RC5000 MegaSwitch

System and Router Software

Installation and Operating Guide

Keywords:
RC5000 MegaSwitch, NMC, LCP, NCP.

Abstract:
This manual contains detailed information on how to installl RC5000 MegaSwitch node software in the NMC. It contains information on how to load, configure and operate an RC5000 MegaSwitch node.

Date:
November 1991

Author:
Inger Bohlbro, Torry Skak Gaarde, Niels Rose Hansen, Leif Obel Jepsen, Jens Kristian Kjærgård and Martin B. Klausen.

PN: 99001446

TABLE OF CONTENTS

Page

1. INTRODUCTION

This guide shall enable the operator to define, configure, install and operate an RcPAX RC5000 MegaSwitch node with:

- up to 2 Mbps router lines
- up to 1 Mbps access lines
- up to 1 Mbps access lines

With respect to the X.25 and X.75 access this installation and operating guide should be used in connection with:

- RcPAX Network System, DCE
  X.25 DCE and X.75 STE
  Installation and Operating Guide (ref. 18)

- RcPAX Network System
  Logical Internal Channel (LIC)
  Installation and Operating Guide (ref. 19)

- RcPAX Operating System
  PoolManager
  Installation and Operating Guide (ref. 20)

This guide is intended for RcPAX NMC operators.

Knowledge of the RC5000 MegaSwitch to the extent described in RcPAX, An Introduction (ref. 1), and RC5000 MegaSwitch General Description (ref. 2) is assumed.

The LCP operations and events implemented for the RC5000 MegaSwitch System and Router modules are described in ref. 5. The LCP operations for DCE, STE, LIC and PoolManager are described in ref. 21, 22, 23 and 24.

1.1 Scope of Guide

The scope of this guide is the RcPAX NMC operations (NMC database definitions, NMC online activities, LCP operations etc.), which are needed to configure, install and operate an RC5000 MegaSwitch node.

The RC5000 MegaSwitch node includes RcPAX modules for the RC5000 CPU as stated in chapter 3. This manual does not include instructions to install those modules in the NMC. Instructions to configure the necessary modules are included.

The instructions in this manual refer to the NMC8000, but should apply to NMC(s) in general.

1.2 History of Guide

Date	Init	Change description
September 89	-	First version
March 91	-	X.25 access included
September 91	-	X.75 access included
November 91	-	MAC242 included

1.3 Structure of Guide

Chapter 2: General Description
contains a general description of the software in an RC5000 MegaSwitch node.

Chapter 3: Requirements
describes the software, hardware and firmware requirements, which must be satisfied by the RcPAX modules for the RC5000 CPU, installed in the RC5000 MegaSwitch.

Chapter 4: Module Definition in NMC
describes all NMC operations, which must be performed in order to define RC5000 MegaSwitch System and Router modules in NMC, at a general NMC level.

Chapter 5: Module Definition in LSM/NCP Group
describes all NMC operations, which must be performed in order to define RC5000 MegaSwitch System and Router modules in an LSM/NCP group in NMC.

Chapter 6: Definition of an RC5000 MegaSwitch node in NMC
describes all NMC operations, which must be performed in order to define RC5000 MegaSwitch modules in a specific RcPAX node in NMC.

Chapter 7: Node Configuration
describes the configuration of RC5000 MegaSwitch System and Router modules. This includes a description of the initial request set for RC5000 MegaSwitch nodes.

Chapter 8: Normal Operation
describes how the RC5000 MegaSwitch System and Router modules should be operated in normal situations.

Chapter 9: Error Operation
describes how the RC5000 MegaSwitch System and Router modules should be operated in error situations.

Chapter 10: Guidelines for Error Reporting
describes which information could be collected, when an error is to be reported.

Appendix A: References
contains the references of the guide.

Appendix B: Exception Codes
contains a list of the codes of the exceptions, which the RC5000 MegaSwitch System and Router modules may send, and their meaning.

Appendix C: Trace Codes
contains a list of the traces, which the RC5000 MegaSwitch System and Router modules may send, and their meaning.

Appendix D: RC5000 autoloadfile
contains instructions for generation of autoloadfile for the RC5000 CPU part of the RC5000 MegaSwitch.

1.4 Using this Guide

This guide contains instructions for the following three tasks.

1.4.1 Installation of RC5000 MegaSwitch in an NMC

- Install the RC5000 MegaSwitch object code SW50050.p22 and megaboot.p22 from SW50050, SW51100.p22 and SW51100.btl from SW51100 and SW50512.p22, SW50512.btl and SW50512.d22 from SW50512 as described in the package descriptions.
- Create the autoloadfile as described in chapter 3 and appendix D.
- Check that paxplib1 and paxplib2 files fulfilling the requirements in chapter 3 are available.
- Follow the instructions in chapter 4.
- Follow the instructions in chapter 5 for each LSM/NCP Group.

1.4.2 Installation of an RC5000 MegaSwitch Node

- Check that the installation described in section 1.4.1 is carried out.
- Create suitable configuration requestset following the guidelines in section 7.2. The requestsets could usually be created by modifying the requestset described in section 7.3, which can be installed from SW50050, SW51100 and SW50512.
- Follow the instructions in chapter 6.

1.4.3 Operation of an RC5000 MegaSwitch Node

- Follow the instructions in chapter 8.
- In case of troubles use chapter 9 together with appendix B and C to solve the problem.
- If no solution is found follow the instructions in chapter 10 to report the error.

2. GENERAL DESCRIPTION

An RC5000 MegaSwitch consists of a number of boards each running their part of the software, see fig. 2.1.

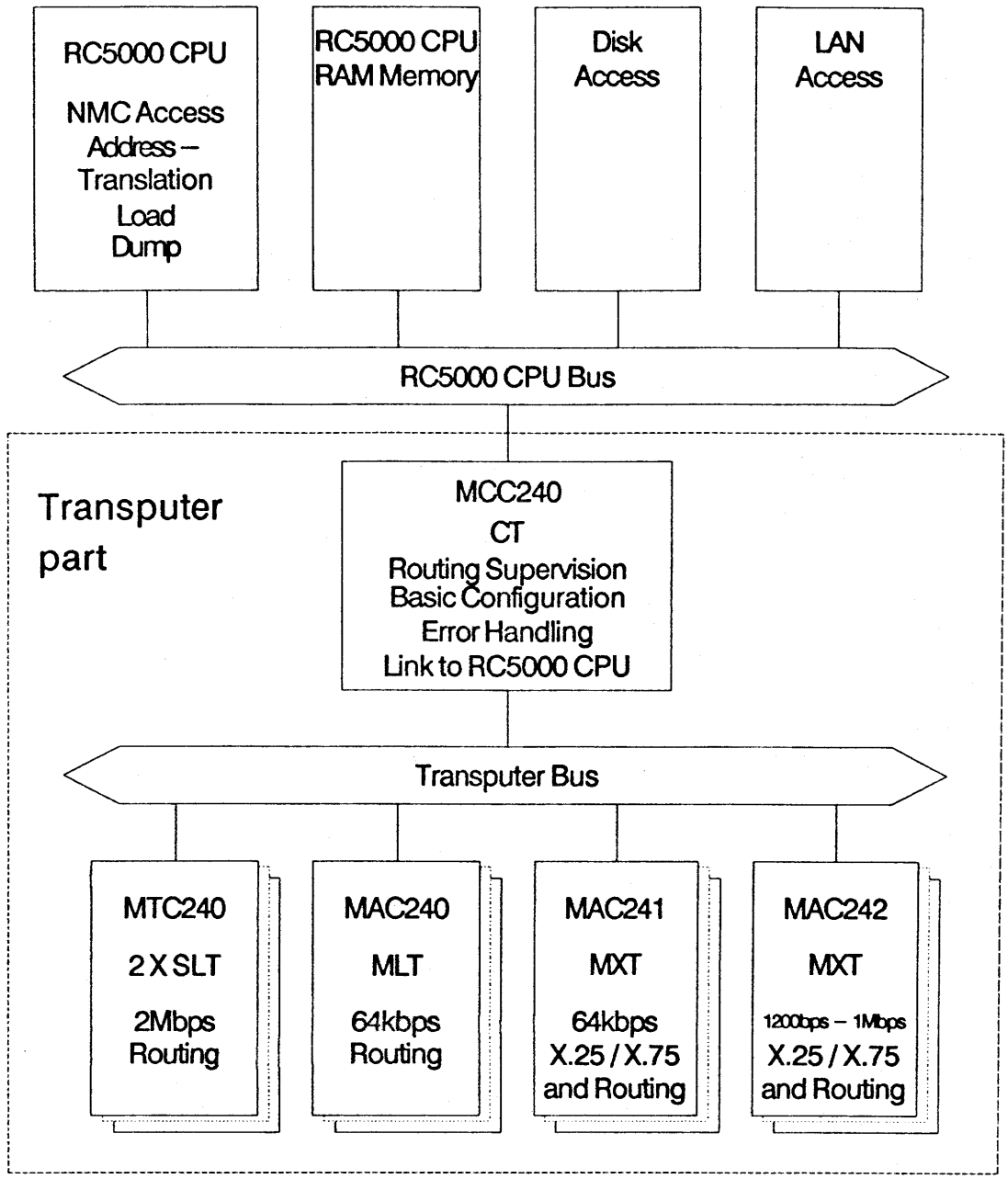

Fig. 2.1. RC5000 MegaSwitch

The MCC240, MTC240's, MAC240's, MAC241's and the MAC242's constitute the transputer part of a MegaSwitch. The transputers on each board are denoted:

CT	Central Transputer	(MCC240)
SLT	Single Router Line Transputer	(MTC240)
MLT	Multi Router Line Transputer	(MAC240)
MXT	X.25/X.75 and Router Line Transputer	(MAC241/242)

Fig. 2.2-2.6 show the software processes that are necessary in an RC5000 MegaSwitch. The processes implementing the LinkBus system and the PI-5 kernel are not shown. If MAC241/242 is used for Routing, the processes correspond to the processes on MAC240.

LinkBus System

The LinkBus system consists of several processes on the transputer part and takes care of the internal communication between the processes on the transputer part, both internal on one transputer and on the links between transputers. The LinkBus system guarantees that deadlock will not occur and treats bursts in a proper way.

MegaDriver takes care of the communication between processes on the RC5000 CPU and processes on the transputer part. MegaDriver communicates with processes in the LinkBus system on MCC240.

Process Control

On the RC5000 CPU Paxopsys takes care of creation and remove of processes.

On the transputer part MegaSys reads the hardware configuration and starts all processes except the X.25/X.75 access processes DCE, STE, LIC, PoolManager and MegaLink. After the creation of the other processes MegaSys is used for collecting statistics of the LinkBus system, reading information about the configuration and error handling. The processes created by MegaSys cannot be removed, instead the transputer part must be started all over.

On MAC241/242 Mopsys takes care of creation and remove of the X.25/X.75 access processes DCE, STE, LIC, PoolManager and MegaLink. This is accomplished by means of the RTP runtime environment (PI-5 kernel) running on MAC241/242.

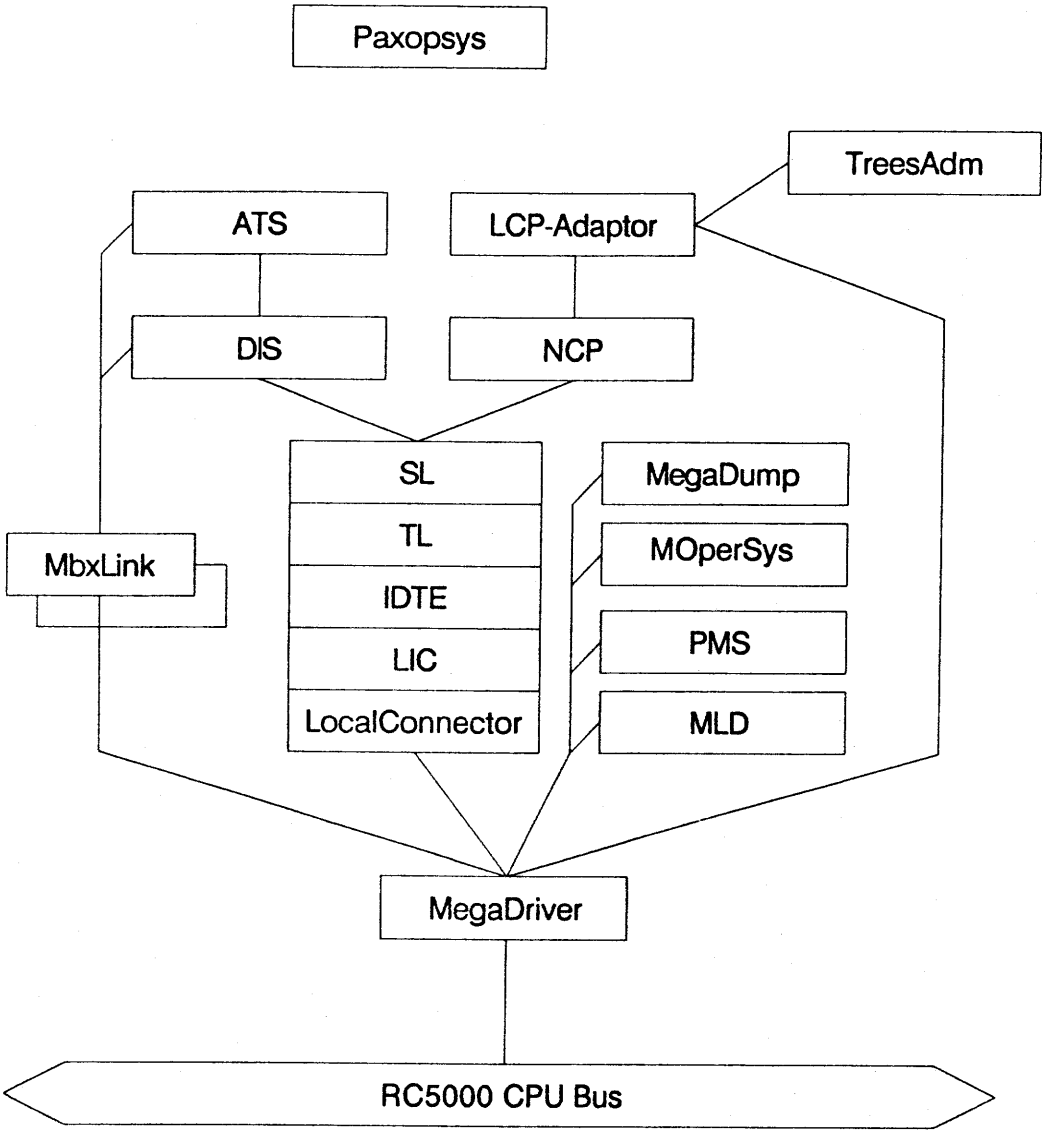

Fig. 2.2. Processes on RC5000 CPU

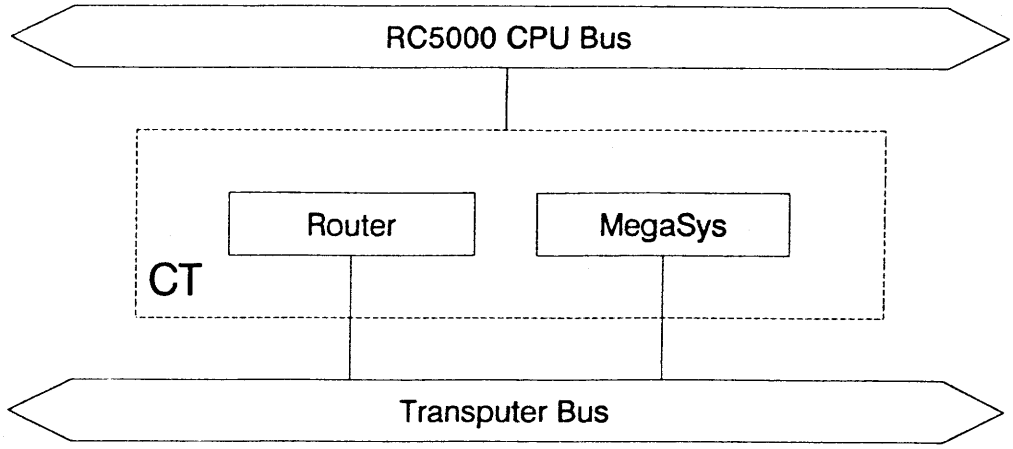

Fig. 2.3. Processes on MCC240

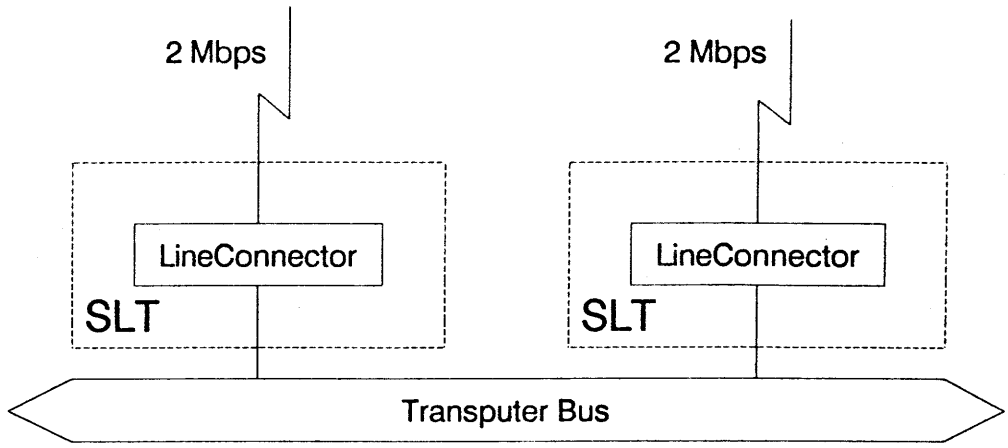

Fig. 2.4. Processes on MTC240

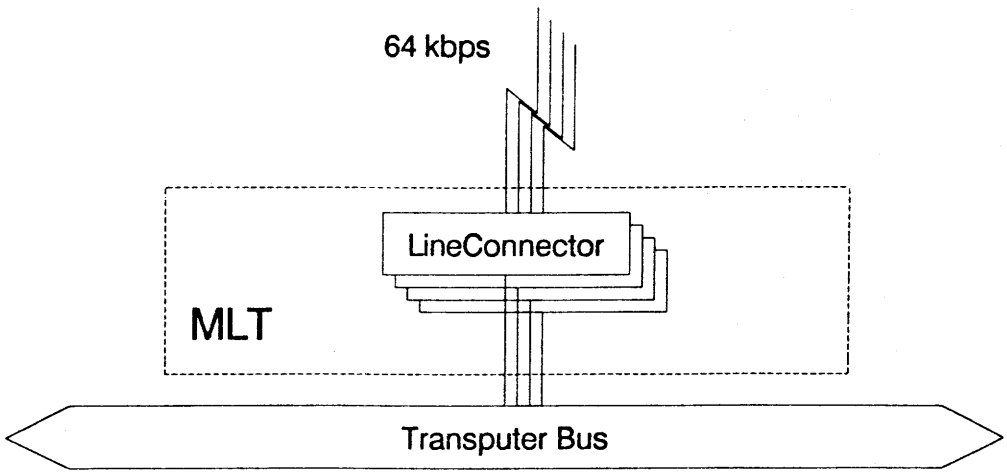

Fig. 2.5. Processes on MAC240

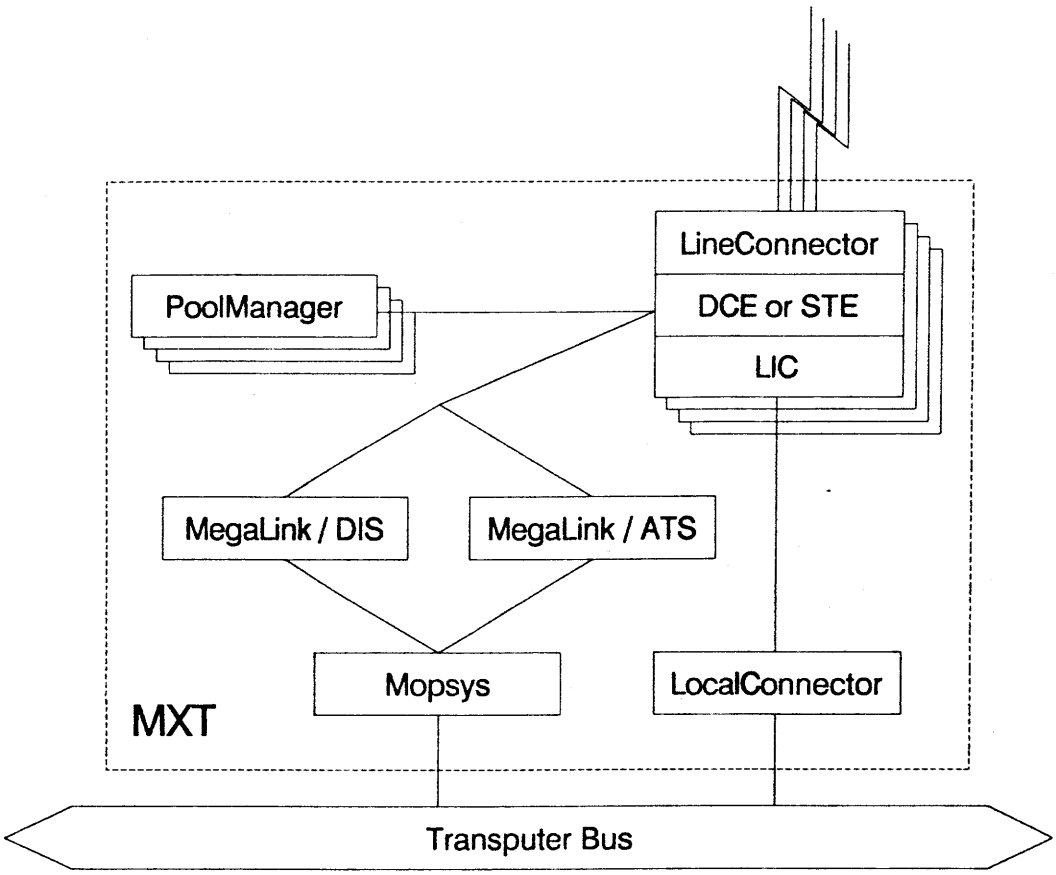

Fig. 2.6. Processes on MAC241, MAC242

Debug Console

The debug console on the RC5000 CPU is used for input to and output from processes.

Processes running on the RC5000 CPU can use the debug console directly.

Processes running on MAC241/242 or MCC240 can also use the debug console. This is accomplished by means of the PI-5 kernel running on MAC241/242 or MCC240 and the MOperSys process running on the RC5000 CPU.

Datagram Service

The Datagram Service is implemented by the Router process, the LineConnector processes controlling router lines and the LocalConnector processes.

The Router process and the LineConnector processes are the basic processes necessary for the RC5000 MegaSwitch to be a transit switching node in the RcPAX datagram network.

The Router process manages the routing tables for the node. The Router is also in charge of the synchronized network clock in the node.

There is a LineConnector for each line. The LineConnector is controlling an XPC chip, which performs protocol handling on the line. The LineConnector receives information from the routing tables from the Router so that it is able to route packets in the RcPAX datagram network. Besides routing packets a LineConnector may be used in testing the quality of the line and for loading the neighbour node or the node itself.

A packet from the datagram network bound for this node is sent to one of more LocalConnectors dependent on the destination extension for the packet. The LocalConnector sends the packet further to the destination extension for the packet. A packet, sent into the datagram network, is routed to the right LineConnector by the LocalConnector.

Two different implementations of the LocalConnector exist. The RC5000 CPU LocalConnector is running on the RC5000 CPU, while the transputer LocalConnector is running on the MAC241/242.

Access Service

The X.25 access and X.75 access services are implemented on MAC241/242. The DCE process implements X.25 layer 3, the STE process implements X.75 layer 3, and the LineConnector process implements X.25/X.75 layer 2. As for router lines there is one LineConnector process for each line. There is a one-to-one correspondence between DCE/STE, LineConnector and the physical line.

The DCE and the STE processes use the Connection Service made available by the corresponding LIC processes. The LIC processes use the Datagram Service made available by the LocalConnector.

The PoolManager manages buffers for DCE, STE, LineConnector, LIC and LocalConnector. Each PoolManager is able to handle one size of buffers corresponding to the defined packet size.

DIS and ATS Services

The DCE and the STE use both the Directory Information Service (DIS) and the Address Translation Service (ATS). These services are made available by the MegaLink, Mopsys and MbxLink processes constituting a link to the ATS and DIS processes on the RC5000 CPU.

The Address Translation Service is built on top of the Directory Information Service. The Directory Information System consists of a number of DIS processes in different nodes communicating by means of the session service.

Session Service

The session service is made available by means of the Session Layer (SL) using the Transport Service made available by the Transport Layer (TL). TL uses the internal access service made available by the Internal DTE (IDTE). IDTE uses the Connection Service made available by the Logical Internal Channel (LIC). LIC uses the Datagram Service made available by the LocalConnector.

Management Service

All MegaSwitch processes may be controlled and monitored from the Network Management Center (NMC). By means of the Session Service the NMC communicates with the Network Control Probe (NCP) on the RC5000 CPU.

The NCP can only communicate directly with processes on the RC5000 CPU. The LCP-Adaptor takes care of communication with processes on the transputer part. All LCP requests, LCP answers and LCP events to processes on the transputer part pass through the LCP-Adaptor.

Two types of LCPs exist on the transputer part, LCPs implemented in occam and LCPs implemented in Real Time Pascal.

The LCPs implemented in occam communicate directly with the LCP-Adaptor.

The LCPs implemented in Real Time Pascal communicate with Mopsys, which takes care of communication with the LCP-Adaptor. For these processes, Mopsys makes the same service available as the NCP does for processes on the RC5000 CPU. Mopsys incarnation 0 is simulated by LCP-Adaptor and is used to redirect create- and remove-operations from the NMC to the Mopsys process on the appropriate MAC241/242 board.

Furthermore, for all LCPs implemented in Real Time Pascal, the LCP-Adaptor converts data in requests, answers and events between the transputer representation and the RC5000 CPU representation based on ASN1 descriptions (parsetrees). The TreesAdm process handles the parsetrees.

MegaDump supervises in co-operation with MegaSys the transputer part and can in case of errors on some transputer make a dump on winchester. Furthermore, MegaDump may be directed to perform a restart of the whole RC5000 MegaSwitch node.

PMS and MLD are used in connection with load over a line of a node in the datagram network. PMS (Platform Mega Supervisor) is used, when the RC5000 MegaSwitch is platform for load of a neighbour node. MLD (Mega Load Driver) is created in the target RC5000 MegaSwitch during a down line load phase in order to handle the down line load protocol received from a platform node via an HDLC line.

Process Identification

The identification of the processes in a running RC5000 MegaSwitch is for most of the modules straight forward, i.e. they can be identified by the LCP Survey (LSUR) acitivty on the NMC.

There are, however, a number of exceptions:

- The DCE process has two LCP's, dce and x25account.
- The STE process has two LCP's, x75 and x75account.
- The LCP-Adaptor process has to LCP's, lcpadaptor and mopsys incarnation 0.
- The MLD process only exists when the node is down line loaded and has no LCP.

- The MOperSys and MegaLink processes are not real LCP's, i.e. only request 1 (create process) is defined. They cannot receive other requests or send events, and they are not shown by the LSUR activity.

RC5000 MegaSwitch System and Router Processes

This Installation and Operating Guide is concerned with the RC5000 MegaSwitch System and Router Processes, i.e.:

- LCP-Adaptor
- MegaDriver
- LocalConnector (mlocalc and mtlocalc)
- MegaDump
- PMS
- MLD
- MbxLink
- Router
- MegaSys
- LineConnector
- Mopsys
- MegaLink
- MOperSys
- TreesAdm

DCE, STE, LIC and PoolManager on MAC241/242 is further described in ref. 18, 19 and 20.

The rest of the processes are part of the RcPAX modules running on the RC5000 CPU. Information of these processes are part of the respective SW-packages in which they are included.

3. REQUIREMENTS

This chapter describes the requirements, which must be satisfied to install an RC5000 MegaSwitch node.

3.1 Software Requirements

This section describes the requirements, which must be satisfied by the RC5000 CPU Software.

The requirements depend on the type of MegaSwitch node:

- Transit node

- Access node
  - without Address Translation
  - with Address Translation

In the following a list of SW-packages is given, describing which plibs in the packages are required for which type of node. The node type may be:

- *All.*
  The package plibs is required for both transit nodes and access nodes.

- *Access.*
  The package plib is required for access nodes only.

- *ATS.*
  The package plib is required for access nodes using the Address Translation facility only.

Furthermore it is indicated which load file plib usually contains the required package plib.

SW2802 RcPAX Boot and Load System

package plib	node type	loadfile plib
blsplib	all	autoloadfile
wldplib	all	autoloadfile
paxsemplib	all	autoloadfile

SW50020 RC5000 Operating System

package plib	node type	loadfile plib
stdplib	all	autoloadfile
system3502	all	autoloadfile

SW50220 RC5000 File System

package plib	node type	loadfile plib
fs3502plib	all	autoloadfile
fsuserplib	all	autoloadfile
lf3502plib	all	autoloadfile
lookwdplib	all	plib1

SW51010 RcPAX Operating System

package plib	node type	loadfile plib
bemexcept	all	plib1
em00plib	all	plib1
em02plib	all	plib1
lcpuseplib	all	plib1
lintesplib	all	plib1
lookmemplib	all	plib1
ncp01plib	all	plib1
ncthplib	all	plib1
paxstdplib	all	plib1
s1plib	all	plib1
srsplib	all	plib1

SW51013 RcPAX Basic Switching System

package plib	node type	loadfile plib
idte	all	plib2
lic	all	plib2

SW51014 RcPAX ECMA Communication

package plib	node type	loadfile plib
ecmassiplib	access	plib2
slplib	all	plib2
tsplib	all	plib2

SW51016 RcPAX Name Server

package plib	node type	loadfile plib
atsplib	ATS	plib2
displib	access	plib2

The package descriptions for SW50050, SW51100 and SW50512 may specify further version constraints.

3.2 Hardware Requirements

The RC5000 MegaSwitch consists of a number of boards as described in ref. 7.

If the RC5000 MegaSwitch is loaded from winchester disc, boards for the winchester disc must be included.

If the RC5000 MegaSwitch will use Down Line Load, the autoload PROMs placed on the RC5000 CPU memory board must include the kind 'mega' (version 11.1 or later).

3.3 Network Requirements

The RC5000 MegaSwitch is used for transit traffic in networks consisting of other RC5000 MegaSwitch nodes and RC5000 Network Switch nodes. RC5000 Network Switch nodes which shall be able to communicate with the RC5000 MegaSwitch through router lines must include the revised version of the ROUTER modules with version no less than the version stated in the SW51100 package description.

4. MODULE DEFINITION IN NMC

This chapter describes all NMC operations, which must be performed in order to define the RC5000 MegaSwitch System and Router modules in NMC, at a general NMC level. It is assumed that the RcPAX modules described in section 3.1 are defined already. These definitions should only be performed once for an NMC.

4.1 EM/LCP Identity

4.1.1 EM/LCP Class Name

All the RC5000 MegaSwitch System and Router modules are defined with the class "paxnet".

4.1.2 EM/LCP Type

This section defines the type Name and Number of the RC5000 Mega-Switch System and Router modules to be defined in the NMC database by means of the NMC "LCP type" (TYP) online activity.

Process Name	LCP Name	LCP Type
LCP-Adaptor	lcpadaptor	114
MegaDump	mdump	125
MegaDriver	mdriver	116
MegaSys	msys	127
LocalConnector	mlocalc	119
LineConnector	mlinec	120
Router	mrouter	121
PMS	pms	143
TreesAdm	treesadm	124
Mopsys	mopsys	150
LocalConnector	mtlocalc	172
MbxLink	mbxlink	174
MegaLink	megalink	176

4.2 Creator Process Name

This section states the names of the creator process of the RC5000 MegaSwitch System and Router modules.

The modules to run on the RC5000 CPU are mainly created by PAX-OPSYS, while the modules to run on the transputer part of the RC5000 MegaSwitch are created by MegaSys or Mopsys, except MegaSys itself, which is created automatically when loaded by Mega-Driver.

Process Name	Creator Process Name
LCP-Adaptor	PAXOPSYS
MegaDump	PAXOPSYS
MegaDriver	PAXOPSYS
MegaSys	(MegaDriver)
LocalConnector (mlocalc)	PAXOPSYS
LineConnector	MegaSys
Router	MegaSys
PMS	PAXOPSYS
MLD	PAXOPSYS/BOOTS
TreesAdm	PAXOPSYS
Mopsys	MegaSys
LocalConnector (mtlocalc)	MegaSys
MbxLink	PAXOPSYS
MegaLink	Mopsys incarnation i *)
MOperSys	PAXOPSYS

*) Each MXT has one Mopsys process with incarnation number equal to the lowest connector number handled by the MXT.

4.3 Event Bit Mask Definition

This section defines the meaning of the single bits in the (old) event mask of the RC5000 MegaSwitch System and Router modules, i.e. which bits in the event mask correspond to which events (EVE).

The bits are counted from 1 to 16, with 1 as the right most bit (least significant bit).

None, as the bit number, means the event can not be ordered or counterordered by the event mask.

General

Bit	Event	Event Number
none	Snoop Event RTP	249
none	Snoop Event occam	250
none	Trace Event	253
none	Exception Event	254
none	Event Lost Event	255

MegaSys

Bit	Event	Event Number
8	Message Lost Event	63
9	Error Event	67
9	Parity	71

MegaDriver

Bit	Event	Event Number
none	CT Error Event	7

Router

Bit	Event	Event Number
2	Router Connection Up	51
3	Router Connection Down	55
4	Router Connection Stop	59
5	Router Connection Testing	63
6	Clock Inaccuracy	79
7	Remote Clock Inaccuracy	83
8	Range Vector Error Event	67
9	Report State	247
10	NNP CRC Error	71
11	End To End CRC Event	75
15	Measured Round Trip Delay	151
16	Load Request	31
none	Router-Router Event	195
none	Broadcast Event	199

LineConnector

Bit	Event	Event Number
2	HDLC Line Up	51
3	HDLC Line Down	55
4	HDLC Line Warning	59
5	Routing Warning	63
6	Level1 Change	67
7	Line Id Warning	97
9	Report State	247
10	NNP CRC Error	71
11	Linetest CRC Error	79
14	Linetest Error	103
15	CRC Threshold Event	75

LocalConnector (mlocalc, mtlocalc)

Bit	Event	Event Number
4	Extension Not Connected	23
5	Extension Disconnected	19
6	Extension Connected	15
none	Illegal Packet (mlocalc only)	55
none	Format Error	67
none	Report State	63

LCP-Adaptor

Bit	Event	Event Number
none	ASN1 Conversion Error	247

TreesAdm

No events.

Mopsys

Bit	Event	Event Number
none	Process Synchronize	7
none	Create MbxLink	11
none	Message Lost	15

MbxLink

Bit	Event	Event Number
none	Create Link	11
none	Message Lost	15

MegaLink

No events.

PMS

Bit	Event	Event Number
1	Autoload Request (NNP)	11
1	Autoload Request (X.25)	15
11	Data Packet Received	35
12	Finis Load	31
13	Break Load	27
14	Init Load	23
15	HDLC Line State Change	7

MegaDump

No events.

4.4 Initial Event Bit Mask at NMC Level

Currently definitions of event bit masks at NMC level is not supported by NMC8000.

If definitions are supported, the values should be equal to the default values of LCP operation 4: Old Set Event Mask. These default values are identical with the values specified at node and module incarnation level in section 6.2.

4.5 DES-file

The description of the LCP operations is included in the NMC database by installing the DES-files of the RC5000 MegaSwitch System and Router modules using the DESLOAD program.

5. MODULE DEFINITION IN LSM/NCP GROUP

This chapter describes all NMC operations, which must be performed in order to include the RC5000 MegaSwitch System and Router modules in an LSM/NCP group in NMC. These operations must be performed once for each LSM/NCP group, in which the RC5000 MegaSwitch modules should be defined. Definitions for the RcPAX modules for the RC5000 CPU described in section 3.1 must be included too.

Before you make the NMC definitions at LSM/NCP group level, which are described in this chapter, you must have made the NMC definitions for the modules at a general NMC level, as described in chapter 4.

5.1 Initial Event Bit Mask at LSM/NCP Group Level

This section defines the recommended value for the initial event bit mask of the RC5000 MegaSwitch System and Router modules at LSM/NCP group level.

These event bit masks are defined in the NMC database by means of the NMC "event mask" (MAS) online activity.

The same event bit masks as for the node and module incarnation level are recommended, i.e. the event bit masks described in section 6.2 are recommended for the LSM/NCP group level.

5.2 Treatment Category

This section defines the recommended values for the treatment categories and logging specifications of the events of the RC5000 System and Router modules.

The definitions must be made in the NMC database for each LSM/NCP group in which the RC5000 MegaSwitch modules should be installed, by means of the NMC "notification" (NOT) online activity.

INITIAL TRC and ACTUAL TRC are identical and should be defined as stated in TRC below. INITIAL LOG and ACTUAL LOG are all set to ON.

Definitions identified by all should be duplicated for the mrouter, mlinec, mlocalc, mtlocalc, msys, mdriver, lcpadaptor, pms, mopsys, mbxlink LCP.

LCP NAME: all

EVENT	DESCRIPTION	TRC
253	Trace Event	00000010 00000000 00000000
254	Exception Event	00000010 00000000 00000000
255	Events Lost Event	00000010 00000000 00000000

LCP NAME: mrouter

EVENT	DESCRIPTION	TRC
31	Load Request	00000010 00000000 00000000
51	Router Connection Up	10000000 00000000 00000000
55	Router Connection Down	10000000 00000000 00000000
59	Router Connection Stop	10000000 00000000 00000000
63	Router Connector Testing	10000000 00000000 00000000
67	Range Vector Error Event	10000000 00000000 00000000
71	NNP CRC Error	10000000 00000000 00000000
75	End To End CRC Event	10000000 00000000 00000000
79	Clock Inaccuracy	00000010 00000000 00000000
83	Remote Clock Inaccuracy	00000010 00000000 00000000
151	Measured Round Trip Delay	00000010 00000000 00000000
195	Router-Router Event	00000010 00000000 00000000
199	Broadcast Event	00000010 00000000 00000000
247	Report State	00000010 00000000 00000000
250	Snoop Event occam	00000010 00000000 00000000

LCP NAME: mlinec

EVENT	DESCRIPTION	TRC
51	HDLC Line Up	00000010 00000000 00000000
55	HDLC Line Down	00000010 00000000 00000000
59	HDLC Line Warning	00000010 00000000 00000000
63	Routing Warning	00000010 00000000 00000000
67	Level1 Change	00000010 00000000 00000000
71	NNP CRC Error	00000010 00000000 00000000
75	CRC Threshold Event	00000010 00000000 00000000
79	Linetest CRC Error	00000010 00000000 00000000
97	Line Id warning	00000010 00000000 00000000
103	Linetest Error	00000010 00000000 00000000
247	Report State	00000010 00000000 00000000
249	Snoop Event RTP	00000010 00000000 00000000
250	Snoop Event occam	00000010 00000000 00000000

LCP NAME: mlocalc

EVENT	DESCRIPTION	TRC
15	Extension Connected	00000010 00000000 00000000
19	Extension Disconnected	00000010 00000000 00000000
23	Extension Not Connected	00000010 00000000 00000000
55	Illegal Packet	10000000 00000000 00000000
63	Report State	00000010 00000000 00000000
67	Format Error	10000000 00000000 00000000
249	Snoop Event RTP	00000010 00000000 00000000

LCP NAME: mtlocalc

EVENT	DESCRIPTION	TRC
15	Extension Connected	00000010 00000000 00000000
19	Extension Disconnected	00000010 00000000 00000000
23	Extension Not Connected	00000010 00000000 00000000
63	Report State	00000010 00000000 00000000
67	Format Error	10000000 00000000 00000000
249	Snoop Event occam	00000010 00000000 00000000
250	Snoop Event occam	00000010 00000000 00000000

LCP NAME: msys

EVENT	DESCRIPTION	TRC
63	Message Lost Event	00000010 00000000 00000000
67	Error Event	00000010 00000000 00000000
71	Parity	00000010 00000000 00000000
250	Snoop Event occam	00000010 00000000 00000000

LCP NAME: mdriver

EVENT	DESCRIPTION	TRC
7	CT Error Event	10000000 00000000 00000000

LCP NAME: lcpadaptor

EVENT	DESCRIPTION	TRC
247	ASN1 Conversion Error	00000010 00000000 00000000
249	Snoop Event RTP	00000010 00000000 00000000

LCP NAME: pms

EVENT	DESCRIPTION	TRC
7	HDLC Line Event	00000010 00000000 00000000
11	NNP Load Request	00000010 00000000 00000000
15	X.25 Load Request	00000010 00000000 00000000
23	Init Load Received	00000010 00000000 00000000
27	Break Load Received	00000010 00000000 00000000
31	Finis Load Received	00000010 00000000 00000000
35	Data Received	00000010 00000000 00000000
249	Snoop Event RTP	00000010 00000000 00000000

LCP NAME: mopsys

EVENT	DESCRIPTION	TRC
7	Process Synchronize	00000010 00000000 00000000
11	Create MbxLink	00000010 00000000 00000000
15	Message Lost	00000010 00000000 00000000

LCP NAME: mbxlink

EVENT	DESCRIPTION	
11	Create Link	00000010 00000000 00000000
15	Message Lost	00000010 00000000 00000000

6. DEFINITION OF AN RC5000 MEGASWITCH NODE IN NMC

This chapter describes all NMC operations, which must be performed in order to define an RC5000 MegaSwitch as an RcPAX node in NMC. The NCP group in which the node shall be included must be created and the definitions described in chapter 5 must be performed in advance.

Initially the address and name of the node (NCP name) is defined by means of NMC (NCP) activity.

6.1 EM/LCP Incarnation Definition

The incarnations listed in section 6.2 must be defined.

6.2 Initial Event Bit Mask

Through the NMC "LCP incarnation" (INC) online activity the following incarnations and event masks must be defined:

LCP NAME	LCP INC	EVENT MASK	BITS
ncp	0	10000010 00000001	1,9,15
paxopsys	0	00000000 00000000	
lic	0	00000010 10000000	8,10
idte	0	00000000 00000000	
ts	0	00000000 00000000	
sl	0	00000000 00000000	
mdriver	0	00000000 00000000	
lcpadaptor	0	00000000 00000000	
mlocalc	1	00000000 00111000	4,5,6
mdump	0	00000000 00000000	
msys	0	00000001 10000000	8,9
mrouter		11000010 10111110	2,3,4,5,6,8,10,15,16
mlinec **)	j	01000010 01011000	4,5,7,10,15
pms	0	01111000 00000001	1,12,13,14,15
scc	82	00000000 11111111	1,2,3,4,5,6,7,8
ats	82	00000000 11111111	1,2,3,4,5,6,7,8
mbxlink	0	00000000 00000000	
mopsys *)	i	00000000 00000000	
dce *)	i,i+1,i+2,i+3	01100000 00000110	2,3,14,15
lic *)	i,i+1,i+2,i+3	00000010 10000000	8,10
poolman *)	i,i+1,i+2,i+3	00000000 00000000	
mtlocalc *)	i	00000000 00111000	4,5,6

*) For each MAC241 board, where i is the lowest connector number on the board.

**) For each j which is a connector number on an MTC240, MAC240 and MAC241 board.

6.3 Load File Description

If the RC5000 MegaSwitch node shall be Down Line Loaded a Load File Description must be defined by the NMC (LFD) Load file Description activity. The Load File Description must also be defined if the WD-FILL operation in the NMC online activity INCO is used. The following items should be defined:

ELEMENT TYPE	ELEMENT NAME
file	*autoloadfile*
file	plib1
file	confireqset
file	plib2
file	psw50050
file	psw51100
file	bsw51100
file	psw50512
file	bsw50512
file	dsw50512

The generation and contents of *autoloadfile* is described in appendix D.

RcPAX modules for the RC5000 CPU described in section 3.1 must be included in plib1 and plib2.

The contents of confireqset are the data-program formatted request sets described in section 7.3.

The files psw50050, psw51100, bsw51100, psw50512, bsw50512 and dsw50512 should be installed at the NMC from SW50050, SW51100 and SW50512. In these software packages the files are named sw50050.p22, sw51100.p22, sw51100.btl, sw50512.p22, sw50512.btl and sw50512.d22.

6.4 Load Component Description

The new RC5000 MegaSwitch must be included in the Down Line Load control by the NMC online activity (LCD) Load Component Description. This definition must be done although Down Line Load of the RC5000 MegaSwitch is not used, because the definition is required by the INCO online activity for the NMC initiated initialization of the node.

7. NODE CONFIGURATION

This chapter contains information related to the configuration of the RC5000 MegaSwitch node. It is described how the modules can be configured by means of LCP operations.

The LCP operations for the RC5000 MegaSwitch System and Router modules are described in the RC5000 MegaSwitch System and Router Software, EM Reference Manual (ref. 5).

The LCP operations described in this section should be examined, and the operations needed to obtain the desired configuration should be included in the initial request set of the node by means of the NMC online activity "Build Request Set" (BRS). Section 7.3 contains an example configuration. In order to be included as an RC5000 loadfile the requestset must be formatted as 'DATA'-programs.

7.1 Create and Start LCP Operations

The processes MegaDriver, MegaDump, LCP-Adaptor, RC5000 CPU LocalConnector and MbxLink are created by Paxopsys through issuing (-,r,1): Create and Start Module addressed to the corresponding non-existing LCP, i.e. to create MegaDriver (mdriver,r,1): Create and Start Module is sent. The default values specified in the DES description of the Create and Start operations can be used.

The process MOperSys is created by Paxopsys through issuing (paxopsys,r,60): Run family 2 process.

MegaSys starts operation, when the transputer code has been loaded to the CT. Router, LineConnectors, Mopsys and transputer LocalConnectors are created automatically when (msys,r,200): Create Router is executed.

The RC5000 CPU LocalConnector must not be created before the Create Router operation is executed by MegaSys.

The processes MegaLink, DCE, STE and PoolManager are created by the corresponding Mopsys through issuing (-,r,1): Create and start module addressed to the corresponding LCP. These processes must not be created before the Create Router operation is executed by MegaSys.

A number of MbxLink processes may be needed dependent on the use of the ATS- and DIS-Services. Each MbxLink process is able to handle 255 outstanding buffers at the ATS- and DIS-modules. Each DCE or STE contributes to the number of outstanding buffers with:

- 12 buffers for DIS.
- a configuration dependent number of buffers for ATS. It is the "number of ATS messages" in (DCE,r,252) or (x75,r,252). A value of 32 is recommended.

This implies that the requirement is satisfied if one MbxLink process is created for each MXT with 4 DCE's.

The identification of the MbxLink process by the DCE or the STE is determined by the LinkBus address of the MbxLink process given to the corresponding MegaLink processes.

7.2 Initialize LCP Operations

This section identifies LCP operations used to initialize the RC5000 MegaSwitch System and Router modules.

The initial event ordering is valid from the time the module is created or started until other events are ordered from the NMC, for example the events corresponding to the event bit mask defined at the NMC.

For the MegaSys, Router, LineConnector and transputer LocalConnector processes, the initial event ordering are determined by LCP operations to the LCP-Adaptor (cf. section 7.2.1).

For the other RC5000 MegaSwitch System and Router modules, the initial event ordering is as follows:

mdriver	: 7, 253, 254, 255
mlocalc	: 23, 55, 63 67, 253, 254, 255
mdump	: 253, 254, 255
mbxlink	: 253, 254, 255
mopsys	: 253, 254, 255
lcpadaptor	: 253, 254, 255
pms	: 7, 11, 15, 23, 27, 31, 39, 253, 254, 255
treesadm	: 253, 254, 255

7.2.1 LCP-Adaptor

1) The connection between LinkBus address and LCP ident must be defined for occam processes receiving LCP operations through the LCP-Adaptor. This is done by (lcpadaptor,r,100): Define LCP-idents.

   The operation is applied for the processes MegaSys, Router, LineConnector and transputer LocalConnector. The following set of parameters should be used:

msys	: 127, 125, 125, 0, 0
mrouter	: 121, 0, 0, 0, 88
mlinec	: 120, 2, 53, 2, 0
mtlocalc	: 172, 64, 115, 2, 0

2) The modules initial event ordering for occam LCPs should be defined by means of (lcpadaptor,r,104): Order Initial Events.

The operation should be applied for the processes MegaSys, Router, LineConnector and transputer LocalConnector. The following set of parameters are recommended:

```
msys       : 127,  63,  67,  71, 253, 254, 255
mrouter    : 121,  31,  51,  55,  59,  63,  67, 71,
             151, 253, 254, 255
mlinec     : 120,  59,  63,  67,  71,  75, 253,
             254, 255
mtlocalc   : 172,  23,  67, 253, 254, 255
```

These LCP operations should be sent before any of the MegaSys, Router, LineConnector or transputer LocalConnector processes are created.

7.2.2 MegaDump

1) A dump catalogue can be defined by (mdump,r,164): Set Dump Path, or the default catalogue /megadumpcat can be used.

2) The dump and restart mode can be changed by (mdump,r,184): Set Dump Mode. However, the default parameters should do.

3) The result of the hardware test is examined through the (mdump,r,176): Wait Hardware Test. The hardware test is initiated automatically at power-up and when MegaDriver is removed. Furthermore, hardware test can be initiated through (mdriver,r,8) Master Clear.

   The amount of time from the hardware test is initiated until it is finished depends on the number and type of boards. If the hardware test is not finished within 30 seconds, (mdump,r,176) returns an answer with illegal state as status. In this case it is necessary to include a delay (srs,r,8) just before (mdump,r,176).

   If the result of (mdump,r,176) is not ok and restart is requested (cf. 2 above), a restart is performed.

   Wait Hardware Test should be issued before (mdriver,r,16): CT Reset Link described in section 7.2.3.

7.2.3 MegaDriver

1) The CT is reset by (mdriver,r,16): CT Reset Link.

2) Transputer code is loaded onto the CT through (mdriver,r,40): CT Load. CT Load must be issued four times to load code to CT, SLT, MLT and MXT. By convention the 'DATA' programs specified as request data are named ctbtl, sltbtl, mltbtl and mxtbtl.

   If there are no MXT (transit node without access), emptybtl should be used instead of mxtbtl.

3) Transfer of LinkBus messages is started through (mdriver,r,48): Start Receive. The correspondence between LinkBus addresses and LCP ident must be defined (cf. section 7.2.1) in advance.

7.2.4 MegaSys

The initialization operations of the MegaSys process can be divided in three groups. First operations to bypass malfunctional hardware, second operations to control the allocation of buffers and third, the only mandatory operation, the creation of the Router process.

1) The configuration of an RC5000 MegaSwitch node can be logically reduced by excluding boards, transputers or lines and the line numbering can be changed by means of (msys,r,184): Exclude Board, (msys,r,188): Exclude Transputer, (msys,r,192): Exclude Line and (msys,r,196): Swop Line.

2) The size of the memory area used for buffers can be changed by (msys,r,204): Set LinkBus Size and (msys,r,208): Set LineConnector Size. Suitable default values are inserted by MegaSys. These default values can be obtained through (msys,r,189): Get LinkBus Size and (msys,r,193): Get LineConnector Size.

   On the MXT the specified memory area is only used when the LineConnector is used for Router lines.

   The amount of free bufferspace on an SLT, an MLT and an MXT is stated in the package description for SW51100 and SW50512.

   The following relations should be fulfilled:

$$L0 + L1 + L2 + L3 + TRx + REx + TCx \leq SLTmax$$

   and

$$L0 + L1 + L2 + L3 + TRx + REx + TCx + CLx + TRy + REy + TCy + CLy + TRz + REz + TCz + CLz + TRu + REu + TCu + CLu \leq MLTmax$$

   and

$$L0 + L1 + L2 + L3 + TRx + REx + TCx + CLx + TRy + REy + TCy + CLy + TRz + REz + TCz + CLz + TRu + REu + TCu + CLu + RTPmem = MXTmax$$

where

x,y,z,u are the connector number of the connectors on the board

Li is the size in bytes of the LinkBusBuffers for link i of the transputer.

TRi is the size in bytes of the transmitbuffer area of connector i.

REi is the size in bytes of the receivebufferarea of connector i.

TCi is the size in bytes of the trace area of connector i.

CLi is the size in bytes of the LinkBusBuffer used by connector i.

SLTmax and MLTmax are the values stated in the SW51100 package description.

MXTmax is stated in the SW50512 package description.

RTPmem is the amount of free memory available for the RTP system. This part of the memory is primarily used for process stacks and heap. On a running system the amount of free RTPmem can be obtained through (mopsys,r,81): Get Free Memory.

The number of transmitbuffers can be calculated by the formula:

$$T_{buffers} = TR \text{ DIV } (300 + 8)$$

The number of receivebuffers can be calculated by the formula:

$$R_{buffers} = (RE - 20) \text{ DIV } (296 + 10)$$

The formulas assume that the maximum packet length, Packet.max.-Length is 291, and that the maximum length communicated through the LinkBus system, Message.max.Length is 300. 296 should be used when calculating $R_{buffers}$ because a 4 bytes LinkBus header should be included giving 295, which is extended to 296 because it should be divideable by 4 (32-bit).

3) If X.25 access is used, the DNIC of the node must be defined through (msys,r,216): Set DNIC.

   As an end of the configuration phase (mrouter,r,200): Create Router must be sent. The parameters must contain the network address of the node and the register identification of the RC5000 CPU.

The operations mentioned in 1 and 2 may only be sent before the Create Router operation, and operations cannot be sent to the Router, LineConnector, transputer LocalConnector and Mopsys processes before the Create Router operation.

7.2.5 Router

The initialization operations of the Router process can be divided in four groups. First, parameters used when estimating node and line resistance, may be configured (1-3 below). Second, the synchronized network clock may be configured (4-6 and 10 below). Third, a number of lines may be reserved for use in the routing of packets (7-8 below). Finally, for statistical reasons, the extension characteristics may be set for some extensions (9 below).

1) The line resistance used when line resistance is based on the linespeed, can be configured for each node by means of (mrouter,r,52): Set Speed To Resistance. The default value of the parameters might be appropriate, thus the operation is not mandatory.

2) If line resistance is to be based on the actual delay over the line, the map transforming delay to line resistance must be defined, by means of (mrouter,r,56): Set Delay To Resistance. Experiments have to be carried out, before recommendations for the parameters can be specified.

3) The node resistance can be set by (mrouter,r,60): Modify Routing. By the same operation the action on receiving NNP packets with CRC errors can be configured. If the default values (0 for node resistance and reset the HDLC connection on NNP CRC errors) will do, then the LCP operation can be omitted in the initialization.

4) One or more nodes in the network can be configured as a master clock, the other nodes are configured as slave clocks accepting the time received from at least one of the master clocks. This is done by (mrouter,r,72): Control Network Clock. It is recommended to configure one or two master clocks in each part of the network with the same net address. For the slave clocks the default configuration might do.

5) At each node the local derivation from GMT (Greenwich Mean Time) can be configured by (mrouter,r,92): Set GMT Derivation. If the network is not widespread, then the default local derivation 0 might be used for all nodes.

6) To all nodes, which are configured by sending LCP operations from an NMC, the (mrouter,r,88): Set Date and Time should be included in the initialization. The operation is sent with byte-count 0 (using the timestamp in the SP-header to initialize the clock).

7) For each line, the default parameters used when starting the line can be configured by (mrouter,r,40): Set Default HDLC Parameters, (mrouter,r,44): Set Default Line Resistance Parameter and (mrouter,r,48): Set Default Timer Parameters. Usually operations with different parameters are issued for MTC and MAC boards.

   Is is recommended to set the HDLC.ADDR parameter to 0 (negotiation). The parameter must never be set to the same role (DCE/DTE) at the two stations on the line.

   It is recommended to set the SEQ.FORMAT parameter to 0 (negotiation). The parameter must never be set to different values (8/128) at the two stations on the line.

   The value for the T1 timer must take the frame size, linespeed and propagation delay into consideration. T1 must be as large, as 3 packets may be transmitted and two propagation delays included before expiration of T1.

   In setting the parameters N2 and T1 the timeout at the LIC should be taken into account. When the line is damaged, it may take at least N2*T1 msec before the HDLC connection is broken, and further TEMP.DOWN msec before the router connection is broken. In order to route the packet retransmitted by the LIC on an operationel route, the connections should be broken before the retransmission is initiated.

   At lines where the remote station is controlled by a COM204 controller, the FLAGS parameter must be set to at least 2.

   In setting the line resistance parameters, it should be noted that line resistance based on linespeed, has proven to be useful for terrestrial links. For satellite links it may be necessary to estimate the line resistance by measuring the delay. If so, the delay to resistance table must be defined, too.

   Recommended values for the parameters are the default values given in LCP operation (mrouter,r,40), (mrouter,r,44) and (mrouter,r,48).

8) During the initialization one or more lines must be reserved and started as a router line. A line is started by means of (mrouter,r,12): Start LineConnector. The operation can be sent without parameters, in which case the parameters configured in 7) is used, or the operation can be sent with parameters, in which case the same recommendation as for 7) is given.

9) To get appropriate statistics the STAT.TYPE may be set differently for different kinds of extensions through (mrouter,r,20): Set Extension Characteristics. This is, however, only necessary if knowledge about the amount of data from specific extensions, for statistical reasons, is wanted.

10) To ensure that the network clock is defined from a neighbour router, before any access modules are created, one or more (mrouter,r,89): Wait Clock Defined operations may be issued. This operation should only be issued after the router lines are started.

7.2.6 LocalConnector

For mtlocalc and mlocalc LocalConnectors default parameter values may be used, i.e. no initializing operations are necessary. A number of operations may, however, be considered:

1) A package storage (PS) is used to save packets that cannot immediately be delivered to users. The amount of time until a package in the package storage will be dropped can be set by (mlocalc/mtlocalc,r,24): Set PS-timer.

2) A user package storage (UPS) is used to save fragments of packets received from the datagram network, i.e. stored in UPS until the complete packet is received. The amount of time until a package in the user package storage will be dropped can be set by (mlocalc/mtlocalc,r,36): Set UPS-timer.

3) The maximum number of packets within a given period that may be routed into the datagram network may be set by (mlocalc/mtlocalc,r,28): Set Flow Control to Other Connectors.

7.2.7 LineConnector

1) For line monitoring it may be considered to send (mlinec,r,56): Set CRC Threshold and (mlinec,r,60): Set Level1 Parameters. However, the default values at the LineConnector may do.

2) As a mean to control the configuration of the network, the lines may be assigned an identification by (mlinec,r,20): Set Line Identification.

3) If two MAC242 are interconnected without use of modems and a non-default linespeed is wanted, (mlinec,r,64): Set Trm Linespeed must be used to set the linespeed.

If applied, the operations should be sent to each incarnation of the LineConnector process.

7.2.8 TreesAdm

The parsetrees used for X.25 and X.75 access are initialized by means of (treesadm,r,44): Load parsetrees.

7.2.9 Mopsys

There are no initializing operations for Mopsys.

7.2.10 MbxLink

There are no initializing operations for MbxLink.

7.2.11 MegaLink

There are no initializing operations for MegaLink.

7.3 Configuration Example

An RC5000 MegaSwitch node with 4 X.25/X.75 access lines (MAC241 or MAC242, connector 2, 3, 4 and 5), 4 64 kbps router lines (MAC240, connector 6, 7, 8 and 9) and 2 2 Mbps router lines (MTC240, connector 10 and 11) can be configured by the following nested structure of request sets, which are sent to the desired processes by srs.

```
confisrs
    srsmega (rootreqset)
        zmstart
        zmtrees
        zmload
        zmrouter
        ztower0
        srsdis82s
        zmacc0205
        zmconn0609
        zmconn1011
        zmwaitc
        srsfinis
```

RC8000 requestset saves of the zm... requestsets are delivered in the appropriate software packages.

The file zmconn2.r8z contains the requestsets zmconnxxyy, where yy = xx + 1 and xx = 2,4,..,32 for initializing the two router lines on MTC240 dependent on the connector number.

The file zmconn4.r8z contains the requests zmconnxxyy, where yy = xx + 3 and xx = 2,6,..,30 for initializing the 4 router lines on MAC240, MAC241 or MAC242 dependent on the connector number.

The file zmacc.r8z contains the requestsets zmacc0205, zmacc0609, zmacc1013, zmaccc1417, zmacc0407, zmacc0811, zmacc1215 and zmacc1619 for initializing MbxLink and MegaLink processes to be used with X.25 and X.75 access on MAC241/242 dependent on connector number.

7.3.1 confisrs

This requestset is executed by confi version 2, and creates the srs process and requests srs to send rootreqset. This is further described in ref. 23.

confisrs contains the following requests:

(confifunc,r,4)	Set Facility
(paxopsys,r,181)	Create & Start Srs
(srs,r,100)	Send Request Set (rootreqset)

7.3.2 srsmega (rootreqset)

This requestset consists of calls to several subrequestsets. The requestset name in the NMC can be selected freely, but the generated dataprogram should be named rootreqset in order to keep confisrs stable.

srsmega contains the following requests:

(srs,r,60)	Set Trace Filter
(srs,r,100)	Send Request Set (zmstart)
(srs,r,100)	Send Request Set (zmx25)
(srs,r,100)	Send Request Set (zmload)
(srs,r.100)	Send Request Set (zmrouter)
(ncp,r,196)	Set Restart NMC Address
(srs,r,100)	Send Request Set (srstower0)
(srs,r,100)	Send Request Set (srsdis82s)
(srs,r,100)	Send Request Set (zmacc0205) *)
(srs,r,100)	Send Request Set (zmconn0609)
(srs,r,100)	Send Request Set (zmconn1011)
(srs,r,100)	Send Request Sest (srsfinis)

*) If configuration of DCE, STE and PoolManager is included in the rootreqset, this part of the configuration must be included after zmacc and as described in ref, 18, 19 and 20.

7.3.3 zmstart

zmstart contains the following requests:

(mdriver,r,1)	Run MegaDriver
(mdump,r,1)	Run MegaDump
(lcpadaptor,r,1)	Run LCP-Adaptor
(paxopsys,r,60)	Run Family 2 Process (MOperSys)
(mdriver,r,8)	Masterclear
(mdump,r,184)	Set Dump Mode
(lcpadaptor,r,100)	Define LCP Idents (127)
(lcpadaptor,r,100)	Define LCP Idents (121)
(lcpadaptor,r,100)	Define LCP Idents (120)
(lcpadaptor,r,100)	Define LCP Idents (172)
(lcpadaptor,r,104)	Order Initial Events (121)
(lcpadaptor,r,104)	Order Initial Events (120)
(lcpadaptor,r,104)	Order Initial Events (127)
(lcpadaptor,r,104)	Order Initial Events (172)

7.3.4 zmtrees/zmtreesun

These requestsets are not necessary for transit nodes without access.

zmtrees does not load and unload the parsetrees, whereas zmtreesun does.

zmtrees/zmtreesun contains the following requests:

(srs,r,36)	Conditional load program (dsw50512) *)
(treesadm,r,1)	Create and Start TreesAdm **)
(treesadm,r,44)	Load parsetrees (atspd22, dispd22)
(treesadm,r,44)	Load parsetrees (mopsysd22, megalinkd22)
(treesadm,r,44)	Load parsetrees (poolmand22, licd22, dced22, x25accntd22, x75d22, x75accntd22)

*) Included in zmtreesun only.
**) Create operation with automatic unload of parsetrees included in zmtreesun only.

7.3.5 zmload/zmloadun

zmload does not load and unload the transputer object programs, whereas zmloadun does.

(srs,r,36)	Conditional load program (bsw51100) **)
(srs,r,36)	Conditional load program (bsw50512) **)
(mdump,r,170)	Wait Hardware Test
(mdriver,r,16)	Reset Link
(mdriver,r,40)	Load (ctbtl)
(mdriver,r,40)	Load (sltbtl)
(mdriver,r,40)	Load (mltbtl)
(mdriver,r,40)	Load (mxtbtl) *)
(mdriver,r,48)	Start Receive
(paxopsys,r,92)	Unload program (ctblt) **)
(paxopsys,r,92)	Unload program (sltbtl) **)
(paxopsys,r,92)	Unload program (mtlbtl) **)
(paxopsys,r,92)	Unload program (emptybtl) **)
(paxopsys,r,92)	Unload program (mxtbtl) *) **)

*) emptybtl should be used for transit nodes without access.
**) included in zmloadun only.

7.3.6 zmrouter

zmrouter contains the following requests:

(srs,r,28)	Modify Next Request
(msys,r,216)	Set DNIC
(srs,r,24)	Modify Next Request (addr & id)
(msys,r,200)	Create Router
(mlocalc,r,1)	Run mlocalc
(srs,r,8)	Delaying (8)
(mrouter,r,40)	Set Default HDLC (MAC boards)
(mrouter,r,40)	Set Default HDLC (2 Mbps lines)
(mrouter,r,44)	Set Default Line Resistance

7.3.7 ztower0

Creates communication modules running on the RC5000 CPU to allow communication with NMS.

ztower0 contains the following requests:

(paxopsys,r,8)	Create and Start Module (ncth)
(idte,r,1)	Start IDTE
(srs,r,16)	Modify Next Request (nuid)
(paxopsys,r,58)	Start ts
(srs,r,s16)	Modify Next Request (nuid)
(paxopsys,r,62)	Start sl
(srs,r,8)	Delaying (12)
(ncp,r,12)	Include NCP at ecma sl

7.3.8 srsdis82s

This requestset initializes the ATS and DIS services. It is not necessary for transit nodes without access.

srsdis82s contains the following requests:

(scc,r,1)	Create and Start DIS
(ats,r,1)	Create and Start ATS
(scc,r,60)	Set parameters
(scc,r,4)	Set Event Mask
(scc,r,172)	Create Cache

7.3.9 zmacc0205

This requestset initializes links for access to ATS and DIS from MAC241. It is not necessary for transit nodes without access.

zmacc0205 contains the following requests:

(mbxlink,r,1)	Create and Start mbxlink
(megalink,r,1)	Run MegaLink (DIS)
(megalink,r,1)	Run MegaLink (ATS)

7.3.10 zmconn0609

This requestset starts connector 6, 7, 8 and 9.

zmconn0609 contains the following requests:

(mrouter,r,12)	Start LineConnector (6)
(mrouter,r,12)	Start LineConnector (7)
(mrouter,r,12)	Start LineConnector (8)
(mrouter,r,12)	Start LineConnector (9)

Bytecount is set to 2 to use the default parameters.

7.3.11 zmconn1011

This requestset starts connector 10 and 11.

zmconn1011 contains the following requests:

(mrouter,r,12)	Start LineConnector (10)
(mrouter,r,13)	Start LineConnector (11)

Bytecount is set to 2 to use the default parameters.

7.3.12 zmwaitc

This requestset waits up to 120 seconds for the network clock to be initialized.

zmwaitc contains the following requests:

(mrouter,r,89)	Wait Clock Defined
(mrouter,r,89)	Wait Clock Defined
(mrouter,r,89)	Wait Clock Defined

7.3.13 srsfinis

Displays node address on RC5000 CPU debug console.

srsfinis contains the following requests:

(paxopsys,r,116)	Send Adam Message
(srs,r,8)	Delaying (10)
(paxopsys,r,116)	Send Adam Message
(srs,r,16)	Modify Next Request (nuid)
(srs,r,64)	Display Text
(ncp,r,16)	Set Poll State

8. NORMAL OPERATION

This chapter describes how the RC5000 MegaSwitch System and Router modules should be operated in normal situations after configuration and installation.

The LCP operations and their parameters are described in the RC5000 MegaSwitch System and Router Software, EM Refereance Manual (ref. 5).

8.1 Line Survey

A survey of all lines in a node can be obtained by using the NMC8000 or PAXMAN linestate (lst) activity or issuing the (mrouter,r,49): Get Connector Survey.

The current state and application of the lines are shown. For RC5000 MegaSwitch, all lines in the node are shown, not only the router lines.

8.2 Router Line

Router lines can be monitored, started and stopped by the NMC8000 or PAXMAN linestate (lst) activity (ref. 25 and 26).

8.3 X.25 and X.75 Access Lines

A number of NMC8000 and PAXMAN activities may be used to handle X.25 and X.75 access lines, see ref. 25 and 26.

Configuration of X.25 access lines may be handled by the NMC8000 DCE description activity (dce) and the initialize DCE activity (idce).

A survey of active X.25 or X.75 lines can be displayed by the NMC8000 and the PAXMAN X.25/X.75 overview activity (xov).

A survey of the two endpoints of the X.25 and the X.75 connections of a node can be displayed by the NMC8000 VC Overview activity (vco).

Statistics of an HDLC line, an interval of logical channels and a single logical channel can be displayed by the NMC8000 X25/X75 State activity (xsta).

With respect to statistics of an HDLC line, the fields that are available for lines controlled by LineConnector do not correspond exactly to the fields displayed. The following fields are available:

- received
- xmitted
- rec RNR

- timeout
- rec error

The fields not available are displayed with the value 0. All available statistic fields can be retrieved by (mlinec,r,14): Get HDLC Statistics. These include also fields not displayed by the XSTA HDLC line statistic activity.

The NMC8000 X25/X75 Management activity (xman) gives facilities to open and close HDLC lines, restart X25/X75, and reset and clear logical channels.

8.4 Linetest

The LineConnector process can generate traffic and test the content of the received packets, thereby testing the quality of the line and encircling line errors.

Usually the connectors at both ends of a line must be configured and operated. The connectors can be RC5000 MegaSwitch LineConnector processes or RC5000 Network Switch linetest processes, or one of each. If loop back is set on the line only one connector is needed.

Linetests can be controlled by the PAXMAN linetest (ltst) activity (ref. 26).

From the NMC8000, linetests is for RC5000 Network Switch connectors controlled by the linetest (ltst) activity, while RC5000 MegaSwitch connectors must be controlled by LCP operations as described in the following.

Actions

1) If an RC5000 Network Switch connector is involved in the linetest, the RC5000 Network Switch linetest must be set to use an alternating HDLC address instead of the symmetric HDLC address which is otherwise used by default. This must be done before the NMC8000 linetest activity can be used. It is done by (linetest,r,92): Start Traffic (described in ref. 25).

2) Dereserve the line by means of (mlinec,r,16): Stop Line to the mlinec incarnation, or preferably dereserve the line at the module which currently has the reservation, e.g. the Router, the DCE or the STE.

3) If loop back is wanted the loop relay in any of the ends of the line can be set by (mlinec,r,100): Set Loop Mode.

4) Configure the linetest by means of (mlinec,r,104): Set Linetest Mode. The operator must be careful in setting the modes. The modesetting in the two connectors must be coordinated.

   TYPE must be set to the same value in the two connectors. For loop back, only TEST type may be used. For traffic between RC5000 Network Switch and RC5000 MegaSwitch, only I type may be used.

   The following combinations of TRAFFIC in the two connectors can be useful:

source1	------>	sink
source1	<----->	source1
source2	<----->	mirror

   CHECK.UP.ON should be set to the same value in the two connectors. If checks are performed LENGTH and PATTERN should be set to the same value in the two connectors, too.

5) Start the linetest by (mlinec,r,112): Start Linetest. Usually the default parameters can be used, i.e. bytecount can be set to 0.

6) During the linetest the following LCP operations are useful to give an impression of the test. (mlinec,r,102 (106)): Get (and Clear) Linetest Statistic and (mlinec,r,14 (18)): Get (and Clear) HDLC Statistics.

7) In case of errors on the line (mlinec,r,101): Get Level1 State, (mlinec,r,54 (58)): Get (and Clear) 2M Level1 Statistics and (mlinec,r,129): Get Event List can be useful in order to diagnostic the error.

8) In case of errors on packets discovered during the linetest, (mlinec,r,109): Get Linetest Error, can be used to read details of the error. Additionally, (mlinec,e,103): Linetest Error and (mlinec,e,79): Linetest CRC Error can be used to read the actual content of erroneous packets.

9) The linetest is stopped by means of (mlinec,r,116): Stop Linetest.

10) If a loop back is set, it should be removed by (mlinec,r,100): Set Loop Mode.

11) The current configuration of the linetest can be read by (mlinec,r,105): Get Linetest Survey.

8.5 Network Clock

A synchronized network clock is maintained in RcPAX. The clock is controlled by LCP operations to the Router process.

The clock in a node is not defined after a rerun of the node. The clock will automatically be defined as soon as a router line to a neighbour node, which has a defined clock, is running.

Actions

1) The master clocks must periodically be adjusted, by (mrouter,r,88): Set Date and Time. The operation can be sent with bytecount 0 (using the timestamp in the SP header to initialize the clock).

2) The synchronization of different master clocks can be monitored by (mrouter,e,79): Clock Inaccuracy and (mrouter,e,83): Remote Clock Inaccuracy.

3) The current time can be read by (mrouter,r,81): Get Date and Time and the current configuration read by (mrouter,r,73): Get Network Clock Data.

8.6 Statistic Type

For each extension the stat type inserted in the packets received from the extension and counted in statistics throughout the network, can be configured.

At default, the RC5000 MegaSwitch uses the following correspondence between extension numbers and stat types. When the conventions for extension numbers are used, this information may be used to distinguish the amount of traffic for various purposes.

STAT.TYPE	EXTENSION.NUMBER	CONVENTION
1	all other extensions	
2	87-92, 1400-1499, NNP Load Request	all load modules
3	NNP Line Information	Router handshake protocol
4	NNP Test	test of line quality
5	NNP Distance Vector	exchange of routing information
6	NNP Network Clock	synchronization of clock
7	NNP Broadcast	broadcast through Router
8	0	Router (i.e. Router-Router packets)
9	901, 97-99, 77	NCP
10	982-987	SCC
12	83-86	Test Abonnent
15	2-54	access lines on an RC5000 MegaSwitch

Actions

1) The stat type for a specific extension can be controlled by means of (mrouter,r,29): Set Extension Characteristics to the Router process.

2) The statistics on the stat type can be read at the Router, for all packets routed in the node, by means of the LCP operations, 30, 34, 38, 42, 46, 50, 54 and 58. The packets passing a specific LineConnector can be read by means of the LCP operations 30, 34, 38 and 42 to LineConnector. The packets passing a specific LocalConnector can be read by means of operations 46, 50, 54 and 58 to the LocalConnector.

8.7 End to End CRC

For each extension it can be configured whether CRC should be computed for packets received from this extension, and the CRC checked, when the packets are delivered to the destination extension.

At default the End to End CRC is not computed.

Actions

1) The end-to-end CRC calculation and the stat type are controlled by means of (mrouter,r,20): Set Extension Characteristics to the Router process.

2) The occurrence of end-to-end CRC errors can be monitored by (mlocalc/mtlocalc,e,67): Format Error from the LocalConnector.

8.8 RC5000 as Load Platform

The RC5000 MegaSwitch can be used as Load Platform in DLL (Down Line Load) of both RC5000 MegaSwitch and RC5000 Network Switch nodes.

The RC5000 MegaSwitch connector used for DLL is excluded from the normal routing and is instead controlled by the PMS module of the RC5000 MegaSwitch.

The PMS is equivalent in operation with the RC5000 CPU PLS module, and communicates with an MLS using the datagram service.

DLL of both RC5000 MegaSwitch nodes and RC5000 Network Switch nodes (DLL targets) from both RC5000 MegaSwitch nodes and RC5000 Network Switch nodes (DLL platforms) is managed by the NMC8000 inco activity and by the PAXMAN configuration activity (cona).

8.9 Dumpfiles

In order to be able to dump the transputer part of the RC5000 MegaSwitch a suitable set of dumpfiles must be created through the (mdump,r,180): Create File. The operations should eventually be preceded by the (mdump,r,164): Set Dump Path to define the dump catalog.

It is recommended to define a dumpfile to each type of transputer. One for CT (type = 1), SLT (type = 2), MLT (type = 3) and if X.25/X.75 access is used one for MXT (type = 4). The size and key parameters can be found in the appropriate package description.

The dumpfiles are named dumplog, dumpct, dumpslt, dumpmlt, dumpmxt. If more dumpfiles for the same transputer type are wanted, the dumpno parameter can be increased (default 0) in which case the

sequence of dumpfiles is scanned to find an empty file. If dumpno <> 0, dumpno is appended to the filename. Additionally a logfile is recommended. The logfile is created by (mdump,r,180): Create File with type = 0.

Whenever a dump has been taken and appropriate actions as described in chapter 9 are finished, the dumpfiles should be reestablished. Information to the logfile are appended so a number of dumps can be taken before the logfile needs to be reestablished.

9. ERROR OPERATION

This chapter describes how the RC5000 MegaSwitch System and Router processes should be operated in error situations after configuration and installation.

RC5000 MegaSwitch errors can be divided into the following categories:

1) *Exceptions on RC5000 CPU*

These exceptions are caused by processes running on the RC5000 CPU.

On the debug console an RTP exception code and a stacktrace of the failing process are displayed.

If possible, an exception event is sent to the NMC.

Normally, it will be necessary to restart the node (NMC8000 INCO activity rerun; 'run s' command to opsys at the debug console).

For all MegaSwitch System and Router modules the exception codes and their meaning are described in appendix B. Specific actions to solve or bypass the exception are also described in appendix B.

The information when reporting an exception should include the exception code, the stacktrace, LCP idents and possible preceding logged events.

2) *Soft exceptions on the transputer part*

These exceptions are caused by processes running on the transputer part. Only the involved process stops.

On the debug console an RTP exception code and a stacktrace of the failing process are displayed.

The following processes will try to generate an exception event: DCE, STE, LIC and PoolManager.

If the exception is caused by DCE, STE, LIC or PoolManager, the processes must be removed and created again. (The NMC8000 idce activity can handle DCE and LIC).

If the exception is caused by other processes, the node must be restarted (NMC8000 INCO activity rerun, or 'run s' command to opsys at debug console).

Exceptions from DCE, LIC and PoolManager are described in ref. 18, 19 and 20.

Exceptions from MegaSwitch System and Router processes are described in appendix B. Specific actions to solve or bypass the exception are also described in appendix B.

The information when reporting an exception should include the exception code, the stacktrace, LCP idents and possible preceding logged events.

3) *Hard exceptions on the transputer part*

These are exceptions caused by processes on the transputer part where the transputer stops and sets the ERROR flag. In these situations the whole transputer part will stop.

If the stopped transputer is an SLT, MLT or MXT, the following are displayed at the debug console:

- RTP exception message and a stacktrace of the failing process from the CT process
- Message from MegaDump, eventually with an errorcode of the transputer in error.

and, if possible, the following events are sent to the NMC:

- (msys,e,36) Error Event from MegaSys
- (mdriver,e,36) CT Event Error from MegaDriver

If the stopped transputer is the CT, only the message from MegaDump is displayed at the debug console.

If configured suitably, MegaDump makes a dump of the the stopped transputer for debugging purposes. Furthermore, MegaDump will usually be configured to restart the node (cf. section 7.2.3). If the restart succeeds, essential information from the dump can be retrieved through the LCP operation (mdump,r,168) 'Lookup File' to MegaDump.

For all MegaSwitch System and Router processes the error codes and their meaning are described in appendix B.

Specific actions to solve or bypass the exception are also described in appendix B.

The information when reporting an exception should include the exception message, the stacktrace, possible errorcodes, LCP idents and possible preceding logged events. If no errorcode is present the dumpfile could be required.

4) *Traces*

Traces may be generated both by processes on the RC5000 CPU and by processes on the transputer part.

The trace is sent as a trace event to the NMC. In some cases also a trace message is displayed at the debug console.

Usually no direct action is taken upon reception of a trace.

For all MegaSwitch System and Router processes the trace codes and their meaning are described in appendix C. Specific actions to be taken are also described in appendix C.

For DCE, STE, LIC and PoolManager the trace codes are described in ref. 18, 19 and 20.

9.1 Handling of Serious Events

This section describes the action to be taken on serious events generated by the MegaSwitch System and Router modules, i.e. events which requires immediate action or increased attention by the operator.

The events and their parameters are described in the System and Router Software, LCP Manual (ref. 5).

9.1.1 MegaSys

Message Lost

1) If Message Lost Events are generated repeatedly, then the buffer configuration should be changed. The LinkBus buffers dropping messages should be assigned more memory. See section 7.2.

Error Event

1) This is a most serious event, the execution of programs are stopped on one or more transputers. See if any error codes are written on the debug console.

2) After restart, read the information from the dump files by means of (mdump,r,168): Lookup File.

Parity Event

1) A parity error on a MAC241 is detected and execution of programs are stopped in the transputer part of the RC5000 MegaSwitch. The ERROR flag of the transputer is usually set, too, causing an Error Event, too.

9.1.2 MegaDriver

CT Error

The same action as for Error Event from MegaSys.

9.1.3 Router

Load Request

1) This is a most serious event as a neighbour node has stopped execution, and requests for autoload. The load might be started using this node as a platform (see section 8.6).

Router Connection Up, Router Connection Down

1) The generation of these events should be monitored. If the events are generated repeatedly, the cause might be found by reading (mlinec,r,129): Get Event List.

Router Connection Stop

1) The current reservation of the line can be read by (mrouter,r,49): Get Connector Survey.

2) In case of CRC errors the line can be tested by using the line-test facility.

3) The line is not used as a router line until (mrouter,r,12): Start LineConnector is sent.

Router Connection Testing

1) Action should be taken if the line is not on purpose reserved for test (by a command to the neighbour Router).

2) By (mrouter,r,49): Get Connector Survey it can be seen whether the line connects two LineConnectors in the same node, which is indicated by generating this event.

Range Vector Error Event

1) This error should be reported to the maintenance.

2) Important information to follow the error report is the class RC5000 Network Switch, RC5000 MegaSwitch of the neighbour Router, and the version number of both Routers.

NNP CRC Error

1) This is a serious event as data is corrupted. The line should be stopped and tested by the linetest facility.

Clock Inaccuracy, Remote Clock Inaccuracy

1) The master clocks (ACTUAL.MASTER and NEIGHBOUR.MASTER) should be adjusted by (mrouter,r,88): Set Date and Time.

2) If Clock Inaccuracy is generated too often, the ACCURACY parameter should be configured larger by (mrouter,r,72): Control Network Clock.

9.1.4 LineConnector

HDLC Line Up

1) No action is to be taken, but the event should not be generated repeatedly without notice.

   For cause 2, check whether the remote station indicates a cause for resetting the connection.

HDLC Line Down

1) The action to be taken is dependent on the parameters. Check the level1 state. If this is ok, use the cause to diagnostic the kind of trouble. Causes in the range 1-29 are used to indicate troubles at the HDLC level, causes in the range 30-39 indicate troubles with the content of packets, while causes in the range 40-49 is used when the state change is initiated by LCP operations.

   1: Lack of receive buffers at the remote station,
   4,8: Check whether the remote station indicates any cause.
   6: Read the content of the received FRMR by means of (mlinec,r,149): Get LineConnector State.
   7: Might be an XID frame, check whether the remote station indicates any cause for resetting the connection.
   9: Check the configuration of the HDLC parameter N1 at both ends, the parameter is read by means of (mlinec,r,41): Get HDLC Configuration.
   10: Error in the acknowledgement of frames, possible error at one of the stations.
   11,19: Check the level1 state.
   23: Check whether the remote station indicates a cause for sending this frame.

30: The quality of the line should be examined.
31: If generated repeatedly then possibly a software error.
33: If generated repeatedly then change the configuration of the LinkBus.
34: The quality of the line should be examined.
35: Possibly a software error.
36: Check whether the user process indicates a cause.
37: Check the configuration of the HDLC parameter N1, the parameter is read by means of (mlinec,r,41): Get HDLC Configuration.
38: Possibly a software error.
39: Not compatible configuration of the Router or LineConnector versions at the two stations.
40-43: No action, caused by intervention from NMC.

HDLC Line Warning

1) The actual loops and parameter settings can be read by means of (mlinec,r,101): Get Level1 State and (mlinec,r,41): Get HDLC Configuration.

2) Remove the loop or start the LineConnectors with correct parameter settings (cf. section 7.2.7 part 7).

Routing Warning

1) Read the detailed routing information by means of (mlinec,r,37): Get Routing Data.

2) Warning 1 indicates the Router or LineConnector versions at the two neighbour nodes are incompatible.
Warning 2 generated repeatedly, and warning 4 indicates software errros.
Warning 5 indicates the data is mirrored (e.g. linetest mirror at the other end) or the line connects two LineConnectors at the same node.

Level1 Change

1) Control the level1 by (mlinec,r,101): Get Level1 State and (for 2 Mbps lines) (mlinec,r,54 (58): Get (and Clear) Level1 Statistic.

2) If necessary, stop the line and test it by means of the linetest facility.

NNP CRC Error

1) This is a serious event as data is corrupted. The line should be stopped and tested by the linetest facility.

CRC Threshold Event

1) Read the HDLC stat by means of (mlinec,r,14(18)): Get (and Clear) HDLC Statistics.

2) Same action as for Level1 Change event.

Line Id Warning

1) Read the actual line ids by (mlinec,r,37): Get Routing Data.

2) Check whether this is the correct configuration.

9.1.5 mlocalc and mtlocalc LocalConnector

Extension Not Connected

1) The cause lack of ressources means the LocalConnector should be created with a larger extension table size (mlocalc only).

2) The cause already connected indicates errors in the configuration of extension numbers.

Illegal Packet (mlocalc only)

1) Cause illegal format indicates software error.

Format Error

1) This is a serious event as it indicates data is corrupted through the datagram network. Either the length or the CRC is incorrect. The event should be paid special attention, and it should be tried to locate the area where the error is arisen.

9.1.6 PMS

Autoload Request (NNP), Autoload Request (X.25)

1) This event should be paid attention, as it means a load should be restarted.

9.2 Handling of Other Error Situations

In this section the handling of errors, which are not notified by an LCP event, are described.

9.2.1 No HDLC Connection

Identification

The line is started at both ends, but the HDLC connection is not established.

Actions

1) Check the state and application of the line by using the linestate (lst) activity, or issuing an (mlinec,r,49): Get Survey request.

2) Check the level1 by means of (mlinec,r,101): Get Level1 State and (for 2 Mbps lines) (mlinec,r,54(58)): Get (and Clear) Level1 Statistics. For MAC242 access board, note the setting of the modem signals.

3) Read (mlinec,r,14(18)): Get (and Clear) HDLC Statistics. Note whether XID/SABM is transmitted and received.

4) Check the HDLC configuration at both ends by means of (mlinec,r,41): Get HDLC Configuration.

5) Examine whether any LCP events are generated by means of (mlinec,r,129): Get Event List.

9.2.2 No Router Connection

Identification

The line is started at both ends as a router line, but the state is not changed to running.

Actions

1) Check the state by means of (mlinec,r,49): Get Survey, and read the latest state changes at the LineConnector by means of (mlinec,r,129): Get Event List. If the HDLC connection is never established, then follow the actions for section 9.2.1.

2) Read (mlinec,r,37): Get Routing Data. Check the node address, and other parameters received from the neighbour.

10. GUIDELINES FOR ERROR REPORTING

This chapter describes which information should be collected, when an error is to be reported.

The LCP operations indicating the actual configuration are mentioned here. Additionally, some statistic operations, which can be used to encircle the level of the error, are mentioned. However, it should be noted, that most of the monitoring LCP operations may be useful in order to locate an error.

For errors concerning the Router process (mrouter,r,71): Get Program, (mrouter,r,97): Get Library Version, (mrouter,r,49): Get Connector Survey and (mrouter,r,22): Get Router Statistics should be read.

For errors concerning the LineConnector process the following LCP operations should be read, (mlinec,r,71): Get Program, (mlinec,r,97): Get Library Version, (mlinec,r,129): Get Event List, (mlinec,r,149): Get LineConnector State, (mlinec,r,41): Get HDLC Configuration, (mlinec,r,37): Get Routing Data. The statistics, such as (mlinec,r,6): Get XPC Statistics, (mlinec,r,14): Get HDLC Statistics, (mlinec,r,22): Get Router Statistics and (mlinec,r,54): Get Level1 Statistics may be useful too.

For the LocalConnector process (mlocalc/mtlocalc,r,71): Get Program, (mtlocalc,r,97): Get Library Version, (mlocalc/mtlocalc,r,14): Get LocalConnector Statistics, (mlocalc/mtlocalc,r,9): Get extension Survey and (mlocalc/mtlocalc,r,13): Get Extension State should be read.

For the LCP-Adaptor process (lcpadaptor,r,71): Get Program, and possibly (lcadaptor,r,13): Get LCP for specific LCPs, should be read.

A. REFERENCES

1. RcPAX
An introduction

2. RC5000 MegaSwitch
General Description
PN: 99001445

3. RC5000 MegaSwitch
System and Router Software
Reference Manual
PN: 99001338

4. RC5000 MegaSwitch
System and Router Software
Installation and Operating Guide
PN: 99001446

5. RC5000 MegaSwitch
System and Router Software
EM Reference Manual
PN: 99001447

6. RC5000
MegaSwitch Hardware
Reference Manual
PN: 99001180

7. RC5000
MegaSwitch Hardware
Installation and Operating Guide
PN: 99001448

8. RC5000
MCC240 - MegaSwitch Central Controller
General Information
PN: 99001182

9. RC5000
MCC240 - MegaSwitch Central Controller
Programmer's Reference Manual
PN: 99001183

10. RC5000
MTC240 - MegaSwitch Transit Controller
General Information
PN: 99001184

11. RC5000
MTC240 - MegaSwitch Transit Controller
Programmer's Reference Manual
PN: 99001185

12. RC5000
MAC240 - MegaSwitch Access Controller
General Information
PN: 99001186

13. RC5000
MAC240 - MegaSwitch Access Controller
Programmer's Reference Manual
PN: 99001187

14. RC5000
MAC241 - MegaSwitch X.25/X.75 Access Controller
General Information
PN: 99001328

15. RC5000
MAC241 - MegaSwitch X.25/X.75 Access Controller
Programmer's Reference Manual
PN: 99001329

16. RC5000
MAC242 V.xx Access Controller
General Information
PN: 99001360

17. RC5000
MAC242 V.xx Access Controller
Programmer's Reference Manual
PN: 99001388

18. RcPAX Network System
X.25 DCE and X.75 STE
Installation and Operating Guide
PN: 99001413

19. RcPAX Network System
Logical Internal Channel (LIC)
Installation and Operating Guide
PN: 99001402

20. RcPAX Operating System
PoolManager
Installation and Operating Guide
PN: 99001420

21. RcPAX Network System
X.25 DCE
EM Reference Manual
PN: 99001415

22. RcPAX Network System
X.75 STE
EM Reference Manual
PN: 99001417

23. RcPAX Network System
Logical Internal Channel (LIC)
EM Reference Manual
PN: 99001403

24. RcPAX Operating System
PoolManager
EM Reference Manual
PN: 99001425

25. Networks Operator's Manual
RC8000 NMC
ENG-NMS.JH.48

26. Network Operator's Manual
PAXMAN
PN: ENG-NMS.115

B. EXCEPTION CODES

This appendix contains a list of the exception codes, which the RC5000 MegaSwitch System and Router modules may generate.

There are 3 different types of codes:

- Errorcode displayed by MegaDump for the transputer in error
- RTP exception code from the stopped process.
- RTP exception message from the CT process.

B.1 Error Codes Displayed by MegaDump

The printout from MegaDump has the following form:

errorcode NN lcp XX:YY

where:

NN is the errorcode (hexadecimal)
XX is the LCP Identity of the process (hexadecimal)
YY is the incarnation number of the process (hexadecimal)

These error codes are given in hexadecimal form. For each error code, the circumstances under which the exception is generated, are described and *solutions are proposed.*

For the processes running on the transputer part of the RC5000 MegaSwitch the following grouping of codes is applied.

	System error	Configuration error
Shared routines	#100-1FF	#1000-1FFF
MegaSys and linkbus	#200-2FF	#2000-2FFF
LineConnector	#300-3FF	#3000-3FFF
Router	#400-4FF	#4000-4FFF
LocalConnector (mtlocalc)	#600-6FF	#6000-6FFFF

B.1.1 Shared Routines

The occam EM routines, which are used by the MegaSys, Router and LineConnector and mtlocalc LocalConnector processes, may send the following error codes.

The error codes #100-10F indicates system error when including the LCP at the LCP-Adaptor.

#100 The LCP is already included at the LCP-Adaptor.
#101 The LCP is already included at the EM-zone.
#102 The LCP is already included at the NCP.

#103 The connection between the LCP and the LCP-Adaptor is already established.
#105 Data error when included at the EM-zone.
#106 Data error when included at the NCP.
#109 Unknown error.

The error codes #150-15F indicates error in the format parameter in call of EM routines.

#150 Unknown format in call of EM.receive.
#151 Unknown format in call of EM.answer.
#152 Unknown format in call of EM.build.event.
#153 Unknown format in call of EM.calc.limits.

The error codes #160-16F indicates error in call of the EM.attach/EM.detach routines.

#160 The buffer used at the EM.detach call is too short.
#161 The buffer used at the EM.attach call is too short.

The error codes #1000-100F indicate error in the configuration, discovered when including the LCP at the LCP-Adaptor.

#1000 The LCP ident is not defined at the LCP-Adaptor.
Define the process in question at the LCP-Adaptor by means of (lcpadaptor,r,100): Define LCP-idents.
#1001 No free resources at the LCP-Adaptor.
#1002 No free resources at the EM-zone.
#1003 No free resources at the NCP.

B.1.2 MegaSys and LinkBus

The MegaSys and LinkBus may generate the following error codes.

The error codes #200-202 indicate system error in MegaSys.

#200 Illegal transputer id. An illegal id is found through the diagnostic bus.
#201 Too many transputers. MegaSys is limited to handle up to 32 transputers.
#202 Illegal boot address. The boot address at occam address #23 is neither the address of a bootlink (#8000 0010 $\leq$ bootlink $\leq$ #8000 001F) or an EPROM address (#7FE0 0000 $\leq$ bootlink).

The error code #2000 is used when exception is set on purpose.

#2000 Exception caused by LCP operation 252: Cause Error.

The error codes #2001-2004 indicate other problems with SLT, MLT or MXT.

#2001 SLT, MLT or MXT error. The CT error signal is set due to an error on one of the SLT, MLT or MXT boards.

#2002 Missing link. One or more transputers cannot be reached. This is probably caused by illegal exclusion of boards or transputers.

#2003 MXT parity error. A DRAM parity error is detected on an MXT board.

#2004 Load timeout. A timeout has occurred during load, initialize or dump.

B.1.3 LineConnector

The LineConnector process may generate the following error codes.

The error code #300 is used when exception is set on purpose.

#300 Exception caused by LCP operation 252: Cause Error.

The error codes #310-31F indicates system error in LineConnector.

#310 Inconsistency is discovered in the state of the LineConnector.

#311 Inconsistency is discovered in the state of the transmit module.

#312 Inconsistency is discovered in the table of receive buffers.

The error codes #320-32F indicates problem with the XPC chip.

#320 The synchronization with the XPC is lost.

#321 The XPC has address or bus error on accessing the transputer memory, in connection with receive.

#322 The XPC has address or bus error on accessing the transputer memory, in connection with transmit.

The error codes #330-33F indicate problems in the RTP buffer management.

#330 Inconsistency in the number of receive buffers queued.

#331 Inconsistency in the number of transmit buffers queued, an ack received while no buffer queued.

#332 Inconsistency in the number of transmit buffers queued, data skipped at XPC buffer area.

The error codes #3000-300F indicates error in the configuration, discovered when creating the LineConnector process.

#3000 LineConnector is configured to use more memory, than exists.
Change the buffer allocation of the LineConnector by means of (msys,r,208): Set LineConnector Size.

B.1.4 Router

No error codes are applied in the Router process.

B.1.5 Transputer LocalConnector

The mtlocalc LocalConnector may generate the following errorcodes:

#600	Exception caused by (mtlocalc,r,252): Cause Error
#601	Inconsistency between PS structure and extension tables
#602	Inconsistency between PS and UPS tables
#603	Inconsistency in UPS structure

B.2 RTP Exception codes

The RTP exception codes have the following form:

```
PPII>> exception, excode = CC: TTT
.
.
.
```

where:

PP	is the process name
II	is the incarnation number
CC	is the exception code
TTT	is an error message text

The RTP exception codes are grouped in general exception codes and in process specific exception codes. Only for the general exception codes the error message text is displayed.

The following System and Router processes may generate process specific exception codes:

Mopsys
MbxLink
TreesAdm
MegaLink
MOperSys
MegaDriver
MegaDump

B.2.1 General RTP exception codes

The 'stack overflow' exception error message may indicate a configuration error. All other general exception codes indicate programming errors.

B.2.2 Mopsys

The exception codes are stated in hexadecimal form:

#64 em_initialize failed
#68 Cannot allocate buffers for input from linkbus
#69 Cannot allocate buffers for output to linkbus
#6A Cannot allocate event buffers
#6B Cannot allocate request/answer buffers
#6C Cannot allocate buffers to force lcp_disconnect
#6D No free buffers for output to linkbus
#70 Namemailbox 'ncp' failed
#71 Namemailbox 'paxopsys' failed
#72 Initialization of ml structure failed
#73 Cannot allocate buffer for event generation
#74 No free buffer for event generation
#75 Cannot allocate buffer for operator zone output
#76 Namemailbox addrmbx failed
#77 Mailbox named 'RTPLBout' not found

B.2.3 MbxLink

The exception codes are stated in hexadecimal form:

#100 em_initialize error
#101 not enough ressources to allocate asn1 and linkbus input buffers
#102 reference not nil

B.2.4 TreesAdm

The exception codes are stated in hexadecimal form:

#101 Link error, cannot find parsetree program. The parsetree is probably not loaded into the RC5000 CPU memory.
#103 Link error, parameter inconsistency for parsetree program.
#106 Link error, parsetree program already linked.
#111 Create error, process not nil; system error.
#112 Create error, program not linked; system error.
#113 Create error, no memory available. Configuration problem.
#121 Unlink error, program not linked; system error.
#122 Unlink error, program busy; system error.
#130 + result
Unload error; system error.

B.2.5 MegaLink

The exception codes are stated in decimal form:

110	get_pax_sem failed, problem with mailbox name in (mega-link,r,1)
1000	cannot create link, no free resources
1011	cannot create link, no free link resources (RC5000 CPU)
1015	cannot create link, mailbox not found (RC5000 CPU)
1023	cannot create link, size error, BUF_SIZE < ASN1_BUFSIZE defined in (mbxlink,r,1)
1035	cannot create link, no free resources (RC5000 CPU)
2009	sync_paxopsys failed, no free resources
3001	link_init failed, no free resources
3011	link_init failed, no free link resources (RC5000 CPU)
3015	link_init failed, mailbox not found (RC5000 CPU)
3023	link_init failed, size error, BUF_SIZE < ASN1_BUFSIZE defined in (mbxlink,r,1)
3035	link_init failed, no free resources (RC5000 CPU)

These exceptions may indicate both configuration errors and program errors.

B.2.6 MOperSys

The exception codes are stated in hexadecimal form:

#64	No free resources

This exception indicates a configuration problem.

B.2.7 MegaDriver

The following errorcodes are issued if severe inconsistencies are found in MegaDriver.

#30	Event cannot be spooled.
#31	Ct_transmit exceeds Message_max_Length.
#33	Mirror process cannot be created.
#34	Mdriver process cannot be created.
#35	Message received exceeds Message_max_Length.
#37	Data-program cannot be unlinked.
#38	Initialization error.
#39	Interrupt level 111 (#6F) cannot be reserved.
#3A	Interrupt level 110 (#6E) cannot be reserved.
#3B	Message received by MegaDriver as an answer (u2 <> message) with an illegal stream (u4 <> 1).

#3C Internal process 'mdriver' cannot be created.

B.2.8 MegaDump

The interpretation of MegaDump error codes depends on the routines which call exception as certain important values are encoded into the error code.

#160-161 Called from init_dump.
Buffers cannot be allocated.

n * #100, n≥1, Called from restart.
ADAM returns error n, when MegaDump links, creates or starts process 's'.

n * #100 + m + #40 Called from ct_operation.
A user request to MegaDriver is returned with result (u2 = m). The u1 value is n.

B.3 RTP Exception Message

The RTP Exception message from the CT process has the following form:

```
> CT
Transputer in error: XX

MM> exception, code = TTT
.
.
.
```

where:

XX is the transputer in error
MM is the module in error
(occam is used for all occam modules)
TTT is the error message

The RTP exception message is displayed for debugging purposes.

C. TRACE CODES

This appendix contains a list of the trace codes, which the RC5000 MegaSwitch System and Router modules may send.

The trace codes are sent as trace events to the controlling NMC(s).

For all modules except Mopsys and MbxLink the trace event consists of a trace ident, which is a text string, and 4 bytes (b1-b4) of trace specific data. For these modules the values of trace ident and b1-b4 are listed together with the circumstances under which the trace is sent. The term 'LinkBus header' is used to indicate that b1-b4 contains the type, dst, src and id.class used to identify a message. The term 'INT' followed by a value is used to indicate that b1-b4 should be read as a 32-bit integer.

For Mopsys and MbxLink the trace event consists of a trace code followed by a number of fields with trace specific data. For these modules the trace codes are listed together with the circumstances under which the trace is sent.

C.1 Shared Routines

The occam EM routines, which are used by the MegaSys, Router, LineConnector and transputer LocalConnector processes, may send the following trace events.

EM:req.size LinkBus header
LCP request, with request length exceeding the amount of data in the buffer, is received.

EM: Init.Done LinkBus header
LCP.Init.Done message received while connected to the LCP-Adaptor.

EM:unknown LinkBus header
A unknown message is received from the LCP-Adaptor.

C.2 MegaSys and LinkBus

No traces are generated by MegaSys and LinkBus.

C.3 MegaDriver

No traces are generated by MegaDriver.

C.4 Router

The Router process may send the following trace events.

unknown dst **LinkBus header**
A message with type Message.Unknown.DST is received from a Connector.

unknown type **LinkBus header**
A unknown message is received from a Connector.

ill EMresult **LinkBus header**
A message from LCP-Adaptor is of a type that is unknown to Router.

unknown src **LinkBus header**
A message is received from a source not expected to communicate with Router.

too short **LinkBus header**
A message less than 4 bytes is received from the LinkBus.

RR too short **LinkBus header**
A message with a Router-Router packet is too short.

bc too short **LinkBus header**
A message with a broadcast packet is too short.

Dgn too short **LinkBus header**
A message with a packet from the datagram network (dgn) for the Router process is too short.

conf incons1 **INT conn.no**
The LineConnector conn.no is telling that it is another connector type (local/line) than what the Router knows from the LinkBus address convention.

conf incons2 **INT conn.no**
The LineConnector conn.no is telling that it is another (MLT/SLT) kind than what the Router knows from its configuration information.

TLV 1 **INT T**
A TLV element without room for the T and L elements in the buffer is received.

TLV 2 **INT T**
A TLV element without room for the V element in the buffer is received.

node too big **INT conn.no**
A range vector is received via LineConnector conn.no with a node number that is too big for the routing tables.

reg too big **INT conn.no**
A range vector is received via LineConnector conn.no with a region number that is too big for the routing tables.

net too big **INT conn.no**
A range vector is received via LineConnector conn.no with a net number that is too big for the routing tables.

illegal time **INT conn.no**
An NNP clock message is received via LineConnector conn.no but the date and time in the message contain an illegal value.

TLV unknown **INT T**
An TLV element is skipped because the type (T) is unknown.

bc ill lgt **INT L**
The length (L) in an TLV element in a broadcast message is considered illegal.

ill format **INT format**
A message with a packet from the datagram network for the Router process has a format field that is illegal.

C.5 LineConnector

The LineConnector process may send the following trace events.

TLV 1 **INT T**
A TLV element without room for the T and L elements in the buffer is received.

TLV 2 **INT T**
A TLV element without room for the V element in the buffer is received.

TLV length **T, L, expected length, 0**
A TLV element has been skipped because the length (L) was considered illegal.

TLV unknown **T, L, 0, 0**
A TLV element has been skipped because the type (T) was unknown. This trace is only generated when a test option is set.

Xpc seq **INT sequence module**
XPC is set up with an HDLC sequence modulo not equal to 8 or 128.

Xpc hdlc addr **INT addr**
XPC is set up with an HDLC address not equal to 1 (B) or 3 (A).

Xpc rec addr **0, 0, 0, 0**
XPCchip has address or bus error on accessing the transputer memory.

Xpc trm addr **0, 0, 0, 0**
XPCchip has address or bus error on accessing the transputer memory.

Trm state **INT Trm.batch.state**
Inconsistency is discovered in the state of the transmit module. This trace event will be followed by an exception event.

Rec state **0, 0, Rec.batch.test.index, Rec.current**
Inconsistency is discovered in the table of receive buffers. This trace event will be followed by an exception event.

c: Ill state **new state, actual state, cause, 0**
Inconsistency is discovered in the state of the LineConnector. This trace event will be followed by an exception event.

Xid 1 **INT group.id**
A group is read from an XID frame, no room for the group length field in the buffer.

Xid 2 **INT group.id**
A group is read from an XID frame, no room for the group data in the buffer.

Xid 3 **INT param.id**
A group is read from an XID frame, no room for the param length field in the buffer.

Xid 4 **INT param.id**
A group is read from an XID frame, no room for the parameter data in the buffer.

Xid 5 **INT received param length**
An 'optional function' parameter has been skipped in an XID frame because the length was considered illegal.

Xid 6 **INT data**
An 'optional function' parameter is received with data not choosing among HDLC sequence modulo 8 or modulo 128.

Xid 7 **INT param.id**
A private parameter group is received in an XID frame, but the first parameter in the group is not 'private id'.

Xid 8 **INT param.length**
A 'private address' parameter has been skipped in an XID frame because the length was considered illegal.

Xid 9 **INT data**
A 'private address' parameter is received with data not choosing among HDLC address 1 (B) or 3 (A).

Xid 10 **0, 0, 0, 0**
A collision of XID command frames has occurred, but not possible to select parameters as unique id is missing in the frame.

R: init **LinkBus header**
Line.Exist.Done message received from the Router while state is initialized.

R: skipped **0, 0, 0, 0**
A message with credit from the Router (RC.To.Line.Credit) is skipped in the transmit module.

R: quest.form **LinkBus header**
A Question message with an unknown format is received from the Router.

R: unknown **LinkBus header**
An unknown message is received from the Router.

U: skipped **0, 0, 0, 0**
A message with credit from a User (Data.to.Line.Credit) is skipped in the transmit module.

U: unknown **LinkBus header**
An unknown message is received from a User process.

too short **LinkBus header**
A message too short to contain the 4 bytes for the LinkBus header is received from the LinkBus.

unknown src **LinkBus header**
A message is received from a source not expecting to communicate with the LineConnector.

RP: ext t.val **INT tvalue**
An unknown T value is received in an Extension.Info message.

RP: t.val **INT t-value**
An unknown T value is received in a Route.Info message.

C.6 RC5000 CPU LocalConnector

The following notation is in addition used in this section

X,,Y means 2 16-bit integers,

i.e.

X = b1 * 256 + b2
Y = b3 * 256 + b4

The RC5000 CPU LocalConnector (mlocalc) may send the following trace events:

ct input **u1, u2, u3, u4**
A message returned from MegaDriver, where u2 ≠ 0.

unknown typ **LinkBus header**
Unknown TYP-field in received message.

not for us **LinkBus header**
Unknown DST-field in received message.

unknown dst **LinkBus header**
An 'Unknown dst'-message has been received.

Ill IdBit **ID.BIT, 0, 0, 0**
A 'Message Lost' with unknown ID.BIT received.

copy error **destination room,,data-size**
Unable to copy data.

ext t.val **T,0,0,0**
An unknown T-value is received in an Extension.info message.

ext incons **ext_no,,ext_index**
Inconsistency between the extension tables in Router and Local-Connector.

R: quest.form **format,0,0,0**
An unknown format is received in a Question message.

The following traces concern Route.Info only.

rng t.val **T, 0, 0, 0**
Illegal T-value.

rng head **room,,4**
No room for T, L, region and dummy fields.

rng length **room,,L**
L-value too big.

rng start **room,,2**
No room for from and count in this V-part.

rng inner **room,,count * 6**
Count too big for this V-part.

The following traces concern Broadcast.To.Line only.

brcst head **room,,8**
No room for broadcast header.

brcst tl **room,,2**
No room for T and L fields.

brcst big tl **room,,4**
No room for extended T and L fields.

brcst t.val **T,,L**
Illegal T-value.

brcst info **room,,L**
L-value too big.

brcst length **L,,17**
Broadcast-algorithm with wrong length.

brcst multi
More than one broadcast-algorithm.

data first
No broadcast-algorithm before data.

b2cst head **room,,22**
No room for reformatting.

b2cst tl **room,,2**
No room for T and L fields.

b2cst big tl **room,,4**
No room for extended T and L fields.

b2cst t.val **T,,L**
Illegal T-value.

b2cst info **room,,L**
L-value too big.

The following traces concern Clock.To.Conn only.

clock tl **room,,2**
No room for T and L fields.

clock big tl **room,,4**
No room for extended T and L fields.

clock t.val **T,,L**
Illegal T-value.

clock info **room,,L**
L-value too big.

clock multi
More than one GMT record.

clock gmt.lgt **L,,8**
Illegal L-value in GMT record.

clock der.lgt **L,,2**
Illegal L-value in GMT-derivation record.

C.7 Transputer LocalConnector

The transputer LocalConnector (mtlocalc) may send the following trace codes:

TLV 1 INT T
A TLV element without room for the T and L elements in the buffer is received.

TLV 2 INT T
A TLV element without room for the V element in the buffer is received.

RP: t.val INT T
An unknown T value is received in a Route.Info message.

RP: ext t.val INT T
An unknown T value is received in an Extension.Info message.

LC: init LinkBus header
Line.Exit.Done message is received from the Router while state is initialized.

R: quest.form LinkBus header
A Question message with an unknown format is received from the Router.

too short LinkBus header
A message too short to contain the 4 bytes for the LinkBus header is received from the LinkBus.

unknown srs LinkBus header
A message is received from a source not expected to communicate with the LocalConnector.

unknown timer u1,u2,u3,u4
An unknown message is received from the timer process.

C.8 LCP-Adaptor

The following terms are used in this section:

Cause

20: Request dropped, because LCP.Init.Done lost
21: Request lost
22: Answer lost
23: Answer timeout
24: LCP.Answer
25: Reset.Stat answer
26: Report.State answer
27: Poll answer
28: Request retransmission not possible

Kind

0: LCP-Request/Answer
1: Reset.Stat
2: Report.State
3: Poll

State

0: Not initialized
1: Initializing
2: Initialized
3: Terminating

Addr means LinkBus-address of appropriate LCP.

The LCP-Adaptor may send the following trace events:

ASN1 ERROR LCP-type, LCP-operation, result, 0
ASN1 conversion error.result may be:
0 OK
1 LCP type not found
2 LCP operation not found
3 ASN1 definition too big
4 ASN1 definition too small
5 Fatal error

Errorcode **LCP-type, LCP-operation,X,Y**
X * 256 + 4 is an extended errorcode when an ASN1 ERROR is encountered.

answer.id? **ID, Addr, Cause, 1**
ID not between 17 and 30.

answer.kind **rec Kind, exp Kind, Cause, 2**
Unexpected Kind.

ans.sender **rec Addr, exp Addr, Cause, 3**
Unexpected Addr

answer.LCP **LCP-inc, Addr, Cause, LCP-type**
Unexpected LCP-ident.

Lost.LCP-state **State, Addr, 4, 0**
LCP.Init.Done lost when in wrong state.

Lost.Id-bit **ID, Addr, 5, 0**
Illegal ID-bit in Message.Lost.

Lost.-count **A, Addr, 6, B**
Count in Message.Lost A*256+B too small.

Lost.+count **A, Addr, 7, b**
Count in Message.Lost A*256+B too big.

Lost.lcp-state **State, Addr, 8, 0**
Message.Lost when in wrong state.

Lost.Spool **0, Addr, 9, 0**
Impossible to spool events-lost event.

Output u2 **u2, 0, 10, 0**
ct_transmit failure.

Input u2 **u2, 0, 11, 0**
ct_receive failure.

Input dst **rec DST, 0, 12, exp DST**
DST not LCP-Adaptor.

Id-bits lost **Addr, i, 19, lcp index**
i Id-bits are lost.

Lcpidx Unknwn **Addr, TYP, 18, LCP index**
Unknown LCP index.

Event ID **rec ID, Addr, 14, exp ID**
ID wrong.

Event.spool **0, Addr, 15, event-no**
Impossible to spool event.

Message.typ? **TYP, Addr, 16, 0**
Unknown TYP-value.

C.9 PMS

The PMS may send the following trace codes:

R.Connect **u1, u2, u3, u4**
Impossible to connect to router.

Data.State **0, 0, State, 0**
Send DataIfReady called in wrong state.

Ack.Done **24, status, 0, 0**
Ack.Done in wrong state.

Credit.Done **0, 0, state, 0**
Credit.Done in wrong state.

Ill IdBit **ID, 0, 0, 0**
Illegal ID-bit in Message.Lost.

Rout.Input **u1, u2, u3, u4**
rec_packet failure.

From.mls **code, state, 0, 0**
Illegal code from MLS.

From.Mega **u1, u2, u3, u4**
ct_receive failure.

Start.Line **3, status, 0, 0**
Start.Line not possible.

Ill LoadReq **128, function, 0, 0**
Function ought to be 6.

From.Bls **code, 0, 0, 0**
Illegal code from target.

Unkw dst **LinkBus Header**
unknown.dst received.

Unkw typ **LinkBus Header**
Illegal TYP-value.

ct output **0, 0, u2, 00**
ct_transmit failure.

timer **0, 0, state, 0**
timeout when in wrong state.

C.10 MLD

Note: The traces in this section will appear on the RC3502 debug-console only. The format is

<text> <number>

If <number> is 0 it is absent.

mdriver link error **link result**
link-call failure.

mdriver create error **create result**
create-call failure.

mdriver analyse_pram error **u2**
ct_analyse_pram failure.

mdriver output error **u2**
ct_output (negative number) failure.

mdriver start_receive error **u2**
Impossible to perform Start.Receive.

Kick watchdog **0**
If watchdog function was not switched off (watchdog = 0) a watchdog restart would have been performed.

Exception **excode**
MLD stopped due to exception in MLD.

u2 error to lcpadap **u2**
MLD stopped due to ct_receive failure.

u2 error from ct_driver **u2**
MLD stopped due to ct_receive failure.

Start.Line.Done, Status **status**
MLD stopped due to Start.Line failure.

run loader error **0**
MLD stopped because it is impossible to link, create and start loader.

Output finis **u2**
MLD stopped caused by loader finished.

Unknown u3 from loader **u3**
Unknown u1-value **u1**
Error in request from loader

lcp-answer error **status**
Start.Receive failure.

unknown typ to lcpadap **TYP**
Illegal TYP-value.

Start.Line.Done unexpected **state**
State was not 3.

Messages Lost **count**
MLD is unable to handle Message.Lost.

<seqno> expected, received seqno
Sequence-number error detected.

Break Load **cause**
Break_load received from platform.

Unknown dllprot.code from line **code**
Protocol error from platform.

Unknown head.typ **TYP**
Illegal TYP-value.

Output credit error **u2**
ct_transmit failure.

C.11 Mopsys

Mopsys may generate trace events with the following trace codes:

#78	No free linkbus output messages, program error
#79	MbxLink user not found, program error
#7A	No more answer buffer dedicated to this MbxLink, configuration problem
#7B	No free user ressources, ressource problem
#7C	Unknown ID in received linkbus message, program error
#7D	Unknown TYP in received linkbus message, program error
#7E	Unexpected LCP operation in LCP answer
#7F	Unexpected lcpid in LCP answer
#80	Unknown protocol element received from LCP-Adaptor
#81	No free request buffer
#82	No free buffer for requested lcp_disconnect
#83	Protocol error, no initialize in progress
#84	Protocol error, no event in progress
#85	Protocol error, no disconnect in progress

The field MORE INFO is not used.

The fields U1 to BUFSIZE are the message attributes.

C.12 MbxLink

MbxLink may generate trace events with the following trace codes:

#200 = release of unknown sequence no, program error
#201 = decode unknown sequence no, program error
#202 = no free link headers, configuration problem
#203 = no free link buffers, configuration problem
#204 = send link no unknown, program error
#205 = receive link no unknown, program error
#206 = illegal request type, program error
#207 = illegal stream no (u4), program error
#208 = illegal id bit, program error
#209 = asn1 init error, load problem
#20A = asn1 encode/decode error, parse tree error

The field 'more info' is used to give further information dependent on the trace code:

#200 : sequence number
#201 : sequence number
#202 : link number
#203 : link number
#204 : link number
#205 : sequence number
#206 : request type
#207 : u4
#208 : id bit
#209 : asn1 result
#20A : RESULT * 4096 + ERROR, where RESULT has the following meaning:

0 = OK
1 = ATS parse tree not found, load problem
2 = DIS parse tree not found, load problem
3 = ATS operation not found
4 = ATS object not found
5 = ATS Fatal error, ERROR holds further debug information

The fields U1 to BUFSIZE are the message attributes.

D. RC5000 AUTOLOAD FILE

An RC5000 MegaSwitch autoload file supporting:

- MegaSwitch down line autoload
- Winchester autoload

can be generated by the crosslink utility with the following call:

```
autoloadfile = crosslink
    system3502,
    stdplib,
    measureplib
    em35022plib,
    lf3502plib,
    fs3502plib,
    fsuserplib,
    fp3502plib,
    blsplib,
    wldplib,
    paxsemplib,
    megaboot
```

where:

system3502, stdplib, measureplib and em35022plib are included in SW50020

and

lf3502plib, fs3502plib, fsuserplib and fp3502plib are included in SW50220

and

blsplib, wldplib and paxsemplib are included in SW2802

and

megaboot is included in SW50050 (the file is named megaboot.p22).

E. INDICES

E.1 Survey of Figures

PN: 99001446