “Read” [Data] bytes. All shown information is returned. The “Change” flags show what parameter has changed since the last read, and is reset by the read action. Each control method contains its own set of change flags so that a read from the IrDA port does not reset the change flags from the serial control port.

The “Write” sequence represents the Command request to set a new parameter(s) using that command number and includes “Enable” flags, “Write” [Data] flags and “Write” [Data] bytes. Only the item(s) with enable flags set are changed. The other values must be present in the packet, but may be “dummy” values since they are not used.

The command or Write sequence to request information only is the command number with the packet data byte count set to zero.

Note: The “Common Notes”, Error and Warning status information are at the end of the tables.

Unit Status, Command [00h],
Unit Status, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Modem	Ref	Redun	Name	Model	Serial	Ver	Test
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Status, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	MCxrEn	DemLck	ModAlm	DemAlm	RefAlm	ClkAlm	UnitTst	ModTst
Byte 5	DemTst	IntfTst	RefSrc	RefBad	OvrTAlm	DHrdFail	OnLine	RdnAlm
Byte 6	NoBck Alm	BckUp Alm	TpcFec Opt	SeqFec Opt	RsFec Opt	IbsMux Opt	SDMS-L Opt	MBurst Opt
Byte 7	FecFail	OptFail	IntfFail	OcxoAlm	TctFec Opt	Spare	Spare	Spare

Unit Status, Read Bytes

Bytes 8-24	Unit Name String Terminated with a 00h
Bytes 25-41	Unit Model Number String Terminated with a 00h
Bytes 42-45	Serial Number, 32b
Bytes 46-62	Unit Software Version String Terminated with a 00h
Byte 63	Undefined
Bytes 64-67	Spare

Unit Status, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	0	0	Redun	Name	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Status, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	x	x	x	x	x	x	x
Byte 5	x	x	x	x	x	x	XferRqst	x
Byte 6	x	x	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Status Write Bytes

Bytes 8-24	Unit Name String Terminated with a 00h
Bytes 25-41	X
Bytes 42-45	X
Bytes 46-62	X
Byte 63	X
Bytes 64-67	Spares

OcxoAlm = OCXO Cold Alarm on L-Band Only.

Unit Config, Command [01h],

Unit Config, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Store	Recall	Retore1	Restore2	Restore3	Restore4	Restore5	Restore6
Byte 1	Restore7	Restore8	Modem	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Config, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	0	0	0	0	0	0	0	0
Byte 5	NoFactry	DemEn	ModEn	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Config Read Bytes

Bytes 8-9	Restore 1 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 10-11	Restore 2 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 12-13	Restore 3 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 14-15	Restore 4 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 16-17	Restore 5 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 18-19	Restore 6 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 20-21	Restore 7 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 22-23	Restore 8 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 24-27	Spare

Unit Config, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Store	Recall	Restore1	Restore2	Restore3	Restore4	Restore5	Restore6
Byte 1	Restore7	Restore8	Modem	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Config, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Store0	Store1	Store2	Store3	Recall0	Recall1	Recall2	Recall3
Byte 5	x	DemEn	ModEn	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Config Write Bytes

Bytes 8-9	Restore 1 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 10-11	Restore 2 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 12-13	Restore 3 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 14-15	Restore 4 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 16-17	Restore 5 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 18-19	Restore 6 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 20-21	Restore 7 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 22-23	Restore 8 Delay, 16b, (0 to 14,400), 0=Disable Restore, 1 Second Increments
Bytes 24-27	Spare

[Store3-Store0] = Configuration Store, 4b, 0=Factory (Once Only), (1 to 8)=User
[Recall3-Recall0] = Configuration Recall, 4b, 0=Factory, (1 to 8)=User
[DemEn] = Demodulator Mode, 1b, 0=Disabled, 1=Enabled
[ModEn] = Modulator Mode, 1b, 0=Disabled, 1=Enabled

Unit Keybrd, Command [02h],
Unit Keybrd, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Entry	LcdActve	LcdIdle	LcdCntst	Activity	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Keybrd, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	FpEn	Mode0	Mode1	Entry0	Entry1	Actve0	Actve1	Idle0
Byte 5	Idle1	Actvty0	Actvty1	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Keybrd, Read Bytes

Byte 8	Lcd Contrast, 8b, (0 to 20)
Byte 9	00h
Bytes 10-11	Spare

Unit Keybrd, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Entry	LcdActve	LcdIdle	LcdCntst	Activity	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Keybrd, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	Mode0	Mode1	Entry0	Entry1	Actve0	Actve1	Idle0
Byte 5	Idle1	Actvty0	Actvty1	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Keybrd, Write Bytes

Byte 8	Lcd Contrast, 8b, (0 to 20)
Byte 9	00h
Bytes 10-11	Spare

[Mode1-Mode0] = Keyboard Mode, 2b, 0=Disabled, 1=Read Only, 2=Read & Write
[Entry1-Entry0] = Keyboard Entry, 2b, 0=Quick, 1=Edit Only, 2=Confirm
[Actve1-Actve0] = Lcd Active Backlight, 2b, 0=Off, 1=1/3, 2=2/3, 3=Full
[Idle1-Idle0] = Lcd Idle Backlight, 2b, 0=Off, 1=1/3, 2=2/3, 3=Full
[Actvty1-Actvty0] = Key Activity, 2b, 0=None, 1=Beep, 2=Blink, 3=Beep & Blink

Unit IrDA, Command [03h],

Unit IrDA, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Rate	Format	Power	Activity	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit IrDA, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	IrDAEn	Mode0	Mode1	Rate0	Rate1	Rate2	Rate3	Frmt0
Byte 5	Frmt1	Frmt2	Pwr0	Pwr1	Actvty0	Actvty1	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit IrDA, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Rate	Format	Power	Activity	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit IrDA, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	Mode0	Mode1	Rate0	Rate1	Rate2	Rate3	Frmt0
Byte 5	Frmt1	Frmt2	Pwr0	Pwr1	Actvty0	Actvty1	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[Mode1-Mode0] = IrDA Mode, 2b, 0=Disabled, 1=Read Only, 2=Read & Write
[Rate3-Rate0] = IrDA Rate, 4b, 0=300, 1=600, 2=1200, 3=2400, 4=4800, 5=9600, 6=19200, 7=38400
[Frmt2-Frmt0] = IrDA Format, 3b, 0=N81, 1=E81, 2=O81, 3=M81, 4=S81
[Pwr1-Pwr0] = IrDA IR Power, 2b, 0=Low Power, 1=Medium Power, 2=Full Power
[Actvty1-Actvty0] = IrDA Activity, 2b, 0=None, 1=Beep, 2=Blink, 3=Beep & Blink

Unit Remote, Command [04h],

Unit Remote, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Protocol	Address	Rate	Format	Port	Activity	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Remote, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Proto0	Proto1	Proto2	Proto3	Rate0	Rate1
Byte 5	Rate2	Rate3	Frmt0	Frmt1	Frmt2	Port	Actvty0	Actvty1
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Remote, Read Bytes

Byte 8	Unit Address, 8b, (0 to 255), 0=None, 255=Global
Byte 9	00h
Bytes 10-11	Spare

Unit Remote, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Protocol	Address	Rate	Format	Port	Activity	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Remote, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Proto0	Proto1	Proto2	Proto3	Rate0	Rate1
Byte 5	Rate2	Rate3	Frmt0	Frmt1	Frmt2	Port	Actvty0	Actvty1
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Remote, Write Bytes

Byte 8	Unit Address, 8b, (0 to 255), 0=None, 255=Global
Byte 9	00h
Bytes 10-11	Spare

[Mode1-Mode0] = Remote Mode, 2b, 0=Disabled, 1=Read Only, 2=Read & Write
[Proto3-Proto0] = Remote Protocol, 4b, 0=VT100, 1=Quiet VT100, 2=Binary Packet A
[Rate3-Rate0] = Remote Rate, 4b, 0=300, 1=600, 2=1200, 3=2400, 4=4800, 5=9600, 6=19200, 7=38400
[Frmt2-Frmt0] = Remote Format, 3b, 0=N81, 1=E81, 2=O81, 3=M81, 4=S81
[Port] = Remote Port, 1b, 0=RS-232, 1=RS-485
[Actvty1-Actvty0] = Remote Activity, 2b, 0=None, 1=Beep, 2=Blink, 3=Beep & Blink

Unit Ref, Command [05h],
Unit Ref, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Source	Freq.	FineTune	Spare	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Ref, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Source	Freq0	Freq1	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Ref, Read Bytes

Bytes 6-7	Reference Fine Tune, Signed 16b, (-128 to +127)
Bytes 8-11	Spare

Unit Ref, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Source	Freq.	FineTune	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Ref, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Source	Freq0	Freq1	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Ref, Write Bytes

Bytes 6-7	Reference Fine Tune, Signed 16b, (-128 to +127), Approximately 0.1 PPM per Step
Bytes 8-11	Spare

[Source] = Reference Source, 1b, 0=Internal, 1=External
[Freq1-Freq0] = Reference Frequency, 2b, 0=1.0MHz, 1=5.0MHz, 2=9.0MHz, 3=10.0MHz

Unit Redundancy, Command [06h],

Unit Redundancy, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	SwRqst	SwHold	Config	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Redundancy, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	SwRqst0	SwRqst1	SndCfg	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Redundancy, Read Bytes

Bytes 8-9	Switch Hold Time, 16b, (0 to 6,000), 100ms Increments
Bytes 10-11	Spare

Unit Redundancy, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	SwRqst	SwHold	Config	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Redundancy, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	SwRqst0	SwRqst1	SndCfg	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Redundancy, Write Bytes

Bytes 8-9	Switch Hold Time, 16b, (0 to 6,000), 100ms Increments
Bytes 10-11	Spare

[Mode1-Mode0] = Redundancy Mode, 2b, 0=Disabled, 1=Internal 1:1, 2=External
[SwRqst1-SwRqst0] = Switch Request, 2b, 0=Any Alarm, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[SndCfg] = Config Backup, 1b, 0=Idle, 1=Send Config to Backup Unit (This Unit Must be Online)

Unit Monitor, Command [07h],

Unit Monitor, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Full	Zero	Spare	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Monitor, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Monitor, Read Bytes

Bytes 6-7	Full Scale Voltage, Signed 16b, (-100 to +100), 100mV Increments
Bytes 8-9	Zero Scale Voltage, Signed 16b, (-100 to +100), 100mV Increments

Unit Monitor, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	Full	Zero	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Monitor, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Monitor, Write Bytes

Bytes 6-7	Full Scale Voltage, Signed 16b, (-100 to +100), 100mV Increments
Bytes 8-9	Zero Scale Voltage, Signed 16b, (-100 to +100), 100mV Increments

[Mode1-Mode0] = Monitor Mode, 2b, 0=AGC Voltage, 1=Eb/No, 2=Mod CXR Level

Unit Alarm, Command [08h],

Unit Alarm, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Ref	TstActve	Hardware	Beep	OxAlm	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Alarm, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RefAlm0	RefAlm1	RefAlm2	TstAlm0	TstAlm1	HrdAlm0	HrdAlm1	Beep0
Byte 5	Beep1	OxAlm0	OxAlm1	OxAlm2	OxAlm3	RefAlm3	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Alarm, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	RefAlm	TstAlm	HrdAlm	Beep	OxAlm	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Alarm, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RefAlm0	RefAlm1	RefAlm2	TstAlm0	TstAlm1	HrdAlm0	HrdAlm1	Beep0
Byte 5	Beep1	OxAlm0	OxAlm1	OxAlm2	OxAlm3	RefAlm3	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[RefAlm3-RefAlm0] = Reference Alarm Mode, 3b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B, 4=Mute CXR, 5=Mute CXR & Alarm A, 6=Mute CXR & Alarm B, 7=Mute CXR & Alarm A&B, 8=Mute BUC/CXR, 9=Mute BUC/CXR & Alarm A, 10=Mute BUC/CXR & Alarm B, 11=Mute BUC/CXR & Alarm A&B (RefAlm3 Bit Available on Full L-Band Only - Mute BUC = 10 MHz Reference to BUC is Disabled On Reference Alarm)
[TstAlm1-TstAlm0] = Test Active Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[HrdAlm1-HrdAlm0] = Hardware Alarm Mode, 2b, 0=Mute CXR, 1= Mute CXR & Alarm A, 2= Mute CXR & Alarm B, 3= Mute CXR & Alarm A&B
[Beep1-Beep0] = Beeper Alarm Mode, 2b, 0=None, 1=On Alarm A, 2=On Alarm B, 3=On Alarm A&B
[OxAlm3-OxAlm0] = OCXO Oven Alarm Mode, 3b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B, 4=Mute CXR, 5=Mute CXR & Alarm A, 6=Mute CXR & Alarm B, 7=Mute CXR & Alarm A&B, 8=Mute BUC/CXR, 9=Mute BUC/CXR & Alarm A, 10=Mute BUC/CXR & Alarm B, 11=Mute BUC/CXR & Alarm A&B (OCXO Oven Alarm Available on Full L-Band Only - Mute BUC = 10 MHz Reference to BUC is Disabled On OCXO Alarm)

Unit Test, Command [09h],

Unit Test, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Modem	CalRef	UpROM	RefAFC	ClkAFC	+3.3V	+5.0V	+12.0V
Byte 1	+21.0V	-12.0V	BootCode	x	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Test, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Modem0	Modem1	Modem2	Modem3	CalRef	0	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Test, Read Bytes

Bytes 8-9	Ref AFC Voltage, Signed 16b, 100mV Increments
Bytes 10-11	SysClk AFC Voltage, Signed 16b, 100mV Increments
Bytes 12-13	+3.3 Voltage, Signed 16b, 100mV Increments
Bytes 14-15	+5.0 Voltage, Signed 16b, 100mV Increments
Bytes 16-17	+12.0 Voltage, Signed 16b, 100mV Increments
Bytes 18-19	+21.0 Voltage, Signed 16b, 100mV Increments
Bytes 20-21	-12.0 Voltage, Signed 16b, 100mV Increments
Bytes 22-23	Boot Error Code, 16b
Bytes 24-25	Boot Error Page Address, 16b
Bytes 26-27	Boot Error Segment Address, 16b
Byte 28	Self Test Step Status, 8b
Bytes 29-33	Self Test Status, 56b
Bytes 34-35	Reference Calibration Status, 16b
Bytes 36-41	Spare

Unit Test, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Modem	CalRef	UpROM	0	0	0	0	0
Byte 1	0	0	BootCode	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Unit Test, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Modem0	Modem1	Modem2	Modem3	CalRef	x	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Unit Test, Write Bytes

Bytes 8-11	*Confirm Serial Number, 32b, Required to Enable Flash ROM Update
Bytes 12-13	x
Bytes 14-15	x
Bytes 16-17	x
Bytes 18-19	x
Bytes 20-21	x
Bytes 22-23	x
Bytes 24-25	x
Bytes 26-27	x
Bytes 28-33	x
Bytes 34-35	x
Bytes 36-41	Spare

*****If **UpROM** Enabled then Serial Number Confirmation Required else don't care
[Modem3-Modem0] = Modem Self Test, 4b, 0=Normal, 1=Lamp Test, 2=Self Test, 3=Lamp & Self Test
[CalRef] = Reference Calibration, 1b, 0=Normal, 1=Enabled
[BootCode] = CPU Boot Code, 1b, 1=Reset Boot Code to Zero

Self Test Step Status

[Byte 28] Value: 1=Self Test Initializing, 2=Alarm On Test, 3=Upper Limit Test, 4=Lower Limit Test, 5=Loop Test 1 (4.92Mbps), 6=Loop Test 2 (2.4kbps), 100=Self Test Completed Successfully. This value is 0 if the self test is inactive and has not been activated since power up (no valid status to report). If the self test fails this value will stay on the step that failed and one of the error flags in bytes 29-33 will be set indicating what failed on that step.

Self Test Status

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 29	DHrdFail	MOpAlm Fail*	DOpAlm Fail*	Reserved	Reserved	Reserved	Reserved	Reserved
Byte 30	SysAlm Bad	MStpAlm Bad	MLoAlm Bad	DStpAlm Bad	DLoAlm Bad	MBitAlm Bad	3.3VAlm	5.0VAlm
Byte 31	12VAlm	21Valm	-12VAlm	SysAlm	MStpAlm	MLoAlm	DStpAlm	DLoAlm
Byte 32	MBitAlm	MlvlAlm	DIQAlm	OvrTAlm	SysAfc Alm	MStpAfc Alm	MLoAfc Alm	DStpAfc Alm
Byte 33	DLoAfc Alm	Lock Failed	EbNo Error	Offset Error	Level Error	Lock Error	SyncLoss Error	BER Error

Reference Calibration Status

[Bytes 34-35] Value: 1=Bad Input Reference, 2=DAC Error, 3=Flash Write Error, 4=Calibration Error, 5=Reference Calibration Completed Successfully. This value is 0 if the Reference Calibration process has not been activated since power up (no valid status to report).
***** If Option Board Installed

Mod Status, Command [40h],

Mod Status, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CXR	BitClk	Test	Spare	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Status, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrEn	CXRAlm	RTSMute	OptAlm	APCLmt	APCAlm	BitAlm	RefAlm
Byte 5	DLckAlm	LvlAlm	LoAlm	StpAlm	SysAlm	PureCXR	Alt1/0	Sideband
Byte 6	Online	OcxoAlm	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Status, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	0	0	0	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod Status, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	x	x	x	x	x	x	x
Byte 5	x	x	x	x	x	x	x	x
Byte 6	x	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

OcxoAlm = OCXO Cold Alarm on L-Band Only

Mod IF, Command [41h],

Mod IF, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Freq	Offset	Level	Output	Mod	Spectrum	Mode	Preamble
Byte 1	AUPC	AEbNo	AMaxLvl	AMinLvl	Mute	Imped	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod IF, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrEn	CxrHrd	Mod0	Mod1	Mod2	SpcInv	BurstInst	Mode
Byte 5	PreLng0	PreLng1	PreLng2	PreLng3	AUPCEn	Mute0	Mute1	ImpSel
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod IF, Read Bytes

Bytes 8-11	CXR Frequency, 32b, (50,000,000 to 90,000,000), 1Hz Increments
Bytes 12-13	CXR Frequency, L-Band Frequency Extension, 48b, (950,000,000 to 1750,000,000), 1Hz Inc.
Bytes 14-17	CXR Offset, Signed 32b, (-1,250,000 to +1,250,000), 1Hz Increments
Bytes 18-19	CXR Level, Signed 16b, (-350 to +50*), 0.1dB Increments (dBm)
Bytes 20-21	AUPC Eb/No, 16b, (30 to 200), 0.1dB Increments
Bytes 22-23	AUPC Max Level, Signed 16b, (-350 to +50*), 0.1dB Increments
Bytes 24-25	AUPC Min Level, Signed 16b, (-350 to +50*), 0.1dB Increments
Bytes 26-35	Spare

Mod IF, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Freq	Offset	Level	Output	Mod	Spectrum	Mode	Preamble
Byte 1	AUPC	AEbNo	AmaxLvl	AMinLvl	Mute	Imped	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod IF, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrEn	x	Mod0	Mod1	Mod2	SpcInv	x	Mode
Byte 5	PreLng0	PreLng1	PreLng2	PreLng3	AUPCEn	Mute0	Mute1	ImpSel
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod IF, Write Bytes

Bytes 8-11	Cxr Frequency, 32b, (50,000,000 to 90,000,000), 1Hz Increments
Bytes 12-13	CXR Frequency, L-Band Frequency Extension, 48b, (950,000,000 to 1750,000,000), 1Hz Inc.
Bytes 14-17	Cxr Offset, Signed 32b, (-1,250,000 to +1,250,000), 1Hz Increments
Bytes 18-19	Cxr Level, Signed 16b, (-350 to +50*), 0.1dB Increments (dBm)
Bytes 20-21	AUPC Eb/No, 16b, (30 to 200), 0.1dB Increments
Bytes 22-23	AUPC Max Level, Signed 16b, (-350 to +50*), 0.1dB Increments
Bytes 24-25	AUPC Min Level, Signed 16b, (-350 to +50*), 0.1dB Increments

Bytes 26-35	Spare

* +30 (+3.0dBm) When 50 Ohms Output Impedance Selected (70 MHz & 140 MHz IF Only).

[CxrEn] = Mod Cxr Output Enable, 1b, 0=Disabled, 1=Enabled
[Mod2-Mod0] = Mod Modulation mode, 3b, 0=BPSK, 1=QPSK
[SpcInv] = Mod Spectrum Invert, 1b, 0=Normal, 1=Inverted
[Mode] = Mod Mode, 1b, 0=Normal, 1=Burst. Only valid setting and return value if the modem is fitted with a burst option.
[PreLng3-PreLng0] = Mod Burst Preamble Length, 4b, 0=32 Symbols, 1=48 Symbols, 2=64 Symbols. Only valid setting and return value if the modem is fitted with a burst option.
[AUPCEn] = Mod AUPC Mode, 1b, 0=Disabled, 1=Enabled
[Mute1-Mute0] = Mod Cxr Mute Mode, 2b, 0=Automatic, 1=Confirm, 2=Manual
[ImpSel] = Mod Output Impedance, 1b, 0=50 Ohms, 1=75 Ohms

Notes on Carrier Frequency and LO Settings:

1. 50 to 90 MHz or 100 to 180 MHz IF type CXR Frequency entries use 4 bytes. L-Band or RF frequency entries use 6 bytes. (Also LO frequencies used in BUC and LNB commands.)
2. All entries assign the Least Significant Byte (LSB) to the lower byte number in the packet. i.e. Byte 12 of the packet data is the LSB of the CXR Frequency.
3. The Upconverter (BUC) LO frequency for the PSM-4900L must be set before an RF frequency relative to that new LO is entered into the CXR Frequency assignment. Future software versions will allow setting both within the same packet message.
4. Setting the Upconverter LO frequency to “0” specifies using L-Band CXR Frequencies in modems with an L-Band interface.

Mod Data, Command [42h],

Mod Data, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BitRate	Mod	FecType	CodeRate	DifEnc	Scrmbler	ClkSrc	FecC0C1
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Data, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mod0	Mod1	Mod2	SeqInst	TrbInst	0	0	0
Byte 5	0	FecType0	CRate0	CRate1	Crate2	CRate3	DifEnc0	DifEnc1
Byte 6	Scrm0	Scrm1	Scrm2	Scrm3	ClkSrc0	ClkSrc1	ClkSrc2	Fec0
Byte 7	Fec1	FecType1	FecType2	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Data, Read Bytes

Bytes 10-13	Bit Rate, 32b, (1,200 to 4,920,000), 1bps Increments (Depends on Mode)
Bytes 14-19	Spare

Mod Data, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BitRate	Mod	FecType	CodeRate	DifEnc	Scrmbler	ClkSrc	FecC0C1
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod Data, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mod0	Mod1	Mod2	x	x	0	0	0
Byte 5	0	FecType0	CRate0	CRate1	Crate2	CRate3	DifEnc0	DifEnc1
Byte 6	Scrm0	Scrm1	Scrm2	Scrm3	ClkSrc0	ClkSrc1	ClkSrc2	Fec0
Byte 7	Fec1	FecType1	FecType2	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Data, Write Bytes

Bytes 10-13	Bit Rate, 32b, (1,200 to 4,920,000), 1bps Increments (Depends on Mode)
Bytes 14-19	Spare

[Mod2-Mod0] = Mod Modulation Mode, 3b, 0=BPSK, 1=QPSK
[FecType2-FecType0] = Mod Fec Mode, 3b, Depends on Hardware FEC Option Installed.
With Optional TPC FEC Installed: 0=Viterbi, 1=TPC Full, 2=TPC Short, 3=TPC Legacy, 4=TPC CT (Rate ¾ Only).
With Optional TCT FEC Installed: 0=Viterbi, 1=TCT Full, 2=TCT Medium, 3=TCT Short.
[CRate3-CRate0] = Mod Code Rate, 4b, 0=Rate 1/2, 1=Rate 3/4, 2=Rate 7/8
[DifEnc1-DifEnc0] = Mod Diff Encoder, 2b, 0=Disabled, 1=Enabled, 2=Differentially Coherent
[Scrm3-Scrm0] = Mod Scrambler, 4b, 0=Disabled, 1=IESS-308, 2=IESS309, 3=V.35, 4=Alt V.35, 5=Intelsat, 6=Alt Intelsat, 7=TPC Synchronous (If TPC Installed)
[ClkSrc2-ClkSrc0] = Mod Clock Source, 3b, 0=Internal, 1=Terminal Timing, 2=External, 3=RCV
[Fec1-Fec0] = Fec C0/C1 Mode, 2b, 0=Normal, 1=Swap C0/C1, 2=Invert C1, 3=Swap C0/C1 & Invert C1

Mod Alarm, Command [43h],

Mod Alarm, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CxrAlm	BitAlm	ApcAlm	TstAlm	HrdAlm	BucAlm	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Alarm, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrAlm0	CxrAlm1	BitAlm0	BitAlm1	BitAlm2	BitAlm3	ApcAlm0	ApcAlm1
Byte 5	TstAlm0	TstAlm1	HrdAlm0	HrdAlm1	BucAlm0	BucAlm1	BucAlm2	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Alarm, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CxrAlm	BitAlm	ApcAlm	TstAlm	HrdAlm	BucAlm	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod Alarm, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrAlm0	CxrAlm1	BitAlm0	BitAlm1	BitAlm2	BitAlm3	ApcAlm0	ApcAlm1
Byte 5	TstAlm0	TstAlm1	HrdAlm0	HrdAlm1	BucAlm0	BucAlm1	BucAlm2	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[CxrAlm1-CxrAlm0] = Mod Cxr Alarm Mode, 2b, 0=Mute Cxr, 1= Mute Cxr & Alarm A, 2= Mute Cxr & Alarm B, 3= Mute Cxr & Alarm A&B
[BitAlm3-BitAlm0] = Mod Bit Clk Alarm Mode, 4b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B, 4=Send A1S, 5=Send A1S & Alarm A, 6=Send A1S & Alarm B, 7=Send A1S & Alarm A&B, 8=Mute Cxr, 9=Mute Cxr & Alarm A, 10=Mute Cxr & Alarm B, 11=Mute Cxr & Alarm A&B
[ApcAlm1-ApcAlm0] = Mod AUPC Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[TstAlm1-TstAlm0] = Mod Test Active Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[HrdAlm1-HrdAlm0] = Mod Hardware Alarm Mode, 2b, 0=Mute CXR, 1= Mute CXR & Alarm A, 2= Mute CXR & Alarm B, 3= Mute CXR & Alarm A&B

The Following are L-Band Only.
[BucAlm2-BucAlm0] = Mod BUC Power Alarm Mode, 3b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B, 4=Mute CXR, 5=Mute CXR & Alarm A, 6=Mute CXR & Alarm B, 7=Mute CXR & Alarm A&B

Mod Test, Command [44h],

Mod Test, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TstMod	Cxr ALC	LoAFC	StpAFC	x	x	x	x
Byte 1	Reserved	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Test, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TstMod0	TstMod1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Test, Read Bytes

Bytes 6-7	Cxr ALC Voltage, Signed 16b, 100mV Increments
Bytes 8-9	LO AFC Voltage, Signed 16b, 100mV Increments
Bytes 10-11	Step AFC Voltage, Signed 16b, 100mV Increments
Bytes 12-15	Symbol Rate, Unsigned 32b, 1 symbol per second Increments
Bytes 14-15	(Reserved) All Zeros
Bytes 16-17	(Reserved) All Zeros
Bytes 18-19	(Reserved) All Zeros
Bytes 20-21	(Reserved) All Zeros
Bytes 22-31	Spare

Mod Test, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TstMod	0	0	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod Test, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TstMod0	TstMod1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Test, Write Bytes

Bytes 6-7	x
Bytes 8-9	x
Bytes 10-11	x
Bytes 12-13	x
Bytes 14-15	x
Bytes 16-17	x
Bytes 18-19	x
Bytes 20-21	x
Bytes 22-31	Spare

[TstMod1-TstMod0] = Mod Test Modulation, 2b, 0=Normal, 1=Pure Cxr, 2=Alt 1/0, 3=Sideband

Mod RS Fec, Command [45h],

Mod RS Fec, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	N	K	Depth	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod RS Fec, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RsMode0	RsMode1	Depth	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod RS Fec, Read Bytes

Byte 6	Reed-Solomon Codeword Length (n), Unsigned 8b, (22 to 255)
Byte 7	Reed-Solomon Message Length (k), Unsigned 8b, (20 to 253)
Bytes 8-9	Spare

Mod RS Fec, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	N	K	Depth	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod RS Fec, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RsMode0	RsMode1	Depth	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod RS Fec, Write Bytes

Byte 6	Reed-Solomon Codeword Length (n), Unsigned 8b, (22 to 255), Custom Mode Only
Byte 7	Reed-Solomon Message Length (k), Unsigned 8b, (20 to 253), Custom Mode Only
Bytes 8-9	Spare

[RsMode1-RsMode0] = RS Mode, 2b, 0=Disabled, 1=IESS-308, 2=IESS-309, 3=Custom
[Depth] = RS Interleave Depth (Custom Mode Only), 1b, 0=4, 1=8

Mod Mux, Command [46h],

Mod Mux, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	ESC Ovrhead	MCC Ovrhead	Ratio	ESC Port	ESC Rate	ESC Format	ESC DTR
Byte 1	ESC DSR	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Mux, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	EscOh0	EscOh1	EscOh2	EscOh3	MccOh0	MccOh1
Byte 5	MccOh2	MccOh3	Port0	Port1	Rate0	Rate1	Rate2	Rate3
Byte 6	Frmt0	Frmt1	CtsMd	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Mux, Read Bytes

Bytes 12-13	Mux Ratio X, Unsigned 16b, (1 to 255), X**:**Y
Bytes 14-15	Mux Ratio Y, Unsigned 16b, (2 to 256), X**:**Y
Bytes 16-23	Spare

Mod Mux, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mux Mode	ESC Ovrhead	MCC Ovrhead	0	ESC Port	ESC Rate	ESC Format	ESC CTS
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod Mux, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	EscOh0	EscOh1	EscOh2	EscOh3	MccOh0	MccOh1
Byte 5	MccOh2	MccOh3	Port0	Port1	Rate0	Rate1	Rate2	Rate3
Byte 6	Frmt0	Frmt1	CtsMd	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod Mux, Write Bytes

Bytes 12-13	x
Bytes 14-15	x
Bytes 16-23	Spare

[Mode1-Mode0] = Mode, 2b, 0=Disabled, 1=Standard IBS, 2=Enhanced IBS, 3=Custom IBS
[EscOh3-EscOh0] = ESC Overhead (Custom Mode Only), 4b, 0=Disabled, 1=300, 2=600, 3=1200, 4=2400, 5=4800, 6=9600, 7=19200, 8=38400
[MccOh3-MccOh0] = MCC Overhead (Custom Mode Only), 4b, 0=Disabled, 1=300, 2=600, 3=1200, 4=2400, 5=4800, 6=9600, 7=19200, 8=38400
[Port1-Port0] = ESC Port, 2b, 0=RS-232, 1=RS-485 2-Wire, 2=RS-485 4-Wire, 3=RS-485 4-Wire Driver On
[Rate3-Rate0] = ESC Rate, 4b, 0=300, 1=600, 2=1200, 3=2400, 4=4800, 5=9600, 6=19200, 7=38400
[Frmt1-Frmt0] = ESC Format, 2b, 0=N71, 1=P71, 2=N81, 3=P81
[CtsMd] = ESC CTS Mode, 1b, 0=Normal (Xmt Flow Control), 1=Ignore

Mod BUC, Command [47h],
Read Change Flags (L-Band Only)

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BucPwr	VOut	VMin	IOut	IMax	IMin	Ref	LoFrq
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod BUC, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	PwrEn0	PwrEn1	PwrEn2	RefEn0	RefEn1	RefEn2	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod BUC, Read Bytes

Bytes 8-9	BUC Voltage Out, Signed 16b, (0 to >600), 100mV Increments
Bytes 10-11	BUC Voltage Min, Signed 16b, (80 to 600), 100mV Increments
Bytes 12-13	BUC Current Out, Signed 16b, (0 to >600), 10mA Increments
Bytes 14-15	BUC Current Max, Signed 16b, (5 to 600), 10mA Increments
Bytes 16-17	BUC Current Min, Signed 16b, (5 to 600), 10mA Increments
Bytes 18-23	BUC LO Frequency, Unsigned 48b, (0 to 50,000,000,000), 1Hz Increments
Bytes 24-33	Spare

Mod BUC, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BucPwr	0	VMin	0	IMax	IMin	Ref	LoFrq
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Mod BUC, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	PwrEn0	PwrEn1	PwrEn2	RefEn0	RefEn1	RefEn2	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Mod BUC, Write Bytes

Bytes 8-9	x
Bytes 10-11	BUC Voltage Min, Signed 16b, (80 to 600), 100mV Increments
Bytes 12-13	x
Bytes 14-15	BUC Current Max, Signed 16b, (5 to 600), 10mA Increments
Bytes 16-17	BUC Current Min, Signed 16b, (5 to 600), 10mA Increments
Bytes 18-23	BUC LO Frequency, Unsigned 48b, (0 to 50,000,000,000), 1Hz Increments
Bytes 24-33	Spare

[PwrEn2-PwrEn0] = BUC Power, 3b, 0=Disabled, 1=Enabled
[RefEn2-RefEn0] = BUC 10MHz Ref, 3b, 0=Disabled, 1=Enabled

Demod Status, Command [80h],

Demod Status, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CXR	Offset	Level	Eb/No	EstBER	SER	Buffer	Test
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Status, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	DemLck	RcvLck	NrwHld	FecLck	RsAlm	MuxAlm	BckAlm	LvlAlm
Byte 5	EbAlm	AgcAlm	RefAlm	LoAlm	StpAlm	SysAlm	IfLoop	FixCxr
Byte 6	BufValid	BufSlip	BufSlip Sign	OptClk Alm	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Status, Read Bytes

Bytes 8-11	Rcv Offset, Signed 32b, (-1,250,000 to +1,250,000), 1Hz Increments
Bytes 12-13	Rcv Cxr Level, Signed 16b, (<-84 to 0), 1dB Increments (dBm)
Bytes 14-15	Rcv Eb/No, 16b, (20 to 200), 0.1dB Increments
Bytes 16-17	Est. BER, 16b, Bits [15-12]=Negative Exponent, Bits [11-0]=Mantissa (1-9, >9=<1)
Bytes 18-19	SER, 16b, Bits [15-12]=Negative Exponent, Bits [11-0]=Mantissa (100=1.00)
Bytes 20-21	Buffer, 16b, (0 to 200), 1% Increments
Bytes 22-27	Spare

Demod Status, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CxrLck	SwpStrt	0	0	EstBER	SER	Buffer	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod Status, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	AbortLck	CtrBuf	x	x	x	x	x	x
Byte 5	x	x	x	x	x	x	x	x
Byte 6	x	x	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Status, Write Bytes

Bytes 8-11	Sweep Start Frequency, Signed 32b, (-1,250,000 to +1,250,000), 1Hz Increments
Bytes 12-13	x
Bytes 14-15	x
Bytes 16-17	x, Write Restarts SER/Est. BER
Bytes 18-19	x, Write Restarts SER/Est. BER
Bytes 20-21	x, Write Clears Slip Flag
Bytes 22-27	Spare

[AbortLck] = Demod Abort Current Lock, 1b, 0=No Change, 1=Abort Lock. The "CxrLck" bit (byte 0, bit 0) is the write enable flag for AbortLck. No status read bit is available for Abort Lock.
[CtrBuf] = Demod Recenter Buffer, 1b, 0=Clear Slip Flag Only, 1=Clear Slip Flag & Recenter Buffer. The "Buffer" bit (Byte 0, bit 6) is the write enable flag for CtrBuf. No status read bit is available for Center Buffer.

Demod IF, Command [81h],

Demod IF, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Freq	SwpRng	SwpMd	SwpTm	Mod	Spectrum	LowLvl	LowEb
Byte 1	Imped	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod IF, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	SwpMd0	SwpMd1	Mod0	Mod1	Mod2	SpcInv	ImpSel	LnbPwr0
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod IF, Read Bytes

Bytes 8-11	CXR Frequency, 32b, (50,000,000 to 90,000,000), 1Hz Increments.
Bytes 12-13	CXR Frequency, L-Band Extension, 48b, (950,000,000 to 1900,000,000), 1Hz Increments
Bytes 14-17	Sweep Range, 32b, (100 to 1,250,000), 1Hz Increments.
Bytes 18-19	Sweep Time, 16b, (0 to 6,000), 100ms Increments, 0=Narrow Disabled.
Bytes 20-21	Low Level, Signed 16b, (-84 to -20), 1dB Increments (Dependent on Symbol Rate).*
Bytes 22-23	Low Eb/No, 16b, (20 to 200), 0.1dB Increments.
Bytes 24-33	Spare

Demod IF, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Freq	SwpRng	SwpMd	SwpTm	Mod	Spectrum	LowLvl	LowEb
Byte 1	Imped	LnbPwr	LnbRef	LnbLO	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod IF, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	SwpMd0	SwpMd1	Mod0	Mod1	Mod2	SpcInv	ImpSel	LnbPwr0
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod IF, Write Bytes

Bytes 8-11	CXR Frequency, 32b, (50,000,000 to 90,000,000), 1Hz Increments
Bytes 12-13	CXR Frequency, L-Band Extension, 48b, (950,000,000 to 1,900,000,000), 1Hz Increments
Bytes 14-17	Sweep Range, 32b, (100 to 1,250,000), 1Hz Increments
Bytes 18-19	Sweep Time, 16b, (0 to 6,000), 100ms Increments, 0=Narrow Disabled
Bytes 20-21	Low Level, Signed 16b, (-84 to -20), 1dB Increments (Dependent on Symbol Rate)*
Bytes 22-23	Low Eb/No, 16b, (20 to 200), 0.1dB Increments
Bytes 24-33	Spare

[SwpMd1-SwpMd0] = Demod Sweep Mode, 2b, 0=Normal, 1=Search
[Mod2-Mod0] = Demod Modulation Mode, 3b, 0=BPSK, 1=QPSK
[SpcInv] = Demod Spectrum Invert, 1b, 0=Normal, 1=Inverted
[ImpSel] = Demod Input Impedance, 1b, 0=50 Ohms, 1=75 Ohms (Fixed 75 Ohms with L-Band)
***** Hybrid and L-Band Versions Use Different Range and is Dependent on Symbol Rate

Notes on Carrier Frequency and LO Settings:

5. 50 to 90 MHz or 100 to 180 MHz IF type CXR Frequency entries use 4 bytes. L-Band or RF frequency entries use 6 bytes. (Also LO frequencies used in BUC and LNB commands.)
6. All entries assign the Least Significant Byte (LSB) to the lower byte number in the packet. i.e. Byte 12 of the packet data is the LSB of the CXR Frequency.
7. The Downconverter (LNB) LO frequency for the PSM-4900L & H must be set before an RF frequency relative to that new LO is entered into the CXR Frequency assignment. Future software versions will allow setting both within the same packet message.
8. Setting the Downconverter LO frequency to “0” specifies using L-Band CXR Frequencies in modems with an L-Band interface.

Demod Data, Command [82h],

Demod Data, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BitRate	Mod	FecType	CodeRate	DifEnc	Scrmbler	ClkSrc	BufDly
Byte 1	BufSize	FecC0C1	FecHold	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Data, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mod0	Mod1	Mod2	SeqInst	TrbInst	0	0	0
Byte 5	0	FecType0	CRate0	CRate1	Crate2	CRate3	DifEnc	0
Byte 6	Scrm0	Scrm1	Scrm2	Scrm3	ClkSrc0	ClkSrc1	ClkSrc2	Fec0
Byte 7	Fec1	FecType1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Data, Read Bytes

Bytes 10-13	Bit Rate, 32b, (1,200 to 4,920,000), 1bps Increments (Depends on Mode)
Bytes 14-17	Buffer Delay, 32b, (8 to >266,000), 100ns Increments (Depends on Bit Rate)
Bytes 18-21	Buffer Size, 32b, (4 to 131,070), 1 Bit Increments
Bytes 22-23	Fec Hold Count, 16b, (0 to 255)
Bytes 24-29	Spare

Demod Data, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BitRate	Mod	FecType	CodeRate	DifEnc	Scrmbler	ClkSrc	BufDly
Byte 1	BufSize	FecC0C1	FecHold	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod Data, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mod0	Mod1	Mod2	SeqInst	TrbInst	0	0	0
Byte 5	0	FecType0	CRate0	CRate1	Crate2	CRate3	DifEnc	0
Byte 6	Scrm0	Scrm1	Scrm2	Scrm3	ClkSrc0	ClkSrc1	ClkSrc2	Fec0
Byte 7	Fec1	FecType1	FecType2	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Data, Write Bytes

Bytes 10-13	Bit Rate, 32b, (1,200 to 4,920,000), 1bps Increments (Depends on Mode)
Bytes 14-17	Buffer Delay, 32b, (8 to >266,000), 100ns Increments (Depends on Bit Rate)
Bytes 18-21	Buffer Size, 32b, (4 to 131,070), 1 Bit Increments
Bytes 22-23	Fec Hold Count, 16b, (0 to 255)
Bytes 24-29	Spare

[Mod2-Mod0] = Demod Modulation Mode, 3b, 0=BPSK, 1=QPSK

[FecType2-FecType0] = Demod Fec Mode, 3b, Depends on Hardware FEC Option Installed.
With Optional TPC FEC Installed: 0=Viterbi, 1=TPC Full, 2=TPC Short, 3=TPC Legacy, 4=TPC CT (Rate ¾ Only).
With Optional TCT FEC Installed: 0=Viterbi, 1=TCT Full, 2=TCT Medium, 3=TCT Short.
[CRate3-CRate0] = Demod Code Rate, 4b, 0=Rate 1/2, 1=Rate 3/4, 2=Rate 7/8
[DifEnc] = Demod Diff Decoder, 2b, 0=Disabled, 1=Enabled
[Scrm3-Scrm0] = Demod Descrambler, 4b, 0=Disabled, 1=IESS-308, 2=IESS309, 3=V.35, 4=Alt V.35, 5=Intelsat, 6=Alt Intelsat, 7=TPC Synchronous (If TPC Installed)
[ClkSrc2-ClkSrc0] = Demod Clock Source, 3b, 0=Rcv, 1=Internal, 2=External, 3=Mod Clk
[Fec1-Fec0] = Fec C0/C1 Mode, 2b, 0=Normal, 1=Swap C0/C1, 2=Invert C1, 3=Swap C0/C1 & Invert C1

Demod Alarm, Command [83h],

Demod Alarm, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CxrAlm	LvlAlm	EbAlm	TstAlm	HrdAlm	BckAlm	LnbAlm	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Alarm, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrAlm0	CxrAlm1	CxrAlm2	LvlAlm0	LvlAlm1	EbAlm0	EbAlm1	TstAlm0
Byte 5	TstAlm1	HrdAlm0	HrdAlm1	BckAlm0	BckAlm1	LnbAlm0	LnbAlm1	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Alarm, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	CxrAlm	LvlAlm	EbAlm	TstAlm	HrdAlm	BckAlm	LnbAlm	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod Alarm, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	CxrAlm0	CxrAlm1	CxrAlm2	LvlAlm0	LvlAlm1	EbAlm0	EbAlm1	TstAlm0
Byte 5	TstAlm1	HrdAlm0	HrdAlm1	BckAlm0	BckAlm1	LnbAlm0	LnbAlm1	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[CxrAlm2-CxrAlm0] = Demod Cxr Lock Alarm Mode, 3b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B, 4=Mute CXR, 5=Mute CXR & Alarm A, 6=Mute CXR & Alarm B, 7=Mute CXR & Alarm A&B
[LvlAlm1-LvlAlm0] = Demod Level Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[EbAlm1-EbAlm0] = Demod Eb/No Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[TstAlm1-TstAlm0] = Demod Test Active Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[HrdAlm1-HrdAlm0] = Demod Hardware Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[BckAlm1-BckAlm0] = Demod Backward Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
The Following are Hybrid & L-Band Only
[LnbAlm1-LnbAlm0] = Demod LNB Power Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B (L-Band Only)

Demod Test, Command [84h],

Demod Test, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	IfLoop	x	AGC	LoAFC	StpAFC	IDcOff	QDcOff	x
Byte 1	x	SymRate	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Test, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	IfLoop0	IfLoop1	Reserved	Reserved	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Test, Read Bytes

Bytes 6-7	AGC Voltage, Signed 16b, 100mV Increments
Bytes 8-9	LO AFC Voltage, Signed 16b, 100mV Increments
Bytes 10-11	Step AFC Voltage, Signed 16b, 100mV Increments
Bytes 12-13	IDcOff Voltage, Signed 16b, 100mV Increments
Bytes 14-15	QDcOff Voltage, Signed 16b, 100mV Increments
Bytes 16-17	(Reserved) All Zeros
Bytes 18-19	(Reserved) All Zeros
Bytes 20-29	Spare

Demod Test, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	IfLoop	0	AGC	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod Test, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	IfLoop0	IfLoop1	x	x	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Test, Write Bytes

Bytes 6-7	x
Bytes 8-9	x
Bytes 10-11	x
Bytes 12-13	x
Bytes 14-15	x
Bytes 16-17	x
Bytes 18-19	x
Bytes 20-29	Spare

[IfLoop1-IfLoop0] = Demod IF Loopback, 2b, 0=Disabled, 1=Enabled

Demod RS Fec, Command [85h],

Demod RS Fec, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	N	K	Depth	EbNo	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod RS Fec, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RsMode0	RsMode1	Depth	EbNo	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod RS Fec, Read Bytes

Byte 6	Reed-Solomon Codeword Length (n), Unsigned 8b, (22 to 255)
Byte 7	Reed-Solomon Message Length (k), Unsigned 8b, (20 to 253)
Bytes 8-9	Spare

Demod RS Fec, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	N	K	Depth	EbNo	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod RS Fec, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	RsMode0	RsMode1	Depth	EbNo	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod RS Fec, Write Bytes

Byte 6	Reed-Solomon Codeword Length (n), Unsigned 8b, (22 to 255), Custom Mode Only
Byte 7	Reed-Solomon Message Length (k), Unsigned 8b, (20 to 253), Custom Mode Only
Bytes 8-9	Spare

[RsMode1-RsMode0] = RS Mode, 2b, 0=Disabled, 1=IESS-308, 2=IESS-309, 3=Custom
[Depth] = RS Interleave Depth (Custom Mode Only), 1b, 0=4, 1=8
[EbNo] = RS Eb/No Mode, 1b, 0=Calculated at RS Decoder Input, 1=Calculated at RS Decoder Output

Demod Mux, Command [86h],

Demod Mux, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	ESC Ovrhead	MCC Ovrhead	Ratio	ESC Port	ESC Rate	ESC Format	ESC DTR
Byte 1	ESC DSR	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Mux, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	EscOh0	EscOh1	EscOh2	EscOh3	MccOh0	MccOh1
Byte 5	MccOh2	MccOh3	Port0	Port1	Rate0	Rate1	Rate2	Rate3
Byte 6	Frmt0	Frmt1	DtrMd	DsrMd	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Mux, Read Bytes

Bytes 12-13	Mux Ratio X, Unsigned 16b, (1 to 255), X:Y
Bytes 14-15	Mux Ratio Y, Unsigned 16b, (2 to 256), X:Y
Bytes 16-23	Spare

Demod Mux, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mux Mode	ESC Ovrhead	MCC Ovrhead	0	ESC Port	ESC Rate	ESC Format	ESC DTR
Byte 1	ESC DSR	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod Mux, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	EscOh0	EscOh1	EscOh2	EscOh3	MccOh0	MccOh1
Byte 5	MccOh2	MccOh3	Port0	Port1	Rate0	Rate1	Rate2	Rate3
Byte 6	Frmt0	Frmt1	DtrMd	DsrMd	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod Mux, Write Bytes

Bytes 12-13	x
Bytes 14-15	x
Bytes 16-23	Spare

[Mode1-Mode0] = Mode, 2b, 0=Disabled, 1=Standard IBS, 2=Enhanced IBS, 3=Custom IBS
[EscOh3-EscOh0] = ESC Overhead (Custom Mode Only), 4b, 0=Disabled, 1=300, 2=600, 3=1200, 4=2400, 5=4800, 6=9600, 7=19200, 8=38400
[MccOh3-MccOh0] = MCC Overhead (Custom Mode Only), 4b, 0=Disabled, 1=300, 2=600, 3=1200, 4=2400, 5=4800, 6=9600, 7=19200, 8=38400
[Port1-Port0] = ESC Port, 2b, 0=RS-232, 1=RS-485 2-Wire, 2=RS-485 4-Wire, 3=RS-485 4-Wire Driver On
[Rate3-Rate0] = ESC Rate, 4b, 0=300, 1=600, 2=1200, 3=2400, 4=4800, 5=9600, 6=19200, 7=38400
[Frmt1-Frmt0] = ESC Format, 2b, 0=N71, 1=P71, 2=N81, 3=P81
[DtrMd] = ESC DTR Mode, 1b, 0=Normal (Rcv Flow Control), 1=Ignore
[DsrMd] = ESC DSR Mode, 1b, 0=Normal, 1=Force Active

Demod LNB, Command [87h],
Read Change Flags (Hybrid & L-Band Only)

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	LnbPwr	Reserved	Reserved	IOut	IMax	IMin	Ref	LoFrq
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod LNB, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	PwrEn0	PwrEn1	PwrEn2	RefEn0	RefEn1	RefEn2	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod LNB, Read Bytes

Bytes 8-9	Reserved
Bytes 10-11	Reserved
Bytes 12-13	LNB Current Out, Signed 16b, (0 to >500), 1mA Increments
Bytes 14-15	LNB Current Max, Signed 16b, (5 to 500), 1mA Increments
Bytes 16-17	LNB Current Min, Signed 16b, (5 to 500), 1mA Increments
Bytes 18-23	LNB LO Frequency, Unsigned 48b, (0 to 50,000,000,000), 1Hz Increments
Bytes 24-33	Spare

Demod LNB, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	BucPwr	0	0	0	IMax	IMin	Ref	LoFrq
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Demod LNB, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	PwrEn0	PwrEn1	PwrEn2	RefEn0	RefEn1	RefEn2	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Demod LNB, Write Bytes

Bytes 8-9	X
Bytes 10-11	X
Bytes 12-13	X
Bytes 14-15	LNB Current Max, Signed 16b, (5 to 500), 1mA Increments
Bytes 16-17	LNB Current Min, Signed 16b, (5 to 500), 1mA Increments
Bytes 18-23	LNB LO Frequency, Unsigned 48b, (0 to 50,000,000,000), 1Hz Increments
Bytes 24-33	Spare

[PwrEn2-PwrEn0] = LNB Power, 3b, 0=Disabled, 1=Enabled
[RefEn2-RefEn0] = LNB 10MHz Ref, 3b, 0=Disabled, 1=Enabled

Interface Status, Command [C0h],

Interface Status, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	I/O	RTS	CTS	DCD	DTR	DSR	Test	TstBER
Byte 1	SynLoss	Errors	Bits	EFS	ErrSec	TotSec	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Status, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	OnLine	0	0	0	RtsN	CtsN	DcdN	DtrN
Byte 5	DsrN	MPtrnEn	DPtrnEn	DPtnLck	TerLoop	SatLoop	Int'f* Failure	Int'f* Reset
Byte 6	Int'f* Alarm	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Status, Read Bytes

Bytes 8-9	Test BER, 16b, Mantissa (1000=1.000)
Bytes 10-11	Test BER, 16b, Negative Exponent
Bytes 12-15	Test BER Sync Loss Count, 32b
Bytes 16-23	Test Error Count, 64b
Bytes 24-31	Test Bit Count, 64b
Bytes 32-33	Test Error Free Seconds, 16b, (0 to 10000), 0.01% Increments
Bytes 34-37	Test Erred Seconds, 32b
Bytes 38-41	Test Total Elapsed Seconds, 32b
Bytes 42-53	Spare

Interface Status, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	0	0	0	0	0	0	0	TstBER
Byte 1	SynLoss	Errors	Bits	EFS	ErrSec	TotSec	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Interface Status, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	0	0	0	x	x	x	x
Byte 5	x	x	x	x	x	x	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Status, Write Bytes

Bytes 8-9	x, Write Restarts BER Test
Bytes 10-11	x, Write Restarts BER Test
Bytes 12-15	x, Write Restarts BER Test
Bytes 16-23	x, Write Restarts BER Test
Bytes 24-31	x, Write Restarts BER Test
Bytes 32-33	x, Write Restarts BER Test
Bytes 34-37	x, Write Restarts BER Test
Bytes 38-41	x, Write Restarts BER Test
Bytes 42-53	Spare

Int'f* is Optional Interface status. Optional Interfaces are currently SDMS (Ethernet Type 1), T1, E1 and Drop & Insert.

Interface I/O, Command [C1h],

Interface I/O, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	RTS	CTS	DCD	DTR	DSR	XData	XClock
Byte 1	RData	RClock	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface I/O, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	RtsMd0	RtsMd1
Byte 5	RtsMd2	CtsMode	DcdMode	DtrMode	DsrMode	XData	XClock0	RData
Byte 6	RClock	XClock1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface I/O, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Mode	RTS	CTS	DCD	DTR	DSR	XData	XClock
Byte 1	RData	RClock	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Interface I/O, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Mode0	Mode1	Mode2	Mode3	Mode4	Mode5	RtsMd0	RtsMd1
Byte 5	RtsMd2	CtsMode	DcdMode	DtrMode	DsrMode	XData	XClock0	RData
Byte 6	RClock	XClock1	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[Mode5-Mode0] = Interface Mode, 6b, 0=Disable, 1=RS-232, 2=RS-449, 3=RS-449/Unterm, 4=V.35, 5=V.36, 6=EIA-530, 7=EIA-530A, 8 = SDMS*, 9=T1*, 10=E1* *If Installed
[RtsMd2-RtsMd0] = RTS Mode, 3b, 0=Normal, 1=Control Mod CXR, 2=Ignore
[CtsMode] = CTS Mode, 1b, 0=Normal, 1=Force Active
[DcdMode] = DCD Mode, 1b, 0=Normal, 1=Force Active
[DtrMode] = DTR Mode, 1b, 0=Normal, 1=Ignore
[DsrMode] = DSR Mode, 1b, 0=Normal, 1=Force Active
[XData] = Xmt Data Mode, 1b, 0=Normal, 1=Inverted
[XClock1-XClock0] = Xmt Clock Mode, 2b, 0=Normal, 1=Inverted, 2=Auto
[RData] = Rcv Data Mode, 1b, 0=Normal, 1=Inverted
[RClock] = Rcv Clock Mode, 1b, 0=Normal, 1=Inverted

Interface Alarm, Command [C2h],

Interface Alarm, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TstAlm	BerAlm	SDMS Alm	Spare	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Alarm, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TstAlm0	TstAlm1	BerAlm0	BerAlm1	SDMS Alm0	SDMS Alm1	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Alarm, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TstAlm	BerAlm	SDMS Alm	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Interface Alarm, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TstAlm0	TstAlm1	BerAlm0	BerAlm1	SDMS Alm0	SDMS Alm1	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[TstAlm1-TstAlm0] = Interface Test Active Alarm Mode, 2b, 0=None, 1= Alarm A, 2=Alarm B, 3=Alarm A&B
[BerAlm1-BerAlm0] = Interface BER Sync Loss Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B
[SDMSAlm1-SDMSAlm0] = Interface SDMS Alarm Mode, 2b, 0=None, 1=Alarm A, 2=Alarm B, 3=Alarm A&B

Interface Test, Command [C3h],

Interface Test, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TerLoop	SatLoop	ModBer	DemBer	Spare	Spare	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Test, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TerLoop	SatLoop	ModBer0	ModBer1	ModBer2	ModBer3	DemBer0	DemBer1
Byte 5	DemBer2	DemBer3	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface Test, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	TerLoop	SatLoop	ModBer	DemBer	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Interface Test, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	TerLoop	SatLoop	ModBer0	ModBer1	ModBer2	ModBer3	DemBer0	DemBer1
Byte 5	DemBer2	DemBer3	Spare	Spare	Spare	Spare	Spare	Spare
Byte 6	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 7	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 8	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 9	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 10	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 11	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 12	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 13	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

[TerLoop] = Interface Terrestrial Direction Data Loopback, 1b, 0=Disabled, 1=Enabled
[SatLoop] = Interface Satellite Direction Data Loopback, 1b, 0=Disabled, 1=Enabled
[ModBer3-ModBer0] = Interface Mod BER Test Pattern Generator, 4b, 0=Disabled, 1=2047, 2=2^23-1
[DemBer3-DemBer0] = Interface Demod BER Test Pattern Detector, 4b, 0=Disabled, 1=2047, 2=2^23-1

Interface SDMS, Command [C4h],

Interface SDMS, Read Change Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	IPAddr	IPMask	MAC	Options	Version	SerialN	Spare	Spare
Byte 1	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 2	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare
Byte 3	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface SDMS, Read Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	Failed	Reset	Alarm	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface SDMS, Read Bytes

Bytes 6-9	IP Address, 32b
Bytes 10-13	Network Mask, 32b
Bytes 14-19	MAC Address, 48b
Bytes 20-36	SDMS Options String Terminated with a 00h
Bytes 37-53	SDMS Version String Terminated with a 00h
Bytes 54-57	SDMS Serial Number, 32b
Bytes 58-65	Spare

Interface SDMS, Write Enable Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	IPAddr	IPMask	0	0	0	0	0	0
Byte 1	0	0	0	0	0	0	0	0
Byte 2	0	0	0	0	0	0	0	0
Byte 3	0	0	0	0	0	0	0	0

Interface SDMS, Write Flags

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 4	x	x	x	Spare	Spare	Spare	Spare	Spare
Byte 5	Spare	Spare	Spare	Spare	Spare	Spare	Spare	Spare

Interface SDMS, Write Bytes

Bytes 6-9	IP Address, 32b
Bytes 10-13	Network Mask, 32b
Bytes 14-19	x
Bytes 20-36	x
Bytes 37-53	x
Bytes 54-57	x
Bytes 58-65	Spare

Common Notes

All numbers are least significant byte first. All strings terminate with a **[00h]** byte and are 16 characters maximum (not including the string terminator).

The **Mode Byte** must be set to **[01h]** for control of the local modem and **[02h]** to control the remote modem (if local to remote internal control channel available).

If an error occurs, **Bit 7** of the **Status Byte** is set to one. Only one flag in the **Read Change Flags** section is set to one indicating the first write enabled option that generated an error (if any). The **Error/Warning Byte** has the type of error that occurred. If bit 7 of the **Status Byte** is not set the **Error/Warning Byte** is a warning (if any).

Status Byte (Returned in All Responses)

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Byte 0	Offline	AlarmA	AlarmB	InfAlm	UnitAlm	DemAlm	ModAlm	Error

Error Codes

[01h] Request Exceeded Available Limits, Request Aborted.
[02h] Request Exceeded Available Limits, Value Set to Limit.
[03h] Requested Option Not Available, Request Aborted.
[04h] Requested Function Read Only, Request Aborted.
[05h] Requested Frequency Exceeded Total Limits, Request Aborted.
[06h] Bad Request, Not Valid Option Number.
[07h] Option Not Installed, Request Aborted.
[08h] Flash ROM Write Error, Request Failed.
[09h] Write Access Denied Error, Request Aborted.
[0Ah] Requested Option Locked, Request Aborted.
[0Bh] Packet has Incorrect Number of Bytes for Selected Command.
[0Ch] Bad Command, Request Aborted.
[0Dh] Bad Unit Configuration for Selected Option
[0Eh] No Mcc Available, Request Aborted.
[0Fh] Mcc Send Buffer Full, Request Aborted.

Warning Codes

[40h*] Demod Fifo Buffer Exceeded Available Limits, Value Set to Limit.
[80h*] Requested Option Not Active Warning.
[01h] Cxr Enable Request Overridden by Cxr Alarm.
[02h] Bit Rate Changed to Available Limit.
[03h] Demod Not Locked Warning.
[04h] Clock Error, Mod & Demod Bit Rates Not Equal.
[05h] Demod in IF Loopback, Requested Cxr Frequency will be Active After IF Loopback Disabled.
[06h] AUPC Maximum Level Changed to Available Limit.
[07h] AUPC Minimum Level Changed to Available Limit.
[08h] Mod Cxr Level Changed to Available Limit.
[09h] No Remote AUPC Data Available Warning.
[0Ah] Reed-Solomon k Factor Changed to Available Limit.
[0Bh] Reed-Solomon n Factor Changed to Available Limit.
[0Ch] Custom IBS ESC Overhead Changed to Available Limit.
[0Dh] Custom IBS MCC Overhead Changed to Available Limit.
[0Eh] Demod Disabled Warning
[0Fh] Mod Disabled Warning
[10h] Cxr Frequency Changed to Available Limit.

*Can be logically ORed with other warning messages.

END OF PROTOCOL