Xeos Technologies Inc.

XMR User Manual

Waterproof Iridium Data Transceiver

Version 1.1

February 2021

Shipped From

Contact Us

Email	support@xeostech.com
Phone	+1 (902) 444-7650
Fax	+1 (902) 444-7651
Website	www.xeostech.com

Specifics

This manual version is written with respect to XMR 6790. To acquire the latest firmware for your device, contact support@xeostech.com

Version History

Version No.	Date	Description
1.0	Jan 2021	Base document
1.1	Feb 2021	Added Subconn connector options

Regular checks for the latest manual are suggested. Be sure to check Xeos Technologies' manuals page to compare versions and download the latest version.

Table of Contents

General Description

The XMR is a bidirectional data relay device, utilized for peripheral data transfer as well as location reporting via the Iridium satellite network. XMR's small size and mountable enclosure make it easy to install quickly in transmission-friendly locations.

Inside the XMR is a 9603 Iridium satellite Short Burst Data core radio transceiver, a specialized low-power Xeos digital controller with GPS, Iridium & GPS antennas.

While the XMR is intended for locally connected surface deployments. Xeos Technologies Inc. (Xeos) manufactures other products for fixed location monitoring and remote tracking and surveillance.

See www.xeostech.com for details or call (902)-444-7650.

Theory of Operation

The XMR is intended for continuous and/or periodic monitoring of high value assets globally. The XMR can facilitate instant and accurate location of assets for recovery or monitoring purposes.

The internal GPS determines location which is stored on board the XMR. An Iridium modem provides two-way communications for retrieval of data and sending commands to the remote tracker. The deployment is facilitated by the miniature device which has both GPS and Iridium antennas on board.

Operators can communicate with the XMR via is integrated RS-232 port locally, as well as over Iridium using XeosOnline and/or email command. Status information can be obtained, including the health of the GPS system and battery voltage. Timings are configurable and can be modified as the nature of the tracking or monitoring operation changes.

The internal relay function allows the XMR to be piggy-backed to an external serial device and used as a communications interface for that device. For instance, you may wish to send commands to a remote camera or receive data from a temperature sensor.

While any email application can be used to send and receive messages between the user and the XMR, XeosOnline is the easiest way to manage and monitor multiple XMR devices. XeosOnline presents the information in a readable format and displays it on street level maps.

Preliminary Setup

Setting up your Iridium Account

XMR makes use of the Iridium satellite system's Short Burst Data (SBD) service for the 9603 transceiver. This service is a global, two-way, real-time, email-based data delivery service that has a maximum outbound (from XMR) message size of 340 bytes and a maximum inbound (to XMR) message size of 270 bytes.

XMR end users must set up an approved data delivery account with their preferred service provider. Xeos is an Iridium VAR and can provide Iridium service if you wish.

Setting up service requires the International Mobile Equipment Identity (IMEI) number. Each 9603 has a unique IMEI number that must be registered with a preferred service provider. Xeos will make these numbers available when the product is delivered.

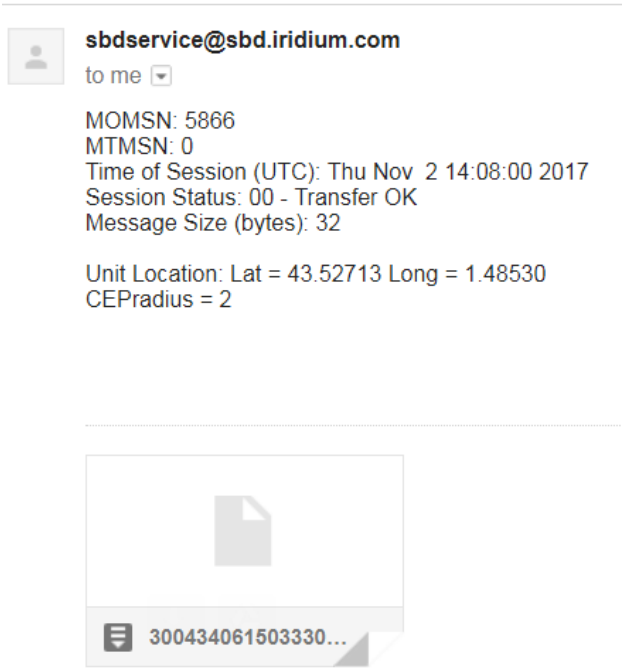
Each IMEI number is capable of being associated with up to five (5) unique destinations (emails, IP addresses, other Iridium IMEIs) which may vary between service providers. When registering the IMEI, please provide the service provider with the temporary Xeos testing account email address. You may contact Xeos for this email. Setting up this email address allows for better technical support during the initial learning period for the product.

Understanding Position Information

There are two types of position information which will be sent via the Iridium Gateway.

Iridium Doppler position

The Iridium Gateway calculates the Iridium transceivers' position on earth when it receives a transmission, using Doppler technology. As a result, it is often very inaccurate. This location is only visible to users getting emails directly from the device, as opposed to emails forwarded by XeosOnline. An example of a "raw" Iridium message via email is below and will always have the IMEI of the device in the subject line, regardless of its name on XeosOnline.



sbdservice@sbd.iridium.com	All messages from Iridium devices come from this address.
MOMSN: 5866	Mobile Originating Message Serial Number; each individual message has its own incrementing serial number. A mobile originating message is one that comes **from** the Iridium device.
MTMSN: 0	Mobile Terminating Message Serial Number; like the MOMSN, messages **to** Iridium devices (like commands) also have an incrementing serial number. Since the message in the example is from an Iridium device, the MT number is zero.
Time of Session (UTC)	The time the message arrived at the Iridium Gateway
Session Status	Each message will have a code determined by how well the message was received; codes 00, 01 and 02 are acceptable and will always have their code name (ex. Transfer OK) next to this number.
Message Size (bytes)	The size of the actual message sent by the Iridium device, which is in the attachment in the email.
*Unit Location	The Doppler position of the device as estimated by Iridium's network. It is **NOT** the GPS position measured by the device.
*CEPradius = 2	The numerical value of how accurate the above position is; with a value of 2, This means that using the Lat/Long that the body has supplied, Iridium is 80% confident (always 80%) that the device sending the message is within a circle, 2 kilometers in radius, with the Lat/Long given as the center of that circle. The higher the CEPradius value, the larger the circle and therefore the less accurate that position.

*These items can be enabled/disabled by your Iridium provider if desired.

Global Positioning System

Location information generated by the device itself is embedded in the SBD attachment sent via the Iridium Gateway and can only be seen through the XeosOnline system or situations where the position information is sent in a plain-text format (XeosOnline message forwarder or using the **$msgenable** command). This position information is accurate to within several feet of the true position.

```
Timestamp: 2018-04-11T19:20:10.001Z, BatteryV: 10.38, Latitude: 44.714227, Longitude: -63.604954, Vul: 11.88
```

XMR Model Variants

Chogori

For applications that have less opportunity for deeper submergence, the Chogori variant has a one inch thread on the bottom of the enclosure for pole mounting.

Subconn

For applications that will experience prolonged and moderate submergence, the Subconn variant can come in a 4-Pin or 6-Pin configuration. This variant is mounted by way of four #8-32 screws to a mounting plate.

XMR Operation

Powering the XMR

The XMR does not have an internal power source, therefore providing that supply immediately turns the unit on. While the power source can be set within operating range, it is recommended that users not use a power source very close to either extreme. Once the power supply dips below 7V, communication may not be possible with the XMR.

If the power supply is subject to power surges, it may be advisable to use a power supply with an average voltage of 24 VDC to avoid damaging the internal circuitry of the XMR. Refer to the voltage ranges for the corresponding appropriate ranges.

LED Indicators

The XMR's enclosure allows visibility of its on/off status using an LED. It is to be used as an indicator for when the device is initially powered. This same LED also pulses red in the event of a self-test failure at start-up.

Behaviours	
Flashing 4 Hz **on power-up**	Device powering up, performing self-test
Flashing 0.5 Hz **after power-up**	Self-test failure

Deployment of the XMR

The XMR contains GPS and Iridium antennas. Deploying the XMR requires the following:

- An XMR with Iridium service activated (using IMEI provided by Xeos)
- An external power source from 7 – 32 VDC capable of supplying max 1.5A
- An external RS-232 serial device to communicate to the XMR
- A suitable deployment environment
    - The top of the XMR must have as good a view of the full sky as possible.
        - Alongside a wall or tall obstruction will cause coverage to suffer.
        - The antenna surface must not be obstructed by metal or wood. Thick amounts of other materials may also cause interference.
    - XMR should not be deployed near (less than 1 meter) other transmitters.

Communicating with the XMR

Communicating via RS-232

Communication with the XMR must be completed using RS-232, but only require Transmit, Receive and a common Ground (no RTS/CTS).

Pre-programming of the XMR is not required since it is pre-configured for most tracking scenarios and configurable settings can be changed over the Iridium link. Confirmation of settings can be done via an XMR diagnostic cable through a terminal emulator using these settings:

Baud Rate	57.6k
Parity	None
Data Bits	8
Stop Bits	1

- All commands to the XMR must be prepended with the dollar sign ($).
- All commands must be appended with a command line, line feed, or both.

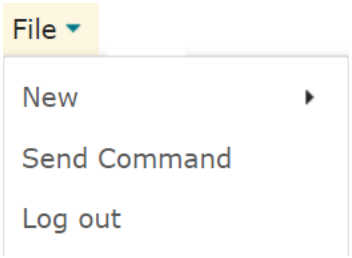
Sending Commands Using XeosOnline

Before using XeosOnline make sure that your account has been set up and your device added to your organization. Contact activations@xeostech.com for more information.

Setting up to Send

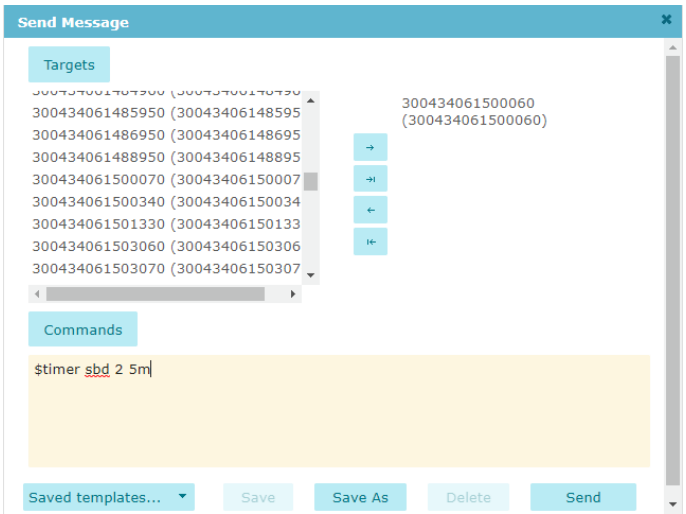
Navigate to the Send Command window.

From the Home Tab, choose **File > Send Command**



Select the units you wish to target with commands and move them over to the right-hand target list using the **->** button.
Type your command(s) into the command box and press **Send**.

Remember to include the dollar sign (**$**) ahead of each command and enter each command on a separate line.



Outgoing messages will appear in the Message Log for the commanded device.

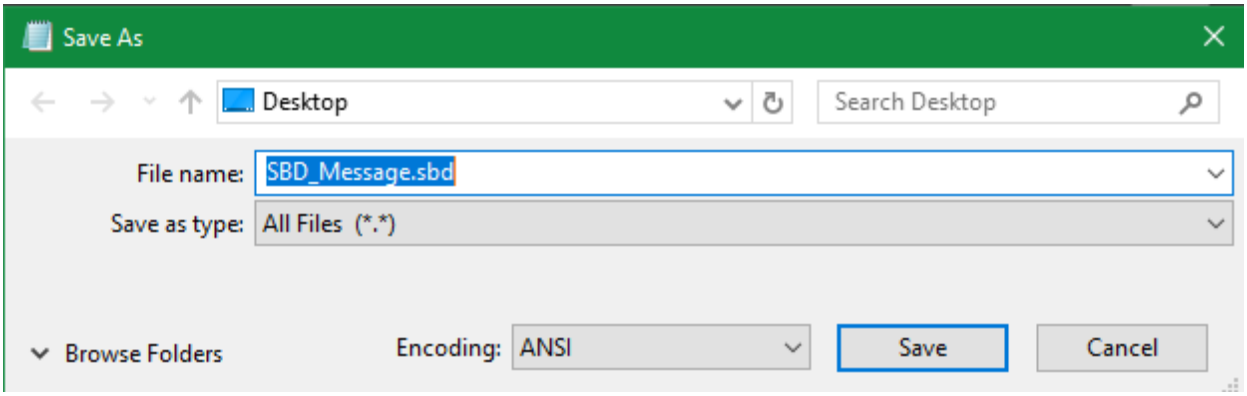
Sending Commands via Email

To receive commands from the Iridium network, the device in use must have a clear view of the sky. If the device is unable to communicate with the Iridium network, commands will remain queued for five days.

Command Format

Creating the File

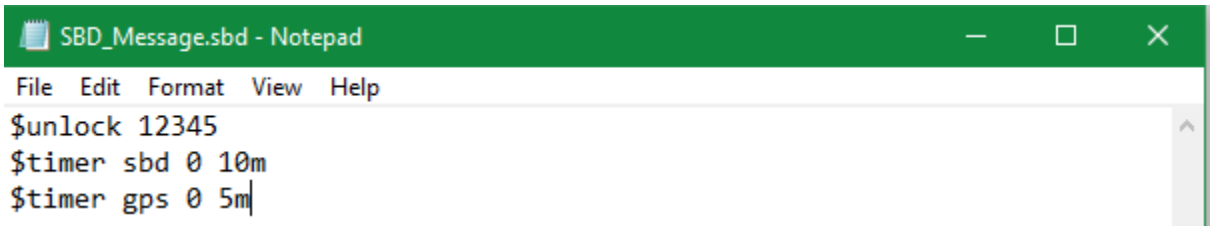
To create an SBD command, open a new file in a text editor (ex. Notepad) and save it using the **.sbd** extension. Make sure the **Save as type** option is set to **All Files** to achieve this.



Command Structure

Commands must be structured in the following way:

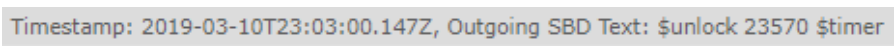
1. Each command **MUST** have a dollar sign (**$**) before each command.
2. The unit's unlock code in the following format: **$unlock XXXXX** where **XXXXX** is the unit's five digit unlock code.
3. A list of commands, one command per line.



The Unlock Code

SBD commands without an unlock code will be ignored by devices that require it. The unlock code is generated by the device itself and can be retrieved from XeosOnline, or the unit can be queried for its unlock code by sending **$unlock** to the device as a command. The device will respond by sharing the 5-digit code with its provisioned destinations.

Navigate to your unit and click on the **Message Log** Tab. The unlock code can be seen in the most recent outgoing messages.



Once XeosOnline has a device's unlock code, all outgoing commands will have the code added at the beginning of the message. The unlock code will only be changed if the firmware is updated, or if commanded via:

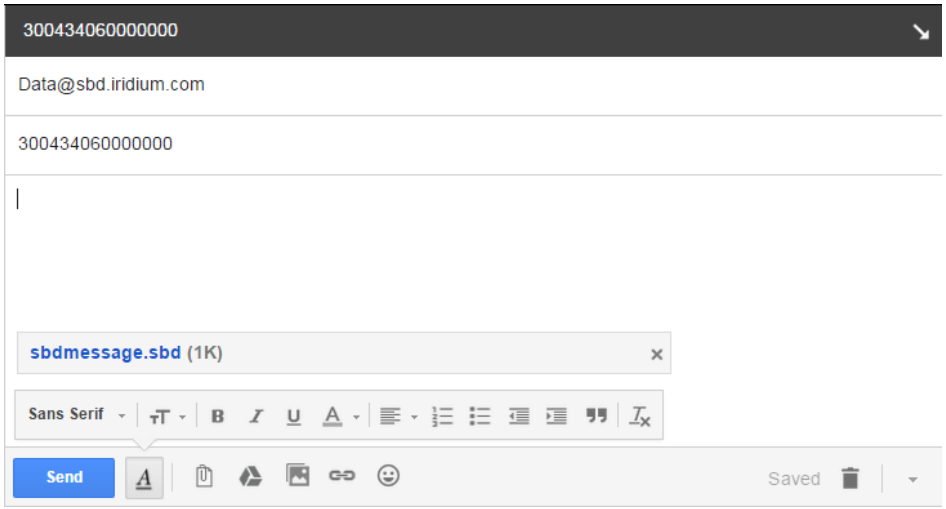
$unlock XXXXX YYYYY

Where **XXXXX** is the current code, and **YYYYY** is the new code to be used in the future.

Sending the Command

To send an sbd command, create a new email message with the following fields:

To	data@sbd.iridium.com
Subject	The target device's IMEI
Body	Empty
Attachments	The .sbd file



A confirmation will be immediately returned from the Iridium Gateway from the address **sbdservice@sbd.iridium.com** indicating that your message is now in the message queue. It will be delivered to the device during its next Iridium check.

Commands can be sent from any email address, but responses will be returned **only** to email addresses on the unit's forwarding list.

Messages from the XMR

There are several messages sent from the XMR, depending on setup and conditions. Below are examples of the most common Mobile Originating messages.

Settings Summary Message

On startup, the XMR will send a summary of timers and a short list of GPS and Iridium settings to confirm to the user the immediate configuration of inactive (Timer Zero) and In-Motion (Timer Two) timers.

```
Flags: Unsolicited, IRD Normal Timing: 3600sec, IRD Alarm Timing: 60sec, GPS Normal Timing: 3600sec,
GPS Alarm Timing: 20sec, GPS Retry Qty: 2, GPS Retry Delay: 20min, GPS Max On Normal: 100sec, GPS Max On Poor: 30sec,
GPS Max Backlog: 40, GPS Max Fix Per Message: 24, Ird Retry Qty: 2, Ird Retry Delay: 2min, Accelerometer Threshold: 100
```

These settings can be seen below in the Settings section.

Version (Type V)

The **$ver** command will return a summary of both firmware and hardware versions:

```
Ascii: 04061815,V,OnyxTracker v1.25-3595. dev:4 Hw:2.3-0 SN=546 GPS: IRD:TA16005 Reset: Count=14, Current=(S), Prev=(cleared)
```

Version Readout	
04061815	Timestamp in UTC (Month/Day/Hour)
V	Type of message (Version)
Firmware Version	Product; Major, minor, build of firmware
Hardware Revision	Hardware revision, set during assembly
Serial	The unit's serial number
GPS Version	Firmware version of GPS chip
Iridium Version	Firmware version of Iridium modem
Reset Count	The number of resets since firmware was uploaded
Current	Cause of last power off
Previous	Cause of previous power off, not used in XMR

Position

Typical position messages are sent in compressed binary format (Message Type 10) to save on data usage and are parsed in XeosOnline. If XeosOnline is not used, GPS Text Short (Message Type 0) can be used to read position messages as P-Type.

Compressed Binary Position

The default format for positions is compressed to save on Iridium data usage. This binary format is parsed into a readable format by XeosOnline, placing its full contents in the Message and Location Logs.

The binary format is stackable and can transmit more than one position per Iridium transmission. In this situation, all recorded positions will be displayed in the Location Log, while the latest position will be displayed in the Message Log.

```
Timestamp: 2019-03-25T18:36:11.001Z, BatteryV: 11.91, Latitude: 44.714272, Longitude: -63.604870, Voltage Unloaded: 12.06
```

Message Log Readout	
Timestamp	Date and time in UTC of the latest position reading
BatteryV: 11.91	Loaded voltage of the power supply; minimum battery voltage observed during the previous Iridium transmission
44.714272	Latitude of fix, decimal degrees
-63.604870	Longitude of fix, decimal degrees
Voltage Unloaded: 12.06	Unloaded voltage of power supply; battery voltage data from measurement taken just prior to the turning on of the Iridium modem

Timestamp	Latitude	Longitude	Alarm	Bearing	Speed (km/h)	Speed (knots)	Altitude (m)	SNR
Mar 25 2019 06:36:08.001 PM	44.714400	-63.604902	false	0.0	0.000	0.000	0.0	44

Location Log Readout	
Timestamp	Date and time in UTC of this specific position reading
44.714400	Latitude of fix, decimal degrees
-63.604902	Longitude of fix, decimal degrees
Alarm: False	Indicates if the device has determined if it is in an alarm state
Bearing	Direction of movement determined by device
Speed (km/h)	Speed measurement in kilometers per hour
Speed (knots)	Speed measurement in knots
Altitude	Not used
SNR	SNR (Signal-to-noise ratio) of GPS Fix, higher is better (>37 is good)

ASCII Position (Type P)

In this format, only the most recent fix is sent at each interval, therefore it is most efficient to have GPS and Iridium intervals equal.

P-type messages are approximately 39 bytes:

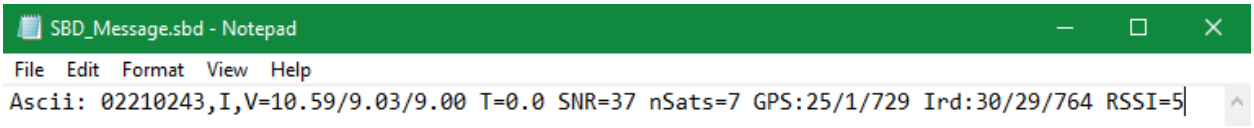
The message type used can be changed using the **$msgenable** command.

Position Readout	
06221600	Timestamp in UTC (Month/Day/Hour)
P	Type of message (Position)
44.71441	Latitude of fix, decimal degrees
-63.60495	Longitude of fix, decimal degrees
49	SNR (Signal-to-noise ratio) of GPS Fix, higher is better
348	Unloaded voltage of device at the time of GPS Fix

Information Message (Type I)

In situations where the device fails a GPS session through low SNR or other factor, the follow-up Iridium session will send a synopsis of the latest statistics from the device. This message will always be in ASCII format regardless of message format settings and approximately 80 bytes.

This message can be prompted using **$sysinf**



Information Readout	
02210243	Timestamp in UTC (Month/Day/Hour)
I	Type of message (Information)
V=10.59/9.03/9.00	Battery voltages unloaded/after Iridium session/after GPS session
T=0.0	Most recent temperature measurement, not used in XMR
SNR=37	MaxSNR (Signal-to-Noise ratio) of the last GPS attempt
nSats=7	Number of connected satellites during last GPS attempt
GPS:25/1/729	Quantity of good fixes since powerup, fails/Quantity of failed fixes/ Seconds total GPS has been powered on since power-up
Ird:30/29/764	Quantity of Iridium messages/Quantity of Iridium sessions/ Seconds total on time for Iridium
RSSI=5	Value reported back from the modem. Always a number between zero and five; five being the strongest signal.

Data Relay

The XMR allows for serial relay functionality, sending data to and from a connected serial peripheral device over the Iridium Satellite Network.

The XMR has a single RS-232 pair with which to communicate with the device for relay of external data to and from Iridium, or to monitor diagnostics.

On start-up, diagnostics will be disabled on the XMR, but can be enabled by issuing the command **$diag 1**, and disabled with **$diag 0**. The XMR will accept commands and send appropriate responses in either "diag" state, but not stream more detailed diagnostic data when diagnostics are off with the exception data incoming over Iridium from the **$outport** command. The XMR will always accept commands over Iridium.

Serial Relay

To send data to or from a connected serial device, connect to the serial port on the XMR. By default, the XMR will use these connection settings:

Baud Rate	57.6k
Parity	None
Data Bits	8
Stop Bits	1

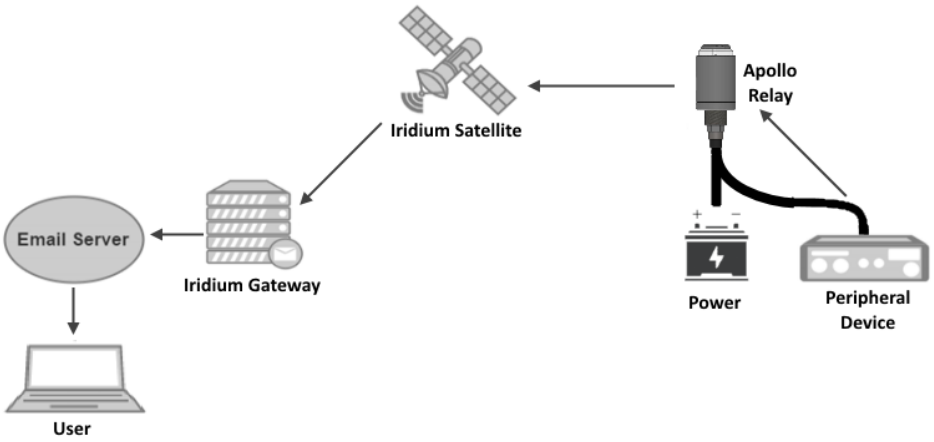
In the event the peripheral device has a different baud rate, the XMR can be changed to match for communication purposes using:

$baudrate XXXXX Y

XXXXX	Baud Rate
Y	Stop Bits (0 or 1)

The XMR will immediately use the commanded baud rate, **but this setting does not survive resets.**

Mobile Originated SBD (From Device)



To send a payload of data from the peripheral device via the XMR to the Iridium network, the peripheral device **must** send the payload in the following formats displayed:

$sendSBD
payload line 1
payload line 2
payload line 3
$finished

Note: Since the Diagnostic/Data Relay port is inactive while not in use, it is important to send a carriage return, line feed or both **before** sending the commands and payload.

All lines are terminated with either a carriage return, line feed, or both.

For example, using a carriage return as the line terminator for the same content as the previous example (represented with "\R"):

\R $sendSBD\R payload line 1\R payload line 2\R payload line 3\R $finished\R

\R $sendSBD\R The payload of the message can be single-lined as well\R $finished\R

The **$sendSBD** and **$finished** commands **must** be immediately preceded/followed by a line terminator character where displayed above. The payload can be data of any type and is not limited to ASCII characters.

Any data that exceeds the 330 byte SBD message limit will be split into several chunks and sent according to the Iridium device's SBD interval. Each Iridium transmission can send up to 8 SBD messages of 330 bytes each.

At default settings, the maximum sized message that can be queued for transmission is 4 kilobytes; this limit can be increased to 24 kilobytes, but this can cause loss of data and other issues if transmission conditions are not ideal. Contact Xeos Technologies if your requirements exceed default settings.

Mobile-originated messages will be transmitted over the Iridium satellite network from the Iridium device and arrive as an e-mail attachment. The first line in the e-mail attachment will have the following format:

devData X,S,Y,Z PAYLOAD

devData a,1,1,1 The payload of the message can be single-lined as well

X	The alphabetic representation of the port the device is connected to
S	The sequence number of the incremented with each new $sendSBD command
Y	The message part number
Z	The total number of parts to the complete message

Following the "devData" line will be all or part of the payload, depending on the size of the payload.

If the full payload of the device is unable to be sent, or if no part of the message is sent successfully, the data is queued for the next Iridium session.

Additional Data Relay Settings

Item	Description	Default	Min	Max
msfMM	Max backlog messages	20	1	200
msfMB	Max backlog bytes	4000	100	24000
msfMH	Max backlog age (hours)	24	1	144
msfMS	Data Relay message timeout (seconds)	5	2	120

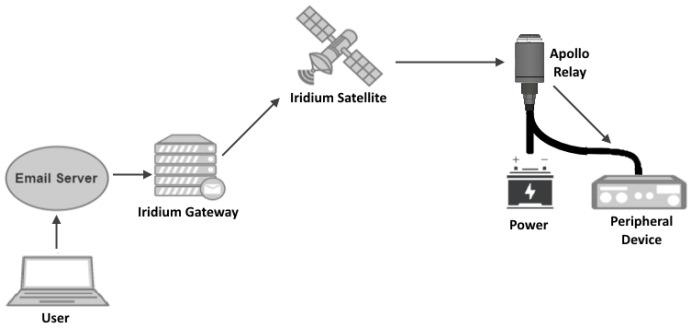
The above settings can be adjusted by sending the command, followed by X, where X is the new value for that setting.

$msfMB 6000

Sending any of the above commands without a new value prints off the currently employed settings for all of the Data Relay settings.

Note: If the **$sendSBD** command has been issued and new payload data has not been received after 5 seconds, the Iridium session will commence as if the **$finished** command was received. This setting can be adjusted using the **$msfMS** (message timeout) command.

Mobile Terminated SBD (To Device)



To send a payload of data to your connected serial device via the XMR, create a file with the extension ".sbd" with its contents adhering to the following format:

$unlock 12345
$outPort 1
payload line 1
payload line 2
payload line 3

Again, all lines are terminated with either a carriage return, line feed, or both. The unlock line will be a 5 digit numerical code, unique to each device. Its purpose is to prevent accidental or malicious commands from being processed by either the XMR, or your connected device.

When the data to be sent is compiled, normal methods of sending commands to Iridium devices can be implemented, either through email or a web service such as XeosOnline. Remember that XeosOnline will remember the unlock code of the device, so long as it has received it in the past and it has not been changed. As such, it does not need to be added to the command block being sent.

All lines after the **"$outPort 1"** line will be delivered to the connected serial device the next time the Apollo Relay is scheduled to contact the Iridium network. The total message, cannot exceed 270 bytes. Upon reception of the **$outPort 1** command, the XMR would output the following data to the connected serial device, according to the above example:

payload line 1
payload line 2
payload line 3

Check Iridium Mailbox

In cases where no data is to be transmitted out of the XMR, but incoming messages may be expected, the following command prompts an immediate Iridium "Message Check:"

$imsgchk

XMR Timer Intervals

The following are the default parameters for each mode.

Timer	<T0>	T1	T2	T3	T4	T5	T6	T7	Min-Max
SBD	1h	3m	1m	1m	1h	2h	3m	10m	25s-1d
GPS	1h	3m	20s	20s	2h	2h	1m	5m	20s-1d
ACS	2s	2s	2s	2s	2s	2s	2s	2s	1s-1m

Legend	
SBD	SBD (Iridium) transmission interval
GPS	GPS acquisition interval
ACS	Acceleration Sample Rate
T0 – T7	Timer number
<T#>	Timer currently in use
#s/m/h/d	Interval in seconds/minutes/hours/days
Min	Minimum allowed interval for this timer
Max	Maximum allowed interval for this timer

Timers		
Timer 0	Normal	Used if none of the following modes are triggered
Timer 2	In-Motion	Triggered when the device has detected sufficient motion
Timer 3	LiveTrack	Triggered by command for immediate fast tracking
Timer 4	No GPS Fix	Used when the GPS module continuously fails to acquire a fix
Timer 5	Low Battery	Used when XMR's power source drops below 7 Volts
Timer 6	Geofence	Triggers when the user's defined Geofence settings are triggered
Timer 7	Watch Circ.	Triggers when the device exits its set Watch Circle

Note: Timer 1, is not used in the XMR. Timer Parameter named ACP is not used.

Setting timers

$timer SYSTEM TIMER-MODE INTERVAL

Examples:
To set the **In-Motion Mode SBD** and **GPS** timer to a **10-minute interval**, issue the following two commands:

$timer SBD 2 10m
$timer GPS 2 10m

To set the **normal mode SBD** timer to a **1-day** interval, and the **normal mode GPS** timer to a **1-hour** interval, issue the following two commands:

$timer SBD 0 1d
$timer GPS 0 1h

The configuration above will result in the unit acquiring a GPS position every hour, which will cause all 24 GPS positions to be sent in a single SBD transmission once daily.

Watch Circle

The watch circle functionality allows users to monitor the position of their mooring by receiving alarm messages when it exits a user-defined circle. This feature aids in the recovery of moorings that could break free from their planned position.

Recognition of a watch circle exit is tied to the GPS interval of the device in Normal Mode **(Timer 0)**; once a GPS position is logged that shows the device is outside the circle, the device immediately transitions to Alarm Mode **(Timer 2)** and transmits according to Alarm Mode timer intervals.

Enabling the Watch Circle

The watch circle is disabled by default, but can be enabled by the following command:

$WCenable 1

The device will respond with a Switch message in ASCII showing watch circle (component C) is moved to the Y (yes) column.

```
Ascii: Switch: Y = GC, N = MO
```

Once functionality is enabled, the specifics of the user's circle can be implemented.

Setting the Watch Circle

The watch circle parameters can be set by sending the **$setcircle** command using this format:

$setcircle Latitude Longitude Radius

The minimum Radius is 50 meters, while the maximum is 15000 meters.

Example:

$setcircle 47.56989 -53.55682 100

a) **$setcircle** is the command
b) Latitude is set to **47.56989**
c) Longitude is set to **-53.55682**
d) Radius is set to **100** meters

Note:
If a watch circle is not set by command, but functionality is enabled, the XMR will place its centre on the average locations so far acquired (up to positions 48 hours old) and the default radius of 100m.

Upon implementing the watch circle, the Rover will return an **S-Type message** (see below):

```
Ascii: 10011801,S,Status / Value Change: Mode = OK gps = 15Mns ird = 15Mns ctr=44.71453/-63.60513 rad = 500m, Sched hr = 0
```

Once the watch circle is set, the unit will operate normally until it approaches the watch circle radius, at which point the device will send a warning message. Once the device exits the watch circle radius, it will begin sending alarm messages with GPS coordinates every 10 minutes **(Timer 2)** until the unit re-enters the watch circle, the radius is expanded via command, or is turned off.

Notes

- A maximum of 24 GPS positions can be included in a single SBD transmission. Unlike other settings, the watch circle's details are erased at each power up.
- Watch circle alert, watch circle warn and inverted mode use the same timer.
- You can adjust the Watch Circle's radius alone by setting the latitude and longitude parameters to 0, followed by the new radius.

Message Enable

The XMR has several message types that can be enabled for GPS and event messages. At factory defaults, the XMR will send compressed binary GPS (not human readable) and event (startup and battery) messages. These message formats can be changed using the **$msgenable** command.

The XMR has the following message types available for use:

Message Number	Message Type
0	GPS Plain Text Short
1	GPS Plain Text Long
10	GPS Bin Compressed

The **$msgenable** setting will be applied to each timer. The message format must be as follows:

$msgenable (mode, always zero) (message types separated by commas)

To enable GPS plain text short (Type 0), GPS Binary Compressed (Type 10), and GPS Text Long (Type 1) on Timer 0 (Normal mode) you would send the following command:

$msgenable 0 0,10,1

The XMR will respond to the command by displaying the numerical values for the used message formats.

Any number of message types can be enabled, but for each additional message type the XMR will use additional Iridium data.

Digital Output Pin

The Digital I/O pin of the XMR can be toggled via command to have a high or low logic output of 3V at a maximum of 5mA. The command is as follows:

$dOut X Y

As seen below, only one of the possible four number combinations leads to a high output.

Parameter	Purpose
$dOut	Base Command
X	Capability to turn the pin on/off; Enabled (1) or disabled (0)
Y	High (1) or Low (0)

Command	Pin Enable/Disable	Output High/Low	Result
$dOut	0	0	Output Low
$dOut	0	1	Output Low
$dOut	1	0	Output Low
$dOut	1	1	Output High

Encryption

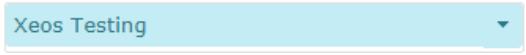
The XMR has some security features enabled by default, such as an unlock codes for receiving commands. Users can also enable end-to-end 256 bit AES encryption between their device and XeosOnline for an additional layer of security. Messages are zero-padded to fill out the 32 byte blocks.

Note: It is advantageous, though not required, to input the encryption key for the XMR into XeosOnline **first**, as the key cannot be retrieved from the XMR once it is installed (though it can be overwritten if unknown/forgotten **over the diagnostic port only**). If the device is in the field with an unknown or mismatched encryption key, commands sent to the device will not be properly encrypted when sent over Iridium, and thus improperly decrypted by the XMR, ending with the command not being valid.

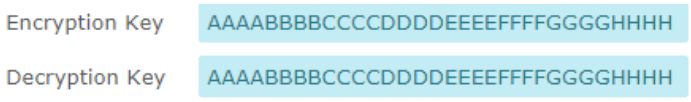
Setting Up Encryption on XeosOnline

1. Log in to XeosOnline and click on the Admin button at the top of the screen.

2. Ensure that your organization is selected in the Organization List



3. Find your device and click the edit device button 
4. Enter your passphrase into the appropriate text boxes and press **Save**



Now that the key is saved, it can be referenced when adding it to the XMR itself.

Generating your Key

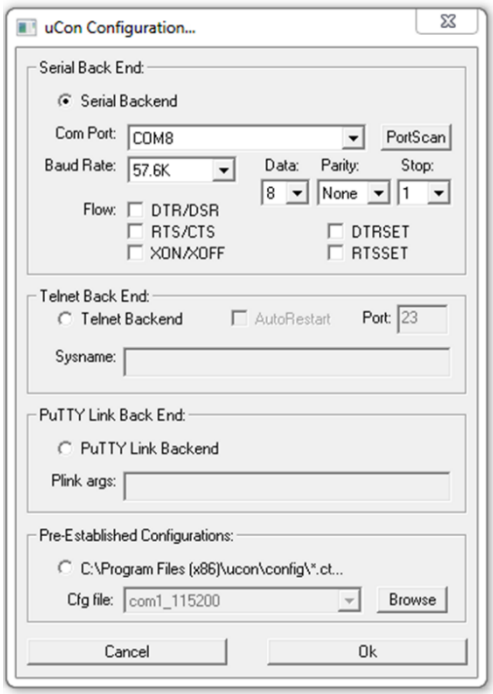
This process uses uCon Terminal Emulator, but any terminal emulator can be used.

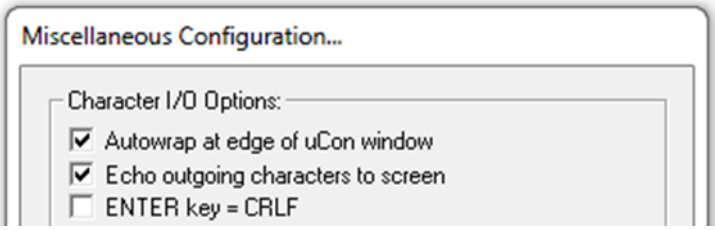
Getting the COM Port

1. Connect the XMR to your PC via the required cable.
2. Take note of the COM port number in Control Panel → Device Manager

Connect Using Terminal Emulator (uCon)

1. Download the uCon Serial Console software
2. Install uCon and open the program
3. Connect to the XMR using the following settings (right), use the COM port from the previous section.
4. Press Ok. Turn on echo by going to Config → Miscellaneous (see below)





5. Enter your encryption passphrase by entering
   $aky PASSPHRASE
   The pass phrase must be exactly 32 characters long.
   Note: This step cannot be done through Iridium.

Enabling Encryption Use

1. To enable encryption, send the command: **$aenbl 1**
2. To disable encryption, send the command: **$aenbl 0**

Flash Memory

The XMR comes equipped with internal flash memory, used to store logged events and GPS positions over the lifetime of the product. This information can be retrieved for data collection or troubleshooting purposes.

Dumping the Flash Memory

- Connect a PC to the XMR using the appropriate communications cable. External power is also required.
- Open the terminal emulator and use the typical port settings:

Baud Rate	57.6k
Parity	None
Data Bits	8
Stop Bits	1

- Send a **$ver** command to output the device's serial number and firmware version
- Send the flash dump command: **$fdumpall**
- The entire memory log of the device will begin outputting into the terminal. When the output slows, the dump is complete.
- Save the output in the appropriate manner for the terminal emulator

The flash logs of the XMR are comma delimited and organized by type of data.

Deleting the Flash Memory

The XMR's memory does not automatically erase. While the XMR can hold several thousand GPS positions it is prudent to clear the flash memory after extended deployments.

To delete the GPS records, send the command: **$flasheraseuserdata**

Erasing of the GPS records using this command can be done over Iridium if necessary.

To delete **ALL** data (via local connection only):

- Enter Engineering Mode: **$engmode 2009**
- Delete all flash memory: **$flasherasechip**

Either of the above deleting processes can take up to 30 seconds, after which the device will reset. Current settings in the XMR such as timers will not be changed by this process.

Other Commands

Below is a list of other commands that can be sent via Iridium or serially to acquire additional information, or configure the XMR. Remember to add the **$** symbol ahead of any commands.

Command	Parameters	Description
ver	---	Show hardware and firmware versions.
Status	---	Show operating status
sysInf	---	GPS/Ird summary
FactoryDefaults	---	Reset all to defaults and reset.
batt	---	Show battery status
switch	X 1/0	Switch component X on/off. No parameter lists components
enable	Y 1/0	Enable component Y on/off. Requires entry to Eng. Mode
ReportHour	Hour (0 – 23)	Set/Show Daily Reporting Hour offset from UTC
RepMinute	Minute (0 – 55)	Minutes offset past the hour for GPS/Iridium
ResetNow	---	Restarts the system. The effect is the same as restarting via the magnet

Testing the XMR

Before testing

The following should be confirmed ahead of testing:

- The XMR should be confirmed **activated** on the Iridium network to ensure that any messages transmitted are able to be received, and directed to send messages to the intended destinations.

Testing

Power-up and Batteries

- The XMR is externally powered; engage the supply to begin the test.
    - The XMR enters operation immediately on power-up.
    - If the red LED continuously flashes after start-up, a self-test failure has occurred. Restart the device and if this persists, contact Xeos.

Transmission

- Place the device upright in a location outside that has as much of the sky visible as possible; buildings and other tall objects can impede a beacon's ability to transmit.
    - If a covert deployment is upcoming, a second test to check the performance in a similar environment to deployment is suggested.
- The following messages are to be expected from the XMR after turning on within five minutes:
    - A general settings message laying out normal and in-motion timers
    - A power-up message displaying the firmware version and serial number:

```
Ascii: 01171326,V,Onyx_LV v1.27-6152. dev:4
Hw:2.3-0 SN=927 GPS:5xp__5.7.7-P1.GCC_N96LG+5xpt_5.7.7-P1.KC IRD:TA16005
Reset: Count=2, Current=(P), Prev=(cleared)
```

    - A GPS Position, the SNR strength of which should consistently be greater than or equal to 40. The Position will appear in both the message and location logs, and the SNR will appear in the location log only.

```
BatteryV: 11.88, Latitude: 44.714394, Longitude: -63.604902, Voltage Unloaded: 12.96
```

Latitude	Longitude	SNR
44.714195	-63.604902	41

The device will then send one GPS message every 60 minutes unless another event occurs. The SNR value for each GPS position should be a value no less than 38.

After this point, transmission intervals are tied to the rates laid out by Timer Zero.

Response to commands

During testing, commands can be sent to the device to change configuration or request statistics. These commands are read in during Iridium sessions and prompt responses from the XMR, confirming that said commands were acted upon.

- Send the **$timer** command, which will prompt the XMR to report its current timer intervals.

Response to $timer

```
Ascii: Tmr:SBD,1h,3m,1m,1m,1h,2h,3m,1h
Tmr:GPS,1h,3m,20s,20s,2h,2h,1m,1h
Tmr:ACP,1h,1h,1h,1h,1h,1h,1h,1h
Tmr:ACS,2s,2s,2s,2s,2s,2s,2s,2s
```

Deployment of the XMR

Below are installation tips for the XMR:

- The top of the XMR must have as good a view of the full sky as possible.
    - Alongside a wall or tall obstruction will cause coverage to suffer.
    - The antenna surface must not be obstructed by metal or wood. Thick amounts of other materials may also cause interference.
- XMR should not be deployed near (less than 1 meter) other transmitters.

Troubleshooting

Red LED/blinking
Indicative of a self-test failure. As of build 6058**,** using the **$status** command will show what failure code is present, but this does not survive resets.
i - Iridium - Modem cannot communicate with CPU of XMR
f - Flash - Flash error is showing as a full chip; erasing the chip may resolve this issue.

No Iridium messages (assuming no self-test failure)

- Determine the nature of the issue (antenna-related, system related or location-related)
  - Ensure the device's Iridium IMEI is active on the Iridium network with your service provider.
  - Ensure the device is provisioned to send messages where intended.
  - Change the location of the device; place the device in an area with as much total view of the sky as possible.
- Dumping the flash memory will display all Iridium sessions that occurred, regardless of success.

Messages are not readable in email (in binary)
Messages are in compressed binary format by default; use the **$msgenable** command to change the format to ASCII.

- Note that this would only apply to GPS positions and not data that is relayed.

No GPS/poor GPS

- The XMR's start-up message lists the firmware version of the GPS chip. If the GPS version is not displayed, the CPU is unable to communicate with the GPS chip.
- Change the location of the device; place the device in an area with as much total view of the sky as possible.
- Dumping the flash memory will display all GPS sessions that occurred, regardless of success.
  - Check the number of satellites seen (greater than 4), SNR level (greater than 38), time the GPS session was on (less than 60 seconds for first fix, less than 20 for other fixes). The response to the **$sysinf** command shows most recent occurrence of these stats.

Device is transmitting too frequently/not enough

- Check timer intervals with **$timer** command
- Check timer currently being used with the **$status** command (Returns TMde=0)
- Change timer interval as required

Device is going into alarm when it should not

- Watch circle is enabled when it should not be enabled.
  - Use **$switch c 0** to turn off watch circle functionality
  - Use the **$setcircle** command to adjust a circle that is of an inadequate size or location.

Appendix A: Firmware Update Procedure

Using the RS-232 port on the XMR, firmware can be updated.

Process

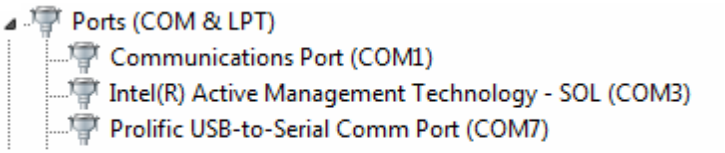
- Remove unit from field
- Update XMR firmware
- Test XMR's ability to transmit
- Once transmission is confirmed, safe to redeploy
- Notify Xeos of successful update

Requirements

- Use of a computer running some version of Windows OS
  - STerm Terminal Emulator Program
- DC Power source
- XMR power/communications cable (A-05-090) or equivalent
- RS-232 to PC cable
- XMR firmware package

Process

1. Power the XMR.
2. Connect the RS-232 cable to the XMR's diagnostic output.
3. Take note of the COM port number in Control Panel → Device Manager



In this case: **COM 7**

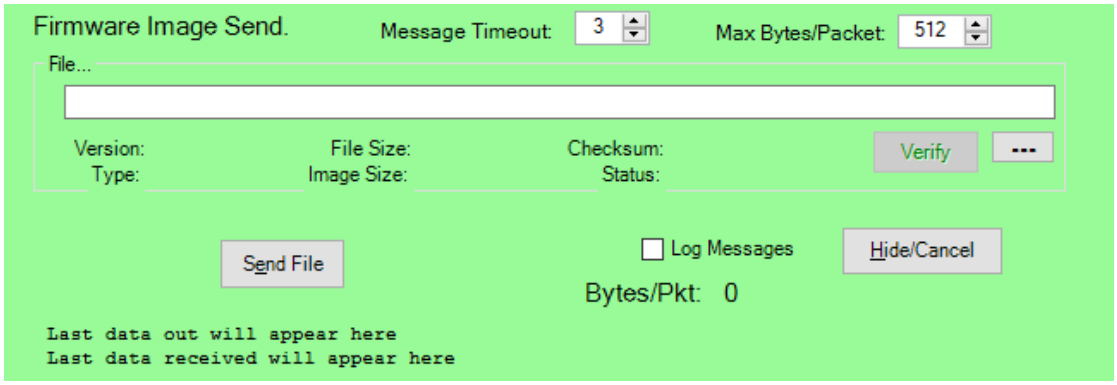
4. Extract the contents of the zipped folder to a location of your choosing:
   a. STerm.exe
   b. Firmware .bin file
5. Open **STerm** and connect to the COM port of the XMR using these settings in the bottom-right corner of the window:

Baud Rate	57.6k
Parity	None
Data Bits	8
Stop Bits	1

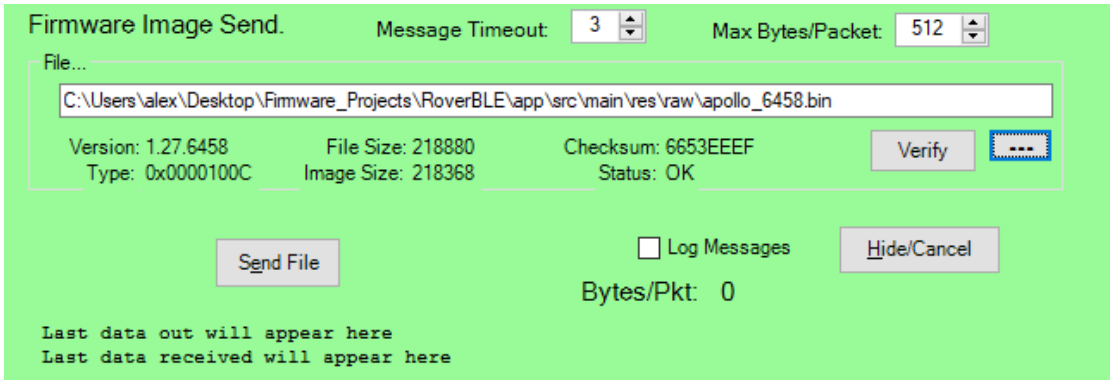
6. Click **Open Port**
7. Commands can be sent to the XMR on the bottom-left. Send a **ver** command to the device to confirm communication is established.

- All commands to Xeos devices start with a dollar sign (**$**) but STerm automatically adds this to all commands.
- A **ver** command will print out the product type and firmware version currently installed in the device.

8. At the top of the window, navigate to **Send File > Send Bootload Image**. A new inset box appears.



9. Use the "**…**" button next to Verify to browse to the .bin file extracted from the firmware package. Note that the firmware version of the new package will be populated on choosing the file.



10. Press **Send File.** The transfer will begin immediately and will take approximately five minutes.
    - If the transfer is interrupted for any reason (disconnect/power loss) the partial image will be discarded without corrupting the firmware. The process can be restarted without issue.
11. When the transfer is fully complete, press **Hide/Cancel**.
12. Return to where commands are issued to the device and send a **resetnow** command.
    - On resetting, the old firmware file is replaced with the new version over a period of approximately five to ten seconds.
13. Send the **ver** command to the device, confirming that the new version has been installed. On confirmation, the procedure is complete.

Testing the XMR

1. Once the XMR firmware has been updated, test its ability to transmit.
2. Make sure the XMR has an active Iridium account. Xeos can setup an account for you if you do not currently have one. Ensure you have access to one of the destinations to which messages are sent.
3. Place the XMR outside in a location where it has a clear view of the sky and connect it to an external power source.
4. Check the messaging destination to see if a transmission has been received. If no message is received within 20 minutes, please check operating manual to ensure it is setup as required.
5. If no message is received, please contact Xeos for assistance in troubleshooting.
6. DO NOT redeploy the XMR until Iridium transmission is confirmed.

Appendix B: Specifications

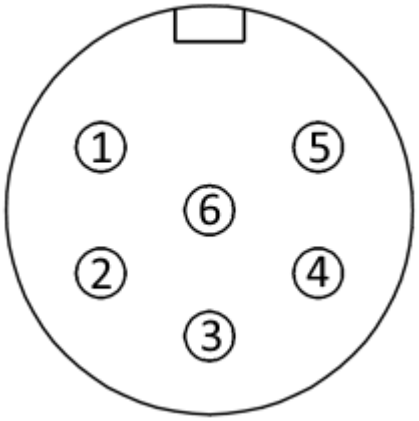
Electrical	
Connector	Chogori/Subconn MCBH-4-FS-TI/Subconn MCBH-6F-TI
IP Rating	IP-67 (Chogori), IP-68 (Subconn)
Supply Input	7 – 32V
Idle Current	2.5mA
GPS Current (~20s avg.)	15mA
Iridium Current (30s avg.)	50mA
Dimensions (Diameter, Height)	2.5", 2.6"

All measurements taken at 12VDC

Electronic	
GPS Hardware	Xeos Technologies 48 channel GPS (SiRFStarV)
Iridium Hardware	Iridium 9603 modem
Antenna	Integrated Iridium and GPS antennas
Operating Temperature	Operating Temperature: -20° C to +60° C

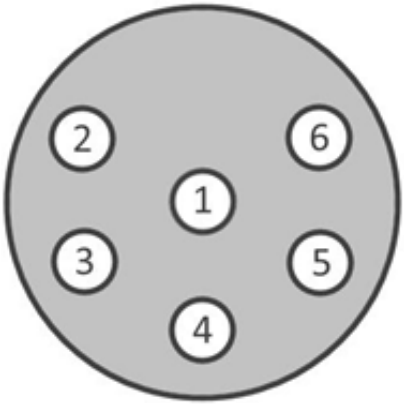
Appendix C: Pinout Diagram

6-Pin Chogori Connector, Male	
Pin	Purpose
1	Ground (-) input
2	Data input to XMR (Rx-RS-232)
3	Data output from XMR (Tx-RS-232)
4	No Connection
5	Positive (+) input
6	Digital I/O



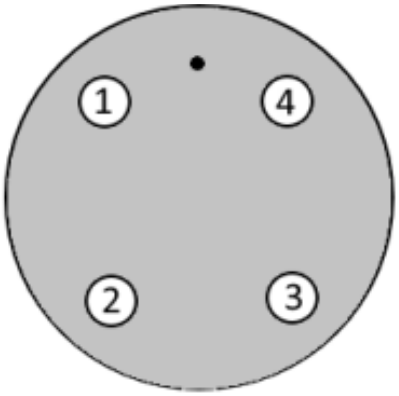
EXTERNAL VIEW

6-Pin Subconn Connector, Female, **MCBH-6F-TI**	
Pin	Purpose
1	Ground (-) input
2	Data input to XMR (Rx-RS-232)
3	Data output from XMR (Tx-RS-232)
4	No Connection
5	Positive (+) input
6	Digital I/O



EXTERNAL VIEW

4-Pin Subconn Connector, Female, **MCBH-4-FS-TI**	
Pin	Purpose
1	Ground (-) input
2	Data input to XMR (Rx-RS-232)
3	Data output from XMR (Tx-RS-232)
4	Positive (+) input



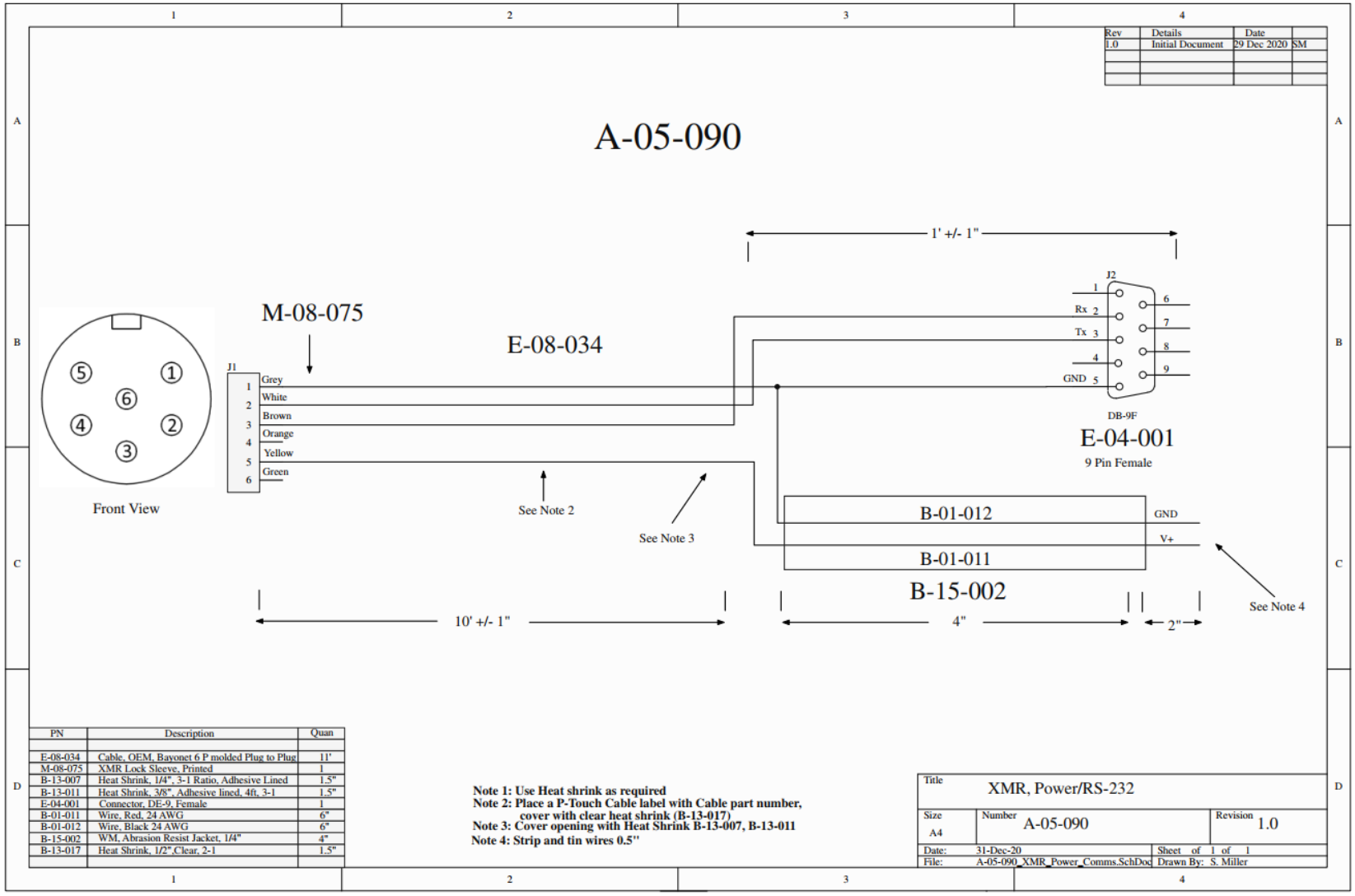
EXTERNAL VIEW

Appendix D: Mechanical Drawings


Ø1.35
4 x Ø 8-32 UNC ↧.25
EQUISPACED
4 OR 6 PIN SUBCONN
2.50
XMR
Xeos Beacons™
Xeos Technologies Inc.
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
ONE PLACE DECIMAL ±0.05
TWO PLACE DECIMAL ±0.005
THREE PLACE DECIMAL ±0.001
Physical Properties
-SUBCONN VARIANT, CONNECTOR DEPENDANT ON FEATURE SET
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF XEOS TECHNOLOGIES. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF XEOS TECHNOLOGIES IS PROHIBITED
TITLE:
XMR
SIZE
A
Rev
2
SCALE
1:1
Sheet
2 of 2

Appendix E: Cable Drawings

A-05-090: XMR Power/Communications Cable

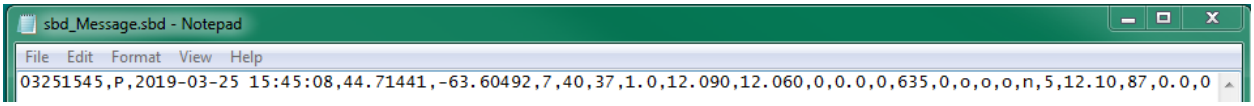


Appendix F: GPS Text Long

GPS Text Long can be used for a more detailed summary of GPS information. This form is only presented in ASCII format and due to its length is a larger file, at around 110 bytes depending on the information sent.

Like GPS Text Short, only the most recent GPS position is sent, therefore this message format cannot be stacked.

Greyed out fields are not used in the XMR.



GPS Text Long Readout	
03251545	Date and Time, MMDDHHMM
P	Position Type message
2019-03-25 15:45:08	Date and time, adds year and second of fix
44.71441	Latitude
-63.60492	Longitude
7	Number of satellites seen
40	Maximum SNR of fix
37	Time to fix
1.0	HDOP (Horizontal Dilution of Precision)
12.090	Loaded Voltage
12.060	Unloaded Voltage
0	Temperature
0.0	Speed
0	Heading
635	GPS Horizontal Error
0	Watch Circle Set 1 (Yes) 0 (No)
o	Geofence 1 In/Out
o	Geofence 2 In/Out
o	Geofence 3 In/Out
n	Alarm State (Motion Only)
5	RSSI (Iridium signal strength)
12.10	Battery Voltage
87	On-seconds, seconds since power-up/last reset
0.0	Degrees Celcius
0	Movement counter

Warranty, Support and Limited Liability

Xeos Technologies Inc. warranties the XMR to be free of defects in material or manufacturing for a period of one year following delivery. Liability is limited to repair or replacement of the defective part and will be done free of charge.

LIMITED WARRANTY: Xeos Technologies Inc. warrants that the product will perform substantially in accordance with the accompanying written materials for a period of one year from the date of receipt.

CUSTOMER REMEDIES: Xeos Technologies Inc. entire liability and your exclusive remedy shall be at Xeos Technologies Inc. option, either (a) return of the price paid or (b) repair or replacement of the product that does not meet Xeos Technologies Inc. Limited Warranty and that is returned to Xeos Technologies Inc. with a copy of your receipt. This Limited Warranty is void if failure of the product has resulted from accident, abuse, or misapplication. Any replacement product will be warranted for the remainder of the original warranty period or ninety (90) days, whichever is longer.

NO OTHER WARRANTIES: Xeos Technologies Inc. disclaims all other warranties, either express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the product or the accompanying written materials. This limited warranty gives you specific legal rights. You may have others, which vary from state to state.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES: In no event shall Xeos Technologies Inc. or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of equipment, for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use this Xeos Technologies Inc. product, even if Xeos Technologies Inc. has been advised of the possibility of such damages.