Dialogic®
Making Innovation Thrive™

Dialogic® DSI Signaling Protocols

RMM Programmer's Manual

November 2009 U35SSS

www.dialogic.com

Copyright and Legal Notice

All contents of this document are furnished for informational use only and are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation or its subsidiaries ("Dialogic"). Reasonable effort is made to ensure the accuracy of the information contained in the document. However, Dialogic does not warrant the accuracy of this information and cannot accept responsibility for errors, inaccuracies or omissions that may be contained in this document.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH DIALOGIC® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN A SIGNED AGREEMENT BETWEEN YOU AND DIALOGIC, DIALOGIC ASSUMES NO LIABILITY WHATSOEVER, AND DIALOGIC DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF DIALOGIC PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF A THIRD PARTY.

Dialogic products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Due to differing national regulations and approval requirements, certain Dialogic products may be suitable for use only in specific countries, and thus may not function properly in other countries. You are responsible for ensuring that your use of such products occurs only in the countries where such use is suitable. For information on specific products, contact Dialogic Corporation at the address indicated below or on the web at www.dialogic.com.

It is possible that the use or implementation of any one of the concepts, applications, or ideas described in this document, in marketing collateral produced by or on web pages maintained by Dialogic may infringe one or more patents or other intellectual property rights owned by third parties. Dialogic does not provide any intellectual property licenses with the sale of Dialogic products other than a license to use such product in accordance with intellectual property owned or validly licensed by Dialogic and no such licenses are provided except pursuant to a signed agreement with Dialogic. More detailed information about such intellectual property is available from Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. **Dialogic encourages all users of its products to procure all necessary intellectual property licenses required to implement any concepts or applications and does not condone or encourage any intellectual property infringement and disclaims any responsibility related thereto. These intellectual property licenses may differ from country to country and it is the responsibility of those who develop the concepts or applications to be aware of and comply with different national license requirements.**

Dialogic, Dialogic Pro, Brooktrout, Diva, Cantata, SnowShore, Eicon, Eicon Networks, NMS Communications, NMS (stylized), Eiconcard, SIPcontrol, Diva ISDN, TruFax, Exnet, EXS, SwitchKit, N20, Making Innovation Thrive, Connecting to Growth, Video is the New Voice, Fusion, Vision, PacketMedia, NaturalAccess, NaturalCallControl, NaturalConference, NaturalFax and Shiva, among others as well as related logos, are either registered trademarks or trademarks of Dialogic Corporation or its subsidiaries. Dialogic's trademarks may be used publicly only with permission from Dialogic. Such permission may only be granted by Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. Any authorized use of Dialogic's trademarks will be subject to full respect of the trademark guidelines published by Dialogic from time to time and any use of Dialogic's trademarks requires proper acknowledgement.

The names of actual companies and products mentioned herein are the trademarks of their respective owners.

This document discusses one or more open source products, systems and/or releases. Dialogic is not responsible for your decision to use open source in connection with Dialogic products (including without limitation those referred to herein), nor is Dialogic responsible for any present or future effects such usage might have, including without limitation effects on your products, your business, or your intellectual property rights.

Publication Date: November 2009

Document Number: U35SSS, Issue 1

Contents

Figures

Tables

Revision History

Issue	Date	Description
1	11-Nov-09	Initial Release

Note: The current version of this guide can be found at:
http://www.dialogic.com/support/helpweb/signaling

1 Introduction

The Dialogic® DSI RMM (Resilient MTP Management) module can be used in dual resilient systems to provide a mechanism to send MTP3 / M3UA messages via either unit in the dual resilient system. This Programmer's Manual is intended for users choosing to develop their own application programs that will interface with and use the functionality provided by the M3UA or MTP3 module.

This manual is largely for reference only. The RMM module should be configured by the management module S7_Mgt. The RMM and S7_mgt modules are part of the Dialogic® DSI Development Package and are implemented in a similar message-based, event-driven manner to the other modules in the Dialogic® DSI product range.

This manual provides an overview of the internal operation of the RMM module and defines the message-based APIs for configuration and management applications to use.

1.1 Abbreviations

Abbreviation	Description
M3UA	MTP3 User Adaptation Layer
MTP3	Message Transfer Part Layer 3
DPC	Destination Point Code
SI	Service Indicator
SS7	Signaling System Number 7
RMM	Resilient MTP Management

1.2 Related Documentation

[1] Dialogic® DSI Software Environment Programmer's Manual

[2] Dialogic® DSI MTP Programmer's Manual

[3] Dialogic® DSI M3UA Programmer's Manual

1.3 Feature Overview

Notable features of the RMM module include:

- User interface common with other Dialogic® DSI Signaling Protocols
- Message based interface
- May be used with MTP3 or M3UA modules
- Supports 14, 16 and 24 bit Point Codes
- The RMM module allows outbound messages from a partner unit in a dual resilient system to be sent via the remote M3UA / MTP3 if the route is unavailable

2 General Description

2.1 System Overview

The RMM module is used in dual resilient systems to improve the resilience of the whole system by passing traffic and status messages between units of the system when appropriate. The RMM module sits in the protocol stack between the User Part modules, such as ISUP or SCCP, and the MTP3 or M3UA modules.

The RMM module can be used to redirect outbound MTP traffic to the remote partner if the local route for the outbound traffic is unavailable.

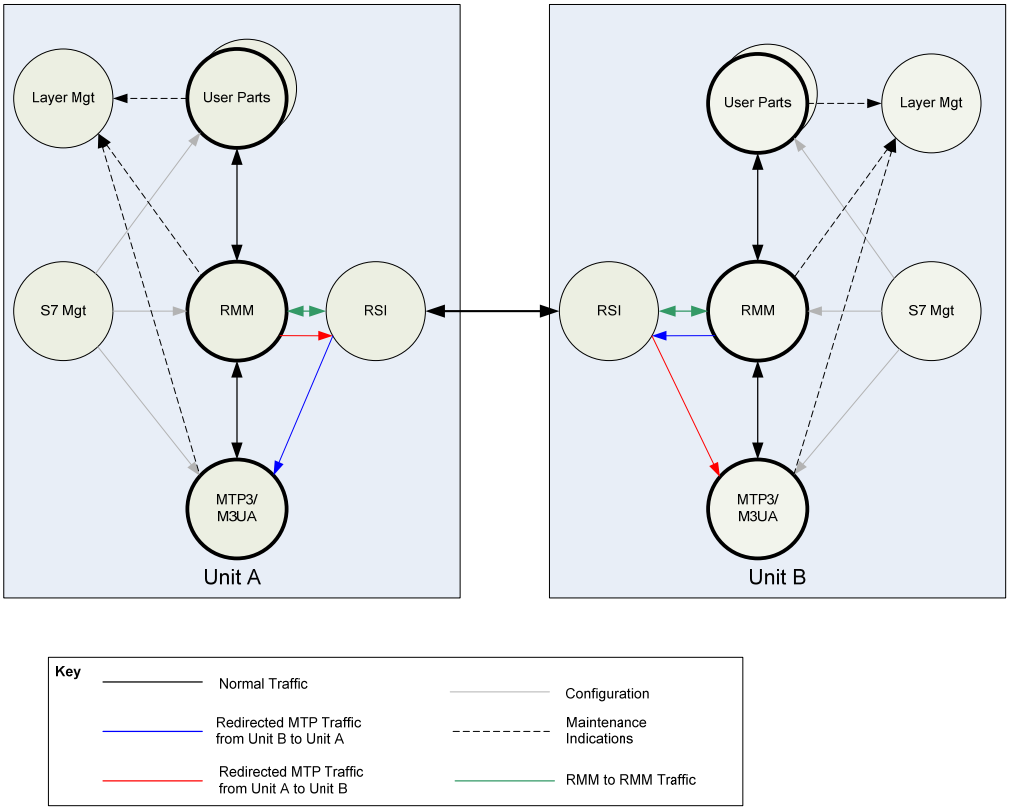
Inbound MTP traffic is automatically forwarded transparently to the user part protocols, which will forward the MTP message to the remote partner (via RSI).

The RMM modules monitor the MTP3 route status indications from the MTP3 or M3UA module and provide an aggregate status to the User Part modules based on the accessibility via either unit.

All modules should be present on both units of the system and the configuration of the MTP routes should be identical. All MTP routes must also be configured in RMM modules.

Figure 1. System Overview of RMM shows an example configuration.

Figure 1. System Overview of RMM



2.2 Module Overview

The RMM module is used to redirect outgoing user part traffic in a dual resilient system to an available MTP / M3UA route. Incoming traffic will be forwarded to the local user part protocol stacks. All routes to the remote destination must be configured in the RMM module, as well as in the layer 3 module (MTP3 / M3UA)

When receiving a MTP-Transfer Request (e.g., from ISUP), the RMM module will perform the following:

- If a local route is available, the MTP-Transfer request will be forwarded to the local MTP stack.
- If the local route is unavailable but the remote route is available, then the MTP-Transfer Request will be forwarded to the remote MTP stack (via RSI). The partner unit will update the local partner with the status of each destination on start-up or when a change of status is indicated.
- If the local and remote routes are unavailable, then the MTP-Transfer request will be forwarded to the local MTP stack and be discarded.

2.3 System.txt

When using RMM the system.txt file will need to contain appropriate configuration to allow the redirected traffic to be sent via the RSI module and to allow messages from the remote partner to be redirected to the correct modules.

For example:

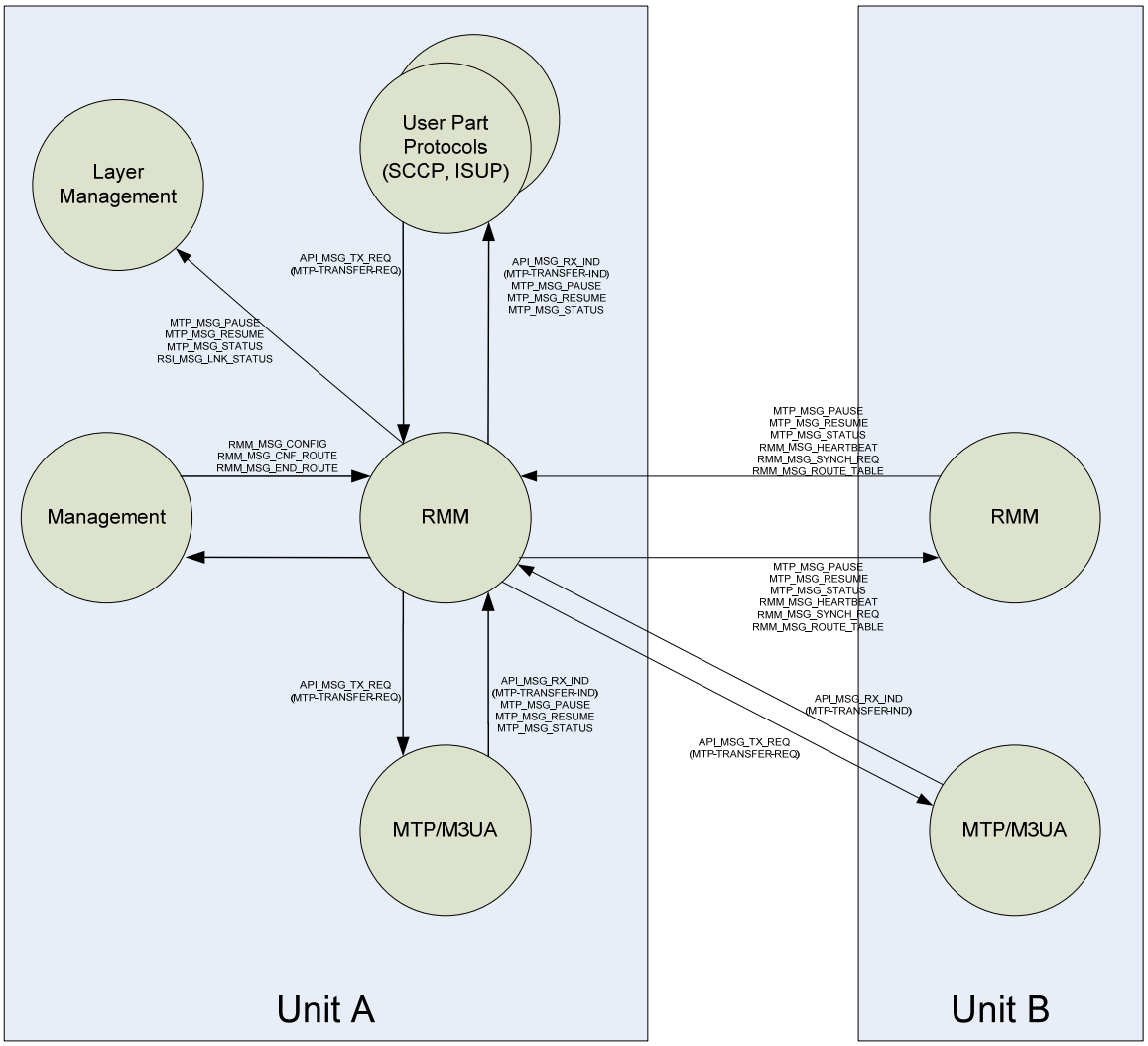
```
*********************************************************************
Example system.txt. RMM and stuff  (UNIT A)
*********************************************************************
* Modules running on the host:
*
LOCAL           0x00            * Timer Task
LOCAL           0xcf            * s7_mgt - Management/config task
LOCAL           0xef            * s7_log - Display and logging utility
LOCAL           0xd0            * SCTPD module
LOCAL           0xd1            * SCTP module
LOCAL           0xd2                * M3UA module
REDIRECT 0x22   0xd2            * Redirect MTP traffic to M3UA

LOCAL           0x33            * SCCP module
LOCAL           0x14            * TCAP module
LOCAL           0x25            * IS41 module
LOCAL           0x15            * MAP module
LOCAL           0x2d            * MTR or MTU module
LOCAL           0x32            * RMM
LOCAL           0xc0            * RSI
LOCAL           0xfd            * rsicmd Module Id
*
* Definitions unique to unit A:
*
REDIRECT 0x42 0x32           * RMM (from remote partner) to RMM
REDIRECT 0x62 0x22           * MTP-Transfer Request from remote partner

* Definitions to other unit:
*
REDIRECT 0x52 0xc0         * RMM messages to remote RMM
REDIRECT 0x72 0xc0         * MTP-Transfer requests to remote partner

NUM_MSGS 10000
*
* Now start-up all local tasks:
*
FORK_PROCESS    ..\..\..\bin\s7_log.exe -fss7.log
FORK_PROCESS    ..\..\..\bin\tim_nt.exe
FORK_PROCESS    ..\..\..\bin\tick_nt.exe
FORK_PROCESS    ..\..\..\bin\sctpd.exe
FORK_PROCESS    ..\..\..\bin\sctp.exe
FORK_PROCESS    ..\..\..\bin\m3ua_nt.exe
FORK_PROCESS    ..\..\..\bin\sccp_nt.exe
FORK_PROCESS    ..\..\..\bin\tcp_nt.exe -t
FORK_PROCESS    ..\..\..\bin\map_nt.exe -t
FORK_PROCESS    ..\..\..\bin\is41_nt.exe
FORK_PROCESS    ..\..\..\bin\rmm.exe -m0x32
FORK_PROCESS    ..\..\..\bin\s7_mgt.exe -d

FORK_PROCESS    ..\..\..\bin\rsi.exe -m0xc0 -r..\..\..\bin\rsi_lnk.exe -l1
FORK_PROCESS    ..\..\..\bin\rsicmd.exe 0 0x32 0 192.168.1.1 9000 0xc0
```

2.4 Module Configuration

The RMM module is configured for operating with a number of user part modules, which lie above the RMM module and with either M3UA or MTP configured as the transport protocol.

Management functionality should be provided by a management module, which configures the RMM module.

Figure 2. RMM Context Diagram



The configuration of the ISUP / SCCP module must be changed so that the MTP module Id is changed to the module Id of the RMM module (default 0x32).

The configuration of the MTP module must be changed so that the User Part configuration is changed whereby all user parts are now sent to the RMM module (default 0x32).

The RMM module configuration contains the remote RMM module Id; this should match the redirected value in the system.txt (in the example above this should be set 0x52).

Each of the routes configured in the RMM module has the following information:

User part Module Id – this should be configured as normal

Local MTP Module Id – this should be configured with the local MTP module Id (default 0x22)

Remote MTP Module – this should match the redirected value in the system.txt (in the example above, this should be set to 0x72).

3 Interface to system services

3.1 System Functions

In addition to the primitive interface and the management interface to the RMM module (which is described in Section 5: Message Reference), the RMM module requires a few basic system services to be supplied by the underlying operating system. This functionality is usually supplied by the appropriate Development package.

The following functions are required for inter-task communication:

Function	Description
GCT_send	Sends a message to another task.
GCT_receive	Accepts next message from input event queue, blocking the task if no message is ready.
GCT_grab	As for GCT_receive but not blocking if no message is ready.

The following functions are required for allocation of inter-task messages:

Function	Description
getm	Allocates a message.
relm	Releases a message.

3.2 Timer Operation

In order to provide internal implementation of the M3UA protocol timers, the RMM module needs to receive a periodic timer tick message. This is usually achieved using the Timer module, in which case the following messages are used by the RMM module:

Message	Description
KEEP_TIME	Issued by the module to initialize the timer services
TM_EXP	Issued by the timer module to notify of time-out.

For further information on these messages, see Related Doc [3]

4 External Message Definitions

4.1 MTP / M3UA

The Dialogic® DSI RMM module is intended to interface to the Dialogic® DSI MTP3 or M3UA Modules for sending traffic into the network or for receiving traffic from the network.

The following primitives are used:

Table 1. Primitives from RMM to MTP3 / M3UA

API_MSG_TX_REQ (0x728a)	MTP-Transfer Request

Table 2. Primitives from MTP3 / M3UA to RMM

API_MSG_RX_IND	MTP-Transfer Indication
MTP_MSG_PAUSE_IND	MTP-Pause Indication
MTP_MSG_RESUME_IND	MTP-Resume Indication
MTP_MSG_STATUS_IND	MTP-Status Indication

4.2 User Parts

The Dialogic® DSI RMM module is intended to interface to the Dialogic® DSI User Part module such as SCCP or ISUP.

The following primitives are used:

Table 3. Primitives from RMM to User Parts

API_MSG_RX_IND	MTP-Transfer Indication
MTP_MSG_PAUSE_IND	MTP-Pause Indication
MTP_MSG_RESUME_IND	MTP-Resume Indication
MTP_MSG_STATUS_IND	MTP-Status Indication

Table 4. Primitives from User Part to RMM

API_MSG_TX_REQ (0x728a)	MTP-Transfer Request

5 Message Reference

5.1 Non-Primitive Interface

In addition to the primitive interface for passing protocol messages and management messages between the M3UA module and the user modules, the RMM module supports a non-primitive interface for implementation specific functionality.

The non-primitive interface is used to by the user for configuration purposes and for communication with the remote the RMM module in the dual resilient system.

This section describes the formats of the messages used in the non-primitive interface.

When the RMM module returns a confirmation message containing a status value, the status will be one of the following:

Mnemonic	Value	Description
NONE	0x00	Success
RMME_BAD_ID	0x01	Inappropriate or invalid id in request message
RMME_BAD_MSG	0x05	Inappropriate or unrecognized message type.
RMME_BAD_PARAM	0x06	Invalid parameters contained in the message
RMME_NO_RESOURCES	0x07	Insufficient internal resources

5.1.1 RMM_MSG_CONFIG

Synopsis

Configure system parameters for the RMM module.

Message Format

Message Header	
Field Name	**Meaning**
type	RMM_MSG_CONFIG (0x7770)
id	0
src	Sending module id
dst	RMM module id
rsp_req	Used to request a confirmation
hclass	0
status	Non zero on error
len	16

Parameter Area		
Offset	**Size**	**Name**
0	2	options
2	1	remote_id
3	1	mgmt_id
4	2	rem_inst
6	10	reserved

Description

First message sent to the RMM module to start the module and configure general parameters.

The configuring module (e.g. S7_MGT) sending the message should request a response and check the returned status to verify that the configuration message has been accepted by the RMM module.

Parameters

options

Bit Number	Description
0-1	Point Code Format:- 00 – 14 bit point codes 01 – 24 bit point codes 10 – 16 bit point codes
2-15	Reserved - Should be set to zero

Therefore, when the RMM module is used in a system, only a single point code is available per instance of the RMM module.

mgmt_id

Module ID of the management module.

remote_id

Module ID of the remote RMM module.

rem_inst

Instance of the remote the RMM module should be set to zero.

reserved

Reserved space. Must be set to zero

5.1.2 RMM_MSG_CNF_ROUTE

Synopsis

Configure M3UA or MTP route for the RMM module, Each route is configured on a DPC and Service Indicator basis

Message Format

Message Header	
Field Name	**Meaning**
type	RMM_MSG_CNF_ROUTE (0x7771)
id	0
src	Sending module id
dst	RMM Module ID
rsp_req	Used to request a confirmation
hclass	0
status	Non-zero on error
len	16

Parameter Area		
Offset	**Size**	**Name**
0	4	DPC
4	1	Service Indicator
5	1	User Id
6	1	Local M3UA/MTP Id
7	1	Remote M3UA/MTP Id
8	2	Reserved
10	2	Reserved
12	2	Options
14	2	Reserved

Description

Sent to the RMM module to configure either an MTP or M3UA route, each route in the RMM must be unique

Parameters

DPC

Destination point code for the route.

Service Indicator

Service Indicator for the route (e.g., ISUP 5).

User Id
User part module Id for the route.

Local M3UA/MTP Id
Module Id for the local M3UA / MTP module.

Remote M3UA/MTP Id
Module Id for the remote M3UA / MTP module.

flags:-

Bit Number	Description
0	Default route for the service indicator
1-15	Reserved - Should be set to zero

reserved
Reserved space. Must be set to zero.

5.1.3 RMM_MSG_HEARTBEAT

Synopsis

Sent between the RMM modules on the local and remote partners

Message Format

Message Header	
Field Name	**Meaning**
type	RMM_MSG_HEARTBEAT (0x7778)
id	0
src	Sending module id
dst	RMM module id
rsp_req	Used to request a confirmation
hclass	0
status	Non zero on error
err_info	0
len	0

Description

Used to determine whether communication with the remote the RMM module is allowed. If no response is returned, then all routes via the remote partner will become unavailable

Parameters

None

5.1.4 RMM_MSG_SYNCH_REQ

Synopsis

Sent from the local to the remote RMM module to request synchronization of remote route status

Message Format

Message Header	
Field Name	**Meaning**
type	RMM_MSG_SYNCH_REQ (0x7779)
id	0
src	Local RMM module id
dst	Remote RMM module id
rsp_req	Used to request a confirmation
hclass	0
status	Non zero on error
len	0

Parameter Area		
Offset	**Size**	**Name**

Description

Used to request the status of all routes that are configured on the remote RMM process. This is sent when the RMM module starts to heartbeat.

Parameters

5.1.5 RMM_MSG_ROUTE_TABLE

Synopsis

Sent in response to the synchronization request; sends the status of all locally configured routes to the remote RMM module.

Message Format

Message Header	
Field Name	**Meaning**
type	RMM_MSG_ROUTE_TABLE (0x777a)
id	0
src	Local RMM module id
dst	Remote RMM module id
rsp_req	Used to request a confirmation
hclass	0
status	Non zero on error
err_info	0
len	see below

Parameter Area		
Offset	**Size**	**Name**
0	**4**	**options**
4	**2**	**Num routes**
6	**4**	**DPC (1st route)**
10	**1**	**Service Indicator (1st Route)**
11	**1**	**Status (1st Route)**
12	**4**	**DPC (2nd Route)**
16	**1**	**Service Indicator (2nd Route)**
17	**1**	**Status (2nd Route)**
:	**:**	
:	**:**	

Description

Used to update the remote RMM module with the status of all routes configured in the local RMM module. More than one Route Table message can be sent in response to the synchronization request. There is a maximum of 50 routes per message.

Parameters

options

Options set for the message; should be set to 0

num_routes

Number of route status indications that are reported in the current message

DPC

Destination Point code for the route

Service Indicator

Service Indicator for the route (e.g., SCCP 3)

Status

Current status of the route:-

Unavailable	-	0x00
Available	-	0x01