ITT NexGen
44965 Aviation Dr., Suite 400
Dulles, VA 20166

SBD Subscriber Device Interface Control Document

For use with the Enhanced Mobile Satellite Services Department of Defense Gateway

Version 1.3

20 May 2010

Revision History

Revision Number	Release Date	Principal Contributors	Description
1.0	**15 Nov 2004**	**SBD Design Group**	**Initial Post-beta Release**
1.1	**5 Apr 2005**	**SBD Design Group**	**Update With SBD Software Release 4.0H**
1.2	**11 Jul 2006**	**SBD Design Group**	**Updated With SBD Software Release 4.2H**
1.3	**20 May 2010**	**SBD Design Group**	**Updated With SBD Software Release 5.2H**

Table of Contents

1 Introduction 1
1.1 Purpose 1
1.2 Audience 1
1.3 Scope 1
1.4 References 1
1.5 Definitions 1
1.6 Acronyms 1
2 Overview 3
2.1 Service Description 3
2.2 Service Architecture 3
3 Vendor Application to GSS Interface 6
3.1 Mobile Originated Messages 6
3.1.1 MO Messages – E-mail Delivery 8
3.1.1.1 MO E-mail Delivery Option: Standard E-mail 8
3.1.1.2 MO E-mail Delivery Option: Inline Binary 9
3.1.1.3 MO E-mail Delivery Option: Inline Text 9
3.1.2 MO Messages – DirectIP Delivery 10
3.1.2.1 Legacy DirectIP Protocol 10
3.1.2.2 Updated DirectIP Protocol 10
3.1.2.3 Vendor Application Server Unavailable 11
3.1.3 MO Messages – SSD-to-SSD Delivery 11
3.2 Mobile Terminated Messages 11
3.2.1 MT Message Queuing 12
3.2.1.1 MT Messages – E-mail Reception 12
3.2.1.2 MT Messages – DirectIP 15
3.2.1.3 MT Messages – SSD-to-SSD 15
3.2.1.4 MT Messages – Web Page 16
3.2.2 MT Ring Alerts 16
3.2.3 MT Delivery Short Codes 16
3.2.3.1 Hold MT Message Delivery 16
3.2.3.2 Leave MT Message in Queue After Delivery 16
4 DTE field application to SSD Interface 17
4.1 RS-232 Interface 17
4.2 AT Commands 17
4.2.1 +SBDAREG – Automatic Registration 17
4.2.2 +SBDC – Clear SBD MOMSN 18
4.2.3 +SBDD – Clear SBD Message Buffer(s) 19
4.2.4 +SBDDET – Detach 19
4.2.5 +SBDDSC – Set Delivery Short Code 20
4.2.6 +SBDGW – Gateway 20
4.2.7 +SBDI – Initiate an SBD Session 21
4.2.8 +SBDIX[A] – Initiate an SBD Session Extended 22
4.2.9 +SBDMTA – Mobile-Terminated Alert 24
4.2.10 +SBDRB – Read Binary Data from the SSD 25
4.2.11 +SBDREG – Network Registration 25
4.2.12 +SBDRT – Read a Text Message from the SSD 27
4.2.13 +SBDS – SBD Status 27
4.2.14 +SBDTC – Transfer Send Buffer to Receive Buffer 28
4.2.15 +SBDWB – Write Binary Data to the SSD 28

4.2.16 +SBDWT – Write a text message to the SSD 29

5 SBD Scenarios 30
5.1 Mobile Originated Message 30
5.2 Mailbox Check / Mobile Terminated (MT) Message 31
5.3 MO and MT Message 32

6 Support 35

List of Figures

Figure 2-1 SBD Service Architecture 3
Figure 2-2 SBD Data Flow 4
Figure 3-1 MO E-mail Message 9
Figure 3-2 MO Inline Binary 9
Figure 3-3 MO Inline Text 10
Figure 3-5 MT E-mail Message 13
Figure 3-6 MT E-mail Reply Message: Success Indication 13
Figure 3-7 MT E-mail Reply Message: Error Indication 14
Figure 5-1 Vendor Application to GSS Interface, MO Message 31
Figure 5-2 Vendor Application to GSS Interface, Mailbox Check / MT Message 31
Figure 5-3 Vendor Application to GSS Interface, Vendor Application Generated E-mail 32
Figure 5-4 Vendor Application to GSS Interface, MT Message 33
Figure 5-5 Vendor Application to GSS Interface, MO Message 34

List of Tables

Table 3-1 MO Message Delivery Methods 6
Table 3-2 SBD Session Descriptors 6
Table 3-3 Session Status Values 8
Table 3-4 VA Received E-mail Message Fields 8
Table 3-5 Legacy DirectIP Data Structure 11
Table 3-6 MT Message Queuing Methods 12
Table 3-7 MT Message Queuing Indication Subject Field Descriptions 14
Table 3-8 MT Message Queuing Acknowledgement Body Field Descriptions 15
Table 4-1 RS-232 Port Settings 17
Table 5-1 DTE Field Application to SSD Interface, MO Message 30
Table 5-2 DTE Field Application to SSD Interface, Mailbox Check / MT Message 31
Table 5-3 DTE Field Application to SSD Interface, Send and Receive 33

1 Introduction

1.1 Purpose

The purpose of this document is to describe the protocol for communication between the DTE field application, the SBD subscriber device, and the vendor application.

1.2 Audience

This document is primarily intended for use by the engineers who are interfacing to the SBD service through the Department of Defense EMSS gateway.

1.3 Scope

This document provides an overview of the SBD system and the interfaces by which the SBD service is accessed and used. Detailed definitions of the AT commands and responses used in the DTE to SSD interface are presented along with message queuing and delivery methods via the NIPRnet/Internet.

1.4 References

[1] EMSS SBD DirectIP Specification, version 2.0
[2] Sebring Interface Control Document
[3] ISU AT Command Reference
[4] 9601 SBD Transceiver Preliminary Product Developers Guide

1.5 Definitions

Forward	The process of passing a message to the end recipient.
Receive	The process of an end device taking delivery of a message. Only the SSD and the vendor application can receive a message. The GSS never receives a message. It only stores it.
Receive Buffer	This is the SSD buffer in which an SBD message sent from the GSS to the SSD will be stored.
Send	The process of creating a message to be transferred to another device. Only the SSD and the vendor application can send a message. The GSS never sends a message, it only forwards a message.
Send Buffer	This is the SSD buffer in which an SBD message to be sent from the SSD to the GSS will be stored.
Store	The process of holding a message until the message can be forwarded.

1.6 Acronyms

API	Application Programming Interface
DSC	Delivery Short Code
DTE	Data Terminal Equipment
FA	Field Application
GSM	Global System for Mobile Communication

GSS	Gateway SBD Subsystem
IMEI	International Mobile Equipment Identifier
ISU	Iridium Subscriber Unit
LBT	L-Band Transceiver
MIME	Multipurpose Internet Mail Extensions
MO	Mobile Originated
MOMSN	Mobile Originated Message Sequence Number
MSN	Message Sequence Number
MT	Mobile Terminated
MTMSN	Mobile Terminated Message Sequence Number
PMVN	Parameter Master Version Number
SBD	Short Burst Data
SEP	Short Burst Data (SBD) – ETC Processor
SIM	Subscriber Information Module
SMS	Short Messaging Service (GSM)
SPP	Short Burst Data (SBD) – Post Processor
SSD	SBD Subscriber Device
SV	Satellite Vehicle
UTC	Coordinated Universal Time
VA	Vendor Application

2 Overview

2.1 Service Description

Short Burst Data is a simple and efficient bi-directional transport capability used to transfer messages with sizes ranging from zero (a mailbox check) to slightly less than 2 kilobytes. Messages that originate from field equipment can be delivered to a variety of destinations. Commonly, data is delivered across terrestrial communications networks (NIPRnet/Internet) to servers and applications that process data from one or multiple SBD subscriber devices (SSDs) in the field. The SBD service also supports the transfer of messages to SSDs, where messages may originate from either terrestrial sources or other SSDs in the field.

Delivery methods and options are initially configured when the SSD unit is first provisioned, but may be easily modified via the EMSS SBD Web Pages (https://sbd.pac.disa.mil) or by e-mailing requests to support@sbd.pac.disa.mil. Configuration requests must be sent by authorized government personnel.

2.2 Service Architecture

Figure 2-1 provides a simplified view of the SBD service architecture and its major interfaces. SSDs in the field communicate directly with the Gateway SBD Subsystem (GSS) and are able to exchange data with other SSDs in the field or with server applications that connect with the GSS through the gateway's terrestrial interfaces (NIPRnet/Internet).

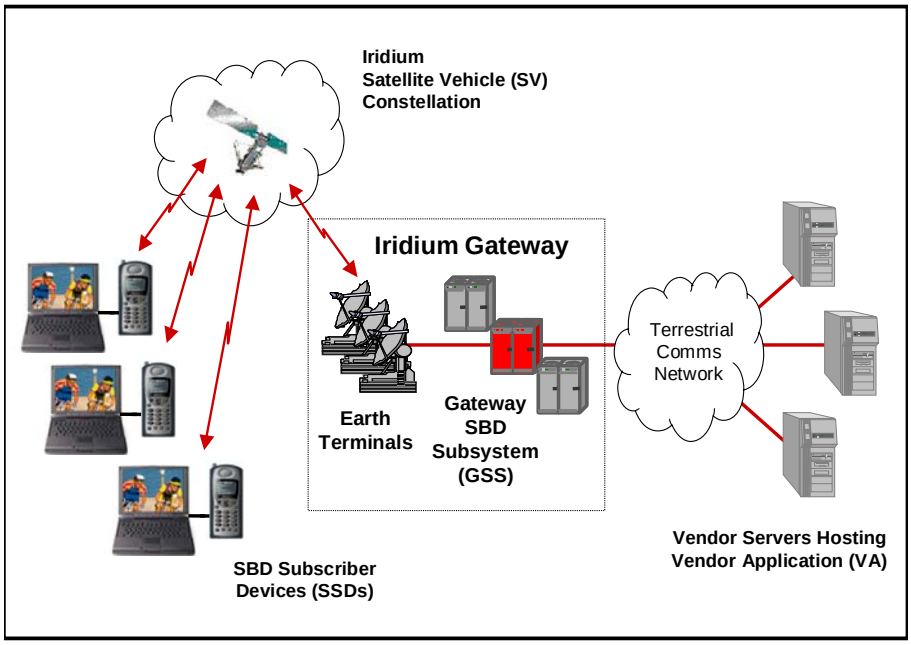

Figure 2-1 SBD Service Architecture

Figure 2-2 illustrates the data flow between the primary components of the SBD architecture. These components are the DTE field application, the SSD, the Iridium satellite network, the GSS, the terrestrial communications network, and the vendor application. The DTE field application may be any remote field equipment connection and SSD that needs to send messages to and/or receive messages

from a centralized information recipient/source or another SSD. The SSD is a provisioned Iridium 9522[A][B] LBT, 9505[A] or 9555 ISU, or 9601 or derivative SBD-only device with the SBD feature activated.[1] The GSS is responsible for storing and forwarding messages from the SSD to the vendor application and storing messages from the vendor application to forward to the SSD.

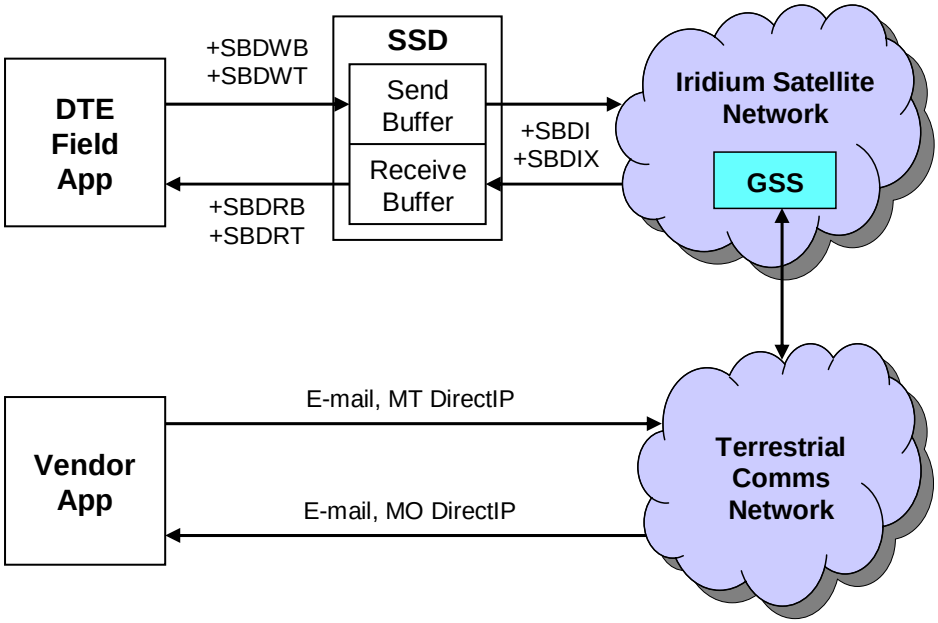

Figure 2-2 SBD Data Flow

Mobile-originated (MO) messages received at the GSS from an SSD to be sent to the vendor application may also be delivered via standard e-mail (SMTP) or the DirectIP protocol (direct TCP/IP socket connection). The delivery options for each SSD may be configured independent of any other SSD, and may be reconfigured as needed via the SBD web pages.

Mobile-terminated (MT) messages from the vendor application to the GSS may be sent using standard e-mail (SMTP) or DirectIP. These messages will be stored at the GSS until the SSD initiates an SBD session to receive the message. An SSD may be configured to receive a ring alert [2] when an MT message is received to prompt SBD session initiation.

The interface between the DTE field applications and the SSD is an AT command driven interface which is used to load and retrieve messages between the SSD and the DTE device. As depicted in Figure 2-2, a message is loaded into the SSD (+SBDWB or +SBDWT), a message transfer session between the SSD and the GSS is initiated (+SBDI, +SBDIX or +SBDIXA), and, finally, if a message was received at the SSD from the GSS, it will be retrieved by the DTE field application (+SBDRB or +SBDRT). Message transfers between the SSD and the GSS use a reliable transport mechanism that

[1] To be SBD-ready, 9522[A] SSDs must have software revision SAC0307 or greater. 9505[A] SSDs must have software revision LAC0307 or greater. All 9601 software revisions are SBD-ready. The SBD service must also be activated for the equipment via the SBD provisioning process.

[2] To be able to process ring alerts, 9522A and 9505A SSDs must have software revision IS06002 or greater. All software revisions for the 9601 and its derivatives are able to process ring alerts. Older 9522 and 9505 SSDs are not able to process ring alerts.

ensures the message is delivered without error. If the SSD was not able to send or receive messages, an indication is passed back to the DTE field application.

The send and receive buffers in the SSD will maintain messages as long as the SSD is powered on. Once a message is transferred from the DTE field application to the send buffer in the SSD, it will remain there even after it is successfully sent to the GSS. If a message is received at the SSD from the GSS, it will remain in the receive buffer even after the DTE field application reads it. The buffers in the SSD will be cleared only when given an explicit command (+SBDD) or when the SSD is power cycled.

3 Vendor Application to GSS Interface

3.1 Mobile Originated Messages

Messages sent from an SSD to the GSS are stored and then forwarded immediately to the destination(s) that have been configured for that SSD. There are a variety of delivery methods, and each has additional configurable options.

There are two methods of delivery from the GSS to a vendor application. MO messages can be transferred via e-mail (SMTP), or the TCP/IP socket-based DirectIP protocol. Details of these delivery methods are provided in the sections below. MO messages may also be delivered directly to the MT delivery queue of another SSD. If this option is selected, the message is never delivered to a vendor application. The various delivery methods and brief overviews are provided in Table 3-1.

Each message sent from the SSD to the GSS generates SBD session descriptors that may also be included in the message delivery. SBD session descriptors are described in Table 3-2. Included in the descriptors is the session status. The possible session status values are detailed in Table 3-3.

Table 3-1 MO Message Delivery Methods

Delivery Method	Overview	Session Descriptors
E-mail	Delivery via standard e-mail. Latency is indeterminate due to traversing the NIPRnet/Internet.	Yes
MO DirectIP	Delivery via TCP/IP sockets. Requires custom DirectIP server application. Latency is near zero.	Yes
SSD-to-SSD	Placed in MT queue for one or multiple SSDs. No vendor application is required.	No

Table 3-2 SBD Session Descriptors

Field Name	Description
MOMSN	This field is the MO message sequence number set by the SSD when the message was sent to the GSS. This MOMSN is a wrap-around integer value in the range of 0 to 65,535 and is incremented by the SSD each time an SBD session is completed successfully.
MTMSN	This field is the MT message sequence number assigned by the GSS and forwarded from the GSS to the SSD. This field will appear if an MT message transfer was attempted regardless of the success of the session. This value is unique per SSD, is a wrap-around integer value in the range of 1 to 65,535, and is incremented by the GSS each time it queues a message to be forwarded to the particular SSD. This field will be 0 (zero) if no MT transfer attempt occurred. If the session failed (e.g. incomplete transfer) where an MT message was attempted to be delivered, the MT message will remain queued for delivery.

<table>
<tr><th>Field Name</th><th>Description</th></tr>
<tr><td>Time of Session</td><td>This field provides a UTC timestamp of the SSD session between the SSD and the GSS. It is sent in two different formats, depending upon the delivery method. For DirectIP this field is transferred as the number of seconds since January 1, 1970 (epoch time) in a four byte integer compatible with UNIX ctime. For e-mail, a text string is sent with the following format: “DDD MMM dd HH:MM:SS yyyy”.
<table>
<tr><th>Field Value</th><th>Description</th></tr>
<tr><td>DDD</td><td>Day of the week (Sun, Mon, Tue, Wed, Thu, Fri, Sat)</td></tr>
<tr><td>MMM</td><td>Month of the year (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec)</td></tr>
<tr><td>dd</td><td>Day of the month (01,...,31)</td></tr>
<tr><td>hh</td><td>Hour (00,...,23)</td></tr>
<tr><td>mm</td><td>Minute (00,...,59)</td></tr>
<tr><td>ss</td><td>Second (00,...,59)</td></tr>
<tr><td>yyyy</td><td>Year</td></tr>
</table></td></tr>
<tr><td>Session Status</td><td>This field provides an indication of success of the SBD session between the SSD and the GSS. It is sent in two different formats, depending upon the delivery method. For DirectIP, the numeric values are used. For e-mail, text values are used. The numeric and text values are shown in Table 3-3.</td></tr>
<tr><td>Message Size</td><td>This field provides an indication of the size (in bytes) of the attached message in decoded form. Note: this is not the size of the MIME encoded data (e-mail).</td></tr>
<tr><td>Unit Latitude (Optional)</td><td>Inclusion of this optional field is configured when the SSD is provisioned. It provides the geographic latitude of the SSD measured in degrees. Positive represents north, negative represents south.[3] When opted to not be included, this value is 0.0 (zero).</td></tr>
<tr><td>Unit Longitude (Optional)</td><td>Inclusion of this optional field is configured when the SSD is provisioned. It provides the geographic longitude of the SSD measured in degrees. Positive represents east, negative represents west.[3] When opted to not be included, this value is 0.0 (zero).</td></tr>
<tr><td>CEP Radius (Optional)</td><td>Inclusion of this optional field is configured when the SSD is provisioned. It provides an estimate of the accuracy of the unit location.[3] When opted to not be included, this value is reported as 0 (zero).</td></tr>
</table>

[3] The latitude and longitude provide a center point, and the CEP radius provides the radius around the center point within which the unit is located. This reported position is accurate 80% of the time.

Table 3-3 Session Status Values

Numeric (Hex)	Text	Description
0x00	00 – Transfer OK	The SBD session completed successfully.
0x01	01 – Transfer OK MT Message Too Large	The MO message transfer, if any, was successful. The MT message queued at the GSS is too large to be transferred within a single SBD session.
0x02	02 – Transfer OK Bad Location	The MO message transfer, if any, was successful. The reported location was determined to be of unacceptable quality. This value is only applicable to SSDs using SBD protocol revision 1, when ring alerts were introduced, or greater. [2]
0x0A	10 – SBD Timeout	The SBD session timed out before session completion.
0x0C	12 – MO Message Too Large	The MO message being transferred by the SSD is too large to be transferred within a single SBD session.
0x0D	13 – Incomplete Transfer	An RF link loss occurred during the SBD session.
0x0E	14 – SBD Protocol Error	An SSD protocol anomaly occurred during SBD session.
0x0F	15 – SBD Denial	The SSD is prohibited from accessing the GSS.

3.1.1 MO Messages – E-mail Delivery

The e-mail delivery method utilizes standard e-mail (SMTP) to deliver the message and SBD session descriptors. The e-mail delivery method has several packaging options that provide flexibility for the vendor application. The packing options include Standard, Inline Binary, and Inline Text. These options are described in detail below. Details of the e-mail message fields are provided in Table 3-4.

Table 3-4 VA Received E-mail Message Fields

Field Name	Description
From	This field typically identifies the sender of the e-mail message. In the case of messages sent to the vendor application, this field is filled with the string service@sbd.pac.disa.mil.
Sent	This field provides the time at which the message was e-mailed from the GSS to the vendor application. The timestamp is in UTC.
Subject	This field provides the IMEI of the SSD that sent the message.

3.1.1.1 MO E-mail Delivery Option: Standard E-mail

Standard E-mail delivery places the SBD session descriptors in the message body and includes the MO SBD message as an attachment to the e-mail. The SBD message is encoded using standard MIME Base64 encoding as defined in RFC 2045. The format of the e-mail message is shown in Figure 3-1 and details of the e-mail message fields are provided in Table 3-4.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 16:51 PM
Subject: SBD Msg From Unit: 304050607080903

MOMSN: 2

Time of Session (UTC): Tue Aug 13 16:51:04 2002
Session Status: 00 - Transfer OK
Message Size (bytes): 1230

Unit Location: Lat = 59.372463 Long = 75.309806
CEPradius = 3

Message is Attached.
```

Figure 3-1 MO E-mail Message

3.1.1.2 MO E-mail Delivery Option: Inline Binary

Inline Binary delivery places the MIME Base 64 encoded message into the body of the e-mail. This can be used to deliver data through firewalls that do not allow attachments, and still provide for the transfer of binary data. The format of the e-mail message is shown in Figure 3-2.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 16:51 PM
Subject: SBD Msg From Unit: 304050607080903

MOMSN: 2
MTMSN: 239
Time of Session (UTC): Tue Aug 13 16:51:04 2002
Session Status: 00 - Transfer OK
Message Size (bytes): 1230

Unit Location: Lat = 59.372463 Long = 75.309806
CEPradius = 3
¶
¶
¶
¶
Avjoiu^dfijsdoi^vsdfksdl#lkjsdflkjsdflkjsdljkfdsl;kfas
Dsfdkhskd*hfjkldsjkfdsjkf)>lfdhlfdslfdjkhlfsdaljhfjhds
Jsdfhjkds%jds!dsfjdsaffdsajkfdfdkl:dskfjdshfjkld”]ljkl
```

Figure 3-2 MO Inline Binary

3.1.1.3 MO E-mail Delivery Option: Inline Text

Inline Text delivery places the message payload into the body of the message. In this case, only ASCII printable characters are allowed. All non-ASCII printable characters will be represented as a “^” (0x5E). One application of this method is to send message to pagers. No SBD session descriptors are provided with this packing option. The format of the e-mail message is shown in Figure 3-3.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 16:51 PM
Subject: SBD Msg From Unit: 304050607080903

This is ASCII: abcdefg01234567ABCDEFG.
This is binary: ^^^^^^^^^^^^^^^^^^^^
```

Figure 3-3 MO Inline Text

3.1.2 MO Messages – DirectIP Delivery

DirectIP transfers utilize the TCP/IP socket-based protocol to quickly and efficiently transfer the message payload and SBD session descriptors directly to a vendor application. The vendor application must include a DirectIP server process that interacts with the GSS client to receive the SBD data transfer. The IP address where the server is running is entered when the SSD is provisioned. **All DirectIP socket connections are opened using port 10800.**

Upon completion of an SBD session between the SSD and the GSS, the GSS opens a socket, connects to the vendor application, and delivers the MO message and SBD session descriptors. Messages to the same vendor application are delivered in a first-in-first-out (FIFO) manner so that they are delivered in the same sequence that they are received. All other messages destined for the same vendor application will be queued behind the first message while it is being delivered. Only one message is delivered per connection.

As of the release of this ICD document, there are two protocols, or structure formats, for MO DirectIP delivery, a "legacy" protocol and an updated, more flexible protocol. The legacy protocol is described below in this document. The new protocol is described in detail in the *EMSS SBD DirectIP Specification 1.0* found on the SBD web site.

Note that support for the legacy protocol will terminate on 1 March 2007. Plans should be made to migrate to the new protocol as soon as possible. Any questions or concerns should be forwarded to SBD technical support at support@sbd.pac.disa.mil.

3.1.2.1 Legacy DirectIP Protocol

The legacy structure for the DirectIP data transfer is shown in Table 3-5. The data is transferred in network order (big endian) and standard data sizes are used. Standard sizes are 1 byte for chars, 4 bytes for integers, and 8 bytes for doubles. The structure places the double values on an 8-byte boundary, which is compatible with structure packing by most languages running under most operating systems.

3.1.2.2 Updated DirectIP Protocol

The new structure for DirectIP data transfers is described in the *EMSS SBD DirectIP Specification 1.0* available at the SBD web site. This new protocol provides a much more robust and flexible format and is more platform independent. The protocol has also been extended to be used for receiving MT messages at the GSS from MT DirectIP vendor client applications. This is also described in detail in the new specification.

Table 3-5 Legacy DirectIP Data Structure

Field Name	Data Type	Length (bytes)	Total Length (bytes)
Auto ID	Unsigned Integer	4	4
IMEI	Char	16	20
Session Status	Unsigned Integer	4	24
MOMSN	Unsigned Integer	4	28
MTMSN	Unsigned Integer	4	32
Unit Location Latitude	Double	8	40
Unit Location Longitude	Double	8	48
CEP Radius	Unsigned Integer	4	52
Time of Session	Unsigned Integer	4	56
Message Payload Size	Unsigned Integer	4	60
Message Payload	Char	0 - 1960	60-2020

3.1.2.3 Vendor Application Server Unavailable

If the initial attempt to connect to the vendor application times out, the subsequent message delivery will not take place and subsequent connection attempts will be made. The timeout values for the first, second and third connection attempts are 5, 15 and 45 seconds, respectively. After the third attempt, connection attempts continue to be made repeating the three timeout values for up to 12 hours. Each individual message has a lifetime of 12 hours starting at the time that the message was received at the GSS. If it is not able to be delivered within this lifetime, it will be marked as “DirectIP_Timeout” and removed from the delivery queue.

Up to 1000 messages may be queued for a specific vendor application. If this limit is exceeded, messages will be deleted from the front of the queue (the “oldest” messages) and will be marked as "DirectIP_QueueOverflow”.

Note that while a message is queued and waiting to be delivered, its status is unavailable via the SBD web pages. Only those messages that are delivered or undeliverable (e.g. timed out) are visible via web queries.

3.1.3 MO Messages – SSD-to-SSD Delivery

SSD-to-SSD allows an SSD to deliver a message directly to the MT delivery queue of another SSD. When this option is used, the MO message is not delivered to a vendor application. Details regarding the MT delivery of the message are provided in section 3.2.1.3. Up to five destination SSDs may be provisioned.

3.2 Mobile Terminated Messages

SBD sessions between the SSD and GSS are mobile originated bi-directional transfers. The SSD must initiate a session to retrieve any MT messages stored at the GSS. This is true even if the session is initiated by the SSD in response to an SBD ring alert (ring alerts are described in section 3.2.1.4). The SSD may deliver an MO message as part of the session, or it may perform a mailbox check and optionally retrieve an MT message. One MT message may be delivered to the SSD during each SBD session.

3.2.1 MT Message Queuing

In order to send an MT message from the vendor application to the DTE field application, the vendor application must first deposit the message at the GSS where it will be queued for delivery to the SSD. The message will remain in the queue awaiting contact from the SSD to retrieve it. There is no limit to the time that a message may remain queued. The MT queue may be cleared via the EMSS SBD web site. If a session fails (e.g. incomplete transfer) where an MT message was attempted to be delivered, the MT message remains queued for delivery in a subsequent session.

The MT message may be deposited at the GSS via four different mechanisms including e-mail, MT DirectIP, web pages, or an MO SBD message from an SSD (SSD-to-SSD). Details of each method are outlined below. All queuing methods are constrained by the following limits:

- Maximum MT message size of 1890 bytes
- Maximum MT message size of 135 bytes for off-the-shelf 9601 SSDs
- Maximum of 50 MT messages queued for each SSD

The various queuing methods and brief overviews are provided in Table 3-6.

Table 3-6 MT Message Queuing Methods

Delivery Method	Overview
E-mail	Delivery to GSS via standard e-mail. Queuing latency is indeterminate due to traversing the NIPRnet/Internet.
MT DirectIP	Delivery to GSS via DirectIP protocol described in the *EMSS SBD DirectIP Specification* available at the SBD web site. Queuing latency is near zero.
SSD-to-SSD	Delivery to GSS via receiving and storing SBD MO messages. Received messages are immediately placed in the MT delivery queue for the specified SSD. Queuing latency is near zero.
Web Pages	Delivery to GSS via a standard web interface that allows messages to be placed directly into the SSD delivery queue. Queuing latency is near zero. Supports ASCII text only.

3.2.1.1 MT Messages – E-mail Reception

Figure 3-4 provides an example MT e-mail message. The rules for formatting the e-mail are outlined below:

- MT messages to all SSD units homed to the EMSS gateway are sent to data@sbd.pac.disa.mil.
- Placing one or more SSD IMEI(s) into the subject line of the e-mail identifies the destination SSD unit(s). If there is more than one destination SSD, simply list the additional IMEIs on the subject line with a space or a comma between them. No other delimiters besides a space or a comma are accepted.
- If the message does not contain a properly formatted sender (From address), the message will be dropped by the GSS.
- The message to the SSD must be included as an attachment to the e-mail.

  - The attachment name must have a ".sbd" filename extension (e.g. myMessageData.sbd). File names are not case-sensitive.
  - File names must be no longer than 80 characters including the ".sbd" extension.
  - The attachment must be MIME encoded using methods as defined in RFC 2045.
  - Multiple messages may be queued by a single e-mail by including the messages in separate attachments. Note that if one of the attachments has an incorrect extension while others are correct (.sbd), no error indication e-mail will be sent.

- The message body plays no role in the message transfer process. Any information contained in the body will be discarded.

```
To: data@sbd.pac.disa.mil
From: VA@VendorDomain.com
Subject: 304050607080903

<Message Attached>
```

Figure 3-4 MT E-mail Message

The GSS will validate the message upon receipt and return a "disposition indication" e-mail to the originator of the message. A sample success indication is provided in Figure 3-5, and a sample error indication e-mail is provided in Figure 3-6. If there is more than one destination SSD, a disposition indication e-mail will be returned for each.

```
To: SBDProcessor@nexgencom.net
From: service@sbd.pac.disa.mil
Subject: Success: SBD Mobile-Terminated Message Queued for Unit:
300001001247240

The following mobile-terminated message was queued for delivery:

IMEI: 300001001247240
Time: Mon Oct 27 17:24:29 2003
Attachment Filename: TestFile518chars.sbd
Attachment Size: 518 bytes
The MTMSN is 6870, and the message is number 12 in the queue.
```

Figure 3-5 MT E-mail Reply Message: Success Indication

```
To: SBDProcessor@nexgencom.net
From: service@sbd.pac.disa.mil
Subject: Error: SBD Mobile-Terminated Message Not Queued for Unit:
300001001247240

The following mobile-terminated message was not queued for delivery:

IMEI: 300001001247240
Time: Mon Oct 27 17:23:30 2003
Attachment Filename: TestFile518chars.sbd
Attachment Size: 518 bytes

Reason: Mobile-termination message queue for the IMEI is full (max of 50
allowed).
```

Figure 3-6 MT E-mail Reply Message: Error Indication

Details of the individual fields in the disposition indication e-mail subject line are given in Table 3-7. The format of the subject line is as follows:

`<Queuing Disposition>: <Result Description>: <SSD IMEI>`

Table 3-7 MT Message Queuing Indication Subject Field Descriptions

<table>
<tr><th>Field Name</th><th>Description</th></tr>
<tr><td>Queuing Disposition</td><td>Indicates the status of the queuing of the MT message. This field will be one of the following two values:

<table><tr><th>Field Value</th><th>Description</th></tr><tr><td>Success</td><td>The message was successfully placed in the SSD delivery queue.</td></tr><tr><td>Error</td><td>The message was not placed in the SSD delivery queue.</td></tr></table></td></tr>
<tr><td>Result Description</td><td>Additional text expanding on the Queuing Disposition</td></tr>
<tr><td>SSD IMEI</td><td>The SSD IMEI is the identity of the unit to which the message was to be queued. This field will be one of the following values:

<table><tr><th>Field Value</th><th>Description</th></tr><tr><td>(No IMEI Specified)</td><td>No IMEI was provided in the subject line of the e-mail sent to the GSS.</td></tr><tr><td>(Invalid IMEI Specified)</td><td>The IMEI in the subject line of the e-mail sent to the GSS was not in the proper format. Additional details can be found in the reason code in the body of the message.</td></tr><tr><td>Actual IMEI</td><td>The actual 15 numeric digit IMEI of the destination SSD.</td></tr></table></td></tr>
</table>

Details of the individual field in the disposition indication e-mail body are in Table 3-8. The format of the body is as follows:

```
<Description>
IMEI: <IMEI number>
Time: <Date>
Attachment Filename: <Name>
Attachment Size: <Size> bytes

[Reason:] <Descriptive Text>
```

Table 3-8 MT Message Queuing Acknowledgement Body Field Descriptions

Field Name	Description
<Description>	Additional text expanding on the Queuing Disposition in the subject line.
IMEI	The number of the unit to which the message was to be queued. This field will be one of the value described below:

Field Value	Description
(none specified)	No IMEI was provided in the subject line of the e-mail sent to the GSS.
Received IMEI	The IMEI received by the GSS. In the case of Success, this is the 15 numeric digit IMEI of the SSD. In the case of Error, this is the invalid alpha numeric string received by the GSS.

Time	The timestamp, in UTC, when the acknowledgement was sent from the GSS.
Attachment Filename	The filename of the attachment received by the GSS.
Attachment Size	The size of the attachment received by the GSS.
[Reason:] <Descriptive Text>	The text "Reason:" will only appear if the Queuing Disposition in the subject line is "Error". The descriptive text will provide additional details regarding the error.

3.2.1.2 MT Messages – DirectIP

Messages may be queued for any SSD using the new DirectIP protocol. There is no mechanism for MT messaging in the legacy protocol. The protocol is described in the *EMSS SBD DirectIP Specification 1.2* available at the SBD web site. Part of the protocol for MT DirectIP is a confirmation message passed back to the vendor prior to closing the socket connection. See the specification for further information.

3.2.1.3 MT Messages – SSD-to-SSD

An SSD may be provisioned to send MO messages directly to the delivery queue of another SSD. No additional user actions are required in this case. When the originating SSD sends the message, it is immediately placed in the delivery queue of the destination SSD. It should be noted that no SBD session descriptors are delivered to either the originating or terminating SSDs in this case. There is also no disposition indication of any kind. The retrieving SSD must initiate an SBD session to retrieve

messages queued in this fashion, exactly as if the MT message was queued using any other MT delivery method.

There are three error conditions that may be reflected in the delivery status of the CDR for undeliverable messages. Messages that have zero bytes or greater than 1890 bytes (9505[a] or 9522[a]) or 135 bytes (off-the-shelf 9601), will not be queued for delivery and will be marked as "SSDtoSSD_PayloadInvalid". Messages destined for SSDs whose MT queue is full will be marked as "SSDtoSSD_QueueFull". Finally, messages destined for an SSD that is not a registered SBD user will be marked as "SSDtoSSD_UnknownSSD".

3.2.1.4 MT Messages – Web Page

Messages may be queued for any SSD via the SBD web pages at https://sbd.pac.disa.mil using the form provided therein. A valid SBD user account is needed in order to access the form.

3.2.2 MT Ring Alerts

An SSD that has SBD ring alert-enabled software [2] may receive notification via an over-the-air ring alert transmission when a new MT message is received at the GSS for that particular SSD. Prior to receiving such notifications, the following criteria must be met:

- The SSD must first be provisioned, or enabled at the GSS, to receive ring alerts.
- The SSD must "register" its location with the GSS. This may be done with an SBD registration message (+SBDREG) or a normal SBD session (+SBDIX).

Note that sessions initiated with the original initiation command (+SBDI) will effectively disable ring alerts to the SSD until the SSD re-registers with the GSS. All SBD transfers to and from and SSD will continue to proceed unaffected whether ring alerts are enabled for the SSD or not. Only the MT notification via ring alert is affected. When a ring alert is received, the answer to ring initiation command (+SBDIXA) should be used to initiate the response. These AT commands are described in detail in section 4.2.

3.2.3 MT Delivery Short Codes

There are two delivery short codes (DSC) that affect MT message deliveries. The DSC value is sent as part of each SBD session. It is made up of eight bit fields, or individual flags (codes). The desired DSCs are set using the +SBDDSC AT command (see section 4.2.5).

3.2.3.1 Hold MT Message Delivery

When this DSC is set (0x80 – most significant bit in the DSC field), the GSS indicates to the calling SSD how many MT messages are queued in the GSS for that SSD, but no MT message is actually sent. The number of MT messages queued reported to the SSD includes the message currently at the front of the queue. Note that when an MT message is delivered, the number of MT messages queued reported to the SSD does not include the message being delivered in that session.

3.2.3.2 Leave MT Message in Queue After Delivery

When this DSC is set (0x40 – next-to-most significant bit in the DSC field), the GSS delivers the MT message queued for the calling SSD. After the delivery, the message is left in its queue, essentially setting up the MT message for redelivery in the future. The number of MT messages queued reported to the SSD does not include the message being delivered and held even though it will remain following the SBD session.

4 DTE field application to SSD Interface

4.1 RS-232 Interface

The RS-232 interface on the SSDs is designed to auto-baud to the DTE device connected to it. The various port configuration options are outlined in Table 4-1.

Table 4-1 RS-232 Port Settings

Port Parameter	Acceptable Values	Default Value
Speed	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	19200
Data Bits	7, 8	8
Stop Bits	1, None	1
Parity	Even, Odd, None	None

4.2 AT Commands

This section describes the subset of AT commands used to implement the SBD functionality. Refer to references [3] and [4] for all other AT commands that are supported by Iridium equipment.

4.2.1 +SBDAREG – Automatic Registration

Set command: +SBDAREG=<mode>

This command sets the SSD's auto-registration mode.

<mode>:

0 Disable automatic registration (default)

1 Set the auto-registration mode to "Automatic"

2 Set the auto-registration mode to "Ask"

When auto-registration is enabled, modes 1 or 2, the SSD monitors its current location and triggers an auto-registration when it determines that the SSD has moved sufficiently far away from its last registered location. Note that auto-registration runs only while the SSD is attached to the gateway (i.e. the registration status is "Not registered" or "Registered").

Auto-registration may only be used with system-provided location, or the gateway-determined location. The FA should use the manual registration command (+SBDREG) when providing its own location (e.g. GPS). **Note that SSD-provided locations are not yet supported by the GSS.**

Upon triggering in mode 1, "Automatic", the SSD autonomously initiates an SBD session in order to perform a registration with the updated location of the SSD, with the session type set to 3 – auto-registration. This session does not transfer any MO or MT messages.

Upon triggering in mode 2, "Ask", the SSD reports to the FA that it should register with the system because the SSD location has changed (see <event> below). It is then the responsibility of the FA to register via +SBDREG or +SBDIX. +SBDIX allows the FA to register while including an MO message and/or retrieving an MT message that is stored at the Gateway.

When auto-registration is enabled, modes 1 or 2, the SSD reports relevant events to the FA by issuing an unsolicited result code +AREG:<event>,<reg error>

where:

<event>:

0 Suggest FA makes a registration attempt (mode 2 only)

1 Auto-registration has been performed successfully (mode 1 only)

2 Auto-registration has failed and will be retried after a delay (mode 1 only)

<reg error>:

GSS-reported values

0 No error

2 Session completed but the requested Location Update was not accepted

3..14 Reserved, but indicate Location Update failure if used

15 Access is denied

SSD-reported values

16 SSD has been locked and may not make SBD calls

17 GSS not responding (local session timeout)

18 Connection lost (RF drop)

19..31 Reserved, but indicate failure if used

32 No network service, unable to initiate call

33 Antenna Fault, unable to initiate call

34 Radio is disabled, unable to initiate call

35 SSD is busy, unable to initiate call (typically performing auto-registration)

36 Try later, must wait 3 minutes since last registration

37.. Reserved, but indicate failure if used

Read Command: +SBDAREG?

This command queries the current automatic MT registration mode. The response is of the form:

+SBDAREG:<mode>

Test Command: +SBDARG?

This command lists the supported mode settings. The response is of the form:

+SBDARG:(supported <mode> settings)

4.2.2 +SBDC – Clear SBD MOMSN

Exec Command: +SBDC

This command will clear (set to 0) the MOMSN number stored in the SSD.

- The message sequence number is maintained even after power cycle.

Command Response:

0 The MOMSN was cleared successfully

1 An error occurred while clearing the MOMSN

4.2.3 +SBDD – Clear SBD Message Buffer(s)

Exec Command: +SBDD<Delete type>

This command is used to clear the send (MO) buffer, receive (MT) buffer or both.

<Delete type>

0 Clear the send buffer

1 Clear the receive buffer

2 Clear both the send and receive buffers

- Using this command or power-cycling the phone is the only means by which both buffers are cleared
- The receive buffer will be cleared when an SBD session is initiated
- Sending a message from the SSD to the GSS does not clear the send buffer
- Reading a message from the SSD does not clear the receive buffer
- Clearing the send buffer (Delete types 0 and 2) will also set the DSC to 0

Command Response:

0 Buffer(s) cleared successfully

1 An error occurred while clearing the buffer(s)

4.2.4 +SBDDET – Detach

Exec Command: +SBDDET

This command initiates an SBD session to detach the SSD, or disable SBD ring alerts, at the GSS.

Command Response:

+SBDDET:<status>,<error>

where:

<status>:

0 Detach successfully performed

1 An error occurred while attempting the detach

<error>:

GSS-reported values

0 No error

1..4 Reserved, but indicate success if used

5..14 Reserved, but indicate failure if used

15 Access is denied

SSD-reported values

16: SSD has been locked and may not make SBD calls

17: Gateway not responding (local session timeout)

18: Connection lost (RF drop)

19..31 Reserved, but indicate failure is used

32 No network service, unable to initiate call

33 Antenna fault, unable to initiate call

34 Radio is disabled, unable to initiate call

35 SSD is busy, unable to initiate call (typically performing auto-registration)

36.. Reserved, but indicate failure if used

This instructs the GSS to disable (detach) SBD ring alerts for the calling SSD. Successful completion of the detach command implies that the GSS has performed the requested detach action and the SSD is no longer registered. This session does not transfer any MO or MT messages.

Note: A user can send an MO message and request to detach at the same time by using the +SBDI command. The +SBDI command always requests to detach.

4.2.5 +SBDDSC – Set Delivery Short Code

Set Command: +SBDDSC=<dsc >

This command sets the value of the delivery short code (DSC) that will be used for all transactions with the GSS. The DSC is a bit masked field, with each individual bit controlling a particular feature. Currently only two features are defined. Flag values can be added together to achieve a combination of settings.

Some fields may be "locked" when the SSD is in a special mode (e.g. SBD registration locks the flag values 0x80 and 0x40 on). However, these modes do not affect the DSC setting maintained in the SSD. In other words, the DSC used in a special mode is only in affect while that mode is active and does not affect subsequent SBD sessions.

When the phone is powered on, the DSC value is 0 (zero). Once a value is set, it is static until either the phone is powered off or a new +SBDDSC command is issued. Also, note that the DSC is cleared when the send buffer is cleared (+SBDD0 or +SBDD2). However, it is not affected when the receive buffer is cleared (+SBDD1).

<dsc>

0..255 DSC to be used for subsequent MO messages (0 default)

0x80 Hold MT message delivery (number of MT messages queued is returned)

0x40 Leave MT message in queue in GSS after delivery

Set Command Response:

0 Value of the DSC successfully changed

1 The value of the DSC is out of the allowable range

Read Command: +SBDDSC?

This command queries the current DSC setting in the SSD.

Read Command Response: +SBDDSC:<dsc>

4.2.6 +SBDGW – Gateway

Exec Command: +SBDGW

This command will read back the Iridium gateway to which the SSD is homed to determine if the SSD is homed to the EMSS gateway or not (i.e. the gateway to which SBD messages are sent to and from).

Command Response:

+SBDGW:<gateway>

where:

<gateway>:

EMSS	The SSD is homed to the EMSS gateway
Non-EMSS	The SSD is not homed to the EMSS gateway

Note that this command is only supported on hardware and firmware released in 2010.

4.2.7 +SBDI – Initiate an SBD Session

Note: The +SBDI command is provided for backwards compatibility with existing SSDs and FAs that do not have SBD ring alert functionality.

Exec Command: +SBDI

This command initiates an SBD session between the SSD and the GSS, setting the DSC according to the value previously specified by the +SBDDSC command. For SBD sessions invoked with this command:

- The detach/attach flag is fixed at the detach setting (disables ring alerts)
- The location update flag is fixed at the "no update" setting
- The SBD session type will always be mobile-originated

SSDs with ring alert functionality may use this command, with the subsequent default flag settings, or the extended +SBDIX command.

This command initiates an SBD session between the SSD and the GSS. If there is an MO message in the send buffer, it will be transferred to the GSS. Similarly, if there is one or more MT messages queued at the GSS, the oldest will be transferred to the SSD and placed into the receive buffer in a first-in-first-out (FIFO) manner, in the order they were received at the GSS.

Command Response:

+SBDI: <MO status>,<MOMSN>,<MT status>,<MTMSN>,<MT length>,<MT queued>

where:

<MO status>:

The MO status provides an indication of the disposition of the message send transaction. The field can take on the following values:

0 No SBD message to send from the SSD

1 SBD message successfully sent from the SSD to the GSS

2 An error occurred while attempting to send SBD message from SSD to GSS

<MOMSN>:

The MOMSN is a value assigned by the SSD when initiating and SBD session regardless of whether or not an MO message is to be transferred. This value is incremented each time an SBD session is successfully completed between the SSD to the GSS. This wrap-around counter will range from 0 to 65535.

<MT status>:

The MT status provides an indication of the disposition of the message receive transaction. The field can take on the following values:

0 No SBD message to receive from the GSS

1 SBD message successfully received from the GSS

2 An error occurred while attempting to perform a mailbox check or receive a message from the GSS

<MTMSN>:

The MTMSN is assigned by the GSS when the MT message is first received at the GSS. The value assigned is tied to the destination SSD such that each SSD's MTMSN is independent of all others. It is incremented each time a message is received at the GSS. This value is indeterminate if the field <MT status> is zero. This wrap-around counter will range from 1 to 65535.

<MT length>:

The MT length is the length in bytes of the MT message received from the GSS. If no message was received, this field will be zero.

<MT queued>:

The MT queued value is a count of MT messages waiting at the GSS to be transferred to the SSD.

4.2.8 +SBDIX[A] – Initiate an SBD Session Extended

Note: The +SBDIX command must be used in place of the +SBDI command for FAs wishing to make use of SBD ring alert functionality.

Exec Command: +SBDIX[A][=<location>]

This command initiates an SBD session between the SSD and the GSS, setting the SBD session type according to the type of command and the DSC according to the value previously specified by the +SBDDSC command. The FA should append an 'A' to the command (i.e. +SBDIXA) when the SBD session is in response to a ring alert. For SBD sessions invoked with this command:

- The detach/attach flag is fixed at the attach setting (enables ring alerts)
- The location update flag is fixed at the "update" setting
- The SBD session type will be mobile-originated for the +SBDIX command
- The SBD session type will be answer-to-ring for the +SBDIXA command

Thus, this command will always attempt an SBD registration, consisting of an attach and location update, during the SBD session in order to support SBD ring alerts. If this is not desired, the +SBDI command should be used.

If there is a message in the send buffer, it will be transferred to the GSS. Similarly, if there is one or more MT messages queued at the GSS, the oldest will be transferred to the SSD and placed into the receive buffer.

Note that processing a location reported from the SSD as shown below is not yet supported at the GSS.

<location> has format:

[+|-]DDMM.MMM,[+|-]dddmm.mmm

where:

DD Degrees latitude (00-89)

MM Minutes latitude (00-59)

MMM Thousandths of minutes latitude (000-999)

ddd Degrees longitude (000-179)

mm Minutes longitude (00-59)

mmm Thousandths of minutes longitude (000-999)

The optional sign indicators specify latitude North (+) or South (-), and longitude East (+) or West (-). If omitted, the default is +.

For example,

AT+SBDIX=5212.483,-00007.350

corresponds to 52 degrees 12.483 minutes North, 0 degrees 7.35 miles West.

Command Response:

+SBDIX:<MO status>,<MOMSN>,<MT status>,<MTMSN>,<MT length>,<MT queued>

where:

<MO status>:

MO session status provides an indication of the disposition of the mobile originated transaction. The field can take on the following values:

GSS-reported values

0 MO message, if any, transferred successfully

1 MO message, if any, transferred successfully, but the MT message in the queue was too big to be transferred

2 MO message, if any, transferred successfully, but the requested Location Update was not accepted

3..4 Reserved, but indicate MO session success if used

5..8 Reserved, but indicate MO session failure if used

10 GSS reported that the call did not complete in the allowed time

11 MO message queue at the Gateway is full

12 MO message has too many segments

13 Gateway reported that the session did not complete

14 Invalid segment size

15 Access is denied

SSD-reported values

16 SSD has been locked and may not make SBD calls

17 Gateway not responding (local session timeout)

18 Connection lost (RF drop)

19..31 Reserved, but indicate MO session failure if used

32 No network service, unable to initiate call

33 Antenna fault, unable to initiate call

34 Radio is disabled, unable to initiate call

35 SSD is busy, unable to initiate call (typically performing auto-registration)

36 Reserved, but indicate failure if used

<MOMSN>:

The MOMSN is a value assigned by the SSD when initiating and SBD session regardless of whether or not an MO message is to be transferred. This value is incremented each time an SBD session is successfully completed between the SSD to the GSS. This wrap-around counter will range from 0 to 65535.

<MT status>:

The MT status provides an indication of the disposition of the mobile terminated transaction. The field can take on the following values:

0 No MT message to receive from the GSS

1 MT message successfully received from the GSS

2 An error occurred while attempting to perform a mailbox check or receive a message from the GSS

<MTMSN>:

The MTMSN is assigned by the GSS when the MT message is first received at the GSS. The value assigned is tied to the destination SSD such that each SSD's MTMSN is independent of all others. It is incremented each time a message is received at the GSS. This value is indeterminate if the field <MT status> is zero. This wrap-around counter will range from 1 to 65535.

<MT length>:

The MT length is the length in bytes of the MT message received from the GSS. If no message was received, this field will be zero.

<MT queued>:

The MT queued value is a count of MT messages waiting at the GSS to be transferred to the SSD.

4.2.9 +SBDMTA – Mobile-Terminated Alert

Set Command: +SBDMTA=<mode>

This command enables or disables ring indications for SBD ring alerts.

<mode>:

0 Disable ring indication

1 Enable ring indication (default)

When ring indication is enabled, the SSD asserts the RI line and issues the unsolicited result code SBDRING when an SBD ring alert is received.

Read command: +SBDMTA?

This command queries the current ring indication mode. The response is of the form:

+SBDMTA:<mode>

Test Command: +SBDMTA=?

This command lists the supported mode settings. The response is of the form:

+SBDMTA:(supported <mode> settings)

4.2.10 +SBDRB – Read Binary Data from the SSD

Exec Command: +SBDRB

This command is used to transfer a binary SBD MT message from the single receive buffer in the SSD to the DTE field application. This buffer will never contain more than one MT message.

- The SBD message is transferred formatted as follows:

  {2-byte message length} + {binary SBD message} + {2-byte checksum}

  - The {2-byte message length} field represents the length, in bytes, of the SBD message not including the length field or the mandatory two-byte checksum. The high order byte will be sent first.
  - The checksum is the least significant 2-bytes of the summation of the entire SBD message. The high order byte will be sent first. For example if the SSD were to send the word "hello" encoded in ASCII to the DTE field application the binary stream would be hex 00 05 68 65 6c 6c 6f 02 14.
  - If there is no received message waiting to be retrieved from the SSD, the message length and checksum fields will be zero.
  - The maximum SBD MT message length is 1890 bytes.
- The receive message buffer will be empty upon power-up. Any data in the buffer will be lost when the SSD is power-cycled.

Command Response:

There are no response codes generated by the SSD for this command.

4.2.11 +SBDREG – Network Registration

Exec Command: +SBDREG[=<location>]

This command initiates an SBD session to perform a manual SBD registration, consisting of an attach and location update. This session does not transfer any MO or MT messages. For SBD sessions invoked with this command:

- The detach/attach flag is fixed at the attach setting (enables ring alerts)
- The location update flag is fixed at the "update" setting
- The SBD session type will be registration
- DSC flag 0x80 (hold MT delivery) is set independent of the currently held DSC value (the held value is not changed

The SSD restricts the number of manual and automatic registrations to one every 3 minutes. Successive attempts within 3 minutes will return an error code indicating that the FA should try later (see error 36 below).

Note: A user can send and receive SBD messages and register at the same time by using the +SBDIX command. The +SBDIX command always performs a registration attempt and should be used for applications requiring SBD ring alerts. The +SBDI command never includes an SBD registration and should be used for applications that do not require SBD ring alerts.

Note that processing a location reported from the SSD as shown below is not yet supported at the GSS.

<location> has format:

[+|-]DDMM.MMM,[+|-]dddmm.mmm

where:

DD Degrees latitude (00-89)

MM Minutes latitude (00-59)

MMM Thousandths of minutes latitude (000-999)

ddd Degrees longitude (000-179)

mm Minutes longitude (00-59)

mmm Thousandths of minutes longitude (000-999)

The optional sign indicators specify latitude North (+) or South (-), and longitude East (+) or West (-). If omitted, the default is +.

For example,

AT+SBDIX=5212.483,-00007.350

corresponds to 52 degrees 12.483 minutes North, 0 degrees 7.35 miles West.

Command Response:

+SBDREG:<status>,<reg err>

where:

<status> indicates the resulting registration status of the SSD:

0 Detached – SSD is detached as a result of a successful +SBDDET or +SBDI command

1 Not registered – SSD is attached but has not provided a good location since it was last detached

2 Registered – SSD is attached with a good location. Note that this may be the case even when the most recent attempt did not provide a good location

3 Registration denied – The gateway is denying service to the SSD

4 Unknown

<reg err>:

GSS-reported values

0 No error

2 Session completed but the requested Location Update was not accepted

3..14 Reserved, but indicate Location Update failure if needed

15 Access is denied

SSD-reported values

16 SSD has been locked and may not make SBD calls

17 GSS not responding (local session timeout)

18 Connection lost (RF drop)

19..31 Reserved, but indicate failure if used

32 No network service, unable to initiate call

33 Antenna fault, unable to initiate call

34 Radio is disabled, unable to initiate call

35 SSD is busy, unable to initiate call (typically performing auto-registration)

36 Try later, must wait 3 minutes since last registration

37.. Reserved, but indicate failure if used

Read Command: +SBDREG?

This command queries the current SBD registration status of the SSD. The response is of the form:

+SBDREG:<status>

<status>:

0 Detached

1 Not registered

2 Registered

3 Registration denied

4 Unknown

The registration status is stored in the SSD non-volatile memory and, therefore, may be queried by the FA after powering up.

4.2.12 +SBDRT – Read a Text Message from the SSD

Exec Command: +SBDRT

This command is used to transfer a text SBD message from the single receive buffer in the SSD to the DTE field application. This command works just like the +SDDRB but does not provide a length indicator or checksum. The intent of this command is to provide a human-friendly interface to SBD for demonstrations and application development.

- Once the command is entered, the SBD message in the receive buffer is sent out of the port.
- The maximum receive SBD MT message length is 1890 bytes.
- The receive message buffer will be empty upon power-up. Any data in the buffer will be lost when the SSD is power-cycled.

Command Response: +SBDRT:<CR>{receive buffer}

4.2.13 +SBDS – SBD Status

Exec Command: +SBDS

This command returns current state of the send and receive buffers.

Command Response: +SBDS: <MO flag>, <MOMSN>, <MT flag>, <MTMSN>

where:

<MO flag>:

The send flag indicates the existence of a message in the send buffer. The response from the SSD will be one of the following numeric codes:

0: No message in send buffer

1: Message in send buffer

<MOMSN>:

The MOMSN identifies the sequence number that will be used during the next SBD session.

<MT flag>:

The receive flag indicates the existence of a message in the receive buffer. The response from the SSD will be one of the following numeric codes:

0: No message in receive buffer

1: Message in receive buffer

<MTMSN>:

The MTMSN identifies the sequence number associated with the MT message in the receive buffer. This value will be (-1) if there is nothing in the receive buffer.

4.2.14 +SBDTC – Transfer Send Buffer to Receive Buffer

Exec Command: +SBDTC

This command will transfer the contents of the send buffer to the receive buffer. Developers of DTE field applications can use this command to test reading and writing to the SSD without actually initiating SBD sessions with the GSS.

Command Response: (Note: *SBM* is equivalent to *SBD*)

SBMT: Outbound SBM Copied to Inbound SBM: Osize = <size>, Isize = <Isize>

- or -

SBDTC: Outbound SBD copied to Inbound SBD: size = <size>

where:

<size>:

The length of the message in bytes

4.2.15 +SBDWB – Write Binary Data to the SSD

Exec Command: +SBDWB=<SBD message length>

This command is used to transfer a binary SBD MO message from the DTE field application to the single send buffer in the SSD. This buffer will never contain more than one MO message.

- Once the command is entered, and the message length is acceptable, the SSD will indicate to the DTE field application that it is prepared to receive the message by sending the ASCII encoded string “READY<CR><LF>” (hex 52 45 41 44 59 0D 0A) to the DTE field application.
- The <SBD message length> parameter represents the length, in bytes, of the SBD message not including the mandatory two-byte checksum.
- The maximum send SBD message length is 1960 bytes. The minimum send SBD message length is 1 byte. If the parameter is out of range, the SSD will issue response 3 (see below).
- Once the DTE field application receives the READY indication from the SSD, the SBD message must be sent from the DTE field application formatted as follows:

  {binary SBD message} + {2-byte checksum}

  - The checksum is the least significant 2-bytes of the summation of the entire SBD message. The high order byte must be sent first. For example, if the DTE field application was to send the word “hello” encoded in ASCII to the SSD the binary stream would be hex 68 65 6c 6c 6f 02 14.

- If any data is currently in the send buffer, it will be overwritten.
- The send buffer is empty upon power-up. Any data in the buffer will be lost when the SSD is power-cycled.

Command Response:

0 MO message successfully written to SSD

1 MO message write timeout. An insufficient number of bytes were transferred to SSD during the transfer period of 60 seconds

2 MO message checksum sent from DTE does not match the checksum calculated at the SSD

3 MO message size is not correct. The maximum send SBD message length is 1960 bytes. The minimum send SBD message length is 1 byte

4.2.16 +SBDWT – Write a text message to the SSD

Exec Command: +SBDWT=<text message>

This command is used to transfer a text SBD message from the DTE field application to the single send buffer in the SSD.

- The text message may be entered on the command line. For example, “AT+SBDWT=hello”.
- The length of <text message> is limited to 120 characters due to the length limit on the AT command line interface.
- The message is terminated when a carriage return is entered.
- The send buffer will be empty upon power-up. Any data in the buffer will be lost when the SSD is power-cycled.

Command Response:

OK	Message successfully stored in the send buffer
ERROR	An error occurred storing the message in the send buffer

5 SBD Scenarios

5.1 Mobile Originated Message

In the case that the DTE field application needs to send data to the vendor application, and there is no MT data queued for delivery to the SSD, the SSD session will include a one-way transfer of MO data. In this scenario, the DTE field application will load an MO message into the SSD, initiate an SBD session, and evaluate and act on the results of the SBD session (Table 5-1). Finally, the GSS will forward the MO message to the vendor application (Figure 5-1).

Table 5-1 DTE Field Application to SSD Interface, MO Message

To SSD (from DTE)	To DTE (from SSD)	Description
AT+SBDWB=351↵		The DTE field application informs the SSD that it will write a 351 byte message to the SSD.
	READY↵	The SSD informs the DTE field application that it is ready to receive the message
Binary transfer		The DTE field application sends the 351 byte message followed by the two byte checksum to the SSD. This transfer is not echoed.
	0↵	The SSD will send a zero result code to the DTE field application indicating that the message was loaded without an error.
	OK	Standard AT command completion response.
AT+SBDI↵		The DTE field application instructs the SSD to initiate an SBD transfer.
	+SBDI: 1, 23, 0, -1, 0, 0↵	The SSD informs the DTE field application that the message was sent successfully using MOMSN 23. No message was received and no messages are queued.
	OK	Standard AT command completion response.
AT+SBDD0↵		The DTE field application instructs the SSD to clear the message from the send buffer.
	0↵	The SSD informs the DTE field application that the message buffer was cleared successfully.
	OK	Standard AT command completion response.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 12:49 PM
Subject: SBD Msg From Unit: 304050607080903

MOMSN: 23
Time of Session (UTC): Tue Aug 13 16:51:04 2002
Session Status: 00 - Transfer OK
Message Size (bytes): 351

Unit Location: Lat = 59.372463 Long = 75.309806
CEPradius = 3

Message is Attached.
```

Figure 5-1 Vendor Application to GSS Interface, MO Message

5.2 Mailbox Check / Mobile Terminated (MT) Message

In the case that the DTE field application needs to receive data from the vendor application, the SSD will be used in a receive-only fashion. If at least one message is waiting for the SSD at the GSS, the message is received by the SSD. There is no difference between the mailbox check and the receive-only session other than that, in the receive-only case, a message is forwarded from the GSS to the SSD. In this scenario, a MT message is sent from the vendor application to the GSS (Figure 5-2); the DTE field application will initiate an SBD session, evaluate the results of the SBD session, and read the MT message from the SSD (Table 5-2). After the SBD session completes, the GSS will forward the empty message to the vendor application indicating the disposition of the SBD session.

```
To: data@sbd.pac.disa.mil
From: VA@VendorDomain.com
Subject: 304050607080903

<561 byte message attached>
```

Figure 5-2 Vendor Application to GSS Interface, Mailbox Check / MT Message

Table 5-2 DTE Field Application to SSD Interface, Mailbox Check / MT Message

To SSD (from DTE)	To DTE (from SSD)	Description
AT+SBDD0		The DTE field application instructs the SSD to clear the send buffer.
	OK	Standard AT command completion response.
AT+SBDI↵		The DTE field application instructs the SSD to initiate an SBD transfer.

To SSD (from DTE)	To DTE (from SSD)	Description
	+SBDI: 0, 498, 1, 237, 561, 2↵	The SSD informs the DTE field application that no message was sent and a 561 byte message was successfully received with MTMSN 237. Two additional messages are queued.
	OK	Standard AT command completion response.
AT+SBDRB↵		The DTE field application instructs the SSD to transfer the received message.
	Binary transfer	The SSD sends a two byte length indicator followed by the 561 byte message followed by the two byte checksum to the DTE field application.
	OK	Standard AT command completion response.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 12:49 PM
Subject: SBD Msg From Unit: 304050607080903

MOMSN: 489
MTMSN: 237
Time of Session (UTC): Tue Aug 13 16:51:04 2002
Session Status: 00 – Transfer OK
Message Size (bytes): 0

Unit Location: Lat = 59.372463 Long = 75.309806
CEPradius = 3

Message is Attached
```

Figure 5-3 Vendor Application to GSS Interface, Vendor Application Generated E-mail

5.3 MO and MT Message

In the case where the DTE field application needs to send and receive data from a central information recipient/source, the SSD will be used in a send and receive fashion. In this scenario, the vendor application will send the MT message to the GSS (Figure 5-4), the DTE field application will load the MO message into the SSD, initiate an SBD session, evaluate the results of the SBD session, and read the received message from the SSD (Table 5-3). Finally, the vendor application will receive the MO message (Figure 5-5).

```
To: data@sbd.pac.disa.mil
From: VA@VendorDomain.com
Subject: 304050607080903
<429 byte message attached>
```

Figure 5-4 Vendor Application to GSS Interface, MT Message

Table 5-3 DTE Field Application to SSD Interface, Send and Receive

To SSD (from DTE)	To DTE (from SSD)	Description
AT+SBDWB=351↵		The DTE field application informs the SSD that it will write a 351 byte message into the SSD.
	READY↵	The SSD informs the DTE field application that it is ready to receive the message
Binary transfer		The DTE field application sends the 351 byte message followed by the two byte checksum to the SSD. This transfer is not echoed.
	0↵	The SSD will send a zero result code to the DTE field application indicating that the message was loaded without an error.
	OK	Standard AT command completion response.
AT+SBDI↵		The DTE field application instructs the SSD to initiate an SBD transfer.
	+SBDI: 1, 2173, 1, 87, 429, 0↵	The SSD informs the DTE field application that the message was sent successfully using MOMSN 2173. A 429 byte message was received using MTMSN 87. No additional messages are queued.
	OK	Standard AT command completion response.
AT+SBDD0↵		The DTE field application instructs the SSD to clear the message from the send buffer.
	0↵	The SSD informs the DTE field application that the message buffer was cleared successfully.
	OK	Standard AT command completion response.

To SSD (from DTE)	To DTE (from SSD)	Description
AT+SBDRB↵		The DTE field application instructs the SSD to transfer the received message.
	Binary transfer	The SSD sends a two-byte length indicator followed by the 429 byte message followed by the two byte checksum to the DTE field application.
	OK	Standard AT command completion response.

```
From: service@sbd.pac.disa.mil
Sent: Tuesday, August 13, 2002 12:49 PM
Subject: SBD Msg From Unit: 304050607080903

MOMSN: 2173
MTMSN: 429
Time of Session (UTC): Tue Aug 13 16:51:04 2002
Session Status: 00 - Transfer OK
Message Size (bytes): 351

Unit Location: Lat = 59.372463 Long = 75.309806
CEPradius = 3

Message is Attached.
```

Figure 5-5 Vendor Application to GSS Interface, MO Message

6 Support

The SBD Service is provided by the DISA EMSS Program Office. See http://inah.pac.disa.mil for more information regarding EMSS Services.

For technical questions regarding the EMSS SBD Service and/or the EMSS SBD Web Pages, send e-mail to support@sbd.pac.disa.mil.