XPA-200-OD

Outdoor SSPA
Installation and Operation Manual

Part Number MN/XPA200OD.IOM
Revision 1
February9,2005

IMPORTANT NOTE: The information contained in this document supersedes all previously published information regarding this product. Product specifications are subject to change without prior notice.

CUSTOMER SUPPORT

Contact the Comtech EF Data Customer Support Department for:

- Product support or training
- Information on upgrading or returning a product
- Reporting comments or suggestions concerning manuals

Contact Customer Support using any of the following methods:

Mail: Comtech EF Data
Customer Support Department
2114 West 7th Street
Tempe, Arizona 85281 USA

Email: service@comtechefdata.com

Internet: www.comtechefdata.com

Phone: 480.333.2200 (Main Comtech EF Data Number)
480.333.4357 (Customer Support Desk)

Fax: 480.333.2161

To return a Comtech EF Data product (in-warranty and out-of-warranty) for repair or replacement:

1. Request a Return Material Authorization (RMA) number from the Comtech EF Data Customer Support Department.

2. Be prepared to supply the Customer Support representative with the model number, serial number, and a description of the problem.

3. To ensure that the product is not damaged during shipping, pack the product in its original shipping carton/packaging.

4. Ship the product back to Comtech EF Data. (Shipping charges should be prepaid.)

For more information regarding the warranty policies, see Warranty Policy, p. v.

Table of Contents

ABOUT THIS MANUAL

This manual provides installation and operation information for the Comtech EF Data Outdoor SSPA. This is a technical document intended for earth station engineers, technicians, and operators responsible for the operation and maintenance of the Outdoor SSPA.

CONVENTIONS AND REFERENCES

CAUTIONS AND WARNINGS

Indicates information critical for proper equipment function.

Indicates a hazardous situation that, if not avoided, may result in minor or moderate injury. CAUTION may also be used to indicate other unsafe practices or risks of property damage.

Indicates a potentially hazardous situation that, if not avoided, could result in death or serious injury.

METRIC CONVERSION

Metric conversion information is located on the inside back cover of this manual. This information is provided to assist the operator in cross-referencing non-metric to metric conversions.

TRADEMARKS

All product names mentioned in this manual may be trademarks or registered trademarks of their respective companies and are hereby acknowledged.

REPORTING COMMENTS OR SUGGESTIONS CONCERNING THIS MANUAL

Comments and suggestions regarding the content and design of this manual will be appreciated. To submit comments, please contact the Comtech EF Data Technical Publications Department: tpubs@comtechefdata.com

Chapter 1. SYSTEM OPERATION

This section contains instructions for operating the XPA-200-OD outdoor SSPA. The primary customer interface to the XPA is via the Remote Communications port. This section defines in detail the customer interface.

1.1 CONNECTOR J3: AC POWER, MAINS, 205 TO 240 VAC.

Table 1. Connector J3 Pinout

Pin	Mating Connector Type
A	Ground
B	L2
C	L1

1.2 CONNECTOR J6: COMM, REMOTE COMMUNICATIONS PORT.

Mating connector: ITT: KPT06J14-19P or MS3116J14-19P connections, see Table 2.

Table 2. Connector J6 Pinout

Pin	Outdoor SSPA
A	RS485_+RX
B	RS485_-RX
C	RS485_+TX
D	RS485_-TX
E	RS232_RD
F	Analog_Pwr_Mon
G	RS232_TD
H	Aux_In
J	Aux_Out
K	SumFLT_COM
L	SumFLT_NO
M	SumFLT_NC
N	GND
P	ONLINE_Status
R	+24V
S	Mute Control
T	Minor_FLT_COM
U	Minor_FLT_NO
V	Minor_FLT_NC

1.3 TURNING ON THE SSPA

The SSPA does not contain a 'Power On/Off' switch. The SSPA is powered ON by connecting the J3 AC Power connector to the appropriate prime power source.

Never turn the unit ON without proper waveguide termination on the J2 "RF OUTPUT" port. Individuals can be exposed to dangerously high electromagnetic levels.

1.4 CONFIGURING THE SSPA

The following paragraphs provide a quick start guide to the most commonly used commands. Full coverage of all user remote commands is contained in Chapter 2.

1.4.1 ATTENUATION

The SSPA's attenuation is user configurable. The SSPA's attenuation can be selected between 00.00 and 25.00 dB in 0.25 dB increments.

Associated Remote Command(s): ATT=

1.4.2 MUTE

The transmitter output state (the 'RF OUT' port, J2) will be ON only if the following three requirements are met:

(1) The SSPA hardware must be fault free (i.e., No faults).
(2) The SSPA must be unmuted (i.e., MUT=0).
(3) The Amplifier must be ON (i.e., AMP=1).

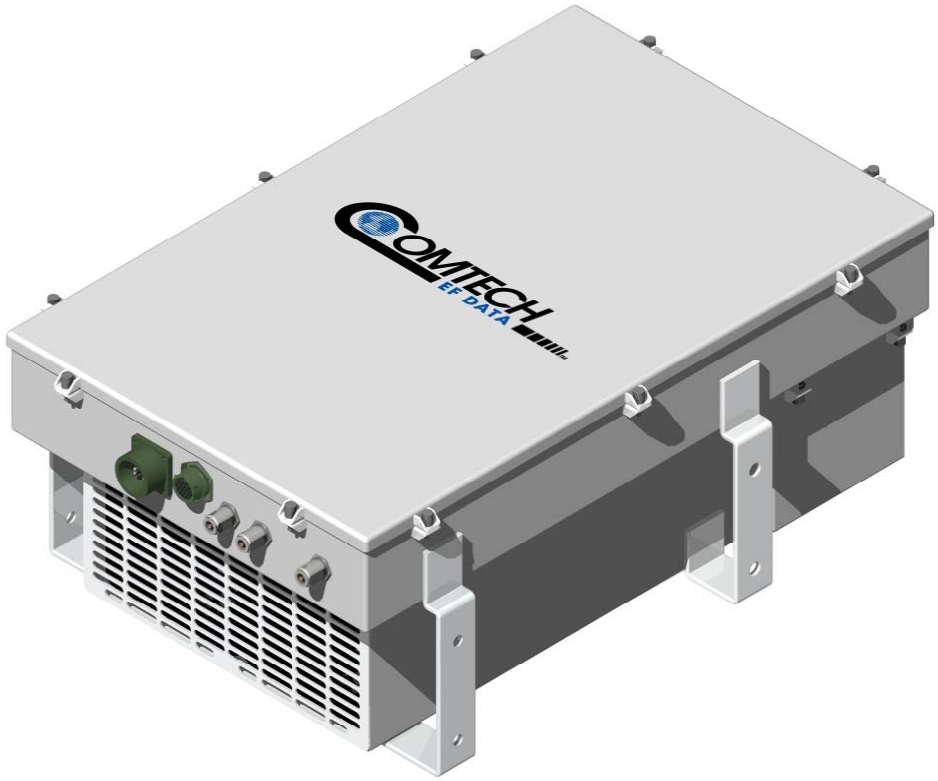
Associated Remote Command(s): MUT=

1.4.3 TX AMPLIFIER

The amplifier ON/OFF state is user controllable. Turning the TX amplifier OFF removes the 10Vdc supply to the RF Power FETs. The TX Amplifier must be ON in order for the SSPA to transmit a RF signal.

Associated Remote Command(s): AMP=

Figure 1. Outdoor Unit



NOTES:

Chapter 2. CUSTOMER COMMANDS

2.1 INTRODUCTION

This section describes the protocol and message command set for remote monitor and control of the SSPA product.

The electrical interface is either an RS-485 multi-drop bus (for the control of many devices) or an RS-232 connection (for the control of a single device), and data is transmitted in asynchronous serial form, using ASCII characters. Control and status information is transmitted in packets of variable length in accordance with the structure and protocol defined in later sections.

2.2 RS-485

For applications where multiple devices are to be monitored and controlled, a full-duplex (4-wire) RS-485 is preferred. Half-duplex (2-wire) RS-485 is possible, but is not preferred.

In full-duplex RS-485 communication there are two separate, isolated, independent, differential-mode twisted pairs, each handling serial data in different directions. It is assumed that there is a 'controller' device (a PC or dumb terminal), which transmits data, in a broadcast mode, via one of the pairs. Many 'target' devices are connected to this pair which all simultaneously receive data from the controller. The controller is the only device with a line-driver connected to this pair; the target devices only have line-receivers connected.

In the other direction, on the other pair, each target has a tri-stateable line driver connected, and the controller has a line-receiver connected. All the line drivers are held in high-impedance mode until one (and only one) target transmits back to the controller.

Each target has a unique address, and each time the controller transmits, in a framed 'packet' of data, the address of the intended recipient target is included. All of the targets receive the packet, but only one (the intended) will reply. The target enables its output line driver, and transmits its return data packet back to the controller in the other direction on the physically separate pair.

2.3 RS-485 (FULL DUPLEX) SUMMARY:

- Two differential pairs - one pair for controller to target, one pair for target to controller.
- Controller-to-target pair has one line driver (controller), and all targets have line-receivers.
- Target-to-controller pair has one line receiver (controller), and all targets have tri-state drivers.

2.4 RS-232

This is a much simpler configuration in which the controller device is connected directly to the target via a two-wire-plus-ground connection. Controller-to-target data is carried, via RS-232 electrical levels on one conductor, and target-to-controller data is carried in the other direction on the other conductor.

2.5 BASIC PROTOCOL

Whether in RS-232 or RS-485 mode, all data is transmitted as asynchronous serial characters, suitable for transmission and reception by a UART. The asynchronous character format is fixed at 8 data bits, no parity, and 1 stop bit. Only two (2) baud rates are supported: 9600 baud and 19200 baud.

All data is transmitted in framed packets. The host controller is assumed to be a PC or ASCII dumb terminal, which is in charge of the process of monitor and control. The controller is the only device that is permitted to initiate, at will, the transmission of data. Targets are only permitted to transmit when they have been specifically instructed to do so by the controller.

All bytes within a packet are printable ASCII characters, less than ASCII code 127. In this context, the Carriage Return and Line Feed characters are considered printable.

All messages from controller to target require a response (with one exception). This will be either to return data that has been requested by the controller, or to acknowledge reception of an instruction to change the configuration of the target. The exception to this is when the controller broadcasts a message (such as Set time/date) using Address 0, when the target is set to RS-485 mode.

2.6 PACKET STRUCTURE

Controller-to-target:

Start of Packet	Target Address	Address De-limiter	Instruction Code	Code Qualifier	Optional Arguments	End of Packet
< ASCII code 60 (1 character)	(4 characters)	/ ASCII code 47 (1 character)	(3 characters)	= or ? ASCII code 61 or 63 (1 character)	(n characters)	Carriage Return ASCII code 13 (1 character)

Example: <0412/MUT=1{CR}

Target-to-controller:

Start of Packet	Target Address	Address De-limiter	Instruction Code	Code Qualifier	Optional Arguments	End of Packet
> ASCII code 62 (1 character)	(4 characters)	/ ASCII code 47 (1 character)	(3 characters)	=, ?, !, or * ASCII code 61, 63, 33 or 42 (1 character)	(From 0 to n characters)	Carriage Return, Line Feed ASCII code 13,10 (2 characters)

Example: >0412/MUT=1{CR}{LF}

Each of the components of the packet is now explained.

2.6.1 START OF PACKET

- Controller to Target: This is the character '<' (ASCII code 60)
- Target to Controller: This is the character '>' (ASCII code 62)

Because this is used to provide a reliable indication of the start of packet, these two characters may not appear anywhere else within the body of the message.

2.6.2 ADDRESS

Up to 9,999 devices can be uniquely addressed. In both RS-232 and RS-485 applications, the permissible range of values is 1 to 9999. It is programmed into a target unit using the remote control port.

The controller sends a packet with the address of a target - the destination of the packet. When the target responds, the address used is the same address, to indicate to the controller the source of the packet. The controller does not have its own address.

2.6.3 INSTRUCTION CODE

This is a three-character alphabetic sequence that identifies the subject of the message. Wherever possible, the instruction codes have been chosen to have some significance. This aids in the readability of the message, should it be displayed in its raw ASCII form. Upper case and lower case alphabetic characters may be used (A-Z, and a-z).

2.6.4 INSTRUCTION CODE QUALIFIER

This is a single character that further qualifies the preceding instruction code.

Code Qualifiers obey the following rules:

1. From Controller to Target, the only permitted values are:

   = (ASCII code 61)
   ? (ASCII code 63)

They have these meanings:

The '=' code (controller to target) is used as the assignment operator, and is used to indicate that the parameter defined by the preceding byte should be set to the value of the argument(s) which follow it.

For example, in a message from controller to target, MUT=1 would mean 'enable the mute function'.

The '?' code (controller to target) is used as the query operator, and is used to indicate that the target should return the current value of the parameter defined by the preceding byte.

For example, in a message from controller to target, MUT? would mean 'return the current state of the mute function'.

2. From Target to Controller, the only permitted values are:

   = (ASCII code 61)
   ? (ASCII code 63)
   ! (ASCII code 33)
   * (ASCII code 42)
   # (ASCII code 35)

They have these meanings:

The '=' code (target to controller) is used in two ways:

First, if the controller has sent a query code to a target (for example MUT?, meaning ‘is mute enabled or disabled?’), the target would respond with MUT=x, where x represents the state in question, 1 being ‘enable’ and 0 being disable.

Second, if the controller sends an instruction to set a parameter to a particular value, and, providing the value sent in the argument is valid, then the target will acknowledge the message by replying with MUT= (with no message arguments).

The ‘?’ code (target to controller) is only used as follows:

If the controller sends an instruction to set a parameter to a particular value, and, if the value sent in the argument is not valid, then the target will acknowledge the message by replying (for example) with MUT? (with no message arguments). This indicates that there was an error in the message sent by the controller.

The ‘*’ code (target to controller) is only used as follows:

If the controller sends an instruction to set a parameter to a particular value, and, if the value sent in the argument is valid, however the target is in the wrong mode (e.g., standby mode in redundancy configuration) that it will not permit that particular parameter to be changed at that time, then the target will acknowledge the message by replying (for example) with MUT* (with no message arguments).

The ‘!’ code (target to controller) is only used as follows:

If the controller sends an instruction code which the target does not recognize, then the target will acknowledge the message by echoing the invalid instruction, followed by the ! character with. Example: XYZ!

The ‘#’ code (target to controller) is only used as follows:

If the controller sends an instruction code which the target cannot currently perform because of hardware resource issues, then the target will acknowledge the message by echoing the invalid instruction, followed by the # character. This response can only occur if the operator sends two or more ‘hardware configuration’ type commands without allowing adequate time between commands for the hardware to be configured. For example, if the operator issued commands to change both the frequency and the attenuation with less than 100 milliseconds between commands, and if this response is returned, then the command has not been excepted and the operator must resend the command.

2.6.5 MESSAGE ARGUMENTS

Arguments are not required for all messages. Arguments are ASCII codes for the characters 0 to 9 (ASCII 48 to 57), period (ASCII 46) and comma (ASCII 44).

2.6.6 END OF PACKET

Controller to Target: This is the ‘Carriage Return’ character (ASCII code 13)

Target to Controller: This is the two-character sequence ‘Carriage Return’, ‘Line Feed’. (ASCII code 13, and code 10.)

Both indicate the valid termination of a packet.

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Attenuation	ATT=	5 bytes, numerical	Command or Query. Valid attenuation level, in dB, at 0.25-dB step size as factory default. Example: ATT=12.25	ATT= (message ok) ATT? (received ok, but invalid arguments found) ATT* (message ok, but not permitted in current mode)	ATT?	ATT=xx.xx (same format as command arguments)
RF Power Amplifier State	AMP=	1 byte, value of 0, 1	Command or Query Turns ON or OFF the RF power amplifiers. 0 = Off 1 = On	AMP= (message ok) AMP? (received ok, but invalid arguments found) AMP* (message ok, but not permitted in current mode)	AMP?	AMP=x (same format as command arguments)
Mute State	MUT=	1 byte, value of 0,1	Command or Query. Mute the unit, where: 0 = Disabled 1 = Enabled Example: MUT=1	MUT= (message ok) MUT? (received ok, but invalid arguments found) MUT* (message ok, but not permitted in current mode)	MUT?	MUT=x (same format as command arguments)
Online Status	N/A	1 byte, value of 0,1	Command or Query. Online status (applies only to redundancy), where: 0 = Disabled 1 = Enabled Example: ONL=1	ONL= (message ok) ONL? (received ok, but invalid arguments found)	ONL?	ONL=x (same format as command arguments)
Redundancy Sate	RED=	1 byte, value of 0, 1	Command or Query Turns ON or OFF the redundancy state. 0 = OFF, 1 = ON	RED= (message ok) RED? (received ok, but invalid arguments found)	RED?	RED=x (same format as command arguments)
Redundancy Mode: Auto or Manual	RAM=	1 byte, value of 0, 1	Command or Query. Sets Auto or Manual mode for redundancy. 0 = Manual, 1 = Auto	RAM= (message ok) RAM? (received ok, but invalid arguments found)	RAM?	RAM=x (same format as command arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Force Back-Up State	FBU=	1 byte, value of 0, 1, 2	Command or Query Force one of the online units to be a back-up for maintenance and test purposes. 0 = Removed force back-up situation 1 = SSPA #1 is forced to be a back-up 2 = SSPA #2 is forced to be a back-up	FBU= (message ok) FBU? (received ok, but invalid arguments found)	FBU?	FBU=x (same format as command arguments)
Remote Address	SPA=	4 bytes, numeric	Command or Query. Set Physical Address-between 0001 to 9999. Resolution 0001 Example: SPA=0412	SPA= (message ok) SPA? (received ok, but invalid arguments found)	SPA?	SPA=x (same format as command arguments)
Remote Baud Rate	SBR=	4 bytes	Command or Query. Set remote baud rate as follows: 9600 = 9600 baud 19K2 = 19200 baud	SBR= (message ok) SBR? (received ok, but invalid arguments found)	SBR?	SBR=xx (same format as command arguments)
Set RTC(Real-Time-Clock) Date	DAT=	6 bytes, numeric	Command or Query. A command in the form **mmddyy**, where; dd = day of the month, between 01 and 31, mm = month of the year, between 01 and 12 and yy = year, between 00 and 96 (2000 to 2096) Example: DAT=042503 would be April 24, 2003	DAT= (message ok) DAT? (received ok, but invalid arguments found) DAT* (message ok, but not permitted in current mode)	DAT?	DAT=xx (same format as command arguments)
Set RTC Time	TIM=	6 bytes, numeric	Command or Query. A command in the form **hhmmss**, indicating the time from midnight, where hh = hours, between 00 and 23; mm = minutes, between 00 and 59, and ss = seconds, between 00 and 59 Example: TIM=231259 would be 23 hours, 12 minutes and 59 seconds from midnight.	TIM = (message ok) TIM? (received ok, but invalid arguments found) TIM * (message ok, but not permitted in current mode)	TIM?	TIM=xx (same format as command arguments)
Clear All Stored Alarms	CAA=	None	Command only Instructs the slave to clear all Stored Events This command takes no arguments.	CAA= (message ok)	N/A	N/A
Serial Number	N/A	9 bytes, numeric 000000000 to 999999999	Query only. Used to Query the units 9 digit serial number. Slave returns its S/N, in the form **xxxxxxxxx**. Example: RSN=000000165	N/A	RSN?	RSN=xxxxxxxxx (see description for details of arguments)
Retrieve Equipment Type	N/A	22 bytes, alpha-numeric	Query only. The unit returns a string indicating the Model Number and the the value of internal software revision installed Example: RET=CPA-300 VER: 1.0.3	N/A	RET?	RET=x….x (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Auto Fault Recovery	AFR=	1 byte, value of 0, 1	Command or Query. The SSPA output will automatically be muted in the event of detected fault. If auto fault recovery is enabled, it will cause the output to go active (unmute) if all faults are cleared. If disabled, the output will remain muted even if all faults are cleared.	AFR = (message ok) AFR? (received ok, but invalid arguments found) AFR* (message ok, but not permitted in current mode)	AFR?	AFR=x (same format as command arguments)
Retrieve next 5 unread Stored Alarms	N/A	145 bytes	Query only The unit returns the oldest 5 Stored Events which have not yet been read over the remote control. Reply format: **Sub-body{CR}Sub-body{CR}Sub-body{CR}Sub-body{CR}Sub-body, where Sub-body=** **YYYYYYYYYY ZZ mmddyy hhmmss,** YYYYYYYYYY being the fault description. ZZ being the alarm type. FT = Fault OK = Clear IF = Information If there are no new events, the unit will reply with LNA*	N/A	LNA?	LNA=**YY..ss** (see description for details of arguments)
Retrieve Number of unread Stored Alarms	N/A	2 bytes, numeric	Query only. Returns the number of Stored Events which remain unread, in the form **xx**. Example reply: TNA=18	N/A	TNA?	TNA=**xx** (see description for details of arguments)
Summary Fault Status	N/A	1 byte, value of 0,1	Query only. Used to Query the status of the Summary Fault Relay. Example: SFS=0 where: 0 = OK 1 = FT	N/A	SFS?	SFS=**x** (see description for details of arguments)
Terminal Status change	N/A	1 byte, value of 0,1	Query only. Used to Query the status of the Terminal Status. Example: TSC=0 Where: 0 = no change in status, 1 = change in status	N/A	TSC?	TSC=**x** (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Circuit Identification	CID=	24 bytes, alpha-numeric	Command or Query Used to identify or name the unit or station. First line is limited to 12 characters. Second line is also limited to 12 characters. No carriage return between first line and second line. Example: CID='cr' --Earth Station 1— ---SSPA #1----	CID= (message ok) CID? (received ok, but invalid arguments found)	CID?	CID=x…x (see description for details of arguments)
Retreive Firmware Number	N/A		Query only Gets the Firmware Number of the unit. Example: <1/FRW= >0001/FRW/10786'cr' BULK=FW/10786'cr' MnC=FW/10787'cr' FPGA=FW/10788'cr''lf'	N/A	FRW?	FRW=FWxxxxx
Retrieve Maintenance Status	N/A	168 bytes, alpha- numeric	Query only. Used to Query the maintenance status of the unit P24VT=024.1'cr' P15VT=015.2'cr' P10V1=010.4'cr' P10V2=010.4'cr' P7V5T=007.8'cr' P5VLT=005.8'cr' N5VLT=-05.7'cr' FANR1=100.0'cr' FANR2=100.0'cr' ATEMP=+40.0'cr' A10V1=010.2'cr' A10V2=010.2'cr' FWPWR=+37.6'cr' RVPWR=02.0'cr''lf' *Note: REFV will appear if REF OSC module is installed.	N/A	RMS?	RMS=***x….x*** (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Concise Maintenance Status	N/A	84 bytes numeric	Query only. Used to Query the Maintenance status of the unit in concise format. Response is comma delimited. Example: CMS=aaa.a,bbb.b,ccc.c,ddd.d,eee.e, fff.f,ggg.g,hhh.h,iii.i,jjj.j,kkk.k,lll.l,mmm.m,nnn.n, 'cr''lf' where: aaa.a = +24V Power Supply bbb.b = +15V Power Supply ccc.c = +10V-1 Power Supply ddd.d = +10V-2 Power Supply eee.e = +7.5V Power Supply fff.f = +5V Power Supply ggg.g = -5V Power Supply hhh.h = Fan #1 speed (in percent) iii.i = Fan #2 speed (in percent) jjj.j = Amplifier temperature in deg. C kkk.k = Ampifier 10V1 lll.l = Amplifier 10V2 mmm.m=Forward RF output power, in dBm Note: nnn.n will appear for Ref Voltage if Reference Osc Module is installed.	N/A	CMS?	CMS=x….x (see description for details of arguments)
Retrieve Utility Status	N/A	27 bytes, alpha-numeric	Query only. Used to Query the utility status of the unit Example: RUS='cr' ADR=0001'cr' BDR=9600'cr'	N/A	RUS?	RUS=x….x (see description for details of arguments)
Concise Utility Status	N/A	11 bytes, alpha-numeric	Query only. Used to Query the Maintenance status of the unit, response is comma delimited. Example: CUS=aaaa,bbbb,ccc,'cr''lf' where: aaaa = Remote Unit Address bbbb = Remote Baud Rate	N/A	CUS?	CUS=x….x (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Retrieve Alarm Status	N/A	117 bytes, alpha-numeric	Query only. Used to Query the Alarm status of the unit Example: RAS=’cr’ P24VT=OK’cr’ P15VT=OK’cr’ P10V1=OK’cr’ P10V2=OK’cr’ P7V5T=OK’cr’ P5VLT=OK’cr’ N5VLT=OK’cr’ FAN#1=OK’cr’ FAN#2=OK’cr’ HSTMP=OK’cr’ SHTDN=OK’cr’ IICST=OK’cr’ FPOUT=OK’cr’’IF’	N/A	RAS?	RAS=x….x (see description for details of arguments)
Concise Alarm Status	N/A	25 bytes, numeric with commas	Query only. Used to Query the Alarm status of the unit, response is comma delimited. Example: CMS=a,b,c,d,e,f,g,h,I,j,k,l,m’cr’’lf’ where: a thru k = 0 or 1, 0 = OK 1 = FT a = +24V Power Supply b = +15V Power Supply c = +10V-A Power Supply d = +10V-B Power Supply e = +7.5V Power Supply f = +5V Power Supply g = -5V Power Supply h = Fan#1 State i = Fan#2 State j = Heatsink Temp k = Shutdown l = llC Status m=Forward Power Alarm	N/A	CAS?	CAS=x….x (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
Retrieve Configuration Status	N/A	45 bytes, alpha-numeric	Query only. Used to Query the configuration status of the unit Example: RCS='cr' ATT=12.75'cr' AMP=1'cr' MUT=1'cr' ONL=1'cr' RED=1-1'cr' GOF=00.00'cr' AFR=1'cr''lf' where: ATT= attenuation in dB AMP= RF power amplifier state, 0=OFF, 1=ON MUT=RF mute state, 0=unmuted, 1=muted ONL=Online status for redundancy RED=Redundancy state and mode, states: 0=OFF, 1=ON, modes: 0 = auto, 1 = manual GOF=Gain Offset in dB AFR= auto fault recovery, 0=manual, 1=auto	N/A	RCS?	RCS=x….x (see description for details of arguments)
Concise Configuration Status	N/A	24 bytes, numeric	Query only. Used to query the summarized version of RCS. Example: CCS=aaaaa,b,c,d,e-e,fffff,g,'cr' Where: aaaaa = attenuation in dB b = RF power amplifier state c = mute state, 0 = unmuted, 1 = muted d = online status e-e = redundancy state and mode fffff = gain offset in dB g = AFR	N/A	CCS?	CCS=x….x (see description for details of arguments)

	Command (Instruction Code and qualifier)	Arguments for Command or Response to Query	Description of arguments (note that all arguments are ASCII numeric codes, that is, ASCII codes between 48 and 57)	Response to Command (target to controller)	Query (Instruction Code and qualifier)	Response to query (target to controller)
RF Power FET Current status	N/A	variable length (Note: the number of FETs is determined on the information string [INF])	Query only Used to display all the FET currents. Example: RFS? Q01=xx'cr' Q02=xx'cr' Q03=xx.x'cr' Q04=xx.x'cr' Q05=xx.x'cr' Q06=xx.x'cr' Q07=xx.x'cr' Q08=xx.x'cr' Q09=xx.x'cr' Q10=xx.x'cr' Q11=xx.x'cr' Q12=xx.x'cr' Q13=xx.x'cr' Q14=xx.x'cr' Q15=xx.x'cr' Q16=xx.x'cr''lf'	N/A	RFS?	RFS=x….x (see description of arguments)
Concise RF Power FET Current Status	N/A	variable length	Query only Concise version of RFS. Example: CFS=xxx,xxx,x.x,x.x,……..,x.x,	N/A	CFS?	CFS=x…..x (see description of RFS. Note that each argument is separated by a comma)

Chapter 3. POWER SUPPLY REPLACEMENT

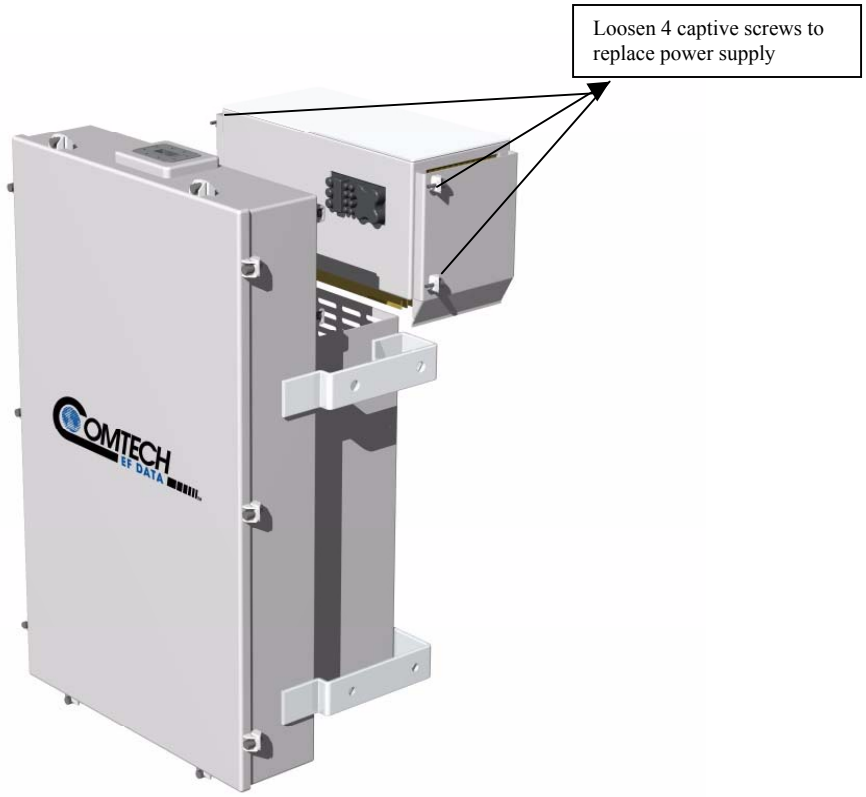


Figure 2. Power Supply Replacement

NOTES:

METRIC CONVERSIONS

Units of Length

Unit	Centimeter	Inch	Foot	Yard	Mile	Meter	Kilometer	Millimeter
1 centimeter	—	0.3937	0.03281	0.01094	6.214×10^{-6}	0.01	—	—
1 inch	2.540	—	0.08333	0.2778	1.578×10^{-5}	0.254	—	25.4
1 foot	30.480	12.0	—	0.3333	1.893×10^{-4}	0.3048	—	—
1 yard	91.44	36.0	3.0	—	5.679×10^{-4}	0.9144	—	—
1 meter	100.0	39.37	3.281	1.094	6.214×10^{-4}	—	—	—
1 mile	1.609×10^{5}	6.336×10^{4}	5.280×10^{3}	1.760×10^{3}	—	1.609×10^{3}	1.609	—
1 mm	—	0.03937	—	—	—	—	—	—
1 kilometer	—	—	—	—	0.621	—	—	—

Temperature Conversions

Unit	° Fahrenheit	° Centigrade
32° Fahrenheit	—	0 (water freezes)
212° Fahrenheit	—	100 (water boils)
-459.6° Fahrenheit	—	273.1 (absolute 0)

Formulas
C = (F - 32) * 0.555
F = (C * 1.8) + 32

Units of Weight

Unit	Gram	Ounce Avoirdupois	Ounce Troy	Pound Avoir.	Pound Troy	Kilogram
1 gram	—	0.03527	0.03215	0.002205	0.002679	0.001
1 oz. avoir.	28.35	—	0.9115	0.0625	0.07595	0.02835
1 oz. troy	31.10	1.097	—	0.06857	0.08333	0.03110
1 lb. avoir.	453.6	16.0	14.58	—	1.215	0.4536
1 lb. Troy	373.2	13.17	12.0	0.8229	—	0.3732
1 kilogram	1.0×10^{3}	35.27	32.15	2.205	2.679	—

2114 WEST 7TH STREET TEMPE ARIZONA 85281 USA
480 • 333 • 2200 PHONE
480 • 333 • 2161 FAX