Satellite Support

Model	CTM - 200
Revision:	Rev 1.5

3066 Beta Avenue | Burnaby, B.C. | V5G 4K4

Revision Control

Description	Revision	Date
Added cmd sat pad	Rev 1.5	21-Oct-2013
Formatting changes	Rev 1.4	03-Sept-2013
Added cmd sat force	Rev 1.3	15-Aug-2013
Added failover MODE LED states, updated Pendant section	Rev 1.2	09-Aug-2013
Customer Release	Rev 1.1	19-July-2013
Initial Draft	Rev 1.0	27-June-2013

Contents

Glossary of Terms

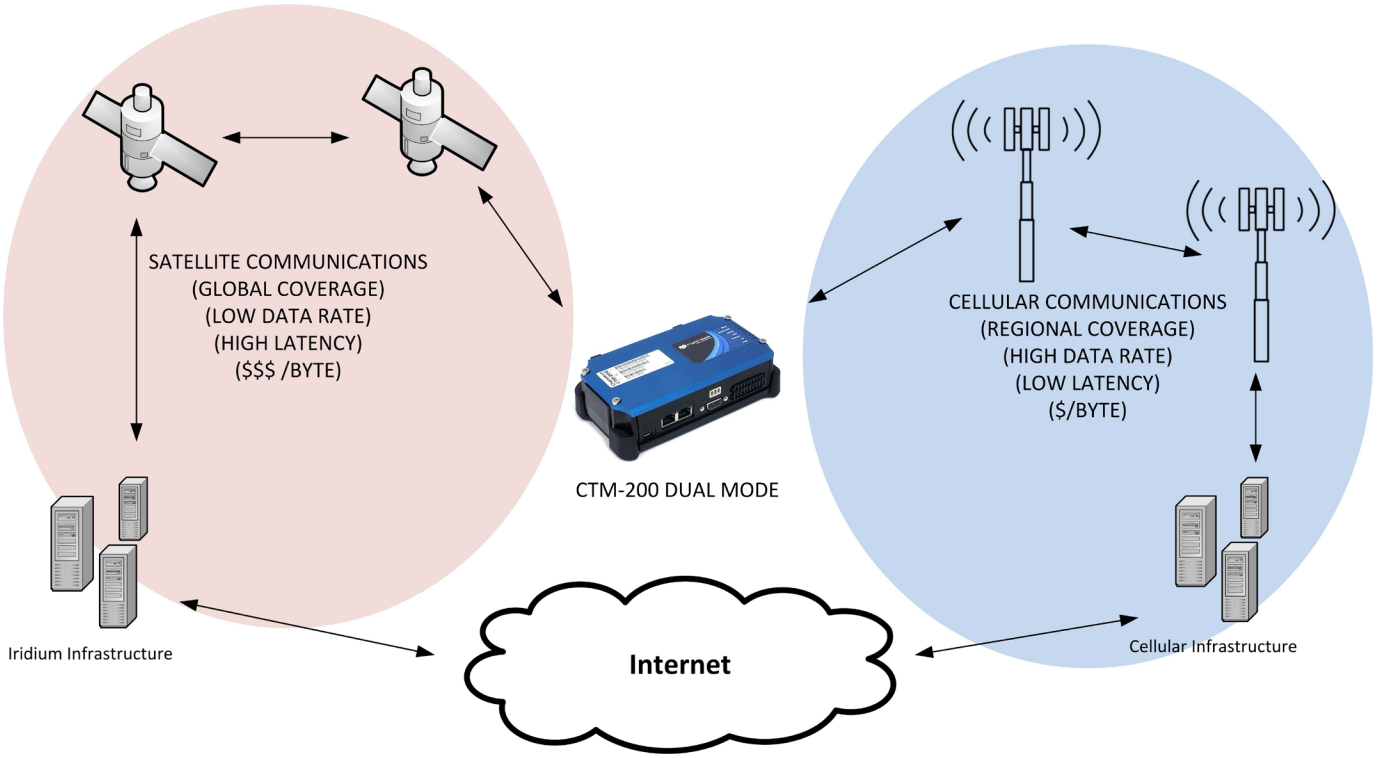
ISU - Iridium Subscriber Unit (e.g. SBD9602) on remote device (e.g. CTM200)
MO - Mobile Originated (e.g. MO messages are messages coming from remote device)
MT - Mobile Terminated (e.g. MT messages are messages sent to remote device)
PAD - Packet Assembly/Disassembly
SQI - Signal Quality Index (a relative measurement of signal quality)

1. Overview

The Chameleon CTM-200 "Dual Mode" Gateway features an Iridium Subscriber Unit (ISU) satellite module (SBD9602) and a mini-PCI express/NGFF form factor cellular wireless module. With the internal ISU, **reports**, **cmd commands** and **Packet Assembly/Disassembly (PAD)** can all be sent via satellite.

Note: CTM-200 firmware 2.0.5.3356 and later is required

Note: Please ensure that the ISU is already provisioned with Iridium and can access their Iridium Gateway via email and/or direct IP. Please contact Cypress Solutions for more information.



Dual Mode Features:

- "Smart switching" Use Cellular when available, when cellular coverage is marginal switch to Satellite for 100% coverage
- "Data Sensitive" Data packets over cellular are gated to limit data cost. Allowances for "critical" reports such as Man-Down alerts.
- Remote configuration and control via cellular or satellite

Users should be aware of the benefits and limitations of using satellite. The table below shows a general comparison between satellite and cellular usage.

	Satellite	Cellular
Coverage	Global	Regional
Bandwidth	Low (bytes)	High (Mbits)
Latency	High (minutes)	Low (milliseconds)
Cost/Byte	$$$	$

Satellites offer greater terrestrial coverage than cellular, but have an associated higher cost. To minimize the data cost of satellite communications, CTM-200 has specific features to limit satellite traffic, which are described in this document.

It is also important for users to understand that typical reports configured for cellular transmission, such as the one shown below, might not be suitable for satellite transmission because of its limited data.

```
$PEVENT,153725.00,A,160713,356215040095550,GPS4:t>240*52[0x0D][0x0A]
$PGPS,153725.00,A,4915.4300,N,12259.8187,W,000.0,359.0,160713,+00021,10,356215040095550*7A[0x0D][0x0A]
$POBDA,A,p1=0,p2=0.00,p3=18,p4=5TEMU52N27Z410952,p7=0,p8=88,p22=31641,p14=0*0C[0x0D][0x0A]
$PPWR,12.59,0,40*3C[0x0D][0x0A]
```

CTM-200 offers some binary messages to help reduce data use, and hence, cost. Binary message are much more efficient, such as message #113 below which is similar to the $PGPS message above. It uses only 37 bytes of data versus the 80+ bytes of the ascii message to transmit time, latitude, longitude, and modem ID.

```
[0x71][0x0C][0x06][0x00][0x00][0x1D][0x00][0x00][0xE4][0x67][0xE4][0x51][0xF1][0x28][0x4B][0x00][0x72][0x52]
[0x44][0xFF][0xC0][0x00][0x33][0x35][0x36][0x32][0x31][0x35][0x30][0x34][0x30][0x34][0x31][0x30][0x31][0x30]
[0x36]
```

2. Satellite message format

2.1 Format of messages sent from CTM-200

Messages sent from the CTM-200 (e.g. **reports**, **cmd command response, PAD**) are MO messages and can be sent in both email and direct IP (TCP) format. Please consult with your account provider to configure on Iridium's SPNet whether MO messages are to be sent via email or direct IP (TCP) or both.

2.1.1 MO email message format

Messages arriving in email format will be similar to the following:

From: sbdservice@sbd.iridium.com
To: <USER EMAIL>
Subject: SBD Msg From Unit: <ISU'S IMEI>
Attachment: <IMEI>_<MO_MSG_SEQNUM>.sbd
Body:
MOMSN: <MO_MSG_SEQ_NUM>
MTMSN: <MT_MSG_SEQ_NUM>
Time of Session (UTC): <Date> Session Status: <STATUS> - Transfer <STATUS MSG> Message Size (bytes): <MO_MSG_SIZE>

Unit Location: Lat = <LATITUDE> Long = <LONGITUDE> CEPradius = <CEPRADIUS>

The payload of interest is contained in the .sbd attachment file, in raw binary format without any headers or additional info. **It can be up to 340 bytes.** The following is an example email and the contents of the accompanying attachment file (CTM-200 reports sent over satellite):

From: sbdservice@sbd.iridium.com
To: kmo@cypress.bc.ca
Subject: SBD Msg From Unit: 300234123412341
Attachment: 300234123412341_000497.sbd

MOMSN: 497
MTMSN: 0
Time of Session (UTC): Wed Jul 17 00:12:59 2013 Session Status: 00 - Transfer OK Message Size (bytes): 105

Unit Location: Lat = 49.28267 Long = -123.01926 CEPradius = 8

300234123412341_000497.sbd contains the following:

```
$PIND,000000,0*23
$PINA,1,0.00*39
$GPRMC,232617.00,A,4915.4103,N,12259.7944,W,000.0,036.0,160713,,*29
```

2.1.2 MO direct IP message format

Messages sent via TCP directly to a destination IP are of the following format:

	Data Type	Length (bytes)	Value
Protocol Revision Number	char	1	0x01
Overall Message Length	unsigned short	2	N+48
MO Header ID	char	1	0x01
MO Header Length	unsigned short	2	0x1C
CDR Reference (Auto ID)	unsigned integer	4	0-4294967295
IMEI	char	15	ASCII numeric characters
Session Status	unsigned char	1	0,1,2,10,12-15
MOMSN	unsigned short	2	1-65535
MTMSN	unsigned short	2	0-65535
Time of Session	unsigned integer	4	Epoch time
MO Location Information ID	char	1	0x03
MO Location Information Length	unsigned short	2	0x0B
Latitude/Longitude	double	7	See table below
CEP Radius	unsigned integer	4	1-2000
MO Payload Header ID	char	1	0x02
MO Payload Length	unsigned short	2	N (up to 340)
MO Payload	char	N (up to 340)	Payload bytes

Latitude/Longitude format:

<table>
<tr><th colspan="8">MSBit LSBit</th><th rowspan="2">Byte Number</th><th rowspan="2">Description and Allowed Values</th></tr>
<tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr>
<tr><td colspan="4">Reserved</td><td colspan="2">Format Code</td><td>NSI</td><td>EWI</td><td>1</td><td>Reserved & Format Code: 0
NSI: North/South Indicator (0=North, 1=South)
EWI: East/West Indicator (0=East, 1=West)</td></tr>
<tr><td colspan="8">Latitude (degrees)</td><td>2</td><td>0-90</td></tr>
<tr><td colspan="8">Latitude (thousandths of a minute, MSByte)</td><td>3</td><td rowspan="2">0-59999</td></tr>
<tr><td colspan="8">Latitude (thousandths of a minute, LSByte)</td><td>4</td></tr>
<tr><td colspan="8">Longitude (degrees)</td><td>5</td><td>0-180</td></tr>
<tr><td colspan="8">Longitude (thousandths of a minute, MSByte)</td><td>6</td><td rowspan="2">0-59999</td></tr>
<tr><td colspan="8">Longitude (thousandths of a minute, LSByte)</td><td>7</td></tr>
</table>

The following is an example TCP message containing the same information as in 4.1.1's example:

[0x01][0x00][0x99][0x01][0x00][0x1C][0x04][0xB8][0xF8][0xEE][0x33][0x30][0x30][0x32][0x33][0x34][0x31][0x32][0x33]
[0x34][0x31][0x32][0x33][0x34][0x31][0x00][0x01][0xF1][0x00][0x00][0x51][0xE5][0xE1][0x8B][0x03][0x00][0x0B][0x01]
[0x31][0x42][0x40][0x7B][0x04][0x83][0x00][0x00][0x00][0x08][0x02][0x00][0x69][0x24][0x50][0x49][0x4E][0x44][0x2C]
[0x30][0x30][0x30][0x30][0x30][0x30][0x2C][0x30][0x2A][0x32][0x33][0x0D][0x0A][0x24][0x50][0x49][0x4E][0x41][0x2C]
[0x31][0x2C][0x30][0x2E][0x30][0x30][0x2A][0x33][0x39][0x0D][0x0A][0x24][0x47][0x50][0x52][0x4D][0x43][0x2C][0x32]
[0x33][0x32][0x36][0x31][0x37][0x2E][0x30][0x30][0x2C][0x41][0x2C][0x34][0x39][0x31][0x35][0x2E][0x34][0x31][0x30]
[0x33][0x2C][0x4E][0x2C][0x31][0x32][0x32][0x35][0x39][0x2E][0x37][0x39][0x34][0x34][0x2C][0x57][0x2C][0x30][0x30]
[0x30][0x2E][0x30][0x2C][0x30][0x33][0x36][0x2E][0x30][0x2C][0x31][0x36][0x30][0x37][0x31][0x33][0x2C][0x2C][0x2A]
[0x32][0x39][0x0D][0x0A]

2.2 Format of messages sent to CTM-200

Messages sent to the CTM-200 (e.g. **cmd commands, PAD**) are MT messages and can also be sent in email format. To prevent unauthorized users from sending messages over the satellite network, it is possible to restrict access to up to 10 emails and/or IPs. This prevents third parties from abusing the system and accruing data charges.

2.2.1 MT email message format

Messages can be sent to the CTM-200 via email. Simply save the payload as the content of an attachment with extension **.sbd** and send the attachment to an Iridium Gateway, such as **data@sbd.iridium.com,** with the remote ISU's IMEI as the subject.

Note: Each attachment can **only contain up to 270 bytes**.

A sample email format is as follows:

From: kmo@cypress.bc.ca
To: data@sbd.iridium.com
Subject: 300234123412341
Attachment: msgtoctm200.sbd

The attachment msgtoctm200.sbd contains the payload to send to the CTM-200:

$1cmd ver

3. Configuration Examples

This section is intended to provide examples of commonly used satellite configurations. While satellite commands are listed in section 6, a complete list of commands can be found online at http://www.cypress.bc.ca/command_reference.html.

Note: All configuration examples must end with a **cmd save** and **cmd pwr mode 2** to take effect.

3.1 Reporting over satellite

Reports can be sent over satellite by choosing **report type 10** for remote servers in **cmd reptype**, **cmd iorep** and **cmd gpsrep**.

Note: Local reports cannot be used with satellite reports for each specific report number

3.1.1 Single report example

This example demonstrates how to configure a single report to report over satellite when the supply voltage is below 10V. The main satellite related commands to be concerned with are in the first 2 lines.

```
cmd sat enable 1          // enable satellite
cmd reptype 1 0 10 0      // satellite type (10) and no local reports allowed
cmd vcccond 1 1 10        // trigger Vcc cond #1 every 1m while Vcc is below 10V
cmd vccreport 1 1         // associate Vcc cond #1 to use report #1
cmd repaddmes 1 116       // add message #116 ($PPWR) to report #1
```

If local reports are also required to be sent in addition to the satellite report, add a new local report based on the same triggering condition. Please note that this local report will not be limited by **cmd sat mt**, so it will be reporting whenever the triggering condition occurs.

```
cmd sat enable 1          // enable satellite
cmd reptype 1 0 10 0      // satellite type (10) and no local reports allowed
cmd reptype 2 3 0 0       // local reporting over UDP - this will send every 1m because of vcccond below
cmd vcccond 1 1 10        // trigger Vcc cond #1 every 1m while Vcc is below the threshold voltage of 10V
cmd vccreport 1 1 2       // associate Vcc cond #1 to use report #1 and report #2
cmd repaddmes 1 116       // add message #116 ($PPWR) to report #1
cmd repaddmes 2 116       // add message #116 ($PPWR) to report #2
```

3.1.2 Satellite minimum time example

To help limit data use, **cmd sat mt** configures the minimum time between identical report numbers being sent over satellite. Reports generated before the minimum time is reached are discarded. If **cmd sat mt** is not configured, then the default of 10 minutes is used (**cmd sat mt 10**.)

This example demonstrates how 2 GPS triggered reports (GPS report 1 set to report every 1 min, and GPS report 2 set to report every 2 minutes) configured to report over satellite works. What is of interest here is that **cmd sat mt 3** is configured to limited GPS reports 1 and 2 to a minimum time of 3 minutes between each respective report.

```
cmd sat enable 1          // enable satellite
cmd sat mt 3              // set the minimum time between each identical report numbers to 3 min
cmd gpscond 1 1 60        // GPS report 1 to trigger every 1m
cmd gpscond 2 1 120       // GPS report 2 to trigger every 2m
cmd gpsrep 1 0 10 0       // GPS report 1 to use satellite
cmd gpsrep 2 0 10 0       // GPS report 2 to use satellite
cmd gpsaddmes 1 113       // add binary message 113 to GPS report 1
cmd gpsaddmes 2 113       // add binary message 113 to GPS report 2
```

The above configuration will result in the following behavior (last 2 columns are the final results):

Time (minutes)	Rep 1 Triggered	Rep 2 Triggered	Time since last rep 1	Time since last rep 2	**GPS Report 1 Result**	**GPS Report 2 Result**
1	Y		0*		**SENT**	
2	Y	Y	1	0*	**DISCARDED**	**SENT**
3	Y		2		**DISCARDED**	
4	Y	Y	3*	2	**SENT**	**DISCARDED**
5	Y		1		**DISCARDED**	
6	Y	Y	2	4*	**DISCARDED**	**SENT**
7	Y		3*		**SENT**	
8	Y	Y	1	2	**DISCARDED**	**DISCARDED**
9	Y		2		**DISCARDED**	
10	Y	Y	3*	4*	**SENT**	**SENT**
11	Y		1		**DISCARDED**	
12	Y	Y	2	2	**DISCARDED**	**DISCARDED**
13	Y		3*		**SENT**	
14	Y	Y	1	4*	**DISCARDED**	**SENT**
…	…	…	…	…	…	…

*First report or minimum time has been reached, so allowed to send

3.1.3 Failover handling example

Satellites can be used as an alternative should the primary reporting channel be disrupted. In this example, an I/O report is configured to report to a remote IP via TCP, with satellite as a failover option. It sets I/O reports 1 and 2 to be based on the same triggering condition, with I/O report 2, which is sent over satellite, only sent when CTM detects cell is down. Note that because **cmd sat mt** is not set, the default value of 10 min is used.

```
cmd sat enable 2          // dynamic mode
cmd iocond 1 1 60         // generate I/O report #1 every 1 min
cmd iocond 2 1 60         // generate I/O report #2 every 1 min
cmd iorep 1 0 5 0         // remote TCP
cmd iorep 2 0 10 0        // satellite type
cmd ioaddmes 1 80         // add message 80 to I/O report #1
cmd ioaddmes 2 80         // add message 80 to I/O report #2
```

The above configuration will result in the following:

Time (minutes)	Time since last I/O rep 2	I/O Report 1	I/O Report 2
1	SAT POWERED OFF	**SENT**	**DISCARDED**
CELL OFFLINE	SAT POWERED ON	-	-
2	0*	**NOT SENT**	**SENT**
3	1	**NOT SENT**	**DISCARDED**
4	2	**NOT SENT**	**DISCARDED**
5	3	**NOT SENT**	**DISCARDED**
6	4	**NOT SENT**	**DISCARDED**
7	5	**NOT SENT**	**DISCARDED**
8	6	**NOT SENT**	**DISCARDED**
9	7	**NOT SENT**	**DISCARDED**
10	8	**NOT SENT**	**DISCARDED**
11	9	**NOT SENT**	**DISCARDED**
12	10*	**NOT SENT**	**SENT**
13	1	**NOT SENT**	**DISCARDED**
14	2	**NOT SENT**	**DISCARDED**
15	3	**NOT SENT**	**DISCARDED**
16	4	**NOT SENT**	**DISCARDED**
17	5	**NOT SENT**	**DISCARDED**
18	6	**NOT SENT**	**DISCARDED**
19	7	**NOT SENT**	**DISCARDED**
20	8	**NOT SENT**	**DISCARDED**
21	9	**NOT SENT**	**DISCARDED**
22	10*	**NOT SENT**	**SENT**
23	1	**NOT SENT**	**DISCARDED**
CELL ONLINE	SAT POWERED OFF	-	-
24	SAT POWERED OFF	**SENT**	**DISCARDED**

*First report or minimum time has been reached, so allowed to send

In the satellite failover handling case, when cell is down the MODE LED displays whether the ISU module is powered on and whether it is registered on the satellite network:

ISU Module Status	ISU Module Registered on Network?	**MODE LED Status**
OFF	NO	**OFF**
ON	NO	**Blink once every second**
ON	YES	**Blink three times every second**

3.1.4 On demand reporting over satellite example

It is also possible to send messages on demand over satellite while in failover/dynamic mode. This could be used with report triggered scripts for versatile reporting (e.g. turn ISU module on, send a report to satellite, then turn ISU module off, etc.)

> Note: **ON** states have priority, meaning **cmd sat pwr 1** and **CTM's decision to turn ISU module ON in failover mode have priority**.

In this example, I/O report 2 is sent on demand over satellite, even when there is a cellular connection:

```
cmd sat enable 2          // dynamic mode
cmd iocond 1 1 60         // generate I/O report #1 every 1 min
cmd iocond 2 1 60         // generate I/O report #2 every 1 min
cmd iorep 1 0 5 0         // remote TCP
cmd iorep 2 0 10 0        // satellite type
cmd ioaddmes 1 80         // add message 80 to I/O report #1
cmd ioaddmes 2 80         // add message 80 to I/O report #2
```

The following two commands can be used to enable or disable satellite on the fly:

```
cmd sat pwr 1             // turn satellite ON

cmd sat pwr 0             // turn satellite OFF
```

With the above commands, it is possible to do the following:

Time (minutes)	Time since last I/O rep 2	**I/O Report 1**	**I/O Report 2**
1	SAT POWERED OFF	**SENT**	**DISCARDED**
CELL OFFLINE	SAT POWERED ON	-	-
2	0*	**NOT SENT**	**SENT**
3	1	**NOT SENT**	**DISCARDED**
CELL ONLINE	SAT POWERED OFF	-	-
4	SAT POWERED OFF	**SENT**	**DISCARDED**
cmd sat pwr 1	SAT POWERED ON	-	-
5	3	**SENT**	**DISCARDED**
6	4	**SENT**	**DISCARDED**
7…11	5…9	**SENT**	**DISCARDED**

12	10*	**SENT**	**SENT**
cmd sat pwr 0	SAT POWERED OFF	-	-
13	SAT POWERED OFF	**SENT**	**DISCARDED**

*First report or minimum time has been reached, so allowed to send

Another example is if CTM is configured to report over satellite if an I/O's threshold is exceeded:

There may be situations where it is desirable to occasionally send a report over satellite. Because of its infrequency, the satellite module could be powered off most of the time, and only powered on to send a report. This could be accomplished in conjunction with report triggered scripts.

In this example, a CTM is configured to report over satellite if an input voltage exceeds a configured threshold voltage. It is expected that the input voltage does not typically exceed the threshold voltage during normal use, so these reports are supposed to be infrequent.

Step 1:

Configure the CTM to trigger a report (e.g. I/O report #1) that will run a script to check the input voltage. If the voltage is above the set threshold, the script will power ON the ISU module and send the desired report (e.g. I/O report #2). If the voltage is below the threshold, the script will power OFF the ISU module.

A sample configuration is as below:

```
cmd sat enable 2          // dynamic mode
cmd autoconnect 0         // disable connecting to external cellular network to disable failover mode
cmd reportscripts 1       // enable report triggered scripts
cmd insetup 1 A 7.5 P 2   // set input #1 to trigger when measured voltage crosses 7.5V threshold
cmd iocond 1 1 60 1       // generate I/O report #1 every 1 min if input #1 is triggered - used to trigger script
cmd iorep 2 0 10          // set I/O report number #2 to use satellite type
cmd ioaddmes 2 21         // add message 21 ($PINA for input #1) to I/O report #2
```

Step 2:

Add the following into a file named report1040.sh (1040 corresponds to I/O report #1):

```
# Grab current input voltage
tVal=$( cmd input | awk 'NR==1' | awk -FV '{print $1}' )
# Grab threshold
tThreshold=$( cmd insetup | awk 'NR==1' | awk '{print $5}' )
# Compare decimal values
tRes=$( echo $tVal $tThreshold | awk '{ if ($1 > $2) print "1"; else print "0"}')
# Enable and send report or disable satellite based on comparison above
if [ $tRes -gt 0 ] ; then
 cmd sat pwr 1
 cmd sendreport 1041
else
 cmd sat pwr 0
fi
```

The previous script will check the input voltage when input #1 is triggered, and enable the ISU module and sent I/O report #2 (1041 corresponds to I/O report #2) if the input voltage is above the configured threshold, or disable the ISU module to save power.

Step 3:
Save the file in **/etc/reports/scripts** on the CTM and run the following two commands in CTM's console:

```
dos2unix report1040.sh
chmod +x report1040.sh
```

Step 4:
Run **cmd save** as done for all configuration changes to save the script.

The previous setup will result in a behavior similar to the following:

Time (minutes)	Input #1	I/O Report 1 script triggered?	Greater than threshold?	ISU module power	I/O Report 2 triggered?	Time since last I/O rep 2	**I/O Report 2 Result**
1	3V			OFF			
2	8V	Y	Y	ON	Y	0	**SENT**
3	8V			ON			
4	8V			ON			
5	3V	Y	N	OFF	N		
6	3V			OFF			
7	3V			OFF			
…	…	…	…	…	…	…	**…**
N	9V	Y	Y	ON	Y	N-2	**SENT if N-2 is greater than 10, else it is DISCARDED**
N+1	9V			ON			
N+2	4V	Y	N	OFF	N		
N+3	4V			OFF			

3.1.5 Man Down Pendant Panic reports over satellite example

Man Down pendant panic reports can also be sent over satellite. However, because panic (PANIC/MPANIC/FPANIC) reports are of high priority, these panic reports are handled differently than all other reports.

Panic reports are still restricted by **cmd sat mt**, but they are matched against a list of previous Man Down pendant IDs instead of a list of previous report numbers sent over satellite. That is, although all panic reports use the same report number set in **cmd panicreport**, panic reports are gated based on pendant ID to prevent one pendant's panic report from starting a **cmd sat mt** timer that bars a subsequent panic report from another pendant from being sent if the mt timer has not elapsed.

> Note: **cmd sat enable 1 should be used or sat force 1 if using sat enable 2**
> Note: Panic reports from the same Pendant will still be limited by **cmd sat mt**.

In this example, the CTM is configured to send Man Down Pendant panic messages over satellite. Two pendants are used in the field.

> Note: **cmd wpan acktype** must be set to '**m**' (modem) because '**s**' (server) type acknowledgement is too slow over satellite.

```
cmd sat enable 1                    // enable satellite
cmd sat mt 2                        // set minimum time between reports from the same pendant to 2 min
cmd panicreport 1                   // use general report #1 for sending PANIC/MPANIC/FPANIC messages
cmd panicreptype 1                  // set panic report to be of binary type
cmd reptype 1 0 10                  // set general report #1 to use satellite type
cmd repaddmes 1 82                  // add message 82 ($GPRMC) to general report #1
cmd wpan stm 1                      // use internal WPAN gateway
cmd wpan adddev ffffffffffffffff    // allow all Man Down Pendants to connect to the internal WPAN gateway
cmd wpan acktype m                  // pendant acktype must be set to m
```

> Note: **cmd panicreptype 0 (default)** will set the panic report to be ASCII (e.g. `$PPEN,0013A200402CB498,17,PANIC*07`) while **cmd panicreptype 1** will set the panic report to be binary (message #93) .

The above configuration will result in a behavior similar to the following:

Time (minutes)	Pendant 1	Pendant 2	Time since last Pendant 1	Time since last Pendant 2	Pendant 1 Report	Pendant 2 Report
1	PANIC	PANIC	0*	0*	SENT	SENT
2	MPANIC	FPANIC	1	1	DISCARDED	DISCARDED
3	FPANIC	MPANIC	2*	2*	SENT	SENT

*First report or minimum time has been reached, so allowed to send

3.1.6 Effect of message ordering while using ASCII Pendant Panic reports

If messages are added to ASCII Pendant panic reports (e.g. configured via **cmd panicreport, cmd panicreptype 0** is used, and extra messages added using **cmd repaddmes**), the message order and type becomes important because added messages may or may not be grouped with the panic report. Understanding this is significant because panic reports are checked against a list of previous Man Down Pendant IDs while all other reports are checked against a list of previous report numbers for **cmd sat mt** requirement.

When reports are generated and sent out, messages added to the reports are added first because they typically provide useful starting information such as IMEI/ESN, GPS location, etc. The actual report is (e.g. $PPEN) then appended to the end of this group of added messages.

For example:

MESSAGE 1	MESSAGE 2	MESSAGE 3	MESSAGE 4	MAIN MSG/REPORT

In addition, to optimize transmitting data, consecutive ASCII messages are merged together into one message when possible. Binary messages are sent individually because they have their own ULCP header.

For example, **cmd repaddmes 1 3 82 112** adds two ASCII messages (message #3, $PMID, and #82, $GPRMC) and a binary message (message #112):

#3 (A)	#82 (A)	#112 (B)	-	MAIN MSG/REPORT

These message types will be represented as **A A B** for simplicity, where **A is ASCII** and **B is binary**. The first two ASCII messages will be merged together, effectively becoming only **A B**. ASCII message **A** (containing message #3 and #82) will be sent first, followed by binary message **B**:

#3, #82 (A)	#112 (B)	-	MAIN MSG/REPORT

Another example is **cmd repaddmes 1 3 112 80 82**, which is represented by **A B A A**, effectively becoming **A B A:**

#3 (A)	#112 (B)	#80, #82 (A)	MAIN MSG/REPORT

The result is that message #3 is sent first, followed by message #112, then messages #80 and #82 are sent together.

Assuming the main panic report appended at the end of the group of messages is represented by P, the above example is represented by **A B A P**:

#3 (A)	#112 (B)	#80, #82 (A)	MAIN MSG/REP (P)

However, because the main report is also an ASCII type, and ASCII messages are grouped together, it is further reduced down to **A B A P → A B A → A B P (changed to P because it contains a panic report),** where **P** contains the ASCII messages #80, #82 as well as the $PPEN panic report:

#3 (A)	#112 (B)	#80, #82 (A), MAIN MSG/REP (A,P)

The consequence is that the entire P message will be checked against the list of previous Man Down Pendant IDs, including message #80 and #82.

In short, the last **group** of ASCII messages will be grouped with the panic report:

cmd repaddmes A B A A

(A)	(B)	(A) (A), MAIN MSG/REP (A,P)

cmd repaddmes B A A A

(B)	(A) (A) (A), MAIN MSG/REP (A,P)

cmd repaddmes A A A B

(A) (A) (A)	(B)	MAIN MSG/REP (A,P)

In all previous examples, **P** messages (orange) are checked against the list of previous Man Down Pendant IDs while yellow and green messages are checked against the list of previous report numbers. This can be used to control whether messages are to be sent with pendant panic reports or not.

> Note: For each report number, messages generated within 5 seconds of the first message will all be allowed through to prevent messages of a group being filtered out by **cmd sat mt** being triggered by the first message.

<u>Sending ASCII messages with ASCII pendant panic messages:</u>

```
cmd sat enable 1                    // enable satellite
cmd sat mt 2                        // set minimum time between reports from the same pendant to 2 min
cmd panicreport 1                   // use general report #1 for sending PANIC/MPANIC/FPANIC messages
cmd panicreptype 0                  // set panic report to be of ascii type
cmd reptype 1 0 10                  // set general report #1 to use satellite type
cmd repaddmes 1 82                  // add message #82 ($GPRMC) to general report #1 to show pendant location
cmd wpan stm 1                      // use internal WPAN gateway
cmd wpan adddev ffffffffffffffff    // allow all Man Down Pendants to connect to the internal WPAN gateway
cmd wpan acktype m                  // pendant acktype must be set to m
```

This setting will result in the following (blue for checked against report number, green for Pendant ID):

Time (minutes)	Pendant 1	Pendant 2	Pendant 1 Panic Report	Pendant 2 Panic Report
1	PANIC	PANIC	MSG 82+$PPEN SENT	MSG 82+$PPEN SENT
2	MPANIC	FPANIC	DISCARDED	DISCARDED
3	FPANIC	MPANIC	MSG 82+$PPEN SENT	MSG 82+$PPEN SENT

Sending mix of binary and ASCII messages with ASCII pendant panic messages:

```
cmd sat enable 1                    // enable satellite
cmd sat mt 2                        // set minimum time between reports from the same pendant to 2 min
cmd panicreport 1                   // use general report #1 for sending PANIC/MPANIC/FPANIC messages
cmd panicreptype 0                  // set panic report to be of ascii type
cmd reptype 1 0 10                  // set general report #1 to use satellite type
cmd repaddmes 1 82 113              // add message #82 ($GPRMC) and #113 to show pendant location
cmd wpan stm 1                      // use internal WPAN gateway
cmd wpan adddev ffffffffffffffff    // allow all Man Down Pendants to connect to the internal WPAN gateway
cmd wpan acktype m                  // pendant acktype must be set to m
```

This setting will result in the following (blue for checked against report number, green for Pendant ID):

Time (minutes)	Pendant 1	Pendant 2	Pendant 1 Panic Report	Pendant 2 Panic Report
1	PANIC		MSG 82+113+$PPEN SENT	
2		FPANIC		MSG 82+113 DISCARDED, $PPEN SENT
3	FPANIC	MPANIC	MSG 82+113+$PPEN SENT	MSG 82+113 SENT, $PPEN DISCARDED
4				
5	PANIC	PANIC	MSG 82+113+$PPEN SENT	MSG 82+113+$PPEN SENT

In this example Pendant 1 generates a PANIC message at time=1. Because binary message 113 was injected between ASCII message 82 and the main $PPEN panic message, messages 82 and 113 are both checked based on their report number. As it is the first time they are sent, they are allowed through. The $PPEN message is also allowed through as it is the first time Pendant 1 sent a panic message.

At time=2, Pendant 2 generates a panic message. Like before, messages 82 and 113 are checked based on their report number. However, as Pendant 1 sent messages 82 and 113 less than mt minutes ago, the newly generated messages 82 and 113 by Pendant 2 will be discarded. Note that the $PPEN message is allowed through as it is the first time Pendant 2 sent a panic message.

At time=3, Pendants 1 and 2 both generate a panic message. Since messages 82 and 113 were sent at time=1, which is mt minutes ago, messages 82 and 113 are allowed to be sent again. The $PPEN message for Pendant 1 is also allowed through as Pendant 1's previous $PPEN message was mt minutes ago. Note that while the $PPEN message for Pendant 2 is discarded because Pendant 2 just sent a panic message 1 minute ago, messages 82 and 113 generated by Pendant 2 are sent because they were generated within 5 seconds of messages 82 and 113 generated by Pendant 1.

At time=4, nothing was generated.

At time=5, both Pendants sent a panic message. Messages 82 and 113 from Pendants 1 and 2 are generated within 5 seconds of each other, so they are both allowed through. $PPEN messages for Pendants 1 and 2 are also both allowed through because 2 minutes or more have elapsed since the last $PPEN message for both Pendants 1 and 2.

Sending mix of binary and ASCII messages with ASCII pendant panic messages (re-arranged):

```
cmd sat enable 1                    // enable satellite
cmd sat mt 2                        // set minimum time between reports from the same pendant to 2 min
cmd panicreport 1                   // use general report #1 for sending PANIC/MPANIC/FPANIC messages
cmd panicreptype 0                  // set panic report to be of ascii type
cmd reptype 1 0 10                  // set general report #1 to use satellite type
cmd repaddmes 1 3 113 82            // add message #3 ($PMID), #113 and #82 ($GPRMC) to show pendant location
cmd wpan stm 1                      // use internal WPAN gateway
cmd wpan adddev ffffffffffffffff    // allow all Man Down Pendants to connect to the internal WPAN gateway
cmd wpan acktype m                  // pendant acktype must be set to m
```

This setting will result in the following (blue for checked against report number, green for Pendant ID):

Time (minutes)	Pendant 1	Pendant 2	Pendant 1 Panic Report	Pendant 2 Panic Report
1	PANIC		MSG 3+113+82+$PPEN SENT	
2		FPANIC		MSG 3+113 DISCARDED MSG 82+$PPEN SENT
3	FPANIC	MPANIC	MSG 3+113+82+$PPEN SENT	MSG 3+113 SENT MSG 82+$PPEN DISCARDED
4				
5	PANIC	PANIC	MSG 3+113+82+$PPEN SENT	MSG 3+113+82+$PPEN SENT

This example is similar to the previous one, except that message 82 is last in **cmd repaddmes**, so it is now merged with $PPEN messages. The result is that it is checked with $PPEN messages against Pendant ID instead of report number. Message 3 is also checked based on report number because there is a binary message (message 113) between ASCII messages 3 and 82.

Sending mix of binary and ASCII messages with binary pendant panic messages (panicreptype 1):

If **cmd panicreptype is 1**, instead of an ASCII $PPEN message, a binary message 93 is generated. The result is that all added message are checked based on report number and only binary message 93 is checked against Pendant ID. This is because binary message are not packaged together like ASCII messages are.

```
cmd sat enable 1                    // enable satellite
cmd sat mt 2                        // set minimum time between reports from the same pendant to 2 min
cmd panicreport 1                   // use general report #1 for sending PANIC/MPANIC/FPANIC messages
cmd panicreptype 1                  // set panic report to be of binary type
cmd reptype 1 0 10                  // set general report #1 to use satellite type
cmd repaddmes 1 82 113              // add message #82 ($GPRMC) and #113 to show pendant location
cmd wpan stm 1                      // use internal WPAN gateway
cmd wpan adddev ffffffffffffffff    // allow all Man Down Pendants to connect to the internal WPAN gateway
cmd wpan acktype m                  // pendant acktype must be set to m
```

This setting will result in the following (blue for checked against report number, green for Pendant ID):

Time (minutes)	Pendant 1	Pendant 2	Pendant 1 Panic Report	Pendant 2 Panic Report
1	PANIC		MSG 82+113+93 SENT	
2		FPANIC		MSG 82+113 DISCARDED, 93 SENT
3	FPANIC	MPANIC	MSG 82+113+93 SENT	MSG 82+113 SENT, 93 DISCARDED
4				
5	PANIC	PANIC	MSG 82+113+93 SENT	MSG 82+113+93 SENT

3.2 Sending commands over satellite

It is possible to send CTM commands via satellite (see section 4.2.) However, due to the nature of satellites, expect delays in transmission and response times of a few minutes. Only **sat enable 1/2** is required as **sat mt** is ignored.

There are two types of commands that can be sent over satellite:

```
$0cmd          // commands that do not require the response to be sent back to the sender

$1cmd          // commands that requires the response to be sent back to the sender
```

Any CTM command can be appended after $0/$1 except for the following commands, which are ignored:

```
cmd event dump
cmd showconfig
cmd factory
```

3.2.1 Send commands without expecting a response

Commands without needing a response should be sent with the following format:

```
$0[command]
```

The first two characters of the message must be **$0**, followed by the CTM command. Multiple commands separated by semicolon "**;**", **<CR>** or **<LF>** can be sent in a single message.

For example, to enable CAN, save and power cycle the CTM, send either of the following command (see section 4.2 for more info on how to send messages to CTM-200 over satellite):

```
$0cmd can prot 1;cmd save;cmd pwr mode 2

$0cmd can prot 1[0x0A]cmd save[0x0D]cmd pwr mode 2
```

3.2.2 Send commands expecting a response

Commands needing a response should be sent with the following format:

```
$1[command]
```

The first two characters of the message must be **$1**, followed by the CTM command. Multiple commands separated by semicolon "**;**", **<CR>** or **<LF>** can be sent in a single message.

For example, to query the current CAN settings of a CTM, send the following command (see section 4.2 for more info on how to send messages to CTM-200 over satellite):

```
$1cmd can show
```

The CTM's CAN settings should arrive via email or direct IP within a few minutes.

> Note: Because the response is greater than 340, it was split over 2 emails/attachments). Also notice how configured reports are also sent in the same payload (second attachment) for efficiency.

```
cmd can enghours 0
cmd can engparam FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE
cmd can odometer 0
cmd can prot 0
cmd can psthruid 1 00000000
cmd can psthruid 2 00000000
cmd can psthruid 3 00000000
cmd can psthruid 4 00000000
cmd can psthruid 5 00000000
cmd can psthruid 6 00000000
cmd can psthruid 7 00000000
cmd can psthruid 8 00000000
c
```

```
md can psthrumin 1 1
cmd can psthrumin 2 1
cmd can psthrumin 3 1
cmd can psthrumin 4 1
cmd can psthrumin 5 1
cmd can psthrumin 6 1
cmd can psthrumin 7 1
cmd can psthrumin 8 1
cmd can qryrate 2
$PMID,356215040410106,356215040410106*10
$GPGGA,175900.00,4915.4096,N,12259.7949,W,1,12,00.7,+00028,M,,M,,0000*6D
```

3.3 Transferring raw data or using PAD over satellite

It is possible to send raw data transferred over satellite with the CTM to a destination TCP server. As such, it is also possible to use PAD with satellite. Please note that **cmd sat mt** is ignored.

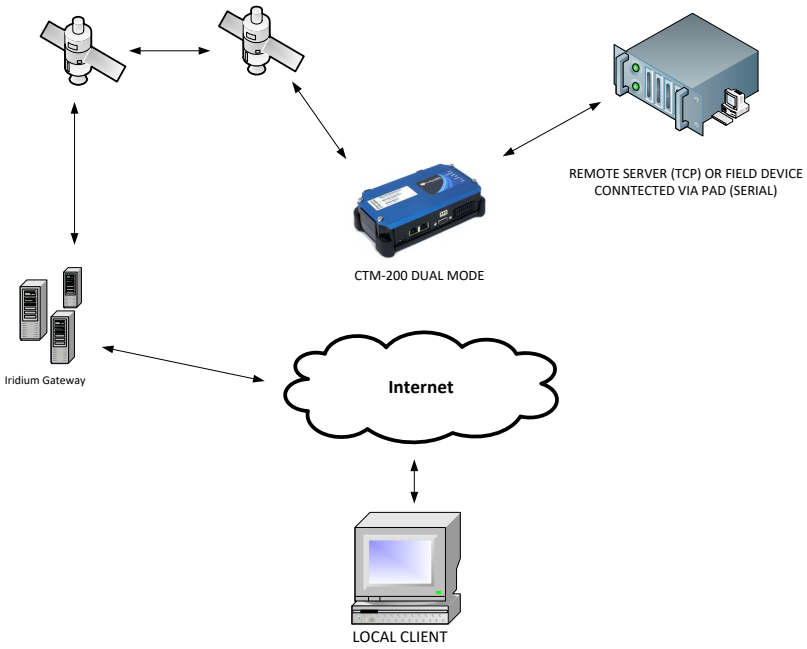
To send data transferred over satellite, use the following configuration:

```
cmd sat enable 1                  // enable satellite
cmd sat pad 1                     // enable raw/PAD data connection
cmd sat ip 123.123.123.123        // remote IP to forward data received over satellite to
cmd sat port 12345                // remote port to forward data received over satellite to
```

If using CTM's PAD server, set sat ip to local IP (not 127.0.0.1), sat port to pad port, and PAD TCP server:

```
cmd sat enable 1                  // enable satellite
cmd sat pad 1                     // enable raw/PAD data connection
cmd sat ip 192.168.1.1            // local IP to send to PAD ps
cmd sat port 12345                // must be same as cmd pad port
cmd pad port 12345
cmd pad mode 1                    // configure CTM as a PAD TCP server
cmd pad fwdl 340                  // set it to forward data after 340 bytes have been reached as that is maximum
                                  payload per packet for satellite
```

The above configuration will allow devices to communicate with each other over a mix of serial/IP and satellite. For example, a device connected to CTM via serial or IP talking to a remote device connected to an Iridium Gateway.


REMOTE SERVER (TCP) OR FIELD DEVICE
CONNTECTED VIA PAD (SERIAL)
CTM-200 DUAL MODE
Iridium Gateway
Internet
LOCAL CLIENT

4. Satellite Commands

4.1 Command: sat enable m

```
sat enable [m]
```

m	Description
0	Satellite **DISABLED** and **POWERED OFF** (**DEFAULT**)
1	Satellite **ENABLED** and **POWERED ON**
2	Dynamic mode (used with `sat pwr`, default state is **DISABLED/POWERED OFF**)

For any features to use satellite, it must be enabled with m=1 or m=2 (dynamic mode). In dynamic mode, the satellite ISU is powered OFF until the CTM detects that it can no longer report via regular channels (e.g. cell/wifi), at which point the ISU is turned ON. It is also controlled by the command **cmd sat pwr** in dynamic mode.

4.2 Command: sat mt t

```
sat mt [t]
```

t	Description
0 to 65535	Minimum time in minutes (**DEFAULT 10**)

Sets the minimum time in minutes between reports sent via satellite for **identical report numbers** and **identical Man Down Pendant IDs.** This command is used to throttle the amount of data being sent over satellite. It affects **reports** only (i.e. commands and PAD are unaffected.)

> **Note: If for some reason the CTM cannot connect to the satellite, outbound messages will be queued and sent once connection has been established. If enough messages have been queued, there is a potential "burst" of data at that time.**

For example, if general report 1 is sent via satellite at time **t**, then the next time another report 1 is allowed to be sent via satellite is after time **t+mt**. Any report 1 generated between that time is discarded.

Since Man Down Pendants share the same report number configured in **cmd panicreport**, Man Down Pendants are tracked based on their IDs.

For example, if pendant 1 sends a panic message (PANIC/FPANIC/MPANIC) at time **t**, any other pendant with a different ID (e.g. pendant 2) can send a panic message **within t+mt** time, while the next panic message from pendant 1 allowed through **is after t+mt.** Panic messages generated before **mt** time has elapsed are discarded.

4.3 Command: sat pwr m

```
sat pwr [m]
```

m	Description
0	Satellite **DISABLED** and **POWERED OFF** (**DEFAULT**)
1	Satellite **ENABLED** and **POWERED ON**

When the satellite ISU is configured for dynamic mode (**cmd sat enable 2**), it can also be manually enabled and disabled using this command for on demand reporting. This command is volatile, so the state is not saved with **cmd save**.

> Note: **ON** states have priority, meaning **cmd sat pwr** cannot override decisions made by CTM-200 in failover mode and turn the satellite module OFF, and vice versa (failover mode cannot power off the satellite module if **cmd sat pwr 1**.)

4.4 Command: sat force n

```
sat force [n]
```

n	Description
0	No message type will force satellite ISU to power ON
1	Pendant's PANIC message will force ISU to power ON (**DEFAULT**)

Bitmapped value selecting what type of message will force satellite ISU to power ON. Used with **cmd sat enable 2**. Pendant PANIC message is **ON by default** to ensure Pendant PANIC messages are sent out.

4.5 Command: sat ip iii.iii.iii.iii

```
sat ip [iii.iii.iii.iii]
```

iii.iii.iii.iii	Description
0.0.0.0 to 255.255.255.255	Remote IP address in decimal (**DEFAULT 192.168.1.1**)

Transfers raw data (e.g. PAD) received over satellite to this IP. Also connects to this IP to receive messages to send over satellite. Uses TCP. **cmd sat pad must be set to 1**.

> Note: If using with CTM's PAD, users must configure CTM's PAD server as TCP PAD server (**cmd pad mode 1**) and must use a local Ethernet IP (e.g. 192.168.1.1) and not 127.0.0.1.

4.6 Command: sat port n

```
sat port [n]
```

n	Description
0 to 65535	Remote port address in decimal (**DEFAULT 23002**)

Used with command **cmd sat ip**, which sets the server's port, and **cmd sat pad**, to enable the function.

Note: If using CTM's PAD, users must set **cmd sat port** equal to **cmd pad port**.

4.7 Command: sat pad n

```
sat pad [n]
```

n	Description
0	Do not connect to server (**DEFAULT**)
1	Connect to server at cmd sat ip and cmd sat port

Enables or disables connecting to the configured destination server (e.g. raw data server or PAD server) via **cmd sat ip** and **cmd sat port**.

4.8 Command: sat stats

```
sat stats
```

Shows the current transmission count in bytes, signal quality index (SQI), registration and IMEI. The SQI has a range of 0 to 5, and each increment represents approximately 2dB improvement in link margin over the previous value. SQI=0 is at or below the minimum receiver sensitivity level, 1 is 2dB of link margin, and 5 is 10dB or more link margin. Please note that these are not calibrated output and only provides relative information about the receive signal, and not an absolute receiver level or a calibrated signal to noise ratio.

4.9 Command: sat flush

```
sat flush
```

Clears messages queued for transmission over satellite. This requires **cmd ctm15x** to be executed first.

4.10 Command: sat reset

```
sat reset
```

Resets the satellite transmission byte count. This requires **cmd ctm15x** to be executed first.

Technical Support

Cypress Solutions Service Support Group
1.844.462.9773 or 778.372.4603
9.00am to 5.00pm PST
support@cypress.bc.ca