

1
2

Project	IEEE 802.20 Working Group on Mobile Broadband Wireless Access < http://grouper.ieee.org/groups/802/20/ >	
Title	Enhanced MIB Chapter of 625k-MC Mode	
Date Submitted	Monday, July 13, 2009	
Authors(s)	Radhakrishna Canchi	Email : cradhak@kyocera-wireless.com
	Kazuhiro Murakami	Email kazuhiro.murakami.xm@kyocera.jp
Re:	IEEE 802.20 Enhanced MIB Chapter – 625k-MC Mode	
Abstract	This contribution updates the draft enhanced MIB chapter for IEEE 802.20 625k-MC Mode.	
Purpose	This is a work in progress, and is for review of the working group prior to the November, 2009 meeting. For consideration and approval of 802.20 WG	
Notice	This document has been prepared to assist the IEEE 802.20 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein	
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.20.	
Patent Policy	The contributor is familiar with IEEE patent policy, as outlined in Section 6.3 of the IEEE-SA Standards Board Operations Manual < http://standards.ieee.org/guides/opman/sect6.html#6.3 > and in <i>Understanding Patent Issues During IEEE Standards Development</i> < http://standards.ieee.org/board/pat/guide.html >.	

1 625k-MC OA & M Radio Network Quality Monitor and Control Enhancement

This Chapter is an added to the baseline specification HC-SDMA [Error! Reference source not found.\[25\]](#).

625k-MC network systems provide radio network quality monitoring and control functionality. The MIB of 625k-MC mode comprises of the managed objects, attributes, actions, and notifications required to manage a BS. The definition of these managed objects, attributes, actions, and notifications, as well as their structure, is presented below.

1.1 625k-MC Mode MIB

1.1.1 Overview

This chapter defines a Management Information Base (MIB) module for managing the 625k-MC mode. Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). The objects in this MIB are defined using the mechanisms specified in the Structure of Management Information (SMI). The MIB module specified is compliant to SMIV2 which is described in RFC 2578 [Error! Reference source not found.\[18\]](#), RFC 2579 [Error! Reference source not found.\[19\]](#), and RFC 2580 [Error! Reference source not found.\[20\]](#).

1.1.2 Definition

```

19 IEEE802dot20-625k-MC-MIB DEFINITIONS ::= BEGIN
20
21 IMPORTS
22   ifIndex
23   FROM IF-MIB
24   MODULE-COMPLIANCE, OBJECT-GROUP
25   FROM SNMPv2-CONF
26   Counter32, Counter64, Integer32, Unsigned32, MODULE-IDENTITY,
27   OBJECT-IDENTITY, OBJECT-TYPE, transmission
28   FROM SNMPv2-SMIRFC 1212
29   TruthValue, enterprises, Gauge, Counter, Cauge32
30   FROM SNMPv2-TCRFC 1155 SMI
31   transmission FROM RFC1213 MIB
32
33 ;
34
35 IEEE802dot20-625k-MC-MIB MODULE-IDENTITY
36   LAST-UPDATED      "2008110072000Z" - November 08, 2008
37   ORGANIZATION      "IEEE 802.20"
38   CONTACT-INFO      "Contact: R. Canchi
39   Postal: 472 Kato Terrace, FREMONT, CA 94539, USA
40   Tel: 510-257-0132
41   Fax: 510 257 0131
42   E-mail: cradhak@ktrc-na.com"
43
44   DESCRIPTION
45     "The MIB module for IEEE802.20 625k-MC mode private module for IEEE
46     802.20 entities"
47     ::= { transmission enterprises 9999 }
48
49   _625k-MCSystem          OBJECT IDENTIFIER

```

```

1      -- DESCRIPTION      "System Elements"
2  | --- ::= { IEEE802dot20-625k-MC-MIB 1 }
3
4
5  | _625k-MCSysAlarms      OBJECT IDENTIFIER
6      -- DESCRIPTION      "Alarms"
7  | --- ::= { _625k-MCSystem 1 }
8
9
10
11 | _625k-MCAlarmScalars_  OBJECT IDENTIFIER
12      -- DESCRIPTION      "Alarm Scalars"
13 | --- ::= { _625k-MCSysAlarms 1 }
14
15
16
17 _625k-MCCommonAlarmStatus OBJECT-TYPE
18 | SYNTAX      Unsigned32INTEGER -- Unsigned32Type
19 | MAX-ACCESS      read-only
20 | STATUS      Currentmandatory
21 | DESCRIPTION
22 |     "Common alarm atatus."
23
24
25
26     (From mibCtl ElementType 16 CommonAlarmStatus)
27     Description for mibCtl Type 14 Unsigned32Type :
28         32 bit unsigned integer.
29     Type derived from mibCtl Type 11 Word32Type :
30         32 bits of raw opaque data.
31     Derived from basic 32 bit word type.
32     "
33     ::= { _625k-MCAlarmScalars 1 }
34
35
36
37 _625k-MCFailReasonForAlarm OBJECT-TYPE
38 | SYNTAX      Unsigned32INTEGER -- Unsigned32Type
39 | MAX-ACCESS      read-only
40 | STATUS      Currentmandatory
41 | DESCRIPTION
42 |     "Fail reason for alarm."
43
44
45
46     (From mibCtl ElementType 15 FailReasonForAlarm)
47     Description for mibCtl Type 14 Unsigned32Type :
48         32 bit unsigned integer.
49     Type derived from mibCtl Type 11 Word32Type :
50         32 bits of raw opaque data.
51     Derived from basic 32 bit word type.
52     "
53     ::= { _625k-MCAlarmScalars 2 }
54
55
56
57
58 _625k-MCAlarmSummaryTable OBJECT-TYPE
59 | SYNTAX SEQUENCE OF _625k-MCAlarmSummaryTableEntry
60 | ACCESSMAX-ACCESS      not-accessible
61 | STATUS      mandatoryCurrent
62 | DESCRIPTION      "Alarm Summary Table"
63 | ::= { _625k-MCSysAlarms 2 }
64
65
66
67 _625k-MCAlarmSummaryTableEntry OBJECT-TYPE
68 | SYNTAX      _625k-MCAlarmSummaryTableEntry

```

```

1 | ACCESSMAX-ACCESS not-accessible
2 | STATUS mandatoryCurrent
3 | DESCRIPTION ""
4 | INDEX { _625k-MCAAlarmSummaryTableIndex }
5 | ::= { _625k-MCAAlarmSummaryTable 1 }
6 |
7 | _625k-MCAAlarmSummaryTableEntry ::= SEQUENCE {
8 |   _625k-MCAAlarmSummaryTableIndex INTEGER, -- AlarmEventType
9 |   _625k-MCAAlarmSummary INTEGER -- AlarmStateType
10 | }
11 |
12 |
13 |
14 | _625k-MCAAlarmSummaryTableIndex OBJECT-TYPE
15 | SYNTAX INTEGER -- AlarmEventType
16 | | ACCESSMAX-ACCESS read-only
17 | | STATUS mandatoryCurrent
18 | | DESCRIPTION
19 | | "
20 | | Description for mibCtl Type 85 AlarmEventType :
21 | | Enumeration of alarm event types.
22 | |
23 | | Defines semantics of events that are also alarms.
24 | | All alarm events are enumerated first in the list of event types.
25 | | The highest alarm event index will never be more than 255.
26 | | [Limits: 0 255 ]
27 | | Type derived from mibCtl Type 3 EventType :
28 | | Enumeration of event types.
29 | |
30 | | Defines semantics of events.
31 | | An event is re. an event log message.
32 | | [Limits: 0 255 ]
33 | | Type derived from mibCtl Type 14 Unsigned32Type :
34 | | 32 bit unsigned integer.
35 | | Type derived from mibCtl Type 11 Word32Type :
36 | | 32 bits of raw opaque data.
37 | | Derived from basic 32 bit word type.
38 | | "
39 | | ::= { _625k-MCAAlarmSummaryTableEntry 1 }
40 |
41 |
42 | _625k-MCAAlarmSummary OBJECT-TYPE
43 | SYNTAX INTEGER -- AlarmStateType
44 | | ACCESSMAX-ACCESS read-only
45 | | STATUS mandatoryCurrent
46 | | DESCRIPTION
47 | | "Summary of all alarms generated by the base station.
48 | |
49 | | Each element contains the summary of a type of alarm (e.g. module
50 | | over temperature). When the management station sees that alarm
51 | | summary is SET, it can, for example, query AlarmModuleOverTemp
52 | | to see which module(s) is over temperature.
53 | |
54 | | (From mibCtl ElementType 5210 AlarmSummary)
55 | | Description for mibCtl Type 80 AlarmStateType :
56 | | Current state of an alarm.
57 | |
58 | | This value is CLEARED when
59 | | the conditions which caused the alarm to occur are taken care of
60 | | and no longer exist.
61 | | The value is SET when due to some conditions, the Base Station
62 | | software decides that an alarm is necessary.
63 | | Typically (though this may not be true for all alarms
64 | | or if the alarm changes state too frequently)
65 | | an event is logged when an alarm is SET and then again when it
66 | | is CLEARED.
67 | | [Limits: 0 1 ]
68 | | Description for mibCtl AlarmStateType 0 CLEARED :

```

```

1      No alarm.
2      Description for mibCtl AlarmStateType 1 SET :
3      Alarm is set.
4      "
5      ::= { _625k-MCAAlarmSummaryTableEntry 2 }
6
7
8
9      _625k-MCSysFiles                OBJECT IDENTIFIER
10     -- DESCRIPTION                  "Files"
11     ::= { _625k-MCSystem 2 }
12
13
14
15     _625k-MCStatsFiles                OBJECT IDENTIFIER
16     -- DESCRIPTION                  "Statistics file"
17     ::= { _625k-MCSysFiles 1 }
18
19
20
21     _625k-MCStatsUploadURL            OBJECT-TYPE
22     SYNTAX                            OCTET STRING (SIZE(0..64)) -- URLType
23     ACCESSMAX-ACCESS                read-write
24     STATUS                            mandatoryCurrent
25     DESCRIPTION
26         "EMS location to upload BS statistics file.
27
28
29
30         (From mibCtl ElementType 2831 StatsUploadURL)
31     Description for mibCtl Type 401 URLType :
32         Universal Resource Locator (URL).
33
34         A Universal Resource Locator (URL) is a text string
35         that specifies a network location for a file.
36         The general format for a URL consists of 2 parts:
37
38         1. Protocol name: lower case letters, followed by a colon.
39         See below for supported protocols.
40         This field may be omitted, to default to the file: protocol.
41
42         2. Additional information, depending on the protocol.
43         For many protocols, a host name is required,
44         which consists of a dotted numerical Internet Protocol (IP)
45     address,
46         or a dotted symbolic name with alphanumerical components,
47         where supported.
48
49         Supported protocols are:
50
51         tftp: is the Trivial File Transfer Protocol.
52         The additional information should begin with two slashes (//)
53         followed by a host name, a slash (/) and a file path.
54         The file path is interpreted by the host system,
55         frequently relative to a special directory set up for this
56     purpose.
57
58         file: is the plain old file protocol.
59         The additional information consists of a file path, which
60         should begin with a slash (/).
61         This is only useful if Base Station has been configured
62         to be an Network File System (NFS) client of the host.
63         The filepath is interpreted on the Base Station, so it
64         must begin with the mount name specified in the NFS configuration.
65     Type derived from mibCtl Type 15 TextType :
66         ASCII or compatible text.
67     Type derived from mibCtl Type 12 OctetType :
68         8 bits of raw opaque data.

```

```

1         Derived from basic 8 bit word type.
2         "
3         ::= { _625k-MCStatsFiles 1 }
4
5
6
7         _625k-MCStatsUploadStatus                                OBJECT-TYPE
8         SYNTAX          INTEGER -- FileUploadStatusType
9         ACCESSMAX-ACCESS          read-only
10        STATUS          mandatoryCurrent
11        DESCRIPTION
12            "Stats file upload status.
13
14
15            (From mibCtl ElementType 2832 StatsUploadStatus)
16            Description for mibCtl Type 403 FileUploadStatusType :
17                File upload status.
18
19
20            Description for mibCtl FileUploadStatusType 0 Unknown :
21                File upload status is unknown.
22            Description for mibCtl FileUploadStatusType 1 Missing :
23                File is missing or invalid.
24            Description for mibCtl FileUploadStatusType 2 PartialUpload :
25                File is in the process of being upload to EMS.
26            Description for mibCtl FileUploadStatusType 4 Complete :
27                File is completely uploaded to EMS.
28            Description for mibCtl FileUploadStatusType 5 Failure :
29                Upload process is failure.
30            Description for mibCtl FileUploadStatusType 6 NotManaged :
31                File upload is not being managed.
32
33        "
34        ::= { _625k-MCStatsFiles 2 }
35
36
37
38        _625k-MCStatsUploadFailReason                            OBJECT-TYPE
39        SYNTAX          INTEGER -- FileUploadFailReasonType
40        ACCESSMAX-ACCESS          read-only
41        STATUS          mandatoryCurrent
42        DESCRIPTION
43            "Reason for last stats file upload failure.
44
45
46            (From mibCtl ElementType 2833 StatsUploadFailReason)
47            Description for mibCtl Type 406 FileUploadFailReasonType :
48                Reason for failure to upload a file..
49
50
51            Description for mibCtl FileUploadFailReasonType 0 NoFailure :
52                File upload in progress or completed without problem.
53            Description for mibCtl FileUploadFailReasonType 1 BadPathSpecified :
54                File upload failed because network path not found.
55            Description for mibCtl FileUploadFailReasonType 2 FlashDiskReadError :
56                File upload failed because of flash disk read error.
57            Description for mibCtl FileUploadFailReasonType 3 Aborted :
58                File upload aborted due to change of specification.
59            Description for mibCtl FileUploadFailReasonType 4 WriteError :
60                Error in putting a file.
61
62        "
63        ::= { _625k-MCStatsFiles 3 }
64
65
66
67        _625k-MCStatsUploadBytes                                OBJECT-TYPE
68        SYNTAX          Unsigned32INTEGER -- Unsigned32Type

```

```

1 | ACCESSMAX-ACCESS read-only
2 | STATUS mandatoryCurrent
3 | DESCRIPTION
4 | "Upload size of BS stats file in bytes.
5 |
6 |
7 |
8 | (From mibCtl ElementType 2834 StatsUploadBytes)
9 | Description for mibCtl Type 14 Unsigned32Type :
10 | 32 bit unsigned integer.
11 | Type derived from mibCtl Type 11 Word32Type :
12 | 32 bits of raw opaque data.
13 | Derived from basic 32 bit word type.
14 | "
15 | ::= { _625k-MCStatsFiles 4 }
16 |
17 |
18 |
19 | _625k-MCStatsUploadDate OBJECT-TYPE
20 | SYNTAX Gauge32 -- AbsoluteTimeType
21 | ACCESSMAX-ACCESS read-only
22 | STATUS mandatoryCurrent
23 | DESCRIPTION
24 | "BS Stats File upload complete time.
25 |
26 |
27 |
28 | (From mibCtl ElementType 2835 StatsUploadDate)
29 | Description for mibCtl Type 801 AbsoluteTimeType :
30 | Absolute time in GPS seconds.
31 |
32 | GPS (Global Positioning System) time in seconds since Jan. 6,
33 | 1980.
34 | Note that this differs from UTC (in addition to a possible
35 | offset due to starting time) due to leap seconds; see
36 | the GpsLeapSecond element.
37 | Type derived from mibCtl Type 18 Gauge32Type :
38 | 32 bits of Gauge data.
39 | Derived from basic 32 bit word type.
40 | "
41 | ::= { _625k-MCStatsFiles 5 }
42 |
43 |
44 |
45 | _625k-MCUploadStatsFile OBJECT-TYPE
46 | SYNTAX Unsigned32INTEGER -- Unsigned32Type
47 | ACCESSMAX-ACCESS read-write -- REALLY: write-only
48 | STATUS mandatoryCurrent
49 | DESCRIPTION
50 | "Upload Stats file.
51 |
52 |
53 |
54 | (From mibCtl ElementType 2836 UploadStatsFile)
55 | Description for mibCtl Type 14 Unsigned32Type :
56 | 32 bit unsigned integer.
57 | Type derived from mibCtl Type 11 Word32Type :
58 | 32 bits of raw opaque data.
59 | Derived from basic 32 bit word type.
60 | "
61 | ::= { _625k-MCStatsFiles 6 }
62 |
63 |
64 |
65 | _625k-MCSysInterfaces OBJECT IDENTIFIER
66 | -- DESCRIPTION "System Interfaces"
67 | ::= { _625k-MCSystem 3 }
68 |

```

```

1
2  _625k-MCInterfaceNetwork          OBJECT IDENTIFIER
3  -- DESCRIPTION                    "Network Interfaces"
4  ::= { _625k-MCSysInterfaces 1 }
5
6
7
8  _625k-MCTypeOfNetworkProtocol     OBJECT-TYPE
9  SYNTAX                            INTEGER -- NetworkProtocolType
10 | ACCESSMAX-ACCESS                 read-only
11 | STATUS                            mandatoryCurrent
12 | DESCRIPTION
13 |     "Type of Network Protocol used with the Network.
14 |
15 |     Type of Network Protocol is Ethernet or ATM.
16 |
17 |     (From mibCtl ElementType 3002 TypeOfNetworkProtocol)
18 |     Description for mibCtl Type 214 NetworkProtocolType :
19 |         Network Protocol type.
20 |     Description for mibCtl NetworkProtocolType 0 Unknown :
21 |         Network protocol type is unknown.
22 |     Description for mibCtl NetworkProtocolType 1 Ethernet :
23 |         Ethernet interface.
24 |     Description for mibCtl NetworkProtocolType 2 ATM :
25 |         ATM interface.
26 |
27 |     "
28 | ::= { _625k-MCInterfaceNetwork 1 }
29
30
31
32 _625k-MCMgmtNetConfigTable         OBJECT-TYPE
33 SYNTAX SEQUENCE OF _625k-MCMgmtNetConfigTableEntry
34 | ACCESSMAX-ACCESS                 not-accessible
35 | STATUS                            mandatoryCurrent
36 | DESCRIPTION                        "Mgmt Network Configuration"
37 | ::= { _625k-MCInterfaceNetwork 2 }
38
39
40
41 _625k-MCMgmtNetConfigTableEntry    OBJECT-TYPE
42 SYNTAX                            _625k-MCMgmtNetConfigTableEntry
43 | ACCESSMAX-ACCESS                 not-accessible
44 | STATUS                            mandatoryCurrent
45 | DESCRIPTION                        ""
46 | INDEX { _625k-MCMgmtNetConfigTableIndex }
47 | ::= { _625k-MCMgmtNetConfigTable 1 }
48
49 _625k-MCMgmtNetConfigTableEntry ::= SEQUENCE {
50   _625k-MCMgmtNetConfigTableIndex  INTEGER, -- MoNerdAddressType
51   _625k-MCEthernetIPAddress        OCTET STRING (SIZE(0..15)), --
52   IPAddressTextType                TextType
53   _625k-MCEthernetIPLocalBits      OCTET STRING (SIZE(0..15)), --
54   IPAddressTextType                TextType
55   _625k-MCEthernetHostName         OCTET STRING (SIZE(0..20)) --
56   TextType X 20
57 }
58
59
60
61 _625k-MCMgmtNetConfigTableIndex    OBJECT-TYPE
62 SYNTAX                            INTEGER -- MoNerdAddressType
63 | ACCESSMAX-ACCESS                 read-only
64 | STATUS                            mandatoryCurrent
65 | DESCRIPTION                        ""
66 |     Description for mibCtl Type 204 MoNerdAddressType :
67 |         Base station network component address.
68

```



```

1         A network address is a subset of Base Station component addresses,
2         restricted to network components only.
3         Network components interface with a telephony switch or similar.
4         [Limits: 0 1 ]
5         Type derived from mibCtl Type 14 Unsigned32Type :
6         32 bit unsigned integer.
7         Type derived from mibCtl Type 11 Word32Type :
8         32 bits of raw opaque data.
9         Derived from basic 32 bit word type.
10        "
11        ::= { _625k-MCMgmtNetConfigTableEntry 1 }
12
13
14
15        _625k-MCEthernetIPAddress          OBJECT-TYPE
16        SYNTAX          OCTET STRING (SIZE(0..15)) -- IPAddressTextType
17        ACCESSMAX-ACCESS          read-write
18        STATUS          mandatoryCurrent
19        DESCRIPTION
20        "Internet Protocol (IP) address for ethernet port of Module.
21
22        This is the actual IP address in use for the ethernet port
23        of a given Module.
24        If IP is not being used on the ethernet port, or there is
25        no ethernet port, then an empty string is provided for this element.
26
27        (From mibCtl ElementType 2811 EthernetIPAddress)
28        Description for mibCtl Type 420 IPAddressTextType :
29        Internet Protocol Address (Text).
30
31        This text must currently be in the dotted abc.def.ghi.jkl format.
32        In the future, hostnames might be allowed.
33        Type derived from mibCtl Type 15 TextType :
34        ASCII or compatible text.
35        Type derived from mibCtl Type 12 OctetType :
36        8 bits of raw opaque data.
37        Derived from basic 8 bit word type.
38        "
39        ::= { _625k-MCMgmtNetConfigTableEntry 2 }
40
41
42
43        _625k-MCEthernetIPLocalBits        OBJECT-TYPE
44        SYNTAX          OCTET STRING (SIZE(0..15)) -- IPAddressTextType
45        ACCESSMAX-ACCESS          read-write
46        STATUS          mandatoryCurrent
47        DESCRIPTION
48        "Ethernet IP (Internet Protocol) local routing bit count.
49
50        This indicates how many of the low-order bits of
51        the IP address of the ethernet connection are used
52        within the local network.
53        The remaining (high-order) bits are the same for all
54        hosts on the local network.
55        This is used as the first part of the routing algorithm.
56        IP addresses that do not share the upper bits of the ethernet
57        IP address and which are not otherwise resolved will be sent
58        through the gateway, if defined.
59
60        For example, 255.255.255.0
61
62        (From mibCtl ElementType 2812 EthernetIPLocalBits)
63        Description for mibCtl Type 420 IPAddressTextType :
64        Internet Protocol Address (Text).
65
66        This text must currently be in the dotted abc.def.ghi.jkl format.
67        In the future, hostnames might be allowed.
68        Type derived from mibCtl Type 15 TextType :

```

```

1         ASCII or compatible text.
2         Type derived from mibCtl Type 12 OctetType :
3         8 bits of raw opaque data.
4         Derived from basic 8 bit word type.
5         "
6         ::= { _625k-MCMgmtNetConfigTableEntry 3 }
7
8
9
10        _625k-MCEthernetHostName          OBJECT-TYPE
11        SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
12        ACCESSMAX-ACCESS          read-write
13        STATUS          mandatoryCurrent
14        DESCRIPTION
15        "Ethernet IP host name for module.
16
17        (From mibCtl ElementType 2813 EthernetHostName)
18        Description for mibCtl Type 15 TextType :
19        ASCII or compatible text.
20        Type derived from mibCtl Type 12 OctetType :
21        8 bits of raw opaque data.
22        Derived from basic 8 bit word type.
23        "
24        ::= { _625k-MCMgmtNetConfigTableEntry 4 }
25
26
27
28
29        _625k-MCUserNetConfigTable        OBJECT-TYPE
30        SYNTAX SEQUENCE OF _625k-MCUserNetConfigTableEntry
31        ACCESSMAX-ACCESS          not-accessible
32        STATUS          mandatoryCurrent
33        DESCRIPTION    "User Network Configuration"
34        ::= { _625k-MCInterfaceNetwork 3 }
35
36
37
38        _625k-MCUserNetConfigTableEntry    OBJECT-TYPE
39        SYNTAX          _625k-MCUserNetConfigTableEntry
40        ACCESSMAX-ACCESS          not-accessible
41        STATUS          mandatoryCurrent
42        DESCRIPTION    ""
43        INDEX { _625k-MCUserNetConfigTableIndex }
44        ::= { _625k-MCUserNetConfigTable 1 }
45
46        _625k-MCUserNetConfigTableEntry ::= SEQUENCE {
47        _625k-MCUserNetConfigTableIndex    INTEGER, -- MoNerdAddressType
48        _625k-MCUserEthernetIPAddress      OCTET STRING (SIZE(0..15)), --
49        IPAddressTextType
50        _625k-MCUserEthernetIPLocalBits    OCTET STRING (SIZE(0..15)), --
51        IPAddressTextType
52        _625k-MCUserEthernetHostName       OCTET STRING (SIZE(0..20)) --
53        TextType X 20
54        }
55
56
57
58        _625k-MCUserNetConfigTableIndex    OBJECT-TYPE
59        SYNTAX          INTEGER -- MoNerdAddressType
60        ACCESSMAX-ACCESS          read-only
61        STATUS          mandatoryCurrent
62        DESCRIPTION    "
63        Description for mibCtl Type 204 MoNerdAddressType :
64        Base station network component address.
65
66        A network address is a subset of Base Station component addresses,
67        restricted to network components only.
68        Network components interface with a telephony switch or similar.

```

```

1         [Limits: 0 1 ]
2         Type derived from mibCtl Type 14 Unsigned32Type :
3         32 bit unsigned integer.
4         Type derived from mibCtl Type 11 Word32Type :
5         32 bits of raw opaque data.
6         Derived from basic 32 bit word type.
7     "
8     ::= { _625k-MCUserNetConfigTableEntry 1 }
9
10
11
12     _625k-MCUserEthernetIPAddress                OBJECT-TYPE
13     SYNTAX OCTET STRING (SIZE(0..15)) -- IPAddressTextType
14     ACCESSMAX-ACCESS read-write
15     STATUS mandatoryCurrent
16     DESCRIPTION
17         "Internet Protocol (IP) address for user ethernet port of Module.
18
19         This is the actual IP address in use for the ethernet port
20         of a given Module.
21         If IP is not being used on the ethernet port, or there is
22         no ethernet port, then an empty string is provided for this element.
23
24         (From mibCtl ElementType 2817 UserEthernetIPAddress)
25         Description for mibCtl Type 420 IPAddressTextType :
26         Internet Protocol Address (Text).
27
28         This text must currently be in the dotted abc.def.ghi.jkl format.
29         In the future, hostnames might be allowed.
30         Type derived from mibCtl Type 15 TextType :
31         ASCII or compatible text.
32         Type derived from mibCtl Type 12 OctetType :
33         8 bits of raw opaque data.
34         Derived from basic 8 bit word type.
35     "
36     ::= { _625k-MCUserNetConfigTableEntry 2 }
37
38
39
40     _625k-MCUserEthernetIPLocalBits                OBJECT-TYPE
41     SYNTAX OCTET STRING (SIZE(0..15)) -- IPAddressTextType
42     ACCESSMAX-ACCESS read-write
43     STATUS mandatoryCurrent
44     DESCRIPTION
45         "Ethernet IP subnet mask for user network.
46
47         Ethernet IP subnet mask for user network
48
49         (From mibCtl ElementType 2818 UserEthernetIPLocalBits)
50         Description for mibCtl Type 420 IPAddressTextType :
51         Internet Protocol Address (Text).
52
53         This text must currently be in the dotted abc.def.ghi.jkl format.
54         In the future, hostnames might be allowed.
55         Type derived from mibCtl Type 15 TextType :
56         ASCII or compatible text.
57         Type derived from mibCtl Type 12 OctetType :
58         8 bits of raw opaque data.
59         Derived from basic 8 bit word type.
60     "
61     ::= { _625k-MCUserNetConfigTableEntry 3 }
62
63
64
65     _625k-MCUserEthernetHostName                OBJECT-TYPE
66     SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
67     ACCESSMAX-ACCESS read-write
68     STATUS mandatoryCurrent

```

```

1 DESCRIPTION
2     "User ethernet IP host name for module.
3
4     (From mibCtl ElementType 2819 UserEthernetHostName)
5     Description for mibCtl Type 15 TextType :
6         ASCII or compatible text.
7     Type derived from mibCtl Type 12 OctetType :
8         8 bits of raw opaque data.
9     Derived from basic 8 bit word type.
10    "
11    ::= { _625k-MCUserNetConfigTableEntry 4 }
12
13
14
15
16    _625k-MCUserNetStatusTable                OBJECT-TYPE
17    SYNTAX SEQUENCE OF _625k-MCUserNetStatusTableEntry
18    ACCESSMAX-ACCESS                not-accessible
19    STATUS mandatoryCurrent
20    DESCRIPTION "Network Status"
21    ::= { _625K-MCInterfaceNetwork 4 }
22
23
24
25    _625k-MCUserNetStatusTableEntry            OBJECT-TYPE
26    SYNTAX _625k-MCUserNetStatusTableEntry
27    ACCESSMAX-ACCESS                not-accessible
28    STATUS mandatoryCurrent
29    DESCRIPTION ""
30    INDEX { _625k-MCUserNetStatusTableIndex }
31    ::= { _625k-MCUserNetStatusTable 1 }
32
33    _625k-MCUserNetStatusTableEntry ::= SEQUENCE {
34        _625k-MCUserNetStatusTableIndex        INTEGER, -- MoNerdAddressType
35        _625k-MCNetworkInOctets                Counter, -- Counter32Type
36        _625k-MCNetworkOutOctets              Counter -- Counter32Type
37    }
38
39
40
41    _625k-MCUserNetStatusTableIndex            OBJECT-TYPE
42    SYNTAX INTEGER -- MoNerdAddressType
43    ACCESSMAX-ACCESS                read-only
44    STATUS mandatoryCurrent
45    DESCRIPTION "
46        Description for mibCtl Type 204 MoNerdAddressType :
47        Base station network component address.
48
49        A network address is a subset of Base Station component addresses,
50        restricted to network components only.
51        Network components interface with a telephony switch or similar.
52        [Limits: 0 1 ]
53        Type derived from mibCtl Type 14 Unsigned32Type :
54        32 bit unsigned integer.
55        Type derived from mibCtl Type 11 Word32Type :
56        32 bits of raw opaque data.
57        Derived from basic 32 bit word type.
58    "
59    ::= { _625k-MCUserNetStatusTableEntry 1 }
60
61
62
63    _625k-MCNetworkInOctets                    OBJECT-TYPE
64    SYNTAX Counter32 -- Counter32Type
65    ACCESSMAX-ACCESS                read-only
66    STATUS mandatoryCurrent
67    DESCRIPTION
68    "In octets user data of network.

```

```

1
2
3
4     (From mibCtl ElementType 1000 NetworkInOctets)
5     Description for mibCtl Type 19 Counter32Type :
6         32 bits of Counter data.
7     Derived from basic 32 bit word type.
8     "
9     ::= { _625k-MCUserNetStatusTableEntry 2 }
10
11
12
13 _625k-MCNetworkOutOctets          OBJECT-TYPE
14 | SYNTAX          Counter32 -- Counter32Type
15 | ACCESSMAX-ACCESS      read-only
16 | STATUS          mandatoryCurrent
17 | DESCRIPTION
18 |     "Out octets user data of network.
19 |
20 |
21 |
22     (From mibCtl ElementType 1001 NetworkOutOctets)
23     Description for mibCtl Type 19 Counter32Type :
24         32 bits of Counter data.
25     Derived from basic 32 bit word type.
26     "
27     ::= { _625k-MCUserNetStatusTableEntry 3 }
28
29
30
31
32 _625k-MCL2TPConfigTable          OBJECT-TYPE
33 | SYNTAX SEQUENCE OF _625k-MCL2TPConfigTableEntry
34 | ACCESSMAX-ACCESS      not-accessible
35 | STATUS          mandatoryCurrent
36 | DESCRIPTION    "L2TP Configuration Table"
37 | ::= { _625k-MCInterfaceNetwork 5 }
38
39
40
41
42 _625k-MCL2TPConfigTableEntry     OBJECT-TYPE
43 | SYNTAX          _625k-MCL2TPConfigTableEntry
44 | ACCESSMAX-ACCESS      not-accessible
45 | STATUS          mandatoryCurrent
46 | DESCRIPTION    ""
47 | INDEX { _625k-MCL2TPConfigTableIndex }
48 | ::= { _625k-MCL2TPConfigTable 1 }
49
50 _625k-MCL2TPConfigTableEntry ::= SEQUENCE {
51     _625k-MCL2TPConfigTableIndex    INTEGER, -- MoNerdAddressType
52     _625k-MCL2TPPeerName            OCTET STRING (SIZE(0..20)), --
53     TextType X 20
54     _625k-MCL2TPPeerIPAddress       OCTET STRING (SIZE(0..15)), --
55     IPAddressTextType
56     _625k-MCL2TPAVPHostName         OCTET STRING (SIZE(0..20)), --
57     TextType X 20
58     _625k-MCL2TPAVPChallAndRes      OCTET STRING (SIZE(0..20)) --
59     TextType X 20
60 }
61
62
63
64 _625k-MCL2TPConfigTableIndex     OBJECT-TYPE
65 | SYNTAX          INTEGER -- MoNerdAddressType
66 | ACCESSMAX-ACCESS      read-only
67 | STATUS          mandatoryCurrent
68 | DESCRIPTION    "
        Description for mibCtl Type 204 MoNerdAddressType :

```

```

1         Base station network component address.
2
3         A network address is a subset of Base Station component addresses,
4         restricted to network components only.
5         Network components interface with a telephony switch or similar.
6         [Limits: 0 1 ]
7         Type derived from mibCtl Type 14 Unsigned32Type :
8         32 bit unsigned integer.
9         Type derived from mibCtl Type 11 Word32Type :
10        32 bits of raw opaque data.
11        Derived from basic 32 bit word type.
12        "
13        ::= { _625k-MCL2TPConfigTableEntry 1 }
14
15
16
17        _625k-MCL2TPPeerName                                OBJECT-TYPE
18        SYNTAX      OCTET STRING (SIZE(0..20)) -- TextType X 20
19        ACCESSMAX-ACCESS      read-write
20        STATUS      mandatoryCurrent
21        DESCRIPTION
22            "L2TP peer name.
23
24            Tunnel switch host name
25
26            (From mibCtl ElementType 2000 L2TPPeerName)
27            Description for mibCtl Type 15 TextType :
28            ASCII or compatible text.
29            Type derived from mibCtl Type 12 OctetType :
30            8 bits of raw opaque data.
31            Derived from basic 8 bit word type.
32            "
33            ::= { _625k-MCL2TPConfigTableEntry 2 }
34
35
36
37        _625k-MCL2TPPeerIPAddress                            OBJECT-TYPE
38        SYNTAX      OCTET STRING (SIZE(0..15)) -- IPAddressTextType
39        ACCESSMAX-ACCESS      read-write
40        STATUS      mandatoryCurrent
41        DESCRIPTION
42            "L2TP peer IP Address.
43
44            Tunnel switch IP Address
45
46            (From mibCtl ElementType 2001 L2TPPeerIPAddress)
47            Description for mibCtl Type 420 IPAddressTextType :
48            Internet Protocol Address (Text).
49
50            This text must currently be in the dotted abc.def.ghi.jkl format.
51            In the future, hostnames might be allowed.
52            Type derived from mibCtl Type 15 TextType :
53            ASCII or compatible text.
54            Type derived from mibCtl Type 12 OctetType :
55            8 bits of raw opaque data.
56            Derived from basic 8 bit word type.
57            "
58            ::= { _625k-MCL2TPConfigTableEntry 3 }
59
60
61
62        _625k-MCL2TPAVPHostName                              OBJECT-TYPE
63        SYNTAX      OCTET STRING (SIZE(0..20)) -- TextType X 20
64        ACCESSMAX-ACCESS      read-write
65        STATUS      mandatoryCurrent
66        DESCRIPTION
67            "BS host name using L2TP.
68

```

```

1
2
3      (From mibCtl ElementType 2008 L2TPAVPHostName)
4      Description for mibCtl Type 15 TextType :
5          ASCII or compatible text.
6      Type derived from mibCtl Type 12 OctetType :
7          8 bits of raw opaque data.
8      Derived from basic 8 bit word type.
9      "
10     ::= { _625k-MCL2TPConfigTableEntry 4 }
11
12
13
14     _625k-MCL2TPAVPChallAndRes          OBJECT-TYPE
15     SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
16     ACCESSMAX-ACCESS          read-write
17     STATUS          mandatoryCurrent
18     DESCRIPTION
19         "AVP challenge and response name.
20
21
22
23     (From mibCtl ElementType 2012 L2TPAVPChallAndRes)
24     Description for mibCtl Type 15 TextType :
25         ASCII or compatible text.
26     Type derived from mibCtl Type 12 OctetType :
27         8 bits of raw opaque data.
28     Derived from basic 8 bit word type.
29     "
30     ::= { _625k-MCL2TPConfigTableEntry 5 }
31
32
33
34
35     _625k-MCL2TPStatusTable              OBJECT-TYPE
36     SYNTAX SEQUENCE OF _625k-MCL2TPStatusTableEntry
37     ACCESSMAX-ACCESS          not-accessible
38     STATUS          mandatoryCurrent
39     DESCRIPTION
40         "L2TP Status Table"
41     ::= { _625k-MCInterfaceNetwork 6 }
42
43
44
45     _625k-MCL2TPStatusTableEntry         OBJECT-TYPE
46     SYNTAX          _625k-MCL2TPStatusTableEntry
47     ACCESSMAX-ACCESS          not-accessible
48     STATUS          mandatoryCurrent
49     DESCRIPTION
50         ""
51     INDEX { _625k-MCL2TPStatusTableIndex }
52     ::= { _625k-MCL2TPStatusTable 1 }
53
54     _625k-MCL2TPStatusTableEntry ::= SEQUENCE {
55     _625k-MCL2TPStatusTableIndex        INTEGER, -- MoNerdAddressType
56     _625k-MCL2TPActiveSession           Unsigned32INTEGER, -- Unsigned32Type
57     _625k-MCL2TPActiveTunnel            Unsigned32INTEGER -- Unsigned32Type
58     }
59
60
61     _625k-MCL2TPStatusTableIndex         OBJECT-TYPE
62     SYNTAX          INTEGER -- MoNerdAddressType
63     ACCESSMAX-ACCESS          read-only
64     STATUS          mandatoryCurrent
65     DESCRIPTION
66         "
67         Description for mibCtl Type 204 MoNerdAddressType :
68         Base station network component address.
69
70         A network address is a subset of Base Station component addresses,

```

```

1         restricted to network components only.
2         Network components interface with a telephony switch or similar.
3         [Limits: 0 1 ]
4         Type derived from mibCtl Type 14 Unsigned32Type :
5         32 bit unsigned integer.
6         Type derived from mibCtl Type 11 Word32Type :
7         32 bits of raw opaque data.
8         Derived from basic 32 bit word type.
9     "
10    ::= { _625k-MCL2TPStatusTableEntry 1 }
11
12
13
14    _625k-MCL2TPActiveSession          OBJECT-TYPE
15    | SYNTAX Unsigned32INTEGER -- Unsigned32Type
16    | ACCESSMAX-ACCESS                read-only
17    | STATUS mandatoryCurrent
18    | DESCRIPTION
19    |     "L2TP active session.
20
21
22
23    (From mibCtl ElementType 2013 L2TPActiveSession)
24    Description for mibCtl Type 14 Unsigned32Type :
25    32 bit unsigned integer.
26    Type derived from mibCtl Type 11 Word32Type :
27    32 bits of raw opaque data.
28    Derived from basic 32 bit word type.
29    "
30    ::= { _625k-MCL2TPStatusTableEntry 2 }
31
32
33
34    _625k-MCL2TPActiveTunnel          OBJECT-TYPE
35    | SYNTAX INTEGERUnsigned32 -- Unsigned32Type
36    | ACCESSMAX-ACCESS                read-only
37    | STATUS mandatoryCurrent
38    | DESCRIPTION
39    |     "L2TP active tunnel.
40
41
42
43    (From mibCtl ElementType 2014 L2TPActiveTunnel)
44    Description for mibCtl Type 14 Unsigned32Type :
45    32 bit unsigned integer.
46    Type derived from mibCtl Type 11 Word32Type :
47    32 bits of raw opaque data.
48    Derived from basic 32 bit word type.
49    "
50    ::= { _625k-MCL2TPStatusTableEntry 3 }
51
52
53
54
55    _625k-MCATMConfigTable            OBJECT-TYPE
56    | SYNTAX SEQUENCE OF _625k-MCATMConfigTableEntry
57    | ACCESSMAX-ACCESS                not-accessible
58    | STATUS mandatoryCurrent
59    | DESCRIPTION "ATM Configuration Table"
60    | ::= { _625k-MCInterfaceNetwork 7 }
61
62
63
64
65    _625k-MCATMConfigTableEntry        OBJECT-TYPE
66    | SYNTAX _625k-MCATMConfigTableEntry
67    | ACCESSMAX-ACCESS                not-accessible
68    | STATUS mandatoryCurrent
69    | DESCRIPTION ""

```



```

1      INDEX { _625k-MCATMConfigTableIndex }
2      ::= { _625k-MCATMConfigTable 1 }
3
4      _625k-MCATMConfigTableEntry ::= SEQUENCE {
5          _625k-MCATMConfigTableIndex      INTEGER, -- MoNerdAddressType
6          _625k-MCATmAddress                OCTET STRING (SIZE(0..40)), --
7          TextType X 40
8          _625k-MCATmVCTypes                INTEGER, -- AtmVCType
9          _625k-MCATmFrameTypes             INTEGER, -- AtmFrameType
10         _625k-MCATmUNIVersion              INTEGER, -- AtmUNIVersionType
11         _625k-MCATmLineStatus              INTEGER, -- LineStatusType
12         _625k-MCATmParameterFailReason    INTEGERUnsigned32, -- Unsigned32Type
13         _625k-MCATmOpenChannelFailReason  INTEGERUnsigned32, -- Unsigned32Type
14         _625k-MCATmChannelNumber          INTEGERUnsigned32, -- Unsigned32Type
15         _625k-MCATmAlarmCauseRegister     INTEGERUnsigned32, -- Unsigned32Type
16         _625k-MCATmPHYIntrCauseRegister   INTEGERUnsigned32 -- Unsigned32Type
17     }
18
19
20
21     _625k-MCATMConfigTableIndex            OBJECT-TYPE
22     SYNTAX                                INTEGER -- MoNerdAddressType
23     ACCESSMAX-ACCESS                      read-only
24     STATUS                                mandatoryCurrent
25     DESCRIPTION                            "
26         Description for mibCtl Type 204 MoNerdAddressType :
27         Base station network component address.
28
29         A network address is a subset of Base Station component addresses,
30         restricted to network components only.
31         Network components interface with a telephony switch or similar.
32         [Limits: 0 1 ]
33         Type derived from mibCtl Type 14 Unsigned32Type :
34         32 bit unsigned integer.
35         Type derived from mibCtl Type 11 Word32Type :
36         32 bits of raw opaque data.
37         Derived from basic 32 bit word type.
38     "
39     ::= { _625k-MCATMConfigTableEntry 1 }
40
41
42
43     _625k-MCATmAddress                      OBJECT-TYPE
44     SYNTAX                                OCTET STRING (SIZE(0..40)) -- TextType X 40
45     ACCESSMAX-ACCESS                      read-write
46     STATUS                                mandatoryCurrent
47     DESCRIPTION                            "
48         "Atm Address.
49
50
51         (From mibCtl ElementType 1950 AtmAddress)
52         Description for mibCtl Type 15 TextType :
53         ASCII or compatible text.
54         Type derived from mibCtl Type 12 OctetType :
55         8 bits of raw opaque data.
56         Derived from basic 8 bit word type.
57     "
58     ::= { _625k-MCATMConfigTableEntry 2 }
59
60
61
62
63     _625k-MCATmVCTypes                      OBJECT-TYPE
64     SYNTAX                                INTEGER -- AtmVCType
65     ACCESSMAX-ACCESS                      read-write
66     STATUS                                mandatoryCurrent
67     DESCRIPTION                            "
68         "Atm VC Type.

```

```

1
2
3
4      (From mibCtl ElementType 1951 AtmVCTypes)
5      Description for mibCtl Type 300 AtmVCType :
6          ATM VC Type.
7      Description for mibCtl AtmVCType 0 Unknown :
8          ATM VC Type is Unknown.
9      Description for mibCtl AtmVCType 1 PVC :
10         ATM VC Type is PVC.
11     Description for mibCtl AtmVCType 2 SVC :
12         ATM VC Type is SVC.
13     Description for mibCtl AtmVCType 3 PVCSVC :
14         ATM VC Type is PVC & SVC.
15     "
16     ::= { _625k-MCATMConfigTableEntry 3 }
17
18
19
20     _625k-MCATmFrameTypes                OBJECT-TYPE
21     SYNTAX                               INTEGER -- AtmFrameType
22     ACCESSMAX-ACCESS                   read-write
23     STATUS                               mandatoryCurrent
24     DESCRIPTION
25         "Atm Frame Type.
26
27
28
29     (From mibCtl ElementType 1952 AtmFrameTypes)
30     Description for mibCtl Type 301 AtmFrameType :
31         ATM Frame Type.
32     Description for mibCtl AtmFrameType 0 Unknown :
33         ATM Frame Type is Unknown.
34     Description for mibCtl AtmFrameType 1 OC48 :
35         ATM Frame Type is OC48.
36     Description for mibCtl AtmFrameType 2 OC36 :
37         ATM Frame Type is OC36.
38     Description for mibCtl AtmFrameType 3 OC24 :
39         ATM Frame Type is OC24.
40     Description for mibCtl AtmFrameType 4 OC18 :
41         ATM Frame Type is OC18.
42     Description for mibCtl AtmFrameType 5 OC12 :
43         ATM Frame Type is OC12.
44     Description for mibCtl AtmFrameType 6 OC9 :
45         ATM Frame Type is OC9.
46     Description for mibCtl AtmFrameType 7 OC3 :
47         ATM Frame Type is OC3.
48     Description for mibCtl AtmFrameType 8 OC1 :
49         ATM Frame Type is OC1.
50     Description for mibCtl AtmFrameType 9 STM16 :
51         ATM Frame Type is STM16.
52     Description for mibCtl AtmFrameType 10 STM4 :
53         ATM Frame Type is STM4.
54     Description for mibCtl AtmFrameType 11 STM1 :
55         ATM Frame Type is STM1.
56     Description for mibCtl AtmFrameType 12 DS3 :
57         ATM Frame Type is DS3.
58     Description for mibCtl AtmFrameType 13 DS2 :
59         ATM Frame Type is DS2.
60     Description for mibCtl AtmFrameType 14 DS1 :
61         ATM Frame Type is DS1.
62     Description for mibCtl AtmFrameType 15 DS0 :
63         ATM Frame Type is DS0.
64     Description for mibCtl AtmFrameType 16 E3 :
65         ATM Frame Type is E3.
66     Description for mibCtl AtmFrameType 17 E2 :
67         ATM Frame Type is E2.
68     Description for mibCtl AtmFrameType 18 E1 :

```

```

1         ATM Frame Type is E1.
2         Description for mibCtl AtmFrameType 19 E0 :
3         ATM Frame Type is E0.
4         "
5         ::= { _625k-MCATMConfigTableEntry 4 }
6
7
8
9         _625k-MCAtmUNIVersion                                OBJECT-TYPE
10        SYNTAX          INTEGER -- AtmUNIVersionType
11        ACCESSMAX-ACCESS      read-write
12        STATUS          mandatoryCurrent
13        DESCRIPTION
14            "Atm UNI version.
15
16
17            (From mibCtl ElementType 1953 AtmUNIVersion)
18            Description for mibCtl Type 302 AtmUNIVersionType :
19            ATM UNI Version Type.
20            Description for mibCtl AtmUNIVersionType 0 Unknown :
21            ATM UNI Version Type is Unknown.
22            Description for mibCtl AtmUNIVersionType 1 V30 :
23            ATM UNI Version Type is 3.0.
24            Description for mibCtl AtmUNIVersionType 2 V31 :
25            ATM UNI Version Type is 3.1.
26            Description for mibCtl AtmUNIVersionType 3 V40 :
27            ATM UNI Version Type is 4.0.
28            "
29        ::= { _625k-MCATMConfigTableEntry 5 }
30
31
32
33
34        _625k-MCAtmLineStatus                                OBJECT-TYPE
35        SYNTAX          INTEGER -- LineStatusType
36        ACCESSMAX-ACCESS      read-only
37        STATUS          mandatoryCurrent
38        DESCRIPTION
39            "Atm line status.
40
41            ATM line status
42
43            (From mibCtl ElementType 1956 AtmLineStatus)
44            Description for mibCtl Type 72 LineStatusType :
45            Line status type.
46            Description for mibCtl LineStatusType 0 LinkUp :
47            Line status is link up.
48            Description for mibCtl LineStatusType 1 LinkDown :
49            Line status is link down.
50            "
51        ::= { _625k-MCATMConfigTableEntry 6 }
52
53
54
55        _625k-MCAtmParameterFailReason                      OBJECT-TYPE
56        SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
57        ACCESSMAX-ACCESS      read-only
58        STATUS          mandatoryCurrent
59        DESCRIPTION
60            "Atm parameter fail reason.
61
62            ATM parameter fail reason
63
64            (From mibCtl ElementType 1957 AtmParameterFailReason)
65            Description for mibCtl Type 14 Unsigned32Type :
66            32 bit unsigned integer.
67            Type derived from mibCtl Type 11 Word32Type :
68            32 bits of raw opaque data.

```

```

1         Derived from basic 32 bit word type.
2         "
3         ::= { _625k-MCATMConfigTableEntry 7 }
4
5
6
7         _625k-MCATmOpenChannelFailReason          OBJECT-TYPE
8         SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
9         ACCESSMAX-ACCESS          read-only
10        STATUS          mandatoryCurrent
11        DESCRIPTION
12        "Atm open channel fail reason.
13
14        ATM open channel fail reason
15
16        (From mibCtl ElementType 1958 AtmOpenChannelFailReason)
17        Description for mibCtl Type 14 Unsigned32Type :
18        32 bit unsigned integer.
19        Type derived from mibCtl Type 11 Word32Type :
20        32 bits of raw opaque data.
21        Derived from basic 32 bit word type.
22        "
23        ::= { _625k-MCATMConfigTableEntry 8 }
24
25
26
27        _625k-MCATmChannelNumber                    OBJECT-TYPE
28        SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
29        ACCESSMAX-ACCESS          read-only
30        STATUS          mandatoryCurrent
31        DESCRIPTION
32        "Atm open channel fail reason.
33
34        ATM open channel fail reason
35
36        (From mibCtl ElementType 1959 AtmChannelNumber)
37        Description for mibCtl Type 14 Unsigned32Type :
38        32 bit unsigned integer.
39        Type derived from mibCtl Type 11 Word32Type :
40        32 bits of raw opaque data.
41        Derived from basic 32 bit word type.
42        "
43        ::= { _625k-MCATMConfigTableEntry 9 }
44
45
46
47        _625k-MCATmAlarmCauseRegister              OBJECT-TYPE
48        SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
49        ACCESSMAX-ACCESS          read-only
50        STATUS          mandatoryCurrent
51        DESCRIPTION
52        "Atm alarm cause register.
53
54        ATM alarm cause register
55
56        (From mibCtl ElementType 1960 AtmAlarmCauseRegister)
57        Description for mibCtl Type 14 Unsigned32Type :
58        32 bit unsigned integer.
59        Type derived from mibCtl Type 11 Word32Type :
60        32 bits of raw opaque data.
61        Derived from basic 32 bit word type.
62        "
63        ::= { _625k-MCATMConfigTableEntry 10 }
64
65
66
67        _625k-MCATmPHYIntrCauseRegister            OBJECT-TYPE
68        SYNTAX          INTEGERUnsigned32 -- Unsigned32Type

```

```

1 | ACCESSMAX-ACCESS read-only
2 | STATUS mandatoryCurrent
3 | DESCRIPTION
4 | "Atm PHY Interrupt cause register.
5 |
6 | ATM PHY interrupt cause register
7 |
8 | (From mibCtl ElementType 1961 AtmPHYIntrCauseRegister)
9 | Description for mibCtl Type 14 Unsigned32Type :
10 | 32 bit unsigned integer.
11 | Type derived from mibCtl Type 11 Word32Type :
12 | 32 bits of raw opaque data.
13 | Derived from basic 32 bit word type.
14 | "
15 | ::= { _625k-MCATMConfigTableEntry 11 }
16 |
17 |
18 |
19 |
20 | _625k-MCA10ConfigTable OBJECT-TYPE
21 | SYNTAX SEQUENCE OF _625k-MCA10ConfigTableEntry
22 | ACCESSMAX-ACCESS not-accessible
23 | STATUS mandatoryCurrent
24 | DESCRIPTION "A10 Configuration Table"
25 | ::= { _625k-MCInterfaceNetwork 9 }
26 |
27 |
28 |
29 | _625k-MCA10ConfigTableEntry OBJECT-TYPE
30 | SYNTAX _625k-MCA10ConfigTableEntry
31 | ACCESSMAX-ACCESS not-accessible
32 | STATUS mandatoryCurrent
33 | DESCRIPTION ""
34 | INDEX { _625k-MCA10ConfigTableIndex }
35 | ::= { _625k-MCA10ConfigTable 1 }
36 |
37 | _625k-MCA10ConfigTableEntry ::= SEQUENCE {
38 |   _625k-MCA10ConfigTableIndex INTEGER, -- MoNerdAddressType
39 |   _625k-MC3GPP2PriPDSNIPAddress OCTET STRING (SIZE(0..15)), --
40 |   IPAddressTextType
41 |   _625k-MC3GPP2PriPDSNSharedSecret OCTET STRING (SIZE(0..64)), --
42 |   TextType X 64
43 |   _625k-MC3GPP2PriPDSNSPI INTEGERUnsigned32, -- Unsigned32Type
44 |   _625k-MC3GPP2SecPDSNIPAddress OCTET STRING (SIZE(0..15)), --
45 |   IPAddressTextType
46 |   _625k-MC3GPP2SecPDSNSharedSecret OCTET STRING (SIZE(0..64)), --
47 |   TextType X 64
48 |   _625k-MC3GPP2SecPDSNSPI INTEGERUnsigned32 -- Unsigned32Type
49 | }
50 |
51 |
52 |
53 | _625k-MCA10ConfigTableIndex OBJECT-TYPE
54 | SYNTAX INTEGER -- MoNerdAddressType
55 | ACCESSMAX-ACCESS read-only
56 | STATUS mandatoryCurrent
57 | DESCRIPTION "
58 | Description for mibCtl Type 204 MoNerdAddressType :
59 | Base station network component address.
60 |
61 | A network address is a subset of Base Station component addresses,
62 | restricted to network components only.
63 | Network components interface with a telephony switch or similar.
64 | [Limits: 0 1 ]
65 | Type derived from mibCtl Type 14 Unsigned32Type :
66 | 32 bit unsigned integer.
67 | Type derived from mibCtl Type 11 Word32Type :
68 | 32 bits of raw opaque data.

```

```

1         Derived from basic 32 bit word type.
2     "
3     ::= { _625k-MCA10ConfigTableEntry 1 }
4
5
6
7     _625k-MC3GPP2PriPDSNIPAddress          OBJECT-TYPE
8     SYNTAX          OCTET STRING (SIZE(0..15)) -- IPAddressTextType
9     ACCESSMAX-ACCESS          read-write
10    STATUS          mandatoryCurrent
11    DESCRIPTION
12        "3GPP2 primary PDSN IP address.
13
14        3GPP2 Primary PDSN IP Address
15
16        (From mibCtl ElementType 2100 3GPP2PriPDSNIPAddress)
17        Description for mibCtl Type 420 IPAddressTextType :
18            Internet Protocol Address (Text).
19
20        This text must currently be in the dotted abc.def.ghi.jkl format.
21        In the future, hostnames might be allowed.
22        Type derived from mibCtl Type 15 TextType :
23            ASCII or compatible text.
24        Type derived from mibCtl Type 12 OctetType :
25            8 bits of raw opaque data.
26        Derived from basic 8 bit word type.
27    "
28    ::= { _625k-MCA10ConfigTableEntry 2 }
29
30
31
32    _625k-MC3GPP2PriPDSNSharedSecret        OBJECT-TYPE
33    SYNTAX          OCTET STRING (SIZE(0..64)) -- TextType X 64
34    ACCESSMAX-ACCESS          read-write
35    STATUS          mandatoryCurrent
36    DESCRIPTION
37        "3GPP2 primary PDSN shared secret.
38
39        3GPP2 primary PDSN shared secret
40
41        (From mibCtl ElementType 2101 3GPP2PriPDSNSharedSecret)
42        Description for mibCtl Type 15 TextType :
43            ASCII or compatible text.
44        Type derived from mibCtl Type 12 OctetType :
45            8 bits of raw opaque data.
46        Derived from basic 8 bit word type.
47    "
48    ::= { _625k-MCA10ConfigTableEntry 3 }
49
50
51
52    _625k-MC3GPP2PriPDSNSPI                OBJECT-TYPE
53    SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
54    ACCESSMAX-ACCESS          read-write
55    STATUS          mandatoryCurrent
56    DESCRIPTION
57        "3GPP2 primary PDSN SPI.
58
59        3GPP2 primary PDSN SPI
60
61        (From mibCtl ElementType 2102 3GPP2PriPDSNSPI)
62        Description for mibCtl Type 14 Unsigned32Type :
63            32 bit unsigned integer.
64        Type derived from mibCtl Type 11 Word32Type :
65            32 bits of raw opaque data.
66        Derived from basic 32 bit word type.
67    "
68    ::= { _625k-MCA10ConfigTableEntry 4 }

```

```

1
2
3
4  _625k-MC3GPP2SecPDSNIPAddress          OBJECT-TYPE
5      SYNTAX          OCTET STRING (SIZE(0..15)) -- IPAddressTextType
6      ACCESSMAX-ACCESS          read-write
7      STATUS          mandatoryCurrent
8      DESCRIPTION
9          "3GPP2 secondary PDSN IP address.
10
11          3GPP2 secondary PDSN IP address
12
13          (From mibCtl ElementType 2103 3GPP2SecPDSNIPAddress)
14          Description for mibCtl Type 420 IPAddressTextType :
15              Internet Protocol Address (Text).
16
17              This text must currently be in the dotted abc.def.ghi.jkl format.
18              In the future, hostnames might be allowed.
19          Type derived from mibCtl Type 15 TextType :
20              ASCII or compatible text.
21          Type derived from mibCtl Type 12 OctetType :
22              8 bits of raw opaque data.
23          Derived from basic 8 bit word type.
24          "
25      ::= { _625k-MCA10ConfigTableEntry 5 }
26
27
28
29  _625k-MC3GPP2SecPDSNSharedSecret        OBJECT-TYPE
30      SYNTAX          OCTET STRING (SIZE(0..64)) -- TextType X 64
31      ACCESSMAX-ACCESS          read-write
32      STATUS          mandatoryCurrent
33      DESCRIPTION
34          "3GPP2 secondary PDSN shared secret.
35
36          3GPP2 secondary PDSN shared secret
37
38          (From mibCtl ElementType 2104 3GPP2SecPDSNSharedSecret)
39          Description for mibCtl Type 15 TextType :
40              ASCII or compatible text.
41          Type derived from mibCtl Type 12 OctetType :
42              8 bits of raw opaque data.
43          Derived from basic 8 bit word type.
44          "
45      ::= { _625k-MCA10ConfigTableEntry 6 }
46
47
48
49  _625k-MC3GPP2SecPDSNSPI                 OBJECT-TYPE
50      SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
51      ACCESSMAX-ACCESS          read-write
52      STATUS          mandatoryCurrent
53      DESCRIPTION
54          "3GPP2 secondary PDSN SPI.
55
56          3GPP2 secondary PDSN SPI
57
58          (From mibCtl ElementType 2105 3GPP2SecPDSNSPI)
59          Description for mibCtl Type 14 Unsigned32Type :
60              32 bit unsigned integer.
61          Type derived from mibCtl Type 11 Word32Type :
62              32 bits of raw opaque data.
63          Derived from basic 32 bit word type.
64          "
65      ::= { _625k-MCA10ConfigTableEntry 7 }
66
67
68

```

```

1
2  _625k-MCA10StatusTable                                OBJECT-TYPE
3  SYNTAX SEQUENCE OF _625k-MCA10StatusTableEntry
4  ACCESSMAX-ACCESS                                not-accessible
5  STATUS mandatoryCurrent
6  DESCRIPTION "A10 Status Table"
7  ::= { _625k-MCInterfaceNetwork 10 }
8
9
10
11 _625k-MCA10StatusTableEntry                            OBJECT-TYPE
12 SYNTAX _625k-MCA10StatusTableEntry
13 ACCESSMAX-ACCESS                                not-accessible
14 STATUS mandatoryCurrent
15 DESCRIPTION ""
16 INDEX { _625k-MCA10StatusTableIndex }
17 ::= { _625k-MCA10StatusTable 1 }
18
19 _625k-MCA10StatusTableEntry ::= SEQUENCE {
20   _625k-MCA10StatusTableIndex    INTEGER, -- MoNerdAddressType
21   _625k-MC3GPP2PDSNIPAddress    OCTET STRING (SIZE(0..15)) --
22   IPAddressTextType
23 }
24
25
26
27 _625k-MCA10StatusTableIndex                            OBJECT-TYPE
28 SYNTAX INTEGER -- MoNerdAddressType
29 ACCESSMAX-ACCESS                                read-only
30 STATUS mandatoryCurrent
31 DESCRIPTION "
32   Description for mibCtl Type 204 MoNerdAddressType :
33   Base station network component address.
34
35   A network address is a subset of Base Station component addresses,
36   restricted to network components only.
37   Network components interface with a telephony switch or similar.
38   [Limits: 0 1 ]
39   Type derived from mibCtl Type 14 Unsigned32Type :
40   32 bit unsigned integer.
41   Type derived from mibCtl Type 11 Word32Type :
42   32 bits of raw opaque data.
43   Derived from basic 32 bit word type.
44   "
45   ::= { _625k-MCA10StatusTableEntry 1 }
46
47
48
49 _625k-MC3GPP2PDSNIPAddress                            OBJECT-TYPE
50 SYNTAX OCTET STRING (SIZE(0..15)) -- IPAddressTextType
51 ACCESSMAX-ACCESS                                read-only
52 STATUS mandatoryCurrent
53 DESCRIPTION
54   "3GPP2 PDSN IP Address.
55
56   Current main using PDSN IP address
57
58   (From mibCtl ElementType 2113 3GPP2PDSNIPAddress)
59   Description for mibCtl Type 420 IPAddressTextType :
60   Internet Protocol Address (Text).
61
62   This text must currently be in the dotted abc.def.ghi.jkl format.
63   In the future, hostnames might be allowed.
64   Type derived from mibCtl Type 15 TextType :
65   ASCII or compatible text.
66   Type derived from mibCtl Type 12 OctetType :
67   8 bits of raw opaque data.
68   Derived from basic 8 bit word type.

```



```

1      "
2      ::= { _625k-MCA10StatusTableEntry 2 }
3
4
5
6      _625k-MCInterfaceRF                                OBJECT IDENTIFIER
7      -- DESCRIPTION      "Radio Frequency Interfaces"
8      ::= { _625k-MCSysInterfaces 2 }
9
10
11
12     _625k-MCCarrierTable                                OBJECT-TYPE
13     SYNTAX SEQUENCE OF _625k-MCCarrierTableEntry
14     ACCESSMAX-ACCESS                                not-accessible
15     STATUS mandatoryCurrent
16     DESCRIPTION      "Carrier Frequency Table"
17     ::= { _625k-MCInterfaceRF 1 }
18
19
20
21     _625k-MCCarrierTableEntry                          OBJECT-TYPE
22     SYNTAX          _625k-MCCarrierTableEntry
23     ACCESSMAX-ACCESS                                not-accessible
24     STATUS mandatoryCurrent
25     DESCRIPTION      ""
26     INDEX { _625k-MCCarrierTableIndex }
27     ::= { _625k-MCCarrierTable 1 }
28
29     _625k-MCCarrierTableEntry ::= SEQUENCE {
30         _625k-MCCarrierTableIndex      INTEGER, -- BaseStationCarrierType
31         _625k-MCCarrierUsage           INTEGER -- CarrierUsageType
32     }
33
34
35
36     _625k-MCCarrierTableIndex                          OBJECT-TYPE
37     SYNTAX          INTEGER -- BaseStationCarrierType
38     ACCESSMAX-ACCESS                                read-only
39     STATUS mandatoryCurrent
40     DESCRIPTION      "
41         Description for mibCtl Type 219 BaseStationCarrierType :
42         Base station carrier number.
43
44         Base station carriers are a contiguous set of carriers
45         that are used by the Base Station;
46         they are numbered from 0 to a current maximum of 32-1.
47         [Limits: 0 15 ]
48         Type derived from mibCtl Type 14 Unsigned32Type :
49         32 bit unsigned integer.
50         Type derived from mibCtl Type 11 Word32Type :
51         32 bits of raw opaque data.
52         Derived from basic 32 bit word type.
53     "
54     ::= { _625k-MCCarrierTableEntry 1 }
55
56
57
58     _625k-MCCarrierUsage                                OBJECT-TYPE
59     SYNTAX          INTEGER -- CarrierUsageType
60     ACCESSMAX-ACCESS                                read-only
61     STATUS mandatoryCurrent
62     DESCRIPTION      "Current assigned usage per base station carrier.
63
64
65
66     (From mibCtl ElementType 54 CarrierUsage)
67     Description for mibCtl Type 220 CarrierUsageType :
68

```

```

1         The assigned use of a radio carrier.
2
3         A radio carrier is a frequency band.
4         The assigned use of a carrier can be Reserved, Control
5         or Traffic.
6         Base station transmits control information on one of the time
7 slots
8         of given Control carrier.
9         Base station does not transmit anything on Reserved carriers.
10        Description for mibCtl CarrierUsageType 0 NotUse :
11        Not Use for this carrier.
12        Description for mibCtl CarrierUsageType 1 TCH :
13        All timeslots in this carrier are for traffic only.
14        Description for mibCtl CarrierUsageType 2 TCHBCH :
15        One timeslot in this carrier is for BCH, others for TCH.
16        "
17        ::= { _625k-MCCarrierTableEntry 2 }
18
19
20
21        _625k-MCBSCC                                OBJECT-TYPE
22        SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
23        ACCESSMAX-ACCESS      read-write
24        STATUS          mandatoryCurrent
25        DESCRIPTION
26        "Base Station Color Code.
27
28        (From mibCtl ElementType 61 BSCC)
29        Description for mibCtl Type 14 Unsigned32Type :
30        32 bit unsigned integer.
31        Type derived from mibCtl Type 11 Word32Type :
32        32 bits of raw opaque data.
33        Derived from basic 32 bit word type.
34        "
35        ::= { _625k-MCInterfaceRF 2 }
36
37
38
39        _625k-MCBSLowestCarrier                      OBJECT-TYPE
40        SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
41        ACCESSMAX-ACCESS      read-write
42        STATUS          mandatoryCurrent
43        DESCRIPTION
44        "The lowest carrier of the base station operating band.
45
46        This is an extended carrier number
47        that identifies the lowest carrier of the bandwidth
48        to which the base station is tuned.
49        This value cannot be changed while the Base Station state
50        is Operating.
51
52        (From mibCtl ElementType 52 BSLowestCarrier)
53        Description for mibCtl Type 15 TextType :
54        ASCII or compatible text.
55        Type derived from mibCtl Type 12 OctetType :
56        8 bits of raw opaque data.
57        Derived from basic 8 bit word type.
58        "
59        ::= { _625k-MCInterfaceRF 3 }
60
61
62
63        _625k-MCBCHModuleAddress                    OBJECT-TYPE
64        SYNTAX          INTEGER -- ModuleAddressType
65        ACCESSMAX-ACCESS      read-only
66        STATUS          mandatoryCurrent
67        DESCRIPTION
68        "Which module is handling the broadcast channel.

```

```

1
2     (From mibCtl ElementType 57 BCHModuleAddress)
3     Description for mibCtl Type 202 ModuleAddressType :
4         Base station bus slot address.
5
6         Most components of the Base Station for which data can
7         be obtained are identified by a ModuleAddressType address
8         and possibly a subsidiary address.
9         [Limits: 0 7 ]
10        Type derived from mibCtl Type 14 Unsigned32Type :
11            32 bit unsigned integer.
12        Type derived from mibCtl Type 11 Word32Type :
13            32 bits of raw opaque data.
14        Derived from basic 32 bit word type.
15        "
16    ::= { _625k-MCInterfaceRF 4 }
17
18
19
20    _625k-MCBCHCarrierNumber                OBJECT-TYPE
21    SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
22    ACCESSMAX-ACCESS read-write
23    STATUS mandatoryCurrent
24    DESCRIPTION
25        "Number Of BCH Carrier.
26
27        (From mibCtl ElementType 58 BCHCarrierNumber)
28        Description for mibCtl Type 15 TextType :
29            ASCII or compatible text.
30        Type derived from mibCtl Type 12 OctetType :
31            8 bits of raw opaque data.
32        Derived from basic 8 bit word type.
33        "
34    ::= { _625k-MCInterfaceRF 5 }
35
36
37
38    _625k-MCRACHCarrierMask                OBJECT-TYPE
39    SYNTAX INTEGERUnsigned32 -- Unsigned32Type
40    ACCESSMAX-ACCESS read-write
41    STATUS mandatoryCurrent
42    DESCRIPTION
43        "RACH carrier mask.
44
45
46        (From mibCtl ElementType 73 RACHCarrierMask)
47        Description for mibCtl Type 14 Unsigned32Type :
48            32 bit unsigned integer.
49        Type derived from mibCtl Type 11 Word32Type :
50            32 bits of raw opaque data.
51        Derived from basic 32 bit word type.
52        "
53    ::= { _625k-MCInterfaceRF 6 }
54
55
56
57
58    _625k-MCRACHSlotMask                OBJECT-TYPE
59    SYNTAX INTEGERUnsigned32 -- Unsigned32Type
60    ACCESSMAX-ACCESS read-write
61    STATUS mandatoryCurrent
62    DESCRIPTION
63        "RACH slot mask.
64
65
66        (From mibCtl ElementType 72 RACHSlotMask)
67        Description for mibCtl Type 14 Unsigned32Type :
68

```

```

1         32 bit unsigned integer.
2         Type derived from mibCtl Type 11 Word32Type :
3         32 bits of raw opaque data.
4         Derived from basic 32 bit word type.
5         "
6         ::= { _625k-MCInterfaceRF 7 }
7
8
9
10        _625k-MCCalibrationInterval          OBJECT-TYPE
11        | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
12        | ACCESSMAX-ACCESS          read-write
13        | STATUS          mandatoryCurrent
14        | DESCRIPTION
15        |     "Calibration interval time.
16
17
18        (From mibCtl ElementType 75 CalibrationInterval)
19        Description for mibCtl Type 14 Unsigned32Type :
20        32 bit unsigned integer.
21        Type derived from mibCtl Type 11 Word32Type :
22        32 bits of raw opaque data.
23        Derived from basic 32 bit word type.
24        "
25        ::= { _625k-MCInterfaceRF 8 }
26
27
28
29
30        _625k-MCSpatialParameter            OBJECT-TYPE
31        | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
32        | ACCESSMAX-ACCESS          read-write
33        | STATUS          mandatoryCurrent
34        | DESCRIPTION
35        |     "Spatial parameter.
36
37
38        (From mibCtl ElementType 78 SpatialParameter)
39        Description for mibCtl Type 14 Unsigned32Type :
40        32 bit unsigned integer.
41        Type derived from mibCtl Type 11 Word32Type :
42        32 bits of raw opaque data.
43        Derived from basic 32 bit word type.
44        "
45        ::= { _625k-MCInterfaceRF 9 }
46
47
48
49
50        _625k-MCCostCalcParameter           OBJECT-TYPE
51        | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
52        | ACCESSMAX-ACCESS          read-write
53        | STATUS          mandatoryCurrent
54        | DESCRIPTION
55        |     "Cost calculation parameter.
56
57
58        (From mibCtl ElementType 79 CostCalcParameter)
59        Description for mibCtl Type 14 Unsigned32Type :
60        32 bit unsigned integer.
61        Type derived from mibCtl Type 11 Word32Type :
62        32 bits of raw opaque data.
63        Derived from basic 32 bit word type.
64        "
65        ::= { _625k-MCInterfaceRF 10 }
66
67
68

```

```

1
2  _625k-MCBSRegistrationCapacity          OBJECT-TYPE
3  | SYNTAX                               INTEGERUnsigned32 -- Unsigned32Type
4  | ACCESSMAX-ACCESS                   read-write
5  | STATUS                               mandatoryCurrent
6  | DESCRIPTION
7  |     "Capacity of UT registration on BS.
8  |
9  |
10 |
11 |     (From mibCtl ElementType 76 BSRegistrationCapacity)
12 |     Description for mibCtl Type 14 Unsigned32Type :
13 |         32 bit unsigned integer.
14 |     Type derived from mibCtl Type 11 Word32Type :
15 |         32 bits of raw opaque data.
16 |     Derived from basic 32 bit word type.
17 |     "
18 | ::= { _625k-MCInterfaceRF 11 }
19 |
20 |
21 |
22 | _625k-MCBSRegistrationTimer            OBJECT-TYPE
23 | | SYNTAX                               INTEGERUnsigned32 -- Unsigned32Type
24 | | ACCESSMAX-ACCESS                   read-write
25 | | STATUS                               mandatoryCurrent
26 | | DESCRIPTION
27 | |     "Timer of keeping UT registration on BS.
28 | |
29 | |
30 | |
31 | |     (From mibCtl ElementType 77 BSRegistrationTimer)
32 | |     Description for mibCtl Type 14 Unsigned32Type :
33 | |         32 bit unsigned integer.
34 | |     Type derived from mibCtl Type 11 Word32Type :
35 | |         32 bits of raw opaque data.
36 | |     Derived from basic 32 bit word type.
37 | |     "
38 | | ::= { _625k-MCInterfaceRF 12 }
39 | |
40 | |
41 | |
42 | | _625k-MCPCHFrequencyHopping          OBJECT-TYPE
43 | | SYNTAX                               INTEGER -- BooleanType
44 | | | ACCESSMAX-ACCESS                   read-write
45 | | | STATUS                               mandatoryCurrent
46 | | | DESCRIPTION
47 | | |     "Propriety of frequency hopping (PCH).
48 | | |
49 | | |
50 | | |
51 | | |     (From mibCtl ElementType 70 PCHFrequencyHopping)
52 | | |     Description for mibCtl Type 16 BooleanType :
53 | | |         Truth value, 0=FALSE, 1=TRUE.
54 | | |
55 | | |         This is a subset of TriStateType; no UNDEFINED value is provided.
56 | | |         [Limits: 0 1 ]
57 | | |     Description for mibCtl BooleanType 0 FALSE :
58 | | |         False.
59 | | |     Description for mibCtl BooleanType 1 TRUE :
60 | | |         True.
61 | | |     "
62 | | | ::= { _625k-MCInterfaceRF 13 }
63 | | |
64 | | |
65 | | |
66 | | | _625k-MCTCHFrequencyHopping        OBJECT-TYPE
67 | | | SYNTAX                               INTEGER -- BooleanType
68 | | | | ACCESSMAX-ACCESS                   read-write

```

```

1 | STATUS          mandatoryCurrent
2 | DESCRIPTION
3 |     "Propriety of frequency hopping (TCH).
4 |
5 |
6 |
7 |     (From mibCtl ElementType 71 TCHFrequencyHopping)
8 |     Description for mibCtl Type 16 BooleanType :
9 |         Truth value, 0=FALSE, 1=TRUE.
10 |
11 |         This is a subset of TriStateType; no UNDEFINED value is provided.
12 |         [Limits: 0 1 ]
13 |     Description for mibCtl BooleanType 0 FALSE :
14 |         False.
15 |     Description for mibCtl BooleanType 1 TRUE :
16 |         True.
17 |     "
18 | ::= { _625k-MCInterfaceRF 14 }
19 |
20 |
21 |
22 |
23 | _625k-MCRFStatusTable          OBJECT-TYPE
24 | SYNTAX SEQUENCE OF _625k-MCRFStatusTableEntry
25 | ACCESSMAX-ACCESS          not-accessible
26 | STATUS          mandatoryCurrent
27 | DESCRIPTION    "RF Status Table"
28 | ::= { _625k-MCInterfaceRF 15 }
29 |
30 |
31 |
32 | _625k-MCRFStatusTableEntry    OBJECT-TYPE
33 | SYNTAX          _625k-MCRFStatusTableEntry
34 | ACCESSMAX-ACCESS          not-accessible
35 | STATUS          mandatoryCurrent
36 | DESCRIPTION    ""
37 | INDEX { _625k-MCRFStatusTableIndex }
38 | ::= { _625k-MCRFStatusTable 1 }
39 |
40 | _625k-MCRFStatusTableEntry ::= SEQUENCE {
41 |     _625k-MCRFStatusTableIndex          INTEGER, -- MoNerdAddressType
42 |     _625k-MCBSAirBitRateUpLink         INTEGERUnsigned32, -- Unsigned32Type
43 |     _625k-MCBSAirBitRateDownLink       INTEGERUnsigned32, -- Unsigned32Type
44 |     _625k-MCBSActiveStream              INTEGERUnsigned32, -- Unsigned32Type
45 |     _625k-MCBSActiveRegistration        INTEGERUnsigned32 -- Unsigned32Type
46 | }
47 |
48 |
49 |
50 | _625k-MCRFStatusTableIndex    OBJECT-TYPE
51 | SYNTAX          INTEGER -- MoNerdAddressType
52 | ACCESSMAX-ACCESS          read-only
53 | STATUS          mandatoryCurrent
54 | DESCRIPTION    "
55 |     Description for mibCtl Type 204 MoNerdAddressType :
56 |         Base station network component address.
57 |
58 |         A network address is a subset of Base Station component addresses,
59 |         restricted to network components only.
60 |         Network components interface with a telephony switch or similar.
61 |         [Limits: 0 1 ]
62 |     Type derived from mibCtl Type 14 Unsigned32Type :
63 |         32 bit unsigned integer.
64 |     Type derived from mibCtl Type 11 Word32Type :
65 |         32 bits of raw opaque data.
66 |     Derived from basic 32 bit word type.
67 |     "
68 | ::= { _625k-MCRFStatusTableEntry 1 }

```

```

1
2
3
4  _625k-MCBSAirBitRateUpLink          OBJECT-TYPE
5  | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
6  | ACCESSMAX-ACCESS      read-only
7  | STATUS          mandatoryCurrent
8  | DESCRIPTION
9  |     "Radio bit rate of up link per Modem control board.
10 |
11 |
12 |     (From mibCtl ElementType 4022 BSAirBitRateUpLink)
13 |     Description for mibCtl Type 14 Unsigned32Type :
14 |         32 bit unsigned integer.
15 |     Type derived from mibCtl Type 11 Word32Type :
16 |         32 bits of raw opaque data.
17 |     Derived from basic 32 bit word type.
18 |     "
19 |     ::= { _625k-MCRFStatusTableEntry 2 }
20 |
21 |
22 |
23 |
24 | _625k-MCBSAirBitRateDownLink        OBJECT-TYPE
25 | | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
26 | | ACCESSMAX-ACCESS      read-only
27 | | STATUS          mandatoryCurrent
28 | | DESCRIPTION
29 | |     "Radio bit rate of down link per Modem control board.
30 | |
31 | |
32 | |     (From mibCtl ElementType 4023 BSAirBitRateDownLink)
33 | |     Description for mibCtl Type 14 Unsigned32Type :
34 | |         32 bit unsigned integer.
35 | |     Type derived from mibCtl Type 11 Word32Type :
36 | |         32 bits of raw opaque data.
37 | |     Derived from basic 32 bit word type.
38 | |     "
39 | |     ::= { _625k-MCRFStatusTableEntry 3 }
40 | |
41 | |
42 | |
43 | |
44 | _625k-MCBSActiveStream              OBJECT-TYPE
45 | | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
46 | | ACCESSMAX-ACCESS      read-only
47 | | STATUS          mandatoryCurrent
48 | | DESCRIPTION
49 | |     "Number of streams currently connected in a base station.
50 | |
51 | |     Number of active streams.
52 | |
53 | |     (From mibCtl ElementType 4020 BSActiveStream)
54 | |     Description for mibCtl Type 14 Unsigned32Type :
55 | |         32 bit unsigned integer.
56 | |     Type derived from mibCtl Type 11 Word32Type :
57 | |         32 bits of raw opaque data.
58 | |     Derived from basic 32 bit word type.
59 | |     "
60 | |     ::= { _625k-MCRFStatusTableEntry 4 }
61 | |
62 | |
63 | |
64 | _625k-MCBSActiveRegistration          OBJECT-TYPE
65 | | SYNTAX          INTEGERUnsigned32 -- Unsigned32Type
66 | | ACCESSMAX-ACCESS      read-only
67 | | STATUS          mandatoryCurrent
68 | | DESCRIPTION

```

```

1      "Number of registrations currently existed in a base station.
2
3      The call capacity is determined by the available resources in a
4      base station.
5
6      (From mibCtl ElementType 4021 BSActiveRegistration)
7      Description for mibCtl Type 14 Unsigned32Type :
8          32 bit unsigned integer.
9      Type derived from mibCtl Type 11 Word32Type :
10         32 bits of raw opaque data.
11     Derived from basic 32 bit word type.
12     "
13     ::= { _625k-MCRFStatusTableEntry 5 }
14
15     _625k-MCSysScalars          OBJECT IDENTIFIER
16     -- DESCRIPTION             "System Scalars"
17     ::= { _625k-MCSystem 4 }
18
19
20
21     _625k-MCBaseStationID      OBJECT-TYPE
22     SYNTAX                     OCTET STRING (SIZE(0..18)) -- TextType X 18
23     ACCESSMAX-ACCESS         read-write
24     STATUS                     mandatoryCurrent
25     DESCRIPTION
26         "Base Station Identification Code.
27
28         This text string must represent in hexadecimal a 42 bit number
29         to be used as the Base Station Identification Code (BSID).
30         The BSID is used by the base station to identify itself to
31         subscriber units.
32         The BSID of a base station must at a minimum
33         differ from that of any other base station
34         where both would be within radio reception distance
35         of any subscriber unit.
36
37         This cannot be changed while the Base Station state is Operating.
38
39         (From mibCtl ElementType 60 BaseStationID)
40         Description for mibCtl Type 15 TextType :
41             ASCII or compatible text.
42         Type derived from mibCtl Type 12 OctetType :
43             8 bits of raw opaque data.
44         Derived from basic 8 bit word type.
45         "
46     ::= { _625k-MCSysScalars 1 }
47
48
49
50     _625k-MCBaseStationTypeID  OBJECT-TYPE
51     SYNTAX                     OCTET STRING (SIZE(0..20)) -- TextType X 20
52     ACCESSMAX-ACCESS         read-write
53     STATUS                     mandatoryCurrent
54     DESCRIPTION
55         "Type ID of base station.
56
57
58
59         (From mibCtl ElementType 66 BaseStationTypeID)
60         Description for mibCtl Type 15 TextType :
61             ASCII or compatible text.
62         Type derived from mibCtl Type 12 OctetType :
63             8 bits of raw opaque data.
64         Derived from basic 8 bit word type.
65         "
66     ::= { _625k-MCSysScalars 2 }
67
68

```



```

1
2  _625k-MCBaseStationGroupID          OBJECT-TYPE
3  SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
4  ACCESSMAX-ACCESS          read-write
5  STATUS          mandatoryCurrent
6  DESCRIPTION
7      "Group ID of base station.
8
9
10
11     (From mibCtl ElementType 67 BaseStationGroupID)
12     Description for mibCtl Type 15 TextType :
13         ASCII or compatible text.
14     Type derived from mibCtl Type 12 OctetType :
15         8 bits of raw opaque data.
16     Derived from basic 8 bit word type.
17     "
18     ::= { _625k-MCSysScalars 3 }
19
20
21
22  _625k-MCBaseStationSubGroupID        OBJECT-TYPE
23  SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
24  ACCESSMAX-ACCESS          read-write
25  STATUS          mandatoryCurrent
26  DESCRIPTION
27      "Sub group ID of base station.
28
29
30
31     (From mibCtl ElementType 68 BaseStationSubGroupID)
32     Description for mibCtl Type 15 TextType :
33         ASCII or compatible text.
34     Type derived from mibCtl Type 12 OctetType :
35         8 bits of raw opaque data.
36     Derived from basic 8 bit word type.
37     "
38     ::= { _625k-MCSysScalars 4 }
39
40
41
42  _625k-MCDesiredStateOfBaseStation     OBJECT-TYPE
43  SYNTAX          INTEGER -- ComponentStateType
44  ACCESSMAX-ACCESS          read-write
45  STATUS          mandatoryCurrent
46  DESCRIPTION
47      "Desired state of base station as a whole.
48
49      This indicates the Base Station state desired by the operator.
50      These desired states are currently supported:
51
52      Operating - for normal operation.
53
54      Ready - to avoid taking any new calls.
55      Existing calls will not be terminated except normally or by
56      command from the operator.
57      While existing calls remain, the base station state will
58      remain as Operating.
59
60      This information is permanently stored on the base station.
61
62     (From mibCtl ElementType 42 DesiredStateOfBaseStation)
63     Description for mibCtl Type 71 ComponentStateType :
64         Component operational state.
65
66     A component begins in the Unknown state.
67     If not detected, it enters and remains in the NotPresent state.
68     If detected, it enters the Uninitialized state, from where it

```

```

1         may go to the Testing and Initializing states and then to the
2         Standby or Operating state depending upon permissions.
3         Due to loss of permissions or resources, it may revert from
4         the Operating state to the Standby state.
5         Due to failure or loss of permission, it may revert to the
6         Uninitialized state, perhaps by way of the ShuttingDown state
7         depending on the device.
8         From the Uninitialized state it may return to more advanced
9         states depending upon permissions.
10        In case of a waiting period before (again) initializing,
11        the component is considered to be Initializing.
12
13        Permissions include administrative permissions (from the
14        operator); excessive failure restrictions; etc.
15        Description for mibCtl ComponentStateType 0 Unknown :
16        Component state not known.
17        Description for mibCtl ComponentStateType 1 NotPresent :
18        Component is not present.
19        Description for mibCtl ComponentStateType 2 PowerOff :
20        Component is present but powered off.
21        Description for mibCtl ComponentStateType 3 Uninitialized :
22        Component is present but not in use.
23
24        The power on/off state of the component is not specified in
25        this case.
26        Description for mibCtl ComponentStateType 4 Testing :
27        Component is being tested.
28        Description for mibCtl ComponentStateType 5 Initializing :
29        Component is being initialized.
30        Description for mibCtl ComponentStateType 6 Ready :
31        Component is ready but not operating.
32        Description for mibCtl ComponentStateType 7 Operating :
33        Component is operating for normal use without restriction.
34
35        The component is either in actual use or may be used at any time,
36        without restriction.
37        Description for mibCtl ComponentStateType 8 Abandoned :
38        Component state is not the desired state due to excessive errors.
39
40        The component state is not that desired, and the Base Station
41        software has abandoned attempts to place the component in
42        the desired state.
43        The actual state of the component is undefined.
44        The Base Station software will resume attempting to place the
45        component in the desired state if the appropriate Reinitialize
46        action element is written with the correct value.
47        Also, the software may resume attempts under other conditions,
48        not all of which may be documented.
49        Description for mibCtl ComponentStateType 9 InitialSetUp :
50        Component is initial set up..
51
52        Initial set up state.
53        Description for mibCtl ComponentStateType 10 Degrading :
54        Component is degrading..
55
56        Degrading state.
57        Description for mibCtl ComponentStateType 11 Restriction :
58        Component is restriction..
59
60        Restriction state.
61        "
62        ::= { _625k-MCSysScalars 5 }
63
64
65
66        _625k-MCTypeOfReboot                OBJECT-TYPE
67        SYNTAX INTEGER -- RebootType
68        ACCESS MAX-ACCESS read-write

```

```

1 | STATUS          mandatoryCurrent
2 | DESCRIPTION
3 |     "Type of reboot for base station.
4 |
5 |
6 |
7 |     (From mibCtl ElementType 63 TypeOfReboot)
8 |     Description for mibCtl Type 250 RebootType :
9 |         Reboot Type.
10 |     Description for mibCtl RebootType 0 Force :
11 |         Force mode.
12 |     Description for mibCtl RebootType 1 Graceful :
13 |         Graceful mode.
14 |     "
15 | ::= { _625k-MCSysScalars 6 }
16 |
17 |
18 |
19 | _625k-MCBaseStationRebootTime          OBJECT-TYPE
20 | SYNTAX          Gauge -- AbsoluteTimeType
21 | ACCESSMAX-ACCESS          read-write
22 | STATUS          mandatoryCurrent
23 | DESCRIPTION
24 |     "Time of base station reboot.
25 |
26 |     This is the base station reboot time (GPS time).
27 |
28 |     (From mibCtl ElementType 62 BaseStationRebootTime)
29 |     Description for mibCtl Type 801 AbsoluteTimeType :
30 |         Absolute time in GPS seconds.
31 |
32 |         GPS (Global Positioning System) time in seconds since Jan. 6,
33 | 1980.
34 |
35 |         Note that this differs from UTC (in addition to a possible
36 |         offset due to starting time) due to leap seconds; see
37 |         the GpsLeapSecond element.
38 |     Type derived from mibCtl Type 18 Gauge32Type :
39 |         32 bits of Gauge data.
40 |     Derived from basic 32 bit word type.
41 |     "
42 | ::= { _625k-MCSysScalars 7 }
43 |
44 |
45 | _625k-MCTypeOfBSDiagnosis              OBJECT-TYPE
46 | SYNTAX          INTEGER -- DiagnosisType
47 | ACCESSMAX-ACCESS          read-write
48 | STATUS          mandatoryCurrent
49 | DESCRIPTION
50 |     "Type of diagnosis for base station.
51 |
52 |
53 |
54 |     (From mibCtl ElementType 64 TypeOfBSDiagnosis)
55 |     Description for mibCtl Type 251 DiagnosisType :
56 |         Diagnosis Type.
57 |     Type derived from mibCtl Type 14 Unsigned32Type :
58 |         32 bit unsigned integer.
59 |     Type derived from mibCtl Type 11 Word32Type :
60 |         32 bits of raw opaque data.
61 |     Derived from basic 32 bit word type.
62 |     "
63 | ::= { _625k-MCSysScalars 8 }
64 |
65 |
66 |
67 | _625k-MCBSdiagnosisStatus              OBJECT-TYPE
68 | SYNTAX          INTEGER -- DiagnosisStatusType

```

```

1 | ACCESSMAX-ACCESS read-only
2 | STATUS mandatoryCurrent
3 | DESCRIPTION
4 | "Diagnosis status for base station.
5 |
6 |
7 |
8 | (From mibCtl ElementType 370 BSDiagnosisStatus)
9 | Description for mibCtl Type 252 DiagnosisStatusType :
10 | Diagnosis status Type.
11 | Type derived from mibCtl Type 14 Unsigned32Type :
12 | 32 bit unsigned integer.
13 | Type derived from mibCtl Type 11 Word32Type :
14 | 32 bits of raw opaque data.
15 | Derived from basic 32 bit word type.
16 | "
17 | ::= { _625k-MCSysScalars 9 }
18 |
19 |
20 |
21 | _625k-MCBSDiagnosisFailReason OBJECT-TYPE
22 | SYNTAX INTEGER -- DiagFailReasonType
23 | ACCESSMAX-ACCESS read-only
24 | STATUS mandatoryCurrent
25 | DESCRIPTION
26 | "Diagnosis fail reason for base station.
27 |
28 |
29 |
30 | (From mibCtl ElementType 371 BSDiagnosisFailReason)
31 | Description for mibCtl Type 253 DiagFailReasonType :
32 | Diagnosis fail reason Type.
33 | Description for mibCtl DiagFailReasonType 1 PDSNPing :
34 | Diagnosis fail reason is PDSN Ping.
35 | Description for mibCtl DiagFailReasonType 2 Calibration :
36 | Diagnosis fail reason is Calibration.
37 | Description for mibCtl DiagFailReasonType 3 AntPath :
38 | Diagnosis fail reason is TRx Antenna Path.
39 | Description for mibCtl DiagFailReasonType 4 LOAlive :
40 | Diagnosis fail reason is Local Oscilator DSP Alive.
41 | Description for mibCtl DiagFailReasonType 5 GCLoopBack :
42 | Diagnosis fail reason is GCLoopBack.
43 | Description for mibCtl DiagFailReasonType 6 SlaveNM :
44 | Diagnosis fail reason is Slave Modem control board.
45 | Description for mibCtl DiagFailReasonType 7 GPSAnt :
46 | Diagnosis fail reason is GPS Antenna.
47 | Description for mibCtl DiagFailReasonType 8 SlotDSP :
48 | Diagnosis fail reason is Modem control board DSP Alive.
49 | Description for mibCtl DiagFailReasonType 9 ATMAlive :
50 | Diagnosis fail reason is ATM Alive.
51 | Description for mibCtl DiagFailReasonType 96 UndefinedName :
52 | Diagnosis fail reason is Undefined Diag Name.
53 | Description for mibCtl DiagFailReasonType 97 TimeOut :
54 | Diagnosis fail reason is Time Out.
55 | Description for mibCtl DiagFailReasonType 98 InvalidStateExec :
56 | Diagnosis fail reason is Invalid State Execute.
57 | Description for mibCtl DiagFailReasonType 99 ExecFail :
58 | Diagnosis fail reason is Execute Fail.
59 | "
60 | ::= { _625k-MCSysScalars 10 }
61 |
62 |
63 |
64 | _625k-MCDiskDbUpdateSequence OBJECT-TYPE
65 | SYNTAX Gauge -- Gauge32Type
66 | ACCESSMAX-ACCESS read-only
67 | STATUS mandatoryCurrent
68 | DESCRIPTION

```

```

1      "Base station Flach update sequence number.
2
3      This number is incremented on disk every time any other
4      database element is actually changed on flash.
5      It is not incremented on redundant sets.
6      This number may also be set to a desired value.
7
8      (From mibCtl ElementType 6 DiskDbUpdateSequence)
9      Description for mibCtl Type 18 Gauge32Type :
10     32 bits of Gauge data.
11     Derived from basic 32 bit word type.
12     "
13     ::= { _625k-MCScalars 11 }
14
15
16
17     _625k-MCStateOfBaseStation          OBJECT-TYPE
18     SYNTAX          INTEGER -- ComponentStateType
19     ACCESSMAX-ACCESS          read-only
20     STATUS          mandatoryCurrent
21     DESCRIPTION
22     "State of base station as a whole.
23
24     This will not have values of Unknown or PowerOff since the
25     base station would be unable to report such values.
26
27     When sufficiently initialized, the state will be Operating if
28     accepting new calls (according to the desired state of the
29     base station) or continuing ongoing calls;
30     or Ready if the desired state is Ready and there are no
31     ongoing calls.
32
33     (From mibCtl ElementType 41 StateOfBaseStation)
34     Description for mibCtl Type 71 ComponentStateType :
35     Component operational state.
36
37     A component begins in the Unknown state.
38     If not detected, it enters and remains in the NotPresent state.
39     If detected, it enters the Uninitialized state, from where it
40     may go to the Testing and Initializing states and then to the
41     Standby or Operating state depending upon permissions.
42     Due to loss of permissions or resources, it may revert from
43     the Operating state to the Standby state.
44     Due to failure or loss of permission, it may revert to the
45     Uninitialized state, perhaps by way of the ShuttingDown state
46     depending on the device.
47     From the Uninitialized state it may return to more advanced
48     states depending upon permissions.
49     In case of a waiting period before (again) initializing,
50     the component is considered to be Initializing.
51
52     Permissions include administrative permissions (from the
53     operator); excessive failure restrictions; etc.
54     Description for mibCtl ComponentStateType 0 Unknown :
55     Component state not known.
56     Description for mibCtl ComponentStateType 1 NotPresent :
57     Component is not present.
58     Description for mibCtl ComponentStateType 2 PowerOff :
59     Component is present but powered off.
60     Description for mibCtl ComponentStateType 3 Uninitialized :
61     Component is present but not in use.
62
63     The power on/off state of the component is not specified in
64     this case.
65     Description for mibCtl ComponentStateType 4 Testing :
66     Component is being tested.
67     Description for mibCtl ComponentStateType 5 Initializing :
68     Component is being initialized.

```

```

1      Description for mibCtl ComponentStateType 6 Ready :
2          Component is ready but not operating.
3      Description for mibCtl ComponentStateType 7 Operating :
4          Component is operating for normal use without restriction.
5
6          The component is either in actual use or may be used at any time,
7          without restriction.
8      Description for mibCtl ComponentStateType 8 Abandoned :
9          Component state is not the desired state due to excessive errors.
10
11          The component state is not that desired, and the Base Station
12          software has abandoned attempts to place the component in
13          the desired state.
14          The actual state of the component is undefined.
15          The Base Station software will resume attempting to place the
16          component in the desired state if the appropriate Reinitialize
17          action element is written with the correct value.
18          Also, the software may resume attempts under other conditions,
19          not all of which may be documented.
20      Description for mibCtl ComponentStateType 9 InitialSetUp :
21          Component is initial set up..
22
23          Initial set up state.
24      Description for mibCtl ComponentStateType 10 Degrading :
25          Component is degrading..
26
27          Degrading state.
28      Description for mibCtl ComponentStateType 11 Restriction :
29          Component is restriction..
30
31          Restriction state.
32      "
33      ::= { _625k-MCSysScalars 12 }
34
35
36
37      _625k-MCBSTotalIndication          OBJECT-TYPE
38      SYNTAX          INTEGER -- IndicationType
39      ACCESSMAX-ACCESS read-only
40      STATUS          mandatoryCurrent
41      DESCRIPTION
42          "Status of BS total indicator.
43
44          In the current implementation,
45          this indication is set to the value On by BS.
46
47          (From mibCtl ElementType 525 BSTotalIndication)
48      Description for mibCtl Type 230 IndicationType :
49          Hardware indication status (LEDs).
50
51
52      Description for mibCtl IndicationType 0 Off :
53          Off.
54      Description for mibCtl IndicationType 1 Amber :
55          Amber.
56      Description for mibCtl IndicationType 2 Red :
57          Red.
58      Description for mibCtl IndicationType 3 Green :
59          Green.
60      Description for mibCtl IndicationType 4 NotPresent :
61          Not present.
62      "
63      ::= { _625k-MCSysScalars 13 }
64
65
66
67      _625k-MCMASTERAddress          OBJECT-TYPE
68      SYNTAX          INTEGER -- ModuleAddressType

```

```

1 | ACCESSMAX-ACCESS read-only
2 | STATUS mandatoryCurrent
3 | DESCRIPTION
4 | "Bus slot address of master Modem control board.
5 |
6 | This indicates which Modem control board is master
7 | of the base station.
8 |
9 | (From mibCtl ElementType 32 MasterAddress)
10 | Description for mibCtl Type 202 ModuleAddressType :
11 | Base station bus slot address.
12 |
13 | Most components of the Base Station for which data can
14 | be obtained are identified by a ModuleAddressType address
15 | and possibly a subsidiary address.
16 | [Limits: 0 7 ]
17 | Type derived from mibCtl Type 14 Unsigned32Type :
18 | 32 bit unsigned integer.
19 | Type derived from mibCtl Type 11 Word32Type :
20 | 32 bits of raw opaque data.
21 | Derived from basic 32 bit word type.
22 | "
23 | ::= { _625k-MCSysScalars 14 }
24 |
25 |
26 |
27 | _625k-MCBSManufactureID OBJECT-TYPE
28 | SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
29 | ACCESSMAX-ACCESS read-only
30 | STATUS mandatoryCurrent
31 | DESCRIPTION
32 | "Base Station manufacture identification number.
33 |
34 | The manufacture identification assigned by Vendor
35 |
36 | (From mibCtl ElementType 201 BSManufactureID)
37 | Description for mibCtl Type 15 TextType :
38 | ASCII or compatible text.
39 | Type derived from mibCtl Type 12 OctetType :
40 | 8 bits of raw opaque data.
41 | Derived from basic 8 bit word type.
42 | "
43 | ::= { _625k-MCSysScalars 15 }
44 |
45 |
46 |
47 | _625k-MCBSSerialNumber OBJECT-TYPE
48 | SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
49 | ACCESSMAX-ACCESS read-only
50 | STATUS mandatoryCurrent
51 | DESCRIPTION
52 | "Base Station serial number.
53 |
54 | This is the character serial number of the base station.
55 | This serial number will be unique among all base stations
56 | of this type regardless of manufacturer.
57 |
58 | (From mibCtl ElementType 203 BSSerialNumber)
59 | Description for mibCtl Type 15 TextType :
60 | ASCII or compatible text.
61 | Type derived from mibCtl Type 12 OctetType :
62 | 8 bits of raw opaque data.
63 | Derived from basic 8 bit word type.
64 | "
65 | ::= { _625k-MCSysScalars 16 }
66 |
67 |
68 |

```

```

1  _625k-MCDiagnosisBaseStation          OBJECT-TYPE
2  SYNTAX                                INTEGER -- BooleanType
3  | ACCESSMAX-ACCESS                    read-write -- REALLY: write-only
4  | STATUS                               mandatoryCurrent
5  DESCRIPTION
6  | "Diagnosis base station.
7  |
8  | This is a write-only element; only a value of TRUE is valid.
9  |
10 | (From mibCtl ElementType 47 DiagnosisBaseStation)
11 | Description for mibCtl Type 16 BooleanType :
12 | Truth value, 0=FALSE, 1=TRUE.
13 |
14 | This is a subset of TriStateType; no UNDEFINED value is provided.
15 | [Limits: 0 1 ]
16 | Description for mibCtl BooleanType 0 FALSE :
17 | False.
18 | Description for mibCtl BooleanType 1 TRUE :
19 | True.
20 | "
21 | ::= { _625k-MCSysScalars 17 }
22 |
23 |
24 |
25 | _625k-MCRebootBaseStation            OBJECT-TYPE
26 | SYNTAX                                INTEGER -- BooleanType
27 | | ACCESSMAX-ACCESS                    read-write -- REALLY: write-only
28 | | STATUS                               mandatoryCurrent
29 | DESCRIPTION
30 | | "Reboot base station.
31 | |
32 | | This is a write-only element; only a value of TRUE is valid.
33 | | All existing calls will be terminated abruptly.
34 | | All components of the base station will be reinitialized
35 | | according to the permanent contents of the Base Station database.
36 | | The base station may be incommunicado for a period of time.
37 | |
38 | | The reinitialization may be delayed by a few seconds to allow
39 | | for a clean shutdown.
40 | |
41 | | (From mibCtl ElementType 44 RebootBaseStation)
42 | | Description for mibCtl Type 16 BooleanType :
43 | | Truth value, 0=FALSE, 1=TRUE.
44 | |
45 | | This is a subset of TriStateType; no UNDEFINED value is provided.
46 | | [Limits: 0 1 ]
47 | | Description for mibCtl BooleanType 0 FALSE :
48 | | False.
49 | | Description for mibCtl BooleanType 1 TRUE :
50 | | True.
51 | | "
52 | | ::= { _625k-MCSysScalars 18 }
53 | |
54 | |
55 | |
56 | _625k-MCBSModelNumber                 OBJECT-TYPE
57 | SYNTAX                                OCTET STRING (SIZE(0..20)) -- TextType X 20
58 | | ACCESSMAX-ACCESS                    read-only
59 | | STATUS                               mandatoryCurrent
60 | DESCRIPTION
61 | | "Base Station model number.
62 | |
63 | | Base Station model number
64 | |
65 | | (From mibCtl ElementType 204 BSModelNumber)
66 | | Description for mibCtl Type 15 TextType :
67 | | ASCII or compatible text.
68 | | Type derived from mibCtl Type 12 OctetType :

```



```

1         8 bits of raw opaque data.
2         Derived from basic 8 bit word type.
3         "
4         ::= { _625k-MCSysScalars 19 }
5
6
7
8         _625k-MCBSManufactureDate          OBJECT-TYPE
9         SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
10        | ACCESSMAX-ACCESS          read-only
11        | STATUS          mandatoryCurrent
12        | DESCRIPTION
13        |     "Base Station manufacture date.
14        |
15        |     Base Station manufacture date
16        |
17        |     (From mibCtl ElementType 205 BSManufactureDate)
18        |     Description for mibCtl Type 15 TextType :
19        |     ASCII or compatible text.
20        |     Type derived from mibCtl Type 12 OctetType :
21        |     8 bits of raw opaque data.
22        |     Derived from basic 8 bit word type.
23        |     "
24        |     ::= { _625k-MCSysScalars 20 }
25
26
27
28        _625k-MCBSHardwareRevision          OBJECT-TYPE
29        SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
30        | ACCESSMAX-ACCESS          read-only
31        | STATUS          mandatoryCurrent
32        | DESCRIPTION
33        |     "Base Station hardware revision.
34        |
35        |     Base Station hardware revision
36        |
37        |     (From mibCtl ElementType 206 BSHardwareRevision)
38        |     Description for mibCtl Type 15 TextType :
39        |     ASCII or compatible text.
40        |     Type derived from mibCtl Type 12 OctetType :
41        |     8 bits of raw opaque data.
42        |     Derived from basic 8 bit word type.
43        |     "
44        |     ::= { _625k-MCSysScalars 21 }
45
46
47
48        _625k-MCMiscComponents              OBJECT IDENTIFIER
49        -- DESCRIPTION          "Miscellaneous Component"
50        ::= { IEEE802dot20-625k-MC-MIB 2 }
51
52
53
54        _625k-MCAntenna                     OBJECT IDENTIFIER
55        -- DESCRIPTION          "Antenna"
56        ::= { _625k-MCMiscComponents 1 }
57
58
59
60        _625k-MCAntennaTable                OBJECT-TYPE
61        SYNTAX SEQUENCE OF _625k-MCAntennaTableEntry
62        | ACCESSMAX-ACCESS          not-accessible
63        | STATUS          mandatoryCurrent
64        | DESCRIPTION          "Antenna Table"
65        | ::= { _625k-MCAntenna 1 }
66
67
68

```

```

1  _625k-MCAntennaTableEntry          OBJECT-TYPE
2  SYNTAX          _625k-MCAntennaTableEntry
3  ACCESSMAX-ACCESS      not-accessible
4  STATUS          mandatoryCurrent
5  DESCRIPTION     ""
6  INDEX { _625k-MCAntennaTableIndex }
7  ::= { _625k-MCAntennaTable 1 }
8
9  _625k-MCAntennaTableEntry ::= SEQUENCE {
10  _625k-MCAntennaTableIndex          INTEGER, -- AntennaAddressType
11  _625k-MCStateOfAntenna             INTEGER -- ComponentStateType
12  }
13
14
15
16  _625k-MCAntennaTableIndex          OBJECT-TYPE
17  SYNTAX          INTEGER -- AntennaAddressType
18  ACCESSMAX-ACCESS      read-only
19  STATUS          mandatoryCurrent
20  DESCRIPTION     "
21  Description for mibCtl Type 210 AntennaAddressType :
22  Component antenna address.
23  [Limits: 0 11 ]
24  Type derived from mibCtl Type 14 Unsigned32Type :
25  32 bit unsigned integer.
26  Type derived from mibCtl Type 11 Word32Type :
27  32 bits of raw opaque data.
28  Derived from basic 32 bit word type.
29  "
30  ::= { _625k-MCAntennaTableEntry 1 }
31
32
33
34  _625k-MCStateOfAntenna              OBJECT-TYPE
35  SYNTAX          INTEGER -- ComponentStateType
36  ACCESSMAX-ACCESS      read-only
37  STATUS          mandatoryCurrent
38  DESCRIPTION     "State of Antenna as a whole.
39
40
41
42  (From mibCtl ElementType 211 StateOfAntenna)
43  Description for mibCtl Type 71 ComponentStateType :
44  Component operational state.
45
46  A component begins in the Unknown state.
47  If not detected, it enters and remains in the NotPresent state.
48  If detected, it enters the Uninitialized state, from where it
49  may go to the Testing and Initializing states and then to the
50  Standby or Operating state depending upon permissions.
51  Due to loss of permissions or resources, it may revert from
52  the Operating state to the Standby state.
53  Due to failure or loss of permission, it may revert to the
54  Uninitialized state, perhaps by way of the ShuttingDown state
55  depending on the device.
56  From the Uninitialized state it may return to more advanced
57  states depending upon permissions.
58  In case of a waiting period before (again) initializing,
59  the component is considered to be Initializing.
60
61  Permissions include administrative permissions (from the
62  operator); excessive failure restrictions; etc.
63  Description for mibCtl ComponentStateType 0 Unknown :
64  Component state not known.
65  Description for mibCtl ComponentStateType 1 NotPresent :
66  Component is not present.
67  Description for mibCtl ComponentStateType 2 PowerOff :
68  Component is present but powered off.

```

```

1      Description for mibCtl ComponentStateType 3 Uninitialized :
2          Component is present but not in use.
3
4          The power on/off state of the component is not specified in
5          this case.
6      Description for mibCtl ComponentStateType 4 Testing :
7          Component is being tested.
8      Description for mibCtl ComponentStateType 5 Initializing :
9          Component is being initialized.
10     Description for mibCtl ComponentStateType 6 Ready :
11         Component is ready but not operating.
12     Description for mibCtl ComponentStateType 7 Operating :
13         Component is operating for normal use without restriction.
14
15         The component is either in actual use or may be used at any time,
16         without restriction.
17     Description for mibCtl ComponentStateType 8 Abandoned :
18         Component state is not the desired state due to excessive errors.
19
20         The component state is not that desired, and the Base Station
21         software has abandoned attempts to place the component in
22         the desired state.
23         The actual state of the component is undefined.
24         The Base Station software will resume attempting to place the
25         component in the desired state if the appropriate Reinitialize
26         action element is written with the correct value.
27         Also, the software may resume attempts under other conditions,
28         not all of which may be documented.
29     Description for mibCtl ComponentStateType 9 InitialSetUp :
30         Component is initial set up..
31
32         Initial set up state.
33     Description for mibCtl ComponentStateType 10 Degrading :
34         Component is degrading..
35
36         Degrading state.
37     Description for mibCtl ComponentStateType 11 Restriction :
38         Component is restriction..
39
40         Restriction state.
41     "
42     ::= { _625k-MCAntennaTableEntry 2 }
43
44
45
46     _625k-MCBSTemperatures                OBJECT IDENTIFIER
47     -- DESCRIPTION                        "BS Temperature"
48     ::= { _625k-MCMiscComponents 3 }
49
50
51
52     _625k-MCBSTemperature                OBJECT-TYPE
53     SYNTAX                               OCTET STRING (SIZE(0..4)) -- DegreesCelsiusType
54     ACCESSMAX-ACCESS                   read-only
55     STATUS                               mandatoryCurrent
56     DESCRIPTION
57         "The temperature of Base station (degrees Celsius).
58
59         The latest recorded temperature of a given BS.
60
61         (From mibCtl ElementType 536 BSTemperature)
62     Description for mibCtl Type 807 DegreesCelsiusType :
63         Temperature in degrees Celsius.
64     Type derived from mibCtl Type 15 TextType :
65         ASCII or compatible text.
66     Type derived from mibCtl Type 12 OctetType :
67         8 bits of raw opaque data.
68     Derived from basic 8 bit word type.

```

```

1      "
2      ::= { _625k-MCBSTemperatures 1 }
3
4
5
6      _625k-MCCableInfo          OBJECT IDENTIFIER
7      -- DESCRIPTION            "Cable Info"
8      ::= { _625k-MCMiscComponents 4 }
9
10
11
12     _625k-MCCableLossValueForLoCal          OBJECT-TYPE
13     SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
14     ACCESSMAX-ACCESS read-only
15     STATUS mandatoryCurrent
16     DESCRIPTION
17         "Value of cable loss for Local Oscilator.
18
19         Value of cable loss.
20
21         (From mibCtl ElementType 801 CableLossValueForLoCal)
22         Description for mibCtl Type 15 TextType :
23             ASCII or compatible text.
24         Type derived from mibCtl Type 12 OctetType :
25             8 bits of raw opaque data.
26         Derived from basic 8 bit word type.
27     "
28     ::= { _625k-MCCableInfo 1 }
29
30
31
32
33     _625k-MCAntCableTable          OBJECT-TYPE
34     SYNTAX SEQUENCE OF _625k-MCAntCableTableEntry
35     ACCESSMAX-ACCESS not-accessible
36     STATUS mandatoryCurrent
37     DESCRIPTION "Antenna Cable"
38     ::= { _625k-MCCableInfo 2 }
39
40
41
42     _625k-MCAntCableTableEntry          OBJECT-TYPE
43     SYNTAX _625k-MCAntCableTableEntry
44     ACCESSMAX-ACCESS not-accessible
45     STATUS mandatoryCurrent
46     DESCRIPTION ""
47     INDEX { _625k-MCAntCableTableIndex }
48     ::= { _625k-MCAntCableTable 1 }
49
50     _625k-MCAntCableTableEntry ::= SEQUENCE {
51         _625k-MCAntCableTableIndex          INTEGER, -- AntennaAddressType
52         _625k-MCCableLossValueForAntenna    OCTET STRING (SIZE(0..20)) --
53         TextType X 20
54     }
55
56
57
58     _625k-MCAntCableTableIndex          OBJECT-TYPE
59     SYNTAX INTEGER -- AntennaAddressType
60     ACCESSMAX-ACCESS read-only
61     STATUS mandatoryCurrent
62     DESCRIPTION "
63         Description for mibCtl Type 210 AntennaAddressType :
64         Component antenna address.
65         [Limits: 0 11 ]
66         Type derived from mibCtl Type 14 Unsigned32Type :
67         32 bit unsigned integer.
68         Type derived from mibCtl Type 11 Word32Type :

```

```

1         32 bits of raw opaque data.
2         Derived from basic 32 bit word type.
3         "
4         ::= { _625k-MCAntCableTableEntry 1 }
5
6
7
8         _625k-MCCableLossValueForAntenna          OBJECT-TYPE
9         SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
10        | ACCESSMAX-ACCESS read-only
11        | STATUS mandatoryCurrent
12        | DESCRIPTION
13        |     "Value of cable loss for Antenna.
14        |
15        |     Value of cable loss.
16        |
17        |     (From mibCtl ElementType 802 CableLossValueForAntenna)
18        |     Description for mibCtl Type 15 TextType :
19        |     ASCII or compatible text.
20        |     Type derived from mibCtl Type 12 OctetType :
21        |     8 bits of raw opaque data.
22        |     Derived from basic 8 bit word type.
23        |     "
24        | ::= { _625k-MCAntCableTableEntry 2 }
25
26
27
28
29        _625k-MCGPSCableTable                      OBJECT-TYPE
30        SYNTAX SEQUENCE OF _625k-MCGPSCableTableEntry
31        | ACCESSMAX-ACCESS not-accessible
32        | STATUS mandatoryCurrent
33        | DESCRIPTION "GPS Cable"
34        | ::= { _625k-MCCableInfo 10 }
35
36
37
38        _625k-MCGPSCableTableEntry                OBJECT-TYPE
39        SYNTAX _625k-MCGPSCableTableEntry
40        | ACCESSMAX-ACCESS not-accessible
41        | STATUS mandatoryCurrent
42        | DESCRIPTION ""
43        | INDEX { _625k-MCGPSCableTableIndex }
44        | ::= { _625k-MCGPSCableTable 1 }
45
46        _625k-MCGPSCableTableEntry ::= SEQUENCE {
47        |     _625k-MCGPSCableTableIndex          INTEGER, -- GpsAddressType
48        |     _625k-MCCableLengthForGps          OCTET STRING (SIZE(0..20)) --
49        |     TextType X 20
50        | }
51
52
53
54        _625k-MCGPSCableTableIndex                OBJECT-TYPE
55        SYNTAX INTEGER -- GpsAddressType
56        | ACCESSMAX-ACCESS read-only
57        | STATUS mandatoryCurrent
58        | DESCRIPTION
59        |     "
60        |     Description for mibCtl Type 209 GpsAddressType :
61        |     Base station GPS component address.
62        |
63        |     [Limits: 0 1 ]
64        |     Type derived from mibCtl Type 14 Unsigned32Type :
65        |     32 bit unsigned integer.
66        |     Type derived from mibCtl Type 11 Word32Type :
67        |     32 bits of raw opaque data.
68        |     Derived from basic 32 bit word type.

```

```

1      "
2      ::= { _625k-MCGPSCableTableEntry 1 }
3
4
5
6      _625k-MCCableLengthForGps          OBJECT-TYPE
7      SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
8      | ACCESSMAX-ACCESS          read-only
9      | STATUS          mandatoryCurrent
10     | DESCRIPTION
11     |     "Cable length for Gps.
12     |
13     |     This cable is used for calibration.
14     |
15     |     (From mibCtl ElementType 803 CableLengthForGps)
16     |     Description for mibCtl Type 15 TextType :
17     |     ASCII or compatible text.
18     |     Type derived from mibCtl Type 12 OctetType :
19     |     8 bits of raw opaque data.
20     |     Derived from basic 8 bit word type.
21     |     "
22     ::= { _625k-MCGPSCableTableEntry 2 }
23
24
25
26     _625k-MCGPS          OBJECT IDENTIFIER
27     -- DESCRIPTION          "GPS"
28     ::= { _625k-MCMiscComponents 6 }
29
30
31
32     _625k-MCGPSTable          OBJECT-TYPE
33     SYNTAX SEQUENCE OF _625k-MCGPSTableEntry
34     | ACCESSMAX-ACCESS          not-accessible
35     | STATUS          mandatoryCurrent
36     | DESCRIPTION          "GPS Table"
37     ::= { _625k-MCGPS 1 }
38
39
40
41     _625k-MCGPSTableEntry          OBJECT-TYPE
42     SYNTAX          _625k-MCGPSTableEntry
43     | ACCESSMAX-ACCESS          not-accessible
44     | STATUS          mandatoryCurrent
45     | DESCRIPTION          ""
46     | INDEX { _625k-MCGPSTableIndex }
47     ::= { _625k-MCGPSTable 1 }
48
49     _625k-MCGPSTableEntry ::= SEQUENCE {
50     |     _625k-MCGPSTableIndex          INTEGER, -- GpsAddressType
51     |     _625k-MCStateOfGps          INTEGER, -- ComponentStateType
52     |     _625k-MCGpsNumberOfSatelliteSeen INTEGERUnsigned32, -- Unsigned32Type
53     |     _625k-MCGpsIndication          INTEGER, -- IndicationType
54     |     _625k-MCGpsSerialNumber          OCTET STRING (SIZE(0..20)) --
55     |     TextType X 20
56     |     }
57
58
59
60     _625k-MCGPSTableIndex          OBJECT-TYPE
61     SYNTAX          INTEGER -- GpsAddressType
62     | ACCESSMAX-ACCESS          read-only
63     | STATUS          mandatoryCurrent
64     | DESCRIPTION          "
65     |     Description for mibCtl Type 209 GpsAddressType :
66     |     Base station GPS component address.
67
68

```

```

1         [Limits: 0 1 ]
2         Type derived from mibCtl Type 14 Unsigned32Type :
3         32 bit unsigned integer.
4         Type derived from mibCtl Type 11 Word32Type :
5         32 bits of raw opaque data.
6         Derived from basic 32 bit word type.
7     "
8     ::= { _625k-MCGPSTableEntry 1 }
9
10
11
12     _625k-MCStateOfGps                                OBJECT-TYPE
13     SYNTAX      INTEGER -- ComponentStateType
14     ACCESSMAX-ACCESS      read-only
15     STATUS      mandatoryCurrent
16     DESCRIPTION
17         "GPS state.
18
19         The state of the GPS (Global Positioning System)
20         on the active local oscillator unit
21
22         (From mibCtl ElementType 420 StateOfGps)
23         Description for mibCtl Type 71 ComponentStateType :
24         Component operational state.
25
26         A component begins in the Unknown state.
27         If not detected, it enters and remains in the NotPresent state.
28         If detected, it enters the Uninitialized state, from where it
29         may go to the Testing and Initializing states and then to the
30         Standby or Operating state depending upon permissions.
31         Due to loss of permissions or resources, it may revert from
32         the Operating state to the Standby state.
33         Due to failure or loss of permission, it may revert to the
34         Uninitialized state, perhaps by way of the ShuttingDown state
35         depending on the device.
36         From the Uninitialized state it may return to more advanced
37         states depending upon permissions.
38         In case of a waiting period before (again) initializing,
39         the component is considered to be Initializing.
40
41         Permissions include administrative permissions (from the
42         operator); excessive failure restrictions; etc.
43         Description for mibCtl ComponentStateType 0 Unknown :
44         Component state not known.
45         Description for mibCtl ComponentStateType 1 NotPresent :
46         Component is not present.
47         Description for mibCtl ComponentStateType 2 PowerOff :
48         Component is present but powered off.
49         Description for mibCtl ComponentStateType 3 Uninitialized :
50         Component is present but not in use.
51
52         The power on/off state of the component is not specified in
53         this case.
54         Description for mibCtl ComponentStateType 4 Testing :
55         Component is being tested.
56         Description for mibCtl ComponentStateType 5 Initializing :
57         Component is being initialized.
58         Description for mibCtl ComponentStateType 6 Ready :
59         Component is ready but not operating.
60         Description for mibCtl ComponentStateType 7 Operating :
61         Component is operating for normal use without restriction.
62
63         The component is either in actual use or may be used at any time,
64         without restriction.
65         Description for mibCtl ComponentStateType 8 Abandoned :
66         Component state is not the desired state due to excessive errors.
67
68         The component state is not that desired, and the Base Station

```

```

1       software has abandoned attempts to place the component in
2       the desired state.
3       The actual state of the component is undefined.
4       The Base Station software will resume attempting to place the
5       component in the desired state if the appropriate Reinitialize
6       action element is written with the correct value.
7       Also, the software may resume attempts under other conditions,
8       not all of which may be documented.
9       Description for mibCtl ComponentStateType 9 InitialSetUp :
10      Component is initial set up..
11
12      Initial set up state.
13      Description for mibCtl ComponentStateType 10 Degrading :
14      Component is degrading..
15
16      Degrading state.
17      Description for mibCtl ComponentStateType 11 Restriction :
18      Component is restriction..
19
20      Restriction state.
21      "
22      ::= { _625k-MCGPSTableEntry 2 }
23
24
25
26      _625k-MCGpsNumberOfSatelliteSeen          OBJECT-TYPE
27      SYNTAX          INTEGER Unsigned32 -- Unsigned32Type
28      ACCESSMAX-ACCESS          read-only
29      STATUS          mandatoryCurrent
30      DESCRIPTION
31      "Number of satellites seen by GPS.
32
33      The number of satellites seen by the GPS (Global Positioning System)
34      on the active local oscillator unit
35
36      (From mibCtl ElementType 421 GpsNumberOfSatelliteSeen)
37      Description for mibCtl Type 14 Unsigned32Type :
38      32 bit unsigned integer.
39      Type derived from mibCtl Type 11 Word32Type :
40      32 bits of raw opaque data.
41      Derived from basic 32 bit word type.
42      "
43      ::= { _625k-MCGPSTableEntry 3 }
44
45
46
47      _625k-MCGpsIndication                      OBJECT-TYPE
48      SYNTAX          INTEGER -- IndicationType
49      ACCESSMAX-ACCESS          read-only
50      STATUS          mandatoryCurrent
51      DESCRIPTION
52      "Status of GPS indicator.
53
54
55      (From mibCtl ElementType 530 GpsIndication)
56      Description for mibCtl Type 230 IndicationType :
57      Hardware indication status (LEDs).
58
59
60
61      Description for mibCtl IndicationType 0 Off :
62      Off.
63      Description for mibCtl IndicationType 1 Amber :
64      Amber.
65      Description for mibCtl IndicationType 2 Red :
66      Red.
67      Description for mibCtl IndicationType 3 Green :
68      Green.

```



```

1      Description for mibCtl IndicationType 4 NotPresent :
2          Not present.
3      "
4      ::= { _625k-MCGPSTableEntry 4 }
5
6
7
8      _625k-MCGpsSerialNumber          OBJECT-TYPE
9      SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
10     | ACCESSMAX-ACCESS          read-only
11     | STATUS          mandatoryCurrent
12     | DESCRIPTION
13     |     "GPS serial number text.
14     |
15     |     Factory set uniquely for each component.
16     |
17     |     (From mibCtl ElementType 570 GpsSerialNumber)
18     |     Description for mibCtl Type 15 TextType :
19     |     ASCII or compatible text.
20     |     Type derived from mibCtl Type 12 OctetType :
21     |     8 bits of raw opaque data.
22     |     Derived from basic 8 bit word type.
23     |     "
24     ::= { _625k-MCGPSTableEntry 5 }
25
26
27
28     _625k-MCPowerAmplifier            OBJECT IDENTIFIER
29     -- DESCRIPTION          "Power Amplifier"
30     ::= { _625k-MCMiscComponents 8 }
31
32
33
34
35     _625k-MCPAUnitTable              OBJECT-TYPE
36     SYNTAX SEQUENCE OF _625k-MCPAUnitTableEntry
37     | ACCESSMAX-ACCESS          not-accessible
38     | STATUS          mandatoryCurrent
39     | DESCRIPTION          "PA Table"
40     ::= { _625k-MCPowerAmplifier 1 }
41
42
43
44     _625k-MCPAUnitTableEntry        OBJECT-TYPE
45     SYNTAX          _625k-MCPAUnitTableEntry
46     | ACCESSMAX-ACCESS          not-accessible
47     | STATUS          mandatoryCurrent
48     | DESCRIPTION          ""
49     | INDEX          { _625k-MCPAUnitTableIndex }
50     ::= { _625k-MCPAUnitTable 1 }
51
52     _625k-MCPAUnitTableEntry ::= SEQUENCE {
53         _625k-MCPAUnitTableIndex      INTEGER, -- PAUnitAddressType
54         _625k-MCRebootPAUnit          INTEGER, -- BooleanType
55         _625k-MCStateOfPAUnit         INTEGER, -- ComponentStateType
56         _625k-MCPAUnitIndication      INTEGER, -- IndicationType
57         _625k-MCPAUnitSerialNumber    OCTET STRING (SIZE(0..20)), --
58         TextType X 20
59         _625k-MCPAUnitModelNumber     OCTET STRING (SIZE(0..20)), --
60         TextType X 20
61         _625k-MCPAUnitManufactureDate OCTET STRING (SIZE(0..20)), --
62         TextType X 20
63         _625k-MCPAUnitHardwareRevision OCTET STRING (SIZE(0..20)), --
64         TextType X 20
65         _625k-MCPAUnitManufactureID   OCTET STRING (SIZE(0..20)), --
66         TextType X 20
67         _625k-MCPAUnitTemperature     OCTET STRING (SIZE(0..4)) --
68         DegreesCelsiusType

```

```

1     }
2
3
4
5     _625k-MCPAUnitTableIndex                                OBJECT-TYPE
6     SYNTAX          INTEGER -- PAUnitAddressType
7     ACCESSMAX-ACCESS      read-only
8     STATUS          mandatoryCurrent
9     DESCRIPTION    "
10        Description for mibCtl Type 207 PAUnitAddressType :
11        Base station power amplifier component unit address.
12
13        A power amplifier unit address is a subset of Base Station
14        component addresses,
15        restricted to power amplifier components only.
16        Power amplifiers boost radio frequency
17        signal levels.
18        [Limits: 0 3 ]
19        Type derived from mibCtl Type 14 Unsigned32Type :
20        32 bit unsigned integer.
21        Type derived from mibCtl Type 11 Word32Type :
22        32 bits of raw opaque data.
23        Derived from basic 32 bit word type.
24    "
25    ::= { _625k-MCPAUnitTableEntry 1 }
26
27
28
29    _625k-MCRebootPAUnit                                    OBJECT-TYPE
30    SYNTAX          INTEGER -- BooleanType
31    ACCESSMAX-ACCESS      read-write -- REALLY: write-only
32    STATUS          mandatoryCurrent
33    DESCRIPTION    "Action to reboot a PA unit.
34
35
36        This is a write-only element; only a value of TRUE is valid.
37
38        (From mibCtl ElementType 506 RebootPAUnit)
39        Description for mibCtl Type 16 BooleanType :
40        Truth value, 0=FALSE, 1=TRUE.
41
42        This is a subset of TriStateType; no UNDEFINED value is provided.
43        [Limits: 0 1 ]
44        Description for mibCtl BooleanType 0 FALSE :
45        False.
46        Description for mibCtl BooleanType 1 TRUE :
47        True.
48    "
49    ::= { _625k-MCPAUnitTableEntry 2 }
50
51
52
53    _625k-MCStateOfPAUnit                                    OBJECT-TYPE
54    SYNTAX          INTEGER -- ComponentStateType
55    ACCESSMAX-ACCESS      read-only
56    STATUS          mandatoryCurrent
57    DESCRIPTION    "State of PA as a whole.
58
59
60
61        (From mibCtl ElementType 212 StateOfPAUnit)
62        Description for mibCtl Type 71 ComponentStateType :
63        Component operational state.
64
65        A component begins in the Unknown state.
66        If not detected, it enters and remains in the NotPresent state.
67        If detected, it enters the Uninitialized state, from where it
68        may go to the Testing and Initializing states and then to the

```

```

1      Standby or Operating state depending upon permissions.
2      Due to loss of permissions or resources, it may revert from
3      the Operating state to the Standby state.
4      Due to failure or loss of permission, it may revert to the
5      Uninitialized state, perhaps by way of the ShuttingDown state
6      depending on the device.
7      From the Uninitialized state it may return to more advanced
8      states depending upon permissions.
9      In case of a waiting period before (again) initializing,
10     the component is considered to be Initializing.
11
12     Permissions include administrative permissions (from the
13     operator); excessive failure restrictions; etc.
14     Description for mibCtl ComponentStateType 0 Unknown :
15     Component state not known.
16     Description for mibCtl ComponentStateType 1 NotPresent :
17     Component is not present.
18     Description for mibCtl ComponentStateType 2 PowerOff :
19     Component is present but powered off.
20     Description for mibCtl ComponentStateType 3 Uninitialized :
21     Component is present but not in use.
22
23     The power on/off state of the component is not specified in
24     this case.
25     Description for mibCtl ComponentStateType 4 Testing :
26     Component is being tested.
27     Description for mibCtl ComponentStateType 5 Initializing :
28     Component is being initialized.
29     Description for mibCtl ComponentStateType 6 Ready :
30     Component is ready but not operating.
31     Description for mibCtl ComponentStateType 7 Operating :
32     Component is operating for normal use without restriction.
33
34     The component is either in actual use or may be used at any time,
35     without restriction.
36     Description for mibCtl ComponentStateType 8 Abandoned :
37     Component state is not the desired state due to excessive errors.
38
39     The component state is not that desired, and the Base Station
40     software has abandoned attempts to place the component in
41     the desired state.
42     The actual state of the component is undefined.
43     The Base Station software will resume attempting to place the
44     component in the desired state if the appropriate Reinitialize
45     action element is written with the correct value.
46     Also, the software may resume attempts under other conditions,
47     not all of which may be documented.
48     Description for mibCtl ComponentStateType 9 InitialSetUp :
49     Component is initial set up..
50
51     Initial set up state.
52     Description for mibCtl ComponentStateType 10 Degrading :
53     Component is degrading..
54
55     Degrading state.
56     Description for mibCtl ComponentStateType 11 Restriction :
57     Component is restriction..
58
59     Restriction state.
60     "
61     ::= { _625k-MCPAUnitTableEntry 3 }
62
63
64
65     _625k-MCPAUnitIndication          OBJECT-TYPE
66     SYNTAX          INTEGER -- IndicationType
67     ACCESS          ACCESSMAX-ACCESS read-only
68     STATUS          mandatoryCurrent

```

```

1 DESCRIPTION
2     "Status of PA Unit indicator.
3
4
5
6     (From mibCtl ElementType 526 PAUnitIndication)
7     Description for mibCtl Type 230 IndicationType :
8         Hardware indication status (LEDs).
9
10
11     Description for mibCtl IndicationType 0 Off :
12         Off.
13     Description for mibCtl IndicationType 1 Amber :
14         Amber.
15     Description for mibCtl IndicationType 2 Red :
16         Red.
17     Description for mibCtl IndicationType 3 Green :
18         Green.
19     Description for mibCtl IndicationType 4 NotPresent :
20         Not present.
21     "
22 ::= { _625k-MCPAUnitTableEntry 4 }
23
24
25
26 _625k-MCPAUnitSerialNumber          OBJECT-TYPE
27     SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
28     ACCESSMAX-ACCESS          read-only
29     STATUS          mandatoryCurrent
30     DESCRIPTION
31         "PA unit serial number text.
32
33         Factory set uniquely for each component.
34
35         (From mibCtl ElementType 560 PAUnitSerialNumber)
36     Description for mibCtl Type 15 TextType :
37         ASCII or compatible text.
38     Type derived from mibCtl Type 12 OctetType :
39         8 bits of raw opaque data.
40     Derived from basic 8 bit word type.
41     "
42 ::= { _625k-MCPAUnitTableEntry 5 }
43
44
45
46 _625k-MCPAUnitModelNumber          OBJECT-TYPE
47     SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20
48     ACCESSMAX-ACCESS          read-only
49     STATUS          mandatoryCurrent
50     DESCRIPTION
51         "PA unit model number.
52
53         Factory set with description of component type, including
54         the major revision level.
55
56         (From mibCtl ElementType 561 PAUnitModelNumber)
57     Description for mibCtl Type 15 TextType :
58         ASCII or compatible text.
59     Type derived from mibCtl Type 12 OctetType :
60         8 bits of raw opaque data.
61     Derived from basic 8 bit word type.
62     "
63 ::= { _625k-MCPAUnitTableEntry 6 }
64
65
66
67 _625k-MCPAUnitManufactureDate      OBJECT-TYPE
68     SYNTAX          OCTET STRING (SIZE(0..20)) -- TextType X 20

```

```

1 | ACCESSMAX-ACCESS read-only
2 | STATUS mandatoryCurrent
3 | DESCRIPTION
4 | "PA unit manufacture date.
5 |
6 | Factory set to month and date of manufacture date of the module.
7 |
8 | (From mibCtl ElementType 562 PAUnitManufactureDate)
9 | Description for mibCtl Type 15 TextType :
10 | ASCII or compatible text.
11 | Type derived from mibCtl Type 12 OctetType :
12 | 8 bits of raw opaque data.
13 | Derived from basic 8 bit word type.
14 | "
15 | ::= { _625k-MCPAUnitTableEntry 7 }
16 |
17 |
18 |
19 | _625k-MCPAUnitHardwareRevision OBJECT-TYPE
20 | SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
21 | | ACCESSMAX-ACCESS read-only
22 | | STATUS mandatoryCurrent
23 | | DESCRIPTION
24 | | "PA unit hardware revision name.
25 | |
26 | | Set at the factory to indicate the minor hardware revision
27 | | level of the module.
28 | |
29 | | (From mibCtl ElementType 563 PAUnitHardwareRevision)
30 | | Description for mibCtl Type 15 TextType :
31 | | ASCII or compatible text.
32 | | Type derived from mibCtl Type 12 OctetType :
33 | | 8 bits of raw opaque data.
34 | | Derived from basic 8 bit word type.
35 | | "
36 | | ::= { _625k-MCPAUnitTableEntry 8 }
37 | |
38 | |
39 | |
40 | | _625k-MCPAUnitManufactureID OBJECT-TYPE
41 | | SYNTAX OCTET STRING (SIZE(0..20)) -- TextType X 20
42 | | | ACCESSMAX-ACCESS read-only
43 | | | STATUS mandatoryCurrent
44 | | | DESCRIPTION
45 | | | "PA unit manufacture ID.
46 | | |
47 | | |
48 | | | (From mibCtl ElementType 564 PAUnitManufactureID)
49 | | | Description for mibCtl Type 15 TextType :
50 | | | ASCII or compatible text.
51 | | | Type derived from mibCtl Type 12 OctetType :
52 | | | 8 bits of raw opaque data.
53 | | | Derived from basic 8 bit word type.
54 | | | "
55 | | | ::= { _625k-MCPAUnitTableEntry 9 }
56 | | |
57 | | |
58 | | |
59 | | |
60 | | | _625k-MCPAUnitTemperature OBJECT-TYPE
61 | | | SYNTAX OCTET STRING (SIZE(0..4)) -- DegreesCelsiusType
62 | | | | ACCESSMAX-ACCESS read-only
63 | | | | STATUS mandatoryCurrent
64 | | | | DESCRIPTION
65 | | | | "The temperature of PAUnit (degrees Celsius).
66 | | | |
67 | | | | The latest recorded temperature of a given PAUnit.
68 | | | |

```

```

1      (From mibCtl ElementType 539 PAUnitTemperature)
2      Description for mibCtl Type 807 DegreesCelsiusType :
3          Temperature in degrees Celsius.
4      Type derived from mibCtl Type 15 TextType :
5          ASCII or compatible text.
6      Type derived from mibCtl Type 12 OctetType :
7          8 bits of raw opaque data.
8      Derived from basic 8 bit word type.
9      "
10     ::= { _625k-MCPAUnitTableEntry 10 }
11
12
13
14
15     _625k-MCPAModuleTable          OBJECT-TYPE
16     SYNTAX SEQUENCE OF _625k-MCPAModuleTableEntry
17     ACCESSMAX-ACCESS              not-accessible
18     STATUS mandatoryCurrent
19     DESCRIPTION "PA Module Table"
20     ::= { _625k-MCPowerAmplifier 2 }
21
22
23
24     _625k-MCPAModuleTableEntry      OBJECT-TYPE
25     SYNTAX _625k-MCPAModuleTableEntry
26     ACCESSMAX-ACCESS              not-accessible
27     STATUS mandatoryCurrent
28     DESCRIPTION ""
29     INDEX { _625k-MCPAModuleTableIndex1, _625k-MCPAModuleTableIndex2 }
30     ::= { _625k-MCPAModuleTable 1 }
31
32     _625k-MCPAModuleTableEntry ::= SEQUENCE {
33         _625k-MCPAModuleTableIndex1  INTEGER, -- PAUnitAddressType
34         _625k-MCPAModuleTableIndex2  INTEGER -- PAModuleAddressType
35     }
36
37
38
39     _625k-MCPAModuleTableIndex1     OBJECT-TYPE
40     SYNTAX INTEGER -- PAUnitAddressType
41     ACCESSMAX-ACCESS              read-only
42     STATUS mandatoryCurrent
43     DESCRIPTION "
44         Description for mibCtl Type 207 PAUnitAddressType :
45         Base station power amplifier component unit address.
46
47         A power amplifier unit address is a subset of Base Station
48         component addresses,
49         restricted to power amplifier components only.
50         Power amplifiers boost radio frequency
51         signal levels.
52         [Limits: 0 3 ]
53         Type derived from mibCtl Type 14 Unsigned32Type :
54         32 bit unsigned integer.
55         Type derived from mibCtl Type 11 Word32Type :
56         32 bits of raw opaque data.
57         Derived from basic 32 bit word type.
58     "
59     ::= { _625k-MCPAModuleTableEntry 1 }
60
61
62
63     _625k-MCPAModuleTableIndex2     OBJECT-TYPE
64     SYNTAX INTEGER -- PAModuleAddressType
65     ACCESSMAX-ACCESS              read-only
66     STATUS mandatoryCurrent
67     DESCRIPTION "
68         Description for mibCtl Type 208 PAModuleAddressType :

```

```

1         Base station power amplifier component module address.
2
3         A power amplifier module address is a subset of Base Station
4         component addresses,
5         [Limits: 0 2 ]
6         Type derived from mibCtl Type 14 Unsigned32Type :
7         32 bit unsigned integer.
8         Type derived from mibCtl Type 11 Word32Type :
9         32 bits of raw opaque data.
10        Derived from basic 32 bit word type.
11        "
12        ::= { _625k-MCPAModuleTableEntry 2 }
13
14
15
16        _625k-MCPowerSupply                OBJECT IDENTIFIER
17        -- DESCRIPTION                    "Power supply"
18        ::= { _625k-MCMiscComponents 9 }
19
20
21
22
23        _625k-MCPowerSupplyTable            OBJECT-TYPE
24        SYNTAX SEQUENCE OF _625k-MCPowerSupplyTableEntry
25        ACCESSMAX-ACCESS                    not-accessible
26        STATUS                            mandatoryCurrent
27        DESCRIPTION                        "Power supply Table"
28        ::= { _625k-MCPowerSupply 1 }
29
30
31
32        _625k-MCPowerSupplyTableEntry      OBJECT-TYPE
33        SYNTAX                            _625k-MCPowerSupplyTableEntry
34        ACCESSMAX-ACCESS                    not-accessible
35        STATUS                            mandatoryCurrent
36        DESCRIPTION                        ""
37        INDEX { _625k-MCPowerSupplyTableIndex }
38        ::= { _625k-MCPowerSupplyTable 1 }
39
40        _625k-MCPowerSupplyTableEntry ::= SEQUENCE {
41        _625k-MCPowerSupplyTableIndex      INTEGER, -- PowerAddressType
42        _625k-MCStateOfPowerSupply         INTEGER, -- ComponentStateType
43        _625k-MCPowerSupplyIndication     INTEGER -- IndicationType
44        }
45
46
47
48        _625k-MCPowerSupplyTableIndex      OBJECT-TYPE
49        SYNTAX                            INTEGER -- PowerAddressType
50        ACCESSMAX-ACCESS                    read-only
51        STATUS                            mandatoryCurrent
52        DESCRIPTION                        "
53        Description for mibCtl Type 211 PowerAddressType :
54        Component power supply address.
55        [Limits: 0 2 ]
56        Type derived from mibCtl Type 14 Unsigned32Type :
57        32 bit unsigned integer.
58        Type derived from mibCtl Type 11 Word32Type :
59        32 bits of raw opaque data.
60        Derived from basic 32 bit word type.
61        "
62        ::= { _625k-MCPowerSupplyTableEntry 1 }
63
64
65
66        _625k-MCStateOfPowerSupply         OBJECT-TYPE
67        SYNTAX                            INTEGER -- ComponentStateType
68        ACCESSMAX-ACCESS                    read-only

```

1 | STATUS mandatoryCurrent
2 | DESCRIPTION
3 | "State of Power supply as a whole.
4 |
5 |
6 | (From mibCtl ElementType 213 StateOfPowerSupply)
7 | Description for mibCtl Type 71 ComponentStateType :
8 | Component operational state.
9 |
10 | A component begins in the Unknown state.
11 | If not detected, it enters and remains in the NotPresent state.
12 | If detected, it enters the Uninitialized state, from where it
13 | may go to the Testing and Initializing states and then to the
14 | Standby or Operating state depending upon permissions.
15 | Due to loss of permissions or resources, it may revert from
16 | the Operating state to the Standby state.
17 | Due to failure or loss of permission, it may revert to the
18 | Uninitialized state, perhaps by way of the ShuttingDown state
19 | depending on the device.
20 | From the Uninitialized state it may return to more advanced
21 | states depending upon permissions.
22 | In case of a waiting period before (again) initializing,
23 | the component is considered to be Initializing.
24 |
25 | Permissions include administrative permissions (from the
26 | operator); excessive failure restrictions; etc.
27 | Description for mibCtl ComponentStateType 0 Unknown :
28 | Component state not known.
29 | Description for mibCtl ComponentStateType 1 NotPresent :
30 | Component is not present.
31 | Description for mibCtl ComponentStateType 2 PowerOff :
32 | Component is present but powered off.
33 | Description for mibCtl ComponentStateType 3 Uninitialized :
34 | Component is present but not in use.
35 |
36 | The power on/off state of the component is not specified in
37 | this case.
38 | Description for mibCtl ComponentStateType 4 Testing :
39 | Component is being tested.
40 | Description for mibCtl ComponentStateType 5 Initializing :
41 | Component is being initialized.
42 | Description for mibCtl ComponentStateType 6 Ready :
43 | Component is ready but not operating.
44 | Description for mibCtl ComponentStateType 7 Operating :
45 | Component is operating for normal use without restriction.
46 |
47 | The component is either in actual use or may be used at any time,
48 | without restriction.
49 | Description for mibCtl ComponentStateType 8 Abandoned :
50 | Component state is not the desired state due to excessive errors.
51 |
52 | The component state is not that desired, and the Base Station
53 | software has abandoned attempts to place the component in
54 | the desired state.
55 | The actual state of the component is undefined.
56 | The Base Station software will resume attempting to place the
57 | component in the desired state if the appropriate Reinitialize
58 | action element is written with the correct value.
59 | Also, the software may resume attempts under other conditions,
60 | not all of which may be documented.
61 | Description for mibCtl ComponentStateType 9 InitialSetUp :
62 | Component is initial set up..
63 |
64 | Initial set up state.
65 | Description for mibCtl ComponentStateType 10 Degrading :
66 | Component is degrading..
67 |
68 | Degrading state.


```

1      Description for mibCtl ComponentStateType 11 Restriction :
2          Component is restriction..
3
4          Restriction state.
5      "
6      ::= { _625k-MCPowerSupplyTableEntry 2 }
7
8
9
10     _625k-MCPowerSupplyIndication          OBJECT-TYPE
11     SYNTAX          INTEGER -- IndicationType
12     ACCESSMAX-ACCESS          read-only
13     STATUS          mandatoryCurrent
14     DESCRIPTION
15         "Status of Power Supply indicator.
16
17
18         (From mibCtl ElementType 527 PowerSupplyIndication)
19     Description for mibCtl Type 230 IndicationType :
20         Hardware indication status (LEDs).
21
22
23     Description for mibCtl IndicationType 0 Off :
24         Off.
25     Description for mibCtl IndicationType 1 Amber :
26         Amber.
27     Description for mibCtl IndicationType 2 Red :
28         Red.
29     Description for mibCtl IndicationType 3 Green :
30         Green.
31     Description for mibCtl IndicationType 4 NotPresent :
32         Not present.
33     "
34     ::= { _625k-MCPowerSupplyTableEntry 3 }
35
36
37
38
39     END

```

1

2